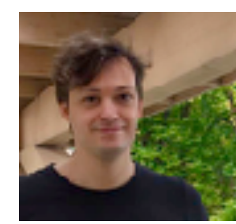
Presentation at EPFL CIS - RIKEN AIP Joint Seminar Series

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





Slides at https://emtiyaz.github.io/papers/Apr13_2022_EPFL_RIKEN.pdf
Summary of research at https://emtiyaz.github.io/papers/symposium_2021.pdf

EPFL master students in our group

Many thanks to Patricia Genet (Internship program assistant) and Sylviane Dalmas for their help!

Nicolas Hubacher
Research Assistant
Jan-Dec 2017

Worked on structured VAEs (coauthor an ICLR 2018 paper), now working as a data-scientist in Switzerland

Frederik Kunstner
Intern,
Feb-Aug 2018

Worked on natural-gradients and Adam, coauthor on a NeurIPS 2018 paper, now a PhD student at UBC

Ehsan Abedi
Intern,
Mar-Aug 2019

Worked on connections between NN and GPs, coauthor on a NeurIPS 2019 paper, now a PhD student in Switzerland (?)

Alex Immer
Intern,
Mar 2019-
Mar 2020

Worked on linearized DNNs (coauthored NeurIPS 2019, AIstats 2020, and ICML 2021 papers, and a thesis), now a PhD student in ETH

Roman Bachmann
Intern,
July 2019-
Feb 2020

Worked on Binary Neural Networks (coauthored an ICML 2021 paper), now a PhD student in EPFL

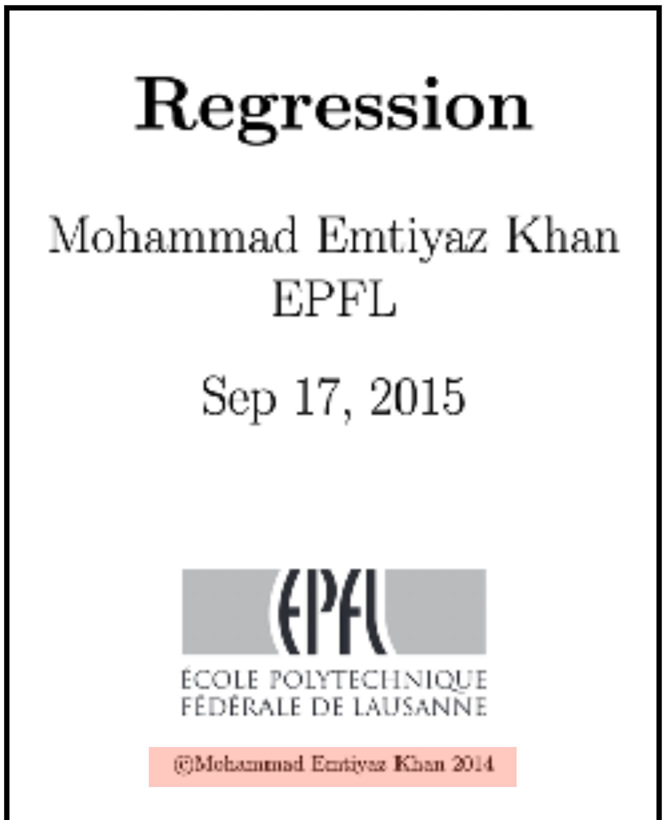
Lucie Perrotta
Intern,
Sep 2019-
Mar 2020

Worked on model misspecification and tempering (did a masters thesis)

Pattern Classification and Machine Learning CS-433

I designed this course in 2014-2015 (around 200 students back then)

How it started?



Regression

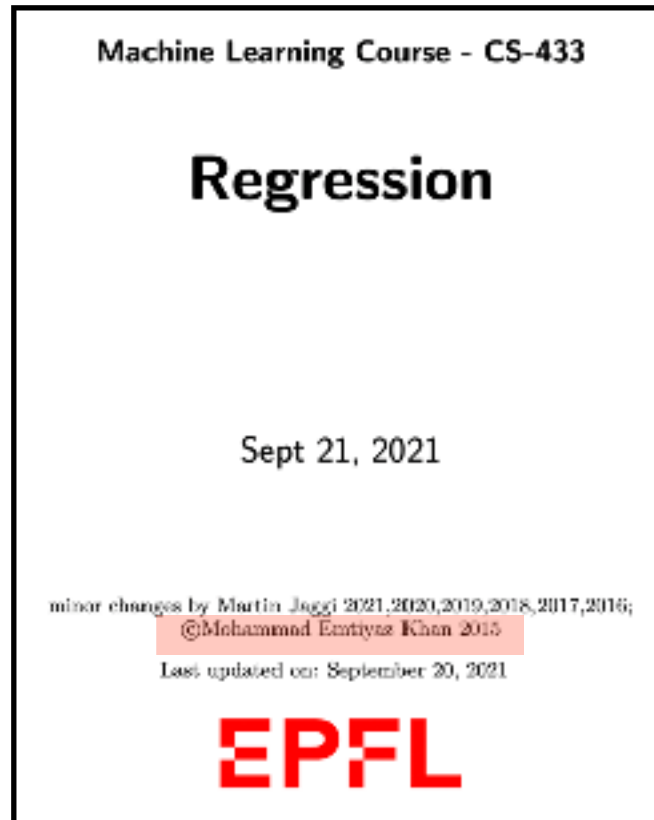
Mohammad Emtiyaz Khan
EPFL

Sep 17, 2015

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

©Mohammad Emtiyaz Khan 2014

How it's going?



Machine Learning Course - CS-433

Regression

Sept 21, 2021

minor changes by Martin Jaggi 2021,2020,2019,2018,2017,2016;
©Mohammad Emtiyaz Khan 2015
Last updated on: September 20, 2021

EPFL

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

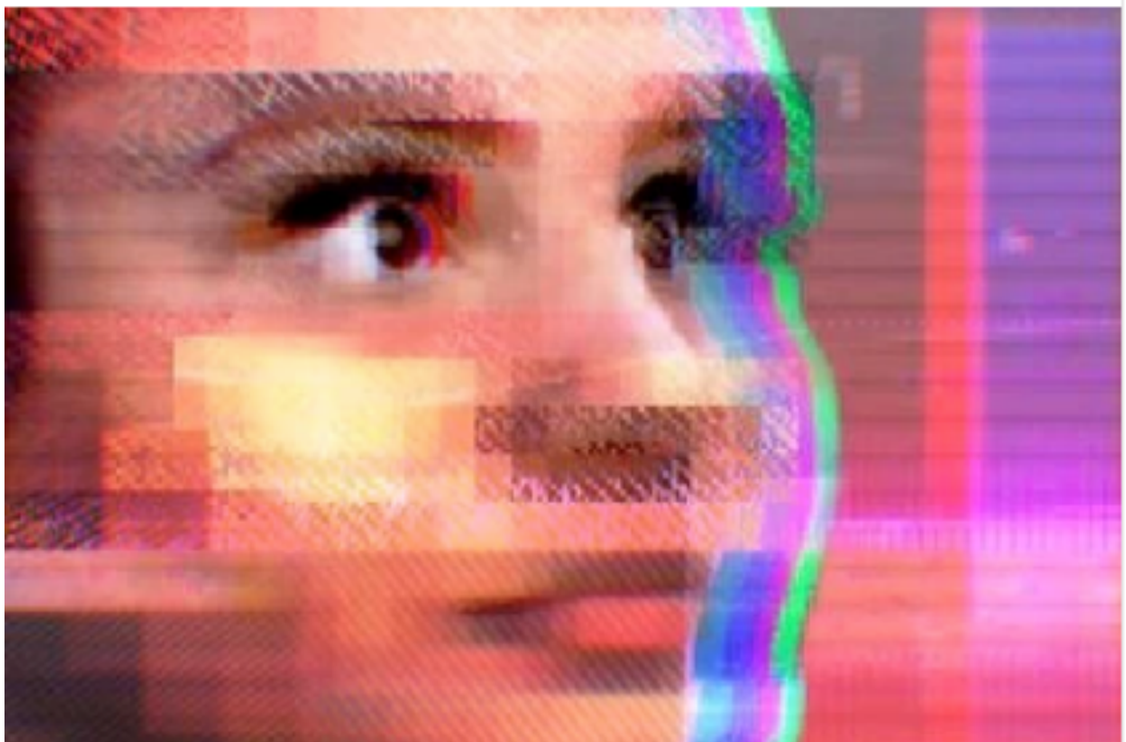
Transfer skills

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)

https://www.abc.net.au/news/2016-03-25/microsoft-created-ai-bot-becomes-racist/7276266

