Bayesian Deep Learning

Maurizio Filippone

EURECOM, Sophia Antipolis, France

January 10th, 2019

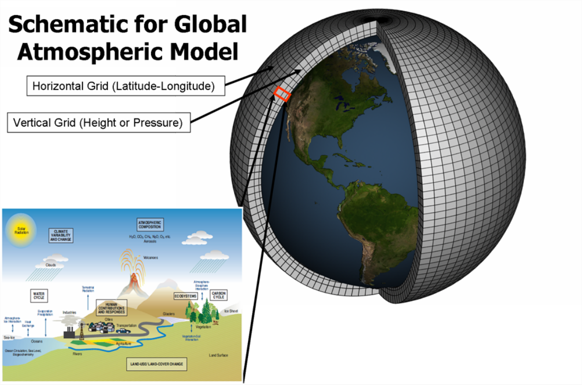
Outline

1. Motivation
2. Probabilistic Deep Nets
   - Scalable Inference
   - Connections with (Deep) Gaussian Processes
3. Some Results
4. Conclusions

Motivation

Quantification of Uncertainty with Expensive Models

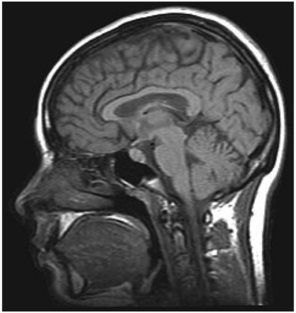
- Climate modeling

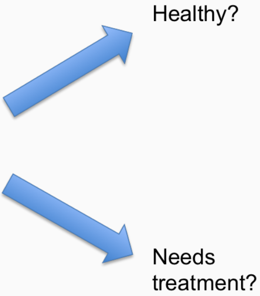


Kennedy and O'Hagan, *J-RSS-B*, 2001

Quantification of Uncertainty with No Models

- Classification and progression of neurodegenerative diseases





Filippone et al., *AoAS*, 2012 – Lorenzi, Filippone et al., *NeuroImage*, 2017 – Lorenzi and Filippone, *ICML*, 2018

A Unified Framework

A model might be expensive to simulate/inaccurate

- Emulate model/discrepancy using a surrogate

A Unified Framework

A model might be expensive to simulate/inaccurate

- Emulate model/discrepancy using a surrogate

A model might not even be available

- Replace it with a flexible statistical model

A Unified Framework

A model might be expensive to simulate/inaccurate

- Emulate model/discrepancy using a surrogate

A model might not even be available

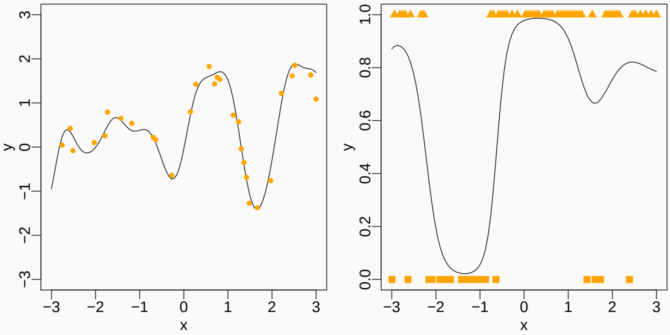
- Replace it with a flexible statistical model

Probabilistic Deep Models for Accurate Modeling and Quantification of Uncertainty

Probabilistic Deep Nets

Learning from Data – Function Estimation

- Take these two examples



- We are interested in estimating a function $\mathbf{f}(\mathbf{x})$ from data
- Most problems in Machine Learning can be cast this way!

