Keynote at Bayes Comp 2025, Jun 20, 2025

Adaptive Bayesian Intelligence
(AGI meets ABI)

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

Summary of recent research at https://emtiyaz.github.io/papers/symposium_2024.pdf
Slides available at https://emtiyaz.github.io/

AI that can learn like us

Quickly adapt & continue to acquire new skills

Human Learning at the age of 6 months.

Converged at the age of 12 months

Transfer skills

at the age of 14 months

Teacher-Student Learning?

Current state of Machine Learning

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

Retraining from Scratch

Even when changes are tiny.

It is costly, undemocratic and unsustainable.

Adaptive Intelligence

How do brains adapt quickly?
What do they optimize and how?

1. Sternberg. A theory of adaptive intelligence and its relation to general intelligence.*Journal of Intelligence (2019)*
2. Sternberg. *Adaptive intelligence.* New York: Cambridge University Press (2021)
3. Sternberg. What is intelligence really? the futile search for a holy grail. Learning & Individual Differences (2024)

Adaptive Bayesian Intelligence

- Adaptive Intelligence = Bayesian Computation
- Part 1: Bayesian Learning Rule [1]
  - Foundational way to derive learning-algorithms
  - Application to Deep Learning [2]
- Part 2: Posterior Correction [3]
  - Foundational way to derive adaptation-algorithms
  - Application to continual learning [4-5]
  - But also for LLM merging, Federated Learning etc.
- Adaptive Bayesian Intelligence: A roadmap.

1. Khan and Rue, The Bayesian Learning Rule, JMLR (2023)
2. Shen et al. Variational Learning is Effective for Large Deep Networks, ICML (2024)
3. Khan. Knowledge Adaptation as Posterior Correction, arXiv (2025)
4. Khan and Swaroop. Knowledge-Adaptation Priors, NeurIPS (2021).
5. Pan et al. Continual Deep Learning by Functional Regularisation of Memorable Past, NeurIPS, 2020

“The fact that many different approaches point to the same actual algorithm is a major strength of Bayesianity”

—E. T. Jaynes, discussion of [1]

1. Zellner, Optimal Information Processing and Bayes’ Theorem. The American Statistician (1988)

Optimization

Gradient Descent
Newton's Method
Multimodal Optimization

Deep-Learning

SGD, RMSprop and Adam
Sharpness-Aware Minimization
Dropout, STE, Label Smoothing
Shampoo….

Bayesian Learning Rule [1]

Approximate Inference

Conjugate Bayes
Laplace's Method
Expectation Maximization
Stochastic Variational Inference
Variational Message Passing

Global-Optimization

Exponential-Weight Aggregation
Natural Evolution Strategy
Gaussian Homotopy
Smoothed Optimization
Weight-perturbed Optimization
Stochastic Search (annealing)
Stochastic Relaxation

1. Khan and Rue, The Bayesian Learning Rule, JMLR (2023).

Variational Formulation of Bayes' Rule

Bayes' Rule: $p_t(\theta) \propto p_0(\theta) \prod_{j=1}^{t} \mathsf{lik}_j(\theta)$

Variational Inference to find an approximation $q_t(\theta)$

$$q_t = \arg\min_{q \in \mathcal{Q}} \sum_{j=1}^{t} \mathbb{E}_q[\underbrace{-\log \mathsf{lik}_j}_{=\ell_j}] + KL(q \| \underbrace{p_0}_{\propto e^{-\ell_0}})$$

$$= \arg\min_{q \in \mathcal{Q}} \sum_{j=0}^{t} \mathbb{E}_q[\ell_j] - \mathscr{H}(q)$$

We will use this variational formulation to discover the inherent Bayesian nature of (non-Bayesian) algorithms.

Exponential Family

Natural parameters — Sufficient Statistics — Expectation parameters

$$q(\theta) \propto \exp\left[\lambda^\top T(\theta)\right] \qquad \mu := \mathbb{E}_q[T(\theta)]$$

$$\mathcal{N}(\theta|m, S^{-1}) \propto \exp\left[-\frac{1}{2}(\theta - m)^\top S(\theta - m)\right]$$

$$\propto \exp\left[(Sm)^\top \theta + \mathrm{Tr}\left(-\frac{S}{2}\theta\theta^\top\right)\right]$$

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. Wainwright and Jordan, Graphical Models, Exp Fams, and Variational Inference Graphical models 2008
2. Malago et al., Towards the Geometry of Estimation of Distribution Algos based on Exp-Fam, FOGA, 2011

Bayesian Learning Rule (BLR) [1]

Deep Learning to find θ

$$\min_{\theta} \bar{\ell}(\theta) = \sum_{j=0}^{t} \ell_j(\theta)$$

SGD or Adam

$$\theta \leftarrow \theta - \rho \; P^{-1} \nabla \bar{\ell}(\theta)$$

Gradient

Variational Learning to find $q_\lambda(\theta)$

$$\min_{q_\lambda \in \mathbb{Q}} \mathscr{L}(q_\lambda) = \sum_{j=1}^{t} \mathbb{E}_{q_\lambda}[\ell_j] + KL(q_\lambda \| p_0)$$

$\propto e^{-\ell_0}$

Bayesian Learning Rule

$$\lambda \leftarrow \lambda - \rho \; F(q_\lambda)^{-1} \nabla \mathscr{L}(q_\lambda)$$

Natural Gradient

$$\lambda \leftarrow \lambda - \rho \; \nabla_\mu \mathscr{L}(\lambda)$$

Algorithms (such as SGD/Adam) are special cases of BLR obtained by choosing specific exp-family q_λ with natural parameter λ and expectation parameter μ.

1. Khan and Rue, The Bayesian Learning Rule, JMLR (2023).

See Section 1.3.1 in Khan and Rue, 2023

Deriving Gradient Descent from BLR

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

BLR:
$$\lambda \leftarrow \lambda - \rho \, \nabla_\mu \Big(\mathbb{E}_q[\bar{\ell}] - \mathcal{H}(q) \Big)$$
$$m \leftarrow m - \rho \, \nabla_m \mathbb{E}_q[\bar{\ell}]$$
$$m \leftarrow m - \rho \, \mathbb{E}_q[\nabla_\theta \bar{\ell}]$$ Bonnet's theorem
$$m \leftarrow m - \rho \nabla \bar{\ell}(m)$$ First-order delta method

GD:
$$\theta \leftarrow \theta - \rho \nabla \bar{\ell}(\theta)$$

Bayesian learning rule:

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

1. Khan and Rue, The Bayesian Learning Rule, JMLR (2023).

Improving Adam using BLR

RMSprop/Adam

1 $\hat{g} \leftarrow \hat{\nabla}\ell(\theta)$

2 $\hat{h} \leftarrow \hat{g}^2$

3 $h \leftarrow (1-\rho)h + \rho\hat{h}$

4 $\theta \leftarrow \theta - \alpha(\hat{g} + \delta m)/(\sqrt{h} + \delta)$

5

BLR with diagonal-cov Gaussian [4]
(Improved Variational Online Newton)

1 $\hat{g} \leftarrow \hat{\nabla}\ell(\theta)$ where $\theta \sim \mathcal{N}(m, \sigma^2)$

2 $\hat{h} \leftarrow \hat{g} \cdot (\theta - m)/\sigma^2$

3 $h \leftarrow (1-\rho)h + \rho\hat{h}$ $+\rho^2(h-\hat{h})^2/(2(h+\delta))$

4 $m \leftarrow m - \alpha(\hat{g} + \delta m)/(h + \delta)$

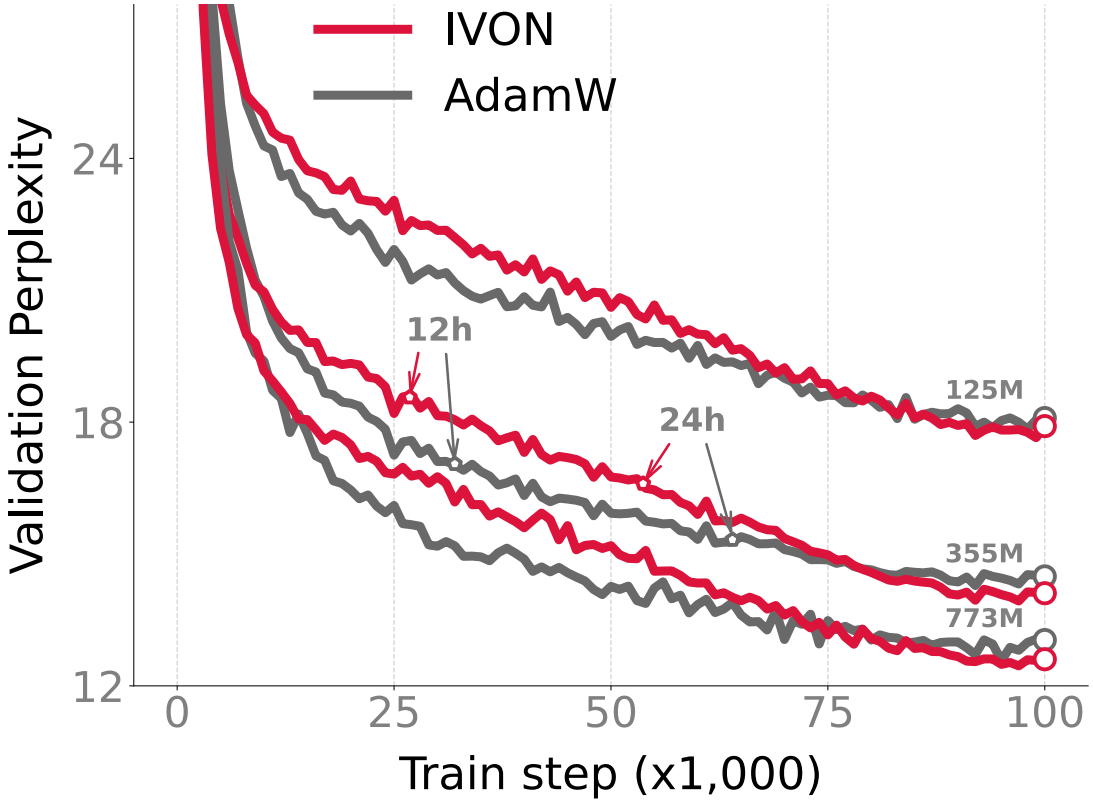
5 $\sigma^2 \leftarrow 1/(N(h+\delta))$

Differences: sampling in line 1, hessian used in line 2 (not g^2),
$h > 0$ constraint in line 3, no square-root over h in line 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).
4. Shen et al. “Variational Learning is Effective for Large Deep Networks.” ICML (2024)

