# Learning Machines Seminars

2020-11-05

*Uncertainty in deep learning*

**Olof Mogren, PhD**

*RISE Research Institutes of Sweden*

Our world is full of uncertainties: measurement errors, modeling errors, or uncertainty due to test-data being out-of-distribution are some examples. Machine learning systems are increasingly being used in crucial applications such as medical decision making and autonomous vehicle control: in these applications, mistakes due to uncertainties can be life threatening.

Deep learning have demonstrated astonishing results for many different tasks. But in general, predictions are deterministic and give only a point estimate as output. A trained model may seem confident in predictions where the uncertainty is high. To cope with uncertainties, and make decisions that are reasonable and safe under realistic circumstances, AI systems need to be developed with uncertainty strategies in mind. Machine learning approaches with uncertainty estimates can enable active learning: an acquisition function can be based on model uncertainty to guide in data collection and tagging. It can also be used to improve sample efficiency for reinforcement learning approaches.

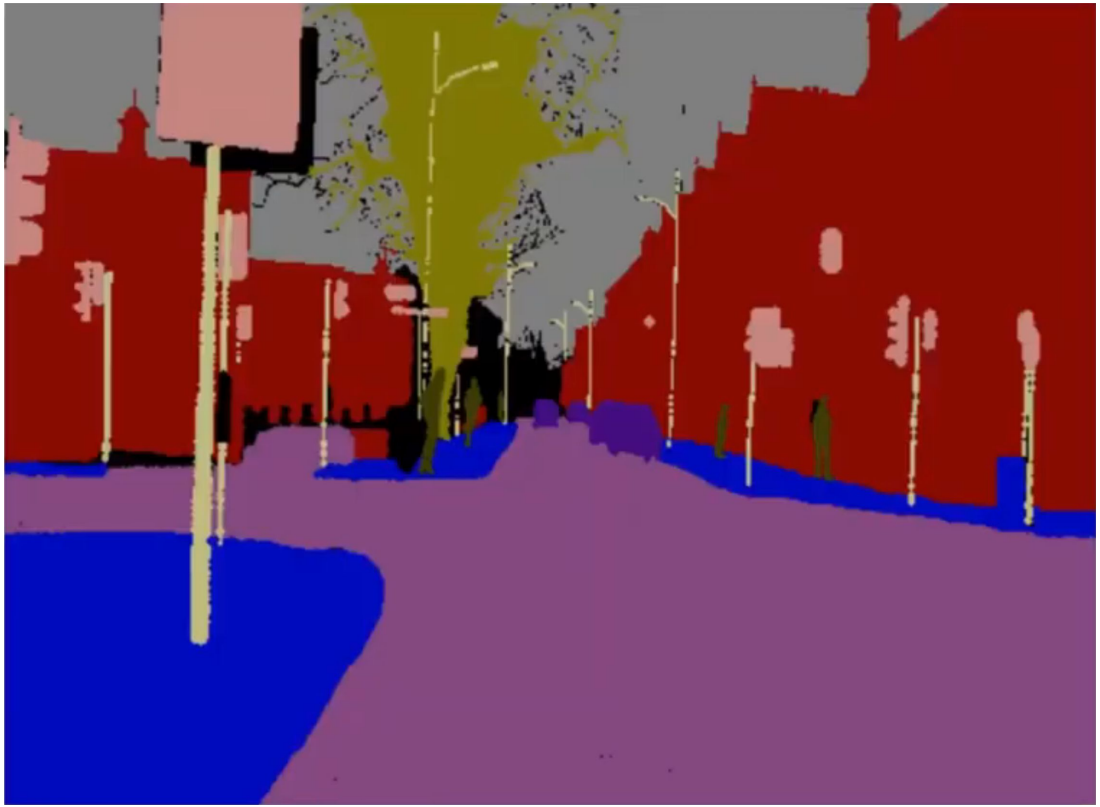
In this talk, we will connect deep learning with Bayesian machine learning, and go through some example approaches to coping with, and leveraging, the uncertainty in data and in modelling, to produce better AI systems in real world scenarios.

RI.
SE

# Automated driving

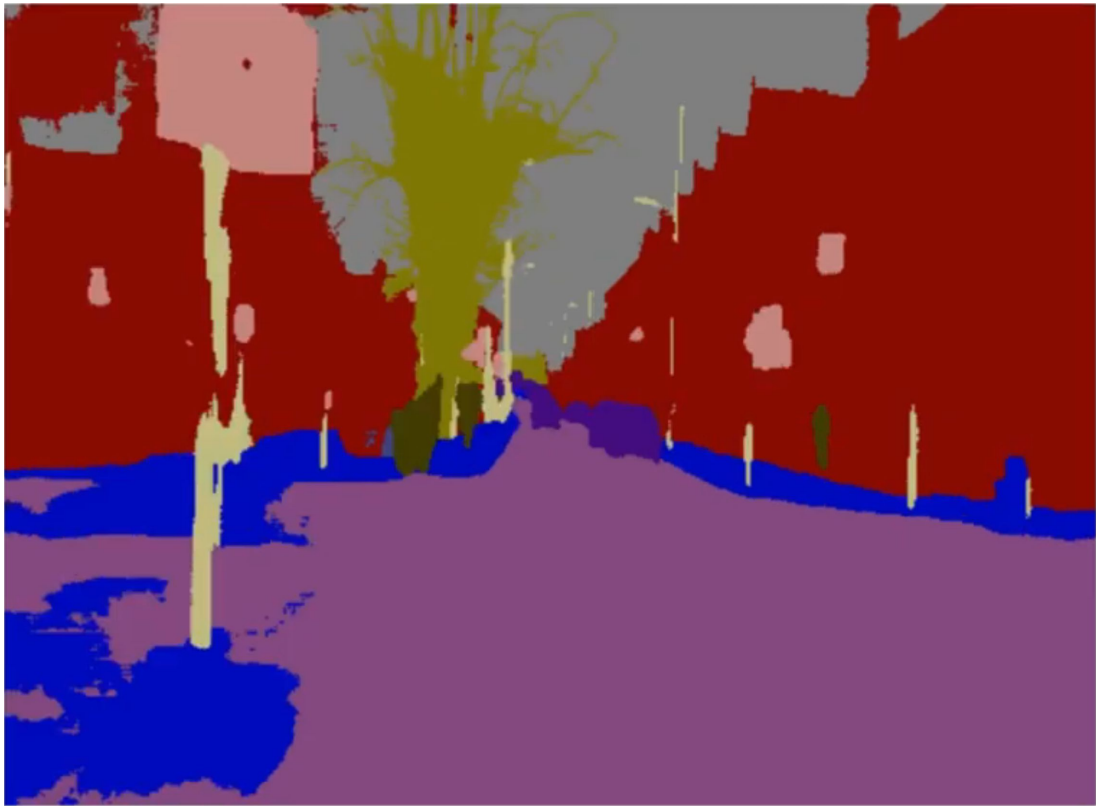
*Kendall, A., & Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision?. In Advances in neural information processing systems (pp. 5574-5584).*

