

**Thesis**

**Unsicherheiten in  
machine-learning-basierten  
Surrogatmodellen für die szenariobasierte  
Validierung autonomer Systeme**

Marten Windler  
4175773

11.08.2025

Betreuer:

M.Sc. Ricardo Bosold  
M.Sc. Daniel Autenrieth

Gutachter:

Prof. Dr.-Ing. Kai Michels  
Dr.-Ing. Dennis Pierl

Nachname Windler Matrikelnr. 4175773  
Vorname Marten

## Hinweise zu den offiziellen Erklärungen

1. Die folgende Seite mit den offiziellen Erklärungen

- A) Eigenständigkeitserklärung
- B) Erklärung zur Veröffentlichung von Bachelor- oder Masterarbeiten
- C) Einverständniserklärung über die Bereitstellung und Nutzung der Bachelorarbeit / Masterarbeit in elektronischer Form zur Überprüfung durch eine Plagiatsoftware

ist entweder direkt in jedes Exemplar der Bachelor- oder Masterarbeit fest mit einzubinden oder unverändert im Wortlaut in jedes Exemplar der Bachelor- oder Masterarbeit zu übernehmen.

**Bitte achten Sie darauf, jede Erklärung in allen drei Exemplaren der Arbeit zu unterschreiben.**

2. In der digitalen Fassung kann auf die Unterschrift verzichtet werden. Die Angaben und Entscheidungen müssen jedoch enthalten sein.

### **Zu B)**

Die Einwilligung kann jederzeit durch Erklärung gegenüber der Universität Bremen, mit Wirkung für die Zukunft, widerrufen werden.

### **Zu C)**

Das Einverständnis der dauerhaften Speicherung des Textes ist freiwillig.

Die Einwilligung kann jederzeit durch Erklärung gegenüber der Universität Bremen, mit Wirkung für die Zukunft, widerrufen werden.

Weitere Informationen zur Überprüfung von schriftlichen Arbeiten durch die Plagiatsoftware sind im Nutzungs- und Datenschutzkonzept enthalten. Diese finden Sie auf der Internetseite der Universität Bremen.

Nachname Windler Matrikelnr. 4175773  
Vorname Marten

**A) Eigenständigkeitserklärung**

Ich versichere, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe. Alle Teile meiner Arbeit, die wortwörtlich oder dem Sinn nach anderen Werken entnommen sind, wurden unter Angabe der Quelle kenntlich gemacht. Gleiches gilt auch für Zeichnungen, Skizzen, bildliche Darstellungen sowie für Quellen aus dem Internet, dazu zählen auch KI-basierte Anwendungen oder Werkzeuge. Die Arbeit wurde in gleicher oder ähnlicher Form noch nicht als Prüfungsleistung eingereicht. Die elektronische Fassung der Arbeit stimmt mit der gedruckten Version überein. Mir ist bewusst, dass wahrheitswidrige Angaben als Täuschung behandelt werden.

☐ Ich habe KI-basierte Anwendungen und/oder Werkzeuge genutzt und diese im Anhang "Nutzung KI-basierte Anwendungen" dokumentiert.

**B) Erklärung zur Veröffentlichung von Bachelor- oder Masterarbeiten**

Die Abschlussarbeit wird zwei Jahre nach Studienabschluss dem Archiv der Universität Bremen zur dauerhaften Archivierung angeboten. Archiviert werden:

- 1) Masterarbeiten mit lokalem oder regionalem Bezug sowie pro Studienfach und Studienjahr 10 % aller Abschlussarbeiten
- 2) Bachelorarbeiten des jeweils ersten und letzten Bachelorabschlusses pro Studienfach und Jahr.

☐ Ich bin damit einverstanden, dass meine Abschlussarbeit im Universitätsarchiv für wissenschaftliche Zwecke von Dritten eingesehen werden darf.

☐ Ich bin damit einverstanden, dass meine Abschlussarbeit nach 30 Jahren (gem. §7 Abs. 2 BremArchivG) im Universitätsarchiv für wissenschaftliche Zwecke von Dritten eingesehen werden darf.

☒ Ich bin nicht damit einverstanden, dass meine Abschlussarbeit im Universitätsarchiv für wissenschaftliche Zwecke von Dritten eingesehen werden darf.

**C) Einverständniserklärung zur Überprüfung der elektronischen Fassung der Bachelorarbeit / Masterarbeit durch Plagiatsoftware**

Eingereichte Arbeiten können nach § 18 des Allgemeinen Teil der Bachelor- bzw. der Masterprüfungsordnungen der Universität Bremen mit qualifizierter Software auf Plagiatvorwürfe untersucht werden.

Zum Zweck der Überprüfung auf Plagiate erfolgt das Hochladen auf den Server der von der Universität Bremen aktuell genutzten Plagiatsoftware.

☐ Ich bin damit einverstanden, dass die von mir vorgelegte und verfasste Arbeit zum oben genannten Zweck dauerhaft auf dem externen Server der aktuell von der Universität Bremen genutzten Plagiatsoftware, in einer institutionseigenen Bibliothek (Zugriff nur durch die Universität Bremen), gespeichert wird.

☒ Ich bin nicht damit einverstanden, dass die von mir vorgelegte und verfasste Arbeit zum o.g. Zweck dauerhaft auf dem externen Server der aktuell von der Universität Bremen genutzten Plagiatsoftware, in einer institutionseigenen Bibliothek (Zugriff nur durch die Universität Bremen), gespeichert wird.

## Kurzfassung

When assessing the trustworthiness of mathematical models, quantifying their associated uncertainties is essential. This involves interpreting different uncertainty types, identifying their sources, selecting quantitative metrics, and applying computational methods for uncertainty estimation and solution analysis. Due to high computational demands, surrogate models are used to approximate complex real-world behaviour. Machine learning-based models offer a way to handle both aleatoric and epistemic uncertainty. This thesis demonstrates their use within a surrogate modelling framework for AUVs, employing Evidential Deep Learning for uncertainty analysis. The study shows how these uncertainties can be managed to assess their impact on aggregated uncertainty indicators. The results suggest that the validation of ML-based surrogate models benefits from such aggregated measures.

Bei der Bewertung der Vertrauenswürdigkeit mathematischer Modelle ist die Quantifizierung ihrer Unsicherheiten entscheidend. Dies umfasst die Interpretation unterschiedlicher Unsicherheitsarten, die Identifikation ihrer Quellen, die Auswahl geeigneter Metriken sowie den Einsatz rechnergestützter Methoden zur Bestimmung und Analyse. Aufgrund hoher Rechenaufwände werden Surrogatmodelle zur Näherung komplexer Zusammenhänge eingesetzt. Modelle auf Basis maschinellen Lernens ermöglichen den Umgang mit aleatorischer und epistemischer Unsicherheit. In dieser Arbeit wird ihre Anwendung im Kontext von AUVs demonstriert, wobei Evidential Deep Learning zur Unsicherheitsanalyse verwendet wird. Die Untersuchung zeigt, wie sich diese Unsicherheiten steuern lassen und welchen Einfluss sie auf aggregierte Unsicherheitskennwerte haben. Die Ergebnisse deuten darauf hin, dass die Validierung ML-basierter Surrogatmodelle von aggregierten Unsicherheitsmaßen profitiert.

# Inhaltsverzeichnis

|                                                        |           |
|--------------------------------------------------------|-----------|
| <b>Kurzfassung</b>                                     | <b>iv</b> |
| <b>1. Einleitung</b>                                   | <b>1</b>  |
| <b>2. Motivation</b>                                   | <b>2</b>  |
| <b>3. Problemstellung</b>                              | <b>4</b>  |
| <b>4. Zielsetzung</b>                                  | <b>5</b>  |
| 4.1. Zielsetzung der Arbeit . . . . .                  | 5         |
| 4.2. Forschungsfragen . . . . .                        | 5         |
| <b>5. Methodische Grundlagen</b>                       | <b>7</b>  |
| <b>6. Verwandte Arbeiten</b>                           | <b>10</b> |
| 6.1. Abgrenzung der Arbeit . . . . .                   | 10        |
| <b>7. Stand der Technik</b>                            | <b>11</b> |
| <b>8. Durchführung</b>                                 | <b>38</b> |
| <b>9. Auswertung</b>                                   | <b>72</b> |
| <b>10. Diskussion</b>                                  | <b>74</b> |
| <b>11. Fazit</b>                                       | <b>78</b> |
| 11.1. Zusammenfassung zentraler Erkenntnisse . . . . . | 78        |
| 11.2. Ausblick . . . . .                               | 79        |
| <b>Liste der Abbildungen</b>                           | <b>81</b> |
| <b>Liste der Tabellen</b>                              | <b>82</b> |
| <b>A. Szenarioselektionmethode</b>                     | <b>98</b> |

# 1. Einleitung

**Einführung** Diese Bachelorarbeit beschäftigt sich mit der Frage, wie Unsicherheiten in machine-learning-basierten Surrogatmodellen im Kontext der szenariobasierten Validierung autonomer Systeme quantifiziert und vergleichbar gemacht werden können. Die Thematik besitzt eine hohe technische Relevanz für die Verifikation komplexer Systeme und wird auch aus gesellschaftlicher Perspektive als förderwürdig eingestuft.

**Gesellschaftliche Relevanz** Das BMBF hebt in Zusammenarbeit mit dem DLR in seiner Förderstrategie zum Maschinellen Lernen die hohe Bedeutung vertrauenswürdiger und erklärbarer KI-Systeme hervor (vgl. Bundesministerium für Bildung und Forschung [1]). In sicherheitskritischen Bereichen wie der maritimen Robotik ist eine verlässliche Unsicherheitsabschätzung essenziell für den Einsatz autonomer Systeme.

**Praktischer Hintergrund** Als Grundlage für die Beteiligung im Projekt dient ein Industriepraktikum im Zeitraum Oktober bis Dezember 2024. Im Rahmen des Bachelorstudiengangs Systems Engineering wurde im Projekt „Entwicklung szenariobasierter Validierungsmethoden für autonome maritime Systeme“ (VaMAI) mitgewirkt.

**Lösungsbedarf und Anwendungskontext** Ein konkreter Bedarf ergibt sich aus dem Bericht der DAM zur Altlastenbeseitigung in Nord- und Ostsee (vgl. Deutsche Allianz für Meeresforschung [2]). Projekte von einem Verbund von Unternehmungen zielen darauf ab, autonome Unterwasserfahrzeuge (AUVs) für die Detektion und Entfernung von Munitionsresten einzusetzen.

## 2. Motivation

Für den Einsatz solcher AUVs kommen unsicherheitsbehaftete Verfahren zum Einsatz, welche in den Ingenieurwissenschaften zunehmend zur Risikoabschätzung und Entscheidungsunterstützung verwendet werden. Die Verifikation solcher Systeme verlagert sich dabei zunehmend von realen in virtuelle Testumgebungen, was den Einsatz synthetischer Daten motiviert. Im Projekt VaMAI sind diese Unsicherheiten auf verarbeitender Ebene zu unterscheiden zwischen denen der KI-Module, sowie deren der Surrogates in der Validierung jener Module.

**Maschinelles Lernen als methodische Grundlage** ML-Verfahren sind heute zentrale Werkzeuge in der Datenverarbeitung und Automatisierungstechnik (vgl. Nof et al. [3]). Im Bereich „supervised“, „unsupervised“ und „reinforcement learning“ werden Modelle trainiert, um Daten zu klassifizieren, zu strukturieren oder durch Belohnung zu optimieren. Klassische Methoden wie Klassifikation und Regression bilden das methodische Fundament dieser Ansätze.

**Unsicherheitsbehaftete Verfahren im ML-Kontext** In der *Unsicherheitsquantifizierung* (UQ) werden im ML-Kontext unter anderem zwei Arten unterschieden (vgl. Hullermeier et al. [4, S.458, „Sources of uncertainty in supervised learning“, AU Z.23-27, EU Z.23-25]):

- **EU:** *modellbedingte* Unsicherheit aufgrund unzureichender Daten oder Modellstruktur; potenziell reduzierbar.
- **AU:** *zufallsbedingte* Unsicherheit, z. B. durch Sensordatenrauschen; nicht reduzierbar durch zusätzliche Daten.

Die AU kann weiter in *homoskedastische* Unsicherheit, das heißt Unsicherheit, die für verschiedene Eingaben konstant bleibt, und *heteroskedastische* Unsicherheit unterteilt werden. Heteroskedastische Unsicherheit hängt von den Eingaben in das Modell ab, wobei einige Eingaben potenziell verrauschtere Ausgaben haben als andere (vgl. Kendall et al. [5, S. 2, Z. 1-5]).

**Zusammenhang und steuerbare Unsicherheiten** Quellen wie Hoarau et al. beschreiben eine dynamische Beziehung zwischen AU und EU bei Untersuchung zusammenhängender Modalitäten in Regression (vgl. Hoarau et al. [6, S. 2, Z.10-14]). Neuere Arbeiten gehen davon

aus, dass diese Unsicherheiten, insbesondere im Zusammenspiel mit evidenzbasierten Verfahren, unter bestimmten Bedingungen gezielt beeinflussbar beziehungsweise steuerbar sind:

„These findings indicate that both aleatoric and epistemic uncertainty can be influenced and reduced through targeted acquisition strategies, particularly in evidence-based modeling frameworks (vgl. Hoarau et al. [6, S.2, Z. 43-45]).“

**Schlussfolgerung** Die steigende Bedeutung verlässlicher Unsicherheitsanalysen in autonomen, sicherheitskritischen Systemen motiviert die vertiefte Untersuchung von ML-Verfahren unter Unsicherheitsgesichtspunkten. Ziel ist es, Modelle nicht nur leistungsfähig, sondern auch vertrauenswürdig und erklärbar zu gestalten.

## 3. Problemstellung

Die zentrale Problemstellung dieser Arbeit liegt in der Bestimmung und Analyse der **aleatorischen Unsicherheit (AU)** und **epistemischen Unsicherheit (EU)** zu einer Gesamtunsicherheit innerhalb eines sogenannten *Kritikalitätsraums*. Im Projekt VaMAI ist der Kritikalitätsraum äquivalent zum Szenarioraum. Dieser umfasst die Startparameter, sowie die Kritikalitätsbewertung des Szenarios durch eine Metrik als quantifizierte Verletzung sicherheitsrelevanter Grenzen. In diesem Fall ist der Kritikalitätsraum der Gesamttraum als höchste Raumeinheit.

In komplexen Anwendungsszenarien, insbesondere bei der simulationsgestützten Validierung autonomer Systeme, stellt der Kritikalitätsraum jene Teilmenge des gesamten Szenarienraums dar, in der das Systemverhalten als potenziell kritisch oder risikobehaftet einzustufen ist, zum Beispiel durch Kollision von Objekten.

Dabei gilt es, zwei unterschiedliche Unsicherheitsquellen präzise zu erfassen und voneinander abzugrenzen:

- **AU**, verursacht durch stochastische Variabilität in den Eingabedaten,
- **EU**, bedingt durch strukturelle Modellunsicherheit und fehlendes Wissen.

Die zentrale Herausforderung besteht darin, diese beiden Unsicherheitsarten über den Kritikalitätsraum hinweg zu erfassen, sodass eine verlässliche Gesamtunsicherheitsmasse entsteht. Diese Größe soll sowohl als Kriterium für die **Vertrauenswürdigkeit** eines Modells als auch für die **Relevanz** bestimmter Szenarien dienen können.

Klassische Verfahren zum Beispiel auf Bayesischen Ansätzen wie Bayesische Neuronale Netze (BNN) zur Unsicherheitsquantifizierung (UQ) stoßen hierbei an ihre methodischen Grenzen. Entweder modellieren sie nur eine der beiden Unsicherheitsarten oder sie berücksichtigen keine explizite räumliche Abgrenzung wie sie durch den Kritikalitätsraum gegeben ist.

Ziel dieser Arbeit ist es daher, bestehende Limitierungen zu überwinden und ein methodisches Fundament für die Unsicherheitsanalyse in sicherheitskritischen ML-Anwendungen zu schaffen.

## 4. Zielsetzung

### 4.1. Zielsetzung der Arbeit

Ziel dieser Arbeit ist die Entwicklung und Analyse eines Frameworks zur quantitativen Bewertung und Untersuchung von UQ in ML-Verfahren für Surrogatmodelle autonomer Systeme. Dabei sollen sowohl AU als auch EU systematisch unterschieden, quantifiziert und mit ingenieurwissenschaftlichen Methoden validiert werden.

Ein besonderes Augenmerk liegt auf der Konzeption einer Meta-Unsicherheit, die Rückschlüsse auf die Vertrauenswürdigkeit unterschiedlicher Modellansätze im Rahmen szenariobasierter Validierung zulässt. Damit soll ein Beitrag zur methodischen Fundierung der Unsicherheitsanalyse in sicherheitskritischen Anwendungen geleistet werden.

### 4.2. Forschungsfragen

Die folgenden Forschungsfragen (Research Questions, *RQ*) bilden den zentralen inhaltlichen Rahmen der Arbeit. Jede Frage ist so formuliert, dass sie durch geeignete Methoden, zum Beispiel Simulation, ML-Training, Unsicherheitsmodellierung, praktisch und theoretisch bearbeitet werden kann.

Die zentralen Forschungsfragen dieser Arbeit lauten:

- **RQ1** *Welche auf maschinellem Lernen basierenden Surrogatmodelle sind für die Quantifizierung von Unsicherheiten geeignet?*
- **RQ2** *Welche allgemeinen und modellspezifischen Faktoren beeinflussen das Lernen von Unsicherheiten in ML-Modellen?*
- **RQ3** *Wie können diese Einflussfaktoren abgeschwächt werden, um die Unsicherheitsabschätzung zu verbessern?*
- **RQ4** *Inwieweit kann ML-basierte Unsicherheitsquantifizierung zuverlässig den in realen Anwendungsszenarien beobachteten Unsicherheiten entsprechen?*

Die spezifischen Aufgaben dieser Arbeit umfassen:

- **A1** Durchführung einer umfassenden Literaturübersicht über UQ-Techniken in der ML-basierten Surrogatmodellierung.
- **A2** Identifizierung und Analyse geeigneter Unsicherheitsmetriken für Surrogatmodelle, wobei zwischen aleatorischer und epistemischer Unsicherheit unterschieden wird.
- **A3** Vergleich modellspezifischer und anwendungsspezifischer Bewertungsmetriken für die Quantifizierung von Unsicherheit.
- **A4** Untersuchung und Vergleich verschiedener UQ-Ansätze für ML-basierte Surrogatmodelle, insbesondere Bayes'sche neuronale Netze und konforme Vorhersagen.
- **A5** Untersuchung des theoretischen Einflusses verschiedener Faktoren auf die Unsicherheits-schätzungen ausgewählter Modelle.
- **A6** Implementierung und Validierung der entwickelten Unsicherheitsmetriken und Modellierungstechniken in einer realistischen Anwendung, wie z. B. der Unsicherheitsabschätzung in kritischen Regionen von Simulationsdaten.

## 5. Methodische Grundlagen

### Machine Learning

ML bezeichnet die Klasse von Algorithmen  $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ , die aus Daten  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$  ein Modell  $f_\theta$  lernen, das Eingaben  $x \in \mathcal{X}$  Vorhersagen  $\hat{y} \in \mathcal{Y}$  zuordnet. Dies geschieht durch Optimierung der Parameter  $\theta$  mittels einer Fehlerfunktion  $\mathcal{L}_{\text{loss}}(y, \hat{y})$ .

### Architekturen

Die Laufzeit zur Anpassung der Gewichtungen eines neuronalen Netzwerks (NN) an eine Zielfunktion kann durch unterschiedliche Architekturen für spezifische Aufgaben optimiert werden. Die Kürzel der Architekturen verweisen jeweils auf ihre charakteristische Struktur.

### NN – Neuronales Netzwerk

Ein NN approximiert eine Funktion  $f_\theta(x) = \sigma(Wx + b)$ . Dabei ist  $\sigma$  eine nichtlineare Aktivierungsfunktion, typischerweise *ReLU* oder *Tanh*. Die Parameter des Modells sind gegeben durch  $\theta = \{W, b\}$ , wobei:

- $W$  die Gewichtsmatrix darstellt, zum Beispiel

$$W = \begin{pmatrix} w_{11} & w_{12} & \dots & w_{1j} \\ w_{21} & w_{22} & \dots & w_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ w_{i1} & w_{i2} & \dots & w_{ij} \end{pmatrix} \in \mathbb{R}^{i \times j}$$

- $b$  ein Bias-Term ist, der unabhängig vom Eingangssignal die Aktivierung verschiebt.

Eine Schicht in einem NN besteht aus:

1. einer linearen Transformation:  $z = Wx + b$ ,
2. einer nichtlinearen Aktivierung:  $f(x) = \sigma(z)$ .

### DNN - Deep Neural Network

Ein **Deep Neural Network (DNN)** erweitert ein NN um mehrere versteckte Schichten, Hidden Layer. Wobei  $L$  die Tiefe des NN äquivalent ist zur Anzahl gewichteter Schichten des NN:

$$f(x) = \sigma_L (W_L \cdot \sigma_{L-1} (\dots \sigma_1 (W_1 \cdot x + b_1) \dots) + b_L)$$

Neben der linearen Transformation erlaubt ein DNN durch das Propagieren über mehrere versteckte Schichten die Modellierung komplexer Nichtlinearitäten und somit ein tieferes Lernen relevanter Merkmale.

### BNN - Bayesian Neural Network

Ein **Bayesian Neural Network (BNN)** modelliert die Gewichte  $\theta$  als Zufallsvariablen  $p(\theta)$ . Die Vorhersage entspricht dem Erwartungswert über alle möglichen Modelle:

$$p(y | x, D) = \int p(y | x, \theta) \cdot p(\theta | D) d\theta$$

Dadurch wird die EU über die Posterior-Verteilung der Gewichte quantifizierbar.

### ENN - Evidential Deep Neural Network

Ein **Evidential Deep Neural Network (ENN)** lernt direkt die Parameter einer Wahrscheinlichkeitsverteilung über Unsicherheiten durch eine Evidenzgröße  $E$ , zum Beispiel innerhalb einer Normal-Inversen-Gamma-Verteilung (NIG)  $p(y | \mu, \sigma^2, e)$ . Vorteilhaft ist, dass AU und EU ohne explizites Monte-Carlo-Sampling bestimmt werden können – im Gegensatz zum bayesschen Ansatz.

#### Parameterbedeutung:

- $\mu$ : Erwartungswert der Zielgröße, zum Beispiel Position, Fehlermaß
- $v$ : Skalenparameter (inverse Varianz), modelliert die Konfidenz
- $\alpha$ : Formparameter der NIG-Verteilung, verknüpft mit epistemischer Evidenz
- $\beta$ : Skalenparameter der Varianz, verknüpft mit aleatorischer Evidenz
- $E = \alpha - 1$ : Gesamt-Evidenz, als Vertrauensmaß des Modells

Die kombinierte Unsicherheit ergibt sich durch:

$$\text{Gesamtunsicherheit} = \underbrace{\frac{\beta}{\alpha - 1}}_{\text{epistemisch}} + \underbrace{\frac{\beta}{v(\alpha - 1)}}_{\text{aleatorisch}} \quad (5.1)$$

Diese Form erlaubt eine einmalige Modellvorhersage zur gleichzeitigen Schätzung beider Unsicherheitsarten.

## 6. Verwandte Arbeiten

**Ulmer et al. (2023)** Ulmer et al. liefern eine umfassende Übersicht über EDL und dessen Einsatz zur UQ (vgl. Ulmer et al. [7]). Die Autoren unterscheiden insbesondere zwischen *Prior Networks*, die Unsicherheiten durch separate Vorhersage von Verteilungsparametern modellieren, und *Posterior Networks*, die auf bayesianischer Approximation basieren. Die Arbeit stellt verschiedene Loss-Funktionen vor, darunter Evidential Losses für Regression und Klassifikation, und diskutiert Vor- und Nachteile von EDL gegenüber klassischen bayesianischen Verfahren wie Variationsinferenz (VI) oder Deep Ensembles. Darüber hinaus beleuchtet der Survey Herausforderungen wie Kalibrierung, Interpretierbarkeit und die praktische Anwendbarkeit von EDL-Methoden. Damit bietet Ulmer et al. einen aktuellen Stand der Forschung und eine fundierte Einordnung evidenzbasierter Verfahren in das Spektrum der UQ.

### 6.1. Abgrenzung der Arbeit

Die vorliegende Arbeit behandelt nicht die Entwicklung neuer ML-Architekturen, sondern setzt bestehende Modellstrukturen EDL gezielt zur Unsicherheitsbewertung in szenariobasierten Surrogatmodellen ein. Es wird keine generelle Systemidentifikation, Validierung oder Regelung entwickelt, sondern der Fokus liegt auf der Analyse und Interpretation von Unsicherheiten, nicht auf der dynamischen Kompensation. Es erfolgt keine Exploration vollständiger Regelungssysteme oder klassischer MPC-Implementierungen, sondern eine Modellbetrachtung im Kontext unsicherer Eingaberaum-Szenarien. Ebenso werden keine physikalischen AUV-Experimente oder Realzeitsysteme implementiert, sondern Simulationen beziehungsweise modellbasierte Methoden betrachtet. Die Arbeit grenzt sich außerdem ab von reiner Fehlerdiagnose, klassischer statistischer Qualitätskontrolle, vollständigen Safety-Cases oder normativer Zertifizierungsmethodik.

## 7. Stand der Technik

Der aktuelle Stand im Bereich ML für die szenariobasierte Validierung autonomer Systeme setzt stark auf datenbasierte Modelle, um komplexe parametrisierte Situationen vorherzusagen, zum Beispiel im Verkehr. Das ingenieurtechnische Ersatzmodell als Surrogat soll Risiken bewerten und die Plausibilität von Szenarien prüfen, zum Beispiel wann eine Steuerung eine Situation vorhersagen soll. Dazu wird im folgenden Rahmen der Forschungsfragen und spezifischen Aufgaben erörtert.

**R1** Welche auf ML basierenden Surrogatmodelle sind für die Quantifizierung von Unsicherheiten geeignet?

| Surrogatmodell  | Eignungsgrund                                                                                                                                                   | Quellen                                                                                                                          |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------|
| GPR             | 1.1. Intrinsische probabilistische Ausgabe<br>1.2. Exakte Varianzschätzung für kleine Datensätze<br>1.3. Kernel-basierte Unsicherheitscharakterisierung         | Rasmussen et al. [8, Kap. 2]<br>Rasmussen et al. [8, S. 16–17]<br>Rasmussen et al. [8, Kap. 2.2]<br>Rasmussen et al. [8, Kap. 4] |
| Deep Ensembles  | 2.1. Redundante Modellierung reduziert Overconfidence<br>2.2. Einfache Integration in existierende Architekturen<br>2.3. Robust gegen Out-of-Distribution-Daten | Lakshminarayanan et al. [9]<br>Lakshminarayanan et al. [9, S. 3]<br>Lakshminarayanan et al. [9, S. 5]<br>Ovadia et al. [10]      |
| Hybrid-Methoden | 3.1. Kombiniert Stärken von Blackbox-Modellen und GPR<br>3.2. Nachrüstbar auf existierende Modelle<br>3.3. Recheneffiziente Unsicherheitspropagation            | Perdikaris et al. [11]<br>Perdikaris et al. [11, S. 5–6]<br>Perdikaris et al. [11, S. 6]<br>Lui et al. [12]                      |
| BNN             | 4.1. Systematische Integration von Priors<br>4.2. Theoretisch fundierte Parameterunsicherheit<br>4.3. Natürliche Regularisierung durch Bayes'sche Inferenz      | Gal et al. [13]<br>Gal et al. [13, Kap. 2.3]<br>Gal et al. [13, S. 40–41]<br>Gal et al. [13, S. 41–42]                           |

Tabelle 7.2.: R1 Vergleich von Unsicherheitsquantifizierungsmethoden in ML

**R2** Welche *allgemeinen* und *modellspezifischen* Faktoren beeinflussen das Lernen von Unsicherheiten in ML-Modellen?

**R2.1** *Allgemeine Faktoren* welche das Lernen von Unsicherheiten in ML-Modellen beeinflussen.

| Unsicherheitsdimensionen   | Mögliche Werte                                                                                                  | Quellen                      |
|----------------------------|-----------------------------------------------------------------------------------------------------------------|------------------------------|
| 1. Lokalisierung im System | Umgebung<br>Modellierung<br>Funktionen<br>Ziele<br>Verfügbare Ressourcen                                        | Kreutz et al. [14, S. 47–52] |
| 2. Natur                   | Epistemisch<br>Aleatorisch                                                                                      | Kreutz et al. [14, S. 54]    |
| 3. Grad der Unsicherheit   | Keine Unsicherheit<br>Mangel an Wissen<br>Mangel an Bewusstsein<br>Mangel an Fähigkeit, Bewusstsein zu erlangen | Kreutz et al. [14, S. 56–58] |
| 4. Entstehungszeit         | Anforderungsdefinition<br>Entwicklungszeit<br>Laufzeit                                                          | Kreutz et al. [14, S. 60]    |

Tabelle 7.4.: R2 Allg. Faktoren Klassifizierung Unsicherheit KI-Systeme

## R2.2 Modellspezifische Faktoren die das Lernen von Unsicherheiten in ML-Modellen beeinflussen.

| BNN                        | Modellspezifische Faktoren                                                                                    | Quellen                        |
|----------------------------|---------------------------------------------------------------------------------------------------------------|--------------------------------|
| 1. Prior-Verteilungen      | Wahl der Priors beeinflusst die Unsicherheit<br>Beeinflusst die Vorhersagegenauigkeit                         | Gal et al. [13, Kap. 2.3]      |
| 2. Likelihood-Funktion     | Bestimmt, wie Unsicherheit in den Beobachtungen modelliert wird<br>Beispielsweise durch Gaussian-Verteilungen | Blundell et al. [15, Kap. 3]   |
| 3. Posterior Approximation | MCMC oder VI werden genutzt<br>Schätzt den Posterior und die Unsicherheit                                     | Gal et al. [13, S. 40–41]      |
| 4. Regularisierung         | Hilft, Overfitting zu vermeiden<br>Unterstützt die Kontrolle der Unsicherheit                                 | Gal et al. [13, S. 41–42]      |
| 5. Bayesische Optimierung  | Optimiert Hyperparameter<br>Berücksichtigt dabei die Unsicherheit                                             | Rasmussen et al. [8, Kap. 7.4] |
| 6. Datenqualität           | Schlechte Datenqualität erhöht die Unsicherheit<br>Gute Qualität kann Unsicherheit verringern                 | Bishop et al. [16, S. 16]      |
| 7. Maßstab                 | Der Maßstab des Modells beeinflusst die Unsicherheitsquantifizierung                                          | Gal et al. [13, S. 29–31]      |

Tabelle 7.6.: R2.2 Modellspezifische Faktoren zur Unsicherheitsklassifizierung BNN

| ENN                                | Modellspezifische Faktoren                                                         | Quellen                    |
|------------------------------------|------------------------------------------------------------------------------------|----------------------------|
| 1. Prior-Funktion                  | Beeinflusst, wie Unsicherheit über Daten und Parameter modelliert wird.            | Sensoy et al. [17]         |
| 2. Evidenzfunktion                 | Modelliert Unsicherheit mit Evidenzparametern wie $\alpha$ , $\beta$ , und $\nu$ . | Sensoy et al. [17, S. 2–3] |
| 3. Posterior Approximation         | Näherung der Posterior-Verteilung zur Unsicherheitsbestimmung.                     | Sensoy et al. [17, S. 4]   |
| 4. Evidenz Optimierung             | Optimierung der Evidenzparameter während des Trainings.                            | Sensoy et al. [17, S. 5]   |
| 5. Evidenzbasierte Regularisierung | Regularisiert Evidenzparameter, um Overfitting zu vermeiden.                       | Sensoy et al. [17, S. 6]   |
| 6. Multimodalität                  | Modelliert mehrere konkurrierende Unsicherheitsvorhersagen gleichzeitig.           | Sensoy et al. [17, S. 6–7] |
| 7. Dateninhärente Unsicherheit     | Unsicherheit, die direkt aus den Daten selbst resultiert.                          | Kendall et al. [18]        |
| 8. Datenqualität                   | Schlechte Datenqualität erhöht die Unsicherheit im Modell.                         | Kendall et al. [18]        |
| 9. Netzwerktiefe                   | Die Tiefe beeinflusst die Fähigkeit, Unsicherheit zu erfassen.                     | Sensoy et al. [17, S. 6]   |

Tabelle 7.8.: R2.2 Modellspezifische Faktoren zur Unsicherheitsklassifizierung ENN

**R3** Wie können diese Einflussfaktoren abgeschwächt werden, um die Unsicherheitsabschätzung zu verbessern?

Wie kann man *allgemeine Faktoren* beim Lernen von Unsicherheiten in ML-Modellen abschwächen?

| Unsicherheitsdimensionen   | Mögliche Abschwächung                                                                                          | Quellen                        |
|----------------------------|----------------------------------------------------------------------------------------------------------------|--------------------------------|
| 1. Lokalisierung im System | Verbesserung der Modellgenauigkeit und der Ressourcennutzung durch gezielte Datensammlung und Modellanpassung. | Kreutz et al. [14, S. 47–52]   |
| 2. Natur                   | Einsatz von hybriden Modellen, die sowohl EU als auch AU modellieren können.                                   | Perdikaris et al. [11, S. 6–9] |
| 3. Grad der Unsicherheit   | Erhöhung des Modellspezifikationsgrades, zum Beispiel durch Mehrfachmodellierung und UQ.                       | Ovadia et al. [10, S. 3]       |
| 4. Entstehungszeit         | Frühzeitige Einbindung von Unsicherheitsmodellierung in die Anforderungsdefinition und Entwicklungszeit.       | Kreutz et al. [14, S. 60]      |

Tabelle 7.10.: R3 Abschwächung allgemeiner Faktoren KI-Systeme

Wie kann man *modellspezifische Faktoren* beim Lernen von Unsicherheiten in ML-Modellen abschwächen?

| Modellspezifische Faktoren | Abschwächung der Unsicherheitsabschätzung                                     | Quellen                        |
|----------------------------|-------------------------------------------------------------------------------|--------------------------------|
| 1. Prior-Verteilungen      | Auswahl robusterer Priors und Anpassung der Priors an die Daten.              | Gal et al. [13, Kap. 2.3]      |
| 2. Likelihood-Funktion     | Optimierung der Likelihood-Modelle, z. B. durch mehrdimensionale Likelihoods. | Blundell et al. [15, Kap. 3]   |
| 3. Posterior Approximation | Verwendung effizienter Verfahren wie VI.                                      | Gal et al. [13, S. 40–41]      |
| 4. Regularisierung         | Anwendung stärkerer Regularisierungstechniken.                                | Gal et al. [13, S. 41–42]      |
| 5. Bayesische Optimierung  | Optimierung von Hyperparametern unter Unsicherheit.                           | Rasmussen et al. [8, Kap. 7.4] |
| 6. Datenqualität           | Verbesserung der Datenqualität durch Bereinigung.                             | Bishop et al. [16, S. 16]      |
| 7. Maßstab                 | Skalierung der Modelle auf größere Datenmengen.                               | Gal et al. [13, S. 29–31]      |

Tabelle 7.12.: R3.1 Abschwächung modellspezifischer Faktoren BNN

**R4** Inwieweit kann ML-basierte UQ zuverlässig den in realen Anwendungsszenarien beobachteten Unsicherheiten entsprechen?

