Name: Vishisht Jain

Purdue Email: [jain322@purdue.edu](mailto:jain322@purdue.edu)

Git Link: https://github.com/Vishisht182/CS390\_Lab2

List of resources: Towardsdatascience, stackoverflow, matplotlib and tensorflow wikis for doubts

Completed:

* Standard ANN
* MNIST digit accuracy > 99%
* MNIST fashion accuracy > 92%
* Cifar 10 accuracy > 70%
* Cifar 100 coarse accuracy > 60%
* Cifar 100 fine accuracy > 50%
* Plots for accuracy and data pre-processing
* Added cmd-line arguments for running program

Questions:

* A CNN in this situation is superior as it’s method of learning features progressively through filters and pooling allows it to understand the parts of the data better and predict better than ANNs, which only parse through data like this as a tensor of numbers with no consideration of a common structure between similar samples.
* Pooling reduces the dimensionality of the data and keeps the important information (if said information is represented by bigger umbers in the tensor). This reduces the load on the network of working on a large multidimensional input and instead progressively work on a smaller one.
* Cifar dataset is a 32x32 RGB image which means an extra third dimension for the images to process, whereas an MNIST dataset has 28x28 greyscale images. Processing the data is computationally easier.
* I increase the accuracy of my CNN by adding multiple conv-pooling sections (64,64 with a 3x3 kernel size to a 128,128 set with a 3x3 kernel size) with increasing number of filters plus batch normalization. This lets my network progressively learn finer and finer features. I also add a large double layer of 512 dense neurons after flattening the final convoluted tensor which I scale down to the output layer. For the loss and optimization, cat. cross entropy is viable for most scenarios and adam is a good all-rounder. The epoch size is set according to if data augmentation is done or not.

Accuracy Graphs for ANN and CNN:

Chart, bar chart

Description automatically generated

Chart, bar chart

Description automatically generated

**CMD-Line output for ANN:**

❯ python Lab1.py tf\_net mnist\_d

Dataset: mnist\_d

Shape of xTrain dataset: (60000, 28, 28).

Shape of yTrain dataset: (60000,).

Shape of xTest dataset: (10000, 28, 28).

Shape of yTest dataset: (10000,).

New shape of xTrain dataset: (60000, 784).

New shape of xTest dataset: (10000, 784).

New shape of yTrain dataset: (60000, 10).

New shape of yTest dataset: (10000, 10).

Building and training TF\_NN.

Epoch 1/6

2000/2000 [==============================] - 14s 5ms/step - loss: 0.9245 - accuracy: 0.8333

Epoch 2/6

2000/2000 [==============================] - 10s 5ms/step - loss: 0.1863 - accuracy: 0.9467

Epoch 3/6

2000/2000 [==============================] - 11s 5ms/step - loss: 0.1532 - accuracy: 0.9568

Epoch 4/6

2000/2000 [==============================] - 11s 5ms/step - loss: 0.1359 - accuracy: 0.9627

Epoch 5/6

2000/2000 [==============================] - 12s 6ms/step - loss: 0.1256 - accuracy: 0.9654

Epoch 6/6

2000/2000 [==============================] - 11s 6ms/step - loss: 0.1098 - accuracy: 0.9700

Testing TF\_NN.

Classifier algorithm: tf\_net

Classifier accuracy: 96.940000%

❯ python Lab1.py tf\_net mnist\_f

Dataset: mnist\_f

Shape of xTrain dataset: (60000, 28, 28).

Shape of yTrain dataset: (60000,).

Shape of xTest dataset: (10000, 28, 28).

Shape of yTest dataset: (10000,).

New shape of xTrain dataset: (60000, 784).

New shape of xTest dataset: (10000, 784).

New shape of yTrain dataset: (60000, 10).

New shape of yTest dataset: (10000, 10).

Building and training TF\_NN.

Epoch 1/6

2000/2000 [==============================] - 15s 5ms/step - loss: 2.6508 - accuracy: 0.7054

Epoch 2/6

2000/2000 [==============================] - 11s 5ms/step - loss: 0.4716 - accuracy: 0.8320

Epoch 3/6

2000/2000 [==============================] - 12s 6ms/step - loss: 0.4254 - accuracy: 0.8478

Epoch 4/6

2000/2000 [==============================] - 11s 6ms/step - loss: 0.4243 - accuracy: 0.8483

Epoch 5/6

2000/2000 [==============================] - 11s 5ms/step - loss: 0.4148 - accuracy: 0.8545

Epoch 6/6

2000/2000 [==============================] - 10s 5ms/step - loss: 0.4049 - accuracy: 0.8575

Testing TF\_NN.

Classifier algorithm: tf\_net

Classifier accuracy: 86.260000%

❯ python Lab1.py tf\_net cifar\_10

Dataset: cifar\_10

Shape of xTrain dataset: (50000, 32, 32, 3).

