


# Approximative Uncertainty in Neural Network Predictions

Magnus Malmström



# Approximative Uncertainty in Neural Network Predictions

**Magnus Malmström**

**Cover illustration:** Visualization of the covariance matrix for handwritten digit classification in the MNIST dataset. The covariance matrix for the image  is estimated using the method proposed in this thesis, for detailed see Chapter 3.

Linköping studies in science and technology. Dissertations.  
No. 2358

**Approximative Uncertainty in Neural Network Predictions**

Magnus Malmström

*magnus.malmstrom@liu.se*  
*www.control.isy.liu.se*  
*Division of Automatic Control*  
*Department of Electrical Engineering*  
*Linköping University*  
*SE-581 83 Linköping*  
*Sweden*

ISBN 978-91-8075-405-7 (print)  
ISBN 978-91-8075-406-4 (PDF)  
ISSN 0345-7524

Unless otherwise stated, this work is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

Copyright © 2023 Magnus Malmström

Printed by LiU-Tryck, Linköping, Sweden 2023

*To my family and friends!*



# Abstract

Suppose data-driven black-box models, e.g., neural networks, should be used as components in safety-critical systems such as autonomous vehicles. In that case, knowing how uncertain they are in their predictions is crucial. However, this needs to be provided for standard formulations of neural networks. Hence, this thesis aims to develop a method that can, out-of-the-box, extend the standard formulations to include uncertainty in the prediction. The proposed method in the thesis is based on a local linear approximation, using a two-step linearization to quantify the uncertainty in the prediction from the neural network. First, the posterior distribution of the neural network parameters is approximated using a Gaussian distribution. The mean of the distribution is at the maximum a posteriori estimate of the parameters, and the covariance is estimated using the shape of the likelihood function in the vicinity of the estimated parameters. The second linearization is used to propagate the uncertainty in the parameters to uncertainty in the model's output. Hence, to create a linear approximation of the nonlinear model that a neural network is.

The first part of the thesis considers regression problems with examples of road-friction experiments using simulated and experimentally collected data. For the model-order selection problem, it is shown that the method does not underestimate the uncertainty in the prediction of overparametrized models.

The second part of the thesis considers classification problems. The concept of calibration of the uncertainty, i.e., how reliable the uncertainty is and how close it resembles the true uncertainty, is considered. The proposed method is shown to create calibrated estimates of the uncertainty, evaluated on classical image data sets. From a computational perspective, the thesis proposes a recursive update of the parameter covariance, enhancing the method's viability. Furthermore, it shows how quantified uncertainty can improve the robustness of a decision process by formulating an information fusion scheme that includes both temporal correlational and correlation between classifiers. Moreover, having access to a measure of uncertainty in the prediction is essential when detecting outliers in the data, i.e., examples that the neural network has yet to see during the training. On this task, the proposed method shows promising results. Finally, the thesis proposes an extension that enables a multimodal representation of the uncertainty.

The third part of the thesis considers the tracking of objects in image sequences, where the object is detected using standard neural network-based object detection algorithms. It formulates the problem as a filtering problem with the prediction of the class and the position of the object viewed as the measurements. The filtering formulation improves robustness towards false classifications when evaluating the method on examples from animal conservation in the Swedish forests.





## Populärvetenskaplig sammanfattning

Det finns en pågående trend i samhället mot mer automation. Ofta behövs det då modeller som beskriver de system som ska automatiseras. Sådana modeller kommer att vara centrala för att ge prediktioner av framtida tillstånd i systemen. Tack vare att det har blivit billigare att mäta på olika system samtidigt som det finns bättre tillgång till kraftfullare beräkningsresurser för modellering har det blivit vanligare att använda flexibla datadrivna modeller. Dessa modeller anpassas sedan till insamlad data för att ge en bra beskrivning av systemet som önskas modelleras. Artificiella neuron nätverk är ett exempel på en datadriven modell. Om dessa datadrivna modeller ska vara användbara för beslutsfattande i säkerhetskritiska system är det viktigt att det går att beskriva osäkerhet i deras prediktioner. Syftet med denna avhandling är att konstruera metoder för att beräkna osäkerhet i prediktioner från neuron nätverk.

Metoden som vi föreslår är baserad på att linjärisera det icke-linjära neuron nätverket med avseende på de parametrar som beskriver nätverket. Detta för att skapa en linjär approximation av modellen. För linjära modeller finns det tidigare arbete gjort för att beräkna osäkerhet i prediktionen, och genom att göra en linjär approximation av neuron nätverket möjliggör vi användande av dessa resultat.

Avhandlingen undersöker tre olika sorters problem där neuron nätverk ofta används som en del av lösningen. Dessa är regressionsproblem, klassificeringsproblem, och objekt detektion i bilder. Regressionsproblemen kommer att kretsa kring ett exempel där däckfriktionen modelleras som en funktion av hur mycket hjulen på bilen glider. För dessa problem undersöks det även hur valet av modellordning påverkar den beräknade osäkerheten. Här visas det att den föreslagna metoden inte underskattar osäkerheten.

För klassificeringsproblem beskrivs olika mått för att mäta osäkerhet samt hur man kan veta ifall man kan lita på den beräknade osäkerheten eller inte. Neuron nätverken som används för att lösa dessa problem blir större och mer komplicerade, detta gör det än viktigare att metoden för att beräkna osäkerhet är beräkningseffektiv. För att göra den föreslagna metoden mer beräkningseffektiv presenteras därför bland annat en rekursiv formulering av metoden. För klassificering undersöks det även hur osäkerheten kan användas för att detektera exempel som inte ingår i den insamlade data neuron nätverket är tränat på. Det presenteras även hur prediktioner från flera nätverk och prediktioner över tid kan fusioneras (vägas samman) så att besluten blir mer robusta. Till sist presenteras en utökning av metoden som möjliggör mer komplicerade osäkerhetsbeskrivningar.

Bidraget i delen som behandlar objekt detektion är att inkludera prediktionen från neuron nätverket i ett filtreringsramverk. Här är det centralt att inkludera osäkerhet i prediktionen. Genom exempel hämtade från naturvård, där svenska rovdjur följs med kamerafällor, visas det i avhandlingen att inkludering av osäkerheten i prediktionen och filtreringsramverket leder till att beslutssystemet blir mindre känsligt för feldetektioner från neuron nätverket.



## Acknowledgments

First and foremost, I would like to thank my supervisor, Fredrik Gustafsson, and my co-supervisors, Daniel Axehill and Isaac Skog, for guidance during these last five years. Thanks for all the interesting discussions and for teaching me to become a better researcher. Just as a control system relies on expert tuning to achieve precision, your mentorship and expertise have fine-tuned my research, ensuring it operates at its best. I genuinely appreciate your sharing your wisdom with me these last years.

The Automatic Control division is a wonderful workplace with a friendly and stimulating atmosphere. For this, I would like to thank my former and current colleagues. A special thanks to Martin Enqvist, the head of the division, for maintaining this welcoming workplace and to Ninna Stensgård for always being ready to help with administrative tasks. Especially, I would like to thank my fellow Ph.D. students and the research engineers for all the activities at and after work. Your dynamic presence has illuminated this journey. I couldn't have identified the path to success without your constructive and entertaining input! A special thanks to Daniel Arnström, Filipe Barbosa, Anton Kullberg, and Anja Hellander for proofreading. I am very grateful for your feedback.

The research work in this thesis has been supported by the Sweden's Innovation Agency, Vinnova, through project iQDeep (project number 2018-02700). Their funding is gratefully acknowledged. The project is a collaboration between LiU and Scania, and I would like to thank everyone involved in the project, especially the AI Technologies group at Scania (EARA), for exciting discussions and input to my work, both at the annual meetings and during the visits in Södertälje.

To my family and friends: I would like to express my heartfelt thanks for making the uncertain research path a little more predictable with your love and friendship. You are both the "variables" that make life interesting and the "constants" that provide comfort. Without your support, this would not have been possible.

Last but not least, I would like to thank anyone who has helped me become a better researcher and teacher during my past five years. The subsequent pages represent the outcome of this enriching journey.

*Linköping, October 2023*  
*Magnus Malmström*



---

# Contents

## I Background

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Background and motivation . . . . .	3
1.2	Problem formulation . . . . .	7
1.3	Contributions . . . . .	8
1.4	Thesis outline . . . . .	9
<b>2</b>	<b>System identification of neural networks</b>	<b>15</b>
2.1	Problem formulation . . . . .	15
2.2	Model sets and model structure . . . . .	16
2.3	Uniqueness . . . . .	20
2.4	Regularity conditions and identifiability . . . . .	25
2.5	Overparameterization . . . . .	26
<b>3</b>	<b>Linearized Laplacian approximation</b>	<b>29</b>
3.1	Laplacian approximation . . . . .	29
3.2	The delta method . . . . .	33
3.3	Ensemble linearized Laplacian approximation . . . . .	36
3.4	Summary . . . . .	37
<b>4</b>	<b>Uncertainty in neural network predictions</b>	<b>39</b>
4.1	Sources of uncertainty . . . . .	39
4.2	Roadmap of uncertainty methods . . . . .	41
4.3	Ensemble methods . . . . .	41
4.4	Learned uncertainty . . . . .	44
4.5	Summary . . . . .	45
<b>5</b>	<b>Concluding remarks</b>	<b>47</b>
5.1	Summary of contributions . . . . .	47
5.2	Conclusions . . . . .	48
5.3	Future work . . . . .	49

<b>Bibliography</b>	<b>51</b>
---------------------	-----------

## **II Publications**

<b>A Asymptotic Prediction Error Variance for Feedforward Neural Networks</b>	<b>63</b>
<b>B On the validity of using the delta method for calculating the uncertainty of the predictions from an overparameterized model</b>	<b>81</b>
<b>C Modeling of the tire-road friction using neural networks including quantification of the prediction uncertainty</b>	<b>97</b>
<b>D Uncertainty quantification in neural network classifiers – a local linear approach</b>	<b>113</b>
<b>E Fusion framework and multimodality for the Laplacian approximation of Bayesian neural networks</b>	<b>137</b>
<b>F Detection of outliers in classification by using quantified uncertainty in neural networks</b>	<b>165</b>
<b>G Extended target tracking utilizing machine-learning software – with applications to animal classification</b>	<b>185</b>

## **Part I**

# **Background**





# 1

---

## Introduction

For predictions from data-driven models to be viable for safety-critical applications, it is crucial to know how confident they are in their predictions. Here neural networks (NNs) are an example of such data-driven black-box model. Hence, methods to quantify the uncertainty in the prediction of NNs are required, which is the topic of this thesis. This introductory chapter gives an overview and motivation of the problem of quantifying the uncertainty in the predictions from the NNs. Furthermore, the contributions of the thesis, as well as an outline of the content of the thesis, are presented.

### 1.1 Background and motivation

It is an important part of science and technology to model the world around us, i.e., for understanding and controlling parts of it. Obtaining such models has traditionally been a tedious and time-consuming process based on physical insight and expert domain knowledge. As the computational resources on modern computers have increased, in combination with it becoming easier and cheaper to measure signals from different systems and save those measurements, the use of data-driven black-box models has become an appealing option. One example of such a flexible black-box model is NNs, which has been used in various applications. They have shown impressive results in everything from image recognition [1], learning the structure of proteins [2], and language modeling [3, 4]. They have also been used in safety-critical applications such as various control tasks [5, 6] and detecting diseases in medical images [7].

Lately, some companies have implemented fully autonomous vehicles, e.g., *Tesla*, *Waymo*, *Embark Trucks Inc.*, *Scania* and *Einride* to mention a few. One example is the self-driving bus currently tested at Scania, as seen in Fig. 1.1. Some fully autonomous vehicles are already on public roads, but using black-box mod-

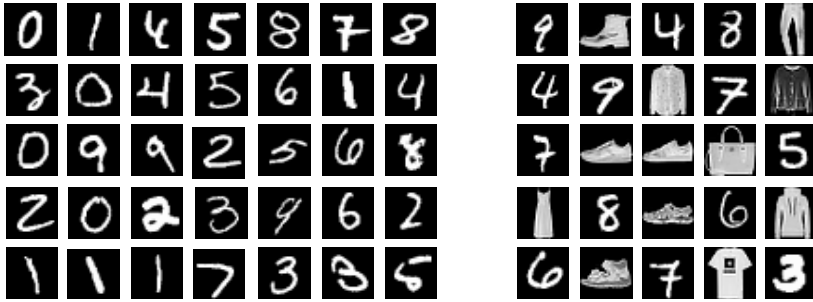


Figure 1.1: One of the early autonomous buses developed by Scania.

els such as NN in fully autonomous vehicles comes with considerable risk. This risk is partly due to their lack of ability to assess *uncertainty* in their predictions. Self-driving car accidents, e.g., [8–10], highlight this risk. For example, the lack of understanding of its surroundings contributed to the Uber accident in 2018 [8]. Hence, with the knowledge of the uncertainty in the prediction, decision processes where NNs are a vital component can make more informed decisions and provide situational awareness needed for autonomy. The lack of ability to express uncertainty is one of many reasons why the use of the NN in safety-critical applications is limited [11–13]. As Box [14] famously said, “*all models are wrong, but some are useful*”. Here, the ability to express some uncertainty in the prediction of the model is one essential component that drastically increases the usefulness of a given model. To this end, the field of automatic control, which has a long history of modeling with the inclusion of uncertainty, could play a crucial role [15].

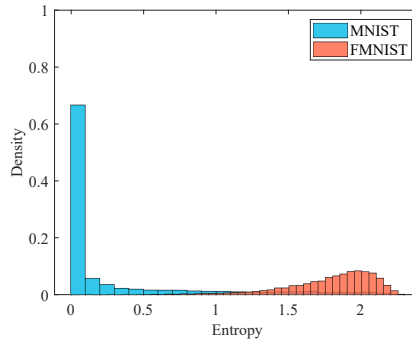
The impressive power of machine-learning models has recently been displayed by large language models such as ChatGPT [3], BARD, and models generating images such as DALL-E [16]. These successes have also revealed our lack of understanding of the reasons behind their decisions. This concern that we lack understanding of the reasons behind the decision taken by these models, in combination with the models’ immense ability to solve challenging tasks, has even led to some researchers and companies suggesting a pause on these models’ research and development [17]. To quantify the uncertainty in predicting these models could be one piece of the puzzle since it provides trustworthiness behind the decision taken by the model, which can counterbalance the lack of intuition provided in predictions from black-box models. In the second step, using the prediction from the model in a decision process, access to uncertainty in the prediction makes the decision more interpretable.

Detecting outliers and adversarial examples is another area where the knowledge of uncertainty in the prediction is crucial. Especially for NNs it has been demonstrated how vulnerable they are to adversarial attacks. For example, [18]



(a) Training data.

(b) Validation data with OOD samples.



(c) Distribution for the prediction entropy in the validation data.

Figure 1.2: Detection of OOD samples (outliers) using the uncertainty in the prediction. As training data, images of handwritten digits (MNIST) are used, while during prediction, OOD samples consisting of images of clothing items (FMNIST) are mixed with the images of the numbers. By studying the predicted entropy in Fig. 1.2c one can clearly distinguish the OOD samples. This detection capability is because they have higher entropy, meaning their predictions are more uncertain compared to images used to train the NN.

shows that with minor adjustments to the input image, an NN misclassified a stop sign as a speed limit sign. Here, uncertainty can be used to detect those adversarial examples.

### Example 1.1

Consider an example where an NN is trained to classify images of handwritten digit images from the MNIST dataset [19], see Fig. 1.2a. However, the images of the handwritten digits have been mixed up with images of clothing items from the Fashion MNIST (FMNIST) dataset [20], see Fig. 1.2b. With knowledge of uncertainty in the prediction, e.g., using the prediction entropy, the adversarial out-of-distribution (OOD) examples can be distinguished. For example, the entropy in the prediction of the FMNIST is much higher than that for images from MNIST

images. See Fig. 1.2c.

Another advantage of the knowledge of the uncertainty in the prediction is that it enables the fusion of the prediction from the NN with predictions from other sources of information. For example, modern advanced driver assistance systems (ADAS) can have many sensors, e.g., lidar, radar, and cameras, to assist the driver. Some of them usually have models based on NNs, e.g., cameras used for object detection. However, for the decision algorithm in the ADAS helping the driver to make the correct decision, information from the sensor measuring the distance from the object could be helpful to fuse with the prediction from the object detection algorithms. Fusion of the prediction could also be done over time. For example, if objects in an image sequence are supposed to be classified, some classifications might be more uncertain than others. Combining them can lead to a better understanding of the scenario. This can be illustrated by an example inspired by the classification of images in ADAS.

#### Example 1.2

Given camera images, an algorithm relying on an NN is employed to detect potential obstacles for the vehicle, enabling the ADAS to assist the driver in making well-informed decisions. For instance, consider the image sequence depicted in Fig. 1.3, where a moose suddenly appears in front of the vehicle. If the algorithm's confidence in recognizing an object exceeds a predefined threshold over several consecutive frames, the ADAS takes automatic action by reducing the vehicle's speed to prevent a collision. This threshold is represented graphically by a black line in Fig. 1.3. In this visual representation, individual predictions are indicated in red, while the fused prediction incorporates information about the uncertainty in the prediction, are shown in blue. By incorporating sequential information and utilizing the fused prediction, the ADAS can detect the presence of the moose earlier. This early detection provides the driver with more time to make a well-informed decision and take action to avoid a collision.

Hence, as described, including information regarding the uncertainty in the prediction comes with many advantages. Some of them are that knowledge of the uncertainty enables:

- (i) Detection of outliers and adversarial examples.
- (ii) Fusion of predictions from other sensors and fusion over time.