Fail because too slow to adapt

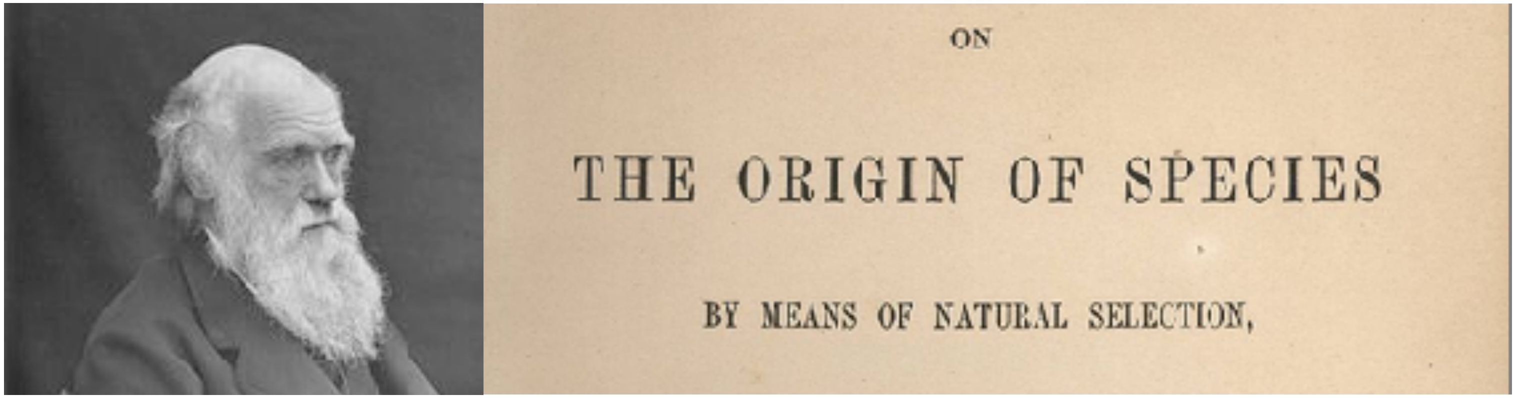
https://www.youtube.com/watch?v=TxobtWAFh8o The video is from 2017

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, 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)

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

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

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

Entropy

Posterior approximation (expo-family)

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

Natural and Expectation parameters of q

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

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: Converting variational inference in non-conjugate models to inferences in conjugate models." 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

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

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

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

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

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

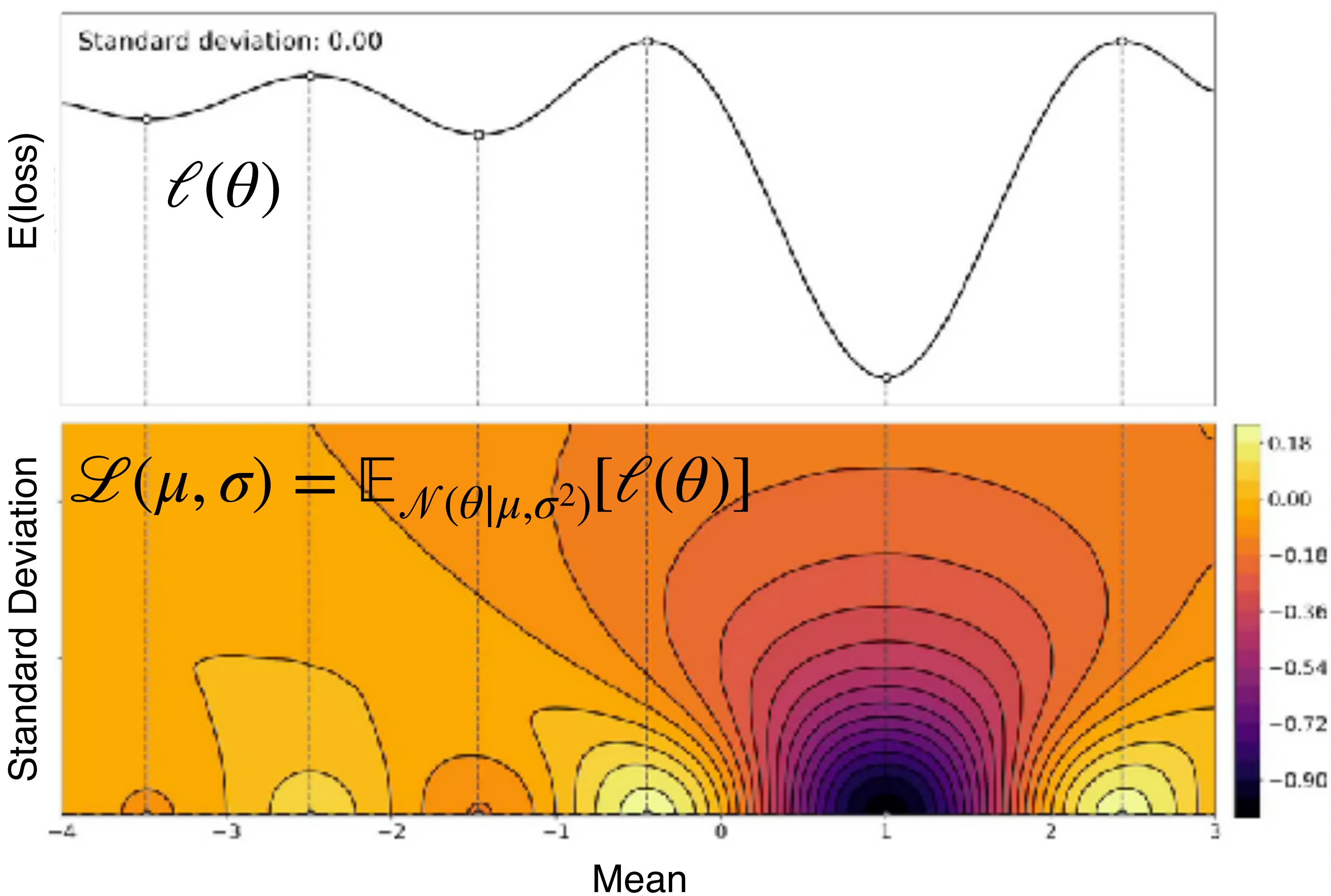
Learning Algorithm	Posterior Approx.	Natural-Gradient Approx.	Sec.
Optimization Algorithms			
Gradient Descent	Gaussian (fixed cov.)	Delta method	1.3
Newton's method	Gaussian	——"——	1.3
Multimodal optimization (New)	Mixture of Gaussians	——"——	3.2
Deep-Learning Algorithms			
Stochastic Gradient Descent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