Shape of yTrain dataset: (50000, 1).

Shape of xTest dataset: (10000, 32, 32, 3).

Shape of yTest dataset: (10000, 1).

New shape of xTrain dataset: (50000, 3072).

New shape of xTest dataset: (10000, 3072).

New shape of yTrain dataset: (50000, 10).

New shape of yTest dataset: (10000, 10).

Building and training TF\_NN.

Epoch 1/6

1667/1667 [==============================] - 16s 7ms/step - loss: 12.1213 - accuracy: 0.1977

Epoch 2/6

1667/1667 [==============================] - 12s 7ms/step - loss: 1.9801 - accuracy: 0.2994

Epoch 3/6

1667/1667 [==============================] - 11s 7ms/step - loss: 1.8568 - accuracy: 0.3361

Epoch 4/6

1667/1667 [==============================] - 14s 8ms/step - loss: 1.8163 - accuracy: 0.3530

Epoch 5/6

1667/1667 [==============================] - 14s 8ms/step - loss: 1.7709 - accuracy: 0.3675

Epoch 6/6

1667/1667 [==============================] - 14s 8ms/step - loss: 1.7553 - accuracy: 0.3690

Testing TF\_NN.

Classifier algorithm: tf\_net

Classifier accuracy: 38.070000%

❯ python Lab1.py tf\_net cifar\_100\_c

Dataset: cifar\_100\_c

Shape of xTrain dataset: (50000, 32, 32, 3).

Shape of yTrain dataset: (50000, 1).

Shape of xTest dataset: (10000, 32, 32, 3).

Shape of yTest dataset: (10000, 1).

New shape of xTrain dataset: (50000, 3072).

New shape of xTest dataset: (10000, 3072).

New shape of yTrain dataset: (50000, 20).

New shape of yTest dataset: (10000, 20).

Building and training TF\_NN.

Epoch 1/6

1667/1667 [==============================] - 19s 8ms/step - loss: 12.6121 - accuracy: 0.0889

Epoch 2/6

1667/1667 [==============================] - 14s 8ms/step - loss: 2.7378 - accuracy: 0.1581

Epoch 3/6

1667/1667 [==============================] - 14s 8ms/step - loss: 2.6202 - accuracy: 0.1899

Epoch 4/6

1667/1667 [==============================] - 11s 7ms/step - loss: 2.5641 - accuracy: 0.2061

Epoch 5/6

1667/1667 [==============================] - 13s 8ms/step - loss: 2.5203 - accuracy: 0.2229

Epoch 6/6

1667/1667 [==============================] - 13s 8ms/step - loss: 12.2171 - accuracy: 0.1749

Testing TF\_NN.

Classifier algorithm: tf\_net

Classifier accuracy: 13.430000%

❯ python Lab1.py tf\_net cifar\_100\_f

Dataset: cifar\_100\_f

Shape of xTrain dataset: (50000, 32, 32, 3).

Shape of yTrain dataset: (50000, 1).

Shape of xTest dataset: (10000, 32, 32, 3).

Shape of yTest dataset: (10000, 1).

New shape of xTrain dataset: (50000, 3072).

New shape of xTest dataset: (10000, 3072).

New shape of yTrain dataset: (50000, 100).

New shape of yTest dataset: (10000, 100).

Building and training TF\_NN.

Epoch 1/6

1667/1667 [==============================] - 20s 8ms/step - loss: 9.1996 - accuracy: 0.0134

Epoch 2/6

1667/1667 [==============================] - 13s 8ms/step - loss: 4.3265 - accuracy: 0.0482

Epoch 3/6

1667/1667 [==============================] - 10s 6ms/step - loss: 4.0588 - accuracy: 0.0768

Epoch 4/6

1667/1667 [==============================] - 10s 6ms/step - loss: 3.9696 - accuracy: 0.0891

Epoch 5/6

1667/1667 [==============================] - 11s 6ms/step - loss: 3.9227 - accuracy: 0.0963

Epoch 6/6

1667/1667 [==============================] - 11s 6ms/step - loss: 3.8391 - accuracy: 0.1109

Testing TF\_NN.

Classifier algorithm: tf\_net

Classifier accuracy: 11.980000%

**CMD-Line output for CNN:**

❯ python Lab1.py tf\_conv mnist\_d

Dataset: mnist\_d

Shape of xTrain dataset: (60000, 28, 28).

Shape of yTrain dataset: (60000,).

Shape of xTest dataset: (10000, 28, 28).

Shape of yTest dataset: (10000,).

New shape of xTrain dataset: (60000, 28, 28, 1).

New shape of xTest dataset: (10000, 28, 28, 1).

