Well-Calibrated Predictive Uncertainty in Medical Imaging with Bayesian Deep Learning

Kalibrierte prädiktive Unsicherheit in der medizinischen Bildgebung mit Bayesian Deep Learning

Von der Fakultät für Maschinenbau
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des akademischen Grades
Doktor-Ingenieur (Dr.-Ing.)
genehmigte

Dissertation

von
Max-Heinrich Viktor Laves

2021

1. Referent: Prof. Dr.-Ing. Tobias Ortmaier
2. Referent: Prof. Dr.-Ing. Eduard Reithmeier

Tag der Promotion: 13. Dezember 2021

Vorwort und Danksagung

Diese Arbeit ist während meiner Tätigkeit als wissenschaftlicher Mitarbeiter am Institut für Mechatronische Systeme der Gottfried Wilhelm Leibniz Universität Hannover entstanden. In dieser Zeit habe ich mehr gelernt, als im Studium zuvor und bin dafür überaus dankbar. Ich hätte nicht gedacht, dass sich mein Interessengebiet nochmal so verschieben würde.

Zunächst möchte ich mich bei allen Kolleginnen und Kollegen bedanken, die mich in meinen fünf Jahren am imes begleitet haben. Das familiäre Arbeitsklima, auch an der Espresso-Maschine, habe ich sehr genossen. Im Besonderen bedanke ich mich bei meiner Arbeitsgruppe „Medizintechnik und Bildverarbeitung“, namentlich ohne besondere Reihenfolge vertreten durch Jan Bergmeier (mein langjähriger Bürokollege), Jacob Fast, Sontje Ihler, Andreas Schoob, Dennis Kundrat, Jorge Badilla, Katrin Nülle, Samuel Müller, Shivaraman Ilango und Johannes Gaa. Darüber hinaus bedanke ich mich bei meinem Gruppenleiter Lüder Kahrs, der mich seit meiner Masterarbeit am imes unterstützt hat und dessen Tür stets offen stand. Rückblickend freue ich mich sehr, dass er mich nicht davon abgehalten hat, mich „mal mit diesem Machine Learning“ zu beschäftigen. Meinem Doktorvater Professor Tobias Ortmaier gilt besonderer Dank, da er mir die Möglichkeit gegeben hat, meine Promotion unter seiner Leitung durchzuführen. Nicht zu vergessen sind die Studentinnen und Studenten, die ich in meiner Zeit am imes betreut habe und die mich bei meiner Forschung unterstützt haben. Namentlich bedanke ich mich bei Jens Bicker, Christian van Kempen, Malkin Gerchow und Malte Tölle, dessen Arbeiten zu hochwertigen Publikationen geführt haben.

Abschließend gilt mein größter Dank meiner Familie und insbesondere meinen Eltern Sigrid und Henning, die mir die Möglichkeit gegeben haben, ein Studium zu absolvieren und mich mit allen Mitteln unterstützt haben. Ich danke auch meiner Frau Julia, die trotz der großen Distanz zwischen unseren Arbeitsorten viel Geduld, Verständnis und Unterstützung gezeigt hat. Ohne Euch hätte ich es nicht geschafft.

Vielen Dank!

Hannover, September 2021 *Max-Heinrich Laves*

Abstract

The use of medical imaging has revolutionized modern medicine over the last century. It has helped provide insight into human anatomy and physiology. Many diseases and pathologies can only be diagnosed with the use of imaging techniques. Due to increasing availability and the reduction of costs, the number of medical imaging examinations is continuously growing, resulting in a huge amount of data that has to be assessed by medical experts.

Computers can be used to assist in and automate the process of medical image analysis. Recent advances in deep learning allow this to be done with reasonable accuracy and on a large scale. The biggest disadvantage of these methods in practice is their black-box nature. Although they achieve the highest accuracy, their acceptance in clinical practice may be limited by their lack of interpretability and transparency. These concerns are reinforced by the core problem that this dissertation addresses: the overconfidence of deep models in incorrect predictions. How do we know if we do not know?

This thesis deals with Bayesian methods for estimation of predictive uncertainty in medical imaging with deep learning. We show that the uncertainty from variational Bayesian inference is miscalibrated and does not represent the predictive error well. To quantify miscalibration, we propose the *uncertainty calibration error*, which alleviates disadvantages of existing calibration metrics. Moreover, we introduce *logit scaling* for deep Bayesian Monte Carlo methods to calibrate uncertainty after training. Calibrated deep Bayesian models better detect false predictions and out-of-distribution data.

Bayesian uncertainty is further leveraged to reduce the economic burden of large data labeling, which is needed to train deep models. We propose *BatchPL*, a sample acquisition scheme that selects highly informative samples for pseudo-labeling in self- and unsupervised learning scenarios. The approach achieves state-of-the-art performance on both medical and non-medical classification data sets.

Many medical imaging problems exceed classification. Therefore, we extended estimation and calibration of predictive uncertainty to deep regression (σ scaling) and evaluated it on different medical imaging regression tasks. To mitigate the problem of hallucinations in deep generative models, we provide a Bayesian approach to deep image prior (*MCDIP*), which is not affected by hallucinations as the model only ever has access to one single image.

Keywords: machine learning, deep learning, medical imaging, uncertainty, variational inference, Bayesian inference, computer-aided diagnosis, instrument tracking, reconstruction

Kurzfassung

Die moderne Medizin wurde durch den Einsatz der medizinischen Bildgebung im letzten Jahrhundert revolutioniert. Viele Krankheiten können nur mit Hilfe von bildgebenden Verfahren diagnostiziert werden. Aufgrund der zunehmenden Verfügbarkeit und der sinkenden Kosten nimmt die Zahl der Untersuchungen mit medizinischer Bildgebung stetig zu, was zu einer riesigen Datenmenge führt, die durch medizinische Experten ausgewertet werden muss.

Computer können zur Unterstützung und Automatisierung der medizinischen Bildanalyse eingesetzt werden. Die Fortschritte im Bereich des Deep Learning ermöglichen dies mit angemessener Genauigkeit und bei hoher Datenmenge. Der größte Nachteil dieser Methoden ist ihr Blackbox-Charakter. Obwohl sie die höchste Genauigkeit erreichen, kann ihre Akzeptanz in der klinischen Praxis durch ihre mangelnde Interpretierbarkeit und Transparenz eingeschränkt sein. Diese Bedenken werden durch das Kernproblem verstärkt, mit dem sich diese Dissertation befasst: die übermäßige Konfidenz der tiefen Modelle in falsche Prädiktionen. Wie können wir wissen, wenn wenn die Modelle etwas nicht wissen?

Diese Arbeit befasst sich mit Bayesschen Methoden zur Schätzung der Unsicherheit in der medizinischen Bildgebung mit Deep Learning. Wir zeigen, dass die Unsicherheit aus Bayesscher Variationsinferenz falsch kalibriert ist und den Vorhersagefehler nicht gut repräsentiert. Um die Miskalibrierung zu quantifizieren, schlagen wir den *Uncertainty Calibration Error* vor, der Nachteile bestehender Kalibrierungsmetriken vermeidet. Außerdem führen wir eine *Logit-Skalierung* für tiefe Bayessche Monte-Carlo-Methoden ein, um die Unsicherheit nach dem Training zu kalibrieren. Kalibrierte Bayessche Modelle erkennen falsche Vorhersagen und Daten, die nicht der Trainingsverteilung entsprechen, besser.

Die Bayessche Unsicherheit wird außerdem genutzt, um den Aufwand der Annotierung großer Datenmengen zu verringern. Wir präsentieren *BatchPL*, ein Schema, das hochinformative Stichproben für eine Pseudoannotierung für selbstüberwachtes Lernen auswählt. Der Ansatz erreicht sowohl bei medizinischen als auch bei nicht-medizinischen Klassifizierungsdatensätzen eine Spitzenleistung.

Viele Probleme der medizinischen Bildgebung gehen über Klassifizierung hinaus. Daher haben wir die Schätzung und Kalibrierung der Unsicherheit auf die tiefe Regression (*σ-Skalierung*) ausgedehnt und sie an verschiedenen medizinischen Regressionsaufgaben evaluiert. Um das Problem der Halluzinationen in tiefen generativen Modellen zu entschärfen, präsentieren wir einen Bayesschen Ansatz (*MCDIP*), der nicht durch Halluzinationen beeinträchtigt wird, da das Modell immer nur Zugang zu einem einzigen Bild hat.

Stichworte: maschinelles Lernen, Deep Learning, medizinische Bildgebung, Unsicherheit, Variationsinferenz, Bayessche Inferenz, computerassistierte Diagnose, Instrumententracking, Rekonstruktion

Contents

Nomenclature ix

1 Introduction 1
1.1 The Importance of Uncertainty in Medical Decision Making 1
1.2 Deep Learning in Medical Imaging 6
1.2.1 Classification 7
1.2.2 Regression 8
1.2.3 Generative Tasks 8
1.3 Supervised Learning and Maximum Likelihood 9
1.4 Bayesian Deep Learning and Variational Inference 11
1.5 Hypotheses and Contributions 16
1.6 Thesis Structure 17

2 Computer-Aided Diagnosis 19
2.1 Calibration of Uncertainty for Variational Inference 19
2.1.1 Introduction 20
2.1.2 Related Work 22
2.1.3 Methods 22
2.1.4 Experiments 34
2.1.5 Results 36
2.1.6 Conclusion 45
2.2 BatchPL: An Efficient Sample Acquisition Scheme for Pseudo-Labeling in Self-Supervised Learning 47
2.2.1 Introduction 47
2.2.2 Related Work 48
2.2.3 Methods 49
2.2.4 Experiments 53
2.2.5 Conclusion 55
2.3 Chapter Conclusion 55

3 Regression in Medical Imaging 57
3.1 Introduction 57
3.2 Methods 59
3.2.1 Conditional Log-Likelihood for Regression 60
3.2.2 Biased estimation of σ 61
3.2.3 σ Scaling for Aleatoric Uncertainty 62

3.2.4 Laplacian Model . . . 62
3.2.5 Well-Calibrated Estimation of Predictive Uncertainty . . . 63
3.2.6 Unbiased Estimator of the Approximate Predictive Variance . . . 63
3.2.7 Expected Uncertainty Calibration Error for Regression . . . 64
3.3 Experiments . . . 65
3.3.1 Training Procedure . . . 66
3.3.2 3D OCT Needle Pose Data Set . . . 67
3.4 Results . . . 68
3.4.1 Posterior Prediction Intervals . . . 69
3.4.2 Detection of Out-of-Distribution Data and Unreliable Predictions 71
3.4.3 Ablation Study on Auxiliary Model Scaling . . . 72
3.5 Chapter Conclusion . . . 72

4 Generative Models in Medical Imaging 75
4.1 Medical Image Denoising with Bayesian Deep Image Prior . . . 75
4.1.1 Introduction . . . 76
4.1.2 Related Work . . . 77
4.1.3 Methods . . . 78
4.1.4 Experiments . . . 80
4.1.5 Results . . . 83
4.1.6 Conclusion . . . 86
4.2 Deformable Medical Image Registration Using a Randomly-Initialized CNN as Regularization Prior . . . 87
4.2.1 Introduction . . . 87
4.2.2 Methods . . . 88
4.3 Results & Conclusion . . . 89
4.3.1 Semantically Regularized Denoising Autoencoders . . . 91
4.3.2 Purpose . . . 91
4.3.3 Methods . . . 92
4.3.4 Results . . . 93
4.3.5 Conclusion . . . 95
4.4 Chapter Conclusion . . . 95

5 Conclusion and Perspectives 97
5.1 Outlook . . . 98

Bibliography 101

A Appendix 116
A.1 Calibration of Uncertainty for Variational Inference . . . 116
A.1.1 Additional Results & Figures . . . 116

- A.2 Regression 125
  - A.2.1 Additional Results and Calibration Diagrams 125
  - A.2.2 Additional Prediction Intervals 130
- A.3 Medical Image Denoising with Bayesian Deep Image Prior 131
  - A.3.1 Additional Figures 131
  - A.3.2 Additional Tables 134

Curriculum Vitae **135**

Nomenclature

Roman Symbols

a	Scalar or random variable
$\boldsymbol{a}$	Vector or vector-valued r.v.
$\boldsymbol{A}$	Matrix or matrix-valued r.v.
$\boldsymbol{I}_n$	Identity matrix with n rows and n columns
$\mathbb{A}$	Set
$\mathbb{R}$	Set of real numbers
$\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$	Multivariate Gaussian distribution over $\boldsymbol{x}$ with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$
$\mathcal{U}(a, b)$	Continuous uniform distribution over the interval $[a, b]$
$p_X(x)$	Probability distribution function of r.v. X at x
$\Pr(A)$	Probability of event A
$\Pr(A \mid B)$	Conditional probability of event A given event B
$a \sim p$	Random variable a with distribution p
$\mathbb{E}_{x \sim p}[f(x)]$	Expectation of $f(x)$ with respect to $p(x)$
$\mathrm{Var}_{x \sim p}[f(x)]$	Variance of $f(x)$ under $p(x)$
$\mathrm{KL}\left[\, p \,\|\, q \,\right]$	Kullback-Leibler divergence of p and q
$\log$	Natural logarithm (base e)
$\exp$	Exponential function

Greek Symbols

η	Learn rate

Other Symbols

§	Section sign (used to abbreviate section, subsection, paragraph, etc.)
$\odot$	Hadamard product (element-wise multiplication of vectors or matrices)

$\neg$	Logical negation (as in probability $\Pr(A \mid \neg B)$ of event A given not B)
$\|\boldsymbol{x}\|^2$	Squared 2-norm of vector $\boldsymbol{x}$

Abbreviations

r.v.	Random variable
MC	Monte Carlo
MCMC	Markov chain Monte Carlo
VI	Variational inference
ELBO	(log) evidence lower bound
SGD	Stochastic gradient descent
CNN	Convolutional neural network
BBB	Bayes by backprop
SSL	Semi-supervised learning
i.i.d.	Independent and identically distributed
w.r.t.	With respect to
cf.	confer (“compare”)
DIP	Deep image prior
NLL	Negative log-likelihood

1 Introduction

> Your overconfidence is your weakness.
>
> *Luke Skywalker*, to the Emperor
> Return of the Jedi

1.1 The Importance of Uncertainty in Medical Decision Making

In their daily routine, clinicians have to make crucial decisions that determine the health of patients. They often ask themselves “What are the possible causes of my patient’s problem?”. Features of the problem are gathered through different approaches, such as the medical interview, physical examination, or medical imaging. This information is usually indicative of different diagnoses. Clinicians can express their assumptions about various possible diagnoses with a degree of uncertainty. More specifically, their assumptions about the causes of the patient’s problems are not based solely on the information from diagnostic tools. They usually have a *prior* assumption based on their personal experience, even before seeing any test results. The test results themselves can be ambiguous and, therefore, have their own degree of uncertainty (expressed via test *sensitivity* and *specificity*), which contributes to the uncertainty in the final diagnosis.

Formally, clinicians express their prior belief with a probability distribution (Roberts 2020)

$$p(\mathsf{disease}) \ .$$

The prior assigns a non-zero probability to possible diseases and zero probability to impossible diseases (e.g., it is unlikely that chest pain is caused by a broken toe). In the worst case, we cannot make any prior assumption and use a noninformative prior $p(\mathsf{diagnosis}) = \mathrm{const}$, making all diagnoses equally probable *a priori*. After observing a test result, clinicians update their assumptions using Bayes’ theorem (Roberts 2020)

$$p(\mathsf{disease} \,|\, \mathsf{result}) = \frac{p(\mathsf{result} \,|\, \mathsf{disease}) p(\mathsf{disease})}{p(\mathsf{result})} \ . \tag{1.1}$$

This is called the *posterior probability* and it allows us to express the uncertainty in a diagnosis after observing a test result. The conditional distribution on the right-hand side is called *likelihood* and describes the probability of the observed test result for different diagnoses. In the medical context the prior is also referred to as *prevalence*. The denominator ensures that the posterior is a valid probability density and is called *evidence*. We will later see that the evidence plays an important role in approximate Bayesian inference (see § 1.4).

Let us consider the following example of how Bayes' theorem can be used to update the diagnostic uncertainty of clinicians (Held and Sabanés Bové 2014). A diagnostic test for a certain disease D has a sensitivity of 90 %. That is, given that the patient has the disease, the probability of a positive test result T is $\Pr(T \mid D) = 0.9$. Further, the test has a 90 % specificity (receiving a negative test result $\neg T$ given a disease-free patient), which we denote by $\Pr(\neg T \mid \neg D) = 0.9$. Assuming a prevalence $\Pr(D) = 0.01$ for this disease (that is, one out of hundred people in a population has this disease), the posterior probability is given by

$$\Pr(D \mid T) = \frac{\Pr(T \mid D)\Pr(D)}{\Pr(T)} \ .$$

In this simple example, we can use the law of total probability to compute the test evidence

$$\Pr(T) = \Pr(T \mid D)\Pr(D) + \Pr(T \mid \neg D)\Pr(\neg D) \ ,$$

with $\Pr(T \mid \neg D) = 1 - \Pr(\neg T \mid \neg D)$ and $\Pr(\neg D) = 1 - \Pr(D)$. Hence, the posterior probability is $\Pr(D \mid T) \approx 0.083$; i.e., the clinician's *confidence* for this disease increases from 1.0 % to 8.3 % (Held and Sabanés Bové 2014). This example illustrates the importance of priors in medical decision making. Without the use of Bayesian inference, a clinician may spuriously come to an *over-confident* conclusion and assume that the probability of having the disease after observing a positive test result would be 90 %.

This example may seem artificial, but on 9th November 2020, Pfizer and BioNTech published a press release in which the overwhelming efficacy of their COVID-19 vaccine was shown using Bayesian analysis with a weakly informative $\mathsf{Beta}(0.700102, 1.0)$ prior (Polack et al. 2020).

In the above example we have computed the posterior probability for certain realizations of simple discrete random variables, which we denoted by $\Pr(\cdot)$. Bayesian inference generally aims at computing the *posterior distribution*

$$p(\theta \mid x) = \frac{p(x \mid \theta)p(\theta)}{\int p(x \mid \theta)p(\theta)\,\mathrm{d}\theta} \tag{1.2}$$

that captures all the information about an unknown parameter θ given observed data x. In this case, $p(\theta \mid x)$ denotes a probability density function and obtaining the evidence requires integration with respect to θ, which usually cannot be done analytically. From the integral in Eq. (1.2) we can see why the evidence is also called *marginal likelihood*. Analytical marginalization is possible for simple linear models and carefully crafted priors that are *conjugate* to the likelihood (cf. (Bishop 2006; Held and Sabanés Bové 2014) for more information about conjugate priors). For more complex models, e.g., neural networks with non-linear activations, the true posterior distribution is intractable; i.e., the integral has no closed-form solution (Blei et al. 2017). Thus, we cannot reason about the uncertainty of θ after observing data (e.g., a test result). Instead, it is tempting to reject the Bayesian treatment

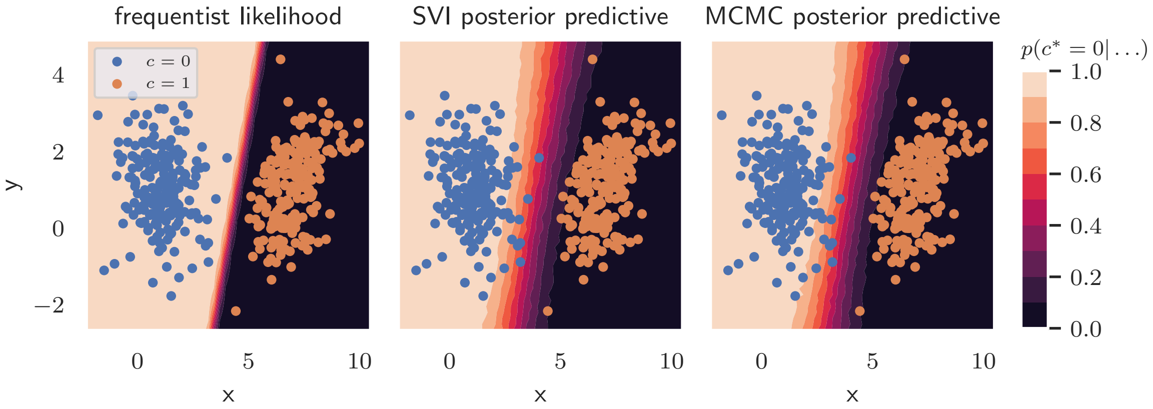


Figure 1.1: Samples from a linearly separable toy data set and the classification results for $c = 0$ from frequentist likelihood and Bayesian approximation with stochastic variational inference and Markov chain Monte Carlo (MCMC). The frequentist treatment leads to over-confident classification.

and try to estimate a single "best" point estimate for the unknown parameter. Clinicians would use the diagnosis that most likely led to the observed test results, which is referred to as the *maximum likelihood* solution:

$$\hat{\theta} = \arg\max_{\theta} \; p(x \,|\, \theta) \; .$$

This not only prevents the determination of uncertainty, but can also lead to an excessive valuation of the test results, as the following example shows.

Given an artificial data set[1] $\mathcal{D} = \{\mathcal{X}, \mathcal{C}\}$ of N linearly separable samples (see Fig. 1.1), where every sample $\boldsymbol{x}_i \in \mathcal{X}$ contains two continuous disease features (e.g., body temperature and diastolic blood pressure) $\boldsymbol{x}_i = [x_i, y_i]^\mathsf{T}$, $x, y \in \mathbb{R}$ and a corresponding binary disease label $C_i \in \mathcal{C}$, where $C_i = c_i, c \in \{0, 1\}$. We aim at finding a linear model

$$\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_i) = \boldsymbol{w}\boldsymbol{x}_i + \boldsymbol{b} \; , \tag{1.3}$$

with parameters $\boldsymbol{\theta} = \{\boldsymbol{w}, \boldsymbol{b}\}$ consisting of a weight matrix $\boldsymbol{w}$ and a bias vector $\boldsymbol{b}$. The model produces *logits* (unnormalized log-odds) for every input $\boldsymbol{x}_i$. Class probabilities are given by a softmax likelihood (cf. § 1.3)

$$p(c = d \,|\, \boldsymbol{x}, \boldsymbol{\theta}) = \frac{\exp\{\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x})_d\}}{\sum_j \exp\{\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x})_j\}} \; . \tag{1.4}$$

We implement the model using a single linear layer neural network and find the parameters $\boldsymbol{\theta}$ using three different approaches. First, a frequentist approach is taken and a point estimate $\hat{\boldsymbol{\theta}}$

---

[1] The code for this example can be found at gist.github.com/mlaves.

is found via maximum likelihood estimation (MLE) using stochastic gradient descent w.r.t. $\boldsymbol{\theta}$ on the data set. The point estimate $\hat{\boldsymbol{\theta}}$ is then plugged into Eq. (1.4) to obtain the softmax likelihood for a new input $\boldsymbol{x}^*$. When evaluating the frequentist likelihood over the input space, we observe severe over-fitting of the training data (see Fig. 1.1 left). The softmax output degenerates to a step function assigning a probability $p(c = 0 \,|\, \boldsymbol{x}, \hat{\boldsymbol{\theta}}) = 1$ to every training sample of class 0 and $p(c = 0 \,|\, \boldsymbol{x}, \hat{\boldsymbol{\theta}}) = 0$ for samples of class 1. Frequentists often interpret the softmax probability as a measure of confidence (Guo et al. 2017). This example illustrates why this is a misnomer and we will show in the remainder of this thesis that this is also true for deep models on real-world data sets (see chapter 2).

In the next step, we move towards a full Bayesian treatment and introduce a prior distribution $p(\boldsymbol{\theta})$ over the parameters of the linear model. We opt for a Gaussian distribution

$$p(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta} \,|\, \mathbf{0}, \boldsymbol{I}) \tag{1.5}$$

and a categorical likelihood

$$p(\mathcal{C} \,|\, \mathcal{X}, \boldsymbol{\theta}) = \prod_{i=1}^{N} \mathsf{Categorical}(c_i \,|\, f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)) \ , \tag{1.6}$$

where $\mathsf{Categorical}(x)$ denotes the *categorical* or *multinoulli distribution*. A categorical distribution is a discrete probability distribution that assigns a non-zero probability to each possible outcome of a random variable x.

The posterior of the parameters $\boldsymbol{\theta}$ after observing the data set $\mathcal{D}$ is given by Eq. (1.2):

$$p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{C}) \propto p(\mathcal{C} \,|\, \mathcal{X}, \boldsymbol{\theta}) p(\boldsymbol{\theta}) \ . \tag{1.7}$$

Maximizing Eq. (1.7) w.r.t. $\boldsymbol{\theta}$ represents an intermediate step towards a fully Bayesian treatment and is referred to as *maximum posterior* (MAP) estimation. In case of a Gaussian prior, MAP estimation is equivalent to performing MLE with added weight decay regularization (see § 1.4). However, in a complete Bayesian setting, we are interested in inferencing the disease label c^* for a new, unseen input $\boldsymbol{x}^*$ after observing the training data:

$$p(c^* \,|\, \boldsymbol{x}^*, \mathcal{X}, C) = \int p(c^* \,|\, \boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{C}) \,\mathrm{d}\boldsymbol{\theta} \ , \tag{1.8}$$

which is referred to as the *posterior predictive distribution*. Here, the posterior distribution over $\boldsymbol{\theta}$ requires computation of the marginal likelihood, which cannot be done analytically.

To conclude this example, we discuss the results of two different approaches to approximate the posterior $p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{C})$. *Variational inference* (VI) is a common technique, where the true posterior is approximated by a simpler variational distribution $q_\phi(\boldsymbol{\theta})$, parameterized by ϕ. Without going into details at this point, we can use standard nonlinear optimization w.r.t. the variational parameters ϕ to bring q_ϕ close to the true posterior (see § 1.4). One way to restrict

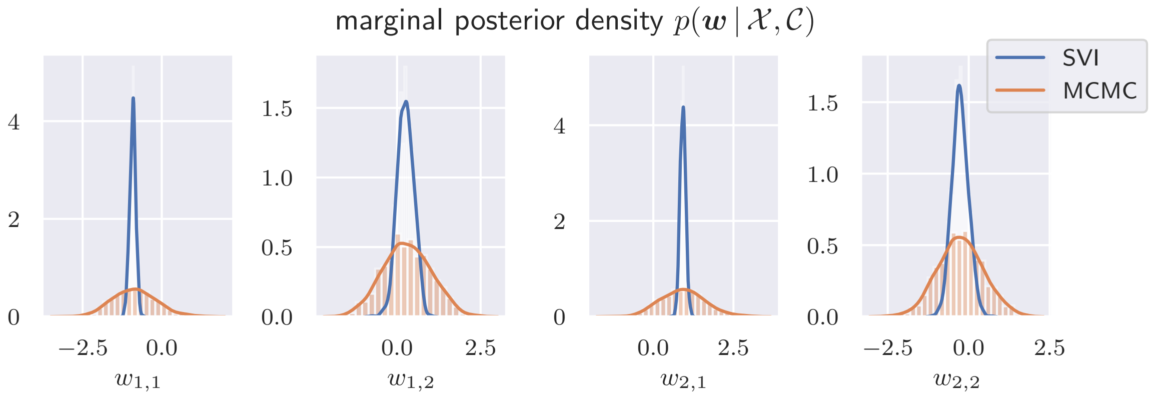


Figure 1.2: Posterior samples for weight matrix $\boldsymbol{w}$ from SVI and NUTS MCMC. The overconfidence from SVI with a factorized Gaussian as variational distribution is obvious. This results in overconfident predictions, e.g., for OoD data.

the family of variational distributions is to assume that q_ϕ factorizes, such that

$$q_\phi(\boldsymbol{\theta}) = \prod_i q_{\phi,i}(\boldsymbol{\theta}_i) \; . \tag{1.9}$$

Variational inference yields a tractable solution to the posterior. However, it must be considered that the approximation may be of unknown quality. The classification results for stochastic variational inference (SVI) with a factorized Gaussian approximation to the true posterior is shown in Fig. 1.1 (center). This approach does not overfit the training data and thus does not yield overconfident predictions for new samples (especially for those that might lie between the two point sets).

Finally, Markov chain Monte Carlo (MCMC) is a very powerful framework, which allows us to sample from any probability distribution, given a function that is proportional to the density and that the function can be evaluated; i.e., the right-hand side of Eq (1.7) (Bishop 2006). This allows us to find the posterior distribution without directly computing the intractable marginal likelihood. In contrast to other methods, MCMC makes no assumptions about the form of the distribution (Neal 1994).

Let $f(\boldsymbol{x}, \boldsymbol{\theta})$ be some unnormalized function that is easy to evaluate for any given $\boldsymbol{\theta}$ and that is proportional to the desired probability distribution $p(\boldsymbol{\theta} \mid \mathcal{D})$; for example the product of a prior distribution and a likelihood function. Directly sampling from $p(\boldsymbol{\theta} \mid \mathcal{D})$ is infeasible and thus, MCMC is performed by using a simpler *proposal distribution* $q(\boldsymbol{\theta}_{i+1} \mid \boldsymbol{\theta}_i)$, from which we can easily draw samples $\boldsymbol{\theta}^* \sim q$. The proposal distribution only depends on the current state $\boldsymbol{\theta}_i$ and, therefore, a sequence $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots$ forms a Markov chain. A key requirement of this chain is *ergodicity*: The distribution over the states of the Markov chain does not change once it has reached a stationary point (*invariance*) and that $q(\boldsymbol{\theta}_M)$ converges to the desired distribution $p(\boldsymbol{\theta} \mid \mathcal{D})$ as $M \to \infty$. It can be shown that MCMC allows us to draw unbiased samples from the true distribution of interest (Neal 1993). A common MCMC implementation is the Metropolis-Hastings algorithm[2] (Hastings 1970).

In this example, we use the recent No U-Turn Sampler (NUTS), which is slower than variational inference, but provides an exact estimate of the posterior distribution (Hoffman and Gelman 2014). Relative wall-times were 1.0 for the frequentist approach, 6.5 for SVI and 37.0 for NUTS (100 burn-in iterations and 1,000 posterior samples). At first glance, the posterior predictive distribution in Fig. 1.1 looks very similar. However, if we compare the posterior distribution of the parameters $\boldsymbol{\theta}$, a clear difference becomes apparent (see Fig. 1.2): The approximate posterior $p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{C})$ from SVI is too narrow and therefore overconfident. The reason for this is, that our variational family has diagonal covariance and cannot express correlations between the parameters. We will see later, however, that this restriction allows the application of SVI to deep convolutional networks (cf. § 1.4). The consequence of this phenomenon is particularly noticeable in the case of *out-of-distribution* (OoD) data (test data that is far away from the training data). Here we consider a test sample to be an OoD sample if (Hein et al. 2019)

$$\boldsymbol{x}^*_{\mathrm{OoD}} = \delta \boldsymbol{x}^* \,, \tag{1.10}$$

with sufficiently large δ. For demonstration purposes, we compute the empirical variance $\boldsymbol{s}^2$ (which we, for now, interpret as measure of uncertainty) from 1,000 posterior samples of the logits for $\boldsymbol{x}^*_{\mathrm{OoD}} = \delta[1,1]^\intercal$ with $\delta = 10$. The results $\boldsymbol{s}^2_{\mathrm{SVI}} = [6.9, 7.5]^\intercal$ and $\boldsymbol{s}^2_{\mathrm{MCMC}} = [101.5, 101.6]^\intercal$ reveal: The posterior from SVI is too narrow and thus overconfident.

Interim Conclusion

Unfortunately, the computational demands of MCMC methods prevent scaling to more complex models or larger data sets, as required for deep learning in medical imaging. Thus, the goal of this thesis is to use Bayesian deep learning with variational inference to obtain an approximate predictive posterior distribution and tackle overconfident predictions (with underestimated uncertainty) by post-hoc calibration. This allows practical estimation of well-calibrated uncertainty which we will demonstrate on a variety of medical imaging tasks that are approached with deep learning.

1.2 Deep Learning in Medical Imaging

This section provides a brief overview of the history and state-of-the-art concepts of medical imaging with deep learning. In many areas of medical imaging, deep learning has recently become the preferred method (Litjens et al. 2017). Several possibilities exist to categorize the prior work, e.g., by medical discipline, anatomical region or applied algorithm. Here, and throughout this thesis, we use the machine learning problems of *classification* and *regression* as distinction. In the context of medical image analysis, the first is often referred to as *computer-aided diagnosis* (CAD), but is not limited to it. The latter is further subdivided into

---

[2] A simple implementation of Metropolis-Hastings MCMC can be found at gist.github.com/mlaves.

regression of scalar or lower-dimensional vector-valued output and *generative models*, where the task is to reconstruct the input under certain constraints.

1.2.1 Classification

Due to the rapid progress in deep classification models (e.g., VGG, ResNet, Inception (He, X. Zhang, et al. 2016; Simonyan and Zisserman 2014; Szegedy et al. 2016)), computer-aided diagnosis was one of the first domains to benefit from deep learning. The goal is to assign an input image $\boldsymbol{x}$ to one of K discrete classes $C_k \in \{C_1, \ldots, C_K\}$. There are various works for most medical imaging modalities, including classification of breast lesions in ultrasound (US) images, pulmonary nodules in computed tomography (CT) scans (J.-Z. Cheng et al. 2016), macular degeneration in optical coherence tomography (OCT) scans (C. S. Lee et al. 2017), cardiac assessment from magnetic resonance imaging (MRI) (Bernard et al. 2018), cardiovascular risk factors or anaemia from retinal fundus images (Mitani et al. 2020; Poplin et al. 2018), or melanoma from dermoscopy images (Haenssle et al. 2018).

Most noteworthy are those works in which the performance of the deep learning algorithm is compared with the outcome of medical experts. Esteva et al. (2017) trained a convolutional neural network on 129,450 clinical photographs consisting of 2,032 different skin diseases and compared the performance of classification to 21 certified dermatologists using biopsy-labelled test images. The CNN achieved performance that was on par with the medical experts. An unmodified Inception v3 architecture with pre-trained weights from the ImageNet data set was used. Kermany et al. (2018) presented classification of retinal OCT images, again with on par performance to trained ophthalmologists. The same Inception v3 architecture was trained on a data set of 108,312 retinal OCT scans showing four different conditions. The data set itself was made available to the public. Most studies used pre-training on large non-medical data sets such as ImageNet to increase the classification performance. Medical data sets with expert labels are usually difficult to obtain and thus pre-training is standard protocol to achieve the desired accuracy.

If classification is performed for each pixel of an input image, this can be used for image segmentation. Segmentation is a common task in medial image analysis and aims at delineation of anatomical regions of interest such as organs, brain structures or tumors. The U-Net was one of the first CNNs used for binary segmentation in biomedical images (Ronneberger et al. 2015). They proposed a U-shaped autoencoder (hence the name) that produces binary masks of the structures of interest from grayscale input images. U-Net was extraordinarily influential on the field of medical segmentation and counts over 31,000 citations to date. Since then, segmentation was applied to all medical imaging modalities, including CT scans, nuclei segmentation in microscopy images, polyp segmentation in colonoscopy videos (Z. Zhou et al. 2018) and volumetric 3D images in general (Çiçek et al. 2016; Milletari et al. 2016). During this thesis, Bayesian segmentation was applied to laryngeal scenes (Laves, Bicker, et al. 2019). In accordance with our expectations, we have observed that predictions with a high degree of uncertainty occur particularly at object boundaries (see § 1.4).

Computer-aided diagnosis with deep learning has proven to perform well on a variety of tasks. This can help in clinical routine to quickly obtain a second opinion for medical decision making, reduce costs by freeing physicians from the time-consuming examination of large amounts of medical images, or bring cost-effective tools for early diagnosis to regions with a shortage of medical experts. In Chapter 2 we will propose to calibrate Bayesian neural networks to obtain predictive uncertainty in classification tasks. This allows detection and rejection of uncertain predictions, which is of utmost importance in medical decision making based on computer-aided diagnosis.

1.2.2 Regression

Compared to classification, regression is less studied in medical imaging. To date, the renowned journal Medical Image Analysis lists 817 articles with the keyword “classification”, but only 498 articles with the keyword “regression”. However, there is a broad field of applications, making it worth investigating. For the task of regression, a continuous scalar or vector-valued target $\boldsymbol{y} \in \mathbb{R}^d$ is estimated given an input image $\boldsymbol{x}$. This has been used for forensic age estimation, where the age of children in months is estimated from CT scans (Halabi et al. 2019) or MRI (Štern et al. 2016) of the hand. With recent CNN architectures, age estimation is possible with a mean absolute error of 0.37 ± 0.51 years. Regression can further be used in histopathology to detect the position of cells (Xie et al. 2018) or estimate the amount of tumor cells (Martel et al. 2019). The examination of biopsies and histological sections is a tedious task and requires a trained pathologist. Deep learning can help to accelerate this process and reduce cost. Segmentation can also be performed as a regression task by predicting the coordinates of object boundaries. This was performed for the segmentation of pulmonary nodules in CT scans (Messay et al. 2015), kidneys in ultrasound images (Yin et al. 2020), or left ventricles in MRI (L. K. Tan et al. 2017). In robot-assisted surgery, convolutional networks have recently been applied to regression of instrument poses from endoscopic images (Laves, Ihler, Fast, et al. 2020) and OCT scans (Gessert et al. 2018) or the localization of natural landmarks (Payer et al. 2019).

1.2.3 Generative Tasks

If the output dimensions of a regression model $\boldsymbol{f} \colon \mathbb{R}^{C \times H \times W} \to \mathbb{R}^{C \times H \times W}$ are increased to be in the range of the input image with number of channels C, and pixel height H and width W, we refer to this as a generative model. Generative tasks in the form of image enhancement and denoising have a long history in medical image analysis (Salinas and Fernandez 2007). Naturally, deep learning has recently been applied to this task. In (Laves, Ihler, Kahrs, et al. 2019b), we used an autoencoder CNN regularized by a pre-trained classifier network to denoise OCT scans without smoothing subtle anatomical details. Autoencoders usually have a data bottleneck between the encoding part and the decoding part, which forces the encoding part to extract a meaningful low-dimensional latent representation from a corrupted input image $\tilde{\boldsymbol{x}}$. This is then fed into the decoding part and mapped back to a reconstructed image

$\hat{x}$ in input space. Denoising can help to enhance, e.g., low-dose CT scans that are of poor quality, thus reducing radiation exposure.

Moreover, generative adversarial networks (GANs) (Goodfellow, Pouget-Abadie, et al. 2014) have been applied to denoising of 2D and 3D medical images (Ran et al. 2019). GANs have further been used to synthesize new realistic-looking medical images together with annotations (e.g., segmentation maps) (Zhao et al. 2018). This can be applied in surgical simulations or to generate training data for other data-driven algorithms. GANs can also be used to solve the task of super-resolution, where an upsampled image is generated with more detail and sharpness than, for example, with bilinear upsampling (Yi et al. 2019).

Finally, we consider whole-image deformable registration to be a generative task, which still is a major challenge in medical image processing. The result is a dense mapping showing pixel-wise non-linear correspondences between a pair of images that best aligns an input image onto a target image by means of some similarity definition. Deformable registration with deep learning is applied in analysis of patient-specific temporal or anatomical changes, e.g., from pre- to post-operative state, or to show inter-patient variances (Dalca et al. 2019).

However, there is a cardinal problem with learning-based generative models in medical imaging. Trained deep networks reconstruct the output from non-linear combinations of learned features. This can lead to a pathological phenomenon called *hallucination* (Yi et al. 2019), where the network embeds anatomical structures (e.g., retinal layers in OCT scans or small blood vessels in fundus imaging) that are not present in the input image. Worse still, images with hallucinated features appear to be valid even to the expert eye, due to the excellent performance of CNNs in generative tasks. In chapter 4, we will propose a novel Bayesian extension to the framework of deep image prior (Ulyanov et al. 2018) that produces hallucination-free images despite using the tools of deep learning.

1.3 Supervised Learning and Maximum Likelihood

This section gives a short introduction into supervised learning with maximum likelihood estimation and introduces the notation used throughout this thesis. For a more detailed insight into the principles of machine learning, we suggest the interested reader to consult textbooks from Bishop (2006), Goodfellow, Bengio, et al. (2016), and Held and Sabanés Bové (2014).

Most introductions to machine learning start with the simplest task of linear regression. Here, we shall skip a few steps and directly introduce the concept of supervised deep learning. The goal of a deep model $\boldsymbol{f}_{\boldsymbol{\theta}}$, parameterized by $\boldsymbol{\theta}$, is to predict a target value $\boldsymbol{y}$ given some new, unseen input $\boldsymbol{x}$ and a training set $\mathcal{D}$ of N independent inputs $\mathcal{X} = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$ and their corresponding (observed) target values $\mathcal{Y} = \{\boldsymbol{y}_1, \dots, \boldsymbol{y}_N\}$. This employs full supervision, as pairs of input data and ground truth outcome are presented to the model during training. Inputs and target values may be affected by noise, for example acquisition noise from a digital camera or X-ray sensor. To account for this, we assume that data points are drawn independently from a Gaussian distribution with unknown mean and unknown, but constant variance σ^2. We now let the deep model predict the mean $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}) = \hat{\boldsymbol{y}}$ for a given input. Thus, the deep model estimates a conditional distribution

$$p(\boldsymbol{y} \,|\, \boldsymbol{x}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{y} \,|\, \hat{\boldsymbol{y}}, \sigma^2) \tag{1.11}$$

to predict $\boldsymbol{y}$ given $\boldsymbol{x}$. Since we have assumed that our data is independent and identically distributed (i.i.d.), the probability of all targets given all inputs is given by (Bishop 2006)

$$p(\mathcal{Y} \,|\, \mathcal{X}, \boldsymbol{\theta}) = \prod_{n=1}^{N} \mathcal{N}(\boldsymbol{y}_n \,|\, \hat{\boldsymbol{y}}_n, \sigma^2) \; . \tag{1.12}$$

This probability distribution is a function of the unknown parameters $\boldsymbol{\theta}$ and is referred to as the *likelihood function* $L(\boldsymbol{\theta})$. It describes how likely it is to observe our data pairs $\{\mathcal{Y} \,|\, \mathcal{X}\}$, given the parameters $\boldsymbol{\theta}$ of the deep model. A model's parameter set that describes the observed data best can be found by maximizing $L(\boldsymbol{\theta})$ w.r.t. the model parameters

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \; L(\boldsymbol{\theta}) \; . \tag{1.13}$$

Here, we do not follow a Bayesian treatment and assume that a single best point estimate $\hat{\boldsymbol{\theta}}$ for the parameter set exists. The factors of Eq. (1.12) are $\in [0, 1]$ and for large data sets, the likelihood can become very small. For later numerical optimization of $\boldsymbol{\theta}$, it is useful to take the logarithm and to perform minimization instead of maximization

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \left[-\sum_{n=1}^{N} \log \mathcal{N}(\boldsymbol{y}_n \,|\, \hat{\boldsymbol{y}}_n, \sigma^2) \right] \; . \tag{1.14}$$

With the probability density function of the Gaussian distribution follows

$$-\log L(\boldsymbol{\theta}) = -\sum_{n=1}^{N} \log \left(\frac{1}{\sqrt{2\pi}\sigma} \exp\left\{ -\frac{\left\|\boldsymbol{y}_n - \hat{\boldsymbol{y}}_n\right\|^2}{2\sigma^2} \right\} \right) \tag{1.15}$$

$$= N \log(\sigma) + \frac{N}{2} \log(2\pi) + \sum_{n=1}^{N} \frac{1}{2\sigma^2} \left\|\boldsymbol{y}_n - \hat{\boldsymbol{y}}_n\right\|^2 \; . \tag{1.16}$$

Ignoring constants and dividing by N leads to the following optimization criterion

$$\mathcal{L}_{\mathrm{MSE}}(\boldsymbol{\theta}) = \frac{1}{N} \sum_{n=1}^{N} \left\|\boldsymbol{y}_n - \hat{\boldsymbol{y}}_n\right\|^2 \; . \tag{1.17}$$

Thus, maximizing a Gaussian log-likelihood in Eq. (1.12) w.r.t. the model parameters $\boldsymbol{\theta}$ is equivalent to minimizing the mean squared error (MSE). Any real-valued mean is a valid parameter for a Gaussian distribution, which makes this approach suitable for regression.

For the task of classification, we model the observations as generalized Bernoulli distributed (also referred to as categorical distribution). For the parameters of a categorical distribution, we need to specify the class probability

$$p(y \mid \boldsymbol{x}, \boldsymbol{\theta}) = \mathsf{Categorical}\left(y \mid \hat{\boldsymbol{y}}\right) \tag{1.18}$$

of class $y = c \in \{1, \ldots, C\}$ given $\boldsymbol{x}$. This probability must lie between 0 and 1 and the probabilities over all classes must sum up to $\sum_{c=1}^{C} p(y = c \mid \boldsymbol{x}, \boldsymbol{\theta}) = 1$. A common way to achieve this is to "squash" the output of the deep model using the *softmax* function

$$\sigma_{\mathrm{SM}}(\hat{\boldsymbol{y}})_c = \frac{\exp(\hat{y}_c)}{\sum_{i=1}^{C} \exp(\hat{y}_i)} \ , \tag{1.19}$$

where $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}) = \hat{\boldsymbol{y}} = [\hat{y}_1, \ldots, \hat{y}_C]^\mathsf{T}$. The softmax function σ_{SM} has several properties that makes it suitable for gradient-based optimization of $\boldsymbol{\theta}$. It is a smooth and differentiable approximation to the $\arg\max$ function and numerically stable if implemented as $\log\left(\sigma_{\mathrm{SM}}\right)$. We now perform maximum likelihood estimation using the following optimization criterion

$$\mathcal{L}_{\mathrm{CE}}(\boldsymbol{\theta}) = -\sum_{n=1}^{N} \log \sigma_{\mathrm{SM}}(\hat{\boldsymbol{y}}_n)_{y_n} \ , \tag{1.20}$$

where y_n denotes the ground truth class label corresponding to input image $\boldsymbol{x}_n$. This optimization criterion is often referred to as cross entropy (CE), which combines the log-softmax with the negative log-likelihood function.

Maximum likelihood estimation is the bread-and-butter tool of machine learning. However, it only yields a single-best point estimate $\hat{\boldsymbol{\theta}}$ and does not allow quantification of parameter uncertainty. Moreover, MLE is prone to severely overfitting the training data: MLE fails in the simplest scenarios if the number of parameters is chosen inappropriately (Bishop 2006).

1.4 Bayesian Deep Learning and Variational Inference

In § 1.1 and the previous section, we have already outlined the disadvantages of maximum likelihood estimation: overfitting and the inability to express uncertainty in model parameters. We shall now move towards a Bayesian approach, which allows us to estimate distributions over all possible parameters. First, we introduce a prior distribution $p(\boldsymbol{\theta} \mid \lambda)$ over the parameters $\boldsymbol{\theta}$ of our deep model $\boldsymbol{f}_{\boldsymbol{\theta}}$. Following Bishop (2006), we consider a Gaussian distribution governed by a precision hyperparameter $\lambda = \sigma^{-2}$ (reciprocal of variance)

$$p(\boldsymbol{\theta} \mid \lambda) = \mathcal{N}\left(\boldsymbol{\theta} \mid \boldsymbol{0}, \lambda^{-1}\boldsymbol{I}\right) = \left(\frac{\lambda}{2\pi}\right)^{N/2} \exp\left\{-\frac{\lambda}{2}\boldsymbol{\theta}^\mathsf{T}\boldsymbol{\theta}\right\} \ , \tag{1.21}$$

where N is the number of parameters in the parameter set $\boldsymbol{\theta}$. The posterior distribution is given by applying Bayes' theorem

$$p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{Y}, \lambda) \propto p(\mathcal{Y} \,|\, \mathcal{X}, \boldsymbol{\theta}) p(\boldsymbol{\theta} \,|\, \lambda) \, . \tag{1.22}$$

We can now find $\boldsymbol{\theta}$ by employing *maximum posterior* (MAP) estimation

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \; p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{Y}, \lambda) \, , \tag{1.23}$$

which—using a Gaussian likelihood—leads to the following optimization criterion

$$\mathcal{L}_{\mathrm{MAP}}(\boldsymbol{\theta}) = \underbrace{\frac{1}{N} \sum_{n=1}^{N} \left\| \boldsymbol{y}_n - \hat{\boldsymbol{y}}_n \right\|^2}_{(1.17)} + \frac{\lambda}{2} \boldsymbol{\theta}^{\mathsf{T}} \boldsymbol{\theta} \, . \tag{1.24}$$

The right term in Eq. (1.24) penalizes large values of $\boldsymbol{\theta}$ and is referred to as *weight decay* or L_2 regularization. Using a Laplacian prior results in performing L_1 regularization. Comparing this with Eq. (1.17), we see that maximizing the posterior is equivalent to minimizing the MSE with added weight decay. The prior in Bayesian neural networks is usually implemented as weight decay (Kendall and Gal 2017) and it is an effective technique to prevent overfitting of over-parameterized models, which is especially important in deep learning, where models tend to have millions of parameters.

Variational Inference MAP estimation still yields a point estimate $\hat{\boldsymbol{\theta}}$ and does not consider parameter uncertainty. However, the true posterior $p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{Y})$ is intractable, even for the simplest neural network with nonlinear activations (Bishop 2006; Gal 2016), and consequently the posterior predictive distribution is intractable as well. In § 1.1, we already discussed the two most polular methods for Bayesian posterior approximation. Markov chain Monte Carlo is nonparametric and asymptotically exact; it allows us to sample from the true posterior (Salimans et al. 2015). On the other hand, MCMC is computationally expensive and does not scale to deep models and large data sets (Blei et al. 2017; Cornish et al. 2019). In general, medical imaging data sets are comparatively small considering the number of independent samples, e.g., number of patients (Laves, Bicker, et al. 2019). However, a single sample is a high dimensional data point as medical images usually have a high pixel resolution and are often volumetric (e.g., CT, MRI, 3D OCT, 3D US). This makes MCMC impracticable and leads us to variational inference (VI).

