Presentation at the KAIST AI Colloquium, Nov 28 2023

How to Build Machines that Adapt Quickly

Mohammad Emtiyaz Khan

RIKEN Center for AI Project, Tokyo

http://emtiyaz.github.io

Summary of recent research at https://emtiyaz.github.io/papers/symposium_2022.pdf

Slides used in this presentation can be found at my website

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 slow or quick to adapt

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

Adaptation in Machine Learning

- Even a small change may need retraining
- Huge amount of resources are required only few can afford (costly & unsustainable) [1,2, 3]
- Difficult to apply in “dynamic” settings (robotics, medicine, epidemiology, climate science, etc.)
- Our goal is to solve such challenges
  - Help in building safe and trustworthy AI
  - But also to reduce “magic” in deep learning

1. Diethe et al. Continual learning in practice, arXiv, 2019.
2. Paleyes et al. Challenges in deploying machine learning: a survey of case studies, arXiv, 2021.
3. https://www.youtube.com/watch?v=hx7BXih7zx8&t=897s

Towards Quick Adaptation

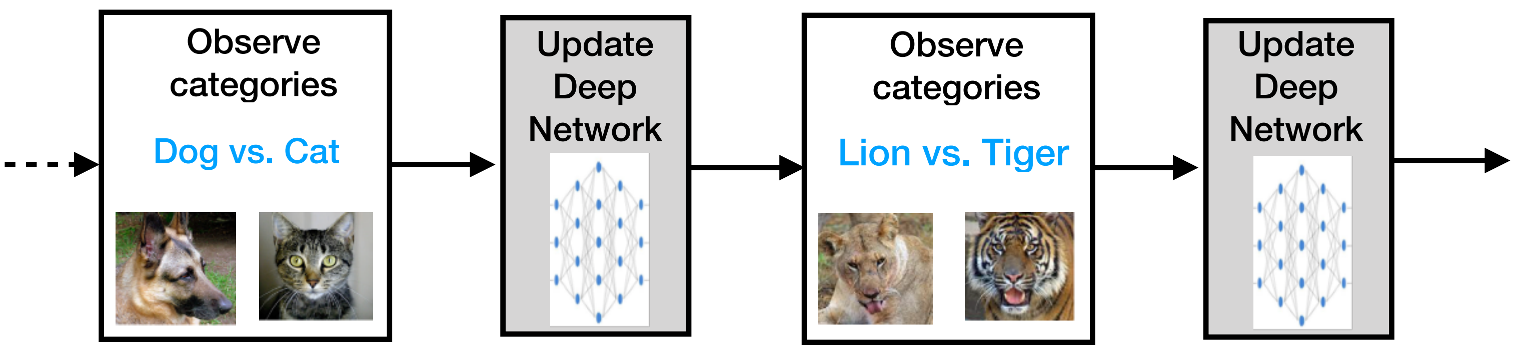
- Better uncertainty [1-4]
  – Bayesian Learning rule (BLR)
- Better regularization [5-8]
  – Knowledge-Adaptation Priors (K-priors)
- Better memory [9]
  – Memory Perturbation Equation (MPE)

1. Khan and Rue, The Bayesian Learning Rule, JMLR (2023).
2. Khan, et al. Fast and scalable Bayesian deep learning by weight-perturbation in Adam, ICML (2018).
3. Osawa et al. Practical Deep Learning with Bayesian Principles, NeurIPS (2019).
4. Lin et al. Handling the positive-definite constraints in the BLR, ICML (2020).
5. Khan and Swaroop. Knowledge-Adaptation Priors, NeurIPS (2021)
6. Pan et al. Continual deep learning by functional regularisation of memorable past, NeurIPS (2020)
7. Daxberger et al. Improving CL by Accurate Gradient Reconstruction of the Past, TMLR (2023).
8. Daheim et al. Model merging by uncertainty-based gradient matching, arXiv 2023.
9. Nickl, Xu, Tailor, Moellenhoff, Khan, The memory-perturbation equation, NeurIPS (2023)

Example: Continual Learning

Standard Deep Learning

Continual Learning: past classes never revisited



Standard training leads to catastrophic forgetting.

Kirkpatrick, James, et al. "Overcoming catastrophic forgetting in neural networks." *Proceedings of the national academy of sciences* 114.13 (2017): 3521-3526.

Bayesian Learning Rule

Better Uncertainty

Weight Regularization

Standard way to is to add a weight-regularizer [1]

$$(\theta - \theta_{\text{old}})^\top F_{\text{old}}(\theta - \theta_{\text{old}})$$

↑ Weight uncertainty

Straightforward improvement in weight-uncertainty is to use variational inference [2-4]

1. Kirkpatrick, James, et al. "Overcoming catastrophic forgetting in neural networks." *PNAS* 2017
2. Khan, et al. "Fast and scalable Bayesian deep learning by weight-perturbation in Adam." *ICML* (2018).
3. Osawa et al. "Practical Deep Learning with Bayesian Principles." NeurIPS (2019).
4. Lin et al. "Handling the positive-definite constraints in the BLR." ICML (2020).

Practical Deep Learning with Bayes

A reliable estimate of Fisher/Hessian/variance

RMSprop

$$g \leftarrow \hat{\nabla}\ell(\theta)$$
$$h \leftarrow g \cdot g$$
$$s \leftarrow (1-\rho)s + \rho h$$
$$\theta \leftarrow \theta - \alpha\, g/\sqrt{s}$$

Bayesian Learning Rule [3]

$$g \leftarrow \hat{\nabla}\ell(\theta)$$
$$h \leftarrow g \cdot \sqrt{s} \cdot \epsilon$$

Perturb g to estimate Hessian

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

Ensure s is always +ve

$$m \leftarrow m - \alpha\; g/s$$
$$\sigma^2 \leftarrow 1/s, \;\; \theta \leftarrow m + \epsilon \sim \mathcal{N}(0, 1/s)$$

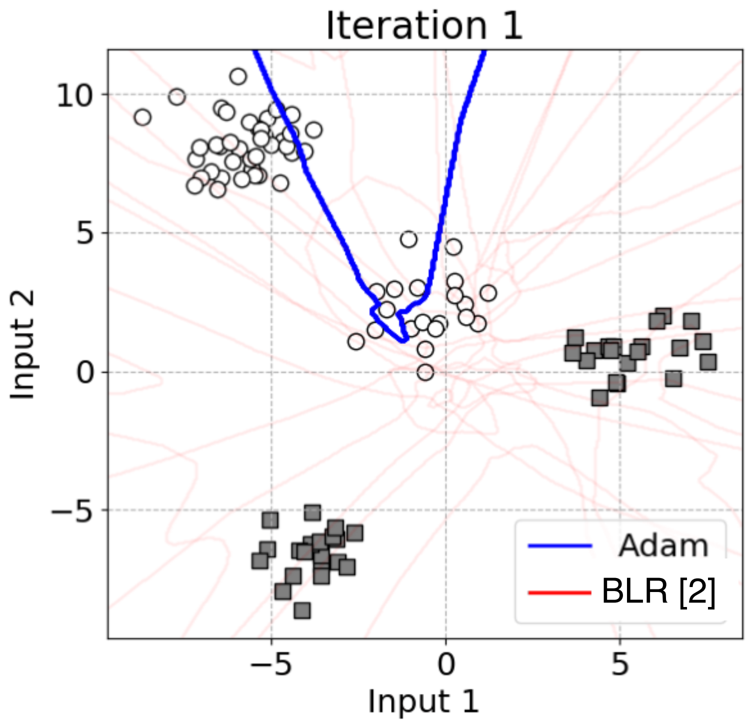
Weight-perturbation to improve variance quality

Costs are exactly the same, but the variance quality is much better!!
2nd-order method that works at scale.

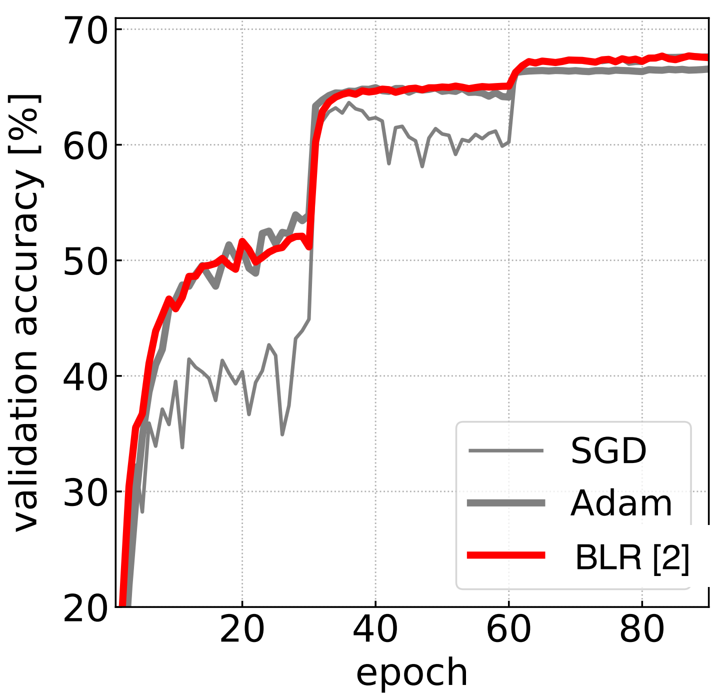
1. Khan, et al. "Fast and scalable Bayesian deep learning by weight-perturbation in Adam." *ICML* (2018).
2. Osawa et al. “Practical Deep Learning with Bayesian Principles.” NeurIPS (2019).
3. Lin et al. “Handling the positive-definite constraints in the BLR.” ICML (2020).

