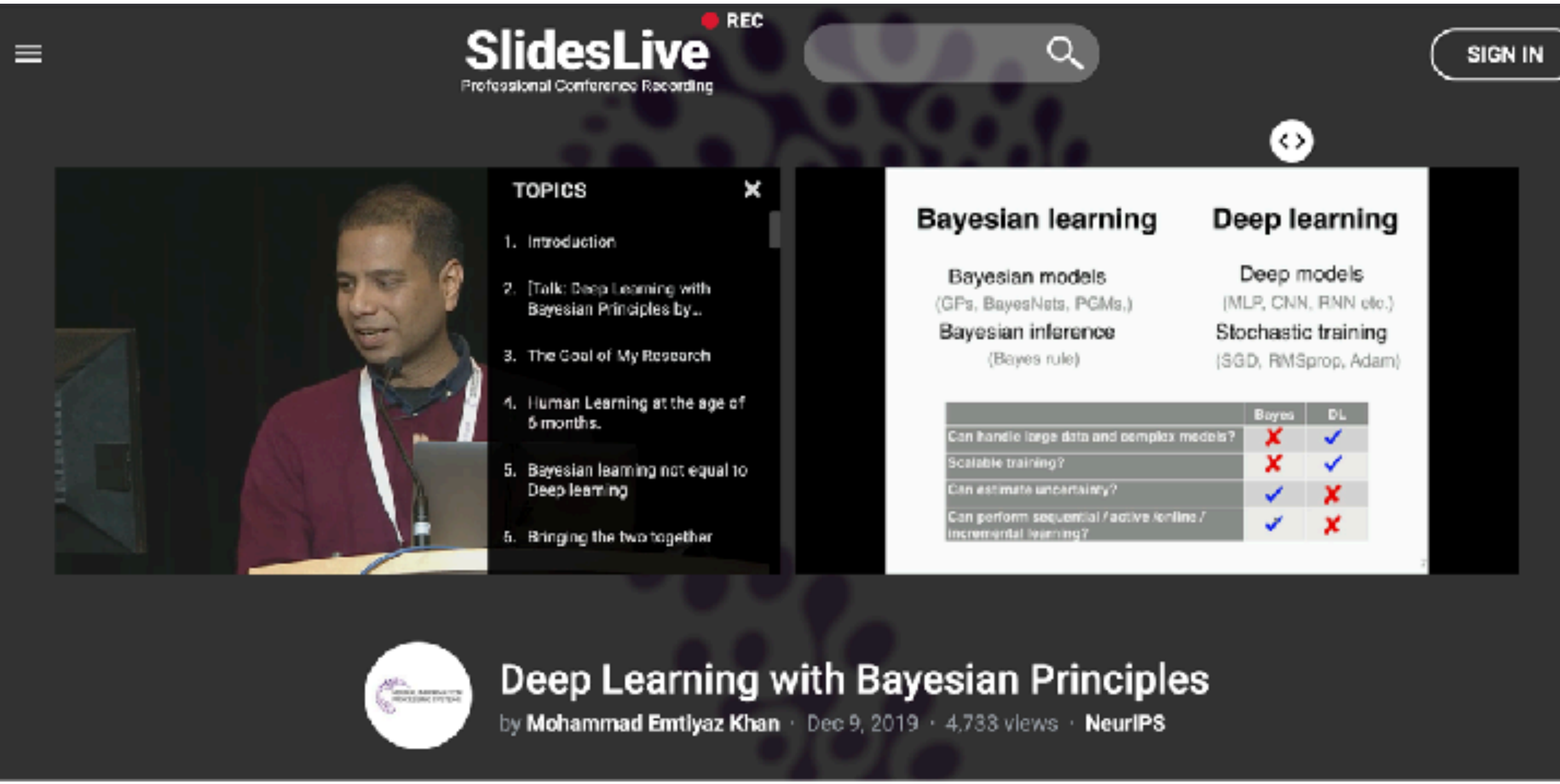
Learning-Algorithms from Bayesian Principles

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

NeurIPS 2019 Tutorial on “Deep Learning with Bayesian Principles”



https://slideslive.com/38921489/deep-learning-with-bayesian-principles

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.

Converged at the age of 12 months

Transfer skills

at the age of 14 months

Bayesian

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

Life-long learning from small chunks of data in a non-stationary world

Bulk learning from a large amount of data in a stationary world

My current research focuses on reducing this gap!

Parisi, German I., et al. "Continual lifelong learning with neural networks: A review." *Neural Networks* (2019)
Friston, K. "The free-energy principle: a unified brain theory?." *Nature reviews neuroscience* (2010)
Geisler, W. S., and Randy L. D. "Bayesian natural selection and the evolution of perceptual systems." *Philosophical Transactions of the Royal Society of London. Biological Sciences* (2002)

Learning-Algorithms from Bayesian Principles

- Bayesian principles as a general principle
  - To design/improve/generalize learning-algorithms
  - By computing “posterior approximations”
- Derive many existing algorithms,
  - Deep Learning (SGD, RMSprop, Adam)
  - Exact Bayes, Laplace, Variational Inference, etc
- Design new deep-learning algorithms
  - Uncertainty, data importance, life-long learning
- Impact: Everything with one common principle.

Deep Learning
vs
Bayesian Learning

Deep Learning (DL)

Frequentist: Empirical Risk Minimization (ERM) or Maximum Likelihood Principle, etc.

$$\min_{\theta} \ \ell(\mathcal{D}, \theta) = \sum_{i=1}^{N} [y_i - f_\theta(x_i)]^2 + \gamma \theta^T \theta$$

Loss

Data

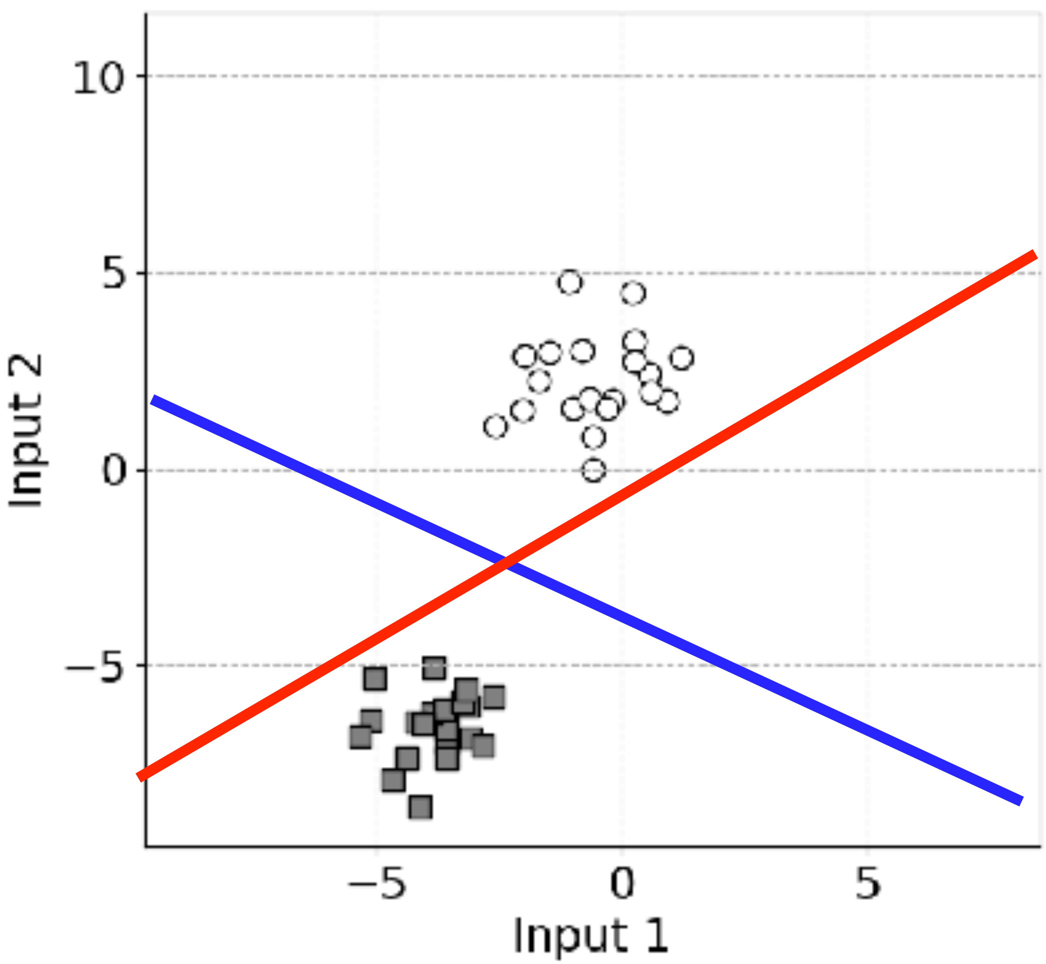
Model Params

Deep Network

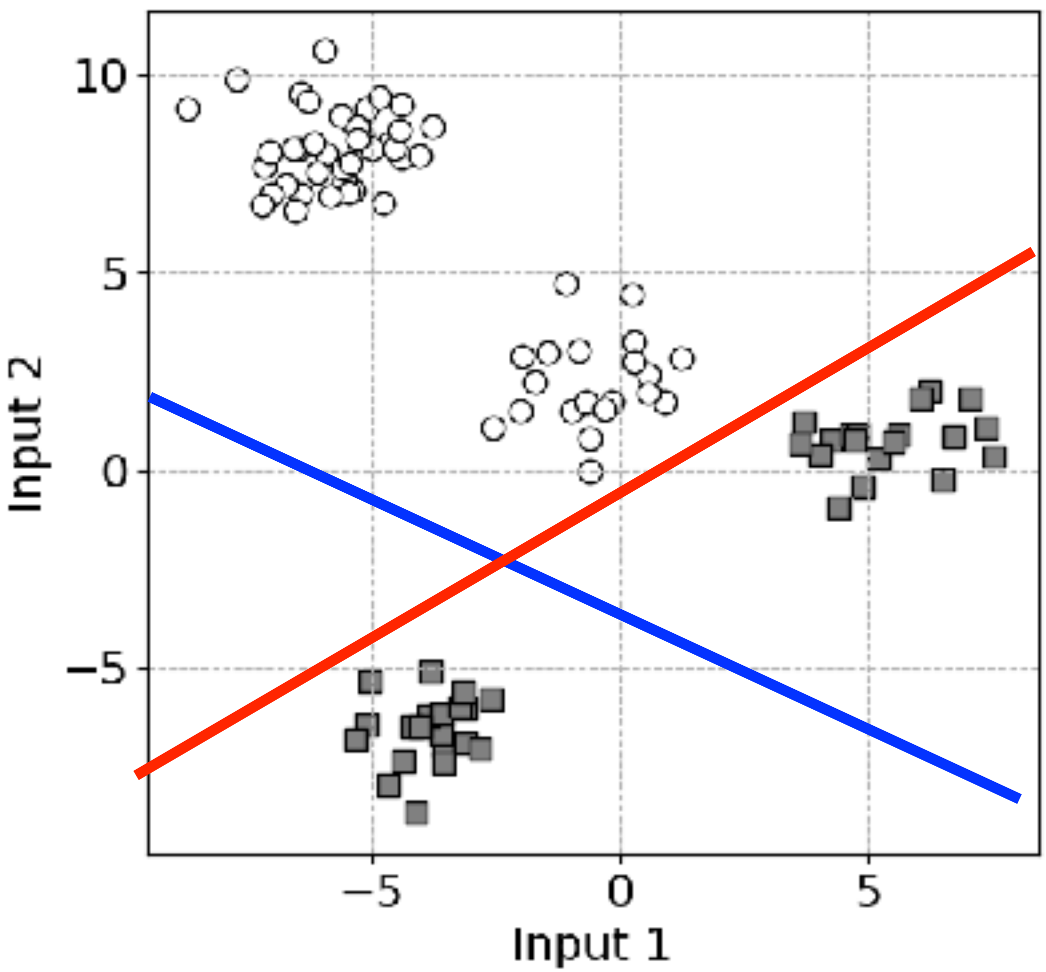
DL Algorithm: $\theta \leftarrow \theta - \rho H_\theta^{-1} \nabla_\theta \ell(\theta)$

Scales well to large data and complex model, and very good performance in practice.

Which is a good classifier?