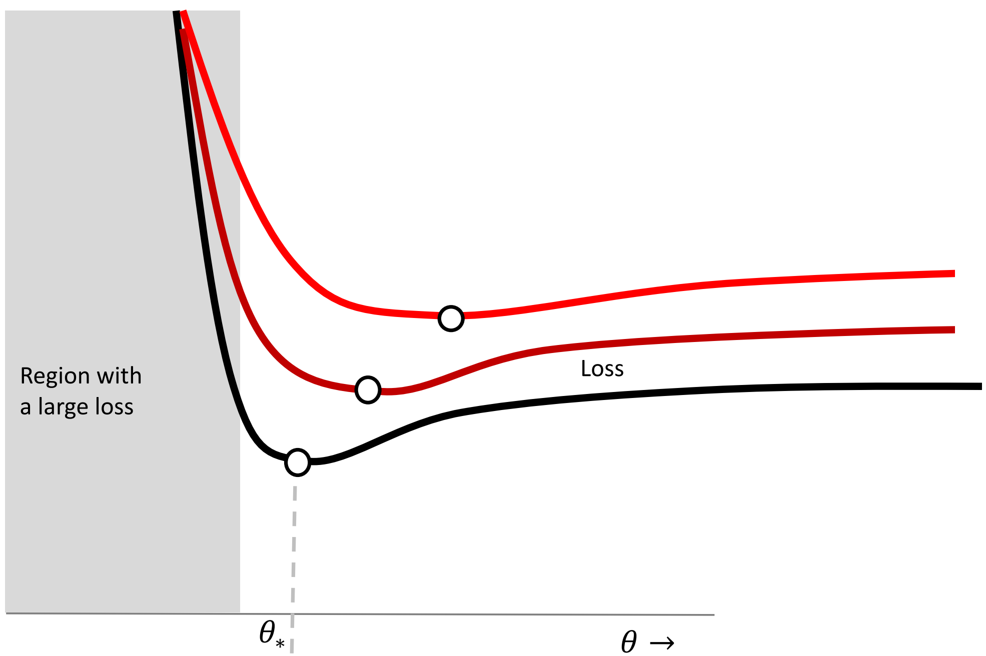
Bayes Prefers Flatter directions

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

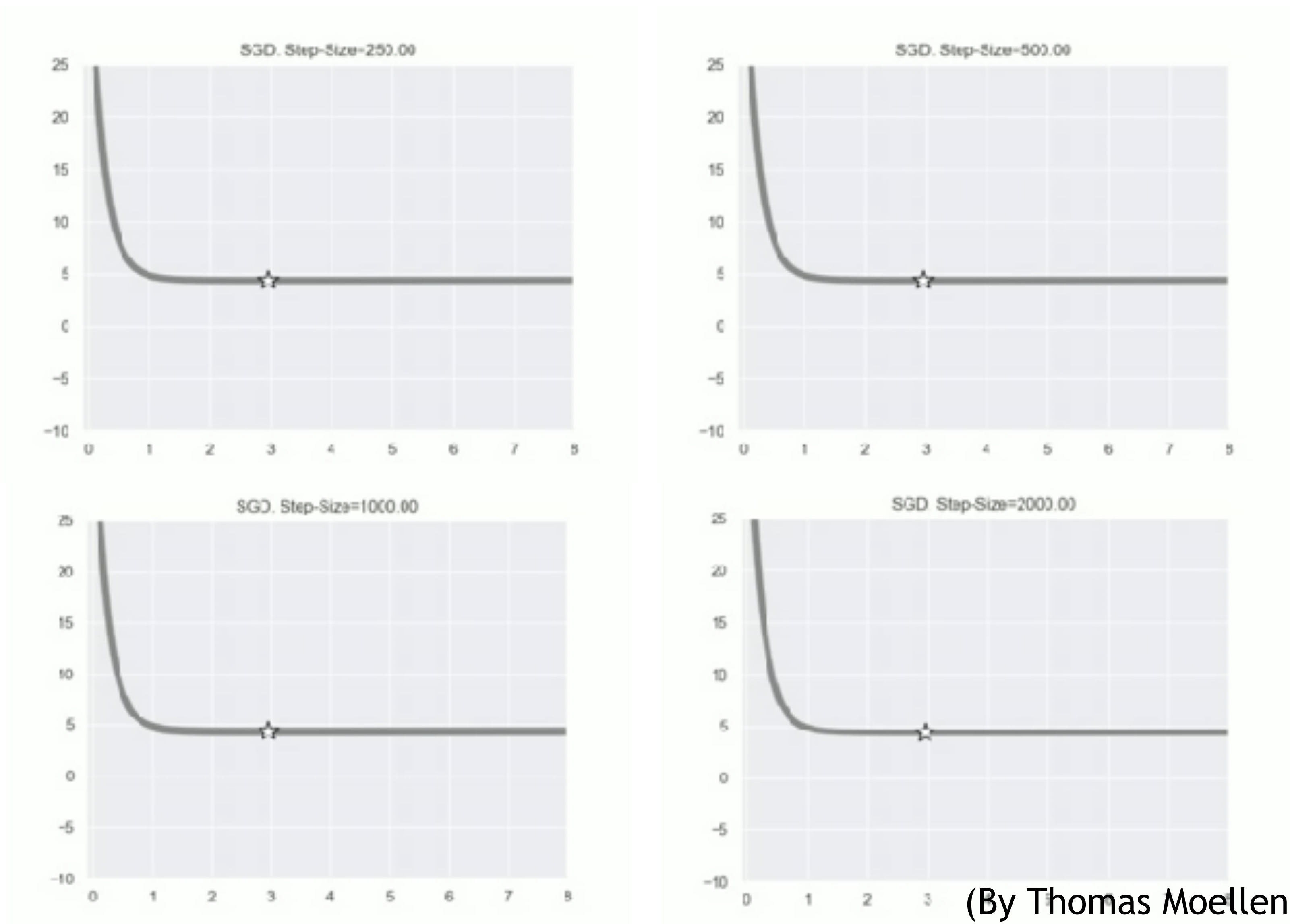
BLR: $m \leftarrow m - \rho \nabla_m \mathbb{E}_q[\ell(\theta)] \implies \nabla_m \mathbb{E}_{q_*}[\ell(\theta)] = 0$