# Automated driving



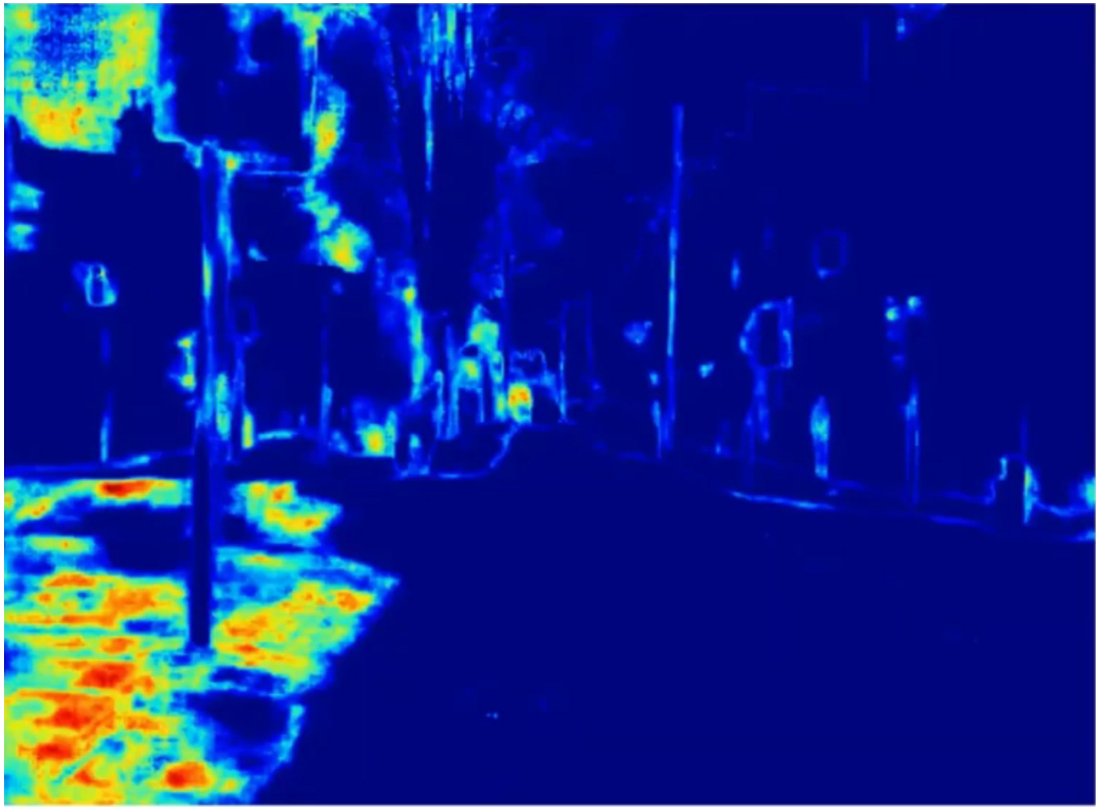
*Kendall, A., & Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision?. In Advances in neural information processing systems (pp. 5574-5584).*

# Automated driving



*Kendall, A., & Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision?. In Advances in neural information processing systems (pp. 5574-5584).*

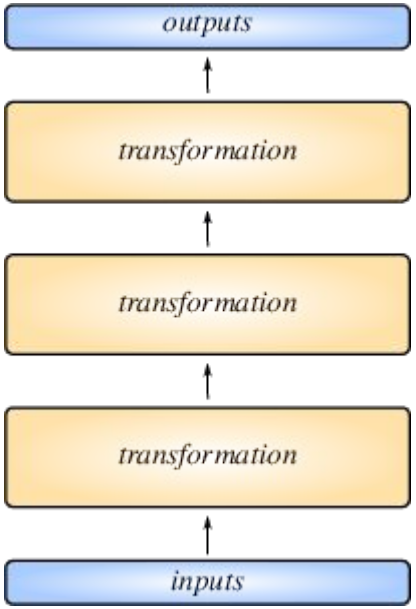
# Automated driving



*Kendall, A., & Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision?. In Advances in neural information processing systems (pp. 5574-5584).*

# Deep learning

- Nested transformations
- **h**(**x**) = **a**(**x**W+**b**)
- End to end training: backpropagation, optimization
- **a**: activation functions
  - Logistic, tanh, relu
  - Classification: Softmax output
- Softmax outputs: cross-entropy loss
  - Probabilistic interpretation




RI.
SE

# Out of distribution data

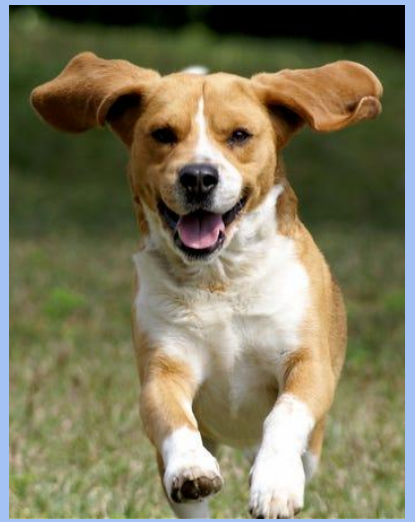
- Train: cats vs dogs
- At test time appears

Training data:

# Out of distribution data

- Train: cats vs dogs
- At test time appears a bird image
- What to do?

Training data:



Testing data:

# Out of distribution data

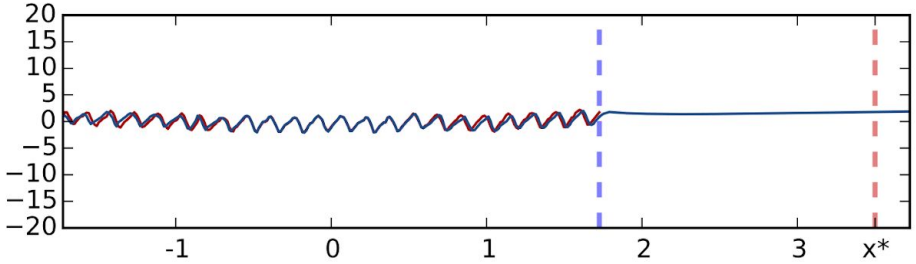
- Train: cats vs dogs
- At test time appears a bird image
- What to do?
- What will the softmax do

?