In variational inference, we aim at finding a simpler, variational approximation to the Bayesian posterior distribution over the parameters $\boldsymbol{\theta}$. VI uses optimization instead of sampling to find the member $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ of a family of distributions $\mathcal{Q}$ (e.g., a multivariate Gaussian) that is close to the exact posterior, defined by the variational parameters $\boldsymbol{\phi}$. More formally, we optimize $q_{\boldsymbol{\phi}}$ w.r.t. $\boldsymbol{\phi}$, such that the *Kullback-Leibler divergence* (KL), which is a measure of similarity between two distributions, is minimized with regard to the true but unknown posterior (Blei et al. 2017; Kullback and Leibler 1951):

$$\boldsymbol{\phi}^* = \arg\min_{\boldsymbol{\phi}} \; \mathrm{KL}\left[q_{\boldsymbol{\phi}}(\boldsymbol{\theta}) \,\|\, p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{Y})\right] \; . \tag{1.25}$$

$\mathrm{KL}\left[q_{\boldsymbol{\phi}}(\boldsymbol{\theta}) \,\|\, p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{Y})\right]$ is defined as (all expectations are taken w.r.t. $q_{\boldsymbol{\phi}}$)

$$\begin{aligned} & \int q_{\boldsymbol{\phi}}(\boldsymbol{\theta}) \log \frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta})}{p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{Y})} \, \mathrm{d}\boldsymbol{\theta} & (1.26) \\ &= \mathbb{E}\left[\log q_{\boldsymbol{\phi}}(\boldsymbol{\theta})\right] - \mathbb{E}\left[\log p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{Y})\right] & (1.27) \\ &= \mathbb{E}\left[\log q_{\boldsymbol{\phi}}(\boldsymbol{\theta})\right] - \mathbb{E}\left[\log p(\boldsymbol{\theta}, \mathcal{Y} \,|\, \mathcal{X})\right] + \log p(\mathcal{Y} \,|\, \mathcal{X}) \; , & (1.28) \end{aligned}$$

which contains the intractable marginal likelihood (or evidence)

$$p(\mathcal{Y} \,|\, \mathcal{X}) = \int p(\mathcal{Y} \,|\, \mathcal{X}, \boldsymbol{\theta}) p(\boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta} \; . \tag{1.29}$$

We can rewrite Eq. (1.28) with

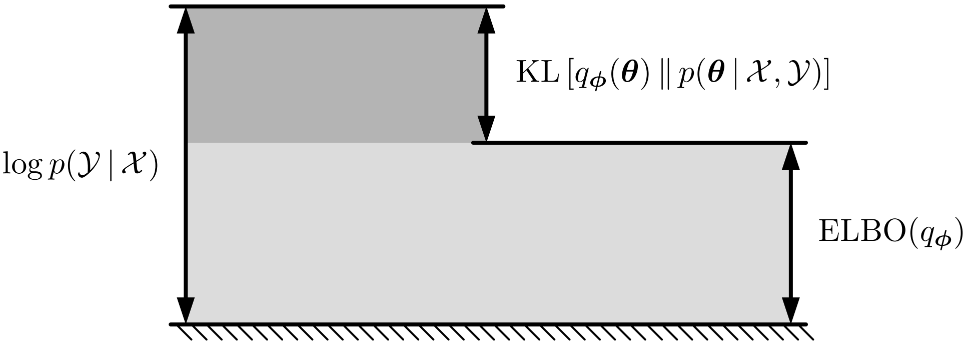
$$\begin{aligned} \mathrm{ELBO}(q_{\boldsymbol{\phi}}) &= \mathbb{E}\left[\log p(\boldsymbol{\theta}, \mathcal{Y} \,|\, \mathcal{X})\right] - \mathbb{E}\left[\log q_{\boldsymbol{\phi}}(\boldsymbol{\theta})\right] & (1.30) \\ &= \mathbb{E}\left[\log p(\boldsymbol{\theta})\right] + \mathbb{E}\left[p(\mathcal{Y} \,|\, \mathcal{X}, \boldsymbol{\theta})\right] - \mathbb{E}\left[\log q_{\boldsymbol{\phi}}(\boldsymbol{\theta})\right] & (1.31) \\ &= \mathbb{E}\left[\log p(\mathcal{Y} \,|\, \mathcal{X}, \boldsymbol{\theta})\right] - \mathrm{KL}\left[q_{\boldsymbol{\phi}}(\boldsymbol{\theta}) \,\|\, p(\boldsymbol{\theta})\right] \; , & (1.32) \end{aligned}$$

such that

$$\log p(\mathcal{Y} \,|\, \mathcal{X}) = \mathrm{KL}\left[q_{\boldsymbol{\phi}}(\boldsymbol{\theta}) \,\|\, p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{Y})\right] + \mathrm{ELBO}(q_{\boldsymbol{\phi}}) \; . \tag{1.33}$$

Since the KL is non-negative, we see that $\log p(\mathcal{Y} \,|\, \mathcal{X}) \geq \mathrm{ELBO}(q_{\boldsymbol{\phi}})$. Thus, $\mathrm{ELBO}(q_{\boldsymbol{\phi}})$ is a lower bound on the (log) evidence, giving it it's name *evidence lower-bound* (Jordan et al. 1999). Maximizing the ELBO w.r.t. $\boldsymbol{\phi}$ is equivalent to minimizing the KL between our variational distribution $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ and the true but unknown posterior $p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{Y})$ (cf. Fig. 1.3). Examining Eq. (1.32), we see that the ELBO consists of the expected likelihood and the negative KL between the variational distribution and the prior. The first encourages the model to explain the observed data well and the latter makes sure, that the approximate posterior is close to the prior. This corresponds to the desired Bayesian treatment. The question remains to which variational family $\mathcal{Q}$ we restrict the approximate distribution $q_{\boldsymbol{\phi}}$.

Mean-Field Variational Inference We have discussed the principles of variational inference and we will now focus on the selection of the variational family $\mathcal{Q}$. As variational inference constitutes an optimization problem, the complexity of the optimization depends on the complexity of $\mathcal{Q}$ (Blei et al. 2017). A common and well-accepted restriction of $\mathcal{Q}$ is the *mean-field variational family*, where each entry i of the parameter vector $\boldsymbol{\theta}$ is assumed to be independent and defined by a separate variational density $q_i(\theta_i)$. In this case, the variational



$\mathrm{KL}\left[q_{\phi}(\boldsymbol{\theta}) \,\|\, p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y})\right]$
$\log p(\mathcal{Y} \mid \mathcal{X})$
$\mathrm{ELBO}(q_{\phi})$

Figure 1.3: $\mathrm{ELBO}(q_{\phi})$ is a lower bound on the (log) marginal likelihood $\log p(\mathcal{Y} \mid \mathcal{X})$. Maximizing the ELBO results in minimizing the KL divergence between the variational distribution $q_{\phi}(\boldsymbol{\theta})$ and the true parameter posterior $p(\boldsymbol{\theta} \mid \mathcal{X}, \mathcal{Y})$.

distribution is factorized as

$$q(\boldsymbol{\theta}) = \prod_i q_i(\theta_i) \ . \tag{1.34}$$

When performing mean-field variational inference (MFVI), the factors are optimized to maximize the ELBO in Eq. (1.32). More specifically, we utilize a fully factorized Gaussian distribution (with diagonal covariance) to model the variational posterior (Graves 2011):

$$q(\boldsymbol{\theta}) = \prod_i \mathcal{N}(\boldsymbol{\theta}_i \mid \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2) \ , \tag{1.35}$$

$$\log q(\boldsymbol{\theta}) = \sum_i \log \mathcal{N}(\boldsymbol{\theta}_i \mid \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2) \ . \tag{1.36}$$

The use of a Gaussian is justified under the *Bayesian central limit theorem* (CLT, also known as Bernstein-von Mises theorem), which states that the posterior approaches a Gaussian distribution in the limit of large data (C. Zhang et al. 2018). Practical implementations of MFVI are *variational autoencoder* (D. Kingma and Welling 2014), *variational dropout* (D. P. Kingma and Ba 2014), *Bayes by Backprop* (Blundell et al. 2015) and *SWAG-diagonal* (Maddox et al. 2019).

Monte Carlo Dropout A recent method for practical variational inference that scales to very deep models and large data sets is *Monte Carlo dropout* (Gal and Ghahramani 2016b). In Monte Carlo dropout, dropout is used before every weight layer of a neural network (Srivastava et al. 2014). Dropout is a stochastic regularization technique, where entries of the input $\boldsymbol{x}$ to a weight layer $\boldsymbol{w}$ are randomly set to zero by elementwise multiplication $\odot$ with

$$\boldsymbol{d} \quad \text{where} \quad d_j \sim \mathsf{Bernoulli}(1-p) \ , \tag{1.37}$$

$$\boldsymbol{y} = \boldsymbol{w}^{\mathsf{T}}(\boldsymbol{d} \odot (\boldsymbol{x}/(1-p))) \ , \tag{1.38}$$

with dropout rate p. This introduces Bernoulli noise during optimization and reduces overfitting of the training data.

Training a neural network with dropout is equivalent to minimizing the KL divergence between an approximate distribution and the true Bayesian posterior. We skip the derivation of MC dropout here and refer the interested reader to § 3.1 in (Gal and Ghahramani 2016a). Using dropout at test time, we can now sample from the approximate posterior $\tilde{\boldsymbol{\theta}} \sim q_\phi(\boldsymbol{\theta})$, which allows us to compute the likelihood $p(\boldsymbol{y}^* \,|\, \boldsymbol{x}^*, \tilde{\boldsymbol{\theta}})$. Further, we can use Monte Carlo integration with T samples to approximate the posterior predictive distribution (Gal 2016)

$$q_\phi(\boldsymbol{y}^* \,|\, \boldsymbol{x}^*) := \frac{1}{T}\sum_{t=1}^{T} p(\boldsymbol{y}^* \,|\, \boldsymbol{x}^*, \tilde{\boldsymbol{\theta}}_t) \xrightarrow[T\to\infty]{} \int p(\boldsymbol{y}^* \,|\, \boldsymbol{x}^*, \boldsymbol{\theta}) q_\phi(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta} \tag{1.39}$$

$$\approx \int p(\boldsymbol{y}^* \,|\, \boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta} \,|\, \mathcal{X}, \mathcal{C})\, \mathrm{d}\boldsymbol{\theta} \tag{1.40}$$

$$= p(\boldsymbol{y}^* \,|\, \boldsymbol{x}^*, \mathcal{X}, \mathcal{Y}) \,. \tag{1.41}$$

Following Gal (2016), we perform moment-matching to estimate the first two moments of the posterior predictive distribution. Given $p(\boldsymbol{y}^* \,|\, \boldsymbol{x}^*, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{y}^* \,|\, \boldsymbol{\theta}(\boldsymbol{x}), \beta^{-1}\boldsymbol{I})$ with precision $\beta > 0$, the first raw moment can be estimated with

$$\hat{\mathbb{E}}[\boldsymbol{y}^*] := \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{f}_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{x}^*) \xrightarrow[T\to\infty]{} \mathbb{E}_{q_\phi(\boldsymbol{y}^*|\boldsymbol{x}^*)}[\boldsymbol{y}^*] \,, \tag{1.42}$$

and the second raw moment can be estimated by

$$\hat{\mathbb{E}}\left[(\boldsymbol{y}^*)^\mathsf{T}(\boldsymbol{y}^*)\right] := \beta^{-1}\boldsymbol{I} + \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{f}_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{x}^*)^\mathsf{T} \boldsymbol{f}_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{x}^*) \xrightarrow[T\to\infty]{} \mathbb{E}_{q_\phi(\boldsymbol{y}^*|\boldsymbol{x}^*)}[(\boldsymbol{y}^*)^\mathsf{T}(\boldsymbol{y}^*)] \,. \tag{1.43}$$

Proofs of Eq. (1.42) & (1.43) can be found in (Gal 2016, § 3.3). This allows us to estimate the predictive variance

$$\widehat{\mathrm{Var}}[\boldsymbol{y}^*] = \beta^{-1}\boldsymbol{I} + \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{f}_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{x}^*)^\mathsf{T} \boldsymbol{f}_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{x}^*) - \hat{\mathbb{E}}[\boldsymbol{y}^*]^\mathsf{T} \hat{\mathbb{E}}[\boldsymbol{y}^*] \xrightarrow[T\to\infty]{} \mathrm{Var}_{q_\phi(\boldsymbol{y}^*|\boldsymbol{x}^*)}[\boldsymbol{y}^*] \,. \tag{1.44}$$

MC dropout will be used throughout this thesis as it is a practical variational Bayesian method for very deep models. It allows us to obtain model uncertainty, also referred to as *epistemic* uncertainty (from Greek *epistēmikós* (knowledge)), which is caused by uncertainty in the model parameters (i.e., the width of the parameter posterior distribution). However, we will show later that its uncertainty is miscalibrated and, therefore, cannot be used out-of-the-box for medical imaging tasks. In fact, the uncertainty is underestimated, which leads to overconfident predictions. We will propose a recalibration method to obtain well-calibrated uncertainty estimates, which can be used to increase robustness of predictions in both classification and regression tasks. Further, we will use calibrated uncertainty to create good pseudo-labels in an unsupervised training scenario.

1.5 Hypotheses and Contributions

In the following, we formulate the hypotheses of this thesis and present our main contributions. The main goal is to use "good" Bayesian uncertainty to make medical imaging with deep learning more robust and more accurate, which leads us to our first research hypothesis:

Hypothesis 1. *Well-calibrated predictive uncertainty in medical imaging with Bayesian deep learning can be obtained by recalibration.*

In Chapter 2 and (Laves, Ihler, Kortmann, et al. 2019) we propose a simple, yet effective calibration method for predictive uncertainty in classification and computer-aided diagnosis. The method uses a simple learnable scalar parameter to scale the posterior predictive distribution. Well-calibrated uncertainty is used to identify unreliable predictions and out-of-distribution samples, which is essential in medical imaging with deep learning. We present the Uncertainty Calibration Error (UCE), a new error metric to quantify miscalibration of uncertainty. With well-calibrated uncertainty, we can identify predictions that are likely to be correct, which leads us to the next research hypothesis:

Hypothesis 2. *Computer-aided diagnosis with deep learning is possible with limited or without any labeled training data when considering well-calibrated predictive uncertainty.*

Furthermore, we will use well-calibrated uncertainty in Chapter 2.2 to create good pseudo-labels in an unsupervised learning scenario. We build upon recent advances in self-supervised and unsupervised learning and utilize the estimated uncertainty to identify highly confident predictions that are probably correct. From this, we produce pseudo-labels and use them for supervised training. This approach is especially interesting for medical imaging with deep learning, because labeling of the data requires medical expertise and is therefore very expensive. Obtaining unlabeled data in clinical routine is usually much easier. Moreover, we do not restrict calibration of uncertainty to classification tasks, which brings us to our last research hypothesis:

Hypothesis 3. *Regressive and generative tasks in medical imaging with deep learning benefit from well-calibrated predictive uncertainty.*

In Chapter 3 and (Laves, Ihler, Fast, et al. 2020), we extend calibration of Bayesian uncertainty to regression. Besides CAD and segmentation, many medical imaging tasks can be approached as a regression problem (cf. § 1.2.2). We analyze and provide theoretical background why deep models for regression are miscalibrated. The UCE is extended to regression and we propose to use credible intervals of the approximate posterior predictive distribution to further assess the calibration of uncertainty.

If the output of a regression model has the same spatial dimensions as the input, we define this as a generative model. In Chapter 4 and (Laves, Tölle, et al. 2020), we extend variational Bayesian inference to medical image denoising and obtain pixel-wise uncertainty. We use a randomly initialized convolutional network as parameterization of the denoised image and perform gradient descent to match the noisy observation, which is known as deep image prior. In this case, the reconstruction does not suffer from hallucinations, which is a mayor issue in inverse medical imaging problems. This approach is further extended to deformable registration (Laves, Ihler, and Ortmaier 2019).

The main contributions of this work are outlined as follows:

1. We propose *Uncertainty Calibration Error* (UCE), a new metric for perfect calibration of uncertainty, derivation of logit scaling for Gaussian dropout, apply logit scaling calibration to a Bayesian classifier obtained from MC dropout, and provide empirical evidence that logit scaling leads to well-calibrated uncertainty which allows robust OoD detection (Laves, Ihler, Kortmann, et al. 2019, 2020).

2. We advance the state-of-the-art in unsupervised learning by combining mutual information maximization and consistency learning with probably good pseudo-labels from well-calibrated uncertainty. We present *BatchPL*, a novel sample acquisition function for efficient pseudo-labeling.

3. We are the first to address calibration of predictive uncertainty for regression tasks in medical imaging, analyze and provide theoretical background why deep models for regression are miscalibrated, propose σ scaling to tackle underestimation of uncertainty, propose the uncertainty calibration error for regression and usage of prediction intervals to assess the quality of the estimated uncertainty (Laves, Ihler, Fast, et al. 2020).

4. We propose to use deep image prior to cope with hallucinations in medical image denoising and provide a novel Bayesian approach with Monte Carlo dropout that yields well-calibrated reconstruction uncertainty and avoids the need for early stopping (Laves, Tölle, et al. 2020).

1.6 Thesis Structure

This thesis is organized in three main chapters, of which each chapter addresses an important sub-field of medical imaging with deep learning, and two accompanying chapters that introduce the general topic and conclude the work.

Chapter 1 gives an introduction to the importance of uncertainty in medical decision making and outlines the main problem that is the subject of this thesis: the miscalibration of uncertainty derived from practical approximate Bayesian methods. In addition, the increasing relevance of deep learning in medical image analysis is underlined by a small literature review. The chapter briefly revisits the mathematical foundations used in this thesis and outlines our main contributions.

In Chapter 2, we focus on computer-aided diagnosis and approach calibration of uncertainty in classification tasks. First, we deal with calibration of deep classifiers itself and show, why current metrics fail to measure miscalibration properly. We define the Uncertainty Calibration Error that avoids pathologies of existing metrics. UCE can further be used as a regularization during training to get better calibrated models. The second part of Chapter 2 uses Bayesian uncertainty to obtain good pseudo-labels in a self-supervised scenario.

Chapter 3 addresses calibration of uncertainty for regression tasks in medical imaging. We first analyze and provide theoretical background why deep models for regression are miscalibrated. Next, we suggest to use σ scaling to tackle underestimation of regression uncertainty. Two new metrics to quantify quality of calibration are presented: the uncertainty calibration error for regression and prediction interval diagrams. Extensive experiments on four different medical regression data sets are conducted with four recent convolutional network architectures to show the effectiveness of the proposed method.

In Chapter 4, we extend uncertainty estimation in regression to generative tasks and address the problem of hallucinations in medical imaging with deep learning. Additionally, we study deformable registration and improve supervised denoising by semantic regularization with a pretrained medical image classifier.

In Chapter 5, this thesis as a whole is summarized and critically concluded. We discuss caveats of the proposed methods, show open questions and possible future work.

Programming code for all experiments of this thesis is available at https://github.com/mlaves.

2 Computer-Aided Diagnosis

Computer-aided diagnosis (CAD) based on deep learning has been demonstrated to achieve a performance similar to that of human experts in classification tasks in medical imaging (Esteva et al. 2017; Kermany et al. 2018). Equipped with deep neural networks, mobile assistance systems can extend the reaching of medical experts in the field and increase access to medical care. However, common tools in deep learning do not provide uncertainty for predictions of disease conditions. When an ambiguous or unknown case is presented to a deep learning model, it lacks the ability to say "I don't know". Therefore, especially in medical imaging and CAD, measuring the uncertainty of predictions is needed for profound decision making. Bayesian neural networks allow us to reason about predictive uncertainty. However, the question arises of what level of quality and reliability this uncertainty is.

The first part of this chapter (§ 2.1) answers this question and deals with the calibration of supervisely trained deep models for multi-class classification. It proposes a new metric for measuring miscalibration and is based on a peer-reviewed publication accepted at the 4th "Bayesian Deep Learning Workshop" at the 33rd "Conference on Neural Information Processing Systems" (NeurIPS) 2019 (Laves, Ihler, Kortmann, et al. 2019). Source code for the first section is available at github.com/mlaves/bayesian-temperature-scaling.

The second part of this chapter (§ 2.2) introduces a novel framework for unsupervised training of multi-class classification models with uncertainty-aware pseudo-labels. We combine consistency learning and mutual information maximization with self-supervised training using good pseudo-labels from predictions with low uncertainty. Our method achieves strong results with state-of-the-art classification accuracy without using any labeled data.

2.1 Calibration of Uncertainty for Variational Inference

The uncertainty obtained by variational Bayesian inference is prone to miscalibration and does not represent the model error well. In this section, different logit scaling methods are extended to variational inference to recalibrate Bayesian uncertainty. The effectiveness of recalibration is evaluated on the public data sets CIFAR-10/100 (Krizhevsky 2009) and SVHN (Netzer et al. 2011) for recent CNN architectures. Various metrics have recently been proposed to measure uncertainty calibration of deep models for classification. However, these metrics either fail to capture miscalibration correctly or lack interpretability. We propose to use the normalized entropy as a measure of uncertainty and derive the *Uncertainty Calibration Error* (UCE), a comprehensible calibration metric for multi-class classification. UCE avoids several pathologies of other metrics, but does not sacrifice interpretability. It can be used for regularization to improve calibration during training without penalizing

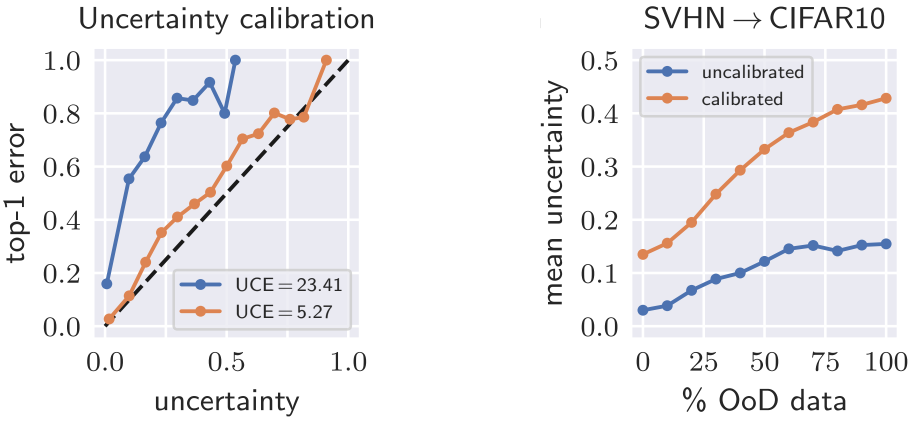


Figure 2.1: Calibration of uncertainty: (Left) reliability diagrams with uncertainty calibration error (UCE) and (right) detection of out-of-distribution (OoD) data. Uncalibrated uncertainty does not correspond well with the top-1 error. Logit scaling is able to recalibrate deep Bayesian neural networks, which enables robust OoD detection. The dashed line in the left figure denotes perfect calibration.

predictions with justified high confidence. Experimental results show that logit scaling considerably reduces miscalibration by means of UCE. Well-calibrated uncertainty enables reliable rejection of uncertain predictions and robust detection of out-of-distribution data (see Fig. 2.1).

2.1.1 Introduction

Advances in deep learning have led to high accuracy predictions for classification tasks, making deep-learning classifiers an attractive choice for safety-critical applications like autonomous driving (C. Chen et al. 2015) or computer-aided diagnosis (Esteva et al. 2017). However, the high accuracy of recent deep learning models is not sufficient for such applications. In cases, where serious decisions are made upon a model's predictions, it is essential to also consider the uncertainty of these predictions. We need to know if the prediction of a model is likely to be incorrect or if invalid input data is presented to a deep model, e.g., data that is far away from the training domain or obtained from a defective sensor. The consequences of a false decision based on an uncertain prediction can be fatal.

A natural expectation is that the certainty of a prediction should be directly correlated with the quality of the prediction. In other words, a prediction with a high certainty is more likely to be accurate than an uncertain prediction which is likely to be incorrect. A common misconception is the assumption that the estimated class likelihood (of a softmax activation) can be directly used as a confidence measure for the predicted class. This expectation is dangerous in the context of critical decision-making. The estimated likelihood of a model trained by minimizing the negative log-likelihood (i.e., cross entropy) is highly overconfident (cf. § 1.1). That is, the estimated likelihood is considerably higher than the observed frequency of accurate predictions with that likelihood (Guo et al. 2017).

Guo et al. (2017) proposed calibration of the likelihood estimation by scaling the logit output of a neural network to achieve a correlation between the predicted likelihood and the expected likelihood. However, they follow a frequentist approach, where they assume a single best point estimate of the parameters (or weights) of a neural network. In frequentist inference, the weights of a deep model are obtained by maximum likelihood estimation (Bishop 2006), and the normalized output likelihood for an unseen test input does not consider uncertainty in the weights (Kendall and Gal 2017). Weight uncertainty (also referred to as model or epistemic uncertainty) is a considerable source of predictive uncertainty for models trained on data sets of limited size (Bishop 2006; Kendall and Gal 2017). Bayesian neural networks and recent advances in their approximation provide valuable mathematical tools for quantification of model uncertainty (Gal and Ghahramani 2016b; D. Kingma and Welling 2014). Instead of assuming the existence of a single best parameter set, we place distributions over the parameters and want to consider all possible parameter configurations, weighted by their posterior. More formally, given a training data set $\mathcal{D}$ of labeled images and an unseen test image $\boldsymbol{x}$ with class label y, we are interested in evaluating the predictive distribution

$$p(y|\boldsymbol{x}, \mathcal{D}) = \int p(y|\boldsymbol{x}, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} \, . \tag{2.1}$$

This integral requires to evaluate the posterior $p(\boldsymbol{\theta}|\mathcal{D})$, involving the intractable marginal likelihood (Gal 2016). The posterior can be approximated using variational inference. It is commonly used to obtain epistemic uncertainty, which is caused by uncertainty in the model weights. However, epistemic uncertainty from VI still tends to be miscalibrated, i.e., the uncertainty does not correspond well with the model error (Gal, Hron, et al. 2017). The quality of uncertainty highly depends on the approximate posterior (Louizos and Welling 2017). Lakshminarayanan et al. (2017) state that VI uncertainty does not allow to robustly detect out-of-distribution data. However, calibrated uncertainty is essential as miscalibration can lead to decisions with catastrophic consequences in the aforementioned task domains.

Therefore, we propose a novel notion for perfect calibration of uncertainty and propose a definition of expected *uncertainty calibration error* (UCE), derived from expected calibration error (ECE) (Naeini et al. 2015). We then show how current calibration techniques (for confidence) based on logit scaling can be extended to calibrate model uncertainty (§ 2.1.3). We compare calibration results for temperature scaling, vector scaling and auxiliary scaling (Guo et al. 2017; Kuleshov et al. 2018) using our metric UCE as well as established ECE (§ 2.1.4). We finally show how calibrated model uncertainty improves out-of-distribution (OoD) detection, as well as predictive accuracy by rejecting high-uncertainty predictions.

In summary the main contributions of this section are

1. a new metric for perfect calibration of uncertainty,
2. derivation of logit scaling for Monte Carlo integration, and
3. empirical evidence that logit scaling leads to well-calibrated model uncertainty which

allows robust OoD detection (in contrast to what is stated in (Lakshminarayanan et al. 2017); shown for different network architectures on CIFAR-10/100 and SVHN.

2.1.2 Related Work

Overconfident predictions of neural networks have been addressed by entropy regularization techniques. Szegedy et al. (2016) presented label smoothing as regularization of models during supervised training for classification. They state that a model trained with one-hot encoded labels is prone to becoming overconfident about its predictions, which causes overfitting and poor generalization. Pereyra et al. (2017) link label smoothing to confidence penalty and propose a simple way to prevent overconfident networks. Low entropy output distributions are penalized by adding the negative entropy to the training objective. However, the referred works do not apply entropy regularization to the calibration of confidence or uncertainty. In the last decades, several non-parametric and parametric calibration approaches such as isotonic regression (Zadrozny and Elkan 2002) or Platt scaling (Platt 1999) have been presented. Recently, temperature scaling has been demonstrated to lead to well-calibrated model likelihood in non-Bayesian deep neural networks (Guo et al. 2017). It uses a single scalar τ to scale the logits and smoothen ($\tau > 1$) or sharpen ($\tau < 1$) the softmax output and thus regularize the entropy. Logit scaling has also been introduced to approximate categorical distributions by the Gumbel-Softmax or Concrete distribution (Jang et al. 2016; Maddison et al. 2016). Recently, (Kull et al. 2019) stated that temperature scaling does not lead to classwise-calibrated models because the single parameter τ cannot calibrate each class individually. They proposed Dirichlet calibration to address this problem. To verify this statement, we will investigate classwise logit scaling in addition to temperature scaling. We will show later that temperature scaling for calibrating our definition of uncertainty in Bayesian deep learning, which takes into account all classes, does not have this shortcoming. More complex methods, such as a neural network as auxiliary recalibration model, have been used in calibrated regression (Kuleshov et al. 2018).

2.1.3 Methods

In this section, we discuss how model uncertainty is obtained by different approximate Bayesian inference techniques and how it can be calibrated with logit scaling. We define the expected uncertainty calibration error as a new metric to quantify miscalibration and describe confidence penalty as an alternative to logit scaling.

Uncertainty Estimation

In this work, we focus on uncertainty from approximately Bayesian methods. We assume a general multi-class classification task with C classes. Let input $\boldsymbol{x} \in \mathcal{X}$ be a random variable with corresponding label $y \in \mathcal{Y} = \{1, \dots, C\}$. Let $\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x})$ be the output (logits) of a neural network with parameters $\boldsymbol{\theta}$, and with model likelihood $p(y = c \,|\, \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}))$ for class c, which is sampled from a probability vector $\boldsymbol{p} = \boldsymbol{\sigma}_{\mathrm{SM}}(\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}))$, obtained by passing the model output through the softmax function $\boldsymbol{\sigma}_{\mathrm{SM}}(\cdot)$. From a frequentist perspective, the softmax likelihood is often interpreted as *confidence* of prediction. Throughout this thesis, we follow this definition.

The frequentist approach assumes a single best point estimate of the parameters (or weights) of a neural network. In frequentist inference, the weights of a deep model are obtained by maximum likelihood estimation (Bishop 2006), and the normalized output likelihood for an unseen test input does not consider uncertainty in the weights (Kendall and Gal 2017). Weight uncertainty (also referred to as model or epistemic uncertainty) is a considerable source of predictive uncertainty for models trained on data sets of limited size (Bishop 2006; Kendall and Gal 2017). Bayesian neural networks and recent advances in their approximation provide valuable mathematical tools for quantification of model uncertainty (Gal and Ghahramani 2016b; D. Kingma and Welling 2014). Instead of assuming the existence of a single best parameter set, we place distributions over the parameters and want to consider all possible parameter configurations, weighted by their posterior. More specifically, given a training data set $\mathcal{D}$ and an unseen test sample $\boldsymbol{x}$ with class label y, we are interested in evaluating the predictive distribution from Eq. (2.1). This integral requires to evaluate the posterior $p(\boldsymbol{\theta}|\mathcal{D})$, which involves the intractable marginal likelihood. A possible solution to this is to approximate the posterior with a more simple, tractable distribution $q(\boldsymbol{\theta})$ by optimization.

In the following, we briefly describe common approximately Bayesian methods which we use in our experiments to obtain weight uncertainty.

Monte Carlo Dropout One practical approximation of the posterior is variational inference with Monte Carlo (MC) dropout (Gal and Ghahramani 2016b). To determine model uncertainty, dropout variational inference is performed by training a model $\boldsymbol{f}_{\boldsymbol{\theta}}$ with dropout (Srivastava et al. 2014) and using dropout at test time to sample from the approximate posterior distribution by performing N stochastic forward passes per test sample (Gal and Ghahramani 2016b; Kendall and Gal 2017). This is also referred to as MC dropout. In MC dropout, the final probability vector of the predictive distribution is computed by MC integration:

$$\boldsymbol{p}(\boldsymbol{x}) = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{\sigma}_{\mathrm{SM}} \left(\boldsymbol{f}_{\boldsymbol{\theta}_i}(\boldsymbol{x}) \right) . \tag{2.2}$$

Gaussian Dropout Gaussian dropout was first proposed by S. Wang and Manning (2013) and linked to variational inference by D. Kingma, Salimans, et al. (2015). Dropout introduces Bernoulli noise during optimization and reduces overfitting of the training data. The resulting output $\boldsymbol{y}_k$ of layer k with dropout is a weighted sum of Bernoulli random variables. Then, the central limit theorem states, that $\boldsymbol{y}_k$ is approximately normally distributed. Instead of sampling from the weights and computing the resulting output, we can directly sample from the implicit Gaussian distribution of dropout

$$\boldsymbol{y}_k \sim \mathcal{N}(\mu_{\boldsymbol{y},k}, \sigma^2_{\boldsymbol{y},k}) \tag{2.3}$$

with

$$\mu_{\boldsymbol{y},k} = \mathbb{E}[y_k] = \sum_j w_{j,k} x_j \ , \tag{2.4}$$

$$\sigma^2_{\boldsymbol{y},k} = \mathrm{Var}[y_k] = p/(1-p) \sum_j w^2_{j,k} x^2_j \ , \tag{2.5}$$

using the reparameterization trick (D. Kingma, Salimans, et al. 2015)

$$y_{j,k} = \mu_{j,k} + \sigma_{j,k}\varepsilon_j \text{ with } \varepsilon_j \sim \mathcal{N}(0,1) \ . \tag{2.6}$$

Gaussian dropout is a continuous approximation to Bernoulli dropout, and in comparison it will better approximate the true posterior distribution and is expected to provide improved uncertainty estimates (Louizos and Welling 2017). To obtain the final probability vector $\boldsymbol{p}(\boldsymbol{x})$, we again use MC integration with N stochastic forward passes.

The dropout rate p is now a learnable parameter and does not need to be chosen carefully by hand. In fact, p could be optimized w.r.t. uncertainty calibration, scaling the variance of the implicit Gaussian of dropout. A similar approach was presented by Gal, Hron, et al. (2017) using the Concrete distribution (Jang et al. 2017; Maddison et al. 2016). However, we focus on metrics for measuring calibration and, therefore, fix p in our subsequent experiments (§ 2.1.4). Gaussian dropout has been used in the context of uncertainty estimation in prior work. In (Louizos and Welling 2017), it is used together with multiplicative normalizing flows to improve the approximate posterior. A similar Gaussian approximation of Batch Normalization was presented in (Teye et al. 2018), where Monte Carlo Batch Normalization is proposed as approximate Bayesian inference.

Bayes by Backprop Blundell et al. (2015) assume a Gaussian distribution with diagonal covariance matrix as variational posterior $q(\boldsymbol{\theta}|\boldsymbol{\phi})$, parameterized by mean $\boldsymbol{\mu}$ and standard deviation $\boldsymbol{\sigma}$, where $\boldsymbol{\phi} = \{\boldsymbol{\mu}, \boldsymbol{\sigma}\}$. A sample of the weights can be obtained by sampling a multivariate unit Gaussian and shift it by $\boldsymbol{\mu}$ and scale it by $\boldsymbol{\sigma}$. Then, the network is trained by minimizing

$$\mathcal{L}(\boldsymbol{\phi}) = \mathrm{KL}[q(\boldsymbol{\theta}|\boldsymbol{\phi}) \| p(\boldsymbol{\theta})] - \mathbb{E}_q[\log p(\mathcal{D}|\boldsymbol{\phi})] \ . \tag{2.7}$$

In case of a zero mean Gaussian prior, the first term can be implemented by weight decay. In contrast to Gaussian dropout, which operates on the implicit distribution of the activations, Bayes by Backprop (BBB) directly operates on the weights. This doubles the number of trainable parameters in practice. MC integration is used to obtain the final probability vector $\boldsymbol{p}(\boldsymbol{x})$.

SWA-Gaussian Stochastic weight averaging (SWA) uses stochastic gradient descent steps around a local loss optimum of a trained network and averages the weights $\boldsymbol{\theta}_{\text{SWA}} = \frac{1}{T}\sum_{i=1}^{T}\boldsymbol{\theta}_i$ of the model from each step i (Izmailov et al. 2018). This explores the loss landscape and averaging helps to find a better weight estimate than converging to a single local optimum. SWA-Gaussian (SWAG) is closely related to Bayes by Backprop (Maddox et al. 2019). It assumes a Gaussian distribution with diagonal covariance matrix as approximate variational posterior. Instead of using backpropagation to directly optimize $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$, it fits a Gaussian by using $\boldsymbol{\mu} = \boldsymbol{\theta}_{\text{SWA}}$ and

$$\boldsymbol{\Sigma}_{\text{diag}} = \operatorname{diag}(\overline{\boldsymbol{\theta}^2} - \boldsymbol{\theta}^2_{\text{SWA}}), \qquad \overline{\boldsymbol{\theta}^2} = \frac{1}{T}\sum_{i=1}^{T}\boldsymbol{\theta}_i^2 \,. \tag{2.8}$$

This doubles the number of parameters at test time. The approximate Gaussian posterior results to $\mathcal{N}(\boldsymbol{\theta}_{\text{SWA}}, \boldsymbol{\Sigma}_{\text{diag}})$ and MC integration with samples $\boldsymbol{\theta}_i \sim \mathcal{N}(\boldsymbol{\theta}_{\text{SWA}}, \boldsymbol{\Sigma}_{\text{diag}})$ is used to compute the final probability vector $\boldsymbol{p}(\boldsymbol{x})$.

Deep Ensembles Training multiple randomly initialized copies of a deep network by performing maximum posterior estimation and ensembling them to get multiple predictions for a single input is not a variational inference method per definition. However, they have been reported to produce surprisingly useful uncertainty estimates in practice that are better calibrated (Lakshminarayanan et al. 2017). Deep ensembles considerably increase the number of parameters at train and test time. We use deep ensembles as non Bayesian baseline for uncertainty estimation.

Related Calibration Metrics

In this subsection, we review related and commonly accepted calibration error metrics.

Expected Calibration Error The expected calibration error (ECE) is one of the most popular calibration error metrics and estimates model calibration by binning the predicted confidences $\hat{p} = \max_c p(y = c \,|\, \boldsymbol{x})$ into M bins from equidistant intervals and comparing them to average accuracies per bin (Guo et al. 2017; Naeini et al. 2015):

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} \big|\operatorname{acc}(B_m) - \operatorname{conf}(B_m)\big| \,, \tag{2.9}$$

with number of test samples n and $\operatorname{acc}(B)$ and $\operatorname{conf}(B)$ denoting the accuracy and confidence of bin B, respectively. Several recent works have described severe pathologies of the ECE metric (Ashukha et al. 2020; Ananya Kumar et al. 2019; Nixon et al. 2019). Most notably, the ECE metric is minimized by a model constantly predicting the marginal distribution of the majority class which makes it impossible to directly optimize it (Aviral Kumar et al.

2018). Additionally, the ECE only considers the maximum class probability and ignores the remaining entries of the probability vector $\boldsymbol{p}(\boldsymbol{x})$.

Adaptive Calibration Error Nixon et al. (2019) proposed the adaptive calibration error (ACE) to address the issue of fixed bin widths of ECE-like metrics. For models with high accuracy or overconfidence, most of the predictions fall into the rightmost bins, whereas only very few predictions fall into the rest of the bins. ACE spaces the bins such that an equal number of predictions contribute to each bin. The final ACE is computed by averaging over per-class ACE values to address the issue raised by Kull et al. (2019). However, this makes the metric more sensitive to the manually selected number of bins M as the number of bins effectively becomes $C \cdot M$, with number of classes C. Using fixed bin widths, the numbers of samples in the sparsely populated bins is further reduced, which increases the variance of each measurement per bin. Using adaptive bins, this results in the lower confidence bins spanning a wide range of values, which increases the bias of the bin's measurement.

Negative Log-Likelihood Deep models for classification are usually trained by minimizing the average negative log-likelihood (NLL):

$$\mathrm{NLL} = \frac{1}{N} \sum_{i=1}^{N} - \log p(y = y_i \mid \boldsymbol{x}_i) \ . \tag{2.10}$$

The NLL is also commonly used as a metric for measuring the calibration of uncertainty. However, the NLL is minimized by increasing the confidence $\max_c p(y = c \mid \boldsymbol{x})$, which favors over-confident models and models with higher accuracy (Ashukha et al. 2020). Therefore, this metric is unable to compare the calibration of models with different accuracies and training a model by minimizing NLL does not necessarily lead to good calibration.

Brier Score The average Brier score is another popular metric for assessing the quality of predictive uncertainty and is defined as (Brier 1950; Lakshminarayanan et al. 2017)

$$\mathrm{BS} = \frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} \left(\mathbf{1}(y_i = c) - p(y = c \mid \boldsymbol{x}_i)\right)^2 \ . \tag{2.11}$$

Similarly to the NLL, the Brier score favors high probabilities for correct predictions and low probabilities for incorrect predictions. Thus, models with higher accuracy tend to show a better Brier score, which makes the metric unsuitable for comparing the quality of uncertainty for models with different accuracies.

Maximum Mean Calibration Error Common recalibration methods are applied post-hoc, e.g., temperature scaling on a separate calibration set. Aviral Kumar et al. (2018)

proposed the maximum mean calibration error (MMCE), a trainable calibration surrogate for the calibration error. It is defined as

$$\mathrm{MMCE}^2(D) = \sum_{i,j \in D} \frac{(\mathbf{1}(\hat{y}_i = y_i) - \hat{p}_i)\,(\mathbf{1}(\hat{y}_j = y_j) - \hat{p}_j)\,k(\hat{p}_i, \hat{p}_j)}{m^2} \tag{2.12}$$

over mini-batch $D \subset \mathcal{D}$ with batch size m, matrix-valued universal kernel k and $\hat{y} = \arg\max_c p(y = c \,|\, \boldsymbol{x})$. Trainable calibration metrics are used in joint optimization with the negative log-likelihood

$$\arg\min_{\boldsymbol{\theta}} \sum_{D} \mathrm{NLL}(D, \boldsymbol{\theta}) + \lambda\,\mathrm{MMCE}(D, \boldsymbol{\theta}) \,. \tag{2.13}$$

Aviral Kumar et al. (2018) claim to have addressed the issue that the ECE is unsuitable for direct optimization due to its high discontinuity in $\boldsymbol{\theta}$. However, MMCE is also minimized by a model constantly predicting the marginal distribution of the classes (i.e., the probability mass function over the data set). This leads to subpar logit temperature when training with MMCE and temperature scaling can further reduce miscalibration (Aviral Kumar et al. 2018).

Interim Conclusion All of the aforementioned calibration metrics have considerable shortcomings and failure modes that prevent reliable measurement of miscalibration. They are either minimized by a random model, are more sensitive to hyperparameters (number of bins), or are influenced by the accuracy of the model. This prevents the practical use of the metrics to assess the uncertainty of deep models in the context of medical image analysis and other safety-critical tasks. In the following, we present a new calibration metric that attempts to avoid these pitfalls.

Uncertainty Calibration Error

To give an insight into our general approach to measuring the calibration of uncertainty, we will first revisit the definition of perfect calibration of confidence (Guo et al. 2017) and show how this concept can be extended to calibration of our definition of uncertainty.

Let $\hat{y} = \arg\max \boldsymbol{p}$ be the most likely class prediction of input $\boldsymbol{x}$ with confidence $\hat{p} = \max \boldsymbol{p}$ and true label y. Then, following Guo et al. (2017), *perfect calibration of confidence* is defined as

$$\Pr\left[\hat{y} = y \,|\, \hat{p} = \alpha\right] = \alpha, \quad \forall \alpha \in [0, 1] \,. \tag{2.14}$$

That is, the probability of a correct prediction $\hat{y} = y$ given the prediction confidence $\hat{p}$ should exactly correspond to the prediction confidence. Instead of using only the probability of the predicted class, we use the entropy of $\boldsymbol{p}$ to express prediction uncertainty:

$$\mathcal{H}(\boldsymbol{p}) = -\sum_{c=1}^{C} p^{(c)} \log p^{(c)} \,. \tag{2.15}$$

Let

$$\boldsymbol{q}(k) := (\mathbb{P}[y=1|\arg\max\ \boldsymbol{p}(\boldsymbol{x})=k],\dots,\mathbb{P}[y=C|\arg\max\ \boldsymbol{p}(\boldsymbol{x})=k]) \tag{2.16}$$

be a probability vector of true marginal class probabilities for all inputs $\boldsymbol{x}$ predicted with class k. Consider the following example: Three i.i.d. inputs $\boldsymbol{x}_{1:3}$ in a binary classification task with ground truth labels $\{1,1,2\}$ have all been predicted with $\arg\max\ \boldsymbol{p}(\boldsymbol{x}_{1:3})=1$. Then, $\boldsymbol{q}(1)=\left(\frac{2}{3},\frac{1}{3}\right)$. With this, we define a model to be perfectly calibrated if

$$\mathcal{H}(\boldsymbol{q}(k)) = \mathcal{H}(\boldsymbol{p}\,|\arg\max\ \boldsymbol{p}=k) \quad \forall\, k\in\{1,\dots,C\}\,. \tag{2.17}$$

From this, we derive an error metric for calibration of uncertainty:

$$\mathbb{E}_{\boldsymbol{p}}\big[\,|\mathcal{H}(\boldsymbol{q})-\mathcal{H}(\boldsymbol{p})|\,\big]\,. \tag{2.18}$$

However, this metric and the use of the entropy as measure of uncertainty lacks interpretability, as the entropy scales with the number of classes C. This does not allow to compare the uncertainty or the calibration of models trained on different data sets. Therefore, we propose to use the normalized entropy to scale the values to a range between 0 and 1:

$$\tilde{\mathcal{H}}(\boldsymbol{p}) := -\frac{1}{\log C}\sum_{c=1}^{C} p^{(c)}\log p^{(c)}\,, \quad \tilde{\mathcal{H}}\in[0,1]\,. \tag{2.19}$$

We further increase interpretability and argue, that the normalized entropy should correlate with the model error. From Eq. (2.14) and Eq. (2.19), we define *perfect calibration of uncertainty* as

$$\Pr\left[\hat{y}\neq y\,|\,\tilde{\mathcal{H}}(\boldsymbol{p})=\alpha\right]=\alpha, \quad \forall\alpha\in[0,1]\,. \tag{2.20}$$

That is, in a batch of inputs that are all predicted with uncertainty of e. g. 0.2, a top-1 error of $20\,\%$ is expected (the top-1 error is computed considering the class with highest probability only). The confidence is interpreted as the probability of belonging to a particular class, which should naturally correlate with the model error of that class. This characteristic does not generally apply to entropy, and thus the question arises why entropy should correspond with the model error.

Theorem 1. *The normalized entropy (uncertainty) $\tilde{\mathcal{H}}(\boldsymbol{p})$ approaches the top-1 error in the limit of number of classes C if the model $\mathbf{p}$ is well-calibrated.*

Proof. With Lemma 1 and $\hat{p}=\max\boldsymbol{p}$ we rewrite the normalized entropy as

$$\tilde{\mathcal{H}}(\boldsymbol{p}) = -\frac{\hat{p}\log\hat{p}}{\log C} - \frac{(1-\hat{p})\log\frac{1-\hat{p}}{C-1}}{\log C}\,. \tag{2.21}$$

Now, in the limit of number of classes C

$$\begin{aligned}
\lim_{C\to\infty} \tilde{\mathcal{H}}(\boldsymbol{p}) &= \lim_{C\to\infty} -\frac{(1-\hat{p})\log\frac{1-\hat{p}}{C-1}}{\log C} && (2.22)\\
&= \lim_{C\to\infty} -(1-\hat{p})\left(\frac{\log(1-\hat{p})}{\log C} - \frac{\log(C-1)}{\log C}\right) && (2.23)\\
&= (1-\hat{p}) && (2.24)
\end{aligned}$$

The top-1 error equals $(1-\hat{p})$ if the model is perfectly calibrated in the sense of Eq. (2.14). $\square$

Lemma 1. *Given a softmax output* $\mathbf{p}$ *with* C *entries and the most likely prediction* $\hat{y} = \arg\max\ \mathbf{p}$ *with likelihood* $\hat{p} = \max\mathbf{p}$*. Then, the remaining entries* $p_{i,i\neq\hat{y}}$ *are approximately uniformly distributed with probability* $\frac{1-\hat{p}}{C-1}$.

Proof. This assumption is approximately correct (1) if $\hat{p} \to 1$ or (2) if $C \to \infty$. Let $\tilde{p}_j = p_i\ \forall i \neq \hat{y}$ and $\tilde{q}_j = \frac{(1-\hat{p})}{C-1}$. Note that $\tilde{p}$ and $\tilde{q}$ are not proper probability distributions as $\sum \tilde{p}_j = \sum \tilde{q}_j = (1-\hat{p})$.

(1) Consider $\mathrm{KL}[\tilde{p}\|\tilde{q}]$ as $\hat{p}$ approaches 1:

$$\begin{aligned}
\lim_{\hat{p}\to 1} \mathrm{KL}\left[\tilde{p} \,\|\, \tilde{q}\right] &= \lim_{\hat{p}\to 1} \sum_{j=1}^{C-1} \tilde{p}_j \log\frac{\tilde{p}_j}{\tilde{q}_j} && (2.25)\\
&= \lim_{\hat{p}\to 1} \sum_{j=1}^{C-1} \tilde{p}_j \log\tilde{p}_j - \sum_{j=1}^{C-1} \tilde{p}_j \log\tilde{q}_j && (2.26)\\
&= \lim_{\hat{p}\to 1} \sum_{j=1}^{C-1} \tilde{p}_j \log\tilde{p}_j - (1-\hat{p})\log\frac{(1-\hat{p})}{C-1} && (2.27)\\
&= 0 && (2.28)
\end{aligned}$$

(2) Let z_i be the logits of a model trained with L2 regularization. The magnitude of the logits $|z_i|$ cannot become arbitrary large and due to the normalizing nature of softmax

$$\lim_{C\to\infty} \frac{\exp z_i}{\sum_{j=1}^{C} \exp z_j} = \lim_{C\to\infty} \frac{1}{C}\,. \qquad (2.29)$$

Alternatively, let $\boldsymbol{z} \in \mathbb{A}^C$ and $\boldsymbol{z}' \in \mathbb{B}^K$ be two logit vectors with $C < K$. If both models have been trained with L2 regularization, the magnitude of the logits $|z_i|, |z'_i|$ cannot become arbitrary large. More specifically, $\mathbb{A} = \mathbb{B} \subset \mathbb{R}$. Due to the normalizing nature of softmax, $\boldsymbol{z}'$ corresponds to a lower softmax temperature and as the temperature decreases with increasing number of classes, softmax approaches a uniform distribution (Jang et al. 2017).

$\square$

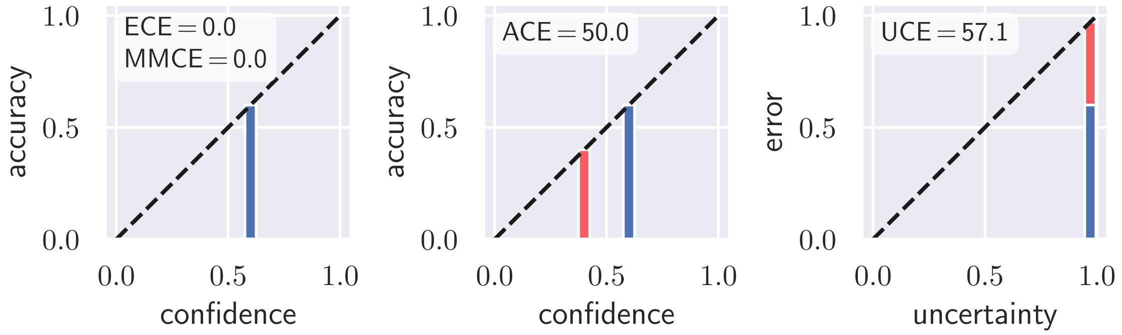


Figure 2.2: Calibration diagrams for a toy experiment with a degenerated model constantly predicting the marginal probabilities $\boldsymbol{p} = (0.6, 0.4)$ in a binary classification task. ECE and MMCE only consider $\max \boldsymbol{p}$ and fail at capturing the miscalibration of class 2 with $p(c = 2) = 0.4$, but $\mathrm{acc}(c = 2) = 0$. The red bars show the measured miscalibration. Uncertainty is given as normalized entropy. The left diagram is computed using ECE as MMCE does not involve binning.

Thus, the normalized entropy gives us an intuitive and interpretable measure of uncertainty. If a model is perfectly calibrated, $\tilde{\mathcal{H}}$ corresponds to the top-1 error. We propose the following notion to quantify miscalibration of uncertainty:

$$\mathbb{E}_{\tilde{\mathcal{H}}}\Big[\big|\Pr\left[\hat{y} \neq y \,|\, \tilde{\mathcal{H}}(\boldsymbol{p}) = \alpha\right] - \alpha\big|\Big], \quad \forall \alpha \in [0, 1]\,. \tag{2.30}$$

We refer to this as Expected Uncertainty Calibration Error (UCE) and approximate with

$$\mathrm{UCE} := \sum_{m=1}^{M} \frac{|B_m|}{n} \big|\mathrm{err}(B_m) - \mathrm{uncert}(B_m)\big|\,, \tag{2.31}$$

using the same binning scheme as in ECE estimation. The error per bin is defined as

$$\mathrm{err}(B_m) := \frac{1}{|B_m|} \sum_{i \in B_m} \mathbf{1}(\hat{y}_i \neq y)\,, \tag{2.32}$$

where $\mathbf{1}(\hat{y}_i \neq y) = 1$ and $\mathbf{1}(\hat{y}_i = y) = 0$. Uncertainty per bin is defined as

$$\mathrm{uncert}(B_m) := \frac{1}{|B_m|} \sum_{i \in B_m} \tilde{\mathcal{H}}(\boldsymbol{p}_i)\,. \tag{2.33}$$

Properties of UCE The proposed UCE metric solves several problems of other metrics. First, the UCE is not zero for a model constantly predicting the marginal class distribution. Estimators of metrics with this pathology (e.g., ECE, MMCE, see Fig. 2.2) suffer from varying bias and, therefore, do not allow comparing miscalibration of different models

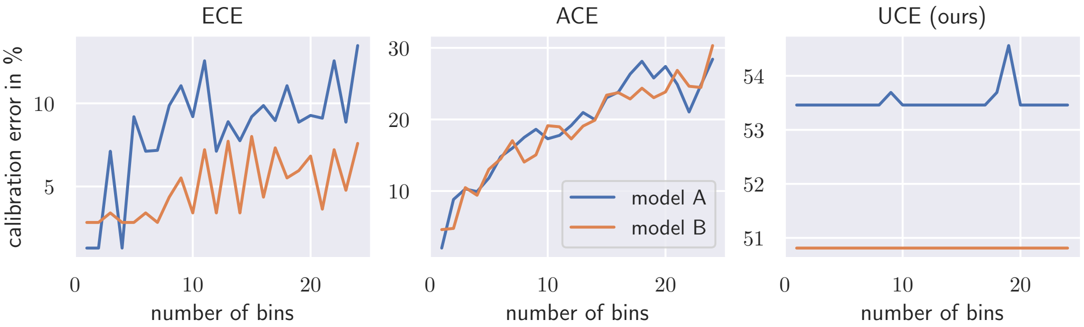


Figure 2.3: Toy experiment with two random models A and B in a binary classification task. UCE is less sensitive to number of bins used in the estimator and provides a consistent ranking of the models. For results from multi-class experiments, see Fig. 2.2.