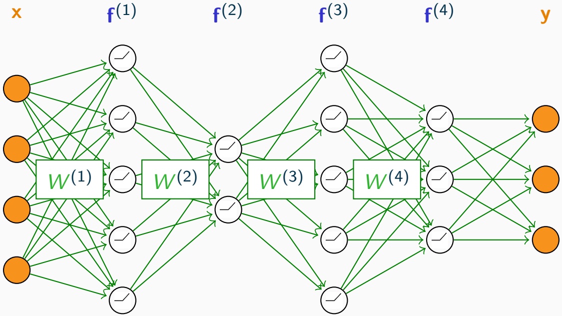
Deep Neural Networks

- Implement a composition of parametric functions

$$\mathbf{f}(\mathbf{x}) = \mathbf{f}^{(L)}\left(\mathbf{f}^{(L-1)}\left(\cdots \mathbf{f}^{(1)}(\mathbf{x})\cdots\right)\right)$$

with

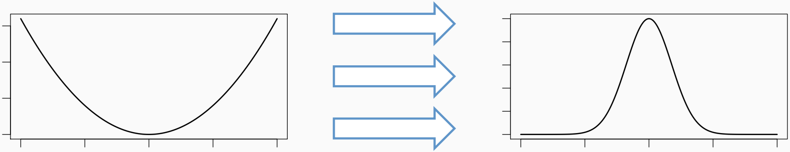
$$\mathbf{f}^{(l)}(\mathbf{h}) = \mathbf{g}\left(\mathbf{h}^\top W^{(l)}\right)$$



Back-propagation – Probabilistic Interpretation Loss

- Inputs : $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$
- Labels : $Y = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$
- Weights : $W = \{W^{(1)}, \ldots, W^{(L)}\}$

Quadratic Loss $\qquad$ $p(Y|X, W) \propto \exp(-\mathrm{Loss})$

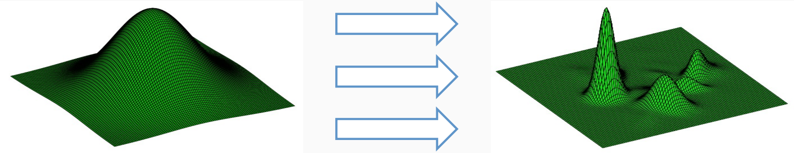


- Back-propagation minimizes a loss function
- … equivalent as optimizing likelihood $p(Y|X, W)$

Bayesian Inference

- Inputs : $X = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$
- Labels : $Y = \{\mathbf{y}_1, \dots, \mathbf{y}_N\}$
- Weights : $W = \{W^{(1)}, \dots, W^{(L)}\}$