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

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



SGD: Implicit Regularization

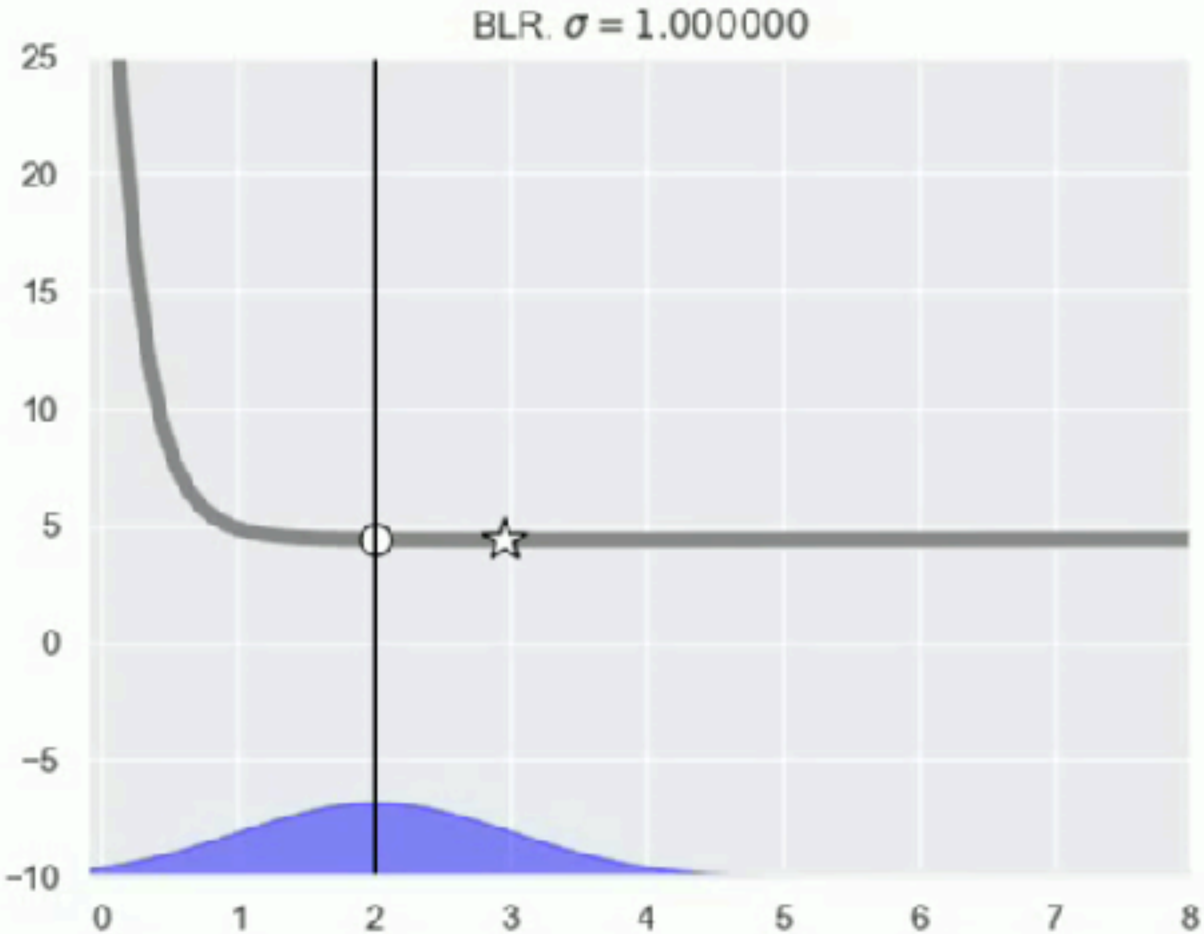


(By Thomas Moellenhoff)

Bayes: Explicit Regularization

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

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



By increasing the variance, we can move the mode arbitrarily far.

Bayesian"noise" has a similar regularization to the SGD noise.

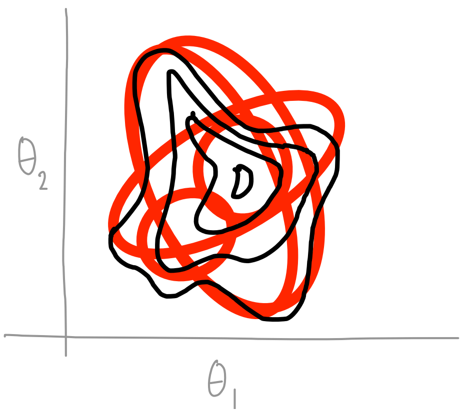
It prefers "flatter" directions.

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

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

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

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

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

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

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

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)

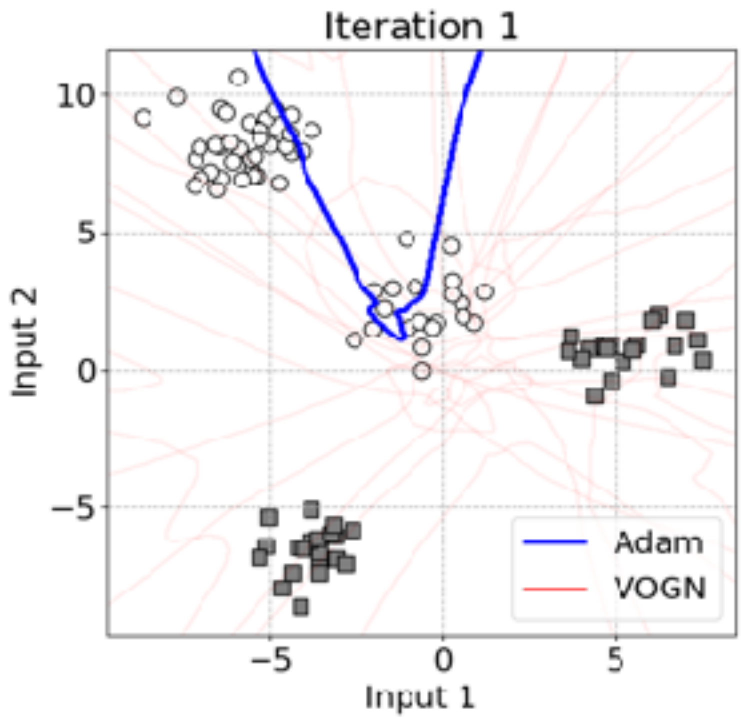
$$g \leftarrow \hat{\nabla}\ell(\theta), \text{ where } \theta \sim \mathcal{N}(m, \sigma^2)$$
$$s \leftarrow (1-\rho)s + \rho(\Sigma_i g_i^2)$$
$$m \leftarrow m - \alpha(s+\gamma)^{-1}\nabla_\theta \ell(\theta)$$
$$\sigma^2 \leftarrow (s+\gamma)^{-1}$$

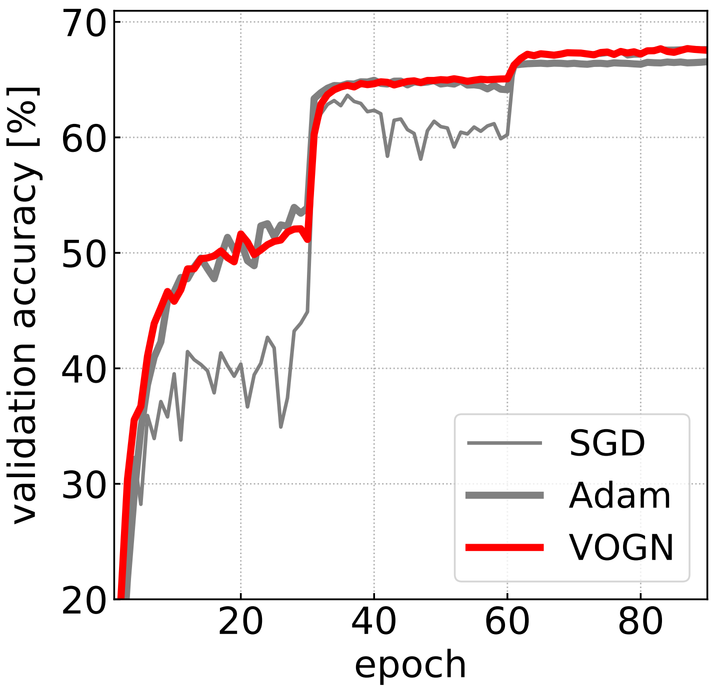
Available at https://github.com/team-approx-bayes/dl-with-bayes

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=LQInlN5EU7E.