Training GPT-2 from Scratch using BLR

Better performance & uncertainty at the same cost [3]



Trained on OpenWebText data (49.2B tokens).

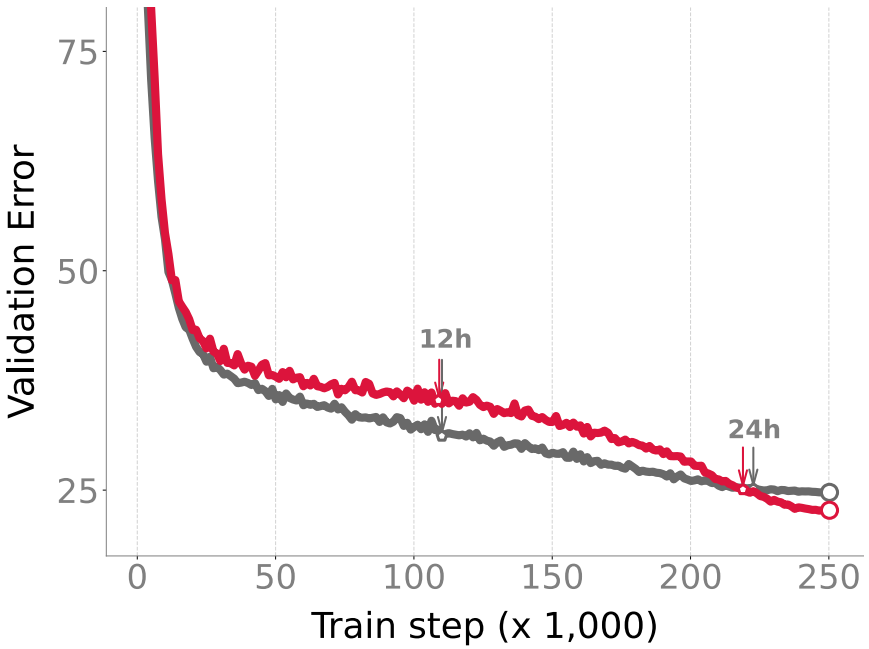
On 773M, we get a gain of 0.5 in perplexity.

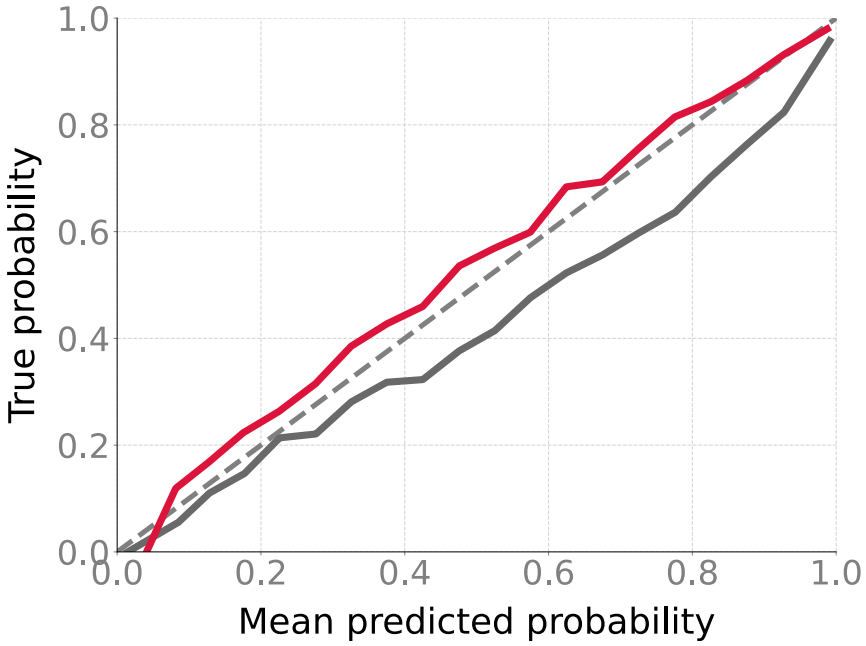
On 355M, we get a gain of 0.4 in perplexity.

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. Shen et al. Variational Learning is Effective for Large Deep Networks, ICML (2024)

Better Calibration

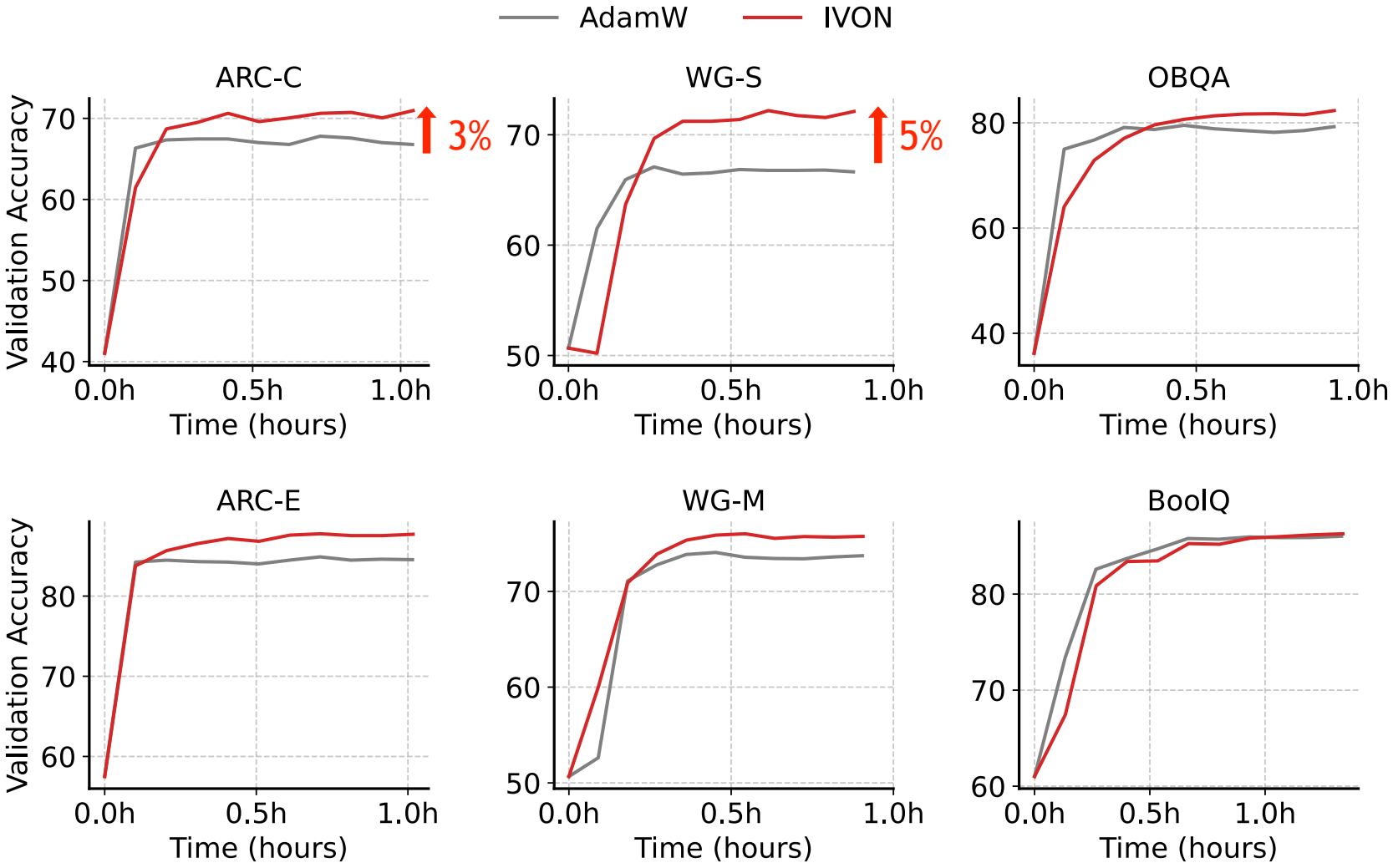
2% better accuracy over AdamW and 1% over SGD. Better calibration (ECE of 0.022 vs 0.066)