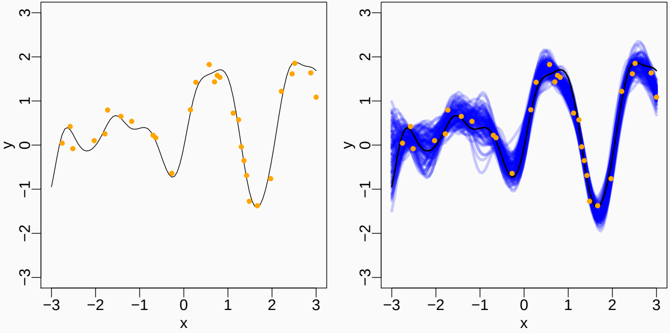
$p(W)$ $\qquad\qquad\qquad$ $p(W|Y,X)$



$$p(W|Y,X) = \frac{p(Y|X,W)p(W)}{\int p(Y|X,W)p(W)dW}$$

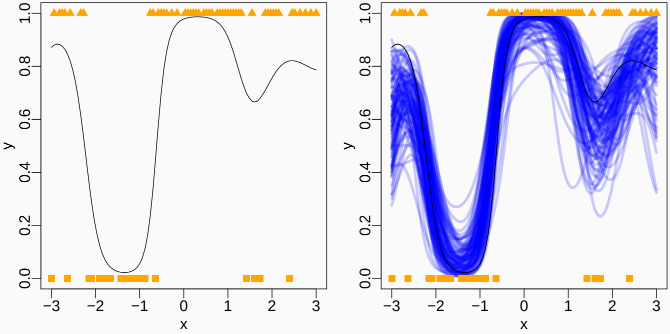
Bayesian Deep Neural Networks

- Regression example



Bayesian Deep Neural Networks

- Classification example



Stochastic Variational Inference

- Bayesian inference is intractable due to this integral

$$\log\left[p(Y|X)\right] = \log\left[\int p(Y|X, W)p(W)dW\right]$$

Stochastic Variational Inference

- Bayesian inference is intractable due to this integral

$$\log\left[p(Y|X)\right] = \log\left[\int p(Y|X,W)p(W)dW\right]$$

- Lower bound for $\log\left[p(Y|X)\right]$

$$\mathrm{E}_{q(W)}\left(\log\left[p\left(Y|X,W\right)\right]\right) - \mathrm{KL}\left[q(W)\|p\left(W\right)\right],$$

where $q(W)$ approximates $p(W|Y,X)$.

- Kullback-Leibler divergence KL – "distance" between q and p

Stochastic Variational Inference

- Bayesian inference is intractable due to this integral

$$\log\left[p(Y|X)\right] = \log\left[\int p(Y|X, W)p(W)dW\right]$$

- Lower bound for $\log\left[p(Y|X)\right]$

$$\mathrm{E}_{q(W)}\left(\log\left[p\left(Y|X, W\right)\right]\right) - \mathrm{KL}\left[q(W)\|p\left(W\right)\right],$$

where $q(W)$ approximates $p(W|Y, X)$.

- Kullback-Leibler divergence KL – "distance" between q and p

Optimize the lower bound wrt the parameters of $q(W)$

Graves, *NIPS*, 2011

Stochastic Variational Inference

- Assume that the likelihood factorizes

$$p(Y|X, W) = \prod_k p(\mathbf{y}_k|\mathbf{x}_k, W)$$

Stochastic Variational Inference

- Assume that the likelihood factorizes

$$p(Y|X, W) = \prod_k p(\mathbf{y}_k|\mathbf{x}_k, W)$$

- Doubly stochastic **unbiased** estimate of the expectation term
  - Mini-batch

$$\mathrm{E}_{q(W)}\left(\log\left[p\left(Y|X, W\right)\right]\right) \approx \frac{n}{m}\sum_{k\in\mathcal{I}_m}\mathrm{E}_{q(W)}\left(\log\left[p(\mathbf{y}_k|\mathbf{x}_k, W)\right]\right)$$

  - Monte Carlo

$$\mathrm{E}_{q(W)}\left(\log\left[p(\mathbf{y}_k|\mathbf{x}_k, W)\right]\right) \approx \frac{1}{N_{\mathrm{MC}}}\sum_{r=1}^{N_{\mathrm{MC}}}\log[p(\mathbf{y}_k|\mathbf{x}_k, \tilde{W}_r)]$$

with $\tilde{W}_r \sim q(W)$.

Stochastic Variational Inference

- Assume a factorized Gaussian approximate posterior:

$$q(W) = \prod_{ijl} q\left(W_{ij}^{(l)}\right) = \prod_{ijl} \mathcal{N}\left(\mu_{ij}^{(l)}, (\sigma^2)_{ij}^{(l)}\right) \tag{1}$$

Stochastic Variational Inference

- Assume a factorized Gaussian approximate posterior:

$$q(W) = \prod_{ijl} q\left(W_{ij}^{(l)}\right) = \prod_{ijl} \mathcal{N}\left(\mu_{ij}^{(l)}, (\sigma^2)_{ij}^{(l)}\right) \tag{1}$$

- Reparameterization trick

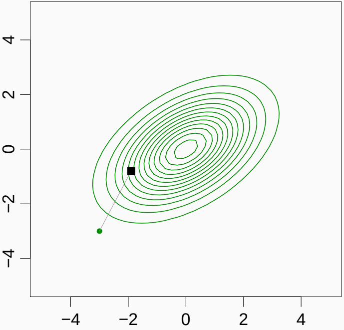
$$(\tilde{W}_r^{(l)})_{ij} = \sigma_{ij}^{(l)} \varepsilon_{rij}^{(l)} + \mu_{ij}^{(l)},$$