Gemäß „Guide to the Uncertainty of Measurement“ (GUM) muss ingenieurstechnisch eine aussagekräftige Stichprobe über das reale Anwendungsszenario beobachtet worden sein, um eine synthetische Charakterisierung zur realen Messunsicherheit empirisch zu modellieren. Die Zuordnung von AU, EU zu den im GUM beschriebenen Unsicherheitskomponenten erfolgt erläuternd im Anhang. Ein möglicher Ansatz zur Realitätsbewertung ist dann die softwaretechnische, datengetriebene Modellierung zur Signalrekonstruktion des Szenarios. Dabei müssen alle identifizierten Unsicherheitsquellen formal beschrieben sowie nach messtechnischen Kriterien kategorisiert werden, um die abgeleiteten Daten mit fundierten Vertrauens- und Zuverlässigkeitsaussagen belegen zu können.

Bezieht man das auf gängige Optimierungen ML-basierter Surrogatmodelle im synthetischen Eingaberaum, liefern Lanini et. al ein Framework zur Bewertung von R4 zur Optimierung in

| Modellspezifische Faktoren         | Abschwächung der Unsicherheitsabschätzung                                         | Quellen                    |
|------------------------------------|-----------------------------------------------------------------------------------|----------------------------|
| 1. Prior-Funktion                  | Verwendung flexibler Prior-Funktionen, die an die Daten angepasst sind.           | Sensoy et al. [17]         |
| 2. Evidenzfunktion                 | Einsatz multimodaler Evidenzfunktionen.                                           | Sensoy et al. [17, S. 3–4] |
| 3. Posterior Approximation         | Nutzung präziser Näherungsverfahren wie MCMC.                                     | Sensoy et al. [17, S. 4]   |
| 4. Evidenz Optimierung             | Optimierung der Evidenzparameter zur Vermeidung hoher Unsicherheit.               | Sensoy et al. [17, S. 5]   |
| 5. Evidenzbasierte Regularisierung | Stärkere Regularisierung der Evidenzparameter.                                    | Sensoy et al. [17, S. 6]   |
| 6. Multimodalität                  | Einbindung multimodaler Modelle zur Abbildung verschiedener Unsicherheitsquellen. | Sensoy et al. [17, S. 6–7] |
| 7. Dateninhärente Unsicherheit     | Reduktion des Rauschens in den Daten.                                             | Kendall et al. [18]        |
| 8. Datenqualität                   | Verbesserung der Datenqualität durch Dataaugmentation.                            | Kendall et al. [18]        |
| 9. Netzwerktiefe                   | Anpassung der Netzwerktiefe an die Komplexität der Aufgabe.                       | Sensoy et al. [17, S. 6]   |

Tabelle 7.14.: R3.2 Abschwächung modellspezifischer Faktoren ENN

einer Anwendungsdomäne. Ihre Recherche differenziert zur Beantwortung weiterhin datenbasierte und modellbasierte Unsicherheitsmetriken (vgl. Lanini et al. [19]).

Manchingal et al. distanzieren sich von der Optimierung auf Anwendungsdomänen. Als Paradigmenwechsel beschreiben Sie epistemische KI. Im Kern soll nicht nur auf Wissen in Form von Daten, sondern auch auf Nicht-Wissen trainiert werden (vgl. Manchingal et al. [20]).

### Modell Evidentielles Wissen

Ein in der Literatur üblicher Ansatz ist das Trainieren von ENN auf Evidentiell Wissen zur Optimierung dessen in der Anwendungsdomäne. Manchingals Grafiken liefern die Übersicht für in VaMai in Betracht gezogene UQ-Techniken in R.4.1, R.4.2, R.4.3. Dabei können insbesondere die Wahrscheinlichkeitsräume mittels Jøsangs Subjektiver Logik (SL) umfassend definiert werden.

„Difference to [...] bayesian theory [is] AU, EU is *explicit* in SL [...]“

[21, vgl. Jøsang S. 42, Z.38 ff.]

Jøsang liefert den Zusammenhang für BNN (vgl. Jøsang [21, vgl. S.42]). Er soll hier auch für EDL gewählt werden. Jøsang erweitert AU, EU als Unsicherheiten erster Ordnung zu Unsicherheiten zweiter Ordnung „Aleatoric Opinion“ (AOPS), „Epistemic Opinion“ (EOPS) (vgl. Jøsang [21, S.22, Z.10-16, S.22 Z. 17-26, S.23, Z. 5-, S.23 Z.9-12]).

SL gibt eine klare Vergleichbarkeits-Hierarchie zur Mengenmodellierung für Zwischenzustände zur Unwissenheit trotz Evidenz für ENN zur Eingrenzung von „Zero Evidence Regions“ (vgl. [21, vgl. S.50]) unabhängig von gewählten in der Literatur stark individualisierten, variierenden Loss-Funktionen (vgl. Ulmer et al. [7]). Für VaMai interessant sind die Limitierungen evidentiellen Wissens in der Visualisierung vom binomiellen zum multinomiellen Modell, die Zuordnung dieser zu Dirichlet PDF's erster Ordnung und schließlich zweiter Ordnung in der Hyperebene zum Erarbeiten einer Meta-Unsicherheit [21, S.39, 3.6.1 „hyper-opinion representation“, „hyper-opinion and dirichlet HPDF“] in der Szenario-Beobachtung [21, S.39, Tabelle 3.3].

Jøsang füllt damit, „[the] missing uniform definition of what is not known“ [21, vgl. S.42] für BNN, ENN. Unabhängig von Jøsangs Logik wird deutlich wie der Wahrscheinlichkeitsraum explizit evidentiell ausmodelliert, und damit über jedwede Metrik und Räume *qualitativ* vergleichbar gemacht werden kann.

Die Wahl der Logik für propabilistische Funktionen, die Glauben ausdrücken sollen, ist in der wissenschaftlichen Literatur bis heute umstritten in Orientierung an erstens Bayes und zweitens Dempster Shafers Theorie (vgl. Bayes, Dempster [22, 23, 24]. Probabilistische Mengenmodellierung als konvexe Menge von Wahrscheinlichkeitsverteilungen der Statistik (SLT) dient als ML-Grundlage (vgl. Caprio et al. [25]). Wobei Caprio et al. kürzliche Annahmen zur *Domain Generalization* (DG) sowie *Ground Truth* (GT) verdeutlicht. Ebenso wird dort der VaMai begleitende Konflikt zwischen *Domain Knowledge* (DK) und DG aufgezeigt. Aktuell höchstes Ziel ist die autonome zeitkontinuierliche Akquise neuer Domänen eines aufgesetzten Systems. VaMai verfolgt Aspekte der DG in erstens hohen Dimensionalitäten, zweitens Generifizierbarkeit, drittens Modellagnostizität, sowie viertens Generalisierbarkeit über Benchmarks zur

Validierung über die freie Wahl von Datensätzen mit dem Ausgangspunkt von Aminis Deep Evidential Regression (vgl. Amini et al. [26]).

Hullermeier et al. zeigen dabei dahinterliegende probabilistische Mengenmodellierung (vgl. Hullermeier et al. [27]) in der Anwendung auf für Evidentielle Klassifikation, Denoeux für Evidentielle Regression (vgl. Denoeux [28]), jeweils in Bezug auf AU, EU. In evidentieller Klassifikation äußert sich AU in der gleichmäßigen, flachen Verteilung der Massenfunktion auf die Klassen, EU in einer hohen Masse auf der Vereinigungsmenge aller Klassen (auf Unbestimmtheit, Ignoranz), was auf mangelndes Wissen des Modells hindeutet. In Evidentieller Regression äußert sich AU in der Breite der Vorhersage-Intervalle, EU in der Streuung oder „Unsicherheit der Unsicherheiten“, also darin, wie sicher das Modell selbst über seine Unsicherheitsangaben ist. Im Gegensatz zur evidenzbasierten Klassifikation ist evidenzbasierte Regression methodenunabhängig stets mit einer epistemischen Unsicherheit verbunden, die aus der Diskretisierung des kontinuierlichen Signals und der jeweiligen Modellannahme des Autors resultiert. Amini zeigt diese Zusammenhänge kompakt: „Die evidenzbasierte Regression lernt gleichzeitig ein kontinuierliches Ziel mit aleatorischer (Daten) und epistemischer (Modell) Unsicherheit. Bei einer Eingabe wird das Netz so trainiert, dass es die Parameter  $[\alpha, \beta, \sigma, \nu]$  einer Evidenzverteilung vorhersagt, die eine Wahrscheinlichkeitsverteilung höherer Ordnung [zur Zeit i.d.R. zweiter Ord.] über die einzelnen Likelihood-Parameter  $(\mu, \sigma^2)$  modelliert.“ (vgl. Amini et al. [26, S.1, Abbildung 1]).

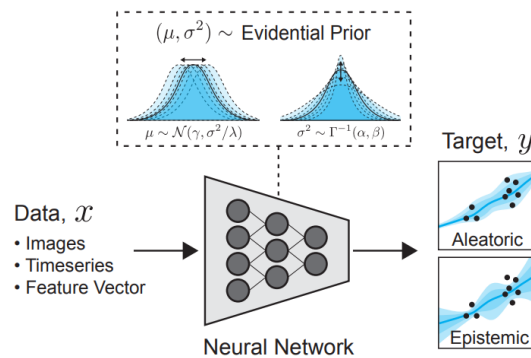


Abbildung 7.1.: R4.1 „Evidential Deep Learning“ nach Amini et al. 2020

Als weiterer Methodenbaustein soll „*epistemic learning*“ nach Manchingal eingeführt werden. R4.1 stellt dies gegenüber dem konventionellem Lernprozess. Ein Datensatz (in R.4.2 „Training set“)  $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$  trägt nach Manchingal implizit die Kosten zum Erhalt der ML Label-Information des Datensatzes als Akquisitions-Funktion. Nicht jeder Datenpunkt bringt demnach denselben Informationsgewinn. Die „Datum“-Punkte stehen für unbeschriftete Datenbeispiele. Ihre Position in einem Diagramm würde zeigen, wie nützlich im Sinne der Reduktion EU und

wie kostenintensiv es wäre, gerade diese Punkte zu labeln, um das Vertrauen im Modell gezielt zu steigern. Das Label-Kosten-Verhältnis ist für Klassifikation sowie Regression gültig.

$$\text{Acquisition}(x) := \frac{I(y; \theta \mid \mathcal{D}, x)}{c(x)} \Big|_{\text{BNN}}$$

$$:= \frac{\text{Var} [\mu \mid x]}{c(x)} \Big|_{\text{ENN}}$$

Es ergibt sich im Active Learning als ML-Untergebiet allgemein nach Kosten-Nutzen-Rechnung mit Informationsgewinn als wählbare Unsicherheitsgröße - hier Information Gain für BNN, Varianz für ENN - des jeweiligen Modells pro Kosten wie Houlby für BNN aufzeigt (vgl. Houlby [29, S.9 „Bayesian active learning by disagreement (BALD)“]).

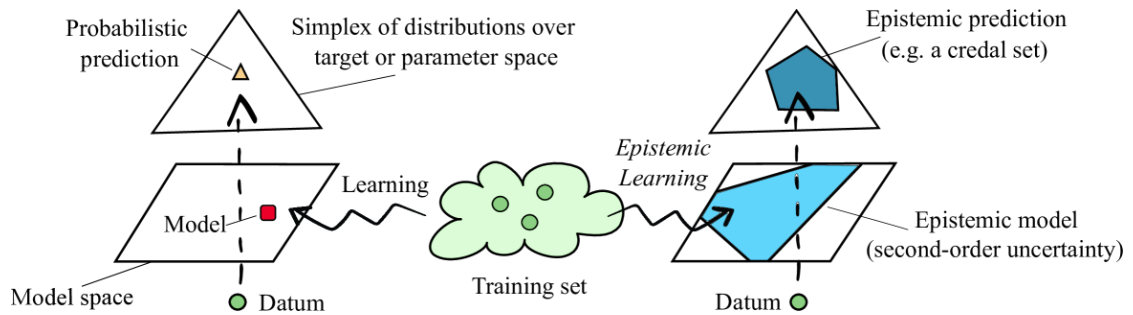


Abbildung 7.2.: R4.2 „Epistemic Learning“ nach Manchingal et al. 2025

Akquisitions-kriterienbasiert wählt evidentiell es Lernen Punkte niedriger Kosten, hohen Nutzens gewählter Unsicherheitsgröße und nicht nur unsichere Punkte wie im klassischen ML-Prozess. Kanten „Learning“, „Epistemic Learning“ der Grafik R.4.2 repräsentieren das Zuordnen eines Punktes aus dem Datenset in den jeweiligen Modellraum. Flächen der jeweiligen Dimension markieren, die mit der Unsicherheit assoziierten Sigma-Bereiche im epistemischen Modell, im Gegensatz zur klassischen Punktdarstellung. In der Grafik Punkt zu Flächen, allgemein der Dimensions-Unterschied ist ebenso im Ziel- oder Parameterraum vorhanden. Das klassische Modell liefert hier eine probabilistische Punktvorhersage im  $n$ -Simplex der Mannigfaltigkeit der Verteilungen über den Ziel- oder Parameterraum. Das epistemische Modell liefert einen Unterraum im  $n$ -Simplex der Mannigfaltigkeit der Verteilungen als probabilistische Fläche  $n$ -ter Dimension, der über Jøsangs SL ausdefiniert werden kann (vgl. Jøsang [subjective logic]).

Abschließend für Evidentiell es Wissen im Modell sollen nach R.4.3 die Hauptansätze nach Manchingal zur UQ zusammengefasst werden wie sie auch in VaMai in Betracht gezogen werden, wobei diese Arbeit den Fokus auf den evidentiellen Ansatz legt.

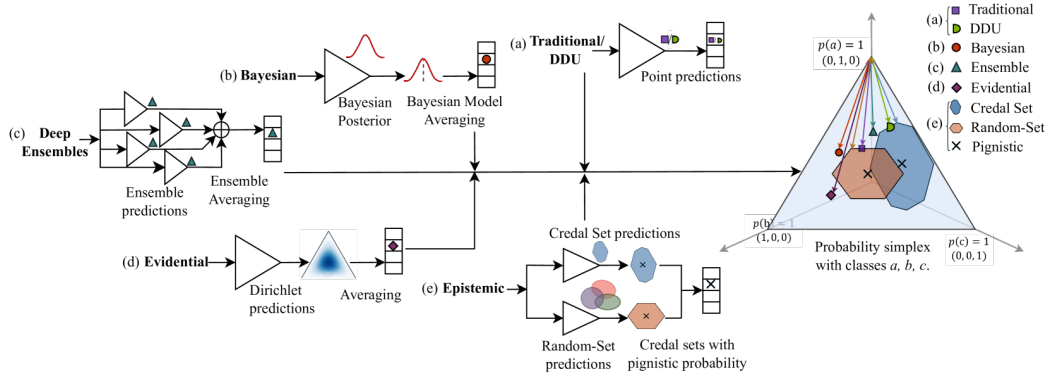


Abbildung 7.3.: R4.3 „Major approaches to uncertainty in AI“ nach Manchingal et al. 2025

**Frequentistisch:** Modelliert nur AU. Keine EU enthalten. Ergebnis sind Punkt-, Intervallschätzungen wie  $\hat{y} \pm \text{CI}$ . Beispiel ist die klassische Regression.

**Bayesisch:** Modelliert EU über Wahrscheinlichkeitsverteilungen auf Parametern  $\theta$ . Ergebnis: posterior  $p(\theta \mid \mathcal{D})$ . Beispiel sind BNN, Dropout.

**Evidentiell:** Modelliert AU und EU über Hyper-Verteilungen wie Dirichlet, Normal-Inverse-Gamma. Ergebnis sind Hyperparameter  $(\alpha, \beta, \nu, \sigma)$ . Die Meta-Unsicherheit ist über Varianz der Hyperparameter messbar.

**Plausibilitätsmengen:** Modelliert EU als konvexe Mengen von Wahrscheinlichkeitsverteilungen. Ergebnis ist  $\mathcal{P} = \{P \in \text{probabilities} : \text{constraints}\}$ . Erlaubt Meta-Unsicherheit darüber, welche Verteilungen plausibel sind.

### Modell evidentielles Nicht-Wissen

Während der vorige Abschnitt evidentiell Wissen behandelt, soll dieser komplementär die Modellierung expliziten Nicht-Wissens für ENN, insbesondere für Eingaben außerhalb des Trainingsbereichs, behandeln. Diese Technik adressiert primär EU, indem das Modell gezielt darauf trainiert wird, bei unbekannten Eingaben geringe Evidenz auszugeben. Dazu werden OOD-Beispiele in das Training integriert und die Loss-Funktion um eine KL-Divergenz ergänzt, welche die evidenzielle Dirichlet-Verteilung möglichst nahe an eine uniforme Verteilung bringt (vgl. Malinin et al. [30, S.3, Z.22-28], Amini et al. [26, S.4, Z.10-16], Ovadia et al. [10, S.4, Z.5-15]):

$$\text{Loss}_{\text{total}} = \text{Loss}_{\text{in}} + \lambda \cdot \text{KL}(\text{Dir}(\alpha) \parallel \text{Uniform}). \quad (7.1)$$

Dadurch wird verhindert, dass das Modell bei unbekannten Eingaben fälschlich hohe Evidenzwerte signalisiert, obwohl es in diesen Regionen unsicher ist. Bei Regression kann analog die Varianz gezielt maximiert werden, indem  $\alpha$  und  $\beta$  auf niedrige Werte gesetzt werden. So lernt das ENN, Nicht-Wissen explizit durch geringe Evidenz und hohe Unsicherheit auszudrücken. Dieses Konzept steht in engem Zusammenhang mit Jøsangs „epistemic opinion“, die bei fehlender Evidenz eine gleichmäßige Unwissenheitsverteilung ausdrückt (vgl. Jøsang [21, S.22]). Die explizite Modellierung evidenziellen Nicht-Wissens bietet so die Möglichkeit Modelle robuster gegen unbekannte Situationen zu machen. Die Praxis liefert dafür Benchmarkmethoden wie CleverHans zur Simulation eines „Angriffs“ auf das Modell mit fälschlichen Angaben (vgl. Papernot et al. [31]).

**A1** Durchführung einer umfassenden Literaturübersicht über UQ-Techniken in der ML-basierten Surrogatmodellierung.

Der Beitrag zur umfassenden Literaturübersicht über UQ-Techniken in der ML-basierten Surrogatmodellierung wird auf ENN eingegrenzt. In ENN soll zwischen *In-* und *Exklusionskriterien* unterschieden werden.

| Kriterium           | Beschreibung                                                                                                                                                                                                                              |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Inklusionskriterien | Evidential Deep Learning oder evidenzbasierte Unsicherheitsmodellierung<br>Peer-reviewed oder relevante arXiv-Preprints<br>Mit Experimenten oder Anwendungen<br>Publikationen 2016–2025<br>Englischsprachig                               |
| Exklusionskriterien | Keine Implementierung oder rein theoretisch ohne EDL-Bezug<br>Nur Bayesian NNs/Ensembles ohne evidenzielle Ansätze<br>Rein technische Papers ohne Unsicherheitsbezug<br>Nicht-englischsprachig<br>Nur Poster oder Abstracts ohne Volltext |

Tabelle 7.16.: A1 Auswahlkriterien für die Literaturübersicht zu EDL

Für die Literaturübersicht wurde ein Fragenkatalog generiert, der mit seinen Schwerpunkten folgende Scopes ergeben hat.

| Scope                            | Beschreibung                                                                                                                               |
|----------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Theoretische Grundlagen       | 1.1. Mathematische Formulierung von EDL<br>1.2. Verbindung zu Subjektiver Logik nach Jøsang<br>1.3. Unterschiede zu Bayesian Deep Learning |
| 2. Architekturen                 | 2.1. Netzarchitekturen für EDL<br>2.2. Modifikationen an klassischen CNNs, RNNs, Transformers<br>2.3. Integration evidenzieller Layer      |
| 3. Loss Funktionen               | 3.1. Evidential Loss vs. Standard Cross Entropy<br>3.2. KL-Regularisierung<br>3.3. Umgang mit fehlerhaften Labels                          |
| 4. Unsicherheitstypisierung      | 4.1. EU vs. AU<br>4.2. Interpretation evidenzieller Parameter                                                                              |
| 5. Benchmarks                    | 5.1. Typische Datensätze (MNIST, CIFAR-10, OOD-Tests)<br>5.2. Vergleich zu Bayesian NNs und Ensembles                                      |
| 6. Anwendungsgebiete             | 6.1. Autonomes Fahren<br>6.2. Industrie 4.0                                                                                                |
| 7. Vergleich zu anderen Methoden | 7.1. BNN<br>7.2. Deep Ensembles<br>7.3. Deterministische Methoden mit Calibration                                                          |
| 8. Herausforderungen             | 8.1. Trainingsstabilität<br>8.2. Overconfidence-Probleme<br>8.3. Skalierbarkeit auf große Modelle                                          |

Tabelle 7.18.: A1 Scopes der Fragen der Literaturübersicht zu EDL

Mit den **A1 Scopes der Fragen der Literaturübersicht zu EDL** wurden folgende **A1 Scopes der Antworten der Literaturübersicht zu EDL** mit zugehörigen Quellen definiert.

| Scope | Beschreibung                                                                                                                                                                       | Quelle       |
|-------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|
| 1.1.  | Sensoy et al. (2018): Evidential Deep Learning to Quantify Classification Uncertainty<br>Sensoy et al. (2023): Evidential Deep Learning: Theory and Practice                       | [17]<br>[32] |
| 1.2.  | Jøsang (2016): Subjective Logic                                                                                                                                                    | [21]         |
| 1.3.  | Kendall & Gal (2017): What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?<br>Peters et al. (2023): On the Mathematical Consistency of Evidential Learning | [18]<br>[33] |
| 2.1.  | Amini et al. (2020): Deep Evidential Regression<br>Tian et al. (2024): Evidential Graph Neural Networks                                                                            | [26]<br>[34] |
| 2.2.  | Sensoy et al. (2020): Uncertainty-Aware Deep Classification<br>Dordevic et al. (2024): Evidential Transformers for Improved Image Retrieval                                        | [35]<br>[36] |
| 2.3.  | Zhou et al. (2022): Evidential Transformer Networks<br>Dordevic et al. (2024): Evidential Transformers for Improved Image Retrieval                                                | [37]<br>[36] |
| 3.1.  | Sensoy et al. (2018): Evidential Loss Function (Evidential Deep Learning)                                                                                                          | [17]         |
| 3.2.  | Sensoy et al. (2020): KL-Regularization in Evidential Networks<br>Amini et al. (2023): Improved KL Regularization for Evidential Regression                                        | [35]<br>[38] |
| 3.3.  | Dhamija et al. (2018): Reducing Model Confidence for Out-of-Distribution Detection                                                                                                 | [39]         |

Fortsetzung auf nächster Seite

| Scope | Beschreibung                                                                            | Quelle |
|-------|-----------------------------------------------------------------------------------------|--------|
| 4.1.  | Kendall & Gal (2017): Aleatoric and Epistemic Uncertainty                               | [18]   |
|       | Tran et al. (2023): Separating Aleatoric and Epistemic Uncertainty in Evidential Models | [40]   |
| 4.2.  | Amini et al. (2020): Interpreting evidential parameters in regression tasks             | [26]   |
| 5.1.  | Amini et al. (2020): Evaluation on MNIST, CIFAR-10, OOD                                 | [26]   |
|       | Bohlke et al. (2023): Evaluation on CIFAR-100C, ImageNet-O                              | [36]   |
| 5.2.  | Lakshminarayanan et al. (2017): Deep Ensembles as Baseline for Uncertainty              | [9]    |
|       | Tran et al. (2023): Comparative Benchmarks                                              | [40]   |
| 6.1.  | Feng et al. (2022): Evidential Uncertainty in Autonomous Driving                        | [41]   |
| 6.2.  | Wang et al. (2022): Uncertainty Modeling in Industrial AI                               | [42]   |
|       | Gupta et al. (2023): Evidential Learning for Industrial Predictive Maintenance          | [43]   |
|       | Chen et al. (2024): Uncertainty Estimation in Vision-Language Models                    | [44]   |
| 7.1.  | Gal & Ghahramani (2016): Dropout as Bayesian Approximation                              | [45]   |
| 7.2.  | Lakshminarayanan et al. (2017): Deep Ensembles                                          | [9]    |
| 7.3.  | Guo et al. (2017): On Calibration of Modern Neural Networks                             | [46]   |
|       | Tran et al. (2023): Comparison of calibration in evidential vs. deterministic methods   | [40]   |
| 8.1.  | Sensoy et al. (2020): Training stability in evidential networks                         | [35]   |
|       | Peters et al. (2023): Loss function degeneracies                                        | [33]   |
| 8.2.  | Dhamija et al. (2018): Overconfidence in neural networks                                | [39]   |
|       | Chen et al. (2024): Overconfidence in multimodal evidential learning                    | [44]   |
| 8.3.  | Zhou et al. (2022): Scalability of evidential transformers                              | [37]   |
|       | Bohlke et al. (2023): Scalability challenges in large Vision Transformers               | [36]   |

Tabelle 7.19.: A1 Scopes der Antworten der Literaturübersicht zu EDL mit aktuellen Quellen (2016–2025)

Hochwertige, zentrale Umfragen sind Ulmer et al. [7], Gawlikowski et al. [47]. Der Zusammenhang der modellbasierten, evidenzbasierten Unsicherheit zu jener der Realdaten ist im Anhang.

**A2** Identifizierung und Analyse geeigneter Unsicherheitsmetriken für Surrogatmodelle, wobei zwischen AU und EU unterschieden wird.

|           | Datenbasierte Metriken                                                                                                                                                                                                                                                                   | Modellbasierte Metriken                                                                                                                                                                                                            |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>AU</b> | <p>Standardabweichung des Rauschens:</p> $\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$ <p>Quelle: Bishop et al. [16, S. 28–30]</p> <p>Varianz der Residuen:</p> $\text{Var} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$ <p>Quelle: Bishop et al. [16, S. 28–30]</p> | <p>Bayesianische Unsicherheit (Posterior):</p> $p(\theta \mid \mathcal{D})$ <p>Quelle: Rasmussen et al. [8, Kap. 2]</p> <p>Kreuzvalidierung:</p> <p>Bewertung auf Trainings- und Testdaten</p> <p>Quelle: Blundell et al. [15]</p> |
| <b>EU</b> | <p>Fehleranalyse auf Testdaten:</p> $\text{Fehler} = \sum_{i=1}^n  y_i - \hat{y}_i $ <p>Quelle: Bishop et al. [16, S. 30–32]</p>                                                                                                                                                         | <p>Konfidenzintervall:</p> $\hat{y} \pm z \cdot \sigma$ <p>Quelle: Gal et al. [13, S. 40–42]</p> <p>Entropie der Posterior-Verteilung:</p> $H(p) = - \sum_i p(x_i) \log(p(x_i))$ <p>Quelle: Sensoy et al. [17]</p>                 |

Tabelle 7.21.: A2 Unsicherheitsmetriken für Surrogatmodelle

**A3** Vergleich *modellspezifischer* und *anwendungsspezifischer* Bewertungsmetriken für die Quantifizierung von Unsicherheit.