This combination gives the user a better understanding of decisions in processes of which NNs are a part. The decisions become more trustworthy since outliers can be detected, and the decisions get more robust since one can infer redundancies in the detection system where information from multiple sensors and models are combined.

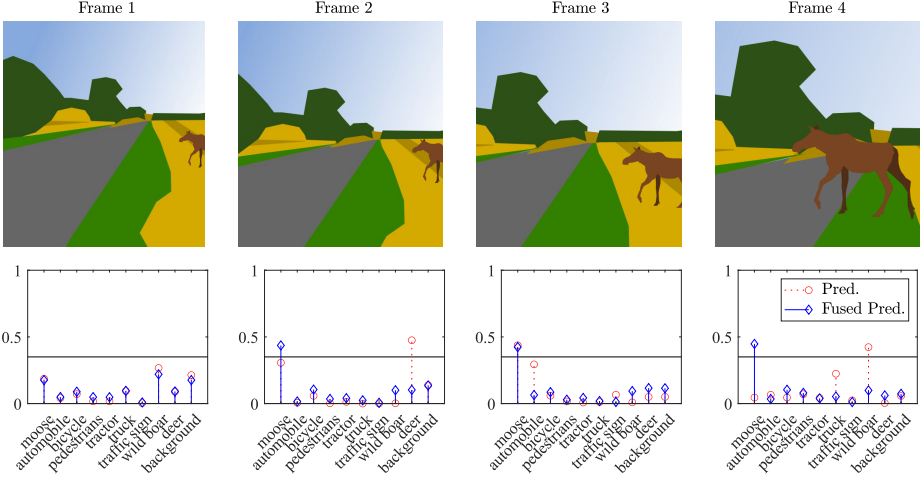


Figure 1.3: Example of a scenario where images from a dash camera are used to detect an object in front of a vehicle. In this scenario, a moose is about to cross the road. The lower part of the image shows the classification of the object in the image, where an NN does the classification. A decision process detects an object in front of the car if the confidence for a classification exceeds a threshold for two frames. In this example, it is shown that the object is detected earlier by fusing the information from earlier images in the sequence.

## 1.2 Problem formulation

The overall goal of this thesis is to develop a method that could be used to quantify the uncertainty in the prediction from NNs. An NN can be seen as a parametric function  $f(x; \theta)$ , where  $x \in \mathcal{X}$  is the input to the model and  $\theta \in \Theta$  is the parameters of the NN, i.e., the weights and biases. Given some input data  $x_n \in \mathcal{X}$  and corresponding measurements of some signal in a system  $y_n \in \mathcal{Y}$ ,  $n = 1, \dots, N$ , the NN is then trained to predict the value of future outputs  $y^*$ . Hence, this thesis aims to develop a method that could give that answer to the question “*what is the probability that the prediction from the NN is correct*”, i.e.,

$$p(y^* = \hat{y}|x^*, \theta) \quad (1.1)$$

Here  $\hat{y} = f(x^*; \hat{\theta}_N)$  is the prediction of the output for the new input,  $x^*$ , given learned parameters  $\hat{\theta}_N$  from the training of the model. The method should also be able to be used out-of-the-box for any trained NN. The focus of the thesis is:

- (i) Develop methods to quantify uncertainty in the prediction of NNs.
- (ii) Validate the method regarding when it is viable and how much the quantified uncertainty can be trusted.
- (iii) Illustrate applications where knowledge of the uncertainty can lead to more robust decisions.

## 1.3 Contributions

The seven papers included in the thesis present the main scientific contributions. This section divides the contributions into two categories.

### 1.3.1 Development and validation

One of the main contributions of the thesis is the development and validation of a method to quantify uncertainty in the predictions from NNs. The method goes under the name linearized Laplacian approximation (LLA) of Bayesian NNs. The method is based on two linearizations, one to compute the uncertainty in the parameters of the NN and one to propagate the uncertainty in the parameters to an uncertainty in the prediction. Paper A presents the regression problems and Paper D the classification problems. The method is experimentally validated, compared to other methods used to quantify the uncertainty in the prediction, and analytically validated using statistical theory. We empirically validate the method using both real-world and simulated data. For example, in Paper C, Paper D, and Paper F, we compare the method to other methods used in the literature to quantify the uncertainty in the prediction. For NNs, it is also common to use overparameterized models, i.e., to use too flexible models. Hence, Paper B investigates how overparameterization affects the quantified uncertainty from the LLA. The paper shows that the method does not underestimate the uncertainty. In Paper F, it is shown how to compute the uncertainty using a recursive formulation efficiently, and the method is extended in Paper E to represent multimodal distributions.

### 1.3.2 Robust decisions

This thesis's second contribution category is to illustrate how using quantified uncertainty in the prediction can lead to more robust and trustworthy decisions when the decision process relies on predictions from NNs. In Paper D, Paper E, and Paper F, the increased robustness is illustrated by, e.g., detecting outliers (OOD samples). Another advantage of the knowledge of uncertainty in the prediction is the inclusion of the prediction from the NN in a fusion framework. Here, combining the information from the NN's prediction with information from other sensors or a sequence of predictions is possible. A fusion framework that can handle sequential fusion and fusion from multiple sensors is developed in Paper E. Paper G extends the concept and shows how including information from previous frames leads to fewer missed detections when tracking an object in an image sequence. In the application in Paper G, the LLA is not used. Instead, the paper shows how standard object-detection algorithms implicitly include uncertainty in the prediction, allowing us to use their prediction in a fusion framework.

## 1.4 Thesis outline

The thesis is divided into two parts, with edited versions of published papers in the second part.

### 1.4.1 Part I

The first part goes through some theoretical background relevant to the uncertainty quantification of predictions from NNs relevant to the second part of the thesis. In Chapter 2, some core concepts of system identification are presented that are needed to develop the method proposed in the thesis. Chapter 3 describes the proposed method, LLA, and provides the reader with some intuition for the suggested method. Chapter 4 presents different sources of uncertainty and strategies to quantify the uncertainty in the prediction. Finally, Chapter 5 concludes the first part of the thesis by summarizing the contributions and suggesting directions for future work. Some content in this first part is based on the author's Licentiate's thesis [21].

### 1.4.2 Part II

The second part of the thesis consists of a collection of papers listed below. The contents of the paper are unchanged compared to the originals. However, some typesetting has been altered in order to comply with the format of the thesis. If not stated otherwise, the author has been the driving force in developing the papers' theory and the manuscripts' writing. The author has also done the software implementations as well as conducting the simulation experiments and processing work of the experiment data. Most of the ideas for the papers have been developed in discussions between the authors and Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. See below for a detailed comment on each publication. For the authors of the paper, the abbreviation for their names are used, i.e., the author of this thesis Magnus Malmström (MM), Isaac Skog (IS), Daniel Axehill (DA), Fredrik Gustafsson (FG), and Anton Kullberg (AK).

### Paper A

Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Asymptotic prediction error variance for feedforward neural networks. In *Proc. of 21st IFAC World Congress, (IFAC)*, Online (Berlin, Germany), 2020. Jul 11-17.

**Summary:** When using predictions from black-box models, such as NNs, in safety-critical applications, it is crucial to know how much to trust predictions from them. Here, the paper proposes a method to include uncertainty in the prediction from NNs using commonly used techniques in system identification. A simulation study evaluating the method shows how the error originating from the model mismatch and the error originating from the noise is affected by increasing the NN size.

**Background and authors' contributions:** The paper's main idea originated from FG and was further developed in discussion with MM, IS, and DA. The idea was implemented in collaboration between MM and IS, and MM wrote the manuscript with input from IS, DA, and FG.

## Paper B

Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. On the validity of using the delta method for calculating the uncertainty of the predictions from an overparameterized model. In *Proc. of 22nd IFAC World Congress, (IFAC)*, Yokohama, Japan, 2023. Jul 9-17.

**Summary:** The paper analyses how overparametization affects the uncertainty in the prediction. That is, when the method presented in Paper A (referred to as the delta method) is used to quantify the uncertainty in the prediction. Two different categories of overparametization are identified, where the uncertainty in the prediction from the overparameterized model is compared to the uncertainty in the prediction from using a canonical (minimal) model. It is shown that, for the overparameterized model, the uncertainty is larger or equal compared to the canonical model. Equality holds when the added parameters do not add flexibility to the model. Hence, for an overparameterized black-box model, the uncertainty quantified by the delta method is not underestimated. The results are shown to hold analytically and are validated using simulation experiments.

**Background and authors' contributions:** The idea of this paper originated from discussions among all the authors. MM was the driving force in formalizing the theory and designing the experiments. MM wrote the manuscript with feedback from IS, DA, and FG.

## Paper C

Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Modeling of the tire-road friction using neural networks including quantification of the prediction uncertainty. In *Proc. of IEEE 24th Int. Conf. on Inf. Fusion (FUSION)*, pages 1–6, Sun City, South Africa/Virtual, 2021. IEEE. Nov 1-4.

**Summary:** Modeling tire-road friction is important to modern cars' advanced driver assistance systems (ADAS). When modeling the friction, it is desirable to include some measure of the uncertainty in the prediction. The paper investigates modeling tire-road friction using several parametric models (both black-box and grey-box). All models include uncertainty in the prediction, where the method presented in Paper A is used to quantify the uncertainty in the prediction for the parametric black-box models. The data used in the paper comes from real-world experiments of a car braking on ice.



**Background and authors' contributions:** The idea of the paper originates from a discussion among all authors regarding how to validate the method proposed in Paper A to quantify uncertainty in prediction from NNs. MM did the processing of the data and implementation of the algorithms. MM wrote the manuscript with feedback from IS, DA, and FG.

## Paper D

Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Uncertainty quantification in neural network classifiers—a local linear approach. *arXiv preprint arXiv:2303.07114*, Under review for possible publication in *Automatica*, 2023.

**Summary:** The paper extends the proposed method from Paper A to cover classification problems. That is, proposing a method to estimate the probability mass function (PMF) and the covariance of the estimated PMF. Here, a theoretical limit of the lowest variance of the parameters, the so-called Cramér-Rao lower bound (CRLB), is derived. We are putting the method into a Bayesian setting where the prior of the parameters is included. The paper suggests a method for a proper risk assessment and a method for fusing predictions from multiple classifiers using the quantified uncertainty. The proposed method is shown to produce a correct estimate of the uncertainty. That is, the uncertainty is calibrated. The evaluation was performed on two classical image classification tasks, i.e., MNIST and CFAR10.

**Background and authors' contributions:** The idea of this paper originated from discussions among all the authors. MM has been the driving force in formalizing the theoretical results, designing the experiments, and writing the code for them. MM wrote the manuscript in collaboration with IS, DA, and FG.

## Paper E

Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Fusion framework and multimodality for the Laplacian approximation of Bayesian neural networks. *arXiv preprint arXiv:2310.08315*, Submitted for possible publication in *IEEE Trans. Aerosp. Electron. Syst.*, 2023.

**Summary:** Having quantified the uncertainty in the prediction of NNs, e.g., using the method proposed in Paper D, fusing the prediction with predictions from other models (e.g., other NNs) is possible. Here, we investigate the fusion of predictions from multiple independently trained NNs. Presented with a sequence of images that belong to the same class, the paper presents a method to fuse predictions considering the correlation over time. The fusion is for the case when using the same classifier for all the images and multiple classifiers. The paper also extends the method proposed in Paper D to represent multimodal distributions. It shows how combining multiple predictions and representing a mul-

timodal distribution improves the calibration of the uncertainty and the ability to detect OOD examples.

**Background and authors' contributions:** The idea of fusing predictions originated from discussions among all the authors, while the multimodal representation originates from MM. MM implemented the algorithms and designed the experiments. Formalizing the theory and writing the manuscript was done by MM with input from IS, DA, and FG.

## Paper F

Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Detection of outliers in classification by using quantified uncertainty in neural networks. In *Proc. of IEEE 25th Int. Conf. on Inf. Fusion (FUSION)*, Linköping, Sweden, 2022. IEEE. Jul 4-7.

**Summary:** Having a frequentist view, the method proposed in Paper A is extended to cover classification tasks. The paper suggests a recursive algorithm for updating the parameters' covariance to decrease the proposed method's computational complexity. The paper also shows how to detect images that the NN finds challenging to classify using the quantified uncertainty in the prediction.

**Background and authors' contributions:** The paper's main idea comes from MM, who is also responsible for implementing and formalizing the theory. MM wrote the manuscript with input from IS, DA, and FG.

## Paper G

Magnus Malmström, Anton Kullberg, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Extended target tracking utilizing machine-learning software – with applications to animal classification. *arXiv preprint arXiv:2310.08316, Submitted for possible publication in IEEE Signal Process. Lett.*, 2023.

**Summary:** The paper proposes a filtering framework to track objects in an image sequence over time. Here, the measurements to the filter are assumed to come from standard object-detection algorithms that detect the objects in the images, where predictions from the algorithms are used as measurements for the filter. The paper shows that including the predicted class of the object from previous frames improves the prediction of the object's class in the current frame. The proposed method is evaluated on camera trap images of Swedish carnivores.

**Background and authors' contributions:** The paper's main idea comes from MM, who is also responsible for implementing and formalizing the theory. The weight of the influence from previous images originates from discussions between MM and AK. MM wrote the manuscript with input from AK, IS, DA, and FG.

## Complete list of publications

All the work the Ph.D. student contributed to during his studies is listed below. Here, the contributions are presented chronologically. The papers included in the thesis are marked with ★.

Magnus Malmström, Isaac Skog, Sara Modarres Razavi, Yuxin Zhao, and Fredrik Gunnarsson. 5G Positioning - A Machine Learning Approach. In *Proc. of IEEE 16th Workshop on Positioning Navig. Commun.*, (WPNC), Bremen, Germany, 2019. 23–24 Oct.

★ Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Asymptotic prediction error variance for feedforward neural networks. In *Proc. of 21st IFAC World Congress, (IFAC)*, Online (Berlin, Germany), 2020. Jul 11–17.

Jacob Eek, David Gustafsson, Ludwig Hollmann, Markus Nordberg, Isaac Skog, and Magnus Malmström. A novel and fast approach for reconstructing CASSI-raman spectra using generative adversarial networks. In *2022 11th Int. Conf. on Image Proc. Theory, Tools and App. (IPTA)*, pages 1–6, Salzburg, Austria, 2022. IEEE. Apr 19–22.

★ Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Modeling of the tire-road friction using neural networks including quantification of the prediction uncertainty. In *Proc. of IEEE 24th Int. Conf. on Inf. Fusion (FUSION)*, pages 1–6, Sun City, South Africa/Virtual, 2021. IEEE. Nov 1–4.

★ Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Detection of outliers in classification by using quantified uncertainty in neural networks. In *Proc. of IEEE 25th Int. Conf. on Inf. Fusion (FUSION)*, Linköping, Sweden, 2022. IEEE. Jul 4–7.

★ Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. On the validity of using the delta method for calculating the uncertainty of the predictions from an overparameterized model. In *Proc. of 22nd IFAC World Congress, (IFAC)*, Yokohama, Japan, 2023. Jul 9–17.

★ Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Uncertainty quantification in neural network classifiers—a local linear approach. *arXiv preprint arXiv:2303.07114*, Under review for possible publication in *Automatica*, 2023.

★ Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Fusion framework and multimodality for the Laplacian approximation of Bayesian neural networks. *arXiv preprint arXiv:2310.08315*, Submitted for possible publication in *IEEE Trans. Aerosp. Electron. Syst.*, 2023.

★ Magnus Malmström, Anton Kullberg, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Extended target tracking utilizing machine-learning software – with applications to animal classification. *arXiv preprint arXiv:2310.08316, Submitted for possible publication in IEEE Signal Process. Lett.*, 2023.

# 2

---

## System identification of neural networks

In this chapter, training an NN is formulated as a problem of estimating a parametric model from measurements. This formulation will make it possible to use tools from estimation theory and system identification theory to develop a method to quantify the uncertainty of the prediction made by NNs. The concepts of model structure, model sets, the notion of a true system, uniqueness, and identifiability will be reviewed. Furthermore, the chapter discusses the model selection problem and will conclude with some remarks regarding overparameterization. Even though the concepts presented in this chapter are valid for general models, the special case of the model based on NN is emphasized.

### 2.1 Problem formulation

Consider the problem of learning a model to describe the relationship between some input (independent variables)  $x \in \mathcal{X}$  and measurements of the output (dependent variable)  $y \in \mathcal{Y}$ , given some training data

$$\mathcal{T} \triangleq \{y_u, x_n\}_{n=1}^N. \quad (2.1)$$

Here,  $N$  is the number of training data points. The measurements  $y$  come from some underlying *true system*  $\mathcal{S}$  under consideration, and it is assumed that some function  $f^*(x)$  describes the true relationship to be modeled. In this thesis, the focus is on using parametric models to model this relationship, i.e.,  $f(x, \theta)$  where  $\theta \in \Theta \subset \mathbb{R}^{n_\theta}$  contains  $n_\theta$  parameters. The thesis will focus on using NNs, but the analysis is valid for most parametric models.

