

The Bayesian Learning Rule for Adaptive AI

Mohammad **Emtiyaz** Khan

RIKEN Center for AI Project, Tokyo

<http://emtiyaz.github.io>



Thanks to Dharmesh Tailor, Siddharth Swaroop, and Thomas Moellenhoff for their help in the preparation of the talk



AI that learns as quickly as humans and animals

Quickly **adapt** to new situations in the future
by **robustly preserving** & using past knowledge

Human Learning at
the age of 6 months.



Converged at the
age of 12 months



Transfer
skills
at the age
of 14
months



Fail because too quick to adapt

TayTweets: Microsoft AI bot manipulated into being extreme racist upon release

Posted Fri 25 Mar 2016 at 4:38am, updated Fri 25 Mar 2016 at 9:17am



TayTweets is programmed to converse like a teenage girl who has "zero chill", according to Microsoft. (Twitter/TayTweets)

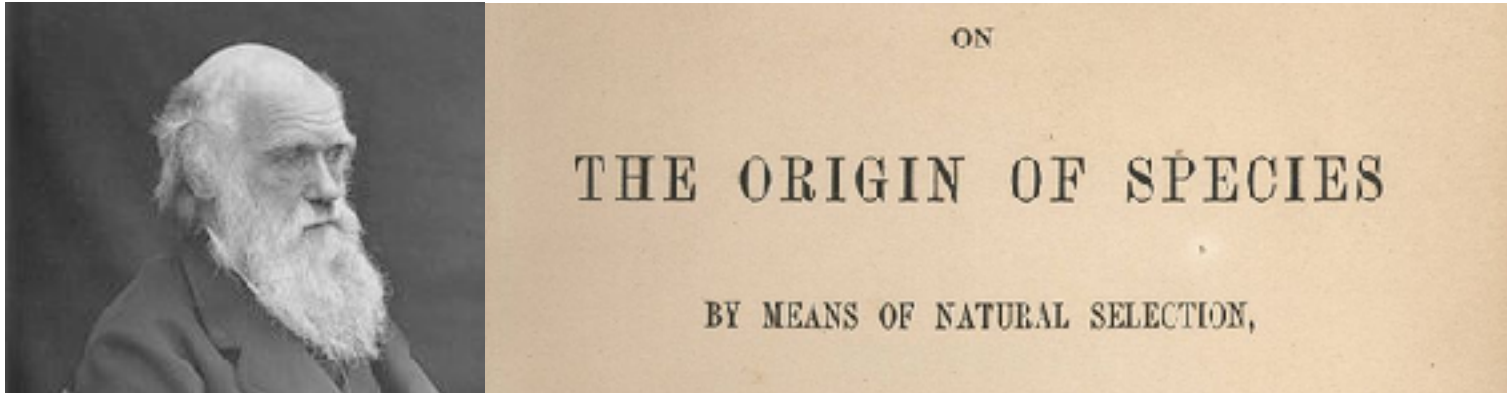
Fail because too slow to adapt



Adaptive & Robust Learning with Bayes

- “Good” algorithms are inherently Bayesian
- Bayesian learning rule [1]
- Robustness: Memorable experiences [2]
- Adaptation: Knowledge-Adaptation Priors [3,4,5]
- Take away: A new perspective of Bayes, essential for adaptive and robust deep learning

1. Khan and Rue, The Bayesian Learning Rule, arXiv, <https://arxiv.org/abs/2107.04562>, 2021
2. Tailor, Chang, Swaroop, Tangkaratt, Solin, Khan. Memorable experiences of ML models (in preparation)
3. Khan et al. Approximate Inference Turns Deep Networks into Gaussian Process, NeurIPS, 2019
4. Pan et al. Continual Deep Learning by Functional Regularisation of Memorable Past, NeurIPS, 2020
5. Khan and Swaroop. Knowledge-Adaptation Priors, NeurIPS, 2021 (<https://arxiv.org/abs/2106.08769>)



The Origin of Algorithms

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

A Bayesian Origin

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

\uparrow
Posterior approximation (expo-family)

$\mathcal{H}(q)$
Entropy

Bayesian Learning Rule [1,2]

Natural and Expectation parameters of q

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

\uparrow Old belief \uparrow Revise using new information through 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: Converting variational inference in non-conjugate models to inferences in conjugate models." Alstats (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

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

Put the expectation (Bayes) back in!

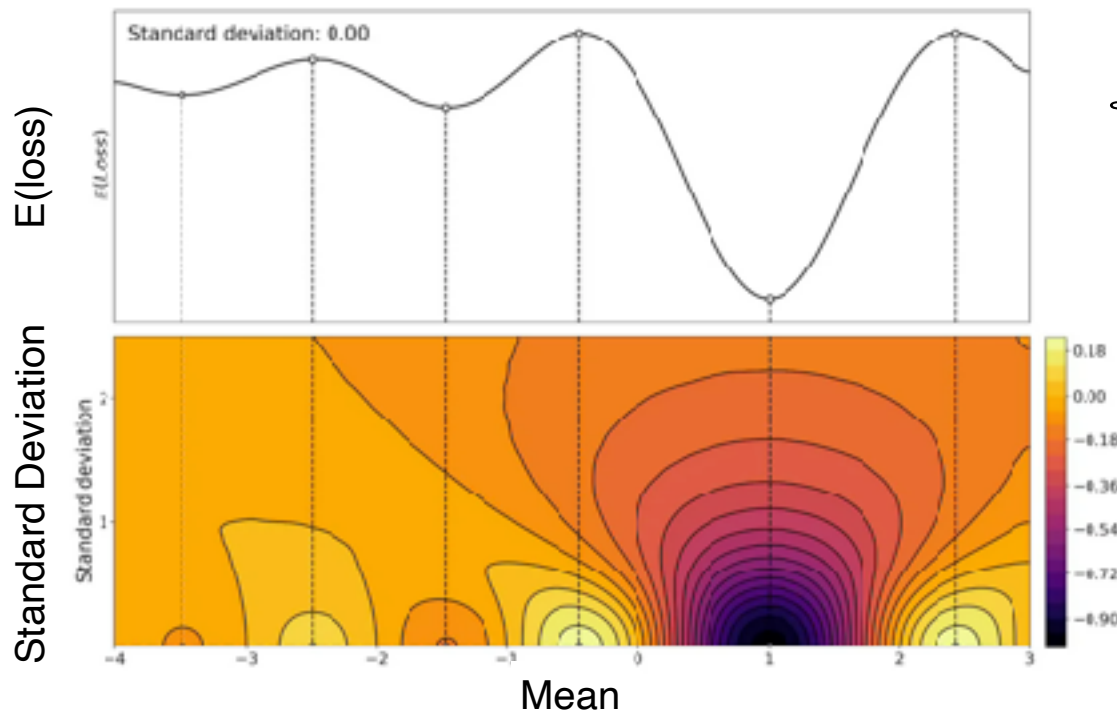
Learning Algorithm	Posterior Approx.	Natural-Gradient Approx.	Sec.
Optimization Algorithms			
Gradient Descent	Gaussian (fixed cov.)	Delta method	1.3
Newton's method	Gaussian	—"—"	1.3
Multimodal optimization <small>(New)</small>	Mixture of Gaussians	—"—"	3.2
Deep-Learning Algorithms			
Stochastic Gradient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

Bayes Objective

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

← Generalized-Posterior approx.



$$\mathcal{L}(\mu, \sigma) = \mathbb{E}_{\mathcal{N}(\theta|\mu, \sigma^2)} [\ell(\theta)]$$

Instead of the original loss, optimize a different (smoothed) one.

A popular idea of “implicit regularization” in DL [4] now, but also common in other fields (RL, search)

