Laplace Approximations for Deep Learning

A Bayesian Odyssey in Uncertainty: from Theoretical Foundations to Real-World Applications

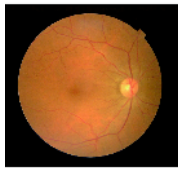
Alexander Immer

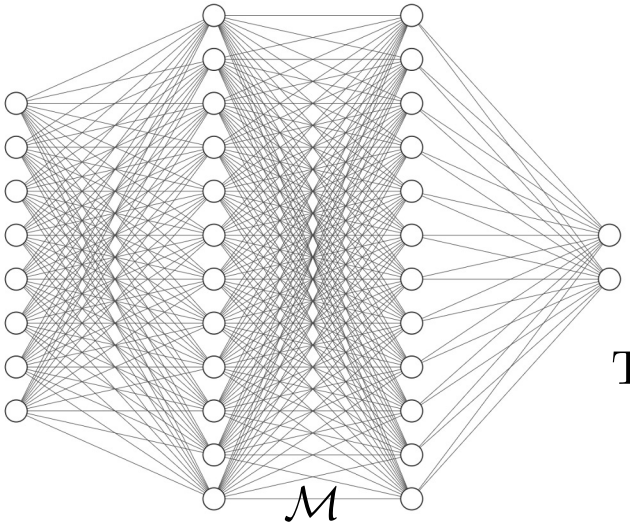
ETH zürich

Max Planck Institute for Intelligent Systems

Deep Learning

Data $\mathcal{D} = \{(\mathrm{x}_n, \mathrm{y}_n)\}_{n=1}^{N}$, weights $\theta \in \mathbb{R}^P$, and model hyperparameters $\mathcal{M}$

$\mathrm{x} =$ 



$\underbrace{\mathrm{f}(\mathrm{x}, \theta) \in \mathbb{R}^K \qquad \mathrm{y} = \text{healthy}}$

Training: $\min_\theta \sum_n \ell(\mathrm{y}_n, \mathrm{f}(\mathrm{x}_n, \theta))$

→ Successful at learning complex patterns from large data sets

[1]Band, Rudner, …, Gal. *Benchmarking Bayesian Deep Learning on Diabetic Retinopathy Detection Tasks.* NeurIPS 2021.

Probabilistic View

$$\min_{\theta} \sum_n \ell(\mathrm{y}_n, \mathrm{f}(\mathrm{x}_n, \theta)) + R(\theta)$$

Data-fit

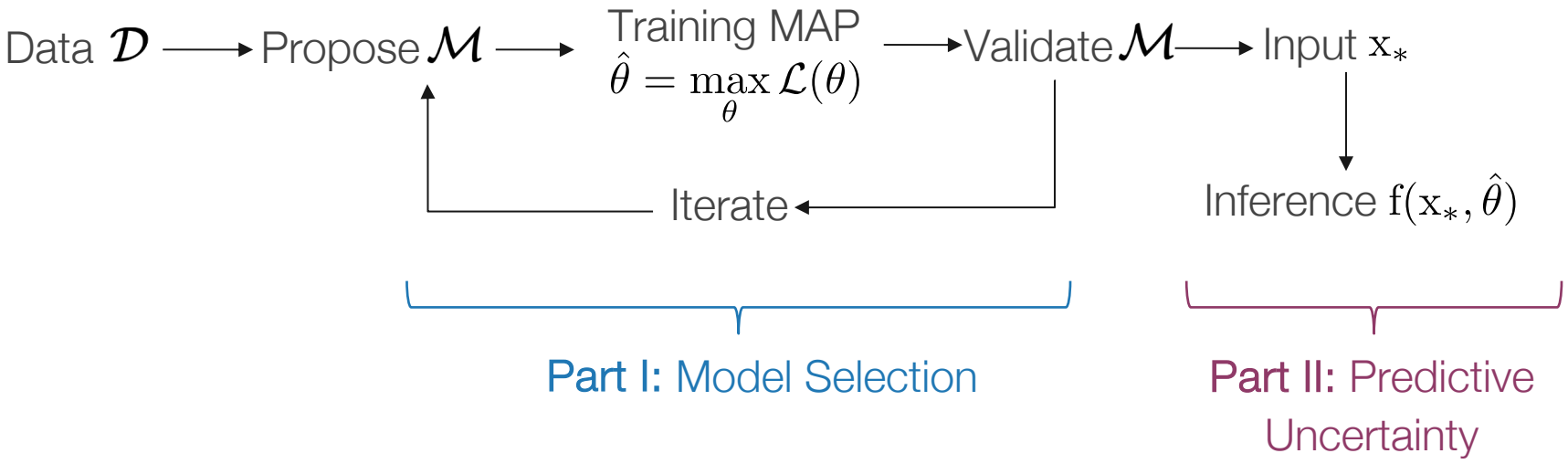
Regularization

$$\max_{\theta} \sum_n \log \underbrace{p(\mathrm{y}_n|\mathrm{x}_n, \theta, \mathcal{M})}_{\text{Likelihood}} + \log \underbrace{p(\theta|\mathcal{M})}_{\text{Prior}}$$