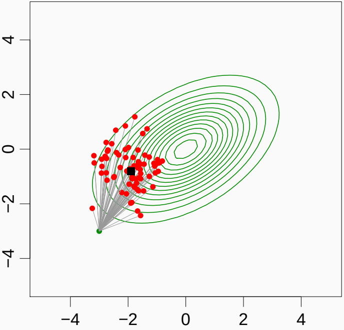
  with $\varepsilon_{rij}^{(l)} \sim \mathcal{N}(0, 1)$
- Optimization wrt $\mu_{ij}^{(l)}, (\sigma^2)_{ij}^{(l)}$ with automatic differentiation

Kingma and Welling, *ICLR*, 2014

Stochastic Gradient Optimization

$$\mathrm{E}\left\{\widetilde{\nabla_{\mathrm{par_q}}}\mathrm{LowerBound}\right\} = \nabla_{\mathrm{par_q}}\mathrm{LowerBound}$$

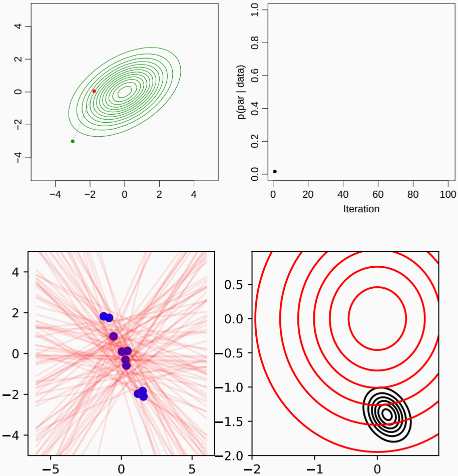




Robbins and Monro, *AoMS*, 1951

Stochastic Variational Inference - Simple Illustration

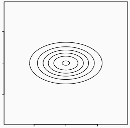
$$\mathrm{par_q}' = \mathrm{par_q} + \frac{\alpha_t}{2}\widetilde{\nabla_{\mathrm{par_q}}}(\mathrm{LowerBound}) \qquad \alpha_t \to 0$$



Form of Approximating Distribution

Approximating distribution $q(W)$ can have the following forms:

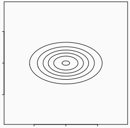
- Fully factorized



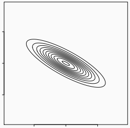
Form of Approximating Distribution

Approximating distribution $q(W)$ can have the following forms:

- Fully factorized



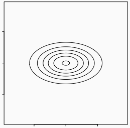
- Full covariance $\Sigma = LL^\top$



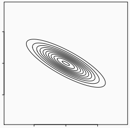
Form of Approximating Distribution

Approximating distribution $q(W)$ can have the following forms:

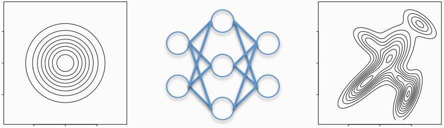
- Fully factorized



- Full covariance $\Sigma = LL^\top$



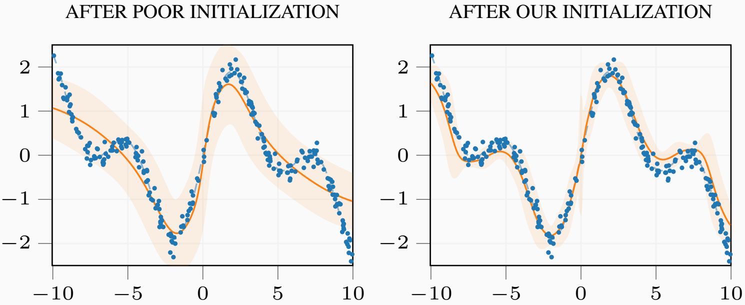
- Normalizing Flows, Real NVPs, Stein VI - change of measure determined by $\det(\text{Jacobian})$



Rezende et al., *ICML*, 2015 – Dinh et al., *ICLR*, 2017 – Liu and Wang, *NIPS*, 2016

