How accurate is the uncertainty estimate from your Bayesian neural networks?

Yingzhen Li

Microsoft Research Cambridge, UK

Why we need uncertainty estimate

Do we have big data?

- 1K datapoints of 10 dimensions vs 1K datapoints of 1K intrinsic dimensions
- 1K datapoints for an NN with 10K parameters vs 1B parameters

Do we have perfect model?

- training data distribution = test data distribution?
- Even so, can we get 100% accuracy with 100% confidence?
- error in labels/supervision signals?

Type of uncertainty

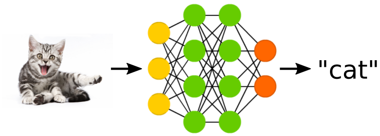
Imagine flipping a coin:

- Epistemic uncertainty: “How much do I believe the coin is fair?”
  - Population statistics
  - Reduces when having more data
- Aleatoric uncertainty: “What’s the next coin flip outcome?”
  - Individual experiment outcome
  - Non-reducible
- Distribution shift: “Am I still flipping the same coin?”

Bayesian neural networks 101

Let's say we want to classify different types of cats

- $\boldsymbol{x}$: input images; $\boldsymbol{y}$: output label
- build a neural network (with param. W):
  $p(\boldsymbol{y}|\boldsymbol{x}, W) = \text{softmax}(f_W(\boldsymbol{x}))$



A Bayesian solution:

Put a prior distribution $p(W)$ over W

- compute posterior $p(W|\mathcal{D})$ given a dataset $\mathcal{D} = \{(\boldsymbol{x}_n, \boldsymbol{y}_n)\}_{n=1}^N$:

$$p(W|\mathcal{D}) \propto p(W) \prod_{n=1}^{N} p(\boldsymbol{y}_n|\boldsymbol{x}_n, W)$$

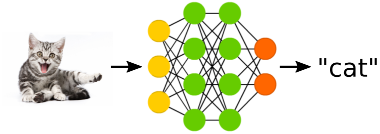
- Bayesian predictive inference:

$$p(\boldsymbol{y}^*|\boldsymbol{x}^*, \mathcal{D}) = \mathbb{E}_{p(W|\mathcal{D})}[p(\boldsymbol{y}^*|\boldsymbol{x}^*, W)]$$

Bayesian neural networks 101

Let's say we want to classify different types of cats

- $\boldsymbol{x}$: input images; $\boldsymbol{y}$: output label
- build a neural network (with param. W):
  $p(\boldsymbol{y}|\boldsymbol{x}, W) = \text{softmax}(f_W(\boldsymbol{x}))$



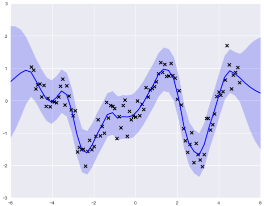
In practice: $p(W|\mathcal{D})$ is intractable

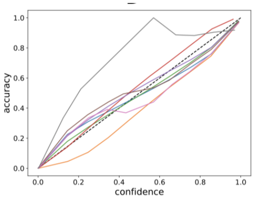
- First find approximation $q(W) \approx p(W|\mathcal{D})$ (e.g. via VI or MCMC)
- In prediction, do Monte Carlo sampling:

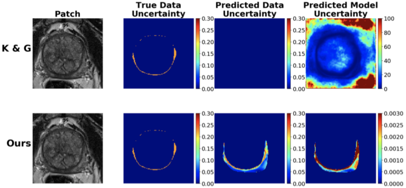
$$p(\boldsymbol{y}^*|\boldsymbol{x}^*, \mathcal{D}) \approx \frac{1}{K}\sum_{k=1}^{K} p(\boldsymbol{y}^*|\boldsymbol{x}^*, W^k), \quad W^k \sim q(W)$$

Empirical evaluations

“Model prediction with 70% confidence should be correct 70% of the time”







- Existing metrics (ECE, calibration improvement, etc.) for evaluating total uncertainty
- Aleatoric uncertainty evaluation needs multi expert labels
- Evaluating epistemic uncertainty is much harder
  - qualitatively: low near data, high far away

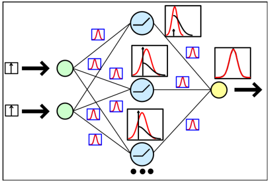
Jungo and Reyes MICCAI 2019, Hu et al. MICCAI 2019

When do we need epistemic uncertainty...

