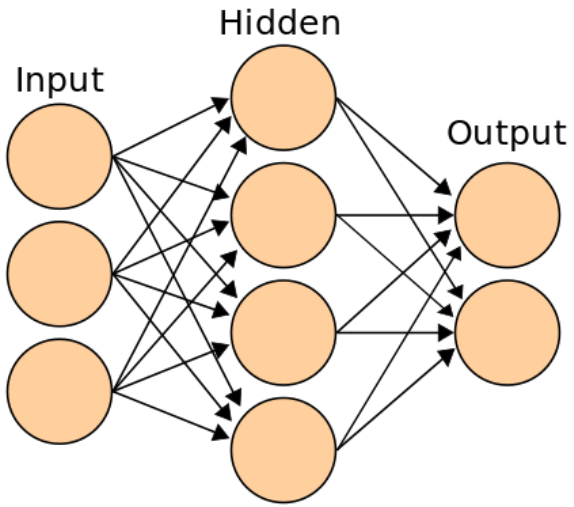
Continual Learning

MACHINE LEARNING ON NON-STATIONARY DATA STREAMS

UNIVERSITY OF PLYMOUTH, MARCH 8, 2024

«Offline» Machine Learning

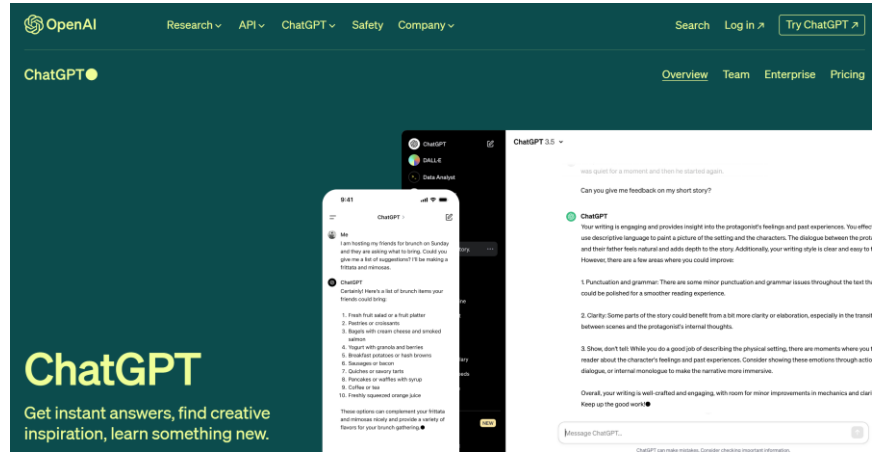
- Model trained on a *fixed, static* dataset

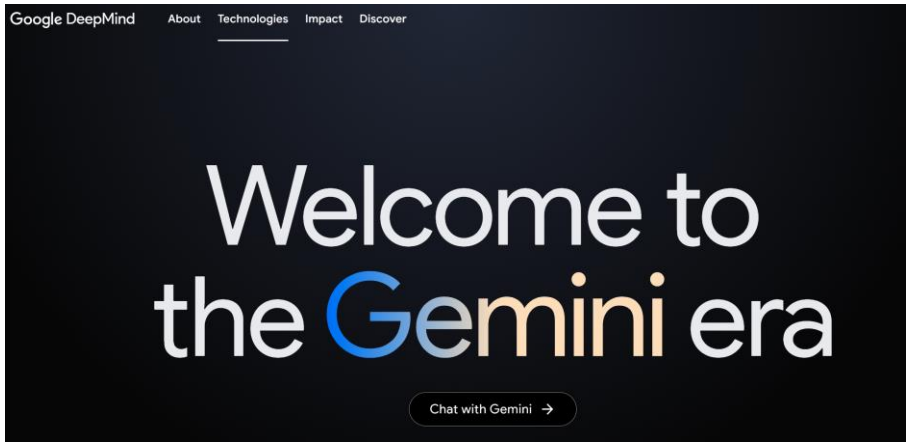


- Generalisation to unseen examples $L^{ML} = \frac{1}{|D_{test}|} \sum_{j=1}^{|D_{test}|} L(f^{ML}(x_j), y_j)$

The perfect model does not exi…

- Large foundation models
  - zero-shot generalisation
  - multimodal
  - can be integrated with external knowledge bases (RAG)
- Is this it? Are we done?





John Donne (1572-1631)

«No model is an island»…

Devotions Upon Emergent Occasions and Seuerall Steps in my Sicknes
Meditation XVII (1624)

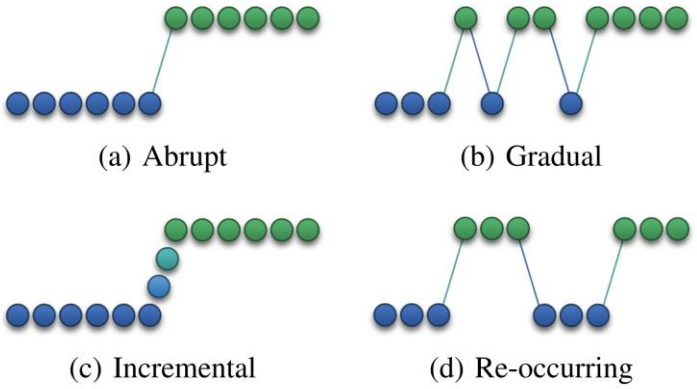
- Offline machine learning models expect examples to be *i.i.d.*
  - Models need to be isolated from changes in the data
- Test distribution may differ from training distribution
  - Test-time domain adaptation
- We may want to adapt the trained model to a different distribution (we don't care about the original distribution)
  - Transfer learning
- The real world is *dynamic!*
  - *Re-training costs (sometimes, even millions of €$£, 100s millions¥)*

Breaking the i.i.d. assumption

Many different nomenclatures to categorise drifts…

$p(y, x) \rightarrow$ joint data distribution

- Concept drift
  - $p_t(y|x)$ changes, $p(x)$ stays the same
- Covariate drift
  - p_t(x) changes, $p(y|x)$ stays the same
- Concept evolution
  - The space spanned by y increases over time
- Abrupt vs. gradual
- Recurring drifts



Dataset drift: real vs. virtual drifts

Virtual drift:

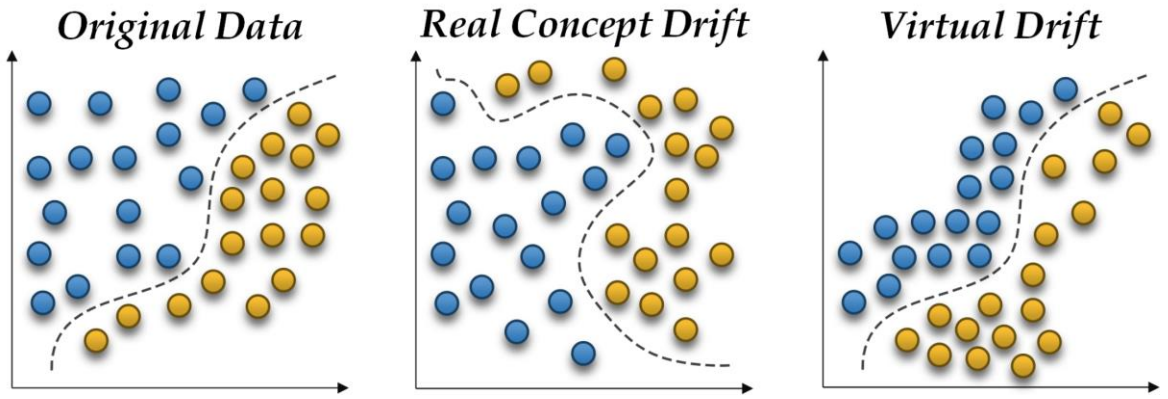
$p(x, y\,|d)$ $\rightarrow$ the drift is induced by a new domain d, but $p(x, y)$ remains the same

