HELMHOLTZ AI

TRUST ME, I DON'T KNOW – UNCERTAINTIES IN MACHINE LEARNING PREDICTIONS

Steve Schmerler

helmholtz.ai

elcorto | @elcorto@chaos.social | s.schmerler@hzdr.de / 2025-11-04

Article | Open access | Published: 19 June 2024

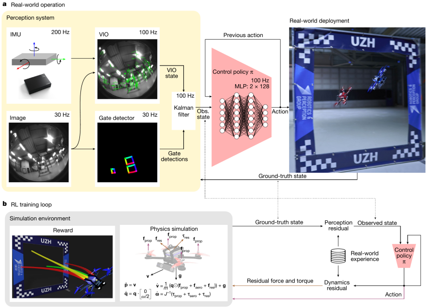
Detecting hallucinations in large language models using semantic entropy

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn & Yarin Gal

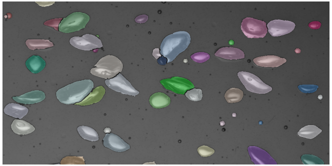
Nature **630**, 625–630 (2024) | Cite this article

Abstract

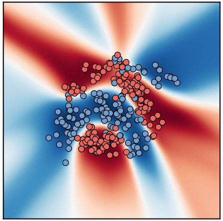
Large language model (LLM) systems, such as ChatGPT[1] or Gemini[2], can show impressive reasoning and question-answering capabilities but often 'hallucinate' false outputs and unsubstantiated answers[3,4]. Answering unreliably or without the necessary information prevents adoption in diverse fields, with problems including fabrication of legal precedents[5] or untrue facts in news articles[6] and even posing a risk to human life in medical domains such as radiology[7]. Encouraging truthfulness through supervision or reinforcement has been only partially successful[8]. Researchers need a general method for detecting hallucinations in LLMs that works even with new and unseen questions to which humans might not know the answer.



