Bayesian Principles for Learning-Machines

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

AI that learn like humans

Learn and adapt quickly 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

Bayesian Principles

This talk

Human learning $\neq$ Deep learning

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

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

My current research focuses on reducing this gap!

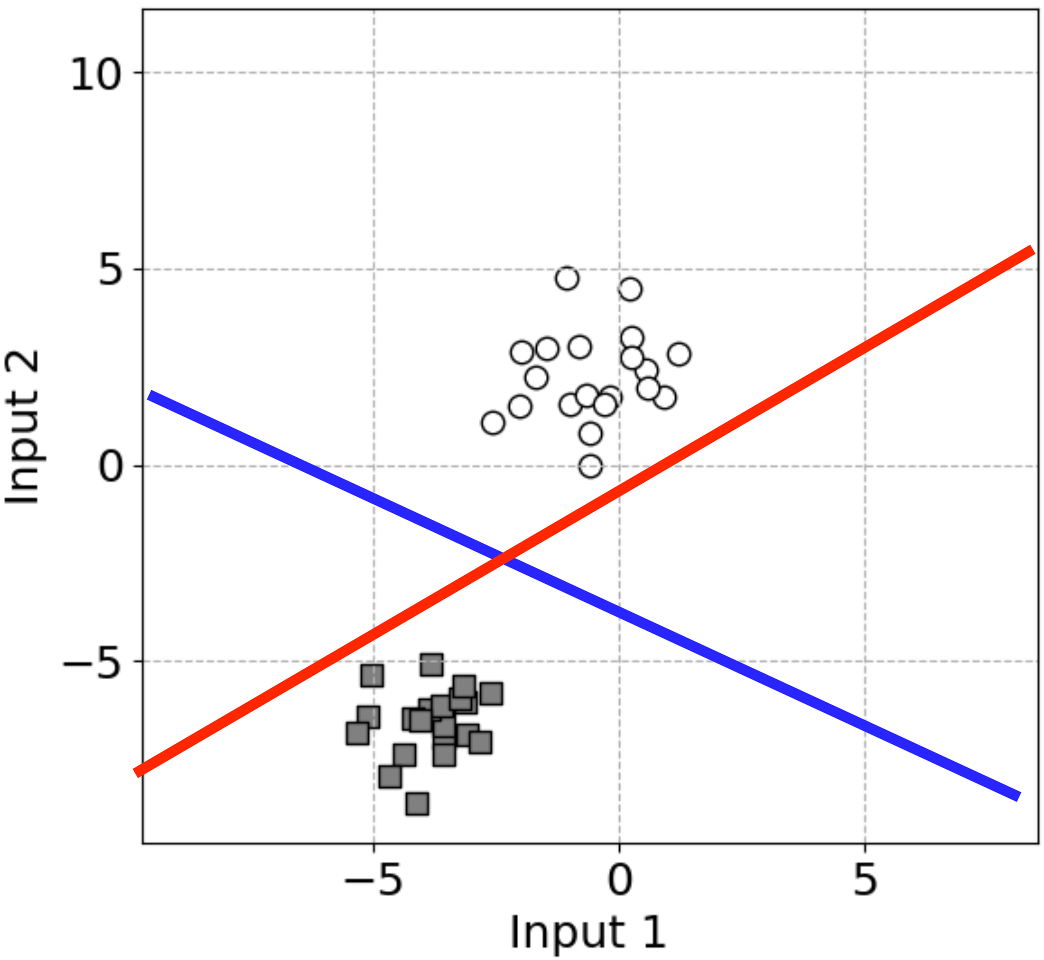
1. Parisi, German I., et al. "Continual lifelong learning with neural networks: A review." *Neural Networks* (2019)

2. Geisler, W. S., and Randy L. D. "Bayesian natural selection and the evolution of perceptual systems." *Philosophical Transactions of the Royal Society of London. Biological Sciences* (2002)

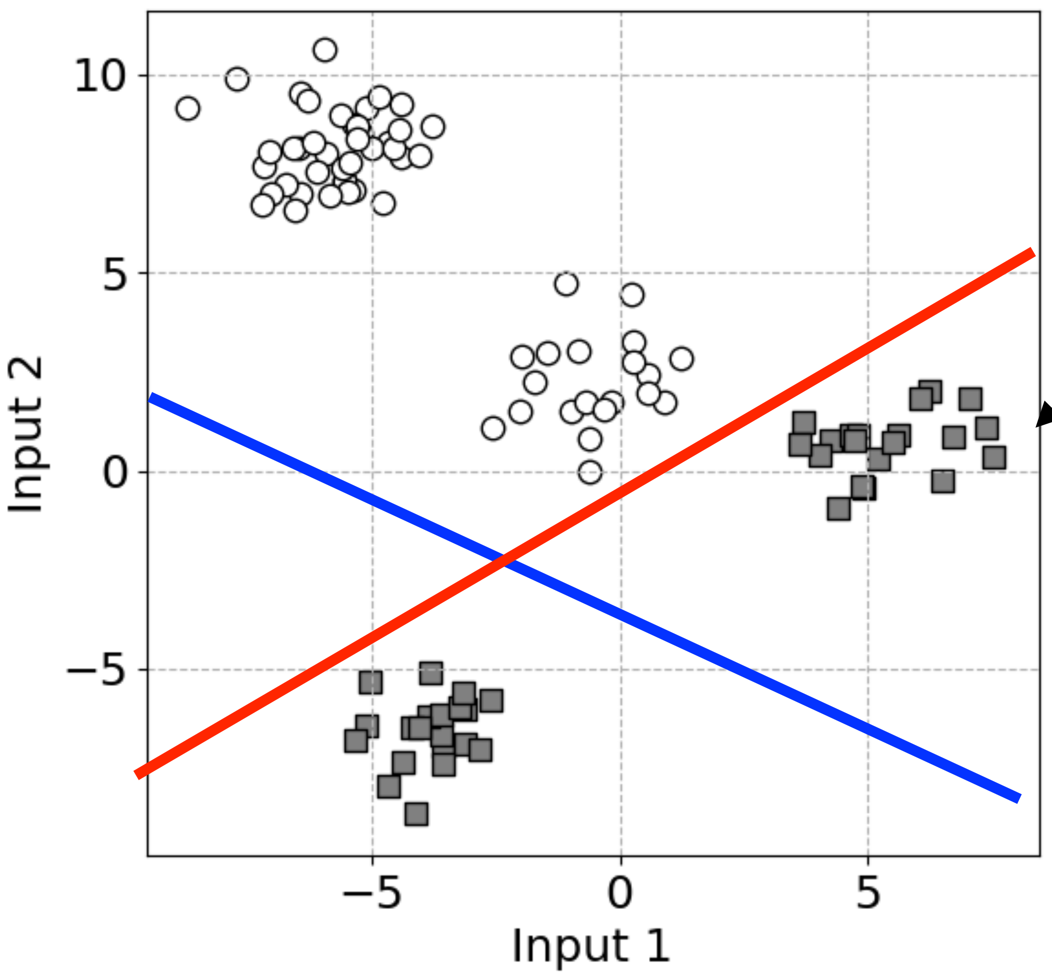
Bayesian (Principles for) Learning Machines

- Uncertainty
  - What you don’t know now, can hurt you later
- Learning
  - Derive learning-algorithms from Bayes
- Knowledge
  - Extract knowledge as memorable examples

Which is a good classifier?



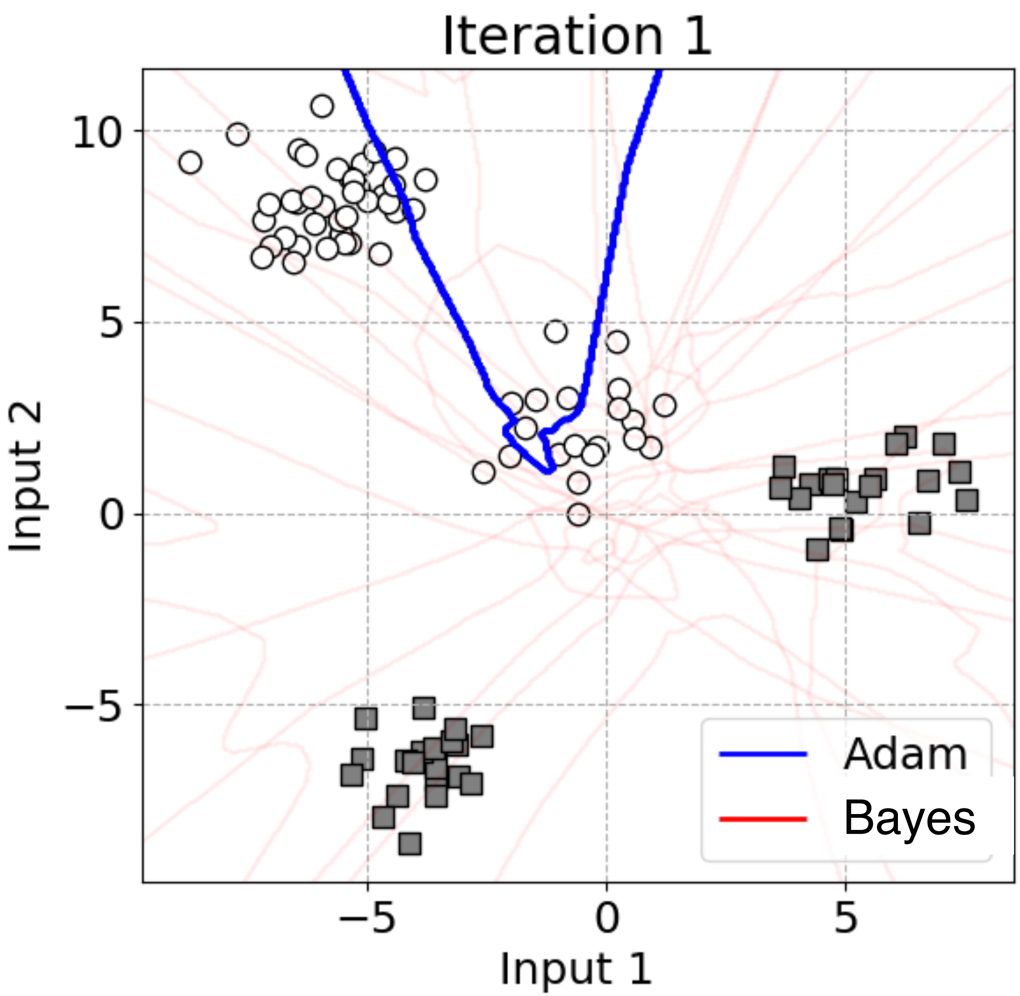
Which is a good classifier?



