Learning-Algorithms from Bayesian Principles

Mohammad Emtiyaz Khan
RIKEN Center for AI Project, Tokyo
http://emtiyaz.github.io

The Goal of My Research

"To understand the fundamental principles of learning from data and use them to develop algorithms that can learn like living beings."

Human Learning:

At the age of 6 months, learning by actively and sequentially collecting limited and correlated data.

Converged
at the age
of
12 months

Transfer Knowledge at the age of 14 months

Human learning $\neq$ **Deep learning**

Humans can learn from limited, sequential, correlated data, with a clear understanding of the world.

Machines require large amount of IID data, and don't really understand the world and cannot reason about it.

My current research focuses on reducing this gap!

Learning-Algorithms from Bayesian Principles

- Practical Bayesian principles
  - To design/improve/generalize learning-algorithms.
  - By computing “posterior” distribution over unknowns.
- Generalization of many existing algorithms,
  - Classical (least-squares, Newton, HMM, Kalman.. etc).
  - Deep Learning (SGD, RMSprop, Adam).
- Helps us design new algorithms
  - Connection to Gaussian Processes.
  - Reinforcement, online, continual learning, reasoning..
- Impact: Everything with one common principle.

Learning-Algorithms by Bayesian Principles

Learning by optimization: $\theta \leftarrow \theta - \rho H^{-1} \nabla_\theta \ell(\theta)$

Learning by Bayes: $\lambda \leftarrow (1-\rho)\lambda - \rho \nabla_\mu \mathbb{E}_q [\ell(\theta)]$

Natural and Expectation parameters of q

e.g., Gaussian distribution

$q(\theta) := \mathcal{N}(\theta | m, V)$

Natural parameters $\{V^{-1}m, V^{-1}\}$

Expectation/moment/ mean parameters $\{\mathbb{E}(\theta), \mathbb{E}(\theta\theta^\top)\}$

$\exp\left[m^\top V^{-1}\theta - \frac{1}{2}\theta^\top V^{-1}\theta\right]$

Learning by Bayes

Learning by optimization: $\theta \leftarrow \theta - \rho H^{-1} \nabla_\theta \ell(\theta)$

Learning by Bayes: $\lambda \leftarrow (1-\rho)\lambda - \rho \nabla_\mu \mathbb{E}_q \left[\ell(\theta)\right]$

Natural and Expectation parameters of q

AIstats 2017
ICML 2017
- Classical algorithms: Least-squares, Newton's method, Kalman filters, Baum-Welch, Forward-backward, etc.
- Bayesian inference: EM, Laplace's method, SVI, VMP.

ICML 2018
NeurIPS 2018
ISITA 2018
ICLR 2018
- Deep learning: SGD, RMSprop, Adam.
- Reinforcement learning: parameter-space exploration, natural policy-search.
- Continual learning: Elastic-weight consolidation.
- Online learning: Exponential-weight average.

NIPS 2017
- Global optimization: Natural evolutionary strategies, Gaussian homotopy, continuation method & smoothed optimization.
- List incomplete…

$q_\lambda(\theta) := \mathcal{N}(m, V)$ Least Squares

$$\lambda \leftarrow (1-\rho)\lambda - \rho \nabla_\mu \mathbb{E}_q\left[\ell(\theta)\right] \quad \Rightarrow \lambda_* = \nabla_{\mu_*} \mathbb{E}_{q_*}\left[\ell(\theta)\right]$$

likelihood prior

$$\mathbb{E}_q\Big[\underbrace{(y - X\theta)^\top (y - X\theta)}_{} + \gamma \theta^\top \theta\Big] := \ell(\theta)$$

$$-\mathbb{E}_{q_\lambda}[\theta]^\top X^\top y + \operatorname{trace}\Big[X^\top X \mathbb{E}_{q_\lambda}[\theta\theta^\top]\Big]$$

$$\nabla_{\mathbb{E}_{q_\lambda}[\theta]} = \begin{pmatrix} -X^\top y & + & 0 \end{pmatrix} = V^{-1} m$$

$$\nabla_{\mathbb{E}_{q_\lambda}[\theta\theta^\top]} = \begin{pmatrix} X^\top X & + & \gamma I \end{pmatrix} = V^{-1}$$

Expectation params

$$\left[X^\top X + \gamma I\right]^{-1} X^\top y$$

Neural Network

$$\mathbb{E}_q \left(\sum_{i=1}^{N} \ell(y_i, f_\theta(x_i)) + \gamma \theta^\top \theta \right)$$

likelihood

prior

neural network

$$(X^\top X + \gamma I)^{-1} X^\top y$$

$$m \leftarrow m - \rho (S + \gamma I)^{-1} g$$

$$S \leftarrow (1 - \rho) S + \rho H$$

Hessian

Gradient

RMSprop

$$\theta \leftarrow \mu$$

$$g \leftarrow \frac{1}{M} \sum_i \nabla_\theta \log p(\mathcal{D}_i | \theta)$$

$$s \leftarrow (1 - \beta) s + \beta g^2$$

$$\mu \leftarrow \mu + \alpha \frac{g}{\sqrt{s} + \delta}$$

Bayes with diagonal Gaussian

$$\theta \leftarrow \mu + \epsilon, \text{ where } \epsilon \sim \mathcal{N}(0, Ns + \lambda)$$

$$g \leftarrow \frac{1}{M} \sum_i \nabla_\theta \log p(\mathcal{D}_i | \theta)$$

$$s \leftarrow (1 - \beta) s + \beta \frac{1}{M} \sum_i \nabla^2_{\theta\theta} \log p(\mathcal{D}_i | \theta)$$

$$\mu \leftarrow \mu + \alpha \frac{g + \lambda \mu / N}{s + \lambda / N}$$

Learning by Bayes

- Learning by Bayes works for
  - (Minimal) ExpFam (AIstats 2017).
  - Some mixtures of ExtFam (see ICML 2019).
  - Kernel exponential family (Upcoming).
- The principle is to choose an appropriate sufficient statistics, which then yields an approximation of the loss.

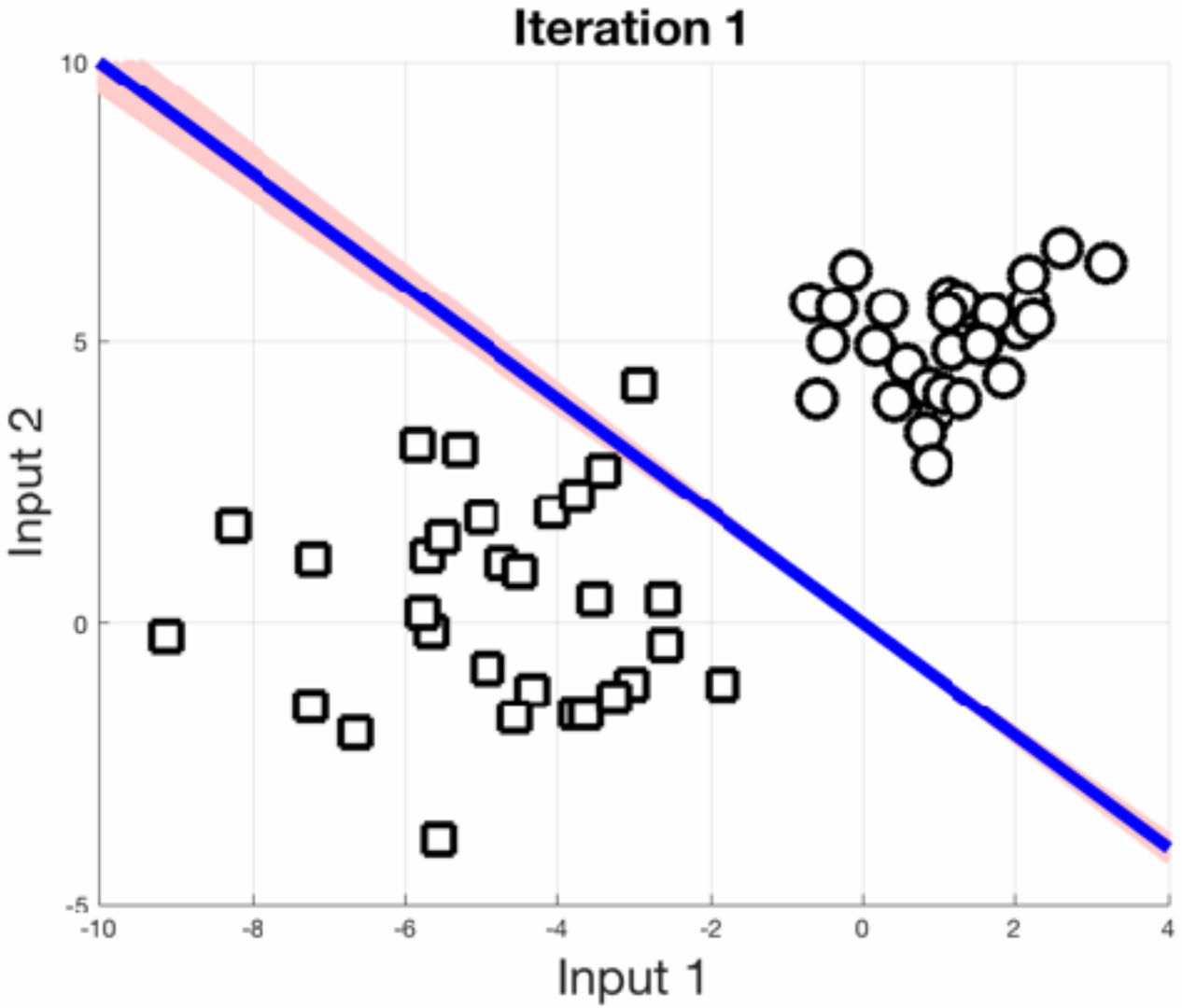
$$\lambda \leftarrow (1-\rho)\lambda - \rho \nabla_{\mu} \mathbb{E}_q \left[\ell(\theta)\right]$$