Minimize error on the joint distribution

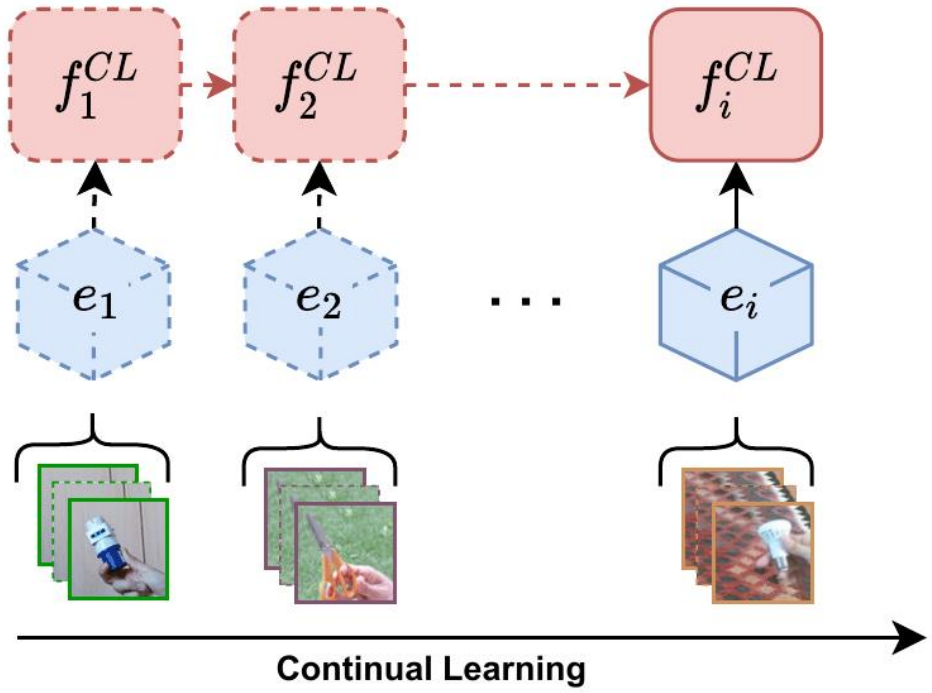
Real drift:

$p_t(x, y) \rightarrow$ the joint distribution changes

Minimize error on future distributions



Continual Learning data stream



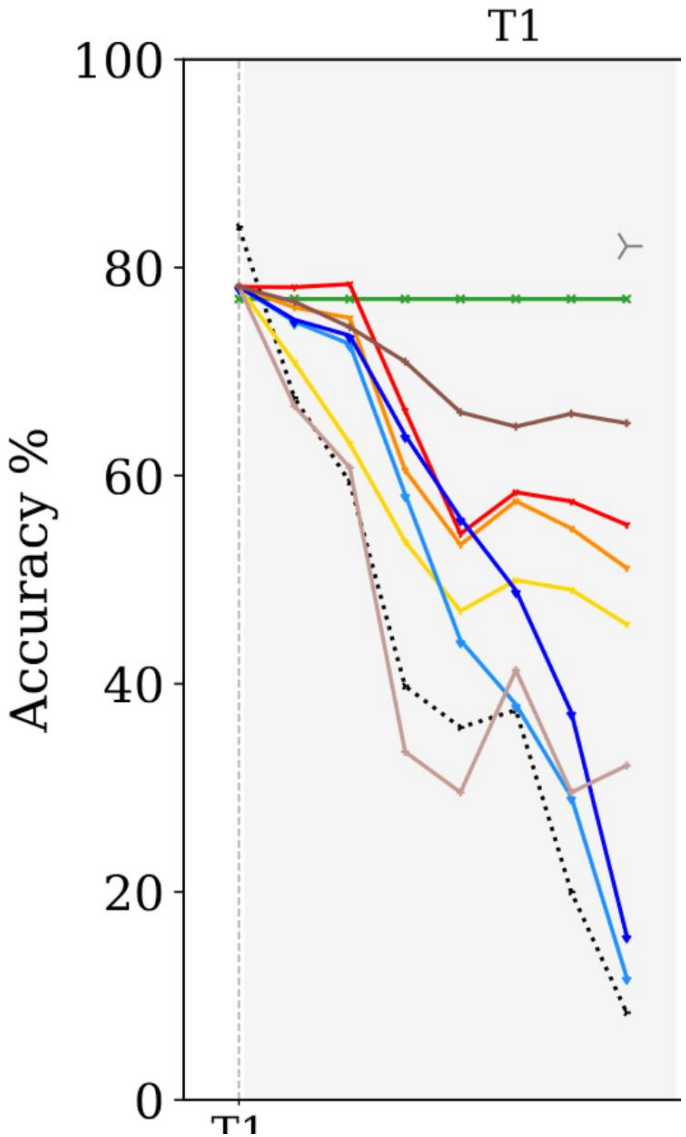
$$L^{CL} = \frac{1}{\sum_{i=1}^{n} |D_{test}^{i}|} \sum_{i=1}^{n} \sum_{j=1}^{|D_{test}^{i}|} L(f^{CL}(x_j^i), y_j^i)$$

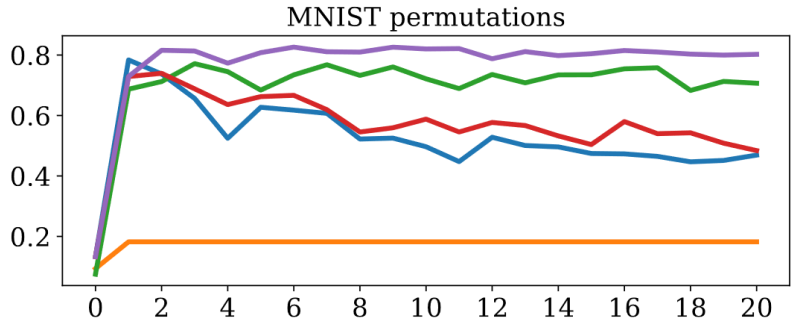
- Drifts between experiences
- No access to previous/future experiences
- Size of each experience may vary
  - Online continual learning = a few examples only
- Computation / memory efficiency
  - «sublinear»