LoRA Finetuning [1]

Llama 2 (7 billion)

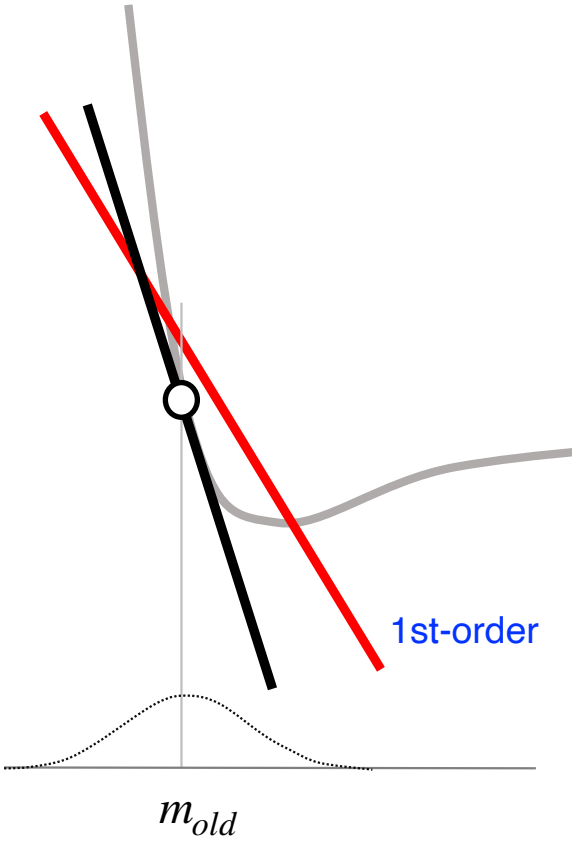


1. Bai et al. "Variational Low-Rank Adaptation Using IVON", FITML workshop at NeurIPS 2024

Eq. 18 in Khan and Nielsen (2018), Eq. 59 Khan and Rue (2023), Eq. 3 in Section 2 in Khan (2025)

Taylor vs Bayes

Why do we recover optimization algorithm from BLR?



GD: $\theta \leftarrow \theta - \rho \nabla_\theta \bar{\ell}(\theta_{old})$

Taylor's surrogate: $\sum_i \theta^\top \nabla \ell_i(\theta_{old})$

BLR with isotropic Gaussian

$$m \leftarrow m - \rho \, \mathbb{E}_{q_{old}}[\nabla \bar{\ell}(\theta)]$$

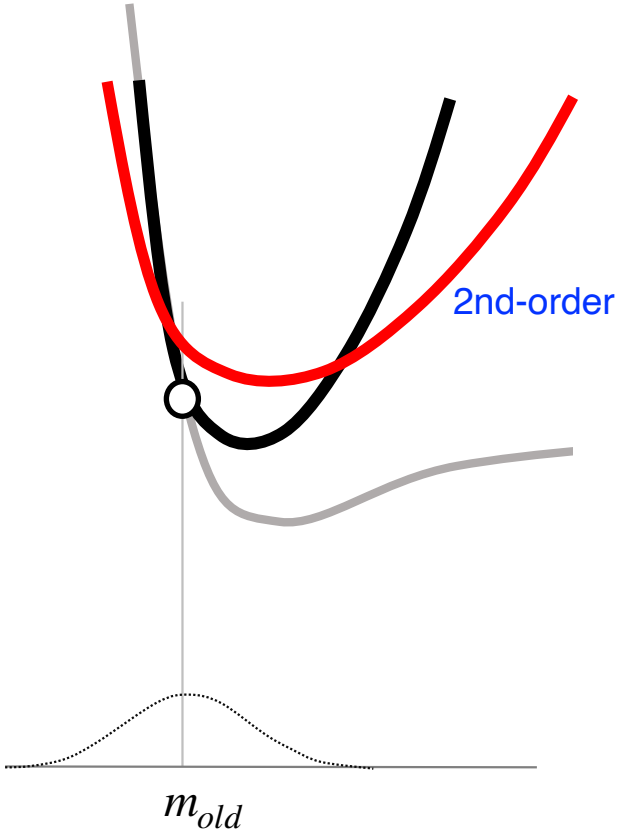
Bayes's surrogate: $\sum_i \theta^\top \mathbb{E}_{q_{old}}[\nabla \ell_i]$

BLR generalizes Taylor!

Bayes Generalizes Taylor

BLR with full cov Gaussian:

$$\sum_i \theta^\top \mathbb{E}_{q_{old}}[\nabla \ell_i] + \frac{1}{2}(\theta - m_{old})^\top \mathbb{E}_{q_{old}}[\nabla^2 \ell_i](\theta - m_{old})$$



BLR with exponential-family:

Suff stats

$$q_{old} \propto \exp(T(\theta)^\top \lambda_{old})$$

Natural gradients

$$= \exp\Big(- \sum_{i=0}^{t} T(\theta)^\top \nabla_\mu \mathbb{E}_{q_{old}}[\ell_i] \Big)$$

Site $\hat{\ell}_{i|old}(\theta)$

Sites are important for adaptation!

Dual-Representation of the BLR

$$q_t \propto \exp(T(\theta)^\top \lambda_t) = \exp\Big(- \sum_{i=0}^{t} T(\theta)^\top \nabla_\mu \mathbb{E}_{q_t}[\ell_i] \Big)$$

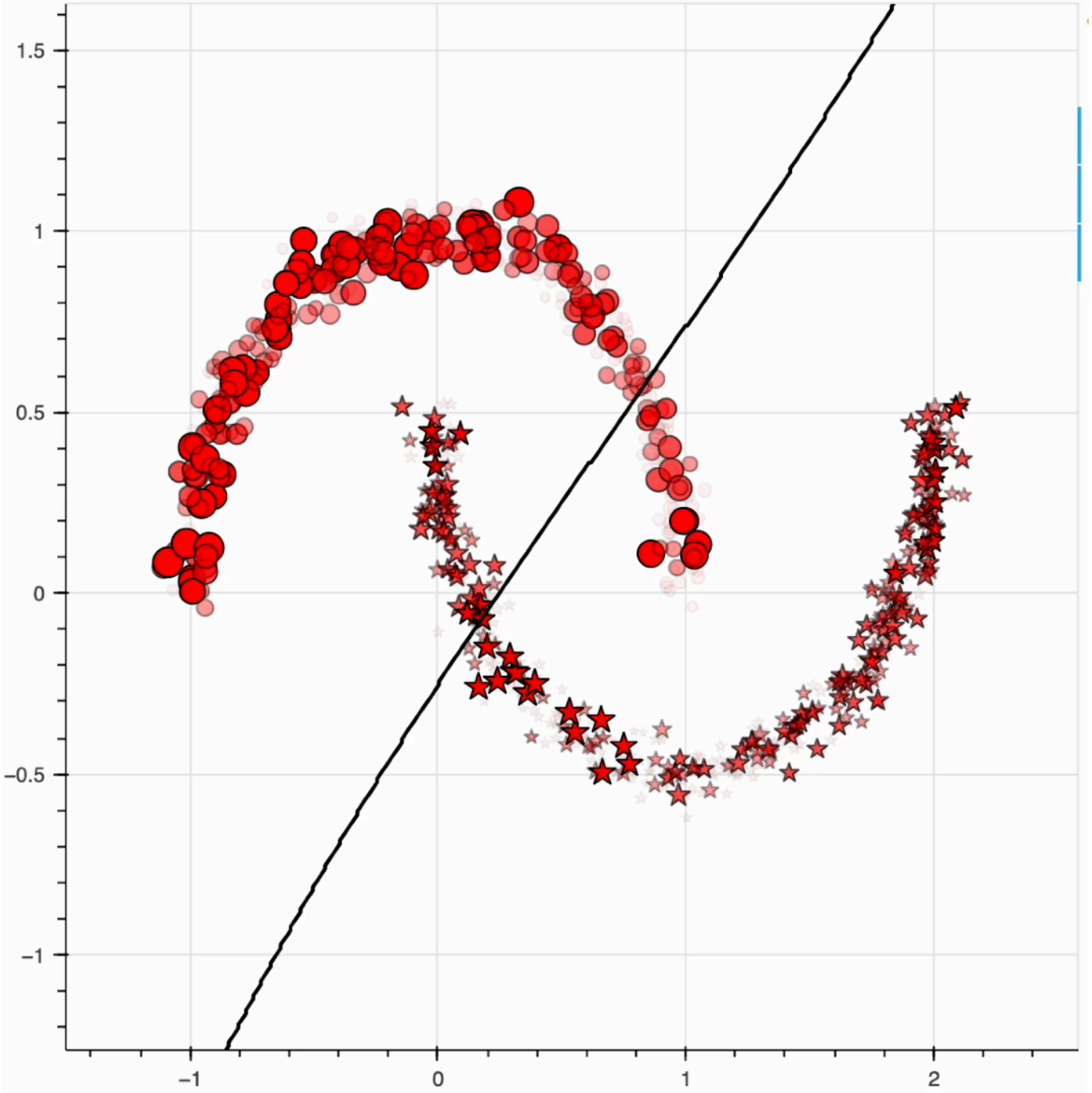
Site $\hat{\ell}_{i|t}(\theta)$

$$q_t \propto \prod_{i=0}^{t} \exp(-\hat{\ell}_{i|t}) \quad \Longleftrightarrow \quad \lambda_t = \sum_{i=0}^{t} \nabla_\mu \mathbb{E}_{q_t}[\ell_i]$$

Posterior — Sites — Natural parameters — Natural gradients

Natural Gradients are additive (representation theorem). Largest ones are the most influential.

