Three types of incremental learning: a framework for continual learning

Gido van de Ven

(based on work with Tinne Tuytelaars and Andreas Tolias)

–

Sony &inCSL seminar

March 2023

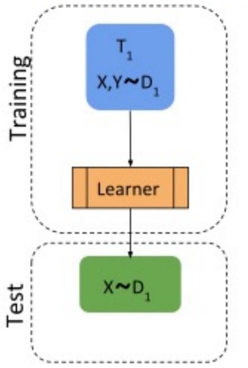
What is continual learning?

- In *classical machine learning*, an algorithm has access to all training data at the same time

- With *continual learning*, two key differences are:
  - the training data arrives incrementally
  - the distribution from which the training data is sampled changes over time

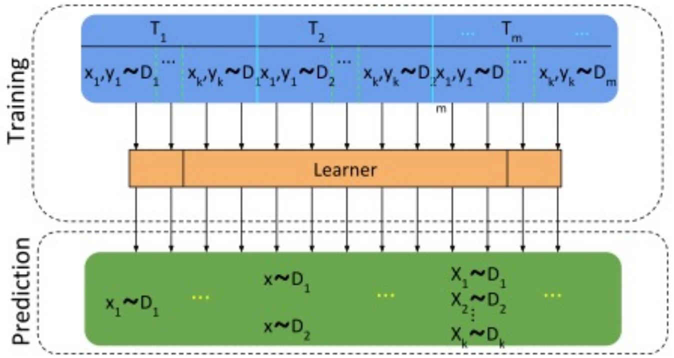
Continual learning in relation to other fields

Standard ML



- One task
- Data available at same time

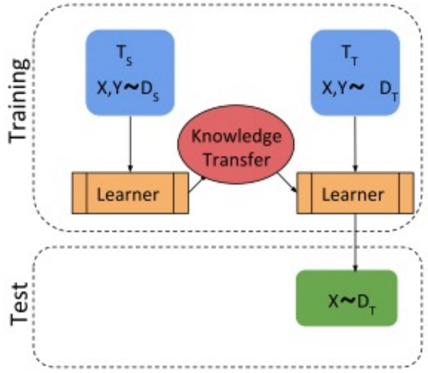
Continual Learning



- Multiple tasks
- Data arrive incrementally
- Goal: all tasks

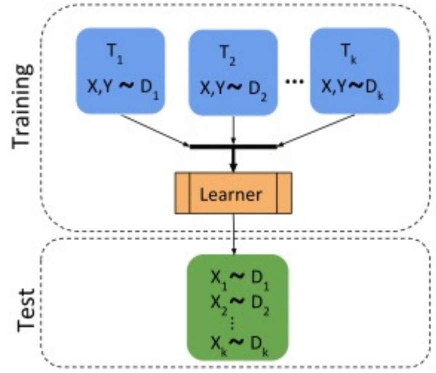
non-stationarity

Transfer Learning



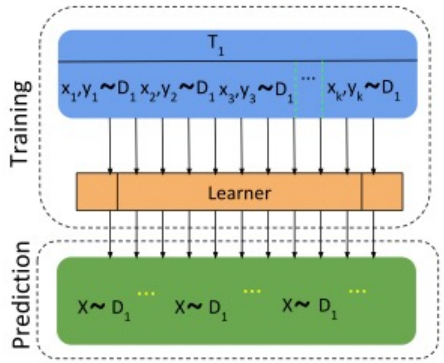
- Multiple tasks
- Data arrive incrementally
- Goal: last task

Multitask Learning



- Multiple tasks
- Data available at same time
- Goal: all tasks

Online Learning

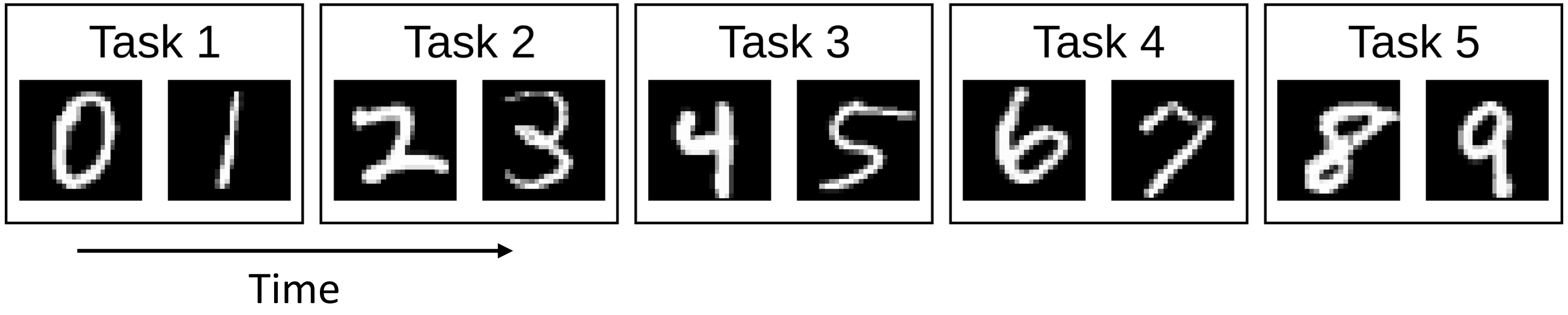


- One task
- Data arrive incrementally

Source: [De Lange et al. (2021, TPAMI)]

