University College London

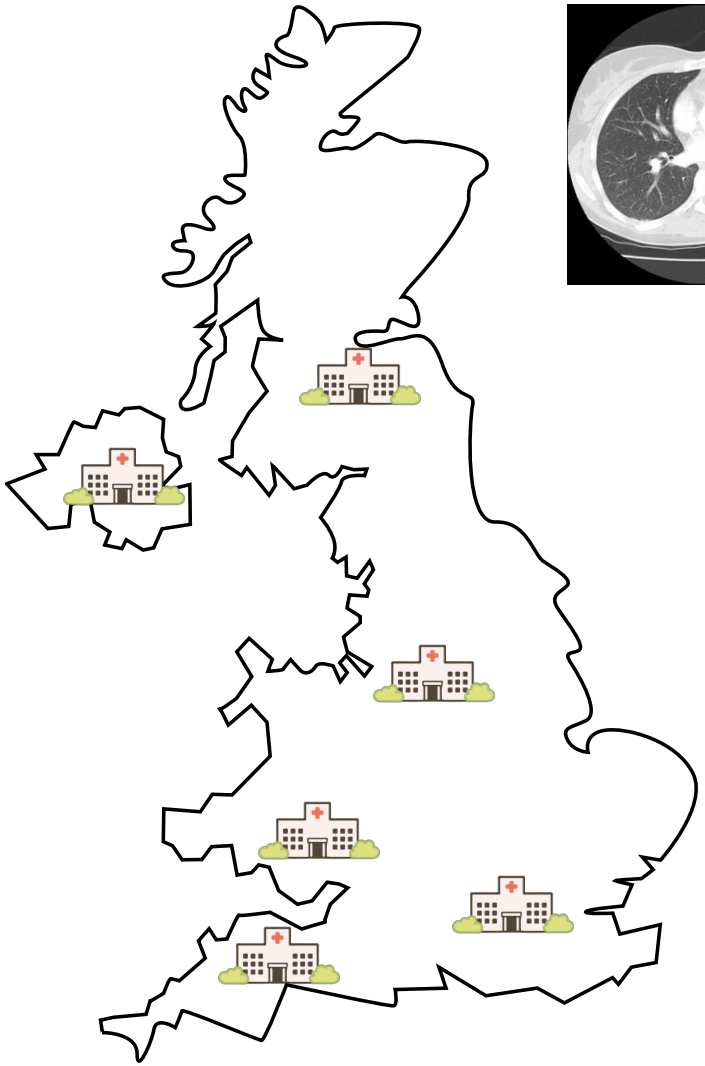


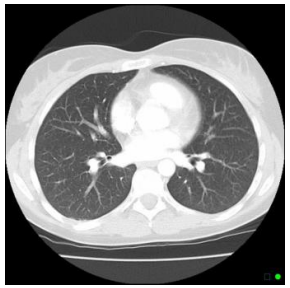
Adapting knowledge using a Bayesian lens

Siddharth Swaroop (s.swaroop@ucl.ac.uk)
Lecturer in AI, UCL Computer Science

Swaroop, Khan and Doshi-Velez, *"Connecting Federated ADMM to Bayes"*, ICLR 2025
Möllenhoff*, Swaroop*, Doshi-Velez and Khan, *"Federated ADMM from Bayesian Duality"*, ICLR 2026
Subarnaduti et al., "*Fast and Slow Variational Continual Learning*", arxiv:2606.24007
Khan* & Swaroop*, *"Knowledge-adaptation priors"*, NeurIPS 2021
Daxberger et al., *"Improving Continual Learning by Accurate Gradient Reconstructions of the Past"*, TMLR 2023 & CoLLAs 2024

Challenges when training models





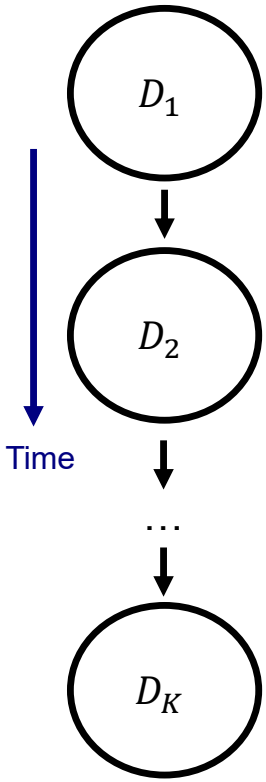
- Example: Detect lung cancer from CT scans
- Challenge 1: data is **distributed** across locations
  - Patient data **cannot leave hospital** server
  - **Heterogenous data split** across hospitals
  - Models need as much data as possible
- Challenge 2: data arrives **sequentially** over time
  - Hospitals **add more scans** every day
  - Want to quickly adapt to new data
- Challenge 3: data may need to be **forgotten**
  - Patients can **revoke consent** (“right to be forgotten”)
  - Want to quickly remove / delete data from model

Knowledge adaptation in machine learning



- Many such examples of **knowledge adaptation**: literature is disconnected
- Federated learning: data is **distributed heterogeneously** across locations
  - Eg user data on **mobile phones**
  - Minimise communication cost while ensuring data is secure
- Continual learning: systems need to **adapt quickly** to changing environments
  - Eg updating LLMs every day (with latest news)
  - Cannot store/replay all past data: computational (or privacy) reasons
- **Bayesian lens** to unify knowledge-adaptation tasks
  - Focus on approximating each information source
  - Also: unlearning, transfer learning, model merging, ...

