**Big Data Systems and Analytics (CS6220 BDS 2019 Fall)**

Divyanshu Goyal | GTID: 903471575 | [dgoyal32@gatech.edu](mailto:dgoyal32@gatech.edu)

**Homework 1–Problem 2.1**

* **Introduction:**

For this assignment, I have done **Problem 2.1**, (Hand-on Experience with Deep Learning Framework for Building a K-class Image Classifier). For the purpose of this assignment I have chosen the following datasets:

1. The CIFAR10 dataset. ( https://www.cs.toronto.edu/~kriz/cifar.html)
2. The MNIST dataset of handwritten digits. ( http://yann.lecun.com/exdb/mnist/)
3. The USPS dataset of handwritten digits(https://www.kaggle.com/bistaumanga/usps-dataset)
4. At&t Face dataset (https://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html)

For the **CIFAR dataset** I have trained a 10-layer CNN neural network. The original CIFAR dataset had 50,000 images in the training set and 10,000 images in the testing set. Since due to limitation of memory of 4GB om my machine. I have taken a subset of 10,000 images for training and 2000 images for testing.

For the **MNIST**, **USPS** and **AT&T** data sets I am using 4-layer CNN’s with varied hyperparameters to get the maximum accuracy. The CNN trained on these following data sets is a 4-layer CNN. The primary reason for choosing a smaller number of layers for these datasets is because these datasets are grayscale (i.e. have only 1 channel) and hence have simple features as compared to the CIFAR dataset.

In the following section I have demonstrated the various size of datasets used for each the training set and other parameters as mentioned in the deliverables section of the assignment.

* **Observations:**

1. In the first table I am providing the input dataset analysis and analysis of the CNN trained on each of the datasets.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Datasets | Size | Resolution | Storage  (per Image) | Storage (Dataset) | Training: Testing (Split) | #weight Filter | Minibatch Size | #epochs |
| CIFAR | 60,000 | 32\*32\*3 | 2.457Kb | 178Mb | 10K:2K | 4 | 128 | 10 |
| MNIST | 70,000 | 28\*28\*1 | 784bytes | 52Mb | 8k:2K | 2 | 200 | 100 |
| USPS | 9,928 | 16\*16\*1 | 1024bytes | 2.8MB | 7291:2007 | 2 | 100 | 125 |
| At&t Face | 400 | 112\*92\*1 | 1030bytes | 4.7MB | 320:80 | 2 | 150 | 20 |

**Input Analysis:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Datasets | CIFAR | MNIST | USPS | At&t Face |
| Layer 1 | 3\*3,3->64(conv) | 4\*4,1 🡪 8 | 4\*4,1 🡪 8 | 4\*4,1 🡪 8 |
| ReLU,Maxpooling(2\*2) | ReLU,Maxpooling(4,4) | ReLU,Maxpooling(4,4) | ReLU,Maxpooling(4,4) |
| Layer 2 | 3\*3,64 🡪 128(conv) | 2\*2,8 🡪 16 | 2\*2,8 🡪 16 | 2\*2,8 🡪 16 |
| ReLU,Maxpooling(2\*2) | ReLU,Maxpooling(4,4) | ReLU,Maxpooling(4,4) | ReLU,Maxpooling(4,4) |
| Layer 3 | 5\*5,128 🡪 256(conv) | Flatten: 2\*2\*16 🡪 64 | Flatten: 2\*2\*16 🡪 64 | Flatten: 2\*2\*16 🡪 64 |
| ReLU,Maxpooling(2\*2) |
| Layer 4 | 5\*,5,256 🡪512(conv) | Fc: 64 🡪 10 | Fc: 64 🡪 10 | Fc: 64 🡪 10 |
| ReLU,Maxpooling(2\*2) |
| Layer 5 | Flatten:2\*2\*512🡪2048 |  |  |  |
| Layer 6 | Fc: 2048 🡪 128 |  |  |  |
| Layer 7 | Fc: 128 🡪 256 |  |  |  |
| Layer 8 | Fc: 256 🡪 512 |  |  |  |
| Layer 9 | Fc: 512 🡪 1024 |  |  |  |
| Layer 10 | Fc: 1024 🡪 10 |  |  |  |