Misclassified by the red line, but not by the blue

What you don't know now, can hurt you later
"Uncertainty matters"

Uncertainty of Deep Nets



One Model vs Many.

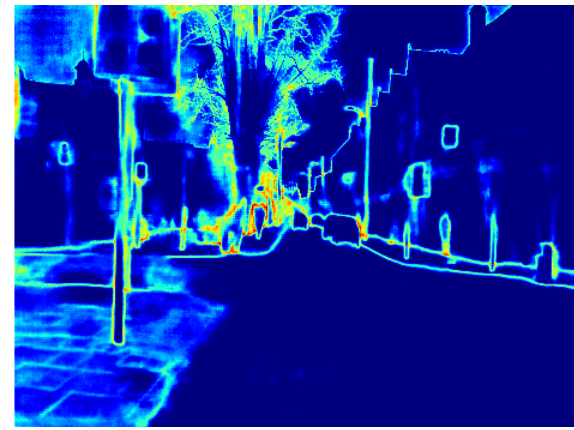
A key idea in Bayes is to estimate distributions over model parameters (e.g., Gaussian).

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

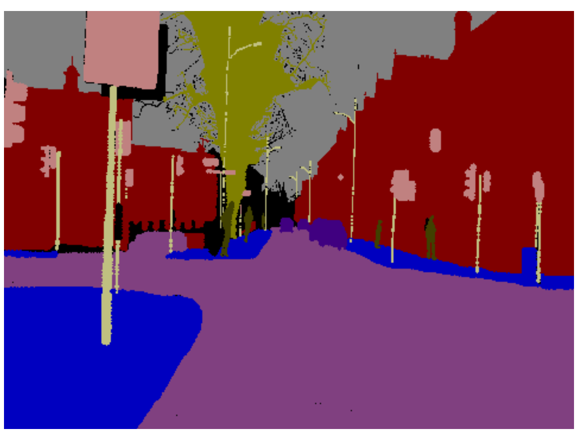
Image Segmentation

Image

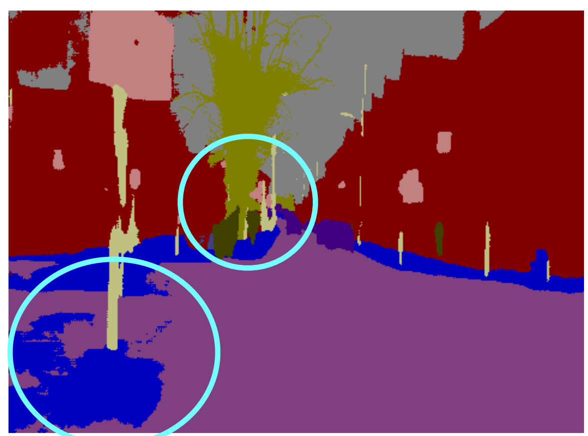
Uncertainty

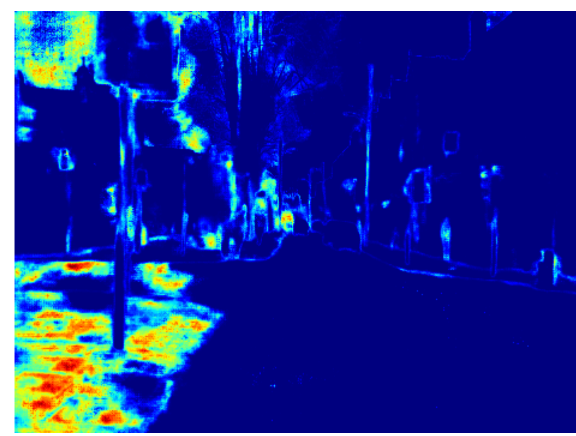


True Segments



Prediction

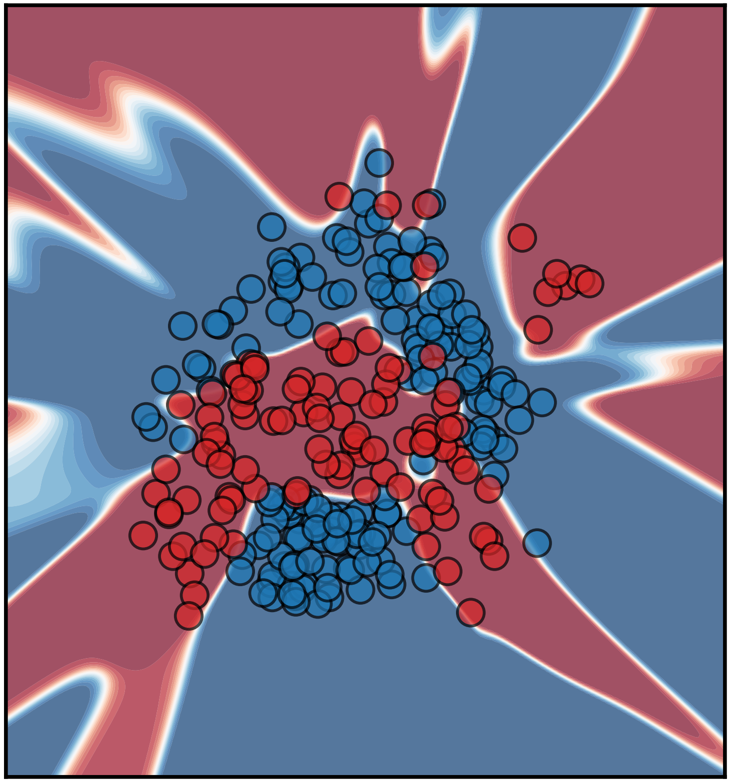




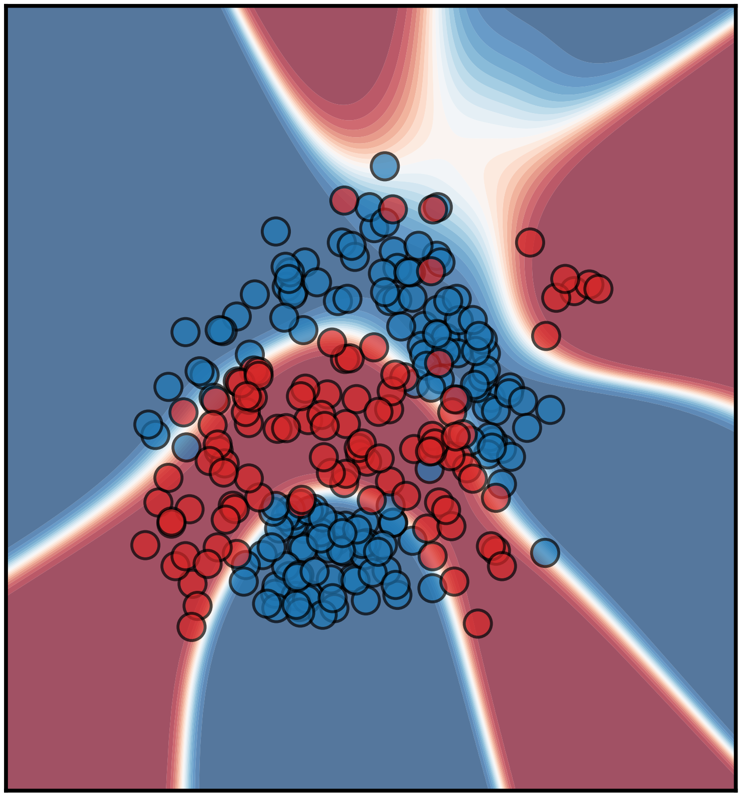
Kendall, Alex, Yarin Gal, and Roberto Cipolla. "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics." *CVPR*. 2018.

Reduce Overfitting

Standard DL



Bayesian DL



Left figure is cross-validation. Right figure is“Marginal Likelihoods”.

Bayesian (Principles for) Learning Machines

- Uncertainty
  - What you don’t know now, can hurt you later
- Learning
  - Derive learning-algorithms from Bayes
- Knowledge
  - Extract knowledge as memorable examples

Bayesian Learning Rule

Entropy

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

Exponential-family Approx.