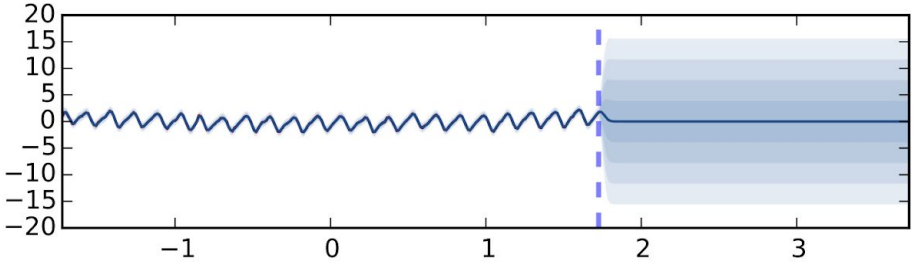
Training data:

Testing data:

# Out of domain data (ctd)




(a) Standard deep learning model




(b) Probabilistic model

Mauna Loa CO 2 concentrations dataset

*Image By Yarin Gal.*

RI. SE

# Uncertainty

- Aleatoric
  - Noise inherent in data observations
  - Uncertainty in data or sensor errors
  - Will not decrease with larger data
  - Irreducible error/Bayes error
- Epistemic
  - Caused by the model
    - Parameters
    - Structure
  - Lack of knowledge of generating distribution
  - Reduced with increasing data

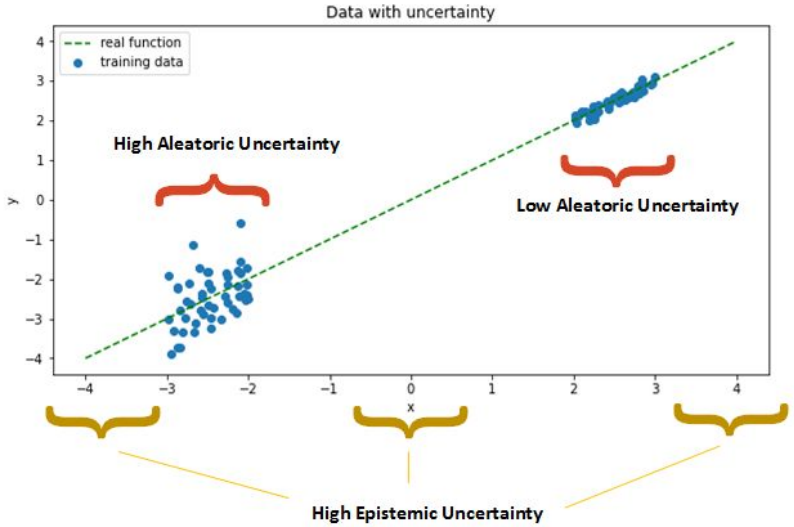
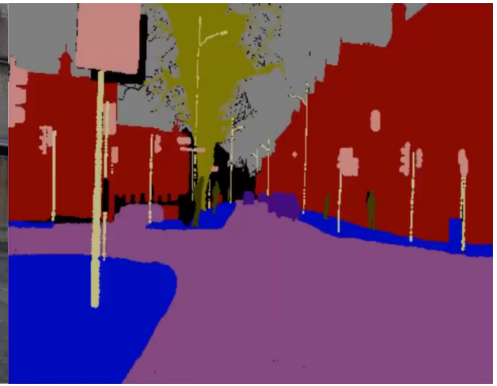


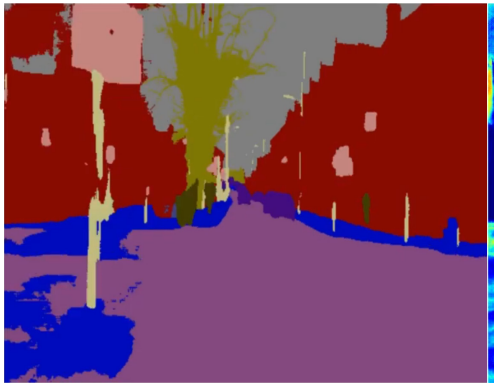

*Image by Michael Kana.*

Input

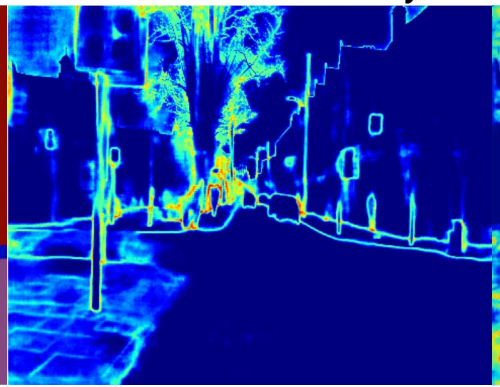
Ground truth



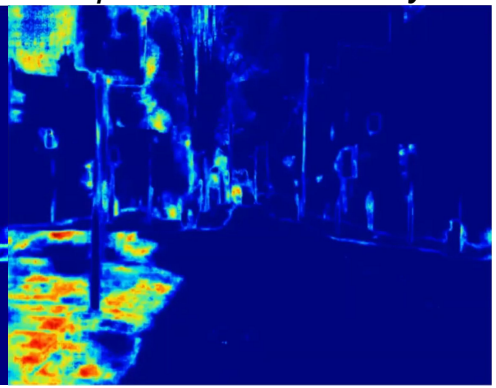
Prediction



Aleatoric uncertainty



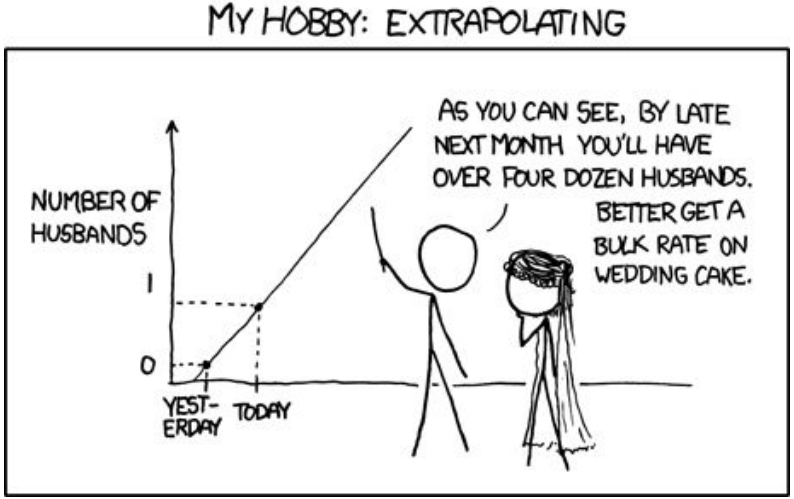
Epistemic uncertainty

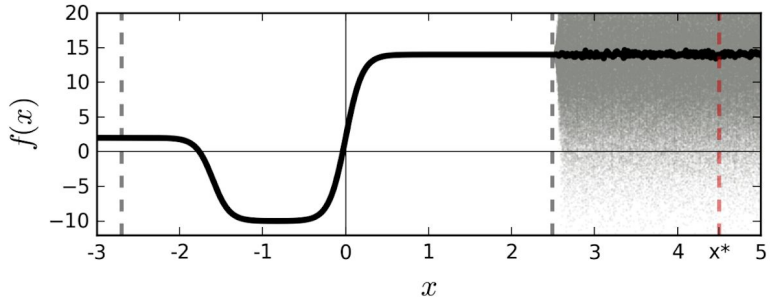


Kendall, A., & Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision?. In Advances in neural information processing systems (pp. 5574-5584).

