Modelling Uncertainty for Learning Systems

> An Overview of Basics, Techniques and Performance Results

Andreas Sedlmeier | andreas.sedlmeier@ifi.lmu.de
Lehrstuhl für Mobile und Verteilte Systeme
Institut für Informatik | Prof. Dr. Claudia Linnhoff-Popien
LMU München

I. Motivation & Basics

Why consider uncertainty?

- Reliability and dependability requirements in industrial systems
- Low acceptable range for wrong predictions or decisions
- This differs from "Web-ML" applications where wrong predictions are often acceptable/accepted
- Most state-of-the-art "Web-ML" approaches ignore uncertainty

Example "Web-ML" VS "Industry-ML"

- **Web-ML:** Wrong classification of a male website visitor as female $\rightarrow$ display of "wrong category" advertisement
- **Industry-ML:** Wrong classification of a medical lesion image $\rightarrow$ malignant cancer stays undetected

Current Situation

Standard Deep Learning (DL) methods do not provide information about the uncertainty of their predictions

- **Reason:** The neural networks (NN) produce **point estimates**
- **Implicit assumption:** The point estimate is representative (This could be wrong, e.g. bc of multimodality, more later)

[Image src] https://xkcd.com/1838/

Basics

> **Basic Premise:** Models are used to perform statistical inference
> **Problem:** Predictions are prone to noise, wrong model inference & inductive assumptions

- **Learning Phase:** Optimization & Prediction
- **Approaches:** (Approximate) Bayesian & Ensemble Learning
- **Application Fields:** Computer Vision, Image Processing, Medical Image Analysis, Natural Language Processing, Reinforcement Learning (e.g. Robotics), Active Learning

Types of Uncertainty

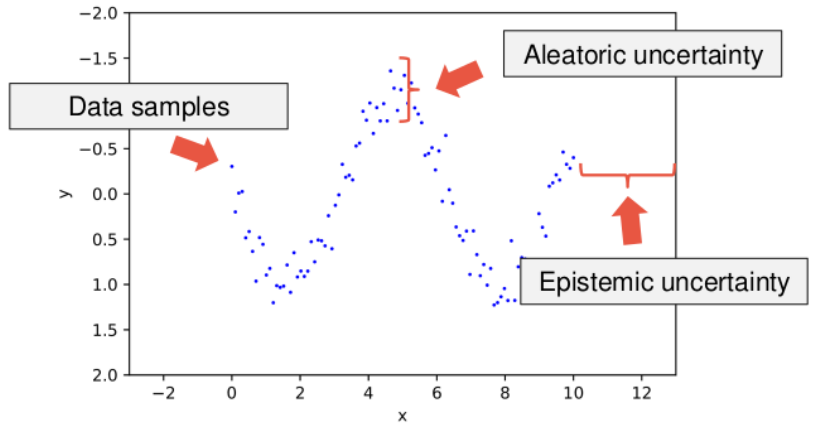
Aleatoric

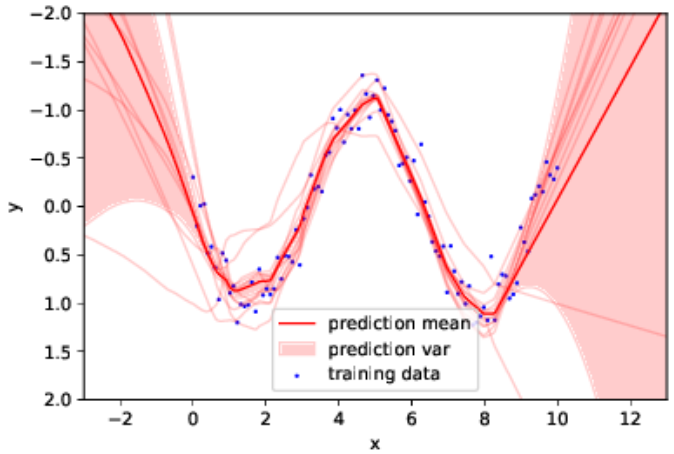
- aka **data uncertainty** aka **risk** [*]
- Stochasticity / Noise
- Irreducable
- Not a property of the model
- Homoscedastic / Heteroscedastic

Epistemic

- aka **knowledge uncertainty**
- lack of data to infer the underlying system's data generating function
- reducable by collecting more data

Predictive Uncertainty = Epistemic + Aleatoric

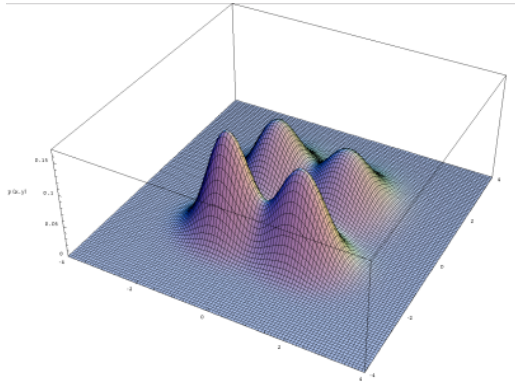




[*] For a discussion on why the differentiation between aleatoric/epistemic is essential, see e.g.
Osband, Ian. "Risk versus uncertainty in deep learning: Bayes, bootstrap and the dangers of dropout." NIPS Workshop on Bayesian Deep Learning. Vol. 192. 2016.

Multimodality

- Cannot be captured by models creating point predictions
- Often encountered with inverse problems
- E.g. determine the angle a two-linked robot arm should move to achieve a target location



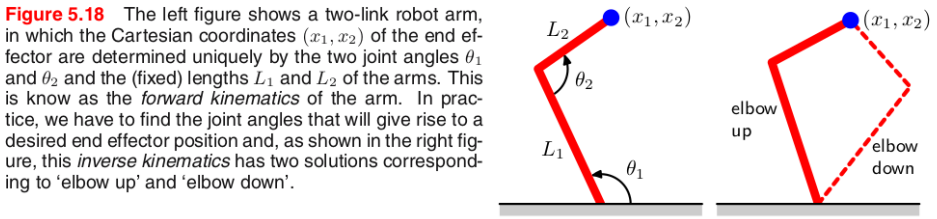
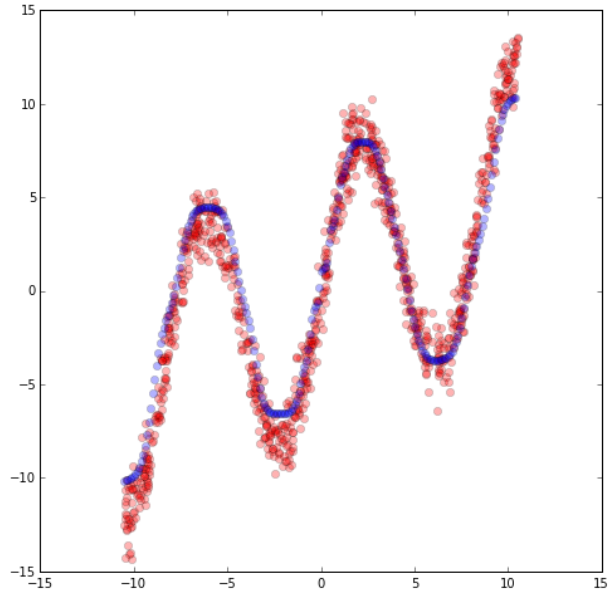