Initialization of SVI matters

- Initialization can be an issue
- We proposed a novel way to initialize SVI well

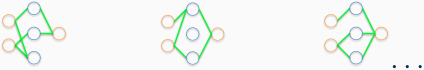


Rossi, Michiardi and Filippone, *arXiv*, 2018

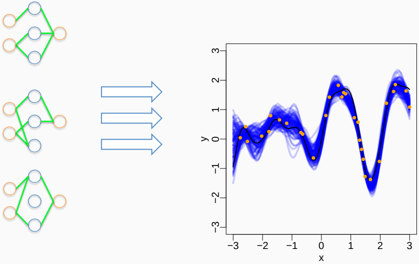
Is There an Easier Way?

- Dropout is Variational Inference with Bernoulli-like $q(W)$
- At training time, apply dropout

Iteration 1 Iteration 2 Iteration 3 . . .



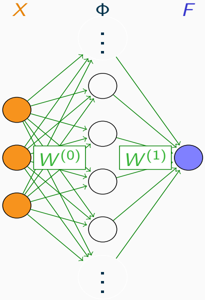
- At test time, "sample" networks with different dropout masks



Gal and Ghahramani, *ICML*, 2016

Gaussian Processes as Infinitely-Wide Shallow Neural Nets

- Take $W^{(i)} \sim \mathcal{N}(\mathbf{0}, \alpha_i I)$
- Central Limit Theorem implies that F is Gaussian

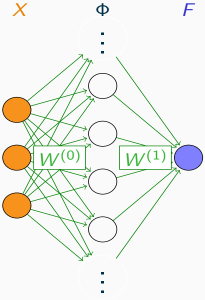


- F has zero-mean
- $\mathrm{cov}(F) = \mathrm{E}_{p(W^{(0)},W^{(1)})}[\Phi(XW^{(0)})W^{(1)}W^{(1)\top}\Phi(XW^{(0)})^\top]$

Neal, *LNS*, 1996 – Rasmussen and Williams, 2006

Gaussian Processes as Infinitely-Wide Shallow Neural Nets

- Take $W^{(i)} \sim \mathcal{N}(\mathbf{0}, \alpha_i I)$
- Central Limit Theorem implies that F is Gaussian



- F has zero-mean
- $\mathrm{cov}(F) = \alpha_1 \mathrm{E}_{p(W^{(0)})}[\Phi(XW^{(0)})\Phi(XW^{(0)})^\top]$
- Some choices of Φ lead to analytic expression of known kernels (RBF, Matérn, arc-cosine, Brownian motion, ...)

Neal, *LNS*, 1996 – Rasmussen and Williams, 2006

Random Feature Expansions for DGPs - Bochner's theorem

- Continuous shift-invariant covariance function

$$k(\mathbf{x}_i - \mathbf{x}_j|\boldsymbol{\theta}) = \sigma^2 \int p(\boldsymbol{\omega}|\boldsymbol{\theta}) \exp\left(\iota(\mathbf{x}_i - \mathbf{x}_j)^\top \boldsymbol{\omega}\right) d\boldsymbol{\omega}$$

Random Feature Expansions for DGPs - Bochner's theorem

- Continuous shift-invariant covariance function

$$k(\mathbf{x}_i - \mathbf{x}_j | \boldsymbol{\theta}) = \sigma^2 \int p(\boldsymbol{\omega} | \boldsymbol{\theta}) \exp\left(\iota (\mathbf{x}_i - \mathbf{x}_j)^\top \boldsymbol{\omega}\right) d\boldsymbol{\omega}$$

- Monte Carlo estimate

$$k(\mathbf{x}_i - \mathbf{x}_j | \boldsymbol{\theta}) \approx \frac{\sigma^2}{N_{\mathrm{RF}}} \sum_{r=1}^{N_{\mathrm{RF}}} \mathbf{z}(\mathbf{x}_i | \tilde{\boldsymbol{\omega}}_r)^\top \mathbf{z}(\mathbf{x}_j | \tilde{\boldsymbol{\omega}}_r)$$

with

$$\tilde{\boldsymbol{\omega}}_r \sim p(\boldsymbol{\omega} | \boldsymbol{\theta})$$

$$\mathbf{z}(\mathbf{x} | \boldsymbol{\omega}) = [\cos(\mathbf{x}^\top \boldsymbol{\omega}), \sin(\mathbf{x}^\top \boldsymbol{\omega})]^\top$$

