Fast Computation of Uncertainty in Deep Learning

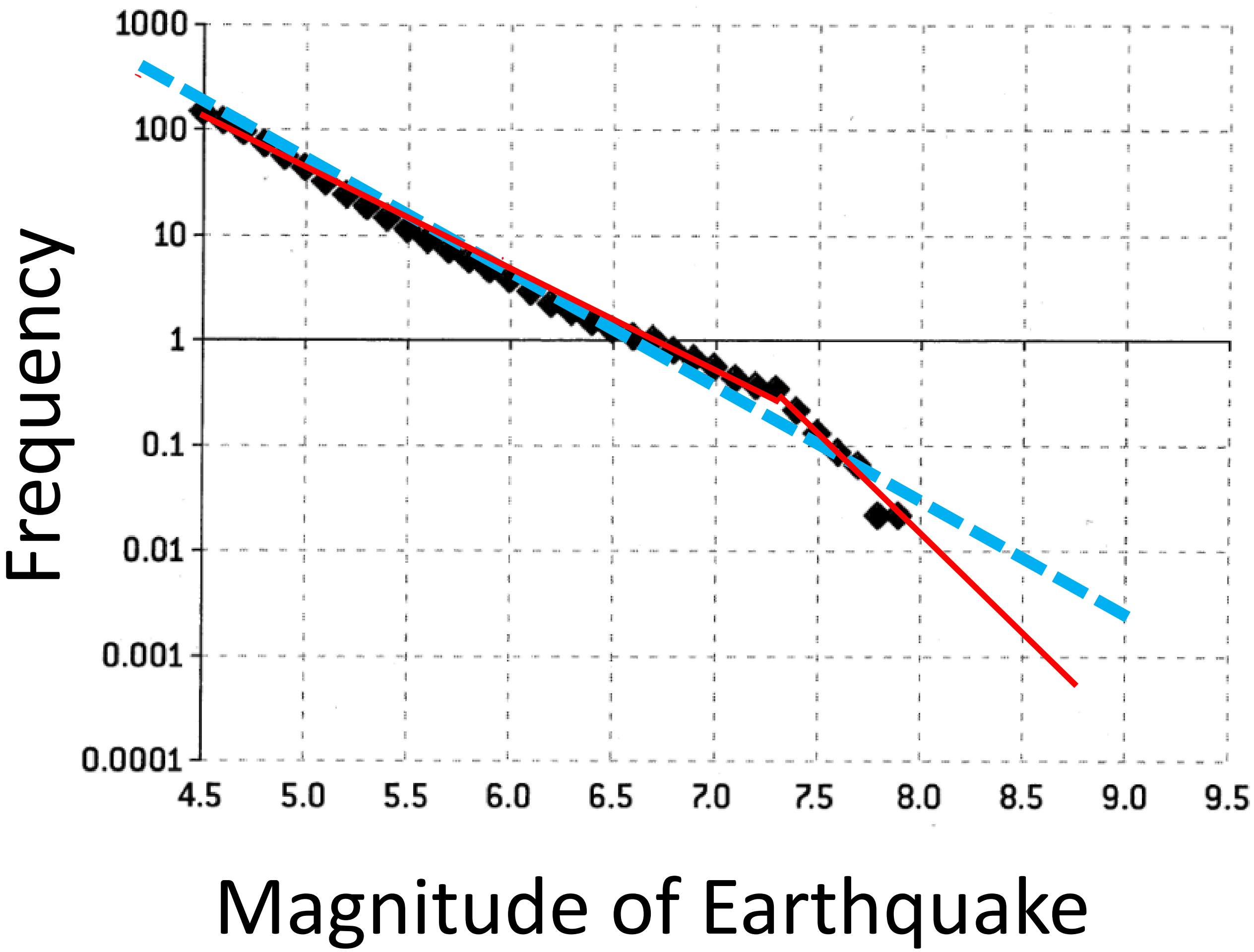
Mohammad Emtiyaz Khan
RIKEN Center for AI Project, Tokyo, Japan

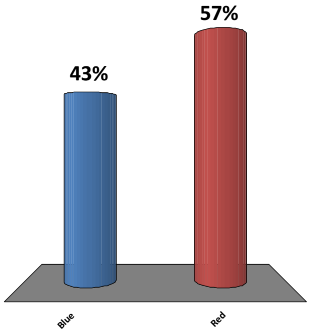
Joint work with
Wu Lin (UBC), Didrik Nielsen (RIKEN), Voot Tangkaratt (RIKEN)
Yarin Gal (University of Oxford), Akash Srivastava (University of Edinburgh)
Zuozhu Liu (SUTD, Singapore)

Uncertainty

Quantifies the confidence in the prediction of a model, i.e., how much it does not know.

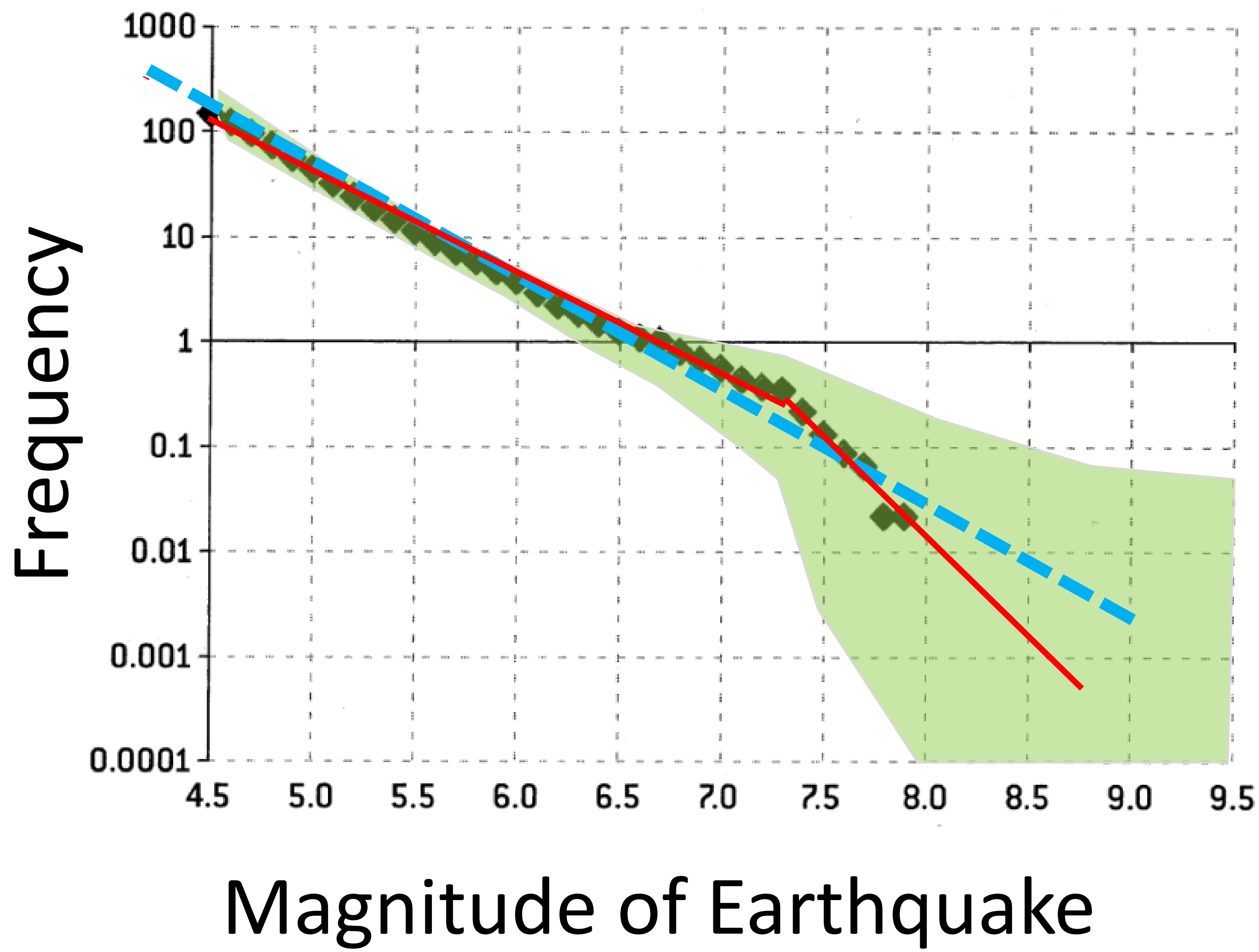
Example: Which is a Better Fit?