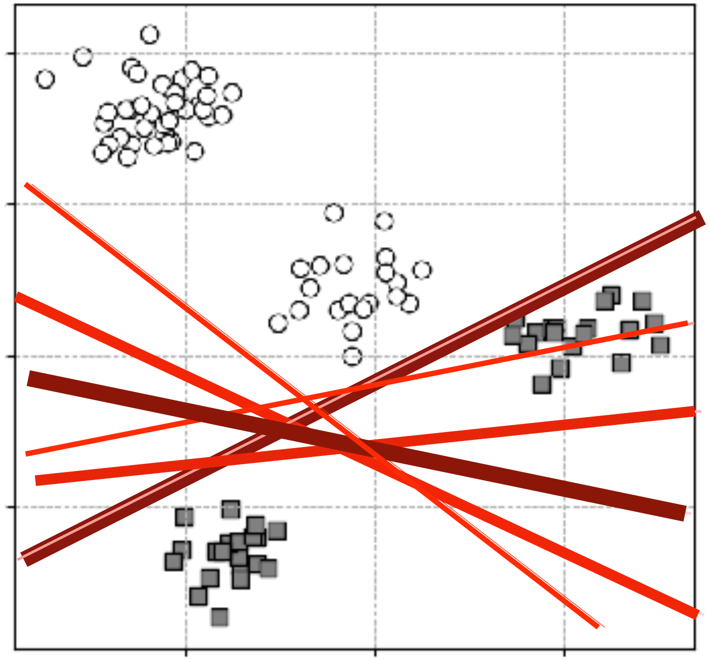


Which is a good classifier?



"What the model does not know"

Sequential Bayesian Inference



$$p(\theta|\mathcal{D}_1) = \frac{p(\mathcal{D}_1|\theta)\color{red}{p(\theta)}}{\int p(\mathcal{D}_1|\theta)p(\theta)d\theta}$$

Set the prior to the previous posterior and recompute:

$$p(\theta|\mathcal{D}_2, \mathcal{D}_1) = \frac{p(\mathcal{D}_2|\theta)\color{red}{p(\theta|\mathcal{D}_1)}}{\int p(\mathcal{D}_2|\theta)p(\theta|\mathcal{D}_1)d\theta}$$

The global property enables sequential update

Bayesian learning

Deep learning

Integration (global)

Differentiation (local)

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{\int p(\mathcal{D}|\theta)p(\theta)d\theta}$$

$$\theta \leftarrow \theta - \rho H_\theta^{-1} \nabla_\theta \ell(\theta)$$

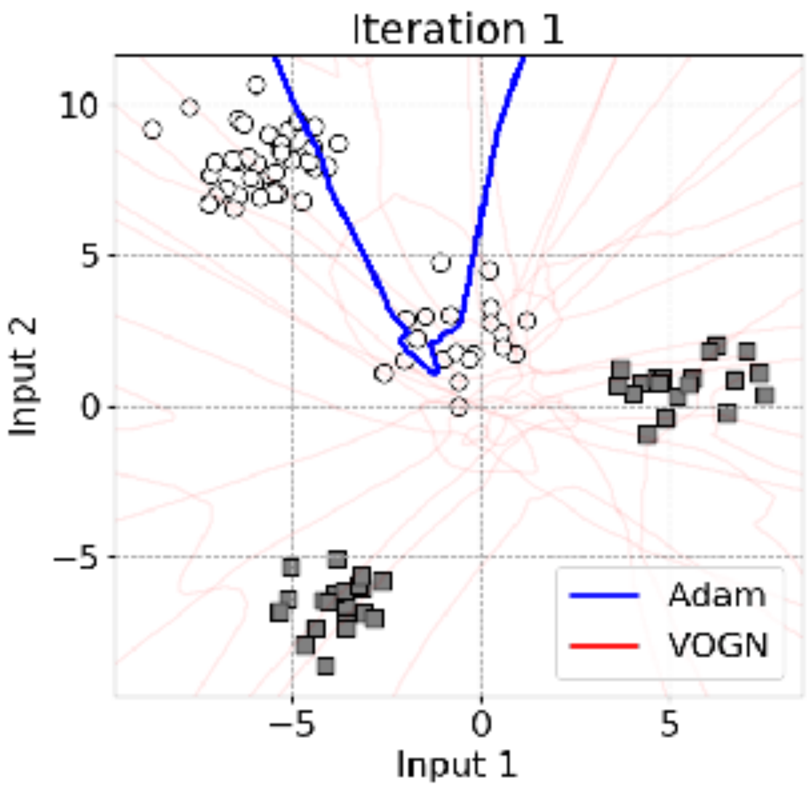
	Bayes	DL
Can handle large data and complex models?	✗	✓
Scalable training?	✗	✓
Can estimate uncertainty?	✓	✗
Can perform sequential / active /online / incremental learning?	✓	✗

Deep Learning with Bayesian Principles

- Bayesian principles as common principles
  - By computing “posterior approximations”
- Derive many existing algorithms,
  - Deep Learning (SGD, RMSprop, Adam)
  - Exact Bayes, Laplace, Variational Inference, etc
- Design new deep-learning algorithms
  - Uncertainty, data importance, life-long learning
- Impact: Many learning-algorithms with a common set of principles.

Bayes for ImageNet

VOGN, an Adam-like algorithm, for uncertainty

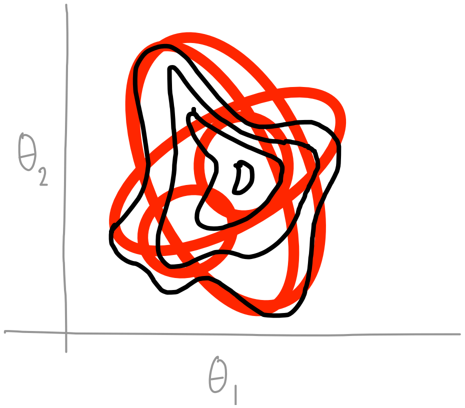


1. Khan, et al. "Fast and scalable Bayesian deep learning by weight-perturbation in Adam." *ICML* (2018).
2. Osawa et al. “Practical Deep Learning with Bayesian Principles.” NeurIPS (2019).

Bayesian principles to derive Learning-Algorithms

Main ideas: Introduce “posterior approximations” and the “Bayesian learning rule” to estimate them

Complex ⟷ Simple



Bayes’ rule

Ensemble

Newton

Gradient Descent

Exponential Family Approximations

Natural parameters → λ; Sufficient Statistics → $T(\theta)$; Expectation parameters → μ

$$q(\theta) \propto \exp\left[\lambda^\top T(\theta)\right] \qquad \mu := \mathbb{E}_q[T(\theta)]$$

$$\begin{aligned}\mathcal{N}(\theta|m, S^{-1}) &\propto \exp\left[-\frac{1}{2}(\theta - m)^\top S(\theta - m)\right] \\ &\propto \exp\left[(Sm)^\top \theta + \mathrm{Tr}\left(-\frac{S}{2}\theta\theta^\top\right)\right]\end{aligned}$$

Gaussian distribution	$q(\theta) := \mathcal{N}(\theta\|m, S^{-1})$
Natural parameters	$\lambda := \{Sm, -S/2\}$
Expectation parameters	$\mu := \{\mathbb{E}_q(\theta), \mathbb{E}_q(\theta\theta^\top)\}$

Bayesian Learning Rule

$$\min_{\theta} \ \ell(\theta) \quad \text{vs} \quad \min_{q \in \mathcal{Q}} \ \mathbb{E}_{q(\theta)}[\ell(\theta)] - \mathcal{H}(q)$$

Entropy

Deep Learning algo: $\theta \leftarrow \theta - \rho H_\theta^{-1} \nabla_\theta \ell(\theta)$

Bayes learning rule: $\lambda \leftarrow \lambda - \rho \nabla_\mu \left(\mathbb{E}_q[\ell(\theta)] - \mathcal{H}(q) \right)$

Natural and Expectation parameters of an exponential family distribution q

Deep Learning algorithms can be obtained by

1. Choosing an appropriate approximation q,
2. Giving away the “global” property of the rule

1. Khan and Lin. "Conjugate-computation variational inference: Converting variational inference in non-conjugate models to inferences in conjugate models.” AIstats (2017).

Deep Learning with Bayesian Principles

- Bayesian principles as common principles
  - By computing "posterior approximations"
- Derive many existing algorithms,
  - Deep Learning (SGD, RMSprop, Adam)
  - Exact Bayes, Laplace, Variational Inference, etc
- Design new deep-learning algorithms
  - Uncertainty, data importance, life-long learning
- Impact: Many learning-algorithms with a common set of principles.

Gradient Descent from Bayes

Gradient descent: $\theta \leftarrow \theta - \rho \nabla_\theta \ell(\theta)$

Bayes Learn Rule: $m \leftarrow m - \rho \nabla_m \ell(m)$

"Global" to "local"
$\mathbb{E}_q[\ell(\theta)] \approx \ell(m)$

$$m \leftarrow m - \rho \nabla_m \mathbb{E}_q[\ell(\theta)]$$

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

Derived by choosing Gaussian with fixed covariance

Gaussian distribution $q(\theta) := \mathcal{N}(m, 1)$

Natural parameters $\lambda := m$

Expectation parameters $\mu := \mathbb{E}_q[\theta] = m$

Entropy $\mathcal{H}(q) := \log(2\pi)/2$

Using stochastic gradients, we get SGD

1. Khan and Rue. "Learning-Algorithms from Bayesian Principles" (2019) (work in progress, an early draft available at https://emtiyaz.github.io/papers/learning_from_bayes.pdf)

Newton's Method from Bayes

Newton's method: $\theta \leftarrow \theta - H_\theta^{-1} \left[\nabla_\theta \ell(\theta)\right]$

$$Sm \leftarrow (1-\rho) Sm - \rho \nabla_{\mathbb{E}_q(\theta)} \mathbb{E}_q[\ell(\theta)]$$

$$-\frac{1}{2} S \leftarrow (1-\rho)\left(-\frac{1}{2} S\right) - \rho \nabla_{\mathbb{E}_q(\theta\theta^\top)} \mathbb{E}_q[\ell(\theta)]$$

$$S \leftarrow (1-\rho) S + \rho 2 \nabla_{\mathbb{E}_q(\theta\theta^\top)} \mathbb{E}_q[\ell(\theta)]$$

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

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

$$-\nabla_\mu \mathcal{H}(q) = \lambda$$

Derived by choosing a multivariate Gaussian

Gaussian distribution	$q(\theta) := \mathcal{N}(\theta \mid m, S^{-1})$
Natural parameters	$\lambda := \{Sm, -S/2\}$
Expectation parameters	$\mu := \{\mathbb{E}_q(\theta), \mathbb{E}_q(\theta\theta^\top)\}$

1. Khan, et al. "Fast and scalable Bayesian deep learning by weight-perturbation in Adam." *ICML* (2018).

Newton's Method from Bayes

Newton's method: $\theta \leftarrow \theta - H_\theta^{-1}\left[\nabla_\theta \ell(\theta)\right]$

Set ρ=1 to get $\quad m \leftarrow m - H_m^{-1}[\nabla_m \ell(m)]$

$$m \leftarrow m - \rho S^{-1} \nabla_m \ell(m)$$
$$S \leftarrow (1-\rho) S + \rho H_m$$

"Global" to "local"
$\mathbb{E}_q[\ell(\theta)] \approx \ell(m)$

Express in terms of gradient and Hessian of loss:

$$\nabla_{\mathbb{E}_q(\theta)} \mathbb{E}_q[\ell(\theta)] = \mathbb{E}_q[\nabla_\theta \ell(\theta)] - 2\mathbb{E}_q[H_\theta] m$$

$$\nabla_{\mathbb{E}_q(\theta\theta^\top)} \mathbb{E}_q[\ell(\theta)] = \mathbb{E}_q[H_\theta]$$

$$Sm \leftarrow (1-\rho) Sm - \rho \nabla_{\mathbb{E}_q(\theta)} \mathbb{E}_q[\ell(\theta)]$$
$$S \leftarrow (1-\rho) S - \rho 2 \nabla_{\mathbb{E}_q(\theta\theta^\top)} \mathbb{E}_q[\ell(\theta)]$$

1. Khan, et al. "Fast and scalable Bayesian deep learning by weight-perturbation in Adam." *ICML* (2018).

RMSprop/Adam from Bayes

RMSprop

$$s \leftarrow (1-\rho)s + \rho[\hat{\nabla}\ell(\theta)]^2$$
$$\theta \leftarrow \theta - \alpha(\sqrt{s}+\delta)^{-1}\hat{\nabla}\ell(\theta)$$

Bayesian Learning rule for multivariate Gaussian

$$S \leftarrow (1-\rho)S + \rho(H_\theta)$$
$$m \leftarrow m - \alpha S^{-1}\nabla_\theta \ell(\theta)$$

To get RMSprop, make the following choices

- Choose Gaussian with diagonal covariance
- Replace Hessian by square of gradients
- Add square root for scaling vector

For Adam, use a Heavy-ball term with KL divergence as momentum (Appendix E in [1])

1. Khan, et al. "Fast and scalable Bayesian deep learning by weight-perturbation in Adam." *ICML* (2018).

Summary

- Gradient descent is derived using a Gaussian with fixed covariance, and estimating the mean
- Newton's method is derived using multivariate Gaussian
- RMSprop is derived using diagonal covariance
- Adam is derived by adding heavy-ball momentum term
- For "ensemble of Newton", use Mixture of Gaussians [1]
- To derive DL algorithms, we need to switch from a "global" to "local" approximation $\mathbb{E}_q[\ell(\theta)] \approx \ell(m)$
- Then, to improve DL algorithms, we just need to add some "global" touch to the DL algorithms

1. Lin, Wu, Mohammad Emtiyaz Khan, and Mark Schmidt. "Fast and Simple Natural-Gradient Variational Inference with Mixture of Exponential-family Approximations." *ICML* (2019).