Figure 5.18 The left figure shows a two-link robot arm, in which the Cartesian coordinates (x_1, x_2) of the end effector are determined uniquely by the two joint angles θ_1 and θ_2 and the (fixed) lengths L_1 and L_2 of the arms. This is know as the *forward kinematics* of the arm. In practice, we have to find the joint angles that will give rise to a desired end effector position and, as shown in the right figure, this *inverse kinematics* has two solutions corresponding to 'elbow up' and 'elbow down'.

[Figure 5.18] C. M. Bishop. Pattern recognition & Machine Learning, 2006.

Multimodality toy example:

- Sinus $\rightarrow$ unimodal
- Inverse Sinus $\rightarrow$ multimodal
- Classic NN (using MSE as loss) fails
- Common solution approach: Mixture Density Networks (MDNs)



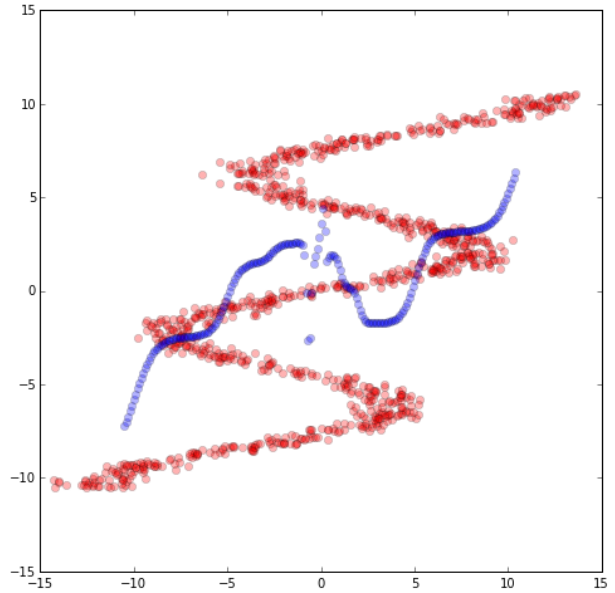
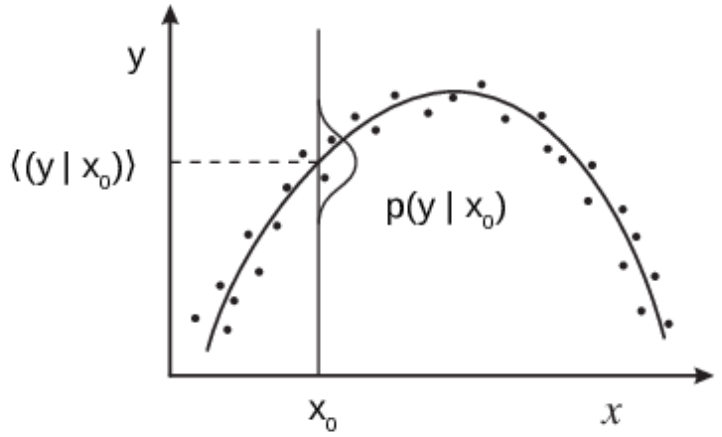


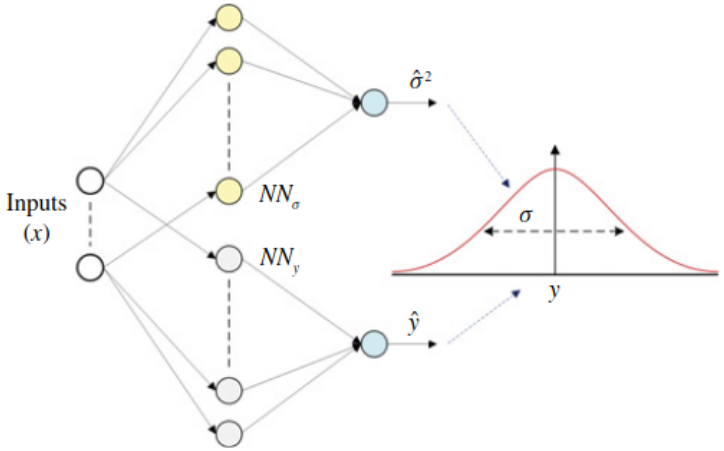
Image src: https://blog.otoro.net/2015/11/24/mixture-density-networks-with-tensorflow/

II. Uncertainty Modelling Techniques

Aleatoric Uncertainty (unimodal):

- Mean variance estimation (MVE) [1]
- Interpret outputs $\mu(x)$ and $\sigma(x)$ as samples from a (heteroscedastic) Gaussian
- Train by minimizing the negative log likelihood (NLL)
- For a review, see [2]





[1] Nix, David A., and Andreas S. Weigend. "Estimating the mean and variance of the target probability distribution." Proceedings of 1994 ieee international conference on neural networks (ICNN'94). Vol. 1. IEEE, 1994.