$$\ell(\theta) \approx a^\top \phi(\theta)$$

$$a := \nabla_{\mu} \mathbb{E}_q[\ell(\theta)]$$

Uncertainty for Logistic-Regression

ICML 2018



Adam

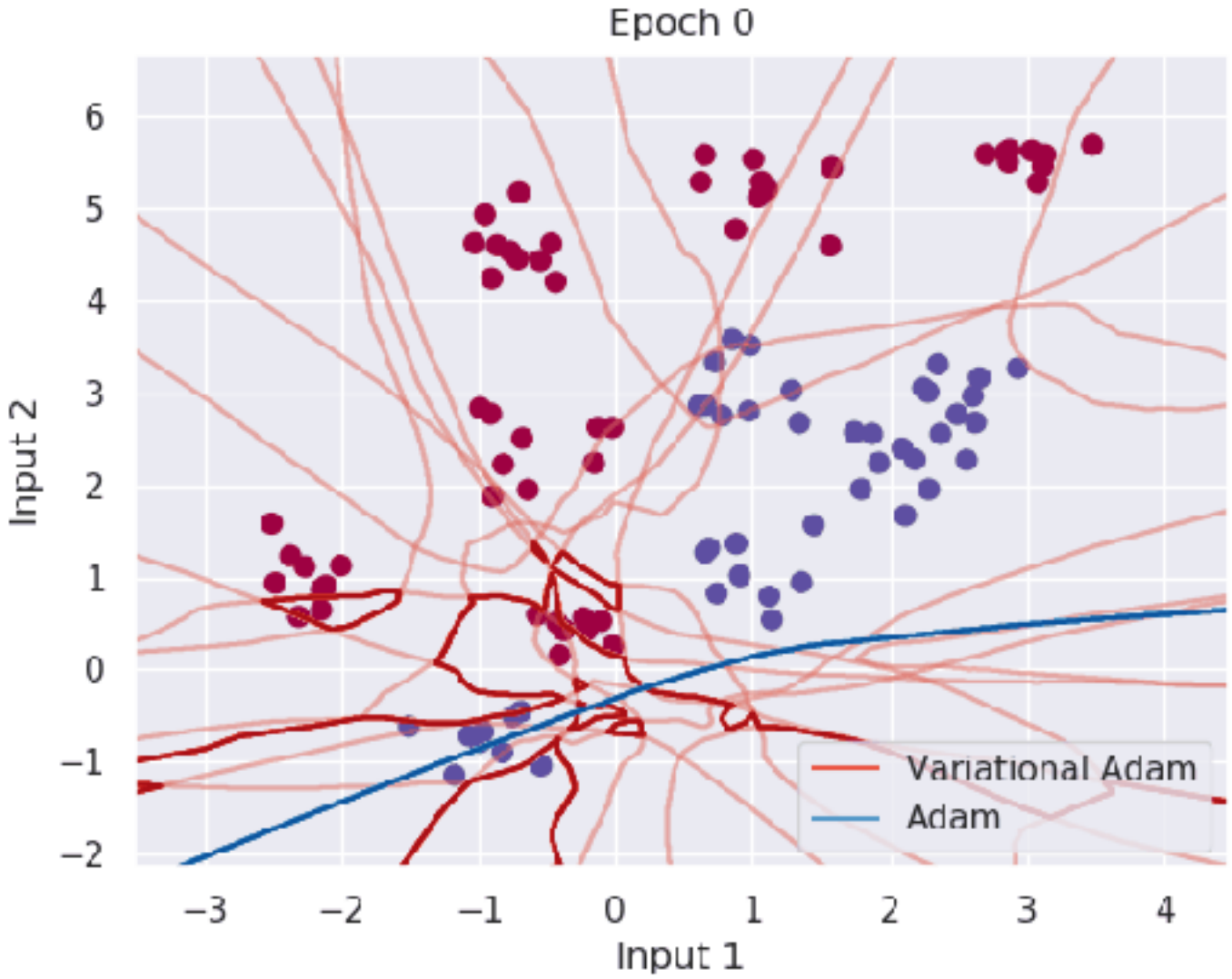
Our method (mean)

Our method (samples)