Uncertainty of Deep Nets

Better uncertainty than Adam but similar accuracy



ImageNet Results



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.

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]

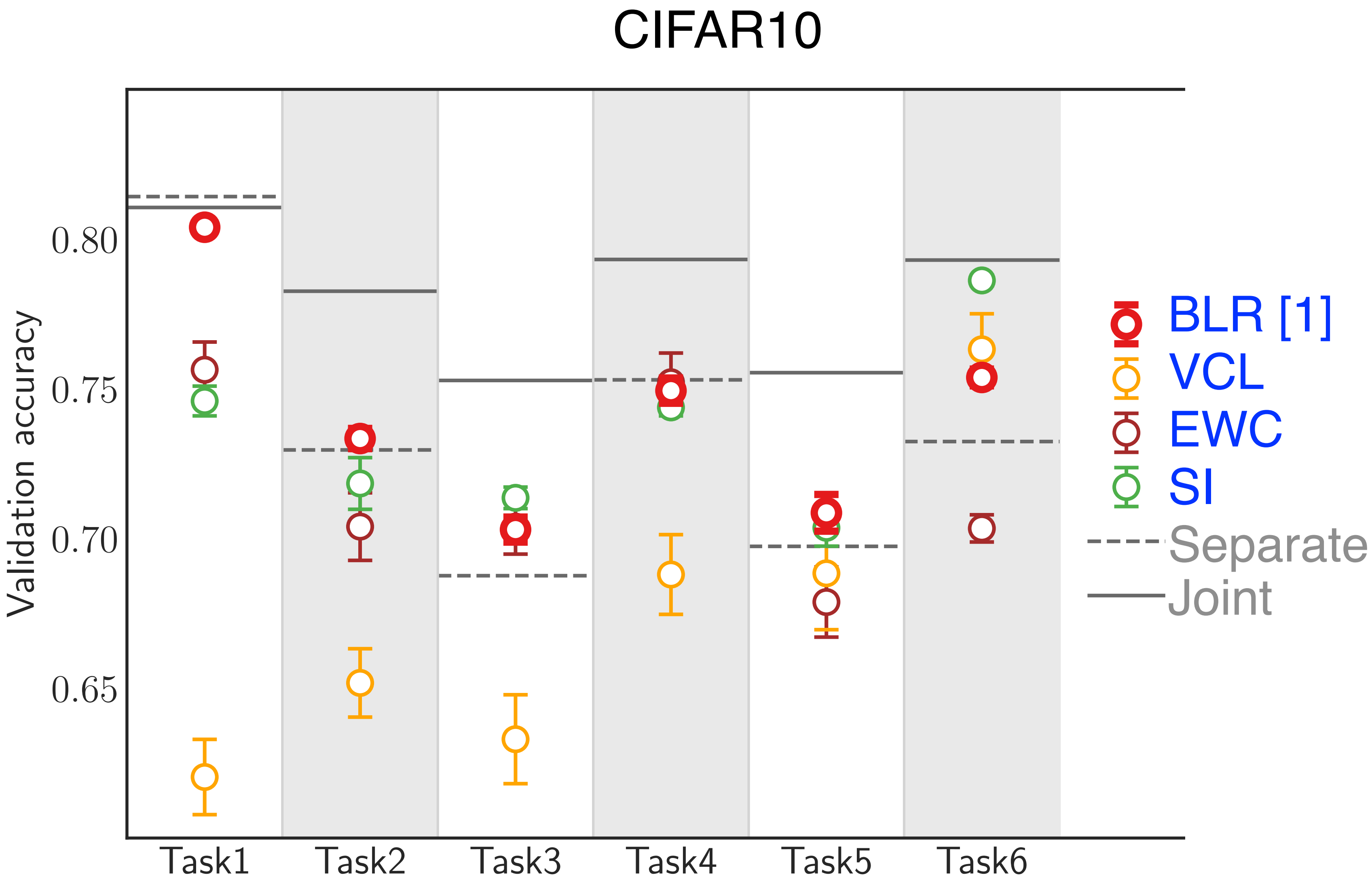
1 Approximate Bayesian Inference Team
RIKEN Center for AI Project, Tokyo, Japan

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).

Continual Learning



1. Osawa et al. “Practical Deep Learning with Bayesian Principles.” NeurIPS (2019).

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

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 Table 1 in Khan and Rue, 2021

All sorts of algorithms can be derived by using two sets of approximations.

By relaxing the approximations, we get an improvement, for example, uncertainty aware deep learning optimizers

1. Khan and Rue, The Bayesian Learning Rule, JMLR (2023)
2. Khan and Lin. "Conjugate-computation variational inference…." AIstats (2017).

Bayesian-SAM

An Adam-style algorithm, derived using the BLR, where variances are automatically learned.

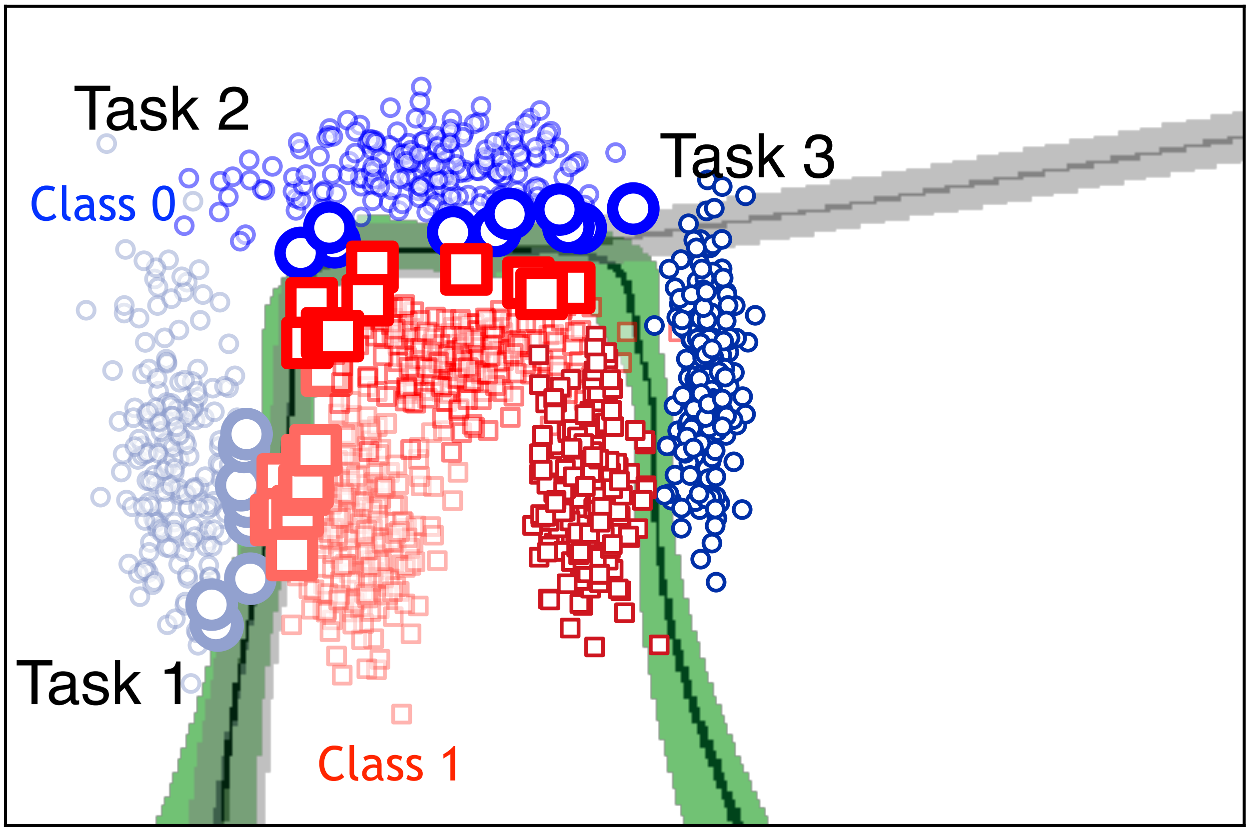
SAM with RMSprop	SAM with BLR
$g_1 \leftarrow \hat{\nabla}\ell(\theta)$	$g_1 \leftarrow \hat{\nabla}\ell(\theta)$
$\epsilon \leftarrow \rho \frac{g_1}{\|g_1\|}$	$\epsilon \leftarrow \frac{\rho'}{s} g_1$
$g \leftarrow \hat{\nabla}\ell(\theta + \epsilon)$	$g \leftarrow \hat{\nabla}\ell(\theta + \epsilon)$
$s \leftarrow (1-\rho)s + \rho g^2$	$s \leftarrow (1-\rho)s + \rho\sqrt{s}\|g_1\|$
$\theta \leftarrow \theta - \alpha(\sqrt{s} + \delta)^{-1} g$	$\theta \leftarrow \theta - \alpha(s + \gamma)^{-1} g$
	$\sigma^2 \leftarrow (s+\gamma)^{-1}, \;\; \theta \leftarrow m + \epsilon'\sigma$