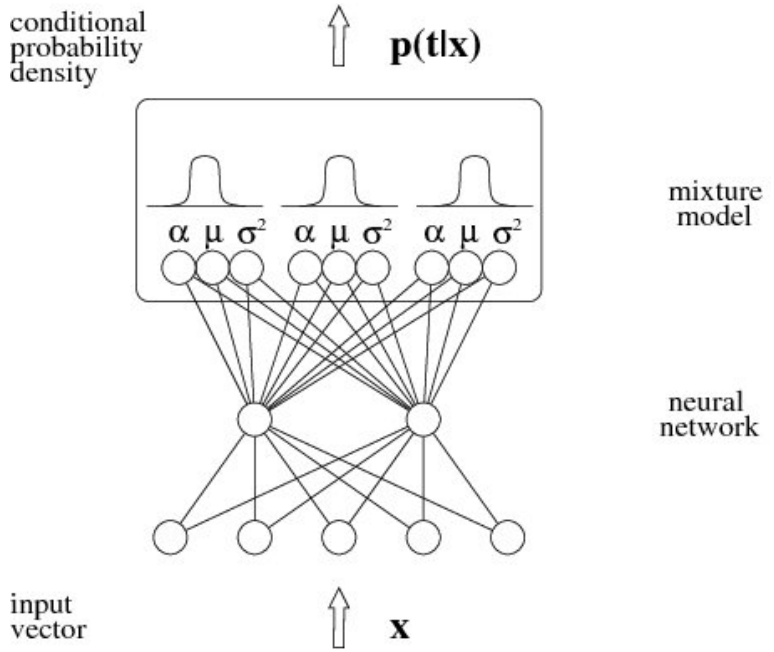
[2] Khosravi, Abbas, et al. "Comprehensive review of neural network-based prediction intervals and new advances." IEEE Transactions on neural networks 22.9 (2011): 1341-1356.

Aleatoric Uncertainty (multimodal):

- Mixture Density Networks (MDNs) - Bishop (1994) [1]
- Idea: replace the Gaussian distribution of the MVE with a mixture model
- Probability density is a linear combination of the form:

$$p(t|x) = \sum_{i=1}^{m} \alpha_i(x)\phi_i(t|x)$$

with $\alpha_i(x)$ being the mixing coefficients and $\phi_i(t|x)$ the conditional density of the target vector t for the i^{th} component



Note: Most formulas in modern work use z instead of α for the mixing coefficients and y instead of t for the label.

[1] Bishop, Christopher M. (1994). Mixture density networks. Technical Report. Aston University, Birmingham. (Unpublished)

Bayesian Neural Networks (BNNs)

- How can NNs correctly estimate uncertainty?
- **Bayesian Neural Networks**: Neil, Radford - 1995 [1]
- Combine Bayesian Methods with NNs
- Place probability distributions over model parameters
- For a long time practically unused, as method didn't scale

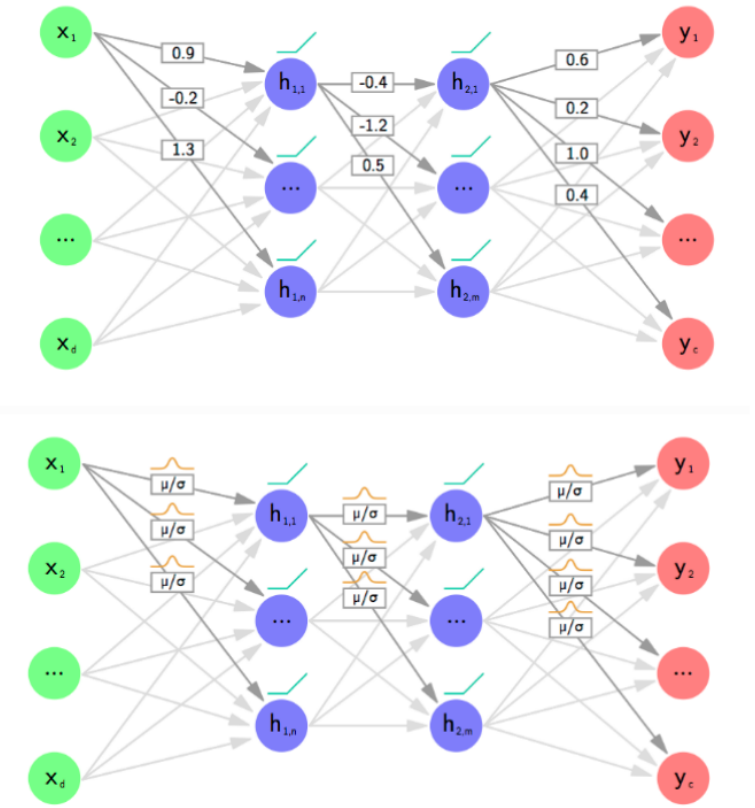


Figure 1: Top: Each filter weight has a fixed value, as in the case of frequentist Convolutional Networks. Bottom: Each filter weight has a distribution, as in the case of Bayesian Convolutional Networks. [16]

[1] Neal, Radford M. Bayesian learning for neural networks. Vol. 118. Springer Science & Business Media, 2012.
[Figure 1] Shridhar, Kumar, Felix Laumann, and Marcus Liwicki. "A comprehensive guide to bayesian convolutional neural network with variational inference." arXiv preprint arXiv:1901.02731 (2019).

Bayesian Inference in BNNs

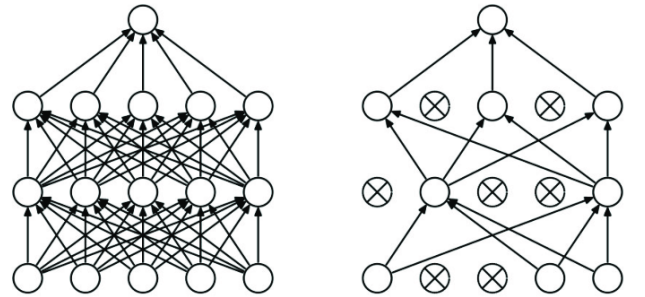
- True posterior intractable = cannot be computed analytically [1]
- Solution ~2015: Variational Inference (VI)
- Goal: Approximate the posterior distribution over the weights of the NN
- Considers the Bayesian Inference problem as an optimization problem
- $\rightarrow$ approximate the posterior distribution over the weights of the NN with a variational distribution q_θ e.g. a simple Gaussian

> **[Further "reading"]** DeepBayes2018 Workshop - Max Welling:
> Advanced methods of variational inference: https://youtu.be/mCBnid-1slI

[1] Huge number of parameters in an NN as well as the functional form does not lend itself to exact integration

Dropout

- Dropout was initially introduced to combat overfitting in DNNs [1]
- Idea: During Training, randomly drop (i.e. apply an independent random Bernoulli mask to the) activations of the NN (except output layer)
- Has become one of the most popular modern approaches to regularization in deep learning



