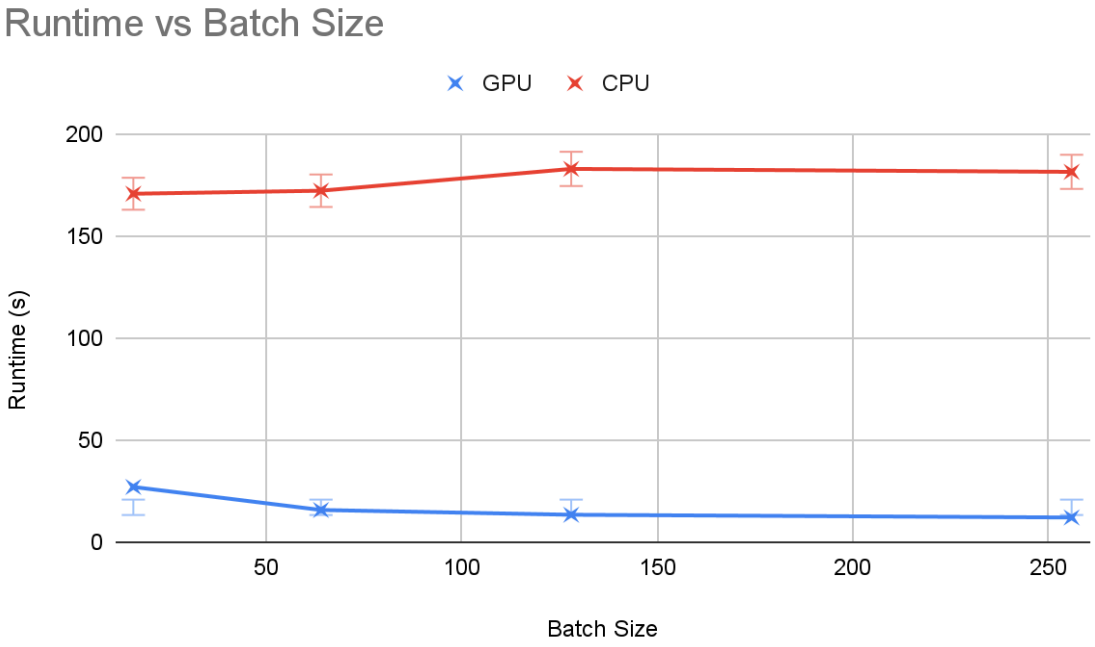
Aneesh Pandoh

A0

In this lab, the model was trained on the MNIST dataset where the training time for 1 epoch was measured to determine the difference performance between CPUs and GPUs as we vary batch size. The timings of an epoch is an average of 10 attempts for every configuration. For the default batch size of 128 the average timing was 13.45 seconds with a standard deviation of 0.31 on the gpu versus the cpu with the timing 182.86 seconds with a standard deviation of 4.75. The table and plot below depicts the differences for a few different batch sizes.

HW | Batch	GPU 16	GPU 64	GPU 128	GPU 256	CPU 16	CPU 64	CPU 128	CPU 256
Runtime(s)	27.07	15.79	13.45	12.12	170.69	172.17	182.86	181.41
STD	0.95	0.89	0.31	0.07	4.91	3.51	4.75	5.25



We can see that the the increase the batch size when using the GPU does end up decreasing the runtime as it can run the batches in parallel while still fitting within the GPU memory where as in the CPU there doesn't seem to be a trend and actually is slightly increasing runtime due the CPU's sequential nature.

The GPU used for this was the Tesla T4 which has 320 Tensor Cores and 2560 CUDA cores and a clock speed of 585 MHz. It has 16 GB of memory and a peak memory bandwidth of 320 GB/s and takes 70W of power. While we can see that the 585 MHz is pretty slow compared to any modern CPU, the massive amount of CUDA cores in combination with high memory bandwidth to feed in data allows for the GPU to significantly more performant for DNN training as the matrix multiplication required for the neural networks can be parallelized to utilize the increased core counts compared to the max ~256 of a cutting edge CPU. It supports precisions FP32, FP16, INT 8 and INT 4.