

The Bayesian Learning Rule

Mohammad **Emtiyaz** Khan

RIKEN Center for AI Project, Tokyo

<http://emtiyaz.github.io>



AI that learn like humans

Quickly adapt to learn new skills, throughout their lives

Human Learning at
the age of 6 months.

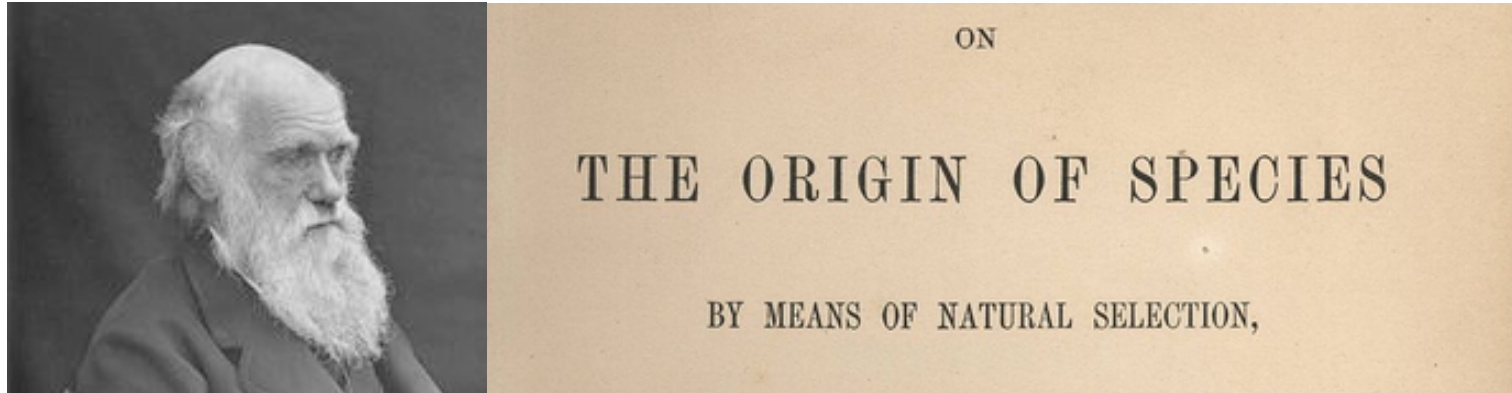


Converged at the
age of 12 months



Transfer
skills
at the age
of 14
months





The Origin of Algorithms

What are the common principles behind popular algorithms?

Principles of “good” algorithms?

- Information Geometry of Bayes
 - To unify/generalize/improve learning-algorithms
 - Optimize for “posterior approximations”
- Bayesian Learning rule (BLR)
 - Derive many algorithms from optimization, deep learning, and Bayesian inference
- Natural Gradients are Everywhere!

Bayesian Learning Rule

New information as natural
gradients

Bayesian learning rule

See Table 1 in Khan and Rue, 2021

Learning Algorithm	Posterior Approx.	Natural-Gradient Approx.	Sec.
Optimization Algorithms			
Gradient Descent	Gaussian (fixed cov.)	Delta method	1.3
Newton's method	Gaussian	—“—	1.3
Multimodal optimization _(New)	Mixture of Gaussians	—“—	3.2
Deep-Learning Algorithms			
Stochastic Gradient Descent	Gaussian (fixed cov.)	Delta method, stochastic approx.	4.1
RMSprop/Adam	Gaussian (diagonal cov.)	Delta method, stochastic approx., Hessian approx., square-root scaling, slow-moving scale vectors	4.2
Dropout	Mixture of Gaussians	Delta method, stochastic approx., responsibility approx.	4.3
STE	Bernoulli	Delta method, stochastic approx.	4.5
Online Gauss-Newton (OGN) _(New)	Gaussian (diagonal cov.)	Gauss-Newton Hessian approx. in Adam & no square-root scaling	4.4
Variational OGN _(New)	—“—	Remove delta method from OGN	4.4
BayesBiNN _(New)	Bernoulli	Remove delta method from STE	4.5
Approximate Bayesian Inference Algorithms			
Conjugate Bayes	Exp-family	Set learning rate $\rho_t = 1$	5.1
Laplace's method	Gaussian	Delta method	4.4
Expectation-Maximization	Exp-Family + Gaussian	Delta method for the parameters	5.2
Stochastic VI (SVI)	Exp-family (mean-field)	Stochastic approx., local $\rho_t = 1$	5.3
VMP	—“—	$\rho_t = 1$ for all nodes	5.3
Non-Conjugate VMP	—“—	—“—	5.3
Non-Conjugate VI _(New)	Mixture of Exp-family	None	5.4