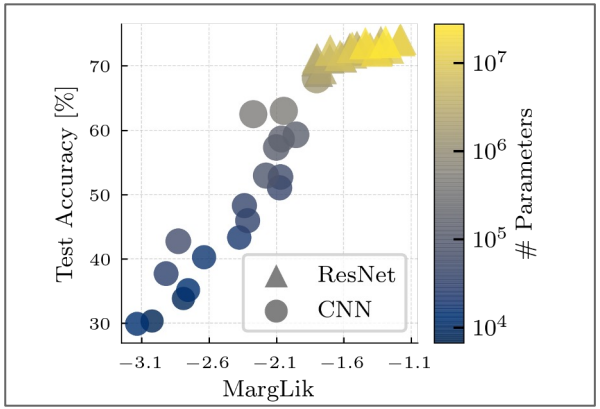
$$\mathcal{L}(\theta)$$

Maximum a-posteriori objective

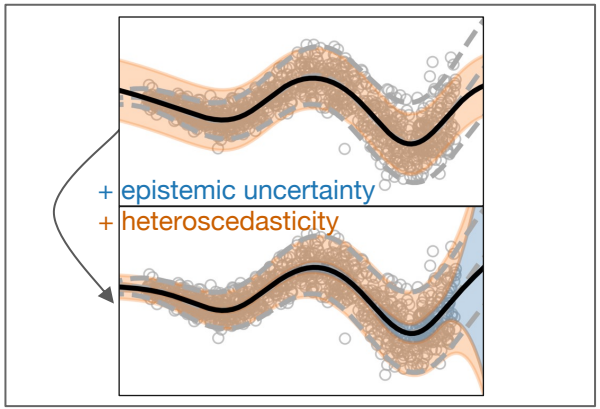
Deep Learning Workflow



Part I: Bayesian Model Selection with Laplace



Part II: Predictive Uncertainties with Laplace



The Bayesian Approach

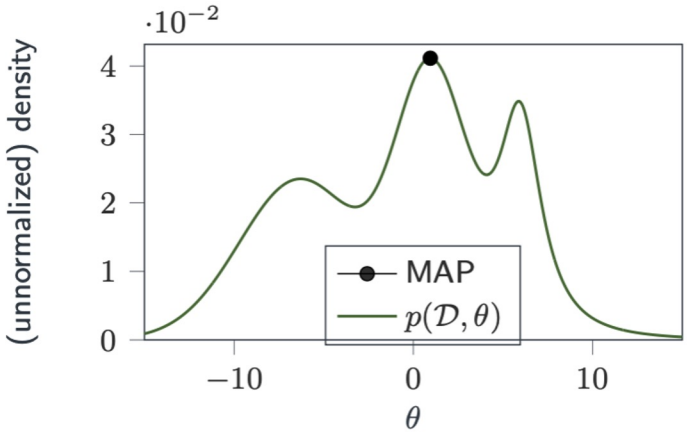
MAP

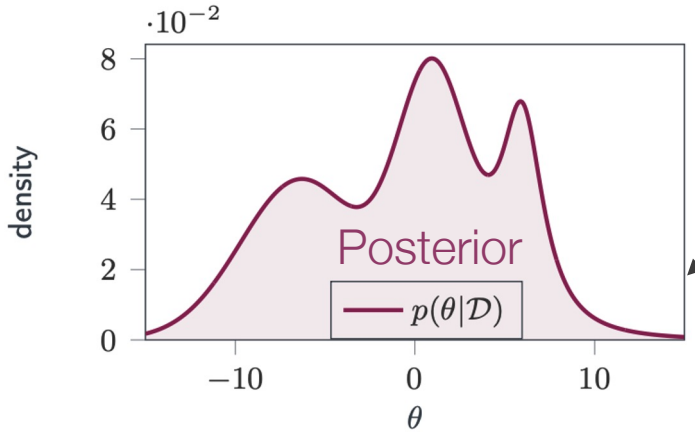
$$\max_{\theta} \underbrace{p(\mathcal{D}|\theta, \mathcal{M})}_{\text{Likelihood}} \underbrace{p(\theta|\mathcal{M})}_{\text{Prior}}$$

divide by

Bayesian

$$\underbrace{\int p(\mathcal{D}|\theta, \mathcal{M})\, p(\theta|\mathcal{M})\, \mathrm{d}\theta}_{\text{Marginal Likelihood}}$$

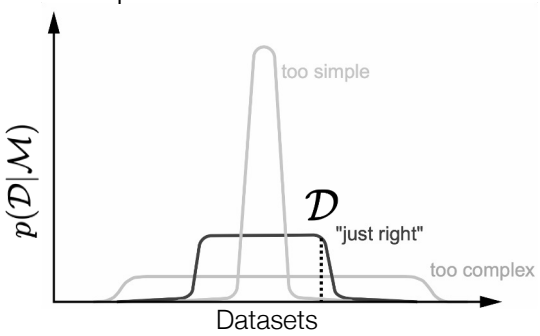




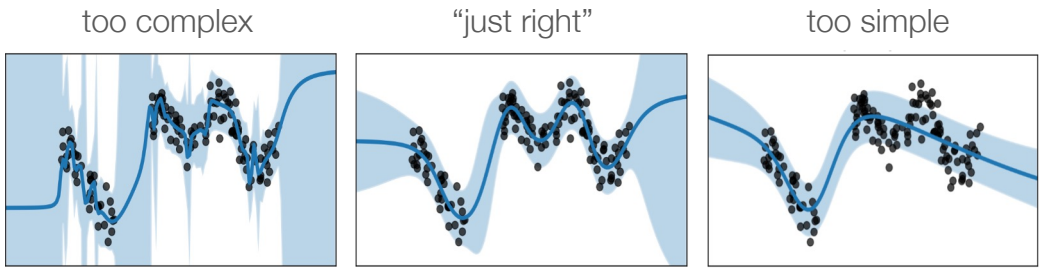
Advantage 1) Bayesian Model Selection

Optimize marginal likelihood: $p(\mathcal{D}|\mathcal{M}) = \int p(\mathcal{D}|\theta, \mathcal{M})\, p(\theta|\mathcal{M})\, \mathrm{d}\theta$

Adapted from Rasmussen et al.



$\mathcal{M}s$

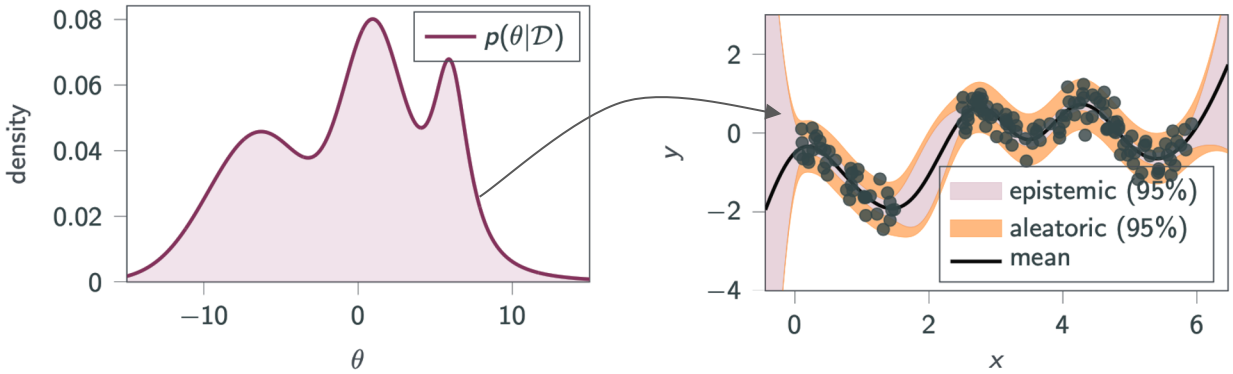


→ Occam's razor: balance data-fit and model complexity[1,2]

[1]Blumer, Ehrenfeucht, Haussler, Warmuth. *Occam's Razor.* Information processing letters 1987.
[2]Rasmussen, Ghahramani: *Occam's Razor.* NIPS 2001.

