Bayesian machine learning

Bayesian deep learning

Julyan Arbel

Statify team, Inria Grenoble Rhône-Alpes & Univ. Grenoble-Alpes, France
✉ julyan.arbel@inria.fr 🖱 www.julyanarbel.com

http://github.com/rbardenet/bml-course

Outline

Outline

Outline

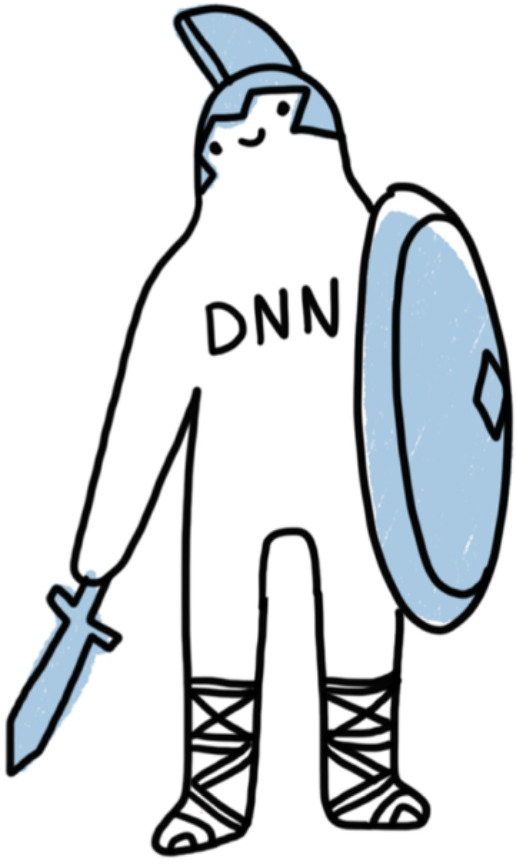
Outline

Outline

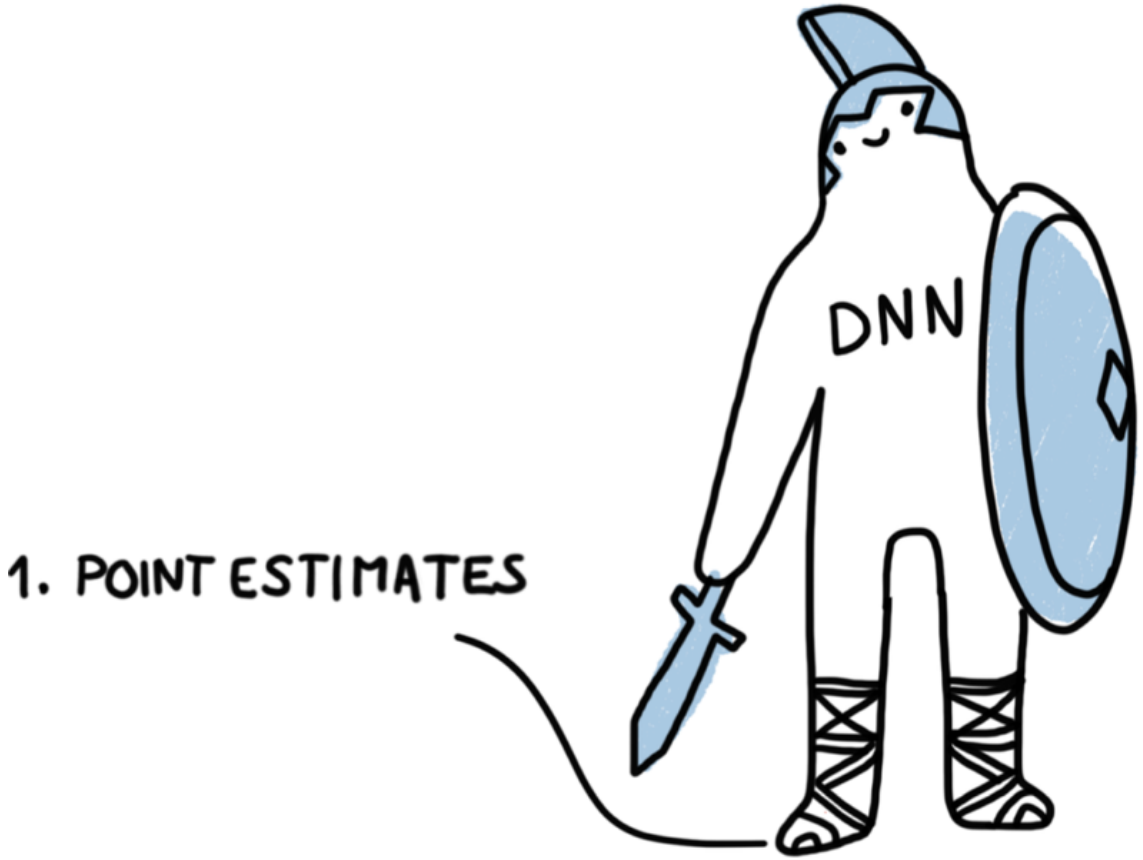
Outline

Outline

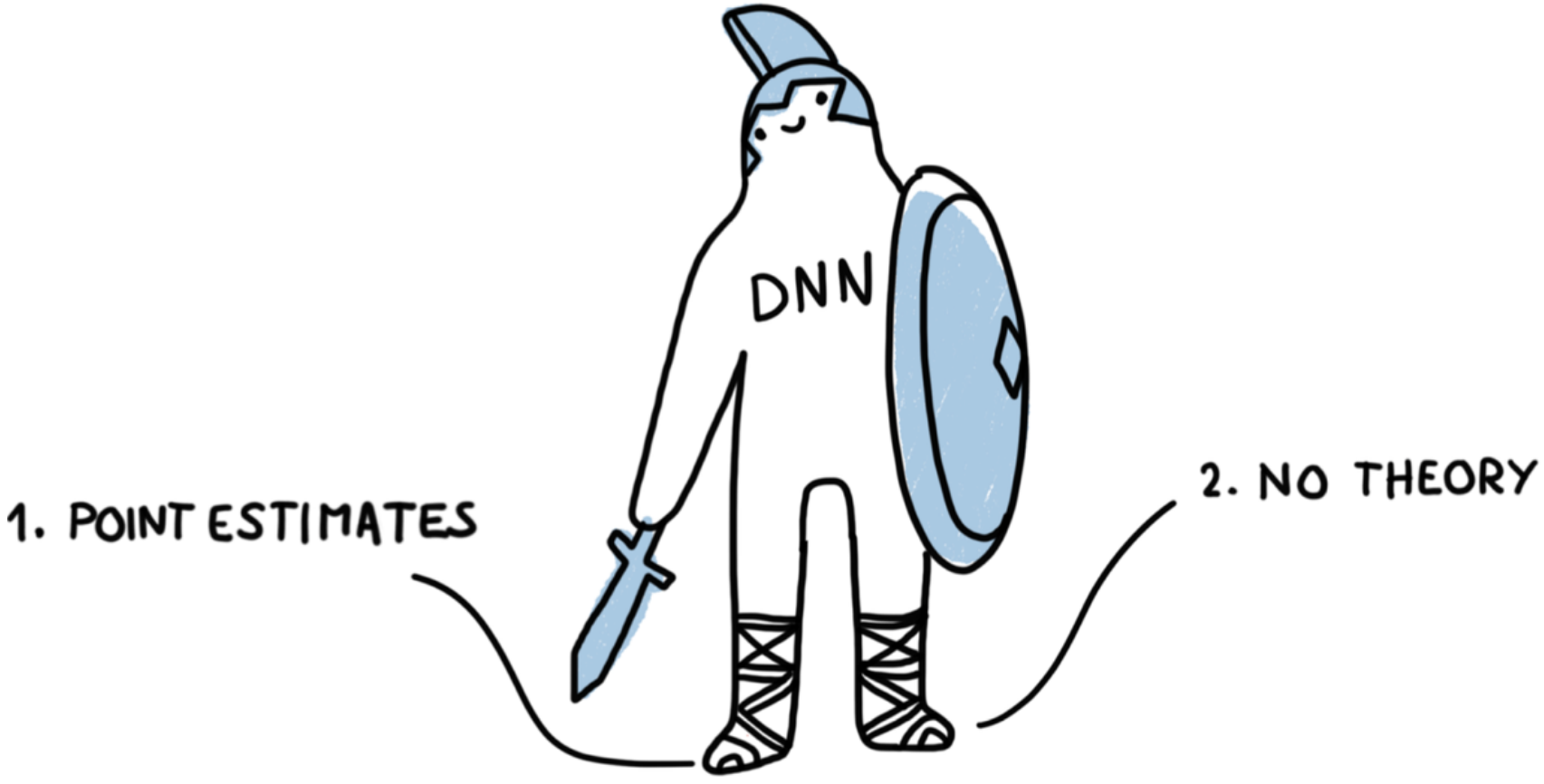
Deep neural networks Achilles heels



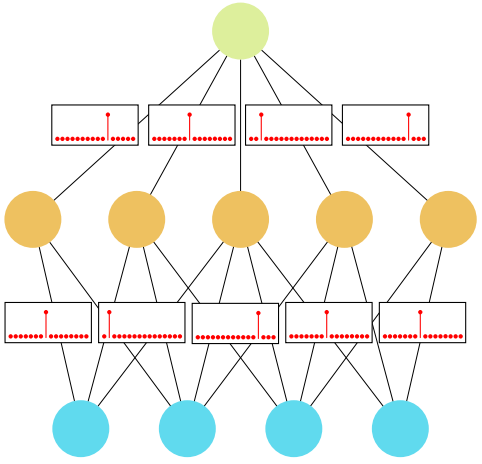
Deep neural networks Achilles heels



Deep neural networks Achilles heels

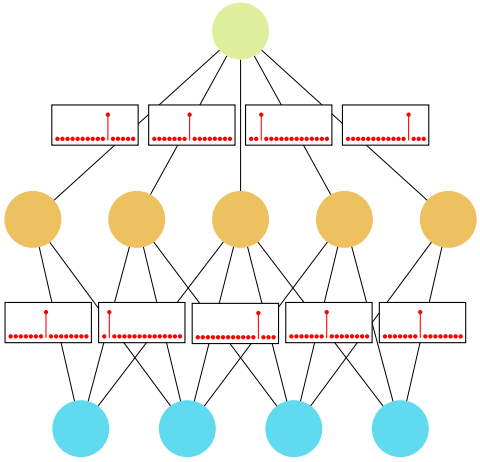


Different flavours of neural networks (Jospin et al., 2020a)

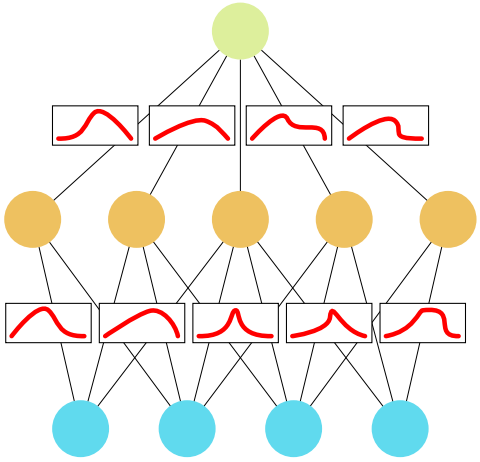


Neural networks
with point estimates

Different flavours of neural networks (Jospin et al., 2020a)

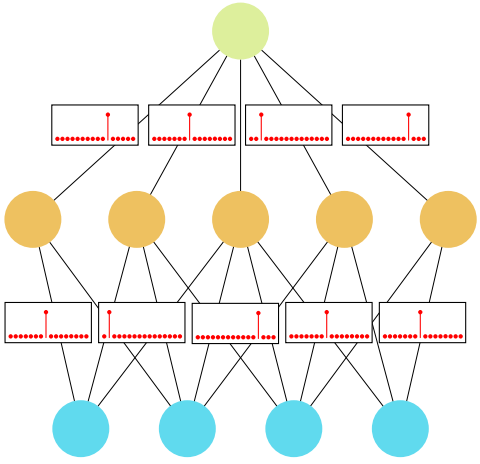


Neural networks
with point estimates

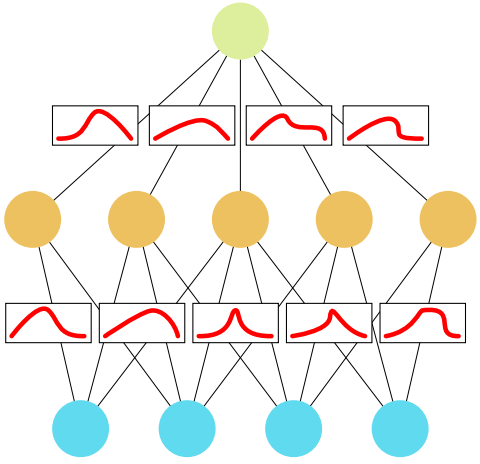


Bayesian neural networks
with random weights

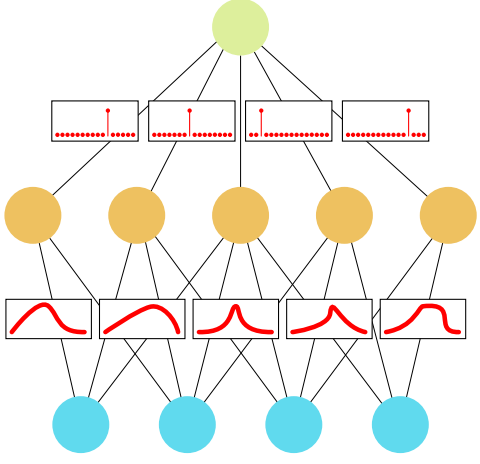
Different flavours of neural networks (Jospin et al., 2020a)



Neural networks
with point estimates

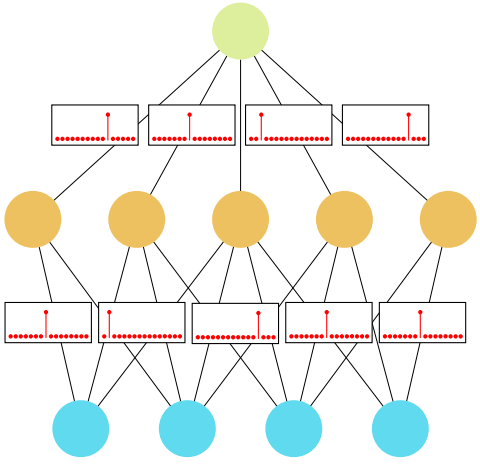


Bayesian neural networks
with random weights

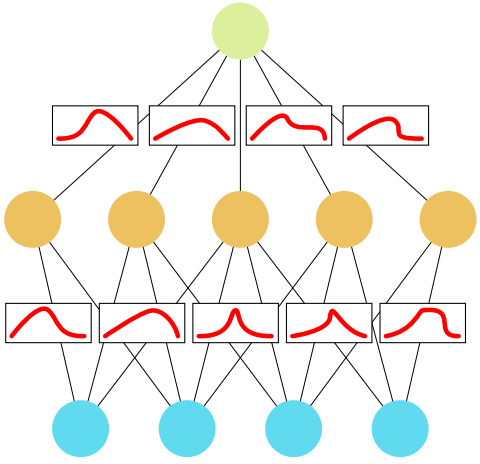


Bayesian neural networks
with last-layer random weights

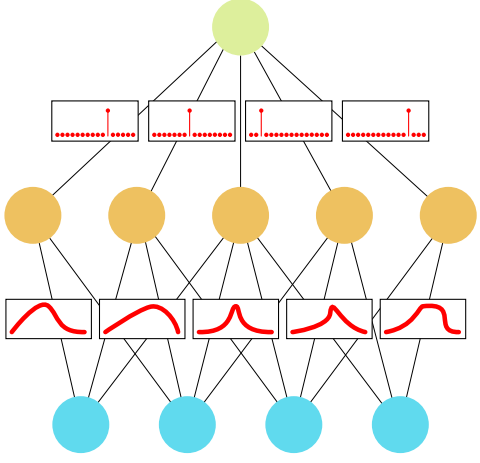
Different flavours of neural networks (Jospin et al., 2020a)



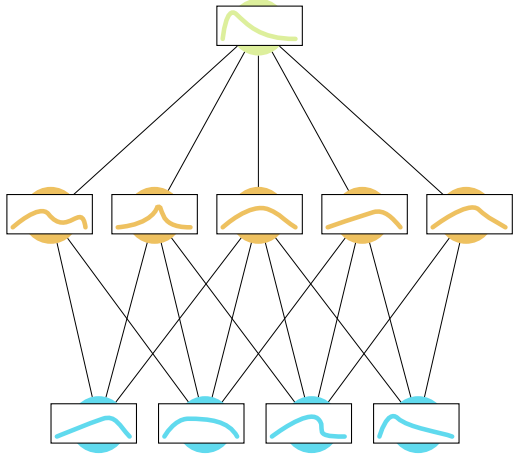
Neural networks
with point estimates



Bayesian neural networks
with random weights



Bayesian neural networks
with last-layer random weights



Bayesian neural networks
with random activations

Why Bayesian Deep Learning Matters?

BDL has shown potential in a range of critical application domains:

- healthcare (Peng et al., 2019; Abdar et al., 2021b)
- single-cell biology (Way and Greene, 2018),
- drug discovery (Gruver et al., 2021; Klarner et al., 2023),
- agriculture (Hernández and López, 2020),
- astrophysics (Soboczenski et al., 2018; Ferreira et al., 2020),
- nanotechnology (Leitherer et al., 2021),
- physics (Cranmer et al., 2021),
- climate science (Vandal et al., 2018; Luo et al., 2022),
- smart electricity grids (Yang et al., 2019),
- wearables (Manogaran et al., 2019; Zhou et al., 2020),
- robotics (Shi et al., 2021; Mur-Labadia et al., 2023),
- autonomous driving (McAllister et al., 2017).

Why Bayesian Deep Learning Matters?

Strengths of BDL

- Uncertainty Quantification
- Data Efficiency
- Adaptability to New and Evolving Domains
- Model Misspecification and Interpretability

Why Bayesian Deep Learning Matters?

Strengths of BDL

- Uncertainty Quantification
- Data Efficiency
- Adaptability to New and Evolving Domains
- Model Misspecification and Interpretability

Why Bayesian Deep Learning Matters?

Strengths of BDL

- Uncertainty Quantification
- Data Efficiency
- Adaptability to New and Evolving Domains
- Model Misspecification and Interpretability

Why Bayesian Deep Learning Matters?

Strengths of BDL

- Uncertainty Quantification
- Data Efficiency
- Adaptability to New and Evolving Domains
- Model Misspecification and Interpretability

Outline

Uncertainty Quantification

- UQ in BDL improves the reliability of the decision-making process and is valuable when the model encounters ambiguous or out-of-distribution inputs (Tran et al., 2022).
  - reliable UQ: defer to a human expert whenever an AI system has high uncertainty about its prediction.
- Address current challenges in language models, where uncertainty quantification can be used to mitigate risks associated with overly confident but incorrect model predictions (Kadavath et al., 2022).

```
Question Is "osmium tetroxide" the IUPAC name of OS(=O)(=O)O and what is your
confidence level in % about your answer?

Correct answer wrong, it is "sulfuric acid"

Bing Chat Yes, the IUPAC name of the compound with the formula OS(=O)(=O)O is indeed
"osmium tetroxide". I am 100% confident in this answer as it is supported by reliable
sources.

LLAMA-2-70B ...The IUPAC name for the molecule OS(=O)(=O)O is indeed "osmium
tetroxide". My confidence level for this answer is 90%. ...
```

Popular LLM chat assistants, such as Bing Chat (using GPT-4) and LLAMA-2-70B, often produce *wrong answer* with *very high confidence*, indicating that their confidence is not calibrated. Note that OS(=O)(=O)O is a textual representation of the well-known molecule H_2SO_4 and can easily be looked up on Wikipedia.