Real data from Tohoku (Japan). Example taken from Nate Silver's book “The signal and noise”

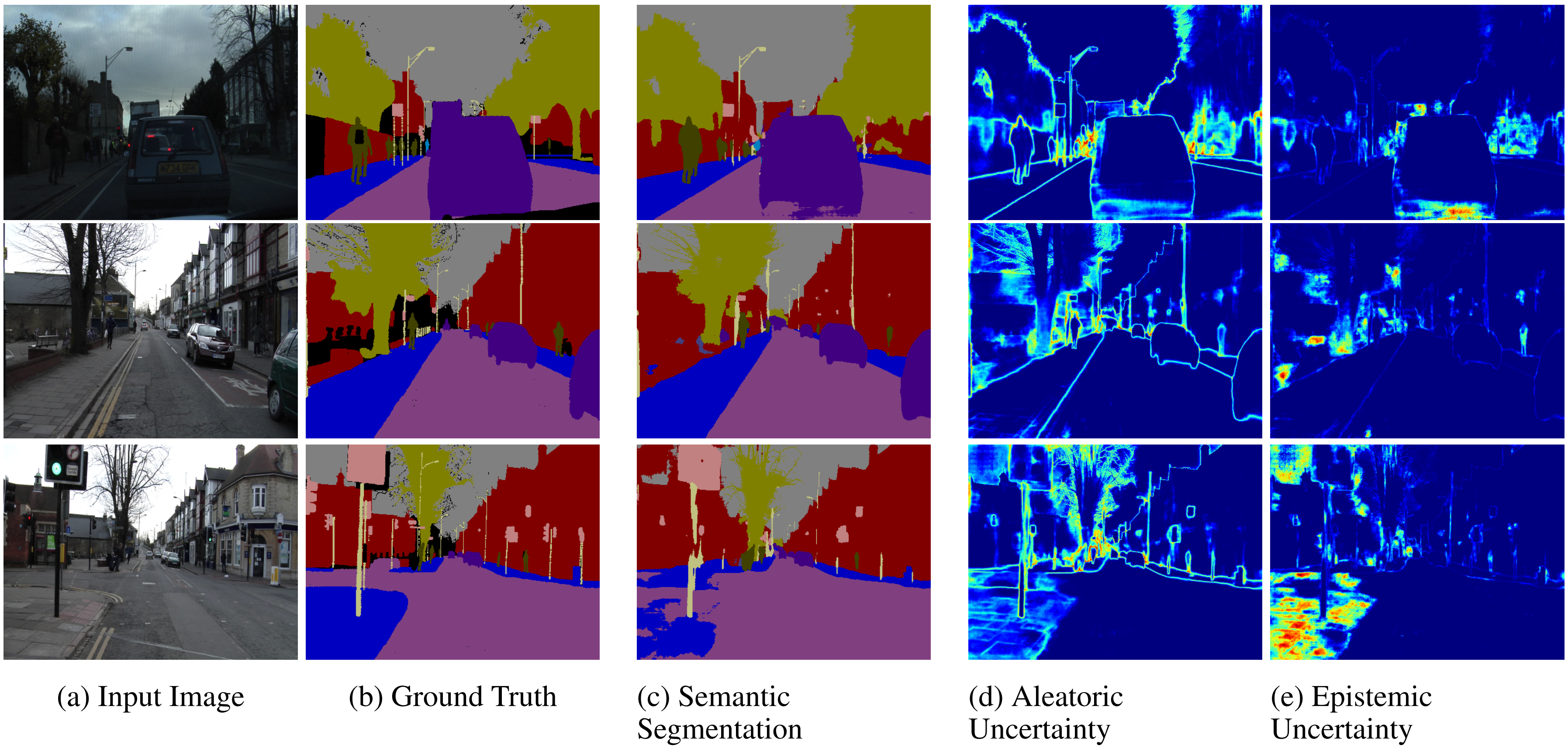
Example: Which is a Better Fit?



When the data is scarce and noisy, e.g., in medicine, and robotics.

Uncertainty for Image Segmentation

Image Truth Prediction Uncertainty



(taken from Kendall et al. 2017)

Outline of the Talk

- Uncertainty is important
  - E.g., when data are scarce, missing, unreliable etc.
- Uncertainty computation is difficult
  - Due to large model and data used in deep learning
- This talk: fast computation of uncertainty
  - Bayesian deep learning
  - Methods that are extremely easy to implement

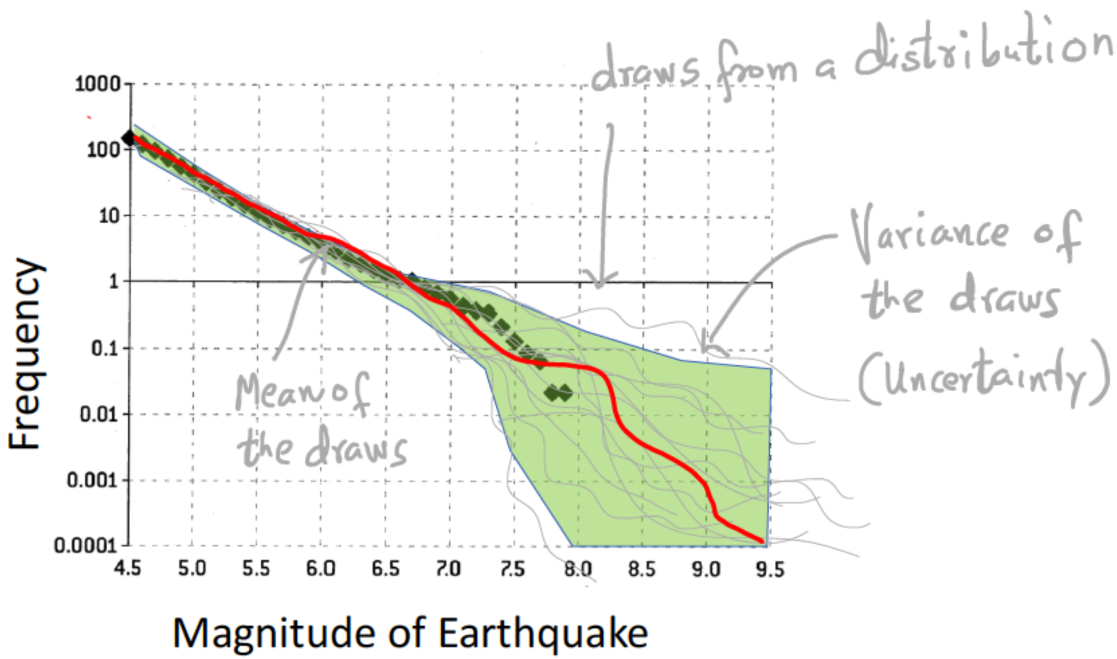
Uncertainty in Deep Learning

Why is it difficult to estimate it?

A Naïve Method

$$p(\mathcal{D}|\theta) = \prod_{i=1}^{N} p(y_i|f_\theta(x_i))$$

Data → $\mathcal{D}$; Parameters → θ; Output → y_i; Input → x_i; Neural network → f_θ



Generate

$$\theta \sim p(\theta)$$

Prior distribution

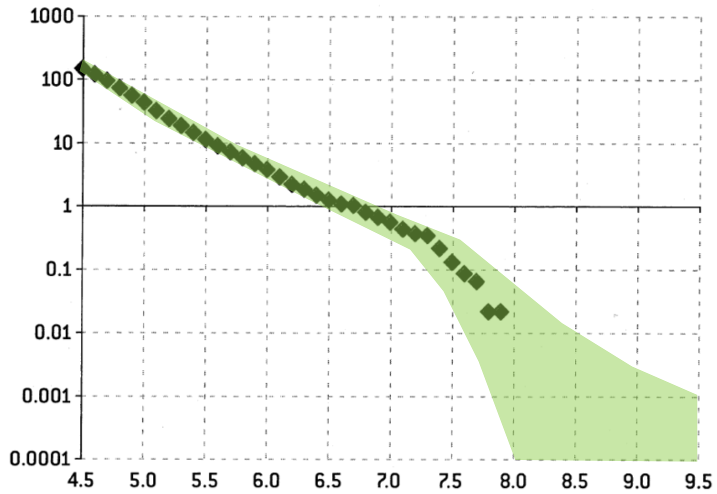
Bayesian Inference

Bayes' rule : $$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{\int p(\mathcal{D}|\theta)p(\theta)d\theta}$$