Rahimi and Recht, *NIPS*, 2008 - Lázaro-Gredilla et al., *JMLR*, 2010

Random Feature Expansions for DGPs

- Define

$$\Phi^{(l)} = \sqrt{\frac{\sigma^2}{N_{\mathrm{RF}}^{(l)}}} \left[\cos\left(F^{(l)}\Omega^{(l)}\right), \sin\left(F^{(l)}\Omega^{(l)}\right)\right]$$

and

$$F^{(l+1)} = \Phi^{(l)} W^{(l)}$$

- We are stacking Bayesian linear models with

$$p\left(W_{\cdot i}^{(l)}\right) = \mathcal{N}\left(\mathbf{0}, I\right)$$

Random Feature Expansions for DGPs

- Define

$$\Phi^{(l)} = \sqrt{\frac{\sigma^2}{N_{\mathrm{RF}}^{(l)}}} \left[\cos\left(F^{(l)}\Omega^{(l)}\right), \sin\left(F^{(l)}\Omega^{(l)}\right)\right]$$

  and

$$F^{(l+1)} = \Phi^{(l)} W^{(l)}$$

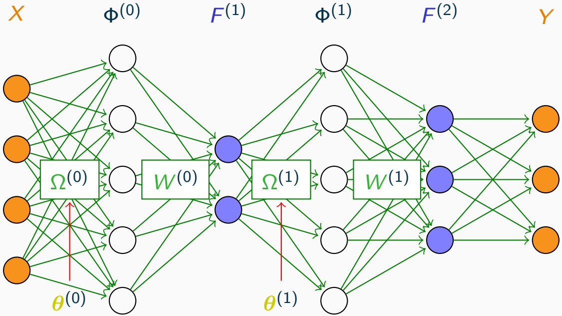
- We are stacking Bayesian linear models with

$$p\left(W_{\cdot i}^{(l)}\right) = \mathcal{N}\left(\mathbf{0}, I\right)$$

- Expansion of arc-cosine kernel yields ReLU activations!

