Presentation at EcoStat, June 5, 2022

The Bayesian Learning Rule

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

Slides at https://emtiyaz.github.io/papers/Jun5_2022_EcoStat.pdf
Summary of research at https://emtiyaz.github.io/papers/symposium_2021.pdf

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

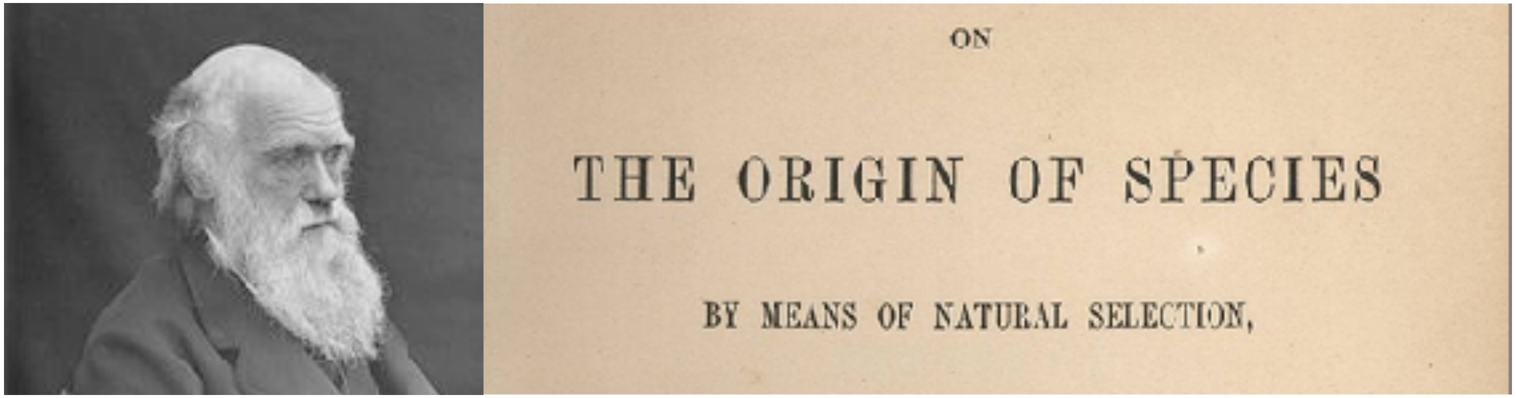
Transfer skills

at the age of 14 months

AI that learn like humans

Quickly adapt to learn new skills, throughout their lives

See Section 6 (discussion) in Khan and Rue, 2021



The Origin of Algorithms

A good algorithm must revise its *past* beliefs by using useful *future* information

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

Principles of “good” algorithms?

- Bayesian principles
  - To unify/generalize/improve learning-algorithms
  - By computing “posterior approximations”
- Bayesian Learning rule (BLR)
  - Derive many existing algorithms
  - Deep Learning (SGD, RMSprop, Adam)
  - Design new algorithms for uncertainty in DL
- Impact: Everything with the same principle

The Bayesian Learning Rule

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

Håvard Rue
CEMSE Division, KAUST
Thuwal, Saudi Arabia
haavard.rue@kaust.edu.sa

Abstract

We show that many machine-learning algorithms are specific instances of a single algorithm called the *Bayesian learning rule*. The rule, derived from Bayesian principles, yields a wide-range of algorithms from fields such as optimization, deep learning, and graphical models. This includes classical algorithms such as ridge regression, Newton's method, and Kalman filter, as well as modern deep-learning algorithms such as stochastic-gradient descent, RMSprop, and Dropout. The key idea in deriving such algorithms is to approximate the posterior using candidate distributions estimated by using natural gradients. Different candidate distributions result in different algorithms and further approximations to natural gradients give rise to variants of those algorithms. Our work not only unifies, generalizes, and improves existing algorithms, but also helps us design new ones.

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

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

A Bayesian Origin

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

Entropy

Posterior approximation (expo-family)

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

Using posterior's information geometry to balance new vs old information

1. Khan and Rue, The Bayesian Learning Rule, arXiv, https://arxiv.org/abs/2107.04562, 2021
2. Khan and Lin. "Conjugate-computation variational inference…." AIstats (2017).