RL



segmentation

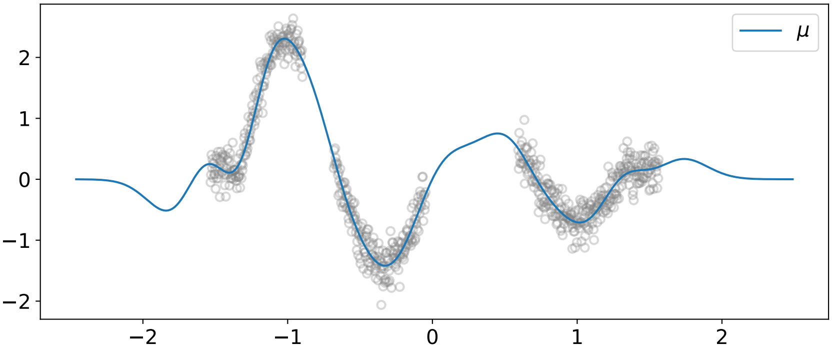


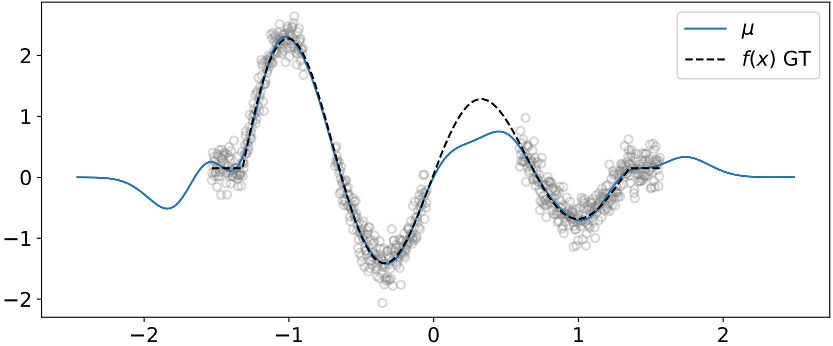
classification

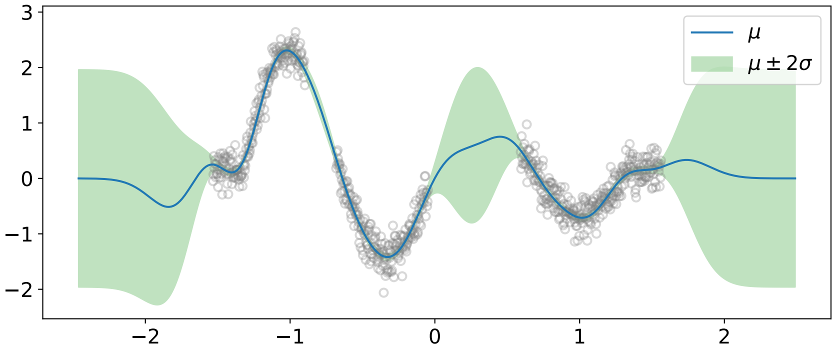
E. Kaufmann et al. "Champion-Level Drone Racing Using Deep Reinforcement Learning". In: *Nature* 620.7976 (2023), pp. 982–987

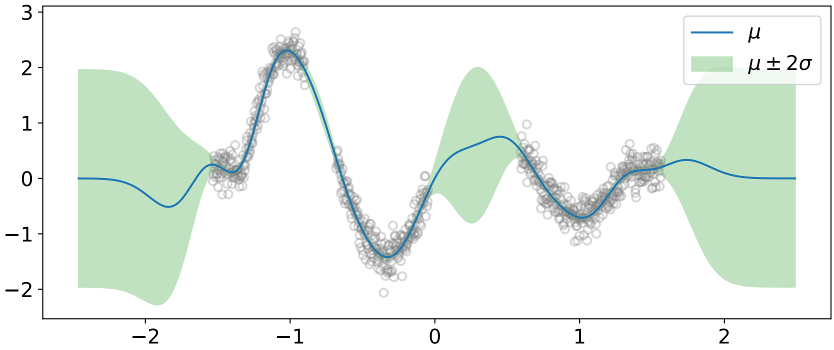
H. Hessenkemper et al. "Bubble Identification from Images with Machine Learning Methods". In: *International Journal of Multiphase Flow* 155 (2022), p. 104169

A. Immer, M. Korzepa, and M. Bauer. "Improving Predictions of Bayesian Neural Nets via Local Linearization". In: *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*. 2021, pp. 703–711