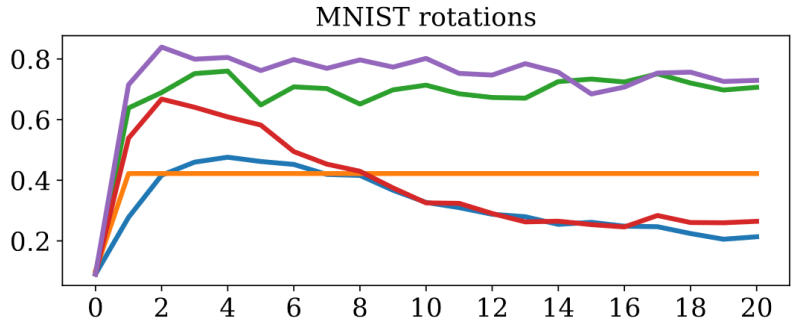
A short disambiguation guide

- Online learning
  - i.i.d. assumption
- Online learning in non-stationary environments, streaming learning
  - Small experiences, like Online CL
  - More rigorous constraints on computational efficiency
  - Temporally-correlated streams
  - Prequential evaluation (test-then-train)
- Multitask learning
  - All tasks available at the same time
- Transfer learning / Domain adaptation
  - Preventing forgetting on the source domain is not crucial

Stability-plasticity dilemma







Matthias DeLange et al., A continual learning survey: Defying forgetting in classification tasks, TPAMI 2021
David Lopez-Paz et al., Gradient Episodic Memory for Continual Learning, NeurIPS 2017

Transfer metrics

- **Forgetting = negative Backward Transfer**

$$a_{k,j} \rightarrow \text{accuracy on } e_j \text{ after training on } e_k$$
$$F_k^j \in [-1, 1] = \max_{l=1,\ldots,k-1} a_{l,j} - a_{k,j} \; \forall j < k$$

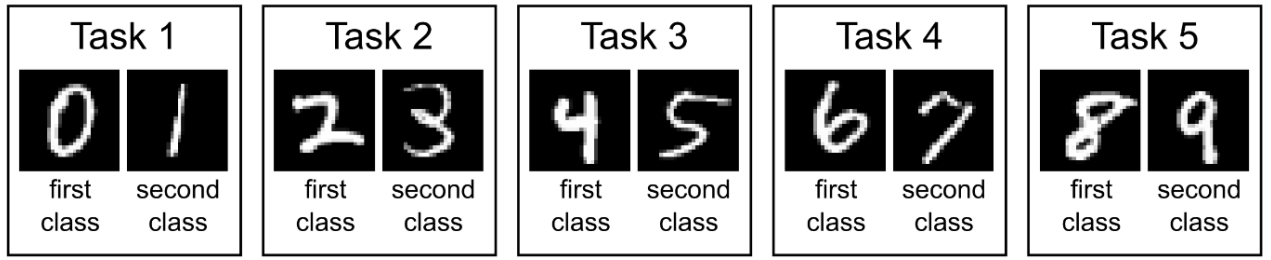
- **Intransigence**

$$a_k^* \rightarrow \text{offline accuracy on } U_{i=1}^{k} e_k$$
$$I_k \in [-1, 1] = a_k^* - a_{k,k}$$

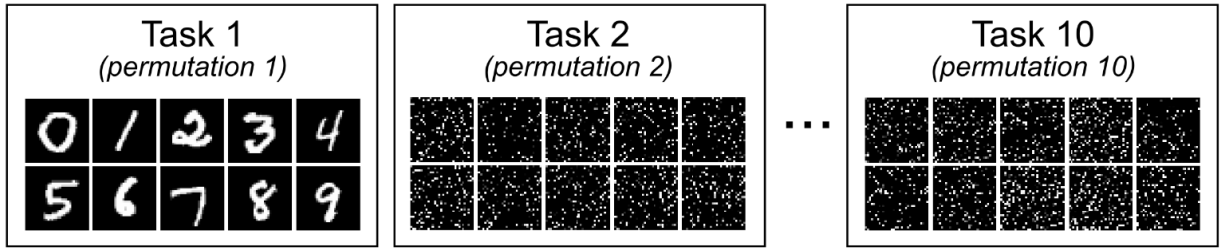
- **Forward Transfer**
  - Negative Intransigence
  - Intransigence, but $a_k^* \rightarrow$ offline accuracy on e_k
  - $FWT_k = a_{k-1,k} - b \rightarrow b$ is the random baseline accuracy

Scenarios, three different ones

- Class-incremental
  - New classes
  - ~ Concept evolution
- Task-incremental
  - Task labels at training/test time
  - ~ Concept drift
- Domain-incremental
  - New examples of previous classes
  - ~ Virtual drift



Split MNIST



Permuted MNIST

- Class-Incremental with Repetition, Blurred boundaries…
  - ~ Recurring drifts, gradual drifts

How to learn continuously?

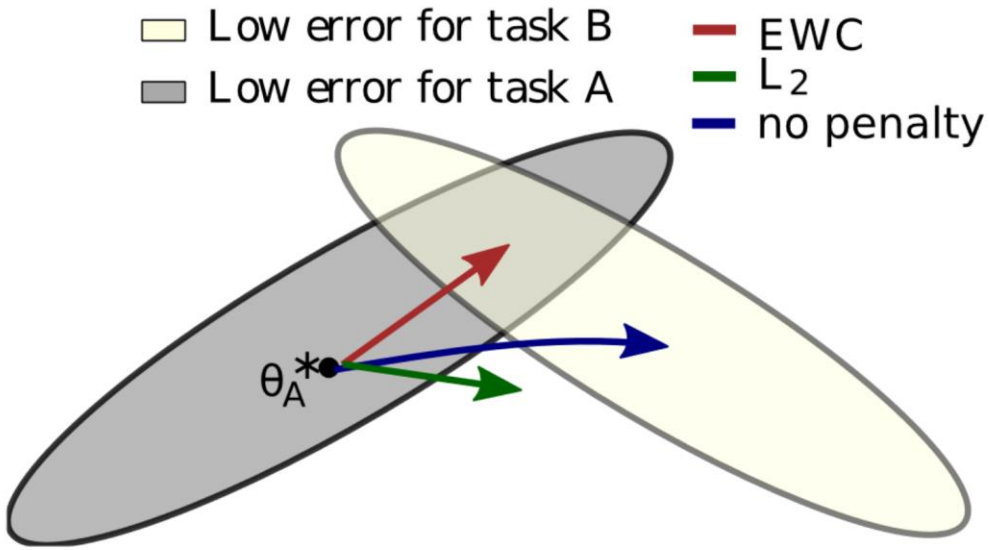
Deep Continual Learning:

- Regularisation strategies
  - Change training loss to improve stability
- Replay strategies
  - Rehearse previous knowledge through an external memory buffer
- Architectural strategies
  - Umbrella term for strategies that dynamically operate on the model architecture
- Hybrid strategies
  - Best of all worlds – also interesting in practice

Importance-based regularisation strategies

Elastic Weight Consolidation (++, Online), Synaptic Intelligence, Rwalk…

- Training loss at experience t
  - $L^{EWC} = CE(\hat{y}, y) + \lambda \sum_{k=1}^{t} \Omega_{\theta_k} (\theta_t - \theta_k)^2$
- Importance is computed at the end of each experience k (averaged over examples)
  - $\Omega_{\theta_k} = E_{e_k}[(\nabla_{\theta_k} CE)^2]$
- One importance vector only
  - $\Omega_{\theta_k} = (1 - \alpha)\, \Omega_{\theta_k} + \alpha\, \Omega_{\theta_{k-1}}$
- They do not work in class-incremental scenarios!