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

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

See Section 1.3.1 in Khan and Rue, 2021

Gradient Descent from Bayes

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

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

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

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

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

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

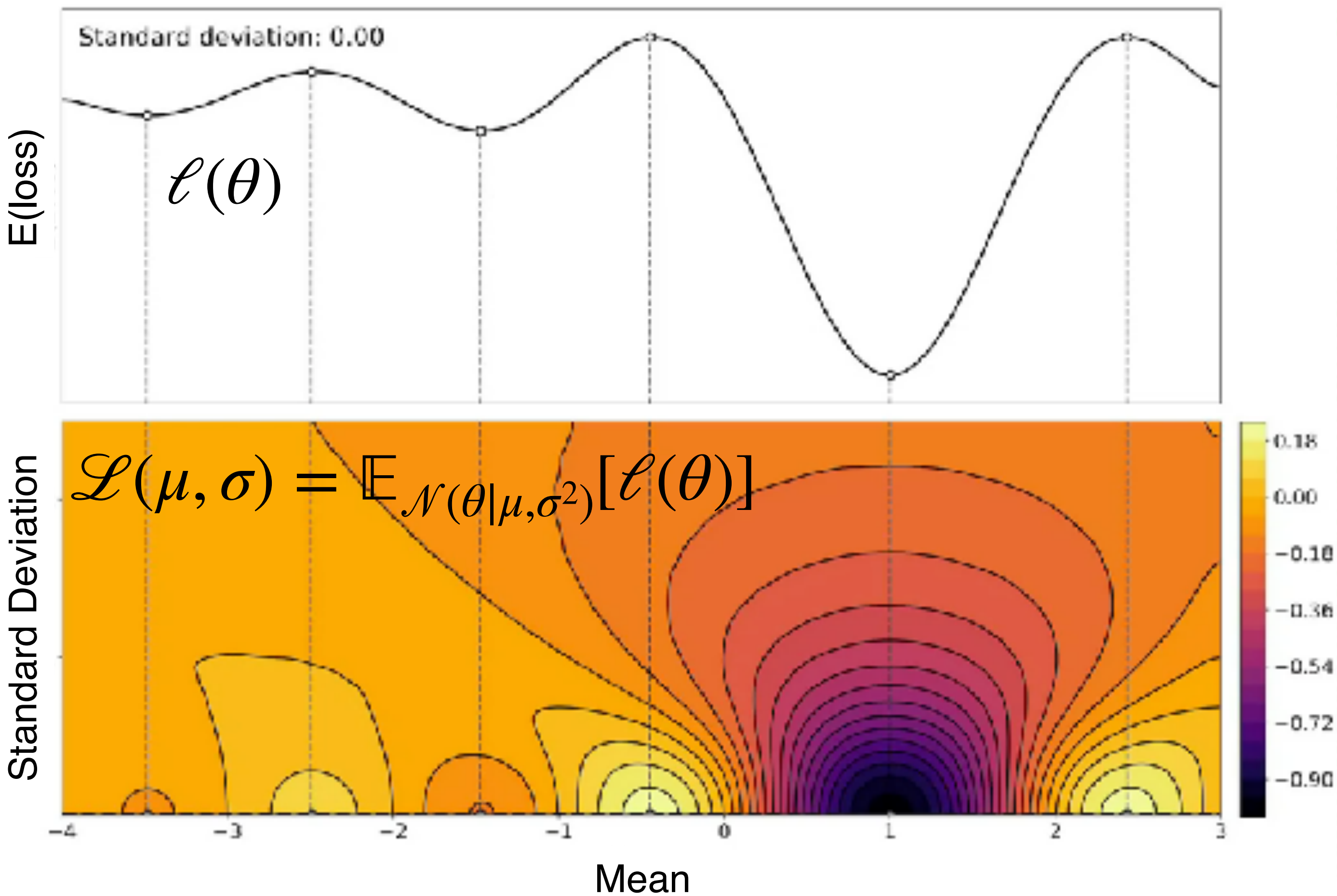
Learning Algorithm	Posterior Approx.	Natural-Gradient Approx.	Sec.
Optimization Algorithms			
Gradient Descent	Gaussian (fixed cov.)	Delta method	1.3
Newton's method	Gaussian	——"——	1.3
Multimodal optimization (New)	Mixture of Gaussians	——"——	3.2
Deep-Learning Algorithms			
Stochastic Gradient Descent	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

Put the expectation (Bayes) back in!