Principle of Trial-and-Error

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

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

The diagram illustrates the components of the loss function. A blue arrow points from the word 'Loss' to the symbol $\ell(\mathcal{D}, \theta)$. Another blue arrow points from the word 'Data' to the symbol \mathcal{D} . A third blue arrow points from the word 'Model Params' to the symbol θ . A fourth blue arrow points from the word 'Deep Network' to the function $f_{\theta}(x_i)$.

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


We will derive them as special instances of a rule exploiting information geometry of Bayes.

Bayesian Learning

Bayes [1]: $\mathbb{E}_q[\text{log-lik}] + \text{KL}(q||\text{prior})$

Generalized Approx Bayes:

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

 Entropy

Posterior approximation (expo-family)

Geometry of Exponential Family

We will exploit the geometry of “minimal” exp-family

Natural
parameters

Sufficient
Statistics

Expectation
parameters

$$q(\theta) \propto \exp \left[\lambda^\top T(\theta) \right]$$

$$\mu := \mathbb{E}_q[T(\theta)]$$

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

Gaussian distribution

$$q(\theta) := \mathcal{N}(\theta|m, S^{-1})$$

Natural parameters

$$\lambda := \{Sm, -S/2\}$$

Expectation parameters

$$\mu := \{\mathbb{E}_q(\theta), \mathbb{E}_q(\theta \theta^\top)\}$$

The Bayesian Learning Rule

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

↑ Posterior approximation (expo-family)

↑ Entropy

Bayesian Learning Rule [1,2] (natural-gradient descent)

Natural and Expectation parameters of q

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

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

↑ Old belief ↑ New information = natural gradients

Exploiting posterior's information geometry to derive existing algorithms as special instances by approximating q and natural gradients.

1. Khan and Rue, The Bayesian Learning Rule, arXiv, <https://arxiv.org/abs/2107.04562>, 2021

2. Khan and Lin. "Conjugate-computation variational inference...." Alstats (2017).

Warning!

- This natural gradient might be different from the one what we (often) encounter in machine learning for Maximum-Likelihood
 - In MLE, the loss is the negative log probability distribution

$$\min_{\theta} -\log q(\theta) \Rightarrow F(\theta)^{-1} \nabla \log q(\theta)$$

- Here, θ , loss and distribution are two different entities, even possible unrelated

$$\min_q \mathbb{E}_q[\ell(\theta)] - \mathcal{H}(q) \Rightarrow F(\lambda)^{-1} \nabla_{\lambda} \mathbb{E}_q[\ell(\theta)]$$

Gradient Descent from Bayesian Learning Rule

(Euclidean) gradients as natural
gradients

Bayesian learning rule:

Learning Algorithm	Posterior Approx.	Natural-Gradient Approx.	Sec.
Optimization Algorithms			
Gradient Descent	Gaussian (fixed cov.)	Delta method	1.3
Newton's method	Gaussian	—"—	1.3
Multimodal optimization _(New)	Mixture of Gaussians	—"—	3.2
Deep-Learning Algorithms			
Stochastic Gradient Descent	Gaussian (fixed cov.)	Delta method, stochastic approx.	4.1
RMSprop/Adam	Gaussian (diagonal cov.)	Delta method, stochastic approx., Hessian approx., square-root scaling, slow-moving scale vectors	4.2
Dropout	Mixture of Gaussians	Delta method, stochastic approx., responsibility approx.	4.3
STE	Bernoulli	Delta method, stochastic approx.	4.5
Online Gauss-Newton (OGN) _(New)	Gaussian (diagonal cov.)	Gauss-Newton Hessian approx. in Adam & no square-root scaling	4.4
Variational OGN _(New)	—"—	Remove delta method from OGN	4.4
BayesBiNN _(New)	Bernoulli	Remove delta method from STE	4.5
Approximate Bayesian Inference Algorithms			
Conjugate Bayes	Exp-family	Set learning rate $\rho_t = 1$	5.1
Laplace's method	Gaussian	Delta method	4.4
Expectation-Maximization	Exp-Family + Gaussian	Delta method for the parameters	5.2
Stochastic VI (SVI)	Exp-family (mean-field)	Stochastic approx., local $\rho_t = 1$	5.3
VMP	—"—	$\rho_t = 1$ for all nodes	5.3
Non-Conjugate VMP	—"—	—"—	5.3
Non-Conjugate VI _(New)	Mixture of Exp-family	None	5.4