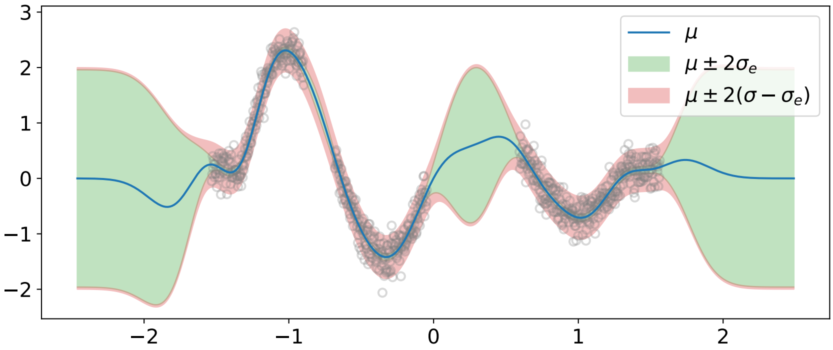

μ
2
1
0
−1
−2
−2
−1
0
1
2


μ
$f(x)$ GT
2
1
0
−1
−2
−2
−1
0
1
2

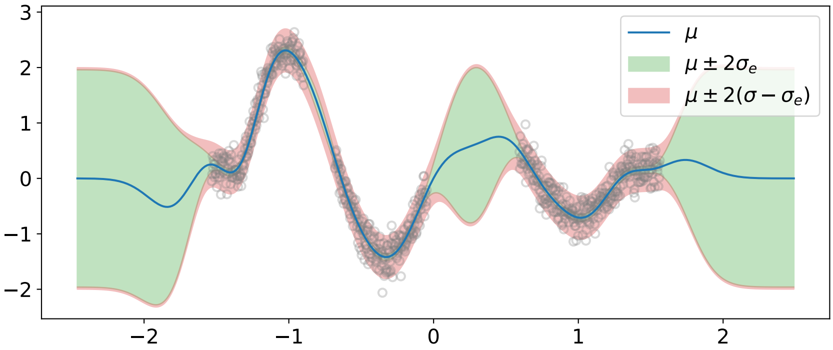

μ
$\mu \pm 2\sigma$
3
2
1
0
−1
−2
−2
−1
0
1
2



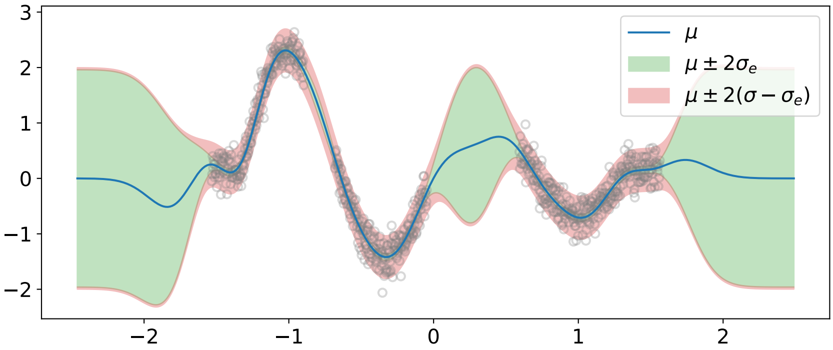
epistemic / model uncertainty: can be reduced by more data



epistemic / model uncertainty: can be reduced by more data



epistemic / model uncertainty: can be reduced by more data
aleatoric / data uncertainty: can *not* be reduced by more data



epistemic / model uncertainty: can be reduced by more data
aleatoric / data uncertainty: can *not* be reduced by more data

The above is obtained by using a Gaussian process, not a NN :-)

Gaussian processes in 1 slide (Bayesian linear regression)

elcorto.github.io/gp_playground

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006
C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006
K. P. Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023

Gaussian processes in 1 slide (Bayesian linear regression)

- model $f_{\boldsymbol{w}}(\boldsymbol{x})$ with parameters/weights $\boldsymbol{w} \in \mathbb{R}^P$ and input $\boldsymbol{x} \in \mathbb{R}^D$

---

elcorto.github.io/gp_playground

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006
C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006
K. P. Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023

Gaussian processes in 1 slide (Bayesian linear regression)

- model $f_{\boldsymbol{w}}(\boldsymbol{x})$ with parameters/weights $\boldsymbol{w} \in \mathbb{R}^P$ and input $\boldsymbol{x} \in \mathbb{R}^D$
- training data $\mathcal{D} = (\mathbf{X} \in \mathbb{R}^{N \times D}, \boldsymbol{y} \in \mathbb{R}^N)$

elcorto.github.io/gp_playground

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006

C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006

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

Gaussian processes in 1 slide (Bayesian linear regression)

- model $f_{\boldsymbol{w}}(\boldsymbol{x})$ with parameters/weights $\boldsymbol{w} \in \mathbb{R}^P$ and input $\boldsymbol{x} \in \mathbb{R}^D$
- training data $\mathcal{D} = (\mathbf{X} \in \mathbb{R}^{N\times D}, \boldsymbol{y} \in \mathbb{R}^N)$
- prior distribution over $\boldsymbol{w}$: $p(\boldsymbol{w})$, e.g. $\mathcal{N}(\boldsymbol{0}, \mathbf{I}_P)$

elcorto.github.io/gp_playground

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006
C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006
K. P. Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023