Deep Learning with Bayesian Principles

- Bayesian principles as common principles
  - By computing “posterior approximations”
- Derive many existing algorithms,
  - Deep Learning (SGD, RMSprop, Adam)
  - Exact Bayes, Laplace, Variational Inference, etc
- Design new deep-learning algorithms
  - Uncertainty, data importance, life-long learning
- Impact: Many learning-algorithms with a common set of principles.

Learning-Algorithms from Bayesian Principles

Bayesian learning rule: $\lambda \leftarrow \lambda - \rho \nabla_{\mu} \left(\mathbb{E}_q[\ell(\theta)] - \mathcal{H}(q) \right)$

Given a loss, we can recover a variety of learning algorithms by choosing an appropriate q

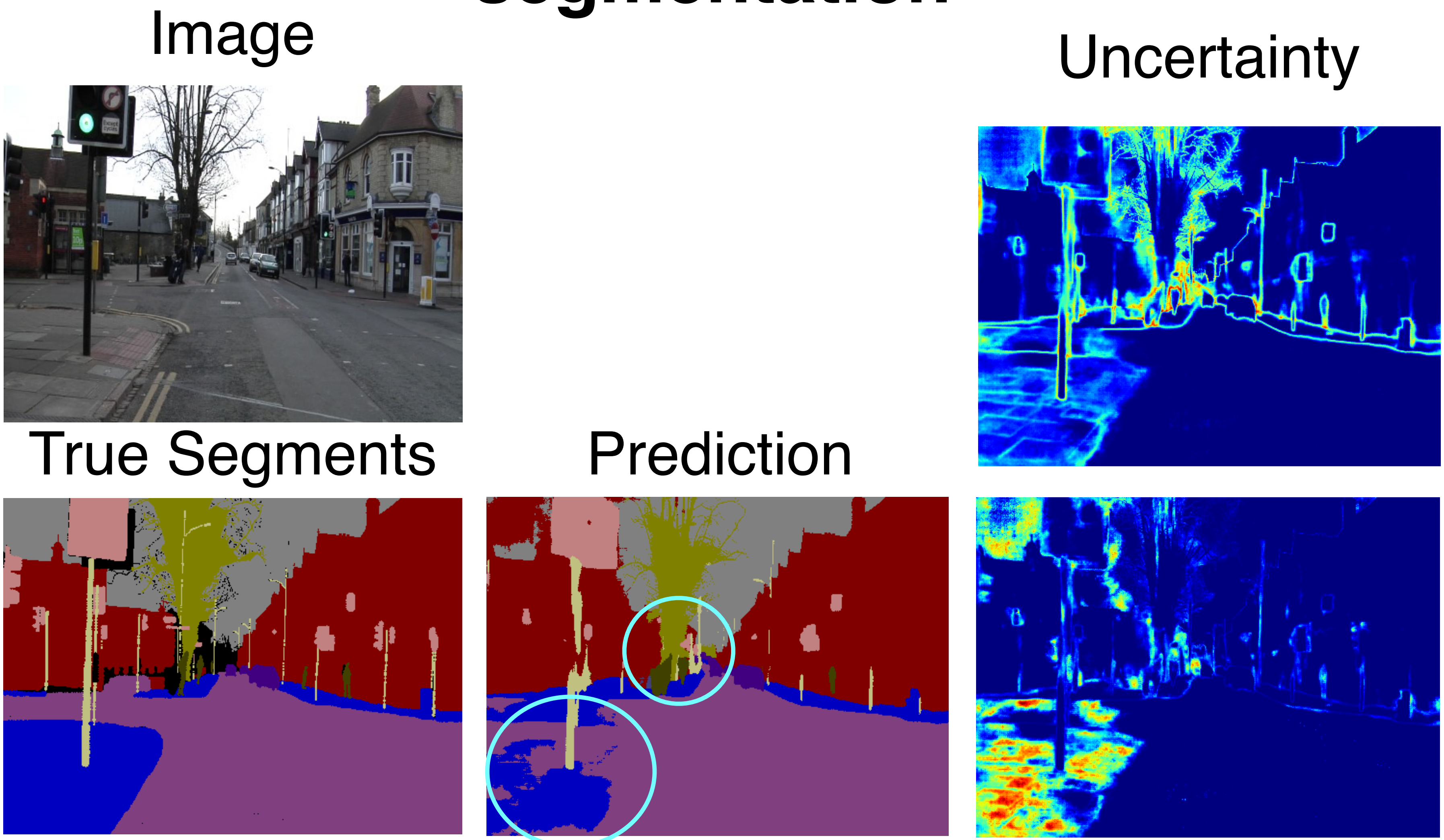
- Classical algorithms: Least-squares, gradient descent, Newton's method, Kalman filters, Baum-Welch, Forward-backward, etc.
- Bayesian inference: EM, Laplace's method, SVI, VMP.
- Deep learning: SGD, RMSprop, Adam.
- Reinforcement learning: parameter-space exploration, natural policy-search.
- Continual learning: Elastic-weight consolidation.
- Online learning: Exponential-weight average.
- Global optimization: Natural evolutionary strategies, Gaussian homotopy, continuation method & smoothed optimization.

1. Khan and Rue. "Learning-Algorithms from Bayesian Principles" (2019) (work in progress, an early draft available at https://emtiyaz.github.io/papers/learning_from_bayes.pdf)

Deep Learning with Bayesian Principles

- Bayesian principles as common principles
  - By computing "posterior approximations"
- Derive many existing algorithms,
  - Deep Learning (SGD, RMSprop, Adam)
  - Exact Bayes, Laplace, Variational Inference, etc
- Design new deep-learning algorithms
  - Uncertainty, data importance, life-long learning
- Impact: Many learning-algorithms with a common set of principles.

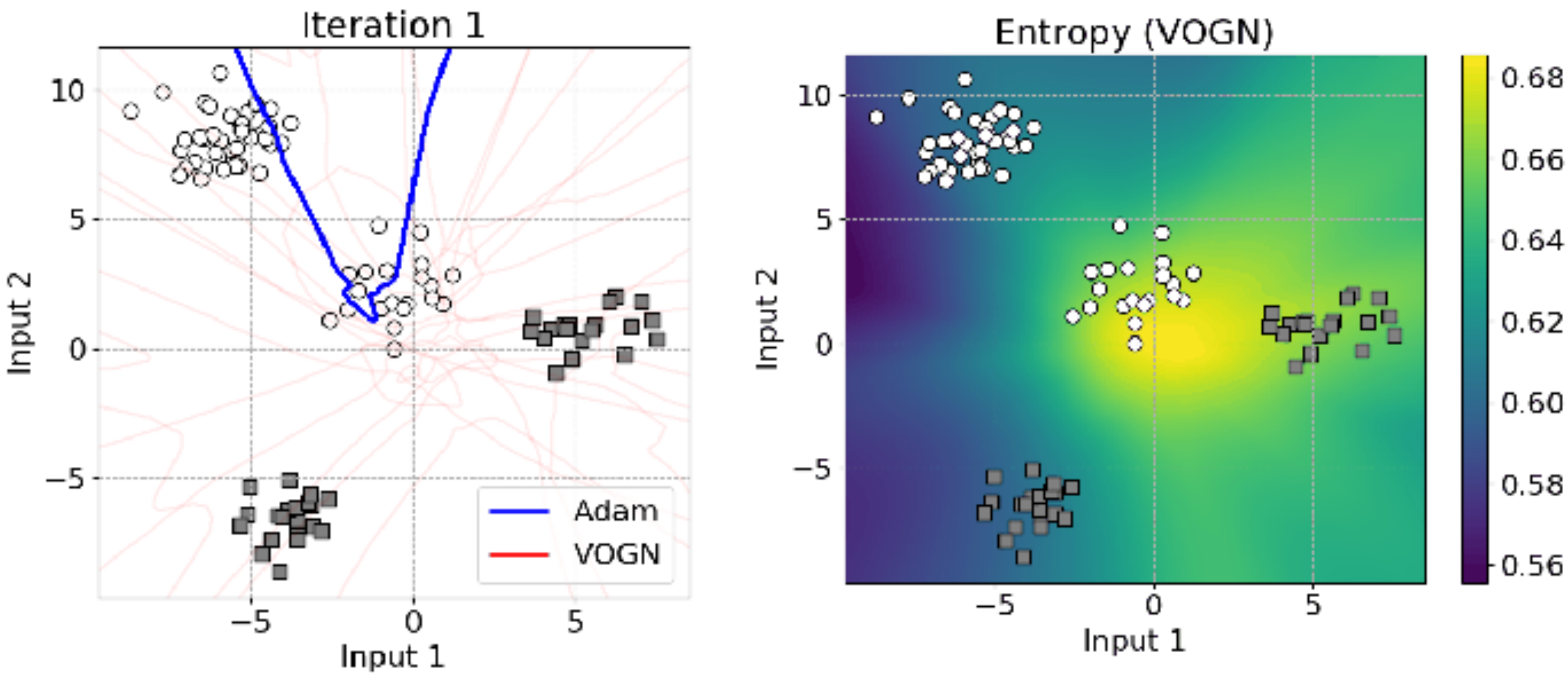
Uncertainty Estimation for Image segmentation



Kendall, Alex, Yarin Gal, and Roberto Cipolla. "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics." *CVPR*. 2018.

Scaling up VI to ImageNet

VOGN, an Adam-like algorithm, for uncertainty



1. Khan, et al. "Fast and scalable Bayesian deep learning by weight-perturbation in Adam." *ICML* (2018).
2. Osawa et al. “Practical Deep Learning with Bayesian Principles.” NeurIPS (2019).

Variational Online Gauss-Newton

- Improve RMSprop with the Bayesian "touch"
  - Remove the "local" approximation $\mathbb{E}_q[\ell(\theta)] \approx \ell(m)$
  - Use a second-order approximation
  - No square root of the scale
- Improve VOGN by using deep learning tricks
  - Momentum, batch norm, data augmentation etc

RMSprop

$$g \leftarrow \hat{\nabla}\ell(\theta)$$
$$s \leftarrow (1-\rho)s + \rho g^2$$
$$\theta \leftarrow \theta - \alpha(\sqrt{s} + \delta)^{-1} g$$

VOGN

$$g \leftarrow \hat{\nabla}\ell(\theta), \text{ where } \theta \sim \mathcal{N}(m, \sigma^2)$$
$$s \leftarrow (1-\rho)s + \rho(\Sigma_i g_i^2)$$
$$m \leftarrow m - \alpha(s+\gamma)^{-1}\nabla_\theta \ell(\theta)$$
$$\sigma^2 \leftarrow (s+\gamma)^{-1}$$

1. Khan, et al. "Fast and scalable Bayesian deep learning by weight-perturbation in Adam." *ICML* (2018).
2. Osawa et al. "Practical Deep Learning with Bayesian Principles." NeurIPS (2019).