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).

See Section 1.2, Eq 2 in Khan and Rue, 2021

Bayes Objective



Instead of the original loss, optimize a different one (Gaussian convolution)

A popular idea of "implicit regularization" in DL [4], but also common in other fields (RL, search, robust optimization)

1. Zellner, A. "Optimal information processing and Bayes's theorem." *The American Statistician* (1988)
2. Many other: Bissiri, et al. (2016), Shawe-Taylor and Williamson (1997), Cesa-Bianchi and Lugosi (2006)
3. Huszar's blog, Evolution Strategies, Variational Optimisation and Natural ES (2017)
4. Smith et al., On the Origin of Implicit Regularization in Stochastic Gradient Descent, ICLR, 2021

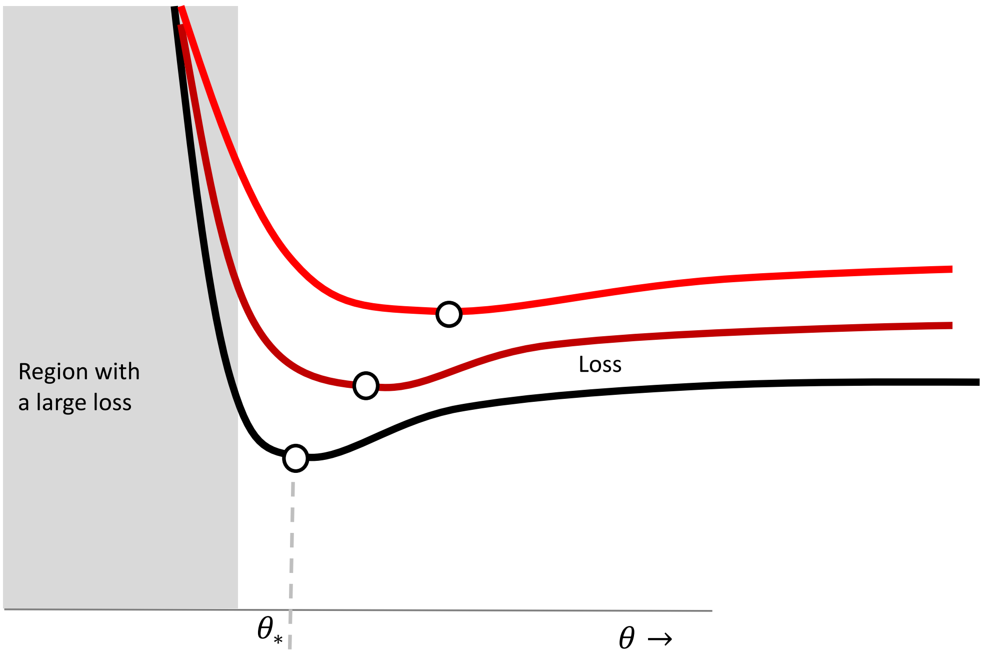
See Eq 25 in Khan and Rue, 2021 (Bonnet's theorem)

Bayes Prefers Flatter directions

GD: $\theta \leftarrow \theta - \rho \nabla_\theta \ell(\theta) \implies \nabla_\theta \ell(\theta_*) = 0$

BLR: $m \leftarrow m - \rho \nabla_m \mathbb{E}_q[\ell(\theta)] \implies \nabla_m \mathbb{E}_{q_*}[\ell(\theta)] = 0$

Bayesian solution injects "noise" which has a similar regularization effect to noise in Stochastic GD. It prefers "flatter" directions.