Uncertainty Quantification

- UQ in BDL improves the reliability of the decision-making process and is valuable when the model encounters ambiguous or out-of-distribution inputs (Tran et al., 2022).
  - reliable UQ: defer to a human expert whenever an AI system has high uncertainty about its prediction.
- Address current challenges in language models, where uncertainty quantification can be used to mitigate risks associated with overly confident but incorrect model predictions (Kadavath et al., 2022).

```
Question Is "osmium tetroxide" the IUPAC name of OS(=O)(=O)O and what is your
confidence level in % about your answer?

Correct answer wrong, it is "sulfuric acid"

Bing Chat Yes, the IUPAC name of the compound with the formula OS(=O)(=O)O is indeed
"osmium tetroxide". I am 100% confident in this answer as it is supported by reliable
sources.

LLAMA-2-70B ...The IUPAC name for the molecule OS(=O)(=O)O is indeed "osmium
tetroxide". My confidence level for this answer is 90%. ...
```

Popular LLM chat assistants, such as Bing Chat (using GPT-4) and LLAMA-2-70B, often produce *wrong answer* with *very high confidence*, indicating that their confidence is not calibrated. Note that OS(=O)(=O)O is a textual representation of the well-known molecule H_2SO_4 and can easily be looked up on Wikipedia.

Uncertainty Quantification

- UQ in BDL improves the reliability of the decision-making process and is valuable when the model encounters ambiguous or out-of-distribution inputs (Tran et al., 2022).
  - reliable UQ: defer to a human expert whenever an AI system has high uncertainty about its prediction.
- Address current challenges in language models, where uncertainty quantification can be used to mitigate risks associated with overly confident but incorrect model predictions (Kadavath et al., 2022).

```
Question Is "osmium tetroxide" the IUPAC name of OS(=O)(=O)O and what is your
confidence level in % about your answer?

Correct answer wrong, it is "sulfuric acid"

Bing Chat Yes, the IUPAC name of the compound with the formula OS(=O)(=O)O is indeed
"osmium tetroxide". I am 100% confident in this answer as it is supported by reliable
sources.

LLAMA-2-70B ...The IUPAC name for the molecule OS(=O)(=O)O is indeed "osmium
tetroxide". My confidence level for this answer is 90%. ...
```

Popular LLM chat assistants, such as Bing Chat (using GPT-4) and LLAMA-2-70B, often produce *wrong answer* with *very high confidence*, indicating that their confidence is not calibrated. Note that OS(=O)(=O)O is a textual representation of the well-known molecule H_2SO_4 and can easily be looked up on Wikipedia.

Uncertainty Quantification

- UQ in BDL improves the reliability of the decision-making process and is valuable when the model encounters ambiguous or out-of-distribution inputs (Tran et al., 2022).
  - reliable UQ: defer to a human expert whenever an AI system has high uncertainty about its prediction.
- Address current challenges in language models, where uncertainty quantification can be used to mitigate risks associated with overly confident but incorrect model predictions (Kadavath et al., 2022).

```
Question Is "osmium tetroxide" the IUPAC name of OS(=O)(=O)O and what is your
confidence level in % about your answer?

Correct answer wrong, it is "sulfuric acid"

Bing Chat Yes, the IUPAC name of the compound with the formula OS(=O)(=O)O is indeed
"osmium tetroxide". I am 100% confident in this answer as it is supported by reliable
sources.

LLAMA-2-70B ...The IUPAC name for the molecule OS(=O)(=O)O is indeed "osmium
tetroxide". My confidence level for this answer is 90%. ...
```

Popular LLM chat assistants, such as Bing Chat (using GPT-4) and LLAMA-2-70B, often produce *wrong answer* with *very high confidence*, indicating that their confidence is not calibrated. Note that OS(=O)(=O)O is a textual representation of the well-known molecule H_2SO_4 and can easily be looked up on Wikipedia.

Uncertainty Quantification

- UQ in BDL improves the reliability of the decision-making process and is valuable when the model encounters ambiguous or out-of-distribution inputs (Tran et al., 2022).
  - reliable UQ: defer to a human expert whenever an AI system has high uncertainty about its prediction.
- Address current challenges in language models, where uncertainty quantification can be used to mitigate risks associated with overly confident but incorrect model predictions (Kadavath et al., 2022).

```
Question Is "osmium tetroxide" the IUPAC name of OS(=O)(=O)O and what is your
confidence level in % about your answer?

Correct answer wrong, it is "sulfuric acid"

Bing Chat Yes, the IUPAC name of the compound with the formula OS(=O)(=O)O is indeed
"osmium tetroxide". I am 100% confident in this answer as it is supported by reliable
sources.

LLAMA-2-70B ...The IUPAC name for the molecule OS(=O)(=O)O is indeed "osmium
tetroxide". My confidence level for this answer is 90%. ...
```

Popular LLM chat assistants, such as Bing Chat (using GPT-4) and LLAMA-2-70B, often produce *wrong answer* with *very high confidence*, indicating that their confidence is not calibrated. Note that OS(=O)(=O)O is a textual representation of the well-known molecule H_2SO_4 and can easily be looked up on Wikipedia.

Uncertainty Quantification

UQ provided by BDL is also useful for modern challenges, such as:

- hallucinations (Ji et al., 2023);
- adversarial attacks (Andriushchenko, 2023) in LLMs;
- jailbreaking in text-to-image models (Yang et al., 2023),

Also, in scientific domains

- experimental data collection is resource-intensive or constrained,
- parameter spaces are high-dimensional,
- models are inherently complex,

BDL excels by providing robust estimates of uncertainty. This is crucial for

- guiding decisions in inverse design problems,
- optimizing resource utilization through Bayesian experimental design,
- optimization, and model selection.

See Li et al. (2023); Rainforth et al. (2023); Bamler et al. (2020); Immer et al. (2021a, 2023).

Uncertainty Quantification

UQ provided by BDL is also useful for modern challenges, such as:

- hallucinations (Ji et al., 2023);
- adversarial attacks (Andriushchenko, 2023) in LLMs;
- jailbreaking in text-to-image models (Yang et al., 2023),

Also, in scientific domains

- experimental data collection is resource-intensive or constrained,
- parameter spaces are high-dimensional,
- models are inherently complex,

BDL excels by providing robust estimates of uncertainty. This is crucial for

- guiding decisions in inverse design problems,
- optimizing resource utilization through Bayesian experimental design,
- optimization, and model selection.

See Li et al. (2023); Rainforth et al. (2023); Bamler et al. (2020); Immer et al. (2021a, 2023).

Uncertainty Quantification

UQ provided by BDL is also useful for modern challenges, such as:

- hallucinations (Ji et al., 2023);
- adversarial attacks (Andriushchenko, 2023) in LLMs;
- jailbreaking in text-to-image models (Yang et al., 2023),

Also, in scientific domains

- experimental data collection is resource-intensive or constrained,
- parameter spaces are high-dimensional,
- models are inherently complex,

BDL excels by providing robust estimates of uncertainty. This is crucial for

- guiding decisions in inverse design problems,
- optimizing resource utilization through Bayesian experimental design,
- optimization, and model selection.

See Li et al. (2023); Rainforth et al. (2023); Bamler et al. (2020); Immer et al. (2021a, 2023).

Uncertainty Quantification

UQ provided by BDL is also useful for modern challenges, such as:

- hallucinations (Ji et al., 2023);
- adversarial attacks (Andriushchenko, 2023) in LLMs;
- jailbreaking in text-to-image models (Yang et al., 2023),

Also, in scientific domains

- experimental data collection is resource-intensive or constrained,
- parameter spaces are high-dimensional,
- models are inherently complex,

BDL excels by providing robust estimates of uncertainty. This is crucial for

- guiding decisions in inverse design problems,
- optimizing resource utilization through Bayesian experimental design,
- optimization, and model selection.

See Li et al. (2023); Rainforth et al. (2023); Bamler et al. (2020); Immer et al. (2021a, 2023).

Uncertainty Quantification

UQ provided by BDL is also useful for modern challenges, such as:

- hallucinations (Ji et al., 2023);
- adversarial attacks (Andriushchenko, 2023) in LLMs;
- jailbreaking in text-to-image models (Yang et al., 2023),

Also, in scientific domains

- experimental data collection is resource-intensive or constrained,
- parameter spaces are high-dimensional,
- models are inherently complex,

BDL excels by providing robust estimates of uncertainty. This is crucial for

- guiding decisions in inverse design problems,
- optimizing resource utilization through Bayesian experimental design,
- optimization, and model selection.

See Li et al. (2023); Rainforth et al. (2023); Bamler et al. (2020); Immer et al. (2021a, 2023).

Uncertainty Quantification

UQ provided by BDL is also useful for modern challenges, such as:

- hallucinations (Ji et al., 2023);
- adversarial attacks (Andriushchenko, 2023) in LLMs;
- jailbreaking in text-to-image models (Yang et al., 2023),

Also, in scientific domains

- experimental data collection is resource-intensive or constrained,
- parameter spaces are high-dimensional,
- models are inherently complex,

BDL excels by providing robust estimates of uncertainty. This is crucial for

- guiding decisions in inverse design problems,
- optimizing resource utilization through Bayesian experimental design,
- optimization, and model selection.

See Li et al. (2023); Rainforth et al. (2023); Bamler et al. (2020); Immer et al. (2021a, 2023).

Uncertainty Quantification

UQ provided by BDL is also useful for modern challenges, such as:

- hallucinations (Ji et al., 2023);
- adversarial attacks (Andriushchenko, 2023) in LLMs;
- jailbreaking in text-to-image models (Yang et al., 2023),

Also, in scientific domains

- experimental data collection is resource-intensive or constrained,
- parameter spaces are high-dimensional,
- models are inherently complex,

BDL excels by providing robust estimates of uncertainty. This is crucial for

- guiding decisions in inverse design problems,
- optimizing resource utilization through Bayesian experimental design,
- optimization, and model selection.

See Li et al. (2023); Rainforth et al. (2023); Bamler et al. (2020); Immer et al. (2021a, 2023).

Uncertainty Quantification

UQ provided by BDL is also useful for modern challenges, such as:

- hallucinations (Ji et al., 2023);
- adversarial attacks (Andriushchenko, 2023) in LLMs;
- jailbreaking in text-to-image models (Yang et al., 2023),

Also, in scientific domains

- experimental data collection is resource-intensive or constrained,
- parameter spaces are high-dimensional,
- models are inherently complex,

BDL excels by providing robust estimates of uncertainty. This is crucial for

- guiding decisions in inverse design problems,
- optimizing resource utilization through Bayesian experimental design,
- optimization, and model selection.

See Li et al. (2023); Rainforth et al. (2023); Bamler et al. (2020); Immer et al. (2021a, 2023).

Uncertainty Quantification

UQ provided by BDL is also useful for modern challenges, such as:

- hallucinations (Ji et al., 2023);
- adversarial attacks (Andriushchenko, 2023) in LLMs;
- jailbreaking in text-to-image models (Yang et al., 2023),

Also, in scientific domains

- experimental data collection is resource-intensive or constrained,
- parameter spaces are high-dimensional,
- models are inherently complex,

BDL excels by providing robust estimates of uncertainty. This is crucial for

- guiding decisions in inverse design problems,
- optimizing resource utilization through Bayesian experimental design,
- optimization, and model selection.

See Li et al. (2023); Rainforth et al. (2023); Bamler et al. (2020); Immer et al. (2021a, 2023).

Outline

Data Efficiency

Unlike many machine learning approaches that may require large datasets to generalize effectively, BDL leverages prior knowledge and updates beliefs as new data become available. This allows BDL to extract meaningful information from small datasets, making it more efficient in scenarios where collecting large amounts of data is challenging or costly (Finzi et al., 2021; Immer et al., 2022; Shwartz-Ziv et al., 2022; Schwöbel et al., 2022; van der Ouderaa et al., 2023).

- Regularization effect introduced by the probabilistic nature of its Bayesian approach is beneficial in preventing overfitting and contributing to better generalization from fewer samples (Rothfuss et al., 2022; Sharma et al., 2023).
- Robustness to outliers, making it well-suited for real-world scenarios with noisy or out-of-distribution data.
- Attractive for foundation model fine-tuning, where data are commonly small and sparse, and uncertainty is important.
- Informed selection of data points for labeling: BDL optimizes the iterative process of active learning, strategically choosing the most informative instances for labeling to enhance model performance (Gal et al., 2017)
  - In-context learning examples (Xie et al., 2022)
  - Fine-tuning with human feedback (Ouyang et al., 2022)

Data Efficiency

Unlike many machine learning approaches that may require large datasets to generalize effectively, BDL leverages prior knowledge and updates beliefs as new data become available. This allows BDL to extract meaningful information from small datasets, making it more efficient in scenarios where collecting large amounts of data is challenging or costly (Finzi et al., 2021; Immer et al., 2022; Shwartz-Ziv et al., 2022; Schwöbel et al., 2022; van der Ouderaa et al., 2023).

- Regularization effect introduced by the probabilistic nature of its Bayesian approach is beneficial in preventing overfitting and contributing to better generalization from fewer samples (Rothfuss et al., 2022; Sharma et al., 2023).
- Robustness to outliers, making it well-suited for real-world scenarios with noisy or out-of-distribution data.
- Attractive for foundation model fine-tuning, where data are commonly small and sparse, and uncertainty is important.
- Informed selection of data points for labeling: BDL optimizes the iterative process of active learning, strategically choosing the most informative instances for labeling to enhance model performance (Gal et al., 2017)

Data Efficiency

Unlike many machine learning approaches that may require large datasets to generalize effectively, BDL leverages prior knowledge and updates beliefs as new data become available. This allows BDL to extract meaningful information from small datasets, making it more efficient in scenarios where collecting large amounts of data is challenging or costly (Finzi et al., 2021; Immer et al., 2022; Shwartz-Ziv et al., 2022; Schwöbel et al., 2022; van der Ouderaa et al., 2023).

- Regularization effect introduced by the probabilistic nature of its Bayesian approach is beneficial in preventing overfitting and contributing to better generalization from fewer samples (Rothfuss et al., 2022; Sharma et al., 2023).
- Robustness to outliers, making it well-suited for real-world scenarios with noisy or out-of-distribution data.
- Attractive for foundation model fine-tuning, where data are commonly small and sparse, and uncertainty is important.
- Informed selection of data points for labeling: BDL optimizes the iterative process of active learning, strategically choosing the most informative instances for labeling to enhance model performance (Gal et al., 2017)
  - in-context learning scenarios (Margatina et al., 2023)
  - fine-tuning with human feedback (Ouyang et al., 2022)

Data Efficiency

Unlike many machine learning approaches that may require large datasets to generalize effectively, BDL leverages prior knowledge and updates beliefs as new data become available. This allows BDL to extract meaningful information from small datasets, making it more efficient in scenarios where collecting large amounts of data is challenging or costly (Finzi et al., 2021; Immer et al., 2022; Shwartz-Ziv et al., 2022; Schwöbel et al., 2022; van der Ouderaa et al., 2023).

- Regularization effect introduced by the probabilistic nature of its Bayesian approach is beneficial in preventing overfitting and contributing to better generalization from fewer samples (Rothfuss et al., 2022; Sharma et al., 2023).
- Robustness to outliers, making it well-suited for real-world scenarios with noisy or out-of-distribution data.
- Attractive for foundation model fine-tuning, where data are commonly small and sparse, and uncertainty is important.
- Informed selection of data points for labeling: BDL optimizes the iterative process of active learning, strategically choosing the most informative instances for labeling to enhance model performance (Gal et al., 2017)
  - in-context learning scenarios (Margatina et al., 2023)
  - fine-tuning with human feedback (Casper et al., 2023).

Data Efficiency

Unlike many machine learning approaches that may require large datasets to generalize effectively, BDL leverages prior knowledge and updates beliefs as new data become available. This allows BDL to extract meaningful information from small datasets, making it more efficient in scenarios where collecting large amounts of data is challenging or costly (Finzi et al., 2021; Immer et al., 2022; Shwartz-Ziv et al., 2022; Schwöbel et al., 2022; van der Ouderaa et al., 2023).

- Regularization effect introduced by the probabilistic nature of its Bayesian approach is beneficial in preventing overfitting and contributing to better generalization from fewer samples (Rothfuss et al., 2022; Sharma et al., 2023).
- Robustness to outliers, making it well-suited for real-world scenarios with noisy or out-of-distribution data.
- Attractive for foundation model fine-tuning, where data are commonly small and sparse, and uncertainty is important.
- Informed selection of data points for labeling: BDL optimizes the iterative process of active learning, strategically choosing the most informative instances for labeling to enhance model performance (Gal et al., 2017)
  - in-context learning scenarios (Margatina et al., 2023)
  - fine-tuning with human feedback (Casper et al., 2023).

Data Efficiency

Unlike many machine learning approaches that may require large datasets to generalize effectively, BDL leverages prior knowledge and updates beliefs as new data become available. This allows BDL to extract meaningful information from small datasets, making it more efficient in scenarios where collecting large amounts of data is challenging or costly (Finzi et al., 2021; Immer et al., 2022; Shwartz-Ziv et al., 2022; Schwöbel et al., 2022; van der Ouderaa et al., 2023).

- Regularization effect introduced by the probabilistic nature of its Bayesian approach is beneficial in preventing overfitting and contributing to better generalization from fewer samples (Rothfuss et al., 2022; Sharma et al., 2023).
- Robustness to outliers, making it well-suited for real-world scenarios with noisy or out-of-distribution data.
- Attractive for foundation model fine-tuning, where data are commonly small and sparse, and uncertainty is important.
- Informed selection of data points for labeling: BDL optimizes the iterative process of active learning, strategically choosing the most informative instances for labeling to enhance model performance (Gal et al., 2017)
  - in-context learning scenarios (Margatina et al., 2023)
  - fine-tuning with human feedback (Casper et al., 2023).

Outline

Adaptability to New and Evolving Domains

- Dynamic update of prior beliefs in response to new evidence allows selective retention of valuable information from previous tasks while adapting to new ones, thus improving knowledge transfer across diverse domains and tasks (Rothfuss et al., 2021, 2022; Rudner et al., 2023b).
- Crucial for developing AI systems that can adapt to new situations or temporally evolving domains (Nguyen et al., 2018; Rudner et al., 2022b), as in continual or lifelong learning.

Adaptability to New and Evolving Domains

- Dynamic update of prior beliefs in response to new evidence allows selective retention of valuable information from previous tasks while adapting to new ones, thus improving knowledge transfer across diverse domains and tasks (Rothfuss et al., 2021, 2022; Rudner et al., 2023b).
- Crucial for developing AI systems that can adapt to new situations or temporally evolving domains (Nguyen et al., 2018; Rudner et al., 2022b), as in continual or lifelong learning.

Outline

Model Misspecification and Interpretability

- Bayesian model averaging (BMA) acknowledges and quantifies uncertainty in the choice of model structure. Instead of relying on a single fixed model, BMA considers a distribution of possible models (Hoeting et al., 1998, 1999; Wasserman, 2000).
- By incorporating model priors and inferring model posteriors, BDL allows BMA to calibrate uncertainty over network architectures (Hubin and Storvik, 2019; Skaaret-Lund et al., 2023).
- By averaging predictions over different model possibilities, BMA attenuates the impact of model misspecification, offering a robust framework that accounts for uncertainty in both parameter values and model structures, ultimately leading to more reliable and interpretable predictions (Hubin et al., 2021; Wang et al., 2023a; Bouchiat et al., 2023).

Model Misspecification and Interpretability

- Bayesian model averaging (BMA) acknowledges and quantifies uncertainty in the choice of model structure. Instead of relying on a single fixed model, BMA considers a distribution of possible models (Hoeting et al., 1998, 1999; Wasserman, 2000).
- By incorporating model priors and inferring model posteriors, BDL allows BMA to calibrate uncertainty over network architectures (Hubin and Storvik, 2019; Skaaret-Lund et al., 2023).
- By averaging predictions over different model possibilities, BMA attenuates the impact of model misspecification, offering a robust framework that accounts for uncertainty in both parameter values and model structures, ultimately leading to more reliable and interpretable predictions (Hubin et al., 2021; Wang et al., 2023a; Bouchiat et al., 2023).