(Ashukha et al. 2020; Vaicenavicius et al. 2019). In contrast to ACE, UCE is not highly sensitive to the numbers of bins and provides a consistent ranking of different models for the same classification task (see Fig. 2.3 and 2.5). Additionally, UCE can be used as a trainable regularizer in similar manner to MMCE. During training, we compute the UCE over mini-batches $D \subset \mathcal{D}$ and add it to the NLL training objective

$$\arg\min_{\boldsymbol{\theta}} \sum_{D} \mathrm{NLL}(D, \boldsymbol{\theta}) + \lambda\, \mathrm{UCE}(D, \boldsymbol{\theta}) \ , \tag{2.34}$$

weighted by a factor λ. UCE is zero for an optimal model and thus does not penalize high confident predictions for models with high accuracy, which is a major disadvantage of plain entropy regularization (Pereyra et al. 2017). Predictions with low uncertainty, but high top-1 error are penalized whereas predictions with high accuracy are encouraged to have low uncertainty.

Temperature Scaling for Variational Inference

State-of-the-art deep neural networks are generally miscalibrated with regard to softmax likelihood (Guo et al. 2017). However, when obtaining model uncertainty with variational inference, this also tends to be not well-calibrated (Gal, Hron, et al. 2017; Lakshminarayanan et al. 2017; Louizos and Welling 2017). Fig. 2.1 (left) shows reliability diagrams (Niculescu-Mizil and Caruana 2005) for ResNet-101 trained on CIFAR-100. The divergence from the identity function reveals miscalibration. Furthermore, it is not possible to robustly detect OoD data from uncalibrated uncertainty (see Fig. 2.1 (right)). If the fraction of OoD data in a batch of test images is $> 50\,\%$, there is almost no increase in mean uncertainty. We first address the problem using temperature scaling, which is the most straightforward logit scaling method for recalibration.

Temperature scaling for variational inference is derived by closely following the derivation of frequentist temperature scaling in (Guo et al. 2017). Let $\{\boldsymbol{z}_{1,j}, \ldots, \boldsymbol{z}_{N,j}\}$ be a set of logit vectors obtained by MC integration with N stochastic forward passes for each input $\boldsymbol{x}_j \in \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M\}$ with true labels $\{y_1, \ldots, y_M\}$. Temperature scaling is the solution $\hat{p}$ to entropy maximization

$$\max_{\hat{p}} \; -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{M} \sum_{c=1}^{C} \hat{p}\left(\boldsymbol{z}_{i,j}\right)^{(c)} \log \hat{p}\left(\boldsymbol{z}_{i,j}\right)^{(c)} , \tag{2.35}$$

subject to

$$\hat{p}(\boldsymbol{z}_{i,j})^{(c)} \geq 0 \quad \forall i, j, c \, , \tag{2.36}$$

$$\sum_{c=1}^{C} \hat{p}(\boldsymbol{z}_j)^{(c)} = 1 \quad \forall j \, , \tag{2.37}$$

$$\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{M} z_{i,j}^{(y_j)} = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{M} \sum_{c=1}^{C} z_{i,j}^{(c)} \hat{p}(\boldsymbol{z}_{i,j})^{(c)} . \tag{2.38}$$

Proof. Guo et al. (2017) solve this constrained optimization problem with the method of Lagrange multipliers. We skip reviewing their proof as one can see that the solution to $\hat{p}$ in the case of MC integration provides

$$\frac{1}{N} \sum_{i=1}^{N} \hat{p}_i\left(\boldsymbol{z}_j\right)^{(c)} = \frac{1}{N} \sum_{i=1}^{N} \frac{e^{\lambda z_{i,j}^{(c)}}}{\sum_{\ell=1}^{C} e^{\lambda z_{i,j}^{(\ell)}}} \tag{2.39}$$

$$= \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{\sigma}_{\mathrm{SM}}\left(\lambda \boldsymbol{f}_{\boldsymbol{\theta}_i}(\boldsymbol{x}_j)\right)^{(c)} , \tag{2.40}$$

which yields temperature scaling for $\lambda = \tau^{-1}$ (Guo et al. 2017). ☐

A scalar parameter cannot rescale the class logits individually. Thus, more complex logit scaling can be derived by using any function at this point to smoothen or sharpen the softmax output (see next section). In this work temperature scaling with $\tau > 0$ is inserted before final softmax activation and before MC integration:

$$\hat{\boldsymbol{p}}(\boldsymbol{x}) = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{\sigma}_{\mathrm{SM}}\left(\tau^{-1} \boldsymbol{f}_{\boldsymbol{\theta}_i}(\boldsymbol{x})\right) . \tag{2.41}$$

First, $\boldsymbol{f}_{\boldsymbol{\theta}}$ is trained until convergence on the training set. Next, we fix the parameters $\boldsymbol{\theta}$ and optimize τ with respect to the negative log-likelihood on a separate calibration set using variational inference. This is equivalent to maximizing the entropy of $\hat{\boldsymbol{p}}$ (Guo et al. 2017).

Classwise Logit Scaling

It is stated by Kull et al. (2019) that temperature scaling would be inferior to more complex calibration methods when compared by means of classwise calibration. In (Guo et al. 2017), temperature scaling is used to calibrate the confidence that takes into account only one class probability. In contrast, we use temperature scaling to calibrate the model uncertainty, expressed via normalized entropy. This considers all class probabilities and thus, we hypothesize that temperature scaling implicitly leads to well-calibrated classwise uncertainty.

To demonstrate this experimentally, we implement vector scaling and auxiliary scaling and compare them using classwise UCE. *Vector scaling* is a multi-class extension of temperature scaling, where an individual scaling factor for each class is used to scale the final softmax output:

$$\hat{\boldsymbol{p}}_i(\boldsymbol{x}) = \boldsymbol{\sigma}_{\text{SM}}\left(\boldsymbol{T}\boldsymbol{f}_{\boldsymbol{\theta}_i}(\boldsymbol{x})\right) \ , \tag{2.42}$$

with $\boldsymbol{T} = \operatorname{diag}(\tau_1, \dots, \tau_C)$. *Auxiliary scaling* makes use of a more powerful auxiliary recalibration model $\boldsymbol{R}_{\boldsymbol{\phi}}$ consisting of a two-layer fully-connected network with C hidden units and leaky ReLU activations after the hidden layer:

$$\hat{\boldsymbol{p}}_i(\boldsymbol{x}) = \boldsymbol{\sigma}_{\text{SM}}\left(\boldsymbol{R}_{\boldsymbol{\phi}}(\boldsymbol{f}_{\boldsymbol{\theta}_i}(\boldsymbol{x}))\right) \ , \tag{2.43}$$

which is inspired by (Kuleshov et al. 2018). The intuition behind this is that recalibration may require a more complex function than simple scaling. Both $\boldsymbol{T}$ and the parameters $\boldsymbol{\phi}$ of the auxiliary model are optimized w.r.t. negative log-likelihood in a separate calibration phase by gradient descent. We initialize with $\tau_j \leftarrow 1$ and $\boldsymbol{\phi}_{1,2} \leftarrow \boldsymbol{I}_C$, respectively. Thus, recalibration is started form the identity function.

It must be emphasized that in contrast to temperature scaling, both vector and aux scaling can change the maximum of the softmax and thus affect model accuracy.

Confidence Penalty

Additionally, we compare temperature scaling to entropy regularization, where low entropy output distributions are penalized by adding the negative entropy $\mathcal{H}$ of the softmax output to the negative log-likelihood training objective, weighted by an additional hyperparameter λ. This leads to the following optimization function:

$$\mathcal{L}_{\text{CP}}(\boldsymbol{\theta}) = -\sum_{\mathcal{X},\mathcal{Y}} \log \boldsymbol{p}_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x}) - \lambda \mathcal{H}\left(\boldsymbol{p}_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x})\right) \ . \tag{2.44}$$

We reproduce the experiment of Pereyra et al. (2017) on supervised image classification and compare the quality of calibration of confidence and uncertainty to logit scaling calibration methods. Calibration by confidence penalty must be performed during the training and cannot be done afterwards. Thus, a separate calibration phase is omitted.

2.1.4 Experiments

The experimental results are presented twofold: First, we evaluate the proposed uncertainty calibration error metric and second, we evaluate the proposed logit scaling methods.

Evaluation of Uncertainty Calibration Error Metric

The uncertainty calibration error is evaluated on multi-class image classification on CIFAR-10 with ResNet-34 and on CIFAR-100 with ResNet-50 (He, X. Zhang, et al. 2016; Krizhevsky and Hinton 2009). We opt not to use medical data sets here as there are no large open access medical data sets with a high number of classes available and because the CIFAR data sets are well accepted in the machine learning community. However, all experiments and results can directly be translated to the medical domain.

The feature extractor of ResNet is used as implemented in PyTorch 1.6 (Paszke et al. 2019) and the last linear layer is implemented using the different Bayesian approximations from § 2.1.3. All models were trained from random initialization. We employed early stopping at highest validation set accuracy. Additional details on the training procedure can be found in the chapter appendix.

First, we compute the accuracies and all calibration error metrics from § 2.1.3 and the UCE on the test sets of CIFAR-10/100 (Krizhevsky and Hinton 2009) for all models. We investigate the effect of the number of bins in the estimators of the metrics involving binning and analyze the ranking of different models under varying softmax temperature τ. Next, we train a ResNet on the data sets CIFAR-10/100, SVHN (Netzer et al. 2011) and Fashion-MNIST (H. Xiao et al. 2017) with added calibration error regularization as in Eq. (2.13) and (2.34). We compare UCE regularization ($\lambda = 10$) to regularization with MMCE ($\lambda = 10$) and to confidence penalty with $\lambda = 0.1$, which penalizes the entropy of the probability vector $\boldsymbol{p}$ (Pereyra et al. 2017). The values for λ have been selected following Aviral Kumar et al. (2018) and Pereyra et al. (2017). We combine the regularization experiments with post-hoc calibration using temperature scaling (Guo et al. 2017).

Additionally, we analyze the utility of the normalized entropy as a measure of uncertainty and perform rejection and out-of-distribution (OoD) detection experiments using $\tilde{\mathcal{H}}$. We define an uncertainty threshold $\mathcal{H}_{\max}$ and reject all predictions from the test set where $\tilde{\mathcal{H}}(\boldsymbol{p}) > \mathcal{H}_{\max}$. A decrease in false predictions of the remaining test set is expected. To demonstrate the OoD detection ability, we provide images from CIFAR-100 to a deep model trained on CIFAR-10 (note that both CIFAR data sets have no mutual classes). In this experiment, we compose a batch of 100 random samples from the test set of the training domain and stepwise replace images with out-of-distribution data. In practice, it is expected that models are applied to a mix of known and unknown classes. After each step, we evaluate the mean batch uncertainty and expect, that the mean uncertainty monotonically increases as a function of the fraction of OoD data.

Evaluation of Logit Scaling

The experimental evaluation of logit scaling for variational inference is presented threefold: First, the proposed logit scaling methods are used to calibrate confidence and uncertainty and are compared with entropy regulation; second, predictions with high uncertainty are rejected; and third, the effect of out-of-distribution data on uncertainty is analyzed. All models were trained from random initialization. In the logit scaling experiments we focus on the use of Gaussian dropout as Bayesian approximation.

Uncertainty Calibration To show the effectiveness of uncertainty calibration, we train ResNet-34 (He, X. Zhang, et al. 2016) and DenseNet-121 (Huang et al. 2017) on the data sets CIFAR-10 and SVHN, as well as ResNet-101 and DenseNet-169 on CIFAR-100 with Gaussian dropout until convergence. Additionally, we reproduce the experiments of (Guo et al. 2017) and analyze calibration of frequentist confidence $\hat{p} = \max \boldsymbol{p}$ along with likelihood values $\hat{p} = \max N^{-1} \sum_{i=1}^{N} \boldsymbol{p}_i$ from MC integration. Subsequently, the models are calibrated using the previously mentioned logit scaling methods. The validation set with 5,000 images is used as calibration set. We additionally train all networks in the exact same manner with confidence penalty loss with fixed $\lambda = 0.1$. The proposed UCE and classwise UCE metrics are used to quantify calibration of uncertainty. Reliability diagrams (top-1 error vs. uncertainty) are used to visualize miscalibration. Classwise UCE values are given in Tab. 2.3 and the reliability diagrams show the corresponding UCE.

Rejection of Uncertain Predictions An example application of well-calibrated uncertainty is the rejection of uncertain predictions. In a medical imaging scenario, a critical decision should only be made on the basis of reliable predictions. We define an uncertainty threshold $\mathcal{H}_{\text{max}}$ and reject all predictions from the test set where $\tilde{\mathcal{H}}(\boldsymbol{p}) > \mathcal{H}_{\text{max}}$. A decrease in false predictions of the remaining test set is expected.

Out-of-Distribution Detection Deep neural networks only provide reliable predictions for data on which they have been trained. In practice, however, the trained network will encounter samples that lie outside the distribution of the training data. Problematically, a miscalibrated model will still produce highly confident estimates for such out-of-distribution (OoD) data (S. Lee et al. 2018).

Bayesian neural networks have not been extensively studied for out-of-distribution detection. Epistemic uncertainty from MC dropout was successfully used to detect OoD samples in neural machine translation (T. Z. Xiao et al. 2019). We reproduce the experiments presented by Lakshminarayanan et al. (2017), where predictive uncertainty obtained from deep ensembles is used to detect if data from CIFAR-10 is provided to a network trained on SVHN. They state that uncertainty produced by VI is overconfident and cannot robustly detect OoD data. We expect that well-calibrated uncertainty from Bayesian methods allows us to detect if data from CIFAR-10 is presented to a deep model trained on SVHN. However, the SVHN data set shows house numbers and the CIFAR data set contains everyday objects and animals; the data domains are overly disjoint. To demonstrate the OoD detection ability under more difficult conditions, we additionally provide images from CIFAR-100 to a deep model trained on CIFAR-10. These experiments are similarly conducted as the previous OoD detection experiment.

Training Settings

For the experiments, the model implementations from PyTorch 1.6 (Paszke et al. 2019) are used and trained with following settings:

- batch size of 256
- AdamW optimizer (Loshchilov and Hutter 2019) with initial learn rate of 0.01 and $\beta_1 = 0.9, \beta_2 = 0.999$
- weight decay of 1e-4
- negative log-likelihood (cross entropy) loss
- reduce-on-plateau learn rate scheduler (patience of 20 epochs) with factor of 0.1
- additional validation set is randomly extracted from the training set $(5,000$ samples)
- only the last linear layer is implemented in a Bayesian manner for MC dropout, Gaussian dropout, BayesByBackprop and SWAG
- the deep ensemble comprises 3 fully individually trained networks
- $N = 25$ forward passes were used Monte Carlo integration
- in MC dropout and Gaussian dropout, a dropout rate of $p = 0.2$ was used
- in SWAG, a learn rate of 3e-6 was used during weight averaging

2.1.5 Results

The results in this section are presented twofold. We first confer results for the evaluation of UCE as calibration metric and subsequently for logit scaling as recalibration method.

Results for Evaluation of Uncertainty Calibration Error

This section evaluates the uncertainty calibration error as suitable metric to measure miscalibration of Bayesian uncertainty. We start by comparing UCE to other metrics, investigate the use of UCE as uncertainty regularization and explain, why UCE regularization works. Further, we provide results for the rejection and OoD detection experiments.

Table 2.1: Classification accuracy and calibration error results for different models on CIFAR-10/100. We used $M = 15$ bins where necessary. Here, all metrics provide the same ranking of models.

Bayes	Dataset	Accuracy	ECE	ACE	UCE	MMCE	Brier	NLL
MC Drop	CIFAR-10	93.6 %	4.3 %	4.3 %	4.0 %	3.8 %	0.11	0.31
Gauss Drop	CIFAR-10	93.2 %	4.4 %	4.4 %	4.1 %	3.8 %	0.11	0.31
BBB	CIFAR-10	93.3 %	4.6 %	4.6 %	4.4 %	4.1 %	0.11	0.34
SWAG	CIFAR-10	94.4 %	3.7 %	3.7 %	3.5 %	3.5 %	0.09	0.28
Ensemble	CIFAR-10	95.0 %	3.2 %	3.2 %	3.0 %	2.8 %	0.08	0.22
MC Drop	CIFAR-100	66.9 %	24.4 %	24.5 %	27.9 %	20.6 %	0.55	2.55
Gauss Drop	CIFAR-100	66.5 %	24.5 %	24.7 %	28.2 %	20.7 %	0.56	2.64
BBB	CIFAR-100	65.1 %	24.9 %	25.1 %	28.9 %	20.9 %	0.57	2.51
SWAG	CIFAR-100	68.3 %	21.8 %	22.1 %	25.7 %	18.3 %	0.52	2.26
Ensemble	CIFAR-100	72.5 %	19.2 %	19.4 %	22.5 %	16.1 %	0.45	1.82

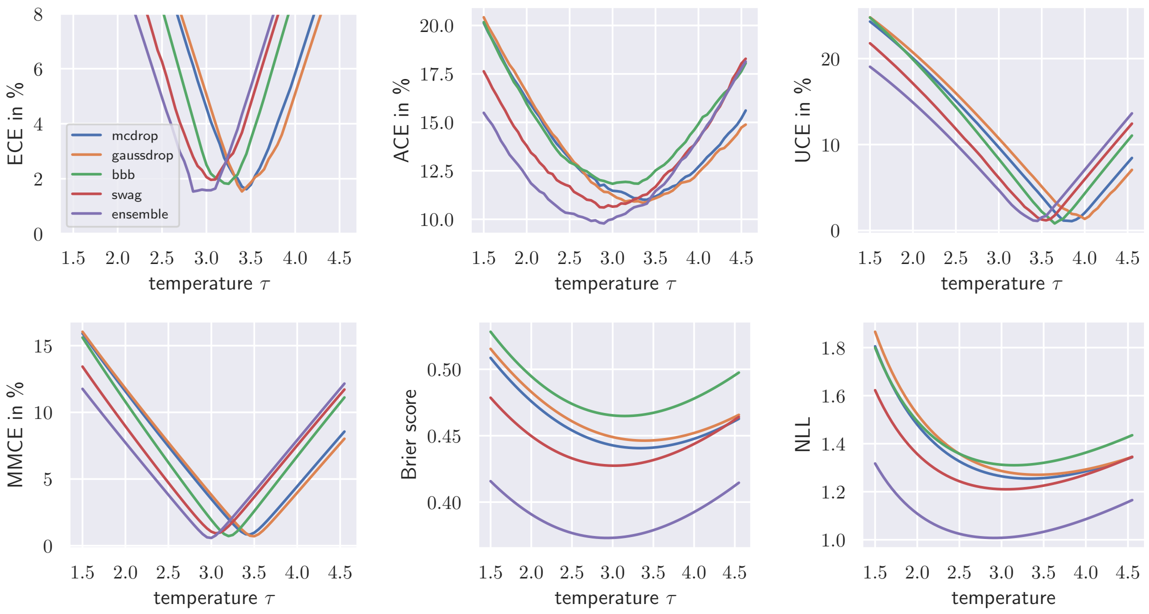


Figure 2.4: Calibration error vs. softmax temperature on CIFAR-100 (also see Fig. A.1 in the appendix).

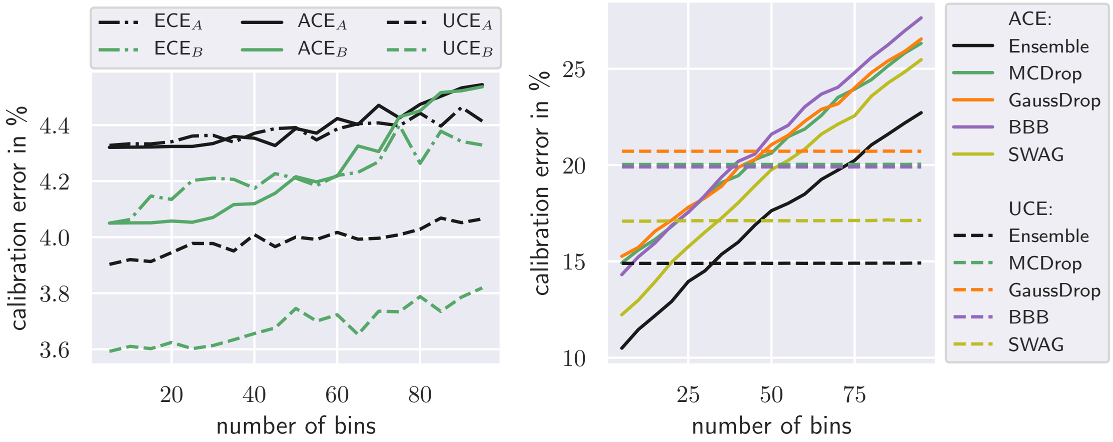


Figure 2.5: (Left) Calibration error values for two individually trained ResNet models A and B on CIFAR-10 test set. (Right) ACE and UCE values for different Bayesian methods on CIFAR-100 test set. ECE and ACE are very sensitive to the number of bins used in the estimator, not yielding a consistent ranking of the models. UCE is less sensitive to the bin number and ranks models consistently, allowing comparison of different models.

Comparison of Calibration Error Metrics Table 2.1 shows test set accuracy and all calibration error results for all model/data set configurations. Without any post-hoc calibration, such as temperature scaling, all metrics provide the same ranking of the models. The deep ensemble and SWAG perform best in terms of test set accuracy and calibration of uncertainty. Brier score and NLL are both highly sensitive to the model accuracy, which is especially apparent on CIFAR-10. For the first three models with similar accuracy, the Brier scores differ only marginally. Thus, both the Brier score and the NLL are unsuitable for comparing the calibration of different models. Ashukha et al. (2020) propose to use the calibrated NLL at optimal temperature for model comparison. However, Fig. 2.4 (and Fig. A.1 in the appendix) plot the metrics over varying softmax temperature and show, that the models with highest accuracy have lowest Brier and NLL, regardless of the temperature. From this we deduce that both Brier and NLL should not be used for comparison of multi-class calibration, even at optimal temperature. The remaining metrics show consistent ranking before and after the point of optimal temperature. The metrics ECE, UCE and MMCE have a narrow region in which the optimal temperature for all models can be found. This allows comparison of calibration of models if they are all over- or underconfident. However, all metrics fail at comparing underconfident models to overconfident models (see model ranking left and right of optimal temperature in Fig. 2.4).

Fig. 2.5 shows the effect of the number of bins M in the estimators of ECE, ACE and UCE. Both ECE and ACE are more sensitive to the number of bins and do not provide a consistent ranking of models under varying bin count. This is due to the fact that fewer bins are populated using $\tilde{\mathcal{H}}$ as uncertainty (cf. Fig. A.4 in the chapter appendix). This can

Table 2.2: Results from SWAG trained with entropy, MMCE and UCE regularization at optimal temperature (+T). All regularization methods considerably reduce miscalibration. We used the weighted MMCE implementation as proposed by Aviral Kumar et al. (2018).

Regularization	Dataset	Accuracy	ECE	ACE	UCE	MMCE	Brier	NLL
unregularized	CIFAR-10	**94.3** %	3.8 %	3.8 %	3.6 %	3.3 %	**0.10**	0.28
Entropy+T	CIFAR-10	94.1 %	2.1 %	4.2 %	2.3 %	1.1 %	**0.10**	0.25
MMCE+T	CIFAR-10	92.0 %	**0.4** %	**1.6** %	0.8 %	**0.1** %	0.12	0.24
UCE+T (ours)	CIFAR-10	92.6 %	0.5 %	**1.6** %	**0.7** %	0.2 %	**0.10**	**0.21**
unregularized	CIFAR-100	68.3 %	21.8 %	22.0 %	25.7 %	18.3 %	0.52	2.26
Entropy+T	CIFAR-100	68.1 %	2.9 %	12.3 %	3.7 %	2.1 %	0.44	1.41
MMCE+T	CIFAR-100	67.7 %	**1.3** %	11.0 %	2.1 %	**0.5** %	0.43	1.20
UCE+T (ours)	CIFAR-100	**70.9** %	2.4 %	**10.4** %	**1.1** %	1.2 %	**0.40**	**1.10**
unregularized	SVHN	96.8 %	2.10 %	2.16 %	1.89 %	1.82 %	0.05	0.19
Entropy+T	SVHN	96.9 %	1.15 %	2.31 %	0.86 %	0.74 %	0.05	0.15
MMCE+T	SVHN	**97.1** %	**0.27** %	**0.85** %	**0.35** %	0.17 %	0.05	**0.12**
UCE+T (ours)	SVHN	**97.1** %	0.38 %	0.92 %	0.38 %	**0.14** %	0.05	**0.12**
unregularized	F-MNIST	94.7 %	3.97 %	3.96 %	3.85 %	3.60 %	0.09	0.29
Entropy+T	F-MNIST	94.7 %	1.86 %	4.28 %	2.13 %	0.96 %	0.09	0.24
MMCE+T	F-MNIST	94.7 %	0.54 %	**1.40** %	0.64 %	0.17 %	**0.08**	**0.15**
UCE+T (ours)	F-MNIST	**94.8** %	**0.52** %	1.75 %	**0.63** %	**0.11** %	**0.08**	0.16

be interpreted as possible downside of the UCE metric as the adaptive binning scheme of ACE explicitly addresses that. However, we argue that consistent ranking due to robustness against bin count results in a metric that is more useful in practice.

Uncertainty Regularization Tab. 2.2 shows results from SWAG trained with entropy, MMCE and UCE regularization. All regularization methods considerably reduce miscalibration compared to unregularized models, as shown by all calibration metrics. At optimal temperature (as suggested by Ashukha et al. (2020)), UCE and MMCE regularization considerably reduce miscalibration for all employed calibration metric outperforming entropy regularization, with UCE achieving highest accuracy on CIFAR-100, SVHN and Fashion-MNIST. We want to stress out that UCE, in contrast to MMCE, was not specifically designed for the use as a calibration regularizer (Aviral Kumar et al. 2018). UCE regularization can be interpreted as entropy penalization for predictions with low accuracy. As UCE is zero for an optimal model, it encourages a model to reach high accuracy.

Why UCE Regularization Works UCE regularization works best when computed class-wise (in similar manner to ACE): $\mathrm{cUCE} = \frac{1}{C}\sum_{c=1}^{C} \mathrm{UCE}(c)$, where $\mathrm{UCE}(c)$ is computed

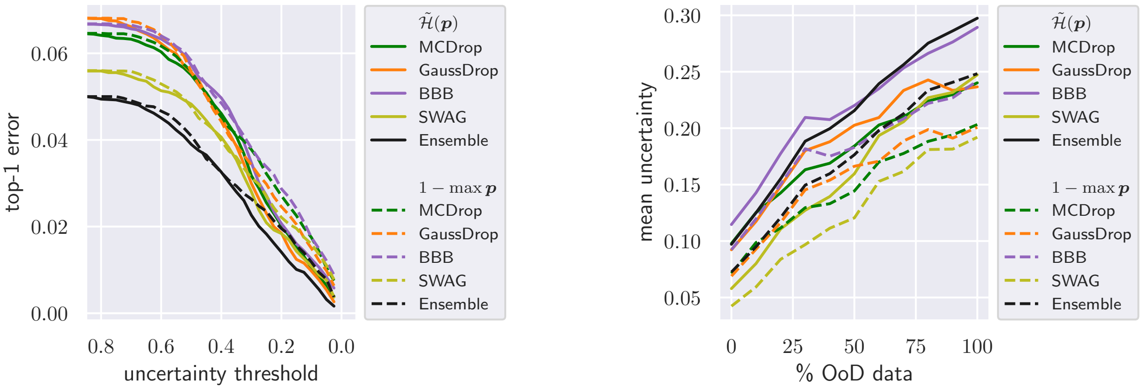


Figure 2.6: (Left) Rejection results on CIFAR-10 for decreasing uncertainty threshold comparing $\tilde{\mathcal{H}}(\boldsymbol{p})$ and $\max \boldsymbol{p}$ as uncertainty metric. In both cases, the top-1 error decreases strictly monotonically with decreasing threshold. (Right) Out-of-distribution detection for CIFAR-10 → CIFAR-100. The normalized entropy $\tilde{\mathcal{H}}(\boldsymbol{p})$ as measure of uncertainty can be used to robustly detect OoD data.

for predictions of class c. Consider the following binary classification example: A batch with mainly samples from class 1 and few samples from class 2 are all predicted as class 1 with high confidence. NLL further pushes the confidence of the predictions to 1.0, favoring overconfidence, whereas UCE is only reduced if the confidence of the overconfidently false predictions is reduced.

Rejection and OoD Detection Fig. 2.6 (left) shows the top-1 error as a function of decreasing uncertainty threshold $\mathcal{H}_{\max}$ and (right) shows the mean batch uncertainty at increasing OoD data. Robust rejection of uncertain predictions and detection of OoD data based on the normalized entropy $\tilde{\mathcal{H}}(\boldsymbol{p})$ is possible and is generally more sensitive to OoD data than the confidence $\max \boldsymbol{p}$.

Results for Evaluation of Logit Scaling

In the second part of this section, we present results for logit scaling as post-hoc calibration method by measuring uncertainty calibration, rejection experiments, and OoD detection.

Uncertainty Calibration Tab. 2.3 reports classwise UCE test set results and Fig. 2.7 shows reliability diagrams for the experimental setup described in the previous section. All logit scaling methods considerably reduce miscalibration on CIFAR-10/100 by means of cECE and cUCE. For the smaller networks on CIFAR-10 and SVHN, the more powerful aux scaling yields lowest cUCE. On CIFAR-100, however, aux scaling increases miscalibration. In this case, the auxiliary model $\boldsymbol{R}$ has $C = 100$ units in the hidden layer and easily overfits the calibration set (we observe calibration set accuracy of $100\,\%$). This results in worse

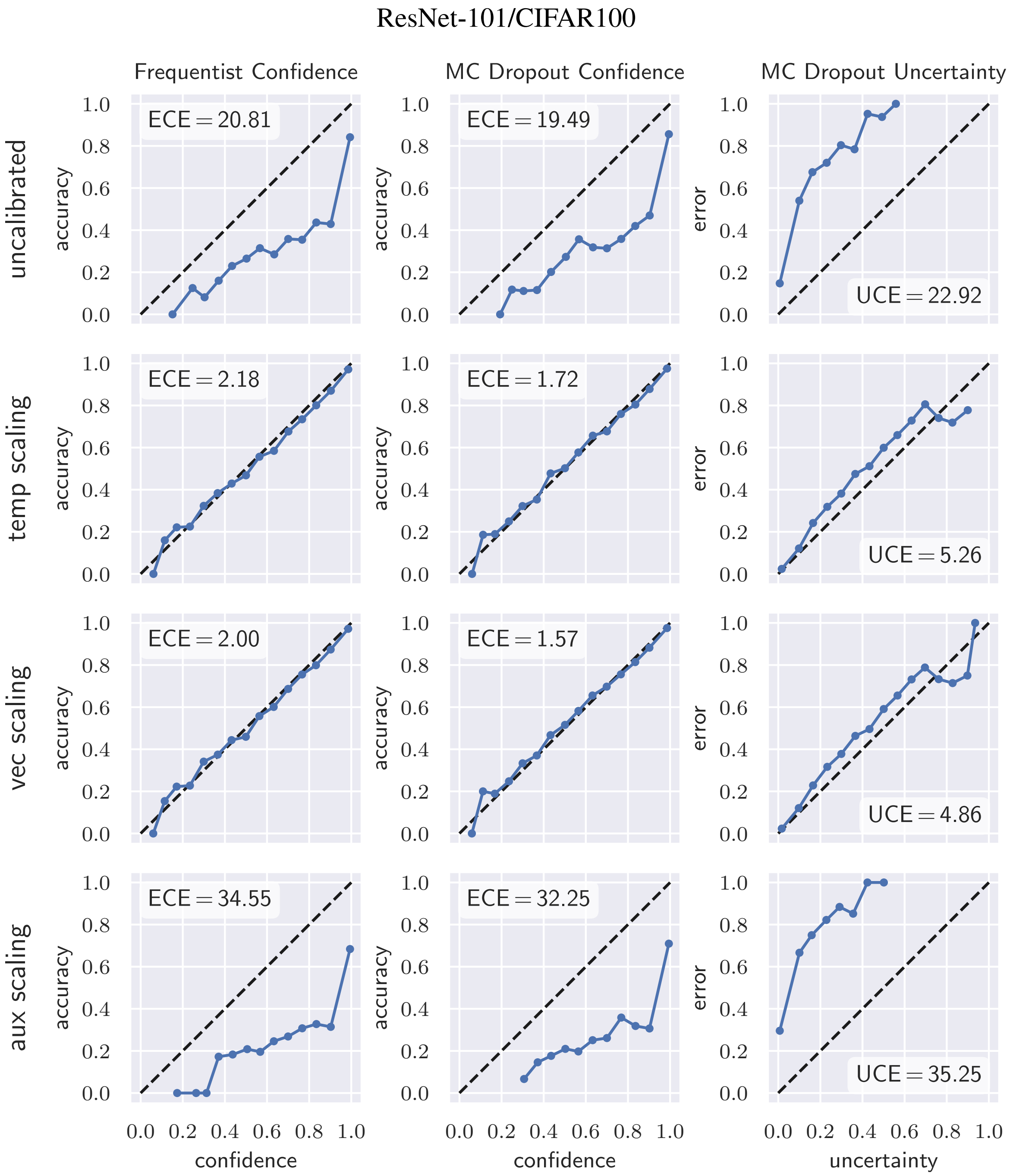


Figure 2.7: Reliability diagrams ($M = 15$ bins) on CIFAR-100 for ResNet-101. Top row: Uncalibrated frequentist confidence, and likelihood and uncertainty obtained by MC Gaussian dropout. The following rows show the results of the logit scaling methods. The dotted lines illustrates perfect calibration. Additional diagrams can be found in the chapter appendix.

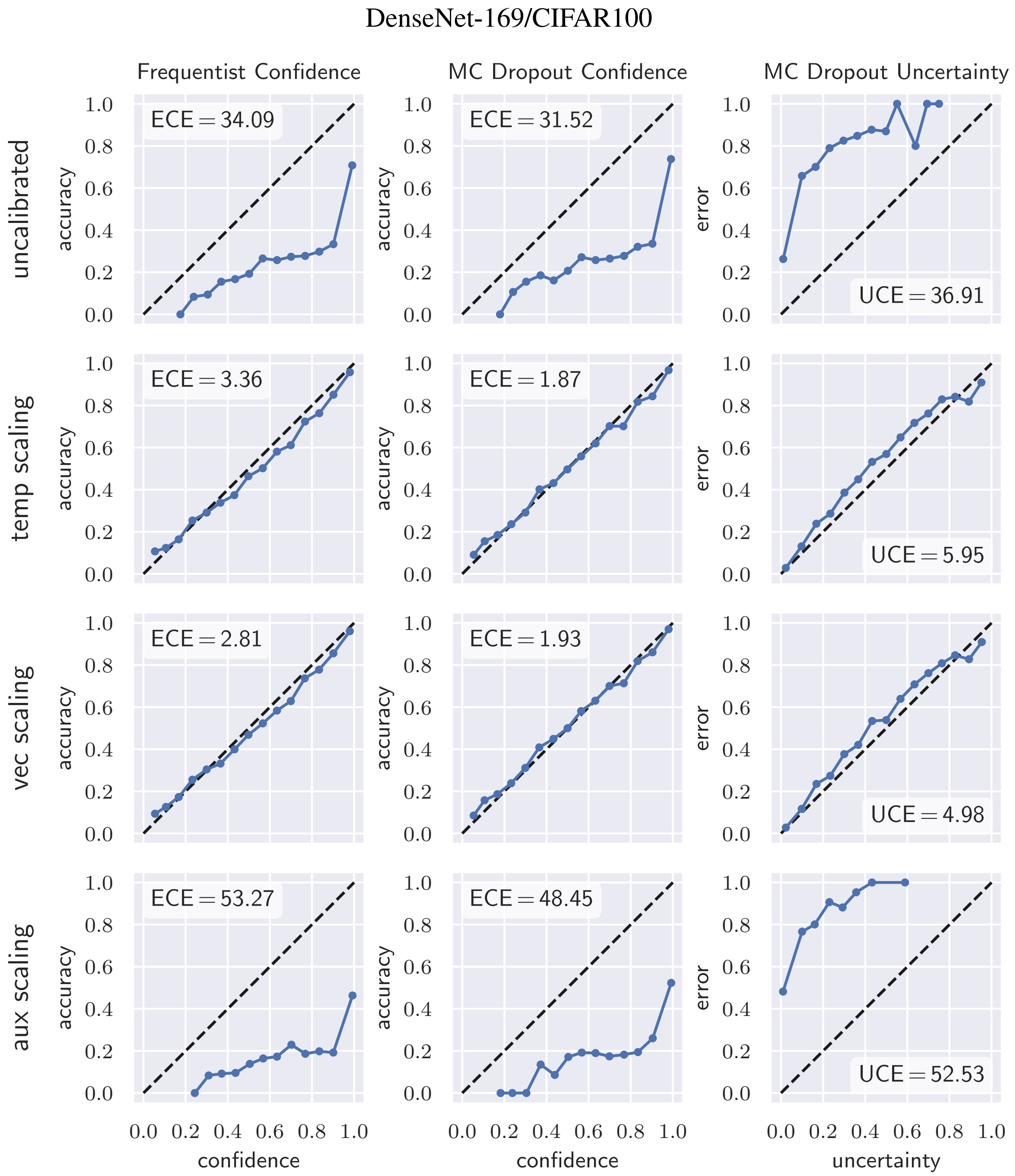
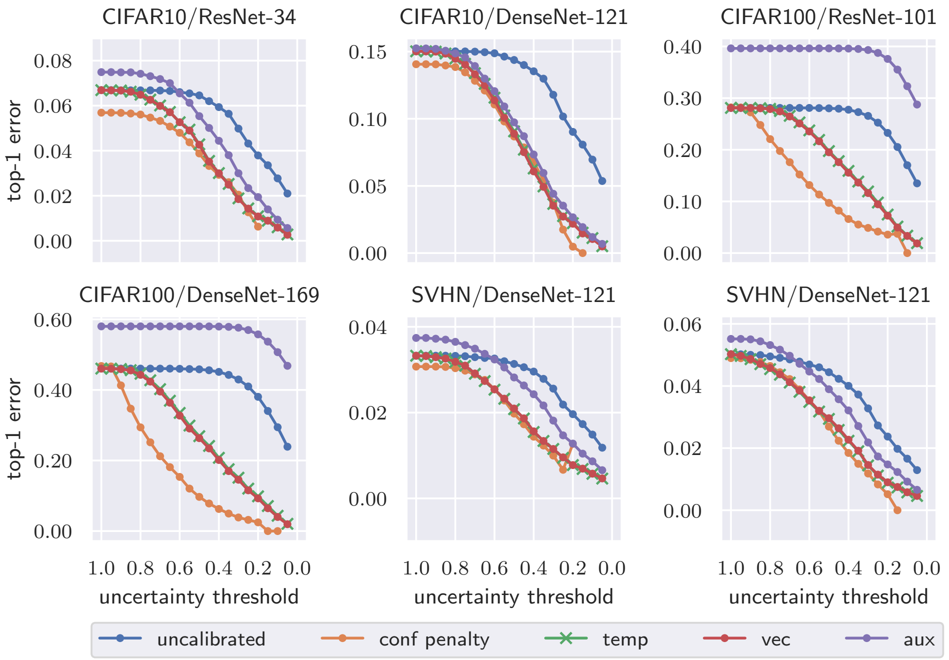


Figure 2.8: Reliability diagrams ($M = 15$ bins) on CIFAR-100 DenseNet-169. Top row: Uncalibrated frequentist confidence, and likelihood and uncertainty obtained by MC Gaussian dropout. The following rows show the results of the logit scaling methods. The dotted lines illustrates perfect calibration. Additional diagrams can be found in the supplemental material.

Rejection of unreliable predictions



CIFAR10/ResNet-34

CIFAR10/DenseNet-121

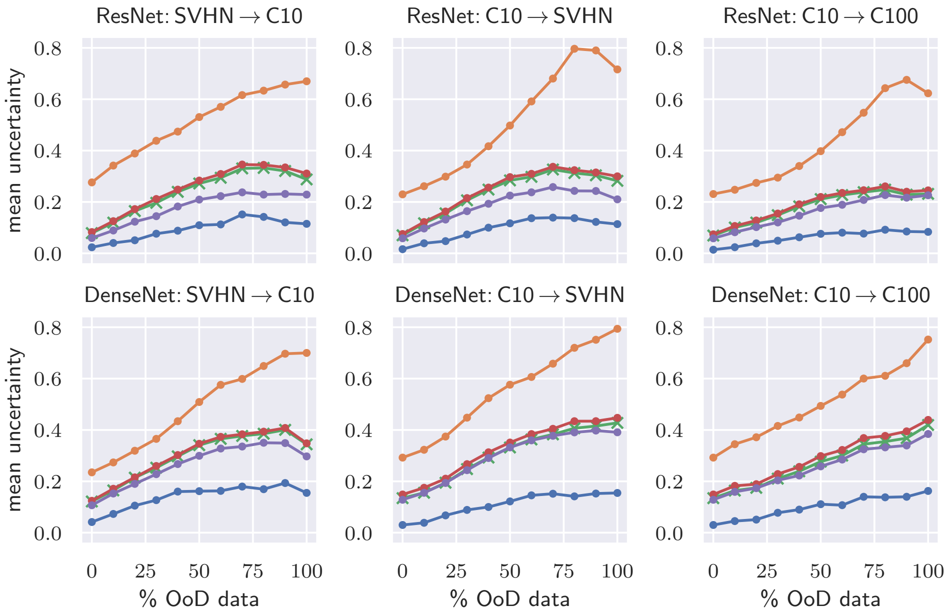
CIFAR100/ResNet-101

CIFAR100/DenseNet-169

SVHN/DenseNet-121

SVHN/DenseNet-121

Out-of-Distribution Detection



ResNet: SVHN → C10

ResNet: C10 → SVHN

ResNet: C10 → C100

DenseNet: SVHN → C10

DenseNet: C10 → SVHN

DenseNet: C10 → C100

Figure 2.9: (Left) The effect of the uncertainty threshold $\mathcal{H}_{\max}$ on the test set error for the rejection of uncertain predictions. (Right) Test set results of out-of-distribution detection.

Table 2.3: Classwise ECE and UCE test set results in % ($M = 15$ bins). 0 % means perfect calibration. RN and DN denote ResNet and DenseNet, respectively.

		uncalibrated		conf. penalty		temp. scaling		vector scaling		aux. scaling	
Data Set	Model	cECE	cUCE	cECE	cUCE	cECE	cUCE	cECE	cUCE	cECE	cUCE
CIFAR-10	RN34	4.46	4.03	8.29	19.8	**1.95**	3.68	2.09	3.73	2.10	**2.38**
CIFAR-10	DN121	10.1	9.52	8.49	18.5	3.05	5.72	3.15	6.09	**2.98**	**4.55**
CIFAR-100	RN101	20.5	23.2	14.6	19.4	10.8	11.5	**10.7**	**11.4**	32.9	35.3
CIFAR-100	DN169	32.4	37.1	15.6	20.6	12.9	13.9	**12.8**	**13.8**	48.9	52.6
SVHN	RN34	2.37	2.07	9.11	22.3	1.47	3.47	1.44	3.43	**1.34**	**1.85**
SVHN	DN121	2.91	2.47	7.53	19.7	2.06	5.08	1.96	4.88	**1.51**	**2.46**

calibration on the test set than the uncalibrated model. A possible solution to this is adding regularization (e.g., early stopping or weight decay) during optimization of $\boldsymbol{R}$. If the model is already well-calibrated (e.g., for SVHN in our experiments), temperature scaling and vector scaling can slightly worsen calibration. In this case, a larger calibration set is preferred or recalibration can be omitted at all. Confidence penalty only slightly reduces miscalibration for larger models on CIFAR-100. On all other configurations, it leads to worse calibration. As hypothesized in § 2.1.2, temperature scaling results in classwise calibrated uncertainty and is only marginally outperformed by the classwise logit scaling methods. The reliability diagrams in Fig. 2.7 give additional insight and show, that calibrated uncertainty corresponds well with the model error. It is worth noting that the likelihood in the Bayesian approach is generally better calibrated than the frequentist confidence.

Rejection of Uncertain Predictions Fig. 2.9 (left) shows the top-1 error as a function of decreasing $\mathcal{H}_{\max}$. For both uncalibrated and calibrated uncertainty, decreasing $\mathcal{H}_{\max}$ reduces the top-1 error. Again, we can observe the underestimation of uncalibrated uncertainty: $\mathcal{H}_{\max}$ has little effect at first and few uncertain predictions are rejected. Using calibrated uncertainty with temperature or vector scaling, the relationship is almost linear, allowing robust rejection of uncertain predictions. Except for aux scaling on CIFAR-100, logit scaling is capable of reducing the top-1 error below 1 %. Further, we observe that confidence penalty can lead to *over*-estimation of uncertainty.

Out-of-Distribution Detection Fig. 2.9 (right) shows the effect of calibrated uncertainty to OoD detection. All calibration approaches are able to improve the detection of OoD data. The benefit of calibration is most noticeable on ResNet (C10 → C100) and DenseNet (SVHN → C10, C10 → SVHN), where the mean uncertainty stays almost constant for OoD data $> 50\,\%$ and thus, robust OoD detection is only possible after calibration. As in Fig. 2.9 (left), we can observe overestimation of uncertainty for confidence penalty. In some cases (e. g. DenseNet SVHN → C10), this causes a more robust OoD detection. This is in contrast to the results presented by Lakshminarayanan et al. (2017), where MC dropout uncertainty was not able to capture OoD data sufficiently.

2.1.6 Conclusion

In this section, calibration of Bayesian uncertainty is discussed. We have proposed to measure uncertainty based on the normalized entropy. From this, we derived the uncertainty calibration error; a new metric that avoids several pathologies of existing calibration errors. The UCE does not only consider the class with the highest probability and is not minimized by a constant model predicting the marginal class distribution. In contrast to the Brier score and NLL, it allows comparison of models with different accuracy. It is not sensitive to a varying number of bins and provides a consistent ranking of models. However, we follow the suggestion of Ashukha et al. 2020 and state that comparison of calibration for different models should only be done at optimal softmax temperature. Regularization with UCE during training reduces miscalibration and does not penalize high accuracy and predictions with justified high confidence. UCE regularization with temperature scaling often performed best in our experiments in terms of calibration. The normalized entropy itself is a useful measure of uncertainty and allows for robust rejection of uncertain predictions and detection of OoD data.

Moreover, we derived logit scaling as entropy maximization technique to recalibrate the uncertainty from variational inference with deep models. Logit scaling calibrates uncertainty with high effectiveness. The experimental results show that better calibrated uncertainty allows more robust predictions and detection of out-of-distribution data; a key feature that is particularly important in safety-critical applications. Logit scaling is easy to implement and more effective than confidence penalty during training. Simple scaling methods are preferred over more complex methods, as they provide similar results and do not tend to overfit the calibration set. Temperature scaling improves uncertainty estimation without affecting the accuracy of the model. Vector and auxiliary scaling also improve calibration of uncertainty, but can have (positive or negative) influence on predictive accuracy. By using entropy, the classwise uncertainty calibrated by vector and auxiliary scaling is not substantially better than that calibrated by temperature scaling. Logit scaling calibrates not only the frequentist confidence but also the Bayesian uncertainty.

With this work, we hope to have provided a new useful metric for reliable evaluation of uncertainty estimation. The UCE is easy to implement and interpretable as it expresses the discrepancy of the uncertainty from the model error, which increases the chance of being accepted by deep learning practitioners.

Outlook

Throughout this work, we used a fixed dropout rate p for MC and Gaussian dropout. In (Gal, Hron, et al. 2017), the Concrete distribution was used as a continuous approximation to the discrete Bernoulli distribution in dropout, which allows optimizing p w.r.t. calibrated uncertainty. Using Gaussian dropout as described above, we can also recalibrate models by optimizing p w.r.t. NLL on the calibration set, which scales σ to reduce underestimation of uncertainty.

In Bayesian active learning we want to train a model with the minimal number of expert queries from a pool of unlabeled data. Calibrated uncertainty can further be useful to acquire the most uncertain samples from pool data to increase information efficiency (Gal, Islam, et al. 2017). Additionally, pseudo-labels can be generated from the least uncertain predictions in semi-supervised learning. Combined with consistency learning and deep clustering approaches, such as entropy maximization (Ji et al. 2019), this can leverage semi-supervised learning and lead to fully self-supervised learning, which we address in the next section.

2.2 BatchPL: An Efficient Sample Acquisition Scheme for Pseudo-Labeling in Self-Supervised Learning

Creating large labeled data sets for supervised learning is costly, especially in medical imaging where labeling can only be performed with expert domain knowledge. Self-supervised learning (SSL) has recently gained attraction and aims at training a deep model without any labels. In this work, we present an SSL framework that uses an uncertainty-aware pseudo-labeling approach, which is bootstrapped by mutual information maximization with consistency learning. We present *BatchPL*, a novel sample acquisition function for pseudo-labeling based on relative entropy between a sample prediction and its batch. The acquisition function selects highly informative samples with low uncertainty and leverages SSL with pseudo-labeling. Our framework outperforms recent SSL approaches and achieves an accuracy of 64.5 % on CIFAR-10 and 99.3 % on both MNIST and Medical MNIST without using any labels.

2.2.1 Introduction

Supervised learning requires large amounts of labeled data. Even more problematic, machine learning tasks in safety-critical areas, such as autonomous driving or medical imaging, need domain experts to label the data, which is costly and often not feasible. However, unlabeled data is usually easy to obtain, e.g., in clinical routine, making semi- and self-supervised learning approaches interesting in these domains. They aim at extracting useful information out of the unlabeled data and can be separated into *generative* or *discriminative* approaches. Generative approaches usually use an auto-encoding structure and try to reconstruct the input from a latent representation (Creswell et al. 2018; D. Kingma and Welling 2014). The representation implicitly provides a clustering of the data in the latent space. However, the clustering arises only implicitly and the required reconstruction adds computational burden. Discriminative approaches are closer related to supervised learning, but use labels that are directly derived from the input data. One method that has recently gained attraction is *consistency learning* (CL), which maximizes the agreement between two nearby data samples (Verma et al. 2019). Current self-supervised pre-training methods considerably outperform fully supervised training (T. Chen et al. 2020; He, Fan, et al. 2020); they use unlabeled data with CL to extract meaningful representations (known as pretext task) and subsequently train a linear classifier on the representations with limited labeled data (i.e., 1 % labels on ImageNet or CIFAR-10/100). A different approach to discriminative self-supervised learning is *pseudo-labeling* (PL), where labels are generated from predictions with high confidence and treated as ground truth in a supervised manner (Grandvalet and Bengio 2005; D.-H. Lee 2013). However, pseudo-labeling often performs worse as many predictions are incorrect due to highly overconfident models (Guo et al. 2017). Recently, Rizve et al. (2021) proposed uncertainty-aware pseudo-labeling as an equally effective alternative to CL, where Bayesian methods are used to obtain better uncertainty estimates and thus more accurate pseudo-labels. Gupta et al. (2020) proposed a similar approach using deep ensembles, but used

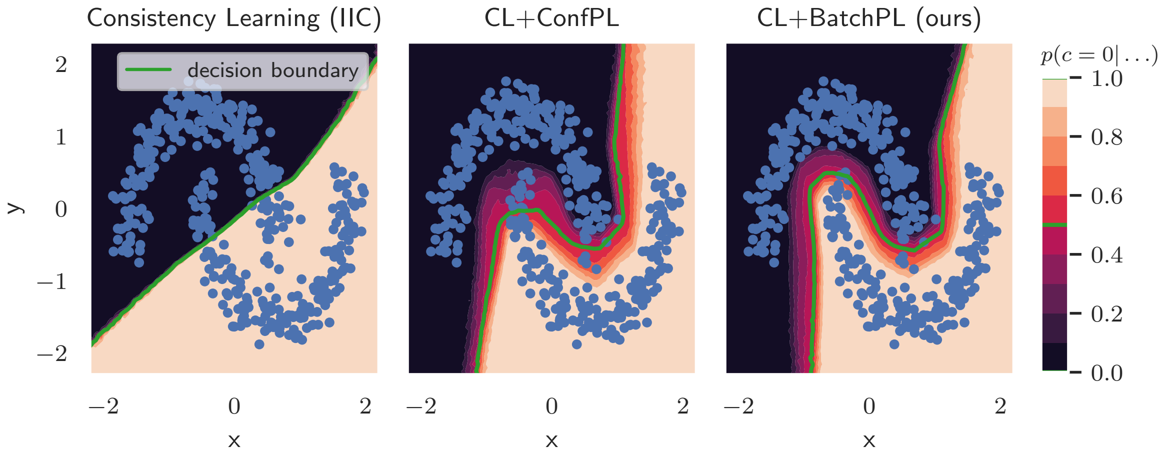


Figure 2.10: Unsupervised toy experiment on the “Two Moons” data set with class imbalance: (Left) Consistency learning with IIC (Ji et al. 2019) separates the data into two clusters violating the cluster assumption; the decision boundary traverses a high-density region with overconfidence. (Center) Pseudo-labeling based on high confidence helps to push the decision boundary into low-density regions but overfocuses on the upper moon with more frequent samples. (Right) BatchPL acquires pseudo-labels for highly informative samples and correctly clusters the data. A three layer fully-connected ReLU network with 32 neurons, MC dropout and 4,000 training iterations was used for all three methods.[1]

purely unsupervised clustering methods to create a small labeled data set for bootstrapping pseudo-label generation.