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

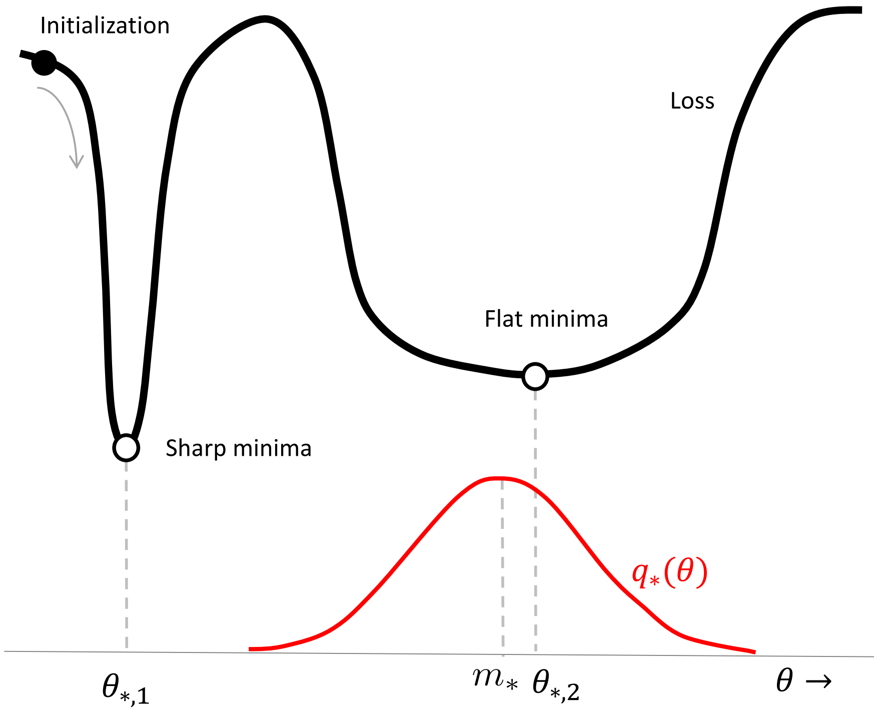
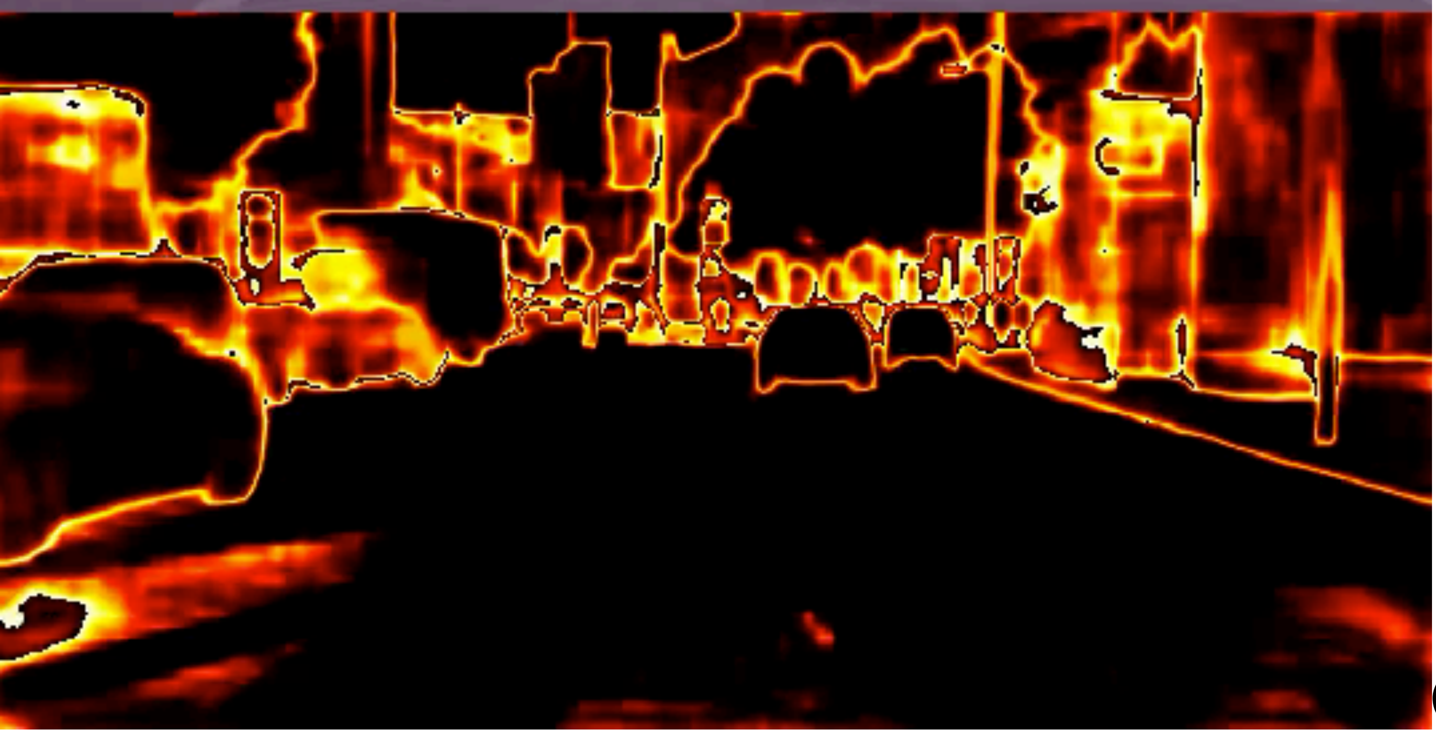