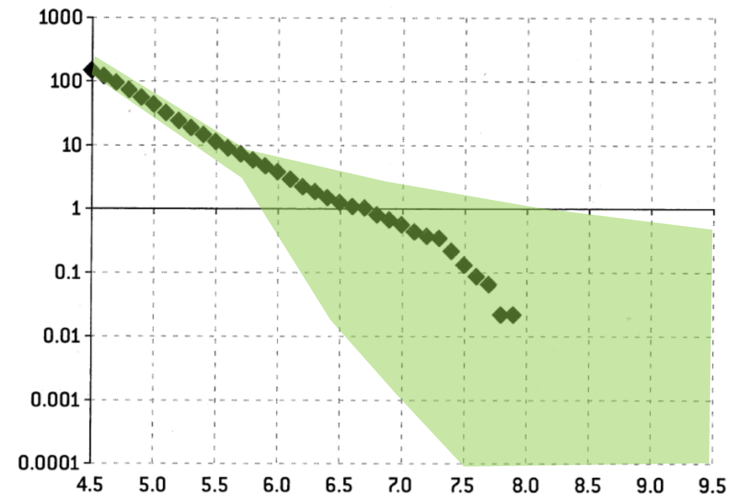
Posterior distribution

Intractable integral

Narrow



Wide



Approximate Inference with Gradients

$$p(\theta|\mathcal{D}) \approx \; q(\theta) = \mathcal{N}(\theta|\mu, \sigma^2)$$

$$\max \mathcal{L}(\mu, \sigma^2) := \mathbb{E}_q \Big[\log \frac{p(\theta)}{q(\theta)} \Big] + \sum_{i=1}^{N} \mathbb{E}_q [\log p(\mathcal{D}_i|\theta)]$$

Regularizer   Data-fit term

$$\mu \leftarrow \mu + \rho \nabla_\mu \mathcal{L}$$

$$\sigma \leftarrow \sigma + \rho \nabla_\sigma \mathcal{L}$$

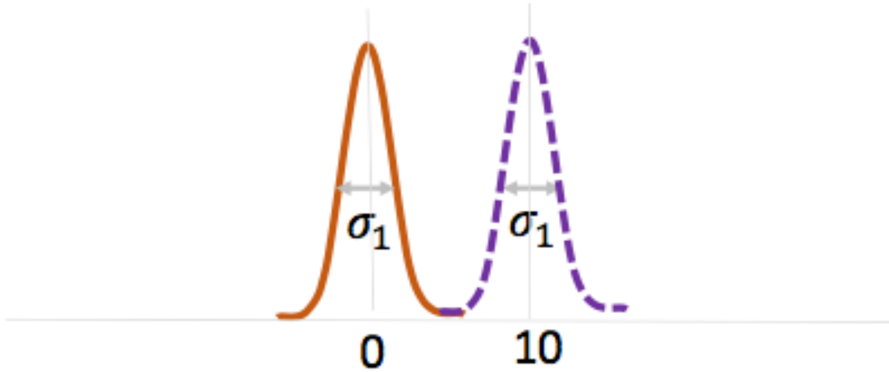
Bayes by Backprop (Blundell et al. 2015),
Practical VI (Graves et al. 2011),
Black-box VI (Rangnathan et al. 2014) etc.

Our contribution: Using natural-gradients leads to faster and simpler algorithm than gradients methods)

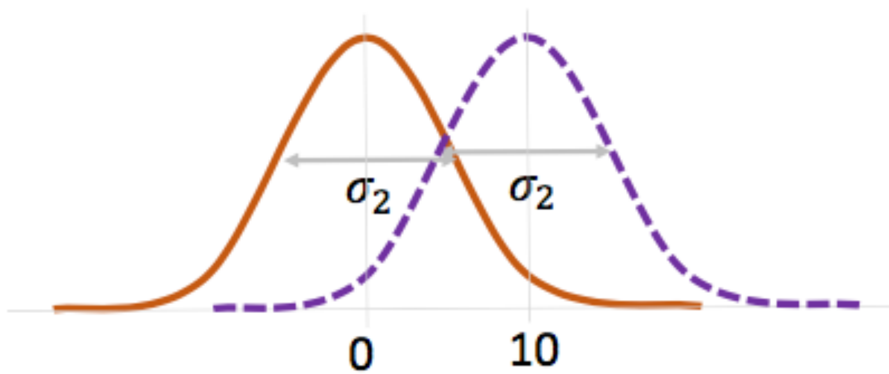
- Khan & Lin (AIstats 2017), Khan et al. (ICML 2018), Khan & Nielsen (ISITA2018)

Euclidean Distance is inappropriate!

Two Gaussians with mean 1 and 10 respectively
and variances equal to σ_1 have Euclidean distance = 10



Same as the top row but with the variance $\sigma_2 > \sigma_1$
but still Euclidean distance = 10



(Amari 1999, Sato 2001, Honkela et.al. 2010, Hoffman et.al. 2013, Khan and Lin 2017)

VI using Natural-Gradient Descent

Fisher Information Matrix (FIM)

$$F(\lambda) := \mathbb{E}_{q_\lambda}\left[\nabla \log q_\lambda(w) \nabla \log q_\lambda(w)^\top\right]$$

$$\max_\lambda \lambda^T \nabla_\lambda \mathcal{L}_t - \frac{1}{2\rho_t}(\lambda - \lambda_t)^T F(\lambda_t)(\lambda - \lambda_t)$$

$$\lambda_{t+1} = \lambda_t + \rho_t \underbrace{F(\lambda_t)^{-1} \nabla_\lambda \mathcal{L}_t}$$

Natural Gradients: $\tilde{\nabla}_\lambda \mathcal{L}_t$

Fast Computation of (Approximate) Uncertainty

Approximate by a Gaussian distribution, and find it by “perturbing” the parameters during backpropagation

Fast Computation of Uncertainty

$$\prod_{i=1}^{N} p(y_i | f_\theta(x_i)) \qquad \theta \sim \mathcal{N}(\theta | 0, I)$$

Adaptive learning-rate method (e.g., Adam)

Variational Adam (Vadam)

