**Homework 4 – Docker MNIST**

1. **Wrtie Dockerfile**

Modify the example Dockerfile so that it can work with our case. First, change the base image to python:3.9-slim-bullseye for a better compatibility with pytorch. Next, change the maintainer information to myself. Delete the PORT part since we are not going to use that. Change the command to run the service to our source file main.py.

Also, since we don’t care about the training result, I also modify the main.py file to make it run for only 1 epoch to for us to run the test faster.

1. **Build the Docker Image and run the docker container**

Use the docker build command to create an image named mnist

docker build -t mnist .

After the image is built, we can run the docker container, and save the running result to

1. **Try diff batch sizes, draw a chart (x axis: batch size, y axis: flops and/or mem)**
2. **Try diff batch sizes, draw a chart (x axis: batch size, y axis: flops and/or mem)**
   1. Computational Complexity

The result of theoretical complexity is significantly lower than the measured FLOPs and execution time. This may be because pen and paper method provides an upper bound, often underestimated due to ignoring constants

* 1. Memory Usage

Measured memory usage is higher, especially with smaller batch sizes. This may because in our code, the memory are not only dedicated to be used in the cov2d-2 layer. Other code in our program may also use the memory during the process of our code execution.