M = 5,
Rho = 0.01,
Gamma = 0.01

Uncertainty for Deep Learning

ICML 2018



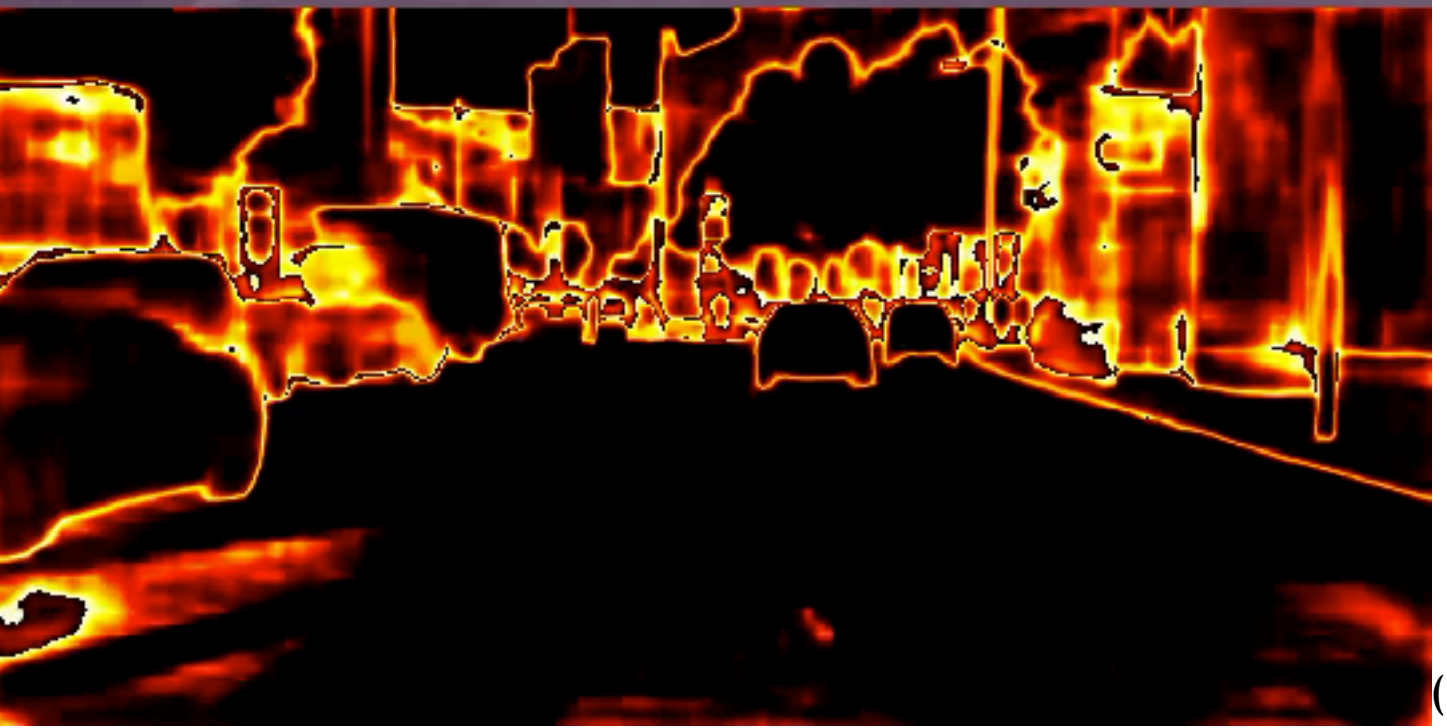
Adam

Ours (mean)

Ours (samples)

(By Runa E.)

Image Segmentation



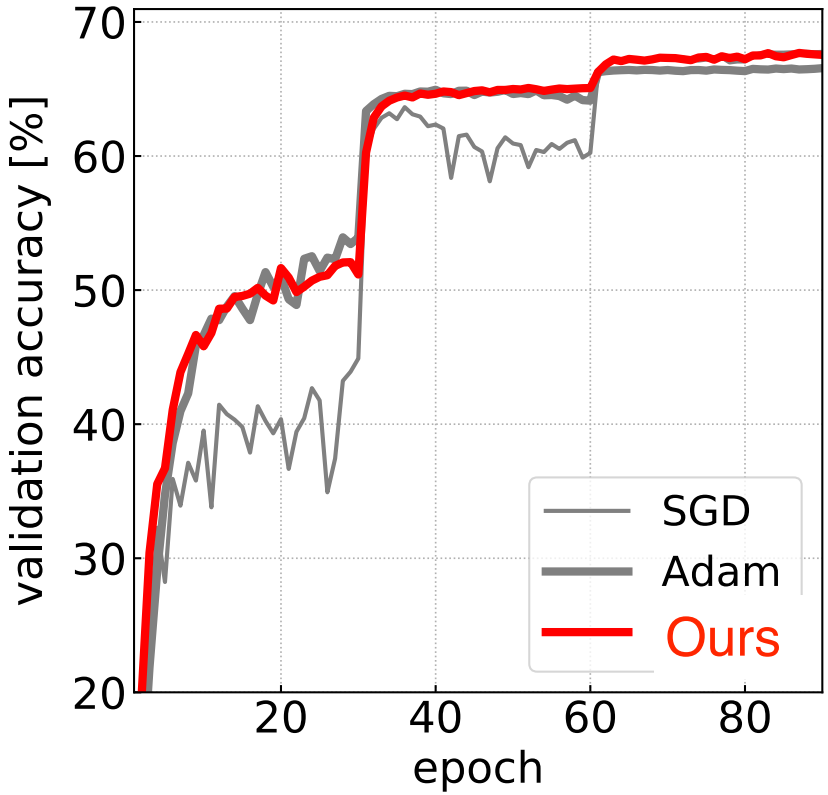
Uncertainty (entropy of class probs)

(By Roman Bachmann)

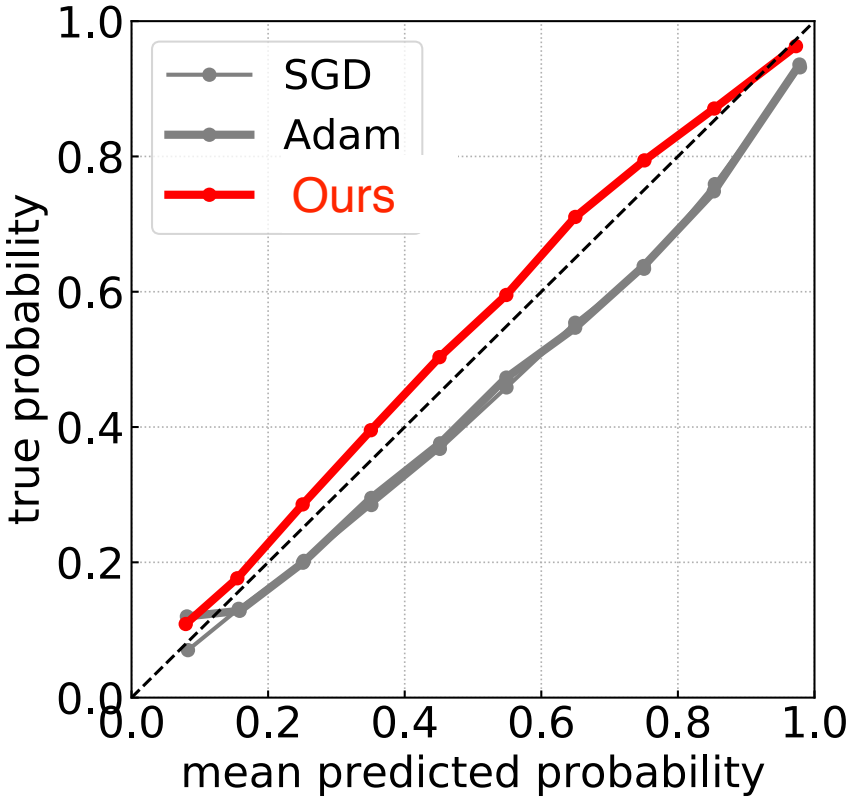
Practical DL with Bayes (on ImageNet)

State-of-the-art performance and convergence rate, while preserving benefits of Bayesian principles (“well-calibrated” uncertainty).