Gradient Descent from BLR

$$\text{GD: } \theta \leftarrow \theta - \rho \nabla_{\theta} \ell(\theta)$$

$$\text{BLR: } m \leftarrow m - \rho \nabla_m \ell(m)$$

“Global” to “local”
(the delta method)

$$\mathbb{E}_q[\ell(\theta)] \approx \ell(m)$$

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

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

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

Bayesian learning rule:

Put the expectation (Bayes) back in and use the Bayesian averaging.

Learning Algorithm	Posterior Approx.	Natural-Gradient Approx.	Sec.
Optimization Algorithms			
Gradient Descent	Gaussian (fixed cov.)	Delta method	1.3
Newton's method	Gaussian	—"—"	1.3
Multimodal optimization <small>(New)</small>	Mixture of Gaussians	—"—"	3.2
Deep-Learning Algorithms			
Stochastic Gradient Descent	Gaussian (fixed cov.)	Delta method, stochastic approx.	4.1
RMSprop/Adam	Gaussian (diagonal cov.)	Delta method, stochastic approx., Hessian approx., square-root scaling, slow-moving scale vectors	4.2
Dropout	Mixture of Gaussians	Delta method, stochastic approx., responsibility approx.	4.3
STE	Bernoulli	Delta method, stochastic approx.	4.5
Online Gauss-Newton (OGN) <small>(New)</small>	Gaussian (diagonal cov.)	Gauss-Newton Hessian approx. in Adam & no square-root scaling	4.4
Variational OGN <small>(New)</small>	—"—"	Remove delta method from OGN	4.4
BayesBiNN <small>(New)</small>	Bernoulli	Remove delta method from STE	4.5
Approximate Bayesian Inference Algorithms			
Conjugate Bayes	Exp-family	Set learning rate $\rho_t = 1$	5.1
Laplace's method	Gaussian	Delta method	4.4
Expectation-Maximization	Exp-Family + Gaussian	Delta method for the parameters	5.2
Stochastic VI (SVI)	Exp-family (mean-field)	Stochastic approx., local $\rho_t = 1$	5.3
VMP	—"—"	$\rho_t = 1$ for all nodes	5.3
Non-Conjugate VMP	—"—"	—"—"	5.3
Non-Conjugate VI <small>(New)</small>	Mixture of Exp-family	None	5.4

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).
3. Lin et al. "Handling the positive-definite constraints in the BLR." *ICML* (2020).

Newton's method from Bayesian Learning Rule

(Gradient, Hessian) as natural
gradients

Newton's Method from BLR

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

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

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

$$\lambda \leftarrow (1 - \rho) \nabla_{\mu} \mathcal{H}(q) + \rho \nabla_{\mu} \mathcal{H}(q) \quad -\nabla_{\mu} \mathcal{H}(q) = \lambda$$

Derived by choosing a **multivariate Gaussian**

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

Natural parameters $\lambda := \{Sm, -S/2\}$

Expectation parameters $\mu := \{\mathbb{E}_q(\theta), \mathbb{E}_q(\theta\theta^\top)\}$

Newton's Method from BLR

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

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

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

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

Delta Method

$$\mathbb{E}_q[\ell(\theta)] \approx \ell(m)$$

Express in terms of gradient and Hessian of loss:

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

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

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

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

RMSprop/Adam from BLR

RMSprop

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

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

BLR for Gaussian approx

$$S \leftarrow (1 - \rho)S + \rho(H_\theta)$$

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

To get RMSprop, make the following choices

- Restrict covariance to be diagonal
- 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])

