Robustifying Machine Learning through Weakening Supervision

Julian Lienen

July 31st, 2023
Version: Print

PADERBORN
UNIVERSITY

Faculty of Computer Science,
Electrical Engineering and Mathematics
Warburger Straße 100
33098 Paderborn

Dissertation

In partial fulfillment of the requirements for the academic degree of
Doctor rerum naturalium (Dr. rer. nat.)

Robustifying Machine Learning through Weakening Supervision

Julian Lienen

1. Reviewer Prof. Dr. Eyke Hüllermeier
Institute of Informatics
Ludwig Maximilian University of Munich

2. Reviewer Dr. Sébastien Destercke, HDR
Centre national de la recherche scientifique (CNRS)
Université de Technologie de Compiègne

3. Reviewer Prof. Dr. Ralph Ewerth
L3S Research Center
Leibniz University Hannover
Technische Informationsbibliothek (TIB) Hannover

Supervisor Prof. Dr. Eyke Hüllermeier

July 31st, 2023

Julian Lienen

Robustifying Machine Learning through

Weakening Supervision

Dissertation, July 31st, 2023

Reviewers: Prof. Dr. Eyke Hüllermeier
Dr. Sébastien Destercke, HDR
Prof. Dr. Ralph Ewerth

Supervisor: Prof. Dr. Eyke Hüllermeier

Paderborn University

Faculty of Computer Science, Electrical Engineering and Mathematics

Department of Computer Science

Warburger Straße 100

33098 Paderborn

Abstract

In machine learning applications, the quality of solutions is typically determined by the interplay between the model class, the learning algorithm including the criterion to be optimized, and the training data used for learning. For the latter, obtaining data with precise target information is the gold standard, enabling (fully) supervised learning. However, mere precision is no guarantee of its correctness. Rather, in practice, labels can often only be acquired with various forms of imperfection, e.g., by observing corrupted annotations or sensor signals afflicted with noise, potentially biasing the induction of predictors. As a result, learning models are usually adapted to the training data for robustness against distortions, such as by adopting certain inductive biases of the learning algorithm, aiming at mitigating negative influences of imperfect data on the learning process. Nevertheless, this often comes along with increased model complexity and impaired generalization.

In this thesis, we take an alternative perspective on this matter by advocating for the (re-)modeling of the training labels themselves to achieve model robustness, while keeping adaptations of the learning model to the training data to a minimum. To this end, we propose to deliberately weaken hitherto precise supervision in order to achieve a more faithful and reliable representation of the beliefs about the underlying ground truth. This way, although being less precise, the label information is rendered more correct, attenuating the detrimental influence of potential biases in the training data.

We consider two forms of imprecisiation, namely by replacing precise labels with *(fuzzy) sets* and *ordinal relations*. The former is realized by modeling (fuzzy) supersets that augment single labels with additional plausible candidates for representing the ground truth. Combined with generalized risk minimization as a methodology to learn from set-valued targets in the realm of so-called superset learning, this approach allows for modulating the influence of individual, potentially harmful instances within the overall learning process. As demonstrated for applications in the domains of computer vision and knowledge graphs, it proves to be an effective means for robustifying models against biases. For the second form of weakening, here dedicated to robust multi-target regression, we suggest to model weakened supervision for instances in the form of relative comparisons of their underlying numerical ground truths. This promotes the applicability of learning methods to

problems where obtaining precise labels is challenging, enabling one to employ models from the field of preference learning for regression tasks. As illustrated for the problem of monocular depth estimation in images, favorable performance in terms of generalization to different datasets and increased cost-effectiveness of the training data acquisition can be achieved.

Zusammenfassung

Die Qualität von Anwendungen des maschinellen Lernens wird typischerweise durch das Zusammenspiel zwischen der Modellklasse, dem Lernalgorithmus einschließlich des zu optimierenden Kriteriums, sowie den Trainingsdaten bestimmt. Für Letzteres stellt die Akquise von präzisen Zielgrößen, die (vollständig) überwachte Lernmethoden ermöglichen, den Goldstandard dar. Jedoch garantiert Präzision der Labels selbst nicht notwendigerweise deren Korrektheit. Vielmehr lassen sich Zielgrößen in der Praxis oftmals nur mit diversen Formen der Imperfektion erfassen, etwa als korrumpierte Annotationen oder mit Verrauschungen behafteten Sensorwerten, wodurch induzierte Prädiktoren potenziell verzerrt werden. Infolgedessen werden Lernmodelle zur Robustheit gegenüber Verzerrungen in der Regel an die Trainingsdaten angepasst, etwa durch die Annahme gewisser *inductive biases* des Lernalgorithmus, die darauf abzielen, negative Einflüsse imperfekter Daten auf den Lernprozess abzuschwächen. Dies geht jedoch oft mit einer erhöhten Modellkomplexität und eingeschränkter Generalisierbarkeit einher.

In der vorliegenden Arbeit nehmen wir eine alternative Perspektive ein, indem wir die Robustheit von Modellen durch die (Re-)Modellierung der Labels im Trainingsdatensatz verbessern, während Anpassungen des Lernmodells an die Trainingsdaten auf ein Minimum reduziert werden. Zu diesem Zweck schlagen wir eine gezielte Abschwächung von zuvor präzisen Labels vor, um eine wahrheitsgetreuere und zuverlässigere Repräsentation der Annahmen über den wahren Wert der Zielgröße zu erreichen. Auf diese Weise wird die Zielgrößeninformation, obwohl weniger präzise, korrekter dargestellt, wodurch der negative Einfluss potenzieller Verzerrungen in den Trainingsdaten abgeschwächt wird.

Wir betrachten zwei Formen der Impräzisierung: Ersetzen von präzisen Labels durch *(unscharfe) Mengen* und durch *Ordnungsrelationen*. Ersteres wird durch die Modellierung von (unscharfen) Obermengen realisiert, die die ursprüngliche Zielgrößeninformation durch zusätzliche plausible Kandidaten für die Repräsentation der Grundwahrheit erweitert. In Kombination mit der verallgemeinerten empirischen Risikominimierung als eine Methode aus dem Bereich des *superset learning* zum Lernen aus mengenwertigen Zielgrößen, ermöglicht dieser Ansatz die Modulierung des Einflusses von individuellen, potenziell verzerrenden Instanzen im gesamten Lernprozess. Wie an Anwendungen aus den Bereichen der Computer Vision und Wissensgraphen demonstriert wird, erweisen sich Methoden dieser Art als wirksames Mittel zur Steigerung der Robustheit von induzierten Prädiktoren gegenüber Verzerrungen. Für die zweite Form der abgeschwächten Überwachung, die besonderen Bezug auf die multivariate Regression nimmt, schlagen wir vor, die Modellierung der Zielgrößen in Form von relativen Vergleichen ihrer zugrunde liegenden numerischen Grundwahrheiten zu realisieren. Dies verbessert die Anwendbarkeit von Lernmethoden auf Probleme, in denen die Akquise von präzisen Signalen schwierig ist, und ermöglicht den Einsatz von Modellen aus dem Bereich des Präferenzlernens für Regressionsprobleme. Wie am Beispiel des Problems der monokularen Tiefenschätzung in Bildern gezeigt wird, kann eine Verbesserung in Bezug auf die Generalisierbarkeit auf verschiedene Datensätze und eine erhöhte Kosteneffizienz der Trainingsdatenerfassung erzielt werden.

Acknowledgement

Before delving into the depths of my thesis, I seize this opportunity to express my heartfelt appreciation to individuals who have played an outstanding role in my journey. The completion of my doctoral thesis would not have been possible without the unwavering and invaluable support of these people.

First and foremost, I would like to express my sincerest gratefulness to my advisor Eyke for granting me the opportunity to work under his guidance and supporting my pursuit of a doctoral degree. His exceptional mentorship has shaped me into a genuine researcher, teaching me to focus on details that matter, as well as to develop a scientific language to convey my (mostly chaotic) thoughts. Moreover, especially in later stages of my endeavor, Eyke provided me with collaborative research projects that significantly broadened my horizons both professionally and personally. I am immensely thankful for the cherished experiences we shared, which will continue to inspire me in the future.

My deep appreciation also goes to the members of my committee, namely Ralph, Sébastien, Matthias, Prof. Christian Scheideler, and again Eyke, for generously agreeing to review this thesis. Ralph, I am particularly grateful to you for being one of my entry points into research, sharing your experiences, guiding me through my first research project, and for our productive collaborations. Special thanks also to Nils! Sébastien, your support has fostered a sense of belonging to a research community, which gave me a feeling that my research matters. Thank you for warmly welcoming me to our field and for being an ongoing source of inspiration. Matthias and Prof. Scheideler, I genuinely appreciate your willingness to be part of my committee, despite our paths crossing less frequently.

Research thrives on collaboration and team spirit. Throughout my years at the university, I have had the privilege of being supported by an outstanding team, which has served as a backbone of my journey. Whether within the former Intelligent Systems and Machine Learning group at Paderborn University or the recently established Artificial Intelligence and Machine Learning group at the Ludwig Maximilian University, I have always felt part of something greater. Working with such talented and pleasant peers has been an absolute joy! I extend my thanks to Elisabeth, whose unconditional help has been precious in nearly every aspect of life. Without her, I would have faced even greater challenges. I also want to acknowledge Marcel, who not only proofread my thesis, but also played a pivotal role in introducing me to research through his supervision of our project group and my master's thesis. Additionally, I express my gratitude to Björn for his repetitive proofreading and the pleasant time we spent together in the office. Of course, I cannot forget the "ZM2 gang" (as a fellow perfectly coined it) – it was a delightful time! I would also like to extend my gratitude to all the other collaboration partners with whom I had the opportunity to work. Together, we have achieved commendable projects of which we can be proud.

During my studies, I took a slight detour into industry, where I had the fortune of meeting inspiring individuals who left a lasting impression on me. Special thanks go to Omar, whose exceptional leadership skills enabled me to have a highly enriching and formative research internship. I am also grateful to Andreas, from whom I learned a great deal both professionally and methodologically. And, of course, I want to express my gratitude to Satya and the entire CSNLP team at Amazon. My time with you in Berlin was truly marvelous!

Finally, my closest personal environment deserves at least as much thanks. If it hadn't been for you, I would never have made it this far. First of all, I would like to thank my second half, Milena, who has surely longed for this day. Thank you for enduring stressful and agonizing days with calmness and encouragement. Furthermore, a great deal of credit for my accomplishments goes to my parents and siblings who raised me (to be continued...), instilled in me a constant curiosity that is essential for a researcher, and always supported me unconditionally. I truly appreciate it. I would also like to extend my thanks to Milena's family for their continuous support along my journey. Last but certainly not least, I want to express my gratitude to my best friends, who have never left my side and on whom I have always been able to rely. Thank you for being there to catch me when I fall, and for the genuine connection we share, free of pretense.

Contents

1 Introduction

Machine learning is today's key enabler of artificial intelligence, gradually permeating every aspect of life. Fueled by the ubiquitous availability of big data, models like ChatGPT [@Ope22] are finding applications across society. Increasingly, software driven by machine learning is taking on cognitive tasks that have previously been limited to humans only, such as in transportation [Ela+22], manufacturing [Kot+21], education [NPL23] and medicine [Rah+21]. Current progress does not suggest that this will change in the near future, but rather that it will intensify.

From a technical point of view, the success of machine learning applications depends on the interplay between the model class, the learning algorithm and, finally, the data on which it is trained. Their proper integration is crucial to achieve adequate generalization capabilities of learned models. In particular, the first two dimensions have received a great deal of attention in the research community, resulting in a large number of proposed methods. For the latter, the quality of the observed data is a decisive factor, not least for the label information. Although methodological advances have led to generalizations of machine learning models to deal with arbitrary manifestations of training data, e.g., to deal with uncertain data in the field of *weakly-supervised learning* [Zho17], the acquisition of data labeled with precise target information has been considered as the gold standard, enabling to apply (fully-)supervised learning methods [AML12]. Nevertheless, beyond tapping new data sources, the modeling of the data itself as an additional degree of freedom has received rather little attention.

But even when precise labels are readily available, mere precision does not inherently ensure the informative value of such supervision during the process of model training. Rather, the correctness of the label information is another crucial quality characteristic that cannot be overlooked, as it plays a vital role in ensuring that the optimization is not misled. More specifically, precise target information may be subject to various forms of imperfection, e.g., due to vagueness or even corruption in the labeling procedure of the training data. Consider image classification as an example, where labels are typically given in the form of "hard" labels that are transformed into a degenerate probability distribution for the purpose of learning probabilistic models, e.g., in the context of a maximum likelihood estimation. Arguably, it is not difficult to construct examples where a deterministic relationship between image features and classes to be determined is not appropriate, such as in the case of blurred or hidden parts of the scene, or in the case of classes that are difficult to distinguish. As another example, precise numerical labels could be obtained from a sensor employed in a manufacturing process to induce a regressive model. However, because the sensor is subject to the noise inherent in the sensing process, it may only produce accurate signals within a tolerance range, so the single value can only be considered to be reliably located within that range, but not necessarily the exact value collected, leading to biased monitoring and hence distorted supervision for training.

Without adequate countermeasures, learning from such data usually results in the induction of biased predictors. As the optimization goal of a model training typically consists of reproducing the label information at hand, the distortion of the data is replicated by the resulting model, which is problematic. The poor predictive quality of such models can lead to harmful behavior when used in real applications. For instance, data biased toward certain ethnic groups could lead to underrepresented groups being treated inappropriately, raising concerns about fairness [Meh+22], or models may be unable to realistically quantify their confidence, resulting in poor model calibration [Guo+17]. Therefore, additional means to address such concerns need to be established.

In order to mitigate the negative influence of training data distortions on the learner, i.e., to promote its *robustness* against undesired biases, a common scheme is to make the model class and its optimization procedure *compatible* with the data at hand. That is, adjustments to the former, e.g., to inductive biases of models [Bax00], can be designed to compensate for their detrimental effect on quality dimensions to be optimized, such as the ability to generalize. Related to the examples mentioned above, a probabilistic image classifier could be regularized to reduce the degree of determinism in the label information [Sze+16], or predictions could be attenuated by rescaling and thus recalibrating a fitted model [WFZ21]. Regression models are often corrected by a bias term, consequently allowing to eliminate systematic biases [KK04]. Nevertheless, such adaptations to the learning model typically add complexity and are tailored to the training data at hand, hence potentially compromising the applicability to data that emerge with different forms of distortions than previously being observed.

In this thesis, we advocate a change of perspective, in which we do not focus on adapting the model class or its optimization procedure to the data, but the other way around. Instead, we propose to *re-model the data* by intentionally adjusting the label information for robustness, thereby making the given data compatible with an existing model class and objective. In that course, we distinguish two forms of a deliberate imprecisiation of hitherto precise labels, namely the replacement by *(fuzzy) supersets* and *ordinal relations*. While the former augments the original label information by further plausible candidates for being the ground truth, ordinal relations allow for abstracting from precise numerical information in favor of relative comparisons. Together with methods from the field of weakly-supervised learning, both forms can address the aforementioned problems with models trained on biased data, such as improving the calibration of a model, keeping the changes to the learning model to a minimum. At the same time, our developed methodologies promote a more faithful way of modeling knowledge, serving as a basis for an awareness of uncertainty already in the training information. This thesis, the structure and content of which are outlined in the following sections, provides a comprehensive collection of our contributions in this regard.

1.1 Thesis Structure

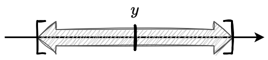
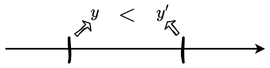
Since we are introducing data modeling as an alternative to the adaptation of the learning model for robustifying machine learning models against biases in the data, Chapter 2 discusses preliminaries of the starting point, namely supervised learning, where models are fit to precisely labeled data. In this context, we briefly discuss challenges in this regime of dealing with imperfect annotations, as well as common tweaks to learning models that address the resulting difficulties. As our developed methodology suggests the construction of weaker forms of label information with increased expressiveness, generalizations of learning methods discussed in Chapter 2 need to be introduced. To this end, we provide an overview of the field of weakly-supervised learning in Chapter 3, with a particular focus on problem settings closely related to our work.

After presenting our main contributions in Chapters 4 to 10, supplementary material to the individual contributions is provided in Appendices A to F, we conclude this thesis with a comprehensive summary of the key findings and accomplishments of this thesis in Chapter 11. Additionally, we identify prevailing challenges and outline potential avenues for future research that are in close connection with the addressed problems. By this, our aim is to inspire future investigations that build upon the foundation we have established, focusing on the integration of modeling training labels as a paradigm to foster compatibility between learning models and data.

1.2 Contributions

The contributions in this work can be divided into two parts based on the different forms of label information weakening. In Table 1.1, an overview is provided, indicating the mapping of each contribution to its respective type of weakening, as well as the addressed problem type. The remainder of this section offers a more detailed overview of the collected works.

Tab. 1.1.: Overview of the individual contributions (I)-(VII) to the two distinguished forms of weakening the label information y (and y') per problem type.

	(Fuzzy) Set-Valued Weakening	Ordinal Weakening
		
(Prob.) Classification	(I), (II), (III), (IV), (V) [Ch. 4, 5, 6, 7, 8]	
Regression	(V), (VI) [Ch. 8, 9]	(VII) [Ch. 10]

(Fuzzy) Set-Valued Weakening

The first part of the thesis comprises methods that contribute to the setting of *(fuzzy) set-valued weakening*. This imprecisiation approach is realized by maintaining sets of plausible candidates besides the initially observed label, which we refer to as *supersets*. Combined with generalized learning methodologies from the field of *superset learning* (cf. Section 3.2), our proposed modeling allows for modulating the influence of individual instances within the overall optimization context. This way, it provides a means to attenuate the harming effects of biasing instances within the training data, promoting the robustness of induced models. The contributions to this part read as follows.

(I) Julian Lienen and Eyke Hüllermeier. "From Label Smoothing to Label Relaxation". In: *Proc. of the 35th AAAI Conference on Artificial Intelligence, AAAI,* *the 33rd Conference on Innovative Applications of Artificial Intelligence, IAAI, the 11th Symposium on Educational Advances in Artificial Intelligence, EAAI, February 2-9, virtual*. AAAI Press, 2021, pp. 8583–8591.

(II) Julian Lienen and Eyke Hüllermeier. "Credal Self-Supervised Learning". In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-14, virtual*. Curran Associates, Inc., 2021, pp. 14370–14382.

(III) Julian Lienen, Caglar Demir, and Eyke Hüllermeier. "Conformal Credal Self-Supervised Learning". In: *CoRR* abs/2205.15239 (2022). Accepted at the 12th Symposium on Conformal and Probabilistic Prediction with Applications, COPA 2023, September 13-15, Limassol, Cyprus.

(IV) Julian Lienen and Eyke Hüllermeier. "Mitigating Label Noise through Data Ambiguation". In: *CoRR* abs/2305.13764 (2023).

(V) Julian Lienen and Eyke Hüllermeier. "Instance Weighting through Data Imprecisiation". In: *Int. J. Approx. Reason.* 134 (2021), pp. 1–14.

(VI) Julian Lienen, Nils Nommensen, Ralph Ewerth, and Eyke Hüllermeier. "Robust Regression for Monocular Depth Estimation". In: *Proc. of the 13th Asian Conference on Machine Learning, ACML, November 17-19, virtual*. Vol. 157. Proc. of Machine Learning Research. PMLR, 2021, pp. 1001–1016.

Contribution (I): From Label Smoothing to Label Relaxation. In the first publication presented in this regard, we depart from conventional supervised classification algorithms that assume precise probability distributions as target information. Committing to a single distribution, which is unlikely to match the exact underlying ground truth, misleads the optimization and results in poorly calibrated models. To address this issue, we propose the framework of *label relaxation* (LR) in Chapter 4 as a data modeling technique in the form of a deliberate imprecisiation of probabilistic labels. This relaxation is realized by replacing single target distributions by sets of probability distributions, also known as *credal sets*, which encompass neighboring distributions close to the initial target. Together with a risk minimization formulation that generalizes to set-valued labels, a learner gains flexibility in identifying the distribution within the constructed (credal) supersets that appear most plausible in the overall optimization context. Compared to previous attempts at achieving more realistic target distributions for probabilistic classification, particularly in the realm of deep learning, the LR framework provides a general yet effective means to improve the model calibration, leading to less biased and thus more robust models.

Contributions (II) and (III): (Conformal) Credal Self-Supervised Learning. In Chapters 5 and 6, we build on the previously introduced form of imprecise label information to employ LR in a self-labeling approach for semi-supervised learning. More precisely, *credal self-supervised learning* (CSSL) replaces a (precise) probabilistic self-labeling mechanism for the unlabeled data by transforming model predictions into pseudo-labels in the form of credal sets that allow for expressing the uncertainty of a model in its predictions. As a result, a more cautious, uncertainty-aware, yet reliable pseudo-labeling is achieved. While the construction of such target sets follows a heuristic, rather ad-hoc strategy incorporating meta information such as the prediction history in CSSL, we advance this idea in *conformal credal self-supervised learning* (CCSSL) by embedding it within a more rigorous algorithmic framework. Specifically, the model that predicts the pseudo-labels is transformed into a conformal predictor [SV08], an uncertainty quantification procedure that wraps pointwise predictors, allowing to attain conformal credal pseudo-labels with theoretical validity guarantees with respect to their inclusion of the ground truth (but unknown) target distribution. Both methods signify a paradigmatic alteration in the methodology for semi-supervised learning, as they shift the complexity of learning model adaptation, e.g., to cope with overconfident and thus unreliable precise probabilistic pseudo-labels, to the modeling of label information, thereby leveraging the expressiveness of richer knowledge representations for pseudo-labels.

Contribution (IV): Mitigating Label Noise through Data Ambiguation. Beyond the two settings described before, we have studied LR also in the context of label noise in supervised learning, where observed (precise) label information is assumed to be corrupted in form of mislabels in the training dataset. In Chapter 7, we propose a methodology that deliberately ambiguates target information by maintaining sets of plausible labels, we call this approach *robust data ambiguation* (RDA), for which the model confidently predicts a label different from the observed, potentially corrupted, training label. Motivated by the training dynamics of today's overparameterized neural networks, RDA has shown its ability to detect mislabeling in early training epochs, allowing for the suppression of deleterious memorization effects as training progresses.

Contribution (V): Instance Weighting through Data Imprecisiation. While label relaxation and its adaptations have focused on probabilistic classification, and have tailored the target modeling to this setting, we also introduce a more generalized form in Chapter 8 for the purpose of instance weighting. As for label relaxation, we suggest to model the target information in the form of a (potentially fuzzy) superset, which can be used as a means to control the weighting of the respective data sample in the overall training loss to be minimized. This is achieved through a similar generalized learning methodology in the context of superset learning as in LR. Together with appropriate heuristics to determine the degree of imprecisiation to modulate the influence of individual instances, this idea has shown promising results in terms of robustness and generalization performance when combined with support vector machines, both for learning with label noise and for semi-supervised learning.

Contribution (VI): Robust Regression for Monocular Depth Estimation. As another computer vision related problem of interest, relevant in many practical applications, contribution (VI) focuses on the domain of *monocular depth estimation* (MDE) in images. Here, given only a single monocular image, a pixel-wise prediction of the distance from the camera to the scene projection is sought. Training information for this task is typically provided in the form of images with pixel-wise numerical depth information produced by an appropriate sensor. However, such sensors are typically prone to noisy measurements, or limited in their operational capability (e.g., due to range limitations), violating statistical assumptions underlying common models used for MDE. Consequently, an adjustment to the precise numerical target information is advisable to achieve a more faithful target modeling for preventing misguidance during the model training, just in line with the motivation of our other contributions. In Chapter 9, we apply the idea of instance weighting through data imprecisiation as in contribution (V) to the problem of depth regression, where we studied the shortcomings of common depth sensors to develop a domain-adapted imprecisiation strategy, leading to more robust metric depth predictions compared to commonly employed learning models in this field.

Ordinal Weakening

The second part of the thesis addresses weakening based on *ordinal relations* as another form of imprecisiated supervision, which steps up to lower the requirements on the generation of training labels. The key idea is to abstract from numerical estimates by relative comparisons between instances with respect to their underlying ground truth, leading again to a weaker form of supervision. The following work contributes to this part.

(VII) Julian Lienen, Eyke Hüllermeier, Ralph Ewerth, and Nils Nommensen. “Monocular Depth Estimation via Listwise Ranking Using the Plackett-Luce Model”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 19-25, virtual*. Computer Vision Foundation / IEEE, 2021, pp. 14595–14604.

Contribution (VII): Monocular Depth Estimation via Listwise Ranking Using the Plackett-Luce Model. In Chapter 10, illustrated for the problem of monocular depth estimation, we suggest to treat multi-target regression as a ranking problem, where hitherto metric data is weakened by replacing it with relative comparisons between individual pixels with respect to their distance to the camera. By adopting this strategy, we achieve a more robust and reliable form of supervision that compensates for misalignment and bias issues associated with numerical labels, overall resulting in a cost-effective way to acquire labels. Methodologically, we interpret relative comparisons as *preferences*, thereby establishing compatibility between metric data for depth regression and techniques employed in the field of *preference learning*. Our approach relies on the so-called *Plackett-Luce model*, a probabilistic model on rankings, which not only allows for predicting relative depth orders between pixels, but also enables the extraction of shift-invariant metric depth predictions from its internal parameters. Notably, our method demonstrates competitive depth ranking performance even in "zero-shot" settings, while also yielding accurate metric predictions.

Supervised Learning: Harnessing Precise Labels

2

The effectiveness of machine learning applications hinges on a triumvirate of essential components: the model class, the optimization procedure, and the training data. The efficacy of the integration of these components ultimately determines the success of the method. While this thesis endeavors to address the harmonization between them by focusing on the modeling of training labels for an overall compatibility of the system, it diverges from conventional approaches of adapting learning models to data, leaving the data unchanged. In this regard, this preliminary chapter presents a point of departure from conventional approaches. Here, we commence in Section 2.1 by formally introducing the problem of supervised learning as one of the principal learning paradigms in machine learning. Subsequently, the discussion delves into the two most prevalent problem types therein, namely classification and regression, which we discuss in Sections 2.2 and 2.3, respectively. In this course, we highlight challenges associated with data imperfections, as well as adaptations to model classes and learning algorithms to mitigate such obstacles. Finally, the chapter concludes with a presentation of instance weighting as a general technique for controlling the influence of individual instances in supervised learning in Section 2.4.

2.1 Problem Formulation

In the context of predictive machine learning, a problem space is typically characterized by its *feature* and *target space* $\mathcal{X}$ and $\mathcal{Y}$, respectively, between which an unknown ground truth mapping $g : \mathcal{X} \longrightarrow \mathcal{Y}$ is assumed, which allows to infer from an instance $\boldsymbol{x} \in \mathcal{X}$ an outcome $y \in \mathcal{Y}$. The goal of supervised learning [AML12] is to select a *model* h, synonymously also referred to as *hypothesis* or *predictor*, from a *model class* (or *hypothesis space*) $\mathcal{H} \subset \mathcal{Y}^{\mathcal{X}} = \{h : \mathcal{X} \longrightarrow \mathcal{Y}\}$ based on data samples of the form $(\boldsymbol{x}, y) \in \mathcal{X} \times \mathcal{Y}$, such that g is approximated by h as well as possible.

We call a set consisting of N such data samples a *training dataset* of length N and write

$$\mathcal{D}_N = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N \subset \mathcal{X} \times \mathcal{Y} \,. \tag{2.1}$$

The optimization procedure in a supervised learning context is then specified within the context of a *learning algorithm* $\mathcal{A} : \mathbb{D} \longrightarrow \mathcal{H}$, which maps datasets $\mathcal{D}_N \in \mathbb{D}$ in the space of possible datasets $\mathbb{D}$ to hypotheses $h \in \mathcal{H}$. Following [AML12], we refer to the conjunction of $\mathcal{A}$ and $\mathcal{H}$ as *learning model*.

Typically, g is assumed to exhibit a non-deterministic relationship, capturing the common presence of noisy interrelationships observed in reality. This assumption considers all instances $(\boldsymbol{x}, y) \in \mathcal{D}_N$ as independent and identically distributed (i.i.d.) samples drawn from random variables X and Y, which follow an underlying joint probability distribution $p^* \in \mathbb{P}(\mathcal{X} \times \mathcal{Y})$ on $\mathcal{X} \times \mathcal{Y}$, i.e., $(X, Y) \sim p^*$. Here, $\mathbb{P}(\cdot)$ denotes the space of all probability distributions on a measurable space. The quality of h approximating the relationships in p^* is then usually measured in terms of a *loss function* $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \longrightarrow \mathbb{R}$, which compares a ground truth label y with the prediction of the model $h(\boldsymbol{x})$. Together with the probabilistic interpretation of the problem in terms of p^*, the expected loss over $\mathcal{X} \times \mathcal{Y}$ is captured by the so-called *risk* $\mathcal{R}(h, p^*)$ of h, which is defined as

$$\mathcal{R}(h, p^*) := \int_{\mathcal{X} \times \mathcal{Y}} \mathcal{L}\left(y, h(\boldsymbol{x})\right) \mathrm{d}p^*(\boldsymbol{x}, y) \,. \tag{2.2}$$

Typically, learning algorithms $\mathcal{A}$ then aim to identify the hypothesis $h^* \in \mathcal{H}$ with minimal risk $\mathcal{R}$, i.e.,

$$h^* \in \operatorname*{argmin}_{h \in \mathcal{H}} \mathcal{R}(h, p^*) \,. \tag{2.3}$$

In practice, just as g, the joint distribution p^* is unknown and, hence, $\mathcal{R}$ cannot be computed. Instead, an approximation of $\mathcal{R}$ is put into place for the sake of optimization on the training data $\mathcal{D}_N$, leading to the so-called *empirical risk* $\mathcal{R}_{\text{emp}}$ [Vap91]:

$$\mathcal{R}_{\text{emp}}(h, \mathcal{D}_N) := \frac{1}{N} \sum\nolimits_{(\boldsymbol{x}, y) \in \mathcal{D}_N} \mathcal{L}\left(y, h(\boldsymbol{x})\right) \tag{2.4}$$

Using this as an optimization criterion, the *empirical risk minimization* (ERM) procedure $\mathcal{A}_{\text{emp}}$ returns the minimizer

$$h^*_{\text{emp}} \in \mathcal{A}_{\text{emp}}(\mathcal{D}_N) := \operatorname*{argmin}_{h \in \mathcal{H}} \mathcal{R}_{\text{emp}}(h, \mathcal{D}_N) \,. \tag{2.5}$$

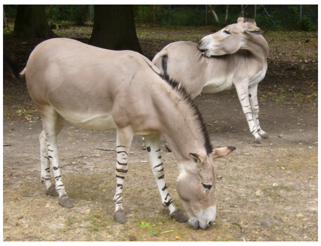
In the field of statistical learning theory [Vap95], theoretical works have investigated under which conditions h^*_{emp} is a satisfiable solution to Eq. (2.2), e.g., by analyzing the consistency of $\mathcal{R}_{\text{emp}}(h^*_{\text{emp}}, \cdot)$ to converge to the actual risk $\mathcal{R}(h^*_{\text{emp}}, \cdot)$ as the dataset size N increases [Vap91]. To this end, hypothesis spaces $\mathcal{H}$ and learning algorithms $\mathcal{A}$ are typically characterized by their structural expressiveness and underlying prior assumptions, thus preventing to fit random concepts by guiding principles of the learning model. Beyond ERM in form of $\mathcal{A}_{\text{emp}}$, also other methods have been formulated, such as structural risk minimization [Vap91] which balances the training loss and the complexity of a model in its optimization formulation, but which is out of scope of this thesis.

2.2 Classification

As an ubiquitous problem in many machine learning applications, *classification* problems assume a target space $\mathcal{Y}$ that is given by a discrete set of $K \geq 2$ classes $\mathcal{Y} = \{y_1, y_2, \ldots, y_K\}$, which indicate subpopulations in $\mathcal{X}$ sharing a unifying semantical category. In such cases, training datasets $\mathcal{D}_N$ typically comprise class labels $y \in \mathcal{Y}$ as target information accompanying instances $\boldsymbol{x} \in \mathcal{X}$. For target spaces with $K = 2$ and $K > 2$ classes, we refer to *binary* and *multi-class* classification, respectively. In addition to the standard classification settings, further variations of the problem exist, such as assuming a particular structure of classes in $\mathcal{Y}$, e.g., ordinal classification [Gut+16], which we will not discuss further within the scope of this thesis.

(a) `Domestic horse`

(b) `Persian onager`



(c) `Somali wild ass`

As an example for binary classification, consider emails represented by their textual content to be classified into either `spam` or `non-spam` as an example. In this case, a dataset $\mathcal{D}_N$ may contain email texts in the feature space of all possible mail texts, along with the corresponding spam indication by a human annotator. In the context of multi-class classification, a task could for instance entail classifying images that depict animals belonging to the Equidae genus, based on their species. Section 2.2 presents exemplary samples of such a dataset, here images labeled as `domestic` `horse`, `Persian onager` or `Somali wild ass`. Throughout the forthcoming preliminary chapters in this thesis, we will refer to this latter example in the case of classification problems.

There exists a plethora of methods approaching this problem, which, among others, range from simple linear models such as logistic regression [Cox58], to kernel-based methods [HSS08] including support vector machines (SVMs) [BGV92] to complex overparameterized models in the regime of deep learning [KSH12; He+16; Vas+17; Bro+20], which have emerged in the past decades thanks to their generalization capabilities and facilitated by an increased affordability and accessibility of high-performance computing resources. Due to the vast number of proposed supervised classification methods, we shall focus in the rest of this section on a particular modeling that is often adopted by models in this regime, namely *probabilistic classification* [HTF09].

Probabilistic Classification

Many of today's deep neural network classifiers, such as those considered in our work described in Chapters 4 to 7, generalize ordinary classification with observational ground truth relations in $\mathcal{Y}^{\mathcal{X}}$ in a probabilistic way, namely that hypotheses are modeled as conditional probability distributions instead of deterministic mappings from $\mathcal{X}$ to $\mathcal{Y}$. More precisely, the non-deterministic nature of p^* is explicitly captured in the prediction by considering models in $\mathcal{H}_{\text{pc}} \subset \mathbb{P}(\mathcal{Y})^{\mathcal{X}} = \{h : \mathcal{X} \longrightarrow \mathbb{P}(\mathcal{Y})\}$, where $\mathbb{P}(\mathcal{Y})$ denotes the space of probability distributions over $\mathcal{Y}$. Hence, classifiers $h \in \mathcal{H}_{\text{pc}}$ do not predict a class $y \in \mathcal{Y}$, but associate each y with a probability score conditioned on the sample features $\boldsymbol{x} \in \mathcal{X}$, thereby reflecting the stochastic nature of p^* more properly. An underlying assumption is that each instance $\boldsymbol{x} \in \mathcal{X}$ is associated with its conditional class distribution $p^*(\cdot \,|\, \boldsymbol{x}) \in \mathbb{P}(\mathcal{Y})$ instead of a deterministic label y as ground truth. Provided this form of ground truth supervision, the risk of a model h can be quantified by means of a probabilistic loss $\mathcal{L}_{\text{prob}} : \mathbb{P}(\mathcal{Y}) \times \mathbb{P}(\mathcal{Y}) \longrightarrow \mathbb{R}$. To emphasize that such hypotheses h are in fact probability distributions, we will denote models in $\mathcal{H}_{\text{pc}}$ by $\widehat{p}$, such that the predicted conditional class distribution is written as $\widehat{p}(\cdot \,|\, \boldsymbol{x})$, where $\widehat{p}(y \,|\, \boldsymbol{x})$ is the probability that y is the true outcome associated with $\boldsymbol{x}$.

Ideally, p^* is given directly as training information. However, as noted above, labels in training datasets such as in Eq. (2.1) are typically observed as deterministic labels

y, not in the form of a distribution as would be required by a probabilistic loss $\mathcal{L}_{\text{prob}}$ as specified above.[1] A common way to overcome this issue is, e.g., to enable learning by maximizing the likelihood

$$p^*_{\text{MLE}} \in \operatorname*{argmax}_{\widehat{p} \in \mathcal{H}_{\text{pc}}} \prod_{i=1}^{N} \widehat{p}\,(y_i \,|\, \boldsymbol{x}_i) \tag{2.6}$$

of observed data in $\mathcal{D}_N$, to transform the deterministic labels $y \in \mathcal{Y}$ of instances $(\boldsymbol{x}, y) \in \mathcal{D}_N$ to degenerate surrogates $p_y \in \mathbb{P}(\mathcal{Y})$ with $p_y(y \,|\, \boldsymbol{x}) = 1$ and $p_y(y' \,|\, \boldsymbol{x}) = 0$ for $y' \neq y$ [BW87; SLF88]. To learn from targets of this form, a particularly popular choice for $\mathcal{L}_{\text{prob}}$ is the cross-entropy (CE) loss, which is given by

$$\mathcal{L}_{\text{CE}}(p_y, \widehat{p}) := -\sum\nolimits_{y' \in \mathcal{Y}} p_y(y') \log \widehat{p}(y') \,, \tag{2.7}$$

with degenerate distributions p_y being employed as target information (for simplicity, we drop the dependence of p_y and $\widehat{p}$ on $\boldsymbol{x}$ here). It can be shown that the minimizer p^*_{emp} of $\mathcal{R}_{\text{emp}}$ with respect to $\mathcal{L}_{\text{CE}}$ is equivalent to the maximum likelihood solution p^*_{MLE} of Eq. (2.6). The vast majority of recently proposed probabilistic classification models have adopted this optimization scheme, rendering it a de-facto standard choice in recently proposed models [GBC16]. Nevertheless, other probabilistic loss formulations have also been proposed in this regard, most notably the mean-squared error [HB21] and variations of the cross-entropy loss [Kor+21].

Challenges and Learning Model Adaptations

Although this way of using deterministic labels in a probabilistic way seems to be justified from an optimization point of view, it can be questioned in light of the non-deterministic modeling of the relationship between $\mathcal{X}$ and $\mathcal{Y}$. Limitations in the explanatory power of the feature space $\mathcal{X}$ may lead to instances with similar feature representations that are different in reality, thus exhibiting aleatoric uncertainty [HW21]. Especially in the presence of conflicting class information in such cases, degenerate target distributions p_y suggest a degree of determinism that seems unjustified, thereby biasing the model in an undesirable way. Of particular relevance in the presence of label noise, where the observed training label y is corrupted, the

[1] In fact, although rather unusual, there are certain settings where labels are provided in the form of probabilistic *confidences* [INS18; Cao+21], thus directly reflecting p^* without a deterministic surrogate. Alternatives may include enriching the training labels with additional quality information, such as estimating the uncertainty in the dataset [NJC21].

extremity of p_y being reproduced by the learner $\widehat{p}$ actually exacerbates the problems described.

In the regime of overparameterized models, such as deep neural networks that comprise many more parameters than data points from which to learn, the flexibility of these models leads to serious practical problems in case degenerate and thus deterministic labels are employed as supervision. When trained with $\mathcal{L}_{\text{CE}}$, overparameterized models tend to be overconfident, i.e., to predict higher probabilistic scores than the likelihood that the prediction actually matches the ground truth label. This is detrimental to the calibration of models [Guo+17; Sil+23] and creates room for unwanted biases to manifest. Last but not least, this leads to ethical concerns, e.g., regarding the fairness of models toward underrepresented groups [Ple+17; Meh+22].

To address these problems, several learning model adaptations have been proposed to ensure compatibility with the degenerate probabilistic modeling of the data, such that the severity of learning from the extremity of the labels is counteracted in a robust manner. Most notably, regularization approaches are often used to mitigate the negative effects of learning from biased data. Applied methods include for instance penalties augmenting the loss function [Sze+16; Per+17; Kor+21], loss rescaling [Lin+17], and model ensembling techniques such as Dropout [Sri+14] or DropConnect [Wan+13]. Furthermore, many ideas have been suggested to specifically tune the calibration, including explicit methods that use some of the training data for calibration [KMF17; Guo+17], but also implicit approaches without additional data [MKH19; WFZ21]. When faced with label noise in the training data [Nat+13], the learning model can be effectively adapted by increasing the robustness of the loss function [GKS17; Wan+19b; ZSL21], noise-suppressing regularizers [Liu+20a], architectural changes [Liu+22b] or an adjusted learning procedure, e.g., by means of semi-supervised learning techniques [Jia+18; Che+21].

As a particularly popular regularization method, label smoothing (LS) [Sze+16] has found its way into many learning models. It can be formulated as a loss function $\mathcal{L}_{\text{LS}}$, which is given by

$$\mathcal{L}_{\text{LS}}(p_y, \widehat{p}) := (1-\alpha)\mathcal{L}_{\text{CE}}(p_y, \widehat{p}) + \alpha\mathcal{L}_{\text{CE}}(u, \widehat{p}) \,, \tag{2.8}$$

where $u \in \mathbb{P}(\mathcal{Y})$ is a fixed distribution and the parameter $\alpha \in (0, 1]$ controls the regularization strength. This loss has proven to be robust to biased data with respect to calibration [MKH19] and noise [Luk+20]. In its original proposal, it

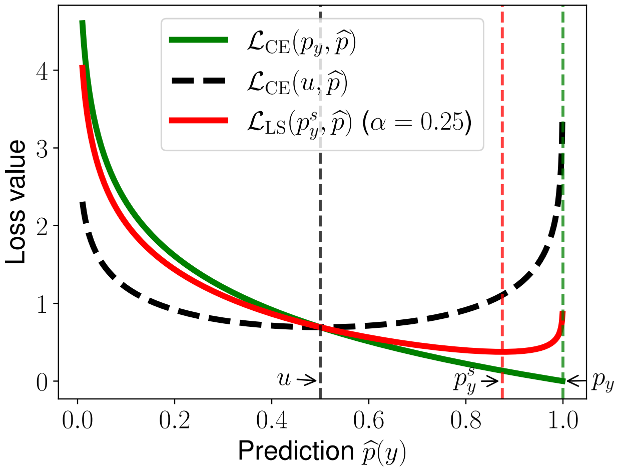


Fig. 2.2.: The loss $\mathcal{L}_{\text{CE}}$ and its regularized form $\mathcal{L}_{\text{LS}}$ for a binary classification problem with degenerate targets p_y, the uniform distribution u and a predictor $\widehat{p}$.

has been suggested to set u to the uniform distribution over $\mathcal{Y}$ with α being fixed for all instances, however, also other options are possible [KA20; LDB20; Gho+21; LCZ22; Var+23]. By adding the regularization penalty $\mathcal{L}_{\text{CE}}(u, \widehat{p})$, the learner is not incentivized to perfectly match p_y, but to match a less extreme surrogate distribution $p_y^s \in \mathbb{P}(\mathcal{Y})$. This distribution p_y^s "smoothens" p_y by taking a probability mass fraction α from $p_y(y)$ and distributing it over all classes in $\mathcal{Y}$ (e.g., uniformly if u is the uniform distribution in Eq. (2.8)).[2] As a result, which is illustrated in Fig. 2.2, $\mathcal{L}_{\text{LS}}$ penalizes predictions $\widehat{p}$ that are too close to a predicted probability of 1. Nevertheless, the choice of the (precise) regularizing distribution u, while entailing beneficial calibration properties, introduces yet another bias. As we will show in Chapter 4, label relaxation, one of our core contributions, can be seen as a generalization of this method that addresses this deficiency.

2.3 Regression

As another relevant problem, *regression* describes the setting where the target space $\mathcal{Y}$ is considered to be a continuous space, typically $\mathcal{Y} = \mathbb{R}$. For example, the task might be to predict the price of a house in Euros based on a variety of explanatory variables, including its size, number of bedrooms, location, and age. Going beyond, regression is often also considered in a multi-target scenario [Bor+15], where $\mathcal{Y}$ is composed of multiple target dimensions, i.e., $\mathcal{Y} = \mathbb{R}^d$ with $d \in \mathbb{N} > 1$. As an example

[2] In this respect, the approach can also be seen as a methodology to adapt the data to the model.

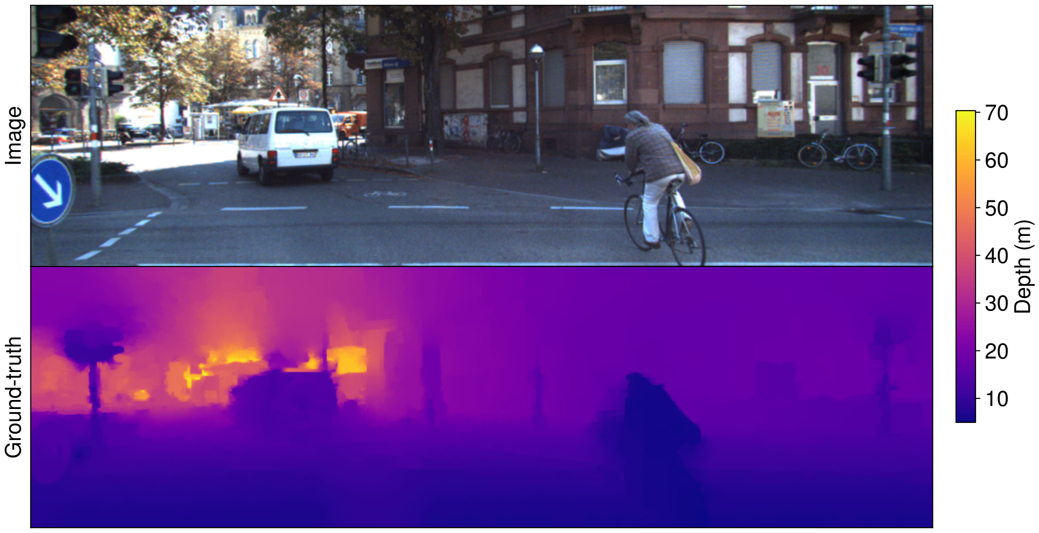


Fig. 2.3.: In monocular depth regression, the task is to predict the projected metric distance to the camera for each pixel given a single image, thus forming a multi-target regression problem. Here, this task is illustrated for an urban street scene taken from the dataset `KITTI` [Gei+13].

in this case, the task of monocular depth estimation (MDE) in images [Zha+20a] calls for predicting a map of depth values that assigns the distance from each pixel's projection to the camera based on a single image of the scene, thus requiring a structured output of individual regression values. As will be discussed in the course of presenting solutions to MDE in our work in Chapters 8 to 10, this problem can be relevant in many practical applications, e.g., in autonomous driving. Fig. 2.3 illustrates this problem in such a context.

Consistent with the non-deterministic modeling discussed above, a stochastic dependence between $\mathcal{X}$ and $\mathcal{Y}$ is assumed in the form of $y = g(\boldsymbol{x}) + \epsilon$, where g is the ground truth relation as introduced in Section 2.1, and $\epsilon \in \mathbb{R}$ is a random error. In the MDE example, ϵ may be depth sensor distortions due to resolution limitations. Learning models for regression then aim to fit a model $h : \mathcal{X} \longrightarrow \mathcal{Y}$, e.g., by minimizing the empirical risk based on an underlying regression loss $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \longrightarrow \mathbb{R}$. Most notably, the least squares criterion [Leg05]

$$\mathcal{L}_2(y, h(\boldsymbol{x})) := (y - h(\boldsymbol{x}))^2 \tag{2.9}$$

is considered to be a common choice for many models, making assumptions about ϵ in terms of homoscedasticity (constant variance) and the absence of autocorrelation (independent errors between individual samples). When warranted, such methods yield efficient solutions to this problem [Dou11]. Regression models h can take many forms, from simple linear regression, which assumes a linear relationship between variables, to more complex nonlinear regression models that can capture more intricate relationships between variables [Del+19].

Challenges and Learning Model Adaptations

However, in practice, model assumptions as sketched above are often violated. For instance, as we argue in Chapter 9, the quality of depth sensors used to construct training data for monocular depth estimation typically degrades with increased distance [WS16]. Also, the quality of sensors is often affected by operating time, so that the noise in temporally adjacent instances is more correlated than those between instances collected at large time intervals. Finally, data sets often also contain outliers that cannot always be accurately removed. In such cases, models with unrealistic assumptions will fail to provide accurate regressors, leading to inefficient and misleading hypotheses [BW75; Luk+19].

To mitigate the negative implications and to achieve robustness against harming biases, model class and optimization compatibility with the data is again urged. A systematic distortion of targets y could be addressed by adding an appropriate model bias to h to approximate g based on observations y. At the optimization level, the design of robust loss functions has received much attention in the past. For example, generalized versions of the least squares criterion have been proposed to weaken the assumptions on the the given data [Kme71; KK04]. As another popular method, the so-called M-estimators [Hub81] generalize the idea of maximum likelihood estimation by allowing to incorporate robust loss functions as an optimization criterion, including the absolute error [WM05], the Huber loss [Hub81] or robust loss generalizations [Bar19]. Beyond this, several methods have been further tailored to specific domains, including human pose estimation [Bel+15], time series forecasting [ZL21], and, as also addressed in our work in Chapter 9, depth regression [IKK19; Ran+22].

2.4 Instance Weighting

In many applications, the selection of instances in a training dataset $\mathcal{D}_N$ can already comprise sampling biases, such that it may not adequately represent the marginal distributions p_X^* and p_Y^*. For instance, some regions in $\mathcal{X}$ may be underrepresented by instances in $\mathcal{D}_N$, or labels in $\mathcal{Y}$ may be unevenly sampled despite a uniform prior distribution. As a result, the empirical risk minimizer h_{emp}^* in Eq. (2.5) may not minimize the actual risk $\mathcal{R}$, leading to unsatisfactory and even questionable solutions in critical aspects such as fairness [Wan+19a]. This problem has already been extensively studied in classical statistics and econometrics, where it is known as *sample selection bias* [Hec79]. Transferred to the field of machine learning, a predominant mechanism to address this issue is to control the influence of individual instances on the overall learning process, commonly known as *instance weighting* [ZLA03]. So far, we have primarily discussed methodologies for tailoring learning models to data. In contrast, which is at the core of this thesis, data itself can also be adjusted for compatibility with an existing learning model. Instance weighting can be seen as a paradigm that lies somewhere in between the two extremes, adjusting the optimization objective while augmenting the data at the same time.

When applied to empirical risk minimization in the context of supervised learning, the *instance-weighted empirical risk* [Hua+06] can be notated by

$$\mathcal{R}_{\text{emp}}^w(h, \mathcal{D}_N) := \frac{1}{N} \sum\nolimits_{i=1}^N w_i \mathcal{L}\left(y_i, h(\boldsymbol{x}_i)\right) \,, \tag{2.10}$$

where $w_i \in \mathbb{R}$ denotes the sample weight of the i-th instance $(\boldsymbol{x}_i, y_i) \in \mathcal{D}_N$, usually scaled to values in $[0, 1]$. The weighted variant of the empirical risk is typically considered in cases where $\mathcal{D}_N$ is assumed to be sampled from a deviating distribution $p \in \mathbb{P}(\mathcal{X} \times \mathcal{Y})$ with $p \neq p^*$ [Hua+06], so w helps to bridge the gap caused by *dataset shifts* [Mor+12].

In the past, many methods to incorporate instance weighting into supervised learning have been proposed. A large fraction of methods induces weights w by a preceding density ratio estimation [Zad04; XPX18], which serves as a basis for emphasizing underrepresented and de-emphasizing overrepresented instances. In this context, the notion of *instance importance* [Sug+07] has been established, for which estimation methods include approaches relying on a Kullback-Leibler divergence criterion [Sug+07; Tsu+09] or kernel mean matching [Hua+06; WGS15]. Recent work further suggests learning such weights jointly with the predictive model [Ren+18], or exploiting connections between the actual (biased) training data sampling and the true distribution from auxiliary information about the latter [Ber+21]. Methods have also been tailored to specific domains [Meh+22] and model classes, such as support vector machines [YSW07; FZL18; Tao+20] or linear regression [Cle79; EGW10]. In addition, instance weighting as a means of de-biasing learning has been the subject of more theoretical studies, e.g., with respect to various importance estimation functions that emerge in different sampling settings [Vog+20], or a binary weighting through subsampling of instances from a biased training data [CL22]. As we will show in Chapter 8 and as has already been considered in [LH15; LH16], we propose an instance weighting approach that relies on a data modeling methodology involving the deliberate imprecisiation of label information, and demonstrate its effectiveness for classification and regression.

3 Weakly-Supervised Learning: Generalizing Models and Data

In this thesis, we consider methods to ensure compatibility between learning models and potentially biasing training data by (re-)modeling the label information, achieved by leveraging more expressive representations of knowledge and beliefs. To this end, we introduce the paradigm of *weakly-supervised learning* here as our main workhorse for generalizing learning methods to richer forms of supervision such as those used in our approaches. We begin with an overview of different forms of weak supervision and learning models that cope with the corresponding type of label appearance in Section 3.1. As being most relevant for our own studies, we then highlight the two settings of superset and semi-supervised learning in Sections 3.2 and 3.3, respectively. Finally, relative comparisons as yet another form of weak supervision are discussed in Section 3.4.

3.1 Overview

Ideally, annotations are available in a precise, unambiguous, and error-free manner, allowing for the application of supervised learning methods to train models with high generalization capabilities. However, obtaining such high-quality labeled data is often not an option, e.g., due to economic constraints. In many cases, the targets are only available in weaker forms, requiring methods that generalize to less precise data. The paradigm of so-called *weakly-supervised learning* [Zho17] comprises methods that allow learning from such weaker label information.

In this regard, three main forms of weak supervision can be distinguished [Zho17; Cab22]: incomplete, inexact, and inaccurate supervision. *Incomplete* supervision arises when data is not generally unambiguously labeled, and the provided training instances either lack label information or have ambiguous label information. *Inexact* supervision involves labels associated with groups rather than individual instances, where the label applies to the entire group and does not necessarily need to be

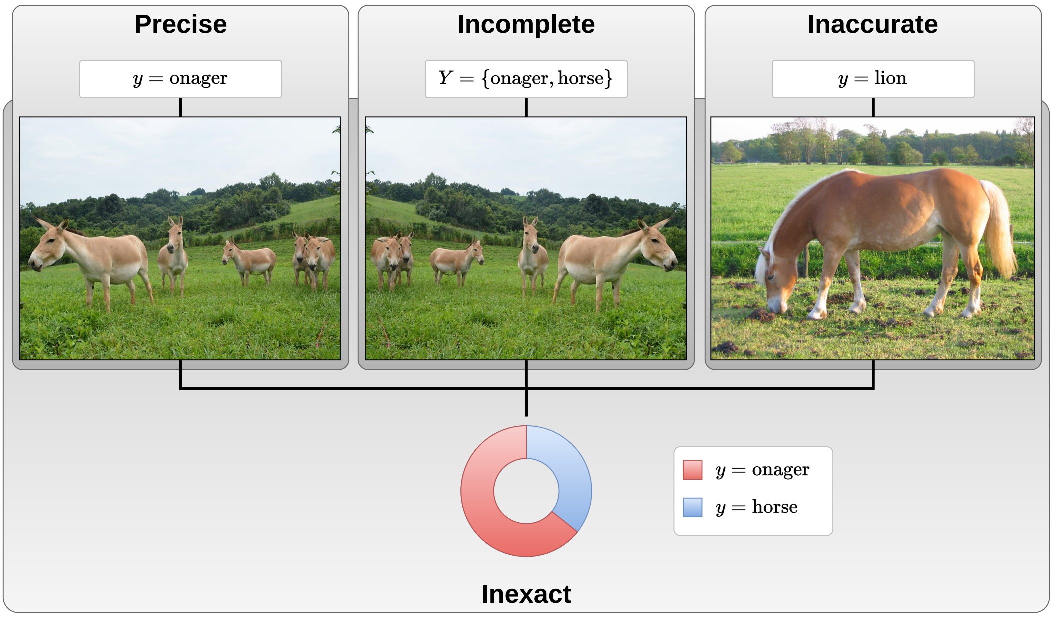


Fig. 3.1.: Types of weak supervision that can be distinguished, here in the case of images to be classified: Ideally, precise label information is given by unambiguous labeling. Conversely, incompletely annotated examples may have multiple candidate values as labels, which implies ambiguity. Inaccurate labels are weak in terms of quality, although they are typically unambiguous, leading to a misleading supervision as in the case of mislabeled target information. In the case of inexact supervision, labels cannot be attributed to individual instances, but to groups of them, as shown here in the case of label proportions over the three examples as one form of inexact supervision.

assigned to each individual instance. Finally, *inaccurate* supervision describes labels that are precise and unambiguous, but may originate from weak annotators that are weakly correlated with the ground truth, leading to (erroneous) labels. Figure 3.1 illustrates this distinction. In the following, we will delve into each of the three forms of weak supervision and explore methods for dealing with them.

3.1.1 Incomplete Supervision

Formally, incomplete supervision in a given training dataset $\mathcal{D}_N$ is characterized by observing samples of the form $(\boldsymbol{x}, Y) \in \mathcal{X} \times \mathcal{S}$, where $\mathcal{S} = 2^{\mathcal{Y}} \setminus \emptyset$ denotes the space of the power set over $\mathcal{Y}$ without the empty set [LD14]. While this technically also includes supervised and unsupervised learning as special cases with $Y = \{y\}$ and $Y = \mathcal{Y}$ for all instances $(\boldsymbol{x}, Y) \in \mathcal{D}_N$, respectively, incomplete supervision typically refers to the case where neither all instances are fully precise nor fully agnostic.

Learning from such supervision then implies coping with the ambiguity in Y, e.g., by identifying the ground truth y in Y.

Incomplete supervision has been studied in several contexts with specific assumptions, such as restrictions on the target sets Y or distinctions in the degrees of ambiguity for different parts of the data. As the most common setting considered in this regard, so-called *superset learning* [HC15], also referred to as *partial label learning* [JG02], makes the assumption that Y are supersets of $\{y\}$, i.e., that the ground truth is always an element of the target sets. In this thesis, we present solutions to this problem in Chapters 4 to 9, which is why we introduce this setting thoroughly in Section 3.2. Furthermore, *semi-supervised learning* [CSZ06] constitutes another special case of incomplete supervision, where a fraction of instances is associated with precise labels $Y = \{y\}$, while the rest is unlabeled, i.e., $Y = \mathcal{Y}$. This setting is considered in our work in Chapters 5, 6 and 8, which is why we highlight it in detail in Section 3.3.

Besides the aforementioned settings that are most relevant to our work, certain settings have also been related to the target space $\mathcal{Y}$. For example in the context of binary classification, another studied form of incomplete supervision is the paradigm of learning from positive and unlabeled data, or *PU learning* [BD20] for short. Here, labeled data comprises only positive examples, while the unlabeled data contains examples from both classes. This setting is a frequently appearing setting in real-life, e.g., as in medical analysis, where only positive diagnoses are recorded, not the absence of certain diseases [Cla+15]. Methods addressing the problem of PU learning include approaches that first determine reliable negative instances in the unlabeled data, and then apply (semi-)supervised learning techniques to determine a classifier on this self-labeled data [Liu+03; He+18], treating all unlabeled data examples as negative with label noise [Gon+21], or incorporating known class priors for model adaptation [BRD19]. As an example of a specific setting in multi-class classification, learning from *complementary labels* [Ish+17] considers settings in which targets are given by $Y_{\bar{y}} = \mathcal{Y} \setminus \{\bar{y}\}$, i.e., instances are labeled with complements of labels $\bar{y} \in \mathcal{Y}$ to which they do *not* belong, e.g., to exclude diseases that can be safely ruled out by specific examinations in medical applications. Methods for learning from such supervision suggest adapting loss functions to this kind of label information [Ish+17; Cho+20], using loss correction techniques [Yu+18], applying generative adversarial networks [Xu+20] or considering it in the context of label noise [IIS22]. However, since labels $Y_{\bar{y}}$ naturally form supersets as introduced earlier, methods approaching this problem may also include models from the field of superset learning (cf. Section 3.2).

3.1.2 Inexact Supervision

While incomplete supervision generalizes the label information in a set-valued manner on a per-instance level, inexact weak supervision captures the label assignment to sets of instances. More precisely, datasets are of the form $\mathcal{D}_N = \{(\tilde{X}, y_i)\}_{i=1}^N \subseteq (2^{\mathcal{X}} \setminus \emptyset) \times \mathcal{Y}$, where we call $\tilde{X} \in (2^{\mathcal{X}} \setminus \emptyset)$ *bags* of instances.[1] Hence, a label information y applies to a group of instances $\tilde{X}$, but it is not generally specified whether y applies to all $\boldsymbol{x} \in \tilde{X}$ individually, a subselection $\tilde{X}' \subset \tilde{X}$, or an aggregated measure over $\tilde{X}$. In the following, we briefly introduce multiple instance learning and label proportions as two alternative settings, typically addressing classification problems, that refer to this distinction.

Multiple Instance Learning

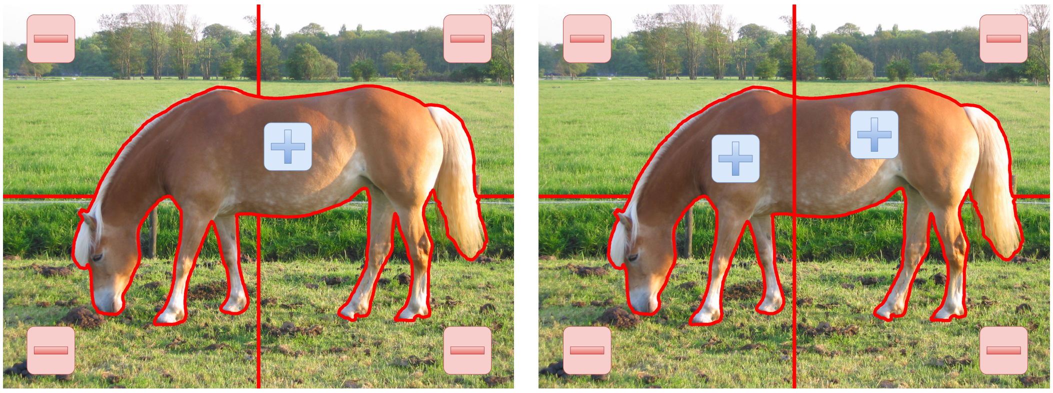


Fig. 3.2.: Following [Car+18], image classification can be interpreted as a multiple instance learning problem as shown here for a bag of instances in form of image regions, which are associated with the label `domestic horse`. The left case illustrates the standard case, where only one region is positively associated with the label, while the right case illustrates collective MIL.

As one line of research, *multiple instance learning* (MIL) [DLL97] describes the setting where a label y applies to a bag $\tilde{X}$ if at least one instance representation $\boldsymbol{x} \in \tilde{X}$ is positive in this relationship, thus covering the first two variations of the problem setup outlined above. As an example, image classification, as introduced before and thoroughly examined in Chapters 4 to 8, can be considered as a multiple instance learning task, where individual regions of an image (i.e., pixel subsets) are elements $\boldsymbol{x}$ of the whole image bag $\tilde{X}$ [Amo13]. If, for instance, an image is

[1] Note that we follow [Zho17] here and abstract from the case of instances occurring multiple times in the same bag, even though this is also conceivable.

labeled with `domestic horse`, and there is such an animal depicted in a certain region of the image, then there is an instance $\boldsymbol{x}$ that is positive with respect to that label. Of course, the problem can be more complicated, e.g., if the horse is spread over multiple regions, then multiple instances $\tilde{X}' \subseteq \tilde{X}$ contribute to this positivity. Figure 3.2 illustrates these two cases, which are distinguished in the literature as *standard* and *collective MIL assumptions* [Car+18].

While MIL can be viewed as an abstract paradigm, methods for approaching this task can be categorized by their assumptions about the problem. Namely, the prediction level may distinguish between instance- and bag-level predictions. While the latter modeling is predominantly adopted, e.g., to predict a disease in medical diagnosis [Sha+21] or the activity of a bag of molecules in drug-level discovery [ML97], and methods range from SVM adaptations [ATH02; DR14] and clustering [Li+14] to instance pooling [Zhu+17] and attention-based deep neural networks [ITW18; Has+20], instance-level prediction is less common. Typically, the task is to identify individual instances $\boldsymbol{x}$ in the bag that lead to the observed label y associated with the entire bag, for instance to localize superpixels in images that exhibit the characteristic leading to the overall image categorization [RHF15]. Another characteristic is the nature of intra-bag relationship between the instances, which are often assumed to be independent and identically distributed. However, some works have also successfully incorporated a more realistic non-i.i.d. modeling [Zha21], including explicitly capturing instance similarities within bags [ZSL09]. Other concerns involve label noise [DG20] or differences in the instance- and bag-level label spaces [Amo10]. Overall, MIL is a broad area of active research, encompassing many different interpretations of the foundational problem formulation. For a more comprehensive overview of the field, we refer to [Car+18].

Label Proportions

Another problem of practical relevance is learning from *label proportions* (LLP) [Qua+09]. Here, bags $\tilde{X}$ are associated with fractions $\pi_{\tilde{X}} = (\pi_{\tilde{X},1}, \dots, \pi_{\tilde{X},K}) \in [0,1]^K$ for K classes that indicate how many instances $\boldsymbol{x} \in \tilde{X}$ are of the respective class $y_i \in \mathcal{Y}$, i.e., $\pi_{\tilde{X},i} := \frac{1}{|\tilde{X}|} \left| \{\boldsymbol{x} \in \tilde{X} \, : \, g(\boldsymbol{x}) = y_i\} \right|$ with $g \in \mathcal{Y}^{\mathcal{X}}$ being the ground truth relation. By definition, it holds that $\sum_{i \in [K]} \pi_{\tilde{X},i} = 1$ for the observed fractions $0 \leq \pi_{\tilde{X},i} \leq 1$. For example, different social media accounts share images with specific content, such as soccer or architecture. Even without observing a label for each of the images, one might be able to obtain global statistics about the class distribution within the collection of images shared by each account. This information is then used to augment the bag of ordinarily unlabeled images. Another example is spam-filtering, where one could observe spam proportions for different mailbox directories, e.g., unanimous spam for the junk folder, but much more relevant mails in the main inbox [Qua+09].

A large fraction of work proposes to solve the problem by learning models on the individual instances of the bags such that the model predictions for those instances match the label proportions observed at the meta level. First studies approached the problem by learning a conditional exponential model that maximizes the likelihood of the observed data, assuming independence of the class-conditional distribution of the data in the bags [Qua+09; Fan+14; Pat+14]. Another work reformulated the problem of empirical risk minimization by targeting the “empirical proportion risk” [Yu+14], generalizing the optimization objective to the inexact weak supervision. Other methods include clustering [SM11], SVM-based model adaptation [Rüp10; Yu+13], deep neural networks [Dul+19; Liu+19b; ZWS22] and even semi-supervised learning methods [TL20]. Among others, LLP has been successfully applied in domains such as high-energy physics [Der+18], medical diagnosis [Bor+18] and video event detection [Lai+14].

3.1.3 Inaccurate Supervision

Inaccurate weak supervision describes a supervised learning setting with fully-supervised datasets $\mathcal{D}_N = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ as introduced before, where y_i is gathered from some annotator $f : \mathcal{X} \longrightarrow \mathcal{Y}$ that weakly correlates with the ground truth g, leading to unreliable and noisy targets [Cab22]. While this setting can also be interpreted as supervised learning with label noise, we address this problem formulation in our work presented in Chapters 7 and 8, methods in the field of weakly-supervised learning have been tailored to certain types of inaccurate annotators f. In the following, we will highlight *crowdsourcing*, *distant supervision* or *transfer learning* as three prominent paradigms in this regard.

Crowdsourcing

Crowdsourcing [Bra08a] in the context of machine learning typically describes the process of leveraging a crowd, usually a larger group of people, to collectively contribute to the acquisition of training data. While this process can include the collection of instances themselves too, e.g., uploaded images to a social media platform, we focus here on the labeling procedure itself. For this purpose, microtask platforms like Amazon Mechanical Turk [Cro12] offer crowdsourcing services in a scalable way. Nevertheless, crowd annotators may disagree, so methods to resolve this disagreement to get a single precise label are required.[2]

Approaches to tackle this issue include selecting the label with the majority of votes [Sno+08], also including repetitions in the labeling process [SPI08] and weighting individual annotators by their reliability [Whi+09]. Recently, more sophisticated model integrations in the deep learning regime have been suggested. For instance, a deep model for joint learning of labeler reliability and label inference can be trained using an expectation-maximization (EM) algorithm formulation [RP18]. Another method uses the labeler performances within a regularization term for model optimization [Tan+19]. Other robust methods suggest employing belief propagation [Kim+22] or support vector machines [YLJ23], or even self-training to tackle data sparsity and imbalance in the observed labels [SLH21]. Of course, the use of crowdsourcing as a labeling source must also be considered in an economic dimension, balancing quality and quantity [WZ16].

Distant Supervision

As another form of inaccurate supervision, *distant supervision* [Min+09] describes a technique that heuristically labels previously unlabeled data by linking it to external knowledge bases granting labeling hints. In its primal proposal, this technique is used to label sentences by relations stored externally in a database. Whenever entities associated with a relation occur in a matching pattern in the sentence text, the label derived from the relational information is assigned to the sentence. Another example is sentiment analysis based on emoticons in Twitter tweets, where a mapping of emoticons to the expected mood of the writer can be used to reason about sentiment, thus providing a form of distant supervision [PP10; Tan+14]. In such cases, an external knowledge base is leveraged to construct labeled data, though in a weak way, as the approaches tend to be rather simplistic. Nevertheless, the knowledge base may not always be aligned with the data to be labeled, resulting in low-quality labels. [Zen+15] addresses this problem by modeling uncertainty-awareness, so

[2] It is worth mentioning that such information may also be modeled in an incomplete way as described in Section 3.1.1.

that deficiencies in the quality of misaligned distant supervision are more explicitly reflected. Also, scaling up the amount of constructed data can mitigate alignment issues [Der+17]. Especially in the domain of natural language processing, distant supervision is employed in many state-of-the-art models [AHH19; Spi+21; Lin+22], but has also found its way into other domains, such as computer vision [Yao+21].

Transfer Learning

Besides distant supervision, labeling information can also be carried over in a weak form from another task, commonly known as *transfer learning* [PY10]. This setting describes the process of reusing knowledge from a source task to improve the performance on a target task, where the source and target feature domains and tasks may be different but related in some way. As an example, consider the estimation of depth in urban scenes as a task, for which one may have observed images of streets in a certain country as source domain. If now such a system is to be deployed in a different country, where the architecture of buildings, the clothing of people and the flora differ from the source domain, learned concepts such as the arrangement of streets and sidewalks can still be transferred to the new target domain, so that the source domain provides potentially useful knowledge. In general, the field of transfer learning is very broad and comes in various facets, which is why we only highlight approaches here that can be considered as a natural form of inaccurate weak supervision, adopting the terminology of [PY10]. We refer to [PY10; Tan+18; Zhu+21] for more comprehensive overviews of transfer learning.

[PY10] distinguishes *inductive*, *transductive* and *unsupervised* transfer learning as three settings that differ in the relations between the source and target domains and tasks. In the inductive setting, the source and target tasks are assumed to be different, while the domains can be both similar or diverging. Methods span from multi-task learning [ZY22], where both tasks are learned jointly and source domain labels can support the target task learning, to self-taught learning [Rai+07], in which no labels of the source domain are available, and the label spaces of both domains may not even be related. Inductive transfer learning approaches have been shown to be effective in supporting the learning of certain tasks in applications like natural language processing [Sha+19; Dai+21] or computer vision [Jie+17; Chu+21], providing additional, albeit weak as not matching the target task, side information to learn target tasks at hand. As opposed to this, transductive transfer learning typically considers the tasks to be the same but assumes shifts in the source and target domains. Hence, the ground truth target distribution p^*_{tgt} does not necessarily match the source distribution p^*_{src}, leading to a distorted, and thus weaker, supervision in the form of examples sampled according to p^*_{src}. Typical techniques to cope with this problem are domain adaptation [Far+21] or dealing with sample selection biases [Liu+16] (cf. Section 2.4), applied to tasks like object localization [Zhu+22] or promoting fairness in trained models [DW21]. Lastly, unsupervised transfer learning describes the setting where no labels are available at all given a different but related source and target task. Methods in this regard typically use clustering approaches that aim to learn a unifying representation between the source and target domain [WSZ08] or adversarial techniques [MF21].

Data Programming

The aforementioned forms of inaccurate weak supervision can be generalized by the so-called *data programming* paradigm [Rat+16], also referred to as *programmatic weak supervision* (PWS) [Zha+22b]. The key idea of PWS is to model weak labeling sources, e.g., stemming from domain heuristics or crowdsourcing sources, as *labeling functions*, which are user-defined programs that return (typically noisy) labels. In *label models*, these sources can then be synthesized in a supervised manner for model training. PWS as a framework provides abstractions for defining such syntheses, thereby specifying conventions for each phase of the weak labeling procedure. It has gained great popularity, also thanks to effective and scalable implementations [Zha+21b].

At the beginning of PWS pipelines to generate supervision, labeling functions are specified to retrieve noisy, thus inaccurate, weak labels from unlabeled data, which can be defined manually [Rat+17] or generated automatically [VR18; Xu+21], e.g., by learning labeling functions themselves [Li+21a], or in an interactive fashion [GGT21; Boe+21]. Labeling functions can be of various types. For example, domain knowledge of (human) experts can be leveraged by specifying heuristics, e.g., by using keyword matching or regular expressions in natural language processing [Men+18; Awa+20] or recognizing visual concepts in images [Fu+20]. Other bases for labeling functions may include knowledge gathered from distant supervision as introduced earlier, e.g., from knowledge bases, pre-trained models or domain-specific tools. Lastly, individual annotators in a crowd can be modeled as labeling functions as well [Rat+17; Lan+20].

Individual weak labeling functions may lead to contradictory label information. Therefore, label models aim to resolve this ambiguity by learning a probabilistic model on top of the labeling functions, which allows to account for interrelationships and dependencies between the individual labelers. Such models have been developed for specific problems, including classification [Rat+19] and sequence tagging [Lis+20; Li+21b], but have recently also been generalized to arbitrary tasks [Shi+22]. Label models might consider the subsequently learned model using the newly generated weak supervision for a joint optimization, as developed for classification [Ren+20] or sequence tagging [Lan+20; PY21]. Such methods have further been considered in semi-supervised learning [Mah+21] and self-training contexts [Kar+21].

3.2 Superset Learning

A particularly relevant setting of incomplete supervision, as already introduced in Section 3.1.1, is the so-called *superset learning* [LD12], also known as learning with ambiguous [HB06] or partial [CST11] labels, for which we have proposed methods in Chapters 4 to 9. In this setting, the goal is to minimize the risk $\mathcal{R}(h, p^*)$ of a model h as introduced in Eq. (2.2) by not directly accessing instances $(\boldsymbol{x}, y)$ with precise annotations $y \in \mathcal{Y}$, but by observing annotations in the form of *supersets* $Y \in \mathcal{S} = 2^{\mathcal{Y}} \setminus \emptyset$, i.e., sets of candidate labels $y' \in \mathcal{Y}$ that seem plausible to be the true label y. Thus, datasets are given in the form

$$\mathcal{D}_N^S := \{(\boldsymbol{x}_i, Y_i)\}_{i=1}^N \subset \mathcal{X} \times \mathcal{S}\,, \tag{3.1}$$

where the instances are realizations of the random variables $(X, S) \sim s^*$ with $s^* \in \mathbb{P}(\mathcal{X} \times \mathcal{S})$ denoting a probability distribution over the feature and superset space.

For example, partial supervision might occur in cases where a precise annotation of the data is not feasible, and the annotator has the option of selecting several potential labels without committing to any one of them. In this regard, consider again the image classification example for the Equidae genus from Section 2.2. A non-zoologist without expert knowledge about this genus could be unsure whether to classify an image depicting an instance of `Persian onager` as such, or alternatively as an `Somali wild ass` or `Eastern kiang` species, which generally do not differ much visually. However, it may be easy to see that this image does not show a domestic horse, such that this label can be discarded with high confidence. Selecting the former as possible labels by means of $Y = \{\texttt{Persian onager}, \texttt{Somali wild ass}, \texttt{Eastern kiang}\}$ makes a more cautious decision, which appears to be more reliable than committing to a single label. Another example might be the deployment of sensors with tolerances $\pm\epsilon \in \mathbb{R}_+$. When sensing data, each observation is not guaranteed to reflect the underlying ground truth value y, but the tolerance range based on ϵ specifies an interval in which it is likely to fall. Therefore, it is reasonable to model the supervision partially by the interval of the tolerance region, i.e., $Y = [y - \epsilon, y + \epsilon]$, which in turn again leads to a more cautious and reliable form of supervision.

Nevertheless, it remains still unclear how to learn from examples $(\boldsymbol{x}, Y)$ to minimize $\mathcal{R}(h, p^*)$, which is the expectation over the space $\mathcal{X} \times \mathcal{Y}$, but not (as observed here) over $\mathcal{X} \times \mathcal{S}$. Before introducing concrete approaches aiming to bridge this gap, we first revisit a common assumption that characterizes the overall "solvability" of the superset learning problem [CST11]. To this end, we denote a joint distribution over $\mathcal{X} \times \mathcal{Y} \times \mathcal{S}$, of which p^* and s^* are the respective marginals over $\mathcal{X} \times \mathcal{Y}$ and $\mathcal{X} \times \mathcal{S}$, by $p_s^* \in \mathbb{P}(\mathcal{X} \times \mathcal{Y} \times \mathcal{S})$. Moreover, following [LD14], we refer to samples $(\boldsymbol{x}, y, Y) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{S}$ distributed according to p_s^* as *complete*. Then, the *superset assumption* [LD14; HC15] states that for any complete example $(\boldsymbol{x}, y, Y) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{S}$, it holds that

$$\mathrm{Pr}_{(\boldsymbol{x},y,Y)\sim p_s^*}(y \in Y) = 1\,. \tag{3.2}$$

Roughly speaking, this assumption implies that the ground truth label y is always observed in the corresponding superset Y, hence the name "superset".[3] In case of the existence of a joint distribution p_s^* satisfying the superset assumption, p^* is said to be *eligible* for s^* as a notion of compatibility, denoted by $p^* \vdash s^*$ [CRB20].

Still, this assumption alone is incapable of characterizing the quality of the provided superset supervision in a nuanced way. Intuitively, smaller supersets tend to provide a more useful information than larger ones (with $Y = \{y\}$ and $Y = \mathcal{Y}$ as extreme cases), but the composition of the individual supersets also matters. For example, superset labels for instances belonging to a certain class y might always include a distractor label $\bar{y} \neq y$. In this case, a model will not be able to detect a classification error in predicting $\bar{y}$ instead of y. To indicate the "difficulty" of the problem and to

[3] While most approaches in the field of superset learning take this assumption for granted, a recent work has also investigated the setting of so-called "unreliable partial-label learning" [Lv+21]. Here, the superset assumption is weakened by not necessarily observing the ground truth y as a candidate in the partial label Y with full probability.

condition the learnability of superset learning problems in a more subtle way, the *ambiguity degree* [CST11] of data distributed according to p_s^* is defined as

$$\gamma := \sup_{(\boldsymbol{x},y,Y)\sim p_s^*\,,\,\bar{y}\in\mathcal{Y}\,:\,\bar{y}\neq y} \Pr(\bar{y} \in Y)\,. \tag{3.3}$$

The ambiguity γ expresses the maximum probability of a distracting label $\bar{y}$ to co-occur with the ground truth label y. For $\gamma = 0$, there is no ambiguity at all and the fully-supervised learning setting is retained. For $\gamma = 1$, there is at least one pair of $\bar{y}$ and y that always occurs together, which prevents a learner from detecting an erroneous prediction between the labels of this pair. A common assumption is that $\gamma < 1$, which is called the *small ambiguity degree condition* [LD14].

From a risk minimization point of view, the set-valued nature of the observed labels $Y \in \mathcal{S}$ requires an adaptation of the objective to be optimized. In general, the risk to be minimized when learning from superset-labeled instances $(\boldsymbol{x}, Y)$ can be defined as *superset risk* $\mathcal{R}^S(h, s^*)$ in the form of

$$\mathcal{R}^S(h, s^*) := \int_{\mathcal{X}\times\mathcal{S}} \mathcal{L}^*\left(Y, h(\boldsymbol{x})\right) \mathrm{d}s^*\left(\boldsymbol{x}, Y\right)\,, \tag{3.4}$$

where $\mathcal{L}^*$ denotes a generalized loss function of the form $\mathcal{L}^* : \mathcal{S} \times \mathcal{Y} \longrightarrow \mathbb{R}$. In the case of empirical risk minimization, approaches then aim to optimize the empirical risk

$$\mathcal{R}^S_{\text{emp}}(h, \mathcal{D}^S_N) := \frac{1}{N} \sum\nolimits_{(\boldsymbol{x},Y)\in\mathcal{D}^S_N} \mathcal{L}^*\left(Y, h(\boldsymbol{x})\right) \tag{3.5}$$

on a superset learning dataset $\mathcal{D}^S_N$ as specified in Eq. (3.1). As first proved in [LD14], the small ambiguity degree condition is sufficient for the ERM learnability of the superset learning problem. Moreover, the same study presented an alternative criterion for sufficiency, which is based on the divergence between $\mathcal{R}$ and $\mathcal{R}^S$ for classification problems. In the following, we will introduce proposed solutions to this issue by highlighting branches of the literature on learning with superset-based supervision, namely *average-* and *identification-based* methods.

3.2.1 Average-Based Superset Learning

In *average-based superset learning*, early methods approached the problem by specifying a loss generalization $\mathcal{L}^*$ that averages over the candidate labels $y' \in Y$, i.e., by means of

$$\mathcal{L}^*_{\text{avg}}(Y, h(\boldsymbol{x})) := \frac{1}{|Y|} \sum\nolimits_{y' \in Y} \mathcal{L}(y', h(\boldsymbol{x}))\,, \tag{3.6}$$

where h is a hypothesis and $\mathcal{L} : \mathcal{Y}^2 \longrightarrow \mathbb{R}$ is a point-wise base loss [JG02; Den13]. In this case, all labels in Y are treated equally.

Among the first approaches of this kind, [HB06] proposes adaptations of k-nearest neighbor clustering, decision trees and rule learners for classification to generalize these models toward superset supervision, e.g., by replacing the majority-based neighborhood voting scheme in k-nearest neighbor classification with the accumulation of each label's membership in the supersets of the neighboring instances. Later, [Cou+09; CST11] were the first to propose a generalized convex loss formulation whose empirical risk minimizer h^*_{emp} is consistent with the *partial 0/1 loss*

$$\mathcal{L}^*_{0/1}\,(Y, h(\boldsymbol{x})) := \mathbb{1}[h(\boldsymbol{x}) \notin Y]\,, \tag{3.7}$$

which is a superset-aware generalization of the classical *0/1 loss*

$$\mathcal{L}_{0/1}\,(y, h(\boldsymbol{x})) := \mathbb{1}[h(\boldsymbol{x}) \neq y]\,. \tag{3.8}$$

As proved in these works, the consistency of $\mathcal{L}^*_{0/1}$ with $\mathcal{L}_{0/1}$ and generalization bounds with respect to the risk on p^* can be bounded based on the ambiguity degree γ. Liu and Dietterich [LD14] further show that a necessary condition for the ERM learnability of the superset learning problem is that $\nexists h \in \mathcal{H}$ with non-zero risk $\mathcal{R}(h, p^*) > 0$ but zero superset risk $\mathcal{R}^S(h, s^*) = 0$ based on $\mathcal{L}_{0/1}$ and $\mathcal{L}^*_{0/1}$, respectively. Building upon these findings, [Yao+20] investigated the use of a cross-entropy based average loss in deep convolutional neural networks. Other works discussed average-based approaches as special cases of more general disambiguation strategies [Yao+20; Wen+21].

A recent study has shown favorable robustness in the case of bounded multi-class losses $\mathcal{L}$ for different data generation processes [Lv+21], for which this work proposes a family of average-based superset learning losses. However, generally speaking, average-based solutions have shown sensitivity to false positive predictions [Yao+20], especially when the discrepancy of the underlying loss $\mathcal{L}$ is high [CRB20].

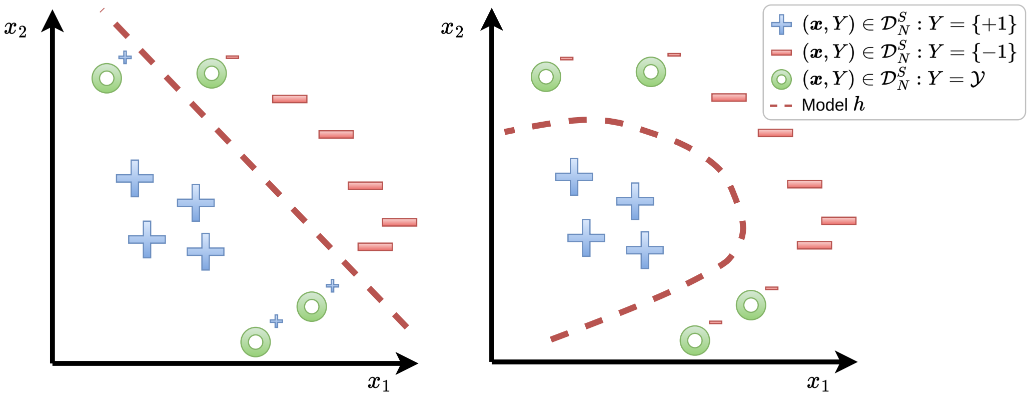
Moreover, as commonly considered in the regime of deep learning, large overparameterized models with more model parameters than training data samples suffer from *memorizing* [FZ20] any candidate label, further questioning the effectiveness of average-based approaches for optimizing such models [Lv+21].

3.2.2 Identification-Based Superset Learning

By treating all candidate labels in the supersets equally, each of them is considered in the loss calculation. Thus, falsely positive labels in the candidate sets also contribute to the overall optimization, potentially misleading the model optimization. In doing so, average-based methods overlook the potential of *disambiguating* to determine labels that appear more plausible than others within the supersets, which would allow for a more informed loss calculation. For this reason, a second, much larger branch of the literature on superset or partial label learning attempts to find the correct label among the candidates, which we refer to as *identification-based superset learning* [Hül14].

Methods in this paradigm typically seek to distinguish labels y' in supersets Y based on their likelihood of representing the ground truth y. Various ideas have been proposed to do this, notably by treating the true label y as a latent variable and then applying maximum likelihood estimation [JG02; LD12], maximizing a discriminatory margin that reveals the correct label [NC08; Wan+20; CTC20; Gon+22], or determining candidate label weights based on topological information in the feature space [ZY15; FA18; CBR21]. A large fraction of such methods has explored graph-based approaches to elicit the plausibility of each individual candidate [ZZL16; Gon+18; XLG19; Lyu+21; WZL22; WZL22]. More recently, self-training [FA19; Fen+20; Lv+20; YG20; Wen+21; Ni+21; WWZ22] and representation learning using contrastive losses [Wan+22a; He+22], as well as class activation values [Zha+22a] in the regime of overparameterized models have been leveraged to effectively disambiguate candidate labels.

More technically, many identification-based superset learning methods fundamentally aim at recovering the underlying ground truth distribution p^* via s^* in order to infer a model $h \in \mathcal{H} \subset \mathcal{Y}^{\mathcal{X}}$, where p^* and s^* are assumed to satisfy $p^* \vdash s^*$. However, multiple distributions $p \in \mathbb{P}(\mathcal{X} \times \mathcal{Y})$ might be eligible with s^*, rendering the problem of finding p^* difficult. As a means to resolve this ambiguity, the inductive biases induced by the hypothesis space $\mathcal{H}$ can be taken into account for guidance, leading to a major scheme applied in identification-based superset learning. Introduced as *data disambiguation* in [Hül14], and later reformulated as the so-called *minimum variability principle* [CRB20], this procedure aims at selecting a single label y' among the candidates in the superset Y that appear to be most plausible given the model assumptions of the hypothesis space $\mathcal{H}$ under consideration.



x_2
x_1
x_2
x_1
$(\boldsymbol{x}, Y) \in \mathcal{D}_N^S : Y = \{+1\}$
$(\boldsymbol{x}, Y) \in \mathcal{D}_N^S : Y = \{-1\}$
$(\boldsymbol{x}, Y) \in \mathcal{D}_N^S : Y = \mathcal{Y}$
Model h

Fig. 3.3.: The concept of data disambiguation illustrated for an exemplary classification dataset $\mathcal{D}_N^S$ for two different hypothesis spaces, namely linear models (left) and more complex, non-linear hypotheses (right). Depending on the hypothesis space, different labels appear most plausible for the ambiguously labeled instances in green color.

To intuitively motivate this principle, consider a binary classification problem with $\mathcal{Y} = \{+1, -1\}$ and features observed in $\mathcal{X} = \mathbb{R}^2$ as illustrated in Fig. 3.3. Let the training data $\mathcal{D}_N^S$ include unambiguously labeled instances of the positive (blue) and negative (red) class, along with some instances ambiguously labeled by supersets $Y = \{+1, -1\}$ (green). In the left plot, a hypothesis space $\mathcal{H}$ consisting of linear functions in $\mathcal{X}$ is considered. Here, a model $h \in \mathcal{H}$ represented by the red dotted line may be regarded as most favorable in the course of the optimization, which rests on disambiguating most of the supersets to be labeled positively. However, when provided with a more complex hypothesis space $\mathcal{H}' \supset \mathcal{H}$ comprising non-linear functions, the learning model may select a decision boundary that focuses on maintaining a large margin between instances of the two classes, thus recognizing the encircling pattern of negative instances around the positive ones as shown in the right plot. In this case, a curved decision boundary $h \in \mathcal{H}'$ appears most plausible by exploiting a more sophisticated topological structure in $\mathcal{X}$.

Formally, the notion of entwined data disambiguation and model identification can be delineated by virtue of risk minimization. More specifically, following the notation as used in [CRB20], it states to identify the distribution

$$\bar{p} \in \operatorname*{argmin}_{p \in \mathbb{P}(\mathcal{X} \times \mathcal{Y}) \,:\, p \vdash s^*} \mathcal{E}(p) \,, \tag{3.9}$$

where

$$\mathcal{E}(p) := \inf_{h \in \mathcal{H}} \mathcal{R}(h, p) \tag{3.10}$$

represents a variance quantity reflecting the minimum risk $\mathcal{R}$ with respect to an underlying (proper) loss $\mathcal{L}$ of a hypothesis $h \in \mathcal{H}$. Effectively, this “minimin” principle disambiguates s^* by an eligible distribution $\bar{p}$ with respect to its plausibility in terms of the minimum loss achievable with hypotheses in $\mathcal{H}$. Referring to model induction via risk minimization, solutions of this principle read as

$$h^* \in \operatorname*{argmin}_{h \in \mathcal{H}} \mathcal{R}\left(h, \bar{p}\right) \,, \tag{3.11}$$

where $\bar{p}$ is a solution of Eq. (3.9). As further shown in [CRB20], this solution exists under the small ambiguity degree condition.

In fact, models h^* as in Eq. (3.11) minimize the superset risk $\mathcal{R}^S(h, s^*)$ as defined in Eq. (3.4), where the generalized loss function $\mathcal{L}^* : \mathcal{S} \times \mathcal{Y} \longrightarrow \mathbb{R}$ is given by the so-called *optimistic superset loss* $\mathcal{L}^*_{\text{OSL}}$ [HC15], also referred to as the *infimum loss* in [CRB20]. It is defined as

$$\mathcal{L}^*_{\text{OSL}}\left(Y, h(\boldsymbol{x})\right) := \min_{y' \in Y} \mathcal{L}\left(y', h(\boldsymbol{x})\right) \,, \tag{3.12}$$

which captures the simultaneous data disambiguation and model identification as described above, and can be targeted in the empirical risk minimization as in Eq. (3.5). The optimism reflected in the name comes from taking the minimum loss over all candidate labels, where only the best fits among the candidate labels in a superset Y have an impact on the loss value. As also previously being proposed in [UC11] for interval-valued targets, this solution has been widely adopted, including in problem settings like classification [CST11; LH16; CRB20], label ranking [HC15; CRB20], robust regression [LH15], algorithm selection [Han+21], as well as in social science studies [Rod+22]. In our contributions presented in Chapters 4 to 9, we rely on $\mathcal{L}^*_{\text{OSL}}$ to improve neural network calibration and generalization in probabilistic classification and regression.

Also regularized variations of $\mathcal{L}^*_{OSL}$ have been considered, e.g., by augmenting it in

$$\mathcal{L}^*_{\text{ROSL}}\left(Y, h(\boldsymbol{x})\right) := \min_{y' \in Y} \mathcal{L}\left(y', h(\boldsymbol{x})\right) + F(h) \tag{3.13}$$

with a penalty term $F : \mathcal{H} \longrightarrow \mathbb{R}$ on the hypothesis h [HDC19]. As the authors point out, extreme hypotheses h may be favored by the optimistic superset learning optimization because they are able to achieve a very low loss per se, e.g., when a highly overparameterized model is capable of perfectly (over)fitting all training data points. The added regularization term can be used to penalize such extreme hypotheses, just like in classical regularization.

A diametrical approach to an optimistic generalization is its pessimistic counterpart. The so-called *pessimistic superset loss* [Wal45; GD15; HDC19], coined in [CRB20] as *supremum loss*, is defined as

$$\mathcal{L}^*_{\text{PSL}}\left(Y, h(\boldsymbol{x})\right) := \max_{y' \in Y} \mathcal{L}(y', h(\boldsymbol{x})) \,. \tag{3.14}$$

It aims at identifying the model h with respect to the largest loss it achieves over all possible instantiations $y' \in Y$, such that h behaves as well as possible even when the ambiguous label information is instantiated in a particularly unfavorable way. As for $\mathcal{L}^*_{\text{ROSL}}$, [GD18] also proposes a regularized version of $\mathcal{L}^*_{\text{PSL}}$ that suggests correcting each evaluation of $\mathcal{L}\left(y', h(\boldsymbol{x})\right)$ for an instantiation $y' \in Y$ by the minimum loss that is achievable for the label y' over all hypotheses in $\mathcal{H}$, thereby reducing the influence of instantiations that necessarily lead to high losses [HDC19].

3.2.3 Fuzzy Supersets

So far, our focus has been primarily on supersets in the form of classical sets as elements of the space $\mathcal{S}$. While most of the literature in the field considers supersets of this kind, also other, more general forms of partial supervision may be observed. In the following, we explore different conceptualizations of *fuzzy* supersets that allow for a more subtle modeling of knowledge.

Fuzzy Labeling

One way to delineate more general supersets involves the utilization of *fuzzy sets* [Zad65], also known as *fuzzy labels* [WD81; Hül14; Cou+19; Cam23] in the realm of superset learning. Fuzzy sets, which extend the binary nature of membership in classical sets to degrees between the strict inclusion and exclusion, facilitate the representation of partial knowledge in an epistemic manner [CD14]: For values $y' \in \mathcal{Y}$, fuzzy sets allow to specify the possibility that y' represents the underlying ground truth y with varying degrees. This way, they allow to distinguish the support for candidate values individually, and thus deeming one value more plausible than another one, without having to commit to the two extremes of being "fully plausible" or not plausible at all.

Technically, a fuzzy set is defined as a function $\pi : \mathcal{U} \longrightarrow [0,1]$ over a universe of discourse $\mathcal{U}$ that assigns gradual membership values in $[0,1]$ to elements in $\mathcal{U}$, which is a generalization of the classical membership $\pi(\cdot) \in \{0,1\}$. In the context of fuzzy labels in superset learning, we consider the label space to be the universe, i.e., we set $\mathcal{U} = \mathcal{Y}$, and further assume that π is normalized in the sense that $\max_{y' \in \mathcal{Y}} \pi(y') = 1$. Then, $\pi(y')$ can be interpreted as an indicator of the plausibility that y' represents the ground truth y, thus expressing uncertainty in the beliefs about y in an epistemic way [HW21]. Here, we will let $\mathcal{F}(\mathcal{Y})$ denote the space of all fuzzy sets over the underlying target space $\mathcal{Y}$.

Let us consider the Equidae example once again, focusing on the classification of species based on images capturing a scene from a long distance. Assuming a zoologist as the annotator, a fuzzy label π as defined before could be constructed such that labels associated with the onager species, such as `Persian onager`, `Somali wild ass` or `Turkmenian kulan`, are regarded as highly plausible with $\pi(\cdot) = 1$ for the respective classes. However, due to the visual similarities between onagers and kiangs, albeit with kiangs generally being slightly larger, the annotator may find subtle indications for the image showing a kiang species. On the other hand, African wild asses, such as the `Nubian wild ass` or `Somali wild ass`, typically exhibit a grayish coat, which distinguishes them more easily from onagers. Consequently, the annotator may assign higher π values to kiang than to African wild ass labels, although not as high as for the onager labels. Naturally, some animal species, like those belonging to the subgenus Hippotigris, commonly known as zebras, would be deemed completely implausible by means of $\pi(\cdot) = 0$.

In [Hül14], generalized risk minimization by means of the data disambiguation principle as introduced earlier has been extended to the fuzzy case, i.e., for observed datasets of the form

$$\mathcal{D}_N^{\text{fuz}} := \{(\boldsymbol{x}_i, \pi_i)\}_{i=1}^N \subset \mathcal{X} \times \mathcal{F}(\mathcal{Y}) \,. \tag{3.15}$$

Fuzzy sets π can be turned into a crisp set approximation by accessing their α-cuts $[\pi]_\alpha := \{y' \in \mathcal{Y} : \pi(y') \geq \alpha\}$. This allows to extract supersets $Y_\alpha \subseteq \mathcal{S}$ that select all elements in the universe of discourse that have a membership of at least α, i.e., $Y_\alpha = [\pi]_\alpha$, which are compatible with superset learning methods on classical sets in $\mathcal{S}$ as discussed before. Nevertheless, given a fixed model $h \in \mathcal{H}$, the data disambiguation principle by means of $\mathcal{L}^*_{\text{OSL}}$ may yield different optimal instantiations $y' \in Y_\alpha$ for each $\alpha \in [0,1]$, leading to contradictory minimizers $h^* \in \mathcal{H}$ over all possible supersets Y_α. This raises the question of which hypothesis should be considered the favorable choice. To address this issue, [Hül14] applies a level-wise reduction scheme to $\mathcal{L}^*_{\text{OSL}}$ by aggregating the individual losses over all supersets Y_α in the so-called *fuzzy optimistic superset loss* (FOSL) $\mathcal{L}^{**}_{\text{FOSL}} : \mathcal{F}(\mathcal{Y}) \times \mathcal{Y} \longrightarrow \mathbb{R}$, which reads as

$$\mathcal{L}^{**}_{\text{FOSL}}(\pi, h(\boldsymbol{x})) := \int_0^1 \mathcal{L}^*_{\text{OSL}}([\pi]_\alpha, h(\boldsymbol{x})) \mathrm{d}\alpha \,. \tag{3.16}$$

In Chapters 8 and 9, we show how $\mathcal{L}^{**}_{\text{FOSL}}$ can be used to leverage fuzzy set-valued label modeling as a flexible and expressive way to model complex domain knowledge about the ground truth y.

Nevertheless, also other risk adaptations have been proposed to generalize learning methods to fuzzy labels. For instance, Campagner [Cam21] suggests to generalize the superset risk $\mathcal{R}^S(h, s^*)$ with respect to the partial 0/1 loss $\mathcal{L}^*_{0/1}$ (cf. Eq. (3.7)) by the fuzzy (empirical) risk based on $\mathcal{L}^{**}_{\text{FER}} : \mathcal{F}(\mathcal{Y}) \times \mathcal{Y} \longrightarrow \mathbb{R}$, which is defined as

$$\mathcal{L}^{**}_{\text{FER}}(\pi, h(\boldsymbol{x})) := 1 - \pi(h(\boldsymbol{x})) \,. \tag{3.17}$$

In the same work, the learnability properties of generalized risk minimization with respect to $\mathcal{L}^{**}_{\text{FER}}$ are studied. Furthermore, nearest neighbor methods have been generalized to fuzzy labels [EG10; DGH14; Wag+20; Cam21], where the classification of an instance is realized by a voting strategy based on the label plausibilities assigned to instances in the neighborhood. Furthermore, [Cam+21] studies fuzzy labels in multi-annotator settings. Finally, *generalized maximum likelihood estimation* [Den11] approaches the model selection not by evaluating the most favorable instantiation of the fuzzy targets as in Eq. (3.16), but by considering all possible instantiations simultaneously (see [Hül14] for a more comprehensive comparison to fuzzy optimistic superset learning).

Fuzzy sets can be linked to possibility theory [Zad99] as a rigorous framework for modeling uncertainty. In fact, fuzzy labels π as introduced before can be equivalently characterized as *possibility distributions* [DP98; DZ01], which allow for expressing uncertainty in an agent's beliefs about the true label y in a quantitative manner, and offer themselves as a complement to classical probability distributions. As such, the interpretation of fuzzy labels as possibility distributions allows for bridging the gap to imprecise probabilities [Qua22; Des22]. As we show in Chapter 4 and build on this in Chapters 5 to 7, beliefs in the form of possibilities $\pi(y')$ can also be seen as upper bounds on the probabilities $p^*(y')$, and thus can be used to constrain subsets in the space of all probability distributions over the target space $\mathbb{P}(\mathcal{Y})$ where the true outcome p^* is assumed. Sets of probability distributions are known as *credal sets* [Lev83], and can be specified in accordance with π by

$$\mathcal{Q}_\pi := \Big\{ p \in \mathbb{P}(\mathcal{Y}) \,|\, \forall\, Y \subseteq \mathcal{Y} : \textstyle\sum_{y \in Y} p(y) \leq \max\limits_{y \in Y} \pi(y) \Big\} \,. \tag{3.18}$$

The probabilistic nature of supersets $\mathcal{Q}_\pi$ facilitates the use of probabilistic losses employed in generalized learning methods as shown in our optimistic superset learning framework *label relaxation* (cf. Chapter 4). Needless to say, there are also other ways to represent uncertain data in an imprecise probabilistic fashion, e.g., probability boxes [DDC08] or n-monotone capacities [GS94], we refer to [Aug+14; Des22] for a broader picture.

Soft Labels in Belief Function Theory

A more general framework for reasoning with epistemically uncertain data is provided by the *belief function theory*, also called the *Dempster-Shafer* or simply *evidence theory* [Dem67; Sha76]. It generalizes subjective Bayesian probability distributions via the concept of belief functions, which encode evidence supporting certain outcomes by functions that assign probabilistic values to all elements in the power set of the possible outcomes, in our case all elements in $2^{\mathcal{Y}}$, not just to single outcomes in $\mathcal{Y}$ as in Bayesian theory. In doing so, the belief function theory allows for a more subtle characterization of the uncertain knowledge about a ground truth such as p^*, e.g., by properly incorporating evidence that supports multiple possible outcomes coincidently. We refer to [YL08] for a more comprehensive overview of the overall framework, and will focus here only on prominent methods that model incomplete supervision by means of this theory.

Formally, the belief function framework specifies a *mass* function $m : 2^{\mathcal{U}} \longrightarrow [0,1]$ over a universe $\mathcal{U}$, here the label space $\mathcal{Y}$, such that

$$\sum\nolimits_{A \in 2^{\mathcal{Y}}} m(A) = 1\,,$$

holds. It is also typically assumed to be normalized in the sense that $m(\emptyset) = 0$. Values $m(A)$ of a set $A \in 2^{\mathcal{Y}}$ express the proportion of available evidence that supports the outcomes in A. In the case of ambiguity of A with $|A| > 1$, this evidence cannot be uniquely assigned to an element $y' \in A$, but does not yield support for $\mathcal{Y} \setminus A$.

Using a given mass function m, one can then specify the belief (or support) $\mathrm{Bel}(A)$ and plausibility $\mathrm{Pl}(A)$ defined as

$$\begin{aligned} \mathrm{Bel}(A) &:= \sum\nolimits_{B \subseteq A} m(B) \\ \mathrm{Pl}(A) &:= \sum\nolimits_{B \cap A \neq \emptyset} m(B)\,, \end{aligned} \tag{3.19}$$

which quantify the degree of evidence that *surely supports* or *could potentially support* a set A, respectively. As such, if m is supposed to model the beliefs about an underlying ground truth p^*, $\mathrm{Bel}(A)$ and $\mathrm{Pl}(A)$ impose bounds on the probability $\mathrm{Pr}_{p^*}(A)$ of a probability measure Pr on $\mathcal{Y}$ induced by p^* with $\mathrm{Pr}_{p^*}(A) = \sum_{y' \in A} p^*(y')$. More precisely, their relationship can be described as

$$\mathrm{Bel}(A) \leq \mathrm{Pr}_{p^*}(A) \leq \mathrm{Pl}(A)\,. \tag{3.20}$$

It is worth noting that possibility distributions, which were previously discussed, can be derived from specific mass functions m. More precisely, if all elements $A_1, \ldots, A_k \in 2^{\mathcal{Y}}$ with $m(A_i) > 0$ for $i \in [k]$ are nested in a specific order $A_1 \subseteq A_2 \subseteq \ldots \subseteq A_k$, a contour function $\pi : \mathcal{Y} \longrightarrow [0,1]$ defined as $\pi(y) = \mathrm{Pl}(\{y\})$ based on the mass function m is in fact a possibility distribution, thus connecting the two theories of belief functions and possibilities.

As an example, consider the task of classifying zebras images according to their genus in $\mathcal{Y} = \{\texttt{plains zebra}, \texttt{mountain zebra}\}$. Although these species exhibit similarities in terms of their body structure, distinctions arise from variations in coat colors and patterns, as well as their habitat context. Let us consider an image associated with the label $y = \texttt{plains zebra}$ and explore the deployment of an automated computer vision system that leverages various visual clues to make accurate predictions. The system examines the body shape, color spectrum of the coat, and the habitat context depicted in the background, with each clue serving as a source of evidence. Given that both species in $\mathcal{Y}$ share a similar body shape, the first clue remains impartial and does not strongly support either class, thus providing full evidence for the entire set $\mathcal{Y}$. Conversely, the second clue pertaining to the color spectrum reveals subtle differences between the two species. While the distinction is not pronounced, it lends slightly stronger support to the correct label, `plains zebra`, with a weight twice that of `mountain zebra`. Finally, the evidence from the habitat context presents a substantial disparity, as plains and mountainous regions possess discernible dissimilarities. Consequently, this clue strongly favors the label `plains zebra`. Collectively, these visual clues can be fused in the form of a mass function m as presented in Table 3.1, leading to the most plausible class being the ground truth. Notably, this approach also acknowledges that the class `mountain zebra` is not entirely implausible, even though the evidence predominantly supports the correct hypothesis.

Hypothesis A on y	$m(A)$	$\mathrm{Bel}(A)$	$\mathrm{Pl}(A)$
Neither	0	0	0
`plains zebra`	$(1 + {}^{2}/_{3}) \cdot {}^{1}/_{3} = {}^{5}/_{9}$	${}^{5}/_{9}$	${}^{8}/_{9}$
`mountain zebra`	${}^{1}/_{3} \cdot {}^{1}/_{3} = {}^{1}/_{9}$	${}^{1}/_{9}$	${}^{4}/_{9}$
`plains zebra` or `mountain zebra`	${}^{1}/_{3}$	1	1

Tab. 3.1.: The resulting mass m that fuses the visual clues to yield partial supervision for a zebra image as described in the example.

Several approaches have leveraged the potential of the belief function theory in learning contexts to model incomplete and uncertain supervision. Many works refer to such label information as "soft labels" [Côm+09; Mut+19], although not to be confused with probabilistic, non-degenerate and thus "soft" labels in probabilistic classification. Among the first works, and later developed further in [DKS19], [Den95] generalizes k-nearest neighbor classification to handle partially supervised data modeled by mass functions m as specified above. Relating to this development, [DZ01] considers the case where class information is given as a possibility distribution in the formalism of belief function theory to propose a distance-based evidential classifier, followed by decision tree model adaptations in similar settings [EMS01; TEM07; JAE08]. Moreover, [Côm+09] suggests a mixture model within the framework of evidence theory to cope with partial supervision. Also, adaptations of EM learning algorithms to allow learning from ambiguous evidential data [Den13], e.g., to fit discriminant models [QDL17], have been proposed. [Den14] further discusses the construction of likelihood-based belief functions in the case of low-quality data. More recently, uncertain target representations have also been employed in transfer learning [Lv+22] or in deep learning applications such as medical image segmentation [Hua+21; HRD23].

Especially when sources of evidence are unreliable, the so-called *discounting* operation [Sha76; GC18] suggests afflicting a mass m with meta knowledge about its reliability represented by a factor $\alpha \in [0, 1]$, resulting in a discounted mass m^α defined as

$$m^\alpha(A) := \begin{cases} \alpha m(A), & \text{for } A \neq \mathcal{Y} \\ (1-\alpha) + \alpha m(\mathcal{Y}), & \text{for } A = \mathcal{Y}\,. \end{cases} \tag{3.21}$$

It is easy to see that this transformation renders m less precise the smaller α is, i.e., the more unreliable the source of evidence is assumed to be. For instance, the visual zebra detector described above could only get blurry images as an input, making its conclusion less reliable. As a result, this operation leads to a "weakened" form of (target) information, which is closely related to the motivation of our label relaxation method (see Chapter 4), as well as label smoothing [Sze+16] (Section 2.2).

Beyond classical discounting as in Eq. (3.21), also *contextualized* adaptations have been proposed [MQD05; MQD08], which discount evidence in the form of m differently for individual elements of the universe. For example, guesses from a zoologist modeled by m might be very accurate for zebras, but not for certain subspecies of spiders. Hence, the reliability for m conditioned on the subspace of horses in $\mathcal{Y}$ is high, while for spider-related classes it is low, leading to different degrees of discounting as in Eq. (3.21). This idea has been successfully used in various settings, e.g., as a learning objective to be inferred from labeled data [Mer+15] or to discount subnetworks in deep learning [Hua+22]. Moreover, the concept of discounting has also found its way into imprecise probabilistic representations, e.g., by means of credal sets [Des10; Mor18], again connecting the two fields.