Accuracy



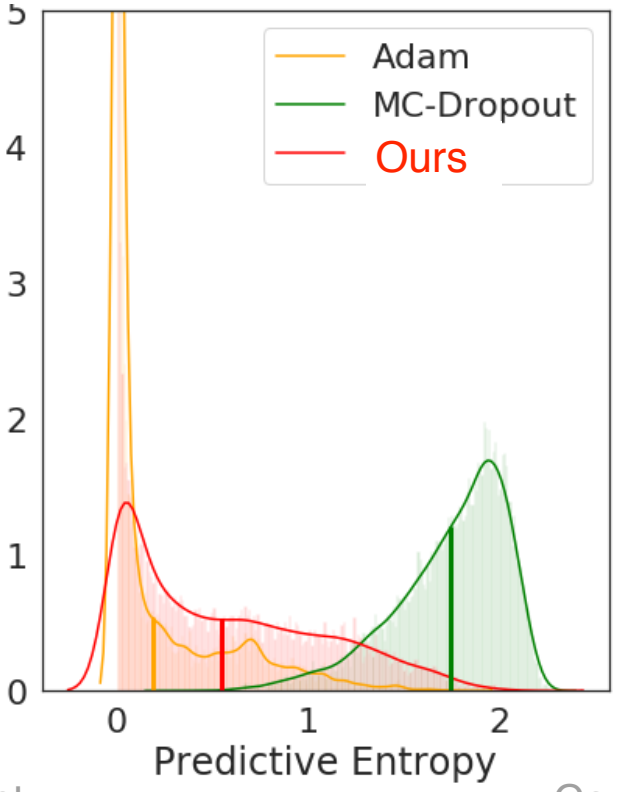
Uncertainty Calibration



Out-of-Distributions Test

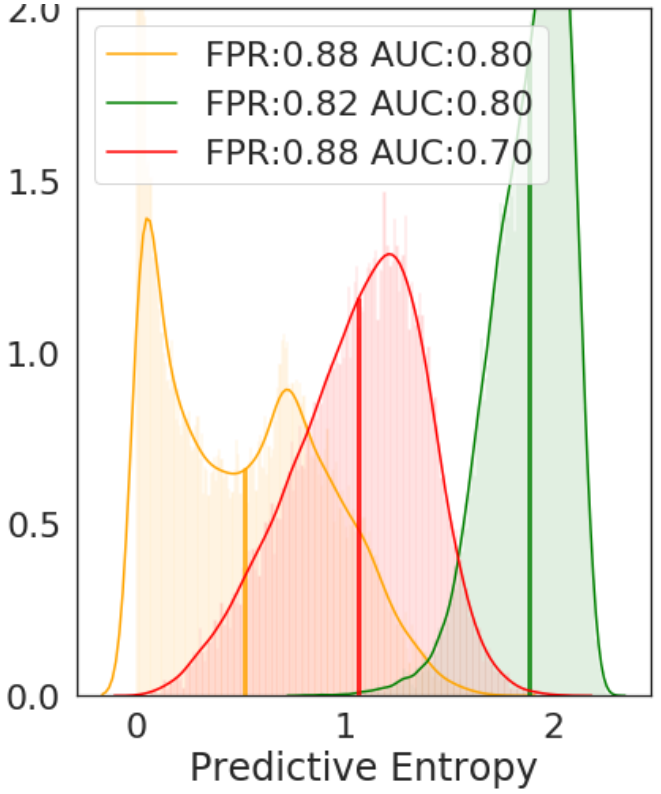
Our method (in red) is confident on “in-distribution” data, and not overconfident on “out-of-distribution” data.

In-Distribution (CIFAR-10)



More Confident

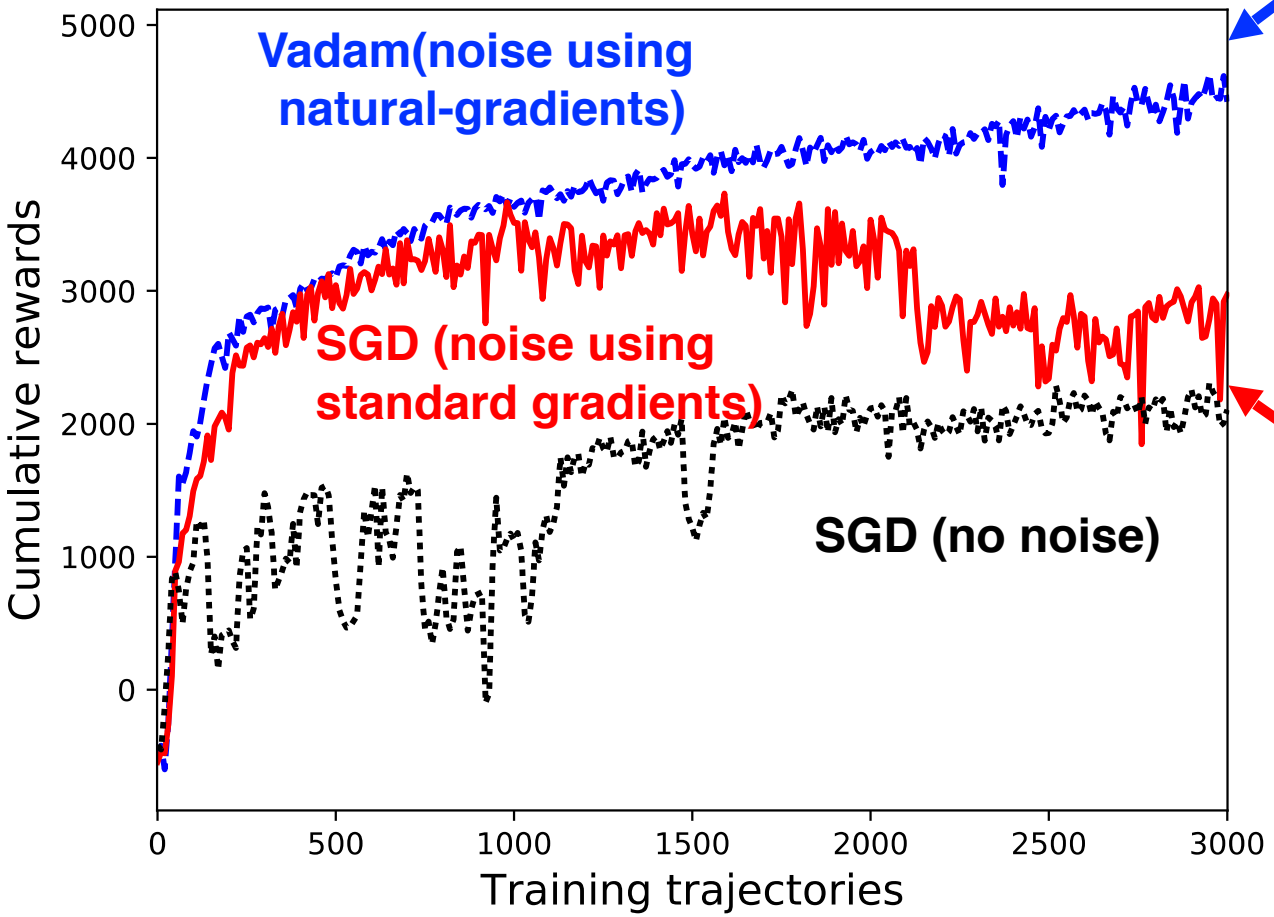
Out-of-Distribution (SVHN)



Less Confident

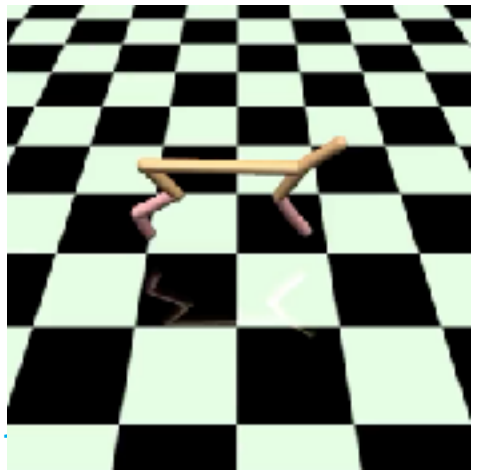
Deep Reinforcement Learning

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. 20

NN Training as Inference in LinReg

$$(X^\top X + \gamma I)^{-1} X^\top y$$

$$\mathbb{E}_q \left(\sum_{i=1}^{N} \ell(y_i, f_\theta(x_i)) + \gamma \theta^\top \theta \right)$$

likelihood

prior

neural network

$$m \leftarrow m - \rho (S + \gamma I)^{-1} g$$

$$S \leftarrow (1 - \rho) S + \rho H$$

Hessian

Gradient

Hessian of a linear model

$$\sum_{i=1}^{N} x_i \frac{1}{\sigma^2} x_i^T + \gamma I$$

Hessian of the neural-network loss

$$\sum_{i=1}^{N} J_i^\top \Lambda_i J_i + \cancel{\nabla^2_{ww} f_\theta(x_i) r_i} + \gamma I$$

Jacobian

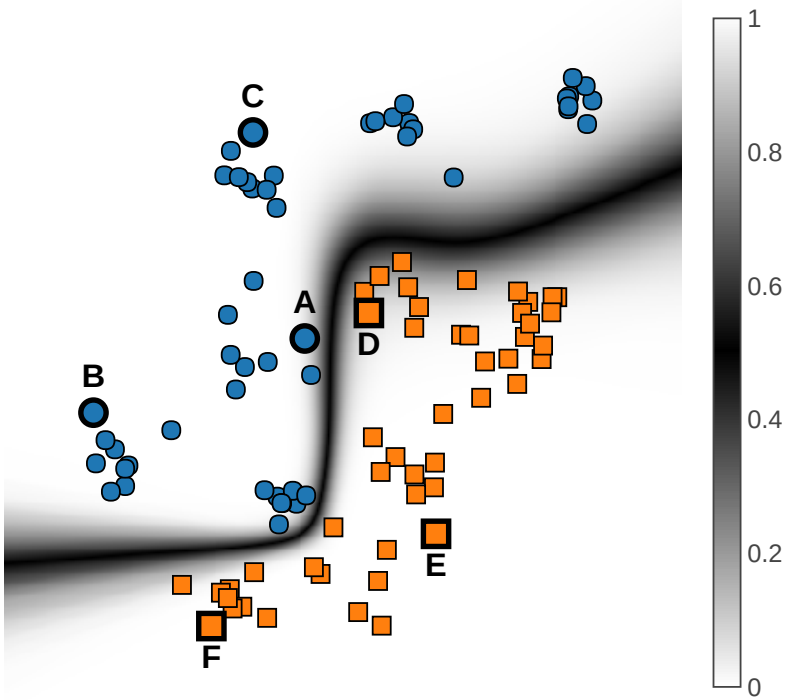
1st & 2nd derivative of the loss wrt output of the last layer

