Uncertainty in Deep Neural Networks

Franz Pernkopf

intelligent systems
TU Graz

Signal Processing and Speech Communication Laboratory
Graz University of Technology
Austria

pernkopf@tugraz.at

Motivation

- Deep neural networks (DNNs) achieve state-of-the-art results in various domains
- Despite their predictive performance
  ⇒ limited usability in safety-critical applications

[Gaw22] J. Gawlikowski et al. A Survey of Uncertainty in Deep Neural Networks. 2022.

Motivation

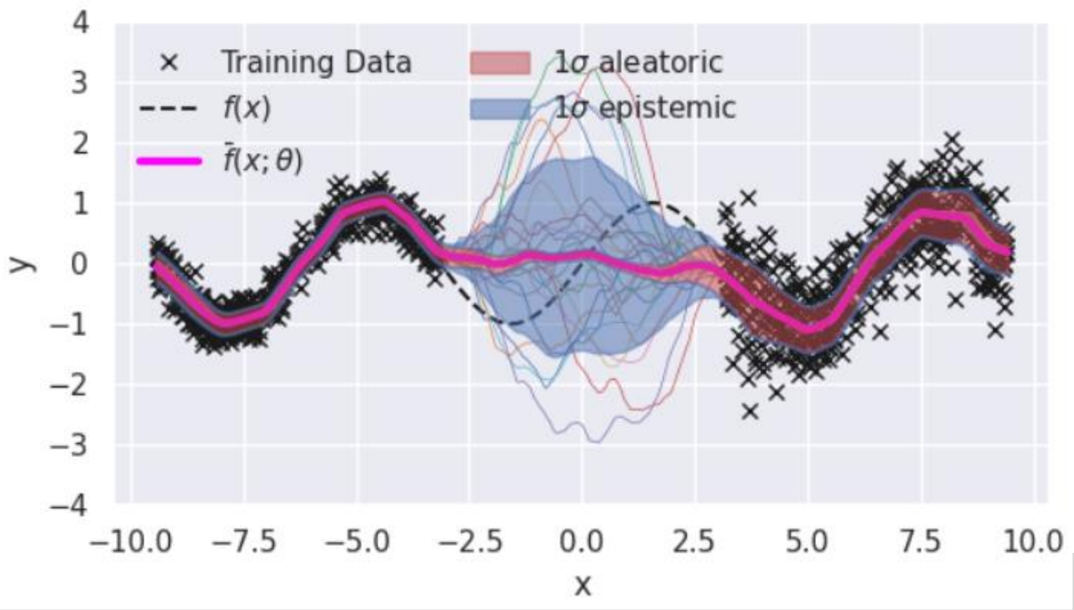
- Deep neural networks (DNNs) achieve state-of-the-art results in various domains
- Despite their predictive performance
  → limited usability in safety-critical applications
- Main factors:
  - ✓ Lack of transparency of DNN's inference
  - ✓ Inability to distinguish between in-domain and out-of-domain (OOD) samples
  - ✓ Sensitivity to domain shifts
  - ✓ Inability to provide reliable uncertainty estimates
  - ✓ Sensitivity to adversarial attacks
- Overcome these limitations:
  Essential to provide reliable uncertainty estimates

[Gaw22] J. Gawlikowski et al. A Survey of Uncertainty in Deep Neural Networks. 2022.

Uncertainty Modeling

Predictive uncertainty of a DNN is composed by:

- **Aleatoric uncertainty**: Captures noise inherent in the data (not reduceable)
- **Epistemic uncertainty**: Uncertainty in the model due to lack of knowledge and data; can be reduced by more data

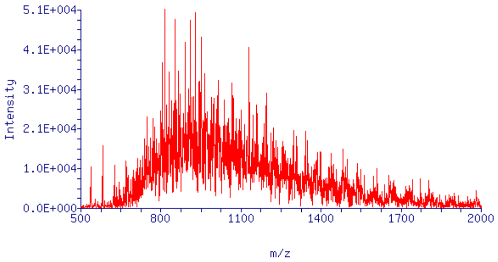


Sources for Uncertainty and Error

- Variabiltiy in the real world
  - ✓ Distribution shift

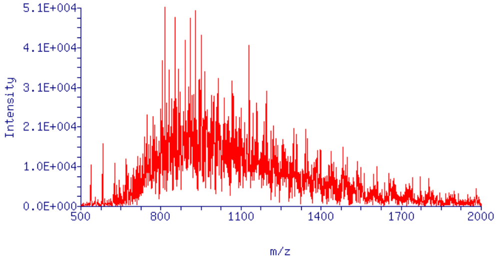
Sources for Uncertainty and Error

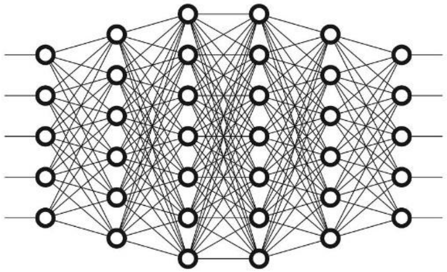
- Variabiltiy in the real world
  - ✓ Distribution shift
- Error and noise in measurement
  - ✓ Sensor noise
  - ✓ Label noise



Sources for Uncertainty and Error

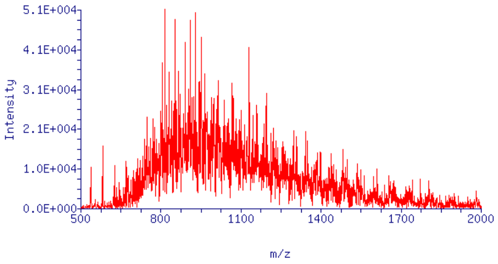
- Variabiltiy in the real world
  - ✓ Distribution shift
- Error and noise in measurement
  - ✓ Sensor noise
  - ✓ Label noise
- Error in DNN model structure
  - ✓ Architecture & size
  - ✓ Deep vs. shallow



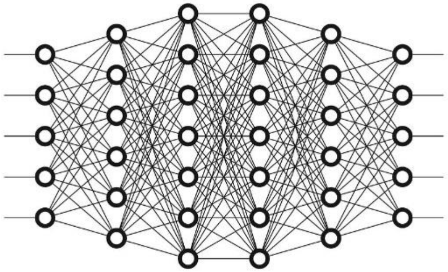


Sources for Uncertainty and Error

- Variabiltiy in the real world
  - ✓ Distribution shift



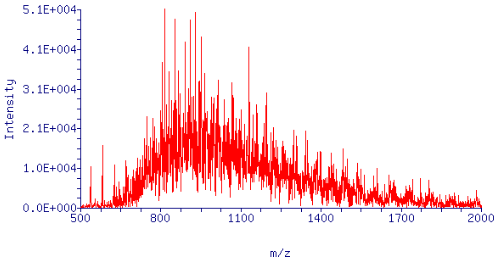
- Error and noise in measurement
  - ✓ Sensor noise
  - ✓ Label noise
- Error in DNN model structure
  - ✓ Architecture & size
  - ✓ Deep vs. shallow



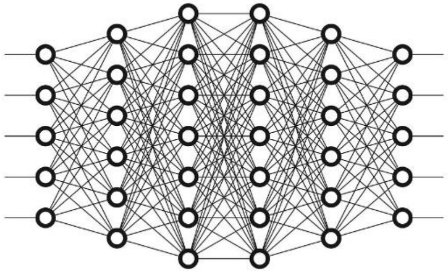
- Error in training
  - ✓ Many parameters to tune: batch size, optimizer, learning rate, regularizer etc.
  - ✓ Lack in training data: imbalance, coverage, size