Practical DL with Bayes

RMSprop

$$g \leftarrow \hat{\nabla} \ell(\theta)$$

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

$$\theta \leftarrow \theta - \alpha(\sqrt{s} + \delta)^{-1} g$$

BLR variant called VOGN

$$g \leftarrow \hat{\nabla} \ell(\theta), \text{ where } \theta \sim \mathcal{N}(m, \sigma^2)$$

$$s \leftarrow (1 - \rho)s + \rho(\sum_i g_i^2)$$

$$m \leftarrow m - \alpha(s + \gamma)^{-1} \nabla_{\theta} \ell(\theta)$$

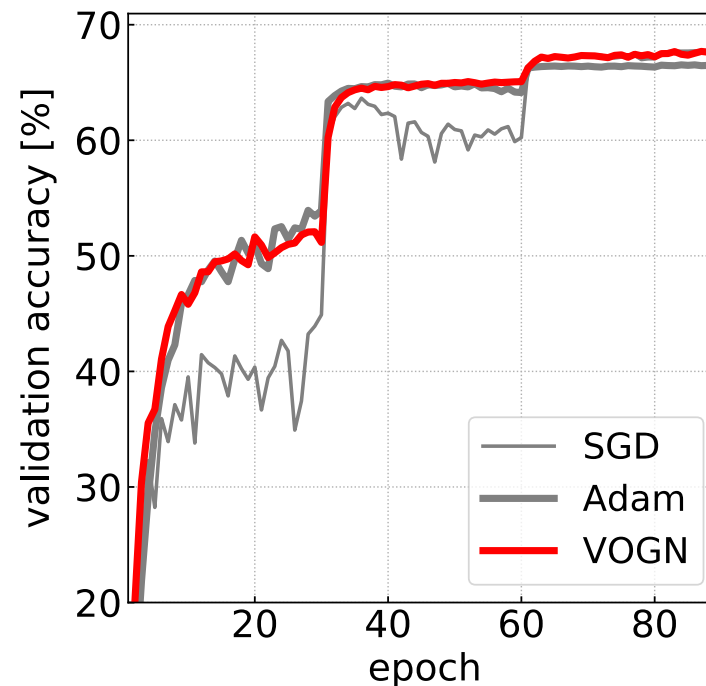
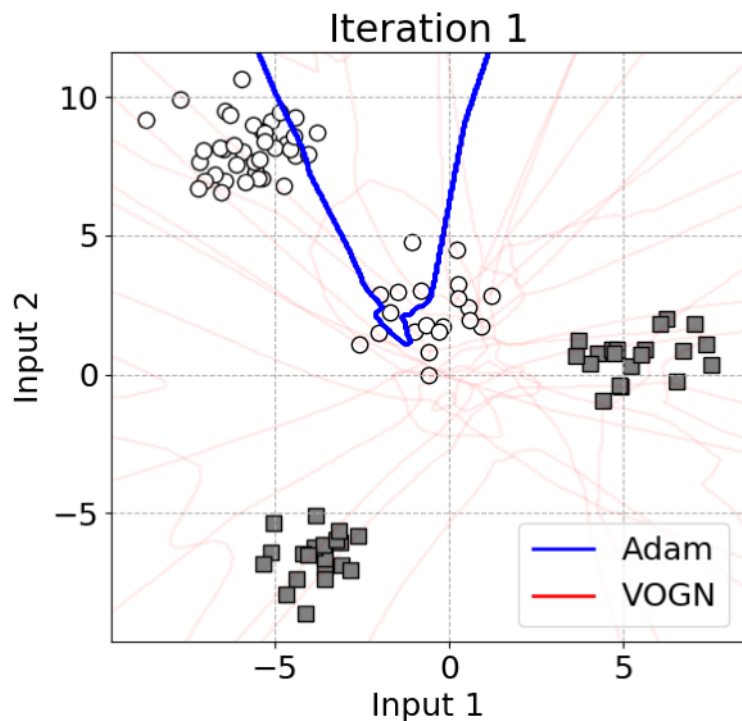
$$\sigma^2 \leftarrow (s + \gamma)^{-1}$$

Available at <https://github.com/team-approx-bayes/dl-with-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).
3. Lin et al. "Handling the positive-definite constraints in the BLR." *ICML* (2020).

Uncertainty of Deep Nets

VOGN: A modification of Adam with similar performance on ImageNet, but better uncertainty



Code available at <https://github.com/team-approx-bayes/dl-with-bayes>

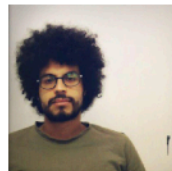
1. Khan, et al. "Fast and scalable Bayesian deep learning by weight-perturbation in Adam." *ICML* (2018).
2. Osawa et al. "Practical Deep Learning with Bayesian Principles." *NeurIPS* (2019).