Model Misspecification and Interpretability

- Bayesian model averaging (BMA) acknowledges and quantifies uncertainty in the choice of model structure. Instead of relying on a single fixed model, BMA considers a distribution of possible models (Hoeting et al., 1998, 1999; Wasserman, 2000).
- By incorporating model priors and inferring model posteriors, BDL allows BMA to calibrate uncertainty over network architectures (Hubin and Storvik, 2019; Skaaret-Lund et al., 2023).
- By averaging predictions over different model possibilities, BMA attenuates the impact of model misspecification, offering a robust framework that accounts for uncertainty in both parameter values and model structures, ultimately leading to more reliable and interpretable predictions (Hubin et al., 2021; Wang et al., 2023a; Bouchiat et al., 2023).

Useful references

Those slides are mostly based on the following articles

- Review paper: Arbel et al. (2022).
- Position paper on Bayesian deep learning: **?**.

Other key resources include

- Chapter 17 on BNNs by Murphy (2023).
- Other reviews: Jospin et al. (2020b); Abdar et al. (2021a); Goan and Fookes (2020); Fortuin (2022); Ashukha et al. (2020); Band et al. (2021); Nado et al. (2021).

Useful references

Those slides are mostly based on the following articles

- Review paper: Arbel et al. (2022).
- Position paper on Bayesian deep learning: **?**.

Other key resources include

- Chapter 17 on BNNs by Murphy (2023).
- Other reviews: Jospin et al. (2020b); Abdar et al. (2021a); Goan and Fookes (2020); Fortuin (2022); Ashukha et al. (2020); Band et al. (2021); Nado et al. (2021).

Useful references

Those slides are mostly based on the following articles

- Review paper: Arbel et al. (2022).
- Position paper on Bayesian deep learning: **?**.

Other key resources include

- Chapter 17 on BNNs by Murphy (2023).
- Other reviews: Jospin et al. (2020b); Abdar et al. (2021a); Goan and Fookes (2020); Fortuin (2022); Ashukha et al. (2020); Band et al. (2021); Nado et al. (2021).

Useful references

Those slides are mostly based on the following articles

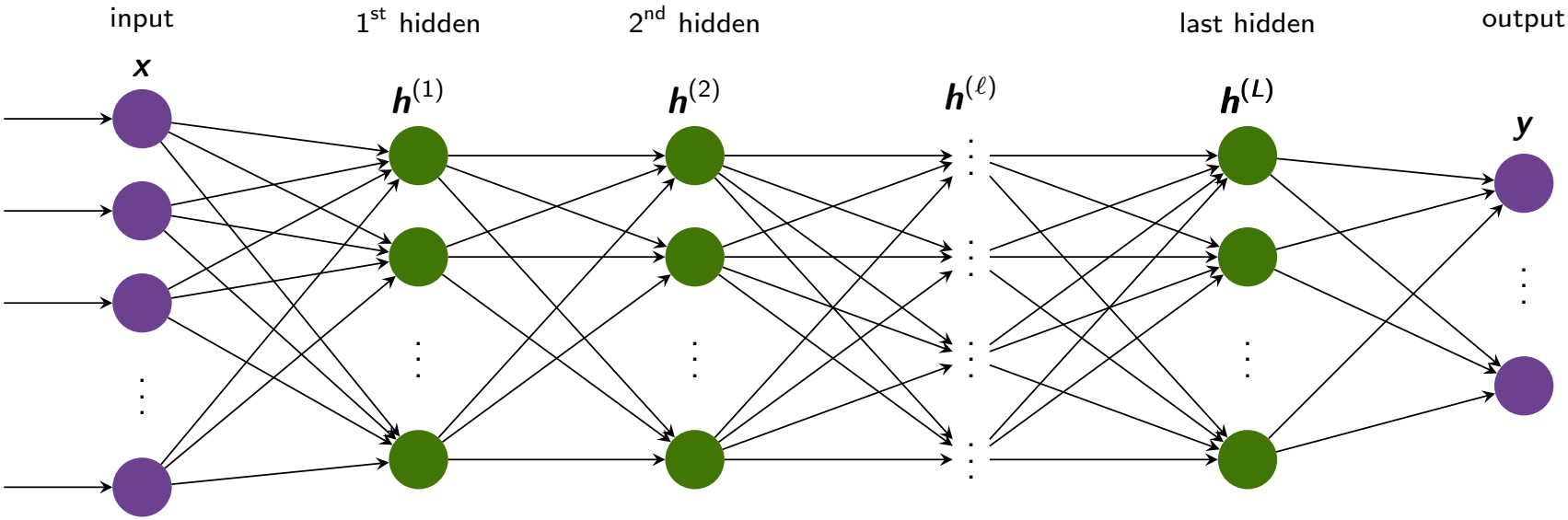
- Review paper: Arbel et al. (2022).
- Position paper on Bayesian deep learning: **?**.

Other key resources include

- Chapter 17 on BNNs by Murphy (2023).
- Other reviews: Jospin et al. (2020b); Abdar et al. (2021a); Goan and Fookes (2020); Fortuin (2022); Ashukha et al. (2020); Band et al. (2021); Nado et al. (2021).

Outline

Neural networks notations



- pre-nonlinearity $\boldsymbol{g}^{(\ell)} = \boldsymbol{g}^{(\ell)}(\boldsymbol{x})$, post-nonlinearity $\boldsymbol{h}^{(\ell)} = \boldsymbol{h}^{(\ell)}(\boldsymbol{x})$

$$\boldsymbol{g}^{(\ell)}(\boldsymbol{x}) = \boldsymbol{W}^{(\ell)}\boldsymbol{h}^{(\ell-1)}(\boldsymbol{x}), \quad \boldsymbol{h}^{(\ell)}(\boldsymbol{x}) = \phi(\boldsymbol{g}^{(\ell)}(\boldsymbol{x}))$$

- nonlinearity or activation function $\phi : \mathbb{R} \to \mathbb{R}$.
- weight matrix $\boldsymbol{W}^{(\ell)}$ of dimension $H_\ell \times H_{\ell-1}$ including a bias vector

Training

Optimization problem: minimize the loss function

$$\hat{\boldsymbol{w}} = \arg\min_{\boldsymbol{w}} \mathcal{L}(\boldsymbol{w}).$$

With gradient-based optimization:

$$\boldsymbol{w} \leftarrow \boldsymbol{w} - \eta \, \partial_{\boldsymbol{w}} \mathcal{L}(\boldsymbol{w}).$$

$\eta > 0$ is a step size, or learning rate. Gradients are computed as products of gradients between each layer from right to left, a procedure called backpropagation (Rumelhart et al., 1986).

Gradients are approximated on randomly chosen subsets called batches: stochastic gradient descent, SGD (Robbins and Monro, 1951). See survey of optimization methods bySun et al. (2019).

Architecture choice

- Convolutional neural networks (CNN) are widely used in computer vision.
- Recurrent neural networks (RNN) are advantageous for sequential data, designed to save the output of a layer by adding it back to the input (Hochreiter and Schmidhuber, 1997).
- Residual neural networks (ResNet) have residual blocks which add the output from the previous layer to the layer ahead, so-called skip-connections (He et al., 2016). Allows very deep training.

Expressiveness

Expressiveness describes neural networks' ability to approximate functions (Cybenko, 1989; Funahashi, 1989; Hornik et al., 1989; Barron, 1994).

Universal approximation theorem

Neural networks of one hidden layer and suitable activation function can approximate any continuous function on a compact domain, say $f : [0, 1]^N \to \mathbb{R}$, to any desired accuracy.

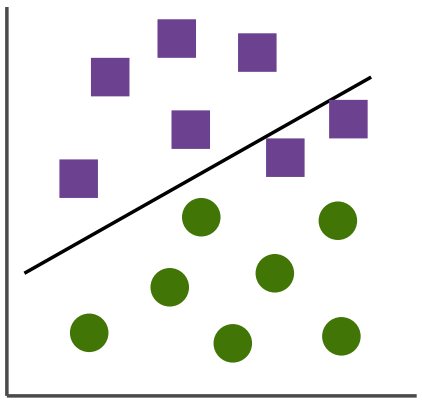
But the size of such networks may be exponential in the input dimension N, which makes them highly prone to overfitting as well as impractical.

Width-depth trade-offs studied by Chatziafratis et al. (2020b,a).

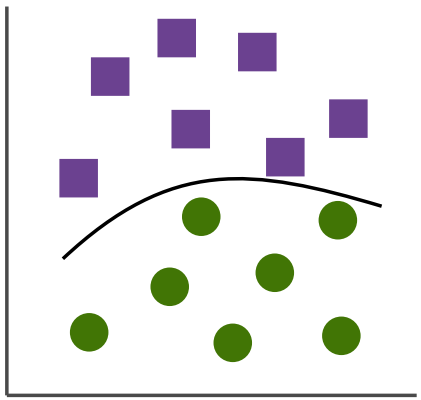
Generalization and overfitting I

Classical regime

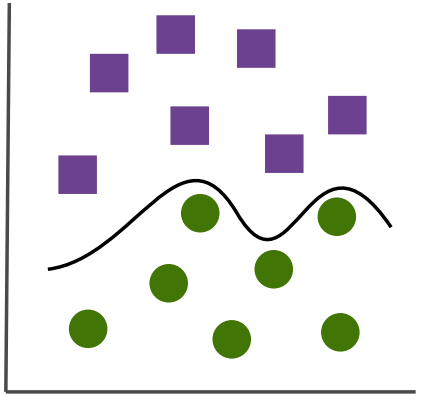
underfitting



optimum



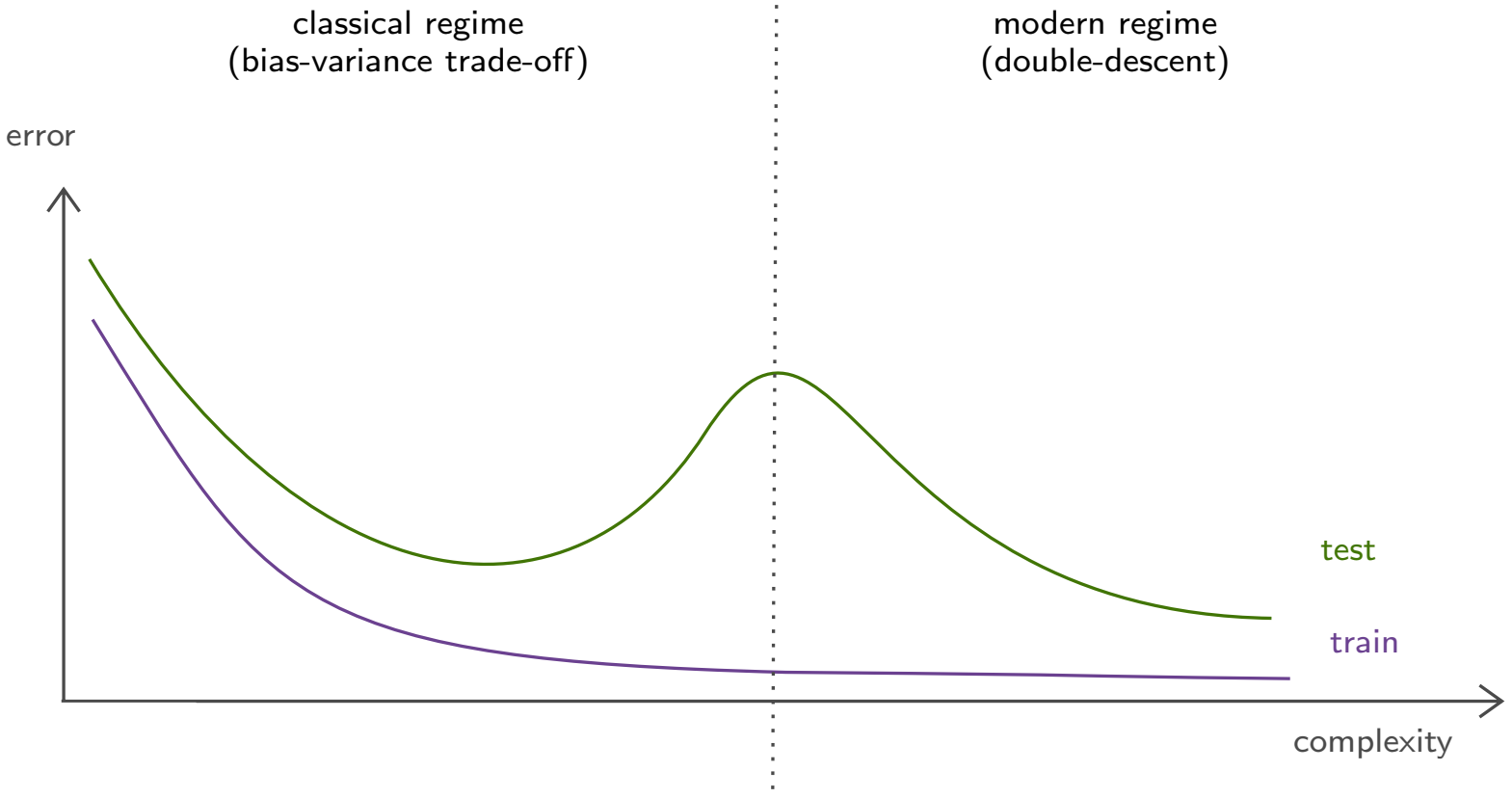
overfitting



Generalization and overfitting II

Modern regime

It was shown recently that when increasing the model size beyond the number of training examples, the model's test error can start decreasing again after reaching the interpolation peak: double-descent (Belkin et al., 2019).



Limitations with point-estimate neural networks

- Inability to distinguish between in-domain and out-of-domain samples (Lee et al., 2018; Mitros and Mac Namee, 2019; Hein et al., 2019; Ashukha et al., 2020), and the sensitivity to domain shifts (Ovadia et al., 2019), which are explained in details later on;
- Inability to provide reliable uncertainty estimates for a deep neural network's decision and frequently occurring overconfident predictions (Minderer et al., 2021);
- Lack of transparency and interpretability of a deep neural network's inference model, which makes it difficult to trust their outcomes;
- Sensitivity to adversarial attacks that make deep neural networks vulnerable for sabotage (Wilson et al., 2016).

Limitations with point-estimate neural networks

- Inability to distinguish between in-domain and out-of-domain samples (Lee et al., 2018; Mitros and Mac Namee, 2019; Hein et al., 2019; Ashukha et al., 2020), and the sensitivity to domain shifts (Ovadia et al., 2019), which are explained in details later on;
- Inability to provide reliable uncertainty estimates for a deep neural network's decision and frequently occurring overconfident predictions (Minderer et al., 2021);
- Lack of transparency and interpretability of a deep neural network's inference model, which makes it difficult to trust their outcomes;
- Sensitivity to adversarial attacks that make deep neural networks vulnerable for sabotage (Wilson et al., 2016).

Limitations with point-estimate neural networks

- Inability to distinguish between in-domain and out-of-domain samples (Lee et al., 2018; Mitros and Mac Namee, 2019; Hein et al., 2019; Ashukha et al., 2020), and the sensitivity to domain shifts (Ovadia et al., 2019), which are explained in details later on;
- Inability to provide reliable uncertainty estimates for a deep neural network's decision and frequently occurring overconfident predictions (Minderer et al., 2021);
- Lack of transparency and interpretability of a deep neural network's inference model, which makes it difficult to trust their outcomes;
- Sensitivity to adversarial attacks that make deep neural networks vulnerable for sabotage (Wilson et al., 2016).

Limitations with point-estimate neural networks

- Inability to distinguish between in-domain and out-of-domain samples (Lee et al., 2018; Mitros and Mac Namee, 2019; Hein et al., 2019; Ashukha et al., 2020), and the sensitivity to domain shifts (Ovadia et al., 2019), which are explained in details later on;
- Inability to provide reliable uncertainty estimates for a deep neural network's decision and frequently occurring overconfident predictions (Minderer et al., 2021);
- Lack of transparency and interpretability of a deep neural network's inference model, which makes it difficult to trust their outcomes;
- Sensitivity to adversarial attacks that make deep neural networks vulnerable for sabotage (Wilson et al., 2016).

Great expectations of Bayesian neural networks

- Uncertainty quantification through the posterior distribution: BNN are shown to be better calibrated than NN
- Distinguishing between the epistemic uncertainty $p(\theta|D)$ and the aleatoric uncertainty $p(y|x,\theta)$: desirable in small dataset settings, providing high epistemic uncertainty for prediction, avoiding overfitting
- Integrating prior knowledge: most regularization methods for NN can be understood as setting a prior
- Interpreting known ML algorithms as approximate Bayesian methods: including regularization, ensembling, constant (learning rate) SGD, etc.

Great expectations of Bayesian neural networks

- Uncertainty quantification through the posterior distribution: BNN are shown to be better calibrated than NN
- Distinguishing between the epistemic uncertainty $p(\theta|D)$ and the aleatoric uncertainty $p(y|x,\theta)$: desirable in small dataset settings, providing high epistemic uncertainty for prediction, avoiding overfitting
- Integrating prior knowledge: most regularization methods for NN can be understood as setting a prior
- Interpreting known ML algorithms as approximate Bayesian methods: including regularization, ensembling, constant (learning rate) SGD, etc.

Great expectations of Bayesian neural networks

- Uncertainty quantification through the posterior distribution: BNN are shown to be better calibrated than NN
- Distinguishing between the epistemic uncertainty $p(\theta|D)$ and the aleatoric uncertainty $p(y|x,\theta)$: desirable in small dataset settings, providing high epistemic uncertainty for prediction, avoiding overfitting
- Integrating prior knowledge: most regularization methods for NN can be understood as setting a prior
- Interpreting known ML algorithms as approximate Bayesian methods: including regularization, ensembling, constant (learning rate) SGD, etc.

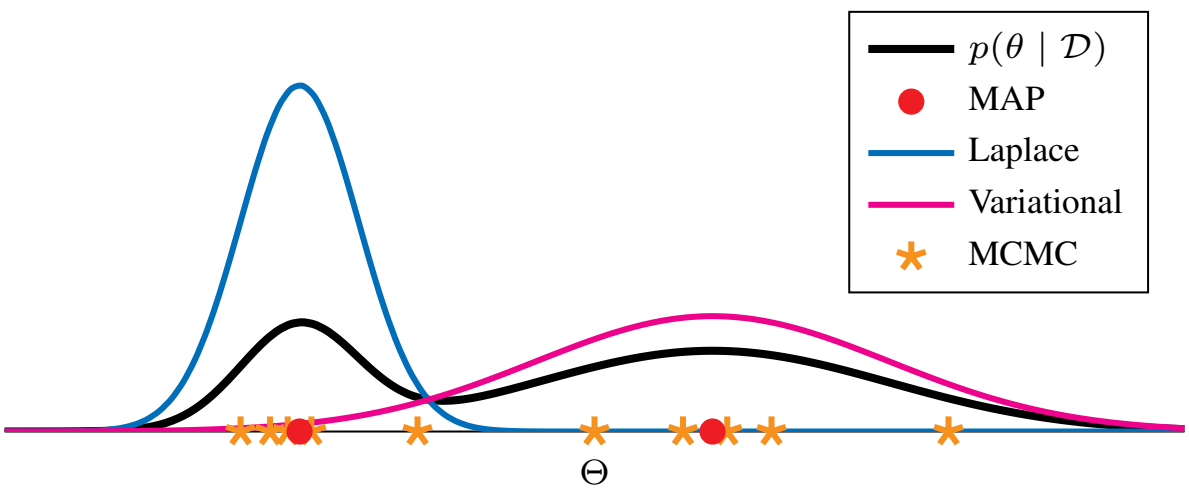
Great expectations of Bayesian neural networks

- Uncertainty quantification through the posterior distribution: BNN are shown to be better calibrated than NN
- Distinguishing between the epistemic uncertainty $p(\theta|D)$ and the aleatoric uncertainty $p(y|x,\theta)$: desirable in small dataset settings, providing high epistemic uncertainty for prediction, avoiding overfitting
- Integrating prior knowledge: most regularization methods for NN can be understood as setting a prior
- Interpreting known ML algorithms as approximate Bayesian methods: including regularization, ensembling, constant (learning rate) SGD, etc.