Advantage 2) Predictive Uncertainty

Posterior predictive: $p(\mathrm{y}_*|\mathrm{x}_*, \mathcal{D}) = \int p(\mathrm{y}_*|\mathrm{x}_*, \theta)\underbrace{p(\theta|\mathcal{D})}_{\text{Posterior}}\,\mathrm{d}\theta$



→ Additionally provide epistemic uncertainty

Aleatoric vs. Epistemic Uncertainty

Potential definition of uncertainties as variance decomposition:

$$\mathbb{V}_{p(\mathrm{y}|\mathrm{x}_*,\mathcal{D})}[\mathrm{y}] = \underbrace{\int \mathbb{V}_{p(\mathrm{y}|\mathrm{x}_*,\theta)}[\mathrm{y}]p(\theta|\mathcal{D})\,\mathrm{d}\theta}_{\text{aleatoric}} + \underbrace{\int \left(\mathbb{E}_{p(\mathrm{y}|\mathrm{x}_*,\theta)}[\mathrm{y}] - \mathbb{E}_{p(\mathrm{y}|\mathrm{x}_*,\mathcal{D})}[\mathrm{y}]\right)^2 p(\theta|\mathcal{D})\,\mathrm{d}\theta}_{\text{epistemic}}$$

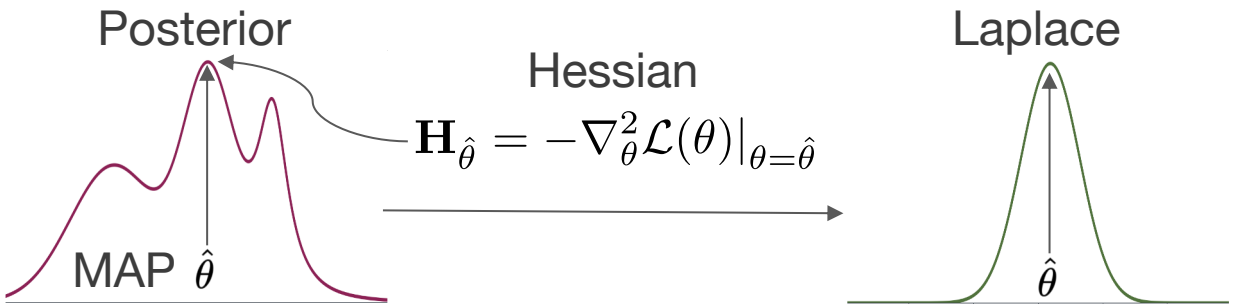
Variance in labels for same input → aleatoric uncertainty

No similar input to x_* → epistemic uncertainty

[1]Kwon, Won, Kim, Paik. *Uncertainty Quantification using Bayesian Neural Networks in Classification.* Computational Statistics & Data Science 2020.

Laplace Approximations

Laplace Approximation



1) MargLik: $\log q(\mathcal{D}|\mathcal{M}) = \log p(\mathcal{D}, \hat{\theta}|\mathcal{M}) - \frac{1}{2}\log|\frac{1}{2\pi}\mathbf{H}_{\hat{\theta}}(\mathcal{M})|$

2) Posterior: $q(\theta) = \mathcal{N}(\theta; \hat{\theta}, \mathbf{H}^{-1}_{\hat{\theta}})$

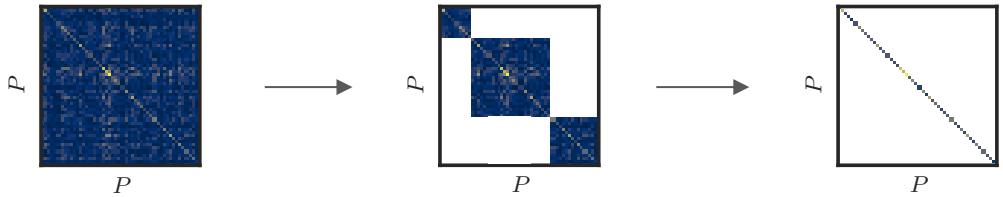
3) Predictive: $q(\mathrm{y}_*|\mathrm{x}_*, \mathcal{D}) = \int p(\mathrm{y}_*|\mathrm{x}_*, \theta)\mathcal{N}(\theta; \hat{\theta}, \mathbf{H}^{-1}_{\hat{\theta}})\,\mathrm{d}\theta$

But: Hessian in $O(P^2)$ not tractable for Deep Learning

[1]Laplace. *Mémoire sur les probabilités*. Mémoires de l'Académie royale des sciences de Paris 1778.
[2]MacKay. *A practical Bayesian framework for backpropagation networks.* Neural Computation 1992.

Efficient Hessian Approximations

Replace full $O(P^2)$ by structured positive-definite approximations $O(P)$[1,2]



Full KFAC[2,3] Diagonal

→ Tractable for extremely large models

[1]Ritter, Botev, Barber. *A Scalable Laplace Approximation for Neural Networks.* ICLR 2018.
[2]Martens, Grosse. *Optimizing Neural Networks with Kronecker-factored Approximate Curvature.* ICML 2015.

Efficient Hessian Approximations

1 Generalized Gauss-Newton (GGN) instead of Hessian:

$$\nabla^2_\theta \log p(\mathcal{D}|\theta) = \nabla^2_\theta \sum_n \log p(\mathrm{y}_n|\mathrm{f}(\mathrm{x}_n, \theta))$$
$$\approx \sum_n \mathbf{J}_\theta(\mathrm{x}_n)^\top \nabla^2_\mathrm{f} \log p(\mathrm{y}_n|\mathrm{f}_n) \mathbf{J}_\theta(\mathrm{x}_n) + \nabla^2_\theta \mathrm{f}(\mathrm{x}_n, \theta)^\top \nabla_\mathrm{f} \log p(\mathrm{y}_n|\mathrm{f}_n)$$

NN Jacobian *O(PK)*

Log loss Hessian w.r.t. NN output function $O(K^2)$

→ Positive-semidefinite and reduced cost

Efficient Hessian Approximations

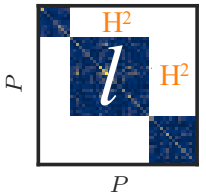
② Decompositions and sampling for Jacobian-vector products:

$$\mathbf{J}_\theta(\mathrm{x})^\top \nabla^2_\mathrm{f} \log p(\mathrm{y}|\mathrm{f}) \mathbf{J}_\theta(\mathrm{x})$$

$= \sum_k \mathrm{v}_k \mathrm{v}_k^T$	GGN	*O(PK)*
$= \mathbb{E}_{p(\mathrm{y}|\mathrm{f})}[\nabla_\mathrm{f} \log p(\mathrm{y}|\mathrm{f})^2]$	Fisher	*O(PS) S≥1*
$\approx \nabla_\mathrm{f} \log p(\mathrm{y}|\mathrm{f})^2$	Empirical Fisher	*O(P)*