1. Khan et al. Fast Dual Variational Inference for Non-Conjugate Latent Gaussian Models. ICML (2013)
2. Khan and Nielsen. Fast yet Simple Natural-Gradient Descent for Variational Inference … ISITA (2018)
3. Khan et al. Approximate Inference Turns Deep Networks into Gaussian Processes. NeurIPS (2019)
4. Adam et al. Dual Parameterization of Sparse Variational Gaussian Processes. NearIPS (2021)
5. Chang et al. Memory-Based Dual Gaussian Processes for Sequential Learning. ICML (2023)
6. Moellenhoff et al. Federated ADMM from Bayesian Duality. arXiv (2025)



The site parameters can be used to generalize Influence Estimators.

Binary classification on the Two Moons dataset.

Big markers with red indicate bigger first derivative for IVON [1]

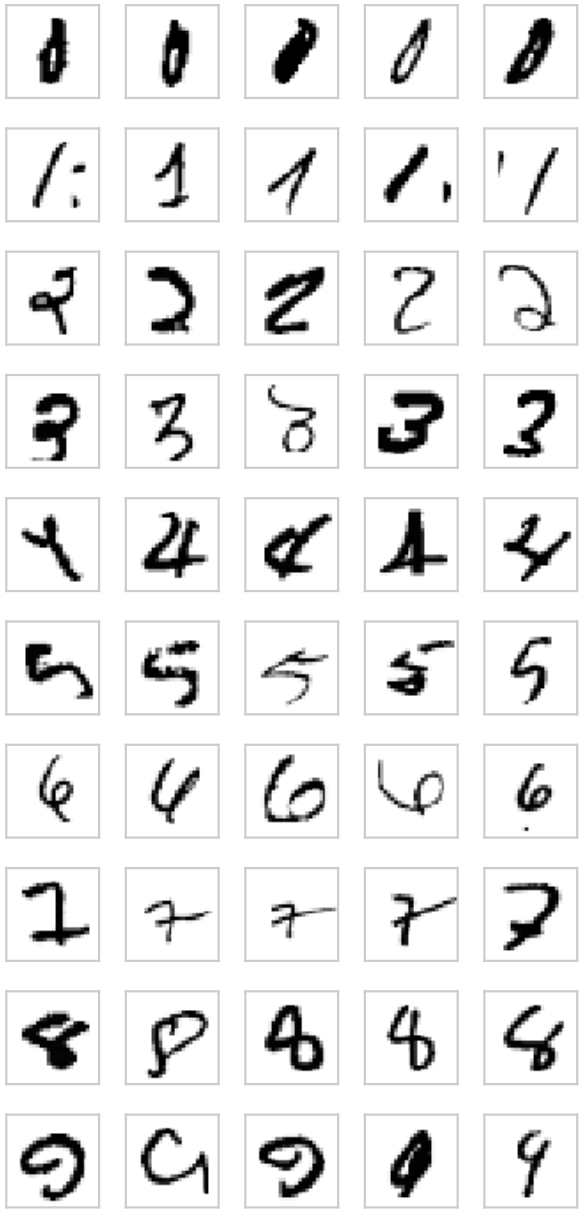
By Rin Intachuen (RIKEN AIP)

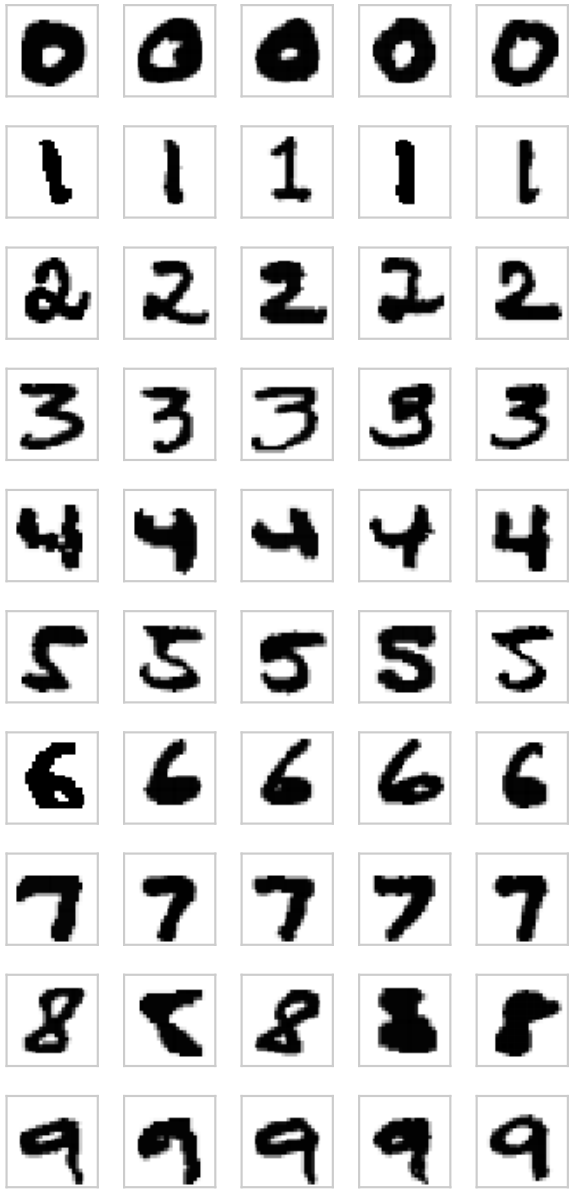
1. Nickl, Xu, Tailor, Moellenhoff, Khan, The memory-perturbation equation, NeurIPS, 2023

High Influence

MNIST with CNN

Low Influence





1. Nickl, Xu, Tailor, Moellenhoff, Khan, The memory-perturbation equation, NeurIPS, 2023

High Influence

Traffic light (ImageNet)

What class is this?

Low Influence

High Influence

Chihuahua class (ImageNet)

Low Influence

Continual Learning

Elastic Weight Consolidation
Variational Continual Learning
Memory Replay Methods
Functional Regularization

Model Merging

Task Arithmetic
Fisher/Hessian-Based Merging
Ensembles Methods

Posterior Correction [1]

Unlearning and Influence

Student-Teacher Learning

Knowledge Distillation
Learning with Privileged information
Incremental SVMs

Federated Learning

FedAvg, FedDyn
Alternating Direction Method of Multipliers (ADMM)
Alternating Minimization Algorithm (AMA)
Partition Variational Inference

1. Khan, Knowledge Adaptation as Posterior Correction, arXiv (2025)

Variational Formulation of Online Bayesian Inference

Bayes' Rule: $p_{t+1}(\theta) \propto p_0(\theta) \prod_{j=1}^{t+1} e^{-\ell_j(\theta)} \propto p_t(\theta)\, e^{-\ell_{t+1}(\theta)}$

Variational formulation:

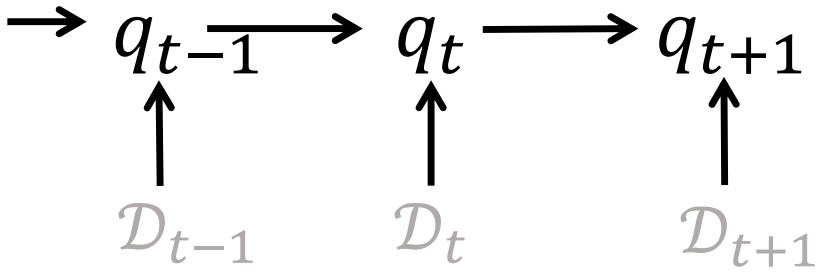
Batch: $q_{t+1} = \arg\min_q \sum_{j=1}^{t+1} \mathbb{E}_q[\ell_j] + KL(q \| p_0)$

Online [1]: $\hat{q}_{t+1} = \arg\min_q \mathbb{E}_q[\ell_{t+1}] + KL(q \| q_t)$

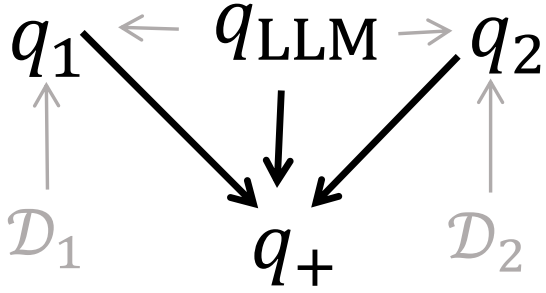
How inaccurate is $\hat{q}_{t+1}$? Can we correct it to exactly recover q_{t+1}? This is the goal of posterior correction.