Distillation-based regularisation strategies

Learning without Forgetting, Dark Experience Replay…

- Knowledge distillation loss between a teacher and a student
  - $\boldsymbol{L^{LWF}} = CE(\hat{y}, y) + \lambda\, KL(f_{\theta^T} \,||\, f_{\theta^S}) = CE(\hat{y}, y) + \lambda \sum_x f_{\theta^T}(x)\ \log(\frac{f_{\theta^T}(x)}{f_{\theta^S}(x)}) =$
    $= CE(\hat{y}, y) + \lambda\, [\sum_x f_{\theta^T}(x)\ \log\left(f_{\theta^T}(x)\right) - \sum_x f_{\theta^T}(x)\ \log\left(f_{\theta^S}(x)\right)] =$
    $= CE(\hat{y}, y) + \lambda\ \boldsymbol{CE}\left(\boldsymbol{f_{\theta^T}}, \boldsymbol{f_{\theta^S}}\right)$
- Teacher is *fixed,* student is trained
- Teacher = $f_{\theta^{t-1}}$ → model at the end of previous experience
- Student = f_{θ^t} → current model
- Softmax temperature T controls the softness of the output probability distribution

Replay strategies

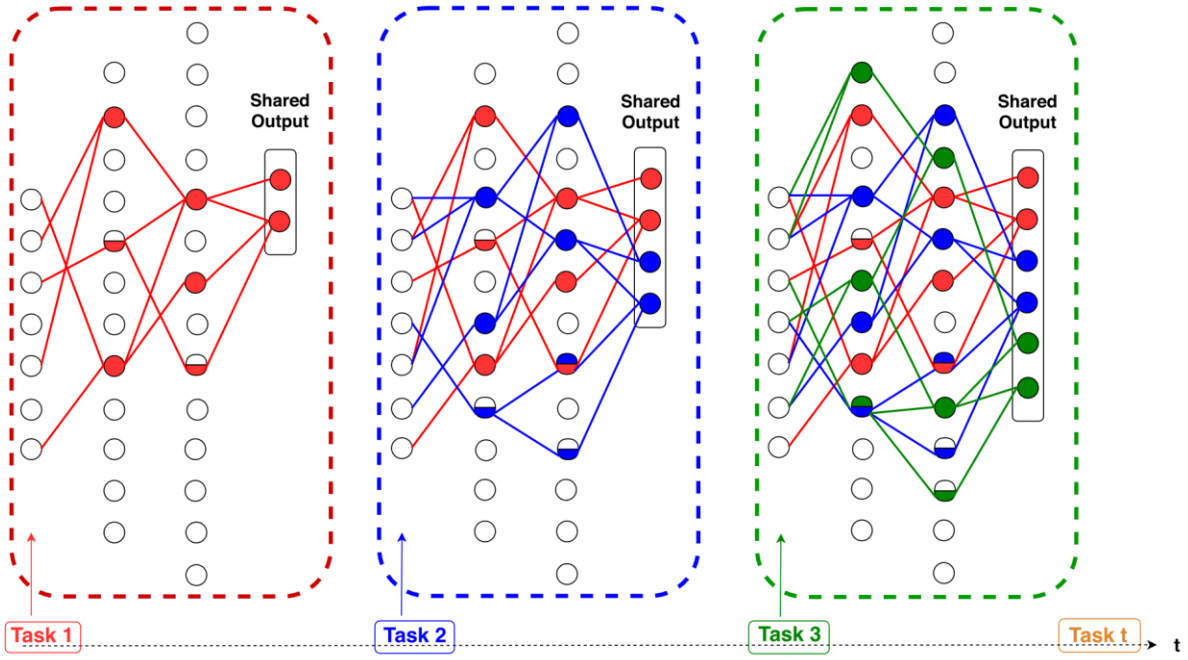
Experience Replay, Gradient Sample Selection, iCaRL…

- External memory buffer (isn't this cheating?)
  - **Memory structure**
    - Fixed-size vs. growing
    - class-balanced, experience-balanced, task-balanced (memory divided in slots)
  - **Insertion/removal policies**
    - reservoir sampling, random… → needs to comply with the memory structure
    - Each slot shrinks over time
  - **Replay policy**
    - concatenate current minibatch with a random sample from the memory
    - Put together current dataset with the entire memory and shuffle (rarely used)
  - Memory updated after each experience
  - Very effective in class-incremental scenarios!