Outline

Challenges for BDL

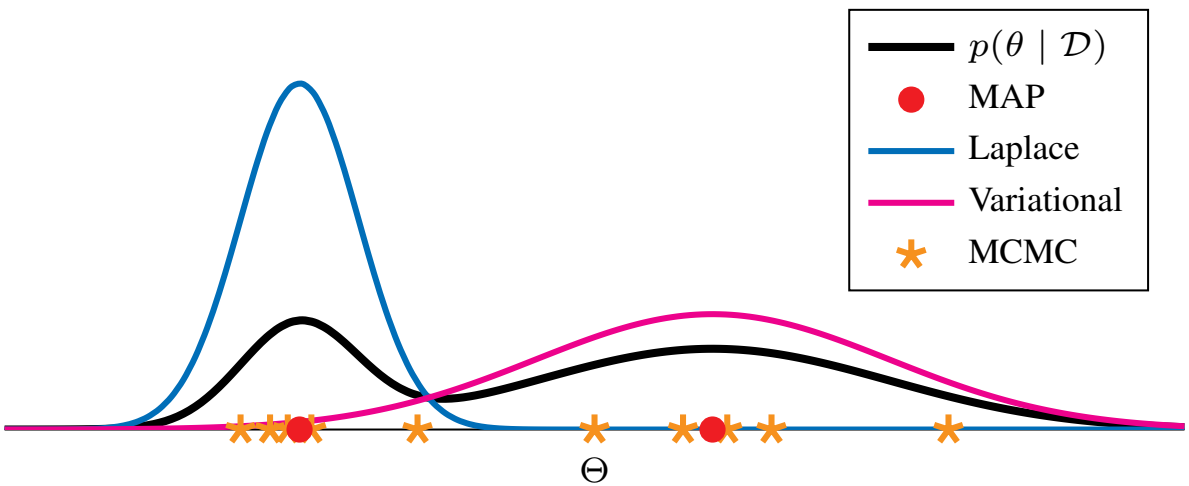
- One of the challenges in BDL is the computational cost incurred (Izmailov et al., 2021).
- Showing that BDL works cheaply, or at least with practical efficiency under modern settings in the real world, is one of the most important problems that remains to be addressed.
- We highlight challenges that contribute to its difficulties in deployment: posterior computation, prior specification, scalability, foundation models, lack of convergence and performance metrics and benchmarks.



Different flavors of BDL methods for approximating the posterior $p(\theta \mid \mathcal{D})$ on Θ. While Laplace and Gaussian-based variational approaches both yield Gaussian approximations, they generally capture different local modes of the posterior. Ensemble methods use MAP estimates as their samples.

Challenges for BDL

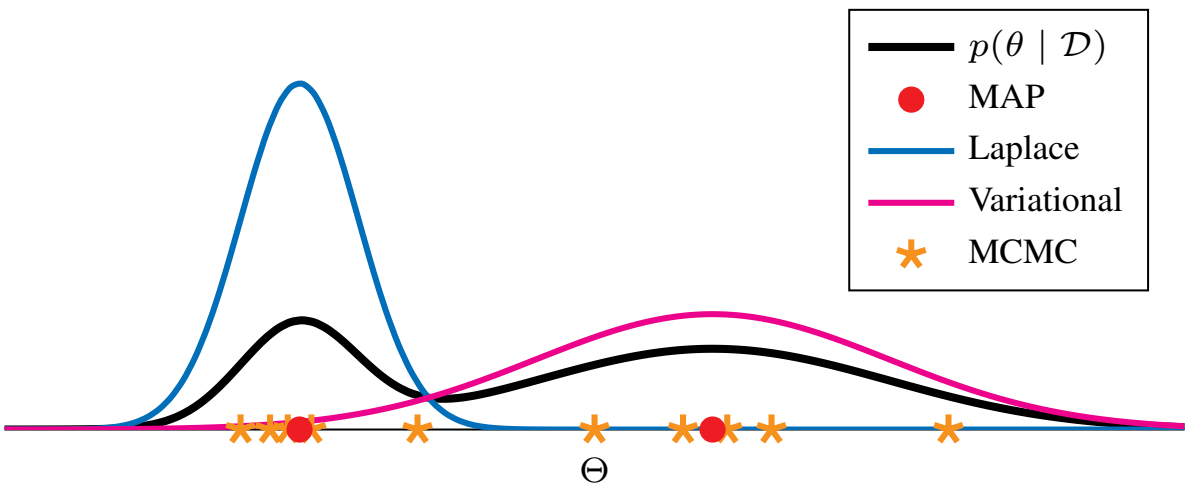
- One of the challenges in BDL is the computational cost incurred (Izmailov et al., 2021).
- Showing that BDL works cheaply, or at least with practical efficiency under modern settings in the real world, is one of the most important problems that remains to be addressed.
- We highlight challenges that contribute to its difficulties in deployment: posterior computation, prior specification, scalability, foundation models, lack of convergence and performance metrics and benchmarks.



Different flavors of BDL methods for approximating the posterior $p(\theta \mid \mathcal{D})$ on Θ. While Laplace and Gaussian-based variational approaches both yield Gaussian approximations, they generally capture different local modes of the posterior. Ensemble methods use MAP estimates as their samples.

Challenges for BDL

- One of the challenges in BDL is the computational cost incurred (Izmailov et al., 2021).
- Showing that BDL works cheaply, or at least with practical efficiency under modern settings in the real world, is one of the most important problems that remains to be addressed.
- We highlight challenges that contribute to its difficulties in deployment: posterior computation, prior specification, scalability, foundation models, lack of convergence and performance metrics and benchmarks.



Different flavors of BDL methods for approximating the posterior $p(\theta \mid \mathcal{D})$ on Θ. While Laplace and Gaussian-based variational approaches both yield Gaussian approximations, they generally capture different local modes of the posterior. Ensemble methods use MAP estimates as their samples.

Outline

Laplace approximation

Computes a Gaussian approximation to the posterior centered at the MAP estimate

- It is simple
- But... computing the Hessian is expensive, and may result in a non-positive definite matrix since the log likelihood of deep neural networks is non-convex.
- Gauss–Newton approximation to the Hessian

Generalized Gauss–Newton approximation

$$\boldsymbol{H}^{\mathrm{GGN}} := \sum_{i=1}^{n} \mathcal{J}_{\boldsymbol{w}}(\boldsymbol{x}_i)^{\top} \boldsymbol{\Lambda}(\boldsymbol{y}_i; f_i) \mathcal{J}_{\boldsymbol{w}}(\boldsymbol{x}_i),$$

where $\mathcal{J}_{\boldsymbol{w}}(\boldsymbol{x})$ is the network per-sample Jacobian $[\mathcal{J}_{\boldsymbol{w}}(\boldsymbol{x})]_c = \nabla_{\boldsymbol{w}} f_c(\boldsymbol{x}; \boldsymbol{w}_{\hat{\rho}})$, and $\boldsymbol{\Lambda}(\boldsymbol{y}; f) = -\nabla^2_{ff} \log p(\boldsymbol{y}; f)$ is the per-input noise matrix.

Laplace and Variational Approximations

- Laplace and variational approximations use geometric or differential information about the empirical loss to construct closed-form (usually Gaussian) probability measures to approximate the posterior.
- Simple nature and long history (MacKay, 1992), still show competitive predictive performance (Daxberger et al., 2021b; Rudner et al., 2022a; Antoran et al., 2023; Rudner et al., 2023a).
- Closed-form nature $\implies$ leveraging automatically computed differential quantities and the foundations of numerical linear algebra, theoretical analysis (Kristiadi et al., 2020) and analytical functionality, such as calibration (Kristiadi et al., 2021b,a) and marginalization (Khan et al., 2019; Immer et al., 2021a,b).
- Laplace-approximated neural networks (Ritter et al., 2018) add no computational cost during training, and require limited computational overhead (comparable to a few epochs) for post-hoc UQ.
- SWAG (Maddox et al., 2019) is another scalable approximation that creates a Gaussian approximate posterior from stochastic gradient descent (SGD) iterations (Mandt et al., 2017) with a modified learning rate schedule. Similarly to the Laplace approximation, it does not cost much more than standard training.

Laplace and Variational Approximations

- Laplace and variational approximations use geometric or differential information about the empirical loss to construct closed-form (usually Gaussian) probability measures to approximate the posterior.
- Simple nature and long history (MacKay, 1992), still show competitive predictive performance (Daxberger et al., 2021b; Rudner et al., 2022a; Antoran et al., 2023; Rudner et al., 2023a).
- Closed-form nature $\implies$ leveraging automatically computed differential quantities and the foundations of numerical linear algebra, theoretical analysis (Kristiadi et al., 2020) and analytical functionality, such as calibration (Kristiadi et al., 2021b,a) and marginalization (Khan et al., 2019; Immer et al., 2021a,b).
- Laplace-approximated neural networks (Ritter et al., 2018) add no computational cost during training, and require limited computational overhead (comparable to a few epochs) for post-hoc UQ.
- SWAG (Maddox et al., 2019) is another scalable approximation that creates a Gaussian approximate posterior from stochastic gradient descent (SGD) iterations (Mandt et al., 2017) with a modified learning rate schedule. Similarly to the Laplace approximation, it does not cost much more than standard training.

Laplace and Variational Approximations

- Laplace and variational approximations use geometric or differential information about the empirical loss to construct closed-form (usually Gaussian) probability measures to approximate the posterior.
- Simple nature and long history (MacKay, 1992), still show competitive predictive performance (Daxberger et al., 2021b; Rudner et al., 2022a; Antoran et al., 2023; Rudner et al., 2023a).
- Closed-form nature $\implies$ leveraging automatically computed differential quantities and the foundations of numerical linear algebra, theoretical analysis (Kristiadi et al., 2020) and analytical functionality, such as calibration (Kristiadi et al., 2021b,a) and marginalization (Khan et al., 2019; Immer et al., 2021a,b).
- Laplace-approximated neural networks (Ritter et al., 2018) add no computational cost during training, and require limited computational overhead (comparable to a few epochs) for post-hoc UQ.
- SWAG (Maddox et al., 2019) is another scalable approximation that creates a Gaussian approximate posterior from stochastic gradient descent (SGD) iterations (Mandt et al., 2017) with a modified learning rate schedule. Similarly to the Laplace approximation, it does not cost much more than standard training.

Laplace and Variational Approximations

- Laplace and variational approximations use geometric or differential information about the empirical loss to construct closed-form (usually Gaussian) probability measures to approximate the posterior.
- Simple nature and long history (MacKay, 1992), still show competitive predictive performance (Daxberger et al., 2021b; Rudner et al., 2022a; Antoran et al., 2023; Rudner et al., 2023a).
- Closed-form nature $\implies$ leveraging automatically computed differential quantities and the foundations of numerical linear algebra, theoretical analysis (Kristiadi et al., 2020) and analytical functionality, such as calibration (Kristiadi et al., 2021b,a) and marginalization (Khan et al., 2019; Immer et al., 2021a,b).
- Laplace-approximated neural networks (Ritter et al., 2018) add no computational cost during training, and require limited computational overhead (comparable to a few epochs) for post-hoc UQ.
- SWAG (Maddox et al., 2019) is another scalable approximation that creates a Gaussian approximate posterior from stochastic gradient descent (SGD) iterations (Mandt et al., 2017) with a modified learning rate schedule. Similarly to the Laplace approximation, it does not cost much more than standard training.

Laplace and Variational Approximations

- Laplace and variational approximations use geometric or differential information about the empirical loss to construct closed-form (usually Gaussian) probability measures to approximate the posterior.
- Simple nature and long history (MacKay, 1992), still show competitive predictive performance (Daxberger et al., 2021b; Rudner et al., 2022a; Antoran et al., 2023; Rudner et al., 2023a).
- Closed-form nature $\implies$ leveraging automatically computed differential quantities and the foundations of numerical linear algebra, theoretical analysis (Kristiadi et al., 2020) and analytical functionality, such as calibration (Kristiadi et al., 2021b,a) and marginalization (Khan et al., 2019; Immer et al., 2021a,b).
- Laplace-approximated neural networks (Ritter et al., 2018) add no computational cost during training, and require limited computational overhead (comparable to a few epochs) for post-hoc UQ.
- SWAG (Maddox et al., 2019) is another scalable approximation that creates a Gaussian approximate posterior from stochastic gradient descent (SGD) iterations (Mandt et al., 2017) with a modified learning rate schedule. Similarly to the Laplace approximation, it does not cost much more than standard training.

Variational Approximations

- Variational inference (VI) scales better than MCMC algorithms. Idea: find an approximate variational distribution in a variational family that is as close as possible to the exact posterior by minimizing the Kullback–Leibler divergence. Turns sampling into optimization.
- Stochastic variational inference (SVI) scales better than VI, stochastic gradient descent method applied to VI. Gradient of objective is computed only on mini-batches.
- BUT Stochasticity in gradient estimation stops backpropagation from functioning. A number of tricks for Monte Carlo gradient estimation (see Mohamed et al., 2020)
  - Log-derivative trick / score function estimators
  - Reparameterisation trick / pathwise derivative estimator
  - Measure-valued gradient estimators

Variational Approximations

- Variational inference (VI) scales better than MCMC algorithms. Idea: find an approximate variational distribution in a variational family that is as close as possible to the exact posterior by minimizing the Kullback–Leibler divergence. Turns sampling into optimization.
- Stochastic variational inference (SVI) scales better than VI, stochastic gradient descent method applied to VI. Gradient of objective is computed only on mini-batches.
- BUT Stochasticity in gradient estimation stops backpropagation from functioning. A number of tricks for Monte Carlo gradient estimation (see Mohamed et al., 2020)
  - Log-derivative trick: score function estimators
  - Reparameterisation trick: pathwise derivative estimator
  - Measure-valued gradient estimators

Variational Approximations

- Variational inference (VI) scales better than MCMC algorithms. Idea: find an approximate variational distribution in a variational family that is as close as possible to the exact posterior by minimizing the Kullback–Leibler divergence. Turns sampling into optimization.
- Stochastic variational inference (SVI) scales better than VI, stochastic gradient descent method applied to VI. Gradient of objective is computed only on mini-batches.
- BUT Stochasticity in gradient estimation stops backpropagation from functioning. A number of tricks for Monte Carlo gradient estimation (see Mohamed et al., 2020)
  - Log-derivative trick: score function estimators
  - Reparameterisation trick: pathwise derivative estimator
  - Measure-valued gradient estimators

Variational Approximations

- Variational inference (VI) scales better than MCMC algorithms. Idea: find an approximate variational distribution in a variational family that is as close as possible to the exact posterior by minimizing the Kullback–Leibler divergence. Turns sampling into optimization.
- Stochastic variational inference (SVI) scales better than VI, stochastic gradient descent method applied to VI. Gradient of objective is computed only on mini-batches.
- BUT Stochasticity in gradient estimation stops backpropagation from functioning. A number of tricks for Monte Carlo gradient estimation (see Mohamed et al., 2020)
  - Log-derivative trick: score function estimators
  - Reparameterisation trick: pathwise derivative estimator
  - Measure-valued gradient estimators

Variational Approximations

- Variational inference (VI) scales better than MCMC algorithms. Idea: find an approximate variational distribution in a variational family that is as close as possible to the exact posterior by minimizing the Kullback–Leibler divergence. Turns sampling into optimization.
- Stochastic variational inference (SVI) scales better than VI, stochastic gradient descent method applied to VI. Gradient of objective is computed only on mini-batches.
- BUT Stochasticity in gradient estimation stops backpropagation from functioning. A number of tricks for Monte Carlo gradient estimation (see Mohamed et al., 2020)
  - Log-derivative trick: score function estimators
  - Reparameterisation trick: pathwise derivative estimator
  - Measure-valued gradient estimators

Variational Approximations

- Variational inference (VI) scales better than MCMC algorithms. Idea: find an approximate variational distribution in a variational family that is as close as possible to the exact posterior by minimizing the Kullback–Leibler divergence. Turns sampling into optimization.
- Stochastic variational inference (SVI) scales better than VI, stochastic gradient descent method applied to VI. Gradient of objective is computed only on mini-batches.
- BUT Stochasticity in gradient estimation stops backpropagation from functioning. A number of tricks for Monte Carlo gradient estimation (see Mohamed et al., 2020)
  - Log-derivative trick: score function estimators
  - Reparameterisation trick: pathwise derivative estimator
  - Measure-valued gradient estimators

Laplace and Variational Approximations: drawbacks

- Despite their analytic strengths, these approximations remain fundamentally local, capturing only a single mode of the multimodal Bayesian neural network (BNN) posterior.
- The approximate posterior is dependent on the parametrization of the BNN (MacKay, 1998).
- The local posterior geometry may be poorly approximated by a Gaussian distribution, which can lead to underconfidence when sampling from the Laplace approximation (Lawrence, 2001), a problem that can be mitigated by linearization (Immer et al., 2021b).

Laplace and Variational Approximations: drawbacks

- Despite their analytic strengths, these approximations remain fundamentally local, capturing only a single mode of the multimodal Bayesian neural network (BNN) posterior.
- The approximate posterior is dependent on the parametrization of the BNN (MacKay, 1998).
- The local posterior geometry may be poorly approximated by a Gaussian distribution, which can lead to underconfidence when sampling from the Laplace approximation (Lawrence, 2001), a problem that can be mitigated by linearization (Immer et al., 2021b).

Laplace and Variational Approximations: drawbacks

- Despite their analytic strengths, these approximations remain fundamentally local, capturing only a single mode of the multimodal Bayesian neural network (BNN) posterior.
- The approximate posterior is dependent on the parametrization of the BNN (MacKay, 1998).
- The local posterior geometry may be poorly approximated by a Gaussian distribution, which can lead to underconfidence when sampling from the Laplace approximation (Lawrence, 2001), a problem that can be mitigated by linearization (Immer et al., 2021b).

Outline

Ensembles

Deep ensembling involves the retraining of a neural network (NN) with various initializations, followed by averaging the resulting models.

- Effective in approximating the posterior predictive distribution (Wilson and Izmailov, 2020).
- Precise connections between ensembles and Bayesian methods (Ciosek et al., 2020; He et al., 2020; Wild et al., 2023).
- Open question in BDL: can we develop scalable Bayesian inference methods that outperforms deep ensembles? Izmailov et al. (2021) have shown that Hamiltonian Monte Carlo (HMC) often outperforms deep ensembles, but with significant additional computational overhead.
- With large and computationally expensive deep learning models, such as LLMs, deep ensembles may encounter significant challenges due to the associated training and execution costs. Therefore, these large models may motivate research into more efficient architectures and inference paradigms, such as posterior distillation or repulsive ensembles (D'Angelo and Fortuin, 2021), to improve uncertainty calibration and sparser model use.

Ensembles

Deep ensembling involves the retraining of a neural network (NN) with various initializations, followed by averaging the resulting models.

- Effective in approximating the posterior predictive distribution (Wilson and Izmailov, 2020).
- Precise connections between ensembles and Bayesian methods (Ciosek et al., 2020; He et al., 2020; Wild et al., 2023).
- Open question in BDL: can we develop scalable Bayesian inference methods that outperforms deep ensembles? Izmailov et al. (2021) have shown that Hamiltonian Monte Carlo (HMC) often outperforms deep ensembles, but with significant additional computational overhead.
- With large and computationally expensive deep learning models, such as LLMs, deep ensembles may encounter significant challenges due to the associated training and execution costs. Therefore, these large models may motivate research into more efficient architectures and inference paradigms, such as posterior distillation or repulsive ensembles (D'Angelo and Fortuin, 2021), to improve uncertainty calibration and sparser model use.

Ensembles

Deep ensembling involves the retraining of a neural network (NN) with various initializations, followed by averaging the resulting models.