BLR variant [3] got 1st prize in NeurIPS 2021 Approximate Inference Challenge

Watch **Thomas Moellenhoff's** talk at <https://www.youtube.com/watch?v=LQInIN5EU7E>.

Mixture-of-Gaussian Posteriors with an Improved Bayesian Learning Rule

Thomas Möllenhoff¹, Yuesong Shen², Gian Maria Marconi¹
Peter Nickl¹, Mohammad Emtiyaz Khan¹



¹ Approximate Bayesian Inference Team
RIKEN Center for AI Project, Tokyo, Japan

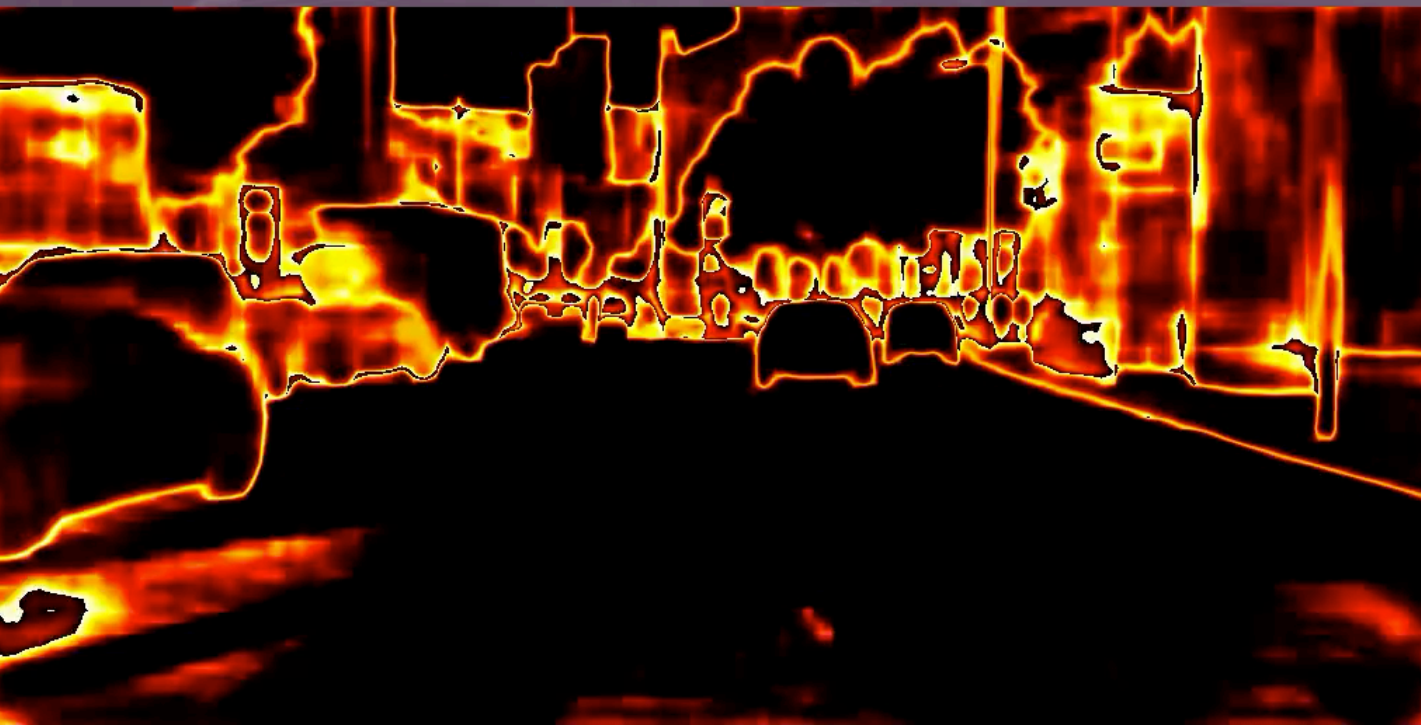
² Computer Vision Group
Technical University of Munich, Germany

Dec 14th, 2021 — NeurIPS Workshop on Bayesian Deep Learning

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).
3. Lin et al. "Handling the positive-definite constraints in the BLR." *ICML* (2020).