Formalising continual learning + Bayes' rule



Posterior Likelihood x Prior

$$p(\boldsymbol{w}|\mathcal{D}_1) \propto p(\mathcal{D}_1|\boldsymbol{w})p(\boldsymbol{w})$$

$$p(\boldsymbol{w}|\mathcal{D}_1, \mathcal{D}_2) \propto p(\mathcal{D}_2|\boldsymbol{w})\cancel{p(\mathcal{D}_1|\boldsymbol{w})p(\boldsymbol{w})}$$

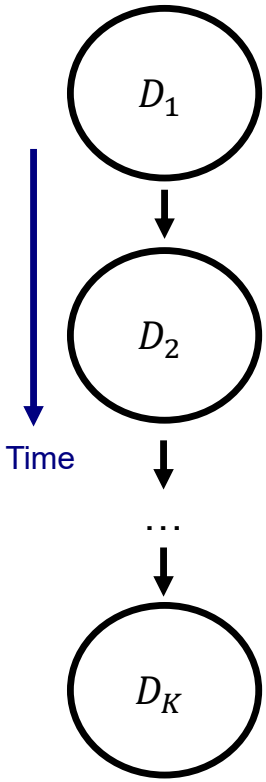
$$p(\boldsymbol{w}|\mathcal{D}_1)$$

No need to access past data

$$p(\boldsymbol{w}|\mathcal{D}_1, ..., \mathcal{D}_K) \propto p(\mathcal{D}_K|\boldsymbol{w})\cancel{...p(\mathcal{D}_1|\boldsymbol{w})p(\boldsymbol{w})}$$

$$p(\boldsymbol{w}|\mathcal{D}_{K-1})$$

Formalising continual learning + Bayes' rule



Posterior Likelihood x Prior

$$p(\boldsymbol{w}|\mathcal{D}_1) \propto p(\mathcal{D}_1|\boldsymbol{w})p(\boldsymbol{w}) \approx t_1(\boldsymbol{w})p(\boldsymbol{w}) \propto q_1(\boldsymbol{w})$$

$$p(\boldsymbol{w}|\mathcal{D}_1,\mathcal{D}_2) \propto p(\mathcal{D}_2|\boldsymbol{w})\cancel{p(\mathcal{D}_1|\boldsymbol{w})p(\boldsymbol{w})} \approx t_2(\boldsymbol{w})\cancel{t_1(\boldsymbol{w})p(\boldsymbol{w})} \propto q_2(\boldsymbol{w})$$

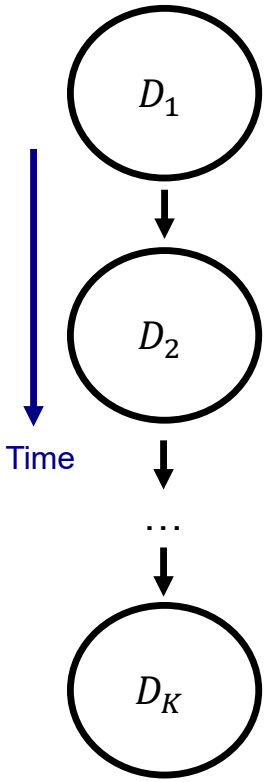
$p(\boldsymbol{w}|\mathcal{D}_1)$ $q_1(\boldsymbol{w})$

No need to access past data

Approach: approximate each likelihood with a distribution, $t_k(\boldsymbol{w}) \approx p(\mathcal{D}_k|\boldsymbol{w})$

- Eg Gaussians
- Like **site distributions**

Continual learning and Federated learning



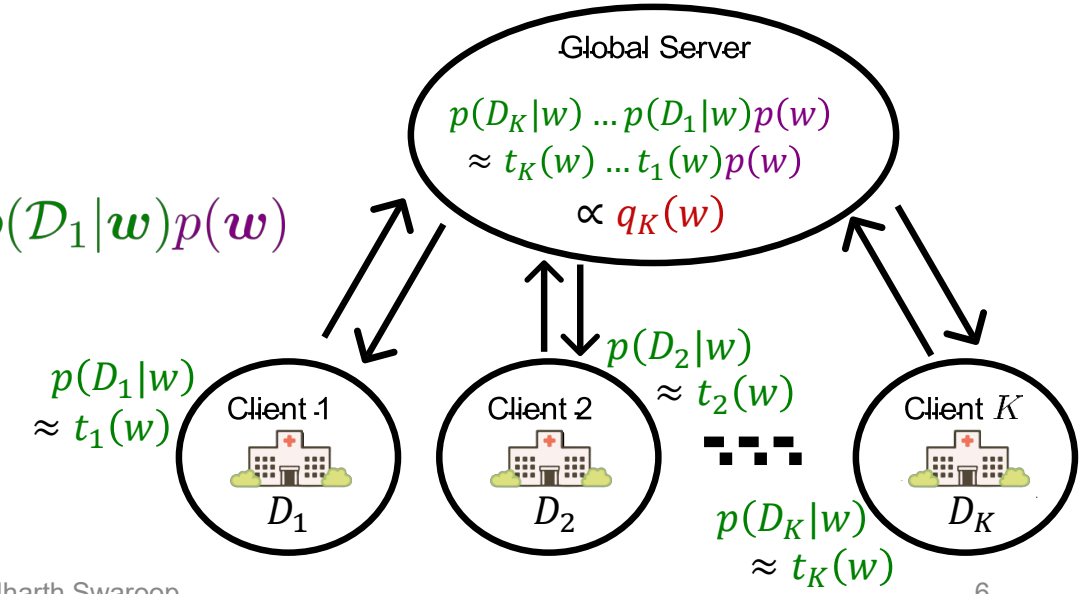
Posterior Likelihood x Prior

$$p(\boldsymbol{w}|\mathcal{D}_1) \propto p(\mathcal{D}_1|\boldsymbol{w})p(\boldsymbol{w}) \approx t_1(\boldsymbol{w})p(\boldsymbol{w}) \propto q_1(\boldsymbol{w})$$

$$p(\boldsymbol{w}|\mathcal{D}_1, \mathcal{D}_2) \propto p(\mathcal{D}_2|\boldsymbol{w})\cancel{p(\mathcal{D}_1|\boldsymbol{w})p(\boldsymbol{w})} \approx t_2(\boldsymbol{w})\cancel{t_1(\boldsymbol{w})p(\boldsymbol{w})} \propto q_2(\boldsymbol{w})$$