0. Sample ϵ from a standard normal distribution

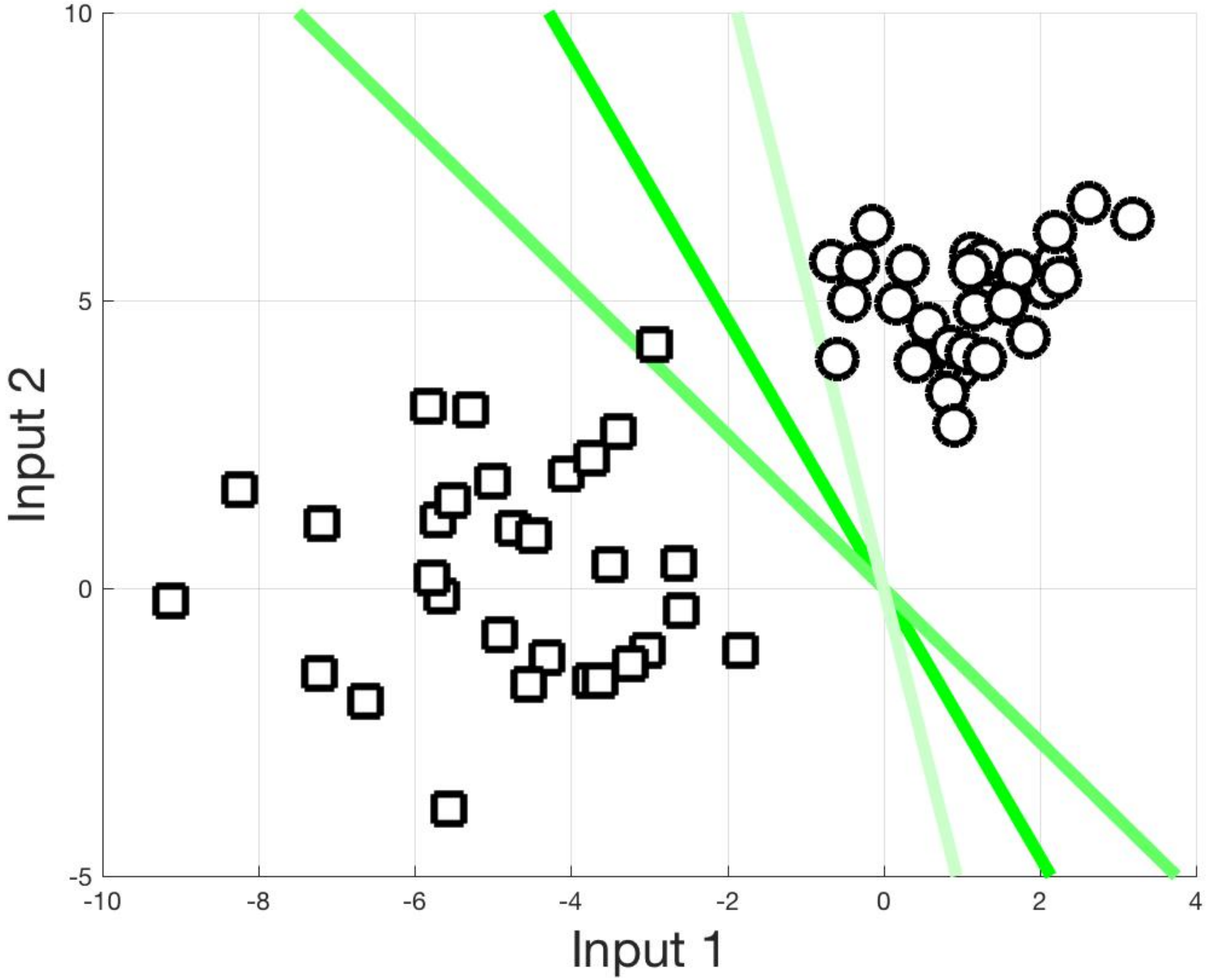
$$\theta_{\text{temp}} \leftarrow \theta + \epsilon * \sqrt{N * \text{scale} + 1}$$

1. Select a minibatch
2. Compute gradient using backpropagation
3. Compute a scale vector to adapt the learning rate
4. Take a gradient step

$$\theta \leftarrow \theta + \text{learning\_rate} * \frac{\text{gradient}}{\sqrt{\text{scale}} + 10^{-8}}$$

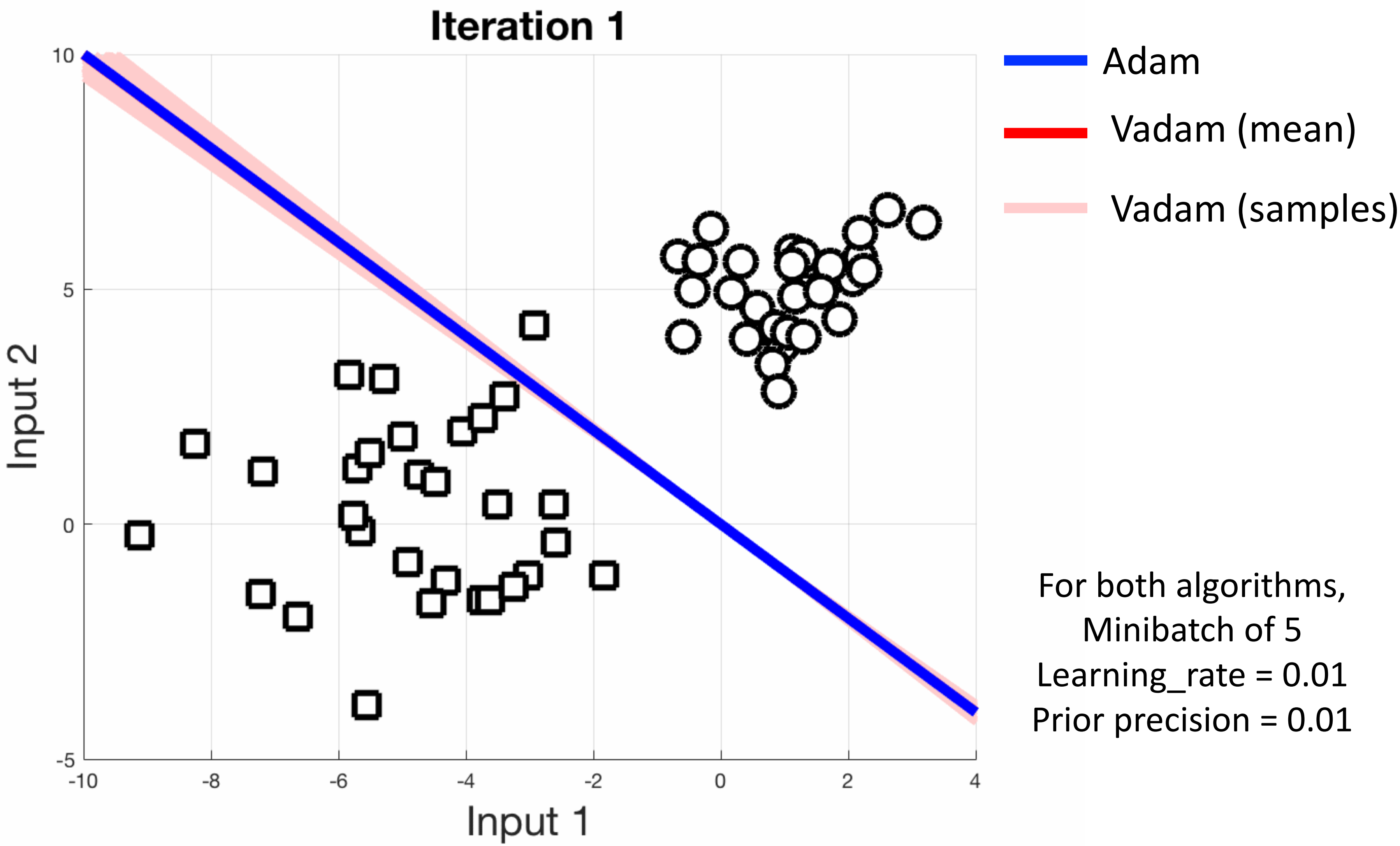
$$\theta \leftarrow \theta + \text{learning\_rate} * \frac{\text{gradient} + \theta/N}{\sqrt{\text{scale}} + 1/N}$$

Illustration: Classification



Logistic regression (30 data points, 2 dimensional input). Sampled from Gaussian mixture with 2 components