Gaussian processes in 1 slide (Bayesian linear regression)

- model $f_{\boldsymbol{w}}(\boldsymbol{x})$ with parameters/weights $\boldsymbol{w} \in \mathbb{R}^P$ and input $\boldsymbol{x} \in \mathbb{R}^D$
- training data $\mathcal{D} = (\mathbf{X} \in \mathbb{R}^{N \times D}, \boldsymbol{y} \in \mathbb{R}^N)$
- prior distribution over $\boldsymbol{w}$: $p(\boldsymbol{w})$, e.g. $\mathcal{N}(\boldsymbol{0}, \mathbf{I}_P)$
- likelihood to express a noise model: $p(y|\boldsymbol{x}, \boldsymbol{w}) = \mathcal{N}(f_{\boldsymbol{w}}(\boldsymbol{x}), \sigma_n^2)$

elcorto.github.io/gp_playground

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006
C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006
K. P. Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023

Gaussian processes in 1 slide (Bayesian linear regression)

- model $f_{\boldsymbol{w}}(\boldsymbol{x})$ with parameters/weights $\boldsymbol{w} \in \mathbb{R}^P$ and input $\boldsymbol{x} \in \mathbb{R}^D$
- training data $\mathcal{D} = (\mathbf{X} \in \mathbb{R}^{N \times D}, \boldsymbol{y} \in \mathbb{R}^N)$
- prior distribution over $\boldsymbol{w}$: $p(\boldsymbol{w})$, e.g. $\mathcal{N}(\boldsymbol{0}, \mathbf{I}_P)$
- likelihood to express a noise model: $p(y|\boldsymbol{x}, \boldsymbol{w}) = \mathcal{N}(f_{\boldsymbol{w}}(\boldsymbol{x}), \sigma_n^2)$
- posterior distribution over $\boldsymbol{w}$ via ***Bayesian inference***

$$p(\boldsymbol{w}|\mathbf{X}, \boldsymbol{y}) = \frac{p(\boldsymbol{y}|\mathbf{X}, \boldsymbol{w})\, p(\boldsymbol{w})}{\int p(\boldsymbol{y}|\mathbf{X}, \boldsymbol{w})\, p(\boldsymbol{w})\, \mathrm{d}\boldsymbol{w}}$$

elcorto.github.io/gp_playground

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006
C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006
K. P. Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023

Gaussian processes in 1 slide (Bayesian linear regression)

- model $f_{\boldsymbol{w}}(\boldsymbol{x})$ with parameters/weights $\boldsymbol{w} \in \mathbb{R}^P$ and input $\boldsymbol{x} \in \mathbb{R}^D$
- training data $\mathcal{D} = (\mathbf{X} \in \mathbb{R}^{N \times D}, \boldsymbol{y} \in \mathbb{R}^N)$
- prior distribution over $\boldsymbol{w}$: $p(\boldsymbol{w})$, e.g. $\mathcal{N}(\boldsymbol{0}, \mathbf{I}_P)$
- likelihood to express a noise model: $p(y|\boldsymbol{x}, \boldsymbol{w}) = \mathcal{N}(f_{\boldsymbol{w}}(\boldsymbol{x}), \sigma_n^2)$
- posterior distribution over $\boldsymbol{w}$ via ***Bayesian inference***

$$p(\boldsymbol{w}|\mathbf{X}, \boldsymbol{y}) = \frac{p(\boldsymbol{y}|\mathbf{X}, \boldsymbol{w})\, p(\boldsymbol{w})}{\int p(\boldsymbol{y}|\mathbf{X}, \boldsymbol{w})\, p(\boldsymbol{w})\, \mathrm{d}\boldsymbol{w}}$$