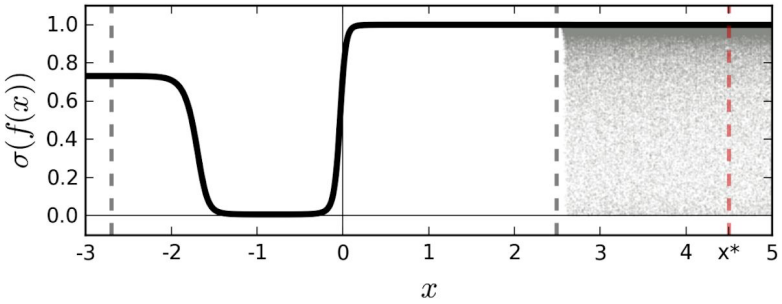
# Softmax outputs

- A cat-dog classifier knows ***nothing*** about warblers
- Outputs from trained softmax layer do not show model confidence




(a) Arbitrary function $f(\mathbf{x})$ as a function of data $\mathbf{x}$ (softmax *input*)

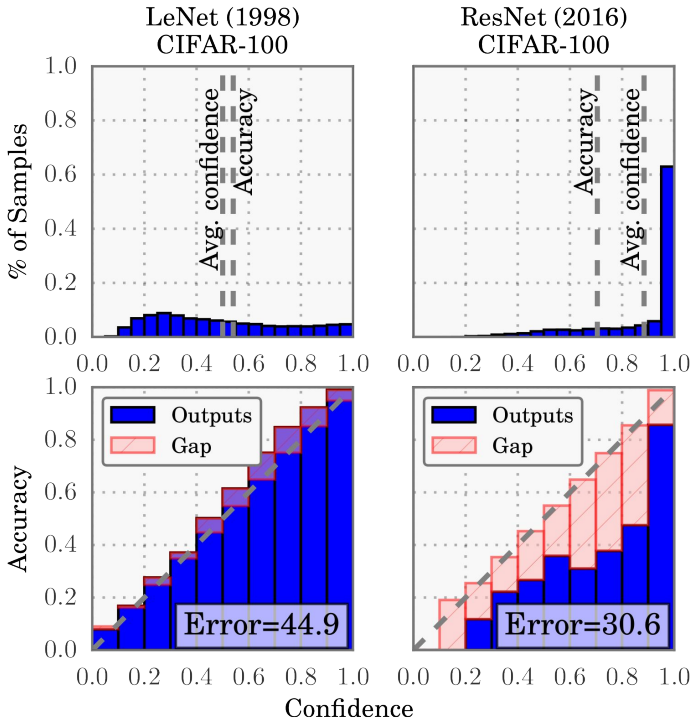



(b) $\sigma(f(\mathbf{x}))$ as a function of data $\mathbf{x}$ (softmax *output*)

*Image By Yarin Gal.*

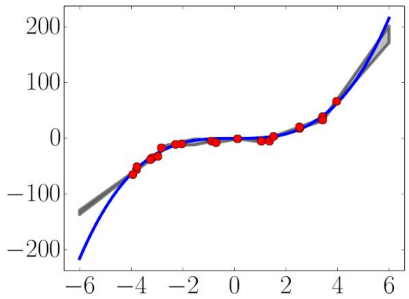
# Calibrating the softmax

- Expected Calibration Error:
  "confidence" matches accuracy
  - E.g. of 100 datapoints where confidence is 0.8, 80 of them should be correct.
- Model calibration declines, due to
  - Increased model capacity
  - Batch norm (allows for larger models)
  - Decreased weight decay
  - Overfitting to NLL loss (but not accuracy)
- Solutions
  - Histogram binning
  - Isotonic regression: piecewise constant function
  - Bayesian binning into quantiles: distribution over binning schemes

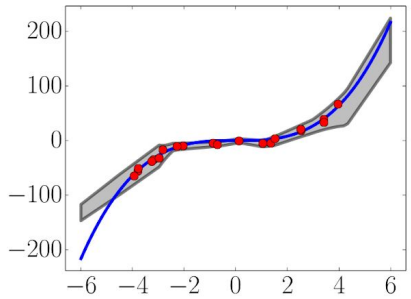



*Guo, C., et al. On calibration of modern neural networks. arXiv:1706.04599. ICML 2017.*

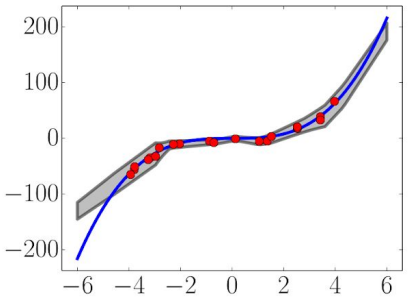
# Deep ensembles



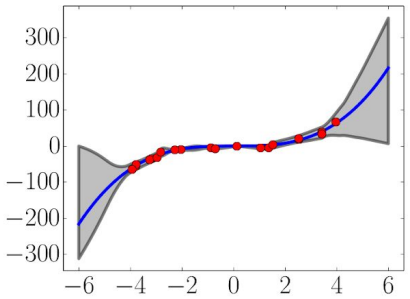

MSE (5 ensemble)




NLL (single)




NLL (single)
+adversarial

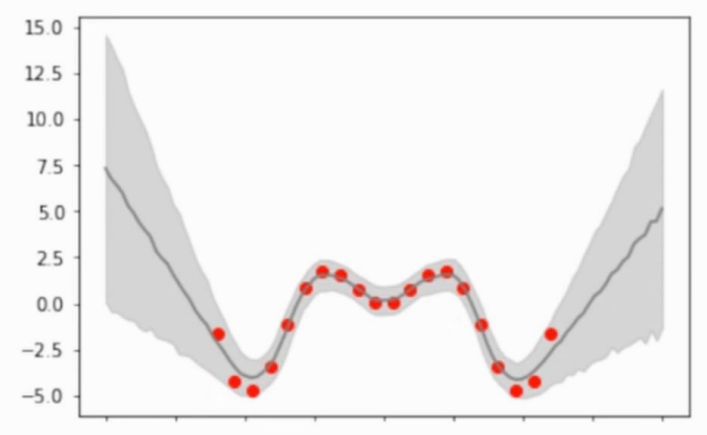



NLL (5 ensemble)
+adversarial

*Balaji, L., Pritzel, A., Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. NIPS. 2017.*

# Monte-Carlo Dropout

- Independently, with prob *p*, set each input to zero
- Exponential ensemble
- Monte-Carlo dropout:
  - Run network several times with different random seed.
- Equivalent to prior
  - (L2 weight decay equivalent to Gaussian prior).