1. Foret et al. Sharpness-Aware Minimization for Efficiently Improving Generalization, ICLR, 2021
2. Moellenhoff and Khan, SAM as an optimal relaxation of Bayes, https://arxiv.org/abs/2210.01620, 2022

Knowledge-Adaptation Prior

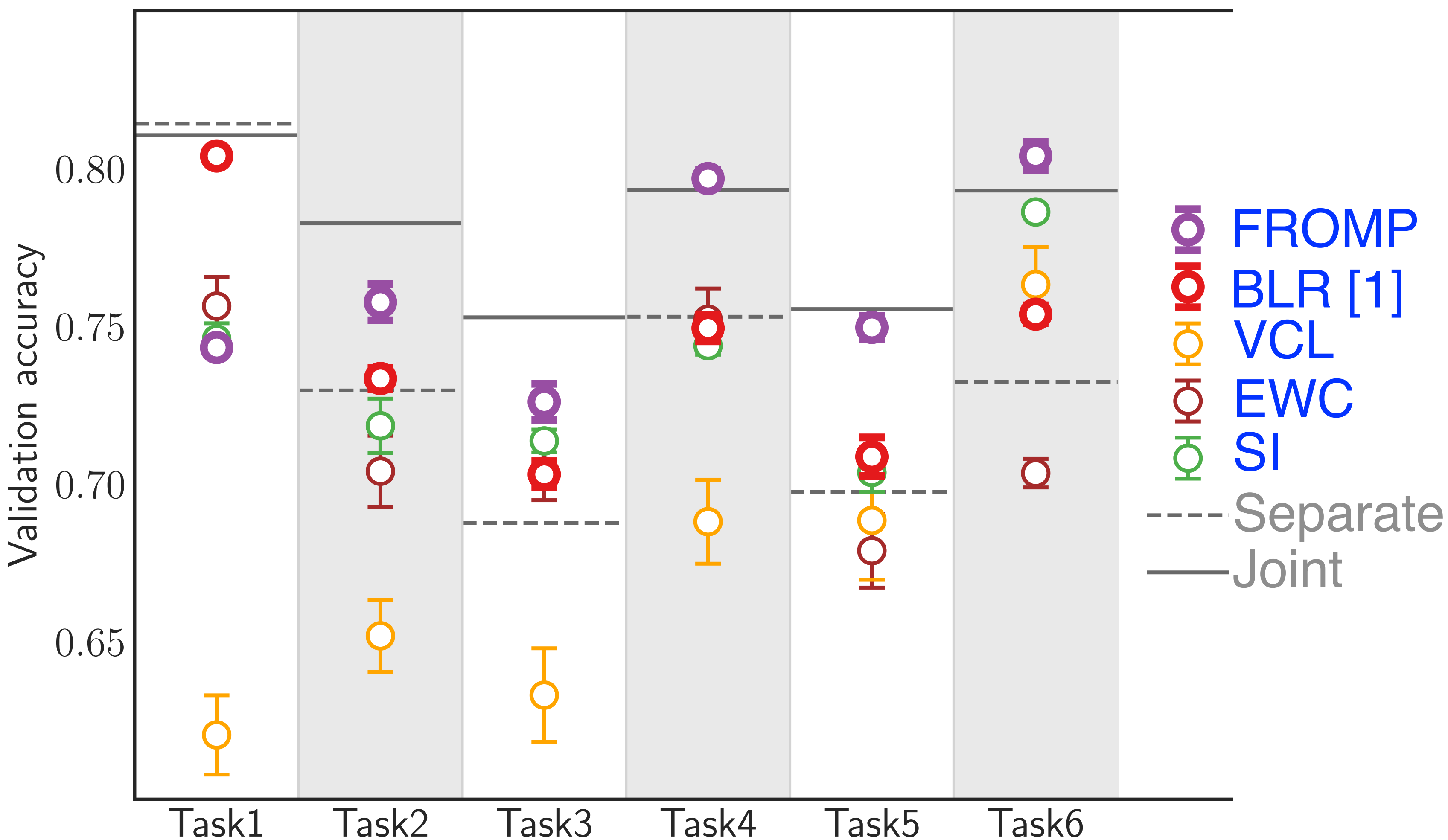
Better Regularization

Functional Regularization of Memorable Examples [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

Improvements over EWC and VOGN



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

Functional Regularization of Memorable Past (FROMP)

Weight-regularizer (EWC) [1]

$$(\theta - \theta_{\text{old}})^\top F_{\text{old}}(\theta - \theta_{\text{old}})$$

↑ Weight uncertainty

Functional regularizer (FROMP) [2]

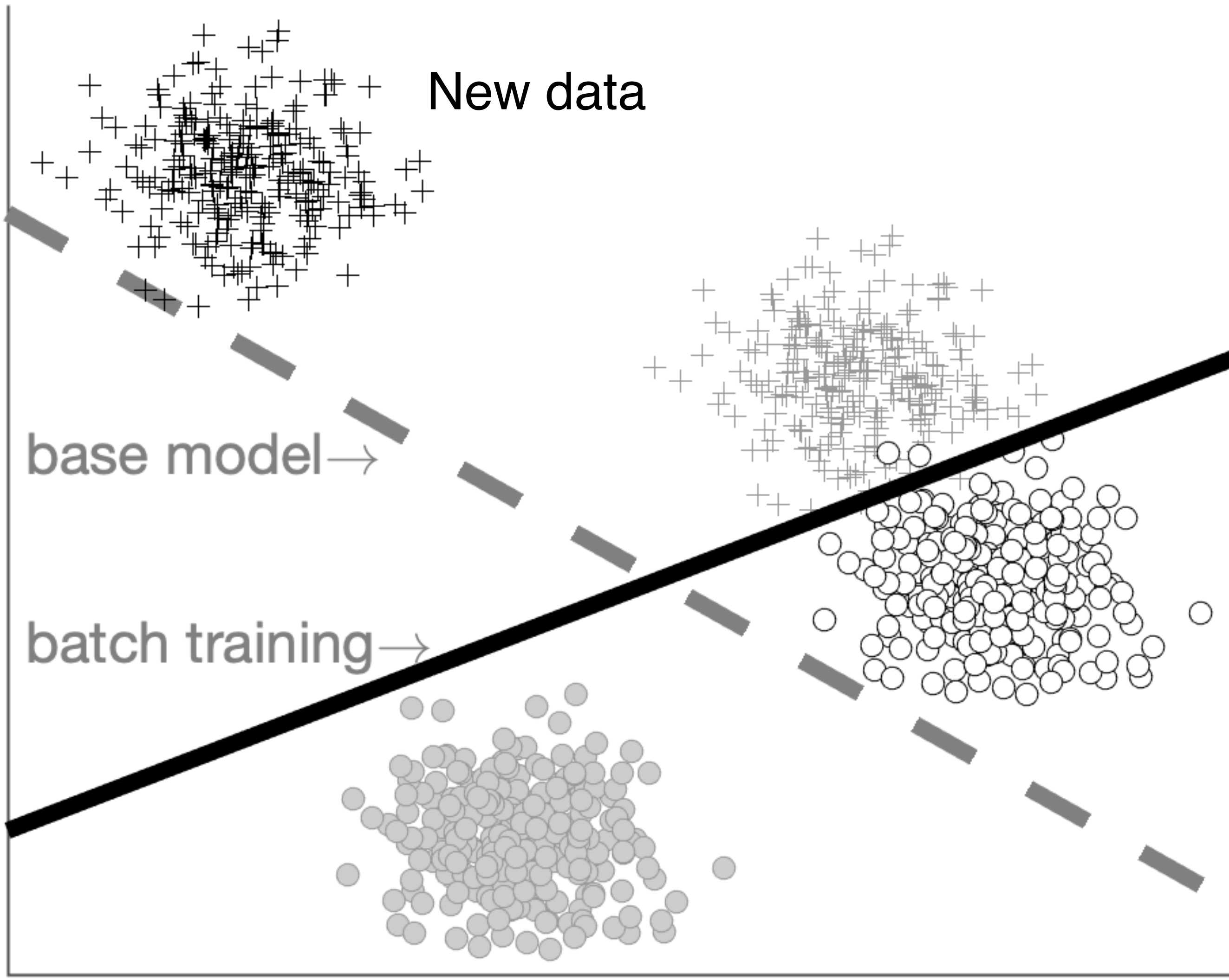
$$[\sigma(\mathbf{f}(\theta)) - \sigma(\mathbf{f}_{old})]^\top K_{old}^{-1}[\sigma(\mathbf{f}(\theta)) - \sigma(\mathbf{f}_{old})]$$

↑ Uncertainty ↑ Predictions

Why does this work? It is a way to replay past gradients, which leads to the idea of K-priors.