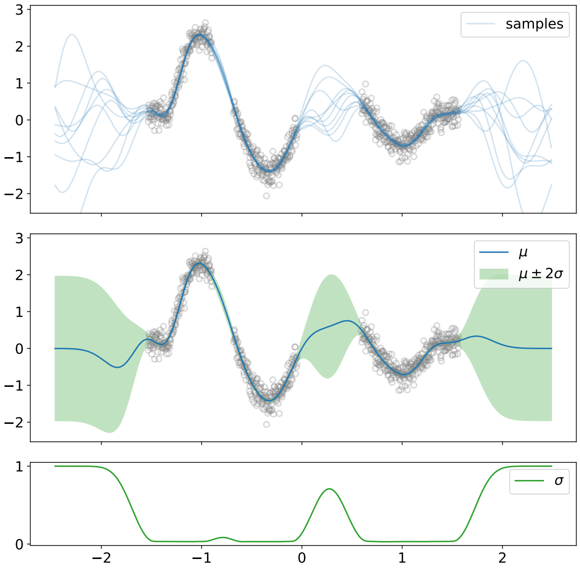
- predictions for new test inputs $\mathbf{X}_* \in \mathbb{R}^{N_* \times D}$: *Bayesian model averaging*

$$\underbrace{p(\boldsymbol{f}_*|\mathbf{X}_*, \mathbf{X}, \boldsymbol{y})}_{\text{posterior predictive}} = \int \underbrace{p(\boldsymbol{f}_*|\mathbf{X}_*, \boldsymbol{w})}_{\text{likelihood}}\, \underbrace{p(\boldsymbol{w}|\mathbf{X}, \boldsymbol{y})}_{\text{posterior}}\, \mathrm{d}\boldsymbol{w} = \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*)$$

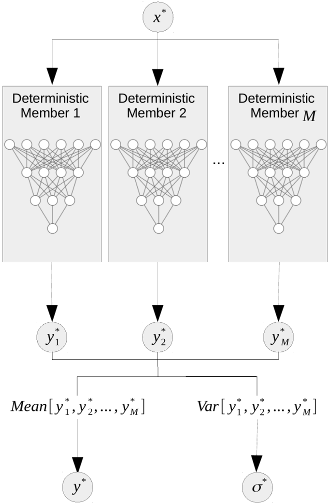
elcorto.github.io/gp_playground

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006
C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006
K. P. Murphy. *Probabilistic Machine Learning: Advanced Topics*. MIT Press, 2023

Gaussian processes: epistemic uncertainty

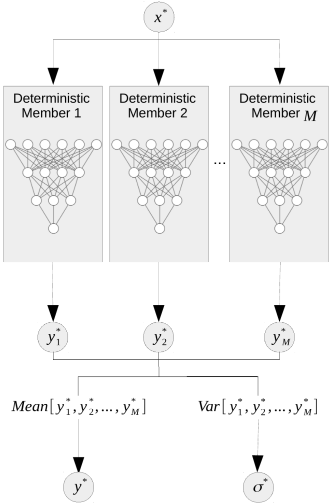


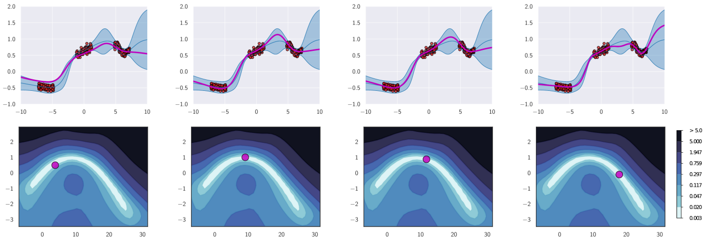
Ensembles of NNs



J. Gawlikowski et al. *A Survey of Uncertainty in Deep Neural Networks*. 2022. URL: http://arxiv.org/abs/2107.03342
A. G. Wilson and P. Izmailov. *Bayesian Deep Learning and a Probabilistic Perspective of Generalization*. Version 4. 2022. URL: http://arxiv.org/abs/2002.08791