Gaussian approx on NN == Posterior of a linear model

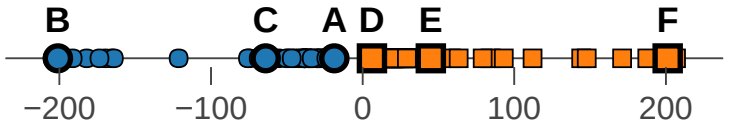
$$y_i \leftarrow f_w(x_i)$$

$$\tilde{y}_i = J_i w + \epsilon_i$$

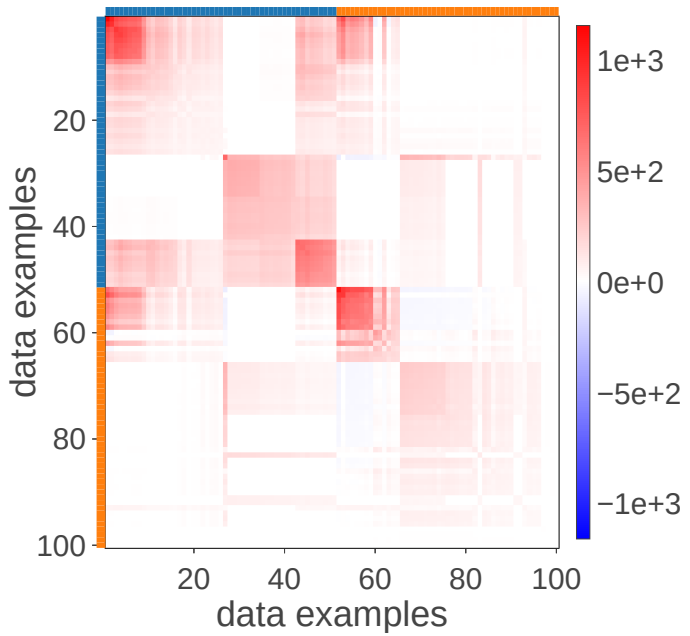
NN Training as GP Inference

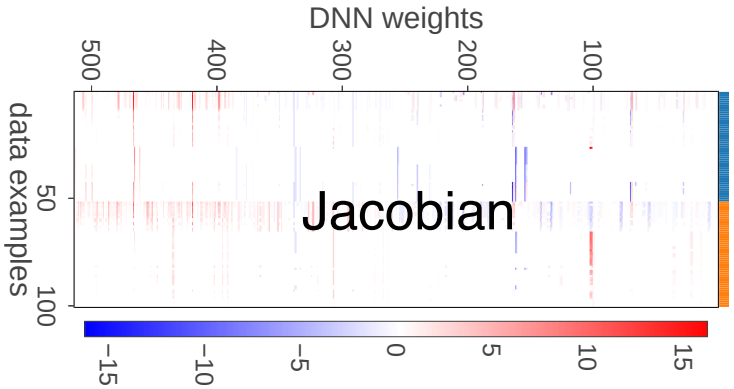


GP Observations $\tilde{y}_i$



GP Kernel (NTK) $J_i J_i^\top$

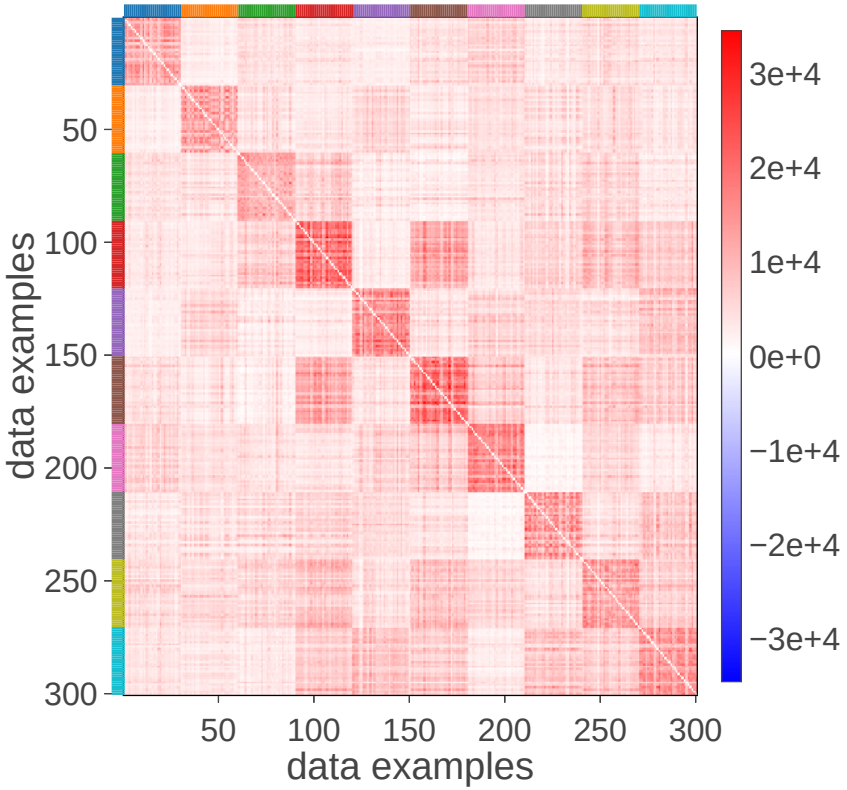




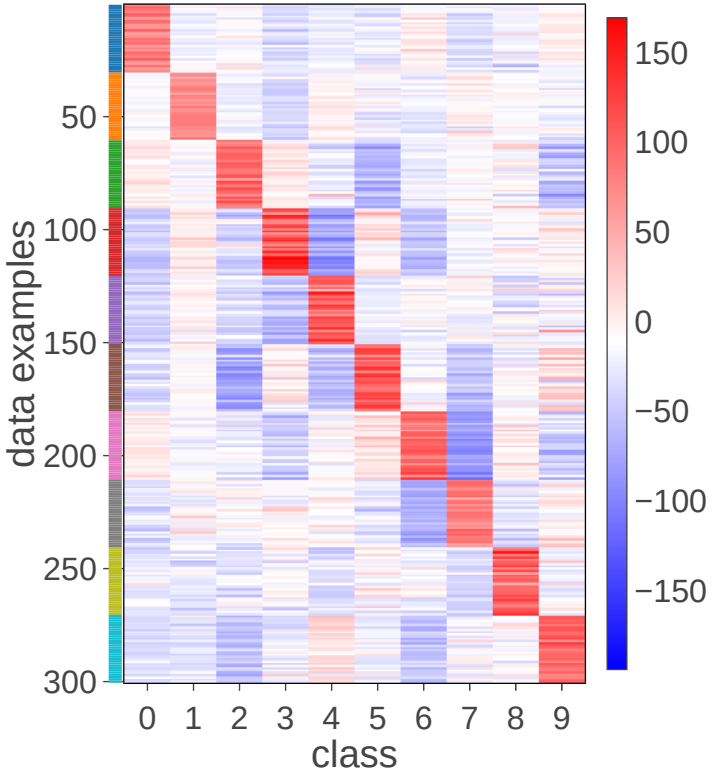
MNIST: Similarity of Examples

Kernels reveals the similarities learned by the NN.
Observations are assigned higher values for correct classes.