→ Reduce cost from *O(PK)* to *O(P)* and not require Jacobian

Efficient Hessian Approximations



③ KFAC[1] is a layer-wise block-diagonal structured approximation:

$$\left[\sum_n \mathbf{J}_\theta(\mathrm{x}_n)^\top \mathbf{\Lambda}(\mathrm{f}_n) \mathbf{J}_\theta(\mathrm{x}_n)\right]_l = \sum_n [\mathrm{a}_{l,n} \otimes \mathrm{g}_{l,n}] \mathbf{\Lambda}(\mathrm{f}_n) [\mathrm{a}_{l,n} \otimes \mathrm{g}_{l,n}]^\top$$

Factors vary with layer type[2]

$$= \underbrace{\sum_n [\mathrm{a}_{l,n} \mathrm{a}_{l,n}^\top] \otimes [\mathrm{g}_{l,n} \mathbf{\Lambda}(\mathrm{f}_n) \mathrm{g}_{l,n}^\top]}_{O(H^4)} \approx \frac{1}{N} \underbrace{\left[\sum_n \mathrm{a}_{l,n} \mathrm{a}_{l,n}^\top\right]}_{O(H^2)} \otimes \underbrace{\left[\sum_n \mathrm{g}_{l,n} \mathbf{\Lambda}(\mathrm{f}_n) \mathrm{g}_{l,n}^\top\right]}_{O(H^2)}$$

→ Reduce storage $O(P^2)$ to $O(P)$ despite having off-diagonal entries!

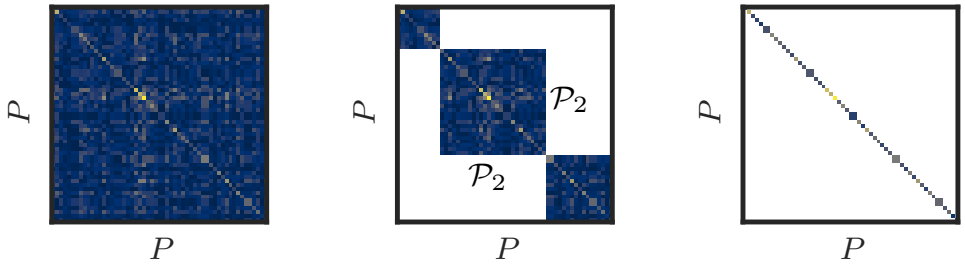
[1]Martens, Grosse. *Optimizing Neural Networks with Kronecker-factored Approximate Curvature.* ICML 2015.
[2]Eschenhagen, Immer, Turner, Schneider, Hennig. *Kronecker-Factored Approximate Curvature for Modern Neural Network Architectures.* NeurIPS 2023.

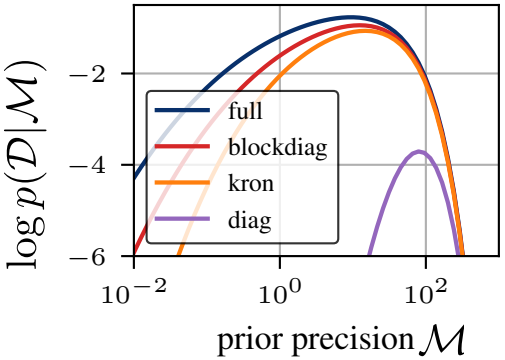
Part I: Bayesian Model Selection for Deep Learning

Structured Laplace Approximations

Structured Hessian approximations yield lower bounds to the Laplace marglik[1]:

$$\mathcal{L} - \tfrac{1}{2}\log|\mathbf{H}| \geq \mathcal{L} - \tfrac{1}{2}\log|\mathbf{H}_{\mathcal{P}}| \geq \mathcal{L} - \tfrac{1}{2}\log|\mathbf{H}_{\mathcal{P}'}|$$





→ Theoretical justification for approximations

Practical Bayesian Model Selection with Laplace

Online Laplace approximation every F steps during training[1]:

1) Gradient-based optimization of differentiable $\mathcal{M}$

$$\mathcal{M} \leftarrow \mathcal{M} + \gamma \nabla_{\mathcal{M}} \log q(\mathcal{D}|\mathcal{M})$$

Gradient-based Selection

→ Can optimize high-dimensional hyperparameters without re-training

2) Comparison of models and checkpoints

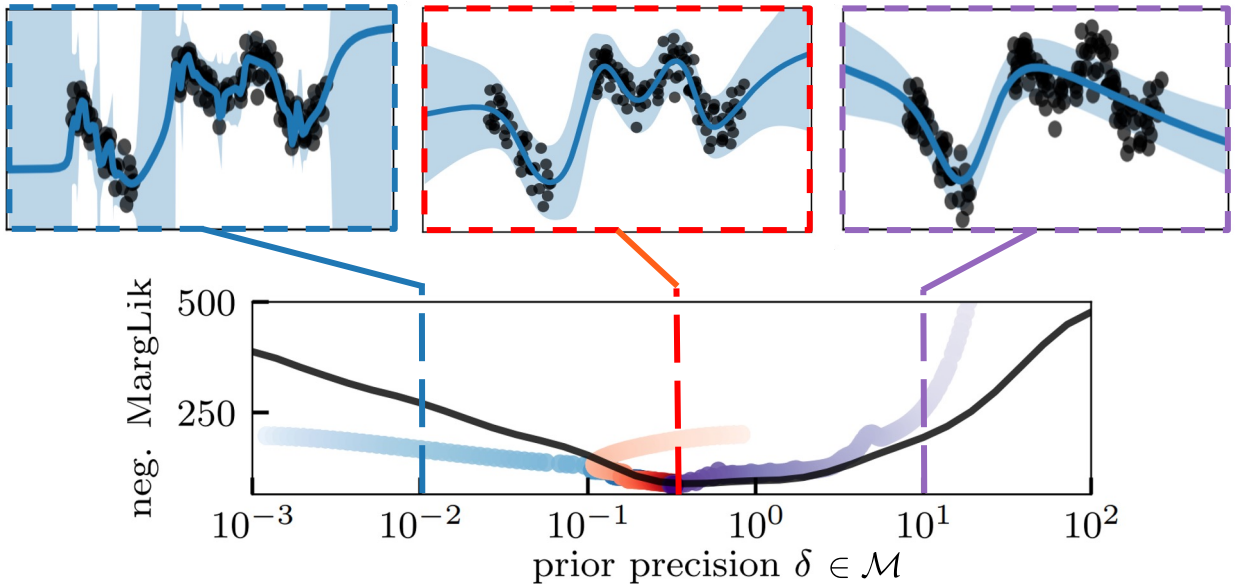
$$\log q(\mathcal{D}|\mathcal{M})$$

Discrete Selection

[1]Immer, Bauer, Fortuin, Rätsch, Khan. *Scalable Bayesian Model Selection for Deep Learning.* ICML 2021.