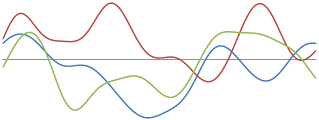
Tasks that require beliefs in acquired knowledge from data:

- Active learning/Bayesian optimisation
  - next datapoint to acquire for better model knowledge
- Reinforcement learning
  - exploration vs exploitation
- Continual learning
  - learning future tasks vs remembering previous tasks

Issues of weight-space inference



(a) weight space view



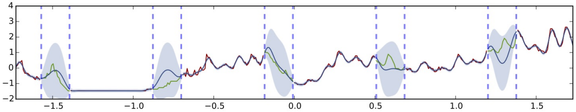
(b) function space view

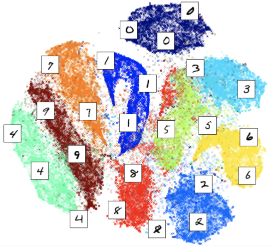
- Hard to specify prior (except for sparsity requirement)
- Symmetric modes in weight posterior
- Quality of uncertainty estimates in function space?
  - sample $W \sim q(W) \Leftrightarrow$ sample $f(\cdot) \sim q_{\text{BNN}}(f|\mathcal{D})$
  - $q(W)$ needs to be simple for computational efficiency
  - $\Rightarrow$ quality of $q_{\text{BNN}}(f|\mathcal{D})$ can be less satisfactory

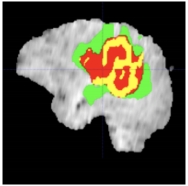
“In-between” uncertainty

“In-between” uncertainty:
uncertainty estimates in regions between data clusters

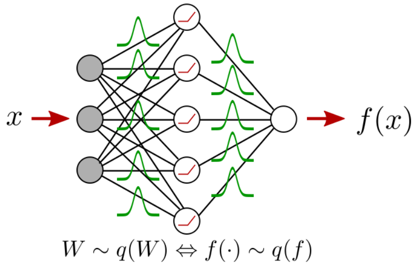
- Missing values (especially in time series)
- Ambiguous inputs

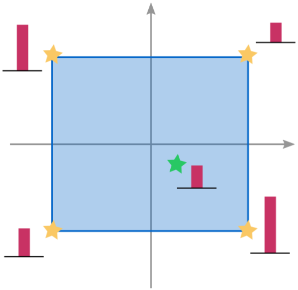






"In-between" uncertainty



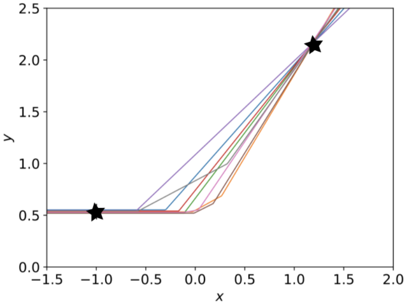


Theorem (mean-field Gaussian, epistemic)

For a one-hidden layer BNN with ReLU activation, any Gaussian mean-field distribution on weights $q(W) = \prod_{ij} \mathcal{N}(W_{ij}; \mu_{ij}, \sigma_{ij}^2)$, and any hyper-cube C that contains $\mathbf{0}$:

The value of the variance function $\mathbb{V}[f(\boldsymbol{x})]$ at any $\boldsymbol{x} \in C$ is bounded by the variance function values at the vertices of C.

"In-between" uncertainty

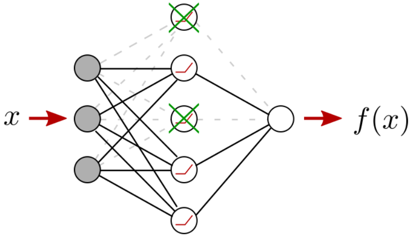


Intuition behind the theory:

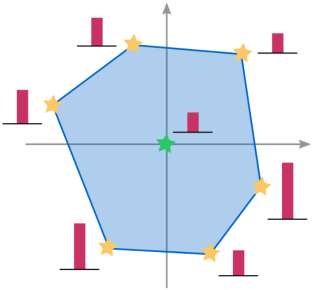
- To fit the data, σ_{ij} of $q(W_{ij})$ needs to be relatively small
- For $\text{ReLU}(wx+b)$, w controls slope, b controls intercept
- "In-between" epistemic uncertainty requires correlations in W