GP Kernel



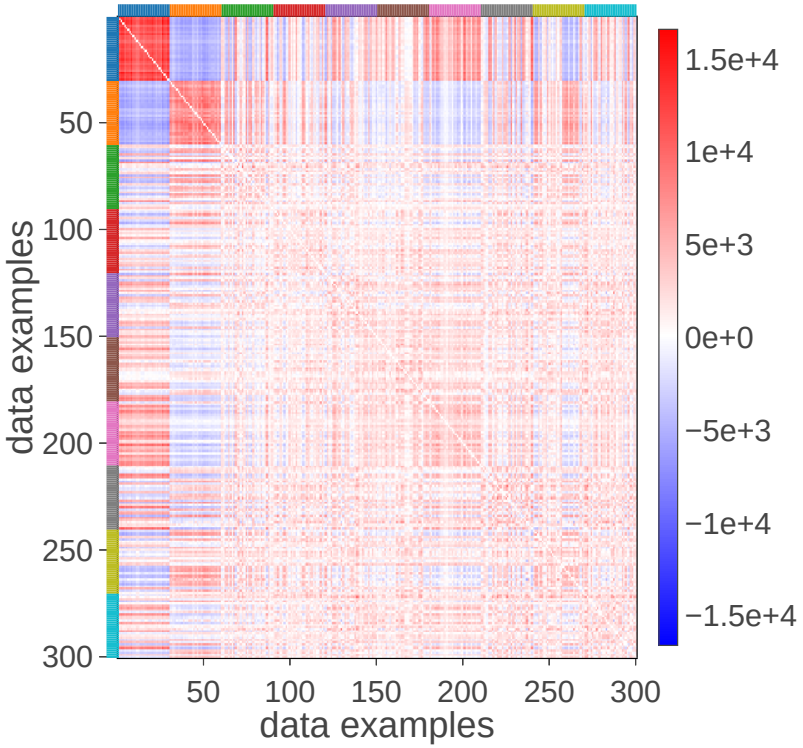
GP observations



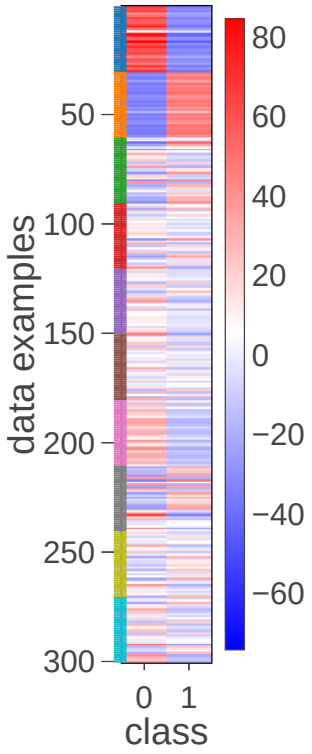
MNIST with only 0 and 1 Digits

When trained only on 0 &1digits, we see similar patterns for 0 and 1.
NN hasn't learned meaningful similarities for out-of-training classes.

GP Kernel



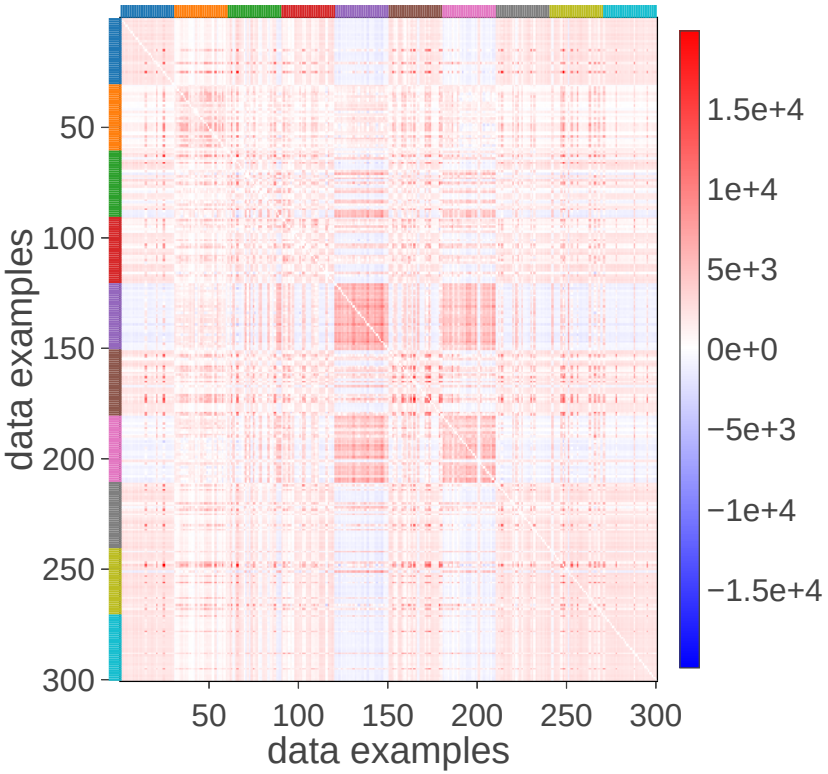
GP observations



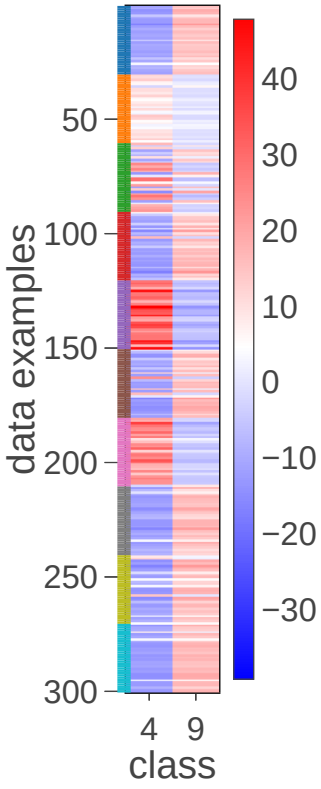
MNIST with only 4 and 9 Digits

For harder tasks, the network is learns spurious correlations.