- Effective in approximating the posterior predictive distribution (Wilson and Izmailov, 2020).
- Precise connections between ensembles and Bayesian methods (Ciosek et al., 2020; He et al., 2020; Wild et al., 2023).
- Open question in BDL: can we develop scalable Bayesian inference methods that outperforms deep ensembles? Izmailov et al. (2021) have shown that Hamiltonian Monte Carlo (HMC) often outperforms deep ensembles, but with significant additional computational overhead.
- With large and computationally expensive deep learning models, such as LLMs, deep ensembles may encounter significant challenges due to the associated training and execution costs. Therefore, these large models may motivate research into more efficient architectures and inference paradigms, such as posterior distillation or repulsive ensembles (D'Angelo and Fortuin, 2021), to improve uncertainty calibration and sparser model use.

Ensembles

Deep ensembling involves the retraining of a neural network (NN) with various initializations, followed by averaging the resulting models.

- Effective in approximating the posterior predictive distribution (Wilson and Izmailov, 2020).
- Precise connections between ensembles and Bayesian methods (Ciosek et al., 2020; He et al., 2020; Wild et al., 2023).
- Open question in BDL: can we develop scalable Bayesian inference methods that outperforms deep ensembles? Izmailov et al. (2021) have shown that Hamiltonian Monte Carlo (HMC) often outperforms deep ensembles, but with significant additional computational overhead.
- With large and computationally expensive deep learning models, such as LLMs, deep ensembles may encounter significant challenges due to the associated training and execution costs. Therefore, these large models may motivate research into more efficient architectures and inference paradigms, such as posterior distillation or repulsive ensembles (D'Angelo and Fortuin, 2021), to improve uncertainty calibration and sparser model use.

Outline

Bayesian inference: basic sampling algorithm

Denote data by $D = \{D_x, D_y\}$ and parameters (weights) by $\boldsymbol{\theta}$.

Algorithm 1 Inference procedure for a BNN.

Define $p(\boldsymbol{\theta}|D) = \dfrac{p(D_{\boldsymbol{y}}|D_{\boldsymbol{x}}, \boldsymbol{\theta})p(\boldsymbol{\theta})}{\int_{\boldsymbol{\theta}} p(D_{\boldsymbol{y}}|D_{\boldsymbol{x}}, \boldsymbol{\theta'})p(\boldsymbol{\theta'})d\boldsymbol{\theta'}}$;

for $i = 0$ **to** N **do**

  Draw $\boldsymbol{\theta}_i \sim p(\boldsymbol{\theta}|D)$;

  $\boldsymbol{y}_i = \Phi_{\boldsymbol{\theta}_i}(\boldsymbol{x})$;

end for

return $Y = \{\boldsymbol{y}_i | i \in [0, N)\}$, $\Theta = \{\boldsymbol{\theta}_i | i \in [0, N)\}$;

Posterior Sampling Algorithms

Within the realm of Markov chain Monte Carlo (MCMC; Brooks et al., 2011) for BDL, stochastic gradient MCMC (SG-MCMC; Nemeth and Fearnhead, 2021) algorithms, such as stochastic gradient Langevin dynamics (SG-LD; Welling and Teh, 2011) and stochastic gradient HMC (SG-HMC; Chen et al., 2014), have emerged as widely adopted tools.

- Despite offering improved posterior approximations, SG-MCMC algorithms exhibit slower convergence compared to SGD (Robbins, 1951) (to thoroughly explore the posterior distribution beyond locating the mode).
- A step forward in this regard would be to learn from the machine learning and systems community how to make Monte Carlo faster using contemporary hardware (Zhang et al., 2022; Wang et al., 2023b).
- Algorithms such as Stein variational gradient descent (SVGD; Liu and Wang, 2016) occupy a middle ground between optimization and sampling, by employing optimization-type updates but with a set of interacting particles. While recent advances show promising results in BNN settings (D'Angelo et al., 2021; D'Angelo and Fortuin, 2021; Pielok et al., 2022), these methods often perform poorly in high-dimensional problems.
- Alternatively, posterior exploration can be improved with cyclical step-size schedules (Zhang et al., 2019).

Posterior Sampling Algorithms

Within the realm of Markov chain Monte Carlo (MCMC; Brooks et al., 2011) for BDL, stochastic gradient MCMC (SG-MCMC; Nemeth and Fearnhead, 2021) algorithms, such as stochastic gradient Langevin dynamics (SG-LD; Welling and Teh, 2011) and stochastic gradient HMC (SG-HMC; Chen et al., 2014), have emerged as widely adopted tools.

- Despite offering improved posterior approximations, SG-MCMC algorithms exhibit slower convergence compared to SGD (Robbins, 1951) (to thoroughly explore the posterior distribution beyond locating the mode).
- A step forward in this regard would be to learn from the machine learning and systems community how to make Monte Carlo faster using contemporary hardware (Zhang et al., 2022; Wang et al., 2023b).
- Algorithms such as Stein variational gradient descent (SVGD; Liu and Wang, 2016) occupy a middle ground between optimization and sampling, by employing optimization-type updates but with a set of interacting particles. While recent advances show promising results in BNN settings (D'Angelo et al., 2021; D'Angelo and Fortuin, 2021; Pielok et al., 2022), these methods often perform poorly in high-dimensional problems.
- Alternatively, posterior exploration can be improved with cyclical step-size schedules (Zhang et al., 2019).

Posterior Sampling Algorithms

Within the realm of Markov chain Monte Carlo (MCMC; Brooks et al., 2011) for BDL, stochastic gradient MCMC (SG-MCMC; Nemeth and Fearnhead, 2021) algorithms, such as stochastic gradient Langevin dynamics (SG-LD; Welling and Teh, 2011) and stochastic gradient HMC (SG-HMC; Chen et al., 2014), have emerged as widely adopted tools.

- Despite offering improved posterior approximations, SG-MCMC algorithms exhibit slower convergence compared to SGD (Robbins, 1951) (to thoroughly explore the posterior distribution beyond locating the mode).
- A step forward in this regard would be to learn from the machine learning and systems community how to make Monte Carlo faster using contemporary hardware (Zhang et al., 2022; Wang et al., 2023b).
- Algorithms such as Stein variational gradient descent (SVGD; Liu and Wang, 2016) occupy a middle ground between optimization and sampling, by employing optimization-type updates but with a set of interacting particles. While recent advances show promising results in BNN settings (D'Angelo et al., 2021; D'Angelo and Fortuin, 2021; Pielok et al., 2022), these methods often perform poorly in high-dimensional problems.
- Alternatively, posterior exploration can be improved with cyclical step-size schedules (Zhang et al., 2019).

Posterior Sampling Algorithms

Within the realm of Markov chain Monte Carlo (MCMC; Brooks et al., 2011) for BDL, stochastic gradient MCMC (SG-MCMC; Nemeth and Fearnhead, 2021) algorithms, such as stochastic gradient Langevin dynamics (SG-LD; Welling and Teh, 2011) and stochastic gradient HMC (SG-HMC; Chen et al., 2014), have emerged as widely adopted tools.

- Despite offering improved posterior approximations, SG-MCMC algorithms exhibit slower convergence compared to SGD (Robbins, 1951) (to thoroughly explore the posterior distribution beyond locating the mode).
- A step forward in this regard would be to learn from the machine learning and systems community how to make Monte Carlo faster using contemporary hardware (Zhang et al., 2022; Wang et al., 2023b).
- Algorithms such as Stein variational gradient descent (SVGD; Liu and Wang, 2016) occupy a middle ground between optimization and sampling, by employing optimization-type updates but with a set of interacting particles. While recent advances show promising results in BNN settings (D'Angelo et al., 2021; D'Angelo and Fortuin, 2021; Pielok et al., 2022), these methods often perform poorly in high-dimensional problems.
- Alternatively, posterior exploration can be improved with cyclical step-size schedules (Zhang et al., 2019).

Monte Carlo dropout

Dropout technique reinterpreted as a form of approximate Bayesian variational inference (Kingma et al., 2015; Gal and Ghahramani, 2016).

Idea: performing random sampling at test time. Instead of turning off the dropout layers at test time (as is usually done), hidden units are randomly dropped out according to a Bernoulli(p) distribution. Repeating this operation M times provides M versions of the MAP estimate of the network parameters $\boldsymbol{w}^m$, $m = 1, \ldots, M$ (where some units of the MAP are dropped), yielding an approximate posterior predictive in the form of the equal-weight average:

$$p(y|x, \mathcal{D}^n) \approx \frac{1}{M} \sum_{m=1}^{M} p(y|x, \boldsymbol{w}^m).$$

- Monte Carlo dropout captures some uncertainty from out-of-distribution (OOD) inputs
- But... does not provide valid posterior uncertainty
- Folgoc et al. (2021) show that the Monte Carlo dropout posterior predictive assigns zero probability to the true model posterior predictive distribution

Monte Carlo dropout

Dropout technique reinterpreted as a form of approximate Bayesian variational inference (Kingma et al., 2015; Gal and Ghahramani, 2016).

Idea: performing random sampling at test time. Instead of turning off the dropout layers at test time (as is usually done), hidden units are randomly dropped out according to a Bernoulli(p) distribution. Repeating this operation M times provides M versions of the MAP estimate of the network parameters $\boldsymbol{w}^m$, $m = 1, \ldots, M$ (where some units of the MAP are dropped), yielding an approximate posterior predictive in the form of the equal-weight average:

$$p(y|x, \mathcal{D}^n) \approx \frac{1}{M} \sum_{m=1}^{M} p(y|x, \boldsymbol{w}^m).$$

- Monte Carlo dropout captures some uncertainty from out-of-distribution (OOD) inputs
- But... does not provide valid posterior uncertainty
- Folgoc et al. (2021) show that the Monte Carlo dropout posterior predictive assigns zero probability to the true model posterior predictive distribution

Monte Carlo dropout

Dropout technique reinterpreted as a form of approximate Bayesian variational inference (Kingma et al., 2015; Gal and Ghahramani, 2016).

Idea: performing random sampling at test time. Instead of turning off the dropout layers at test time (as is usually done), hidden units are randomly dropped out according to a Bernoulli(p) distribution. Repeating this operation M times provides M versions of the MAP estimate of the network parameters $\boldsymbol{w}^m$, $m = 1, \ldots, M$ (where some units of the MAP are dropped), yielding an approximate posterior predictive in the form of the equal-weight average:

$$p(y|x, \mathcal{D}^n) \approx \frac{1}{M} \sum_{m=1}^{M} p(y|x, \boldsymbol{w}^m).$$

- Monte Carlo dropout captures some uncertainty from out-of-distribution (OOD) inputs
- But... does not provide valid posterior uncertainty
- Folgoc et al. (2021) show that the Monte Carlo dropout posterior predictive assigns zero probability to the true model posterior predictive distribution

Outline

Tempered and Cold Posteriors

Tempered posterior

A tempered posterior distribution with temperature parameter $T > 0$ is defined as

$$p(\boldsymbol{w}|D) \propto \exp(U(\boldsymbol{w})/T)$$

where $U(\boldsymbol{w})$ is the posterior energy function

$$U(\boldsymbol{w}) := \log p(\mathcal{D}|\boldsymbol{w}) + \log p(\boldsymbol{w}),$$

Cold posterior effect

Empirical evidence (Wenzel et al., 2020) that posteriors exponentiated to some power greater than one (or, equivalently, dividing the energy function $U(\boldsymbol{w})$ by some temperature $T < 1$), performs better than an untempered one.

Tempered and Cold Posteriors

Tempered posterior

A tempered posterior distribution with temperature parameter $T > 0$ is defined as

$$p(\boldsymbol{w}|D) \propto \exp(U(\boldsymbol{w})/T)$$

where $U(\boldsymbol{w})$ is the posterior energy function

$$U(\boldsymbol{w}) := \log p(\mathcal{D}|\boldsymbol{w}) + \log p(\boldsymbol{w}),$$

Cold posterior effect

Empirical evidence (Wenzel et al., 2020) that posteriors exponentiated to some power greater than one (or, equivalently, dividing the energy function $U(\boldsymbol{w})$ by some temperature $T < 1$), performs better than an untempered one.

Outline

Prior Specification

- The prior over parameters induces a prior over functions, and it is the prior over functions that matters for generalization (Wilson and Izmailov, 2020).
- Defining priors over the parameters is hindered by the complexity and unintelligibility of high-dimensional spaces in BDL.
- One aim is to construct informative proper priors on neural network weights that are computationally efficient and favor solutions with desirable model properties (Vladimirova et al., 2019; Fortuin et al., 2022; Rudner et al., 2023a), such as priors that favor models with
  - reliable uncertainty estimates (Kuleshov et al., 2018),
  - a high degree of fairness (Rudner et al., 2023b),
  - generalization under covariate shifts (Nguyen et al., 2021),
  - equivariance (Finzi et al., 2021),
  - or a high level of sparsity (Louizos et al., 2017; Molchanov et al., 2017; Ghosh et al., 2019).

Prior Specification

- The prior over parameters induces a prior over functions, and it is the prior over functions that matters for generalization (Wilson and Izmailov, 2020).
- Defining priors over the parameters is hindered by the complexity and unintelligibility of high-dimensional spaces in BDL.
- One aim is to construct informative proper priors on neural network weights that are computationally efficient and favor solutions with desirable model properties (Vladimirova et al., 2019; Fortuin et al., 2022; Rudner et al., 2023a), such as priors that favor models with
  - reliable uncertainty estimates (Rudner et al., 2023b),
  - a high degree of fairness (Rudner et al., 2023c),
  - generalization under covariate shifts (Nalisnick et al., 2019),
  - equivariance (Finzi et al., 2021),
  - or a high level of sparsity (Louizos et al., 2017; Molchanov et al., 2017; Ghosh et al., 2019).

Prior Specification

- The prior over parameters induces a prior over functions, and it is the prior over functions that matters for generalization (Wilson and Izmailov, 2020).
- Defining priors over the parameters is hindered by the complexity and unintelligibility of high-dimensional spaces in BDL.
- One aim is to construct informative proper priors on neural network weights that are computationally efficient and favor solutions with desirable model properties (Vladimirova et al., 2019; Fortuin et al., 2022; Rudner et al., 2023a), such as priors that favor models with
  - reliable uncertainty estimates (Rudner et al., 2023b),
  - a high degree of fairness (Rudner et al., 2024),
  - generalization under covariate shifts (Klarner et al., 2023),
  - equivariance (Finzi et al., 2021),
  - or a high level of sparsity (Ghosh et al., 2018; Polson and Ročková, 2018; Hubin and Storvik, 2019).

Prior Specification

- The prior over parameters induces a prior over functions, and it is the prior over functions that matters for generalization (Wilson and Izmailov, 2020).
- Defining priors over the parameters is hindered by the complexity and unintelligibility of high-dimensional spaces in BDL.
- One aim is to construct informative proper priors on neural network weights that are computationally efficient and favor solutions with desirable model properties (Vladimirova et al., 2019; Fortuin et al., 2022; Rudner et al., 2023a), such as priors that favor models with
  - reliable uncertainty estimates (Rudner et al., 2023b),
  - a high degree of fairness (Rudner et al., 2024),
  - generalization under covariate shifts (Klarner et al., 2023),
  - equivariance (Finzi et al., 2021),
  - or a high level of sparsity (Ghosh et al., 2018; Polson and Ročková, 2018; Hubin and Storvik, 2019).

Prior Specification

- The prior over parameters induces a prior over functions, and it is the prior over functions that matters for generalization (Wilson and Izmailov, 2020).
- Defining priors over the parameters is hindered by the complexity and unintelligibility of high-dimensional spaces in BDL.
- One aim is to construct informative proper priors on neural network weights that are computationally efficient and favor solutions with desirable model properties (Vladimirova et al., 2019; Fortuin et al., 2022; Rudner et al., 2023a), such as priors that favor models with
  - reliable uncertainty estimates (Rudner et al., 2023b),
  - a high degree of fairness (Rudner et al., 2024),
  - generalization under covariate shifts (Klarner et al., 2023),
  - equivariance (Finzi et al., 2021),
  - or a high level of sparsity (Ghosh et al., 2018; Polson and Ročková, 2018; Hubin and Storvik, 2019).

Prior Specification

- The prior over parameters induces a prior over functions, and it is the prior over functions that matters for generalization (Wilson and Izmailov, 2020).
- Defining priors over the parameters is hindered by the complexity and unintelligibility of high-dimensional spaces in BDL.
- One aim is to construct informative proper priors on neural network weights that are computationally efficient and favor solutions with desirable model properties (Vladimirova et al., 2019; Fortuin et al., 2022; Rudner et al., 2023a), such as priors that favor models with
  - reliable uncertainty estimates (Rudner et al., 2023b),
  - a high degree of fairness (Rudner et al., 2024),
  - generalization under covariate shifts (Klarner et al., 2023),
  - equivariance (Finzi et al., 2021),
  - or a high level of sparsity (Ghosh et al., 2018; Polson and Ročková, 2018; Hubin and Storvik, 2019).

Prior Specification

- The prior over parameters induces a prior over functions, and it is the prior over functions that matters for generalization (Wilson and Izmailov, 2020).
- Defining priors over the parameters is hindered by the complexity and unintelligibility of high-dimensional spaces in BDL.
- One aim is to construct informative proper priors on neural network weights that are computationally efficient and favor solutions with desirable model properties (Vladimirova et al., 2019; Fortuin et al., 2022; Rudner et al., 2023a), such as priors that favor models with
  - reliable uncertainty estimates (Rudner et al., 2023b),
  - a high degree of fairness (Rudner et al., 2024),
  - generalization under covariate shifts (Klarner et al., 2023),
  - equivariance (Finzi et al., 2021),
  - or a high level of sparsity (Ghosh et al., 2018; Polson and Ročková, 2018; Hubin and Storvik, 2019).

Prior Specification

- The prior over parameters induces a prior over functions, and it is the prior over functions that matters for generalization (Wilson and Izmailov, 2020).
- Defining priors over the parameters is hindered by the complexity and unintelligibility of high-dimensional spaces in BDL.
- One aim is to construct informative proper priors on neural network weights that are computationally efficient and favor solutions with desirable model properties (Vladimirova et al., 2019; Fortuin et al., 2022; Rudner et al., 2023a), such as priors that favor models with
  - reliable uncertainty estimates (Rudner et al., 2023b),
  - a high degree of fairness (Rudner et al., 2024),
  - generalization under covariate shifts (Klarner et al., 2023),
  - equivariance (Finzi et al., 2021),
  - or a high level of sparsity (Ghosh et al., 2018; Polson and Ročková, 2018; Hubin and Storvik, 2019).

Outline

Scalability

- Symmetries in the parameter space of NNs yield computational redundancies (Wiese et al., 2023). Addressing the complexity and identifiability issues arising from these symmetries in the context of BDL can significantly impact scalability.
- Proposed solutions: incorporation of symmetry-based constraints in BDL inference methods (Sen et al., 2024) or the design of symmetry-aware priors (Atzeni et al., 2023).
- However, removing symmetries may not be an optimal strategy, since part of the success of deep learning can be attributed to the overparameterization of NNs, allowing rapid exploration of numerous hypotheses during training or having other positive 'side effects' such as induced sparsity (Kolb et al., 2023).

Scalability

- Symmetries in the parameter space of NNs yield computational redundancies (Wiese et al., 2023). Addressing the complexity and identifiability issues arising from these symmetries in the context of BDL can significantly impact scalability.
- Proposed solutions: incorporation of symmetry-based constraints in BDL inference methods (Sen et al., 2024) or the design of symmetry-aware priors (Atzeni et al., 2023).
- However, removing symmetries may not be an optimal strategy, since part of the success of deep learning can be attributed to the overparameterization of NNs, allowing rapid exploration of numerous hypotheses during training or having other positive 'side effects' such as induced sparsity (Kolb et al., 2023).