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

Intuition behind K-priors

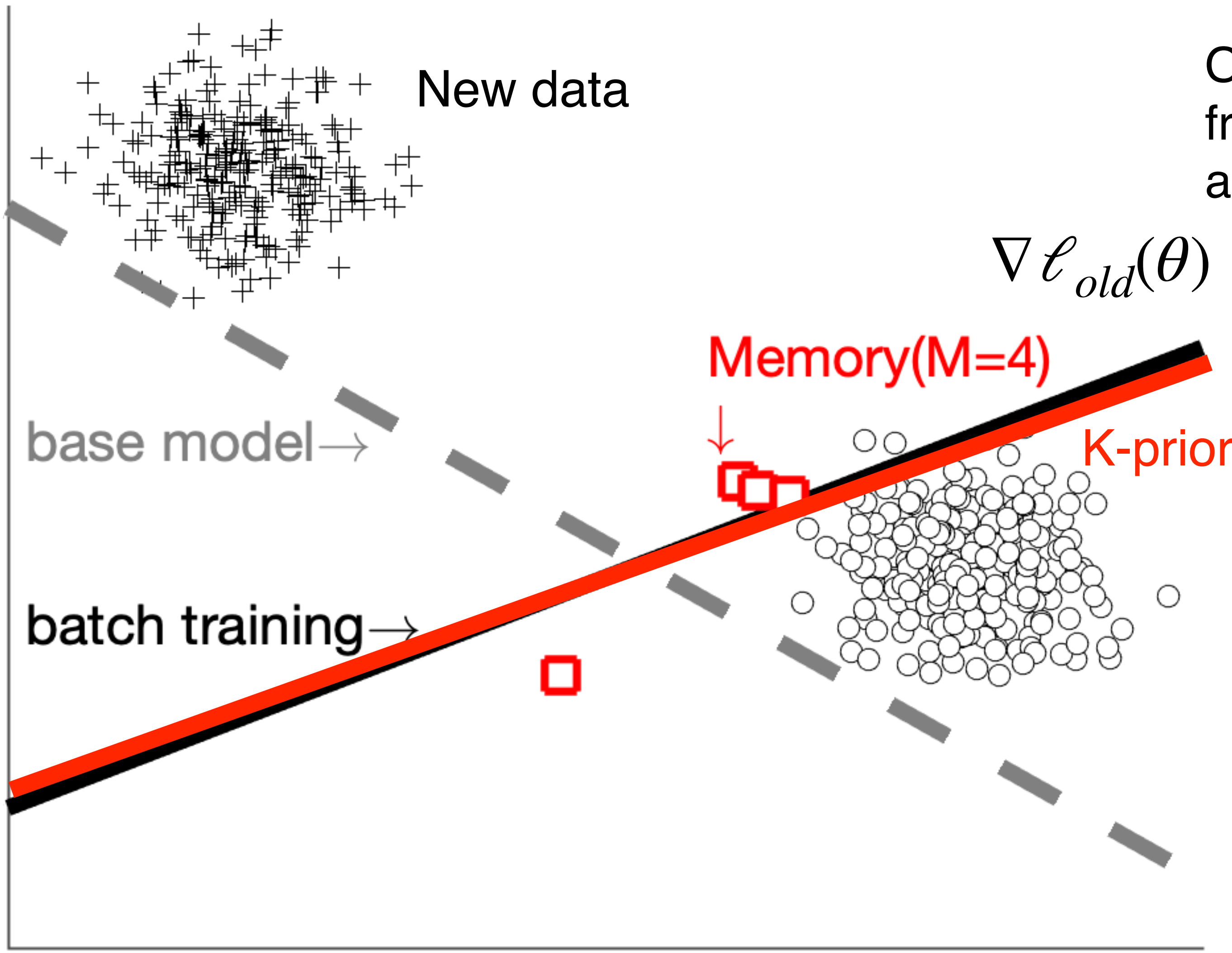


Often, only a small fraction of old data is affected.

Binary classification with Logistic regression

Each task N=500, each class 250 examples.

Intuition behind K-priors



Often, only a small fraction of old data is affected.

$$\nabla \ell_{old}(\theta) \approx \nabla Kprior(\theta)$$

Binary classification with Logistic regression

Each task N=500, each class 250 examples.

Easy to see in Linear Regression

Weight-space Function-space

$$\arg\min_{\theta} \ell_{old} = \|\theta\|^2 + \|y - X\theta\|^2 \qquad F_{old} = I + X^\top X$$

$$(\theta - \theta_{old})^\top F_{old}(\theta - \theta_{old}) = (\theta - \theta_{old})^\top (I + X^\top X)(\theta - \theta_{old})$$

Entirely in weight-space (EWC) [1]

$$= \|\theta - \theta_{old}\|^2 + \|X\theta - X\theta_{old}\|^2$$

Weight-space Function-space

Knowledge-adaptation prior [3]

$$= (X\theta - X\theta_{old})^\top K^{-1}(X\theta - X\theta_{old})$$

Entirely in function-space (FROMP) [2]

In linear regression, they are equivalent and are all ways to reconstruct the old problem (or its gradients)

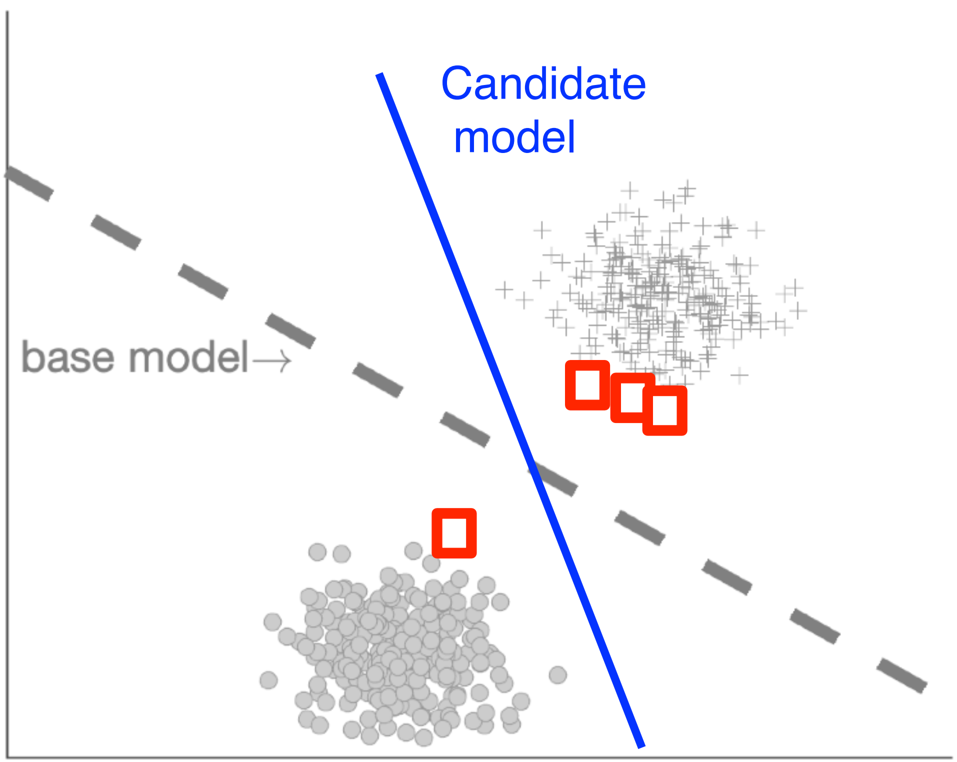
1. Kirkpatrick, James, et al. "Overcoming catastrophic forgetting in neural networks." *PNAS* 2017
2. Pan et al. Continual Deep Learning by Functional Regularisation of Memorable Past, NeurIPS, 2020
3. Khan and Swaroop. Knowledge-Adaptation Priors, NeurIPS, 2021

Knowledge-Adaptation Priors

Combine weight and function-space divergences

Weight-space   Function-space

$$\mathcal{K}(\theta) = \tau \mathbb{D}_w(\theta \| \theta_{\text{old}}) + \mathbb{D}_f(\mathbf{f}(\theta) \| \mathbf{f}(\theta_{\text{old}}))$$