Image
Segmentation



Uncertainty
(entropy of
class probs)

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 use the Delta method (a local approximation) $\mathbb{E}_q[\ell(\theta)] \approx \ell(m)$
- Then, to improve DL algorithms, we just need to add some “global” touch by relaxing the local approximation

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

Our use of natural-gradients here is not a matter of choice. In fact, natural-gradients are inherently present in *all solutions of the Bayesian objective* in Eq. 2. For example, a solution of Eq. 2 or equivalently a fixed point of Eq. 3, satisfies the following,

$$\nabla_{\mu} \mathbb{E}_{q_*} [\bar{\ell}(\boldsymbol{\theta})] = \nabla_{\mu} \mathcal{H}(q_*), \text{ which implies } \tilde{\nabla}_{\lambda} \mathbb{E}_{q_*} [-\bar{\ell}(\boldsymbol{\theta})] = \boldsymbol{\lambda}_*, \quad (5)$$

for candidates with constant base-measure. This is obtained by setting the gradient of Eq. 2 to 0, then noting that $\nabla_{\mu} \mathcal{H}(q) = -\boldsymbol{\lambda}$ (App. B), and then interchanging ∇_{μ} by $\tilde{\nabla}_{\lambda}$ (because of Eq. 4). In other words, natural parameter of the best $q_*(\boldsymbol{\theta})$ is equal to the natural gradient of the expected negative-loss. The importance of natural-gradients is entirely missed in the Bayesian/variational inference literature, including textbooks, reviews, tutorials on this topic [Bishop, 2006, Murphy, 2012, Blei et al., 2017, Zhang et al., 2018a] where natural-gradients are often put in a special category.

We will show that natural gradients retrieve essential higher-order information about the loss landscape which are then assigned to appropriate natural parameters using Eq. 5. The information-matching is due to the presence of the entropy term there, which is an important quantity for the optimality of Bayes in general [Jaynes, 1982, Zellner, 1988, Littlestone and Warmuth, 1994, Vovk, 1990], and which is generally absent in non-Bayesian formulations (Eq. 1). The entropy term in general leads to exponential-weighting in Bayes' rule. In our context, it gives rise to natural-gradients and, as we will soon see, automatically determines the complexity of the derived algorithm through the complexity of the class of distributions \mathcal{Q} , yielding a principled way to develop new algorithms.

Overall, our work demonstrates the importance of natural-gradients and information geometry for algorithm design in ML. This is similar in spirit to Information Geometric Optimization [Ollivier et al., 2017], which focuses on the optimization of black-box, deterministic functions. In contrast, we derive generic learning algorithms by using the same Bayesian principles. The BLR we use is a generalization of the method proposed in Khan and Lin [2017], Khan and Nielsen [2018] specifically for approximate Bayesian inference. Here, we establish it as a general learning rule to derive many old and new learning algorithms, which include both Bayesian and non-Bayesian ones, way beyond its original proposal. We do not claim that these successful algorithms work well because they are derived from the BLR. Rather, we use the BLR to simply unravels the inherent Bayesian nature of these "good" algorithms. In this sense, the BLR can be seen as a variant of Bayes' rule, useful for generic algorithm design.

Principles of “good” algorithms?

- Information Geometry of Bayes
 - To unify/generalize/improve learning-algorithms
 - Optimize for “posterior approximations”
- Bayesian Learning rule (BLR)
 - Derive many algorithms from optimization, deep learning, and Bayesian inference
- Natural Gradients are Everywhere!

NeurIPS 2019 Tutorial

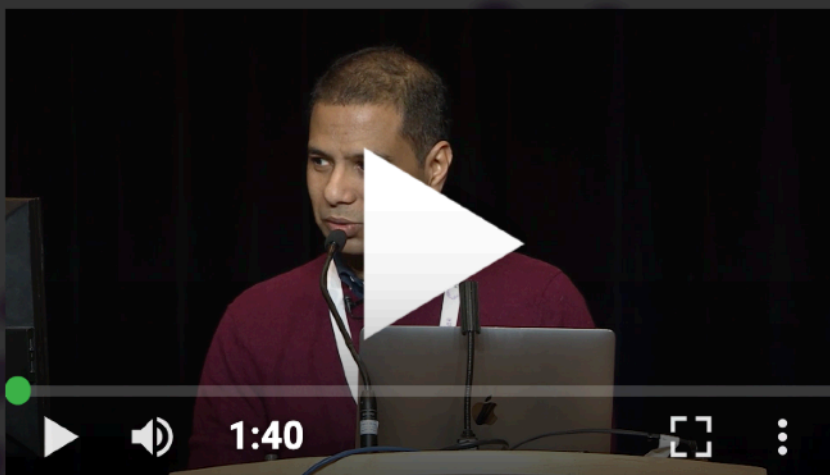
#NeurIPS 2019

Follow

Views 151 807

Presentations 263

Followers 200



Human Learning at the age of 6 months.