The canonical continual learning example: Split MNIST

- MNIST dataset is split in multiple parts/episodes/tasks that must be learned sequentially
- After all tasks have been learned, the model should be good at all tasks
- Typically, no or only a small amount of data from past tasks can be stored

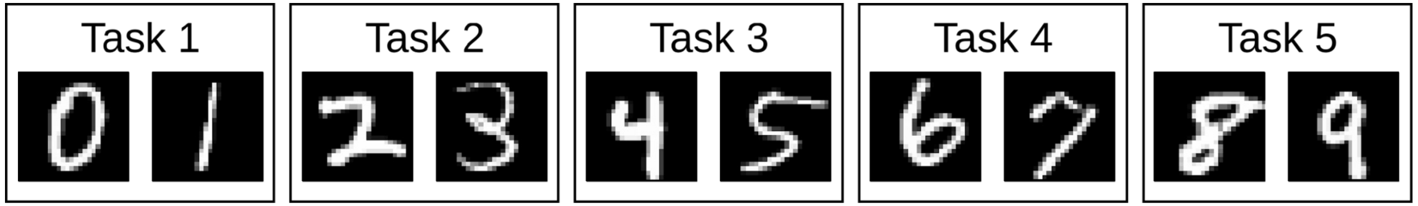


Important problem: ***catastrophic forgetting***

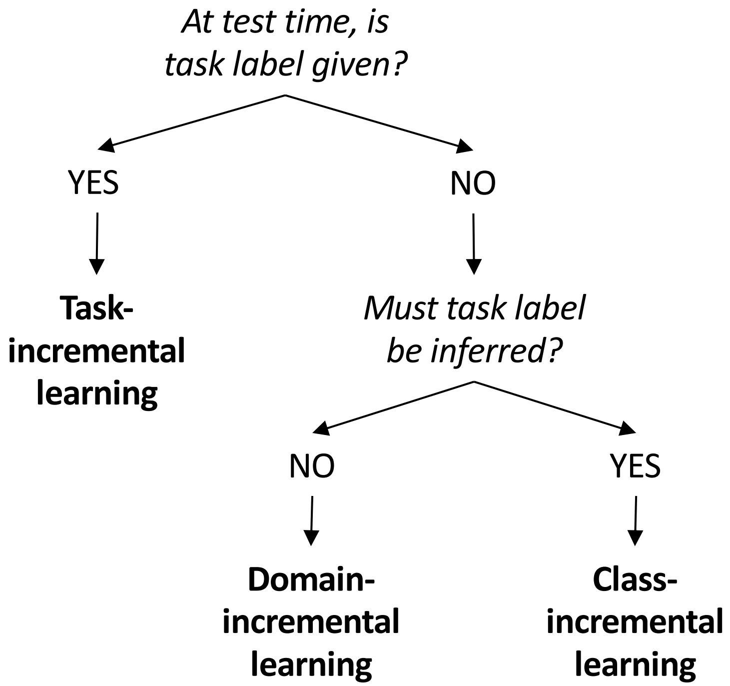
➢ When learning a new task, deep neural networks tend to rapidly forget past tasks

Three continual learning scenarios

Split MNIST:



	Type of choice
Task-incremental	Choice between the two digits of the task
Domain-incremental	Is the digit odd or even?
Class-incremental	Choice between all ten digits

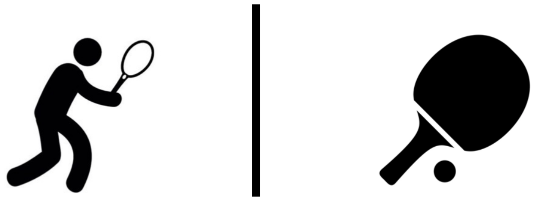


Sources: van de Ven & Tolias (2018, NeurIPS workshop), van de Ven et al. (2022, Nature Machine Intelligence)

Three continual learning scenarios: intuitively

- Task-incremental learning *(Task-IL)*
  - Incrementally learn a set of clearly distinguishable tasks

Important challenge: achieve positive transfer between tasks

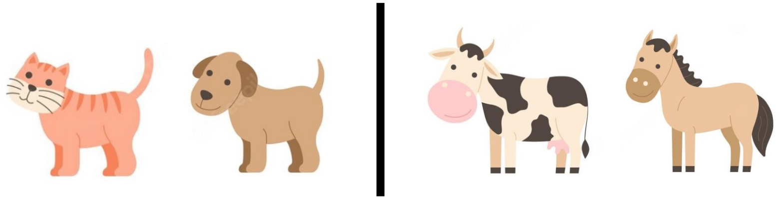


- Domain-incremental learning *(Domain-IL)*
  - Learn the same type of problem in different contexts