Cutajar, Bonilla, Michiardi, Filippone, *ICML*, 2017

Random Feature Expansions make Deep GPs become DNNs

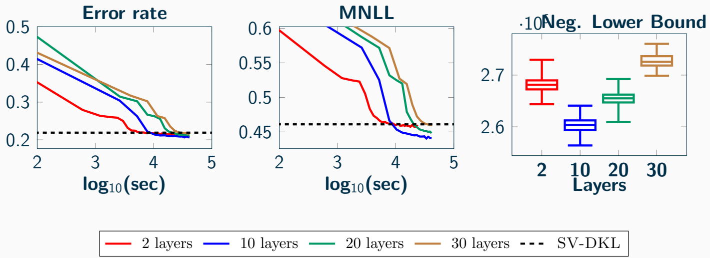


Cutajar, Bonilla, Michiardi, Filippone, *ICML*, 2017

Some Results

Airline dataset

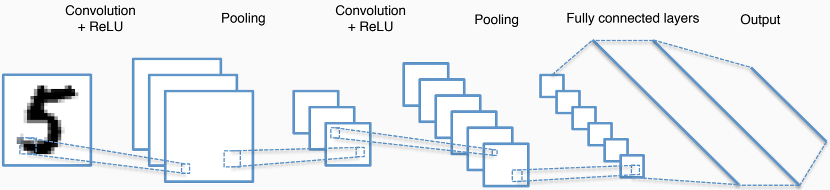
$$(n = 5\mathrm{M}+,\ d = 8)$$



Cutajar, Bonilla, Michiardi, Filippone, *ICML*, 2017

Convolutional Nets

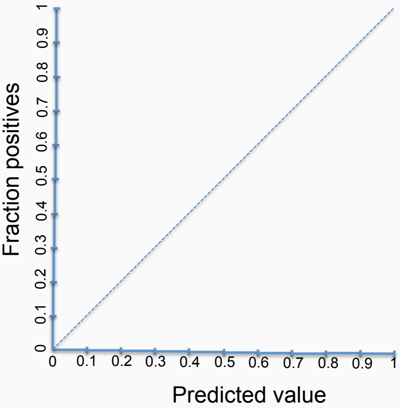
- Convolutional nets are widely used...
- ...but they are known to be overconfident!



Guo et al., *ICML*, 2017

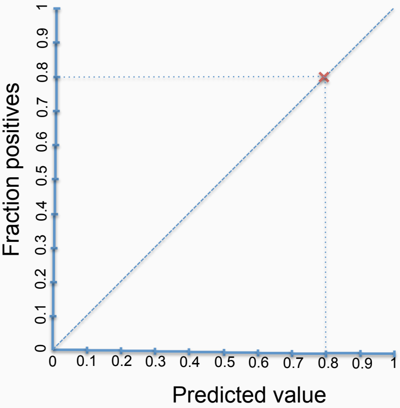
Calibration as a Measure of Quantification of Uncertainty

- Reliability diagrams



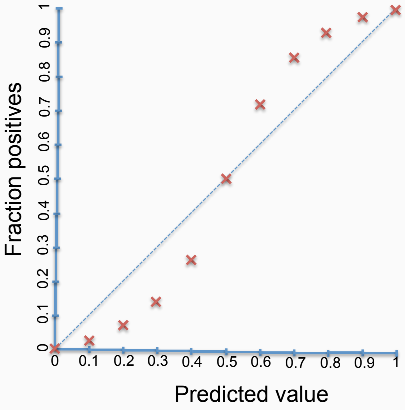
Calibration as a Measure of Quantification of Uncertainty

- Reliability diagrams



Calibration as a Measure of Quantification of Uncertainty

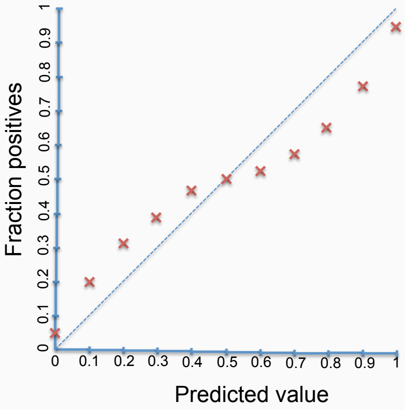
- Reliability diagrams - Under-confident predictions



- We can extract the Expected Calibration Error (ECE) score
- The BRIER score is another measure of calibration

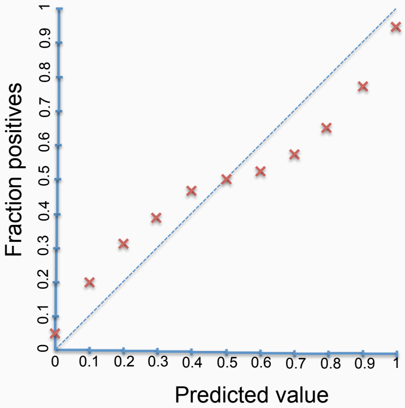
Calibration as a Measure of Quantification of Uncertainty