3.3 Semi-Supervised Learning

In semi-supervised learning (SSL), another special case of incomplete supervision as discussed in Section 3.1.1, it is assumed to observe data for which only a fraction of instances is (precisely) labeled, while the rest is unlabeled. Hence, we observe the two sets $\mathcal{D}_L = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^L \subset \mathcal{X} \times \mathcal{Y}$ and $\mathcal{D}_U = \{\boldsymbol{x}_i\}_{i=1}^U \subset \mathcal{X}$ of labeled and unlabeled instances, respectively. While the relation between the magnitudes $|\mathcal{D}_L|$

and $|\mathcal{D}_U|$ is generally not constrained, it is often assumed that $L \ll U$, e.g., due to the economic costs of labeling data.

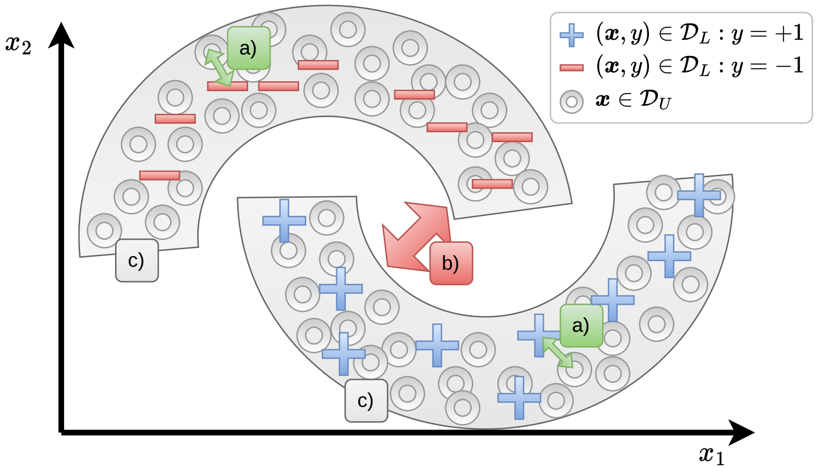
A classic example of SSL is the availability of almost unlimited amounts of image data on social media platforms. In the context of our running example, one might observe a small fraction of carefully annotated labels of Equidae species, which would require the employment of professional zoologists, which may be limited not only by financial constraints, but also by the lack of available experts. However, many tourists travel around the world, taking pictures of all kinds of species and often sharing these images online. Such unlabeled images can be considered as instances in $\mathcal{D}_U$. Another example is depth sensing for autonomous driving: While many cars have dashcams installed, e.g., for insurance reasons, built-in depth sensors are rather rare. Augmenting a curated depth estimation dataset with additional road scenes could help a model to generalize in different environments. It is not unreasonable to think that such data could provide valuable information for a model to learn in order to improve prediction quality, even without (direct) access to label information.

At an abstract level, the literature on semi-supervised learning usually distinguishes between two main tasks: *inductive* and *transductive* semi-supervised learning [EH20]. Inductive methods aim at inferring a model h as in classical supervised learning, while it is tried to leverage the provided unlabeled data $\mathcal{D}_U$ in addition to the data in $\mathcal{D}_L$ annotated with precise labels. As opposed to this, transductive methods focus more on inferring labels for the instances $\boldsymbol{x} \in \mathcal{D}_U$ specifically rather than fitting a model for subsequent prediction tasks.

In Chapters 5, 6 and 8, we present a range of method proposals that tackle the problem of inductive semi-supervised learning, which is why our focus will be on discussing aspects of this paradigm in the remainder of this section. We begin by providing a general overview of common underlying assumptions that facilitate learning in semi-supervised settings in Section 3.3.1. Subsequently, adopting the taxonomy as presented in [EH20], we describe the most relevant approaches within the inductive paradigm in Sections 3.3.2 to 3.3.4, and relate them to our contributions in this respect. For a more comprehensive overview of methods in the field of semi-supervised learning that go beyond the work as presented here, we recommend consulting [CSZ06; EH20; Yan+22].

3.3.1 Common Assumptions

The incorporation of $\mathcal{D}_U$ alongside $\mathcal{D}_L$ to enhance model performance, in comparison to training solely on $\mathcal{D}_L$, does not universally guarantee improvement [BLP08]. Therefore, methods in this regime typically rely on assumptions whose satisfaction entail favorable learning behavior. More precisely, the information to reason about $p^*_X(\boldsymbol{x})$ provided in $\mathcal{D}_U$ must entail clues about $p^*(y \,|\, \boldsymbol{x})$ [Zhu05; EH20]. As a prominent assumption that establishes a connection in this regard, the *cluster assumption* [CSZ06] states that the data forms clusters, in which all instances are likely to share the same label. This assumption is the subject of many theoretical analyses in the field [Rig07; LW07; SNZ08; BLP08; Küg+20; Zha+22c], and has recently also been reformulated as the *expansion assumption* [Wei+21] for the analysis of modern deep neural networks. Nevertheless, this assumption can be seen as a generalization of more subtle assumptions [EH20], which we will distinguish in the following. We refer the interested reader to [ML23] for a more comprehensive overview of considered settings, including stronger, more finegranular combinations of assumptions.



x_2
a)
b)
c)
c)
a)
$(\boldsymbol{x}, y) \in \mathcal{D}_L : y = +1$
$(\boldsymbol{x}, y) \in \mathcal{D}_L : y = -1$
$\boldsymbol{x} \in \mathcal{D}_U$
x_1

Fig. 3.4.: Illustration of the a) smoothness, b) low-density, and c) manifold assumption in semi-supervised learning using the two-moons classification example [Zho+03] in a two-dimensional feature space.

To illustrate the underlying assumptions in a more abstract manner, we turn to the well-known *two-moons problem* [Zho+03] as a classic example, visually depicted in Fig. 3.4. This problem comprises two intertwined semicircles of data points within a two-dimensional space, with each semicircle being associated with a distinct class label. The two moons problem serves as a widely employed toy scenario to showcase the efficacy of semi-supervised learning algorithms. By its very nature, this problem encapsulates numerous prevalent assumptions as discussed here, making it an ideal setting to demonstrate the exploitation of such assumptions by methods within the realm of semi-supervised learning.

Smoothness Assumption

As a common assumption about the structure of the feature space $\mathcal{X}$, the *smoothness assumption*, also referred to as *continuity assumption*, states that two instances $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ close to each other in $\mathcal{X}$ are likely to share the same label $y \in \mathcal{Y}$. For instance, two images showing similar appearances of zebras (roughly) share a large fraction of pixel values, so that their feature representation is (roughly) the same. This assumption allows reasoning about the label of unlabeled instances $\boldsymbol{x} \in \mathcal{D}_U$ based on labeled instances $\mathcal{D}_L$, such that the label y of instances $(\boldsymbol{x}', y) \in \mathcal{D}_L$ close to $\boldsymbol{x}$ can be transitively propagated. In Fig. 3.4 a), this smoothness can be observed for the individual semicircles, where the label information is shared in these coherent structures. This assumption has been widely accepted [CSZ06; FCL14; Luo+18a] and has laid the foundation for popular regularization methods such as consistency regularization [BAP14; SJT16; LA17], which we will discuss in more detail in Sections 3.3.3 and 3.3.4.

Low-Density Assumption

In addition to smoothness, the *low-density assumption* expresses that instances with different labels are separable through a low-density region, i.e., where p^*_X can be assumed to be low. Relating to the cluster assumption, this considers clusters to be separable by regions without instances of different labels being close to each other. As an example, consider again a computer vision model that classifies animals. In this case, it is assumed that the decision boundary of this model is located in regions of the image space whose images are very unlikely to be observed in reality, e.g., showing a black and white striped lion in a zebra enclosure.[4] Fig. 3.4 b) shows such a low-density region between the two semicircles, through which the decision boundary would be desirable. This is also an assumption that is underlying many maximum-margin supervised learning methods, e.g., support vector machines

[4] Even though we should not be surprised to see such images thanks to the advent of artistic generative neural networks.

[CV95], and has been transferred to the field of semi-supervised learning (e.g., as in [BD98; GB04; CZ05; Lee13; BS20]). There, theoretical results, e.g., for self-labeling methods (cf. Section 3.3.3), have shown the criticality of this assumption to gain any improvement in the generalization [OG21a].

Manifold Assumption

Furthermore, numerous studies build upon the so-called *manifold assumption*, which posits that the observed data in the instance space $\mathcal{X}$ (often high-dimensional) resides on a lower-dimensional manifold [BNS06]. This manifold is characterized by local Euclidean properties and homogeneity of labels. Returning to the illustrative example of the two moons depicted in Fig. 3.4, we observe that each half-moon resides on a distinct manifold, allowing the instances within them to be projected onto a one-dimensional line. The manifold assumption has served as a key motivation for the development of specialized methods, including regularization frameworks [BN06], and has been studied theoretically with respect to improvements in the generalization performance [Gol+09; Niy13; ML23], as well as the sample complexity [GLS17; ML23] over methods that use only $\mathcal{D}_L$ as training information. It is worth noting that the manifold assumption is typically combined with other assumptions to achieve increased effectiveness [LW07].

3.3.2 Unsupervised Preprocessing

As a first type of inductive semi-supervised learning methods, *unsupervised preprocessing* methods treat the data in $\mathcal{D}_U$ independently of the labeled fraction $\mathcal{D}_L$. Traditional approaches aim at extracting features from the provided feature information in $\mathcal{X}$, e.g., based on auto-encoders [Vin+08; Ras+15; ADH17], to detect patterns in the data or to learn representations by denoising perturbed inputs. This is typically accompanied by a reduction in the dimensionality, which can be motivated by the manifold assumption as discussed above [Sch+22]. Also, some domains require this step to achieve a numerical representation as a prerequisite, e.g., to transform raw textual data into vectorial representations as in natural language processing [Mik+13]. Also, clustering of both the labeled and unlabeled instances has been suggested as a preprocessing step [Gol+09], which allows to infer labels for instances in $\mathcal{D}_U$, subsequently being used as supervision to induce the final model.

In the recent years, unsupervised pre-training has emerged as a highly promising technique particularly in the realm of deep learning and has demonstrated remarkable efficacy across a wide range of practically relevant applications, including computer vision [KZB19; ND20], natural language processing [Rad+18], a multi-modal combination of the two former [Kir+23] or speech recognition [Liu+22a]. Closely related to transfer learning (see Section 3.1.3), unlabeled data is used to initialize a model for arbitrary downstream tasks, effectively guiding it toward meaningful regions within the feature space. While traditionally deep belief networks [HOT06] and stacked auto-encoders [Vin+10] attracted wide usage, modern unsupervised pre-training methods often embrace the concept of *self-supervision* [KZB19; Bal+23]. Among the myriad of techniques employed, many methods cast unsupervised pre-training as an instance classification problem, training models to discriminate instances based on data augmentations [Che+20; Cab+23]. Moreover, metric-learning based approaches have been adopted [Gri+20; Dua+21], as well as self-distillation as an unsupervised self-labeling technique [Car+21; Li+22a] (see next section). Notably, popular generative models in the domain of natural language processing, such as GPT-3 [Bro+20] or PaLM [Cho+22], are also employing unsupervised pre-training through masking techniques, followed by fine-tuning to specific tasks with smaller sets of labeled instances.

3.3.3 Self-Labeling

As another form of inductive semi-supervised learning, *self-labeling* methods [Zhu05; TGH15], also known as *self-training*, are wrapping conventional learning models by training procedures that infer labels for the unlabeled data in $\mathcal{D}_U$ based on an underlying base learner, possibly ensembles of multiple models. Thus, in the simplest case, a learning model fits a hypothesis $h \in \mathcal{H}$ on $\mathcal{D}_L$, most commonly by empirical risk minimization $\mathcal{A}_{\text{emp}}(\mathcal{D}_L)$, which is then used to generate *pseudo-labels* [Lee13] based on $h(\boldsymbol{x})$ for all or a subset of instances $\boldsymbol{x} \in \mathcal{D}_U$, subsequently being added to an updated labeled dataset $\mathcal{D}'_L \subset \mathcal{X} \times \mathcal{Y}$ via

$$\mathcal{D}'_L = \mathcal{D}_L \cup \{(\boldsymbol{x}, h(\boldsymbol{x})) \,|\, \boldsymbol{x} \in \mathcal{D}_U\} \,. \tag{3.22}$$

While deterministic predictors $h(\boldsymbol{x}) \in \mathcal{Y}$ can typically have their pseudo-labels added directly to $\mathcal{D}'_L$, probabilistic models $\widehat{p} : \mathcal{X} \longrightarrow \mathbb{P}(\mathcal{Y})$ (such as those introduced in Section 2.2 for probabilistic classification) need to either transform the prediction

$\widehat{p}(\boldsymbol{x})$ into a deterministic label via $\operatorname{argmax}_{y' \in \mathcal{Y}} \widehat{p}(y' \mid \boldsymbol{x})$ [Lee13], or to use the non-deterministic prediction $\widehat{p}(\cdot \mid \boldsymbol{x})$ directly as a "soft" label information [Ber+20]. However, since the model prediction may be flawed, not all pseudo-labels are useful and should be added to $\mathcal{D}'_L$. Criteria for determining the reliability of pseudo-labels encompass the probabilistic confidence of individual models [Soh+20], uncertainty quantification of model predictions [Riz+21] and (dis)agreement among ensemble members [ZL10]. Ultimately, once the extended labeled dataset $\mathcal{D}'_L$ is obtained, the model can undergo retraining or fine-tuning using the same routine as for the initial model training, e.g., $h' = \mathcal{A}_{\text{emp}}(\mathcal{D}'_L)$ as in the ERM case. This maintains the model's agnosticism toward the wrapping procedure, preserving the integrity of the underlying training process.

The previously described simplistic setting represents the prevalent form of self-training [Yar95], where models generate pseudo-labels to learn from their own predictions. However, more sophisticated learning procedures exist that involve collective training of multiple models, such as co-training [BM98]. In co-training, an independent coalition of models is trained, typically with different perspectives on the data, such as observing distinct subsets of features in $\mathcal{X}$ [Wan09]. Each learner within the coalition leverages the pseudo-labels predicted by the other members, resulting in higher variance among the individual base learners. This increased variance contributes to the overall robustness of these methods against noisy and potentially misleading pseudo-labels. In such settings, the hypothesis spaces $\mathcal{H}$ for the individual coalition members can also be varied, leading to ensembles formed by a range of diverse hypotheses [ZG04; WZ07]. Another line of research considers boosting approaches [GdA01; Mal+09], where ensembles of models are arranged sequentially, and pseudo-labels are propagated through the resulting chain. This sequential propagation allows for the correction of pseudo-labels in later stages of the chain, enabling relaxation of the requirements imposed on the individual models within the chain.

Thanks to the ever-increasing generalization capabilities of today's deep neural networks, e.g., due to shared pre-trained weights inferred from large-scale data (cf. Section 3.3.2), self-training is ubiquitous in recent semi-supervised learning methods. After being proposed for deep neural network training in [Lee13], it now stands as the state-of-the-art approach for harnessing large unlabeled datasets alongside smaller labeled datasets in various domains, including image classification [Soh+20; Cai+22], semantic segmentation [Wan+22c; Yan+23], object detection [Li+22b], language understanding tasks [He+20; Du+21], or speech recognition [Par+20], provided enough data even task-agnostic with a single model [Tou+23; Oqu+23].

Many different approaches to filter out unreliable pseudo-labels in the course of an incremental self-training have been suggested, including instance weighting [RYS20], curriculum learning [Zha+21a], uncertainty-aware selection [Riz+21; Wan+22b] or approximating Bayesian optimality of the pseudo-label selection [Rod+23]. As we show in Chapters 5 and 6, we also demonstrate the suitability of credal sets based on possibility distributions in this context. Additionally, self-labeling has been explored within the framework of knowledge distillation, where a teacher model generates supervision for a student model [Pha+21], and some methods even consider the teacher and student to be the same model [Che+20]. Other approaches suggest mixing pseudo-labels with targets from the (initial) labeled dataset $\mathcal{D}_L$ [Ber+19] or consider non-degenerate probability distributions adapted to the prediction history and data prior [Ber+20]. In Chapter 8, we further present a self-labeling approach for binary classification within the framework of instance weighting through data imprecisiation, which is also applied to the problem of monocular depth estimation.

3.3.4 Intrinsically Semi-Supervised Methods

In the previous subsections, we discussed techniques that precede or wrap around conventional learning algorithms. Beyond this, another category is formed by *intrinsically semi-supervised* methods [EH20], which naturally extend supervised methods to deal with unlabeled data. Our proposed methods for semi-supervised learning in Chapters 5, 6 and 8 intrinsically incorporate unlabeled data by means of a maximum-margin model formulation and a technique called consistency regularization, which we will introduce in the following. Additionally, for the sake of completeness, we provide a brief overview of manifold-based and generative methods as two additional types of intrinsically semi-supervised learning methodologies.

Maximum-Margin Models

In classification settings, *maximum-margin models* aim to identify a decision boundary that maximizes the margin between the decision function and the closest sample from each class. These methods are closely related to the low-density assumption (see Section 3.3.1), as regions with lower density p^*_X suggest placing the decision boundary there to maximize the margin [AMB05; Ben+09]. For instance, in binary classification with $\mathcal{Y} = \{-1, +1\}$, this concept can be expressed using margin losses of the form $\mathcal{L}(y, h(\boldsymbol{x})) = f(y \cdot h(\boldsymbol{x}))$, where $f : \mathbb{R} \longrightarrow \mathbb{R}$ is a non-increasing function, and the model $h : \mathcal{X} \longrightarrow \mathbb{R}$ predicts scores that allow to yield a deterministic class label $\mathcal{Y}$ via $\operatorname{sign}(h(\boldsymbol{x}))$. Support vector machines are typical models of this kind, which can be optimized in an ERM formulation with the so-called *hinge loss* $\mathcal{L}_{\text{hinge}} : \mathcal{Y} \times \mathbb{R} \longrightarrow \mathbb{R}$ [Vap98]. Formally, it is defined as

$$\mathcal{L}_{\text{hinge}}\left(y, h(\boldsymbol{x})\right) := \max\{0, 1 - y \cdot h(\boldsymbol{x})\}\,, \tag{3.23}$$

where the loss depends on how far the prediction $h(\boldsymbol{x})$ is from the correct side of the margin. Consequently, predictions $h(\boldsymbol{x}) < 1$ for $y = +1$ and $h(\boldsymbol{x}) > -1$ for $y = -1$ are penalized, urging to maintain a clear margin between instances of the positive and negative class.

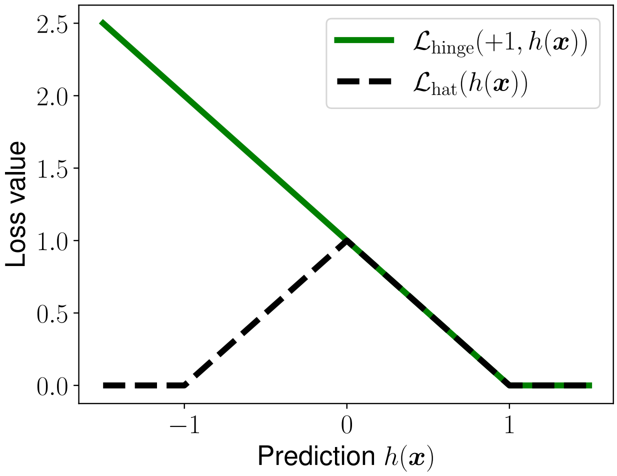


Fig. 3.5.: A comparison of $\mathcal{L}_{\text{hinge}}$ and $\mathcal{L}_{\text{hat}}$ with respect to the ground truth $y = +1$ and the corresponding prediction $h(\boldsymbol{x})$. $\mathcal{L}_{\text{hat}}$ penalizes predictions close to 0 in a symmetric manner regardless of the predicted sign, thus inherently enabling learning from unlabeled instances in $\mathcal{D}_U$.

In a semi-supervised learning setting, a natural adaptation of this loss is given by the *hat loss* $\mathcal{L}_{\text{hat}} : \mathbb{R} \longrightarrow \mathbb{R}$ [Zhu10], which ignores the label information y through

$$\mathcal{L}_{\text{hat}}\left(h(\boldsymbol{x})\right) := \max\{0, 1 - |h(\boldsymbol{x})|\}\,, \tag{3.24}$$

and can obviously also incorporate instances in $\mathcal{D}_U$. This loss, which is compared to $\mathcal{L}_{\text{hinge}}$ in Fig. 3.5, prefers decision boundaries in low-density regions, it penalizes any prediction with $-1 \leq h(\boldsymbol{x}) \leq 1$, and thus inherently enforces a large margin by exploiting the corresponding assumption. Similar adaptations of SVM margin models have been used extensively in the context of semi-supervised learning [BD98; WS07; DZZ17]. As shown in [LH16] and in our work in Chapter 8, the fuzzy optimistic superset loss $\mathcal{L}^{**}_{\text{FOSL}}$ (cf. Eq. (3.16)) entails $\mathcal{L}_{\text{hat}}$ as a special case for a specific target modeling, which connects the idea of SSL via maximum-margin models to superset learning. Moreover, Gaussian processes [LJ04] and metric learning [AC19] have also been considered to learn a discriminatory model based on unlabeled data that separates classes through low-density regions. [LZZ18] introduces further a generative framework for learning models with large margin optimization targets.

Consistency Regularization

The smoothness assumption, as defined earlier, assumes that similar instances $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}$ should have the same label. Consequently, a model h should consistently assign labels to such instances, irrespective of the actual label observation. In other words, we should expect h to predict similar labels for instances $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ in $\mathcal{D}_U$ that are close to each other in $\mathcal{X}$, even if the ground truth labels are unknown. This gives rise to the concept of *consistency regularization* (CR) [Ber+19], which encourages models to provide consistent predictions for similar instances. By enforcing consistency, the model's decision boundary is smoothed, which facilitates the alignment of regions in the feature space populated by instances from $\mathcal{D}_U$ with regions where reliable label information is available from $\mathcal{D}_L$. In essence, consistency regularization incorporates structural information about the feature space, enabling the model to make more informed predictions.

To formally characterize consistency regularization, let $\mathcal{P} \subset \mathcal{X}^{\mathcal{X}}$ denote a set of perturbation functions that apply data augmentation operations to input features. For example, in computer vision, $\mathcal{P}$ could involve filters adjusting image brightness or performing horizontal flips. Following [Wei+21], we further define the notion of a *neighborhood* $\mathcal{N} : \mathcal{X} \longrightarrow 2^{\mathcal{X}}$ around an instance $\boldsymbol{x} \in \mathcal{X}$ as a subspace in $\mathcal{X}$ that encompasses smooth transitions away from $\boldsymbol{x}$. This neighborhood can be described as

$$\mathcal{N}(\boldsymbol{x}) := \{\boldsymbol{x}' \in \mathcal{X} \,|\, \mathcal{B}(\boldsymbol{x}) \cap \mathcal{B}(\boldsymbol{x}') \neq \emptyset\}\,, \tag{3.25}$$

where

$$\mathcal{B}(\boldsymbol{x}) := \{\boldsymbol{x}' \in \mathcal{X} \,:\, \exists P \in \mathcal{P} \text{ s.t. } \|\boldsymbol{x}' - P(\boldsymbol{x})\| \leq \delta\} \tag{3.26}$$

denotes the set of instances within a distance of $\delta \in \mathbb{R}_+$ from $\boldsymbol{x}$ in the feature space. Consequently, consistency regularization at the input level involves minimizing

$$\mathbb{E}_{\boldsymbol{x} \sim p_X^*} \left[\max_{\boldsymbol{x}' \in \mathcal{N}(\boldsymbol{x})} \mathcal{L}\left(h(\boldsymbol{x}), h(\boldsymbol{x}')\right) \right]\,, \tag{3.27}$$

where $\mathcal{L}$ often represents a probabilistic loss. Thus, the goal is to ensure that predictions within the neighborhood $\mathcal{N}(\boldsymbol{x})$ are consistent, considering a reasonable distance δ. It is not difficult to see that the smoothness assumption together with the low-density assumption is critical for a homogeneous neighborhood $\mathcal{N}$. Furthermore, with respect to the common training scheme in the form of student-teacher setups [Gou+21], the consistency target $h(\boldsymbol{x}')$ in Eq. (3.27) can also be generalized as a target predicted by a teacher model h', e.g., from a different hypothesis space $\mathcal{H}' \supset \mathcal{H}$ with more model parameters as is common in knowledge distillation [HVD15], or with $h = h'$ in the self-distillation setting [Yan+22].

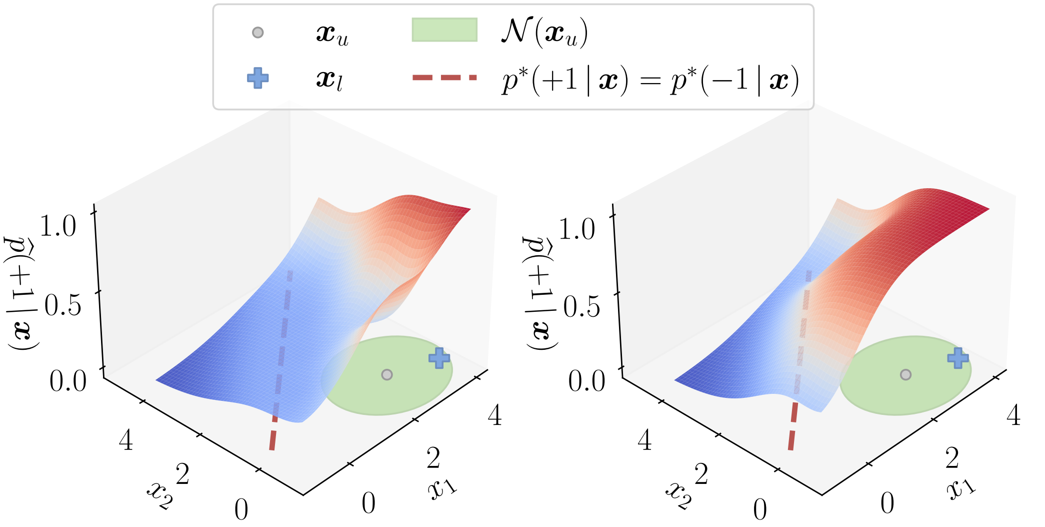


Fig. 3.6.: An illustrative example of consistency regularization leading to an improved generalization of a model $\widehat{p}$ through the consistency constraint in Eq. (3.27): For an unlabeled instance $\boldsymbol{x}_u \in \mathcal{D}_U$, CR promotes the propagation of the reliable labeling information y observed for a labeled instance $\boldsymbol{x}_l \in \mathcal{N}(\boldsymbol{x}_u)$ by enforcing consistent predictions among the instances in $\mathcal{N}(\boldsymbol{x}_u)$.

Fig. 3.6 illustrates consistency regularization in the feature space for a probabilistic classifier $\widehat{p} : \mathcal{X} \longrightarrow \mathbb{P}(\mathcal{Y})$, here in a binary classification setting with $\mathcal{Y} = \{+1, -1\}$. Think of $\widehat{p}$ as an overparameterized and thus flexible large model that can fit (almost) arbitrary concepts. For a label instance $(\boldsymbol{x}_l, +1) \in \mathcal{D}_L$, the model may already predict the correct label in the sense that $\operatorname{argmax}_{y' \in \mathcal{Y}} \widehat{p}(y' \mid \boldsymbol{x}_l) = +1$, while it is uncertain about the label of an unlabeled instance $\boldsymbol{x}_u \in \mathcal{D}_U$ (cf. the left plot), which is also an instance of the positive class. Here, $\boldsymbol{x}_l$ is assumed to be sufficiently close to $\boldsymbol{x}_u$ in the feature space such that $\boldsymbol{x}_l \in \mathcal{N}(\boldsymbol{x}_u)$ for some arbitrarily fixed δ.[5] While the learner receives a fully reliable signal from $(\boldsymbol{x}_l, +1) \in \mathcal{D}_L$, the consistency regularization

[5] In this example, we abstract from the mapping of instances in $\mathcal{X}$ to a latent feature space, which is typically done in feedforward neural networks at the penultimate layer level before feeding learned representations to a classification layer.

term in Eq. (3.27) urges $\widehat{p}$ to be consistent among all predictions in $\mathcal{N}(\boldsymbol{x}_u)$, which requires predicting the same class for $\boldsymbol{x}_u$ as for $\boldsymbol{x}_l$. The right plot in Fig. 3.6 shows the resulting smoothed model $\widehat{p}$, in which a label propagation is achieved by consistency regularization.

Among the first methods of this kind, ladder networks [Ras+15] augment a supervised classification loss with an unsupervised denoising reconstruction error for noisy perturbations $\mathcal{P}(\boldsymbol{x})$, which allows to incorporate instances from $\mathcal{D}_U$ in the learning process and promotes the learning of useful feature representations in an auto-encoder fashion. This idea was further developed by adversarial perturbations to increase the robustness of the method [Miy+19]. [Xie+20] focused on more sophisticated functions $\mathcal{P}$, such as learned policies for computer vision and natural language processing applications. Beyond this, MixMatch [Ber+19], ReMixMatch [Ber+20], FixMatch [Soh+20] and FlexMatch [Zha+21a] combined consistency regularization with entropy minimization and pseudo-label mechanisms, laying the foundation for a wide range of similar methods, e.g., as in [Kuo+20; GWL21; Kim+21; LXH21], which can also be applied to problems beyond the predominantly considered image classification [Gos+21; Yan+23; Bae+23]. Our credal self-supervised learning method proposals (see Chapters 5 and 6) build upon this as well.

However, consistency regularization as in Eq. (3.27) does not necessarily have to be defined at the input level, but can also involve perturbations $\mathcal{P}' : \mathcal{Y}^{\mathcal{X}} \longrightarrow \mathcal{Y}^{\mathcal{X}}$ at the model level [BAP14; EH20]. For example, so-called Π networks [SJT16] suggest feeding two perturbed versions of $\boldsymbol{x}$ to two randomizations of h through Dropout, rendering the consistency loss as $\mathcal{L}\left(\mathcal{P}'(h)(\mathcal{P}(\boldsymbol{x})), \mathcal{P}'(h)(\mathcal{P}(\boldsymbol{x}))\right)$ in an even more drastic regularization. Many methods also incorporate exponential moving averaged temporal ensembles of model predictions [LA17] or even model weights [TV17] over multiple iterations for $\mathcal{P}'$, enforcing consistency over the course of multiple optimization steps. Temporal aggregation of model weights has also been considered through stochastic averaging [Izm+18]. Such an averaging of model weights over a number of training iterations embedded in student-teacher setups is a common scheme in today's semi- and self-supervised learning methods [Cai+21], since model averaging as a form of regularization reduces the risk of potential confirmation biases [TV17; Ara+20]. Recently, also a combination of input and model consistency has been formulated, improving the generalization performance of previous methods in the field of computer vision [Fan+23]. Beyond the mentioned works, we refer to [Yan+22] for a more complete overview of different formulations of consistency regularization.

Manifold Models

A different research direction focuses on leveraging the manifold assumption as described in Section 3.3.1, i.e., that data lies on lower-dimensional manifolds where instances with similar labels are grouped together. Semi-supervised learning approaches aim to utilize the manifold structure to propagate labels from labeled instances within each manifold and to discover discriminative functions for separating the composed manifolds. Engelen and Hoos [EH20] distinguish two main schemes of methodologies, namely *manifold regularization* and *manifold approximation*.

Manifold regularization methods typically model the similarity of instances in terms of a geodesic distance between instances in $\mathcal{D}_L \cup \mathcal{D}_U$, which is used to induce weights for a graph Laplacian employed in a penalty term that enforces similar predictions of the model to be learned for instances with small geodesic distance [BNS06; Niy13].[6] More precisely, a regularization term penalizes disparate predictions $h(\boldsymbol{x}) \neq h(\boldsymbol{x}')$ for two (possibly unlabeled) instances $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$ whose adjacency weight in the graph is high or that have a small geodesic distance, thereby exploiting the manifold assumption of homogeneous labels in subspaces. This idea has been combined with several classical supervised learning methods, including support vector machines [BNS06; QTS12], neural networks [WRC08; Zha+20b] and generative models [ZL05; Lec+18]. Moreover, it has also been combined with model ensembling [Gen+12] and representation learning [Luo+18b], e.g., to learn similar latent embeddings for instances with the same label.

Alternatively, manifold methods can take a different approach by directly approximating the underlying manifolds themselves. These approximations can then be utilized as additional training information during the model training process. Traditionally, dimensionality reduction techniques have been employed to discover lower-dimensional representations of the data, effectively addressing the challenge of learning these subspaces [CK10]. For a more topological perspective on manifolds, Rifai et al. [Rif+11a; Rif+11b] propose the use of contractive auto-encoders, which extend classical auto-encoders by incorporating an additional penalty term on the Jacobian matrix of the output activations with respect to the input. This penalty term promotes insensitivity to small perturbations along the manifolds. Moreover, [PRA14] realizes manifold approximation by the means of learning an atlas [PRA13], which defines a collection of charts representing smaller regions with simple geome-

[6] Note that this form of regularization does not necessarily involve manifolds by assumption, but can also be generalized by the so-called *Laplacian regularization* [ST17; Cab+21].

tries. These charts serve as a basis for kernels used in support vector machines. It has also been shown that generative models, which are detailed in the next section as another intrinsically semi-supervised paradigm, are capable of effectively learning manifolds [KSF17].

Generative Models

Besides discriminative models, which focus on learning $p^*(y \mid \boldsymbol{x})$ as primarily discussed above, *generative models* aim at reconstructing $p^*(\boldsymbol{x}, y)$ directly and have also found their way into SSL applications. The reconstruction of p^*, which is supported by observations in $\mathcal{D}_U$ informing about p^*_X, allows to leverage the approximation of the joint distribution to be used for prediction purposes. [EH20] distinguishes between mixture models, generative adversarial networks and variational auto-encoders as generative SSL methods. The former assumes p^* to be composed of a combination of individual distributions per label, e.g., Gaussian distributions for each class in classification settings, whose learning employs the unlabeled information in a maximum likelihood estimation [CCC03], and have also been combined with self-training [OG21b]. However, assumptions like the correctness of the mixture model are often violated in practice [EH20], which is why we focus here on the latter two paradigms, i.e., generative adversarial networks and variational auto-encoders, as practically more relevant methods in the context of semi-supervised learning in recency [Yan+22].

Generative adversarial networks (GANs) [Goo+14] typically consist of a generator G and a discriminator model D, where G is trained to generate data points similar to p^*_X such that the discriminator D cannot distinguish them from being artificially generated by G or being observed in the training dataset. This is realized by an optimization procedure, in which both models compete to produce as real as possible data (G) and to increase the discriminative power to detect model generations (D) at the same time. While GANs in their raw form are unsupervised, several adaptations have been proposed to utilize the label information available in $\mathcal{D}_L$, e.g., for predictive tasks in classification settings. Early approaches have adapted the discriminator D to distinguish between individual classes rather than just “fake” and “non-fake” [Goo+14; Spr16], e.g., by predicting probability distributions over $\mathcal{Y}$ and an additional dummy class indicating fake images [Ode16; Sal+16]. For instances in $\mathcal{D}_U$, it is only of relevance whether the input is fake or not, so the probabilistic outputs for all classes in $\mathcal{Y}$ are aggregated and not distinguished individually.

Moreover, [Dai+17] notes that the distribution produced by G should complement the true underlying distribution p_X^*, leading to a more effective discriminator D for classification. Also, exploiting manifolds [Qi+18; Xia+20b] and applying consistency regularization [Wei+18; SBK20] has been studied to improve semi-supervised GANs. Furthermore, so-called "triple GANs" [Li+17] separate the discrimination and classification task by using two separate models, and have been the basis for several methodological advances [Gan+17; Wu+19; Liu+20c]. Since the application of GANs to semi-supervised learning is a broad area of research, we refer to [SZ22] for a more complete picture.

As another type of generative models being used for semi-supervised learning, again predominantly in classification settings, variational auto-encoders (VAEs) [KW14] model each instance $\boldsymbol{x}$ to be generated based on latent variables $\boldsymbol{z} \in \mathcal{Z}$ representing the parameters of a variational distribution, thus aiming at inducing a distribution $p(\boldsymbol{x} \mid \boldsymbol{z}) \in \mathbb{P}(\mathcal{X})$. This is usually realized in an encoder-decoder model architecture, where the encoder maps the input space $\mathcal{X}$ to the latent space $\mathcal{Z}$ by $p(\boldsymbol{z} \mid \boldsymbol{x})$ and the decoder the other way around, which can be used as a generator for new instances in $\mathcal{X}$ by certain queries in $\mathcal{Z}$. The first use of VAEs for semi-supervised learning has been in the form of a preprocessing step to learn meaningful latent representations $\boldsymbol{z} \in \mathcal{Z}$ generating the instances in both $\mathcal{D}_L$ and $\mathcal{D}_U$, which are used in a subsequent model induction [Kin+14]. In the same work, the authors further propose to model missing labels y as an additional latent to $\boldsymbol{z}$ for the generation of $\boldsymbol{x}$, so that classification can then be realized by accessing the posterior for y given $\boldsymbol{x}$, or to combine this idea with the aforementioned preprocessing by two separate latent spaces. It has also been suggested to use additional auxiliary variables [Joy+20], e.g., by modeling a class-specific latent relationship between $\boldsymbol{x}$ and y, leading to more expressive generative models [Maa+16]. Furthermore, other methodological improvements overcome the need to specify the dimensionality of $\mathcal{Z}$ in advance by using infinite mixture models [ADH17], learn disentangled representations [Nar+17; Li+19] or address multimodality [Kut+21].

3.4 Relative Comparisons as a Form of Weak Supervision

In this section, we want to highlight a special form of weak supervision that is often overlooked in overviews of weakly-supervised learning methodologies but can be found in many contemporary applications. To this end, recall a supervised learning scenario as introduced in Section 2.1, where we seek to approximate a ground truth function $g : \mathcal{X} \longrightarrow \mathcal{Y}$ for some target space $\mathcal{Y}$. Typically, quantitative feedback $y \in \mathcal{Y}$ is acquired from an oracle reflecting g, e.g., a sensor or human labeling. In such cases, it is usually assumed that g is well-defined and accessing quantitative supervision is generally feasible [AML12], at least for a fraction of data points that reasonably covers the feature and target space $\mathcal{X}$ and $\mathcal{Y}$.

However, such assumptions may be violated in practical applications. In many cases, g itself is not well-defined, i.e., it may be non-identifiable and thus ambiguously defined as a family of functions [MH10], or g is subject to incoherence [Fol92]. For instance, if g is considered to represent a (human) decision maker choosing between different options, e.g., modeled in a probabilistic manner, it has become evident that such models do not necessarily follow a well-defined subjective probability measure [Ell61], rendering the specification of supervision y in an absolute and unambiguous way impossible. But even if g is identifiable, it may be the case that quantitative data $(\boldsymbol{x}, y) \in \mathcal{X} \times \mathcal{Y}$ cannot be readily sampled, at least not at a reasonable cost. For example, humans fail to estimate magnitudes of stimuli [SBC05], leading to heavily distorted, thus unsatisfactory feedback for learning systems. Or take monocular depth estimation in images again as another example, where constraints in sensor ranges often limit the variety of sampled data, e.g., in the case of "in-the-wild" scenes [Che+16]. While humans are again usually unable to provide meaningful (precise) estimates of y in this scenario, precise label information might be gathered from proxy oracles, such as flow predictions used for depth prediction [Xia+18]. However, this type of supervision may not align with the underlying scale of g, or introduce further biases propagated from the proxy task.

To address the aforementioned shortcomings, *relative* or *ordinal comparisons* [Xia19] have been proposed as another form of weak supervision in which qualitative instead of quantitative information is provided. Such qualitative information can be observed at different levels, e.g., between individual labels in $\mathcal{Y}$ for a fixed instance $\boldsymbol{x} \in \mathcal{X}$, also known as *label ranking* [FH10]. Here, we focus on comparisons between different instances in $\mathcal{X}$ without having access to any label information beyond the instance-wise comparison, a setting called *object ranking* [ZV11]. We will refer to this setting as a particularly illustrative case of a form of weak supervision, which we have adopted in our work in Chapter 10. More precisely, data is provided in the form of orders $\boldsymbol{x}_1 \succ \boldsymbol{x}_2 \succ \ldots \succ \boldsymbol{x}_n$ between instances $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n \in \mathcal{X}$ reflecting comparisons $g(\boldsymbol{x}_1) > g(\boldsymbol{x}_2) > \ldots > g(\boldsymbol{x}_n)$, without having direct access to g or a numerical proxy of it.[7] Since supervision is limited to the comparisons of objects rather than associating them with precise numerical values, the requirements on an oracle used for data sampling is reduced, which is economically beneficial. At the same time, it promotes a more cautious and uncertainty-aware representation of knowledge, as a relative label information $\boldsymbol{x} \succ \boldsymbol{x}'$ now only expresses a "regional" relationship between $g(\boldsymbol{x})$ and $g(\boldsymbol{x}')$, i.e., the former precise value lies *somewhere* higher than the latter, thus taking advantage of a certain tolerance.

As an example for object rankings as a form of weak supervision, suppose the ground truth $g : \mathcal{X} \longrightarrow \mathbb{R}$ to represent a human's sense of taste for edible fruits described by feature representations in $\mathcal{X}$, where the degree of taste is given in a numerical form. Arguably, g is a complex sensory system that is usually not directly accessible, and humans themselves are not capable of quantifying their own taste in an absolute way.[8] Nevertheless, it is not entirely unreasonable to assume that g can be seen as a function that maps fruits to a numerical taste value $y \in \mathbb{R}$ [Web+15], albeit not in an identifiable way due to scale-invariance, i.e., there is no profound range of values to which the target domain must be necessarily fixed. Hence, supervision in the form of tuples $(\boldsymbol{x}, y)$ obtained from humans with precise estimates y becomes infeasible to be acquired or is arbitrarily misleading, e.g., a human would possibly have to adjust all previously constructed examples $(\boldsymbol{x}, y)$ when a new fruit not yet considered is queried. Instead, humans are usually able to give relative feedback, for instance that they prefer apples over bananas. This way, humans only have to express that their taste g is higher for one option than for another one, without making too specific assumptions about the underlying numerical quantification. At the same time, this oracle is also more robust to biases due to uncertainty, further supporting the appropriateness of this form of weak supervision. Fig. 3.7 schematically illustrates this example of forming weak supervision as a proxy for inaccessible absolute feedback.

3.4.1 Learning from Relative Comparison

When dealing with relative comparisons, many approaches treat them as *preferences* [Dom+11] from a subjective point of view, just as in the fruit example above. This gives rise to so-called *preference learning* [FH10], in which comparisons $\boldsymbol{x}_1 \succ \boldsymbol{x}_2$ are

[7] Different relative comparison lengths n have been considered in the literature. We speak of *pairs* for $n = 2$, while comparisons with $n > 2$ are called *lists* [Liu11].

[8] Here, we simplify the concept of taste for the sake of illustration, abstracting from more subtle basic tastes such as sweetness, sourness, or saltiness.

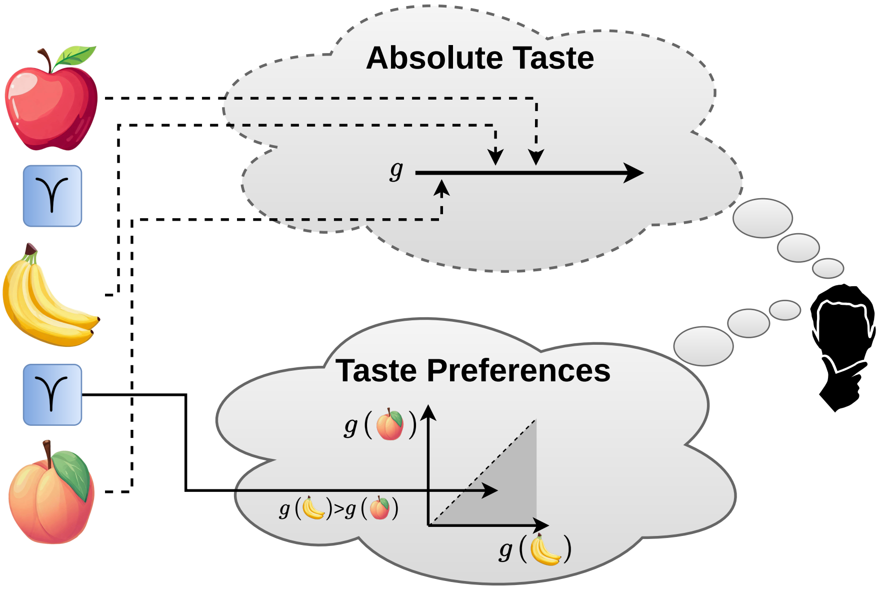


Fig. 3.7.: Schematic illustration of preferences as a weak yet accessible form of feedback: A human decision maker might have an absolute sense of taste for fruits modeled by a function g, which is not directly accessible due to insufficient ability of introspection (dashed lines). However, preference relations (blue) that compare evaluations of g for different fruits are easier to acquire, as they relieve from committing to specific precise numerical evaluations of g. Instead, regions that comparing the evaluations of g for different fruits are to be determined, leading to accessible feedback in the form of pairwise relations (solid line).

interpreted as $\boldsymbol{x}_1$ being a more "desirable" choice than $\boldsymbol{x}_2$ [Bra08b]. As this paradigm encompasses a wide range of learning tasks and methods, most notably in the realm of *learning-to-rank* [Liu11]. Here, we will only briefly discuss approaches in this field that relate to our view of relative comparisons as a form of weak supervision. For a more complete overview, we refer to [FH10].

Utility Functions

From a subjective point of view, modeling preferences with respect to an underlying notion of "usefulness", where the subjective value of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ is compared in this respect, has been a major scheme in preference learning. In economics and decision theory, this has been modeled by the so-called *utility* $u : \mathcal{X} \longrightarrow \mathbb{R}$ [Fis70; DDH83], which assigns instances in $\mathcal{X}$ a numerical score of the aforementioned usefulness. Thus, relations between instances $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}$ are then based on the sign of $u(\boldsymbol{x}_1) - u(\boldsymbol{x}_2)$, which is completely in line with g in the fruit example. Similar to the scenario there, humans are generally unable to precisely estimate u, but are willing to compare multiple items based on their perceived utility.[9]

Learning utility functions from relative comparisons has been of interest in many settings. Among the first, [Tes88] proposes a symmetric neural network that takes two objects as input, which then directly predicts the preference relation. [Her+98] approaches learning a scalar utility using a support vector machine adaptation. [Joa02] applied a similar method to search engines, and several methodological adaptations have been proposed since [KKA10; Are10; WM16; Yil+19]. Another prominent class of models for learning utility functions is the *random utility model* (RUM) [Mcf80], which assumes that the utility for each item $\boldsymbol{x} \in \mathcal{X}$ is characterized by an unknown distribution, and the utility assignment, e.g., by a human decision maker, is subject to a sampling from this distribution.

A popular example of such a model is the so-called *Plackett-Luce* (PL) model [Luc59; Pla75], which specifies a probability distribution over ranking data by means of an internal latent utility quantification. More precisely, it is characterized by a parameter vector $\boldsymbol{v} = (v_1, \dots, v_I) \in \mathbb{R}_+^I$ of I utility scores, where I is the number of items to rank. For a ranking of items τ with $\boldsymbol{x}_{\tau(1)} \succ \boldsymbol{x}_{\tau(2)} \succ \dots \succ \boldsymbol{x}_{\tau(I)}$, e.g., options to choose from that follow an underlying unknown utility function g as described above, where v_i represents the latent utility for option $\boldsymbol{x}_i$, the ranking probability for τ under the PL model with parameters $\boldsymbol{v}$ is given by

$$\Pr(\tau \mid \boldsymbol{v}) = \prod_{i=1}^{I} \frac{v_{\tau(i)}}{\sum_{l=i}^{I} v_{\tau(l)}} \,. \tag{3.28}$$

The PL model parameters $\boldsymbol{v}$ are typically derived by maximizing the likelihood $\Pr(\tau \mid \boldsymbol{v})$ of observed preference relations τ [Xia+08; MG15; ZPX16; Oos21; Oos22; NZ23], thus being fully compatible with empirical risk minimization using a loss formulation [Xia+08]. In Chapter 10, we describe an application of such a model for the purpose of depth estimation, allowing to fit a scale-invariant metric depth predictor to use relative comparisons of individual pixels with respect to their depth values as weak supervision to solve a regression task. Beyond this, PL models have also been considered in the context of superset learning [MT17; FHC17; Liu+19a; ZX19], combining the two fields of relative and incomplete supervision.

[9] It should be noted that the decision of a human may be context-dependent on the set of items it is presented. This is subject of studies in *choice theory* [BLP98], from which we abstract here for the sake of simplicity.

Preference-Based Reinforcement Learning

Closely connected to learning utility functions, the specification of reward models in reinforcement learning to measure an agent's performance poses a complex problem in many real-world applications, as it requires extensive domain knowledge and affects the learning process. To address this issue, *preference-based reinforcement learning* [Wir+17] has emerged to induce reward models based on preference feedback, making the reward model itself an additional subject of the learning problem. This way, weaker supervision is leveraged to bypass the tedious specification of reward models in a data-driven manner. For preferences acquired from humans, this problem is also known as *reinforcement learning from human feedback* (RLHF) [Chr+17; Zie+19]. Learning approaches include inducing policies that guide the trajectory sampling directly from preferences [WFT12], or learning a surrogate model in the form of a preference model [Für+12; WF13] or a utility function [WFN16; Chr+17] as an intermediate step to policy learning. Methods of this kind have demonstrated their effectiveness for many real-world problems [Wir+17; KBH23], including RLHF as a technique for aligning large-language models in the domain of natural language processing to realize dialog systems [Ouy+22], for instance as in ChatGPT [@Ope22].

Other Applications

Besides the aforementioned application domains, relative comparisons have been used in various other domains. For instance, many methods for learning similarity [Ken48; HZW13] and metrics [SJ03; AGU15; XD20] have relied on relative supervision, as specifying quantitative supervision for problems of this kind is cumbersome. Also, comparisons have been considered as a form of oracle feedback in active learning [Xio+15; Her+21], thus weakening the assumptions on employed oracles. From an application point of view, relative supervision has proven to be effective in many different settings, including depth estimation [Zor+15; Che+16; Ewe+17; Xia+20a], reflectance learning [NMY15], fine-grained classification [Wah+14], person recognition [Sad+01; ZGX13], recommender systems [Neg+18] or document summarization [GMG20].

In the context of regression problems, the integration of precise numerical supervision with relative comparisons as an additional (weak) source of label information has been explored, too. A notable approach in this regard is known as *combined* *regression and ranking* [Scu10]. This approach formulates an optimization criterion that combines a conventional regression task with a ranking loss, assigning appropriate weights to each, thereby yielding improvements in both regression- and ranking-based performance metrics. This framework is particularly advantageous in scenarios involving rare events or skewed distributions, where it outperforms standard approaches by exploiting the constraints imposed by the relative comparisons. The combined regression and ranking approach has also been tailored to specific regression domains, such as depth estimation [LS18] and algorithm selection [Han+20]. Furthermore, its applicability extends beyond regression tasks, e.g., to improve model calibration [She+22].

4 From Label Smoothing to Label Relaxation

Author Contribution Statement

Inspired by previous works of Eyke Hüllermeier and under his guidance, the author developed the idea of label relaxation to apply optimistic superset learning in the context of probabilistic classification. The paper was initially written by the author, and subsequently revised by both authors. Furthermore, the implementation and experimentation was done by the author.

Supplementary Material

An appendix to the paper is provided in Appendix A. The code of the official implementation is provided at `https://github.com/julilien/LabelRelaxation` (Apache 2.0 license).

From Label Smoothing to Label Relaxation

Julian Lienen, Eyke Hüllermeier
Heinz Nixdorf Institute and Department of Computer Science
Paderborn University
33098 Paderborn, Germany
{julian.lienen,eyke}@upb.de

Abstract

Regularization of (deep) learning models can be realized at the model, loss, or data level. As a technique somewhere in-between loss and data, label smoothing turns deterministic class labels into probability distributions, for example by uniformly distributing a certain part of the probability mass over all classes. A predictive model is then trained on these distributions as targets, using cross-entropy as loss function. While this method has shown improved performance compared to non-smoothed cross-entropy, we argue that the use of a smoothed though still precise probability distribution as a target can be questioned from a theoretical perspective. As an alternative, we propose a generalized technique called label relaxation, in which the target is a set of probabilities represented in terms of an upper probability distribution. This leads to a genuine relaxation of the target instead of a distortion, thereby reducing the risk of incorporating an undesirable bias in the learning process. Methodically, label relaxation leads to the minimization of a novel type of loss function, for which we propose a suitable closed-form expression for model optimization. The effectiveness of the approach is demonstrated in an empirical study on image data.

Introduction

In standard settings of supervised learning, the result of a learning process is essentially determined by the interplay of the model class, the learning algorithm resp. the loss function this algorithm seeks to minimize in order to identify the presumably optimal model, and the training data. Of utmost practical importance, especially for flexible models such as neural networks, is a regularization of the learner, so as to prevent it from overfitting the training data. In the case where models are non-deterministic and produce probabilistic predictions, for example class probabilities in the case of classification, overfitting also manifests itself in overly confident predictors with a tendency to assign probabilities close to the extremes of 0 or 1.

While the role of the model class and the loss function in helping to regularize the learner is quite obvious, this is arguably less true for the training data. The role of the data becomes especially important when supervision is only indirect in the sense that the target of the predictor is not directly observed. Again, class probabilities constitute an important example: Even if probabilistic predictions are sought, the data will normally not provide such probabilities as training information. Instead, it typically consists of examples with a single class label attached. In such cases, the formalization of the learning problem also involves the *modeling of the training data*.

This aspect, the modeling of data, is at the core of this paper. Compared to the model class and learning algorithm, it has received rather little attention in the literature so far, and is mostly done in an implicit way. For example, when training a model by optimizing losses such as log-loss or Brier score, an observed class label is implicitly treated as a degenerate (one-point) distribution, which assigns the entire probability mass to that label. Unsurprisingly, feeding the learner with extreme distributions of that kind aggravates the problems of overfitting and over-confidence.

So-called *label smoothing* (Szegedy et al. 2016) has recently been proposed to address these issues. The idea is to remove a certain amount of probability mass from the observed class and spread it across the other classes, thereby making the distribution less extreme. While probability mass can be spread in any way, the authors suggest a uniform distribution over all classes. This modification of the data encourages the model to be less confident about the predictions, which effectively narrows the gap between the logits of the observed class and the others. This method has proved successful in various applications, such as classification of image data using deep convolutional neural networks (Szegedy et al. 2016; Müller, Kornblith, and Hinton 2019).

Although label smoothing undoubtedly provokes a regularization effect (Lukasik et al. 2020), it can be questioned from a data modeling point of view. In particular, since the smoothed probability distribution is still unlikely to match the true underlying conditional class probability, it is likely to introduce a bias that may harm the generalization performance (Li, Dasarathy, and Berisha 2020). Indeed, while label smoothing helps to calibrate the degree of confidence of a model, and improves compared to the use of the conventional cross-entropy loss with one-point probabilities, explicit calibration methods such as temperature scaling (Guo et al. 2017) turn out to calibrate models even better (Müller, Kornblith, and Hinton 2019).

In this paper, we propose *label relaxation* as an alternative approach to data modeling. To avoid a possibly undesirable bias, the key idea is to replace a degenerate probability distribution associated with an observed class label, not by a single smoothed distribution, but by a larger *set* of candidate distributions. All distributions in this set still assign the highest probability to the observed class, but the concrete degree is not fixed, and the remaining mass can be distributed freely over the other classes. This way, the learner itself can decide on the most appropriate distribution. In other words, instead of predetermining an alleged ground-truth distribution as a target, this distribution will be determined in a data-driven way as a result of the learning process itself.

To put label relaxation into practice, we devise a suitable generalization of the Kullback-Leibler (KL) divergence loss, which is able to compare a predicted probability distribution with a class of candidate distributions. The effectiveness of learning by minimizing this generalized loss is demonstrated on commonly used image datasets. While being competitive to label smoothing and other related regularization techniques in terms of classification performance, label relaxation does indeed improve in terms of calibration, i.e., accurate estimation of probabilities, often even compared to explicit calibration techniques that require extra data.

Related Work

As already said, different actions could be taken to improve learning, including the manipulation or modification of the original training data. In this regard, one can distinguish between methods acting on the instance, feature, and label level. While there are approaches to augment the instance set, e.g., by synthetically generating additional training examples (Cireşan et al. 2010; Krizhevsky, Sutskever, and Hinton 2012), or to introduce noise in the input features (e.g., (Vincent et al. 2008; van der Maaten et al. 2013)), our focus is on the adjustment of the labels.

One of the most prominent approaches of this kind, label smoothing (Szegedy et al. 2016), was already mentioned. It turns deterministic observations of class labels into probability distributions, assigning a predefined amount of probability mass to the non-observed classes; by default, a uniform distribution is used for that purpose. The newly generated targets are then used together with a conventional cross-entropy loss. As a result, the learner no longer tries to perfectly predict the original class (with probability 1), which may lead to drastic differences among the class logits and cause numerical instabilities. Label smoothing has been applied successfully not only to the domain of image classification, but also to other domains such as machine translation. More recently, Müller, Kornblith, and Hinton (2019) observed a calibration effect produced through label smoothing. Moreover, the authors analyzed the activation patterns in penultimate layers in neural networks. Apparently, label smoothing supports a regular distribution of the classes in these layers, in which clusters of instances associated with a class are well separated and tend to be equi-distant. Additionally, label smoothing has also turned out to be effective against label noise (Lukasik et al. 2020).

While label smoothing in its original form distributes probability mass to the non-observed classes uniformly, distributions other than uniform are of course conceivable. Although the work does not directly build upon label smoothing, (Hinton, Vinyals, and Dean 2015) follows a similar approach. Here, a teacher network predicts target probabilities which are then used for training a student network to *distill* the teacher's knowledge. In a different approach, a bootstrapping technique is proposed that makes use of the model's own distribution to adjust the training labels (Reed et al. 2015). Similarly, self-distillation approaches gathering the target labels from the model itself have shown regularizing effects to improve generalization performance (Zhang et al. 2019; Yun et al. 2020).

As an alternative approach to prevent the model becoming too overconfident and closely related to label smoothing, Pereyra et al. (2017) propose the penalization of confident distributions by adding the negative entropy of the predicted distribution to the original loss. Following this principle, Dubey et al. (2018) transfer the method to fine-grained classification. Related to this, with similar effects, the so-called focal loss (Lin et al. 2020) aims to reduce the loss for "well-classified" instances, i.e., predictions close to the actual target, by dynamically scaling cross-entropy loss. With this, designed to cope with class imbalance in object detection problems, the danger of overconfidence is reduced by flattening the loss near the true target and, thereby, shrinking the gradients for confident predictions.

In addition to the approaches outlined above, further ideas to adjust the given labels in order to achieve better generalization properties can be found in the literature. For instance, the approach by Xie et al. (2016) randomly flips targets with a fixed probability, resulting in training on a dataset ensemble with shared weights. As a result, an averaging effect lowers the risk of overfitting. Motivated by (Hinton, Vinyals, and Dean 2015), Li et al. (2017) propose a related distillery approach considering noisy side information. Bagherinezhad et al. (2018) describe a model that iteratively refines label probabilities from previous model predictions in a chain of multiple networks. By refining the labels over all models, data is augmented by soft targets to prevent overfitting.

In neural network learning, the predicted probabilities should ideally match the true distribution. However, as shown empirically by Guo et al. (2017), modern neural networks tend to be calibrated very poorly. While there exists a wide range of calibration methods, including isotonic regression (Zadrozny and Elkan 2002), Bayesian binning techniques (Naeini, Cooper, and Hauskrecht 2015), or beta calibration (Kull, Filho, and Flach 2017), a simple technique called temperature scaling proved to provide strong performance compared to its competitors (Guo et al. 2017). However, most calibration methods require additional data to determine the calibration parameters, being left with less data for training the model. Typically, this comes with a loss of generalization performance. Although label smoothing reduces the calibration error compared to non-smoothed training, it is still slightly inferior to temperature scaling (Müller, Kornblith, and Hinton 2019).

Label Relaxation

In the following, we detail our idea of label relaxation as an alternative to label smoothing, i.e., the idea of modeling deterministic data, namely observed class labels, in terms of a set of probability distributions instead of a single target distribution. We also propose a generalization of an underlying loss function, which compares probabilistic predictions with a set of candidate distributions, and derive a closed-form expression for the case of the KL divergence.

Motivation

Consider a conventional setting of supervised learning, in which we are interested in learning a probabilistic classifier $\hat{p} : \mathcal{X} \longrightarrow \mathbb{P}(\mathcal{Y})$, where $\mathcal{X}$ is an instance space, $\mathcal{Y} = \{y_1, \ldots, y_K\}$ a set of class labels, and $\mathbb{P}(\mathcal{Y})$ the space of probability distributions over $\mathcal{Y}$. To this end, we typically proceed from training data $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N \subset \mathcal{X} \times \mathcal{Y}$, i.e., observations in the form of instances labeled by one of the classes. Thus, even if we assume a ground-truth (conditional) probability distribution $p_i^* = p^*(\cdot \,|\, \boldsymbol{x}_i)$ to exist for each $\boldsymbol{x}_i \in \mathcal{X}$, this distribution will normally not be provided as training information. Instead, the training will be based on the deterministic label y_i, which is (explicitly or implicitly) treated as a degenerate (one-point) distribution $p_i \in \mathbb{P}(\mathcal{Y})$ such that $p_i(y_i \,|\, \boldsymbol{x}_i) = 1$ and $p_i(y \,|\, \boldsymbol{x}_i) = 0$ for $y \neq y_i$.

Needless to say, making the realistic assumption of a non-deterministic dependency between $\mathcal{X}$ and $\mathcal{Y}$, the true distribution p_i^* will normally be less extreme than the surrogate p_i. Therefore, providing the former as training information may suggest a level of determinism that is actually not warranted. As a consequence, the learner will be encouraged to make extreme predictions, which suggests a high degree of confidence, leading to biased probability estimates and a tendency to overfit the training data — all the more when training flexible models such as neural networks.

In label smoothing, a surrogate distribution p is replaced by a less extreme surrogate $p^s = (1 - \alpha)\, p + \alpha\, u$ as a target for the learner, where $u \in \mathbb{P}(\mathcal{Y})$ is a fixed distribution and $\alpha \in (0, 1]$ a smoothing factor. As shown by Szegedy et al. (2016), the resulting cross-entropy H, which often serves as a loss function for the learner, for a prediction $\hat{p} \in \mathbb{P}(\mathcal{Y})$ is of the form

$$\begin{aligned} H(p^s, \hat{p}) &= (1 - \alpha) H(p, \hat{p}) + \alpha H(u, \hat{p}) \qquad (1) \\ &= (1 - \alpha) H(p, \hat{p}) + \alpha \left(D_{KL}(u || \hat{p}) + H(u) \right) \ . \end{aligned}$$

Since $H(p) = 0$ for a degenerate p, the first term on the right-hand side simplifies to $H(p, \hat{p}) = D_{KL}(p || \hat{p}) + H(p) = D_{KL}(p || \hat{p})$, with D_{KL} the Kullback-Leibler divergence. Moreover, assuming u to be independent of $\hat{p}$, $H(u)$ can be treated as a constant with no influence on loss minimization. Furthermore, as pointed out by Szegedy et al. (2016), the divergence $D_{KL}(u || \hat{p})$ essentially corresponds to the negative entropy of $\hat{p}$ for the case where u is the uniform distribution. Thus, we eventually end up with a loss function of the form

$$L(p^s, \hat{p}) = (1 - \alpha)\, D_{KL}(p || \hat{p}) + \alpha\, H(\hat{p}) \ , \qquad (2)$$

i.e., a loss that augments the original cross-entropy loss by a penalty that enforces a higher entropy for the prediction $\hat{p}$, and which hence serves the purpose of regularization, as empirical results have confirmed (Pereyra et al. 2017). Training a learner with the loss (2) will obviously lead to less extreme predictions (for $\alpha = 1$, the learner will always predict the uniform distribution on $\mathcal{Y}$).

Thus, there are different ways of looking at label smoothing. According to what we just explained, it can be seen as a regularization technique, which may explain its practical usefulness. On the other side, coming back to the discussion we started with, it can also be seen as an attempt at presenting the training information in a more "faithful" way: A smoothed target probability p^s is arguably more realistic than a degenerate distribution p assigning the fully probability mass to a single class label.

However, it is still unlikely that the adjusted distribution p^s matches the ground-truth p^*. Therefore, using p^s as a more or less arbitrary target, the learner will still be biased in a possibly undesirable way. Related to this, one may wonder whether a systematic penalization of the learner for "overly correct" predictions, i.e., predictions $\hat{p}$ that are closer to the original p (the truly observed class label) than p^s, is indeed appropriate. At least in some cases, such predictions could be justified, and indeed be closer to the ground-truth.

As a presumably better but at least more faithful representation of our knowledge about the ground-truth p^*, we propose to replace the original target p by a *set* $Q \subset \mathbb{P}(\mathcal{Y})$ of candidate probabilities that are "sufficiently close" to the original target p. While the replacement of p by a single distribution p^s can be seen as a distortion of the original target, this can be considered as a *relaxation* of the target: As long as the learner predicts any distribution $\hat{p} \in Q$ inside the candidate set, it should not be penalized at all, i.e., the loss should be 0. This is to some extent comparable to the use of loss functions like the ϵ-insensitive loss in support vector regression, where the loss is 0 in the ϵ-neighborhood of the original target; essentially, this means that the original target, which is a real number, is relaxed and replaced by a set in the form of an interval.

Note that, by using set-valued targets Q_i for the training instances $\boldsymbol{x}_i$, a regularization effect can also be expected: By accepting all predictions as correct that are sufficiently close to the original target distribution, the learner is still allowed to produce extreme predictions but no longer urged to do so. Instead, the learner is more flexible and can freely choose a target $p_i^r \in Q$ that appears most appropriate. Since the p_i^r are the result of a learning process and determined in a data-driven way, one expects them to be closer to the p_i^* than the surrogates p_i^s, which are chosen arbitrarily. This is completely in line with the idea of *data disambiguation* in the context of learning from set-valued data (Hüllermeier and Cheng 2015). See Fig. 1 for an illustration of the conceptual differences between label smoothing and label relaxation.

Loss Formulation

To formalize the ideas sketched above, we leverage the theory of imprecise probabilities (Walley 1991). A convenient way to express a set of probability distributions is to provide *upper probabilities*, i.e., upper bounds on the probabilities of events. So-called *possibility distributions* $\pi : \mathcal{Y} \longrightarrow [0, 1]$

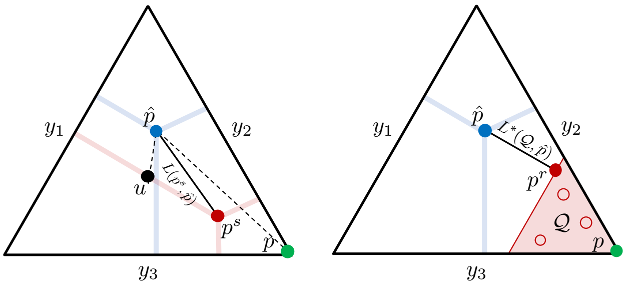


Figure 1: An illustration of label smoothing (left) and label relaxation (right), using a barycentric representation, in which points correspond to (3-class) distributions and probabilities are given by the lengths of the projections to the sides of the triangle. In the former, the original distribution p (in the lower right corner) is shifted toward the uniform distribution u, and the loss of a prediction $\hat{p}$ depends on the (KL) distance to p^s, hence on the distance to p as well as u. In label relaxation, p is replaced by a set $\mathcal{Q}$ of distributions, indicated by the shaded region. The learner is free to choose any of the distributions inside this set (non-filled circles inside the region), and the loss is determined by the minimal distance between $\hat{p}$ and any of the distributions in $\mathcal{Q}$ (filled red circle).

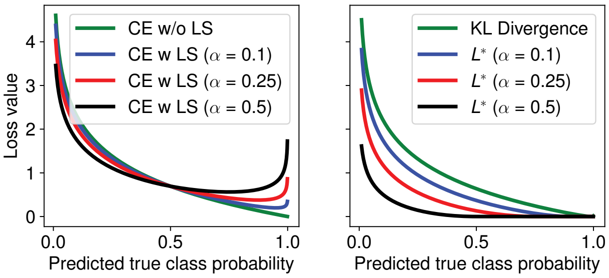


Figure 2: A comparison of the different losses discussed in this paper for binary classification. Left: Cross-entropy losses with and without label smoothing. Right: LR loss L^* based on the Kullback-Leibler divergence.

are often interpreted in this way (Dubois and Prade 2004), i.e., in the sense that $\pi(y)$ is an upper bound on $p^*(y)$. More generally, since a possibility distribution π induces a measure Π on $\mathcal{Y}$ defined by $\Pi(Y) = \max_{y \in Y} \pi(y)$ for all $Y \subseteq \mathcal{Y}$, the set of probability distributions associated with a distribution π is given by

$$Q_\pi := \left\{ p \in \mathbb{P}(\mathcal{Y}) \,|\, \forall\, Y \subseteq \mathcal{Y} : \sum_{y \in Y} p(y) \leq \max_{y \in Y} \pi(y) \right\} \enspace .$$

Note that a possibility distribution π is assumed to be normalized in the sense that $\pi(y) = 1$ for at least one $y \in Y$. In other words, there is at least one alternative that appears completely plausible. In our case, this alternative naturally corresponds to the class label that has actually been observed for a training instance: Potentially, this class may have a (conditional) probability of 1.

However, by assigning a certain degree $\pi(y) > 0$ of possibility also to the other classes, we can express that these classes are not completely excluded either. More specifically, consider a distribution of the following kind:

$$\pi_i(y) = \begin{cases} 1 & \text{if } y = y_i \\ \alpha & \text{if } y \neq y_i \end{cases} \enspace ,$$

where $\alpha \in [0, 1]$ is a parameter. By definition, the associated set $Q^\alpha_{\pi_i}$ is then given by the set of probability distributions p that assign a probability mass of *at most* 1 to the observed class y_i and *at most* α to the other classes:

$$Q^\alpha_i := \left\{ p \in \mathbb{P}(\mathcal{Y}) \,|\, \sum_{y_i \neq y \in \mathcal{Y}} p(y) \leq \alpha \right\} \qquad (3)$$

Replacing the class labels y_i observed as training information by sets Q^α_i as new targets for the learner, we need to define a suitably generalized loss function L^*. Since the learner is still assumed to produce probabilistic predictions, the loss should be able to compare a predicted distribution $\hat{p}_i$ with a candidate set Q^α_i. According to what we said before, namely that a prediction inside Q^α_i should be considered as perfect, a natural definition is

$$L^*(Q, \hat{p}) := \min_{p \in Q} L(p, \hat{p}) \enspace , \qquad (4)$$

where L is a standard loss on probability distributions, i.e., a loss $L : \mathbb{P}(\mathcal{Y})^2 \longrightarrow \mathbb{R}$. Interestingly, (4) can be seen as a special case of what has been introduced under the notion of *optimistic superset loss* in the context of superset learning (Hüllermeier and Cheng 2015), and more recently as *infimum loss* by Cabannes, Rudi, and Bach (2020). In the following, we shall refer to (4) as *label relaxation* (LR) loss.

As a theoretically convenient case, instantiating L with the Kullback-Leibler divergence, that is,

$$L(p, \hat{p}) := D_{KL}(p||\hat{p}) = \sum_{y \in \mathcal{Y}} p(y) \log \frac{p(y)}{\hat{p}(y)} \enspace ,$$

(4) simplifies as follows for sets Q^α_i of the form (3):

$$L^*(Q^\alpha_i, \hat{p}_i) = \begin{cases} 0 & \text{if } \hat{p}_i \in Q^\alpha_i \\ D_{KL}(p^r_i||\hat{p}_i) & \text{otherwise} \end{cases} \enspace , \qquad (5)$$

where

$$p^r_i(y) = \begin{cases} 1 - \alpha & \text{if } y = y_i \\ \alpha \cdot \frac{\hat{p}_i(y)}{\sum_{y' \neq y_i} \hat{p}_i(y')} & \text{otherwise} \end{cases} \enspace . \qquad (6)$$

We refer to the technical appendix for a formal proof of this result.

Fig. 2 shows a comparison of label smoothing (cross-entropy losses with and without smoothing, left side) with our label relaxation loss (right side). As can be seen (and is proven in the appendix), L^* based on the Kullback-Leibler divergence is convex, which makes the optimization computationally feasible. Moreover, label smoothing is not monotone and again increases for predictions close to 1, while the LR loss vanishes for values $\geq 1 - \alpha$. This cut reflects the relaxation of the problem. For multi-class problems, since the proposed label relaxation loss projects the predicted probabilities from $\hat{p}_i$ according to its own distribution to p^r_i, it is invariant to the concretely predicted probabilities for classes not equal to the observed class.

Interestingly, the resulting losses as shown in Fig. 2 for varying α parameters seem to be very related to the focal loss as introduced in (Lin et al. 2020). However, while the focal loss deemphasizes predictions in the "well-classified" region by *almost* flat regions, our loss completely eliminates the loss in such a region for the genuine relaxation.

Evaluation

To demonstrate the effectiveness of label relaxation, an empirical evaluation on image classification datasets assessing the classification performance and calibration is conducted.

Experimental Setting

Within the empirical evaluation of our method proposal, we compare models trained by conventional cross-entropy (CE), label smoothing (LS), confidence penalizing (CP) as described by Pereyra et al. (2017), and the focal loss (FL) of Lin et al. (2020) to our label relaxation (LR) approach. To this end, we study the performances on neural networks for the task of image classification. Although the losses are completely general and not specifically tailored to any domain, this problem serves as a good representative and has been used in related studies in the past.

In addition to assessing the generalization accuracy in terms of the classification rate, we also measure the degree of calibration of the networks, i.e., the quality of the predicted class probabilities. To this end, we use the estimated expected calibration error (ECE) as done by Guo et al. (2017). This measure requires probabilities to be discretized through binning, and as suggested by Müller, Kornblith, and Hinton (2019), we fix the number of bins to 15. To compare label smoothing and our approach with explicit calibration methods, non-calibration and temperature scaling (Guo et al. 2017) serve as baselines.

Within our study, we consider MNIST (LeCun et al. 1998), Fashion-MNIST (Xiao, Rasul, and Vollgraf 2017), CIFAR-10 and CIFAR-100 (Krizhevsky and Hinton 2009) as image datasets. While MNIST and Fashion-MNIST both have 60k training and 10k test examples, CIFAR-10 and CIFAR-100 each consist of 50k training and 10k test instances. For the first two datasets, we train our models on a simple fully connected, ReLU activated neural network structure with two hidden layers consisting of 1024 neurons each. For the latter two datasets, we train the commonly used deep architectures VGG16 (Simonyan and Zisserman 2015), ResNet56 (V2) (He et al. 2016) and DenseNet-BC-100-12 (Huang et al. 2017). While we repeated every experiment for MNIST and Fashion-MNIST with 10 different seeds, we run each of the latter experiments 5 times. The runs were conducted on 20 Nvidia RTX 2080 Ti and 10 Nvidia GTX 1080 Ti GPUs.

For a fair comparison, all hyperparameters are fixed, except the parameter α in the case of label relaxation and smoothing loss, β as degree of confidence penalization in CP, and γ as being used to adjust the focal loss. For every combination of model and dataset, we empirically determined hyperparameters (such as the learning rate schedule and additional regularization) that work reasonably well for all losses. Since all losses are quite similar to each other, this was possible without favoring some of them while putting others at a disadvantage. To diminish regularization effects by additional means, we tried to exclude other techniques (such as extensive weight decay or Dropout (Srivastava et al. 2014)) as much as possible, thereby emphasizing the effect of the different loss functions while still achieving performances close to the originally published results.

To optimize the models, SGD with a Nesterov momentum of 0.9 has been used as optimizer. In all experiments, the batch size has been fixed to 64. Depending on the model, we set the initial learning rates to 0.01 (VGG), 0.05 (simple dense), and 0.1 (ResNet and DenseNet). For each model, we optimized the learning rate schedule for generalization performance by dividing the learning rate by a constant factor (ranging from 0.1 to $\sqrt{0.1}$). We trained for either 25 (MNIST), 50 (Fashion-MNIST), 200 (CIFAR-10), or 300 (CIFAR-100) epochs. Furthermore, we used data augmentation by randomly horizontally flipping and shifting the input images in width and height. We refer to the appendix for a more comprehensive overview of the fixed hyperparameters.

Since the parameters α for LR and LS, β for CP, and γ for FL are of critical importance, they have been optimized separately on a separate hold-out validation set consisting of $1/6$ of the original training data. In the first experiments, we optimize this parameter for the highest classification rate, whereas in the second evaluation, we focus on a low ECE. In both cases, we assessed values $\alpha \in \{0.01, 0.025, 0.05, 0.1, 0.2, 0.3, 0.4\}$, $\beta \in \{0.1, 0.3, 0.5, 1, 2, 4, 8\}$, and $\gamma \in \{0.1, 0.2, 0.5, 1, 2, 3.5, 5\}$ as suggested as reasonable parameters in the corresponding publications. The best model is then retrained on the original training data and evaluated on a separate test set. For each seed, the original training and test splits are merged and resampled to increase the variance of the experiments. This way, we achieve a better estimation of the generalization error. However, as a consequence, the presented results are not directly comparable to previously published results based on the original splits, although this special case is also covered in our experiments.

For temperature scaling, a separate hold-out validation set is used to optimize the parameter T among the values $T \in \{0.25, 0.5, 0.75, 1, 1.1, 1.2, \ldots, 2, 2.5, 3\}$. This parameter highly depends on the actually trained model and does not generalize well, i.e., a value performing well on an inner optimization run does not necessarily imply good calibration on the model finally trained. Therefore, the evaluation scenario is slightly different compared to the optimization of α: For the latter, as opposed to the case of temperature scaling, the hold-out validation set is included in the final training using the optimized parameters. This can be regarded as the price being paid for explicitly calibrating the model with temperature scaling, as opposed to the implicit calibration achieved by label smoothing and label relaxation. Here, we use 15% of the training data for calibration, which is almost comparable to the validation set used for optimizing α.

Loss	MNIST		Fashion-MNIST		Avg. Rank	
	Acc.	ECE	Acc.	ECE	Acc.	ECE
CE ($\alpha = 0$)	0.985 ± 0.002	0.010 ± 0.001	0.912 ± 0.003	0.129 ± 0.184	2	3.5
LS (α opt. for acc.)	$\mathbf{0.988} \pm 0.001$	0.106 ± 0.144	$\mathbf{0.915} \pm 0.002$	0.155 ± 0.128	**1**	5
CP (β opt. for acc.)	0.985 ± 0.002	0.012 ± 0.002	0.911 ± 0.004	0.075 ± 0.005	3	3.5
FL (γ opt. for acc.)	0.984 ± 0.002	0.009 ± 0.002	0.911 ± 0.002	0.062 ± 0.002	4.5	2
LR (α opt. for acc.)	0.985 ± 0.001	$\mathbf{0.007} \pm 0.002$	0.912 ± 0.003	$\mathbf{0.059} \pm 0.008$	2	**1**
CE ($\alpha = 0, T$ opt.)	0.983 ± 0.001	$\mathbf{0.003} \pm 0.001$	0.908 ± 0.004	0.030 ± 0.003	4	2.5
LS (α opt. for ECE)	$\mathbf{0.987} \pm 0.001$	0.014 ± 0.001	$\mathbf{0.915} \pm 0.003$	0.016 ± 0.002	**1**	3.5
CP (β opt. for ECE)	0.984 ± 0.001	0.011 ± 0.001	0.911 ± 0.003	0.072 ± 0.003	2.5	4
FL (γ opt. for ECE)	0.982 ± 0.001	0.004 ± 0.001	0.907 ± 0.003	$\mathbf{0.011} \pm 0.002$	5	**1.5**
LR (α opt. for ECE)	0.985 ± 0.002	$\mathbf{0.003} \pm 0.001$	0.911 ± 0.003	0.015 ± 0.003	2	**1.5**

Table 1: Results on MNIST and Fashion-MNIST using a simple 2-layer dense architecture. Bold entries indicate the best combination with regard to the corresponding metric per dataset and optimization scheme. The resulting ranks are averaged over both datasets for the respective metric.

Results

Table 1 shows the results of all assessed loss variants with regard to their classification performance and calibration error on MNIST and Fashion-MNIST. As can be seen, with a single exception, our label relaxation approach provides the lowest calibration error on both datasets regardless of the optimization target (accuracy or ECE). Although models trained with the focal loss deliver competitive calibration results, they generalize worse than LR optimized models. Label smoothing delivers the highest classification rate, while lacking calibration abilities. By still having a competitive classification rate compared to LS, our method offers a reasonable compromise between strong generalization (in terms of classification rate) and good calibration.

Since the accuracies on MNIST and Fashion-MNIST are already quite high, a more insightful evaluation is given by the experiments on CIFAR-10 and CIFAR-100, using multiple popular deep convolutional network architectures. Table 2 summarizes the results for both datasets and the different topologies. With few exceptions, LR minimizes the calibration error in terms of ECE among the assessed losses. At the same time, in accordance with the results presented before, it provides competitive classification rates. While FL-based models also yield relatively low calibration errors, they sometimes drop significantly in terms of classification performance (e.g., VGG16 and DenseNet-BC on CIFAR-100 optimized for ECE). Also, although temperature scaling uses separate data to explicitly optimize the temperature for a low calibration error, the implicit calibration of LR outperforms temperature scaling in most of the cases. Thus, relying on losses that implicitly calibrate models seems to be a reasonable strategy for model calibration.

To get a better overview of the presented results, Table 3 shows the resulting aggregated ranks for all datasets and models per metric and parameter optimization target (accuracy or ECE). For both optimization schemes, LR clearly dominates the other losses in terms of the calibration error. At the same time, the overall classification performance is reasonably close to the best loss, especially when applying accuracy-based hyperparameter optimization. As the results demonstrate, it balances both metrics and provides a compelling alternative to the other losses, particularly for applications in which the aim is to predict probabilities matching the underlying true probabilities of the classes.

Conclusion

We proposed label relaxation as an alternative to label smoothing, an established technique for preventing overfitting and over-confidence in classifier learning: Instead of replacing the original (degenerate) distribution associated with an observed class label by another, smoother yet still precise distribution, we relax the problem by letting the learner choose from a larger set of such distributions. This kind of "imprecisiation" of training data relieves the learner from the need to reproduce unrealistically definite observations, very much like label smoothing, but also allows it to predict probabilities in a flexible way. This flexibility appears to be important, not only for accurate classification, but even more so for producing less biased and better calibrated probability estimates.

These reflections are confirmed by an empirical study in image classification. Here, the calibration of deep convolutional neural network models could be improved without a loss in classification accuracy compared to label smoothing, penalizing confident output distributions and focal loss-based optimization. Label relaxation even outperforms explicit calibration methods like temperature scaling, which, due to requiring extra data for calibration, often pay with a drop in classification performance.

The idea of modeling targets in supervised learning in terms of imprecise probabilities, combined with the minimization of generalized losses penalizing deviations from the set of associated precise distributions, is very general and could be instantiated in various ways. Here, we considered the problem of classification and generalized the KL divergence. However, motivated by the promising empirical results, we also plan to look at other problems and other combinations of "data imprecisiation" and loss functions. Even-

Model	Loss	CIFAR-10		CIFAR-100		Avg. Rank	
		Acc.	ECE	Acc.	ECE	Acc.	ECE
VGG16	CE ($\alpha = 0$)	**0.930** ± 0.002	0.041 ± 0.001	0.708 ± 0.003	0.196 ± 0.003	**1.5**	3.5
	LS (α opt. for acc.)	0.929 ± 0.001	0.148 ± 0.119	**0.711** ± 0.003	0.149 ± 0.049	**1.5**	3.5
	CP (β opt. for acc.)	0.927 ± 0.001	0.059 ± 0.002	0.703 ± 0.003	0.228 ± 0.013	3	4.5
	FL (γ opt. for acc.)	0.921 ± 0.001	0.038 ± 0.004	0.700 ± 0.005	0.190 ± 0.009	5	2.5
	LR (α opt. for acc.)	0.927 ± 0.002	**0.033** ± 0.008	0.701 ± 0.006	**0.133** ± 0.069	3.5	**1**
	CE ($\alpha = 0, T$ opt.)	0.922 ± 0.001	**0.017** ± 0.003	0.689 ± 0.005	0.053 ± 0.003	3.5	2
	LS (α opt. for ECE)	**0.932** ± 0.002	0.028 ± 0.010	**0.711** ± 0.003	0.085 ± 0.005	**1**	4
	CP (β opt. for ECE)	0.922 ± 0.003	0.050 ± 0.003	0.700 ± 0.004	0.209 ± 0.004	3	5
	FL (γ opt. for ECE)	0.918 ± 0.003	0.027 ± 0.001	0.684 ± 0.007	0.048 ± 0.014	5	2.5
	LR (α opt. for ECE)	0.926 ± 0.001	0.022 ± 0.001	0.703 ± 0.006	**0.046** ± 0.005	2	**1.5**
ResNet56 (V2)	CE ($\alpha = 0$)	0.940 ± 0.002	0.041 ± 0.002	0.737 ± 0.003	0.126 ± 0.003	3	3
	LS (α opt. for acc.)	0.938 ± 0.002	0.132 ± 0.145	0.733 ± 0.004	0.110 ± 0.061	4.5	4
	CP (β opt. for acc.)	0.939 ± 0.003	0.046 ± 0.004	**0.738** ± 0.004	0.151 ± 0.007	2	4
	FL (γ opt. for acc.)	**0.941** ± 0.002	**0.036** ± 0.006	**0.738** ± 0.005	0.107 ± 0.017	**1**	**1.5**
	LR (α opt. for acc.)	0.938 ± 0.003	0.059 ± 0.090	**0.738** ± 0.003	**0.092** ± 0.030	2.5	2.5
	CE ($\alpha = 0, T$ opt.)	0.933 ± 0.002	0.030 ± 0.002	0.709 ± 0.005	0.041 ± 0.006	5	3.5
	LS (α opt. for ECE)	**0.940** ± 0.002	0.017 ± 0.002	0.730 ± 0.004	0.053 ± 0.003	2	3
	CP (β opt. for ECE)	**0.940** ± 0.002	0.044 ± 0.001	**0.741** ± 0.003	0.140 ± 0.003	**1**	5
	FL (γ opt. for ECE)	0.938 ± 0.002	0.017 ± 0.002	0.738 ± 0.005	0.024 ± 0.003	3	2
	LR (α opt. for ECE)	0.939 ± 0.002	**0.016** ± 0.002	0.729 ± 0.003	**0.017** ± 0.003	3.5	**1**
DenseNet-BC (100-12)	CE ($\alpha = 0$)	**0.929** ± 0.003	0.050 ± 0.002	**0.706** ± 0.005	0.229 ± 0.005	**1**	4
	LS (α opt. for acc.)	0.927 ± 0.004	0.046 ± 0.011	0.704 ± 0.008	**0.182** ± 0.063	4	**1.5**
	CP (β opt. for acc.)	**0.929** ± 0.002	0.056 ± 0.004	0.698 ± 0.011	0.252 ± 0.018	3	5
	FL (γ opt. for acc.)	0.928 ± 0.003	0.047 ± 0.004	0.703 ± 0.001	0.223 ± 0.001	3.5	3
	LR (α opt. for acc.)	0.928 ± 0.002	**0.039** ± 0.014	**0.706** ± 0.003	0.203 ± 0.023	2	**1.5**
	CE ($\alpha = 0, T$ opt.)	0.921 ± 0.003	**0.009** ± 0.004	0.687 ± 0.006	0.096 ± 0.006	4	**2**
	LS (α opt. for ECE)	**0.928** ± 0.003	0.020 ± 0.002	**0.704** ± 0.015	**0.077** ± 0.035	**1**	2.5
	CP (β opt. for ECE)	**0.928** ± 0.002	0.054 ± 0.002	**0.704** ± 0.003	0.237 ± 0.003	**1**	5
	FL (γ opt. for ECE)	0.915 ± 0.003	0.016 ± 0.001	0.681 ± 0.004	0.133 ± 0.004	5	3
	LR (α opt. for ECE)	0.922 ± 0.002	0.017 ± 0.003	0.703 ± 0.008	0.085 ± 0.006	3	2.5

Table 2: Results on CIFAR-10 and CIFAR-100 for the assessed model architectures. Here, bold entries indicate the best performances among the loss variants per dataset, model and optimization scheme. The ranks are averaged over both datasets as done before.

Loss	Acc. Opt.		ECE Opt.		Overall	
	Acc.	ECE	Acc.	ECE	Acc.	ECE
CE	**1.88**	3.5	4.13	2.5	3	3
LS	2.75	3.5	**1.25**	3.25	**2**	3.38
CP	2.75	4.25	1.88	4.75	2.31	4.5
FL	3.5	2.25	4.5	2.25	4	2.25
LR	2.5	**1.5**	2.63	**1.63**	2.56	**1.56**

Table 3: Average ranks of the losses with regard to the accuracy and ECE when a) optimizing the accuracy, b) optimizing the ECE and c) the overall ranking.

tually, a broader study of different instantiations should lead to a deeper understanding and general methodology of label relaxation.

Acknowledgments

This work was partially supported by the German Research Foundation (DFG) under Grant No. 3050231323. The authors gratefully acknowledge the funding of this project by computing time provided by the Paderborn Center for Parallel Computing (PC^2).

References

Bagherinezhad, H.; Horton, M.; Rastegari, M.; and Farhadi, A. 2018. Label Refinery: Improving ImageNet Classification through Label Progression. *CoRR* abs/1805.02641.

Cabannes, V.; Rudi, A.; and Bach, F. R. 2020. Structured Prediction with Partial Labelling through the Infimum Loss. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, Virtual Event, July 13-18, 2020*, volume 119, 1230–1239. PMLR.

Cireşan, D. C.; Meier, U.; Gambardella, L. M.; and Schmidhuber, J. 2010. Deep, big, simple neural nets for handwritten digit recognition. *Neural Computation* 22(12): 3207–3220.

Dubey, A.; Gupta, O.; Raskar, R.; and Naik, N. 2018. Maximum-Entropy Fine Grained Classification. In Bengio, S.; Wallach, H. M.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, 635–645.

Dubois, D.; and Prade, H. 2004. Possibility Theory, Probability Theory and Multiple-Valued Logics: A Clarification. *Annals of Mathematics and Artificial Intelligence* 32: 35–66.

Guo, C.; Pleiss, G.; Sun, Y.; and Weinberger, K. Q. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, August 6-11, 2017*, volume 70 of *Proceedings of Machine Learning Research*, 1321–1330. PMLR.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Identity mappings in deep residual networks. In *Proceedings of the 14th European Conference on Computer Vision, ECCV 2016, Amsterdam, The Netherlands, October 11-14, 2016, Part IV*, 630–645. Springer.

Hinton, G. E.; Vinyals, O.; and Dean, J. 2015. Distilling the Knowledge in a Neural Network. *CoRR* abs/1503.02531.

Huang, G.; Liu, Z.; van der Maaten, L.; and Weinberger, K. Q. 2017. Densely Connected Convolutional Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 2261–2269. IEEE Computer Society.

Hüllermeier, E.; and Cheng, W. 2015. Superset Learning Based on Generalized Loss Minimization. In Appice, A.; Rodrigues, P. P.; Costa, V. S.; Gama, J.; Jorge, A.; and Soares, C., eds., *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2015, Porto, Portugal, September 7-11, 2015, Proceedings, Part II*, volume 9285 of *Lecture Notes in Computer Science*, 260–275. Springer.

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. Technical report, University of Toronto, Toronto, Canada.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In Bartlett, P. L.; Pereira, F. C. N.; Burges, C. J. C.; Bottou, L.; and Weinberger, K. Q., eds., *Proceedings of the 26th Annual Conference on Neural Information Processing Systems, NIPS 2012, Lake Tahoe, NV, USA, December 3-6, 2012*, 1106–1114.

Kull, M.; Filho, T. S.; and Flach, P. 2017. Beta calibration: a well-founded and easily implemented improvement on logistic calibration for binary classifiers. In Singh, A.; and Zhu, J., eds., *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, Fort Lauderdale, FL, USA, April 20-22, 2017*, volume 54 of *Proceedings of Machine Learning Research*, 623–631. PMLR.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86(11): 2278–2324.

Li, W.; Dasarathy, G.; and Berisha, V. 2020. Regularization via Structural Label Smoothing. In Chiappa, S.; and Calandra, R., eds., *23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, Online [Palermo, Sicily, Italy], August 26-28, 2020*, volume 108 of *Proceedings of Machine Learning Research*, 1453–1463. PMLR.

Li, Y.; Yang, J.; Song, Y.; Cao, L.; Luo, J.; and Li, L. 2017. Learning from Noisy Labels with Distillation. In *Proceedings of the IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 1928–1936. IEEE Computer Society.

Lin, T.; Goyal, P.; Girshick, R. B.; He, K.; and Dollár, P. 2020. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* 42(2): 318–327.

Lukasik, M.; Bhojanapalli, S.; Menon, A. K.; and Kumar, S. 2020. Does Label Smoothing Mitigate Label Noise? In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, Virtual Event, July 13-18, 2020*, volume 119, 6448–6458. PMLR.

Müller, R.; Kornblith, S.; and Hinton, G. E. 2019. When Does Label Smoothing Help? In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS 2019, Vancouver, BC, Canada, December 8-14, 2019*, 4696–4705.

Naeini, M. P.; Cooper, G. F.; and Hauskrecht, M. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence, Austin, Texas, USA, January 25-30, 2015*, 2901–2907. AAAI Press.

Pereyra, G.; Tucker, G.; Chorowski, J.; Kaiser, L.; and Hinton, G. E. 2017. Regularizing Neural Networks by Penalizing Confident Output Distributions. *CoRR* abs/1701.06548.

Reed, S. E.; Lee, H.; Anguelov, D.; Szegedy, C.; Erhan, D.; and Rabinovich, A. 2015. Training Deep Neural Networks on Noisy Labels with Bootstrapping. In *Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*.

Simonyan, K.; and Zisserman, A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015*.

Srivastava, N.; Hinton, G. E.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* 15(1): 1929–1958.

Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the Inception Architecture for Computer Vision. In *Proceedings of the IEEE Conference on*

Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, 2818–2826. IEEE Computer Society.

van der Maaten, L.; Chen, M.; Tyree, S.; and Weinberger, K. Q. 2013. Learning with marginalized corrupted features. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, 410–418.

Vincent, P.; Larochelle, H.; Bengio, Y.; and Manzagol, P. 2008. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning, ICML 2008, Helsinki, Finland, June 5-9, 2008*, volume 307 of *ACM International Conference Proceeding Series*, 1096–1103. ACM.

Walley, P. 1991. *Statistical reasoning with imprecise probabilities*. Chapman & Hall.

Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-MNIST: A Novel Image Dataset for Benchmarking Machine Learning Algorithms. *CoRR* abs/1708.07747.

Xie, L.; Wang, J.; Wei, Z.; Wang, M.; and Tian, Q. 2016. DisturbLabel: Regularizing CNN on the Loss Layer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 4753–4762. IEEE Computer Society.

Yun, S.; Park, J.; Lee, K.; and Shin, J. 2020. Regularizing Class-Wise Predictions via Self-Knowledge Distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 13873–13882. IEEE.

Zadrozny, B.; and Elkan, C. 2002. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 23-26, 2002, Edmonton, Alberta, Canada*, 694–699. ACM.

Zhang, L.; Song, J.; Gao, A.; Chen, J.; Bao, C.; and Ma, K. 2019. Be Your Own Teacher: Improve the Performance of Convolutional Neural Networks via Self Distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, 3712–3721. IEEE.

5 Credal Self-Supervised Learning

Author Contribution Statement

Based on the previous work on label relaxation, the idea of integrating it into a recent semi-supervised learning framework originates from the author. Both authors contributed to the further development of this idea. The paper was initially written by the author, and subsequently revised by both authors. Furthermore, the implementation and experimentation was done by the author.

Supplementary Material

An appendix to the paper is provided in Appendix B. The code of the official implementation is provided at `https://github.com/julilien/CSSL` (Apache 2.0 license).

Credal Self-Supervised Learning

Julian Lienen
Department of Computer Science
Paderborn University
Paderborn 33098, Germany
julian.lienen@upb.de

Eyke Hüllermeier
Institute of Informatics
University of Munich (LMU)
Munich 80538, Germany
eyke@ifi.lmu.de

Abstract

Self-training is an effective approach to semi-supervised learning. The key idea is to let the learner itself iteratively generate "pseudo-supervision" for unlabeled instances based on its current hypothesis. In combination with consistency regularization, pseudo-labeling has shown promising performance in various domains, for example in computer vision. To account for the hypothetical nature of the pseudo-labels, these are commonly provided in the form of probability distributions. Still, one may argue that even a probability distribution represents an excessive level of informedness, as it suggests that the learner precisely knows the ground-truth conditional probabilities. In our approach, we therefore allow the learner to label instances in the form of credal sets, that is, sets of (candidate) probability distributions. Thanks to this increased expressiveness, the learner is able to represent uncertainty and a lack of knowledge in a more flexible and more faithful manner. To learn from weakly labeled data of that kind, we leverage methods that have recently been proposed in the realm of so-called superset learning. In an exhaustive empirical evaluation, we compare our methodology to state-of-the-art self-supervision approaches, showing competitive to superior performance especially in low-label scenarios incorporating a high degree of uncertainty.

1 Introduction

Recent progress and practical success in machine learning, especially in deep learning, is largely due to an increased availability of data. However, even if data collection is cheap in many domains, labeling the data so as to make it amenable to supervised learning algorithms might be costly and often comes with a significant effort. As a consequence, many data sets are only partly labeled, i.e., only a few instances are labeled while the majority is not. This is the key motivation for semi-supervised learning (SSL) methods [7], which seek to exploit both labeled and unlabeled data simultaneously.

As one simple yet effective methodology, so-called *self-training* [26], often also referred to as pseudo-labeling, has proven effective in leveraging unlabeled data to improve over training solely on labeled data. The key idea is to let the learner itself generate "pseudo-supervision" for unlabeled instances based on its own current hypothesis. In the case of probabilistic classifiers, such pseudo-targets are usually provided in the form of (perhaps degenerate) probability distributions. Obviously, since pseudo-labels are mere guesses and might be wrong, this comes with the danger of biasing the learning process, a problem commonly known as confirmation bias [46]. Therefore, self-training is nowadays typically combined with additional regularization means, such as consistency regularization [2, 40], or additional uncertainty-awareness [38].

Labeling an instance $\boldsymbol{x}$ with a probability distribution on the target space $\mathcal{Y}$ is certainly better than committing to a precise target value, for example a single class label in classification, as the latter would suggest a level of conviction that is not warranted. Still, one may argue that even a probability distribution represents an excessive level of informedness. In fact, it actually suggests that the learner

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

precisely knows the ground-truth conditional probability $p(y \mid \boldsymbol{x})$. In our approach, we therefore allow the learner to label instances in the form of *credal sets*, that is, sets of (candidate) probability distributions [27]. Thanks to this increased expressiveness, the learner is able to represent uncertainty and a lack of knowledge about the true label (distribution) in a more flexible and more faithful manner. For example, by assigning the biggest credal set consisting of all probability distributions, it is able to represent complete ignorance — a state of knowledge that is arguably less well represented by a uniform probability distribution, which could also be interpreted as full certainty about this distribution being the ground truth. Needless to say, this ability is crucial to avoid a confirmation bias and account for the heteroscedastic nature of uncertainty, which varies both spatially (i.e., among different regions in the instance space) and temporally: typically, the learner is less confident in early stages of the training process and becomes more confident toward the end.

Existing methods are well aware of such problems but handle them in a manner that is arguably ad-hoc. The simple yet effective SSL framework FixMatch [41], for instance, applies a thresholding technique to filter out presumably unreliable pseudo-labels, which often results in unnecessarily delayed optimization, as many instances are considered only lately in the training. Other approaches, such as MixUp [53], also apply mixing strategies to learn in a more cautious manner from pseudo-labels [1, 3, 4]. Moreover, as many of these approaches, including FixMatch, rely on the principle of entropy minimization [19] to separate classes well, the self-supervision is generated in the form of rather peaked or even degenerate distributions to learn from, which amplifies the problems of confirmation bias and over-confidence [33].

An important implication of credal pseudo-labeling is the need for extending the underlying learning algorithm, which must be able to learn from weak supervision of that kind. To this end, we leverage the principle of generalized risk minimization, which has recently been proposed in the realm of so-called superset learning [22]. This approach supports the idea of *data disambiguation*: The learner is free to (implicitly) choose any distribution inside a credal set that appears to be most plausible in light of the other data (and its own learning bias). Thus, an implicit trade-off between cautious learning and entropy minimization can be realized: Whenever it seems reasonable to produce an extreme distribution, the learner is free but not urged to do so. Effectively, this not only reduces the risk of a potential confirmation bias due to misleading or over-confident pseudo-labels, it also allows for incorporating all unlabeled instances in the learning process from the beginning without any confidence thresholding, leading to a fast and effective semi-supervised learning method.

To prove the effectiveness of this novel type of pseudo-labeling, we proceed from FixMatch as an effective state-of-the-art SSL framework and replace conventional probabilistic pseudo-labeling by a credal target set modeling. In an exhaustive empirical evaluation, we study the effects of this change compared to both hard and soft probabilistic target modeling, as well as measuring the resulting network calibration of induced models to reflect biases. Our experiments not only show competitive to superior generalization performance, but also better calibrated models while cutting the time to train the models drastically.

2 Related Work

In semi-supervised learning, the goal is to leverage the potential of unlabeled in addition to labeled data to improve learning and generalization. As it constitutes a broad research field with a plethora of approaches, we will focus here on classification methods as these are most closely related to our approach. We refer to [7] and [48] for more comprehensive overviews.

As one of the earliest ideas to incorporate unlabeled data in conventional supervised learning, *self-training* has shown remarkably effective in various domains, including natural language processing [12] and computer vision [11, 17, 39]. The technique is quite versatile and can be applied for different learning methods, ranging from support vector machines [30] to decision trees [45] and neural networks [35]. It can be considered as *the* basic training pattern in distillation, such as self-distillation from a model to be trained itself [23] or within student-teacher settings [36, 51]. Recently, this technique also lifted completely unsupervised learning in computer vision to a new level [6, 20].

As a common companion of self-training for classification, especially in computer vision, *consistency regularization* is employed to ensure similar model predictions when facing multiple perturbed versions of the same input [2, 37, 40], resulting in noise-robustness as similarly achieved by other (stochastic) ensembling methods such as Dropout [42]. Strong augmentation techniques used in this regard, e.g., CTAugment [3] or RandAugment [9], allow one to learn from instances outside of the (hitherto labeled) data distribution and, thus, lead to more accurate models [10]. For semi-supervised learning in image classification, the combination of consistency regularization with pseudo-labeling is widely adopted and has proven to be a simple yet effective strategy [1, 4, 25, 37, 41, 50, 55].

As pseudo-labeling comprises the risk of biasing the model by wrong predictions, especially when confidence is low, uncertainty awareness has been explicitly considered in the generic self-supervision framework UPS to construct more reliable targets [38]. Within their approach, the model uncertainty is estimated by common Bayesian sampling techniques, such as MC-Dropout [15] or DropBlock [16], which is then used to sort out unlabeled instances for which the model provides uncertain predictions. Related to this, the selection of pseudo-labels based on the model certainty has also been used in specific domains, such as text classification [32] or semantic segmentation [54].

2.1 FixMatch

As already mentioned, FixMatch [41] combines recent advances in consistency regularization and pseudo-labeling into a simple yet effective state-of-the-art SSL approach. It will serve as a basis for our new SSL method, as it provides a generic framework for fair comparisons between conventional and our credal pseudo-labeling.

In each training iteration, FixMatch considers a batch of B labeled instances $\mathcal{B}_l = \{(\boldsymbol{x}_i, p_i)\}_{i=1}^{B} \subset \mathcal{X} \times \mathbb{P}(\mathcal{Y})$ and μB unlabeled instances $\mathcal{B}_u = \{\boldsymbol{x}_i\}_{i=1}^{\mu B} \subset \mathcal{X}$, where $\mathcal{X}$ denotes the input feature space, $\mathcal{Y}$ the set of possible classes, $\mathbb{P}(\mathcal{Y})$ the set of probability distributions over $\mathcal{Y}$, and $\mu \geq 1$ the multiplicity of unlabeled over labeled instances in each batch. Here, the probabilistic targets in $\mathcal{B}_l$ are given as degenerate "one-hot" distributions. When aiming to induce probabilistic classifiers of the form $\hat{p} : \mathcal{X} \longrightarrow \mathbb{P}(\mathcal{Y})$, FixMatch distinguishes between two forms of augmentation: While $\mathcal{A}_s : \mathcal{X} \longrightarrow \mathcal{X}$ describes "strong" augmentations that perturbs the image in a drastic manner by combining multiple operations, simple flip-and-shift transformations are captured by "weak" augmentations $\mathcal{A}_w : \mathcal{X} \longrightarrow \mathcal{X}$.

As an iteration-wise loss to determine gradients, a combination of the labeled loss $\mathcal{L}_l$ and unlabeled loss $\mathcal{L}_u$ is calculated. For the former, the labeled input instances from $\mathcal{B}_l$ are weakly augmented and used in a conventional cross-entropy loss $H : \mathbb{P}(\mathcal{Y})^2 \longrightarrow \mathbb{R}$. For the latter, the model prediction $q := \hat{p}(\mathcal{A}_w(\boldsymbol{x}))$ on a weakly-augmented version of each unlabeled instance $\boldsymbol{x}$ is used to construct a (hard) pseudo-label $\tilde{q} \in \mathbb{P}(\mathcal{Y})$ when meeting a predefined confidence threshold τ. While $\tilde{q}$ is in FixMatch a degenerate probability distribution by default, one could also inject soft probabilities, which, however, turned out to be less effective (cf. [41]). The pseudo-label is then compared to a strongly-augmented version of the same input image. Hence, the unlabeled loss $\mathcal{L}_u$ is given by

$$\mathcal{L}_u := \frac{1}{\mu B} \sum_{\boldsymbol{x} \in \mathcal{B}_u} \mathbb{I}_{\max q \geq \tau} H(\tilde{q}, \hat{p}(\mathcal{A}_s(\boldsymbol{x}))) \enspace .$$

3 Credal Self-Supervised Learning

In this section, we introduce our credal self-supervised learning (CSSL) framework for the case of classification, assuming the target to be categorical with values in $\mathcal{Y} = \{y_1, \ldots, y_K\}$. Before presenting more technical details, we start with a motivation for the credal (set-valued) modeling of target values and a sketch of the basic idea of our approach.

3.1 Motivation and Basic Idea

In supervised learning, we generally assume the dependency between instances $\boldsymbol{x} \in \mathcal{X}$ and associated observations (outcomes) to be of stochastic nature. More specifically, one typically assumes a "ground-truth" in the form of a conditional probability distribution $p^*(\cdot \,|\, \boldsymbol{x}) \in \mathbb{P}(\mathcal{Y})$. Thus, for every $y \in \mathcal{Y}$, $p^*(y \,|\, \boldsymbol{x})$ is the probability to observe y as a value for the target variable in the context $\boldsymbol{x}$. Ideally, the distribution $p^*(\cdot \,|\, \boldsymbol{x})$ would be provided as training information to the learner, along with every training instance $\boldsymbol{x}$. In practice, however, supervision comes in the form of concrete values of the target, i.e., a realization $y \in \mathcal{Y}$ of the random variable $Y \sim p^*(\cdot \,|\, \boldsymbol{x})$, and the corresponding degenerate distribution p_y assigning probability mass 1 to y (i.e., $p_y(y \,|\, \boldsymbol{x}) = 1$ and $p_y(y' \,|\, \boldsymbol{x}) = 0$ for $y' \neq y$) is taken as a surrogate for p^*.

Obviously, turning the true distribution $p^* = p^*(\cdot \mid \boldsymbol{x})$, subsequently also called a "soft label", into a more extreme distribution p_y (i.e., a single value y), referred to as "hard label", may cause undesirable effects. In fact, it suggests a level of determinism that is not warranted and tempts the learner to over-confident predictions that are poorly calibrated, especially when training with losses such as cross-entropy [33]. So-called label smoothing [44] seeks to avoid such effects by replacing hard labels p_y with (hypothetical) soft labels $\hat{p}$ that are "close" to p_y.

Coming back to semi-supervised learning, training of the learner and self-supervision can be characterized for existing methods such as FixMatch as follows:

- The learner is trained on hard labels, which are given for the labeled instances $\boldsymbol{x}_l$ and constructed by the learner itself for unlabeled instances $\boldsymbol{x}_u$.
- Construction of (pseudo-)labels is done in two steps: First, the true soft label $p^* = p^*(\cdot \mid \boldsymbol{x}_u)$ is predicted by a distribution $\hat{p} = \hat{p}(\cdot \mid \boldsymbol{x}_u)$, and the latter is turned into a hard label $\hat{p}_y$ afterward, provided $\hat{p}$ suggests a sufficient level of certainty (support for y).