In this work, we combine consistency learning and self-supervised learning with pseudo-labels for classification obtained from Bayesian uncertainty, which to the best of our knowledge has not yet been done. We propose *BatchPL*, a sample acquisition scheme for pseudo-labeling that selects highly informative samples (see Fig. 2.10). Our approach outperforms recent self-supervised approaches in multi-class classification without using any labeled data and can be used with any deep learning architecture. We do not focus on pretext tasks, but fully train a deep classifier end-to-end without supervision. Labeled data is only used at test time to compute the final accuracy metrics.

2.2.2 Related Work

Consistency Learning The current state-of-the-art in self-supervised learning is based on the consistency learning framework, which dates back to D. Zhou et al. (2003) and is based on the *cluster assumption*: Two similar data samples are likely to have the same label. A simple way to create two samples with the same label is to use different augmentations of the same data sample (T. Chen et al. 2020). A consistency loss then aims at maximizing the consistency of the paired predictions on the augmentations (Ji et al. 2019). Two augmentations of the same image are considered a positive pair. Consistency learning can further be extended by using two different images as negative pair and minimize the agreement, which is referred to as contrastive learning (T. Chen et al. 2020). However, in this section, we focus on positive pairs only and do not perform negative learning. A key component in this framework is the consistency loss function. It must be chosen in such way that degenerate solutions are avoided; i.e., all samples are assigned to the same class. Consistency learning has been used in unsupervised pre-training in multi-class classification (T. Chen et al. 2020; He, Fan, et al. 2020), image-to-image translation (Zhu et al. 2017), person re-identification (Wu et al. 2019) and unsupervised image clustering (Bachman et al. 2019; Ji et al. 2019).

Pseudo-Labeling A straightforward method to perform self-supervised learning is to use a supervised objective function (i.e., cross-entropy) with self-generated pseudo-labels. This approach usually needs a minimal amount of labeled data to bootstrap itself (D.-H. Lee 2013). The quality of the pseudo-labels greatly depends on the sample acquisition scheme. Simply using the predictions that are above a certain confidence threshold usually results in pseudo-labels with low accuracy due to poor calibration and high overconfidence of deep models (Guo et al. 2017; Rizve et al. 2021). Rizve et al. (2021) argue that PL performs on par to CL when using temperature scaled deep Bayesian models with better calibrated uncertainty. Pseudo-labeling has been used in semi-supervised image classification (D.-H. Lee 2013; Rizve et al. 2021) and unsupervised image clustering (Caron et al. 2018; Gupta et al. 2020). An implicit combination of PL and CL was recently proposed by Sohn et al. (2020) as *FixMatch*: Weakly augmented images are used to create pseudo-labels, which are subsequently used to minimize a supervised training objective on heavily augmented versions of the images.

2.2.3 Methods

In this section, we propose to use consistency learning with mutual information maximization to bootstrap pseudo-label generation with FixMatch using uncertainty from deep Bayesian models. We introduce a novel sample acquisition function based on the relative entropy between a data point and its batch that selects highly informative samples.

Bootstrapping with Mutual Information Maximization

Invariant information clustering (IIC) is a recent method that enables unsupervised training of deep image classifiers by maximizing the mutual information between two differently augmented data samples (Ji et al. 2019). Maximizing the mutual information effectively minimizes the conditional entropy of the paired predictions, which favors a deterministic one-hot output. This encourages the model to become overconfident and leads to violation of the cluster assumption, creating decision boundaries that traverse through regions with high

---

[1] The code for this example can be found at gist.github.com/mlaves.

data density (see Fig. 2.10 (left) and (Oliver et al. 2018)). However, it suffices to get at least some correct labels per class, which can be used to bootstrap our pseudo-labeling framework. We will briefly review IIC and subsequently describe our bootstrapping procedure.

Let $t \sim \mathcal{T}$ be a random sample from a set of augmentation operations (e.g., random cropping, flipping, color jitter, and rotations, all with different magnitudes) and let $\{\boldsymbol{x}, \boldsymbol{x}'\}$ be a paired data sample of input image $\boldsymbol{x} \in \mathcal{X}$ and its augmented version $\boldsymbol{x}' = t(\boldsymbol{x})$. The goal of IIC is to maximize the mutual information $\mathcal{I}[y, y']$ w.r.t. the model parameters

$$\arg\max_{\boldsymbol{\theta}} \; \mathcal{I}\left[\boldsymbol{f_\theta}(\boldsymbol{x}), \boldsymbol{f_\theta}(\boldsymbol{x}')\right] \; . \tag{2.45}$$

Following Ji et al. (2019), $\mathcal{I}[y, y']$ is computed by

$$\mathcal{I}[\boldsymbol{P}] = \sum_{c=1}^{C} \sum_{c'=1}^{C} \boldsymbol{P}_{cc'} \log \frac{\boldsymbol{P}_{cc'}}{\boldsymbol{P}_c \boldsymbol{P}_{c'}} \; , \tag{2.46}$$

with the joint probability distribution given by the $C \times C$ matrix

$$\boldsymbol{P} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{f_\theta}(\boldsymbol{x}_i) \boldsymbol{f_\theta}(\boldsymbol{x}'_i)^\mathsf{T} \; . \tag{2.47}$$

with entries $\boldsymbol{P}_{cc'} = P(y = c, y' = c')$. Marginalization in Eq. (2.47) is performed over the mini-batch with size n. The joint probability distribution is symmetrized using $(\boldsymbol{P} + \boldsymbol{P}^\mathsf{T})/2$. The marginals $\boldsymbol{P}_c = \boldsymbol{P}(y_i = c)$ and $\boldsymbol{P}_{c'} = \boldsymbol{P}(y = c')$ are computed by summing over the rows or columns of $\boldsymbol{P}$. We use sole IIC training in the very first iterations as a warm-up phase to subsequently be able to generate pseudo-labels as described in § 2.2.3.

BatchPL: Pseudo-Label Selection Based on Relative Entropy

The violation of the cluster assumption from IIC training can be mitigated by employing consistency learning with pseudo-labels (cf. Fig. 2.10). The question arises how to acquire well-predicted samples to compute pseudo-labels from after bootstrapping with IIC. A naive way would be to simply select predictions that are below a predefined uncertainty threshold. However, in the early stages of IIC training, only a few prediction classes are populated, while some classes are ignored at all (e.g., 1's and 7's from MNIST are first grouped into the same cluster, cf. Fig. 3 from Ji et al. (2019)). Pseudo-labeling reinforces the attention to the classes that are already predicted with high confidence, which leads to ignorance and underpopulation of the other classes and the training can get stuck. We name this failure phenomenon *overfocusing*. Obtaining pseudo-labels for samples from classes of which the model is already very confident about would add little to no new information to the pseudo-labeled set. It is more effective to select samples that are predicted with low uncertainty, but are underrepresented in the current prediction set (i.e., current batch).

To address this issue, we present *BatchPL*, a sample acquisition function for efficient

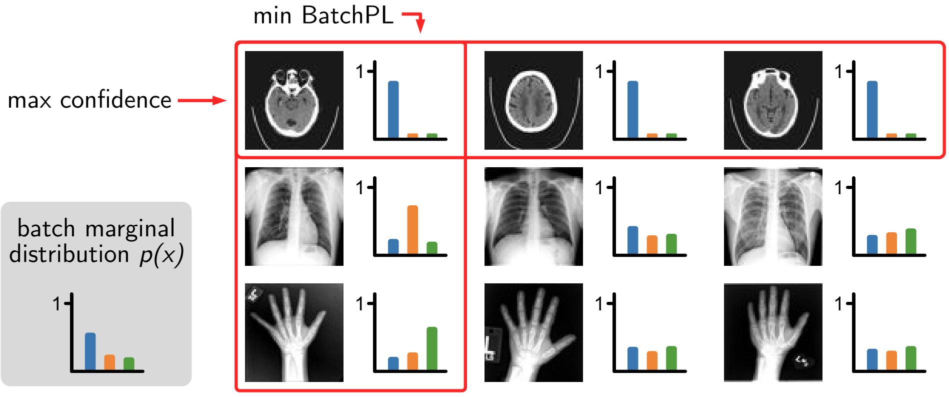


Figure 2.11: Overview of BatchPL sample acquisition on an example batch. Using confidence (i.e., highest softmax probability) for sample acquisition would results in overfocusing on samples with high similarity. BatchPL increases pseudo-label efficiency by penalizing classes that are predicted with overconfidence. Figure inspired by Kirsch et al. (2019).

selection of pseudo-labels. BatchPL balances the entropy of a single prediction and the relative entropy (or Kullback-Leibler divergence) between the single prediction and the marginalized predictions of its data batch. It is defined as

$$\mathsf{BatchPL}\left[y^* \,\|\, y\right] = \mathcal{H}[p(y^* \,|\, \boldsymbol{x}^*)] - \mathrm{KL}\left[p(y) \,\|\, p(y^* \,|\, \boldsymbol{x}^*)\right] \ , \tag{2.48}$$

with current prediction $p(y^* \,|\, \boldsymbol{x}^*)$ and prediction $p(y)$ from marginalization over the current batch (or data set in general). Predictions with low BatchPL value are selected for pseudo-labeling. The use of this KL divergence as a measure of dissimilarity between two distributions dampens the selection of samples that are already frequently predicted with high confidence. The direction of the KL divergence is of no particular importance here, as we do not optimize Eq. (2.48) directly. Nevertheless, we opt for KL divergence from $p(y \,|\, \boldsymbol{x}^*)$ to $p(y)$ (see next section). Note that we omitted the conditioning on the network parameters $\boldsymbol{\theta}$ here and will cover that later when moving towards a Bayesian treatment.

Why BatchPL Works A low BatchPL value indicates that a sample is predicted with low uncertainty and thus is suited for pseudo-label generation, but is from an underrepresented class within its batch predictions (see Fig. 2.11). BatchPL has several favorable properties: It trades off the entropy $\mathcal{H}[y^*]$ of the current prediction and the relative entropy $\mathrm{KL}\left[y \,\|\, y^*\right]$ between the marginal distribution $p(y)$ of the batch and the current prediction. The entropy describes the level of uncertainty about y^* and the relative entropy is a measure of dissimilarity between the marginal distribution and the distribution of current prediction. The first term is low when the model is confident about y^* and the second term is high when the batch marginal distribution considerably differs from the distribution of the current prediction. This results in a sample selection that is more spread across all classes. More specifically, the KL divergence expands to $\mathcal{H}[y, y^*] - \mathcal{H}[y]$, comprising of the cross entropy between y and y^*, and the negative entropy of y. When $p(y)$ approaches a uniform distribution (i.e., all classes are well-represented by the current model), the penalizing effect of the KL divergence in BatchPL is mitigated and it behaves more like sample selection based on entropy (i.e., uncertainty) .

Using BatchPL to select samples for pseudo-labeling avoids that the model overfocuses on what is already well-known. Consider the following example: A batch of four binary samples with ground truth class labels $\{1, 1, 2, 2\}$ was predicted with the softmax outputs $\{(\frac{9}{10}, \frac{1}{10}), (\frac{9}{10}, \frac{1}{10}), (\frac{6}{10}, \frac{4}{10}), (\frac{2}{10}, \frac{8}{10})\}$. An uncertainty or confidence based acquisition scheme would select the first two predictions in this batch for pseudo-labeling and manifest the model's focus on class 1. BatchPL on the other hand results to $\{0.1, 0.1, 0.7, 0.02\}$. This favors the selection of the last sample, as pseudo-labeling this is more effective as the first two samples.

Computing BatchPL

Let $\boldsymbol{x}^* \in \mathcal{B}$ be a data point from the current mini-batch $\mathcal{B} \subseteq \mathcal{D}$ with size n of a data set $\mathcal{D}$ of unlabeled images $\mathcal{X}$ with unknown ground truth class $y \in \mathcal{Y}$. Let $\boldsymbol{f_\theta}\colon \mathcal{X} \to \mathcal{Y}$ be a deep variational Bayesian classification model with variational distribution $q(\boldsymbol{\theta})$ using MC dropout (Gal and Ghahramani 2016b) and softmax output obtained by Monte Carlo integration

$$p(y^* \,|\, \boldsymbol{x}^*, \mathcal{D}) = \mathbb{E}_{\boldsymbol{\theta} \sim p(\boldsymbol{\theta} \,|\, \mathcal{D})}[p(y^* \,|\, \boldsymbol{x}^*, \boldsymbol{\theta})] \approx \frac{1}{T} \sum_{t=1}^{T} \boldsymbol{f}_{\tilde{\boldsymbol{\theta}}_t}(\boldsymbol{x}^*) \tag{2.49}$$

with samples $\tilde{\boldsymbol{\theta}}_t \sim q(\boldsymbol{\theta})$ and T stochastic forward passes. In the following, we omit the conditioning on $\mathcal{D}$ for brevity. The goal of BatchPL is to compute the entropy $\mathcal{H}[y^*]$ of the distribution $p(y^* \,|\, \boldsymbol{x}^*)$ for a current sample $\boldsymbol{x}^*$ and the KL divergence between the batch marginal distribution

$$p(y) = \mathbb{E}_{\boldsymbol{x}}\, \mathbb{E}_{\boldsymbol{\theta}}[p(y \,|\, \boldsymbol{x}, \boldsymbol{\theta})] \approx \frac{1}{nT} \sum_{j=1}^{n} \sum_{t=1}^{T} \boldsymbol{f}_{\tilde{\boldsymbol{\theta}}_t}(\boldsymbol{x}_j) \tag{2.50}$$

and the distribution for the current sample. Finally, BatchPL is computed by

$$-\sum_{c=1}^{C} p(y^* = c \,|\, \boldsymbol{x}^*) \log p(y^* = c \,|\, \boldsymbol{x}^*) - \sum_{c=1}^{C} p(y = c) \log \frac{p(y^* = c \,|\, \boldsymbol{x}^*)}{p(y = c)}\,. \tag{2.51}$$

We compute $\mathsf{BatchPL}[y^* \,\|\, y]$ for all predictions y^* in the current batch and select the samples for pseudo-labeling where BatchPL is below a predefined threshold β.

Bounds on BatchPL If $p(y^* \mid \boldsymbol{x}^*) = p(y) = \mathcal{U}(1, C)$ with uniform distribution $\mathcal{U}(1, C)$ over C classes, BatchPL reaches its upper bound

$$\sup \ \mathsf{BatchPL}[y^* \,\|\, y] = \log C \ . \tag{2.52}$$

However, there is no lower bound on BatchPL, as there is no general upper bound on the KL divergence:

$$\inf \ \mathsf{BatchPL}[y^* \,\|\, y] = -\infty \ . \tag{2.53}$$

This can make empirical selection of a threshold value β difficult. However, we received good results using pseudo-labels from sample predictions with $\mathsf{BatchPL}[y^* \,\|\, y] < 0$.

2.2.4 Experiments

We evaluate pseudo-labeling with BatchPL in the following self-supervised scenario. A convolutional network is trained on a classification task without using any labels during optimization. We use an imbalanced version of the Two Moons (upper moon:lower moon = 500:400) as toy data set, Medical MNIST as a medical data set, and MNIST and CIFAR-10 in order to compare our results to results from related works. Sole IIC training is used in the first n epochs to bootstrap pseudo-label generation following the training procedure described by Ji et al. (2019). In all experiments, we use m classification heads in parallel with additional overclustering. Overclustering produces an auxiliary clustering with a greater number of clusters than actually present and has empirically been proven to improve overall accuracy (Ji et al. 2019). It is implemented using a linear classification layer with more units $k \cdot C$ than the number of classes C. After some sole iterations of IIC as warm-up, we perform the following consecutive steps in each epoch:

1. IIC training,
2. pseudo-label generation with BatchPL,
3. supervised training with pseudo-labels.

For consistency learning in the IIC step, we maximize Eq. (2.45) using a weakly augmented and a strongly augmented version of the input image. We train the overclustering heads and the actual classification heads in alternating epochs. During test time, the Hungarian algorithm is used to find a mapping between the predicted classes and the ground truth to estimate classification accuracy. We use a BatchPL threshold of $\beta = 0$ and the Adam optimizer (D. P. Kingma and Ba 2014) with a fixed learning rate of 10^{-4} in all subsequent experiments.

MNIST A VGG-like network is used (Simonyan and Zisserman 2014) with $n = 10$ IIC warm-up epochs, a batch size of 750, overclustering with $k = 5$, and $m = 5$ classification heads. Weak augmentation consists of random random cropping with squared size of 24 pixels

Table 2.4: Accuracy from best out of 3 runs with different random initialization. Self-supervised training with BatchPL outperforms IIC on Two Moons, Medical MNIST, and CIFAR-10. The results for IIC on MNIST and CIFAR-10 are reported by Ji et al. (2019). Results from fully supervised training are given as comparison.

Data set	Random Network	IIC	BatchPL (ours)	Fully Supervised
Two Moons	51.3 %	86.4 %	**93.7** %	99.8 %
MNIST	26.1 %	**99.3** %	**99.3** %	99.7 %
MedicalMNIST	38.2 %	98.1 %	**99.3** %	99.9 %
CIFAR-10	13.1 %	61.7 %	**64.5** %	93.5 %

and strong augmentation consists of random cropping with random sizes of $\{16, 20, 24\}$ pixels, random rotation in the range $[-25, 25]$ degree, and random color jitter.

Medical MNIST This data set aims at being a drop-in replacement for MNIST with medical imaging modalities (Maranhão 2020). However, in contrast to MNIST it comprises of 58,954 single channel images with size 64×64 from the 6 classes AbdomenCT, BreastMRI, CXR, ChestCT, Hand, HeadCT. We normalize the images and scale them to 28×28 pixels. The training procedure is similar to that of MNIST, but with added random horizontal flipping and without random rotations. Additionally, all images are Sobel filtered in a preprocessing step, which helps to suppress focusing on simple features such as background patterns and color and emphasizes shape.

CIFAR-10 ResNet-34 is used (He, X. Zhang, et al. 2016) with $n = 50$ IIC warm-up epochs, a batch size of 660, overclustering with $k = 7$, and $m = 5$ classification heads. We use the same augmentation strategy as for Medical MNIST.

Results

The experimental results are summarized in Tab. 2.4. IIC violates the cluster assumption on the imbalanced Two Moons toy data set and is not able to correctly separate the data (see Fig. 2.10). Pseudo-labeling with BatchPL pushes the decision boundary into low-density regions and does not overfocus on the majority class, as confidence based pseudo-labeling does (cf. Fig. 2.10 center). On MNIST, IIC and BatchPL perform on par and almost reach the accuracy of fully supervised training. BatchPL considerably outperforms IIC on Medical MNIST and CIFAR-10. Especially on the former, the benefit of BatchPL becomes apparent. We observe that IIC mixes up two very similar classes in the beginning of the training. The IIC training objective makes the model becoming overconfident about these wrong predictions and getting stuck in this local minimum. This reliably happens on Medical MNIST, but also sometimes on MNIST (e.g., mixing up 1's and 7's or 5's and 6's), depending on the random initialization. BatchPL addresses this by explicitly avoiding to overfocus on the overconfident predictions.

On the more challenging CIFAR-10 data set, we observe that BatchPL mixes up classes that are closely related, such as ‘deer’ and ‘horse’ or ‘automobile’ and ‘truck’, as it relies on self-extracted visual features. However, to the best of our knowledge, BatchPL provides state-of-the-art performance in fully unsupervised CIFAR-10 and replaces IIC as the former state-of-the art (as of writing this thesis).

2.2.5 Conclusion

In this section, we presented an efficient framework for uncertainty-aware pseudo-labeling in self-supervised learning with application to medical images. The core of the framework is BatchPL, a novel sample acquisition function for pseudo-labeling that is based on the relative entropy between a sample prediction and the predictions of its batch. Our experiments have shown that BatchPL is advantageous in cases where other self-supervised methods fail. When dealing with class imbalances, BatchPL does not overfocus on the majority class. Additionally, our framework achieved state-of-the-art performance on common multi-class classification tasks.

BatchPL can easily be extended to negative samples. If a class can be clearly rejected for a given sample, BatchPL can be redefined to efficiently select negative samples for the use with negative cross-entropy employing negative pseudo-labels. Moreover, BatchPL can be used in semi-supervised training with some labeled data. In future experiments, we also plan to use BatchPL on segmentation tasks.

2.3 Chapter Conclusion

In this chapter, we have addressed the calibration of predictive uncertainty in the context of classification and computer-aided diagnosis. The first part (§ 2.1) presented the uncertainty calibration error, a new method for measuring miscalibration and to regularize neural networks during training for improved calibration. In our experiments, it outperformed other commonly used regularization methods.

Subsequently, the second part of this chapter (§ 2.2) introduced a novel framework for unsupervised training of multi-class classification models using uncertainty-aware self-labeling. The presented approach achieved state-of-the-art performance on both medical and non-medical classfication data sets. These findings confirm hypothesis 1 and 2 (cf. § 1.5).

3 Regression in Medical Imaging

In this chapter, we apply estimation of predictive uncertainty by variational Bayesian inference with Monte Carlo dropout to regression tasks and show why predictive uncertainty is systematically underestimated. We suggest using σ *scaling* with a single scalar value; a simple, yet effective calibration method for both aleatoric and epistemic uncertainty. The performance of our approach is evaluated on a variety of common medical regression data sets using different state-of-the-art convolutional network architectures. In our experiments, σ scaling is able to reliably recalibrate predictive uncertainty. It is easy to implement and maintains the accuracy. Well-calibrated uncertainty in regression allows robust rejection of unreliable predictions or detection of out-of-distribution samples.

The work in this chapter was partly published at the peer-reviewed “Medical Imaging with Deep Learning” (MIDL) 2020 conference and presented as long oral (best 12 % of submitted papers) (Laves, Ihler, Fast, et al. 2020). An extended version of our work was submitted to the peer-reviewed Journal of Machine Learning for Biomedical Imaging (MELBA) and published in April 2021 (Laves, Ihler, Fast, et al. 2021). Besides the co-authors of the published work, we thank Vincent Modes and Mark Wielitzka for their insightful comments. The source code for all experiments in this chapter is publicly available at: github.com/mlaves/well-calibrated-regression-uncertainty

3.1 Introduction

Predictive uncertainty should be considered in any medical imaging task that is approached with deep learning. Well-calibrated uncertainty is of great importance for decision-making and is anticipated to increase patient safety. It allows to robustly reject unreliable predictions or out-of-distribution samples. In this chapter, we address the problem of miscalibration of regression uncertainty with application to medical image analysis.

For the task of regression, we aim to estimate a continuous target value $\boldsymbol{y} \in \mathbb{R}^d$ given an input image $\boldsymbol{x}$. Regression in medical imaging with deep learning has been applied to forensic age estimation from hand CT/MRI (Halabi et al. 2019; Štern et al. 2016), natural landmark localization (Payer et al. 2019), cell detection in histology (Xie et al. 2018), or instrument pose estimation (Gessert et al. 2018). By predicting the coordinates of object boundaries, segmentation can also be performed as a regression task. This has been done for segmentation of pulmonary nodules in CT (Messay et al. 2015), kidneys in ultrasound (Yin et al. 2020), or left ventricles in MRI (L. K. Tan et al. 2017). In registration of medical images, a continuous displacement field is predicted for each coordinate of $\boldsymbol{x}$, which has also recently been addressed by CNNs for regression (Dalca et al. 2019).

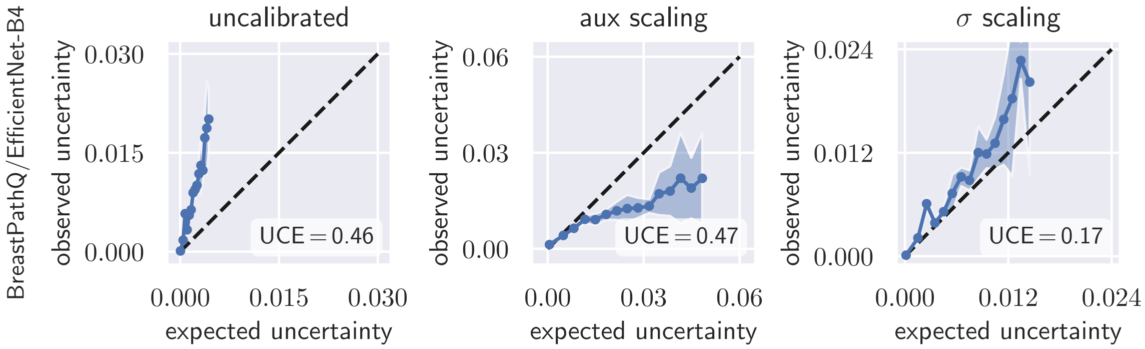


Figure 3.1: Calibration plots (expected uncertainty vs. observed uncertainty) and uncertainty calibration error (UCE) for EfficientNet-B4 on BreastPathQ test set. Dashed lines denote perfect calibration. The discrepancy to the identity function reveals miscalibration. Uncalibrated uncertainty is underestimated and does not correspond well with the model error (left). Uncertainty can be calibrated most effectively with σ scaling (right). Solid lines show the mean and shaded areas show standard deviation from 5 repeated runs.

In medical imaging, it is crucial to consider the predictive uncertainty of deep learning models. Bayesian neural networks (BNN) and their approximation provide mathematical tools for reasoning the uncertainty (Bishop 2006; D. Kingma and Welling 2014). In general, predictive uncertainty can be split into two types: aleatoric and epistemic uncertainty (Kendall and Gal 2017; Tanno, Worrall, et al. 2017). This distinction was first made in the context of risk management (Hora 1996). Aleatoric uncertainty arises from the data directly; e.g. sensor noise or motion artifacts. In regression, it is derived from the conditional log-likelihood under the maximum likelihood framework and can be captured by a deep model directly (see § 3.2.1). Epistemic uncertainty is caused by uncertainty in the model parameters due to a limited amount of training data (Bishop 2006). A well-accepted approach to quantify epistemic uncertainty is variational inference with Monte Carlo (MC) dropout, where dropout is used at test time to sample from the approximate posterior (Gal and Ghahramani 2016b).

Uncertainty quantification in regression problems in medical imaging has been addressed by prior work. Medical image enhancement with image quality transfer (IQT) has been extended to a Bayesian approach to obtain pixel-wise uncertainty (Tanno, Ghosh, et al. 2016). Additionally, CNN-based IQT was used to estimate both aleatoric and epistemic uncertainty in MRI super-resolution (Tanno, Worrall, et al. 2017). Dalca et al. (2019) estimated uncertainty for a deformation field in medical image registration using a probabilistic CNN. Registration uncertainty has also been addressed outside the deep learning community (Luo et al. 2019). Schlemper et al. (2018) used sub-network ensembles to obtain uncertainty estimates in cardiac MRI reconstruction. Aleatoric and epistemic uncertainty was also used in multitask learning for MRI-based radiotherapy planning (Bragman et al. 2018).

Uncertainty obtained by deep BNNs tends to be miscalibrated, i.e. it does not correlate well with the model error (Laves, Ihler, Kortmann, et al. 2019). Fig. 3.1 shows calibration plots (observed uncertainty vs. expected uncertainty) for uncalibrated and calibrated uncertainty. The predicted uncertainty (taking into account both epistemic and aleatoric uncertainty) is underestimated and does not allow robust detection of uncertain predictions at test time.

Calibration of uncertainty in regression has been addressed in prior work outside medical imaging. In (Kuleshov et al. 2018), inaccurate uncertainties from Bayesian models for regression are recalibrated using a technique inspired by Platt scaling. Given a pre-trained, miscalibrated model $\boldsymbol{H}$, an auxiliary model $\boldsymbol{R} : [0,1]^d \rightarrow [0,1]^d$ is trained, that yields a calibrated regressor $\boldsymbol{R} \circ \boldsymbol{H}$. In (Phan et al. 2018), this method was applied to bounding box regression. However, an auxiliary model with enough capacity will always be able to recalibrate, even if the predicted uncertainty is completely uncorrelated with the real uncertainty. Furthermore, Kuleshov et al. (2018) state that calibration via $\boldsymbol{R}$ is possible if enough independent and identically distributed (i.i.d.) data is available. In medical imaging, large data sets are usually hard to obtain, which can cause $\boldsymbol{R}$ to overfit the calibration set. This downside was addressed in (Levi et al. 2019), which is most related to our work. They proposed to scale the standard deviation of a Gaussian model to recalibrate aleatoric uncertainty. In contrast to our work, they do not take into account epistemic uncertainty, which is an important source of uncertainty, especially when dealing with small data sets in medical imaging.

This chapter extends a preliminary version of this work presented at the Medical Imaging with Deep Learning (MIDL) 2020 conference (Laves, Ihler, Fast, et al. 2020). We continue this work by providing a new derivation of our definition of perfect calibrtaion, new experimental results, analysis and discussion. Additionally, prediction intervals are computed to further assess the quality of the estimated uncertainty. We find that prediction intervals are estimated too narrow and that recalibration can mitigate this problem.

To the best of our knowledge, calibration of predictive uncertainty for regression tasks in medical imaging has not been addressed. Our main contributions are:

(1) We suggest to use σ scaling in a separate calibration phase to tackle underestimation of aleatoric and epistemic uncertainty (§ 3.2.5),

(2) we propose to use the uncertainty calibration error and prediction intervals to assess the quality of the estimated uncertainty (§ 3.2.7), and

(3) we perform extensive experiments on four different data sets to show the effectiveness of the proposed method (§ 3.3).

3.2 Methods

In this section, we discuss estimation of aleatoric and epistemic uncertainty for regression and show why uncertainty is systematically miscalibrated. We propose to use σ scaling to jointly calibrate aleatoric and epistemic uncertainty.

3.2.1 Conditional Log-Likelihood for Regression

We revisit regression under the maximum posterior (MAP) framework to derive direct estimation of heteroscedastic aleatoric uncertainty. That is, the aleatoric uncertainty varies with the input and is not assumed to be constant. The goal of our regression model is to predict a target value $\boldsymbol{y}$ given some new input $\boldsymbol{x}$ and a training set $\mathcal{D}$ of m inputs $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m\}$ and their corresponding (observed) target values $\boldsymbol{Y} = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m\}$. We assume that $\boldsymbol{y}$ has a Gaussian distribution $\mathcal{N}\left(\boldsymbol{y}; \hat{\boldsymbol{y}}(\boldsymbol{x}), \hat{\sigma}^2(\boldsymbol{x})\right)$ with mean equal to $\hat{\boldsymbol{y}}(\boldsymbol{x})$ and variance $\hat{\sigma}^2(\boldsymbol{x})$. A neural network with parameters $\boldsymbol{\theta}$

$$\boldsymbol{f}_{\boldsymbol{\theta}}\left(\boldsymbol{x}\right) = \left[\hat{\boldsymbol{y}}(\boldsymbol{x}), \hat{\sigma}^2(\boldsymbol{x})\right], \; \hat{\boldsymbol{y}} \in \mathbb{R}^d, \; \hat{\sigma}^2 \in \mathbb{R}, \hat{\sigma}^2 \geq 0 \tag{3.1}$$

outputs these values for a given input (Nix and Weigend 1994). We use a Gaussian to model the likelihood and define

$$p\left(\boldsymbol{y}|\boldsymbol{x}\right) = \mathcal{N}\left(\boldsymbol{y}; \hat{\boldsymbol{y}}(\boldsymbol{x}), \hat{\sigma}^2(\boldsymbol{x})\right) \; . \tag{3.2}$$

By assuming a Gaussian prior over the parameters $\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\theta}; \boldsymbol{0}, \lambda^{-1}\boldsymbol{I})$, MAP estimation becomes maximum-likelihood estimation with added weight decay (Bishop 2006). With m i.i.d. random samples, the conditional log-likelihood $\log p(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{\theta})$ is given by

$$\log p(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{\theta}) = \sum_{i=1}^{m} \log p\left(\boldsymbol{y}^{(i)} | \boldsymbol{x}^{(i)}; \hat{\boldsymbol{y}}_{\boldsymbol{\theta}}^{(i)}, \left(\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}\right)^2\right) \tag{3.3}$$

$$\sum_{i=1}^{m} \log\left(\frac{1}{\sqrt{2\pi}\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}} \exp\left\{-\frac{\left\|\boldsymbol{y}^{(i)} - \hat{\boldsymbol{y}}_{\boldsymbol{\theta}}^{(i)}\right\|^2}{2\left(\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}\right)^2}\right\}\right) \tag{3.4}$$

$$= -\frac{m}{2}\log\left(2\pi\right) - \sum_{i=1}^{m} \log\left(\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}\right) + \frac{1}{2\left(\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}\right)^2}\left\|\boldsymbol{y}^{(i)} - \hat{\boldsymbol{y}}_{\boldsymbol{\theta}}^{(i)}\right\|^2 . \tag{3.5}$$

The dependency on $\boldsymbol{x}$ has been omitted to simplify the notation. Maximizing the log-likelihood in Eq. (3.5) w.r.t. $\boldsymbol{\theta}$ is equivalent to minimizing the negative log-likelihood (NLL), which leads to the following optimization criterion (with weight decay)

$$\mathcal{L}_{\mathrm{G}}(\boldsymbol{\theta}) = \sum_{i=1}^{m} \left(\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}\right)^{-2}\left\|\boldsymbol{y}^{(i)} - \hat{\boldsymbol{y}}_{\boldsymbol{\theta}}^{(i)}\right\|^2 + \log\left((\hat{\sigma}_{\boldsymbol{\theta}}^{(i)})^2\right) . \tag{3.6}$$

Here, $\hat{\boldsymbol{y}}_{\boldsymbol{\theta}}$ and $\hat{\sigma}_{\boldsymbol{\theta}}$ are estimated jointly by finding $\boldsymbol{\theta}$ that minimizes Eq. (3.6). This can be achieved using gradient descent in a standard training procedure. In this case, $\hat{\sigma}_{\boldsymbol{\theta}}$ captures the uncertainty that is inherent in the data (aleatoric uncertainty). To avoid numerical instability due to potential division by zero, we directly estimate $\log \hat{\sigma}^2(\boldsymbol{x})$ and implement Eq. (3.6) in similar practice to Kendall and Gal (2017).

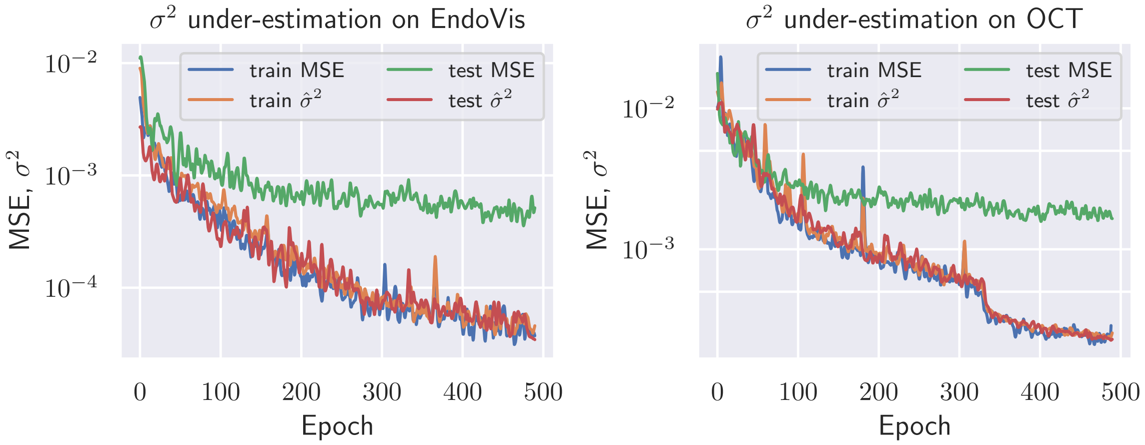


Figure 3.2: Biased estimation of aleatoric uncertainty σ^2. The deep model overfits estimation of $\boldsymbol{y}$ on the training set. On unseen test data, the MSE of predictive mean is higher and σ^2 is underestimated. Early stopping (e.g. at epoch 50) would result in an unbiased estimator, but this would not be optimal in terms of test MSE.

3.2.2 Biased estimation of σ

Ignoring their dependence through $\boldsymbol{\theta}$, the solution to Eq. (3.6) decouples estimation of $\hat{\boldsymbol{y}}$ and $\hat{\sigma}$. In case of a Gaussian likelihood, minimizing Eq. (3.6) w.r.t. $\hat{\boldsymbol{y}}^{(i)}$ yields

$$\hat{\boldsymbol{y}}^{(i)} = \underset{\hat{\boldsymbol{y}}^{(i)}}{\arg\min}\ \mathcal{L}_{\mathrm{G}} = \boldsymbol{y}^{(i)}\ \forall\, i\,. \tag{3.7}$$

Minimizing (3.6) w.r.t. $(\hat{\sigma}^{(i)})^2$ yields

$$\left(\hat{\sigma}^{(i)}\right)^2 = \underset{(\hat{\sigma}^{(i)})^2}{\arg\min}\ \mathcal{L}_{\mathrm{G}} = \|\boldsymbol{y}^{(i)} - \hat{\boldsymbol{y}}^{(i)}\|^2\ \forall\, i\,. \tag{3.8}$$

That is, estimation of σ^2 should perfectly reflect the squared error. However, in Eq. (3.8) σ^2 is estimated relative to the estimated mean $\hat{\boldsymbol{y}}$ and, therefore, biased. In fact, the maximum likelihood solution systematically underestimates σ^2, which is a phenomenon of overfitting the training set (Bishop 2006). The squared error $\|\boldsymbol{y} - \hat{\boldsymbol{y}}\|^2$ will be lower on the training set and $\hat{\sigma}^2$ on new samples will be systematically too low (see Fig. 3.2). This is a problem especially in deep learning, where large models have millions of parameters and tend to overfit. To solve this issue, we introduce a simple learnable scalar parameter s to rescale the biased estimation of σ^2.

3.2.3 σ Scaling for Aleatoric Uncertainty

We first derive σ scaling for aleatoric uncertainty. Using a Gaussian model, we scale the standard deviation σ with a scalar value s to recalibrate the probability density function

$$p\left(\boldsymbol{y}|\boldsymbol{x};\hat{\boldsymbol{y}}(\boldsymbol{x}),\hat{\sigma}^2(\boldsymbol{x})\right)=\mathcal{N}\left(\boldsymbol{y};\hat{\boldsymbol{y}}(\boldsymbol{x}),(s\cdot\hat{\sigma}(\boldsymbol{x}))^2\right)\ . \tag{3.9}$$

This results in the following minimization objective:

$$\mathcal{L}_{\mathrm{G}}(s)=m\log(s)+\tfrac{1}{2}s^{-2}\sum_{i=1}^{m}\left(\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}\right)^{-2}\left\|\boldsymbol{y}^{(i)}-\hat{\boldsymbol{y}}_{\boldsymbol{\theta}}^{(i)}\right\|^2 . \tag{3.10}$$

Eq. (3.10) is optimized w.r.t. s with fixed $\boldsymbol{\theta}$ using gradient descent in a separate calibration phase after training to calibrate aleatoric uncertainty measured by $\hat{\sigma}_{\boldsymbol{\theta}}^2$. In case of a single scalar, the solution to Eq. (3.10) can also be written in closed form as

$$s=\pm\sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}\right)^{-2}\left\|\boldsymbol{y}^{(i)}-\hat{\boldsymbol{y}}_{\boldsymbol{\theta}}^{(i)}\right\|^2}\ . \tag{3.11}$$

We apply σ scaling to jointly calibrate aleatoric and epistemic uncertainty in § 3.2.5.

3.2.4 Laplacian Model

Using $\mathsf{Laplace}(\hat{\boldsymbol{y}}(\boldsymbol{x}),\hat{\sigma}(\boldsymbol{x}))$ as model, the conditional log-likelihood is given by

$$\log p(\boldsymbol{Y}\mid\boldsymbol{X},\boldsymbol{\theta})=\sum_{i=1}^{m}\log\left(\frac{1}{2\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}}\exp\left\{-\frac{\left|\boldsymbol{y}^{(i)}-\hat{\boldsymbol{y}}_{\boldsymbol{\theta}}^{(i)}\right|}{\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}}\right\}\right) \tag{3.12}$$

$$=-\sum_{i=1}^{m}\log\left(2\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}\right)+\left(\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}\right)^{-1}\left|\boldsymbol{y}^{(i)}-\hat{\boldsymbol{y}}_{\boldsymbol{\theta}}^{(i)}\right|\ , \tag{3.13}$$

which results in the following minimization criterion:

$$\mathcal{L}_{\mathrm{L}}(\boldsymbol{\theta})=\sum_{i=1}^{m}\frac{1}{\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}}\left|\boldsymbol{y}^{(i)}-\hat{\boldsymbol{y}}_{\boldsymbol{\theta}}^{(i)}\right|+\log\left(\hat{\sigma}_{\boldsymbol{\theta}}^{(i)}\right)\ . \tag{3.14}$$

Using $\mathcal{L}_{\mathrm{L}}(\boldsymbol{\theta})$ instead of $\mathcal{L}_{\mathrm{G}}(\boldsymbol{\theta})$ results in applying an L1 metric on the predictive mean. In some cases, this led to better results. However, we have not conducted extensive experiments with it and leave it to future work.

3.2.5 Well-Calibrated Estimation of Predictive Uncertainty

So far we have assumed a MAP point estimate for $\boldsymbol{\theta}$ which does not consider uncertainty in the parameters. To quantify both aleatoric and epistemic uncertainty, we extend $\boldsymbol{f_\theta}$ into a fully Bayesian model under the variational inference framework with Monte Carlo dropout (Gal and Ghahramani 2016b). In MC dropout, the model $\boldsymbol{f}_{\tilde{\boldsymbol{\theta}}}$ is trained with dropout (Srivastava et al. 2014) and dropout is applied at test time by performing N stochastic forward passes to sample from the approximate Bayesian posterior $\tilde{\boldsymbol{\theta}} \sim q(\boldsymbol{\theta})$. Following (Kendall and Gal 2017), we use MC integration to approximate the predictive variance

$$\hat{\Sigma}^2 = \underbrace{\frac{1}{N}\sum_{n=1}^{N}\left(\hat{\boldsymbol{y}}_n - \frac{1}{N}\sum_{n=1}^{N}\hat{\boldsymbol{y}}_n\right)^2}_{\text{epistemic}} + \underbrace{\frac{1}{N}\sum_{n=1}^{N}\hat{\sigma}_n^2}_{\text{aleatoric}} \tag{3.15}$$

and use $\hat{\Sigma}^2$ as a measure of predictive uncertainty. If the neural network has multiple outputs ($d > 1$), the predictive variance is calculated per output and the mean across d forms the final uncertainty value. Eq. (3.15) is an unbiased estimator of the approximate predictive variance (see proof in Appendix 3.2.6). From Eq. (3.24) of our proof follows, that $\hat{\Sigma}^2$ is expected to equal the true variance $\Sigma = \mathbb{E}[(\hat{\boldsymbol{y}} - \boldsymbol{y})^2]$. Thus, we define perfect calibration of regression uncertainty as

$$\mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}\left[\mathbb{E}[(\hat{\boldsymbol{y}} - \boldsymbol{y})^2] \,\middle|\, \hat{\Sigma}^2 = \alpha^2\right] = \alpha^2 \quad \forall \left\{\alpha^2 \in \mathbb{R} \,|\, \alpha^2 \geq 0\right\} \ , \tag{3.16}$$

which extends the definition of (Levi et al. 2019) to both aleatoric and epistemic uncertainty. We expect that additionally accounting for epistemic uncertainty is particularly beneficial for smaller data sets. However, even in deep learning with Bayesian principles, the approximate posterior predictive distribution can overfit on small data sets. In practice, this leads to underestimation of the predictive uncertainty.

One could regularize overfitting by early stopping that prevents large differences between training and test loss, which would circumvent underestimation of σ^2. However, our experiments show that early stopping is not optimal with regard to accuracy, i.e. the squared error of $\hat{\boldsymbol{y}}$ on both training and testing data (see Fig. 3.2). In contrast, the model with lowest mean error on the validation set underestimates predictive uncertainty considerably. Therefore, we apply σ scaling to recalibrate the predictive uncertainty $\hat{\Sigma}^2$. This allows a lower squared error while reducing underestimation of uncertainty as shown experimentally in the following section.

3.2.6 Unbiased Estimator of the Approximate Predictive Variance

We show that the expectation of the predictive sample variance from MC dropout, as given in (Kendall and Gal 2017), equals the true variance of the approximate posterior predictive distribution.

Proposition 1. *Given N MC dropout samples $\boldsymbol{f}_{\boldsymbol{\theta}_n} = [\hat{\boldsymbol{y}}_n, \hat{\sigma}_n^2]$ from our approximate predictive distribution $p(\boldsymbol{y}^* | \boldsymbol{x}^*, \mathcal{D}) = \mathcal{N}(\boldsymbol{y}^*; \boldsymbol{y}, \Sigma^2)$, the predictive sample variance*

$$\hat{\Sigma}^2 = \frac{1}{N}\sum_{n=1}^{N}\left(\hat{\boldsymbol{y}}_n - \frac{1}{N}\sum_{n=1}^{N}\hat{\boldsymbol{y}}_n\right)^2 + \frac{1}{N}\sum_{n=1}^{N}\hat{\sigma}_n^2 \tag{3.17}$$

is an unbiased estimator of the approximate predictive variance.

Proof.

$$\begin{aligned}
\mathbb{E}\left[\hat{\Sigma}^2\right] &= \mathbb{E}\left[\frac{1}{N}\sum_{n=1}^{N}\left(\hat{\boldsymbol{y}}_n - \frac{1}{N}\sum_{n=1}^{N}\hat{\boldsymbol{y}}_n\right)^2 + \frac{1}{N}\sum_{n=1}^{N}\hat{\sigma}_n^2\right] && (3.18)\\
&= \mathbb{E}\left[\frac{1}{N}\sum_{n=1}^{N}\left(\hat{\boldsymbol{y}}_n - \frac{1}{N}\sum_{n=1}^{N}\hat{\boldsymbol{y}}_n\right)^2\right] + \mathbb{E}\left[\frac{1}{N}\sum_{n=1}^{N}\hat{\sigma}_n^2\right] && (3.19)\\
&\text{with} \quad \frac{1}{N}\sum_{n=1}^{N}\hat{\boldsymbol{y}}_n = \bar{\boldsymbol{y}} \quad \text{follows} && (3.20)\\
&= \mathbb{E}\left[\frac{1}{N}\sum_{n=1}^{N}(\hat{\boldsymbol{y}}_n - \bar{\boldsymbol{y}})^2\right] + \hat{\sigma}^2 && (3.21)\\
&= \mathbb{E}\left[\frac{1}{N}\sum_{n=1}^{N}(\hat{\boldsymbol{y}}_n - \bar{\boldsymbol{y}})^2 + \bar{\boldsymbol{y}}^2 - \bar{\boldsymbol{y}}^2 + \boldsymbol{y}^2 - \boldsymbol{y}^2 + 2\bar{\boldsymbol{y}}\boldsymbol{y} - 2\bar{\boldsymbol{y}}\boldsymbol{y}\right] + \hat{\sigma}^2 && (3.22)\\
&= \mathbb{E}\left[\frac{1}{N}\sum_{n=1}^{N}(\hat{\boldsymbol{y}}_n - \boldsymbol{y})^2 - (\bar{\boldsymbol{y}} - \boldsymbol{y})^2\right] + \hat{\sigma}^2 && (3.23)\\
&= \mathbb{E}\left[(\hat{\boldsymbol{y}} - \boldsymbol{y})^2\right] - \mathbb{E}\left[(\bar{\boldsymbol{y}} - \boldsymbol{y})^2\right] + \hat{\sigma}^2 && (3.24)\\
&= \Sigma^2 - \hat{\sigma}^2 + \hat{\sigma}^2 && (3.25)\\
\mathbb{E}\left[\hat{\Sigma}^2\right] &= \Sigma^2 && (3.26)
\end{aligned}$$

Note that the predicted heteroscedastic aleatoric uncertainty $\hat{\sigma}^2$ equals the bias $\mathbb{E}[(\bar{\boldsymbol{y}} - \boldsymbol{y})^2]$ in Eq. (3.24) when the aleatoric uncertainty is perfectly calibrated, thus $\mathbb{E}[(\bar{\boldsymbol{y}} - \boldsymbol{y})^2] = \hat{\sigma}^2$. □

3.2.7 Expected Uncertainty Calibration Error for Regression

We extend the definition of the uncertainty calibration error for classification (cf. § 2.1) to quantify miscalibration of uncertainty in regression

$$\mathbb{E}_{\hat{\Sigma}^2}\left[\left|\left(\mathbb{E}[(\hat{\boldsymbol{y}} - \boldsymbol{y})^2] \,\middle|\, \hat{\Sigma}^2 = \alpha^2\right) - \alpha^2\right|\right] \quad \forall\left\{\alpha^2 \in \mathbb{R} \,|\, \alpha^2 \geq 0\right\}\ , \tag{3.27}$$

using the second moment of the error. On finite data sets, this can be approximated with the expected uncertainty calibration error (UCE) for regression. Following (Guo et al. 2017), the uncertainty output $\hat{\Sigma}^2$ of a deep model is partitioned into K bins with equal width. A weighted average of the difference between the variance and predictive uncertainty is used:

$$\mathrm{UCE} := \sum_{k=1}^{K} \frac{|B_k|}{m} \big| \mathrm{var}(B_k) - \mathrm{uncert}(B_k) \big| \,, \tag{3.28}$$

with number of inputs m and set of indices B_k of inputs, for which the uncertainty falls into the bin k. The variance per bin is defined as

$$\mathrm{var}(B_k) := \frac{1}{|B_k|} \sum_{i \in B_m} \frac{1}{N} \sum_{n=1}^{N} \left(\hat{\boldsymbol{y}}_{i,n} - \boldsymbol{y}_i \right)^2 \,, \tag{3.29}$$

with N stochastic forward passes, and the uncertainty per bin is defined as

$$\mathrm{uncert}(B_k) := \frac{1}{|B_k|} \sum_{i \in B_k} \hat{\Sigma}_i^2 \,. \tag{3.30}$$

Note that computing the second moment from Eq. (3.27) also incorporates MC samples, which can introduce some bias in the evaluation. The UCE considers both aleatoric and epistemic uncertainty and is given in % throughout this work. Additionally, we plot $\mathrm{var}(B_k)$ vs. $\mathrm{uncert}(B_k)$ to create calibration diagrams.

3.3 Experiments

We use four data sets and three common deep network architectures to evaluate recalibration with σ scaling. The data sets were selected to represent various regression tasks in medical imaging with different dimension d of target value $\boldsymbol{y} \in \mathbb{R}^d$:

(1) Estimation of tumor cellularity in histology whole slides of cancerous breast tissue from the BreastPathQ SPIE challenge data set ($d = 1$) (Martel et al. 2019). The public data set consists of 2579 images, from which 1379/600/600 are used for training/validation/testing. The ground truth label is a single scalar $y \in [0, 1]$ denoting the ratio of tumor cells to non-tumor cells.

(2) Hand CT age regression from the RSNA pediatric bone age data set ($d = 1$) (Halabi et al. 2019). The task is to infer a person's age in months from CT scans of the hand. This data set is the largest used in this chapter and has 12,811 images, from which we use 6811/2000/4000 images for training/validation/testing.

(3) Surgical instrument tracking on endoscopic images from the EndoVis endoscopic vision challenge 2015[1] data set ($d = 2$). This data set contains 8,984 video frames from 6

---

[1] endovissub-instrument.grand-challenge.org

different robot-assisted laparoscopic interventions showing surgical instruments with ground truth pixel coordinates of the instrument's center point $\boldsymbol{y} \in \mathbb{R}^2$. We use 4483/2248/2253 frames for training/validation/testing. As the public data set is only sparsely annotated, we created our own ground truth labels, which can be found in our code repository.

(4) 6DoF needle pose estimation on optical coherence tomography (OCT) scans from our own data set[2]. This data set contains 5,000 3D-OCT scans with the accompanying needle pose $\boldsymbol{y} \in \mathbb{R}^6$, from which we use 3300/850/850 for training/validation/testing. Additional details on this data set can be found in Appendix 3.3.2.