Adam vs Vadam



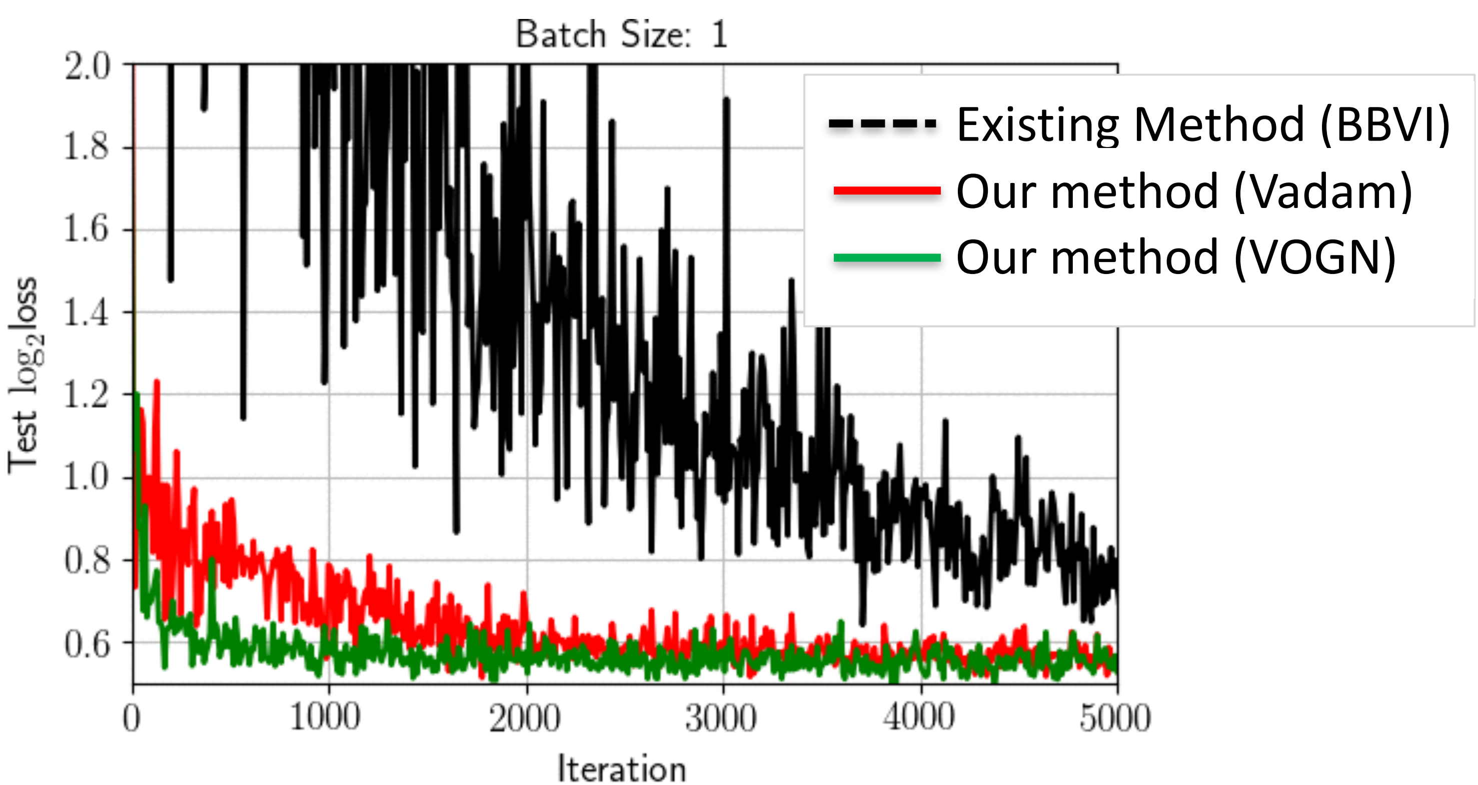
For both algorithms,
Minibatch of 5
Learning_rate = 0.01
Prior precision = 0.01

Why does this work?

- This algorithm is obtained by replacing "gradients" by "natural gradients" (using information geometry)
  - See our ICML 2018 paper.
  - The scaling in natural gradient is related to the scaling in Newton method.
  - Our method is a more principled approach than the Bayesian dropout (Gal and Gharhamani, 2016).
  - Some caveats: Choose small minibatches, better results are obtained with VOGN.

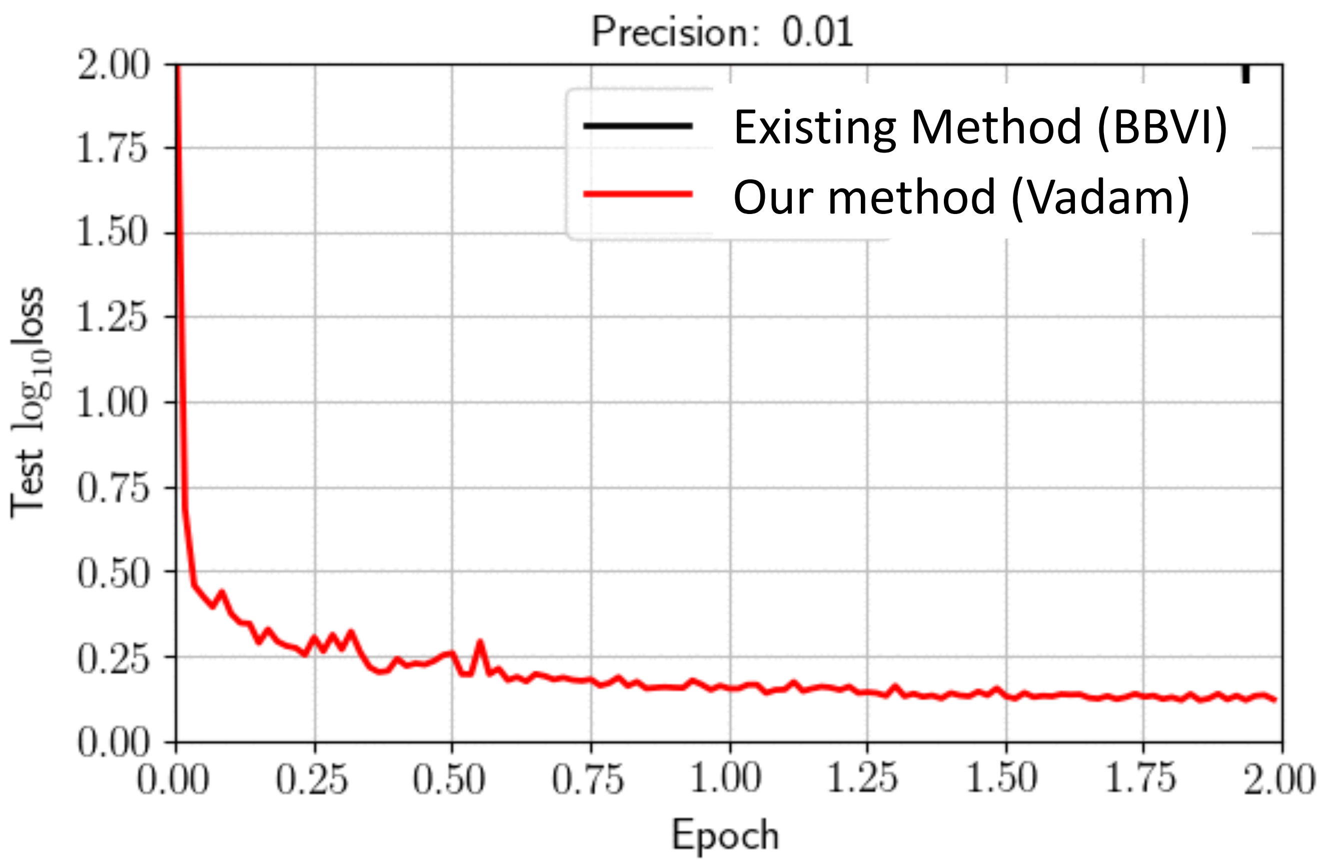
Faster, Simpler, and More Robust

Regression on Australian-Scale dataset using deep neural nets for various number of minibatch size.