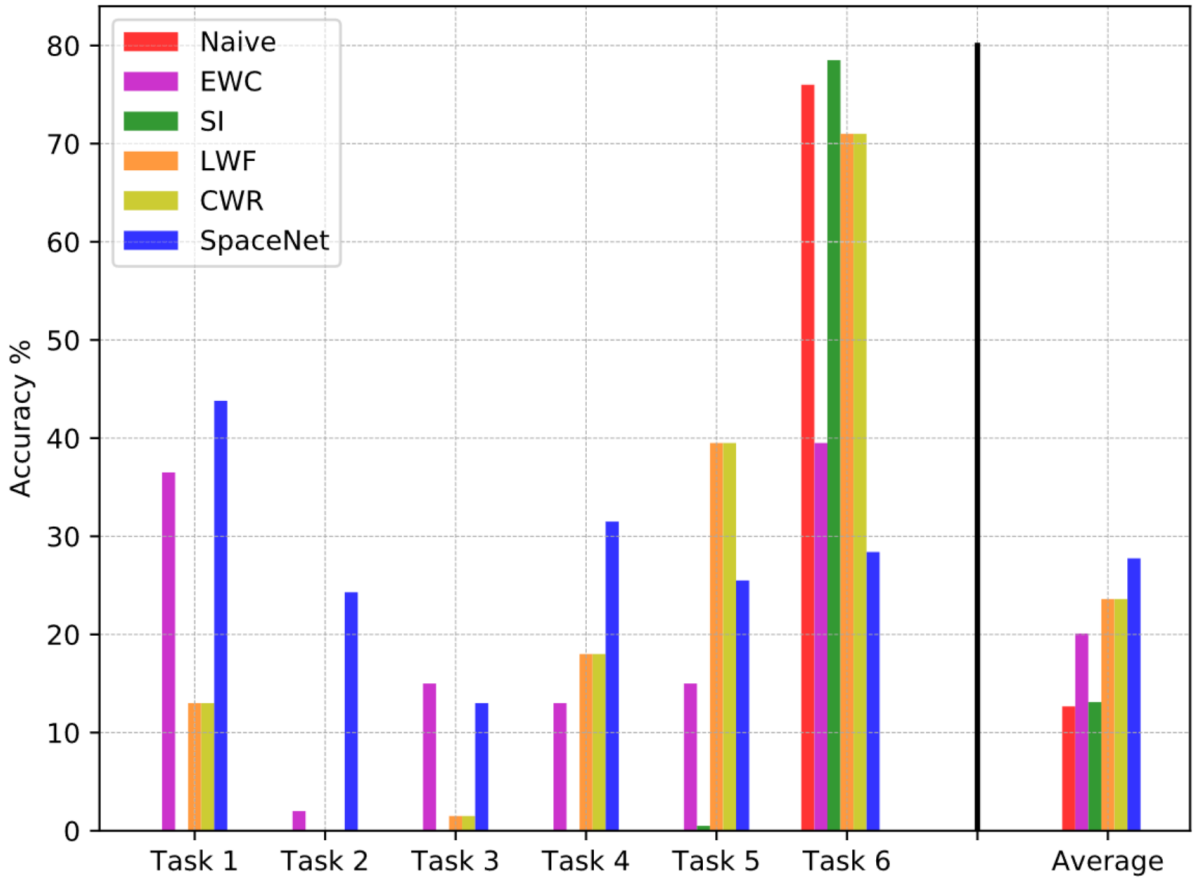
Architectural strategies

Progressive neural networks, PackNet, Supermasks in superposition, *SpaceNet*

- «Forget-free» neural networks
  - what about no task labels?
  - Completely separated experts → no transfer
- Sparsity → expand and compress when needed
  - difficult to learn sparse architectures
- SpaceNet
  - Compatible with class-incremental scenarios
  - Drop phase to drop connections
  - Grow phase to increase network size
  - Sparsity reduce interference
  - Weight importance to sparsify



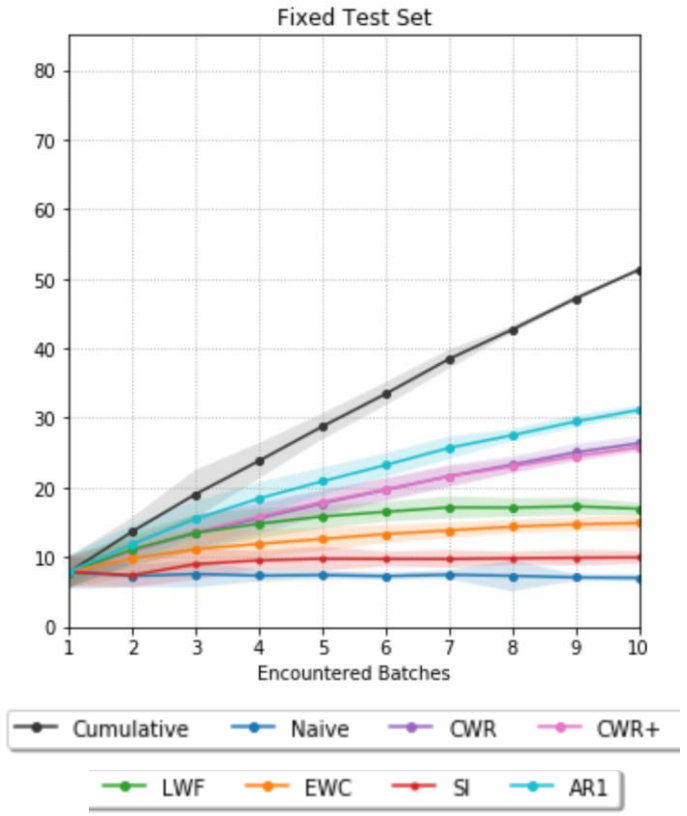
SpaceNet on Split CIFAR100



Hybrid strategies

AR1, ARR, Progress & Compress

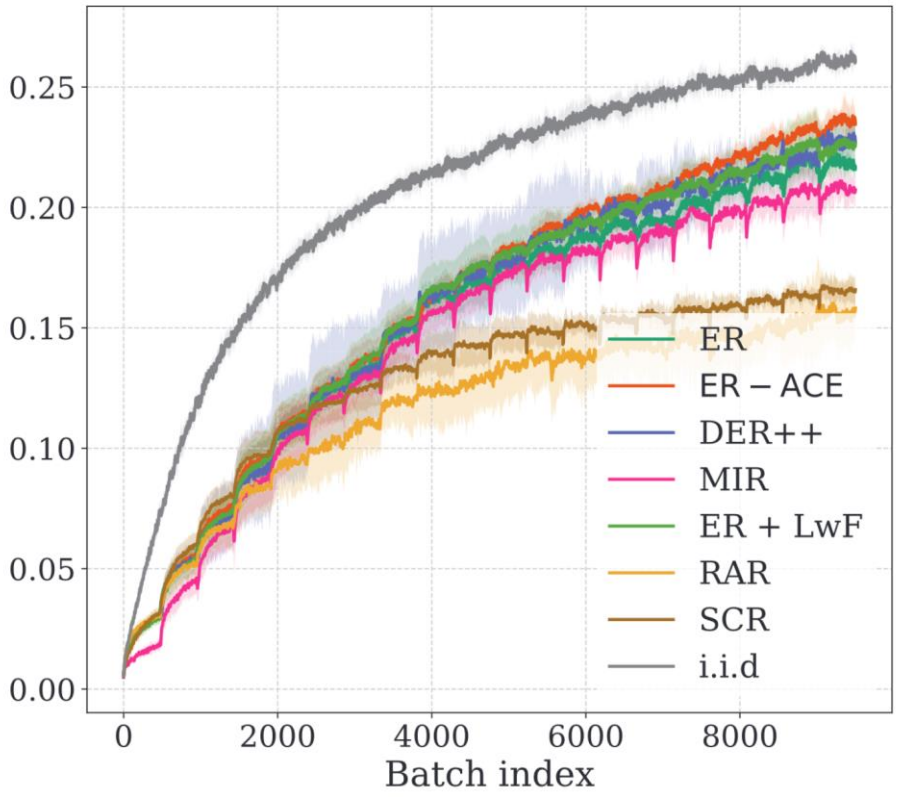
- Combine architectural approach (CWR) with regularization
  - Any regularization works
- Pre-trained model (usually on ImageNet)
  - Freezing feature extractor → limited learning
  - Learning the feature extractor → forgetting
  - Starting from scratch may affect the result
- CWR takes care of the drift at the output layer
  - Two sets of weights, one “fast”, one “slow”



From the lab to the world

No killer application, yet

- Forgetting is not the only issue
- Insufficient generalisation
  - Online continual learning
  - Replay is crucial