This approach can be challenged for several reasons. First, training probabilistic predictors on hard labels comes with the disadvantages mentioned above and tends to bias the learner. Second, pseudo-labels $\hat{p}_y$ constructed by the learner tend to be poor approximations of the ground truth p^*. In fact, there will always be a discrepancy between p^* and its prediction $\hat{p}$, and this discrepancy is further increased by replacing $\hat{p}$ with $\hat{p}_y$. Third, leaving some of the instances — those for which the prediction is not reliable enough — completely unlabeled causes a loss of information. Roughly speaking, while a part of the training data is overly precise[1], another part remains unnecessarily imprecise and hence unused. This may slow down the training process and cause other undesirable problems such as "path-dependency" (training is influenced by the order of the unlabeled instances).

To avoid these disadvantages, we propose to use soft instead of hard labels as training information for the learner. More specifically, to account for possible uncertainty about a true soft label p^*, we model information about the target in the form of a set $Q \subseteq \mathbb{P}(\mathcal{Y})$ of distributions, in the literature on imprecise probability also called a *credal set* [27]. Such a "credal label" is supposed to cover the ground truth, i.e., $p^* \in Q$, very much like a confidence interval in statistics is supposed to cover (with high probability) some ground-truth parameter to be estimated.

This approach is appealing, as it enables the learner to model its belief about the ground-truth p^* in a cautious and faithful manner: Widening a credal label Q may weaken the training information but maintains or even increases its validity. Moreover, credal labeling elegantly allows for covering the original training information as special cases: A hard label y provided for a labeled instance $\boldsymbol{x}_l$ corresponds to a singleton set $Q_y = \{p_y\}$, and the lack of any information in the case of an unlabeled instance $\boldsymbol{x}_u$ is properly captured by taking $Q = \mathbb{P}(\mathcal{Y})$. Starting with this information, the idea is to modify it in two directions:

- Imprecisiation: To avoid possible disadvantages of hard labels Q_y, these can be made less precise through label relaxation [29], which is an extension of the aforementioned label smoothing. Technically, it means that Q_y is replaced by a credal set Q containing, in addition to p_y itself, distributions p close to p_y.
- Precisiation: The non-informative and maximally imprecise labels $Q = \mathbb{P}(\mathcal{Y})$ for unlabeled instances $\boldsymbol{x}_u$ are successively (iteration by iteration) "shrunken" and made more precise. This is done by replacing a set Q with a smaller subset $Q' \subset Q$, provided the exclusion of certain candidate distributions $p \in Q$ is sufficiently supported by the learner.

3.2 Credal Labeling

Credal sets are commonly assumed to be convex, i.e., $p, q \in Q$ implies $\lambda p + (1 - \lambda) q \in Q$ for all distributions p, q and $\lambda \in (0, 1)$. In our context, arbitrary (convex) credal sets $Q \subseteq \mathbb{P}(\mathcal{Y})$ could in principle be used for the purpose of labeling instances. Yet, to facilitate modeling, we restrict ourselves to credal sets induced by so-called *possibility distributions* [13].

A possibility or plausibility measure Π is a set-function $2^{\mathcal{Y}} \longrightarrow [0, 1]$ that assigns a degree of plausibility $\Pi(Y)$ to every subset (event) $Y \subseteq \mathcal{Y}$. Such a measure is induced by a possibility

[1] To some extent, this can be alleviated through measures such as instance weighing, i.e., by attaching a weight to a pseudo-labeled instances [38].

distribution $\pi : \mathcal{Y} \longrightarrow [0,1]$ via $\Pi(Y) = \max_{y \in Y} \pi(y)$ for all $Y \subseteq \mathcal{Y}$. Interpreting degrees of plausibility as upper probabilities, the set of (candidate) probability distributions p (resp. measures P) in accordance with π resp. Π is given by those for which $P(Y) \leq \Pi(Y)$ for all events Y:

$$Q_\pi = \Big\{ p \in \mathbb{P}(\mathcal{Y}) \,|\, \forall\, Y \subseteq \mathcal{Y} : P(Y) = \sum_{y \in Y} p(y) \leq \max_{y \in Y} \pi(y) = \Pi(Y) \Big\} \ . \tag{1}$$

Roughly speaking, for each $y \in \mathcal{Y}$, the possibility $\pi(y)$ determines an upper bound for $p^*(y)$, i.e., the highest probability that is deemed plausible for y. Note that, to guarantee $Q_\pi \neq \emptyset$, possibility distributions must be normalized in the sense that $\max_{y \in \mathcal{Y}} \pi(y) = 1$. In other words, there must be at least one outcome $y \in \mathcal{Y}$ that is deemed completely plausible.

In our context, this outcome is naturally taken as the label $y = \operatorname{argmax}_{y' \in \mathcal{Y}} \hat{p}(y')$ with the highest predicted probability. A simple way of modeling then consists of controlling the degree of imprecision (ignorance of the learner) by a single parameter $\alpha \in [0,1]$, considering credal sets of the form

$$Q_y^\alpha = \big\{ p \in \mathbb{P}(\mathcal{Y}) \,|\, p(y) \geq 1 - \alpha \big\} \ . \tag{2}$$

Thus, Q_y^α consists of all distributions p that allocate a probability mass of at least $1 - \alpha$ to y and hence at most α to the other labels $\mathcal{Y} \setminus \{y\}$. As important special cases we obtain $Q_y^0 = \{p_y\}$, i.e., the degenerate distribution that assigns probability 1 to the label y, and $Q_y^1 = \mathbb{P}(\mathcal{Y})$ modeling complete ignorance about the ground truth p^*.

Needless to say, more sophisticated credal sets could be constructed on the basis of a distribution $\hat{p}$, for example leveraging the concept of probability-possibility transformations [14]. Yet, to keep the modeling as simple as possible, we restrict ourselves to sets of the form (2) in this work.

3.3 Learning from Credal Labels

Our approach requires the learner to be able to learn from credal instead of probabilistic or hard labels. To this end, we refer to the generic approach to so-called superset learning as proposed in [22]. Essentially, this approach is based on minimizing a generalization $\mathcal{L}^*$ of the original (probabilistic) loss $\mathcal{L} : \mathbb{P}(\mathcal{Y})^2 \longrightarrow \mathbb{R}$. More specifically, the so-called *optimistic superset loss* [22], also known as *infimum loss* [5], compares credal sets with probabilistic predictions as follows:

$$\mathcal{L}^*(Q, \hat{p}) = \min_{p \in Q} \mathcal{L}(p, \hat{p}) \tag{3}$$

For the specific case where credal sets are of the form (2) and the loss $\mathcal{L}$ is the Kullback-Leibler divergence D_{KL}, (3) simplifies to

$$\mathcal{L}^*(Q_y^\alpha, \hat{p}) = \left\{ \begin{array}{cl} 0 & \text{if } \hat{p} \in Q_y^\alpha \\ D_{KL}(p^r || \hat{p}) & \text{otherwise} \end{array} \right. , \tag{4}$$

where

$$p^r(y') = \left\{ \begin{array}{cl} 1 - \alpha & \text{if } y' = y \\ \alpha \cdot \frac{\hat{p}(y')}{\sum_{y'' \neq y} \hat{p}(y'')} & \text{otherwise} \end{array} \right. \tag{5}$$

is the projection of the prediction $\hat{p}$ onto the boundary of Q. This loss has been proven to be convex, making its optimization practically feasible [29].

The loss (3) is an optimistic generalization of the original loss in the sense that it corresponds to the loss $\mathcal{L}(p, \hat{p})$ for the most favorable instantiation $p \in Q$. This optimism is motivated by the idea of *data disambiguation* [22] and can be justified theoretically [5]. Roughly speaking, minimizing the sum of generalized losses over all training examples $(\boldsymbol{x}_i, Q_i)$ comes down to (implicitly) choosing a precise probabilistic target inside every credal set, i.e., replacing $(\boldsymbol{x}_i, Q_i)$ by $(\boldsymbol{x}_i, p_i)$ with $p_i \in Q_i$, in such a way that the original loss (empirical risk) can be made as small as possible.

3.4 Credal Self-Supervised Learning Framework

Our idea of credal self-supervised learning (short CSSL) offers a rather generic framework for designing self-supervised learning methods. In this paper, we focus on image classification as a concrete and practically relevant application, and combine CSSL with the consistency regularization framework provided by FixMatch (cf. Fig. 1).

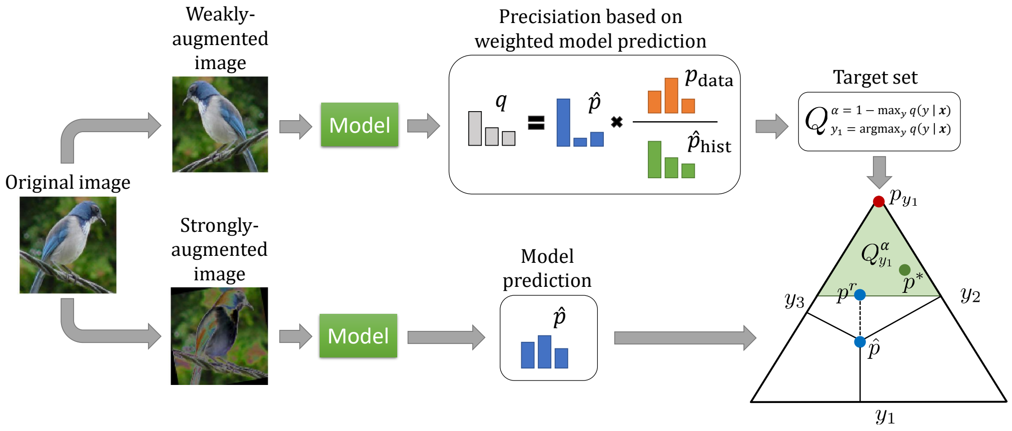


Figure 1: Schematic overview of the learning framework for unlabeled instances following [41]: Given classes $\mathcal{Y} = \{y_1, y_2, y_3\}$, a credal target set Q is generated from the prediction on a weakly-augmented version of the input image. As illustrated in the barycentric coordinate system on the right bottom, Q (red shaded region) covers the ground-truth distribution p^* (green point). The degenerate distribution p_{y_1} assigning probability 1 to y_1 corresponds to the red point on the top. To calculate the final loss, the model prediction $\hat{p}$ on a strongly-augmented version of the original image is compared to Q, whereby its projection onto Q is depicted by p^r.

To describe the algorithm, we again assume batches of labeled $\mathcal{B}_l$ and unlabeled instances $\mathcal{B}_u$ (cf. Section 2.1). For the former, we measure the labeled loss $\mathcal{L}_l$ in terms of the cross-entropy H between the observed target distributions of the (labeled) training instances and the current model predictions $\hat{p}$ on weakly-augmented features. At this point, one could apply the idea of imprecisiation through label relaxation as motivated in Section 3.1. However, as this work focuses on the self-supervision part, i.e., on $\mathcal{B}_u$ rather than $\mathcal{B}_l$, we stick to hard labels. This also facilitates the interpretation of experimental results later on, as it avoids the mixing of different effects.

For the unlabeled instances $\boldsymbol{x}_i \in \mathcal{B}_u$, we follow the consistency regularization idea of FixMatch and use the predictions on weakly augmented versions of the unlabeled instances as a reference for the target set construction. More precisely, we take the class $y_i = \operatorname{argmax}_{y \in \mathcal{Y}} \hat{p}_i(y)$ of the current model prediction $\hat{p}_i = \hat{p}(\mathcal{A}_w(\boldsymbol{x}_i)))$ as a reference to construct instance-wise targets $Q^{\alpha_i}_{y_i}$ as specified in (2).

A rather straightforward approach to determining the uncertainty level α_i is to set $\alpha_i = 1 - \hat{p}_i(y_i)$. Yet, motivated by the idea of distribution alignment as proposed in [3], we suggest to weight the predictions $\hat{p}_i$ by the proportion of the class prior $\tilde{p} \in \mathbb{P}(\mathcal{Y})$ and a moving average of the last model predictions $\bar{p} \in \mathbb{P}(\mathcal{Y})$, so that $\alpha_i = 1 - q_i(y) / \sum_{y' \in \mathcal{Y}} q_i(y')$ with the weighted (pseudo-)probability scores

$$q_i(y) = \hat{p}_i(y') \times \frac{\tilde{p}(y')}{\bar{p}(y')} \quad . \tag{6}$$

According to this way of modeling pseudo-labels in terms of credal sets, the size (imprecision) of a set is in direct correspondence with the confidence of the learner. Therefore, this approach leads to some sort of (implicit) disambiguation: With increasing confidence, the target sets are becoming more precise, successively fostering entropy minimization without imposing overly constrained targets. Also, as mentioned before, this approach allows for using all instances for training from the very beginning, without losing any of them due to confidence thresholding. The weighting mechanism based on the class prior $\tilde{p}$ and the prediction history $\bar{p}$ (second factor on the right-hand side of (6)) accounts for the consistency and hence the confidence in a particular class prediction. If $\tilde{p}(y') \gg \bar{p}(y')$, the label y' is under-represented in the past predictions, suggesting that its true likelihood might be higher than predicted by the learner, and vice versa in the case where $\tilde{p}(y') \ll \bar{p}(y')$. As this procedure is simple and computationally efficient, it facilitates the applicability compared to computationally demanding uncertainty methods such as MC-Dropout.

The proposed target sets are then used within the unlabeled loss $\mathcal{L}_u$ according to (4), which is used in addition to the labeled loss $\mathcal{L}_l$. Hence, the final loss is given by

$$\mathcal{L} = \underbrace{\frac{1}{|\mathcal{B}_l|} \sum_{(\boldsymbol{x}_i, p_i) \in \mathcal{B}_l} H(p_i, \hat{p}_i)}_{\mathcal{L}_l} + \lambda_u \underbrace{\frac{1}{|\mathcal{B}_u|} \sum_{\boldsymbol{x}_i \in \mathcal{B}_u} \mathcal{L}^*(Q_{y_i}^{\alpha_i}, \hat{p}_i)}_{\mathcal{L}_u} \quad . \tag{7}$$

The pseudo-code of the complete algorithm can be found in the appendix.

We conclude this section with a few remarks on implementation details. Since we are building upon FixMatch, we keep the same augmentation policy as suggested by the authors. Thus, we employ CTAugment by default. We refer to the appendix for further ablation studies, including RandAugment as augmentation policy. Moreover, we consider the same optimization algorithm as used before, namely SGD with Nesterov momentum, for which we use cosine annealing as learning rate schedule [31]. Similar to FixMatch, we set the learning rate to $\eta \cos \frac{7\pi k}{16K}$, where η is the initial learning rate, k the current training step and K the total number of steps (2^{20} by default). As we are keeping the algorithmic framework the same and do not require any form of confidence thresholding, we can reduce the number of parameters compared to FixMatch, which further facilitates the use of this approach. We also use an exponential moving average of model parameters for our final model, which comes with appealing ensembling effects that typically improve model robustness and has been considered by various recent approaches for un- or semi-supervised learning [6, 41].

4 Experiments

To compare our idea of credal pseudo-labeling, we conduct an exhaustive empirical evaluation with common image classification benchmarks. More precisely, we follow the semi-supervised learning evaluation setup as described in [41] and perform experiments on CIFAR-10/-100 [24], SVHN [34], and STL-10 [8] with varying fractions of labeled instances sampled from the original data sets, also considering label-scarce settings with only a few labels per class. For CIFAR-10, SVHN, and STL-10, we train a Wide ResNet-28-2 [52] with 1.49 M parameters, while we consider Wide ResNet-28-8 (23.4 M parameters) models for the experiments on CIFAR-100. To guarantee a fair comparison to existing methods related to FixMatch, we keep the hyperparameters the same as in the original experiments. We refer to the appendix for a more comprehensive overview of the experimental details. We repeat each run 5 times with different seeds for a higher significance and average the results for the model weights of the last 20 epochs as done in [41].

As baselines, we report the results for Mean Teacher [46], MixMatch [4], UDA [50], ReMixMatch [3], and EnAET [49] as state-of-the art semi-supervised learning methods. Moreover, as we directly compete against the probabilistic hard-labeling employed in FixMatch, we compare our approach to this (with CTAugment) and related methods, namely AlphaMatch [18], CoMatch [28], and ReRankMatch [47]. Since these approaches extend the basic framework of FixMatch by additional means (e.g., loss augmentations), the direct comparison with FixMatch is maximally fair under these conditions, avoiding side-effects as much as possible. In addition, we show the results of the student-teacher method Meta Pseudo Labels (Meta PL) [36].

4.1 Generalization Performance

In the first experiments, we train the aforementioned models on the four benchmark data sets for different numbers of labeled instances to measure the generalization performance of the induced models. The results are provided in Table 1.

As can be seen, CSSL is especially competitive when the number of labels is small, showing that the implicit uncertainty awareness of set-based target modeling becomes effective. But CSSL is also able to provide compelling performance in the case of relatively many labeled instances, although not substantially improving over conventional hard pseudo-labeling. For instance, it is approximately on par with state-of-the-art performance on CIFAR-100 and SVHN. Focusing on the comparison to FixMatch, credal self-supervision improves the performance in almost all cases over hard pseudo-labeling with confidence thresholding, providing further evidence for the adequacy of our method.

Table 1: Averaged misclassification rates for 5 different seeds using varying numbers of labeled instances (**bold** font indicates the best performing method and those within two standard deviations per data set and label number). Approaches using different models, so that the comparison may not be entirely fair, are marked with $*$. We also show the results for STL-10 as reported in [18], using smaller models due to computational resource limitations, which we mark by †.

	CIFAR-10			CIFAR-100			SVHN			STL-10†
	40 lab.	250 lab.	4000 lab.	400 lab.	2500 lab.	10000 lab.	40 lab.	250 lab.	1000 lab.	1000 lab.
Mean Teacher	-	32.32 ±2.30	9.19 ±0.19	-	53.91 ±0.57	35.83 ±0.24	-	3.57 ±0.11	3.42 ±0.07	-
MixMatch	47.54 ±11.50	11.05 ±0.86	6.42 ±0.10	67.61 ±1.32	39.94 ±0.37	28.31 ±0.33	42.55 ±14.53	3.98 ±0.23	3.50 ±0.28	14.84 ±1.24
UDA	29.05 ±5.93	8.82 ±1.08	4.88 ±0.18	59.28 ±0.88	33.13 ±0.22	24.50 ±0.25	52.63 ±20.51	5.69 ±2.76	2.46 ±0.24	13.43 ±1.06
ReMixMatch	19.10 ±9.64	**5.44** ±0.05	4.72 ±0.13	44.28 ±2.06	27.43 ±0.31	**23.03** ±0.56	**3.34** ±0.20	2.92 ±0.48	2.65 ±0.08	11.58 ±0.78
EnAET	-	7.6 ±0.34	5.35 ±0.48	-	-	-	-	3.21 ±0.21	2.92	-
AlphaMatch	8.65 ±3.38	**4.97** ±0.29	-	**38.74** ±0.32	**25.02** ±0.27	-	**2.97** ±0.26	2.44 ±0.32	-	**9.64** ±0.75
CoMatch	**6.91** ±1.39	**4.91** ±0.33	-	-	-	-	-	-	-	-
ReRankMatch	18.25 ±9.44	6.02 ±1.31	4.40 ±0.06	69.62 ±1.33	31.75 ±0.33	**22.32** ±0.65	20.25 ±4.43	2.44 ±0.07	**2.19** ±0.09	-
Meta PL*	-	-	**3.89** ±0.07	-	-	-	-	-	**1.99** ±0.07	-
FixMatch (CTA)	11.39 ±3.35	**5.07** ±0.33	4.31 ±0.15	49.95 ±3.01	28.64 ±0.24	**23.18** ±0.11	7.65 ±7.65	2.64 ±0.64	2.28 ±0.19	**10.72** ±0.63
CSSL (CTA)	**6.50** ±0.90	**5.48** ±0.49	4.43 ±0.10	43.43 ±1.39	28.39 ±1.09	**23.25** ±0.28	3.67 ±2.36	**2.18** ±0.12	**1.99** ±0.13	**10.54** ±0.71

Table 2: Averaged misclassification rates and expected calibration errors (ECE) using 15 bins for 5 different seeds.

	CIFAR-10				SVHN			
	40 lab.		4000 lab.		40 lab.		1000 lab.	
	Err.	ECE	Err.	ECE	Err.	ECE	Err.	ECE
FixMatch	11.39 ±3.35	0.087 ±0.051	**4.31** ±0.15	0.030 ±0.002	**7.65** ±7.65	**0.040** ±0.044	2.28 ±0.19	0.010 ±0.002
FixMatch (DA)	**7.73** ±1.92	0.048 ±0.012	4.64 ±0.10	0.027 ±0.001	**5.21** ±2.85	**0.031** ±0.020	**2.04** ±0.38	0.010 ±0.001
LSMatch	**8.37** ±1.63	**0.038** ±0.012	5.60 ±1.32	**0.024** ±0.007	**3.82** ±1.46	0.086 ±0.046	2.13 ±0.11	0.018 ±0.011
CSSL	**6.50** ±0.90	**0.032** ±0.005	**4.43** ±0.10	**0.023** ±0.001	**3.67** ±2.36	**0.022** ±0.029	**1.99** ±0.13	**0.007** ±0.001

4.2 Network Calibration

In a second study, we evaluate FixMatch-based models in terms of network calibration, i.e., the quality of the predicted class probabilities. For this purpose, we calculate the expected calibration error (ECE) as done in [21] using discretized probabilities into 15 bins. The calibration errors provide insight into the bias of the models induced by the different learning methods.

Besides the "raw" hard-labeling, we also consider the distribution alignment (DA)-variant of FixMatch (as described in [41]), as well as an adaptive variant using label smoothing [44], which we dub *LSMatch* (see appendix for an algorithmic description). For label smoothing, we use a uniform distribution policy and calculate the distribution mass parameter α in an adaptive manner as we do for CSSL (cf. Section 3.4). As a result, LSMatch can be regarded as the natural counterpart of our approach for a more cautious learning using less extreme targets. We refer to [29] for a more thorough analysis of the differences between smoothed and credal labels. Since both methods realize an implicit calibration [29, 33], we omit explicit calibration methods that require additional data. Besides, we experiment with an uncertainty-filtering variant of FixMatch following UPS [38], for which we provide results in the supplement.

In accordance with the studies in [29], the results provided in Table 2 show improved calibration compared to classical probabilistic modeling. As these effects are achieved without requiring an additional calibration data split, it provides an appealing method to induce well calibrated and generalizing models. Nevertheless, the margin to the other baselines gets smaller with an increasing number of labeled instances, which is plausible as it implies an increased level of certainty.

4.3 Efficiency

In addition to accuracy, a major concern of learning algorithms is run-time efficiency. In this regard, we already noted that thresholding mechanisms may severely delay the incorporation of unlabeled instances in the learning process. For example, for a data set with 2^{26} images, FixMatch requires up to 2^{20} updates to provide the competitive results reported. This not only excludes potential users without access to computational resources meeting these demands, it also consumes a substantial amount of energy [43].

Table 3: Averaged misclassification rates after $1/8$ (CIFAR-10) and $1/32$ (SVHN) of the original iterations used for the results in Tab. 1 (**bold** font indicates the single best performing method).

	CIFAR-10		SVHN	
	40 lab.	4000 lab.	40 lab.	1000 lab.
FixMatch ($\tau = 0.0$)	18.50 ±2.92	6.88 ±0.11	13.82 ±13.57	**2.73** ±0.04
FixMatch ($\tau = 0.8$)	11.99 ±2.32	7.08 ±0.13	3.52 ±0.44	2.85 ±0.08
FixMatch ($\tau = 0.95$)	14.73 ±3.29	8.26 ±0.09	5.85 ±5.10	3.03 ±0.07
LSMatch	11.60 ±2.68	7.24 ±0.21	7.04 ±3.29	2.76 ±0.05
CSSL	**10.04** ±3.32	**6.78** ±0.94	**3.50** ±0.49	2.84 ±0.06

As described before, CSSL allows for incorporating all instances from the very beginning. To measure the implied effects, i.e., a faster convergence, apart from using more effective optimizers, we train the models from the former experiment on CIFAR-10 and SVHN for only an eighth and thirty-second of the original number of epochs, respectively. As we would grant CSSL and LSMatch an unfair advantage compared to confidence thresholding in FixMatch, we experiment with different thresholds $\tau \in \{0, 0.8, 0.95\}$ for FixMatch. The (averaged) learning curves are provided in the supplement.

As shown by the results in Table 3, CSSL achieves the best performance in the label-scarce cases, which confirms the adequacy of incorporating all instances in a cautious manner from the very beginning. Likewise, LSMatch shows competitive performance following the same intuition. In contrast, FixMatch with a high confidence threshold shows slow convergence, which can be mitigated by lowering the thresholding. However, FixMatch with $\tau = 0.0$ provides unsatisfying performance as it drastically increases the risk of confirmation biases; $\tau = 0.8$ seems to be a reasonable trade-off between convergence speed and validity of the model predictions. Nevertheless, when providing higher numbers of labels, the performance gain in incorporating more instances from early on is fairly limited, making the concern of efficiency arguably less important in such cases.

5 Conclusion

Existing (probabilistic) methods for self-supervision typically commit to single probability distributions as pseudo-labels, which, as we argued, represents the uncertainty in such labels only insufficiently and comes with the risk of incorporating an undesirable bias. Therefore, we suggest to allow the learner to use credal sets, i.e., sets of (candidate) distributions, as pseudo-labels. In this way, a more faithful representation of the learner's (lack of) knowledge about the underlying ground truth distribution can be achieved, and the risk of biases is reduced. By leveraging the principle of generalized risk minimization, we realize an iterative disambiguation process that implements an implicit trade-off between cautious self-supervision and entropy minimization.

In an exhaustive empirical evaluation in the field of image classification, the enhanced expressiveness and uncertainty-awareness compared to conventional probabilistic self-supervision proved to yield superior generalization performance, especially in the regime of label-scarce semi-supervised learning. Moreover, the experiments have shown an improved network calibration when trained with credal self-supervision, as well as an increased efficiency when considering small compute budgets for training.

Motivated by these promising results, we plan to further elaborate on the idea of credal target modeling and extend it in various directions. For example, as we considered rather simple target sets so far, a thorough investigation of more sophisticated modeling techniques should be conducted. Such techniques may help, for instance, to sift out implausible classes early on, and lead to more precise pseudo-labels without compromising validity. Besides, as already mentioned, our CSSL framework is completely generic and not restricted to applications in image processing. Although we used it to extend FixMatch in this paper, it can extend any other self-training method, too. Elaborating on such extensions is another important aspect of future work.

Acknowledgments and Disclosure of Funding

This work was supported by the German Research Foundation (DFG) (Grant No. 420493178). Moreover, the authors gratefully acknowledge the funding of this project by computing time provided by the Paderborn Center for Parallel Computing (PC^2) and the research group of Prof. Dr. Marco Platzner.

References

[1] Eric Arazo, Diego Ortego, Paul Albert, Noel E. O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *Proc. of the International Joint Conference on Neural Networks, IJCNN, Glasgow, United Kingdom, July 19-24*, pages 1–8. IEEE, 2020.

[2] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems, NIPS, Montreal, Quebec, Canada, December 8-13*, pages 3365–3373, 2014.

[3] David Berthelot, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. ReMixMatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *Proc. of the 8th International Conference on Learning Representations, ICLR, Addis Ababa, Ethiopia, April 26-30*. OpenReview.net, 2020.

[4] David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. MixMatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS, Vancouver, BC, Canada, December 8-14*, pages 5050–5060, 2019.

[5] Vivien Cabannes, Alessandro Rudi, and Francis R. Bach. Structured prediction with partial labelling through the infimum loss. In *Proceedings of the 37th International Conference on Machine Learning, ICML, virtual, July 13-18*, volume 119, pages 1230–1239. PMLR, 2020.

[6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *CoRR*, abs/2104.14294, 2021.

[7] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. *Semi-Supervised Learning*. The MIT Press, 2006.

[8] Adam Coates, Andrew Y. Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proc. of the 14th International Conference on Artificial Intelligence and Statistics, AISTATS, Fort Lauderdale, FL, USA, April 11-13*, volume 15 of *JMLR Proceedings*, pages 215–223. JMLR.org, 2011.

[9] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. RandAugment: Practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

[10] Zihang Dai, Zhilin Yang, Fan Yang, William W. Cohen, and Ruslan Salakhutdinov. Good semi-supervised learning that requires a bad GAN. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, Long Beach, CA, USA, December 4-9*, pages 6510–6520, 2017.

[11] Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *Proc. of the IEEE International Conference on Computer Vision, ICCV, Santiago, Chile, December 7-13*, pages 1422–1430. IEEE Computer Society, 2015.

[12] Jingfei Du, Edouard Grave, Beliz Gunel, Vishrav Chaudhary, Onur Celebi, Michael Auli, Ves Stoyanov, and Alexis Conneau. Self-training improves pre-training for natural language understanding. *CoRR*, abs/2010.02194, 2020.

[13] Didier Dubois and Henri Prade. Possibility theory, probability theory and multiple-valued logics: A clarification. *Annals of Mathematics and Artificial Intelligence*, 32:35–66, 2004.

[14] Didier Dubois, Henri Prade, and Sandra Sandri. *On Possibility/Probability Transformations*, pages 103–112. Springer, 1993.

[15] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proc. of the 33nd International Conference on Machine Learning, ICML, New York City, NY, USA, June 19-24*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1050–1059. JMLR.org, 2016.

[16] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V. Le. DropBlock: A regularization method for convolutional networks. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, NeurIPS, Montreal, Quebec, Canada, December 3-8*, pages 10750–10760, 2018.

[17] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular depth estimation. In *Proc. of the IEEE/CVF International Conference on Computer Vision, ICCV, Seoul, Korea (South), October 27 - November 2*, pages 3827–3837. IEEE, 2019.

[18] Chengyue Gong, Dilin Wang, and Qiang Liu. AlphaMatch: Improving consistency for semi-supervised learning with alpha-divergence. *CoRR*, abs/2011.11779, 2020.

[19] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in Neural Information Processing Systems 17: Annual Conference on Neural Information Processing Systems, NIPS, Vancouver, BC, Canada, December 13-18*, pages 529–536, 2004.

[20] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

[21] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proc. of the 34th International Conference on Machine Learning, ICML, Sydney, NSW, Australia, August 6-11*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR, 2017.

[22] Eyke Hüllermeier and Weiwei Cheng. Superset learning based on generalized loss minimization. In *Proc. of the European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, Porto, Portugal, September 7-11, Proc. Part II*, volume 9285 of *LNCS*, pages 260–275. Springer, 2015.

[23] Kyungyul Kim, ByeongMoon Ji, Doyoung Yoon, and Sangheum Hwang. Self-knowledge distillation with progressive refinement of targets. *arXiv:2006.12000 [cs, stat]*, abs/2006.12000, 2021.

[24] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, Toronto, Canada, 2009.

[25] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *Proc. of the 5th International Conference on Learning Representations, ICLR, Toulon, France, April 24-26*. OpenReview.net, 2017.

[26] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, International Conference on Machine Learning, ICML, Atlanta, GA, USA, June 16-21*, volume 3, 2013.

[27] Isaac Levi. *The Enterprise of Knowledge: An Essay on Knowledge, Credal Probability, and Chance*. MIT Press, 1983.

[28] Junnan Li, Caiming Xiong, and Steven C. H. Hoi. CoMatch: Semi-supervised learning with contrastive graph regularization. *CoRR*, abs/2011.11183, 2020.

[29] Julian Lienen and Eyke Hüllermeier. From label smoothing to label relaxation. In *Proc. of the 35th AAAI Conference on Artificial Intelligence, virtual, February 2-9*, 2021.

[30] Julian Lienen and Eyke Hüllermeier. Instance weighting through data imprecisiation. *Int. J. Approx. Reason.*, 134:1–14, 2021.

[31] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *Proc. of the 5th International Conference on Learning Representations, ICLR, Toulon, France, April 24-26*. OpenReview.net, 2017.

[32] Subhabrata Mukherjee and Ahmed Hassan Awadallah. Uncertainty-aware self-training for few-shot text classification. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

[33] Rafael Müller, Simon Kornblith, and Geoffrey E. Hinton. When does label smoothing help? In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS, Vancouver, BC, Canada, December 8-14*, pages 4696–4705, 2019.

[34] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NIPS, Granada, Spain, November 12-17, Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[35] Avital Oliver, Augustus Odena, Colin Raffel, Ekin Dogus Cubuk, and Ian J. Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, NeurIPS, Montreal, Quebec, Canada, December 3-8*, pages 3239–3250, 2018.

[36] Hieu Pham, Zihang Dai, Qizhe Xie, Minh-Thang Luong, and Quoc V. Le. Meta Pseudo Labels. *arXiv:2003.10580 [cs, stat]*, abs/2003.10580, 2021.

[37] Antti Rasmus, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko. Semi-supervised learning with Ladder networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems, NIPS, Montreal, Quebec, Canada, December 7-12*, pages 3546–3554, 2015.

[38] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh Singh Rawat, and Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *CoRR*, abs/2101.06329, 2021.

[39] Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-supervised self-training of object detection models. In *Proc. of the 7th IEEE Workshop on Applications of Computer Vision / IEEE Workshop on Motion and Video Computing, WACV/MOTION, Breckenridge, CO, USA, January 5-7*, pages 29–36. IEEE Computer Society, 2005.

[40] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems, NIPS, Barcelona, Spain, December 5-10*, pages 1163–1171, 2016.

[41] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

[42] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, 2014.

[43] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in NLP. In *Proc. of the 57th Conference of the Association for Computational Linguistics, ACL, Florence, Italy, July 28 - August 2, Volume 1: Long Papers*, pages 3645–3650. Association for Computational Linguistics, 2019.

[44] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the Inception architecture for computer vision. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Las Vegas, NV, USA, June 27-30*, pages 2818–2826. IEEE Computer Society, 2016.

[45] Jafar Tanha, Maarten van Someren, and Hamideh Afsarmanesh. Semi-supervised self-training for decision tree classifiers. *Int. J. Mach. Learn. Cybern.*, 8(1):355–370, 2017.

[46] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NeurIPS, Long Beach, CA, USA, December 4-9*, pages 1195–1204, 2017.

[47] Trung Quang Tran, Mingu Kang, and Daeyoung Kim. ReRankMatch: Semi-supervised learning with semantics-oriented similarity representation. *CoRR*, abs/2102.06328, 2021.

[48] Jesper E. van Engelen and Holger H. Hoos. A survey on semi-supervised learning. *Mach. Learn.*, 109(2):373–440, 2020.

[49] Xiao Wang, Daisuke Kihara, Jiebo Luo, and Guo-Jun Qi. EnAET: A self-trained framework for semi-supervised and supervised learning with ensemble transformations. *IEEE Trans. Image Process.*, 30:1639–1647, 2021.

[50] Qizhe Xie, Zihang Dai, Eduard H. Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

[51] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with Noisy Student improves ImageNet classification. *arXiv:1911.04252 [cs, stat]*, abs/1911.04252, 2020.

[52] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *Proc. of the British Machine Vision Conference, BMVC, York, UK, September 19-22*, 2016.

[53] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. Mixup: Beyond empirical risk minimization. In *Proc. of the 6th International Conference on Learning Representations, ICLR, Vancouver, BC, Canada, April 30 - May 3*. OpenReview.net, 2018.

[54] Zhedong Zheng and Yi Yang. Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. *Int. J. Comput. Vis.*, 129(4):1106–1120, 2021.

[55] Tianyi Zhou, Shengjie Wang, and Jeff Bilmes. Time-consistent self-supervision for semi-supervised learning. In *Proc. of the 37th International Conference on Machine Learning, ICML, virtual, July 13-18*, volume 119 of *Proceedings of Machine Learning Research*, pages 11523–11533. PMLR, 2020.

6 Conformal Credal Self-Supervised Learning

Author Contribution Statement

The idea of combining credal self-supervised learning with conformal prediction originates from the author. Both the author of this thesis and Eyke Hüllermeier contributed to the further development of this idea. The paper was initially written by the author, and subsequently revised by all authors. While the main implementation and experimentation on image classification datasets was done by the author, Caglar Demir contributed experiments for the domain of knowledge graph embeddings.

Supplementary Material

An appendix to the paper is provided in Appendix C. The code of the official implementation is provided at `https://github.com/julilien/C2S2L` (Apache 2.0 license).

Proceedings of Machine Learning Research 204:1–20, 2023 Conformal and Probabilistic Prediction with Applications

Conformal Credal Self-Supervised Learning

Julian Lienen[*] julian.lienen@upb.de
Caglar Demir caglar.demir@upb.de
Paderborn University, Germany

Eyke Hüllermeier eyke@lmu.de
University of Munich (LMU), Germany

Editor: Harris Papadopoulos, Khuong An Nguyen, Henrik Boström and Lars Carlsson

Abstract

In semi-supervised learning, the paradigm of self-training refers to the idea of learning from pseudo-labels suggested by the learner itself. Recently, corresponding methods have proven effective and achieve state-of-the-art performance, e.g., when applied to image classification problems. However, pseudo-labels typically stem from ad-hoc heuristics, relying on the quality of the predictions though without guaranteeing their validity. One such method, so-called credal self-supervised learning, maintains pseudo-supervision in the form of sets of (instead of single) probability distributions over labels, thereby allowing for a flexible yet uncertainty-aware labeling. Again, however, there is no justification beyond empirical effectiveness. To address this deficiency, we make use of conformal prediction, an approach that comes with guarantees on the validity of set-valued predictions. As a result, the construction of credal sets of labels is supported by a rigorous theoretical foundation, leading to better calibrated and less error-prone supervision for unlabeled data. Along with this, we present effective algorithms for learning from credal self-supervision. An empirical study demonstrates excellent calibration properties of the pseudo-supervision, as well as the competitiveness of our method on several image classification benchmark datasets.

Keywords: Credal sets, semi-supervised learning, inductive conformal prediction, self-training, label relaxation

1. Introduction

In the recent years, machine learning applications, particularly deep learning models, have benefited greatly from the extensive amount of available data. While traditional supervised learning methods commonly rely on curated, precise target information, there is an essentially unlimited availability of unlabeled data, e.g., crowd-sourced images in computer vision applications. Semi-supervised learning (Chapelle et al., 2006) aims to leverage this source of information for further optimizing models, as successfully demonstrated by a wide range of algorithms (Berthelot et al., 2019, 2020; Sohn et al., 2020; Xie et al., 2020a). Among other approaches, so-called *self-training* (Lee, 2013) emerged as a simple yet effective paradigm to facilitate learning from additional unlabeled data. To this end, the model to be trained is used to predict "pseudo-labels" for the unlabeled data, which are then added to the labeled training data.

When learning probabilistic classifiers, e.g., to classify images, pseudo-labels can come in different forms. Traditionally, such supervision is typically in the form of a probability

[*] Corresponding author

distribution, either degenerate ("one-hot" encoded, hard) (Lee, 2013; Sohn et al., 2020) or non-degenerate (soft) (Berthelot et al., 2019, 2020). As pointed out by Lienen and Hüllermeier (2021), this form can be questioned from a data modeling perspective: Not only is a single distribution unlikely to match the true ground-truth, and may hence to bias the learner, but also incapable of expressing uncertainty about the ground-truth distribution. While uncertainty-aware approaches address these shortcomings by accompanying the targets with additional qualitative information (e.g., as in (Rizve et al., 2021)), so-called *credal self-supervised learning* (CSSL) (Lienen and Hüllermeier, 2021) replaces the targets by *credal sets*, i.e., sets of probability distributions assumed to guarantee (or at least likely) covering the true distribution. Such set-valued targets, for which CSSL employs a generalized risk minimization to enable learning, relieve the learner from committing to a single distribution, while facilitating uncertainty-awareness through increased expressivity.

Although all of the aforementioned approaches have demonstrated their effectiveness to the problem of semi-supervised learning, none of them entails meaningful guarantees about the validity of the pseudo-labels — their quality is solely subject to the model confidence itself without an objective error probability. Moreover, rather ad-hoc heuristics are considered to control the influence of vague beliefs on the learning process (e.g., by selecting pseudo-labels surpassing a confidence threshold). For instance, CSSL derives credal pseudo-supervision based on the predicted probability scores. Although it exhibits uncertainty-awareness by using entropy minimization (less entropy results in smaller sets), this may be misleading for overconfident yet poorly generalizing models. Again, no objective guarantee can be given for the validity of the credal set. This raises the quest for a pseudo-labeling procedure providing exactly that kind of guarantee, i.e., validating pseudo-supervision beyond mere empirical effectiveness.

Fortunately, the framework of conformal prediction (CP) (Vovk et al., 2005; Shafer and Vovk, 2008) provides a tool-set to satisfy such demands. As such, CP induces prediction regions quantifying the uncertainty for candidate values as outcome for a query instance by measuring their non-conformity ("strangeness") with previously observed data with known outcomes. Much like in classical hypothesis testing, it computes p-values for candidate outcomes (e.g., for each class y') as a criterion for hypothesis rejection, i.e., whether to include a value y' in the prediction set with a certain amount of confidence. Based on very mild technical assumptions, CP provides formal guarantees for the validity of prediction regions and is able to control the probability of an invalid prediction (not covering the true label).

In our work, we combine the technique of inductive conformal prediction, an efficient variant of CP, with the idea of credal self-supervised learning, leading to a method we dub *conformal credal self-supervised learning* (CCSSL). Instead of deriving credal sets based on an ad-hoc estimate of the model confidence, we proceed from a possibilistic interpretation of conformal predictions (Cella and Martin, 2022) to construct conformal credal labels. By this, pseudo-labels of that kind provide the validity guarantees implied by conformal predictors, laying a rigorous theoretical foundation for a set-valued pseudo-labeling strategy and thus paving the way for more thorough theoretical analyses of self-training in semi-supervised learning. To enable learning from conformal credal self-supervision, we further provide an effective algorithmic solution for generalized empirical risk minimization on set-valued targets. An exhaustive empirical study on several image classification datasets demonstrates the usefulness of our method for effective and reliable semi-supervised learning — leveraging well calibrated pseudo-supervision, it is shown to improve the state-of-the-art in terms of generalization performance.

2. Related Work

Semi-supervised learning describes the learning paradigm where the aim is to leverage the potential of unlabeled in addition to labeled data to improve learning and generalization. As it is easy to lose track of related work due to the wide range of proposed approaches in the recent years, we focus here on classification methods applied to computer vision applications that are closely related to our work. For a more comprehensive overview, we refer to (Chapelle et al., 2006) and (van Engelen and Hoos, 2020).

Nowadays, so-called *self-training* (or self-supervised) methods protrude among these methods by providing a simple yet effective learning methodology to make use of unlabeled data. Such methods can be found in a wide range of domains, including computer vision (Rosenberg et al., 2005; Doersch et al., 2015; Godard et al., 2019; Xie et al., 2020b) and natural language processing (Du et al., 2021). As self-training can be considered as a general learning paradigm, where a model suggests itself labels to learn from, it has been wrapped around various model types, e.g., support vector machines (Lienen and Hüllermeier, 2021), decision trees (Tanha et al., 2017) and most prominently with neural networks (Oliver et al., 2018). Notably, it lays the foundation for so-called distillation models, e.g., in self-distillation (Kim et al., 2021) or student-teacher setups (Xie et al., 2020b; Pham et al., 2021). It is further popular for unsupervised pretraining, e.g., as described in (Grill et al., 2020; Caron et al., 2021).

Uncertainty-awareness (Hüllermeier and Waegeman, 2021) is a critical aspect for a cautious pseudo label selection to learn from, without exposing the model to the risk of confirmation biases (Arazo et al., 2020). This aspect has been considered in previous works by different means. (Rizve et al., 2021) employs Bayesian sampling techniques, such as MC-Dropout (Gal and Ghahramani, 2016) or DropBlock (Ghiasi et al., 2018), to estimate uncertainty of a prediction, which is then used as an additional filter criterion. (Ren et al., 2020) uses an adaptive instance weighting approach to control the influence of individual pseudo labels. Moreover, credal self-supervised learning (Lienen and Hüllermeier, 2021) expresses certainty by the size of maintained credal sets as pseudo-supervision. Also, learning from softened probability distribution can also be considered as a way to suppress over-confidence tendencies to learn in a more cautious way (Berthelot et al., 2019; Arazo et al., 2020). Lastly, several domain-specific adaptions have been proposed, e.g., for text classification (Mukherjee and Awadallah, 2020) or semantic segmentation (Zheng and Yang, 2021).

As prominently used throughout this work, conformal prediction (Vovk et al., 2005; Shafer and Vovk, 2008; Balasubramanian et al., 2014) provides an elegant framework to naturally express model prediction uncertainty in a set-valued form. We refer the interested reader to (Zeni et al., 2020) for a more comprehensive overview beyond the covered literature here. Whereas conformal prediction initially proceeded from a transductive form in an online setting (Vovk et al., 2005), its high complexity called for more efficient variants. To this end, several variations have been proposed, where inductive conformal prediction

(Papadopoulos et al., 2007) most prominently draw attention by assuming a separated calibration split available used for non-conformity measurement. Beyond this, several alternative approaches varying data assumption or algorithmic improvements have been proposed (e.g., as described in (Lei et al., 2018; Kim et al., 2020)). Recently, Cella and Martin (2021, 2022) suggest a reinterpretation of conformal transducer as plausibility contours, allowing to derive validity guarantees on predictive distributions rather than prediction regions, which is being used throughout our work.

3. Conformal Credal Pseudo-Labeling

In this section, we revisit credal pseudo-labeling and introduce conformal prediction as a framework for reliable prediction and an integral part of our conformal credal pseudo-labeling approach.

3.1. Credal Pseudo-Labeling

In supervised classification, one typically assumes instances $\boldsymbol{x} \in \mathcal{X}$ of an instance space $\mathcal{X}$ to be associated with a ground-truth in the form of a conditional probability distribution $p^*(\cdot \,|\, \boldsymbol{x}) \in \mathbb{P}(\mathcal{Y})$ over the class space $\mathcal{Y} = \{y_1, ..., y_K\}$. Training a probabilistic classifier $\hat{p} : \mathcal{X} \longrightarrow \mathbb{P}(\mathcal{Y})$ commonly involves the optimization of a probabilistic surrogate loss function $\mathcal{L} : \mathbb{P}(\mathcal{Y}) \times \mathbb{P}(\mathcal{Y}) \longrightarrow \mathbb{R}_+$ (e.g., cross-entropy) comparing the model prediction $\hat{p}$ to a proxy distribution p reflecting the true target $p^*(\cdot \,|\, \boldsymbol{x})$. Although it would be desirable, most classification datasets do not give direct access to p^*, but only to a realization $y \in \mathcal{Y}$ of the random variable $Y \sim p^*(\cdot \,|\, \boldsymbol{x})$, whence a degenerate distribution p_y with $p_y(y) = 1$ and $p_y(y') = 0$ for $y' \neq y$ is often considered as a proxy. Other data modeling techniques such as label smoothing (Szegedy et al., 2016) also consider "softened" versions of approaching p_y.

What methods of this kind share is their reliance on a single probability distribution as target information. As already said, this information does not allow for representing any uncertainty about the ground-truth distribution p^*: Predicting a precise distribution $\hat{p}$, the learner pretends a level of certainty that is not warranted. This is a sharp conflict with the observation that, in practice, such predictions tend to be poorly calibrated and overconfident (Müller et al., 2019), and are hence likely to bias the learner.

To overcome these shortcomings, Lienen and Hüllermeier (2021) suggest to the replace a single probabilistic prediction by a *credal set* $\mathcal{Q} \subseteq \mathbb{P}(\mathcal{Y})$, i.e., as set of (candidate) probability distributions. This set $\mathcal{Q}$ is supposed to cover the ground-truth p^*, very much like a confidence interval is supposed to cover a ground-truth parameter in classical statistics. Thus, the learner is able to represent its uncertainty about the ground-truth p^* in a more faithful way. More specifically, the learner's (epistemic) uncertainty is in direct correspondence with the size of the credal set. It can represent complete ignorance about p^* (by setting $\mathcal{Q} = \mathbb{P}(\mathcal{Y})$), complete certainty ($\mathcal{Q} = \{p\}$), but also epistemic states in-between these extremes.

One way to specify credal sets is by means of so-called *possibility distributions* (Dubois and Prade, 2004) $\pi : \mathcal{Y} \longrightarrow [0, 1]$, which induce a possibility measure $\Pi : 2^{\mathcal{Y}} \longrightarrow [0, 1]$ by virtue of $\Pi(Y) = \max_{y \in Y} \pi(y)$ for all $Y \subseteq \mathcal{Y}$. Possibility degrees can be interpreted as upper probabilities, so that a possibility distribution π specifies the following set of probability distributions:

$$\mathcal{Q}_\pi := \Big\{ p \in \mathbb{P}(\mathcal{Y}) \,|\, \forall Y \subseteq \mathcal{Y} : \\ P(Y) = \sum_{y \in Y} p(y) \leq \max_{y \in Y} \pi(y) = \Pi(Y) \Big\} \ . \tag{1}$$

To guarantee $\mathcal{Q}_\pi \neq \emptyset$, the distribution π is normalized such that $\max_{y \in \mathcal{Y}} \pi(y) = 1$, i.e., there is at least one label y considered fully plausible.

When considering the setting of semi-supervised learning, credal labeling is employed as a pseudo labeling technique within the framework of credal self-supervised learning (CSSL) (Lienen and Hüllermeier, 2021): For each unlabeled instance, a credal set is maintained to express the current belief about p^* in terms of a confidence region $\mathcal{Q} \subseteq \mathbb{P}(\mathcal{Y})$. To this end, CSSL derives a possibility distribution π by an ad-hoc heuristic that assigns full plausibility $\pi(\hat{y}) = 1$ to the class $\hat{y}$ for which the predicted probability is highest, and determines a constant plausibility degree α for all other classes $y' \neq \hat{y}$; the latter depends on the learner's confidence as well as the class prior and the prediction history.

Although CSSL has proven to provide competitive generalization performance, the credal set construction heuristic lacks a solid theoretical foundation and does not provide any quality guarantees. Especially in the case of overconfident models, the credal sets may not reflect uncertainty properly and misguide the learner. Moreover, the class prior and the prediction history employed in the specification of possibility distribution constitute yet another potential source of bias in the learning process. This raises the question whether credal pseudo-labeling can be based on a more solid theoretical foundation and equipped with validity guarantees. An affirmative answer to this question is offered by the framework of conformal prediction.

3.2. Inductive Conformal Prediction

Conformal prediction (CP) (Vovk et al., 2005) is a distribution-free uncertainty quantification framework that judges a prediction for a query by its "non-conformity" with data observed before. While it has been originally introduced and extensively analyzed in an online setting, we refer to an offline ("batch-wise") variant as typically considered in supervised learning settings. Several alternative variants of conformal prediction have been proposed in the past, most notably transductive and inductive methods. In its original formulation, the former requires a substantial amount of training and is clearly unsuitable in large-scale learning scenarios such as typical deep learning applications. This issue has been addressed in so-called *inductive conformal prediction* (ICP) (Papadopoulos, 2008) by alleviating the computational demands through additional calibration data.

Assume we are given training data $\mathcal{D}_{\text{train}} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N \subset (\mathcal{X} \times \mathcal{Y})^N$ and calibration data $\mathcal{D}_{\text{calib}} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^L \subset (\mathcal{X} \times \mathcal{Y})^L$ comprising N resp. L i.i.d. observations. For a given query (resp. test) instance $\boldsymbol{x}_{N+1}$, the goal of conformal prediction is to provide an uncertainty quantification with provable guarantees about the likeliness of each possible candidate $\hat{y} \in \mathcal{Y}$ being the true outcome associated with $\boldsymbol{x}_{N+1}$. This is done by measuring the conformity of $(\boldsymbol{x}_{N+1}, \hat{y})$ with $\mathcal{D}_{\text{calib}}$ for each candidate label $\hat{y}$.

To this end, conformal prediction calculates a *non-conformity measure* $\alpha : (\mathcal{X} \times \mathcal{Y})^m \times (\mathcal{X} \times \mathcal{Y}) \longrightarrow \mathbb{R}$, which indicates how well a pair of a query instance and a candidate label conforms to an observed sequence of m pairs. Typically, the non-conformity $\alpha(\mathcal{D}, (\boldsymbol{x}, y))$ involves the training of an underlying predictor (a model $\hat{p}$ in our case) on $\mathcal{D}$, whose prediction $\hat{p}(\boldsymbol{x})$ is then compared to y. In a probabilistic learning scenario, common choices are

$$\alpha(\mathcal{D}, (\boldsymbol{x}, y)) = \max_{y_j \neq y} \hat{p}_{\mathcal{D}}(\boldsymbol{x})(y_j) - \hat{p}_{\mathcal{D}}(\boldsymbol{x})(y) \tag{2}$$

and

$$\alpha(\mathcal{D}, (\boldsymbol{x}, y)) = \frac{\max_{y_j \neq y} \hat{p}_{\mathcal{D}}(\boldsymbol{x})(y_j)}{\hat{p}_{\mathcal{D}}(\boldsymbol{x})(y) + \gamma} \quad , \tag{3}$$

where $\hat{p}_{\mathcal{D}}$ is a probabilistic classifier trained on $\mathcal{D}$, $\hat{p}(\cdot)(y')$ denotes the output probability for class y' and $\gamma \geq 0$ is a sensitivity parameter (Papadopoulos et al., 2007). In our case, we set $\mathcal{D}$ to the training data $\mathcal{D}_{\text{train}}$.

In ICP, non-conformity scores α_i are calculated for all $(\boldsymbol{x}_i, y_i) \in \mathcal{D}_{\text{calib}}$, i.e., it is determined for each calibration instance how well it conforms with the underlying training data $\mathcal{D}_{\text{train}}$. For the uncertainty quantification of the prediction for a query instance $\boldsymbol{x}_{N+1}$, the non-conformity $\alpha_{N+1}^{\hat{y}} := \alpha(\mathcal{D}_{\text{train}}, (\boldsymbol{x}_{N+1}, \hat{y}))$ is calculated for each candidate label $\hat{y} \in \mathcal{Y}$. Given the non-conformity values $\alpha_1, \ldots, \alpha_L$ associated with the calibration data, these non-conformity values can be used to construct *p-values* for each candidate label $\hat{y}$, comparable to the notion of p-values in traditional statistics:

$$\pi_{N+1}(\hat{y}) = \frac{|\{\alpha_i \geq \alpha_{N+1}^{\hat{y}} \,|\, i \in \{1, ..., L\}\}| + 1}{L + 1} \quad . \tag{4}$$

Consequently, one can define a conformal predictor Γ_δ with confidence $1 - \delta$ by

$$\Gamma_\delta(\mathcal{D}_{\text{calib}}, \boldsymbol{x}_{N+1}) = \Big\{ \hat{y} \in \mathcal{Y} : \pi_{N+1}(\hat{y}) \geq \delta \Big\}. \tag{5}$$

Such a Γ_δ can be shown to cover the true label y_{N+1} associated with $\boldsymbol{x}_{N+1}$ with high probability:

$$\Pr\Big(y_{N+1} \in \Gamma_\delta(\mathcal{D}_{\text{calib}}, \boldsymbol{x}_{N+1}) \Big) \geq 1 - \delta \quad , \tag{6}$$

where the probability is taken over $\boldsymbol{x}_{N+1}$ and $\mathcal{D}_{\text{calib}}$. This property is typically referred to as (marginal) *validity*, however, following (Cella and Martin, 2022), we refer to (6) as *weak validity*. Note that this holds for any underlying probability distribution, any $\delta \in (0, 1)$, and $N \in \mathbb{N}_+$. Nevertheless, practically speaking, small calibration datasets $\mathcal{D}_{\text{calib}}$ may be problematic if a certain granularity in the confidence degree is desired (Johansson et al., 2015).

3.2.1. Conformalized Predictive Distributions

While the notion of weak validity applies to set-valued prediction regions, its application as quantification of *predictive distributions* is not obvious. In the realm of probabilistic classification (as considered here), one may wonder how CP can be applied to provide a meaningful uncertainty quantification. Recently, Cella and Martin (2021) proposed an interpretation of p-values π (cf. Eq. (4)) in terms of possibility degrees, i.e., upper probabilities of the event that the respective candidate label is the true outcome. Consequently, these possibilities π induce possibility measures Π, such that

$$\Pi(A) := \sup_{\hat{y} \in A} \pi(\hat{y}), \qquad A \subseteq \mathcal{Y} \, . \tag{7}$$

With $\sup_{\hat{y} \in \mathcal{Y}} \pi(\hat{y}) = 1$ for all $\mathcal{D}_{\text{train}}$ and assuming that π is stochastically no smaller than the uniform distribution $\mathcal{U}(0,1)$ under any underlying (ground-truth) probability distribution, probabilistic predictors satisfy the *strong validity* property defined as follows:

$$\Pr\left(\Pi(A) \leq \delta, y_{N+1} \in A\right) \leq \delta \tag{8}$$

holds for any $\delta \in (0,1)$, $A \subseteq \mathcal{Y}$, training data $\mathcal{D}_{\text{train}}$ and any underlying true probability distribution.

3.3. Conformal Credal Pseudo-Labeling

Strongly valid possibility distributions π of this kind can be directly employed in the credal set formulation in (1) to bound the space of probability distributions. Relating to the self-training paradigm we consider in a semi-supervised setting, where a model suggests itself credal sets as pseudo-supervision to learn from, we refer to such credal sets as *conformal credal pseudo-labels*. Note that such pseudo-labels do not necessarily need to be constructed by an inductive conformal prediction procedure, but can used any CP methodology with the same guarantees.

To guarantee property (8) for the possibility distribution π defined in (4), this distribution needs to be normalized such that $\max_{\hat{y} \in \mathcal{Y}} \pi(\hat{y}) = 1$ to ensure strong validity and that $\mathcal{Q}_\pi \neq \emptyset$. A commonly applied normalization is suggested in (Cella and Martin, 2021):

$$\pi(\hat{y}) = \frac{\pi(\hat{y})}{\max_{y' \in \mathcal{Y}} \pi(y')} \quad . \tag{9}$$

By the described conformal credal labeling method, we replace the ad-hoc construction of sets in CSSL by a more profound technique with formal guarantees. More precisely, we build the learner's uncertainty-awareness on top of an *objective* quality criterion deduced from its accuracy on the calibration data. This not only provides strong validity guarantees, but also paves the path to a more rigorous theoretical analysis of self-supervised credal learning. Nevertheless, these guarantees come at a cost: By applying a conformal prediction procedure, one either provokes an increased computational overhead (for instance in a transductive CP setting) or a decrease in data efficiency by requiring additional calibration data. However, as will be seen in the empirical evaluation, the improved pseudo-label quality makes up for this in terms of generalization performance.

4. Conformal Credal Self-Supervised Learning

In the following, we provide an overview of our methodology for effective semi-supervised learning from conformal credal pseudo-labels.

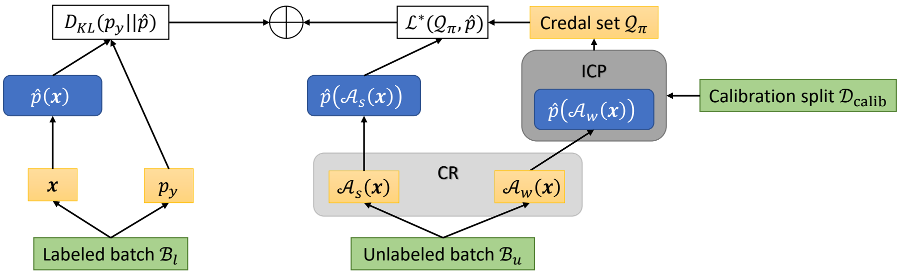

$D_{KL}(p_y||\hat{p})$
$\mathcal{L}^*(\mathcal{Q}_\pi, \hat{p})$
Credal set $\mathcal{Q}_\pi$
ICP
$\hat{p}(\boldsymbol{x})$
$\hat{p}(\mathcal{A}_s(\boldsymbol{x}))$
$\hat{p}(\mathcal{A}_w(\boldsymbol{x}))$
Calibration split $\mathcal{D}_{\text{calib}}$
$\boldsymbol{x}$
p_y
CR
$\mathcal{A}_s(\boldsymbol{x})$
$\mathcal{A}_w(\boldsymbol{x})$
Labeled batch $\mathcal{B}_l$
Unlabeled batch $\mathcal{B}_u$

Figure 1: Overview over CCSSL: While learning on the labeled batch $\mathcal{B}_l$ is performed conventionally, the labels for the unlabeled batch $\mathcal{B}_u$ are constructed by an inductive conformal prediction (ICP) procedure provided calibration data $\mathcal{D}_{\text{calib}}$, employing consistency regularization (CR) for confirmation bias mitigation.

4.1. Overview

In our approach, which we call *conformal credal self-supervised learning* (CCSSL), we combine credal self-supervised learning as proposed in (Lienen and Hüllermeier, 2021) with conformal credal labels. We thus replace the credal set construction in CSSL to ensure guarantees of the pseudo-label's validity. Although both CSSL and CCSSL are not specifically tailored to a particular domain, we adopt the framework of the former with a focus on image classification here. Fig. 1 gives an overview of the algorithm, its pseudo-code can be found in the appendix.[1]

In each iteration, we observe batches of labeled $\mathcal{B}_l = \{(\boldsymbol{x}_i, p_{y_i})\}_{i=1}^B$ and unlabeled instances $\mathcal{B}_u = \{\boldsymbol{x}_i\}_{i=1}^{\mu B}$, where $\mu \geq 1$ is the multiplicity of unlabeled over labeled instances in each batch. Here, we consider degenerate distributions p_{y_i} as target information in $\mathcal{B}_l$. For the instances $(\boldsymbol{x}, p_y) \in \mathcal{B}_l$, we compute the Kullback-Leibler divergence $D_{KL}(p_y \,||\, \hat{p}(\boldsymbol{x}))$ as loss $\mathcal{L}_l$.

To calculate the loss on unlabeled instances $\boldsymbol{x} \in \mathcal{B}_u$ leveraging our conformal credal labeling approach, we adopt the *consistency regularization* (CR) framework (Bachman et al., 2014; Sajjadi et al., 2016), which aims to mitigate the so-called *confirmation bias* (Arazo et al., 2020) that describes the manifestation of misbeliefs of an accurate learner. CR enforces consistent predictions for different perturbed appearances of an instance, and has been successfully combined with pseudo-labeling, whereby Sohn et al. (2020) propose a particularly simple scheme: An unlabeled instance $\boldsymbol{x} \in \mathcal{X}$ is augmented by a weak (e.g., horizontal flipping) and a strong (e.g., RandAugment (Cubuk et al., 2020) or CTAugment (Berthelot et al., 2020)) augmentation policy $\mathcal{A}_w : \mathcal{X} \longrightarrow \mathcal{X}$ and $\mathcal{A}_s : \mathcal{X} \longrightarrow \mathcal{X}$, respectively. The weakly-augmented representation $\mathcal{A}_w(\boldsymbol{x})$ is then used to construct a pseudo-label based on $\hat{p}(\mathcal{A}_w(\boldsymbol{x}))$, which is finally compared to $\hat{p}(\mathcal{A}_s(\boldsymbol{x}))$ to compute the loss.

Transferred to our methodology, we derive conformal predictions for the weakly-augmented instance $\mathcal{A}_w(\boldsymbol{x})$ in an inductive manner based on a previously separated calibration dataset $\mathcal{D}_{\text{calib}}$. To this end, a (strongly valid) possibility distribution π over all possible

[1] The appendix, along with the official implementation, is externally available at https://github.com/julilien/C2S2L.

labels $y \in \mathcal{Y}$ is determined for the prediction $\hat{p}(\mathcal{A}_w(\boldsymbol{x}))$, which is used to construct a credal target set $\mathcal{Q}_\pi$ according to (1). Finally, $\mathcal{Q}_\pi$ is compared to the prediction $\hat{p}(\mathcal{A}_s(\boldsymbol{x}))$ in terms of a generalized probabilistic loss $\mathcal{L}^* : 2^{\mathbb{P}(\mathcal{Y})} \times \mathbb{P}(\mathcal{Y}) \longrightarrow \mathbb{R}_+$, which we shall detail in Section 4.3.

Altogether, the training objective is given by

$$\begin{aligned}\mathcal{L} = & \overbrace{\frac{1}{|\mathcal{B}_l|} \sum_{(\boldsymbol{x}_i, p_{y_i}) \in \mathcal{B}_l} D_{KL}(p_{y_i} \,||\, \hat{p}(\boldsymbol{x}_i))}^{\mathcal{L}_l} \\ & + \lambda_u \underbrace{\frac{1}{|\mathcal{B}_u|} \sum_{\boldsymbol{x}_i \in \mathcal{B}_u} \mathcal{L}^*(Q_{\pi_i}, \hat{p}((\mathcal{A}_s(\boldsymbol{x})_i))}_{\mathcal{L}_u} \ ,\end{aligned} \tag{10}$$

where $\lambda_u \geq 0$ weights the importance of the unlabeled loss part $\mathcal{L}_u$.

4.2. Validity Mitigates Confirmation Biases

Many of the recent SSL approaches, including CSSL, leverage techniques such as consistency regularization to mitigate confirmation biases. Intuitively speaking, it enforces a smooth decision boundary in the neighborhood of (augmented) unlabeled instances, also referred to as *local consistency* (Wei et al., 2021). For instance, classifiers trained on images showing cars provided in the training dataset predict consistent labels for similar cars that differ in the color, perspective changes or in an alternative scenery. If a certain proximity of the pseudo-labeled instances to the known labeled instances in the latent feature space is preserved, known as the *expansion assumption*, CR can lead to a global class-wise consistency, which can "de-noise" wrong pseudo-labels. Hence, it serves as a means to alleviate the problem of misguidance through mislabels.

However, the expansion assumption often appears too unrealistic in practice. For instance, violations happen in cases where the neighborhood of an (unlabeled) instance is not homogeneously populated by instances of the same true class. Typical examples for such situations are highly uncertain classification problems in which individual classes are hard to separate, e.g., distinguishing model variants of a car based on subtle visual differences such as badges, or data with imbalanced class frequencies. In the latter case, some classes dominate underrepresented ones, such that the neighborhood of unlabeled instances may be mostly populated by instances from different classes, thereby violating the expansion assumption. Consequently, from an empirical risk minimization point of view, it might be more reasonable to attribute larger regions populated by unlabeled instances from the minority class to a majority classes. This leaves the former overlooked, so that the correction of wrong pseudo-labels is not possible anymore.

Credal labels as constructed in CSSL are incapable of solving the issue of the expansion violation assumption and hence ineffective CR: Similar to single-point probabilistic pseudo-labeling methods, the pseudo-labels have no error guarantees that the actual true class is adequately incorporated in the target. As opposed to this, conformal credal pseudo-labels can provide such, which is why their validity guarantee can be regarded as a second fallback when combined with CR. When the expansion assumption is violated, a higher validity of credal sets oppose committing to a misbelief in a too premature manner by a more cautious learning behavior. As will be seen in the experimental evaluation on commonly used semi-supervised image classification benchmarks, conformal (and hence valid) credal sets indeed show a more robust behavior towards confirmation biases when facing highly uncertain neighborhoods (cf. Sec. 5.2). In addition, aiming to isolate the contributions of CR and validity in a different setting, we provide further experiments on imbalanced data in the appendix.

4.3. Generalized Credal Learning

As set-valued targets are provided as (pseudo-)supervision for the unlabeled instances, a generalization of a single-point probability loss $\mathcal{L}$ is required. Here, we follow (Lienen and Hüllermeier, 2021) and leverage a generalized empirical risk minimization approach by minimizing the *optimistic superset loss* (Hüllermeier and Cheng, 2015; Cabannes et al., 2020)

$$\mathcal{L}^*(\mathcal{Q}_\pi, \hat{p}) := \min_{p \in \mathcal{Q}_\pi} \mathcal{L}(p, \hat{p}) \enspace , \tag{11}$$

with $\mathcal{L} = D_{KL}$ in our case. This generalization is motivated by the idea of *data disambiguation* (Hüllermeier and Cheng, 2015), in which targets are disambiguated within imprecise sets according to their plausibility in the overall loss minimization context over all data points, leading to a minimal empirical risk with respect to the original loss $\mathcal{L}$.

While credal sets as considered in CSSL are of rather simplistic nature, credal sets with arbitrary (but normalized) possibility distributions as introduced in Section 3.3 impose a more complex optimization problem. More precisely, consider a possibility distribution π and denote $\pi_i := \pi(y_i)$. Without loss of generality, let the possibilities be ordered, i.e., $0 \leq \pi_1 \leq \ldots, \leq \pi_K = 1$. Then, it has been shown that the following holds for any distribution $p \in \mathcal{Q}_\pi$ with $p_i := p(y_i)$ (Delgado and Moral, 1987):

$$p \in \mathcal{Q}_\pi \Leftrightarrow \sum_{k=1}^{i} p_k \leq \pi_i \enspace , \qquad i \in \{1, ..., K\} \tag{12}$$

The resulting set of inequality constraints induces a convex polytope (Kroupa, 2006), such that the optimization problem in (11) becomes the problem of finding the closest point in a convex polytope (here $\mathcal{Q}_\pi$) with minimal distance to a given query point ($\hat{p}$). We provide examples illustrating such convex credal polytopes in the appendix.

Optimization problems of this kind are not new, and many approximate solutions have been proposed in the past, including projected gradient algorithms (Bahmani and Raj, 2011), conditional gradient descent (alias Frank-Wolfe algorithms) (Frank and Wolfe, 1956; Jaggi, 2013), and other projection-free stochastic methods (Li et al., 2020). However, we found that one can take advantage of the structure of the problem (12) to induce a precise and average-case efficient algorithmic solution to (11).

Algorithm 1 lists the procedure to determine the loss $\mathcal{L}^*(\mathcal{Q}_\pi, \hat{p})$. The key idea is to consider each face of the convex polytope being associated with a particular possibility constraint π_i. The algorithm then determines the optimal face on which the prediction $\hat{p}$ can be projected without violating the constraint for any $\pi_j \leq \pi_i$, which is guaranteed to provide the optimal solution for the classes y_j. In an iterative scheme, this procedure is

Algorithm 1 Generalized Credal Learning Loss

```
Require: Predicted distribution $\hat{p} \in \mathbb{P}(\mathcal{Y})$, (normalized) possibility distribution $\pi : \mathcal{Y} \longrightarrow [0,1]$
  if $\hat{p} \in \mathcal{Q}_\pi$ then return $D_{KL}(\hat{p} \,||\, \hat{p}) = 0$
  Initialize set of unassigned classes $Y = \mathcal{Y}$
  while $Y$ is not empty do
    Determine $y^* \in Y$ with highest $\pi(y^*)$, such that the probabilities
        $$\bar{p}(y) = \left( \pi(y^*) - \sum_{y' \notin Y} p^r(y') \right) \cdot \frac{\hat{p}(y)}{\sum_{y' \in Y'} \hat{p}(y')}$$
        for all $y \in Y' := \{ y \in Y \,|\, \pi(y) \leq \pi(y^*) \}$ do not violate the constraints in Eq. (12)
    Assign $p^r(y) = \bar{p}(y)$ for all $y \in Y'$
    $Y = Y \setminus Y'$
  end while
  return $D_{KL}(p^r \,||\, \hat{p})$
```

applied to all classes, leading to the distribution $p \in \mathcal{Q}_\pi$ with minimal D_{KL} to $\hat{p}$. Hence, we can state the following theorem.

Theorem 1 (Optimality) *Given a credal set $\mathcal{Q}_\pi$ induced by a normalized possibility distribution $\pi : \mathcal{Y} \longrightarrow [0,1]$ with $\max_{y \in \mathcal{Y}} \pi(y) = 1$ according to (1), Algorithm 1 returns the solution of $\mathcal{L}^*(\mathcal{Q}_\pi, \hat{p})$ as defined in (11) for an arbitrary distribution $\hat{p} \in \mathbb{P}(\mathcal{Y})$.*

Due to space limitations, we provide the proof of Theorem 1 in the appendix. We further provide a discussion of the computational complexity of Algorithm 1 therein, showing that the worst case complexity is cubic in the number of labels — the average case complexity is much lower, however, making the method amenable to large data sets as demonstrated in our experimental evaluation.

5. Experiments

To demonstrate the effectiveness of our method, we present empirical results for the domain of image classification as an important and practically relevant application. We refer to the appendix for additional results, including a study on knowledge graphs to demonstrate generalizability of our approach, as well as a more comprehensive overview over experimental settings for reproducibility.

5.1. Experimental Setting

Following previous semi-supervised learning evaluation protocols (Sohn et al., 2020; Lienen and Hüllermeier, 2021), we performed experiments on CIFAR-10/-100 (Krizhevsky and Hinton, 2009), SVHN (Netzer et al., 2011) (without the extra data split) and STL-10 (Coates et al., 2011) with various numbers of sub-selected labels. To this end, we trained Wide ResNet-28-2 (Zagoruyko and Komodakis, 2016) models for CIFAR-10, SVHN and

STL-10, whereas we considered Wide ResNet-28-8 as architecture for CIFAR-100. For each combination, we conducted a Bayesian optimization to tune the hyperparameters with 20 runs each on a separate validation split, while we report the final test performances per model trained with the tuned parameters. Each model was trained for 2^{18} iterations. We repeated each experiment for 3 different seeds, whereby we re-used the best hyperparameters for all seeds due to the high computational complexity.

For CCSSL, we distinguish two variants employing either the non-conformity measure (2) or (3), which we refer to as *CCSSL-diff* and *CCSSL-prop* respectively. As baselines, we consider FixMatch (Sohn et al., 2020) and its distribution alignment version as hard and soft probabilistic pseudo-labeling technique, respectively. Moreover, we compare our method to UDA (Xie et al., 2020a) as another soft variant. Recently, FlexMatch (Zhang et al., 2021) has been proposed as an advancement of FixMatch by adding curriculum learning encompassing uncertainty-awareness. Finally, we report results of CSSL (Lienen and Hüllermeier, 2021) as methodically closest related work to our approach. In all cases, we employ RandAugment as strong augmentation policy to realize consistency regularization. As all of the mentioned approaches were embedded in the basic FixMatch framework, we achieve a fair comparison alleviating side-effects.

5.2. Generalization Performance

Table 1: Averaged accuracies over 3 seeds for different numbers of labels. **Bold** entries indicate the best performing method per column.

	CIFAR-10			CIFAR-100			SVHN			STL-10
	40 lab.	250 lab.	4000 lab.	400 lab.	2500 lab.	10000 lab.	40 lab.	250 lab.	1000 lab.	1000 lab.
UDA	86.44 ±2.70	**94.81** ±0.22	95.31 ±0.10	49.41 ±2.96	68.33 ±0.31	75.98 ±0.45	84.82 ±10.6	96.41 ±1.04	97.14 ±0.24	83.94 ±1.49
FixMatch	87.14 ±2.61	93.81 ±1.02	95.08 ±0.18	47.73 ±1.88	66.82 ±0.26	76.66 ±0.18	86.26 ±14.2	96.23 ±1.50	**97.32** ±0.09	85.34 ±0.92
FixMatch DA	89.54 ±5.90	94.00 ±0.56	95.13 ±0.21	51.31 ±2.67	69.98 ±0.30	76.67 ±0.08	86.00 ±16.3	95.85 ±1.62	97.02 ±0.16	85.59 ±1.21
FlexMatch	91.21 ±3.46	94.08 ±0.64	94.62 ±0.27	49.99 ±0.39	**71.47** ±1.06	77.01 ±0.17	85.78 ±1.37	96.52 ±0.29	96.54 ±0.30	85.24 ±1.49
CSSL	**91.70** ±4.77	94.59 ±0.15	95.41 ±0.04	52.54 ±1.60	67.81 ±0.64	77.56 ±0.22	**87.27** ±5.69	95.54 ±1.63	96.69 ±0.75	85.07 ±1.11
CCSSL-diff	90.13 ±3.33	93.56 ±0.21	95.43 ±0.06	**54.13** ±1.97	69.33 ±0.78	77.40 ±0.18	86.52 ±6.84	95.67 ±1.51	96.97 ±0.23	85.26 ±0.79
CCSSL-prop	89.24 ±2.56	94.34 ±0.27	**95.48** ±0.06	53.48 ±2.75	67.90 ±0.37	**77.72** ±0.08	85.38 ±7.67	**96.87** ±0.20	97.04 ±0.38	**85.67** ±1.14

Table 1 shows the generalization performance of all methods with respect to the accuracy for various amounts of labeled data. Note that the calibration set for the conformal credal variant is taken from the labeled instances, i.e., the effective number of instances to learn from is further decreased. In the appendix, we provide a quantification of the network calibration, i.e., the quality of the predicted probability distributions.

As can be seen, CCSSL leads to competitive generalization when a sufficient amount of labeled instances is provided, often even performing best among the compared methods. In the label-scarce settings, the separation of a calibration set (which is taken from the labeled data part) has a stronger effect, and the performance is slightly inferior to CSSL. However, the performance still does not drop, which confirms the effectiveness of the improved pseudo-label quality over CSSL. In most other cases, CCSSL appears to be superior compared to CSSL.

On CIFAR-100 with 400 labels, the learner often faces neighborhoods populated by instances with heterogeneous class distributions. As a result, consistency regularization becomes less effective as the rich class space harms the continuity of regions covering instances of a particular class, leading to the manifestation of misbeliefs in these regions. As consistently observed in the learning curves for CIFAR-100 shown in Fig. 2, CSSL with CR suffers from this issue with increased confidence. In the terminal phase of the training, credal sets are kept rather vague, which is why misbeliefs do not affect the overall performance too much. However, later phases show a performance degradation, which can be attributed to the fact that (potentially mislabeled) credal pseudo-labels become smaller and have thus a higher weight in the overall loss minimization. As opposed to that, CCSSL learns more cautiously thanks to the conformal construction of credal sets and can continuously improve the generalization performance.

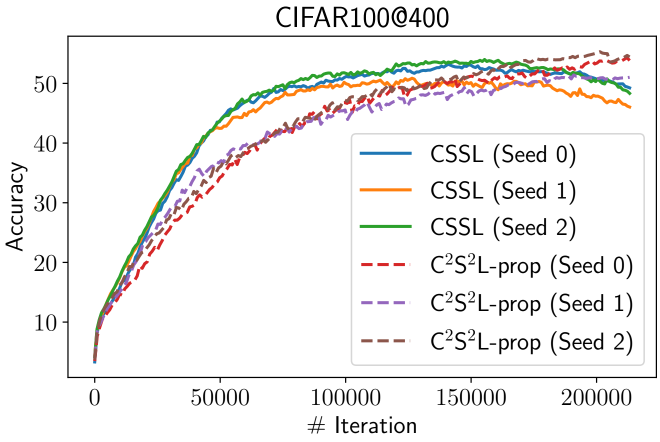


Figure 2: Test accuracies over the course of the training on CIFAR-100 with 400 labels.

5.3. Pseudo-Label Quality

To assess the quality of the (credal) pseudo-labels, we report the validity according to (8) by measuring the error rate in terms of $\mathbb{1}(\pi(y) \leq \delta)$ given the pseudo-supervisions π for the unlabeled training instances with true labels y (for significance levels $\delta \in \{0.05, 0.1, 0.25\}$). To ensure fairness, we compute these scores for trained models so as not to give an advantage to CCSSL, which can in contrast to CSSL rely on the strong validity guarantee throughout the training.

Table 2: The final validity as specified in Eq. (8) of all credal pseudo-labels for different significance levels δ averaged over 3 random seeds. **Bold** entries indicate the best method per column, the standard deviation is a factor of $1e^{-3}$.

	CIFAR-10						SVHN					
	250 lab.			1000 lab.			250 lab.			1000 lab.		
δ	0.05	0.1	0.25	0.05	0.1	0.25	0.05	0.1	0.25	0.05	0.1	0.25
CSSL	0.027 ±0.6	0.033 ±1.1	**0.042** ±0.7	0.033 ±1.1	0.037 ±4.8	0.042 ±1.1	0.038 ±0.6	0.044 ±1.4	0.053 ±0.8	0.030 ±1.6	0.034 ±6.1	0.040 ±2.1
CCSSL-diff	0.026 ±2.9	0.032 ±1.9	0.047 ±4.7	0.029 ±1.5	0.034 ±1.5	0.042 ±0.6	0.028 ±1.0	0.035 ±1.1	0.042 ±1.3	0.024 ±1.2	0.028 ±1.6	0.032 ±1.2
CCSSL-prop	**0.022** ±2.1	**0.031** ±1.5	0.043 ±1.7	**0.024** ±2.7	**0.029** ±3.4	**0.039** ±1.9	**0.014** ±1.1	**0.020** ±1.1	**0.024** ±0.3	**0.021** ±1.1	**0.023** ±2.7	**0.029** ±1.5

In the context of self-training, a lower error rate is desirable to obtain less noisy and more informative self-supervision. As shown in Table 2, the two variants of CCSSL indeed achieve a consistent improvement in the validity of the pseudo-labels over CSSL in terms of the error rate, although the supervision provided by the latter is already quite accurate and does not violate the strong validity property for these instances either. Moreover, CCSSL-prop leads to more accurate sets compared to CCSSL-diff. Note that in standard conformal prediction one is typically interested in error rates that match the respective significance levels. In the setting considered here, this is undermined by relatively small calibration sets $\mathcal{D}_{\text{calib}}$, as well as the fact that the unlabeled instances, whose validity is presented here, have already been observed in previous training iterations, thus leading to optimistic error rates. Nevertheless, this optimism turns out to be beneficial for effective self-training, as our empirical results confirm.

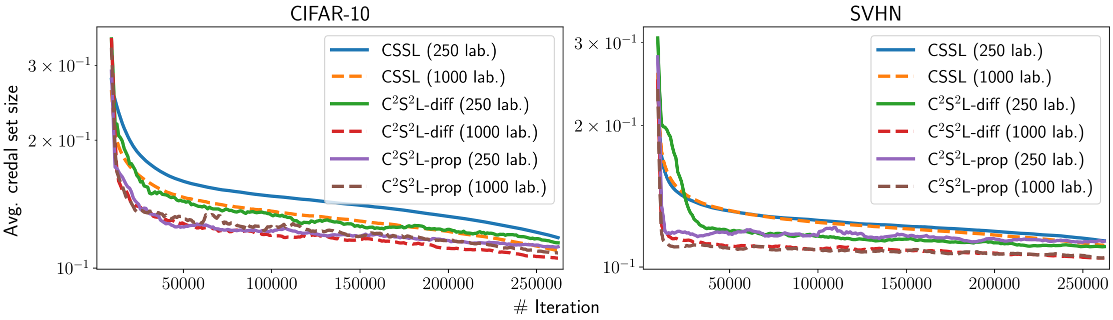


Figure 3: The credal set efficiency in terms of the mean possibilities π for each class, averaged over all pseudo-labels in each iteration for all 3 seeds.

Furthermore, the efficiency of the credal sets, i.e., their sizes, has an effect on the learning behavior. Fig. 3 compares the credal set sizes of CSSL and CCSSL over the course of the training. As can be seen, CCSSL constructs smaller credal sets, which in combination with the improved validity leads to a more effective supervision. Moreover, the credal sets sizes decrease with higher numbers of labels, which is due to a higher prediction quality involved in the credal set construction. Apart from the results on CIFAR-10 with 250 labels, one can further see that the credal set size deviates only slightly between the two non-conformity measures employed in CCSSL.

6. Conclusion

In the context of semi-supervised learning, previous pseudo-labeling approaches lack validity guarantees for the quality of the pseudo-supervision. Such methods often suffer from misleading supervision, leaving much potential unused. In our work, we address these shortcomings by a conformal credal labeling approach leveraging the framework of conformal prediction, which entails validity guarantees for the constructed credal pseudo-labels. Our empirical study confirms the adequacy of this approach when combined with consistency regularization in terms of generalization performance, as well as the calibration and efficiency of pseudo-labels compared to previous methods. At the same time, the combination of a rigorously studied uncertainty quantification framework with pseudo-labeling paves the way for more thorough theoretical analyses in the field of self-training in semi-supervised learning.

In future work, we plan to investigate approaches to achieve *conditional* (per-instance) validity, i.e., conformal credal labels specifically tailored to (conditioned on) the query instance $\boldsymbol{x}_{N+1}$, which would lead to even stronger guarantees for the quality of pseudo-labels. Approximations of this type of validity for conformal prediction have already been provided (Vovk, 2013; Bellotti, 2021). Moreover, a more rigorous analysis of the efficiency of alternative non-conformity scores needs to be performed, including their robustness to label noise as typically present in real-world data.

Acknowledgments

This work was partially supported by the German Research Foundation (DFG) within the Collaborative Research Center "On-The-Fly Computing" (CRC 901 project no. 160364472). Moreover, the authors gratefully acknowledge the funding of this project by computing time provided by the Paderborn Center for Parallel Computing (PC^2).

References

Eric Arazo, Diego Ortego, Paul Albert, Noel E. O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *Proc. of the International Joint Conference on Neural Networks, IJCNN, Glasgow, United Kingdom, July 19-24*, pages 1–8. IEEE, 2020.

Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems, NIPS, Montreal, Quebec, Canada, December 8-13*, pages 3365–3373, 2014.

Sohail Bahmani and Bhiksha Raj. A unifying analysis of projected gradient descent for ℓ_p-constrained least squares. *CoRR*, abs/1107.4623, 2011.

Vineeth Balasubramanian, Shen-Shyang Ho, and Vladimir Vovk. *Conformal Prediction for Reliable Machine Learning: Theory, Adaptations and Applications*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st edition, 2014. ISBN 0123985374.

Anthony Bellotti. Approximation to object conditional validity with inductive conformal predictors. In Lars Carlsson, Zhiyuan Luo, Giovanni Cherubin, and Khuong An Nguyen, editors, *Conformal and Probabilistic Prediction and Applications, virtual, September 8-10*, volume 152 of *Proceedings of Machine Learning Research*, pages 4–23. PMLR, 2021.

David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. MixMatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS, Vancouver, BC, Canada, December 8-14*, pages 5050–5060, 2019.

David Berthelot, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. ReMixMatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *Proc. of the 8th International Conference on Learning Representations, ICLR, Addis Ababa, Ethiopia, April 26-30.* OpenReview.net, 2020.

Vivien Cabannes, Alessandro Rudi, and Francis R. Bach. Structured prediction with partial labelling through the infimum loss. In *Proceedings of the 37th International Conference on Machine Learning, ICML, virtual, July 13-18*, volume 119, pages 1230–1239. PMLR, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 9630–9640. IEEE, 2021.

Leonardo Cella and Ryan Martin. Valid inferential models for prediction in supervised learning problems. In *International Symposium on Imprecise Probability: Theories and Applications, ISIPTA, Granada, Spain, July 6-9*, volume 147 of *Proceedings of Machine Learning Research*, pages 72–82. PMLR, 2021.

Leonardo Cella and Ryan Martin. Validity, consonant plausibility measures, and conformal prediction. *Int. J. Approx. Reason.*, 141:110–130, 2022.

Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. *Semi-Supervised Learning*. The MIT Press, 2006.

Adam Coates, Andrew Y. Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proc. of the 14th International Conference on Artificial Intelligence and Statistics, AISTATS, Fort Lauderdale, FL, USA, April 11-13*, volume 15 of *JMLR Proceedings*, pages 215–223. JMLR.org, 2011.

Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. RandAugment: Practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

M. Delgado and S. Moral. On the concept of possibility-probability consistency. *Fuzzy Sets and Systems*, 21(3):311–318, 1987. ISSN 0165-0114.

Carl Doersch, Abhinav Gupta, and Alexei A. Efros. Unsupervised visual representation learning by context prediction. In *Proc. of the IEEE International Conference on Computer Vision, ICCV, Santiago, Chile, December 7-13*, pages 1422–1430. IEEE Computer Society, 2015.

Jingfei Du, Edouard Grave, Beliz Gunel, Vishrav Chaudhary, Onur Celebi, Michael Auli, Veselin Stoyanov, and Alexis Conneau. Self-training improves pre-training for natural language understanding. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 5408–5418. Association for Computational Linguistics, 2021.

Didier Dubois and Henri Prade. Possibility theory, probability theory and multiple-valued logics: A clarification. *Annals of Mathematics and Artificial Intelligence*, 32:35–66, 2004.

Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3:95–110, 1956.

Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proc. of the 33nd International Conference on Machine Learning, ICML, New York City, NY, USA, June 19-24*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1050–1059. JMLR.org, 2016.

Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V. Le. DropBlock: A regularization method for convolutional networks. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, NeurIPS, Montreal, Quebec, Canada, December 3-8*, pages 10750–10760, 2018.

Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular depth estimation. In *Proc. of the IEEE/CVF International Conference on Computer Vision, ICCV, Seoul, Korea (South), October 27 - November 2*, pages 3827–3837. IEEE, 2019.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

Eyke Hüllermeier and Weiwei Cheng. Superset learning based on generalized loss minimization. In *Proc. of the European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, Porto, Portugal, September 7-11, Proc. Part II*, volume 9285 of *LNCS*, pages 260–275. Springer, 2015.

Eyke Hüllermeier and Willem Waegeman. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Mach. Learn.*, 110(3):457–506, 2021.

Martin Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *Proc. of the 30th International Conference on Machine Learning, ICML, Atlanta, GA, USA, June 16-21*, volume 28 of *JMLR Workshop and Conference Proceedings*, pages 427–435. JMLR.org, 2013.

Ulf Johansson, Ernst Ahlberg, Henrik Boström, Lars Carlsson, Henrik Linusson, and Cecilia Sönströd. Handling small calibration sets in mondrian inductive conformal regressors. In *Proc. of the Statistical Learning and Data Sciences - Third International Symposium, SLDS, Egham, UK, April 20-23*, volume 9047 of *Lecture Notes in Computer Science*, pages 271–280. Springer, 2015.

Byol Kim, Chen Xu, and Rina Foygel Barber. Predictive inference is free with the jackknife+-after-bootstrap. In *Advances in Neural Information Processing Systems 33:*

Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12, 2020.

Kyungyul Kim, Byeongmoon Ji, Doyoung Yoon, and Sangheum Hwang. Self-knowledge distillation with progressive refinement of targets. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 6547–6556. IEEE, 2021.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, Toronto, Canada, 2009.

Tomáš Kroupa. How many extreme points does the set of probabilities dominated by a possibility measure have. In *Proc. of 7th Workshop on Uncertainty Processing WUPES*, volume 6, pages 89–95, 2006.

Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, International Conference on Machine Learning, ICML, Atlanta, GA, USA, June 16-21*, volume 3, 2013.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111, 2018.

Yan Li, Xiaofeng Cao, and Honghui Chen. Fully projection-free proximal stochastic gradient method with optimal convergence rates. *IEEE Access*, 8:165904–165912, 2020.

Julian Lienen and Eyke Hüllermeier. Credal self-supervised learning. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-14*, pages 14370–14382, 2021.

Julian Lienen and Eyke Hüllermeier. From label smoothing to label relaxation. In *Proc. of the 35th AAAI Conference on Artificial Intelligence, virtual, February 2-9*, 2021.

Julian Lienen and Eyke Hüllermeier. Instance weighting through data imprecisiation. *Int. J. Approx. Reason.*, 134:1–14, 2021. ISSN 0888-613X.

Subhabrata Mukherjee and Ahmed Hassan Awadallah. Uncertainty-aware self-training for few-shot text classification. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

Rafael Müller, Simon Kornblith, and Geoffrey E. Hinton. When does label smoothing help? In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS, Vancouver, BC, Canada, December 8-14*, pages 4696–4705, 2019.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *Advances* *in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NIPS, Granada, Spain, November 12-17, Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

Avital Oliver, Augustus Odena, Colin Raffel, Ekin Dogus Cubuk, and Ian J. Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, NeurIPS, Montreal, Quebec, Canada, December 3-8*, pages 3239–3250, 2018.

Harris Papadopoulos. *Inductive conformal prediction: Theory and application to neural networks*. INTECH Open Access Publisher Rijeka, 2008.

Harris Papadopoulos, Volodya Vovk, and Alexander Gammerman. Conformal prediction with neural networks. In *19th IEEE International Conference on Tools with Artificial Intelligence, ICTAI, Patras, Greece, October 29-31, Volume 2*, pages 388–395. IEEE Computer Society, 2007.

Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V. Le. Meta pseudo labels. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 11557–11568. Computer Vision Foundation / IEEE, 2021.

Zhongzheng Ren, Raymond A. Yeh, and Alexander G. Schwing. Not all unlabeled data are equal: Learning to weight data in semi-supervised learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S. Rawat, and Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. In *Proc. of the 9th International Conference on Learning Representations, ICLR, virtual, May 3-7*. OpenReview.net, 2021.

Chuck Rosenberg, Martial Hebert, and Henry Schneiderman. Semi-supervised self-training of object detection models. In *Proc. of the 7th IEEE Workshop on Applications of Computer Vision / IEEE Workshop on Motion and Video Computing, WACV/MOTION, Breckenridge, CO, USA, January 5-7*, pages 29–36. IEEE Computer Society, 2005.

Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems, NIPS, Barcelona, Spain, December 5-10*, pages 1163–1171, 2016.

Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. *J. Mach. Learn. Res.*, 9:371–421, 2008.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the Inception architecture for computer vision. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Las Vegas, NV, USA, June 27-30*, pages 2818–2826. IEEE Computer Society, 2016.

Jafar Tanha, Maarten van Someren, and Hamideh Afsarmanesh. Semi-supervised self-training for decision tree classifiers. *Int. J. Mach. Learn. Cybern.*, 8(1):355–370, 2017. ISSN 1868-808X.

Jesper E. van Engelen and Holger H. Hoos. A survey on semi-supervised learning. *Mach. Learn.*, 109(2):373–440, 2020.

Vladimir Vovk. Conditional validity of inductive conformal predictors. *Mach. Learn.*, 92 (2-3):349–376, 2013.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer-Verlag, Berlin, Heidelberg, 2005. ISBN 0387001522.

Colin Wei, Kendrick Shen, Yining Chen, and Tengyu Ma. Theoretical analysis of self-training with deep networks on unlabeled data. In *9th International Conference on Learning Representations, ICLR, Virtual Event, Austria, May 3-7*. OpenReview.net, 2021.

Qizhe Xie, Zihang Dai, Eduard H. Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020a.

Qizhe Xie, Minh-Thang Luong, Eduard H. Hovy, and Quoc V. Le. Self-training with noisy student improves imagenet classification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10684–10695. Computer Vision Foundation / IEEE, 2020b.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *Proc. of the British Machine Vision Conference, BMVC, York, UK, September 19-22*, 2016.

Gianluca Zeni, Matteo Fontana, and Simone Vantini. Conformal prediction: a unified review of theory and new challenges. *CoRR*, abs/2005.07972, 2020.

Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-14*, pages 18408–18419, 2021.

Zhedong Zheng and Yi Yang. Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. *Int. J. Comput. Vis.*, 129(4):1106–1120, 2021.

7 Mitigating Label Noise through Data Ambiguation

Author Contribution Statement

The idea of leveraging credal labels to deliberately ambiguate for robustness against label noise originates from the author, and was developed by the author. Both authors contributed to the further refinement of this idea. The paper was initially written by the author, and subsequently revised by both authors. Furthermore, the implementation and experimentation was done by the author.

Supplementary Material

An appendix to the paper is provided in Appendix D. The code of the official implementation is provided at `https://github.com/julilien/MitigatingLabelNoiseDataAmbiguation` (Apache 2.0 license).

Mitigating Label Noise through Data Ambiguation

Julian Lienen
Department of Computer Science
Paderborn University
Paderborn 33098, Germany
julian.lienen@upb.de

Eyke Hüllermeier
Institute of Informatics, LMU Munich
Munich Center for Machine Learning
Munich 80538, Germany
eyke@lmu.de

Abstract

Label noise poses an important challenge in machine learning, especially in deep learning, in which large models with high expressive power dominate the field. Models of that kind are prone to memorizing incorrect labels, thereby harming generalization performance. Many methods have been proposed to address this problem, including robust loss functions and more complex label correction approaches. Robust loss functions are appealing due to their simplicity, but typically lack flexibility, while label correction usually adds substantial complexity to the training setup. In this paper, we suggest to address the shortcomings of both methodologies by "ambiguating" the target information, adding additional, complementary candidate labels in case the learner is not sufficiently convinced of the observed training label. More precisely, we leverage the framework of so-called superset learning to construct set-valued targets based on a confidence threshold, which deliver imprecise yet more reliable beliefs about the ground-truth, effectively helping the learner to suppress the memorization effect. In an extensive empirical evaluation, our method demonstrates favorable learning behavior on synthetic and real-world noise, confirming the effectiveness in detecting and correcting erroneous training labels.

1 Introduction

Label noise refers to the presence of incorrect or unreliable annotations in the training data, which can negatively impact the generalization performance of learning methods. Dealing with label noise constitutes a major challenge for the application of machine learning methods to real-world applications, which often exhibit noisy annotations in the form of erroneous labels or distorted sensor values. This is no less a concern for large models in the regime of deep learning as well, which have become increasingly popular due to their high expressive power, and are not immune to the harming nature of label noise. Existing methods addressing this issue include off-the-shelf robust loss functions, e.g., as in [Wang et al., 2019], and label correction, for example by means of replacing labels assumed to be corrupted [Wu et al., 2021]. While the former appeals with its effortless integration into classical supervised learning setups, methods of the latter kind allow for a more effective suppression of noise by remodeling the labels. However, this comes at the cost of an increased model complexity, typically reducing the efficiency of the training [Liu et al., 2020].

The training dynamics of models in the considered regime have been thoroughly studied [Chang et al., 2017, Arazo et al., 2019, Liu et al., 2020], which in fact entail two learning phases that can be distinguished: Initially, as shown in the left plot of Fig. 1, the model shows reasonable learning behavior by establishing correct relations between features and targets, classifying even mislabeled instances mostly correctly. In this phase, the loss minimization is dominated by the clean fraction, such that the mislabeling does not affect the learning too much. However, after the clean labels are

Preprint. Under review.

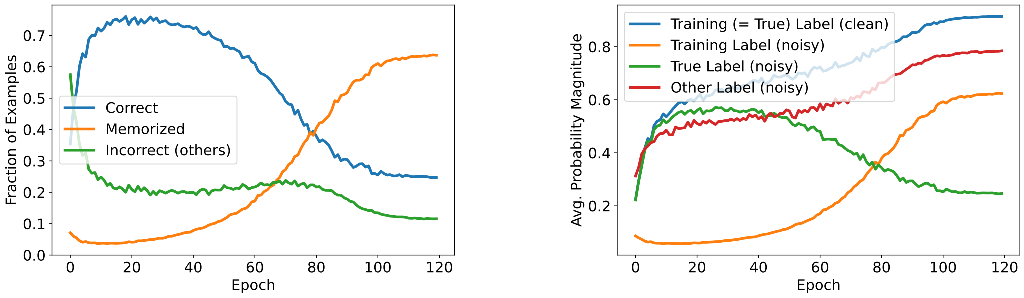


Figure 1: For ResNet34 models trained with cross-entropy on CIFAR-10 with 25 % of corrupted instances (averaged over five seeds), the left plot shows the fractions of examples that are correctly classified, memorized with respect to their corrupted training label, or incorrectly classified with a label other than the ground-truth or training label, confirming the result in [Liu et al., 2020]. The right plot illustrates the predicted probability magnitudes for clean or noisy labels.

sufficiently well fit, the loss minimization starts to concentrate predominantly on the mislabeled part, thereby overfitting mislabels and thus harming generalization.

A closer look at the probabilistic predictions in the first learning phase reveals that the confidence degrees allow for distinguishing between erroneous training labels and the underlying ground-truth classes. The right plot in Fig. 1 illustrates that models are typically not incentivized to optimize for predicting the corrupted training labels in the first epochs, but infer relatively high probability scores for the (unknown) ground-truth class. The learned knowledge of the model at this point could be regarded as a distillation process from the noisy labels – the model is implicitly able to distinguish between clean and noisy instances, and retains only useful information. Evidently, the model *itself* could serve as a source for modeling the beliefs about the ground-truth, taking all (mostly clean) instance-label relations into account to reason about the true labeling. This idea has been adopted by label correction methods that predict *pseudo-labels* for instances that appear to be mislabeled, thus suggesting labels that are considered by the model to be more plausible [Reed et al., 2015, Wu et al., 2021]. Nevertheless, especially in early phases of the training, substituting the original label as a possibly correct information entails the risk of drawing conclusions too hastily, potentially aggravating the negative effects of label noise. Instead, it seems advisable to not completely exclude the possibility of the original training label being the target either to realize a cautious learning behavior, as the learner is still in its infancy and should not discard true labels too early.

Following this motivation, we propose a *complementary* target modeling approach that is guided by the learner's confidence. Specifically, our approach accounts for the potential existence of other plausible labels beyond the observed training label, which may appear plausible in the context of the overall optimization, while not degrading the initial label's plausibility either. To achieve this, we employ so-called *supersets* [Liu and Dietterich, 2014, Hüllermeier and Cheng, 2015] to maintain sets of candidate labels modeling the beliefs about the true outcome. In this course, we deliberately "ambiguate" the targets by adding classes to the superset that exceed a specified confidence threshold. More precisely, we represent the ambiguous target information in the form of so-called *credal sets*, i.e., sets of probability distributions, to train probabilistic classifiers via generalized risk minimization in the spirit of label relaxation [Lienen and Hüllermeier, 2021b]. We realize our approach, which we dub *Robust Data Ambiguation* (RDA), in an easy off-the-shelf loss function that dynamically derives the target sets from the model predictions without the need of any additional parameter – this is implicitly done in the loss calculation, without requiring any change to a conventional learning routine. This way, we combine the simplicity of robust losses with the data modeling capabilities of more complex label correction approaches. We demonstrate the effectiveness of our method on commonly used image classification datasets with both synthetic and real-world noise, confirming the adequacy of our proposed robust loss.

2 Related Work

Coping with label noise in machine learning contexts is a broad field with an extensive amount of literature in the recent times. Here, we distinguish four views on this issue, namely, robust loss functions, regularization, sample selection and label correction methods. For a more comprehensive overview, we refer to [Song et al., 2020, Wei et al., 2022b] as recent surveys.

Robust Losses: The task of designing robust optimization criteria has a long-standing history in classical statistics, e.g., to alleviate the sensitivity towards outliers. As a prominent member of such methods, the mean absolute error (MAE) steps up to mitigate the shortcomings of the mean squared error. When relating to the context of probabilistic classification, analyses link robustness of loss functions towards label noise to the symmetry of the function [Ghosh et al., 2017], leading to suchlike cross-entropy adaptations [Wang et al., 2019, Ma et al., 2020]. A large strain of research proposes to balance MAE and cross-entropy, e.g., by the negative Box-Cox transformation [Zhang and Sabuncu, 2018] or controlling the order of Taylor series for the categorical cross-entropy [Feng et al., 2020], whereas also alternative loss formulations have been considered [Yu et al., 2020, Wei and Liu, 2021]. Besides, methodologies accompanying classical losses for robustness have been proposed, such as gradient-clipping [Menon et al., 2020] or sub-gradient optimization methods [Ma and Fattahi, 2022].

Regularization: Regularizing losses for classification has also been considered as a mean to cope with label noise. As one prominent example, label smoothing [Szegedy et al., 2016] has shown similar beneficial properties as loss correction when dealing with label noise [Lukasik et al., 2020]. Advancements also enable applicability in high noise regimes [Wei et al., 2022a]. Among the first works building upon this observation, [Arpit et al., 2017] characterizes two phases in learning from data with label noise. First, the model learns to correctly classify most of the instances (including the ground-truth labels of misclassified training instances), followed by the memorization of mislabels. The works shows that explicit regularization, for instance Dropout Srivastava et al. [2014], is effective in combating memorization, improving generalization. Building upon this insight, [Liu et al., 2020] proposes a penalty term to counteract memorization that stems from the (more correct) early-learning of the model. In addition, sparse regularization enforces the model to predict sharp distributions with sparsity [Zhou et al., 2021b]. Lastly, [Iscen et al., 2022] describes a regularization term based on the consistency of an instance with its neighborhood in feature space.

Sample Selection: Designed to prevent the aforementioned memorization of mislabeled instances, a wide variety of methods rely on the so-called small loss selection criterion [Jiang et al., 2018, Gui et al., 2021], which is intended to distinguish clean samples the model recognizes well, from which can be learned in an undistorting manner. This distinctions allows to model a range of models, including a gradual increase of the clean sample set with increasing model confidence [Shen and Sanghavi, 2019, Cheng et al., 2021, Wang et al., 2022], co-training [Han et al., 2018, Yu et al., 2019, Wei et al., 2020] or re-weighting instances [Chen et al., 2021]. Furthermore, it makes the complete plethora of classical semi-supervised learning methods amenable to the task of label noise robustness. Here, the non-small loss examples are considered as unlabeled in the first place [Li et al., 2020, Nishi et al., 2021, Zheltonozhskii et al., 2022]. Often, such methodology is also combined with consistency regularization [Bachman et al., 2014], as prominently used in classical semi-supervised learning [Sohn et al., 2020, Liu et al., 2020, Yao et al., 2021, Liu et al., 2022].

Label Correction: Traditionally, correcting label noise has also been considered by learning probabilistic transition matrices [Goldberger and Ben-Reuven, 2017, Patrini et al., 2017, Zhu et al., 2022, Kye et al., 2022], e.g., to re-weight or correct losses. In this regard, it has been considered to learn a model and (re-)model labels based on the model predictions jointly [Tanaka et al., 2018]. Closely related to sample selection as discussed before, [Song et al., 2019] proposes to first select clean samples in a co-teaching method, which then refurbishes noisy samples by correcting their label information. Furthermore, Arazo et al. [2019] suggests to use two-component mixture models for detecting mislabels that are to be corrected, thus serving as a clean sample criterion. Moreover, [Wu et al., 2021, Tu et al., 2023] approaches the problem of label correction by a meta-learning formulation that models a two-stage optimization process of the model and the training labels. In addition, [Liu et al., 2022] describes an approach to learn the individual instances' noise by an additional over-parameterization. Finally, ensembling has also been considered as a source for label correction [Lu and He, 2022].

3 Method

The following section outlines the motivation behind our method RDA to model targets in an ambiguous manner, introduces the theoretical foundation, and elaborates on how this approach realizes robustness against label noise.

3.1 Motivation

Deep neural networks models, often overparameterized with significantly more parameters than required to fit training data points at hand, have become the de-facto standard choice for most practically relevant problems in deep learning. Here, we consider probabilistic models of the form $\hat{p} : \mathcal{X} \longrightarrow \mathbb{P}(\mathcal{Y})$ with $\mathcal{X} \subset \mathbb{R}^d$ and $\mathcal{Y} = \{y_1, \ldots, y_K\}$ being the feature and target space, respectively. We denote the training set of size N as $\mathcal{D}_N = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N \subset \mathcal{X} \times \mathcal{Y}$, where each instance $\boldsymbol{x}_i \in \mathcal{X}$ is associated with an underlying (true) label y_i^*. The latter is conceived as being produced by the underlying (stochastic) data-generating process, i.e., as the realization of a random variable $Y \sim p^*(\cdot \,|\, \boldsymbol{x})$.[1] However, the actual label y_i encountered in the training data might be corrupted, which means that $y_i \neq y_i^*$. Reasons for corruption can be manifold, including erroneous measurement and annotation by a human labeler.

As real-world data typically comprises noise in the annotation process, e.g., due to human labeling errors, their robustness towards label noise is of paramount importance. Alas, models in the regime of deep learning trained with conventional (probabilistic) losses, such as cross-entropy, lack this robustness. As been thoroughly discussed in the analysis of the training dynamics of such models [Chang et al., 2017, Arazo et al., 2019, Liu et al., 2020], two phases can be distinguished in the training of alike models, namely a "correct concept learning" and a *memorization* [Arpit et al., 2017] or *forgetting* [Toneva et al., 2019] phase (cf. Fig. 1). While the model behaves in a desirable manner in the first phase by assimilating reasonable relations between instances and labels, leading to mostly correct predictions on most training instances, the transition to a more and more memorizing model harms the generalization performance.

Looking closer at the learning dynamics in an idealized setting, one can observe that overparameterized models project instances of the same ground-truth class y^* to a similar feature embedding, regardless of having observed a correct or corrupted training label in the first training phase. Fig. 2 illustrates the learned feature activations of the penultimate layer of a multilayer perceptron with a classification head trained on MNIST over the course of the training (see the appendix for further experimental details). In the beginning, the learner predicts a relatively high probability $\widehat{p}(y_i^* \,|\, \boldsymbol{x}_i)$ for noisy instances $(\boldsymbol{x}_i, y_i)$ despite the observed corrupted training label $y_i \neq y_i^*$, as the loss is dominated by the cross-entropy of the clean instances, which leads to a similar marginalization behavior as in the non-corrupted case. Here, the proximity of mislabeled instances to the decision boundary can be related to the "hardness" of instances to be learned, as also been studied in previous work [Chang et al., 2017]. In later stages, the feature activations of the noisy instances shift in a rotating manner, successively being pushed

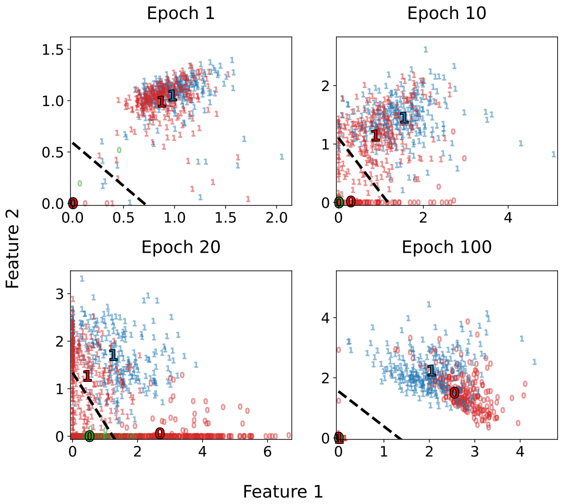


Figure 2: Learned feature representations of the training instances observed at the penultimate layer a MLP comprising an encoder and a classification head at different stages in the training. The data consists of correctly (blue or green resp.) and incorrectly (red) labeled images of zeros and ones from MNIST. The dashed line depicts the linear classifier.