Adam to VOGN

"Adam" to "VOGN" in two lines of code change.

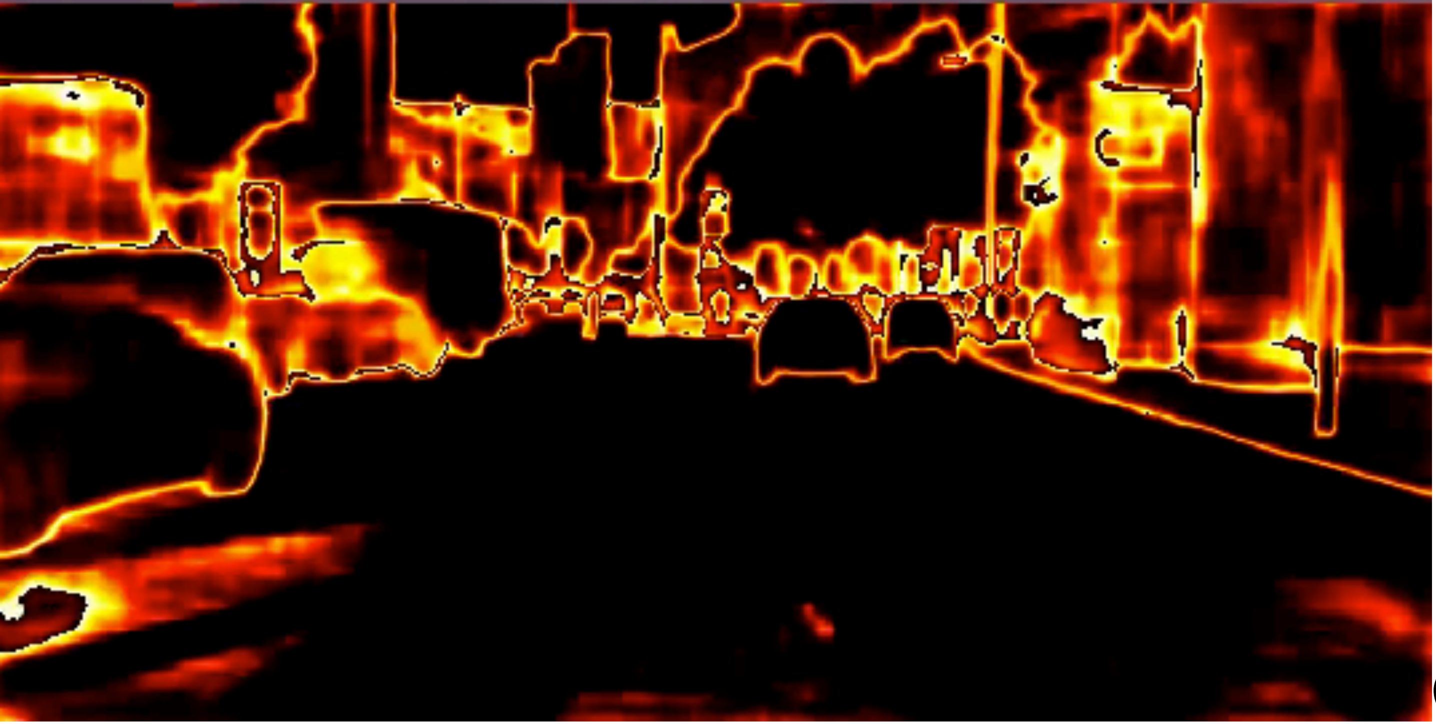
```
import torch
+import torchsso

train_loader = torch.utils.data.DataLoader(train_dataset)
model = MLP()

-optimizer = torch.optim.Adam(model.parameters())
+optimizer = torchsso.optim.VOGN(model, dataset_size=len(train_loader.dataset))
```

Available at https://github.com/team-approx-bayes/dl-with-bayes

Uses many practical tricks of DL to scale Bayes

1. Khan, et al. "Fast and scalable Bayesian deep learning by weight-perturbation in Adam." *ICML* (2018).
2. Osawa et al. "Practical Deep Learning with Bayesian Principles." NeurIPS (2019).

Image Segmentation

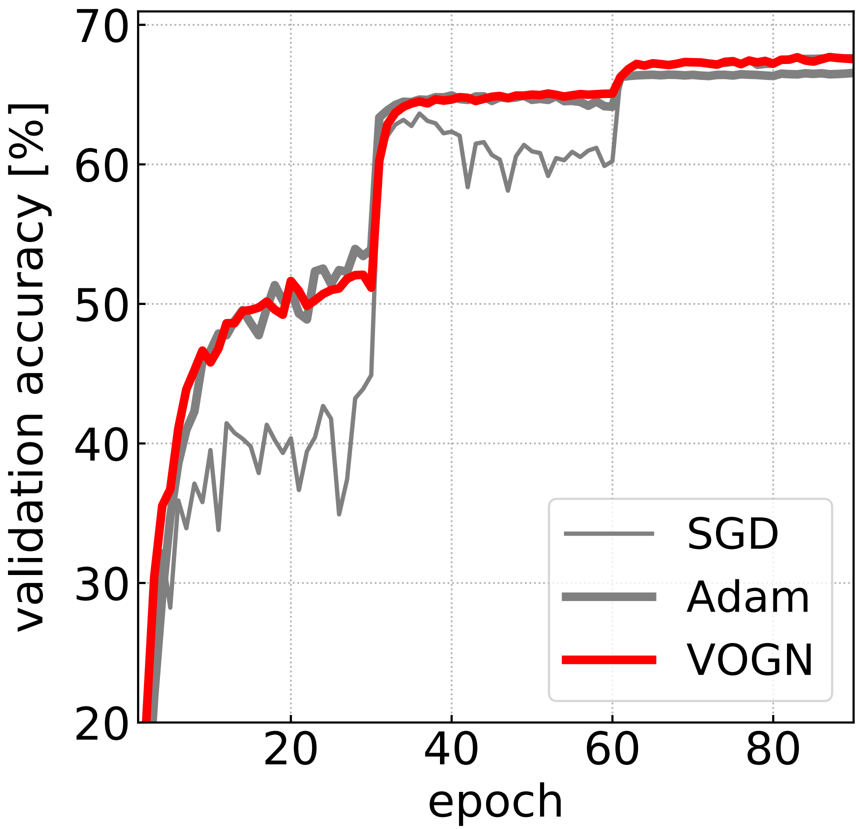


Uncertainty (entropy of class probs)

(By Roman Bachmann)

VOGN on ImageNet

State-of-the-art performance and convergence rate, while preserving benefits of Bayesian principles



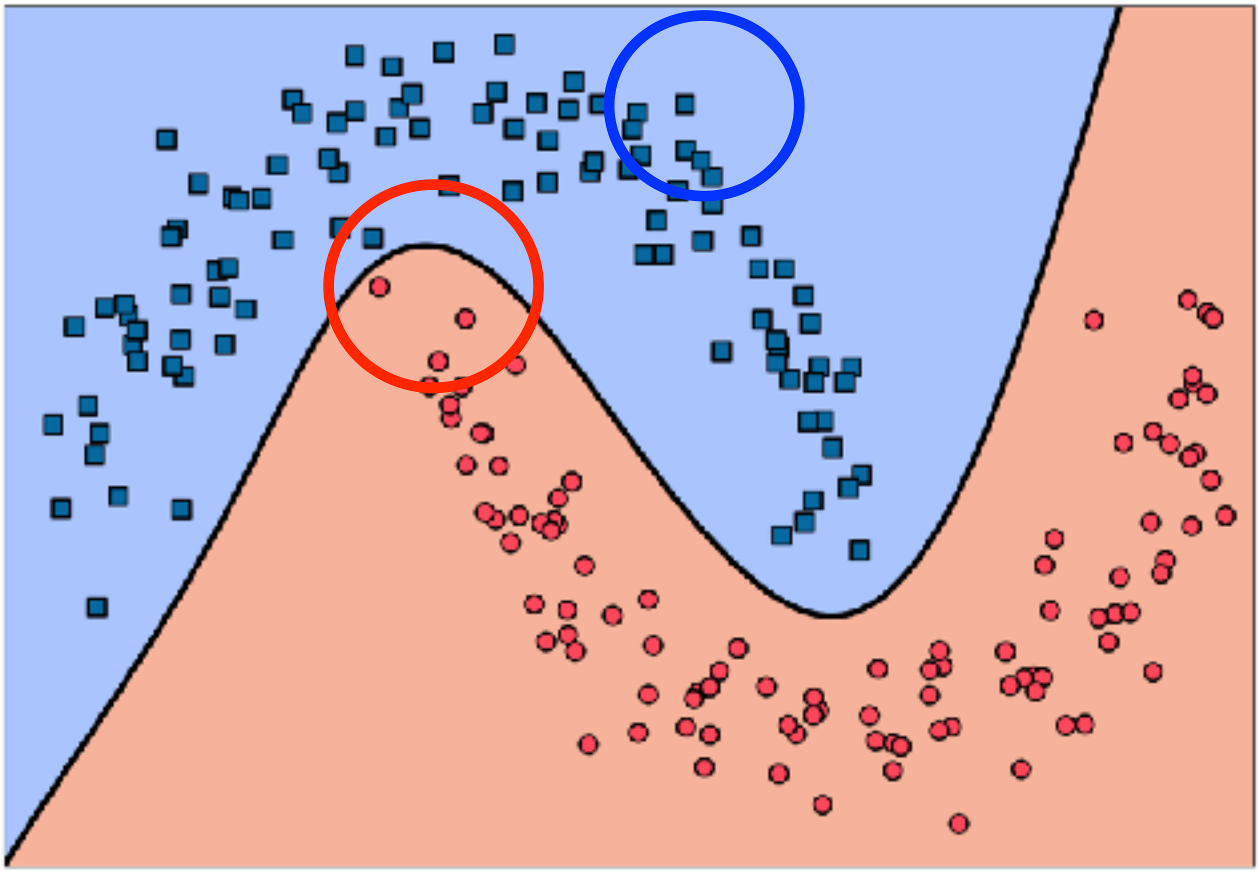
1. Osawa et al. “Practical Deep Learning with Bayesian Principles.” NeurIPS (2019).

Deep Learning with Bayesian Principles

- Bayesian principles as common principles
  - By computing “posterior approximations”
- Derive many existing algorithms,
  - Deep Learning (SGD, RMSprop, Adam)
  - Exact Bayes, Laplace, Variational Inference, etc
- Design new deep-learning algorithms
  - Uncertainty, data importance, life-long learning
- Impact: Many learning-algorithms with a common set of principles.