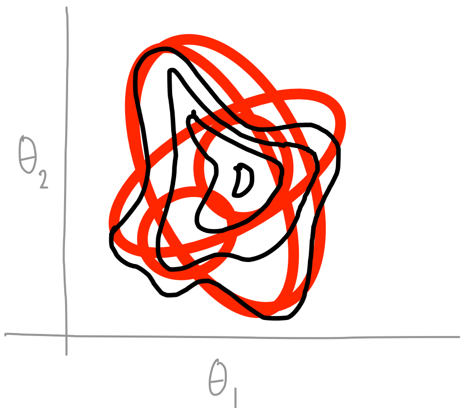


See Section 1.3 and 3.2 in Khan and Rue, 2021

Deriving Learning-Algorithms from the Bayesian Learning Rule

Posterior Approximation $\longleftrightarrow$ Learning-Algorithm

Complex ⟷ Simple



Bayes' rule

Mixture of Newton

Newton

Gradient Descent

See Section 1.3.2 in Khan and Rue, 2021

Newton's Method from Bayes

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

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

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

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

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

Derived by choosing a multivariate Gaussian

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

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).

See Section 1.3.2 in Khan and Rue, 2021

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 $m \leftarrow m - H_m^{-1}[\nabla_m \ell(m)]$

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

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

Express in terms of gradient and Hessian of loss:

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

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

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

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

BLR Variants

RMSprop

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

Variational Online Gauss-Newton (VOGN)

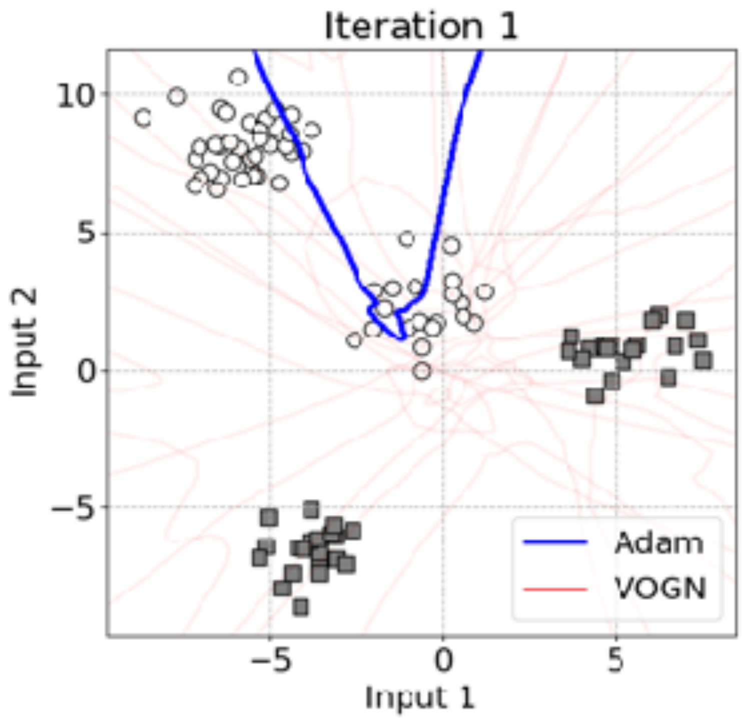
$$g \leftarrow \hat{\nabla}\ell(\theta), \text{ where } \theta \sim \mathcal{N}(m, \sigma^2)$$
$$s \leftarrow (1-\rho)s + \rho(\Sigma_i g_i^2)$$
$$m \leftarrow m - \alpha(s+\gamma)^{-1}\nabla_\theta \ell(\theta)$$
$$\sigma^2 \leftarrow (s+\gamma)^{-1}$$

