Overcoming catastrophic forgetting in neural networks*

Giacomo Frigo (ID 626201), g.frigo@studenti.unipi.it

* Kirkpatrick et al., 2017. URL: arxiv.org/pdf/1612.00796.pdf

Introduction

- Continual Learning is crucial to achieve artificial general intelligence
- *catastrophic forgetting*: knowledge of previously learnt task(s) is lost as information relevant to current task is incorporated
- CL strategies*:
  - Task-specific Components
  - Data Replay (i.e. '*Exact Replay*', '*Distilled Replay*', '*Deep Generative Replay*'..)
  - Regularized Optimization
- EWC selectively slows down learning on the weigths important for old tasks
- Biologically inspired solution: Mammalian brain avoid catastrophic forgetting by synapses strengthening

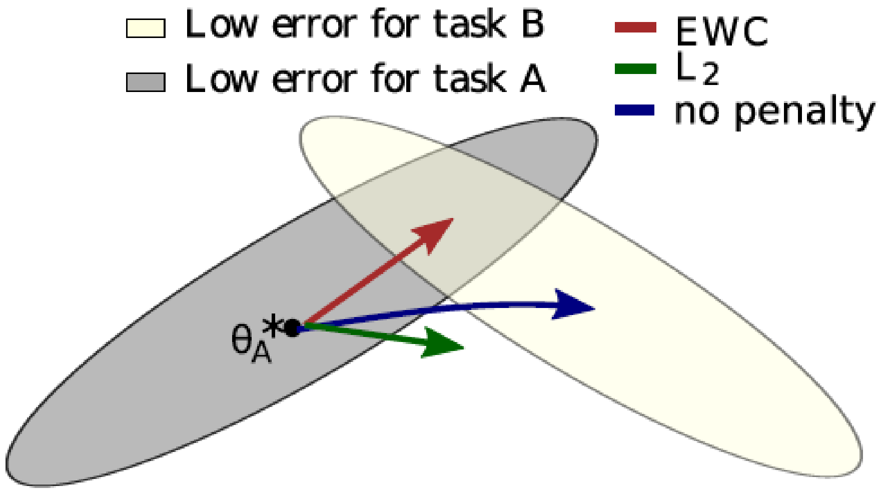
*Gido M. van de Ven and Andreas S. Tolias, Three scenarios for continual learning, 2019, 1904.07734.pdf (arxiv.org)

Elastic Weight Consolidation (EWC)

- EWC slows down learning on certain weights based on how important they are to previously seen tasks.

- While learning task B, EWC protect the performance of previously seen task A by constraining the parameters to stay in a region of low error for task A

- The parameters constraint is implemented as a quadratic penalty



Elastic Weight Consolidation (EWC)

From a probabilistic prospective neural network training can be described using Bayes rule:

$$log\ P(\theta | D) = log\ P(D | \theta) + log\ P(\theta) - log\ P(D)$$

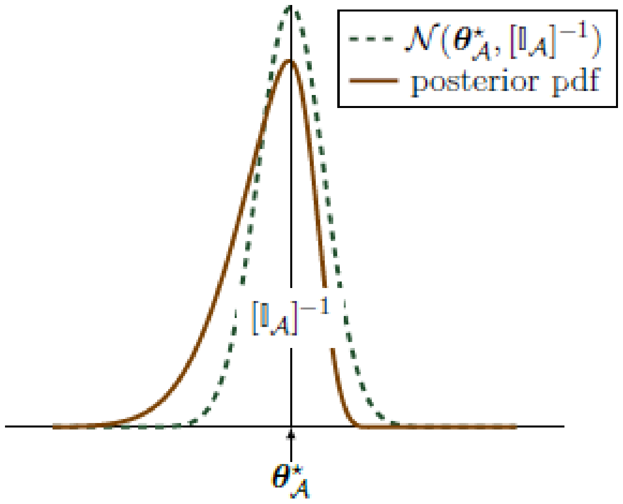
Assuming we have two indipendent task, respectively represented by data D_a and D_b we can re-arrange the above equation as:

$$log\ P(\theta | D) = log\ P(D_b | \theta) + \underbrace{log\ P(\theta | D_a)}_{\text{intractable}} - log\ P(D_b)$$

The posterior is approximated through Laplace approximation by Mackay*

*David JC MacKay. A practical bayesian framework for backpropogation networks, 1992

Elastic Weight Consolidation (EWC)