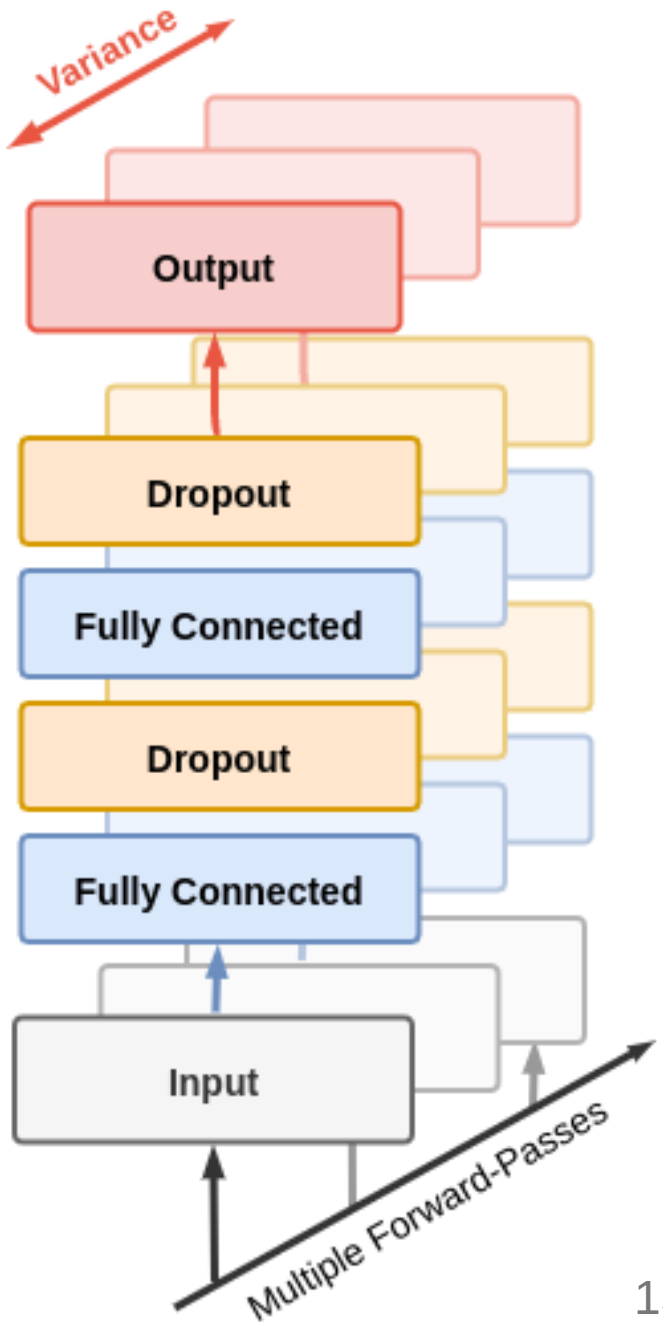
[1] Srivastava, Nitish, et al. "Dropout: a simple way to prevent neural networks from overfitting." The journal of machine learning research 15.1 (2014): 1929-1958.
[Image] Roffo, Giorgio. "Ranking to learn and learning to rank: On the role of ranking in pattern recognition applications." arXiv preprint arXiv:1706.05933 (2017).

Dropout Variational Inference aka MC Dropout

- Yarin Gal proposes MC (Monte-Carlo) Dropout as a practical approximate Bayesian inference technique [1]
- Idea: Use Dropout during the prediction phase
- Can also be applied to CNNs [3]

> "In fact, we shall see that we can we can get uncertainty information from existing deep learning models for free" [2]

> We show that the dropout objective, in effect, minimises the Kullback–Leibler divergence between an approximate distribution and the posterior of a deep Gaussian process. [2]



[1] Gal, Yarin, and Zoubin Ghahramani. "Dropout as a bayesian approximation: Representing model uncertainty in deep learning." international conference on machine learning. 2016.
[2] Gal, Yarin. "Uncertainty in deep learning." University of Cambridge 1.3 (2016).
[3] Gal, Yarin, and Zoubin Ghahramani. "Bayesian convolutional neural networks with Bernoulli approximate variational inference." arXiv preprint arXiv:1506.02158 (2015).

Critique:

> **Ian Osband** (2018): [1]
> Recent work has sought to understand dropout through a Bayesian lens, highlighting the connection to variational inference and arguing that the resultant dropout distribution approximates a Bayesian posterior. This narrative has proved popular despite the fact that [the] dropout distribution can be a poor approximation to most reasonable Bayesian posteriors.

[1] Osband, Ian, John Aslanides, and Albin Cassirer. "Randomized prior functions for deep reinforcement learning." Advances in Neural Information Processing Systems. 2018.

Problem1: Dropout distribution does not concentrate with observed data

Consequence: [*]

> No agent employing dropout for posterior approximation can tell the difference between observing a set of data once and observing it $N \gg 1$ times. This can lead to arbitrarily poor decision making [...]

- Possible solution: **Concrete Dropout**: Tune the dropout rate from data [1]
- Comment Osband: "Concrete dropout asymptotically improves the quality of the variational approximation, but provides no guarantees on its rate of convergence or error relative to exact Bayesian inference" [**]

---

[*] This would be a possible explanation, why Dropout failed for Out-of-distribution detection using epistemic uncertainty, as evaluated in:
A. Sedlmeier, et al. "Uncertainty-Based Out-of-Distribution Classification in Deep Reinforcement Learning," in 12th International Conference on Agents and Artificial Intelligence (ICAART 2020), 2020.
[**] Further discussion: What is the current state of dropout as Bayesian approximation?
https://web.archive.org/web/20190327225938if_/https://www.reddit.com/r/MachineLearning/comments/7bm4b2/d_what_is_the_current_state_of_dropout_as/
[1] Gal, Yarin, Jiri Hron, and Alex Kendall. "Concrete dropout." Advances in neural information processing systems. 2017.

Problem 2: VI can severely underestimate model uncertainty

- The objective function commonly used for VI is the ELBO (Expectation Lower Bound), which is known to underestimate the posterior variance
- For a comprehensive review see: Blei (2017): Variational inference: A review for statisticians [1]

> "The relative accuracy of variational inference and MCMC is still unknown. We do know that variational inference generally underestimates the variance of the posterior density; this is a consequence of its objective function"

[1] Blei, David M., Alp Kucukelbir, and Jon D. McAuliffe. "Variational inference: A review for statisticians." Journal of the American statistical Association 112.518 (2017): 859-877.