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

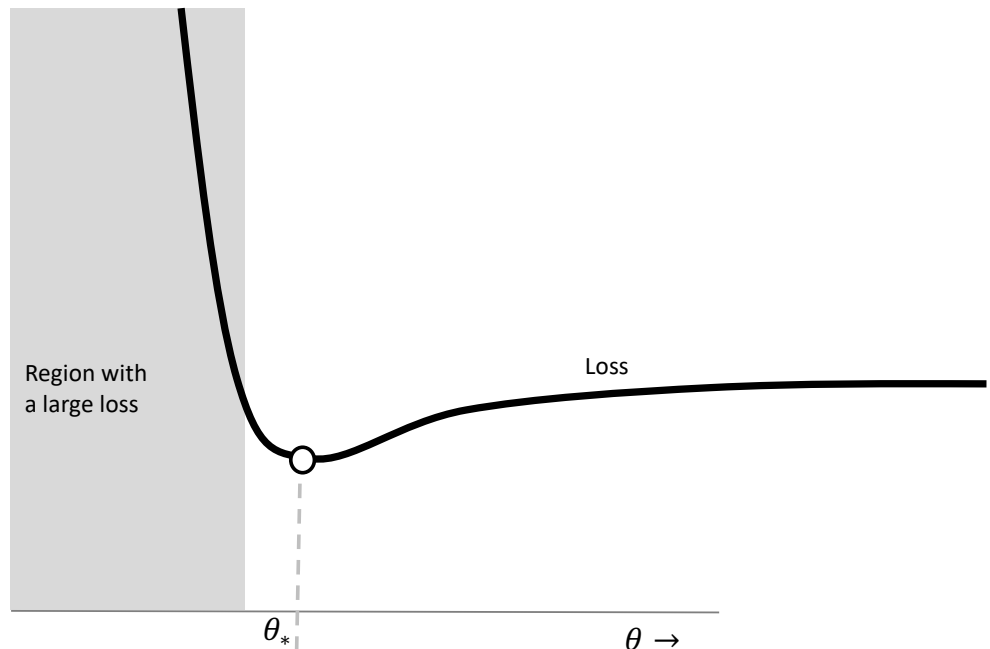
Bayes Prefers Flatter directions

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

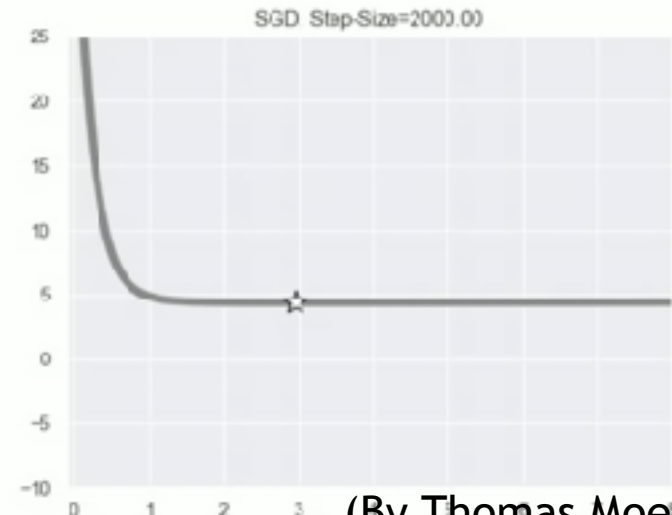
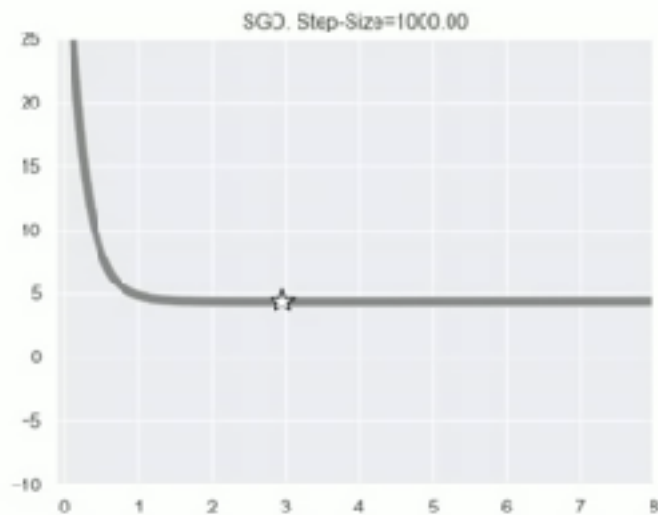
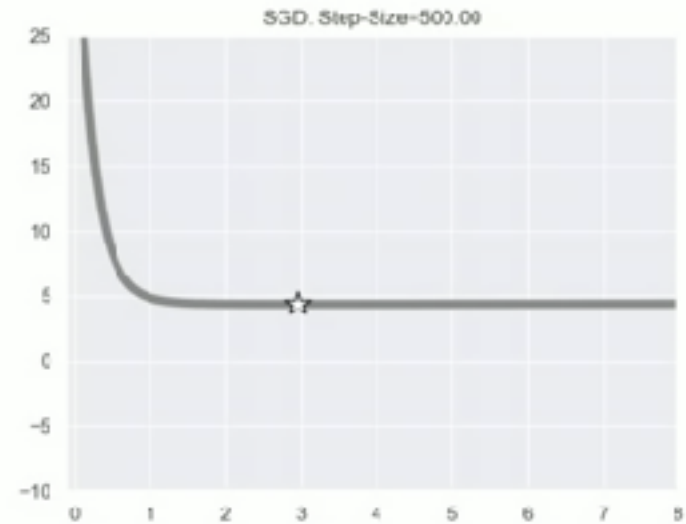
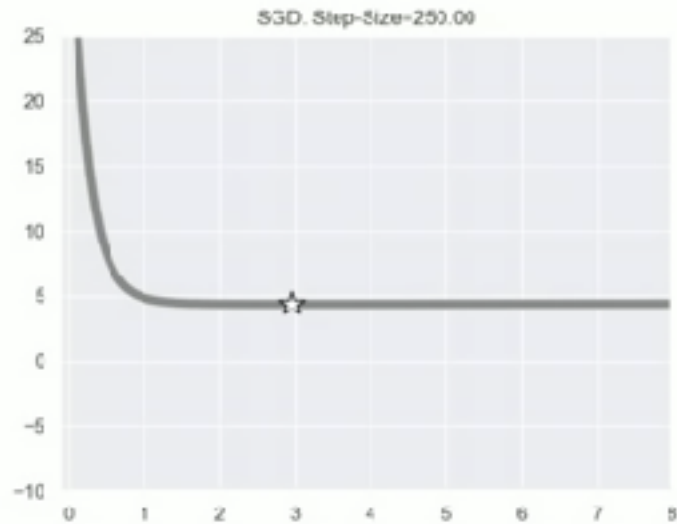
$$\text{BLR: } m \leftarrow m - \rho \nabla_m \mathbb{E}_q[\ell(\theta)]$$

$$\implies \nabla_m \mathbb{E}_{q_*}[\ell(\theta)] = 0 \quad \implies \mathbb{E}_{q_*}[\nabla_{\theta} \ell(\theta)] = 0$$

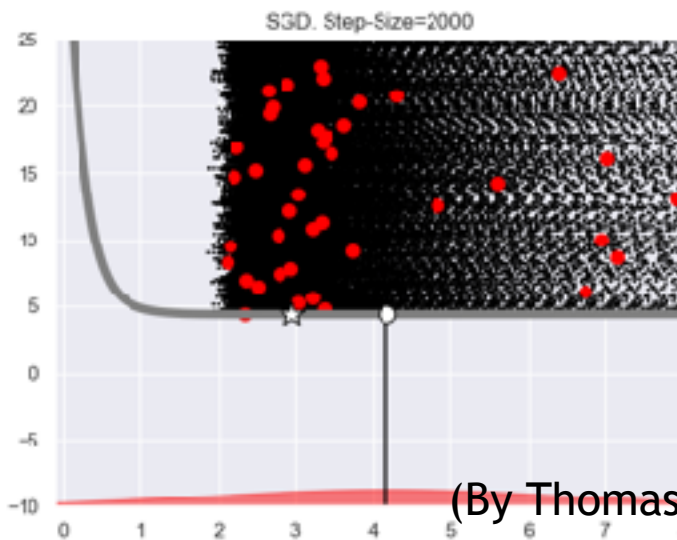
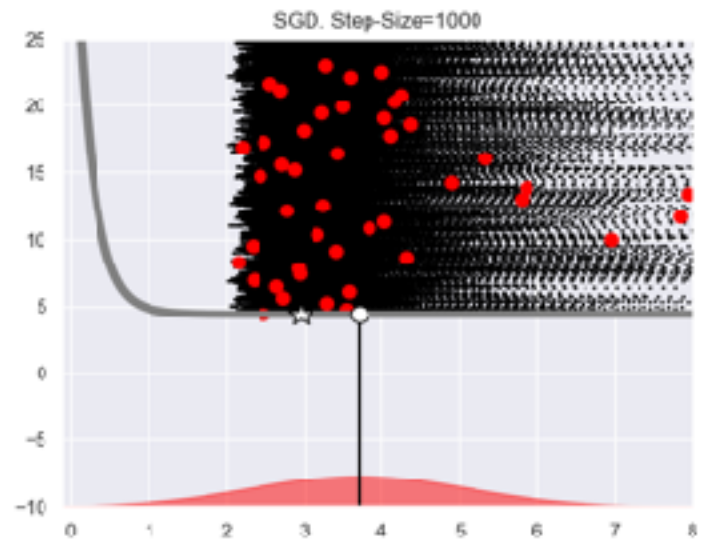
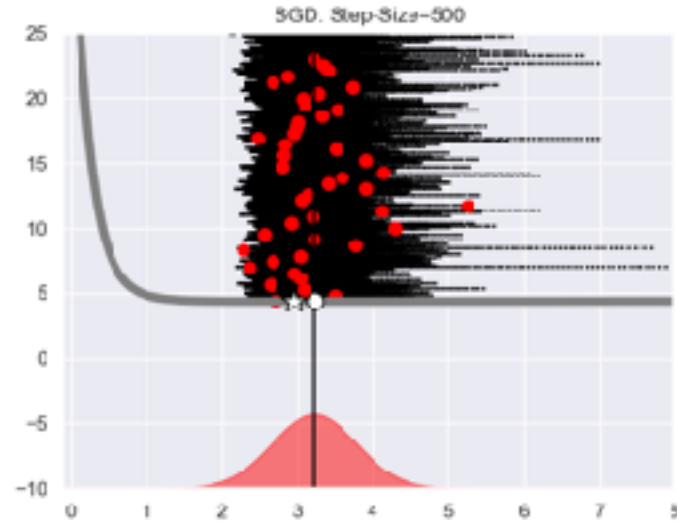
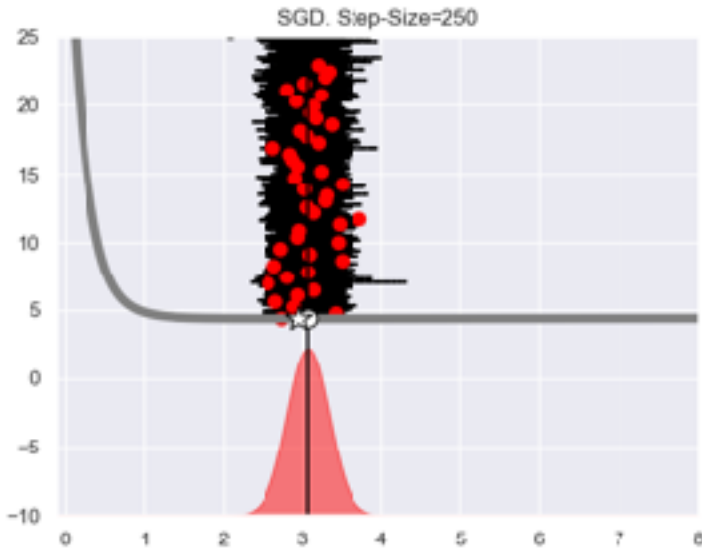
Bayesian solution injects “noise” which has a similar regularization effect to noise in Stochastic GD. It prefers “flatter” directions.