The parameters are learned, a.k.a. estimated, to learn the model which best describes the data by minimizing some loss function  $L_N(\theta)$ , i.e.,

$$\hat{\theta}_N = \arg \min_{\theta} L_N(\theta). \quad (2.2)$$

If the loss function is chosen according to some statistical properties of the measurements, the loss function becomes the (log) likelihood. The loss function is sometimes also referred to as the cost function. The input will be assumed to be real numbers throughout the thesis, i.e.,  $\mathcal{X} \subset \mathbb{R}^{n_x}$ . For the output, mainly two cases will be considered, either  $\mathcal{Y} \subset \mathbb{R}^{n_y}$  where  $n_y$  is the dimension of the output, i.e., a regression problem, or  $\mathcal{Y} = \{1, \dots, M\}$ , i.e., a classification problem with  $M$  classes. For the regression problem the loss function is given as

$$L_N(\theta) \triangleq \frac{1}{N} \sum_{n=1}^N \|y_n - f(x_n; \theta)\|^2, \quad (2.3)$$

i.e., the mean squared error loss, and for the classification problem as

$$L_N(\theta) \triangleq \sum_{n=1}^N \ln f_{y_n}(x_n; \theta). \quad (2.4)$$

i.e., the cross-entropy loss function. Here, the subindex  $n$  denotes the  $n$ 'th training data point, and  $y_n$  in (2.4) is used as an index operator for the subscript  $m$  of  $f_m(x; \theta)$ . For the regression problem in (2.3), the statistical assumption is that the true system is measured with additive observation noise  $e$ , which is identically and independently distributed according to some distribution with variance  $\lambda_0$  and mean  $E[e] = 0$ . The statistical assumption in (2.4) is that the measurements are categorically distributed.

The parametric model, together with the estimated parameters, is then used to predict a new output  $\hat{y} = f(x^*; \hat{\theta}_N)$  of the system given some input  $x^*$ . In this thesis, the aim is to analyze how to quantify the uncertainty in the prediction, i.e., to compute

$$p(\hat{y} = y^* | x^*, \theta), \quad (2.5)$$

where  $y^*$  denotes the true output for the input  $x^*$ .

## 2.2 Model sets and model structure

To carry out the analysis of the prediction-error variance and put the estimation of the parametric model  $f(x; \theta)$  in a system identification framework, the concept of models, model structure, model sets, and true system used in, e.g., [31], need to be introduced and formalized. Thus, a *prediction model* is defined as:

**Definition 2.1 (Prediction model).** A prediction model  $f(x_{1:m}, y_{1:m-1})$  predicts the current output of a system given information about current and past inputs as well as past outputs.

The identification problem is to find a suitable model that can describe the measured data as well as possible. A search over a set of candidate models is conducted in order to find this, which leads us to the natural definition of *model set* as:

**Definition 2.2 (Model set).** A model set  $\mathcal{M}^*$  is a collection of candidate models.

**Example 2.3**

One example of a model set is, e.g., the set of all linear models,

$$\mathcal{M}^* = \{\text{all linear models}\}. \quad (2.6a)$$

Another example of a model set is the set of all fully connected NNs,

$$\mathcal{M}^* = \{\text{fully connected NNs}\}. \quad (2.6b)$$

A final example is a finite set of specific models

$$\mathcal{M}^* = \{f_1(x_{1:m}, y_{1:m-1}), \quad f_2(x_{1:m}, y_{1:m-1}), \quad f_3(x_{1:m}, y_{1:m-1})\}. \quad (2.6c)$$

Most model sets of interest contain an infinite number of models. Hence, an exhaustive search over all those model sets is intractable. Instead, the search is usually done over a subset of these models. This subset is often constructed by parameterizing the set “smoothly” over some “nice” areas, which refers to the fact that the model is differentiable with respect to the parameters and that the parameters come from an open set. This could be done by restricting the parametric model  $f(x; \theta)$  to have a limited number of parameters  $n_\theta$  and that the gradient of the model with respect to the parameters should be well defined, i.e., the model should be differentiable with respect to the parameters [31]. This gives the formal definition of *model structure* as:

**Definition 2.4 (Model structure).** A model structure  $\mathcal{M}$  is a differential mapping from a connected open subset  $\Theta$  of  $\mathbb{R}$  to a model set  $\mathcal{M}^*$ , such that the gradient of the predictor function is smooth. That is

$$\mathcal{M}: \theta \in \Theta \rightarrow \mathcal{M}(\theta) = f(x; \theta) \in \mathcal{M}^*, \quad (2.7)$$

where  $\mathcal{M}(\theta)$  is one model in the model structure.

Knowing if the model set includes  $\mathcal{S}$  is often interesting. That is, if there exists a  $\theta^0$  such that  $\mathcal{M}(\theta^0) = \mathcal{S}$ , which is denoted  $\mathcal{S} \in \mathcal{M}^*$ . This assumption is hard to fulfill in practice, but the concept is theoretically important when estimating the models [31].

### 2.2.1 Neural networks

In many cases, models that are linear in the parameters are not flexible enough to represent the given data. For more flexible models, more complex model sets need to be used [32]. An example of a model set with more complex models is the set containing NNs. The recursions below can describe one model in this set.

$$h^{(0)} = x, \quad (2.8a)$$

$$a^{(l+1)} = \left( h^{(l)} \quad 1 \right)^\top W^{(l)}, \quad l = 0, \dots, L, \quad (2.8b)$$

$$h^{(l)} = \sigma(a^{(l)}), \quad l = 1, \dots, L, \quad (2.8c)$$

$$y = a^{(L)}. \quad (2.8d)$$

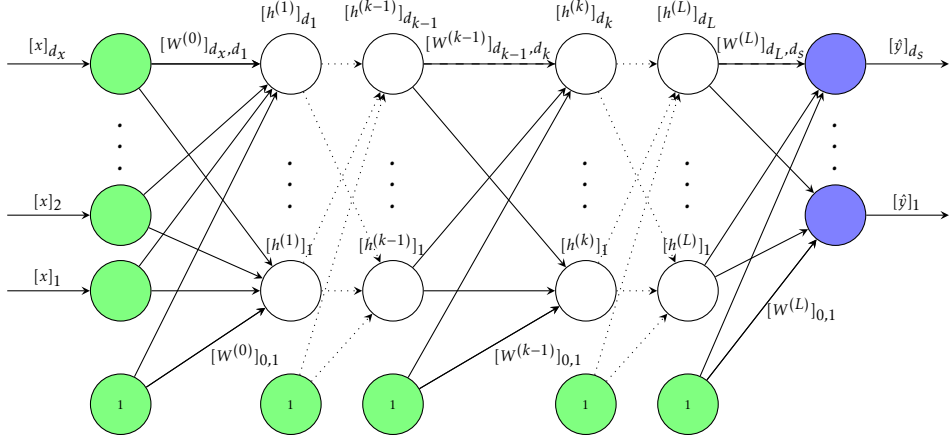


Figure 2.1: Schematic illustration of a fully-connected NN with  $n_x$  inputs,  $n_y$  outputs, and  $L$ -layers where every layer has  $d_l$  nodes  $l = 1, \dots, L$ . The bias term in hidden layers (nodes marked with the number 1), as well as the input in the NN are indicated with green nodes. The white nodes apply the nonlinear activation function  $\sigma(\cdot)$  to the sum of all inputs. The blue nodes are the output nodes which sums all the inputs to them.

The number of layers in the network is denoted  $L$ ,  $W^{(l)}$  are the weights of the  $l$ 'th layer,  $a^{(l)}$  is the contribution from layer  $(l-1)$  to layer  $l$  and  $h^{(l)}$  is the activation of the contribution from the  $(l-1)$ 'th layer at the  $l$ 'th layer, referred to as the *hidden node*. In Fig. 2.1, a schematic illustration of a fully-connected NN with  $n_x$  inputs,  $n_y$  outputs, and  $d_l$  nodes in the  $l$ 'th hidden layer. This gives that the dimension of  $W^{(l)}$  is  $(d_{l-1} + 1) \times d_l$ .

Furthermore,  $\sigma(\cdot)$  is a so-called activation function operating element-wise. Some of the most commonly used activation functions are the sigmoid function  $\sigma(u) = 1/(1 + e^{-u})$ , the hyperbolic tangent  $\sigma(u) = \tanh(u)$ , the rectified linear unit (ReLU)  $\sigma(u) = \max\{u, 0\}$ , the leaky ReLU  $\sigma(u) = \max\{u, 0.1u\}$ , and the exponential linear unit (ELU)

$$\sigma(u) = \begin{cases} u, & u \geq 0, \\ \alpha(e^u - 1), & u < 0, \end{cases} \quad (2.9)$$

where  $\alpha$  is a hyperparameter that the user can choose. In Fig. 2.2, these activation functions and their derivatives are shown.

Collecting the biases and the weights of the NN into a parameter vector, i.e.

$$\theta = \{W^{(l)}\}_{l=0}^L. \quad (2.10)$$

an NN can be written as a parametric model,  $f(x; \theta)$ .



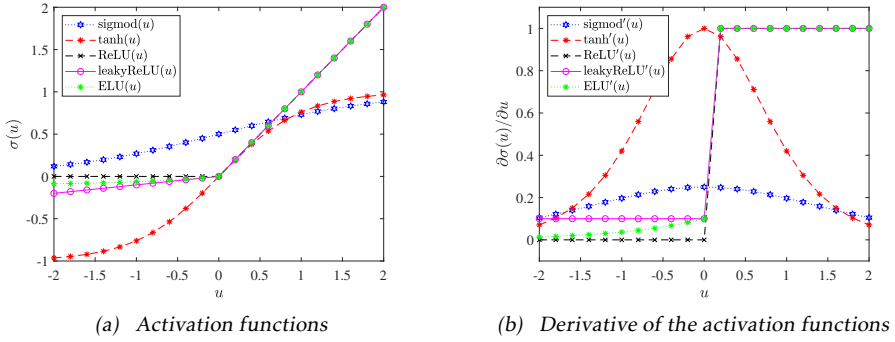


Figure 2.2: Some of the most commonly used activation functions together with their derivatives.

### 2.2.2 Model-order selection

In system identification, an important aspect is to select the size of the model set under consideration, i.e., choosing the number of model parameters based on the trade-off between flexibility and overparameterization. Overparameterization refers to using a model with unnecessarily many parameters, i.e., a too-flexible model. Selecting the correct number of model parameters is also called model-order selection or the order-selection problem [31]. For models that are linear in the parameters, such as polynomials, selecting the correct model-order is equivalent to selecting the number of basis functions. For NNs, the model-order depends on the number of hidden layers and nodes in those hidden layers.

A commonly used approach to select the model-order is to study the loss function  $L_N(\theta)$  for different choices of the number of parameters  $n_\theta$ . Evaluated on training data, it is natural that the loss function decreases as the number of parameters increases since the model is more flexible. However, this might not be desirable since models with a larger number of parameters may suffer from overparameterization and higher variance in their prediction. Instead, it is possible to study a version of the loss function with a penalty factor for using models with higher order. Two classical penalty functions are Akaike's information criterion (AIC) suggested by [33] and Bayesian information criterion (BIC) suggested by [34]. In the case of Gaussian observation noise, AIC and BIC are given by

$$L_N(\hat{\theta}_N) \left( 1 + \frac{2d}{N} \right), \quad (2.11a)$$

and

$$L_N(\hat{\theta}_N) \left( 1 + \frac{d \log N}{N} \right), \quad (2.11b)$$

respectively.

A related approach to model-order selection is the so-called effective number of parameters [35]. Here, instead of talking about the number of used parameters in the model, one is concerned with which combinations of parameters are most important. One approach to control the number of effective parameters is to use so-called  $l_2$  regularizations where the two norms of the size of the parameters are penalized in the loss function (2.3) and (2.4).

## 2.3 Uniqueness

Due to the ambiguities in the model structure, the choice of parameters for NN is not unique. This ambiguity comes from symmetries in the NN model structure and the activation function, as well as possible overparameterizations in terms of nodes needed to describe the true input-output relationship; see [36] and [37] for details. Hence, the parameter estimate  $\hat{\theta}_N$  is non-unique and will depend on factors such as the training data realization, the initial condition, and the choice of optimization algorithm and its implementation. Let us define the set of parameter vectors that minimize the loss function (2b) as

$$\mathcal{D}_{\hat{\theta}_N} \triangleq \{\hat{\theta}_N \in \Theta : \hat{\theta}_N = \arg \min_{\theta} L_N(\theta)\}. \quad (2.12)$$

Depending on whether the true model belongs to the chosen model set  $\mathcal{M}^*$  or not, several different sources contributing to the prediction uncertainty will be considered. If  $f^* = S \in \mathcal{M}^*$  there exists a non-empty parameter set

$$\mathcal{D}_{\theta^0} \triangleq \{\theta^0 \in \Theta : f(x; \theta^0) = f^*(x) \quad \forall x\}, \quad (2.13)$$

of parameter vectors where  $\theta^{i0} \in \mathcal{D}_{\theta^0}$  is the  $i$ 'th parameter vector such that the NN model describes the input-output relationship perfectly. Otherwise, if  $S \notin \mathcal{M}$ , from [31], it is known that the parameter estimate converges to a vector  $\theta^{i*}$  that gives the best fit of the training data in the least-squares sense. That is,  $\theta^{i*} \in \mathcal{D}_{\theta^*}$ , where

$$\mathcal{D}_{\theta^*} \triangleq \{\theta^* \in \Theta : \theta^* = \lim_{N \rightarrow \infty} \arg \min_{\theta} L_N(\theta)\}. \quad (2.14)$$

If  $S \in \mathcal{M}^*$ , then  $\mathcal{D}_{\theta^*} \equiv \mathcal{D}_{\theta^0}$  and the parameter estimate converges to one vector in this set  $\mathcal{D}_{\theta^0}$ .

### 2.3.1 Canonical representation of a model

A canonical model is an irreducible, or minimal, representation of the model if there is no model with fewer parameters that can represent the given data, as well as the canonical model. A related concept is redundant parameters, i.e., parameters that do not add flexibility to the model. If a model has redundant parameters, it is also reducible.

To handle the fact that the choice of parameters in the NN, in general, is non-unique and hence not globally identifiable, see Def. 2.10, a canonical representation of a parametric model  $f_c : \mathbb{R}^{n_x} \times \mathbb{R}^{n_{\theta_c}} \rightarrow \mathbb{R}^{n_y}$  with a corresponding canonical, i.e., unique and irreducible, parameter vector  $\theta^c \in \mathbb{R}^{d_c}$  is introduced. Here, the number of parameters in the canonical representation  $n_{\theta_c}$  is less or equal to the number of parameters in the non-canonical representation of the model  $n_{\theta}$ ,  $n_{\theta_c} \leq n_{\theta}$ . For the canonical representation of a parametric model, it holds that given any  $\theta$ , there exists a unique  $\theta_c$  such that

$$f(x; \theta) = f_c(x; \theta_c), \quad \forall x. \quad (2.15)$$

Hence,  $f_c \in \mathcal{M}^*$  can represent any input-output relation that  $f \in \mathcal{M}^*$  can, but the parameterization is assumed unique and potentially of lower dimension. Furthermore, assume that there exist  $k$  differentiable mappings  $T_i$ ,  $i = 1, \dots, k$ , relating any parameter vector  $\theta^i$  in the original representation of the model and the corresponding one  $\theta_c$  in the canonical representation of the model such that  $\theta_c = T_i(\theta^i)$ .

There could be the case that multiple choices of  $\theta^i$  result in irreducible models, i.e., models with the same number of parameters that give the same prediction. In that case, any of those models could be considered the canonical representation of the model. Therefore, it is a choice of which of these irreducible models should be the canonical representation of the model and which parameters  $\theta^i$  should be considered to be  $\theta_c$ . After a choice of which realization of the model that should be considered canonical has been made, one could consider  $\theta_c$  unique and relate them to the other irreducible parameters by some mapping  $T_i$ .

### 2.3.2 Symmetries in neural networks

In [36], it is shown that for a two-layer NN with sigmoid activation functions, there exist only two kinds of transformations on  $\theta_c$  such that  $\theta_c = T_i(\theta^i)$ , namely:

- (i) Interchange of nodes at the same layer, i.e., interchange  $h_i^{(l)}$  and  $h_j^{(l)}$  with each other.
- (ii) Change of sign of the parameters due to symmetries in the activation function, i.e.,  $W_{j,k}^{i(l)} = -W_{j,k}^{c(l)}$  and  $W_{\cdot,j}^{i(l-1)} = -W_{\cdot,j}^{c(l-1)}$ . For sigmoid activation functions, which are symmetric around 0.5, the sign change will also result in a move of information to the bias term. This added information could be described as  $W_{0,k}^{i(l)} = W_{0,k}^{c(l)} + W_{j,k}^{c(l)}$ . For activation functions symmetric around zero, e.g., tanh, moving any information to the bias term is unnecessary. See Fig. 2.2a for illustrations of the activation functions.

These two transformations for a two-layer NN with a sigmoid as activation function will generate a family with  $2^{d_1} d_1!$  elements. See [36, 38] for more details of constructing such a mapping in an irreducible two-layer NN with sigmoid activation functions.

*Remark 2.5.* One can notice that these transformations are linear, i.e.,  $\theta_c = T_i \theta^i$ . Hence, the parameters for all irreducible NNs, i.e., NNs that have the same number of parameters as the canonical representation, can be written as a linear combination of the parameters of the canonical representation.

### Example 2.6

To illustrate these symmetries, consider an example where measurements are generated by the model

$$f^*(x) = D \sin \left( C \arctan \left( (1 - E)x + E/B \arctan(Bx) \right) \right). \quad (2.16)$$

The signal in (2.16) describes how the normalized traction force (NTF) depends on the wheel slip,  $x$ . The parameters  $B$ ,  $C$ ,  $D$ , and  $E$  depend on the surface. Here they are chosen as  $B = 14$ ,  $C = 1.6$ ,  $D = 0.6$ , and  $E = -0.2$ . For further description of the tire-road friction model, see, e.g., [39, 40] and Paper C.