Deep Learning with Bayesian Principles

by **Mohammad Emtiyaz Khan** · Dec 9, 2019



Latest

Popular

...



FROM SYSTEM 1 DEEP LEARNING TO SYSTEM 2 DEEP LEARNING

Yoshua Bengio

December 11th - 2:15pm



50:00

From System 1 Deep Learning to System 2 Deep Learning

by [Yoshua Bengio](#)

17,953 views · Dec 11, 2019



NEURIPS WORKSHOP ON MACHINE LEARNING FOR CREATIVITY AND DESIGN 3.0 2

December 14th - 10:30am



1:30:00

NeurIPS Workshop on Machine Learning for Creativity and Design...

by [Aaron Hertzmann](#), [Adam Roberts](#), ...

9,654 views · Dec 14, 2019



DEEP LEARNING WITH BAYESIAN PRINCIPLES

Mohammad Emtiyaz Khan

December 9th - 8:30am



2:00:00

Deep Learning with Bayesian Principles

by [Mohammad Emtiyaz Khan](#)

8,084 views · Dec 9, 2019



EFFICIENT PROCESSING OF DEEP NEURAL NETWORK: FROM ALGORITHMS TO HARDWARE ARCHITECTURES

Vivienne See

December 9th - 11:15am



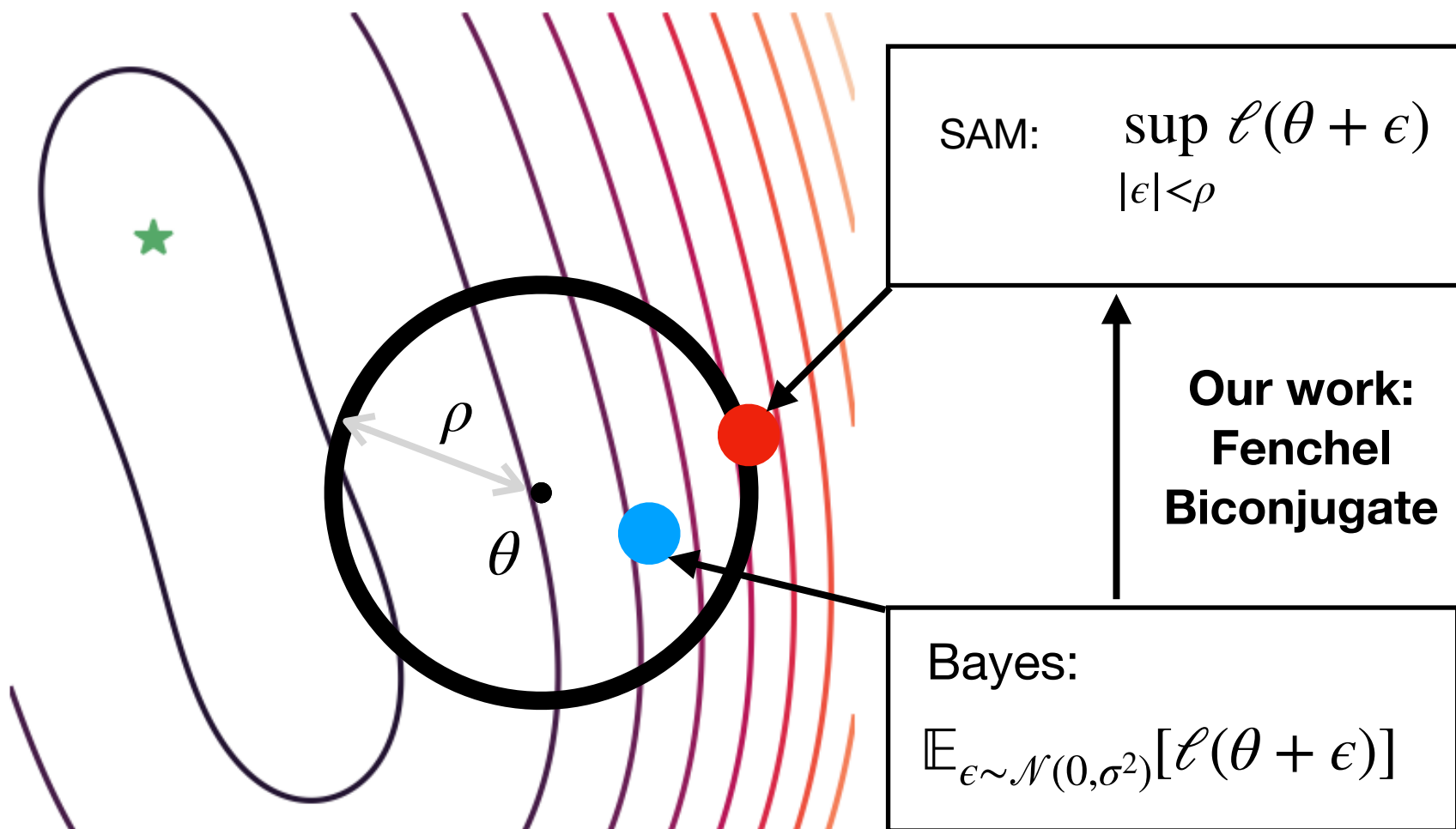
2:00:00

Efficient Processing of Deep Neural Network: from Algorithms to...

by [Vivienne See](#)

7,163 views · Dec 9, 2019

SAM as an Optimal relaxation of Bayes



1. Foret et al. Sharpness-Aware Minimization for Efficiently Improving Generalization, ICLR, 2021
2. Moellenhoff and Khan, SAM as an Optimal Relaxation of Bayes, Under review, 2022

What's Next

- Bayesian “Duality” Principle
 - The BLR unravels a duality perspective of good algorithms
 - Unifies many results from many fields