[1] In applications such as image classification, the ground-truth conditional distributions $p^*(\cdot \,|\, \boldsymbol{x}) \in \mathbb{P}(\mathcal{Y})$ are often degenerate, i.e., $p^*(y^* \,|\, \boldsymbol{x}) \approx 1$ and $p^*(y \,|\, \boldsymbol{x}) \approx 0$ for $y \neq y^*$.

towards the other discriminatory face of the decision boundary. This goes hand in hand with a decrease and eventual increase of the predicted probability scores $\max_{y \in \mathcal{Y}} \widehat{p}(y \,|\, \boldsymbol{x})$, consistent with the observations made in Fig. 1.

It appears natural to seek for means that keep corrupted instances $(\cdot, y_i)$ close to clean samples $(\cdot, y_i^*)$ in the feature space, and not let the learning bias in the later stages pull them over towards the label corruption. In the context of the overall optimization, the model itself represents a source for justification whether a class shall be regarded as a plausible outcome or not. Hence, we suggest to use the model predictions *simultaneously* with the training labels as a source for modeling our beliefs about the ground-truth. Consequently, we shall consult not only the individual training labels, but also the *complete* training dataset that found its way into the current model hypothesis in conjunction, complementing the former as a second piece of evidence. This represents a distillation of knowledge obtained from the (mostly clean) data at hand, and we argue that the confidence $\widehat{p}$ is a suitable surrogate to represent plausibility in this regard.

3.2 Credal Labeling

Inquiring the model prediction $\widehat{p}(\boldsymbol{x}_i)$ in addition to the training label y_i may require to augment the (hitherto single) label by an additional plausible candidate label $\hat{y}_i \in \operatorname{argmax}_{y' \in \mathcal{Y}} \widehat{p}(y' \,|\, \boldsymbol{x}_i)$ with $\hat{y}_i \neq y_i$, making the target effectively ambiguous, and hence less convenient for conventional pointwise classification losses. However, from a data modeling perspective, it is important to recognize that the imprecisiation of the now ambiguous target information pays off with higher validity, i.e., it is more likely that the true label is covered by the target labels. This is completely in line with *data imprecisiation* in the context of so-called *superset learning* [Lienen and Hüllermeier, 2021a]. Thus, deliberately ambiguating a corrupt training target information appears desirable for dampening the influence of mislabeling.

Before detailing the representation of the aforementioned beliefs, we shall revisit the conventional probabilistic learning setting. To train probabilistic classifiers $\widehat{p}$ as specified above, such as by minimizing the cross-entropy loss, traditional methods transform deterministic target labels $y_i \in \mathcal{Y}$ as commonly observed in classification datasets into degenerate probability distributions $p_{y_i} \in \mathbb{P}(\mathcal{Y})$ with $p_{y_i}(y_i) = 1$ and $p_{y_i}(y) = 0$ for $y \neq y_i$. As a result, the predicted distribution $\widehat{p}$ can be compared to p_{y_i} using a probabilistic loss $\mathcal{L} : \mathbb{P}(\mathcal{Y}) \times \mathbb{P}(\mathcal{Y}) \longrightarrow \mathbb{R}$ to be optimized in an empirical risk minimization. It is easy to see that full plausibility is assigned to the observed training label y_i, while the other labels are regarded as fully implausible.

Ambiguity in a probabilistic sense can be attained by enriching the target distribution representation in a set-valued way. To this end, Lienen and Hüllermeier [2021b] propose to use *credal sets*, i.e., sets of probability distributions, as a means to express beliefs about the true class conditional distribution p^*. Relating to the introductory motivation, we may want to not only consider p_{y_i} for the training label y_i as a fully plausible target distribution, but also p_y for a plausible candidate label $y \neq y_i$ as well – and interpolations between the two extremes. Credal sets allow one to model such sets of candidate distributions. Moreover, one may also want to consider distributions near the degenerate targets, as it appears unlikely that p^* is any (extreme) degenerate distribution at all. In other words, one may seek to further *relax* the extremity of the targets by considering neighboring distributions around p_{y_i} (and possibly p_y) as candidate target distributions, relieving from commitment to only degenerate distributions [Lienen and Hüllermeier, 2021b].

To derive credal sets of the described form, possibility theory [Dubois and Prade, 2004] offers a suitable framework by means of so-called *possibility distributions* $\pi : \mathcal{Y} \longrightarrow [0, 1]$. It induces a possibility measure Π on $\mathcal{Y}$ by $\Pi(Y) = \max_{y \in Y} \pi(y)$. A distribution π_i associated with an instance $\boldsymbol{x}_i$ assigns a possibility (or plausibility) to each class in $\mathcal{Y}$ for being the true class outcome associated with $\boldsymbol{x}_i$. Thus, $\pi_i(y)$ can be interpreted as an upper probability on $p^*(y \,|\, \boldsymbol{x}_i)$ inducing a convex credal set:

$$Q_{\pi_i} := \Big\{ p \in \mathbb{P}(\mathcal{Y}) \,|\, \forall\, Y \subseteq \mathcal{Y} : \sum_{y \in Y} p(y) \leq \max_{y \in Y} \pi_i(y) \Big\} \tag{1}$$

3.3 Data Ambiguation For Robust Learning

With the theoretical framework to model ambiguous targets in a probabilistic manner, we can put the vision of a robust ambiguation method on a rigorous theoretic foundation. Following the idea of assigning full plausibility to highly confident class predictions in light of the overall optimization context, we elicit the confidence-thresholded possibility distribution π_i for a training instance $(\boldsymbol{x}_i, y_i)$ by

$$\pi_i(y) = \begin{cases} 1 & \text{if } y = y_i \vee \hat{p}(y \,|\, \boldsymbol{x}_i) \geq \beta \\ \alpha & \text{otherwise} \end{cases}, \tag{2}$$

where $\beta \in [0, 1]$ denotes the confidence threshold for the prediction $\widehat{p}(y \,|\, \boldsymbol{x}_i)$, and $\alpha \in [0, 1)$ is the label relaxation parameter [Lienen and Hüllermeier, 2021b]. In fact, the construction of labels by means of Eq. (2) realizes a complementary rather than substitutive pseudo-labeling approach due to retaining full possibility $\pi_i(y_i) = 1$ for the originally observed training label.

To learn from target sets $\mathcal{Q}_{\pi_i}$ employing π_i, we make use of generalized risk minimization through the *optimistic superset loss* [Hüllermeier and Cheng, 2015]. This loss is defined as

$$\mathcal{L}^*(\mathcal{Q}_{\pi_i}, \hat{p}) := \min_{p \in \mathcal{Q}_{\pi_i}} \mathcal{L}(p, \hat{p}) \,, \tag{3}$$

which generalizes $\mathcal{L}$ by a principled data disambiguation of the imprecise targets in light of the entire training data, leading to a minimal empirical risk with respect to $\mathcal{L}$ in the context of the overall loss minimization.

A common choice for $\mathcal{L}$ is the Kullback-Leibler divergence D_{KL}, for which Eq. (3) simplifies to

$$\mathcal{L}^*(\mathcal{Q}_{\pi_i}, \widehat{p}) = \begin{cases} 0 & \text{if } \widehat{p} \in \mathcal{Q}_{\pi_i} \\ D_{KL}(p^r \,||\, \widehat{p}) & \text{otherwise} \end{cases}, \tag{4}$$

where

$$p^r(y) = \begin{cases} (1-\alpha) \cdot \frac{\widehat{p}(y)}{\sum_{y' \in \mathcal{Y}: \pi_i(y') = 1} \widehat{p}(y')} & \text{if } \pi_i(y) = 1 \\ \alpha \cdot \frac{\widehat{p}(y)}{\sum_{y' \in \mathcal{Y}: \pi_i(y') = \alpha} \widehat{p}(y')} & \text{otherwise} \end{cases}$$

is projecting $\widehat{p}$ onto the boundary of $\mathcal{Q}_{\pi_i}$. This loss has proven to be convex and has the same computational complexity as standard losses like cross-entropy Lienen and Hüllermeier [2021b]. In the appendix, we provide a comprehensive algorithmic description of the loss function, confirming the simplicity of our proposal.

Fig. 3 illustrates the core idea of our method, which we will refer to as *Robust Data Ambiguation* (RDA). For a given instance $(\boldsymbol{x}, y_1)$, with y_1 being corrupted from the ground-truth y_2, we initially observe a probabilistic target centered at p_{y_1} (and potentially being relaxed by some $\alpha > 0$). Without changing the target, label relaxation would compute the loss by comparing the model prediction $\widehat{p}$ to a (precise) distribution p^r projecting $\widehat{p}$ towards p_{y_1}. In this example, the model predicts y_2 with high confidence, thereby exceeding β. With our method, full plausibility would be assigned to y_2 in addition to y_1, leading to a credal set $\mathcal{Q}$ as shown in the right plot. To compute the loss, $\widehat{p}$ is now compared to a less extreme target p^r that is projected onto the larger credal set, not urging the learner to predict distributions close to p_{y_1}. Consequently, by ambiguating the target set, we relieve the learner from memorizing wrong training labels, but, at the same time, still staying cautious in committing to potentially erroneous model predictions. This de-emphasizes the learning from wrongly labeled training instances, while not affecting the optimization on clean samples in an undesired manner. At the same time, the imprecisiation allows the model predictions $\widehat{p} \in \mathcal{Q}$ to evolve towards p_{y_2} driven by the loss minimization of similar instances that are correctly labeled with y_2.

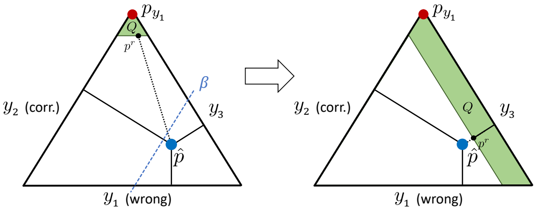


Figure 3: A barycentric visualization of the confidence-thresholded ambiguation for a corrupt training label y_1 and a ground-truth y_2 in the target space $\mathcal{Y} = \{y_1, y_2, y_3\}$: Starting from a credal set $\mathcal{Q}$ centered at p_{y_1} (left plot), the prediction $\widehat{p}$ predicts a probability mass greater than β for y_2. Consequently, full possibility is assigned to y_2, leading to $\mathcal{Q}$ as shown to the right.

The parameter β can be seen as a reflector of the trust spent on the model prediction $\widehat{p}$ itself to reason about the plausibility of a class $y \neq y_i$ being the true outcome y^*. High values for β indicate that only highly confident predictions shall adjust the target, whereas less extreme values result in a less cautious model reasoning. Conversely, this also means that classes y with $\widehat{p}(y) < \beta$ are considered as completely implausible. Hence, small values for β could be also interpreted as exclusion criteria for classes. As shown in a more extensive analysis of the β parameter, the former interpretation is practically more useful in the case of robust classification.

Generally, β could be arbitrarily chosen in $[0, 1]$, also adopted to the training progress. As a simple yet effective rule of thumb, we suggest to model β in a decreasing manner: While the model is relatively uncertain in the first epochs, one should not spent to much attention to the model predictions themselves. However, with further progress, $\widehat{p}$ becomes more informative for the confidence-driven possibility elicitation, suggesting to use smaller β values. We found empirically the cosine decaying definition as

$$\beta_T = \beta_1 + \frac{1}{2}(\beta_0 - \beta_1)\left(1 + \cos\left(\frac{T}{T_{max}}\pi\right)\right) \quad , \tag{5}$$

where T and T_{max} denote the current and maximum number of epochs, as well as β_0 and β_1 represent the starting and ending β respectively. Nevertheless, as will be shown in the empirical evaluation, also static values for β work reasonably well, such that the number of additional hyperparameters can be reduced to a single one.

In summary, our robust loss formulation RDA captivates with no adjustment to the training procedure – all computations can be encapsulated in the loss formulation itself with only accessing the model output to derive the possibility distributions –, but providing expressive (on-the-fly) target modeling as offered in (much more complex) label correction methods. This way, it constitutes an easy and, as will be shown in the experiments, effective way to robustify the learning of large models for (probablistic) classification.

4 Experiments

To demonstrate the effectiveness of our method to cope with label noise, we conduct an empirical analysis on a variety of image classification datasets as a practically relevant problem domain. Needless to say, RDA is not specifically tailored to this domain, as it completely refrains from using modality-specific components. Here, we consider CIFAR-10 and -100 [Krizhevsky and Hinton, 2009], as well as the large-scale datasets WebVision [Li et al., 2017] and Clothing1M [Xiao et al., 2015] as benchmark datasets. For the former, we model synthetic noise by both symmetrically and asymmetrically randomizing the labels for a fraction of examples, while WebVision and Clothing1M both comprise real-world noise by their underlying crawling process. Moreover, we report results on CIFAR-10(0)N [Wei et al., 2022b] as another real-world noise datasets based on human annotators. We refer to the appendix for a more detailed description of the datasets and the corruption process, as well as experiments on additional benchmark datasets.

As baselines, we take a wide range of commonly applied loss functions into account. Proceeding from the conventional cross-entropy (CE), we report results for the regularized CE adaptations label smoothing (LS) and label relaxation (LR), as well as the popular robust loss functions generalized cross-entropy (GCE) [Zhang and Sabuncu, 2018], normalized cross-entropy (NCE) [Ma et al., 2020], combinations of NCE with AUL and AGCE [Zhou et al., 2021a] and CORES Cheng et al. [2021]. The mentioned loss functions share that they do not assume any additional model parameters, e.g., to track the trajectories of model predictions. For completeness, we shall also report results violating this assumptions, namely ELR [Liu et al., 2020] and SOP [Liu et al., 2022] as two state-of-the-art representatives for regularization and label correction methods, respectively, albeit constituting an unfair comparison to our off-the-shelf loss. In the appendix, we show further results for a natural baseline to our approach in the realm of superset learning, as well as experiments that combine our method with sample selection.

We follow common practice in evaluating methods against label noise in the regime of overparameterized models by training ResNet34 models on the smaller scale benchmark datasets CIFAR-10/-100. For the larger datasets, we consider ResNet50 models pretrained on ImageNet. All trained models use the same training procedure and optimizer. A more thorough overview over all experimental details, such as hyperparameters and the technical infrastructure, can be found in the appendix. We repeated each run five times with different random seeds, reporting the accuracy on the test splits to measure generalization performance.

Table 1: Test accuracies and standard deviations on the test split for models trained on CIFAR-10(0) with synthetic noise (symmetric or asymmetric). The results are averaged over runs with different seeds, **bold** entries mark the best method without any additional model parameters. Underlined results indicate the best method overall.

Loss	Add. Param.	CIFAR-10				CIFAR-100			
		Sym.			Asym.	Sym.			Asym.
		25 %	50 %	75 %	40 %	25 %	50 %	75 %	40 %
CE	✗	79.05 ±0.67	55.03 ±1.02	30.03 ±0.74	77.90 ±0.31	58.27 ±0.36	37.16 ±0.46	13.66 ±0.45	62.83 ±0.25
LS ($\alpha = 0.1$)	✗	76.66 ±0.69	53.95 ±1.47	29.03 ±1.21	78.07 ±0.69	59.75 ±0.24	37.61 ±0.61	13.53 ±0.51	63.76 ±0.51
LS ($\alpha = 0.25$)	✗	77.48 ±0.32	53.08 ±1.95	28.29 ±0.65	77.35 ±0.76	59.84 ±0.57	39.80 ±0.38	14.18 ±0.44	63.33 ±0.25
LR ($\alpha = 0.1$)	✗	80.53 ±0.39	57.55 ±0.95	29.83 ±0.87	77.83 ±0.37	57.52 ±0.58	36.77 ±0.54	13.23 ±0.14	62.46 ±0.15
LR ($\alpha = 0.25$)	✗	80.43 ±0.09	60.18 ±1.01	31.36 ±0.91	78.35 ±0.72	57.67 ±0.11	37.15 ±0.14	13.41 ±0.24	62.85 ±0.53
GCE	✗	90.82 ±0.10	83.36 ±0.65	54.34 ±0.37	77.37 ±0.94	68.06 ±0.31	58.66 ±0.28	**26.85** ±1.28	61.08 ±0.51
NCE	✗	79.05 ±0.12	63.94 ±1.74	38.23 ±2.63	76.84 ±0.41	19.32 ±0.81	11.09 ±1.03	6.12 ±7.57	24.67 ±0.71
NCE+AGCE	✗	87.57 ±0.10	83.05 ±0.81	51.16 ±6.44	69.75 ±2.33	64.15 ±0.23	39.64 ±1.66	7.67 ±1.25	53.87 ±1.60
NCE+AUL	✗	88.89 ±0.29	84.18 ±0.42	**65.98** ±1.56	80.87 ±0.34	69.76 ±0.31	57.41 ±0.41	17.72 ±1.27	61.33 ±0.55
CORES	✗	88.60 ±0.28	82.44 ±0.29	47.32 ±17.03	82.22 ±0.55	60.36 ±0.67	46.01 ±0.44	18.23 ±0.28	65.06 ±0.41
RDA (ours)	✗	**91.48** ±0.22	**86.47** ±0.42	48.11 ±15.41	**85.95** ±0.40	**70.03** ±0.32	**59.83** ±1.15	26.75 ±8.83	**69.62** ±0.54
ELR	✓	92.45 ±0.08	88.39 ±0.36	72.58 ±1.63	82.18 ±0.42	73.66 ±1.87	48.72 ±26.93	38.35 ±10.26	74.19 ±0.23
SOP	✓	92.58 ±0.08	89.21 ±0.33	76.16 ±4.88	84.61 ±0.97	72.04 ±0.67	64.28 ±1.44	40.59 ±1.62	64.27 ±0.34

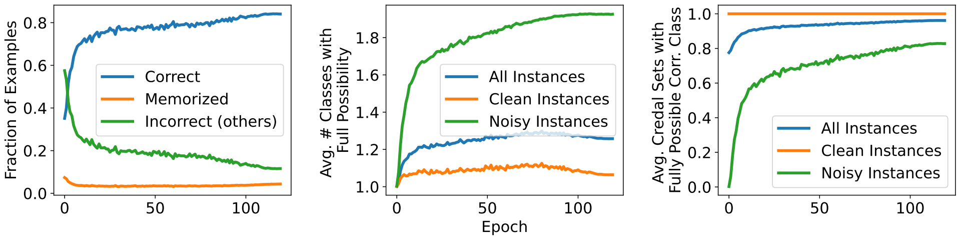


Figure 4: The top plot shows the fraction of mislabeled training instances for which the models predict the ground-truth label (blue), the wrong training label (orange) or a different label (green). The middle and bottom plots show the credal set size and validity respectively. All plots are averaged over the five models trained on CIFAR-10 with 50 % synthetic symmetric noise.

4.1 Synthetic Noise

Table 1 reports the results for the synthetic corruptions on CIFAR-10/-100. As can be seen, our approach provides consistent improvements in terms of generalization performance over the robust off-the-shelf loss functions. Cross-entropy and its regularized adaptations LS and LR appear sensitive to label noise, which confirms the need for robustness on the loss level. Interestingly, although being slightly inferior in most symmetric noise cases, our method proposal appears still competitive compared to ELR and SOP despite their increased expressivity through additional parameters to track the label noise per instance. For asymmetric noise, our method could even outperforms such methods. Nevertheless, our method gets less effective with higher amounts of noise.

When looking at the learning dynamics in training with our robust loss proposal, Fig. 4 reveals an effective ambiguation in the course of the learning process. The left plot shows the effective attenuation of any memorization while improving the correctness of model predictions for the wrongly labeled instances at the same time. The plot in the middle depicts an increase of the credal set size in terms of classes with full plausibility for the noisy instances. Together with the right plot showing the validity of the credal sets, i.e., the fraction of credal sets that assign full possibility to the ground-truth class, one can easily see that the ambiguation is indeed able to select the ground-truth class as training label. Furthermore, the credal set size for clean instances is barely affected, supporting the adequacy of our model. Notably, our method also shows self-correcting behavior after ambiguating with a wrong class midway. While the validity of the credal sets increases (roughly) monotonically, the credal set sizes become smaller towards the end of the training, supporting this claim. In the appendix, we provide additional plots in other noise settings showing consistent effects.

Table 2: Test accuracies and standard deviations on the test split for models trained on CIFAR-10(0)N without any noise (clean) and real-world noise. The results are averaged over runs with different seeds, **bold** entries mark the best method without any additional model parameters. Underlined results indicate the best method overall.

Loss	Add. Param.	CIFAR-10N						CIFAR-100N	
		Clean	Random 1	Random 2	Random 3	Aggregate	Worst	Clean	Noisy
CE	✗	**94.12** ±0.17	82.96 ±0.23	83.16 ±0.52	83.49 ±0.34	88.74 ±0.13	64.93 ±0.79	75.29 ±0.15	52.88 ±0.14
LS ($\alpha = 0.1$)	✗	93.92 ±0.03	82.76 ±0.47	82.10 ±0.21	82.12 ±0.37	88.63 ±0.11	63.10 ±0.38	75.71 ±0.19	53.48 ±0.45
LS ($\alpha = 0.25$)	✗	93.71 ±0.40	82.95 ±1.57	83.86 ±2.05	82.61 ±0.25	87.03 ±2.29	66.14 ±6.89	75.69 ±0.17	53.98 ±0.27
LR ($\alpha = 0.1$)	✗	93.77 ±0.06	83.00 ±0.36	82.64 ±0.31	82.82 ±0.21	88.41 ±0.29	66.62 ±0.33	74.79 ±0.16	52.01 ±0.04
LR ($\alpha = 0.25$)	✗	93.63 ±0.06	82.14 ±0.49	81.87 ±0.34	82.46 ±0.11	88.07 ±0.45	66.44 ±0.14	74.51 ±0.17	52.22 ±0.29
GCE	✗	93.22 ±0.08	88.85 ±0.19	88.96 ±0.32	88.73 ±0.11	90.85 ±0.32	77.24 ±0.47	72.29 ±0.19	55.43 ±0.47
NCE	✗	87.67 ±0.25	81.88 ±0.27	81.02 ±0.32	81.48 ±0.13	84.62 ±0.49	69.40 ±0.10	32.31 ±0.31	21.12 ±0.67
NCE+AGCE	✗	92.56 ±0.07	89.48 ±0.28	88.95 ±0.10	89.25 ±0.29	90.65 ±0.44	81.27 ±0.44	72.00 ±0.09	51.42 ±0.65
NCE+AUL	✗	93.09 ±0.10	89.42 ±0.22	89.36 ±0.15	88.94 ±0.55	90.92 ±0.19	81.28 ±0.47	74.18 ±0.21	56.58 ±0.41
CORES	✗	93.09 ±0.08	86.09 ±0.57	86.48 ±0.27	86.02 ±0.22	89.23 ±0.10	76.80 ±0.96	73.70 ±0.17	53.04 ±0.29
RDA (ours)	✗	94.09 ±0.19	**90.43** ±0.03	**90.09** ±0.29	**90.40** ±0.01	**91.71** ±0.38	**82.91** ±0.83	**76.21** ±0.64	**59.22** ±0.26
ELR	✓	94.21 ±0.11	91.35 ±0.29	91.46 ±0.29	91.39 ±0.03	92.68 ±0.03	84.82 ±0.42	76.66 ±0.11	62.80 ±0.27
SOP	✓	92.84 ±0.20	89.16 ±0.40	89.02 ±0.33	88.99 ±0.31	90.54 ±0.16	80.65 ±0.13	76.12 ±0.32	59.32 ±0.41

Table 3: Large-scale test accuracies on WebVision and Clothing1M using ResNet50 models. The baseline results are for WebVision taken from [Zhou et al., 2021a], whereas the reported accuracies on Clothing1M were taken from [Liu et al., 2020].

Loss	WebVision	Clothing1M
CE	66.96	68.04
GCE	61.76	69.75
AGCE	69.4	-
NCE+AGCE	67.12	-
RDA (ours)	**70.23**	**71.42**

4.2 Real-World Noise

Consistent to the previous observations, our robust ambiguation loss also work reasonably well for real-world noise. As presented in Table 2, RDA leads to superior generalization performance compared to baselines losses without any additional model parameters in almost any case. Moreover, it also consistently outperforms SOP on CIFAR-10N, whereas it leads to similar results for CIFAR-100N.

For the large-scale datasets WebVision and Clothing1M, whose results are presented in Table 3, one can observe that the differences between the baselines and our approach appear rather subtle, but still in favor of our method proposal.

5 Conclusion

Large models are typically prone to memorizing noisy labels in classification tasks. To address this issue, various loss functions have been proposed that enhance the robustness of conventional loss functions against label noise. Although such techniques are appealing due to their simplicity, they typically lack the capacity to incorporate additional knowledge regarding the instances, such as beliefs about the true label. In response, pseudo-labeling methods for label correction have emerged to provide more sophisticated target modeling, albeit at the cost of increased model training complexity.

In our work, we address the shortcomings of previous methods by a simple off-the-shelf loss function that takes the confidence in the model into account to deliberately ambiguate the targets. For labels that appear plausible in light of the overall model training, which we derive from the model's confidence, this information is used to construct a set-valued target set to represent the beliefs about the true outcome in a complementary, more faithful manner. Our empirical evaluation confirms the adequacy of our proposal.

Our approach poses several interesting future research directions. Among those, a more informed determination of β could be realized by a quantification of epistemic uncertainty [Hüllermeier and Waegeman, 2021] of individual model prediction, as highly uncertain guesses should be considered rather cautiously. Also, our method proposal could also be leveraged for various downstream tasks, e.g., to detect anomalies in mostly homogeneous data. Finally, considering learning trajectories could provide further information to reason about the ground-truth to improve over method, as we are currently only considering the model prediction at a time.

References

E. Arazo, D. Ortego, P. Albert, N. E. O'Connor, and K. McGuinness. Unsupervised label noise modeling and loss correction. In *Proc. of the 36th International Conference on Machine Learning, ICML, June 9-15, Long Beach, California, USA*, volume 97 of *Proc. of Machine Learning Research*, pages 312–321. PMLR, 2019.

D. Arpit, S. Jastrzebski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. C. Courville, Y. Bengio, and S. Lacoste-Julien. A closer look at memorization in deep networks. In *Proc. of the 34th International Conference on Machine Learning, ICML, August 6-11, Sydney, NSW, Australia*, volume 70 of *Proc. of Machine Learning Research*, pages 233–242. PMLR, 2017.

P. Bachman, O. Alsharif, and D. Precup. Learning with pseudo-ensembles. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems, NIPS, December 8-13, Montreal, Quebec, Canada*, pages 3365–3373, 2014.

H. Chang, E. G. Learned-Miller, and A. McCallum. Active bias: Training more accurate neural networks by emphasizing high variance samples. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, December 4-9, Long Beach, CA, USA*, pages 1002–1012, 2017.

W. Chen, C. Zhu, and Y. Chen. Sample prior guided robust model learning to suppress noisy labels. *CoRR*, abs/2112.01197, 2021.

H. Cheng, Z. Zhu, X. Li, Y. Gong, X. Sun, and Y. Liu. Learning with instance-dependent label noise: A sample sieve approach. In *9th International Conference on Learning Representations, ICLR, May 3-7, Virtual Event, Austria*. OpenReview.net, 2021.

D. Dubois and H. Prade. Possibility theory, probability theory and multiple-valued logics: A clarification. *Annals of Mathematics and Artificial Intelligence*, 32:35–66, 2004.

L. Feng, S. Shu, Z. Lin, F. Lv, L. Li, and B. An. Can cross entropy loss be robust to label noise? In *Proc. of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI*, pages 2206–2212. ijcai.org, 2020.

A. Ghosh, H. Kumar, and P. S. Sastry. Robust loss functions under label noise for deep neural networks. In *Proc. of the 31st AAAI Conference on Artificial Intelligence, February 4-9, San Francisco, California, USA*, pages 1919–1925. AAAI Press, 2017.

J. Goldberger and E. Ben-Reuven. Training deep neural-networks using a noise adaptation layer. In *5th International Conference on Learning Representations, ICLR, April 24-26, Toulon, France, Conference Track Proc.* OpenReview.net, 2017.

X. Gui, W. Wang, and Z. Tian. Towards understanding deep learning from noisy labels with small-loss criterion. In *Proc. of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI, August 19-27, Virtual Event / Montreal, Canada*, pages 2469–2475. ijcai.org, 2021.

B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. W. Tsang, and M. Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, NeurIPS, December 3-8, Montréal, Canada*, pages 8536–8546, 2018.

E. Hüllermeier and W. Cheng. Superset learning based on generalized loss minimization. In *Proc. of the European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, September 7-11, Porto, Portugal, Proc. Part II*, volume 9285 of *LNCS*, pages 260–275. Springer, 2015.

E. Hüllermeier and W. Waegeman. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Mach. Learn.*, 110(3):457–506, 2021.

A. Iscen, J. Valmadre, A. Arnab, and C. Schmid. Learning with neighbor consistency for noisy labels. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 18-24, New Orleans, LA, USA*, pages 4662–4671. IEEE, 2022.

L. Jiang, Z. Zhou, T. Leung, L. Li, and L. Fei-Fei. MentorNet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *Proc. of the 35th International Conference on Machine Learning, ICML, July 10-15, Stockholmsmässan, Stockholm, Sweden*, volume 80 of *Proc. of Machine Learning Research*, pages 2309–2318. PMLR, 2018.

A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, Toronto, Canada, 2009.

S. M. Kye, K. Choi, J. Yi, and B. Chang. Learning with noisy labels by efficient transition matrix estimation to combat label miscorrection. In *17th European Conference on Computer Vision ECCV, Tel Aviv, October 23-27, Israel, Proc. Part XXV*, volume 13685 of *Lecture Notes in Computer Science*, pages 717–738. Springer, 2022.

J. Li, R. Socher, and S. C. H. Hoi. DivideMix: Learning with noisy labels as semi-supervised learning. In *8th International Conference on Learning Representations, ICLR, April 26-30, Addis Ababa, Ethiopia*. OpenReview.net, 2020.

W. Li, L. Wang, W. Li, E. Agustsson, and L. V. Gool. WebVision database: Visual learning and understanding from web data. *CoRR*, abs/1708.02862, 2017.

J. Lienen and E. Hüllermeier. Instance weighting through data imprecisiation. *Int. J. Approx. Reason.*, 134:1–14, 2021a.

J. Lienen and E. Hüllermeier. From label smoothing to label relaxation. In *Proc. of the 35th AAAI Conference on Artificial Intelligence, February 2-9, Virtual Event*, pages 8583–8591. AAAI Press, 2021b.

L. Liu and T. Dietterich. Learnability of the superset label learning problem. In *Proc. ICML 2014, Int. Conf. on Machine Learning*, Beijing, China, 2014.

S. Liu, J. Niles-Weed, N. Razavian, and C. Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, 2020, Virtual Event*, 2020.

S. Liu, Z. Zhu, Q. Qu, and C. You. Robust training under label noise by over-parameterization. In *Proc. of the 39th International Conference on Machine Learning, ICML, July 17-23, Baltimore, Maryland, USA*, volume 162 of *Proc. of Machine Learning Research*, pages 14153–14172. PMLR, 17–23 Jul 2022.

Y. Lu and W. He. SELC: self-ensemble label correction improves learning with noisy labels. In L. D. Raedt, editor, *Proc. of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI, July 23-29, Vienna, Austria*, pages 3278–3284. ijcai.org, 2022.

M. Lukasik, S. Bhojanapalli, A. K. Menon, and S. Kumar. Does label smoothing mitigate label noise? *ArXiv*, abs/2003.02819, 2020.

J. Ma and S. Fattahi. Blessing of nonconvexity in deep linear models: Depth flattens the optimization landscape around the true solution. *CoRR*, abs/2207.07612, 2022.

X. Ma, H. Huang, Y. Wang, S. Romano, S. M. Erfani, and J. Bailey. Normalized loss functions for deep learning with noisy labels. In *Proc. of the 37th International Conference on Machine Learning, ICML, July 13-18, Virtual Event*, volume 119 of *Proc. of Machine Learning Research*, pages 6543–6553. PMLR, 2020.

A. K. Menon, A. S. Rawat, S. J. Reddi, and S. Kumar. Can gradient clipping mitigate label noise? In *8th International Conference on Learning Representations, ICLR, April 26-30, Addis Ababa, Ethiopia*. OpenReview.net, 2020.

K. Nishi, Y. Ding, A. Rich, and T. Höllerer. Augmentation strategies for learning with noisy labels. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 19-25, Virtual Event*, pages 8022–8031. Computer Vision Foundation / IEEE, 2021.

G. Patrini, A. Rozza, A. K. Menon, R. Nock, and L. Qu. Making deep neural networks robust to label noise: A loss correction approach. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR, July 21-26, Honolulu, HI, USA*, pages 2233–2241. IEEE Computer Society, 2017.

S. E. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan, and A. Rabinovich. Training deep neural networks on noisy labels with bootstrapping. In *3rd International Conference on Learning Representations, ICLR, May 7-9, San Diego, CA, USA, Workshop Track Proc.*, 2015.

Y. Shen and S. Sanghavi. Learning with bad training data via iterative trimmed loss minimization. In *Proc. of the 36th International Conference on Machine Learning, ICML, June 9-15, Long Beach, California, USA*, volume 97 of *Proc. of Machine Learning Research*, pages 5739–5748. PMLR, 2019.

K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. Raffel, E. D. Cubuk, A. Kurakin, and C. Li. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, Virtual Event*, 2020.

H. Song, M. Kim, and J. Lee. SELFIE: refurbishing unclean samples for robust deep learning. In *Proc. of the 36th International Conference on Machine Learning, ICML, June 9-15, Long Beach, California, USA*, volume 97 of *Proc. of Machine Learning Research*, pages 5907–5915. PMLR, 2019.

H. Song, M. Kim, D. Park, and J. Lee. Learning from noisy labels with deep neural networks: A survey. *CoRR*, abs/2007.08199, 2020.

N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, 2014.

C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 27-30, Las Vegas, NV, USA*, pages 2818–2826. IEEE Computer Society, 2016.

D. Tanaka, D. Ikami, T. Yamasaki, and K. Aizawa. Joint optimization framework for learning with noisy labels. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 18-22, Salt Lake City, UT, USA*, pages 5552–5560. Computer Vision Foundation / IEEE Computer Society, 2018.

M. Toneva, A. Sordoni, R. T. des Combes, A. Trischler, Y. Bengio, and G. J. Gordon. An empirical study of example forgetting during deep neural network learning. In *7th International Conference on Learning Representations, ICLR, May 6-9, New Orleans, LA, USA*. OpenReview.net, 2019.

Y. Tu, B. Zhang, Y. Li, L. Liu, J. Li, Y. Wang, C. Wang, and C. Zhao. Learning from noisy labels with decoupled meta label purifier. *CoRR*, abs/2302.06810, 2023.

H. Wang, R. Xiao, Y. Dong, L. Feng, and J. Zhao. ProMix: Combating label noise via maximizing clean sample utility. *CoRR*, abs/2207.10276, 2022.

Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, and J. Bailey. Symmetric cross entropy for robust learning with noisy labels. In *IEEE/CVF International Conference on Computer Vision, ICCV, October 27 - November 2, Seoul, Korea (South)*, pages 322–330. IEEE, 2019.

H. Wei, L. Feng, X. Chen, and B. An. Combating noisy labels by agreement: A joint training method with co-regularization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 13-19, Seattle, WA, USA*, pages 13723–13732. Computer Vision Foundation / IEEE, 2020.

J. Wei and Y. Liu. When optimizing f-divergence is robust with label noise. In *9th International Conference on Learning Representations, ICLR, May 3-7, Virtual Event*. OpenReview.net, 2021.

J. Wei, H. Liu, T. Liu, G. Niu, M. Sugiyama, and Y. Liu. To smooth or not? when label smoothing meets noisy labels. In *Proc. of the 39th International Conference on Machine Learning, ICML, July 17-23, Baltimore, Maryland, USA*, volume 162 of *Proc. of Machine Learning Research*, pages 23589–23614. PMLR, 2022a.

J. Wei, Z. Zhu, H. Cheng, T. Liu, G. Niu, and Y. Liu. Learning with noisy labels revisited: A study using real-world human annotations. In *The Tenth International Conference on Learning Representations, ICLR, April 25-29, Virtual Event*. OpenReview.net, 2022b.

Y. Wu, J. Shu, Q. Xie, Q. Zhao, and D. Meng. Learning to purify noisy labels via meta soft label corrector. In *Proc. of the Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI, February 2-9, Virtual Event*, pages 10388–10396. AAAI Press, 2021.

T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang. Learning from massive noisy labeled data for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 7-12, Boston, MA, USA*, pages 2691–2699. IEEE Computer Society, 2015.

Y. Yao, Z. Sun, C. Zhang, F. Shen, Q. Wu, J. Zhang, and Z. Tang. Jo-SRC: A contrastive approach for combating noisy labels. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 19-25, Virtual Event*, pages 5192–5201. Computer Vision Foundation / IEEE, 2021.

X. Yu, B. Han, J. Yao, G. Niu, I. W. Tsang, and M. Sugiyama. How does disagreement help generalization against label corruption? In *Proc. of the 36th International Conference on Machine Learning, ICML, June 9-15, Long Beach, California, USA*, volume 97 of *Proc. of Machine Learning Research*, pages 7164–7173. PMLR, 2019.

Y. Yu, K. H. R. Chan, C. You, C. Song, and Y. Ma. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, Virtual Event*, 2020.

Z. Zhang and M. R. Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, NeurIPS, December 3-8, Montréal, Canada*, pages 8792–8802, 2018.

E. Zheltonozhskii, C. Baskin, A. Mendelson, A. M. Bronstein, and O. Litany. Contrast to divide: Self-supervised pre-training for learning with noisy labels. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV, January 3-8, Waikoloa, HI, USA*, pages 387–397. IEEE, 2022.

X. Zhou, X. Liu, J. Jiang, X. Gao, and X. Ji. Asymmetric loss functions for learning with noisy labels. In *Proc. of the 38th International Conference on Machine Learning, ICML, Virtual Event, July 18-24*, volume 139 of *Proc. of Machine Learning Research*, pages 12846–12856. PMLR, 2021a.

X. Zhou, X. Liu, C. Wang, D. Zhai, J. Jiang, and X. Ji. Learning with noisy labels via sparse regularization. In *IEEE/CVF International Conference on Computer Vision, ICCV, October 10-17, Montreal, QC, Canada*, pages 72–81. IEEE, 2021b.

Z. Zhu, J. Wang, and Y. Liu. Beyond images: Label noise transition matrix estimation for tasks with lower-quality features. In *Proc. of the 39th International Conference on Machine Learning, ICML, July 17-23, Baltimore, Maryland, USA*, volume 162 of *Proc. of Machine Learning Research*, pages 27633–27653. PMLR, 2022.

8 Instance Weighting through Data Imprecisiation

Author Contribution Statement

The original idea of this work has been presented in the two previous workshop papers [LH15; LH16] by Shenzen Lu and Eyke Hüllermeier. Under Eyke Hüllermeier's guidance, this approach was revisited by the author, who developed an optimization procedure to put the idea into practice, and new experiments were developed and conducted, incl. scenarios that have not been considered as application domains for this idea before. While large parts of the papers were recompiled from previous contributions by Eyke Hüllermeier, the paper was extended by the author and repeatedly revised by both authors. The implementation associated with this work was done by the author.

Supplementary Material

The code of the official implementation is provided at `https://github.com/julilien/InstanceWeightingDataImprecisiation` (Apache 2.0 license).

International Journal of Approximate Reasoning 134 (2021) 1–14

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

Instance weighting through data imprecisiation

Julian Lienen [a,*], Eyke Hüllermeier [b]

[a] Heinz Nixdorf Institute and Department of Computer Science, Paderborn University, Paderborn 33098, Germany
[b] Institute of Informatics, University of Munich (LMU), Munich 80538, Germany

ARTICLE INFO

Article history:
Received 20 December 2020
Received in revised form 6 April 2021
Accepted 7 April 2021
Available online 15 April 2021

Keywords:
Instance weighting
Data imprecisiation
Superset learning
Robustness
Weakly supervised learning
Monocular depth estimation

ABSTRACT

In machine learning, instance weighting is commonly used to control the influence of individual data points in a learning process. The general idea is to improve results (e.g., the accuracy of a predictor) by restricting the influence of training examples that do not appear to be representative and may bias the learner in an undesirable way. The simplest and most common approach is to modulate the influence of each data point through multiplicative scaling. In this paper, we elaborate on the idea of instance weighting through *data imprecisiation* as a viable alternative to existing methods, and formalize this approach within the framework of superset learning. Roughly speaking, the idea is to reduce the influence of training examples by turning a precise data point into an imprecise observation. Within the framework of optimistic superset learning, a generic approach to superset learning, this effectively comes down to modifying an underlying loss function on a per-instance basis. We illustrate our approach for the case of binary classification with support vector machines, showing that it compares favorably with existing approaches to instance weighting in support vector machines. In a further case study, we demonstrate the usefulness of instance weighting through data imprecisiation for the practical problem of depth estimation in monocular images.

1. Introduction

When analyzing data or learning predictive models via a process of inductive inference, there are various ways in which the results eventually produced by the learner can be influenced. Most obviously, the choice of the underlying model class incorporates important prior assumptions, which strongly contribute to the inductive bias implemented by the learning algorithm. Interestingly, however, not only the model can be tweaked, but also the data on which the model is learned. A simple though very important example is the *weighting* of individual training examples, with the goal to control the influence of individual data points on the result of the learning process. In machine learning, this is known as instance weighting [37], and is most commonly used to increase the (relative) influence of training examples that appear to be representative, compared to those that do not — such as noise and outliers, which may bias the learner in a wrong direction. The simplest approach is *scaling*, that is, multiplying the contribution of each data point to some overall objective function (e.g., a loss function in supervised learning) with a numerical weight factor.

In this paper, we advocate a new approach for modulating the influence of individual (potentially noisy) data points, which is quite different from existing techniques. Roughly speaking, the idea is to turn (precise) data points into *imprecise*

[*] Corresponding author.
E-mail addresses: julian.lienen@upb.de (J. Lienen), eyke@ifi.lmu.de (E. Hüllermeier).

https://doi.org/10.1016/j.ijar.2021.04.002

observations, and then to learn from the latter using suitable methods, like those recently developed in the framework of *superset learning* [26]. Admittedly, a "data imprecisiation" of that kind may appear absurd at first sight: Why should one deliberately replace precise data with presumably less informative imprecise data? As will be seen, however, imprecisiation will come with the desirable effect of reducing the influence of training examples, and hence provides a viable alternative to existing methods for instance weighting. Intuitively, the less precise a training example is made, the less the learner is forced to closely fit the original data point.

Technically, we leverage the framework of *optimistic superset learning* [17,18], a generic approach to superset learning, which effectively comes down to modifying the loss function underlying the learning problem on a per-instance basis. Just like scaling, this modification leads to *reducing* the loss, and hence diminishing the influence of a training example. However, the concrete modification, i.e., the way in which the loss is modified, is different from scaling — and, as will be argued, sometimes more appropriate.

We illustrate our approach of instance weighting through data imprecisiation for the special case of binary classification with support vector machines, i.e., for the modification of the hinge loss. More specifically, we compare data imprecisiation with existing methods for instance weighting in two case studies, the first on learning from noisy data, and the second on self-training. Additionally, a further study is conducted on the ill-posed problem of monocular depth estimation, where we incorporate imprecisiation of sensed depth information to weight instances.

2. Related work

Different variants of instance weighting have been used in various contexts, e.g., within the framework of importance sampling in conventional statistics [21]. In the field of machine learning, instance weighting has been considered for different problems, most notably classification and regression [8,12], mainly with the goal to reduce sensitivity towards noise and outliers. More recently, instance weighting mechanisms have also been proposed to increase robustness in deep learning [33].

In the specific case of classification with support vector machines, which will be used for illustrating our approach in this paper, a plethora of adaptations has been proposed to incorporate instance weighting. Wu and Liu [50] train a weighted SVM (WSVM) in an iterative manner with a simple weighting function, while Yang et al. [53] apply a more sophisticated fuzzy clustering approach to determine instance importance. In addition to the incorporation of instance weighting, robust SVM versions using modified loss functions have been suggested to lower the influence of outliers [9,49,52]. Another idea to increase tolerance towards noise is to add perturbations to the input features [43]. Additionally, incorporating the weights in the optimization problem as a parameter to be optimized has been studied by Lapin et al. [23], along with the ability to represent privileged information [46].

Our work mainly deals with *label* noise as opposed to *feature* noise. In this setting, methods to learn from noise levels for each class were introduced [30,40], thereby incorporating instance weighting in an implicit way. Furthermore, Biggio et al. [2] suggest a related approach in an adversarial setting. Moreover, Liu and Tao [27] propose an instance importance re-weighting method based on estimated conditional probabilities of the labels. The work in [28] uses re-weighting based on structural class taxonomies, which addresses the problem of instances belonging to multiple categories by essentially re-weighting the individual one-vs-rest models per class, thereby facilitating noise-tolerance.

Related to the assumption of label noise, the influence of instances can also be controlled by "softening" the labels. In the context of classification, this is referred to as *label smoothing* [42]. In label smoothing, deterministic observations are replaced by probability distributions on the set of all class labels with smoothed (i.e., non-degenerate) probabilities, mainly to lower the risk of training overconfident models [29]. While label smoothing looks similar to imprecisiation at first sight, the former is actually meant to achieve an effect of *regularization* instead of *relaxation*: using cross-entropy as a loss function, the task of the learner is to reproduce less extreme target probabilities as closely as possible. As opposed to this, and as will be discussed in the further course of this work, the relaxation realized by the imprecisiation relieves the learner from reproducing a specific precise target by allowing for disambiguation in a data-driven manner. As demonstrated by Lienen and Hüllermeier [25], this can reduce the bias incorporated in the training data, leading to better calibrated models.

In addition to the techniques outlined above, further ideas for increasing robustness can be found in the literature. For example, Xie et al. [51] randomly flip targets with a fixed probability, resulting in training on different data sets with shared weights. As a result, an averaging (ensembling) effect lowers the risk of overfitting. Motivated by Hinton et al. [15], where target predictions of a complex teacher network are used to train a weaker student using soft targets, Li et al. [24] propose a related *distillery* approach considering noisy side information. Bagherinezhad et al. [1] describe a model iteratively refining label probabilities from previous model predictions in a chain of multiple networks. By refining the labels over all models, data is augmented by soft targets to prevent overfitting.

3. Background on superset learning

Superset learning is a specific type of learning from weak supervision, in which the outcome (response) associated with a training instance is only characterized in terms of a *set* of possible candidates. Thus, superset learning is somehow in-between supervised and semi-supervised learning, with the latter being a special case (in which supersets are singletons for the labeled examples and cover the entire output space for the unlabeled ones). There are numerous applications in which only partial information about outcomes is available [26]. Correspondingly, the superset learning problem has received increasing attention and has been studied by various authors in recent years, albeit under different names [10,13,20,31].

3.1. Setting

We consider a standard setting of supervised learning with an input (instance) space $\mathcal{X}$ and an output space $\mathcal{Y}$. The goal is to learn a mapping from $\mathcal{X}$ to $\mathcal{Y}$ that captures, in one way or the other, the dependence of outcomes (responses) on inputs (predictors). The learning problem essentially consists of choosing an optimal hypothesis h^* from a given hypothesis space $\mathcal{H}$, based on a set of training data

$$\mathcal{D} = \left\{ (\boldsymbol{x}_n, y_n) \right\}_{n=1}^{N} \in (\mathcal{X} \times \mathcal{Y})^N \ . \tag{1}$$

More specifically, optimality typically refers to optimal prediction accuracy, i.e., a hypothesis is sought whose expected prediction loss or *risk*

$$\mathcal{R}(h) = \int L\big(y, h(\boldsymbol{x})\big)\, d\, P(\boldsymbol{x}, y) \tag{2}$$

is minimal; here, $L : \mathcal{Y} \times \mathcal{Y} \longrightarrow \mathbb{R}$ is a loss function, and P is an (unknown) probability measure on $\mathcal{X} \times \mathcal{Y}$ modeling the underlying data generating process.

In superset learning, we are interested in the case where parts of the data are not observed precisely. More specifically, focusing on the outcome values[1] $y_n \in \mathcal{Y}$, we assume that only supersets $Y_n \subseteq \mathcal{Y}$ are provided as training information. Thus, the learning algorithm does not have direct access to the (precise) data (1), but only to the (imprecise, coarse, ambiguous) observations

$$\mathcal{O} = \left\{ (\boldsymbol{x}_n, Y_n) \right\}_{n=1}^{N} \in (\mathcal{X} \times 2^{\mathcal{Y}})^N \ , \tag{3}$$

where Y_n is assumed to cover the underlying true (precise and possibly noisy) data y_n — hence the name “superset learning”. Let us emphasize that, in contrast to imperfections such as noise at the level of the data, supersets Y_n and the uncertainty associated with them are of *epistemic* nature [11]: They do not constitute a property of the data itself but rather represent the available *knowledge* about the data.

In the following, we denote by $\mathbf{Y} = Y_1 \times \cdots \times Y_N$ the (Cartesian) product of the supersets observed for $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$. Moreover, each $\boldsymbol{y} = (y_1, \ldots, y_N) \in \mathbf{Y}$ is called an *instantiation* of the imprecisely observed data. More generally, we call a sample $\mathcal{D}$ in (1) an instantiation of $\mathcal{O}$ if the instances $\boldsymbol{x}_n$ coincide and $y_n \in Y_n$ for all $n \in [N] := \{1, \ldots, N\}$.

3.2. Generalized loss functions

Hüllermeier and Cheng [18] introduce an approach to superset learning that is motivated by the idea of performing model identification and “data disambiguation” simultaneously. This idea is realized by means of a generalized risk minimization approach, using an extended loss function that compares precise predictions with set-valued observations. If $L : \mathcal{Y}^2 \longrightarrow \mathbb{R}_+$ is the original loss, the *optimistic superset loss* (OSL) is defined as follows:

$$\begin{aligned} L^* : 2^{\mathcal{Y}} \times \mathcal{Y} \longrightarrow \mathbb{R}_+ \,, \\ (Y, \hat{y}) \mapsto \min \left\{ L(y, \hat{y}) \mid y \in Y \right\} \end{aligned} \tag{4}$$

More recently, the same loss function has also been proposed (and justified theoretically) under the name *infimum loss* by Cabannes et al. [3]. Hüllermeier [17] further extends this loss to the case where data is more flexibly characterized by fuzzy subsets $\tilde{Y} \in \mathcal{F}(\mathcal{Y})$. Formally, a fuzzy set is specified by a membership function $\mathcal{Y} \longrightarrow [0,1]$, which is a generalization of the characteristic function of a set [22]. For each $y \in \mathcal{Y}$, the value $\tilde{Y}_n(y)$ denotes the degree of membership of the element y in the fuzzy set $\tilde{Y}_n$, where $\tilde{Y}_n(y) = 1$ indicates full membership and $\tilde{Y}_n(y) = 0$ no membership. Following a standard reduction scheme leveraging the level-cut representation of fuzzy sets, the fuzzy-version of the OSL loss (FOSL) is obtained as a generalization of (4) as follows:

$$\begin{aligned} L^{**} : \mathcal{F}(\mathcal{Y}) \times \mathcal{Y} \longrightarrow \mathbb{R}_+ \,, \\ (\tilde{Y}, \hat{y}) \mapsto \int_0^1 L^*\left([\tilde{Y}]_\alpha, \hat{y} \right) d\alpha \,, \end{aligned} \tag{5}$$

where $[\tilde{Y}]_\alpha := \{ y \mid \tilde{Y}(y) \geq \alpha \}$ is the α-cut of $\tilde{Y}$.

[1] The principle of optimistic loss minimization introduced below can also be extended to the case of imprecision in the instance features.

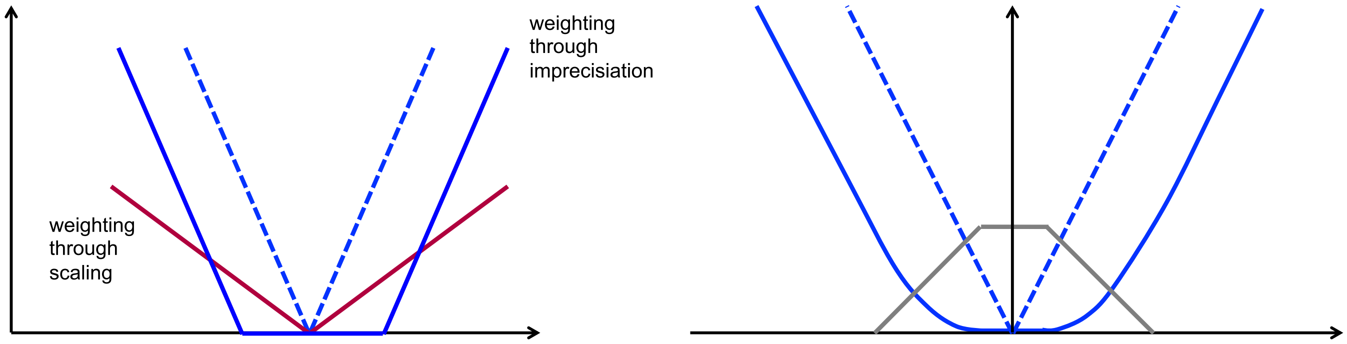


Fig. 1. (Color online.) Left: Different modification of the L_1 loss (dashed line) in regression. In contrast to scaling, (interval-based) data imprecisiation leads to “stretching” the loss function. Right: Example of a loss function $\hat{y} \mapsto L^{**}(\tilde{Y}, \hat{y})$, where $\tilde{Y}$ is the trapezoidal fuzzy set shown in gray, and L is the L_1 loss.

3.3. Generalized loss minimization

With these loss functions at hand, superset learning can be realized as generalized loss minimization, i.e., by replacing the original loss L in the learning algorithm by L^* or L^{**}. In the simplest case, the learner simply implements the empirical risk minimization (ERM) principle by minimizing the generalized loss on the training data. Variants of ERM (using regularization, for example), can be realized in a quite similar way.

While conceptually simple, let us note that the optimization problem to be solved by the learner may become more difficult, simply because L^* and L^{**} may not necessarily inherit all mathematical properties of L — we will come back to this issue in Section 5.1 further below.

4. Data imprecisiation

In addition to learning from genuinely imprecise or ambiguous data, the framework of superset learning can also be used for learning from standard (precise) data, which — via a process of “imprecisiation” — is deliberately turned into imprecise data. As already said, different effects can be achieved in this way. In particular, data imprecisiation offers a means to control the influence of individual observations on the overall result of the learning process: the more imprecise an observation is made, the less it will influence the model induced from the data.

Indeed, the optimistic superset loss L^* (and likewise L^{**}) is a relaxation of the original loss L in the sense that $L^* \leq L$. More specifically, the larger the set Y, the smaller the loss:

$$Y \supseteq Y' \Rightarrow \forall \hat{y} \in \mathcal{Y} : L^*(Y, \hat{y}) \leq L^*(Y', \hat{y})$$

Thus, the loss $L(y, \hat{y})$ incurred for a prediction $\hat{y}$ can be weakened by replacing the original observation y with a (fuzzy) subset around y, and the larger the subset, the smaller the loss. This observation is at the core of our idea of instance weighting within the framework of superset learning.

4.1. Special cases

Interestingly, several existing loss functions (and related machine learning methods) are recovered as special cases of this approach, i.e., for specific combinations of output space $\mathcal{Y}$, loss function L, and imprecisiation of the data. In regression, for example, if L is the L_1 loss and precise outcomes $y_n \in \mathbb{R}$ are replaced by ϵ-intervals $Y_n = [y_n - \epsilon, y_n + \epsilon]$ around the original data points, the ϵ-insensitive loss function used in support vector regression is recovered [36]. In general, relaxing a loss function through imprecisiation leads to “stretching” the original loss, as opposed to the scaling effect achieved by standard weighting (cf. Fig. 1, left). Roughly speaking, this is because the predictor is targeting a set instead of a precise outcome. In the case of interval-data in regression, for example, the generalized loss depends on the distance of the prediction to the target interval.

The FOSL-version of the L_1 loss is even more flexible in this regard. For example, by using a *triangular* fuzzy set $\tilde{Y}$ with membership function $z \mapsto \max\{0, 1 - |y - z|/\delta\}$, we obtain the Huber-loss, a well-known loss function in robust statistics [16]:

$$L^{**}(\tilde{Y}, \hat{y}) = \begin{cases} \frac{1}{2}(y - \hat{y})^2/\delta & \text{if } |y - \hat{y}| \leq \delta \\ |y - \hat{y}| - \frac{1}{2}\delta & \text{if } |y - \hat{y}| > \delta \end{cases}$$

More generally, the properties of the ϵ-insensitive loss and the Huber-loss can be combined through imprecisiation by a trapezoidal fuzzy set $\tilde{Y}$ with core $[y-\epsilon, y+\epsilon]$ and support[2] $[y-\delta, y+\delta]$. The loss L^{**} thus obtained is insensitive in the core, behaves quadratically in the boundary region $(y-\delta, y-\epsilon) \cup (y+\epsilon, y+\delta)$, and like L_1 outside the support (cf. Fig. 1).

4.2. Loss functions for classification

In the setting of classification, the set $\mathcal{Y}$ is finite and comprises K classes $\{c_1, \ldots, c_K\}$. The loss function most commonly used in this setting is the simple 0/1 loss $L(y, \hat{y}) = [\![y \neq \hat{y}]\!]$. Now, assume an outcome to be characterized in terms of a fuzzy subset $\tilde{Y}$ of $\mathcal{Y}$, that is, in terms of a membership degree $\tilde{Y}(c_i)$ for each class label $c_i \in \mathcal{Y}$, where the latter can be considered as a degree of plausibility that the true class label is given by c_i. The FOSL loss (5) obtained for this imprecisiation is then given by $L^{**}(\tilde{Y}, \hat{y}) = 1 - \tilde{Y}(\hat{y})$. As can be seen, the higher the membership degree of the predicted class $\hat{y}$, the smaller the loss, which is intuitively plausible. We obtained an interesting special case for a fuzzy set of the type

$$\tilde{Y}(c) = \begin{cases} 1 & \text{if } c = c_k \\ 1-w & \text{if } c \neq c_k \end{cases} , \tag{6}$$

for some $k \in [K]$ and $w \in [0, 1]$. Making use of this fuzzy set for modeling the observation of class label c_k can be seen as a *discounting* of this observation: While c_k is still regarded as completely plausible, the other class labels are no longer completely excluded. Alternatively, w can be seen as a degree of certainty that the observed class is indeed c_k. For a fuzzy observation of that kind,

$$L^{**}(\tilde{Y}, \hat{y}) = \begin{cases} 0 & \text{if } \hat{y} = c_k \\ w & \text{if } \hat{y} \neq c_k \end{cases}$$

effectively reduces the penalty for a misclassification from 1 to w. Or, stated differently, the training example is *weighted* by the factor w.

More generally, for fuzzy supersets of the form (6), the α-cuts in (5) are given by $[\tilde{Y}]_\alpha = \{c_k\}$ if $1 - w < \alpha \leq 1$ and $[\tilde{Y}]_\alpha = \mathcal{Y}$ otherwise. Therefore, the resulting loss L^{**} takes the form

$$L^{**}(\tilde{Y}, \hat{y}) = w \cdot L(c_k, \hat{y}) + (1-w) \cdot L^{*}(\mathcal{Y}, \hat{y}) \ . \tag{7}$$

Note that the second term, $L^*(\mathcal{Y}, \hat{y})$, does not depend on c_k, only on the prediction $\hat{y}$. For many losses L, the extension L^* satisfies $L^*(\mathcal{Y}, \hat{y}) = \min_{c \in \mathcal{Y}} L(c, y) = 0$ – the 0/1 loss discussed before is an example of that kind. In this case, (7) coincides with multiplicative scaling. However, as we will see in the following section, there are also losses L for which $L^*(\mathcal{Y}, \hat{y}) > 0$. Then, some predictions $\hat{y}$ might be systematically preferred over (less penalized than) others, regardless of the true class label c_k. An important example is given by *margin losses*, which penalize predictions close to 0, i.e., close to the decision boundary.

As already said, it is interesting to compare this kind of data imprecisiation with the use of "soft labels", e.g., as done in label smoothing [42]. As one important difference, note that imprecisiation allows for increasing the plausibility of certain classes (represented by their respective degree of membership in the fuzzy set) without decreasing the plausibility of the others. Compared to softening, in which a constant probability mass of 1 can only be shifted between the class labels, this is a true relaxation, and hence a true discounting of the training instance. According to (6), for example, the actually observed label c_k remains a completely plausible candidate, and a learner predicting this label is not penalized at all. In a soft version, the probability of c_k would be reduced from 1 to some smaller value, and even a learner predicting this label would incur a loss. Lienen and Hüllermeier [25] studied this in depth, also for multi-class problems, and demonstrated improved calibration capabilities of deep neural networks when training with imprecisiation.

4.3. Margin losses

As mentioned before, yet another interesting case is given by *margin losses*, which constitute an important class of loss functions in binary classification [34]. Roughly speaking, while the 0/1 loss merely checks whether a prediction is on the right side of the decision boundary, margin losses depend on *how much* on the right or wrong side the prediction is. Thereby, such losses enforce a "large margin" effect, helping to separate the classes as much as possible.

More formally, a margin loss is a function of the form

$$L(y, s) = f(ys) \ , \tag{8}$$

where $y \in \mathcal{Y} = \{-1, +1\}$ encodes the class label (negative and positive), $s \in \mathbb{R}$ is a score produced by a *scoring classifier* $h : \mathcal{X} \longrightarrow \mathbb{R}$, and $f : \mathbb{R} \longrightarrow \mathbb{R}$ is non-increasing. Here, a negative score $s = h(\boldsymbol{x}) < 0$ is meant to suggest that $\boldsymbol{x}$ belongs to negative class, whereas a positive score suggests $\boldsymbol{x}$ to be positive. As can be seen, the larger (smaller) the score in the case of

[2] The core of a fuzzy set is the set of elements with membership degree 1, the support is the (closure) of the set of elements with positive membership.

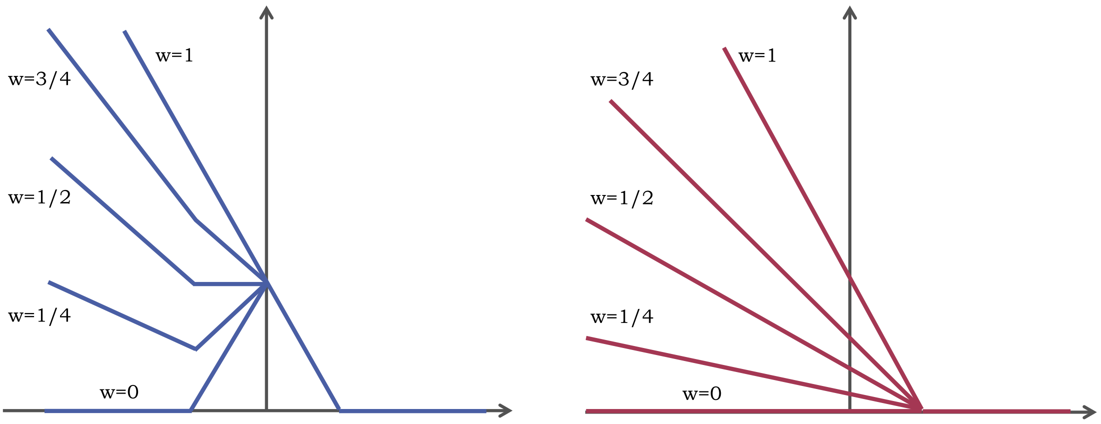


Fig. 2. (Color online.) Left: FOSL loss function (5) for different weightings w of the training example. Right: Standard version of weighted hinge loss, obtained by multiplying the original loss with w.

a positive (negative) class, the smaller the loss. Important examples of (8) include the logistic loss $f(ys) = \log\left(1+\exp(-ys)\right)$ closely connected with logistic regression, the hinge loss

$$L(y,s) \;=\; f(ys) = \max\left\{1-ys, 0\right\} \tag{9}$$

underlying support vector machines [36,45], and the exponential loss $f(ys) = \exp(-ys)$ used in boosting algorithms [35].

Now, let the output again be characterized in terms of a fuzzy subset $\tilde{Y}$ of $\mathcal{Y}$, i.e., in terms of membership degrees $\tilde{Y}(-1)$ and $\tilde{Y}(+1)$ for the two possible classes, negative and positive. More concretely, let us again assume the special case

$$\tilde{Y}(c) \;=\; \begin{cases} 1 & \text{if } c = y \\ 1-w & \text{if } c = \bar{y} \end{cases} \quad , \tag{10}$$

where $\{y, \bar{y}\} = \{-1, +1\}$ and w is a degree of confidence in y. For this case, one readily verifies that the loss (5), in accordance with (7), takes the following form:

$$\begin{aligned} L^{**}(\tilde{Y}, s) &= f_w(ys) \\ &= w \cdot f(ys) + (1-w) \cdot f(|ys|) \end{aligned} \tag{11}$$

Note that, if $ys > 0$, which means that the prediction $s = h(\boldsymbol{x})$ is in favor of the more likely class y, then f_w coincides with the original margin loss f. In fact, the difference between the original loss and (11) only concerns the negative part. As an illustration, the graph of (11) for the case of the hinge loss is shown in Fig. 2 (left).

As shown by this example, (11) does neither preserve monotonicity nor convexity when w gets small. From a computational perspective, this is of course an undesirable property, since optimization greatly benefits from these properties. Apart from that, the non-monotone behavior of the loss may also look a bit surprising, at least at first sight. It does, however, make perfect sense. This becomes clear when realizing that, in contrast to the simple 0/1 loss, a margin loss pursues two goals at the same time, namely correct classification and separation of the data. To meet the first objective, the penalty should decrease with decreasing w, just like the 0/1 loss does; this is the reason why $f_w \leq f_{w'}$ for $w \leq w'$. However, an increase of the margin should also be rewarded at the same time. This is reflected by the two components in Eq. (7). Taking both effects (a discounted penalty for misclassification and a reward for an increased margin) together, it can happen that a correct classification with a small margin is penalized more than an incorrect classification with a large margin.

5. Instance weighting for SVM

The arguably most obvious idea for incorporating instance weighting in binary support vector machines is by minimizing the weighted instead of the original hinge loss (9). The former is simply obtained by scaling, i.e., multiplying the original loss with a weight that reflects the (assumed) atypicality of the instance (cf. Fig. 2). This approach was adopted, for example, in [49] to train an SVM on noisy data. The soft-margin regularized loss formulation of the weighted SVM (WSVM) is given by

$$J(\boldsymbol{\theta}) = \frac{1}{2}\|\boldsymbol{\theta}\|_2^2 + C\sum_{i=1}^{N} w_i \max\left\{0, 1 - y_i\langle\boldsymbol{\theta}, \boldsymbol{x}_i\rangle\right\}, \tag{12}$$

with $\boldsymbol{\theta}$ the SVM model parameters, N the number of instances $(\boldsymbol{x}_i, y_i)$, and w_i the weight of the i^{th} example. The complexity parameter C controls the influence of the error (violation of margin constraints) in the optimization term.

Instead of using the scaled version of the original hinge loss, our idea is to use the FOSL-version (11) obtained through data imprecisiation. Thus, our approach to instance weighting in SVM based on data imprecisiation (SVM-DI) comes down to minimizing

$$\hat{J}(\boldsymbol{\theta}) = \frac{1}{2}\|\boldsymbol{\theta}\|_2^2 + C\sum_{i=1}^{N} f_{w_i}\big(y_i\langle\boldsymbol{\theta},\boldsymbol{x}_i\rangle\big)\,. \tag{13}$$

Especially interesting is the observation that, in the case $w_i = 0$ reflecting full uncertainty (maximal imprecision), our generalized loss coincides with the so-called "hat loss", which is commonly used in semi-supervised support vector classification [56]. Interpreting the absence of any label information as complete ignorance, this makes perfect sense: despite not expressing any preference in favor of one of the classes, the hat loss still encourages a margin effect. In contrast to this, the weighted hinge loss simply vanishes (and yields 0 everywhere), which means that unlabeled data points have no influence on the training process anymore. One may argue, therefore, that our approach of data imprecisiation is more consistent with the (limiting) case of semi-supervised learning than the standard WSVM approach using the weighted hinge loss: while the former "converges" to semi-supervised SVM proper when $w \to 0$, the latter does not.

5.1. *Concave-convex optimization*

While minimizing (12) is a convex optimization problem, the generalized version (13) is non-convex. Therefore, we leverage the concave-convex procedure (CCCP) by Yuille and Rangarajan [54] to determine $\boldsymbol{\theta}$. To this end, f_w is reformulated as a sum of a convex part $f_w^{\text{vex}}(x)$ and a concave part f_w^{cav} as follows:

$$\begin{aligned} f_w(x) &= wf(x) + (1-w)f(|x|) \\ &= \underbrace{f(x) + \max\big\{0,(w-1)(x+1)\big\}}_{f_w^{\text{vex}}} + \underbrace{\min\big\{0, 2(1-w)x\big\}}_{f_w^{\text{cav}}} \end{aligned} \tag{14}$$

Using $f_w^{\text{vex}}(x)$ and f_w^{cav}, $\hat{J}(\boldsymbol{\theta})$ can be expressed as

$$\hat{J}^{\text{CCCP}}(\boldsymbol{\theta}) = \underbrace{\frac{1}{2}\|\boldsymbol{\theta}\|_2^2 + C\sum_{i=1}^{N} f_{w_i}^{\text{vex}}(y_i\langle\boldsymbol{\theta},\boldsymbol{x}_i\rangle)}_{\hat{J}_{\text{vex}}^{\text{CCCP}}(\boldsymbol{\theta})} + \underbrace{C\sum_{i=1}^{N} f_{w_i}^{\text{cav}}(y_i\langle\boldsymbol{\theta},\boldsymbol{x}_i\rangle)}_{\hat{J}_{\text{cav}}^{\text{CCCP}}(\boldsymbol{\theta})}\,. \tag{15}$$

Algorithm 1 is then used to solve the non-convex problem in an iterative manner with nested convex optimization problems. Note that CCCP requires the partial derivatives $\frac{\partial \hat{J}_{\text{cav}}^{\text{CCCP}}(\boldsymbol{\theta})}{\partial\theta_i}$, which might not exist due to the non-differentiability of f_w^{cav} at 0. To overcome this issue, we smoothed f_w^{cav} by a polynomial of degree two within a small interval around the origin.

Algorithm 1 The Concave-Convex Procedure (CCCP) applied to SVM-DI model induction.

```
Initialize $\boldsymbol{\theta}^0$ with a best guess (e.g., by training a conventional model beforehand)
while $\boldsymbol{\theta}^t$ not converged do
    $\boldsymbol{\theta}^{t+1} = \text{argmin}_{\boldsymbol{\theta}}\, \hat{J}_{\text{vex}}^{\text{CCCP}}(\boldsymbol{\theta}) + (\hat{J}_{\text{cav}}^{\text{CCCP}})'(\boldsymbol{\theta}^t)\cdot\boldsymbol{\theta}$
end while
```

5.2. *Learning from noisy data*

In our experimental study below, we consider the problem of learning from noisy data as one application of instance weighting. Learning from noisy data through instance weighting first of all requires weights to be assigned to all training examples, reflecting their respective degree of distrust. To this end, Wu and Liu [50] propose the principle of the one-step weighted SVM (OWSVM), which obtains such weights by training a standard SVM in a first step. More specifically, each training example is weighted based on its distance to the decision boundary of this SVM. Then, given the weights, OWSVM retrains the model using the weighted hinge loss, i.e., by minimizing (12).

Adopting the same principle, our adaptation based on data imprecisiation is essentially the same as OWSVM, except for the weighted loss: Instead of using a simple weighting of the hinge loss, we use the FOSL loss L^{**}, i.e., we minimize (13). To assure that the weights w_i are in $[0,1]$, these weights are determined as $w_i = s(y\langle\boldsymbol{\theta},\boldsymbol{x}\rangle)$, where $\boldsymbol{\theta}$ is the parameter of the standard SVM trained in the first step, and s is a sigmoidal function such that $s(-1) = \varepsilon = 1 - s(+1)$. $\varepsilon \in (0,1)$ denotes a hyperparameter used to adjust the extremity of the weighting.

Collobert et al. [9] use a related approach to learn SVMs on a non-convex ramp loss, using the weights of a pre-trained SVM as initial weights $\boldsymbol{\theta}_0$ in Algorithm 1. The ramp loss is another modification (relaxation) of the hinge loss (9), which is defined as follows:

Table 1
Data sets, their OpenML [44] identifiers and characteristics used within the experimental studies.

Data set	ID	# Instances	# Features
Credit-g	31	1000	20
Diabetes	37	768	8
Haberman	43	306	3
Monks1	333	556	6
Monks2	334	601	6
Transplant	885	131	3
Banknote	1462	1372	4
Blood Transf.	1464	748	4
Parkinsons	1488	195	23
Sa-heart	1498	462	9
Wdbc	1510	569	30

$$L_r(y,s) = \min\Big\{ \max\{0, 1-ys\}, 1-r \Big\} \tag{16}$$

Thus, the ramp loss truncates the hinge loss by a constant at $s = r$, where $r \in (-\infty, 0]$ is a parameter of the loss (and hence a hyperparameter of the learning algorithm).

5.3. Self-training

Another interesting application of instance weighting is self-training [4]. In a semi-supervised learning setting, given labeled and unlabeled data $\mathcal{D}^{\text{lab}} \in (\mathcal{X} \times \mathcal{Y})^N$ and $\mathcal{D}^{\text{unlab}} \in \mathcal{X}^N$, respectively, a learner induces a first model from $\mathcal{D}^{\text{lab}}$. This model is then used to (hypothetically) label the instances $\boldsymbol{x} \in \mathcal{D}^{\text{unlab}}$, which are then used to enrich the training data and retrain the model. The whole process of enriching the training data and retraining can be iterated until a stable solution is reached.

Enriching the data can be done in different ways. Commonly, for example, those instances are added to the training data on which the learner is certain enough, while the other are left out. Of course, if the learner is able to handle weighted examples, a more elegant approach is to include all (hitherto unlabeled) examples weighted by their respective degree of certainty, e.g., similar to the strategy by Wu and Liu [50] as mentioned before. This is what we are going to try in our experimental study below (starting with a weight of 0 for the unlabeled examples in the first step).

6. Experiments with benchmark data

In our experimental studies with standard benchmark data, we compare different variants of weighted support vector machines in two settings, namely classification on noisy data and self-training. Experiments are conducted on 11 binary classification data sets with varying numbers of instances and features, whose characteristics are summarized in Table 1. We report the average and standard deviation over 20 repetitions of each experiment.

6.1. Robust binary classification

In the first study, we empirically compare our method SVM-DI with other robust loss minimization techniques in the setting of noisy data, i.e., classification problems in which the observed class labels are corrupted with noise [14]. To this end, we evaluated the methods on cross-validation folds $(\mathcal{D}^i_{\text{train}}, \mathcal{D}^i_{\text{test}})$ with $i \in \{1, \ldots, 5\}$ of the original data set $\mathcal{D}$ to provide better out-of-sample error estimates. For each of them, we randomly flipped the labels of the training instances in $\mathcal{D}^i_{\text{train}}$ (independently of each other) with probabilities $p \in \{0, 0.1, 0.2, 0.3, 0.4\}$. This data is used in a nested 5-fold cross-validation to optimize the considered hyperparameters. The results finally reported are the averaged misclassification rates on the (untouched) sets $\mathcal{D}^i_{\text{test}}$ for all of the 5 outer folds. Thus, we employ a nested 5 × 5 cross-validation for assessment.

In addition to the conventional SVM, we include OWSVM and a robust support vector machine (RSVM) as described by Collobert et al. [9] in our study. While OWSVM can be considered as a natural counterpart to SVM-DI, a comparison with RSVM is interesting as well, mainly because the latter also solves a non-convex optimization problem — minimization of the non-convex ramp loss (16) — using the CCCP algorithm.

Since the parameter C is used by all methods for complexity control, this parameter is fixed beforehand to facilitate comparability. For SVM-DI, the hyperparameter ε is optimized within the interval $[0.001, 0.1]$. In the case of RSVM, the ramp loss threshold is tuned as a hyperparameter with values $r \in [-5, 0]$.

Table 2 shows the results of the first experimental study for increasing noise levels on the different data sets. As a first observation, the OWSVM approach increases robustness compared to a conventional SVM, which confirms the desired effect of instance weighting. More importantly, compared to the other methods, SVM-DI tends to be more robust with an increasing level of noise, while being competitive in less noisy cases; in the case of 40% noise, the increase in performance is significant compared to SVM and RSVM according to a one-sided Wilcoxon signed-rank test [48] at a level 0.05. These

Table 2
Experimental results for the first setting: Average misclassification rates with standard deviations on test data for different methods, data sets and noise levels.

Noise	Data	SVM	RSVM	OWSVM	SVM-DI
0%	1462	**0.009** ± 0.001	0.020 ± 0.005	**0.009** ± 0.001	0.010 ± 0.001
	1464	**0.235** ± 0.002	0.238 ± 0.001	0.236 ± 0.002	0.236 ± 0.002
	1488	0.119 ± 0.011	0.181 ± 0.015	**0.116** ± 0.012	0.123 ± 0.013
	1498	**0.278** ± 0.006	0.287 ± 0.010	0.284 ± 0.007	0.284 ± 0.008
	1510	**0.023** ± 0.003	0.042 ± 0.006	**0.023** ± 0.002	0.033 ± 0.005
	31	0.240 ± 0.005	**0.237** ± 0.004	0.247 ± 0.006	0.247 ± 0.005
	333	0.320 ± 0.007	0.325 ± 0.011	**0.300** ± 0.015	0.319 ± 0.008
	334	0.343 ± 0.000	0.343 ± 0.000	0.343 ± 0.000	0.343 ± 0.000
	37	**0.225** ± 0.004	0.234 ± 0.005	0.230 ± 0.005	0.229 ± 0.004
	43	**0.264** ± 0.003	0.267 ± 0.008	0.266 ± 0.003	0.265 ± 0.004
	885	**0.009** ± 0.007	0.016 ± 0.008	0.012 ± 0.008	0.023 ± 0.009
10%	1462	0.023 ± 0.001	0.023 ± 0.001	**0.018** ± 0.001	**0.018** ± 0.001
	1464	0.238 ± 0.002	0.238 ± 0.001	0.238 ± 0.001	**0.237** ± 0.002
	1488	0.155 ± 0.023	0.201 ± 0.022	**0.149** ± 0.022	0.151 ± 0.023
	1498	0.287 ± 0.015	0.292 ± 0.013	**0.284** ± 0.012	0.291 ± 0.015
	1510	0.039 ± 0.005	0.035 ± 0.006	**0.032** ± 0.006	0.038 ± 0.006
	31	0.255 ± 0.010	0.256 ± 0.012	**0.253** ± 0.010	0.259 ± 0.008
	333	0.334 ± 0.001	0.335 ± 0.000	**0.330** ± 0.021	0.332 ± 0.005
	334	0.343 ± 0.000	0.343 ± 0.000	0.343 ± 0.000	0.343 ± 0.000
	37	0.237 ± 0.006	0.237 ± 0.008	0.238 ± 0.014	**0.234** ± 0.006
	43	**0.265** ± 0.002	0.271 ± 0.016	**0.265** ± 0.002	0.266 ± 0.002
	885	0.042 ± 0.017	0.030 ± 0.012	**0.023** ± 0.013	0.035 ± 0.015
20%	1462	0.026 ± 0.003	0.026 ± 0.002	**0.022** ± 0.002	**0.022** ± 0.002
	1464	**0.238** ± 0.002	0.239 ± 0.003	**0.238** ± 0.001	**0.238** ± 0.002
	1488	0.187 ± 0.021	0.226 ± 0.031	**0.176** ± 0.022	0.182 ± 0.025
	1498	0.306 ± 0.019	0.312 ± 0.019	**0.302** ± 0.022	0.310 ± 0.022
	1510	0.060 ± 0.009	0.057 ± 0.029	**0.048** ± 0.008	0.055 ± 0.007
	31	0.271 ± 0.008	0.273 ± 0.009	**0.270** ± 0.006	0.279 ± 0.008
	333	0.334 ± 0.002	**0.333** ± 0.003	0.336 ± 0.019	0.334 ± 0.002
	334	0.343 ± 0.000	0.343 ± 0.000	0.343 ± 0.000	0.343 ± 0.000
	37	0.260 ± 0.018	0.255 ± 0.027	**0.247** ± 0.013	0.255 ± 0.013
	43	0.266 ± 0.003	0.266 ± 0.004	**0.265** ± 0.004	**0.265** ± 0.003
	885	0.071 ± 0.029	0.065 ± 0.022	**0.043** ± 0.021	0.066 ± 0.030
30%	1462	0.029 ± 0.003	0.030 ± 0.002	**0.026** ± 0.003	0.028 ± 0.004
	1464	0.241 ± 0.011	0.244 ± 0.020	0.239 ± 0.005	**0.237** ± 0.002
	1488	0.256 ± 0.047	0.289 ± 0.052	0.249 ± 0.051	**0.244** ± 0.043
	1498	0.327 ± 0.028	0.331 ± 0.022	**0.326** ± 0.023	0.328 ± 0.023
	1510	0.085 ± 0.018	**0.072** ± 0.016	0.076 ± 0.018	0.078 ± 0.017
	31	0.295 ± 0.010	0.292 ± 0.009	**0.289** ± 0.012	0.294 ± 0.010
	333	**0.339** ± 0.010	**0.339** ± 0.011	0.342 ± 0.038	0.340 ± 0.013
	334	**0.343** ± 0.001	**0.343** ± 0.000	0.344 ± 0.004	**0.343** ± 0.000
	37	0.289 ± 0.025	0.282 ± 0.027	**0.276** ± 0.022	0.291 ± 0.027
	43	0.266 ± 0.003	0.271 ± 0.024	0.268 ± 0.014	**0.265** ± 0.003
	885	0.116 ± 0.060	0.115 ± 0.058	**0.093** ± 0.047	0.112 ± 0.058
40%	1462	0.035 ± 0.004	0.034 ± 0.003	0.033 ± 0.004	**0.031** ± 0.004
	1464	0.258 ± 0.034	0.269 ± 0.045	0.252 ± 0.025	**0.251** ± 0.026
	1488	0.351 ± 0.059	0.382 ± 0.054	0.349 ± 0.056	**0.347** ± 0.065
	1498	0.353 ± 0.024	0.360 ± 0.037	0.357 ± 0.026	**0.349** ± 0.025
	1510	0.171 ± 0.061	0.175 ± 0.067	0.175 ± 0.068	**0.165** ± 0.063
	31	0.341 ± 0.022	**0.333** ± 0.020	0.344 ± 0.019	0.338 ± 0.018
	333	**0.401** ± 0.039	0.407 ± 0.045	0.412 ± 0.049	0.402 ± 0.037
	334	0.384 ± 0.044	**0.372** ± 0.042	0.401 ± 0.043	0.381 ± 0.046
	37	0.320 ± 0.025	0.325 ± 0.024	**0.314** ± 0.020	0.325 ± 0.020
	43	0.300 ± 0.066	0.314 ± 0.077	0.307 ± 0.068	**0.299** ± 0.066
	885	0.228 ± 0.124	0.228 ± 0.120	**0.205** ± 0.118	0.228 ± 0.128

results support our conjecture that instance weighting through data imprecisiation can be more effective than other types of instance weighting.

6.2. Semi-supervised self-training

In a second study, the setting of self-training as described in Section 5.3 is examined. Here, we consider a conventional weighted (hinge loss) SVM (WSVM) for comparison to our method.

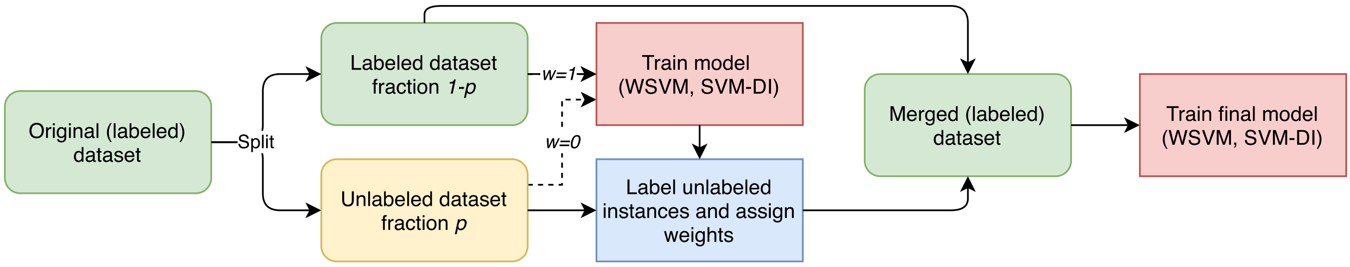

Original (labeled) dataset
Split
Labeled dataset fraction 1-p
Unlabeled dataset fraction p
w=1
w=0
Train model (WSVM, SVM-DI)
Label unlabeled instances and assign weights
Merged (labeled) dataset
Train final model (WSVM, SVM-DI)

Fig. 3. (Color online.) The basic setting of the semi-supervised self-training experiments.

Table 3
Experimental results for the second setting: Average misclassification rates on test data for different methods, data sets and split proportions p.

Data	$p = 0.5$		$p = 0.75$		$p = 0.9$	
	WSVM	SVM-DI	WSVM	SVM-DI	WSVM	SVM-DI
1462	**0.010** ± 0.002	0.011 ± 0.002	0.014 ± 0.002	**0.013** ± 0.002	0.019 ± 0.003	**0.018** ± 0.002
1464	0.238 ± 0.001	0.238 ± 0.000	0.238 ± 0.002	0.238 ± 0.000	0.241 ± 0.008	**0.238** ± 0.001
1488	0.165 ± 0.032	**0.158** ± 0.031	0.203 ± 0.047	**0.199** ± 0.053	0.216 ± 0.049	**0.182** ± 0.027
1498	0.290 ± 0.013	**0.289** ± 0.014	**0.304** ± 0.017	0.308 ± 0.017	0.333 ± 0.018	**0.331** ± 0.016
1510	**0.029** ± 0.005	0.037 ± 0.005	**0.036** ± 0.007	0.042 ± 0.006	**0.051** ± 0.010	0.056 ± 0.011
31	**0.249** ± 0.006	0.250 ± 0.008	0.262 ± 0.010	**0.260** ± 0.010	0.307 ± 0.023	0.307 ± 0.024
333	**0.326** ± 0.008	0.333 ± 0.006	0.332 ± 0.007	**0.331** ± 0.008	0.357 ± 0.027	**0.356** ± 0.025
334	0.343 ± 0.000	0.343 ± 0.000	0.343 ± 0.000	0.343 ± 0.000	0.343 ± 0.000	0.343 ± 0.000
37	0.240 ± 0.015	**0.234** ± 0.009	0.260 ± 0.040	**0.254** ± 0.027	0.278 ± 0.037	**0.275** ± 0.039
43	**0.264** ± 0.005	0.265 ± 0.002	0.264 ± 0.003	0.264 ± 0.001	0.264 ± 0.001	0.264 ± 0.004
885	**0.022** ± 0.015	0.028 ± 0.010	**0.041** ± 0.026	0.045 ± 0.020	0.073 ± 0.037	**0.071** ± 0.028

To this end, the training data of a (fully-labeled) data set $\mathcal{D}$ is randomly split into labeled and unlabeled parts $\mathcal{D}^{\text{lab}}$ and $\mathcal{D}^{\text{unlab}}$ in a stratified manner. First, an initial model (WSVM or SVM-DI) is trained on the given data, assigning a weight of $w = 1$ to labeled and $w = 0$ to unlabeled instances. After the initial training, the model (SVM) $\boldsymbol{\theta}$ is used to label the instances $\boldsymbol{x} \in \mathcal{D}^{\text{unlab}}$, with the class predictions suggested by $\boldsymbol{\theta}$. Moreover, each instance is assigned the weight $w(\boldsymbol{x}) = (1 + |\langle \boldsymbol{\theta}, \boldsymbol{x} \rangle|)^{-1}$. In a second step, a final model is then trained on the new data consisting of $\mathcal{D}^{\text{lab}}$ enriched by $\mathcal{D}^{\text{unlab}}$, again using either WSVM or SVM-DI. Fig. 3 depicts the basic scheme. As already said, this procedure could in principle be iterated. However, increasing the number of iterations has not shown notable effects in our study.

The results are shown in Table 3. While SVM-DI and WSVM are on a par for equally-sized splits ($p = 0.5$), SVM-DI tends to become more and more superior for increasing p (here shown for $p \in \{0.75, 0.9\}$); in the case of $p = 0.9$, the differences are significant according to a Wilcoxon signed-rank test at the level 0.05. These results are coherent with those of the first study, suggesting that instance weighting through data imprecisiation is not only a viable alternative to standard scaling, but tends to have advantages in cases where the uncertainty is high and/or precise information very limited.

7. Case study: monocular depth estimation

As an additional real-world case study involving noisy data, we elaborate on the problem of depth estimation of objects in monocular images, i.e., predicting the depth (distance from the camera) of a projection solely given by single images. As ground truth, dense maps with pixel-wise depth information are typically given. Here, we consider a simplified version of the problem: Instead of predicting dense depth maps [55], we aim to predict an order relation for pairs of points on an image, namely whether the first or the second point is closer to the camera. That is, the features of a training instance i on an image I are of the form $(I, l_{i,1}, l_{i,2})$, where $l_{i,1}$ and $l_{i,2}$ indicate two points on I. If $z_{i,1}$ and $z_{i,2}$ are the (ground truth) depth values of the respective points, the target label is $+1$ if $z_{i,1} < z_{i,2}$ and -1 otherwise. For simplicity, we do not explicitly model the case of a tie ($z_{i,1} = z_{i,2}$). Instead, we simplify the problem by aiming to predict whether $l_{i,1}$ is closer to the camera or not.

Although the ground truth values are precise, i.e., the depth information is given as a single value per pixel, commonly used depth sensors, such as Kinect, are known to produce distorted values (cf. [47]). As a consequence, binary depth relations might also be wrong (and hence misleading for the learning algorithm), especially in the case of very small absolute depth differences. To address this issue, an obvious idea is to use these absolute differences for instance weighting: Point pair instances with small differences get less weight, while more certain relations with higher differences get a higher weight. For our experiments, we assigned instance weights w_i according to $w_i = \tanh\left(2 \cdot |z_{i,1} - z_{i,2}|\right)$.

Since the reformulated depth prediction problem constitutes a binary classification problem, it can be tackled by a single decision boundary. Therefore, we compare our FOSL loss adaptation as used in the previous experiments to the conventional hinge loss with and without instance weighting. Moreover, as FOSL is capable of considering unlabeled instances with weight 0 in a semi-supervised manner, we also consider the case of augmenting the (labeled) training examples by raw images without depth maps. In practice, this is a fairly common scenario: only a small subset of the available images are labeled,

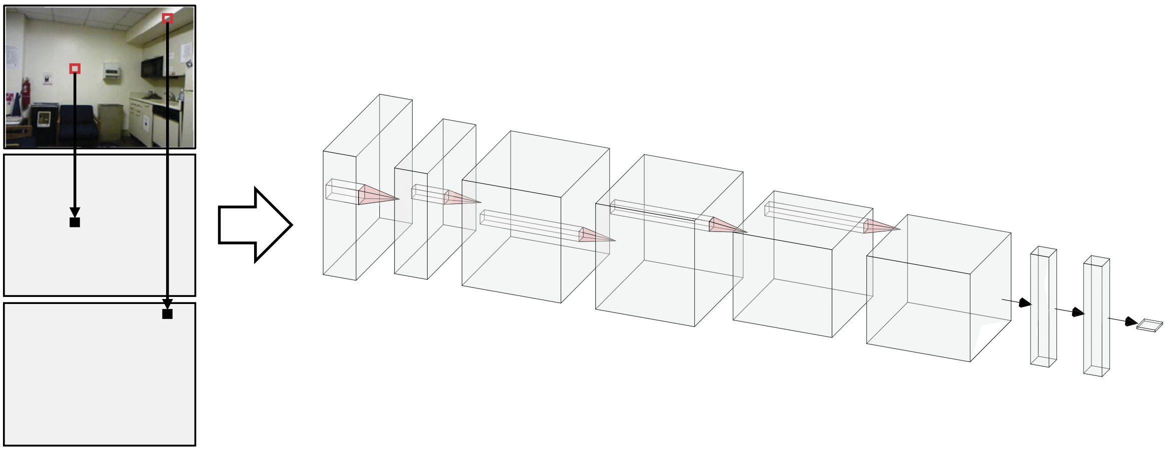


Fig. 4. (Color online.) Sketch of the architecture being used within the experiments: Two points (red) are sampled in the original image, for which two masks are attached to the 3-channel image. The resulting 5-channel input is then fed into a convolutional neural network inspired by the hourglass architecture from [5], where the individual building blocks are Inception modules [41] followed by BatchNormalization and average pooling layers. The final layer is a linearly activated output neuron for the score prediction.

since annotation is costly. For the conventional hinge loss, we consider a semi-supervised baseline similar to the WSVM approach compared in Sec. 6.2. More precisely, we iteratively used the current model hypothesis to label drawn unlabeled instances and assign weights as we did for WSVM. For labeled instances, we considered them as completely reliable and assigned weights of value 1 in this case.

Here, we use the NYUD-v2 data set [38] that includes 1449 aligned, densely annotated RGB images of (adjusted) size 640 × 480 together with ground truth depth mappings of the same size sensed with Kinect. Additionally, it provides more than 400k frames with depth annotations that are not perfectly synchronized with the input images. For the purpose of our study, we considered them as unlabeled. Within our experiments on the (synchronized) labeled images solely, we used the same training and test split of the images as in [57] and [5]. For training, we sampled 50 random pairs per image in each epoch. Similar to the sampling approach as described by Chen et al. [5], we constrained the maximum distance for each sampled pair to be between 13 and 19. For the semi-supervised experiment, we sample $10,000$ images per epoch with 50 point pairs each from the available raw images and sample random points without any distance constraint. For the test data, we fixed 100 randomly sampled pairs per image, skipping examples with a depth difference below 0.1 meters (effectively assigning them a weight of 0) due to the imprecision of the Kinect sensor.

The problem of inducing depth information from raw monocular images is complex and ill-posed. This is the reason why modern approaches typically involve deep learning models [55]. Following this trend, we apply a state-of-the-art deep convolutional neural network architecture similar to the encoder part of the hourglass architecture as used in [5–7]. As input, we feed in the images bi-linearly downsampled to 160 × 120, concatenated with two channels masking the two sample points: The chosen pixels are masked by 1, while the other pixels have a channel value of 0. Thus, the input shape of the model is $160 \times 120 \times 5$. At the final stage, a fully-connected classification layer with two dense hidden layers consisting of 256 neurons each is attached to the model structure to predict the final class. Altogether, the model consists of approximately 900k parameters. The architecture is sketched in Fig. 4.

As hyperparameters, we used a batch size of 64, trained for 50 epochs and set the initial learning rate to 0.0001 for optimization with AMSGrad [32]. We multiplied the learning rate by $\sqrt{0.1}$ after 30 and 40 epochs. In addition to using BatchNormalization [19], we also optimized the L2 regularization penalty of all convolutional layers on a separate validation set. To regularize the last dense layers, we used Dropout [39] with a rate of 0.2. Each experiment is conducted 3 times with different seeds.

The resulting error rates on the test set are given in Table 4. As can be seen, instance weighting increases the classification performance compared to non-weighting, for both hinge loss and FOSL. Furthermore, the results indicate the effectiveness of optimizing the non-convex FOSL with conventional deep learning methods. As the results show, the performance is competitive to conventional hinge loss, although it involves a more complex optimization problem. Moreover, incorporating unlabeled instances turns out to be beneficial to learn from, resulting in the best performance among the considered models, whereas self-training turns out to slightly harm the performance of the conventional hinge loss. This further supports FOSL as being a compelling choice to use in semi-supervised contexts with the hat loss as a special case for zero weights.

8. Conclusion

We proposed a generic approach to instance weighting through data imprecisiation. Within the framework of (optimistic) superset learning, this effectively comes down to modifying an underlying loss function on a per-instance basis. Interestingly, several existing loss functions can be recovered as special cases of this approach, including the Huber-loss for robust

Table 4
Error rates on 100 randomly sampled point pairs for each image of the NYUD-v2 test set. *: Instance weights have only been assigned to self-labeled instances to reduce their influence compared to originally labeled instances.

Loss	Instance Weighting	Semi-Supervision	Error rate
Hinge	No	No	0.263 ± 0.014
Hinge	Yes	No	0.231 ± 0.011
Hinge	Yes*	Yes	0.237 ± 0.021
FOSL	Yes	No	0.233 ± 0.009
FOSL	Yes	Yes	**0.218** ± 0.023

regression, the ϵ-insensitive loss used in support vector regression, and the hat loss used in semi-supervised learning with SVMs.

A specific realization of this approach, namely instance weighting for (binary) support vector machines (SVM-DI), was analyzed in more detail and used to demonstrate its effectiveness in two experimental studies. As suggested by the results of these studies, instance weighting through data imprecisiation is highly competitive to standard techniques and tends to be superior in cases of very sparse or highly uncertain training data. Furthermore, we also showed the effectiveness of our approach on the ill-posed problem of monocular depth estimation, where pairwise depth relations between points in an image are weighted in accordance with the absolute distance of these points.

Motivated by these promising results, we plan to further elaborate on the idea of data imprecisiation in future work. In particular, by looking at other learning problems (such as regression) and methods, we plan to investigate further applications of our framework. Moreover, there are other variants of supervised learning, such as transfer learning, for which data imprecisiation appears to be highly promising.

Declaration of competing interest

None.

Acknowledgements

This work was supported by the German Research Foundation (DFG) (Grant No. 3050231323). Moreover, the authors gratefully acknowledge the funding of this project by computing time provided by the Paderborn Center for Parallel Computing (PC^2).

References

[1] Hessam Bagherinezhad, Maxwell Horton, Mohammad Rastegari, Ali Farhadi, Label Refinery: improving ImageNet classification through label progression, CoRR, arXiv:1805.02641 [abs], 2018.
[2] Battista Biggio, Blaine Nelson, Pavel Laskov, Support vector machines under adversarial label noise, in: Proceedings of the 3rd Asian Conference on Machine Learning, ACML, Taoyuan, Taiwan, November 13–15, 2011, in: JMLR Proceedings, vol. 20, 2011, pp. 97–112, JMLR.org.
[3] Vivien Cabannes, Alessandro Rudi, Francis R. Bach, Structured prediction with partial labelling through the infimum loss, in: Proceedings of the 37th International Conference on Machine Learning, ICML, Online, July 13–18, 2020, in: Proceedings of Machine Learning Research, vol. 119, PMLR, 2020, pp. 1230–1239.
[4] Olivier Chapelle, Bernhard Schölkopf, Alexander Zien, Semi-Supervised Learning, The MIT Press, 2006.
[5] Weifeng Chen, Zhao Fu, Dawei Yang, Jia Deng, Single-image depth perception in the wild, in: Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems, NIPS, December 5-10, 2016, Barcelona, Spain, 2016, pp. 730–738.
[6] Weifeng Chen, Shengyi Qian, Jia Deng, Learning single-image depth from videos using quality assessment networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Long Beach, CA, USA, June 16–20, 2019, Computer Vision Foundation / IEEE, 2019, pp. 5604–5613.
[7] Weifeng Chen, Shengyi Qian, David Fan, Noriyuki Kojima, Max Hamilton, Jia Deng, OASIS: a large-scale dataset for single image 3D in the wild, in: Proceedings of the IEEE Conference on Computer Vision and Pattern, Recognition, CVPR, Online, June 13–19, 2020, Computer Vision Foundation / IEEE, 2020, pp. 676–685.
[8] William S. Cleveland, Robust locally weighted regression and smoothing scatterplots, J. Am. Stat. Assoc. 74 (368) (1979) 829–836.
[9] Ronan Collobert, Fabian H. Sinz, Jason Weston, Léon Bottou, Trading convexity for scalability, in: Proceedings of the 23rd International Conference on Machine Learning, ICML, Pittsburgh, PA, USA, June 25–29, 2006, in: ACM International Conference Proceeding Series, vol. 148, ACM, 2006, pp. 201–208.
[10] Timothée Cour, Benjamin Sapp, Ben Taskar, Learning from partial labels, J. Mach. Learn. Res. 12 (2011) 1501–1536.
[11] Inés Couso, Didier Dubois, Statistical reasoning with set-valued information: ontic vs. epistemic views, Int. J. Approx. Reason. 55 (7) (2014) 1502–1518.
[12] Ehab E. Elattar, John Yannis Goulermas, Q. Henry Wu, Electric load forecasting based on locally weighted support vector regression, IEEE Trans. Syst. Man Cybern., Part C, Appl. Rev. 40 (4) (2010) 438–447.
[13] Yves Grandvalet, Logistic regression for partial labels, in: Proceedings of the 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU, Annecy, France, July 1-5, 2002, 2002, pp. 1935–1941.
[14] Bo Han, Ivor W. Tsang, Ling Chen, On the convergence of a family of robust losses for stochastic gradient descent, in: Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, Part I, Riva del Garda, Italy, September 19–23, 2016, in: LNCS, vol. 9851, Springer, 2016, pp. 665–680.
[15] Geoffrey E. Hinton, Oriol Vinyals, Jeffrey Dean, Distilling the knowledge in a neural network, CoRR, arXiv:1503.02531 [abs], 2015.
[16] Peter J. Huber, Robust Statistics, Wiley, 1981.

[17] Eyke Hüllermeier, Learning from imprecise and fuzzy observations: data disambiguation through generalized loss minimization, Int. J. Approx. Reason. 55 (7) (2014) 1519–1534.
[18] Eyke Hüllermeier, Weiwei Cheng, Superset learning based on generalized loss minimization, in: Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, Part II, Porto, Portugal, September 7–11, 2015, in: LNCS, vol. 9285, Springer, 2015, pp. 260–275.
[19] Sergey Ioffe, Christian Szegedy, Batch normalization: accelerating deep network training by reducing internal covariate shift, in: Proceedings of the 32nd International Conference on Machine Learning, ICML, Lille, France, July 6–11, 2015, in: JMLR Workshop and Conference Proceedings, vol. 37, 2015, pp. 448–456, JMLR.org.
[20] Rong Jin, Zoubin Ghahramani, Learning with multiple labels, in: Advances in Neural Information Processing Systems 15: Annual Conference on Neural Information Processing Systems, NIPS, Vancouver, BC, Canada, December 9-14, 2002, MIT Press, 2002, pp. 897–904.
[21] Herman Kahn, Andrew Marshall, Methods of reducing sample size in Monte Carlo computations, J. Oper. Res. Soc. Am. 1 (5) (1953) 263–278.
[22] George J. Klir, Tina A. Folger, Fuzzy Sets, Uncertainty, and Information, Prentice Hall, 1988.
[23] Maksim Lapin, Matthias Hein, Bernt Schiele, Learning using privileged information: SVM+ and weighted SVM, Neural Netw. 53 (2014) 95–108.
[24] Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, Li-Jia Li, Learning from noisy labels with distillation, in: Proceedings of the IEEE International Conference on Computer Vision, ICCV, Venice, Italy, October 22–29, 2017, IEEE Computer Society, 2017, pp. 1928–1936.
[25] Julian Lienen, Eyke Hüllermeier, From label smoothing to label relaxation, in: Proceedings of the 35th AAAI Conference on Artificial Intelligence, AAAI, Online, February 2-9, 2021, AAAI Press, 2021.
[26] Li-Ping Liu, Thomas G. Dietterich, A conditional multinomial mixture model for superset label learning, in: Advances in Neural Information Processing Systems 25: Annual Conference on Neural Information Processing Systems, NIPS, Lake Tahoe, NV, United States, December 3-6, 2012, 2012, pp. 557–565.
[27] Tongliang Liu, Dacheng Tao, Classification with noisy labels by importance reweighting, IEEE Trans. Pattern Anal. Mach. Intell. 38 (3) (2016) 447–461.
[28] Julian J. McAuley, Arnau Ramisa, Tibério S. Caetano, Optimization of robust loss functions for weakly-labeled image taxonomies, Int. J. Comput. Vis. 104 (3) (2013) 343–361.
[29] Rafael Müller, Simon Kornblith, Geoffrey E. Hinton, When does label smoothing help?, in: Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS, Vancouver, BC, Canada, December 8-14, 2019, 2019, pp. 4696–4705.
[30] Nagarajan Natarajan, Inderjit S. Dhillon, Pradeep Ravikumar, Ambuj Tewari, Learning with noisy labels, in: Advances in Neural Information Processing Systems 26: Annual Conference on Neural Information Processing Systems, NIPS, Lake Tahoe, NV, United States, December 5–8, 2013, 2013, pp. 1196–1204.
[31] Nam Nguyen, Rich Caruana, Classification with partial labels, in: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, NV, USA, August 24–27, 2008, ACM, 2008, pp. 551–559.
[32] Sashank J. Reddi, Satyen Kale, Sanjiv Kumar, On the convergence of Adam and beyond, in: Proceedings of the 6th International Conference on Learning Representations, ICLR, Vancouver, BC, Canada, April 30 - May 3, 2018, 2018. OpenReview.net.
[33] Mengye Ren, Wenyuan Zeng, Bin Yang, Raquel Urtasun, Learning to reweight examples for robust deep learning, in: Proceedings of the 35th International Conference on Machine Learning, ICML, Stockholm, Sweden, July 10–15, 2018, in: Proceedings of Machine Learning Research, vol. 80, PMLR, 2018, pp. 4331–4340.
[34] Saharon Rosset, Ji Zhu, Trevor Hastie, Margin maximizing loss functions, in: Advances in Neural Information Processing Systems 16: Annual Conference on Neural Information Processing Systems, NIPS, Vancouver and Whistler, BC, Canada, December 8–13, 2003, MIT Press, 2003, pp. 1237–1244.
[35] Robert E. Schapire, The strength of weak learnability, Mach. Learn. 5 (2) (1990) 197–227.
[36] Bernhard Schölkopf, Alexander Johannes Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, Adaptive Computation and Machine Learning Series, MIT Press, 2002.
[37] Hidetoshi Shimodaira, Improving predictive inference under covariate shift by weighting the log-likelihood function, J. Stat. Plan. Inference 90 (2) (2000) 227–244.
[38] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, Rob Fergus, Indoor segmentation and support inference from RGBD images, in: Proceedings of the 12th European Conference on Computer Vision, ECCV, Part V, Florence, Italy, October 7–13, 2012, in: LNCS, vol. 7576, Springer, 2012, pp. 746–760.
[39] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, Ruslan Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958.
[40] Guillaume Stempfel, Liva Ralaivola, Learning SVMs from sloppily labeled data, in: Proceedings of the 19th International Conference on Artificial Neural Networks, ICANN, Part I, Limassol, Cyprus, September 14–17, 2009, in: LNCS, vol. 5768, Springer, 2009, pp. 884–893.
[41] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Boston, MA, USA, June 7-12, 2015, IEEE Computer Society, 2015, pp. 1–9.
[42] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, Zbigniew Wojna, Rethinking the Inception architecture for computer vision, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Las Vegas, NV, USA, June 27–30, 2016, IEEE Computer Society, 2016, pp. 2818–2826.
[43] Theodore B. Trafalis, Robin C. Gilbert, Robust support vector machines for classification and computational issues, Optim. Methods Softw. 22 (1) (2007) 187–198.
[44] Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, Luís Torgo, OpenML: networked science in machine learning, ACM SIGKDD Explor. Newsl. 15 (2) (2013) 49–60.
[45] Vladimir Vapnik, Statistical Learning Theory, John Wiley & Sons, 1998.
[46] Vladimir Vapnik, Akshay Vashist, A new learning paradigm: learning using privileged information, Neural Netw. 22 (5–6) (2009) 544–557.
[47] Oliver Wasenmüller, Didier Stricker, Comparison of Kinect V1 and V2 depth images in terms of accuracy and precision, in: Proceedings of the Asian Conference on Computer Vision, ACCV, International Workshops, Part II, Taipei, Taiwan, November 20–24, 2016, in: LNCS, vol. 10117, Springer, 2016, pp. 34–45.
[48] Frank Wilcoxon, Individual Comparisons by Ranking Methods, Springer, New York, 1992, pp. 196–202.
[49] Yichao Wu, Yufeng Liu, Robust truncated hinge loss support vector machines, J. Am. Stat. Assoc. 102 (479) (2007) 974–983.
[50] Yichao Wu, Yufeng Liu, Adaptively weighted large margin classifiers, J. Comput. Graph. Stat. 22 (2) (2013) 416–432.
[51] Lingxi Xie, Jingdong Wang, Zhen Wei, Meng Wang, Qi Tian, DisturbLabel: regularizing CNN on the loss layer, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Las Vegas, NV, USA, June 27–30, 2016, IEEE Computer Society, 2016, pp. 4753–4762.
[52] Linli Xu, Koby Crammer, Dale Schuurmans, Robust support vector machine training via convex outlier ablation, in: Proceedings of the 21st AAAI Conference on Artificial Intelligence, AAAI, Boston, MA, USA, July 16-20, 2006, vol. 1, AAAI Press, 2006, pp. 536–542.
[53] Xulei Yang, Qing Song, Yue Wang, A weighted support vector machine for data classification, Int. J. Pattern Recognit. Artif. Intell. 21 (5) (2007) 961–976.
[54] Alan L. Yuille, Anand Rangarajan, The concave-convex procedure (CCCP), in: Advances in Neural Information Processing Systems 14: Annual Conference on Neural Information Processing Systems: Natural and Synthetic, NIPS, Vancouver, BC, Canada, December 3-8, 2001, MIT Press, 2001, pp. 1033–1040.
[55] Chaoqiang Zhao, Qiyu Sun, Chongzhen Zhang, Yang Tang, Feng Qian, Monocular depth estimation based on deep learning: an overview, CoRR, arXiv: 2003.06620 [abs], 2020.