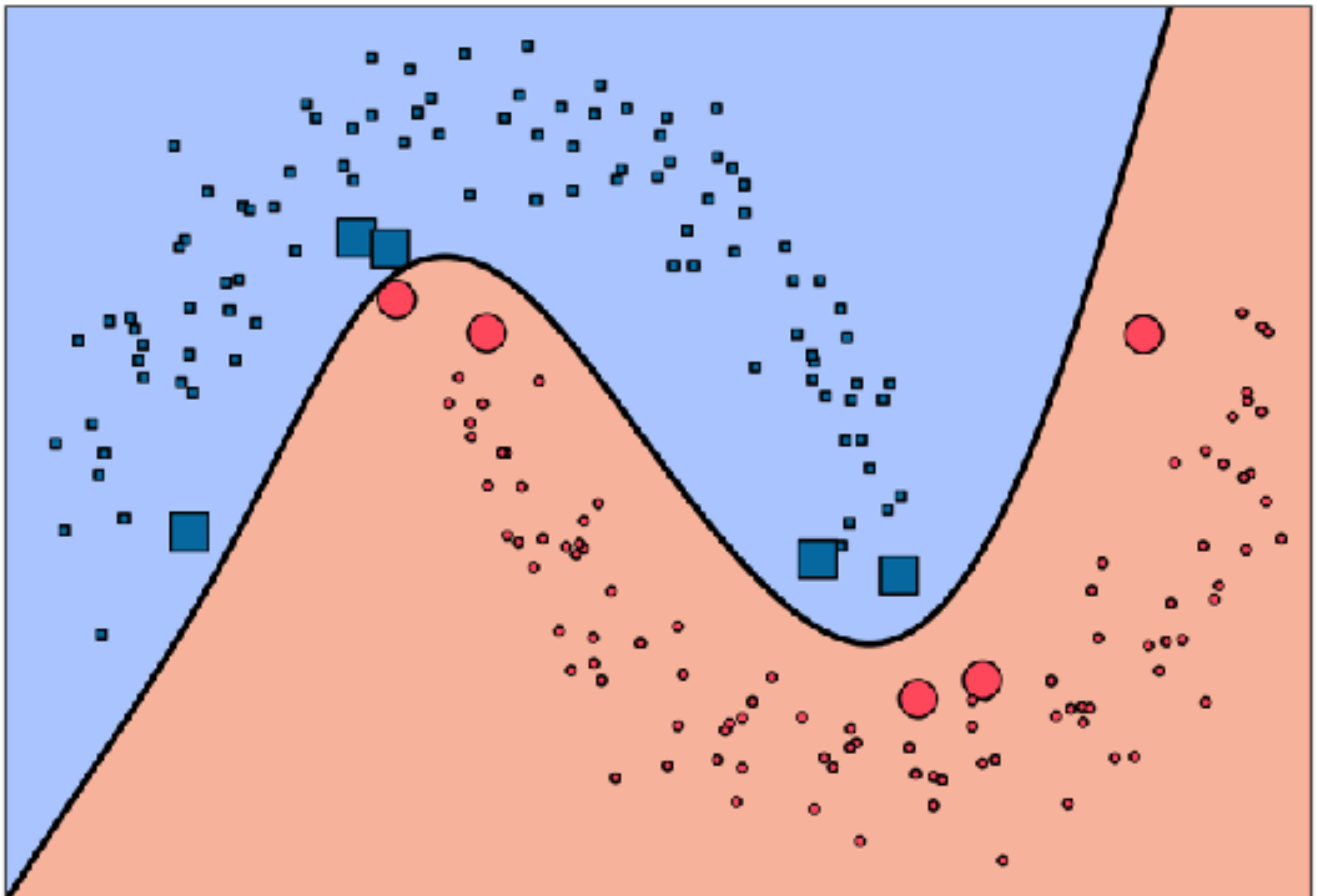
Importance of Data Examples

Which examples are most important for the classifier? Red circle vs Blue circle.



Model view vs Data view

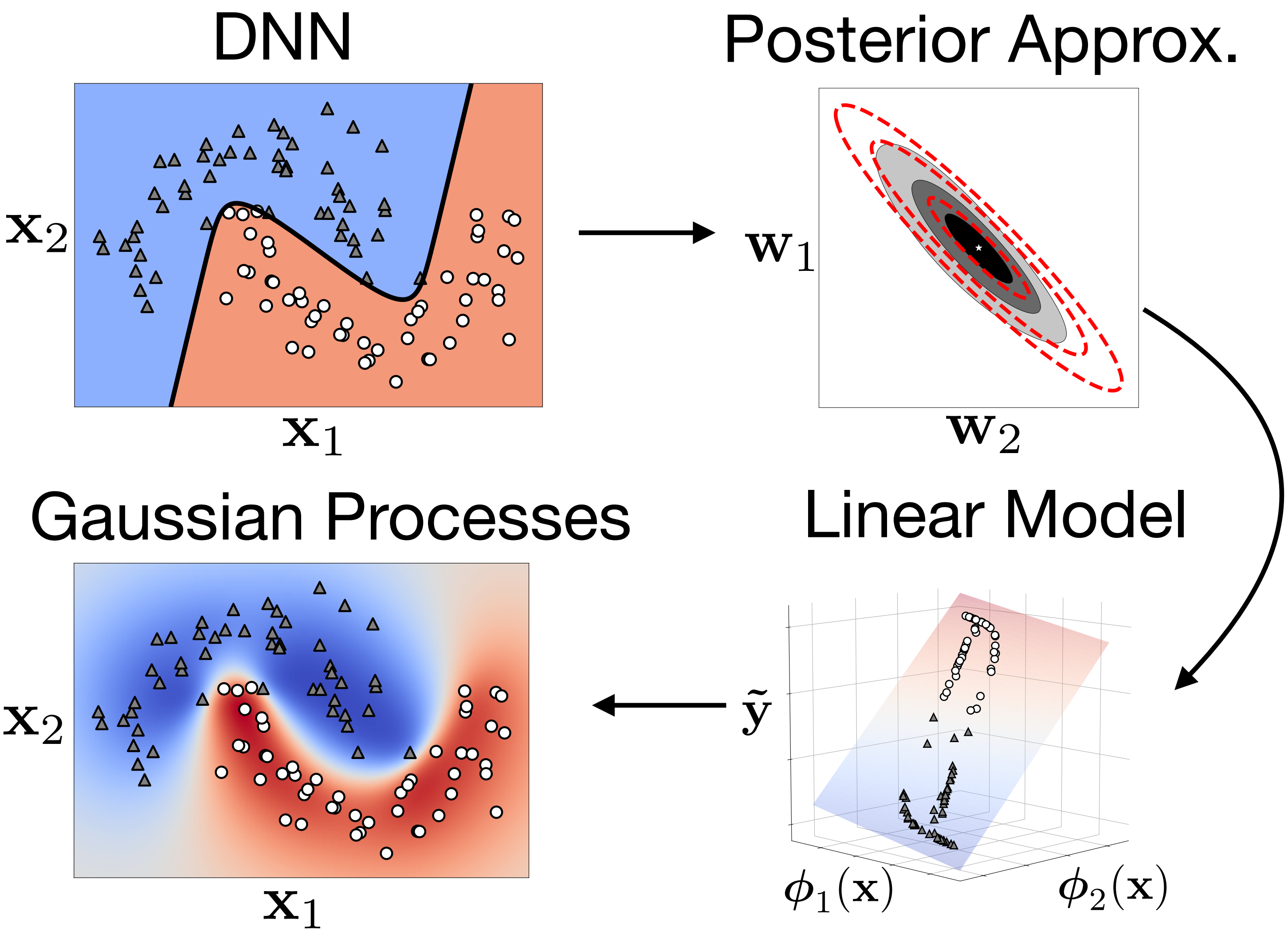
Bayes "automatically" defines data-Importance



Data view

(By Roman Bachmann)

DNN to GP



1. Khan et al., Approximate Inference Turns Deep Networks into Gaussian Processes, NeurUPS, 2019

“Global” to “Local”

Posterior approximations connect “global” parameters (e.g. DNN weights) to “local” parameters (e.g. data examples)

$$\sum_{i=1}^{N} \ell(y_i, f_\theta(x_i)) \approx \sum_{i=1}^{N} \frac{1}{\sigma_i^2} [\tilde{y}_i - \phi_i(x_i)^\top \theta]^2$$

neural network

“Dual” variables

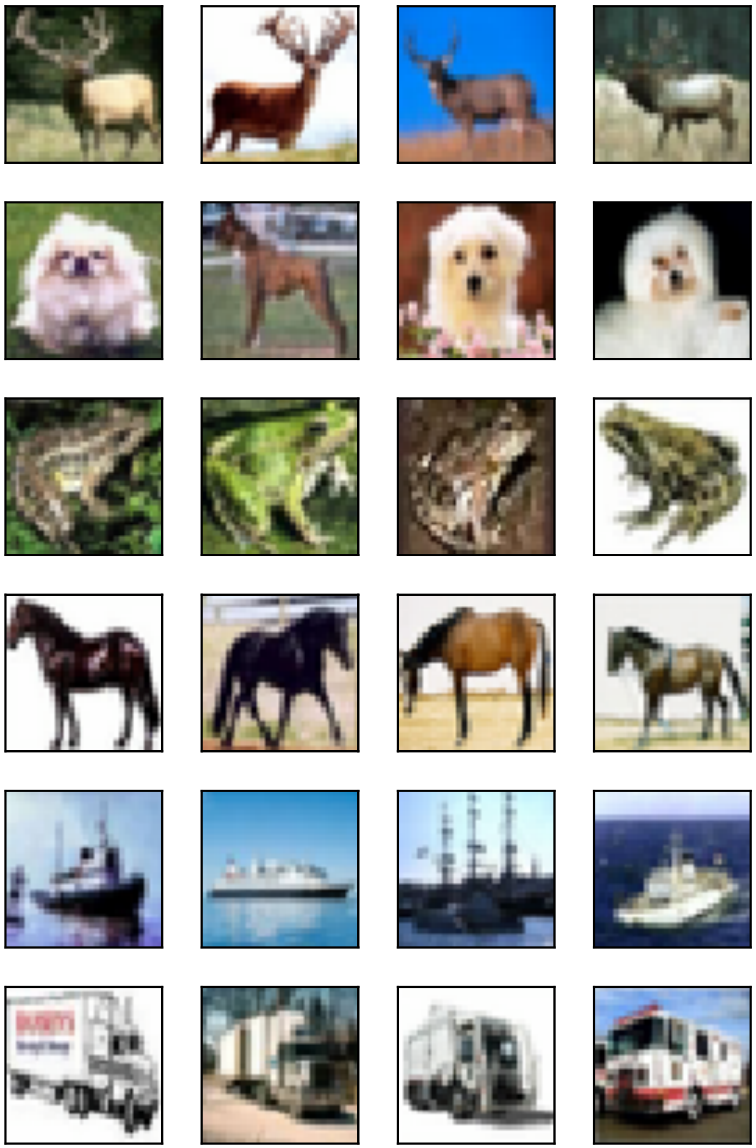
The local parameters can be seen as “dual” variables that define the “importance” of the data

1. Khan et al. "Fast dual variational inference for non-conjugate latent gaussian models." *ICML* (2013).
2. Khan et al. "Approximate Inference Turns Deep Networks into Gaussian Processes." *NeurIPS* (2019).