Given some measurement of (2.16), consider the problem in (2.3) where  $f(x; \theta)$  is a two-layer NN, i.e.,

$$f(x; \theta) = \theta_5 \sigma(\theta_1 x + \theta_3) + \theta_6 \sigma(\theta_2 x + \theta_4) + \theta_7. \quad (2.17)$$

See Fig. 2.3 for a schematic illustration of such a model. The signal generated by (2.16) can be modeled exactly using a model such as in (2.17), see Paper A. However, due to the symmetries in the NN, there is no unique parametrization. The parameters are found by training the NN using the ADAM optimizer, [41], with the standard settings for the hyperparameters, initializing the parameters using the Xavier initialization, [42]. The experiment is repeated 200 times for different parameter initializations and data separations into training batches. The batches are subsets of training-data, using one batch at a time in the optimization algorithm. Define the canonical representation of the model by

$$f_c(x; \theta_c) = 0.6 \sigma(-6.8x + 0.0064) - 1.4 \sigma(-40x + 0.0039) + 0.42, \quad (2.18)$$

where the other values of the parameters minimizing (2.3) are referred to as the non-canonical representation of the model.<sup>1</sup>

In Fig. 2.4a, the two parameters connecting the hidden layer with the output layer, i.e.,  $\theta_5 = W_{11}^{(1)}$  and  $\theta_6 = W_{12}^{(1)}$ , are plotted for the different realizations. One can see that the parameters converge to some finite sets of different values, i.e., eight different clusters. The number of clusters coincides with the predicted number of the clusters for a two-layer NN with  $d_1 = 2$  presented above in this section. However, in Fig. 2.4a, it can be seen that the path the optimization solver takes from the initialization of the parameters to the optimal parameters looks far from direct. Even though a parameter is initialized close to a specific cluster, it might converge to another cluster, e.g., the initialization near the red cluster converges to the blue cluster, see Fig. 2.4b.

<sup>1</sup>Notice that since the canonical representation of the model and non-canonical representation of the model have the same size, there are no redundant parameters in the non-canonical model, i.e., all are irreducible.

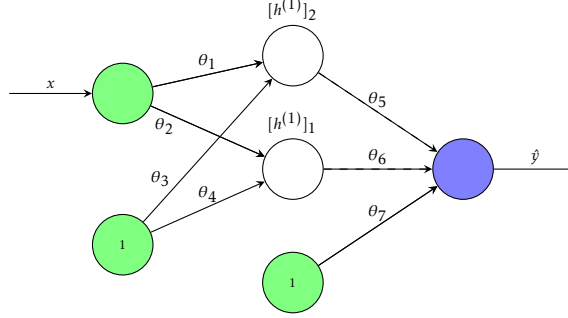


Figure 2.3: Schematic illustration of a fully-connected NN with one input, one output, and one hidden layer with two nodes in the hidden layer. The bias term in the hidden layers (nodes marked with the number 1), as well as the input in the NN are indicated with green nodes. The white nodes apply the nonlinear activation function  $\sigma(\cdot)$  to the sum of all inputs. The blue node is the output node which sums all its input.

The prediction is equally good for all the clusters, so a natural next question is if the uncertainty in the parameters for the different clusters is different. Paper B further investigates this question. Next, it is shown how a mapping between the different clusters can be constructed.

#### Example 2.7

Returning to the setup in Ex. 2.6, one can compare the values of the canonical realization of the parameters (in the red cluster) given in (2.18) and a realization from the light blue cluster given by

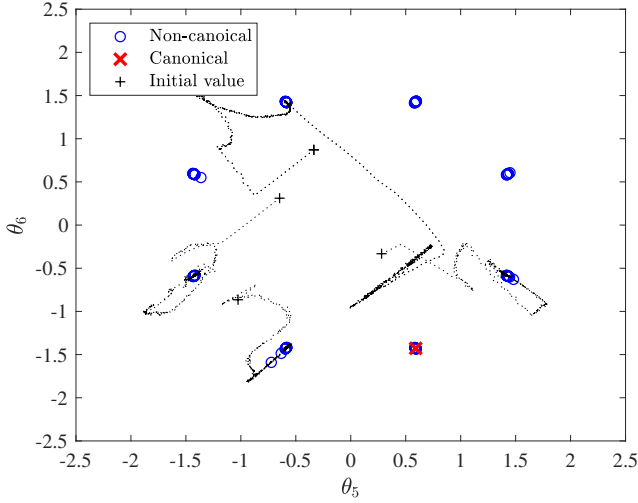
$$f(x; \theta^i) = -1.4\sigma(-40x + 0.0061) + 0.60\sigma(-6.8x + 0.0036) + 0.42. \quad (2.19)$$

It is possible to see that the parameters are related via the transformation given by

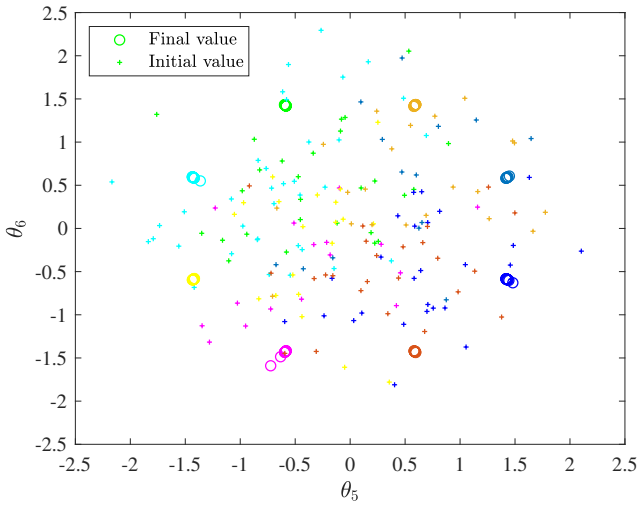
$$T_i = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad (2.20)$$

i.e., interchanging the nodes in the same layer corresponding to similar operations (bias or linear combination) with each other, which is the first sort of transformation described earlier in this section.

Noteworthy is that the somewhat abstract function  $T_i$  is only needed for the forthcoming analysis of the suggested method in Paper A and not for the application of the discussed methods, which is shown in Paper B.



(a) Parameters for different realizations. For a couple of realizations, the path of intermediate iterations from the initial parameter values to the optimum is visualized.



(b) Clusters of the parameters for different realizations. The final value and the initialization of the parameters are color-coded such that realizations that converge to the same cluster have the same color.

Figure 2.4: Illustration of how symmetry in the NN can lead to different optimal parameters for different realizations. The parameters are clustered in eight clusters. Here, the parameters connecting the hidden layer to the output layer are visualized, i.e.,  $\theta_5$  and  $\theta_6$ , where the numbering of the parameters can be seen in Fig. 2.3 and (2.17).

## 2.4 Regularity conditions and identifiability

To pursue the analysis of the prediction error variance, some regularity conditions regarding both the observed data generated by the true system and the smoothness of the models in the model structure and the loss function need to be imposed.

### 2.4.1 Identifiability and informativity

Given a training-data set, the user chooses a model set to estimate a model. However, the estimator might not converge, or the model might not be reasonable. That the estimate does not converge can be a consequence of both the choice of model structure for which the search is conducted and the consequence of that the data set is not informative enough. Here, we are using the definition of informativity from [31] and [43], but modified for the static systems.

**Definition 2.8 (Informativity).** A data set,  $x_{1:N}$  is informative enough with respect to a model set  $\mathcal{M}^*$  if for any two models  $f(x; \theta_1)$  and  $f(x; \theta_2)$  in the model set

$$f(x; \theta_1) - f(x; \theta_2) = 0, \quad \forall x \implies \theta_1 = \theta_2. \quad (2.21)$$

#### Example 2.9

Consider the model set of polynomials of degree  $n$ , which is an example of a nonlinear static model. For the data to be informative enough, at least  $n + 1$  unique data points are needed.

For further discussions on informativity, see, e.g., [43]. Throughout this thesis, it will be assumed that the generated training-data is informative enough such that all models in the true system have been persistently excited.

Considering the chosen model structure, one has to make sure that the model structure is globally identifiable, i.e., two choices of parameters  $\theta$  will result in a different output from the model. First, let us define the concept of (local) identifiability of a model using the uniqueness-oriented definition suggested in [31].

**Definition 2.10 (Identifiability).** A parametric model structure is locally identifiable at value  $\theta^*$  if  $\exists \delta > 0$  such that for all  $\|\theta - \theta^*\| < \delta$

$$f(\cdot; \theta^*) = f(\cdot; \theta) \implies \theta = \theta^*. \quad (2.22)$$

For a model structure to be globally identifiable at  $\theta^*$ , the same must hold as  $\delta \rightarrow \infty$ , i.e., it has to be locally identifiable at all  $\theta^*$ .

Taking the smoothness of the models in the model sets into consideration, the following assumption from [44] is needed.

**Assumption 2.11.** Let  $\Theta$  be compact, and assume that the model is three times continuously differentiable. Assume as well that both the model and its derivatives, component-wise, are bounded. For the loss function, it will be assumed that it is three times differentiable with respect to the parameters and the prediction error and the norm of derivatives and cross-derivatives are bounded, as described in [44].

These assumptions are needed to show that the discussed estimators are consistent, which is used in Paper A to show the asymptotic results of the estimator.

## 2.4.2 Implications for neural networks

Due to the symmetries in NNs, the model structure will not be globally identifiable with a global minimizer to (2.2), but rather to a set of local minimizers where each one is locally identifiable. The fact that the model is not globally identifiable is handled by the transformation  $T_i$  between the local minima, introduced in Sec. 2.3.2.

In [36], it is also shown that apart from the transformations generated by  $T_i$  given in Sec. 2.3.2, the parameter  $\theta$  is globally identifiable for an irreducible two-layer NN. This holds for an NN where the activation function  $\sigma(\cdot)$  is such that the class of functions  $\{\sigma(ax + b), a > 0\} \cup \{\sigma \equiv 1\}$  is linearly independent. One can interpret this condition as the contributions from all the hidden nodes in  $h^{(1)}$  are different and not constant.

## 2.5 Overparameterization

As stated in Sec. 2.2.2, an essential problem in system identification is selecting the correct model-order. This is a compromise such that the model is flexible enough to represent the underlying true system but not too flexible such that it models the noise. Selecting the correct model-order is especially difficult when using NNs. However, what is often done in practice is to use already established structures that are flexible enough to model the true system, e.g., GoogLeNet [45], AlexNet [1], Resnet [46], LeNet [47]. The use of standard structures leads to the fact that the models are often overparameterized. Hence, when introducing a method to quantify the uncertainty in the prediction from NNs, it is important to determine how overparameterization affects the uncertainty. Overparameterization is the topic of Paper B. For the method proposed in the thesis, the uncertainty is not overestimated compared to the canonical (minimal) representation needed to model the true system. This result is under the assumption that there are more training-data points than parameters ( $N \gg n_\theta$ ). However, this might not be true for NNs, as is shown by the so-called double descent.

In statistical modeling, it is well known that as the model-order ( $n_\theta$ ) increases, the uncertainty in the prediction on validation data decreases until a certain point where the model interpolates the training-data, i.e., overfits the training-data, and as a consequence the uncertainty on validation data increases. However, NNs seem to break this known fact since when the model-order increases even further,



the uncertainty on validation data decreases again, in a second minimum, the so-called double descent [48]. Some approaches suggested in the literature try to explain this phenomenon. For example, the commonly used activation functions such as ReLU, combined with using weight decay during NN training, have a regularizing effect, similar to penalizing the size of the parameters [49]. Another explanation of the double descent is that as the model-order increases beyond the number of points in the training-data set,  $N < n_\theta$ , an increase in the model-order leads to a decrease in the norm of the parameters in the model, i.e., also a regularizing effect [50]. This regularizing effect prevents the NN from overfitting to the training data.



# 3

---

## Linearized Laplacian approximation

This chapter provides some background theory for the linearized Laplacian approximation (LLA). The LLA is the proposed method used throughout the thesis to quantify the uncertainty in the predictions from NNs. It is a method to approximate any trained NN as a Bayesian NN (BNN). In other words, put the prediction from an NN into a context that naturally includes the uncertainty in the prediction. In short, LLA can be described in three steps:

- (i) Train the model to find  $\hat{\theta}_N$  by minimizing the loss function, e.g., (2.3) and (2.4).
- (ii) Compute the covariance  $P_\theta$  of the parameters of the NN by linearizing the loss function. That is, the Laplacian approximation of a BNN.
- (iii) Propagate the uncertainty of the parameters to uncertainty in the prediction by linearizing the NN with respect to the parameters. This linearization is referred to as the delta method.

This chapter covers the last two steps. For the first step, interested readers are referred to classical books such as [32, 51].

### 3.1 Laplacian approximation

In practice, training a BNN can be computationally expensive, if not practically impossible. Approximating an already trained NN as a BNN is an approach still to gain the advantages of including uncertainty in the prediction while still not explicitly learning the uncertainty in the prediction. One such approximation is the so-called Laplacian approximation of a BNN. The idea of the Laplacian approximation is to assume that the parameters are Gaussian distributed with

the mean given by the values of the parameters and the covariance given by the curvature of the loss landscape. That is, for a model  $f(x; \theta)$ ,

$$\theta \sim \mathcal{N}(\hat{\theta}_N, P_N^\theta). \quad (3.1)$$

The Laplacian approximation per se is a well-known method to approximate the posterior distribution with a Gaussian distribution. Here, the mean is given by the maximum a posteriori estimate and the covariance by the inverse of the Fisher information matrix [52]. The method is motivated by Bernstein-von Mises theorem, which states that, under some regularity conditions, the posterior converges in the limit of infinite data to a multivariate normal distribution [53]. The intuition of the approximation is to approximate the loss function with a quadratic one, i.e., to approximate the problem as a linear regression problem. Another motivation of the method is that sufficiently close to a (local) minimum, a linear model can approximate a nonlinear model well. The method is also commonly used for NNs and was early introduced to approximate BNNs [54]. However, the question is, how accurate is it to approximate the loss function as a quadratic one?

---

### Example 3.1

---

Consider the model

$$f(x; \theta) = \tanh(\theta_2 x + \theta_1). \quad (3.2)$$

The model (3.2) is an NN consisting of one node and one hidden layer. Assume that the true system under consideration has the same structure as (3.2) with  $\theta_2 = 1$  and  $\theta_1 = -1$ . Given some measurements of the true system, the mean squared error (2.3) is minimized between the measurements and a model with structure as in (3.2). For simplicity, assume that the optimization algorithm has found the true parameters, i.e.,  $\hat{\theta}_N = \theta^0$ . The loss function can then be linearized around the estimated parameters  $\hat{\theta}_N$ . That is

$$\tilde{f}(x; \theta) = f(x; \theta) + \varphi^\top(x)(\hat{\theta}_N - \theta) \quad (3.3a)$$

$$\varphi(x) = \begin{bmatrix} 1 - \tanh^2([\hat{\theta}_N]_2 x + [\hat{\theta}_N]_1) & x(1 - \tanh^2([\hat{\theta}_N]_2 x + [\hat{\theta}_N]_1)) \end{bmatrix}. \quad (3.3b)$$

Fig. 3.1 depicts the loss function  $L_N(\theta)$  for the nonlinear model  $f(x; \theta)$  and the linearized model  $\tilde{f}(x; \theta)$ . Fig. 3.1a shows the one-dimensional loss function where  $\theta_2 = \theta_2^0$ , while Fig. 3.1b shows the two-dimensional loss function. Here, one can see that the linear approximation is valid in the proximity of the true parameters. However, the approximation worsens when moving away from the true parameters. It is the curvature of the loss function using the linearized model, which, around the current parameter estimate, gives the uncertainty of the parameters in the Laplacian approximation.

---

Ex. 3.1 illustrates some limitations of using the Laplacian approximation. These limitations are that it is a local approximation that might not be accurate too far away from the linearization point. The method also requires that the optimization method used to train the model has found a local minimum of the loss function.

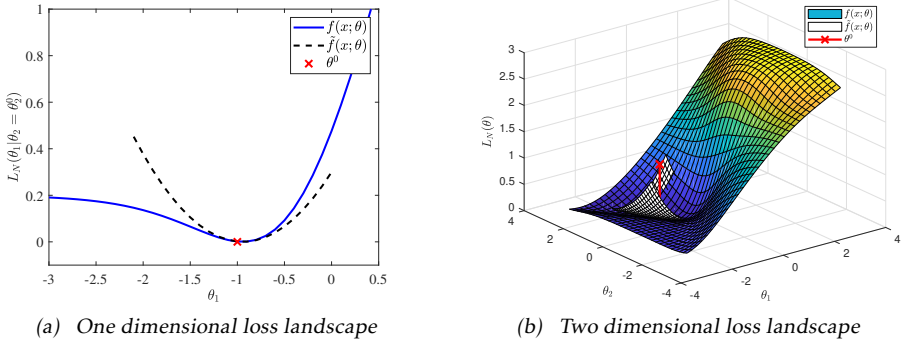


Figure 3.1: Visualization of the loss landscape for a nonlinear model and the quadratic approximation of that loss function using a linearized model. This visualization illustrates the curvature of the loss function using the linearized model, which gives the covariance of the parameters in the Laplacian approximation.

### 3.1.1 Hessian of the loss function

From Bernstein-von Mises theorem [53], if the true model belongs to the considered model set, the maximum a posteriori estimate  $\hat{\theta}_N$  converges in distribution to

$$\hat{\theta}_N \xrightarrow{d} \mathcal{N}(\hat{\theta}_N; \theta^0, \mathcal{I}_\theta^{-1}), \quad (3.4)$$

when the information in the training data  $\mathcal{T}$  tends to infinity. The same holds for the maximum likelihood estimate when the true model belongs to the considered model set [31]. Here,  $\theta^0$  denotes the true parameters and  $\mathcal{I}_\theta$  the Fisher information matrix. When choosing the loss function as the likelihood function, Paper A and Paper D show that the inverse of the Fisher information matrix gives the covariance of the Laplacian approximation. In Paper A, the mean squared loss function (2.3) is considered while in Paper D considers the cross entropy loss (2.4).