[56] Xiaojin Zhu, Andrew B. Goldberg, Introduction to Semi-Supervised Learning, Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool Publishers, 2009.

[57] Daniel Zoran, Phillip Isola, Dilip Krishnan, William T. Freeman, Learning ordinal relationships for mid-level vision, in: Proceedings of the IEEE International Conference on Computer Vision, ICCV, Santiago, Chile, December 7–13, 2015, IEEE Computer Society, 2015, pp. 388–396.

9 Robust Regression for Monocular Depth Estimation

Author Contribution Statement

The idea of explicitly modeling sensor noise for robust monocular depth estimation using superset learning originates from the author, and was further refined by all authors in joint discussions. Eyke Hüllermeier and Ralph Ewerth provided guidance throughout this process. The paper was initially written by the author, and subsequently revised by all authors. Furthermore, the implementation and experimentation was done by the author.

Supplementary Material

An appendix to the paper is provided in Appendix E. The code of the official implementation is provided at `https://github.com/julilien/RobustMDE` (Apache 2.0 license).

Robust Regression for Monocular Depth Estimation

Julian Lienen JULIAN.LIENEN@UPB.DE
Paderborn University, Germany

Nils Nommensen NILS.NOMMENSEN@TIB.EU
Ralph Ewerth RALPH.EWERTH@TIB.EU
L3S Research Center, Leibniz University Hannover and TIB Hannover, Germany

Eyke Hüllermeier EYKE@IFI.LMU.DE
University of Munich (LMU), Germany

Editors: Vineeth N Balasubramanian and Ivor Tsang

Abstract

Learning accurate models for monocular depth estimation requires precise depth annotation as e.g. gathered through LiDAR scanners. Because the data acquisition with sensors of this kind is costly and does not scale well in general, less advanced depth sources, such as time-of-flight cameras, are often used instead. However, these sensors provide less reliable signals, resulting in imprecise depth data for training regression models. As shown in idealized environments, the noise produced by commonly used RGB-D sensors violates standard statistical assumptions of regression methods, such as least squares estimation. In this paper, we investigate whether robust regression methods, which are more tolerant toward violations of statistical assumptions, can mitigate the effects of low-quality data. As a viable alternative to established approaches of that kind, we propose the use of so-called superset learning, where the original data is replaced by (less precise but more reliable) set-valued data. To evaluate and compare the methods, we provide an extensive empirical study on common benchmark data for monocular depth estimation. Our results clearly show the superiority of robust variants over conventional regression.

Keywords: Robust regression, monocular depth estimation, superset learning, data imprecisiation

1. Introduction

In many computer vision applications, such as 3D scene understanding or autonomous driving, the estimation of depth in visual perception is of crucial importance. Often, signals are only observed in the form of monocular images used as input to predict pixel-wise depth. Due to its ill-posed nature, the estimation of depth based on single images is a complex task, which has recently been tackled by machine learning methods, more specifically by deep neural networks trained on large amounts of data samples.

Various data sets provide single images in different scenes along with depth maps gathered from sensors, which are made available as supervision for training monocular depth estimation models. As the acquisition of data with highly accurate depth sensors, e.g., through laser-based LiDAR systems, is costly, most high-volume metric depth data sets were constructed based on less accurate RGB-D sensors, such as infrared (IR) or time-of-flight (TOF) cameras. As a prominent sensor of this kind, Kinect V1 has been employed to construct the widely used NYUD-v2 data set (Silberman et al., 2012), especially for depth in indoor scenes.

Despite their popularity and applicability, data sets constructed with such sensors incorporate a considerable degree of noise. As studied in idealized environments, the distortion of commonly used sensors increases with higher spatial depth (Khoshelham and Elberink, 2012; Wasenmüller and Stricker, 2016; Ahn et al., 2019). While this can also be observed for laser-based sensors (Rosenberger et al., 2018), the problem is especially severe for less sophisticated IR or TOF sensors. As a prominent example, studies analyzing Kinect V1 sensors yield an exponentially increasing standard deviation for higher depth values to be measured, while an increasing offset of the sensed value to the underlying true depth value can be observed (Nguyen et al., 2012; Wasenmüller and Stricker, 2016). Moreover, due to physical properties of the sensors, e.g., interference of emitted rays, the error terms for each individual data term can not be assumed to be independent of other observed signals.

These properties are in conflict with standard statistical model assumptions of conventional regression methods, such as least squares that has also been considered as an optimization criterion in the domain of depth estimation (Carvalho et al., 2018). For instance, it is often assumed to observe noise with constant variance (*homoscedasticity*), and that errors are independent between samples (no *autocorrelation*). Provided such assumptions, traditional methods deliver efficient estimators with several appealing asymptotic guarantees (Dougherty, 2011).

Obviously, these assumptions are violated for most non-synthetic depth estimation data sets. Although several alternatives were suggested to address the aforementioned issues by weaker model assumptions (e.g., as in (Barron, 2019; Irie et al., 2019; Ranftl et al., 2020)), the explicit consideration of robustness in the modeling of monocular depth estimation has received rather little attention so far. This work aims to fill this gap by providing an overview of existing robust regression methods and investigating their effectiveness in the context of depth estimation.

In addition to established methods for robust regression, we also propose to realize the recent idea of "data imprecisiation" to achieve robustness in depth estimation. Here, precise but possibly distorted (biased or noisy) data is turned into imprecise (set-valued) but probably more correct and reliable data, and a model is then trained on the modified data using so-called superset learning (Hüllermeier, 2014).

An exhaustive empirical evaluation demonstrates the effectiveness of robust variants over conventional regression methods on popular depth estimation benchmarks, and especially confirms the adequacy of the superset modelling approach in cases of erroneous and misleading training information.

2. Robust Regression

In this section, we survey related work on robust regression, specifically focusing on losses that have been used in the domain of depth estimation.

2.1. Standard Regression Methods

In the setting of regression, one commonly assumes a stochastic dependency of the form $y = f(\boldsymbol{x}) + \epsilon$, i.e., samples $y \in \mathcal{Y} = \mathbb{R}$ of the target (output) variable are functions of the input (instance) $\boldsymbol{x} \in \mathcal{X}$ afflicted with (additive) random errors $\varepsilon \in \mathbb{R}$. In the context of depth estimation, instances $\boldsymbol{x}$ could be descriptions of the pixels of an image, and outputs y the corresponding depth values. Given training data in the form of a set of input/output pairs $(\boldsymbol{x}_i, y_i)$, $i = 1, \dots, n$, the task is to learn a function $\hat{f} : \mathcal{X} \rightarrow \mathcal{Y}$ that allows for predicting the target value for any query instance given as an input. Typically, this is accomplished by finding a function that minimizes a certain loss $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$ on the training data, i.e., that minimizes the training error $\sum_{i=1}^{n} \mathcal{L}(y_i, \hat{f}(\boldsymbol{x}_i))$ (or a regularized version of this error).

If such assumptions are violated in practice, and outcomes are observed with a high degree of noise, one may want to weaken the assumptions, leading to more robust estimators. Long-standing research has been conducted on achieving robustness in classical statistics, leading to several approaches that improve estimation performance on data that does not comply with strong model assumptions. For instance, generalized versions of least squares regression have been suggested to cope with heteroscedasticity (Kariya and Kurata, 2004), e.g., by weighting the residuals according to the inverse of the variance of the error. Likewise, alternative loss functions, for example the absolute ($\mathcal{L}_1$) instead of the squared ($\mathcal{L}_2$) error, have been considered to alleviate sensitivity to outliers. Often, however, the minimization of such losses comes with other issues of practical relevance, such as unstable or ambiguous solutions (Dodge, 1987).

A famous class of extremum estimation methods are the so-called M-estimators (Huber, 1981), which generalize the idea of maximum likelihood estimation by providing an interface to inject more robust cost functions as optimization criterion. As one of such functions, Huber (Huber, 1981) introduced a robust loss that combines the squared and the absolute loss to diminish the sensitivity to outliers.

2.2. Robustness in Depth Estimation

In the domain of (supervised) monocular depth estimation, a plethora of different loss functions has been suggested to induce regression models, ranging from $\mathcal{L}_1$- (Ma and Karaman, 2018; Ranftl et al., 2020) and $\mathcal{L}_2$-based (Carvalho et al., 2018; Ranftl et al., 2020) losses to model-specific measures (Kendall and Gal, 2017; Wu et al., 2019; Bhat et al., 2021). Also, several loss augmentations have been proposed, e.g., to consider smoothness in the prediction (Li and Snavely, 2018) or to treat targets in a different representation (Fu et al., 2018; Li and Snavely, 2018). Although ablation studies often compare loss functions and their effects (e.g., as in (Carvalho et al., 2018; Ranftl et al., 2020)), to the best of our knowledge, an explicit investigation of the robustness of losses in the context of depth estimation is still missing.

As one of the earlier approaches to achieve robustness, the previously mentioned Huber-loss has been applied in the domain of depth estimation, although in a reversed form (Laina et al., 2016; Carvalho et al., 2018). Its original (robust) form as used for depth estimation is given by

$$\mathcal{L}_{\text{Huber}}(y, \hat{y}) := \begin{cases} \frac{(y-\hat{y})^2 + c^2}{2c} & \text{if } |y - \hat{y}| \leq c \\ |y - \hat{y}| & \text{otherwise} \end{cases} , \tag{1}$$

where y is the observed value, $\hat{y}$ the model prediction, and the parameter c is typically defined as 20% of the maximum residual in each batch calculation. The Huber loss inherits

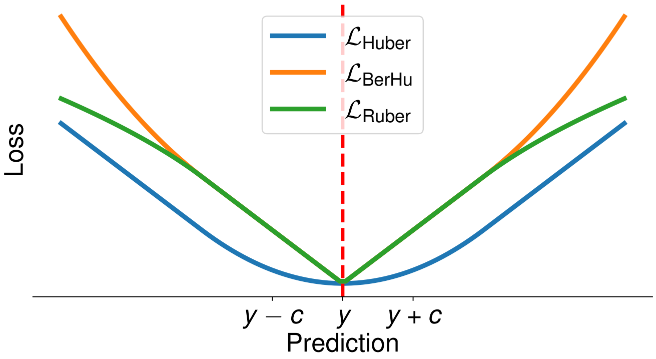


Figure 1: Variants of the Huber loss as used in the domain of monocular depth estimation.

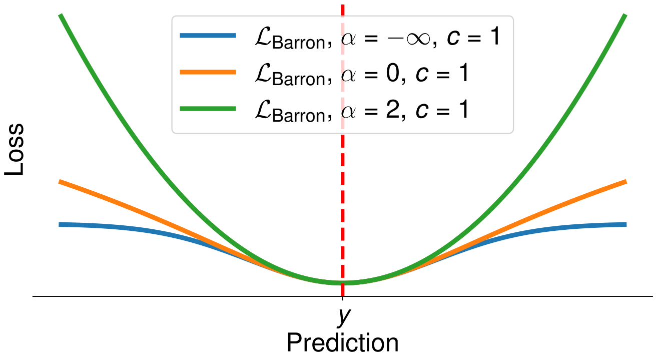


Figure 2: Special cases of $\mathcal{L}_{\text{Barron}}$ as presented in (Barron, 2019).

the advantage of $\mathcal{L}_1$ to deemphasize the influence of outliers while overcoming the non-differentiability of this loss at zero. As the more popular method in the depth estimation domain, let us denote the BerHu loss as the reversed version of $\mathcal{L}_{\text{Huber}}$ by $\mathcal{L}_{\text{BerHu}}$.

The loss formulation has also been adopted by smoothening the $\mathcal{L}_1$ part for further robustness, leading to the so-called Ruber loss (Irie et al., 2019), which is defined as

$$\mathcal{L}_{\text{Ruber}}(y, \hat{y}) := \begin{cases} |y - \hat{y}| & \text{if } |y - \hat{y}| \leq c \\ \sqrt{2c|y - \hat{y}| - c^2} & \text{otherwise} \end{cases} . \tag{2}$$

In their work, the authors show improved robustness, along with the optimization of the parameter c in a data-driven manner. Fig. 1 illustrates the Huber-like losses as used within the domain of depth estimation.

As one loss coming from a related field, namely flow prediction, the so-called "generalized Charbonnier" loss (Sun et al., 2010) with a smoothed $\mathcal{L}_1$ loss term as special case showed promising robustness properties for the problem of depth estimation (Chen and Koltun, 2014). Closely related to this, Barron (2019) suggests a more expressive robust loss variant, which even extends the Charbonnier loss. It is given by

$$\mathcal{L}_{\text{Barron}}(y, \hat{y}) := \frac{|\alpha - 2|}{\alpha} \left(\left(\frac{((y - \hat{y})/c)^2}{|\alpha - 2|} + 1 \right)^{\alpha/2} - 1 \right) , \tag{3}$$

where $\alpha \in \mathbb{R}$ and $c \in \mathbb{R}_+$ are hyperparameters to control the robustness and scale respectively. Special cases of the loss are depicted in Fig. 2. Interestingly, this loss delivers the generalized Charbonnier loss, as well as $\mathcal{L}_2$ and a smoothed version of $\mathcal{L}_1$ as special cases.

Current state-of-the-art methods often employ a scale-invariant version of the $\mathcal{L}_2$ loss in log-space (Eigen et al., 2014), which, for a set of n observations $y_1, \ldots, y_n$, is given by

$$\mathcal{L}_{\text{SIError}}(y, \hat{y}) := \frac{1}{n} \sum_{i=1}^{n} g_i^2 - \frac{\lambda}{n^2} \left(\sum_{i=1}^{n} g_i \right)^2 , \tag{4}$$

where g_i is the residual of the i^{th} instance in log space, i.e., $g_i = \log y_i - \log y_i^*$ and $\lambda \in [0, 1]$ is a hyperparameter. This variant has further been augmented by an additional scaling parameter α (Lee et al., 2019; Bhat et al., 2021). We refer to the scaled variant of this loss as $\mathcal{L}_{\text{ScaledSIError}}$. Although not specifically designed to cope with outliers, its depth interpretation in log space diminishes the severity of heteroscedasticity in least squares optimization, and it has been shown to yield state-of-the-art generalization performance (Bhat et al., 2021).

As another robust loss formulation, this time applied in the disparity space, Ranftl et al. (2020) propose a loss variant that trims an $\mathcal{L}_1$ loss by disregarding the 20% largest residuals in each image, which we refer to as $\mathcal{L}_{\text{trim}}$. This is in contrast to M-estimators as the weighted least squares method, where residuals with a high variance are down-weighted.

3. Superset Learning

As an alternative to cope with low-quality data, we advocate the idea of “data imprecisiation”, which in turn is grounded in the framework of superset learning. In the following, we give a brief introduction to superset learning in general, followed by two concrete proposals for robust depth estimation.

3.1. Background on Superset Learning

Recall that, in learning a depth estimator given images with their corresponding depth maps, one typically considers pixels as individual training instances attached with single values from a target space $\mathcal{Y}$, in the case of depth regression usually with $\mathcal{Y} = \mathbb{R}_+$. Given this ground truth data, the task is to learn a model (hypothesis) predicting values $\hat{y} \in \mathcal{Y}$ that fit the training data as much as possible (but not too much to avoid overfitting). To measure the optimality of the prediction, losses of the form $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ are employed, like those presented before.

In superset learning, we consider the case where data is not necessarily observed precisely. Instead of precise outcomes $y \in \mathcal{Y}$ provided as supervision, we only assume that *subsets* $Y \subseteq \mathcal{Y}$ of the output space are given as training information. Thus, a single observation is of the form $(\boldsymbol{x}, Y) \in \mathcal{X} \times 2^{\mathcal{Y}}$. The set-valued data is supposed to cover the underlying precise but unobserved data in the sense that $y \in Y$ (hence the name "superset learning").[1]

Provided data of that kind, Hüllermeier (2014) proposed an approach to superset learning motivated by the idea of performing model identification and "data disambiguation" at the same time. To this end, the underlying loss function $\mathcal{L} : \mathcal{Y}^2 \rightarrow \mathbb{R}_+$ is extended to the *optimistic superset loss* (OSL) $\mathcal{L}^* : 2^{\mathcal{Y}} \times \mathcal{Y} \longrightarrow \mathbb{R}_+$ defined by the map

$$(Y, \hat{y}) \mapsto \min \big\{ \mathcal{L}(y, \hat{y}) \,|\, y \in Y \big\}. \tag{5}$$

More recently, the same loss has also been introduced under the notion of *infimum loss* (Cabannes et al., 2020). Superset learning then seeks to perform generalized risk minimization, i.e., to minimize the OLS loss (or a regularized version thereof) instead of the original loss $\mathcal{L}$ on the training data.

3.2. From Set-valued to Fuzzy Data

The OSL (5) can be generalized further to the case where data is characterized in terms of fuzzy sets (Klir and Folger, 1988). The latter generalize conventional sets in the sense of allowing gradual membership of elements, where the degree of membership is typically specified in terms of a real number in the unit interval. Thus, a fuzzy subset $\tilde{Y}$ of $\mathcal{Y}$ can be identified with a membership function of the form $\tilde{Y} : \mathcal{Y} \longrightarrow [0, 1]$, where $\tilde{Y}(y) = 1$ indicates full membership of y, $\tilde{Y}(y) = 0$ no membership, and $0 < \tilde{Y}(y) < 1$ that y belongs to the fuzzy set to a certain degree (Klir and Folger, 1988). The fuzzy-version of the OSL loss (which we refer to as FOSL) is obtained as a generalization of $\mathcal{L}^*$, using a reduction scheme based on a standard level-cut representation of fuzzy sets:

$$\begin{aligned} \mathcal{L}^{**} : \mathcal{F}(\mathcal{Y}) \times \mathcal{Y} &\longrightarrow \mathbb{R}_+ \,, \\ (\tilde{Y}, \hat{y}) &\mapsto \int_0^1 \mathcal{L}^* \left([\tilde{Y}]_\alpha, \hat{y} \right) \, d\alpha \,, \end{aligned} \tag{6}$$

where $\mathcal{F}(\mathcal{Y})$ denotes the set of all fuzzy subsets of $\mathcal{Y}$ and $[\tilde{Y}]_\alpha := \{ y \,|\, \tilde{Y}(y) \geq \alpha \}$ is the α-cut of $\tilde{Y}$.

3.3. Data Imprecisiation

In addition to learning from genuinely imprecise data, the framework of superset learning can also be used for learning from standard (precise) data, which — via a process of "imprecisiation" — is deliberately turned into imprecise data (Hüllermeier, 2014). Different effects can be achieved in this way. In particular, data imprecisiation offers a means to control the influence of individual observations on the overall result of the learning process: the more imprecise an observation is made, the less it will influence the model induced from the data (Lienen and Hüllermeier, 2021).

Indeed, the optimistic superset loss (5), and likewise the fuzzy version (6), is a relaxation of the original loss $\mathcal{L}$ in the sense that $\mathcal{L}^* \leq \mathcal{L}$. More specifically, the larger the set Y, the smaller the loss: $Y \supseteq Y'$ implies $\mathcal{L}^*(Y, \hat{y}) \leq \mathcal{L}^*(Y', \hat{y})$ for all $\hat{y} \in \mathcal{Y}$. Thus, the loss

[1] Note that the precise data y may already be corrupted with noise.

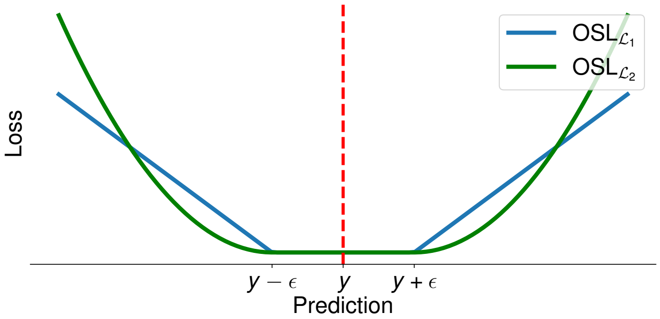


Figure 3: ϵ-insensitive OSL variants for interval data.

$\mathcal{L}(y, \hat{y})$ incurred for a prediction $\hat{y}$ can be weakened by replacing the original observation y with a (fuzzy) subset around y, and the larger the subset, the smaller the loss. Therefore, "imprecisiating" a data point by replacing the original (precise) observation y with a (fuzzy) set-valued outcome Y can be seen as a means for reducing the influence of possibly noisy or unreliable data, and hence for making learning more robust. In the following, we shall discuss two concrete approaches of that kind in the context of regression for depth estimation.

3.4. Interval Data

As already said, the values produced by depth sensors are often quite noisy, and the assumptions of a precise noise model do normally not apply. A somewhat crude but robust alternative is to model the information about the underlying true depth in terms of a tolerance interval around the precise measurement y. Thus, the learning algorithm is merely provided with the information that the sought depth is most likely an element of this interval. Depending on the length of the interval, this information might be relatively weak.

More importantly, however, it is also most likely *correct*. Therefore, compared to more precise but presumably wrong information, it is less likely to bias the learner in a wrong direction. In fact, because the loss is 0 as long as the learner predicts any value inside the interval, it is completely free to choose the value that appears most plausible (in light of the other observations and its underlying model assumptions), without incurring any penalty. As confirmed by empirical studies (Cabannes et al., 2020), this provides the learner with an opportunity to disambiguate the data and increases robustness toward misleading observations.

More specifically, we model the data in terms of ϵ-intervals $Y = [y-\epsilon, y+\epsilon]$. Interestingly, we thus establish a close connection to the well-known method of support vector regression (SVR) (Schölkopf and Smola, 2001). In fact, the OSL extension of the $\mathcal{L}_1$ loss obtained for data of that kind exactly coincides with the ϵ-insensitive loss used in SVR. Fig. 3 depicts this loss as well as the OSL extension of the $\mathcal{L}_2$ loss.

In the approach realized in this paper, all intervals are centered around the original observations and share the same length 2ϵ. We consider ϵ as a hyperparameter that is tuned on a validation set. Let us note, however, that intervals could in principle also be customized for each observation individually. This way, different types of domain knowledge

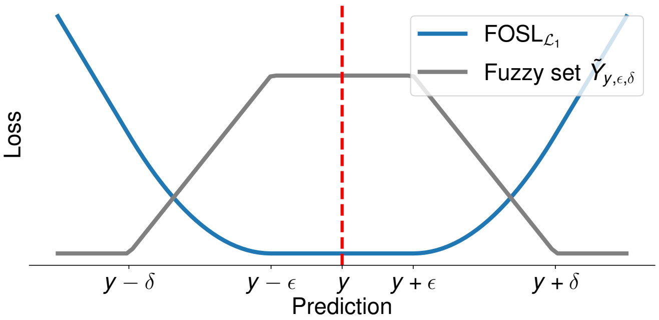


Figure 4: The FOSL variant based on $\mathcal{L}_1$ for a trapezoidal fuzzy superset $\tilde{Y}_{y,\epsilon,\delta}$.

could be incorporated, for example that measurements in a certain region of an image are more reliable than in another region, or that some measurement have a stronger tendency to over- than to underestimate the true depth.

3.5. Fuzzy data

Going beyond a distinction between plausible and implausible values, as purported by an interval, fuzzy sets allow for modeling data in a more elaborate way. As an interesting special case, the Huber loss is reproduced as the OSL-extension of the $\mathcal{L}_1$ loss when replacing precise measurements y by *triangular* fuzzy sets $\tilde{Y}_{y,\delta}(z) = \max\{0, 1 - |y-z|/\delta\}$.

Even more appropriate for the case of robust depth estimation is the FOSL loss obtained for *trapezoidal* fuzzy sets of the form

$$\tilde{Y}_{y,\epsilon,\delta}(z) = \begin{cases} \frac{z-y+\delta}{\delta-\epsilon} & \text{if } y-\delta \leq z \leq y-\epsilon \\ 1 & \text{if } y-\epsilon \leq z \leq y+\epsilon \\ \frac{y+\delta-z}{\delta-\epsilon} & \text{if } y+\epsilon \leq z \leq y+\delta \\ 0 & \text{otherwise} \end{cases}, \tag{7}$$

which combine attenuation properties of the Huber-loss with the relaxation effects produced by the ϵ-insensitivity of the SVR loss. More specifically, by incorporating $\tilde{Y}_{y,\epsilon,\delta}$ in (5) we obtain

$$\mathcal{L}^{**}\left(\tilde{Y}_{y,\epsilon,\delta}, \hat{y}\right) := \begin{cases} 0 & \text{if } \hat{y} \in [y-\epsilon, y+\epsilon] \\ \frac{(y\pm\epsilon-\hat{y})^2}{2(\delta-\epsilon)} & \text{if } \hat{y} \in (y\pm\delta, y\pm\epsilon) \\ |y\pm\epsilon-\hat{y}| - \frac{\delta-\epsilon}{2} & \text{otherwise} \end{cases}, \tag{8}$$

where $\epsilon, \delta \in \mathbb{R}_+$ with $\epsilon \geq \delta$ are hyperparameters. Similar to the interval-based loss, ϵ and δ can be optimized on validation data. Fig. 4 shows the resulting loss function.

4. Evaluation

To demonstrate the effectiveness of robust methods for depth estimation, we conduct an extensive empirical evaluation on common indoor benchmark data. First, we give an overview over the data, baselines, metrics, and implementation details, followed by the presentation of the results.

4.1. Experimental Settings

4.1.1. Datasets

In our studies, we consider two sources for training a depth predictor. First, we use *NYUD-v2* (Silberman et al., 2012) as a homogeneous indoor[2] data set based on the Kinect V1 sensor, which has been studied broadly and for which approximations of the sensor noise are provided (e.g., as in (Nguyen et al., 2012; Wasenmüller and Stricker, 2016)). Second, as a data set that unifies multiple sources with individual error terms, we consider *SunRGBD* (Song et al., 2015) as an additional heterogeneous source to learn from. This data set uses four different sensors, namely Kinect V1, Kinect V2, RealSense, and Xtion (cf. (Song et al., 2015) for more detailed descriptions).

For *NYUD-v2*, we use a subset of 10k preprocessed instances as also used in (Bhat et al., 2021). For training, we rescale each input image and depth map to the size of 224×224, while we evaluate on the Eigen split of 654 test samples using the commonly applied cropping in the original resolution (480 × 640).

To train models on *SunRGBD*, we use the original training and test splits as provided by the authors of the data set. While the training set consists of $10,355$ indoor RGB-D images, the test split comprises 2860 images. The resolutions are kept the same as for *NYUD-v2*.

Since both data sets involve noisy depth sensors, models reconstructing the sensor noise observed in the training data benefit from the evaluation on the corresponding test sets when constructed on the same base. Rather, we aim to measure the model performances on the basis of highly accurate signals. To this end, we evaluate the induced models on the LiDAR-based dataset *iBims-1* (Koch et al., 2018) and *DIODE* (Vasiljevic et al., 2019).

iBims-1 makes use of a digital single-lens reflex camera attached with a high-precision laser scanner to acquire images along with their pixel-wise depth, approximately matching the depth value distribution of *NYUD-v2*. The data set consists of 100 indoor RGB-D image of resolution 480 × 640. Within our studies, we use this data set as validation set to optimize model hyperparameters.

For the final model assessment, we use the provided indoor validation set of *DIODE*, consisting of 335 high-quality RGB-D images of resolution 768 × 1024, which provides a diverse set of indoor scenes used to measure the generalization performance of the assessed models. To compute metrics on the test data, we upscaled all model predictions to the original size.

4.1.2. Baselines

As baselines, we consider the loss functions discussed before. That is, we depart from $\mathcal{L}_1$ and $\mathcal{L}_2$ as the most obvious choices to train regression models. Beyond that, as used within the domain of depth estimation, we consider the Huber-loss variants $\mathcal{L}_{\text{Huber}}$, $\mathcal{L}_{\text{BerHu}}$, and $\mathcal{L}_{\text{Ruber}}$. Moreover, $\mathcal{L}_{\text{Barron}}$ and $\mathcal{L}_{\text{trim}}$ as losses explicitly approaching robustness are also included. As a loss used to train current SOTA models, we further evaluate models learned with $\mathcal{L}_{\text{ScaledSIError}}$.

[2] Here, we focus on indoor data sets as the test data in such scenes is usually more precise compared to outdoor scenery.

Apart from that, in order to seek to improve conventional $\mathcal{L}_2$ optimization, we apply the weighted least squares criterion, which we refer to as $\mathcal{L}_{\text{WeightedL2}}$. Here, we use an approximation of the standard deviation of the Kinect V1 sensors as provided in (Nguyen et al., 2012), namely $\sigma(x) = 0.0012 + 0.0012(x - 0.4)^2$ for weighting.

To demonstrate the effectiveness of the superset modelling approaches, we provide results for both the interval- and fuzzy set-based modelling approach. For the former, we investigate variants based on $\mathcal{L}_1$ and $\mathcal{L}_2$, denoted by $\text{OSL}_{\mathcal{L}_1}$ and $\text{OSL}_{\mathcal{L}_2}$, respectively. For the latter, we consider the FOSL variant on the basis of $\mathcal{L}_1$ as $\text{FOSL}_{\mathcal{L}_1}$.

4.1.3. Metrics

In order to measure the performance of the individual models, we present the results for 6 regression methods as commonly reported in the field of depth estimation. The error metrics are defined for ground truth depth values $y \in \mathbb{R}_+$ and model predictions $\hat{y} \in \mathbb{R}_+$ for an image as follows:

- Absolute relative error (REL): $\frac{1}{n}\sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{y_i}$
- Average $\log_{10}$ error: $\frac{1}{n}\sum_{i=1}^{n} |\log_{10}(y_i) - \log_{10}(\hat{y}_i)|$
- Root mean squared error (RMS): $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$
- Threshold accuracies δ_i: percentage of $\hat{y}$ s.t. $\max\left(\frac{y}{\hat{y}}, \frac{\hat{y}}{y}\right) = \delta < 1.25^i$

The final results are averaged over all test images. We provide results for more metrics in the supplement.

4.1.4. Implementation Details

For our experiments, we use a simple U-Net architecture employing an EfficientNetB0 encoder pretrained on ImageNet. For the decoder part, we use a stack of repeating convolutional, BatchNormalization, ReLU, and bilinear upsampling layers. In total, the model comprises approximately 15 million parameters and is kept the same across all experiments.

To provide a fair comparison of all losses incorporating several hyperparameters, we optimized hyperparameters for both the optimizer (Adam in our case) and the individual losses within a random search with 20 trials. Each model is trained for 25 epochs with a batch size of 16. As mentioned before, *iBims-1* was used to calculate the validation scores. The model providing the lowest validation score throughout the runs was considered for the final testing. For statistical significance of the results, we conducted each experiment three times with different seeds.

To allow for reproducing our results, a more comprehensive overview about implementation details and a detailed model description is provided in the supplement.

4.2. Homogeneous Depth Sensor: NYUD-v2

In the first experiment, we assess models trained on subsets of *NYUD-v2*. As discussed before, this data set annotated by Kinect V1 depths incorporates a relatively high degree of noise and violates classical statistical assumptions. To assess the robustness of the different

Table 1: Averaged results and standard deviations on models trained on subsets of various sizes of *NYUD-v2* on *DIODE* and *NYUD-v2*. The best results indicated in bold per number of instances and metric.

# Insts.	Loss	*DIODE*						*NYUD-v2*		
		REL (↓)	$\log_{10}$ (↓)	RMS (↓)	δ_1 (↑)	δ_2 (↑)	δ_3 (↑)	REL (↓)	RMS (↓)	δ_1 (↑)
2k	$\mathcal{L}_2$	0.492 ± 0.030	0.223 ± 0.001	1.839 ± 0.015	0.316 ± 0.007	0.547 ± 0.006	0.703 ± 0.007	0.375 ± 0.045	1.015 ± 0.091	0.463 ± 0.039
	$\mathcal{L}_1$	0.463 ± 0.025	0.228 ± 0.001	1.891 ± 0.035	0.306 ± 0.005	0.534 ± 0.001	0.694 ± 0.002	0.327 ± 0.024	0.934 ± 0.043	0.512 ± 0.014
	$\mathcal{L}_{\text{Huber}}$	0.433 ± 0.021	0.230 ± 0.016	1.873 ± 0.084	0.293 ± 0.031	0.542 ± 0.030	0.703 ± 0.031	0.281 ± 0.004	0.826 ± 0.005	0.554 ± 0.003
	$\mathcal{L}_{\text{BerHu}}$	0.440 ± 0.016	0.225 ± 0.003	1.854 ± 0.012	0.304 ± 0.008	0.548 ± 0.009	0.713 ± 0.003	0.284 ± 0.017	0.851 ± 0.021	0.553 ± 0.019
	$\mathcal{L}_{\text{Ruber}}$	0.434 ± 0.008	0.232 ± 0.008	1.878 ± 0.034	0.297 ± 0.016	0.536 ± 0.022	0.700 ± 0.022	0.285 ± 0.024	0.835 ± 0.044	0.571 ± 0.021
	$\mathcal{L}_{\text{Barron}}$	0.450 ± 0.010	0.224 ± 0.002	1.850 ± 0.018	0.313 ± 0.010	0.553 ± 0.005	0.719 ± 0.006	0.310 ± 0.024	0.883 ± 0.057	0.531 ± 0.038
	$\mathcal{L}_{\text{trim}}$	0.451 ± 0.026	0.229 ± 0.006	1.878 ± 0.026	0.328 ± 0.004	0.553 ± 0.001	0.702 ± 0.008	0.362 ± 0.034	1.045 ± 0.151	0.481 ± 0.015
	$\mathcal{L}_{\text{ScaledSIError}}$	**0.427** ± 0.001	0.225 ± 0.007	1.825 ± 0.024	0.300 ± 0.020	0.554 ± 0.019	**0.721** ± 0.009	**0.258** ± 0.024	**0.763** ± 0.046	**0.613** ± 0.031
	$\mathcal{L}_{\text{WeightedL2}}$	0.486 ± 0.026	0.221 ± 0.001	1.837 ± 0.007	0.320 ± 0.007	0.551 ± 0.003	0.708 ± 0.003	0.371 ± 0.035	1.007 ± 0.068	0.465 ± 0.025
	$\text{OSL}_{\mathcal{L}_1}$	0.454 ± 0.028	**0.216** ± 0.005	**1.803** ± 0.028	**0.332** ± 0.017	**0.562** ± 0.014	0.712 ± 0.006	0.325 ± 0.035	0.867 ± 0.062	0.532 ± 0.031
	$\text{OSL}_{\mathcal{L}_2}$	0.472 ± 0.057	0.219 ± 0.007	1.815 ± 0.032	0.317 ± 0.020	0.548 ± 0.016	0.703 ± 0.012	0.361 ± 0.057	0.981 ± 0.093	0.495 ± 0.037
	$\text{FOSL}_{\mathcal{L}_1}$	0.448 ± 0.005	0.229 ± 0.008	1.875 ± 0.045	0.302 ± 0.018	0.535 ± 0.016	0.701 ± 0.005	0.282 ± 0.012	0.832 ± 0.025	0.561 ± 0.016
10k	$\mathcal{L}_2$	0.446 ± 0.007	0.227 ± 0.004	1.859 ± 0.011	0.307 ± 0.008	0.545 ± 0.006	0.706 ± 0.006	0.301 ± 0.015	0.876 ± 0.025	0.525 ± 0.003
	$\mathcal{L}_1$	0.432 ± 0.004	0.228 ± 0.012	1.851 ± 0.052	0.308 ± 0.019	0.548 ± 0.023	0.709 ± 0.020	0.252 ± 0.019	0.741 ± 0.035	0.625 ± 0.018
	$\mathcal{L}_{\text{Huber}}$	0.441 ± 0.008	0.231 ± 0.012	1.868 ± 0.041	0.313 ± 0.017	0.554 ± 0.018	0.713 ± 0.012	0.260 ± 0.004	0.754 ± 0.026	0.628 ± 0.011
	$\mathcal{L}_{\text{BerHu}}$	0.431 ± 0.002	0.229 ± 0.003	1.857 ± 0.010	0.314 ± 0.005	0.554 ± 0.004	0.714 ± 0.004	0.222 ± 0.005	0.688 ± 0.012	0.672 ± 0.010
	$\mathcal{L}_{\text{Ruber}}$	0.427 ± 0.013	0.226 ± 0.002	1.843 ± 0.004	0.311 ± 0.005	0.553 ± 0.005	0.721 ± 0.006	0.231 ± 0.015	0.690 ± 0.024	0.664 ± 0.025
	$\mathcal{L}_{\text{Barron}}$	0.458 ± 0.012	0.226 ± 0.009	1.857 ± 0.040	0.304 ± 0.020	0.545 ± 0.019	0.708 ± 0.016	0.289 ± 0.032	0.815 ± 0.060	0.569 ± 0.043
	$\mathcal{L}_{\text{trim}}$	0.430 ± 0.011	0.234 ± 0.005	1.880 ± 0.020	0.290 ± 0.014	0.537 ± 0.015	0.701 ± 0.012	0.247 ± 0.026	0.747 ± 0.069	0.615 ± 0.043
	$\mathcal{L}_{\text{ScaledSIError}}$	**0.411** ± 0.010	0.237 ± 0.011	1.875 ± 0.045	0.301 ± 0.027	0.546 ± 0.029	0.713 ± 0.019	**0.196** ± 0.003	**0.649** ± 0.018	**0.702** ± 0.011
	$\mathcal{L}_{\text{WeightedL2}}$	0.433 ± 0.013	0.225 ± 0.002	1.846 ± 0.016	0.314 ± 0.005	0.550 ± 0.009	0.711 ± 0.009	0.278 ± 0.016	0.811 ± 0.033	0.564 ± 0.022
	$\text{OSL}_{\mathcal{L}_1}$	0.417 ± 0.009	0.211 ± 0.002	1.771 ± 0.012	0.334 ± 0.011	0.579 ± 0.008	0.735 ± 0.001	0.279 ± 0.010	0.784 ± 0.017	0.618 ± 0.007
	$\text{OSL}_{\mathcal{L}_2}$	0.423 ± 0.007	**0.208** ± 0.003	**1.757** ± 0.021	**0.339** ± 0.009	**0.582** ± 0.006	**0.736** ± 0.006	0.305 ± 0.020	0.841 ± 0.024	0.558 ± 0.010
	$\text{FOSL}_{\mathcal{L}_1}$	0.413 ± 0.008	0.233 ± 0.013	1.876 ± 0.058	0.331 ± 0.021	0.557 ± 0.027	0.716 ± 0.030	0.229 ± 0.021	0.718 ± 0.032	0.662 ± 0.014

losses, we perform a cross-data set generalization study: While training on the noisy *NYUD-v2* data, we measure the performance of the models on the high-quality *DIODE* data set with the help of *iBims-1* as validation data in the hyperparameter optimization. Thereby, we consider varying amounts of training data being used to investigate the effect of more instances that might provide more stable estimates. Along with that, we further report the results on the Eigen test split for comparison. However, one notes that these test examples are gathered in the same way as the training data and thus incorporate the same noise that we approach to dump.

As can be seen in Table 1, the scaled SI error outperforms the other losses with regard to the *NYUD-v2* test data. However, when considering the cleaner *DIODE* benchmark data, the scaled SI loss shows less robust behavior, often not even improving baselines such as $\mathcal{L}_2$ itself. This demonstrates the inappropriateness to assess depth models on such noisy benchmark data.

On the contrary, most of the more robust loss variants improve over the conventional $\mathcal{L}_2$ loss, especially, when there is more training data provided. Notably, with only few exceptions, the superset loss variants outperform the baselines $\mathcal{L}_2$ and $\mathcal{L}_1$ in almost all cases, often even significantly. $\text{OSL}_{\mathcal{L}_1}$ turns out to work reasonably well when a small number of instances is provided, whereas $\text{OSL}_{\mathcal{L}_2}$ improves over the other methods for higher numbers of instances. Interestingly, albeit comprising it as a special case, $\text{FOSL}_{\mathcal{L}_1}$ often provides slightly inferior performance compared to $\text{OSL}_{\mathcal{L}_1}$. As the latter involves two loss-specific hyperparameters, this is most likely because of misleading draws in the hyperparameter optimization due to the larger hyperparameter space. While this shows the appealing property of $\text{OSL}_{\mathcal{L}_1}$ only having a single parameter to tune, spending more computational budget could leverage the increasing expressiveness of $\text{FOSL}_{\mathcal{L}_1}$ further.

Table 2: Averaged results and standard deviations of models trained on 2k instances from *NYUD-v2* on *DIODE* for varying noise levels. As before, best results per noise level and metric are indicated in bold.

Noise $\hat{\epsilon}$	Loss	REL (↓)	$\log_{10}$ (↓)	RMS (↓)	δ_1 (↑)	δ_2 (↑)	δ_3 (↑)
0.5	$\mathcal{L}_2$	0.851 ± 0.071	0.258 ± 0.013	2.090 ± 0.107	0.180 ± 0.022	0.398 ± 0.036	0.622 ± 0.033
	$\mathcal{L}_1$	0.477 ± 0.022	0.226 ± 0.003	1.859 ± 0.011	0.314 ± 0.010	0.541 ± 0.011	0.695 ± 0.011
	$\mathcal{L}_{\text{Huber}}$	0.576 ± 0.079	0.221 ± 0.008	1.847 ± 0.057	0.294 ± 0.038	0.540 ± 0.036	0.698 ± 0.016
	$\mathcal{L}_{\text{BerHu}}$	0.448 ± 0.003	0.220 ± 0.006	1.832 ± 0.041	**0.325** ± 0.006	0.558 ± 0.005	0.716 ± 0.008
	$\mathcal{L}_{\text{Ruber}}$	0.440 ± 0.009	0.225 ± 0.004	1.854 ± 0.014	0.315 ± 0.014	0.544 ± 0.008	0.706 ± 0.004
	$\mathcal{L}_{\text{Barron}}$	0.707 ± 0.045	0.234 ± 0.007	1.913 ± 0.050	0.233 ± 0.019	0.474 ± 0.030	0.671 ± 0.015
	$\mathcal{L}_{\text{trim}}$	0.461 ± 0.006	0.271 ± 0.004	1.874 ± 0.012	0.273 ± 0.003	0.503 ± 0.003	0.680 ± 0.002
	$\mathcal{L}_{\text{ScaledSIError}}$	0.606 ± 0.147	0.471 ± 0.221	2.512 ± 0.426	0.127 ± 0.101	0.252 ± 0.193	0.370 ± 0.259
	$\mathcal{L}_{\text{WeightedL2}}$	0.731 ± 0.009	0.241 ± 0.002	1.955 ± 0.013	0.225 ± 0.007	0.453 ± 0.007	0.661 ± 0.004
	$\text{OSL}_{\mathcal{L}_1}$	0.438 ± 0.037	**0.217** ± 0.004	**1.806** ± 0.025	0.323 ± 0.004	**0.561** ± 0.007	**0.719** ± 0.008
	$\text{OSL}_{\mathcal{L}_2}$	0.782 ± 0.072	0.243 ± 0.012	1.999 ± 0.083	0.201 ± 0.026	0.428 ± 0.038	0.649 ± 0.029
	$\text{FOSL}_{\mathcal{L}_1}$	**0.425** ± 0.015	0.224 ± 0.006	1.848 ± 0.043	0.310 ± 0.014	0.551 ± 0.012	0.709 ± 0.006
1.0	$\mathcal{L}_2$	1.466 ± 0.216	0.344 ± 0.039	3.167 ± 0.106	0.136 ± 0.042	0.279 ± 0.079	0.457 ± 0.089
	$\mathcal{L}_1$	0.482 ± 0.012	0.217 ± 0.002	1.814 ± 0.010	0.337 ± 0.006	**0.565** ± 0.005	0.707 ± 0.006
	$\mathcal{L}_{\text{Huber}}$	1.036 ± 0.039	0.282 ± 0.005	2.300 ± 0.021	0.142 ± 0.013	0.333 ± 0.017	0.566 ± 0.008
	$\mathcal{L}_{\text{BerHu}}$	0.484 ± 0.029	**0.216** ± 0.003	1.833 ± 0.020	0.322 ± 0.005	0.554 ± 0.006	0.712 ± 0.003
	$\mathcal{L}_{\text{Ruber}}$	0.453 ± 0.014	0.222 ± 0.000	1.847 ± 0.011	0.313 ± 0.010	0.554 ± 0.001	0.717 ± 0.005
	$\mathcal{L}_{\text{Barron}}$	1.063 ± 0.127	0.289 ± 0.018	2.367 ± 0.157	0.135 ± 0.024	0.320 ± 0.041	0.549 ± 0.040
	$\mathcal{L}_{\text{trim}}$	0.547 ± 0.018	0.366 ± 0.002	1.955 ± 0.008	0.213 ± 0.008	0.315 ± 0.010	0.489 ± 0.001
	$\mathcal{L}_{\text{ScaledSIError}}$	0.654 ± 0.066	0.460 ± 0.184	2.546 ± 0.387	0.128 ± 0.119	0.240 ± 0.204	0.341 ± 0.253
	$\mathcal{L}_{\text{WeightedL2}}$	0.785 ± 0.093	0.253 ± 0.007	2.029 ± 0.066	0.203 ± 0.036	0.422 ± 0.040	0.639 ± 0.022
	$\text{OSL}_{\mathcal{L}_1}$	0.457 ± 0.027	**0.216** ± 0.003	**1.812** ± 0.033	0.327 ± 0.009	0.560 ± 0.011	0.715 ± 0.007
	$\text{OSL}_{\mathcal{L}_2}$	1.113 ± 0.156	0.286 ± 0.020	2.375 ± 0.178	0.168 ± 0.015	0.345 ± 0.036	0.567 ± 0.043
	$\text{FOSL}_{\mathcal{L}_1}$	**0.449** ± 0.013	**0.216** ± 0.003	1.822 ± 0.022	**0.338** ± 0.013	0.562 ± 0.008	**0.718** ± 0.004

Nevertheless, the improvements in robustness can only be observed with relatively small margins. To highlight the effects of noisy data in a more exposing way, we show the results of an additional experiment injecting artificial noise into the original training data. To do so, we sample each observed training depth z from a normal distribution $\mathcal{N}(z, \hat{\sigma})$ with $\hat{\sigma}(x) := 0.01x^2 + \hat{\epsilon}$, where $\hat{\epsilon} \in \{0.5, 1.0\}$ is a parameter controlling the noise level. Note that we incorporate a heteroscedastic noise that increases with higher depth values.

The results in Table 2 demonstrate the incapability of conventional least squares optimization to provide reliable estimators under high noise. Accordingly, $\mathcal{L}_{\text{ScaledSIError}}$ does not lead to models learned in a robust manner either. As opposed to that, most of the conventional robust methods, especially the superset losses, turn out to be appropriate choices. Noteworthy, $\text{OSL}_{\mathcal{L}_1}$ provides the best performance for $\hat{\epsilon} = 0.5$, whereas $\text{FOSL}_{\mathcal{L}_1}$ proves its robustness capabilities for a higher noise of $\hat{\epsilon} = 1.0$.

4.3. Heterogeneous Depth Sensors: SunRGBD

While a single sensor was used to construct *NYUD-v2*, one may be interested in the case where multiple data sources are combined. In fact, this aggravates the problem of heterogeneous errors violating classical statistical assumptions as discussed before. In the following,

Table 3: Averaged results and standard deviations on models trained on 2k instances and the complete data set of *SunRGBD* on *DIODE*. The best model is indicated in bold per number of instances and metric.

# Insts.	Loss	*DIODE*						*SunRGBD*		
		REL (↓)	$\log_{10}$ (↓)	RMS (↓)	δ_1 (↑)	δ_2 (↑)	δ_3 (↑)	REL (↓)	RMS (↓)	δ_1 (↑)
2k	$\mathcal{L}_2$	0.512 ± 0.062	0.264 ± 0.043	1.922 ± 0.061	0.292 ± 0.033	0.515 ± 0.044	0.671 ± 0.039	0.432 ± 0.026	**1.135** ± 0.021	0.423 ± 0.010
	$\mathcal{L}_1$	0.432 ± 0.013	0.222 ± 0.006	1.837 ± 0.041	0.323 ± 0.006	0.554 ± 0.010	0.717 ± 0.011	0.423 ± 0.029	1.196 ± 0.052	0.415 ± 0.021
	$\mathcal{L}_{\text{Huber}}$	0.440 ± 0.016	0.218 ± 0.004	1.812 ± 0.028	0.328 ± 0.011	0.571 ± 0.011	0.726 ± 0.009	0.448 ± 0.028	1.175 ± 0.052	0.416 ± 0.019
	$\mathcal{L}_{\text{BerHu}}$	0.429 ± 0.017	0.220 ± 0.010	1.810 ± 0.051	0.316 ± 0.025	0.561 ± 0.022	0.726 ± 0.019	0.445 ± 0.014	1.222 ± 0.031	0.404 ± 0.009
	$\mathcal{L}_{\text{Ruber}}$	0.421 ± 0.011	0.218 ± 0.006	1.796 ± 0.035	0.321 ± 0.013	0.570 ± 0.008	0.729 ± 0.010	0.449 ± 0.011	1.218 ± 0.024	0.402 ± 0.008
	$\mathcal{L}_{\text{Barron}}$	0.463 ± 0.005	0.219 ± 0.005	1.823 ± 0.030	0.325 ± 0.012	0.560 ± 0.009	0.718 ± 0.011	0.480 ± 0.029	1.239 ± 0.059	0.403 ± 0.015
	$\mathcal{L}_{\text{trim}}$	0.500 ± 0.026	0.223 ± 0.003	1.863 ± 0.055	0.334 ± 0.014	0.565 ± 0.006	0.703 ± 0.009	**0.419** ± 0.044	1.150 ± 0.032	**0.429** ± 0.017
	$\mathcal{L}_{\text{ScaledSIError}}$	0.419 ± 0.004	0.228 ± 0.004	1.836 ± 0.019	0.309 ± 0.014	0.550 ± 0.011	0.714 ± 0.009	0.446 ± 0.009	1.224 ± 0.024	0.399 ± 0.006
	$\mathcal{L}_{\text{WeightedL2}}$	0.462 ± 0.018	0.223 ± 0.006	1.839 ± 0.038	0.328 ± 0.008	0.551 ± 0.008	0.708 ± 0.012	0.422 ± 0.026	1.159 ± 0.035	0.425 ± 0.013
	$\text{OSL}_{\mathcal{L}_1}$	0.424 ± 0.014	**0.206** ± 0.003	**1.730** ± 0.019	**0.346** ± 0.008	**0.591** ± 0.006	**0.734** ± 0.008	0.468 ± 0.011	1.192 ± 0.035	0.403 ± 0.008
	$\text{OSL}_{\mathcal{L}_2}$	0.471 ± 0.039	0.209 ± 0.004	1.756 ± 0.022	0.327 ± 0.010	0.577 ± 0.005	0.731 ± 0.013	0.464 ± 0.033	1.194 ± 0.059	0.408 ± 0.020
	$\text{FOSL}_{\mathcal{L}_1}$	**0.413** ± 0.009	0.219 ± 0.002	1.801 ± 0.008	0.318 ± 0.009	0.562 ± 0.078	0.728 ± 0.001	0.435 ± 0.013	1.220 ± 0.024	0.402 ± 0.009
Full	$\mathcal{L}_2$	0.418 ± 0.014	0.207 ± 0.003	1.733 ± 0.023	0.342 ± 0.003	0.585 ± 0.009	0.750 ± 0.011	0.470 ± 0.002	1.238 ± 0.007	0.398 ± 0.003
	$\mathcal{L}_1$	0.408 ± 0.008	0.219 ± 0.004	1.787 ± 0.026	0.304 ± 0.010	0.574 ± 0.008	0.746 ± 0.006	0.462 ± 0.014	1.250 ± 0.015	0.394 ± 0.006
	$\mathcal{L}_{\text{Huber}}$	0.394 ± 0.010	0.208 ± 0.009	1.715 ± 0.050	0.345 ± 0.030	0.604 ± 0.023	0.766 ± 0.014	0.483 ± 0.007	1.260 ± 0.021	0.388 ± 0.008
	$\mathcal{L}_{\text{BerHu}}$	0.391 ± 0.008	0.210 ± 0.009	1.720 ± 0.048	0.332 ± 0.026	0.603 ± 0.016	0.767 ± 0.014	0.485 ± 0.015	1.283 ± 0.014	0.386 ± 0.002
	$\mathcal{L}_{\text{Ruber}}$	0.376 ± 0.003	0.203 ± 0.002	1.679 ± 0.011	0.338 ± 0.011	0.617 ± 0.009	0.788 ± 0.001	0.489 ± 0.012	1.278 ± 0.017	0.388 ± 0.002
	$\mathcal{L}_{\text{Barron}}$	0.415 ± 0.009	0.208 ± 0.002	1.727 ± 0.017	0.335 ± 0.005	0.590 ± 0.002	0.762 ± 0.004	0.480 ± 0.010	1.263 ± 0.016	0.393 ± 0.003
	$\mathcal{L}_{\text{trim}}$	0.426 ± 0.011	0.218 ± 0.009	1.803 ± 0.049	0.325 ± 0.019	0.565 ± 0.015	0.734 ± 0.013	0.483 ± 0.023	1.316 ± 0.072	0.391 ± 0.010
	$\mathcal{L}_{\text{ScaledSIError}}$	0.377 ± 0.006	0.204 ± 0.007	1.690 ± 0.037	0.345 ± 0.014	0.609 ± 0.011	0.768 ± 0.008	0.471 ± 0.012	1.269 ± 0.017	0.387 ± 0.005
	$\mathcal{L}_{\text{WeightedL2}}$	0.418 ± 0.012	0.208 ± 0.007	1.748 ± 0.045	0.345 ± 0.013	0.581 ± 0.014	0.745 ± 0.015	**0.463** ± 0.005	1.226 ± 0.011	0.401 ± 0.004
	$\text{OSL}_{\mathcal{L}_1}$	**0.372** ± 0.019	**0.187** ± 0.007	**1.598** ± 0.050	**0.364** ± 0.003	**0.632** ± 0.011	**0.796** ± 0.012	0.475 ± 0.011	1.229 ± 0.014	0.401 ± 0.006
	$\text{OSL}_{\mathcal{L}_2}$	0.425 ± 0.054	0.195 ± 0.006	1.659 ± 0.037	0.354 ± 0.019	0.607 ± 0.015	0.765 ± 0.020	0.480 ± 0.016	**1.199** ± 0.016	**0.403** ± 0.006
	$\text{FOSL}_{\mathcal{L}_1}$	0.381 ± 0.004	0.210 ± 0.003	1.706 ± 0.018	0.324 ± 0.024	0.599 ± 0.014	0.769 ± 0.008	0.483 ± 0.016	1.242 ± 0.018	0.392 ± 0.003

we study the performance of models trained on *SunRGBD* for a varying number of training instances. In the appendix, we present further results on the test split of *SunRGBD*.

With less conformant error terms, the optimization with weaker model assumptions turns out to be reasonable. Table 3 shows the results for models trained on a subset of 2k instances and the full data set. As expected, the optimization based on $\mathcal{L}_2$ turns out to be misleading, especially for a small number of instances. Here, all robust variants turn out to work significantly better, most notably the OSL-based methods. $\text{OSL}_{\mathcal{L}_1}$ delivers the best performance with regard to the presented metrics. All superset losses improve over their respective baselines $\mathcal{L}_1$ and $\mathcal{L}_2$.

For a larger number of training examples, this trend continues, with $\text{OSL}_{\mathcal{L}_1}$ also providing the best performance for all reported metrics. Although less drastically, $\text{OSL}_{\mathcal{L}_2}$ and $\text{FOSL}_{\mathcal{L}_1}$ still outperform $\mathcal{L}_2$ and $\mathcal{L}_1$ respectively, further confirming the adequacy of an imprecisiation-based modeling. All in all, these results are in accordance with the initial motivation: By weakening the assumptions about the given data, we can leverage more robust loss alternatives for more accurate depth estimators.

5. Conclusion

We motivated the use of robust regression in depth estimation and revisited related loss functions, either applied in classical regression or specifically tailored to the domain of monocular depth estimation. Moreover, as an alternative to established approaches, we proposed the idea of "data imprecisiation" combined with superset learning. Instead of assuming precise but unreliable depth sensor signals as ground truth, the idea is to replace these targets by (fuzzy) intervals, leading to an imprecise but more reliable representation of the ground truth.

In an extensive empirical evaluation, we could demonstrate the effectiveness of robust losses compared to conventional approaches such as OLS. Especially in the regime of little data with high noise, the superset learning approach turns out to achieve state-of-the-art performance.

Motivated by these results, we plan to further elaborate on the modeling of data to further improve robustness. In particular, going beyond a global (homogeneous) imprecisiation, we plan to investigate modeling on a per-instance basis, e.g., by distinguishing the reliability of instances based on the depth value itself or the position in the image.

Acknowledgments

This work was supported by the German Research Foundation (DFG) under Grant No. 420493178. Moreover, computational resources were provided by the Paderborn Center for Parallel Computing (PC^2).

References

Min Sung Ahn, Hosik Chae, Donghun Noh, Hyunwoo Nam, and Dennis W. Hong. Analysis and noise modeling of the Intel RealSense D435 for mobile robots. In *Proc. of the 16th International Conference on Ubiquitous Robots, UR*, pages 707–711. IEEE, 2019.

Jonathan T. Barron. A general and adaptive robust loss function. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 4331–4339. Computer Vision Foundation / IEEE, 2019.

Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. AdaBins: Depth estimation using adaptive bins. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*, pages 4009–4018, June 2021.

Vivien Cabannes, Alessandro Rudi, and Francis R. Bach. Structured prediction with partial labelling through the infimum loss. In *Proc. of the 37th International Conference on Machine Learning, ICML*, volume 119, pages 1230–1239. PMLR, 2020.

Marcela Carvalho, Bertrand Le Saux, Pauline Trouvé-Peloux, Andrés Almansa, and Frédéric Champagnat. On regression losses for deep depth estimation. In *Proc. of the IEEE International Conference on Image Processing, ICIP*, pages 2915–2919. IEEE, 2018.

Qifeng Chen and Vladlen Koltun. Fast MRF optimization with application to depth reconstruction. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 3914–3921. IEEE Computer Society, 2014.

Yadolah Dodge. An introduction to L1-norm based statistical data analysis. *Computational Statistics & Data Analysis*, 5(4):239–253, 1987. ISSN 0167-9473.

Christopher Dougherty. *Introduction to econometrics.* Oxford University Press, 2011.

David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems, NIPS*, pages 2366–2374, 2014.

Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 2002–2011. IEEE Computer Society, 2018.

Peter J. Huber. *Robust Statistics.* Wiley Series in Probability and Statistics. Wiley, 1981.

E. Hüllermeier. Learning from imprecise and fuzzy observations: Data disambiguation through generalized loss minimization. *Int. J. Approx. Reason.*, 55(7):1519–1534, 2014.

Go Irie, Takahito Kawanishi, and Kunio Kashino. Robust learning for deep monocular depth estimation. In *Proc. of the IEEE International Conference on Image Processing, ICIP*, pages 964–968. IEEE, 2019.

Takeaki Kariya and Hiroshi Kurata. *Generalized least squares.* John Wiley & Sons, 2004.

Alex Kendall and Yarin Gal. What uncertainties do we need in Bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NeurIPS*, pages 5574–5584, 2017.

Kourosh Khoshelham and Sander Oude Elberink. Accuracy and resolution of Kinect depth data for indoor mapping applications. *Sensors*, 12(2):1437–1454, 2012.

George J. Klir and Tina A. Folger. *Fuzzy sets, uncertainty and information.* Prentice Hall, 1988.

Tobias Koch, Lukas Liebel, Friedrich Fraundorfer, and Marco Körner. Evaluation of CNN-based single-image depth estimation methods. In *Proc. of the European Conference on Computer Vision, ECCV, Workshops Part III*, volume 11131 of *LNCS*, pages 331–348. Springer, 2018.

Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *Proc. of the 4th International Conference on 3D Vision, 3DV*, pages 239–248. IEEE Computer Society, 2016.

Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *CoRR*, abs/1907.10326, 2019.

Zhengqi Li and Noah Snavely. MegaDepth: Learning single-view depth prediction from internet photos. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 2041–2050. IEEE Computer Society, 2018.

Julian Lienen and Eyke Hüllermeier. Instance weighting through data imprecisiation. *Int. J. Approx. Reason.*, 134:1–14, July 2021. ISSN 0888-613X.

Fangchang Ma and Sertac Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *Proc. of the IEEE International Conference on Robotics and Automation, ICRA*, pages 1–8. IEEE, 2018.

Chuong V. Nguyen, Shahram Izadi, and David R. Lovell. Modeling Kinect sensor noise for improved 3D reconstruction and tracking. In *Proc. of the 2nd International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission*, pages 524–530. IEEE Computer Society, 2012.

René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

Philipp Rosenberger, Martin Holder, Marina Zirulnik, and Hermann Winner. Analysis of real world sensor behavior for rising fidelity of physically based Lidar sensor models. In *Proc. of the IEEE Intelligent Vehicles Symposium, IV, Changshu*, pages 611–616. IEEE, 2018.

B. Schölkopf and AJ. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* MIT Press, 2001.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *Proc. of the 12th European Conference on Computer Vision, ECCV, Part V*, volume 7576 of *LNCS*, pages 746–760. Springer, 2012.

Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. SUN RGB-D: A RGB-D scene understanding benchmark suite. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 567–576. IEEE Computer Society, 2015.

Deqing Sun, Stefan Roth, and Michael J. Black. Secrets of optical flow estimation and their principles. In *Proc. of the 23rd IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 2432–2439. IEEE Computer Society, 2010.

Igor Vasiljevic, Nicholas I. Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z. Dai, Andrea F. Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R. Walter, and Gregory Shakhnarovich. DIODE: A dense indoor and outdoor depth dataset. *CoRR*, abs/1908.00463, 2019.

Oliver Wasenmüller and Didier Stricker. Comparison of Kinect V1 and V2 depth images in terms of accuracy and precision. In *Proc. of the Asian Conference on Computer Vision, ACCV, Workshops, Part II*, volume 10117 of *LNCS*, pages 34–45. Springer, 2016.

Yicheng Wu, Vivek Boominathan, Huaijin G. Chen, Aswin C. Sankaranarayanan, and Ashok Veeraraghavan. PhaseCam3D - Learning phase masks for passive single view depth estimation. In *Proc. of the IEEE International Conference on Computational Photography, ICCP*, pages 1–12. IEEE, 2019.

10 Monocular Depth Estimation via Listwise Ranking using the Plackett-Luce Model

Author Contribution Statement

The idea of modeling relative depth estimation using a Plackett-Luce model was developed by the author and Eyke Hüllermeier, and was further refined by all authors in joint discussions. Eyke Hüllermeier and Ralph Ewerth provided guidance throughout this process. The paper was initially written by the author, and subsequently revised by all authors. Furthermore, the implementation and experimentation was done by the author.

Supplementary Material

An appendix to the paper is provided in Appendix F. The code of the official implementation is provided at `https://github.com/julilien/PLDepth` (Apache 2.0 license).

Monocular Depth Estimation via Listwise Ranking using the Plackett-Luce Model

Julian Lienen[1,*] Eyke Hüllermeier[2] Ralph Ewerth[3,4] Nils Nommensen[3,4]
[1] Paderborn University [2] University of Munich (LMU)
[3] L3S Research Center, Leibniz University Hannover [4] TIB Hannover

Abstract

In many real-world applications, the relative depth of objects in an image is crucial for scene understanding. Recent approaches mainly tackle the problem of depth prediction in monocular images by treating the problem as a regression task. Yet, being interested in an order relation *in the first place, ranking methods suggest themselves as a natural alternative to regression, and indeed, ranking approaches leveraging pairwise comparisons as training information ("object A is closer to the camera than B") have shown promising performance on this problem. In this paper, we elaborate on the use of so-called* listwise *ranking as a generalization of the pairwise approach. Our method is based on the Plackett-Luce (PL) model, a probability distribution on rankings, which we combine with a state-of-the-art neural network architecture and a simple sampling strategy to reduce training complexity. Moreover, taking advantage of the representation of PL as a random utility model, the proposed predictor offers a natural way to recover (shift-invariant) metric depth information from ranking-only data provided at training time. An empirical evaluation on several benchmark datasets in a "zero-shot" setting demonstrates the effectiveness of our approach compared to existing ranking and regression methods.*

1. Introduction

Estimating depth in monocular images constitutes a problem of practical importance when aiming to understand the geometry of a scene, e.g., in autonomous driving systems or for augmented reality applications. Due to its ill-posed nature, methods approaching this problem nowadays typically incorporate complex models, trained on large amounts of data using machine learning methods.

The majority of existing approaches tackles depth estimation (whether per-pixel or per-object) as a regression problem, i.e., as the problem of learning a model to predict a (pseudo-)metric map (e.g., [1, 9, 17, 18]). However, on the one hand, accurate prediction of metric depth actually depends on the intrinsic camera parameters, which are often not available. On the other hand, instead of predicting absolute depth, it is often enough to predict the *relative* depth of pixels or higher level concepts (such as objects), that is, to sort them from closest to farthest away from the camera.

One may then argue that regression is solving an unnecessarily difficult task, and rather advocate a formalization of depth estimation as a *ranking* task [12]. So-called "learning-to-rank" methods can be used to minimize suitable performance metrics based on relative errors. As absolute depth measurements are not necessarily needed, ranking has the additional advantage that it potentially allows for learning from weaker training information. This includes depth annotations that are not metric but can be regarded as pseudo-metric data, e.g., disparity maps constructed from stereo images or videos [6, 14, 20], or human-annotated data [5, 7]. Without the need for metric RGB-D data produced by depth sensors, the diversity of training datasets can be drastically increased due to cheaper data acquisition [33].

Existing ranking methods are essentially based on pairwise comparisons of the form "object A is closer to the camera than B" [5, 33, 34, 35]. Pairwise relations of that kind are sampled from a depth map as training information, and predictive models are induced by minimizing pairwise ranking losses. While these approaches have proven effective, the quadratic number of possible pairs that can be constructed renders them rather inefficient and necessitates sophisticated sampling strategies to eliminate less informative pairs [34]. Besides, breaking a linear order into pairwise comparisons necessarily comes with a certain loss of information. In par-

[*] Corresponding author: julian.lienen@upb.de.

ticular, information about the transitivity of order relations, which is implicitly contained in a linear order, will be lost.

To avoid these drawbacks, so-called “listwise ranking” [32] has been proposed as an alternative to pairwise methods. In the listwise approach, higher order rankings of arbitrary length can be considered as training information. In this paper, we elaborate on the use of listwise ranking for depth estimation in images. More specifically, we propose a listwise ranking method based on the well-known Plackett-Luce (PL) model [22, 24], which allows for learning probability distributions over rankings from pseudo-metric data. Moreover, taking advantage of the representation of PL as a random utility model [27], we suggest a natural way to recover translation-invariant approximations of the underlying metric depth information. Along with that, we propose a state-of-the-art neural network architecture as a backbone, together with a simple sampling strategy to construct training examples from raw pseudo-depth data.

In a zero-shot evaluation, where we compare models on data not considered for training, we study the cross-dataset performance of our model and compare it with state-of-the-art approaches. Thereby, we demonstrate that listwise ranking is an effective approach for rank-based error minimization, and our model constitutes an appropriate choice for the prediction of depth orders in unseen scenes, as well as providing promising results in recovering metric depth.

2. Related Work

In learning to rank, the goal is to infer ranking models from training data in the form of rankings (permutations) of individual items. According to a rough categorization of methods, one can distinguish between pointwise, pairwise, and listwise approaches [21]. While single items are considered as training examples in pointwise learning-to-rank methods, relations between items are typically used as training examples in the other categories, either relations of order two (pairwise) or arbitrary length (listwise). In the case of pointwise learning-to-rank, examples are usually annotated by a score that determines their individual usefulness, from which, for instance, regression models can be induced. For pairwise approaches, where examples are typically given as single relations among two items, existing methods range from SVM-based classifiers [15] to boosting methods [11] and ranking networks [2]. Similarly, several listwise ranking methods have been proposed, in which examples are represented by higher order (potentially partial) item rankings. One of the most well-known representative is ListMLE [32], a maximum likelihood estimation method to infer Plackett-Luce probability distributions over rankings.

Several approaches to tackle the problem of estimating depth in images using relative depth information for training have been proposed. Among the first, Zoran et al. [35] classify individual point pairs from an image, which are then combined into a global solution for a complete dense map over all image pixels. Following a similar motivation, Chen et al. [5] train a deep neural network architecture by using a pairwise ranking loss, directly predicting a dense-map in an end-to-end fashion. This approach has also been adopted in subsequent works and improved in various directions, for example by using a different model architecture [33], additional data [6], or improved sampling strategy [34]. Furthermore, Ewerth et al. [10] propose a method to estimate relative depth using a RankBoost model. Alternative approaches also exploit ordinal depth information [20], either directly or to pretrain regression models [4].

To learn models that work well for arbitrary scenes, e.g., in both indoor and outdoor scenarios, diversity of training data is crucial. Commonly used metric data produced by depth sensors typically provide limited diversity, e.g., NYUD-v2 [26] with indoor-only or KITTI [13] with only street scenes. Since maximal depth capacities of sensors constrain the recognizable depth, they fail to capture scenes “in the wild”. This is why Chen et al. [5] propose a human-annotated dataset with pairwise point samples, for which the “closer-to-camera” relation is captured. However, as it provides ground truth information for only two points in each image, and the human annotation process is quite costly, other strategies aiming to automatically extract depth information have been proposed. For instance, stereo images [33] or sequences of images in videos [6] have been facilitated to predict structural disparity maps from the motion of elements. Combinations of such methods have been considered, too [31]. As none of them delivers metric information per pixel, the information produced must be considered as pseudo-depth, which, as previously explained, is still sufficient for depth relations. Although scale-invariant regression methods are also capable of learning from such data [20, 25], their ability to generalize to new datasets with structurally different scenes is fairly limited, at least for the task of depth ordering, as our empirical evaluation will confirm later on.

3. Plackett-Luce Model for Depth Estimation

In the following, we introduce our proposal of a Plackett-Luce model for depth estimation as illustrated in Fig. 1, along with a description of the model architecture and sampling strategy to construct training examples from raw depth data.

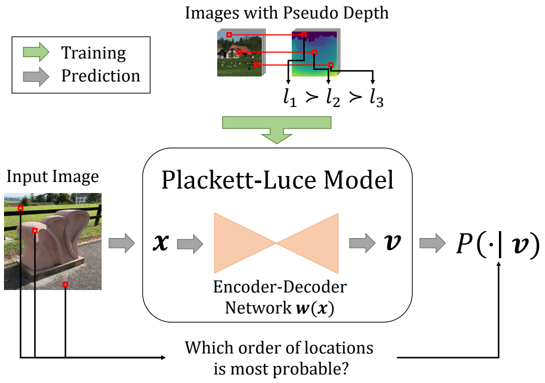

Images with Pseudo Depth
Training
Prediction
$l_1 \succ l_2 \succ l_3$
Input Image
Plackett-Luce Model
$\boldsymbol{x}$
$\boldsymbol{v}$
$P(\cdot \mid \boldsymbol{v})$
Encoder-Decoder Network $\boldsymbol{w}(\boldsymbol{x})$
Which order of locations is most probable?

Figure 1. Overview of our method: The PL model incorporates a deep neural network to predict scores for each pixel in an input image, which are then turned into probabilities for rankings of queried image locations. For training, we sample rankings from images annotated by pseudo depth.

3.1. Problem Formulation

We assume training information in the form of RGB images I together with (pseudo-)depth annotations D, i.e., tuples $(I, D) \in \mathbb{R}^{h \times w \times 3} \times \mathbb{R}^{h \times w}$, where h and w denote the image height and width, respectively. Moreover, $D[l]$ denotes the (pseudo-)depth of a position $l \in \{1, \ldots, h\} \times \{1, \ldots, w\}$ identified by a height and width coordinate. Without loss of generality, lower values $D[l]$ encode shorter distances to the camera.

We are mainly interested in the order relation of the locations in an image I as induced by the (pseudo-)depth D. Formally, the relation between n locations $M = \{l_1, l_2, \ldots, l_n\}$ can be represented in terms of a permutation π of $[n] := \{1, \ldots, n\}$ such that $D[l_{\pi(i)}] < D[l_{\pi(i+1)}]$ for $i \in \{1, \ldots, n-1\}$. This permutation encodes the ranking $l_{\pi(1)} \succ l_{\pi(2)} \succ \cdots \succ l_{\pi(n)}$, i.e., location $l_{\pi(1)}$ is closest, then $l_{\pi(2)}$, etc. At query time, when I is given but D is not, the task of a rank-based depth estimation model is to predict the "closer-to-camera" relation $\succ$, that is, to produce an accurate order-preserving estimate of D. Formally, this estimate can again be represented in terms of a permutation, which is then compared to the ground truth permutation π.

3.2. Listwise Depth Ranking

We model information about rankings in a *probabilistic* way, which has several advantages, especially from a learning point of view (for example, it makes the problem amenable to general inference principles such as maximum likelihood estimation). A well-known probability model on rankings is the Plackett-Luce (PL) model, which is parameterized by a vector $\boldsymbol{v} = (v_1, \ldots, v_K) \in \mathbb{R}_+^K$, where K is the number of items (length of the ranking). Referring to the interpretation of a ranking in terms of a preferential order, the value v_i is also called the (latent) utility of the i^{th} item — subsequently, we shall use the more neutral notion of PL score or parameter. The probability of a permutation π of $[K]$ is then given by

$$P(\pi \mid \boldsymbol{v}) = \prod_{i=1}^{K-1} \frac{v_{\pi(i)}}{\sum_{k=i}^{K} v_{\pi(k)}} \, , \tag{1}$$

where $\pi(i)$ is the index of the item on the i^{th} rank. One easily verifies that, the larger the score v_i, the higher the probability that the i^{th} item will show up on a top rank. Moreover, the mode of the distribution, i.e., the ranking with the highest probability, is obtained by sorting the items in decreasing order of their scores.

The PL model has the appealing property that each marginal of a PL model is again a PL model (with the same parameters). More specifically, if $J = \{j_1, \ldots, j_k\} \subseteq [K]$ is a subset of the K items, then the corresponding marginal of (1) is a PL model with parameters $v_{j_1}, \ldots, v_{j_k}$. This property greatly facilitates learning and inference from possibly incomplete rankings that do not comprise all K items. In fact, learning to rank with the PL model essentially comes down to estimating the score vector $\boldsymbol{v} = (v_1, \ldots, v_K)$.

In the case of depth estimation, items correspond to the pixels of an image, and the task of the learner is to predict the scores of these pixels. To make this possible, we assume that the score of a pixel can be expressed as a function of its context on the image. Thus, a parameter v_i is defined through a function $\phi_i : \mathcal{X} \longrightarrow \mathbb{R}$ on an input space $\mathcal{X}$ [8], where $\mathcal{X} = \mathbb{R}^{h \times w \times 3}$ corresponds to the space of all possible images of size $h \times w$. Assuming all images to have the same size, we set the overall number of alternatives K to $h \times w$.

In the domain of depth estimation, the most obvious way to represent the functions $\phi_1, \ldots, \phi_K$ is to model them as a (joint) deep convolutional neural network. Thus, each function ϕ_i is represented in terms of a set of network parameters $\boldsymbol{w}_i$, a subset of the parameters $\boldsymbol{w}$ of the entire (joint) network. In the experimental section, different state-of-the-art model architectures will be assessed for that purpose.

For an image $\boldsymbol{x} \in \mathcal{X}$, let $\boldsymbol{w}(\boldsymbol{x})$ denote the output of the neural network under parameterization $\boldsymbol{w}$ and

$$(v_1, \ldots, v_K) = (\phi_1(\boldsymbol{x}), \ldots, \phi_K(\boldsymbol{x})) = \exp(\boldsymbol{w}(\boldsymbol{x})) \tag{2}$$

the induced (non-negative) PL parameters. Thus, the entire PL model for the image $\boldsymbol{x}$ is eventually specified by the network parameters $\boldsymbol{w}$. Given a ranking π of (a subset of) the pixels of $\boldsymbol{x}$ as training information, one can thus determine the probability $P(\pi \mid \boldsymbol{x}, \boldsymbol{w})$ of that ranking under $\boldsymbol{w}$ according to (1). More generally, given training information in the form of a collection of images with rankings, $\{(\boldsymbol{x}_i, \pi_i)\}_{i=1}^L$, learning an optimal model can be realized as maximum likelihood estimation [32]:

$$\boldsymbol{w}^* \in \arg\min_{\boldsymbol{w}} - \sum_{i=1}^{L} \log P(\pi \mid \boldsymbol{x}, \boldsymbol{w}) \,. \qquad (3)$$

3.3. Metric Depth Estimation

Going beyond the prediction of rankings, one may wonder whether there is any possibility to recover metric depth information from a learned PL model. At first sight, this would be surprising, because the model is only trained on qualitative information in the form of rankings, and predicts probabilities instead of metric depth. Yet, the PL model also comprises a quantitative part, namely the scores v_i, which, as will be explained in the following, are in direct correspondence with the underlying metric information.

The PL model is a specific random utility model (RUM) [23]. In this class of models, it is assumed that the true order $z_1 < z_2 < \ldots < z_n$ of n real numbers — think of them as the true depth values of the pixels in a image — is "randomized" through (independent) measurement noise: Each value z_i is replaced by the measurement $X_i = z_i + \epsilon_i$, where ϵ_i is an error term, and what is observed as a ranking is the order of the measurements $X_1, \ldots, X_n$. In particular, the true order relation $z_i < z_j$ between two items is reversed if the corresponding error terms satisfy $\epsilon_i - \epsilon_j > z_j - z_i$, and the smaller the distance $|z_i - z_j|$, the more likely such a mistake is going to happen. Thus, the probability of a ranking error is indicative of the distance between z_i and z_j.

The PL model is obtained for the special case where the error terms ϵ_i follow a Gumbel distribution with fixed shape parameter [27]. More specifically, the so-called Thurstone model with parameters $z_1, \ldots, z_n$ is equivalent to the PL model (1) with parameters $v_i = \exp(z_i)$, $i = 1, \ldots, n$. In the context of depth estimation, the model can thus be interpreted as follows: The true depth of the i^{th} image object (pixel) is given by z_i, but due to measurement noise, these distances are not observed precisely. Accepting the assumption of a Gumbel distribution[1], a PL model fitted to the observed (noisy) rankings of image objects yields estimates $\hat{v}_i$ of $v_i = \exp(z_i)$. Thus, a natural estimate of the underlying metric depth is given by $\hat{z}_i = \log(\hat{v}_i)$.

We note that, since the PL model (1) is invariant toward multiplicative scaling (i.e., $P(\pi \mid \boldsymbol{v}) \equiv P(\pi \mid \lambda \boldsymbol{v})$ for $\lambda > 0$), the parameter $\boldsymbol{v}$ can only be determined up to a multiplicative factor. Correspondingly, the parameter $\boldsymbol{z}$ can only be determined up to an additive constant. This is indeed plausible: Assuming that the probability of reversing the order of two image objects only depends on their true distance $|z_i - z_j|$, this probability will not change by shifting the entire scene (i.e., moving the camera closer or farther away). In addition to this shift invariance, there is also a scaling effect, albeit of a more indirect nature. This effect is caused by fixing the shape parameter of the Thurstone model to 1. Therefore, instead of a simple log-transformation, we shall use an affine transformation of the form $\hat{z} = s \log(\hat{v}) + t$, with $s, t \in \mathbb{R}$ fitted to the image at hand.

3.4. Model

Regarding the underlying neural network, taking an image $\boldsymbol{x}$ as input and producing $\boldsymbol{w}(\boldsymbol{x})$ as used in (2) as output, we suggest two variants of our listwise ranking approach. The first one, dubbed *PLDepthResNet*, uses the same model architecture as suggested by Xian et al. [33]. As a second model, by consideration of recent neural architecture research, we propose *PLDepthEffNet* as a closely related architecture relying on EfficientNet [29] as backbone. Without further notice, the variant EfficientNetB5 is used as encoder, while the decoder part is a stack of repeating convolutional, BatchNormalization, ReLU and bilinear upsampling layers until the original shape is recovered. Similar to the model in [33], different scale features from the encoder branch are fed into the corresponding levels of the decoder part. Instead of fusing these features by addition, we concatenate at the respective layers. As a result, we obtain a model with approximately 45 million parameters for PLDepthEffNet, which is similar to the size of PLDepthResNet with 42 million parameters, while increasing the model performance at the same time (cf. empirical evaluation).

For both PLDepthResNet and PLDepthEffNet, we use encoders pretrained on ImageNet. Consequently, we standardize input images to match the preprocessing on ImageNet. During training, we freeze the encoder part and only allow the BatchNormalization layers to adjust to the new input data as typically done in transfer learning.

3.5. Sampling

In the past, different strategies to construct pairwise relations from raw depth data have been proposed, including superpixel sampling [35], random sampling [5], and combinations of multiple structure-guided strategies [34]. Ac-

[1] This distribution looks similar to the normal distribution. Even if not provably correct, it is certainly not implausible.

cording to Xian et al. [34], random sampling of pairwise relations from raw depth data may harm the model's performance, due to training on uninformative or even misleading examples. Even worse, due to imprecision in the ground truth data, the risk of incorrectly ordered items increases with larger samples.

To address these issues, we propose a random sampling strategy that is almost as simple as pure random sampling, and which allows for incorporating the depth structure of the given image while leading to a relatively low training complexity. For R n-ary rankings to be queried per training tuple (I, D), $N \cdot R$ item sets M with n individual image locations are sampled, where $N > 1$ is a parameter. For each ranking set M, we order all image locations l by $D[l]$ to construct a ground truth permutation π. Given π, we sum up all pairwise depth differences $|D[l_{\pi(i)}] - D[l_{\pi(i+1)}]|$, $i \in [n-1]$. Afterwards, we sort all $N \cdot R$ rankings per image in a decreasing order according to this sum of depth difference and select the top R rankings as training examples. This way, we consider those rankings that seem to be most informative, since their relative depth values are maximized among the samples. Other strategies, such as the minimum among all pairwise depth differences in a ranking, are of course also possible as a proxy of the amount of information.

It is worth to mention that the Plackett-Luce model does not support partial rankings, i.e., neither allows for ties nor incomparability between items. Thus, as opposed to strategies incorporating equality relations, as e.g. [5], such relations are not explicitly considered here. To avoid sampling point pairs that are almost equally far away from the camera, we add a penalty of -10 to the depth difference sum for each compared image location pair l_1 and l_2 if their depth difference is such that $\max\left\{\frac{D[l_1]}{D[l_2]}, \frac{D[l_2]}{D[l_1]}\right\} < 1 + \tau$, where the parameter τ is set to $\tau = 0.03$ in our experiments.

4. Experiments

To demonstrate the effectiveness of our method, we conduct an exhaustive empirical evaluation on several benchmark datasets. Before presenting the results, we first introduce the datasets, followed by a brief description of the baseline methods and metrics used for assessment.

4.1. Datasets

To train our models, we use the recently introduced pseudo-metric "High-Resolution Web Stereo Image" (*HR-WSI*) dataset [34]. It consists of $20,378$ diverse, high resolution images annotated with pseudo-depth maps generated from flow predictions. For hyperparameter optimization, a separate set of 400 images was used. Since the flow predictions provided as depth annotation failed for some image regions, a consistency mask is attached to each prediction to allow for sampling only from pixels that provide a reasonable depth value. To this end, a forward-backward flow consistency check has been applied. Furthermore, the annotations have been preprocessed to also assign a constant depth value to sky regions. Despite its relatively small size, we found this dataset to provide highly informative image and depth pairs to learn from.

In the experiments, we compare our model to various baselines in a "zero-shot" generalization study on datasets that were not used within the training processes. Thus, we follow the basic evaluation scheme by Ranftl et al. [25]. As datasets, we consider *Ibims* [16], *Sintel* [3], *DIODE* [30], and *TUM* [28]. In the supplementary material, we detail the characteristics of each dataset, such as their data diversity. With this choice of benchmark targets, we capture indoor, outdoor, and computer generated scenes, which provides a good basis for assessing the generalization performance of different models, and their ability to predict depth orders in a wide variety of applications.

4.2. Baselines

We compare our PL-based approach to state-of-the-art depth estimation models using depth relations as training information. To this end, we consider the ResNet-based model trained on "Relative Depth from Web" (ReDWeb), "Depth in the Wild" (DIW), and YouTube3D as described by Chen et al. [6], hereinafter referred to as YouTube3D, and the same model as used by Xian et al. [34] trained on HR-WSI (referred to as Xian 2020). Both approaches have shown compelling generalization performance, corroborating our motivation to use relative data for supervision.

Besides models trained on relative depth information, regression models are obviously also capable of inferring rankings, simply by sorting the image locations based on their values in a predicted dense depth map. Therefore, we consider state-of-the-art (pseudo-)regression methods as additional baselines, namely, DenseDepth [1], BTS [18], MegaDepth [20], MannequinChallenge (MC) [19], and MiDaS [25]. Furthermore, we also evaluated MonoDepth2 [14] as a completely unsupervised resp. self-supervised method.

While we considered most baselines as described in the related work, let us note that the authors of MiDaS provide a model trained on approximately 2 million examples, which is far more than most of the other methods we compare with. To account for this, we re-implemented their approach and retrained the model on HR-WSI for a fairer comparison. For a complete overview of all baselines, including a categorization of the respective training data diversity, we refer to the supplementary material.

4.3. Metrics

To evaluate our models, we report the "ordinal error" on sampled point pairs as done by Xian et al. [34]. For two points l_1 and l_2 sampled from an example (I, D), with I being the image and D a dense (pseudo-)depth map as specified before, the ground truth ordinal relation $r(l_1, l_2, D)$ is given by $+1$ for $D[l_1] > D[l_2]$, -1 for $D[l_2] > D[l_1]$ and 0 otherwise. The ordinal error is then given by

$$ord(\mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{(I,D,l_1,l_2) \in \mathcal{D}} \mathbb{1}\big(r(l_1, l_2, D) \neq r(l_1, l_2, f(I))\big), \tag{4}$$

where f is the function predicting depth or, in the case of a PL model, scores for each pixel of the input image I, resulting in a dense depth map just as given by D, and $\mathcal{D}$ denotes the set of all point pairs sampled from the test dataset images and depth maps.

As already noted, we omit all equal pairs, i.e., relations with $r(\cdot, \cdot, \cdot) = 0$. Hence, we report ord on unequal pairs only without any equality thresholding. Thus, there is no need to rely on re-scaling and -translating as done in [25] and [34] to identify reasonable equality thresholds, which comes with additional complications for the evaluation process.

Often, depth orders have varying priorities, i.e., closer elements are more critical for correct ordering than elements far away from the camera. For example, an autonomous vehicle has less time to react to elements very close to the car and must rely on valid input for safe interactions. This is reflected by metrics like the discounted cumulative gain (DCG), which measures the usefulness of rankings by accumulating graded relevances of ranking items discounted with decreasing rank. More precisely, for every image location l associated with a dense depth map D, we set the relevance score of l in D to $rel(l, D) = \frac{1}{D[l]+1}$. Given these scores, we can specify the DCG score for a ranking $l_{\pi(1)} \succ l_{\pi(2)} \succ \cdots \succ l_{\pi(n)}$ by

$$DCG(\pi, D) = \sum_{i=1}^{n} \frac{rel(l_{\pi(i)}, D)}{\log_2(i+1)} \, . \tag{5}$$

For our experiments, we used the normalized DCG (nDCG), which divides (5) by the best DCG possible on D.

For the metric comparison, we assess the root-mean-square error (RMSE) between the dense ground truth and predicted depth maps and the percentage of predictions $\hat{z}$ such that $\max\left(\frac{\hat{z}}{z}, \frac{z}{\hat{z}}\right) = \delta > 1.25$ for the ground truth depth z. To calculate the metrics, we normalized the given ground truth scores by the maximum depth capacity of the corresponding dataset (cf. the dataset characteristics in the supplement) to obtain error values on a similar scale.

4.4. Results

To show the effectiveness of our method proposal,we first compare different losses using the same model architecture and training dataset, followed by a comparison of our method to the baselines. Every reported result is the average of three runs with different randomization seeds.

4.4.1 Loss Comparison

There are many experimental studies in the literature showing improved performance of a method, but not isolating the key factors contributing to the improvement, e.g., the neural network architecture, loss function, training procedure, training data, etc. To assess the influence of a listwise approach to ranking more clearly, we evaluate three methods trained on the same data and with the same neural network architecture, namely (scale-invariant, SI) regression, pairwise, and listwise ranking. It is true that the model, loss, and data may strongly interact with each other (i.e., a loss might work well with a certain architecture on a particular dataset, while the same architecture may harm the performance of a different method). Nevertheless, we found that the ResNet-based architecture as proposed by Xian et al. [33] and subsequently also used in [25] serves as a good basis for a fair comparison.

For our experiments, we re-implemented the SI mean-squared error loss as also used in MiDaS and the pairwise ranking loss as described in [5] and [33]. As training information, we used HR-WSI as a state-of-the-art diverse pseudo-depth dataset. We refer to the supplement for a detailed description of all hyperparameters.

All three methods require different sampling strategies: While the SI-regression uses the complete (masked) image, pair- and listwise methods involve different amounts of sampled points selected per ranking. For a fair comparison, we adopted the number of sampled rankings in the listwise case to the number of drawn pairwise relations, such that one approach does not see much more points than the other during training. In the case of pairwise rankings, we randomly sampled 1k point pairs per image and epoch, resulting in a maximum of 2k seen points per image and epoch. For our listwise approach, we found a size of 5 to achieve a good trade-off between highly informative rankings and effi-

Table 1. Ordinal errors on 50k randomly sampled pairs per loss, using the architecture from [34] trained on HR-WSI (lower is better).

Loss	Ibims	Sintel	DIODE	TUM	Avg. Rank
SI-Regression	0.308	0.311	0.334	0.222	3
Pairwise	0.281	0.299	0.291	**0.192**	1.75
Listwise	**0.273**	**0.289**	**0.285**	0.218	**1.25**

cient training. Hence, we sampled 400 rankings of ranking size 5 per image and epoch. Here, we explicitly stick to random-only sampling to alleviate side effects.

Table 1 presents the results of the method comparison on 50k randomly sampled location pairs per image. As can be seen, the relative models outperform the SI-regression method, suggesting to serve as a better surrogate loss for optimizing the ordinal error. Moreover, our listwise approach seems to perform slightly better than the pairwise approach, although the difference does not appear to be significant.

4.4.2 Ordinal Prediction

After having compared the loss function on a shared model and data level, we now analyze individual depth estimation models with regard to their ordinal error and nDCG performance as trained by the respective authors, who made an attempt at optimizing the interplay between data, network architectures, and training procedures.

For the baseline models, we used the best provided pretrained models by the authors or, if official implementations were not available, by popular and carefully tested re-implementations. For our PL models, we kept most of the training hyperparameters the same (see supplementary for more details). Within our sampling strategy, we set the factor $N = 5$ (cf. Section 3.5). For MiDaS, we also used our proposed EfficientNet-based architecture, which delivers superior performance compared to the formerly used architecture, for reasons of fairness. Here, as opposed to the version of MiDaS within the loss comparison, where we primary focused on comparing different problem considerations, we employ the trimmed absolute deviation loss providing the best performance among the regarded alternatives (cf. [25]).

Table 2. Ordinal errors on benchmark datasets with 50k randomly sampled relations for each image (lower is better).

Model	Ibims	Sintel	DIODE	TUM	Avg. Rank
DenseDepth	0.208	0.384	0.317	0.224	5.75
MegaDepth	0.297	0.324	0.316	0.227	7.5
BTS	**0.190**	0.384	0.323	0.251	6.25
MC	0.272	0.387	0.378	0.206	7.25
MiDaS	0.269	0.278	0.263	0.207	3.75
MonoDepth2	0.375	0.425	0.407	0.336	9.75
YouTube3D	0.272	0.292	0.288	0.199	4.75
Xian 2020	0.225	0.278	0.263	**0.184**	2.25
PLDepthResNet	0.245	0.284	0.277	0.213	4.75
PLDepthEffNet	0.213	**0.272**	**0.256**	0.204	**2**

Table 3. nDCG on benchmark datasets with 100 randomly sampled rankings of size 500 for each image (higher is better).

Model	Ibims	Sintel	DIODE	TUM	Avg. Rank
DenseDepth	0.916	0.986	0.821	0.986	4.75
MegaDepth	0.911	0.989	0.815	0.983	7.5
BTS	**0.918**	0.986	0.825	0.983	4.75
MC	0.908	0.986	0.828	0.987	5.5
MiDaS	0.913	0.991	0.806	0.987	6.25
MonoDepth2	0.896	0.981	**0.836**	0.961	7.75
YouTube3D	0.911	0.993	0.816	0.988	4.75
Xian 2020	0.916	0.993	0.817	**0.990**	2.75
PLDepthResNet	0.914	0.993	0.817	0.985	5
PLDepthEffNet	0.916	**0.994**	0.819	0.988	**2.5**

Table 2 reports the individual ordinal errors on unequal relations for the four benchmark datasets, again on 50k randomly sampled location pairs per image. As can be seen, our PLDepthEffNet achieves the lowest averaged rank over all datasets, while outperforming the other methods on half of the datasets at the same time, demonstrating the effectiveness of the listwise ranking approach to optimize the ordinal error metric. Supporting the observations made in the previous experiment, the generalization capabilities of MegaDepth as another scale-invariant regression method, even by having access to over 600k diverse instances, to correctly rank elements are fairly limited. Moreover, in agreement with the previous results, the ranking approaches are consistently among the best models, suggesting ranking losses to be the favorite choice as surrogates for ordinal error minimization.

Additionally, Table 3 reports the results for nDCG as performance metric on 100 randomly sampled rankings of size 500 per image. In accordance with the ordinal errors, ranking methods are well suited to optimize this metric. Here, the top-3 models are all of that kind, with PLDepthEffNet slightly better performing than Xian 2020.

4.4.3 Metric Prediction

As motivated theoretically in Section 3.3, our method provides an interface to recover metric depth information approximated from observed rankings. Here, we compare our model to the baselines with regard to the two metric error measures RMSE and $\delta > 1.25$ using the same models as in Section 4.4.2. As all benchmark datasets have different scales and might be shifted arbitrarily, we rescale and shift the predictions to the resolution of the ground truth as de-

Table 4. Evaluation results on benchmark datasets with regard to metric depth error measures (lower is better in both cases).

Model	Ibims		Sintel		DIODE		TUM		Avg. Rank	
	RMSE	$\delta > 1.25$	RMSE	$\delta > 1.25$	RMSE	$\delta > 1.25$	RMSE	$\delta > 1.25$	RMSE	$\delta > 1.25$
DenseDepth	**0.016**	20.9	0.128	39.6	0.110	53.5	0.084	69.7	5.25	4.5
MegaDepth	0.020	35.9	0.119	35.5	0.094	55.3	0.082	70.8	6	7
BTS	**0.016**	**18.9**	0.133	41.8	0.112	54.4	0.089	72.4	7	6.25
MC	0.018	31.3	0.128	38.8	0.120	58.7	**0.074**	**67.8**	5.25	5.5
MiDaS	0.019	33.2	**0.091**	**27.7**	**0.081**	53.5	0.085	71.1	4	4.75
MonoDepth2	0.023	42.6	0.143	43.8	0.122	61.1	0.088	72.5	9.75	10
YouTube3D	0.019	31.8	0.101	31.1	0.096	54.5	0.077	68.4	4.75	5.25
Xian 2020	0.018	31.5	0.096	30.5	0.085	**51.4**	0.080	69.4	**3**	**3.25**
PLDepthResNet	0.019	30.9	0.099	30.7	0.092	53.1	0.084	71.9	5	4.75
PLDepthEffNet	0.017	29.1	0.093	29.3	0.085	52.7	0.083	71.6	**3**	3.5

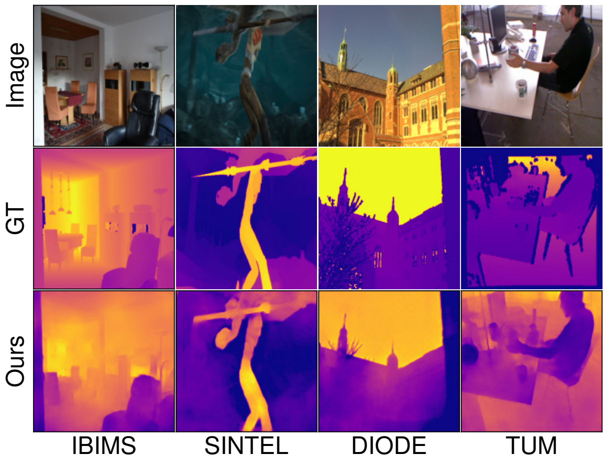


Figure 2. Sample predictions given by the reconstructed metric scores of the PLDepthEffNet model as used in the experiments.

scribed in [25] by optimizing a least-squares criterion.

The results are given in Table 4. As can be seen, although our model was solely trained on rankings, it is capable of recovering the underlying depth structure relatively precisely. Noteworthy, it is superior to all regression baselines and on a par with Xian 2020 for RMSE, although this ranking baseline additionally incorporates a smooth gradient loss term for sharp boundaries, directly accessing the metric depth information at training time. While it delivers the highest $\delta > 1.25$ accuracy, our approach still proves to be very competitive in this regard.

Fig. 2 shows exemplary predictions of our model. Obviously, the model is able to capture tiniest object details, such as tree branches in the image from DIODE, and predicting sharp object boundaries. This shows that, even with simple sampling strategies, listwise ranking is able to reflect and predict such small details, without any need for very complex strategies based on the depth structure of an image.

5. Conclusion

We have proposed to tackle the problem of depth ordering in images as a listwise ranking problem, for which we employed a Plackett-Luce model tailored to the domain of monocular depth estimation. Thus, compared to estimating the exact depth values, we solve an arguably simpler problem, at least if the goal is to minimize an ordinal error metric. Besides, compared to precise numerical data required by regression models for training, a ranking approach allows for leveraging weaker and more diverse training data. Although not directly trained on metric data, our model is capable of providing precise (shift-invariant) depth predictions, essentially by exploiting the relationship between the (latent) distance between image objects and the probability of reversing their order in a ranking.

Through an exhaustive zero-shot cross-dataset evaluation, we showed that our approach, combined with a state-of-the-art neural network as backbone, achieves superior ranking performance compared to previous approaches. In particular, it improves upon existing pairwise ranking methods, in spite of using a much simpler and more efficient sampling technique. Remarkably, our model also performs very competitive on metric error measures.

Motivated by these promising results, we plan to elaborate on further improvements of the listwise ranking approach. This includes an investigation of the effect of varying the ranking size, as well as an extension toward learning from partial rankings and incorporating equality relations. In addition, as we only applied random sampling so far, we plan to develop more sophisticated sampling strategies leading to more informative rankings to learn from.

Acknowledgement. This work was supported by the German Research Foundation (DFG) under Grant 3050231323. Moreover, computational resources were provided by the Paderborn Center for Parallel Computing (PC^2).

References

[1] Ibraheem Alhashim and Peter Wonka. High quality monocular depth estimation via transfer learning. *CoRR*, abs/1812.11941, 2018.

[2] Christopher J. C. Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Gregory N. Hullender. Learning to rank using gradient descent. In Luc De Raedt and Stefan Wrobel, editors, *Proceedings of the 22nd International Conference on Machine Learning, (ICML), August 7-11, 2005, Bonn, Germany*, volume 119 of *ACM International Conference Proceeding Series*, pages 89–96. ACM, 2005.

[3] Daniel J. Butler, Jonas Wulff, Garrett B. Stanley, and Michael J. Black. A naturalistic open source movie for optical flow evaluation. In Andrew W. Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Proceedings of the 12th European Conference on Computer Vision (ECCV), Part VI, October 7-13, 2012, Florence, Italy*, volume 7577 of *Lecture Notes in Computer Science*, pages 611–625. Springer, 2012.

[4] Yuanzhouhan Cao, Tianqi Zhao, Ke Xian, Chunhua Shen, Zhiguo Cao, and Shugong Xu. Monocular depth estimation with augmented ordinal depth relationships. *IEEE Trans. Circuits Syst. Video Technol.*, 30(8):2674–2682, 2020.

[5] Weifeng Chen, Zhao Fu, Dawei Yang, and Jia Deng. Single-image depth perception in the wild. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems, December 5-10, 2016, Barcelona, Spain*, pages 730–738, 2016.

[6] Weifeng Chen, Shengyi Qian, and Jia Deng. Learning single-image depth from videos using quality assessment networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 16-20, 2019, Long Beach, CA, USA*, pages 5604–5613. Computer Vision Foundation / IEEE, 2019.

[7] Weifeng Chen, Shengyi Qian, David Fan, Noriyuki Kojima, Max Hamilton, and Jia Deng. OASIS: A large-scale dataset for single image 3d in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 13-19, 2020, Seattle, WA, USA*, pages 676–685. IEEE, 2020.

[8] Weiwei Cheng, Krzysztof Dembczynski, and Eyke Hüllermeier. Label ranking methods based on the Plackett-Luce model. In Johannes Fürnkranz and Thorsten Joachims, editors, *Proceedings of the 27th International Conference on Machine Learning ICML, June 21-24, 2010, Haifa, Israel*, pages 215–222. Omnipress, 2010.

[9] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems, December 8-13, 2014, Montreal, Quebec, Canada*, pages 2366–2374, 2014.

[10] Ralph Ewerth, Matthias Springstein, Eric Müller, Alexander Balz, Jan Gehlhaar, Tolga Naziyok, Krzysztof Dembczynski, and Eyke Hüllermeier. Estimating relative depth in single images via RankBoost. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), July 10-14, 2017, Hong Kong, China*, pages 919–924. IEEE Computer Society, 2017.

[11] Yoav Freund, Raj D. Iyer, Robert E. Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. *J. Mach. Learn. Res.*, 4:933–969, 2003.

[12] J. Fürnkranz and E. Hüllermeier, editors. *Preference Learning*. Springer-Verlag, 2011.

[13] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *Int. J. Robotics Res.*, 32(11):1231–1237, 2013.

[14] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October 27 - November 2, 2019, Seoul, Korea (South)*, pages 3827–3837. IEEE, 2019.

[15] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 23-26, 2002, Edmonton, Alberta, Canada*, pages 133–142. ACM, 2002.

[16] Tobias Koch, Lukas Liebel, Friedrich Fraundorfer, and Marco Körner. Evaluation of CNN-based single-image depth estimation methods. In Laura Leal-Taixé and Stefan Roth, editors, *Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Part III, September 8-14, 2018, Munich, Germany*, volume 11131 of *Lecture Notes in Computer Science*, pages 331–348. Springer, 2018.

[17] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *Proceedings of the 4th International Conference on 3D Vision (3DV), October 25-28, 2016, Stanford, CA, USA*, pages 239–248. IEEE Computer Society, 2016.

[18] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *CoRR*, abs/1907.10326, 2019.

[19] Zhengqi Li, Tali Dekel, Forrester Cole, Richard Tucker, Noah Snavely, Ce Liu, and William T. Freeman. Learning the depths of moving people by watching frozen people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 16-20, 2019, Long Beach, CA, USA*, pages 4521–4530. Computer Vision Foundation / IEEE, 2019.

[20] Zhengqi Li and Noah Snavely. MegaDepth: Learning single-

view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 18-22, 2018, Salt Lake City, UT, USA*, pages 2041–2050. IEEE Computer Society, 2018.

[21] Tie-Yan Liu. Learning to rank for information retrieval. In Fabio Crestani, Stéphane Marchand-Maillet, Hsin-Hsi Chen, Efthimis N. Efthimiadis, and Jacques Savoy, editors, *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), July 19-23, 2010, Geneva, Switzerland*, page 904. ACM, 2010.

[22] R. Duncan Luce. *Individual Choice Behavior: A Theoretical analysis*. Wiley, New York, NY, USA, 1959.

[23] D. Mcfadden. Econometric models for probabilistic choice among products. *The Journal of Business*, 53:13–29, 1980.

[24] R. Plackett. The analysis of permutations. *Journal of The Royal Statistical Society Series C-applied Statistics*, 24:193–202, 1975.

[25] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[26] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In Andrew W. Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Proceedings of the 12th European Conference on Computer Vision (ECCV), Part V, October 7-13, 2012, Florence, Italy*, volume 7576 of *Lecture Notes in Computer Science*, pages 746–760. Springer, 2012.

[27] Hossein Azari Soufiani, David C. Parkes, and Lirong Xia. Random utility theory for social choice. In Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pages 126–134, 2012.

[28] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), October 7-12, 2012, Vilamoura, Algarve, Portugal*, pages 573–580. IEEE, 2012.

[29] Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning (ICML), 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 6105–6114. PMLR, 2019.

[30] Igor Vasiljevic, Nicholas I. Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z. Dai, Andrea F. Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R. Walter, and Gregory Shakhnarovich. DIODE: A dense indoor and outdoor depth dataset. *CoRR*, abs/1908.00463, 2019.

[31] Chaoyang Wang, Simon Lucey, Federico Perazzi, and Oliver Wang. Web stereo video supervision for depth prediction from dynamic scenes. In *Proceedings of the 7th International Conference on 3D Vision (3DV), September 16-19, 2019, Québec City, QC, Canada*, pages 348–357. IEEE, 2019.

[32] Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: Theory and algorithm. In William W. Cohen, Andrew McCallum, and Sam T. Roweis, editors, *Proceedings of the 25th International Conference on Machine Learning ICML, Helsinki, Finland, June 5-9, 2008*, volume 307 of *ACM International Conference Proceeding Series*, pages 1192–1199. ACM, 2008.

[33] Ke Xian, Chunhua Shen, Zhiguo Cao, Hao Lu, Yang Xiao, Ruibo Li, and Zhenbo Luo. Monocular relative depth perception with web stereo data supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (CVPR), June 18-22, 2018, Salt Lake City, UT, USA*, pages 311–320. IEEE Computer Society, 2018.

[34] Ke Xian, Jianming Zhang, Oliver Wang, Long Mai, Zhe Lin, and Zhiguo Cao. Structure-guided ranking loss for single image depth prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 13-19, 2020, Seattle, WA, USA*, pages 608–617. IEEE, 2020.

[35] Daniel Zoran, Phillip Isola, Dilip Krishnan, and William T. Freeman. Learning ordinal relationships for mid-level vision. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV), December 7-13, 2015, Santiago, Chile*, pages 388–396. IEEE Computer Society, 2015.

11 Conclusion and Outlook

In this thesis, we propose a novel perspective on model optimization in machine learning applications, emphasizing the value of data modeling as a means to achieve robustness for a learning model. Rather than adjusting the model class and its optimization procedure to accommodate the available data, we advocate for a paradigm shift wherein the data is made compatible with the learning model by adapting the label information, rather than the other way around as is commonly done. In this course, we considered two types of weakening supervision, namely by means of *supersets* and *ordinal relations*. After providing an overview of conventional supervised learning methods and thoroughly discussing methods in the field of weakly-supervised learning that serve as a foundation for our own proposals, we presented concrete implementations of the two data modeling approaches across diverse model types and domains.

In Chapter 4, we introduced label relaxation (LR) as a flexible framework for modeling probabilistic target information, which replaces single probability distributions with less precise but more reliable credal sets, thereby reducing the negative impact of potential biases in the training data by promoting their compatibility with the learning model. Our studies have shown that learning from credal labels in the context of superset learning not only improves critical quality dimensions of models, such as their calibration, but also allows for an expressive knowledge representation in the face of uncertainty. Especially when combined with large neural networks, this methodology has established the paradigm of credal label learning with great potential. As an instantiation of this paradigm for semi-supervised learning, credal pseudo-labeling, as described in Chapters 5 and 6, addresses limitations in the expressiveness of commonly used precise probabilistic pseudo-labels. By replacing precise pseudo-labels with credal sets, it relieves from the need of complex model adaptations, such as to cope with unreliable pseudo-labels, through a more cautious and inherently uncertainty-aware self-labeling procedure, thereby mitigating confirmation biases. Furthermore, we show with our robust data ambiguation proposal in Chapter 7 that credal label learning can serve as the basis for a dynamic denoising procedure to deal with label noise, which ambiguates the label information of instances that appear to be mislabeled, thereby improving the robustness to corrupt labels.

While the aforementioned works are probabilistic instantiations of the optimistic superset loss, we revisited the latter in Chapter 8 in a more general context for instance weighting. There, we described how the influence of individual instances on the overall loss minimization can be controlled by the degree of label imprecisiation. In the face of label noise, this methodology has demonstrated its robustness to misleading label information and has been successfully applied to the problem of semi-supervised learning, yielding a large-margin loss function as a special case for a specific target modeling. Furthermore, this approach has been applied to monocular depth estimation in Chapter 9, where data imprecisiation allows models to cope with violations of traditional statistical assumptions in regression. In this course, fuzzy sets have also been used as a more flexible generalization of classical sets, allowing for an even more tailored modeling of the knowledge about the targets.

Finally, we devised a novel perspective on monocular depth estimation in Chapter 10 by treating it as a preference learning problem, proposing to model target information by lists of relative comparisons as a form of weak supervision. We have argued that weakening metric labels by ordinal labels can reduce the requirements on the quality of employed training data, allowing for tapping weaker and thus less expensive sources of label information. Furthermore, we have shown that the Plackett-Luce model as a random utility model can be used to also retrieve metric information up to a scale, yielding competitive performance in this respect.

11.1 Future Research Directions

In the last section, we suggest potential courses for future research directions, extending the groundwork laid by this thesis and inspiring further advancements to bolster the robustness of machine learning models through the (re-)modeling of training data. We not only outline abstract directions stemming from our contributions but also connect them with promising ideas, poised to address challenges arising from our method proposals.