Sources for Uncertainty and Error

- Variabiltiy in the real world
  - ✓ Distribution shift



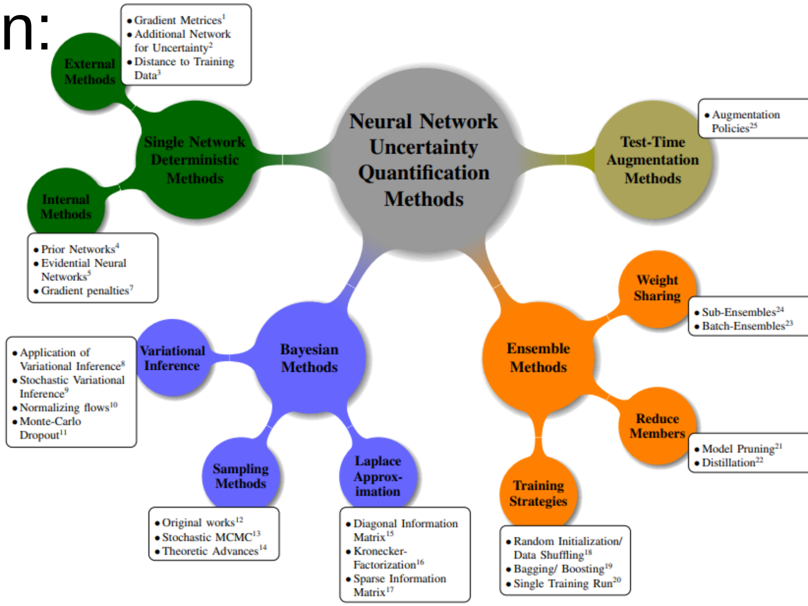
- Error and noise in measurement
  - ✓ Sensor noise
  - ✓ Label noise
- Error in DNN model structure
  - ✓ Architecture & size
  - ✓ Deep vs. shallow



- Error in training
  - ✓ Many parameters to tune: batch size, optimizer, learning rate, regularizer etc.
  - ✓ Lack in training data: imbalance, coverage, size
- Errors caused by unknown data
  - ✓ Out-of-domain (OOD) data

Outline

Methods for uncertainty estimation:



- ✓ Single deterministic models
- ✓ Bayesian neural networks
- ✓ Ensemble methods
- ✓ Particle-optimization based variational inference
- ✓ Single multi-headed model

Some experiments & results

[Gaw22] J. Gawlikowski et al. A Survey of Uncertainty in Deep Neural Networks. 2022.

Single Deterministic Methods

- Class probabilities of a single (deterministic) network (with softmax output layer) can be interpreted as uncertainty
- These uncertainties are over-confident

  uncertainties are poorly calibrated

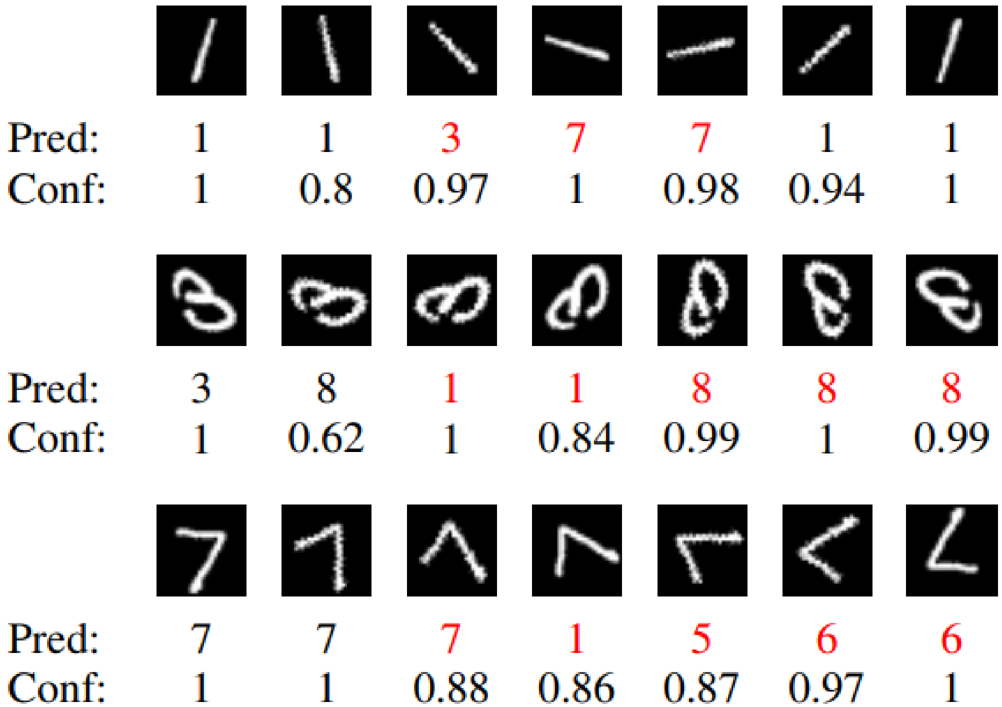
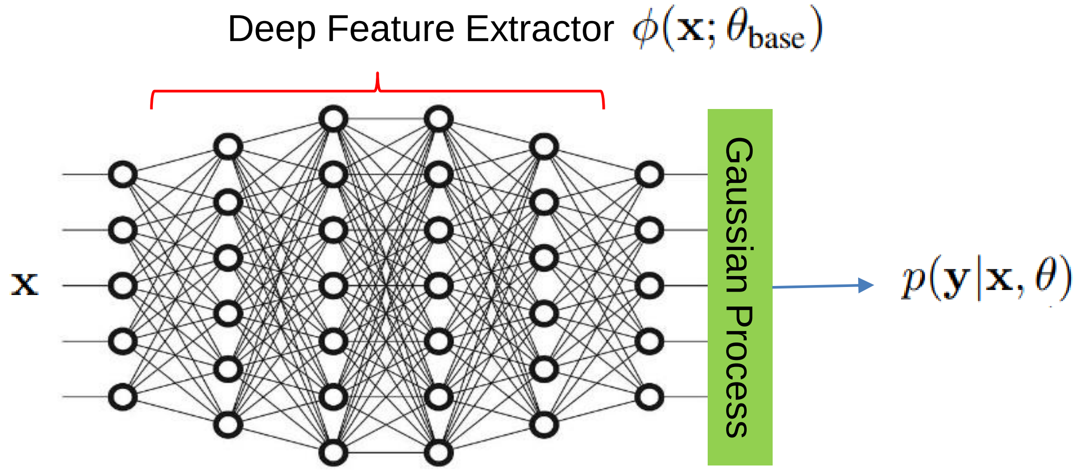


Fig. 5: Predictions received from a LeNet network trained on MNIST's handwritten digits from 0 to 9 and evaluated on different rotations of test samples.

Single Deterministic Methods

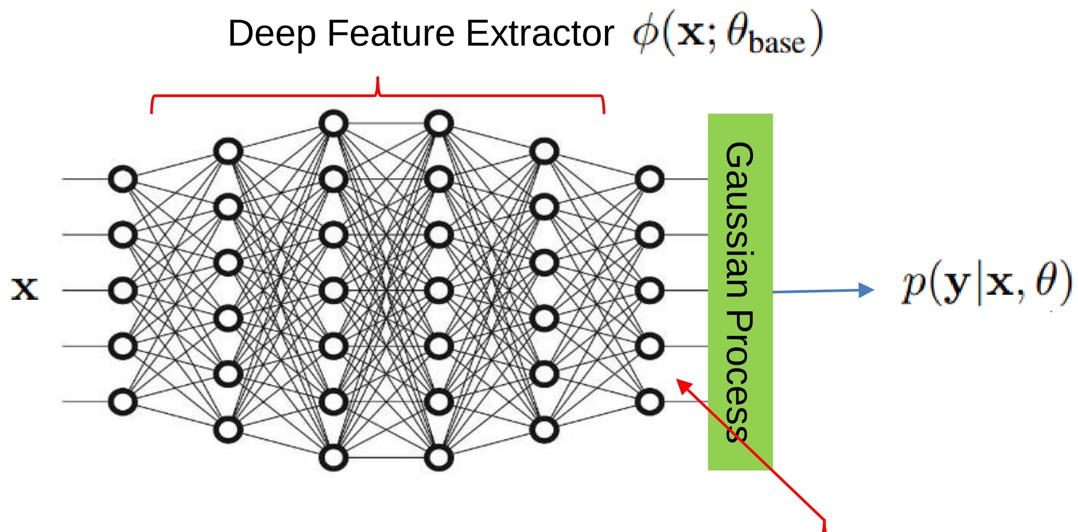
- Spectral-normalized Neural Gaussian Process (SNGP) [Liu20]
    1) Deep feature extractor for input transformation
    2) Gaussian process at output layer (Laplace approximation)



[Liu20] J. Liu, Z. Lin, A. Padhy, D. Tran, T. Bedrax Weiss, Tania and B. Lakshminarayanan, Simple and Principled Uncertainty Estimation with Deterministic Deep Learning via Distance Awareness, NeurIPS 2020.