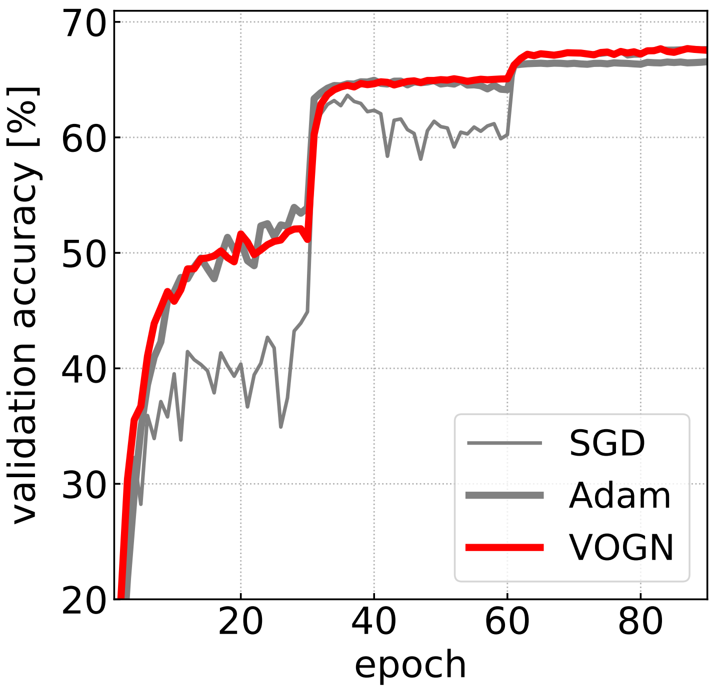
Available at https://github.com/team-approx-bayes/dl-with-bayes

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 but match the performance on ImageNet





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 Möllenhoff[1], Yuesong Shen[2], Gian Maria Marconi[1], Peter Nickl[1], Mohammad Emtiyaz Khan[1]

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).