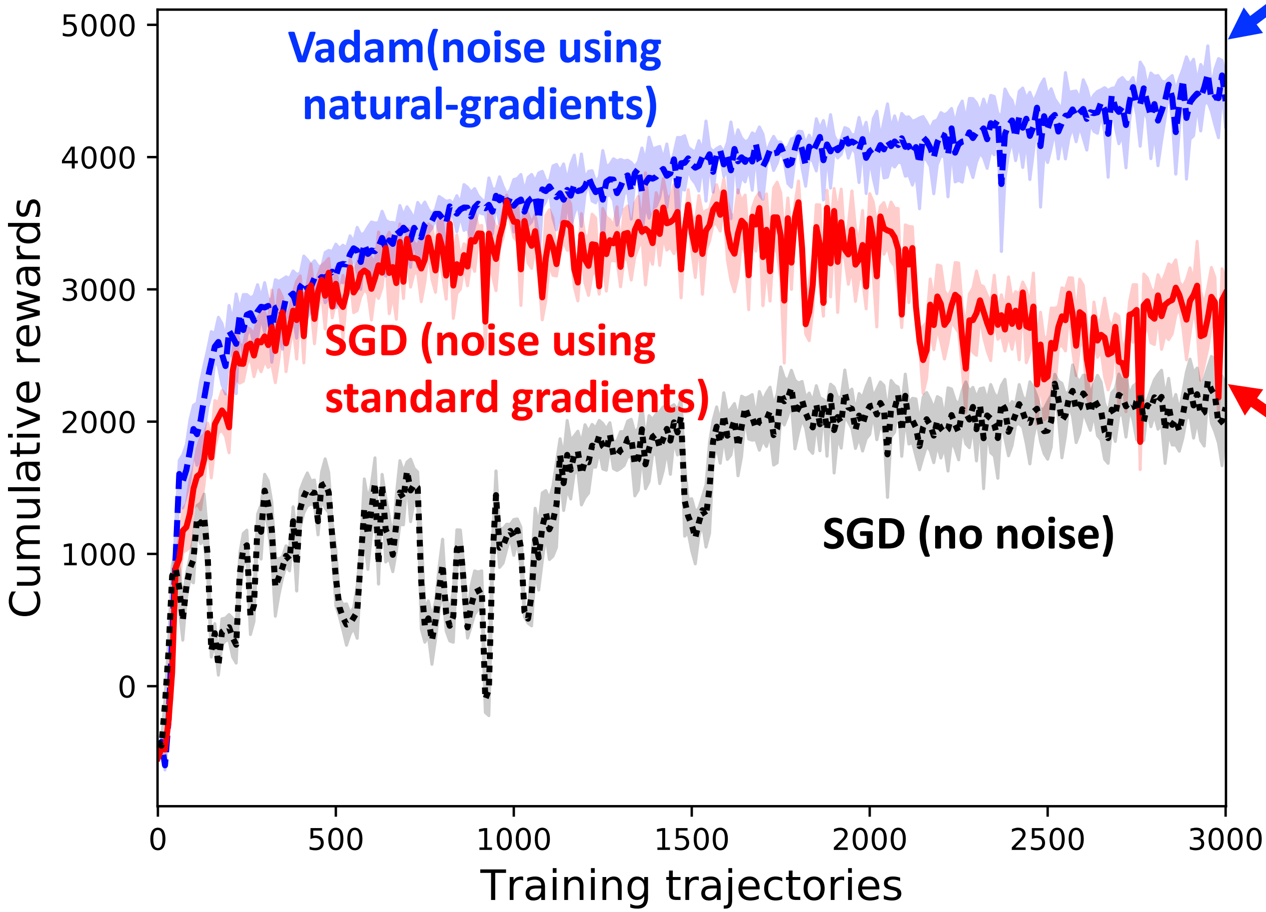
Faster, Simpler, and More Robust

Results on MNIST digit classification (for various values of Gaussian prior precision parameter λ)

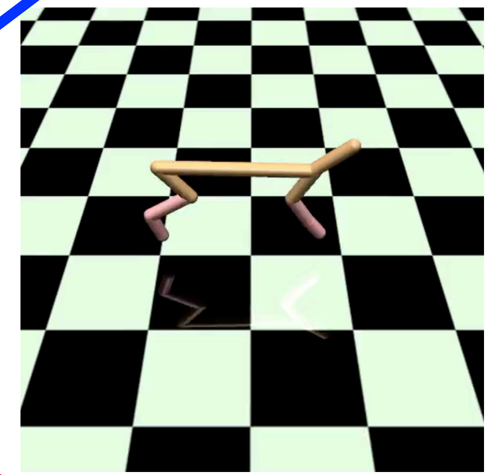


Parameter-Space Noise for Deep RL

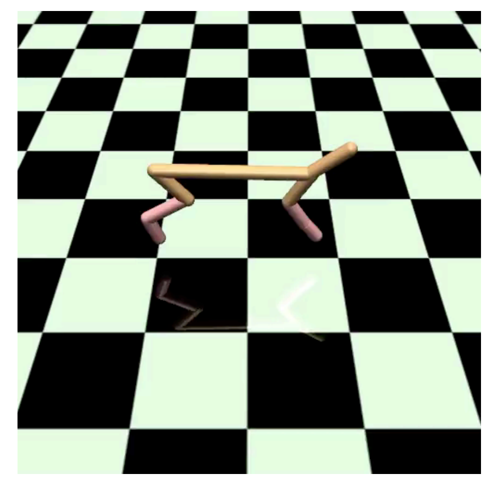
On OpenAI Gym Cheetah with DDPG
with DNN with [400,300] ReLU



Reward 5264



Reward 2038

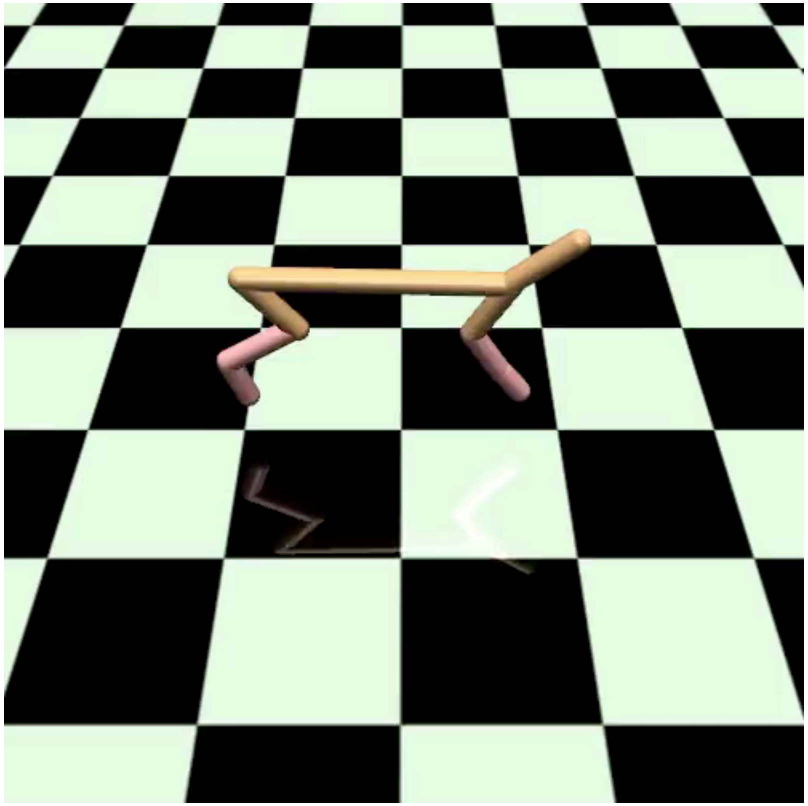


Ruckstriesh et.al.2010, Fortunato et.al. 2017, Plapper et.al. 2017

Deep Reinforcement Learning

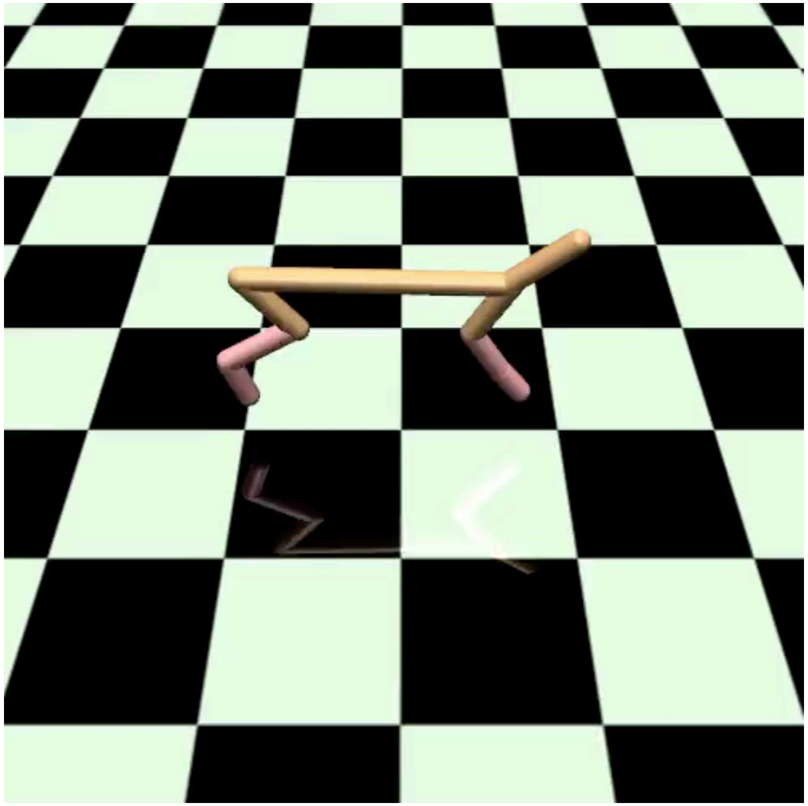
No Exploration (SGD)