- Reliability diagrams - Overconfident predictions



Calibration as a Measure of Quantification of Uncertainty

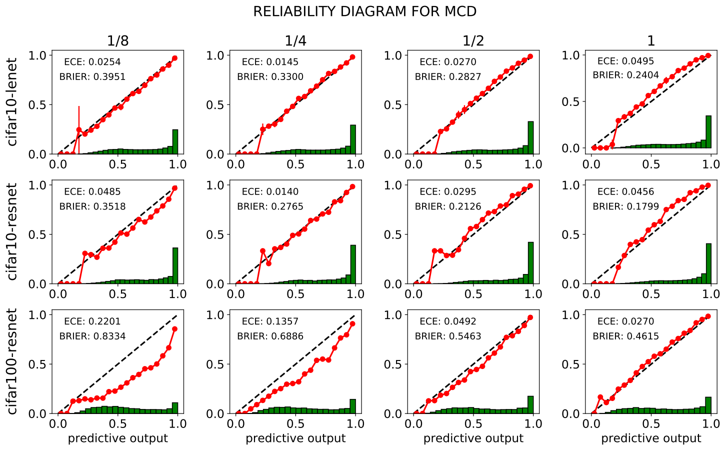
- Reliability diagrams - Overconfident predictions



Reliability diagrams of modern Deep CNNs look like this!
Bayesian treatment of filters fixes it!

Bayesian CNNs are calibrated

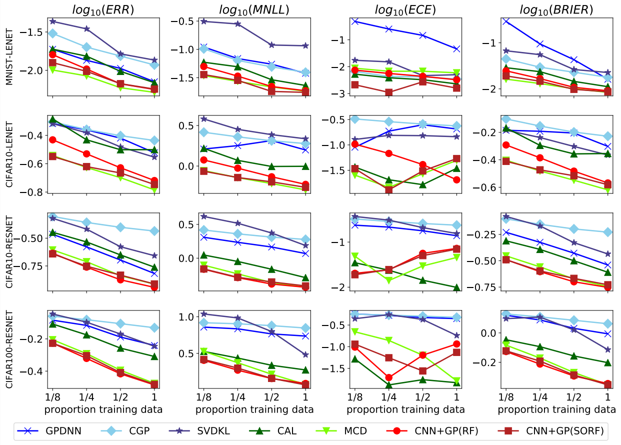
- Inferring parameters of convolutional filter recovers calibration
- Example with Monte Carlo Dropout



Tran et al., *AISTATS*, 2019

Performance Evaluation of Bayesian CNNs

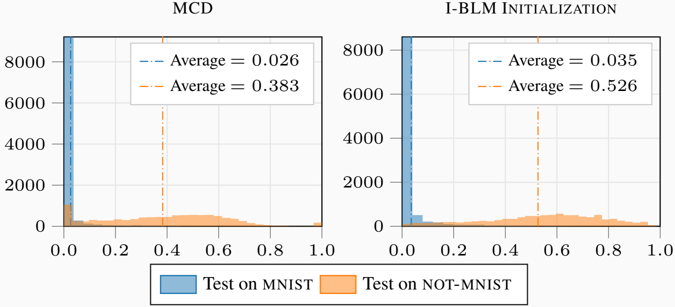
- Bayesian CNNs are calibrated and achieve better performance than post calibrated CNNs



Tran et al., *AISTATS*, 2019

Knowing When the Model Doesn't Know

- Training on MNIST and test on not-MNIST



Conclusions

Conclusions

- Inference for Deep Nets is hard
  - Scalable stochastic-based approximate inference but...
  - ... it is difficult to assess the impact approximations on quantification of uncertainty

Conclusions

- Inference for Deep Nets is hard
  - Scalable stochastic-based approximate inference but...
  - ... it is difficult to assess the impact approximations on quantification of uncertainty
- The connection between Deep Nets and Deep Gaussian processes can have implications on
  - Understanding Deep Learning
  - Deriving sensible priors for Deep Learning
  - Improving inference borrowing algebraic/computational tricks from kernel literature

Conclusions

- Inference for Deep Nets is hard
  - Scalable stochastic-based approximate inference but...
  - ... it is difficult to assess the impact approximations on quantification of uncertainty
- The connection between Deep Nets and Deep Gaussian processes can have implications on
  - Understanding Deep Learning
  - Deriving sensible priors for Deep Learning
  - Improving inference borrowing algebraic/computational tricks from kernel literature
- Cool stuff
  - New hardware
  - Bayesian compression

We are hiring PhDs, Post-docs and Assistant Professors

Acknowledgments

Thank you!

AXA Research Fund