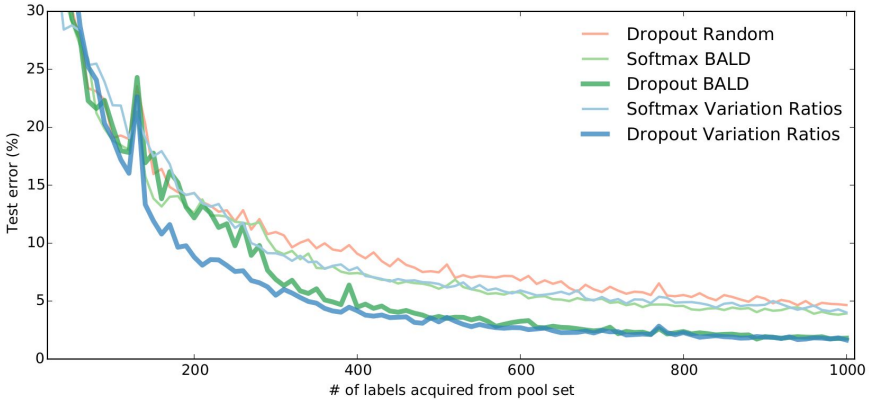

*Gal, Y., Ghahramani, Z., Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning, ICML 2018.*

RI.
SE

# MC-Dropout for

## Active learning

- High uncertainty - high information
- Data efficiency




## Deep RL

- Thompson sampling
- Data efficiency

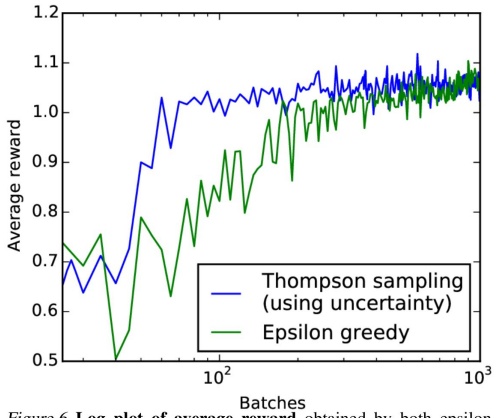
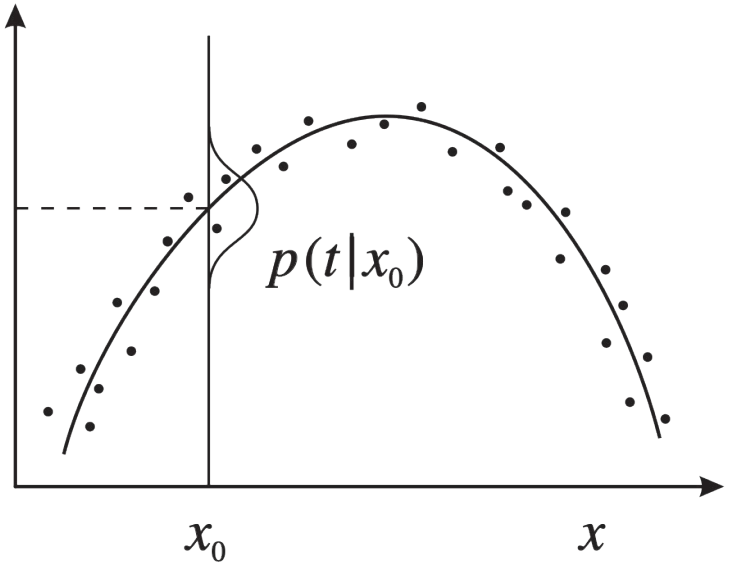



*Figure 6.* **Log plot of average reward** obtained by both epsilon greedy (in green) and our approach (in blue), as a function of the number of batches.

*Gal, Y., Ghahramani, Z., Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning, ICML 2018.*

# Density mixtures networks

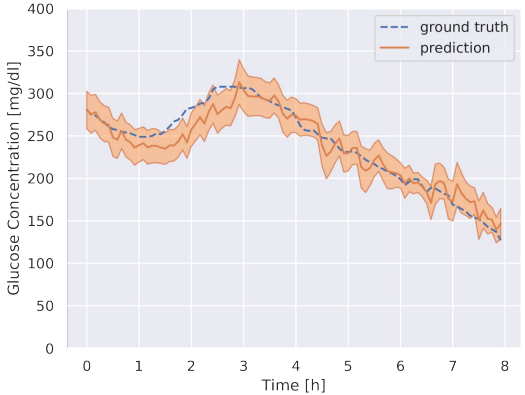
- Distributional parameter estimation
- Regression model with Gaussian output $t$
  - Train using NLL loss
- Enough mixture components
  - → arbitrary distribution approximation

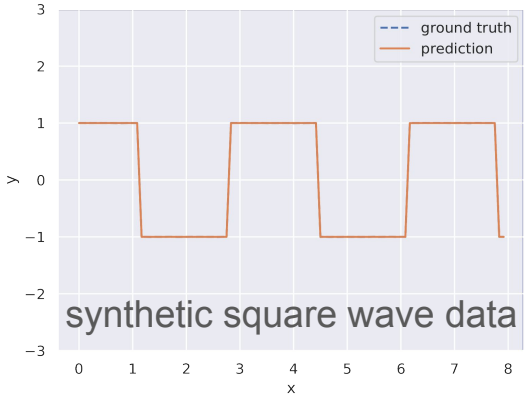


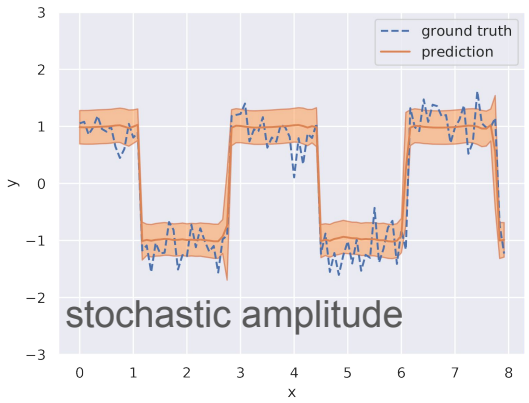

*Bishop, C.M., Mixture density networks, 1994.*

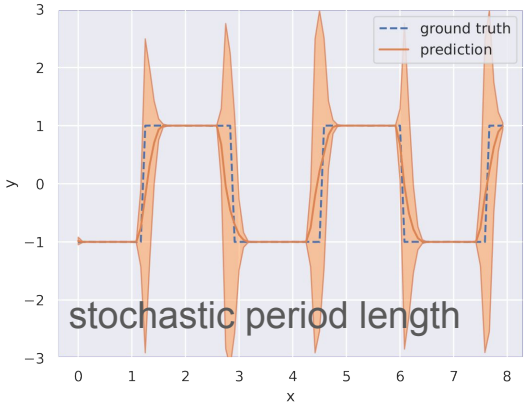
# Recurrent density networks: blood glucose predictions

blood glucose test data (Ohio T1DM dataset)