$p(\boldsymbol{w}|\mathcal{D}_1)$ $q_1(\boldsymbol{w})$

$$p(\boldsymbol{w}|\mathcal{D}_1, ..., \mathcal{D}_K) \propto p(\mathcal{D}_K|\boldsymbol{w})...p(\mathcal{D}_1|\boldsymbol{w})p(\boldsymbol{w})$$



Sites are through gradients



- How can we approximate $p(D_K|w) \approx t_K(w)$?
  - Look at optimal: we match **gradients**
- Consider (generalised) linear model, MAP solution:

Gradients sum over k:

$$w^* = \underset{w}{\operatorname{argmin}} \frac{1}{2}\|w\|^2 - \sum_k \log p(D_k|w)$$

$$0 = w^* - \sum_k -\nabla \ell_k(w^*)$$

$$w^* = \sum_k -\nabla \ell_k(w^*)$$ Match to each t_k

Match in gradient-space:

$$-\nabla \ell_k(w^*) = \nabla \log t_k(w)\Big|_{w=w^*}$$

$$-w^T \nabla \ell_k(w^*) = \log t_k(w)$$

$$t_k(w) = \exp\left(-w^\top \nabla \ell_k(w^*)\right)$$

- When an **ELBO** (and exponential family): holds, even **during training**:

$$t_k(w) = \exp\left(-\boldsymbol{T}(w)^\top \nabla \mathcal{L}_k(w^*)\right)$$

Bayesian benefit



- Approximate each likelihood with a distribution, $t_k(\boldsymbol{w}) \approx p(\mathcal{D}_k|\boldsymbol{w})$

- What form should $t_k(w)$ take?
  - Point mass vs Gaussian (mean and **variance**)
  - Uncertainties in output-space are useful



Continual learning vs Federated learning

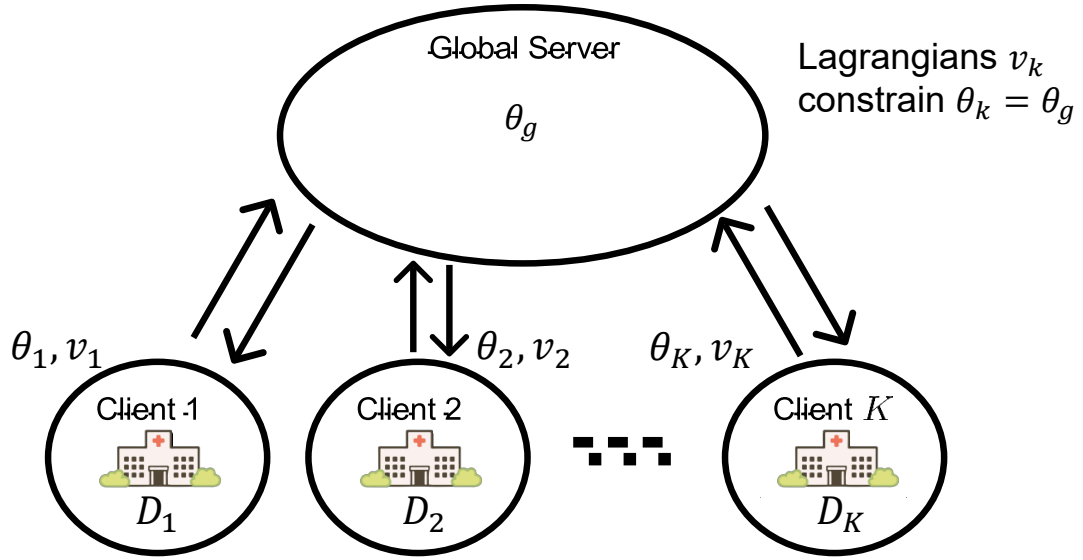


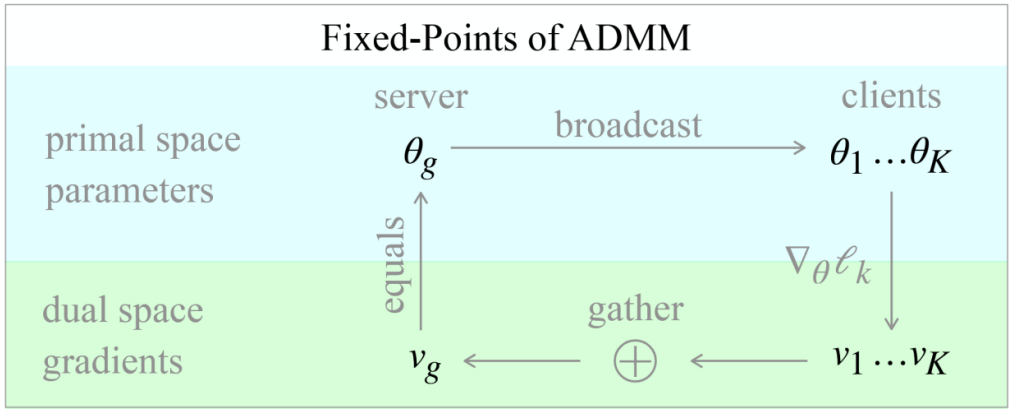
- Continual learning: **need to commit** to an approximation $t_k(\boldsymbol{w}) \approx p(\mathcal{D}_k|\boldsymbol{w})$
  - Cannot revisit data, so **want best approximation** possible

- Federated learning: can **iteratively update** all t_k over many communication rounds
  - Need good algorithm to update t_k
  - **Bayesian-ADMM**: generalise ADMM using Bayesian duality