New shape of yTrain dataset: (60000, 10).

New shape of yTest dataset: (10000, 10).

Building and training TF\_CNN.

Epoch 1/10

2000/2000 [==============================] - 72s 35ms/step - loss: 0.2229 - accuracy: 0.9333

Epoch 2/10

2000/2000 [==============================] - 74s 37ms/step - loss: 0.0686 - accuracy: 0.9782

Epoch 3/10

2000/2000 [==============================] - 72s 36ms/step - loss: 0.0506 - accuracy: 0.9848

Epoch 4/10

2000/2000 [==============================] - 73s 37ms/step - loss: 0.0421 - accuracy: 0.9873

Epoch 5/10

2000/2000 [==============================] - 71s 35ms/step - loss: 0.0342 - accuracy: 0.9896

Epoch 6/10

2000/2000 [==============================] - 69s 34ms/step - loss: 0.0288 - accuracy: 0.9908

Epoch 7/10

2000/2000 [==============================] - 69s 34ms/step - loss: 0.0294 - accuracy: 0.9908

Epoch 8/10

2000/2000 [==============================] - 69s 35ms/step - loss: 0.0239 - accuracy: 0.9924

Epoch 9/10

2000/2000 [==============================] - 68s 34ms/step - loss: 0.0204 - accuracy: 0.9936

Epoch 10/10

2000/2000 [==============================] - 68s 34ms/step - loss: 0.0195 - accuracy: 0.9939

Testing TF\_CNN.

Classifier algorithm: tf\_conv

Classifier accuracy: 99.430000%

❯ python Lab1.py tf\_conv mnist\_f

Dataset: mnist\_f

Shape of xTrain dataset: (60000, 28, 28).

Shape of yTrain dataset: (60000,).

Shape of xTest dataset: (10000, 28, 28).

Shape of yTest dataset: (10000,).

New shape of xTrain dataset: (60000, 28, 28, 1).

New shape of xTest dataset: (10000, 28, 28, 1).

New shape of yTrain dataset: (60000, 10).

New shape of yTest dataset: (10000, 10).

Building and training TF\_CNN.

Epoch 1/10

2000/2000 [==============================] - 152s 75ms/step - loss: 0.6639 - accuracy: 0.7660

Epoch 2/10

2000/2000 [==============================] - 157s 79ms/step - loss: 0.3501 - accuracy: 0.8743

Epoch 3/10

2000/2000 [==============================] - 161s 80ms/step - loss: 0.2864 - accuracy: 0.8965

Epoch 4/10

2000/2000 [==============================] - 162s 81ms/step - loss: 0.2602 - accuracy: 0.9066

Epoch 5/10

2000/2000 [==============================] - 183s 91ms/step - loss: 0.2297 - accuracy: 0.9197

Epoch 6/10

2000/2000 [==============================] - 186s 93ms/step - loss: 0.2120 - accuracy: 0.9242

Epoch 7/10

2000/2000 [==============================] - 185s 92ms/step - loss: 0.1961 - accuracy: 0.9269

Epoch 8/10

2000/2000 [==============================] - 191s 96ms/step - loss: 0.1837 - accuracy: 0.9340

Epoch 9/10

2000/2000 [==============================] - 242s 121ms/step - loss: 0.1642 - accuracy: 0.9415

Epoch 10/10

2000/2000 [==============================] - 202s 101ms/step - loss: 0.1522 - accuracy: 0.9445

Testing TF\_CNN.

Classifier algorithm: tf\_conv

Classifier accuracy: 92.940000%

❯ python Lab1.py tf\_conv cifar\_10

Dataset: cifar\_10

Shape of xTrain dataset: (50000, 32, 32, 3).

Shape of yTrain dataset: (50000, 1).

Shape of xTest dataset: (10000, 32, 32, 3).

Shape of yTest dataset: (10000, 1).

New shape of xTrain dataset: (50000, 32, 32, 3).

New shape of xTest dataset: (10000, 32, 32, 3).

New shape of yTrain dataset: (50000, 10).

New shape of yTest dataset: (10000, 10).

Building and training TF\_CNN.

Epoch 1/10

1667/1667 [==============================] - 173s 103ms/step - loss: 1.8012 - accuracy: 0.3819

Epoch 2/10

1667/1667 [==============================] - 166s 100ms/step - loss: 1.0382 - accuracy: 0.6377

Epoch 3/10

1667/1667 [==============================] - 167s 100ms/step - loss: 0.8580 - accuracy: 0.7005

Epoch 4/10

1667/1667 [==============================] - 174s 105ms/step - loss: 0.7271 - accuracy: 0.7501

Epoch 5/10

1667/1667 [==============================] - 177s 106ms/step - loss: 0.6447 - accuracy: 0.7792

Epoch 6/10

1667/1667 [==============================] - 180s 108ms/step - loss: 0.5712 - accuracy: 0.8034

Epoch 7/10

1667/1667 [==============================] - 184s 111ms/step - loss: 0.5106 - accuracy: 0.8263

Epoch 8/10

1667/1667 [==============================] - 185s 111ms/step - loss: 0.4519 - accuracy: 0.8432

Epoch 9/10

1667/1667 [==============================] - 189s 113ms/step - loss: 0.4046 - accuracy: 0.8587

Epoch 10/10

1667/1667 [==============================] - 189s 113ms/step - loss: 0.3628 - accuracy: 0.8750

Testing TF\_CNN.