Single Deterministic Methods

- Spectral-normalized Neural Gaussian Process (SNGP)
  1) Deep feature extractor for input transformation
  2) Gaussian process at output layer (Laplace approximation)



- Important: Ensure distance awareness in feature space

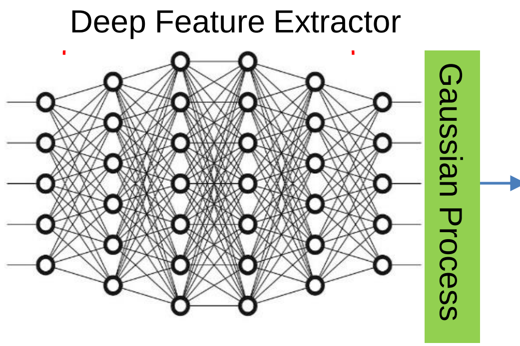
Bi-Lipschitz constraint on deep feature extractor

$$K_L d_I(\mathbf{x}_1,\mathbf{x}_2) \le d_F(\phi(\mathbf{x}_1),\phi(\mathbf{x}_2)) \le K_U d_I(\mathbf{x}_1,\mathbf{x}_2)$$

Distance in input space

Distance in feature space

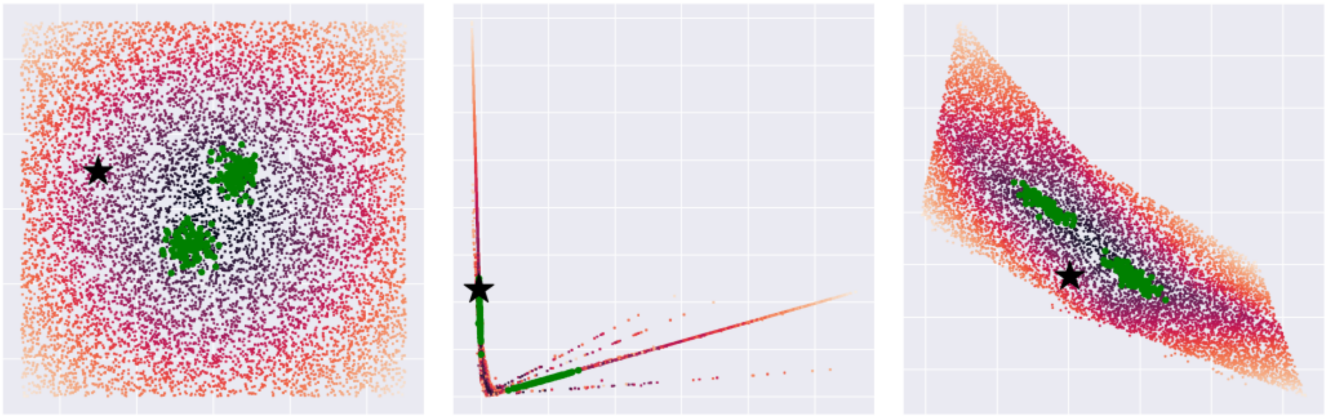
Single Deterministic Methods



- Important: Bi-Lipschitz constraint on deep feature extractor [Liu20, AmS21]

⇒ spectral normalization of weights (i.e. largest singular value ≤1)

⇒ residual connections



(a) Input (b) Trained without constraint (c) Trained with constraint

A 2D classification task where the classes are two Gaussian blobs (drawn in green)

✓ Feature representation is sensitive to changes in input (no feature collapse)

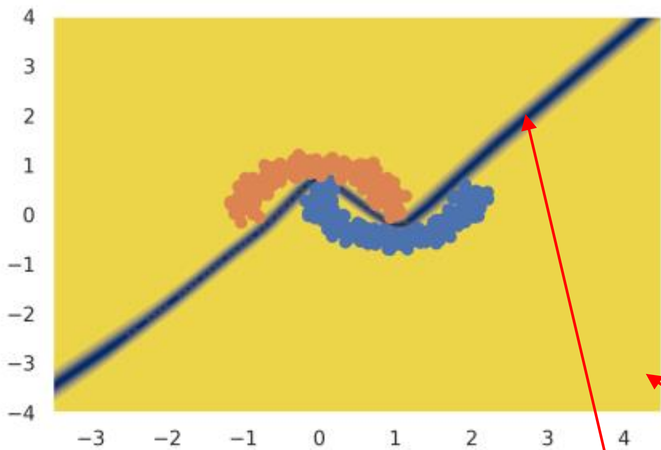
✓ Feature representation is smooth ⇒ generalization and robustness

[Liu20] J. Liu, Z. Lin, A. Padhy, D. Tran, T. Bedrax Weiss, Tania and B. Lakshminarayanan, Simple and Principled Uncertainty Estimation with Deterministic Deep Learning via Distance Awareness, NeurIPS 2020.
[AmS21] van Amersfoort, J., Smith, L., Jesson, A., Key, O., & Gal, Y. “On feature collapse and deep kernel learning for single forward pass uncertainty”. *arXiv preprint arXiv:2102.11409*, 2021

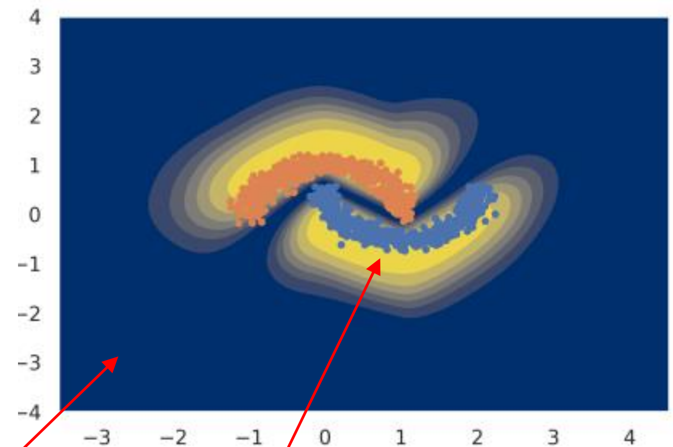
Single Deterministic Methods

- Spectral-normalized Neural Gaussian Process (SNGP)

Uncertainty on two moons data set:



ResNet + Softmax



Spectral-normalized Neural Gaussain Process

High uncertainty

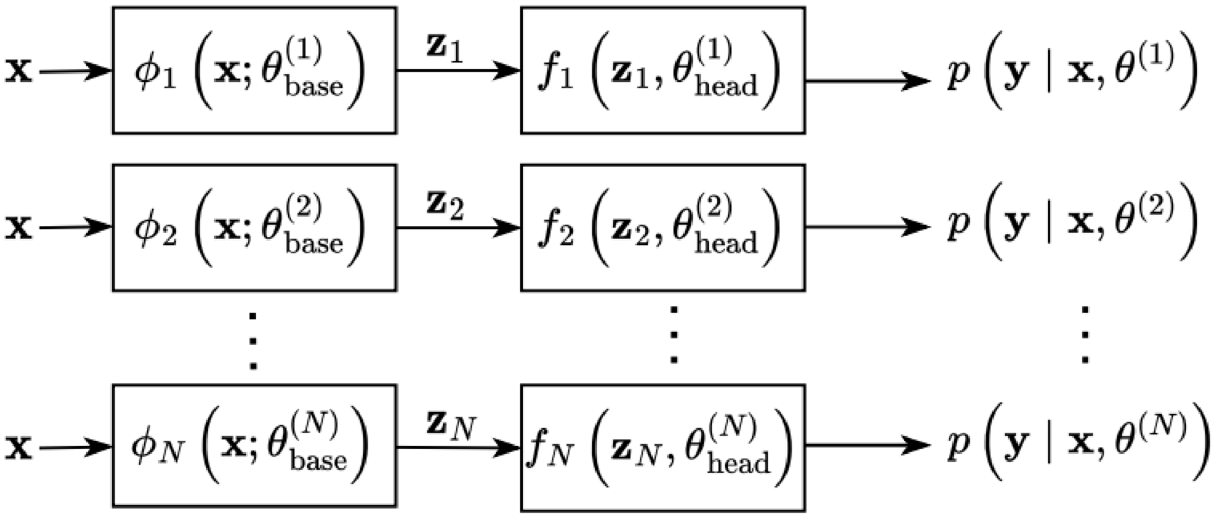
Low uncertainty

⇒ Cannot disentangle aleatoric and epistemic uncertainty

- Instead of Gaussian Process other models have been used as well

Ensemble Methods

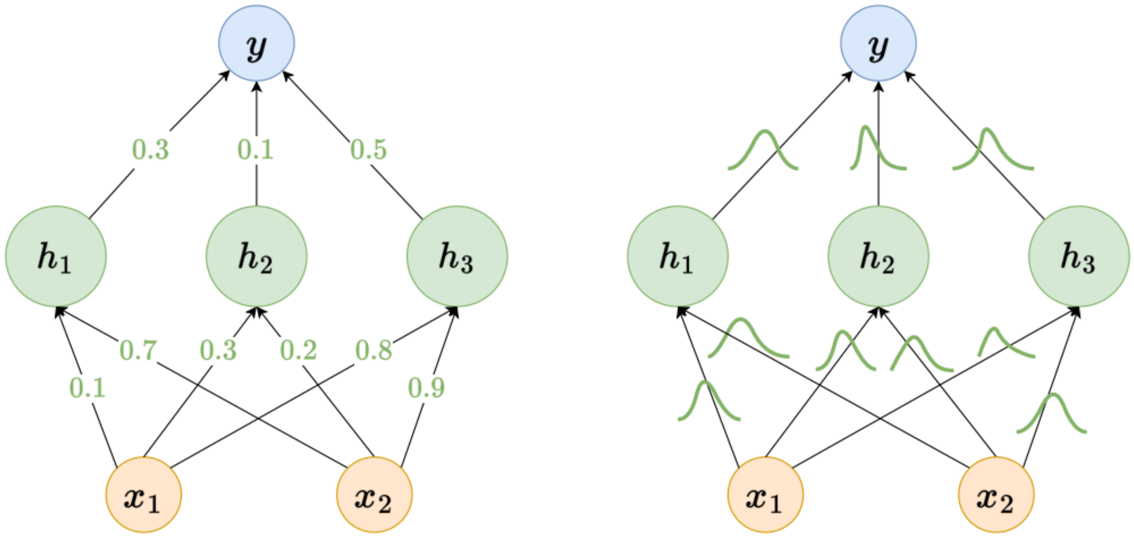
Ensemble Networks



- Several randomly initialized networks are trained
- Prediction/uncertainty estimation: Output of ensemble members is combined

Bayesian neural networks

Bayesian Neural Network



- Network parameters θ
- $\mathcal{D} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{N} = (\mathbf{X}, \mathbf{Y})$ training data
- Posterior: $p(\theta|\mathbf{X}, \mathbf{Y}) \propto \prod_{i=1}^{n} p(\boldsymbol{y}_i | f(\boldsymbol{x}_i; \theta)) \, p(\theta)$
- Prediction: $p(\boldsymbol{y}^* | \boldsymbol{x}^*, \mathcal{D}) = \int p(\boldsymbol{y}^* | f(\boldsymbol{x}^*; \theta)) \, p(\theta | \mathcal{D}) \, \mathrm{d}\theta$

- Integral for prediction is approximated by Monte Carlo averaging
- Posterior distribution is intractable ⇒ approximate inference

Methods for approximating the weight posterior distribution

- Sampling based methods
  - ✓ Hamiltonian-Monte-Carlo (HMC) sampling
  - ✓ Considered as the "gold-standard" solution
  - ✓ Enormous run-time required for good estimate

Methods for approximating the weight posterior distribution

- Sampling based methods
  - ✓ Hamiltonian-Monte-Carlo (HMC) sampling
  - ✓ Considered as the "gold-standard" solution
  - ✓ Enormous run-time required for good estimate

- Variational inference (VI)
  - ✓ Approximate multi-modal posterior with oversimplified tractable distribution (e.g., factorized uni-modal Gaussians)
  - ✓ Limits approximation quality

Methods for approximating the weight posterior distribution

- Sampling based methods
  - ✓ Hamiltonian-Monte-Carlo (HMC) sampling
  - ✓ Considered as the "gold-standard" solution
  - ✓ Enormous run-time required for good estimate

- Variational inference (VI)
  - ✓ Approximate multi-modal weight posterior with oversimplified tractable distribution (e.g., factorized uni-modal Gaussians)
  - ✓ Limits approximation quality
- Particle-optimization-based VI (POVI)
  - ✓ Iteratively updates a set of particles, such that its empirical probability measure approximates the correct posterior

Particle-optimization-based Variational Inference

Particle-optimization-based Variational Inference

- Weight-space particle methods (POVI)
  - ✓ Considers *n* weight configurations of a neural network: $\{\theta^{(i)}\}_{i=1}^{n}$
  - ✓ Weights are updated using gradient of the posterior:

$$\theta_{l+1}^{(i)} \leftarrow \theta_l^{(i)} - \epsilon_l \mathrm{v}(\theta_l^{(i)})$$

Learning rate

$$\text{with} \quad \mathrm{v}(\theta_l^{(i)}) = \nabla_{\theta_l^{(i)}} \log \underbrace{p(\theta_l^{(i)} \mid \mathbf{x})}_{\text{POSTERIOR}}$$

  - ✓ Predictions of members are combined: Bayesian model averaging
  - ✓ Problem: Particles may converge to same mode of posterior

Particle-optimization-based Variational Inference

- Weight-space particle methods (POVI)
  - ✓ Considers *n* weight configurations of a neural network: $\{\theta^{(i)}\}_{i=1}^{n}$
  - ✓ Weights are updated using gradient of the posterior:

$$\theta_{l+1}^{(i)} \leftarrow \theta_l^{(i)} - \epsilon_l \mathrm{v}(\theta_l^{(i)})$$

$$\text{with} \qquad \mathrm{v}(\theta_l^{(i)}) = \nabla_{\theta_l^{(i)}} \log \underbrace{p(\theta_l^{(i)} \mid \mathbf{x})}_{\text{POSTERIOR}}$$

Learning rate

  - ✓ Predictions of members are combined: Bayesian model averaging
  - ✓ Problem: Particles may converge to same mode of posterior
- Repulsive component to maintain diversity (inspired by SVGD)

$$\mathrm{v}(\theta_l^{(i)}) = \nabla_{\theta_l^{(i)}} \log \underbrace{p(\theta_l^{(i)} \mid \mathbf{x})}_{\text{POSTERIOR}} - \underbrace{\mathcal{R}\left(\sum_{j=1}^{n} \nabla_{\theta_l^{(i)}} k\left(\theta_l^{(i)}, \theta_l^{(j)}\right)\right)}_{\text{REPULSION TERM}}$$

  - ✓ e.g. RBF kernel
  - ✓ Gradient of kernel moves particles away from close neighbors

[DAF21] F. D'Angelo, V. Fortuin, “Repulsive Deep Ensembles are Bayesian.” NeurIPS, 2021

Particle-optimization-based Variational Inference

- ✓ Problem: Over-parameterized models may have different weights which map to the same function ⇨ loss of diversity in ensemble

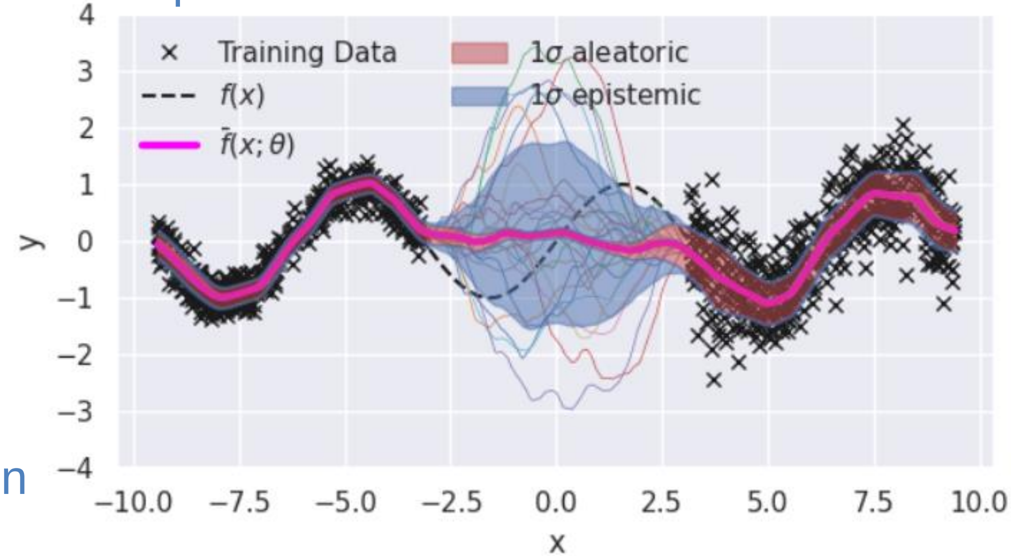
Particle-optimization-based Variational Inference