Für den Vergleich werden Anwendungen im Ingenieurwesen vorgeschlagen:

| Domäne               | Typische Anwendungen                       |
|----------------------|--------------------------------------------|
| Strukturmechanik     | Spannungsanalyse, Lebensdauerberechnung    |
| Aerodynamik          | CFD-Simulation, Drag/Lift-Koeffizienten    |
| Fahrzeugbau          | Crashsimulation, Akustik, Thermomanagement |
| Akustik              | Modalanalyse, Schallausbreitung            |
| Materialwissenschaft | Materialeigenschaften, Werkstoffdesign     |
| Geophysik            | Erdbebensimulation, Bodenmechanik          |
| Prozesssimulation    | Chemische Reaktoren, Prozessoptimierung    |
| Energie              | Batterie-Management, Netzstabilität        |

Tabelle 7.23.: Anwendungsdomänen im Ingenieurwesen für ML-Surrogatmodelle nach Gawlikowski, Ulmer

Als Beispiel-Domäne zum Vergleich der *modellspezifischen* und *anwendungsspezifischen* Unsicherheitsmetriken für Surrogatmodelle im Autonomen Fahren wird der Fahrzeugbau gewählt. Die genannten Anwendungsdomänen sind zusammengetragen aus Gawlikowski et al. [47] sowie Ulmer et al. [7]. Beide Werke behandeln typische Einsatzfelder von ML-basierten Surrogatmodellen im Ingenieurwesen.

|           | Modellspezifische Metriken                                                                                                                                                                                                                                                                                                                                                                                                                                             | Anwendungsspezifische Metriken                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>AU</b> | <p>RMSE : Abweichung zwischen vorhergesagten und tatsächlichen Sensordaten (z. B. Positionen von Objekten)<br/> <i>Quelle:</i> Bishop et al. [16, S. 224–226]</p> <p>MAE : Mittlere absolute Abweichung bei Objekterkennung oder Trajektorien<br/> <i>Quelle:</i> Bishop et al. [16, S. 224–226]</p> <p>KL (<math>P  Q</math>) : Abweichung zwischen Vorhersage- und Ground-Truth-Wahrscheinlichkeitsverteilungen<br/> <i>Quelle:</i> Rasmussen et al. [8, Kap. 2]</p> | <p>Accuracy : Korrektheit der Objekterkennung (z. B. Fußgänger erkannt/nicht erkannt)<br/> <i>Quelle:</i> Bishop et al. [16, S. 40–42]</p> <p>F1-Score : Balance zwischen Präzision und Recall bei der Objekterkennung<br/> <i>Quelle:</i> Bishop et al. [16, S. 40–42]</p> <p>AUC : Trennschärfe bei der Klassifikation sicherer vs. unsicherer Szenarien<br/> <i>Quelle:</i> Bishop et al. [16, S. 40–42]</p>                                      |
| <b>EU</b> | <p><math>\mathcal{L}(\theta)</math> : Loss-Funktion inkl. Unsicherheitsanteile (z. B. evidentieller Loss)<br/> <i>Quelle:</i> Sensoy et al. [17]</p> <p>BMA : Mittelung mehrerer Modellposteriors für robustere Unsicherheiten<br/> <i>Quelle:</i> Blundell et al. [15]</p> <p>CI : Konfidenzintervalle für kritische Größen (z. B. Zeit bis Kollision)<br/> <i>Quelle:</i> Gal et al. [13, S. 40–42]</p>                                                              | <p>Fehler <math>\epsilon</math> : Abweichung von vorhergesagten zu tatsächlichen Sicherheitsabständen<br/> <i>Quelle:</i> Bishop et al. [16, S. 233–235]</p> <p><math>H(p)</math> : Entropie der Klassifikationsausgabe als Maß epistemischer Unsicherheit<br/> <i>Quelle:</i> Ovadia et al. [10]</p> <p>AUC-PR : Leistungsfähigkeit bei seltenen, kritischen Klassen (z. B. seltene Gefahrensituationen)<br/> <i>Quelle:</i> Ovadia et al. [10]</p> |

Tabelle 7.25.: Vergleich modellspezifischer und anwendungsspezifischer Unsicherheitsmetriken für Surrogatmodelle im autonomen Fahren

**A4** Untersuchung und Vergleich verschiedener UQ-Ansätze für ML-basierte Surrogatmodelle, insbesondere BNN und CP.

Zur Untersuchung von A4 wird der Vergleich zwischen BNNs und CP auf Grundlage identischer Modellarchitekturen und Datensätze durchgeführt. Für BNNs werden die Implementierungen von Herrn Mavani in NumPyro/JAX genutzt (HMC und SVI) [48]. Für CP wird ein separates Skript erstellt, das auf denselben Netzen basiert, jedoch CI mittels Konformitäts-Quantilen auf einem Calibration-Set berechnet.

| Aspekt                     | Details                                                                                                                                                                             |
|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Vergleichskriterien</b> | RMSE, MAE der Punktvorhersagen<br>ECE<br>Coverage-Rate der CI<br>Durchschnittliche Breite der CI<br>Rechenzeit für Training und Vorhersage                                          |
| <b>Ziel der Analyse</b>    | Unterschiede der Verfahren hinsichtlich:<br>Trennung EU und AU (nur bei BNNs)<br>Strikten Garantien für Unsicherheitsintervalle (nur bei CP)<br>Ressourcenbedarf und Skalierbarkeit |

Tabelle 7.27.: A4 Vergleichskriterien und Analyseziele für EDL zu anderen Methoden

**A5** Untersuchung des theoretischen Einflusses verschiedener Faktoren auf die UQ ausgewählter Modelle.

**Vergleichs-Score** Der Vergleichs-Score wird berechnet als:

$$\begin{aligned}
 \text{Score} &= w_{\text{err}} \cdot \frac{\text{RMSE}}{\text{Norm}_{\text{RMSE}}} + w_{\text{ECE}} \cdot \frac{\text{ECE}}{\text{Norm}_{\text{ECE}}} + w_{\text{width}} \cdot \frac{\text{IntervalWidth}}{\text{Norm}_{\text{Width}}} + w_{\text{runtime}} \cdot \frac{\text{Runtime}}{\text{Norm}_{\text{Runtime}}} \\
 &= 0,5 \cdot \frac{\text{RMSE}}{10,0} + 0,3 \cdot \frac{\text{ECE}}{1,0} + 0,1 \cdot \frac{\text{IntervalWidth}}{5,0} + 0,1 \cdot \frac{\text{Runtime}}{30,0}
 \end{aligned}$$

**Begründung der Norm-Parametrisierung für Vergleichs-Score** Empirisch ergab sich aus der Ausführung des Vergleichsskripts: Erstens, die RMSE schwankt zwischen 0,3 und 3,0, daher wird  $\text{Norm}_{\text{RMSE}} = 10,0$  gesetzt, um ausreichend Spielraum zu lassen. Zweitens, die ECE ist eine Wahrscheinlichkeit im Intervall  $[0, 1]$ , sodass sich  $\text{Norm}_{\text{ECE}} = 1,0$  direkt ergibt. Drittens, die Intervallbreite liegt auf Hardwarekonfiguration 1 zwischen 2,0 und 4,5, weshalb  $\text{Norm}_{\text{Width}} = 5,0$  als konservative Obergrenze gewählt wird. Viertens, die Laufzeit reicht von etwa 2 s (CP) bis 25 s (HMC), daher wird  $\text{Norm}_{\text{Runtime}} = 30,0$  gewählt, um die gesamte Spanne abzudecken.

|    | BNN                                                                                                                                                                                                                                                                                                                | CP                                                                                                                                                                                                                                                                                               |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| AU | <p>Posterior-Verteilung <math>p(\theta \mid \mathcal{D})</math><br/> <i>Quelle:</i> Blundell et al. [15, Kap. 3]</p> <p>Unsicherheitsquantifizierung über Modellparameter auf Trainingsdaten<br/> <i>Quelle:</i> Gal et al. [13, S. 40–42]</p> <p>MC Dropout<br/> <i>Quelle:</i> Gal et al. [13, S. 40–42]</p>     | <p>Konfidenzintervall <math>\hat{y} \pm z \cdot \sigma</math><br/> <i>Quelle:</i> Vovk et al. [49, S. 4–5]</p> <p>Unsicherheitsquantifizierung durch Konfidenzintervalle<br/> <i>Quelle:</i> Angelopoulos et al. [50]</p> <p>Ensemble-Methoden<br/> <i>Quelle:</i> Angelopoulos et al. [50]</p>  |
| EU | <p>Bayesianische Unsicherheit <math>p(\theta \mid \mathcal{D})</math><br/> <i>Quelle:</i> Gal et al. [13, S. 40–42]</p> <p>BI: Unsicherheitsquantifizierung über bay. Modellparameter<br/> <i>Quelle:</i> Mac Kay et al. [51]</p> <p>VI: Posterior-Approximation<br/> <i>Quelle:</i> Gal et al. [13, S. 40–42]</p> | <p>Vertrauensregionen<br/> <i>Quelle:</i> Angelopoulos et al. [50]</p> <p>Prädiktive Unsicherheitsquantifizierung durch Konfidenzintervalle<br/> <i>Quelle:</i> Shafer et al. [52, S. 63–65]</p> <p>GPR: UQ-Ansatz mit Kernel-Methoden und Training<br/> <i>Quelle:</i> Rasmussen et al. [8]</p> |

Tabelle 7.29.: A5 Vergleich der UQ-Ansätze für BNN und CP

**Begründung der Gewichtungsfaktoren für Vergleichs-Score** Erstens, das Fehlermaß RMSE geht mit  $w_{\text{err}} = 0,5$  am stärksten in die Bewertung ein, da die Vorhersagegenauigkeit im Zentrum der Modellgüte steht. Zweitens, die Kalibrierung wird durch  $w_{\text{ECE}} = 0,3$  gewichtet, da eine realistische Unsicherheitsquantifizierung für verlässliche Modelle essenziell ist. Drittens, die mittlere Intervallbreite wird mit  $w_{\text{width}} = 0,1$  berücksichtigt, da sie als sekundäres Kriterium eher die Informationsdichte als die primäre Modellqualität beeinflusst. Viertens, die Laufzeit geht mit  $w_{\text{runtime}} = 0,1$  ein, um recheneffiziente Modelle bevorzugt zu behandeln, ohne Genauigkeit und Kalibrierung zu überlagern.

| Methode | RMSE     | MAE      | Coverage | ECE   | MCIW   | Score   |
|---------|----------|----------|----------|-------|--------|---------|
| BNN HMC | 6.845    | 5.697    | 0.010    | 0.990 | 2.404  | 1.3988  |
| BNN SVI | 1367.638 | 1046.232 | 0.020    | 0.980 | 70.001 | 69.8022 |
| CP      | 0.096    | 0.075    | 0.870    | 0.130 | 0.267  | 0.0493  |

Tabelle 7.31.: Vergleich der Modelle BNN (HMC, SVI) und CP hinsichtlich Performanz und Unsicherheitsquantifizierung.

### Interpretation der Ergebnisse

1. **BNN HMC**: Liefert moderaten Fehler ( $\text{RMSE} \approx 6,8$ ), geringe EU, aber extrem geringe Coverage. Die Unsicherheitsintervalle decken die wahren Werte kaum ab, was ein Overconfidence-Problem zeigt.
2. **BNN SVI**: Scheitert deutlich. Sehr hoher RMSE ( $\approx 1367$ ) und große epistemische Varianz deuten auf Instabilität oder falsche Konvergenz hin.
3. **CP**: Erzielt sehr niedrige Fehler ( $\text{RMSE} \approx 0,096$ ), gute Coverage (ca. 87 %) und den besten Score. Die Methode empfiehlt sich auf diesem Datensatz.

**A6** Untersuchung des theoretischen Einflusses verschiedener Faktoren auf die Unsicherheits-schätzungen ausgewählter Modelle.

A6 soll mit minimaler Sensitivitätsanalyse am Beispiel ENN ausgeführt werden, um den theoretischen Modellparameter-Einfluss auf die UQ zu quantifizieren. Variiert werden die Evidenzparameter  $\nu$  sowie Trainingsdatengröße  $N_{train}$  für geringen Analyseaufwand.

|          | BNN                                                                                                                                                                                                                                   | CP                                                                                                                                                                                                        | ENN                                                                                           |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|
| <b>A</b> | Post. $p(\theta \mid \mathcal{D})$<br><i>Quelle:</i> Blundell et al. [15, Kap. 3]<br>UQ ü. Modellparameter<br><i>Quelle:</i> Gal et al. [13, S. 40–42]<br>MC Dropout<br><i>Quelle:</i> Gal et al. [13, S. 40–42]                      | CI $\hat{y} \pm z \cdot \sigma$<br><i>Quelle:</i> Vovk et al. [49, S. 4–5]<br>UQ ü. Konfidenzintervalle<br><i>Quelle:</i> Angelopoulos et al. [50]<br>Ensemble<br><i>Quelle:</i> Angelopoulos et al. [50] | Prediktive Vert. $p(y \mid \mu, \alpha, \beta, \nu)$<br>Evidenzparameter $\nu, \alpha, \beta$ |
| <b>B</b> | Modellparameter: Gewichte $w_i$<br><i>Quelle:</i> Mac Kay et al. [51, S. 448–450]<br>Einfluss von Priors<br><i>Quelle:</i> Rasmussen et al. [8, Kap. 5]<br>Hyperparameter-Sensitivität<br><i>Quelle:</i> Rasmussen et al. [8, Kap. 5] | UQ über CI<br><i>Quelle:</i> Shafer et al. [52, S. 63–65]<br>Einfluss von CR<br><i>Quelle:</i> Vovk et al. [49]                                                                                           | Evidenzparameter $\nu, \alpha, \beta$<br>steuern Unsicherheitsanteile.                        |
| <b>C</b> | Bayessche Inferenz zur UQ<br><i>Quelle:</i> Gal et al. [13, S. 41–42]<br>MC-Simulation mit Dropout<br><i>Quelle:</i> Blundell et al. [15]                                                                                             | Ensemble zur UQ<br><i>Quelle:</i> Angelopoulos et al. [50]<br>CR, CI-Analyse<br><i>Quelle:</i> Shafer et al. [52]                                                                                         | Keine Sampling-Approx. nötig<br>KL-Regularisierung steuert $E$                                |

Tabelle 7.33.: A6 Übersicht der UQ-Methoden BNN, CP und ENN

### Parametrisierung der Sensitivitätsanalyse ENN

$$\nu \in \{1, 5, 10, 20\}$$

Architektur: Dense(64)  $\rightarrow$  Dense(64)  $\rightarrow$  Output:  $(\gamma, \log \nu, \log \alpha, \log \beta)$

$$\text{Varianzen: } \begin{cases} \text{Aleatorisch: } \sigma_{\text{alea}}^2 = \frac{\beta}{\alpha-1} \\ \text{Epistemisch: } \sigma_{\text{epis}}^2 = \frac{\beta}{\nu(\alpha-1)} \end{cases}$$

| $\nu$ | $\bar{\sigma}_{\text{total}}^2$ | $\bar{\sigma}_{\text{aleatoric}}^2$ | $\bar{\sigma}_{\text{epistemic}}^2$ | RMSE     | RUI    |
|-------|---------------------------------|-------------------------------------|-------------------------------------|----------|--------|
| 1     | $1.6191 \pm 0.1344$             | $0.8095 \pm 0.0672$                 | $0.8095 \pm 0.0672$                 | 155.7554 | 0.0082 |
| 5     | $1.0007 \pm 0.1152$             | $0.8339 \pm 0.0960$                 | $0.1668 \pm 0.0192$                 | 155.8031 | 0.0064 |
| 10    | $0.8591 \pm 0.0482$             | $0.7810 \pm 0.0439$                 | $0.0781 \pm 0.0044$                 | 155.0450 | 0.0060 |
| 20    | $0.8376 \pm 0.0804$             | $0.7978 \pm 0.0766$                 | $0.0399 \pm 0.0038$                 | 156.6731 | 0.0058 |

Tabelle 7.35.: A6 Min. Sensitivitätsanalyse ENN mit  $\pm 2\sigma$

Der Parameter  $\nu$  steuert im ENN die Gewichtung zwischen EU und AU, wobei kleine  $\nu$ -Werte die EU dominieren lassen und große  $\nu$ -Werte die Unsicherheitszerlegung stärker zugunsten der AU Komponente verschieben.

### Parametrisierung der Sensitivitätsanalyse BNN

$$\text{Dropout-Raten: } \delta \in \{0.05, 0.10, 0.20, 0.50\}$$

Architektur: Dense(64)  $\rightarrow$  Dropout( $\delta$ )  $\rightarrow$  Dense(64)  $\rightarrow$  Dropout( $\delta$ )  $\rightarrow$  Output:  $\mu, \log \sigma^2$

$$\text{Loss: } \mathcal{L}_{\text{BNN}} = \frac{1}{2} \log \sigma^2 + \frac{(y - \mu)^2}{2\sigma^2}$$

Die Dropout-Rate im BNN beeinflusst direkt die EU, da höhere Dropout-Raten mehr Modellvariabilität zulassen und somit größere EU in den Vorhersagen erzeugen.

### Parametrisierung der Sensitivitätsanalyse CP

$$\alpha \in \{0.20, 0.10, 0.05, 0.01\}$$

| $p_{\text{dropout}}$ | $\bar{\sigma}_{\text{total}}^2$ | $\bar{\sigma}_{\text{aleatoric}}^2$ | $\bar{\sigma}_{\text{epistemic}}^2$ | RMSE   | RUI    |
|----------------------|---------------------------------|-------------------------------------|-------------------------------------|--------|--------|
| 0.05                 | $0.0079 \pm 0.0162$             | $0.0051 \pm 0.0108$                 | $0.0028 \pm 0.0059$                 | 6.8592 | 0.0129 |
| 0.10                 | $0.0164 \pm 0.0382$             | $0.0099 \pm 0.0219$                 | $0.0065 \pm 0.0167$                 | 6.5767 | 0.0195 |
| 0.20                 | $0.0347 \pm 0.0859$             | $0.0198 \pm 0.0460$                 | $0.0150 \pm 0.0409$                 | 7.2445 | 0.0257 |
| 0.50                 | $0.1482 \pm 0.3429$             | $0.0783 \pm 0.1419$                 | $0.0699 \pm 0.2080$                 | 8.9771 | 0.0429 |

Tabelle 7.37.: A6 Min. Sensitivitätsanalyse BNN mit  $\pm 2\sigma$

Modell: Random Forest Regressor (n=100)

Kalibrierung: Residual-basiertes Intervall:  $[\hat{y} - q_{1-\alpha}, \hat{y} + q_{1-\alpha}]$

| $\alpha$ | Coverage | Mean CI-Width | $\bar{\sigma}_{\text{aleatoric}}^2$ | $\bar{\sigma}_{\text{epistemic}}^2$ | $\bar{\sigma}_{\text{total}}^2$ | RUI    |
|----------|----------|---------------|-------------------------------------|-------------------------------------|---------------------------------|--------|
| 0.20     | 0.820    | 1.0022        | $0.2088 \pm 0.9139$                 | $0.3612 \pm 0.4821$                 | $0.5700 \pm 0.4821$             | 1.8978 |
| 0.10     | 0.913    | 1.3215        | $0.2088 \pm 0.9139$                 | $0.3612 \pm 0.4821$                 | $0.5700 \pm 0.4821$             | 1.8978 |
| 0.05     | 0.953    | 1.7910        | $0.2088 \pm 0.9139$                 | $0.3612 \pm 0.4821$                 | $0.5700 \pm 0.4821$             | 1.8978 |
| 0.01     | 1.000    | 3.0993        | $0.2088 \pm 0.9139$                 | $0.3612 \pm 0.4821$                 | $0.5700 \pm 0.4821$             | 1.8978 |

Tabelle 7.39.: A6 Min. Sensitivitätsanalyse CP mit  $\pm 2\sigma$

Das Signifikanzniveau  $\alpha$  bestimmt bei CP die Breite der Vorhersageintervalle, wobei kleinere  $\alpha$ -Werte zu höheren Konfidenzniveaus und damit breiteren, konservativeren Unsicherheitsintervallen führen.

**Vergleichende Auswertung Modelle BNN, CP, ENN** Die drei Tabellen zeigen die Sensitivitätsanalysen für ENN, BNN und CP unter Variation ihrer jeweiligen Steuerungsgrößen ( $\nu$ , Dropout-Rate,  $\alpha$ ). Als gemeinsame Kennzahlen wurden die mittlere Gesamtvarianz  $\overline{\sigma}(y)^2$ , die mittlere aleatorische Varianz  $\bar{\sigma}_{\text{AU}}^2$ , die mittlere epistemische Varianz  $\bar{\sigma}_{\text{EU}}^2$ , daneben der RMSE sowie der RUI ausgewertet.

Im direkten Vergleich zeigen ENN insgesamt höhere Varianzen, zum Beispiel bis  $\overline{\sigma}(y)^2 \approx 1.6$ , während der RMSE relativ stabil bei etwa 155 liegt. Die Zerlegung der Unsicherheit zeigt bei ENN für kleine  $\nu$  etwa gleiche Anteile zwischen  $\bar{\sigma}_{\text{AU}}^2$  und  $\bar{\sigma}_{\text{EU}}^2$ . Mit steigender Evidenz  $\nu$  sinkt  $\bar{\sigma}_{\text{EU}}^2$  deutlich, wodurch sich der Anteil der Unsicherheit zunehmend auf die aleatorische Komponente verlagert. Dies verdeutlicht, dass zu geringe  $\nu$ -Werte eine *Overconfidence* verhindern können, da das Modell dann größere epistemische Unsicherheit zulässt.

Die BNN liefern deutlich kleinere Varianzen, teilweise unter  $\overline{\sigma(y)}^2 < 0.02$ , bei gleichzeitig wesentlich niedrigerem RMSE zwischen 6 und 9. Höhere Dropout-Raten führen jedoch zu steigenden Werten sowohl für  $\overline{\sigma}_{\text{EU}}^2$  als auch für  $\overline{\sigma}_{\text{AU}}^2$ , was auf zunehmende Unsicherheitsstreuung hinweist. Allerdings kann ein zu niedriges Dropout-Raten-Niveau bei BNNs *Overconfidence* fördern, da epistemische Unsicherheit dann unterrepräsentiert bleibt.

CP liefert über alle  $\alpha$ -Level hinweg konstante mittlere Varianzwerte, da der zugrundeliegende Random-Forest-Fit unverändert bleibt. Hier beträgt  $\overline{\sigma(y)}^2 \approx 0.57$ , zusammengesetzt aus einer moderaten epistemischen Varianz  $\overline{\sigma}_{\text{EU}}^2 \approx 0.36$  und einer geringeren aleatorischen Komponente  $\overline{\sigma}_{\text{AU}}^2 \approx 0.21$ . Die RUI-Werte liegen bei CP mit ca. 1.9 deutlich höher, was eine relativ große Unsicherheitsbreite im Verhältnis zum RMSE (hier  $< 0.4$ ) reflektiert. Durch die direkte Intervallberechnung schützt CP per Konstruktion stärker gegen *Overconfidence*, bleibt jedoch konservativer.

Insgesamt zeigen die Modelle unterschiedliche Profile: Erstens, ENN weisen die höchsten Unsicherheitsniveaus bei konstant hohem RMSE auf. Zweitens, BNN erzielen geringere Fehler und niedrigere Unsicherheiten, doch reagieren sensibel auf die gewählte Dropout-Rate. Drittens, CP liefert stabile, eher breite Intervalle und sichert robuste Kalibrierung. Zusammenfassend erlauben die Kennzahlen  $\overline{\sigma(y)}^2$ ,  $\overline{\sigma}_{\text{AU}}^2$ ,  $\overline{\sigma}_{\text{EU}}^2$ , RMSE und RUI eine methodenübergreifende Bewertung der Unsicherheiten und helfen wie grundlegend aufgezeigt *Overconfidence* frühzeitig zu erkennen.

| Methode    | Sensitivitätsaspekte                                                                                                                                                                                                                                         |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>BNN</b> | Breitere Posterior bei schwachen Priors; stärkere EU.<br>Höhere Dropout-Rate → erhöht EU.                                                                                                                                                                    |
| <b>CP</b>  | CP-Breite steigt bei höherem Konfidenzniveau.<br>Kleinere Calibration-Sets führen zu unsteten CPs.<br>Coverage sinkt bei zu engen Regionen; Intervallbreite steigt bei höherem $\alpha$ .<br>Heteroskedastische Daten erzeugen stark variierende CP-Breiten. |
| <b>ENN</b> | Hohe EU bei kleinen $\nu$ -Werten.<br>Niedrige $\nu \rightarrow$ mehr epistemische Varianz.<br>Größere Datenmengen reduzieren Unsicherheit.<br>Zu starke KL-Regularisierung reduziert Modellvertrauen übermäßig.                                             |

Tabelle 7.41.: A6 Zusammenfassung Sensitivität UQ-Methoden BNN, CP, ENN

## 8. Durchführung

In der Durchführung soll *A6 Implementierung und Validierung der entwickelten Unsicherheitsmetriken und Modellierungstechniken in einer realistischen Anwendung*, wie z. B. der *Unsicherheitsabschätzung in kritischen Regionen von Simulationsdaten* beantwortet werden. Amini [26] liefert einen bekannten validierten Ansatz aus öffentlicher Forschung. Ein Projektteilnehmer im folgenden gekürzt als DH aus VaMAI liefert einen bekannten validierten Ansatz von Unternehmensseite.

### Bekannter Ansatz zur Raumerfassung

#### VaMAI - Kritikalitätsraum - Validierung - Annahmen

VaMAI verfolgt mit dem Kritikalitätsraum die Idee einer Kritikalitätsabdeckung unter anderem über die Arbeit von DH. DH lieferte dafür bereits einen Ansatz mittels „Gaussian Process“ als Surrogatmodell für die szenarienbasierte Validierung der autonomen Systeme im Unternehmen. Ein Alternativer Ansatz zur Unsicherheitsanalyse dabei sind BNN, ENN. Umgangssprachlich hat DH den Raum so gut abgedeckt, dass mit einer gewissen Wahrscheinlichkeit kein Kritikalitätsbereich mehr in diesem Raum vorhanden ist. Als Beweisidee äußert sich das wie folgt: Zeige, dass durch die gewählte Methode - zum Beispiel Gaussian Processes und Unsicherheitsaggregation - die Wahrscheinlichkeit dafür, dass ein unentdeckter Kritikalitätsraum existiert, unter einem Schwellenwert liegt.

$$\mathbb{P}(\exists x \in \Omega \setminus \widehat{\Omega}_k : \text{Kritikalität}(x) > \tau) < \varepsilon.$$

Das heißt, die Wahrscheinlichkeit, dass irgendwo im Raum  $\Omega$  noch ein Punkt existiert, dessen Kritikalität den Schwellenwert überschreitet, aber nicht durch diese Methode erkannt wurde, ist hinreichend klein zum Beispiel unter 5 % mit  $\mu \pm 2\sigma$ .  $\Omega \subset \mathbb{R}^n$  ist der gesamte Zustands- oder Szenarienraum.  $\Omega_k := \{x \in \Omega \mid k(x) \geq \tau\}$  ist der echte Kritikalitätsraum.  $\widehat{\Omega}_k := \{x \in \Omega \mid k(x) \geq \tau\}$  ist der durch die Methode erkannte Kritikalitätsraum.  $\varepsilon \in (0, 1)$  ist das akzeptierte Fehlerniveau. Zu beweisen ist:  $\mathbb{P}(\Omega_k \setminus \widehat{\Omega}_k \neq \emptyset) \leq \varepsilon$ . Dazu wird folgende Vorgehensweise in Betracht gezogen:

1. Unsicherheitsmodellierung mittels machine-learning basiertem Surrogatmodell

- Liefert für jeden Punkt  $x \in \Omega$  eine Wahrscheinlichkeitsverteilung mit Varianz  $\text{Var}(x) = \sigma_{\text{epistemisch}}^2 + \sigma_{\text{aleatorisch}}^2$ . Dieser Aspekt wird in der Arbeit von Herr N. Mavani für BNN nach meiner Projektteilnahme mittels dieser Thesis, auf meiner Seite mittels ENN untersucht.
- Kritikalität ist allgemein im Projekt Funktion der Größe  $k(x)$ .

## 2. Sampling- / Klassifikationsabdeckung

- Raum  $\Omega$  wurde mit Dichte  $\delta$  gesampled - zum Beispiel mittels Gitter, Trajektorien, Pfaden - von VaMAI Projektteilnehmer Herr Imholze mit dem Ziel einer möglichst hohen Abdeckung.
- Wenn  $\forall x \in \Omega$  ein Sample  $x' \in \hat{\Omega}_k$  mit  $\|x - x'\| \leq \delta'$  existiert, gilt „dichte Abdeckung“.
- Generell ist in Hinblick auf in VaMAI abgezielte Methoden wie „Few Shot Learning“ von Herr Mavani mit Sampleumfängen  $n \in [20, 30]$ ,  $n \in \mathbb{N}$  zu beachten, dass der zentrale Grenzwertsatz der auch die EDL später erläuterte Student-t-Verteilung betrifft, nach Faustformel ab einem Stichprobenumfang  $n \geq 30$  gültig ist.

## 3. Uniforme Konvergenz / PAC-ähnliche Argumentation

- Zeige, dass mit Vorhersagewahrscheinlichkeit, Unsicherheitsabschätzung die Methodik mit zunehmender Samplezahl  $N$  gegen den wahren Kritikalitätsraum konvergiert.

$$\mathbb{P}\left(\sup_{x \in \Omega} |f(x) - \hat{k}(x)| \geq \eta\right) \leq \varepsilon.$$

- Wenn  $\widehat{k(x)} \approx k(x)$  auf ganz  $\Omega$ , gilt auch  $\hat{\Omega}_k \approx \Omega_k$ . Es folgt, dass die Wahrscheinlichkeit einen echten Kritikalitätsraum *nicht* zu erkennen kleiner als  $\epsilon$  ist. Dieser Aspekt ist von Herr DH im Rahmen seiner Masterarbeit mittels Gaussian Processes gezeigt worden.

**Szenarienselektion** Im Rahmen dieser Beweisidee ist die Szenarioselektion in VaMAI einzuordnen. Die Szenarienselektion dient der gezielten Auswahl von Szenarien auf Grundlage der Unsicherheiten, die durch das Surrogatmodell bereitgestellt werden. Diese Unsicherheiten setzen sich aus einem epistemischen und einem aleatorischen Anteil zusammen. Zunächst wird ein initiales Szenariosample  $\hat{\Omega}_0$  definiert, auf dem ein erstes Surrogatmodell trainiert wird. Anschließend erfolgt eine Bewertung der Modellunsicherheit über den gesamten Raum  $\Omega$ , wobei insbesondere Regionen mit hoher epistemischer Unsicherheit identifiziert werden. Im dritten Schritt wird ein adaptives Sampling durchgeführt: Neue Szenarien mit hoher epistemischer Unsicherheit sowie ausgeprägter Kritikalität  $k(x)$  werden gezielt zur Ergänzung des bisherigen

Szenariosamples ausgewählt. Daraufhin wird das Surrogatmodell mit den hinzugefügten Szenarien erneut trainiert und validiert. Im Anschluss daran wird überprüft, ob ein festgelegtes Abdeckungs- oder Konvergenzkriterium - beispielsweise hinreichende Testgenauigkeit, Unsicherheitsreduktion oder Konvergenz gemäß PAC-Bedingung - erfüllt ist. Ist dies nicht der Fall, beginnt der Selektionsprozess erneut mit der Analyse der Modellunsicherheit. Ziel der Methode ist es, mit möglichst wenigen, aber gezielt gewählten Szenarien eine aussagekräftige und belastbare Approximation des Kritikalitätsraums innerhalb von  $\Omega$  zu erreichen. Vergleiche dazu die Übersicht im Anhang.

## Bekannter Ansatz - Evidential Deep Learning

Zur Validierung der eingesetzten Methodik für ENN wird der von Amini (vgl. Amini [26]) veröffentlichte Benchmark reproduziert. Hierfür wird die im Anhang dargestellte Aufwandschätzung berücksichtigt, die auf die im Unternehmen verfügbare Hardware abgestimmt ist, siehe Hardwarekonfiguration 1 im Anhang.

**Vergleich Amini ENN Regression** Amini Sektion 4 (vgl. Amini [26]) Experiment liefert Ergebnisse für ENN.

$\{\text{RMSE, NLL}\} \times \{\text{dropout, ensembles, evidential}\} \times \{\text{boston, concrete, energy, kin8nm, naval, power, protein, wine, yacht}\}$

Der Benchmark konnte in erster Konfiguration annähernd reproduziert werden.

```
python3 neurips2020/run_uci_dataset_tests.py -datasets boston concrete
energy-efficiency kin8nm naval power-plant protein wine yacht
-num-trials 1 -num-epochs 5
```

mit Laufzeit von 9 Minuten 10.2 Sekunden  $\leq 10$  Minuten als „Proof of concept“. Die Konvergenz zu Aminis Werten ( $\Delta(\text{RMSE}) = [-0.5; -0.6]$ ;  $\Delta\text{NLL} = -0.08$ ) wird deutlich für die Konfiguration hinreichend umfangreicher Datensätze  $\{\text{kin8nm, naval, protein}\}$  unter der Hardware-Limitierung aus Tabelle 1, Gerät 1, Hardware-Spezifikation, Anhang.

**Vergleich Amini ENN Klassifikation** Aminis Reproduzierbarkeits-Sektion (vgl. Amini [26]) als Skript `gen_depth_results.py` ausgeführt, liefert Ergebnisse für ENN für die Aufgabe „Monocular depth estimation“. Der Benchmark konnte reproduziert werden mit einer Konfiguration  $N_{\text{batch}} = 25$ ,  $N_{\text{adv}} = 3$  unter der Hardware-Limitierung aus Tabelle 1, Gerät 1, Hardware-Spezifikation, Anhang.