1. Nguyen et al. Variational continual learning. ICLR (2018)

Continual Learning

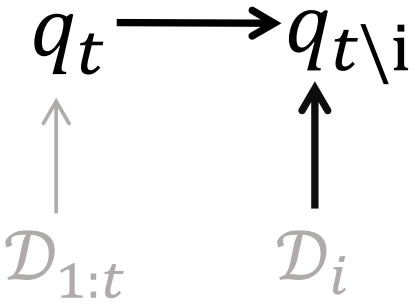


Model Merging

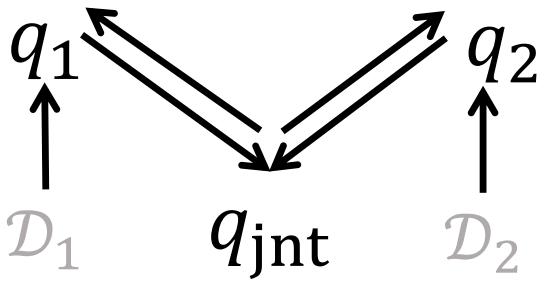


Posterior Correction [1]

Unlearning and Influence

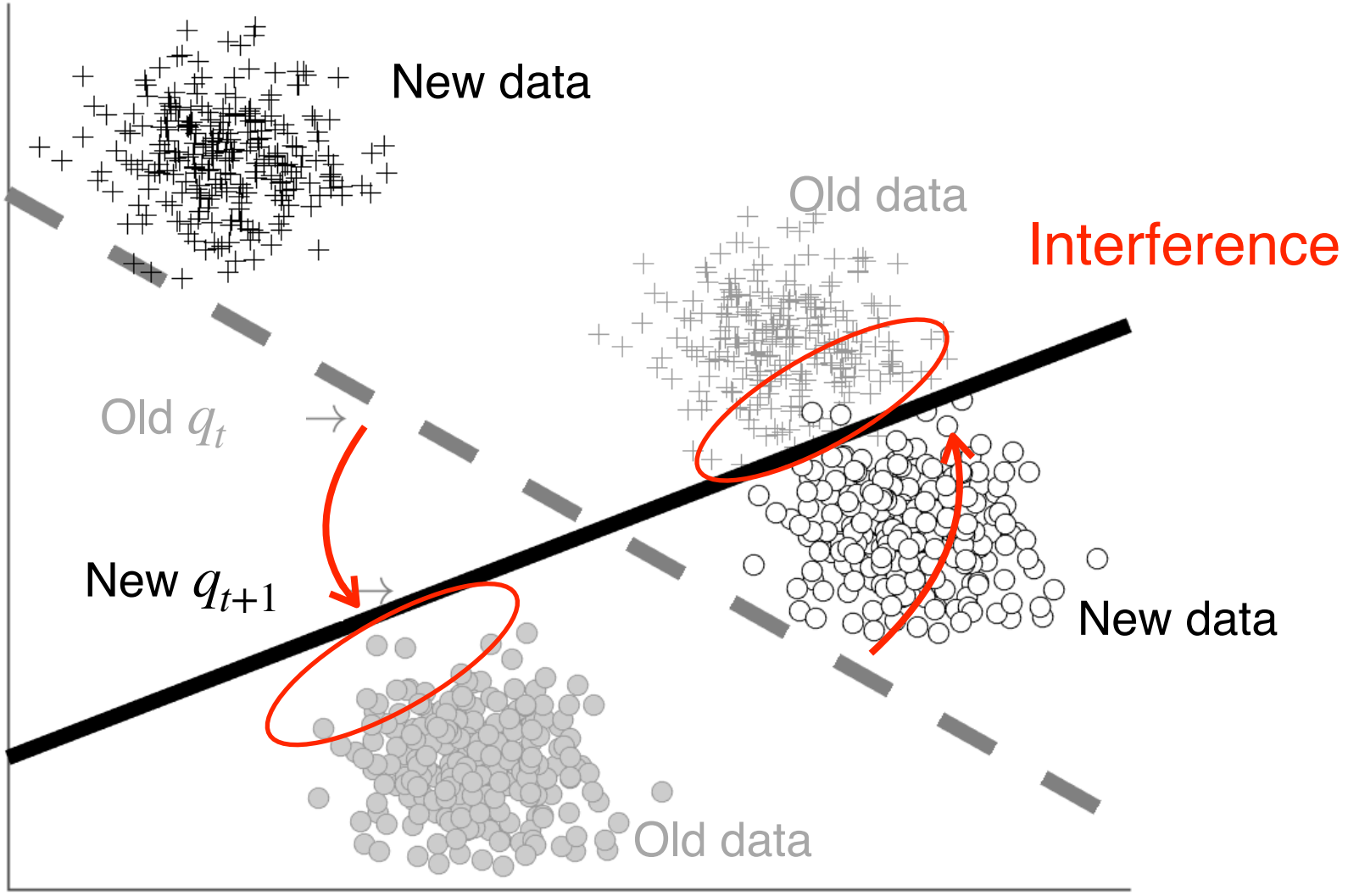


Federated Learning



1. Khan, Knowledge Adaptation as Posterior Correction, arXiv (2025)

Correct the Past due to the Interference Created by the Future



Eq. 4 in Khan (2025)

Posterior Correction

We will use the site functions to correct the posterior!

$$\text{Batch:} \quad q_{t+1} = \arg\min_q \sum_{j=1}^{t+1} \mathbb{E}_q[\ell_j] + KL(q\|p_0) \qquad \frac{q_t}{\prod_{i=0}^{t} \exp(-\hat{\ell}_{j|t})}$$

$$= \arg\min_q \mathbb{E}_q[\ell_{t+1}] + KL(q\|q_t) + \sum_{j=0}^{t} \mathbb{E}_q[\ell_j - \hat{\ell}_{j|t}]$$

Correction

$$\text{Online:} \quad \hat{q}_{t+1} = \arg\min_q \mathbb{E}_q[\ell_{t+1}] + KL(q\|q_t)$$

Very simple proof (3 lines). Exact recovery in general!

1. Khan, Knowledge Adaptation as Posterior Approximation, arXiv (2025)

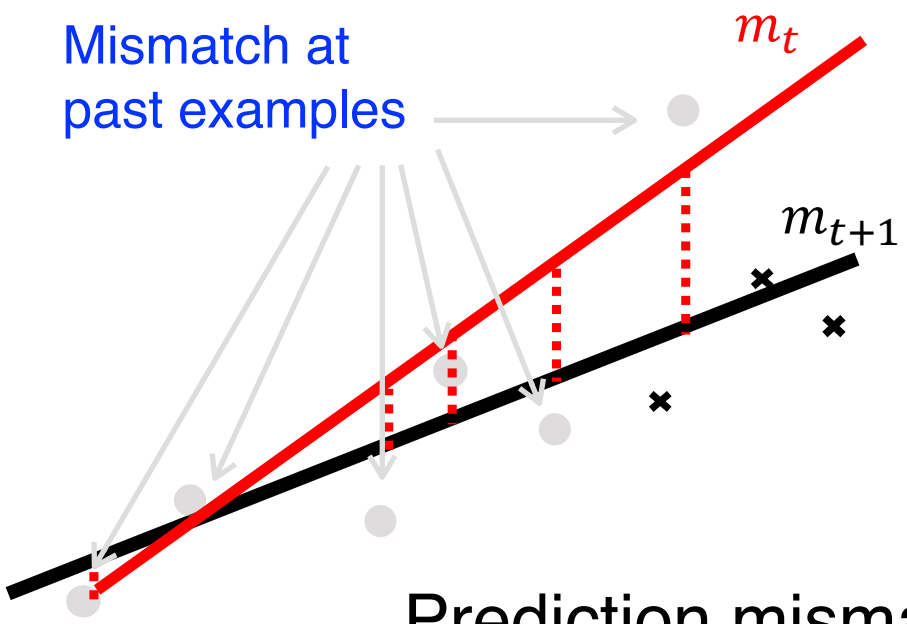
Eq. 7 in Khan (2025)

Correction as Prediction Mismatch

Linear regression with isotropic Gaussian posterior

$$m_{t+1} = \arg\min_m \; \mathbb{E}_q[\frac{1}{2}(y_{t+1} - x_{t+1}^\top \theta)^2] + KL\left[\mathcal{N}(m, I) \| \mathcal{N}(m_t, I)\right]$$

$$+ \sum_{j=1}^{t} \frac{1}{2}(x_j^\top m_t - x_j^\top m)^2 \; + \ldots$$



Error due to mean-field is fixed by the correction!

$$\frac{1}{2}(m - m_t)^\top \left(\sum_{j=1}^{t} x_j x_j^\top \right)(m - m_t)$$

Prediction mismatch is simpler to implement!