- ✓ Problem: Over-parameterized models may have different weights which map to the same function ➡ loss of diversity in ensemble

- Function-space particle methods (f-POVI)
  - ✓ Formulation in function space [Wan19]: Particles represent functions

$$f^{(1)}(\mathcal{X}), \ldots, f^{(n)}(\mathcal{X})$$

  - ✓ Function space is parameterized by network $f(\mathcal{X}; \theta_l)$
  - ✓ Optimization requires approximations…
  - ✓ Repulsion term is evaluated at data points



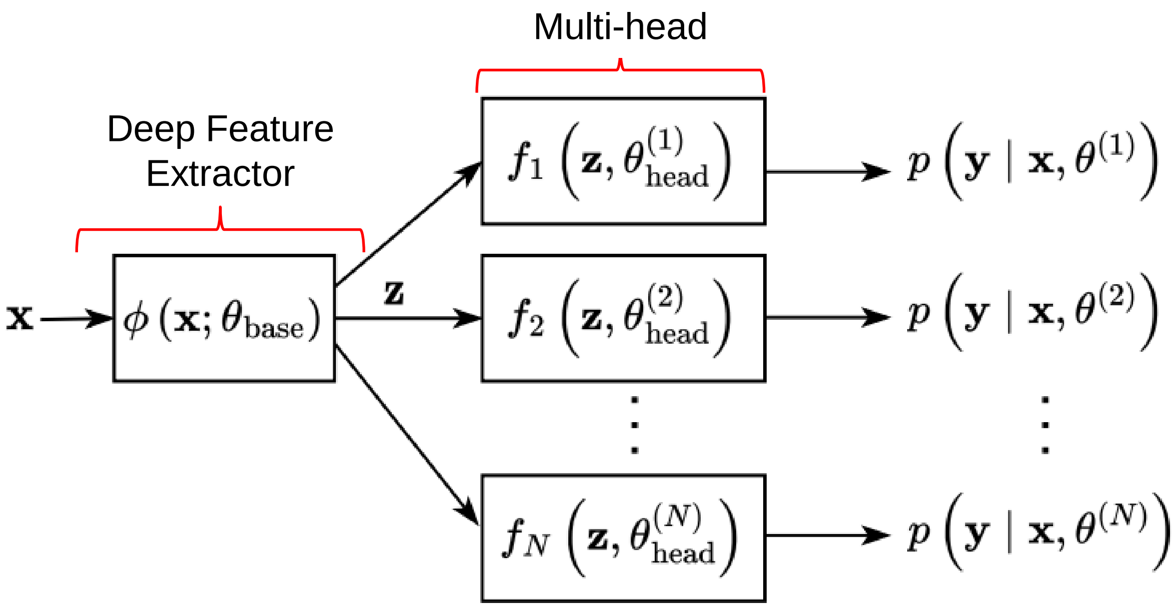
- Functional diversity
  - ✓ Good for predictions
  - ✓ Good for uncertainty estimation

[Wan19] Z. Wang, T. Ren, J. Zhu, and B. Zhang. Function space particle optimization for Bayesian neural networks. ICLR, 2019.

Single Multi-headed Model

Single Multi-headed Model (MH-f-POVI)

- Combining Ideas
  1) Deep feature extractor for input transformation
  2) Function-space POVI on feature space for stochastic output layers



- Model is composed of a shared base model and several heads

$$f^{(i)}(\mathbf{x}; \theta_{\text{base}}, \theta^{(i)}_{\text{head}}) = f^{(i)}_{\text{head}}(\phi(\mathbf{x}; \theta_{\text{base}}); \theta^{(i)}_{\text{head}})$$

- Diverse predictions are enforced by function-space repulsive loss

[So23] S. Steger, B. Klein, H. Fröning and F. Pernkopf. Lightweight Uncertainty Modelling Using Function Space Particle Optimization. submitted, 2023.

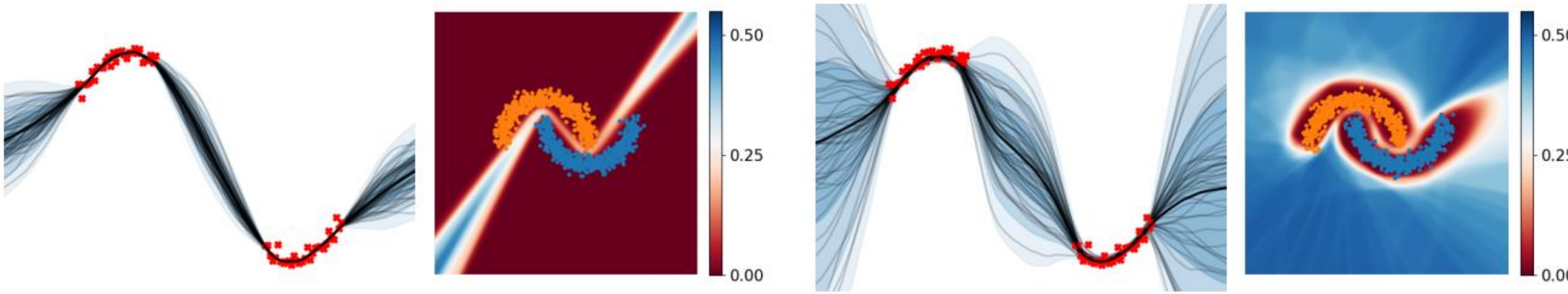
Single Multi-headed Model (MH-f-POVI)

- Advantages

  - ✓ Modelling of aleatoric and epistemic uncertainty; uncertainty can be represented by output heads
  - ✓ Computationally efficient model
  - ✓ We can use pre-trained models (assuming good feature space representation)

Experiment & Results

Synthetic Data



(a) Deep Ensemble (b) MH-f-POVI

Figure 1: Predictions of deep ensembles and the proposed multi-head (MH) network with function space loss (MH-f-POVI). For regression, we show the prediction of single particles, the mean and the standard deviation. For classification on the two-moons data, we show the standard deviation of the predicted probabilities $p(\mathbf{y} \mid \mathbf{x}, \theta)$. Deep ensembles are overly confident in regions without training data, while MH-f-POVI predictions are enforced to be diverse outside of the training data.

Uncertainty and Evaluation Metrics

Uncertainty:

- Single model: softmax entropy $\mathbb{H}[p(\mathbf{y}|\mathbf{x},\theta)]$
- Ensemble models and MH-f-POVI
  - ✓ Uncertainty decomposition: Quantify aleatoric and epistemic uncertainty as [Dep8]:

$$\underbrace{\mathbb{H}[\mathbb{E}_{p(\theta|\mathbf{X},\mathbf{Y})}[p(\mathbf{y}|\mathbf{x},\theta)]]}_{\text{PREDICTIVE ENTROPY}} = \underbrace{\mathbb{E}_{p(\theta|\mathbf{X},\mathbf{Y})}[\mathbb{H}[p(\mathbf{y}|\mathbf{x},\theta)]]}_{\text{ALEATORIC}} + \underbrace{\mathbb{I}[\mathbf{y};\theta \mid \mathbf{x},\mathbf{X},\mathbf{Y}]}_{\text{EPISTEMIC}}$$

[Dep18] S. Depeweg, J.-M. Hernandez-Lobato, F. Doshi-Velez, and S. Udluft. Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive learning. pp. 1184–1193. PMLR, 2018

Uncertainty and Evaluation Metrics

Uncertainty:

- Single model: softmax entropy $\mathbb{H}[p(\mathbf{y}|\mathbf{x},\theta)]$
- Ensemble models and MH-f-POVI
  - ✓ Uncertainty decomposition: Quantify aleatoric and epistemic uncertainty as [Dep18]:

$$\underbrace{\mathbb{H}[\mathbb{E}_{p(\theta|\mathbf{X},\mathbf{Y})}[p(\mathbf{y}|\mathbf{x},\theta)]]}_{\text{PREDICTIVE ENTROPY}} = \underbrace{\mathbb{E}_{p(\theta|\mathbf{X},\mathbf{Y})}[\mathbb{H}[p(\mathbf{y}|\mathbf{x},\theta)]]}_{\text{ALEATORIC}} + \underbrace{\mathbb{I}[\mathbf{y};\theta \mid \mathbf{x},\mathbf{X},\mathbf{Y}]}_{\text{EPISTEMIC}}$$