SGD: Implicit Regularization

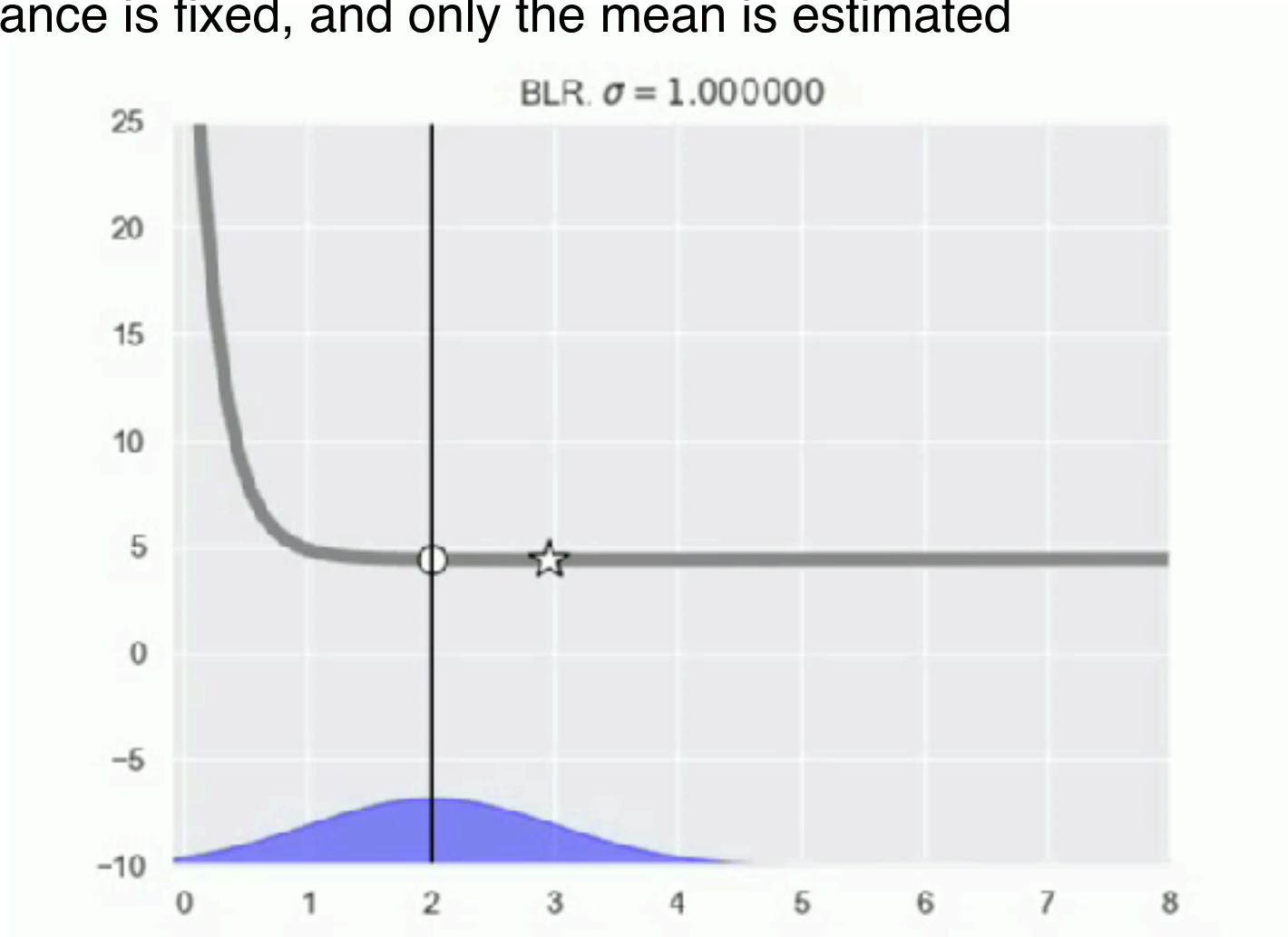


SGD: Implicit Regularization

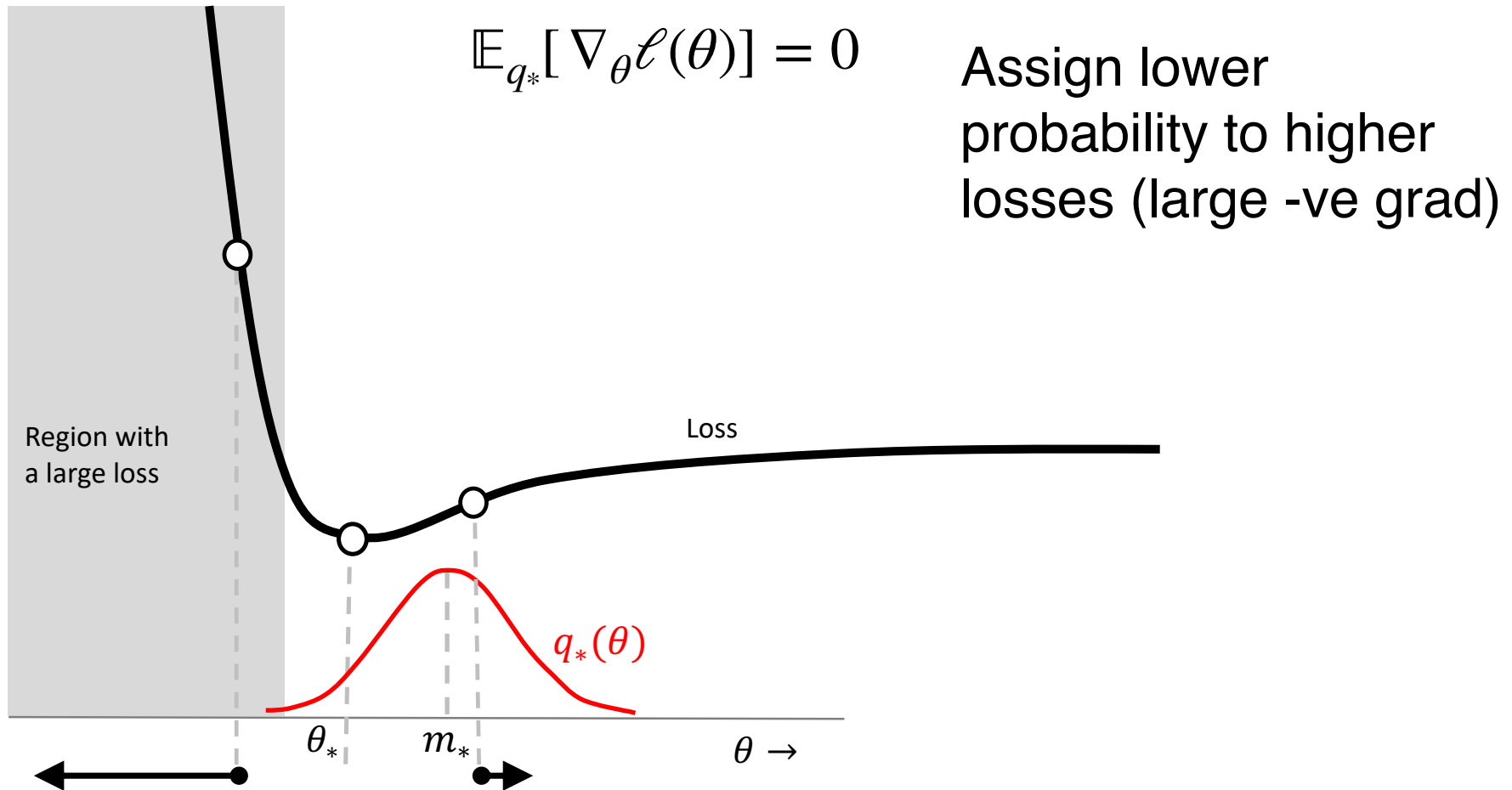


Bayes: Implicit Regularization

Estimating Gaussian posteriors where the variance is fixed, and only the mean is estimated