All outputs are normalized such that $\boldsymbol{y} \in [0,1]^d$. The employed network architectures are ResNet-101, DenseNet-201 and EfficientNet-B4 (He, X. Zhang, et al. 2016; Huang et al. 2017; M. Tan and Le 2019), as they represent the state-of-the-art of deep models. The last linear layer of all networks is replaced by two linear layers predicting $\hat{\boldsymbol{y}}$ and $\hat{\sigma}^2$ as described in § 3.2.1. For MC dropout, we use dropout before the last linear layers. Dropout is further added after each of the four layers of stacked residual blocks in ResNet. In DenseNet and EfficientNet, we use the default configuration of dropout during training and testing. The networks are trained until no further decrease in mean squared error (MSE) on the validation set can be observed. More details on the training procedure can be found in § 3.3.1.

Calibration is performed after training in a separate calibration phase using the validation data set. We plug the predictive uncertainty $\hat{\Sigma}^2$ into Eq. (3.10) (instead of $\hat{\sigma}^2$) and minimize w.r.t. s. Additionally, we compare σ scaling to a more powerful auxiliary recalibration model $\boldsymbol{R}$ consisting of a two-layer fully-connected network with 16 hidden units and ReLU activations (inspired by (Kuleshov et al. 2018), see § 3.1).

3.3.1 Training Procedure

The model implementations from PyTorch 1.3 (Paszke et al. 2019) are used and trained with the following settings:

- training for 500 epochs with batch size of 16
- Adam optimizer with initial learn rate of $3 \cdot 10^{-4}$ and weight decay with $\lambda = 10^{-7}$
- reduce-on-plateau learn rate scheduler (patience of 20 epochs) with factor of 0.1
- in MC dropout, $N = 25$ forward passes were performed with dropout with $p = 0.5$ used for ResNet (as described in (Gal and Ghahramani 2016b)). In DenseNet ($p = 0.2$) and EfficientNet ($p = 0.4$) standard dropout p of the architecture is used.
- Additional validation and test sets are used if provided by the data sets; otherwise, a train/validation/test split of approx. 50% / 25% / 25% is used
- Source code for all experiments is available at github.com/mlaves/well-calibrated-regression-uncertainty

---

[2] Our OCT pose estimation data set is publicly available at github.com/mlaves/3doct-pose-dataset

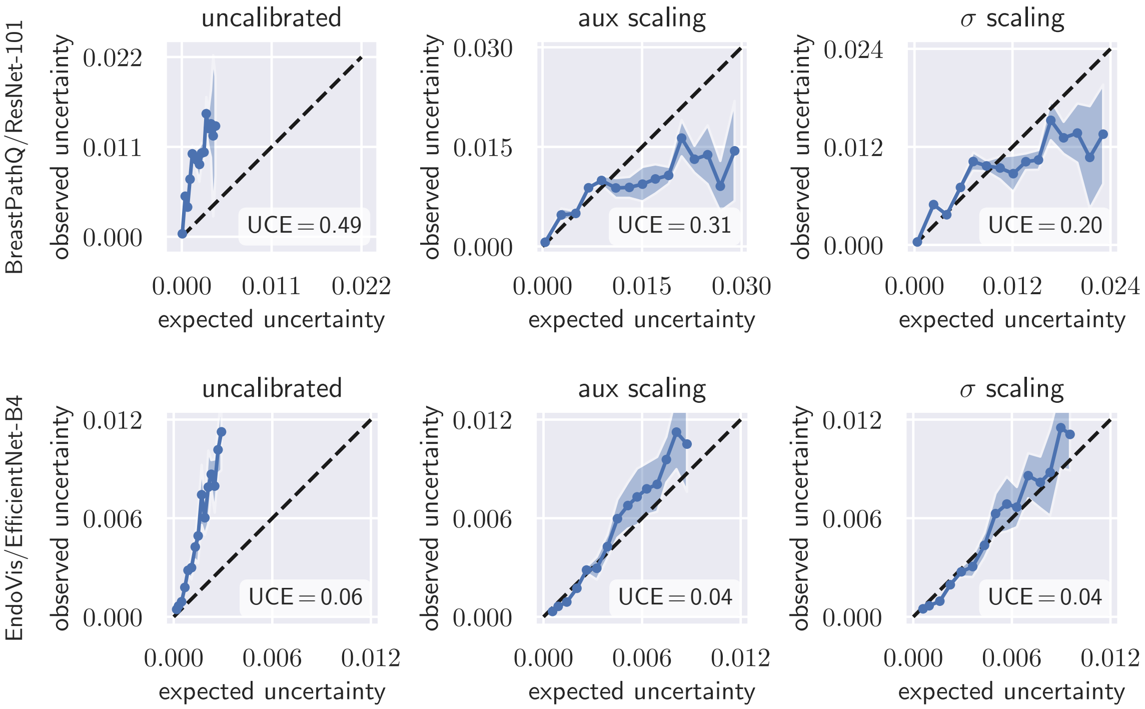


Figure 3.3: Calibration plots for ResNet-101 on BreastPathQ (top row) and EfficientNet-B4 on EndoVis (bottom row). Aux scaling tends to overfit the calibration set, which results in higher UCE compared to simple σ scaling. Dashed lines denote perfect calibration.

3.3.2 3D OCT Needle Pose Data Set

Our data set was created by attaching a surgical needle to a high-precision six-axis hexapod robot (H-826, Physik Instrumente GmbH & Co. KG, Germany) and observing the needle tip with 3D optical coherence tomography (OCS1300SS, Thorlabs Inc., USA). The data set consists of 5,000 OCT acquisitions with $(64 \times 64 \times 512)$ voxels, covering a volume of approx. $(3 \times 3 \times 3)\,\mathrm{mm}^3$. Each acquisition is taken at a different robot configuration and labeled with the corresponding 6DoF pose $\boldsymbol{y} \in \mathbb{R}^6$. To process the volumetric data with CNNs for planar images, we calculate 3 planar projections along the spatial dimensions using the $\arg\max$ operator, scale them to equal size and stack them together as three-channel image (see Fig. 3.4). A similar approach was presented in (Laves, Schoob, et al. 2017) and (Gessert et al. 2018). The data are characterized by a high amount of speckle noise, which is a typical phenomenon in optical coherence tomography. The data set is publicly available at github.com/mlaves/3doct-pose-dataset.

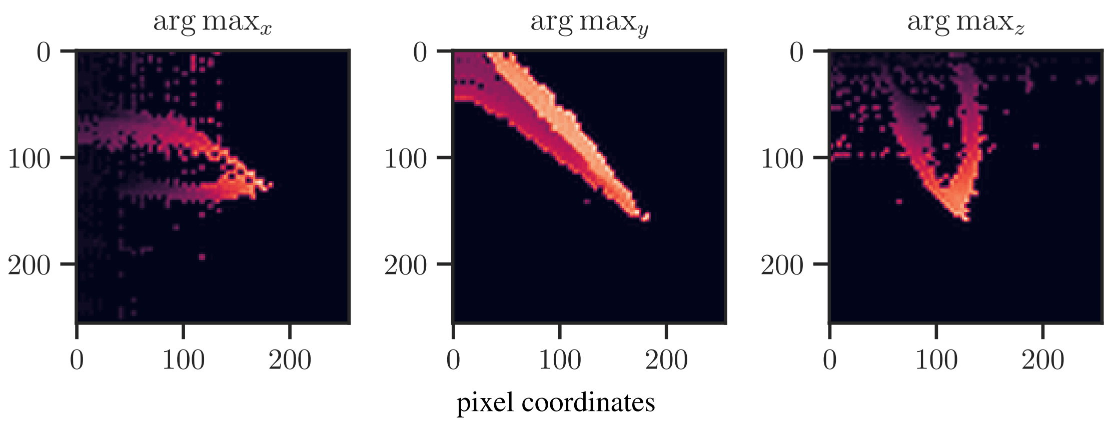


Figure 3.4: Example image from OCT data set showing $\arg\max$ projections of a surgical needle tip acquired by optical coherence tomography.

3.4 Results

To quantify miscalibration, we use the proposed expected uncertainty calibration error for regression (§ 3.2.7). We visualize (mis-)calibration in Fig. 3.1 and Fig. 3.3 using calibration diagrams, which show expected uncertainty vs. observed uncertainty. The discrepancy to the identity function reveals miscalibration. The calibration diagrams clearly show the underestimation of uncertainty for the uncalibrated models. After calibration with both aux and σ scaling, the estimated uncertainty better reflects the actual uncertainty. Figures for all configurations are listed in Appendix A.2.1.

Table 3.1 reports UCE values of all data set/model combinations on the respective test sets. The negative log-likelihood also measures miscalibration; the values on the test set can be found in Tab. A.1 in the appendix. In general, recalibration considerably reduces miscalibration. On the data sets BoneAge, EndoVis and OCT, both scaling methods perform similarly well. However, on the BreastPathQ data set, σ scaling clearly outperforms aux scaling in terms of UCE. BreastPathQ is the smallest data set and thus has the smallest calibration set size. We hypothesize that the more powerful auxiliary model $\boldsymbol{R}$ overfits the calibration set (see BreastPathQ/DenseNet-201 in Tab. 3.1), which leads to an increase of UCE on the test set. An ablation study on BreastPathQ for the auxiliary model can be found in § 3.4.3.

We also compare our approach to Levi et al. (2019) in Tab. 3.1, which only considers aleatoric uncertainty. The aleatoric uncertainty is well-calibrated if it reflects the bias $(\mathbb{E}[\hat{\boldsymbol{y}}_n] - \boldsymbol{y})^2$, which is given by the squared error between the expectation of the stochastic predictions $\hat{\boldsymbol{y}}_n$ and the ground truth. Therefore, the UCE for aleatoric-only is computed by $\mathrm{UCE} = \sum_{k=1}^{K} \frac{|B_k|}{m} \big| \mathrm{err}(B_k) - \mathrm{uncert}(B_k) \big|$, where $\mathrm{err}(\cdot)$ is the mean squared error and $\mathrm{uncert}(\cdot)$ is the mean aleatoric uncertainty per bin. Consideration of epistemic uncertainty is beneficial on smaller data sets (BreastPathQ), where our approach outperforms Levi et al. (2019). On larger data sets, the benefit diminishes and both approaches are equally calibrated.

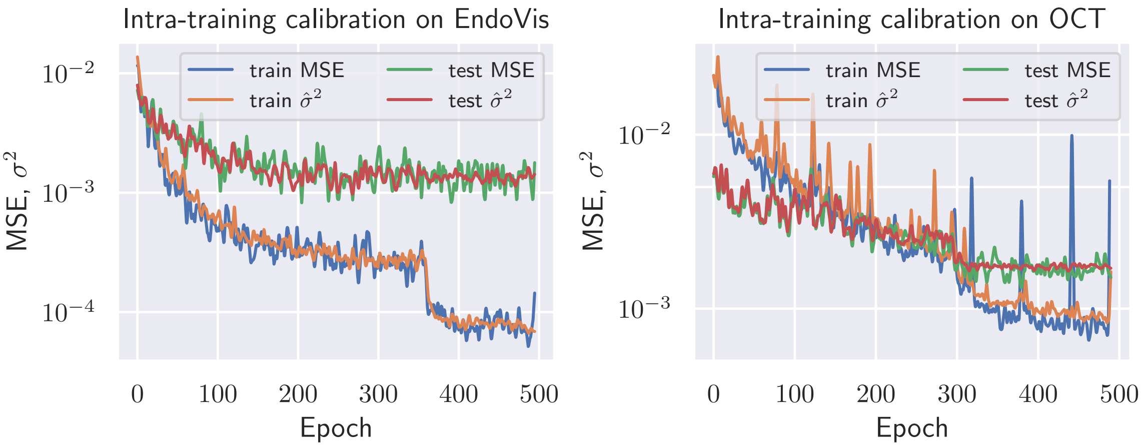


Figure 3.5: (Left) Intra-training calibration of aleatoric uncertainty with σ scaling. The deep model no longer underestimates $\hat{\sigma}^2$ on unseen test data. (Right) The MSE of predictive mean is higher and σ^2 is underestimated. Note: Calibration is only applied at test time.

Additionally, we report UCE values from a DenseNet ensemble for comparison. In contrast to what is reported by Lakshminarayanan et al. (2017), the deep ensemble tends to be calibrated worse. Only on BoneAge, the ensemble is better calibrated prior to recalibration of the other methods. After recalibration, both approaches outperform the deep ensemble.

Fig. 3.5 shows the result of intra-training calibration of aleatoric uncertainty. It indicates that the gap between training and test loss is successfully closed. For the remaining experiments, however, the calibration is performed after the training.

3.4.1 Posterior Prediction Intervals

In addition to the calibration diagrams, we compute prediction intervals from the uncalibrated and calibrated posterior predictive distribution. Well-calibrated prediction intervals provide a reliable measure of precision of the estimated target value. In Bayesian inference, prediction intervals define an interval within which the true target value $\boldsymbol{y}^*$ of a new, unobserved input $\boldsymbol{x}^*$ is expected to fall with a specific probability (Held and Sabanés Bové 2014; Heskes 1997). This is also referred to as the credible interval of the posterior predictive distribution. For $\gamma \in (0, 1)$, a $\gamma \cdot 100\,\%$ prediction interval is defined through z_l and z_u such that

$$\int_{z_l}^{z_u} p(\boldsymbol{y}^* \,|\, \boldsymbol{x}^*, \mathcal{D}) \, \mathrm{d}\boldsymbol{y}^* = \gamma \,, \tag{3.31}$$

with posterior predictive distribution $p(\boldsymbol{y}^* \,|\, \boldsymbol{x}^*, \mathcal{D})$. We compute the 50 %, 90 %, 95 %, and 99 % prediction interval using the root of the predictive variance from Eq. (3.15); that is, the $\hat{\boldsymbol{y}} \pm z\hat{\Sigma}$ intervals with $z \in \{\Phi(0.5), \Phi(0.9), \Phi(0.95), \Phi(0.99)\}$ (estimated interval), with probit function $\Phi(p) = \sqrt{2}\mathrm{erf}^{-1}(p)$ and $\mathrm{erf}(p)$ is the Gaussian error function. This assumes

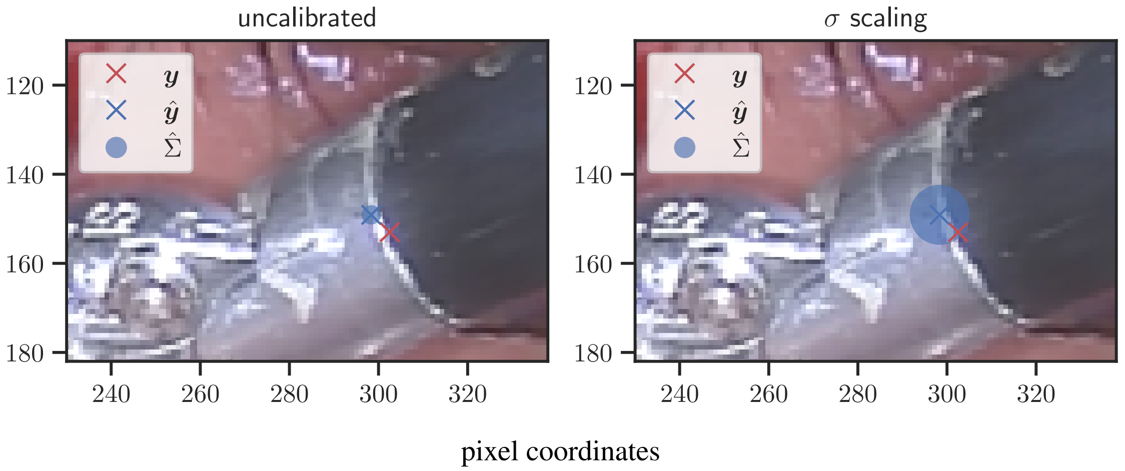


Figure 3.6: Example result from EndoVis test set. The task is to predict pixel coordinates of the forceps shaft center. Before calibration, the uncertainty is underestimated and the true instrument position $\boldsymbol{y}$ does not fall into the prediction region $\hat{\boldsymbol{y}} \pm \hat{\Sigma}$. After calibration with σ scaling, the uncertainty better reflects the predictive error.

Table 3.1: Uncertainty calibration error test set results for different datasets and model architectures (averaged over 5 runs). High UCE values indicate miscalibration. In addition, the resulting s for σ scaling is given. We also report UCE values for an ensemble of DenseNets. Bold font indicates lowest values in each experiment.

			Levi et al.				ours				
Data Set	Model	MSE	none	aux	σ	s	none	aux	σ	s	ensemble
BreastPathQ	ResNet-101	6.4e-3	0.51	0.35	0.28	2.91	0.49	0.31	**0.20**	2.37	0.51
	DenseNet-201	7.0e-3	0.21	0.38	**0.15**	1.62	**0.11**	0.36	0.15	1.33	
	EfficientNet-B4	6.4e-3	0.49	0.65	**0.10**	2.30	0.46	0.47	0.17	1.77	
BoneAge	ResNet-101	5.3e-3	0.28	0.07	0.06	1.46	0.28	**0.02**	0.06	1.40	0.09
	DenseNet-201	3.5e-3	0.31	**0.05**	**0.05**	2.98	0.31	**0.05**	**0.05**	2.54	
	EfficientNet-B4	3.5e-3	0.30	0.05	0.10	4.83	0.30	**0.03**	0.12	3.98	
EndoVis	ResNet-101	4.0e-4	**0.04**	0.10	0.09	6.07	**0.04**	**0.04**	**0.04**	3.50	0.08
	DenseNet-201	1.1e-3	0.09	0.05	0.05	3.24	**0.04**	**0.04**	**0.04**	2.57	
	EfficientNet-B4	8.9e-4	0.06	0.05	0.06	2.25	0.06	**0.04**	**0.04**	1.79	
OCT	ResNet-101	2.0e-3	0.17	0.02	0.02	2.74	0.17	**0.01**	0.02	2.14	0.67
	DenseNet-201	1.3e-3	0.08	**0.01**	0.02	1.60	0.04	0.03	0.02	1.26	
	EfficientNet-B4	1.4e-3	0.12	**0.01**	**0.01**	2.65	0.12	**0.01**	**0.01**	1.94	

that the posterior predictive distribution is Gaussian, which is not generally the case. To assess the calibration of the posterior prediction interval, we compute the percentage of how many of the ground truth values of the test set actually fall within the respective intervals (observed interval). In Fig. 3.7, selected plots of observed vs. estimated prediction intervals are shown. A complete list of prediction intervals can be found in Appendix A.2.2.

In general, the uncalibrated prediction intervals are estimated to be too narrow, which is a direct consequence of the underestimated predictive variance. For example, the uncalibrated 90 % interval on DenseNet-201/BoneAge actually only contains approx. 50 % of the ground truth values. On this data set, the prediction intervals are considerably improved after recalibration (Fig. 3.7 left). If a network is already well-calibrated, recalibration can lead to overestimation of the lower prediction intervals (Fig. 3.7 right). However, in all cases, the 99 % prediction interval contains approx. 99 % of the ground truth test set values after recalibration. This is not the case without the proposed calibration methods. Fig. 3.6 shows a practical example of the $\hat{\boldsymbol{y}} \pm \hat{\Sigma}$ prediction region from the EndoVis test set. Even though the posterior predictive distribution is not necessarily Gaussian, the calibrated results fit the prediction intervals well. This is especially the case for BoneAge, which is the largest data set used in this chapter.

3.4.2 Detection of Out-of-Distribution Data and Unreliable Predictions

Deep neural networks only yield reliable predictions for data which follow the same distribution as the training data. A shift in distribution could occur when a model trained on CT data from a specific CT device is applied to data from another manufacturer's CT device, for example. This could potentially lead to wrong predictions with low uncertainty, which we tackle with recalibration. To create a moderate distribution shift, we preprocess images from the BoneAge data set using Contrast Limited Adapative Histogram Equalization (CLAHE) (Pizer et al. 1987) with a clip-limit of 0.03 and report histograms of the uncertainties (see Fig. 3.8). Additionally, a severe distribution shift is created by presenting images from the BreastPathQ data set to the models trained on BoneAge. Lakshminarayanan et al. (2017) state that deep ensembles provide better-calibrated uncertainty than Bayesian neural networks with MC dropout variational inference. Therefore, we train an ensemble of 5 randomly initialized DenseNet-201 and compare Bayesian uncertainty with σ scaling to ensemble uncertainty under distribution shift. The results with σ scaling are comparable to those from a deep ensemble for a moderate shift, but without the need to train multiple models on the same data set. A severe shift leads to only slightly increased uncertainties from the calibrated MC dropout model, while the deep ensemble is more sensitive.

Additionally, we apply the well-calibrated models to detect and reject uncertain predictions, as crucial decisions in medical practice should only be made on the basis of reliable predictions. An uncertainty threshold Σ^2_{max} is defined and all predictions from the test set are rejected where $\hat{\Sigma}^2 > \Sigma^2_{\text{max}}$ (see Fig. 3.9). From this, a decrease in overall MSE is expected. We additionally compare rejection on the basis of σ scaled uncertainty to uncertainty from the aforementioned ensemble. In case of σ scaling, the test set MSE decreases monotonically as a function of the uncertainty threshold, whereas the ensemble initially shows an increasing MSE (see Fig.3.9).

3.4.3 Ablation Study on Auxiliary Model Scaling

We investigate the overfitting behavior of aux scaling by reducing the number of hidden layer units h of the two-layer auxiliary model with ReLU activations. Aux scaling is more powerful than σ scaling, which can lead to overfitting the calibration set. Fig. 3.10 shows calibration diagrams for the auxiliary model ablations. Reducing h leads to a minor calibration improvement, but at $h = 2$, the model outputs a constant uncertainty, which is close to the overall mean of the observed uncertainty. A single-layer single-unit model without bias would be equivalent to σ scaling.

3.5 Chapter Conclusion

In this chapter, well-calibrated predictive uncertainty in medical imaging obtained by variational inference with deep Bayesian models is discussed. Both aux and σ scaling calibration methods considerably reduce miscalibration of predictive uncertainty in terms of UCE. If the deep model is already well-calibrated, σ scaling does not negatively affect the calibration, which results in $s \to 1$. More complex calibration methods such as aux scaling have to be used with caution, as they can overfit the data set used for calibration. If the calibration set is sufficiently large, they can outperform simple scaling. However, models trained on large data sets are generally better calibrated and the benefit diminishes. Compared to the work of Levi et al. (2019), accounting for epistemic uncertainty is particularly beneficial for smaller data sets, which is helpful in medical practice where access to large labeled data sets is less common and is associated with great costs.

Posterior prediction intervals provide another insight into the calibration of deep models. After recalibration, the 99 % posterior prediction intervals correctly contain approx. 99 % of the ground truth test set values. In some cases, lower prediction intervals are estimated to be too wide after calibration. This is especially the case for smaller data sets and we conjecture that small calibration sets may not contain enough i.i.d. data for calibrating lower prediction intervals and that the assumption of a Gaussian predictive distribution is too strong in this case. On the smallest data set BreastPathQ, aux scaling seems to perform better in terms of prediction intervals, but not in terms of UCE.

Well-calibrated uncertainties from MC dropout are able to detect a moderate shift in the data distribution. However, deep ensembles perform better under a severe distribution shift. BNNs with calibrated uncertainty by σ scaling outperform ensemble uncertainty in the rejection task, which we attribute to the generally poorer calibration of ensembles on in-distribution data.

σ scaling is simple to implement, does not change the predictive mean $\hat{\boldsymbol{y}}$, and therefore guarantees to conserve the model's accuracy. It is preferable to regularization (e.g., early

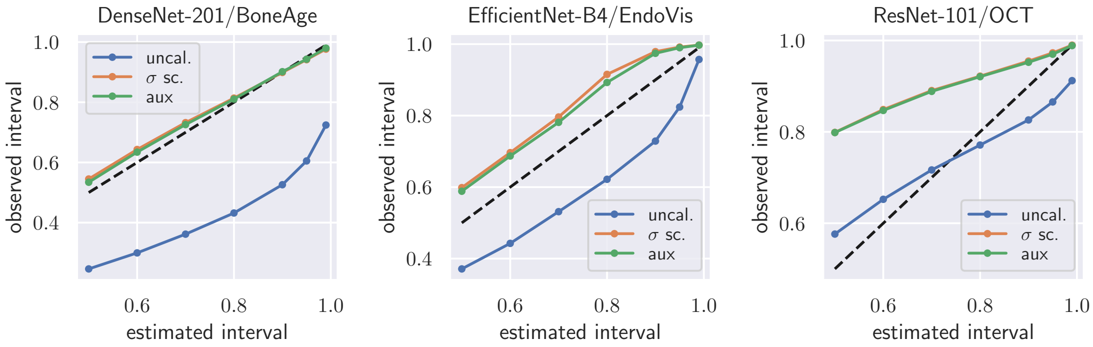


Figure 3.7: Observed vs. estimated posterior prediction intervals on the test sets. (Left & center) The uncalibrated prediction interval is too narrow due to underestimation of uncertainty. (Right) Calibration can lead to overestimation of predictive intervals, if the network is already well-calibrated. Dashed lines denote 1:1 mapping.

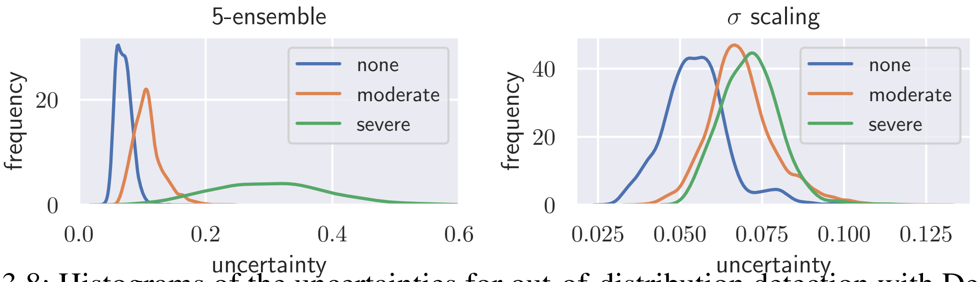


Figure 3.8: Histograms of the uncertainties for out-of-distribution detection with DenseNet-201 on BoneAge test set. (Left) Uncertainties from a non-Bayesian ensemble of five DenseNets and (right) Bayesian uncertainties calibrated with σ scaling. The distribution shifts have been created with pre-processing by CLAHE (moderate) and images from a different domain (severe).

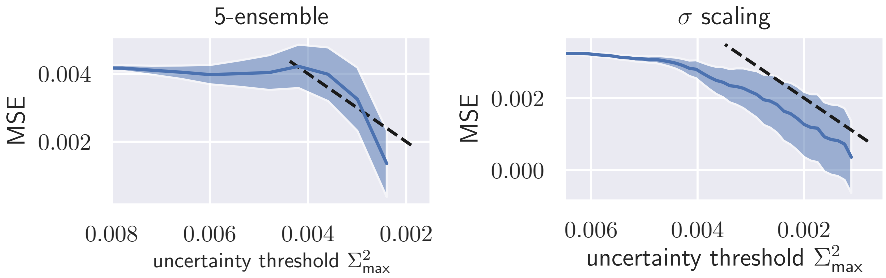


Figure 3.9: Rejection of uncertain predictions with DenseNet-201 on BoneAge test set with $\hat{\Sigma}^2 > \Sigma^2_{\text{max}}$. The shaded area width visualizes the percentage of rejected samples. The dashed line visualizes linear relationship.

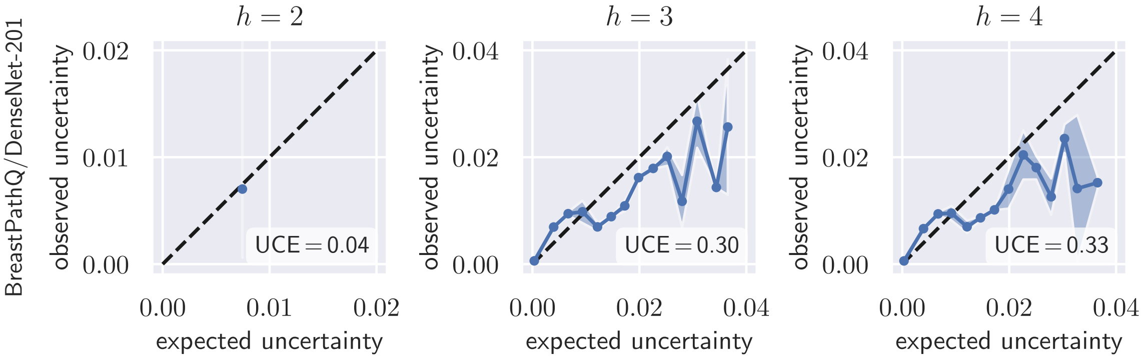


Figure 3.10: Calibration diagrams for aux scaling with different number of hidden layer units h on BreastPathQ/DenseNet-201.

stopping) or more complex recalibration methods in calibrated uncertainty estimation with Bayesian deep learning. The disconnection between training and test NLL can successfully be closed, which creates highly accurate models with reliable uncertainty estimates. However, there are many factors (e.g., network capacity, weight decay, dropout configuration) influencing the uncertainty that have not been discussed here and will be addressed in future work.

4 Generative Models in Medical Imaging

The first part of this chapter was partly published at the peer-reviewed “Uncertainty for Safe Utilization of Machine Learning in Medical Imaging” (UNSURE) 2020 workshop at MICCAI and presented as long oral; it was also selected as best paper (Laves, Tölle, et al. 2020). In § 4.2, a non Bayesian approach to deformable registration with deep image prior is presented, which was published as a poster at the peer-reviewed “Medical Imaging with Deep Learning” (MIDL) 2019 conference (Laves, Ihler, and Ortmaier 2019). Subsequently, a submission that emerged from and extends the contents of this chapter was accepted as long oral (best 7.2 % of all submissions) at the peer-reviewed MIDL 2021 conference and invited to be published in the international journal “Medical Image Analysis” (Tölle[1] et al. 2021). The source code for all experiments in this chapter is publicly available at: github.com/mlaves/uncertainty-deep-image-prior

4.1 Medical Image Denoising with Bayesian Deep Image Prior

Noise in medical imaging affects all modalities, including X-ray, magnetic resonance imaging (MRI), computed tomography (CT), ultrasound (US) or optical coherence tomography (OCT) and can obstruct important details for medical diagnosis (Agostinelli et al. 2013; Gondara 2016; Laves, Ihler, Kahrs, et al. 2019b). Besides “classical” approaches with linear and non-linear filters, such as the Wiener or wavelet filter (Chang et al. 2000; Rabbani et al. 2009), convolutional neural networks have proven to yield superior performance in denoising of natural and medical images (Laves, Ihler, Kahrs, et al. 2019b; K. Zhang et al. 2017).

Image denoising involves solving an inverse problem. However, uncertainty quantification in inverse medical imaging tasks with deep learning has received little attention. Deep models trained on large data sets tend to hallucinate and create artifacts in the reconstructed output that are not anatomically present. We use a randomly initialized convolutional network as parameterization of the reconstructed image and perform gradient descent to match the observation, which is known as deep image prior. In this case, the reconstruction does not suffer from hallucinations as no prior training is performed. In this chapter, we extend this to a Bayesian approach with Monte Carlo dropout to quantify both aleatoric and epistemic uncertainty. The presented method is evaluated on the task of denoising different medical imaging modalities. The experimental results show that our approach yields well-calibrated uncertainty. That is, the predictive uncertainty correlates with the predictive error. This allows for reliable uncertainty estimates and can tackle the problem of hallucinations and artifacts in inverse medical imaging tasks.

---

[1] Shared first authorship

4.1.1 Introduction

The task of denoising is an inverse image problem and aims at reconstructing a clean image $\hat{\boldsymbol{x}}$ from a noisy observation $\tilde{\boldsymbol{x}} = \boldsymbol{c} \circ \boldsymbol{x}$. A common assumption of the noise model $\boldsymbol{c}$ of the image $\tilde{\boldsymbol{x}}$ is additive white Gaussian noise with zero mean and standard deviation σ (Salinas and Fernandez 2007; K. Zhang et al. 2017). Given a noisy image $\tilde{\boldsymbol{x}}$, the denoising can be expressed as optimization problem of the form

$$\hat{\boldsymbol{x}} = \arg\min_{\hat{\boldsymbol{x}}} \left\{ \mathcal{L}(\tilde{\boldsymbol{x}}, \hat{\boldsymbol{x}}) + \lambda \mathcal{R}(\hat{\boldsymbol{x}}) \right\} . \tag{4.1}$$

The reconstruction $\hat{\boldsymbol{x}}$ should be close to $\tilde{\boldsymbol{x}}$ by means of a similarity metric $\mathcal{L}$, but with substantially less noise. The regularizer $\mathcal{R}$ expresses a prior on the reconstructed images, which leads to $\hat{\boldsymbol{x}}$ having less noise than $\tilde{\boldsymbol{x}}$. One usually imposes a smoothness constraint by penalizing first or higher order spatial derivatives of the image (Sotiras et al. 2013). More recently, denoising autoencoders have successfully been used to implicitly learn a regularization prior from a data set with corrupted and uncorrupted data samples (Jain and Seung 2009). Autoencoders are usually composed of an encoding and decoding part with a data bottleneck in between. The encoder extracts important visual features from the noisy input image and the decoder reconstructs the input from the extracted features using learned image statistics.

This, however, creates the root problem of medical image denoising with deep learning that is addressed in this paper. The reconstruction is in accordance with the expectation of the denoising autoencoder based on previously learned information. At worst, the reconstruction can contain false image features, that look like valid features, but are not actually present in the input image. Due to the excellent denoising performance of autoencoders, those false features can be indistinguishable from valid features to a layperson and are embedded in an otherwise visually appealing image. This phenomenon is known as *hallucination* and, while acceptable in the reconstruction of natural images (N. Wang et al. 2014), must be avoided at all costs in medical imaging (see Fig. 4.1). Hallucinations can lead to false diagnoses and thus severely compromise patient safety.

To further increase the reliability in the denoised medical images, the reconstruction uncertainty has to be considered. Bayesian autoencoders provide the mathematical framework to quantify a per-pixel reconstruction uncertainty (Bishop 2006; Z. Cheng et al. 2019; D. Kingma and Welling 2014). This allows the detection of hallucinations and other artifacts, given that the uncertainty is well-calibrated; i. e. the uncertainty corresponds well with the reconstruction error (Laves, Ihler, Fast, et al. 2020).

In this work, we employ *deep image prior* (Lempitsky et al. 2018) to cope with hallucinations in medical image denoising and provide a Bayesian approach with Monte Carlo (MC) dropout (Gal and Ghahramani 2016b) that yields well-calibrated reconstruction uncertainty. We present experimental results on denoising images from low-dose X-ray, ultrasound and OCT. Compared to previous work, our approach leads to better uncertainty estimates and is less prone to overfitting of the noisy image.

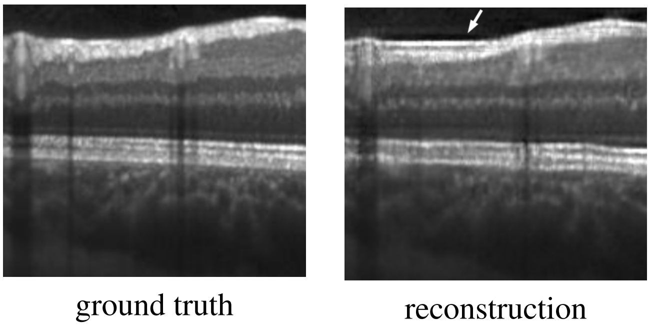


Figure 4.1: Hallucinations in reconstructed retinal OCT scan from supervisely trained CNN. (Left) Ground truth OCT scan. (Right) The white arrow denotes a hallucinated retinal layer that is anatomically incorrect. Hallucinations are the result of reconstructing an unseen noisy input using previously learned image statistics.

4.1.2 Related Work

Image priors Besides manually crafted priors such as 3D collaborative filtering (Dabov et al. 2007), convolutional denoising autoencoders have been used to implicitly learn an image prior from data (Gondara 2016; Jain and Seung 2009). Lempitsky et al. have recently shown that the excellent performance of deep networks for inverse image tasks, such as denoising, is based not only on their ability to learn image priors from data, but also on the structure of a convolutional image generator itself (Lempitsky et al. 2018). An image generator network $\hat{\boldsymbol{x}} = \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z})$ with randomly-initialized parameters $\boldsymbol{\theta}$ is interpreted as parameterization of the image. The parameters $\boldsymbol{\theta}$ of the network are found by minimizing the pixel-wise squared error $\|\tilde{\boldsymbol{x}} - \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z})\|$ with stochastic gradient descent (SGD). The input $\boldsymbol{z}$ is sampled from a uniform distribution with additional perturbations by normally distributed noise in every iteration. This is referred to as deep image prior (DIP). They provided empirical evidence that the structure of a CNN alone is sufficient to capture enough image statistics to provide state-of-the-art performance in inverse imaging tasks. During the process of SGD, low-frequency image features are reconstructed first, followed by higher frequencies, which makes human supervision necessary to retrieve the optimal denoised image. Therefore, this approach heavily relies on early stopping in order to not overfit the noise. However, a key advantage of deep image prior is the absence of hallucinations, since there is no prior learning. A Bayesian approach could alleviate overfitting and additionally provide reconstruction uncertainty.

Bayesian deep learning Bayesian neural networks allow estimation of predictive uncertainty (Bishop 2006) and we generally differentiate between aleatoric and epistemic uncertainty (Kendall and Gal 2017). Aleatoric uncertainty results from noise in the data (e. g. speckle noise in US or OCT). It is derived from the conditional log-likelihood under the maximum likelihood estimation (MLE) or maximum posterior (MAP) framework and can be captured directly by a deep network (i. e. by subdividing the last layer of an image generator network). Epistemic uncertainty is caused by uncertainty in the model parameters.

In deep learning, we usually perform MLE or MAP inference to find a single best estimate $\hat{\boldsymbol{\theta}}$ for the network parameters. This does not allow estimation of epistemic uncertainty and we therefore place distributions over the parameters. In Bayesian inference, we want to consider all possible parameter configurations, weighted by their posterior. Computing the posterior predictive distribution involves marginalization of the parameters $\boldsymbol{\theta}$, which is intractable. A common approximation of the posterior distribution is variational inference with Monte Carlo dropout (Gal and Ghahramani 2016b). It allows estimation of epistemic uncertainty by Monte Carlo sampling from the posterior of a network, that has been trained with dropout.

Bayesian deep image prior Cheng et al. recently provided a Bayesian perspective on the deep image prior in the context of natural images, which is most related to our work (Z. Cheng et al. 2019). They interpret the convolutional network as spatial random process over the image coordinate space and use stochastic gradient Langevin dynamics (SGLD) as Bayesian approximation (Welling and Teh 2011) to sample from the posterior. In SGLD, an MC sampler is derived from SGD by injecting Gaussian noise into the gradients after each backward pass. The authors claim to have solved the overfitting issue with DIP and to be able to provide uncertainty estimates. In the following, we will show that this is not the case for medical image denoising, even when using the code provided by the authors. Further, the uncertainty estimates from SGLD do not reflect the predictive error with respect to the noise-free ground truth image.

4.1.3 Methods

Aleatoric Uncertainty with Deep Image Prior

We first revisit the concept of deep image prior for denoising and subsequently extend it to a Bayesian approach with Monte Carlo dropout to estimate both aleatoric and epistemic uncertainty. Let $\tilde{\boldsymbol{x}}$ be a noisy image, $\boldsymbol{x}$ the true but generally unknown noise-free image and $\boldsymbol{f_\theta}$ an image generator network with parameter set $\boldsymbol{\theta}$, that outputs the denoised image $\hat{\boldsymbol{x}}$. In deep image prior, the optimal parameter point estimate $\hat{\boldsymbol{\theta}}$ is found by maximum likelihood estimation with gradient descent, which results in minimizing the squared error

$$\hat{\boldsymbol{\theta}} = \arg\min \|\tilde{\boldsymbol{x}} - \boldsymbol{f_\theta}(\boldsymbol{z})\|^2 \tag{4.2}$$

between the generated image $\boldsymbol{f_\theta}$ and the noisy image $\tilde{\boldsymbol{x}}$. The input $\boldsymbol{z} \sim \mathcal{U}(\mathbf{0}, 0.1\boldsymbol{I})$ of the neural network has the same spatial dimensions as $\tilde{\boldsymbol{x}}$ and is sampled from a uniform distribution. To ensure that $\hat{\boldsymbol{x}}$ has less noise, carefully chosen early stopping must be applied (see § 4.1.5).

To quantify aleatoric uncertainty, we assume that the image signal $\tilde{\boldsymbol{x}}$ is sampled from a spatial random process and that each pixel i follows a Gaussian distribution $\mathcal{N}(\tilde{x}_i; \hat{x}_i, \hat{\sigma}_i^2)$ with mean $\hat{x}_i$ and variance $\hat{\sigma}_i^2$. We split the last layer such that the network outputs these values for each pixel

$$\boldsymbol{f}_{\boldsymbol{\theta}} = \left[\hat{\boldsymbol{x}}, \hat{\boldsymbol{\sigma}}^2\right] \ . \tag{4.3}$$

Now, MLE is performed by minimizing the full negative log-likelihood, which leads to the following optimization criterion (Kendall and Gal 2017; Laves, Ihler, Fast, et al. 2020)

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^{N} \hat{\sigma}_i^{-2} \left\| \tilde{x}_i - \hat{x}_i \right\|^2 + \log \hat{\sigma}_i^2 \ , \tag{4.4}$$

where N is the number of pixels per image. Here, $\hat{\boldsymbol{\sigma}}^2$ captures the pixel-wise aleatoric uncertainty and is jointly estimated with $\hat{\boldsymbol{x}}$ by finding $\boldsymbol{\theta}$ that minimizes Eq. (4.4) with SGD. For numerical stability, Eq. (4.4) is implemented such that the network directly outputs $-\log \hat{\boldsymbol{\sigma}}^2$.

Epistemic Uncertainty with Bayesian Deep Image Prior

Next, we move towards a Bayesian view to additionally quantify the epistemic uncertainty. The image generator $\boldsymbol{f}_{\boldsymbol{\theta}}$ is extended into a Bayesian neural network under the variational inference framework with MC dropout (Gal and Ghahramani 2016b). A prior distribution $p(\boldsymbol{\theta}) \sim \mathcal{N}(\boldsymbol{0}, \lambda^{-1}\boldsymbol{I})$ is placed over the parameters and the network $\boldsymbol{f}_{\tilde{\boldsymbol{\theta}}}$ is trained with dropout by minimizing Eq. (4.4) with added weight decay. For inference, T stochastic forward passes with applied dropout are performed to sample from the approximate Bayesian posterior $\tilde{\boldsymbol{\theta}} \sim q(\boldsymbol{\theta})$. This allows us to approximate the posterior predictive distribution

$$p(\hat{\boldsymbol{x}}|\tilde{\boldsymbol{x}}) = \int p(\hat{\boldsymbol{x}}|\boldsymbol{\theta}, \tilde{\boldsymbol{x}}) p(\boldsymbol{\theta}|\tilde{\boldsymbol{x}}) \, \mathrm{d}\boldsymbol{\theta} \ , \tag{4.5}$$

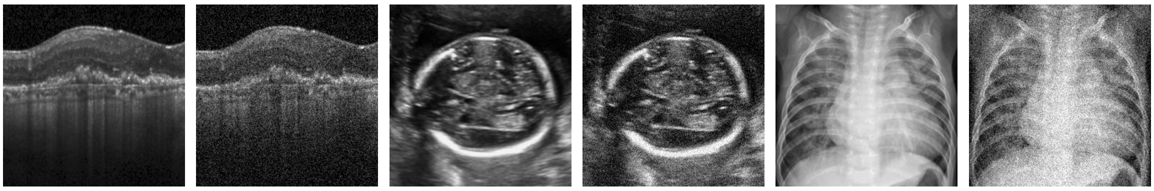
which is wider than the distribution from MLE or MAP, as it accounts for uncertainty in $\boldsymbol{\theta}$. We use Monte Carlo integration to estimate the predictive mean

$$\hat{\boldsymbol{x}} = \frac{1}{T} \sum_{t=1}^{T} \hat{\boldsymbol{x}}_t \tag{4.6}$$

and predictive variance (Kendall and Gal 2017; Laves, Ihler, Fast, et al. 2020)

$$\hat{\boldsymbol{\sigma}}^2 = \underbrace{\frac{1}{T} \sum_{t=1}^{T} \left(\hat{\boldsymbol{x}}_t - \frac{1}{T} \sum_{t=1}^{T} \hat{\boldsymbol{x}}_t \right)^2}_{\text{epistemic}} + \underbrace{\frac{1}{T} \sum_{t=1}^{T} \hat{\boldsymbol{\sigma}}_t^2}_{\text{aleatoric}} \tag{4.7}$$

with $\boldsymbol{f}_{\tilde{\boldsymbol{\theta}}_t} = [\hat{\boldsymbol{x}}_t, \hat{\boldsymbol{\sigma}}_t^2]$. In this work, we use $T = 25$ MC samples with dropout probability of $p = 0.3$. The resulting $\hat{\boldsymbol{x}}$ is used as estimation of the noise-free image and $\hat{\boldsymbol{\sigma}}^2$ is used as uncertainty map. We use the mean over the pixel coordinates as scalar uncertainty value U.



$\boldsymbol{x}_{\mathrm{OCT}}$ $\tilde{\boldsymbol{x}}_{\mathrm{OCT}}$ $\boldsymbol{x}_{\mathrm{US}}$ $\tilde{\boldsymbol{x}}_{\mathrm{US}}$ $\boldsymbol{x}_{\mathrm{xray}}$ $\tilde{\boldsymbol{x}}_{\mathrm{xray}}$

Figure 4.2: Images used to evaluate the denoising performance. The task is to reconstruct a noise-free image from $\tilde{\boldsymbol{x}}$ without having access to $\boldsymbol{x}$. OCT and US images are characterized by speckle noise which can be simulated by additive Gaussian noise. Low-dose X-ray shows uneven photon density that can be simulated by Poisson noise.

Calibration of Uncertainty

Following recent literature, we define predictive uncertainty to be well-calibrated if it correlates linearly with the predictive error (Guo et al. 2017; Laves, Ihler, Fast, et al. 2020; Levi et al. 2019). More formally, miscalibration is quantified with

$$\mathbb{E}_{\hat{\sigma}^2}\left[\left|\left(\|\tilde{\boldsymbol{x}}-\hat{\boldsymbol{x}}\|^2 \,\middle|\, \hat{\sigma}^2=\sigma^2\right)-\sigma^2\right|\right] \quad \forall\left\{\sigma^2\in\mathbb{R} \,|\, \sigma^2\geq 0\right\} . \tag{4.8}$$

That is, if all pixels in a batch were estimated with uncertainty of 0.2, we expect the predictive error (MSE) to also equal 0.2. To approximate Eq. (4.8) on an image with finite pixels, we use the uncertainty calibration error (UCE) metric presented in (Laves, Ihler, Fast, et al. 2020), which involves binning the uncertainty values and computing a weighted average of absolute differences between MSE and uncertainty per bin.

4.1.4 Experiments

We refer to the presented Bayesian approach to deep image prior with Monte Carlo dropout as MCDIP and evaluate its denoising performance and the calibration of uncertainty on three different medical imaging modalities (see Fig. 4.2). The first test image $\boldsymbol{x}_{\mathrm{OCT}}$ shows an OCT scan of a retina affected by choroidal neovascularization. Next, $\boldsymbol{x}_{\mathrm{US}}$ shows an ultrasound of a fetal head for gestational age estimation. The third test image $\boldsymbol{x}_{\mathrm{xray}}$ shows a chest x-ray for pneumonia assessment. All test images are arbitrarily sampled from public data sets (Heuvel et al. 2018; Kermany et al. 2018) and have a resolution of 512×512 pixel.

Images from optical coherence tomography and ultrasound are prone to speckle noise due to interference phenomena (Michailovich and Tannenbaum 2006). Speckle noise can obscure small anatomical details and reduce image contrast. It is worth mentioning that speckle patterns also contain information about the microstructure of the tissue. However, this information is not perceptible to a human observer, therefore, the denoising of such images is desirable. Noise in low-dose X-ray originates from an uneven photon density and can be modeled with Poisson noise (S. Lee et al. 2018; Žabić et al. 2013). In this work, we approximate the Poisson noise with Gaussian noise since Poisson(λ) approaches a Normal distribution as $\lambda \to \infty$ (see § 4.1.4). We first create a low-noise image $\boldsymbol{x}$ by smoothing and downsampling the original image from public data sets using the `ANTIALIAS` filter from the Python Imaging Library (`PIL`) to 256×256 pixel. Downsampling involves averaging over highly correlated neighboring pixels affected by uncorrelated noise. This decreases the observation noise by sacrificing image resolution (see next subsection). The downsampled image acts as ground truth to which we compute the peak signal-to-noise ratio (PSNR) and the structural similarity (SSIM) of the denoised image $\hat{\boldsymbol{x}}$. Further, we compute the UCE and provide calibration diagrams (MSE vs. uncertainty) to show the (mis-)calibration of the uncertainty estimates.

We compare the results from MCDIP to standard DIP and to DIP with SGLD from Cheng et al. (Z. Cheng et al. 2019). SGLD posterior inference is performed by averaging over T posterior samples $\hat{\boldsymbol{x}} = \frac{1}{T}\sum_{t=1}^{T} \hat{\boldsymbol{x}}_t$ after a Monte Carlo burn-in phase. The posterior variance is used as an estimator of the epistemic uncertainty $\frac{1}{T}\sum_{t=1}^{T} \left(\hat{\boldsymbol{x}} - \hat{\boldsymbol{x}}_t\right)^2$. Cheng et al. claim that their approach does not require early stopping and yields better denoising performance. Additionally, we train the SGLD approach with the loss function from Eq. (4.7) to consider aleatoric uncertainty and denote this with SGLD+NLL. We implement SGLD using the Adam optimizer, which works better in practice and is more related to preconditioned SGLD (Li et al. 2016).

Downsampling

Here, we provide justification why downsampling of an image by averaging neighboring pixels reduces the noise level and can be used as an approximation to a ground truth noise-free image (by sacrificing image resolution).

Proposition 2. *Downsampling of an image reduces the observation noise.*

Proof. Let $X = \mu_x + \varepsilon_x$ and $Y = \mu_y + \varepsilon_y$ be two neighboring pixels affected by additive i.i.d. noise $\varepsilon_x, \varepsilon_y \sim \mathcal{N}(0, \sigma^2)$. The pixels are assumed to be uncorrelated to noise. Pixels in a local neighborhood are highly correlated and assumed to be of high similarity $\mu_x \approx \mu_y = \mu$. Let $Z = \frac{1}{2}(X + Y)$ be the average of two neighboring pixels (i.e. the result of downsampling). The expectation is given by

$$\begin{aligned}
\mathbb{E}[Z] &= \frac{1}{2}\left(\mathbb{E}[X] + \mathbb{E}[Y]\right) && (4.9)\\
&= \frac{1}{2} 2\,\mathbb{E}[X] && (4.10)\\
&= \mu && (4.11)
\end{aligned}$$

and the variance is given by

$$\mathrm{Var}\left[Z\right] = \mathrm{Var}\left[\frac{1}{2}\left(X+Y\right)\right] \tag{4.12}$$

$$= \frac{1}{2^2}\left(\mathrm{Var}\left[X\right] + \mathrm{Var}\left[Y\right]\right) \tag{4.13}$$

$$= \frac{1}{2^2} 2\mathrm{Var}\left[X\right] \tag{4.14}$$

$$= \frac{1}{2}\sigma^2 \, . \tag{4.15}$$

Thus, if the similarity of neighboring pixels is sufficiently high, downsampling reduces the variance of average pixel Z by a factor of 2. ☐

Naturally, two neighboring pixels are not exactly equal. However, downsampling can also be viewed as superposing two signals, each with a highly correlated and an uncorrelated part. Without providing proof, the amplitude of the addition of two signals can be viewed as vector addition. In the uncorrelated case, the two signals are perpendicular to each other and in the correlated case, the angle between the two signals is acute. Thus, the correlated parts of the two signals have a higher impact on the resulting addition than the uncorrelated (noise) parts. In the ideal case, where the noise is uncorrelated and the signals are in parallel, the same noise reduction as above follows.