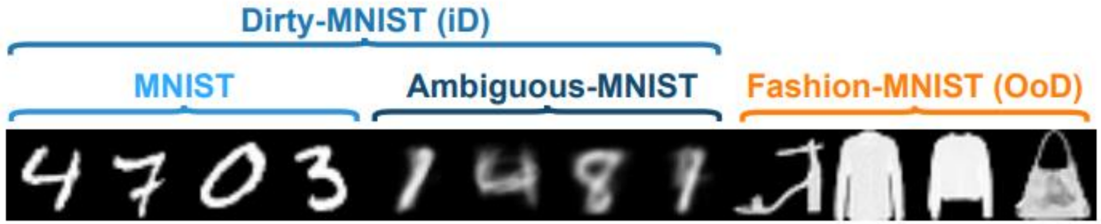
Reliability of uncertainty:

- Ability to detect out-of-domain (OOD) data
- AUROC between correctly identified in-domain (ID) samples and incorrect classified (ID) and OOD samples

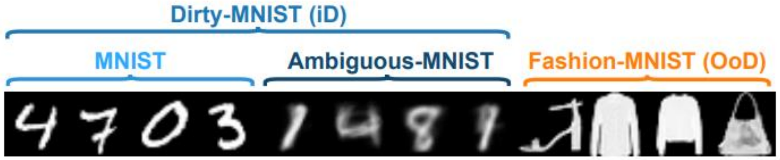
[Dep18] S. Depeweg, J.-M. Hernandez-Lobato, F. Doshi-Velez, and S. Udluft. Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive learning. pp. 1184–1193. PMLR, 2018

Uncertainty Decomposition

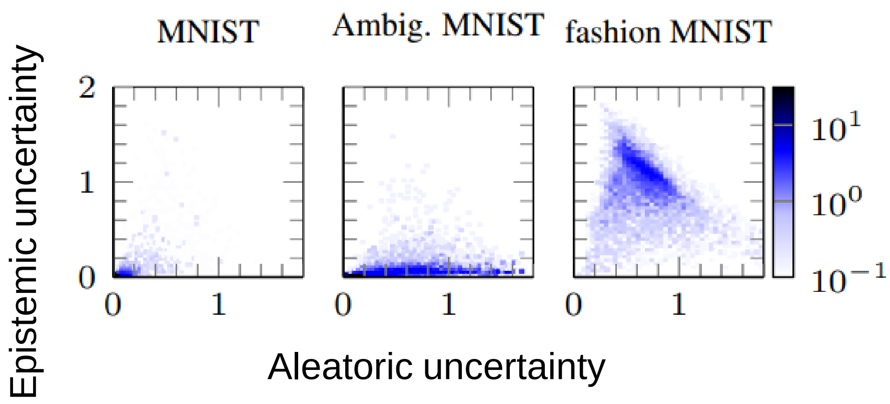
Data



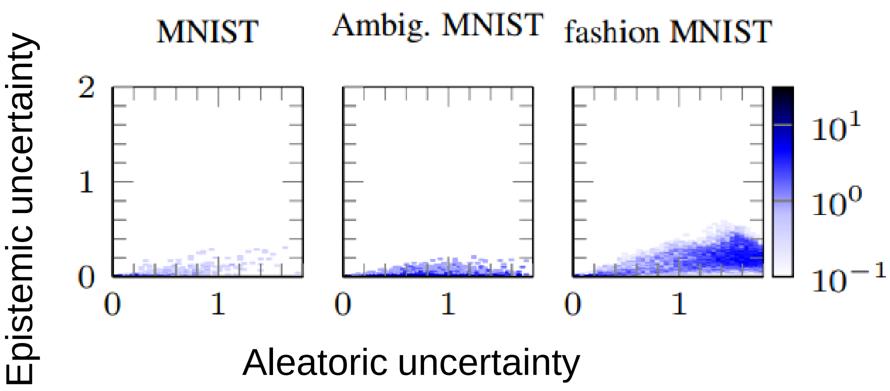
Uncertainty Decomposition



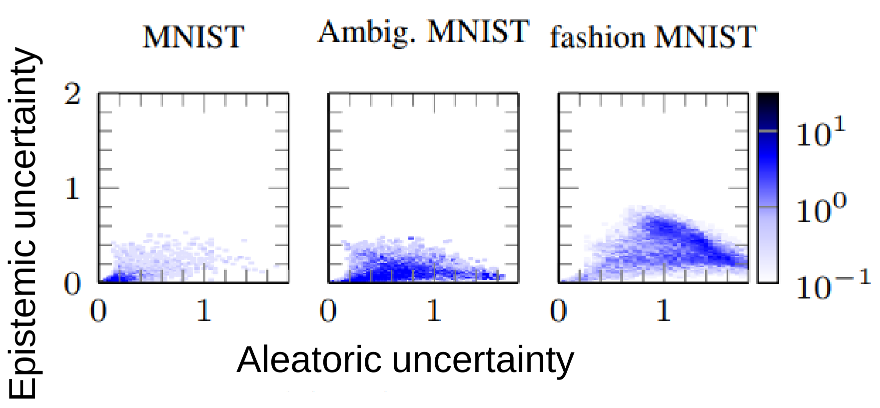
Histograms of aleatoric versus epistemic uncertainty on ID and OOD data



(a) MH-f-POVI (x_C = PATCHES)

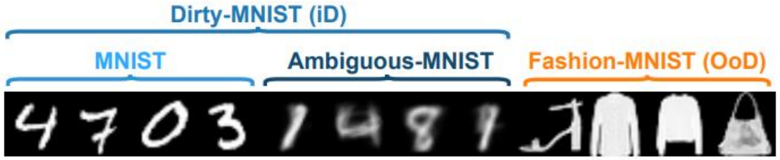


(b) MH-POVI



(c) Deep ensemble

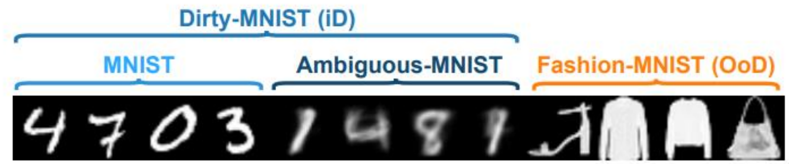
Uncertainty Decomposition



Uncertainty decomposition performance

	Method	Acc. (↑)	Epistemic Uncertainty	MNIST vs f-MNIST. AUROC (↑)	Ambig. vs f-MNIST AUROC (↑)	Param. (↓)
Dirty MNIST (*ResNet-18*)	Single model	98.89%	Softmax Entropy Softmax Density	98.42%$_{\pm 1.03}$ 98.75%$_{\pm 0.72}$	81.80%$_{\pm 7.01}$ 84.33%$_{\pm 5.55}$	100 %
	MH-f-POVI (*ours*) ($\mathbf{x}_C$ = KMNIST)	99.20%	Pred. Entropy Mutual Inf.	*99.76%*$_{\pm 0.07}$ 99.64%$_{\pm 0.10}$	97.84%$_{\pm 0.84}$ *99.52%*$_{\pm 0.17}$	102 %
	MH-f-POVI (*ours*) ($\mathbf{x}_C$ = PATCHES)	99.26%	Pred. Entropy Mutual Inf.	**99.80%**$_{\pm 0.06}$ 99.68%$_{\pm 0.11}$	97.88%$_{\pm 0.82}$ *99.51%*$_{\pm 0.20}$	
	MH-f-POVI (*ours*) ($\mathbf{x}_C$ = NOISE)	99.19%	Pred. Entropy Mutual Inf.	99.64%$_{\pm 0.14}$ 99.53%$_{\pm 0.17}$	94.13%$_{\pm 2.44}$ 98.41%$_{\pm 0.56}$	
	MH-POVI (*ours*)	*99.32%*	Pred. Entropy Mutual Inf.	99.37%$_{\pm 0.50}$ 99.21%$_{\pm 0.36}$	90.53%$_{\pm 4.33}$ 96.49%$_{\pm 1.54}$	
	5-Ensemble	**99.37%**	Pred. Entropy Mutual Inf.	99.55%$_{\pm 0.16}$ 98.62%$_{\pm 0.33}$	92.13%$_{\pm 2.39}$ 92.02%$_{\pm 3.03}$	500 %