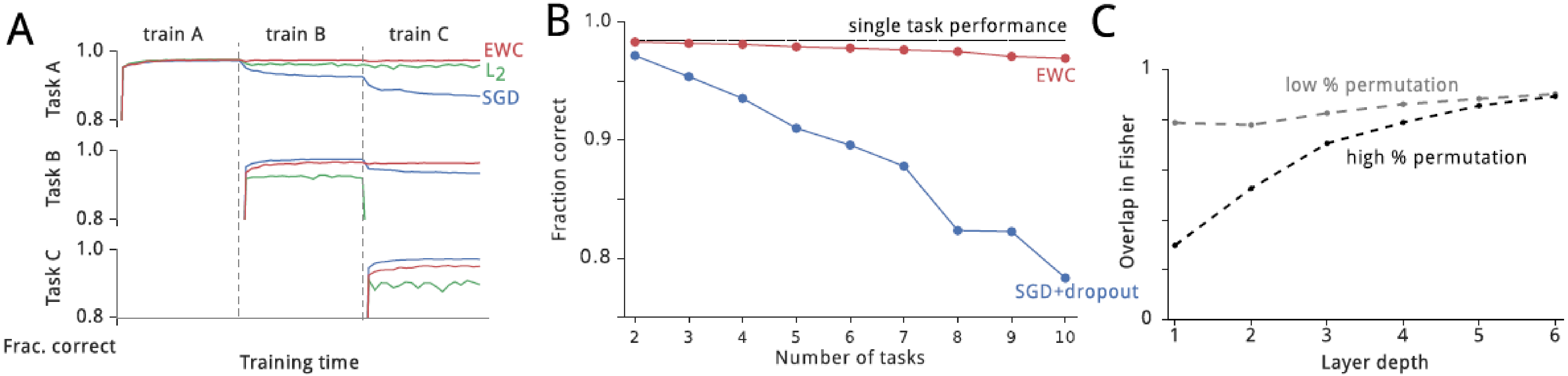
The posterior is approximated as a Gaussian distribution with mean given by parameters $\theta*_a$ and variance given by the inverse of the *Fisher information matrix* diagonal.

The function L that is minimized in EWC is:

$$L(\theta) = L_B(\theta) + \sum_i \frac{\lambda}{2} F_i(\theta_i - \theta^*_{A,i})^2$$

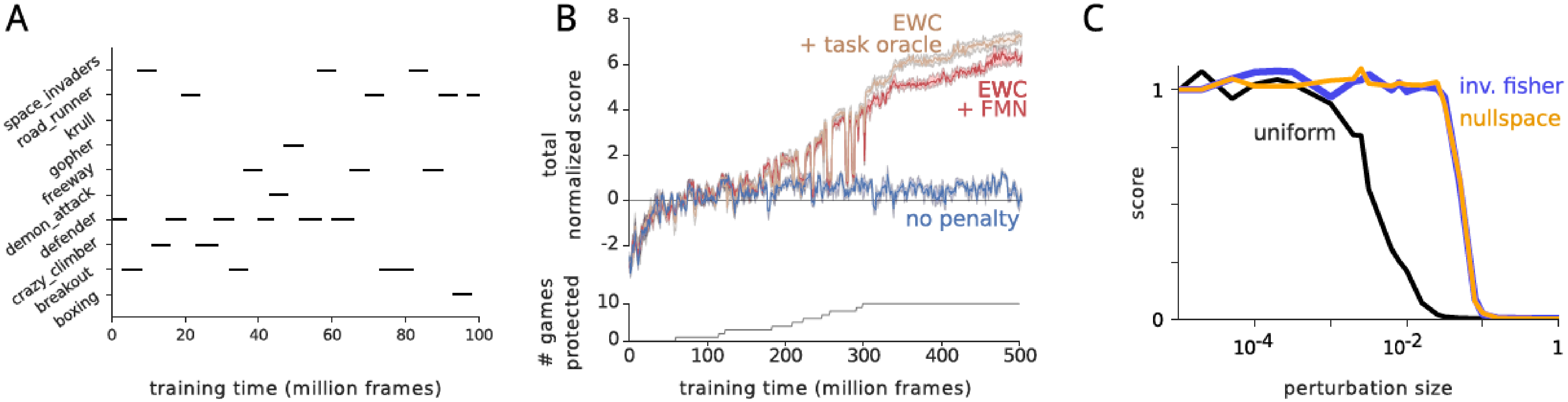
EWC in a supervised learning context

Set of tasks constructed from MNIST dataset, applying a fixed, random pixels permutation for each task



EWC in a reinforcement learning context

- EWC in a Deep Q Network
- Atari 2600 task set, with 10 games



Conclusions

- EWC compered with some related previous works has a lower run time, achieved also by using simplification, like the posterior distribution approximation.
- Dr. Ferenc Huszàr published a paper in December 2017, in which he show that cases in which there are more than two tasks, the quadratic penalties used in EWC are no well justified and might lead to double-counting data from earlier tasks.
- From EWC paper: « When moving to a third task, task C, EWC will try to keep network parameters close to the learned parameters of both task A and B. This can be enforced either with two seperate penalties or as one by noting that the sum of two quadratic penalties is itself a quadratic penalty »
- θ_B^* was obtained while penalizing departure from θ_A^*, therefore a penalty around θ_B^* already encapsulates the penalty around θ_A^*. In genaral EWC is introducing a systematic bias favouring tasks learned earlier.
- The EWC paper authors reply to Dr. Huszàr, stating that their paper failed to discuss the problem explicitly but their choice has been empirically validated.

Thank you!

Giacomo Frigo (ID 626201), g.frigo@studenti.unipi.it