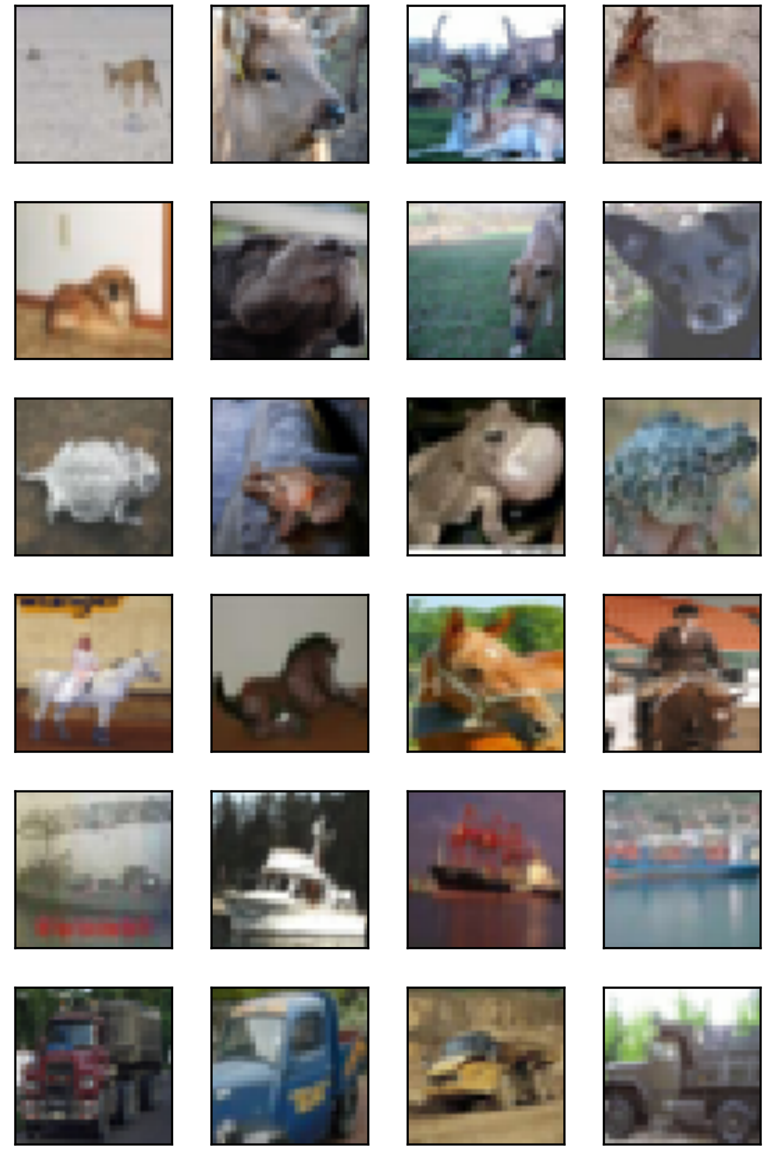
Least Important

Most Important

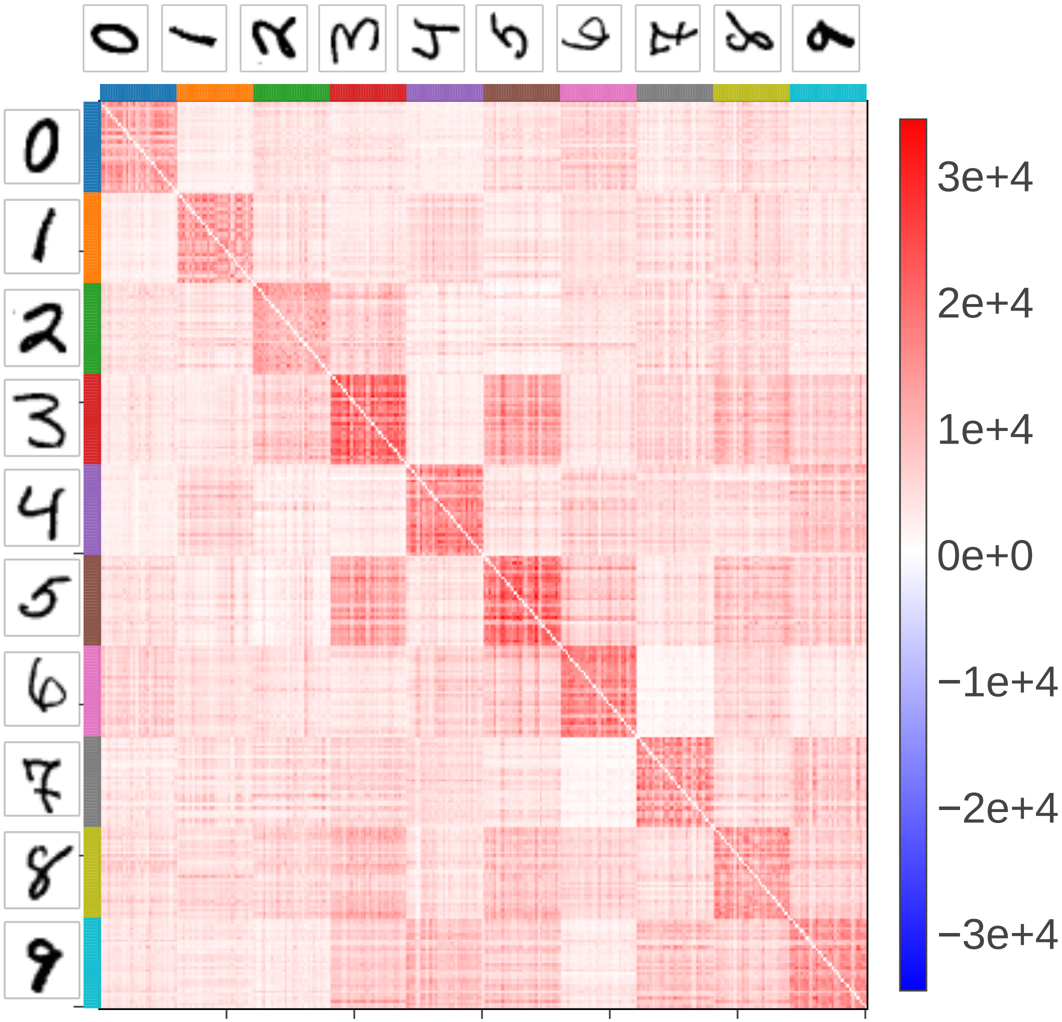
Least Important



Most Important



Similarity (Kernel) Matrix



$$K_{ij} := \phi_i^\top \phi_j$$

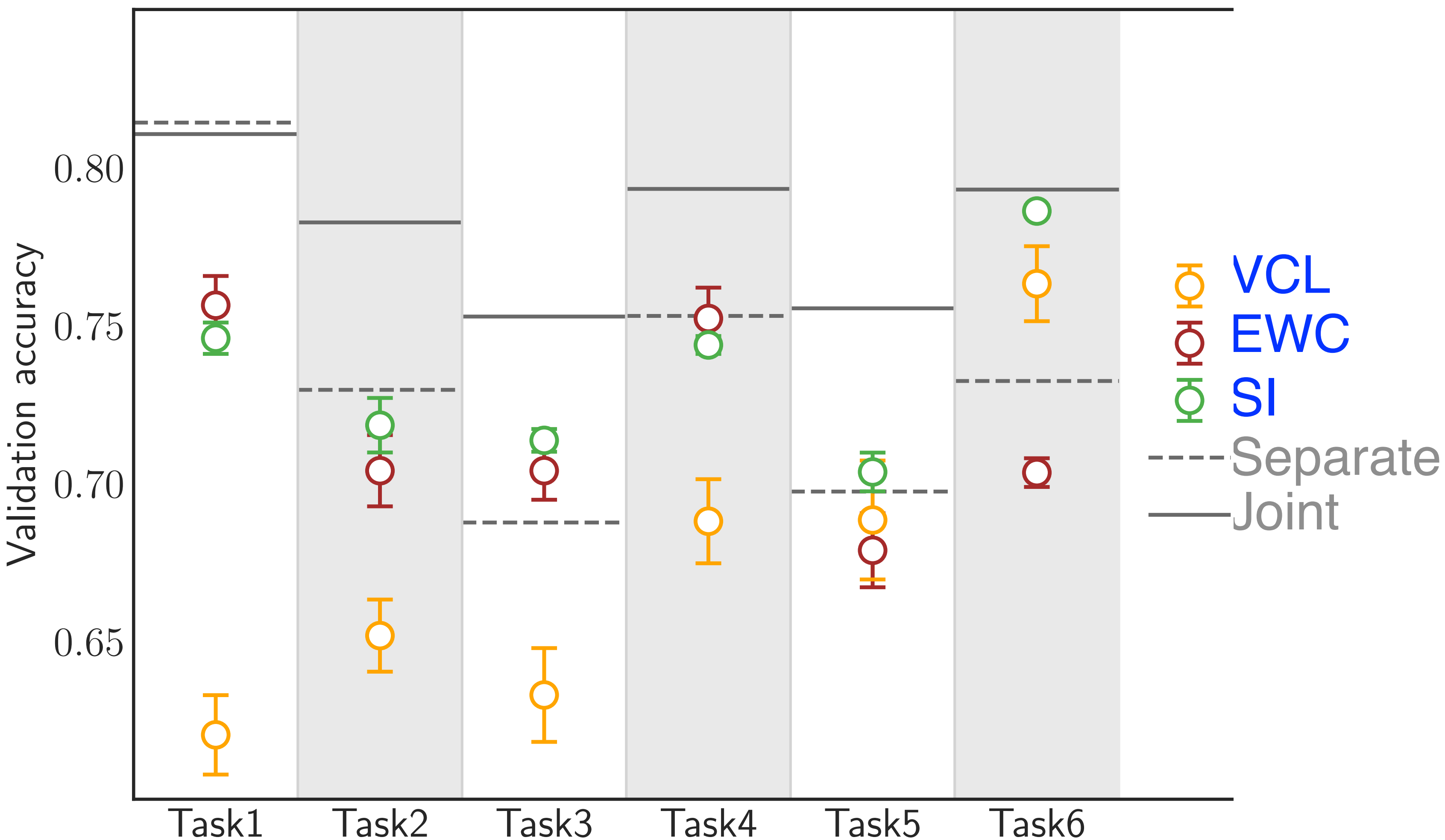
For DNN, with a specific Gaussian approximation, we obtain Neural Tangent Kernel

1. Khan et al. "Approximate Inference Turns Deep Networks into Gaussian Processes." *NeurIPS* (2019).

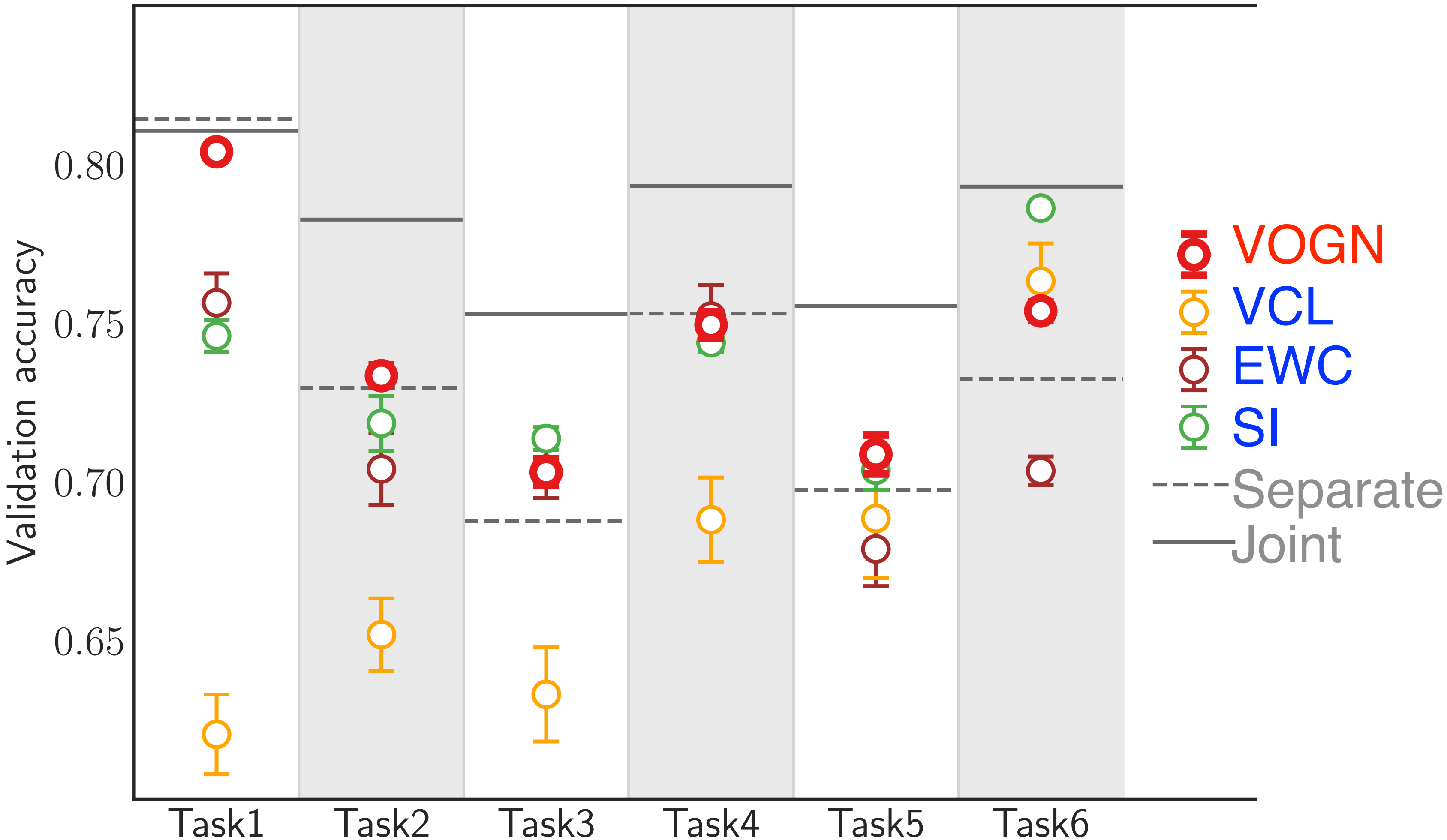
Deep Learning with Bayesian Principles

- Bayesian principles as common principles
  - By computing “posterior approximations”
- Derive many existing algorithms,
  - Deep Learning (SGD, RMSprop, Adam)
  - Exact Bayes, Laplace, Variational Inference, etc
- Design new deep-learning algorithms
  - Uncertainty, data importance, life-long learning
- Impact: Many learning-algorithms with a common set of principles.

Principle is Broken: Better Approximation don't give better results!