Important challenge: alleviate catastrophic forgetting

- Class-incremental learning *(Class-IL)*
  - Incrementally learn a growing number of classes

Important challenge: learn to discriminate between objects not observed together

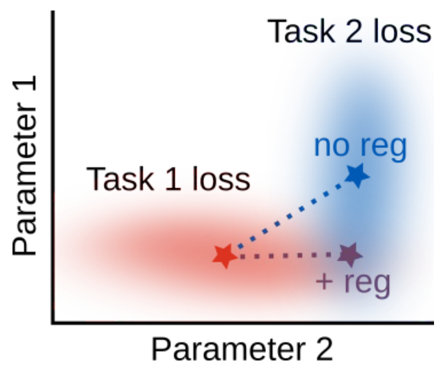


Images designed by Freepik

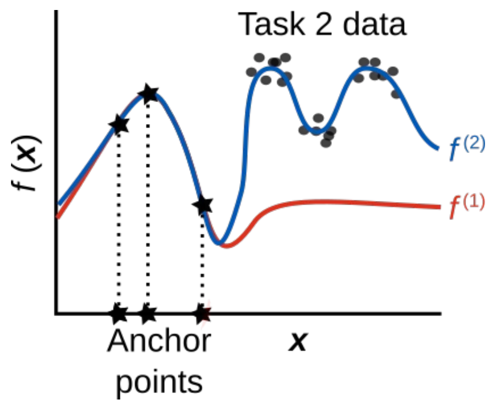
Sources: van de Ven & Tolias (2018, NeurIPS workshop), van de Ven et al. (2022, Nature Machine Intelligence)

Categorizations of continual learning strategies

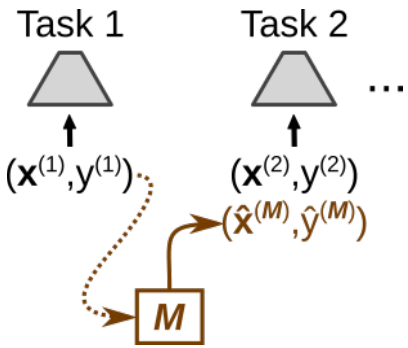
Parameter regularization



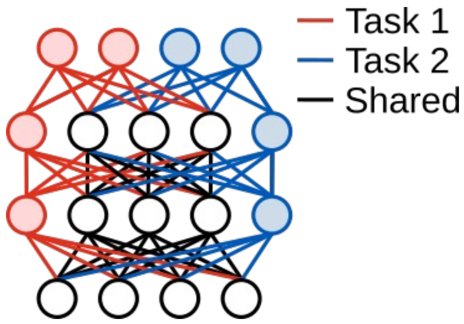
Functional regularization



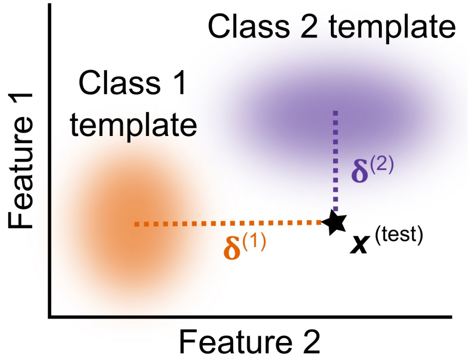
Replay



Context-specific components



Template-based classification



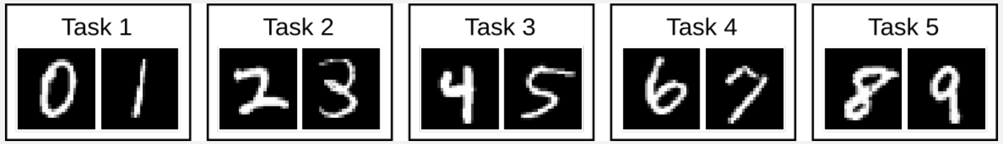
Source: [van de Ven et al. (2022, Nature Machine Intelligence)]

Baselines: finetuning (*lower target*) & joint training (*upper target*)

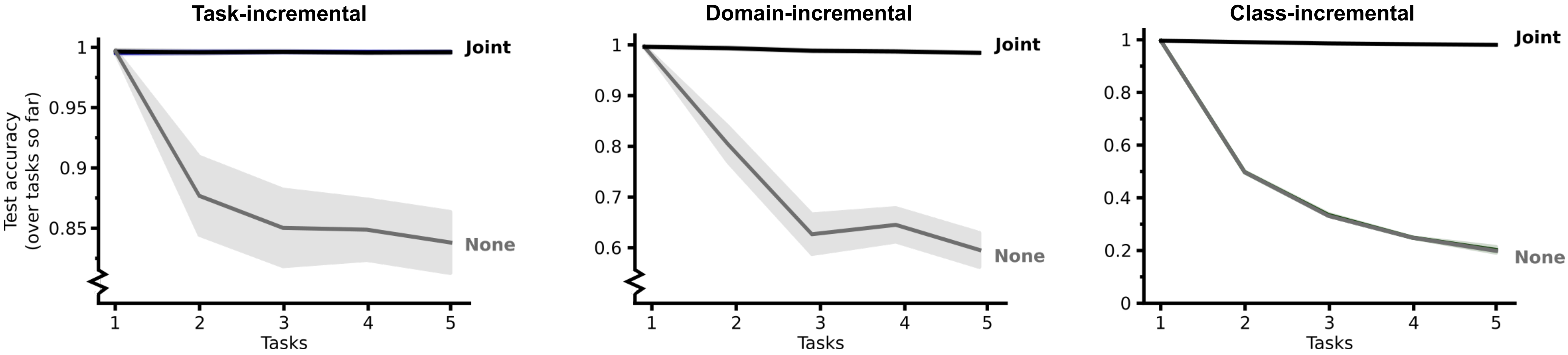
None: Network sequentially trained on each task in the standard way (*lower target*)