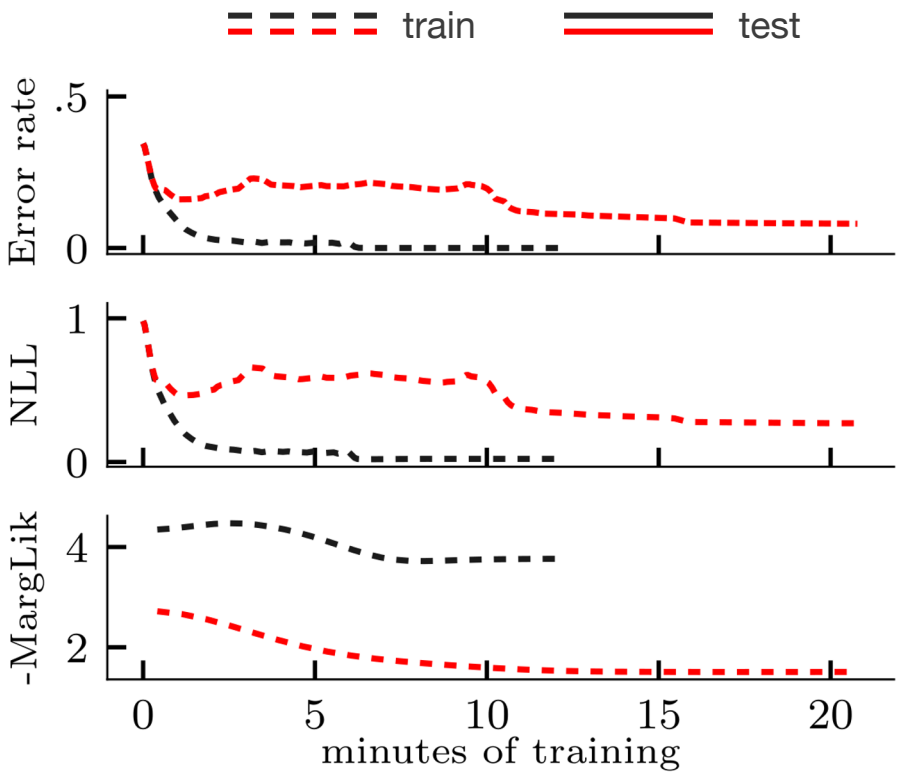
Example: Gradient-Based Model Selection

Optimize prior precision (regularization) with parameters in one training run



→ Validation-based selection requires many iterations

CIFAR-10 Classification



$\mathcal{M}$ = regularization per layer

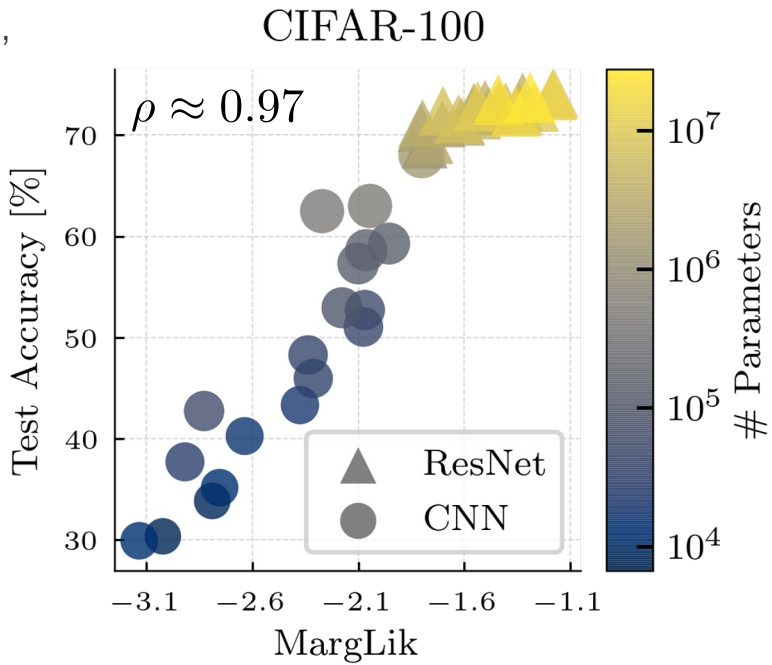
Compare train performance:
Standard vs MargLik

Compare test performance:
→ MargLik generalizes better

→ Optimize thousands of regularization parameters in only 2x runtime

Marginal Likelihood for Architecture Comparison

$\mathcal{M}$ = layer-wise prior precision, architecture



→ Discrete selection possible after optimizing the prior

Examples of Bayesian Model Selection

- Regularization: weight decay/prior precision

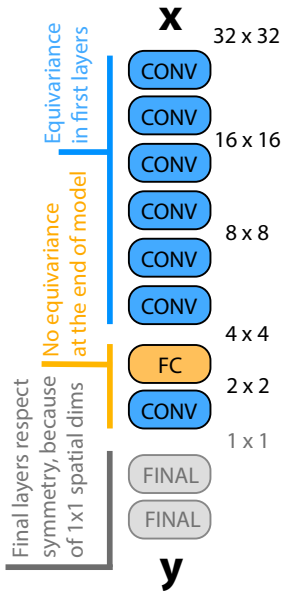
Immer, Bauer, Fortuin, Rätsch, Khan. *Scalable Bayesian Model Selection for Deep Learning.* ICML 2021.
Antoran, Barbano, …, Jin. *Uncertainty Estimation for Computed Tomography with a Linearised Deep Image Prior.* TMLR 2023.
Dhahri, Immer, Charpentier, Günnemann, Fortuin. *Shaving Weights with Occam's Razor*. Preprint 2024.

- Selecting architecture, structure, variables

Zhou*, Yang*, Wang, Pan. *BayesNAS: A Bayesian Approach for Neural Architecture Search.* CVPR 2019.
van der Ouderaa, Immer, van der Wilk. *Learning Layer-wise Equivariances Automatically using Gradients.* NeurIPS 2023.
Bouchiat, Immer, Yeche, Rätsch, Fortuin. *Improving Neural Additive Models with Bayesian Principles.* ICML 2024.

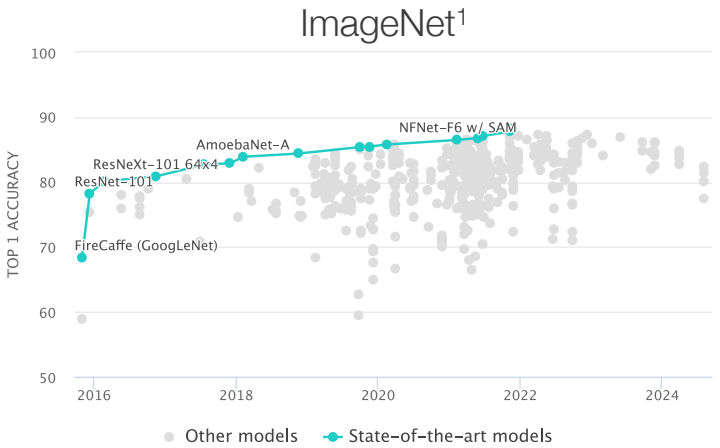
- Learning invariances/data augmentation

van der Wilk, Bauer, John, Hensman. *Learning Invariances using the Marginal Likelihood*. NeurIPS 2018.
Immer*, van der Ouderaa*, Fortuin, Rätsch, van der Wilk. *Invariance Learning in Deep Neural Networks […].* NeurIPS 2022.