Uncertainty Decomposition

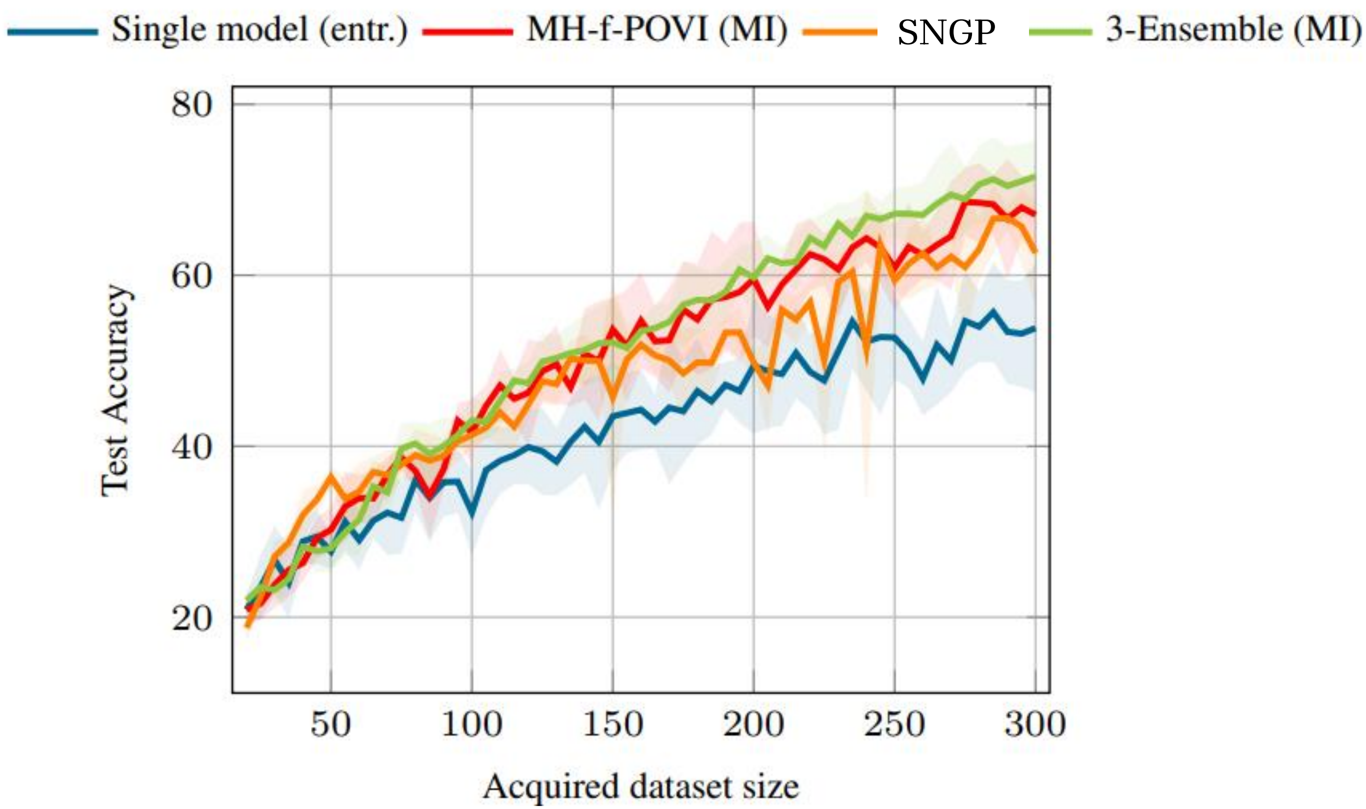


Uncertainty decomposition performance

	Method	Acc. (↑)	Epistemic Uncertainty	MNIST vs f-MNIST. AUROC (↑)	Ambig. vs f-MNIST AUROC (↑)	Param. (↓)
Dirty MNIST (*ResNet-18*)	Single model	98.89%	Softmax Entropy Softmax Density	$98.42\%_{\pm 1.03}$ $98.75\%_{\pm 0.72}$	$81.80\%_{\pm 7.01}$ $84.33\%_{\pm 5.55}$	100 %
	MH-f-POVI (*ours*) ($\mathbf{x}_C$ = KMNIST)	99.20%	Pred. Entropy Mutual Inf.	*99.76%* $_{\pm 0.07}$ $99.64\%_{\pm 0.10}$	$97.84\%_{\pm 0.84}$ *99.52%* $_{\pm 0.17}$	102 %
	MH-f-POVI (*ours*) ($\mathbf{x}_C$ = PATCHES)	99.26%	Pred. Entropy Mutual Inf.	**99.80%** $_{\pm 0.06}$ $99.68\%_{\pm 0.11}$	$97.88\%_{\pm 0.82}$ *99.51%* $_{\pm 0.20}$	
	MH-f-POVI (*ours*) ($\mathbf{x}_C$ = NOISE)	99.19%	Pred. Entropy Mutual Inf.	$99.64\%_{\pm 0.14}$ $99.53\%_{\pm 0.17}$	$94.13\%_{\pm 2.44}$ $98.41\%_{\pm 0.56}$	
	MH-POVI (*ours*)	*99.32%*	Pred. Entropy Mutual Inf.	$99.37\%_{\pm 0.50}$ $99.21\%_{\pm 0.36}$	$90.53\%_{\pm 4.33}$ $96.49\%_{\pm 1.54}$	
	5-Ensemble	**99.37%**	Pred. Entropy Mutual Inf.	$99.55\%_{\pm 0.16}$ $98.62\%_{\pm 0.33}$	$92.13\%_{\pm 2.39}$ $92.02\%_{\pm 3.03}$	500 %

Active Learning

- Training samples are iteratively acquired based on the epistemic uncertainty
- Most informative samples ⟹ high epistemic uncertainty
- After data acquisition, the model is retrained



Summary

Overview of NN methods for uncertainty estimation

- ✓ Single deterministic model
- ✓ Ensemble methods
- ✓ Bayesian neural networks
- ✓ Particle-optimization based variational inference
- ✓ Single multi-headed model