## 8. Durchführung

---

| Datensatz         | RMSE                                |                                     |                                     | NLL                                   |                                       |                                       |
|-------------------|-------------------------------------|-------------------------------------|-------------------------------------|---------------------------------------|---------------------------------------|---------------------------------------|
|                   | Dropout                             | Ensemble                            | Evidential                          | Dropout                               | Ensemble                              | Evidential                            |
| boston            | 3.678<br>2.896<br>$\Delta = +0.782$ | 4.956<br>2.998<br>$\Delta = +1.958$ | 3.672<br>2.886<br>$\Delta = +0.786$ | 4.636<br>2.755<br>$\Delta = +1.881$   | 2.981<br>2.838<br>$\Delta = +0.143$   | 2.621<br>2.701<br>$\Delta = -0.080$   |
| concrete          | 6.905<br>5.367<br>$\Delta = +1.538$ | 6.705<br>5.263<br>$\Delta = +1.442$ | 6.496<br>5.163<br>$\Delta = +1.333$ | 5.254<br>3.213<br>$\Delta = +2.041$   | 3.271<br>3.193<br>$\Delta = +0.078$   | 3.277<br>3.138<br>$\Delta = +0.139$   |
| energy-efficiency | 2.721<br>1.200<br>$\Delta = +1.521$ | 2.516<br>0.987<br>$\Delta = +1.529$ | 2.487<br>0.984<br>$\Delta = +1.503$ | 4.699<br>-0.964<br>$\Delta = +5.663$  | 1.920<br>-1.315<br>$\Delta = +3.235$  | 1.879<br>-1.376<br>$\Delta = +3.255$  |
| kin8nm            | 0.095<br>0.096<br>$\Delta = -0.001$ | 0.084<br>0.080<br>$\Delta = +0.004$ | 0.088<br>0.077<br>$\Delta = +0.011$ | 1.114<br>-1.213<br>$\Delta = +2.327$  | -1.113<br>-1.496<br>$\Delta = +0.383$ | -0.992<br>-1.537<br>$\Delta = +0.545$ |
| naval             | 0.005<br>0.003<br>$\Delta = +0.002$ | 0.005<br>0.002<br>$\Delta = +0.003$ | 0.005<br>0.002<br>$\Delta = +0.003$ | -2.438<br>-6.233<br>$\Delta = +3.795$ | -3.931<br>-6.967<br>$\Delta = +3.036$ | -3.747<br>-6.987<br>$\Delta = +3.240$ |
| power-plant       | 3.349<br>4.049<br>$\Delta = -0.700$ | 3.467<br>4.021<br>$\Delta = -0.554$ | 3.377<br>3.987<br>$\Delta = -0.610$ | 5.281<br>2.795<br>$\Delta = +2.486$   | 2.678<br>2.765<br>$\Delta = -0.087$   | 2.671<br>2.755<br>$\Delta = -0.084$   |
| protein           | 4.355<br>4.512<br>$\Delta = -0.157$ | 3.939<br>4.480<br>$\Delta = -0.541$ | 4.169<br>4.430<br>$\Delta = -0.261$ | 4.266<br>2.871<br>$\Delta = +1.395$   | 2.742<br>2.837<br>$\Delta = -0.095$   | 2.706<br>2.779<br>$\Delta = -0.073$   |
| wine              | 0.826<br>0.648<br>$\Delta = +0.178$ | 0.829<br>0.631<br>$\Delta = +0.198$ | 0.722<br>0.609<br>$\Delta = +0.113$ | 2.255<br>-0.574<br>$\Delta = +2.829$  | 1.390<br>-0.618<br>$\Delta = +2.008$  | 1.187<br>-0.679<br>$\Delta = +1.866$  |
| yacht             | 7.046<br>1.711<br>$\Delta = +5.335$ | 9.197<br>1.337<br>$\Delta = +7.860$ | 5.557<br>1.252<br>$\Delta = +4.305$ | 5.170<br>-1.637<br>$\Delta = +6.807$  | 3.041<br>-2.257<br>$\Delta = +5.298$  | 2.401<br>-2.341<br>$\Delta = +4.742$  |

Tabelle 8.2.: Vergleich reproduzierte Ergebnisse mit Benchmark *Amini et al. (2020)*

## Eigenentwicklung

**Datenpipeline** Grundlegend benötigt wurde eine Datenpipeline. In Entwicklung mit Herr N. Mavani sollte für den Vergleich von BNN, ENN als Baseline von Mavani Seite auf Depeweg (vgl. Depeweg [83]), von meiner Seite auf Ulmer (vgl. Ulmer et al. [7]) aufgesetzt werden.

Dazu wurden von Depeweg (vgl. Depeweg [83]) verwendete Datensätze verwendet und ergänzt um im Anhang, Abschnitt Datensätze gelistete. Aufgesetzte Stufen der Datenpipeline sind eine Vorverarbeitung zum Umwandeln von Formaten, Einlesen in Torch Dataset Klassen, Torch Trainingsklassen, Visualisierungsklassen.

Aus den Datensätzen wurden Trainings und Visualisierungen abgeleitet und generifiziert. Ebenso wurden recherchierte Loss-Funktionen generifiziert. Trainings wurden auf einen modernen Standard ergänzt wie Checkpoints, Patience-Epochen, vergleichbar und generifiziert. Nach Generifizierung bleiben einer Klasse überwiegend nur noch konfigurierbare Anteile vorbehalten. Eine Konfiguration wurde als Refactor geplant und soll von dort aus zu einer zentralen Konfiguration hochgereicht werden.

Im Projekt behandelte Metriken wurden aufgefasst mit einer Klasse Metrik, Metrik-Thresholds, Metrik-Registry, Metrik-Registry-Definitions deren Name die Funktion hinreichend vorgibt.

Integriert wurden Lösungen von Herr M.Sc. Mavani der sich zur BNN Auswertung für ein Python Pyro Jax Backend entschieden und damit Lösungen nach den Pyroworkflows für Hamilton Monte Carlo (HMC), Stochastic Variational Inference (SVI) geliefert hat.

**Zwischenfazit Datenpipeline** Die Recherche belegt einen hochindividualisierten Grad in der Wahl der Metriken zur Belegung der Unsicherheiten in einer Datenpipeline. Ebenso wie die Skript-Ausgestaltung softwareseitig zur Ermittlung dieser. Josang zeigt für BNN auf, dass dies durch explizite Ausmodellierung robust gestaltet werden kann, und erläutert dazu ein explizit methodisches Vorgehen. Um das Einbeziehen beliebiger Metriken möglich zu machen, wurde ein Hauptdurchlauf bezüglich EDL fertig gestellt, der zu 80% auf die Tensor-Schnittstellen passend zu den gewählten Datensets durchläuft. Ein grundlegendes Zwischenziel war die Möglichkeit der Erarbeitung der Optimierung eigener Modelle bezüglich der verglichenen Architekturen, wobei in meinem Ansatz ENN bevorzugt wurden. Weiterhin wurde darauf abgezielt bei zukünftigen fertigen Trainings zum Beispiel ENN, BNN Vergleiche herstellen zu können. Da folgende Ausarbeitungen der Thesis bevorzugt wurden, ist die Fertigstellung nicht weiter priorisiert worden.

## Experiment zur Approximation der Generalisierung von Hypothesen zur Unsicherheit über verschiedene Datensätze

Grundsätzlich ergab sich aus der Recherche ein überdurchschnittlich hoher Anteil an Papern, die eine Loss-Variation zur Bewertung der Performance von vorhandenen Modellen BNN, ENN durchgeführt haben, mit der besten Übersicht wie erwähnt in Ulmer et al. (vgl. Ulmer et al. [7]). Loss-Funktions Kandidaten  $\mathcal{L}$  werden über den Datensätzen  $\mathcal{D}$  von Depeweg ausgeführt und gemittelt bezüglich der Accuracy ( $Acc$ ) auf dem  $i$ -ten Datensatz,  $\overline{Acc} = \sum_{i=1}^n Acc(x)_i$ . Es wurde grob approximiert, um eine Tendenz erkennbar zu unterscheiden. Erwartet wurde, dass kein eindeutiger Kandidat der Loss-Funktionen über die Accuracy favorisierbar ist, um die Vertrauenswürdigkeit der Netze über verschiedene Datensätze zu bewerten. Die Tabelle fasst nachfolgenden Ausdruck zusammen. Erneut sei dabei erwähnt, dass keine nachhaltige Stichprobe über einen vollparametrisierten Test erhoben, sondern aus manuellen Skriptdurchläufen approximiert und gemittelt wurde, um eine Tendenz festzustellen.

$$\begin{aligned} \mathcal{D} \times \mathcal{L}, \\ \mathcal{D} = \{\text{BH, CCPP, CCS, CBM, EE, I, K8NM, WQR, WQW}\}, \\ \mathcal{L} = \{\text{NLL, ABS, MSE, KL, SCALED, VARIATIONAL, FULL}\} \end{aligned}$$

|                        | nll        | abs        | mse        | kl         | scaled     | var        | full       |
|------------------------|------------|------------|------------|------------|------------|------------|------------|
| $\overline{Acc}_{BNN}$ | [75%, 85%] | [65%, 80%] | [60%, 75%] | [70%, 85%] | [70%, 80%] | [75%, 85%] | [80%, 90%] |
| $\overline{Acc}_{ENN}$ | [80%, 90%] | [70%, 85%] | [70%, 85%] | [75%, 90%] | [75%, 85%] | [80%, 90%] | [85%, 95%] |

Tabelle 8.4.: Bewertung BNN, ENN Loss über verschiedene Datensätze

**Zwischenfazit zur Loss-Variation** Die Loss-Variation erreicht die besten Ergebnisse unter Accuracy im Hinblick auf die „vollparametrisierte“, „full“ Loss-Funktion unter ENN, sowie folgend BNN. Dies *widerlegt* die Hypothese, dass der kombinierte „full“ Loss - angepasst auf mehrere Ziele - den anderen Loss-Kandidaten im Vergleich *nicht* vorzuziehen ist für beide Netzwerke im Hinblick auf die Ergebnisse auf den Testdaten. Dieses Zwischenergebnis soll in der weiteren Durchführung berücksichtigt werden, indem Kritikalitätsraum-abhängige Eigenschaften zur Optimierung in die Loss-Funktion eingebunden werden.

## Visueller Abgleich BNN-ENN

Nach Reproduktion von Aminis Benchmark, wird die Methode EDL auf einer Beispielfunktion nach Depeweg mit einem visuellem Regressions-Benchmark mit Herr Mavanis Ergebnissen

verglichen (vgl. Depeweg [83, S.56, (4.1)]). In ersten Entwicklungen aus denen ich die Ergebnisse erhalten habe, umfingen diese deutlich über hundert Sample-Punkte auf einem vom BNN verrauschten  $2 - \sigma$ -Standardabweichung um den prädikativen Mittelwert in der vom Netz beobachteten Funktion, um wie wahre Funktion (8.1).

$$y = 7 \cdot \sin(x) + 3 \cdot \left| \cos\left(\frac{x}{2}\right) \right| \cdot \epsilon \quad (8.1)$$

Im Verlauf sollten diese von Herr Mavani reduziert werden auf zwanzig bis dreißig Sample-Punkte. Es sollen grundlegende Aspekte verglichen werden. Diese Aspekte zeigen die projektbegleitenden Unterschiede der Arbeiten auf. Für ENN sollte die Aufteilung in EU, AU untersucht werden. Da der Aspekt der direkten Erfassung von ENN attraktiv für weitere Untersuchungen erscheint. Der zentrale Unterschied ist dabei, dass die Teilung der additiven Komponenten EU, AU auf Varianz-Ebene  $\sigma^2$  stattfindet, wie in Abbildung 8.3. zu sehen. Während die Standardabweichung als Gesamtunsicherheit der Streuung unter der Herr Mavani seine Thesis-Ziele optimiert eine Dimension darunter  $\sigma$  stattfinden. Umgangssprachlich lässt sich das wie folgt trennen, dass Herr Mavani über die Samplezahl die Frage verfolgt, wie breit ein Bereich sei, in dem  $\sigma$ -abhängig viele Werte liegen, während bei mir der Frage nachgegangen wird, wie hoch der aleatorische und epistemische Anteil ist. Da in meinem Fall die EU, AU weiter erklärt, analysiert und getrennt werden sollte, ist für EDL die Varianz notwendig.

**Sample-Zahl** Die (Abbildung 8.1., Abbildung 8.2.) erzeugen Vorhersagen über Monte-Carlo-Sampling mit über 100 Einzelvorhersagen, was zu geglätteten Unsicherheitsintervallen führt. Das evidenzbasierte Netz verwendet kein Sampling, sondern gibt direkte Unsicherheitsverteilungen aus - was zu stärker sichtbarer Streuung führt, insbesondere in Regionen mit hoher EU oder AU.

**Aleatorische Varianz** Im evidenzbasierten Ansatz (Abbildung 8.3., orange Fläche) wird die AU explizit separat dargestellt. Diese bleibt in den datenreichen Bereichen konstant und dominiert dort die Gesamtvarianz, während sie bei den nur implizit enthalten ist.

**Epistemische Varianz** Die epistemische Unsicherheit (Abbildung 8.3., rosa Fläche) ist beim evidenzbasierten Netz hauptsächlich in datenarmen Randbereichen sichtbar. Im Gegensatz dazu überdecken die Unsicherheitsbereiche der bayesianischen Methode Abbildung 8.1. Abbildung 8.2) beides gemeinsam und sind somit weniger trennscharf.

**Heteroskedastischer Anteil** Beide Methoden erfassen die Heteroskedastizität der Daten. Beim evidenzbasierten Netz ist sie explizit modelliert und visualisiert (Verengung und Weitung

der Unsicherheitsbereiche in Abbildung 8.3.), bei den implizit über die Streuung der Samples sichtbar.

**Homoskedastischer Anteil** Ein konstanter Unsicherheitsanteil - Basisrauschen - ist bei beiden Methoden vorhanden, beim evidenzbasierten Modell durch die flache orange Fläche innerhalb des Datenbereichs erkennbar, bei den schwerer abgrenzbar, da keine explizite Zerlegung erfolgt.

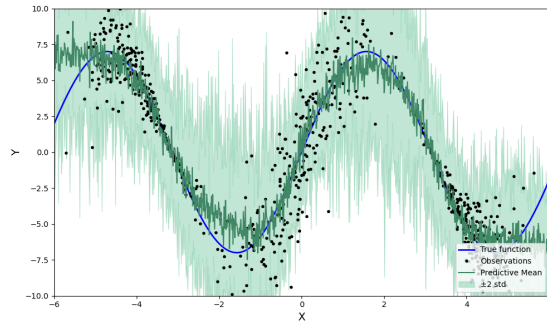
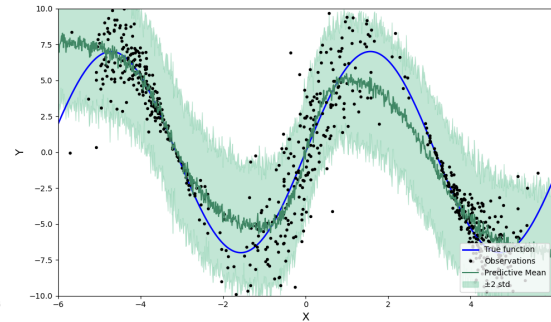


Abbildung 8.1.: Ninand Mavani Visual Benchmark BNN HMC auf (8.1)



SVI auf (8.1)

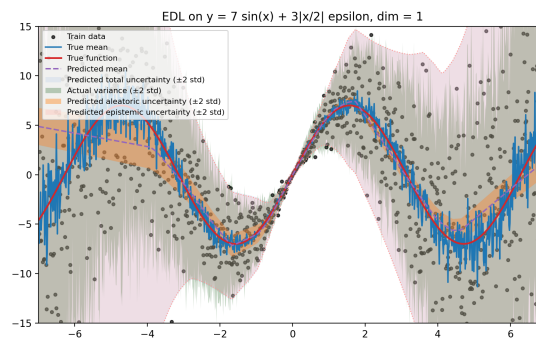


Abbildung 8.3.: Marten Windler Visual Benchmark ENN auf (8.1)

## VaMAI - Kritikalitätsraum - Validierung - Experiment - Fit der Vorhersage zur Ground Truth

Herr Autenrieths Kritikalitätsraum-Software wurde eingebunden in die bisherige Eigenentwicklung. Die Umgebung liefert einen Raum in dem beispielhafte Signalumgebungen zusammengesetzt werden können. Er ist bereits grundlegend unsicherheitsbehaftbar. Zuerst wurde eine Visualisierung nach dem Ansatz der „Deep Evidential Regression“ ergänzt (vgl. Amini

[amini2020]). Als nächstes wurde ein EDL-Training unter Aminis vorhandenen bereits optimierten Modellen (siehe Rekonstruktion Aminis Ergebnisse) in den Hauptablauf integriert. Die synthetischen Daten ersetzen in diesem Fall reale Messungen, Simulationsergebnisse  $\{x_i, y_i\}$ .  $x_i$  ist ein Eingabepunkt der als kleinste Einheit bereits als Szenario im Parameterraum zählen kann.  $y_i$  liefert die Kritikalität in diesem Punkt. Das EDL-Modell beantwortet die Frage, mit welcher Unsicherheit es glaube, dass  $y$  an einem neuen Punkt läge.

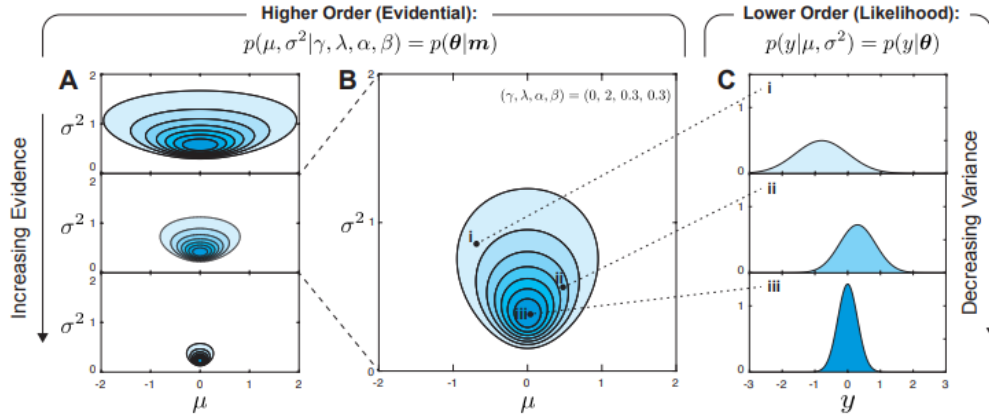


Abbildung 8.4.: R4.4 Angen. Grund-Verteilung auf unbek. Daten für EDL-Modell n. Amini

Das evidentielle Modell nach Amini ordnet das als  $y \sim \mathcal{N}(\mu, \sigma^2)$  zu.  $(\mu, \sigma^2)$  kennt das Modell nicht aus Vorwissen. Es nimmt stattdessen an, erstens  $\mu$  ist unsicherheitsbehaftet, damit folgt  $\mu \sim \mathcal{N}(\gamma, \sigma^2/\lambda)$ . Zweitens  $\sigma^2$  ist unsicherheitsbehaftet, damit folgt  $\sigma^2 \sim \text{NIG}(\alpha, \beta)$ . So sucht das Modell  $p(y|\gamma, \lambda, \alpha, \beta)$ . Die Daten liefern die Evidenz als Ergebnis. Das EDL-Modell schätzt direkt die Hyperparameter ausgehend von Eingaben  $x$ ,  $f(x) \rightarrow \{\gamma(x), \lambda(x), \alpha(x), \beta(x)\}$ . Es ergibt sich für jeden Punkt  $x$  eine bedingte Unsicherheitsverteilung für  $y$ . Amini verdeutlicht dies kompakt in Gleichungen (3), (4), (5), Abbildung R.4.4 (vgl. Amini 2020 [amini2020]).  
Ergebende Unsicherheitsmasse ist Student-t-verteilt (vgl. Gleichungen (6), (7)), damit ist Mittelwert  $\mu$  als „Position“, Skalierung  $\sigma^2$  als „Streuung“, Freiheitsgrad  $\nu$  als „Ordnung“ bekannt.

$$f : \mathbb{R}^d \rightarrow \mathbb{R}^4, \quad x \mapsto (\gamma, \lambda, \alpha, \beta) \quad \Rightarrow \quad p(y | x) \sim \text{Student-}t \left( \mu = \gamma, \sigma^2 = \frac{\alpha\lambda}{\beta(1+\lambda)}, \nu = 2\alpha \right)$$

R4.4 visualisiert die Zusammensetzung der Student- $t$ -Verteilung in Abhängigkeit von der Evidenz im Ausdruck der Ordnung: Bei niedriger Ordnung sowie Evidenz ( $\nearrow \alpha \Rightarrow \nearrow \nu$ ) ergibt sich eine breite, unsichere Verteilung. Während höhere Ordnung sowie Evidenz ( $\searrow \alpha \Rightarrow \searrow \nu$ ) zu einer schmalen, sichereren Verteilung führt, welche gegen die Normalverteilung konver-

giert. Alle weiteren Proportionalitäten für die evidentiellen Parameter sind in (5) einsehbar (vgl. Amini [amini2020]). Erste vom Modell geschätzte Wahrscheinlichkeitsmasse um einen Schätzwert ergibt sich aus Gleichung (4) ohne Regularisierung in R4.4. platikurtisch hellblau. Regularisierungsbedingungen formen die Verteilung unter Evidenz-Kriterien leptokurtisch dunkelblau. Machine-Learning-basiert regularisieren Loss-Funktion nach dem Min-Max-Prinzip angenommene Fehler- gegen Zielgrößen, hier durch Amini beschränkt Varianz gegenüber Evidenz (vgl. Gleichung (8), (9), (10) unter (5)). Amini wählt die Kullback-Leibler-Divergenz als Maß für die Ausgangs- zur Optimalverteilung. Überlässt dabei jedoch ein Detail dem geneigten Leser, dessen Behandlung im Stand der Technik mittels Josangs SL aufgezeigt wurde.

„[...] the KL between any NIG and the zero evidence NIG prior is undefined.“  
[26, S. 5, „Minimizing evidence on errors“, Z. 6 ff.]

Herr Autenrieths Kritikalitätsräume erlauben folgende Parametrisierung:

- **Gaussverteilung:**

$$f(\mathbf{x}) = A \cdot \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu})\right)$$

mit:

- $\boldsymbol{\mu} \in \mathbb{R}^d$ : Mittelwert
- $\Sigma = \text{diag}(\boldsymbol{\sigma}^2)$  mit  $\boldsymbol{\sigma} \in \mathbb{R}_{>0}^d$ : Standardabweichungen
- $A > 0$ : maximale Amplitude (optional)
- $\text{range\_factor} \in \mathbb{R}_{>0}$ : Ausdehnung (optional)
- $\text{active\_dims}$ : aktive Eingabedimensionen  $\subseteq \{1, \dots, D\}$

- **Statischer Hyperwürfel:**

$$f(\mathbf{x}) = \begin{cases} c, & \text{wenn } \mathbf{x} \in \prod_{i=1}^d [a_i, b_i] \\ 0, & \text{sonst} \end{cases}$$

mit:

- $[a_i, b_i]$ : Begrenzung pro Dimension
- $c \in \mathbb{R}$ : Konstanter Wert

• **Rampe:**

$$f(\mathbf{x}) = \text{clip} \left( \sum_{i=1}^d \alpha_i x_i, \min, \max \right)$$

mit:

- $\alpha \in \mathbb{R}^d$ : Koeffizienten
- bounds:  $[a_i, b_i]$  für jede aktive Dimension
- min\_value, max\_value: Begrenzung des Funktionswerts

• **Rauschen:**

$$f(\mathbf{x}) = A + \mathcal{N}(0, \sigma^2)$$

mit:

- $A$ : Basiswert der Amplitude
- $\sigma$ : Standardabweichung des Rauschens (optional)
- seed: zufallsbasierte Reproduzierbarkeit (optional)

• **Sinus:**

$$f(\mathbf{x}) = \sum_{i=1}^n A_i \cdot \sin(2\pi f_i \cdot \mathbf{w}_i^\top \mathbf{x} + \phi_i) + \text{offset}$$

mit:

- $A_i$ : Amplitude
- $f_i$ : Frequenz
- $\mathbf{w}_i \in \mathbb{R}^d$ : Gewichtungsvektoren
- $\phi_i$ : Phase (optional)
- offset: Verschiebung nach oben/unten (optional)
- minimum\_value: untere Schranke (optional)
- bounds: Definitionsbereich (optional)
- active\_dims: aktive Eingabedimensionen  $\subseteq \{1, \dots, D\}$

Herr Autenrieths Kritikalitätsraums Unsicherheitsbehaftung wird zunächst äquivalent zu Aminis Modell R.4.4 refaktoriisiert indem Aminis „Evidential Deep Learning“ Repository wegen fehlender Projektstruktur direkt integriert wird. Visualisiert wird zunächst die dem Modell unbekannte ursprüngliche Unsicherheit mit der das Signal belegt ist, die aleatorische, epistemische Varianz, sowie eine Gesamtansicht mit dem ursprünglich gesampelten Kritikalitätsraum. Das

VaMAI Standardbeispiel liefert dazu ein Rampen-, Gauss- und multinomielles Gausssignal in einer Domäne 10 LE x 10 LE x 2 LE. Die Ausgabe des unoptimierten Modells trennt zwar EU und AU. Doch die Gesamtunsicherheitsmasse des Modells kommt der wahren unterliegenden Unsicherheit ungenügend nach. Es soll als visueller Ausgangspunkt weiterverwendet werden. Die Parametrisierung wurde zwecks UI-Optimierung für kurze Ladezeit verwendet.

**Approximations-Parametrisierung ENN** Zur Bewertung von Approximationen des ENN wird folgende Bewertung herangezogen. Es wird ein Original-Signalverlauf  $y(x)$  zum vom Netz rekonstruierten  $\hat{y}(x)$  abgebildet als Fehler-Signal  $e(x)$ :

$$e(x) = y(x) - \hat{y}(x) := \underbrace{\frac{1}{\sigma \cdot \sqrt{2\pi}} \cdot \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)}_{\text{Amplitude } A_1 \text{ Gauss-Signal 1}} - \underbrace{\frac{1}{\sigma \cdot \sqrt{2\pi}} \cdot \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)}_{\text{Amplitude } A_2 \text{ Gauss-Signal 2}} \quad \text{mit } A_1 \geq A_2 \quad (8.2)$$

$$:= \underbrace{A_1 \cdot \mathbf{1}_{\{|x-x_1| \leq \delta, |y-y_1| \leq \delta\}}}_{\text{Hyperwürfel 1}} - \underbrace{A_2 \cdot \mathbf{1}_{\{|x-x_2| \leq \delta, |y-y_2| \leq \delta\}}}_{\text{Hyperwürfel 2}} \quad \text{mit } A_1 \geq A_2 \quad (8.3)$$

Aminis Modell ergibt die aleatorische Varianz als inverse Varianz

$$\sigma^2 = \frac{\beta}{1-\alpha} := \frac{1}{\lambda} \text{ nach Amini (5.2)}$$

Ebenso die epistemische Varianz in Abhängigkeit vom Mittelwert des Signals

$$\sigma_{\text{epistemisch}}^2 = \frac{\beta}{\lambda(\alpha-1)} \quad \text{für } \alpha > 1 \text{ nach Amini (5.2)}$$

Eine der beiden Bedingungen ist als Vorwissen des Autors von Signalvergleichen mittels EDL zum Abgleich einer guten Approximation sinnvoll, um die Amplitude zur Approximation abzuschätzen. Für das im Experiment verwendete Gauß-Signal ergibt sich die optimale Approximation bei einer Amplitude  $A = 1$ . Abweichungen von diesem Wert werden als Unterapproximation ( $A < 1$ ), Überapproximation ( $A > 1$ ) der Gesamtunsicherheit gewertet - interpretiert als additive Varianz nach (5.1). Gleiches gilt für den Hyperwürfel. AU, EU werden tabellarisch eingeordnet.  $e(x)$  quantifiziert damit die Approximation für zu betrachtende Unsicherheitsarten über die Amplitude zur Signal-Klasse.

An dieser Stelle wurde der Under-, Over-, Fit Begriff diskutiert und für unpräzise empfunden,

weil für die Unsicherheitsmassen  $\mu$ , AU, EU nicht untersucht wird, wie sich das Modell gegenüber einer dynamischen Veränderung der Sample-Zahl in Bezug auf Generalisierbarkeit, Robustheit in der Modellkomplexität verhält. Ziel war es zunächst vielmehr, statische Signalzustände sowie Sample-Zahl-Variationen direkt zu approximieren und daraus Rückschlüsse auf die Unsicherheitsverteilung zu ziehen.

| Modellverhalten     | AU              | EU     | Signalabdeckung | Risiko                     |
|---------------------|-----------------|--------|-----------------|----------------------------|
| <b>Unterapprox.</b> | Hoch / beliebig | Hoch   | Meist breit     | Ignoriert Muster           |
| <b>Approximativ</b> | Realistisch     | Gering | Angemessen      | Gut generalisierend        |
| <b>Überapprox.</b>  | Niedrig         | Gering | Eng             | Übersicher, fehleranfällig |

Tabelle 8.5.: ENN Modellverhalten und Unsicherheitsmerkmale in der Signalrekonstruktion

**Ausgangs-Konfiguration „Prototyping“** Um das UI für die Ausgangssituation des VaMAI Standardbeispiels zu erarbeiten, wurde folgende Ausgangs-Konfiguration mit Aminis ENN eingebunden. Feedforward-Netzwerk mit einer versteckten Schicht (`Dense(64, relu)`), Ausgangs-layer `DenseNormalGamma`. Optimierer: Adam, Lernrate:  $1 \cdot 10^{-3}$ . Verlustfunktion: `EvidentialRegression` mit Regularisierung `coeff=1e-2`. Trainingsdauer: 100 Epochen. Trainingsdaten: Latin Hypercube Sampling (101 Samples), Rauschen: Gauß-verteiltes Rauschen mit  $\sigma = 0,01$ . Evaluation mittels RMSE auf Testdaten (80/20 Split), 5 Wiederholungen. Ausgaben:  $\gamma, \nu, \alpha, \beta$  zur Berechnung von aleatorischer und epistemischer Varianz.

**Ausgangsfunktionen** Allgemein wurde das VaMAI Standardbeispiel auf zwei Funktionen reduziert zur Untersuchung in einer Domäne 10 LE x 10 LE x 1 LE: Erstens eine Gauss-Funktion,

$$f(x, y) = A \cdot \exp \left( - \left( \frac{(x - \mu_x)^2}{2\sigma_x^2} + \frac{(y - \mu_y)^2}{2\sigma_y^2} \right) \right)$$

$$\text{mit: } \mu_x = 5, \quad \mu_y = 5, \quad \sigma_x = 1, \quad \sigma_y = 1, \quad A = 1$$

$$\Rightarrow f(x, y) = \exp \left( - \frac{(x - 5)^2 + (y - 5)^2}{2} \right),$$

Diese Funktion hat ihr Maximum bei (5, 5) und fällt exponentiell ab, erreicht jedoch nie exakt den Wert Null. Zweitens ein Hyperwürfel,

$$f(x, y) = \begin{cases} 1, & \text{wenn } |x - 5| \leq \delta \text{ und } |y - 5| \leq \delta \\ 0, & \text{sonst} \end{cases}$$

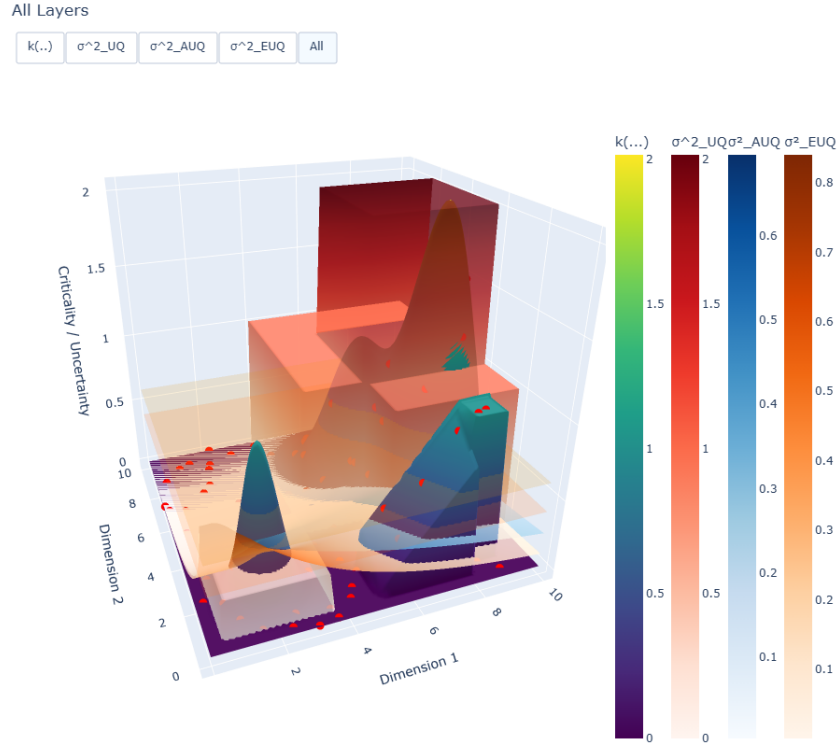


Abbildung 8.5.: R Grundl. Ausgangs-UI Kritikalität, Varianz total, aleat., epist.