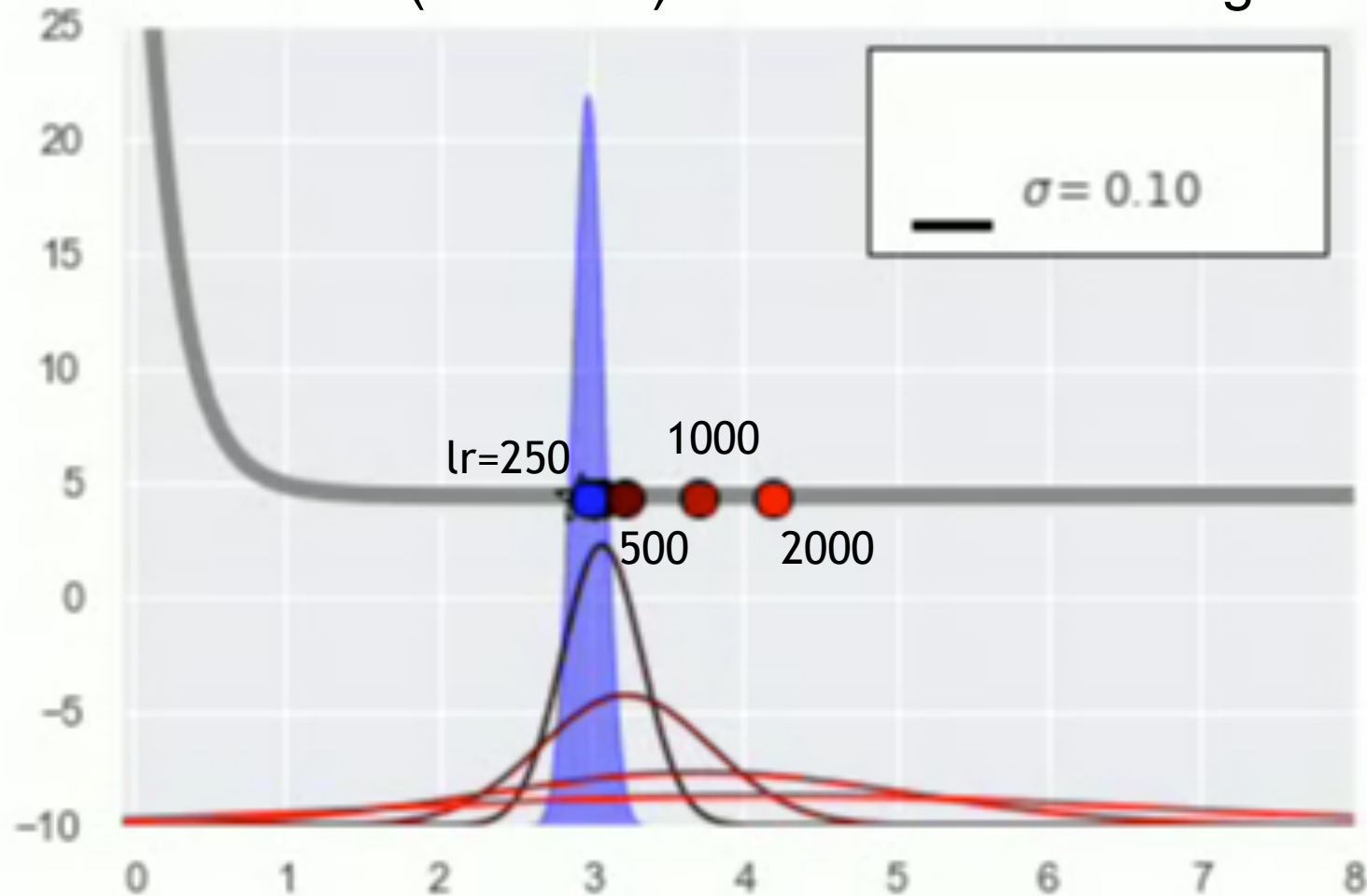


Bayes: Implicit Regularization



Bayes: Implicit Regularization

Bayes solutions (blue) with different variances vs SGD solutions (red lines) with different learning rates.

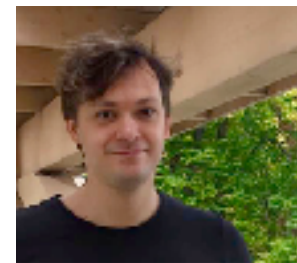


**I am once again asking
for you to be a Bayesian!**

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

Put the expectation (Bayes) back in!

The BLR variants [1,2,3] led to the winning solution for the NeurIPS 2021 challenge for “approximate inference in BDL”. Watch **Thomas Moellenhoff’s** talk at <https://www.youtube.com/watch?v=LQInIN5EU7E>



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

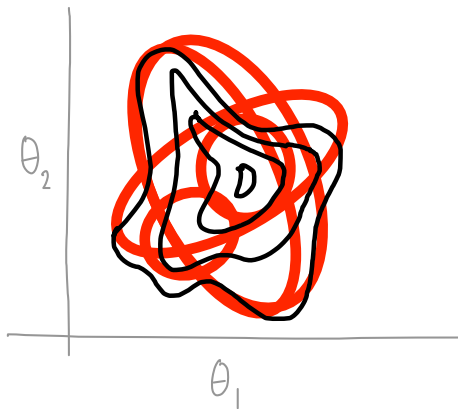
1. Khan, et al. "Fast and scalable Bayesian deep learning by weight-perturbation in Adam." *ICML* (2018).
2. Osawa et al. "Practical Deep Learning with Bayesian Principles." *NeurIPS* (2019).
3. Lin et al. "Handling the positive-definite constraints in the BLR." *ICML* (2020).

Deriving Learning-Algorithms from the Bayesian Learning Rule

Posterior Approximation \longleftrightarrow Learning-Algorithm

Complex

Simple



Bayes' rule

Mixture
of Newton

Newton

Gradient
Descent

Newton's Method from Bayes

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

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

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

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

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 Bayes

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:

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

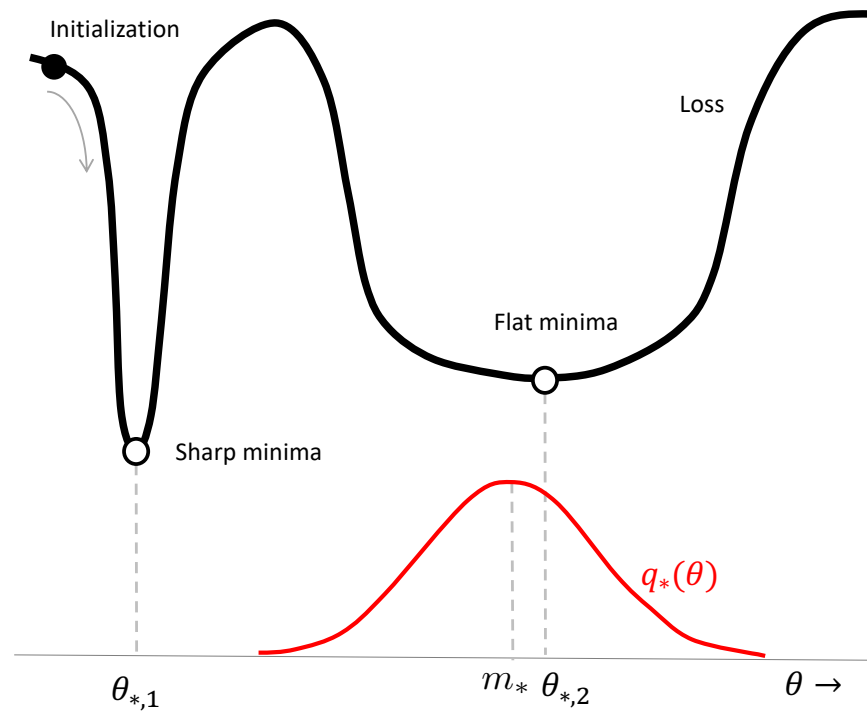
$$\nabla_{\mathbb{E}_q(\theta\theta^{\top})} \mathbb{E}_q[\ell(\theta)] = \mathbb{E}_q[H_{\theta}]$$