Joint: Network trained on all tasks at the same time (*upper target*)

Empirical comparison on Split MNIST according to each scenario



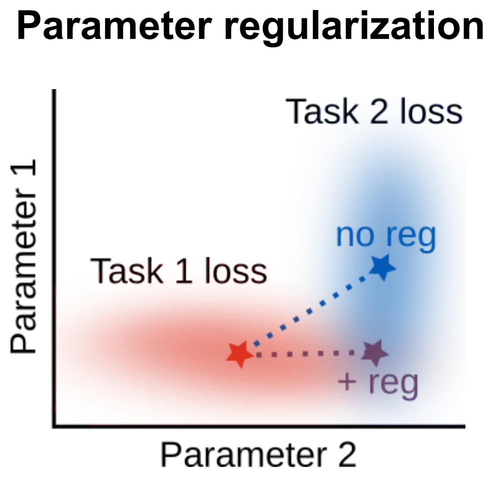
Task-incremental learning	Choice between two digits of same task (*e.g.*, 0 or 1?)
Domain-incremental learning	Is the digit odd or even?
Class-incremental learning	Choice between all ten digits

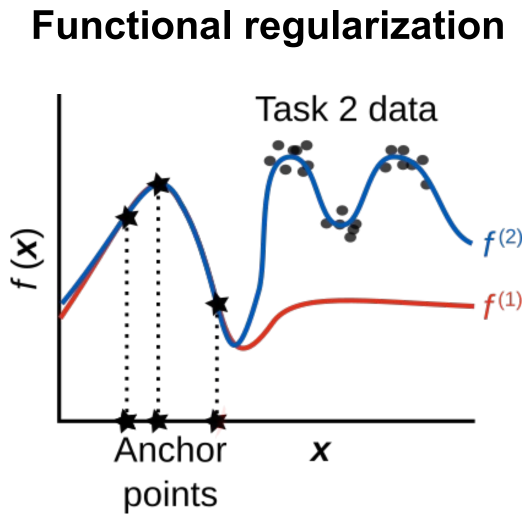


Code for these experiments: https://github.com/GMvandeVen/continual-learning

Regularization

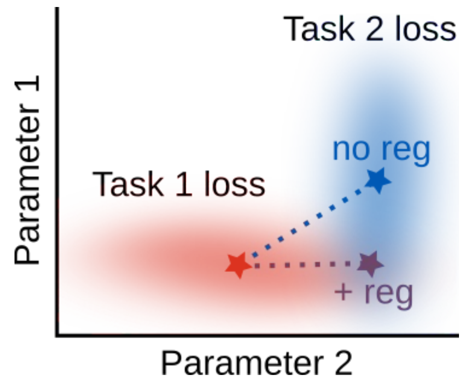
- In continual learning, regularization typically means adding a penalty term to the loss function to **encourage the model to stay close to a previous version of itself**.
- Often, the version relative to which changes are penalized is a copy of the model stored after finishing training on the last task
- Two forms of regularization:





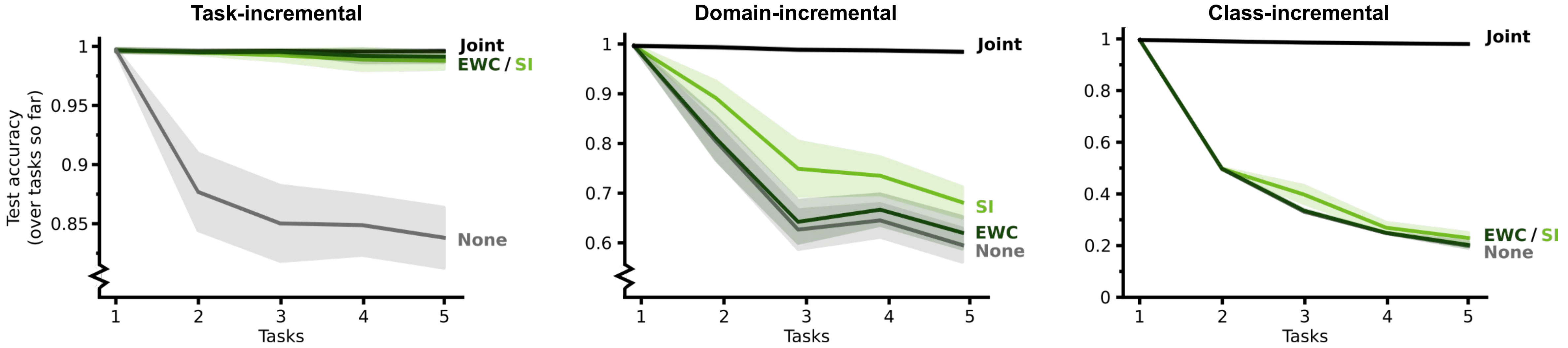
Parameter regularization

- Parameters important for past tasks are encouraged not to change too much when learning a new task
- Can often be interpreted as sequential approximate Bayesian inference on the network's parameters
- Representative methods:
    - Elastic Weight Consolidation [**EWC**] (Kirkpatrick et al., 2017 PNAS)
    - Synaptic Intelligence [**SI**] (Zenke et al., 2017 ICML)



$$\mathcal{L}_{\text{total}} = \mathcal{L} + \|\theta - \theta^*\|_\Sigma$$

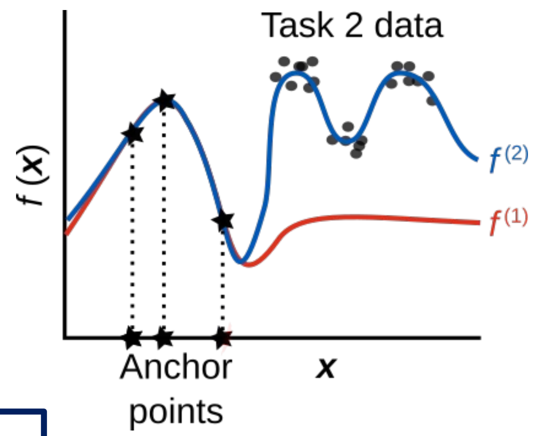
θ^*: parameters relative to which changes are penalized
Σ: estimate of how important parameters are
$\|.\|_\Sigma$: weighted norm



Code for these experiments: https://github.com/GMvandeVen/continual-learning

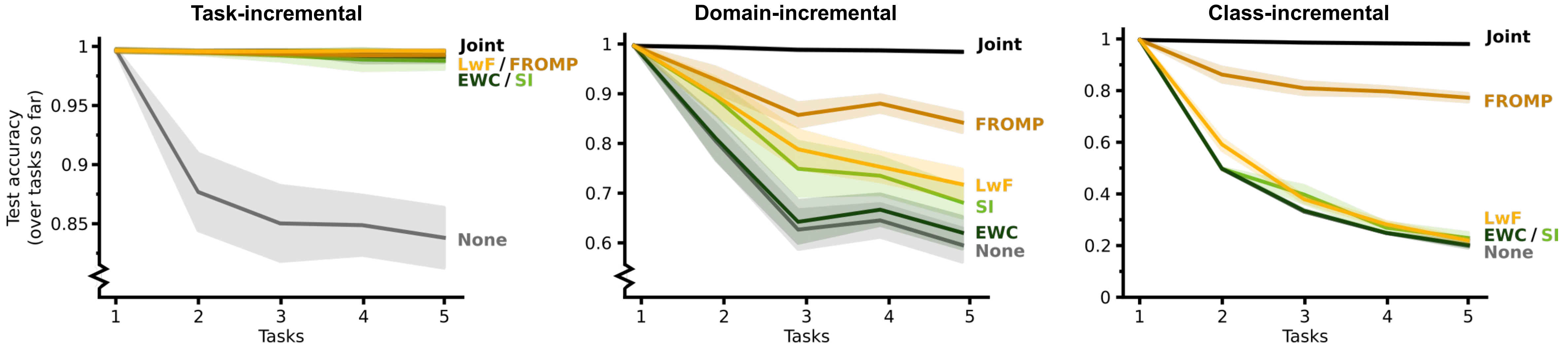
Functional regularization

- The input-output mapping learned previously is encouraged not to change too much at a particular set of inputs (the 'anchor points')
- Also referred to as knowledge distillation
- Representative methods:
  - Learning without Forgetting [**LwF**] (Li & Hoiem, 2017 TPAMI)
  - Functional Regularization Of Memorable Past [**FROMP**] (Pan et al., 2020 NeurIPS)



$$\mathcal{L}_{\text{total}} = \mathcal{L} + \langle f_\theta, f_{\theta^*} \rangle_{\mathcal{A}}$$

f_{θ^*}: function relative to which changes are penalized
$\mathcal{A}$: set of 'anchor points' at which the divergence between f_θ and f_{θ^*} is measured