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

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

Natural Gradient

Natural and Expectation parameters of an exponential family distribution q

By changing *Q*, we can recover DL algorithms (and more)

1. Khan and Lin. "Conjugate-computation variational inference: Converting variational inference in non-conjugate models to inferences in conjugate models." AIstats (2017).
2. Khan and Rue. "Learning-Algorithms from Bayesian Principles" (2020) (work in progress, an early draft available at https://emtiyaz.github.io/papers/learning_from_bayes.pdf)

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

Learning Algorithm	Posterior Approx.	Algorithmic Approx.	Sec.
Optimization Algorithms			
Gradient Descent	Gaussian (fixed cov.)	Delta approx.	1.4
Newton's method	Gaussian	——"——	1.4
Multimodel optimization (New)	Mixture of Gaussians	——"——	3.2
Deep-Learning Algorithms			
Stochastic Gradient Descent	Gaussian (fixed cov.)	Delta approx., Stochastic approx.	4.1
RMSprop/Adam	Gaussian (diagonal cov.)	Delta approx., Stochastic approx., Hessian approx., Square-root scaling, Slow-moving scale vectors	4.2, 4.3
Dropout	Mixture of Gaussians	Delta approx., Stochastic approx., Responsibility approx.	4.4
STE	Bernoulli	Delta approx., Stochastic approx.	4.6
Online Gauss-Newton (OGN) (New)	Gaussian (diagonal cov.)	Gauss-Newton Hessian approx. in Adam & no square-root scaling	4.5
Variational OGN (New)	——"——	Remove Delta approx. from OGN	4.5
Bayesian Binary NN (New)	——"——	Remove Delta approx. from STE	4.6
Approximate Bayesian Inference Algorithms			
Conjugate Bayes	Exp-family	Set learning rate $\rho_t = 1$	5.1
Laplace's method	Gaussian	Delta approx.	5.2
Expectation-Maximization	Exp-Family + Gaussian	Delta approx. for the parameters	5.3
Stochastic VI (SVI)	Exp-family (mean-field)	Stochastic approx., local rate $\rho_t = 1$	5.4
VMP	——"——	Set learning rate $\rho_t = 1$	5.4
Non-Conjugate VMP	——"——	——"——	5.4
Non-Conjugate VI (New)	Mixture of Exp-family	None	5.5

Khan and Rue. "Learning-Algorithms from Bayesian Principles" (2020)

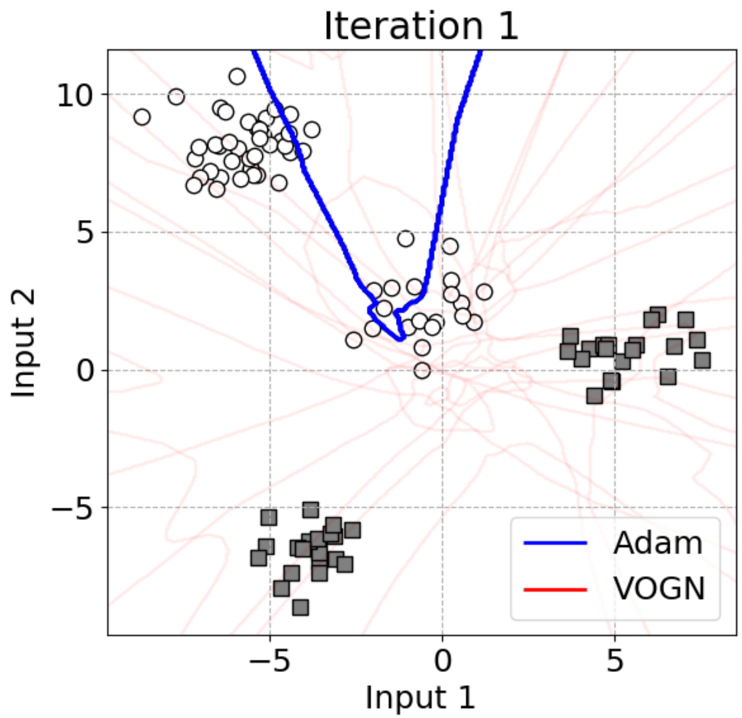
Work in progress (draft available at https://emtiyaz.github.io/papers/learning_from_bayes.pdf)

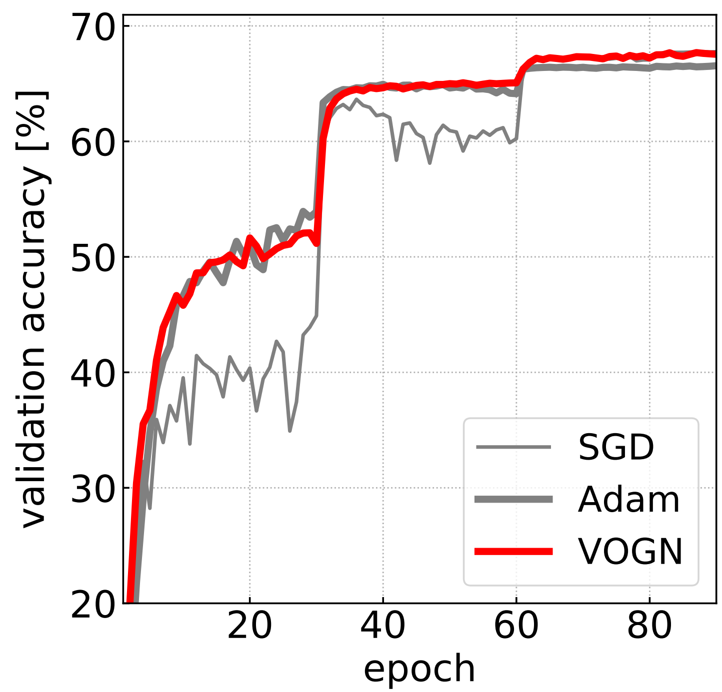
We can compute uncertainty using a variant of Adam.

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).

Uncertainty of Deep Nets

VOGN: A modification of Adam but match the performance on ImageNet

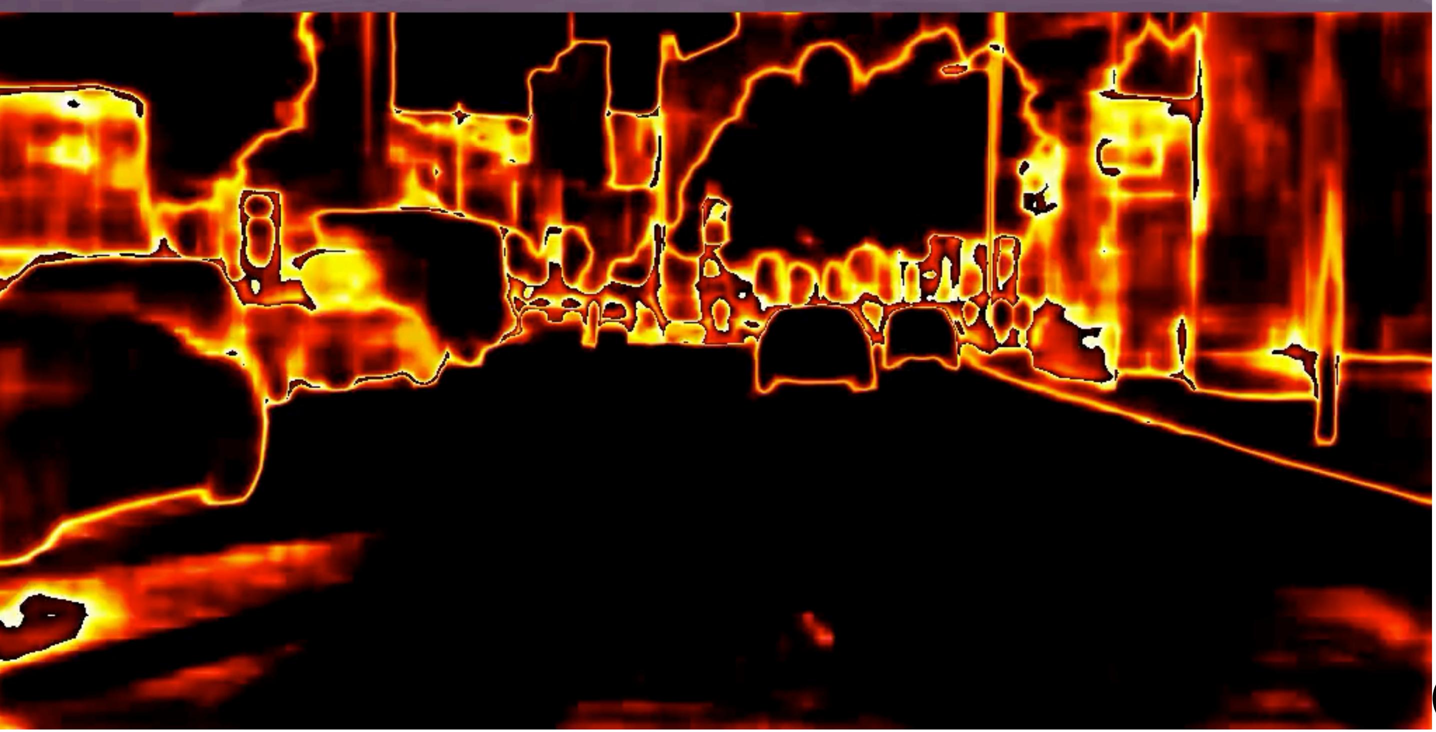




Code available at https://github.com/team-approx-bayes/dl-with-bayes

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).