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

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

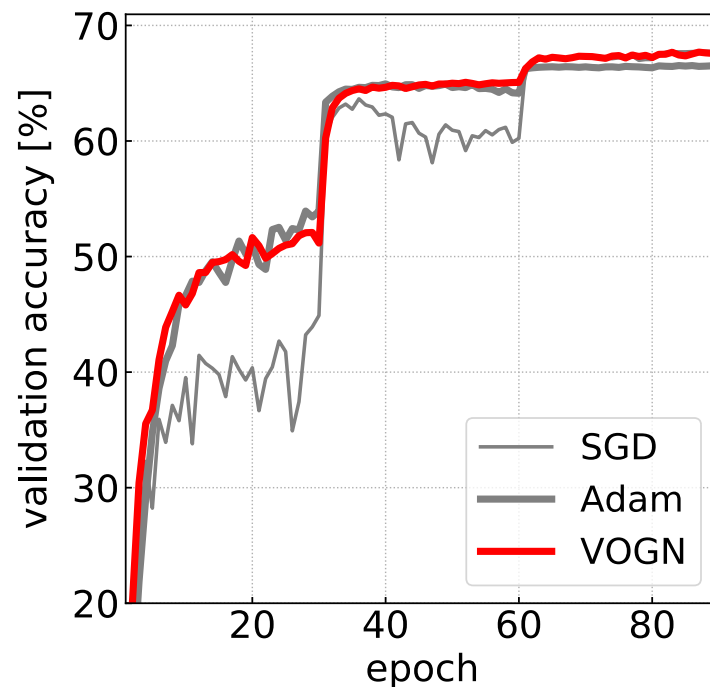
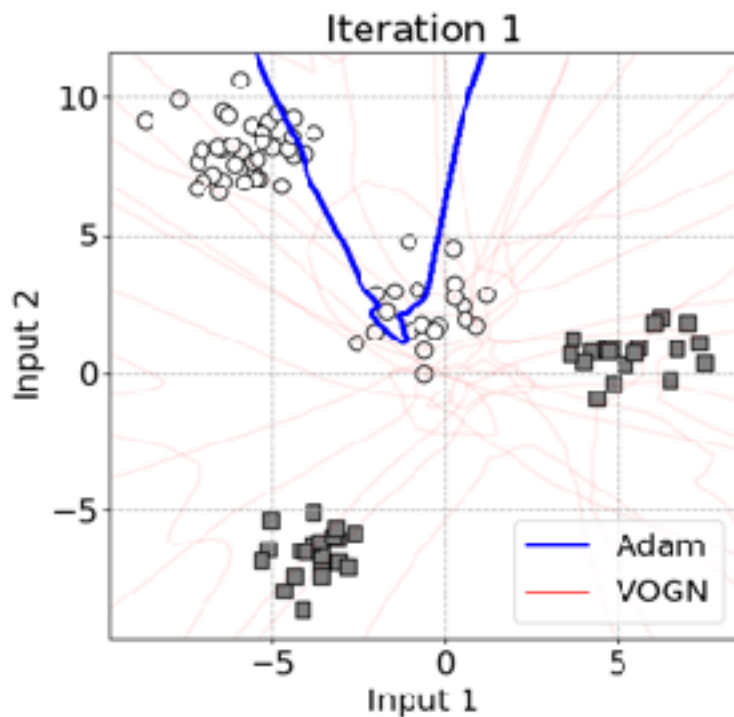
Bayes leads to robust solutions

Avoiding sharp minima



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 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)

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

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

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

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

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

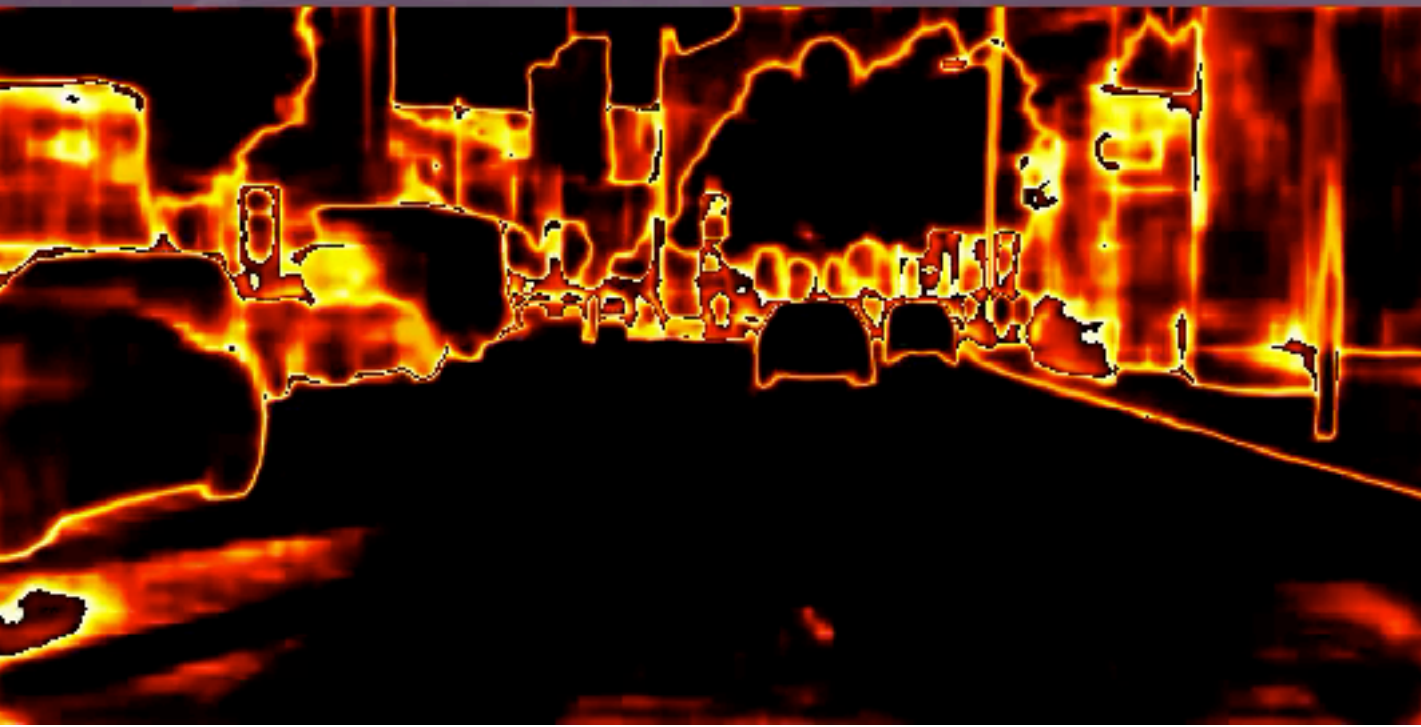
The BLR variant from [3] led to the **winning solution** for the NeurIPS 2021 challenge for “approximate inference in deep learning”. Watch **Thomas Moellenhoff’s** talk at <https://www.youtube.com/watch?v=LQInIN5EU7E>.



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).

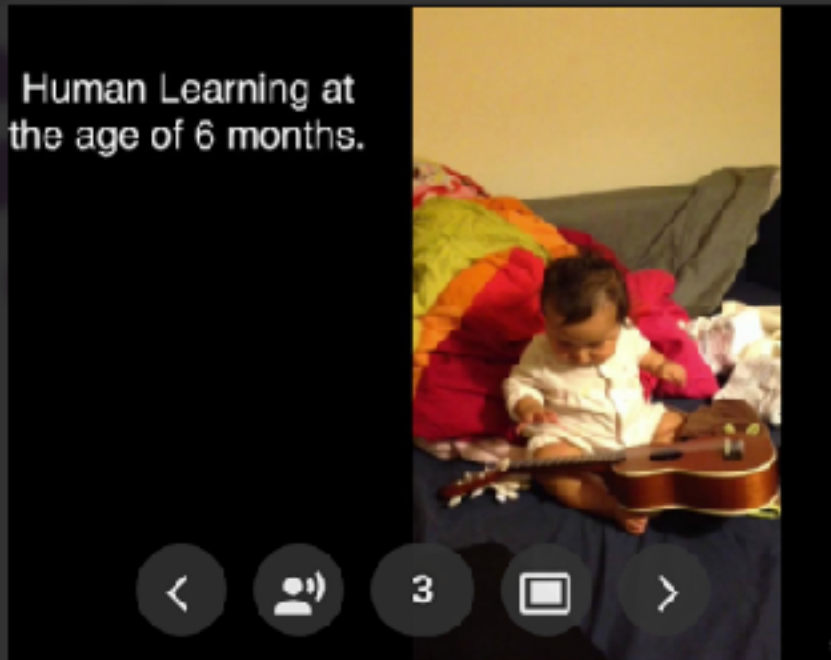
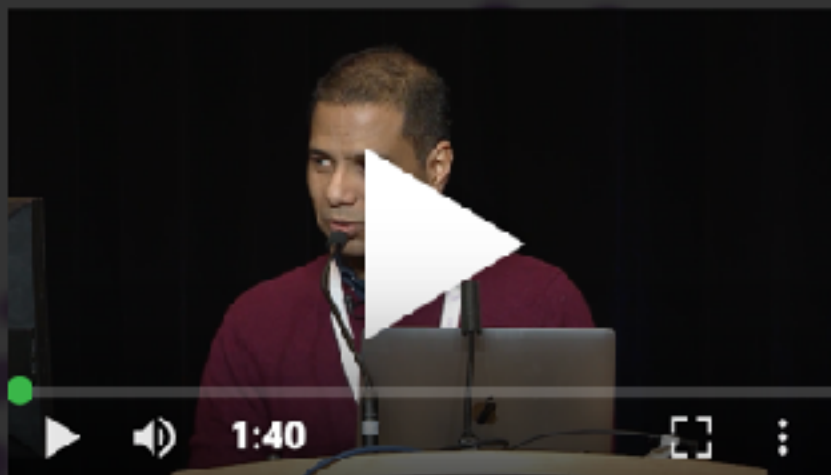