Generalizing and rigorizing relaxation. So far, label relaxation for constructing credal sets as introduced in Chapter 4 has followed a rather simplistic approach. The imprecisiation degree modeled as a hyperparameter is shared by all classes except a reference class with full possibility. Although we have considered credal sets with arbitrary underlying possibility distributions in Chapter 6 in the context of semi-supervised learning, in which plausibilities naturally arise from a conformal uncertainty quantification, a more rigorous investigation of alternative relaxation strategies is needed as future work. For example, label relaxation in its standard form ignores semantic similarities between classes, not deeming some classes more plausible than others given an instance of a reference class. This issue has been addressed in belief function theory by contextual discounting, which we discussed in Section 3.2.3, and has already been discussed in the context of imprecise probabilities [Des10]. Exploiting semantic similarities further relates to more structured prediction problems, such as ordinal classification, for which adaptations of label smoothing (cf. Section 2.2 and Chapter 4) have been considered [Liu+20b; Var+23]. Again, LR seems to be a promising alternative to label smoothing for such problems, too. Moreover, our framework of learning from credal labels can also be linked to robust statistics via neighborhood models [MMD19], allowing to generalize the notion of relaxation to arbitrary distributions. Although in a slightly different setting, first steps in this direction have been ventured [Dew+23]. Particularly well illustrated in the case of the linear-vacuous model [Wal91], which contaminates precise probabilistic information by agnostic evidence based on a parameterizable distortion degree, the construction of credal sets using such models can be viewed as a natural counterpart to label smoothing in possibilistic spaces. Finally, relaxation must not be necessarily limited to the imprecisiation of label information, but could be also applied at a feature level [Hül14]. Albeit potentially increasing the complexity of the optimization problem, especially when facing high dimensionality, it appears to be a promising complement to LR at a target level as introduced before. Overall, considering the aforementioned approaches could lead to a more general instantiation of our relaxation idea.

Optimizing generalized credal learning. Generalizing credal labeling to arbitrary sets may require a reconsideration of the analytical loss formulation of LR. While the use of simplistic credal sets as in Chapters 4, 5 and 7 leads to a closed-form analytical solution of the optimistic superset loss, whose convexity has been formally proved, more sophisticated credal set boundaries might not necessarily be trivial to integrate into it. In Chapter 6, we presented an algorithmic solution for generalizing LR to credal sets imposed by arbitrary (normalized) possibility distributions, yielding the provably optimal solution for the optimistic superset loss. However, the algorithmic solution, although lower on average, has a cubic complexity with respect to the number of classes in the worst case, rendering it rather inefficient in large-scale applications. Future work may focus on exploiting knowledge about extreme points of the credal sets, or on developing approximations as is common in constrained convex optimization.

Quantifying uncertainty more efficiently. The construction of credal labels is closely related to the quantification of aleatoric and epistemic uncertainty [HW21]. Credal sets are considered as a representation of uncertainty [HDS22], e.g., in second-order predictive models [BHW23] like our models in Chapters 5 and 6. While the aleatoric uncertainty is assumed to be irreducible, the reduction of its epistemic counterpart is crucial for strong generalization and robustness. To this end, especially in self-labeling for semi-supervised learning to mitigate the negative influence of confirmation biases, a model needs to be aware of its own epistemic uncertainty. Therefore, we leverage (inductive) conformal prediction as a method of uncertainty quantification in Chapter 6, leading to credal pseudo-labels of higher quality. However, this approach comes at the cost of additional validation instances, which need to be separated from the precise training data. This is particularly problematic in label scarce settings, since smaller validation datasets typically lead to an inefficient, less accurate uncertainty quantification by an inductive conformal prediction procedure. As is also being relevant in applications as described in Chapter 7, an appropriate epistemic uncertainty quantification method could be used as a viable misclassification filter criterion that can be tailored to the learner's knowledge in different sub-regions of the instance space. Future work could address this issue through a (credal) pseudo-labeling mechanism that mitigates such shortcomings, for instance by developing models that are inherently capable of expressing their uncertainty [KG17]. Coupling the epistemics of a model tighter with its self-supervision in a principled manner could lead to a more data-efficient way for self-training, thus broadening the applicability of this method.

Pushing data disambiguation to the limits. All bespoken instantiations of data disambiguation by means of the optimistic superset loss (OSL) facilitate the disambiguation of ambiguous data in an optimistic sense, which can become problematic for models with quasi-infinite expressiveness, such as extremely large models. For example, today's large-scale deep neural networks with billions of parameters are capable of overfitting large amounts of data. In this case, the optimization of the OSL is not guaranteed to achieve a plausible instantiation of data in light of the overall loss minimization and fails to resolve potential biases in the data. As discussed in Section 3.2.2, this problem has been addressed by a regularized OSL variant, where certain parameter subspaces are penalized. However, to date, concrete realizations of the latter have not yet been thoroughly studied, especially not in the realm of deep learning at a large scale. Future works on regularizing models while preserving data disambiguation capabilities could address critical practical concerns, such as the adversarial robustness of generative language models [Wan+21], which are penetrating more and more areas of life. Alternatively, adversarial robustness could also be approached by constructing additional data to learn from [Sch+18], for which data programming as described in Section 3.1.3 appears promising. As a particularly cost-effective instantiation of the latter, heuristics could be developed that return labels in form of supersets, thereby lowering the requirements on labeling functions and providing a more cautious yet correct supervision to learn from. Since data of this kind can be directly leveraged by generalized learning models as those trained with the OSL formulation, and no label model must be explicitly devised for denoising, such a methodology could drastically scale up training data.

Integrating the two proposed forms of weakening. In this thesis, the two main branches of superset-based and ordinal weakening are distinguished. Naturally, partial rankings, which represent incomplete relative orders among a set of items, as often observed in practice, can be regarded as a superset consisting of all possible linear extensions, i.e., possible completions of the rankings that preserve the order of relations available in the partial ranking, connecting the two branches. As a first approach exploiting this observation to bridge the gap, [HC15] reduces the problem of learning from partial rankings within optimistic superset learning to standard classification problems in a meta-learning formulation. However, it does not directly approach minimizing the optimistic superset loss as in our proposed methods, which is considered computationally too complex. A possible direction for further research on solving this problem could be an integration of relative information with its probabilistic modeling, for which first ideas have been discussed in [MD13]. For example, credal sets constructed to model relative information could be employed in a credal labeling approach similar to label relaxation, thus aiming directly at minimizing the optimistic superset loss. Beyond integrating the pillars of superset and preference learning as a potential direction for future research, data programming to construct relative comparisons as a form of weak supervision may allow additional data sources to be accessible for training. By not requiring to commit to precise values, as previously discussed in the context of superset heuristics, label denoising is likely to become less complex, potentially resulting in a cheap and reliable supporting target source.

Bibliography

[ADH17] Ehsan Abbasnejad, Anthony R. Dick, and Anton van den Hengel. “Infinite Variational Autoencoder for Semi-Supervised Learning”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, July 21-26, Honolulu, HI, USA*. IEEE Computer Society, 2017, pp. 781–790 (cit. on pp. 47, 57).

[AML12] Yaser S. Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin. *Learning From Data*. AMLBook, 2012 (cit. on pp. 1, 9, 10, 58).

[AS10] Fabio Aiolli and Alessandro Sperduti. “A Preference Optimization Based Unifying Framework for Supervised Learning Problems”. In: *Preference Learning*. Springer, 2010, pp. 19–42.

[AC19] T. M. Feroz Ali and Subhasis Chaudhuri. “A Semi-Supervised Maximum Margin Metric Learning Approach for Small Scale Person Re-Identification”. In: *Proc. of the IEEE International Conference on Computer Vision, ICCV, October 27 - November 2, Seoul, Korea (South), Workshops*. IEEE, 2019, pp. 1848–1857 (cit. on p. 52).

[AHH19] Christoph Alt, Marc Hübner, and Leonhard Hennig. “Fine-Tuning Pre-Trained Transformer Language Models to Distantly Supervised Relation Extraction”. In: *Proc. of the 57th Conference of the Association for Computational Linguistics, ACL, July 28 - August 2, Florence, Italy, Volume 1: Long Papers*. Association for Computational Linguistics, 2019, pp. 1388–1398 (cit. on p. 28).

[AMB05] Yasemin Altun, David Allen McAllester, and Mikhail Belkin. “Maximum Margin Semi-Supervised Learning for Structured Variables”. In: *Advances in Neural Information Processing Systems 18: Annual Conference on Neural Information Processing Systems, NIPS, December 5-8, Vancouver, BC, Canada*. MIT Press, 2005 (cit. on p. 50).

[AGU15] Ehsan Amid, Aristides Gionis, and Antti Ukkonen. “A Kernel-Learning Approach to Semi-Supervised Clustering with Relative Distance Comparisons”. In: *Proc. of the 26th European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, September 7-11, Porto, Portugal, Part I*. Vol. 9284. Lecture Notes in Computer Science. Springer, 2015, pp. 219–234 (cit. on p. 62).

[Amo13] Jaume Amores. “Multiple instance classification: Review, Taxonomy and Comparative Study”. In: *Artif. Intell.* 201 (2013), pp. 81–105 (cit. on p. 24).

[Amo10] Jaume Amores. “Vocabulary-Based Approaches for Multiple-Instance Data: A Comparative Study”. In: *Proc. of the 20th International Conference on Pattern Recognition, ICPR, August 23-26, Istanbul, Turkey*. IEEE Computer Society, 2010, pp. 4246–4250 (cit. on p. 25).

[ATH02] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. “Support Vector Machines for Multiple-Instance Learning”. In: *Advances in Neural Information Processing Systems 15: Annual Conference on Neural Information Processing Systems, NIPS, December 9-14, Vancouver, BC, Canada*. MIT Press, 2002, pp. 561–568 (cit. on p. 25).

[Ara+20] Eric Arazo, Diego Ortego, Paul Albert, Noel E. O’Connor, and Kevin McGuinness. “Pseudo-Labeling and Confirmation Bias in Deep Semi-Supervised Learning”. In: *Proc. of the International Joint Conference on Neural Networks, IJCNN, July 19-24, Glasgow, United Kingdom*. IEEE, 2020, pp. 1–8 (cit. on p. 54).

[Are10] Robert Arens. “Learning SVM Ranking Functions from User Feedback Using Document Metadata and Active Learning in the Biomedical Domain”. In: *Preference Learning*. Springer, 2010, pp. 363–383 (cit. on p. 61).

[Aug+14] Thomas Augustin, Frank P.A. Coolen, Gert De Cooman, and Matthias CM Troffaes. *Introduction to Imprecise Probabilities*. John Wiley & Sons, 2014 (cit. on p. 40).

[Awa+20] Abhijeet Awasthi, Sabyasachi Ghosh, Rasna Goyal, and Sunita Sarawagi. “Learning from Rules Generalizing Labeled Exemplars”. In: *8th International Conference on Learning Representations, ICLR, April 26 - May 1, Addis Ababa, Ethiopia*. OpenReview.net, 2020 (cit. on p. 29).

[BAP14] Philip Bachman, Ouais Alsharif, and Doina Precup. “Learning with Pseudo-Ensembles”. In: *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems, NIPS, December 8-13, Montreal, QC, Canada*. Curran Associates, Inc., 2014, pp. 3365–3373 (cit. on pp. 46, 54).

[Bae+23] Jongbeom Baek, Gyeongnyeon Kim, Seonghoon Park, Honggyu An, Matteo Poggi, and Seungryong Kim. “MaskingDepth: Masked Consistency Regularization for Semi-Supervised Monocular Depth Estimation”. In: (2023). arXiv: `2212.10806 [cs.CV]` (cit. on p. 54).

[Bal+23] Randall Balestriero, Mark Ibrahim, Vlad Sobal, Ari Morcos, Shashank Shekhar, Tom Goldstein, Florian Bordes, Adrien Bardes, Grégoire Mialon, Yuandong Tian, Avi Schwarzschild, Andrew Gordon Wilson, Jonas Geiping, Quentin Garrido, Pierre Fernandez, Amir Bar, Hamed Pirsiavash, Yann LeCun, and Micah Goldblum. “A Cookbook of Self-Supervised Learning”. In: *CoRR* abs/2304.12210 (2023) (cit. on p. 48).

[Bar19] Jonathan T. Barron. “A General and Adaptive Robust Loss Function”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 16-20, Long Beach, CA, USA*. Computer Vision Foundation / IEEE, 2019, pp. 4331–4339 (cit. on p. 17).

[BW87] Eric B. Baum and Frank Wilczek. “Supervised Learning of Probability Distributions by Neural Networks”. In: *Advances in Neural Information Processing Systems 0: Annual Conference on Neural Information Processing Systems, NIPS, Denver, CO, USA*. American Institue of Physics, 1987, pp. 52–61 (cit. on p. 13).

[Bax00] Jonathan Baxter. “A Model of Inductive Bias Learning”. In: *J. Artif. Intell. Res.* 12 (2000), pp. 149–198 (cit. on p. 2).

[BD20] Jessa Bekker and Jesse Davis. “Learning from Positive and Unlabeled Data: A Survey”. In: *Mach. Learn.* 109.4 (2020), pp. 719–760 (cit. on p. 23).

[BRD19] Jessa Bekker, Pieter Robberechts, and Jesse Davis. “Beyond the Selected Completely at Random Assumption for Learning from Positive and Unlabeled Data”. In: *Proc. of the 30th European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, September 16-20, Würzburg, Germany, Part II*. Vol. 11907. Lecture Notes in Computer Science. Springer, 2019, pp. 71–85 (cit. on p. 23).

[Bel+15] Vasileios Belagiannis, Christian Rupprecht, Gustavo Carneiro, and Nassir Navab. “Robust Optimization for Deep Regression”. In: *Proc. of the IEEE International Conference on Computer Vision, ICCV, December 7-13, Santiago, Chile*. IEEE Computer Society, 2015, pp. 2830–2838 (cit. on p. 17).

[BN06] Mikhail Belkin and Partha Niyogi. “Convergence of Laplacian Eigenmaps”. In: *Advances in Neural Information Processing Systems 19: Annual Conference on Neural Information Processing Systems, NIPS, December 4-7, Vancouver, BC, Canada*. MIT Press, 2006, pp. 129–136 (cit. on p. 47).

[BNS06] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. “Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples”. In: *J. Mach. Learn. Res.* 7 (2006), pp. 2399–2434 (cit. on pp. 47, 55).

[BLP08] Shai Ben-David, Tyler Lu, and Dávid Pál. “Does Unlabeled Data Provably Help? Worst-Case Analysis of the Sample Complexity of Semi-Supervised Learning”. In: *Proc. of the 21st Annual ACM Conference on Computational Learning Theory, COLT, July 9-12, Helsinki, Finland*. Omnipress, 2008, pp. 33–44 (cit. on p. 45).

[Ben+09] Shai Ben-David, Tyler Lu, Dávid Pál, and Miroslava Sotáková. “Learning Low Density Separators”. In: *Proc. of the 12th International Conference on Artificial Intelligence and Statistics, AISTATS, April 16-18, Clearwater Beach, FL, USA*. Vol. 5. JMLR Proc. JMLR.org, 2009, pp. 25–32 (cit. on p. 50).

[BHW23] Viktor Bengs, Eyke Hüllermeier, and Willem Waegeman. “On Second-Order Scoring Rules for Epistemic Uncertainty Quantification”. In: *CoRR* abs/2301.-12736 (2023) (cit. on p. 174).

[BD98] Kristin P. Bennett and Ayhan Demiriz. “Semi-Supervised Support Vector Machines”. In: *Advances in Neural Information Processing Systems 11: Annual Conference on Neural Information Processing Systems, NIPS, November 30 - December 5, Denver, CO, USA*. MIT Press, 1998, pp. 368–374 (cit. on pp. 47, 51).

[Ber+21] Patrice Bertail, Stéphan Clémençon, Yannick Guyonvarch, and Nathan Noiry. “Learning from Biased Data: A Semi-Parametric Approach”. In: *Proc. of the 38th International Conference on Machine Learning, ICML, July 18-24, virtual*. Vol. 139. Proc. of Machine Learning Research. PMLR, 2021, pp. 803–812 (cit. on p. 18).

[Ber+20] David Berthelot, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. “ReMixMatch: Semi-Supervised Learning with Distribution Matching and Augmentation Anchoring”. In: *8th International Conference on Learning Representations, ICLR, April 26 - May 1, Addis Ababa, Ethiopia*. OpenReview.net, 2020 (cit. on pp. 49, 50, 54).

[Ber+19] David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. “MixMatch: A Holistic Approach to Semi-Supervised Learning”. In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS, December 8-14, Vancouver, BC, Canada*. Curran Associates, Inc., 2019, pp. 5050–5060 (cit. on pp. 50, 52, 54).

[BLP98] James R. Bettman, Mary Frances Luce, and John W. Payne. “Constructive Consumer Choice Processes”. In: *J. Consum. Res.* 25.3 (1998), pp. 187–217 (cit. on p. 61).

[BW75] Peter Bloomfield and Geoffrey S. Watson. “The Inefficiency of Least Squares”. In: *Biometrika* 62.1 (1975), pp. 121–128 (cit. on p. 17).

[BM98] Avrim Blum and Tom M. Mitchell. “Combining Labeled and Unlabeled Data with Co-Training”. In: *Proc. of the 11th Annual ACM Conference on Computational Learning Theory, COLT, July 24-26, Madison, WI, USA*. ACM, 1998, pp. 92–100 (cit. on p. 49).

[Boe+21] Benedikt Boecking, Willie Neiswanger, Eric P. Xing, and Artur Dubrawski. “Interactive Weak Supervision: Learning Useful Heuristics for Data Labeling”. In: *9th International Conference on Learning Representations, ICLR, May 3-7, virtual*. OpenReview.net, 2021 (cit. on p. 29).

[Bor+15] Hanen Borchani, Gherardo Varando, Concha Bielza, and Pedro Larrañaga. “A Survey on Multi-Output Regression”. In: *WIREs Data Mining Knowl. Discov.* 5.5 (2015), pp. 216–233 (cit. on p. 15).

[Bor+18] Gerda Bortsova, Florian Dubost, Silas N. Ørting, Ioannis Katramados, Laurens Hogeweg, Laura H. Thomsen, Mathilde M. W. Wille, and Marleen de Bruijne. “Deep Learning from Label Proportions for Emphysema Quantification”. In: *Proc. of the 21st International Conference on Medical Image Computing and Computer Assisted Intervention, MICCAI, September 16-20, Granada, Spain, Part II*. Vol. 11071. Lecture Notes in Computer Science. Springer, 2018, pp. 768–776 (cit. on p. 26).

[BGV92] Bernhard E. Boser, Isabelle Guyon, and Vladimir Vapnik. “A Training Algorithm for Optimal Margin Classifiers”. In: *Proc. of the 5th Annual ACM Conference on Computational Learning Theory, COLT, July 27-29, Pittsburgh, PA, USA*. ACM, 1992, pp. 144–152 (cit. on p. 12).

[Bra08a] Daren C. Brabham. “Crowdsourcing as a Model for Problem Solving”. In: *Convergence* 14 (2008), pp. 75–90 (cit. on p. 26).

[Bra08b] Ronen I. Brafman. “Preferences, Planning and Control”. In: *Proc. of the 11th International Conference on Principles of Knowledge Representation and Reasoning, KR, September 16-19, Sydney, Australia*. AAAI Press, 2008, pp. 2–5 (cit. on p. 60).

[Bro+20] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. “Language Models are Few-Shot Learners”. In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, virtual*. Curran Associates, Inc., 2020, pp. 1877–1901 (cit. on pp. 12, 48).

[BS20] Michael C. Burkhart and Kyle Shan. “Deep Low-Density Separation for Semi-Supervised Classification”. In: *Proc. of the 20th International Conference on Computational Science, ICCS, June 3-5, Amsterdam, The Netherlands, Part III*. Vol. 12139. Lecture Notes in Computer Science. Springer, 2020, pp. 297–311 (cit. on p. 47).

[Cab22] Vivien Cabannes. “From Weakly Supervised Learning to Active Labeling”. Theses. Ecole Normale Supérieure (ENS), 2022 (cit. on pp. 21, 26).

[Cab+23] Vivien Cabannes, Bobak Toussi Kiani, Randall Balestriero, Yann LeCun, and Alberto Bietti. “The SSL Interplay: Augmentations, Inductive Bias, and Generalization”. In: *CoRR* abs/2302.02774 (2023) (cit. on p. 48).

[Cab+21] Vivien Cabannes, Loucas Pillaud-Vivien, Francis R. Bach, and Alessandro Rudi. “Overcoming the Curse of Dimensionality with Laplacian Regularization in Semi-Supervised Learning”. In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-14, virtual*. Curran Associates, Inc., 2021, pp. 30439–30451 (cit. on p. 55).

[CRB20] Vivien Cabannes, Alessandro Rudi, and Francis R. Bach. “Structured Prediction with Partial Labelling through the Infimum Loss”. In: *Proc. of the 37th International Conference on Machine Learning, ICML, July 13-18, virtual*. Vol. 119. Proc. of Machine Learning Research. PMLR, 2020, pp. 1230–1239 (cit. on pp. 31, 33, 35–37).

[CBR21] Vivien A. Cabannes, Francis R. Bach, and Alessandro Rudi. “Disambiguation of Weak Supervision leading to Exponential Convergence Rates”. In: *Proc. of the 38th International Conference on Machine Learning, ICML, July 18-24, virtual*. Vol. 139. Proc. of Machine Learning Research. PMLR, 2021, pp. 1147–1157 (cit. on p. 34).

[Cai+22] Zhaowei Cai, Avinash Ravichandran, Paolo Favaro, Manchen Wang, Davide Modolo, Rahul Bhotika, Zhuowen Tu, and Stefano Soatto. “Semi-Supervised Vision Transformers at Scale”. In: *CoRR* abs/2208.05688 (2022) (cit. on p. 49).

[Cai+21] Zhaowei Cai, Avinash Ravichandran, Subhransu Maji, Charless C. Fowlkes, Zhuowen Tu, and Stefano Soatto. “Exponential Moving Average Normalization for Self-Supervised and Semi-Supervised Learning”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 19-25, virtual*. Computer Vision Foundation / IEEE, 2021, pp. 194–203 (cit. on p. 54).

[Cam21] Andrea Campagner. “Learnability in "Learning from Fuzzy Labels"”. In: *Proc. of the 30th IEEE International Conference on Fuzzy Systems, FUZZ-IEEE, July 11-14, Luxembourg*. IEEE, 2021, pp. 1–6 (cit. on p. 39).

[Cam23] Andrea Campagner. “Learning from Fuzzy Labels: Theoretical Issues and Algorithmic Solutions”. In: *Int. J. Approx. Reason.* (2023), p. 108969 (cit. on p. 38).

[Cam+21] Andrea Campagner, Davide Ciucci, Carl-Magnus Svensson, Marc Thilo Figge, and Federico Cabitza. “Ground Truthing from Multi-Rater Labeling with Three-way Decision and Possibility Theory”. In: *Inf. Sci.* 545 (2021), pp. 771–790 (cit. on p. 39).

[Cao+21] Yuzhou Cao, Lei Feng, Senlin Shu, Yitian Xu, Bo An, Gang Niu, and Masashi Sugiyama. “Multi-Class Classification from Single-Class Data with Confidences”. In: *CoRR* abs/2106.08864 (2021) (cit. on p. 13).

[Car+18] Marc-André Carbonneau, Veronika Cheplygina, Eric Granger, and Ghyslain Gagnon. “Multiple Instance Learning: A Survey of Problem Characteristics and Applications”. In: *Pattern Recognit.* 77 (2018), pp. 329–353 (cit. on pp. 24, 25, 291).

[Car+21] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. "Emerging Properties in Self-Supervised Vision Transformers". In: *Proc. of the IEEE/CVF International Conference on Computer Vision, ICCV, October 10-17, Montreal, QC, Canada*. IEEE, 2021, pp. 9630–9640 (cit. on p. 48).

[CTC20] Jing Chai, Ivor W. Tsang, and Weijie Chen. "Large Margin Partial Label Machine". In: *IEEE Trans. Neural Networks Learn. Syst.* 31.7 (2020), pp. 2594–2608 (cit. on p. 34).

[CSZ06] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. *Semi-Supervised Learning*. The MIT Press, 2006 (cit. on pp. 23, 44–46).

[CZ05] Olivier Chapelle and Alexander Zien. "Semi-Supervised Classification by Low Density Separation". In: *Proc. of the 10th International Workshop on Artificial Intelligence and Statistics, AISTATS, January 6-8, Bridgetown, Barbados*. Society for Artificial Intelligence and Statistics, 2005 (cit. on p. 47).

[CK10] Ratthachat Chatpatanasiri and Boonserm Kijsirikul. "A Unified Semi-Supervised Dimensionality Reduction Framework for Manifold Learning". In: *Neurocomputing* 73.10-12 (2010), pp. 1631–1640 (cit. on p. 55).

[Che+20] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. "Big Self-Supervised Models are Strong Semi-Supervised Learners". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, virtual*. Curran Associates, Inc., 2020, pp. 22243–22255 (cit. on pp. 48, 50).

[Che+16] Weifeng Chen, Zhao Fu, Dawei Yang, and Jia Deng. "Single-Image Depth Perception in the Wild". In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems, NIPS, December 5-10, Barcelona, Spain*. Curran Associates, Inc., 2016, pp. 730–738 (cit. on pp. 58, 62).

[Che+21] Hao Cheng, Zhaowei Zhu, Xingyu Li, Yifei Gong, Xing Sun, and Yang Liu. "Learning with Instance-Dependent Label Noise: A Sample Sieve Approach". In: *9th International Conference on Learning Representations, ICLR, May 3-7, virtual*. OpenReview.net, 2021 (cit. on p. 14).

[Cho+20] Yu-Ting Chou, Gang Niu, Hsuan-Tien Lin, and Masashi Sugiyama. "Unbiased Risk Estimators Can Mislead: A Case Study of Learning with Complementary Labels". In: *Proc. of the 37th International Conference on Machine Learning, ICML, July 13-18, virtual*. Vol. 119. Proc. of Machine Learning Research. PMLR, 2020, pp. 1929–1938 (cit. on p. 23).

[Cho+22] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. “PaLM: Scaling Language Modeling with Pathways”. In: *CoRR* abs/2204.02311 (2022) (cit. on p. 48).

[Chr+17] Paul F. Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. “Deep Reinforcement Learning from Human Preferences”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, December 4-9, Long Beach, CA, USA*. Curran Associates, Inc., 2017, pp. 4299–4307 (cit. on p. 62).

[Chu+21] Tianshu Chu, Xinmeng Li, Huy V. Vo, Ronald M. Summers, and Elena Sizikova. “Improving Weakly Supervised Lesion Segmentation using Multi-Task Learning”. In: *Medical Imaging with Deep Learning, July 7-9, Lübeck, Germany*. Vol. 143. Proc. of Machine Learning Research. PMLR, 2021, pp. 60–73 (cit. on p. 28).

[Cla+15] Marc Claesen, Frank De Smet, Pieter Gillard, Chantal Mathieu, and Bart De Moor. “Building Classifiers to Predict the Start of Glucose-Lowering Pharmacotherapy Using Belgian Health Expenditure Data”. In: *CoRR* abs/1504.07389 (2015) (cit. on p. 23).

[CL22] Stephan Clémençon and Pierre Laforgue. “Statistical Learning from Biased Training Samples”. In: *Electron. J. Stat.* 16.2 (2022), pp. 6086–6134 (cit. on p. 19).

[Cle79] William S. Cleveland. “Robust Locally Weighted Regression and Smoothing Scatterplots”. In: *J. Am. Stat. Assoc.* 74 (1979), pp. 829–836 (cit. on p. 18).

[Côm+09] Etienne Côme, Latifa Oukhellou, Thierry Denoeux, and Patrice Aknin. “Learning from Partially Supervised Data using Mixture Models and Belief Functions”. In: *Pattern Recognit.* 42.3 (2009), pp. 334–348 (cit. on p. 42).

[CV95] Corinna Cortes and Vladimir Vapnik. “Support-Vector Networks”. In: *Mach. Learn.* 20.3 (1995), pp. 273–297 (cit. on p. 47).

[Cou+09] Timothée Cour, Benjamin Sapp, Chris Jordan, and Benjamin Taskar. "Learning from Ambiguously Labeled Images". In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 20-25, Miami, FL, USA*. IEEE Computer Society, 2009, pp. 919–926 (cit. on p. 33).

[CST11] Timothée Cour, Benjamin Sapp, and Ben Taskar. "Learning from Partial Labels". In: *J. Mach. Learn. Res.* 12 (2011), pp. 1501–1536 (cit. on pp. 30–33, 36).

[Cou+19] Inés Couso, Christian Borgelt, Eyke Hüllermeier, and Rudolf Kruse. "Fuzzy Sets in Data Analysis: From Statistical Foundations to Machine Learning". In: *IEEE Comput. Intell. Mag.* 14.1 (2019), pp. 31–44 (cit. on p. 38).

[CD14] Inés Couso and Didier Dubois. "Statistical Reasoning with Set-Valued Information: Ontic vs. Epistemic Views". In: *Int. J. Approx. Reason.* 55.7 (2014), pp. 1502–1518 (cit. on p. 38).

[Cox58] David R Cox. "The Regression Analysis of Binary Sequences". In: *J. R. Stat. Soc., B: Stat. Methodol.* 20.2 (1958), pp. 215–232 (cit. on p. 12).

[CCC03] Fábio Gagliardi Cozman, Ira Cohen, and Marcelo Cesar Cirelo. "Semi-Supervised Learning of Mixture Models". In: *Proc. of the 20th International Conference on Machine Learning, ICML, August 21-24, Washington, DC, USA*. AAAI Press, 2003, pp. 99–106 (cit. on p. 56).

[Cro12] Kevin Crowston. "Amazon Mechanical Turk: A Research Tool for Organizations and Information Systems Scholars". In: *Proc. of the Working Conference on Shaping the Future of ICT Research: Methods and Approaches, IFIP WG 8.2, December 13-14, Tampa, FL, USA*. Vol. 389. IFIP Advances in Information and Communication Technology. Springer, 2012, pp. 210–221 (cit. on p. 27).

[Dai+21] Wenliang Dai, Samuel Cahyawijaya, Yejin Bang, and Pascale Fung. "Weakly-Supervised Multi-Task Learning for Multimodal Affect Recognition". In: *CoRR* abs/2104.11560 (2021) (cit. on p. 28).

[Dai+17] Zihang Dai, Zhilin Yang, Fan Yang, William W. Cohen, and Ruslan Salakhutdinov. "Good Semi-Supervised Learning That Requires a Bad GAN". In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, December 4-9, Long Beach, CA, USA*. Curran Associates, Inc., 2017, pp. 6510–6520 (cit. on p. 57).

[DDH83] Gerard Debreu, Gerard Debreu, and Werner Hildenbrand. "Representation of a Preference Ordering by a Numerical Function". In: *Mathematical Economics: Twenty Papers of Gerard Debreu*. Econometric Society Monographs. Cambridge: Cambridge University Press, 1983, pp. 105–110 (cit. on p. 60).

[Del+19] Manuel Fernández Delgado, Manisha Sanjay Sirsat, Eva Cernadas, Sadi Alawadi, Senén Barro, and Manuel Febrero-Bande. "An Extensive Experimental Survey of Regression Methods". In: *Neural Netw.* 111 (2019), pp. 11–34 (cit. on p. 17).

[Dem67] Arthur P. Dempster. “Upper and Lower Probabilities Induced by a Multivalued Mapping”. In: *Ann. Math. Stat.* 38.2 (1967), pp. 325–339 (cit. on p. 40).

[Den95] Thierry Denoeux. “A k-Nearest Neighbor Classification Rule based on Dempster-Shafer Theory”. In: *IEEE Trans. Syst. Man Cybern.* 25.5 (1995), pp. 804–813 (cit. on p. 42).

[Den14] Thierry Denoeux. “Likelihood-Based Belief Function: Justification and some Extensions to Low-Quality Data”. In: *Int. J. Approx. Reason.* 55.7 (2014), pp. 1535–1547 (cit. on p. 42).

[Den11] Thierry Denoeux. “Maximum Likelihood Estimation from Fuzzy Data using the EM Algorithm”. In: *Fuzzy Sets Syst.* 183.1 (2011), pp. 72–91 (cit. on p. 39).

[Den13] Thierry Denoeux. “Maximum Likelihood Estimation from Uncertain Data in the Belief Function Framework”. In: *IEEE Trans. Knowl. Data Eng.* 25.1 (2013), pp. 119–130 (cit. on pp. 33, 42).

[DKS19] Thierry Denoeux, Orakanya Kanjanatarakul, and Songsak Sriboonchitta. “A new Evidential *K*-Nearest Neighbor Rule based on Contextual Discounting with Partially Supervised Learning”. In: *Int. J. Approx. Reason.* 113 (2019), pp. 287–302 (cit. on p. 42).

[DZ01] Thierry Denoeux and Lalla Merieme Zouhal. “Handling Possibilistic Labels in Pattern Classification using Evidential Reasoning”. In: *Fuzzy Sets Syst.* 122.3 (2001), pp. 409–424 (cit. on pp. 40, 42).

[Der+17] Jan Deriu, Aurélien Lucchi, Valeria De Luca, Aliaksei Severyn, Simon Müller, Mark Cieliebak, Thomas Hofmann, and Martin Jaggi. “Leveraging Large Amounts of Weakly Supervised Data for Multi-Language Sentiment Classification”. In: *Proc. of the 26th International Conference on World Wide Web, WWW, April 3-7, Perth, Australia*. ACM, 2017, pp. 1045–1052 (cit. on p. 28).

[DGH14] Joaquín Derrac, Salvador García, and Francisco Herrera. “Fuzzy Nearest Neighbor Algorithms: Taxonomy, Experimental Analysis and Prospects”. In: *Inf. Sci.* 260 (2014), pp. 98–119 (cit. on p. 39).

[Der+18] Lucio Mwinmaarong Dery, Benjamin Nachman, Francesco Rubbo, and Ariel Schwartzman. “Weakly Supervised Classification For High Energy Physics”. In: *J. Phys. Conf. Ser.* 1085.4 (2018), p. 042006 (cit. on p. 26).

[Des10] Sébastien Destercke. “A New Contextual Discounting Rule for Lower Probabilities”. In: *Proc. of the 13th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU, June 28 - July 2, Dortmund, Germany, Part II*. Vol. 81. Communications in Computer and Information Science. Springer, 2010, pp. 198–207 (cit. on pp. 43, 173).

[Des22] Sébastien Destercke. “Uncertain Data in Learning: Challenges and Opportunities”. In: *Proc. of the 11th Symposium on Conformal and Probabilistic Prediction with Applications, COPA, August 24-26, Brighton, United Kingdom*. Vol. 179. Proc. of Machine Learning Research. PMLR, 2022, pp. 322–332 (cit. on p. 40).

[DDC08] Sébastien Destercke, Didier Dubois, and Eric Chojnacki. “Unifying Practical Uncertainty Representations - I: Generalized p-boxes”. In: *Int. J. Approx. Reason.* 49.3 (2008), pp. 649–663 (cit. on p. 40).

[Dew+23] Miheer Dewaskar, Christopher Tosh, Jeremias Knoblauch, and David B. Dunson. “Robustifying Likelihoods by Optimistically Re-Weighting Data”. In: (2023). arXiv: `2303.10525 [stat.ME]` (cit. on p. 173).

[DLL97] Thomas G. Dietterich, Richard H. Lathrop, and Tomás Lozano-Pérez. “Solving the Multiple Instance Problem with Axis-Parallel Rectangles”. In: *Artif. Intell.* 89.1-2 (1997), pp. 31–71 (cit. on p. 24).

[DZZ17] Shifei Ding, Zhibin Zhu, and Xiekai Zhang. “An Overview on Semi-Supervised Support Vector Machine”. In: *Neural Comput. Appl.* 28.5 (2017), pp. 969–978 (cit. on p. 51).

[Dom+11] Carmel Domshlak, Eyke Hüllermeier, Souhila Kaci, and Henri Prade. “Preferences in AI: An Overview”. In: *Artif. Intell.* 175.7-8 (2011), pp. 1037–1052 (cit. on p. 59).

[DR14] Gary Doran and Soumya Ray. “A Theoretical and Empirical Analysis of Support Vector Machine Methods for Multiple-Instance Classification”. In: *Mach. Learn.* 97.1-2 (2014), pp. 79–102 (cit. on p. 25).

[Dou11] Christopher Dougherty. *Introduction to Econometrics*. Oxford University Press, 2011 (cit. on p. 16).

[Du+21] Jingfei Du, Edouard Grave, Beliz Gunel, Vishrav Chaudhary, Onur Celebi, Michael Auli, Veselin Stoyanov, and Alexis Conneau. “Self-Training Improves Pre-Training for Natural Language Understanding”. In: *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, June 6-11, virtual*. Association for Computational Linguistics, 2021, pp. 5408–5418 (cit. on p. 49).

[DW21] Wei Du and Xintao Wu. “Fair and Robust Classification Under Sample Selection Bias”. In: *Proc. of the 30th ACM International Conference on Information and Knowledge Management, CIKM, November 1-5, virtual*. ACM, 2021, pp. 2999–3003 (cit. on p. 29).

[Dua+21] Jiali Duan, Yen-Liang Lin, Son Dinh Tran, Larry S. Davis, and C.-C. Jay Kuo. “SLADE: A Self-Training Framework for Distance Metric Learning”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 19-25, virtual*. Computer Vision Foundation / IEEE, 2021, pp. 9644–9653 (cit. on p. 48).

[DP98] Didier Dubois and Henri Prade. “Possibility Theory: Qualitative and Quantitative Aspects”. In: *Quantified Representation of Uncertainty and Imprecision*. Dordrecht: Springer Netherlands, 1998, pp. 169–226 (cit. on p. 40).

[DG20] Stefan Duffner and Christophe Garcia. “Multiple Instance Learning for Training Neural Networks under Label Noise”. In: *Proc. of the International Joint Conference on Neural Networks, IJCNN, July 19-24, Glasgow, United Kingdom*. IEEE, 2020, pp. 1–7 (cit. on p. 25).

[Dul+19] Gabriel Dulac-Arnold, Neil Zeghidour, Marco Cuturi, Lucas Beyer, and Jean-Philippe Vert. “Deep Multi-Class Learning from Label Proportions”. In: *CoRR* abs/1905.12909 (2019) (cit. on p. 26).

[EG10] Mohamed M. El-Zahhar and Neamat Farouk El Gayar. “A Semi-Supervised Learning Approach for Soft Labeled Data”. In: *Proc. of the 10th International Conference on Intelligent Systems Design and Applications, ISDA, November 29 - December 1, Cairo, Egypt*. IEEE, 2010, pp. 1136–1141 (cit. on p. 39).

[Ela+22] Badr Ben Elallid, Nabil Benamar, Abdelhakim Senhaji Hafid, Tajjeeddine Rachidi, and Nabil Mrani. “A Comprehensive Survey on the Application of Deep and Reinforcement Learning Approaches in Autonomous Driving”. In: *J. King Saud Univ. Comput. Inf. Sci.* 34.9 (2022), pp. 7366–7390 (cit. on p. 1).

[EGW10] Ehab E. Elattar, John Yannis Goulermas, and Q. Henry Wu. “Electric Load Forecasting Based on Locally Weighted Support Vector Regression”. In: *IEEE Trans. Syst. Man Cybern. Part C* 40.4 (2010), pp. 438–447 (cit. on p. 18).

[Ell61] Daniel Ellsberg. “Risk, Ambiguity, and the Savage Axioms”. In: *Q. J. Econ.* 75.4 (1961), pp. 643–669 (cit. on p. 58).

[EMS01] Zied Elouedi, Khaled Mellouli, and Philippe Smets. “Belief Decision Trees: Theoretical Foundations”. In: *Int. J. Approx. Reason.* 28.2-3 (2001), pp. 91–124 (cit. on p. 42).

[EH20] Jesper E. van Engelen and Holger H. Hoos. “A Survey on Semi-Supervised Learning”. In: *Mach. Learn.* 109.2 (2020), pp. 373–440 (cit. on pp. 44, 45, 50, 54–56).

[Ewe+17] Ralph Ewerth, Matthias Springstein, Eric Müller, Alexander Balz, Jan Gehlhaar, Tolga Naziyok, Krzysztof Dembczynski, and Eyke Hüllermeier. “Estimating Relative Depth in Single Images via RankBoost”. In: *Proc. of the IEEE International Conference on Multimedia and Expo, ICME, July 10-14, Hong Kong, China*. IEEE Computer Society, 2017, pp. 919–924 (cit. on p. 62).

[FHC17] Mohsen Ahmadi Fahandar, Eyke Hüllermeier, and Inés Couso. “Statistical Inference for Incomplete Ranking Data: The Case of Rank-Dependent Coarsening”. In: *Proc. of the 34th International Conference on Machine Learning, ICML, August 6-11, Sydney, NSW, Australia*. Vol. 70. Proc. of Machine Learning Research. PMLR, 2017, pp. 1078–1087 (cit. on p. 61).

[Fan+14] Kai Fan, Hongyi Zhang, Songbai Yan, Liwei Wang, Wensheng Zhang, and Jufu Feng. "Learning a Generative Classifier from Label Proportions". In: *Neurocomputing* 139 (2014), pp. 47–55 (cit. on p. 26).

[Fan+23] Yue Fan, Anna Kukleva, Dengxin Dai, and Bernt Schiele. "Revisiting Consistency Regularization for Semi-Supervised Learning". In: *Int. J. Comput. Vis.* 131.3 (2023), pp. 626–643 (cit. on p. 54).

[FCL14] Yuan Fang, Kevin Chen-Chuan Chang, and Hady Wirawan Lauw. "Graph-Based Semi-Supervised Learning: Realizing Pointwise Smoothness Probabilistically". In: *Proc. of the 31th International Conference on Machine Learning, ICML, June 21-26, Beijing, China*. Vol. 32. JMLR Workshop and Conference Proc. JMLR.org, 2014, pp. 406–414 (cit. on p. 46).

[Far+21] Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R. Arabnia. "A Brief Review of Domain Adaptation". In: *Advances in Data Science and Information Engineering*. Springer International Publishing, 2021, pp. 877–894 (cit. on p. 29).

[FZ20] Vitaly Feldman and Chiyuan Zhang. "What Neural Networks Memorize and Why: Discovering the Long Tail via Influence Estimation". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, virtual*. Curran Associates, Inc., 2020, pp. 2881–2891 (cit. on p. 34).

[FA18] Lei Feng and Bo An. "Leveraging Latent Label Distributions for Partial Label Learning". In: *Proc. of the 27th International Joint Conference on Artificial Intelligence, IJCAI, July 13-19, Stockholm, Sweden*. ijcai.org, 2018, pp. 2107–2113 (cit. on p. 34).

[FA19] Lei Feng and Bo An. "Partial Label Learning with Self-Guided Retraining". In: *Proc. of the 33rd AAAI Conference on Artificial Intelligence, AAAI, the 31st Innovative Applications of Artificial Intelligence Conference, IAAI, the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI, January 27 - February 1, Honolulu, HI, USA*. AAAI Press, 2019, pp. 3542–3549 (cit. on p. 34).

[Fen+20] Lei Feng, Jiaqi Lv, Bo Han, Miao Xu, Gang Niu, Xin Geng, Bo An, and Masashi Sugiyama. "Provably Consistent Partial-Label Learning". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, virtual*. Curran Associates, Inc., 2020, pp. 10948–10960 (cit. on p. 34).

[Fis70] Peter C Fishburn. *Utility Theory for Decision Making*. Tech. rep. Research analysis corp McLean VA, 1970 (cit. on p. 60).

[Fol92] Richard Foley. "Being Knowingly Incoherent". In: *Noûs* 26.2 (1992), pp. 181–203 (cit. on p. 58).

[Fu+20] Daniel Y. Fu, Mayee F. Chen, Frederic Sala, Sarah M. Hooper, Kayvon Fatahalian, and Christopher Ré. “Fast and Three-Rious: Speeding Up Weak Supervision with Triplet Methods”. In: *Proc. of the 37th International Conference on Machine Learning, ICML, July 13-18, virtual*. Vol. 119. Proc. of Machine Learning Research. PMLR, 2020, pp. 3280–3291 (cit. on p. 29).

[FZL18] Sheng Fu, Sanguo Zhang, and Yufeng Liu. “Adaptively Weighted Large-Margin Angle-Based Classifiers”. In: *J. Multivar. Anal.* 166 (2018), pp. 282–299 (cit. on p. 18).

[FH10] Johannes Fürnkranz and Eyke Hüllermeier. “Preference Learning: An Introduction”. In: *Preference Learning*. Springer, 2010, pp. 1–17 (cit. on pp. 58–60).

[Für+12] Johannes Fürnkranz, Eyke Hüllermeier, Weiwei Cheng, and Sang-Hyeun Park. “Preference-Based Reinforcement Learning: A Formal Framework and a Policy Iteration Algorithm”. In: *Mach. Learn.* 89.1-2 (2012), pp. 123–156 (cit. on p. 62).

[GGT21] Sainyam Galhotra, Behzad Golshan, and Wang-Chiew Tan. “Adaptive Rule Discovery for Labeling Text Data”. In: *Proc. of the International Conference on Management of Data, SIGMOD, June 20-25, virtual*. ACM, 2021, pp. 2217–2225 (cit. on p. 29).

[Gan+17] Zhe Gan, Liqun Chen, Weiyao Wang, Yunchen Pu, Yizhe Zhang, Hao Liu, Chunyuan Li, and Lawrence Carin. “Triangle Generative Adversarial Networks”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, December 4-9, Long Beach, CA, USA*. Curran Associates, Inc., 2017, pp. 5247–5256 (cit. on p. 57).

[GMG20] Yang Gao, Christian M. Meyer, and Iryna Gurevych. “Preference-Based Interactive Multi-Document Summarisation”. In: *Inf. Retr. J.* 23.6 (2020), pp. 555–585 (cit. on p. 62).

[Gei+13] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. “Vision meets Robotics: The KITTI Dataset”. In: *Int. J. Robotics Res.* 32.11 (2013), pp. 1231–1237 (cit. on pp. 16, 291).

[Gen+12] Bo Geng, Dacheng Tao, Chao Xu, Linjun Yang, and Xian-Sheng Hua. “Ensemble Manifold Regularization”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 34.6 (2012), pp. 1227–1233 (cit. on p. 55).

[GKS17] Aritra Ghosh, Himanshu Kumar, and P. S. Sastry. “Robust Loss Functions under Label Noise for Deep Neural Networks”. In: *Proc. of the 31st AAAI Conference on Artificial Intelligence, AAAI, February 4-9, San Francisco, CA, USA*. AAAI Press, 2017, pp. 1919–1925 (cit. on p. 14).

[Gho+21] Asish Ghoshal, Xilun Chen, Sonal Gupta, Luke Zettlemoyer, and Yashar Mehdad. “Learning Better Structured Representations Using Low-Rank Adaptive Label Smoothing”. In: *9th International Conference on Learning Representations, ICLR, May 3-7, virtual*. OpenReview.net, 2021 (cit. on p. 15).

[GS94] Itzhak Gilboa and David Schmeidler. “Additive Representations of Non-Additive Measures and the Choquet Integral”. In: *Ann. Oper. Res.* 52.1 (1994), pp. 43–65 (cit. on p. 40).

[GLS17] Amir Globerson, Roi Livni, and Shai Shalev-Shwartz. “Effective Semisupervised Learning on Manifolds”. In: *Proc. of the Conference on Learning Theory, COLT, July 7-10, Amsterdam, The Netherlands*. Vol. 65. Proc. of Machine Learning Research. PMLR, 2017, pp. 978–1003 (cit. on p. 47).

[Gol+09] Andrew B. Goldberg, Xiaojin Zhu, Aarti Singh, Zhiting Xu, and Robert D. Nowak. “Multi-Manifold Semi-Supervised Learning”. In: *Proc. of the 12th International Conference on Artificial Intelligence and Statistics, AISTATS, April 16-18, Clearwater Beach, FL, USA*. Vol. 5. JMLR Proc. JMLR.org, 2009, pp. 169–176 (cit. on p. 47).

[Gon+18] Chen Gong, Tongliang Liu, Yuanyan Tang, Jian Yang, Jie Yang, and Dacheng Tao. “A Regularization Approach for Instance-Based Superset Label Learning”. In: *IEEE Trans. Cybern.* 48.3 (2018), pp. 967–978 (cit. on p. 34).

[Gon+21] Chen Gong, Hong Shi, Tongliang Liu, Chuang Zhang, Jian Yang, and Dacheng Tao. “Loss Decomposition and Centroid Estimation for Positive and Unlabeled Learning”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 43.3 (2021), pp. 918–932 (cit. on p. 23).

[GWL21] Chengyue Gong, Dilin Wang, and Qiang Liu. “AlphaMatch: Improving Consistency for Semi-Supervised Learning With Alpha-Divergence”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 19-25, virtual*. Computer Vision Foundation / IEEE, 2021, pp. 13683–13692 (cit. on p. 54).

[Gon+22] Xiuwen Gong, Jiahui Yang, Dong Yuan, and Wei Bao. “Generalized Large Margin kNN for Partial Label Learning”. In: *IEEE Trans. Multim.* 24 (2022), pp. 1055–1066 (cit. on p. 34).

[GBC16] Ian J. Goodfellow, Yoshua Bengio, and Aaron C. Courville. *Deep Learning*. Adaptive Computation and Machine Learning. MIT Press, 2016 (cit. on p. 13).

[Goo+14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. “Generative Adversarial Nets”. In: *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems, NIPS, December 8-13, Montreal, QC, Canada*. Curran Associates, Inc., 2014, pp. 2672–2680 (cit. on p. 56).

[Gos+21] Jann Goschenhofer, Rasmus Hvingelby, David Rügamer, Janek Thomas, Moritz Wagner, and Bernd Bischl. “Deep Semi-Supervised Learning for Time Series Classification”. In: *Proc. of the 20th IEEE International Conference on Machine Learning and Applications, ICMLA, December 13-16, Pasadena, CA, USA*. IEEE, 2021, pp. 422–428 (cit. on p. 54).

[Gou+21] Jianping Gou, Baosheng Yu, Stephen J. Maybank, and Dacheng Tao. “Knowledge Distillation: A Survey”. In: *Int. J. Comput. Vis.* 129.6 (2021), pp. 1789–1819 (cit. on p. 53).

[GB04] Yves Grandvalet and Yoshua Bengio. “Semi-Supervised Learning by Entropy Minimization”. In: *Advances in Neural Information Processing Systems 17: Annual Conference on Neural Information Processing Systems, NIPS, December 13-18, Vancouver, BC, Canada*. MIT Press, 2004, pp. 529–536 (cit. on p. 47).

[GdA01] Yves Grandvalet, Florence d’Alché-Buc, and Christophe Ambroise. “Boosting Mixture Models for Semi-Supervised Learning”. In: *Proc. of the International Conference on Artificial Neural Networks, ICANN, August 21-25, Vienna, Austria*. Vol. 2130. Lecture Notes in Computer Science. Springer, 2001, pp. 41–48 (cit. on p. 49).

[Gri+20] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. “Bootstrap Your Own Latent - A New Approach to Self-Supervised Learning”. In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, virtual*. Curran Associates, Inc., 2020, pp. 21271–21284 (cit. on p. 48).

[GD18] Romain Guillaume and Didier Dubois. “A Maximum Likelihood Approach to Inference Under Coarse Data Based on Minimax Regret”. In: *Proc. of the 9th International Conference on Soft Methods in Probability and Statistics, SMPS, September 17-21, Compiègne, France*. Vol. 832. Advances in Intelligent Systems and Computing. Springer, 2018, pp. 99–106 (cit. on p. 37).

[GD15] Romain Guillaume and Didier Dubois. “Robust Parameter Estimation of Density Functions under Fuzzy Interval Observations”. In: *9th International Symposium on Imprecise Probability: Theories and Applications, ISIPTA, July 20-24, Pescara, Italy*. 2015, pp. 147–156 (cit. on p. 37).

[Guo+17] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. “On Calibration of Modern Neural Networks”. In: *Proc. of the 34th International Conference on Machine Learning, ICML, August 6-11, Sydney, NSW, Australia*. Vol. 70. Proc. of Machine Learning Research. PMLR, 2017, pp. 1321–1330 (cit. on pp. 2, 14).

[Gut+16] Pedro Antonio Gutiérrez, María Pérez-Ortiz, Javier Sánchez-Monedero, Francisco Fernández-Navarro, and César Hervás-Martínez. “Ordinal Regression Methods: Survey and Experimental Study”. In: *IEEE Trans. Knowl. Data Eng.* 28.1 (2016), pp. 127–146 (cit. on p. 11).

[GC18] Romain Guyard and Véronique Cherfaoui. “Study of Discounting Methods Applied to Canonical Decomposition of Belief Functions”. In: *Proc. of the 21st International Conference on Information Fusion, FUSION, July 10-13, Cambridge, United Kingdom*. IEEE, 2018, pp. 2505–2512 (cit. on p. 43).

[Han+21] Jonas Hanselle, Alexander Tornede, Marcel Wever, and Eyke Hüllermeier. "Algorithm Selection as Superset Learning: Constructing Algorithm Selectors from Imprecise Performance Data". In: *Advances in Knowledge Discovery and Data Mining - 25th Pacific-Asia Conference, PAKDD, May 11-14, virtual, Proc., Part I*. Vol. 12712. Lecture Notes in Computer Science. Springer, 2021, pp. 152–163 (cit. on p. 36).

[Han+20] Jonas Hanselle, Alexander Tornede, Marcel Wever, and Eyke Hüllermeier. "Hybrid Ranking and Regression for Algorithm Selection". In: *Proc. of the 43rd German Conference on AI, KI: Advances in Artificial Intelligence, September 21-25, Bamberg, Germany*. Vol. 12325. Lecture Notes in Computer Science. Springer, 2020, pp. 59–72 (cit. on p. 63).

[Has+20] Noriaki Hashimoto, Daisuke Fukushima, Ryoichi Koga, Yusuke Takagi, Kaho Ko, Kei Kohno, Masato Nakaguro, Shigeo Nakamura, Hidekata Hontani, and Ichiro Takeuchi. "Multi-Scale Domain-Adversarial Multiple-Instance CNN for Cancer Subtype Classification with Unannotated Histopathological Images". In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 13-19, Seattle, WA, USA*. Computer Vision Foundation / IEEE, 2020, pp. 3851–3860 (cit. on p. 25).

[HTF09] Trevor Hastie, Robert Tibshirani, and Jerome H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd ed. Springer Series in Statistics. Springer, 2009 (cit. on p. 12).

[He+18] Fengxiang He, Tongliang Liu, Geoffrey I. Webb, and Dacheng Tao. "Instance-Dependent PU Learning by Bayesian Optimal Relabeling". In: *CoRR* abs/1808.-02180 (2018) (cit. on p. 23).

[He+20] Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. "Revisiting Self-Training for Neural Sequence Generation". In: *8th International Conference on Learning Representations, ICLR, April 26 - May 1, Addis Ababa, Ethiopia*. OpenReview.net, 2020 (cit. on p. 49).

[He+16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep Residual Learning for Image Recognition". In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 27-30, Las Vegas, NV, USA*. IEEE Computer Society, 2016, pp. 770–778 (cit. on p. 12).

[He+22] Shuo He, Lei Feng, Fengmao Lv, Wen Li, and Guowu Yang. "Partial Label Learning with Semantic Label Representations". In: *Proc. of the 28th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 14-18, Washington DC, DC, USA*. ACM, 2022, pp. 545–553 (cit. on p. 34).

[Hec79] James J. Heckman. "Sample Selection Bias as a Specification Error". In: *Econometrica* 47.1 (1979), pp. 153–161 (cit. on p. 18).

[Her+98] Ralf Herbrich, Thore Graepel, Peter Bollmann-Sdorra, and Klaus Obermayer. "Supervised Learning of Preference Relations". In: *Proc. des Fachgruppentreffens Maschinelles Lernen, FGML* (1998), pp. 43–47 (cit. on p. 61).

[Her+21] Marek Herde, Denis Huseljic, Bernhard Sick, and Adrian Calma. “A Survey on Cost Types, Interaction Schemes, and Annotator Performance Models in Selection Algorithms for Active Learning in Classification”. In: *IEEE Access* 9 (2021), pp. 166970–166989 (cit. on p. 62).

[HOT06] Geoffrey E. Hinton, Simon Osindero, and Yee Whye Teh. “A Fast Learning Algorithm for Deep Belief Nets”. In: *Neural Comput.* 18.7 (2006), pp. 1527–1554 (cit. on p. 48).

[HVD15] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. “Distilling the Knowledge in a Neural Network”. In: *CoRR* abs/1503.02531 (2015) (cit. on p. 53).

[HSS08] Thomas Hofmann, Bernhard Schölkopf, and Alexander J. Smola. “Kernel Methods in Machine Learning”. In: *Ann. Stat.* 36.3 (2008), pp. 1171–1220 (cit. on p. 12).

[Hua+06] Jiayuan Huang, Alexander J. Smola, Arthur Gretton, Karsten M. Borgwardt, and Bernhard Schölkopf. “Correcting Sample Selection Bias by Unlabeled Data”. In: *Advances in Neural Information Processing Systems 19: Annual Conference on Neural Information Processing Systems, NIPS, December 4-7, Vancouver, BC, Canada*. MIT Press, 2006, pp. 601–608 (cit. on p. 18).

[Hua+22] Ling Huang, Thierry Denoeux, Pierre Vera, and Su Ruan. “Evidence Fusion with Contextual Discounting for Multi-Modality Medical Image Segmentation”. In: *Proc. of the 25th International Conference on Medical Image Computing and Computer Assisted Intervention, MICCAI, September 18-22, Singapore, Part V*. Vol. 13435. Lecture Notes in Computer Science. Springer, 2022, pp. 401–411 (cit. on p. 43).

[Hua+21] Ling Huang, Su Ruan, Pierre Decazes, and Thierry Denoeux. “Evidential Segmentation of 3D PET/CT Images”. In: *Proc. of the 6th International Conference on Belief Functions: Theory and Applications, BELIEF, October 15-19, Shanghai, China*. Vol. 12915. Lecture Notes in Computer Science. Springer, 2021, pp. 159–167 (cit. on p. 43).

[HRD23] Ling Huang, Su Ruan, and Thierry Denoeux. “Application of Belief Functions to Medical Image Segmentation: A Review”. In: *Inf. Fusion* 91 (2023), pp. 737–756 (cit. on p. 43).

[HZW13] Wei Huang, Peng Zhang, and Min Wan. “A Novel Similarity Learning Method via Relative Comparison for Content-Based Medical Image Retrieval”. In: *J. Digit. Imaging* 26.5 (2013), pp. 850–865 (cit. on p. 62).

[Hub81] Peter J. Huber. *Robust Statistics*. Wiley Series in Probability and Statistics. Wiley, 1981 (cit. on p. 17).

[HB21] Like Hui and Mikhail Belkin. “Evaluation of Neural Architectures trained with square Loss vs Cross-Entropy in Classification Tasks”. In: *9th International Conference on Learning Representations, ICLR, May 3-7, virtual*. OpenReview.net, 2021 (cit. on p. 13).

[Hül14] Eyke Hüllermeier. “Learning from Imprecise and Fuzzy Observations: Data Disambiguation through Generalized Loss Minimization”. In: *Int. J. Approx. Reason.* 55.7 (2014), pp. 1519–1534 (cit. on pp. 34, 35, 38–40, 173).

[HB06] Eyke Hüllermeier and Jürgen Beringer. “Learning from Ambiguously Labeled Examples”. In: *Intell. Data Anal.* 10.5 (2006), pp. 419–439 (cit. on pp. 30, 33).

[HC15] Eyke Hüllermeier and Weiwei Cheng. “Superset Learning Based on Generalized Loss Minimization”. In: *Proc. of the 26th European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, September 7-11, Porto, Portugal, Part II*. Vol. 9285. LNCS. Springer, 2015, pp. 260–275 (cit. on pp. 23, 31, 36, 175).

[HDC19] Eyke Hüllermeier, Sébastien Destercke, and Inés Couso. “Learning from Imprecise Data: Adjustments of Optimistic and Pessimistic Variants”. In: *Proc. of the 13th International Conference on Scalable Uncertainty Management, SUM, December 16-18, Compiègne, France*. Vol. 11940. Lecture Notes in Computer Science. Springer, 2019, pp. 266–279 (cit. on p. 37).

[HDS22] Eyke Hüllermeier, Sébastien Destercke, and Mohammad Hossein Shaker. “Quantification of Credal Uncertainty in Machine Learning: A Critical Analysis and Empirical Comparison”. In: *Proc. of the 38th Conference on Uncertainty in Artificial Intelligence, UAI, August 1-5, Eindhoven, The Netherlands*. Vol. 180. Proc. of Machine Learning Research. PMLR, 2022, pp. 548–557 (cit. on p. 174).

[HW21] Eyke Hüllermeier and Willem Waegeman. “Aleatoric and Epistemic Uncertainty in Machine Learning: An Introduction to Concepts and Methods”. In: *Mach. Learn.* 110.3 (2021), pp. 457–506 (cit. on pp. 13, 38, 174).

[ITW18] Maximilian Ilse, Jakub M. Tomczak, and Max Welling. “Attention-Based Deep Multiple Instance Learning”. In: *Proc. of the 35th International Conference on Machine Learning, ICML, July 10-15, Stockholmsmässan, Stockholm, Sweden*. Vol. 80. Proc. of Machine Learning Research. PMLR, 2018, pp. 2132–2141 (cit. on p. 25).

[IKK19] Go Irie, Takahito Kawanishi, and Kunio Kashino. “Robust Learning for Deep Monocular Depth Estimation”. In: *Proc. of the IEEE International Conference on Image Processing, ICIP, September 22-25, Taipei, Taiwan*. IEEE, 2019, pp. 964–968 (cit. on p. 17).

[Ish+17] Takashi Ishida, Gang Niu, Weihua Hu, and Masashi Sugiyama. “Learning from Complementary Labels”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, December 4-9, Long Beach, CA, USA*. Curran Associates, Inc., 2017, pp. 5639–5649 (cit. on p. 23).

[INS18] Takashi Ishida, Gang Niu, and Masashi Sugiyama. “Binary Classification from Positive-Confidence Data”. In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, NeurIPS, December 3-8, Montréal, Canada*. Curran Associates, Inc., 2018, pp. 5921–5932 (cit. on p. 13).

[IIS22] Hiroki Ishiguro, Takashi Ishida, and Masashi Sugiyama. “Learning from Noisy Complementary Labels with Robust Loss Functions”. In: *IEICE Trans. Inf. Syst.* 105-D.2 (2022), pp. 364–376 (cit. on p. 23).

[Izm+18] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry P. Vetrov, and Andrew Gordon Wilson. “Averaging Weights Leads to Wider Optima and Better Generalization”. In: *Proc. of the 34th Conference on Uncertainty in Artificial Intelligence, UAI, August 6-10, Monterey, CA, USA*. AUAI Press, 2018, pp. 876–885 (cit. on p. 54).

[JAE08] Ilyes Jenhani, Nahla Ben Amor, and Zied Elouedi. “Decision Trees as Possibilistic Classifiers”. In: *Int. J. Approx. Reason.* 48.3 (2008), pp. 784–807 (cit. on p. 42).

[Jia+18] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. “MentorNet: Learning Data-Driven Curriculum for Very Deep Neural Networks on Corrupted Labels”. In: *Proc. of the 35th International Conference on Machine Learning, ICML, July 10-15, Stockholmsmässan, Stockholm, Sweden*. Vol. 80. Proc. of Machine Learning Research. PMLR, 2018, pp. 2309–2318 (cit. on p. 14).

[Jie+17] Zequn Jie, Yunchao Wei, Xiaojie Jin, Jiashi Feng, and Wei Liu. “Deep Self-Taught Learning for Weakly Supervised Object Localization”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, July 21-26, Honolulu, HI, USA*. IEEE Computer Society, 2017, pp. 4294–4302 (cit. on p. 28).

[JG02] Rong Jin and Zoubin Ghahramani. “Learning with Multiple Labels”. In: *Advances in Neural Information Processing Systems 15: Annual Conference on Neural Information Processing Systems, NIPS, December 9-14, Vancouver, BC, Canada*. MIT Press, 2002, pp. 897–904 (cit. on pp. 23, 33, 34).

[Joa02] Thorsten Joachims. “Optimizing Search Engines using Clickthrough Data”. In: *Proc. of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 23-26, Edmonton, AB, Canada*. ACM, 2002, pp. 133–142 (cit. on p. 61).

[Joy+20] Tom Joy, Sebastian M. Schmon, Philip H. S. Torr, N. Siddharth, and Tom Rainforth. “Rethinking Semi-Supervised Learning in VAEs”. In: *CoRR* abs/2006.-10102 (2020) (cit. on p. 57).

[KKA10] Toshihiro Kamishima, Hideto Kazawa, and Shotaro Akaho. “A Survey and Empirical Comparison of Object Ranking Methods”. In: *Preference Learning*. Springer, 2010, pp. 181–201 (cit. on p. 61).

[Kar+21] Giannis Karamanolakis, Subhabrata Mukherjee, Guoqing Zheng, and Ahmed Hassan Awadallah. “Self-Training with Weak Supervision”. In: *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, June 6-11, virtual*. Association for Computational Linguistics, 2021, pp. 845–863 (cit. on p. 30).

[KK04] Takeaki Kariya and Hiroshi Kurata. *Generalized Least Squares*. John Wiley & Sons, 2004 (cit. on pp. 2, 17).

[KBH23] Timo Kaufmann, Viktor Bengs, and Eyke Hüllermeier. *Reinforcement Learning from Human Feedback for Cyber-Physical Systems: On the Potential of Self-Supervised Pretraining*. Machine Learning for Cyber-Physical Systems, ML4CPS, March 29-31, Hamburg, Germany. 2023 (cit. on p. 62).

[KG17] Alex Kendall and Yarin Gal. “What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?” In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, December 4-9, Long Beach, CA, USA*. Curran Associates, Inc., 2017, pp. 5574–5584 (cit. on p. 174).

[Ken48] Maurice G. Kendall. *Rank Correlation Methods*. C. Griffin, 1948 (cit. on p. 62).

[Kim+21] Byoungjip Kim, Jinho Choo, Yeong-Dae Kwon, Seongho Joe, Seungjai Min, and Youngjune Gwon. “SelfMatch: Combining Contrastive Self-Supervision and Consistency for Semi-Supervised Learning”. In: *CoRR* abs/2101.06480 (2021) (cit. on p. 54).

[Kim+22] Hoyoung Kim, Seunghyuk Cho, Dongwoo Kim, and Jungseul Ok. “Robust Deep Learning from Crowds with Belief Propagation”. In: *Proc. of the 25th International Conference on Artificial Intelligence and Statistics, AISTATS, March 28-30, virtual*. Vol. 151. Proc. of Machine Learning Research. PMLR, 2022, pp. 2803–2822 (cit. on p. 27).

[Kin+14] Diederik P. Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. “Semi-Supervised Learning with Deep Generative Models”. In: *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems, NIPS, December 8-13, Montreal, QC, Canada*. Curran Associates, Inc., 2014, pp. 3581–3589 (cit. on p. 57).

[KW14] Diederik P. Kingma and Max Welling. “Auto-Encoding Variational Bayes”. In: *2nd International Conference on Learning Representations, ICLR, April 14-16, Banff, AB, Canada*. 2014 (cit. on p. 57).

[Kir+23] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloé Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross B. Girshick. “Segment Anything”. In: *CoRR* abs/2304.02643 (2023) (cit. on p. 48).

[Kme71] Jan Kmenta. *Elements of Econometrics*. Macmillan series in economics. Macmillan, 1971 (cit. on p. 17).

[KZB19] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. “Revisiting Self-Supervised Visual Representation Learning”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 16-20, Long Beach, CA, USA*. Computer Vision Foundation / IEEE, 2019, pp. 1920–1929 (cit. on p. 48).

[Kor+21] Simon Kornblith, Ting Chen, Honglak Lee, and Mohammad Norouzi. “Why Do Better Loss Functions Lead to Less Transferable Features?” In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-14, virtual*. Curran Associates, Inc., 2021, pp. 28648–28662 (cit. on pp. 13, 14).

[Kot+21] Thanasis Kotsiopoulos, Panagiotis G. Sarigiannidis, Dimosthenis Ioannidis, and Dimitrios Tzovaras. “Machine Learning and Deep Learning in Smart Manufacturing: The Smart Grid Paradigm”. In: *Comput. Sci. Rev.* 40 (2021), p. 100341 (cit. on p. 1).

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems 25: Annual Conference on Neural Information Processing Systems, NIPS, December 3-6, Lake Tahoe, NV, USA*. Curran Associates, Inc., 2012, pp. 1106–1114 (cit. on p. 12).

[KA20] Ujwal Krothapalli and A. Lynn Abbott. “Adaptive Label Smoothing”. In: *CoRR* abs/2009.06432 (2020) (cit. on p. 15).

[Küg+20] Julius von Kügelgen, Alexander Mey, Marco Loog, and Bernhard Schölkopf. “Semi-Supervised Learning, Causality, and the Conditional Cluster Assumption”. In: *Proc. of the 36th Conference on Uncertainty in Artificial Intelligence, UAI, August 3-6, virtual*. Vol. 124. Proc. of Machine Learning Research. AUAI Press, 2020, pp. 1–10 (cit. on p. 45).

[KMF17] Meelis Kull, Telmo de Menezes e Silva Filho, and Peter A. Flach. “Beta Calibration: A Well-Founded and Easily Implemented Improvement on Logistic Calibration for Binary Classifiers”. In: *Proc. of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS, April 20-22, Fort Lauderdale, FL, USA*. Vol. 54. Proc. of Machine Learning Research. PMLR, 2017, pp. 623–631 (cit. on p. 14).

[KSF17] Abhishek Kumar, Prasanna Sattigeri, and Tom Fletcher. “Semi-Supervised Learning with GANs: Manifold Invariance with Improved Inference”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, December 4-9, Long Beach, CA, USA*. Curran Associates, Inc., 2017, pp. 5534–5544 (cit. on p. 56).

[Kuo+20] Chia-Wen Kuo, Chih-Yao Ma, Jia-Bin Huang, and Zsolt Kira. “FeatMatch: Feature-Based Augmentation for Semi-Supervised Learning”. In: *Proc. of the 16th European Conference on Computer Vision, ECCV, August 23-28, Glasgow, United Kingdom, Part XVIII*. Vol. 12363. Lecture Notes in Computer Science. Springer, 2020, pp. 479–495 (cit. on p. 54).

[Kut+21] Svetlana Kutuzova, Oswin Krause, Douglas McCloskey, Mads Nielsen, and Christian Igel. "Multimodal Variational Autoencoders for Semi-Supervised Learning: In Defense of Product-of-Experts". In: *CoRR* abs/2101.07240 (2021) (cit. on p. 57).

[LW07] John D. Lafferty and Larry A. Wasserman. "Statistical Analysis of Semi-Supervised Regression". In: *Advances in Neural Information Processing Systems 20: Annual Conference on Neural Information Processing Systems, NIPS, December 3-6, Vancouver, BC, Canada*. Curran Associates, Inc., 2007, pp. 801–808 (cit. on pp. 45, 47).

[Lai+14] Kuan-Ting Lai, Felix X. Yu, Ming-Syan Chen, and Shih-Fu Chang. "Video Event Detection by Inferring Temporal Instance Labels". In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 23-28, Columbus, OH, USA*. IEEE Computer Society, 2014, pp. 2251–2258 (cit. on p. 26).

[LA17] Samuli Laine and Timo Aila. "Temporal Ensembling for Semi-Supervised Learning". In: *5th International Conference on Learning Representations, ICLR, April 24-26, Toulon, France*. OpenReview.net, 2017 (cit. on pp. 46, 54).

[Lan+20] Ouyu Lan, Xiao Huang, Bill Yuchen Lin, He Jiang, Liyuan Liu, and Xiang Ren. "Learning to Contextually Aggregate Multi-Source Supervision for Sequence Labeling". In: *Proc. of the 58th Annual Meeting of the Association for Computational Linguistics, ACL, July 5-10, virtual*. Association for Computational Linguistics, 2020, pp. 2134–2146 (cit. on pp. 29, 30).

[LJ04] Neil D. Lawrence and Michael I. Jordan. "Semi-Supervised Learning via Gaussian Processes". In: *Advances in Neural Information Processing Systems 17: Annual Conference on Neural Information Processing Systems, NIPS, December 13-18, Vancouver, BC, Canada*. MIT Press, 2004, pp. 753–760 (cit. on p. 52).

[Lec+18] Bruno Lecouat, Chuan Sheng Foo, Houssam Zenati, and Vijay Ramaseshan Chandrasekhar. "Semi-Supervised Learning With GANs: Revisiting Manifold Regularization". In: *6th International Conference on Learning Representations, ICLR, April 30 - May 3, Vancouver, BC, Canada, Workshop Track Proc.* OpenReview.net, 2018 (cit. on p. 55).

[Lee13] Dong-Hyun Lee. "Pseudo-Label: The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks". In: *Workshop on Challenges in Representation Learning, International Conference on Machine Learning, ICML, June 16-21, Atlanta, GA, USA*. Vol. 3. 2013 (cit. on pp. 47–49).

[LCZ22] Dongkyu Lee, Ka Chun Cheung, and Nevin L. Zhang. "Adaptive Label Smoothing with Self-Knowledge in Natural Language Generation". In: *Proc. of the Conference on Empirical Methods in Natural Language Processing, EMNLP, December 7-11, Abu Dhabi, United Arab Emirates*. Association for Computational Linguistics, 2022, pp. 9781–9792 (cit. on p. 15).

[Leg05] Adrien-Marie Legendre. *Nouvelles Méthodes pour la Détermination des Orbites des Comètes*. Nineteenth Century Collections Online (NCCO): Science, Technology, and Medicine: 1780-1925. F. Didot, 1805 (cit. on p. 16).

[Lev83] Isaac Levi. *The Enterprise of Knowledge: An Essay on Knowledge, Credal Probability, and Chance*. MIT Press, 1983 (cit. on p. 40).

[Li+17] Chongxuan Li, Taufik Xu, Jun Zhu, and Bo Zhang. "Triple Generative Adversarial Nets". In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, December 4-9, Long Beach, CA, USA*. Curran Associates, Inc., 2017, pp. 4088–4098 (cit. on p. 57).

[LZZ18] Chongxuan Li, Jun Zhu, and Bo Zhang. "Max-Margin Deep Generative Models for (Semi-)Supervised Learning". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 40.11 (2018), pp. 2762–2775 (cit. on p. 52).

[Li+22a] Chunyuan Li, Jianwei Yang, Pengchuan Zhang, Mei Gao, Bin Xiao, Xiyang Dai, Lu Yuan, and Jianfeng Gao. "Efficient Self-Supervised Vision Transformers for Representation Learning". In: *10th International Conference on Learning Representations, ICLR, April 25-29, virtual*. OpenReview.net, 2022 (cit. on p. 48).

[Li+22b] Hengduo Li, Zuxuan Wu, Abhinav Shrivastava, and Larry S. Davis. "Rethinking Pseudo Labels for Semi-Supervised Object Detection". In: *Proc. of the 36th AAAI Conference on Artificial Intelligence, AAAI, the 34th Conference on Innovative Applications of Artificial Intelligence, IAAI, the 12th Symposium on Educational Advances in Artificial Intelligence, EAAI, February 22 - March 1, virtual*. AAAI Press, 2022, pp. 1314–1322 (cit. on p. 49).

[Li+21a] Jiacheng Li, Haibo Ding, Jingbo Shang, Julian J. McAuley, and Zhe Feng. "Weakly Supervised Named Entity Tagging with Learnable Logical Rules". In: *Proc. of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP, Volume 1: Long Papers, August 1-6, virtual*. Association for Computational Linguistics, 2021, pp. 4568–4581 (cit. on p. 29).

[LXH21] Junnan Li, Caiming Xiong, and Steven C. H. Hoi. "CoMatch: Semi-Supervised Learning with Contrastive Graph Regularization". In: *Proc. of the IEEE/CVF International Conference on Computer Vision, ICCV, October 10-17, Montreal, QC, Canada*. IEEE, 2021, pp. 9455–9464 (cit. on p. 54).

[LDB20] Weizhi Li, Gautam Dasarathy, and Visar Berisha. "Regularization via Structural Label Smoothing". In: *Proc. of the 23rd International Conference on Artificial Intelligence and Statistics, AISTATS, August 26-28, virtual*. Vol. 108. Proc. of Machine Learning Research. PMLR, 2020, pp. 1453–1463 (cit. on p. 15).

[Li+19] Yang Li, Quan Pan, Suhang Wang, Haiyun Peng, Tao Yang, and Erik Cambria. "Disentangled Variational Auto-Encoder for Semi-Supervised learning". In: *Inf. Sci.* 482 (2019), pp. 73–85 (cit. on p. 57).

[Li+21b] Yinghao Li, Pranav Shetty, Lucas Liu, Chao Zhang, and Le Song. “BERTifying the Hidden Markov Model for Multi-Source Weakly Supervised Named Entity Recognition”. In: *Proc. of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP, Volume 1: Long Papers, August 1-6, virtual*. Association for Computational Linguistics, 2021, pp. 6178–6190 (cit. on p. 30).

[Li+14] Zhan Li, Guohua Geng, Jun Feng, Jinye Peng, Chao Wen, and Junli Liang. “Multiple Instance Learning based on Positive Instance Selection and Bag Structure Construction”. In: *Pattern Recognit. Lett.* 40 (2014), pp. 19–26 (cit. on p. 25).

[LS18] Zhengqi Li and Noah Snavely. “MegaDepth: Learning Single-View Depth Prediction From Internet Photos”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 18-22, Salt Lake City, UT, USA*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 2041–2050 (cit. on p. 63).

[LDH22] Julian Lienen, Caglar Demir, and Eyke Hüllermeier. “Conformal Credal Self-Supervised Learning”. In: *CoRR* abs/2205.15239 (2022). Accepted at the 12th Symposium on Conformal and Probabilistic Prediction with Applications, COPA 2023, September 13-15, Limassol, Cyprus (cit. on p. 5).

[LH21a] Julian Lienen and Eyke Hüllermeier. “Credal Self-Supervised Learning”. In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-14, virtual*. Curran Associates, Inc., 2021, pp. 14370–14382 (cit. on p. 5).

[LH21b] Julian Lienen and Eyke Hüllermeier. “From Label Smoothing to Label Relaxation”. In: *Proc. of the 35th AAAI Conference on Artificial Intelligence, AAAI, the 33rd Conference on Innovative Applications of Artificial Intelligence, IAAI, the 11th Symposium on Educational Advances in Artificial Intelligence, EAAI, February 2-9, virtual*. AAAI Press, 2021, pp. 8583–8591 (cit. on p. 4).

[LH21c] Julian Lienen and Eyke Hüllermeier. “Instance Weighting through Data Imprecisiation”. In: *Int. J. Approx. Reason.* 134 (2021), pp. 1–14 (cit. on p. 5).

[LH23] Julian Lienen and Eyke Hüllermeier. “Mitigating Label Noise through Data Ambiguation”. In: *CoRR* abs/2305.13764 (2023) (cit. on p. 5).

[Lie+21a] Julian Lienen, Eyke Hüllermeier, Ralph Ewerth, and Nils Nommensen. “Monocular Depth Estimation via Listwise Ranking Using the Plackett-Luce Model”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 19-25, virtual*. Computer Vision Foundation / IEEE, 2021, pp. 14595–14604 (cit. on p. 8).

[Lie+21b] Julian Lienen, Nils Nommensen, Ralph Ewerth, and Eyke Hüllermeier. “Robust Regression for Monocular Depth Estimation”. In: *Proc. of the 13th Asian Conference on Machine Learning, ACML, November 17-19, virtual*. Vol. 157. Proc. of Machine Learning Research. PMLR, 2021, pp. 1001–1016 (cit. on p. 5).

[Lin+17] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. “Focal Loss for Dense Object Detection”. In: *Proc. of the IEEE International Conference on Computer Vision, ICCV, October 22-29, Venice, Italy*. IEEE Computer Society, 2017, pp. 2999–3007 (cit. on p. 14).

[Lin+22] Xudong Lin, Fabio Petroni, Gedas Bertasius, Marcus Rohrbach, Shih-Fu Chang, and Lorenzo Torresani. “Learning To Recognize Procedural Activities with Distant Supervision”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 18-24, New Orleans, LA, USA*. IEEE, 2022, pp. 13843–13853 (cit. on p. 28).

[Lis+20] Pierre Lison, Jeremy Barnes, Aliaksandr Hubin, and Samia Touileb. “Named Entity Recognition without Labelled Data: A Weak Supervision Approach”. In: *Proc. of the 58th Annual Meeting of the Association for Computational Linguistics, ACL, July 5-10, virtual*. Association for Computational Linguistics, 2020, pp. 1518–1533 (cit. on p. 30).

[Liu+22a] Alexander H. Liu, Wei-Ning Hsu, Michael Auli, and Alexei Baevski. “Towards End-to-End Unsupervised Speech Recognition”. In: *IEEE Spoken Language Technology Workshop, SLT, January 9-12, Doha, Qatar*. IEEE, 2022, pp. 221–228 (cit. on p. 48).

[Liu+19a] Ao Liu, Zhibing Zhao, Chao Liao, Pinyan Lu, and Lirong Xia. “Learning Plackett-Luce Mixtures from Partial Preferences”. In: *Proc. of the 33rd AAAI Conference on Artificial Intelligence, AAAI, the 31st Innovative Applications of Artificial Intelligence Conference, IAAI, the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI, January 27 - February 1, Honolulu, HI, USA*. AAAI Press, 2019, pp. 4328–4335 (cit. on p. 61).

[Liu+03] Bing Liu, Yang Dai, Xiaoli Li, Wee Sun Lee, and Philip S. Yu. “Building Text Classifiers Using Positive and Unlabeled Examples”. In: *Proc. of the 3rd IEEE International Conference on Data Mining, ICDM, December 19-22, Melbourne, FL, USA*. IEEE Computer Society, 2003, pp. 179–188 (cit. on p. 23).

[Liu+19b] Jiabin Liu, Bo Wang, Zhiquan Qi, Yingjie Tian, and Yong Shi. “Learning from Label Proportions with Generative Adversarial Networks”. In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS, December 8-14, Vancouver, BC, Canada*. Curran Associates, Inc., 2019, pp. 7167–7177 (cit. on p. 26).

[LD12] Li-Ping Liu and Thomas G. Dietterich. “A Conditional Multinomial Mixture Model for Superset Label Learning”. In: *Advances in Neural Information Processing Systems 25: Annual Conference on Neural Information Processing Systems, NIPS, December 3-6, Lake Tahoe, NV, USA*. Curran Associates, Inc., 2012, pp. 557–565 (cit. on pp. 30, 34).

[LD14] Li-Ping Liu and Thomas G. Dietterich. “Learnability of the Superset Label Learning Problem”. In: *Proc. of the 31th International Conference on Machine Learning, ICML, June 21-26, Beijing, China*. Vol. 32. JMLR Workshop and Conference Proc. JMLR.org, 2014, pp. 1629–1637 (cit. on pp. 22, 31–33).

[Liu+20a] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. “Early-Learning Regularization Prevents Memorization of Noisy Labels”. In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, virtual*. Curran Associates, Inc., 2020, pp. 20331–20342 (cit. on p. 14).

[Liu+22b] Sheng Liu, Zhihui Zhu, Qing Qu, and Chong You. “Robust Training under Label Noise by Over-Parameterization”. In: *Proc. of the 39th International Conference on Machine Learning, ICML, July 17-23, Baltimore, MD, USA*. Vol. 162. Proc. of Machine Learning Research. PMLR, 2022, pp. 14153–14172 (cit. on p. 14).

[Liu11] Tie-Yan Liu. *Learning to Rank for Information Retrieval*. Springer, 2011 (cit. on pp. 59, 60).

[Liu+20b] Xiaofeng Liu, Fangfang Fan, Lingsheng Kong, Zhihui Diao, Wanqing Xie, Jun Lu, and Jane You. “Unimodal Regularized Neuron Stick-Breaking for Ordinal Classification”. In: *Neurocomputing* 388 (2020), pp. 34–44 (cit. on p. 173).

[Liu+20c] Yi Liu, Guangchang Deng, Xiangping Zeng, Si Wu, Zhiwen Yu, and Hau-San Wong. “Regularizing Discriminative Capability of CGANs for Semi-Supervised Generative Learning”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 13-19, Seattle, WA, USA*. Computer Vision Foundation / IEEE, 2020, pp. 5719–5728 (cit. on p. 57).

[Liu+16] Zhen Liu, Jun-an Yang, Hui Liu, and Wei Wang. “Transfer Learning by Sample Selection Bias Correction and Its Application in Communication Specific Emitter Identification”. In: *J. Commun.* 11.4 (2016), pp. 417–427 (cit. on p. 29).

[LH15] Shenzhen Lu and Eyke Hüllermeier. “Locally Weighted Regression through Data Imprecisiation”. In: *Proc. of the 25th Workshop on Computational Intelligence, November 26-27, Dortmund, Germany*. Vol. 54. Karlsruhe: KIT Scientific Publishing, 2015, pp. 97–103 (cit. on pp. 19, 36, 125).

[LH16] Shenzhen Lu and Eyke Hüllermeier. “Support Vector Classification on Noisy Data using Fuzzy Superset Losses”. In: *Proc. of the 26th Workshop on Computational Intelligence, November 24-25, Dortmund, Germany*. KIT Scientific Publishing, 2016, pp. 1–8 (cit. on pp. 19, 36, 51, 125).

[Luc59] Robert Duncan Luce. *Individual Choice Behavior: A Theoretical analysis*. New York, NY, USA: Wiley, 1959 (cit. on p. 61).

[Luk+20] Michal Lukasik, Srinadh Bhojanapalli, Aditya Krishna Menon, and Sanjiv Kumar. “Does Label Smoothing Mitigate Label Noise?” In: *Proc. of the 37th International Conference on Machine Learning, ICML, July 13-18, virtual*. Vol. 119. Proc. of Machine Learning Research. PMLR, 2020, pp. 6448–6458 (cit. on p. 14).

[Luk+19] Adewale Lukman, Kayode Ayinde, Sek Kun, and Emmanuel Adewuyi. “A Modified New Two-Parameter Estimator in a Linear Regression Model”. In: *Modelling and Simulation in Engineering* 2019 (May 2019), pp. 1–10 (cit. on p. 17).

[Luo+18a] Yucen Luo, Jun Zhu, Mengxi Li, Yong Ren, and Bo Zhang. “Smooth Neighbors on Teacher Graphs for Semi-Supervised Learning”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 18-22, Salt Lake City, UT, USA*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 8896–8905 (cit. on p. 46).

[Luo+18b] Yucen Luo, Jun Zhu, Mengxi Li, Yong Ren, and Bo Zhang. “Smooth Neighbors on Teacher Graphs for Semi-Supervised Learning”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 18-22, Salt Lake City, UT, USA*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 8896–8905 (cit. on p. 55).

[Lv+21] Jiaqi Lv, Lei Feng, Miao Xu, Bo An, Gang Niu, Xin Geng, and Masashi Sugiyama. “On the Robustness of Average Losses for Partial-Label Learning”. In: *CoRR* abs/2106.06152 (2021) (cit. on pp. 31, 33, 34).

[Lv+20] Jiaqi Lv, Miao Xu, Lei Feng, Gang Niu, Xin Geng, and Masashi Sugiyama. “Progressive Identification of True Labels for Partial-Label Learning”. In: *Proc. of the 37th International Conference on Machine Learning, ICML, July 13-18, virtual*. Vol. 119. Proc. of Machine Learning Research. PMLR, 2020, pp. 6500–6510 (cit. on p. 34).

[Lv+22] Ying Lv, Bofeng Zhang, Guobing Zou, Xiaodong Yue, Zhikang Xu, and Haiyan Li. “Domain Adaptation with Data Uncertainty Measure Based on Evidence Theory”. In: *Entropy* 24.7 (2022), p. 966 (cit. on p. 43).

[Lyu+21] Gengyu Lyu, Songhe Feng, Tao Wang, Congyan Lang, and Yidong Li. “GM-PLL: Graph Matching Based Partial Label Learning”. In: *IEEE Trans. Knowl. Data Eng.* 33.2 (2021), pp. 521–535 (cit. on p. 34).

[Maa+16] Lars Maaløe, Casper Kaae Sønderby, Søren Kaae Sønderby, and Ole Winther. “Auxiliary Deep Generative Models”. In: *Proc. of the 33nd International Conference on Machine Learning, ICML, June 19-24, New York City, NY, USA*. Vol. 48. JMLR Workshop and Conference Proc. JMLR.org, 2016, pp. 1445–1453 (cit. on p. 57).

[Mah+21] Ayush Maheshwari, Oishik Chatterjee, KrishnaTeja Killamsetty, Ganesh Ramakrishnan, and Rishabh K. Iyer. “Semi-Supervised Data Programming with Subset Selection”. In: *Findings of the Association for Computational Linguistics: ACL/IJCNLP, August 1-6, virtual*. Vol. ACL/IJCNLP 2021. Findings of ACL. Association for Computational Linguistics, 2021, pp. 4640–4651 (cit. on p. 30).

[Mal+09] Pavan Kumar Mallapragada, Rong Jin, Anil K. Jain, and Yi Liu. “SemiBoost: Boosting for Semi-Supervised Learning”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 31.11 (2009), pp. 2000–2014 (cit. on p. 49).

[MH10] Maurizio Manuguerra and Gillian Z Heller. “Ordinal Regression Models for Continuous Scales”. In: *Int. J. Biostat.* 6.1 (2010) (cit. on p. 58).

[ML97] Oded Maron and Tomás Lozano-Pérez. “A Framework for Multiple-Instance Learning”. In: *Advances in Neural Information Processing Systems 10: Annual Conference on Neural Information Processing Systems, NIPS, December 1-6, Denver, CO, USA*. MIT Press, 1997, pp. 570–576 (cit. on p. 25).

[MG15] Lucas Maystre and Matthias Grossglauser. “Fast and Accurate Inference of Plackett-Luce Models”. In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems, NIPS, December 7-12, Montreal, QC, Canada*. Curran Associates, Inc., 2015, pp. 172–180 (cit. on p. 61).

[Mcf80] Daniel Mcfadden. “Econometric Models for Probabilistic Choice among Products”. In: *J. Bus.* 53 (1980), pp. 13–29 (cit. on p. 61).

[Meh+22] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. “A Survey on Bias and Fairness in Machine Learning”. In: *ACM Comput. Surv.* 54.6 (2022), 115:1–115:35 (cit. on pp. 2, 14, 18).

[Men+18] Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. “Weakly-Supervised Neural Text Classification”. In: *Proc. of the 27th ACM International Conference on Information and Knowledge Management, CIKM, October 22-26, Torino, Italy*. ACM, 2018, pp. 983–992 (cit. on p. 29).

[Mer+15] David Mercier, Frédéric Pichon, Éric Lefèvre, and François Delmotte. “Learning Contextual Discounting and Contextual Reinforcement from Labelled Data”. In: *Symbolic and Quantitative Approaches to Reasoning with Uncertainty*. Cham: Springer International Publishing, 2015, pp. 472–481 (cit. on p. 43).

[MQD05] David Mercier, Benjamin Quost, and Thierry Denoeux. “Contextual Discounting of Belief Functions”. In: *Proc. of the 8th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty, ECSQARU, July 6-8, Barcelona, Spain*. Vol. 3571. Lecture Notes in Computer Science. Springer, 2005, pp. 552–562 (cit. on p. 43).

[MQD08] David Mercier, Benjamin Quost, and Thierry Denoeux. “Refined Modeling of Sensor Reliability in the Belief Function Framework using Contextual Discounting”. In: *Inf. Fusion* 9.2 (2008), pp. 246–258 (cit. on p. 43).

[ML23] Alexander Mey and Marco Loog. “Improved Generalization in Semi-Supervised Learning: A Survey of Theoretical Results”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 45.4 (2023), pp. 4747–4767 (cit. on pp. 45, 47).

[MF21] Gabriel Michau and Olga Fink. “Unsupervised Transfer Learning for Anomaly Detection: Application to Complementary Operating Condition Transfer”. In: *Knowl. Based Syst.* 216 (2021), p. 106816 (cit. on p. 29).

[Mik+13] Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. “Distributed Representations of Words and Phrases and their Compositionality”. In: *Advances in Neural Information Processing Systems 26: Annual Conference on Neural Information Processing Systems, NIPS, December 5-8, Lake Tahoe, NV, USA*. Curran Associates, Inc., 2013, pp. 3111–3119 (cit. on p. 47).

[Min+09] Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. “Distant Supervision for Relation Extraction without Labeled Data”. In: *Proc. of the 47th Annual Meeting of the Association for Computational Linguistics, and the 4th International Joint Conference on Natural Language Processing, ACL/IJCNLP, August 2-7, Singapore*. The Association for Computer Linguistics, 2009, pp. 1003–1011 (cit. on p. 27).

[MD13] Enrique Miranda and Sébastien Destercke. “Extreme Points of the Credal Sets Generated by Elementary Comparative Probabilities”. In: *Proc. of the 12th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty, ECSQARU, July 8-10, Utrecht, The Netherlands*. Vol. 7958. Lecture Notes in Computer Science. Springer, 2013, pp. 424–435 (cit. on p. 175).

[MMD19] Enrique Miranda, Ignacio Montes, and Sébastien Destercke. “A Unifying Frame for Neighbourhood and Distortion Models”. In: *11th International Symposium on Imprecise Probability: Theories and Applications, ISIPTA, July 3-6, Ghent, Belgium*. Vol. 103. Proc. of Machine Learning Research. PMLR, 2019, pp. 304–313 (cit. on p. 173).

[Miy+19] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. “Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 41.8 (2019), pp. 1979–1993 (cit. on p. 54).

[MT17] Cristina Mollica and Luca Tardella. “Bayesian Plackett-Luce Mixture Models for Partially Ranked Data”. In: *Psychometrika* 82.2 (2017), pp. 442–458 (cit. on p. 61).

[Mor18] Serafín Moral. “Discounting Imprecise Probabilities”. In: *The Mathematics of the Uncertain: A Tribute to Pedro Gil*. Cham: Springer International Publishing, 2018, pp. 685–697 (cit. on p. 43).

[Mor+12] Jose G. Moreno-Torres, Troy Raeder, Rocío Alaíz-Rodríguez, Nitesh V. Chawla, and Francisco Herrera. “A Unifying View on Dataset Shift in Classification”. In: *Pattern Recognit.* 45.1 (2012), pp. 521–530 (cit. on p. 18).

[MKH19] Rafael Müller, Simon Kornblith, and Geoffrey E. Hinton. “When Does Label Smoothing Help?” In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS, December 8-14, Vancouver, BC, Canada*. Curran Associates, Inc., 2019, pp. 4696–4705 (cit. on p. 14).

[Mut+19] Siti Mutmainah, Samir Hachour, Frédéric Pichon, and David Mercier. “On Learning Evidential Contextual Corrections from Soft Labels Using a Measure of Discrepancy Between Contour Functions”. In: *Proc. of the 13th International Conference on Scalable Uncertainty Management, SUM, December 16-18, Compiègne, France*. Vol. 11940. Lecture Notes in Computer Science. Springer, 2019, pp. 382–389 (cit. on p. 42).

[Nar+17] Siddharth Narayanaswamy, Brooks Paige, Jan-Willem van de Meent, Alban Desmaison, Noah D. Goodman, Pushmeet Kohli, Frank D. Wood, and Philip H. S. Torr. “Learning Disentangled Representations with Semi-Supervised Deep Generative Models”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, December 4-9, Long Beach, CA, USA*. Curran Associates, Inc., 2017, pp. 5925–5935 (cit. on p. 57).

[NMY15] Takuya Narihira, Michael Maire, and Stella X. Yu. “Learning Lightness from Human Judgement on Relative Reflectance”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 7-12, Boston, MA, USA*. IEEE Computer Society, 2015, pp. 2965–2973 (cit. on p. 62).

[Nat+13] Nagarajan Natarajan, Inderjit S. Dhillon, Pradeep Ravikumar, and Ambuj Tewari. “Learning with Noisy Labels”. In: *Advances in Neural Information Processing Systems 26: Annual Conference on Neural Information Processing Systems, NIPS, December 5-8, Lake Tahoe, NV, USA*. Curran Associates, Inc., 2013, pp. 1196–1204 (cit. on p. 14).

[Neg+18] Sahand Negahban, Sewoong Oh, Kiran Koshy Thekumparampil, and Jiaming Xu. “Learning from Comparisons and Choices”. In: *J. Mach. Learn. Res.* 19 (2018), 40:1–40:95 (cit. on p. 62).

[ND20] Alejandro Newell and Jia Deng. “How Useful Is Self-Supervised Pretraining for Visual Tasks?” In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 13-19, Seattle, WA, USA*. Computer Vision Foundation / IEEE, 2020, pp. 7343–7352 (cit. on p. 48).

[NZ23] Duc Nguyen and Anderson Y. Zhang. “Efficient and Accurate Learning of Mixtures of Plackett-Luce Models”. In: *CoRR* abs/2302.05343 (2023) (cit. on p. 61).

[NC08] Nam Nguyen and Rich Caruana. “Classification with Partial Labels”. In: *Proc. of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 24-27, Las Vegas, NV, USA*. ACM, 2008, pp. 551–559 (cit. on p. 34).

[Ni+21] Peng Ni, Suyun Zhao, Zhi-Gang Dai, Hong Chen, and Cui-Ping Li. "Partial Label Learning via Conditional-Label-Aware Disambiguation". In: *J. Comput. Sci. Technol.* 36.3 (2021), pp. 590–605 (cit. on p. 34).

[NPL23] Hannele Niemi, Roy D Pea, and Yu Lu. *AI in Learning: Designing the Future*. Springer Nature, 2023 (cit. on p. 1).

[Niy13] Partha Niyogi. "Manifold Regularization and Semi-Supervised Learning: Some Theoretical Analyses". In: *J. Mach. Learn. Res.* 14.1 (2013), pp. 1229–1250 (cit. on pp. 47, 55).

[NJC21] Curtis G. Northcutt, Lu Jiang, and Isaac L. Chuang. "Confident Learning: Estimating Uncertainty in Dataset Labels". In: *J. Artif. Intell. Res.* 70 (2021), pp. 1373–1411 (cit. on p. 13).

[Ode16] Augustus Odena. "Semi-Supervised Learning with Generative Adversarial Networks". In: *CoRR* abs/1606.01583 (2016) (cit. on p. 56).

[Oos21] Harrie Oosterhuis. "Computationally Efficient Optimization of Plackett-Luce Ranking Models for Relevance and Fairness". In: *Proc. of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR, July 11-15, virtual*. ACM, 2021, pp. 1023–1032 (cit. on p. 61).

[Oos22] Harrie Oosterhuis. "Learning-to-Rank at the Speed of Sampling: Plackett-Luce Gradient Estimation with Minimal Computational Complexity". In: *Proc. of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR, July 11-15, Madrid, Spain*. ACM, 2022, pp. 2266–2271 (cit. on p. 61).

[Oqu+23] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael G. Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. "DINOv2: Learning Robust Visual Features without Supervision". In: *CoRR* abs/2304.07193 (2023) (cit. on p. 49).

[Ouy+22] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. "Training Language Models to Follow Instructions with Human Feedback". In: *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems, NeurIPS, November 28 - Friday 9, New Orleans, LA, USA*. Curran Associates, Inc., 2022, pp. 27730–27744 (cit. on p. 62).

[OG21a] Samet Oymak and Talha Cihad Gulcu. “A Theoretical Characterization of Semi-Supervised Learning with Self-Training for Gaussian Mixture Models”. In: *Proc. of the 24th International Conference on Artificial Intelligence and Statistics, AISTATS, April 13-15, virtual*. Vol. 130. Proc. of Machine Learning Research. PMLR, 2021, pp. 3601–3609 (cit. on p. 47).

[OG21b] Samet Oymak and Talha Cihad Gulcu. “A Theoretical Characterization of Semi-Supervised Learning with Self-Training for Gaussian Mixture Models”. In: *Proc. of the 24th International Conference on Artificial Intelligence and Statistics, AISTATS, April 13-15, virtual*. Vol. 130. Proc. of Machine Learning Research. PMLR, 2021, pp. 3601–3609 (cit. on p. 56).

[PP10] Alexander Pak and Patrick Paroubek. “Twitter as a Corpus for Sentiment Analysis and Opinion Mining”. In: *Proc. of the International Conference on Language Resources and Evaluation, LREC, May 17-23, Valletta, Malta*. European Language Resources Association, 2010 (cit. on p. 27).

[PY10] Sinno Jialin Pan and Qiang Yang. “A Survey on Transfer Learning”. In: *IEEE Trans. Knowl. Data Eng.* 22.10 (2010), pp. 1345–1359 (cit. on p. 28).

[Par+20] Daniel S. Park, Yu Zhang, Ye Jia, Wei Han, Chung-Cheng Chiu, Bo Li, Yonghui Wu, and Quoc V. Le. “Improved Noisy Student Training for Automatic Speech Recognition”. In: *21st Annual Conference of the International Speech Communication Association, Interspeech, October 25-29, virtual*. ISCA, 2020, pp. 2817–2821 (cit. on p. 49).

[PY21] Jerrod Parker and Shi Yu. “Named Entity Recognition through Deep Representation Learning and Weak Supervision”. In: *Findings of the Association for Computational Linguistics: ACL/IJCNLP, August 1-6, virtual*. Vol. ACL/IJCNLP 2021. Findings of ACL. Association for Computational Linguistics, 2021, pp. 3828–3839 (cit. on p. 30).

[Pat+14] Giorgio Patrini, Richard Nock, Tibério S. Caetano, and Paul Rivera. “(Almost) No Label No Cry”. In: *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems, NIPS, December 8-13, Montreal, QC, Canada*. Curran Associates, Inc., 2014, pp. 190–198 (cit. on p. 26).

[Per+17] Gabriel Pereyra, George Tucker, Jan Chorowski, Lukasz Kaiser, and Geoffrey E. Hinton. “Regularizing Neural Networks by Penalizing Confident Output Distributions”. In: *5th International Conference on Learning Representations, ICLR, April 24-26, Toulon, France, Workshop Track Proc.* OpenReview.net, 2017 (cit. on p. 14).

[Pha+21] Hieu Pham, Zihang Dai, Qizhe Xie, and Quoc V. Le. “Meta Pseudo Labels”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 19-25, virtual*. Computer Vision Foundation / IEEE, 2021, pp. 11557–11568 (cit. on p. 50).

[PRA13] Nikolaos Pitelis, Chris Russell, and Lourdes Agapito. “Learning a Manifold as an Atlas”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 23-28, Portland, OR, USA*. IEEE Computer Society, 2013, pp. 1642–1649 (cit. on p. 55).

[PRA14] Nikolaos Pitelis, Chris Russell, and Lourdes Agapito. “Semi-Supervised Learning Using an Unsupervised Atlas”. In: *Proc. of the 25th European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, September 15-19, Nancy, France, Part II*. Vol. 8725. Lecture Notes in Computer Science. Springer, 2014, pp. 565–580 (cit. on p. 55).

[Pla75] Robin Plackett. “The Analysis of Permutations”. In: *J. R. Stat. Soc. Ser. C Appl. Stat.* 24 (1975), pp. 193–202 (cit. on p. 61).

[Ple+17] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon M. Kleinberg, and Kilian Q. Weinberger. “On Fairness and Calibration”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, December 4-9, Long Beach, CA, USA*. Curran Associates, Inc., 2017, pp. 5680–5689 (cit. on p. 14).

[Qi+18] Guo-Jun Qi, Liheng Zhang, Hao Hu, Marzieh Edraki, Jingdong Wang, and Xian-Sheng Hua. “Global Versus Localized Generative Adversarial Nets”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 18-22, Salt Lake City, UT, USA*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 1517–1525 (cit. on p. 57).

[QTS12] Zhiquan Qi, Yingjie Tian, and Yong Shi. “Laplacian Twin Support Vector Machine for Semi-Supervised Classification”. In: *Neural Netw.* 35 (2012), pp. 46–53 (cit. on p. 55).

[Qua+09] Novi Quadrianto, Alexander J. Smola, Tibério S. Caetano, and Quoc V. Le. “Estimating Labels from Label Proportions”. In: *J. Mach. Learn. Res.* 10 (2009), pp. 2349–2374 (cit. on pp. 25, 26).

[Qua22] Erik Quaeghebeur. “Introduction to the Theory of Imprecise Probability”. In: *Uncertainty in Engineering: Introduction to Methods and Applications*. Cham: Springer International Publishing, 2022, pp. 37–50 (cit. on p. 40).

[QDL17] Benjamin Quost, Thierry Denoeux, and Shoumei Li. “Parametric Classification with Soft Labels using the Evidential EM Algorithm: Linear Discriminant Analysis versus Logistic Regression”. In: *Adv. Data Anal. Classif.* 11.4 (2017), pp. 659–690 (cit. on p. 42).

[Rad+18] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. “Improving Language Understanding by Generative Pre-Training”. In: (2018) (cit. on p. 48).

[Rah+21] Amir Masoud Rahmani, Efat Yousefpoor, Mohammad Sadegh Yousefpoor, Zahid Mehmood, Amir Haider, Mehdi Hosseinzadeh, and Rizwan Ali Naqvi. “Machine Learning (ML) in Medicine: Review, Applications, and Challenges”. In: *Mathematics* 9.22 (2021), p. 2970 (cit. on p. 1).

[Rai+07] Rajat Raina, Alexis J. Battle, Honglak Lee, Benjamin Packer, and Andrew Y. Ng. “Self-Taught Learning: Transfer Learning from Unlabeled Data”. In: *Proc. of the 24th International Conference on Machine Learning, ICML, June 20-24, Corvallis, OR, USA*. Vol. 227. ACM International Conference Proceeding Series. ACM, 2007, pp. 759–766 (cit. on p. 28).

[Ran+22] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. “Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-Shot Cross-Dataset Transfer”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 44.3 (2022), pp. 1623–1637 (cit. on p. 17).

[Ras+15] Antti Rasmus, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko. “Semi-Supervised Learning with Ladder Networks”. In: *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems, NIPS, December 7-12, Montreal, QC, Canada*. Curran Associates, Inc., 2015, pp. 3546–3554 (cit. on pp. 47, 54).

[RHF15] Mohammad Rastegari, Hannaneh Hajishirzi, and Ali Farhadi. “Discriminative and Consistent Similarities in Instance-Level Multiple Instance Learning”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 7-12, Boston, MA, USA*. IEEE Computer Society, 2015, pp. 740–748 (cit. on p. 25).

[Rat+17] Alexander Ratner, Stephen H. Bach, Henry R. Ehrenberg, Jason Alan Fries, Sen Wu, and Christopher Ré. “Snorkel: Rapid Training Data Creation with Weak Supervision”. In: *Proc. VLDB Endow.* 11.3 (2017), pp. 269–282 (cit. on p. 29).

[Rat+19] Alexander Ratner, Braden Hancock, Jared Dunnmon, Frederic Sala, Shreyash Pandey, and Christopher Ré. “Training Complex Models with Multi-Task Weak Supervision”. In: *Proc. of the 33rd AAAI Conference on Artificial Intelligence, AAAI, the 31st Innovative Applications of Artificial Intelligence Conference, IAAI, the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI, January 27 - February 1, Honolulu, HI, USA*. AAAI Press, 2019, pp. 4763–4771 (cit. on p. 30).

[Rat+16] Alexander Ratner, Christopher De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. “Data Programming: Creating Large Training Sets, Quickly”. In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems, NIPS, December 5-10, Barcelona, Spain*. Curran Associates, Inc., 2016, pp. 3567–3575 (cit. on p. 29).

[Ren+18] Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. “Learning to Reweight Examples for Robust Deep Learning”. In: *Proc. of the 35th International Conference on Machine Learning, ICML, July 10-15, Stockholmsmässan, Stockholm, Sweden*. Vol. 80. Proc. of Machine Learning Research. PMLR, 2018, pp. 4331–4340 (cit. on p. 18).

[Ren+20] Wendi Ren, Yinghao Li, Hanting Su, David Kartchner, Cassie S. Mitchell, and Chao Zhang. “Denoising Multi-Source Weak Supervision for Neural Text Classification”. In: *Findings of the Association for Computational Linguistics: EMNLP, November 16-20, virtual*. Vol. EMNLP 2020. Findings of ACL. Association for Computational Linguistics, 2020, pp. 3739–3754 (cit. on p. 30).

[RYS20] Zhongzheng Ren, Raymond A. Yeh, and Alexander G. Schwing. “Not All Unlabeled Data are Equal: Learning to Weight Data in Semi-Supervised Learning”. In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, virtual*. Curran Associates, Inc., 2020, pp. 21786–21797 (cit. on p. 50).

[Rif+11a] Salah Rifai, Yann N. Dauphin, Pascal Vincent, Yoshua Bengio, and Xavier Muller. “The Manifold Tangent Classifier”. In: *Advances in Neural Information Processing Systems 24: Annual Conference on Neural Information Processing Systems, NIPS, December 12-14, Granada, Spain*. Curran Associates, Inc., 2011, pp. 2294–2302 (cit. on p. 55).

[Rif+11b] Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio. “Contractive Auto-Encoders: Explicit Invariance During Feature Extraction”. In: *Proc. of the 28th International Conference on Machine Learning, ICML, June 28 - July 2, Bellevue, WA, USA*. Omnipress, 2011, pp. 833–840 (cit. on p. 55).