See Section 4.4, Fig. 1 in Khan and Rue, 2021

Bayes leads to robust solutions

Avoiding sharp minima

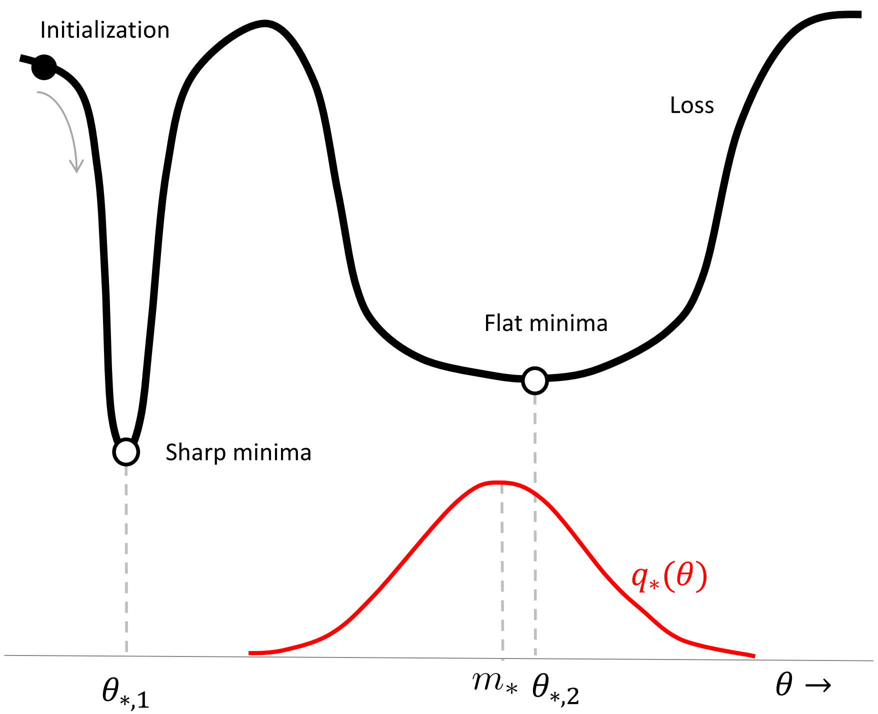
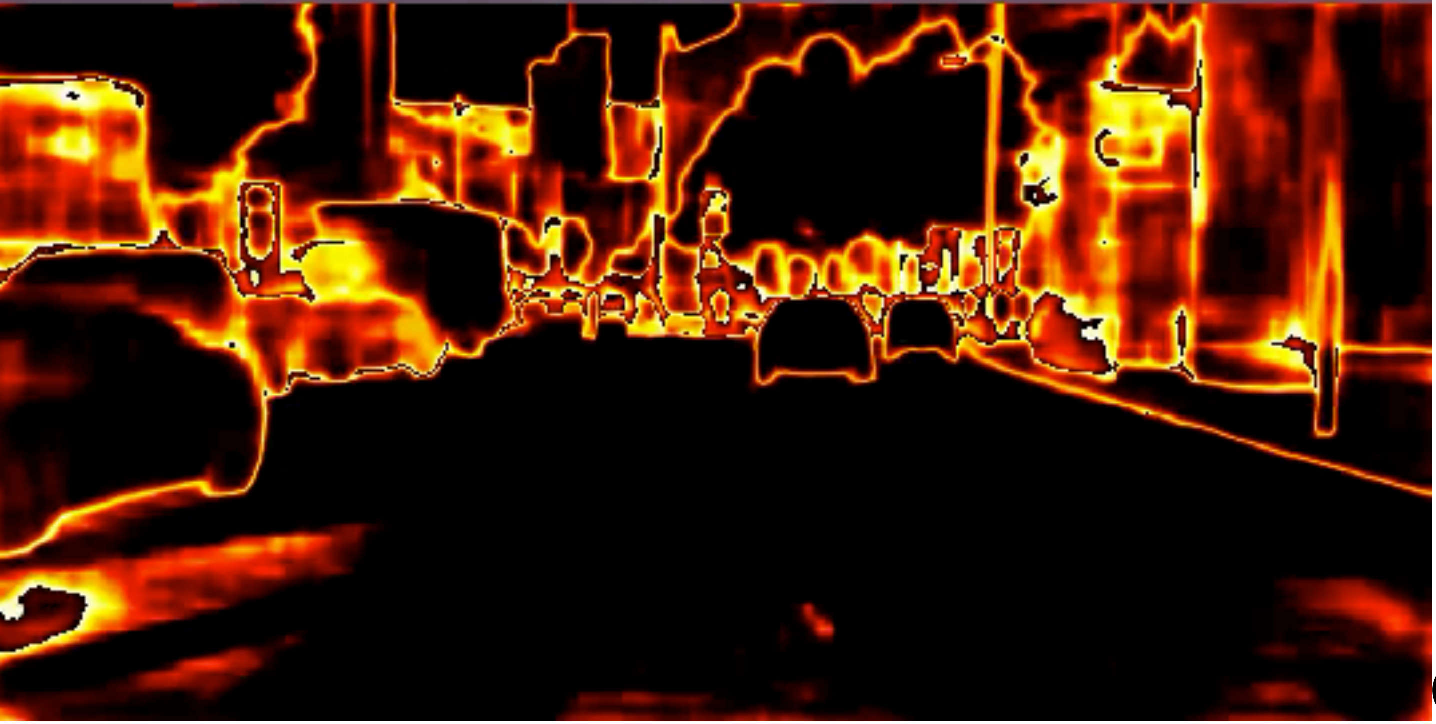
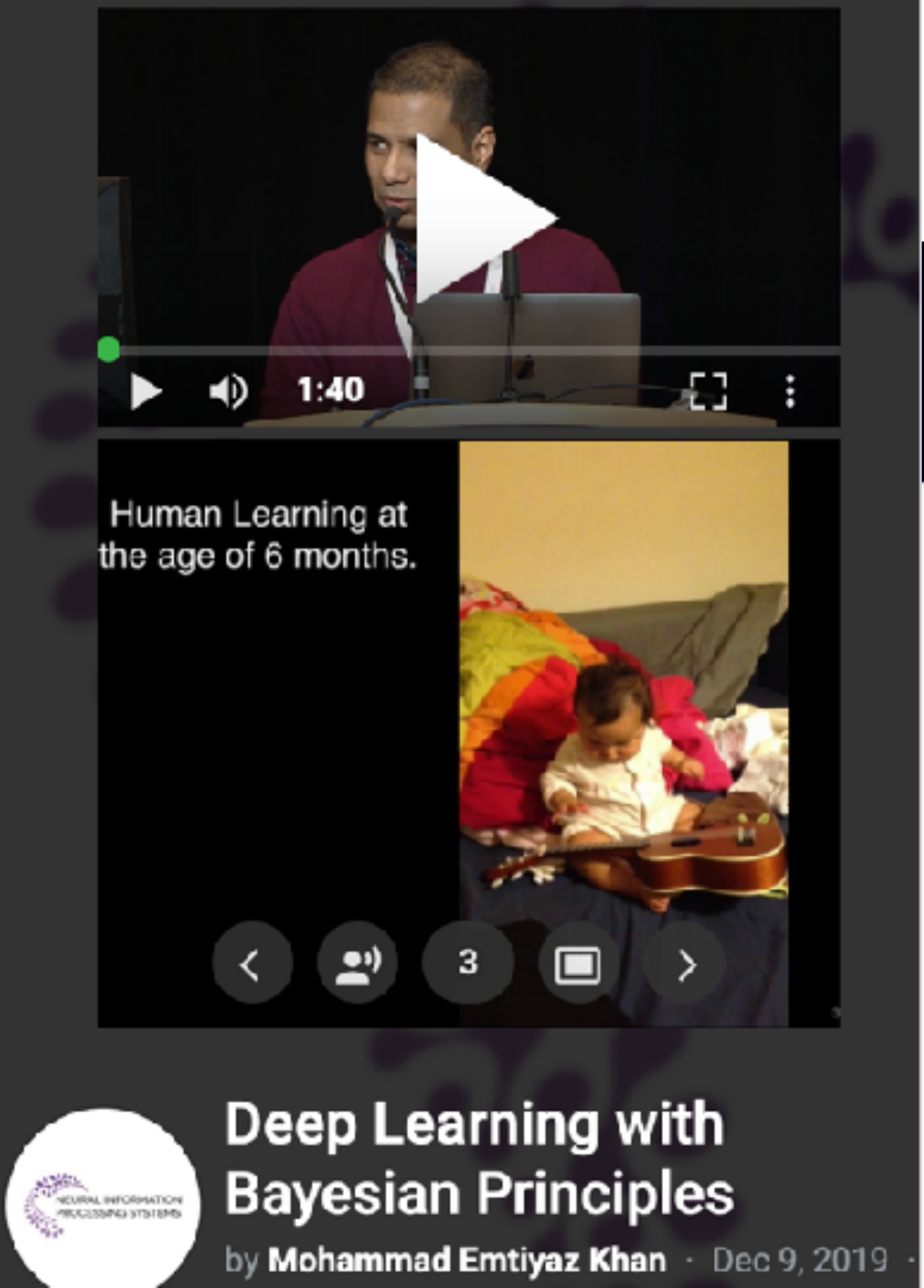