Image
Segmentation



Uncertainty
(entropy of
class probs)

NeurIPS 2019 Tutorial



Deep Learning with Bayesian Principles

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

#NeurIPS 2019

Follow

Views 151 807

Presentations 263

Followers 200

Latest

Popular

...



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...

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

9,654 views · Dec 14, 2019



Deep Learning with Bayesian Principles

by [Mohammad Emtiyaz Khan](#)

4,084 views · Dec 5, 2019



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

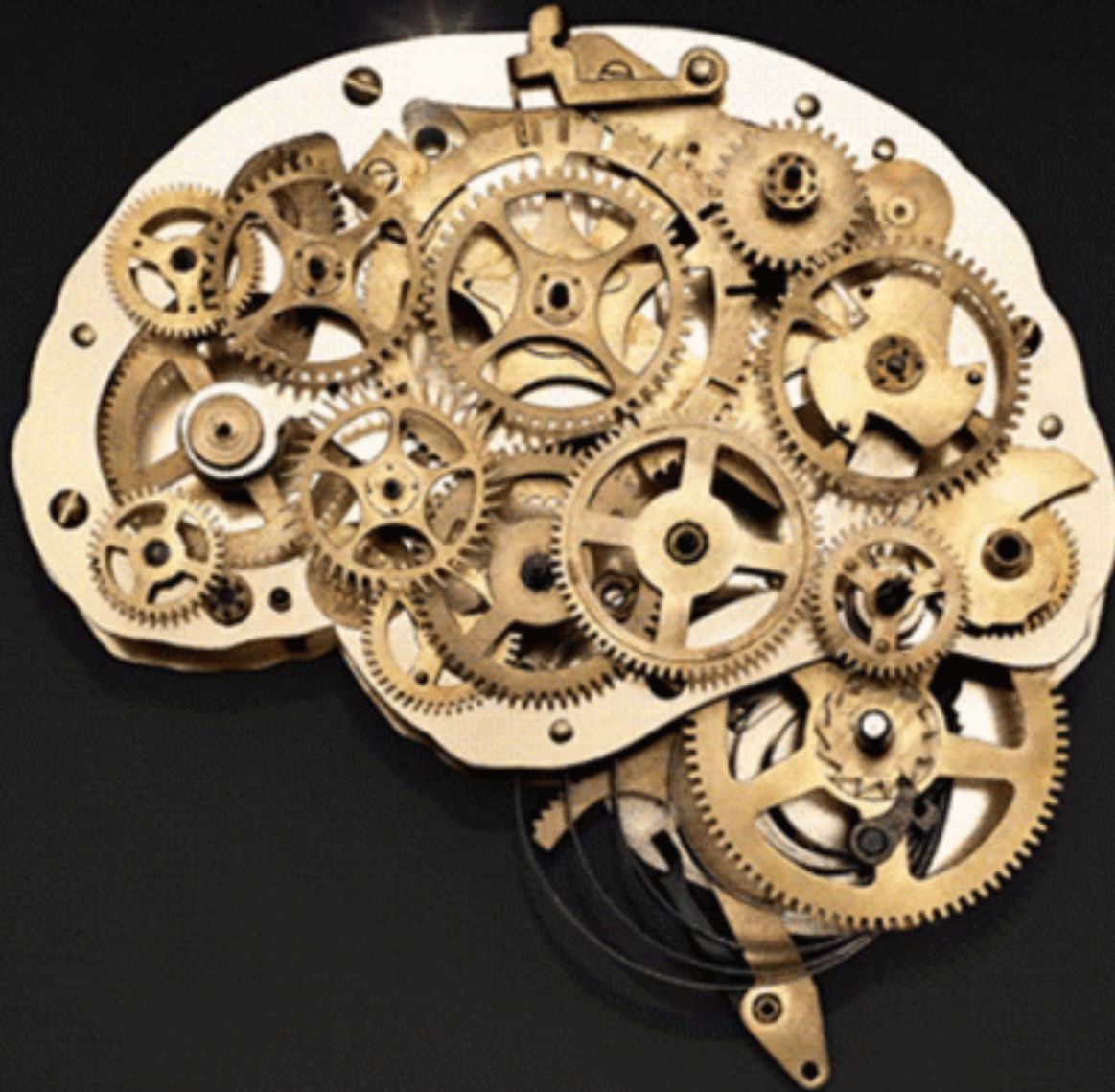
by [Wiyenne Sze](#)

7,162 views · Dec 9, 2019

Robustness

Good algorithms can tell apart
relevant vs irrelevant information

Perturbation, Sensitivity, and Duality



via steampunktendencies.com

BLR Solutions & Their Duality

$$\ell(\theta) = \sum_{i=0}^N \ell_i(\theta) \quad \lambda \leftarrow (1 - \rho)\lambda - \sum_{i=0}^N \rho \nabla_{\mu} \mathbb{E}_q[\ell_i(\theta)]$$

$$\lambda^* = \sum_{i=0}^N \underbrace{\nabla_{\mu^*} \mathbb{E}_{q^*}[-\ell_i(\theta)]}_{\tilde{\lambda}_i^*}$$

Global and local natural parameter

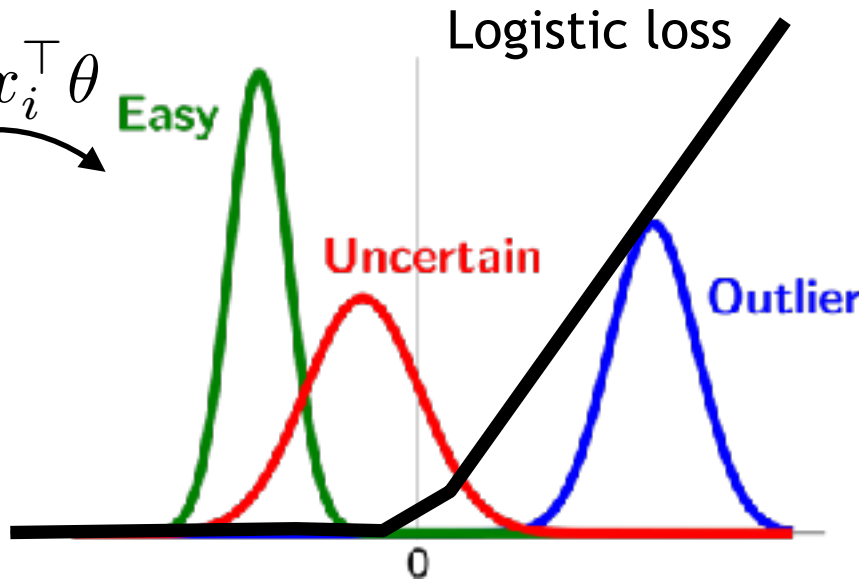
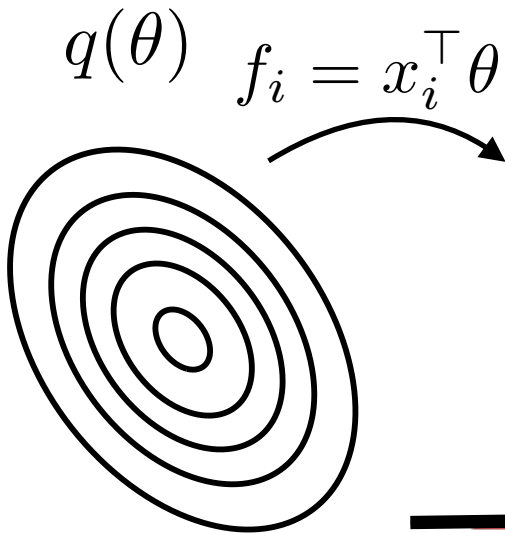
Local parameters are **Lagrange Multipliers**, measuring the sensitivity of BLR solutions to local perturbation [1]. They can be used to tell apart relevant vs irrelevant data.