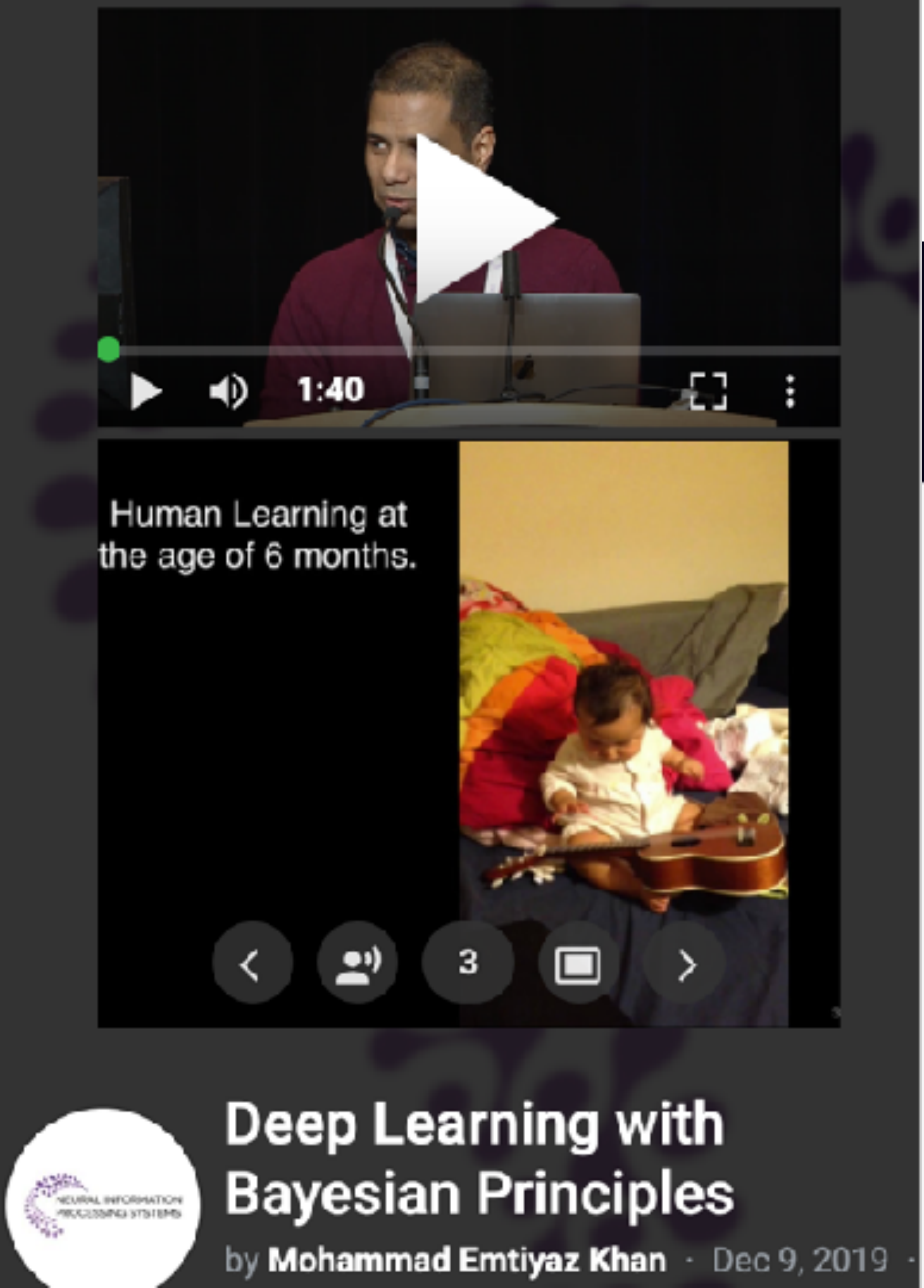
Image Segmentation

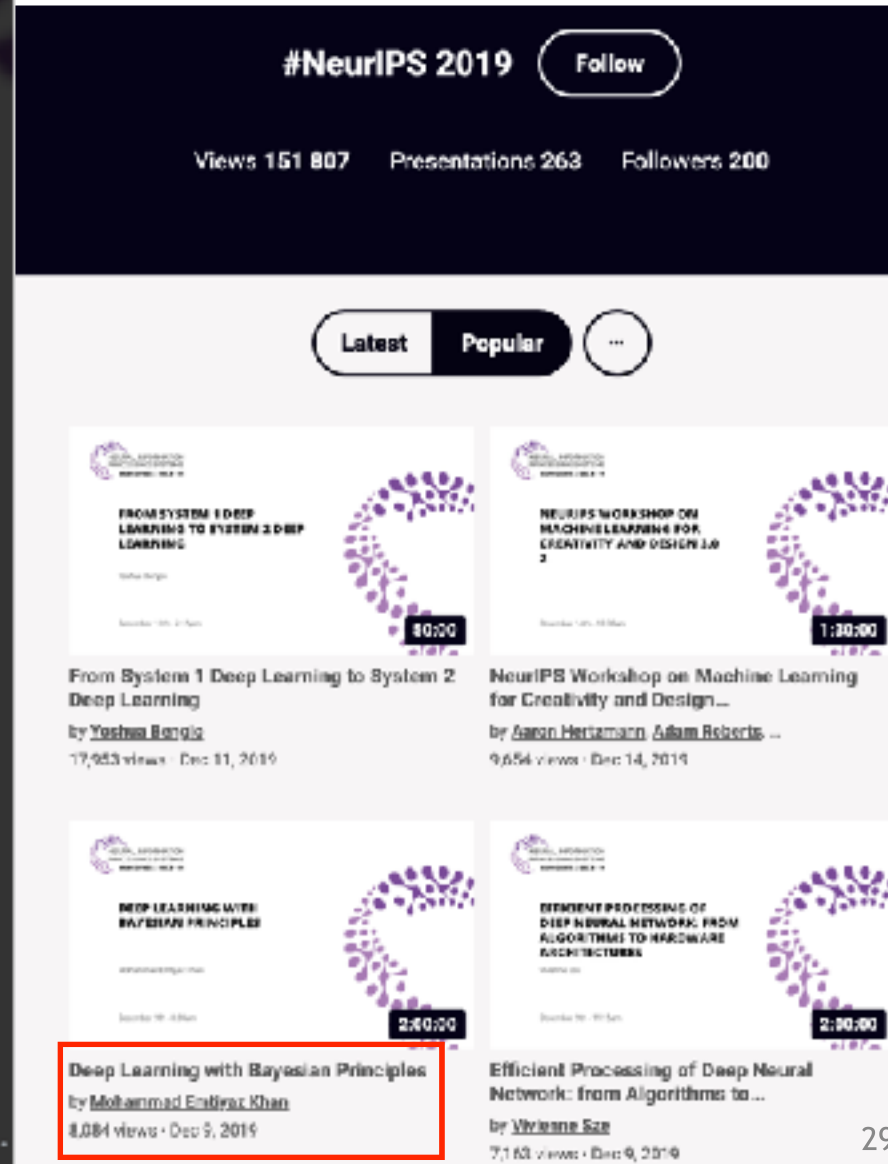


Uncertainty (entropy of class probs)

(By Roman Bachmann)

NeurIPS 2019 Tutorial





Robustness

Good algorithms can tell apart
relevant vs irrelevant information

Perturbation, Sensitivity, and Duality

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

See Section 5.4 in Khan and Rue, 2021 for local parameterization
See Section 3 in ADAM et al. 2021 for dual parameterization

BLR Solutions & Their Duality

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

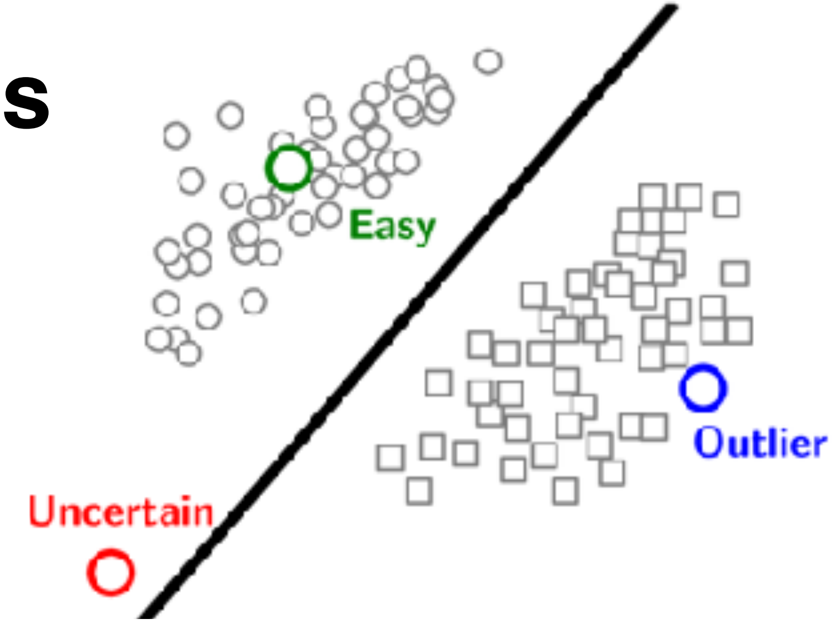
$$\lambda^* = \sum_{i=0}^{N} \underbrace{\nabla_{\mu^*} \mathbb{E}_{q^*}[-\ell_i(\theta)]}_{\widetilde{\lambda}_i^*}$$

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.