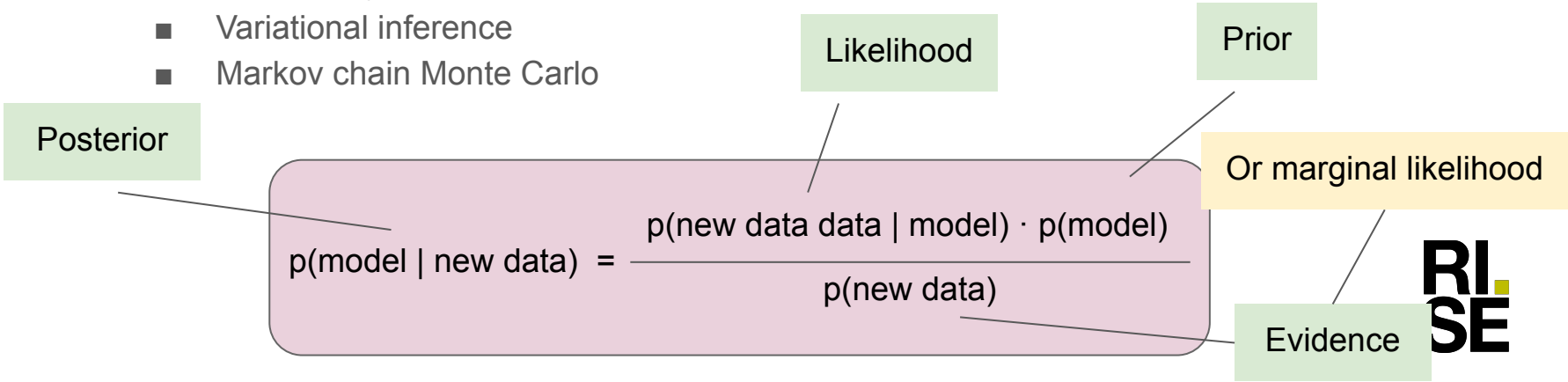




*Martinsson, J., Schliep, A., Eliasson, B., **Mogren, O.**, Blood glucose prediction with variance estimation using recurrent neural networks.Journal of Healthcare Informatics Research. 2020.*

# Bayesian machine learning

- Encoding and incorporating prior belief
  - Distribution over model parameters
- Posterior over model parameters
- Inference: marginalizing over latent parameters
- Computationally demanding
  - Evidence term requires expensive integral
  - Simple models: Conjugate priors
  - Approximate Bayesian methods:
    - Variational inference
    - Markov chain Monte Carlo



# Bayesian modelling

$$\mathbf{w}^{\text{MLE}} = \arg\max_{\mathbf{w}} \log P(\mathcal{D}|\mathbf{w})$$
$$= \arg\max_{\mathbf{w}} \sum_i \log P(\mathbf{y}_i|\mathbf{x}_i, \mathbf{w}).$$

$$\mathbf{w}^{\text{MAP}} = \arg\max_{\mathbf{w}} \log P(\mathbf{w}|\mathcal{D})$$
$$= \arg\max_{\mathbf{w}} \log P(\mathcal{D}|\mathbf{w}) + \log P(\mathbf{w}).$$

$$P(\hat{\mathbf{y}}|\hat{\mathbf{x}}) = \mathbb{E}_{P(\mathbf{w}|\mathcal{D})}[P(\hat{\mathbf{y}}|\hat{\mathbf{x}}, \mathbf{w})].$$

expectation under the posterior distribution on weights is equivalent to using an ensemble of an uncountably infinite number of models

RI. SE

# Variational inference

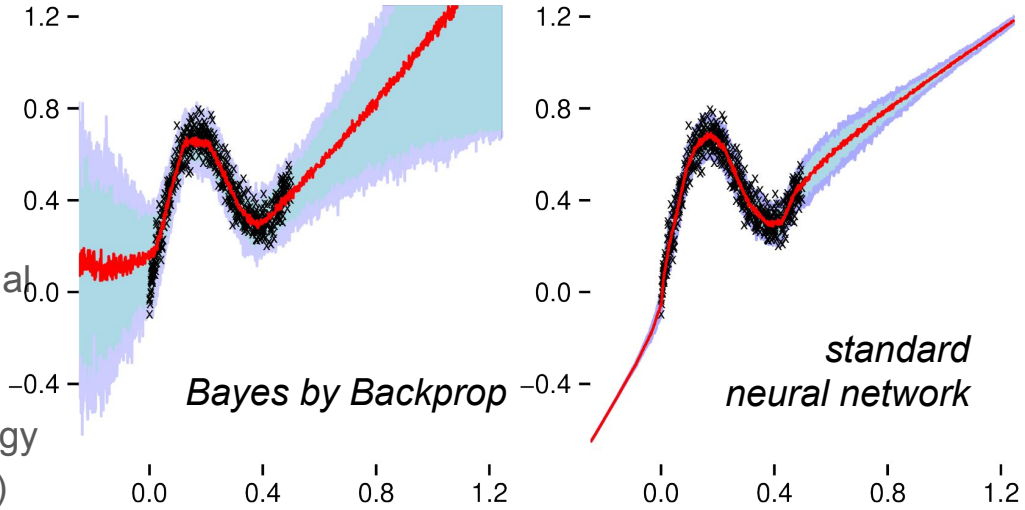
- True posterior *p*(w|X,Y) is intractable in general
- Define an approximating variational distribution $q_\theta$.
- Minimize KL btw q and p wrt $\theta$.
- Predictive distribution $p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{X}, \mathbf{Y}) \approx \int p(\mathbf{y}^*|\mathbf{x}^*, \boldsymbol{\omega}) q_\theta^*(\boldsymbol{\omega}) \mathrm{d}\boldsymbol{\omega} =: q_\theta^*(\mathbf{y}^*|\mathbf{x}^*)$
- Equivalent to maximizing the *evindence lower bound:*

$$\mathcal{L}_{\mathrm{VI}}(\theta) := \int q_\theta(\boldsymbol{\omega}) \log p(\mathbf{Y}|\mathbf{X}, \boldsymbol{\omega}) \mathrm{d}\boldsymbol{\omega} - \mathrm{KL}(q_\theta(\boldsymbol{\omega}) || p(\boldsymbol{\omega})) \leq \log p(\mathbf{Y}|\mathbf{X}) = \log \text{ evidence}$$

RI. SE

# Bayesian neural networks

- A prior on each weight
  - Random variable
  - Distribution over possible values
- Variational approximations
  - Numerical integration over variational posterior
  - Bayes by Backprop:
    - Minimize variational free energy (ELBO on marginal likelihood)
- Improve generalization




*Regression of noisy data with interquatile ranges. Black crosses are training samples. Red lines are median predictions. Blue/purple region is interquartile range.*


*MacKay, D.J.C., A Practical Bayesian Framework for Backpropagation Networks, Neural Computation, 1992,*
*Graves, A., Practical Variational Inference for Neural Networks, NIPS 2011*
*Blundell, et.al., Weight uncertainty in neural networks, ICML 2015*