Knowledge-Adaptation Prior

Posterior correction with isotropic Gaussian reduces to "prediction or gradient mismatch" (K-priors) [1]

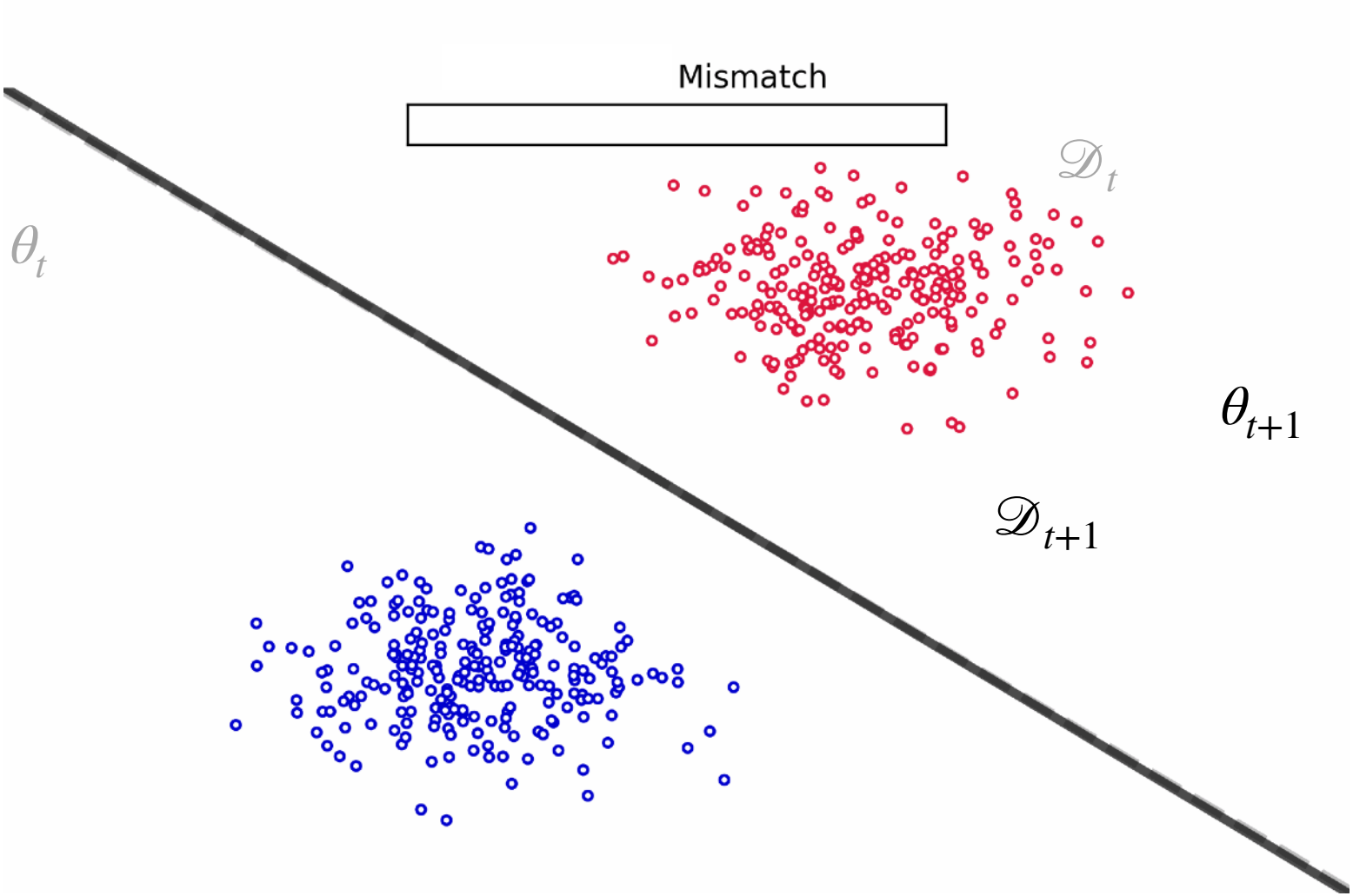
$$\theta_{t+1} = \arg\min_{\theta} \ell_{t+1} + \frac{\rho}{2}\|\theta - \theta_t\|^2 + \sum_{j=1}^{t} \ell_j \left(\hat{y}_j(\theta_t),\, \hat{y}_j(\theta)\right)$$

Many adaptation methods reduce this mismatch [2-9] and Posterior Correction generalizes it!

1. Khan and Swaroop. Knowledge-Adaptation Priors, NeurIPS (2021).
2. Kirkpatrick et al. Overcoming catastrophic forgetting in neural networks. PNAS, 2017.
3. Benjamin et al. Measuring and regularizing networks in function space. ICLR 2019.
4. Hinton et al. Distilling the knowledge in a neural network, arXiv, 2015.
5. Buzzega et al. Dark experience for general continual learning: a strong, simple baseline. NeurIPS 2020.
6. Cauwenberghs and Poggio. Incremental and decremental SVM learning. NeurIPS, 2001.
7. Vapnik and Izmailov. Learning using privileged information: similarity control and …. JMLR, 2015.
8. Lopez-Paz and Ranzato. Gradient episodic memory for continual learning, NIPS'17
9. Csató and Opper. Sparse on-line Gaussian processes. Neural computation, 2002.