Classifier algorithm: tf\_conv

Classifier accuracy: 79.780000%

❯ python Lab1.py tf\_conv cifar\_100\_c

Dataset: cifar\_100\_c

Shape of xTrain dataset: (50000, 32, 32, 3).

Shape of yTrain dataset: (50000, 1).

Shape of xTest dataset: (10000, 32, 32, 3).

Shape of yTest dataset: (10000, 1).

New shape of xTrain dataset: (150000, 28, 28, 3).

New shape of xTest dataset: (30000, 28, 28, 3).

New shape of yTrain dataset: (150000, 20).

New shape of yTest dataset: (30000, 20).

Building and training TF\_CNN.

Epoch 1/10

1667/1667 [==============================] - 440s 262ms/step - loss: 2.5433 - accuracy: 0.2409

Epoch 2/10

1667/1667 [==============================] - 445s 267ms/step - loss: 1.6825 - accuracy: 0.4736

Epoch 3/10

1667/1667 [==============================] - 462s 277ms/step - loss: 1.4021 - accuracy: 0.5585

Epoch 4/10

1667/1667 [==============================] - 492s 295ms/step - loss: 1.2110 - accuracy: 0.6147

Epoch 5/10

1667/1667 [==============================] - 510s 306ms/step - loss: 1.0725 - accuracy: 0.6581

Epoch 6/10

1667/1667 [==============================] - 480s 288ms/step - loss: 0.9588 - accuracy: 0.6926

Epoch 7/10

1667/1667 [==============================] - 493s 296ms/step - loss: 0.8640 - accuracy: 0.7207

Epoch 8/10

1667/1667 [==============================] - 507s 304ms/step - loss: 0.7868 - accuracy: 0.7434

Epoch 9/10

1667/1667 [==============================] - 525s 315ms/step - loss: 0.7215 - accuracy: 0.7657

Epoch 10/10

1667/1667 [==============================] - 501s 301ms/step - loss: 0.6722 - accuracy: 0.7799

Testing TF\_CNN.

Classifier algorithm: tf\_conv

Classifier accuracy: 62.593333%

❯ python Lab1.py tf\_conv cifar\_100\_f

Dataset: cifar\_100\_f

Shape of xTrain dataset: (50000, 32, 32, 3).

Shape of yTrain dataset: (50000, 1).

Shape of xTest dataset: (10000, 32, 32, 3).

Shape of yTest dataset: (10000, 1).

New shape of xTrain dataset: (150000, 28, 28, 3).

New shape of xTest dataset: (30000, 28, 28, 3).

New shape of yTrain dataset: (150000, 100).

New shape of yTest dataset: (30000, 100).

Building and training TF\_CNN.

Epoch 1/10

1667/1667 [==============================] - 395s 236ms/step - loss: 3.8607 - accuracy: 0.1264

Epoch 2/10

1667/1667 [==============================] - 443s 266ms/step - loss: 2.5900 - accuracy: 0.3356

Epoch 3/10

1667/1667 [==============================] - 453s 272ms/step - loss: 2.1612 - accuracy: 0.4273

Epoch 4/10

1667/1667 [==============================] - 485s 291ms/step - loss: 1.8993 - accuracy: 0.4826

Epoch 5/10

1667/1667 [==============================] - 516s 310ms/step - loss: 1.7097 - accuracy: 0.5278

Epoch 6/10

1667/1667 [==============================] - 482s 289ms/step - loss: 1.5530 - accuracy: 0.5641

Epoch 7/10

1667/1667 [==============================] - 488s 293ms/step - loss: 1.4335 - accuracy: 0.5926

Epoch 8/10

1667/1667 [==============================] - 508s 305ms/step - loss: 1.3345 - accuracy: 0.6150

Epoch 9/10

1667/1667 [==============================] - 521s 312ms/step - loss: 1.2453 - accuracy: 0.6369

Epoch 10/10

1667/1667 [==============================] - 541s 325ms/step - loss: 1.1815 - accuracy: 0.6549

Testing TF\_CNN.

Classifier algorithm: tf\_conv

Classifier accuracy: 51.630000%