Limitations of Bayesian Model Selection

- Required probabilistic model
- Can still require manual tuning or validation
- No benefit on saturated benchmarks

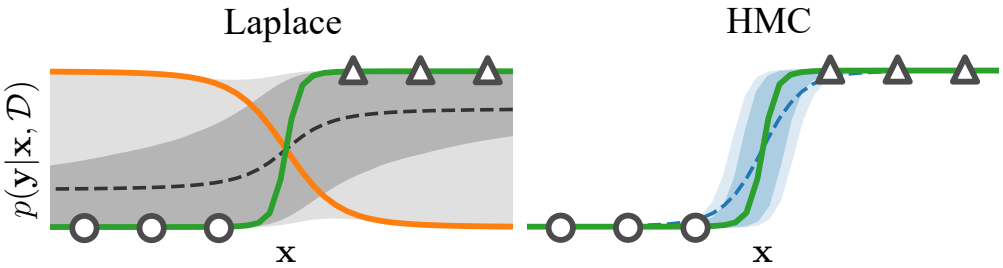


[1]https://paperswithcode.com/sota/image-classification-on-imagenet accessed on 18th Sept 2024.

Part 2: Predictive Uncertainties

Laplace Posterior Predictive Underfitting

$$q(\mathrm{y}_*|\mathrm{x}_*, \mathcal{D}) = \int p(\mathrm{y}_*|\mathrm{x}_*, \theta)\mathcal{N}(\theta; \hat{\theta}, \mathbf{H}_{\hat{\theta}}^{-1})\,\mathrm{d}\theta$$



Problem: Laplace posterior predictive underfits[1]

[1]Lawrence. *Variational Inference in Probabilistic Models.* PhD Thesis 2001.

Linearized Laplace

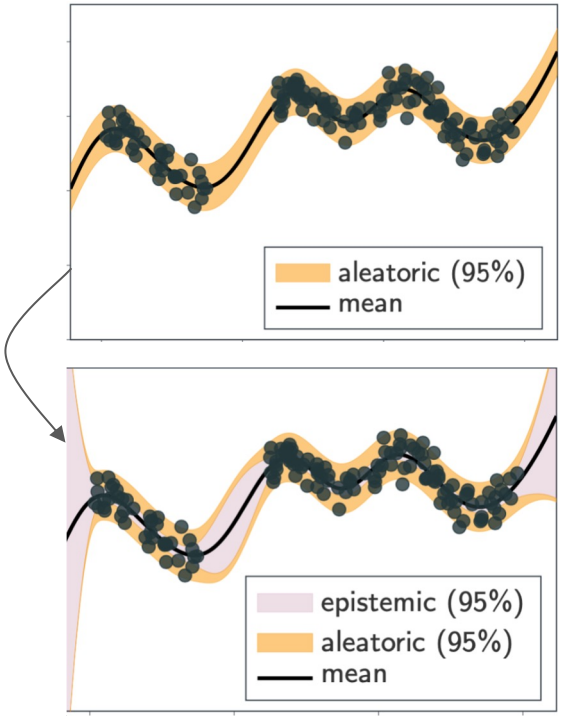
Hessian approximations linearize implicitly[2]

$$\mathrm{f}^{\mathrm{lin}}_{\hat{\theta}}(\mathrm{x}, \theta) = \mathrm{f}(\mathrm{x}, \hat{\theta}) + \mathbf{J}_{\hat{\theta}}(\mathrm{x})(\theta - \hat{\theta})$$

We should keep this in mind for predictions:

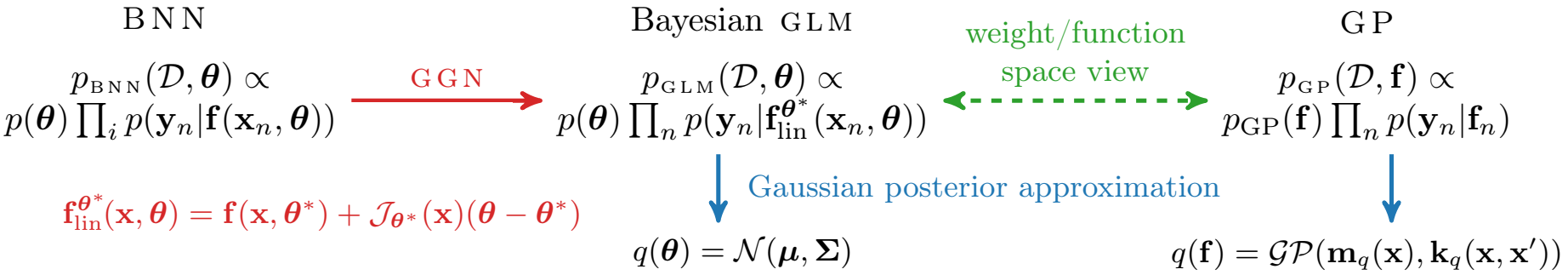
$$q(\mathrm{y}_* | \mathrm{x}_*, \mathcal{D}) \approx \int p(\mathrm{y}_* | \mathrm{f}^{\mathrm{lin}}_{\hat{\theta}}(\mathrm{x}_*, \theta)) \mathcal{N}(\theta; \hat{\theta}, \mathbf{H}^{-1}_{\hat{\theta}}) \, \mathrm{d}\theta$$

→ Stabilizes and improves Laplace predictive



[1]Lawrence. *Variational Inference in Probabilistic Models.* PhD Thesis 2001.
[2]Bottou, Curtis, Nocedal. *Optimization methods for large-scale machine learning.* SIAM Review 2018.

Linearized Model as a Gaussian Process



→ Enables different posterior approximation techniques

Immer, Korzepa, Bauer. *Improving predictions of Bayesian neural nets via local linearization.* AISTATS 2021.
Deng, Zhou, Zhu. *Accelerated Linearized Laplace Approximation for Bayesian Deep Learning*. NeurIPS 2022.