RI.
SE

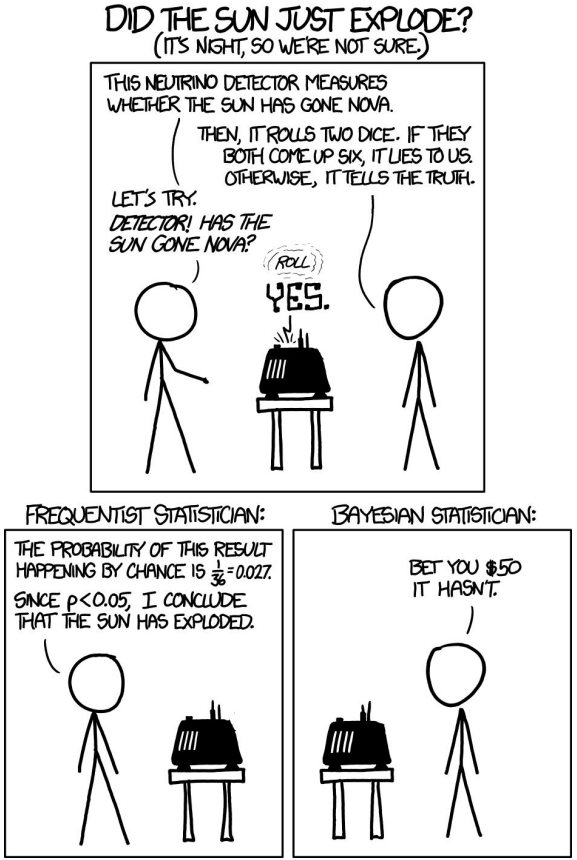
# Note on Bayesian methods

**Advantages:**

- Coherent
- Conceptually straightforward
- Modular
- Useful predictions

**Limitations:**

- Subjective. Assumptions.
- Computationally demanding
- Use of approximations weakens the coherence argument

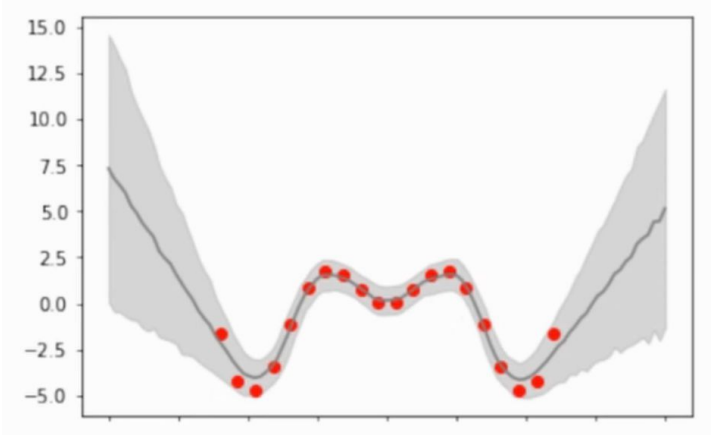





RI. SE

# Monte-Carlo Dropout

- Approximate posterior.
- MC Dropout is equivalent to an approximation of a deep Gaussian process.

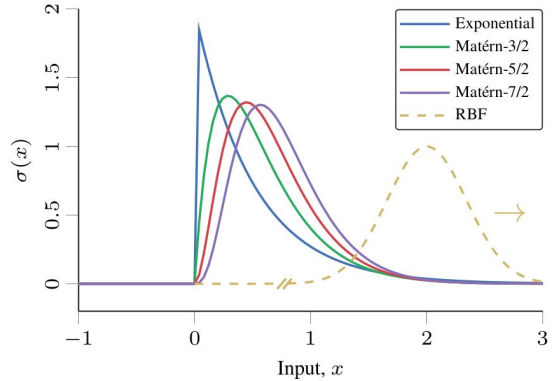



*Gal, Y., Ghahramani, Z., Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning, ICML 2018.*

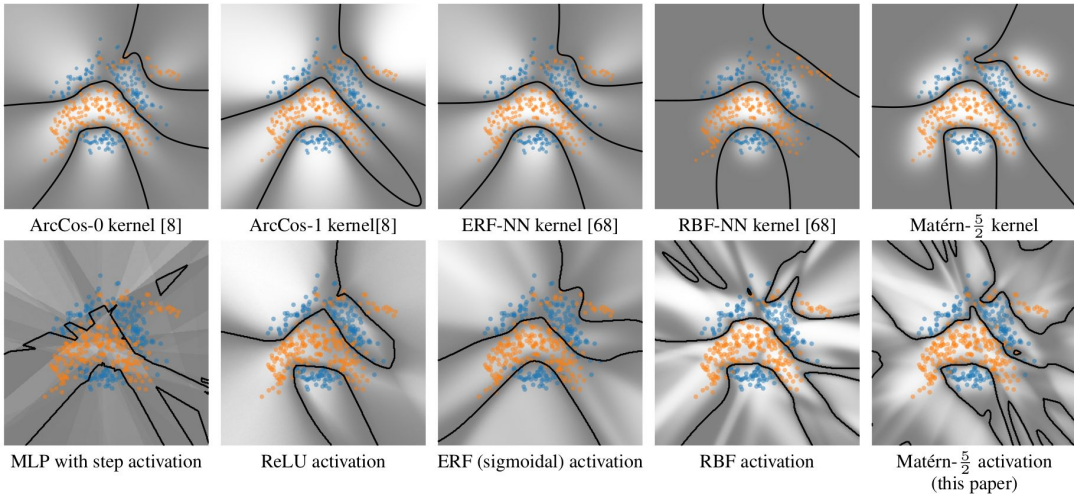
# Stationary Activations for Uncertainty Calibration in Deep Learning

- Matérn activation function
- MC-Dropout

$$\sigma(x) = \frac{q}{\Gamma(\nu + 1/2)} \Theta(x)\, x^{\nu - 1/2}\, \exp(-\lambda\, x)$$




(a) Activation function

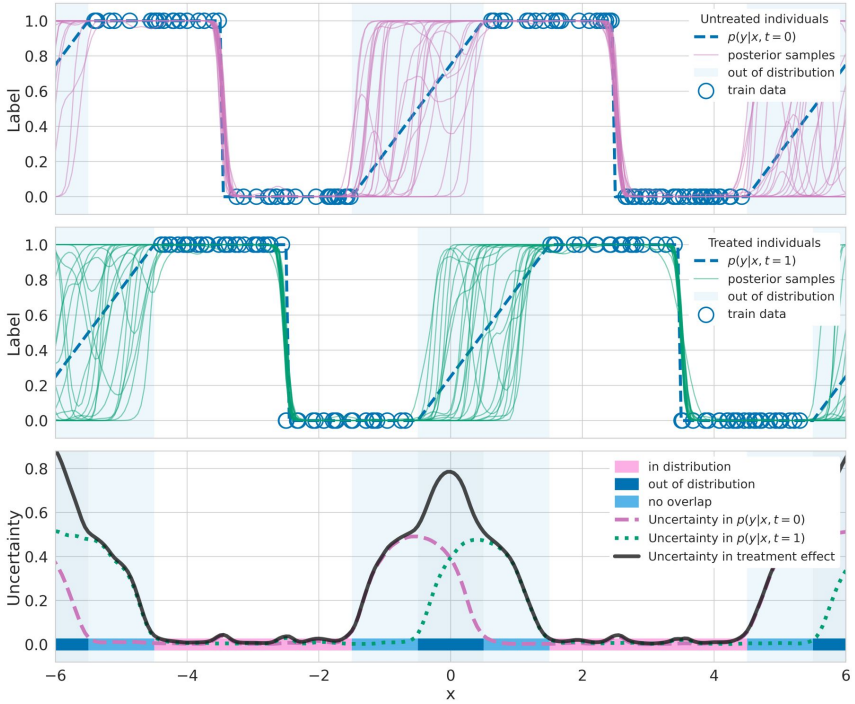



*White: Confident*
*Grey: Uncertain*
*Black: Decision boundary*
*Points: Training data*