In optimization theory, which the algorithms used to train the NN are based on, a central concept is the Hessian of the loss function [55]. That is the second derivative of the loss function with respect to the decision variables, e.g., here are the parameters. Close to the optimum, the direction in which the Hessian of the loss function is the largest is the direction in which the parameters' values are the most certain, i.e., the covariance of the parameters is the lowest. Hence, a close connection exists between the Hessian of the loss function and the inverse of the Fisher information matrix [31, 56–58].

However, in practice for optimization methods, it is common to use an approximation of the Hessian [55]. One reason to use the approximations is because the cost of computing the true Hessian is high compared to the gained performance. This is more evident when compared to the performance using multiple steps with a slightly worse stepsize, which can take less computational resources.

Here, the worse stepsize is a consequence of using an approximation of the Hessian, but using the approximation is also the reason for the lower computational cost. This reduction in computation time is especially true if the model has many parameters, for instance, as an NN have. The lower computational cost is one of the reasons why two of the most used optimization (training) algorithms for NNs, RMS-prop and ADAM-optimizer, rely on an approximation of the Hessian [41, 59].

One common approach to approximate the Hessian of the loss function is by a quadratic approximation, which is, e.g., used in the Gauss-Newton method. The motivation for using this approximation is that close to the optimum, the nonlinear loss function is close to quadratic; hence, the approximation of the Hessian of the loss function is close to the true Hessian. Here, one can recall that similar arguments were used to motivate the choice of covariance of the parameters in the Laplacian approximation of BNNs.

### Example 3.2

To illustrate the connection between the Hessian of the loss function and the uncertainty in the parameters, consider again the setting of Ex. 2.6. In Fig. 3.2, the loss function for the canonical representation (2.18) of the NN is shown. Here, the parameters connecting the hidden layer to the output layer are changed while fixing all the other parameters. The loss function is steeper in one direction. A projection of the eigenvector corresponding to the Hessian matrix's largest eigenvalue is the loss function's steeper direction. In contrast, the projection of the eigenvector corresponding to the smallest eigenvalue of the Hessian matrix is a flat direction in the loss function. The smaller eigenvalues of the Hessian would represent larger parameter covariance and a more uncertain prediction, and vice versa. Using this interpretation of the Hessian, one can see that directions with larger curvature in the loss function correspond to directions where the parameter estimate is confident. In comparison, directions with a smaller curvature correspond to directions where the parameter estimate is more uncertain compared to the other direction. One can also see that the solver moves slower in the direction corresponding to the uncertain combination of the parameters compared to the combinations of parameters that are certain.

## 3.1.2 Approximation of the covariance

An NN often has millions of parameters, which might result in the amount of data needed to store  $P_N^\theta$  being larger than the available memory capacity. Here, a common approach to handle this is to approximate  $P_N^\theta$ . There exist many different low-rank approximations, but one of the most common ones is to approximate  $P_N^\theta$  as a block-diagonal matrix [60] or to use the approximation

$$P_N^\theta \approx \begin{bmatrix} P_N^{\theta_r} & 0 \\ 0 & 0 \end{bmatrix}, \quad (3.5)$$

where  $P_N^{\theta_r}$  denotes the covariance of the estimated parameters  $\theta_r$  corresponding to the weights and biases of the  $r$  last layers in the NN [27, 61]. Another approach

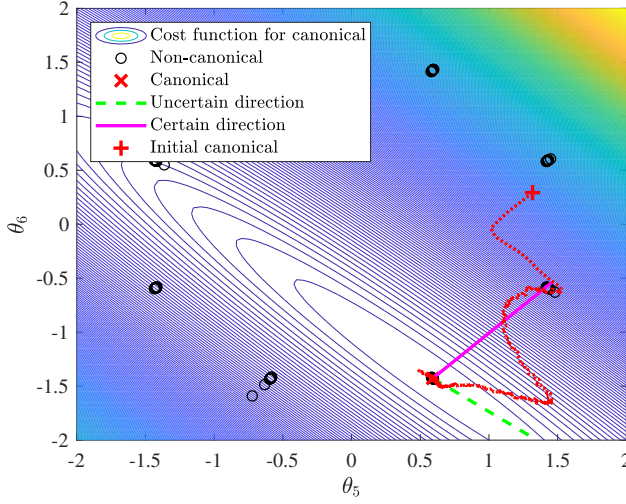


Figure 3.2: Loss function as a function of the parameters connecting the output layer and the last hidden layer, i.e.,  $\theta_5$  and  $\theta_6$ , see Fig. 2.3 and (2.17). The figure also includes the directions of the projected eigenvectors corresponding to the smallest and largest eigenvalue of the covariance matrix of the parameters. That is, in which directions the parameters are the most certain and most uncertain.

to approximate the  $P_N^\theta$  is a singular value decomposition to compute the effective number of parameters [35]. Some combinations of parameters have more influence on the prediction than others. These directions are directions where the parameters are certain, e.g., the steeper direction in Fig. 3.2.

## 3.2 The delta method

The second linearization in the LLA is used to propagate the uncertainty in the parameters to uncertainty in the prediction. To linearize is a well-known trick in the statistical literature and is sometimes referred to as the delta method [57]. It has also successfully been used for NNs see e.g., [36, 62–67]. The main idea is that by linearizing the model, the parameters enter additively, i.e.,

$$f(x; \theta) = f(x; \hat{\theta}_N) + \left. \frac{\partial f(x; \theta)}{\partial \theta} \right|_{\theta=\hat{\theta}_N} (\hat{\theta}_N - \theta). \quad (3.6)$$

Assuming that the distribution of the parameters is known and Gaussian distributed  $\mathcal{D}_\theta = \mathcal{N}(\theta | \hat{\theta}_N, P_\theta)$ , as is the case for the Laplacian approximation. Then, the linearized model will also be Gaussian-distributed according to

$$f(x^\star; \theta) \sim \mathcal{N}(f(x^\star; \hat{\theta}_N), P_N^f) \quad (3.7)$$

for new inputs  $x^\star$  where the covariance of the distribution is given by

$$P_N^f = \left( \frac{\partial f(x^\star; \theta)}{\partial \theta} \Big|_{\theta=\hat{\theta}_N} \right)^\top P_N^\theta \frac{\partial f(x^\star; \theta)}{\partial \theta} \Big|_{\theta=\hat{\theta}_N}. \quad (3.8)$$

Since the idea behind the delta method is a second linearization or Taylor approximation, the method, in combination with the Laplacian approximation, will by us, from hereon, be referred to as the linearized Laplacian approximation (LLA). The linearization can also be interpreted as a projection of the uncertainty from the parameters onto the prediction [36, 62–64]. With this interpretation, the norm of the projection gives the uncertainty in the prediction [68].

Creating an ensemble of predictions and computing a mean and covariance from the ensemble is a straightforward approach to quantifying the uncertainty in the prediction from a model. The delta method is one approach to move the uncertainty in the parameters to uncertainty in the prediction. A second approach is to sample parameters from  $\mathcal{D}_\theta$  and evaluate the model for all the samples. An advantage of creating the ensemble of predictions using the LLA compared to creating it directly by sampling parameters from the Laplacian approximation is that it, in comparison, requires a lower computational effort. This is a result of mainly two factors. Firstly, the dimension of the distribution from which the samples are drawn is significantly larger for the Laplacian approximation. This is a consequence of the Laplacian approximations drawing samples of parameters of the NN, often in the order of hundreds or millions. In contrast, the LLA draws samples from the prediction of the NN, which has a significantly lower dimension (often between one and a hundred). Secondly, and more importantly, the LLA does not require more than two forward passes<sup>1</sup>, i.e., evaluations of the NN multiple times. The lower computational complexity is essential when using the NN in a safety-critical application with limited computational capacity.

However, one has to pay a price in accuracy for this lower computational complexity. This trade-off can be illustrated by an example using the model from Ex. 3.1.

---

### Example 3.3

---

Consider the same model with the same true system as in Ex. 3.1. For a new input  $x^\star$ , we are interested in the probability that the prediction from the model is correct, i.e., the probability density function (PDF)  $p(y^\star = \hat{y}|x^\star, \theta)$ , where  $\hat{y} = f(x^\star; \hat{\theta}_N)$ . Assume that one is using the Laplacian approximation to quantify the uncertainty in the parameters and that it is

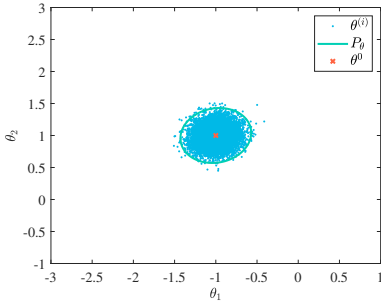
$$P_\theta = \begin{bmatrix} 0.02 & 0.002 \\ 0.002 & 0.02 \end{bmatrix}. \quad (3.9)$$

For this example, we can analytically compute the PDF for the delta method, while a second approach is to approximate the PDF by sampling from  $\mathcal{D}_\theta = \mathcal{N}(\theta|\hat{\theta}_N, P_\theta)$  and evaluate the model  $f(x; \theta)$  for all the samples. The samples

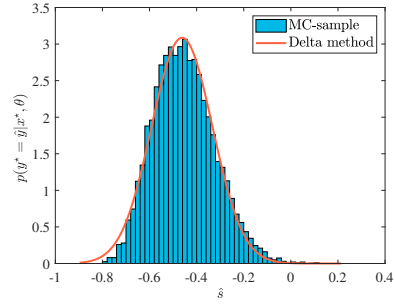
---

<sup>1</sup>The LLA requires one forward pass to compute  $f(x; \theta)$  and one to compute the derivative of  $f(x; \theta)$  with respect to  $\theta$ .

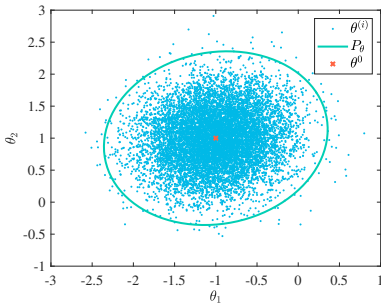




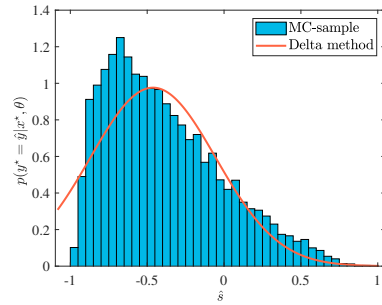
(a) Distribution of the parameters (small).



(b) Distribution of the predicted value, when the uncertainty of the parameters are small.



(c) Distribution of the parameters (large).



(d) Distribution of the predicted value, when the uncertainty of the parameters are large.

Figure 3.3: Visualization of the accuracy of the delta method. To the left are the samples from  $\mathcal{N}(\theta|\hat{\theta}_N, P_\theta)$  shown for two choices of covariance matrix  $P_\theta$ . The right figures show the uncertainty in predicting the function in (3.2). The red line is the delta method approximation, and the histogram is from evaluating (3.2) using the samples on the left.

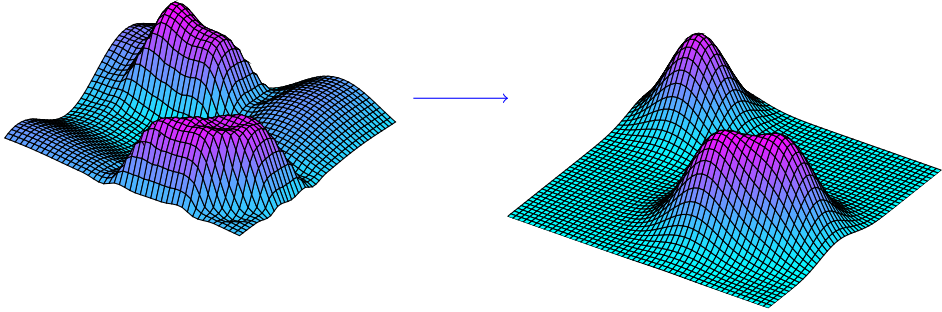


Figure 3.4: Illustration of an ELLA can be used to approximate a multimodal distribution. On the left is the original distribution, and on the right is the approximation using an ensemble of three LLA.

when using  $P_\theta$  in (3.9) is seen Fig. 3.3a, and the samples where the covariance of the parameter has been scaled with a factor of ten in Fig. 3.3c. As shown in Fig. 3.3b, the PDF from the delta method approximates the empirical PDF computed using the sampling scheme well. However, as the uncertainty in the parameters increases, the delta method approximation accuracy decreases. This decrease is seen in Fig. 3.3d where the covariance of the parameter has been scaled with a factor of ten, and the approximation is not accurate anymore.

### 3.3 Ensemble linearized Laplacian approximation

The LLA is a local approximation that requires the found parameters to be close to one local minimum of the loss function. The requirement comes from that both the first linearization is accurate enough, and the uncertainty in the parameters is not large such that the second linearization is accurate enough, see Ex. 3.1 and Ex. 3.3. One approach to tackle the first limitation, i.e., that the LLA only can represent distributions with one mode, is to combine multiple LLA from independently trained models. This extension is presented in Paper E and referred to as the ensemble LLA (ELLA).

#### Example 3.4

Assume a similar setup as in Ex. 3.2. In this case, one can expect the distribution to be multimodal, e.g., as the left distribution on Fig. 3.4. In this case, the LLA could only capture one of the modes since it is a local approximation. However, by combining multiple LLAs, it is possible to capture better the overall shape of the distribution, e.g., the right part of Fig. 3.4.

To model a multimodal landscape can be motivated by symmetries in NNs described in Sec. 2.3.2 and the transformation between optima in the loss landscape described in [69]. The ELLA is similar to the Gaussian mixture model, where multiple Gaussian distributions approximate a non-Gaussian distribution. Given the unimodal Gaussian distributions, there is a question of how to combine them, i.e., should it be equally likely to end up in any of the modes, or should one mode be more likely than the other? In Paper E, it is chosen that every mode is equally likely.

A disadvantage with the ELLA is that it requires training multiple models. Hence, it adds additional computational complexity compared to only using one NN. Since the NNs are trained independently, it is also the case that some of the trained NNs end up in the same mode. A remedy to this problem is to use so-called repulsive sampling during the training [70]. In repulsive sampling, the NNs are trained in parallel, using a penalty factor to ensure that the NNs do not end up in the same (local) optimum.

### 3.4 Summary

To summarise, LLA is based on two linearizations. The first linearization is used to approximate the parameters' covariance, referred to as the Laplacian approximation. The second linearization propagates the uncertainty in the parameters quantified using the Laplacian approximation to uncertainty in the prediction, hence the name LLA. The ELLA is an extension where multiple independently trained NNs are combined to give the flexibility to represent multimodal distributions.



# 4

---

## Uncertainty in neural network predictions

This chapter provides an overview of the sources of uncertainty in the prediction from NNs and a roadmap of different methods used in the literature. The purpose of the overview is to put the suggested method presented in Chapter 3 and the method presented in Paper G into a broader context.

### 4.1 Sources of uncertainty

The uncertainty in the prediction stems from three different sources: errors caused by the optimization algorithm that is used to train the NN, errors in the data (aleatoric uncertainty), and errors in the model (epistemic uncertainty). In this thesis, the focus is on uncertainty from the two latter sources. The error in the data comes from noise in the measurements, i.e., a stochastic error. An approach to handle the stochastic measurements and decrease the error caused by the data is to include more data in the training data used to train the NN. Some of the errors caused by the model could come from the fact that the model class is not flexible enough to describe the true system. This error leads to a systematic bias in estimating the parameters and, hence, in the prediction from the model. A remedy to handle these errors is to increase the model's flexibility. However, the systematic bias could also be a result of a lack of training data in a given region, which, as for the stochastic error, is solved by increasing the amount of training data.

The difference between the stochastic error (variance) and systematical error (bias) can be visualized in the estimate of the parameters using, e.g., how an increased number of training data points leads to a reduced stochastic error. Here is illustrated by an example. The example considers the same nonlinear system as in Ex. 2.6, i.e., using an NN to estimate the normalized traction force (NTF) using measurements of the wheel slip as inputs. However, compared to Ex. 2.6,

for simplicity, consider a model that is linear in its parameters, i.e.,

$$f(x; \theta) = \varphi^\top(x) \theta, \quad (4.1)$$

where  $\varphi^\top(x)$  is some (potentially) nonlinear transformation of the input  $x$ . There is, however, a connection between NNs and models that are linear in their parameters, i.e., one could consider an NN as a collection of nonlinear basis functions that are linearly combined in the output layer [71]. For the NN case, the basis function could be given as

$$\varphi^\top = \left( h^{(L)} \quad 1 \right)^\top, \quad (4.2a)$$

where

$$\theta = W^{(L)}. \quad (4.2b)$$

---

#### Example 4.1

---

Given a varying number of noisy measurements of the NTF for a given wheel slip, the problem is to model this relationship using a model given by (4.1). The number of measurements varies between 10 and 1000, the mean of the noise is zero, and the variance is  $10^{-3}$ , i.e., a signal-to-noise ratio (SNR) of 30 dB. Assume that the true model describing the relationship between the NTF and wheel slip is given by

$$f^0(x; \theta_1) = \varphi_1^\top \theta_1, \quad (4.3a)$$

$$\varphi_1 = \left( \sigma(6x + 6) \quad \sigma(-40x + 0.0061) \quad \sigma(-6.8x + 0.0036) \quad 1 \right)^\top, \quad (4.3b)$$

for the parameters  $\theta_1 = (0.40 \quad 1.44 \quad -0.60 \quad -0.42)^\top$ , where  $x$  is the wheel slip. Apart from using the model described in (4.3) where the true system is in the model set  $\mathcal{S} \in \mathcal{M}^*$ , the model

$$f^*(x; \theta_2) = \varphi_{2,m}^\top \theta_2, \quad (4.4a)$$

$$\varphi_2 = \left( \sigma(-40x + 0.0061) \quad \sigma(-6.8x + 0.0036) \quad 1 \right)^\top, \quad (4.4b)$$

is also used to model the measurements. That is a model where the true system is not included in the model set,  $\mathcal{S} \notin \mathcal{M}^*$ . Both (4.3) and (4.4) are linear in the parameters; hence, the parameters can be estimated by the linear least squares solution (LLSS).