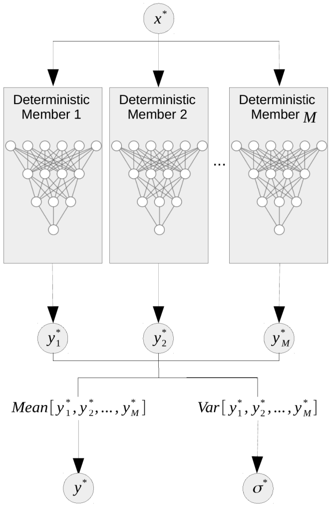
Ensembles of NNs

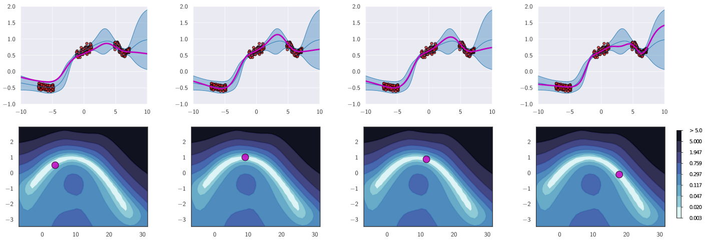




J. Gawlikowski et al. *A Survey of Uncertainty in Deep Neural Networks*. 2022. URL: http://arxiv.org/abs/2107.03342
A. G. Wilson and P. Izmailov. *Bayesian Deep Learning and a Probabilistic Perspective of Generalization*. Version 4. 2022. URL: http://arxiv.org/abs/2002.08791

Ensembles of NNs

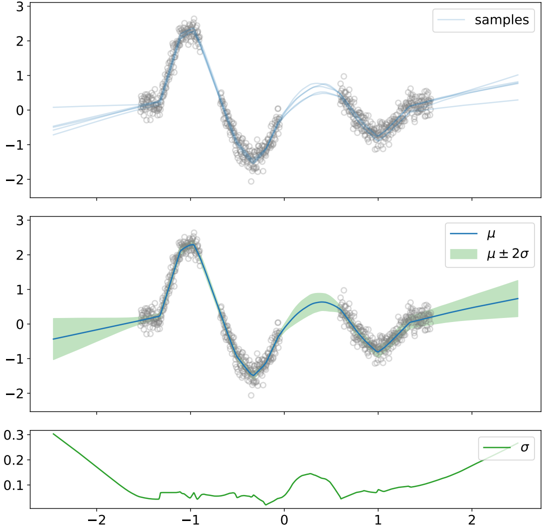




Vary random init + train network $f_{\boldsymbol{w}_i}(\boldsymbol{x})$ = sample weights from unknown posterior: $\boldsymbol{w}_i \sim p(\boldsymbol{w}|\mathcal{D})$

J. Gawlikowski et al. *A Survey of Uncertainty in Deep Neural Networks*. 2022. URL: http://arxiv.org/abs/2107.03342
A. G. Wilson and P. Izmailov. *Bayesian Deep Learning and a Probabilistic Perspective of Generalization*. Version 4. 2022. URL: http://arxiv.org/abs/2002.08791

Ensembles of NNs



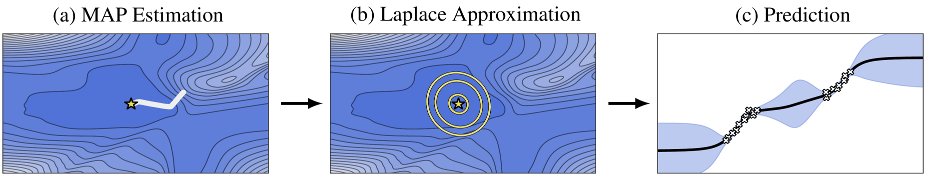
- net: 1-128-128-1
- MSE loss
- train $M = 5$ models, $f_j := f_{\boldsymbol{w}_j}(x), j = 1, \ldots, M$