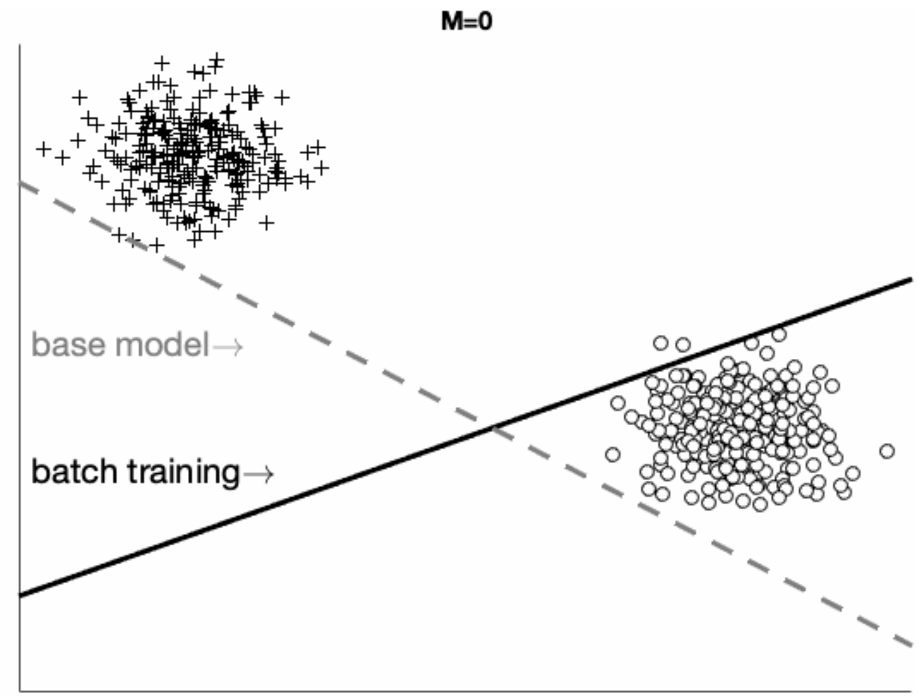


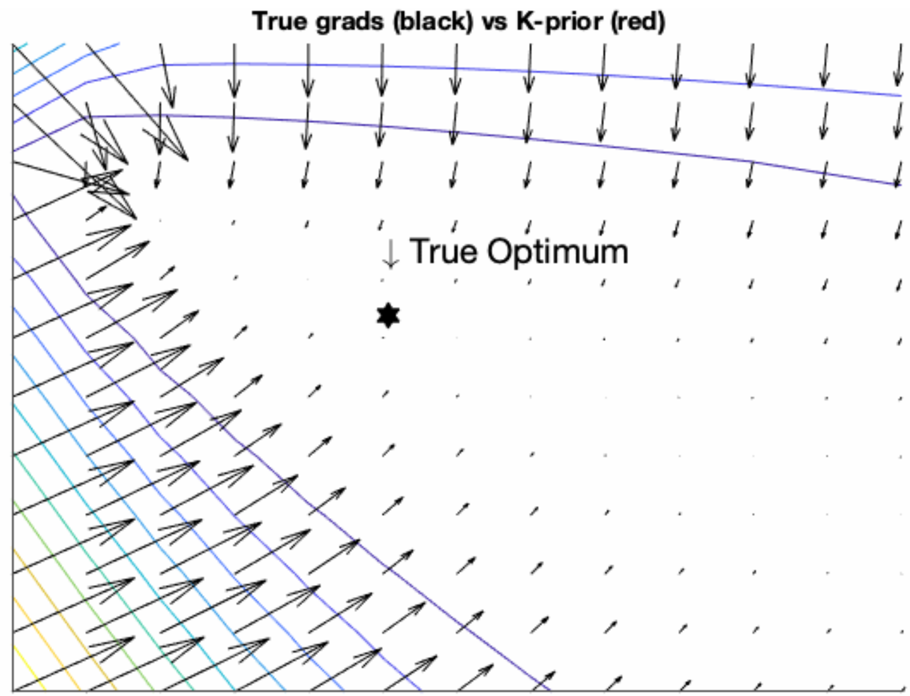
$$\begin{bmatrix} f_w^1 \\ f_w^2 \\ f_w^3 \\ f_w^4 \end{bmatrix} \quad \begin{bmatrix} f_{w_*}^1 \\ f_{w_*}^2 \\ f_{w_*}^3 \\ f_{w_*}^4 \end{bmatrix}$$

K-prior is a way to replay past gradients

A General Principle of Adaptation

Reconstruct past gradients

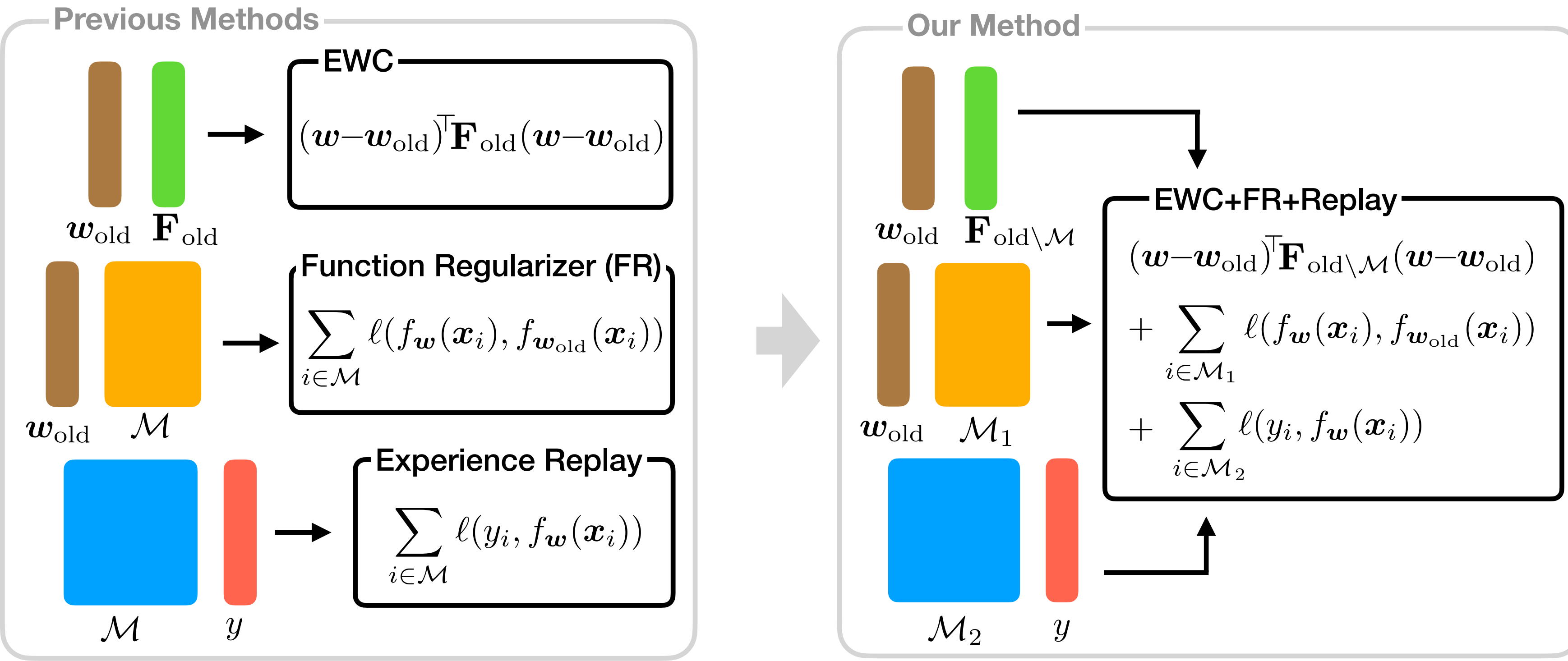




1. Khan and Swaroop. Knowledge-Adaptation Priors, NeurIPS, 2021 (https://arxiv.org/abs/2106.08769)

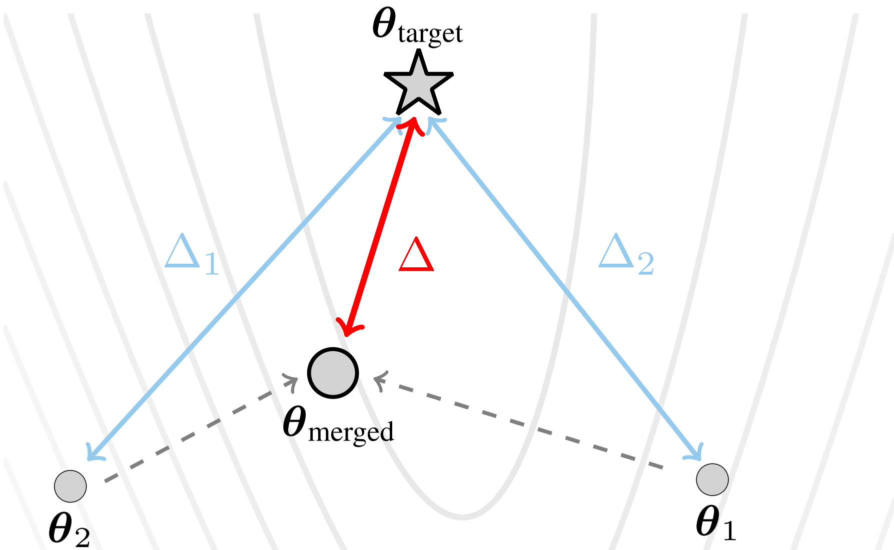
How to combine EWC + FR + Replay

Combine approaches to (successively) reduce grad-reconstruction error



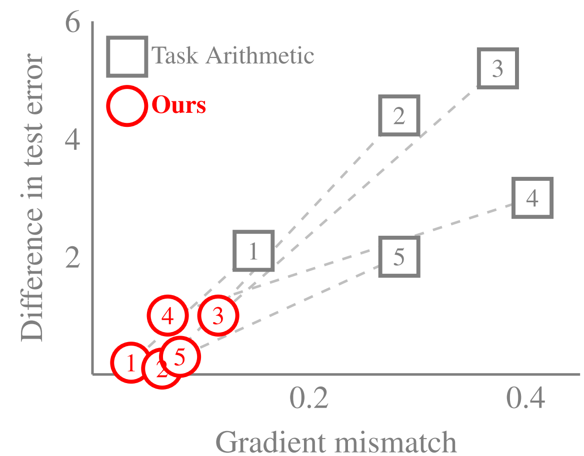
1. Daxberger et al. Improving CL by Accurate Gradient Reconstruction of the Past, TMLR 2023.