In Fig. 4.1a, the estimate of the parameter corresponding to the third nonlinear basis function in (4.3) (second in (4.4)), is plotted as the models are estimated using more samples. From Fig. 4.1a, one can observe that the stochastic error decreases as the number of training data increases. Due to the systematic error, the parameters converge to different values, as seen in the difference in the horizontal lines. However, despite the systematic error in Fig. 4.1b, it is nearly impossible to distinguish the predictions from the two models.

---

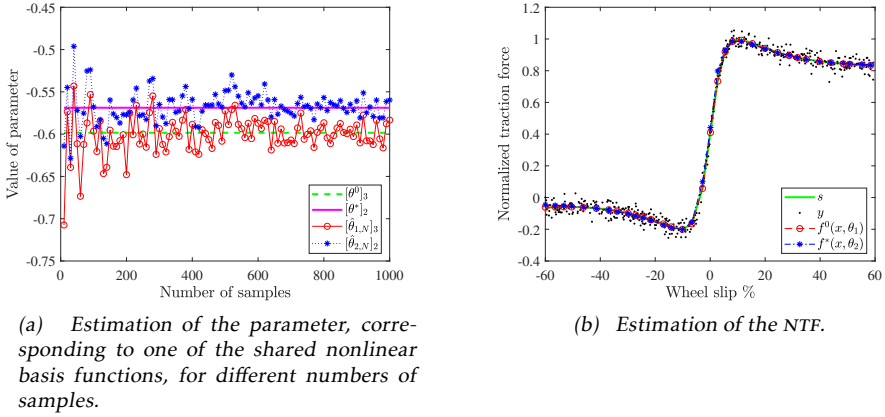


Figure 4.1: Illustration of the stochastic and systematic deterministic uncertainty, both using a model with and without the true system in the model set, see (4.3) and (4.4).

## 4.2 Roadmap of uncertainty methods

The uncertainty in the prediction of NNs can be quantified using many different methods [37, 72–75]. For a survey of different methods, see [76]. There are numerous approaches to categorize these different methods, which can have broader or narrower categories. This thesis identifies two broad top categories, namely methods using an ensemble of NNs and methods where the structure of the NN is changed such that the NN is learning the uncertainty in the prediction.

Fig. 4.2 shows a diagram categorizing different methods presented in the literature into the categories mentioned above. The core method presented in the thesis, LLA, (Paper A, Paper D, Paper E) belongs to the category of ensemble approaches while the method proposed in Paper G belongs to the category where the NN learns its own uncertainty. The sub-categories of the proposed method have been marked red in the diagram.

## 4.3 Ensemble methods

A straightforward approach to quantify the uncertainty in the prediction is computing a mean and variance using an ensemble of predictions.

### 4.3.1 Multiple models

One method in that category is the so-called deep ensemble [83]. The methods train multiple models on the same data, where all models in the ensemble predict new outputs. Training multiple models does not guarantee that the models will converge to different optima, i.e., all the models could be identical. The NNs could be trained in parallel using so-called repulsive training to make it less



Figure 4.2: Schematic illustrations over the categorizing of the methods to quantify uncertainty in the prediction. (i): [54, 77], (ii): [78–82] (iii): Paper G, (iv): [70, 83], (v): Paper A, Paper D, Paper E, [60, 61, 65, 66], (vi): [84, 85], (vii): [86–88], (viii): [89].



likely that they are identical [70]. Combining models has also been shown to improve prediction performance, e.g., [90, 91]. Multiple independently trained models can also model uncertainty caused by error training algorithms. The ensemble can represent the uncertainty caused by the training algorithm because the multiple realizations can represent the different local optima the training algorithm could find.

### 4.3.2 Multiple inputs

Where training a single NN is computationally expensive, training an ensemble of NNs can often be infeasible, particularly if one uses enormous NNs with many parameters. Hence, a method to create these ensembles without training multiple NNs has been suggested in the literature, e.g., the so-called test time augmentation [89]. In time augmentation, to create an ensemble of predictions, first, an ensemble of inputs to predict is created. This ensemble of inputs is created by augmenting (changing/transforming) the input for which one would like to know the uncertainty in the prediction. Then, the NNs predict the values of all the augmented inputs to create an ensemble from which uncertainty in the prediction can be computed. In Fig. 4.2, test time augmentation is categorized as multiple inputs. The method is commonly used in applications where there is little training data available and where there exist transformations of the input that should not affect the prediction of the input. One example of such an application is in medical imaging, where the prediction of a rotated image should be the same as the prediction of a non-rotated one. An advantage of test time augmentation is that it is straightforward to implement. However, test time augmentation does not incorporate how close to its optimal value the parameters in the NN have converged to after the training phase. Another disadvantage is that it requires specifying a transformation that should keep the prediction of the output the same for the augmented input. This transformation can be challenging to find. For example, rotational invariance, which is valid for some applications where images are classified, might not be valid for others, e.g., if the NN should classify handwritten images, a rotated image of a six is a nine.

### 4.3.3 Approximating the parameter distribution

Another category of methods is those that approximate the parameter distribution of the NN. The approximation is made during training where values of the parameters are sampled during prediction to create an ensemble of predictions [60, 61, 65, 66, 84–88], Paper A, and Paper D.

Methods such as Monte-Carlo (MC) dropout [86, 87] and MC batchnorm [88] use regularization techniques to approximate the distribution of the parameters of the NN. For example, some parameters are dropped during training in dropout, so the network should not overfit the training data. In MC dropout, the idea is to also use dropout during the prediction to create an ensemble of predictions.

Other methods use the curvature of the loss function to create the ensemble. For example, [60, 61] saves samples of parameters later in the training, i.e.,

when the training is close to the optima of the loss function. The Laplacian approximation, see Sec. 3.1, is another approach where the parameter distribution is assumed Gaussian and where the covariance is given by a linearization [65, 66, 84, 85].

A disadvantage of many ensemble methods is that they require multiple forward passes of the NN to create the ensemble. A remedy for this problem is LLA presented in Paper A and in Paper D. Here, the uncertainty in the parameters is propagated to uncertainty in the prediction such that only two forward pass are required. For more information on LLA, see Chapter 3.

However, the LLA is a local approximation and can only represent distributions with one mode, which is also the case for many methods relying on approximating the parameter covariance of the NN. For the LLA, the extension ELLA presented in Paper E and Sec. 3.3 can be used to represent distributions with multiple modes, see, e.g., Ex. 3.1 and Ex. 3.3. However, this method requires training multiple NNs. In Fig. 4.2, ELLA could have been classified as multiple models, but since it is based on a Laplacian approximation, in this thesis, it is chosen to belong to the category which approximate the parameter distribution.

Another area for improvement with methods that rely on creating ensembles is that they need help representing the model's bias due to a model mismatch. The bias could, e.g., be caused by using a model with too low flexibility, having a too high regularization on the parameters, or using an overparameterized model.

## 4.4 Learned uncertainty

Methods from the second category, where the uncertainty in the prediction is encoded, can represent the bias due to the model mismatch, i.e., the models that learn their own uncertainty. This thesis will further divide the methods into three different sub-categories: sampling-based methods, methods explicitly modeling the uncertainty, and methods implicitly modeling the uncertainty, see Fig. 4.2.

### 4.4.1 Sampling-based methods

By modeling some distribution for the parameters or the prediction, it is possible to obtain uncertainty in the prediction from the NN by sampling from the posterior distribution of the prediction, i.e., using a BNN [54]. To learn the posterior by sampling is a difficult task. However, there have been some recent advances to use clever Markov Chain MC (MCMC) sampling implementations to train the BNN [77, 92]. One example is to combine the training algorithm, e.g., stochastic gradient, with MCMC sampling to approximate a BNN [92]. However, although the formulation is simple and elegant, the posterior has no closed expression to draw samples from. Specifying meaningful prior for NNs is also challenging and poorly understood [93]. Due to the challenges in training the NNs by sampling, methods exist to approximate the BNN by already trained NNs. See Sec. 4.3.3 for further description of these approximations.

### 4.4.2 Explicitly modeling the uncertainty

Instead of learning the uncertainty in the prediction using some sampling scheme, one could explicitly model it. To explicitly model the uncertainty can be, e.g., done by changing the model to include an uncertainty measure in the prediction [78–82]. A consequence of changing the model is that it also requires changing the loss function. Hence, more advanced training algorithms might be required. Explicitly modeling the uncertainty could also be done by training a second NN to learn the uncertainty [94]. Explicitly learning the uncertainty in the prediction requires further assumptions regarding the distribution. A disadvantage of explicitly learning the uncertainty is that it often requires more training data due to the more complicated loss function and training algorithm. For these methods, the learned uncertainty might also be incorrect, or there is an uncertainty in the learned uncertainty.

### 4.4.3 Implicitly modeling the uncertainty

Some constructions of the model can implicitly include information from which uncertainty in the prediction can be attained. One example of implicitly including uncertainty in the prediction is the NN architecture-based object detection models Single Shot MultiBox Detector (SSD) [95]. Here, the SSD algorithm provides multiple predictions of the object's class and position in the image. However, these are usually truncated, so only the most likely output is used. Hence, in Paper G, it is suggested to use multiple predictions to quantify the uncertainty in the prediction from the NN. Multiple predictions, in turn, enable the use of the prediction from the NN in a tracking framework.

## 4.5 Summary

To summarize, there exist three different significant sources of uncertainty. Firstly, due to an error caused by the training algorithm, i.e., the network has not converged to the correct parameters. Secondly, due to errors in the data, and thirdly, due to errors in the model. The method presented in this thesis covers uncertainty due to the two later sources. Here, many different approaches exist to quantify uncertainty in the prediction from NNs. The gold standard would be to use a BNN directly. However, it might not be feasible due to high computational complexity and lack of a closed-form expression of the posterior for the BNN. Instead, one could, e.g., let the NN learn its own uncertainty or approximate the BNN using ensembles. A disadvantage of methods relying on learning their own uncertainty is that they require more intricate training algorithms. The fact that they need intricate training algorithms might make them more vulnerable to the uncertainty caused by the error in the training phase. Regarding the methods based on ensembles, one disadvantage is that they cannot model the uncertainty due to a model mismatch. That is because, for the ensemble, the model structure is fixed, and if a model mismatch exists, it will be present in all the models in

the ensemble. On the other hand, they do not need an advanced training algorithm. Some methods can even be used out-of-the-box for already trained NNs. Another advantage is that some methods based on the ensemble, which requires training multiple NNs, can consider the uncertainty due to the error caused by the training algorithm. However, since they require training of multiple models, they have a high computational complexity.

# 5

---

## Concluding remarks

The first part of the thesis has given some background theory, which forms the basis for the methods developed in the second part. We have provided an overview of methods to quantify uncertainty in the predictions from NNs, where the emphasis has been on the suggested methods used in Part II of the thesis. This final chapter of the first part summarizes the scientific contributions of the thesis and gives some conclusions of the work. The chapter will conclude with some possible future directions for further research.

### 5.1 Summary of contributions

This section summarises the main contributions of the thesis. The contribution can be divided into two categories: the development of the method and how to make more robust decisions using the developed method of quantified uncertainty.

#### 5.1.1 Uncertainty quantification in the prediction of neural networks

One of the main contributions of the thesis is the development of the two methods to quantify uncertainty in predictions from NNs. Firstly, the LLA used for regression and classification tasks suggested in Paper A and Paper D, and secondly, the method for objected detection tasks suggested in Paper G. An extension of the LLA, which enables the modeling of multimodal distributions, is also presented in Paper E. To provide a context in which the suggested methods fit, Chapter 4 of the thesis presents an overview of methods used in the literature to quantify uncertainty in the prediction from NNs. The thesis uses simulation data and real-world data from experiments on road-friction data and camera trap images to

validate the methods.

The thesis compares the LLA to different methods to quantify the uncertainty in the prediction of NNs in Paper D, and in Paper C compare the LLA to other data-driven models that including uncertainty in their prediction. The comparison validates that the LLA creates a reliable estimate of the uncertainty in the prediction. The uncertainty estimate from the LLA is shown to be reliable and more calibrated, or at least as calibrated, compared to commonly used methods in the literature. Since overparameterization is common when modeling using NNs, in Paper B, it analyzes how overparameterization affects the quantified uncertainty from the LLA. Here, the uncertainty is shown not to be underestimated. The thesis provides non-standard computation of the Fisher information matrix for the regression and classification tasks required for the LLA.

### 5.1.2 Robust decision making and trustworthy decisions

Two of the main benefits of having access to the uncertainty in the prediction from a model is that the decisions based on those predictions can become more robust and trustworthy. As shown in Paper D, Paper E, and Paper G, with access to the uncertainty in the prediction, predictions from NNs can be used in a sensor-fusion framework where multiple predictions can be combined. These predictions could come from other models predicting the same signal or previous predictions from the same model. Hence, introducing redundancies for the decision process, including predictions from NNs, can lead to more robust decisions. Outliers and OOD examples can be detected using the measure of the uncertainty in the prediction as shown in Paper D, Paper E, Paper F and Paper G. A measure of the uncertainty in the prediction is important for any safety-critical system where a missed detection or an incorrect interpretation of the situation can lead to severe problems. Access to the uncertainty also provides a better understanding of why the decision system is making a specific decision. The knowledge that a decision process is more robust and its decisions can be better understood leads to a more trustworthy system.

## 5.2 Conclusions

This section summarises the scientific contributions of the papers in Part II of the thesis. From the theoretical analysis, the simulations, and the experiments in the thesis, one can conclude that the proposed methods produce a reliable estimate of the uncertainty in the prediction. The conclusion is based on comparing the LLA to other models and other methods used to quantify the uncertainty in the prediction. For the LLA, it is possible to conclude that using the method for overparameterized models is accurate enough. It is viable because the method does not underestimate the quantified uncertainty in the prediction, i.e., the predictions are not overly confident. To be able to use the method for overparameterized models is especially of high importance for NNs since they often are overparameterized in order to be able to model the true system, even if it is complicated.

For detection of OOD examples, it is also shown that knowledge of the uncertainty is highly important. Overall, the fact that having access to the uncertainty lets us combine multiple predictions in a sensor-fusion framework. As a result, having access to this leads to the conclusion that using the uncertainty makes it possible to make a more robust decision in decision processes that include predictions from NNs.

## 5.3 Future work

Several directions exist in which this thesis work can be extended. One direction would be to continue the integration of the prediction of the NNs into a sensor-fusion framework using both the OOD detection and the sequential fusion of prediction to create more robust decision systems.

The thesis suggests that for large NNs with many parameters, a part of the parameters in the network can be considered fixed, and only the uncertainty for the parameters in the layers toward the network's end needs to be computed. It is only computed for the last layers because the amount of data needed to store the full covariance matrix might be larger than the available memory capacity. Here, an interesting future research direction would be to investigate how accurate this approximation is. In particular, is it of higher importance to learn the parameters in the later layers that combine the created basis functions from the earlier layers? A related suggested research direction is to investigate other methods to approximate the covariance matrix. As shown in Paper D, covariance scaling is required to get reliable uncertainty quantification of the prediction from the NN. That this scaling is necessary indicates that the approximation of the covariance matrix might miss to capture some uncertainty and not be completely accurate.

As shown in Paper E, fused predictions from multiple NNs provided excellent OOD detection capability, while the multimodal method provided the most calibrated quantified uncertainty in the prediction. Hence, a future direction would be the combination of them, i.e., how should the samples from the different modes be drawn considering the uncertainty in the different modes?

Using the quantified uncertainty to do model selection or pruning of the NN could be another future research direction. Pruning can reduce the memory requirements of the NN, which is vital for their use on edge devices.





---

## Bibliography

- [1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Adv. in Neural Inf. Process. Syst. (NeurIPS)* 25, pages 1097–1105, Lake Tahoe, NV USA, 2012. 3-8 Dec.
- [2] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *Proc. of the 34th Int. Conf. on Mach. Learn. (ICML)*., pages 1263–1272, Sydney, Australia, 2017. 06–11 Aug.
- [3] Open-AI. GPT-4 Technical Report. Technical report (tr), Open-AI, 3 2023.
- [4] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *Proc. of 1st Int. Conf. for Learn. Representations (ICLR)*, Scottsdale, AZ, USA, 2013. 2-4, May.
- [5] Rickard Karlsson and Gustaf Hendeby. Speed estimation from vibrations using a deep learning CNN approach. In *IEEE Sensors Letters*, volume 5, pages 1–4. IEEE, 2021.
- [6] Qiyang Li, Jingxing Qian, Zining Zhu, Xuchan Bao, Mohamed K Helwa, and Angela P Schoellig. Deep neural networks for improved, impromptu trajectory tracking of quadrotors. In *Proc. of IEEE Int. Conf. on Robot. and Autom. (ICRA)*, pages 5183–5189, Singapore, Singapore, 2017. IEEE. 19 May–3 June.
- [7] Jennie Karlsson, Ida Arvidsson, Freja Sahlin, Kalle Åström, Niels Christian Overgaard, Kristina Lång, and Anders Heyden. Classification of point-of-care ultrasound in breast imaging using deep learning. In *Medical Imaging 2023: Computer-Aided Diagnosis*, volume 12465, pages 192–200. SPIE, 2023.
- [8] NTSB. Highway accident report HAR-19/03 HWY18MH010. Technical specification (ts), National Transportation Safety Board (NTSB), 03 2018.
- [9] BBC News. Tesla in fatal California crash was on Autopilot. Technical report, BBC News, 31 March 2018. [Last accessed: 9 Sep. 2020].