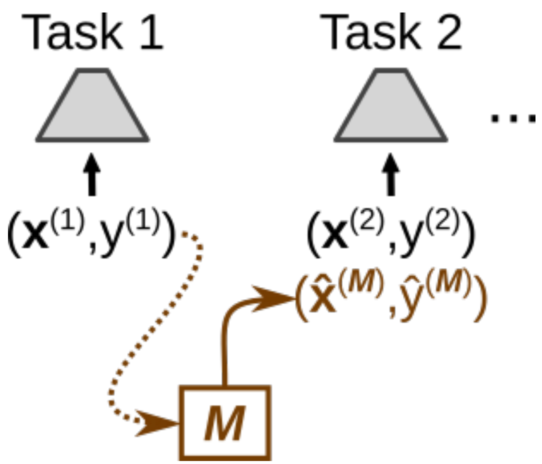


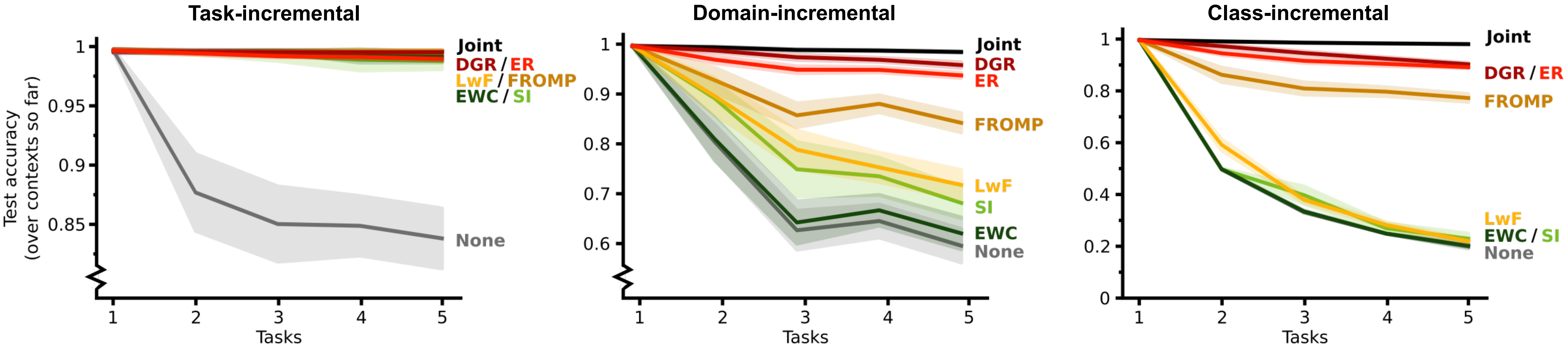
Memory buffer size (**FROMP**): 100 examples per class

Code for these experiments: https://github.com/GMvandeVen/continual-learning

Replay

- Current training data is complemented with data representative of past observations
- The replayed data can be sampled from a memory buffer or a generative model
- Representative methods:
  - Experience Replay [ER] (Chaudhry et al., 2019 arXiv)
  - Deep Generative Replay [DGR] (Shin et al., 2017 NeurIPS)



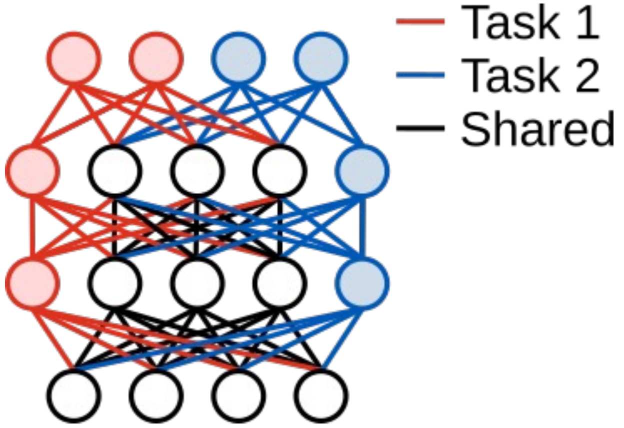


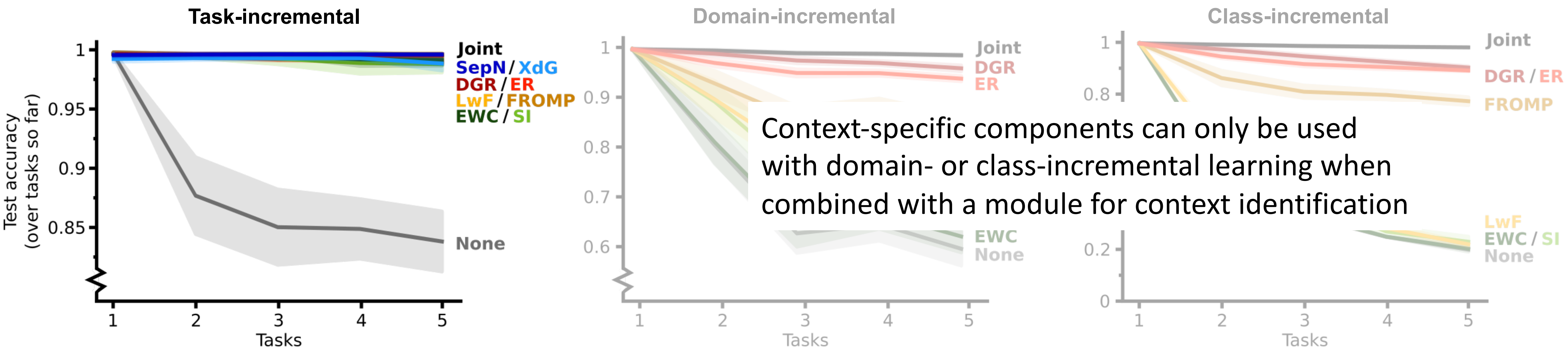
Memory buffer size (FROMP, ER): 100 examples per class