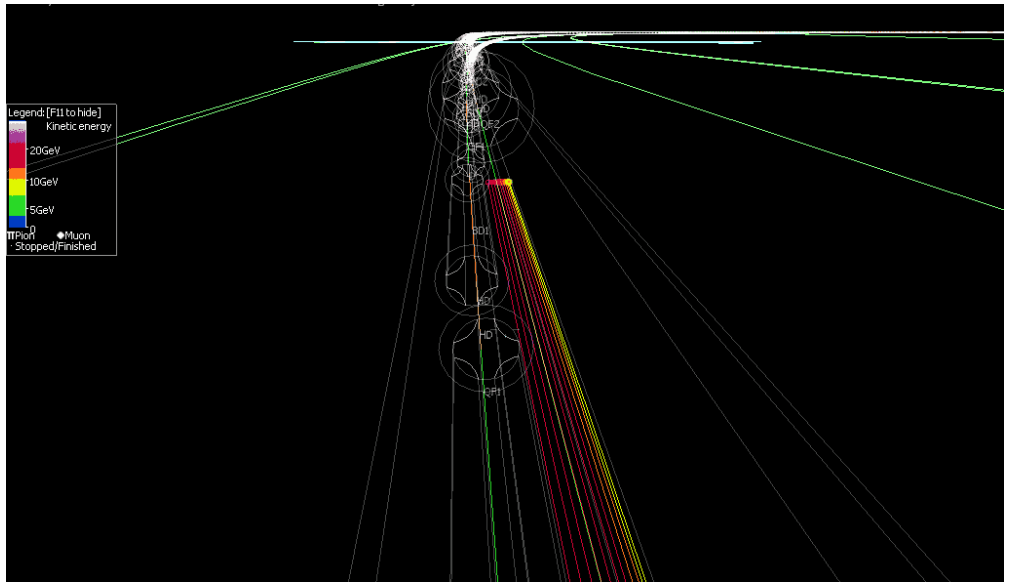


Classic applications…

- Robotics
  - beyond sim2real
- Continual Object Recognition/Detection
- Large Language/Vision Models
  - Mitigate large re-training costs

…and less classic ones

- Space (ESA invitation to tender) and satellite operations
  - Limited computational resources, no Internet access…
- The Electron Ion Collider
  - So much data we can't count (nor store)

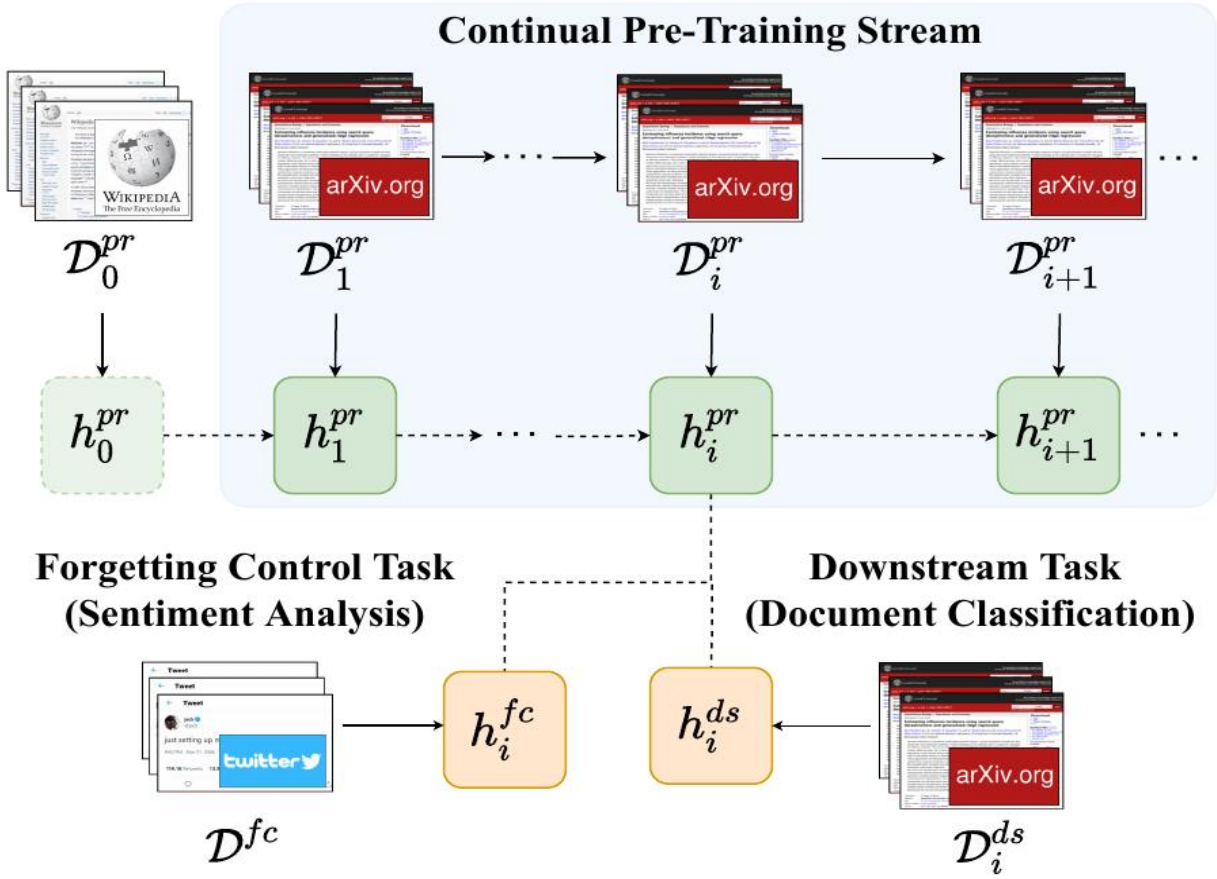


Summing up

- Continual learning is a scientific challenge
  - relatively recent topic
  - a lot of space for improvement
  - **there must be a way (?)**