- [10] Reuters. Tesla in fatal California crash was on Autopilot. Technical report, Reuters, 21 Aug 2017. [Last accessed: 9 Sep. 2020].
- [11] Saeed Asadi Bagloee, Madjid Tavana, Mohsen Asadi, and Tracey Oliver. Autonomous vehicles: challenges, opportunities, and future implications for transportation policies. In *J. of Modern Trans.*, volume 24, pages 284–303. Springer, December 2016.
- [12] Sorin Grigorescu, Bogdan Trasnea, Tiberiu Cocias, and Gigel Macesanu. A survey of deep learning techniques for autonomous driving. In *J. of Field Robotics*, volume 37, pages 362–386. Wiley Online Library, January 2020.
- [13] Andrei Paleyes, Raoul-Gabriel Urma, and Neil D Lawrence. Challenges in deploying machine learning: a survey of case studies. In *Adv. in Neural Inf. Process. Syst. (NeurIPS) 34 Workshop: ML Retrospectives, Surveys & Meta-Analyses (ML-RSA)*, volume 33, Virtual, 7–12 Dec 2020.
- [14] George EP Box. Science and statistics. *J. of the Am. Stat. Assoc. (JSTOR)*, 71(356):791–799, 1976.
- [15] Anuradha M Annaswamy, Karl H Johansson, George J Pappas, et al. Control for societal-scale challenges: Road map 2030. *IEEE Control Systems Society Publication*, 2023.
- [16] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *Proc. of the 38th Int. Conf. on Mach. Learn. (ICML)*, pages 8821–8831, Virtual, 2021. PMLR. 18–24 Jul.
- [17] Future for life. Pause Giant AI Experiments: An Open Letter. Technical report, Future for life institute, 22 March 2023. [Last accessed: 15 Sep. 2023].
- [18] Kevin Eykholt, Ivan Evtimov, Earlene Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust Physical-World Attacks on Deep Learning Visual Classification. In *Proc. of IEEE Conf. on Comput. Vision and Pattern Recognition (CVPR)*, pages 1625–1634, 2018. Salt Lake City, UT, USA, June 18–22.
- [19] Yann LeCun, Corinna Cortes, and Christopher J.C. Burges. The MNIST database of handwritten digits, 1998.
- [20] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.
- [21] Magnus Malmström. *Uncertainties in Neural Networks A System Identification Approach*. Licentiate thesis, Dept. Elect. Eng., Linköping University, Linköping, Sweden, Apr 2021.

- [22] Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Asymptotic prediction error variance for feedforward neural networks. In *Proc. of 21st IFAC World Congress, (IFAC)*, Online (Berlin, Germany), 2020. Jul 11-17.
- [23] Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. On the validity of using the delta method for calculating the uncertainty of the predictions from an overparameterized model. In *Proc. of 22nd IFAC World Congress, (IFAC)*, Yokohama, Japan, 2023. Jul 9-17.
- [24] Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Modeling of the tire-road friction using neural networks including quantification of the prediction uncertainty. In *Proc. of IEEE 24th Int. Conf. on Inf. Fusion (FUSION)*, pages 1–6, Sun City, South Africa/ Virtual, 2021. IEEE. Nov 1-4.
- [25] Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Uncertainty quantification in neural network classifiers—a local linear approach. *arXiv preprint arXiv:2303.07114*, Under review for possible publication in *Automatica*, 2023.
- [26] Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Fusion framework and multimodality for the Laplacian approximation of Bayesian neural networks. *arXiv preprint arXiv:2310.08315*, Submitted for possible publication in *IEEE Trans. Aerosp. Electron. Syst.*, 2023.
- [27] Magnus Malmström, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Detection of outliers in classification by using quantified uncertainty in neural networks. In *Proc. of IEEE 25th Int. Conf. on Inf. Fusion (FUSION)*, Linköping, Sweden, 2022. IEEE. Jul 4-7.
- [28] Magnus Malmström, Anton Kullberg, Isaac Skog, Daniel Axehill, and Fredrik Gustafsson. Extended target tracking utilizing machine-learning software – with applications to animal classification. *arXiv preprint arXiv:2310.08316*, Submitted for possible publication in *IEEE Signal Process. Lett.*, 2023.
- [29] Magnus Malmström, Isaac Skog, Sara Modarres Razavi, Yuxin Zhao, and Fredrik Gunnarsson. 5G Positioning - A Machine Learning Approach. In *Proc. of IEEE 16th Workshop on Positioning Navig. Commun. , (WPNC)*, Bremen, Germany, 2019. 23-24 Oct.
- [30] Jacob Eek, David Gustafsson, Ludwig Hollmann, Markus Nordberg, Isaac Skog, and Magnus Malmström. A novel and fast approach for reconstructing CASSI-raman spectra using generative adversarial networks. In *2022 11th Int. Conf. on Image Proc. Theory, Tools and App. (IPTA)*, pages 1–6, Salzburg, Austria, 2022. IEEE. Apr 19-22.
- [31] Lennart Ljung. *System identification: theory for the user (2nd edition)*. PTR Prentice Hall: Upper Saddle River, NJ, USA, 1999.

- [32] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT Press, 2016. London, England.
- [33] Hirotugu Akaike. A new look at the statistical model identification. *IEEE Trans. Autom. Control*, 19(6):716–723, 1974.
- [34] Gideon Schwarz. Estimating the dimension of a model. *Ann. of Stat.*, 6(2):461–464, 1978.
- [35] Anna Marconato, Maarten Schoukens, Yves Rolain, and Johan Schoukens. Study of the effective number of parameters in nonlinear identification benchmarks. In *Proc. of IEEE 52nd Conf. on Decision and Contr. , (CDC)*, pages 4308–4313, Florence, Italy, 2013. 10–13 Dec.
- [36] J. T. Gene Hwang and A. Adam Ding. Prediction Intervals for Artificial Neural Networks. In *J. Am. Stat. Assoc. (JSTOR)*, volume 92, pages 748–757. Taylor & Francis, 1997.
- [37] Alexander D’Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D Hoffman, Farhad Hormozdiari, Neil Houlsby, Shaobo Hou, Ghas-sen Jerfel, Alan Karthikesalingam, Mario Lucic, Yian Ma, Cory McLean, Di-ana Mincu, Akinori Mitani, Andrea Montanari, Zachary Nado, Vivek Natara-jan, Christopher Nielson, Thomas F. Osborne, Rajiv Raman, Kim Ramasamy, Rory Sayres, Jessica Schrouff, Martin Seneviratne, Shannon Sequeira, Harini Suresh, Victor Veitch, Max Vladymyrov, Xuezhi Wang, Kellie Webster, Steve Yadlowsky, Yun Taedong, Xiaohua Zhai, and D. Sculley. Underspecification presents challenges for credibility in modern machine learning. In *arXiv preprint arXiv:2011.03395*, 2020.
- [38] Héctor J Sussmann. Uniqueness of the weights for minimal feedforward nets with a given input-output map. *Neural netw.*, 5(4):589–593, 1992.
- [39] HB Pacejka and IJM Besselink. Magic formula tyre model with transient properties. *Veh. syst. dynamics-Int. J. of Veh. Mechanics and Mobility*, 27(S1):234–249, 1997.
- [40] Fredrik Gustafsson. Slip-based tire-road friction estimation. *Automatica*, 33(6):1087–1099, 1997.
- [41] Diederik P. Kingma and Jimmy Ba. ADAM: A Method for Stochastic Opti-mization. In *Proc. of 3rd Int. Conf. for Learn. Representations (ICLR)*, 7–9 May 2015. San Diego, CA, USA.
- [42] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proc. of 16th Int. Conf. on Artificial Intell. and Statistics. (AISTATS)*, pages 249–256, Sardinia, Italy, 2010.
- [43] Michel Gevers, Alexandre Sanfelice Bazanella, Xavier Bombois, and Ljubisa Miskovic. Identification and the information matrix: how to get just suffi-ciently rich? *IEEE Trans. Autom. Control*, 54(12):2828–2840, 2009.

- [44] Lennart Ljung and Peter E Caines. Asymptotic normality of prediction error estimators for approximate system models. *Stochastics*, 3, 1980.
- [45] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proc. of IEEE Conf. on Comput. Vision and Pattern Recognition (CVPR)*, pages 1–9, 7–12 Jun 2015. Boston, MA, USA.
- [46] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. of IEEE Conf. on Comput. Vision and Pattern Recognition (CVPR)*, pages 770–778, Las Vegas, NV, USA, 2016. 27–30 Jun.
- [47] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86(11):2278–2324, 1998.
- [48] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proc. of the National Academy of Sciences*, 116(32):15849–15854, 2019.
- [49] Rahul Parhi and Robert D Nowak. Deep learning meets sparse regularization: A signal processing perspective. *arXiv preprint arXiv:2301.09554*, 2023.
- [50] Tomas McKelvey. Analysis of Interpolating Regression Models and the Double Descent Phenomenon. In *Proc. of 22nd IFAC World Congress, (IFAC)*, Yokohama, Japan, 2023. Jul 9–17.
- [51] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. New York, NY, USA.
- [52] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. New York, NY, USA, pages 213–220, 315–319.
- [53] Iain Johnstone. High dimensional bernstein-von mises: simple examples. *Institute of Mathematical Statistics collections*, 6:87–98, 01 2010.
- [54] David JC MacKay. A practical Bayesian framework for backpropagation networks. In *Neural computation*, volume 4, pages 448–472. MIT Press, 1992.
- [55] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media,; New York, NY, USA, 2006.
- [56] Steven M Kay. *Fundamentals of statistical signal processing Estimation theory*. Prentice Hall PTR, cop. 1993: Upper Saddle River, NJ, USA, 1993.
- [57] Hannelore Liero and Silvelyn Zwanzig. *Introduction to the theory of statistical inference*. Chapman and Hall CRC Texts in Statistical Science, Boca Raton, FL, USA, 2011.

- [58] F. Gustafsson. *Statistical Sensor Fusion*. Studentlitteratur: Lund Sweden, 2018.
- [59] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 2012.
- [60] James Martens and Roger Grosse. Optimizing neural networks with Kronecker-factored approximate curvature. In *Proc. of the 32nd Int. Conf. on Mach. Learn. (ICML)*, Lille, France, 2015. 6–11 Jul.
- [61] Agustinus Kristiadi, Matthias Hein, and Philipp Hennig. Being Bayesian, even just a bit, fixes overconfidence in ReLU networks. In *Proc. of the 37th Int. Conf. on Mach. Learn. (ICML)*, Online, 2020. 13-18 July.
- [62] Isabelle Rivals and Léon Personnaz. Construction of confidence intervals for neural networks based on least squares estimation. In *Elsevier J. Neural Netw.*, volume 13, pages 463–484. Elsevier, January 2000.
- [63] G. Papadopoulos, P.J. Edwards, and A.F Murray. Confidence estimation methods for neural networks: a practical comparison. In *IEEE Trans. Neural Netw.*, volume 12, pages 1278–1287, November 2001.
- [64] G. Chrysosoiuris, M. Lee, and A. Ramsey. Confidence interval prediction for neural network models. In *IEEE Trans. Neural Netw.*, volume 7, pages 229–232, January 1996.
- [65] Alexander Immer, Maciej Korzepa, and Matthias Bauer. Improving predictions of Bayesian neural nets via local linearization. In *Proc. of 24th Int. Conf. on Artificial Intell. and Statistics. (AISTATS)*, pages 703–711, San Diego, CA, USA, 2021. PMLR. 13-15 Apr.
- [66] Zhijie Deng, Feng Zhou, and Jun Zhu. Accelerated linearized Laplace approximation for Bayesian deep learning. *arXiv preprint arXiv:2210.12642*, 2022.
- [67] Shouling He and Jiang Li. Confidence Intervals for Neural Networks and Applications to Modeling Engineering Materials. In *Artificial Neural Netw.* IntechOpen, 2011.
- [68] Håkan Hjalmarsson and Jonas Martensson. A geometric approach to variance analysis in system identification. In *IEEE Trans. Autom. Control*, volume 56, pages 983–997. IEEE, 2010.
- [69] Samuel K Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. In *Proc. of 11th Int. Conf. for Learn. Representations (ICLR)*, AKigali Rwanda, 1–5 May, 2022. arXiv preprint arXiv:2209.04836.

- [70] Francesco D’Angelo and Vincent Fortuin. Repulsive deep ensembles are Bayesian. In *Adv. in Neural Inf. Process. Syst. (NeurIPS)* 35, volume 34, pages 3451–3465, 6–14 Dec 2021. Virtual.
- [71] Jonas Sjöberg, Håkan Hjalmarsson, and Lennart Ljung. Neural networks in system identification. *IFAC Proceedings Volumes*, 27(8):359–382, 1994.
- [72] Zoubin Ghahramani. Probabilistic machine learning and artificial intelligence. In *Nature*, volume 521, pages 452 – 459, February 2015.
- [73] Kinjal Patel and Steven Waslander. Accurate prediction and uncertainty estimation using decoupled prediction interval networks. In *arXiv preprint arXiv:2202.09664*, 2022.
- [74] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua V Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model’s uncertainty? evaluating predictive uncertainty under dataset shift. In *Adv. in Neural Inf. Process. Syst. (NeurIPS)* 33, Vancouver, Canada, 2019. 8–14 Dec.
- [75] Shuyu Lin, Ronald Clark, Niki Trigoni, and Stephen Roberts. Uncertainty estimation with a VAE-classifier hybrid model. In *Proc. of IEEE Int. Conf. on Acoust., Speech and Signal Processing (ICASSP)*, pages 3548–3552, Singapore, Singapore,, 2022. IEEE. 22–17 May.
- [76] Jakob Gawlikowski, Cedrique Rovile Njiteutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, Shahzad Muhammad, Bamler Wen Yang, Richard, and Zhu Xiao Xiang. A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, pages 1–77, 2023.
- [77] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight Uncertainty in Neural Networks. In *Proc. of the 32nd Int. Conf. on Mach. Learn. (ICML)*., pages 1613–1622, Lille, France, 2015. 6–11 Jul.
- [78] Fredrik K Gustafsson, Martin Danelljan, Goutam Bhat, and Thomas B Schön. Energy-Based Models for Deep Probabilistic Regression. In *Proc. of 16th European Conf. on Comput. Vision (ECCV)*, pages 325–343, Glasgow, UK/Online, 2020. 23–28 Aug.
- [79] Alex Kendall and Yarin Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In *Adv. in Neural Inf. Process. Syst. (NeurIPS)* 31, pages 5574–5584. Curran Associates, Inc., 4–9 Dec 2017. Long Beach, CA, USA, 4–9 Dec.
- [80] Bertrand Charpentier, Daniel Zügner, and Stephan Günnemann. Posterior network: Uncertainty estimation without ood samples via density-based pseudo-counts. In *Adv. in Neural Inf. Process. Syst. (NeurIPS)* 34, volume 33, Virtual, 07–12 Dec 2020.

- [81] Andrey Malinin and Mark Gales. Predictive uncertainty estimation via prior networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Adv. in Neural Inf. Process. Syst. (NeurIPS) 31*. Curran Associates, Inc., 4–9 Dec 2017. Long Beach, CA, USA.
- [82] Abdelrahman Eldesokey, Michael Felsberg, and Fahad Shahbaz Khan. Propagating confidences through CNNs for sparse data regression. In *British Mach. Vision Conf. (BMVC)*, page 14, 2018. Newcastle, UK, Sep 3-6.
- [83] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Adv. in Neural Inf. Process. Syst. (NeurIPS) 31*. Curran Associates, Inc., 4–9 Dec 2017. Long Beach, CA, USA, 4–9 Dec.
- [84] Wesley J Maddox, Pavel Izmailov, Timur Garipov, Dmitry P Vetrov, and Andrew Gordon Wilson. A simple baseline for Bayesian uncertainty in deep learning. In *Adv. in Neural Inf. Process. Syst. (NeurIPS) 33*, Vancouver, Canada, 2019. 8–14 Dec.
- [85] Kazuki Osawa, Siddharth Swaroop, Mohammad Emtiyaz E Khan, Anirudh Jain, Runa Eschenhagen, Richard E Turner, and Rio Yokota. Practical deep learning with Bayesian principles. In *Adv. in Neural Inf. Process. Syst. (NeurIPS) 33*, Vancouver, Canada, 2019. 8–14 Dec.
- [86] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *Proc. of the 33rd Int. Conf. on Mach. Learn. (ICML)*., pages 1050–1059, New York, NY, USA, 2016. 20–22 Jun.
- [87] Yrin Gal. *Uncertainty in Deep Learning*. PhD thesis, University of Cambridge, January 2017.
- [88] Mattias Teye, Hossein Azizpour, and Kevin Smith. Bayesian Uncertainty Estimation for Batch Normalized Deep Networks. In *Proc. of the 35th Int. Conf. on Mach. Learn. (ICML)*., pages 4907–4916, 6–11 Jul 2018. Stockholm, Sweden ,6–11 Jul.
- [89] Murat Seckin Ayhan and Philipp Berens. Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks. In *1st Conf. on Medical Imaging with Deep Learn. (MIDL)*, Amsterdam, The Netherlands, 2018. 4–6 Jul.
- [90] J. G. Carney, P. Cunningham, and U. Bhagwan. Confidence and prediction intervals for neural network ensembles. In *Proc. of the 6th Int. Joint Conf. Neural Netw. (IJCNN)*, Washington DC, DC, USA, 1999. 10-16, July.
- [91] Eddy Ilg, Ozgun Cicek, Silvio Galesso, Aaron Klein, Osama Makansi, Frank Hutter, and Thomas Brox. Uncertainty estimates and multi-hypotheses networks for optical flow. In *Proc. of 15th European Conf. on Comput. Vision (ECCV)*, pages 652–667, Munich, Germany, 2018. 8-14 Sep.