Image Segmentation

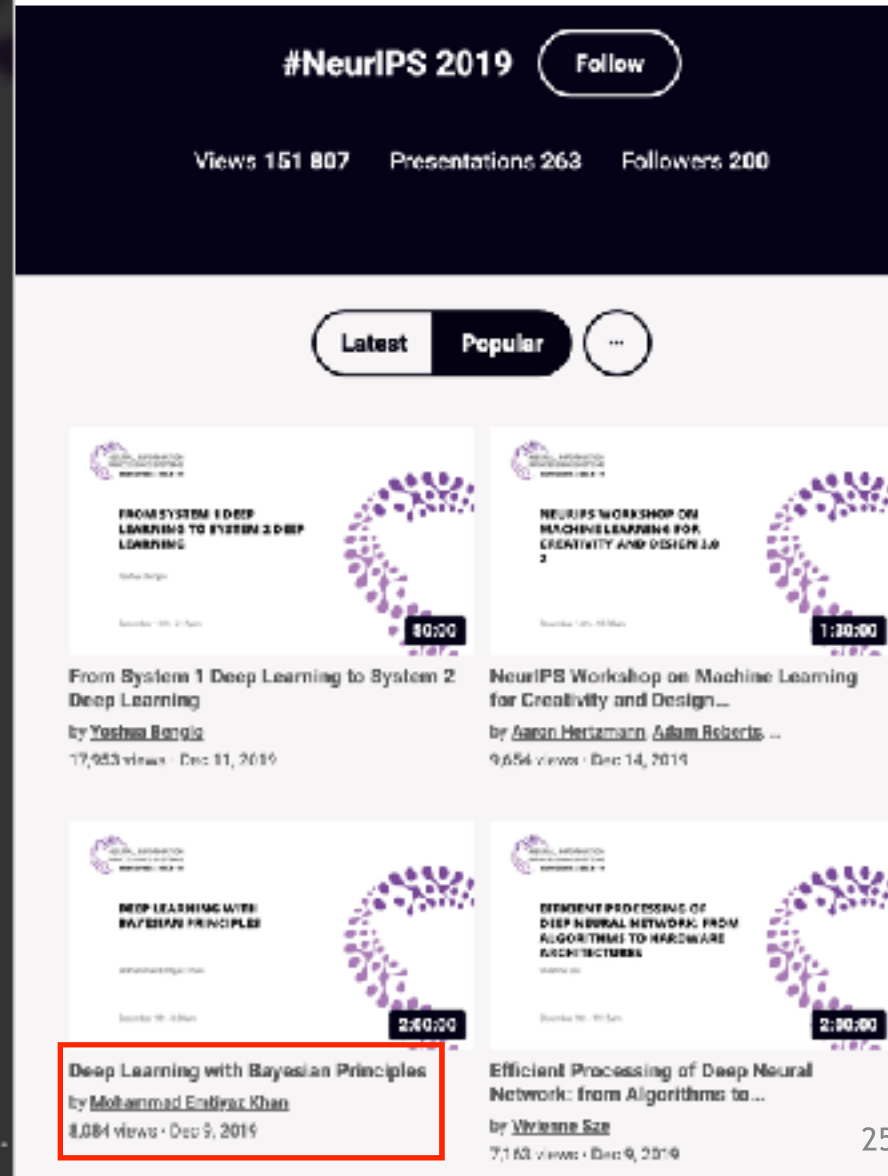


Uncertainty (entropy of class probs)

(By Roman Bachmann)



NeurIPS 2019 Tutorial



How do adapt the knowledge?
Perturbation, Sensitivity, and Duality

https://tenor.com/view/clockwork-gears-brain-gif-16784329

The webpage is available at https://bayesduality.github.io/, and Twitter account @BayesDuality

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.

Summary

- Bayesian principles
  - To unify/generalize/improve learning-algorithms
  - By computing “posterior approximations”
- Bayesian Learning rule (BLR)
  - Derive many existing algorithms
  - Deep Learning (SGD, RMSprop, Adam)
  - Design new algorithms for uncertainty in DL
- Impact: Everything with the same principle

Approximate Bayesian Inference Team

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

We have many open positions!
Come, join us.


Emtiyaz Khan
Team Leader


Pierre Alquier
Research Scientist


Gian Maria Marconi
Postdoc


Thomas Möllenhoff
Postdoc


Lu Xu
Postdoc


Jooyeon Kim
Postdoc

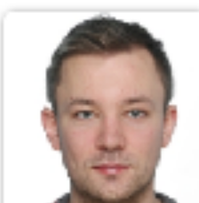

Wu Lin
PhD Student
University of British Columbia


David Tomàs Cuesta
Rotation Student, *Okinawa Institute of Science and Technology*


Dharmesh Tailor
Remote Collaborator
University of Amsterdam

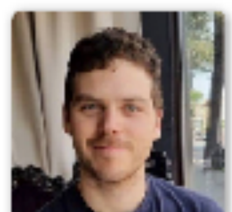

Erik Daxberger
Remote Collaborator
University of Cambridge


Tojo Rakotoaritina
Rotation Student, *Okinawa Institute of Science and Technology*


Peter Nickl
Research Assistant


Happy Buzaaba
Part-time Student
University of Tsukuba


Siddharth Swaroop
Remote Collaborator
University of Cambridge


Alexandre Piché
Remote Collaborator
MILA


Paul Chang
Remote Collaborator
Aalto University