Image Segmentation



Uncertainty (entropy of class probs)

(By Roman Bachmann)

Learning-Algorithms from Bayesian Principles

Mohammad Emtiyaz Khan
RIKEN center for Advanced Intelligence Project
Tokyo, Japan

Håvard Rue
CEMSE Division
King Abdullah University of Science and Technology
Thuwal, Saudi Arabia

Version of November 3, 2020
DRAFT ONLY

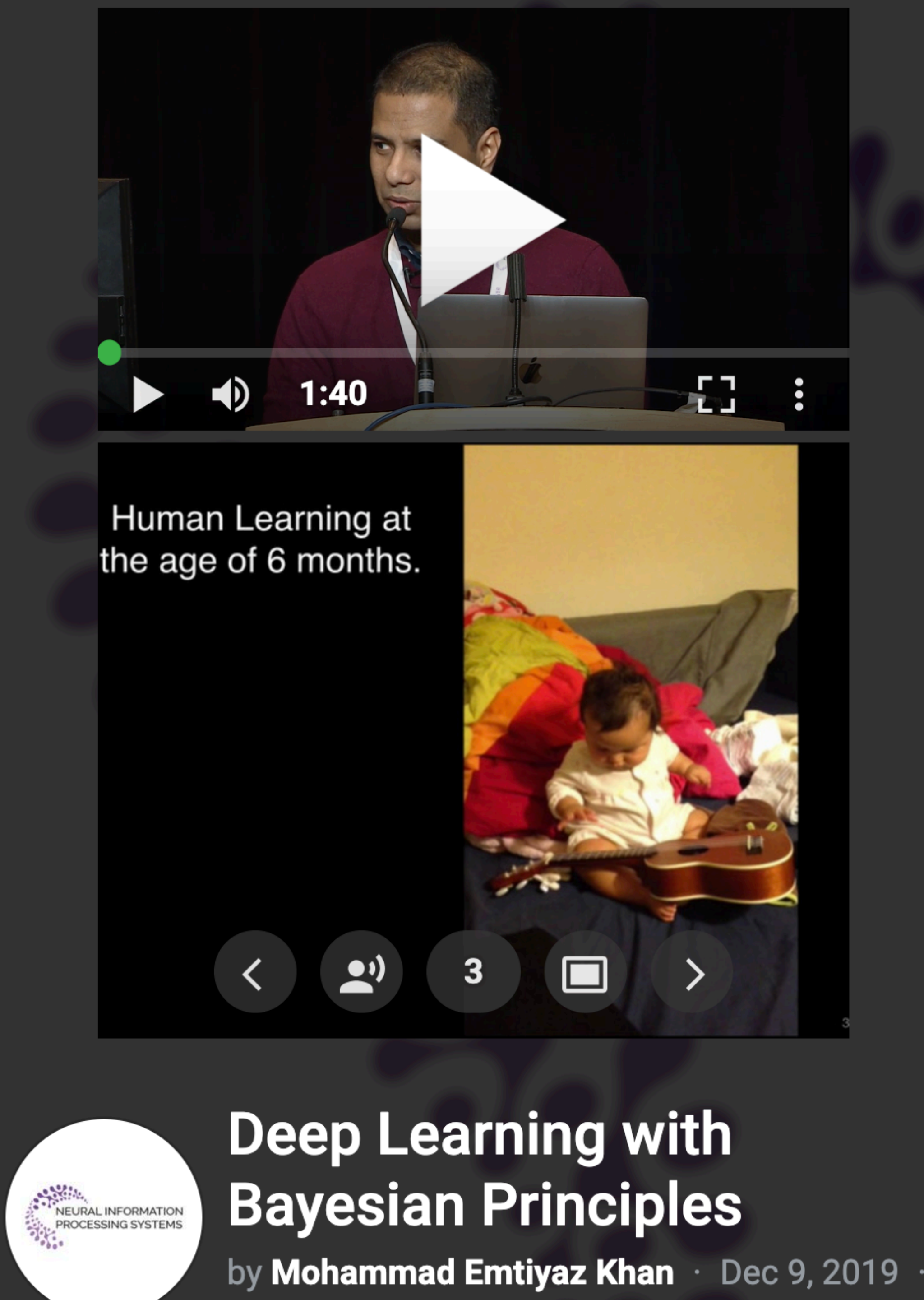
Abstract

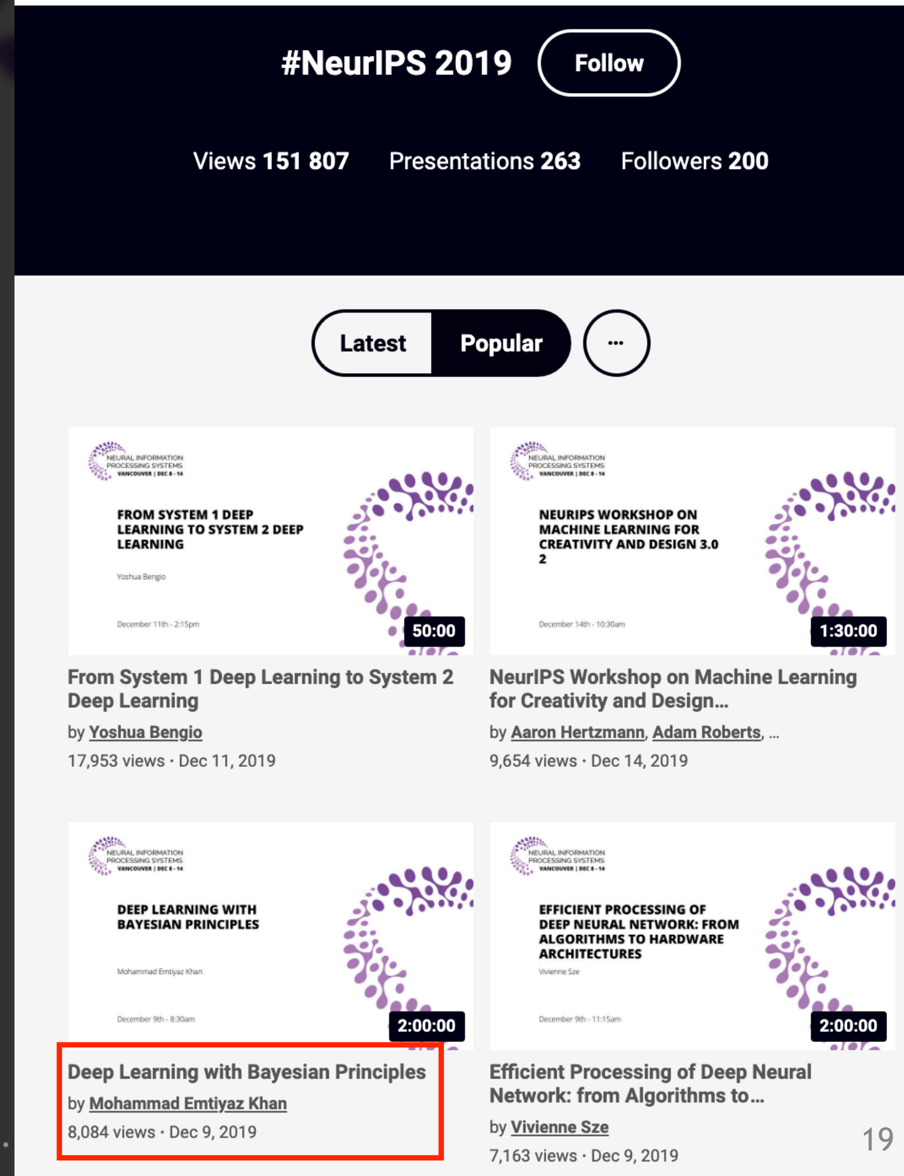
We show that many machine-learning algorithms are specific instances of a *single* algorithm called the Bayesian learning rule. The rule, derived from Bayesian principles, yields a wide-range of algorithms from fields such as optimization, deep learning, and graphical models. This includes classical algorithms such as ridge regression, Newton's method, and Kalman filter, as well as modern deep-learning algorithms such as stochastic-gradient descent, RMSprop, Adam, and Dropout. The key idea is to estimate posterior approximations using the Bayesian learning rule. Different approximations then result in different algorithms and further algorithmic approximations give rise to variants of those algorithms. Our work shows that Bayesian principles not only unify, generalize, and improve existing learning-algorithms, but also help us design new ones.