GP Kernel



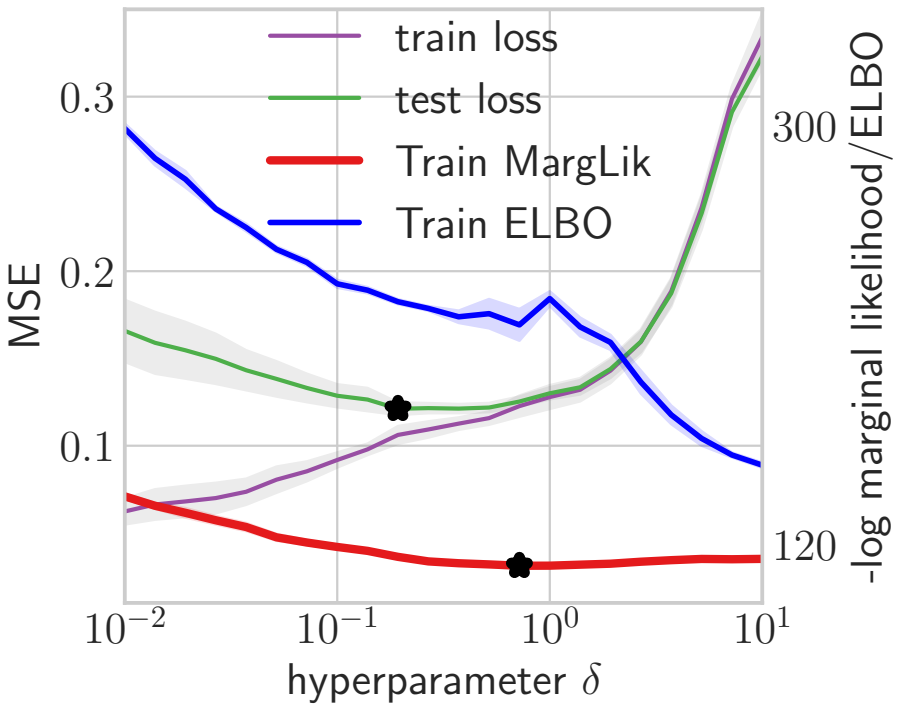
GP observations



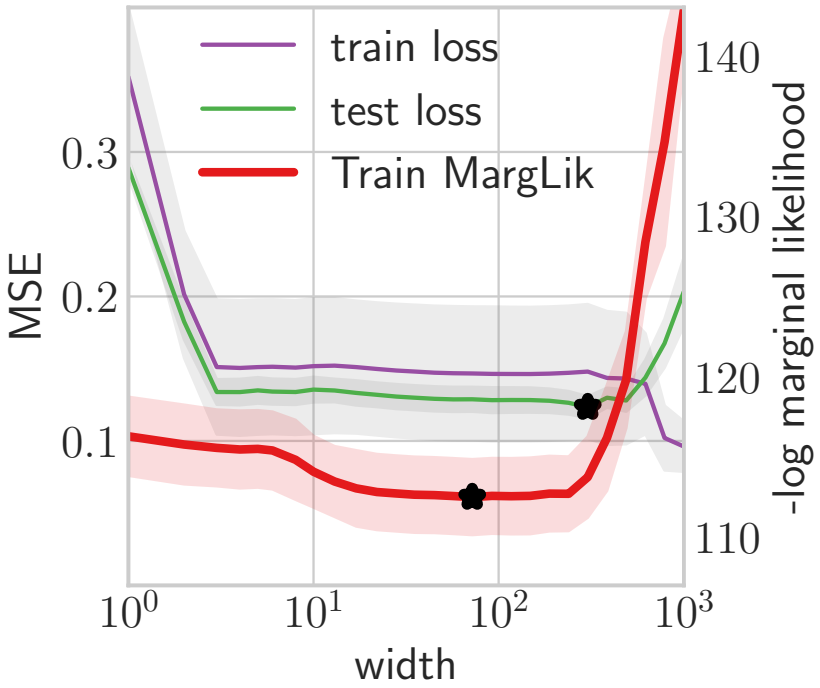
Model Selection

Marginal likelihood of the GP can be used to tune hyperparameters.
In preliminary experiments, it gives better results than using ELBO!

Regularization parameter

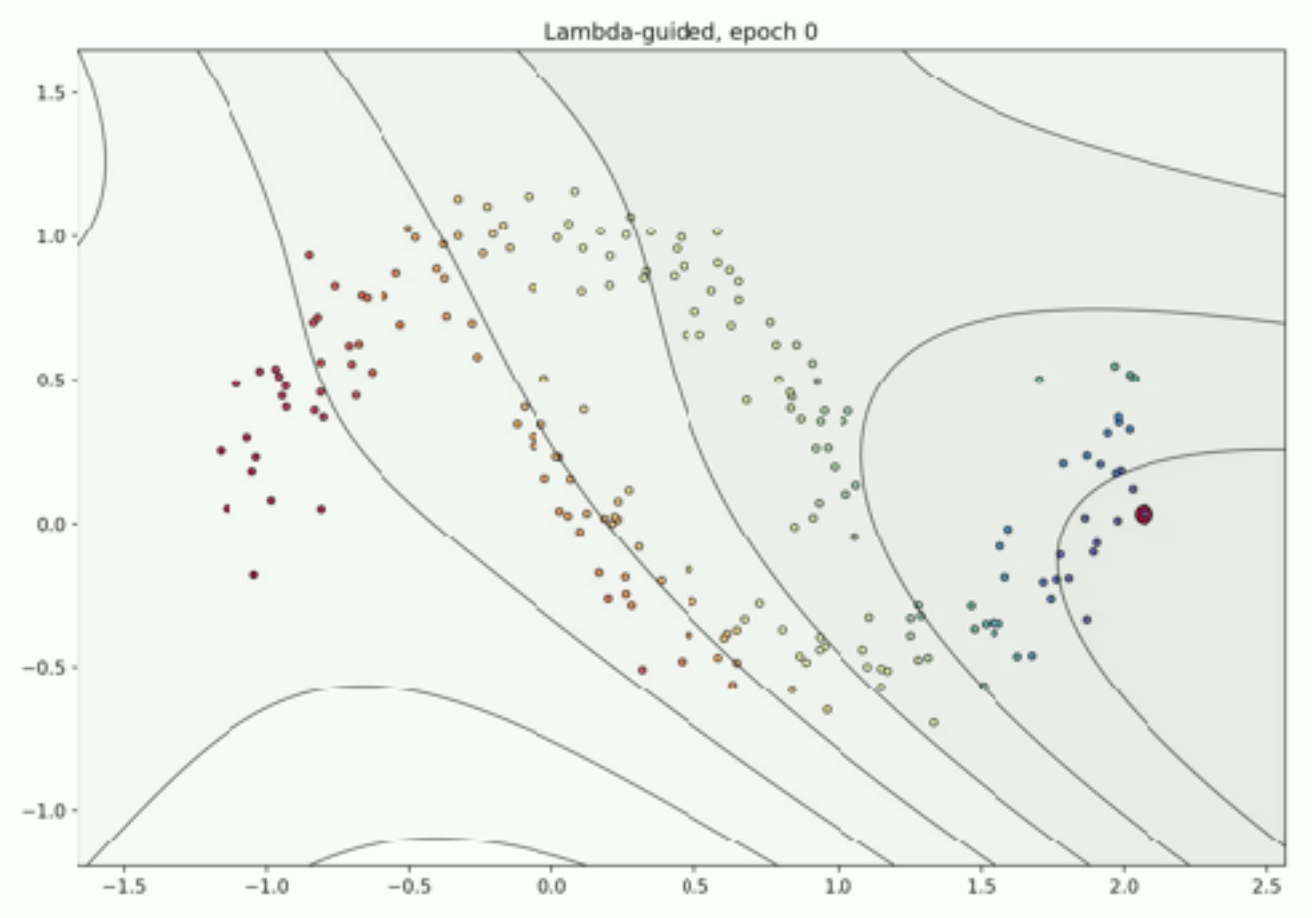


NN width



Relevance of Examples

Given a minibatch at each iteration, we select examples with less noise (low variance of epsilon_i in the approximated linear model).



(By Roman Bachmann)

How Does This Advance AI?

- Posterior Approximations are essentially representation of old data.
  - eg, Gaussians represent 2nd-order statistics.
- This representation can be employed
  - To avoid forgetting (continual learning).
  - To select examples (active learning).
  - To interact with the world (reinforcement learning).
  - To intervene (causal/interpretable learning).

References

Available at https://emtiyaz.github.io/publications.html

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

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] [Slides]

Practical Deep Learning with Bayesian Principles, (UNDER REVIEW) K. OSAWA, S. SWAROOP, A. JAIN, R. ESCHENHAGEN, R.E. TURNER, R. YOKOTA, **M.E. KHAN**. [arXiv]

Approximate Inference Turns Deep Networks into Gaussian Processes, (UNDER REVIEW) **M.E. KHAN**, A. IMMER, E. ABEDI, M. KORZEPA. [arXiv]

A 5 page review

Fast yet Simple Natural-Gradient Descent for Variational Inference in Complex Models

Mohammad Emtiyaz Khan
RIKEN Center for Advanced Intelligence Project
Tokyo, Japan
emtiyaz.khan@riken.jp

