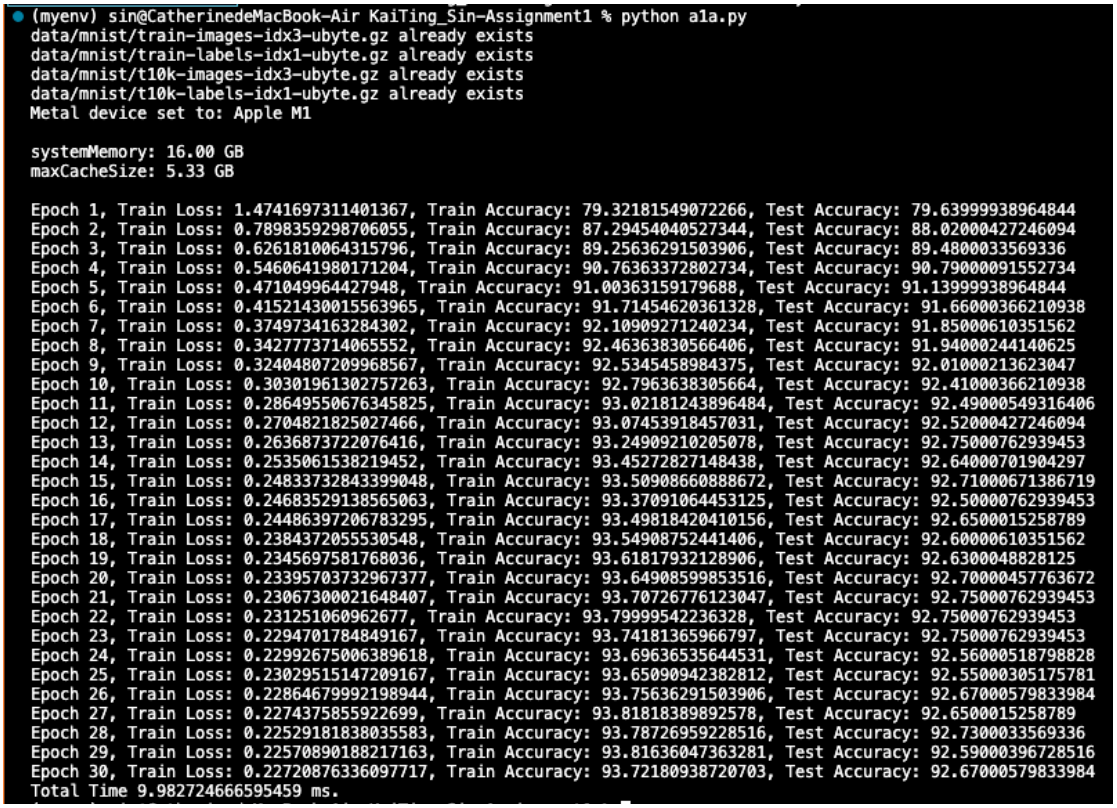
CSE598 Assignment 1

In this assignment, we are going to develop models for application of handwritten digit recognition using TensorFlow and MNIST dataset.

Task 1(file a1a.py)

Use logistic regression to classify image of handwritten digits.
First, implemented functions from utils.py to read the MNIST dataset into numpy arrays. Then, Process the train and test data in batches to reduce running time since we have massive data. Last draw samples from iterating the two datasets and compute loss and accuracy for training data and testing data after each epoch.