With shorthand $\langle z \rangle := \frac{1}{M} \sum_j z_j$, we have at each x

$$\mu = \langle f \rangle$$

$$\sigma^2 = \langle f - \mu \rangle^2 = \langle f^2 \rangle - \langle f \rangle^2$$

Laplace approximation



Post-processing step after NN training. With gradient $\boldsymbol{g} = \nabla L|_{\boldsymbol{w}^*}$, Hessian $\mathbf{H} = \partial^2 L|_{\boldsymbol{w}^*}$ and $\boldsymbol{h} = \boldsymbol{w} - \boldsymbol{w}^*$, approximate loss to 2nd order

$$L(\boldsymbol{w}) \approx L(\boldsymbol{w}^*) + \underbrace{\boldsymbol{g}^\top}_{=\boldsymbol{0}} \boldsymbol{h} + \frac{1}{2} \boldsymbol{h}^\top \mathbf{H}\, \boldsymbol{h}$$

Posterior over $\boldsymbol{w}$ (i.e. over models):

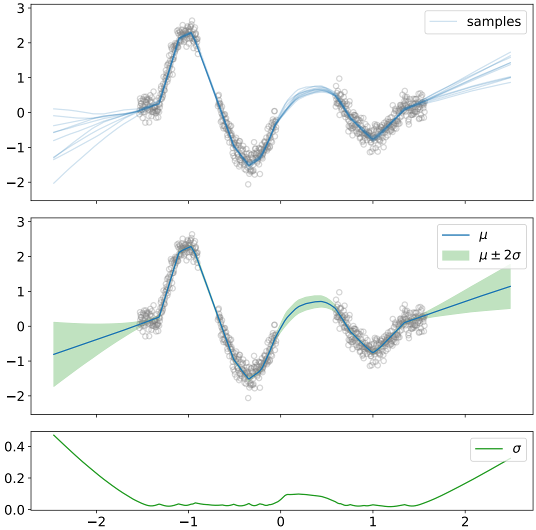
$$p(\boldsymbol{w}|\mathcal{D}) \approx \mathcal{N}(\boldsymbol{w}^*, \boldsymbol{\Sigma}_w) \quad \text{where } \boldsymbol{\Sigma}_w = \mathbf{H}^{-1}$$

---

github.com/aleximmer/Laplace

E. Daxberger et al. *Laplace Redux – Effortless Bayesian Deep Learning*. Version 3. 2022. URL: http://arxiv.org/abs/2106.14806

Laplace approximation



- net: 1-128-128-1
- MSE loss
- train 1 model
- last layer Laplace, full Hessian

Libraries for uncertainty estimation / quantification

Lightning UQ Box

TorchUncertainty

MAPIE

Approximate Bayesian Methods

UQ-Method	Regression	Classification	Segmentation	Pixel Wise Regression
Bayesian Neural Network VI ELBO (BNN_VI_ELBO)	✅	✅	✅	⏳
Bayesian Neural Network VI (BNN_VI)	✅	⏳	⏳	⏳
Deep Kernel Learning (DKL)	✅	✅	❌	❌
Deterministic Uncertainty Estimation (DUE)	✅	✅	❌	❌
Laplace Approximation (Laplace)	✅	✅	❌	❌
Monte Carlo Dropout (MC-Dropout)	✅	✅	✅	✅
Stochastic Gradient Langevin Dynamics (SGLD)	✅	✅	⏳	⏳
Spectral Normalized Gaussian Process (SNGP)	✅	✅	❌	❌
Stochastic Weight Averaging Gaussian (SWAG)	✅	✅	✅	✅
Variational Bayesian Last Layer (VBLL)	✅	✅	❌	❌
Deep Ensemble	✅	✅	✅	✅
Masked Ensemble	✅	✅	⏳	⏳
Density Uncertainty Layer	✅	✅	❌	❌