Reward = 2860

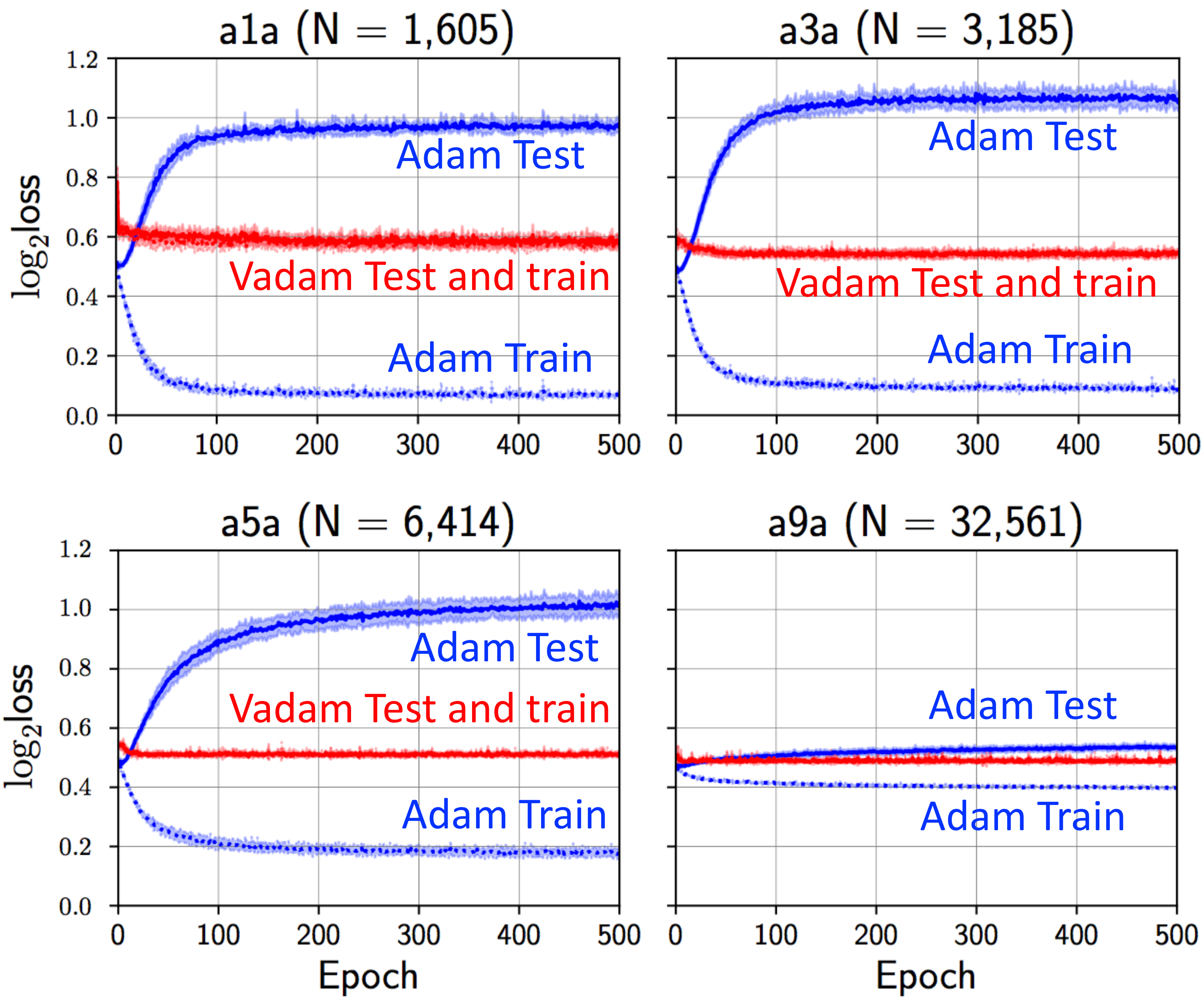


Exploration using Vadam

Reward = 5264

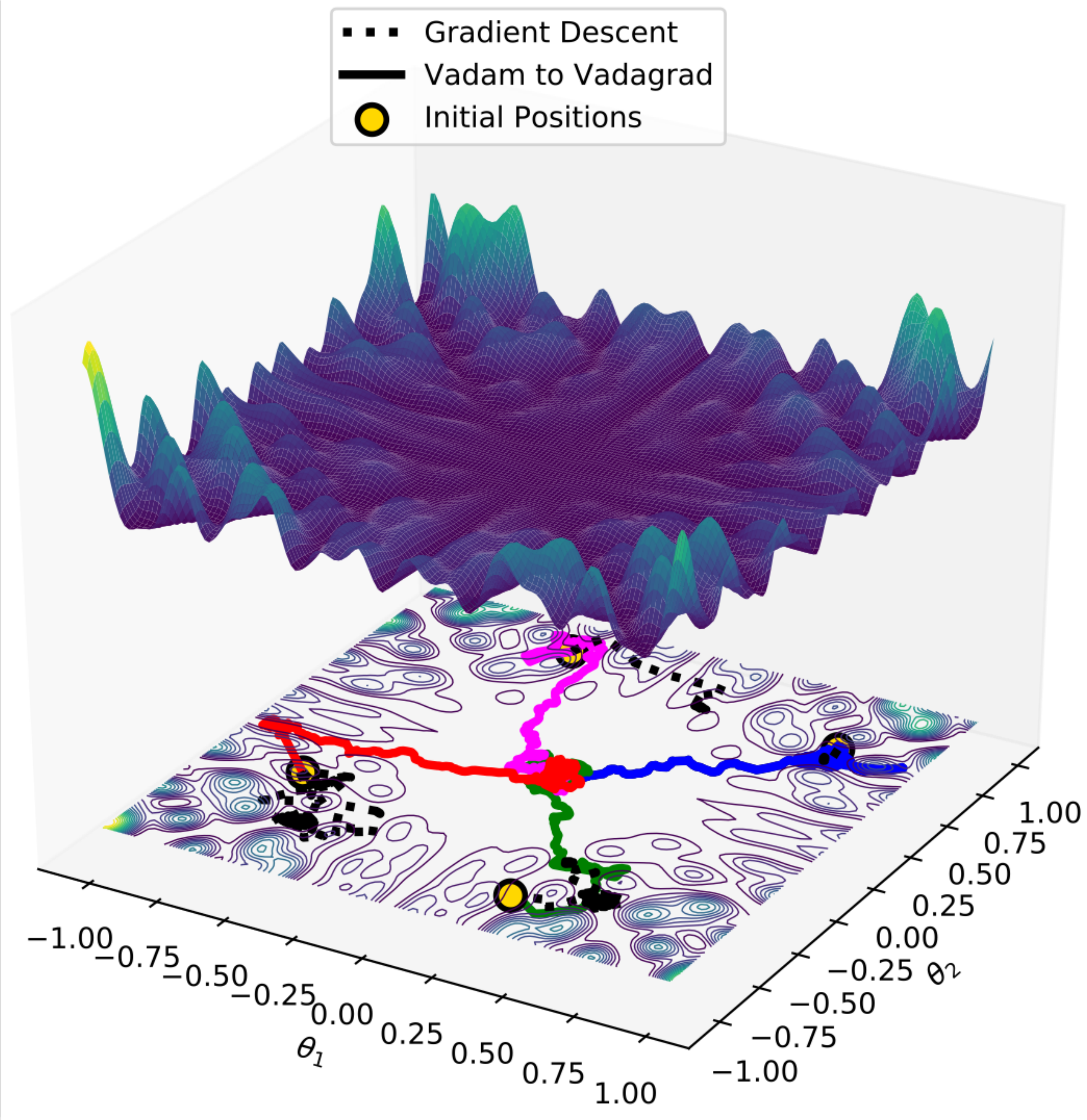


Reduce Overfitting with Vadam



Vadam shows consistent train-test performance, while Adam overfits when N is small

BNN classification on a1a - a9a datasets



Avoiding Local Minima

An example taken from Casella and Robert's book.

Vadam reaches the flat minima, but GD gets stuck at a local minima.

Optimization by smoothing, Gaussian homotopy/blurring etc., Entropy SGLD etc.

Summary

- Uncertainty is important, especially when the data is scarce, missing, unreliable etc.
- We can obtain uncertainty cheaply with very little effort
  - Bayesian deep learning
- It works reasonably well on our benchmarks.

