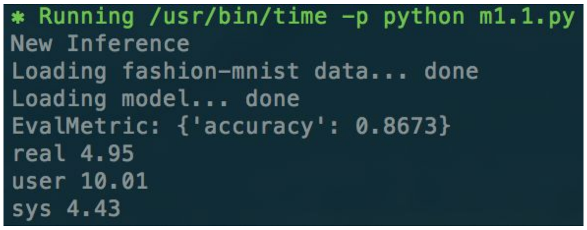
Nick Horcher, nhorche2
JJ Xu, jjxu3
Evan Lisoos, lissoos2

Checkpoint 1

1.1 Baseline Forward Pass

After running the code in rai and using
`/usr/bin/time -p`
we get the following

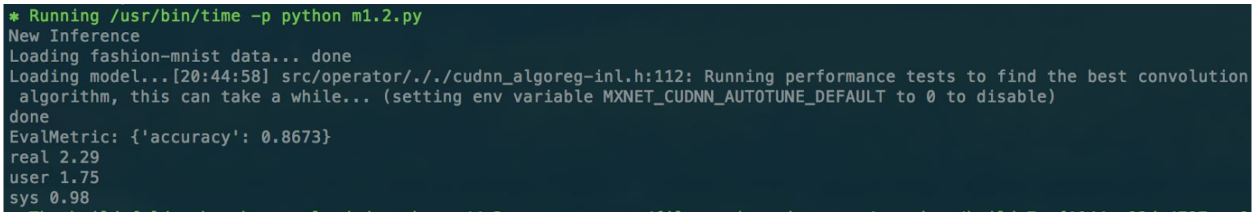


```
* Running /usr/bin/time -p python m1.1.py
New Inference
Loading fashion-mnist data... done
Loading model... done
EvalMetric: {'accuracy': 0.8673}
real 4.95
user 10.01
sys 4.43
```

We note that the accuracy reported by our instance of the run (0.8673) does indeed match the specified accuracy. From the time command, we see that the program took 4.95 seconds to run.

1.2 Baseline GPU Implementation

Similar to above, we run the GPU implementation with the time command.



```
* Running /usr/bin/time -p python m1.2.py
New Inference
Loading fashion-mnist data... done
Loading model...[20:44:58] src/operator/././cudnn_algoreg-inl.h:112: Running performance tests to find the best convolution
 algorithm, this can take a while... (setting env variable MXNET_CUDNN_AUTOTUNE_DEFAULT to 0 to disable)
done
EvalMetric: {'accuracy': 0.8673}
real 2.29
user 1.75
sys 0.98
```

The accuracy (0.8673) has not changed from above and still matches the specified accuracy. We observe that the total amount of time elapsed decreased to 2.29 seconds.

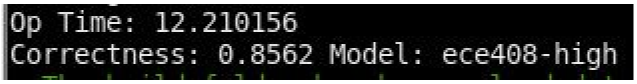
1.3 Generating an NVPROF Profile

Time (%)	Time (ms)	Kernel Function
36.50%	49.299	cudnn::detail::implicit_convolve_sgemm
28.25%	38.151	sgemm_sm35_ldg_tn_128x8x256x16x32
14.34%	19.369	cudnn::detail::activation_fw_4d_kernel
10.65%	14.380	cudnn::detail::pooling_fw_4d_kernel
5.62%	7.5966	[CUDA memcpy HtoD]

Checkpoint 2

After writing the CPU code and initializing the environment for checkpoint 2 to use m2.1.py we get the results:

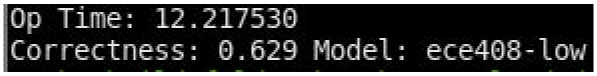
Op Time: **12.210156**
Correctness: **0.8562** with the Model: **ece408-high**



```
Op Time: 12.210156
Correctness: 0.8562 Model: ece408-high
```

AND

Op Time: **12.217530**
Correctness: **0.629** with the Model: **ece408-low**



```
Op Time: 12.217530
Correctness: 0.629 Model: ece408-low
```

Checkpoint 3

Op Time: 0.448302
Correctness: 0.8562 Model: ece408-high

NVPROF

Time (%)	**Time (ms)**	**Kernel Function**
80.29%	428.70ms	forward_kernel(...)
7.34%	39.206ms	sgemm_sm35_ldg_tn_128x8x256x16x32
3.67%	19.571ms	mshadow::cuda::MapPlanLargeKernel
3.63%	19.385ms	cudnn::detail::activation_fw_4d_kernel
2.72%	14.502ms	cudnn::detail::pooling_fw_4d_kernel

API Calls

Time (%)	**Time (ms)**	**Kernel Function**
42.50%	1.89817s	cudaStreamCreateWithFlags
25.83%	1.15367s	cudaFree
19.08%	852.44ms	cudaMemGetInfo
10.04%	448.28ms	cudaDeviceSynchronize
1.76%	78.692ms	cudaStreamSynchronize

Division of Labor

We all worked together to understand what needed to be done and decide how to implement the code. We went with an implementation very similar to the book. JJ created and setup the kernel function call. Evan created and the kernel code itself. We all worked together to find and fix the bugs. Nick did the report.