Foong et al. NeurIPS 2019 Bayesian deep learning workshop

"In-between" uncertainty



$h(x) \sim q(h) \Leftrightarrow f(\cdot) \sim q(f)$

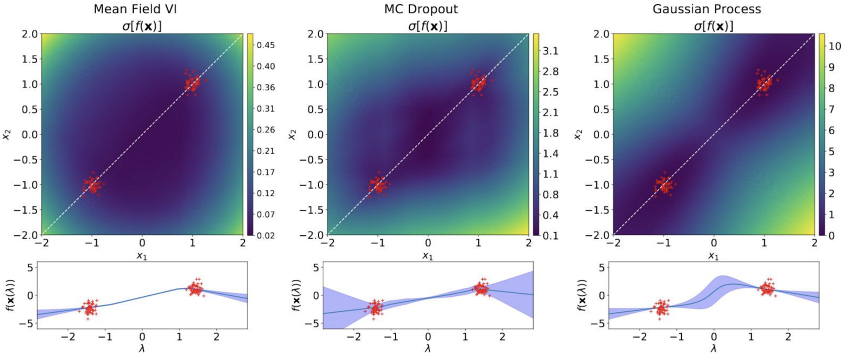


Theorem (MC-dropout for hidden units, epistemic)

For a one-hidden layer BNN with ReLU activation, any dropout rate, and any set of input points S where its convex hull contains $\mathbf{0}$:

The value of the variance function $\mathbb{V}[f(\mathbf{0})]$ is bounded by the variance function values at the points in S.

"In-between" uncertainty

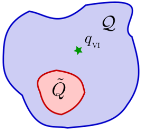


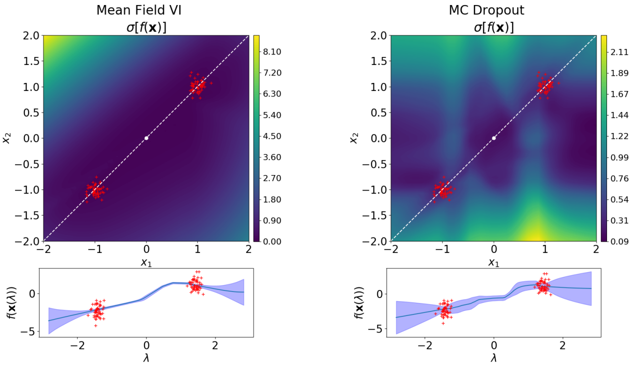
Foong et al. NeurIPS 2019 Bayesian deep learning workshop

“In-between” uncertainty

“Should I worry about this result when I’m using deeper BNNs?”

- Two-layer cases: $\exists$ mean-field Gaussian $\tilde{q}(W)$ s.t. (epistemic) variance function shows good “in-between” uncertainty
- Can BNN training methods find it?



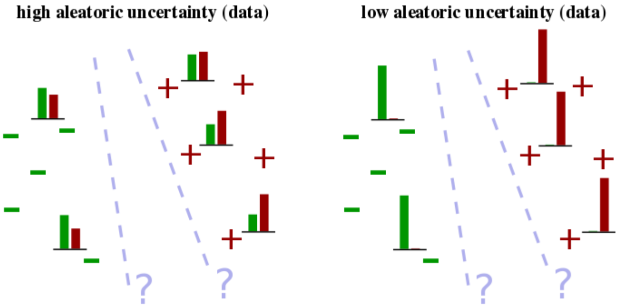


Foong et al. NeurIPS 2019 Bayesian deep learning workshop

"In-between" uncertainty

"Should I worry about this result when I'm using deeper BNNs?"

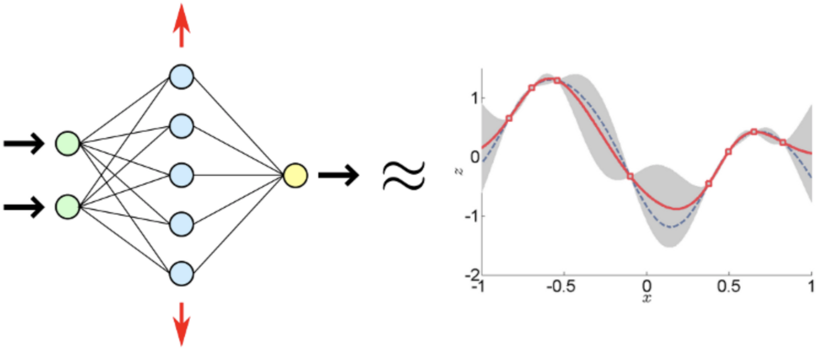
- Aleatoric uncertainty can still be high:
  e.g. $q(W) \approx \delta(W_0)$ and $\mathrm{softmax}(f_{W_0}(\boldsymbol{x}))$ is flat
- Classification/segmentation tasks require heteroskedastic aleatoric uncertainty
  $\Rightarrow$ need more datapoints and/or multi expert labels for good estimation
- Epistemic uncertainty in decision boundary still needed