1. ADAM, Chang, Khan, Solin, Dual parameterization of SVGP, NeurIPS, 2021

Memorable Experiences

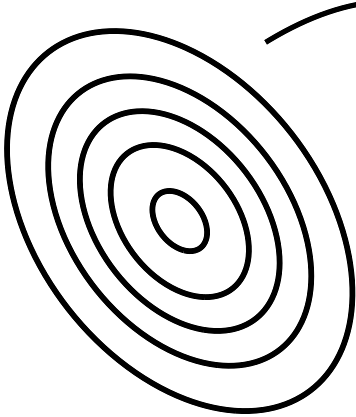


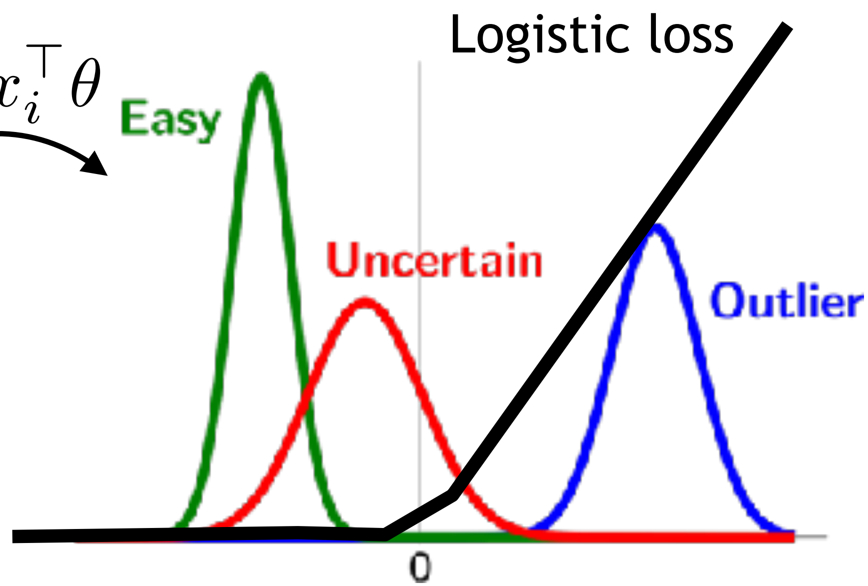
$$\lambda^* = \sum_{i=0}^{N} \underbrace{\nabla_{\mu^*} \mathbb{E}_{q^*}[-\ell_i(\theta)]}_{\widetilde{\lambda}_i^*}$$

"Global" posterior $q(\theta)$

Local predictions $q(f_i)$

$f_i = x_i^\top \theta$





Lower Sensitivity to easy example.

Such sensitivity analysis leads to memorable experiences

Memorable Experiences

1. Schneider et al. DeepOBS: A Deep Learning Optimizer Benchmark Suite". ICLR 2018

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”

1. Tailor, Chang, Swaroop, Tangkaratt, Solin, Khan. Memorable experiences of ML models (in preparation)

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

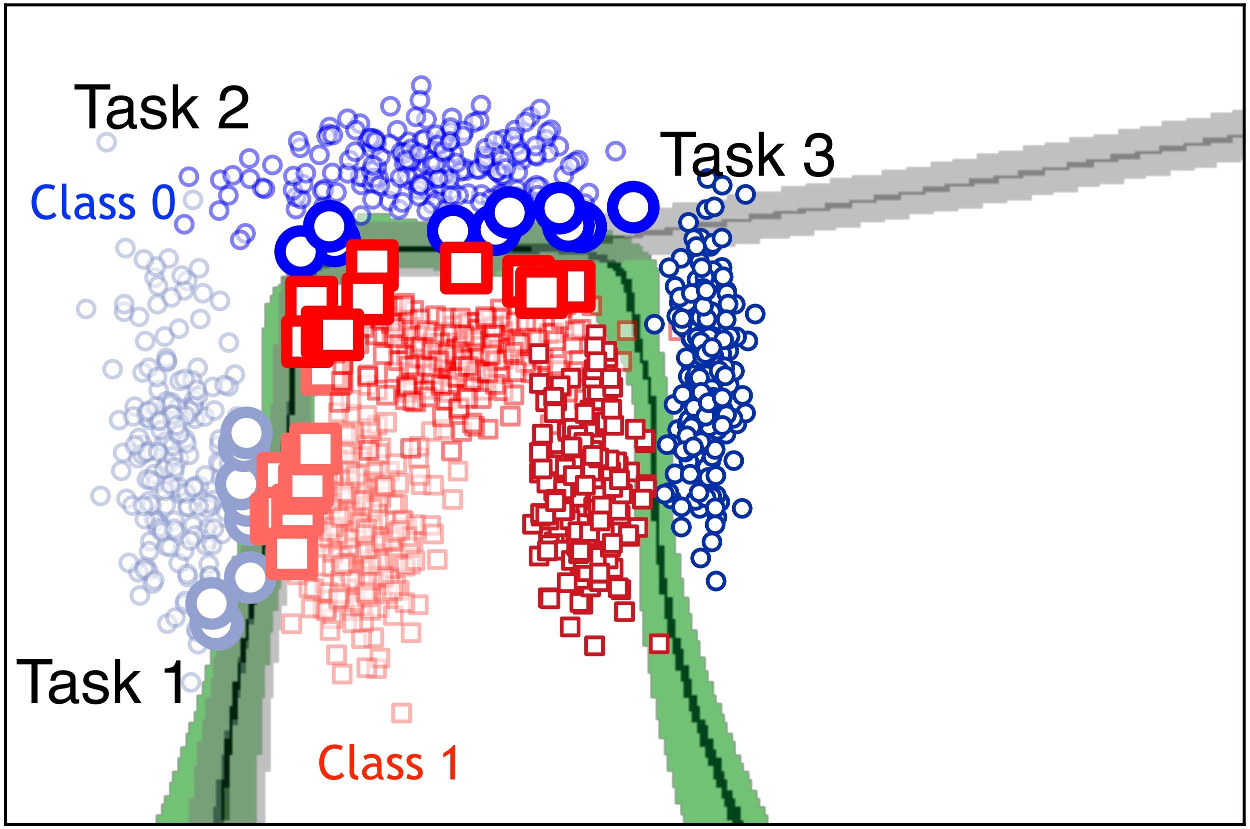
Received total funding of around USD 3 million through JST's CREST-ANR and Kakenhi Grants.

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]

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

New data

Weight-regularizer

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

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

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

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

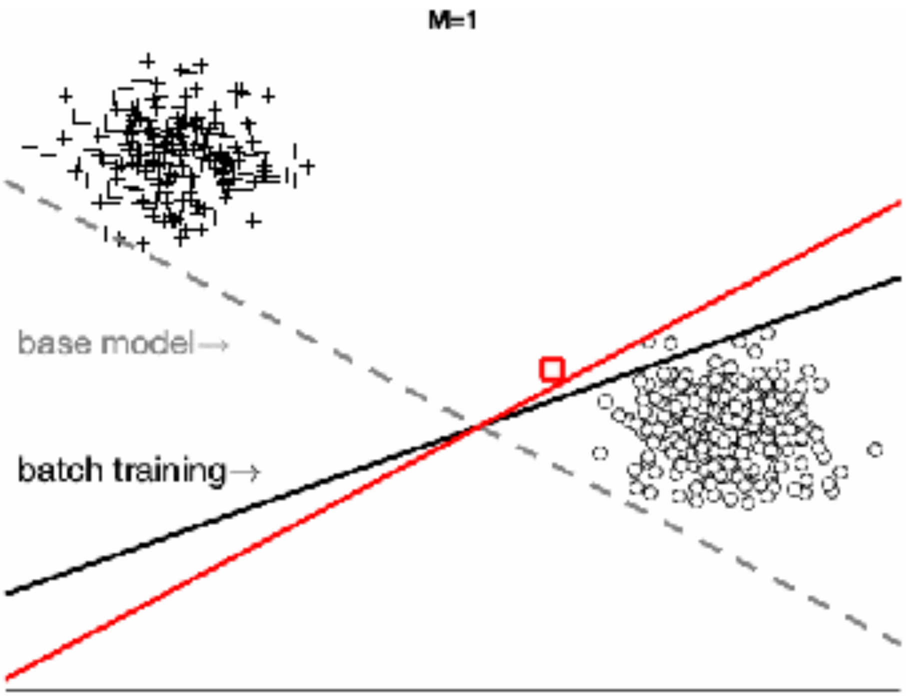
See Section 3 and 4, and App A in [2]