Available at

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

NeurIPS 2019 Tutorial



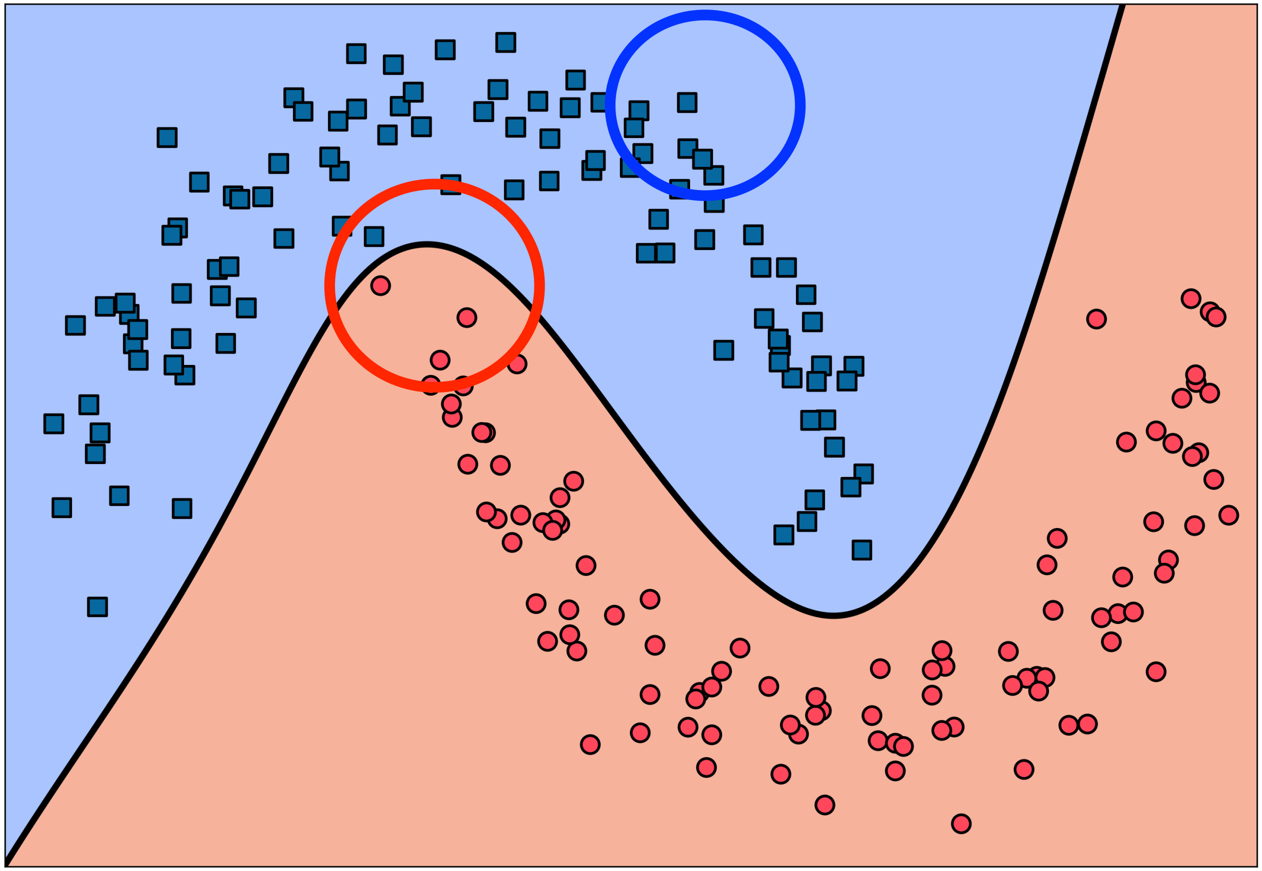


Bayesian (Principles for) Learning Machines

- Uncertainty
  – What you don’t know now, can hurt you later
- Learning
  – Derive learning-algorithms from Bayes
- Knowledge
  – Extract knowledge as memorable examples

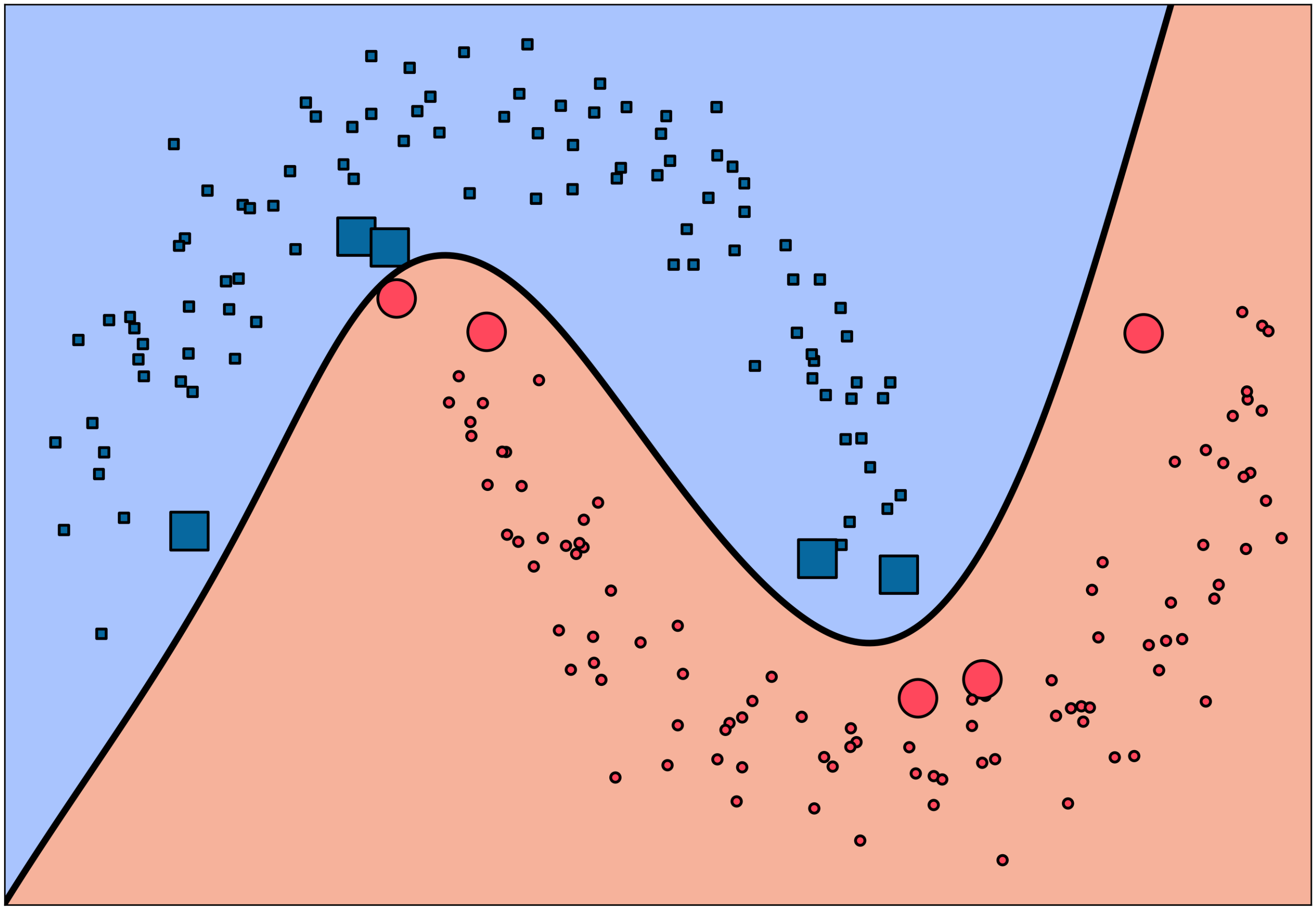
Relevance of Data Examples

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



Model view vs Data view

Bayes "automatically" defines data-relevance



Data view (Very much like SVMs)

(By Roman Bachmann)

Bayes Duality

- Gaussian approx fom Bayes learning rule turn NN into Linear models & Gaussian Process (GPs) [1].

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

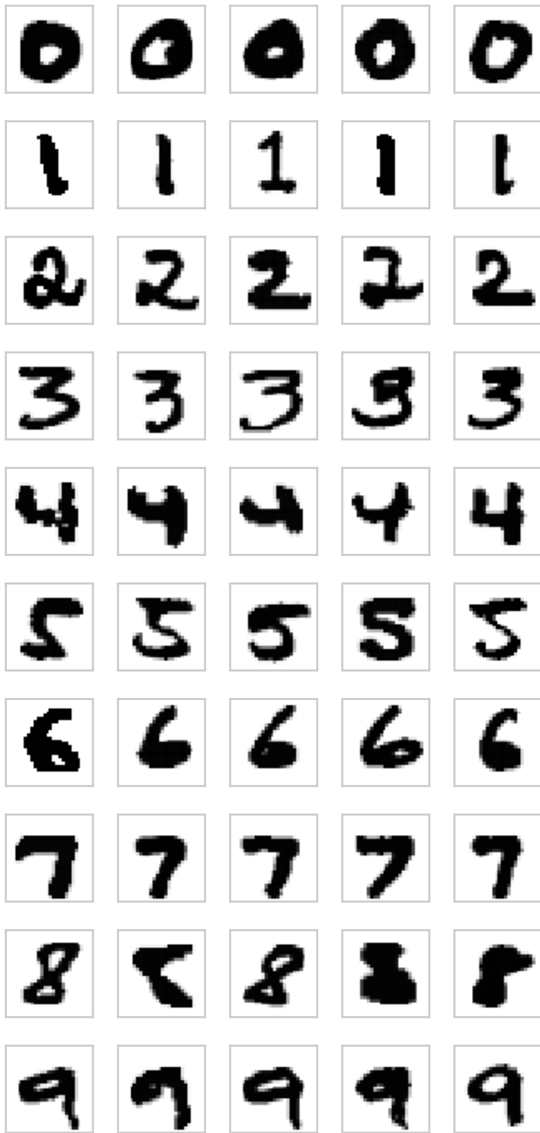
“Dual” variables obtained from $\nabla_\mu \mathbb{E}_q[\ell_i(\theta)]$
(For Gaussian approx, obtained from Jacobian, residual etc.)

- σ_i^2 define the “relevance” of the data examples. We call more relevant ones the “memorable examples”.

- Natural-gradients give “dual variables” (Bayes Duality)

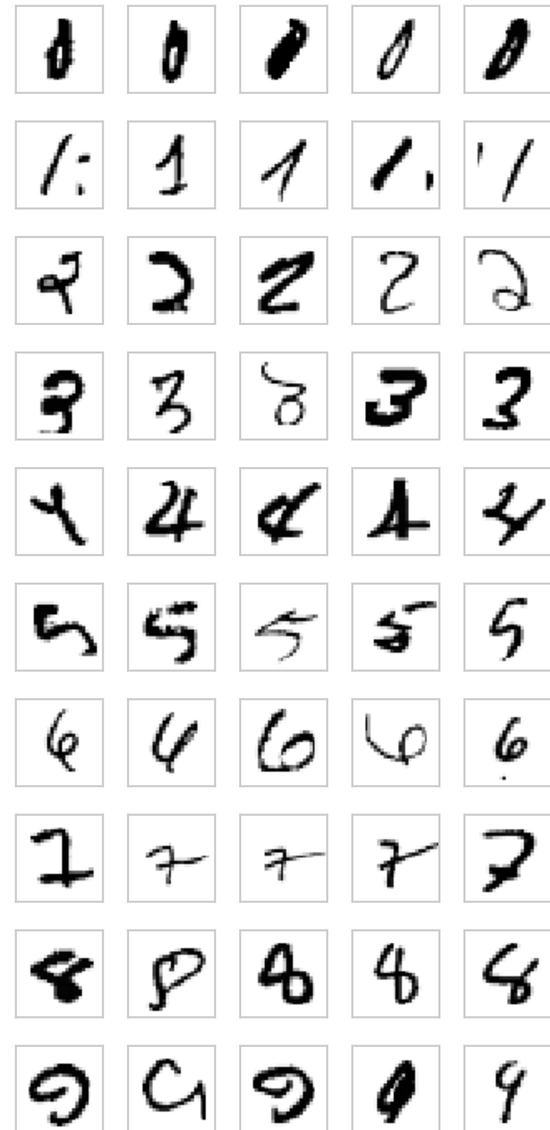
1. Khan et al. "Approximate Inference Turns Deep Networks into Gaussian Processes." *NeurIPS* (2019).