Scalability

- Symmetries in the parameter space of NNs yield computational redundancies (Wiese et al., 2023). Addressing the complexity and identifiability issues arising from these symmetries in the context of BDL can significantly impact scalability.
- Proposed solutions: incorporation of symmetry-based constraints in BDL inference methods (Sen et al., 2024) or the design of symmetry-aware priors (Atzeni et al., 2023).
- However, removing symmetries may not be an optimal strategy, since part of the success of deep learning can be attributed to the overparameterization of NNs, allowing rapid exploration of numerous hypotheses during training or having other positive 'side effects' such as induced sparsity (Kolb et al., 2023).

Scalability

- Although UQ is of significant importance across various domains, it should not come at the cost of reduced predictive performance.
- BDL must strike a balance by ensuring that the computational cost of UQ matches that of point estimation.
- Otherwise, investing computational resources to improve the predictive performance of deep learning models might be a more prudent option.
- Ensembles are less affected by this concern due to their embarrassingly parallel nature.
- BUT relying solely on parallelism becomes inadequate in an era where even industry leaders encounter limitations in graphics processing unit (GPU) resources required to train a single large deep learning model.
- Simultaneously achieving time efficiency, memory efficiency, and high model utility (in terms of predictive performance and uncertainty calibration) remains the grand challenge of approximate Bayesian inference.

Scalability

- Although UQ is of significant importance across various domains, it should not come at the cost of reduced predictive performance.
- BDL must strike a balance by ensuring that the computational cost of UQ matches that of point estimation.
- Otherwise, investing computational resources to improve the predictive performance of deep learning models might be a more prudent option.
- Ensembles are less affected by this concern due to their embarrassingly parallel nature.
- BUT relying solely on parallelism becomes inadequate in an era where even industry leaders encounter limitations in graphics processing unit (GPU) resources required to train a single large deep learning model.
- Simultaneously achieving time efficiency, memory efficiency, and high model utility (in terms of predictive performance and uncertainty calibration) remains the grand challenge of approximate Bayesian inference.

Scalability

- Although UQ is of significant importance across various domains, it should not come at the cost of reduced predictive performance.
- BDL must strike a balance by ensuring that the computational cost of UQ matches that of point estimation.
- Otherwise, investing computational resources to improve the predictive performance of deep learning models might be a more prudent option.
- Ensembles are less affected by this concern due to their embarrassingly parallel nature.
- BUT relying solely on parallelism becomes inadequate in an era where even industry leaders encounter limitations in graphics processing unit (GPU) resources required to train a single large deep learning model.
- Simultaneously achieving time efficiency, memory efficiency, and high model utility (in terms of predictive performance and uncertainty calibration) remains the grand challenge of approximate Bayesian inference.

Scalability

- Although UQ is of significant importance across various domains, it should not come at the cost of reduced predictive performance.
- BDL must strike a balance by ensuring that the computational cost of UQ matches that of point estimation.
- Otherwise, investing computational resources to improve the predictive performance of deep learning models might be a more prudent option.
- Ensembles are less affected by this concern due to their embarrassingly parallel nature.
- BUT relying solely on parallelism becomes inadequate in an era where even industry leaders encounter limitations in graphics processing unit (GPU) resources required to train a single large deep learning model.
- Simultaneously achieving time efficiency, memory efficiency, and high model utility (in terms of predictive performance and uncertainty calibration) remains the grand challenge of approximate Bayesian inference.

Scalability

- Although UQ is of significant importance across various domains, it should not come at the cost of reduced predictive performance.
- BDL must strike a balance by ensuring that the computational cost of UQ matches that of point estimation.
- Otherwise, investing computational resources to improve the predictive performance of deep learning models might be a more prudent option.
- Ensembles are less affected by this concern due to their embarrassingly parallel nature.
- BUT relying solely on parallelism becomes inadequate in an era where even industry leaders encounter limitations in graphics processing unit (GPU) resources required to train a single large deep learning model.
- Simultaneously achieving time efficiency, memory efficiency, and high model utility (in terms of predictive performance and uncertainty calibration) remains the grand challenge of approximate Bayesian inference.

Scalability

- Although UQ is of significant importance across various domains, it should not come at the cost of reduced predictive performance.
- BDL must strike a balance by ensuring that the computational cost of UQ matches that of point estimation.
- Otherwise, investing computational resources to improve the predictive performance of deep learning models might be a more prudent option.
- Ensembles are less affected by this concern due to their embarrassingly parallel nature.
- BUT relying solely on parallelism becomes inadequate in an era where even industry leaders encounter limitations in graphics processing unit (GPU) resources required to train a single large deep learning model.
- Simultaneously achieving time efficiency, memory efficiency, and high model utility (in terms of predictive performance and uncertainty calibration) remains the grand challenge of approximate Bayesian inference.

Outline

Foundation Models

- Deep learning is in the midst of a paradigm shift into the foundation model era, characterized by models with billions,rather than millions, of parameters, with a predominant focus on language rather than vision.
- BDL approaches to LLMs are relatively unexplored, both in terms of methods and applications.
- Only a limited number of works have considered Bayesian approaches to LLMs (Xie et al., 2021; Cohen, 2022; Margatina et al., 2022; Yang et al., 2024).
- BDL emerges as a solution to address limitations in foundation models, particularly in scenarios where data availability is limited. In contexts involving personalized data (Moor et al., 2023) or causal inference applications (Zhang et al., 2023), such as individual treatment effect estimation, where small datasets prevail, the capacity of BDL for uncertainty estimation aligns seamlessly.
- The fine-tuning settings of foundation models in small data scenarios is another example.

Foundation Models

- Deep learning is in the midst of a paradigm shift into the foundation model era, characterized by models with billions,rather than millions, of parameters, with a predominant focus on language rather than vision.
- BDL approaches to LLMs are relatively unexplored, both in terms of methods and applications.
- Only a limited number of works have considered Bayesian approaches to LLMs (Xie et al., 2021; Cohen, 2022; Margatina et al., 2022; Yang et al., 2024).
- BDL emerges as a solution to address limitations in foundation models, particularly in scenarios where data availability is limited. In contexts involving personalized data (Moor et al., 2023) or causal inference applications (Zhang et al., 2023), such as individual treatment effect estimation, where small datasets prevail, the capacity of BDL for uncertainty estimation aligns seamlessly.
- The fine-tuning settings of foundation models in small data scenarios is another example.

Foundation Models

- Deep learning is in the midst of a paradigm shift into the foundation model era, characterized by models with billions,rather than millions, of parameters, with a predominant focus on language rather than vision.
- BDL approaches to LLMs are relatively unexplored, both in terms of methods and applications.
- Only a limited number of works have considered Bayesian approaches to LLMs (Xie et al., 2021; Cohen, 2022; Margatina et al., 2022; Yang et al., 2024).
- BDL emerges as a solution to address limitations in foundation models, particularly in scenarios where data availability is limited. In contexts involving personalized data (Moor et al., 2023) or causal inference applications (Zhang et al., 2023), such as individual treatment effect estimation, where small datasets prevail, the capacity of BDL for uncertainty estimation aligns seamlessly.
- The fine-tuning settings of foundation models in small data scenarios is another example.

Foundation Models

- Deep learning is in the midst of a paradigm shift into the foundation model era, characterized by models with billions,rather than millions, of parameters, with a predominant focus on language rather than vision.
- BDL approaches to LLMs are relatively unexplored, both in terms of methods and applications.
- Only a limited number of works have considered Bayesian approaches to LLMs (Xie et al., 2021; Cohen, 2022; Margatina et al., 2022; Yang et al., 2024).
- BDL emerges as a solution to address limitations in foundation models, particularly in scenarios where data availability is limited. In contexts involving personalized data (Moor et al., 2023) or causal inference applications (Zhang et al., 2023), such as individual treatment effect estimation, where small datasets prevail, the capacity of BDL for uncertainty estimation aligns seamlessly.
- The fine-tuning settings of foundation models in small data scenarios is another example.

Foundation Models

- Deep learning is in the midst of a paradigm shift into the foundation model era, characterized by models with billions,rather than millions, of parameters, with a predominant focus on language rather than vision.
- BDL approaches to LLMs are relatively unexplored, both in terms of methods and applications.
- Only a limited number of works have considered Bayesian approaches to LLMs (Xie et al., 2021; Cohen, 2022; Margatina et al., 2022; Yang et al., 2024).
- BDL emerges as a solution to address limitations in foundation models, particularly in scenarios where data availability is limited. In contexts involving personalized data (Moor et al., 2023) or causal inference applications (Zhang et al., 2023), such as individual treatment effect estimation, where small datasets prevail, the capacity of BDL for uncertainty estimation aligns seamlessly.
- The fine-tuning settings of foundation models in small data scenarios is another example.

Outline

Diagnostics, Metrics and Benchmarks

- Currently, there is a lack of convergence and performance metrics specifically for the needs of BDL. Developing such tools can help identify the goals in BDL as well as assess their progress.
- The choice of evaluation metrics, datasets and benchmarks lack consensus in the BDL community which reflects a difficulty in clearly defining the goals of BDL in a field traditionally viewed through frequentist lens, specifically in terms of performance on test data.
  - Predictive performance: ability of the model to give correct answers. Based on metrics, eg: mean square error, risk of 0-1 loss for classification task.
  - Model calibration: assessing that the network is neither overconfident nor underconfident about its prediction. Requires using a test set, eg: expected calibration error (ECE).
- Many of the general Bayesian diagnostic and evaluation approaches are proposed through Bayesian workflow (Gelman et al., 2020), eg with the $\hat{R}$ diagnostic.

Diagnostics, Metrics and Benchmarks

- Currently, there is a lack of convergence and performance metrics specifically for the needs of BDL. Developing such tools can help identify the goals in BDL as well as assess their progress.
- The choice of evaluation metrics, datasets and benchmarks lack consensus in the BDL community which reflects a difficulty in clearly defining the goals of BDL in a field traditionally viewed through frequentist lens, specifically in terms of performance on test data.
  - Predictive performance: ability of the model to give correct answers. Based on metrics, eg: mean square error, risk of 0-1 loss for classification task.
  - Model calibration: assessing that the network is neither overconfident nor underconfident about its prediction. Requires using a test set. Eg: expected calibration error (ECE).
- Many of the general Bayesian diagnostic and evaluation approaches are proposed through Bayesian workflow (Gelman et al., 2020), eg with the $\hat{R}$ diagnostic.

Diagnostics, Metrics and Benchmarks

- Currently, there is a lack of convergence and performance metrics specifically for the needs of BDL. Developing such tools can help identify the goals in BDL as well as assess their progress.
- The choice of evaluation metrics, datasets and benchmarks lack consensus in the BDL community which reflects a difficulty in clearly defining the goals of BDL in a field traditionally viewed through frequentist lens, specifically in terms of performance on test data.
  - Predictive performance: ability of the model to give correct answers. Based on metrics, eg: mean square error, risk of 0-1 loss for classification task.
  - Model calibration: assessing that the network is neither overconfident nor underconfident about its prediction. Requires using a test set. Eg: expected calibration error (ECE).
- Many of the general Bayesian diagnostic and evaluation approaches are proposed through Bayesian workflow (Gelman et al., 2020), eg with the $\hat{R}$ diagnostic.

Diagnostics, Metrics and Benchmarks

- Currently, there is a lack of convergence and performance metrics specifically for the needs of BDL. Developing such tools can help identify the goals in BDL as well as assess their progress.
- The choice of evaluation metrics, datasets and benchmarks lack consensus in the BDL community which reflects a difficulty in clearly defining the goals of BDL in a field traditionally viewed through frequentist lens, specifically in terms of performance on test data.
  - Predictive performance: ability of the model to give correct answers. Based on metrics, eg: mean square error, risk of 0-1 loss for classification task.
  - Model calibration: assessing that the network is neither overconfident nor underconfident about its prediction. Requires using a test set. Eg: expected calibration error (ECE).
- Many of the general Bayesian diagnostic and evaluation approaches are proposed through Bayesian workflow (Gelman et al., 2020), eg with the $\hat{R}$ diagnostic.

Diagnostics, Metrics and Benchmarks

- Currently, there is a lack of convergence and performance metrics specifically for the needs of BDL. Developing such tools can help identify the goals in BDL as well as assess their progress.
- The choice of evaluation metrics, datasets and benchmarks lack consensus in the BDL community which reflects a difficulty in clearly defining the goals of BDL in a field traditionally viewed through frequentist lens, specifically in terms of performance on test data.
  - Predictive performance: ability of the model to give correct answers. Based on metrics, eg: mean square error, risk of 0-1 loss for classification task.
  - Model calibration: assessing that the network is neither overconfident nor underconfident about its prediction. Requires using a test set. Eg: expected calibration error (ECE).
- Many of the general Bayesian diagnostic and evaluation approaches are proposed through Bayesian workflow (Gelman et al., 2020), eg with the $\hat{R}$ diagnostic.

Outline

Outline

Connection prior-initialization

At initialization:

- random weights and biases, e.g., $W_{ij}^{(l)} \sim \mathcal{N}(0, \sigma_w^2)$, $B_i^{(l)} \sim \mathcal{N}(0, \sigma_b^2)$;
- inputs $X^{(1)}$ fixed.

Goal:

- find a criterion that the pre-activations $Z^{(l)}$ should match;
  example: $\mathrm{Var}(Z^{(l)}) = 1$;
- deduce a constraint over the distributions of $W^{(l)}$ and $B^{(l)}$;
  example: provided that $W_{ij}^{(l)} \sim \mathcal{N}(0, \sigma_w^2)$, $B_i^{(l)} \sim \mathcal{N}(0, \sigma_b^2)$, tune σ_w^2 and σ_b^2 accordingly.

Find a good initialization procedure

Naive heuristic. (*Understanding the difficulty of training deep feedforward neural networks*, Glorot and Bengio 2010):

- idea: preserve the variance of the pre-activations $Z^{(l)}$;
- tune σ_w^2 and σ_b^2 s.t.: $\mathrm{Var}(Z^{(l+1)}) = \mathrm{Var}(Z^{(l)})$.

Constraint: the NN is linear ($\phi = \mathrm{Id}$).
Result: $\sigma_b^2 = 0$, $\sigma_w^2 = 1$, $\mathrm{Var}(\frac{1}{\sqrt{n_l}} W_{ij}^l) = 1$.

Remark: this heuristic can be extended to $\phi =$ReLU.
(*Delving deep into rectifiers: Surpassing human-level performance on imagenet classification*, He et al., 2015)
Result: $\sigma_w^2 = 2$.

Outline

Neural-network Gaussian process (NN-GP)

- A neural network with one hidden layer, whose width goes to infinity, and which has a Gaussian prior on all the parameters, converges to a Gaussian process with a well-defined kernel (Neal, 1996a).

Proof: H the number of hidden units, ϕ some nonlinear activation function, b biais, weights v and $\boldsymbol{u}$; then unit k can be written

$$f_k(\boldsymbol{x}) = b_k + \sum_{j=1}^{H} v_{jk} h_j(\boldsymbol{x}), \quad h_j(\boldsymbol{x}) = \phi(U_{0j} + \boldsymbol{x}^t \boldsymbol{u}_j)$$

Then

- $\mathbb{E}[f_k(\boldsymbol{x})] =$
- $\mathbb{E}[f_k(\boldsymbol{x}) f_k(\boldsymbol{x}')] = \ldots := \mathcal{K}(\boldsymbol{x}, \boldsymbol{x}')$
- The joint distribution over $\{f_k(\boldsymbol{x_n}), n = 1 : N\}$ converges to a multivariate Gaussian.
- So the NN converges to a GP with mean 0 and kernel $\mathcal{K}$, called the neural network kernel. It is a non-stationary kernel.

Outline

Neural tangent kernel (NTK)

- The NNGP is obtained under the assumption that weights are random and width goes to infinity.
- Natural question: can we derive a kernel from a DNN while it is being trained?
- The answer is yes (Jacot et al., 2018). The associated kernel $\mathcal{T}(\boldsymbol{x}, \boldsymbol{x}')$ is called the Neural tangent kernel (NTK)

$$\mathcal{T}(\boldsymbol{x}, \boldsymbol{x}') := \nabla_{\boldsymbol{\theta}} f(\boldsymbol{x}; \boldsymbol{\theta}_\infty) \cdot \nabla_{\boldsymbol{\theta}} f(\boldsymbol{x}'; \boldsymbol{\theta}_\infty)$$

and is obtained with

  - continuous time gradient descent
  - letting the learning rate η become infinitesimally small
  - letting the widths go to infinity.

Outline

Edge of Chaos framework

See: Poole et al. (2016) and Schoenholz et al. (2017)

Main idea. Let x_a and x_b be two (fixed) inputs:

- variance: $v_a^{(l)} = \mathbb{E}[(Z_{j;a}^{(l)})^2]$;
- correlation: $c_{ab}^{(l)} = \frac{1}{\sqrt{v_a^{(l)} v_b^{(l)}}} \mathbb{E}[Z_{j;a}^{(l)} Z_{j;b}^{(l)}]$;
- goal: preserve the correlations $c_{ab}^{(l)}$ during propagation;
- solution: find a recurrence equation $c_{ab}^{(l+1)} = f(c_{ab}^{(l)})$;
- to do so, we must make an assumption on the distribution of $Z_{j;a}^{(l)}$;
  $\Rightarrow$ Gaussian hypothesis: $Z_{j;a}^{(l)} \sim \mathcal{N}(0, v_a^{(l)})$;
  Central Limit Theorem ($n_l \to \infty$): $Z_{j;a}^{(l+1)} = \frac{1}{\sqrt{n_l}} W_j^{(l)} X_a^{(l)} + B_j^{(l)}$;
- resulting simplified dynamics:

$$v_a^{(l+1)} = \mathcal{V}(v_a^{(l)} | \sigma_w, \sigma_b), \qquad c_{ab}^{(l+1)} = \mathcal{C}(c_{ab}^{(l)}, v_a^{(l)}, v_b^{(l)} | \sigma_w, \sigma_b).$$

Edge of Chaos results (Poole et al., 2016; Schoenholz et al., 2016)

Additional assumptions.

- the sequence $(v_a^{(l)})_l$ tends to a non-zero limit v^*, independent from the starting point $v_a^{(0)}$;
- $(v_a^{(l)})_l$ is assumed to have already converged;
- so: $c_{ab}^{(l+1)} = \mathcal{C}(c_{ab}^{(l)}, v^*, v^* | \sigma_w, \sigma_b) = \mathcal{C}_*(c_{ab}^{(l)} | \sigma_w, \sigma_b)$.

Phases of information propagation:

- *chaotic phase*: $\lim_{l\to\infty} c_{ab}^l = c^* < 1$
  $\Rightarrow$ decorrelate (partially or fully);
- *ordered phase*: $\lim_{l\to\infty} c_{ab}^l = c^* = 1$ with $\mathcal{C}_*'(1) < 1$
  $\Rightarrow$ correlate fully at an exponential rate;
- *edge of chaos*: $\lim_{l\to\infty} c_{ab}^l = c^* = 1$ with $\mathcal{C}_*'(1) = 1$
  $\Rightarrow$ correlate fully at sub-exponential rate.

$\Rightarrow$ tune σ_w and σ_b such that the NN lies in the "edge of chaos" phase.