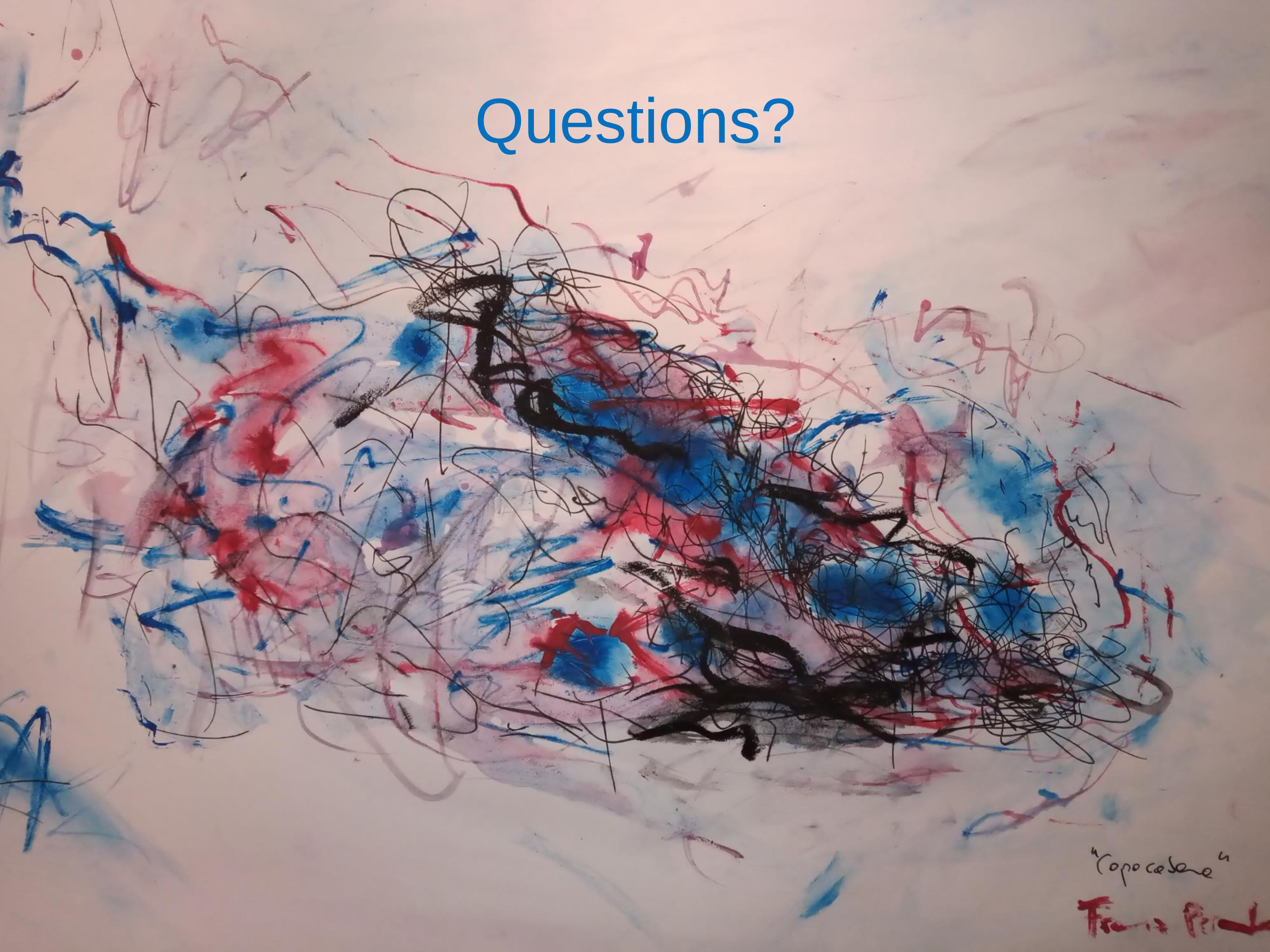
Results

- ✓ Uncertainty decomposition in aleatoric and epistemic uncertainty
- ✓ Multi-head model is able to detect out-of-domain data
- ✓ Active learning scenario
- ✓ Multi-headed model significantly reduce the model size

Intelligent Systems Research Group

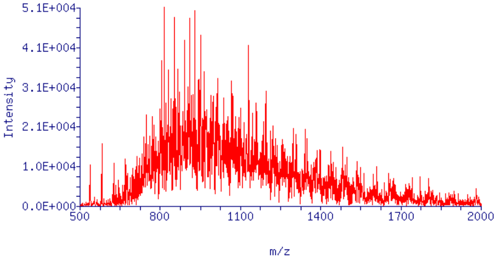
Robust ML Models	Uncertainty Modelling	Health Monitoring of Refractory	Physics-Constrained Neural Networks	Inference on Graphical Models	Hybrid Models & Domain Adaptation	Complex Systems & Explainable AI	Causality
		 		 			
infineon FFG Österreichische Forschungsförderungsgesellschaft	COMET	RHI MAGNESITA	SIEMENS Ingenuity for life	TU Graz	Christian Doppler Forschungsgesellschaft	CERN	TU Graz

Questions?

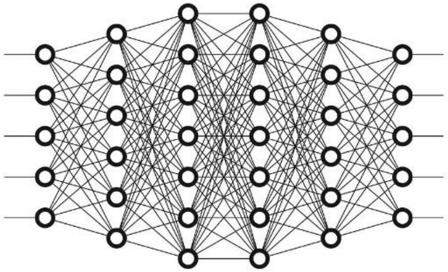


Sources for Uncertainty and Error

- Variabiltiy in the real world

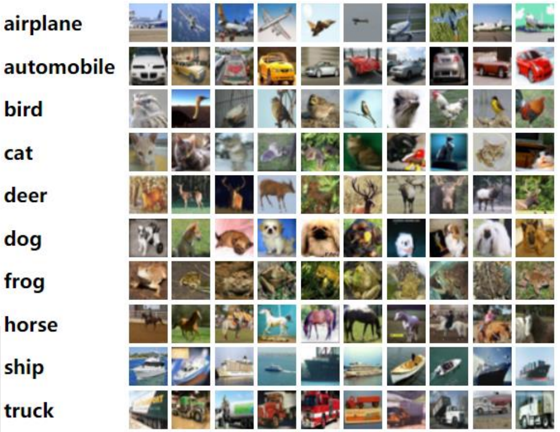


- Error and noise in measurement



- Error in DNN model structure
- Error in training
- Errors caused by unknown data

CIFAR10



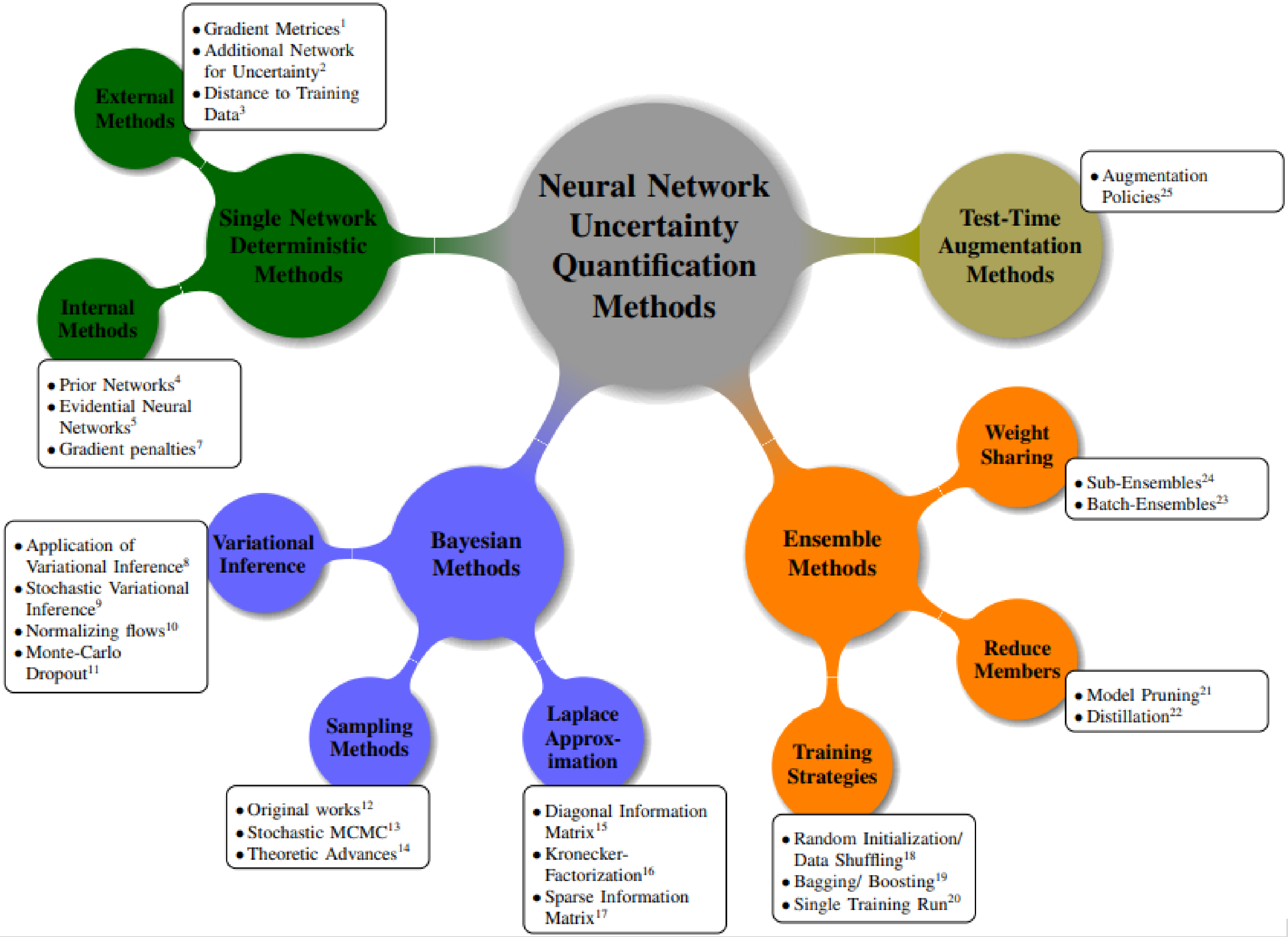
Particle-optimization-based Variational Inference

- Function-space particle methods
  - ✓ Repulsion term is evaluated at data points

- Where does it make sense to evaluate the NN functions for the repulsion term
  - ✓ Low-dimensional data: Evaluate NN on noisy data to cover input domain
  - ✓ High-dimensional data: Adding noise often does not make sense

- Instead of estimating the density of data in high-dimensional input space
  - ✓ Estimate density in feature space
  - ✓ Use Bi-Lipschitz constraints to preserve distance awareness

Methods for estimating uncertainty



[Gaw22] J. Gawlikowski et al. A Survey of Uncertainty in Deep Neural Networks. 2022.

Research Challenges in Machine Learning