Analyzing Catastrophic Forgetting in MLPs on Permuted-MNIST

Pavan Prasad Gorintla

CS 599: Foundations of Deep Learning — Lab 2

1 Problem and Setup

We follow the lab brief to study catastrophic forgetting using Permuted-MNIST with T=10 tasks, a standard MLP with depths $\{2, 3, 4\}$ (256 hidden units), and a sequential training schedule of $50 + 9 \times 20 = 230$ epochs. We report the Resulting Task Matrix R and compute ACC and BWT exactly as defined in the instructions.

2 Methods

Model: MLP with ReLU activations, optional dropout ≤ 0.5, and softmax output (10 classes).
Loss: Negative log-likelihood (sparse cross entropy) with optional L1/L2/L1+L2 kernel regularization.
Optimizers: SGD, Adam, RMSprop.
Data: For each task t, a fixed random pixel permutation is applied to MNIST.

3 Metrics

$\mathrm{ACC} = \frac{1}{T}\sum_{i=1}^{T} R_{T,i}, \quad \mathrm{BWT} = \frac{1}{T-1}\sum_{i=1}^{T-1}(R_{T,i} - R_{i,i}).$
We also report TBWT and CBWT (bonus).

4 Results (Fill from outputs)

Table 1 shows depth ablation with dropout and optimizer fixed. Replace placeholders with values from `metrics.json`.

5 Discussion

Summarize: effect of depth, dropout, optimizer, and L1/L2 on forgetting; trends visible in R; which choices reduce negative BWT; limitations and future work

Setting	ACC	BWT	TBWT	CBWT
Depth=2, loss=NLL, opt=SGD	---	---	---	---
Depth=3, loss=NLL, opt=SGD	---	---	---	---
Depth=4, loss=NLL, opt=SGD	---	---	---	---

Table 1: Depth ablation summary.

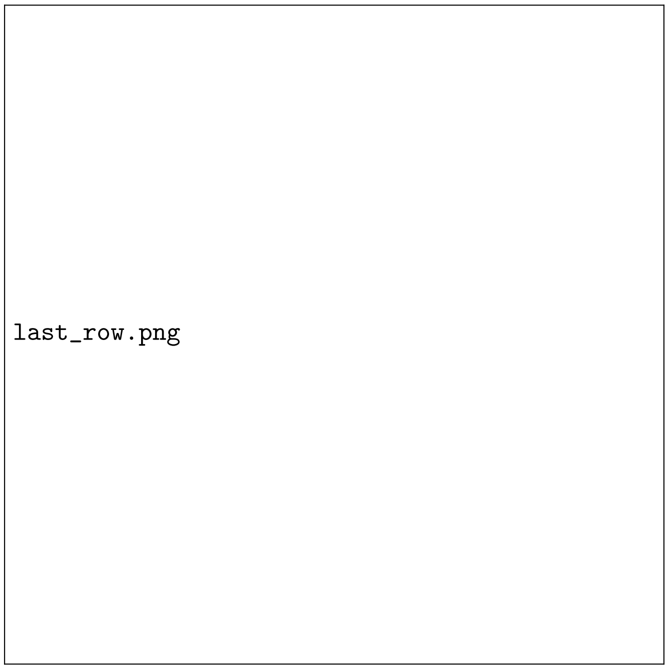

Figure 1: Accuracy by task after finishing all tasks (example run).

(e.g., EWC, replay).

6 References

Cite the lab brief and continual-learning papers as required.