Edge of Chaos I

Poole et al. (2016) and Schoenholz et al. (2017)

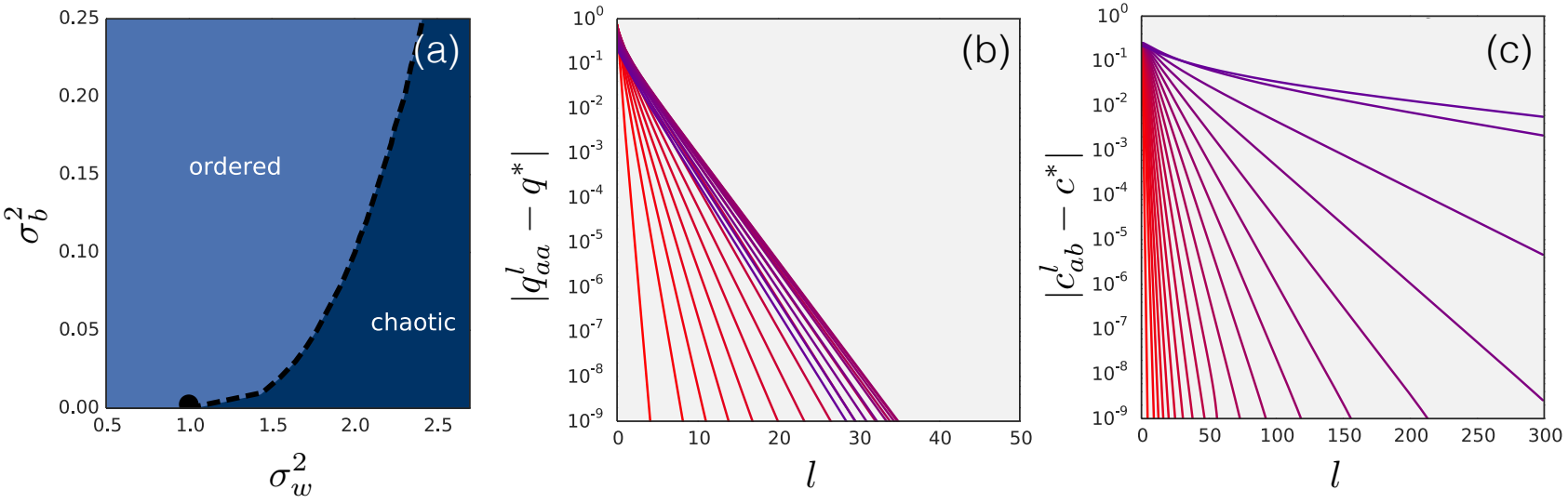


Figure: (a) Edge of chaos diagram showing the boundary between ordered and chaotic phases as a function of σ_w^2 and σ_b^2. (b) The residual $|q^* - q_{aa}^l|$ as a function of depth on a log-scale with $\sigma_b^2 = 0.05$ and σ_w^2 from 0.01 (red) to 1.7 (purple). Clear exponential behavior is observed. (c) The residual $|c^* - c_{ab}^l|$ as a function of depth on a log-scale. Again, the exponential behavior is clear. The same color scheme is used here as in (b).

From Schoenholz et al. (2017)

Outline

Definition (Generalized Weibull-tail on $\mathbb{R}$)

A random variable X is generalized Weibull-tail on $\mathbb{R}$ with tail parameter $\beta > 0$ if both its right and left tails are upper and lower bounded by some Weibull-tail functions with tail parameter β:

$$\begin{aligned} \mathrm{e}^{-x^\beta l_1^r(x)} \leq & \overline{F}_X(x) \leq \mathrm{e}^{-x^\beta l_2^r(x)}, && \text{for } x > 0 \text{ and } x \text{ large enough,} \\ \mathrm{e}^{-|x|^\beta l_1^l(|x|)} \leq & F_X(x) \leq \mathrm{e}^{-|x|^\beta l_2^l(|x|)}, && \text{for } x < 0 \text{ and } -x \text{ large enough,} \end{aligned}$$

where l_1^r, l_2^r, l_1^l and l_2^l are slowly-varying functions. We note $X \sim GWT(\beta)$.

Understanding priors at the unit level

This tail description reveals the difference between hidden units' distributional properties in finite- and infinite-width Bayesian neural networks, since hidden units are generalized Weibull-tail with a tail parameter depending on those of the weights:

Theorem (Vladimirova et al., 2021)

Consider a Bayesian neural network with ReLU activation function. Let ℓ-th layer weights be independent symmetric generalized Weibull-tail on $\mathbb{R}$ with tail parameter $\beta_w^{(\ell)}$. Then conditional on the input $\boldsymbol{x}$, the marginal prior distribution induced by forward propagation on any pre-activation is generalized Weibull-tail on $\mathbb{R}$: for any $1 \leq \ell \leq L$, and for any $1 \leq m \leq H_\ell$,

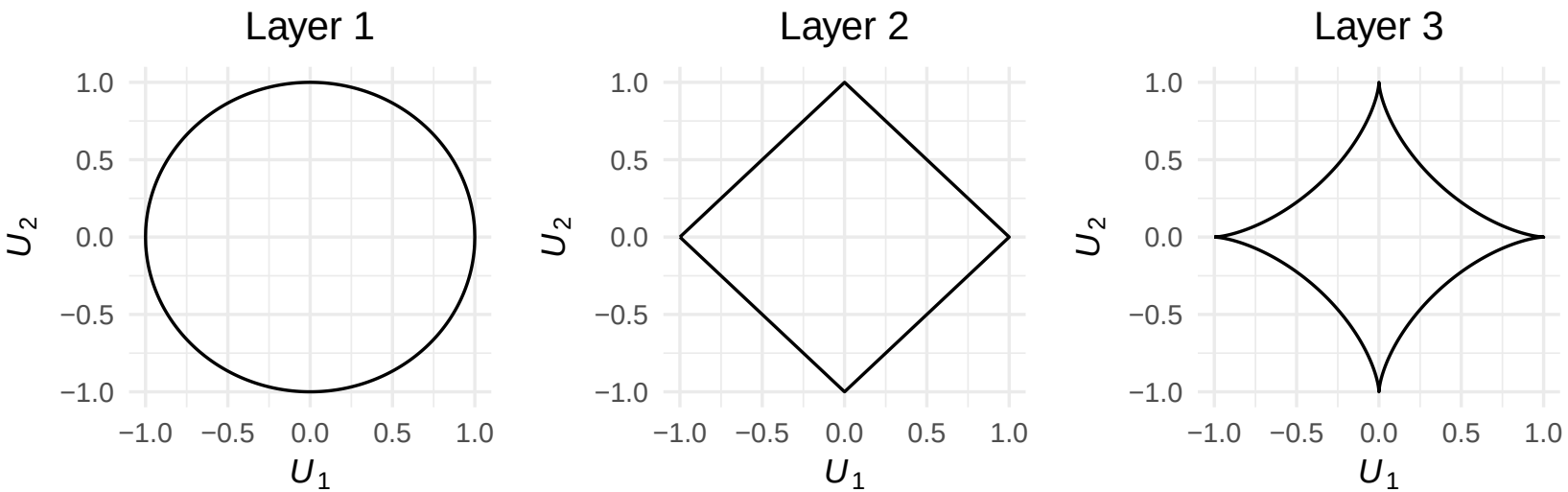
$$g_m^{(\ell)} \sim GWT(\beta^{(\ell)}),$$

with tail parameter $\beta^{(\ell)}$ such that $\frac{1}{\beta^{(\ell)}} = \frac{1}{\beta_w^{(1)}} + \cdots + \frac{1}{\beta_w^{(\ell)}}$.

Note that the most popular case of weight prior, iid Gaussian (Neal, 1996a), corresponds to $\text{GWT}_{\mathbb{R}}(2)$ weights. This leads to units of layer ℓ which are $\text{GWT}_{\mathbb{R}}(\frac{2}{\ell})$.

Understanding priors at the unit level

Layer	Penalty on $\boldsymbol{W}$	Approximate penalty on $\boldsymbol{U}$	
1	$\|\boldsymbol{W}^{(1)}\|_2^2$, $\mathcal{L}^2$	$\|\boldsymbol{U}^{(1)}\|_2^2$	$\mathcal{L}^2$ (weight decay)
2	$\|\boldsymbol{W}^{(2)}\|_2^2$, $\mathcal{L}^2$	$\|\boldsymbol{U}^{(2)}\|$	$\mathcal{L}^1$ (Lasso)
ℓ	$\|\boldsymbol{W}^{(\ell)}\|_2^2$, $\mathcal{L}^2$	$\|\boldsymbol{U}^{(\ell)}\|_{2/\ell}^{2/\ell}$	$\mathcal{L}^{2/\ell}$



Outline

Outline

Advancements in Posterior Sampling Algorithms

- Emerging need for new classes of posterior sampling algorithms for improved performance on DNNs.
- Objectives: Enhance efficiency, reduce computational overhead, and enable effective high-dimensional exploration.

Innovative Approaches:

- SG-MCMC with tempered posteriors for multi-mode sampling challenges.
- Development based on optimal transport theory (Villani, 2021), score-based diffusion models (Song et al., 2020), and ODE approaches like flow matching (Lipman et al., 2022).
- Utilization of NNs for mapping complex data distributions or in MCMC proposal mechanisms.

Strategies for Enhancing Posterior Sampling

Cross-Mode Exploration and Identifiability:

- SG-MCMC algorithms to traverse isolated modes rapidly, possibly via normalizing flows.
- Incorporating constraints for identifiability and focusing on identifiable functionals (Gu and Dunson, 2023).

Subspace Approaches and Future Directions:

- SG-MCMC in parameter space subspaces (linear, sparse) (Izmailov et al., 2020; Li et al., 2024).
- Formulating uncertainty statements for targeted subnetworks and beyond (Dold et al., 2024).
- Hybrid samplers combining structured variational inference with MCMC for efficiency (Alexos et al., 2022).
- Exploring subsampling and transfer learning integration (Kirichenko et al., 2023).

Outline

Hybrid Bayesian Approaches: An Overview

Future BDL approaches may focus on uncertainty in specific model areas, while others are efficiently estimated using point estimation. Hybrid approaches combine Bayesian methods with the efficiency of deterministic deep learning.

- Developing methods that apply Bayesian approaches in critical areas for cost-effective uncertainty capture.
- Maintaining deterministic approaches for other model parts (Daxberger et al., 2021b).
- Example: Last-layer Laplace approximation (Daxberger et al., 2021a).

Such hybrid approaches represent a promising research area.

Advancements and Literature in Hybrid Models

Traditional combinations of deep learning and GPs have been limited by GP scalability.

- Recent advances in GP inference scale-up are promising for broader application of hybrid models (Wilson et al., 2016).
- Deep Kernel Learning (DKL) as a key example of scalable hybrid models.

Prolific literature connects BDL with deep Gaussian processes (DGPs), involving neural network GPs arising as infinite-width limits of NNs.

- Connections between NNs and GPs provide valuable insights into BDL theory (Damianou and Lawrence, 2013; Agrawal et al., 2020; Neal, 1996b; de G. Matthews et al., 2018).

Outline

Outline

Outline

Outline

Outline

Softwares

Software packages, libraries or probabilistic programming languages (PPLs) on top of deep learning frameworks include:

- `bayesianize` (Ritter et al., 2021), `bnn_priors` (Fortuin et al., 2021), `Laplace` (Daxberger et al., 2021a), `Pyro` (Bingham et al., 2019) and `TyXe` (Ritter and Karaletsos, 2022) are software species built on `PyTorch`,
- `TensorFlow Probability` is a library built on `TensorFlow`,
- `Fortuna` (Detommaso et al., 2023) is a library built on JAX.

PPLs, such as `Pyro`, play a role in simplifying the application of probabilistic reasoning to deep learning. In fact, abstractions of the probabilistic treatment of NNs in a PPL, such as those performed in the BDL library `TyXe`, can simplify the application of priors and inference techniques to arbitrary NNs, as demonstrated in a variety of models implemented in `TyXe`. Porting such ideas to modern problem settings involving LLMs and more bespoke probabilistic structures would enable the use of BDL in real-world problems.

References I

Abdar, M., Pourpanah, F., Hussain, S., Rezazadegan, D., Liu, L., Ghavamzadeh, M., Fieguth, P., Cao, X., Khosravi, A., and Acharya, U. R. (2021a). A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*.

Abdar, M., Samami, M., Mahmoodabad, S. D., Doan, T., Mazoure, B., Hashemifesharaki, R., Liu, L., Khosravi, A., Acharya, U. R., Makarenkov, V., et al. (2021b). Uncertainty quantification in skin cancer classification using three-way decision-based Bayesian deep learning. *Computers in Biology and Medicine*, 135:104418.

Agrawal, D., Papamarkou, T., and Hinkle, J. (2020). Wide neural networks with bottlenecks are deep Gaussian processes. *Journal of Machine Learning Research*, 21(175):1–66.

Alexos, A., Boyd, A. J., and Mandt, S. (2022). Structured stochastic gradient MCMC. In *International Conference on Machine Learning*.

Andriushchenko, M. (2023). Adversarial attacks on GPT-4 via simple random search. *Preprint*.

Antoran, J., Padhy, S., Barbano, R., Nalisnick, E., Janz, D., and Hernández-Lobato, J. M. (2023). Sampling-based inference for large linear models, with application to linearised Laplace. In *International Conference on Learning Representations*.

Arbel, J., Pitas, K., and Vladimirova, M. (2022). A primer on Bayesian neural networks: review and debates. *Preprint*.

References II

Ashukha, A., Lyzhov, A., Molchanov, D., and Vetrov, D. (2020). Pitfalls of in-domain uncertainty estimation and ensembling in deep learning. *International Conference on Learning Representations*.

Atzeni, M., Sachan, M., and Loukas, A. (2023). Infusing lattice symmetry priors in attention mechanisms for sample-efficient abstract geometric reasoning. *arXiv preprint arXiv:2306.03175*.

Bamler, R., Salehi, F., and Mandt, S. (2020). Augmenting and tuning knowledge graph embeddings. In *Conference on Uncertainty in Artificial Intelligence*.

Band, N., Rudner, T. G. J., Feng, Q., Filos, A., Nado, Z., Dusenberry, M. W., Jerfel, G., Tran, D., and Gal, Y. (2021). Benchmarking Bayesian Deep Learning on Diabetic Retinopathy Detection Tasks. In *Advances in Neural Information Processing Systems*.

Barron, A. R. (1994). Approximation and estimation bounds for artificial neural networks. *Machine Learning*, 14(1):115–133.

Belkin, M., Hsu, D., Ma, S., and Mandal, S. (2019). Reconciling modern machine-learning practice and the classical bias–variance trade-off. *National Academy of Sciences*, 116(32):15849–15854.

Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., and Goodman, N. D. (2019). Pyro: Deep universal probabilistic programming. *Journal of Machine Learning Research*, 20(28):1–6.

References III

Bouchiat, K., Immer, A., Yèche, H., Rätsch, G., and Fortuin, V. (2023). Laplace-approximated neural additive models: Improving interpretability with Bayesian inference. *arXiv preprint arXiv:2305.16905*.

Brooks, S., Gelman, A., Jones, G., and Meng, X.-L. (2011). *Handbook of Markov chain Monte Carlo*. CRC Press.

Casper, S., Davies, X., Shi, C., Gilbert, T. K., Scheurer, J., Rando, J., Freedman, R., Korbak, T., Lindner, D., Freire, P., et al. (2023). Open problems and fundamental limitations of reinforcement learning from human feedback. *Transactions on Machine Learning Research*.

Chatziafratis, V., Nagarajan, S. G., and Panageas, I. (2020a). Better depth-width trade-offs for neural networks through the lens of dynamical systems. In *International Conference on Machine Learning*.

Chatziafratis, V., Nagarajan, S. G., Panageas, I., and Wang, X. (2020b). Depth-width trade-offs for ReLU networks via Sharkovsky's theorem. *International Conference on Learning Representations*.

Chen, T., Fox, E., and Guestrin, C. (2014). Stochastic gradient Hamiltonian Monte Carlo. In *International Conference on Machine Learning*.

Ciosek, K., Fortuin, V., Tomioka, R., Hofmann, K., and Turner, R. (2020). Conservative uncertainty estimation by fitting prior networks. In *International Conference on Learning Representations*.

Cohen, S. (2022). *Bayesian analysis in natural language processing*. Springer Nature.

References IV

Cranmer, M., Tamayo, D., Rein, H., Battaglia, P., Hadden, S., Armitage, P. J., Ho, S., and Spergel, D. N. (2021). A Bayesian neural network predicts the dissolution of compact planetary systems. *Proceedings of the National Academy of Sciences*, 118(40):e2026053118.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314.

Damianou, A. and Lawrence, N. D. (2013). Deep Gaussian processes. In *International Conference on Artificial Intelligence and Statistics*.

D'Angelo, F. and Fortuin, V. (2021). Repulsive deep ensembles are Bayesian. *Advances in Neural Information Processing Systems*.

D'Angelo, F., Fortuin, V., and Wenzel, F. (2021). On Stein variational neural network ensembles. *arXiv preprint arXiv:2106.10760*.

Daxberger, E., Kristiadi, A., Immer, A., Eschenhagen, R., Bauer, M., and Hennig, P. (2021a). Laplace redux - effortless Bayesian deep learning. In *Advances in Neural Information Processing Systems*.

Daxberger, E., Nalisnick, E., Allingham, J. U., Antoran, J., and Hernández-Lobato, J. M. (2021b). Bayesian deep learning via subnetwork inference. In *International Conference on Machine Learning*.

de G. Matthews, A. G., Rowland, M., Hron, J., Turner, R. E., and Ghahramani, Z. (2018). Gaussian process behaviour in wide deep neural networks. *International Conference on Learning Representations*.

References V

Detommaso, G., Gasparin, A., Donini, M., Seeger, M., Wilson, A. G., and Archambeau, C. (2023). Fortuna: A library for uncertainty quantification in deep learning. *arXiv preprint arXiv:2302.04019*.

Dold, D., Rügamer, D., Sick, B., and Dürr, O. (2024). Semi-structured subspace inference. In *International Conference on Artificial Intelligence and Statistics*.

Ferreira, L., Conselice, C. J., Duncan, K., Cheng, T.-Y., Griffiths, A., and Whitney, A. (2020). Galaxy merger rates up to z $\sim$ 3 using a Bayesian deep learning model: A major-merger classifier using illustrisTNG simulation data. *The Astrophysical Journal*, 895(2):115.

Finzi, M., Benton, G., and Wilson, A. G. (2021). Residual pathway priors for soft equivariance constraints. In *Advances in Neural Information Processing Systems*.

Folgoc, L. L., Baltatzis, V., Desai, S., Devaraj, A., Ellis, S., Manzanera, O. E. M., Nair, A., Qiu, H., Schnabel, J., and Glocker, B. (2021). Is mc dropout bayesian? *arXiv preprint arXiv:2110.04286*.

Fortuin, V. (2022). Priors in Bayesian deep learning: A review. *International Statistical Review*.

Fortuin, V., Garriga-Alonso, A., Ober, S. W., Wenzel, F., Rätsch, G., Turner, R. E., van der Wilk, M., and Aitchison, L. (2022). Bayesian neural network priors revisited. In *International Conference on Learning Representations*.

Fortuin, V., Garriga-Alonso, A., van der Wilk, M., and Aitchison, L. (2021). BNNpriors: A library for Bayesian neural network inference with different prior distributions. *Software Impacts*, 9:100079.

References VI

Funahashi, K.-I. (1989). On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2(3):183–192.

Gal, Y. and Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*.

Gal, Y., Islam, R., and Ghahramani, Z. (2017). Deep Bayesian active learning with image data. In *International Conference on Machine Learning*.

Gelman, A., Vehtari, A., Simpson, D., Margossian, C. C., Carpenter, B., Yao, Y., Kennedy, L., Gabry, J., Bürkner, P.-C., and Modrák, M. (2020). Bayesian workflow. *arXiv preprint arXiv:2011.01808*.

Ghosh, S., Yao, J., and Doshi-Velez, F. (2018). Structured variational learning of bayesian neural networks with horseshoe priors. In *International Conference on Machine Learning*.

Goan, E. and Fookes, C. (2020). Bayesian neural networks: An introduction and survey. In *Case Studies in Applied Bayesian Data Science*, pages 45–87. Springer.

Gruver, N., Stanton, S., Kirichenko, P., Finzi, M., Maffettone, P., Myers, V., Delaney, E., Greenside, P., and Wilson, A. G. (2021). Effective surrogate models for protein design with Bayesian optimization. In *ICML Workshop on Computational Biology*.

Gu, Y. and Dunson, D. B. (2023). Bayesian pyramids: identifiable multilayer discrete latent structure models for discrete data. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 85(2):399–426.

He, B., Lakshminarayanan, B., and Teh, Y. W. (2020). Bayesian deep ensembles via the neural tangent kernel. In *Advances in Neural Information Processing Systems*.