Khan* & Swaroop*, *"Knowledge-adaptation priors"*, NeurIPS 2021
Subarnaduti et al., "*Fast and Slow Variational Continual Learning*", arxiv
Swaroop, Khan and Doshi-Velez, *"Connecting Federated ADMM to Bayes"*, ICLR 2025
Möllenhoff*, Swaroop*, Doshi-Velez and Khan, *"Federated ADMM from Bayesian Duality"*, ICLR 2026

Federated learning: ADMM



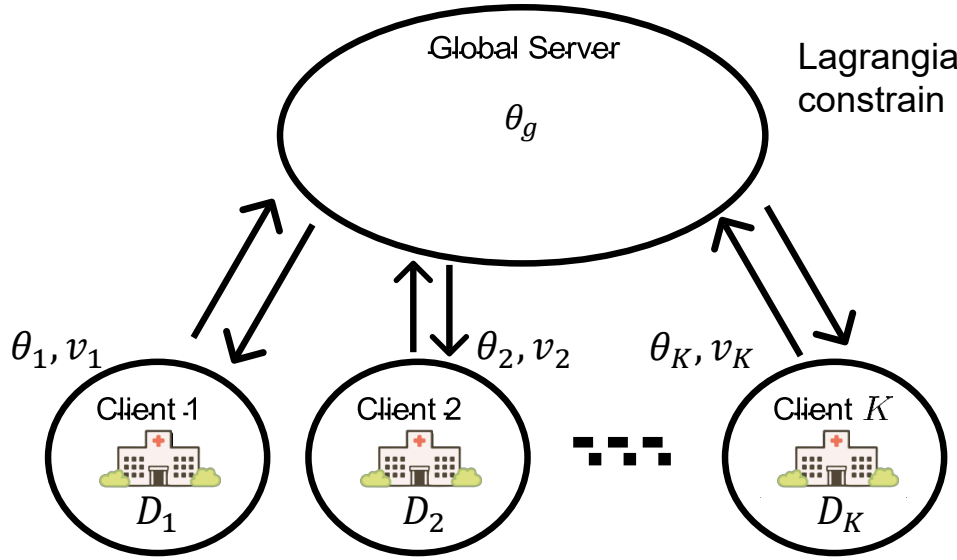


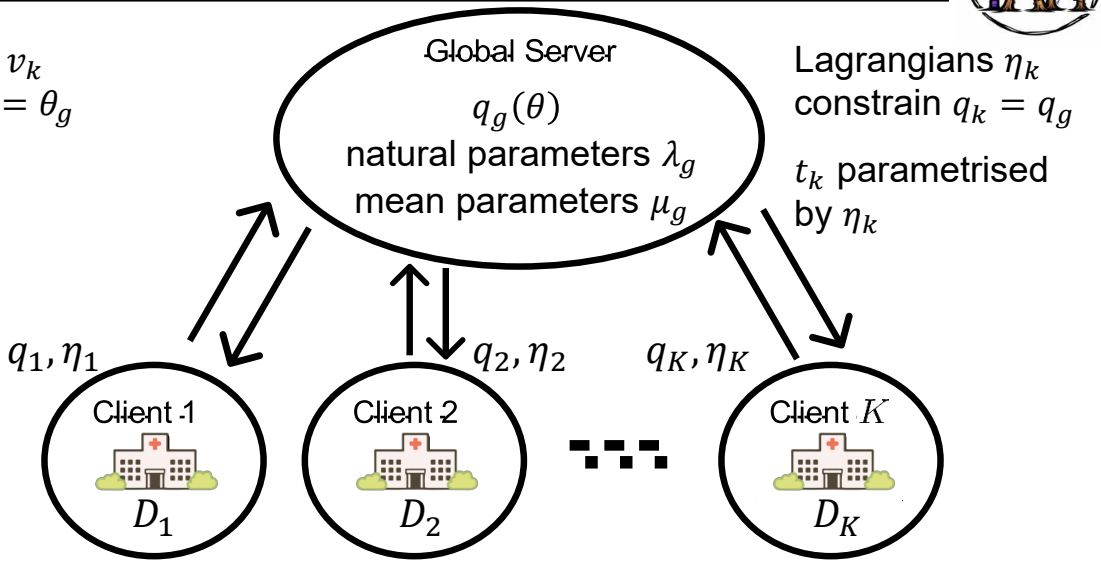
- **Iteratively update t_k** over communication rounds
  - Need good algorithm to update t_k
- ADMM: learn θ_g by solving constrained problem

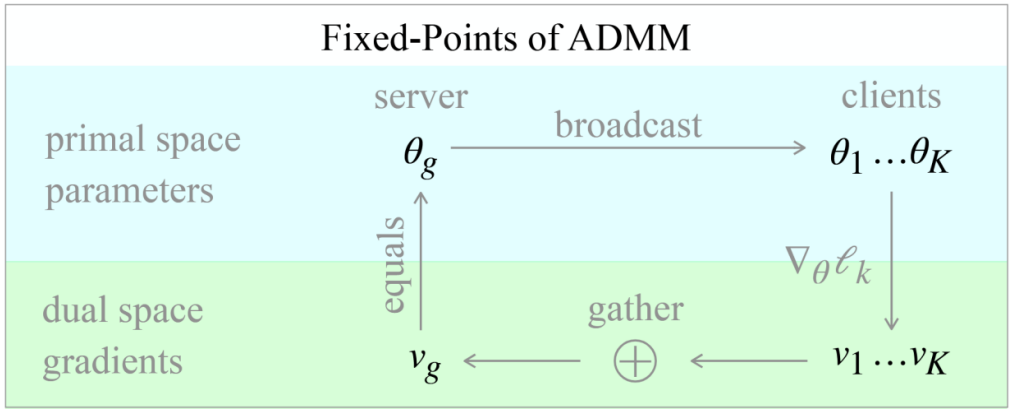
$$\min_{\theta_g, \theta_{1:K}} \sum_{k=1}^{K} \ell_k(\theta_k) + \ell_0(\theta_g),$$