Code for these experiments: https://github.com/GMvandeVen/continual-learning

Context-specific components

- Parts of the network are only used for specific tasks
- Commonly used example: multi-headed output layer
- Requires knowledge of task identity at test time
- Representative methods:
  - Context-dependent Gating [XdG] ([Masse et al., 2018 PNAS]())
  - Separate Networks [SepN]





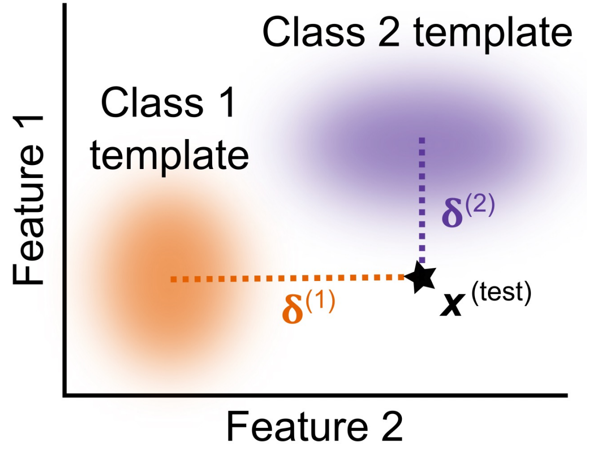
Context-specific components can only be used with domain- or class-incremental learning when combined with a module for context identification

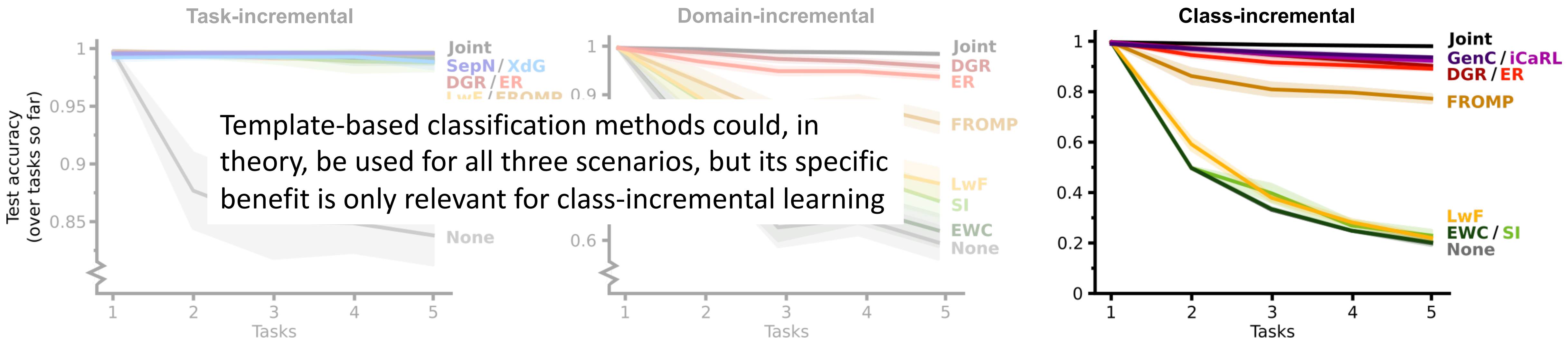
Memory buffer size (FROMP, ER): 100 examples per class

Code for these experiments: https://github.com/GMvandeVen/continual-learning

Template-based classification

- A 'template' is learned for each class, and classification is performed based on which template is most suitable for sample to be classified
- Examples of templates are prototypes or generative models
- Allows comparing classes 'at test time', rather than during training
- Representative methods:
  - Incremental Classifier and Representation Learning [**iCaRL**] ([Rebuffi et al., 2017 CVPR]())
  - Generative Classifier [**GenC**] ([van de Ven et al., 2021 CVPR-W]())





Template-based classification methods could, in theory, be used for all three scenarios, but its specific benefit is only relevant for class-incremental learning