Single Forward Pass Methods

UQ-Method	Regression	Classification	Segmentation	Pixel Wise Regression
Quantile Regression (QR)	✅	❌	❌	✅
Deep Evidential (DE)	✅	⏳	⏳	✅
Mean Variance Estimation (MVE)	✅	❌	❌	✅
ZigZag	✅	✅	❌	❌
Mixture Density Networks	✅	❌	❌	⏳

Generative Models

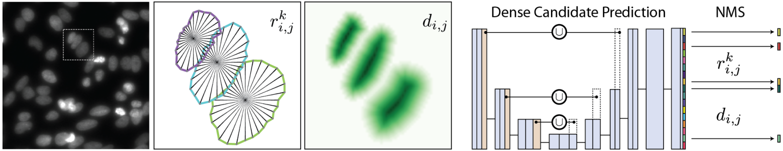
UQ-Method	Regression	Classification	Segmentation	Pixel Wise Regression
Classification And Regression Diffusion (CARD)	✅	✅	❌	❌
Probabilistic UNet	❌	❌	✅	❌
Hierarchical Probabilistic UNet	❌	❌	✅	❌

Post-Hoc methods

UQ-Method	Regression	Classification	Segmentation	Pixel Wise Regression
Test Time Augmentation (TTA)	✅	✅	⏳	⏳
Temperature Scaling	❌	✅	⏳	❌
Conformal Quantile Regression (Conformal QR)	✅	❌	❌	⏳
Regularized Adaptive Prediction Sets (RAPS)	❌	✅	❌	❌
Image to Image Conformal	❌	❌	❌	✅

github.com/lightning-uq-box/lightning-uq-box | github.com/ENSTA-U2IS-AI/torch-uncertainty
github.com/normal-computing/posteriors | github.com/scikit-learn-contrib/MAPIE

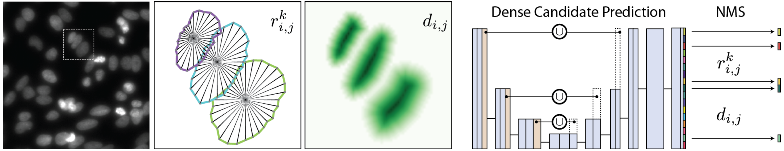
From our work: instance segmentation with StarDist

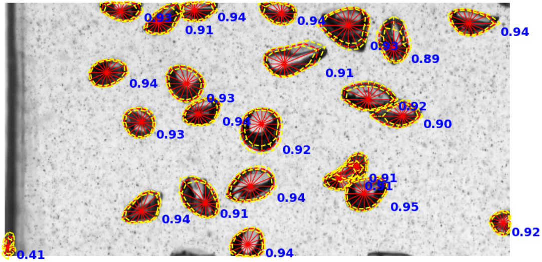


github.com/stardist/stardist

Q. M. K. Siddiqui, S. Starke, and P. Steinbach. "Uncertainty Estimation in Instance Segmentation with Star-convex Shapes". In: *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2024, pp. 1413–1422

From our work: instance segmentation with StarDist





- UQ: ensembles, MC dropout
- clustering of segmentation masks from ensemble/dropout prediction

github.com/stardist/stardist

Q. M. K. Siddiqui, S. Starke, and P. Steinbach. "Uncertainty Estimation in Instance Segmentation with Star-convex Shapes". In: *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2024, pp. 1413–1422

Talk to us!

- helmholtz.ai/you-helmholtz-ai/ai-consulting/
- contact@helmholtz.ai
- consultant-helmholtz.ai@hzdr.de
- mattermost.hzdr.de/uncertainty-quantification
- mattermost.hzdr.de/helmholtz-ai

Slides!

https://doi.org/10.6084/m9.figshare.27891222