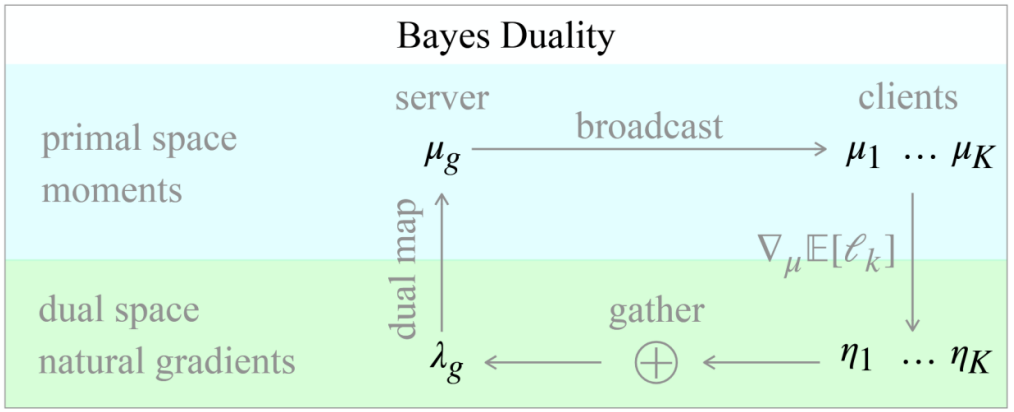
$$\text{such that } \theta_g = \theta_k \text{ for all } k$$

Federated learning: BayesADMM









Recovers ADMM (fixed-cov Gaussian)

Federated learning: BayesADMM



Every iteration, it turns out: $\boldsymbol{\eta}_k = \nabla \mathcal{L}_k(\boldsymbol{\mu}_k) \implies t_k \propto \exp(-\boldsymbol{T}(\boldsymbol{w})^\top \nabla \mathcal{L}_k)$

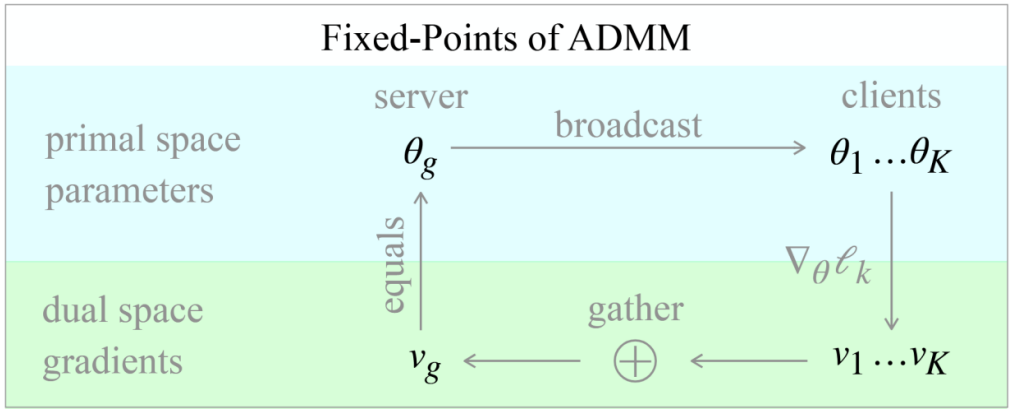
Client updates:

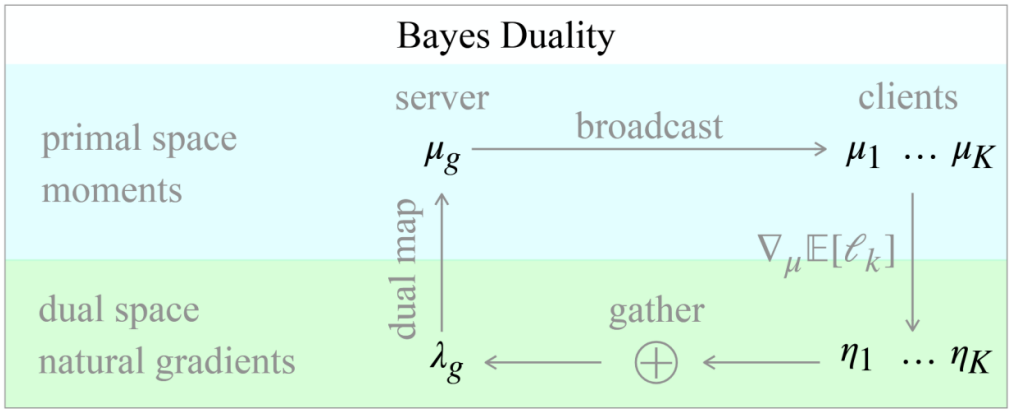
$$\boldsymbol{\theta}_k \leftarrow \arg\min_{\boldsymbol{\theta}_k} \ \ell_k(\boldsymbol{\theta}_k) + \mathbf{v}_k^\top \boldsymbol{\theta}_k + \frac{\rho}{2}\|\boldsymbol{\theta}_k - \boldsymbol{\theta}_g\|^2, \qquad \boldsymbol{\mu}_k \leftarrow \arg\min_{\boldsymbol{\mu}_k} \ \mathcal{L}_k(\boldsymbol{\mu}_k) + \langle \boldsymbol{\eta}_k, \boldsymbol{\mu}_k \rangle + \rho \mathrm{KL}(q_k \,\|\, q_g)$$

$$\mathbf{v}_k \leftarrow \mathbf{v}_k + \rho(\boldsymbol{\theta}_k - \boldsymbol{\theta}_g), \qquad \boldsymbol{\eta}_k \leftarrow \boldsymbol{\eta}_k + \rho(\boldsymbol{\lambda}_k - \boldsymbol{\lambda}_g)$$

Server update:

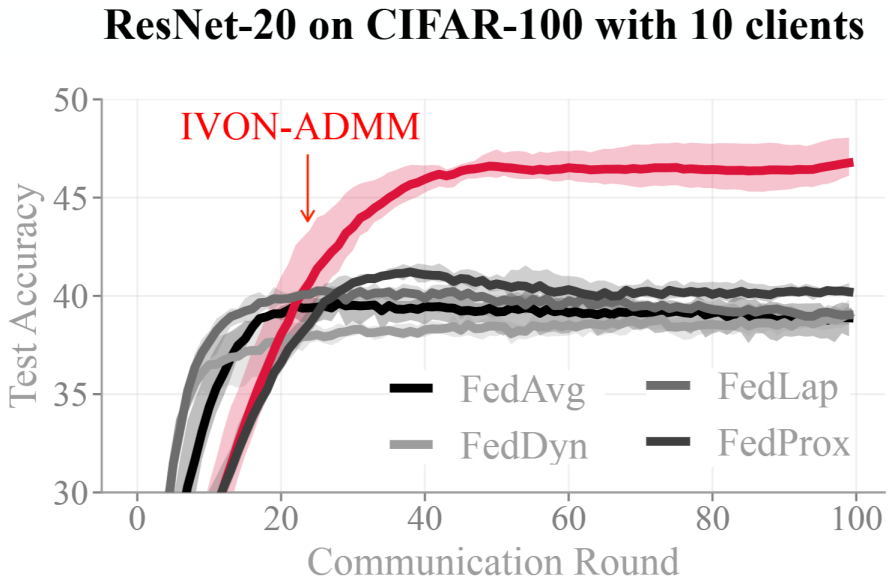
$$\boldsymbol{\theta}_g \leftarrow \arg\min_{\boldsymbol{\theta}_g} \ \ell_0(\boldsymbol{\theta}_g) - \sum_{k=1}^K \mathbf{v}_k^\top \boldsymbol{\theta}_g + \sum_{k=1}^K \frac{\rho}{2}\|\boldsymbol{\theta}_g - \boldsymbol{\theta}_k\|^2. \qquad \boldsymbol{\mu}_g \leftarrow \arg\min_{\boldsymbol{\mu}_g} \ \mathrm{KL}(q_g \,\|\, \pi_0) - \sum_{k=1}^K \langle \boldsymbol{\eta}_k, \boldsymbol{\mu}_g \rangle + \sum_{k=1}^K \rho \mathrm{KL}(q_g \,\|\, q_k).$$

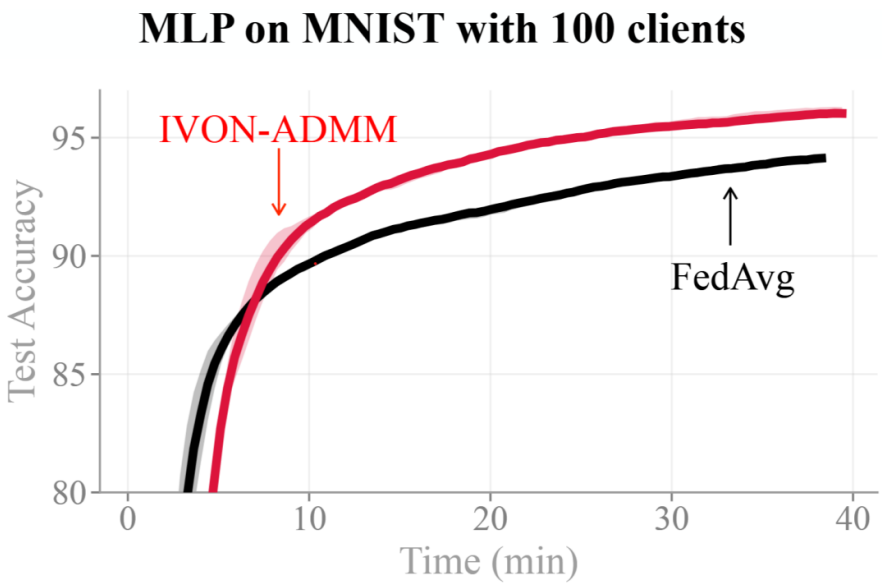




Recovers ADMM (fixed-cov Gaussian)

Federated learning results



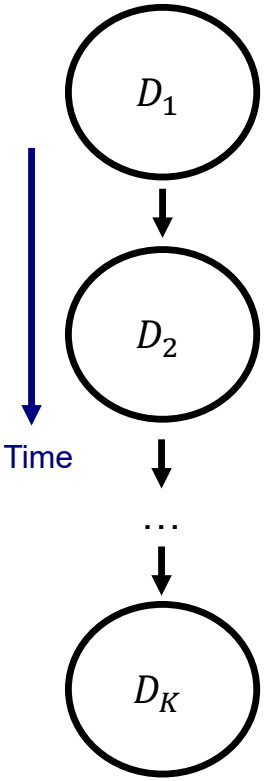


- IVON-ADMM: variational version (using IVON optimiser)

- Higher **accuracy** and computationally **cheap**
  - IVON-ADMM has **2x communication cost** per round (mean and variance)
  - Across different benchmarks and settings (see paper for more)

Swaroop, Khan and Doshi-Velez, *"Connecting Federated ADMM to Bayes"*, ICLR 2025
Möllenhoff*, Swaroop*, Doshi-Velez and Khan, *"Federated ADMM from Bayesian Duality"*, ICLR 2026

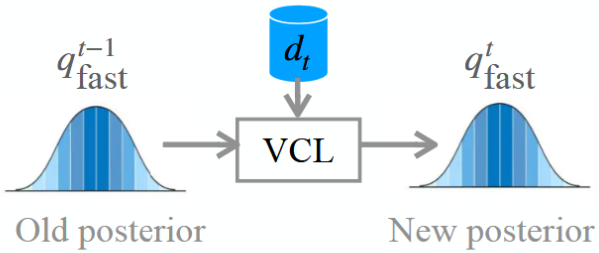
Fast and Slow Variational Continual Learning