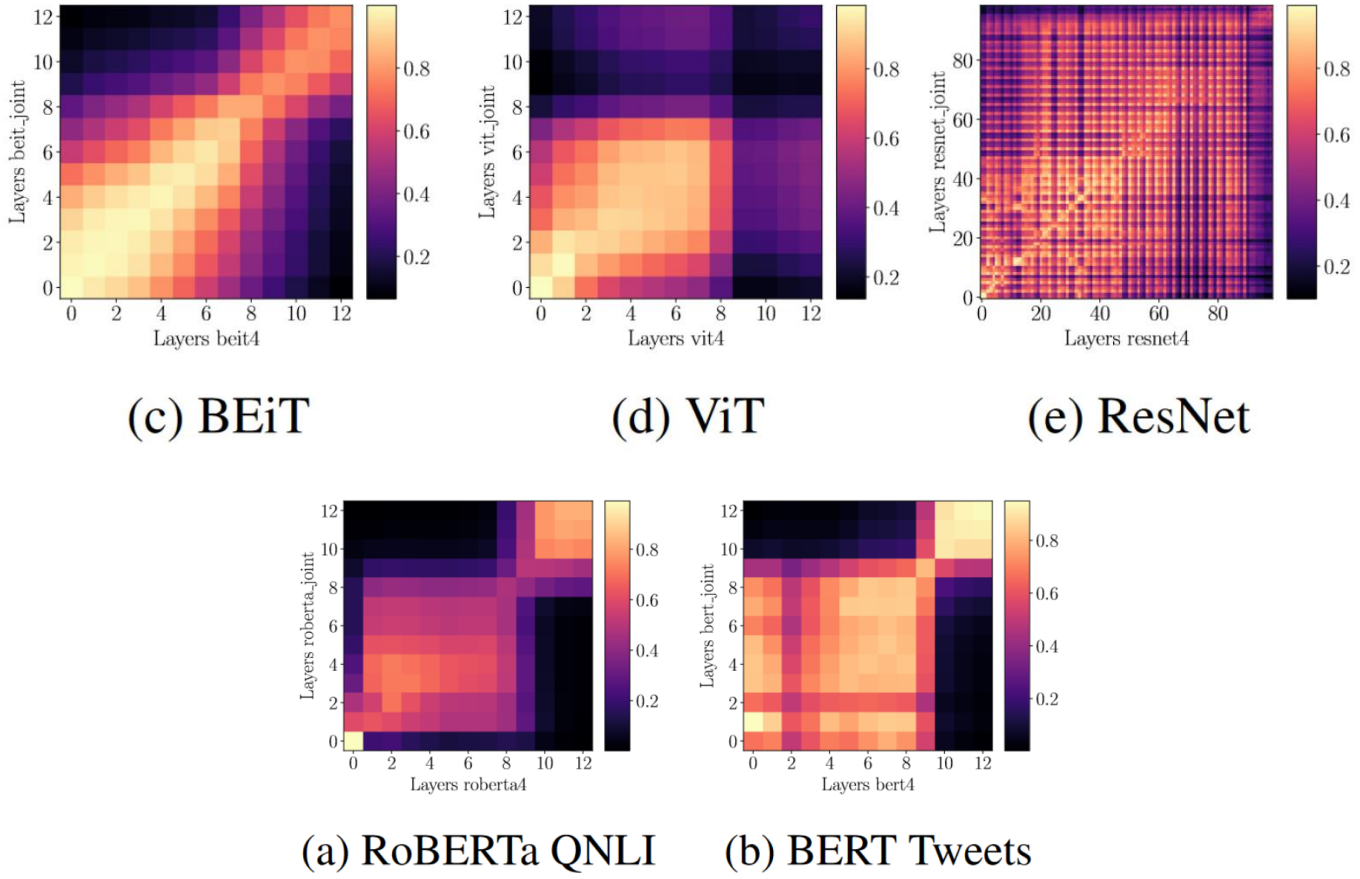
- Continual learning is a real-world challenge
  - Sustainable AI
  - Saving energy, money
  - Privacy-preserving personalisation / on the edge

Advanced topics – Continual Pre-Training



Advanced topics – Continual Pre-Training

Model	Accuracy				
ResNet	94.72				
ViT Base	90.56				
BEiT Base	90.15				
Exp.	e1	e2	e3	e4	e5
ResNet Pr.	89.88	81.29	80.82	77.78	74.35
ViT Pr.	90.29	81.36	81.47	79.71	77.42
BEiT Pr.	88.37	86.45	86.73	87.07	86.46