Link Between Poisson Distribution and Normal Distribution

We approximate the Poisson noise to simulate a low-dose X-ray image with a Normal distribution. It is well-known that the limiting distribution of $\mathsf{Poisson}(\lambda)$ is Normal as $\lambda \to \infty$ (Hogg et al. 2018). For completeness, we list a common proof using the moment generating function of a standardized Poisson random variable:

Theorem 2. *The $\mathsf{Poisson}(\lambda)$ distribution can be approximated with a Normal distribution as $\lambda \to \infty$.*

Proof. Let $X_\lambda \sim \mathsf{Poisson}(\lambda),\ \lambda \in \{1, 2, \ldots\}$. The probability mass function of X_λ is given by

$$f_{X_\lambda}(x) = \frac{\lambda^x e^{-\lambda}}{x!} \quad x \in \{0, 1, 2, \ldots\} \, . \tag{4.16}$$

The moment generating function is given by (Hogg et al. 2018)

$$M_{X_\lambda}(t) = \mathbb{E}[e^{tX_\lambda}] = e^{\lambda(e^t - 1)} \, . \tag{4.17}$$

The standardized Poisson random variable

$$Z = \frac{X_\lambda - \lambda}{\sqrt{\lambda}} \tag{4.18}$$

has the limiting moment generating function

$$\begin{aligned}
\lim_{\lambda\to\infty} M_Z(t) &= \lim_{\lambda\to\infty} \mathbb{E}\left[\exp\left(t\cdot\frac{X_\lambda-\lambda}{\sqrt{\lambda}}\right)\right] && (4.19)\\
&= \lim_{\lambda\to\infty} \exp\left(-t\sqrt{\lambda}\right)\mathbb{E}\left[\exp\left(\frac{tX_\lambda}{\sqrt{\lambda}}\right)\right] && (4.20)\\
&= \lim_{\lambda\to\infty} \exp\left(-t\sqrt{\lambda}\right)\exp\left(\lambda\left(e^{t/\sqrt{\lambda}}-1\right)\right) && (4.21)\\
&= \lim_{\lambda\to\infty} \exp\left(-t\sqrt{\lambda}+\lambda\left(t\lambda^{-1/2}+t^2\lambda^{-1}/2+t^3\lambda^{-3/2}/6+\ldots\right)\right) && (4.22)\\
&= \lim_{\lambda\to\infty} \exp\left(t^2/2+t^3\lambda^{-1/2}/6+\ldots\right) && (4.23)\\
&= \exp\left(t^2/2\right) && (4.24)
\end{aligned}$$

which is the moment generating function of a standard normal random variable. □

4.1.5 Results

The results are presented threefold: We show (1) possible overfitting in Fig. 4.3 by plotting the PSNR between the reconstruction $\hat{\boldsymbol{x}}$ and the ground truth image $\boldsymbol{x}$; (2) denoising performance by providing the denoised images in Fig. 4.4 and PSNR in Tab. 4.1 after convergence (i. e. after 50k optimizer steps); and (3) goodness of uncertainty in Fig. 4.5 by providing calibration diagrams and uncertainty maps.

Our experiments confirm what is already known: The non-Bayesian DIP quickly overfits the noisy image. The narrow peaks in PSNR values during optimization show that manually performed early stopping is essential to obtain a reconstructed image with less noise (see Fig. 4.3). The PSNR between $\hat{\boldsymbol{x}}$ and the ground truth $\boldsymbol{x}$ approaches the value of the PSNR between the noisy image $\tilde{\boldsymbol{x}}$ and the ground truth, thus reconstructing the noise as well. However, the SGLD approach shows almost identical overfitting behavior in our experiments. This is in contrast to what is stated by Chen et al., even when using the original implementation of SGLD provided by the authors (Z. Cheng et al. 2019). SGLD+NLL additionally considers aleatoric uncertainty and converges to a higher PSNR level. This indicates that SGLD+NLL does not overfit the noisy image completely. MCDIP on the other hand does not show a sharp peak in Fig. 4.3 and safely converges to its highest PSNR value. This requires no manual early stopping to obtain a denoised image. The reconstructed X-ray images after convergence in Fig. 4.4 underline this: MCDIP does not reconstruct the noise. The PSNR values in Tab. 4.1 confirm these observations. Although it was not the intention of this work to reach highest-possible PSNR values, MCDIP even outperforms the other methods with early-stopping applied (see Tab 4.2).

The calibration diagrams and corresponding UCE in Fig. 4.5 suggest that SGLD+NLL is better calibrated than MCDIP. However, due to overfitting the noisy image without early stopping, the MSE from SGLD+NLL concentrates around 0.0, which results in low UCE

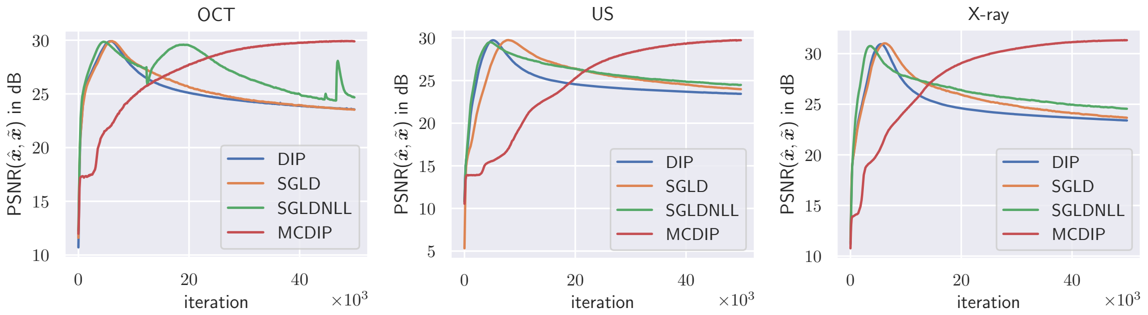
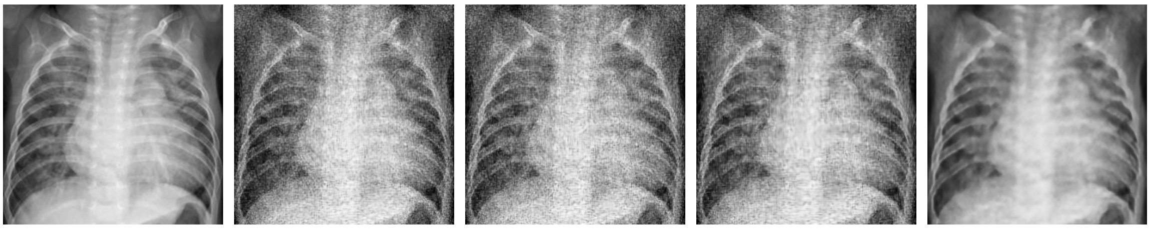


Figure 4.3: Peak signal-to-noise ratio between denoised image $\hat{\boldsymbol{x}}$ and ground truth $\boldsymbol{x}$ vs. number of optimizer iterations. DIP and SGLD(+NLL) quickly overfit the noisy image. MCDIP converges to its highest PSNR value and does not overfit $\tilde{\boldsymbol{x}}$. The plots show means from 3 runs with different random initialization.



ground truth DIP SGLD SGLD+NLL MCDIP

Figure 4.4: Denoised X-ray images after convergence. Only MCDIP does not show overfitted noise. Additional reconstructions can be found in Appendix A.3.1.

values. On the US and OCT image, the uncertainty from SGLD+NLL collapses to a single bin in the calibration diagram and does not allow to reason about the validness of the reconstructed image (see Fig. 4.5). The uncertainty map from MCDIP shows high uncertainty at edges in the image and the mean uncertainty value (denoted by U) is close to the noise level in all three test images.

SGLD With Step Size Decay

Additionally, we implement SGLD with step size decay as described by Welling and Teh (2011). The step size ϵ is used to scale the parameter update in the SGD step (i.e. the learning rate) and defines the variance of the noise injected into the gradients. Here, we reduce the step size at each step t exponentially with $\epsilon_t = 0.999^t \epsilon_0$. To satisfy the step size property (Eq. (2) in (Welling and Teh 2011)), we fix the step size once it decreases below 1e-8. We observe no overfitting of the noisy image with step size decay (see Fig. 4.6). However, the quality of the resulting denoised image is very sensitive to the decay scheme. A decrease that is too low (i.e. $\epsilon_t = 0.9999^t \epsilon_0$) results in overfitting; a decrease that is too high (i.e. $\epsilon_t = 0.99^t \epsilon_0$) results in convergence to a subpar reconstruction. This is equivalent to carefully applied early stopping and therefore nullifies the advantage of SGLD for denoising of medical images.

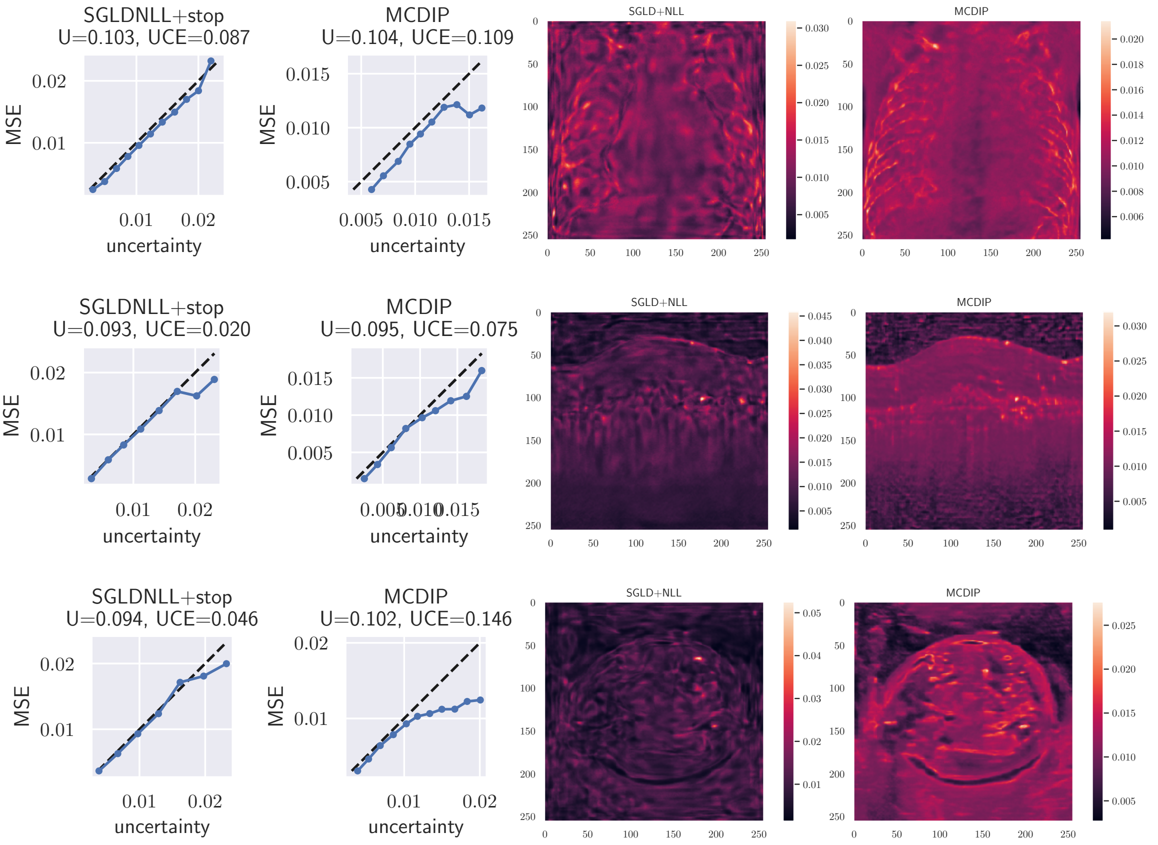


Figure 4.5: Calibration diagrams and uncertainty maps for SGLD+NLL after early stopping and MCDIP after convergence (best viewed with digital zoom). (Left) The calibration diagrams show MSE vs. uncertainty and provide mean uncertainty (U) and UCE values. (Right) Uncertainty maps show per-pixel uncertainty.

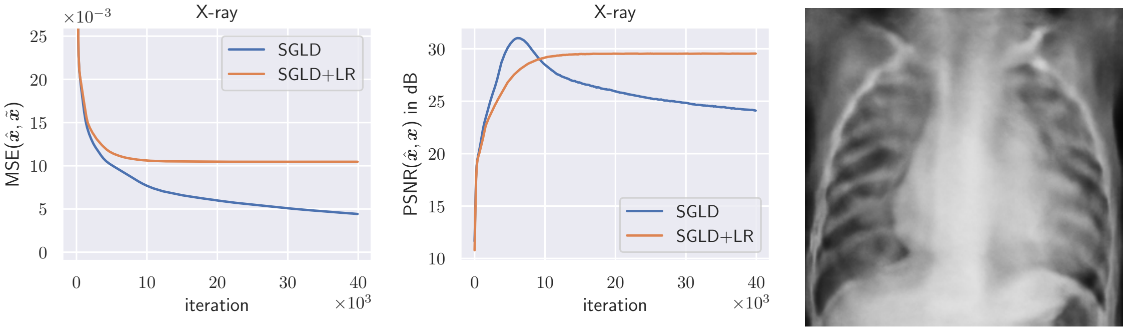


Figure 4.6: Comparison of SGLD and SGLD+LR (with step size decay). Carefully chosen step size decay impedes overfitting the noisy image. (Right) Reconstruction of SGLD+LR after convergence (no early stopping applied).

Table 4.1: PSNR values after convergence (at least 50k iterations). Note that our goal was not to reach highest possible PSNR, but to show overfitting in convergence.

PSNR	DIP	SGLD	SGLD+NLL	MCDIP (ours)
OCT	23.64 ± 0.19	23.58 ± 0.12	24.82 ± 0.12	$\mathbf{29.88} \pm 0.03$
US	23.55 ± 0.11	23.81 ± 0.15	24.55 ± 0.08	$\mathbf{29.67} \pm 0.07$
X-ray	23.28 ± 0.08	23.50 ± 0.12	24.60 ± 0.04	$\mathbf{31.19} \pm 0.10$

Table 4.2: PSNR with early-stopping.

PSNR	DIP	SGLD	SGLD+NLL	MCDIP (ours)
OCT	29.88 ± 0.02	29.89 ± 0.05	29.77 ± 0.07	$\mathbf{29.92} \pm 0.03$
US	29.74 ± 0.05	$\mathbf{29.78} \pm 0.02$	29.54 ± 0.03	29.70 ± 0.07
X-ray	30.91 ± 0.05	30.98 ± 0.09	30.74 ± 0.03	$\mathbf{31.22} \pm 0.1$

4.1.6 Conclusion

In this paper, we provided a new Bayesian approach to the deep image prior. We used variational inference with Monte Carlo dropout and the full negative log-likelihood to both quantify epistemic and aleatoric uncertainty. The presented approach is applied to medical image denoising of three different modalities and provides state-of-the-art performance in denoising with deep image prior. Hallucinations in denoising are made impossible compared to other deep learning methods, as the neural network only has access to one single image at all time. Our Bayesian treatment does not need carefully applied early stopping and yields well-calibrated uncertainty. We observe the estimated mean uncertainty value to be close to the noise level of the images.

The question remains why Bayesian deep image prior with SGLD does not work as well as expected and is outperformed by MC dropout. First, SGLD as described by Welling et al. requires a strong decay of the step size to ensure convergence to a mode of the posterior (Welling and Teh 2011). Cheng et al. did not implement this and we followed their approach (Z. Cheng et al. 2019). After implementing the described step size decay, SGLD did not overfit the noisy image (see § 4.1.5). However, this requires a carefully chosen step size decay which is equivalent to early stopping.

The deep image prior framework is especially interesting in medical imaging as it does not require supervised training and thus does not suffer from hallucinations and other artifacts. The presented approach can further be applied to deformable registration or other inverse image tasks in the medical domain.

4.2 Deformable Medical Image Registration Using a Randomly-Initialized CNN as Regularization Prior

In this section, we present deformable unsupervised medical image registration using a randomly-initialized deep convolutional neural network (CNN) as regularization prior. Conventional registration methods predict a transformation by minimizing dissimilarities between an image pair. The minimization is usually regularized with manually engineered priors, which limits the potential of the registration. By learning transformation priors from a large dataset, CNNs have achieved great success in deformable registration. However, learned methods are restricted to domain-specific data and the required amounts of medical data are difficult to obtain. Our approach uses the idea of deep image priors to combine convolutional networks with conventional registration methods based on manually engineered priors. The proposed method is applied to brain MRI scans. We show that our approach registers image pairs with state-of-the-art accuracy by providing dense, pixel-wise correspondence maps. It does not rely on prior training and is therefore not limited to a specific image domain.

4.2.1 Introduction

Deformable registration is a major challenge in medical image processing. The result is a dense mapping showing pixel-wise non-linear correspondences between a pair of images that best aligns the input image $\boldsymbol{I}$ onto the target image $\boldsymbol{T}$ by means of some similarity definition $\mathcal{L}$. Deformable registration is applied in the analysis of patient-specific temporal or anatomical changes, e.g., from pre-operative to post-operative state, or to show inter-patient variances (Sotiras et al. 2013). Deformable registration is also performed in atlas-based segmentation, where an input image is matched onto a target image with known segmentation (Cabezas et al. 2011).

Existing registration methods can be separated into two categories. The first category is based on non-learning methods which estimate a registration $\boldsymbol{w}$ by optimizing a cost function of the form

$$\arg\min_{\boldsymbol{w}} \left\{ \mathcal{L}(\boldsymbol{T}, \boldsymbol{w} \circ \boldsymbol{I}) + \lambda \mathcal{R}(\boldsymbol{w}) \right\}, \tag{4.25}$$

where $\boldsymbol{w} \circ \boldsymbol{I}$ denotes $\boldsymbol{I}$ warped by $\boldsymbol{w}$. As registration is an inverse problem, Eq. (4.25) is closely related to Eq. (4.1) from the previous section. A common assumption of $\boldsymbol{w}$ is a displacement or velocity vector field $\boldsymbol{u}(\boldsymbol{x})$. The final deformation results in $\boldsymbol{\phi}(\boldsymbol{x}) = \boldsymbol{x} + \boldsymbol{u}(\boldsymbol{x})$ which maps every pixel coordinate $\boldsymbol{x}$ to other pixel coordinates. The first term in (4.25) is referred to as data term, which is typically chosen to be a pixel intensity error measure. Optimization of the data term alone is considered ill-posed. The second term $\mathcal{R}$, weighted by trade-off factor λ, is a regularizer that shapes the registration by any chosen prior, which helps solving the ill-posed problem. Common regularization is done by enforcing smoothness onto the displacement vector field by penalizing first or higher order spatial derivatives of $\boldsymbol{u}$ (Werlberger et al. 2010). The result of the registration algorithm heavily depends on the cost function and therefore on the chosen prior of $\mathcal{R}$.

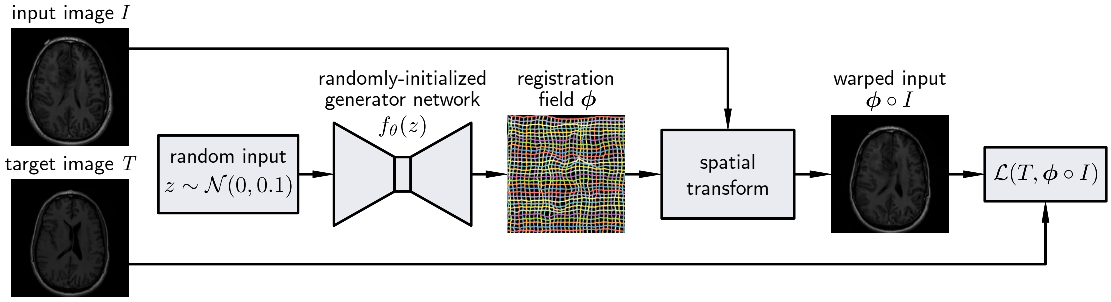


Figure 4.7: Overview of our method. The randomly-initialized generator network $\boldsymbol{f_\theta}$ acts as parameterization of the registration field $\boldsymbol{\phi}$. The parameters $\boldsymbol{\theta}$ are optimized for every image pair individually by gradient descent.

The second category implicitly learns the regularization prior by training a convolutional network on a large database of domain-specific images. Early approaches rely on ground truth registrations (Sokooti et al. 2017), which are hard to obtain especially in medical imaging. More recent methods (Balakrishnan et al. 2019) propose unsupervised registration using the spatial transformer function (Jaderberg et al. 2015). However, these methods either only support small displacements or require segmentation maps of the image pairs during training to assist the convergence (Hu et al. 2018). Additionally, the trained networks are limited to register images from the training domain (e.g., CT or MRI).

Inspired by the idea of deep image priors (Lempitsky et al. 2018), we subsequently propose our learning-free method for deformable medical image registration using the structure of an untrained convolutional network as regularization prior.

4.2.2 Methods

Lempitsky et al. have recently shown that excellent performance of CNNs for inverse image problems, such as denoising, is not only based on their ability to learn image priors from data, but is also based on the structure of a convolutional image generator itself Lempitsky et al. 2018. They gave evidence that the structure of a network alone is sufficient to capture enough image statistics to provide state-of-the-art performance in inverse image tasks.

Leveraged by this idea, we reformulate the task of deformable image registration by using the structure of a convolutional network as regularizer (see Fig. 4.7). An image generator network $\boldsymbol{u} = \boldsymbol{f_\theta}(\boldsymbol{z})$ with randomly-initialized parameters $\boldsymbol{\theta}$ is interpreted as parameterization of the dense displacement field $\boldsymbol{u} \in \mathbb{R}^{2 \times H \times W}$ from which the deformation $\boldsymbol{\phi} = \boldsymbol{x} + \boldsymbol{u}$ between an input image $\boldsymbol{I} \in \mathbb{R}^{C \times H \times W}$ and a target image $\boldsymbol{T} \in \mathbb{R}^{C \times H \times W}$ can be obtained by adding to the identity warp $\boldsymbol{x}$. The input $\boldsymbol{z} \in \mathbb{R}^{C' \times H \times W} \sim \mathcal{N}(\boldsymbol{0}, 0.1\boldsymbol{I})$ has the same spatial dimensions as $\boldsymbol{\phi}$ and is sampled from a random normal distribution in every iteration. This leads to the following optimization problem

$$\arg\min_{\theta} \left\{ \mathcal{L}\left(\boldsymbol{T}, (\boldsymbol{x} + \boldsymbol{f_\theta}(\boldsymbol{z})) \circ \boldsymbol{I}\right) \right\}, \tag{4.26}$$

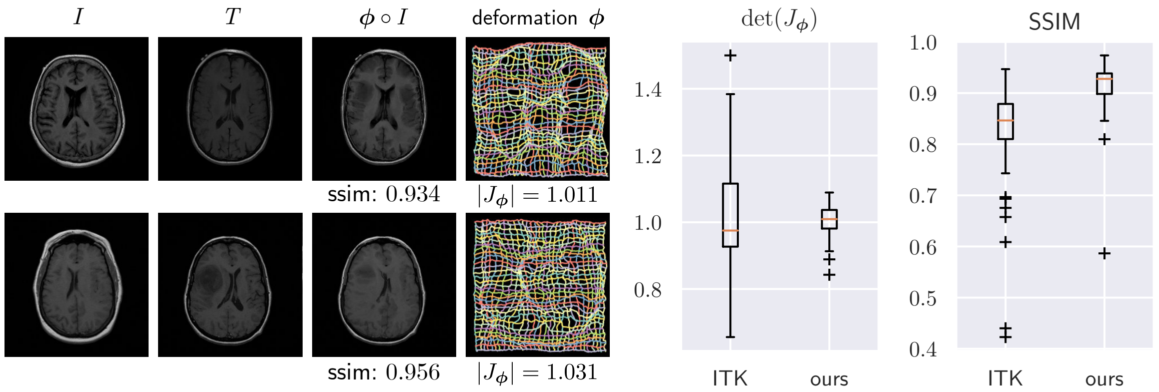


Figure 4.8: Results of our approach compared to a state-of-the-art method from the Insight ToolKit (ITK). Left: Two example MRI pairs from the data set. Right: Boxplots of the means of $\det(\boldsymbol{J}_\phi)$ and SSIM between $\boldsymbol{T}$ and $\boldsymbol{\phi} \circ \boldsymbol{I}$.

where $(\boldsymbol{x} + \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{z})) \circ \boldsymbol{I}$ denotes the differentiable spatial transformer function (Jaderberg et al. 2015). Eq. (4.26) is optimized for every image pair $\{\boldsymbol{I}, \boldsymbol{T}\}$ using the Adam gradient descent optimizer (D. P. Kingma and Ba 2014). As data term, we chose pixel-wise mean absolute error $\mathcal{L}(\boldsymbol{T}, \boldsymbol{\phi} \circ \boldsymbol{I}) = |(\boldsymbol{\phi} \circ \boldsymbol{I}) - \boldsymbol{T}|$. The architecture of the image generator network $\boldsymbol{f}_{\boldsymbol{\theta}}$ is chosen according to (Lempitsky et al. 2018). It has an encoder-decoder structure with skip connections between the encoding and decoding part. To begin the optimization from close to an identity warp, we initialize the parameters with $\theta_i \sim \mathcal{N}(0, 0.01)$.

4.3 Results & Conclusion

We demonstrate our approach on the task of 2D brain magnetic resonance imaging (MRI) registration. The data used in this work contain 109 pairs of MRI scans from The Cancer Genome Atlas (TCGA 2019) showing lower-grade gliomas. The resulting displacement field is used to warp the pathological images onto the healthy images. We use the structural similarity index (SSIM) Z. Wang et al. 2004 between $\boldsymbol{\phi} \circ \boldsymbol{I}$ and $\boldsymbol{T}$ and the mean of the determinants of Jacobians $\det(\boldsymbol{J}_\phi)$ (Ashburner 2007) of the deformation as evaluation metrics. The latter metric shows regularity of $\boldsymbol{\phi}$. We compare our method to state-of-the-art methods from the Insight ToolKit (ITK) registration framework by combining an initial affine registration and a subsequent deformable displacement field registration (Avants et al. 2012). Results for exemplary image pairs and boxplots of results for all image pairs are shown in Fig. 4.8. Additional results including registration fields are shown in Figure 4.9.

The results reveal that the structure of a convolutional network can act as regularization in deformable medical image registration with state-of-the-art performance. This connects traditional non-learning methods and learning-based methods by using randomly-initialized convolutional networks as prior.

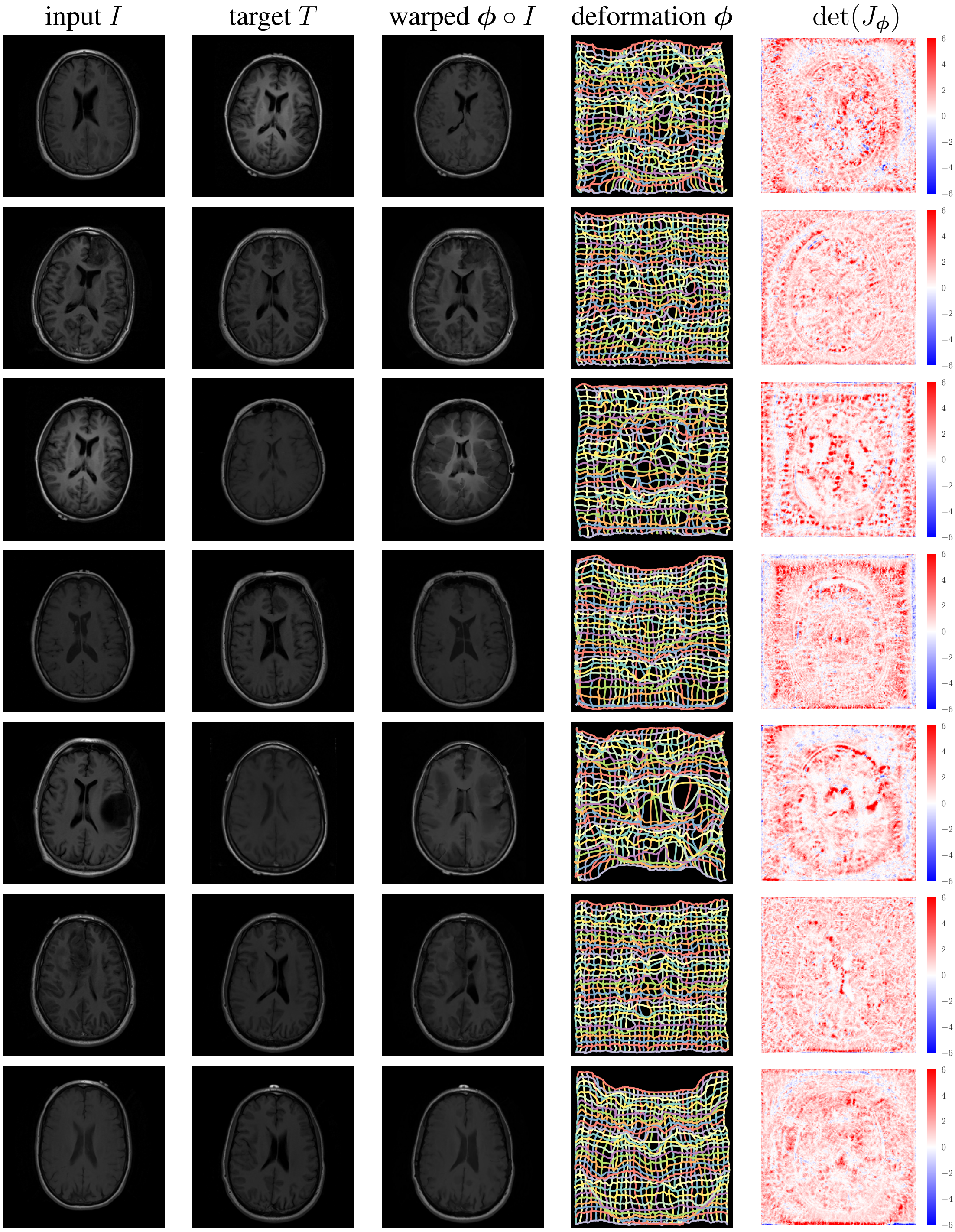


Figure 4.9: Additional results showing image pairs $\{\boldsymbol{I}, \boldsymbol{T}\}$, warped input $\boldsymbol{\phi} \circ \boldsymbol{I}$, estimated deformation grid $\boldsymbol{\phi}$ and map of determinants of the Jacobian matrix $\boldsymbol{J}_{\boldsymbol{\phi}} = \nabla\boldsymbol{\phi}$ for every entry of $\boldsymbol{\phi}$. $\boldsymbol{J}_{\boldsymbol{\phi}}$ shows local regularity of the deformation field. The deformation is diffeomorphic, where $\det(\boldsymbol{J}_{\boldsymbol{\phi}}) > 0$.

4.3.1 Semantically Regularized Denoising Autoencoders

In this section, a supervised denoising approach that preserves disease characteristics on retinal optical coherence tomography images is presented. We propose *semantic denoising autoencoders*, which combine a convolutional denoising autoencoder with a priorly trained ResNet image classifier as regularizer during training. This promotes the perceptibility of delicate details in the denoised images that are important for diagnosis and filters out only informationless background noise. With our approach, higher peak signal-to-noise ratios with PSNR = 31.0 dB and higher classification performance of $F_1 = 0.92$ can be achieved for denoised images compared to state-of-the-art denoising. We show that semantically regularized autoencoders are capable of denoising retinal OCT images without blurring details of diseases. Here, "semantically" means that the denoising network is regularized considering the medical content of the image. The third part of this chapter was published at the "European Conference on Biomedical Optics" 2019 and presented as oral (Laves, Ihler, Kahrs, et al. 2019b).

4.3.2 Purpose

Optical coherence tomography is the most common imaging technique for diagnosis in ophthalmology. However, due to image acquisition based on interference of coherent light, OCT suffers from speckle noise. This results in grainy images with low contrast and obscured features where the diagnosis of medical conditions requires trained expert observers. Denoising of OCT has been addressed in the literature already and can be separated into two categories (Salinas and Fernandez 2007). The first one employs denoising during OCT acquisition by e.g. averaging multiple frames of the same object. This prolongs the acquisition process and is therefore not applicable for dynamic objects. The second category comprises post-processing methods as inverse image problems, which try to reconstruct a clean image $\hat{\boldsymbol{x}}$ from a noisy observation $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{c}$. A common assumption of the noise model $\boldsymbol{c}$ of the observation $\tilde{\boldsymbol{x}}$ in OCT imaging is additive white Gaussian noise with zero mean and standard deviation σ (Salinas and Fernandez 2007; K. Zhang et al. 2017).

Given a noisy OCT observation $\tilde{\boldsymbol{x}}$, the denoising can be expressed as optimization problem of the form (cf. Eq. (4.1), (4.25))

$$\hat{\boldsymbol{x}} = \arg\min_{\hat{\boldsymbol{x}}} \left\{ \mathcal{L}(\tilde{\boldsymbol{x}}, \hat{\boldsymbol{x}}) + \lambda \mathcal{R}(\hat{\boldsymbol{x}}) \right\} , \tag{4.27}$$

which tries to find a reconstruction $\hat{\boldsymbol{x}}$ that is close to $\tilde{\boldsymbol{x}}$ by means of some similarity measure $\mathcal{L}$, but has considerably less noise. The term $\mathcal{R}$, weighted by a trade-off factor λ, regularizes the optimization of (4.27) in order to impose the condition of $\hat{\boldsymbol{x}}$ having less noise than $\tilde{\boldsymbol{x}}$. The regularizer $\mathcal{R}$ generally expresses a chosen prior on the denoised images, such as the total variation (TV) (Chambolle 2004), or first and higher order derivatives of the image. In recent years, autoencoders (AE) have been applied to denoising tasks, in which the regularization prior is learned from corrupted and uncorrupted data samples $\{\boldsymbol{x}, \tilde{\boldsymbol{x}}\}$ (Bengio et al. 2013;

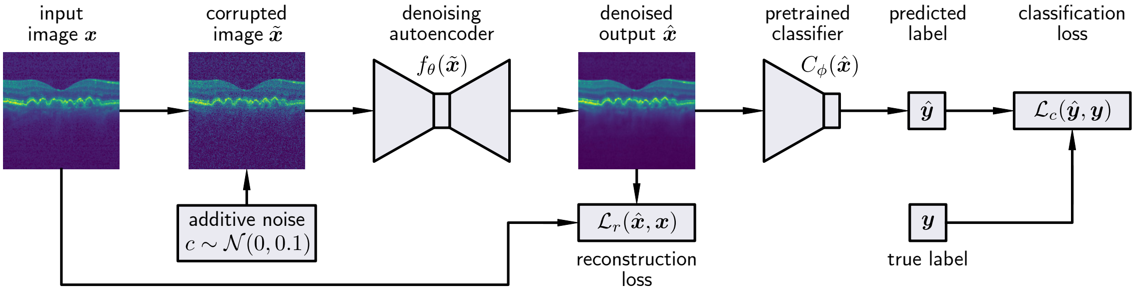

input image $\boldsymbol{x}$
corrupted image $\tilde{\boldsymbol{x}}$
denoising autoencoder
$f_\theta(\tilde{\boldsymbol{x}})$
denoised output $\hat{\boldsymbol{x}}$
pretrained classifier
$C_\phi(\hat{\boldsymbol{x}})$
predicted label
$\hat{\boldsymbol{y}}$
classification loss
$\mathcal{L}_c(\hat{\boldsymbol{y}}, \boldsymbol{y})$
additive noise $c \sim \mathcal{N}(0, 0.1)$
$\mathcal{L}_r(\hat{\boldsymbol{x}}, \boldsymbol{x})$
reconstruction loss
$\boldsymbol{y}$
true label

Figure 4.10: Overview of sDAE training procedure. The autoencoder f_θ tries to reconstruct a clean image $\hat{\boldsymbol{x}}$ from a noisy observation $\tilde{\boldsymbol{x}}$ while being regulated by a pretrained classifier C_ϕ.

K. Zhang et al. 2017). The performance of AEs for denoising is not only due to their ability to learn priors from data, but also due to the structure of the image generator itself (Lempitsky et al. 2018). Denoising autoencoders (DAE) usually have a data bottleneck between the encoding part and the decoding part, which forces the encoding part to extract a meaningful low-dimensional latent representation from a corrupted input image $\tilde{\boldsymbol{x}}$. This is then fed into the decoding part and mapped back to a reconstructed image $\hat{\boldsymbol{x}}$ in input space. Although DAEs provide excellent performance in denoising, they suffer from smoothing out subtle details that are important for medical diagnosis.

AEs have recently been used to regularize the training of diagnostic classifiers in medical imaging (Creswell et al. 2018; Laves, Ihler, Kahrs, et al. 2019a). However, the opposing approach where a diagnostic classifier regularizes the process of DAE has not been addressed so far. Therefore, this paper describes a domain-specific post-processing method for denoising medical images with preservation of delicate disease characteristics by proposing *semantic denoising autoencoders* (sDAE).

4.3.3 Methods

In this section, the sDAE approach is presented in detail. First, a ResNet-34 image classifier (He, X. Zhang, et al. 2016) C_ϕ pretrained on ImageNet is fine-tuned on a dataset of OCT images described below. This acts as medical expert as it has been shown that the performance of convolutional neural networks (CNNs) in classifying retinal conditions is on par to that of trained ophthalmologists (Kermany et al. 2018). Second, the ErfNet CNN autoencoder (Romera et al. 2018) f_θ is trained to reconstruct input images $\boldsymbol{x}$ corrupted by additive gaussian white noise resulting in $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{c}$ with $\boldsymbol{c} \sim \mathcal{N}(\mathbf{0}, 0.1\,\mathbf{I})$. The parameters θ of the AE are optimized by minimizing the pixel-wise mean squared reconstruction error $\mathcal{L}_r(f_\theta(\tilde{\boldsymbol{x}}), \boldsymbol{x})$. Essentially, an autoencoder learns a low-dimensional representation similar to principal component analysis (PCA). When training with a large dataset, noise tends to “average out” and the AE reconstructs distinct and relevant (noise-free) image features. In order to promote enhancement of these features, the trained ResNet with fixed weights ϕ is used as additional

	original	corrupted	TV	wavelet	AD	DAE	sDAE (ours)
PSNR	∞	20.9 ± 0.24	29.3 ± 1.2	28.0 ± 1.0	28.2 ± 1.3	**31.4** ± 1.78	31.1 ± 1.65
SSIM	1.0	0.44 ± 0.03	0.85 ± 0.05	0.81 ± 0.05	0.83 ± 0.05	**0.89** ± 0.04	**0.89** ± 0.04
F_1	0.94	0.89	0.79	0.83	0.55	0.86	**0.92**

Table 4.3: Mean results of denoising reported for the test set with mean peak signal-to-noise ratio (PSNR) in dB, structural similarity index (SSIM) and mean classification F_1 scores. Values for uncorrupted $\boldsymbol{x}$ and corrupted images $\tilde{\boldsymbol{x}}$ are given for comparison. Bold values denote best results.

optimization criterion $\mathcal{L}_c$. It is applied to the reconstructed, denoised image and tries to predict the retinal disease class (see Fig. 4.10). This regularizes the AE during training and enhances disease characteristics in denoised images. The proposed approach is therefore optimized using the weighted loss function

$$\arg\min_{\theta} \left\{ \mathcal{L}_r(f_\theta(\tilde{\boldsymbol{x}}), \boldsymbol{x}) + \lambda \mathcal{L}_c(C_\phi(f_\theta(\tilde{\boldsymbol{x}})), \boldsymbol{y}) \right\} \tag{4.28}$$

with true disease label $\boldsymbol{y}$ of image $\boldsymbol{x}$ and cross entropy loss for $\mathcal{L}_c$. The denoised reconstruction $\hat{\boldsymbol{x}} = f_\theta(\tilde{\boldsymbol{x}})$ can be obtained after convergence of the training. Regularization factor for $\mathcal{L}_c$ was empirically set to $\lambda = 0.01$.

The dataset used to train the sDAE consists of 84,484 retinal OCT images from 4,657 patients showing the disease states drusen, diabetic macular edema (DME), choroidal neovascularization (CNV) and normal and is publicly available (Kermany et al. 2018). We split the dataset using 4,000 OCT scans (1,000 from each class) for validation during training and another 4,000 scans for reporting final results (test set). The images for validation and testing were extracted patient by patient in order to prevent that data from one patient is included in more than one of the partial datasets.

The aforementioned method is implemented with PyTorch 1.5 and trained for 100 epochs using the Adam optimizer with an initial learning rate of $\eta = 10^{-4}$ (D. P. Kingma and Ba 2014). A reduce-on-plateau learning rate scheduling is realized to reduce η with a factor of 10^{-1} when observing saturation of the training loss. The weight configuration with lowest loss value on the validation set is chosen for testing (early stopping).

4.3.4 Results

To assess denoising performance, the proposed method is compared to total variation (TV) minimization (Chambolle 2004), BayesShrink wavelet denoising (Chang et al. 2000), anisotropic diffusion (AD) denoising (Perona and Malik 1990) and an unregularized DAE regarding peak signal-to-noise ratio (PSNR), structural similarity index (Z. Wang et al. 2004) and classification performance of ResNet using the F_1 score. The DAE can be seen as a

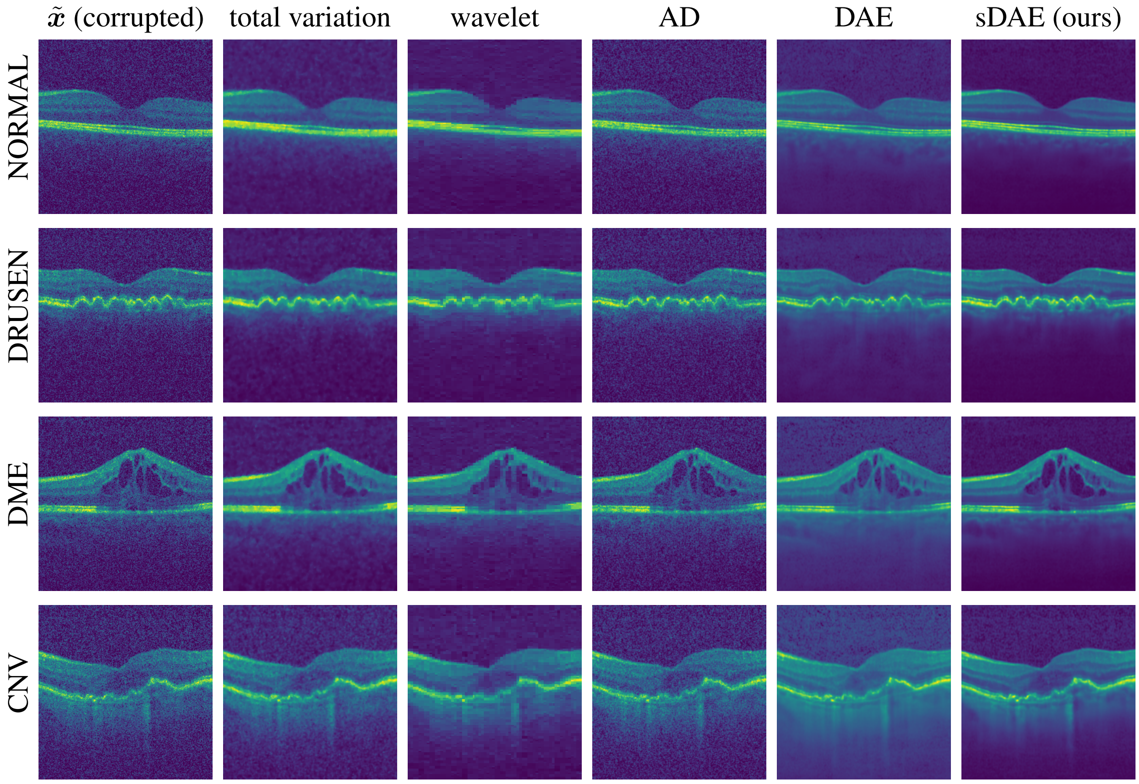


Figure 4.11: Results of our approach compared to state-of-the-art denoising for retinal OCT disease conditions from the test set. Digital zoom is recommended for optimal comparison.

special case of our approach, where $\lambda = 0$, such that it is only trained for reconstruction. The results are summarized in Tab. 4.3. We additionally provide the results of uncorrupted $\boldsymbol{x}$ and corrupted images $\tilde{\boldsymbol{x}}$ as baseline. Our approach not only provides the highest disease classification accuracy with $F_1 = 0.92$ after denoising, but also has a peak signal-to-noise ration with $\mathrm{PSNR} = 31.1\,\mathrm{dB}$, which is only exceeded by the DAE. However, the SSIM suggests that sDAE and DAE have similar reconstruction performance.

Fig. 4.11 visualizes qualitative results for example OCT scans from the test set showing different disease conditions. The methods are used to restore the input image $\boldsymbol{x}$ from the corrupted image $\tilde{\boldsymbol{x}}$ (first column). In contrast to state-of-the-art denoising, our approach is able to distinctively preserve the retinal layers while removing speckle noise. Pathological alterations of the retina are clearly visible and the explanatory power for diagnosis is not reduced. Mean processing time of sDAE for one image is 13.1 ms on an NVIDIA GeForce GTX 1080 Ti.

4.3.5 Conclusion

It has been shown that the proposed semantic denoising autoencoder is capable of denoising retinal OCT images without suppressing characteristics of diseases. This was achieved by regularizing the denoising autoencoder during training with another CNN, which was previously trained for disease classification. The denoising performance of sDAE is similar to that of an unregularized autoencoder, but sDAE preserves details important for diagnosis. The trained decoder can also be used to generate new images by sampling the latent space. Future work therefore aims on variational autoencoder and generative adversarial networks for OCT denoising. It should be noted, however, that speckle noise can also contain significant information as it creates a unique fingerprint of tissue. This information is hard to be interpreted by humans, and CNNs can be valuable tools to acquire and utilize this information in the future. The presented approach can also be translated to other inverse image problems such as single image super-resolution or compression artifacts removal or other medical imaging modalities such as computed tomography or magnetic resonance imaging.

4.4 Chapter Conclusion

In this chapter, we first presented MCDIP, a novel Bayesian approach to the concept of deep image prior with Monte Carlo dropout (§ 4.1). MCDIP alleviates the overfitting disadvantages of deep image prior, but keeps its robustness to hallucinations; a failure of deep generative models that must be avoided at all costs in the context of medical imaging. In our experiments, the denoising performance was on par to state-of-the-art methods and yielded well-calibrated pixel-wise uncertainty estimates.

Moreover, we applied deep image prior in a non-Bayesian fashion to deformable registration. This approach provides an implicit regularization of the deformation field and outperformed other non-learning based methods in terms of registration accuracy and smoothness of deformation. In future work, we plan to extend this to a Bayesian approach, which is expected to further provide diffeomorphic deformations and exhibit high uncertainty, where the deformation happens to be non-diffeomorphic.

Finally, we investigated a different approach to medical image denoising and presented semantically regularized autoencoders. A pre-trained image classifier was used as a regularizer during training of a denoising autoencoder. We hypothesized that this would preserve disease-related image features in the denoised image, which was confirmed by the considerably higher classification accuracy on the denoised images compared to other methods.

5 Conclusion and Perspectives

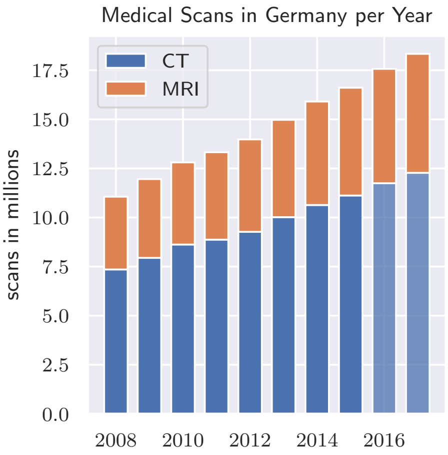


Figure 5.1: CT and MRI scans per year in Germany. Shaded bars are predicted (Statista 2019).

Medical imaging has revolutionized medicine in the last century. It has helped to literally give insight into the human anatomy and physiology. Many diseases and pathologies can only be diagnosed with the use of some imaging technique. In 2017, almost 18 million CT and MRI scans were expectedly performed in Germany (see Fig. 5.1). Due to increasing availability and the reduction of costs, the number of medical imaging examinations is continuously growing, resulting in a huge amount of data that has to be assessed by medical experts. Computer-aided diagnosis aims at automating the process of image-based diagnosis with the use of digital image analysis; its beginning dates back to the early 1980s (Doi 2007). However, only the recent advancement of deep learning has made CAD feasible at large scale (Esteva et al. 2017; Kermany et al. 2018). The biggest disadvantage of deep learning methods in practice is their black-box nature. Even though they achieve the highest levels of accuracy in diagnosis, their acceptance may be limited by their lack of interpretability and transparency. These concerns are reinforced by the core problem that is addressed in this thesis: the overconfidence of deep models when making false predictions. How do we know when we do not know?

This thesis deals with Bayesian methods for estimation of predictive uncertainty in medical imaging with deep learning. We have shown that predictive uncertainty from variational Bayesian inference is prone to miscalibration and does not represent the model error. The *uncertainty calibration error* was proposed to alleviate disadvantages of existing metrics to measure miscalibration. We defined perfect calibration of uncertainty using the normalized entropy and proved that the normalized entropy approaches the top-1 error for perfectly calibrated models in multi-class classification. *Logit scaling* for Monte Carlo dropout was derived and used for post-hoc calibration. UCE can additionally be used as regularization during training to improve calibration. In our experiments, it outperformed the commonly used entropy regularization. We have shown empirically that well-calibrated models are capable of rejecting unreliable predictions and detecting out-of-distribution data; an adversary that is likely to be encountered in clinical practice. This confirmed hypothesis 1.

The creation of large labeled medical data sets for machine learning is associated with high costs. To reduce the economic burden of data labeling, we have shown that well-calibrated uncertainty can leverage self-supervised learning by generating good pseudo-labels. Confidence based label generation can create an overfocus to certain classes and results in ignoring the other classes. We tackled this issue with *BatchPL*, a sample acquisition scheme that selects highly informative samples for pseudo-labeling. Combined with consistency learning, our approach achieved state-of-the-art performance on both medical and non-medical classification data sets. This confirmed hypothesis 2.

Medical imaging with deep learning can also be applied to other tasks than classification. Regression tasks cover many areas of application, including forensic age estimation, natural landmark localization, instrument pose estimation and tracking, cell detection in histology, and deformable registration. We extended estimation and calibration of predictive uncertainty (σ *scaling*) to deep regression and evaluated it on different medical imaging regression tasks. In addition to UCE, we computed posterior prediction intervals to evaluate the quality of the estimated uncertainty, which showed that the calibrated 99 % prediction interval correctly contains 99 % of the ground truth values. Calibrated uncertainty in regression was also able to detect a shift in the data distribution, with calibrated Bayesian models outperforming deep ensembles. This confirmed the first part of hypothesis 3.

If the output of a regression model is itself an image, the model can be trained to solve generative tasks such as image enhancement or denoising. However, an unsolved problem in supervised learning for generative tasks are hallucinations, where a deep model embeds features of training set images into outputs of images at test time. Hallucinations can remove or include pathological structures, which prevents the actual use of deep generative models in medical imaging. To mitigate this problem, we provided a Bayesian approach to deep image prior for denoising different medical modalities (*MCDIP*). The method of deep image prior is not affected by hallucinations (Ulyanov et al. 2018), as the model only ever has access to one single image. Our Bayesian treatment not only solved the problem of overfitting of deep image prior, but also yielded well-calibrated uncertainty maps. MCDIP outperformed standard DIP and recent state-of-the-art MCMC approaches to DIP. This confirmed the second part of hypothesis 3. Additionally, we presented semantically regularized denoising autoencoders and showed that regularizing an autoencoder with a classification model is beneficial in denoising OCT scans.

We hope that with this thesis we have made a valuable contribution to increasing the acceptance of deep learning methods for medical imaging by physicians and bringing them closer to the clinical routine. We argue that predictive uncertainty has to be considered in any medical imaging problem that is approached with deep learning.

5.1 Outlook

Even though Bayesian statistics dates back to the late 18th century, only very recent advancements in approximate and variational Bayesian methods have enabled to reason about uncertainty in high dimensional data such as medical images. The medical imaging community slowly starts to adopt these methods and we are happy to see the emergence of conferences dedicated to uncertainty estimation such as the “Uncertainty for Safe Utilization of Machine Learning in Medical Imaging” (UNSURE) workshop[1] or the “Quantification of Uncertainties in Biomedical Image Quantification” (Qubiq) challenge[2] at the renowned MICCAI conference. The applications of Bayesian methods to medical imaging shown in this thesis only cover a very limited set of problems.

In the future, we expect to see new methods for uncertainty quantification. The methods used in our works involve some sort of Monte Carlo sampling, increasing the computational complexity. Sampling-free approaches could remove the additional effort and reduce the costs required for Bayesian analysis. Besides uncertainty estimation itself, we expect to see practical applications of uncertainty in, e.g., clinical risk management or medical decision making. Uncertainty estimation is needed to translate machine learning techniques into clinical practice and to ensure its safe application under real-world conditions.