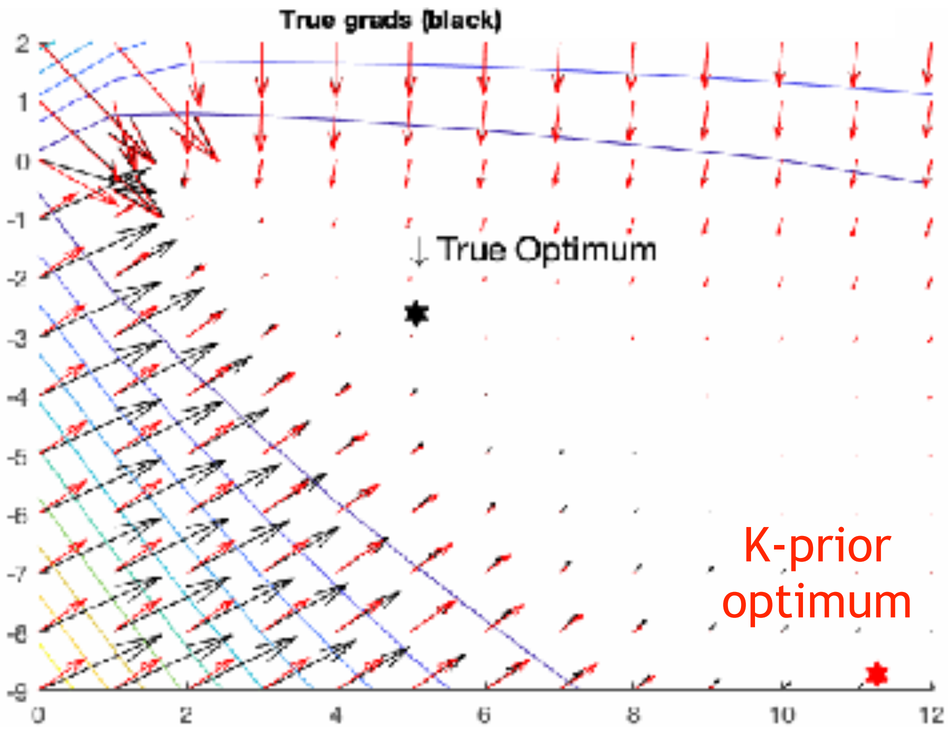
K-Priors and Bayes-Duality

- 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

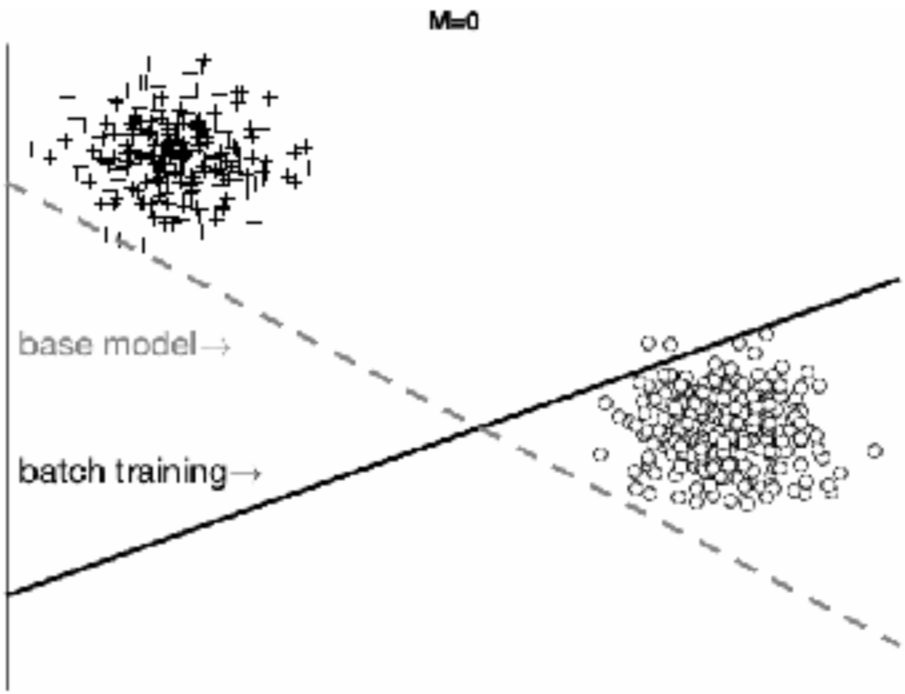
1. Khan et al. Approximate Inference Turns Deep Networks into Gaussian Process, NeurIPS, 2019
2. Khan and Swaroop. Knowledge-Adaptation Priors, NeurIPS, 2021 (https://arxiv.org/abs/2106.08769)

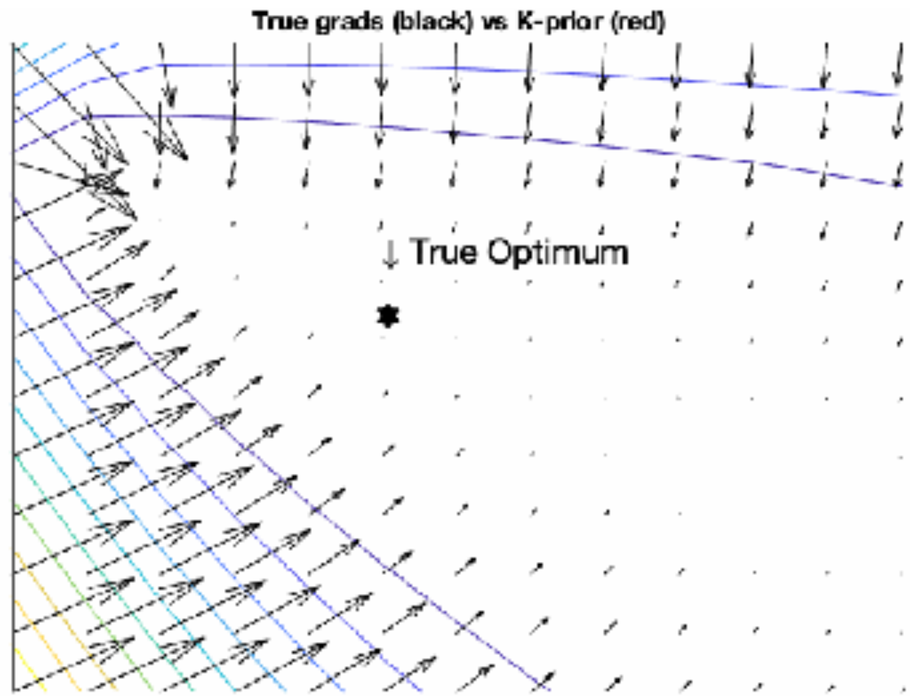
Faithful Gradient Reconstruction





Faithful Gradient Reconstruction





No labels required, so $\mathscr{M}$ 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

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

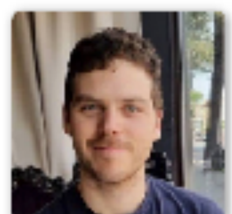
Erik Daxberger
Remote Collaborator
University of Cambridge

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