Memorable Experiences

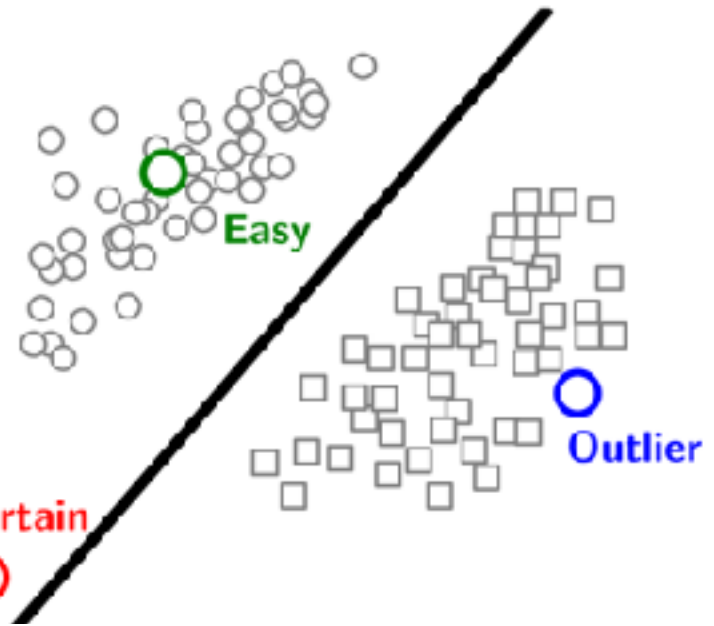
$$\lambda^* = \sum_{i=0}^N \underbrace{\nabla_{\mu^*} \mathbb{E}_{q^*} [-\ell_i(\theta)]}_{\tilde{\lambda}_i^*}$$

“Global”
posterior

Local predictions $q(f_i)$



Uncertain



Lower Sensitivity
to easy example.

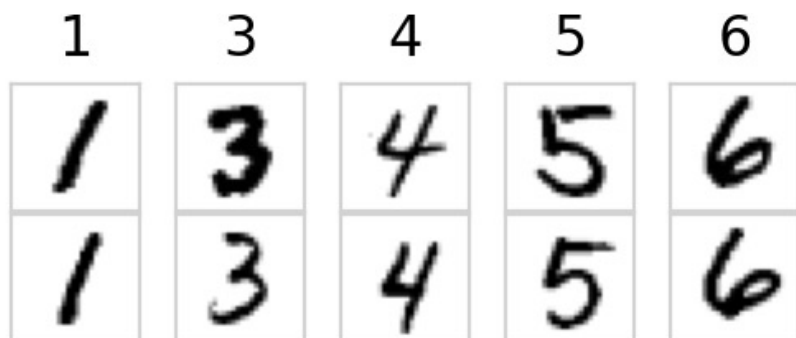
Such sensitivity
analysis leads to
memorable
experiences

Memorable Experiences

MNIST

FMNIST

Easy



Outliers



Uncertain



Advantages of Memorable Experiences

- Through posterior approximations, the criteria to categorize examples **naturally emerges**
 - Generalizes existing concepts such as support vectors, influence functions, inducing inputs etc
- Local parameters are available for free and applies to almost “any” ML problem
 - Supervised, unsupervised, RL
 - Discrete/continuation loss and model parameters
- The sensitivity of posterior leads to “Bayes Duality”

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

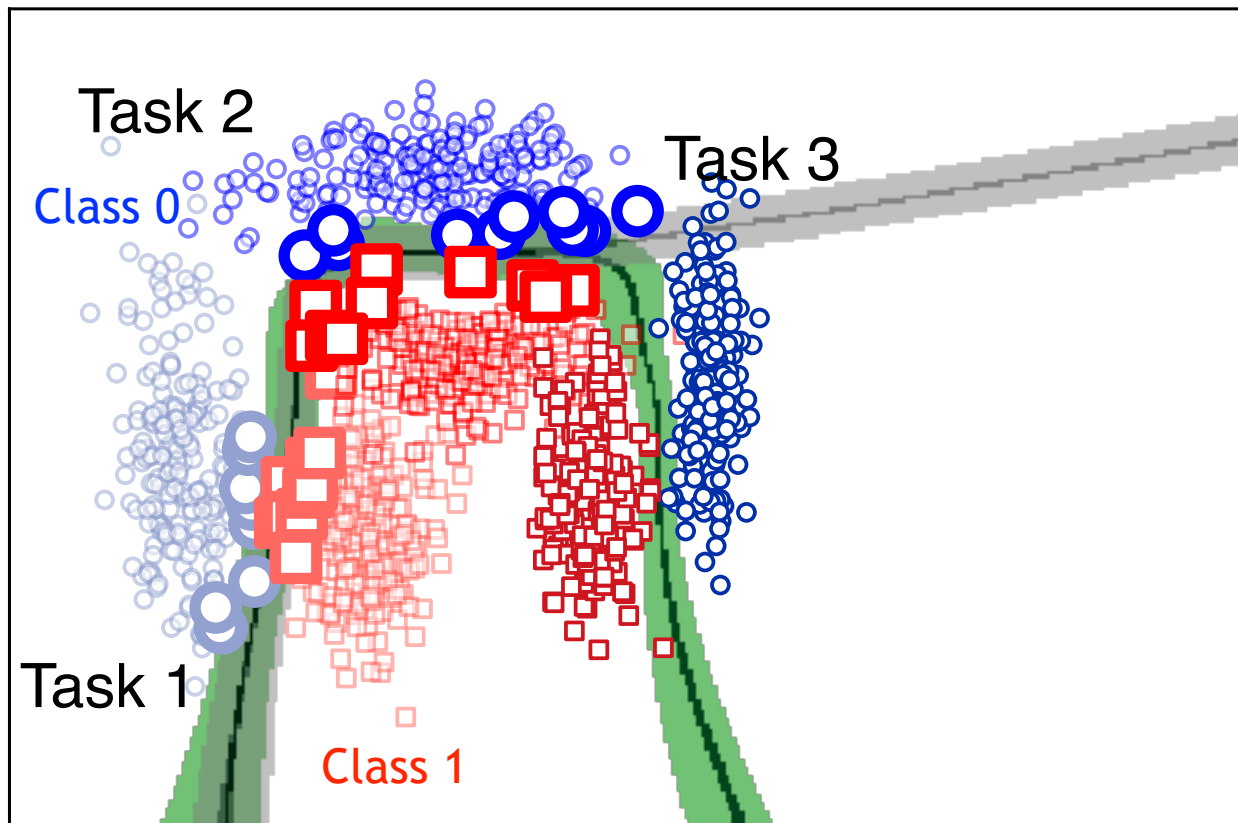
Received total funding of around **USD 3 million** through JST's CREST-ANR and Kakenhi Grants.

Adaptation

Continual Learning without forgetting the past (by using memorable examples)

Continual Learning

Avoid forgetting by using memorable examples [1,2]



1. Khan et al. Approximate Inference Turns Deep Networks into Gaussian Process, NeurIPS, 2019
2. Pan et al. Continual Deep Learning by Functional Regularisation of Memorable Past, NeurIPS, 2020

Functional Regularization of Memorable Past (FROMP) [4]

Previous approaches used weight-regularization [1,2]

$$q_{new}(\theta) = \min_{q \in \mathcal{Q}} \underbrace{\mathbb{E}_{q(\theta)}[\ell_{new}(\theta)]}_{\text{New data}} - \mathcal{H}(q) - \underbrace{\mathbb{E}_{q(\theta)}[\log q_{old}(\theta)]}_{\text{Weight-regularizer}}$$

We replace it by a functional regularizer using a “Gaussian Process view” of DNNs [2]

$$[\sigma(\mathbf{f}(\theta)) - \sigma(\mathbf{f}_{old})]^\top K_{old}^{-1} [\sigma(\mathbf{f}(\theta)) - \sigma(\mathbf{f}_{old})]$$

$\mathbb{E}_{\tilde{q}_\theta(\mathbf{f})} [\log \tilde{q}_{\theta_{old}}(\mathbf{f})]$
 \downarrow
 \uparrow

Kernels weighs examples according to their memorability Forces network-outputs to be similar

1. Kirkpatrick, James, et al. "Overcoming catastrophic forgetting in neural networks." *PNAS* 2017
2. Nguyen et al., Variational Continual Learning, ICLR, 2018
3. Khan et al. Approximate Inference Turns Deep Networks into Gaussian Process, NeurIPS, 2019
4. Pan et al. Continual Deep Learning by Functional Regularisation of Memorable Past, NeurIPS, 2020