---

[1] https://unsuremiccai.github.io

[2] https://qubiq21.grand-challenge.org

Bibliography

Parts of this thesis have been published in international, peer-reviewed conferences and journals. In total, two journal papers and eight conference papers have been published as first author, that are part of this thesis. The consent of all co-authors for this was obtained.

Own Publications

Laves, Max-Heinrich, Jens Bicker, Lüder A. Kahrs, and Tobias Ortmaier (2019). "A dataset of laryngeal endoscopic images with comparative study on convolution neural network-based semantic segmentation". In: *International Journal of Computer Assisted Radiology and Surgery* 14.3, pp. 483–492. DOI: `10.1007/s11548-018-01910-0`.

Laves, Max-Heinrich, Sontje Ihler, Jacob F Fast, Lüder A Kahrs, and Tobias Ortmaier (2020). "Well-Calibrated Regression Uncertainty in Medical Imaging with Deep Learning". In: *Medical Imaging with Deep Learning*.

– (2021). "Recalibration of Aleatoric and Epistemic Regression Uncertainty in Medical Imaging". In: *Journal of Machine Learning for Biomedical Imaging* 2021:008. arXiv:2104.12376, pp. 1–26.

Laves, Max-Heinrich, Sontje Ihler, Lüder A. Kahrs, and Tobias Ortmaier (2019a). "Retinal OCT disease classification with variational autoencoder regularization". In: *Proceedings of Computer Assisted Radiology and Sugery Congress (CARS)*. arXiv:1904.00790.

– (2019b). "Semantic denoising autoencoders for retinal optical coherence tomography". In: *SPIE/OSA European Conference on Biomedical Optics*. Vol. 11078, pp. 86–89. DOI: `10.1117/12.2526936`.

Laves, Max-Heinrich, Sontje Ihler, Karl-Philipp Kortmann, and Tobias Ortmaier (2019). "Well-calibrated Model Uncertainty with Temperature Scaling for Dropout Variational Inference". In: *Bayesian Deep Learning Workshop (NeurIPS)*. arXiv:1909.13550.

– (2020). "Calibration of Model Uncertainty for Dropout Variational Inference". In: *arXiv*. arXiv:2006.11584.

Laves, Max-Heinrich, Sontje Ihler, and Tobias Ortmaier (2019). "Deformable Medical Image Registration Using a Randomly-Initialized CNN as Regularization Prior". In: *Medical Imaging with Deep Learning*. arXiv:1908.00788.

Laves, Max-Heinrich, Andreas Schoob, Lüder A. Kahrs, Tom Pfeiffer, Robert Huber, and Tobias Ortmaier (2017). “Feature tracking for automated volume of interest stabilization on 4D-OCT images”. In: *SPIE Medical Imaging*. Vol. 10135, pp. 256–262. DOI: `10.1117/12.2255090`.

Laves, Max-Heinrich, Malte Tölle, and Tobias Ortmaier (2020). “Uncertainty Estimation in Medical Image Denoising with Bayesian Deep Image Prior”. In: *UNSURE Workshop (MICCAI)*. arXiv:2008.08837.

Tölle[3], Malte, Max-Heinrich Laves[3], and Alexander Schlaefer (2021). “A Mean-Field Variational Inference Approach to Deep Image Prior for Inverse Problems in Medical Imaging”. In: *Medical Imaging with Deep Learning*. openreview:DvV_blKLiB4.

General Bibliography

Agostinelli, Forest, Michael R Anderson, and Honglak Lee (2013). “Adaptive Multi-Column Deep Neural Networks With Application to Robust Image Denoising”. In: *Advances in Neural Information Processing Systems*, pp. 1493–1501.

Ashburner, John (2007). “A fast diffeomorphic image registration algorithm”. In: *NeuroImage* 38.1, pp. 95–113. DOI: `10.1016/j.neuroimage.2007.07.007`.

Ashukha, Arsenii, Alexander Lyzhov, Dmitry Molchanov, and Dmitry Vetrov (2020). “Pitfalls of In-Domain Uncertainty Estimation and Ensembling in Deep Learning”. In: *International Conference on Learning Representations*.

Avants, Brian B, Nicholas J Tustison, Gang Song, Baohua Wu, Michael Stauffer, Matthew M McCormick, Hans J Johnson, and James C Gee (2012). “A unified image registration framework for ITK”. In: *International Workshop on Biomedical Image Registration*, pp. 266–275.

Bachman, Philip, R Devon Hjelm, and William Buchwalter (2019). “Learning Representations by Maximizing Mutual Information Across Views”. In: *Advances in Neural Information Processing Systems*, pp. 15535–15545.

Balakrishnan, Guha, Amy Zhao, Mert R. Sabuncu, John Guttag, and Adrian V. Dalca (2019). “VoxelMorph: A Learning Framework for Deformable Medical Image Registration”. In: *IEEE Transactions on Medical Imaging* 38.8, pp. 1788–1800. DOI: `10.1109/TMI.2019.2897538`.

Bengio, Yoshua, Li Yao, Guillaume Alain, and Pascal Vincent (2013). “Generalized Denoising Auto-Encoders as Generative Models”. In: *Advances in Neural Information Processing Systems*, pp. 899–907.

---

[3] Shared first authorship

Bernard, O., A. Lalande, C. Zotti, F. Cervenansky, X. Yang, et al. (2018). "Deep Learning Techniques for Automatic MRI Cardiac Multi-Structures Segmentation and Diagnosis: Is the Problem Solved?" In: *IEEE Transactions on Medical Imaging* 37.11, pp. 2514–2525. DOI: `10.1109/TMI.2018.2837502`.

Bishop, Christopher M. (2006). *Pattern Recognition and Machine Learning*. Springer. ISBN: 978-0-387-31073-2.

Blei, David M, Alp Kucukelbir, and Jon D McAuliffe (2017). "Variational Inference: A Review for Statisticians". In: *Journal of the American Statistical Association* 112.518, pp. 859–877. DOI: `10.1080/01621459.2017.1285773`.

Blundell, Charles, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra (2015). "Weight Uncertainty in Neural Networks". In: *International Conference on Machine Learning*, pp. 1613–1622.

Bragman, Felix JS, Ryutaro Tanno, Zach Eaton-Rosen, Wenqi Li, David J Hawkes, Sebastien Ourselin, Daniel C Alexander, Jamie R McClelland, and M Jorge Cardoso (2018). "Uncertainty in multitask learning: joint representations for probabilistic MR-only radiotherapy planning". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 3–11.

Brier, Glenn W (1950). "Verification of Forecasts Expressed in Terms of Probability". In: *Monthly Weather Review* 78.1, pp. 1–3.

Cabezas, Mariano, Arnau Oliver, Xavier Lladó, Jordi Freixenet, and Meritxell Bach Cuadra (2011). "A review of atlas-based segmentation for magnetic resonance brain images". In: *Computer Methods and Programs in Biomedicine* 104.3, e158–e177. DOI: `10.1016/j.cmpb.2011.07.015`.

Caron, Mathilde, Piotr Bojanowski, Armand Joulin, and Matthijs Douze (2018). "Deep Clustering for Unsupervised Learning of Visual Features". In: *European Conference on Computer Vision*.

Chambolle, Antonin (2004). "An Algorithm for Total Variation Minimization and Applications". In: *Journal of Mathematical Imaging and Vision* 20.1–2, 89–97. DOI: `10.1023/B:JMIV.0000011325.36760.1e`.

Chang, S. G., Bin Yu, and M. Vetterli (2000). "Adaptive wavelet thresholding for image denoising and compression". In: *IEEE Transactions on Image Processing* 9.9, pp. 1532–1546. DOI: `10.1109/83.862633`.

Chen, Chenyi, Ari Seff, Alain Kornhauser, and Jianxiong Xiao (2015). "DeepDriving: Learning Affordance for Direct Perception in Autonomous Driving". In: *International Conference on Computer Vision*.

Chen, Ting, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton (2020). “A simple framework for contrastive learning of visual representations”. In: *arXiv*. arXiv:2002.05709.

Cheng, Jie-Zhi, Dong Ni, Yi-Hong Chou, Jing Qin, Chui-Mei Tiu, Yeun-Chung Chang, Chiun-Sheng Huang, Dinggang Shen, and Chung-Ming Chen (2016). “Computer-Aided Diagnosis with Deep Learning Architecture: Applications to Breast Lesions in US Images and Pulmonary Nodules in CT Scans”. In: *Scientific Reports* 6.1, pp. 1–13. DOI: `10.1038/srep24454`.

Cheng, Zezhou, Matheus Gadelha, Subhransu Maji, and Daniel Sheldon (2019). “A Bayesian Perspective on the Deep Image Prior”. In: *Conference on Computer Vision and Pattern Recognition*, pp. 5443–5451.

Çiçek, Özgün, Ahmed Abdulkadir, Soeren S Lienkamp, Thomas Brox, and Olaf Ronneberger (2016). “3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation”. In: *International Conference on Medical Image Computing and Computer-assisted Intervention*, pp. 424–432. DOI: `10.1007/978-3-319-46723-8_49`.

Cornish, Rob, Paul Vanetti, Alexandre Bouchard-Cote, George Deligiannidis, and Arnaud Doucet (2019). “Scalable Metropolis-Hastings for Exact Bayesian Inference with Large Datasets”. In: *International Conference on Machine Learning*, pp. 1351–1360.

Creswell, Antonia, Alison Pouplin, and Anil A Bharath (2018). “Denoising adversarial autoencoders: classifying skin lesions using limited labelled training data”. In: *IET Computer Vision* 12.8, pp. 1105–1111.

Dabov, Kostadin, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian (2007). “Image Denoising By Sparse 3-D Transform-domain Collaborative Filtering”. In: *IEEE Transactions on Image Processing* 16.8, pp. 2080–2095. DOI: `10.1109/TIP.2007.901238`.

Dalca, Adrian V., Guha Balakrishnan, John Guttag, and Mert R. Sabuncu (2019). “Unsupervised learning of probabilistic diffeomorphic registration for images and surfaces”. In: *Medical Image Analysis* 57, pp. 226–236. DOI: `https://doi.org/10.1016/j.media.2019.07.006`.

Doi, Kunio (2007). “Computer-aided diagnosis in medical imaging: Historical review, current status and future potential”. In: *Computerized Medical Imaging and Graphics* 31.4-5, pp. 198–211. DOI: `10.1016/j.compmedimag.2007.02.002`.

Esteva, Andre, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun (2017). “Dermatologist-level classification of skin cancer with deep neural networks”. In: *Nature* 542.7639, pp. 115–118. DOI: `10.1038/nature21056`.

Gal, Yarin (2016). “Uncertainty in Deep Learning”. PhD thesis. Department of Engineering, University of Cambridge.

Gal, Yarin and Zoubin Ghahramani (2016a). “Appendix: Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning”. In: *International Conference on Machine Learning*, pp. 1050–1059.

– (2016b). “Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning”. In: *International Conference on Machine Learning*, pp. 1050–1059.

Gal, Yarin, Jiri Hron, and Alex Kendall (2017). “Concrete Dropout”. In: *Neural Information Processing Systems*, pp. 3581–3590.

Gal, Yarin, Riashat Islam, and Zoubin Ghahramani (2017). “Deep Bayesian Active Learning with Image Data”. In: *International Conference on Machine Learning*, pp. 1183–1192.

Gessert, Nils, Matthias Schlüter, and Alexander Schlaefer (2018). “A deep learning approach for pose estimation from volumetric OCT data”. In: *Medical Image Analysis* 46, pp. 162–179. DOI: `10.1016/j.media.2018.03.002`.

Gondara, Lovedeep (2016). “Medical Image Denoising Using Convolutional Denoising Autoencoders”. In: *International Conference on Data Mining Workshops*, pp. 241–246. DOI: `10.1109/ICDMW.2016.0041`.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep Learning*. MIT Press. URL: `http://www.deeplearningbook.org`.

Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio (2014). “Generative Adversarial Nets”. In: *Advances in Neural Information Processing Systems*, pp. 2672–2680.

Grandvalet, Yves and Yoshua Bengio (2005). “Semi-supervised Learning by Entropy Minimization”. In: *Advances in Neural Information Processing Systems*, pp. 529–536.

Graves, Alex (2011). “Practical Variational Inference for Neural Networks”. In: *Advances in Neural Information Processing Systems*, pp. 2348–2356.

Guo, Chuan, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger (2017). “On Calibration of Modern Neural Networks”. In: *International Conference on Machine Learning*, pp. 1321–1330.

Gupta, Divam, Ramachandran Ramjee, Nipun Kwatra, and Muthian Sivathanu (2020). “Unsupervised Clustering using Pseudo-semi-supervised Learning”. In: *International Conference on Learning Representations*. URL: `https://openreview.net/forum?id=rJlnxkSYPS`.

Haenssle, H.A., C. Fink, R. Schneiderbauer, F. Toberer, T. Buhl, et al. (2018). “Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists”. In: *Annals of Oncology* 29.8, pp. 1836–1842. DOI: `10.1093/annonc/mdy166`.

Halabi, Safwan S., Luciano M. Prevedello, Jayashree Kalpathy-Cramer, Artem B. Mamonov, Alexander Bilbily, Mark Cicero, Ian Pan, Lucas Araújo Pereira, Rafael Teixeira Sousa, Nitamar Abdala, Felipe Campos Kitamura, Hans H. Thodberg, Leon Chen, George Shih, Katherine Andriole, Marc D. Kohli, Bradley J. Erickson, and Adam E. Flanders (2019). "The RSNA Pediatric Bone Age Machine Learning Challenge". In: *Radiology* 290.2, pp. 498–503. DOI: `10.1148/radiol.2018180736`.

Hastings, W. Keith (1970). "Monte Carlo sampling methods using Markov chains and their applications". In: *Biometrika* 57.1, pp. 97–109. DOI: `doi.org/10.1093/biomet/57.1.97`.

He, Kaiming, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick (2020). "Momentum Contrast for Unsupervised Visual Representation Learning". In: *Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2016). "Deep Residual Learning for Image Recognition". In: *Conference on Computer Vision and Pattern Recognition*, pp. 770–778.

Hein, Matthias, Maksym Andriushchenko, and Julian Bitterwolf (2019). "Why ReLU Networks Yield High-Confidence Predictions Far Away From the Training Data and How to Mitigate the Problem". In: *Conference on Computer Vision and Pattern Recognition*, pp. 41–50.

Held, Leonhard and Daniel Sabanés Bové (2014). *Applied Statistical Inference*. 1st ed. Springer. ISBN: 978-3-642-37887-4. DOI: `10.1007/978-3-642-37887-4`.

Heskes, Tom (1997). "Practical confidence and prediction intervals". In: *Neural Information Processing Systems*, pp. 176–182.

Heuvel, Thomas LA van den, Dagmar de Bruijn, Chris L de Korte, and Bram van Ginneken (2018). "Automated measurement of fetal head circumference using 2D ultrasound images". In: *PLOS One* 13.8, e0200412. DOI: `10.1371/journal.pone.0200412`.

Hoffman, Matthew D. and Andrew Gelman (2014). "The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo". In: *Journal of Machine Learning Research* 15.1, pp. 1593–1623.

Hogg, Robert V, Joseph McKean, and Allen T Craig (2018). *Introduction to Mathematical Statistics*. 8th ed. Pearson. ISBN: 0134686993.

Hora, Stephen C (1996). "Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management". In: *Reliability Engineering & System Safety* 54.2-3, pp. 217–223.

Hu, Yipeng, Marc Modat, Eli Gibson, Wenqi Li, Nooshin Ghavami, Ester Bonmati, Guotai Wang, Steven Bandula, Caroline M. Moore, Mark Emberton, Sébastien Ourselin, J. Alison

Noble, Dean C. Barratt, and Tom Vercauteren (2018). “Weakly-supervised convolutional neural networks for multimodal image registration”. In: *Medical Image Analysis* 49, pp. 1–13. DOI: `https://doi.org/10.1016/j.media.2018.07.002`.

Huang, Gao, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger (2017). “Densely connected convolutional networks”. In: *Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708.

Izmailov, Pavel, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson (2018). “Averaging weights leads to wider optima and better generalization”. In: *Uncertainty in Artificial Intelligence*. arXiv:1803.05407.

Jaderberg, Max, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu (2015). “Spatial Transformer Networks”. In: *Advances in Neural Information Processing Systems*, pp. 2017–2025.

Jain, Viren and Sebastian Seung (2009). “Natural image denoising with convolutional networks”. In: *Advances in Neural Information Processing Systems*, pp. 769–776.

Jang, Eric, Shixiang Gu, and Ben Poole (2016). “Categorical Reparameterization with Gumbel-Softmax”. In: *Bayesian Deep Learning Workshop (NeurIPS)*.

– (2017). “Categorical Reparameterization with Gumbel-Softmax”. In: *International Conference on Learning Representations*.

Ji, Xu, João F Henriques, and Andrea Vedaldi (2019). “Invariant Information Clustering for Unsupervised Image Classification and Segmentation”. In: *International Conference on Computer Vision*, pp. 9865–9874.

Jordan, Michael I, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul (1999). “An Introduction to Variational Methods for Graphical Models”. In: *Machine Learning* 37.2, pp. 183–233. DOI: `10.1023/A:1007665907178`.

Kendall, Alex and Yarin Gal (2017). “What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?” In: *Neural Information Processing Systems*, pp. 5574–5584.

Kermany, Daniel S., Michael Goldbaum, Wenjia Cai, Carolina C.S. Valentim, Huiying Liang, et al. (2018). “Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning”. In: *Cell* 172.5, pp. 1122–1131. DOI: `10.1016/j.cell.2018.02.010`.

Kingma, Diederik, Tim Salimans, and Max Welling (2015). “Variational dropout and the local reparameterization trick”. In: *Advances in Neural Information Processing Systems*, pp. 2575–2583.

Kingma, Diederik and Max Welling (2014). “Auto-Encoding Variational Bayes”. In: *International Conference on Learning Representations*.

Kingma, Diederik P. and Jimmy Ba (2014). “Adam: A method for stochastic optimization”. In: *arXiv*. arXiv:1412.6980.

Kirsch, Andreas, Joost van Amersfoort, and Yarin Gal (2019). “BatchBALD: Efficient and Diverse Batch Acquisition for Deep Bayesian Active Learning”. In: *Advances in Neural Information Processing Systems 32*, pp. 7026–7037.

Krizhevsky, Alex (2009). *Learning Multiple Layers of Features from Tiny Images*. Tech. rep. URL: `https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf`.

Krizhevsky, Alex and Geoffrey Hinton (2009). *Learning multiple layers of features from tiny images*.

Kuleshov, Volodymyr, Nathan Fenner, and Stefano Ermon (2018). “Accurate Uncertainties for Deep Learning Using Calibrated Regression”. In: *International Conference on Machine Learning*. Vol. 80, pp. 2796–2804.

Kull, Meelis, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach (2019). “Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with Dirichlet calibration”. In: *Advances in Neural Information Processing Systems*, pp. 12295–12305.

Kullback, Solomon and Richard A Leibler (1951). “On Information and Sufficiency”. In: *The Annals of Mathematical Statistics* 22.1, pp. 79–86.

Kumar, Ananya, Percy S Liang, and Tengyu Ma (2019). “Verified Uncertainty Calibration”. In: *NeurIPS*, pp. 3792–3803.

Kumar, Aviral, Sunita Sarawagi, and Ujjwal Jain (2018). “Trainable Calibration Measures for Neural Networks from Kernel Mean Embeddings”. In: *International Conference on Machine Learning*, pp. 2805–2814.

Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell (2017). “Simple and scalable predictive uncertainty estimation using deep ensembles”. In: *Neural Information Processing Systems*, pp. 6402–6413.

Lee, Cecilia S, Doug M Baughman, and Aaron Y Lee (2017). “Deep Learning Is Effective for Classifying Normal versus Age-Related Macular Degeneration OCT Images”. In: *Ophthalmology Retina* 1.4, pp. 322–327. DOI: `10.1016/j.oret.2016.12.009`.

Lee, Dong-Hyun (2013). “Pseudo-Label: The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks”. In: *ICML Workshop on Challenges in Representation Learning*.

Lee, Sangyoon, Min Seok Lee, and Moon Gi Kang (2018). “Poisson–Gaussian noise analysis and estimation for low-dose X-ray images in the NSCT domain”. In: *Sensors* 18.4, p. 1019.

Lempitsky, Victor, Andrea Vedaldi, and Dmitry Ulyanov (2018). “Deep Image Prior”. In: *Conference on Computer Vision and Pattern Recognition*, pp. 9446–9454. DOI: `10.1109/CVPR.2018.00984`.

Levi, Dan, Liran Gispan, Niv Giladi, and Ethan Fetaya (2019). “Evaluating and Calibrating Uncertainty Prediction in Regression Tasks”. In: *arXiv*. arXiv:1905.11659.

Li, Chunyuan, Changyou Chen, David Carlson, and Lawrence Carin (2016). “Preconditioned stochastic gradient Langevin dynamics for deep neural networks”. In: *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, pp. 1788–1794.

Litjens, Geert, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen Awm Van Der Laak, Bram Van Ginneken, and Clara I Sánchez (2017). “A survey on deep learning in medical image analysis”. In: *Medical Image Analysis* 42, pp. 60–88. DOI: `10.1016/j.media.2017.07.005`.

Loshchilov, Ilya and Frank Hutter (2019). “Decoupled Weight Decay Regularization”. In: *International Conference on Learning Representations*.

Louizos, Christos and Max Welling (2017). “Multiplicative normalizing flows for variational bayesian neural networks”. In: *International Conference on Machine Learning*, pp. 2218–2227.

Luo, Jie, Alireza Sedghi, Karteek Popuri, Dana Cobzas, Miaomiao Zhang, Frank Preiswerk, Matthew Toews, Alexandra Golby, Masashi Sugiyama, William M Wells, et al. (2019). “On the applicability of registration uncertainty”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 410–419.

Maddison, Chris J., Andriy Mnih, and Yee Whye Teh (2016). “The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables”. In: *Bayesian Deep Learning Workshop (NeurIPS)*.

Maddox, Wesley J, Pavel Izmailov, Timur Garipov, Dmitry P Vetrov, and Andrew Gordon Wilson (2019). “A Simple Baseline for Bayesian Uncertainty in Deep Learning”. In: *Advances in Neural Information Processing Systems*, pp. 13153–13164.

Maranhão, Andrew (2020). *Medical MNIST: 58954 Medical Images of 6 Classes*. kaggle.com/andrewmvd/medical-mnist.

Martel, A. L., S. Nofech-Mozes, S. Salama, S. Akbar, and M. Peikari (2019). “Assessment of Residual Breast Cancer Cellularity after Neoadjuvant Chemotherapy using Digital Pathology [Data set]”. In: *The Cancer Imaging Archive*. DOI: `10.7937/TCIA.2019.4YIBTJNO`.

Messay, Temesguen, Russell C. Hardie, and Timothy R. Tuinstra (2015). “Segmentation of pulmonary nodules in computed tomography using a regression neural network approach and its application to the Lung Image Database Consortium and Image Database Resource

Initiative dataset”. In: *Medical Image Analysis* 22.1, pp. 48–62. DOI: 10.1016/j.media.2015.02.002.

Michailovich, Oleg V and Allen Tannenbaum (2006). “Despeckling of Medical Ultrasound Images”. In: *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control* 53.1, pp. 64–78. DOI: 10.1109/TUFFC.2006.1588392.

Milletari, Fausto, Nassir Navab, and Seyed-Ahmad Ahmadi (2016). “V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation”. In: *International Conference on 3D Vision*, pp. 565–571. DOI: 10.1109/3DV.2016.79.

Mitani, Akinori, Abigail Huang, Subhashini Venugopalan, Greg S Corrado, Lily Peng, Dale R Webster, Naama Hammel, Yun Liu, and Avinash V Varadarajan (2020). “Detection of anaemia from retinal fundus images via deep learning”. In: *Nature Biomedical Engineering* 4.1, pp. 18–27. DOI: 10.1038/s41551-019-0487-z.

Naeini, Mahdi Pakdaman, Gregory F. Cooper, and Milos Hauskrecht (2015). “Obtaining Well Calibrated Probabilities Using Bayesian Binning”. In: *AAAI Conference on Artificial Intelligence*, pp. 2901–2907.

Neal, Radford M. (1993). *Probabilistic Inference Using Markov chain Monte Carlo Methods*. Tech. rep. CRG-TR-93-1. Department of Computer Science, University of Toronto Toronto, Ontario, Canada.

– (1994). “Bayesian Learning for Neural Networks”. PhD thesis. Department of Computer Science, University of Toronto.

Netzer, Yuval, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng (2011). “Reading Digits in Natural Images with Unsupervised Feature Learning”. In: *Deep Learning and Unsupervised Feature Learning Workshop (NeurIPS)*.

Niculescu-Mizil, Alexandru and Rich Caruana (2005). “Predicting Good Probabilities With Supervised Learning”. In: *International Conference on Machine Learning*, pp. 625–632.

Nix, David A and Andreas S Weigend (1994). “Estimating the mean and variance of the target probability distribution”. In: *Proceedings of IEEE International Conference on Neural Networks*. Vol. 1, pp. 55–60.

Nixon, Jeremy, Michael W Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran (2019). “Measuring Calibration in Deep Learning”. In: *CVPR Workshops*, pp. 38–41.

Oliver, Avital, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow (2018). “Realistic Evaluation of Deep Semi-Supervised Learning Algorithms”. In: *Advances in Neural Information Processing Systems*, pp. 3235–3246.

Paszke, Adam, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, et al. (2019). “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Neural Information Processing Systems*, pp. 8024–8035.

Payer, Christian, Darko Štern, Horst Bischof, and Martin Urschler (2019). “Integrating spatial configuration into heatmap regression based CNNs for landmark localization”. In: *Medical Image Analysis* 54, pp. 207–219. DOI: `10.1016/j.media.2019.03.007`.

Pereyra, Gabriel, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton (2017). “Regularizing neural networks by penalizing confident output distributions”. In: *arXiv*. arXiv:1701.06548.

Perona, P. and J. Malik (1990). “Scale-space and edge detection using anisotropic diffusion”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12.7, pp. 629–639. DOI: `10.1109/34.56205`.

Phan, Buu, Rick Salay, Krzysztof Czarnecki, Vahdat Abdelzad, Taylor Denouden, and Sachin Vernekar (2018). “Calibrating Uncertainties in Object Localization Task”. In: *Bayesian Deep Learning Workshop (NeurIPS)*. arXiv:1811.11210.

Pizer, Stephen M, E Philip Amburn, John D Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B Zimmerman, and Karel Zuiderveld (1987). “Adaptive histogram equalization and its variations”. In: *Computer vision, graphics, and image processing* 39.3, pp. 355–368.

Platt, John C. (1999). “Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods”. In: *Advances in Large-Margin Classifiers*, pp. 61–74.

Polack, Fernando P, Stephen J Thomas, Nicholas Kitchin, Judith Absalon, Alejandra Gurtman, Stephen Lockhart, John L Perez, Gonzalo Pérez Marc, Edson D Moreira, Cristiano Zerbini, et al. (2020). “Safety and efficacy of the BNT162b2 mRNA Covid-19 vaccine”. In: *New England Journal of Medicine* 383.27, pp. 2603–2615. DOI: `10.1056/NEJMoa2034577`.

Poplin, Ryan, Avinash V Varadarajan, Katy Blumer, Yun Liu, Michael V McConnell, Greg S Corrado, Lily Peng, and Dale R Webster (2018). “Prediction of cardiovascular risk factors from retinal fundus photographs via deep learning”. In: *Nature Biomedical Engineering* 2.3, pp. 158–164. DOI: `10.1038/s41551-018-0195-0`.

Rabbani, Hossein, Reza Nezafat, and Saeed Gazor (2009). “Wavelet-domain Medical Image Denoising Using Bivariate Laplacian Mixture Model”. In: *IEEE Transactions on Biomedical Engineering* 56.12, pp. 2826–2837. DOI: `10.1109/TBME.2009.2028876`.

Ran, Maosong, Jinrong Hu, Yang Chen, Hu Chen, Huaiqiang Sun, Jiliu Zhou, and Yi Zhang (2019). “Denoising of 3D Magnetic Resonance Images Using a Residual Encoder-Decoder

Wasserstein Generative Adversarial Network". In: *Medical Image Analysis* 55, pp. 165–180. DOI: 10.1016/j.media.2019.05.001.

Rizve, Mamshad Nayeem, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah (2021). "In Defense of Pseudo-Labeling: An Uncertainty-Aware Pseudo-label Selection Framework for Semi-Supervised Learning". In: *International Conference on Learning Representations*.

Roberts, Siobhan (2020-08). "How to Think Like an Epidemiologist". In: *The New York Times*, p. D8. URL: https://www.nytimes.com/2020/08/04/science/coronavirus-bayes-statistics-math.html.

Romera, Eduardo, José M. Álvarez, Luis M. Bergasa, and Roberto Arroyo (2018). "ERFNet: Efficient Residual Factorized ConvNet for Real-time Semantic Segmentation". In: *IEEE Transactions on Intelligent Transportation Systems* 19.1, pp. 263–272. DOI: 10.1109/TITS.2017.2750080.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *International Conference on Medical Image Computing and Computer-assisted Intervention*, pp. 234–241. DOI: 10.1007/978-3-319-24574-4_28.

Salimans, Tim, Diederik Kingma, and Max Welling (2015). "Markov Chain Monte Carlo and Variational Inference: Bridging the Gap". In: *International Conference on Machine Learning*, pp. 1218–1226.

Salinas, H. M. and D. C. Fernandez (2007). "Comparison of PDE-Based Nonlinear Diffusion Approaches for Image Enhancement and Denoising in Optical Coherence Tomography". In: *IEEE Transactions on Medical Imaging* 26.6, pp. 761–771. DOI: 10.1109/TMI.2006.887375.

Schlemper, Jo, Guang Yang, Pedro Ferreira, Andrew Scott, Laura-Ann McGill, Zohya Khalique, Margarita Gorodezky, Malte Roehl, Jennifer Keegan, Dudley Pennell, et al. (2018). "Stochastic deep compressive sensing for the reconstruction of diffusion tensor cardiac MRI". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 295–303.

Simonyan, Karen and Andrew Zisserman (2014). "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *arXiv*. arXiv:1409.1556.

Sohn, Kihyuk, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel (2020). "FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence". In: *arXiv*. arXiv:2001.07685.

Sokooti, Hessam, Bob de Vos, Floris Berendsen, Boudewijn P. F. Lelieveldt, Ivana Išgum, and Marius Staring (2017). "Nonrigid Image Registration Using Multi-scale 3D Convolutional

Neural Networks". In: *Medical Image Computing and Computer Assisted Intervention*, pp. 232–239.

Sotiras, Aristeidis, Christos Davatzikos, and Nikos Paragios (2013). "Deformable Medical Image Registration: A survey". In: *IEEE Transactions on Medical Imaging* 32.7, pp. 1153–1190. DOI: `10.1109/TMI.2013.2265603`.

Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov (2014). "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". In: *Journal of Machine Learning Research* 15.56, pp. 1929–1958.

Statista (2019). *Number of computed tomography (CT) scan examinations in Germany from 2005 to 2017*. statista.com/statistics/963439/computed-tomography-scan-examinations-in-germany. (Visited on 2020-12-14).

Štern, Darko, Christian Payer, Vincent Lepetit, and Martin Urschler (2016). "Automated Age Estimation from Hand MRI Volumes Using Deep Learning". In: *Medical Image Computing and Computer Assisted Intervention*, pp. 194–202.

Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna (2016). "Rethinking the Inception Architecture for Computer Vision". In: *Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826. DOI: `10.1109/CVPR.2016.308`.

Tan, Li Kuo, Yih Miin Liew, Einly Lim, and Robert A. McLaughlin (2017). "Convolutional neural network regression for short-axis left ventricle segmentation in cardiac cine MR sequences". In: *Medical Image Analysis* 39, pp. 78–86. DOI: `10.1016/j.media.2017.04.002`.

Tan, Mingxing and Quoc V Le (2019). "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks". In: *International Conference on Machine Learning*.

Tanno, Ryutaro, Aurobrata Ghosh, Francesco Grussu, Enrico Kaden, Antonio Criminisi, and Daniel C Alexander (2016). "Bayesian image quality transfer". In: *Medical Image Computing and Computer-Assisted Intervention*, pp. 265–273.

Tanno, Ryutaro, Daniel E Worrall, Aurobrata Ghosh, Enrico Kaden, Stamatios N Sotiropoulos, Antonio Criminisi, and Daniel C Alexander (2017). "Bayesian image quality transfer with CNNs: exploring uncertainty in dMRI super-resolution". In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 611–619.

TCGA (2019). *The Cancer Genome Atlas Program*. `https://www.cancer.gov/about-nci/organization/ccg/research/structural-genomics/tcga`. Accessed: 2019-04-01.

Teye, Mattias, Hossein Azizpour, and Kevin Smith (2018). "Bayesian uncertainty estimation for batch normalized deep networks". In: *arXiv*. arXiv:1802.06455.

Ulyanov, Dmitry, Andrea Vedaldi, and Victor Lempitsky (2018-06). “Deep Image Prior”. In: *Conference on Computer Vision and Pattern Recognition*. DOI: 10.1109/CVPR.2018.00984.

Vaicenavicius, Juozas, David Widmann, Carl Andersson, Fredrik Lindsten, Jacob Roll, and Thomas Schön (2019). “Evaluating model calibration in classification”. In: *AISTATS*. Vol. 89, pp. 3459–3467.

Verma, Vikas, Alex Lamb, Juho Kannala, Yoshua Bengio, and David Lopez-Paz (2019). “Interpolation Consistency Training for Semi-Supervised Learning”. In: *International Joint Conference on Artificial Intelligence*.

Wang, Nannan, Dacheng Tao, Xinbo Gao, Xuelong Li, and Jie Li (2014). “A comprehensive survey to face hallucination”. In: *International Journal of Computer Vision* 106.1, pp. 9–30.

Wang, Sida and Christopher Manning (2013). “Fast dropout training”. In: *International Conference on Machine Learning*, pp. 118–126.

Wang, Zhou, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli (2004). “Image Quality Assessment: From Error Visibility to Structural Similarity”. In: *IEEE Transactions on Image Processing* 13.4, pp. 600–612. DOI: 10.1109/TIP.2003.819861.

Welling, Max and Yee W Teh (2011). “Bayesian learning via stochastic gradient Langevin dynamics”. In: *International Conference on Machine Learning*, pp. 681–688.

Werlberger, Manuel, Thomas Pock, and Horst Bischof (2010). “Motion estimation with non-local total variation regularization”. In: *Conference on Computer Vision and Pattern Recognition*, pp. 2464–2471. DOI: 10.1109/CVPR.2010.5539945.

Wu, Ancong, Wei-Shi Zheng, and Jian-Huang Lai (2019). “Unsupervised Person Re-Identification by Camera-Aware Similarity Consistency Learning”. In: *International Conference on Computer Vision*.

Xiao, Han, Kashif Rasul, and Roland Vollgraf (2017). “Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms”. In: *arXiv*. arXiv:1708.07747.

Xiao, Tim Z., Aidan N. Gomez, and Yarin Gal (2019). “Wat heb je gezegd? Detecting Out-of-Distribution Translations with Variational Transformers”. In: *Bayesian Deep Learning Workshop (NeurIPS)*.

Xie, Yuanpu, Fuyong Xing, Xiaoshuang Shi, Xiangfei Kong, Hai Su, and Lin Yang (2018). “Efficient and robust cell detection: A structured regression approach”. In: *Medical Image Analysis* 44, pp. 245–254. DOI: 10.1016/j.media.2017.07.003.

Yi, Xin, Ekta Walia, and Paul Babyn (2019). “Generative adversarial network in medical imaging: A review”. In: *Medical Image Analysis* 58, p. 101552. DOI: `10.1016/j.media.2019.101552`.

Yin, Shi, Qinmu Peng, Hongming Li, Zhengqiang Zhang, Xinge You, Katherine Fischer, Susan L. Furth, Gregory E. Tasian, and Yong Fan (2020). “Automatic kidney segmentation in ultrasound images using subsequent boundary distance regression and pixelwise classification networks”. In: *Medical Image Analysis* 60, p. 101602. DOI: `10.1016/j.media.2019.101602`.

Žabić, Stanislav, Qiu Wang, Thomas Morton, and Kevin M Brown (2013). “A low dose simulation tool for CT systems with energy integrating detectors”. In: *Medical Physics* 40.3, p. 031102. DOI: `10.1118/1.4789628`.

Zadrozny, Bianca and Charles Elkan (2002). “Transforming Classifier Scores into Accurate Multiclass Probability Estimates”. In: *KDD*, pp. 694–699. DOI: `10.1145/775047.775151`.

Zhang, Cheng, Judith Bütepage, Hedvig Kjellström, and Stephan Mandt (2018). “Advances in Variational Inference”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41.8, pp. 2008–2026. DOI: `10.1109/TPAMI.2018.2889774`.

Zhang, K., W. Zuo, Y. Chen, D. Meng, and L. Zhang (2017). “Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising”. In: *IEEE Transactions on Image Processing* 26.7, pp. 3142–3155. DOI: `10.1109/TIP.2017.2662206`.

Zhao, He, Huiqi Li, Sebastian Maurer-Stroh, and Li Cheng (2018). “Synthesizing retinal and neuronal images with generative adversarial nets”. In: *Medical Image Analysis* 49, pp. 14–26. DOI: `10.1016/j.media.2018.07.001`.

Zhou, Dengyong, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf (2003). “Learning with Local and Global Consistency”. In: *Advances in Neural Information Processing Systems*, pp. 321–328.

Zhou, Zongwei, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang (2018). “UNet++: A Nested U-Net Architecture for Medical Image Segmentation”. In: *International Workshop on Deep Learning in Medical Image Analysis*, pp. 3–11. DOI: `10.1007/978-3-030-00889-5_1`.

Zhu, Jun-Yan, Taesung Park, Phillip Isola, and Alexei A Efros (2017). “Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks”. In: *International Conference on Computer Vision*, pp. 2223–2232.

A Appendix

A.1 Calibration of Uncertainty for Variational Inference

A.1.1 Additional Results & Figures

Here, we list additional results for the experiments from § 2.1.4.

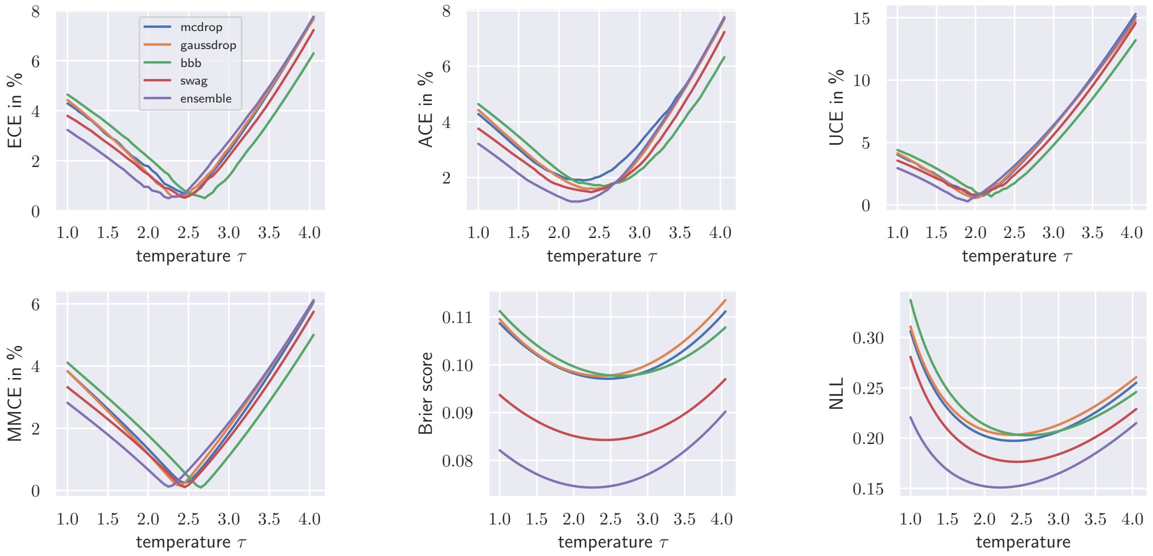


Figure A.1: Calibration error vs. softmax temperature on CIFAR-10. All metrics provide inconsistent ranking of models over τ. The metrics ECE, UCE and MMCE have a narrow region in which the optimal temperature for all models can be found. They show more a consistent ranking before and after the point of optimal temperature. This allows comparison of calibration of models if they are all over- or under-confident. However, all metrics fail at comparing underconfident models to overconfident models. Even at optimal temperature, Brier score and NLL fail at comparing calibration of models with different accuracy, as the metrics are always lower for models with better accuracy.

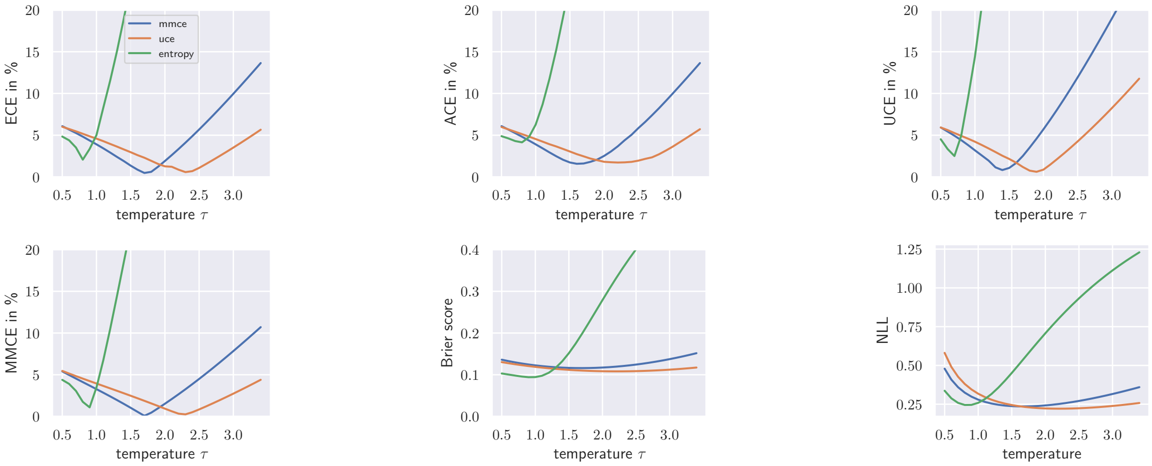


Figure A.2: Calibration error vs. softmax temperature from SWAG trained with different regularization on CIFAR-10. Both MMCE and UCE regularization lead to less overconfident models and reduce miscalibration (optimal temperature is closer to $\tau = 1$). Entropy regularization leads to underconfident models and is not as effective as MMCE and UCE regularization on CIFAR-10. MMCE and UCE regularization at optimal temperature outperform entropy regularization at optimal temperature for all metrics except Brier score.

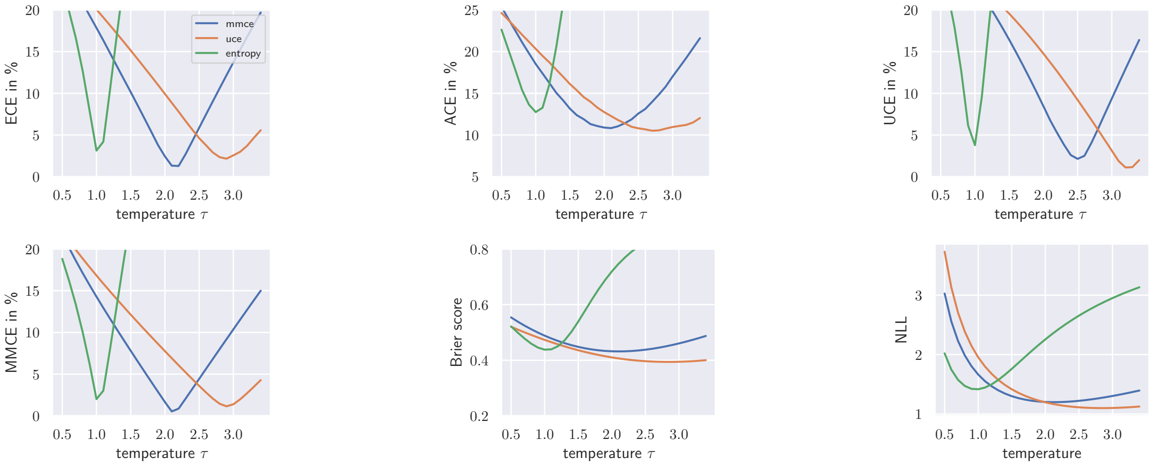


Figure A.3: Calibration error vs. softmax temperature from SWAG trained with different regularization on CIFAR-100. In this experiment, entropy regularization without temperature scaling ($\tau = 1$) was surprisingly effective and outperforms MMCE and UCE regularization. At optimal temperature both MMCE and UCE regularization outperform entropy regularization for all metrics.

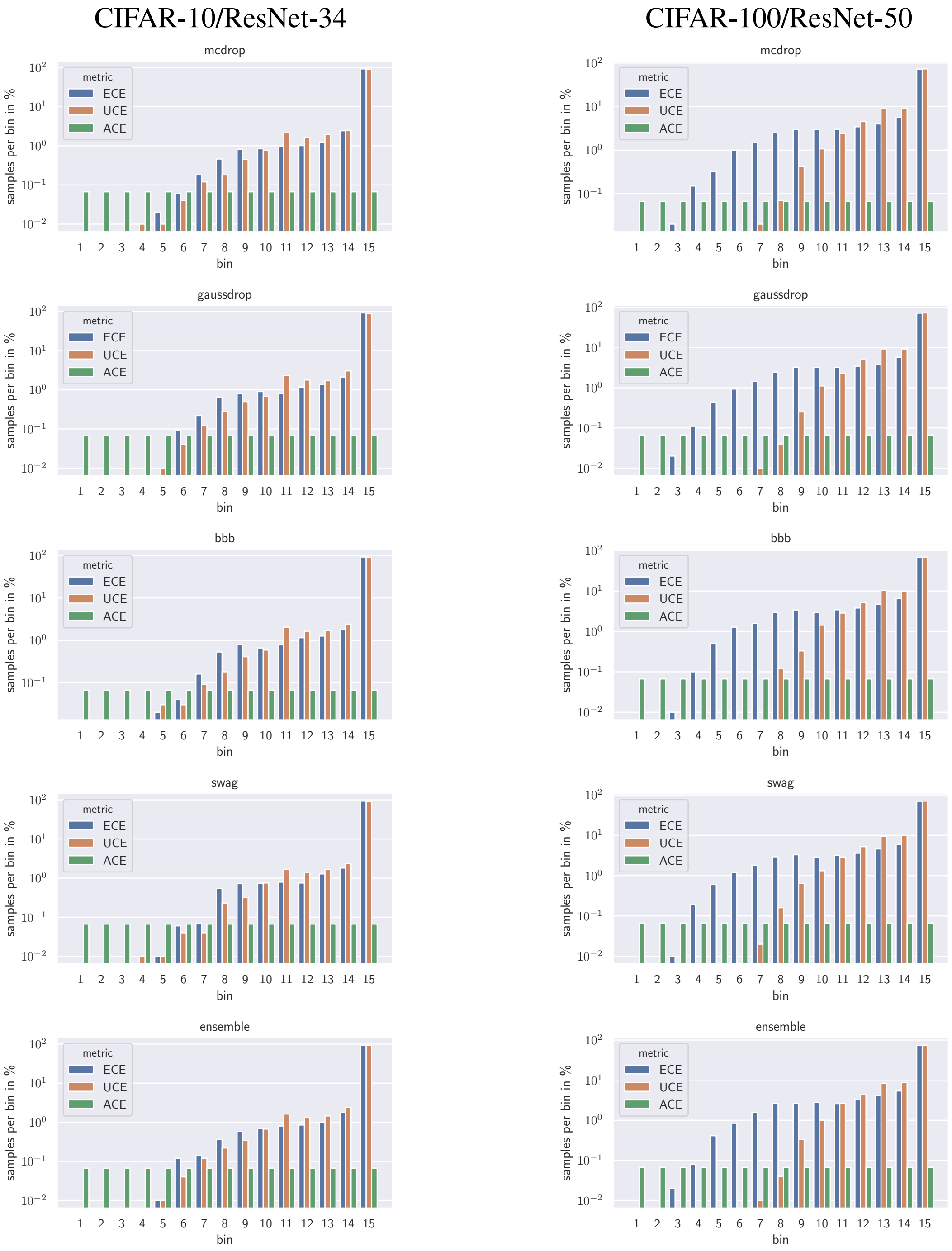


Figure A.4: Binning estimator sample distribution for ResNet-34 on CIFAR-10 (left) and for ResNet-50 on CIFAR-100 (right) with $M = 15$ bins. ECE and UCE use fixed bin widths and ACE uses an adaptive binning scheme. On CIFAR-100, UCE favors fewer bins, which makes UCE more insensitive to the total number of bins. Due to the adaptive binning, ACE is highly sensitive to the bin count.

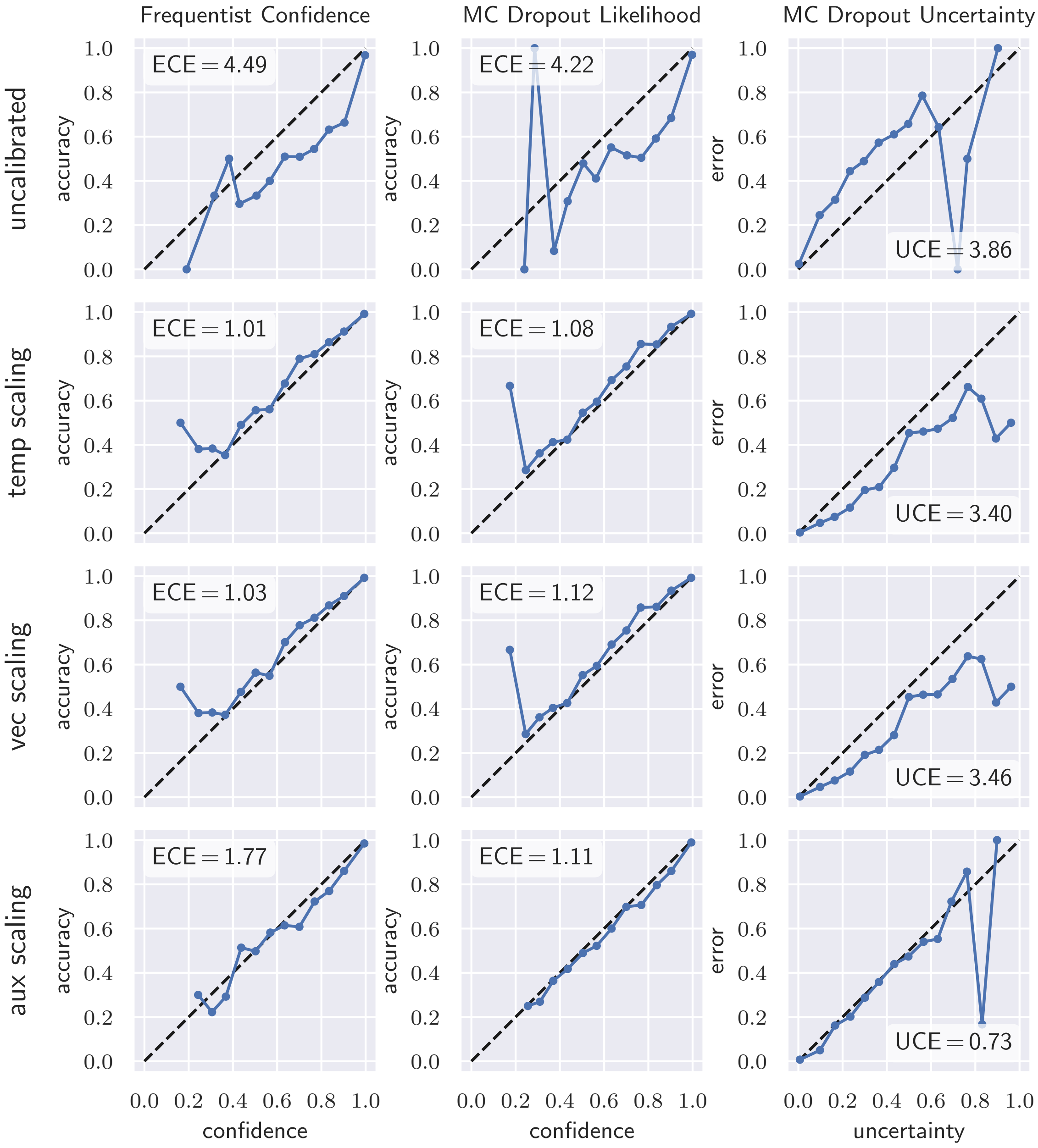


Figure A.5: Reliability diagrams ($M = 15$ bins) for ResNet-34 on CIFAR-10.