*Meronen, L., Irwanto, C., & Solin, A. Stationary Activations for Uncertainty Calibration in Deep Learning. arXiv preprint arXiv:2010.09494. NeurIPS 2020.*

# Causal-Effect Inference Failure Detection

- Counterfactual deep learning models
- Epistemic uncertainty - covariate shift
- MC Dropout




*Jesson, A., Mindermann, S., Shalit, U., Gal, Y., Identifying Causal-Effect Inference Failure with Uncertainty-Aware Models, NeurIPS 2020*

# NeurIPS 2020

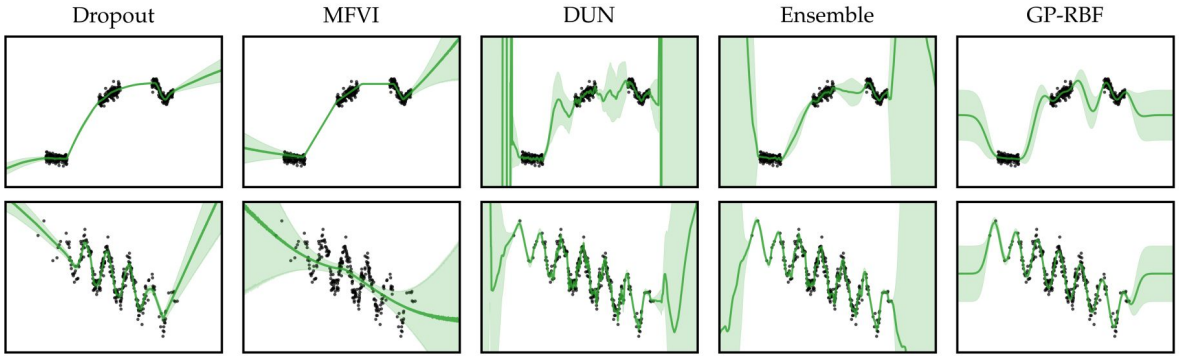
Antorán et.al., Depth Uncertainty in Neural Networks

Wenzel, et.al., Hyperparameter Ensembles for Robustness and Uncertainty Quantification

Valdenegro-Toro, et.al., Deep Sub-Ensembles for Fast Uncertainty Estimation in Image Classification

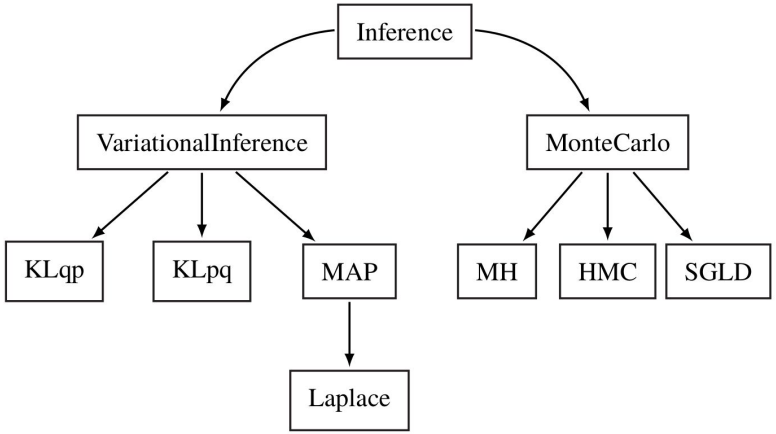
Lindinger, et.al., Beyond the Mean-Field: Structured Deep Gaussian Processes Improve the Predictive Uncertainties

Liu, et.al., Simple and Principled Uncertainty Estimation with Deterministic Deep Learning via Distance Awareness



# Getting started

- Bayesian Layers: A module for neural network uncertainty (Tran, et.al., 2019)
  - Implements variational approximation
- Edwardlib: A library for probabilistic modeling, inference, and criticism. (edwardlib.org)



# References


MacKay, D.J.C., A Practical Bayesian Framework for Backpropagation Networks, Neural Computation, 1992
Bishop, C.M., Mixture density networks, 1994
Graves, A., Practical Variational Inference for Neural Networks, NIPS 2011
Blundell, et.al., Weight uncertainty in neural networks, ICML 2015
Gal, Y., Ghahramani, Z., Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning, ICML 2015
Gal, Y., Uncertainty in Deep Learning, PhD thesis, 2016
Kendall, A., Gal, Y., What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?, arXiv:1703.04977, NIPS 2017.
Balaji, L., Pritzel, A., Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. NIPS. 2017.
Guo, C., et al. On calibration of modern neural networks. arXiv:1706.04599. ICML 2017
D. Tran, M. W. Dusenberry, D. Hafner, and M. van der Wilk. Bayesian Layers: A module for neural network uncertainty. NeurIPS 2019.
Martinsson, J., Schliep, A., Eliasson, B., Mogren, O., Blood glucose prediction with variance estimation using recurrent neural networks. Journal of Healthcare Informatics Research, JHIR, 2020.
Wilson, A.G. The case for Bayesian deep learning. arXiv:2001.10995, 2020.
Meronen, L., Irwanto, C., & Solin, A.  Stationary Activations for Uncertainty Calibration in Deep Learning. arXiv preprint arXiv:2010.09494. NeurIPS 2020.
Jesson, A., Mindermann, S., Shalit, U., Gal, Y., Identifying Causal-Effect Inference Failure with Uncertainty-Aware Models, NeurIPS 2020

Geoffrey E. Hinton and Drew van Camp. Keeping the neural networks simple by minimizing the description length of the weights
John S. Denker and Yann leCun., Transforming Neural-Net Output Levels to Probability Distributions
Radford M. Neal, Bayesian Learning for Neural Networks
David J.C. MacKay., A Practical Bayesian Framework for Backprop Networks

https://medium.com/@ODSC/introduction-to-bayesian-deep-learning-f7568f524c90
https://www.inovex.de/blog/uncertainty-quantification-deep-learning/
https://towardsdatascience.com/what-uncertainties-tell-you-in-bayesian-neural-networks-6fbd5f85648e
https://towardsdatascience.com/my-deep-learning-model-says-sorry-i-dont-know-the-answer-that-s-absolutely-ok-50ffa562cb0b


RI.
SE