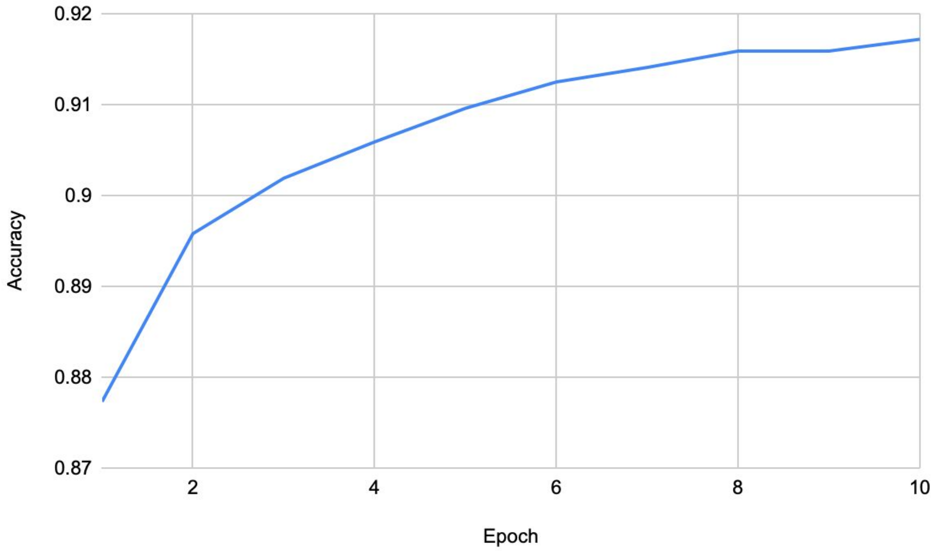
HW2 Report

Xinyi Li, Jing-Yao Chen, Zichuan Tian

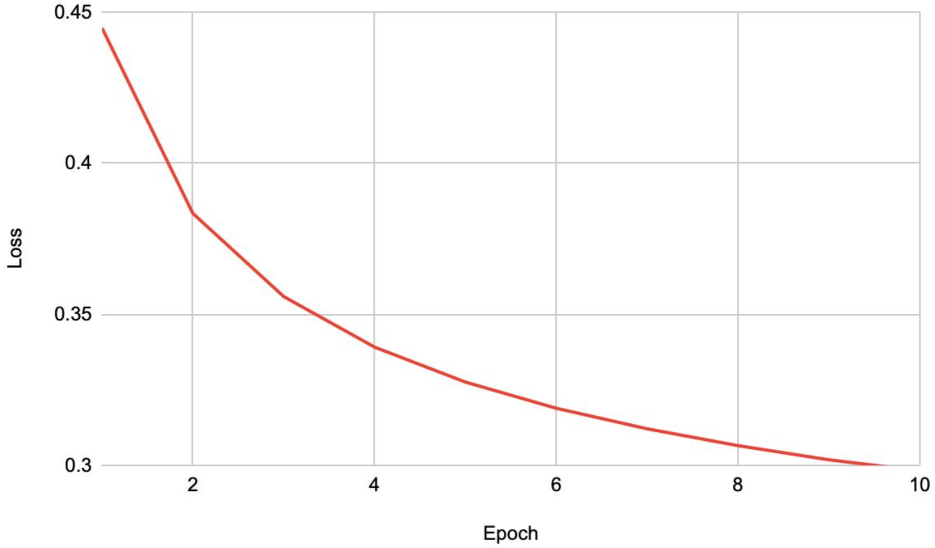
Part 1: Task 1:

We evaluated Accuracy and Loss vs number of Epochs. Plotted below:

=========================Accuracy vs Epochs=============================



==========================Loss vs Epochs===============================



We found that loss decrease and accuracy increase and they converge after around 100 epochs.

Task 2:

Number of nodes	Time (on each node) s	Accuracy	Error
1	156	0.9156	0.0844
2	256, 287	0.9207	0.0793
3	224, 333, 336	0.9209	0.0791

Single node mode spends 156s, achieving accuracy of 0.9156, error of 0.0844.
Double node mode spends
Double node mode spends 256s on node 0, and 287s on node 1. Achieving accuracy of 0.9207 and error of 0.0793
Triple node mode spends 223.9s on node 0, 332.8s on node 1, 335.6s on node 2, accuracy of 0.9209, error of 0.0791.
We think that node 0 finishes faster because parameter server runs on node 0, so it spends less time on network traffic.
Our claim is supported by the fact that memory and cpu usage on node 0 is about 50% higher than that on node 1 and 2.

Task 3:

Batch size	Time (s)	Loss	Accuracy	Peak memory usage (MB)	cpu
128	17.227	0.5000	0.85	560	21
256	13.6	0.65	0.81	640	23
512	10.7	0.88	0.73	730	21
1024	9.0	1.2	0.65	912	20
4096	8.4	2.6	0.26	780	22

As batch size bumping up, we observed that the program cost less time, and loss increases, accuracy decreases. We didn’t observe a trend in memory and cpu usage.

Part 2:
Task 1 & 2:

Batch size	Time (s)	Loss	Accuracy	Peak memory usage (MB)	cpu
128	81	0.0492	0.9843	663	60%
256	80	0.0647	0.9803	730	58%
512	87	0.07825	0.9765	840	61%
1024	90	0.1406	0.9577	958	57%
4096	90	0.4049	0.8880	2190	88%

As batch size bumping up, we observed that loss increases, accuracy decreases. We didn’t observe a trend in memory and cpu usage or time spent.