[Rig07] Philippe Rigollet. “Generalization Error Bounds in Semi-Supervised Classification Under the Cluster Assumption”. In: *J. Mach. Learn. Res.* 8 (2007), pp. 1369–1392 (cit. on p. 45).

[Riz+21] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S. Rawat, and Mubarak Shah. “In Defense of Pseudo-Labeling: An Uncertainty-Aware Pseudo-Label Selection Framework for Semi-Supervised Learning”. In: *9th International Conference on Learning Representations, ICLR, May 3-7, virtual*. OpenReview.net, 2021 (cit. on pp. 49, 50).

[Rod+23] Julian Rodemann, Jann Goschenhofer, Emilio Dorigatti, Thomas Nagler, and Thomas Augustin. “Approximate Bayes Optimal Pseudo-Label Selection”. In: *CoRR* abs/2302.08883 (2023) (cit. on p. 50).

[Rod+22] Julian Rodemann, Dominik Kreiss, Eyke Hüllermeier, and Thomas Augustin. “Levelwise Data Disambiguation by Cautious Superset Classification”. In: *Proc. of the 15th International Conference on Scalable Uncertainty Management, SUM, October 17-19, Paris, France*. Vol. 13562. Lecture Notes in Computer Science. Springer, 2022, pp. 263–276 (cit. on p. 36).

[RP18] Filipe Rodrigues and Francisco C. Pereira. “Deep Learning from Crowds”. In: *Proc. of the 32nd AAAI Conference on Artificial Intelligence, AAAI, February 2-7, New Orleans, LA, USA*. AAAI Press, 2018, pp. 1611–1618 (cit. on p. 27).

[Rüp10] Stefan Rüping. “SVM Classifier Estimation from Group Probabilities”. In: *Proc. of the 27th International Conference on Machine Learning, ICML, June 21-24, Haifa, Israel*. Omnipress, 2010, pp. 911–918 (cit. on p. 26).

[Sad+01] Javid Sadr, Sayan Mukherjee, K. Thoresz, and Pawan Sinha. “The Fidelity of Local Ordinal Encoding”. In: *Advances in Neural Information Processing Systems 14: Annual Conference on Neural Information Processing Systems, NIPS, December 3-8, Vancouver, BC, Canada*. MIT Press, 2001, pp. 1279–1286 (cit. on p. 62).

[SJT16] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. “Regularization With Stochastic Transformations and Perturbations for Deep Semi-Supervised Learning”. In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems, NIPS, December 5-10, Barcelona, Spain*. Curran Associates, Inc., 2016, pp. 1163–1171 (cit. on pp. 46, 54).

[SZ22] Ali Reza Sajun and Imran Zualkernan. “Survey on Implementations of Generative Adversarial Networks for Semi-Supervised Learning”. In: *Applied Sciences* 12.3 (2022) (cit. on p. 57).

[Sal+16] Tim Salimans, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. “Improved Techniques for Training GANs”. In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems, NIPS, December 5-10, Barcelona, Spain*. Curran Associates, Inc., 2016, pp. 2226–2234 (cit. on p. 56).

[Sch+18] Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. “Adversarially Robust Generalization Requires More Data”. In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems, NeurIPS, December 3-8, Montréal, Canada*. Curran Associates, Inc., 2018, pp. 5019–5031 (cit. on p. 175).

[Sch+22] Stefan C. Schonsheck, Scott Mahan, Timo Klock, Alexander Cloninger, and Rongjie Lai. “Semi-Supervised Manifold Learning with Complexity Decoupled Chart Autoencoders”. In: *CoRR* abs/2208.10570 (2022) (cit. on p. 47).

[SJ03] Matthew Schultz and Thorsten Joachims. “Learning a Distance Metric from Relative Comparisons”. In: *Advances in Neural Information Processing Systems 16: Annual Conference on Neural Information Processing Systems, NIPS, December 8-13, Vancouver and Whistler, BC, Canada*. MIT Press, 2003, pp. 41–48 (cit. on p. 62).

[Scu10] David Sculley. “Combined Regression and Ranking”. In: *Proc. of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, July 25-28, Washington, DC, USA*. ACM, 2010, pp. 979–988 (cit. on p. 63).

[Sha76] Glenn Shafer. *A Mathematical Theory of Evidence*. Vol. 42. Princeton University Press, 1976 (cit. on pp. 40, 43).

[SV08] Glenn Shafer and Vladimir Vovk. “A Tutorial on Conformal Prediction”. In: *J. Mach. Learn. Res.* 9 (2008), pp. 371–421 (cit. on p. 6).

[Sha+19] Bo Shao, Yeyun Gong, Junwei Bao, Jianshu Ji, Guihong Cao, Xiaola Lin, and Nan Duan. "Weakly Supervised Multi-Task Learning for Semantic Parsing". In: *Proc. of the 28th International Joint Conference on Artificial Intelligence, IJCAI, August 10-16, Macao, China*. ijcai.org, 2019, pp. 3375–3381 (cit. on p. 28).

[Sha+21] Zhuchen Shao, Hao Bian, Yang Chen, Yifeng Wang, Jian Zhang, Xiangyang Ji, and Yongbing Zhang. "TransMIL: Transformer based Correlated Multiple Instance Learning for Whole Slide Image Classification". In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-14, virtual*. Curran Associates, Inc., 2021, pp. 2136–2147 (cit. on p. 25).

[SPI08] Victor S. Sheng, Foster J. Provost, and Panagiotis G. Ipeirotis. "Get another Label? Improving Data Quality and Data Mining using Multiple, Noisy Labelers". In: *Proc. of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 24-27, Las Vegas, NV, USA*. ACM, 2008, pp. 614–622 (cit. on p. 27).

[She+22] Xiang-Rong Sheng, Jingyue Gao, Yueyao Cheng, Siran Yang, Shuguang Han, Hongbo Deng, Yuning Jiang, Jian Xu, and Bo Zheng. "Joint Optimization of Ranking and Calibration with Contextualized Hybrid Model". In: *CoRR* abs/2208.06164 (2022) (cit. on p. 63).

[SLH21] Ye Shi, Shao-Yuan Li, and Sheng-Jun Huang. "Learning from Crowds with Sparse and Imbalanced Annotations". In: *CoRR* abs/2107.05039 (2021) (cit. on p. 27).

[Shi+22] Changho Shin, Winfred Li, Harit Vishwakarma, Nicholas Carl Roberts, and Frederic Sala. "Universalizing Weak Supervision". In: *10th International Conference on Learning Representations, ICLR, April 25-29, virtual*. OpenReview.net, 2022 (cit. on p. 30).

[Sil+23] Telmo Silva Filho, Hao Song, Miquel Perello-Nieto, Raul Santos-Rodriguez, Meelis Kull, and Peter Flach. "Classifier Calibration: A Survey on how to Assess and Improve Predicted Class Probabilities". In: *Mach. Learn.* (2023), pp. 1–50 (cit. on p. 14).

[SNZ08] Aarti Singh, Robert D. Nowak, and Xiaojin Zhu. "Unlabeled Data: Now it Helps, Now it Doesn't". In: *Advances in Neural Information Processing Systems 21: Annual Conference on Neural Information Processing Systems, NIPS, December 8-11, Vancouver, BC, Canada*. Curran Associates, Inc., 2008, pp. 1513–1520 (cit. on p. 45).

[ST17] Dejan Slepcev and Matthew Thorpe. "Analysis of p-Laplacian Regularization in Semi-Supervised Learning". In: *CoRR* abs/1707.06213 (2017) (cit. on p. 55).

[Sno+08] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. "Cheap and Fast - But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks". In: *Proc. of the Conference on Empirical Methods in Natural Language Processing, EMNLP, October 25-27, Honolulu, HI, USA*. ACL, 2008, pp. 254–263 (cit. on p. 27).

[Soh+20] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. "FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, virtual*. Curran Associates, Inc., 2020, pp. 596–608 (cit. on pp. 49, 54).

[SLF88] Sara A. Solla, Esther Levin, and Michael Fleisher. "Accelerated Learning in Layered Neural Networks". In: *Complex Syst.* 2.6 (1988) (cit. on p. 13).

[Spi+21] Timo Spinde, Manuel Plank, Jan-David Krieger, Terry Ruas, Bela Gipp, and Akiko Aizawa. "Neural Media Bias Detection Using Distant Supervision With BABE - Bias Annotations By Experts". In: *Findings of the Association for Computational Linguistics: EMNLP, November 7-11, Punta Cana, Dominican Republic*. Association for Computational Linguistics, 2021, pp. 1166–1177 (cit. on p. 28).

[Spr16] Jost Tobias Springenberg. "Unsupervised and Semi-Supervised Learning with Categorical Generative Adversarial Networks". In: *4th International Conference on Learning Representations, ICLR, May 2-4, San Juan, Puerto Rico*. 2016 (cit. on p. 56).

[Sri+14] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". In: *J. Mach. Learn. Res.* 15.1 (2014), pp. 1929–1958 (cit. on p. 14).

[SBC05] Neil Stewart, Gordon DA Brown, and Nick Chater. "Absolute Identification by Relative Judgment". In: *Psychol. Rev.* 112.4 (2005), p. 881 (cit. on p. 58).

[SM11] Marco Stolpe and Katharina Morik. "Learning from Label Proportions by Optimizing Cluster Model Selection". In: *Proc. of the 22nd European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, September 5-9, Athens, Greece, Part III*. Vol. 6913. Lecture Notes in Computer Science. Springer, 2011, pp. 349–364 (cit. on p. 26).

[Sug+07] Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul von Bünau, and Motoaki Kawanabe. "Direct Importance Estimation with Model Selection and Its Application to Covariate Shift Adaptation". In: *Advances in Neural Information Processing Systems 20: Annual Conference on Neural Information Processing Systems, NIPS, December 3-6, Vancouver, BC, Canada*. Curran Associates, Inc., 2007, pp. 1433–1440 (cit. on p. 18).

[SBK20] Jiaze Sun, Binod Bhattarai, and Tae-Kyun Kim. “MatchGAN: A Self-Supervised Semi-Supervised Conditional Generative Adversarial Network”. In: *Proc. of the 15th Asian Conference on Computer Vision, ACCV, November 30 - December 4, Kyoto, Japan, Part IV*. Vol. 12625. Lecture Notes in Computer Science. Springer, 2020, pp. 608–623 (cit. on p. 57).

[Sze+16] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. “Rethinking the Inception Architecture for Computer Vision”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 27-30, Las Vegas, NV, USA*. IEEE Computer Society, 2016, pp. 2818–2826 (cit. on pp. 2, 14, 43).

[Tan+18] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. “A Survey on Deep Transfer Learning”. In: *Proc. of the 27th International Conference on Artificial Neural Networks and Machine Learning, ICANN, October 4-7, Rhodes, Greece, Part III*. Vol. 11141. Lecture Notes in Computer Science. Springer, 2018, pp. 270–279 (cit. on p. 28).

[Tan+14] Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. “Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification”. In: *Proc. of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL, Volume 1: Long Papers, June 22-27, Baltimore, MD, USA*. The Association for Computer Linguistics, 2014, pp. 1555–1565 (cit. on p. 27).

[Tan+19] Ryutaro Tanno, Ardavan Saeedi, Swami Sankaranarayanan, Daniel C. Alexander, and Nathan Silberman. “Learning From Noisy Labels by Regularized Estimation of Annotator Confusion”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 16-20, Long Beach, CA, USA*. Computer Vision Foundation / IEEE, 2019, pp. 11244–11253 (cit. on p. 27).

[Tao+20] Xinmin Tao, Qing Li, Chao Ren, Wenjie Guo, Qing He, Rui Liu, and Junrong Zou. “Affinity and Class Probability-Based Fuzzy Support Vector Machine for Imbalanced Data Sets”. In: *Neural Netw.* 122 (2020), pp. 289–307 (cit. on p. 18).

[TV17] Antti Tarvainen and Harri Valpola. “Mean Teachers are Better Role Models: Weight-Averaged Consistency Targets Improve Semi-Supervised Deep Learning Results”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, December 4-9, Long Beach, CA, USA*. Curran Associates, Inc., 2017, pp. 1195–1204 (cit. on p. 54).

[Tes88] Gerald Tesauro. “Connectionist Learning of Expert Preferences by Comparison Training”. In: *Advances in Neural Information Processing Systems 1: Annual Conference on Neural Information Processing Systems, NIPS, Denver, CO, USA*. Morgan Kaufmann, 1988, pp. 99–106 (cit. on p. 61).

[Tou+23] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. "LLaMA: Open and Efficient Foundation Language Models". In: *CoRR* abs/2302.13971 (2023) (cit. on p. 49).

[TEM07] Salsabil Trabelsi, Zied Elouedi, and Khaled Mellouli. "Pruning Belief Decision Tree Methods in Averaging and Conjunctive Approaches". In: *Int. J. Approx. Reason.* 46.3 (2007), pp. 568–595 (cit. on p. 42).

[TGH15] Isaac Triguero, Salvador García, and Francisco Herrera. "Self-Labeled Techniques for Semi-Supervised Learning: Taxonomy, Software and Empirical Study". In: *Knowl. Inf. Syst.* 42.2 (2015), pp. 245–284 (cit. on p. 48).

[TL20] Kuen-Han Tsai and Hsuan-Tien Lin. "Learning from Label Proportions with Consistency Regularization". In: *Proc. of the 12th Asian Conference on Machine Learning, ACML, November 18-20, Bangkok, Thailand*. Vol. 129. Proc. of Machine Learning Research. PMLR, 2020, pp. 513–528 (cit. on p. 26).

[Tsu+09] Yuta Tsuboi, Hisashi Kashima, Shohei Hido, Steffen Bickel, and Masashi Sugiyama. "Direct Density Ratio Estimation for Large-Scale Covariate Shift Adaptation". In: *J. Inf. Process.* 17 (2009), pp. 138–155 (cit. on p. 18).

[UC11] Lev V. Utkin and Frank P.A. Coolen. "Interval-Valued Regression and Classification Models in the Framework of Machine Learning". In: *7th International Symposium on Imprecise Probability: Theories and Applications, ISIPTA, July 25-28, Innsbruck, Austria*. Vol. 11. Citeseer. 2011, pp. 371–380 (cit. on p. 36).

[Vap95] Vladimir Naumovich Vapni. *The Nature of Statistical Learning Theory*. Springer, 1995 (cit. on p. 11).

[Vap91] Vladimir Vapnik. "Principles of Risk Minimization for Learning Theory". In: *Advances in Neural Information Processing Systems 4: Annual Conference on Neural Information Processing Systems, NIPS, December 2-5, Denver, CO, USA*. Morgan Kaufmann, 1991, pp. 831–838 (cit. on pp. 10, 11).

[Vap98] Vladimir Vapnik. *Statistical Learning Theory*. John Wiley & Sons, 1998 (cit. on p. 51).

[Var+23] Víctor Manuel Vargas, Pedro Antonio Gutiérrez, Javier Barbero-Gómez, and César Hervás-Martínez. "Soft Labelling based on Triangular Distributions for Ordinal Classification". In: *Inf. Fusion* 93 (2023), pp. 258–267 (cit. on pp. 15, 173).

[VR18] Paroma Varma and Christopher Ré. "Snuba: Automating Weak Supervision to Label Training Data". In: *Proc. VLDB Endow.* 12.3 (2018), pp. 223–236 (cit. on p. 29).

[Vas+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, NIPS, December 4-9, Long Beach, CA, USA*. Curran Associates, Inc., 2017, pp. 5998–6008 (cit. on p. 12).

[Vin+08] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. “Extracting and Composing Robust Features with Denoising Autoencoders”. In: *Proc. of the 25th International Conference on Machine Learning, ICML, June 5-9, Helsinki, Finland*. Vol. 307. ACM International Conference Proceeding Series. ACM, 2008, pp. 1096–1103 (cit. on p. 47).

[Vin+10] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. “Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion”. In: *J. Mach. Learn. Res.* 11 (2010), pp. 3371–3408 (cit. on p. 48).

[Vog+20] Robin Vogel, Mastane Achab, Stéphan Clémençon, and Charles Tillier. “Weighted Empirical Risk Minimization: Sample Selection Bias Correction based on Importance Sampling”. In: *CoRR* abs/2002.05145 (2020) (cit. on p. 19).

[Wag+20] Nicolas Wagner, Violaine Antoine, Jonas Koko, and Romain Lardy. “Fuzzy k-NN Based Classifiers for Time Series with Soft Labels”. In: *Proc. of the 18th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU, June 15-19, Lisbon, Portugal, Part III*. Vol. 1239. Communications in Computer and Information Science. Springer, 2020, pp. 578–589 (cit. on p. 39).

[Wah+14] Catherine Wah, Grant Van Horn, Steve Branson, Subhransu Maji, Pietro Perona, and Serge J. Belongie. “Similarity Comparisons for Interactive Fine-Grained Categorization”. In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 23-28, Columbus, OH, USA*. IEEE Computer Society, 2014, pp. 859–866 (cit. on p. 62).

[Wal45] Abraham Wald. “Statistical Decision Functions Which Minimize the Maximum Risk”. In: *Ann. Math.* 46 (1945), p. 265 (cit. on p. 37).

[Wal91] Peter Walley. *Statistical Reasoning with Imprecise Probabilities*. Vol. 42. Chapman & Hall, 1991 (cit. on p. 173).

[Wan+13] Li Wan, Matthew D. Zeiler, Sixin Zhang, Yann LeCun, and Rob Fergus. “Regularization of Neural Networks using DropConnect”. In: *Proc. of the 30th International Conference on Machine Learning, ICML, June 16-21, Atlanta, GA, USA*. Vol. 28. JMLR Workshop and Conference Proc. JMLR.org, 2013, pp. 1058–1066 (cit. on p. 14).

[Wan09] Xiaojun Wan. “Co-Training for Cross-Lingual Sentiment Classification”. In: *Proc. of the 47th Annual Meeting of the Association for Computational Linguistics, and the 4th International Joint Conference on Natural Language Processing, ACL/IJCNLP, August 2-7, Singapore*. The Association for Computer Linguistics, 2009, pp. 235–243 (cit. on p. 49).

[Wan+21] Boxin Wang, Chejian Xu, Shuohang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Hassan Awadallah, and Bo Li. “Adversarial GLUE: A Multi-Task Benchmark for Robustness Evaluation of Language Models”. In: *Proc. of the Neural Information Processing Systems Track on Datasets and Benchmarks, NeurIPS Datasets and Benchmarks, December, virtual*. 2021 (cit. on p. 175).

[WFZ21] Deng-Bao Wang, Lei Feng, and Min-Ling Zhang. “Rethinking Calibration of Deep Neural Networks: Do Not Be Afraid of Overconfidence”. In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-14, virtual*. Curran Associates, Inc., 2021, pp. 11809–11820 (cit. on pp. 2, 14).

[WZL22] Deng-Bao Wang, Min-Ling Zhang, and Li Li. “Adaptive Graph Guided Disambiguation for Partial Label Learning”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 44.12 (2022), pp. 8796–8811 (cit. on p. 34).

[Wan+20] Haobo Wang, Yuzhou Qiang, Chen Chen, Weiwei Liu, Tianlei Hu, Zhao Li, and Gang Chen. “Online Partial Label Learning”. In: *Proc. of the 31rd European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, September 14-18, Ghent, Belgium, Part II*. Vol. 12458. Lecture Notes in Computer Science. Springer, 2020, pp. 455–470 (cit. on p. 34).

[Wan+22a] Haobo Wang, Ruixuan Xiao, Yixuan Li, Lei Feng, Gang Niu, Gang Chen, and Junbo Zhao. “PiCO: Contrastive Label Disambiguation for Partial Label Learning”. In: *10th International Conference on Learning Representations, ICLR, April 25-29, virtual*. OpenReview.net, 2022 (cit. on p. 34).

[WS07] Junhui Wang and Xiaotong Shen. “Large Margin Semi-Supervised Learning”. In: *J. Mach. Learn. Res.* 8 (2007), pp. 1867–1891 (cit. on p. 51).

[WZ16] Lu Wang and Zhi-Hua Zhou. “Cost-Saving Effect of Crowdsourcing Learning”. In: *Proc. of the 25th International Joint Conference on Artificial Intelligence, IJCAI, July 9-15, New York, NY, USA*. IJCAI/AAAI Press, 2016, pp. 2111–2117 (cit. on p. 27).

[Wan+19a] Mei Wang, Weihong Deng, Jiani Hu, Xunqiang Tao, and Yaohai Huang. “Racial Faces in the Wild: Reducing Racial Bias by Information Maximization Adaptation Network”. In: *Proc. of the IEEE International Conference on Computer Vision, ICCV, October 27 - November 2, Seoul, Korea (South)*. IEEE, 2019, pp. 692–702 (cit. on p. 18).

[WZ07] Wei Wang and Zhi-Hua Zhou. “Analyzing Co-Training Style Algorithms”. In: *Proc. of the 18th European Conference on Machine Learning, ECML, September 17-21, Warsaw, Poland*. Vol. 4701. Lecture Notes in Computer Science. Springer, 2007, pp. 454–465 (cit. on p. 49).

[Wan+22b] Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, and Bernt Schiele. “FreeMatch: Self-Adaptive Thresholding for Semi-Supervised Learning”. In: *CoRR* abs/2205.07246 (2022) (cit. on p. 50).

[Wan+19b] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. “Symmetric Cross Entropy for Robust Learning With Noisy Labels”. In: *Proc. of the IEEE International Conference on Computer Vision, ICCV, October 27 - November 2, Seoul, Korea (South)*. IEEE, 2019, pp. 322–330 (cit. on p. 14).

[Wan+22c] Yuchao Wang, Haochen Wang, Yujun Shen, Jingjing Fei, Wei Li, Guoqiang Jin, Liwei Wu, Rui Zhao, and Xinyi Le. “Semi-Supervised Semantic Segmentation Using Unreliable Pseudo-Labels”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 18-24, New Orleans, LA, USA*. IEEE, 2022, pp. 4238–4247 (cit. on p. 49).

[WSZ08] Zheng Wang, Yangqiu Song, and Changshui Zhang. “Transferred Dimensionality Reduction”. In: *Proc. of the 19th European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD, September 15-19, Antwerp, Belgium, Part II*. Vol. 5212. Lecture Notes in Computer Science. Springer, 2008, pp. 550–565 (cit. on p. 29).

[WS16] Oliver Wasenmüller and Didier Stricker. “Comparison of Kinect V1 and V2 Depth Images in Terms of Accuracy and Precision”. In: *Proc. of the 13th Asian Conference on Computer Vision, ACCV, November 20-24, Taipei, Taiwan, Workshops, Part II*. Vol. 10117. Lecture Notes in Computer Science. Springer, 2016, pp. 34–45 (cit. on p. 17).

[WD81] Frits T. Beukema toe Water and Robert P. W. Duin. “Dealing with A Priori Knowledge by Fuzzy Labels”. In: *Pattern Recognit.* 14.1-6 (1981), pp. 111–115 (cit. on p. 38).

[Web+15] Jordannah Webb, Dieuwerke P. Bolhuis, Sara Cicerale, John E. Hayes, and Russell S J Keast. “The Relationships Between Common Measurements of Taste Function”. In: *Chemosens. Percept.* 8 (2015), pp. 11–18 (cit. on p. 59).

[Wei+21] Colin Wei, Kendrick Shen, Yining Chen, and Tengyu Ma. “Theoretical Analysis of Self-Training with Deep Networks on Unlabeled Data”. In: *9th International Conference on Learning Representations, ICLR, May 3-7, virtual*. OpenReview.net, 2021 (cit. on pp. 45, 52).

[Wei+18] Xiang Wei, Boqing Gong, Zixia Liu, Wei Lu, and Liqiang Wang. “Improving the Improved Training of Wasserstein GANs: A Consistency Term and Its Dual Effect”. In: *6th International Conference on Learning Representations, ICLR, April 30 - May 3, Vancouver, BC, Canada*. OpenReview.net, 2018 (cit. on p. 57).

[Wen+21] Hongwei Wen, Jingyi Cui, Hanyuan Hang, Jiabin Liu, Yisen Wang, and Zhouchen Lin. “Leveraged Weighted Loss for Partial Label Learning”. In: *Proc. of the 38th International Conference on Machine Learning, ICML, July 18-24, virtual*. Vol. 139. Proc. of Machine Learning Research. PMLR, 2021, pp. 11091–11100 (cit. on pp. 33, 34).

[WGS15] Junfeng Wen, Russell Greiner, and Dale Schuurmans. “Correcting Covariate Shift with the Frank-Wolfe Algorithm”. In: *Proc. of the 24th International Joint Conference on Artificial Intelligence, IJCAI, July 25-31, Buenos Aires, Argentina*. AAAI Press, 2015, pp. 1010–1016 (cit. on p. 18).

[WRC08] Jason Weston, Frédéric Ratle, and Ronan Collobert. “Deep Learning via Semi-Supervised Embedding”. In: *Proc. of the 25th International Conference on Machine Learning, ICML, June 5-9, Helsinki, Finland*. Vol. 307. ACM International Conference Proceeding Series. ACM, 2008, pp. 1168–1175 (cit. on p. 55).

[Whi+09] Jacob Whitehill, Paul Ruvolo, Tingfan Wu, Jacob Bergsma, and Javier R. Movellan. “Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise”. In: *Advances in Neural Information Processing Systems 22: Annual Conference on Neural Information Processing Systems, NIPS, December 7-10, Vancouver, BC, Canada*. Curran Associates, Inc., 2009, pp. 2035–2043 (cit. on p. 27).

[WM05] Cort J. Willmott and Kenji Matsuura. “Advantages of the Mean Absolute Error (MAE) over the Root Mean Square Error (RMSE) in Assessing Average Model Performance”. In: *Clim. Res.* 30.1 (2005), pp. 79–82 (cit. on p. 17).

[WFT12] Aaron Wilson, Alan Fern, and Prasad Tadepalli. “A Bayesian Approach for Policy Learning from Trajectory Preference Queries”. In: *Advances in Neural Information Processing Systems 25: Annual Conference on Neural Information Processing Systems, NIPS, December 3-6, Lake Tahoe, NV, USA*. Curran Associates, Inc., 2012, pp. 1142–1150 (cit. on p. 62).

[WM16] Nic Wilson and Mojtaba Montazery. “Preference Inference through Rescaling Preference Learning”. In: *Proc. of the 25th International Joint Conference on Artificial Intelligence, IJCAI, July 9-15, New York, NY, USA*. IJCAI/AAAI Press, 2016, pp. 2203–2209 (cit. on p. 61).

[Wir+17] Christian Wirth, Riad Akrour, Gerhard Neumann, and Johannes Fürnkranz. “A Survey of Preference-Based Reinforcement Learning Methods”. In: *J. Mach. Learn. Res.* 18 (2017), 136:1–136:46 (cit. on p. 62).

[WF13] Christian Wirth and Johannes Fürnkranz. “A Policy Iteration Algorithm for Learning from Preference-Based Feedback”. In: *Proc. of the 12th International Symposium on Intelligent Data Analysis, IDA, October 17-19, London, United Kingdom*. Vol. 8207. Lecture Notes in Computer Science. Springer, 2013, pp. 427–437 (cit. on p. 62).

[WFN16] Christian Wirth, Johannes Fürnkranz, and Gerhard Neumann. “Model-Free Preference-Based Reinforcement Learning”. In: *Proc. of the 30th AAAI Conference on Artificial Intelligence, AAAI, February 12-17, Phoenix, AZ, USA*. AAAI Press, 2016, pp. 2222–2228 (cit. on p. 62).

[WWZ22] Dong-Dong Wu, Deng-Bao Wang, and Min-Ling Zhang. “Revisiting Consistency Regularization for Deep Partial Label Learning”. In: *Proc. of the 39th International Conference on Machine Learning, ICML, July 17-23, Baltimore, MD, USA*. Vol. 162. Proc. of Machine Learning Research. PMLR, 2022, pp. 24212–24225 (cit. on p. 34).

[Wu+19] Si Wu, Guangchang Deng, Jichang Li, Rui Li, Zhiwen Yu, and Hau-San Wong. "Enhancing TripleGAN for Semi-Supervised Conditional Instance Synthesis and Classification". In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 16-20, Long Beach, CA, USA*. Computer Vision Foundation / IEEE, 2019, pp. 10091–10100 (cit. on p. 57).

[Xia+08] Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. "Listwise Approach to Learning to Rank: Theory and Algorithm". In: *Proc. of the 25th International Conference on Machine Learning, ICML, June 5-9, Helsinki, Finland*. Vol. 307. ACM International Conference Proceeding Series. ACM, 2008, pp. 1192–1199 (cit. on p. 61).

[Xia19] Lirong Xia. *Learning and Decision-Making from Rank Data*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2019 (cit. on p. 58).

[XPX18] Rui Xia, Zhenchun Pan, and Feng Xu. "Instance Weighting with Applications to Cross-Domain Text Classification via Trading off Sample Selection Bias and Variance". In: *Proc. of the 27th International Joint Conference on Artificial Intelligence, IJCAI, July 13-19, Stockholm, Sweden*. ijcai.org, 2018, pp. 4489–4495 (cit. on p. 18).

[Xia+18] Ke Xian, Chunhua Shen, Zhiguo Cao, Hao Lu, Yang Xiao, Ruibo Li, and Zhenbo Luo. "Monocular Relative Depth Perception With Web Stereo Data Supervision". In: *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 18-22, Salt Lake City, UT, USA*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 311–320 (cit. on p. 58).

[Xia+20a] Ke Xian, Jianming Zhang, Oliver Wang, Long Mai, Zhe Lin, and Zhiguo Cao. "Structure-Guided Ranking Loss for Single Image Depth Prediction". In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 13-19, Seattle, WA, USA*. Computer Vision Foundation / IEEE, 2020, pp. 608–617 (cit. on p. 62).

[Xia+20b] Xuezhi Xiang, Zeting Yu, Ning Lv, Xiangdong Kong, and Abdulmotaleb El-Saddik. "Attention-Based Generative Adversarial Network for Semi-Supervised Image Classification". In: *Neural Process. Lett.* 51.2 (2020), pp. 1527–1540 (cit. on p. 57).

[Xie+20] Qizhe Xie, Zihang Dai, Eduard H. Hovy, Thang Luong, and Quoc Le. "Unsupervised Data Augmentation for Consistency Training". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, virtual*. Curran Associates, Inc., 2020, pp. 6256–6268 (cit. on p. 54).

[Xio+15] Sicheng Xiong, Yuanli Pei, Rómer Rosales, and Xiaoli Z. Fern. "Active Learning from Relative Comparisons". In: *IEEE Trans. Knowl. Data Eng.* 27.12 (2015), pp. 3166–3175 (cit. on p. 62).

[XD20] Austin Xu and Mark A. Davenport. “Simultaneous Preference and Metric Learning from Paired Comparisons”. In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, virtual*. Curran Associates, Inc., 2020, pp. 454–465 (cit. on p. 62).

[XLG19] Ning Xu, Jiaqi Lv, and Xin Geng. “Partial Label Learning via Label Enhancement”. In: *Proc. of the 33rd AAAI Conference on Artificial Intelligence, AAAI, the 31st Innovative Applications of Artificial Intelligence Conference, IAAI, the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI, January 27 - February 1, Honolulu, HI, USA*. AAAI Press, 2019, pp. 5557–5564 (cit. on p. 34).

[Xu+20] Yanwu Xu, Mingming Gong, Junxiang Chen, Tongliang Liu, Kun Zhang, and Kayhan Batmanghelich. “Generative-Discriminative Complementary Learning”. In: *Proc. of the 34th AAAI Conference on Artificial Intelligence, AAAI, the 32nd Innovative Applications of Artificial Intelligence Conference, IAAI, the 10th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI, February 7-12, New York, NY, USA*. AAAI Press, 2020, pp. 6526–6533 (cit. on p. 23).

[Xu+21] Yi Xu, Jiandong Ding, Lu Zhang, and Shuigeng Zhou. “DP-SSL: Towards Robust Semi-Supervised Learning with A Few Labeled Samples”. In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-14, virtual*. Curran Associates, Inc., 2021, pp. 15895–15907 (cit. on p. 29).

[YL08] Ronald R. Yager and Liping Liu, eds. *Classic Works of the Dempster-Shafer Theory of Belief Functions*. Vol. 219. Studies in Fuzziness and Soft Computing. Springer, 2008 (cit. on p. 40).

[YG20] Yan Yan and Yuhong Guo. “Partial Label Learning with Batch Label Correction”. In: *Proc. of the 34th AAAI Conference on Artificial Intelligence, AAAI, the 32nd Innovative Applications of Artificial Intelligence Conference, IAAI, the 10th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI, February 7-12, New York, NY, USA*. AAAI Press, 2020, pp. 6575–6582 (cit. on p. 34).

[Yan+23] Lihe Yang, Lei Qi, Litong Feng, Wayne Zhang, and Yinghuan Shi. “Revisiting Weak-to-Strong Consistency in Semi-Supervised Semantic Segmentation”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 18-22, Vancouver, BC, Canada*. 2023, pp. 7236–7246 (cit. on pp. 49, 54).

[YLJ23] Wenjun Yang, Chaoqun Li, and Liangxiao Jiang. “Learning from Crowds with Robust Support Vector Machines”. In: *Sci. China Inf. Sci.* 66.3 (2023) (cit. on p. 27).

[Yan+22] Xiangli Yang, Zixing Song, Irwin King, and Zenglin Xu. “A Survey on Deep Semi-Supervised Learning”. In: *IEEE Trans. Knowl. Data Eng.* (2022), pp. 1–20 (cit. on pp. 44, 53, 54, 56).

[YSW07] XuLei Yang, Qing Song, and Yue Wang. “A Weighted Support Vector Machine for Data Classification”. In: *Int. J. Pattern Recognit. Artif. Intell.* 21.5 (2007), pp. 961–976 (cit. on p. 18).

[Yao+20] Yao Yao, Jiehui Deng, Xiuhua Chen, Chen Gong, Jianxin Wu, and Jian Yang. “Deep Discriminative CNN with Temporal Ensembling for Ambiguously-Labeled Image Classification”. In: *Proc. of the 34th AAAI Conference on Artificial Intelligence, AAAI, the 32nd Innovative Applications of Artificial Intelligence Conference, IAAI, the 10th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI, February 7-12, New York, NY, USA*. AAAI Press, 2020, pp. 12669–12676 (cit. on p. 33).

[Yao+21] Yuan Yao, Ao Zhang, Xu Han, Mengdi Li, Cornelius Weber, Zhiyuan Liu, Stefan Wermter, and Maosong Sun. “Visual Distant Supervision for Scene Graph Generation”. In: *Proc. of the IEEE/CVF International Conference on Computer Vision, ICCV, October 10-17, Montreal, QC, Canada*. IEEE, 2021, pp. 15796–15806 (cit. on p. 28).

[Yar95] David Yarowsky. “Unsupervised Word Sense Disambiguation Rivaling Supervised Methods”. In: *Proc. of the 33rd Annual Meeting of the Association for Computational Linguistics, ACL, June 26-30, Cambridge, MA, USA*. Morgan Kaufmann Publishers / ACL, 1995, pp. 189–196 (cit. on p. 49).

[Yil+19] Ilkay Yildiz, Peng Tian, Jennifer G. Dy, Deniz Erdogmus, James M. Brown, Jayashree Kalpathy-Cramer, Susan Ostmo, J. Peter Campbell, Michael F. Chiang, and Stratis Ioannidis. “Classification and Comparison via Neural Networks”. In: *Neural Netw.* 118 (2019), pp. 65–80 (cit. on p. 61).

[Yu+14] Felix X. Yu, Sanjiv Kumar, Tony Jebara, and Shih-Fu Chang. “On Learning with Label Proportions”. In: *CoRR* abs/1402.5902 (2014) (cit. on p. 26).

[Yu+13] Felix X. Yu, Dong Liu, Sanjiv Kumar, Tony Jebara, and Shih-Fu Chang. “$\propto$SVM for Learning with Label Proportions”. In: *Proc. of the 30th International Conference on Machine Learning, ICML, June 16-21, Atlanta, GA, USA*. Vol. 28. JMLR Workshop and Conference Proc. JMLR.org, 2013, pp. 504–512 (cit. on p. 26).

[Yu+18] Xiyu Yu, Tongliang Liu, Mingming Gong, and Dacheng Tao. “Learning with Biased Complementary Labels”. In: *Proc. of the 15th European Conference on Computer Vision, ECCV, September 8-14, Munich, Germany, Part I*. Vol. 11205. Lecture Notes in Computer Science. Springer, 2018, pp. 69–85 (cit. on p. 23).

[Zad65] Lotfi A. Zadeh. “Fuzzy Sets”. In: *Inf. Control.* 8.3 (1965), pp. 338–353 (cit. on p. 38).

[Zad99] Lotfi A. Zadeh. “Fuzzy Sets as a Basis for a Theory of Possibility”. In: *Fuzzy Sets Syst.* 100 (1999), pp. 9–34 (cit. on p. 40).

[Zad04] Bianca Zadrozny. “Learning and Evaluating Classifiers under Sample Selection Bias”. In: *Proc. of the 21st International Conference on Machine Learning, ICML, July 4-8, Banff, AB, Canada*. Vol. 69. ACM International Conference Proceeding Series. ACM, 2004 (cit. on p. 18).

[ZLA03] Bianca Zadrozny, John Langford, and Naoki Abe. “Cost-Sensitive Learning by Cost-Proportionate Example Weighting”. In: *Proc. of the 3rd IEEE International Conference on Data Mining, ICDM, December 19-22, Melbourne, FL, USA*. IEEE Computer Society, 2003, p. 435 (cit. on p. 18).

[Zen+15] Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. “Distant Supervision for Relation Extraction via Piecewise Convolutional Neural Networks”. In: *Proc. of the Conference on Empirical Methods in Natural Language Processing, EMNLP, September 17-21, Lisbon, Portugal*. The Association for Computational Linguistics, 2015, pp. 1753–1762 (cit. on p. 27).

[ZL21] Zijian Zeng and Meng Li. “Bayesian Median Autoregression for Robust Time Series Forecasting”. In: *Int. J. Forecast.* 37.2 (2021), pp. 1000–1010 (cit. on p. 17).

[Zha+21a] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. “FlexMatch: Boosting Semi-Supervised Learning with Curriculum Pseudo Labeling”. In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-14, virtual*. Curran Associates, Inc., 2021, pp. 18408–18419 (cit. on pp. 50, 54).

[Zha+22a] Fei Zhang, Lei Feng, Bo Han, Tongliang Liu, Gang Niu, Tao Qin, and Masashi Sugiyama. “Exploiting Class Activation Value for Partial-Label Learning”. In: *10th International Conference on Learning Representations, ICLR, April 25-29, virtual*. OpenReview.net, 2022 (cit. on p. 34).

[ZWS22] Jianxin Zhang, Yutong Wang, and Clayton Scott. “Learning from Label Proportions by Learning with Label Noise”. In: *CoRR* abs/2203.02496 (2022) (cit. on p. 26).

[Zha+22b] Jieyu Zhang, Cheng-Yu Hsieh, Yue Yu, Chao Zhang, and Alexander Ratner. “A Survey on Programmatic Weak Supervision”. In: *CoRR* abs/2202.05433 (2022) (cit. on p. 29).

[Zha+21b] Jieyu Zhang, Yue Yu, Yinghao Li, Yujing Wang, Yaming Yang, Mao Yang, and Alexander Ratner. “WRENCH: A Comprehensive Benchmark for Weak Supervision”. In: *Proc. of the Neural Information Processing Systems Track on Datasets and Benchmarks, NeurIPS Datasets and Benchmarks, December, virtual*. Vol. 1. 2021 (cit. on p. 29).

[ZY15] Min-Ling Zhang and Fei Yu. “Solving the Partial Label Learning Problem: An Instance-Based Approach”. In: *Proc. of the 24th International Joint Conference on Artificial Intelligence, IJCAI, July 25-31, Buenos Aires, Argentina*. AAAI Press, 2015, pp. 4048–4054 (cit. on p. 34).

[ZZL16] Min-Ling Zhang, Bin-Bin Zhou, and Xu-Ying Liu. “Partial Label Learning via Feature-Aware Disambiguation”. In: *Proc. of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 13-17, San Francisco, CA, USA*. ACM, 2016, pp. 1335–1344 (cit. on p. 34).

[Zha21] Weijia Zhang. “Non-I.I.D. Multi-Instance Learning for Predicting Instance and Bag Labels with Variational Auto-Encoder”. In: *Proc. of the 30th International Joint Conference on Artificial Intelligence, IJCAI, August 19-27, virtual*. ijcai.org, 2021, pp. 3377–3383 (cit. on p. 25).

[Zha+22c] Wenqiao Zhang, Lei Zhu, James Hallinan, Shengyu Zhang, Andrew Makmur, Qingpeng Cai, and Beng Chin Ooi. “BoostMIS: Boosting Medical Image Semi-Supervised Learning with Adaptive Pseudo Labeling and Informative Active Annotation”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 18-24, New Orleans, LA, USA*. IEEE, 2022, pp. 20634–20644 (cit. on p. 45).

[ZY22] Yu Zhang and Qiang Yang. “A Survey on Multi-Task Learning”. In: *IEEE Trans. Knowl. Data Eng.* 34.12 (2022), pp. 5586–5609 (cit. on p. 28).

[Zha+20a] Chaoqiang Zhao, Qiyu Sun, Chongzhen Zhang, Yang Tang, and Feng Qian. “Monocular Depth Estimation Based On Deep Learning: An Overview”. In: *CoRR* abs/2003.06620 (2020) (cit. on p. 16).

[Zha+20b] Huimin Zhao, Jianjie Zheng, Wu Deng, and Yingjie Song. “Semi-Supervised Broad Learning System Based on Manifold Regularization and Broad Network”. In: *IEEE Trans. Circuits Syst. I Regul. Pap.* 67-I.3 (2020), pp. 983–994 (cit. on p. 55).

[ZPX16] Zhibing Zhao, Peter Piech, and Lirong Xia. “Learning Mixtures of Plackett-Luce Models”. In: *Proc. of the 33nd International Conference on Machine Learning, ICML, June 19-24, New York City, NY, USA*. Vol. 48. JMLR Workshop and Conference Proc. JMLR.org, 2016, pp. 2906–2914 (cit. on p. 61).

[ZX19] Zhibing Zhao and Lirong Xia. “Learning Mixtures of Plackett-Luce Models from Structured Partial Orders”. In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems, NeurIPS, December 8-14, Vancouver, BC, Canada*. Curran Associates, Inc., 2019, pp. 10143–10153 (cit. on p. 61).

[ZGX13] Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. “Reidentification by Relative Distance Comparison”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 35.3 (2013), pp. 653–668 (cit. on p. 62).

[Zho+03] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. “Learning with Local and Global Consistency”. In: *Advances in Neural Information Processing Systems 16: Annual Conference on Neural Information Processing Systems, NIPS, December 8-13, Vancouver and Whistler, BC, Canada*. MIT Press, 2003, pp. 321–328 (cit. on pp. 45, 291).

[ZG04] Yan Zhou and Sally A. Goldman. “Democratic Co-Learning”. In: *Proc. of the 16th IEEE International Conference on Tools with Artificial Intelligence, ICTAI, November 15-17, Boca Raton, FL, USA*. IEEE Computer Society, 2004, pp. 594–602 (cit. on p. 49).

[Zho17] Zhi-Hua Zhou. “A Brief Introduction to Weakly Supervised Learning”. In: *Natl. Sci. Rev.* 5.1 (2017), pp. 44–53 (cit. on pp. 1, 21, 24).

[ZL10] Zhi-Hua Zhou and Ming Li. “Semi-Supervised Learning by Disagreement”. In: *Knowl. Inf. Syst.* 24.3 (2010), pp. 415–439 (cit. on p. 49).

[ZSL09] Zhi-Hua Zhou, Yu-Yin Sun, and Yu-Feng Li. “Multi-Instance Learning by Treating Instances as non-I.I.D. Samples”. In: *Proc. of the 26th International Conference on Machine Learning, ICML, June 14-18, Montreal, QC, Canada*. Vol. 382. ACM International Conference Proceeding Series. ACM, 2009, pp. 1249–1256 (cit. on p. 25).

[Zhu+22] Lei Zhu, Qi She, Qian Chen, Yunfei You, Boyu Wang, and Yanye Lu. “Weakly Supervised Object Localization as Domain Adaption”. In: *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, June 18-24, New Orleans, LA, USA*. IEEE, 2022, pp. 14617–14626 (cit. on p. 29).

[Zhu+17] Wentao Zhu, Qi Lou, Yeeleng Scott Vang, and Xiaohui Xie. “Deep Multi-Instance Networks with Sparse Label Assignment for Whole Mammogram Classification”. In: *Proc. of the 20th International Conference on Medical Image Computing and Computer Assisted Intervention, MICCAI, September 11-13, Quebec City, QC, Canada, Part III*. Vol. 10435. Lecture Notes in Computer Science. Springer, 2017, pp. 603–611 (cit. on p. 25).

[Zhu10] Xiaojin Zhu. “Semi-Supervised Learning”. In: *Encyclopedia of Machine Learning*. Springer, 2010, pp. 892–897 (cit. on p. 51).

[ZL05] Xiaojin Zhu and John D. Lafferty. “Harmonic Mixtures: Combining Mixture Models and Graph-Based Methods for Inductive and Scalable Semi-Supervised Learning”. In: *Proc. of the 22nd International Conference on Machine Learning, ICML, August 7-11, Bonn, Germany*. Vol. 119. ACM International Conference Proceeding Series. ACM, 2005, pp. 1052–1059 (cit. on p. 55).

[Zhu05] Xiaojin Jerry Zhu. “Semi-Supervised Learning Literature Survey”. In: (2005) (cit. on pp. 45, 48).

[ZSL21] Zhaowei Zhu, Yiwen Song, and Yang Liu. “Clusterability as an Alternative to Anchor Points When Learning with Noisy Labels”. In: *Proc. of the 38th International Conference on Machine Learning, ICML, July 18-24, virtual*. Vol. 139. Proc. of Machine Learning Research. PMLR, 2021, pp. 12912–12923 (cit. on p. 14).

[Zhu+21] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. “A Comprehensive Survey on Transfer Learning”. In: *Proc. IEEE* 109.1 (2021), pp. 43–76 (cit. on p. 28).

[Zie+19] Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul F. Christiano, and Geoffrey Irving. “Fine-Tuning Language Models from Human Preferences”. In: *CoRR* abs/1909.08593 (2019) (cit. on p. 62).

[Zor+15] Daniel Zoran, Phillip Isola, Dilip Krishnan, and William T. Freeman. "Learning Ordinal Relationships for Mid-Level Vision". In: *Proc. of the IEEE International Conference on Computer Vision, ICCV, December 7-13, Santiago, Chile*. IEEE Computer Society, 2015, pp. 388–396 (cit. on p. 62).

[ZV11] Roelof van Zwol and Srinivas Vadrevu. "Object Ranking". In: *Proc. of the 20th ACM International Conference on Information and Knowledge Management, CIKM, October 24-28, Glasgow, United Kingdom*. ACM, 2011, pp. 2613–2614 (cit. on p. 58).

Webpages

[@Ope22] OpenAI. *Introducing ChatGPT*. 2022. URL: `https://openai.com/blog/chatgpt` (visited on Apr. 20, 2023) (cit. on pp. 1, 62).

A

Appendix to From Label Smoothing to Label Relaxation

From Label Smoothing to Label Relaxation[*]
- Supplementary Material -

Julian Lienen, Eyke Hüllermeier
Heinz Nixdorf Institute and Department of Computer Science
Paderborn University, Germany
{julian.lienen,eyke}@upb.de

A Appendix

A.1 Analytical Determination of Equation (5)

Proposition 1. *Let $Q^\alpha := Q_i^\alpha$ be the target set as defined in Equation (3) for the true class $y := y_i \in \mathcal{Y}$. Moreover, let $\hat{p} \notin Q^\alpha$ be a probability distribution over the classes $\mathcal{Y}$ and L^* denote the label relaxation loss as defined in Equation (4). Then, $p^r \in \min_{p \in Q^\alpha} D_{KL}(p||\hat{p})$.*

Proof. The statement of this proposition can also be rephrased as follows: $\forall p \in Q^\alpha$ with $p \neq p^r$, $L^*(p, \hat{p}) \geq L^*(p^r, \hat{p})$. To prove the statement, let us assume that $\exists p \in Q^\alpha : L^*(p, \hat{p}) < L^*(p^r, \hat{p})$, which would be a contradiction.

We know that

$$\begin{aligned}
L^*(p^r, \hat{p}) = &\ (1-\alpha) \log \frac{1-\alpha}{\hat{p}(y)} + \sum_{y' \in \mathcal{Y} : y' \neq y} \frac{\alpha \cdot \hat{p}(y')}{\sum_{\hat{y} \in \mathcal{Y} : \hat{y} \neq y} \hat{p}(\hat{y})} \log \frac{\frac{\alpha \cdot \hat{p}(y')}{\sum_{\hat{y} \in \mathcal{Y} : \hat{y} \neq y} \hat{p}(\hat{y})}}{\hat{p}(y')} \\
= &\ (1-\alpha) \log \frac{1-\alpha}{\hat{p}(y)} + \sum_{y' \in \mathcal{Y} : y' \neq y} \frac{\alpha \cdot \hat{p}(y')}{1 - \hat{p}(y)} \log \frac{\alpha}{1 - \hat{p}(y)} \\
= &\ (1-\alpha) \log \frac{1-\alpha}{\hat{p}(y)} + \frac{\alpha}{1 - \hat{p}(y)} \log \frac{\alpha}{1 - \hat{p}(y)} \sum_{y' \in \mathcal{Y} : y' \neq y} \hat{p}(y') \\
= &\ (1-\alpha) \log \frac{1-\alpha}{\hat{p}(y)} + \frac{\alpha \cdot (1 - \hat{p}(y))}{1 - \hat{p}(y)} \log \frac{\alpha}{1 - \hat{p}(y)} \\
= &\ (1-\alpha) \log \frac{1-\alpha}{\hat{p}(y)} + \alpha \log \frac{\alpha}{1 - \hat{p}(y)} \quad .
\end{aligned}$$

[*] To appear in *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, February 2021.

Thus, the loss simplifies to the Kullback-Leibler divergence on the two cases of the class being y or any other class. Without loss of generality, we will use this simplification in the further course.

Together with

$$L^*(p,\hat{p}) = p(y)\log\frac{p(y)}{\hat{p}(y)} + \sum_{y'\in\mathcal{Y}:y'\neq y} p(y')\log\frac{p(y')}{\hat{p}(y')} \ ,$$

we get

$$p(y)\log\frac{p(y)}{\hat{p}(y)} + \sum_{y'\in\mathcal{Y}:y'\neq y} p(y')\log\frac{p(y')}{\hat{p}(y')} - (1-\alpha)\log\frac{1-\alpha}{\hat{p}(y)} - \alpha\log\frac{\alpha}{1-\hat{p}(y)} < 0 \ .$$

Using the log sum inequality[1], we infer

$$\begin{aligned}\sum_{y'\in\mathcal{Y}:y'\neq y} p(y')\log\frac{p(y')}{\hat{p}(y')} &\geq \left(\sum_{y'\in\mathcal{Y}:y'\neq y} p(y')\right)\log\frac{\sum_{y'\in\mathcal{Y}:y'\neq y} p(y')}{\sum_{y'\in\mathcal{Y}:y'\neq y} \hat{p}(y')} \\ &= (1-p(y))\log\frac{1-p(y)}{1-\hat{p}(y)} \ .\end{aligned}$$

Due to $\hat{p}(y) < (1-\alpha)$ and $p(y) \geq (1-\alpha)$, we get

$$\begin{aligned}& p(y)\log\frac{p(y)}{\hat{p}(y)} + (1-p(y))\log\frac{1-p(y)}{1-\hat{p}(y)} - (1-\alpha)\log\frac{1-\alpha}{\hat{p}(y)} - \alpha\log\frac{\alpha}{1-\hat{p}(y)} \\
= \ & p(y)\log p(y) - p(y)\log\hat{p}(y) + (1-p(y))\log(1-p(y)) - (1-p(y))\log(1-\hat{p}(y)) \\
& \qquad - (1-\alpha)\log(1-\alpha) + (1-\alpha)\log\hat{p}(y) - \alpha\log\alpha + \alpha\log(1-\hat{p}(y)) \\
= \ & p(y)\log p(y) + (1-p(y))\log(1-p(y)) - (1-\alpha)\log(1-\alpha) - \alpha\log\alpha \\
& \qquad + \underbrace{[(1-\alpha)-p(y)]}_{\leq 0}\log\underbrace{\hat{p}(y)}_{<(1-\alpha)} + \underbrace{[\alpha-(1-p(y))]}_{\geq 0}\log\underbrace{(1-\hat{p}(y))}_{>\alpha} \\
> \ & p(y)\log p(y) + (1-p(y))\log(1-p(y)) - (1-\alpha)\log(1-\alpha) - \alpha\log\alpha \\
& \qquad + [(1-\alpha)-p(y)]\log(1-\alpha) + [\alpha-(1-p(y))]\log\alpha \\
= \ & p(y)\log p(y) + (1-p(y))\log(1-p(y)) \\
& \qquad + [(1-\alpha)-p(y)-(1-\alpha)]\log(1-\alpha) + [\alpha-(1-p(y))-\alpha]\log\alpha \\
= \ & p(y)\log p(y) + (1-p(y))\log(1-p(y)) - p(y)\log(1-\alpha) - (1-p(y))\log\alpha\end{aligned}$$

With Gibb's inequality[2], we can infer

$$\begin{aligned}&[p(y)\log p(y) + (1-p(y))\log(1-p(y))] \\ &\qquad\qquad - [p(y)\log(1-\alpha) + (1-p(y))\log\alpha] \geq 0 \ ,\end{aligned}$$

which contradicts the assumption. □

[1] For $a_1,\ldots,a_n,b_1,\ldots,b_n \in \mathbb{R}_+$, the log sum inequality states that $\sum_{i\in[n]} a_i\log\frac{a_i}{b_i} \geq a\log\frac{a}{b}$, whereby $a := \sum_{i\in[n]} a_i$ and $b := \sum_{i\in[n]} b_i$.

[2] Gibb's inequality states that for two arbitrary discrete probability distributions $q_1, q_2 \in \mathbb{P}(\mathcal{S})$ over a discrete set $\mathcal{S}$, $\sum_{s\in\mathcal{S}} q_1(s)\log q_1(s) \geq \sum_{s\in\mathcal{S}} q_1(s)\log q_2(s)$ holds.

Proposition 2. *Let Q^α be defined as in Proposition 1 for the true class $y \in \mathcal{Y}$. Furthermore, let $\hat{p}$ be an arbitrary probability distribution over $\mathcal{Y}$. Then, using the Kullback-Leibler divergence D_{KL} as loss L within L^*, the loss as defined in Equation (4) is given by Equation (5).*

Proof. To prove the proposition, one can consider the two cases whether $\hat{p}$ is in Q^α or not. Let $p, q \in \mathbb{P}(\mathcal{Y})$ be any arbitrary probability distributions over the class space $\mathcal{Y}$. Then, the divergence properties of D_{KL} yield $D_{KL}(p||q) = 0$ if and only if $p = q$, as well as $D_{KL}(p||q) \geq 0$, $\forall\, p, q \in \mathbb{P}(\mathcal{Y})$.

For the first case, namely $\hat{p} \in Q^\alpha$, the first property of D_{KL} delivers a zero loss for $\hat{p}$ taken as target within Q^α. The latter case is an implication of Proposition 1. Hence, $L^*(Q^\alpha, \hat{p}) = D_{KL}(p^r||\hat{p})$ for $\hat{p} \notin Q^\alpha$. □

Proposition 3. *Given the fixed target set Q^α for the true class $y \in \mathcal{Y}$ as defined before, $L^*(Q^\alpha, \hat{p})$ is convex with regard to its second parameter $\hat{p}$, i.e.,*

$$\begin{aligned}&\forall p_1, p_2 \in \mathbb{P}(\mathcal{Y}), \forall \lambda \in [0,1] : \\ &\qquad L^*(Q^\alpha, \lambda p_1 + (1-\lambda) p_2) \leq \lambda L^*(Q^\alpha, p_1) + (1-\lambda) L^*(Q^\alpha, p_2) \ .\end{aligned}$$

Proof. Let p_1, p_2 and λ be fixed, and let $\tilde{p} = \lambda p_1 + (1-\lambda) p_2$ denote the considered combination of p_1 and p_2. Then, in order to prove the statement, one can distinguish three cases:

Case 1: $p_1, p_2 \in Q^\alpha$. Due to $\lambda \in [0,1]$, we have $\tilde{p} \in Q^\alpha$ and thus, per definition, $L^*(Q^\alpha, \tilde{p}) = 0$. With the non-negativity of the Kullback-Leibler divergence, we can infer $L^*(Q^\alpha, \cdot) \geq 0$, which implies the convexity.

Case 2: $p_1, p_2 \notin Q^\alpha$. As a consequence, $\tilde{p} \notin Q^\alpha$ holds. With the application of the log sum inequality, we can follow:

$$\begin{aligned}
L^*(Q^\alpha, \tilde{p}) =&\ (1-\alpha) \log \frac{1-\alpha}{\tilde{p}(y)} + \alpha \log \frac{\alpha}{1-\tilde{p}(y)} \\
=&\ [\lambda(1-\alpha) + (1-\lambda)(1-\alpha)] \log \frac{\lambda(1-\alpha) + (1-\lambda)(1-\alpha)}{\lambda p_1(y) + (1-\lambda) p_2(y)} \\
&\quad + [\lambda\alpha + (1-\lambda)\alpha] \log \frac{\lambda\alpha + (1-\lambda)\alpha}{\lambda(1-p_1(y)) + (1-\lambda)(1-p_2(y))} \\
\leq&\ \lambda(1-\alpha) \log \frac{\lambda(1-\alpha)}{\lambda p_1(y)} + (1-\lambda)(1-\alpha) \log \frac{(1-\lambda)(1-\alpha)}{(1-\lambda) p_2(y)} \\
&\quad + \lambda\alpha \log \frac{\lambda\alpha}{\lambda(1-p_1(y))} + (1-\lambda)\alpha \log \frac{(1-\lambda)\alpha}{(1-\lambda)(1-p_2(y))} \\
=&\ \lambda \left[(1-\alpha) \log \frac{1-\alpha}{p_1(y)} + \alpha \log \frac{\alpha}{1-p_1(y)} \right] \\
&\quad + (1-\lambda) \left[(1-\alpha) \log \frac{1-\alpha}{p_2(y)} + \alpha \log \frac{\alpha}{1-p_2(y)} \right] \\
=&\ \lambda L^*(Q^\alpha, p_1) + (1-\lambda) L^*(Q^\alpha, p_2)
\end{aligned}$$

Case 3: Either $p_1 \in Q^\alpha$ and $p_2 \notin Q^\alpha$ or vice versa (the proof is analogous). Obviously, $\tilde{p}$ can be an element of Q^α or not.

For the first case, namely that $\tilde{p} \in Q^\alpha$, we have $L^*(Q^\alpha, \tilde{p}) = 0$ and

$$\lambda L^*(Q^\alpha, p_1) + (1 - \lambda) L^*(Q^\alpha, p_2) = (1 - \lambda) L^*(Q^\alpha, p_2) \geq 0 \ \ .$$

For the latter case, we know that $L^*(Q^\alpha, \tilde{p})$ simplifies to the KL divergence D_{KL} on the two cases whether the class is y or any other class, whereby $D_{KL}(q||\cdot)$ is known to be convex for a fixed distribution q and non-negative. Then, with $1 - \alpha$ being the lower bound probability for y of distributions in Q^α, $L^*(Q^\alpha, \tilde{p})$ converges to 0 for $\tilde{p}(y) \rightarrow 1 - \alpha$.

Taking both cases together, $L^*(Q^\alpha, \cdot)$ comes down to the (convex) KL divergence for distributions $\tilde{p} \notin Q^\alpha$ up to the boundary of $\tilde{p}(y) = 1 - \alpha$, where the loss then degenerates to 0 in a continuous manner. As a consequence, the loss is convex for any $\tilde{p}$ in this case. □

A.2 Experimental Details

In the following, a more comprehensive overview over the experimental setting and chosen hyperparameters within our empirical studies is presented. In all experiments, SGD was used as optimizer with a Nesterov momentum of 0.9, while the batch size was set to 64. The experimental runs were conducted on 10 NVIDIA RTX 2080 Ti and 20 NVIDIA GTX 1080 Ti.

A.2.1 Simple Dense Architecture

For MNIST and Fashion-MNIST, a simple neural network with two hidden layers, each consisting of 1024 ReLu-activated neurons, was used. In the case of training on MNIST, 25 epochs were used, while the model training on Fashion-MNIST lasted for 50 epochs. Here, an initial learning rate of 0.05 has been used for both datasets, which was multiplied by $\sqrt{0.1}$ after 30 epochs for Fashion-MNIST. No further regularization was applied.

A.2.2 ResNet

To train models using the ResNet-56 (V2) architecture, our implementation is based on the publicly available code in [3]. Due to the model size, we augmented the training data as described in Subsection A.2.5. For CIFAR-10, we trained for 200 epochs with an initial learning rate of 0.1 multiplied by $\sqrt{0.1}$ after 50, 100 and 150 epochs. For CIFAR-100, 300 epochs were trained with the same initial learning rate multiplied by $\sqrt{0.1}$ after 75, 150 and 225 epochs. For this architecture, no additional Dropout layers were used within our experiments, while BatchNormalization followed each convolutional layer of the model.

A.2.3 VGG

For VGG16, we adopted the implementation as provided in [4], where some minor adaptions in order to be able to derive reasonable models for CIFAR-10 and

[3] `https://keras.io/examples/cifar10_resnet/`

[4] `https://keras.io/api/applications/vgg/`

CIFAR-100 were made. Due to the high amount of parameters, we were required to add BatchNormalization and Dropout (ranging from $p = 0.3$ to $p = 0.5$ for increasing depth) to the architecture. Furthermore, we used augmented data as described in Subsection A.2.5. For all losses, we used the same techniques and parameters. The initial learning rate was set to 0.01 in all experiments. To induce models on CIFAR-10, we trained for 200 epochs, whereby we divided the learning rate by 10 after 50, 100 and 150 epochs. In the case of CIFAR-100, our model is trained for 300 epochs with the learning rate divided by 10 after 150 and 225 epochs.

A.2.4 DenseNet

Within our experiments, we used the more efficient variant DenseNet-BC with a depth of 100 and a growth rate of 12 based on the original implementation as provided in [5]. For both CIFAR-10 and CIFAR-100, we used the same augmentation scheme as before, while we did not use Dropout in our runs. In both dataset cases, we used an initial learning rate of 0.1. While we trained on CIFAR-10 for 200 epochs with dividing the learning rate by 10 after 50, 100 and 150 epochs, we used 300 epochs for CIFAR-100 with the same divisor after 150 and 225 epochs.

A.2.5 Preprocessing and Data Augmentation

For preprocessing, we used a rather simple approach: Each image pixel is divided by 255 to get values in $[0, 1]$, followed by subtracting the pixel means of the training data. Furthermore, as described for the individual architectures, data augmentation is used for some models as an additional regularization mean. When applied, we used the implementation of [6], where we flipped horizontally and shifted in width and height (each with a fraction of 0.1).

[5] https://github.com/liuzhuang13/DenseNet

[6] https://www.tensorflow.org/api_docs/python/tf/keras/preprocessing/image/ImageDataGenerator

A.3 Barycentric Loss Visualization

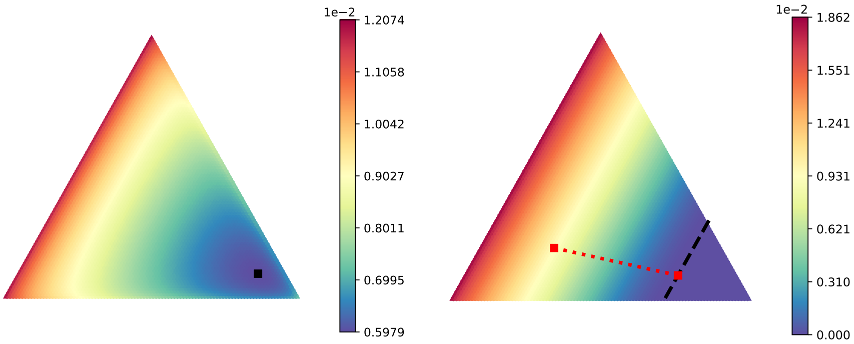


Figure 1: Barycentric visualizations of label smoothing (left plot) and L^* based on the Kullback-Leibler divergence (right plot) for three classes with $\alpha = 0.25$: The lower right corner represents the degenerate (one-point) distribution for the true target. In the case of label smoothing, the black point represents the smoothed target. For L^*, the border of the induced set Q^α (cf. Equation (3)) is sketched by the black dashed line. The red points indicate an arbitrary prediction outside Q^α (left point), which is projected onto the set boundary according to its distribution (right point).

B

Appendix to Credal Self-Supervised Learning

A Credal Self-Supervised Learning: Supplementary Material

A.1 Algorithmic Description of CSSL

Algorithm 1 provides the pseudo-code of the batch-wise loss calculation in CSSL.

Algorithm 1 CSSL with adaptive precisiation α

```
Require: Batch of labeled instances with degenerate ground truth distributions $\mathcal{B}_l = \{(\boldsymbol{x}_i, p_i)\}_{i=1}^{B} \in (\mathcal{X} \times \mathcal{Y})^B$, unlabeled batch ratio $\mu$, batch $\mathcal{B}_u = \{\boldsymbol{x}_i\}_{i=1}^{\mu B}$ of unlabeled instances, unlabeled loss weight $\lambda_u$, model $\hat{p} : \mathcal{X} \longrightarrow \mathbb{P}(\mathcal{Y})$, strong and weak augmentation functions $\mathcal{A}_s, \mathcal{A}_w : \mathcal{X} \longrightarrow \mathcal{X}$, class prior $\tilde{p} \in \mathbb{P}(\mathcal{Y})$, averaged model predictions $\bar{p} \in \mathbb{P}(\mathcal{Y})$
$\mathcal{L}_l = \frac{1}{B} \sum_{(\boldsymbol{x},p) \in \mathcal{B}_l} H(p, \hat{p}(\mathcal{A}_w(\boldsymbol{x})))$
Initialize pseudo-labeled batch $\mathcal{U} = \emptyset$
for all $\boldsymbol{x} \in \mathcal{B}_u$ do
    Derive pseudo label $q$ from $\hat{p}(\mathcal{A}_w(\boldsymbol{x}))$, $\tilde{p}$ and $\bar{p}$ acc. to Eq. (6)
    Determine reference class $y := \operatorname{argmax}_{y' \in \mathcal{Y}} q(y')$
    $\alpha = 1 - q(y) / \sum_{y' \in \mathcal{Y}} q(y')$
    Construct target set $Q_y^\alpha$ as in Eq. (2)
    $\mathcal{U} = \mathcal{U} \cup \{(\boldsymbol{x}, Q_y^\alpha)\}$
end for
$\mathcal{L}_u = \frac{1}{\mu B} \sum_{(\boldsymbol{x}, Q_y^\alpha) \in \mathcal{U}} \mathcal{L}^*(Q_y^\alpha, \hat{p}(\mathcal{A}_s(\boldsymbol{x})))$
return $\mathcal{L}_l + \lambda_u \mathcal{L}_u$
```

A.2 Algorithmic Description of LSMatch

In Algorithm 2, we provide details on the label smoothing variant of FixMatch as investigated in the experiments, which we call *LSMatch*.

Algorithm 2 LSMatch with adaptive distribution mass α

```
Require: Batch of labeled instances with degenerate ground truth distributions $\mathcal{B}_l = \{(\boldsymbol{x}_i, p_i)\}_{i=1}^{B} \in (\mathcal{X} \times \mathcal{Y})^B$, unlabeled batch ratio $\mu$, batch $\mathcal{B}_u = \{\boldsymbol{x}_i\}_{i=1}^{\mu B}$ of unlabeled instances, unlabeled loss weight $\lambda_u$, model $\hat{p} : \mathcal{X} \longrightarrow \mathbb{P}(\mathcal{Y})$, strong and weak augmentation functions $\mathcal{A}_s, \mathcal{A}_w : \mathcal{X} \longrightarrow \mathcal{X}$, class prior $\tilde{p} \in \mathbb{P}(\mathcal{Y})$, averaged model predictions $\bar{p} \in \mathbb{P}(\mathcal{Y})$
$\mathcal{L}_l = \frac{1}{B} \sum_{(\boldsymbol{x},p) \in \mathcal{B}_l} H(p, \hat{p}(\mathcal{A}_w(\boldsymbol{x})))$
Initialize pseudo-labeled batch $\mathcal{U} = \emptyset$
for all $\boldsymbol{x} \in \mathcal{B}_u$ do
    Derive pseudo label $q$ from $\hat{p}(\mathcal{A}_w(\boldsymbol{x}))$, $\tilde{p}$ and $\bar{p}$ acc. to Eq. (6)
    Determine reference class $y := \operatorname{argmax}_{y' \in \mathcal{Y}} q(y')$
    $\alpha = 1 - q(y) / \sum_{y' \in \mathcal{Y}} q(y')$
    Construct smoothed target $q'$ with $q'(y) = 1 - \frac{(|\mathcal{Y}|-1) \cdot \alpha}{|\mathcal{Y}|}$ and $q'(y') = \frac{\alpha}{|\mathcal{Y}|}$ for $y' \neq y$
    $\mathcal{U} = \mathcal{U} \cup \{(\boldsymbol{x}, q')\}$
end for
$\mathcal{L}_u = \frac{1}{\mu B} \sum_{(\boldsymbol{x}, q') \in \mathcal{U}} H(q', \hat{p}(\mathcal{A}_s(\boldsymbol{x})))$
return $\mathcal{L}_l + \lambda_u \mathcal{L}_u$
```

A.3 Evaluation Details

A.3.1 Experimental Settings

As discussed in the paper, we follow the experimental setup as described in [6]. For a fair comparison, we keep the hyperparameters the same as used within the experiments for FixMatch, which we provide in Table 1. Note that the parameter τ does not apply to CSSL nor LSMatch, as these approaches do not rely on any confidence thresholding.

Table 1: Hyperparameters as being used within the experiments for CSSL, FixMatch, LSMatch and all other method derivates (if not stated otherwise).

Symbol	Description	Used value(s)
λ_u	Unlabeled loss weight	1
μ	Multiplicity of unlab. over lab. insts.	7
B	Labeled batch size	64
η	Initial learning rate	0.03
β	SGD momentum	0.9
Nesterov	Indicator for Nesterov SGD variant	True
wd	Weight decay	0.001 (CIFAR-100), 0.0005 (other data sets)
τ	Confidence threshold	0.95
K	Training steps	2^{20}

For CTAugment (and later RandAugment as considered in Section A.4.2), we use the same operations and parameter ranges as in reported in [6] for comparability reasons. In case of CTAugment, we keep the bin weight threshold at 0.8 and use an exponential decay of 0.99 for the weight updates. In the latter case, we also follow a purely random sampling, that slightly differs from the original formulation in [2]. We refer to [1, 6] for a more comprehensive overview over the methods and their parameters.

A.3.2 Technical Infrastructure

To put our approach into practice, we re-used the original FixMatch code base[1] provided by the authors for the already available baselines, models, augmentation strategies and the evaluation, and extended it by our implementations. To this end, we leverage TensorFlow[2] as a recent deep learning framework, whereas the image augmentation functions are provided by Pillow[3]. To execute the runs, we used several Nvidia Titan RTX, Nvidia Tesla V100, Nvidia RTX 2080 Ti and Nvidia GTX 1080 Ti accelerators in modern cluster environments. The code related to our work is available at `https://github.com/julilien/CSSL`.

A.3.3 Efficiency Experiments: Learning Curves

Figure 1 shows the learning curves of the runs considered in the efficiency study in Section 4.3 (averaged over 5 seeds). As can be seen, both CSSL and LSMatch improve the learning efficiency in label-scarce settings by not relying on any form of confidence thresholding. Although LSMatch turns out to converge slightly faster when observing 40 labels from SVHN, the eventual generalization performance of CSSL is clearly superior. For higher amounts of labels, the results are almost indistinguishable.

A.4 Additional Experiments

A.4.1 Simple Synthetic Example

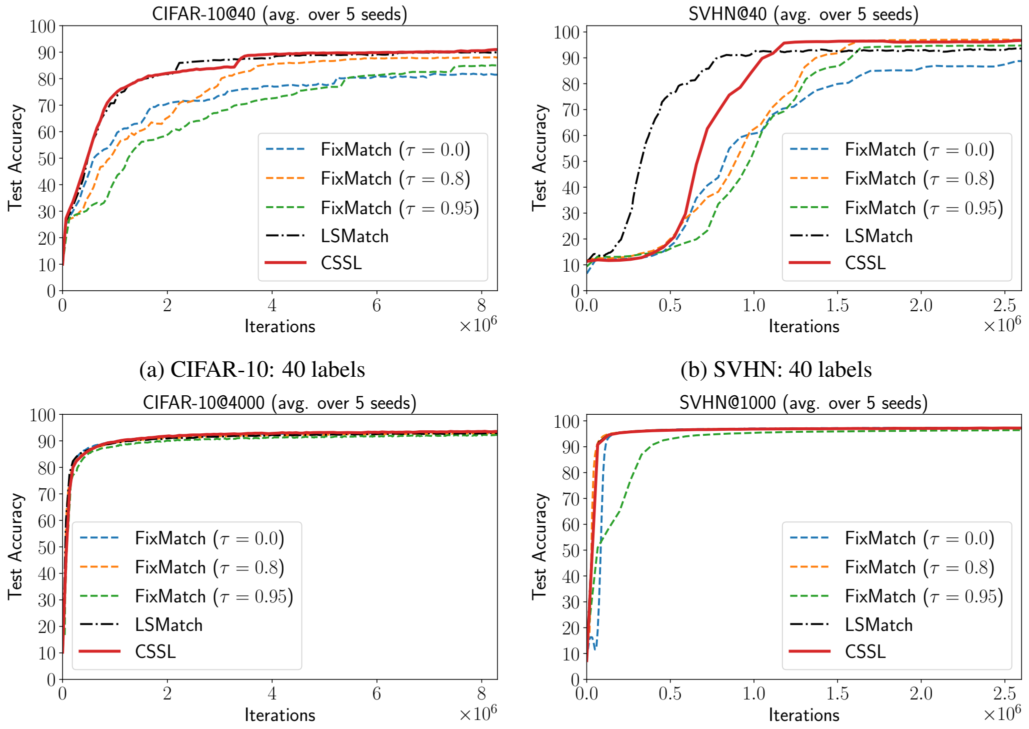
To illustrate the disambiguation principle underlying the idea of credal self-supervised learning, we consider a synthetic (semi-supervised) binary classification problem in a one-dimensional feature space. To this end, we sample 25 labeled and 500 unlabeled instances uniformly from the unit interval. As ground-truth, we define the true probability of the positive class by a sigmoidal shaped function.

Provided this data, we train a simple multi-layer perceptron with a single hidden layer consisting of 100 neurons activated by a sigmoid function. As output layer, we use a softmax-activated dense layer with two neurons. To make use of the unlabeled data, we employ self-training with either hard, soft or credal labels for 100 iterations each. In all three cases, we do not apply any form of distribution alignment or confidence thresholding. We use SGD as optimizer with a learning rate of 0.5 (no

[1] The code is publicly available at `https://github.com/google-research/fixmatch` under the Apache-2.0 License.

[2] `https://www.tensorflow.org/`, Apache-2.0 License

[3] `https://python-pillow.org/`, HPND License



(a) CIFAR-10: 40 labels

(b) SVHN: 40 labels

(c) CIFAR-10: 4000 labels

(d) SVHN: 1000 labels

Figure 1: Averaged learning curves for different dataset configurations.

momentum) for all methods. We repeat each model training (on the same data) 5 times with different seeds.

Fig. 2 shows the labeled and unlabeled data points[4] (red crosses and green circles respectively) and their ground-truth positive class probability (red dashed line), as well as the soft and hard labeling baselines (orange and green lines). The model trained with credal self-supervised learning is indicated by the blue line.

In this setting, self-training of a simple neural network with deterministic labeling leads to a flat (instead of sigmoidal) function most of the time, because the learner tends to go with the majority in the labeled training data. With probabilistic labels, the results become a bit better: the learned functions tend to be increasing but still deviates a lot from the ground-truth sigmoid. Our credal approach yields the best result, being closer to the sigmoid (albeit not matching it perfectly). These results are perfectly in agreement with our intuition and motivation of our approach: Self-labeling examples in an overly “aggressive” (and over-confident) way may lead to self-confirmation and a bias in the learning process.

A.4.2 Augmentation Ablation Study

In an additional experiment, we perform an ablation study to measure the impact of the augmentation policy on the generalization performance and the network calibration. Here, we compare CSSL with RandAugment and CTAugment as strong augmentation policies. We further distinguish between RandAugment either with or without *cutout* [3], which is a technique to randomly mask out parts of the image. Note that cutout is included in the set of operations we use for CTAugment and was found to be required for strong generalization performance within the FixMatch framework (cf. [6]). Here, we report results on CIFAR-10 for 40 and 4000 labels, in both cases trained for 2^{19} steps. We use the same hyperparameters as enlisted in Tab. 1 and report the averaged results for 3 seeds.

[4] Note that the unlabeled instances have a probability of 0.5 assigned only for the sake of visualizability, their class distribution also follows the sigmoidal ground-truth.

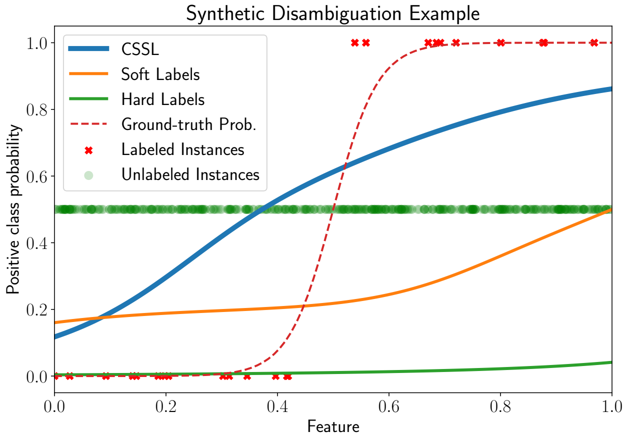


Figure 2: Illustration of CSSL in contrast to conventional probabilistic self-labeling on a simple data-generating process (averaged over 5 seeds).

As can be seen in Table 2, our evaluation confirms the observation in [6] that cutout is a required operation to achieve competitive performance. While RandAugment with cutout as augmentation policy does not show any notable difference compared to CTAugment, employing the former policy without cutout leads to drastically inferior generalization performance and network calibration.

Table 2: Averaged misclassification rates and standard deviations (3 seeds) for various augmentation policies used in CSSL on CIFAR-10 (**bold** font indicates the best performing method and those within a range of two standard deviations from the best method).

Augmentation	40 labels		4000 labels	
	Err.	ECE	Err.	ECE
CTAugment	**6.92** ±0.34	**0.035** ±0.005	**5.11** ±0.64	**0.028** ±0.001
RandAugment (w/o cutout)	28.81 ±19.11	0.131 ±0.102	7.42 ±0.76	0.039 ±0.001
RandAugment (w/ cutout)	**6.74** ±0.18	**0.031** ±0.002	**5.13** ±0.66	**0.029** ±0.000

A.5 Barely Supervised Experiments

We also consider the scenario of *barely supervised learning* [6], where a learner is given only a single labeled instance per class. We train all models with 2^{19} update iterations on CIFAR-10 and SVHN, using the hyperparameters as described in Section A.3.1. We report the averaged misclassification rates on 5 different folds.

As the uncertainty with such few labels is relatively high, we have experimented with a CSSL version that lower bounds the precisiation degrees α, i.e., it defines a minimal size for the target sets. This further increases the degree of cautiousness, but, however, can be seen as an unfair advantage over the other baselines as it adds additional awareness about the highly uncertain nature of the faced problems. In our experiments, we bound the precisiation degree by $\alpha \geq 0.03$, which we determined empirically as a reasonable choice for the two datasets.

Table 3 shows the results. As can be seen, CSSL is able to reduce the error rate compared to conventional hard pseudo-labels in both cases. Moreover, it reduces the intra-dataset variance, which reflects a higher stability of the learning process. Bounding the target set sizes by $\alpha \geq 0.03$ slightly improves the performance on CIFAR-10, but does not show any advantage on SVHN. On the contrary, label smoothing shows an unstable learning behavior in some cases, leading to poor generalization performances for some folds on both data sets. These cases suggest that LSMatch suffers particularly from extremely scarce labeling.

Table 3: Averaged misclassification rates and standard deviations (5 seeds) for the barely supervised experiments with 10 labels (**bold** font indicates the single best performing method).

Model	CIFAR-10	SVHN
FixMatch	31.74 ±17.77	30.57 ±23.82
LSMatch	42.14 ±25.33	44.29 ±27.40
CSSL	26.90 ±13.57	**9.69** ±4.63
CSSL ($\alpha \geq 0.03$)	**26.10** ±8.02	16.63 ±13.21

A.6 Uncertainty-Based Matching

As the set-valued target modeling in CSSL leads to a form of uncertainty-awareness, one may think of extending classical probabilistic pseudo-labels by additional means measuring the model uncertainty. To this end, UPS [5] augments classical confidence thresholded pseudo-labeling by an additional uncertainty sampling technique (e.g., using MC-Dropout [4]), which is employed in the mechanism to filter out unreliable pseudo-labels. More precisely, it assumes an uncertainty threshold κ besides the confidence threshold τ to calculate the loss on unlabeled instances by

$$\mathcal{L}_u := \frac{1}{\mu B} \sum_{(\boldsymbol{x},q) \in \mathcal{B}_u} \mathbb{I}_{\max \hat{p}(\boldsymbol{x}) \geq \tau} \mathbb{I}_{\max u(\hat{p}(\boldsymbol{x})) \leq \kappa} H(q, \hat{p}(\boldsymbol{x})) \enspace ,$$

where the sum is taken over all unlabeled instances in $\mathcal{B}_u$ with individual pseudo labels q constructed from $\hat{p}(\boldsymbol{x})$, and $u(\cdot)$ is the sampled uncertainty. Note that UPS in its original formulation further uses negative labels, which we omit here for a more the sake of a fair comparison.

We employ the uncertainty-based filtering technique within the framework of FixMatch to compare it with our form of uncertainty-awareness. In the following, we call this variant *UPSMatch*. To induce reasonable thresholds τ and κ, we empirically optimize the hyperparameters τ and κ in the spaces $\{0.7, 0.95\}$ and $\{0.1, 0.25, 0.5\}$ respectively on a separate validation set. For MC-Dropout, we use 8 sampling iterations (as opposed to 10 in the original approach) due to computational resource concerns, whereby the drop rate for Dropout is set to 0.3 as also used in [5]. Moreover, we use hard pseudo-labels as in UPS and FixMatch.

Tab. 4 shows the results for UPSMatch and baselines for 5 different seeds after 2^{20} training steps. On CIFAR-10, UPSMatch improves over FixMatch for 40 labels in terms of generalization performance and network calibration. This is reasonable as UPSMatch implements a more cautious learning behavior by augmenting the pseudo-label selection criterion and reduces the risk of confirmation biases. Similarly, UPSMatch outperforms FixMatch for 40 labels on SVHN, whereas it provides both worse generalization performance and better calibration in the other cases. Nevertheless, CSSL turns out to be superior compared to both selective methodologies. However, let us emphasize that a more thorough investigation of the interaction between uncertainty-based sampling and consistency regularization as employed in FixMatch needs to be performed, as well as a more sophisticated hyperparameter optimization.

Table 4: Averaged misclassification rates and expected calibration errors (ECE, 15 bins) for 5 seeds (**bold** font indicates the best performing method and those within a range of two standard deviations from the best method).

	CIFAR-10				SVHN			
	40 lab.		4000 lab.		40 lab.		1000 lab.	
	Err.	ECE	Err.	ECE	Err.	ECE	Err.	ECE
FixMatch	11.39 ±3.35	0.087 ±0.051	**4.31** ±0.15	0.030 ±0.002	**7.65** ±7.65	**0.040** ±0.044	2.28 ±0.19	0.010 ±0.002
CSSL	**6.50** ±0.90	**0.032** ±0.005	**4.43** ±0.10	**0.023** ±0.001	**3.67** ±2.36	**0.022** ±0.029	**1.99** ±0.13	**0.007** ±0.001
UPSMatch	10.48 ±2.11	0.058 ±0.010	4.92 ±0.39	0.027 ±0.003	**3.81** ±1.99	**0.025** ±0.027	2.71 ±0.47	**0.008** ±0.001

References

[1] David Berthelot, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. ReMixMatch: Semi-supervised learning with distribution matching and augmenta-

tion anchoring. In *Proc. of the 8th International Conference on Learning Representations, ICLR, Addis Ababa, Ethiopia, April 26-30*. OpenReview.net, 2020.

[2] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. RandAugment: Practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

[3] Terrance Devries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. *CoRR*, abs/1708.04552, 2017.

[4] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proc. of the 33nd International Conference on Machine Learning, ICML, New York City, NY, USA, June 19-24*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1050–1059. JMLR.org, 2016.

[5] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh Singh Rawat, and Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *CoRR*, abs/2101.06329, 2021.

[6] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

C

Appendix to Conformal Credal Self-Supervised Learning

Proceedings of Machine Learning Research 204:1–18, 2023 Conformal and Probabilistic Prediction with Applications

Appendix to Conformal Credal Self-Supervised Learning

Julian Lienen[*] JULIAN.LIENEN@UPB.DE
Caglar Demir CAGLAR.DEMIR@UPB.DE
Paderborn University, Germany

Eyke Hüllermeier EYKE@LMU.DE
University of Munich (LMU), Germany

Editor: Harris Papadopoulos, Khuong An Nguyen, Henrik Boström and Lars Carlsson

Appendix A. Pseudo-Code of CCSSL

Algorithm 2 CCSSL with consistency regularization

```
Require: Batch of labeled instances with degenerate ground truth distributions $\mathcal{B}_l = \{(\boldsymbol{x}_i, p_i)\}_{i=1}^B \in (\mathcal{X} \times \mathcal{Y})^B$, unlabeled batch ratio $\mu$, batch $\mathcal{B}_u = \{\boldsymbol{x}_i\}_{i=1}^{\mu B}$ of unlabeled instances, unlabeled loss weight $\lambda_u$, model $\hat{p} : \mathcal{X} \longrightarrow \mathbb{P}(\mathcal{Y})$, strong and weak augmentation functions $\mathcal{A}_s, \mathcal{A}_w : \mathcal{X} \longrightarrow \mathcal{X}$, calibration data $\mathcal{D}_{\text{calib}} \subset (\mathcal{X} \times \mathcal{Y})^L$, inductive conformal prediction procedure $ICP : (\mathcal{X} \times \mathcal{Y})^L \times \mathbb{P}(\mathcal{Y}) \longrightarrow (\mathcal{Y} \longrightarrow [0,1])$
$\mathcal{L}_l = \frac{1}{B} \sum_{(\boldsymbol{x},p) \in \mathcal{B}_l} D_{KL}(p \,||\, \hat{p}(\mathcal{A}_w(\boldsymbol{x})))$
Initialize pseudo-labeled batch $\mathcal{U} = \emptyset$
for all $\boldsymbol{x} \in \mathcal{B}_u$ do
    Derive possibility distribution $\pi = ICP(\mathcal{D}_{\text{calib}}, \hat{p}(\mathcal{A}_w(\boldsymbol{x})))$
    Apply normalization to $\pi$ such that $\max_{y \in \mathcal{Y}} \pi(y) = 1$ (e.g., as in Eq. (9))
    Construct credal set $\mathcal{Q}_\pi$ as in Eq. (1)
    $\mathcal{U} = \mathcal{U} \cup \{(\boldsymbol{x}, Q_\pi)\}$
end for
$\mathcal{L}_u = \frac{1}{\mu B} \sum_{(\boldsymbol{x}, Q_\pi) \in \mathcal{U}} \mathcal{L}^*(Q_\pi, \hat{p}(\mathcal{A}_s(\boldsymbol{x})))$ {Can be solved via generalized credal learning procedure (Alg. 1 in main paper)}
return $\mathcal{L}_l + \lambda_u \mathcal{L}_u$
```

Appendix B. Experimental Details

B.1. Settings

To conduct the experiments as presented in the paper, we followed the basic semi-supervised learning evaluation scheme as in (Sohn et al., 2020; Lienen and Hüllermeier, 2021). However, as opposed to previous evaluations, we reduce the number of iterations to 2^{18} with a batch size of 32, which allows for a proper hyperparamter optimization of all methods. To this end, we employ a Bayesian optimization[1] with 20 runs for each combination of dataset and

[*] Corresponding author

[1] We used the Bayesian optimization implementation as offered by *Weights & Biases* (Biewald, 2020) with default parameters.

number of labels on a separate validation split. Moreover, we use Hyperband (Li et al., 2017) with $\eta = 3$ and 20 minimum epochs (that is, iterating over all unlabeled instances once) for early stopping. Due to the computational complexity of this procedure, we determined the best hyperparameter on a fixed seed and applied those parameters to all repetitions with different seeds for the same dataset and number of labels combination. Albeit not being ideal, such routine still improves fairness compared to previous evaluations which do not apply the same hyperparameter tuning procedure to all regarded baselines.

Table 3: Hyperparameter search spaces considered in the optimization.

Method	Parameter	Values
All	Initial learning rate	$\{0.005, 0.01, 0.03, 0.05, 0.1\}$
	Unlabeled batch multiplicity μ	$\{3, 7\}$
	Weight decay	$\{0.0005, 0.0001\}$
	Unlabeled loss weight λ_u	$\{1\}$
FixMatch (DA), UDA	Confidence threshold τ	$\{0.7, 0.8, 0.9, 0.95\}$
	Temperature	$\{0.5, 1\}$
FlexMatch	Cutoff threshold	$\{0.8, 0.9, 0.95\}$
	Threshold warmup	$\{\text{True}, \text{False}\}$
CCSSL-diff	Calibration split	$\{0.1, 0.25, 0.5\}$
CCSSL-prop	Calibration split	$\{0.1, 0.25, 0.5\}$
	Non-conf. sensitivity γ	$\{0.01, 0.1, 1\}$

Tab. 3 shows the considered parameter spaces. To train the models, we use SGD with a Nesterov momentum of 0.9. We further employ cosine annealing as learning rate schedule (Loshchilov and Hutter, 2017). Moreover, we apply exponential moving averaging with a fixed decay of 0.999 to the weights.

B.2. Code and Environment

Our official implementation is publicly available.[2] Therein, we implemented all methods using PyTorch[3], where we reused the official implementations if available. We proceeded from a popular FixMatch re-implementation in PyTorch[4] for the image classification experiments, which we carefully checked for any differences to the original repository, and embedded all other baselines into it. To conduct the experiments, we used several Nvidia A100 GPUs in a modern high performance cluster environment.

[2] https://github.com/julilien/C2S2L

[3] https://pytorch.org/, BSD-style license

[4] https://github.com/kekmodel/FixMatch-pytorch, MIT license

Appendix C. Additional Results

C.1. Predictor Calibration

For completeness, we present the quality of the prediction probability distributions in the large-scale image classification experiments with respect to their expected calibration errors in Tab. 4. Both CCSSL variants demonstrate favorable calibration properties, whereas CCSSL-prop often outperforms all other methods.

Table 4: Averaged ECE scores with 15 bins over 3 seeds for different numbers of labels. **Bold** entries indicate the best performing method per column. The standard deviation is a factor of $1e^{-2}$.

	CIFAR-10			CIFAR-100			SVHN			STL-10
	40 lab.	250 lab.	4000 lab.	400 lab.	2500 lab.	10000 lab.	40 lab.	250 lab.	1000 lab.	1000 lab.
UDA	0.159 ±7.9	**0.051** ±0.2	0.046 ±0.1	0.420 ±2.6	0.232 ±0.5	0.173 ±0.5	0.124 ±1.1	0.037 ±1.3	0.030 ±0.1	0.140 ±0.9
FixMatch	0.136 ±4.3	0.059 ±1.0	0.048 ±0.1	0.417 ±2.3	0.254 ±0.5	0.168 ±0.1	0.275 ±34.9	0.038 ±1.3	**0.027** ±0.1	0.128 ±0.8
FixMatch DA	0.131 ±11.1	0.057 ±0.6	0.047 ±0.1	**0.347** ±2.4	0.234 ±0.4	0.169 ±0.1	**0.101** ±3.1	0.041 ±1.5	0.029 ±0.1	0.127 ±0.8
FlexMatch	0.114 ±3.7	0.057 ±0.5	0.052 ±0.2	0.404 ±0.3	0.232 ±0.5	0.169 ±0.2	0.126 ±1.3	0.039 ±0.3	0.033 ±0.4	0.130 ±0.9
CSSL	**0.079** ±4.4	0.053 ±0.1	0.045 ±0.0	0.368 ±0.8	0.233 ±0.6	0.167 ±0.2	0.121 ±5.1	0.041 ±1.1	0.032 ±0.1	0.126 ±0.9
CCSSL-diff	0.139 ±3.2	0.058 ±0.1	0.045 ±0.1	0.352 ±1.6	**0.222** ±0.3	0.165 ±0.1	0.117 ±7.2	0.044 ±1.2	0.030 ±0.3	0.129 ±0.6
CCSSL-prop	0.101 ±2.2	0.054 ±0.2	**0.044** ±0.1	**0.347** ±2.3	0.227 ±0.2	**0.162** ±0.2	0.128 ±8.6	**0.033** ±0.1	0.028 ±0.2	**0.124** ±0.8

C.2. Learning Curves

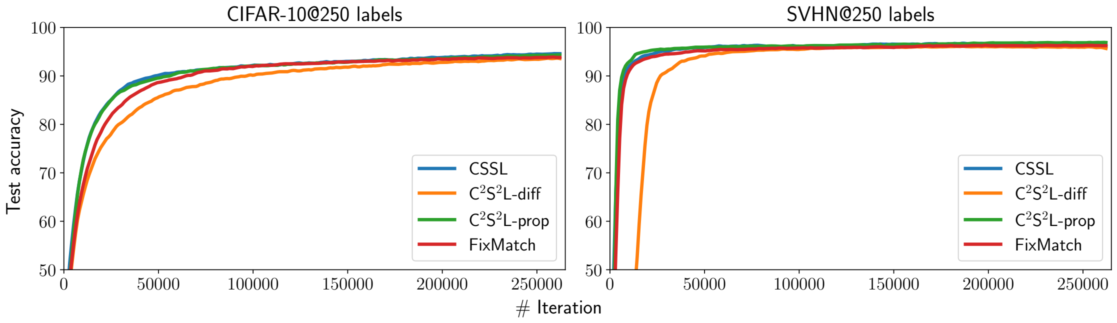


Figure 4: Test accuracies over the course of the training. The results are averaged over 3 different random seeds.

In addition, we provide the learning curves in terms of test accuracy per training iteration averaged over 3 random seeds on CIFAR-10 and SVHN with 250 labels each in Fig. 4. As can be seen, CCSSL-prop shows a similar training efficiency as CSSL, whereas CCSSL-diff realizes a more cautious learning than all other methods. This becomes particularly visible for SVHN. Here, CCSSL-prop shows an even improved training efficiency compared to CSSL. Remarkably, the CCSSL variants have less labeled supervision available as part of it is separated in form of the calibration split, which can be one reason for the cautiousness of CCSSL-diff in the first iterations. Together with the validity gains in the pseudo-label quality, especially CCSSL-prop demonstrates its effectiveness to the task of semi-supervised learning.

C.3. Mitigation of Confirmation Biases: Imbalanced Data

As discussed in the main paper, consistency regularization (CR) and the validity of conformal (credal) pseudo-labels serve as means to tackle confirmation biases. In addition to the previously shown experiments, we consider here another experimental setting aiming to isolate their individual contributions to the mitigation of confirmation biases. Namely, we look at EMNIST-ByClass (Cohen et al., 2017) consisting of 814,255 handwritten letters (both lower and upper case) and digits, constituting 62 imbalanced classes in total.[5] The class imbalance leads to an attenuation of CR as frequently occurring classes dominate underrepresented ones. More technically, the neighborhood of instances belonging to an underrepresented region may be mostly populated by instances from different classes, thereby violating the expansion assumption (Wei et al., 2021). Consequently, from an empirical risk minimization point of view, it might be more reasonable to attribute larger regions populated by unlabeled instances from the minority class to a majority classes. This leaves the former overlooked, so that a “de-noising” of the pseudo-labels is not possible anymore. Again, the validity guarantees provided by the conformal prediction framework serve as a fallback here.

For this dataset, we consider either 250 or 500 labeled instances and train for 2^{15} iterations, keeping all other experimental parameters the same as before. Adopting the reported optimal hyperparameters for CSSL in (Lienen and Hüllermeier, 2021), we used a fixed learning rate of 0.03, SGD with Nesterov momentum of 0.9 and trained a Wide ResNet-28-2 with a batch size of 32 for three different seeds. Table 5 shows the resulting generalization performances for the individual methods. Moreover, Fig. 5 presents the learning curves, which show similar but even more extreme trends as also observed in the CIFAR-100 experiments presented in Sec. 5.2 of the main paper.

Table 5: Test accuracies on EMNIST-ByClass. The presented results and their standard deviations are computed over 3 seeds.

	250 lab.	500 lab.
CSSL	49.96 ±4.08	62.74 ±2.15
CCSSL-diff	**59.22** ±3.93	67.39 ±3.74
CCSSL-prop	57.90 ±5.85	**67.62** ±4.61

[5] We refer to Fig. 2 in (Cohen et al., 2017) for a detailed overview over the class distribution.

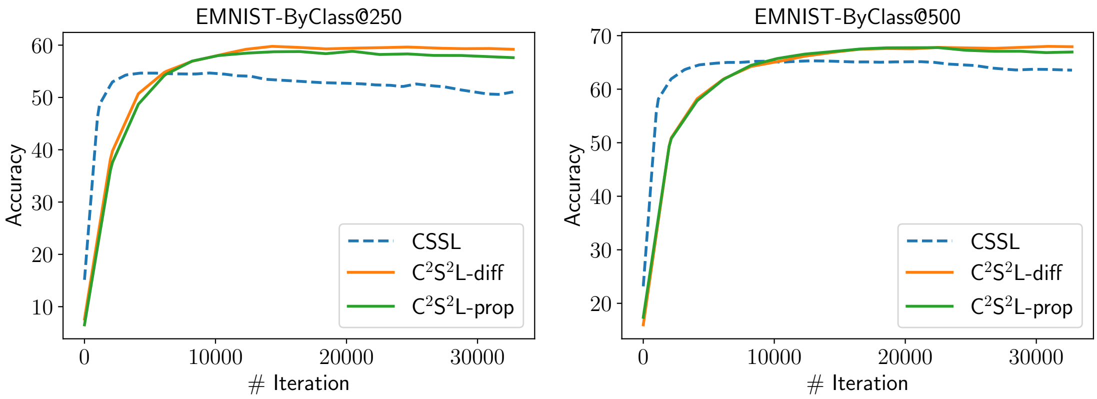


Figure 5: Averaged learning curves in terms of test accuracies over three seeds on EMNIST-ByClass with 250 and 500 labels.

C.4. Ablation Studies

In the following, we present further ablation studies to investigate properties of CCSSL more thoroughly. If not stated otherwise, we set the initial learning rate to 0.03, $\lambda_u = 1$, $\mu = 7$ and the weight decay to 0.0005. Also, we consider calibration size fractions of 0.25 by default.

Possibility Distribution Normalization Conformal Credal Pseudo-Labeling involves normalizing possibility distributions $\pi : \mathcal{Y} \longrightarrow [0,1]$ to satisfy $\max_{y \in \mathcal{Y}} \pi(y) = 1$. In Eq. (9) of the paper, we introduced a proportion-based normalization technique, to which we refer as *normalization 1*. As an alternative, one can consider the following normalization (Cella and Martin, 2021), to which we refer as *normalization 2*:

$$\pi(\hat{y}) = \begin{cases} 1 & \text{if } \hat{y} = \operatorname{argmax}_{y' \in \mathcal{Y}} \pi(y'), \\ \pi(\hat{y}) & \text{otherwise.} \end{cases} \tag{13}$$

It is easy to see that (13) leads to smaller credal sets due to $0 \leq \pi(\cdot) \leq 1$.

Table 6: Test accuracies per normalization and CCSSL variant for 250 labels each. The presented results and their standard deviations are computed over 3 seeds.

	CIFAR-10		SVHN	
	Norm. 1	Norm. 2	Norm. 1	Norm. 2
CCSSL-diff	92.38 ±1.14	92.71 ±0.81	95.93 ±1.78	95.70 ±1.81
CCSSL-prop	92.26 ±2.30	91.84 ±1.13	96.61 ±1.08	95.72 ±1.50

Tab. 6 shows the results. Although the second normalization strategy leads to smaller credal sets, it leads to inferior generalization performance for the proportion-based non-conformity measure, suggesting that an overly extreme credal set construction may be suboptimal. This supports the adequacy of the first normalization strategy as employed in CCSSL by default.

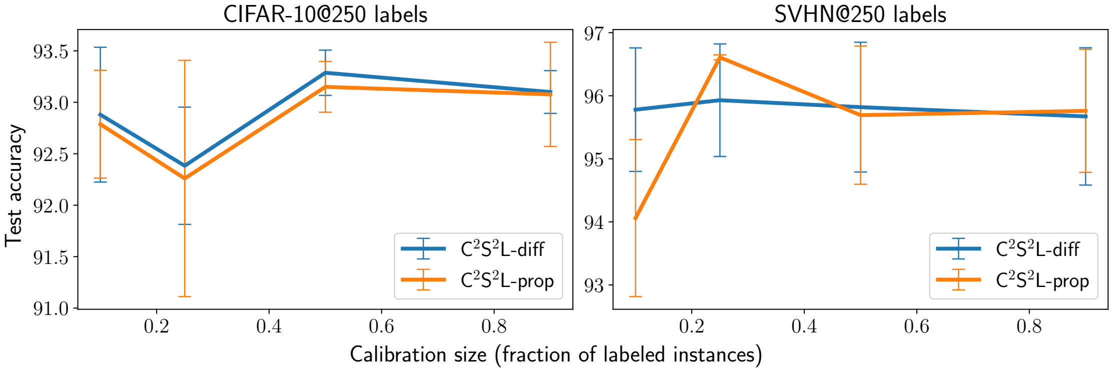


Figure 6: Test accuracy and the standard deviation per calibration size for the two variants of CCSSL. The results are averaged over 3 different random seeds.

Calibration Split Size The calibration size used to determine the number of calibration instances affects the quality of the credal sets. Here, we consider calibration split proportions in $\{0.1, 0.25, 0.5, 0.9\}$. As can be seen in Fig. 6, the differences in the performances do not vary too much. On CIFAR-10 with 250 labels, no clear trend can be observed. However, the deviations of runs with a fraction of 0.25 appear significantly higher than for the other sizes. This could be due to the sacrifice of too many labeled instances without achieving too precise pseudo-supervision. Lower and higher calibration sizes may overcome this by benefiting from either of these two extremes (higher pseudo-label quality through many labeled instances or larger calibration sets). In case of SVHN, CCSSL-prop seems to be more sensitive to the calibration size and achieves the best results with a calibration size fraction of 0.25. Here, too small calibration sizes clearly lead to unsatisfying results, demonstrating again the influence of the pseudo-label quality on the overall generalization performance.

Proportion-based Non-Conformity Sensitivity As defined in Eq. (3), the proportion-based non-conformity measure $\alpha(\mathcal{D}, (\boldsymbol{x}, y))$ involves the parameter $\gamma \geq 0$, which represents the sensitivity towards the influence of the prediction $\hat{p}_{\mathcal{D}}(\boldsymbol{x})(y)$ on the scoring for a given tuple $(\boldsymbol{x}, y)$.

In Tab. 7, we report results of CCSSL-prop for $\gamma \in \{0.01, 0.1, 0.5, 1, 10\}$. While smaller γ values lead to better results for CIFAR-10, SVHN benefits from slightly higher γ values. These results show that this parameter indeed has an effect on the overall results, i.e., it is reasonable to consider it as a hyperparameter.

Table 7: Averaged test accuracies and their standard deviations over 3 random seeds for various γ values used in CCSSL-prop. For each dataset, we considered 250 labeled examples to be given.

γ	CIFAR-10	SVHN
0.01	**93.64** ±0.33	95.60 ±2.17
0.1	93.24 ±1.16	95.91 ±1.76
0.5	93.19 ±1.02	**96.82** ±0.10
1	93.01 ±1.23	96.61 ±0.08
10	92.29 ±1.94	95.41 ±2.07

C.5. Knowledge Graph Embedding Experiments

We were interested in evaluating conformal credal self-supervised learning in the link prediction problem on knowledge graphs. Knowledge graphs represent structured collections of facts (Hogan et al., 2020), which are stored in graphs connecting entities via relations. These collections of facts have been used in a wide range of applications, including web search, question answering, and recommender systems (Nickel et al., 2015). The task of identifying missing links in knowledge graphs is referred to as *link prediction*. Knowledge graph embedding (KGE) models have been particularly successful at tackling the link prediction task, among many others (Nickel et al., 2015). In semi-supervised link prediction, only a fraction of facts is given. The task is then to “enrich” the input graph to detect relations that connect entities, which are subsequently also used in the learning of graph embeddings.

Experimental Setting In our experiments, we follow a standard training and evaluation setup commonly used in the KGE domain (Ruffinelli et al., 2020; Cao et al., 2021). We consider three multiplicative interaction-based KGE embedding models: DistMult (Yang et al., 2014), ComplEx (Trouillon et al., 2016), and QMult (Demir et al., 2021). To train the models, we model the problem as a 1vsAll classification (see (Demir et al., 2022) and (Ruffinelli et al., 2020) for more details about the 1vsAll training regime). Here, we compare conventional pseudo-labeling as described in (Lee, 2013) to CCSSL-diff. In the following, we refer to the former as PL. We do not employ consistency regularization or any other confirmation bias mitigation technique, demonstrating the flexibility of our framework. In our experiments, we used the two benchmark datasets UMLS and KINSHIP (Dettmers et al., 2018), whose characteristics are provided in Table 8.

Table 8: Overview of the datasets considered in the KGE experiments.

	# Entities	# Relations	$\lvert\mathcal{D}_{\text{train}}\rvert$	$\lvert\mathcal{D}_{\text{val}}\rvert$	$\lvert\mathcal{D}_{\text{test}}\rvert$
UMLS	136	93	10,432	1,304	1,965
KINSHIP	105	51	17,088	2,136	3,210

For each model (DistMult, ComplEx, and QMult), we applied a grid-search over the learning rates $\{0.01, 0.1, 0.001\}$, batch sizes $\{512, 1024\}$ and the number of epochs $\{5, 50\}$ to tune the parameters on a separate validation set. For CCSSL-diff, we initially divided $\mathcal{D}_{\text{train}}$ into training, calibration and unlabeled splits with 40:40:20 ratios, respectively. For the conventional pseudo-labeling procedure, we divided $\mathcal{D}_{\text{train}}$ into training, and unlabeled splits with 40:60 ratios, respectively. To ensure that each model is trained with the exact training split, we select the first 40% of all triples as training split.

As opposed to domains like image classification, where a small number of classes, for which labeled instances are provided, allows for a certain degree of interpolation, knowledge graphs involve a typically larger vocabulary of entities. By subselecting data from the knowledge graph, there is a much higher chance to miss some entities. Not observing parts of the vocabulary renders the task of learning their embeddings effectively as an unsupervised learning problem. This is why our considered training splits are relatively large. Arguably, CCSSL gets an unfair advantage here by providing (labeled) calibration data beyond the labeled training data. However, this data is excluded from being used as pseudo-labeled training data either, and can not contribute to the learned embeddings directly. It leaves large parts of the knowledge graph unconnected as it prevents to enrich that part of the graph by pseudo-labels, having an influence on the generalizability of the learned KGE model.

Link Prediction Results Table 9 reports the link prediction performance on UMLS and KINSHIP. Overall, the results suggest that incorporating CCSSL in knowledge graph embedding model leads to better generalization performance in 16 out of 18 metrics on two benchmark datasets. Also, the credal pseudo-label construction is effective as it improves the results over the PL baseline. To address our discussions about fairness in the split modeling, we conduct two more experiments with 40:10:50 and 30:20:50 split ratios to quantify the impact of the calibration signal in the link prediction task, where the PL baselines observes splits with ratios 40:60 and 50:50, respectively.

Table 9: Link prediction results on UMLS and KINSHIP. The 40:40:20 split ratio for CCSSL-diff and a 40:60 for conventional pseudo-labeling (PL). Bold entries denote best results per method, dataset and metric.

	UMLS			**KINSHIP**		
	MRR	H@1	H@3	MRR	H@1	H@3
DistMult-PL	0.229	0.128	0.243	0.262	0.160	0.284
DistMult-CCSSL	**0.246**	**0.161**	**0.265**	**0.274**	**0.170**	**0.299**
ComplEx-PL	**0.282**	0.160	**0.358**	0.333	0.226	0.382
ComplEx-CCSSL	0.253	**0.191**	0.261	**0.344**	**0.255**	**0.381**
QMult-PL	0.269	0.157	**0.309**	0.323	0.228	0.351
QMult-CCSSL	**0.294**	**0.217**	**0.309**	**0.328**	**0.238**	**0.363**

Table 10 reports the link prediction performances for the 40:10:50 (CCSSL) and 40:60 (PL) split ratio. Interestingly, the results do not vary much (only in 3 out of 18 cases). This confirms that the labeled data split has critical influence on the overall results in KGE link prediction, whereas the contribution of the self-supervised part is limited. As said before, semi-supervised learning in knowledge graph embedding is much more depending on the training data compared to image classification.