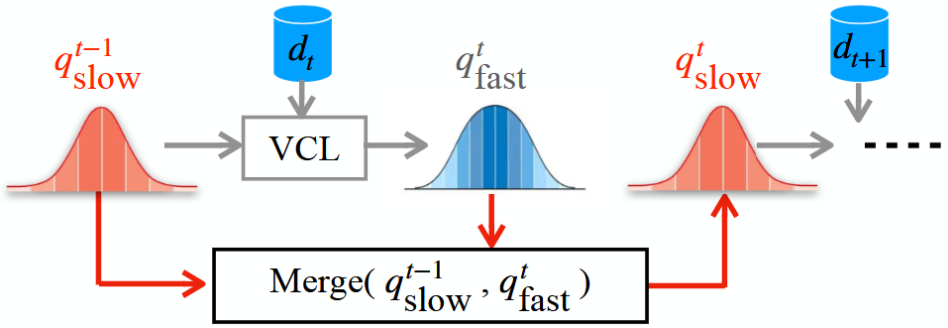
$$q_k(\boldsymbol{w}) \propto t_k(\boldsymbol{w}) q_{k-1}(\boldsymbol{w})$$

- Fast and slow adaptation
  - Fast: quickly adapt to new data (learn new t)
  - Slow: consolidate knowledge

Variational Continual Learning (VCL)

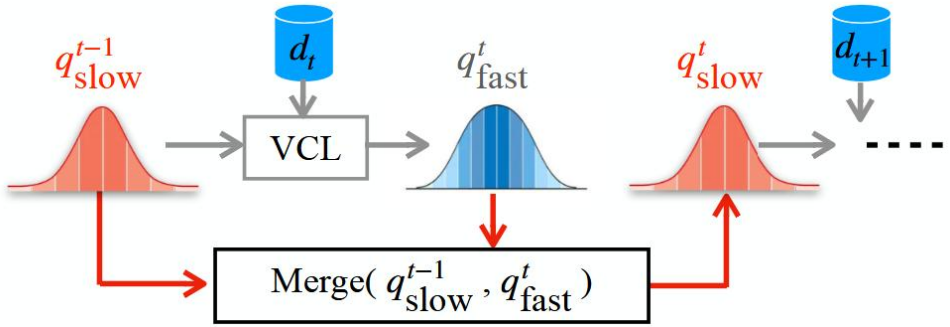


Fast and Slow VCL



Weigh using (weight) uncertainties

Fast and Slow Variational Continual Learning



VCL (fast): $q_t^{\text{fast}}(\boldsymbol{w}) = \text{argmin}_{q \in \mathcal{Q}} \; \mathcal{L}_t(\boldsymbol{w}) + \mathcal{D}_{\mathcal{KL}}(q(\boldsymbol{w}) \,\|\, q_{t-1}^{\text{slow}}(\boldsymbol{w}))$

Merge (slow): $q_t^{\text{slow}} \propto q_{t-1}^{\text{slow}} \left(\frac{q_t^{\text{fast}}}{q_{t-1}^{\text{slow}}} \right)^{\gamma}$

Analysis: $q_t^{\text{fast}} \propto q_{t-1}^{\text{slow}} \exp\left[-\boldsymbol{T}^\top \nabla \mathcal{L}_t \right]$

$t_t(w)$

$\prod_{k=1}^{t} t_k^{\gamma}$

$$q_t^{\text{slow}} \propto q_{t-1}^{\text{slow}} \exp\left[-\gamma \boldsymbol{T}^\top \nabla \mathcal{L}_t \right] \propto p(\boldsymbol{w}) \prod_{k=1}^{t} \exp\left[-\gamma \boldsymbol{T}^\top \nabla \mathcal{L}_k \right]$$

Continual learning results: Domain-incremental



345 classes, 600k images | 27k images | 8 domains, 120k images

Continual learning Methods	DomainNet		CDDB-Hard		CORe50
	A_T (↑)	F_T (↓)	A_T (↑)	F_T (↓)	A_T (↑)
Model: Vision Transformer (ViT-B/16)					
EWC (Kirkpatrick et al., 2017)	47.62±0.9	−1.93	50.59±1.1	−27.62	74.82±1.2
EWC* (Huszár, 2018)	51.01±0.8	−1.79	70.94±0.9	−1.54	85.35±1.2
LwF (Li & Hoiem, 2017)	45.01±0.4	−1.99	60.94±0.3	−16.24	75.35±0.3
L2P (Wang et al., 2022c)	40.15±0.0	−2.25	61.08±0.0	−9.23	78.45±0.0
DualPrompt (Wang et al., 2022b)	43.79±0.03	−2.03	64.80±0.01	−8.74	80.68±0.02
CODA-P (Smith et al., 2023)	54.29±0.20	−1.34	73.15±0.10	−0.99	85.31±0.10
S-iPrompts (Wang et al., 2022a)	50.62±0.05	−2.85	74.51±0.02	−1.30	83.50±0.01
PINA (Wang et al., 2024b)	54.86±0.01	−2.24	77.35±0.01	−0.98	87.26±0.02
CoVON-NoM	**58.96**±0.32	**−1.86**	**80.11**±0.45	**−0.67**	**93.84**±0.34
CoVON	**59.89**±0.08	**−1.61**	**82.24**±0.12	**−0.47**	**95.14**±0.09

Subarnaduti et al., "*Fast and Slow Variational Continual Learning*", arxiv:2606.24007

Improved K-priors: weight and function-space



- Use both **weight and function-space terms** to improve approximation

$$\log p(\mathcal{D}_1|\boldsymbol{w}) + \log p(\boldsymbol{w}) \approx \underbrace{\log t_1(\boldsymbol{w}) + \log p(\boldsymbol{w})}_{\mathcal{K}(\boldsymbol{w};\boldsymbol{w}_1,\mathcal{M})}$$