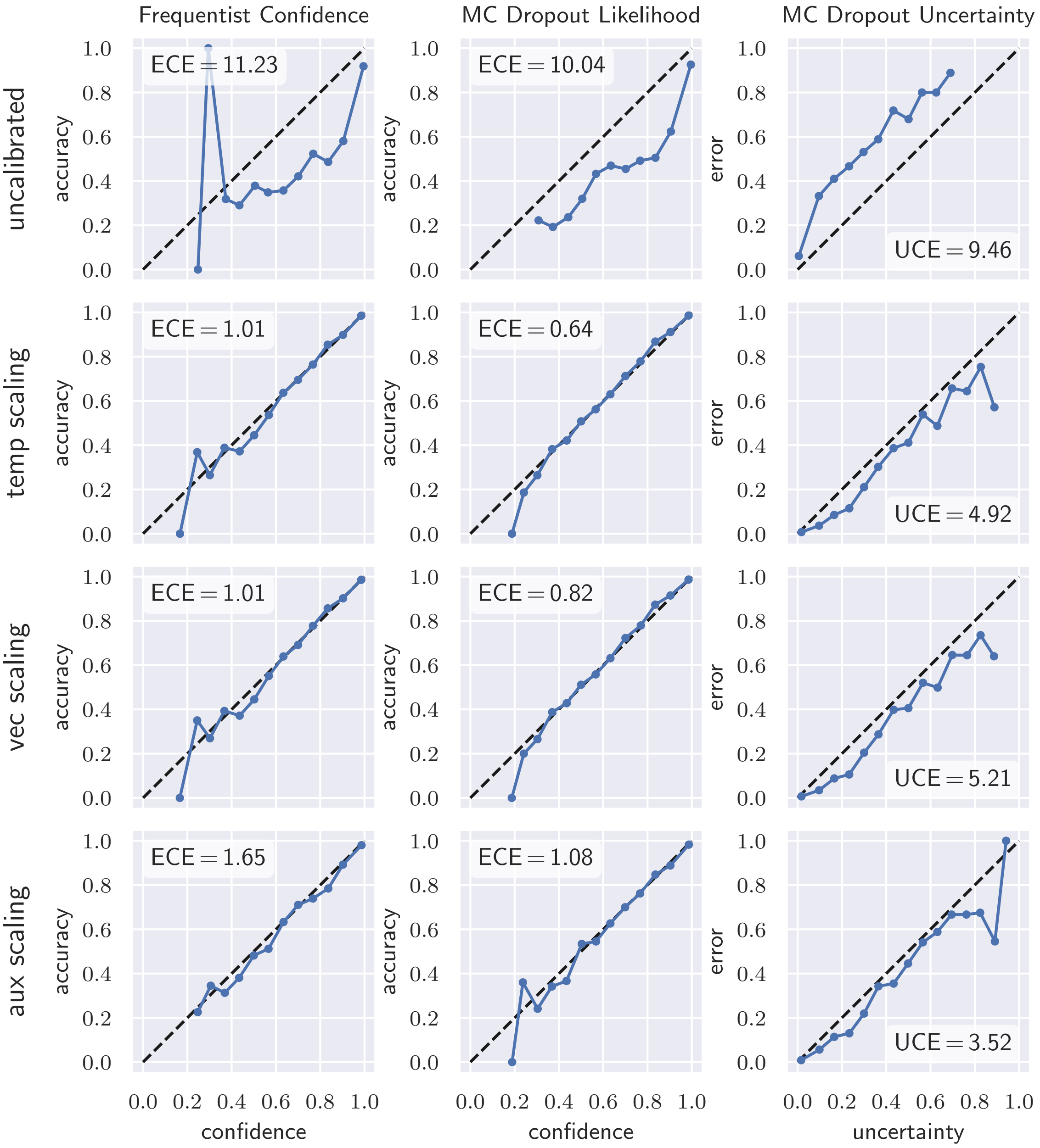


Figure A.6: Reliability diagrams ($M = 15$ bins) for DenseNet-121 on CIFAR-10.

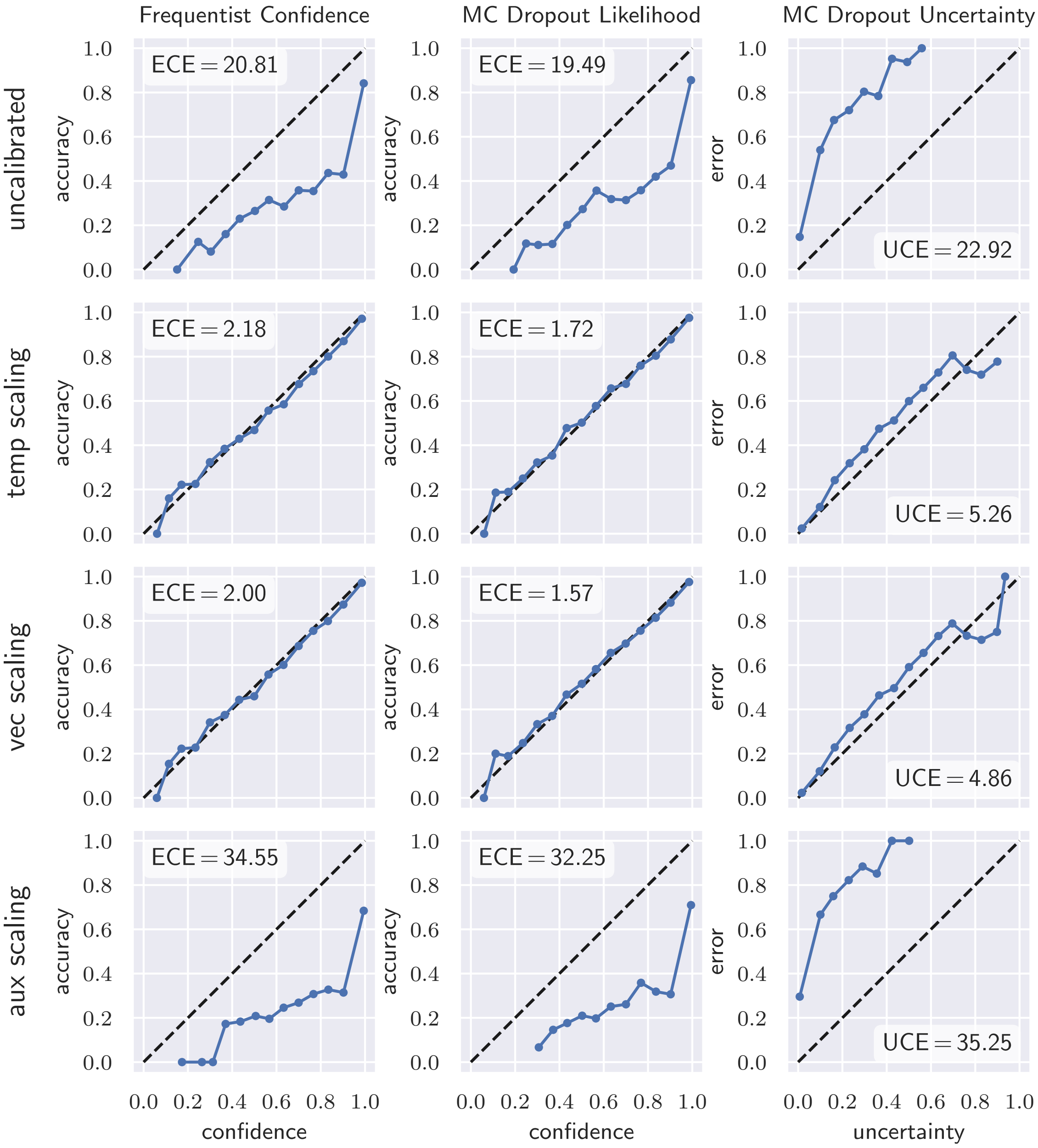


Figure A.7: Reliability diagrams ($M = 15$ bins) for ResNet-101 on CIFAR-100.

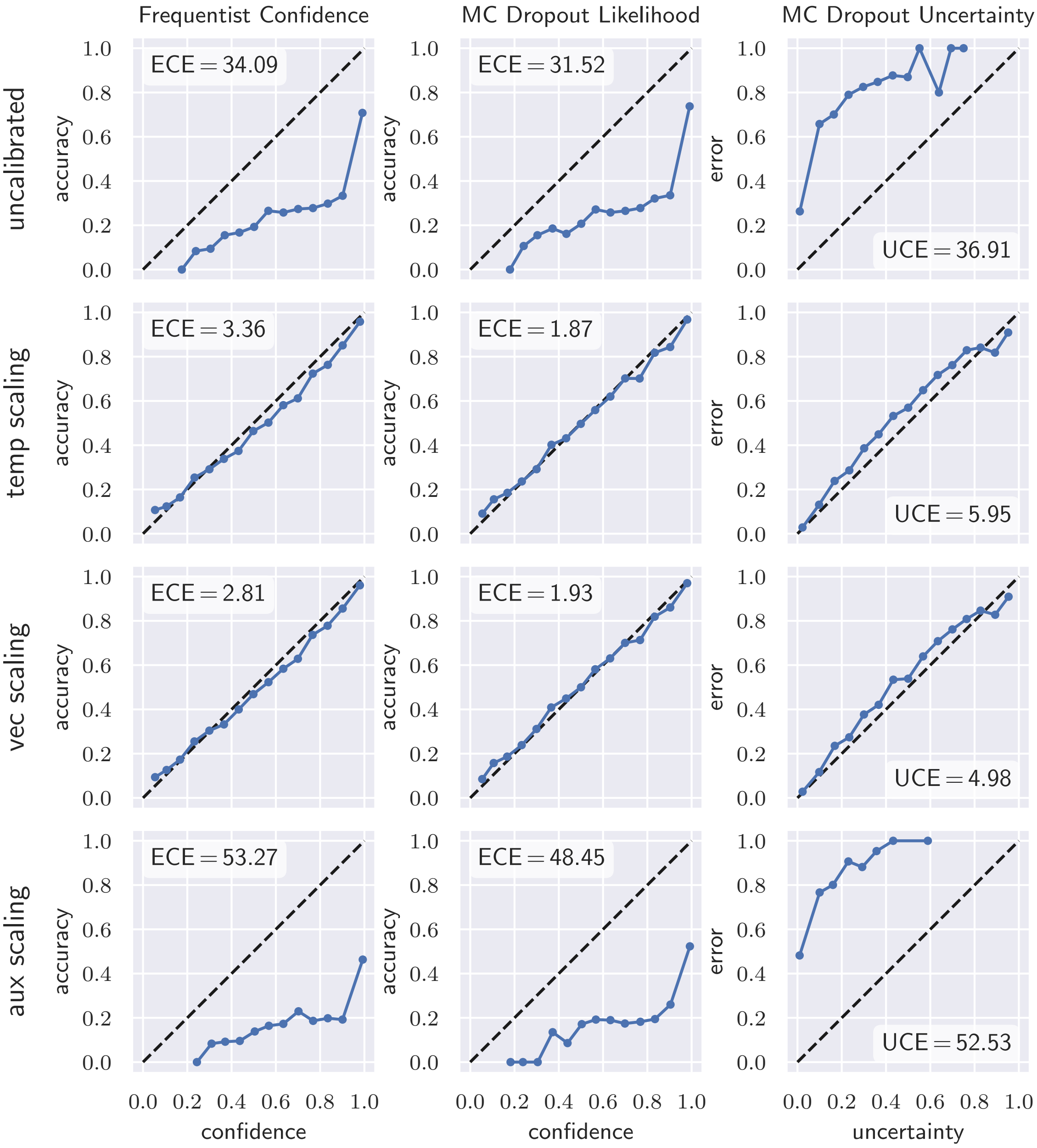


Figure A.8: Reliability diagrams ($M = 15$ bins) for DenseNet-169 on CIFAR-100.

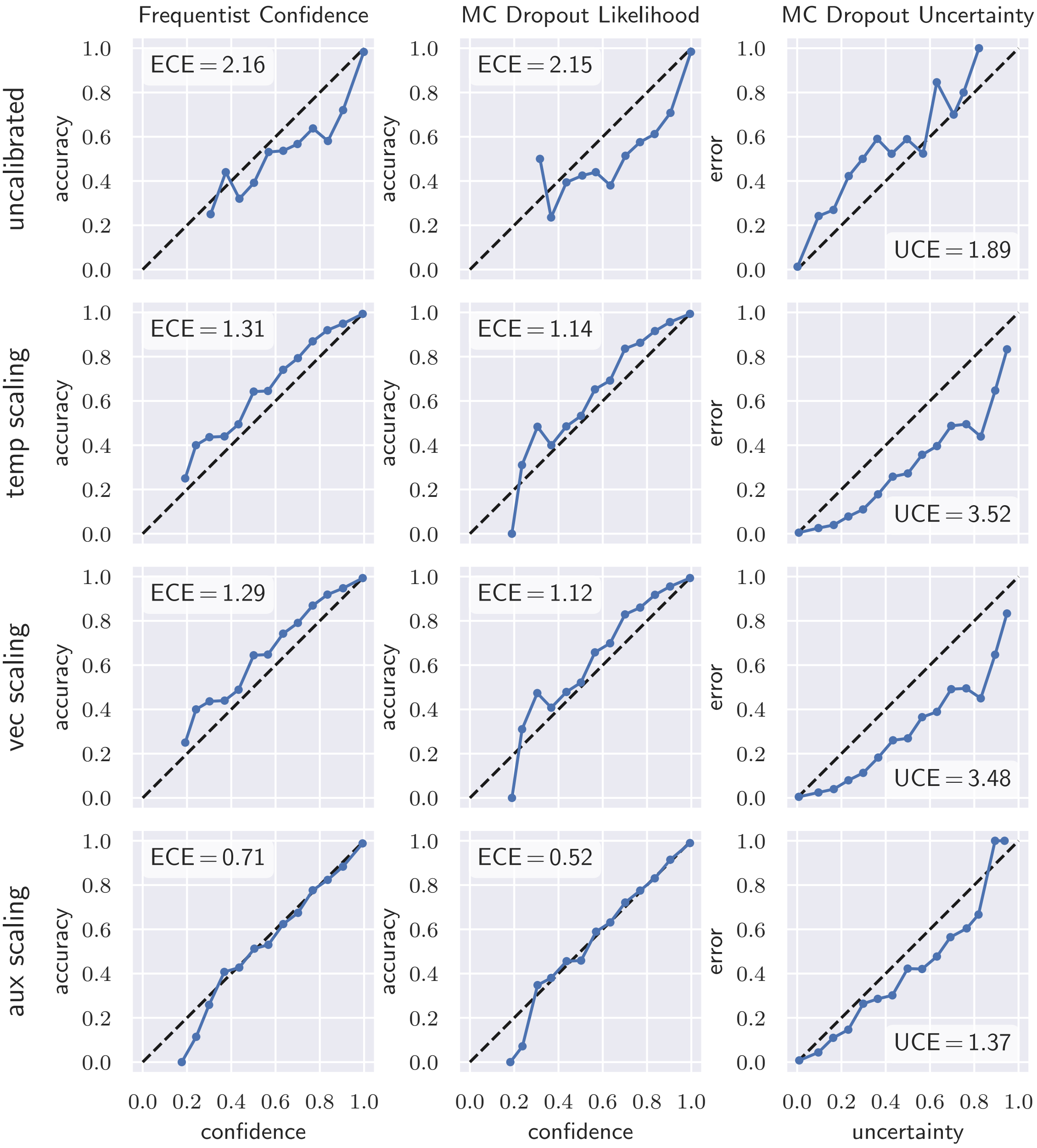


Figure A.9: Reliability diagrams ($M = 15$ bins) for ResNet-134 on SVHN.

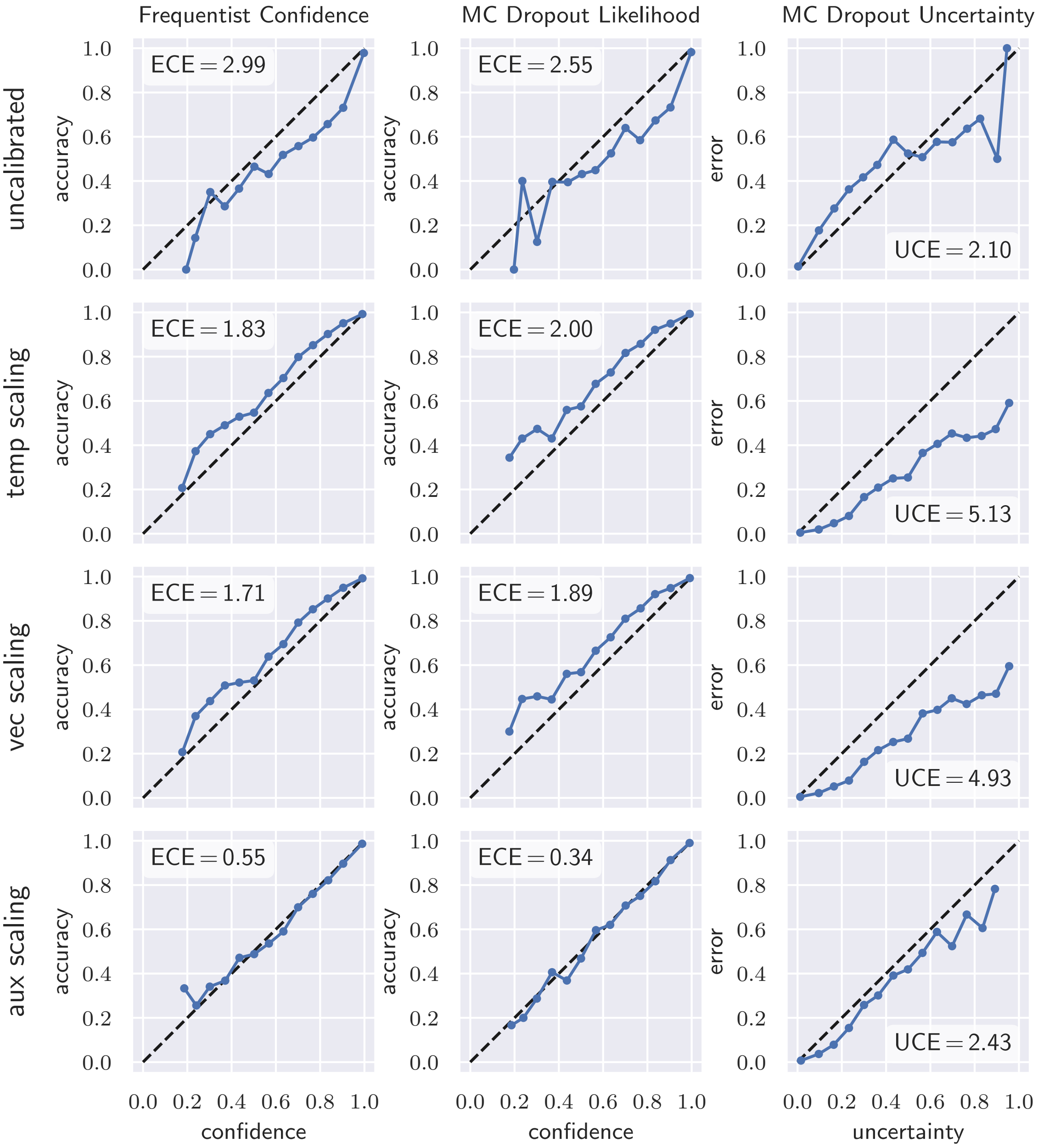


Figure A.10: Reliability diagrams ($M = 15$ bins) for DenseNet-121 on SVHN.

A.2 Regression

A.2.1 Additional Results and Calibration Diagrams

Here, we list additional results for the experiments from § 3.3. All test set runs have been repeated 5 times. Solid lines denote mean and shaded areas denote standard deviation calculated from the repeated runs.

Table A.1: Negative log-likelihood test set results for different datasets and model architectures (averaged over 5 runs). High NLL values indicate miscalibration. We also report NLL values for an ensemble of DenseNets. Bold font indicates lowest values in each experiment.

			Levi et al.			ours			
Data Set	Model	MSE	none	aux	σ	none	aux	σ	ensemble
BreastPathQ	ResNet-101	6.4e-3	-0.78	-5.06	-5.06	-2.89	**-5.17**	-5.16	0.11
	DenseNet-201	7.0e-3	-5.16	-5.84	-5.70	-5.67	**-6.03**	-5.78	
	EfficientNet-B4	6.4e-3	-3.11	-5.99	-5.53	-4.73	**-6.16**	-5.62	
BoneAge	ResNet-101	5.3e-3	-3.90	**-4.34**	**-4.34**	-3.99	**-4.34**	**-4.34**	0.07
	DenseNet-201	3.5e-3	1.74	**-4.70**	-4.69	-0.75	**-4.70**	-4.69	
	EfficientNet-B4	3.5e-3	13.61	-4.74	-4.67	6.40	**-4.75**	-4.64	
EndoVis	ResNet-101	4.0e-4	-0.53	-6.32	-6.33	-3.85	**-6.76**	-6.72	0.04
	DenseNet-201	1.1e-3	-0.72	**-6.10**	-5.99	-4.94	-6.05	-6.04	
	EfficientNet-B4	8.9e-4	-5.10	-6.06	-6.07	-5.94	**-6.17**	**-6.17**	
OCT	ResNet-101	2.0e-3	-1.08	**-5.24**	**-5.24**	-3.38	**-5.24**	**-5.24**	0.10
	DenseNet-201	1.3e-3	-5.05	-5.61	-5.61	-5.51	**-5.62**	-5.61	
	EfficientNet-B4	1.4e-3	-1.72	**-5.58**	-5.57	-4.25	**-5.58**	-5.57	

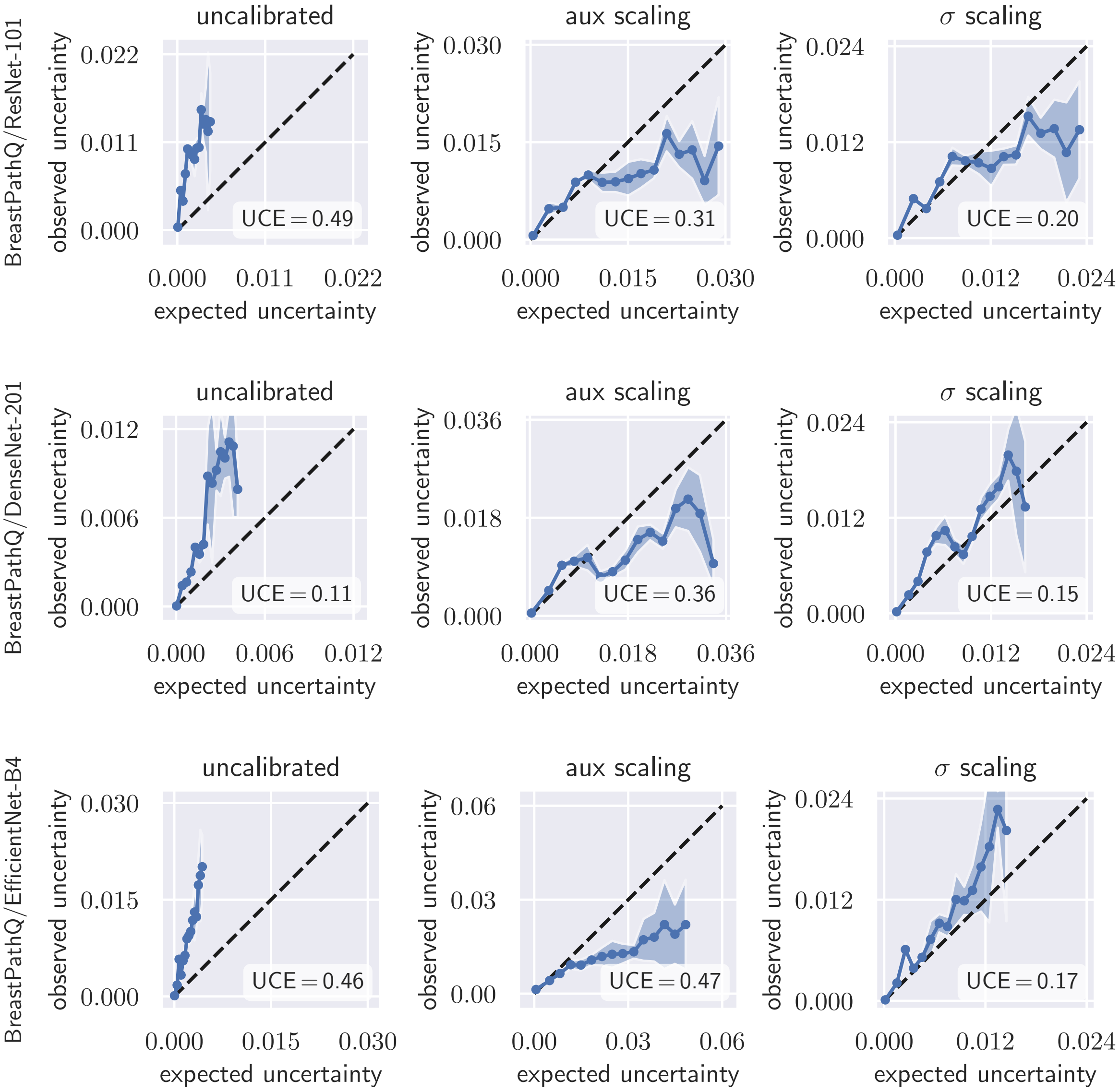


Figure A.11: BreastPathQ test set.

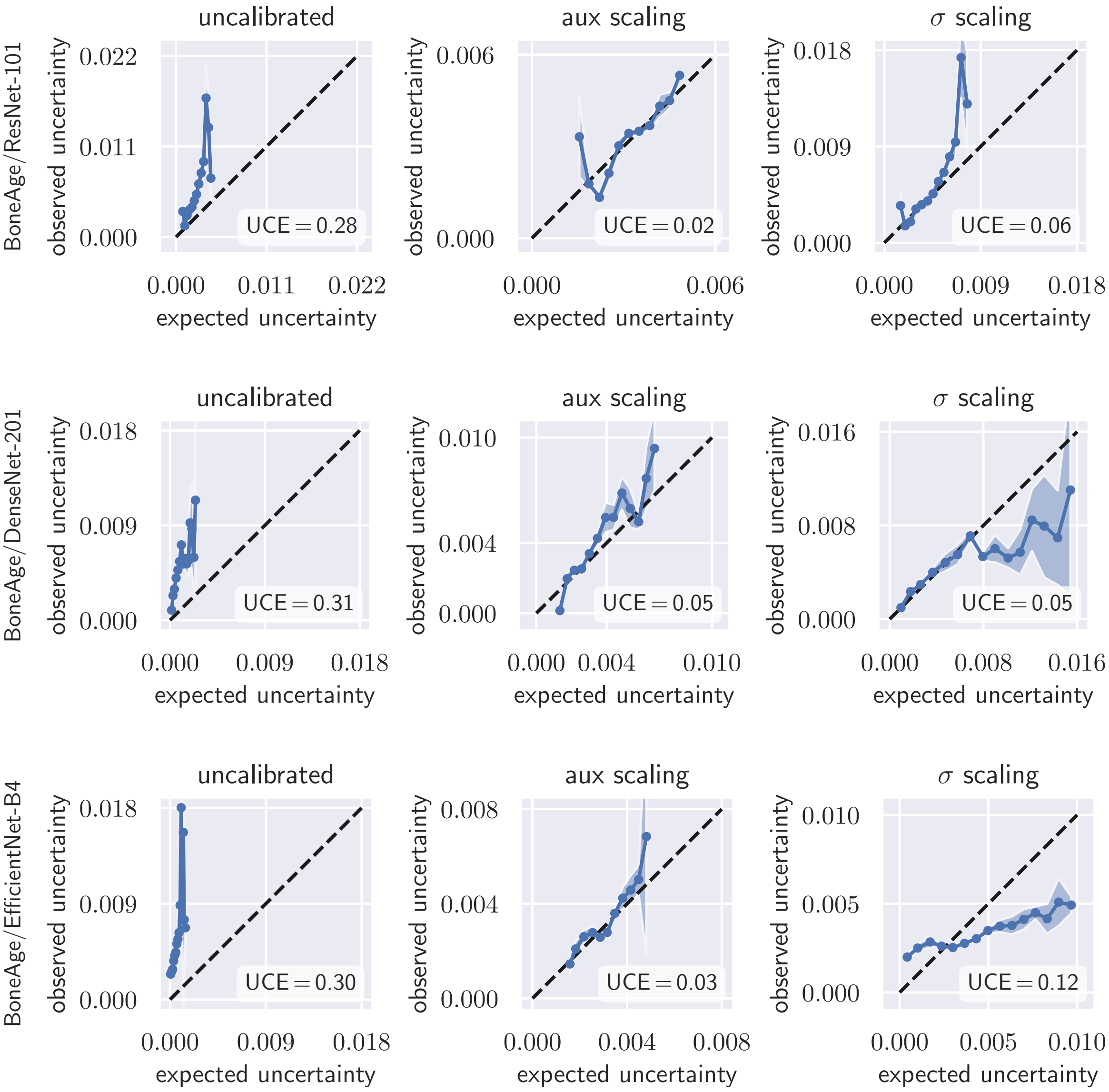


Figure A.12: BoneAge test set.

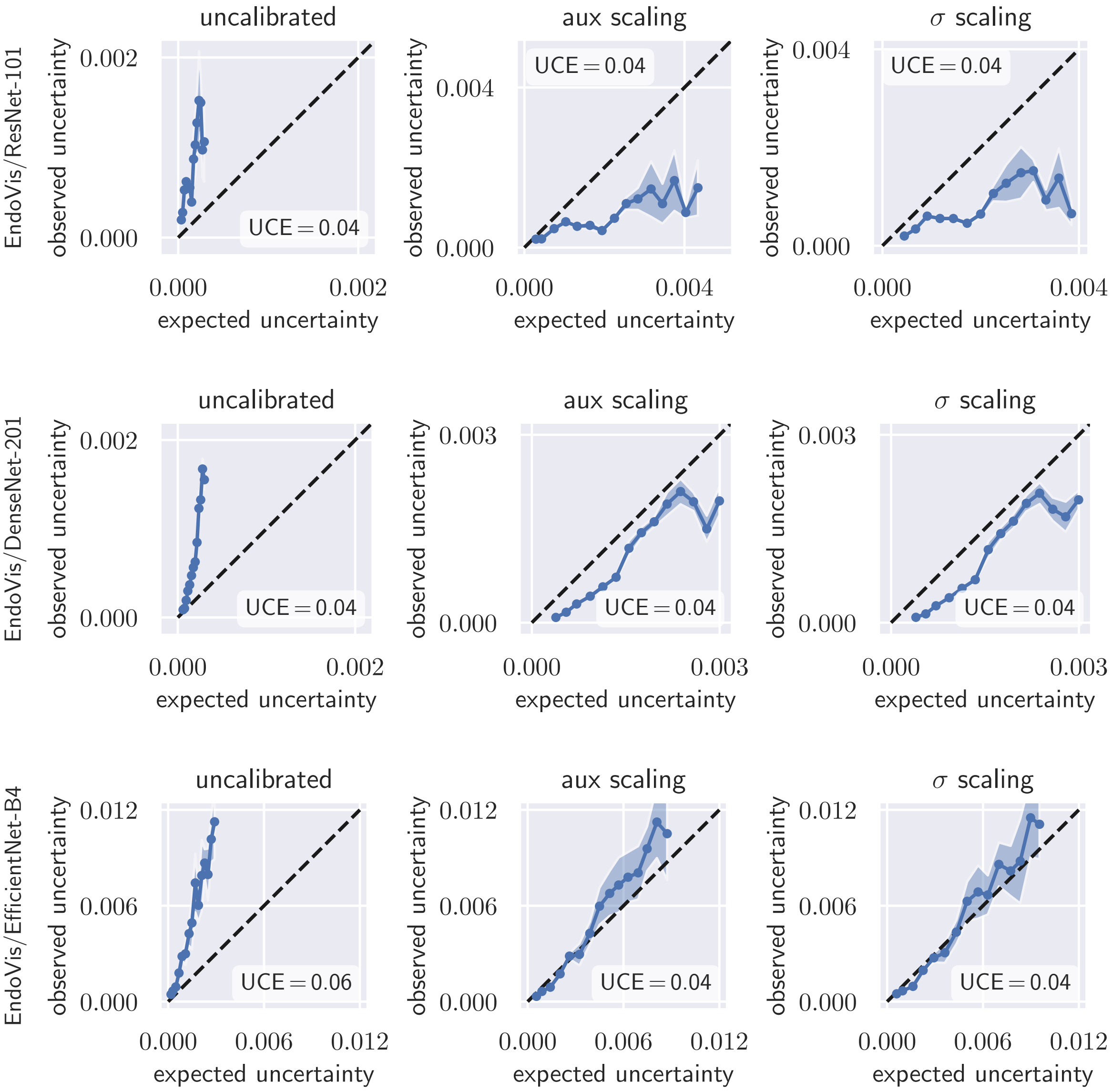


Figure A.13: EndoVis test set.

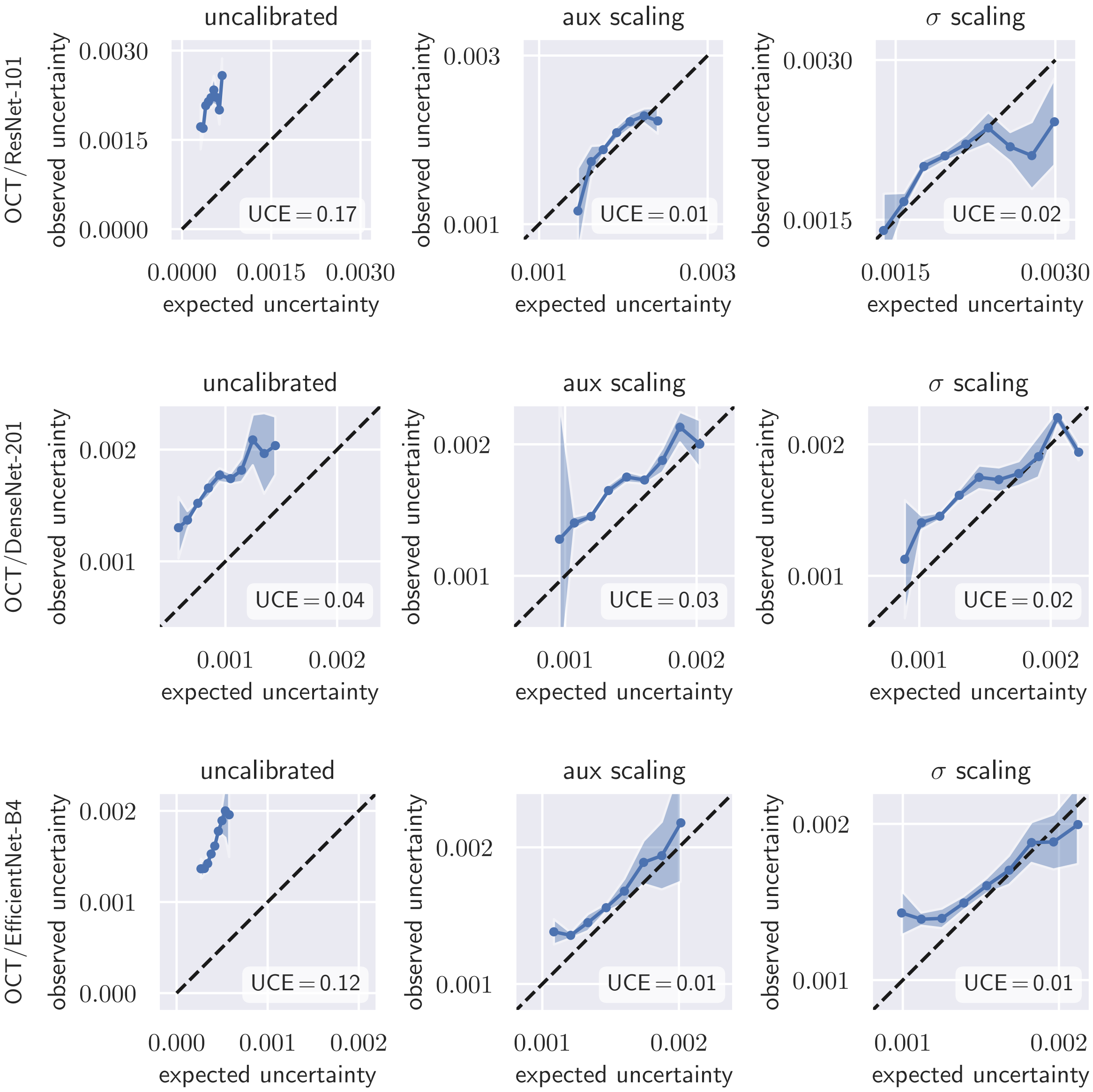


Figure A.14: OCT test set.

A.2.2 Additional Prediction Intervals

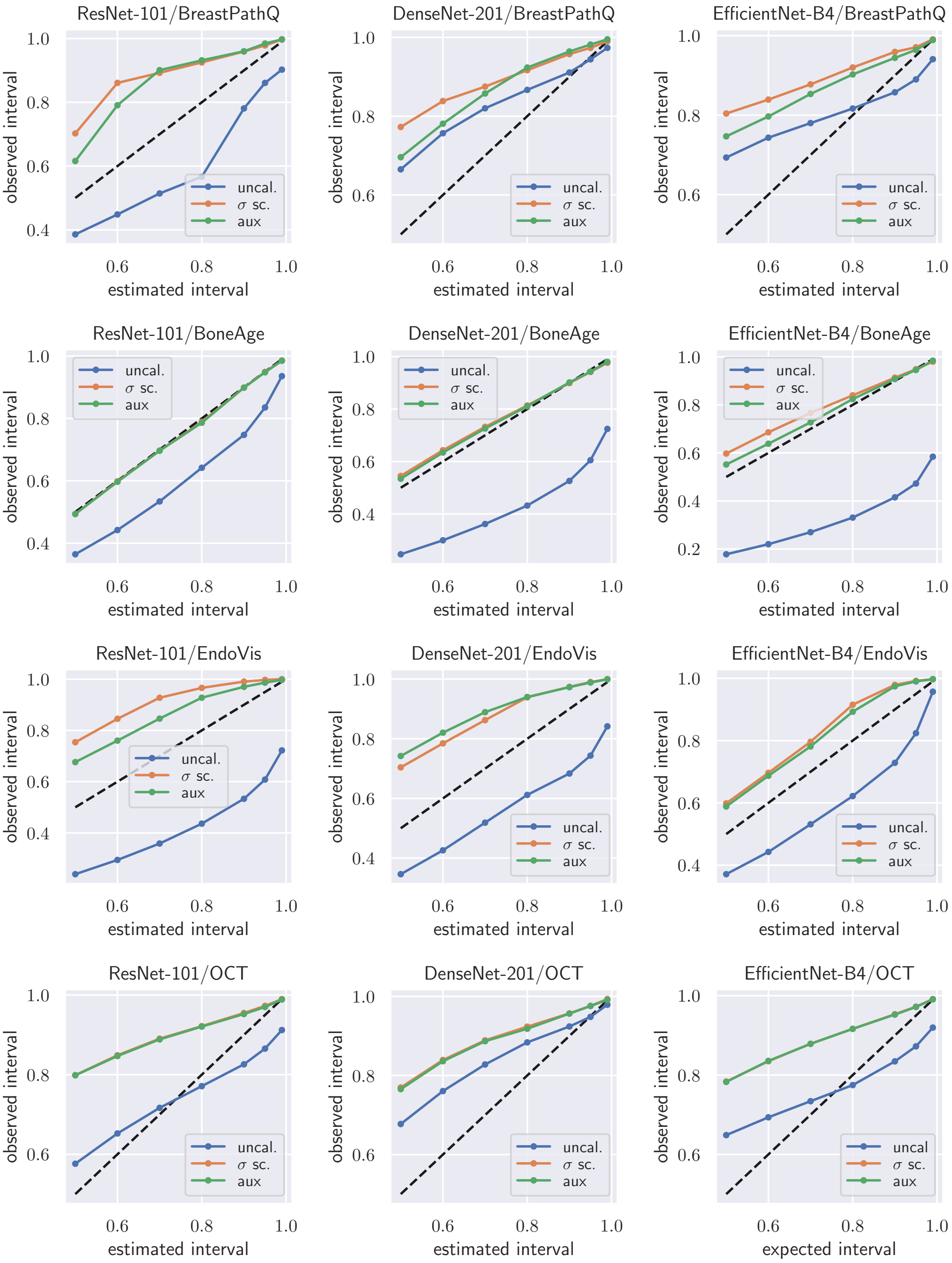


Figure A.15: Observed vs. estimated posterior prediction intervals for all networks.

A.3 Medical Image Denoising with Bayesian Deep Image Prior

A.3.1 Additional Figures

Here, we list additional results for the experiments from § 4.1.4.

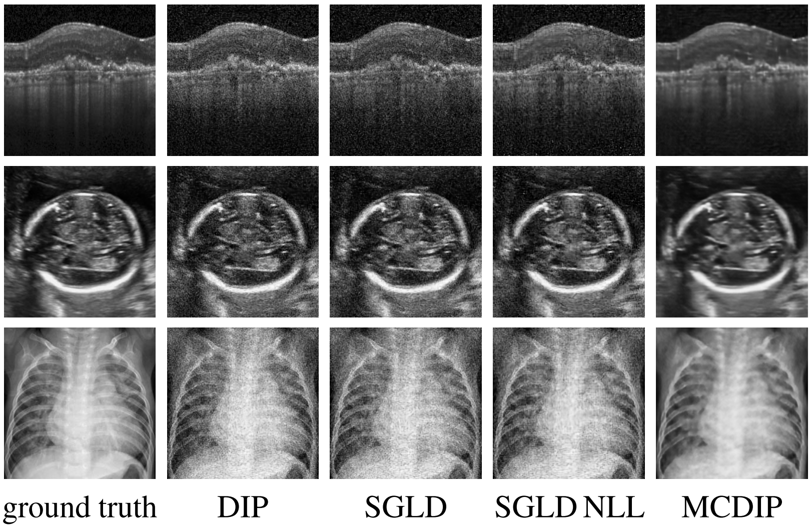


Figure A.16: Denoised images after convergence.

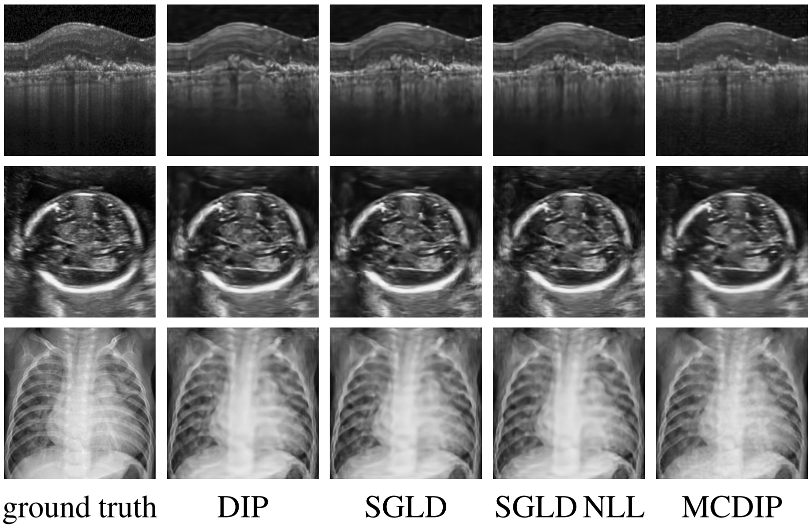


Figure A.17: Denoised images with early-stopping applied.

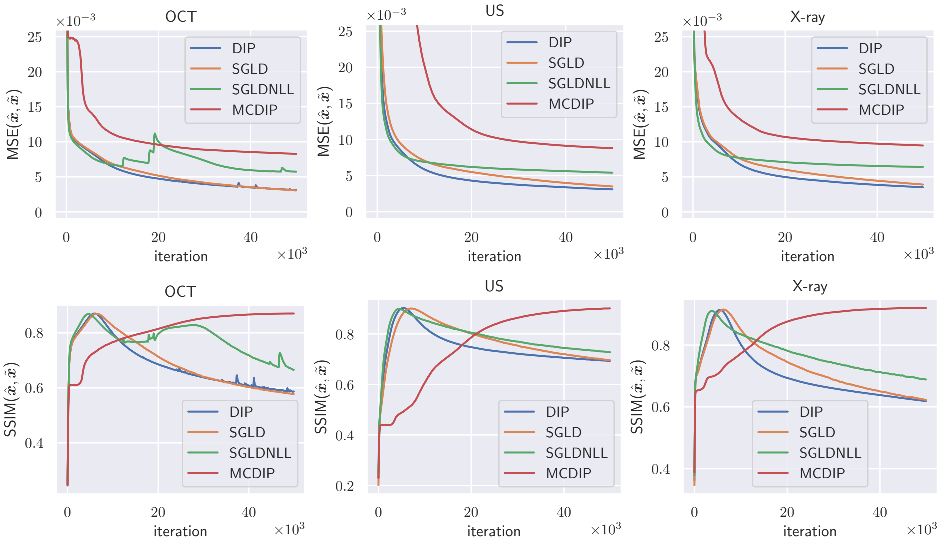


Figure A.18: MSE (top row) between denoised $\hat{\boldsymbol{x}}$ image and noisy image $\tilde{\boldsymbol{x}}$ and SSIM (bottom row) between denoised $\hat{\boldsymbol{x}}$ image and ground truth $\boldsymbol{x}$ vs. iteration. Only MCDIP does not overfit the noisy image and converges with highest similarity to the ground truth. Despite the claim of the authors, SGLD suffers from overfitting and creates the need for carefully applied early stopping (Z. Cheng et al. 2019). Note: We compared both our own implementation of SGLD and the original code provided by the authors Z. Cheng et al. (2019). The plots show means from 3 runs with different random initialization.

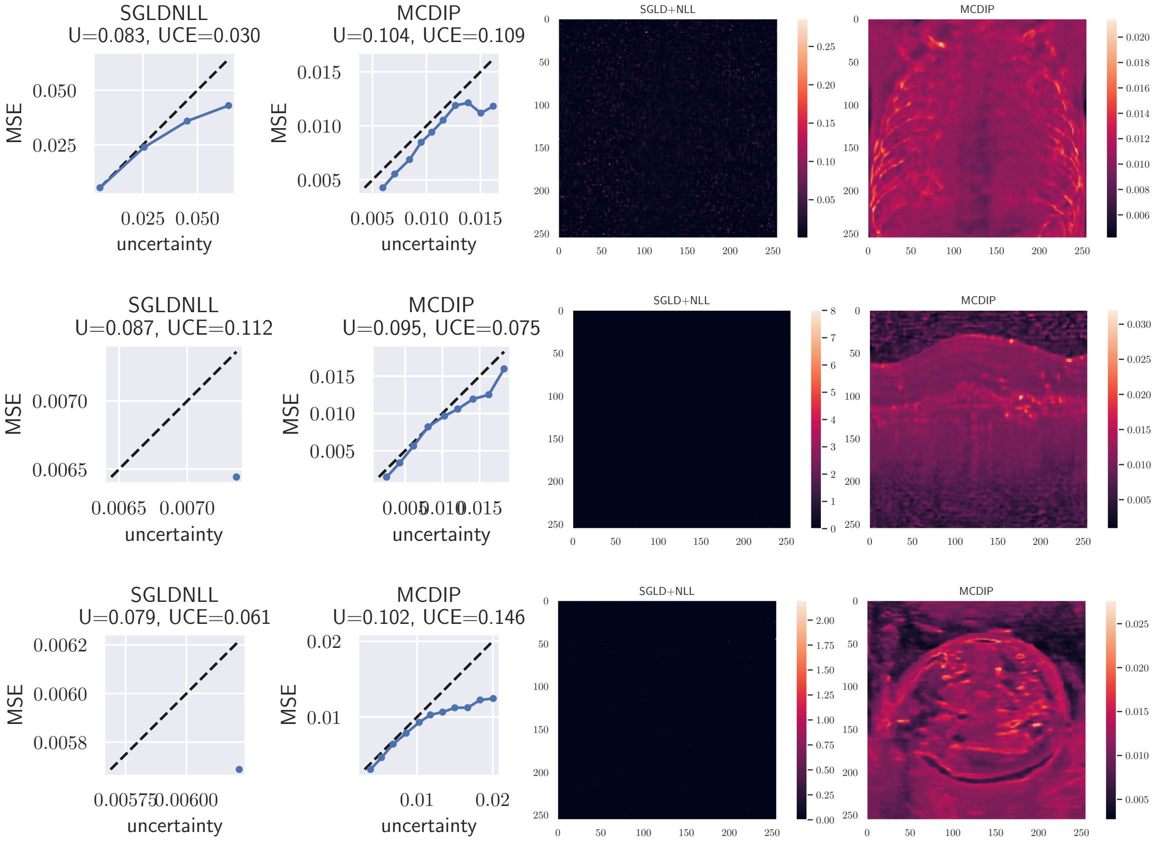


Figure A.19: Calibration diagrams and uncertainty maps for SGLD+NLL and MCDIP after convergence (best viewed with digital zoom). (Left) The calibration diagrams show MSE vs. uncertainty and provide mean uncertainty (U) and UCE values. (Right) Uncertainty maps show per-pixel uncertainty. Due to overfitting, the MSE and uncertainty from SGLD+NLL concentrates around 0.0.

A.3.2 Additional Tables

Table A.2: SSIM after convergence.

SSIM	DIP	SGLD	SGLD+NLL	MCDIP
OCT	0.582 ± 0.0	0.574 ± 0.0	0.66 ± 0.0	$\mathbf{0.872} \pm 0.0$
US	0.687 ± 0.0	0.703 ± 0.0	0.723 ± 0.0	$\mathbf{0.902} \pm 0.0$
X-ray	0.625 ± 0.0	0.631 ± 0.0	0.686 ± 0.0	$\mathbf{0.922} \pm 0.0$

Table A.3: SSIM with early-stopping.

SSIM	DIP	SGLD	SGLD+NLL	MCDIP
OCT	0.872 ± 0.0	0.872 ± 0.0	0.872 ± 0.0	0.872 ± 0.0
US	0.902 ± 0.0	$\mathbf{0.903} \pm 0.0$	0.899 ± 0.0	$\mathbf{0.903} \pm 0.0$
X-ray	0.915 ± 0.0	0.917 ± 0.0	0.912 ± 0.0	$\mathbf{0.923} \pm 0.0$

Max-H. Laves

Curriculum Vitae

Education and Academic Experience

01/2022–present	**Research Scientist**, *Philips Healthcare*, Hamburg.
01/2021–12/2021	**Postdoctoral Researcher**, *Institute of Medical Technology and Intelligent Systems*, Hamburg University of Technology.
04/2019–12/2020	**Team Lead of Research Group**, *Medical Technology and Image Processing*, Institute of Mechatronic Systems, Leibniz Universität Hannover.
10/2015–12/2020	**PhD Student**, Institute of Mechatronic Systems, Leibniz Universität Hannover. PhD thesis: *Well-Calibrated Predictive Uncertainty in Medical Imaging with Deep Learning*
10/2012–09/2015	**Master of Science**, Mechanical engineering, Leibniz Universität Hannover, final grade 1.2 (with honours, GPA equiv. 3.8/4.0), Master's thesis: *Three-dimensional tracking of soft tissue deformations for incision planning in lasery surgery*.
10/2009–09/2012	**Bachelor of Science**, Mechanical engineering, Leibniz Universität Hannover, final grade 2.2 (GPA equiv. 2.8/4.0), Bachelor's thesis: *Registration of an operating table using the kinect sensor system*.

Teaching

10/2019–12/2020	Lecturer of tutorial for *Computer and robot assisted surgery* (approx. 350 students per semester)
10/2015–09/2019	Lecturer of tutorial for *Engineering mechanics 1 & 2 for electrical engineers* (approx. 450 students per semester)
04/2017–09/2017	Lecturer of first semester project *Adaptive cruise control* where students build and program a mobile robot to drive autonomously (approx. 50 students per semester)

Military Service

07/2008–03/2009	Military service at the German Army Aviation School in Bückeburg

School Education

2001–2008	Gymnasium *Gymnasium Isernhagen*, Majors subjects: physics, math, English, A-level final grade 2.0 (GPA equiv. 3.0/4.0)
1999–2001	Orientation stage *Gottfried-Keller-Schule Hannover*
1995–1999	Elementary school *Brüder-Grimm Schule Hannover*