Model Merging for LLMs



$$\Delta = \sum_{t=1}^{2} \underbrace{\nabla \bar{\ell}_t(\boldsymbol{\theta}_{\text{target}}) - \nabla \bar{\ell}_t(\boldsymbol{\theta}_t)}_{\text{Gradient Mismatch}} \approx \sum_{t=1}^{2} \mathbf{H}_t \Delta_t$$

RoBERTa on IMDB



Toxicity removal from GPT (1.3B)

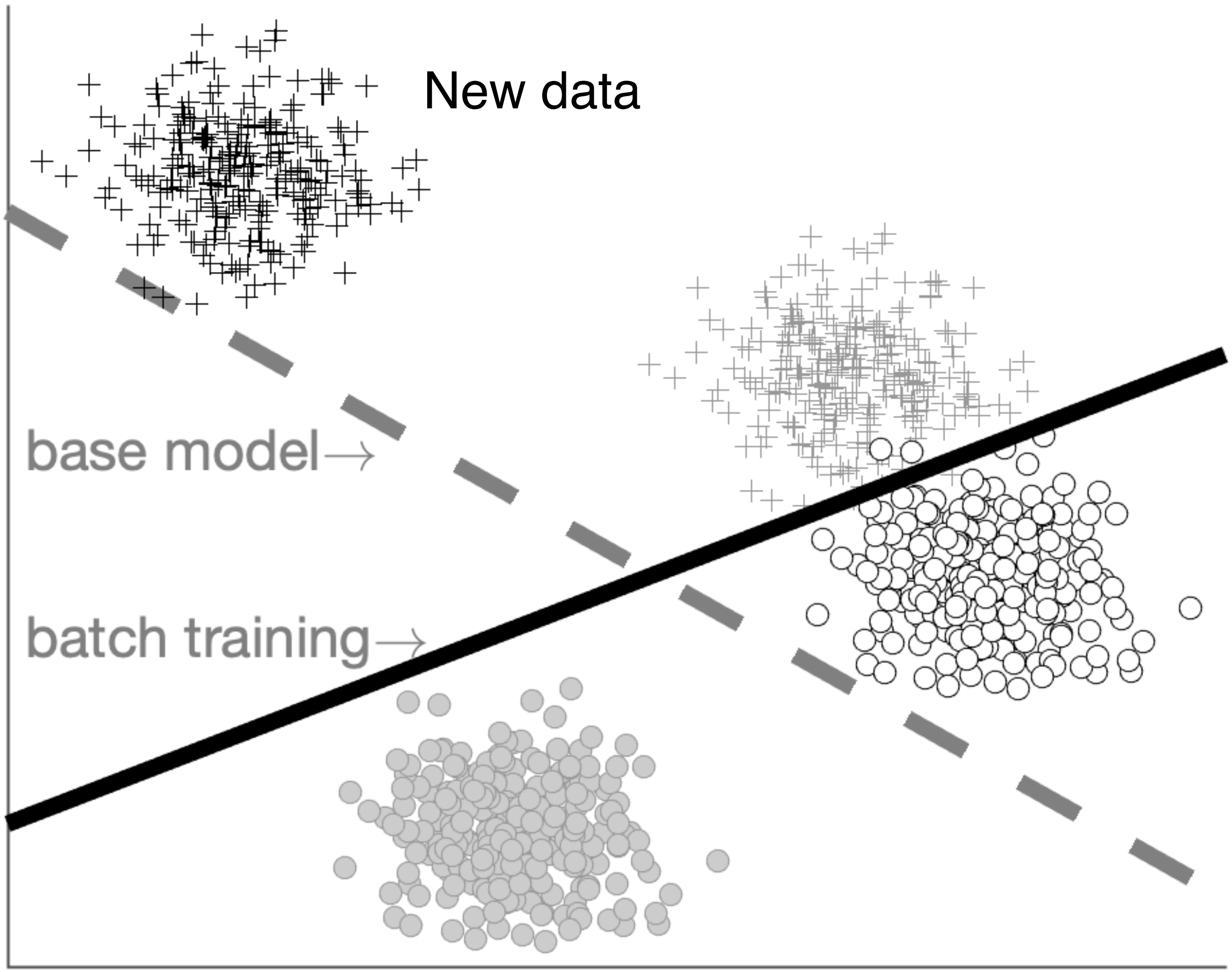
Model	$\boldsymbol{\theta}$	Toxicity		Fluency
		100·Avg.	Num. Toxic	PPL(↓)
GPT2_{117M}	$\boldsymbol{\theta}_{\text{LLM}}$	11.2	15.4 %	24.9
	TA	9.8	13.1 %	30.3
	ours	**9.6** (↓0.2)	**12.8 %** (↓0.3)	**26.9** (↓3.4)
$\text{GPT-J}_{1.3B}$	$\boldsymbol{\theta}_{\text{LLM}}$	11.9	16.6 %	12.6
	TA	10.7	14.5 %	**12.7**
	ours	**10.2** (↓0.5)	**14.0 %** (↓0.5)	12.8 (↓0.1)

1. Daheim et al. Model merging by uncertainty-based gradient matching, arXiv 2023.

Memory-Perturbation Equation

Better Memory

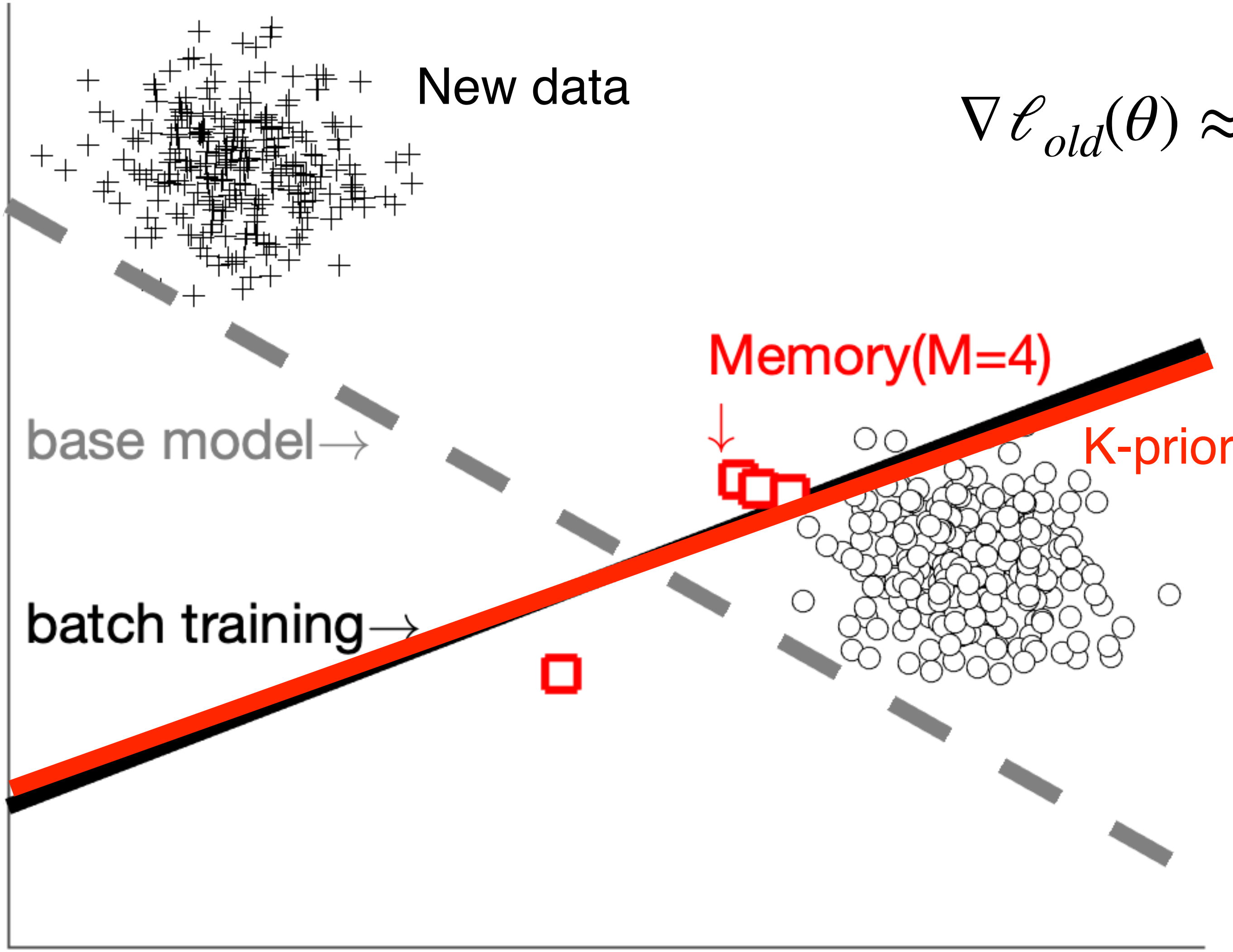
Intuition behind K-priors



Binary classification with Logistic regression

Each task N=500, each class 250 examples.