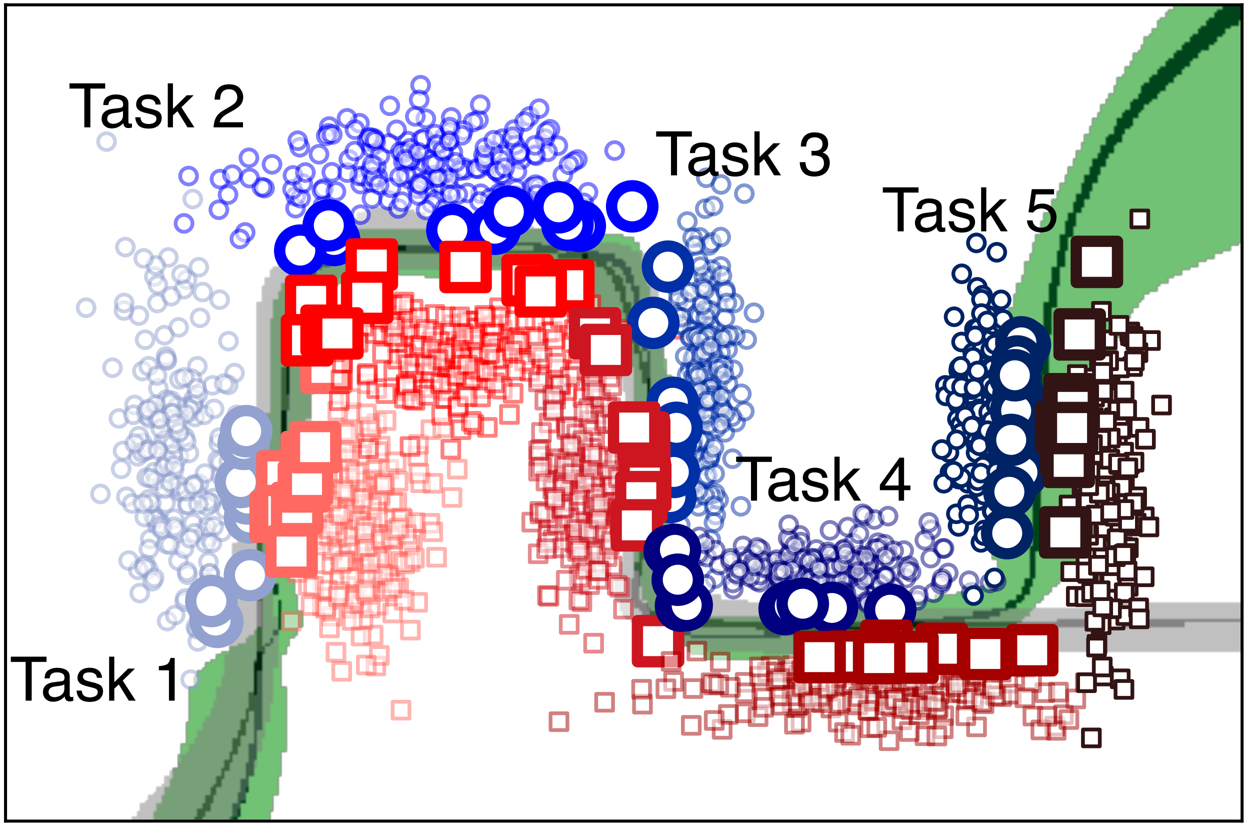


VOGN improves the gap

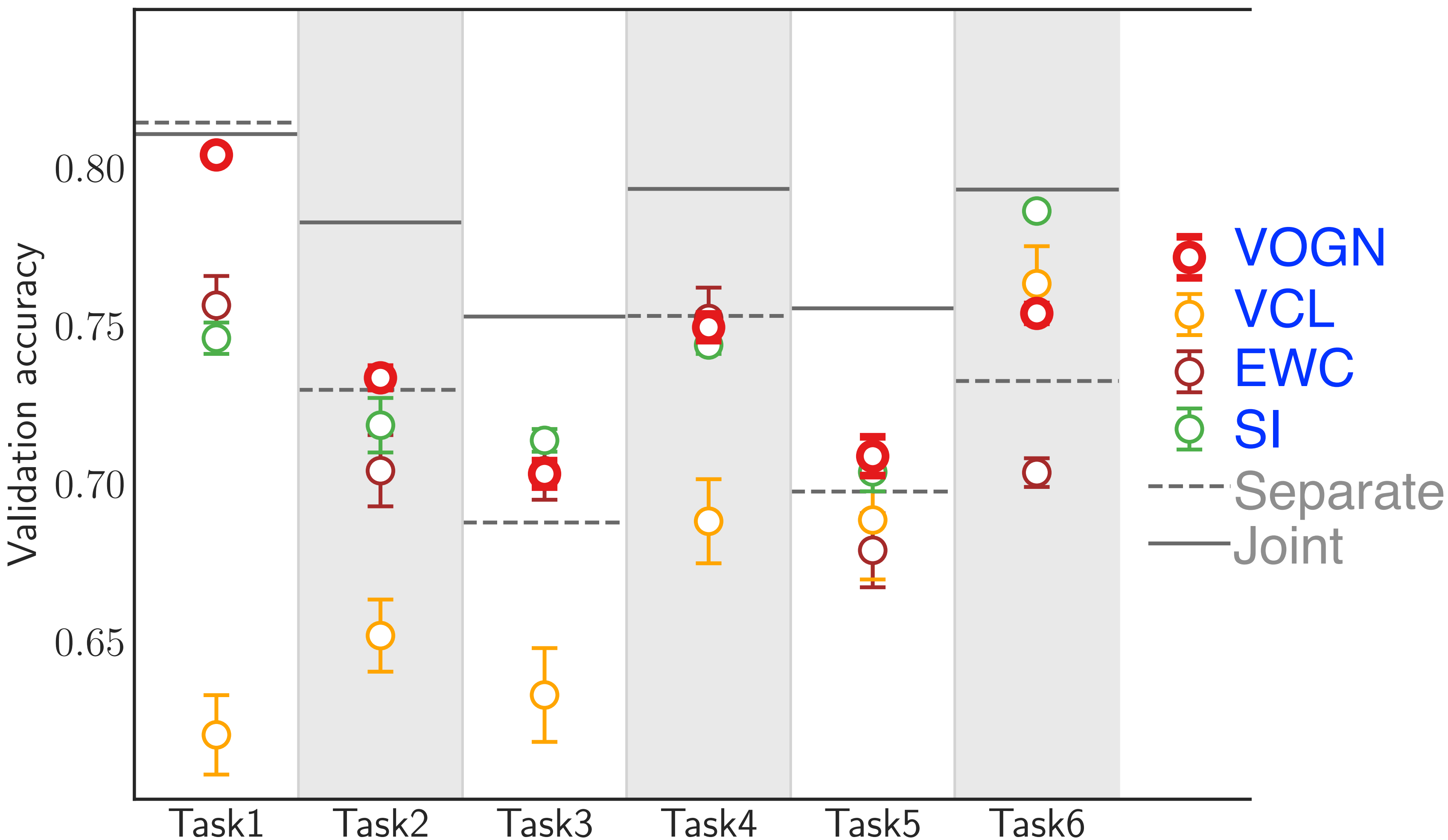


Functional Regularization of Memorable Past (FROMP)

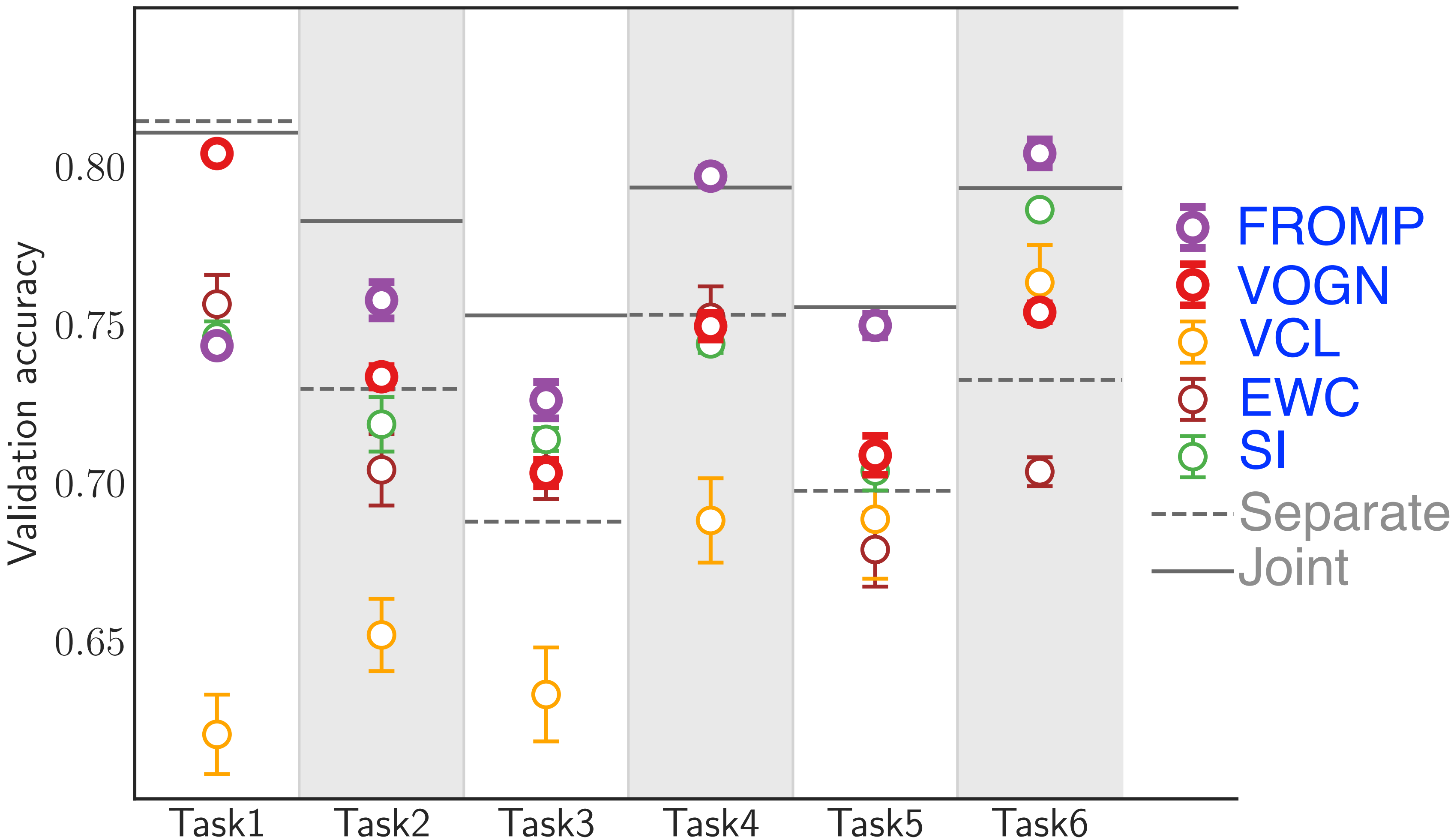
Identify, memorize, and regularize the past using Laplace Approximation (similar to EWC)



FROMP improves over EWC!