Hierbei ist  $\delta > 0$  die Halbseitenlänge des Würfels. Für  $\delta = 0,5$  ergibt sich ein Quadrat um den Punkt  $(5, 5)$  im Bereich  $[4,5, 5,5] \times [4,5, 5,5]$ .

**Ausgangs-Sampling** Das ursprüngliche Latin-Hypercube-Sampling von Herr Autenriehts Kritikalitätsraum soll in der Samplezahl variiert werden mit  $n = \{n_{min} := 10, n_{\frac{\Delta}{2}}, n_{\Delta}\}$ .  $n_{\Delta}$  als Maximum soll abhängig von der Konvergenz der Loss-Funktion sein, die sich an diesem Punkt nicht mehr signifikant ändert. Dieser wird durch eine Schleife über die Trainings bestimmt aus vorgegebenen Samplegrößen zum Vereinfachen des Experiments,  $n = \{10, 50, 100, 150, 200\}$ .  $n_{\frac{\Delta}{2}}$  entspricht dem intervallhalbierenden Wert von  $[n_{min}, n_{\Delta}]$ . Im Experiment werden zuerst  $n_{min}$ , dann anhand der Konvergenzrichtung sowie vorgegebenen  $n, n_{\Delta}$  bestimmt und schließlich  $n_{\frac{\Delta}{2}}$ . Für jede gewählte Samplesize  $n$  wurde das Modell über eine feste Anzahl Epochen  $e := \{2000, \dots, 8000\}$  trainiert. Nach jeder Epoche wurden der Trainings- und der Validierungsverlust aufgezeichnet. Die Epoche mit dem minimalen Validierungsverlust wurde als beste Epoche bestimmt. Je höher die Epoche, umso vergleichbarer konvergieren Trainings- und Validierungsloss. Je stärker Trainings- und Validierungsloss konvergieren, umso mehr wird dem Modell ein

generalisierbarer Fit an die Ausgangsdaten zugeordnet.

**Under-, Over-, Fit-Definition** Der Begriff *Fit* beschreibt, wie gut ein Modell die zugrunde liegende Datenverteilung approximiert. Mathematisch versucht das Modell  $f_\theta(x)$ , die unbekannte Zielfunktion  $y = f(\dots)$  möglichst genau zu approximieren. Ein guter Fit bedeutet, dass sowohl der Trainings-Loss als auch der Validierungs-Loss niedrig sind und ähnliche Werte aufweisen. Der Trainings-Loss  $L_{\text{train}}$  misst den Fehler zwischen den Modellvorhersagen  $\hat{y}$  und der Ground Truth  $y$  auf den Trainingsdaten, während der Validierungs-Loss  $L_{\text{val}}$  den Fehler auf einem unabhängigen Validierungsdatensatz angibt.

Sind beide Loss-Werte hoch, liegt ein *Underfit* vor. Das Modell hat also die Struktur der Daten nicht erfasst. Sind beide Werte niedrig und ähnlich groß, spricht man von einem guten Fit. Da das Modell sowohl die Trainingsdaten gut approximiert, als auch auf neuen Daten gut generalisiert. Ist der Trainings-Loss niedrig, der Validierungs-Loss jedoch deutlich höher, liegt ein Überfit oder Overfit vor. Das Modell hat sich zu stark an die Trainingsdaten angepasst und generalisiert schlechter.

Die *Ground Truth* bezeichnet die wahren Zielwerte  $y$  im Datensatz, gegen die die Vorhersagen  $\hat{y}$  verglichen werden. In diesem Experiment sind dies analytische Werte aus einer bekannten Funktion wie dem Gauß-Signal, dem Hyperwürfel-Signal. Sie dienen als Referenz für die Berechnung von Loss und RMSE.

In diesem Experiment beschreibt der *Fit* die Übereinstimmung zwischen den Modellvorhersagen  $\hat{y}$  und der bekannten Ground Truth  $y$ , gemessen durch den Trainings- und Validierungs-Loss. Ein guter Fit liegt vor, wenn beide Loss-Werte niedrig und ähnlich sind, was auf eine gelungene Annäherung der Ground-Truth-Funktion bei gleichzeitiger Generalisierungsfähigkeit hinweist. Im Folgenden ist weiterhin zwischen *Approximation* und *Fit* zu unterscheiden. Approximiert wird der Signalwert  $y$  als Größe abhängig in VaMAI von der aus meiner Projektperspektive offen parametrisierten Kritikalität  $k(\dots)$ , die vom ENN mittels  $\gamma = \mu$  direkt als Netz-Ausgabe-Parameter geschätzt wird. Hier lässt sich eine direkte Fehlerapproximation  $e(x) = y - \hat{y}$  des Original-Signals als Ground-Truth mit dem Fehler-Signal durchführen wie durch die vorherigen Amplitudengleichungen für Gauss- und Hyperwürfel angedeutet. Diese Signalapproximation wurde im Projektverlauf getrennt vom Fit-Begriff. Der wie oben beschrieben erstens von der Loss-Konvergenz, zweitens von der Abweichung einer Fehlermetrik wie *RMSE* zur Ground-Truth abhängig gemacht wurde. Dieser Unterschied soll im Folgenden durch die ermittelten optimalen Samplegrößen aus dem Experiment im Abgleich zur Konvergenz aus zur Vereinfachung fest vorgegebenen Set  $n = \{10, 50, 100, 150, 200\} \rightarrow \{n_{\min}, n_{\frac{\Delta}{2}}, n_{\Delta}\}$  zugeordnet werden. Dazu wird die rechte Seite der Zuordnung den drei Ergebnisgrößen des Netzes -  $\mu = \gamma$  als Mittelwert,  $\sigma_{alea}^2$  als aleatorische Unsicherheit,  $\sigma_{epist}^2$  als epistemische Unsicherheit - über den Raum zugeordnet.

Es wird für jede Konfiguration  $\{\text{Signal} =: s, \text{Epoche} = ep, \text{Neuronen} =: ne, \text{Schichten} =: l\}$  eine  $3 \times 3$ -Matrix der Ergebnisoberflächen, sowie separat des assoziierten Loss in  $1 \times 3$ -Matrix, sowie assoziierter Ground-Truth-zur-Vorhersage-Plot als  $1 \times 3$ -Matrix zur Bewertung abgebildet. Die Ergebnisoberflächen definieren sich in Abhängigkeit der evidentiellen Parameter, sowie den Parameterkoordinaten der Oberfläche - für die z-Koordinate wurden Aminins Gleichungen aleatorischer, epistemischer Unsicherheit gewählt.  $u, v$  sind beliebig zu vergleichende Dimensionen am VaMAI Standardbeispiel  $0 \leq u \leq 10, 0 \leq v \leq 10, (u, v) \in \mathbb{R}^3$ . Sie werden als Ausschnitt der für VaMAI geforderten Hochdimensionalität im allgemeinen  $R^n \subseteq \mathbb{K}^n$  betrachtet.

$u, v \in \mathbb{R}^3 \subseteq \mathbb{K}^n$  Zurordnung Oberfläche-Parameterkoordinaten

$$\text{Aleatorische Varianz: } \mathbf{r}_{\text{aleatorisch}}(u, v) = \begin{pmatrix} u \\ v \\ \frac{\beta(u, v)}{\alpha(u, v) - 1} \end{pmatrix}$$

$$\text{Epistemische Varianz: } \mathbf{r}_{\text{epistemisch}}(u, v) = \begin{pmatrix} u \\ v \\ \frac{\beta(u, v)}{v(u, v) \cdot (\alpha(u, v) - 1)} \end{pmatrix}$$

$$\text{Vorhersage } \mu = \gamma \text{ (Näherung an } y\text{): } \mathbf{r}_{\mu}(u, v) = \begin{pmatrix} u \\ v \\ \gamma(u, v) \end{pmatrix}$$

Dazu wird die Ausgangskonfiguration wie folgt überschrieben.

**Überschriebene Ausgangskonfiguration 1** Trainings-Parametrisierung mit zwei vollverbundenen Schichten, 64 Neuronen, 2000 Epochen führt zu Unterapproximation Gauss-Funktion, Hyperwürfel-Funktion.

Einzelnes 3D-Signal EDL, Gauss-Funktion, Unterapproximation für  $n = 100$ ,  $n_{\Delta} = 200$ ,  $n_{\frac{\Delta}{2}} = 150$ ,  $ep = 2000$ ,  $ne = 64$ ,  $l = 2$

$$\gamma = \mu$$

Aleatorische  
Unsicherheit

Epistemische  
Unsicherheit

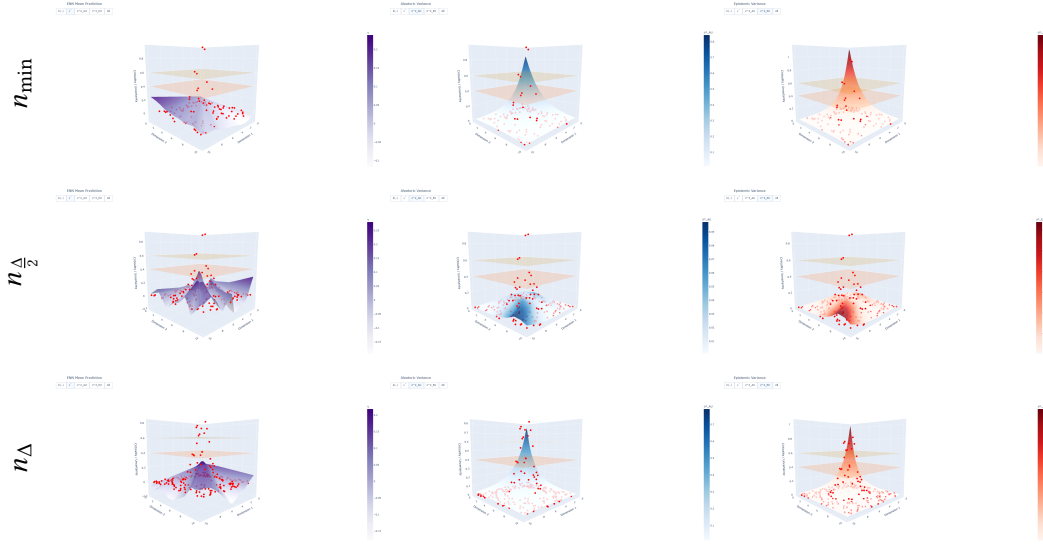


Abbildung 8.6.: Vergleich der evidentiellen Ausgaben für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

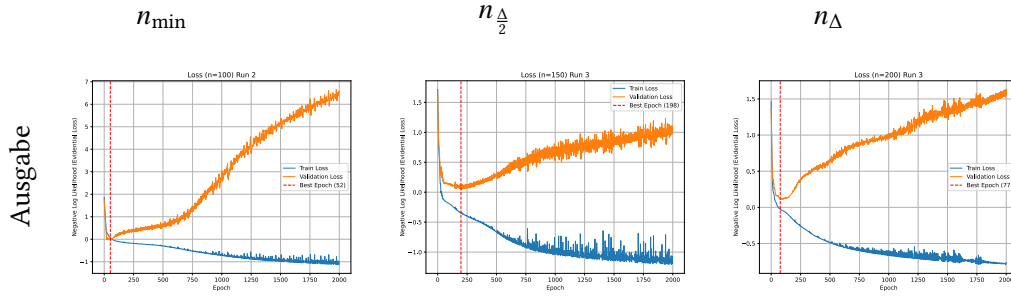


Abbildung 8.7.: Vergleich des Trainings- und Validierungsloss für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

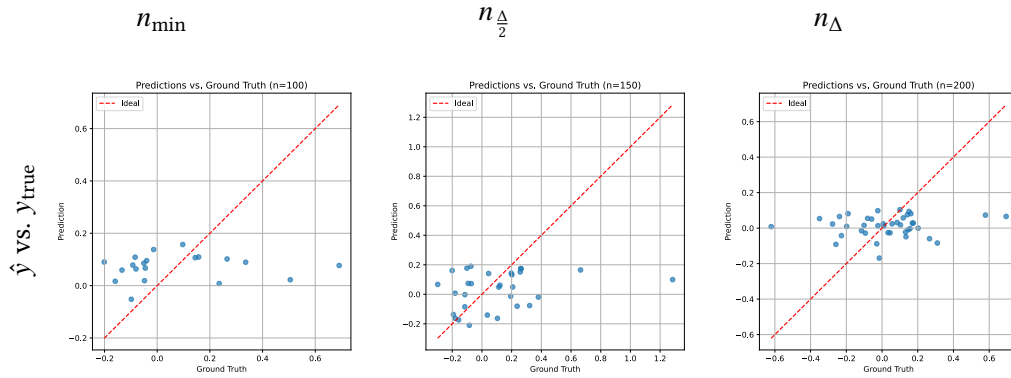


Abbildung 8.8.: Vergleich Ground-Truth  $y_{\text{true}}$  zu Vorhersage  $\hat{y}$  mit  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

**Interpretation, Einzelnes 3D-Signal EDL, Gauss-Funktion, Unterapproximation für**

$n = 100, n_{\Delta} = 150, n_{\frac{\Delta}{2}} = 200, ep = 2000, ne = 64, l = 2$

- Das Signal wird unterapproximiert.
- Die Konvergenz zur besten Epoche ergibt die Maximalwerte von 232, 240, 528 Epochen.
- Das iterative Training wählt basierend auf der Konvergenz von Training- und Validation-Loss höhere Samplingzahlen für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

Einzelnes 3D-Signal EDL, Hyperwürfel-Funktion, Unterapproximation für  $n = 100$ ,

$n_{\Delta} = 200, n_{\frac{\Delta}{2}} = 150, ep = 2000, ne = 64, l = 2$

$\gamma = \mu$

Aleatorische  
Unsicherheit

Epistemische  
Unsicherheit

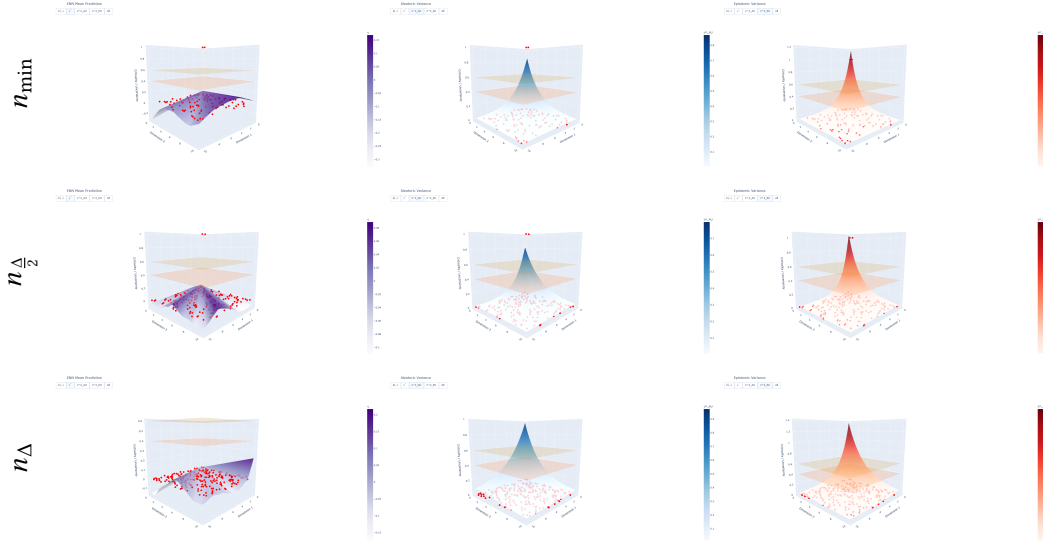


Abbildung 8.9.: Vergleich der evidentiellen Ausgaben für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

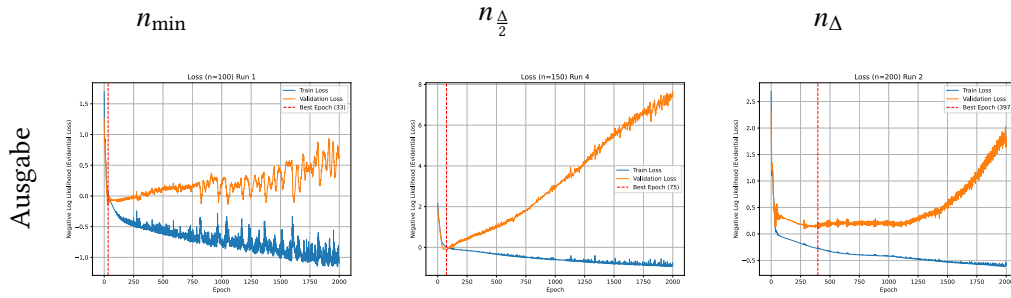


Abbildung 8.10.: Vergleich des Trainings- und Validierungsloss für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

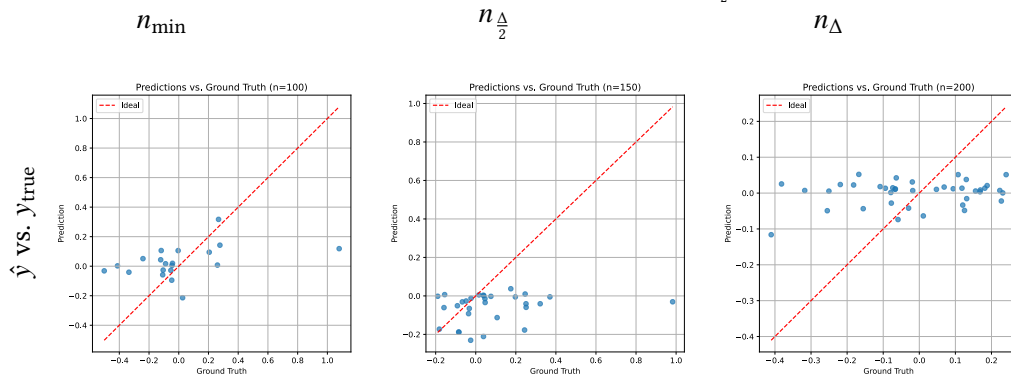


Abbildung 8.11.: Vergleich Ground-Truth  $y_{\text{true}}$  zu Vorhersage  $\hat{y}$  mit  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

**Interpretation, Einzelnes 3D-Signal EDL, Hyperwürfel-Funktion, Unterapproximation**  
für  $n = 100$ ,  $n_{\Delta} = 150$ ,  $n_{\frac{\Delta}{2}} = 200$ ,  $ep = 2000$ ,  $ne = 64$ ,  $l = 2$

- Das Signal wird unterapproximiert.
- Die Konvergenz zur besten Epoche ergibt die Maximalwerte von 554, 215, 449 Epochen.
- Das iterative Training wählt basierend auf der Konvergenz von Training- und Validation-Loss höhere Samplingzahlen für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .
- Der Sprung zur Oberfläche des Hyperwürfels bestärkt die Abhängigkeit vom Latin Hypercube Sampling abgetastet zu werden, damit das ENN den Ausschlag der Werte erkennt.

Einzelnes 3D-Signal EDL, Gauss-Funktion, Überapproximation für  $n = 100$ ,  $n_{\Delta} = 200$ ,  $n_{\frac{\Delta}{2}} = 150$ ,  $ep = 4000$ ,  $ne = 128$ ,  $l = 2$

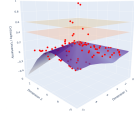
$$\gamma = \mu$$

Aleatorische  
Unsicherheit

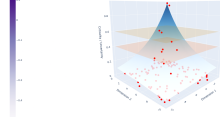
Epistemische  
Unsicherheit

$n_{\min}$

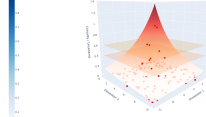
3D Plot



3D Plot

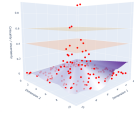


3D Plot

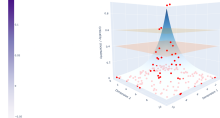


$n_{\frac{\Delta}{2}}$

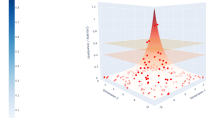
3D Plot



3D Plot

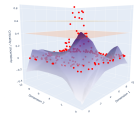


3D Plot

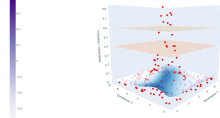


$n_{\Delta}$

3D Plot



3D Plot



3D Plot

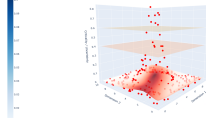


Abbildung 8.12.: Vergleich der evidentiellen Ausgaben für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

$n_{\min}$

$n_{\frac{\Delta}{2}}$

$n_{\Delta}$

Ausgabe

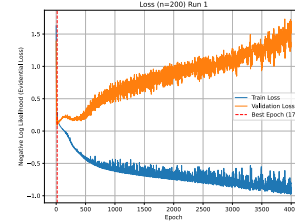
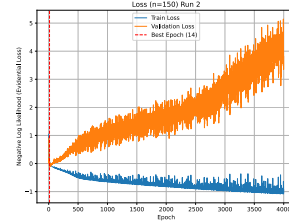
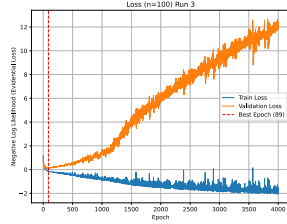


Abbildung 8.13.: Vergleich des Trainings- und Validierungsloss für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

$n_{\min}$

$n_{\frac{\Delta}{2}}$

$n_{\Delta}$

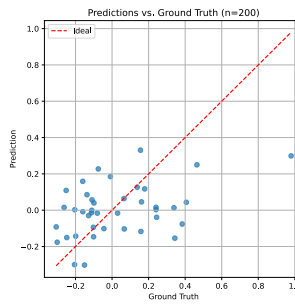
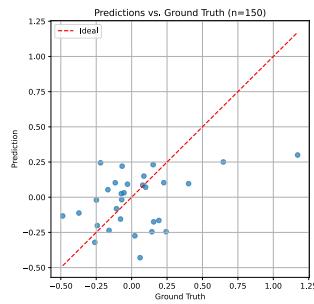
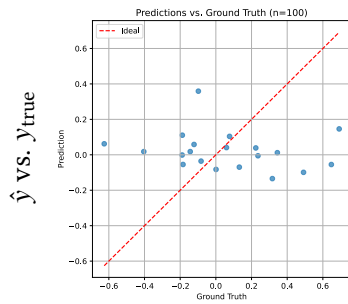


Abbildung 8.14.: Vergleich Ground-Truth  $y_{\text{true}}$  zu Vorhersage  $\hat{y}$  mit  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

**Interpretation, Einzelnes 3D-Signal EDL, Hyperwürfel-Funktion, Unterapproximation**  
für  $n = 100$ ,  $n_{\Delta} = 150$ ,  $n_{\frac{\Delta}{2}} = 200$ ,  $ep = 4000$ ,  $ne = 128$ ,  $l = 2$

- Das Signal wird überapproximiert.
- Die Konvergenz zur besten Epoche ergibt die Maximalwerte von 5, 14, 17 Epochen.
- Das iterative Training wählt basierend auf der Konvergenz von Training- und Validation-Loss höhere Samplingzahlen für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .
- Die Überkonfidenz äußert sich in den besonders in  $\gamma$  für  $n = 200$  Samples, bei der links und rechts vom zu approximierenden Gauss-Signal die Unsicherheitsmasse deutlich über die virtuelle Ebene, welche die meisten Samples verbinden würde, in ihrem Wert  $y$  überschreitet.

Einzelnes 3D-Signal EDL, Hyperwürfel-Funktion, Überapproximation für  $n = 10$ ,  
 $n_{\Delta} = 50$ ,  $n_{\frac{\Delta}{2}} = 100$   $ep = 4000$ ,  $ne = 128$ ,  $l = 2$

$$\gamma = \mu$$

Aleatorische  
Unsicherheit

Epistemische  
Unsicherheit

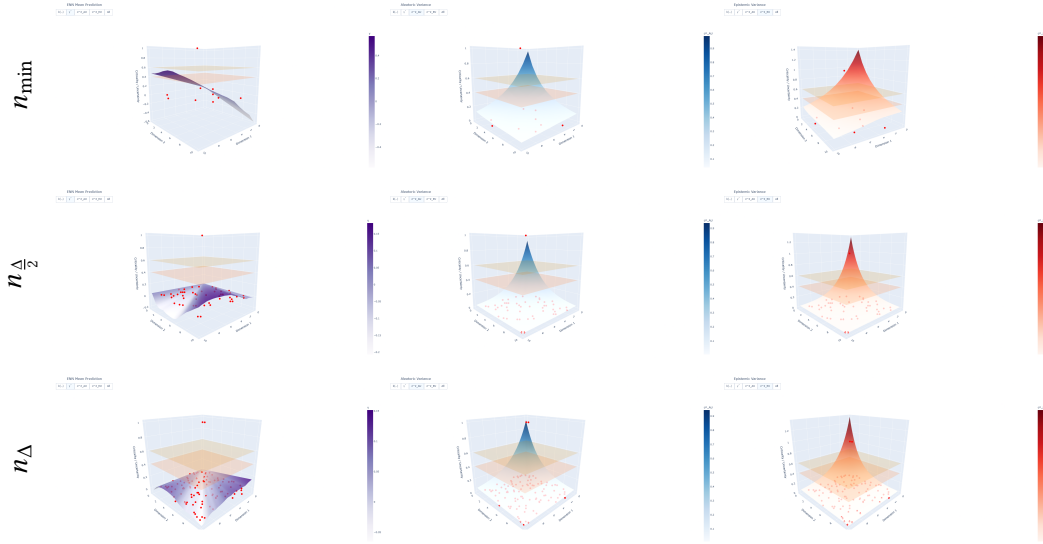


Abbildung 8.15.: Vergleich der evidentiellen Ausgaben für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

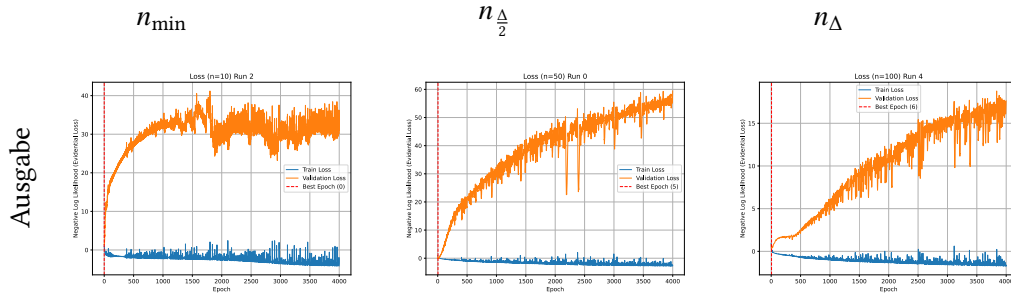


Abbildung 8.16.: Vergleich des Trainings- und Validierungsloss für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

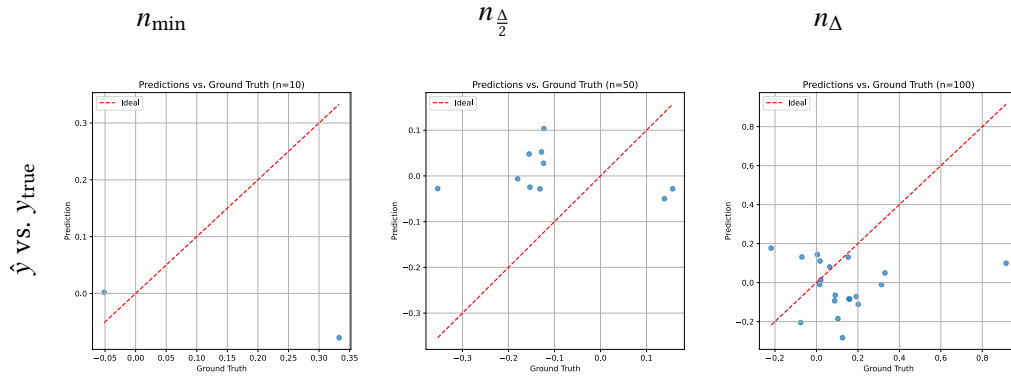


Abbildung 8.17.: Vergleich Ground-Truth  $y_{\text{true}}$  zu Vorhersage  $\hat{y}$  mit  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .

**Interpretation, Einzelnes 3D-Signal EDL, Hyperwürfel-Funktion, Unterapproximation**  
für  $n = 100$ ,  $n_{\Delta} = 150$ ,  $n_{\frac{\Delta}{2}} = 200$ ,  $ep = 4000$ ,  $ne = 128$ ,  $l = 2$

- Das Signal wird überapproximiert.
- Die Konvergenz zur besten Epoche ergibt die Maximalwerte von 0, 5, 6 Epochen.
- Das iterative Training wählt basierend auf der Konvergenz von Training- und Validation-Loss niedrigere Samplingzahlen für  $n_{\min}$ ,  $n_{\frac{\Delta}{2}}$  und  $n_{\Delta}$ .
- Die Überkonfidenz äußert sich besonders mit dem frühen Divergieren über alle niedrigen Sample-Zahlen.

### VaMAI - Kritikalitätsraum - Validierung - Experiment - Approximationen zur Steuerung der UQ

**Multi 3D-Signal EDL Unterapproximation zu annähernder Approximation** Folgendes Multi-3D-Signal zweier bereits mit Unsicherheit belegter Gauss-Signale

$$F_{\text{Gauss}}(x, y) = \exp\left(-\frac{(x-2)^2}{2 \cdot 0.5^2} - \frac{(y-2)^2}{2 \cdot 0.5^2}\right) + \exp\left(-\frac{(x-6)^2}{2 \cdot 0.5^2} - \frac{(y-6)^2}{2 \cdot 0.5^2}\right)$$

$$F_{\text{Hyperwürfel}}(x, y) = \begin{cases} 1, & \text{wenn } |x-2| \leq 0.5 \text{ und } |y-2| \leq 0.5, \\ 1, & \text{wenn } |x-6| \leq 0.5 \text{ und } |y-6| \leq 0.5, \\ 0, & \text{sonst} \end{cases}$$

erzielte auf bisheriger Grundlage mit folgenden Parametrisierungen Unterapproximationen.

| Schichten | Neuronen pro Schicht | Epochen |
|-----------|----------------------|---------|
| 4         | 128                  | 8000    |
| 2         | 256                  | 4000    |
| 2         | 256                  | 2000    |

Tabelle 8.6.: Vergleich von Netzwerkarchitekturen und Trainingsparametern in Konfiguration 4

Zu beobachten ist, dass eine Erhöhung der Schicht- und Neuronen-Anzahl im 3D-Fall dazu beiträgt das größte Teil-Signal präziser auf dem Domänenboden abzubilden bis es die Originalposition des größten Teil-Signals erreicht. Jedoch fällt das Volumen zu flach aus für den angelegten Eingang der Störfunktion. Eine weitere Beobachtung ist, dass Aminis Regressionsansatz bei mehreren Signalformen in der Domäne, im Volumen die Gesamtunsicherheitsmasse der aleatorischen und epistemischen Varianz aller auftretenden Signale erfasst. Ergänzend dazu

orientiert sich die Position der erfassten Gesamtunsicherheitsmasse jedoch im ersten Fall an der größten Dichte von Maximalwerten der auftretenden Signal-Formen. Wählt man schließlich der ursprünglichen Parametrisierung von Amini entgegenkommend 2 Schichten, 64 Neuronen, 2000 Epochen kommt man zum besten gemittelten Ergebnis aus der aleatorischen und epistemischen Varianz zwischen den beiden verwendeten Signalen.