Use trained model $\boldsymbol{w}_1$, and memory set

$$= \underbrace{\mathbb{D}_{\boldsymbol{w}}(\boldsymbol{w}||\boldsymbol{w}_1)}_{\text{weight-space}} + \underbrace{\mathbb{D}_f(\mathbf{f}(\boldsymbol{w})||\mathbf{f}(\boldsymbol{w}_1))}_{\text{function-space}}$$

- Can exactly match if (i) Generalised Linear Models, (ii) use all data inputs

- Then, relax assumptions and **minimise gradient error**
  - Derive better K-prior equations (Daxberger et al., 2023)
  - Reveals **important datapoints** to store in memory
  - More recently: can look at error directly (no gradients) (Khan, 2025)

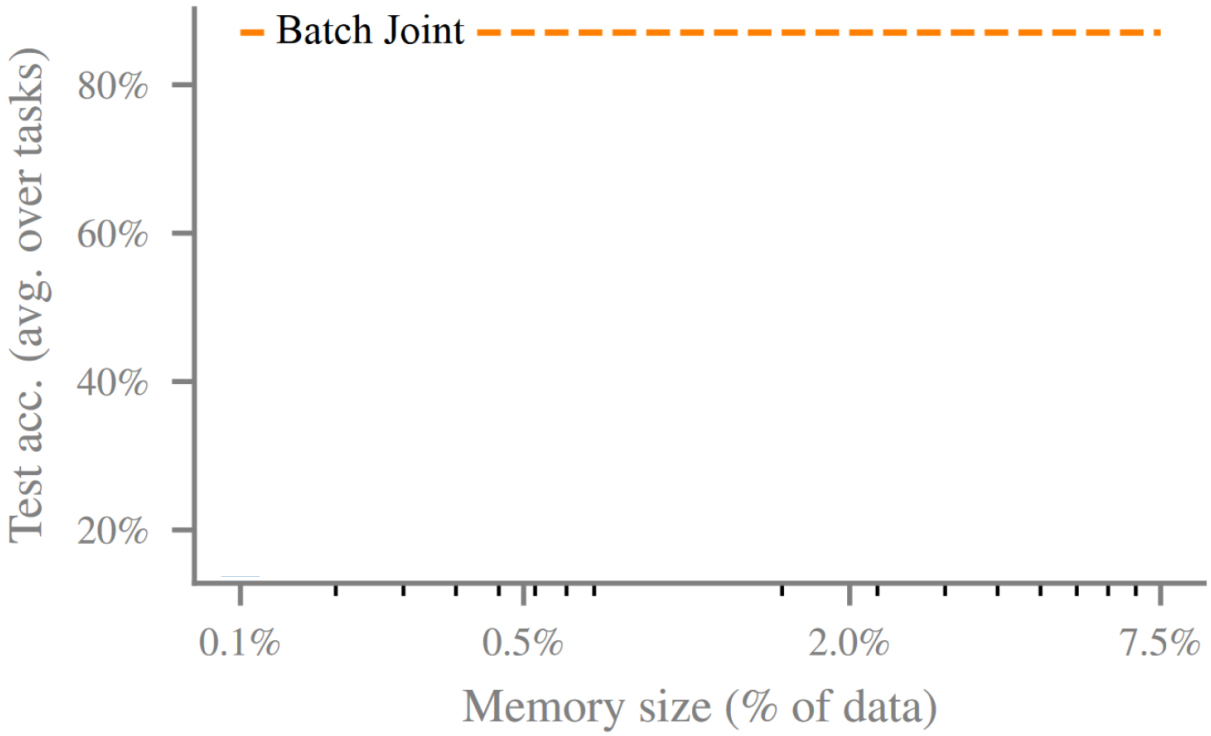
Continual learning results: ImageNet-1000



- Improved K-prior combines
  (i) online Laplace (EWC),
  (ii) replay,
  (iii) function-regularisation

- ImageNet-1000: 10 tasks, with 100 classes and ~120K data points each

- Improved K-prior:
  - Better performance than weight-space
  - Cheaper than Batch



Daxberger, Swaroop et al., *"Improving Continual Learning by Accurate Gradient Reconstructions of the Past"*, TMLR 2023 & CoLLAs 2024

Summary



- Knowledge-adaptation tasks are **related** (federated learning, continual learning, …)
  - **Bayesian approaches can** naturally handle such tasks
  - Useful to have some uncertainty, eg **predictive variance**
- **Federated learning**: combining Bayes + ADMM
- **Continual learning**: fast and slow
  - May not want to do a full update, and should weigh using uncertainty
- Use **better approximations** t_k, using weight-space + function-space
- Future work
  - **Unlearning**: better methods, and can we get guarantees?
  - **Many more tasks** require knowledge adaptation, eg model merging

Acknowledgements





Also thanks to: Subarnaduti Paul, Bai Cong, Yohan Jung, Nico Daheim, Martin Mundt, Erik Daxberger, Kazuki Osawa, Runa Eschenhagen, José Miguel Hernández-Lobato, Richard Turner, Finale Doshi-Velez, Weiwei Pan

Brief summary (repeated here)

- Knowledge-adaptation tasks are related
- **Bayesian approaches** can naturally handle such tasks
- Looked at **federated learning** (Bayesian ADMM) and **continual learning** (fast and slow)

Swaroop, Khan and Doshi-Velez, *"Connecting Federated ADMM to Bayes"*, ICLR 2025
Möllenhoff*, Swaroop*, Doshi-Velez and Khan, *"Federated ADMM from Bayesian Duality"*, ICLR 2026
Subarnaduti et al., "*Fast and Slow Variational Continual Learning*", arxiv:2606.24007
Khan* & Swaroop*, *"Knowledge-adaptation priors"*, NeurIPS 2021
Daxberger et al., *"Improving Continual Learning by Accurate Gradient Reconstructions of the Past"*, TMLR 2023 & CoLLAs 2024