VI using different divergences

- Possible solution to Problem 2: Use alpha-divergences as alternative to VI's KL objective
- This avoids VI's uncertainty underestimation
- Hernandez-Lobato: Black-box alpha divergence [2]
- Yingzhen and Gal: Dropout inference in Bayesian neural networks with alpha-divergences [3]

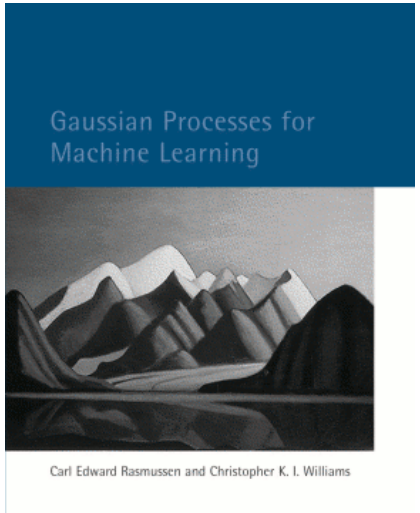
[2] Hernandez-Lobato, Jose, et al. "Black-box alpha divergence minimization." International Conference on Machine Learning. PMLR, 2016.
[3] Li, Yingzhen, and Yarin Gal. "Dropout inference in Bayesian neural networks with alpha-divergences." arXiv preprint arXiv:1703.02914 (2017).

Alternative approaches

- **Bayes by Backprop**: Blundell, Charles, et al. "Weight Uncertainty in Neural Network." International Conference on Machine Learning. 2015.
- An alternative approach to VI was recently presented that tries to efficiently compute the Bayesian posterior:
  - Dusenberry, Michael W., et al. **"Efficient and Scalable Bayesian Neural Nets with Rank-1 Factors."** arXiv preprint arXiv:2005.07186 (2020).

(Deep) Gaussian Processes

- What are Gaussian Processes (GPs)?
  - Nonparametric Bayesian Model - capacity grows with the available data
  - GPs define a probability distribution over possible functions
  - Considered the "gold standard for faithfully representing predictive uncertainty"
  - BUT: Scalability problems: The standard GP exhibits a runtime complexity O(N^3) and memory complexity of O(N^2), where N is the size of the dataset [1]
  - $\rightarrow$ Restriced to problems with fewer than about ten thousand training points
- Recent scalability developments:
  - Deep Gaussian Processes: Combine DNNs with GPs [1]
  - Exact Gaussian Processes on a Million Data Points: [NeurIPS 2019]
    - Uses multi-GPU parallelization, linear conjugate gradients, accessing the kernel matrix only through matrix multiplication, ...



[1] Leveraging uncertainty information from deep neural networks for disease detection: https://www.nature.com/articles/s41598-017-17876-z
[2] Damianou, Andreas, and Neil Lawrence. "Deep gaussian processes." Artificial Intelligence and Statistics. 2013.

Further reading:

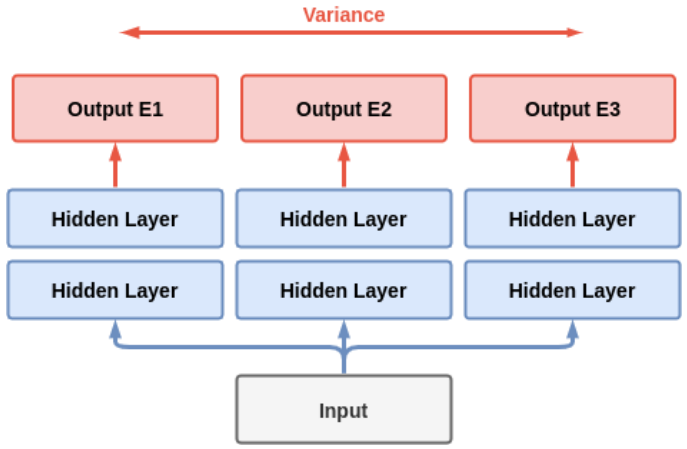
> - Rasmussen, C. E. & Williams, C. K. I. Gaussian processes for machine learning, vol. 1 (MIT press Cambridge, 2006).
> - Gaussian Processes are Not So Fancy: https://planspace.org/20181226-gaussian_processes_are_not_so_fancy/
> - Gaussian Process, not quite for dummies: https://yugeten.github.io/posts/2019/09/GP/
> - A Visual Exploration of Gaussian Processes: https://distill.pub/2019/visual-exploration-gaussian-processes/

Ensemble Techniques

> Simple and scalable predictive uncertainty estimation using deep ensembles
> - Lakshminarayanan, Pritzel, Blundell (2017) [1]

One of the first works to apply ensemble ideas to deep NNs in order to investigate predictive uncertainty performance:

- proposes an alternative to Bayesian NNs
- simple to implement
- readily parallelizable
- requires very little hyperparameter tuning
- yields high quality predictive uncertainty estimates



Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell. "Simple and scalable predictive uncertainty estimation using deep ensembles." Advances in neural information processing systems 30 (2017): 6402-6413.

Details

- For regression tasks, use a network that outputs $\mu(x)$ and $\sigma(x)$
- Treat the observed values as samples from a (heteroscedastic) Gaussian
- Train by minimizing the negative log-likelihood:

$$-\log p_\theta(y_n|\mathbf{x}_n) = \frac{\log \sigma_\theta^2(x)}{2} + \frac{(y-\mu_\theta(x))^2}{2\sigma_\theta^2(x)} + \text{constant}$$

- Coding detail: Enforce positivity of the $\sigma(x)$ output neuron! [1]
- To get the final prediction, the ensemble is treated as a uniformly-weighted mixture model (i.e. predictions/predicted class probabilities are averaged)

[1] The authors enforce the positivity constraint on the variance by passing the second output through the softplus function log(1 + exp(·)), and add a minimum variance of 10^{-6} for numerical stability

Evaluation

Problem: Empirical evidence of uncertainty estimates are not available in general, quality of predictive uncertainty evaluation is a challenging task.

What do we want?

> Well-calibrated predictions that are robust to model misspecification and dataset shift. - Lakshminarayanan (2016)

What is calibration?

- a frequentist notion of uncertainty
- measures the discrepancy between predictions and (empirical) long-run frequencies
- The quality of calibration can be measured by proper scoring rules such as log predictive probabilities and the Brier score.
- calibration is independant of accuracy: predictions can be accurate yet miscalibrated, as well as calibrated but inaccurate

What are scoring rules?