Für weitere Schritte wird das wie folgt interpretiert. Im Projekt VaMAI möchte man die aleatorische und epistemische Varianz pro Signal präziser erfassen, sodass eben geschilderte Beobachtungen nicht hinreichend sind. Lösungsansätze könnten beispielsweise das Eingrenzen von Signal-Positionen in der Domäne durch eine beliebige Abtastung sein. Das abgetastete Teilvolumen wird dann vom Evidentiellen Netzwerk auf seine Teilunsicherheiten untersucht. Das Ergebnis setzt man additiv zusammen. Die Erwartungshaltung ist, dass sich so, sowohl die Position der Teil-Unsicherheitsmassen als auch die Volumen der Teil-Unsicherheitsmassen in Bezug auf die als Boden angesehene Fläche der verglichenen Dimensionen korrekt bestimmen lassen.

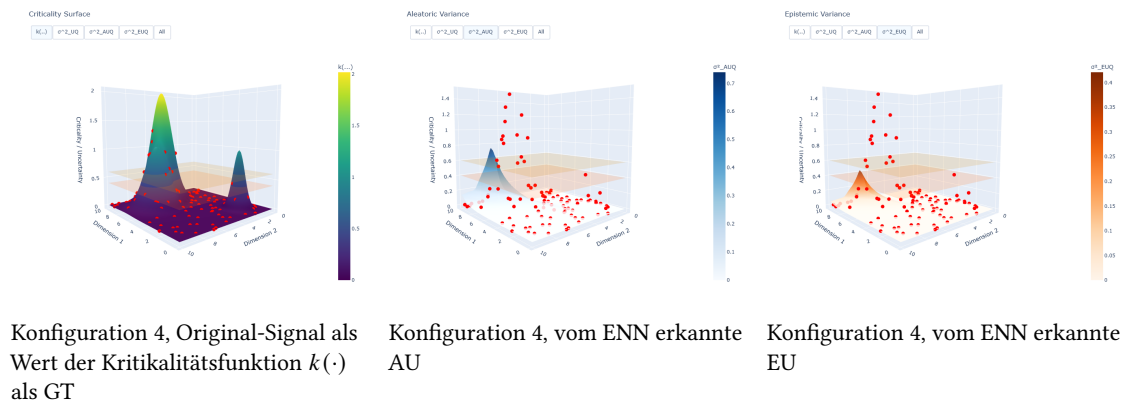


Abbildung 8.18.: Vergleich von GT, AU und EU unter Konfiguration 4

**Multi 3D-Signal EDL Überapproximation** Es wurde im Zeitrahmen unter im letzten Paragraphen erwähnter Parametrisierung kein offensichtlicher Überapproximation erzielt, der alle Signal-Klassen visuell detailgetreu abbildet. Interpretiert wird, dass das größte Teil-Signal in Originalposition des größten Teil-Signals auf dem Domänenboden der verglichenen Dimensionen als erste Beobachtung des letzten Paragraphen unter der hohen Schicht-, Neuronen-, Epochenzahl aus der letzten Tabelle den Überapproximation von Aminis Regressionsansatz bei mehreren 3D-Signalen abbildet.

**Störfunktion** Für stochastische Abweichungen wird in Herr Authenrieths Kritikalitätssoftware die Klasse Noise verwendet. Sie überlagert normalverteiltes Rauschen additiv auf Punktmengen oder Gitter. Die Initialisierung erfolgt über eine Amplitude  $a > 0$ , eine Standardabweichung  $\sigma > 0$  und optional einen Seed zur Steuerung der Pseudozufallsquelle. Es gilt  $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ ,  $\tilde{\varepsilon} = a \cdot \varepsilon$ . Die Zufallsgenerierung basiert auf `numpy.default_rng` und erlaubt sowohl Determinismus als auch Nichtdeterminismus. Entsprechend VaMAI wird im Folgenden gezielt letzteres berücksichtigt.

**Einzelnes 3D-Signal EDL Unterapproximation zu annähernder Approximation und Störfunktion** Grundlegend ergibt die Auswertung für das einzelne Guass-Signal *mit* Störfunktion sehr ähnliche Dynamik im Ergebnis des ENN wie im ursprünglichen Abschnitt. Eine Unterapproximation ergibt ebenfalls eine zu flache Unsicherheitsmasse mit überkonfidentem Zuwachs zur Domänengrenze. Eine zunehmende Störfunktions-Parametrisierung mit Amplitude von 0,01 zu 1, Standardabweichung konstant 1, ergibt zwei Effekte. Das Abflachen der vom ENN erfassten Gesamtunsicherheitsmasse mit der Tendenz zu einer guten Approximation-Dynamik. Zweitens das Verrauschen der Gesamtunsicherheitsmasse. Das Verrauschen mit Amplitude 0,2 äußert sich in fleckweisen noch flacheren Volumenabschnitten, um die größte resultierende Gesamtunsicherheitsmasse in einem Abschnitt hoher Sample-Dichte. Interpretiert man weitere Lösungsansätze für dieses Verhalten, erlaubt die Differenz der Ergebnis-Massen eine eindeutige Filterung. Dabei kann erwartet werden, dass kleine Amplituden der Störfunktion durch einen Hochpassfilter, größere Amplituden durch einen Tiefpass, sowie Kombinationen durch Bandpass auf die abgezielte Unsicherheitsmasse gefiltert werden kann. In komplexeren Fällen der Überlagerung wäre ein Ansatz wieder eine ortsweise Abtastung der Signale und additive Kombination der Lösung.

**Einzelnes 3D-Signal EDL Überapproximation mit Störfunktion** Mit Störfunktion kann *nicht* die selbe Überapproximations Dynamik wie im vorherigen Abschnitt zum einzelnen Gauss-Signal erzielt werden. Es verzögert sich *nicht* lediglich durch die Epochenzahl. Bereits eine Amplitude von 0,2 sorgt bei im vorherigen Abschnitt verwendeter Epochenzahl von viertausend für gleich große um die wahre Unsicherheitsmasse äquidistant verteilte kleine Unsicherheitsmassen, in dem Fall drei, welche die Unsicherheit des Netzes ausdrücken.

**Multi 3D-Signal EDL Unterapproximation zu annähernder Approximation und Überapproximation mit Störfunktion** Wie im vorherigen Abschnitt ohne Störfunktion verzerren sich die Einzel-Signal-Beobachtungen lediglich durch die vom Netz erfasste Gesamtunsicherheitsmasse mit abweichendem Volumen von der Erwartungshaltung einer hinreichend den

ursprünglichen Signalformen entsprechenden. Ebenso erhöht sich die Unsicherheit des Netzes wieder ausgedrückt, dadurch das die aus mehreren Signalformen zusammengesetzte Unsicherheitsmasse aufgeteilt wird, auf die vom Netz äquidistant angelegten Teilvolumina um den Punkt der höchsten Signaldichte. Dies entspricht nicht der von VaMAI geforderten Ähnlichkeit zu den ursprünglichen Signalformen - wenn auch hier nicht untersucht auf  $\gamma$ , sondern in Ausschlägen der aleatorischen, epistemischen Varianz. Außerdem erzielt es je nach Störfunktionsamplitude zu hohe oder niedrige Ergebnisse für die aleatorische Varianz.

**Zwischenfazit der Durchführung - Über- und Unterkonfidenz** Schwachstelle durchgeführter Experimente ist ein Über- und Unterschätzen von Grenzwerten in Abhängigkeit von der Hyperparameter-Wahl, sowie die nach oben ungebundene Loss-Funktion wie aufgezeigt im Paragraphen „Einzelnes 3D-Signal EDL Überapproximation“. Mittels Josangs SL wurden erweiterte Parametrisierungen aufgezeigt, welche die Behandlung des Unterschätzens bis zur Definition von „Zero Evidence Regions“ behandeln. Das Überschätzen ist hier, so wie in der Literatur im Fokus. Kritikalität in VaMAI fordert das Einhalten der gegebenen Signal-Grenzen unabdingbar ein. Ein normiertes Verhalten ließe die additive Beziehung aus der Eingangsgleichung zur Gesamtunsicherheit über den Raum quantifizierbar zu. Ulmer et al. zeigen dafür in (28) unter (26) auf Oh und Shins Ansatz auf, die „Negative Log Likelihood“ Loss-Funktion mit Regularisierung nach Aminis Regressionsansatz unter (27) um einen Dritt-Term zu Erweitern. Dazu wurden verantwortliche Parameter  $\nu$ ,  $\alpha$  in (26) von Oh und Shin explizit mittels partieller Differenziation gelöst, um mit (29) eine eindeutige analytische Obergrenze - unintuitiv - über eine Minimum-Funktion  $U_{\nu, \alpha} = \min(U_{\nu}, U_{\alpha})$  auf einem Mini-Batch aufzuzeigen (vgl. Ulmer et al. [7, S.17, Z. 20-25, S.18 Z. 1-8]). Die analytische Lösung von Oh und Shin lautet:

$$U_{\nu} = \frac{\beta(\nu + 1)}{\alpha \cdot \nu}, \quad U_{\alpha} = \frac{2 \cdot \beta \cdot (\nu + 1)}{\nu} \cdot \left[ \exp \left( \psi \left( \alpha + \frac{1}{2} \right) - \psi(\alpha) \right) - 1 \right]$$

$\alpha$ ,  $\beta$ ,  $\nu$  sind wie folgt abhängig.  $\beta$  gilt als frei definierbarer Skalenparameter mit in der Literatur häufig gewählter Belegung (vgl. Bishop 2006 „Student-t-Modelle“ [16], Gelman et al. 2013 „Models for robust inference“ [84]):

$$\beta = \begin{cases} 1 & \text{unabhängig,} \\ \beta_0 \cdot (\alpha - 1) & \text{abhängig von } \alpha \end{cases}$$

$\nu$  und  $\alpha$  definieren die Loss-Begrenzung über die Fehlerwerte der Loss-Funktion mit einer lipschitz-stetigen Übergangsbedingung von quadratischer zu linearer Komponente wie im Anhang gezeigt.

$$\mathcal{L}_{\text{MSE}} = \begin{cases} (y_i - \gamma)^2 & \text{wenn } (y_i - \gamma)^2 \leq U_{\nu, \alpha}, \\ 2 \cdot \sqrt{U_{\nu, \alpha}} \cdot |y_i - \gamma| & \text{wenn } (y_i - \gamma)^2 \geq U_{\nu, \alpha} \end{cases} \quad (8.4)$$

$$\begin{aligned} \text{Übergangsbedingung:} & \quad (y_i - \gamma)^2 = 2\sqrt{U_{\nu, \alpha}} \cdot |y_i - \gamma| - U_{\nu, \alpha} \\ \text{Lösung:} & \quad |y_i - \gamma| = \sqrt{U_{\nu, \alpha}} \end{aligned}$$

Die Min-Funktion begrenzt den quadratischen Anteil der Steigung der Loss-Funktion, um genau den Teil, in dem er stärker steigt, als die lineare Funktion.

Herr Autenrieths Kritikalitätsraum ist mit einem oberen und unteren Schwellenwert für die Kritikalität behaftet, in den Beispielen die zwei horizontalen Ebenen.

$$C: \mathcal{X} \rightarrow [0.4, 0.6]^3 \subset [0, 10] \times [0, 10] \times [0, 1] \subset \mathbb{R}^3$$

$C$  soll die Loss-Regularisierung eingrenzen. Als Erweiterung von Oh und Shin lässt sich eine zweite kritikalitätsabhängige Unsicherheitsgrenze einführen durch:

$$U^*(x) := \min(U_\nu, U_\alpha, U_0 \cdot (1 - C(x)))$$

Dies erzwingt einen strengeren Übergang für die prädiktive Unsicherheit in hochkritischen Regionen des Eingaberaums, da erstens größere Fehler stärker bestraft werden. Zweitens, die Unsicherheit nicht skaliert werden kann, um den Fehler zu kompensieren. Drittens, epistemische und aleatorische Unsicherheit konservativer geschätzt werden, um den Fehler niedrig zu halten. Umgangssprachlich ließe sich das so zusammenfassen: Das Modell darf zwar seine Unsicherheit erhöhen. Aber nur, wenn der Punkt nicht kritisch ist. In sicherheitsrelevanten Regionen muss das Modell sich konservativer festlegen. Das ist ein Ansatz zur Lösung der evidentiellen Überkonfidenz, die bereits im Stand der Technik im Vergleich von BNN, ENN, CP aufgezeigt wurde. Damit kann der Ansatz gegenüber CP wettbewerbsfähiger gestaltet werden. Ein zweiter Ansatz zur expliziten Kontrolle der prädiktiven Unsicherheiten ist die Verlustfunktion um einen Schranken-Term zu ergänzen, der Varianzen außerhalb des tolerierten Kritikalitäts-Intervalls  $[0.4, 0.6]$  sanktioniert. Wiedeholt zu (5.1) ist dabei

$$\text{Var}_{\text{gesamt}}(x) = \underbrace{\frac{\beta}{\alpha - 1}}_{\text{aleatorisch}} + \underbrace{\frac{\beta^2}{(\alpha - 1)^2(\alpha - 2)}}_{\text{epistemisch}}$$

die Gesamtvarianz des evidentiellen Modells. Der Schranken-Term lautet:

$$\mathcal{L}_{\text{bound}(w)} = \lambda \cdot (\max(0, \text{Var}_{\text{gesamt}}(x) - 0,6)^2 + \max(0, 0,4 - \text{Var}_{\text{gesamt}}(x))^2) \quad (8.5)$$

$$\text{mit } \text{Var}_{\text{gesamt}}(x) = \begin{cases} \frac{\beta}{\alpha-1} + \frac{\beta^2}{(\alpha-1)^2(\alpha-2)}, & \text{konvergent für } \alpha > 2 \\ \left( \frac{\beta}{\alpha-1} + \frac{\beta^2}{(\alpha-1)^2(\alpha-2)} \right) \rightarrow \infty, & \text{divergent für } \alpha \leq 2 \end{cases}$$

(8.2) wird unter Hinzunahme von (8.3) als (8) und (9) zusammengefasst in (10) aus Aminis Evidential Regression, mit (8.4) zur Gesamtverlustfunktion (8.5) addiert:

$$\mathcal{L}_i(w) = \mathcal{L}_{\text{NLL},i}(w) + \lambda \cdot \mathcal{L}_{\text{R},i}(w) \quad (8.6)$$

$$\mathcal{L}_{\text{clip}}(w) = \begin{cases} (y - \gamma)^2, & \text{falls } (y - \gamma)^2 \leq U^*(x) \\ 2\sqrt{U^*(x)} \cdot |y - \gamma| - U^*(x), & \text{sonst} \end{cases} \quad (8.7)$$

$$\mathcal{L}_{\text{total}}(w) = \mathcal{L}_{\text{NLL}}(w) + \lambda_{\text{clip}} \cdot \mathcal{L}_{\text{clip}}(w) + \lambda_{\text{bound}} \cdot \mathcal{L}_{\text{bound}}(x) \quad (8.8)$$

Dies ist ein Ansatz sowohl die Kontrolle über die Überkonfidenz im Trainingsverhalten zu regulieren, als auch die finalen Varianzwerte an die Grenzen der Kritikalitätsfunktion zu binden. Dies soll jetzt noch einmal unter selber Konfiguration an „Einzelnes 3D-Signal EDL Überapproximation“ aufgezeigt werden.

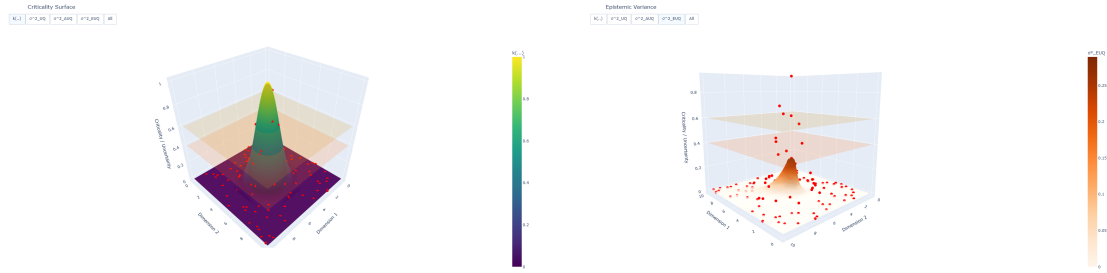
**Behandlung Überkonfidenz - „Einzelnes 3D-Signal EDL in Obergrenze geklippte Approximation“** Mittels selber Konfiguration wie im „Einzelnes 3D-Signal EDL Überapproximation“ in Schicht-, Neuronen-, Epochenzahl und Anpassung

$$\mathcal{L}_{\text{total}}(w) = \mathcal{L}_{\text{NLL}}(w) + \lambda_{\text{clip}} \cdot \mathcal{L}_{\text{clip}}(w) + \lambda_{\text{bound}} \cdot \mathcal{L}_{\text{bound}}(w)$$

$$\text{mit } \lambda_{\text{clip}} = 8.0, \quad \lambda_{\text{bound}} = 20.0, \quad \beta_0 = 1.0, \quad \nu = 3.0, \quad \text{Regularisierung } \lambda = 10^{-3},$$

benannt als Konfiguration 5, ist das epistemische Signal im Kritikalitätsbereich geklippt worden. Variiert wurden die Epochenzahl zwischen 2000 und 4000 mit dem besten Ergebnis in Tendenz zu 2000. Erste Ergebnisse unter  $\lambda_{\text{clip}} = 2$ ,  $\lambda_{\text{bound}} = 5$  fielen zu flach aus.  $\lambda_{\text{clip}}$ ,  $\lambda_{\text{bound}}$  wirken Ausgabe-amplifizierend auf AU, EU bei Verdopplung der Parameter im selben Verhältnis.  $\lambda_{\text{clip}} = [8, 16]$ ,  $\lambda_{\text{bound}} = [20, 40]$  mitteln das geklippte Signal in Herr Autentrieths Kritikalitätsgrenzen als bestes Ergebnis für die Amplitude. Niedrigere Parametrisierungen liefern die beste

Nachbildung der Position des Gauss-Signals mit zu flachem Volumen. Ein weiterer sensibler Faktor ist die Epochenzahl zwischen 2000 und 2500. Bei 2500 Epochen gilt: Die Krümmung des Gauss für das obere Drittel wird noch besser nachgebildet, doch die Gauss-Nachbildung geht im unteren Teil bereits wieder in eine Unterapproximations-Dynamik über. Bei 2000 Epochen ist der Ausblick auf den Kompromiss Volumen, Position am besten wie in der Abbildung zu sehen. Verfehlt jedoch leider noch die gewählten Kritikalitätsgrenzen, um 0.1 Kritikalität aufgrund einer ungetesteten Parameter-Kombination. Weiteres Experimentier-Budget erscheint hier sinnvoll, um den Kompromiss zwischen Volumen und Position zu lösen. Über alle Versuche mit geklippter Loss-Funktion ist die Oberfläche im oberen Drittel einheitlich zur Sample-Oberfläche nachgebildet worden, ohne Ausreißer. Die Loss-Erweiterung erfüllt damit ihren Zweck. Unsicherheit des Netzes wurde in gleichen Dynamiken wie in vorherigen Versuchen ausgedrückt in wellenförmig abflachender Konvergenz zum Domänenrand oder äquidistanten Massen um den Punkt größter Sampleichte.



Konfiguration 5, Original-Signal als Wert der Kritikalitätsfunktion  $k(\cdot)$  als GT

Konfiguration 5, vom ENN erkannte EU

Abbildung 8.19.: Vergleich von GT und EU unter Konfiguration 5

**Behandlung Unterkonfidenz und Nichtwissen** Es wird eine *Zero Evidence Region* (ZER) definiert als Bereich im Eingaberaum, in dem die EU des Modells unterhalb eines minimalen Schwellenwertes liegt. Basierend auf Josangs SL ergibt sich die Unsicherheitsmasse für kontinuierliche Modelle analog zur multivariaten Meinung als:

$$u(x) := \frac{1}{\alpha(x)} \quad \text{mit Evidenz } E = \alpha(x) - 1$$

Es soll eine untere Evidenzschranke  $\alpha_{\min} > 0$  definiert werden über  $\alpha$ , sowie alternativ über die epistemische Varianz  $\text{Var}_{\text{epist}}$ :

$$\text{ZER}(x) := \begin{cases} 1, & \text{falls } \alpha(x) < \alpha_{\min} \\ 0, & \text{sonst} \end{cases} \quad \text{ZER}(x) := \begin{cases} 1, & \text{falls } \text{Var}_{\text{epist}}(x) > v_{\max} \\ 0, & \text{sonst} \end{cases}$$

Diese Regionen signalisieren dem System explizit, dass eine verlässliche Aussage nicht getroffen werden kann. Das Modell kann *grundlegend* Nichtwissen aufzeigen.

Dies kann in die Loss-Regularisierung aufgenommen werden als:

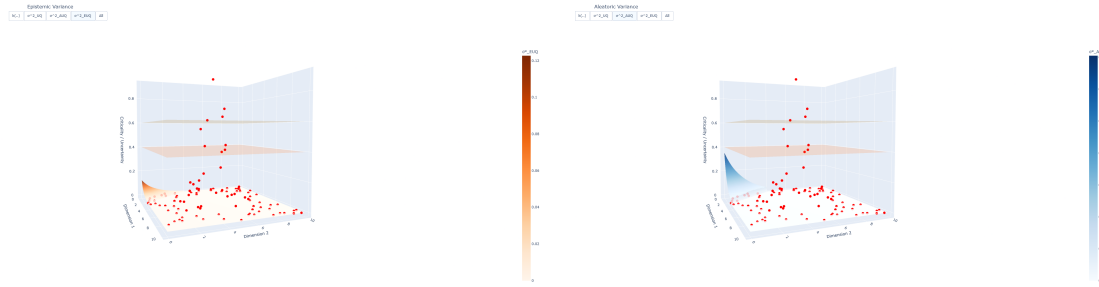
$$\mathcal{L}_{\text{total}}(w) = (1 - \text{ZER}(x)) \cdot [\mathcal{L}_{\text{NLL}}(w) + \lambda_{\text{clip}} \cdot \mathcal{L}_{\text{clip}}(w) + \lambda_{\text{bound}} \cdot \mathcal{L}_{\text{bound}}(x)]$$

$$\text{mit } \lambda_{\text{clip}} = 4.0, \quad \lambda_{\text{bound}} = 20.0, \quad \lambda = 10^{-3}, \quad U_0 = 2.0, \quad v = 4.0$$

Dies soll jetzt noch einmal ausgehend von Konfiguration 5 an „Einzelnes 3D-Signal EDL Unterapproximation“ aufgezeigt werden. Die Evidenz und „Zero-Evidence“ kann über die diversen Parametrisierungen, die Josang liefert, spezialisiert werden.

Wählt man  $\lambda_{\text{clip}}$ ,  $\lambda_{\text{bound}}$  äquivalent zum letzten Versuch „Einzelnes 3D-Signal EDL in Obergrenze geklippte Approximation“ steuert man die Konvergenz des Modells über noch vorhandenen Eingabedaten, die nicht vom ZER-Operator getroffen werden. Trainiert man das Netz über die bisher etablierten Konfiguration von 5 Läufen, 2000 Epochen, zwei vollverbunden Schichten, 128 Neuronen, maximal  $\lambda_{\text{clip}} = 8$ , maximal  $\lambda_{\text{bound}} = 40$  divergiert der Loss ab dem fünften Lauf überdurchschnittlich häufig zu keinem Ausgabewert *nan*. Reduziert man als Konfiguration 6 die  $\lambda$ -Parameter von dort an, erhält man über fünf Läufe Training eine Durchschnittsausgabe keiner (Wert „null“) AU, EU über den gesamten Raum. Eine zu den vorherigen Approximationen vergleichbare Lösung, die dem Original-Signal nachkommt, liegt ab 1000 Episoden aufwärts im Zwischenbereich erwähnter Konfigurationen. Höheres  $U_0$ ,  $v$  fängt an die Approximationsdynamik des ENN zu verbessern. Der grundlegende Mechanismus zur Etablierung eines unteren Schwellenwerts, der alle darunterliegenden Evidenzen nahe Null zu Null schätzt, funktioniert. Damit ist die grundlegende Funktion aufgezeigt wie in der streng abfallenden Steigung der Varianzen im Übergang zu null gegenüber dem Gesamttraum zu sehen. Es wird an dieser Stelle keine weitere Optimierung angestrebt.

**Meta-Unsicherheit - Vergleich BNN und ENN nach Josang** Zum Schluss soll Herr Mavanis Ansatz für BNN mit dem der ENN nach Ansatz von Josangs AOPS, EOPS für vergleichende Meta-Unsicherheit beispielhaft in Herr Autenrieths Kritikalitätsraum implementiert werden.



Konfiguration 6, vom ENN erkannte AU

Konfiguration 6, vom ENN erkannte EU

Abbildung 8.20.: Vergleich von GT, AU und EU unter Konfiguration 6

**Vorbereitung** Herr Mavanis BNN Ansätze in Jax wurden zur Bearbeitung dieser Thesis verworfen, weil sie im Zeitrahmen zum Ende meiner Projektphase zum 24.07. zu ungenügenden Ergebnissen führten. Der HMC-Ansatz wurde nach Umstellung für diesen Zweck ebenfalls verworfen, wegen überstiegenem Budget in Trainingszeit. Es wurde in Herr Mavanis Repository unter Branch des Autoren ein Pyro Äquivalent implementiert sowie variiert mit drei, vier und fünf vollverbundenen Schichten, 32, 64, 128 Neuronen, 2000 bis im Schnitt 5000 Epochen inklusive Aufwärmprozess, 500 bis 2000 SVI Posterior Samples, Invertierung des Mittelwerts als Ausgabeneuron, Variation des BNN Modell Rangs, Unter- und Übergewichtung der Ausgabe-neuronen, Übersampling der Signalspitze, Miterfassung aleatorischer Varianz, Verrauschen der epistemischen Varianz, Änderung der Aktivierungsfunktionen mittels ReLu, SiLu sowie zusätzlicher Lern-Raten-Abfall. Das beste Ergebnis wurde mit SVI, zwei vollverbundenen versteckten Schichten, 64 Neuronen, 2000 Epsioden und ohne Lern-Raten-Abfall erzielt. Das Gauss-Signal konnte auf dem Boden in der Position annähernd korrekt positioniert werden. Bei korrekter Position lieferte jede Parameter-Änderung zu sensibel-varrierende Ergebnisse in Bezug auf das Volumen, ohne auffällige Konvergenzen. In Bezug auf das Volumen konnte ein zum ENN vergleichbarer Unterapproximation mit einem vergleichbar aus der Signalform austretendem wellenartigen Volumen erzielt werden. Wurde das Volumen annähernd korrekt geschätzt, reagierte die Position zu sensibel auf Parameter-Änderungen. Hauptergebnis der Experimentierphase ist das zu stark mittlende Verhalten des BNN über erwähnte Parameter-Variation in verwendeten Ansätzen. So wird bei großer Samplezahl am gewählten Beispiel aus Herr Autenriehts Kritikalitätsraum regelmäßig eine zur Gauss-Steigung invertierte konkave Ebene aufgespannt, wichtige gesamplete Spitzenwerte ignoriert. Die Intention die Ausgabe zu Invertieren war nicht zielführend. Zusammenfassend wurde in keinem Fall ein relevantes vergleichbares Ergebnis auf dem einfachen Gauss-Beispiel erzielt. Deshalb wird ein synthetisches Ergebnis des BNN generiert, um den Vorschlag für die Methodik aufzuzeigen und abzuschließen.

**Meta-Unsicherheitsvergleich zwischen ENN und synthetischem BNN** Zur Bewertung der strukturellen Unterschiede in den Unsicherheitsannahmen zwischen einem ENN und einem BNN soll ein Divergenzmaß über die geschätzten Unsicherheitsverteilungen eingeführt werden. Da kein hinreichendes BNN-Modell vorliegt, wird die EU und AU des BNN synthetisch approximiert, indem ein Rauschterm  $\delta(x) \sim \mathcal{N}(0, \sigma^2)$  zur epistemischen ENN-Varianz addiert und anschließend in das Intervall  $[0.4, 0.6]$  projiziert wird.

Für jedes  $x \in \mathcal{X}$  wird berechnet:

$$\begin{aligned}\text{Var}_{\text{gesamt}}^{\text{ENN}}(x) &= \text{Var}_{\text{epist}}^{\text{ENN}}(x) + \text{Var}_{\text{alea}}^{\text{ENN}}(x) \\ \text{Var}_{\text{gesamt}}^{\text{BNN}}(x) &= \text{Var}_{\text{epist}}^{\text{ENN}}(x) + \text{Var}_{\text{alea}}^{\text{ENN}}(x) + \delta(x)\end{aligned}$$

Die Meta-Unsicherheit wird anschließend durch die Jensen-Shannon-Divergenz (JSD) zwischen den beiden Gesamtunsicherheitsverteilungen definiert:

$$\text{Meta-Unsicherheit} := \text{JSD} \left( \mathcal{D}_{\text{gesamt}}^{\text{ENN}} \parallel \mathcal{D}_{\text{gesamt}}^{\text{BNN}} \right)$$

Je höher dieser Wert, desto größer ist die Divergenz in der modellierten Unsicherheitsstruktur und damit die meta-informative Abweichung zwischen den Netzarchitekturen.

Das definieren eines Ansatzes zur Meta-Unsicherheit erlaubt ebenso einen Ansatz zur Bewertung durch ein Framework.

## Framework zur vergleichenden Unsicherheitsanalyse der Meta-Unsicherheit

**1. Eingabe und Setup** Zur Analyse werden zwei Modelle betrachtet: ein ENN sowie ein BNN, dessen Unsicherheitsverhalten synthetisch simuliert wird, wie vorher definiert. Die Berechnung erfolgt auf einem gemeinsamen Testdatensatz  $\mathcal{X}_{\text{test}}$  am bisher verwendeten Beispiel des Gauss-Signals in der Implementierung von Herr Autenrieths Kritikalitätsraum.

### 2. Kernmetriken je Modell

- **Epistemische Varianz:**  $\text{Var}_{\text{epist}}$
- **Aleatorische Varianz:**  $\text{Var}_{\text{alea}}$
- **Gesamtvarianz:**  $\text{Var}_{\text{gesamt}} = \text{Var}_{\text{epist}} + \text{Var}_{\text{alea}}$
- **Accuracy:** Anteil korrekter Vorhersagen auf  $\mathcal{X}_{\text{test}}$

**3. Erweiterte Unsicherheitsattribute** Erweiterte Unsicherheitsattribute sind all diejenigen Bewertungsmetriken, die in der entwickelten Software Metrik-Registrierung enthalten und aus der vorherigen Recherche abgeleitet sind.