Least Memorable



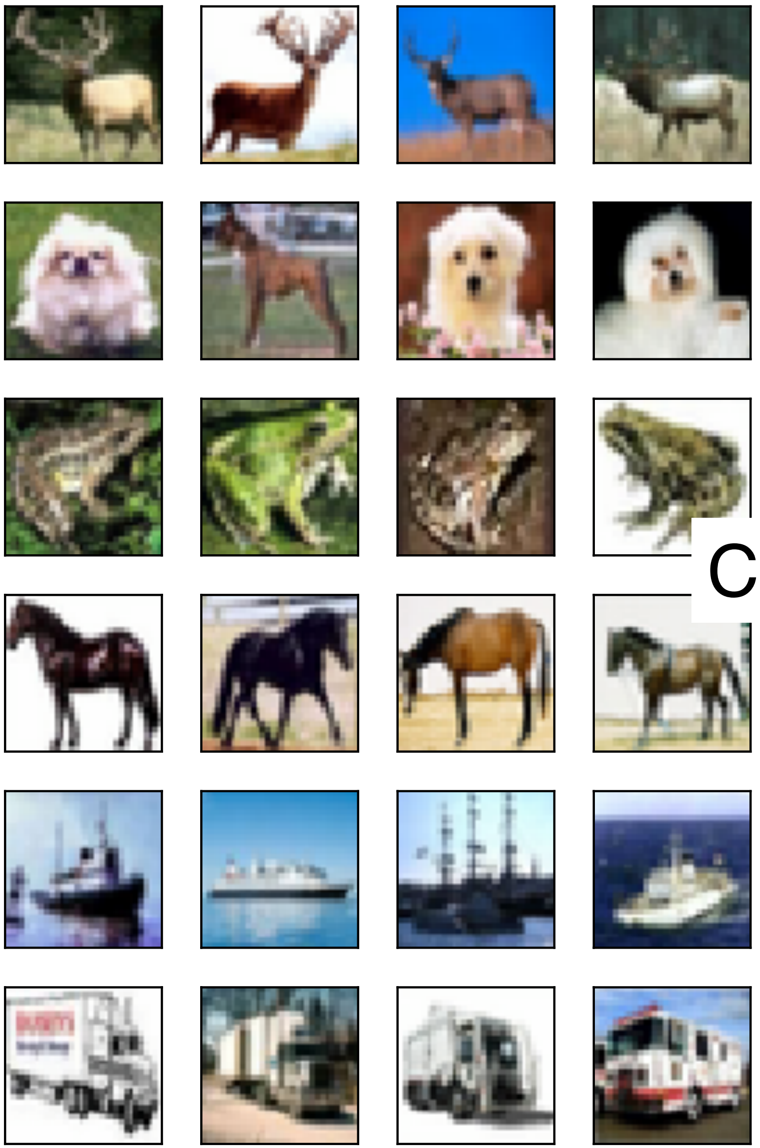
MNIST

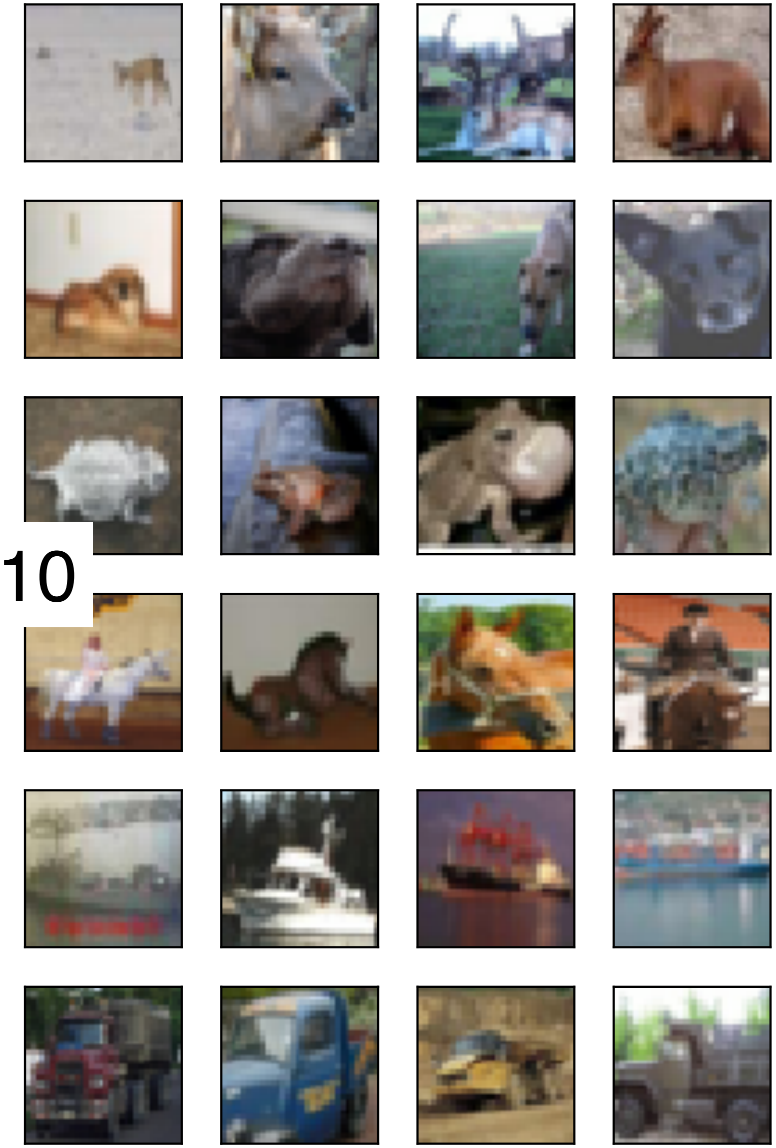
Most Memorable



Least Memorable

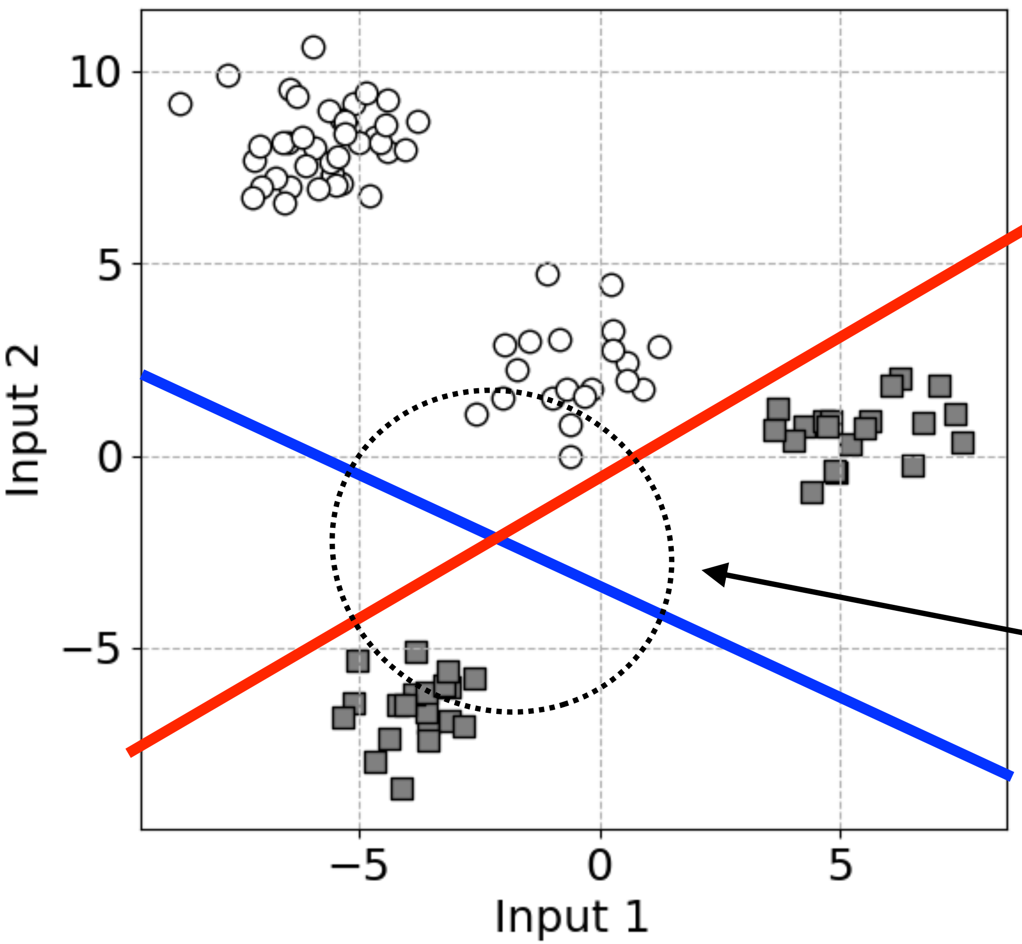
Most Memorable





CIFAR-10

Life-Long Learning with Bayes

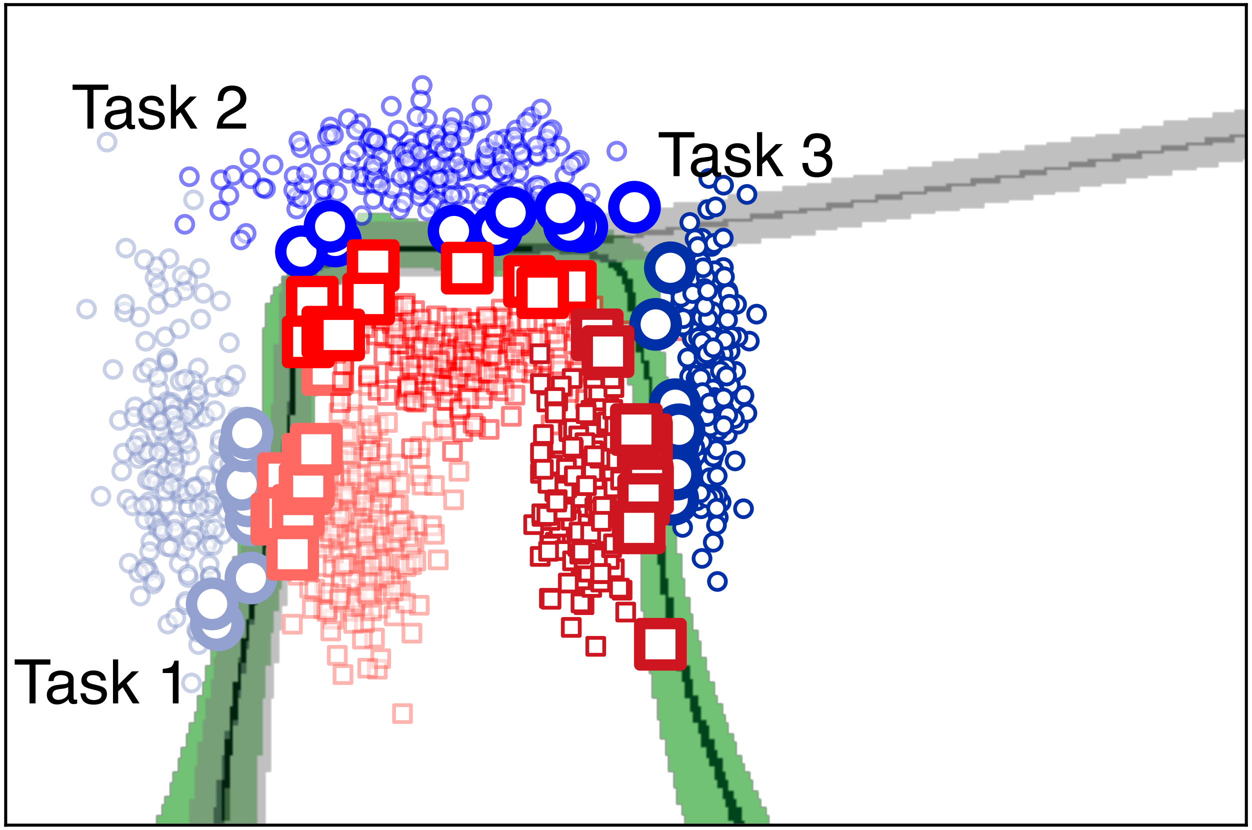


Identify memorable examples, and make sure that predictions don't change too much

1. Kirkpatrick, James, et al. "Overcoming catastrophic forgetting in neural networks." *PNAS 2017*
2. *Pan et al. Continual Deep Learning by Functional Regularisation of Memorable Past, NeurIPS, 2020*

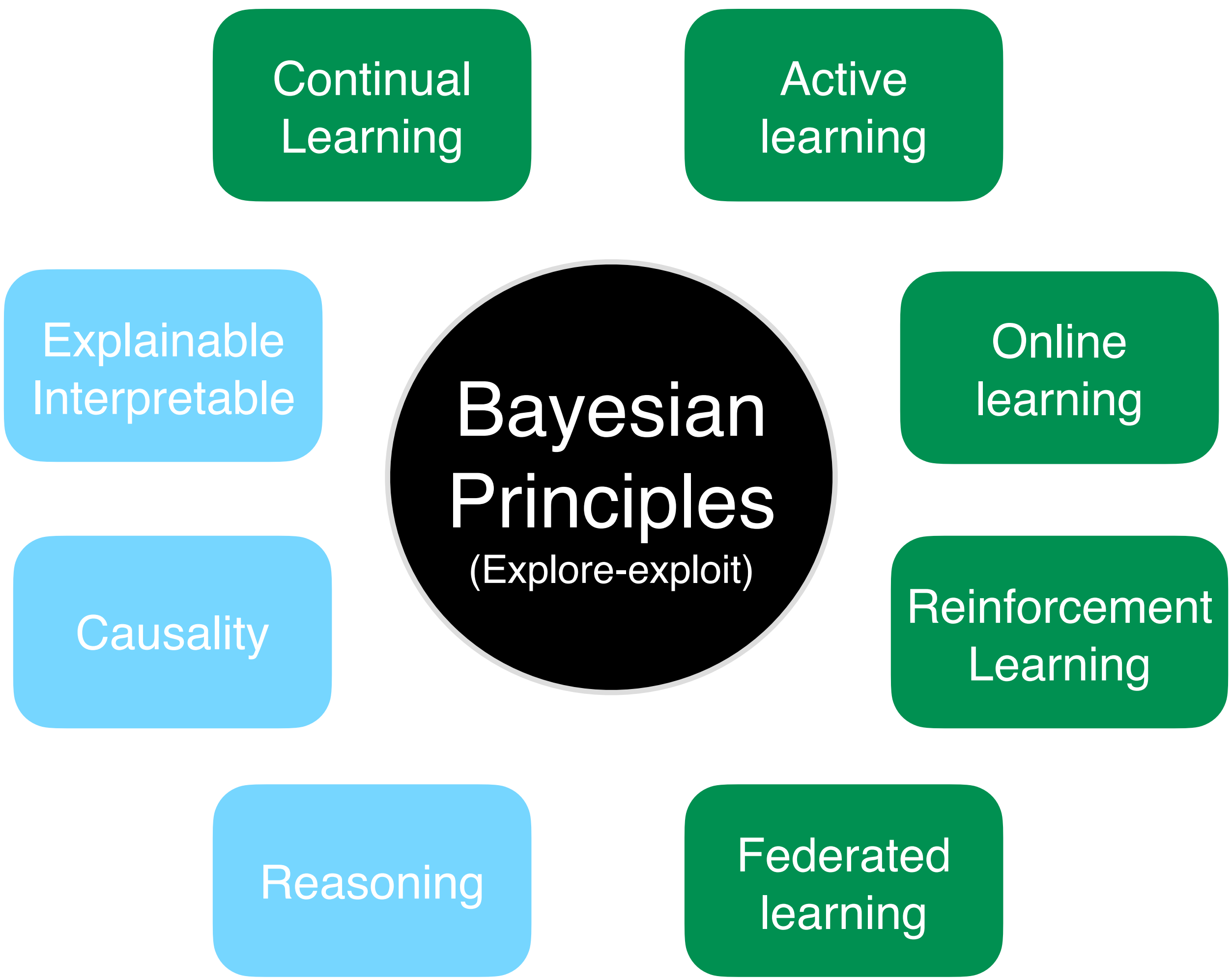
Functional Regularization of Memorable Past (FROMP)

Regularize the function outputs.
Simply adds an additional term in Adam.



1. Pan et al. Continual Deep Learning by Functional Regularisation of Memorable Past, NeurIPS, 2020

Bayes is indispensable for an AI that learns as efficiently as we do



How to design AI that learn like us?

- Uncertainty -> Learning -> Knowledge
- Three questions
  - Q1: What do we know? (model)
  - Q2: What do we not know? (uncertainty)
  - Q3: What do we need to know? (action & exploration)
- Posterior approximation is the key
  - (Q1) Models == representation of the world
  - (Q2) Posterior approximations == representation of the model
  - (Q3) The Bayes-dual representation will enable
    - represent learned knowledge,
    - reuse them in novel situations,
    - interact with the environment to collect new knowledge