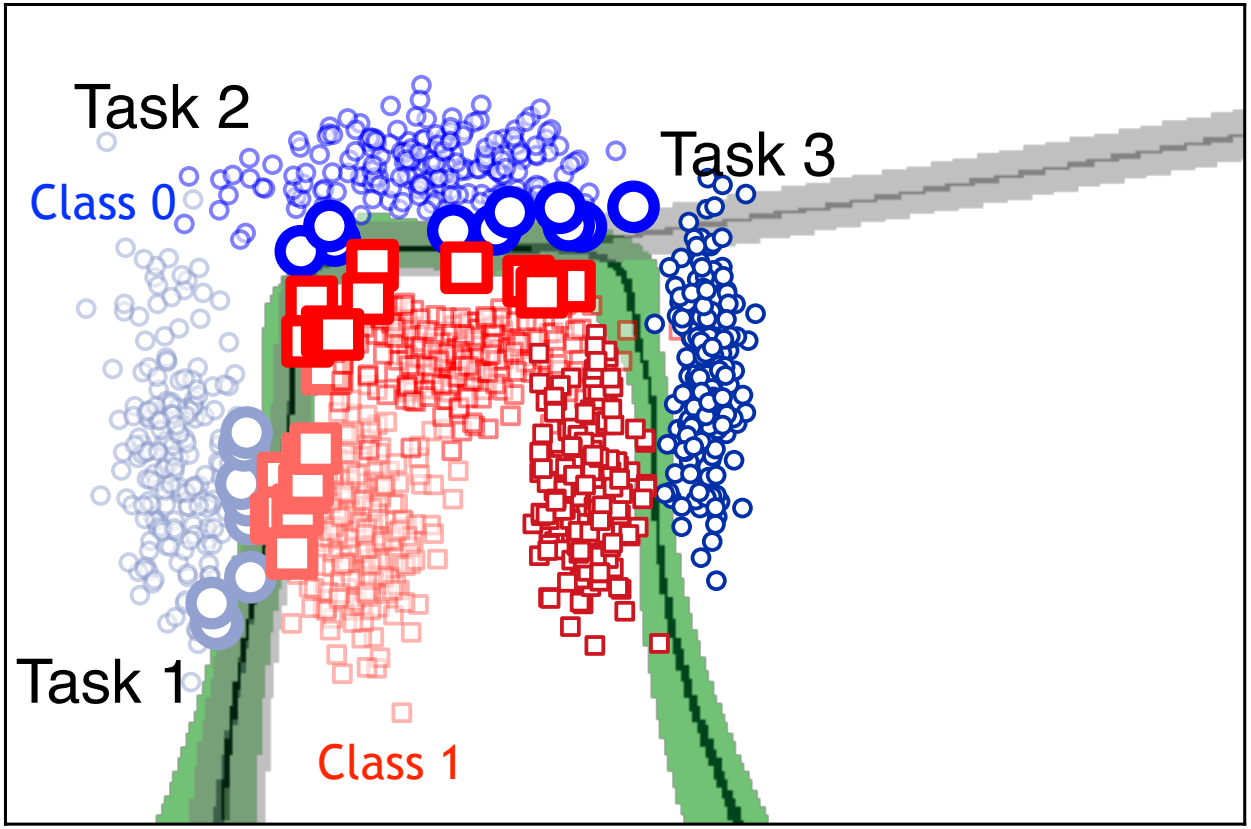
From Quick to Slow Adaptation

Correction as Information Gain



Quick Adaptation with Compact Memory

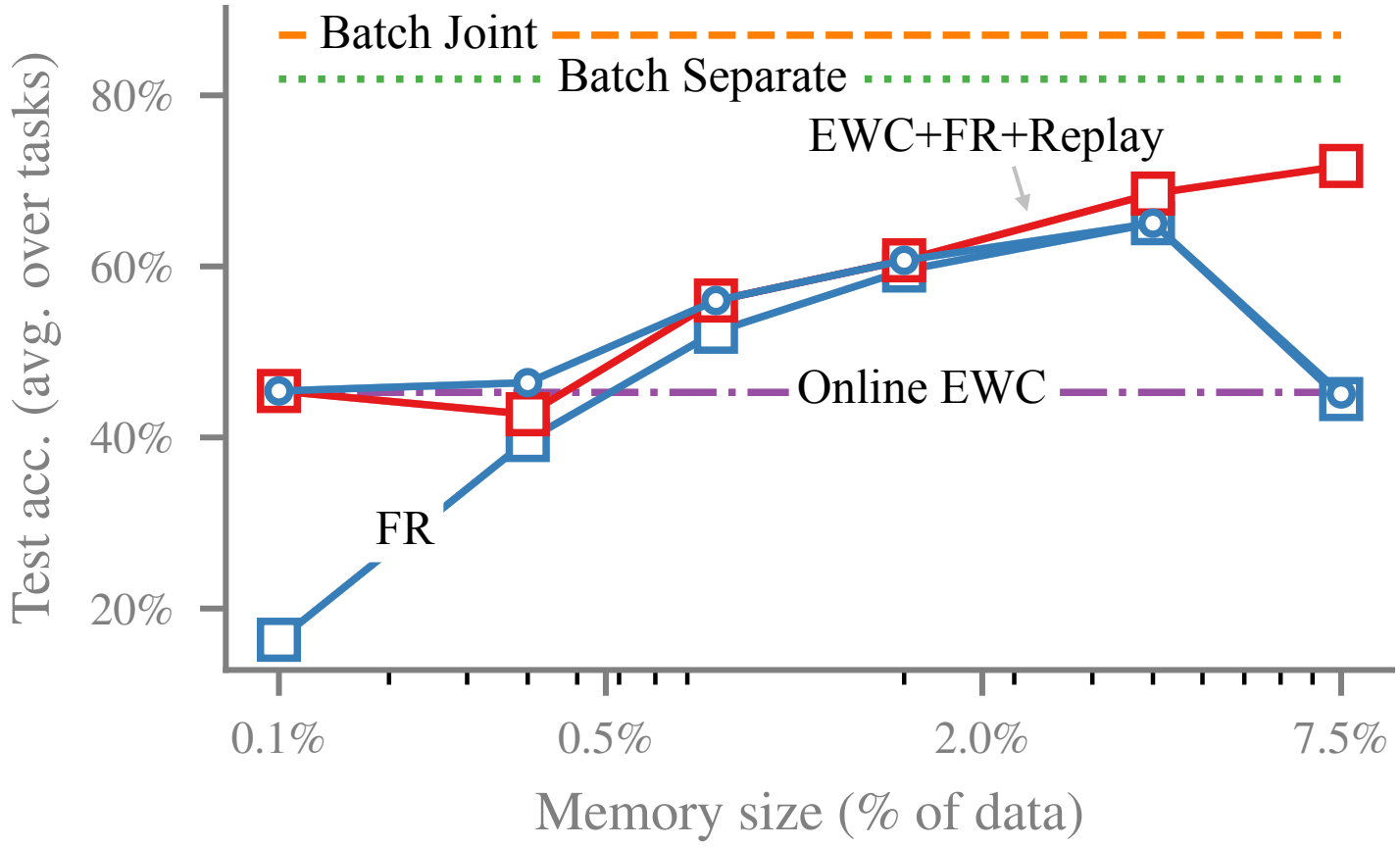
Choose memories where interference is more likely.
Small correction $\Longrightarrow$ Small memory $\Longrightarrow$ Quick adaptation



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

Combine Methods to Reduce Correction

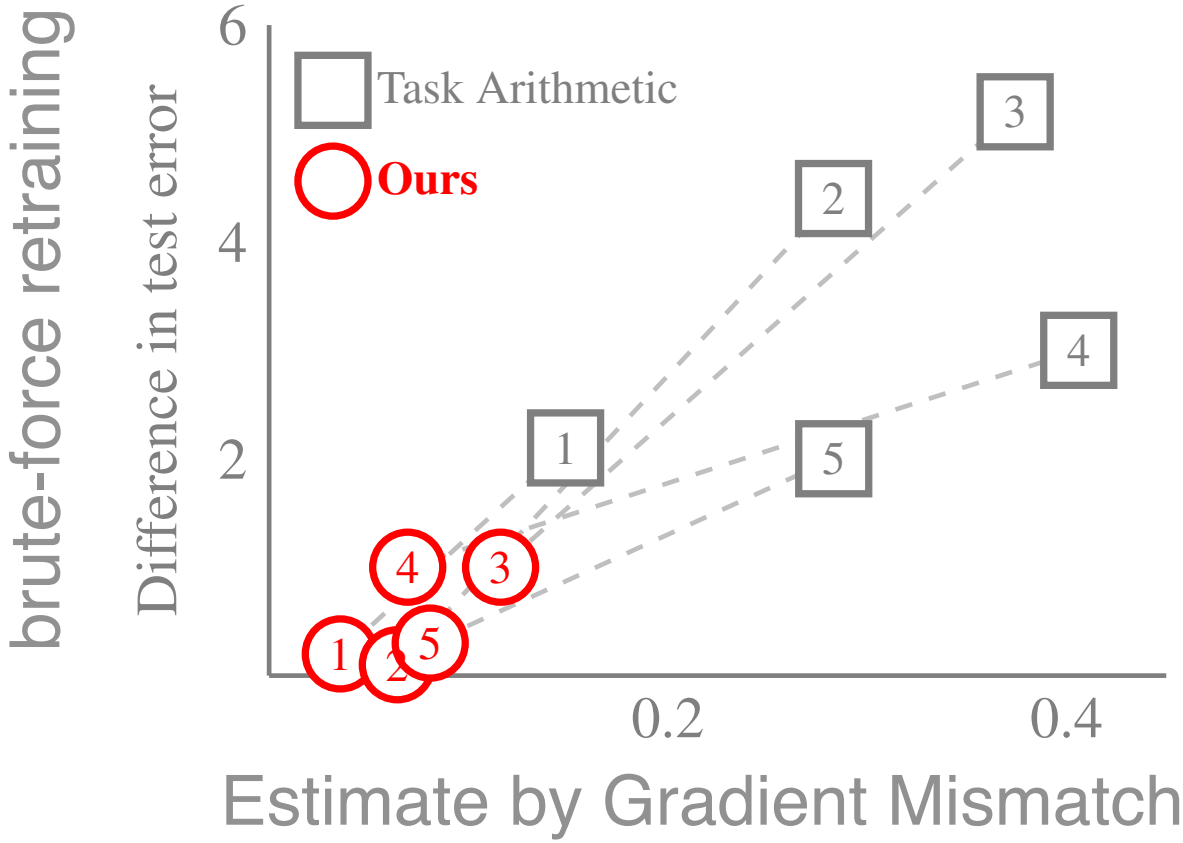
Get 78% accuracy with 7.5% (random) memory



ImageNet with ResNet-18 [1]

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

Reducing Correction Improves Performance in LLM fine-tuning



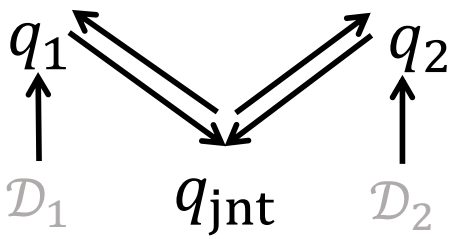
RoBERTa on IMDB

1. Daheim et al. Model merging by uncertainty-based gradient matching, ICLR (2024).

Summary of Federated Learning, Model Merging, and Memories etc.

Recover q_{jnt} from q_1 and q_2

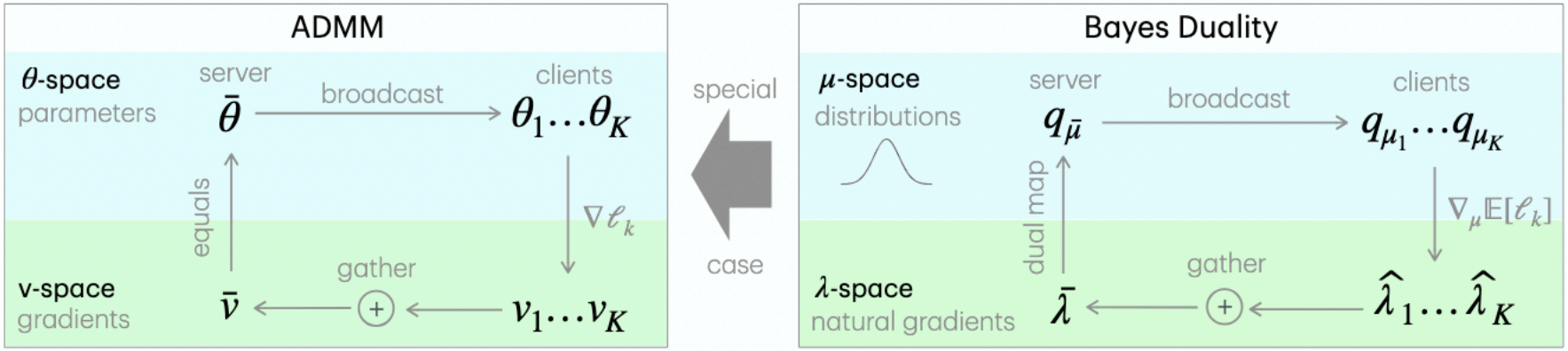
$$q_{jnt} = \arg\min_{q} KL(q \| q_1 q_2) + \sum_{j=1}^{2} \mathbb{E}_q[\ell_j - \hat{\ell}_{j|j}]$$



By choosing different q, we get different strategies (better q gives better merging) [1,2]. Same is true for federated learning [3,4]. All of them will benefit from compact memories designed to reduce corrections [5].

1. Daheim et al. Model merging by uncertainty-based gradient matching, ICLR (2024).
2. Monzon et al. How to Weight Multitask Finetuning? Fast Previews via Bayesian Model-Merging, 2024
3. Swaroop, Khan, Doshi, Connecting Federated ADMM to Bayes, ICLR 2025
4. Moellenhoff et al. Federated ADMM from Bayes Duality, arXiv, 2025
5. Nickl, Xu, Tailor, Moellenhoff, Khan, The memory-perturbation equation, NeurIPS (2023)

ADMM as a special case of Bayes (Dual)



Algorithm 1 BayesADMM (Fig. 2b) for Gaussians with diagonal covariance. Additional steps when compared to FederatedADMM are highlighted in red. Implementation details are in App. D.