**4. Bewertung und tabellarischer Vergleich** Abschließend für das Framework zur vergleichenden Unsicherheitsanalyse der Meta-Unsicherheit soll ein beispielhafter Vergleich auf einem synthetischen Ergebnis durchgeführt werden.

| Metrik                               | ENN     | BNN (synthetisch) | Differenz |
|--------------------------------------|---------|-------------------|-----------|
| Epistemische Varianz (Mittel)        | 0.0439  | 0.4000            | +0.3561   |
| Aleatorische Varianz (Mittel)        | 0.0407  | 0.4000            | +0.3593   |
| Gesamtvarianz (Mittel)               | 0.0846  | 0.8000            | +0.7154   |
| JSD der Gesamtvarianz                | –       | –                 | 0.4405    |
| Accuracy                             | 95.72 % | 95.71 %           | -0.01 %   |
| Kritikalitätsgewichtete Unsicherheit | 0.0167  | 0.0491            | +0.0324   |

Tabelle 8.7.: Vergleich ausgewählter Unsicherheitsmetriken zwischen ENN und synthetisch approximiertem BNN

**5. Interpretation der Meta-Unsicherheit** Nimmt man eine JSD von null als Gleichheit der resultierenden Verteilungen der Gesamtunsicherheit in der Gesamtvarianz an, so ist ein Betrag von 0.4405 vergleichsweise hoch. EU, AU im synthetischen BNN im Vergleich zum ENN könnten die strukturelle Abweichung der Unsicherheitsmodellierung aus allen bisherigen Ergebnissen der Thesis, und verknüpfter Abschlussarbeiten bestätigen. Dies würde eine vergleichende Implementierung in der Datenpipeline mit abgeschlossener BNN-Training bedingen. Nimmt man diese Situation an, könnte man sie wie folgt auswerten, um die Ambivalenzen in den Beziehungen von UQ aufzuzeigen und durch Erweiterung sinnvoll gegenüberzustellen. Trotz nahezu identischer Accuracy (95.72 % vs. 95.71 %) verdeutlichen die Differenzen in den Varianzmetriken und der kritikalitätsgewichteten Unsicherheit eine deutlich unterschiedliche Vertrauensstruktur in den Vorhersagen beider Modelle. Grundlegend wird damit aufgezeigt, dass eine Kalibrierung über verschiedene Unsicherheitsattribute so möglich, und zum Vergleich im Verbund sinnvoll ist. Die Aussagekraft der *Meta-Unsicherheit* als Analysewerkzeug für Unsicherheitsannahmen hängt dabei unmittelbar vom Design der Experimente in machine-learning-basierten Surrogatmodellen ab. Weitere Attribute wie in der Datenpipeline aufgezeigt können genutzt werden, um die Unsicherheitsmodellierung zu spezialisieren.

## 9. Auswertung

**R1** Über die Ausarbeitung der Thesis wurden Bayesische Neuronale Netze, Evidentielle Neuronale Netze, Conformal Prediction mit einer minimalen Sensitivitätsanalyse als Surrogatmodell verglichen. Bayesische Neuronale Netze haben in der Ausarbeitung ein positives Verhalten gegenüber Approximationen von konvexen oder konkaven Ebenen über eine globale Raumsituation in meinen Konfigurationen gezeigt. Zusätzlich haben sie stark streuendes Rauschen sehr viel besser als der evidentielle Ansatz nachgebildet. Evidentielle Neuronale Netze haben jedoch zusätzlich spitze schwach gesamplete Amplituden aus einem zu approximierenden Grundsignal wie einer die Sampling-Ebene durchbrechenden Gauss-Glocke berücksichtigt. Bayesische Neuronale Netze haben diese unter meinen Architektur-Variationen basierend auf Herr Mavanis SVI Ansatz ignoriert. Evidentielle Neuronale Netze sind aus der Recherche als Überkonfident bezeichnet worden. In der Thesis-Ausarbeitung wurde durch die Loss-Erweiterung ein Ansatz erarbeitet das verwendete Evidentielle Neuronale Netz wettbewerbsfähiger gegenüber der konservativen doch schnelleren Schätzung einer Conformal Prediction zu machen. Abschließend hat sich im Projekt VaMAI bis hierin kein Favorit für alle Projektteilnehmer herausgestellt.

**R2** Zur Beantwortung welche allgemeinen und modellspezifischen Faktoren das Lernen von Unsicherheiten in ML-Modellen beeinflussen, wurde ein Fragenkatalog erarbeitet, recherchiert, belegt und in Tabellenform wiedergegeben, die als Referenz für VaMAI Teilnehmer dienen sollen.

**R3** Um UQ zu verbessern wurde an einer Auswahl von aus R2 gegebenen Einflussfaktoren allgemeine Stichpunkte an Quellen belegt wiedergegeben.

**R4** Inwieweit ML-basierte UQ zuverlässig den in realen Anwendungsszenarien beobachtbaren Unsicherheiten entsprechen kann ist grundlegend beantwortet worden. Eine präzisere Antwort ist mit der Wahl der Realdaten verbunden.

**A1** Es wurden mittels In- und Exklusionskriterien für eine Recherche, eine Eingrenzung des Umfangs als „Scope“ erarbeitet und entgegen einem eigens aufgestellten Fragenkatalog erarbeitet sowie wiedergegeben.

**A2** Es wurden grundsätzliche Unsicherheitsmetriken zum Einarbeiten in den Sachverhalt aufgezeigt. Alle Metriken waren in einer Form zur Projektausarbeitung direkt oder indirekt zum Vergleich oder Verständnis relevant.

**A3** Anwendungsfelder aus dem Ingenieurwesen wurden aufgezeigt. Darauf folgend wurden modellspezifische und anwendungsspezifische Metriken wiedergegeben.

**A4** Es wurden Aspekte von Bayesischen sowie zusätzlich Evidentiellen Neuronalen Netzen wiedergegeben und mit Conformal Prediction verglichen.

**A5** Der theoretische Einfluss einzelner Größen wie aleatorischer und epistemischer Varianz wurde an den Beispielen aus A4 vertieft und gegenübergestellt als Grundlage für A6. Hierbei wurde insbesondere die in der Recherche häufig bemangelte Ausgestaltung einer expliziten Logik mittels Josangs „Subjective Logic“ Ausarbeitung für *BNN* unterlegt.

**A6** In der Durchführung der Implementierung und Validierung der entwickelten Unsicherheitsmetriken und Modellierungstechniken in simulierten Daten hat sich folgendes ergeben. Bei der Bewertung der Parameter verstärken sich Konvergenzeffekte von Trainings- und Validierungsloss über die Samplezahl  $n$  sichtbar an den Beispielen der einzelnen 3D-Signale Gauss, Hyperwürfel in Under- und Overfit. Dies wird linear beobachtet unterstützt von der Epochenzahl.

In der Under- und Overfit Bewertung verwendeter 3D-Signale variiert die epistemische und aleatorische Varianz insgesamt kaum ersichtlich im Verhältnis im Raum zur Parametervariation. Die  $\gamma$ -Ergebnisse fallen dafür sehr viel differenzierter aus.

In der Approximations-Bewertung ist dies umgekehrt. In dieser wird über die 3D-Signale deutlich, dass sich Volumen und Position der Unsicherheiten der einzelnen 3D-Signale Gauss, Hyperwürfel verschieden stark im Volumen amplifizieren lassen. Es fehlt jedoch eine Normierung des Ausschlags, um aus der quantitativen Referenz eine qualitative Aussage über die Amplitude der epistemischen, aleatorischen Varianz zu treffen. Es wurde gezeigt, dass die Amplitude steuerbar ist. Es wurde gezeigt, dass die Steuerung der Amplitude in Kombination mit Erweiterung der von Amini vorgeschlagenen Loss-Funktion Ergebnisse liefern kann, die Effekte in recherchierten Papern grundlegend eindämmen.

## 10. Diskussion

**Diskussion R1** Die Resultate weisen darauf hin, dass Bayesische Neuronale Netze bei bestimmten Signalprofilen und unter stark streuendem Rauschen präzisere Approximationen erzielen können. Evidentielle Neuronale Netze ergänzen diese Eigenschaft durch eine höhere Sensitivität für selten gesampelte, ausgeprägte Signalamplituden. Die Erweiterung der Loss-Funktion führte zu einer messbaren Verbesserung gegenüber der Standardvariante, ohne jedoch eine eindeutige Überlegenheit gegenüber Conformal Prediction zu etablieren.

**R2** Der entwickelte Fragenkatalog bietet eine strukturierte Übersicht relevanter Einflussgrößen auf das Erlernen von Unsicherheiten. Durch die systematische Anordnung lässt er sich sowohl als Referenz innerhalb des Projekts als auch für methodenübergreifende Vergleiche einsetzen.

**R3** Die gezielte Betrachtung ausgewählter Faktoren aus R2 verdeutlicht, dass Änderungen im Modellaufbau und bei den Trainingsparametern die Qualität der Unsicherheitsquantifizierung deutlich beeinflussen können. Die hinterlegten Quellenbezüge belegen diese Einschätzung und liefern Anhaltspunkte für eine Umsetzung.

**R4** Die Frage nach der Verlässlichkeit ML-basierter Unsicherheitsquantifizierung in realen Anwendungskontexten konnte in allgemeiner Form beantwortet werden. Für eine präzisere Einschätzung wären jedoch domänenspezifische Realdaten erforderlich, um methodische Grenzen und Stärken unter praxisnahen Bedingungen zu überprüfen.

**A1** Die Eingrenzung des Themenfeldes durch In- und Exklusionskriterien führte zu einer zielgerichteten Recherche. So wurde vermieden, dass die Recherche sich davon thematisch entfernt.

**A2** Die Darstellung grundlegender Unsicherheitsmetriken trug zu einem einheitlichen Verständnis für mich als Projektteilnehmer bei. Dieses Fundament erleichterte mir die methodische Gegenüberstellung und die Interpretation der Ergebnisse. Dies sollte dementsprechend auch für andere Projektteilnehmer gelten.

**A3** Die aufgeführten Anwendungsbeispiele aus dem Ingenieurwesen verdeutlichen den möglichen Transfer in der Bedeutung der vorgestellten Metriken. Die Kombination aus modell- und anwendungsspezifischen Aspekten schafft eine Grundlage für eine Auswahl von Ansätzen im Ingenieurwesen.

**A4** Der Vergleich zwischen Bayesischen und Evidentiellen Neuronalen Netzen sowie Conformal

Prediction machte deutlich, dass die Methoden je nach Szenario unterschiedliche Vorteile bieten. Diese Erkenntnis kann als Entscheidungshilfe bei der Modellauswahl dienen.

**A5** Josangs „Subjective Logic“ wird in meinen Augen dem in der Literatur angemerktem Robustheitsanspruch für sicherheitskritische Systeme gerecht, da eine eindeutige Logik-Hierarchie vorgegeben wird. Diese wurde an relevant empfundenen Stellen aufgezeigt. Sie würden in einer weiterführenden Arbeit den Transfer auf Evidentielle Neuronale Netzwerke im Hinblick auf eine Meta-Unsicherheit erlauben, sollte dieser weiter im Projekt verfolgt werden.

**A6** Die Experimente zeigten, dass sich mit zunehmender Stichprobengröße und höherer Epochenzahl deutliche Konvergenzeffekte zwischen Trainings- und Validierungsloss einstellen. Die aleatorische Varianz entsprach in den Untersuchungen linear dem Eingangsverhalten, sofern der ganze Raum untersucht wurde ohne eingeführten ZER-Operator. Die epistemische Varianz stellte sich in den Untersuchungen überwiegend als höher, als die aleatorische Varianz heraus. Zudem wurde deutlich, dass sich die Amplituden der Unsicherheiten gezielt an Bereiche anpassen lassen, was für weiterführende Studien eine relevante Stellgröße darstellen kann. Ein Über- und Unterfit wurden in  $\gamma$  gegenübergestellt und verbundene aleatorische, epistemische Varianz aufgezeigt. Mit der erarbeiteten Trennung von Approximation und Fit wäre es angebracht, den Trainings- und Validierungsloss von VaMAIs Ziel niedriger Samplezahlen aufzulockern, um mit extremeren Loss-Konvergenzen über Trainings- und Validierungsdaten Ergebnisse zu erzielen, die einen Fit vom Ausgangssignal  $y$  gegenüber  $\gamma$  darstellen. Dieses Verständnis kann im Verbund mit weiteren Aspekten zur vollständigen Fit-Definition für evidentielle Netzwerke in VaMAI beitragen. Zur Ausarbeitung einer normativen Skala für epistemische und aleatorische Varianz, die über Linearitätsbeobachtungen hinausgeht, wäre es in meinen Augen sinnvoll, diese mit der Raumabdeckung aus in Beziehung zu setzen. Ebenso wären bei der Ausarbeitung der Plots für Vorhersagen zu Stichproben ein weiterer Schritt das Einführen von Quantilen, zum expliziteren erfassen der Abdeckung der Ground-Truth. Dies wäre in Relation zu setzen zu den Amplituden-Ausschlägen der aleatorischen und epistemischen Varianz. Beobachtet wurden sowohl Änderungen in Position als auch im Volumen für epistemische, als auch aleatorische Varianz. Hier wäre eine normierende Methode für eine weitere Ausarbeitung sinnvoll. Dies würde es erlauben die Gesamtunsicherheit über den Kritikalitätsraum gemäß der Eingangsformel der Gesamtunsicherheit additiv und normiert zusammenzusetzen und im Idealfall eindeutig an der Ground-Truth zu belegen.

**Bewertung der Modellierungsansätze** Die Kombination aus evidenzbasierter Verlustfunktion und kritikalitätsabhängiger Begrenzung der Unsicherheit durch  $U^*(x)$  bietet einen stabilen Ansatz zur Kontrolle von Überkonfidenz in sensitiven Regionen. Die eingeführte Lipschitz-Begrenzung durch die min-Funktion verhindert stark ansteigende Gradienten in extremen

Fehlersituationen. Allerdings kann dies auch dazu führen, dass in niedrig-kritischen Bereichen das Modell zu konservativ wird.

Die Regularisierungsparameter  $\nu$ ,  $\alpha$  und  $\beta$  sind dabei sensitiv in ihrer Wirkung. Insbesondere die Wahl von  $\beta$  beeinflusst das Verhalten der Gesamtvarianz und sollte für reale Anwendungen datenspezifisch validiert werden.

**Unsicherheitsquantifizierung im Kontext realer Systeme** Die Trennung von AU und EU erfolgt hier auf Basis der evidenzbasierten Wahrscheinlichkeitsmodellierung. Dies ist theoretisch konsistent mit der Unsicherheitsdefinition der GUM, jedoch praktisch schwer zu validieren, solange keine realen Messdaten mit bekannter Varianzstruktur vorliegen.

Die genutzten Kritikalitätsfunktionen sind synthetisch definiert. Somit bleibt offen, ob das trainierte Netz diese Trennung bei realen, verrauschten Daten gleichermaßen leisten kann. Eine Erweiterung um reale Sensorikdaten wäre hier ein nächster Schritt.

**Einfluss der Kritikalitätsfunktion und Regularisierung** Die Kritikalitätsfunktion  $C(x)$  wurde explizit modelliert, um Unsicherheiten in sensiblen Bereichen zu begrenzen. Diese Eingrenzung erweist sich als effektives Steuerungsinstrument im Trainingsprozess, stellt aber eine idealisierte Repräsentation realer Risiken dar. In praktischen Anwendungen müsste  $C(x)$  durch Expertenwissen ersetzt werden.

Ergänzend wurde eine ZER-Regelung über einen unteren Varianz-Schwellenwert eingeführt, um Regionen mit extrem geringer Evidenz explizit zu markieren. Dies erhöht die Erklärbarkeit des Modells, setzt aber voraus, dass die Parametrisierung der Evidenzverteilung sinnvoll gewählt wurde.

**Limitationen** Ein wesentlicher Nachteil der durchgeführten Studie ist die fehlende Validierung mit realen Anwendungsdaten. Weder das Verhalten der Kritikalitätsfunktion noch die Effektivität der Verlustbegrenzung wurden im produktiven Kontext getestet.

Die gesamte Arbeit basiert auf synthetisch erzeugten Datensätzen. Dies ermöglicht zwar kontrollierte Experimente, schränkt aber die Generalisierbarkeit auf reale Szenarien ein.

**Ausblick und Weiterentwicklungsmöglichkeiten** Ein möglicher nächster Schritt wäre die Übertragung der entwickelten Methoden auf reale Daten mit bekannter Unsicherheitsstruktur, zum Beispiel aus der Sensorik, Messtechnik oder dem Bereich autonomer Systeme. Auch eine Erweiterung des ENN um Rekurrenzmechanismen zum Beispiel LSTM zur Modellierung von Zeitdynamik wäre in begrenzter Abstraktion von Signalen denkbar.

Ein weiteres Forschungspotenzial liegt in der quantitativen Ausführung des Konzepts der Meta-Unsicherheit – also der Unsicherheit über die Unsicherheitsmaße selbst. Hier könnten widersprüchliche Unsicherheitsquantifizierungen gegeneinander verglichen werden, um eine eindeutige Aussage zu treffen.

Die Arbeit zeigt, dass eine evidenzbasierte Unsicherheitsquantifizierung mit kritikalitätsabhängiger Regularisierung ein vielversprechender Ansatz zur Risikominimierung in sicherheitsrelevanten Systemen sein kann.

## 11. Fazit

In diesem Abschnitt sollen zentrale Erkenntnisse der Arbeit zusammengefasst werden und in Perspektive gestellt werden für Weiterentwicklungen, Anwendungsfelder.

### 11.1. Zusammenfassung zentraler Erkenntnisse

Im Rahmen dieser Arbeit wurde ein evidenzbasiertes Verfahren zur Unsicherheitsquantifizierung in ML-basierten Surrogatmodellen methodisch angewandt und exemplarisch umgesetzt. Durch den Einsatz eines ENN konnten sowohl AU als auch EU direkt aus den Eingabedaten gelernt und getrennt interpretiert werden. Die modellierten Unsicherheiten erwiesen sich als differenzierbar, kalibrierbar und visuell interpretierbar im Kontext von Szenarien, die mit einer Kritikalitätsfunktion belegt sind.

Herr Autenrieths Kritikalitätsräume als zu untersuchender Eingaberaum sind ein statisches System auf Basis algebraischer Gleichungen zum Simulieren von Nichtlinearitäten belegt mit pseudo-dynamischer Störfunktion.

$$y(t) = f(u(t)) + \epsilon(t)$$

Der Kritikalitätsraum ist statisch, da der Ausgang ausschließlich vom aktuellen Eingang abhängt. Es enthält in der Thesis weder Zustandsgrößen noch Differenzialgleichungen, weist keinen Zeitverzug auf und besitzt kein Speicherverhalten. Die additive Störfunktion  $\epsilon(t)$  kann als pseudo-dynamisch interpretiert werden, da sie zeitveränderlich ist, jedoch selbst kein eigenes Systemverhalten mit innerer Dynamik modelliert.

Obwohl ein ENN formal ein statisches Feedforward-System ist, kann es zur Laufzeit die Effekte dynamischer Systeme approximieren, etwa durch seine nichtlineare Projektion in die Unsicherheitsparameter. Die während des Trainings gelernten Gewichte speichern dabei ein implizites Gedächtnis über die Struktur der Eingaben. Dieses speicherartige Verhalten betrifft jedoch nicht den Inferenzprozess selbst, der zustandslos erfolgt.

Das formal statische Verhalten beider Systeme spricht - ohne formalen inneren Zustand sowie Dynamik des Systems - gegen die klassische Belegung von Steuer-, Beobacht- und damit Regelbarkeit der Systeme aus der Regelungstechnik.

In der Arbeit wurden in Bezug auf die Beobachtbar- sowie Steuerbarkeit die folgende Fragen positiv beantwortet: Erstens, kann das ENN durch geeignete Parametrisierung so gesteuert werden, dass es die Struktur des Kritikalitätsraums (zum Beispiel Unsicherheitsniveaus oder Schwellen) korrekt annähert? Damit ist es funktional steuerbar im Parameterraum. Zweitens, kann ich aus den Ausgaben des ENN (zum Beispiel Varianzen, Evidenz, Kritikalität) auf Strukturen im zugrunde liegenden Kritikalitätsraum schließen? Somit ist es eine beobachtbare Approximation, obwohl das System an sich zustandslos ist.

Die Kombination von Unsicherheitsmetriken, Evidenzparametern und Maßzahlen ermöglicht aus dieser Thesis ermöglicht zudem die Entwicklung eines Meta-Unsicherheitsmaßes zur Bewertung der Vertrauenswürdigkeit von Szenarien. Der Vergleich mit BNN-Modellen zeigte, dass evidenzbasierte Ansätze nicht nur effizienter, sondern auch deutlich robuster gegenüber struktureller Unsicherheit agieren können. Insgesamt demonstriert die Arbeit das Potenzial evidenzieller Verfahren zur methodisch fundierten und anwendungsnahen Bewertung datengetriebener Modelle unter Unsicherheit.

## 11.2. Ausblick

**EDL-Meta-Modelle, Realzeit-Validierung, Transfer auf andere Systeme** Aufbauend auf den in dieser Arbeit entwickelten Methoden zur evidenzbasierten Unsicherheitsquantifizierung ergeben sich mehrere weiterführende Forschungsperspektiven. Ein naheliegender Schritt ist die Entwicklung sogenannter EDL-Meta-Modelle, die nicht nur punktweise Unsicherheiten schätzen, sondern übergeordnetes Verhalten aggregieren und daraus adaptive Steuerungs- oder Bewertungssignale ableiten. Solche Modelle könnten als unsicherheitsregelnde Komponente in größeren Systemarchitekturen fungieren und beispielsweise dynamisch zwischen Regelstrategien oder Szenarien umschalten. Ein weiterer vielversprechender Forschungspfad liegt in der Integration der EDL-Methodik in Realzeit-Validierungsumgebungen, insbesondere in Verbindung mit MPC oder szenariobasierten Testframeworks. Hierbei müssten nicht nur Vorhersagewerte, sondern auch die resultierenden Unsicherheiten in Echtzeit prozessiert und rückgeführt werden – etwa zur dynamischen Anpassung von Sicherheitsmargen oder zur intelligenten Szenarienselektion. Darüber hinaus eröffnet sich das Potenzial zur Übertragung der Methodik auf andere technische Domänen, in denen datengetriebene Modelle in sicherheitsrelevante Entscheidungsprozesse eingebunden sind. Dazu zählen unter anderem autonome Landfahrzeuge, Robotik, industrielle Prozessüberwachung oder die adaptive Steuerung cyber-physischer Systeme. Entscheidend für einen erfolgreichen Transfer ist dabei die kontextspezifische Anpassung der Merkmalsextraktion und die Auswahl geeigneter Evidenzstrukturen. Insgesamt bietet die Kombination aus transparent modellierter Unsicherheit, mathematisch fundierter

Aggregation und systemischer Interpretierbarkeit eine zukunftsweisende Grundlage für robuste, erklärbare und vertrauenswürdige KI-basierte Systeme auf dem diese Arbeit aufsetzen konnte, um es beispielhaft anzuwenden.

## Liste der Abbildungen

|       |                                                                                                                                     |    |
|-------|-------------------------------------------------------------------------------------------------------------------------------------|----|
| 7.1.  | R4.1 „Evidential Deep Learning“ nach Amini et al. 2020 . . . . .                                                                    | 21 |
| 7.2.  | R4.2 „Epistemic Learning“ nach Manchingal et al. 2025 . . . . .                                                                     | 22 |
| 7.3.  | R4.3 „Major approaches to uncertainty in AI“ nach Manchingal et al. 2025 . . .                                                      | 23 |
| 8.1.  | Ninand Mavani Visual Benchmark BNN HMC auf (8.1) . . . . .                                                                          | 45 |
| 8.2.  | Ninand Mavani Visual Benchmark BNN SVI auf (8.1) . . . . .                                                                          | 45 |
| 8.3.  | Marten Windler Visual Benchmark ENN auf (8.1) . . . . .                                                                             | 45 |
| 8.4.  | R4.4 Angen. Grund-Verteilung auf unbek. Daten für EDL-Modell n. Amini . .                                                           | 46 |
| 8.5.  | R Grndl. Ausgangs-UI Kritikalität, Varianz total, aleat., epist. . . . .                                                            | 51 |
| 8.6.  | Vergleich der evidentiellen Ausgaben für $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . .                             | 54 |
| 8.7.  | Vergleich des Trainings- und Validierungsloss für $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . .                    | 54 |
| 8.8.  | Vergleich Ground-Truth $y_{\text{true}}$ zu Vorhersage $\hat{y}$ mit $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . . | 54 |
| 8.9.  | Vergleich der evidentiellen Ausgaben für $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . .                             | 56 |
| 8.10. | Vergleich des Trainings- und Validierungsloss für $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . .                    | 56 |
| 8.11. | Vergleich Ground-Truth $y_{\text{true}}$ zu Vorhersage $\hat{y}$ mit $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . . | 56 |
| 8.12. | Vergleich der evidentiellen Ausgaben für $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . .                             | 58 |
| 8.13. | Vergleich des Trainings- und Validierungsloss für $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . .                    | 58 |
| 8.14. | Vergleich Ground-Truth $y_{\text{true}}$ zu Vorhersage $\hat{y}$ mit $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . . | 58 |
| 8.15. | Vergleich der evidentiellen Ausgaben für $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . .                             | 60 |
| 8.16. | Vergleich des Trainings- und Validierungsloss für $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . .                    | 60 |
| 8.17. | Vergleich Ground-Truth $y_{\text{true}}$ zu Vorhersage $\hat{y}$ mit $n_{\min}$ , $n_{\frac{\Delta}{2}}$ und $n_{\Delta}$ . . . . . | 60 |
| 8.18. | Vergleich von GT, AU und EU unter Konfiguration 4 . . . . .                                                                         | 62 |
| 8.19. | Vergleich von GT und EU unter Konfiguration 5 . . . . .                                                                             | 67 |
| 8.20. | Vergleich von GT, AU und EU unter Konfiguration 6 . . . . .                                                                         | 69 |
| A.1.  | Szenarioselektionsmethode . . . . .                                                                                                 | 98 |

## Liste der Tabellen

|                                                                                                                                       |    |
|---------------------------------------------------------------------------------------------------------------------------------------|----|
| 7.2. R1 Vergleich von Unsicherheitsquantifizierungsmethoden in ML . . . . .                                                           | 12 |
| 7.4. R2 Allg. Faktoren Klassifizierung Unsicherheit KI-Systeme . . . . .                                                              | 13 |
| 7.6. R2.2 Modellspezifische Faktoren zur Unsicherheitsklassifizierung BNN . . . . .                                                   | 14 |
| 7.8. R2.2 Modellspezifische Faktoren zur Unsicherheitsklassifizierung ENN . . . . .                                                   | 15 |
| 7.10. R3 Abschwächung allgemeiner Faktoren KI-Systeme . . . . .                                                                       | 16 |
| 7.12. R3.1 Abschwächung modellspezifischer Faktoren BNN . . . . .                                                                     | 17 |
| 7.14. R3.2 Abschwächung modellspezifischer Faktoren ENN . . . . .                                                                     | 18 |
| 7.16. A1 Auswahlkriterien für die Literaturübersicht zu EDL . . . . .                                                                 | 25 |
| 7.18. A1 Scopes der Fragen der Literaturübersicht zu EDL . . . . .                                                                    | 26 |
| 7.19. A1 Scopes der Antworten der Literaturübersicht zu EDL mit aktuellen Quellen<br>(2016–2025) . . . . .                            | 27 |
| 7.21. A2 Unsicherheitsmetriken für Surrogatmodelle . . . . .                                                                          | 28 |
| 7.23. Anwendungsdomänen im Ingenieurwesen für ML-Surrogatmodelle nach Gawlikowski, Ulmer . . . . .                                    | 29 |
| 7.25. Vergleich modellspezifischer und anwendungsspezifischer Unsicherheitsmetriken für Surrogatmodelle im autonomen Fahren . . . . . | 30 |
| 7.27. A4 Vergleichskriterien und Analyseziele für EDL zu anderen Methoden . . . . .                                                   | 31 |
| 7.29. A5 Vergleich der UQ-Ansätze für BNN und CP . . . . .                                                                            | 32 |
| 7.31. Vergleich der Modelle BNN (HMC, SVI) und CP hinsichtlich Performanz und Unsicherheitsquantifizierung. . . . .                   | 33 |
| 7.33. A6 Übersicht der UQ-Methoden BNN, CP und ENN . . . . .                                                                          | 34 |
| 7.35. A6 Min. Sensitivitätsanalyse ENN mit $\pm 2\sigma$ . . . . .                                                                    | 35 |
| 7.37. A6 Min. Sensitivitätsanalyse BNN mit $\pm 2\sigma$ . . . . .                                                                    | 36 |
| 7.39. A6 Min. Sensitivitätsanalyse CP mit $\pm 2\sigma$ . . . . .                                                                     | 36 |
| 7.41. A6 Zusammenfassung Sensitivität UQ-Methoden BNN, CP, ENN . . . . .                                                              | 37 |
| 8.2. Vergleich reproduzierte Ergebnisse mit Benchmark <i>Amini et al. (2020)</i> . . . . .                                            | 41 |
| 8.4. Bewertung BNN, ENN Loss über verschiedene Datensätze . . . . .                                                                   | 43 |
| 8.5. ENN Modellverhalten und Unsicherheitsmerkmale in der Signalrekonstruktion . . . . .                                              | 50 |

|      |                                                                                                        |    |
|------|--------------------------------------------------------------------------------------------------------|----|
| 8.6. | Vergleich von Netzwerkarchitekturen und Trainingsparametern in Konfiguration 4 . . . . .               | 61 |
| 8.7. | Vergleich ausgewählter Unsicherheitsmetriken zwischen ENN und synthetisch approximiertem BNN . . . . . | 71 |
| 1.   | Hardwarespezifikationen der Geräte 1 bis 4 . . . . .                                                   | 95 |

# Literaturverzeichnis

## Beitrag

- [7] D. Ulmer, C. Hardmeier und J. Frellsen, „Prior and Posterior Networks: A Survey on Evidential Deep Learning Methods For Uncertainty Estimation“, *Transactions on Machine Learning Research*, Jg. 4, Nr. 1, S. 1–47, 2023. DOI: 10.48550/arXiv.2110.03051.
- [9] B. Lakshminarayanan, A. Pritzel und C. Blundell, „Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles“, in *Advances in Neural Information Processing Systems*, 2017.
- [10] Y. Ovadia und et al., „Can You Trust Your Model’s Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift“, in *Advances in Neural Information Processing Systems*, 2019.
- [11] P. Perdikaris, M. Raissi, A. Damianou, N. D. Lawrence und G. E. Karniadakis, „Nonlinear information fusion algorithms for data-efficient multi-fidelity modelling“, *Proceedings of the Royal Society A*, Jg. 473, Nr. 2198, 2017.
- [12] Z. Liu, Y. Zhang und G. E. Karniadakis, „Multi-fidelity Deep Neural Networks: Algorithms and Applications“, *Journal of Computational Physics*, Jg. 424, S. 109 716, 2020.
- [15] C. Blundell, J. Cornebise, K. Kavukcuoglu und D. Wierstra, „Weight Uncertainty in Neural Networks“, in *International Conference on Machine Learning (ICML)*, 2015.
- [18] A. Kendall und Y. Gal, „What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?“, *NeurIPS*, 2017.
- [20] S. K. Manchingal und F. Cuzzolin, *Position: Epistemic Artificial Intelligence is Essential for Machine Learning Models to ‘Know When They Do Not Know’*, 2025. arXiv: 2505.04950v1 [CS . AI ].
- [22] T. Bayes, „An Essay towards Solving a Problem in the Doctrine of Chances“, *Philosophical Transactions of the Royal Society of London*, Jg. 53, S. 370–418, 1763.