- Scoring rules measure the quality of predictive uncertainty
- They assigns a numerical score to a predictive distribution $p(y|x)$, with better calibrated predictions receiving higher scores

Proper scoring rules:

- minimizing NLL (Negative log likelihood, note: equal to softmax cross entropy loss)
- Brier score (equivalent to the MSE between predicted probabilities and one-hot labels)

Generalization:

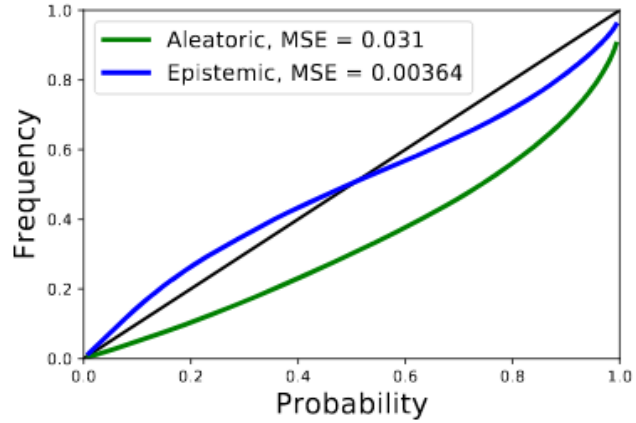
- We also want generalization of the predictive uncertainty to domain shift (aka out-of-distribution (OOD) data)
- $\rightarrow$ measure if the network knows what it knows

How to analyze calibration?

Reliability Diagrams aka Calibration Plots

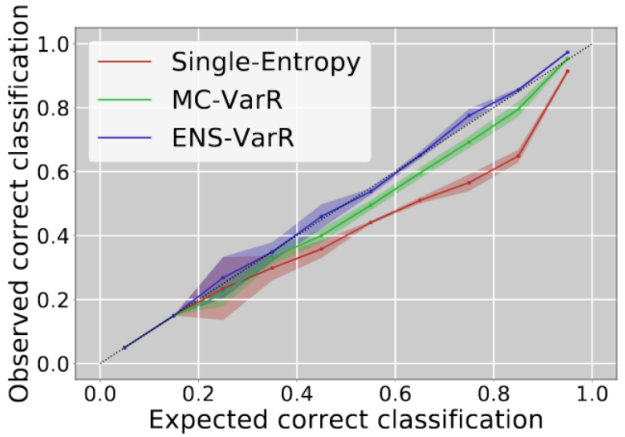
Kendall [1]:

> "To form calibration plots for classification models, we discretize our model's predicted probabilities into a number of bins, for all classes and all pixels in the test set. We then plot the frequency of correctly predicted labels for each bin of probability values. Better performing uncertainty estimates should correlate more accurately with the line $y = x$ in the calibration plots."



Beluch [2]:

> "To assess calibration quality we determine whether the expected fraction of correct classifications (as predicted by the model confidence, i.e.the uncertainty over predictions) matches the observed fraction of correct classifications. When plotting both values against each other, a well-calibrated model lies close to the diagonal."

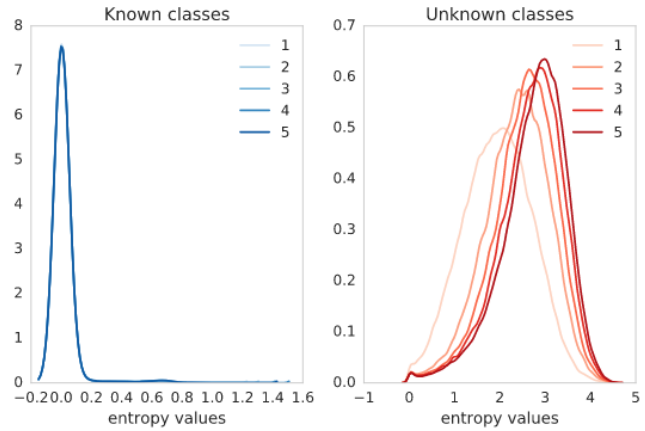


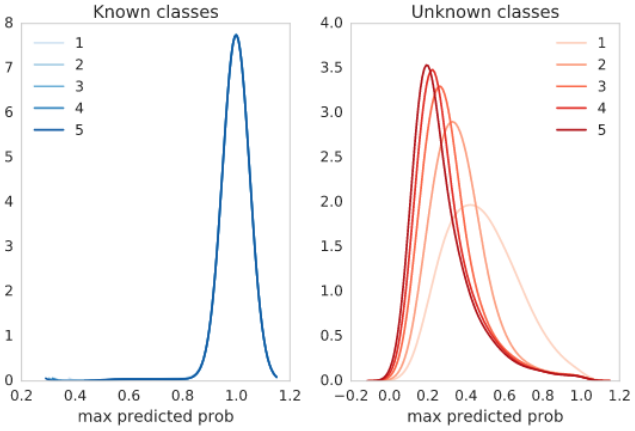
[1] Kendall, Alex, and Yarin Gal. "What uncertainties do we need in bayesian deep learning for computer vision?." Advances in neural information processing systems. 2017.
[2] Beluch, William H., et al. "The power of ensembles for active learning in image classification." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.
[Further reading] Zadrozny, Bianca, and Charles Elkan. "Transforming classifier scores into accurate multiclass probability estimates." Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. 2002.

Results in Lakshminarayanan et al.: [1]

Out-of-distribution (OOD) results on ImageNet

- ImageNet trained only on dogs
- Tested on non-dogs
- Ensemble performs as expected:
  - Predictive entropy is a lot higher for the unknown classes (non-dogs), i.e. the model is uncertain
  - while max. predicted probability is lower

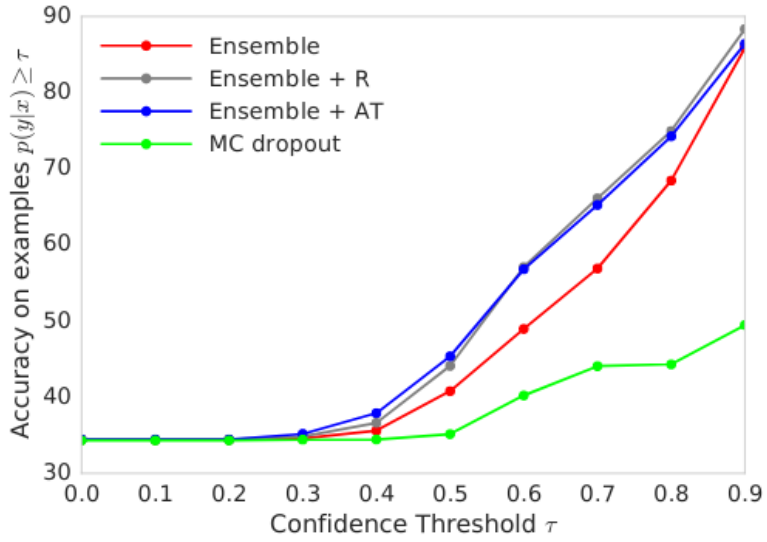




[1] Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell. "Simple and scalable predictive uncertainty estimation using deep ensembles." Advances in neural information processing systems 30 (2017): 6402-6413.