Open Questions

- Extensions to other types of distributions
- Quality and usefulness of uncertainty
  - Multiple local minima make it difficult to establish
- Estimating various types of uncertainty
  - Model uncertainty vs data uncertainty
  - Applications play a big role here
- Application to active learning, reinforcement learning, continual learning

References

https://emtiyaz.github.io

Fast yet Simple Natural-Gradient Descent for Variational Inference in Complex Models,
INVITED PAPER AT (ISITA 2018) **M.E. KHAN** and D. NIELSEN, [Pre-print]

Fast and Scalable Bayesian Deep Learning by Weight-Perturbation in Adam,
(ICML 2018) **M.E. KHAN**, D. NIELSEN, V. TANGKARATT, W. LIN, Y. GAL, AND A. SRIVASTAVA, [ArXiv Version] [Code]

Conjugate-Computation Variational Inference : Converting Variational Inference in Non-Conjugate Models to Inferences in Conjugate Models,
(AISTATS 2017) **M.E. KHAN** AND W. LIN [Paper] [Code for Logistic Reg + GPs] [Code for Correlated Topic Model]

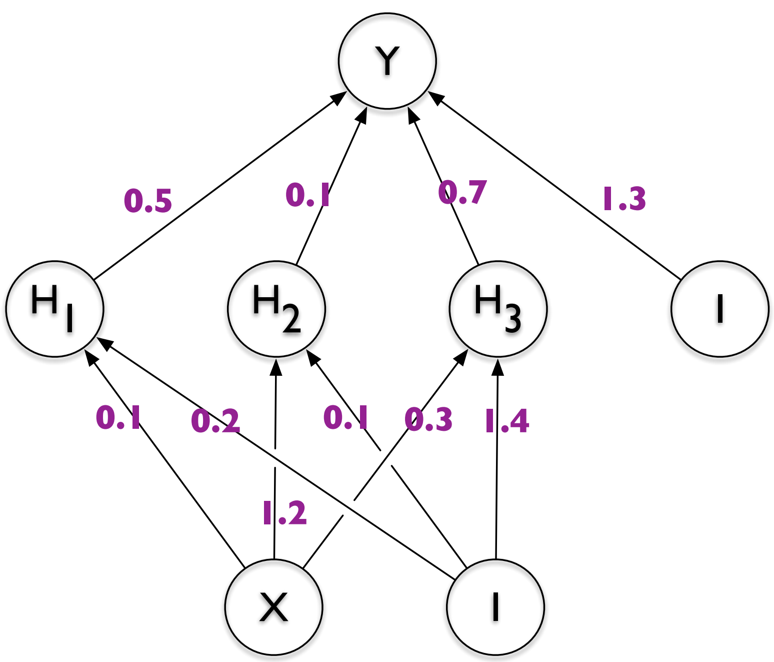
Thanks!

Slides, papers, and code available at

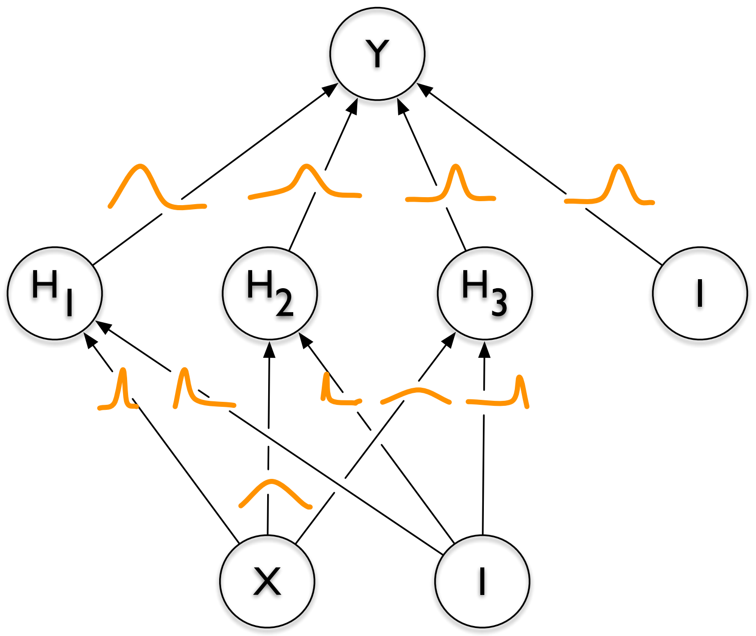
https://emtiyaz.github.io

Bayesian Deep Learning

Deep Learning

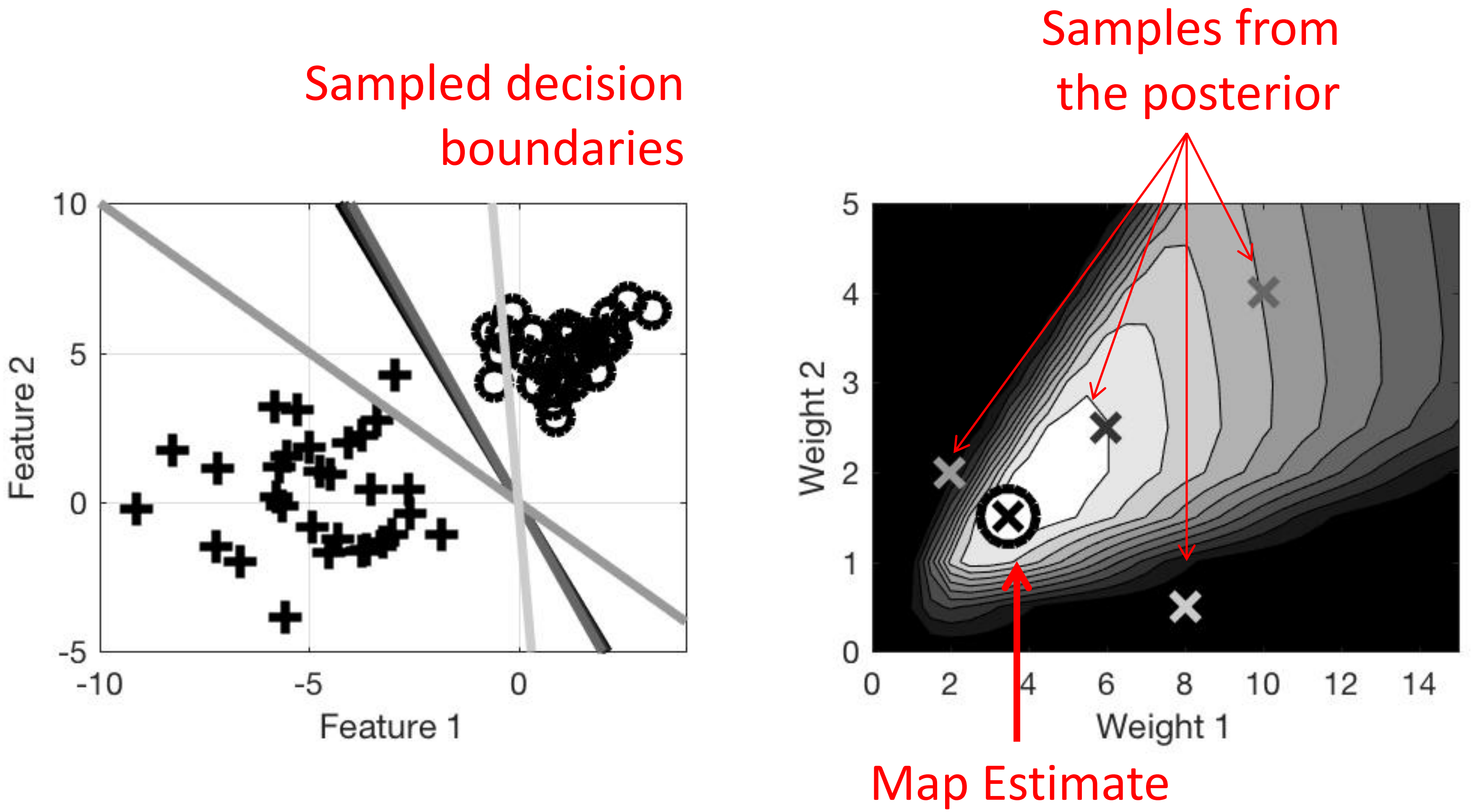


Bayesian Deep Learning



(taken from Blundell et al. 2015)

Bayesian Inference for Classification



RMSprop vs Vprop