Hyperparameters: Prior precision $\delta > 0$, step-sizes $\rho > 0$ and $\gamma > 0$.
Initialize: $\mathbf{v}_k \leftarrow 0$, $\mathbf{u}_k \leftarrow 0$, $\bar{\mathbf{m}} \leftarrow 0$, $\bar{\mathbf{s}} \leftarrow \delta$, $\alpha \leftarrow 1/(1+\rho K)$.

while not converged **do**
  Broadcast $\bar{\mathbf{m}}$ and $\bar{\mathbf{s}}$ to all clients.
  for each client $1, \ldots, K$ in parallel **do**
    Local training on $\ell_k(\boldsymbol{\theta}) + \boldsymbol{\theta}^\top \mathbf{v}_k - \frac{1}{2}\boldsymbol{\theta}^\top(\mathbf{u}_k \boldsymbol{\theta}) + \frac{\rho}{2}\|\boldsymbol{\theta} - \bar{\mathbf{m}}\|^2_{\bar{\mathbf{s}}}$ ▷ Using IVON [53]
    $\mathbf{v}_k \leftarrow \mathbf{v}_k + \gamma\,(\mathbf{s}_k \mathbf{m}_k - \bar{\mathbf{s}}\bar{\mathbf{m}})$
    $\mathbf{u}_k \leftarrow \mathbf{u}_k + \gamma\,(\mathbf{s}_k - \bar{\mathbf{s}})$ ▷ An additional dual variable.
  end for
  Gather $\mathbf{m}_k$, $\mathbf{v}_k$ and $\mathbf{s}_k$, $\mathbf{u}_k$ from all clients.
  $\bar{\mathbf{m}} \leftarrow (1-\alpha)\,\mathtt{Mean}(\mathbf{s}_{1:K}\mathbf{m}_{1:K}) + \alpha\,\mathtt{Sum}(\mathbf{v}_{1:K})$
  $\bar{\mathbf{s}} \leftarrow (1-\alpha)\,\mathtt{Mean}(\mathbf{s}_{1:K}) + \alpha\,[\delta \mathbf{1} + \mathtt{Sum}(\mathbf{u}_{1:K})]$ ▷ Two additional steps for precision $\bar{\mathbf{s}}$
  $\bar{\mathbf{m}} \leftarrow \bar{\mathbf{m}}/\bar{\mathbf{s}}$
end while

Scenario	Method	Test accuracy (↑ larger is better) 10 rounds	25 rounds	50 rounds
MLP, 10 clients homogeneous FMNIST(*)	FedAvg	72.3±0.4	77.7±0.3	80.0±0.2
	FedProx	72.2±0.3	77.4±0.1	80.3±0.1
	FedDyn	75.3±0.8	77.5±0.8	78.2±0.5
	FedLap	72.1±0.2	77.1±0.1	80.2±0.1
	FedLap-Cov	75.0±0.6	79.8±0.4	81.8±0.1
	BayesADMM@$\bar{m}$	**80.4**±0.2	**83.1**±0.1	**83.4**±
	BayesADMM	**80.6**±0.2	**83.5**±0.1	**84.1**±
MLP, 10 clients hetereogeneous FMNIST(*)	FedAvg	70.4±0.9	74.3±0.5	76.0±0.7
	FedProx	69.9±0.4	74.7±0.6	76.9±0.9
	FedDyn	73.0±0.6	74.6±0.4	74.6±0.5
	FedLap	71.3±0.9	74.3±0.4	77.6±0.7
	FedLap-Cov	74.6±0.7	78.3±1.0	80.5±0.6
	BayesADMM@$\bar{m}$	**77.0**±0.8	**81.4**±0.4	**82.1**±
	BayesADMM	**77.0**±0.8	**81.5**±0.5	**82.3**±
CNN from Acar et al. [1] 10 clients heterogeneous CIFAR-10	FedAvg	62.8±3.1	65.4±1.8	66.0±1.5
	FedProx	**64.3**±2.0	65.9±1.6	66.3±1.4
	FedDyn	63.6±1.1	64.7±0.6	65.4±1.0
	FedLap	60.2±2.4	66.4±1.1	66.5±1.2
	FedLap-Cov	58.4±2.4	65.4±1.1	67.5±1.2
	BayesADMM@$\bar{m}$	**63.8**±1.4	**69.5**±0.8	**70.2**±
	BayesADMM	**63.8**±1.4	**69.5**±0.8	**70.3**±

+6%

+8%

+5%

Adaptive Bayesian Intelligence

- Adaptive Intelligence = Bayesian Computation
- Part 1: Bayesian Learning Rule [1]
  - Foundational way to derive learning-algorithms
  - Application to Deep Learning [2]
- Part 2: Posterior Correction [3]
  - Foundational way to derive adaptation-algorithms
  - Application to continual learning [4-5]
  - But also for LLM merging, Federated Learning etc.
- Adaptive Bayesian Intelligence: A roadmap.

1. Khan and Rue, The Bayesian Learning Rule, JMLR (2023)
2. Shen et al. Variational Learning is Effective for Large Deep Networks, ICML (2024)
3. Khan. Knowledge Adaptation as Posterior Correction, arXiv (2025)
4. Khan and Swaroop. Knowledge-Adaptation Priors, NeurIPS (2021).
5. Pan et al. Continual Deep Learning by Functional Regularisation of Memorable Past, NeurIPS, 2020

Questions for the future

- What should the algorithm remember?
- And what new experiences should it seek?
- Memory should be chosen to minimize the corrections that may arise in the future.
- New experiences should be chosen to enable easy-enough corrections (not too daunting for the learner)
- Future is unknown but the algorithm has the freedom to explore by “fixing the past & choosing the future”

Fixing
Choosing

Webpage:https://bayesduality.github.io/

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

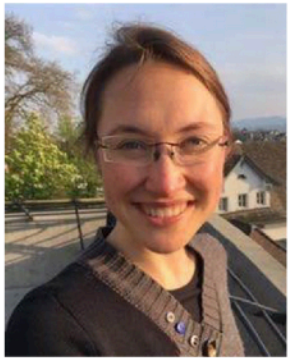
Received total funding of JPY 220M + EUR 500K through the CREST-ANR grant! Thanks to JST for their generous funding!

Bayes-Duality Workshop (June 25-27, 2025)

https://bayesduality.github.io/workshop_2025.html

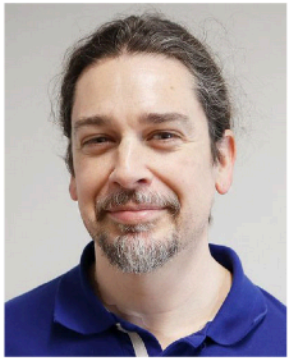
Abeba Birhane

Trinity College Dublin, Ireland



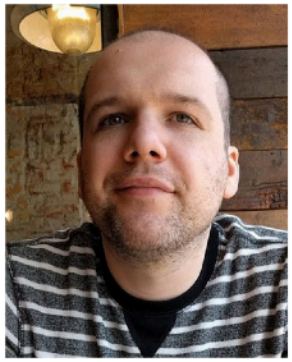
Rebekka Burkholz

Helmholtz Center CISPA, Germany



André Martins

Instituto Superior Tecnico, Portugal



Razvan Pascanu

Deepmind, UK

Marcus Rohrbach

TU Darmstadt, Germany

Mark van der Wilk

University of Oxford, UK

David Rügamer

Ludwig-Maximilians-Universität München, Germany

Diverse topics: Bayes, Optimization, Information Geometry, Continual Learning, Federated Learning, Active learning, RL, Model understanding, Data Attributions, LLMs, etc.

Adaptive Bayesian Intelligence Team

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

Emtiyaz Khan
Team Leader

Thomas Möllenhoff
Research Scientist

Keigo Nishida
Special Postdoctoral Researcher
RIKEN BDR

Hugo Monzón Maldonado
Postdoctoral Researcher

Christopher Johannes Anders
Postdoctoral Researcher

Yohan Jung
Postdoctoral Researcher

Anita Yang
Part-Time Student
The University of Tokyo

Bai Cong
Part-Time Student
Tokyo Institute of Technology

Eiki Shimizu
Part-Time Student
Institute of Statistical Mathematics

Giulia Lanzillotta
Intern
ETH Zurich

Adrian R. Minut
Intern
Sapienza, University of Rome

Florian Seligmann
Intern
Karlsruhe Institute of Technology

Guiomar Pescador Barrios
Intern
Imperial College London

Henrique Da Silva Gameiro
Intern
EPFL, Switzerland

Joseph Austerweil
Visiting Scientist
University of Winsconsin-Madison

Pierre Alquier
Visiting Scientist
ESSEC Business School

Geoffrey Wolfer
Visiting Scientist
Waseda University

Rio Yokota
Visiting Scientist
Tokyo Institute of Technology

Dharmesh Tailor
Remote Collaborator
University of Amsterdam

And many of our collaborators!