Intuition behind K-priors



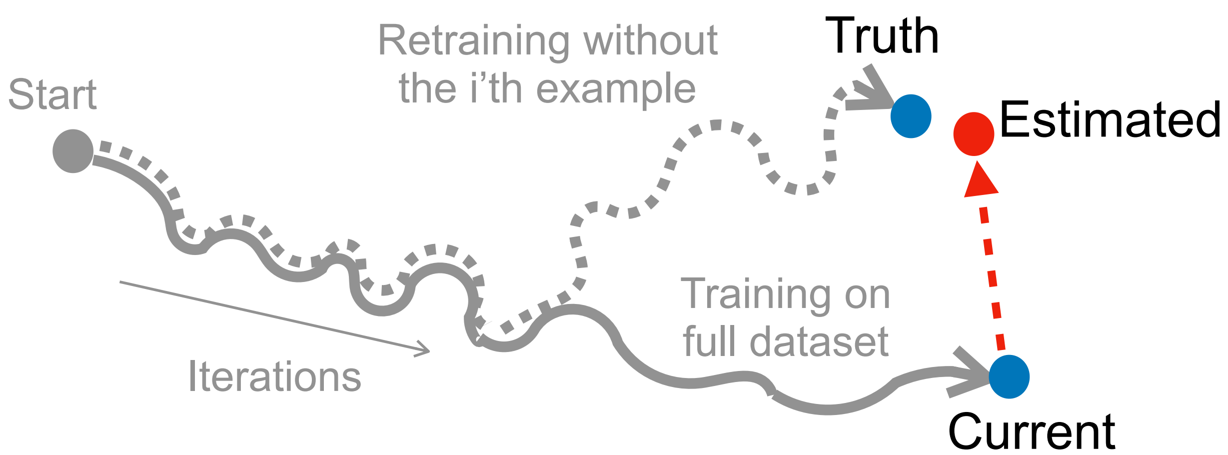
$$\nabla \ell_{old}(\theta) \approx \nabla Kprior(\theta)$$

Binary classification with Logistic regression

Each task N=500, each class 250 examples.

Memory and Sensitivity

Past information with most influence on the present

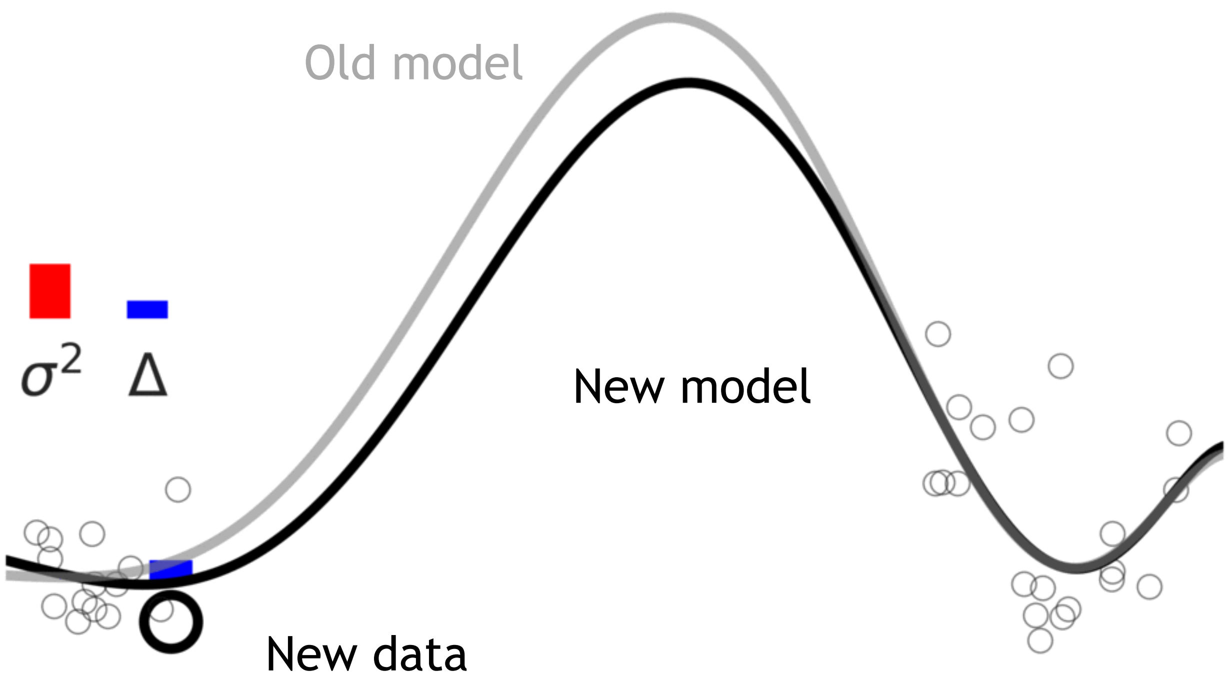


Computing the algorithm-deviation by retraining is expensive. We want to estimate it without retraining!

Memory Perturbation

How sensitive is a model to its training data?

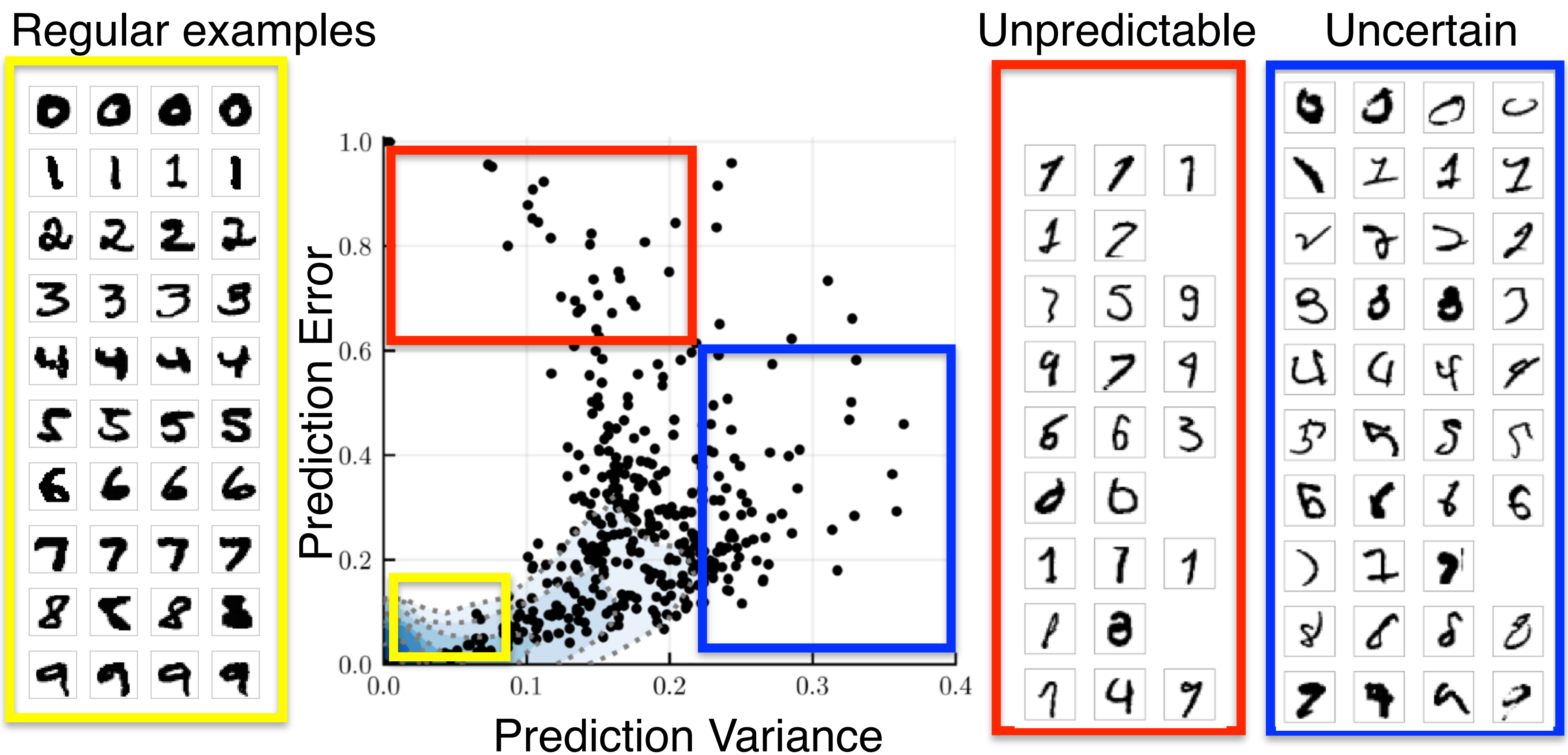
Deviation (Δ) = predictionError *predictionVariance



1. Cook. Detection of Influential Observations in Linear Regression. Technometrics. ASA 1977
2. Nickl, Xu, Tailor, Moellenhoff, Khan, The memory-perturbation equation, NeurIPS, 2023