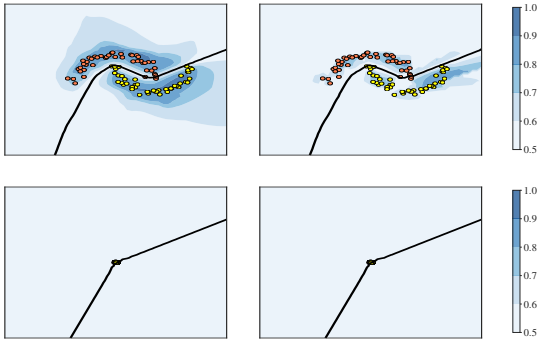
Laplace Posterior Predictive

- Last-layer is very cheap and effective baseline

Kristiadi, Hein, Hennig. *Being Bayesian, even just a bit, fixes overconfidence in relu networks.* ICML 2020.
Ober, Rasmussen. *Benchmarking the neural linear model for regression*. AABI 2019.

- Laplace can improve ensembling further

Eschenhagen, Daxberger, Hennig, Kristiadi. *Mixtures of Laplace Approximations […].* BDL@NeurIPS 2021.



(c) Bayesian (last-layer) (d) Bayesian (all-layer)

- Linearized predictive can have a closed-form (approximation)

Immer, Korzepa, Bauer. I*mproving predictions of Bayesian neural nets via local linearization.* AISTATS 2021.
Deng, Zhou, Zhu. *Accelerated Linearized Laplace Approximation for Bayesian Deep Learning*. NeurIPS 2022.

Limitations

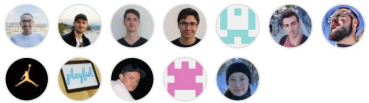
- Scalability issues in some cases[1]
- Linearization can be expensive[2]
- Post-hoc cannot fix badly trained network

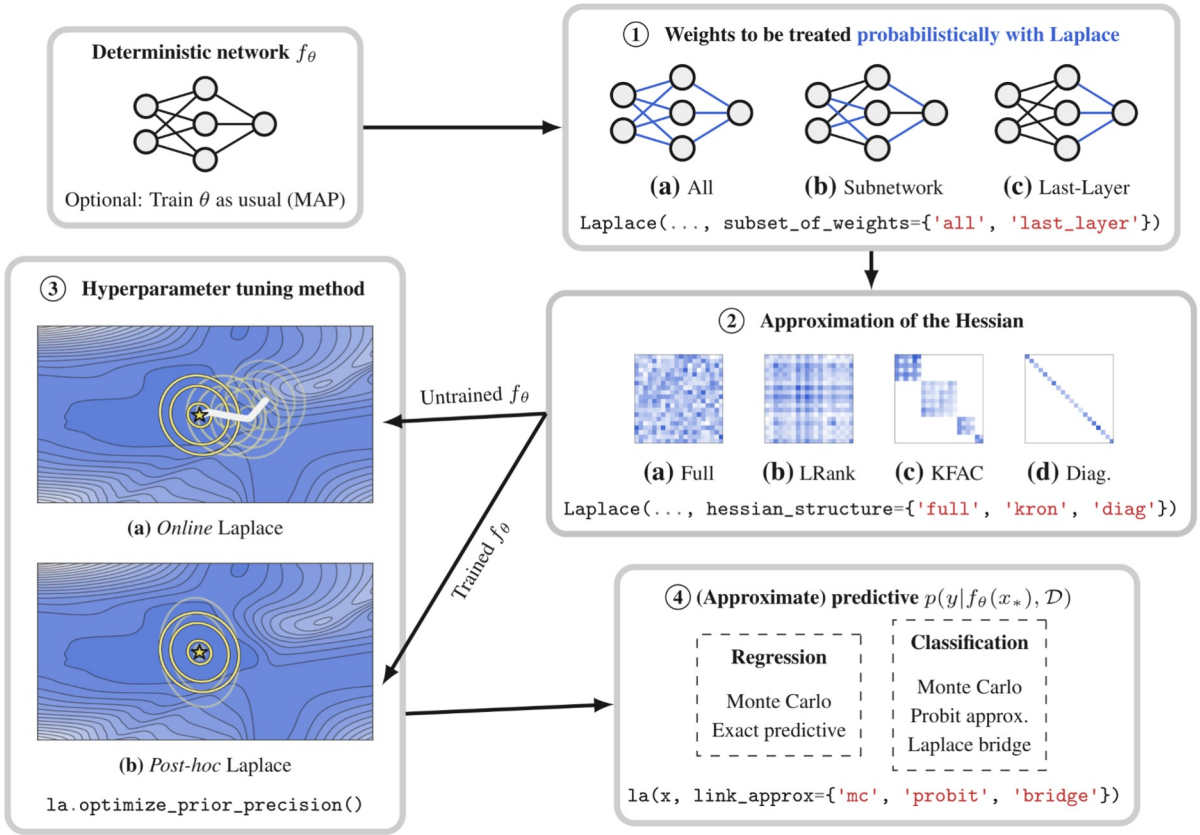
[1]Kepf, Wanna, Miani, Moore, …, Warburg. *Laplacian Segmentation Networks […].* MICCAI 2024.
[2]Antoran, Janz, Allingham, …, Hernandez-Lobato. *Adapting the Linearised Laplace Model Evidence for Modern Deep Learning.* ICML 2022.

laplace-torch[1] on github.com/AlexImmer/Laplace

Fork 70 | Star 447

Contributors 12





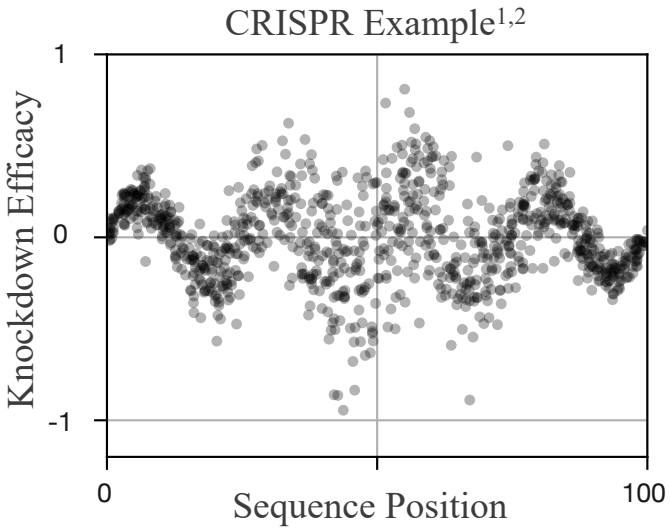
[1]Daxberger*, Kristiadi*, Immer*, Eschenhagen*, Bauer, Hennig. *Laplace Redux - Effortless Bayesian Deep Learning.* NeurIPS 2021.

Thank you!