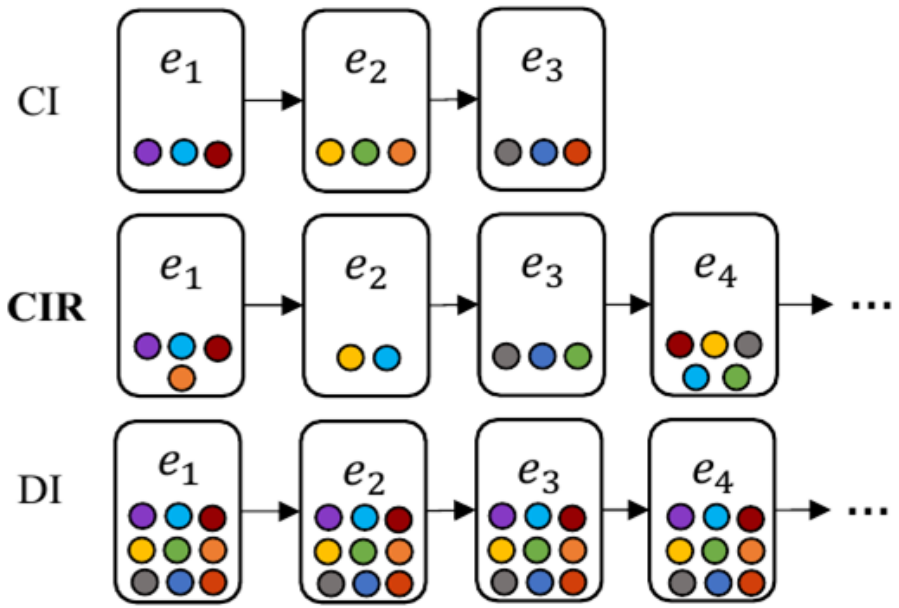


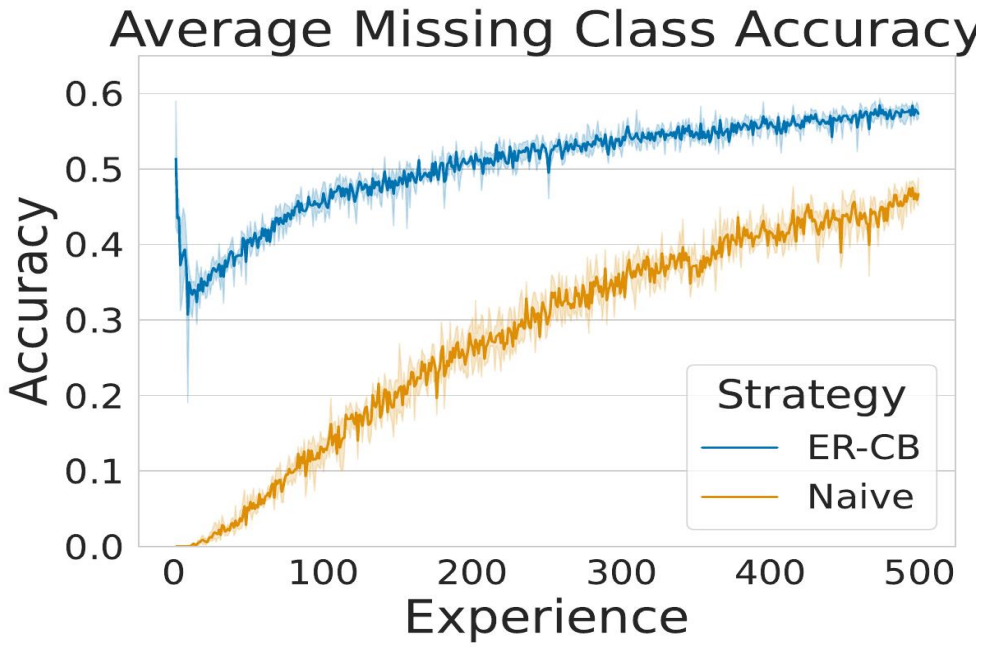
(c) BEiT (d) ViT (e) ResNet

(a) RoBERTa QNLI (b) BERT Tweets

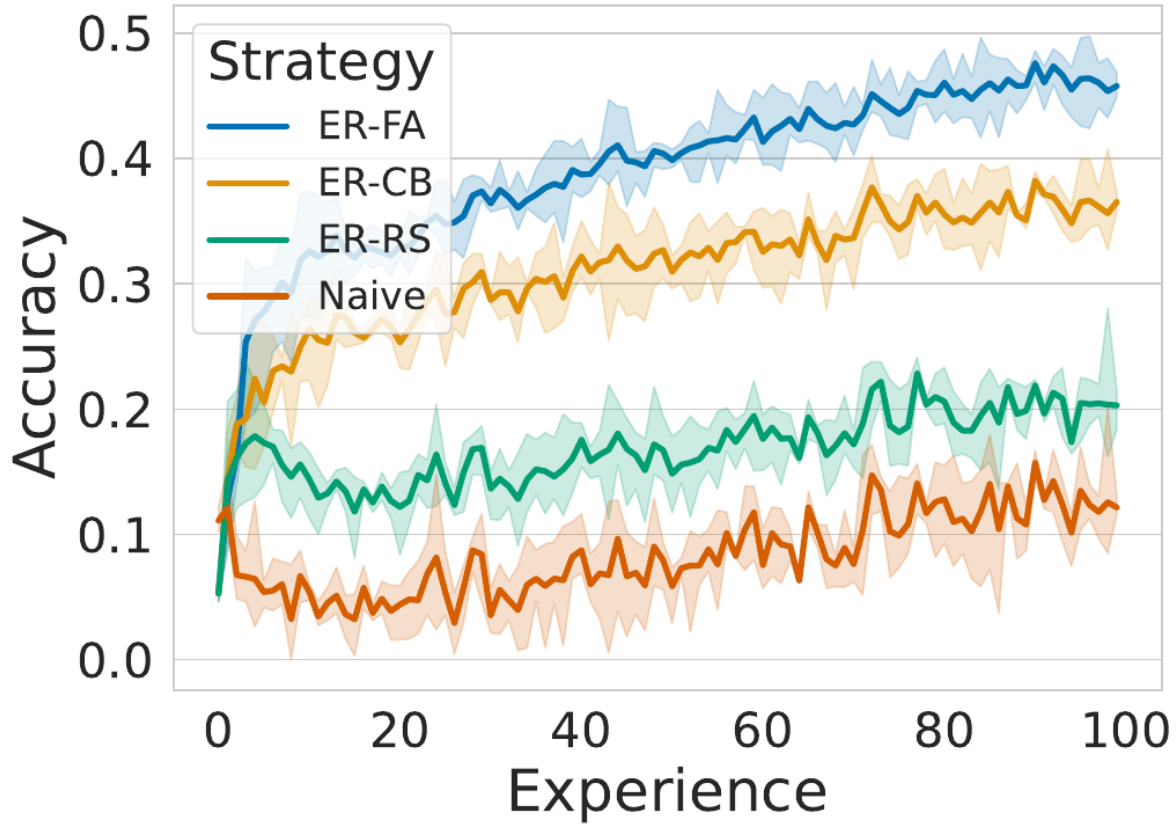
Advanced topics – Class-incremental with repetition

Repetition ≠ Replay





Advanced topics – Class-incremental with repetition



ContinualAI

www.continualai.org

1° ContinualAI Unconference

https://unconf.continualai.org/

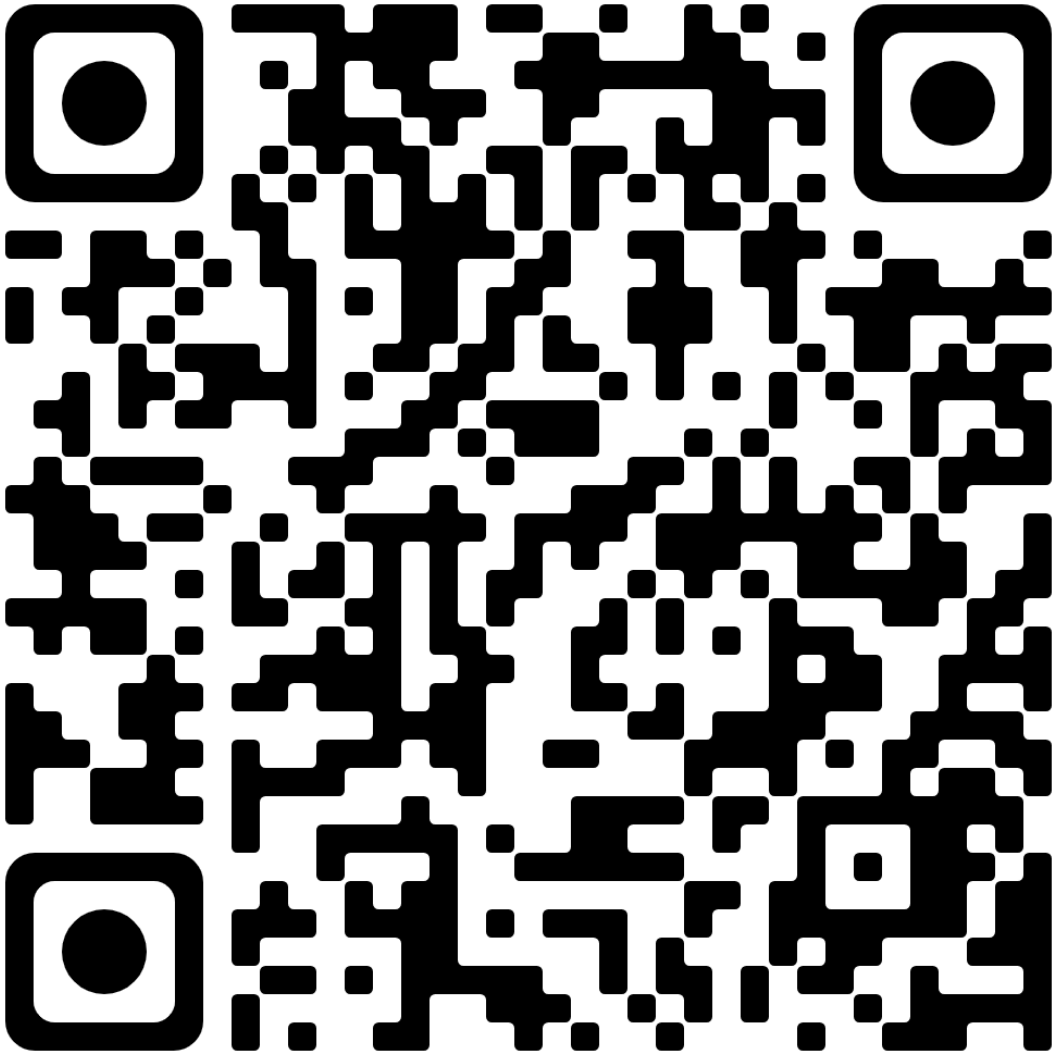
Conference on Lifelong Learning Agents CLVision Workshop

https://lifelong-ml.cc/Conferences/2024

https://sites.google.com/view/clvision2024/

Avalanche

https://avalanche.continualai.org/



Want to run some code?

Colab notebook here: https://bit.ly/3P2bVg1