OOD on MNIST:

- Trained on MNIST, tested on combined MNIST (known) and NotMNIST (unknown) data
- Deep ensembles perform robust
- MC Dropout produces overconfident wrong predictions
- (This matches previous observation that VI can underestimate uncertainty)



(Ensemble + AT: Adversarially trained ensemble)
As confidence is a continuous variable, it appears the authors binned the values using a bin-size of 0.1. (Paper does not state this clearly).
Quote: "We filter out test examples, corresponding to a particular confidence threshold and plot the accuracy for this threshold."

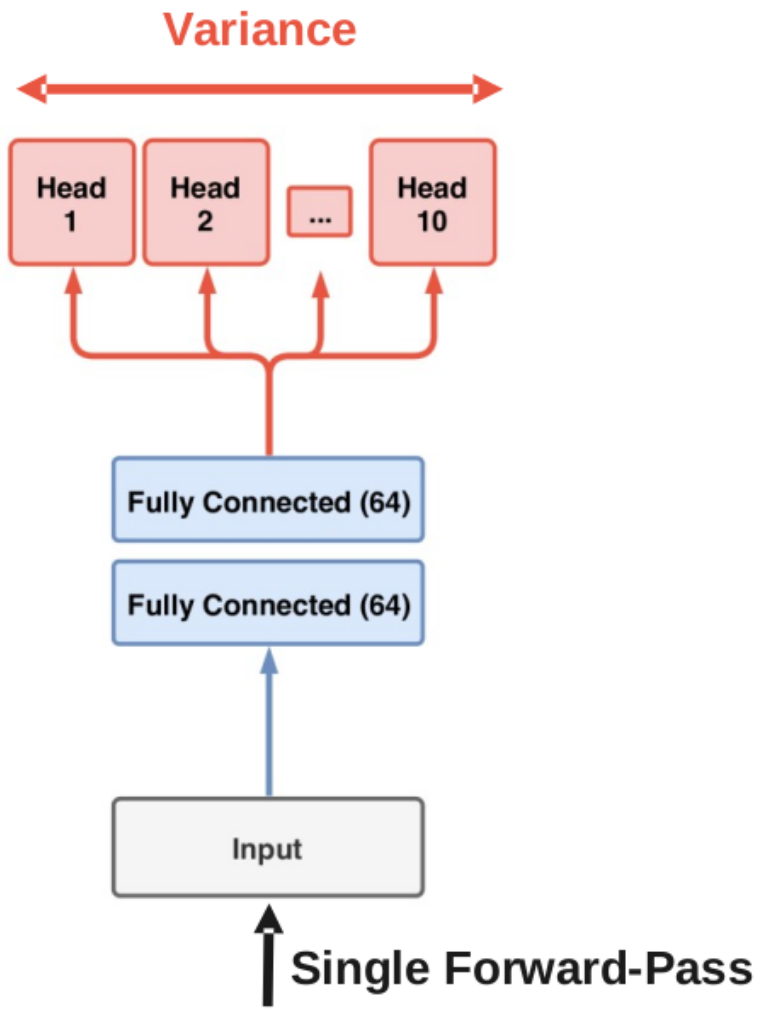
Thoughts on Bayesian NNs / Ensembles

- Read (the introduction of) the Lakshminarayanan Paper for a highly interesting overview of the goal, problem and methods in the field of uncertainty and probabilistic methods.

> The quality of predictive uncertainty obtained using Bayesian NNs crucially depends on (i) the degree of approximation due to computational constraints and (ii) if the prior distribution is ‘correct’, as priors of convenience can lead to unreasonable predictive uncertainties.
> [...]
> Interestingly, dropout may also be interpreted as ensemble model combination where the predictions are averaged over an ensemble of NNs (with parameter sharing). The ensemble interpretation seems more plausible particularly in the scenario where the dropout rates are not tuned based on the training data, since any sensible approximation to the true Bayesian posterior distribution has to depend on the training data.

Bootstrap Ensemble

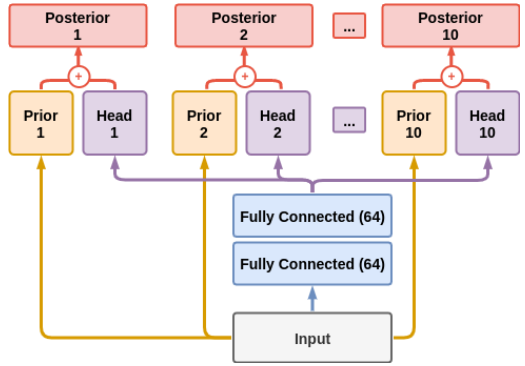
- Bootstrapped DQN (Deep Q-Network) proposed as efficient ensemble architecture [1]
- Single NN $\rightarrow$ ensemble members share most weights
- Output-Layer is split into so-called "heads", representing the individual ensemble outputs
- Bootstrap training procedure implemented by using boolean mask $\rightarrow$ training data set of each member differs slightly



[1] Osband, Ian, et al. "Deep exploration via bootstrapped DQN." Advances in neural information processing systems 29 (2016): 4026-4034.

Bootstrap Prior Networks

- Improvement of the previous architecture to increase ensemble diversity
- Solution to the observation that naive ensembles trained from random initializations can fit the data exactly which leads to almost zero uncertainty anywhere in the space
- Adds output of a fixed (untrainable) prior network to each head



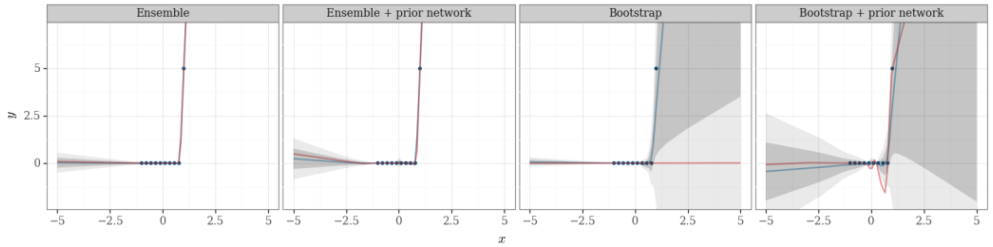


Figure 13: Posterior predictive distributions for ensemble uncertainty.