Function space inference

Radford Neal's derivation:

- BNN with mean-field prior $\rightarrow$ Gaussian process (GP) prior

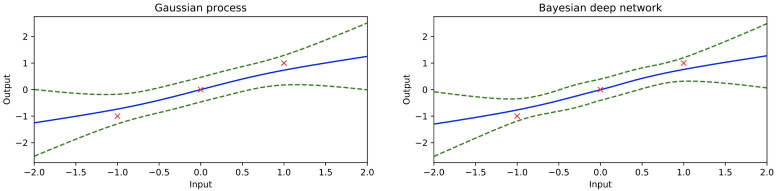


GPSS 2019 BNN tutorial, `http://gpss.cc/gpss19/program`

Function space inference

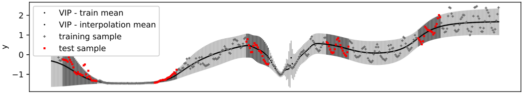
Recent extensions of Radford Neal's result:

- deep and wide BNNs with mean-field prior $\rightarrow$ GP prior
- Neural Tangent Kernel (NTK): for very wide NNs
  - NN regression $\approx$ kernel regression, in gradient descent dynamics
  - Laplace/variational Gaussian BNNs $\approx$ GP posterior with NTK

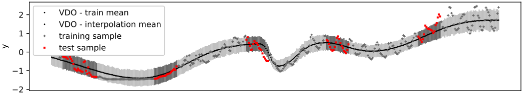


Matthews et al. 2018, Lee et al. 2018, Garriga-Alonso et al. 2019, Novak et al. 2019, Jacot et al. 2018, Khan et al. 2019

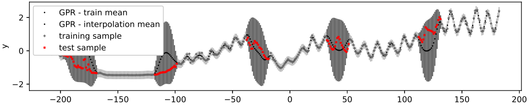
Function space inference



(c) VIP-BNN



(d) Variational dropout (VDO-BNN)



(e) GP regression (GPR)

Variational implicit processes:

- prior over NN weights $p(W)$
  $\Leftrightarrow$ prior over functions $p_{\text{BNN}}(f)$
- $p_{\text{BNN}}(f)$ implicitly defined (intractable, unlike GPs)
- posterior approximation: $q_{\text{GP}}(f|\mathcal{D}) \approx p_{\text{BNN}}(f|\mathcal{D})$
- Empirical Bayes: optimise $p(W)$

Ma et al. ICML 2019

What we have covered today...

Using Bayesian methods for deep learning:

- Need to compute calibration metrics
- Be careful when choosing weight-space inference method
- Think more about uncertainty estimation in function space

Thank you!

References

Neal 1994. Bayesian Learning for Neural Networks. PhD thesis

Matthews et al. 2018. Gaussian Process Behaviour in Wide Deep Neural Networks. ICLR 2018

Lee et al. 2018. Deep Neural Networks as Gaussian Processes. ICLR 2018

Ma et al. 2019. Variational Implicit Processes. ICML 2019

Foong et al. 2019. Pathologies of Factorised Gaussian and MC Dropout Posteriors in Bayesian Neural Networks. arXiv:1909.00719

Garriga-Alonso et al. 2019. Deep Convolutional Networks as shallow Gaussian Processes. ICLR 2019.

Novak et al. 2019. Bayesian Deep Convolutional Networks with Many Channels are Gaussian Processes. ICLR 2019.

Jacot et al. 2019. Neural tangent kernel: Convergence and generalization in neural networks. NeurIPS 2018.

Khan et al. 2019. Approximate Inference Turns Deep Networks into Gaussian Processes. NeurIPS 2019

Hu et al. 2019. Supervised Uncertainty Quantification for Segmentation with Multiple Annotations. MICCAI 2019.

Jungo and Reyes 2019. Assessing Reliability and Challenges of Uncertainty Estimations for Medical Image Segmentation. MICCAI 2019.

`http://mlg.eng.cam.ac.uk/yarin/blog_images/Solar_GP_SE.jpg`

`https://bigsnarf.wordpress.com/2016/11/17/t-sne-attack-data/`