Co-Authors

Matthias Bauer
Kouroche Bouchiat
Peter Bühlmann
Bertrand Charpentier
Ryan Cotterell
Erik Daxberger
Rayen Dhahri
Runa Eschenhagen
Vincent Fortuin
Stephan Günnemann
Philipp Hennig
Lucas Torroba Hennigen
Francis Jacob
Mohammad Emtiyaz Khan
Maciej Korzepa
Agustinus Kristiadi
Kjong-Van Lehmann
Alexander Marx
Emanuele Palumbo
Gunnar Rätsch
Frank Schneider
Bernhard Schölkopf
Christoph Schultheiss
Stefan Stark
Richard Turner
Tycho van der Ouderaa
Mark van der Wilk
Julia Vogt
Hugo Yèche

Estimating Heteroscedastic Aleatoric Uncertainty



Model mean $\mu(\mathrm{x})$ **and** variance $\sigma^2(\mathrm{x})$ of a Gaussian likelihood:

$$\log p(\mathrm{y}|\mathrm{x},\mu,\sigma^2) \propto -\tfrac{1}{2}\log\sigma^2(\mathrm{x}) - \frac{(\mathrm{y}-\mu(\mathrm{x}))^2}{2\sigma^2(\mathrm{x})}$$

Objective balances mean and variance

→ Hard to optimize and regularize[2,3]

[1]Wessels, Mendez-Mancilla, …, Sanjana. *Massively parallel Cas13 screens reveal principles for guide RNA design*. Nature Biotechnology 2020.
[2]Stirn, Wessels, …, Knowles. *Faithful Heteroscedastic Regression with Neural Networks*. AISTATS 2023.
[3]Seitzer, Tavakoli, Antic, Martius. *On the Pitfalls of Heteroscedastic Uncertainty Estimation with Probabilistic Neural Networks.* ICLR 2022.

Problem with Mean-Variance Parameterization

Mean-variance parameterization does not yield concave log likelihood:

$$\frac{\partial^2 \log \mathcal{N}(\mathrm{y}|\mu, \sigma^2)}{\partial \sigma^2 \partial \sigma^2} = \frac{\sigma^2 - 2(\mu - \mathrm{y})^2}{2\sigma^6}$$

Can be positive or negative.
→ potentially indefinite Hessian
→ not concave

→ Cannot apply Hessian or Laplace approximations naively

Revisiting the Natural Parameterization

Natural parameterization always yields a concave log likelihood[1,2] with

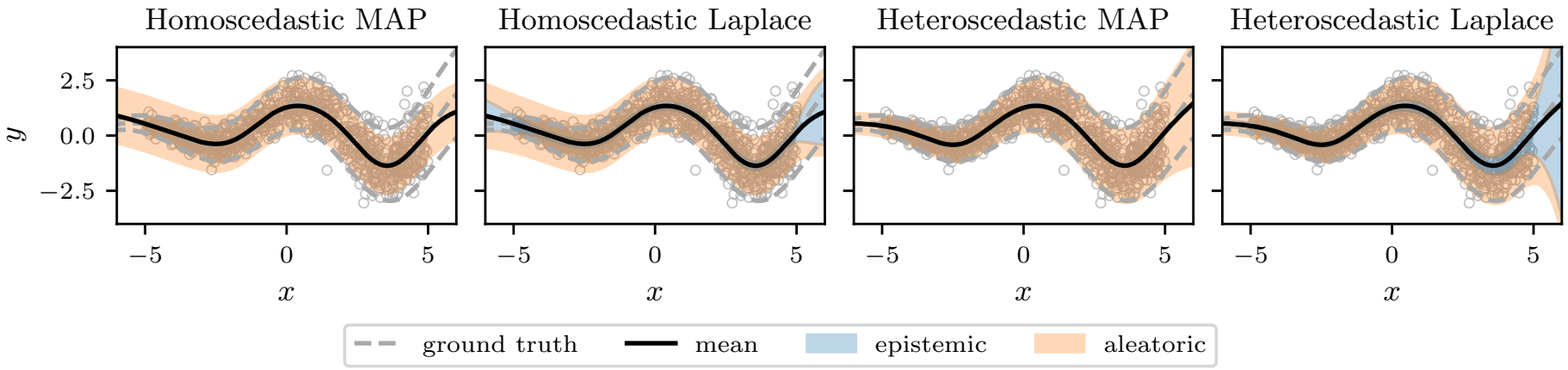
$$\eta_1 = \frac{\mu}{\sigma^2} \quad \text{and} \quad \eta_2 = -\frac{1}{2\sigma^2}$$

→ Hessian and Laplace approximations straightforward

→ Neural network models both natural parameters

[1]Le, Smola, Canu. *Heteroscedastic Gaussian process regression.* ICML 2005.
[2]Martens. *New insights and perspectives on the natural gradient method*. JMLR 2020.

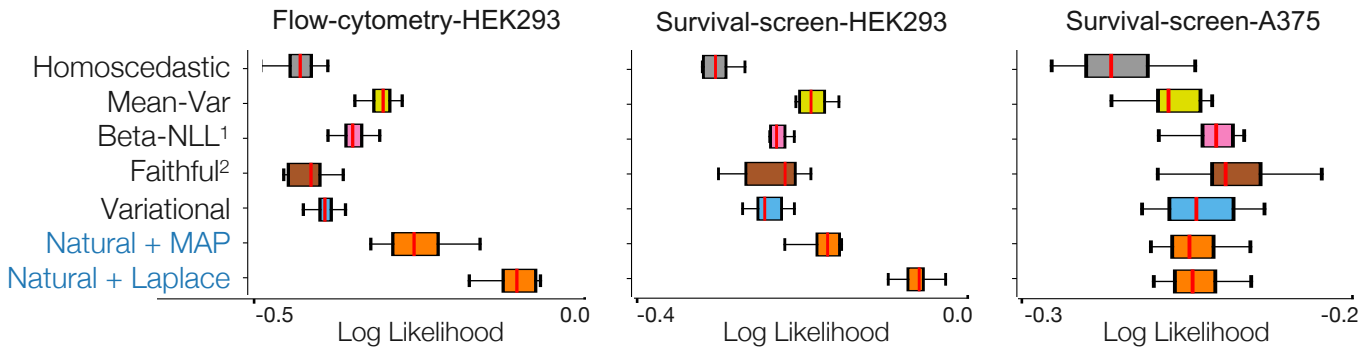
Heteroscedastic Regression Illustration



Immer, Palumbo, Marx, Vogt. *Effective Bayesian Heteroscedastic Regression with Deep Neural Networks.* NeurIPS 2023.

Epistemic Uncertainty in Heteroscedastic Regression

Problem: predict gene knockdown efficacy of guides in CRISPR system[2]



→ Improvements due to both natural parameterization and Laplace

[1]Seitzer, Tavakoli, Antic, Martius. *On the Pitfalls of Heteroscedastic Uncertainty Estimation with Probabilistic Neural Networks.* ICLR 2022.
[2]Stirn, Wessels, …, Knowles. *Faithful Heteroscedastic Regression with Neural Networks*. AISTATS 2023.