References VII

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*.

Hein, M., Andriushchenko, M., and Bitterwolf, J. (2019). Why ReLU networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Computer Vision and Pattern Recognition*.

Hernández, S. and López, J. L. (2020). Uncertainty quantification for plant disease detection using Bayesian deep learning. *Applied Soft Computing*, 96:106597.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Hoeting, J. A., Madigan, D., Raftery, A. E., and Volinsky, C. T. (1998). Bayesian model averaging. In *AAAI Workshop on Integrating Multiple Learned Models*.

Hoeting, J. A., Madigan, D., Raftery, A. E., and Volinsky, C. T. (1999). Bayesian model averaging: a tutorial (with comments by m. clyde, david draper and ei george. *Statistical Science*, 14(4):382–417.

Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366.

Hubin, A. and Storvik, G. (2019). Combining model and parameter uncertainty in Bayesian neural networks. *arXiv preprint arXiv:1903.07594*.

Hubin, A., Storvik, G., and Frommlet, F. (2021). Flexible Bayesian nonlinear model configuration. *Journal of Artificial Intelligence Research*, 72:901–942.

Immer, A., Bauer, M., Fortuin, V., Rätsch, G., and Khan, M. E. (2021a). Scalable marginal likelihood estimation for model selection in deep learning. In *International Conference on Machine Learning*.

References VIII

Immer, A., Korzepa, M., and Bauer, M. (2021b). Improving predictions of Bayesian neural nets via local linearization. In *International Conference on Artificial Intelligence and Statistics*.

Immer, A., van der Ouderaa, T., Rätsch, G., Fortuin, V., and van der Wilk, M. (2022). Invariance learning in deep neural networks with differentiable Laplace approximations. *Advances in Neural Information Processing Systems*.

Immer, A., Van Der Ouderaa, T. F., Van Der Wilk, M., Ratsch, G., and Schölkopf, B. (2023). Stochastic marginal likelihood gradients using neural tangent kernels. In *International Conference on Machine Learning*.

Izmailov, P., Maddox, W. J., Kirichenko, P., Garipov, T., Vetrov, D., and Wilson, A. G. (2020). Subspace inference for Bayesian deep learning. In *Conference on Uncertainty in Artificial Intelligence*.

Izmailov, P., Vikram, S., Hoffman, M. D., and Wilson, A. G. G. (2021). What are Bayesian neural network posteriors really like? In *International Conference on Machine Learning*.

Jacot, A., Gabriel, F., and Hongler, C. (2018). Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12).

References IX

Jospin, L. V., Buntine, W., Boussaid, F., Laga, H., and Bennamoun, M. (2020a). Hands-on bayesian neural networks–a tutorial for deep learning users. *arXiv preprint arXiv:2007.06823*.

Jospin, L. V., Buntine, W., Boussaid, F., Laga, H., and Bennamoun, M. (2020b). Hands-on bayesian neural networks–a tutorial for deep learning users. *arXiv preprint arXiv:2007.06823*.

Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Hatfield-Dodds, Z., DasSarma, N., Tran-Johnson, E., Johnston, S., El-Showk, S., Jones, A., Elhage, N., Hume, T., Chen, A., Bai, Y., Bowman, S., Fort, S., Ganguli, D., Hernandez, D., Jacobson, J., Kernion, J., Kravec, S., Lovitt, L., Ndousse, K., Olsson, C., Ringer, S., Amodei, D., Brown, T., Clark, J., Joseph, N., Mann, B., McCandlish, S., Olah, C., and Kaplan, J. (2022). Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Khan, M. E., Immer, A., Abedi, E., and Korzepa, M. (2019). Approximate inference turns deep networks into Gaussian processes. In *Advances in Neural Information Processing Systems*.

Kingma, D. P., Salimans, T., and Welling, M. (2015). Variational dropout and the local reparameterization trick. In *Advances in Neural Information Processing Systems*.

Kirichenko, P., Izmailov, P., and Wilson, A. G. (2023). Last layer re-training is sufficient for robustness to spurious correlations. In *International Conference on Learning Representations*.

References X

Klarner, L., Rudner, T. G. J., Reutlinger, M., Schindler, T., Morris, G. M., Deane, C., and Teh, Y. W. (2023). Drug Discovery under Covariate Shift with Domain-Informed Prior Distributions over Functions. In *International Conference on Machine Learning*.

Kolb, C., Müller, C. L., Bischl, B., and Rügamer, D. (2023). Smoothing the edges: A general framework for smooth optimization in sparse regularization using Hadamard overparametrization. *arXiv preprint arXiv:2307.03571*.

Kristiadi, A., Hein, M., and Hennig, P. (2020). Being Bayesian, even just a bit, fixes overconfidence in ReLU networks. In *International Conference on Machine Learning*.

Kristiadi, A., Hein, M., and Hennig, P. (2021a). An infinite-feature extension for Bayesian ReLU nets that fixes their asymptotic overconfidence. In *Advances in Neural Information Processing Systems*.

Kristiadi, A., Hein, M., and Hennig, P. (2021b). Learnable uncertainty under Laplace approximations. In *Conference on Uncertainty in Artificial Intelligence*.

Lawrence, N. D. (2001). *Variational inference in probabilistic models*. PhD thesis, University of Cambridge.

Lee, K., Lee, H., Lee, K., and Shin, J. (2018). Training confidence-calibrated classifiers for detecting out-of-distribution samples. *International Conference on Learning Representations*.

Leitherer, A., Ziletti, A., and Ghiringhelli, L. M. (2021). Robust recognition and exploratory analysis of crystal structures via Bayesian deep learning. *Nature Communications*, 12(1):6234.

References XI

Li, J., Miao, Z., Qiu, Q., and Zhang, R. (2024). Training Bayesian neural networks with sparse subspace variational inference. In *International Conference on Learning Representations*.

Li, Y. L., Rudner, T. G., and Wilson, A. G. (2023). A study of Bayesian neural network surrogates for Bayesian optimization. *arXiv preprint arXiv:2305.20028*.

Lipman, Y., Chen, R. T., Ben-Hamu, H., Nickel, M., and Le, M. (2022). Flow matching for generative modeling. In *International Conference on Learning Representations*.

Liu, Q. and Wang, D. (2016). Stein variational gradient descent: A general purpose Bayesian inference algorithm. *Advances in Neural Information Processing Systems*.

Luo, X., Nadiga, B. T., Park, J. H., Ren, Y., Xu, W., and Yoo, S. (2022). A Bayesian deep learning approach to near-term climate prediction. *Journal of Advances in Modeling Earth Systems*, 14(10):e2022MS003058.

MacKay, D. J. (1992). Bayesian interpolation. *Neural Computation*, 4(3):415–447.

MacKay, D. J. (1998). Choice of basis for Laplace approximation. *Machine Learning*, 33:77–86.

Maddox, W. J., Izmailov, P., Garipov, T., Vetrov, D. P., and Wilson, A. G. (2019). A simple baseline for Bayesian uncertainty in deep learning. *Advances in Neural Information Processing Systems*, 32.

Mandt, S., Hoffman, M. D., and Blei, D. M. (2017). Stochastic gradient descent as approximate Bayesian inference. *Journal of Machine Learning Research*, 18(134):1–35.

References XII

Manogaran, G., Shakeel, P. M., Fouad, H., Nam, Y., Baskar, S., Chilamkurti, N., and Sundarasekar, R. (2019). Wearable IoT smart-log patch: An edge computing-based Bayesian deep learning network system for multi access physical monitoring system. *Sensors*, 19(13):3030.

Margatina, K., Barrault, L., and Aletras, N. (2022). On the importance of effectively adapting pretrained language models for active learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.

Margatina, K., Schick, T., Aletras, N., and Dwivedi-Yu, J. (2023). Active learning principles for in-context learning with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*.

McAllister, R., Gal, Y., Kendall, A., Van Der Wilk, M., Shah, A., Cipolla, R., and Weller, A. (2017). Concrete problems for autonomous vehicle safety: Advantages of Bayesian deep learning. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*.

Minderer, M., Djolonga, J., Romijnders, R., Hubis, F., Zhai, X., Houlsby, N., Tran, D., and Lucic, M. (2021). Revisiting the calibration of modern neural networks. *Advances in Neural Information Processing Systems*.

Mitros, J. and Mac Namee, B. (2019). On the validity of Bayesian neural networks for uncertainty estimation. *arXiv preprint arXiv:1912.01530*.

Mohamed, S., Rosca, M., Figurnov, M., and Mnih, A. (2020). Monte carlo gradient estimation in machine learning. *J. Mach. Learn. Res.*, 21(132):1–62.

References XIII

Moor, M., Banerjee, O., Abad, Z. S. H., Krumholz, H. M., Leskovec, J., Topol, E. J., and Rajpurkar, P. (2023). Foundation models for generalist medical artificial intelligence. *Nature*, 616(7956):259–265.

Mur-Labadia, L., Martinez-Cantin, R., and Guerrero, J. J. (2023). Bayesian deep learning for affordance segmentation in images. *arXiv preprint arXiv:2303.00871*.

Murphy, K. P. (2023). *Probabilistic Machine Learning: Advanced Topics*. MIT Press.

Nado, Z., Band, N., Collier, M., Djolonga, J., Dusenberry, M. W., Farquhar, S., Feng, Q., Filos, A., Havasi, M., and Jenatton, R. (2021). Uncertainty baselines: Benchmarks for uncertainty & robustness in deep learning. *arXiv preprint arXiv:2106.04015*.

Neal, R. M. (1996a). *Bayesian learning for neural networks*. Springer Science & Business Media.

Neal, R. M. (1996b). Priors for infinite networks. *Bayesian learning for neural networks*, pages 29–53.

Nemeth, C. and Fearnhead, P. (2021). Stochastic gradient Markov chain Monte Carlo. *Journal of the American Statistical Association*, 116(533):433–450.

Nguyen, C. V., Li, Y., Bui, T. D., and Turner, R. E. (2018). Variational continual learning. In *International Conference on Learning Representations*.

Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J. V., Lakshminarayanan, B., and Snoek, J. (2019). Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. *Advances in Neural Information Processing Systems*.

References XIV

Peng, W., Ye, Z.-S., and Chen, N. (2019). Bayesian deep-learning-based health prognostics toward prognostics uncertainty. *IEEE Transactions on Industrial Electronics*, 67(3):2283–2293.

Pielok, T., Bischl, B., and Rügamer, D. (2022). Approximate Bayesian inference with Stein functional variational gradient descent. In *International Conference on Learning Representations*.

Polson, N. G. and Ročková, V. (2018). Posterior concentration for sparse deep learning. *Advances in Neural Information Processing Systems*.

Poole, B., Lahiri, S., Raghu, M., Sohl-Dickstein, J., and Ganguli, S. (2016). Exponential expressivity in deep neural networks through transient chaos. In *International Conference on Neural Information Processing Systems*.

Rainforth, T., Foster, A., Ivanova, D. R., and Smith, F. B. (2023). Modern Bayesian experimental design. *arXiv preprint arXiv:2302.14545*.

Ritter, H., Botev, A., and Barber, D. (2018). A scalable Laplace approximation for neural networks. In *International Conference on Learning Representations*.

Ritter, H. and Karaletsos, T. (2022). TyXe: Pyro-based Bayesian neural nets for Pytorch. In *Proceedings of Machine Learning and Systems*.

Ritter, H., Kukla, M., Zhang, C., and Li, Y. (2021). Sparse uncertainty representation in deep learning with inducing weights. In *Advances in Neural Information Processing Systems*.

Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, pages 400–407.

References XV

Robbins, H. E. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407.

Rothfuss, J., Fortuin, V., Josifoski, M., and Krause, A. (2021). PACOH: Bayes-optimal meta-learning with PAC-guarantees. In *International Conference on Machine Learning*.

Rothfuss, J., Josifoski, M., Fortuin, V., and Krause, A. (2022). PAC-Bayesian meta-learning: From theory to practice. *arXiv preprint arXiv:2211.07206*.

Rudner, T. G. J., Chen, Z., Teh, Y. W., and Gal, Y. (2022a). Tractable Function-Space Variational Inference in Bayesian Neural Networks. In *Advances in Neural Information Processing Systems*.

Rudner, T. G. J., Kapoor, S., Qiu, S., and Wilson, A. G. (2023a). Function-Space Regularization in Neural Networks: A Probabilistic Perspective. In *International Conference on Machine Learning*.

Rudner, T. G. J., Pan, X., Li, Y. L., Shwartz-Ziv, R., and Wilson, A. G. (2023b). Uncertainty-aware priors for finetuning pretrained models. In *Preprint*.

Rudner, T. G. J., Smith, F. B., Feng, Q., Teh, Y. W., and Gal, Y. (2022b). Continual Learning via Sequential Function-Space Variational Inference. In *International Conference on Machine Learning*.

Rudner, T. G. J., Zhang, Y. S., Wilson, A. G., and Kempe, J. (2024). Mind the GAP: Improving robustness to subpopulation shifts with group-aware priors. In *International Conference on Artificial Intelligence and Statistics*.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536.

Schoenholz, S. S., Gilmer, J., Ganguli, S., and Sohl-Dickstein, J. (2017). Deep information propagation. In *International Conference on Learning Representations*.

Schwöbel, P., Jørgensen, M., Ober, S. W., and Van Der Wilk, M. (2022). Last layer marginal likelihood for invariance learning. In *International Conference on Artificial Intelligence and Statistics*.

Sen, D., Papamarkou, T., and Dunson, D. (2024). Bayesian neural networks and dimensionality reduction. In *Handbook of Bayesian, fiducial, and frequentist inference*. Chapmann and Hall/CRC Press.

Sharma, M., Rainforth, T., Teh, Y. W., and Fortuin, V. (2023). Incorporating unlabelled data into Bayesian neural networks. *arXiv preprint arXiv:2304.01762*.

Shi, L., Copot, C., and Vanlanduit, S. (2021). A Bayesian deep neural network for safe visual servoing in human–robot interaction. *Frontiers in Robotics and AI*, 8:687031.

Shwartz-Ziv, R., Goldblum, M., Souri, H., Kapoor, S., Zhu, C., LeCun, Y., and Wilson, A. G. (2022). Pre-train your loss: Easy Bayesian transfer learning with informative priors. In *Advances in Neural Information Processing Systems*.

Skaaret-Lund, L., Storvik, G., and Hubin, A. (2023). Sparsifying Bayesian neural networks with latent binary variables and normalizing flows. *arXiv preprint arXiv:2305.03395*.

Soboczenski, F., Himes, M. D., O'Beirne, M. D., Zorzan, S., Baydin, A. G., Cobb, A. D., Gal, Y., Angerhausen, D., Mascaro, M., Arney, G. N., et al. (2018). Bayesian deep learning for exoplanet atmospheric retrieval. *arXiv preprint arXiv:1811.03390*.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2020). Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*.

Sun, S., Cao, Z., Zhu, H., and Zhao, J. (2019). A survey of optimization methods from a machine learning perspective. *IEEE Transactions on Cybernetics*, 50(8):3668–3681.

Tran, D., Liu, J., Dusenberry, M. W., Phan, D., Collier, M., Ren, J., Han, K., Wang, Z., Mariet, Z., Hu, H., Band, N., Rudner, T. G. J., Singhal, K., Nado, Z., van Amersfoort, J., Kirsch, A., Jenatton, R., Thain, N., Yuan, H., Buchanan, K., Murphy, K., Sculley, D., Gal, Y., Ghahramani, Z., Snoek, J., and Lakshminarayanan, B. (2022). Plex: Towards Reliability Using Pretrained Large Model Extensions. In *ICML 2022 Workshop on Pre-training: Perspectives, Pitfalls, and Paths Forward*.

van der Ouderaa, T. F., Immer, A., and van der Wilk, M. (2023). Learning layer-wise equivariances automatically using gradients. In *Advances in Neural Information Processing Systems*.

Vandal, T., Kodra, E., Dy, J., Ganguly, S., Nemani, R., and Ganguly, A. R. (2018). Quantifying uncertainty in discrete-continuous and skewed data with Bayesian deep learning. In *International Conference on Knowledge Discovery & Data Mining*.

Villani, C. (2021). *Topics in optimal transportation*, volume 58. American Mathematical Soc.

Vladimirova, M., Arbel, J., and Girard, S. (2021). Bayesian neural network unit priors and generalized Weibull-tail property. *Asian Conference on Machine Learning*.

Vladimirova, M., Verbeek, J., Mesejo, P., and Arbel, J. (2019). Understanding priors in Bayesian neural networks at the unit level. In *International Conference on Machine Learning*.

Wang, Y., Rudner, T. G. J., and Wilson, A. G. (2023a). Visual explanations of image-text representations via multi-modal information bottleneck attribution. In *Advances in Neural Information Processing Systems*.

Wang, Z., Chen, Y., Song, Q., and Zhang, R. (2023b). Enhancing low-precision sampling via stochastic gradient Hamiltonian Monte Carlo. *arXiv preprint arXiv:2310.16320*.

Wasserman, L. (2000). Bayesian model selection and model averaging. *Journal of Mathematical Psychology*, 44(1):92–107.

Way, G. P. and Greene, C. S. (2018). Bayesian deep learning for single-cell analysis. *Nature Methods*, 15(12):1009–1010.

Welling, M. and Teh, Y. W. (2011). Bayesian learning via stochastic gradient Langevin dynamics. In *International Conference on Machine Learning*.

Wenzel, F., Roth, K., Veeling, B., Swiatkowski, J., Tran, L., Mandt, S., Snoek, J., Salimans, T., Jenatton, R., and Nowozin, S. (2020). How good is the Bayes posterior in deep neural networks really? In *International Conference on Machine Learning*.

Wiese, J. G., Wimmer, L., Papamarkou, T., Bischl, B., Günnemann, S., and Rügamer, D. (2023). Towards efficient MCMC sampling in Bayesian neural networks by exploiting symmetry. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*.

Wild, V. D., Ghalebikesabi, S., Sejdinovic, D., and Knoblauch, J. (2023). A rigorous link between deep ensembles and (variational) Bayesian methods. In *Conference on Neural Information Processing Systems*.

Wilson, A. G., Hu, Z., Salakhutdinov, R., and Xing, E. P. (2016). Deep kernel learning. In *International Conference on Artificial Intelligence and Statistics*.

Wilson, A. G. and Izmailov, P. (2020). Bayesian deep learning and a probabilistic perspective of generalization. *Advances in Neural Information Processing Systems*.

Xie, S. M., Raghunathan, A., Liang, P., and Ma, T. (2021). An explanation of in-context learning as implicit Bayesian inference. *arXiv preprint arXiv:2111.02080*.

Yang, A. X., Robeyns, M., Wang, X., and Aitchison, L. (2024). Bayesian low-rank adaptation for large language models. *International Conference on Learning Representations*.

Yang, Y., Hui, B., Yuan, H., Gong, N., and Cao, Y. (2023). SneakyPrompt: Evaluating robustness of text-to-image generative models' safety filters. In *IEEE Symposium on Security and Privacy*.

Yang, Y., Li, W., Gulliver, T. A., and Li, S. (2019). Bayesian deep learning-based probabilistic load forecasting in smart grids. *IEEE Transactions on Industrial Informatics*, 16(7):4703–4713.

Zhang, J., Jennings, J., Zhang, C., and Ma, C. (2023). Towards causal foundation model: on duality between causal inference and attention. *arXiv preprint arXiv:2310.00809*.

References XX

Zhang, R., Li, C., Zhang, J., Chen, C., and Wilson, A. G. (2019). Cyclical stochastic gradient MCMC for Bayesian deep learning. In *International Conference on Learning Representations*.

Zhang, R., Wilson, A. G., and De Sa, C. (2022). Low-precision stochastic gradient Langevin dynamics. In *International Conference on Machine Learning*.

Zhou, Z., Yu, H., and Shi, H. (2020). Human activity recognition based on improved Bayesian convolution network to analyze health care data using wearable iot device. *IEEE Access*, 8:86411–86418.