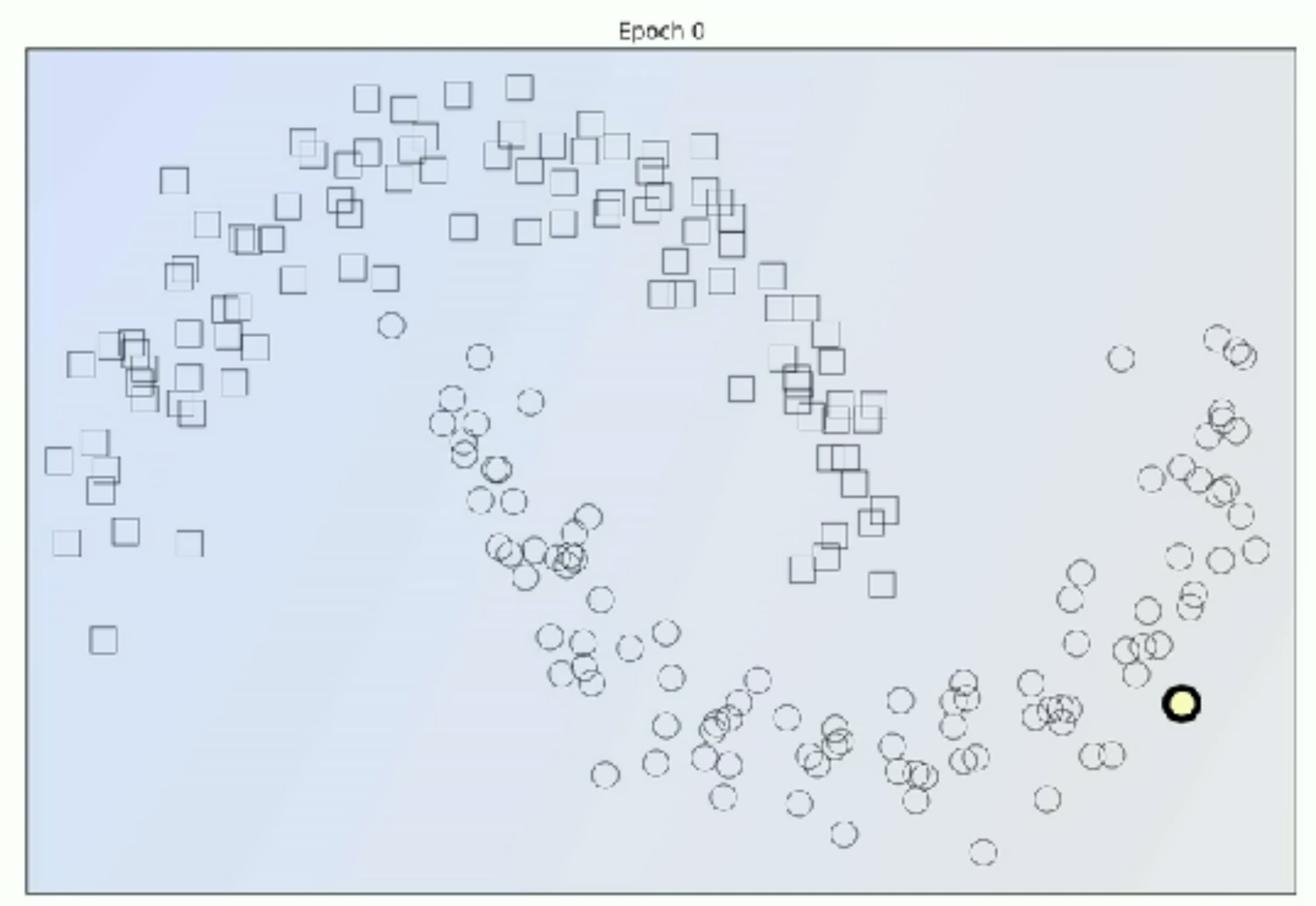


FROMP improves over EWC!



Active Deep Learning

Select “Important” examples while training with Adam



(By Roman Bachmann)

Deep Learning with Bayesian Principles

- Bayesian principles as common principles
  - By computing “posterior approximations”
- Derive many existing algorithms,
  - Deep Learning (SGD, RMSprop, Adam)
  - Exact Bayes, Laplace, Variational Inference, etc
- Design new deep-learning algorithms
  - Uncertainty, data importance, life-long learning
- Impact: Many learning-algorithms with a common set of principles.

Open Challenges

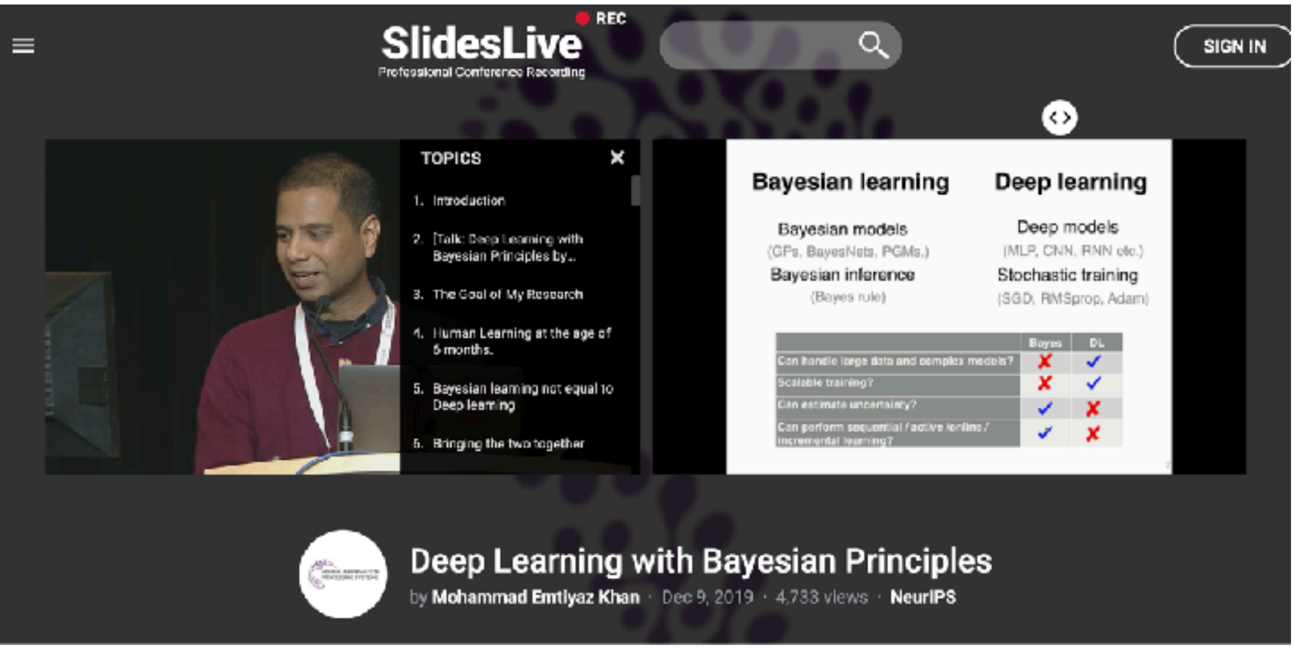
- Deep Learning + Bayes Learning
  - Principles of “trial and error” and “bayes” together
- How to achieve Life-long deep learning?
- How to compute better posterior approx?
- How to compute higher-order gradients?

Towards Life-long learning

- For life-long learning, we need
  - Perception: how you want to see the world?
  - Action: what you want to see in the world?
- Posterior approximation connects the two
  - Models are representation of the world
  - Approximations are representation of the model
  - They help us learn the model through actions
  - Act to appropriately “fill” the data space

1. Friston, K. "The free-energy principle: a unified brain theory?." *Nature neuroscience* (2010)

NeurIPS 2019 Tutorial on
“Deep Learning with Bayesian Principles”



https://slideslive.com/38921489/deep-learning-with-bayesian-principles

With a significant help from

Roman Bachmann (RIKEN-AIP)

Xiangming Meng (RIKEN-AIP)

Learning-Algorithms from Bayesian Principles

Coming soon!

A preliminary version is at

https://emtiyaz.github.io/papers/learning_from_bayes.pdf

Havard Rue (KAUST)

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.

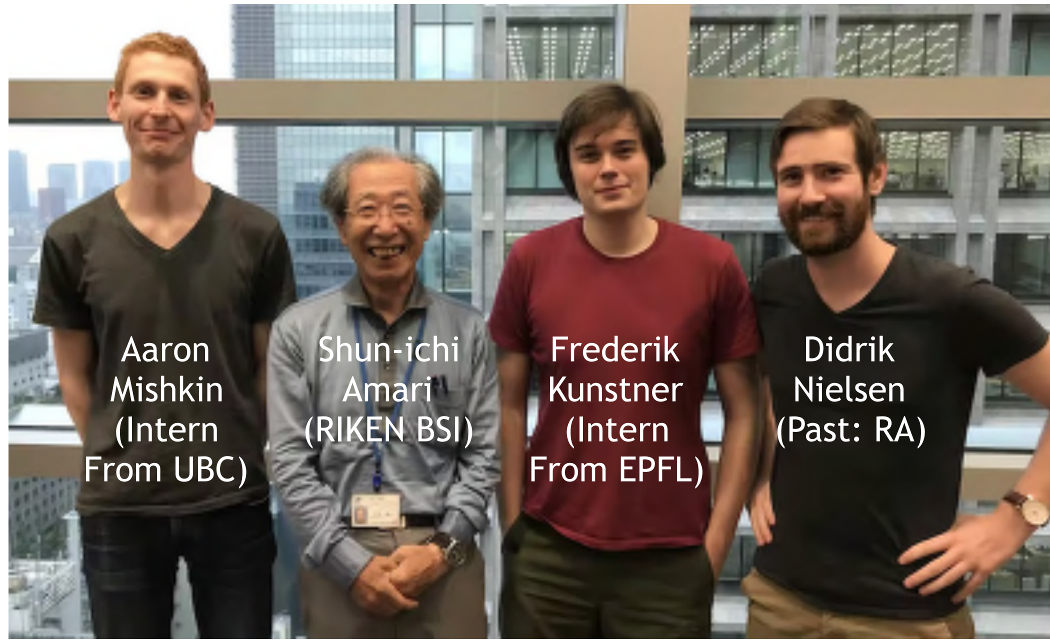
This paper discusses some recent methods that generalize the use of natural gradients to such large and complex non-conjugate models. We show that, for exponential-family approximations, a duality between their natural and expectation parameter spaces enables a simple natural gradient update. The resulting 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

Acknowledgements

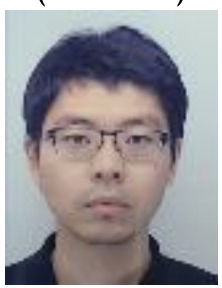
Slides, papers, & code are at emtiyaz.github.io

Wu Lin
(Past: RA)

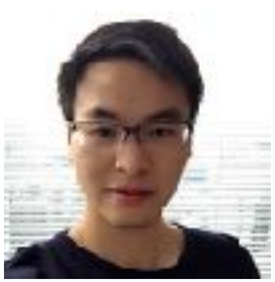
Nicolas Hubacher
(Past: RA)



Masashi Sugiyama
(Director RIKEN-AIP)



Voot Tangkaratt
(Postdoc, RIKEN-AIP)



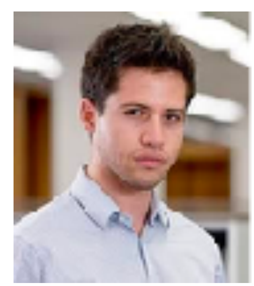
Zuozhu Liu
(Intern from SUTD)

RAIDEN

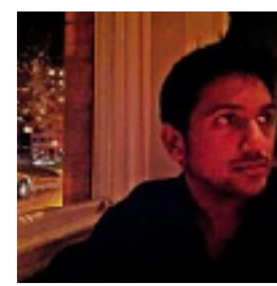
External Collaborators

Mark Schmidt
(UBC)

Reza Babanezhad
(UBC)



Yarin Gal
(UOxford)



Akash Srivastava
(UEdinburgh)

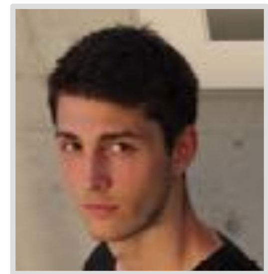
Acknowledgements

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

Kazuki Osawa (Tokyo Tech)

Rio Yokota (Tokyo Tech)

Anirudh Jain (Intern from IIT-ISM, India)



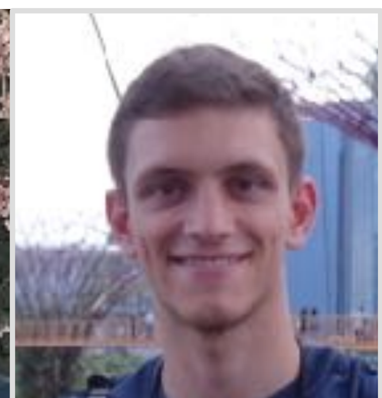
Runa Eschenhagen (Intern from University of Osnabruck)

Siddharth Swaroop (University of Cambridge)

Rich Turner (University of Cambridge)

Alexander Immer (Intern from EPFL)

Ehsan Abedi (Intern from EPFL)



Maciej Korzepa (Intern from DTU)

Pierre Alquier (RIKEN AIP)

Havard Rue (KAUST)

Approximate Bayesian Inference Team