Table 10: Link prediction results on UMLS and KINSHIP. The 40:10:50 split ratio for CCSSL-diff and a 40:60 for pseudo-labelling (PL). Bold entries denote best results per method, dataset and metric.

	UMLS			**KINSHIP**		
	MRR	H@1	H@3	MRR	H@1	H@3
DistMult-PL	0.229	0.128	0.243	0.262	0.160	0.284
DistMult-CCSSL	**0.246**	**0.161**	**0.265**	**0.275**	**0.170**	**0.299**
ComplEx-PL	**0.282**	0.160	**0.358**	0.333	0.226	0.382
ComplEx-CCSSL	0.253	**0.191**	0.261	**0.335**	**0.232**	**0.378**
QMult-PL	0.269	0.157	**0.309**	0.323	0.228	.351
QMult-CCSSL	**0.294**	**0.217**	**0.309**	**0.328**	**0.238**	**0.363**

Table 11: Link prediction results on UMLS and KINSHIP. The 30:20:50 split ratio for CCSSL-diff and a 50:50 for pseudo-labelling (PL). Bold entries denote best results per method, dataset and metric.

	UMLS			**KINSHIP**		
	MRR	H@1	H@3	MRR	H@1	H@3
DistMult-PL	**0.263**	**0.164**	**0.268**	**0.302**	**0.198**	**0.328**
DistMult-CCSSL	0.240	0.161	0.253	0.245	0.140	0.267
ComplEx-PL	**0.308**	**0.205**	**0.351**	**0.392**	**0.302**	**0.431**
ComplEx-CCSSL	0.228	0.159	0.241	0.255	0.148	0.285
QMult-PL	**0.330**	**0.223**	**0.362**	**0.381**	**0.292**	**0.422**
QMult-CCSSL	0.224	0.150	0.225	0.221	0.118	0.242

Motivated by the findings in the results shown in Tables 9 and 10, we reduced the training set for CCSSL by 25% and increased the training set for pseudo-labeling by 25%. This setting leads to better generalization performance for PL in all metrics, again giving further evidence for our reasoning about the splits. This case clearly demonstrates a limitation of our method: CCSSL is especially favorable in settings where a sufficient amount of labeled instances are provided, particularly when facing structural data such as knowledge graphs.

Appendix D. Generalized Credal Learning: Theoretical Results

In this section, we provide theoretical results on generalized credal learning as introduced in Section 4.2 of the main paper.

D.1. Proof of Theorem 1

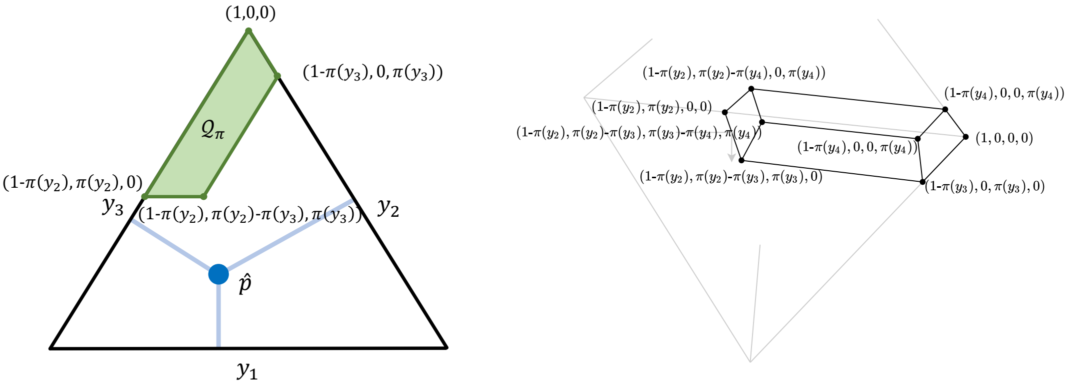

(1,0,0)
(1-$\pi(y_3)$, 0, $\pi(y_3)$)
$\mathcal{Q}_\pi$
(1-$\pi(y_2)$, $\pi(y_2)$, 0)
y_3
(1-$\pi(y_2)$, $\pi(y_2)$-$\pi(y_3)$, $\pi(y_3)$)
y_2
$\hat{p}$
y_1
(1-$\pi(y_2)$, $\pi(y_2)$-$\pi(y_4)$, 0, $\pi(y_4)$)
(1-$\pi(y_4)$, 0, 0, $\pi(y_4)$)
(1-$\pi(y_2)$, $\pi(y_2)$, 0, 0)
(1-$\pi(y_2)$, $\pi(y_2)$-$\pi(y_3)$, $\pi(y_3)$-$\pi(y_4)$, $\pi(y_4)$)
(1, 0, 0, 0)
(1-$\pi(y_4)$, 0, 0, $\pi(y_4)$)
(1-$\pi(y_2)$, $\pi(y_2)$-$\pi(y_3)$, $\pi(y_3)$, 0)
(1-$\pi(y_3)$, 0, $\pi(y_3)$, 0)

Figure 7: Schematic illustration of a credal set $\mathcal{Q}_\pi$ as a convex polytope in a barycentric coordinate space of all distributions $\mathbb{P}(\mathcal{Y})$ for three and four classes. $\mathcal{Q}_\pi$ is induced by a normalized possibility distribution π with $1 = \pi(y_1) \geq \ldots \geq \pi(y_K) \geq 0$.

In the following, we consider possibility distributions $\pi : \mathcal{Y} \longrightarrow [0, 1]$, where $\pi_i := \pi(y_i)$ abbreviates the possibility of class y_i. Without loss of generality, we assume the possibilities be ordered and normalized, i.e., $0 \leq \pi_1 \leq \ldots \leq \pi_K = 1$ for K classes. Furthermore, we also denote the respective probability of a class y_i given a distribution $p \in \mathbb{P}(\mathcal{Y})$ by $p_i := p(y_i)$.

As described in Section 4.2, the set of inequalities that defines the boundary of a credal set $\mathcal{Q}_\pi$ induces a convex polytope. In Fig. 7, such a credal set is illustrated for three and four classes in a barycentric visualization. The extreme points are marked with the respective probabilities.

In Algorithm 1, we provide an algorithm to solve the problem of finding the closest point in the convex polytope $\mathcal{Q}_\pi$ to a query distribution $\hat{p}$. In order to proof Theorem 1 that states the optimality of this approach, we will introduce the following three lemmas:

1. Termination

2. Optimal projection

3. Optimal face

Lemma 1 (Termination) *Given a normalized possibility distribution* $\pi : \mathcal{Y} \longrightarrow [0, 1]$, *Algorithm 1 terminates for an arbitrary probability distribution* $\hat{p} \in \mathbb{P}(\mathcal{Y})$.

Proof In case of $\hat{p} \in \mathcal{Q}_\pi$, we immediately return with a result. For $\hat{p} \notin \mathcal{Q}_\pi$, we fix the probabilities of all $y \in Y'$ in each iteration of Algorithm 1, which are then removed from Y. Thereby, at least for the class $\bar{y} \in Y$ with smallest possibility $\pi(\bar{y})$

$$\bar{p}(\bar{y}) = \left(\pi(\bar{y}) - \sum_{y' \notin Y} p^r(y') \right) \cdot \frac{\hat{p}(\bar{y})}{\hat{p}(\bar{y})} \leq \pi(\bar{y})$$

holds, which does not violate the possibility constraints as in Eq. (12). Here, the set of classes whose probabilities p^r are set is given by $Y' = \{\bar{y}\}$. Consequently, at least one element in Y is removed per step, which eventually results in an empty set Y and the termination of Algorithm 1. ■

In the next lemma, we characterize the optimality of a projection according to the distribution $p^r \in \mathbb{P}(\mathcal{Y})$ as determined in Algorithm 1. In this course, $\bar{Y} \subseteq \mathcal{Y}$ denotes the set of arbitrary, but already fixed probabilities $p^r(y)$ for $y \in \bar{Y}$. This set shall represent classes with optimal probability scores determined in previous iterations. Without loss of generality, we assume that $\forall y \in \bar{Y} : \pi(y) \leq \min_{y' \in \mathcal{Y} \setminus \bar{Y}} \pi(y')$. Moreover, for a certain possibility degree π_i, let us define the set of classes with at most π_i possibility as

$$Y_{\pi_i} := \{ y \in \mathcal{Y} \,|\, \pi(y) \leq \pi_i \} \ . \tag{14}$$

For the remaining classes in $\mathcal{Y} \setminus \bar{Y}$, let p^r be constructed as follows:

$$p^r(y) = \begin{cases} \left(\pi_i - \sum_{y' \in \bar{Y}} p^r(y') \right) \cdot \frac{\hat{p}(y)}{\sum_{y' \in Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y')} & \text{if } y \in Y_{\pi_i} \setminus \bar{Y} \\ (1 - \pi_i) \cdot \frac{\hat{p}(y)}{\sum_{y' \in \mathcal{Y} \setminus Y_{\pi_i}} \hat{p}(y')} & \text{if } y \in \mathcal{Y} \setminus Y_{\pi_i} \end{cases} . \tag{15}$$

Moreover, we use the notion of a *half space* associated with a possibility constraint π_i, which is defined as follows.

Definition 2 (Half-space) *For a possibility constraint* π_i *of a (normalized) possibility distribution* $\pi : \mathcal{Y} \longrightarrow [0, 1]$ *with* $\max_{y \in \mathcal{Y}} \pi(y) = 1$, *we define*

$$\textit{half-space}_{\pi_i} := \left\{ p \in \mathbb{P}(\mathcal{Y}) \,|\, \sum_{y \in \mathcal{Y} : \pi(y) \leq \pi_i} p(y) = \pi_i \right\}$$

as half-space associated with π_i.

Given distributions p^r of the form (15) for a possibility constraint π_i, which is by construction element of half-space$_\pi$ (due to $\forall y \in \bar{Y} : \pi(y) \leq \pi_i$), we can make the following statement about its optimality.

Lemma 3 (Optimal projection) *Given a set $\bar{Y} \subseteq \mathcal{Y}$ of classes with arbitrarily fixed probabilities, a (normalized) possibility distribution $\pi : \mathcal{Y} \longrightarrow [0,1]$ with $\max_{y \in \mathcal{Y}} \pi(y) = 1$ and the set half-space$_{\pi_i}$, the projection $p^r(y) \in$ half-space$_{\pi_i}$ as defined before is optimal in the sense that $\nexists p \in$ half-space$_{\pi_i}$ with $p(y) = p^r(y) \, \forall y \in \bar{Y}$ for which $\exists y \in Y_{\pi_i} \setminus \bar{Y} : p(y) \neq p^r(y)$ such that $D_{KL}(p \,||\, \hat{p}) < D_{KL}(p^r \,||\, \hat{p})$.*

Proof Let us define $A := \sum_{y \in \bar{Y}} p^r(y)$ as the sum of the (previously) fixed probabilities. From the definition of p^r, we know:

$$
\begin{aligned}
D_{KL}(p^r \,||\, \hat{p}) = & \sum_{y \in \mathcal{Y} \setminus Y_{\pi_i}} \frac{(1-\pi_i)\hat{p}(y)}{\sum_{y' \in \mathcal{Y} \setminus Y_{\pi_i}} \hat{p}(y')} \log \frac{\frac{(1-\pi_i)\hat{p}(y)}{\sum_{y' \in \mathcal{Y} \setminus Y_{\pi_i}} \hat{p}(y')}}{\hat{p}(y)} \\
& + \sum_{y \in Y_{\pi_i} \setminus \bar{Y}} \frac{(\pi_i - A)\hat{p}(y)}{\sum_{y' \in Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y')} \log \frac{\frac{(\pi_i - A)\hat{p}(y)}{\sum_{y' \in Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y')}}{\hat{p}(y)} \\
& + \sum_{y \in \bar{Y}} p^r(y) \log \frac{p^r(y)}{\hat{p}(y)} \\
= & \frac{(1-\pi_i)}{\sum_{y' \in \mathcal{Y} \setminus Y_{\pi_i}} \hat{p}(y')} \left(\sum_{y \in \mathcal{Y} \setminus Y_{\pi_i}} \hat{p}(y) \right) \log \frac{(1-\pi_i)}{\sum_{y' \in \mathcal{Y} \setminus Y_{\pi_i}} \hat{p}(y')} \\
& + \frac{(\pi_i - A)}{\sum_{y' \in Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y')} \left(\sum_{y \in Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y) \right) \log \frac{(\pi_i - A)}{\sum_{y' \in Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y')} \\
& + \sum_{y \in \bar{Y}} p^r(y) \log \frac{p^r(y)}{\hat{p}(y)} \\
= & (1-\pi_i) \log \frac{(1-\pi_i)}{\sum_{y' \in \mathcal{Y} \setminus Y_{\pi_i}} \hat{p}(y')} + (\pi_i - A) \log \frac{(\pi_i - A)}{\sum_{y' \in Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y')} \\
& + \sum_{y \in \bar{Y}} p^r(y) \log \frac{p^r(y)}{\hat{p}(y)}
\end{aligned}
\tag{16}
$$

Now, suppose $\exists p \in$ half-space$_{\pi_i}$ with $p(y) = p^r(y) \, \forall y \in \bar{Y}$ for which $\exists y \in Y_{\pi_i} \setminus \bar{Y}$: $p(y) \neq p^r_{\pi_i}(y)$ such that $D_{KL}(p \,||\, \hat{p}) < D_{KL}(p^r \,||\, \hat{p})$, which would lead to a contradiction of the lemma.

$$
\begin{aligned}
&D_{KL}(p\,||\,\hat{p}) - D_{KL}(p^r\,||\,\hat{p}) \\
&= \sum_{y\in\mathcal{Y}\setminus Y_{\pi_i}} p(y)\log\frac{p(y)}{\hat{p}(y)} + \sum_{y\in Y_{\pi_i}\setminus\bar{Y}} p(y)\log\frac{p(y)}{\hat{p}(y)} + \sum_{y\in\bar{Y}} p(y)\log\frac{p(y)}{\hat{p}(y)} \\
&\quad - \sum_{y\in\mathcal{Y}\setminus Y_{\pi_i}} p^r(y)\log\frac{p^r(y)}{\hat{p}(y)} - \sum_{y\in Y_{\pi_i}\setminus\bar{Y}} p^r(y)\log\frac{p^r(y)}{\hat{p}(y)} - \sum_{y\in\bar{Y}} p^r(y)\log\frac{p^r(y)}{\hat{p}(y)} \\
&= \sum_{y\in\mathcal{Y}\setminus Y_{\pi_i}} p(y)\log\frac{p(y)}{\hat{p}(y)} + \sum_{y\in Y_{\pi_i}\setminus\bar{Y}} p(y)\log\frac{p(y)}{\hat{p}(y)} \\
&\quad - \sum_{y\in\mathcal{Y}\setminus Y_{\pi_i}} p^r(y)\log\frac{p^r(y)}{\hat{p}(y)} - \sum_{y\in Y_{\pi_i}\setminus\bar{Y}} p^r(y)\log\frac{p^r(y)}{\hat{p}(y)} \\
&\geq \left(\sum_{y\in\mathcal{Y}\setminus Y_{\pi_i}} p(y)\right)\log\frac{\left(\sum_{y\in\mathcal{Y}\setminus Y_{\pi_i}} p(y)\right)}{\left(\sum_{y\in\mathcal{Y}\setminus Y_{\pi_i}} \hat{p}(y)\right)} + \left(\sum_{y\in Y_{\pi_i}\setminus\bar{Y}} p(y)\right)\log\frac{\left(\sum_{y\in Y_{\pi_i}\setminus\bar{Y}} p(y)\right)}{\left(\sum_{y\in Y_{\pi_i}\setminus\bar{Y}} \hat{p}(y)\right)} \\
&\quad - \sum_{y\in\mathcal{Y}\setminus Y_{\pi_i}} p^r(y)\log\frac{p^r(y)}{\hat{p}(y)} - \sum_{y\in Y_{\pi_i}\setminus\bar{Y}} p^r(y)\log\frac{p^r(y)}{\hat{p}(y)}
\end{aligned}
$$

Here, the last equation can be derived from the log sum inequality.[6] As we are projecting onto the half space associated with π_i, we can further see that $\sum_{y\in\mathcal{Y}\setminus Y_{\pi_i}} p(y) = 1-\pi_i$ and $\sum_{y\in Y_{\pi_i}\setminus\bar{Y}} p(y) = \pi_i - A$, which also applies to p^r for the same class subsets.

As a result, together with (16), we get

$$
\begin{aligned}
&= (1-\pi_i)\log\frac{(1-\pi_i)}{\left(\sum_{y\in\mathcal{Y}\setminus Y_{\pi_i}} \hat{p}(y)\right)} + (\pi_i - A)\log\frac{(\pi_i - A)}{\left(\sum_{y\in Y_{\pi_i}\setminus\bar{Y}} \hat{p}(y)\right)} \\
&\quad - (1-\pi_i)\log\frac{(1-\pi_i)}{\left(\sum_{y\in\mathcal{Y}\setminus Y_{\pi_i}} \hat{p}(y)\right)} - (\pi_i - A)\log\frac{(\pi_i - A)}{\left(\sum_{y\in Y_{\pi_i}\setminus\bar{Y}} \hat{p}(y)\right)} = 0 \not< 0 \ ,
\end{aligned}
$$

which leads to a contradiction. ■

For the next lemma, we introduce the notion of a *face* as follows:

Definition 4 (Face) *For a possibility constraint π_i of a (normalized) possibility distribution $\pi : \mathcal{Y} \longrightarrow [0,1]$ and distributions $p \in \mathbb{P}(\mathcal{Y})$ with $\sum_{y\in\mathcal{Y}:\pi(y)\leq\pi_i} p(y) = \pi_i$ (i.e., $p \in$ half-space$_{\pi_i}$), we define*

$$
face_{\pi_i} := \left\{ p \,|\, \sum_{k=1}^{j} p_k \leq \pi_j, \quad j \in \{1, ..., i\} \right\}
$$

[6] For $a_1,\dots,a_n,b_1,\dots,b_n \in \mathbb{R}_+$, the log sum inequality states that $\sum_{i\in[n]} a_i \log\frac{a_i}{b_i} \geq a\log\frac{a}{b}$, whereby $a := \sum_{i\in[n]} a_i$ and $b := \sum_{i\in[n]} b_i$.

as face associated with π_i*.*

Effectively, face$_{\pi_i}$ considers the subspace of half-space$_{\pi_i}$ that does not violate any of the possibility constraints $\pi_j \leq \pi_i$. One can readily see that each iteration of Algorithm 1 determines the highest possibility constraint $\pi(y^*)$ whose projection p^r as defined before is element of face$_{\pi(y^*)}$. Given this fact, we show the optimality of this selected face in the next lemma.

Lemma 5 (Optimal face) *Given a set* $\bar{Y} \subseteq \mathcal{Y}$ *of classes with arbitrarily fixed probabilities, a (normalized) possibility distribution* $\pi : \mathcal{Y} \longrightarrow [0,1]$ *with* $\max_{y \in \mathcal{Y}} \pi(y) = 1$ *and an arbitrary distribution* $\hat{p} \notin \mathcal{Q}_\pi$*, Algorithm 1 selects face*$_{\pi_i}$ *in each iteration that is optimal in the sense that* $\nexists j \neq i$ *with* $p^* \in \operatorname{argmin}_{p \in face_{\pi_j}} D_{KL}(p \,||\, \hat{p})$*,* $p^*(y) = p^r(y) \, \forall y \in \bar{Y}$ *and* $\exists y \in Y_{\pi_i} \setminus \bar{Y} : p^*(y) \neq p^r(y)$*, such that* $D_{KL}(p^* \,||\, \hat{p}) < D_{KL}(p^r \,||\, \hat{p})$ *for* $p^r \in \operatorname{argmin}_{p \in face_{\pi_i}} D_{KL}(p \,||\, \hat{p})$*.*

Proof Again, we define $A := \sum_{y \in \bar{Y}} p^r(y)$. Now, let us assume $\exists j \neq i$ with the properties described in this lemma, which leads us to a contradiction. To proof it, we can distinguish three cases.

Case 1: $\pi_j > \pi_i$. It is easy to see that a violation with respect to Eq. (12) for possibility $\leq \pi_j$ exists, i.e., the projection constructed as in (15) for π_j is not element of face$_{\pi_j}$. In this case, the distribution $p^* \in \operatorname{argmin}_{\text{face}_{\pi_j}} D_{KL}(p \,||\, \hat{p})$ is located on an “edge” of the face, i.e., $\exists m : \sum_{k=1}^{m} p^*_k = \pi_m$.

If $m = i$, then Lemma 3 implies that $p^*(y) = p^r(y) \, \forall y \in Y_{\pi_i}$, which contradicts the assumptions.

In case of $m \neq i$, we can derive the following results using a similar scheme as in the proof of Lemma 3:

$$
\begin{aligned}
& D_{KL}(p^* \,||\, \hat{p}) - D_{KL}(p^r \,||\, \hat{p}) \\
&= \sum_{y \in \mathcal{Y} \setminus Y_{\pi_i}} p^*(y) \log \frac{p^*(y)}{\hat{p}(y)} + \sum_{y \in Y_{\pi_i} \setminus \bar{Y}} p^*(y) \log \frac{p(y)}{\hat{p}(y)} + \sum_{y \in \bar{Y}} p(y) \log \frac{p(y)}{\hat{p}(y)} \\
&\quad - \sum_{y \in \mathcal{Y} \setminus Y_{\pi_i}} p^r(y) \log \frac{p^r(y)}{\hat{p}(y)} - \sum_{y \in Y_{\pi_i} \setminus \bar{Y}} p^r(y) \log \frac{p^r(y)}{\hat{p}(y)} - \sum_{y \in \bar{Y}} p^r(y) \log \frac{p^r(y)}{\hat{p}(y)} \\
&\geq \left(\sum_{y \in \mathcal{Y} \setminus Y_{\pi_i}} p^*(y) \right) \log \frac{\left(\sum_{y \in \mathcal{Y} \setminus Y_{\pi_i}} p^*(y) \right)}{\left(\sum_{y \in \mathcal{Y} \setminus Y_{\pi_i}} \hat{p}(y) \right)} + \left(\sum_{y \in Y_{\pi_i} \setminus \bar{Y}} p^*(y) \right) \log \frac{\left(\sum_{y \in Y_{\pi_i} \setminus \bar{Y}} p^*(y) \right)}{\left(\sum_{y \in Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y) \right)} \\
&\quad - \sum_{y \in \mathcal{Y} \setminus Y_{\pi_i}} p^r(y) \log \frac{p^r(y)}{\hat{p}(y)} - \sum_{y \in Y_{\pi_i} \setminus \bar{Y}} p^r(y) \log \frac{p^r(y)}{\hat{p}(y)}
\end{aligned}
$$

As we know that p^* does not violate any possibility constraints in $Y_{\pi_j} \supset Y_{\pi_i}$ due to $p^* \in \text{face}_{\pi_j}$, but is also not on the same edge as implied by π_i, it holds that

$$\sum_{y \in Y_{\pi_i} \setminus \bar{Y}} p^*(y) < \pi_i - A \enspace .$$

Moreover, as we know that there is no possibility violation by $p^r(y) \, \forall \in Y_{\pi_i}$ associated with π (cf. Lemma 3), it must hold $\sum_{Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y) > \pi_i - A$. Otherwise, $\hat{p}$ would be element of $\mathcal{Q}_\pi$.

Altogether, one can readily follow

$$\begin{aligned} = & \underbrace{\left(\sum_{y \in \mathcal{Y} \setminus Y_{\pi_i}} p^*(y) \right)}_{>1-\pi_i} \log \frac{\overbrace{\left(\sum_{y \in \mathcal{Y} \setminus Y_{\pi_i}} p^*(y) \right)}^{>1-\pi_i}}{\left(\sum_{y \in \mathcal{Y} \setminus Y_{\pi_i}} \hat{p}(y) \right)} + \underbrace{\left(\sum_{y \in Y_{\pi_i} \setminus \bar{Y}} p^*(y) \right)}_{<\pi_i - A} \log \frac{\overbrace{\left(\sum_{y \in Y_{\pi_i} \setminus \bar{Y}} p^*(y) \right)}^{<\pi_i - A}}{\left(\sum_{y \in Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y) \right)} \\ & - (1 - \pi_i) \log \frac{(1 - \pi_i)}{\sum_{y' \in \mathcal{Y} \setminus Y_{\pi_i}} \hat{p}(y')} - (\pi_i - A) \log \frac{(\pi_i - A)}{\sum_{y' \in Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y')} > 0 \not< 0 \enspace , \end{aligned} \tag{17}$$

leading to a contradiction.

Case 2: $\pi_j = \pi_i$. Lemma 3 implies the optimality of p^r in this case.

Case 3: $\pi_j < \pi_i$. Here, one can again distinguish whether there exists a violation of p^r constructed for π_j or not. In the first case, we can apply exactly the same idea as before by showing that the projection $p^* \in \operatorname{argmin}_{p \in \text{face}_{\pi_j}} D_{KL}(p \, || \, \hat{p})$ is located on an edge that is associated with a π_m with $m < i$.

In case there is no violation, we know that $\sum_{k=1}^{j} p_k^* = \pi_j$. This leads to $\sum_{k=1}^{i} p_k^* \leq \pi_i$. Together with $\sum_{Y_{\pi_i} \setminus \bar{Y}} \hat{p}(y) > \pi_i - A$, one can derive a similar equation as in Lemma 17 to show that

$$D_{KL}(p^* \, || \, \hat{p}) - D_{KL}(p^r \, || \, \hat{p}) \geq 0 \not< 0 \enspace ,$$

leading again to a contradiction.

■

By combining the previous results, we are ready to proof the optimality of Algorithm 1.

Theorem 6 (Optimality) *Given a credal set $\mathcal{Q}_\pi$ induced by a normalized possibility distribution $\pi : \mathcal{Y} \longrightarrow [0, 1]$ with $\max_{y \in \mathcal{Y}} \pi(y) = 1$ according to (1), Algorithm 1 returns the solution of $\mathcal{L}^*(\mathcal{Q}_\pi, \hat{p})$ as defined in (11) for an arbitrary distribution $\hat{p} \in \mathbb{P}(\mathcal{Y})$.*

Proof Combining the three previous lemmas, as well as the fact that the solution of $\operatorname{argmin}_{p \in \mathcal{Q}_\pi} D_{KL}(p \,||\, \hat{p})$ is always characterized by an extreme point on one of the faces of the convex polytope $\mathcal{Q}_\pi$ for $\hat{p} \notin \mathcal{Q}_\pi$, leads to Theorem 6: In each iteration, we choose the optimal projection on the optimal face. Thus, we maintain the optimal probabilities $p^r(y)$ for all $y \notin Y$.

In case of $\hat{p} \in \mathcal{Q}_\pi$, Algorithm 1 returns $D_{KL}(\hat{p} \,||\, \hat{p}) = 0$, which is optimal by definition of D_{KL}. ■

D.2. Complexity

The complexity of Algorithm 1 requires a proper specification of how y^* is determined in the while loop. In our implementation, we sort the classes $y \in \mathcal{Y} = \{y_1, \dots, y_K\}$ according to their possibilities $\pi(y)$ in a descending manner first, which can be done in $\mathcal{O}(K \log K)$. Then, the while loop iterates over the sorted classes and can continue with the next while-loop until a matching constraint $\pi(y^*)$ could be determined. This violation check involves iterating over all classes $y \in Y$ with $\pi(y) \leq \pi(y^*)$. By sorting the elements in advance, this becomes efficient.

Worst-Case Complexity In the worst-case, every iteration of Algorithm 1 has to iterate over all remaining elements in Y. As said before, checking violations of the possibility constraints requires iterating over all involved classes. Thus, the worst-case complexity can be (loosely) bounded by

$$\sum_{i=0}^{K-1} (i+1)(K-i) = \frac{K^3}{6} + \frac{K^2}{2} + \frac{K}{3} = \mathcal{O}(K^3) \ .$$

Average-Case Complexity Although we are not providing a rigorous analysis of the average-case complexity here, we characterize the efficiency of our algorithm in several cases.

We can observe that the worst case applies whenever the projection of a query distribution $\hat{p} \notin \mathcal{Q}_\pi$ on the convex polytope $\mathcal{Q}_\pi$ is the distribution p^* with $p^*(y_i) = \pi(y_i) - \pi(y_{i-1})$ for all $i \in \{1, \dots, K-1\}$ and $p^*(y_K) = \pi(y_K)$ for sorted classes y_i according to their possibilities. This is the case when $\hat{p}$ is in the cone associated with this extreme point p^* (Škulj, 2022), which is given by

$$\left\{ p \notin \mathcal{Q}_\pi \,|\, p^* \in \operatorname*{argmin}_{p' \in \mathcal{Q}_\pi} D_{KL}(p' \,||\, p) \right\} .$$[7]

This set, however, depends on the size of the faces resp. the credal set and is typically rather small. Moreover, it gets proportionally smaller with higher values of K.

In the (trivial) case of $\hat{p} \in \mathcal{Q}_\pi$, we achieve linear complexity $\mathcal{O}(K)$ as we have to check the possibility constraints for all K classes only once. In the other cases, the complexity depends on the face on which we need to project $\hat{p}$: When projecting on face_{π_i}, we do not have to consider any class y with $\pi(y) \leq \pi_i$. Roughly speaking, the larger the faces associated with high possibilities become, the higher the chance of (optimally) projecting directly on

[7] More precisely, one would have to distinguish the cases where a p^r projection as in Eq. (15) is perfectly matching the extreme point p^*, but we omit it here for simplicity.

this face and not requiring any loop iterations over classes with smaller possibility values, leading to a sublinear amount of face projections and thus reducing the cubic complexity.

References

Lukas Biewald. Experiment tracking with weights and biases, 2020. URL https://www.wandb.com/. Software available from wandb.com.

Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Xiaochun Cao, and Qingming Huang. Dual quaternion knowledge graph embeddings. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI, Virtual Event, February 2-9*, pages 6894–6902. AAAI Press, 2021.

Leonardo Cella and Ryan Martin. Valid inferential models for prediction in supervised learning problems. In *International Symposium on Imprecise Probability: Theories and Applications, ISIPTA, Granada, Spain, July 6-9*, volume 147 of *Proceedings of Machine Learning Research*, pages 72–82. PMLR, 2021.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. EMNIST: an extension of MNIST to handwritten letters. *CoRR*, abs/1702.05373, 2017.

Caglar Demir, Diego Moussallem, Stefan Heindorf, and Axel-Cyrille Ngonga Ngomo. Convolutional hypercomplex embeddings for link prediction. In *Proc. of The 13th Asian Conference on Machine Learning, ACML, virtual, November 17-19*, volume 157 of *Proceedings of Machine Learning Research*, pages 656–671. PMLR, 2021.

Caglar Demir, Julian Lienen, and Axel-Cyrille Ngonga Ngomo. Kronecker decomposition for knowledge graph embeddings. In *Proc. of the 33rd ACM Conference on Hypertext and Social Media, HT, Barcelona, Spain, June 28 - July 1*, pages 1–10. ACM, 2022.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2d knowledge graph embeddings. In *Proc. of the 32nd AAAI Conference on Artificial Intelligence, AAAI, New Orleans, Louisiana, USA, February 2-7*, pages 1811–1818. AAAI Press, 2018.

Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, José Emilio Labra Gayo, Sabrina Kirrane, Sebastian Neumaier, Axel Polleres, et al. Knowledge graphs. *arXiv preprint arXiv:2003.02320*, 2020.

Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, International Conference on Machine Learning, ICML, Atlanta, GA, USA, June 16-21*, volume 3, 2013.

Lisha Li, Kevin G. Jamieson, Giulia DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *J. Mach. Learn. Res.*, 18:185:1–185:52, 2017.

Julian Lienen and Eyke Hüllermeier. Credal self-supervised learning. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-14*, pages 14370–14382, 2021.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *Proc. of the 5th International Conference on Learning Representations, ICLR, Toulon, France, April 24-26*. OpenReview.net, 2017.

Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2015.

Daniel Ruffinelli, Samuel Broscheit, and Rainer Gemulla. You CAN teach an old dog new tricks! on training knowledge graph embeddings. In *Proc. of the 8th International Conference on Learning Representations, ICLR, Addis Ababa, Ethiopia, April 26-30*. OpenReview.net, 2020.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, virtual, December 6-12*, 2020.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *Proc. of the 33nd International Conference on Machine Learning, ICML, New York City, NY, USA, June 19-24*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 2071–2080. JMLR.org, 2016.

Colin Wei, Kendrick Shen, Yining Chen, and Tengyu Ma. Theoretical analysis of self-training with deep networks on unlabeled data. In *9th International Conference on Learning Representations, ICLR, Virtual Event, Austria, May 3-7*. OpenReview.net, 2021.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*, 2014.

Damjan Škulj. Normal cones corresponding to credal sets of lower probabilities, 2022.

D

Appendix to Mitigating Label Noise through Data Ambiguation

Mitigating Label Noise through Data Ambiguation: Supplementary Material

Julian Lienen
Department of Computer Science
Paderborn University
Paderborn 33098, Germany
julian.lienen@upb.de

Eyke Hüllermeier
Institute of Informatics, LMU Munich
Munich Center for Machine Learning
Munich 80538, Germany
eyke@lmu.de

A Algorithmic RDA Loss Description

Algorithm 1 provides a pseudo-code description of the loss calculation per batch in the RDA approach.

Algorithm 1 Robust Data Ambiguation (RDA) Loss Calculation

```
Require: Training instance $(\boldsymbol{x}, y) \in \mathcal{X} \times \mathcal{Y}$, model prediction $\widehat{p}(\boldsymbol{x}) \in \mathbb{P}(\mathcal{Y})$, confidence threshold $\beta \in [0, 1]$, relaxation parameter $\alpha \in [0, 1)$
Construct $\pi$ as in Eq. (4) with
    $$\pi(y') = \begin{cases} 1 & \text{if } y' = y \vee \hat{p}(y' \,|\, \boldsymbol{x}) \geq \beta \\ \alpha & \text{otherwise} \end{cases}$$
return $\mathcal{L}^*(Q_\pi, \hat{p}(\boldsymbol{x}))$ as specified in Eq. (4), where $Q_\pi$ is derived from $\pi$
```

B Evaluation Details

B.1 Dataset and Data Preprocessing

As datasets, we consider a wide range of commonly studied image classification datasets. While CIFAR-10 and -100 [Krizhevsky and Hinton, 2009], MNIST [LeCun et al., 1998], FashionMNIST [Xiao et al., 2017] and SVHN [Netzer et al., 2011] (the latter three are used in an additional study in Sec. C.3) are widely known and well-studied, WebVision [Li et al., 2017] and Clothing1M [Xiao et al., 2015] comprise real-world noise from human annotations. For WebVision (version 1.0), we consider the Google image subset of ca. 66,000 images from the top-50 classes resized to 256x256. Clothing1M consists of 1 million training images with noisy and 10,000 test images with clean labels. Here, we also resize the images to 256x256.

For training, we apply data augmentation to the training images as commonly being done in image classification. To do so, we randomly crop images of size 32 (CIFAR-10(0)(N), SVHN), 227 (WebVision) or 224 (Clothing1M), followed by horizontally flipping the image with probability of 0.5. In case of MNIST and FashionMNIST, we keep the images unchanged, but resize them to size 32. The reported dimensions are used to preserve comparability with previous results (e.g., as reported in [Liu et al., 2020]).

B.2 Synethtic Noise Model

While CIFAR-10N and -100N [Wei et al., 2022], WebVision and Clothing1M provide real-world noise in the standard annotations, we modeled additional label corruptions for the rest of the datasets

Preprint. Under review.

in a synthetic manner. Thereby, we distinguish symmetric and asymmetric noise: For the former, each class is treated equally, whereas the asymmetric noise applies individual corruptions to each class.

In case of the symmetric noise, we flip a parameterized fraction $\rho \in [0, 1]$ of instances by uniformly sampling the label from all classes, i.e., we sample a corrupted label via $y \sim \text{Unif}(\mathcal{Y})$ from the uniform distribution $\text{Unif}(\mathcal{Y})$ over the class space $\mathcal{Y}$ with probability ρ. Hence, also correct label can be chosen again. In our studies, we considered values $\rho \in \{0.25, 0.5, 0.75\}$.

In case of asymmetric noise, we use the same setup as suggested in [Patrini et al., 2017]. For CIFAR-10, we flip “truck” to “automobile”, “bird” to “airplane”, “deer” to “horse”, as well as interchange “cat” and “dog”. For CIFAR-100, the classes are grouped into 20 clusters with 5 classes each, whereby labels are flipped within these clusters in a round-robin fashion. In both cases, we apply asymmetric random flips with probability $\rho = 0.4$.

B.3 Hyperparameters

Table 1: Hyperparameters fixed for all baselines and our method per dataset. ResNet50 models proceed from model weights pretrained on ImageNet.

	CIFAR-10(N)	CIFAR-100(N)	WebVision	Clothing1M
Model	ResNet34	ResNet34	ResNet50 (pretrained)	ResNet50 (pretrained)
Batch size	128	128	64	32
Learning rate (LR)	0.02	0.02	0.02	0.002
LR decay	cosine	cosine	cosine	cosine
Weight decay	5×10^{-4}	5×10^{-4}	5×10^{-4}	1×10^{-3}
Epochs	120	150	100	15

In our paper, we follow common evaluation settings as in recent label noise robustness approaches [Zhou et al., 2021, Liu et al., 2022]. To this end, we keep the optimizer hyperparameters as being used in previous reports, allowing for a fair comparison. Table 1 gives an overview over used hyperparameters. In all case, we used SGD with momentum of 0.9 as optimizer.

For the individual losses, we used the optimal parameters being reported in the respective works. This is reasonable as these parameters were optimized in a similar manner on the same data, such that we can re-use these optimization results. For our losses, we fixed the parameters as specified above and optimized the β parameters using random search with 20 iterations employing a 5-fold cross validation on the (partially noisy) training data. To ensure this optimization does not give our method an unfair advantage, we performed the same operation to samples of the baseline experiments, where we noticed that the optimization preferred similar hyperparameter combinations as also reported in the related work.

For results presented in the main paper, we used a cosine decaying strategy as shown in Eq. (5) with $\beta_0 = 0.75$ and $\beta_1 = 0.6$ on all datasets except CIFAR-100. On the latter, we used a fixed schedule with $\beta = 0.5$. In Section C.2, we further present an ablation study showing results for different β schedules. In all cases, we set $\alpha = 0.05$.

B.4 Technical Infrastructure

The code of our empirical evaluation is made publicly available[1]. All experiments are implemented using PyTorch[2] as deep learning framework using Nvidia CUDA[3] as acceleration backend. Image manipulations as being used for data agumentation are provided by Pillow[4]. To perform the experiments, we used several Nvidia A100 and RTX 2080 Ti accelerators in a modern high performance computing environment.

[1] See https://github.com/julilien/MitigatingLabelNoiseDataAmbiguation for our official implementation.

[2] https://pytorch.org/, BSD-3 license

[3] https://developer.nvidia.com/cuda-toolkit

[4] https://python-pillow.org/, HPND License

C Further Experiments

C.1 Comparison to Full Ambiguation

In this section, we compare our method RDA to a full ambiguation adaptation: For prediction $\widehat{p}$ exceeding β for a class different to the training label, we ambiguate the target information by a completely agnostic credal sets. Hence, this instance is completely ignored in the loss optimization. Table 2 shows the respective results, where significant improvements over the complete ambiguation can be observed for our method. Hence, incorporating credal sets with two fully plausible classes appears reasonable from a loss minimization context.

Table 2: Averaged test accuracies and standard deviations computed over three runs with different seeds. **Bold** entries mark the best method.

CIFAR-10N	RDA	Complete Ambig.
Clean	**94.09** ±0.19	93.77 ±0.37
Random1	**90.43** ±0.03	87.04 ±0.18
Random2	**90.09** ±0.29	89.36 ±0.16
Random3	**90.40** ±0.01	89.04 ±0.21
Aggregate	**91.71** ±0.38	88.73 ±0.64
Worst	**82.91** ±0.83	76.57 ±0.91

C.2 Parameter Ablation

C.2.1 Schedules

In our paper, we proposed to use a cosine decaying strategy to determine β, reflecting an increase in the certainty of the model over the course of the training. Here, we compare this strategy to two alternative schedules, namely a constant and a linear function. Table 3 shows the respective results, where the more sophisticated schedules appear to be superior compared to the constant function.

Table 3: Averaged test accuracies and standard deviations computed over three runs with different seeds. **Bold** entries mark the best method.

Schedule	CIFAR-10		
	25 %	50 %	75 %
Constant ($\beta = 0.75$)	90.72 ±0.44	83.97 ±0.81	54.33 ±11.70
Linear ($\beta_0 = 0.75$, $\beta_1 = 0.6$)	91.35 ±0.29	85.16 ±1.48	35.50 ±1.52
Cosine ($\beta_0 = 0.75$, $\beta_1 = 0.6$)	**91.41** ±0.27	**86.46** ±0.47	**56.83** ±1.71

C.2.2 Varying β_0 and β_1

Table 4: Results for different β_1 parameters with $\beta_0 = 0.75$ using a cosine decaying β schedule. **Bold** entries mark the best parameter.

β_1	CIFAR-10	
	25 %	50 %
0.75	90.93	83.74
0.6	91.32	84.21
0.5	**91.65**	**85.98**
0.4	85.72	66.11
0.3	81.89	41.93
0.2	78.49	11.79

In another study, we look how changes in β_0 and β_1 for the cosine schedule behave in terms of generalization performance. The respective results are shown in Table 4 and 5.

Table 5: Results for different β_0 parameters with $\beta_1 = 0.5$ using a cosine decaying β schedule. **Bold** entries mark the best parameter.

β_0	CIFAR-10	
	25 %	50 %
0.8	**90.18**	**82.63**
0.75	89.41	81.71
0.7	88.17	80.58
0.6	86.14	78.44
0.5	69.20	65.71

C.3 Simplified Setting

Relating to the phenomenon of neural collapse [Papyan et al., 2020], we study the effects of our method in a simplified setting. More precisely, we consider multi-layer perceptron models consisting of a feature encoder and a classification head. The models consist of 6 dense layers with a width of 2048 neurons. To conform with previous studies [Nguyen et al., 2022], we restrict the number of features at the last encoder layer to the number of classes K in the datasets. We consider the datasets MNIST, FashionMNIST and SVHN, from which we sample classes $K \in \{2, 3, 5, 10\}$. Apart from the model, we use the same hyperparameters as described in Table 1 for CIFAR-10. We used the same setup to construct Figure 3 as shown in the main paper. For $K = 2$, we can readily investigate the learned feature representations over the course of the training in a convenient manner.

These experiments aim to provide further evidence for the generalization of our method in more restricted settings under simplified model assumptions. Table 6 shows the results, where a consistent improvement of our method compared to typically considered neural collapsing functions can be observed. This suggests that our method is also applicable for more restricted models than typically considered overparameterized models.

Table 6: Test accuracies on the three additional benchmark datasets using a simplified MLP with a restricted feature space at the penultimate layer. **Bold** entries mark the best method.

Loss	MNIST				FashionMNIST				SVHN			
	$K=2$	$K=3$	$K=5$	$K=10$	$K=2$	$K=3$	$K=5$	$K=10$	$K=2$	$K=3$	$K=5$	$K=10$
CE	73.19	67.59	68.44	66.24	78.60	68.33	65.68	59.51	76.60	65.65	58.59	50.57
LS ($\alpha = 0.1$)	74.04	66.44	72.19	71.24	81.45	72.23	67.60	62.78	79.66	68.97	61.74	52.72
LR ($\alpha = 0.1$)	73.66	66.00	66.34	64.58	80.15	69.43	64.20	57.25	76.63	67.00	58.62	49.98
RDA (ours)	**97.59**	**90.78**	**87.86**	**82.75**	**92.45**	**91.63**	**82.36**	**77.13**	**86.77**	**73.03**	**65.67**	**53.98**

C.4 Combination with Sample Selection

In additional experiments, we integrated our RDA approach in a sample selection approach based on the small loss criterion [Gui et al., 2021], as also been applied in more sophisticated (semi-supervised) approaches that add substantial complexity to the learning setup. To this end, we train the model for 3 epochs on CIFAR-10(N)/-100 with cross-entropy based on the training examples, and take the 10 % training instances with the smallest cross-entropy loss. For these instances, we fix the label, i.e., we do not allow any ambiguity such that these instances serve as a corrective. Then, we train the model with our RDA loss as described in Algorithm 1 with the hyperparameters reported in Tab. 1. In the following, we will refer to this variant as *RDA**.

As can be seen in the results present in Tables 7 and 8, RDA* can achieve almost consistently better performance with this addition, confirming its flexibility in being employed in more sophisticated setups. Notably, RDA* shows the largest improvement in robustness in the high noise regime (75 % noise).

Table 7: Test accuracies and standard deviations on the test split for models trained on CIFAR-10(0) with synthetic noise. The results are averaged over runs with different seeds, **bold** entries mark the best performing method per column.

Method	CIFAR-10			CIFAR-100		
	25 %	50 %	75 %	25 %	50 %	75 %
RDA	91.48 ±0.22	86.47 ±0.42	48.11 ±15.41	**70.03** ±0.32	59.83 ±1.15	26.75 ±8.83
RDA*	**91.79** ±0.21	**86.78** ±0.50	**67.08** ±2.09	69.98 ±0.17	**60.18** ±1.18	**30.82** ±9.45

Table 8: Test accuracies and standard deviations on the test split for models trained on CIFAR-10N with synthetic noise. The results are averaged over runs with different seeds, **bold** entries mark the best performing method per column.

Method	CIFAR-10N					
	Clean	Random 1	Random 2	Random 3	Aggregate	Worst
RDA	94.09 ±0.19	90.43 ±0.03	90.09 ±0.29	90.40 ±0.01	91.71 ±0.38	82.91 ±0.83
RDA*	**94.15** ±0.05	**90.76** ±0.13	**90.55** ±0.37	**90.86** ±0.09	**91.84** ±0.18	**83.79** ±0.24

D Additional Training Behavior Plots

D.1 CIFAR-10

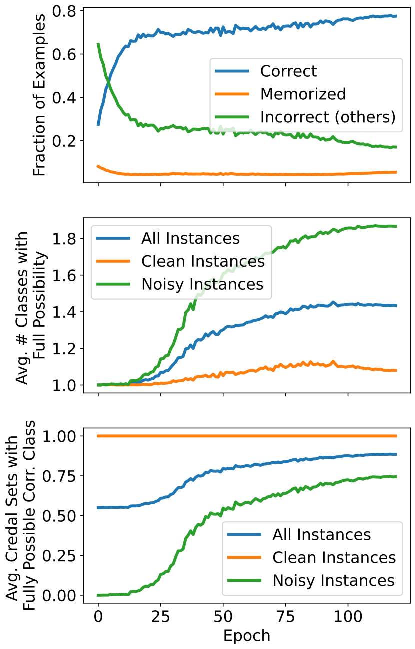


Figure 1: The training behavior of our method on CIFAR-10 with 50 % label noise averaged over five different seeds.

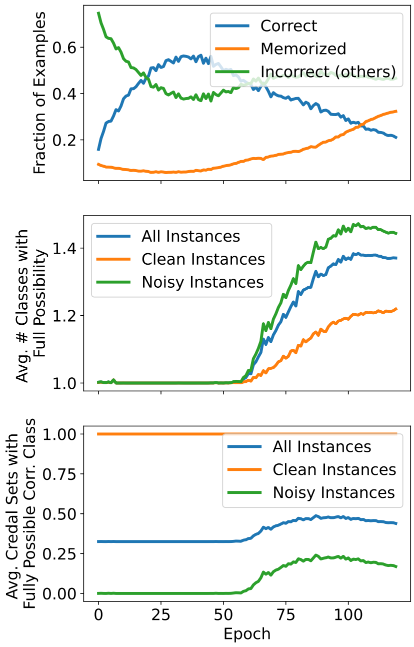


Figure 2: The training behavior of our method on CIFAR-10 with 75 % label noise averaged over five different seeds.

D.2 CIFAR-100

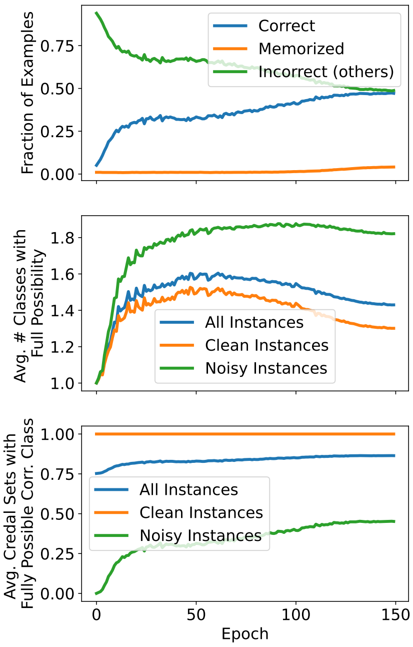


Figure 3: The training behavior of our method on CIFAR-100 with 25 % label noise averaged over five different seeds.

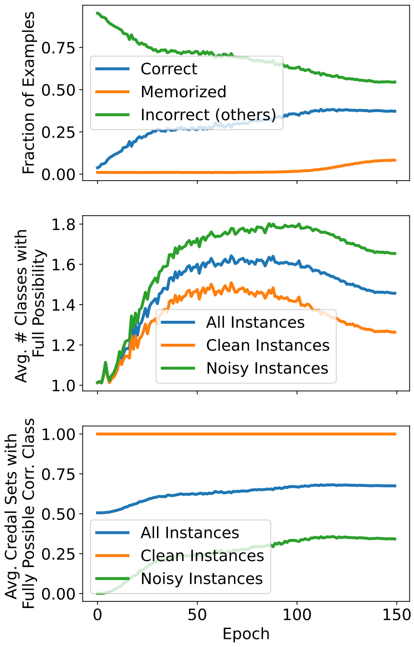


Figure 4: The training behavior of our method on CIFAR-100 with 50 % label noise averaged over five different seeds.

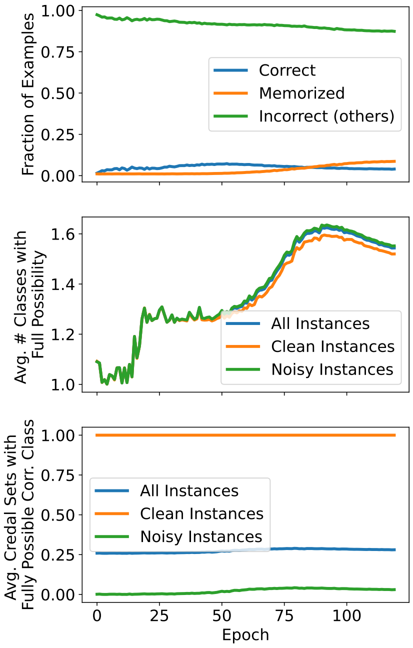


Figure 5: The training behavior of our method on CIFAR-100 with 75 % label noise averaged over five different seeds.

References

X. Gui, W. Wang, and Z. Tian. Towards understanding deep learning from noisy labels with small-loss criterion. In *Proc. of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI, August 19-27, Virtual Event / Montreal, Canada*, pages 2469–2475. ijcai.org, 2021.

A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. Technical report, University of Toronto, Toronto, Canada, 2009.

Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86(11):2278–2324, 1998.

W. Li, L. Wang, W. Li, E. Agustsson, and L. V. Gool. WebVision database: Visual learning and understanding from web data. *CoRR*, abs/1708.02862, 2017.

S. Liu, J. Niles-Weed, N. Razavian, and C. Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NeurIPS, December 6-12, 2020, Virtual Event*, 2020.

S. Liu, Z. Zhu, Q. Qu, and C. You. Robust training under label noise by over-parameterization. In *Proc. of the 39th International Conference on Machine Learning, ICML, July 17-23, Baltimore, Maryland, USA*, volume 162 of *Proc. of Machine Learning Research*, pages 14153–14172. PMLR, 17–23 Jul 2022.

Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems, NIPS, November 12-17, Granada, Spain, Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

D. A. Nguyen, R. Levie, J. Lienen, G. Kutyniok, and E. Hüllermeier. Memorization-dilation: Modeling neural collapse under noise. *CoRR*, abs/2206.05530, 2022.

V. Papyan, X. Y. Han, and D. L. Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proc. of the National Academy of Sciences of the United States of America*, 117:24652 – 24663, 2020.

G. Patrini, A. Rozza, A. K. Menon, R. Nock, and L. Qu. Making deep neural networks robust to label noise: A loss correction approach. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR, July 21-26, Honolulu, HI, USA*, pages 2233–2241. IEEE Computer Society, 2017.

J. Wei, Z. Zhu, H. Cheng, T. Liu, G. Niu, and Y. Liu. Learning with noisy labels revisited: A study using real-world human annotations. In *The Tenth International Conference on Learning Representations, ICLR, April 25-29, Virtual Event*. OpenReview.net, 2022.

H. Xiao, K. Rasul, and R. Vollgraf. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017.

T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang. Learning from massive noisy labeled data for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 7-12, Boston, MA, USA*, pages 2691–2699. IEEE Computer Society, 2015.

X. Zhou, X. Liu, J. Jiang, X. Gao, and X. Ji. Asymmetric loss functions for learning with noisy labels. In *Proc. of the 38th International Conference on Machine Learning, ICML, Virtual Event, July 18-24*, volume 139 of *Proc. of Machine Learning Research*, pages 12846–12856. PMLR, 2021.

E

Appendix to Robust Regression for Monocular Depth Estimation

– Supplementary Material –
Robust Regression for Monocular Depth Estimation

Julian Lienen JULIAN.LIENEN@UPB.DE
Paderborn University, Germany

Nils Nommensen NILS.NOMMENSEN@TIB.EU
Ralph Ewerth RALPH.EWERTH@TIB.EU
L3S Research Center, Leibniz University Hannover and TIB Hannover, Germany

Eyke Hüllermeier EYKE@IFI.LMU.DE
University of Munich (LMU), Germany

Editors: Vineeth N Balasubramanian and Ivor Tsang

1. Experimental Settings

To demonstrate the effectiveness of robust depth estimation methods, we present results for models trained with a variety of robust loss functions. As most of them involve loss-specific hyperparameters, as well as the optimization algorithm itself, we conducted a hyperparameter optimization for each induced model.

To this end, we employed a random search with 20 trials per run. In each trial, the model was trained using the regarded hyperparameter configuration for 25 epochs with a batch size of 16. When training on a subset of a given data set, we randomly sampled the desired number of instances from the original training split. We used Adam with default parameters as optimizer and optimized the initial learning rate $\eta \in [1e^{-4}, 1e^{-1}]$. As learning rate schedule, we applied cosine annealing (Loshchilov and Hutter, 2017). Moreover, we augmented the training and depth pairs by randomly flipping the images horizontally, augmenting the colors with varying hue, saturation, brightness and contrast, and randomly swapped the red and blue color channels. To select the model for the final assessment, the validation root mean squared error was calculated on *iBims-1* (Koch et al., 2018).

Table 1 enlists the hyperparameters along with their considered search spaces being optimized. We kept the notation the same as in the referred original publications.

Given the finally selected models, we evaluated the depth metrics as described in the paper and further extended in this supplementary material. As *NYUD-v2* (Silberman et al., 2012) and *SunRGBD* (Song et al., 2015) only provide depth values up to 10 m, we only assessed depth values in the ground truth data of the benchmark data sets with up to this value.

We ran all experiments on a modern high-performance cluster with several Nvidia RTX 1080 Ti, 2080 Ti and Titan RTX accelerators. In total, the experimental evaluation took about 4500 GPU hours.

Table 1: Loss-specific hyperparameters and their search spaces being considered within the random search.

Loss	Hyperparameter Space(s)
$\mathcal{L}_{\text{Huber}}$ (Laina et al., 2016)	$c \in [0.1, 0.9]$
$\mathcal{L}_{\text{BerHu}}$ (Laina et al., 2016)	$c \in [0.1, 0.9]$
$\mathcal{L}_{\text{Ruber}}$ (Irie et al., 2019)	$c \in [0.1, 0.9]$
$\mathcal{L}_{\text{Barron}}$ (Barron, 2019)	$\alpha \in [0, 2], c \in [0.1, 50]$
$\mathcal{L}_{\text{trim}}$ (Ranftl et al., 2020)	$U \in [0.1, 0.9]$ {Largest % of residuals being trimmed}
$\mathcal{L}_{\text{ScaledSIError}}$ (Lee et al., 2019)	$\lambda \in [0.1, 0.9], \alpha \in [0.1, 25]$
$\text{OSL}_{\mathcal{L}_1}$	$\epsilon \in [0, 0.25]$
$\text{OSL}_{\mathcal{L}_2}$	$\epsilon \in [0, 0.25]$
$\text{FOSL}_{\mathcal{L}_1}$	$\epsilon \in [0, 0.25], \delta \in [0, 2]$

2. Model Architecture

Fig. 1 shows the EfficientNet-based (Tan and Le, 2019) architecture as being used within our empirical studies. More precisely, we considered a EfficientNetB0 encoder pretrained on ImageNet and freezed the weights during training. While the encoder part comprises approx. 4 million parameters, we effectively maintained 11 million decoder weights, resulting in a total of 15 million parameters. As can be seen in the figure, the decoder part consists of stacked upsamling components applying convolutional, BatchNormalization, ReLU activation and bilinear upsampling layers.

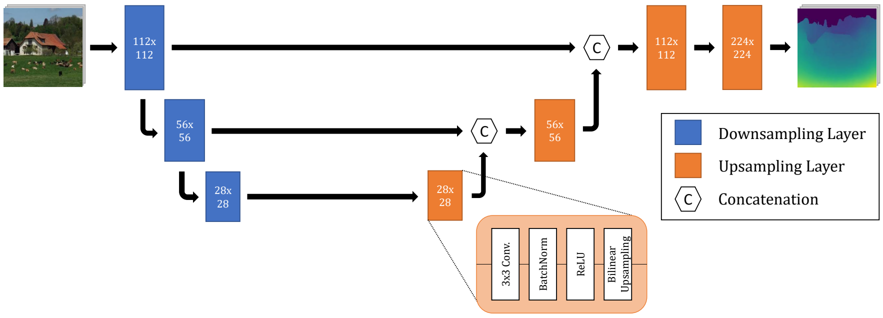


Figure 1: EfficientNet-based U-Net architecture as being used within the empirical evaluation. The blue downsampling layers are specified by the employed backbone. The layer captions denote the corresponding output dimensionality of the respective layers.

3. Additional Experimental Results

3.1. Metrics

Beyond the reported metrics in the paper, we further present results on the following additional metrics.

- Root mean squared error log (RMSLog): $\sqrt{\frac{1}{n}\sum_{i=1}^{n}(\log y_i - \log \hat{y}_i)^2}$
- Squared relative difference (SQREL): $\frac{1}{n}\sum_{i=1}^{n}\frac{(y_i-\hat{y}_i)^2}{y_i}$

3.2. Homogeneous Depth Sensor: NYUD-v2

Tab. 2 extends Tab. 1 as presented in the main paper by providing results on *DIODE* for the additional metrics, as well as further errors on the Eigen test split of *NYUD-v2*. In accordance to the results presented in the paper, our loss proposals provide the best performance on *DIODE*. In the case of *NYUD-v2*, the scaled SI error turns out to perform best.

Table 2: Averaged results over three runs for additional metrics on *DIODE* and the Eigen test split of *NYUD-v2* for a varying number of instances with the corresponding standard deviations. The best model is indicated in bold per number of instances and metric.

# Insts.	Loss	*DIODE*		*NYUD-v2*				
		RMSLog (↓)	SQREL (↓)	$\log_{10}$ (↓)	RMSLog (↓)	SQREL (↓)	δ_2 (↑)	δ_3 (↑)
2k	$\mathcal{L}_2$	0.607 ± 0.013	1.144 ± 0.058	0.135 ± 0.012	0.393 ± 0.037	0.497 ± 0.101	0.757 ± 0.042	0.899 ± 0.025
	$\mathcal{L}_1$	0.598 ± 0.007	1.133 ± 0.076	0.124 ± 0.002	0.381 ± 0.036	0.407 ± 0.059	0.802 ± 0.015	0.926 ± 0.006
	$\mathcal{L}_{\text{Huber}}$	0.589 ± 0.037	1.084 ± 0.088	0.109 ± 0.001	0.316 ± 0.004	0.305 ± 0.014	0.843 ± 0.006	0.949 ± 0.002
	$\mathcal{L}_{\text{BerHu}}$	0.580 ± 0.004	1.074 ± 0.034	0.110 ± 0.004	0.320 ± 0.012	0.317 ± 0.034	0.836 ± 0.013	0.946 ± 0.006
	$\mathcal{L}_{\text{Ruber}}$	0.592 ± 0.020	1.085 ± 0.024	0.107 ± 0.005	0.313 ± 0.013	0.320 ± 0.050	0.847 ± 0.017	0.947 ± 0.010
	$\mathcal{L}_{\text{Barron}}$	0.585 ± 0.013	1.090 ± 0.022	0.115 ± 0.008	0.331 ± 0.020	0.360 ± 0.045	0.824 ± 0.026	0.938 ± 0.012
	$\mathcal{L}_{\text{trim}}$	0.664 ± 0.087	1.183 ± 0.069	0.129 ± 0.007	0.371 ± 0.023	0.575 ± 0.227	0.778 ± 0.024	0.912 ± 0.016
	$\mathcal{L}_{\text{ScaledSIError}}$	0.572 ± 0.016	**1.043** ± 0.008	**0.097** ± 0.007	**0.284** ± 0.018	**0.265** ± 0.043	**0.877** ± 0.019	**0.961** ± 0.008
	$\mathcal{L}_{\text{WeightedL2}}$	0.582 ± 0.005	1.132 ± 0.044	0.134 ± 0.009	0.387 ± 0.028	0.483 ± 0.076	0.761 ± 0.030	0.903 ± 0.019
	$\text{OSL}_{\mathcal{L}_1}$	**0.567** ± 0.024	1.091 ± 0.048	0.123 ± 0.012	0.359 ± 0.035	0.432 ± 0.093	0.794 ± 0.038	0.929 ± 0.021
	$\text{OSL}_{\mathcal{L}_2}$	0.592 ± 0.050	1.203 ± 0.115	0.137 ± 0.016	0.395 ± 0.044	0.561 ± 0.142	0.746 ± 0.055	0.889 ± 0.034
	$\text{FOSL}_{\mathcal{L}_1}$	0.590 ± 0.019	1.103 ± 0.035	0.112 ± 0.005	0.329 ± 0.018	0.336 ± 0.007	0.832 ± 0.018	0.948 ± 0.010
10k	$\mathcal{L}_2$	0.585 ± 0.010	1.088 ± 0.008	0.115 ± 0.001	0.336 ± 0.002	0.343 ± 0.032	0.824 ± 0.006	0.940 ± 0.005
	$\mathcal{L}_1$	0.586 ± 0.030	1.072 ± 0.032	0.095 ± 0.004	0.280 ± 0.011	0.251 ± 0.034	0.883 ± 0.014	0.964 ± 0.005
	$\mathcal{L}_{\text{Huber}}$	0.588 ± 0.027	1.123 ± 0.030	0.096 ± 0.001	0.284 ± 0.003	0.279 ± 0.019	0.878 ± 0.003	0.959 ± 0.002
	$\mathcal{L}_{\text{BerHu}}$	0.593 ± 0.008	1.093 ± 0.010	0.086 ± 0.001	0.258 ± 0.003	0.208 ± 0.009	0.904 ± 0.004	0.973 ± 0.000
	$\mathcal{L}_{\text{Ruber}}$	0.578 ± 0.003	1.068 ± 0.012	0.087 ± 0.004	0.261 ± 0.011	0.218 ± 0.021	0.901 ± 0.010	0.971 ± 0.004
	$\mathcal{L}_{\text{Barron}}$	0.579 ± 0.021	1.103 ± 0.013	0.106 ± 0.009	0.311 ± 0.026	0.315 ± 0.053	0.849 ± 0.029	0.951 ± 0.012
	$\mathcal{L}_{\text{trim}}$	0.602 ± 0.019	1.080 ± 0.022	0.098 ± 0.013	0.313 ± 0.070	0.244 ± 0.048	0.884 ± 0.024	0.964 ± 0.012
	$\mathcal{L}_{\text{ScaledSIError}}$	0.600 ± 0.025	1.096 ± 0.040	**0.079** ± 0.002	**0.237** ± 0.004	**0.171** ± 0.007	**0.926** ± 0.002	**0.981** ± 0.001
	$\mathcal{L}_{\text{WeightedL2}}$	0.580 ± 0.007	1.065 ± 0.021	0.107 ± 0.004	0.314 ± 0.011	0.295 ± 0.028	0.848 ± 0.012	0.950 ± 0.007
	$\text{OSL}_{\mathcal{L}_1}$	0.541 ± 0.004	1.053 ± 0.005	0.107 ± 0.002	0.311 ± 0.004	0.324 ± 0.019	0.844 ± 0.004	0.947 ± 0.004
	$\text{OSL}_{\mathcal{L}_2}$	**0.538** ± 0.006	**1.039** ± 0.023	0.115 ± 0.004	0.335 ± 0.012	0.379 ± 0.036	0.821 ± 0.017	0.932 ± 0.010
	$\text{FOSL}_{\mathcal{L}_1}$	0.597 ± 0.033	1.102 ± 0.045	0.091 ± 0.004	0.272 ± 0.013	0.232 ± 0.033	0.891 ± 0.015	0.966 ± 0.008

Tab. 3 further shows results on *DIODE* and *NYUD-v2* for the study involving artificial noise added to the training data. As can be seen, $\mathcal{L}_{\text{Ruber}}$ performs best on the noisy test data, whereas it provides inferior performance to most of the superset losses on the clear *DIODE* data. This suggests that it is more prone to reproduce the sensor errors.

Table 3: Further results on *DIODE* and the Eigen test split of *NYUD-v2* when trained on 2k instances from *NYUD-v2* with artificial noise (averaged results over three runs). The best model is indicated in bold per noise degree and metric.

Noise $\hat{\epsilon}$	Loss	*DIODE*		*NYUD-v2*							
		RMSLog (↓)	SQREL (↓)	REL (↓)	$\log_{10}$ (↓)	RMS (↓)	δ_1 (↑)	δ_2 (↑)	δ_3 (↑)	RMSLog (↓)	SQREL (↓)
0.5	$\mathcal{L}_2$	0.652 ± 0.031	2.074 ± 0.281	0.789 ± 0.055	0.226 ± 0.012	1.677 ± 0.098	0.221 ± 0.012	0.484 ± 0.026	0.700 ± 0.031	0.605 ± 0.031	1.669 ± 0.218
	$\mathcal{L}_1$	0.597 ± 0.013	1.126 ± 0.034	0.343 ± 0.026	0.130 ± 0.008	0.976 ± 0.061	0.474 ± 0.027	0.773 ± 0.030	0.912 ± 0.017	0.388 ± 0.021	0.419 ± 0.053
	$\mathcal{L}_{\text{Huber}}$	0.568 ± 0.018	1.301 ± 0.177	0.464 ± 0.090	0.150 ± 0.021	1.175 ± 0.175	0.427 ± 0.068	0.705 ± 0.065	0.862 ± 0.041	0.431 ± 0.047	0.732 ± 0.243
	$\mathcal{L}_{\text{BerHu}}$	0.556 ± 0.014	1.080 ± 0.027	0.306 ± 0.008	0.112 ± 0.003	0.860 ± 0.028	**0.545** ± 0.014	0.829 ± 0.008	0.941 ± 0.003	**0.326** ± 0.008	0.348 ± 0.022
	$\mathcal{L}_{\text{Ruber}}$	0.580 ± 0.008	1.079 ± 0.012	**0.286** ± 0.012	0.112 ± 0.004	**0.853** ± 0.024	0.535 ± 0.010	**0.833** ± 0.010	**0.946** ± 0.006	0.328 ± 0.014	**0.307** ± 0.023
	$\mathcal{L}_{\text{Barron}}$	0.591 ± 0.016	1.591 ± 0.130	0.667 ± 0.028	0.200 ± 0.005	1.473 ± 0.045	0.249 ± 0.006	0.551 ± 0.013	0.772 ± 0.023	0.538 ± 0.016	1.216 ± 0.105
	$\mathcal{L}_{\text{trim}}$	0.612 ± 0.039	1.417 ± 0.022	0.510 ± 0.002	0.174 ± 0.003	1.269 ± 0.011	0.360 ± 0.001	0.635 ± 0.002	0.818 ± 0.001	0.527 ± 0.034	0.803 ± 0.023
	$\mathcal{L}_{\text{ScaledSIError}}$	1.143 ± 0.491	1.796 ± 0.545	0.521 ± 0.175	0.369 ± 0.187	1.753 ± 0.472	0.154 ± 0.142	0.306 ± 0.269	0.437 ± 0.351	0.914 ± 0.412	0.993 ± 0.486
	$\mathcal{L}_{\text{WeightedL2}}$	0.613 ± 0.007	1.671 ± 0.028	0.618 ± 0.009	0.189 ± 0.002	1.362 ± 0.015	0.315 ± 0.006	0.590 ± 0.005	0.778 ± 0.005	0.522 ± 0.004	1.076 ± 0.025
	$\text{OSL}_{\mathcal{L}_1}$	**0.555** ± 0.012	1.117 ± 0.037	0.369 ± 0.057	0.129 ± 0.013	0.932 ± 0.078	0.491 ± 0.043	0.774 ± 0.045	0.907 ± 0.029	0.371 ± 0.037	0.462 ± 0.104
	$\text{OSL}_{\mathcal{L}_2}$	0.621 ± 0.026	1.884 ± 0.242	0.770 ± 0.079	0.223 ± 0.018	1.552 ± 0.149	0.208 ± 0.026	0.487 ± 0.048	0.717 ± 0.044	0.595 ± 0.047	1.581 ± 0.307
	$\text{FOSL}_{\mathcal{L}_1}$	0.583 ± 0.021	**1.078** ± 0.050	0.300 ± 0.012	**0.107** ± 0.006	0.857 ± 0.029	0.529 ± 0.021	0.825 ± 0.017	0.939 ± 0.009	0.343 ± 0.031	0.34 ± 0.02
1.0	$\mathcal{L}_2$	0.898 ± 0.093	5.929 ± 0.764	1.387 ± 0.387	0.333 ± 0.073	2.899 ± 0.714	0.130 ± 0.097	0.268 ± 0.161	0.438 ± 0.170	0.861 ± 0.151	5.079 ± 2.088
	$\mathcal{L}_1$	0.561 ± 0.006	1.117 ± 0.022	0.366 ± 0.014	0.133 ± 0.003	1.007 ± 0.031	0.460 ± 0.007	0.758 ± 0.013	0.904 ± 0.009	0.381 ± 0.008	0.464 ± 0.031
	$\mathcal{L}_{\text{Huber}}$	0.701 ± 0.010	2.834 ± 0.151	1.073 ± 0.102	0.286 ± 0.020	2.384 ± 0.229	0.112 ± 0.026	0.307 ± 0.055	0.556 ± 0.052	0.735 ± 0.043	3.017 ± 0.472
	$\mathcal{L}_{\text{BerHu}}$	0.566 ± 0.008	1.107 ± 0.036	0.375 ± 0.036	0.132 ± 0.008	0.993 ± 0.059	0.482 ± 0.022	0.772 ± 0.029	0.905 ± 0.018	0.389 ± 0.028	0.498 ± 0.079
	$\mathcal{L}_{\text{Ruber}}$	0.571 ± 0.001	1.096 ± 0.035	**0.295** ± 0.030	**0.113** ± 0.008	**0.880** ± 0.071	**0.532** ± 0.030	**0.833** ± 0.026	**0.945** ± 0.014	**0.325** ± 0.023	**0.354** ± 0.085
	$\mathcal{L}_{\text{Barron}}$	0.721 ± 0.041	3.004 ± 0.581	1.001 ± 0.134	0.270 ± 0.027	2.059 ± 0.254	0.152 ± 0.052	0.367 ± 0.075	0.592 ± 0.066	0.707 ± 0.061	2.541 ± 0.592
	$\mathcal{L}_{\text{trim}}$	0.575 ± 0.003	1.361 ± 0.036	0.488 ± 0.019	0.167 ± 0.003	1.231 ± 0.018	0.369 ± 0.009	0.648 ± 0.011	0.828 ± 0.008	0.470 ± 0.009	0.724 ± 0.045
	$\mathcal{L}_{\text{ScaledSIError}}$	1.124 ± 0.407	1.912 ± 0.352	0.593 ± 0.115	0.410 ± 0.188	1.920 ± 0.448	0.134 ± 0.144	0.265 ± 0.260	0.381 ± 0.320	1.025 ± 0.424	1.209 ± 0.411
	$\mathcal{L}_{\text{WeightedL2}}$	0.670 ± 0.033	1.861 ± 0.266	0.667 ± 0.100	0.201 ± 0.021	1.464 ± 0.144	0.291 ± 0.050	0.553 ± 0.058	0.750 ± 0.044	0.555 ± 0.050	1.249 ± 0.301
	$\text{OSL}_{\mathcal{L}_1}$	**0.559** ± 0.016	1.093 ± 0.018	0.351 ± 0.059	0.122 ± 0.014	0.933 ± 0.075	0.511 ± 0.044	0.796 ± 0.045	0.918 ± 0.028	0.354 ± 0.037	0.441 ± 0.110
	$\text{OSL}_{\mathcal{L}_2}$	0.783 ± 0.033	3.304 ± 0.811	1.211 ± 0.276	0.310 ± 0.051	2.546 ± 0.566	0.125 ± 0.073	0.288 ± 0.129	0.493 ± 0.132	0.818 ± 0.083	3.812 ± 1.484
	$\text{FOSL}_{\mathcal{L}_1}$	**0.559** ± 0.007	**1.089** ± 0.030	0.322 ± 0.014	0.118 ± 0.003	0.910 ± 0.029	0.516 ± 0.015	0.811 ± 0.010	0.931 ± 0.007	0.342 ± 0.009	0.384 ± 0.036

3.3. Heterogeneous Depth Sensors: SunRGBD

In Tab. 4, we provide more results on *DIODE* and the official test split of *SunRGBD* when trained on the corresponding training part. While the results on *DIODE* are matching the observations made with regard to the other metrics, $\mathcal{L}_{\text{WeightedL2}}$ performs remarkably well on the noisy *SunRGBD* test data when trained on the full data. It is worth to note that the performance on the latter test data gets worse the more data is observed. This could potentially be due to a domain-shift between *iBims-1* and *SunRGBD*.

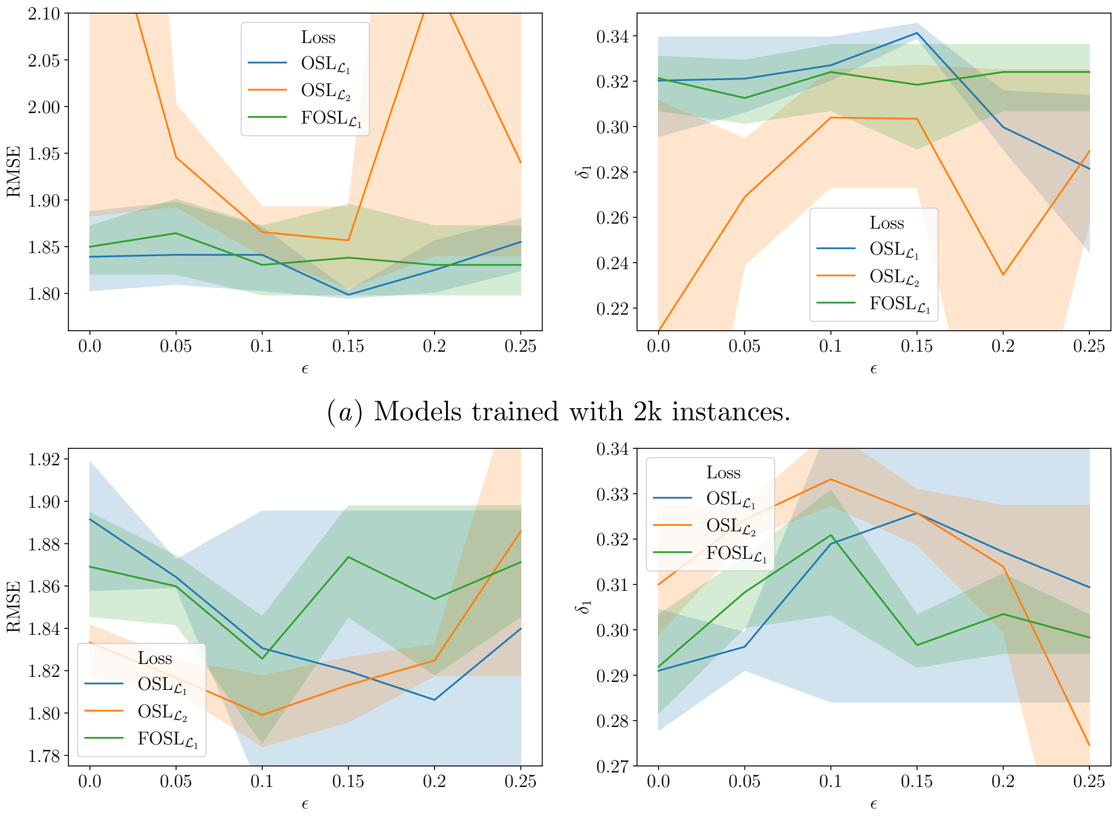
3.4. Hyperparameter Sensitivity

All superset losses employ a hyperparameter ϵ that determines the degree of imprecisiation involved in the training. Here, we assess the sensitivity of this parameter when training on *NYUD-v2* with either 2k or 10k subsamples in the same experimental setting as considered before. To this end, we fix ϵ to values in $[0, 0.25]$ and solely optimize the initial learning rate η on *iBims-1* as we did in the main experiments, and report the RMSE and δ_1 accuracy on *DIODE*. For $\text{FOSL}_{\mathcal{L}_1}$, we set the fuzzy set support parameter δ to 1. For statistical significance, we conducted each experiment five times.

As can be seen in Figure 2(*a*) for 2k instances, $\text{OSL}_{\mathcal{L}_1}$ benefits from higher degrees of imprecisiation towards $\epsilon = 0.15$ compared to simple $\mathcal{L}_1$ ($\epsilon = 0$). On the contrary, least squares optimization using $\text{OSL}_{\mathcal{L}_2}$ leads to worse performance with higher variance. In case of 10k training instances (Figure 2(*b*)), $\text{OSL}_{\mathcal{L}_2}$ provides the best performance for $\epsilon = 0.1$, whereas the variance of $\text{OSL}_{\mathcal{L}_1}$ increases with higher degrees of imprecisiation. In any case, $\text{FOSL}_{\mathcal{L}_1}$ shows relatively low sensitivity to the degree of imprecisiation, while providing competitive performance at the same time.

Table 4: Additional results on *DIODE* and the official *SunRGBD* test split (average over three runs). The best model is indicated in bold per number of instances and metric.

# Insts.	Loss	*DIODE*		*SunRGBD*				
		RMSLog (↓)	SQREL (↓)	$\log_{10}$ (↓)	δ_2 (↑)	δ_3 (↑)	RMSLog (↓)	SQREL (↓)
2k	$\mathcal{L}_2$	0.797 ± 0.195	1.221 ± 0.117	0.148 ± 0.007	**0.747** ± 0.002	**0.911** ± 0.005	0.472 ± 0.045	0.708 ± 0.055
	$\mathcal{L}_1$	0.577 ± 0.010	1.057 ± 0.056	0.146 ± 0.007	0.725 ± 0.026	0.896 ± 0.016	0.446 ± 0.020	0.718 ± 0.089
	$\mathcal{L}_{\text{Huber}}$	0.561 ± 0.010	1.064 ± 0.039	0.145 ± 0.007	0.730 ± 0.026	0.901 ± 0.016	0.442 ± 0.018	0.782 ± 0.089
	$\mathcal{L}_{\text{BerHu}}$	0.581 ± 0.026	1.030 ± 0.053	0.154 ± 0.002	0.712 ± 0.012	0.887 ± 0.009	0.499 ± 0.047	0.782 ± 0.054
	$\mathcal{L}_{\text{Ruber}}$	0.562 ± 0.013	**1.020** ± 0.036	0.150 ± 0.003	0.710 ± 0.010	0.888 ± 0.007	0.454 ± 0.007	0.798 ± 0.039
	$\mathcal{L}_{\text{Barron}}$	0.568 ± 0.013	1.101 ± 0.022	0.151 ± 0.006	0.709 ± 0.021	0.886 ± 0.015	0.457 ± 0.016	0.896 ± 0.110
	$\mathcal{L}_{\text{trim}}$	0.600 ± 0.026	1.215 ± 0.060	0.150 ± 0.011	0.746 ± 0.009	0.910 ± 0.007	0.477 ± 0.063	**0.680** ± 0.101
	$\mathcal{L}_{\text{ScaledSIError}}$	0.579 ± 0.009	1.053 ± 0.015	0.152 ± 0.003	0.706 ± 0.010	0.883 ± 0.008	0.458 ± 0.008	0.784 ± 0.036
	$\mathcal{L}_{\text{WeightedL2}}$	0.582 ± 0.018	1.107 ± 0.054	**0.143** ± 0.005	0.740 ± 0.018	0.905 ± 0.010	**0.441** ± 0.016	0.703 ± 0.078
	$\text{OSL}_{\mathcal{L}_1}$	**0.538** ± 0.018	1.054 ± 0.019	0.145 ± 0.004	0.723 ± 0.014	0.894 ± 0.010	0.453 ± 0.011	0.795 ± 0.040
	$\text{OSL}_{\mathcal{L}_2}$	0.541 ± 0.007	1.073 ± 0.063	0.144 ± 0.007	0.729 ± 0.026	0.902 ± 0.017	0.452 ± 0.019	0.796 ± 0.102
	$\text{FOSL}_{\mathcal{L}_1}$	0.565 ± 0.007	**1.020** ± 0.018	0.151 ± 0.003	0.708 ± 0.009	0.884 ± 0.005	0.459 ± 0.008	0.788 ± 0.041
Full	$\mathcal{L}_2$	0.535 ± 0.006	0.972 ± 0.035	0.152 ± 0.001	0.706 ± 0.003	0.885 ± 0.003	0.458 ± 0.003	0.862 ± 0.006
	$\mathcal{L}_1$	0.561 ± 0.010	0.998 ± 0.036	0.154 ± 0.002	0.698 ± 0.007	0.877 ± 0.004	0.464 ± 0.005	0.845 ± 0.044
	$\mathcal{L}_{\text{Huber}}$	0.533 ± 0.022	0.960 ± 0.023	0.156 ± 0.003	0.691 ± 0.011	0.873 ± 0.009	0.469 ± 0.008	0.906 ± 0.021
	$\mathcal{L}_{\text{BerHu}}$	0.551 ± 0.027	0.944 ± 0.045	0.158 ± 0.001	0.687 ± 0.004	0.869 ± 0.003	0.484 ± 0.015	0.927 ± 0.053
	$\mathcal{L}_{\text{Ruber}}$	0.521 ± 0.005	0.900 ± 0.022	0.158 ± 0.001	0.688 ± 0.003	0.868 ± 0.003	0.477 ± 0.003	0.940 ± 0.047
	$\mathcal{L}_{\text{Barron}}$	0.535 ± 0.006	0.979 ± 0.016	0.155 ± 0.001	0.696 ± 0.005	0.877 ± 0.003	0.466 ± 0.004	0.906 ± 0.038
	$\mathcal{L}_{\text{trim}}$	0.563 ± 0.021	1.032 ± 0.048	0.159 ± 0.008	0.693 ± 0.014	0.872 ± 0.011	0.505 ± 0.053	0.963 ± 0.125
	$\mathcal{L}_{\text{ScaledSIError}}$	0.519 ± 0.016	0.913 ± 0.027	0.158 ± 0.002	0.686 ± 0.006	0.867 ± 0.005	0.473 ± 0.004	0.868 ± 0.039
	$\mathcal{L}_{\text{WeightedL2}}$	0.540 ± 0.017	0.983 ± 0.047	**0.150** ± 0.001	**0.711** ± 0.005	0.888 ± 0.003	**0.455** ± 0.004	**0.839** ± 0.017
	$\text{OSL}_{\mathcal{L}_1}$	**0.486** ± 0.015	**0.826** ± 0.069	0.155 ± 0.002	0.702 ± 0.008	0.875 ± 0.005	0.460 ± 0.006	0.929 ± 0.033
	$\text{OSL}_{\mathcal{L}_2}$	0.507 ± 0.013	0.976 ± 0.114	0.152 ± 0.001	0.706 ± 0.005	**0.889** ± 0.004	0.456 ± 0.002	0.912 ± 0.041
	$\text{FOSL}_{\mathcal{L}_1}$	0.550 ± 0.023	0.934 ± 0.023	0.157 ± 0.001	0.694 ± 0.007	0.878 ± 0.003	0.466 ± 0.004	0.943 ± 0.056



(*a*) Models trained with 2k instances.

(*b*) Models trained with 10k instances.

Figure 2: RMSE and δ_1 with the respective standard deviations on the *DIODE* test data for models trained on either 2k or 10k instances from *NYUD-v2*.

3.5. Discussion

In our experiments, we consider various scenarios that may apply to real-life use cases. In most scenarios, either facing few training instances in a relatively uniform domain (*NYUD-v2* with 2k instances) or training data captured by heterogeneous depth sensors (*Sun-RGBD*), OSL$_{\mathcal{L}_1}$ represents an easy to optimize yet well performing method for robust depth regression. Nevertheless, OSL$_{\mathcal{L}_2}$ becomes more effective when observing more instances in the homogeneous sensor setting, whereas FOSL$_{\mathcal{L}_1}$ has shown promising performance in the high noise scenario. However, the increased expressiveness of complex fuzzy set-based superset modeling comes with a larger hyperparameter search space, making its optimization more resource demanding. This method could potentially benefit from more than 20 trials in the random search as conducted within our experiments, as well as from more sophisticated yet domain tailored fuzzy set modeling.

4. Exemplary Predictions

In Fig. 3, we provide sample predictions of selected baselines and our superset learning-based methods on *DIODE*. Here, we consider models trained on a subset of 2k instances from *NYUD-v2* with different degrees of noise injection. As can be seen, the robust superset learning-based losses keep providing reliable predictions even for high degrees of noise, for which $\mathcal{L}_2$-related losses fail.

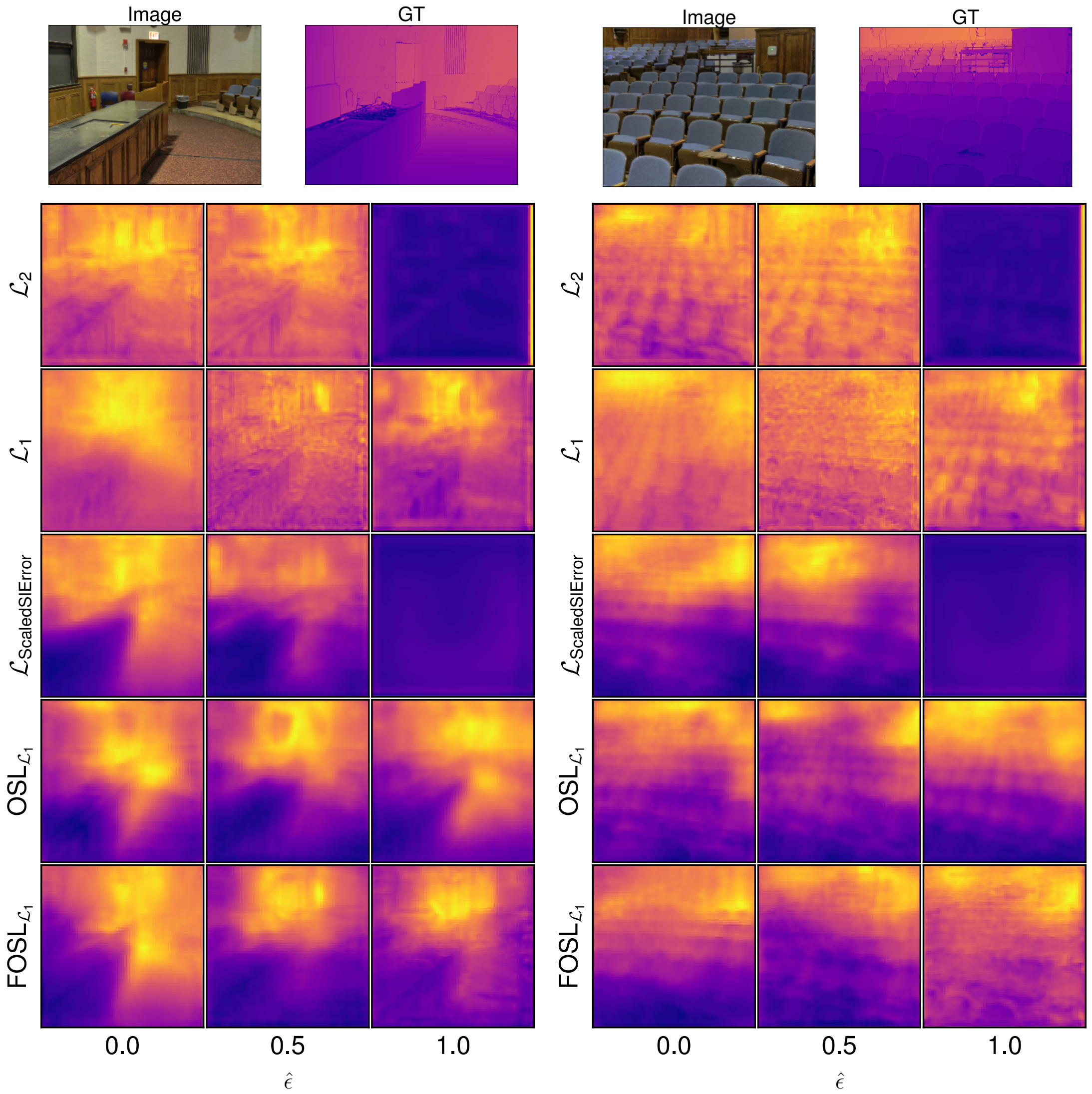


Figure 3: Exemplary predictions for two *DIODE* images of models trained on a subset of discussed loss functions. Here, we compare models trained with different noise levels $\hat{\epsilon}$ injected into the training data as discussed in the paper.

References

Jonathan T. Barron. A general and adaptive robust loss function. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 4331–4339. Computer Vision Foundation / IEEE, 2019.

Go Irie, Takahito Kawanishi, and Kunio Kashino. Robust learning for deep monocular depth estimation. In *Proc. of the IEEE International Conference on Image Processing, ICIP*, pages 964–968. IEEE, 2019.

Tobias Koch, Lukas Liebel, Friedrich Fraundorfer, and Marco Körner. Evaluation of CNN-based single-image depth estimation methods. In *Proc. of the European Conference on Computer Vision, ECCV, Workshops Part III*, volume 11131 of *LNCS*, pages 331–348. Springer, 2018.

Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *Proc. of the 4th International Conference on 3D Vision, 3DV*, pages 239–248. IEEE Computer Society, 2016.

Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *CoRR*, abs/1907.10326, 2019.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *Proc. of the 5th International Conference on Learning Representations, ICLR*. OpenReview.net, 2017.

René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *Proc. of the 12th European Conference on Computer Vision, ECCV, Part V*, volume 7576 of *LNCS*, pages 746–760. Springer, 2012.

Shuran Song, Samuel P. Lichtenberg, and Jianxiong Xiao. SUN RGB-D: A RGB-D scene understanding benchmark suite. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 567–576. IEEE Computer Society, 2015.

Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946, 2019.

F

Appendix to Monocular Depth Estimation via Listwise Ranking using the Plackett-Luce Model

– Supplementary Material –
Monocular Depth Estimation via Listwise Ranking using the Plackett-Luce Model

1. Benchmark Dataset Characteristics

Table 1 shows the characteristics of the benchmark datasets, including their respective maximum depth values (capacity). We use these values to limit the recognized depth in the images as described in [9]. These values were also used to normalize the ground truth depths to get metric errors on a similar scale for all datasets.

Table 1. Properties of datasets being used within the zero-shot cross-dataset evaluation.

Dataset	# Test Instances	Sensor Type	Diversity	Depth Capacity (in m)
Ibims [5]	100	Laser	Indoor	50
Sintel [2]	1064	Synthetic	Animated	72
DIODE [12]	771	Laser	Indoor, outdoor	350
TUM [10]	1815	Motion Parallax	Indoor	10

Since the depth annotations of Sintel [2] are given in the inverse depth space, we did not apply the transformation $\frac{1}{D[l]+1}$ to calculate the nDCG scores as described in our paper (cf. Section 4.3). Instead, we directly used the already inverted depth values. Furthermore, in the case of TUM, we used the preprocessed dataset version as provided by the authors of [7]. Thereby, we considered the motion parallax (so-called “Plane-Plus-Parallax”) depth map, which was constructed based on flow predictions between multiple views. We found these depth signals to provide a more precise and reliable source for calculating our error metrics compared to the originally provided Kinect sensor values.

2. Baseline Characteristics

Table 2 gives an overview of all baseline models considered within our empirical evaluation, together with the data being used for training. The diversity of the training data categorizes the individual datasets in terms of the variety of their captured scenes. MC represents a special case due to only incorporating images showing humans, but also indoor and outdoor.

Table 2. Baselines together with their training data considered within our empirical study.

Model	Loss Class	Training set	# Examples	Training Data Diversity
DenseDepth [1]	(Scale-invariant) Regression	NYU	50k	Low
MegaDepth [8]		MegaDepth	626k	High
BTS [6]		NYU	24k	Low
MC [7]		MC	136k	Medium
MiDaS [9]		HR-WSI	20k	High
MonoDepth2 [4]	Self-Sup.	KITTI	40k	Low
YouTube3D [3]	Relative	RW+DIW+YT3D	1219k	High
Xian 2020 [13]		HR-WSI	20k	High

3. Experimental Details

For the loss comparison (cf. Section 4.4.1), we compared our model on the ResNet-based architecture (PLDepthResNet) to the scale-invariant regression [9] and pairwise ranking [13] approach. Thereby, we optimized all models for 50 epochs with Adam and a batch size of 40 on four Nvidia Titan RTX. For the scale-invariant regression and our PL model, we used an initial learning rate of 0.001 multiplied by $\sqrt{0.1}$ after 25 epochs, while the pairwise ranking approach used an initial learning rate of 0.01 with the same learning rate schedule. In all three cases, input images were resized to 448×448 and data has been augmented by horizontally flipping with a 50% chance.

In the second experimental study, where we trained our model on the proposed EfficientNet-based architecture (PLDepthEffNet, cf. Fig. 3), we used a smaller initial learning rate of 0.0001 with the other parameters kepth the same. For each image, we sampled 100 rankings of size 5 per epoch. We further evaluated the scale-invariant regression variant on the same model architecture, where we used the same hyperparameters as for PLDepthEffNet, but with a higher learning rate of 0.001. The learning rate schedule was kept the same as for the previous experiments.

4. Additional Experiments on HR-WSI

As an additional experiment, we report the ordinal error, nDCG, RMSE and $\delta > 1.25$ as specified in the paper on the dataset HR-WSI [13]. Here, we compare only models being trained on this dataset, namely our PL model, MiDaS and Xian 2020. The presented results were computed on the separate validation set of 400 instances, which was used for model optimization of the mentioned approaches. Thereby, we consider the same trained models as used for the ordinal and metric error calculation in Section 4.4.2 and 4.4.3 of our paper.

Table 3 shows the averaged results for three runs. As can be seen, our model is superior with regard to most of the metrics. Only for the nDCG, the scale-invariant regression variant MiDaS turns out to be superior, although only slightly. Fig. 1 further shows exemplary predictions on HR-WSI.

Table 3. Results on the validation split of HR-WSI for the models being trained on the corresponding training split. The same experimental settings as for the model comparison apply here. ↓ refers to “lower is better”, while ↑ denotes the opposite.

Model	Ord. Err. (↓)	nDCG (↑)	RMSE (↓)	$\delta > 1.25$ (↓)
MiDaS [9]	0.192	**0.839**	0.088	0.294
Xian 2020 [13]	0.166	0.838	0.154	0.558
PLDepthEffNet	**0.164**	0.837	**0.069**	**0.192**

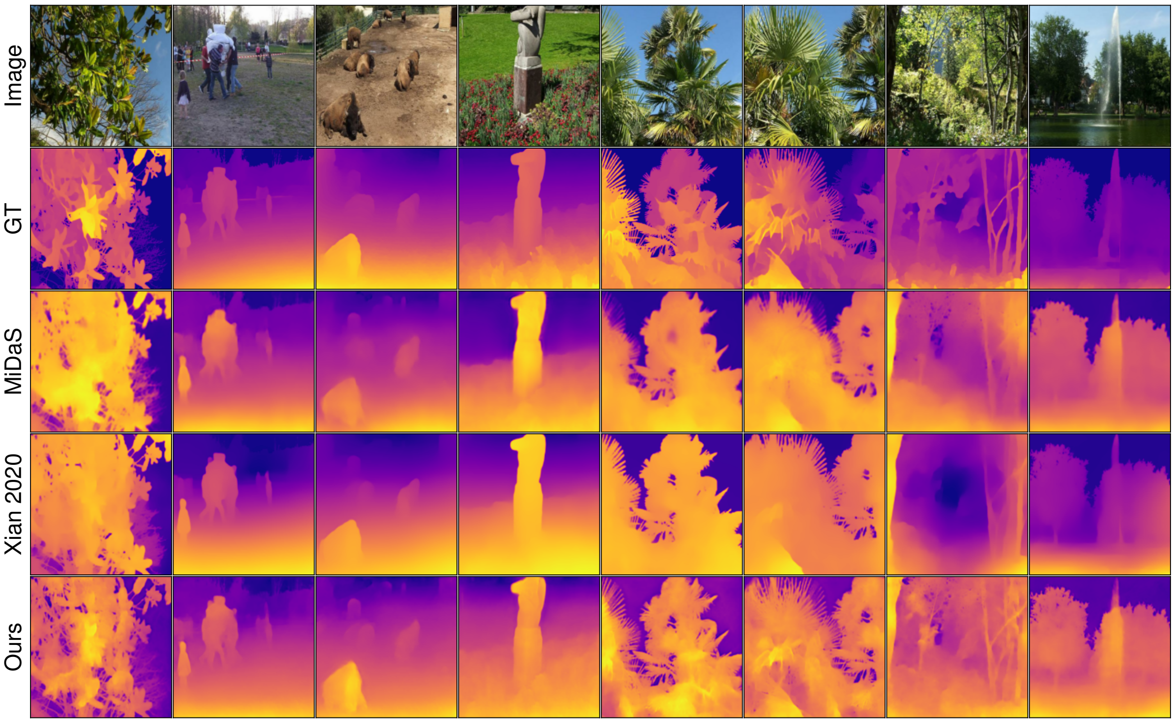


Figure 1. Exemplary predictions of the models trained on HR-WSI for validation set samples.

5. Additional Model Predictions

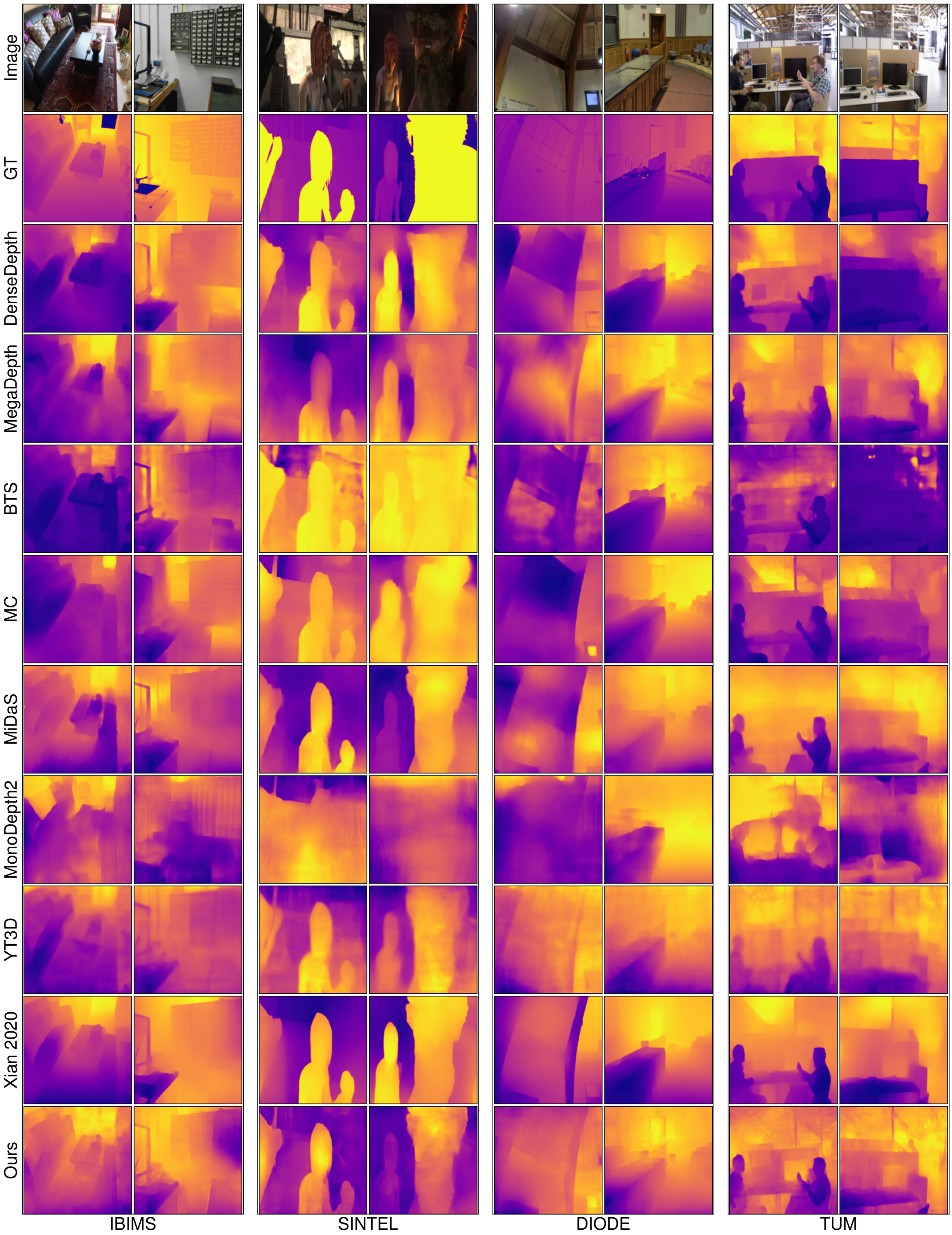


Figure 2. Model predictions for samples of the benchmark datasets (rescaled and shifted acc. to the ground truth depth values).

6. PLDepthEffNet Model Architecture

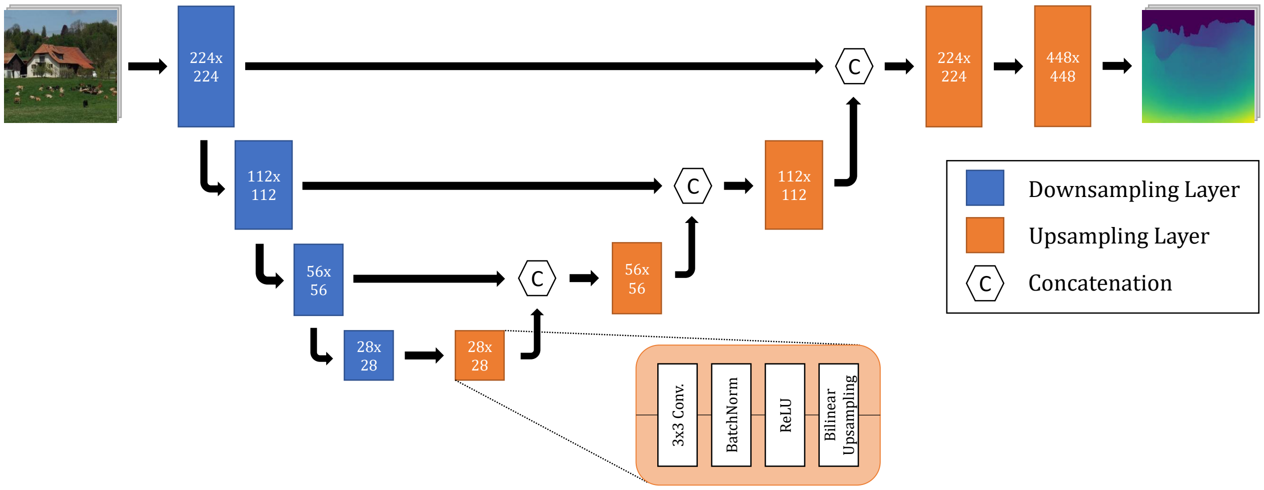


Figure 3. PLDepthEffNet U-net model architecture as proposed in the paper. The blue downsampling layers are specified by the used EfficientNet [11] backbone. The layer captions specify the corresponding output dimensionality of the respective layers.

References

[1] Ibraheem Alhashim and Peter Wonka. High quality monocular depth estimation via transfer learning. *CoRR*, abs/1812.11941, 2018.

[2] Daniel J. Butler, Jonas Wulff, Garrett B. Stanley, and Michael J. Black. A naturalistic open source movie for optical flow evaluation. In Andrew W. Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Proceedings of the 12th European Conference on Computer Vision (ECCV), Part VI, October 7-13, 2012, Florence, Italy*, volume 7577 of *Lecture Notes in Computer Science*, pages 611–625. Springer, 2012.

[3] Weifeng Chen, Shengyi Qian, and Jia Deng. Learning single-image depth from videos using quality assessment networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 16-20, 2019, Long Beach, CA, USA*, pages 5604–5613. Computer Vision Foundation / IEEE, 2019.

[4] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J. Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October 27 - November 2, 2019, Seoul, Korea (South)*, pages 3827–3837. IEEE, 2019.

[5] Tobias Koch, Lukas Liebel, Friedrich Fraundorfer, and Marco Körner. Evaluation of CNN-based single-image depth estimation methods. In Laura Leal-Taixé and Stefan Roth, editors, *Proceedings of the European Conference on Computer Vision (ECCV) Workshops, Part III, September 8-14, 2018, Munich, Germany*, volume 11131 of *Lecture Notes in Computer Science*, pages 331–348. Springer, 2018.

[6] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar guidance for monocular depth estimation. *CoRR*, abs/1907.10326, 2019.

[7] Zhengqi Li, Tali Dekel, Forrester Cole, Richard Tucker, Noah Snavely, Ce Liu, and William T. Freeman. Learning the depths of moving people by watching frozen people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 16-20, 2019, Long Beach, CA, USA*, pages 4521–4530. Computer Vision Foundation / IEEE, 2019.

[8] Zhengqi Li and Noah Snavely. MegaDepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 18-22, 2018, Salt Lake City, UT, USA*, pages 2041–2050. IEEE Computer Society, 2018.

[9] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[10] Jürgen Sturm, Nikolas Engelhard, Felix Endres, Wolfram Burgard, and Daniel Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), October 7-12, 2012, Vilamoura, Algarve, Portugal*, pages 573–580. IEEE, 2012.

[11] Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning (ICML), 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 6105–6114. PMLR, 2019.

[12] Igor Vasiljevic, Nicholas I. Kolkin, Shanyi Zhang, Ruotian Luo, Haochen Wang, Falcon Z. Dai, Andrea F. Daniele, Mohammadreza Mostajabi, Steven Basart, Matthew R. Walter, and Gregory Shakhnarovich. DIODE: A dense indoor and outdoor depth dataset. *CoRR*, abs/1908.00463, 2019.

[13] Ke Xian, Jianming Zhang, Oliver Wang, Long Mai, Zhe Lin, and Zhiguo Cao. Structure-guided ranking loss for single image depth prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 13-19, 2020, Seattle, WA, USA*, pages 608–617. IEEE, 2020.

List of Source Code Repositories

In the following, we list hyperlinks to all relevant source code repositories for the respective contributions, together with their respective license.

Chapter 4: From Label Smoothing to Label Relaxation

`https://github.com/julilien/LabelRelaxation` (Apache 2.0 license)

Chapter 5: Credal Self-Supervised Learning

`https://github.com/julilien/CSSL` (Apache 2.0 license)

Chapter 6: Conformal Credal Self-Supervised Learning

`https://github.com/julilien/C2S2L` (Apache 2.0 license)

Chapter 7: Mitigating Label Noise through Data Ambiguation

`https://github.com/julilien/MitigatingLabelNoiseDataAmbiguation` (Apache 2.0 license)

Chapter 8: Instance Weighting through Data Imprecisiation

`https://github.com/julilien/InstanceWeightingDataImprecisiation` (Apache 2.0 license)

Chapter 9: Robust Regression for Monocular Depth Estimation

`https://github.com/julilien/RobustMDE` (Apache 2.0 license)

Chapter 10: Monocular Depth Estimation via Listwise Ranking using the Plackett-Luce Model

`https://github.com/julilien/PLDepth` (Apache 2.0 license)

List of Figures

Colophon

This thesis was typeset with $\LaTeX\,2_{\varepsilon}$. It uses the *Clean Thesis* style developed by Ricardo Langner. The design of the *Clean Thesis* style is inspired by user guide documents from Apple Inc.

Download the *Clean Thesis* style at `http://cleanthesis.der-ric.de/`.

Declaration

I hereby declare that this dissertation is my original work, composed independently, and without the use of any unauthorized materials and additional, non-indicated help. All sources and references utilized are properly acknowledged and cited. This dissertation has not been previously submitted to any other faculty or institution. Furthermore, I confirm that I have not undergone an unsuccessful doctoral examination, nor have I been stripped of any previously earned doctoral degrees.

Paderborn, July 31st, 2023

Julian Lienen