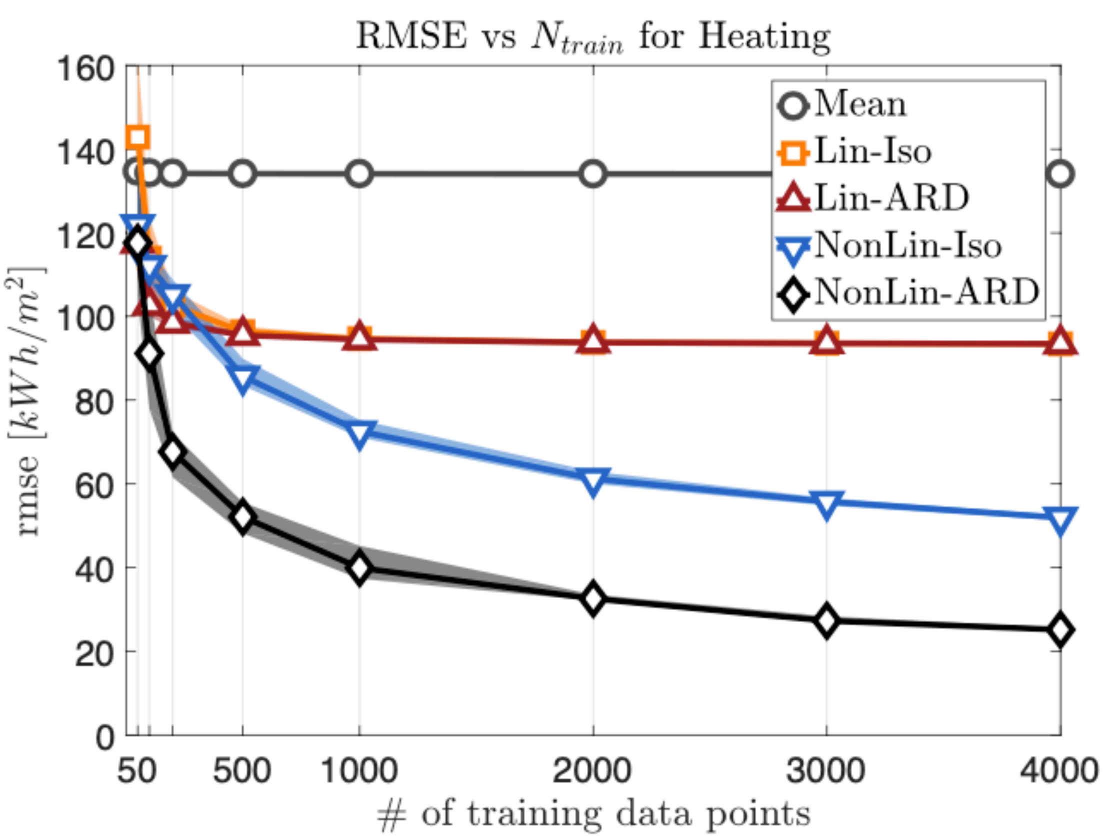
Gaussian-Process-Based Emulators for Building Performance Simulation

Parag Rastogi[1,*], Mohammad Emtiyaz Khan[2], Marilyne Andersen[1]
[1]Interdisciplinary Laboratory of Performance-Integrated Design (LIPID),
Ecole Polytechnique Federale de Lausanne (EPFL), Switzerland.
[2]RIKEN Center for Advanced Intelligence Project, Tokyo, Japan.

Abstract

In this paper, we present an emulator of a building-energy performance simulator. Previous work on emulators for this application has largely focused on linear models. Since the simulator itself is a collection of differential equations, we expect non-linear models to be better emulators than linear models. The emulator we present in this paper is based on Gaussian-process (GP) regression models. We show that the proposed non-linear model is 3-4 times more accurate than linear models in predicting the energy outputs of the simulator. For energy outputs in the range 10-800 kWh/m^2, our model achieves an average error of 10-25 kWh/m^2 compared to an average error of 30-100 kWh/m^2 from using linear models. In addition to being very accurate, our emulator also heavily reduces the computational burden for building designers who rely on simulators. By providing performance feedback for building designs very quickly (in just a few milliseconds), we expect our approach to be particularly useful for exercises that involve a large number of simulations, e.g., Uncertainty Analysis (UA), Sensitivity Analysis (SA), robust design, and optimisa-

Nonlinear models work extremely well



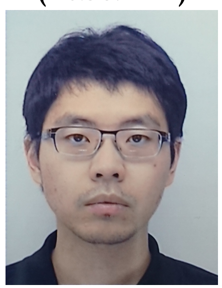
Acknowledgements

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

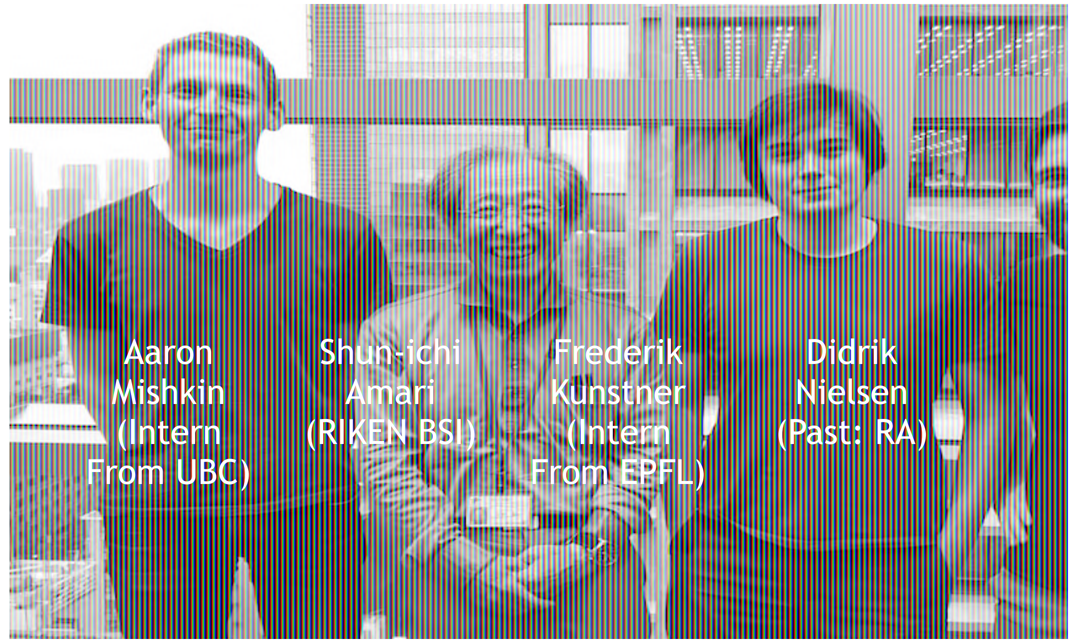
Wu Lin
(Past: RA)

Nicolas Hubacher
(Past: RA)

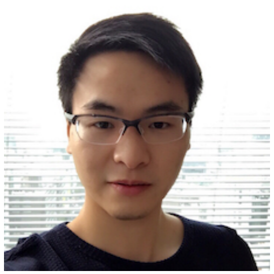
Masashi Sugiyama
(Director RIKEN-AIP)



Voot Tangkaratt
(Postdoc, RIKEN-AIP)



External Collaborators

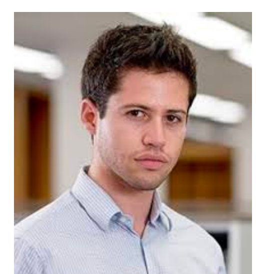


Zuozhu Liu
(Intern from SUTD)

RAIDEN &
Tsubame

Mark Schmidt
(UBC)

Reza Babanezhad
(UBC)



Yarin Gal
(UOxford)

Akash Srivastava
(UEdinburgh)

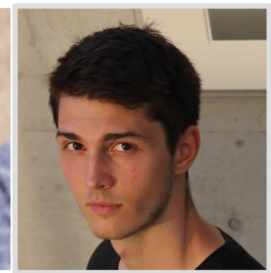
Acknowledgements

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

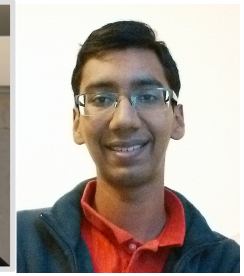
Kazuki Osawa (Tokyo Tech)

Rio Yokota (Tokyo Tech)

Anirudh Jain (Intern from IIT-ISM, India)



Runa Eschenhagen (Intern from University of Osnabruck)

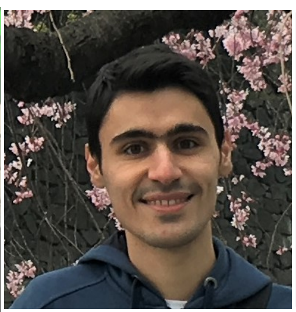


Siddharth Swaroop (University of Cambridge)

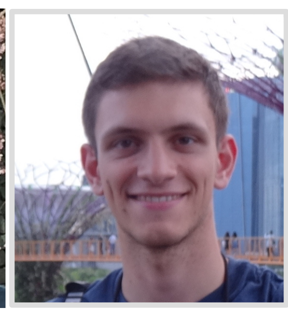
Rich Turner (University of Cambridge)

PingBo Pan (Intern from UT Sydney)

Alexander Immer (Intern from EPFL)



Ehsan Abedi (Intern from EPFL)

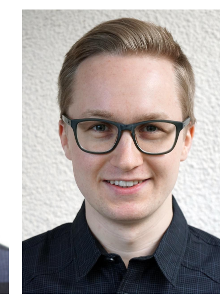


Maciej Korzepa (Intern from DTU)

Pierre Alquier (RIKEN AIP)

Havard Rue (KAUST)

Xiangming Meng Former Post-Doc at RIKEN



Roman Bachmann (Intern from EPFL)

Approximate Bayesian Inference Team


Emtiyaz Khan
Team Leader


Pierre Alquier
Research Scientist


Gian Maria Marconi
Postdoc


Thomas Möllenhoff
Postdoc

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

We have openings for “part-time” student positions and also a postdoc/tech-staff position.


Wu Lin
PhD Student
University of British Columbia


Dharmesh Tailor
Research Assistant


Fariz Ikhwantri
Part-time Student
Tokyo Institute of Technology


Happy Buzaaba
Part-time Student
University of Tsukuba

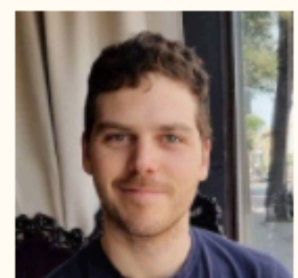

Evgenii Egorov
Remote Collaborator
Skoltech


Siddharth Swaroop
Remote Collaborator
University of Cambridge

Dimitri Meunier
Remote Collaborator
ENSAE Paris


Peter Nickl
Remote Collaborator
TU Darmstadt


Erik Daxberger
Remote Collaborator
University of Cambridge


Alexandre Piché
Remote Collaborator
MILA