- [92] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient Langevin dynamics. In *Proc. of the 28th Int. Conf. on Mach. Learn. (ICML)*., pages 681–688, Bellevue, WA, USA, 2011. Citeseer. 28 Jun–2 Jul.
- [93] Aidan Scannell, Riccardo Mereu, Paul Chang, Ella Tamir, Joni Pajarinen, and Arno Solin. Sparse function-space representation of neural networks. *arXiv preprint arXiv:2309.02195*, 2023.
- [94] Maithra Raghu, Katy Blumer, Rory Sayres, Ziad Obermeyer, Bobby Kleinberg, Sendhil Mullainathan, and Jon Kleinberg. Direct uncertainty prediction for medical second opinions. In *Proc. of the 36th Int. Conf. on Mach. Learn. (ICML)*., pages 5281–5290, Long Beach, CA, USA, 09–15 Jun 2019. PMLR.
- [95] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In *Proc. of 14th European Conf. on Comput. Vision (ECCV)*, pages 21–37, Amsterdam, The Netherlands, 2016. Springer. October 11–14.



## **Part II**

# **Publications**



# Papers

The papers associated with this thesis have been removed for copyright reasons. For more details about these see:

<https://doi.org/10.3384/9789180754064>

**PhD Dissertations**  
**Division of Automatic Control**  
**Linköping University**

**M. Millnert:** Identification and control of systems subject to abrupt changes. Thesis No. 82, 1982. ISBN 91-7372-542-0.

**A. J. M. van Overbeek:** On-line structure selection for the identification of multivariable systems. Thesis No. 86, 1982. ISBN 91-7372-586-2.

**B. Bengtsson:** On some control problems for queues. Thesis No. 87, 1982. ISBN 91-7372-593-5.

**S. Ljung:** Fast algorithms for integral equations and least squares identification problems. Thesis No. 93, 1983. ISBN 91-7372-641-9.

**H. Jonson:** A Newton method for solving non-linear optimal control problems with general constraints. Thesis No. 104, 1983. ISBN 91-7372-718-0.

**E. Trulsson:** Adaptive control based on explicit criterion minimization. Thesis No. 106, 1983. ISBN 91-7372-728-8.

**K. Nordström:** Uncertainty, robustness and sensitivity reduction in the design of single input control systems. Thesis No. 162, 1987. ISBN 91-7870-170-8.

**B. Wahlberg:** On the identification and approximation of linear systems. Thesis No. 163, 1987. ISBN 91-7870-175-9.

**S. Gunnarsson:** Frequency domain aspects of modeling and control in adaptive systems. Thesis No. 194, 1988. ISBN 91-7870-380-8.

**A. Isaksson:** On system identification in one and two dimensions with signal processing applications. Thesis No. 196, 1988. ISBN 91-7870-383-2.

**M. Viberg:** Subspace fitting concepts in sensor array processing. Thesis No. 217, 1989. ISBN 91-7870-529-0.

**K. Forsman:** Constructive commutative algebra in nonlinear control theory. Thesis No. 261, 1991. ISBN 91-7870-827-3.

**F. Gustafsson:** Estimation of discrete parameters in linear systems. Thesis No. 271, 1992. ISBN 91-7870-876-1.

**P. Nagy:** Tools for knowledge-based signal processing with applications to system identification. Thesis No. 280, 1992. ISBN 91-7870-962-8.

**T. Svensson:** Mathematical tools and software for analysis and design of nonlinear control systems. Thesis No. 285, 1992. ISBN 91-7870-989-X.

**S. Andersson:** On dimension reduction in sensor array signal processing. Thesis No. 290, 1992. ISBN 91-7871-015-4.

**H. Hjalmarsson:** Aspects on incomplete modeling in system identification. Thesis No. 298, 1993. ISBN 91-7871-070-7.

**I. Klein:** Automatic synthesis of sequential control schemes. Thesis No. 305, 1993. ISBN 91-7871-090-1.

**J.-E. Strömberg:** A mode switching modelling philosophy. Thesis No. 353, 1994. ISBN 91-7871-430-3.

**K. Wang Chen:** Transformation and symbolic calculations in filtering and control. Thesis No. 361, 1994. ISBN 91-7871-467-2.

**T. McKelvey:** Identification of state-space models from time and frequency data. Thesis No. 380, 1995. ISBN 91-7871-531-8.

**J. Sjöberg:** Non-linear system identification with neural networks. Thesis No. 381, 1995. ISBN 91-7871-534-2.

**R. Germundsson:** Symbolic systems – theory, computation and applications. Thesis No. 389, 1995. ISBN 91-7871-578-4.

**P. Pucar:** Modeling and segmentation using multiple models. Thesis No. 405, 1995. ISBN 91-7871-627-6.

**H. Fortell:** Algebraic approaches to normal forms and zero dynamics. Thesis No. 407, 1995. ISBN 91-7871-629-2.

**A. Helmersson:** Methods for robust gain scheduling. Thesis No. 406, 1995. ISBN 91-7871-628-4.

**P. Lindskog:** Methods, algorithms and tools for system identification based on prior knowledge. Thesis No. 436, 1996. ISBN 91-7871-424-8.

**J. Gunnarsson:** Symbolic methods and tools for discrete event dynamic systems. Thesis No. 477, 1997. ISBN 91-7871-917-8.

**M. Jirstrand:** Constructive methods for inequality constraints in control. Thesis No. 527, 1998. ISBN 91-7219-187-2.

**U. Forssell:** Closed-loop identification: Methods, theory, and applications. Thesis No. 566, 1999. ISBN 91-7219-432-4.

**A. Stenman:** Model on demand: Algorithms, analysis and applications. Thesis No. 571, 1999. ISBN 91-7219-450-2.

**N. Bergman:** Recursive Bayesian estimation: Navigation and tracking applications. Thesis No. 579, 1999. ISBN 91-7219-473-1.

**K. Edström:** Switched bond graphs: Simulation and analysis. Thesis No. 586, 1999. ISBN 91-7219-493-6.

**M. Larsson:** Behavioral and structural model based approaches to discrete diagnosis. Thesis No. 608, 1999. ISBN 91-7219-615-5.

**F. Gunnarsson:** Power control in cellular radio systems: Analysis, design and estimation. Thesis No. 623, 2000. ISBN 91-7219-689-0.

**V. Einarsson:** Model checking methods for mode switching systems. Thesis No. 652, 2000. ISBN 91-7219-836-2.

**M. Norrlöf:** Iterative learning control: Analysis, design, and experiments. Thesis No. 653, 2000. ISBN 91-7219-837-0.

**F. Tjärnström:** Variance expressions and model reduction in system identification. Thesis No. 730, 2002. ISBN 91-7373-253-2.

**J. Löfberg:** Minimax approaches to robust model predictive control. Thesis No. 812, 2003. ISBN 91-7373-622-8.

**J. Roll:** Local and piecewise affine approaches to system identification. Thesis No. 802, 2003. ISBN 91-7373-608-2.

**J. Elbornsson:** Analysis, estimation and compensation of mismatch effects in A/D converters. Thesis No. 811, 2003. ISBN 91-7373-621-X.

**O. Härkegård:** Backstepping and control allocation with applications to flight control. Thesis No. 820, 2003. ISBN 91-7373-647-3.

**R. Wallin:** Optimization algorithms for system analysis and identification. Thesis No. 919, 2004. ISBN 91-85297-19-4.

**D. Lindgren:** Projection methods for classification and identification. Thesis No. 915, 2005. ISBN 91-85297-06-2.

**R. Karlsson:** Particle Filtering for Positioning and Tracking Applications. Thesis No. 924, 2005. ISBN 91-85297-34-8.

**J. Jansson:** Collision Avoidance Theory with Applications to Automotive Collision Mitigation. Thesis No. 950, 2005. ISBN 91-85299-45-6.

**E. Geijer Lundin:** Uplink Load in CDMA Cellular Radio Systems. Thesis No. 977, 2005. ISBN 91-85457-49-3.

**M. Enqvist:** Linear Models of Nonlinear Systems. Thesis No. 985, 2005. ISBN 91-85457-64-7.

**T. B. Schön:** Estimation of Nonlinear Dynamic Systems — Theory and Applications. Thesis No. 998, 2006. ISBN 91-85497-03-7.

**I. Lind:** Regressor and Structure Selection — Uses of ANOVA in System Identification. Thesis No. 1012, 2006. ISBN 91-85523-98-4.

**J. Gillberg:** Frequency Domain Identification of Continuous-Time Systems Reconstruction and Robustness. Thesis No. 1031, 2006. ISBN 91-85523-34-8.

**M. Gerdin:** Identification and Estimation for Models Described by Differential-Algebraic Equations. Thesis No. 1046, 2006. ISBN 91-85643-87-4.

**C. Grönwall:** Ground Object Recognition using Laser Radar Data – Geometric Fitting, Performance Analysis, and Applications. Thesis No. 1055, 2006. ISBN 91-85643-53-X.

**A. Eidehall:** Tracking and threat assessment for automotive collision avoidance. Thesis No. 1066, 2007. ISBN 91-85643-10-6.

**F. Eng:** Non-Uniform Sampling in Statistical Signal Processing. Thesis No. 1082, 2007. ISBN 978-91-85715-49-7.

**E. Wernholt:** Multivariable Frequency-Domain Identification of Industrial Robots. Thesis No. 1138, 2007. ISBN 978-91-85895-72-4.

**D. Axehill:** Integer Quadratic Programming for Control and Communication. Thesis No. 1158, 2008. ISBN 978-91-85523-03-0.

**G. Hendeby:** Performance and Implementation Aspects of Nonlinear Filtering. Thesis No. 1161, 2008. ISBN 978-91-7393-979-9.

**J. Sjöberg:** Optimal Control and Model Reduction of Nonlinear DAE Models. Thesis No. 1166, 2008. ISBN 978-91-7393-964-5.

**D. Törnqvist:** Estimation and Detection with Applications to Navigation. Thesis No. 1216, 2008. ISBN 978-91-7393-785-6.

**P.-J. Nordlund:** Efficient Estimation and Detection Methods for Airborne Applications. Thesis No. 1231, 2008. ISBN 978-91-7393-720-7.

**H. Tidefelt:** Differential-algebraic equations and matrix-valued singular perturbation. Thesis No. 1292, 2009. ISBN 978-91-7393-479-4.

**H. Ohlsson:** Regularization for Sparseness and Smoothness — Applications in System Identification and Signal Processing. Thesis No. 1351, 2010. ISBN 978-91-7393-287-5.

**S. Moberg:** Modeling and Control of Flexible Manipulators. Thesis No. 1349, 2010. ISBN 978-91-7393-289-9.

**J. Wallén:** Estimation-based iterative learning control. Thesis No. 1358, 2011. ISBN 978-91-7393-255-4.

**J. D. Hol:** Sensor Fusion and Calibration of Inertial Sensors, Vision, Ultra-Wideband and GPS. Thesis No. 1368, 2011. ISBN 978-91-7393-197-7.

**D. Ankelhed:** On the Design of Low Order H-infinity Controllers. Thesis No. 1371, 2011. ISBN 978-91-7393-157-1.

**C. Lundquist:** Sensor Fusion for Automotive Applications. Thesis No. 1409, 2011. ISBN 978-91-7393-023-9.

**P. Skoglar:** Tracking and Planning for Surveillance Applications. Thesis No. 1432, 2012. ISBN 978-91-7519-941-2.

**K. Granström:** Extended target tracking using PHD filters. Thesis No. 1476, 2012. ISBN 978-91-7519-796-8.

**C. Lyzell:** Structural Reformulations in System Identification. Thesis No. 1475, 2012. ISBN 978-91-7519-800-2.

**J. Callmer:** Autonomous Localization in Unknown Environments. Thesis No. 1520, 2013. ISBN 978-91-7519-620-6.

**D. Petersson:** A Nonlinear Optimization Approach to H2-Optimal Modeling and Control. Thesis No. 1528, 2013. ISBN 978-91-7519-567-4.

**Z. Sjanic:** Navigation and Mapping for Aerial Vehicles Based on Inertial and Imaging Sensors. Thesis No. 1533, 2013. ISBN 978-91-7519-553-7.



**F. Lindsten:** Particle Filters and Markov Chains for Learning of Dynamical Systems. Thesis No. 1530, 2013. ISBN 978-91-7519-559-9.

**P. Axelsson:** Sensor Fusion and Control Applied to Industrial Manipulators. Thesis No. 1585, 2014. ISBN 978-91-7519-368-7.

**A. Carvalho Bittencourt:** Modeling and Diagnosis of Friction and Wear in Industrial Robots. Thesis No. 1617, 2014. ISBN 978-91-7519-251-2.

**M. Skoglund:** Inertial Navigation and Mapping for Autonomous Vehicles. Thesis No. 1623, 2014. ISBN 978-91-7519-233-8.

**S. Khoshfetrat Pakazad:** Divide and Conquer: Distributed Optimization and Robustness Analysis. Thesis No. 1676, 2015. ISBN 978-91-7519-050-1.

**T. Ardeshiri:** Analytical Approximations for Bayesian Inference. Thesis No. 1710, 2015. ISBN 978-91-7685-930-8.

**N. Wahlström:** Modeling of Magnetic Fields and Extended Objects for Localization Applications. Thesis No. 1723, 2015. ISBN 978-91-7685-903-2.

**J. Dahlin:** Accelerating Monte Carlo methods for Bayesian inference in dynamical models. Thesis No. 1754, 2016. ISBN 978-91-7685-797-7.

**M. Kok:** Probabilistic modeling for sensor fusion with inertial measurements. Thesis No. 1814, 2016. ISBN 978-91-7685-621-5.

**J. Linder:** Indirect System Identification for Unknown Input Problems: With Applications to Ships. Thesis No. 1829, 2017. ISBN 978-91-7685-588-1.

**M. Roth:** Advanced Kalman Filtering Approaches to Bayesian State Estimation. Thesis No. 1832, 2017. ISBN 978-91-7685-578-2.

**I. Nielsen:** Structure-Exploiting Numerical Algorithms for Optimal Control. Thesis No. 1848, 2017. ISBN 978-91-7685-528-7.

**D. Simon:** Fighter Aircraft Maneuver Limiting Using MPC: Theory and Application. Thesis No. 1881, 2017. ISBN 978-91-7685-450-1.

**C. Veibäck:** Tracking the Wanders of Nature. Thesis No. 1958, 2018. ISBN 978-91-7685-200-2.

**C. Andersson Naesseth:** Machine learning using approximate inference: Variational and sequential Monte Carlo methods. Thesis No. 1969, 2018. ISBN 978-91-7685-161-6.

**Y. Jung:** Inverse system identification with applications in predistortion. Thesis No. 1966, 2018. ISBN 978-91-7685-171-5.

**Y. Zhao:** Gaussian Processes for Positioning Using Radio Signal Strength Measurements. Thesis No. 1968, 2019. ISBN 978-91-7685-162-3.

**R. Larsson:** Flight Test System Identification. Thesis No. 1990, 2019. ISBN 978-91-7685-070-1.

**P. Kasebzadeh:** Learning Human Gait. Thesis No. 2012, 2019. ISBN 978-91-7519-014-3.

**K. Radnosrati:** Time of flight estimation for radio network positioning. Thesis No. 2054, 2020. ISBN 978-91-7929-884-5.

**O. Ljungqvist:** Motion planning and feedback control techniques with applications to long tractor-trailer vehicles. Thesis No. 2070, 2020. ISBN 978-91-7929-858-6.

**G. Lindmark:** Controllability of Complex Networks at Minimum Cost. Thesis No. 2074, 2020. ISBN 978-91-7929-847-0.

**K. Bergman:** Exploiting Direct Optimal Control for Motion Planning in Unstructured Environments. Thesis No. 2133, 2021. ISBN 978-91-7929-677-3.

**P. Boström-Rost:** Sensor Management for Target Tracking Applications. Thesis No. 2137, 2021. ISBN 978-91-7929-672-8.

**A. Fontan:** Collective decision-making on networked systems in presence of antagonistic interactions. Thesis No. 2166, 2021. ISBN 978-91-7929-017-7.

**S. Parvini Ahmadi:** Distributed Optimization for Control and Estimation. Thesis No. 2207, 2022. ISBN 978-91-7929-197-6.

**F. Ljungberg:** Identification of Nonlinear Marine Systems. Thesis No. 2258, 2022. ISBN 978-91-7929-493-9.

**A. Zenere:** Integration of epigenetic, transcriptomic and proteomic data. Thesis No. 2294, 2023. ISBN 978-91-8075-068-4.

**K. Nielsen:** Localization of Autonomous Vehicles in Underground Mines. Thesis No. 2318, 2023. ISBN 978-91-8075-167-4.

**D. Arnström:** Real-Time Certified MPC: Reliable Active-Set QP Solvers. Thesis No. 2324, 2023. ISBN 978-91-8075-218-3.

The background of the entire page is an abstract illustration of a mountain range. The mountains are composed of various geometric shapes, primarily triangles and quadrilaterals, in shades of orange, red, and purple. A fine grid of thin lines is overlaid on the entire scene, creating a wireframe effect. The mountains are set against a solid orange background.

## **FACULTY OF SCIENCE AND ENGINEERING**

Linköping Studies in Science and Technology, Dissertations No. 2358, 2023  
Department of Electrical Engineering

Linköping University  
SE-581 83 Linköping, Sweden

[www.liu.se](http://www.liu.se)