Memory buffer size (**FROMP**, **ER**, **iCaRL**): 100 examples per class

Code for these experiments: https://github.com/GMvandeVen/continual-learning

Overview: Split CIFAR-100

Strategy	Method	Budget	GM	Task-IL	Domain-IL	Class-IL
Baselines	*None – lower target*			*61.43 (± 0.36)*	*18.42 (± 0.33)*	*7.71 (± 0.18)*
	Joint – upper target			*78.78 (± 0.25)*	*46.85 (± 0.51)*	*49.78 (± 0.21)*
Context-specific components	Separate Networks	-	-	76.83 (± 0.25)	-	-
	XdG	-	-	69.86 (± 0.34)	-	-
Parameter regularization	EWC	-	-	76.34 (± 0.29)	21.65 (± 0.55)	8.24 (± 0.25)
	SI	-	-	74.84 (± 0.39)	22.58 (± 0.42)	8.10 (± 0.24)
Functional regularization	LwF	-	-	78.59 (± 0.24)	29.45 (± 0.39)	25.57 (± 0.27)
	FROMP	100	-	not run	not run	not run
Replay	DGR	-	yes	71.40 (± 0.32)	20.52 (± 0.43)	9.67 (± 0.22)
	ER	100	-	76.43 (± 0.24)	39.00 (± 0.34)	37.57 (± 0.21)
Template-based classification	Generative Classifier	-	yes	-	-	46.83 (± 0.18)
	iCaRL	100	-	-	-	37.83 (± 0.21)

Shown is final test accuracy (as %, averaged over all tasks) on Split CIFAR-100. 'Budget' indicates number of samples per class stored in memory, 'GM' indicates generative model was learned using extra parameters. Experiments were run 10 times, reported is the mean (± SEM). Source: van de Ven et al. (2022, Nature Machine Intelligence)

Summary

- *Continual learning is not a unitary problem*: there are **three scenarios** that differ substantially in terms of difficulty and in terms of the effectiveness of different computational strategies
- **Regularization-based methods** often have relatively low memory and computational costs, but they struggle in certain settings
- **Replay** can work well in all three scenarios, but has relatively high memory and computational costs
- **Class-incremental learning** seems to require either replay (*to allow comparing classes during training*) or template-based classification (*to allow comparing classes during inference*)
- More details: *van de Ven et al. (2022, Nature Machine Intelligence)*

Funding acknowledgements

This research project has been supported by an IBRO-ISN Research Fellowship, by the ERC-funded project KeepOnLearning (reference number 101021347), by the Lifelong Learning Machines (L2M) program of the Defence Advanced Research Projects Agency (DARPA) via contract number HR0011-18-2-0025 and by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DoI/IBC) contract number D16PC00003. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA, IARPA, DoI/IBC, or the U.S. Government.

IBRO DARPA IARPA erc

Abbreviations and references of compared methods

- Context-specific components
  - ***Context-dependent Gating* (XdG)**
    Masse NY, Grant GD, Freedman DJ (2018) Alleviating catastrophic forgetting using context-dependent gating and synaptic stabilization. *PNAS* **115**: E10467-E10475.
- Parameter regularization
  - ***Elastic Weight Consolidation* (EWC)**
    Kirkpatrick J, Pascanu R, Rabinowitz N, Veness J, Desjardins G, …, Hadsell R (2017) Overcoming catastrophic forgetting in neural networks. *PNAS* **114**: 3521-3526.
  - ***Synaptic Intelligence* (SI)**
    Zenke F, Poole B, Ganguli S (2017) Continual learning through synaptic intelligence. *ICML*: 3987-3995.
- Functional regularization
  - ***Learning without Forgetting* (LwF)**
    Li Z, Hoiem D (2017) Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **40***:* 2935-2947.
  - ***Functional Regularization Of Memorable Past* (FROMP)**
    Pan P, Swaroop S, Immer A, Eschenhagen R, Turner R, Khan ME (2020) Continual deep learning by functional regularisation of memorable past. *NeurIPS*: 4453-4464.
- Replay
  - ***Deep Generative Replay* (DGR)**
    Shin H, Lee JK, Kim J, Kim J (2017) Continual learning with deep generative replay. *NeurIPS*: 2994-3003.
  - ***Brain-Inspired Replay* (BI-R)**
    van de Ven GM, Siegelmann HT, Tolias AS (2020) Brain-inspired replay for continual learning with artificial neural networks. *Nature Communications* **11**: 4069.
  - ***Experience Replay* (ER)**
    Rolnick D, Ahuja A, Schwarz J, Lillicrap T, Wayne G (2019) Experience replay for continual learning. *NeurIPS*: 32
    Chaudhry A, Rohrbach M, Elhoseiny M, Ajanthan T, Dokania PK, Torr PH, Ranzato MA (2019) On tiny episodic memories in continual learning. *arXiv preprint*: 1902.10486.
  - ***Averaged Gradient Episodic Memory* (A-GEM)**
    Chaudhry A, Ranzato MA, Rohrbach M, Elhoseiny M (2019) Efficient Lifelong Learning with A-GEM. *ICLR*.
- Template-based classification
  - **Generative Classifier**
    van de Ven GM, Zhe L, Tolias AS (2021) Class-incremental learning with generative classifiers. *CVPR-W proceedings*: 3611-3620.
  - ***Incremental Classifier and Representation Learning* (iCaRL)**
    Rebuffi SA, Kolesnikov A, Sperl G, Lampert CH (2017) icarl: Incremental classifier and representation learning. *CVPR proceedings*: 2001-2010.