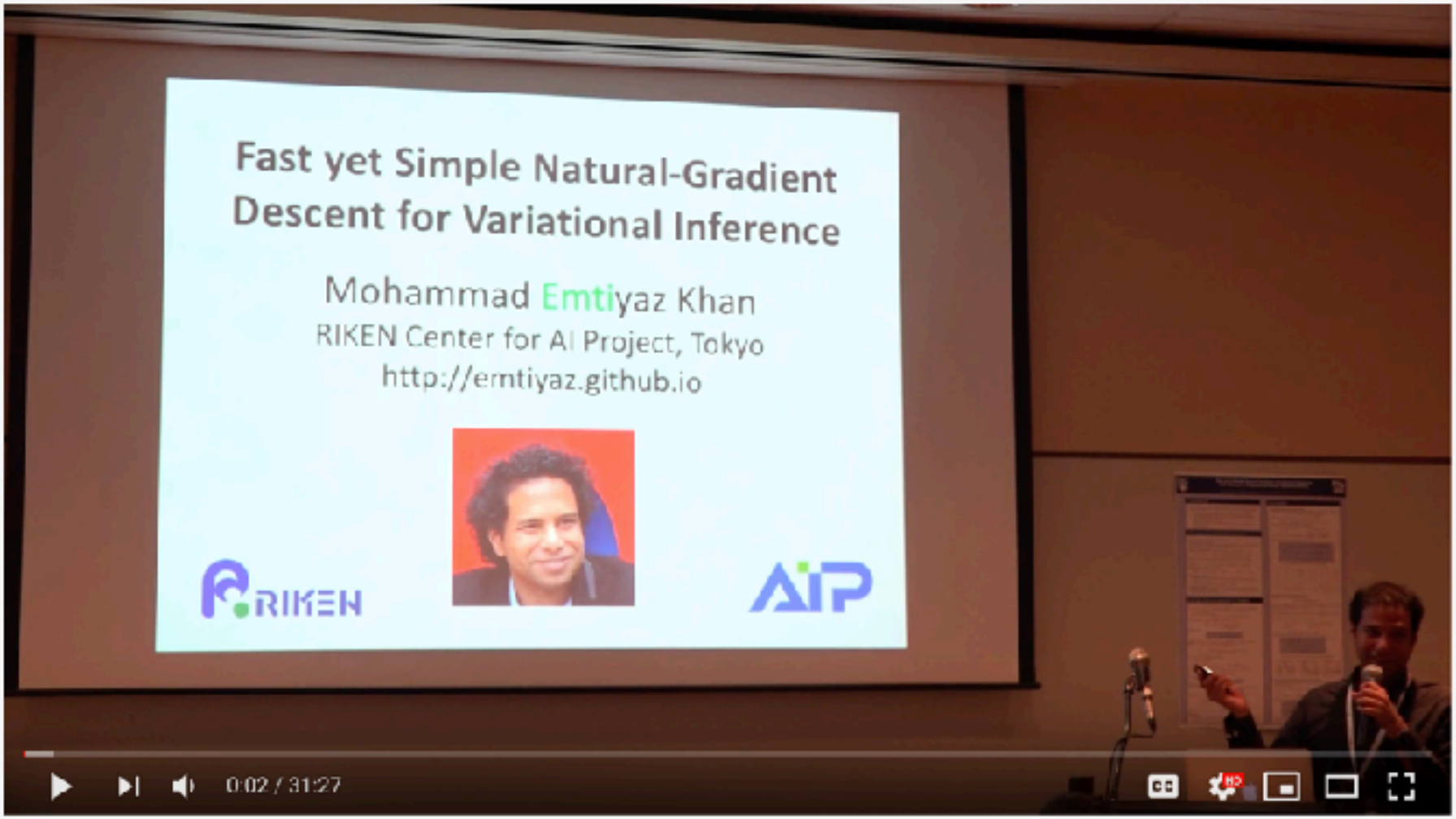
Didrik Nielsen
RIKEN Center for Advanced Intelligence Project
Tokyo, Japan
didrik.nielsen@riken.jp

***Abstract*—Bayesian inference plays an important role in advancing machine learning, but faces computational challenges when applied to complex models such as deep neural networks. Variational inference circumvents these challenges by formulating Bayesian inference as an optimization problem and solving it using gradient-based optimization. In this paper, we argue in favor of *natural-gradient* approaches which, unlike their *gradient*-based counterparts, can improve convergence by exploiting the information geometry of the solutions. We show how to derive fast yet simple natural-gradient updates by using a duality associated with exponential-family distributions. An attractive feature of these methods is that, by using natural-gradients, they are able to extract accurate local approximations for individual model components. We summarize recent results for Bayesian deep learning showing the superiority of natural-gradient approaches over their gradient counterparts.**

***Index Terms*—Bayesian inference, variational inference, natural gradients, stochastic gradients, information geometry, exponential-family distributions, nonconjugate models.**

prove the rate of convergence [7]–[9]. Unfortunately, these approaches only apply to a restricted class of models known as *conditionally-conjugate* models, and do not work for non-conjugate models such as Bayesian neural networks.

This paper discusses some recent methods that generalize the use of natural gradients to such large and complex non-conjugate models. We show that, for exponential-family approximations, a duality between their natural and expectation parameter spaces enables a simple natural gradient update. The resulting updates are equivalent to a recently proposed method called Conjugate-computation Variational Inference (CVI) [10]. An attractive feature of the method is that it naturally obtains *local* exponential-family approximations for individual model components. We discuss the application of the CVI method to Bayesian neural networks and show some recent results from a recent work [11] demonstrating



Emtiyaz Khan: Fast yet Simple Natural-Gradient Descent for Variational Inference

Acknowledgements

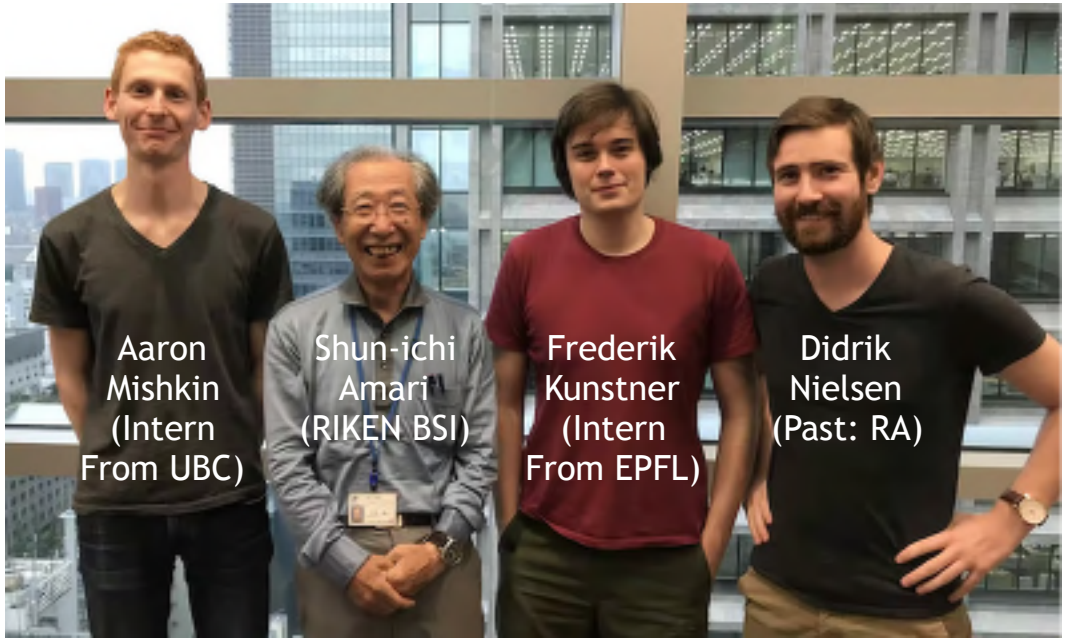
Slides, papers, & code are at emtiyaz.github.io

Wu Lin
(Past: RA)

Nicolas Hubacher
(Past: RA)

Masashi Sugiyama
(Director RIKEN-AIP)

Voot Tangkaratt
(Postdoc, RIKEN-AIP)



External Collaborators

Zuozhu Liu
(Intern from SUTD)

RAIDEN

Mark Schmidt
(UBC)

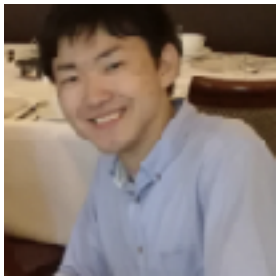
Reza Babanezhad
(UBC)

Yarin Gal
(UOxford)

Akash Srivastava
(UEdinburgh)

Acknowledgements

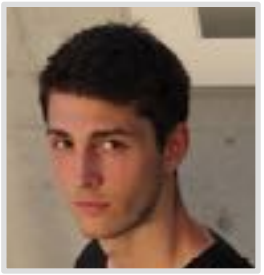
Slides, papers, & code are at emtiyaz.github.io



Kazuki Osawa
(Tokyo Tech)

Rio Yokota
(Tokyo Tech)

Anirudh Jain
(Intern from IIT-ISM, India)



Runa Eschenhagen
(Intern from University of Osnabruck)

Siddharth Swaroop
(University of Cambridge)

Rich Turner
(University of Cambridge)

Alexander Immer
(Intern from EPFL)

Ehsan Abedi
(Intern from EPFL)

Maciej Korzepa
(Intern from TU Denmark)

Pierre Alquier
(RIKEN AIP)

Approximate Bayesian Inference Team

Looking for interns, research assistants, post-docs, and collaborators.