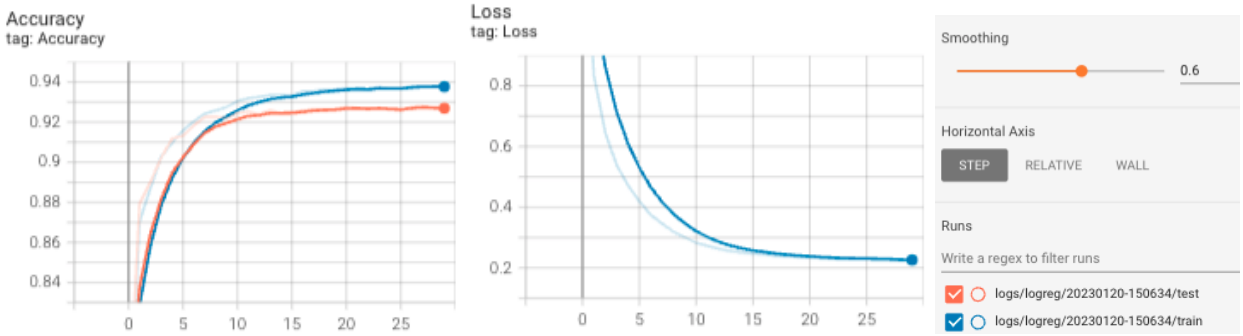


```
(myenv) sin@CatherinedeMacBook-Air KaiTing_Sin-Assignment1 % python a1a.py
data/mnist/train-images-idx3-ubyte.gz already exists
data/mnist/train-labels-idx1-ubyte.gz already exists
data/mnist/t10k-images-idx3-ubyte.gz already exists
data/mnist/t10k-labels-idx1-ubyte.gz already exists
Metal device set to: Apple M1

systemMemory: 16.00 GB
maxCacheSize: 5.33 GB

Epoch 1, Train Loss: 1.47416973114013 67, Train Accuracy: 79.32181549072266, Test Accuracy: 79.63999938964844
Epoch 2, Train Loss: 0.7898359298706055, Train Accuracy: 87.29454040527344, Test Accuracy: 88.02000427246094
Epoch 3, Train Loss: 0.6261810064315796, Train Accuracy: 89.25636291503906, Test Accuracy: 89.4800033569336
Epoch 4, Train Loss: 0.5460641980171204, Train Accuracy: 90.76363372802734, Test Accuracy: 90.79000091552734
Epoch 5, Train Loss: 0.471049964427948, Train Accuracy: 91.00363159179688, Test Accuracy: 91.13999938964844
Epoch 6, Train Loss: 0.41521430015563965, Train Accuracy: 91.71454620361328, Test Accuracy: 91.66000366210938
Epoch 7, Train Loss: 0.3749734163284302, Train Accuracy: 92.10909271240234, Test Accuracy: 91.85000610351562
Epoch 8, Train Loss: 0.3427773714065552, Train Accuracy: 92.46363830566406, Test Accuracy: 91.94000244140625
Epoch 9, Train Loss: 0.32404807209968567, Train Accuracy: 92.5345458984375, Test Accuracy: 92.01000213623047
Epoch 10, Train Loss: 0.30301961302757263, Train Accuracy: 92.7963638305664, Test Accuracy: 92.41000366210938
Epoch 11, Train Loss: 0.28649550676345825, Train Accuracy: 93.02181243896484, Test Accuracy: 92.49000549316406
Epoch 12, Train Loss: 0.2704821825027466, Train Accuracy: 93.07453918457031, Test Accuracy: 92.52000427246094
Epoch 13, Train Loss: 0.2636873722076416, Train Accuracy: 93.24909210205078, Test Accuracy: 92.75000762939453
Epoch 14, Train Loss: 0.2535061538219452, Train Accuracy: 93.45272827148438, Test Accuracy: 92.64000701904297
Epoch 15, Train Loss: 0.24833732843399048, Train Accuracy: 93.50908660888672, Test Accuracy: 92.71000671386719
Epoch 16, Train Loss: 0.24683529138565063, Train Accuracy: 93.37091064453125, Test Accuracy: 92.50000762939453
Epoch 17, Train Loss: 0.24486397206783295, Train Accuracy: 93.49818420410156, Test Accuracy: 92.6500015258789
Epoch 18, Train Loss: 0.2384372055530548, Train Accuracy: 93.54908752441406, Test Accuracy: 92.60000610351562
Epoch 19, Train Loss: 0.2345697581768036, Train Accuracy: 93.61817932128906, Test Accuracy: 92.6300048828125
Epoch 20, Train Loss: 0.23395703732967377, Train Accuracy: 93.64908599853516, Test Accuracy: 92.70000457763672
Epoch 21, Train Loss: 0.23067300021648407, Train Accuracy: 93.70726776123047, Test Accuracy: 92.75000762939453
Epoch 22, Train Loss: 0.231251060962677, Train Accuracy: 93.79999542236328, Test Accuracy: 92.75000762939453
Epoch 23, Train Loss: 0.2294701784849167, Train Accuracy: 93.74181365966797, Test Accuracy: 92.75000762939453
Epoch 24, Train Loss: 0.22992675006389618, Train Accuracy: 93.69636535644531, Test Accuracy: 92.56000518798828
Epoch 25, Train Loss: 0.23029515147209167, Train Accuracy: 93.65090942382812, Test Accuracy: 92.55000305175781
Epoch 26, Train Loss: 0.22864679992198944, Train Accuracy: 93.75636291503906, Test Accuracy: 92.67000579833984
Epoch 27, Train Loss: 0.2274375855922699, Train Accuracy: 93.81818389892578, Test Accuracy: 92.6500015258789
Epoch 28, Train Loss: 0.22529181838035583, Train Accuracy: 93.78726959228516, Test Accuracy: 92.7300033569336
Epoch 29, Train Loss: 0.22570890188217163, Train Accuracy: 93.81636047363281, Test Accuracy: 92.59000396728516
Epoch 30, Train Loss: 0.22720876336097717, Train Accuracy: 93.72180938720703, Test Accuracy: 92.67000579833984
Total Time 9.982724666595459 ms.
```