 - convex duality, Kernel methods, Bayesian nonparametric methods, Deep Learning, Robust statistics, and Information Geometry
 - Helps to solve the Adaptation problem

The Bayes-Duality Project

Toward AI that learns adaptively, robustly, and continuously, like humans



Emtiyaz Khan

Research director
(Japan side)

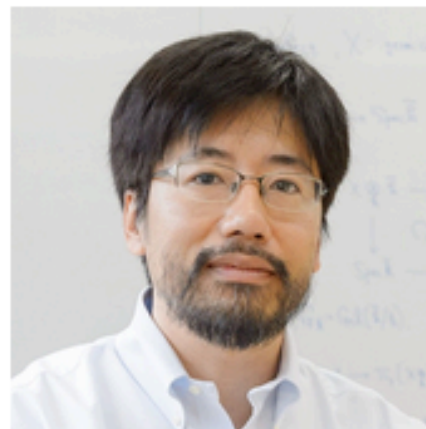
Approx-Bayes team at
RIKEN-AIP and OIST



Julyan Arbel

Research director
(France side)

Statify-team, Inria
Grenoble Rhône-Alpes



Kenichi Bannai

Co-PI (Japan side)

Math-Science Team at
RIKEN-AIP and Keio
University



Rio Yokota

Co-PI
(Japan side)

Tokyo Institute of
Technology

Received total funding of around **USD 3 million** through JST's CREST-ANR and Kakenhi Grants.

Approximate Bayesian Inference Team

<https://team-approx-bayes.github.io/>



Emtiyaz Khan
Team Leader



Thomas Möllenhoff
Research Scientist



Hugo Monzón Maldonado
Postdoc



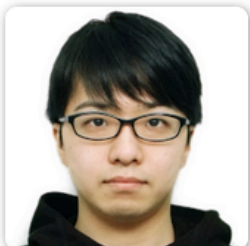
Happy Buzaaba
Postdoc



Erik Daxberger
Remote Collaborator
University of Cambridge



Paul Chang
Remote Collaborator
Aalto University



Keigo Nishida
Postdoc
RIKEN BDR



Gian Maria Marconi
Postdoc



Negar Safinianaini
Postdoc



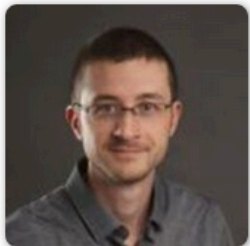
Lu Xu
Postdoc



Alexandre Piché
Remote Collaborator
MILA



Ang Mingliang
Remote Collaborator
National University of Singapore



Geoffrey Wolfer
Postdoc



Wu Lin
PhD Student
University of British Columbia



Peter Nickl
Research Assistant



Dharmesh Tailor
Remote Collaborator
University of Amsterdam