Memory Maps using the BLR

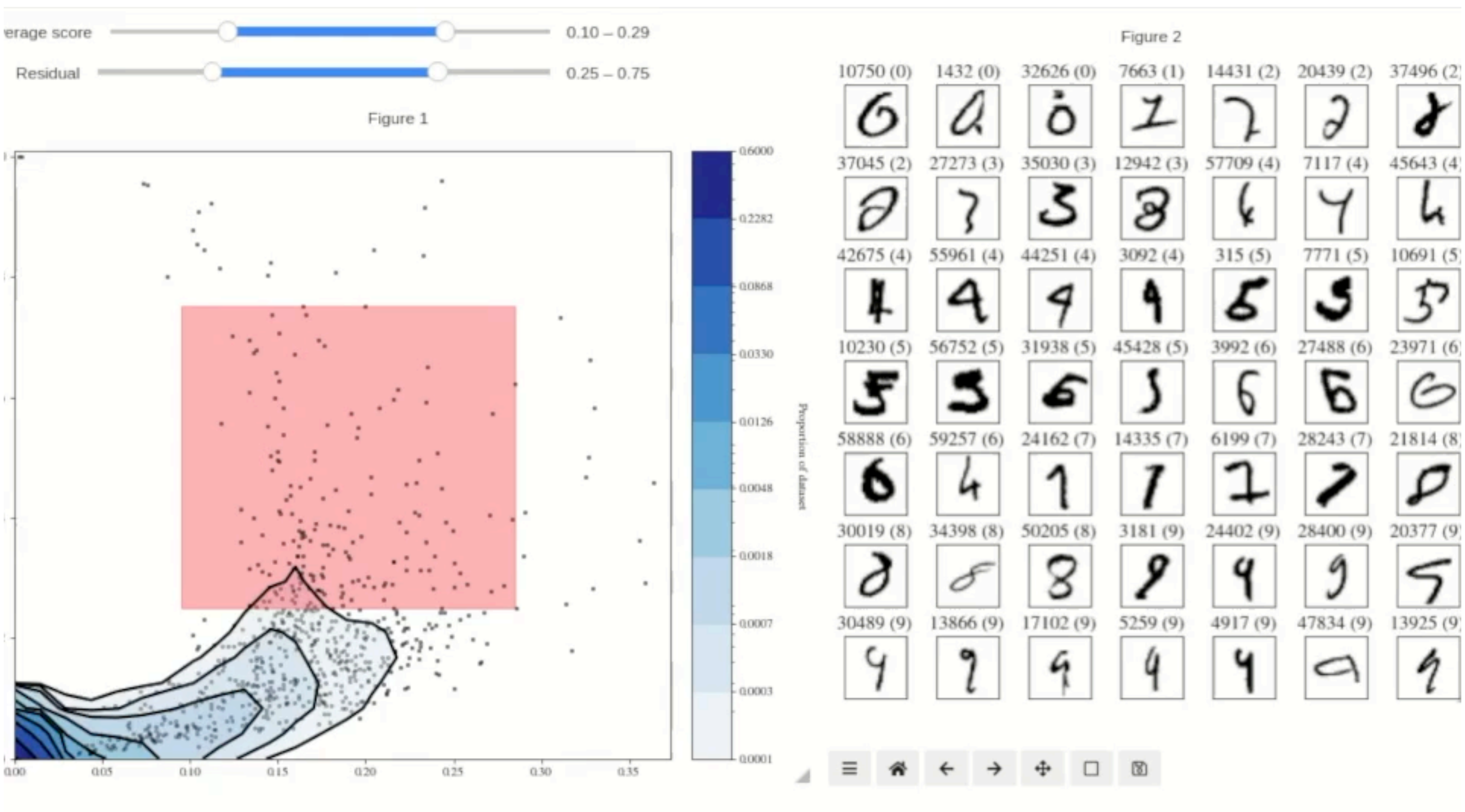
Understand generic ML models and algorithms.



1. Tailor, Chang, Swaroop, Nalisnick, Solin, Khan, Memory maps to understand models (under review)

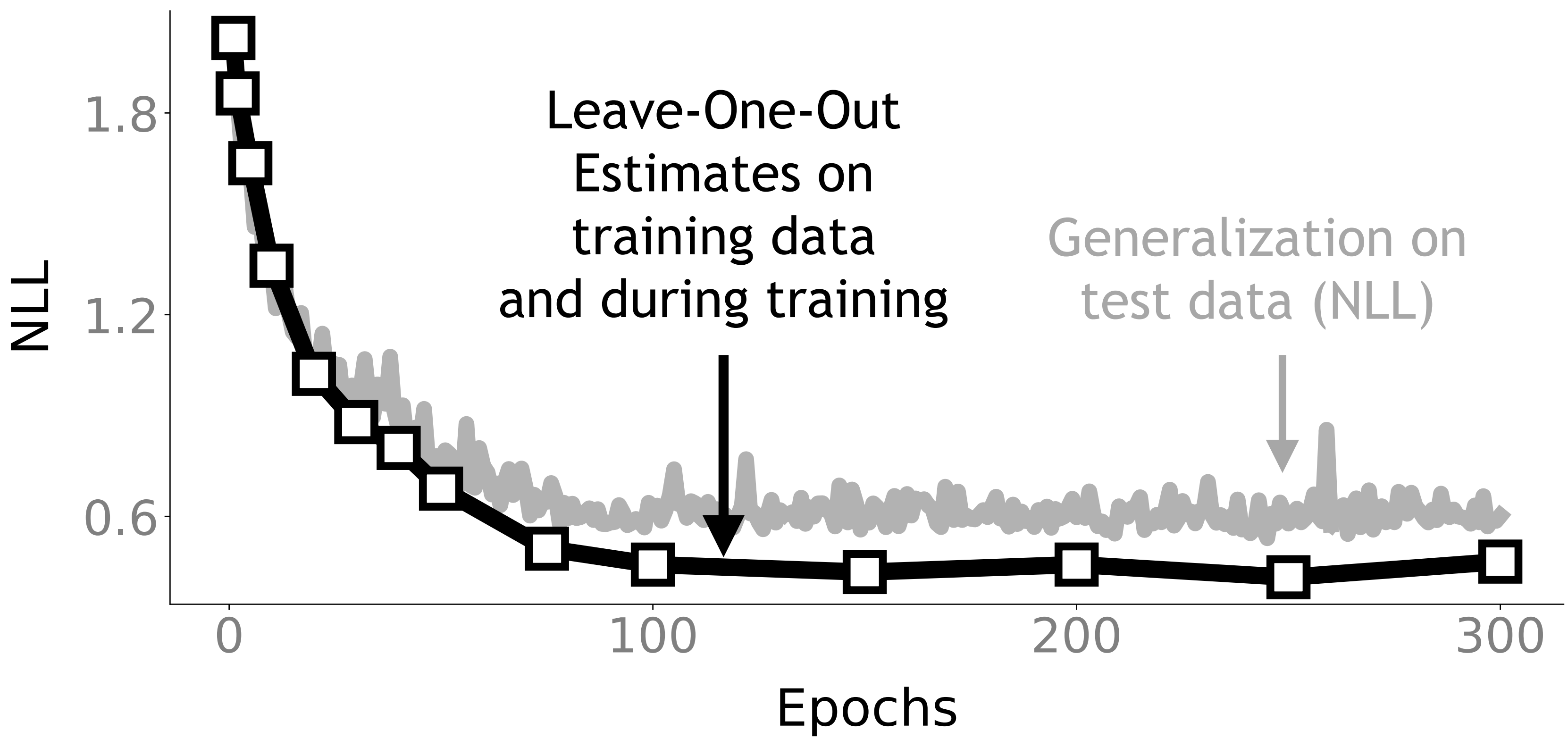
A Tool for Data-Scientists

Understand the memory of a model.



Predict Generalization during Training

CIFAR10 on ResNet-20 using BLR [1]. SGD or Adam also works but better uncertainty gives better estimates.



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

Towards Quick Adaptation

- Better uncertainty [1-4]
  - Bayesian Learning rule (BLR)
- Better regularization [5-8]
  - Knowledge-Adaptation Priors (K-priors)
- Better memory [9]
  - Memory Perturbation Equation (MPE)

1. Khan and Rue, The Bayesian Learning Rule, JMLR (2023).
2. Khan, et al. Fast and scalable Bayesian deep learning by weight-perturbation in Adam, ICML (2018).
3. Osawa et al. Practical Deep Learning with Bayesian Principles, NeurIPS (2019).
4. Lin et al. Handling the positive-definite constraints in the BLR, ICML (2020).
5. Khan and Swaroop. Knowledge-Adaptation Priors, NeurIPS (2021)
6. Pan et al. Continual deep learning by functional regularisation of memorable past, NeurIPS (2020)
7. Daxberger et al. Improving CL by Accurate Gradient Reconstruction of the Past, TMLR (2023).
8. Daheim et al. Model merging by uncertainty-based gradient matching, arXiv 2023.
9. Nickl, Xu, Tailor, Moellenhoff, Khan, The memory-perturbation equation, NeurIPS (2023)

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.

Approximate Bayesian Inference Team

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


Emtiyaz Khan
Team Leader


Thomas Möllenhoff
Research Scientist


Geoffrey Wolfer
Special Postdoctoral Resesarcher


Hugo Monzón Maldonado
Postdoctoral Researcher


Keigo Nishida
Postdoctoral Researcher
RIKEN BDR


Gian Maria Marconi
Postdoctoral Researcher


Lu Xu
Postdoctoral Researcher


Peter Nickl
Research Assistant


Etash Guha
Intern
Georgia Tech


Joseph Austerweil
Visiting Scientist
University of Winsconsin-Madison

Pierre Alquier
Visiting Scientist
ESSEC Business School


Dharmesh Tailor
Remote Collaborator
University of Amsterdam

Many thanks to our group members and collaborators (many not on this slide).

We have open positions and are always looking for new collaborations.