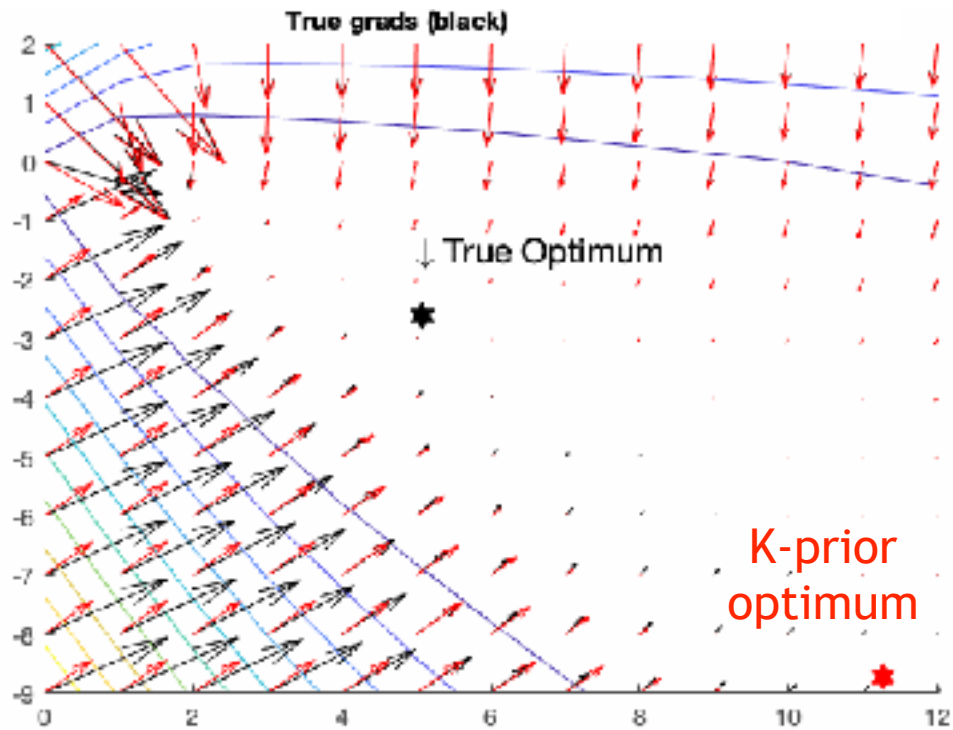
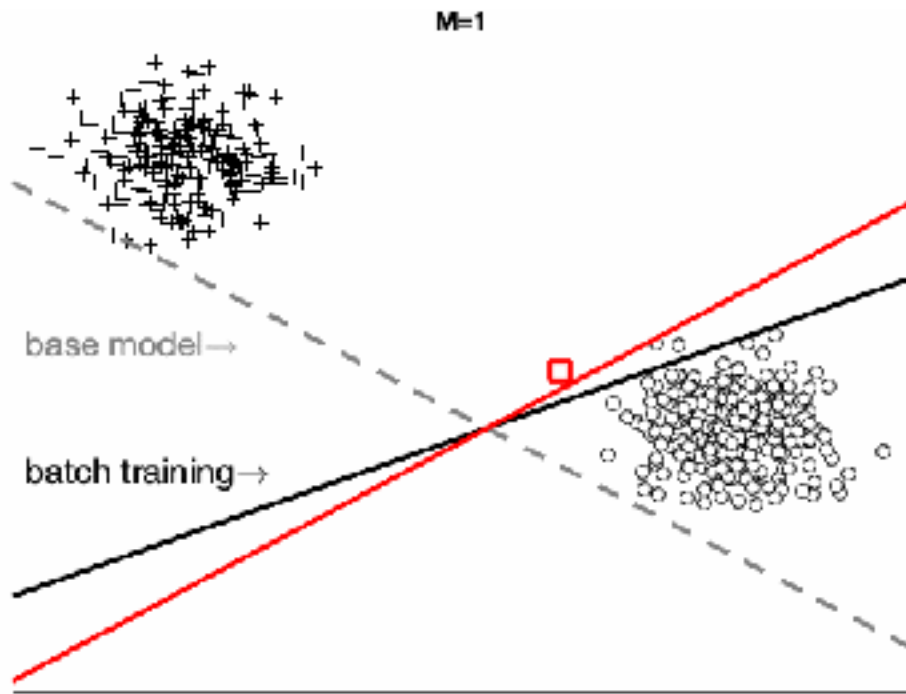
K-Priors and Bayes-Duality

- Dual parameterization of DNNs
 - expressed as Gaussian Process [1]
 - Found using the Bayesian learning rule
- The functional regularizer can provably reconstruct the gradient of the past faithfully [2]
 - Knowledge-Adaptation priors (K-priors)
 - There is a strong evidence that “good” adaptive algorithms must use K-priors

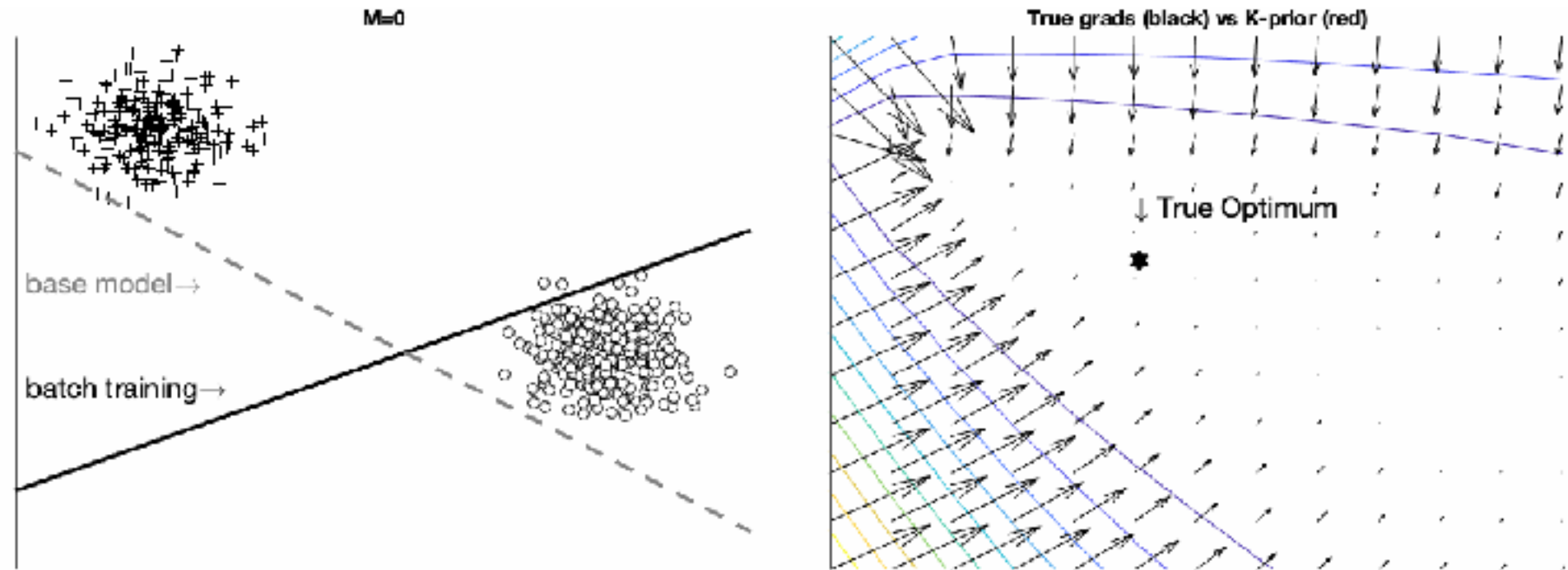
1. Khan et al. Approximate Inference Turns Deep Networks into Gaussian Process, NeurIPS, 2019

2. Khan and Swaroop. Knowledge-Adaptation Priors, NeurIPS, 2021 (<https://arxiv.org/abs/2106.08769>)

Faithful Gradient Reconstruction



Faithful Gradient Reconstruction



No labels required, so \mathcal{M} can include any inputs!

Summary

- A new perspective of Bayes, essential for adaptive and robust deep learning
- Approximate posteriors are crucial
 - Bayesian learning rule [1]
 - Robustness: Memorable experiences [2]
 - Adaptation: K-Priors [3,4,5]
- Bayes-duality for AI that learns like humans

1. Khan and Rue, The Bayesian Learning Rule, arXiv, <https://arxiv.org/abs/2107.04562>, 2021
2. Tailor, Chang, Swaroop, Tangkaratt, Solin, Khan. Memorable experiences of ML models (in preparation)
3. Khan et al. Approximate Inference Turns Deep Networks into Gaussian Process, NeurIPS, 2019
4. Pan et al. Continual Deep Learning by Functional Regularisation of Memorable Past, NeurIPS, 2020
5. Khan and Swaroop. Knowledge-Adaptation Priors, NeurIPS, 2021 (<https://arxiv.org/abs/2106.08769>)

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



Yyu Lin
PhD Student
University of British Columbia



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



Dharmesh Tallor
Remote
Collaborator
University of Amsterdam



Erik Daxberger
Remote
Collaborator
University of Cambridge



Tojo Rakotoarindra
Rotation Student,
Okinawa Institute of Science and Technology



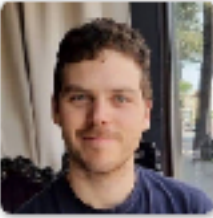
Peter Nidd
Research Assistant



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