- [23] A. P. Dempster, „Upper and Lower Probabilities Induced by a Multivalued Mapping“, *Annals of Mathematical Statistics*, Jg. 38, Nr. 2, S. 325–339, 1967. DOI: 10.1214/aoms/1177698950.
- [24] G. Shafer, *A Mathematical Theory of Evidence*. Princeton, NJ: Princeton University Press, 1976, ISBN: 0-691-10042-X.
- [25] M. Caprio, M. Sultana, E. Elia und F. Cuzzolin, „Credal Learning Theory“, *arXiv preprint arXiv:2402.00957v4*, Okt. 2024, Version 4, revised 23 October 2024. DOI: 10.48550/arXiv.2402.00957. arXiv: 2402.00957 [cs.LG].
- [27] A. Javanmardi, D. Stutz und E. Hüllermeier, „Conformalized Credal Set Predictors“, in *Proceedings of the 38th Conference on Neural Information Processing Systems (NeurIPS)*, Veröffentlicht auf Open Access LMU am 18.Dez.2024, New York, NY, USA, Dez. 2024.
- [28] T. Denoeux, „Evidential neural networks for regression and their application to uncertainty quantification“, *arXiv preprint arXiv:2208.00647*, Aug. 2022. arXiv: 2208.00647 [stat.ML].
- [29] N. Houlsby, F. Huszár, Z. Ghahramani und M. Lengyel, „Bayesian Active Learning for Classification and Preference Learning“, *CoRR*, Jg. abs/1112.5745, 2011.
- [30] A. Malinin und M. Gales, „Uncertainty Estimation in Neural Networks using Prior Networks“, *arXiv preprint arXiv:1802.10501*, 2019.
- [32] M. Sensoy, L. Kaplan und M. Kandemir, „Evidential Deep Learning: Theory and Practice“, *arXiv preprint arXiv:2301.XXXXX*, 2023.
- [33] J. Peters u. a., „On the Mathematical Consistency of Evidential Learning“, *arXiv preprint arXiv:2302.XXXXX*, 2023.
- [34] Z. Tian u. a., „Evidential Graph Neural Networks“, *arXiv preprint arXiv:2401.XXXXX*, 2024.
- [35] M. Sensoy, L. Kaplan und M. Kandemir, „Uncertainty-Aware Deep Classification Using Evidential Reasoning“, *arXiv preprint arXiv:2003.XXXXX*, 2020.
- [36] Dordevic u. a., „Evidential Transformers For Improved Image Retrieval“, *arXiv preprint arXiv:2409.XXXXX*, 2023.
- [37] Z. Zhou u. a., „Evidential Transformer Networks“, *arXiv preprint arXiv:2204.XXXXX*, 2022.
- [38] A. Amini u. a., „Improved KL Regularization for Evidential Regression“, *arXiv preprint arXiv:2305.XXXXX*, 2023.
- [39] A. Dhamija, M. Günther und T. E. Boulton, „Reducing Network Agnostophobia“, in *NeurIPS*, 2018.

- [40] D. A. Tran u. a., „Separating Aleatoric and Epistemic Uncertainty in Evidential Deep Learning“, *arXiv preprint arXiv:2304.XXXXX*, 2023.
- [41] D. Feng u. a., „A Review of Uncertainty Estimation in Deep Learning for Autonomous Driving“, *IEEE Transactions on Intelligent Transportation Systems*, 2022.
- [42] L. Wang u. a., „Uncertainty Modeling in Industrial AI“, *IEEE Transactions on Industrial Informatics*, 2022.
- [43] A. Gupta u. a., „Evidential Learning for Industrial Predictive Maintenance“, *arXiv preprint arXiv:2306.XXXXX*, 2023.
- [44] X. Chen u. a., „Uncertainty Estimation in Vision-Language Models“, *arXiv preprint arXiv:2401.XXXXX*, 2024.
- [45] Y. Gal und Z. Ghahramani, „Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning“, in *ICML*, 2016.
- [46] C. Guo, G. Pleiss, Y. Sun und K. Q. Weinberger, „On Calibration of Modern Neural Networks“, *ICML*, 2017.
- [50] A. N. Angelopoulos und S. Bates, „A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification“, *arXiv preprint arXiv:2107.07511*, 2021.
- [51] D. J. MacKay, „A Practical Bayesian Framework for Backpropagation Networks“, *Neural Computation*, Jg. 4, Nr. 3, S. 448–472, 1992.
- [52] G. Shafer und V. Vovk, „A Tutorial on Conformal Prediction“, *Journal of Machine Learning Research*, Jg. 9, S. 371–421, 2008.

## **Buch (Sammelwerk)**

- [3] S. Y. Nof, Hrsg., *Springer Handbook of Automation*, Cham, 2023. DOI: 10.1007/978-3-540-78831-7.

## **Buch (Monographie)**

- [8] C. E. Rasmussen und C. K. I. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.
- [16] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.
- [21] A. Jøsang, *Subjective Logic: A Formalism for Reasoning Under Uncertainty*. Springer, 2016.

- [49] V. Vovk, A. Gammerman und G. Shafer, *Algorithmic Learning in a Random World*. Springer, 2005.

## Zeitschriftenaufsatz

- [4] E. Hüllermeier und W. Waegeman, *Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods*, 2021. DOI: 10.1007/s10994-021-05946-3.
- [5] A. Kendall und Y. Gal, „What uncertainties do we need in Bayesian deep learning for computer vision?“, *NIPS’17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, S. 5580–5590, 2017.
- [19] J. Lanini, M. T. D. Huynh, G. Scebba, N. Schneider und R. Rodríguez-Pérez, „UNIQUE: A Framework for Uncertainty Quantification Benchmarking“, *J. Chem. Inf. Model.*, Jg. 64, Nr. 22, S. 8379–8386, 2024. DOI: 10.1021/acs.jcim.4c01578. eprint: 2024Nov14.
- [47] J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher, M. Shahzad, W. Yang, R. Bamler und X. X. Zhu, „A survey of uncertainty in deep neural networks“, *Artificial Intelligence Review*, Jg. 56, Nr. S1, S. 1513–1589, 2023. DOI: 10.1007/s10462-023-10562-9.

## Internetdokument

- [1] Bundesministerium für Bildung und Forschung (BMBF), *Förderschwerpunkt Künstliche Intelligenz: Maschinelles Lernen / Data Analytics*, Online-Dokument, Deutsches Zentrum für Luft- und Raumfahrt (DLR), Zugriff am 10.04.2025, 2025.
- [2] Deutsche Allianz Meeresforschung. „Munition im Meer: Sachstand und Perspektiven“. Zugriff am 04.09.2025, besucht am 4. Sep. 2025. Adresse: <https://www.allianz-meeresforschung.de/app/uploads/2024/10/241021-dam-factsheet-munition.pdf>.
- [6] A. Hoarau, B. Quost, S. Destercke und W. Waegeman, *Reducing Aleatoric and Epistemic Uncertainty through Multi-modal Data Acquisition*, arXiv preprint, arXiv:2501.18268v1, 2025.
- [14] Andreas Kreutz. „Unsicherheit in autonomen Systemen: Nicht nur eine Frage der Perzeption: Teil 1“, besucht am 12. Apr. 2025. Adresse: <https://safe-intelligence.fraunhofer.de/artikel/unsicherheit-in-autonomen-systemen-teil-1>.

## Thesis

- [13] Y. Gal, „Uncertainty in Deep Learning“, Diss., University of Cambridge, 2016.
- [83] S. Depeweg, *Modeling Epistemic and Aleatoric Uncertainty with Bayesian Neural Networks and Latent Variables*, Dissertation, Fakultät für Informatik, München, Deutschland, 2019.

## Anhang

### Datensätze

- [53] K. Bache und M. Lichman, *Boston Housing Data*, Dataset, UCI Machine Learning Repository, 2013. DOI: 10.24432/C5K31K.
- [54] P. Tfekci und H. Kaya, *Combined Cycle Power Plant*, UCI Machine Learning Repository, DOI: <https://doi.org/10.24432/C5002N>, 2014.
- [55] I.-C. Yeh, *Concrete Compressive Strength*, UCI Machine Learning Repository, DOI: <https://doi.org/10.24432/C5PK67>, 1998.
- [56] A. Coraddu, L. Oneto, A. Ghio, S. Savio, D. Anguita und M. Figari, *Condition Based Maintenance of Naval Propulsion Plants*, Dataset, UCI Machine Learning Repository, 2014. DOI: 10.24432/C5K31K.
- [57] A. Tsanas und A. Xifara, *Energy Efficiency*, UCI Machine Learning Repository, DOI: <https://doi.org/10.24432/C51307>, 2012.
- [58] R. A. Fisher, *The Iris Dataset*, Dataset, UCI Machine Learning Repository, 1936.
- [59] P. Cortez, A. Cerdeira, F. Almeida, T. Matos und J. Reis, *Wine Quality Dataset*, Dataset, UCI Machine Learning Repository, 2009.

### Software

- [60] DuckDB Developers, *DuckDB*, Software, 2024.
- [61] EDL PyTorch Developers, *EDL PyTorch*, Software, 2024.
- [62] Loguru Developers, *Loguru*, Software, 2024.
- [63] Matplotlib Developers, *Matplotlib*, Software, 2024.
- [64] Jupyter Notebook Developers, *Jupyter Notebook*, Software, 2024.

- [65] NumPy Developers, *NumPy*, Software, 2024.
- [66] Openpyxl Developers, *Openpyxl*, Software, 2024.
- [67] Pandas Developers, *Pandas*, Software, 2024.
- [68] Properscoring Developers, *Properscoring*, Software, 2024.
- [69] Pyro PPL Developers, *Pyro PPL*, Software, 2024.
- [70] Pytest Developers, *Pytest*, Software, 2024.
- [71] Ruff Developers, *Ruff*, Software, 2024.
- [72] Scikit-learn Developers, *Scikit-learn*, Software, 2024.
- [73] SciPy Developers, *SciPy*, Software, 2024.
- [74] Seaborn Developers, *Seaborn*, Software, 2024.
- [75] Sphinx Developers, *Sphinx*, Software, 2024.
- [76] TensorFlow Developers, *TensorFlow*, Software, 2024.
- [77] PyTorch Developers, *PyTorch*, Software, 2024.
- [78] Torchmetrics Developers, *Torchmetrics*, Software, 2024.
- [79] Torchvision Developers, *Torchvision*, Software, 2024.
- [80] Xlrd Developers, *Xlrd*, Software, 2024.

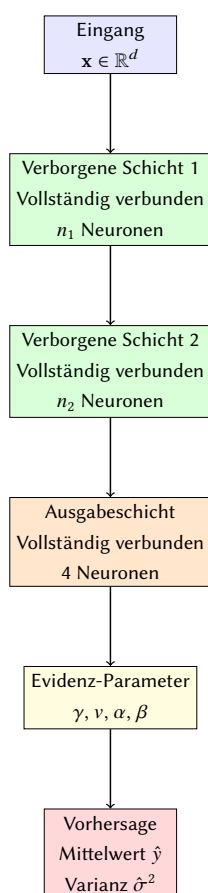
## Repositories

- [17] M. Sensoy, L. Kaplan und M. Kandemir. „Evidential deep learning to quantify classification uncertainty“. Accessed: 2025-04-24. Adresse: <https://muratsensoy.github.io/uncertainty.html>.
- [26] A. Amini, W. Schwarting, A. Soleimany und D. Rus. „Deep evidential regression“. Accessed: 2025-04-24. Adresse: <https://github.com/aamini/evidential-deep-learning>.
- [31] N. Papernot, F. Faghri, N. Carlini, I. Goodfellow, R. Feinman, A. Kurakin, C. Xie, Y. Sharma, T. Brown, A. Roy, A. Matyasko, V. Behzadan, K. Hambarzumyan, Z. Zhang, Y.-L. Juang, Z. Li, R. Sheatsley, A. Garg, J. Uesato, W. Gierke, Y. Dong, D. Berthelot, P. Hendricks, J. Rauber, R. Long und P. McDaniel, „Technical Report on the CleverHans v2.1.0 Adversarial Examples Library“, CleverHans Lab, Techn. Ber., 2016.
- [48] N. Mavani. „Uncertainty in ML based Surrogate“. Accessed: 2025-06-25. Adresse: [https://github.com/NinadMavani/MA\\_Uncertainty\\_in\\_ML\\_based\\_Surrogates/](https://github.com/NinadMavani/MA_Uncertainty_in_ML_based_Surrogates/).

- [81] M. Windler, *Fork-Patch Amini 2020 Evidential-Deep-Learning*, <https://github.com/bBswhTdb0qEq75URS/evidential-deep-learning>, Commit 552b4613e3d024c6dab95cd71c1971b0e3805588. Accessed: 2025-07-01, 2025.
- [82] M. Windler. „Unsicherheiten in machine-learning-basierten Surrogatmodellen für die szenariobasierte Validierung autonomer Systeme“. Accessed: 2025-06-26. Adresse: [https://github.com/bBswhTdb0qEq75URS/BA\\_\\_U-i-mlb-Sm-f-d-s-V-a-S/tree/main/BA\\_\\_Projekt](https://github.com/bBswhTdb0qEq75URS/BA__U-i-mlb-Sm-f-d-s-V-a-S/tree/main/BA__Projekt).

# Visualisierungen Architekturen

## ENN Regression nach Amini [26]



### Details:

#### Input:

$\prod_{i=1}^k d_i$ : Eingabedimension = Produkt  
Achsen-Dimensionen

#### Versteckte Schichten:

$n_1 = 64$ : Neuronen Schicht 1 nach Amini 2020

$n_2 = 64$ : Neuronen Schicht 2 nach Amini 2020

Akt. Funktion: z. B. ReLU, Tanh

Dropout-Rate (falls genutzt)

#### Ausgabe Schicht:

4 Neuronen (für Evidenz-Parameter)

#### Evidenz-Parameter:

$\gamma$ : Schätzwert für Mittelwert (Softplus f. pos. Ausgabe)

$v$ : Stärke der Evidenz (Linear)

$\alpha$ : Formparameter der Varianzverteilung (Softplus f. pos. Ausgabe)

$\beta$ : Skalenparameter der Varianzverteilung (Softplus f. pos. Ausgabe)

#### Loss:

Negative Log-Likelihood +

Evidenz-Regularizer

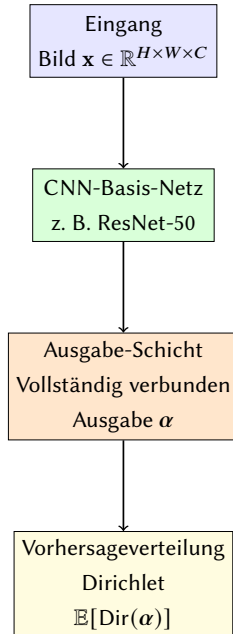
Regularisierungsgewicht  $\lambda$

#### Optimizer:

z. B. Adam, Lernrate  $\nu$

Siehe Hinweis auf den nächsten Seiten von Architekturdarstellungen aus Platzgründen.

## ENN Klassifikation nach Amini [26]



### Details:

#### Input:

$H, W$ : Bildhöhe und -breite  
 $C$ : Anzahl Kanäle (z. B. RGB = 3)

#### CNN-Backbone:

Architektur: z. B. ResNet-50 (Amini 2020 [26])  
 Anzahl Convolutional Blocks  
 Kernelgrößen, Filteranzahl  
 Pooling-Typ (z. B. MaxPooling)  
 Pretrained (ja/nein)

#### Vollverbundene Schicht:

Größe der FC-Schicht  
 Aktivierungsfunktion (z. B.  $\text{ReLU}(x) = \max(0, x)$ )

#### Ausgabe-Schicht:

$\alpha$ : Dirichlet-Parameter pro Klasse  
 Anzahl Klassen  $K$

#### Vorhersage:

Erwartungswert:

$$\hat{p}_c = \frac{\alpha_c}{\sum_j \alpha_j}$$

#### Unsicherheitsmaße:

Varianz aus Dirichlet berechenbar  
 Epistemische Unsicherheit sichtbar

#### Loss:

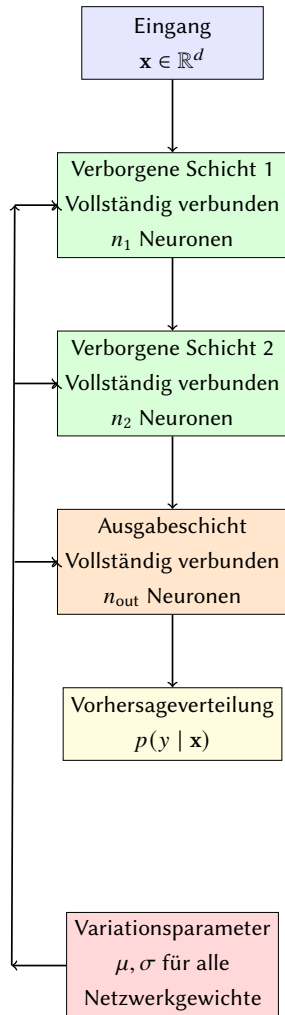
Expected MSE Loss  
 + KL-Regularizer  
 Regularisierungsgewicht  $\lambda$

#### Optimizer:

z. B. Adam, Lernrate  $\nu = 5 * 10^{-3}$

Die Übersichten wurden zu einer der Recherche begleitenden Reflektion ausgearbeitet und haben keinen Vollständigkeitsanspruch, insbesondere in Bezug auf Quellen oder Eingabedimensionen, die nicht zur Trainingsparametrisierung der Durchführung passen. Es wurden auch Konfigurationen mit Eingaben getestet, die nicht in der Durchführung enthalten oder für diese relevant sind.

## BNN SVI nach Pyro [69]



### Details:

#### Input:

$\prod_{i=1}^k d_i$ : Eingabedimension = Produkt Achsen-Dimensionen (SVI für niedrigdimensionale Lösungen)

#### Versteckte Schichten:

$n_1 = 64$ : Neuronen in Schicht 1  
 $n_2 = 64$ : Neuronen in Schicht 2  
 Aktivierungsfunktion: z. B. ReLU, Tanh  
 Dropout-Rate (falls genutzt)

#### Ausgabe Schicht:

$n_{\text{out}} = 2$  Neuronen  
 Aktivierungsfunktion (z. B. Linear, Softmax)

#### Variationsparameter (SVI):

Jedes Gewicht  $\theta$  hat Variationsparameter  $\mu, \sigma$  [69, S. 6, Z. 20–23]  
 Samples  $\theta^{(s)} \sim q(\theta)$   
 ELBO-Optimierung [69, S. 6, Z. 5–8]  
 KL-Divergenz zur Prior

#### Vorhersage:

$\hat{y} = \frac{1}{S} \sum_s y^{(s)}$   
 Vorhersage-Varianz reflektiert Gewichtsunsicherheit

#### „Loss“:

„minimizing the negative Evidence Lower Bound (ELBO)“ [69, S.6 Z.5-8]  

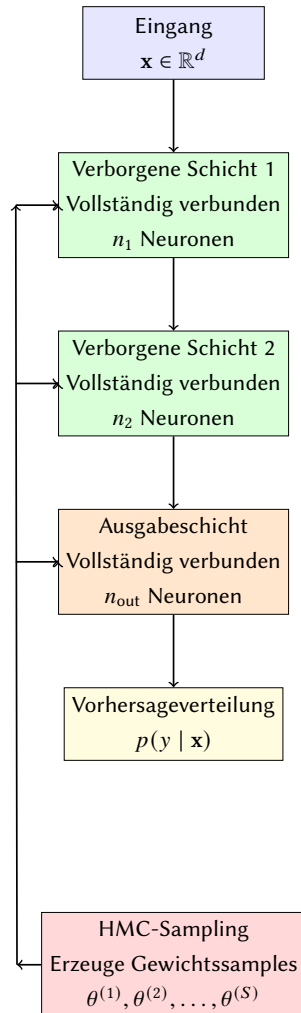
$$\min_{\phi} \left( -\mathbb{E}_{q_{\phi}(\theta)} [\log p(y | x, \theta)] + \text{KL} (q_{\phi}(\theta) \parallel p(\theta)) \right)$$

#### Optimizer:

Adam [69, S. 6, Z. 9–10]  
 Lernrate  $\nu = [1 * 10^{-4} : 1 * 10^{-3}]$

Die Übersichten wurden zu einer der Recherche begleitenden Reflektion ausgearbeitet und haben keinen Vollständigkeitsanspruch, insbesondere in Bezug auf Quellen oder Eingabedimensionen, die nicht zur Trainingsparametrisierung der Durchführung passen. Es wurden auch Konfigurationen mit Eingaben getestet, die nicht in der Durchführung enthalten oder für diese relevant sind.

## BNN HMC nach Pyro [69]



### Details:

#### Eingabe:

$\prod_{i=1}^k d_i$ : Eingabedimension = Produkt Achsen-Dimensionen

#### Versteckte Schichten:

$n_1$ : Neuronen in Schicht 1

$n_2$ : Neuronen in Schicht 2

Aktivierungsfunktion: Softmax, ReLU

Dropout-Rate (falls genutzt)

#### Ausgabe Schicht:

$n_{out}$  Neuronen

Aktivierungsfunktion (z. B. linear, softmax)

#### HMC-Sampling:

Samples  $\theta^{(s)}$  aus  $p(\theta | D)$

Anzahl Samples  $S$

Leapfrog-Schritte

Schrittweite

Akzeptanzrate

#### Vorhersage:

Vorhersagemittel:

$$\hat{y} = \frac{1}{S} \sum_s y^{(s)}$$

Vorhersage-Varianz reflektiert

Gewichtsunsicherheit

#### „Loss“:

z. B. Mean Squared Error

Negative Log-Likelihood

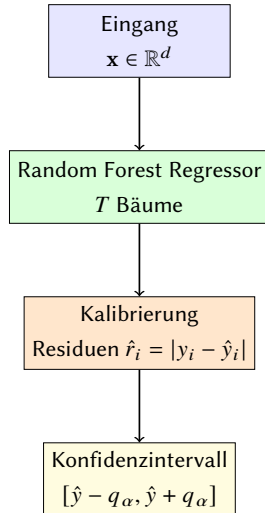
#### Optimizer:

z. B. adaptiver Algorithmus wie NUTS mit

$step\_size \in [0.1, 0.01]$

Die Übersichten wurden zu einer der Recherche begleitenden Reflektion ausgearbeitet und haben keinen Vollständigkeitsanspruch, insbesondere in Bezug auf Quellen oder Eingabedimensionen, die nicht zur Trainingsparametrisierung der Durchführung passen. Es wurden auch Konfigurationen mit Eingaben getestet, die nicht in der Durchführung enthalten oder für diese relevant sind.

## CP mit nach Angelopoulos [50]



### Details:

#### Modell:

Random Forest Regression

$T = 100$  Entscheidungsbäume

Trainiert auf  $\mathcal{D}_{\text{train}}$

#### Kalibrierung:

$\mathcal{D}_{\text{cal}}$  enthält Residuen  $\hat{r}_i$

Konfidenzquantil:  $q_\alpha = \text{Quantil}_{1-\alpha}(|y_i - \hat{y}_i|)$

#### Vorhersage:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^T f_t(\mathbf{x})$$

Konfidenzintervall:

$$[\hat{y} - q_\alpha, \hat{y} + q_\alpha]$$

#### Unsicherheiten:

*Epistemisch*: Streuung zwischen

Baumvorhersagen

$$\text{Var}_{\text{trees}}(\hat{y}_t)$$

*Aleatorisch*: Residualvarianz  $\sigma^2 = \text{Var}(\hat{r}_i)$

#### Parametervariation:

$$\alpha \in \{0.20, 0.10, 0.05, 0.01\}$$

#### Loss:

Root Mean Squared Error (RMSE)

Die Übersichten wurden zu einer der Recherche begleitenden Reflektion ausgearbeitet und haben keinen Vollständigkeitsanspruch, insbesondere in Bezug auf Quellen oder Eingabedimensionen, die nicht zur Trainingsparametrisierung der Durchführung passen. Es wurden auch Konfigurationen mit Eingaben getestet, die nicht in der Durchführung enthalten oder für diese relevant sind.

## Hardware-Spezifikation

Tabelle 1.: Hardwarespezifikationen der Geräte 1 bis 4

|                        |                                                                   |
|------------------------|-------------------------------------------------------------------|
| <b>Name</b>            | Yoga Pro 7 14APH8                                                 |
| <b>Prozessor</b>       | AMD Ryzen 7 7840HS (8 Kerne, 16 Threads, 3.8 – 5.1 GHz)           |
| <b>Speicher</b>        | 1 TB M.2 2280 PCIe®4.0 x4 NVMe®SSD                                |
| <b>Arbeitsspeicher</b> | 32 GB LPDDR5x-6400, 6400 MHz                                      |
| <b>Grafikspeicher</b>  | Radeon 780M 3.8 GHz 4GB (On-Board)                                |
| <b>Betriebssystem</b>  | Windows 11 Home (24H2)                                            |
| <b>Name</b>            | HP EliteBook 845 G11                                              |
| <b>Prozessor</b>       | AMD Ryzen 7 8840                                                  |
| <b>Speicher</b>        | 500 GB PCIe®Gen4x4 NVMe™M.2 SSD                                   |
| <b>Arbeitsspeicher</b> | 16 GB DDR5-5600 MHz (2x 8 GB)                                     |
| <b>Grafikspeicher</b>  | Radeon 780M Graphics 3,3 GHz On-Board                             |
| <b>Betriebssystem</b>  | Windows 11 Enterprise                                             |
| <b>Name</b>            | Dell Linux Server (Small-Office)                                  |
| <b>Prozessor</b>       | AMD Ryzen 7, Intel i7 oder vergleichbar, mindestens oder aufwärts |
| <b>Speicher</b>        | 1 TB M.2 2280 PCIe®4.0 x4 NVMe®SSD                                |
| <b>Arbeitsspeicher</b> | Mindestens 16 GB LPDDR5x-6400, 6400 MHz                           |
| <b>Grafikspeicher</b>  | Vernachlässigbar                                                  |
| <b>Betriebssystem</b>  | Linux Mint 22.1 „Xia“ LTS-2029-04                                 |
| <b>Name</b>            | Linux Server (einer Abteilung zugeordnet)                         |
| <b>Prozessor</b>       | Unbekannt                                                         |
| <b>Speicher</b>        | Mind. 500 GB pro Benutzer, Rest unbekannt                         |
| <b>Arbeitsspeicher</b> | Unbekannt                                                         |
| <b>Grafikspeicher</b>  | Vernachlässigbar                                                  |
| <b>Betriebssystem</b>  | Unbekannt                                                         |

## Herleitung der Unsicherheitsäquivalenz zwischen Evidential Neural Networks und GUM

Die Messunsicherheit gemäß GUM (Guide to the Expression of Uncertainty in Measurement) wird im Kern als Varianz des Messwerts verstanden:  $u(x)^2 = \text{Var}(x) = \mathbb{E}[(x - \mu)^2]$ . Messunsicherheiten werden meist durch Konfidenzintervalle angegeben:  $x \pm k \cdot u(x)$ , wobei  $k$  das Vertrauensniveau bestimmt.

**Ausgabe eines Evidential Neural Network.** Ein ENN liefert keine einzelne Punktvorhersage, sondern eine Wahrscheinlichkeitsverteilung, oft eine Normal-Inverse-Gamma-Verteilung (NIG). Es schätzt Parameter  $\mu$  (Mittelwert),  $\nu$  (Evidenz über die Präzision) sowie  $\alpha, \beta$  (Parameter der Inverse-Gamma-Verteilung für  $\sigma^2$ ). Die bedingte Verteilung über  $y$  lautet:  $p(y | \mu, \alpha, \beta, \nu) = \int \mathcal{N}(y | \mu, \sigma^2/\nu) \cdot \text{InvGamma}(\sigma^2 | \alpha, \beta) d\sigma^2$ . Diese Mischverteilung führt zu einer Student- $t$ -Verteilung für  $y$ .

**Varianz eines Evidential Neural Network.** Die Vorhersagevarianz (hPredictive Variance) ergibt sich zu  $\text{Var}(y) = \mathbb{E}[\sigma^2] + \frac{\beta}{(\alpha-1)\nu}$  mit  $\mathbb{E}[\sigma^2] = \frac{\beta}{\alpha-1}$ . Die Vorhersageunsicherheit lässt sich schreiben als  $u(y)^2 = \underbrace{\mathbb{E}[\sigma^2]}_{\text{aleatorisch}} + \underbrace{\frac{\mathbb{E}[\sigma^2]}{\nu}}_{\text{epistemisch}}$ .

**Äquivalenz zu GUM.** Dies entspricht exakt der GUM-Definition, die die Gesamtmessunsicherheit als Summe verschiedener Unsicherheitsquellen beschreibt. Somit kann die Varianz eines ENN direkt als Messunsicherheit gemäß GUM interpretiert werden. Das zugehörige Konfidenzintervall lautet:  $[y_{\text{pred}} - k \cdot u(y), y_{\text{pred}} + k \cdot u(y)]$ .

**Zusammenfassung.** Die in Evidential Neural Networks modellierte Predictive Variance entspricht der Messunsicherheit nach GUM, da sie sowohl aleatorische Streuung als auch epistemische Unsicherheit integriert. Damit lassen sich Unsicherheiten eines ENN direkt in die standardisierte Unsicherheitsangabe eines Messprozesses überführen:  $u(y)^2 = \frac{\beta}{\alpha-1} + \frac{\beta}{(\alpha-1)\nu}$ .

## Herleitung der Loss-Erweiterung im ENN nach Oh und Shin

**Beweis der Lipschitz-Stetigkeit von  $\mathcal{L}(x)$ :**

Definiere die Funktion als:

$$\mathcal{L}(x) = \begin{cases} x^2 & \text{für } x^2 < U \\ 2\sqrt{U} \cdot |x| - U & \text{für } x^2 \geq U \end{cases} \quad \text{mit } U := U_{\nu, \alpha}$$

**1. Stetigkeit an der Übergangsstelle  $|x| = \sqrt{U}$ :**

$$\lim_{x \rightarrow \sqrt{U}^-} \mathcal{L}(x) = (\sqrt{U})^2 = U \quad \text{und} \quad \lim_{x \rightarrow \sqrt{U}^+} \mathcal{L}(x) = 2\sqrt{U} \cdot \sqrt{U} - U = 2U - U = U$$

$\Rightarrow \mathcal{L}$  ist stetig an  $x = \pm\sqrt{U}$

**2. Ableitung (Gradient) der beiden Zweige:**

- Für  $|x| < \sqrt{U}$ :  $\mathcal{L}'(x) = 2x \Rightarrow |\mathcal{L}'(x)| \leq 2\sqrt{U}$
- Für  $|x| > \sqrt{U}$ :  $\mathcal{L}'(x) = \pm 2\sqrt{U} \Rightarrow |\mathcal{L}'(x)| = 2\sqrt{U}$

**3. Schluss:** Die Ableitung ist überall definiert oder beschränkt (außer an  $x = 0$ , wo die Funktion zwar nicht differenzierbar ist, aber trotzdem eine endliche Steigung hat), also ist  $\mathcal{L}$  global Lipschitz-stetig mit:

$$L = \sup_{x \in \mathbb{R}} |\mathcal{L}'(x)| = 2\sqrt{U}$$

$\Rightarrow \mathcal{L}$  ist Lipschitz-stetig mit Konstante  $L = 2\sqrt{U}$

# A. Szenarioselektionsmethode

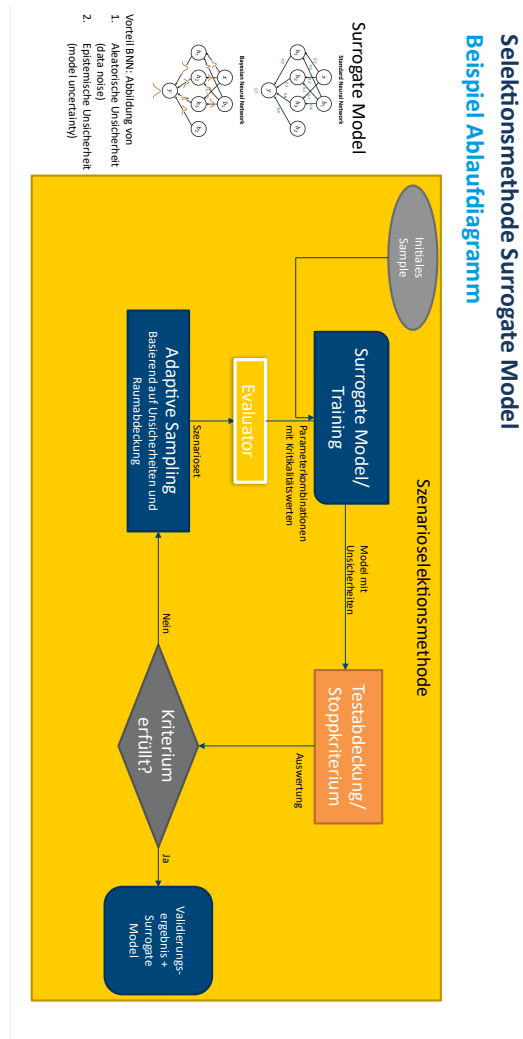


Abbildung A.1.: Szenarioselektionsmethode