[1] Osband, Ian, John Aslanides, and Albin Cassirer. "Randomized prior functions for deep reinforcement learning." Advances in Neural Information Processing Systems. 2018.
[Figure 13] See supplementals of above paper

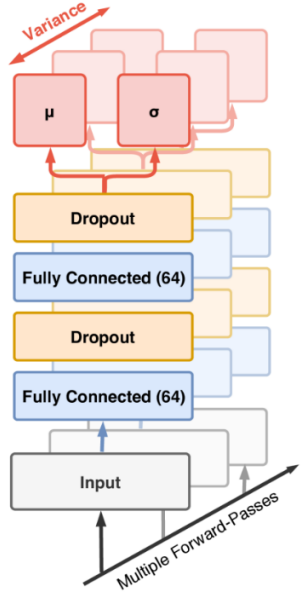
III. Applications & Evaluations

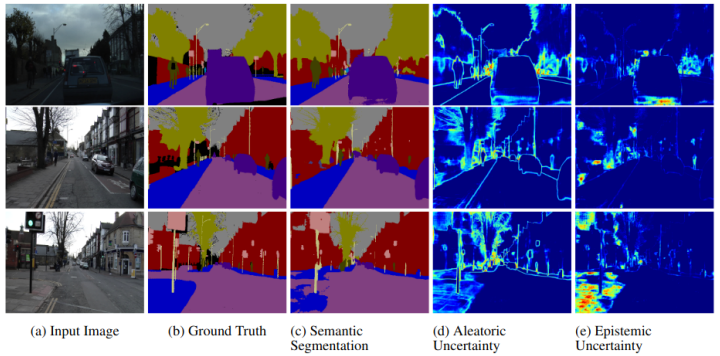
Computer Vision

Kendall, Gal, et al. : What uncertainties do we need in bayesian deep learning for computer vision?

Interesting aspects:

- Combine aleatoric and epistemic uncertainty modelling into single model:
- MVE (they call it MAP inference) for aleatoric + MC Dropout for epistemic uncertainty
- Loss uses L1 distance (Laplacian Prior instead of L2 distance - Gaussian prior)
- Modelling uncertainty increases performance (works as loss attenuation)
- Modelling aleatoric uncertainty increases performance more than epistemic
- Combining both results in best performance
- Uncertainties behave as expected: Precision is lower, when image contains more points that the model is uncertain about





Kendall, Alex, and Yarin Gal. "What uncertainties do we need in bayesian deep learning for computer vision?." Advances in neural information processing systems. 2017.

The power of ensembles for active learning in image classification

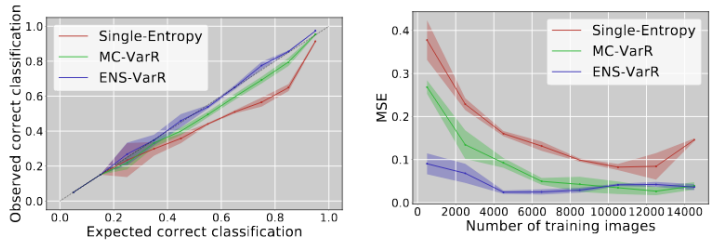
CVPR 2018 (Authors from Bosch Center for Artificial Intelligence)

- Compare ensemble-based architectures against Monte-Carlo Dropout

Results:

- Ensemble-based uncertainties outperform other methods of uncertainty estimation (in particular MC Dropout)

> "We find that the difference in active learning performance can be explained by a combination of decreased model capacity and lower diversity of MC Dropout ensembles"



(a) (b)

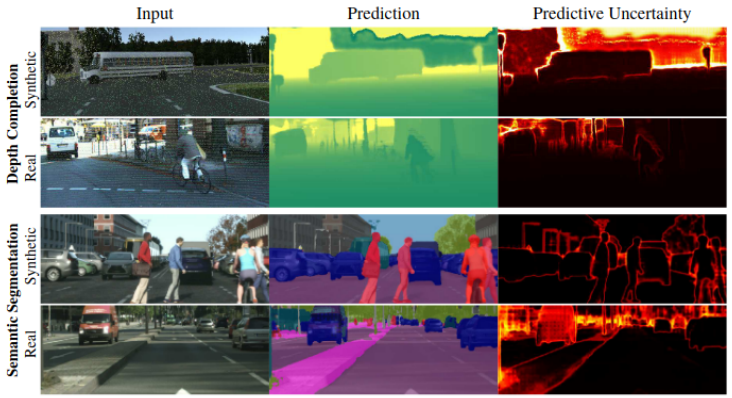
Figure 3: a) Calibration plot after 3 acquisition steps (6, 500 images) for CIFAR-10 and the DenseNet. Ideal calibration is on the dashed diagonal. b) Mean squared error for the calibration lines for different number of acquired images.

Beluch, William H., et al. "The power of ensembles for active learning in image classification." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

Evaluating Scalable Bayesian Deep Learning Methods for Robust Computer Vision

CVPR 2020 (ETH Zürich & Uppsala University)

- Confirm that ensembling consistently outperforms MC Dropout and provides more reliable and practical uncertainty estimates
- Depth Completion & Semantic Segmentation tasks
- Attribute the success of ensembling to its ability to capture multi-modality present in the posterior distribution



Gustafsson, Fredrik K., Martin Danelljan, and Thomas B. Schon. "Evaluating scalable bayesian deep learning methods for robust computer vision." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. 2020.

Other fields of application

RL

- Kahn: "Uncertainty-aware reinforcement learning for collision avoidance" (Bootstrapping & Dropout)
- Osband: "Deep exploration via bootstrapped DQN" as well as follow-up work
- Sedlmeier: "Uncertainty-based Out-of-Distribution Classification in Deep Reinforcement Learning"

Medical

- Leveraging uncertainty information from deep neural networks for disease detection
  https://www.nature.com/articles/s41598-017-17876-z

[1] Kahn, Gregory, et al. "Uncertainty-aware reinforcement learning for collision avoidance." arXiv preprint arXiv:1702.01182 (2017).
[2] Osband, Ian, et al. "Deep exploration via bootstrapped DQN." Advances in neural information processing systems 29 (2016): 4026-4034.
[3] Sedlmeier, Andreas et al. "Uncertainty-based Out-of-Distribution Classification in Deep Reinforcement Learning." In Proceedings of the 12th International Conference on Agents and Artificial Intelligence

Thank you! Questions?

Andreas Sedlmeier | andreas.sedlmeier@ifi.lmu.de
Lehrstuhl für Mobile und Verteilte Systeme
Institut für Informatik | Prof. Dr. Claudia Linnhoff-Popien
LMU München