TensorBoard:



The best accuracy is about 92%. Time spent on completing this task is about 5-6 hours including debugging and environment setting.

Task 2(file a1b.py)

For improvement of the model of task 1, I tried to classify handwritten digits using a Multilayer Perceptron (MLP) Classifier.

First, flatten our 28x28 pixel images into a 784-length vector for each image. Then, normalized the grayscale values from 0-255 to 0-1 to reduce the complexity on computation of the neural network. Change the categories 1-9 into a binary matrix. Finally, build and fit the model then evaluate the loss and accuracy of it.

```
Epoch 18/30
60/60 - 1s - loss: 0.0129 - accuracy: 0.9982 - val_loss: 0.0641 - val_accuracy: 0.9807 - 615ms/epoch - 10ms/step
Epoch 19/30
60/60 - 1s - loss: 0.0113 - accuracy: 0.9987 - val_loss: 0.0590 - val_accuracy: 0.9813 - 620ms/epoch - 10ms/step
Epoch 20/30
60/60 - 1s - loss: 0.0095 - accuracy: 0.9991 - val_loss: 0.0593 - val_accuracy: 0.9819 - 632ms/epoch - 11ms/step
Epoch 21/30
60/60 - 1s - loss: 0.0084 - accuracy: 0.9994 - val_loss: 0.0599 - val_accuracy: 0.9818 - 622ms/epoch - 10ms/step
Epoch 22/30
60/60 - 1s - loss: 0.0074 - accuracy: 0.9995 - val_loss: 0.0601 - val_accuracy: 0.9814 - 634ms/epoch - 11ms/step
Epoch 23/30
60/60 - 1s - loss: 0.0066 - accuracy: 0.9996 - val_loss: 0.0590 - val_accuracy: 0.9815 - 625ms/epoch - 10ms/step
Epoch 24/30
60/60 - 1s - loss: 0.0059 - accuracy: 0.9997 - val_loss: 0.0581 - val_accuracy: 0.9820 - 633ms/epoch - 11ms/step
Epoch 25/30
60/60 - 1s - loss: 0.0052 - accuracy: 0.9997 - val_loss: 0.0599 - val_accuracy: 0.9813 - 680ms/epoch - 11ms/step
Epoch 26/30
60/60 - 1s - loss: 0.0048 - accuracy: 0.9998 - val_loss: 0.0587 - val_accuracy: 0.9819 - 660ms/epoch - 11ms/step
Epoch 27/30
60/60 - 1s - loss: 0.0042 - accuracy: 0.9998 - val_loss: 0.0598 - val_accuracy: 0.9820 - 649ms/epoch - 11ms/step
Epoch 28/30
60/60 - 1s - loss: 0.0038 - accuracy: 0.9999 - val_loss: 0.0603 - val_accuracy: 0.9814 - 619ms/epoch - 10ms/step
Epoch 29/30
60/60 - 1s - loss: 0.0034 - accuracy: 0.9999 - val_loss: 0.0601 - val_accuracy: 0.9821 - 629ms/epoch - 10ms/step
Epoch 30/30
60/60 - 1s - loss: 0.0031 - accuracy: 0.9999 - val_loss: 0.0603 - val_accuracy: 0.9809 - 635ms/epoch - 11ms/step
```

```
loss and accuracy:  [0.060323685407638555, 0.9809000492095947]
Total Time 21.69793701171875 ms.
```

The neural network structure:

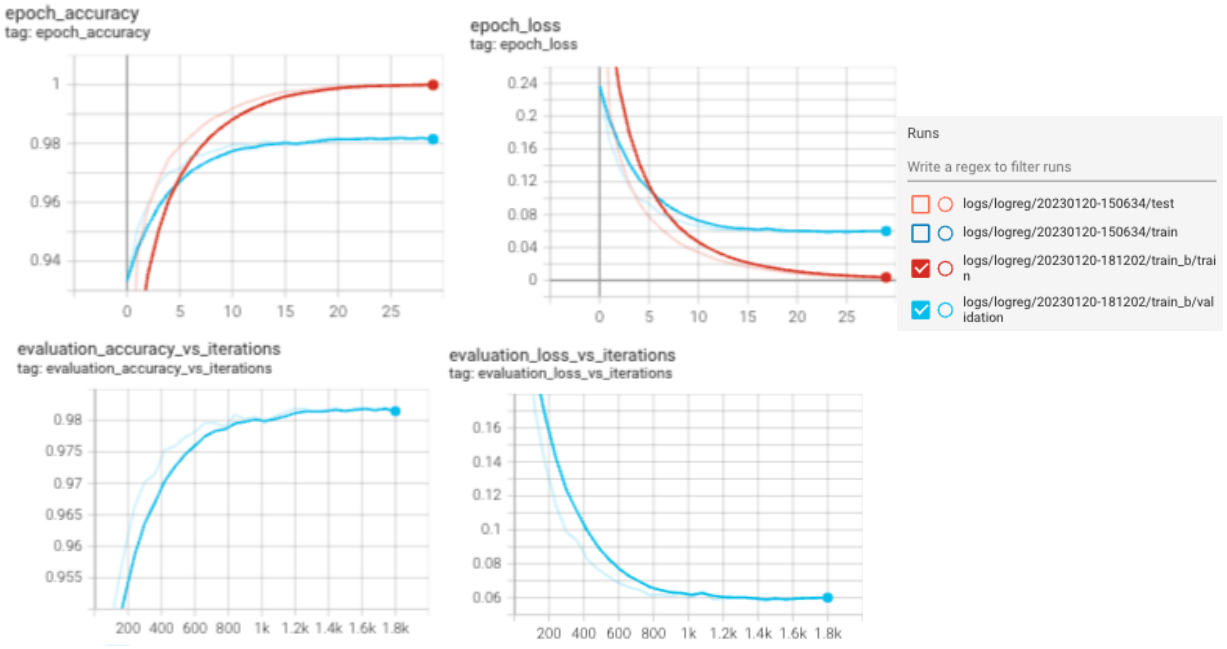
Visible Layer (784 Inputs) >> Hidden Layer (784 Neurons) >> Output Layer (10 Outputs)

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 784)               615440

 dense_1 (Dense)             (None, 10)                7850

=================================================================
Total params: 623,290
Trainable params: 623,290
Non-trainable params: 0
```

TensorBoard:



With the same epochs, the MLP model has a better accuracy than the logistic regression model that is up to 98%. Moreover, the time processing spent is about 21ms, which took more time than the logistic regression model (9.85ms). It may be caused by the decrease of the batch size that I set. If epochs increase, the time spent would be more significant.

Time spent on completing task 2 is around 8-9 hours including research on method applicable on handwritten digits recognition, development of the model, visualization for Tensorboard and debugging.

Interesting problems met during this assignment

For the environment setting. I have encountered some problems. First is installing TensorFlow. For Apple M1 computer users, we need to install different version of the packages, including tensorflow-deps, tensorflow-macos, and tensorflow-metal, so I'm not sure if the reference list of dependencies is met. Follow the instruction on the official site, latest versions are installed. TensorFlow is also tested correctly by running the example code logreg_example.py. However, when completing the task 1 and tried to run it. I got an error message:

”tensorflow.python.framework.errors_impl.NotFoundError: could not find registered platform with id: 0x11c2f35c0 ”. I searched for solution and found out that it might be caused by mac m1 version. The problem most probably has to do with recent changes on TensorFlow side for version 2.11 where a new optimizer API has been. Thus, I dropped back to the following versions: tensorflow-macos==2.9 and tensorflow-metal==0.5.0 and the problem has fixed for now.

The other difficulty is the output of test dataset’s accuracy scalars graph in TensorBoard in task 2. Using callbacks in the fit() function of the model can log events for TensorBoard of each epoch. However, it can’t log events of each epoch for the test data in evaluate() function so that the graph epoch_accuracy, epoch_loss only contain train and validation dataset.