1. In the second table I am providing the analysis for outputs of the respective CNN on each dataset.

**Output Analysis:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Datasets | Training Time | Training Accuracy | Testing Time | Testing Accuracy | Trained Model Size |
| CIFAR | 1976.34 sec | 85.71 | 5.738sec | 56.15 | 59MB |
| MNIST | 396.50sec | 96.0375 | 0.297sec | 92.95 | 72KB |
| USPS | 173.96sec | 97.95 | 0.1566sec | 92.72 | 68KB |
| At&t Face | 309.19 sec | 100 | 0.156sec | 76.25 | 384KB |

1. Next I demonstrate the Outlier test analysis. So, for the purpose of outlier set I curated a dataset of images of sheeps from google images and resized them to match the appropriate dataset sizes. The data set contains 10 images and is resized so that it can be used with the same trained CNN classifier for the respective datasets. And below are the results for the outlier test.

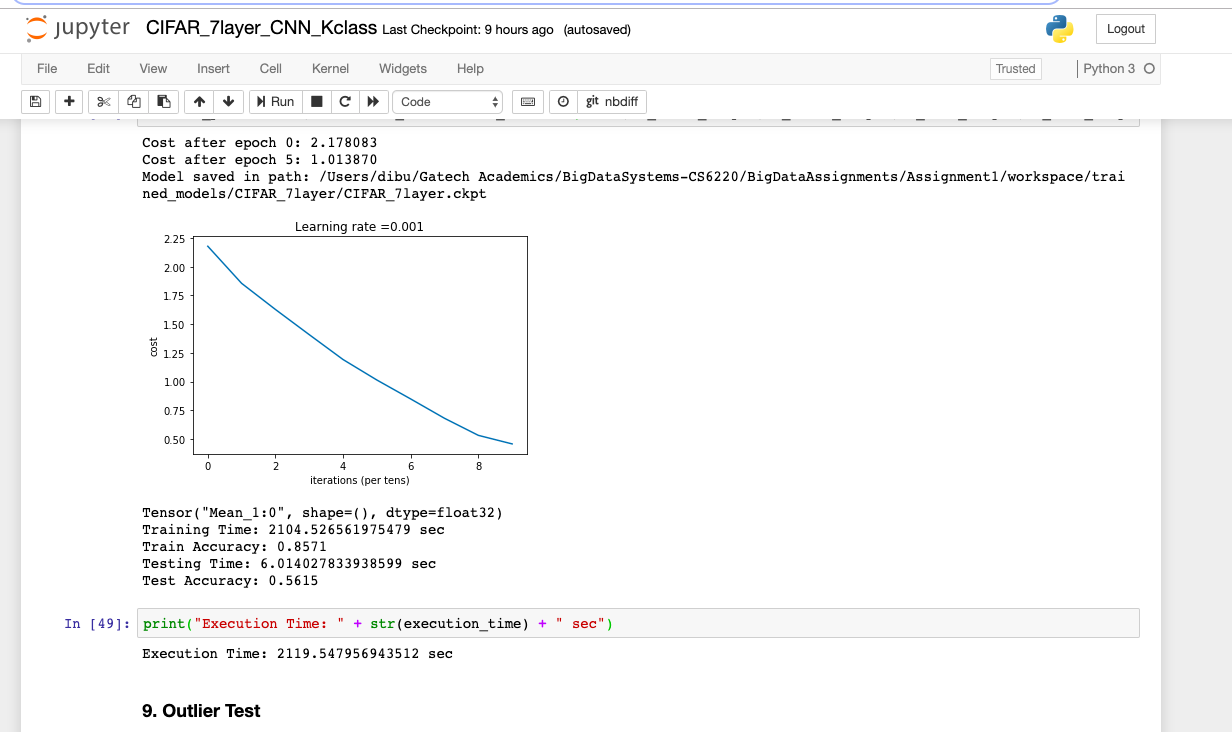
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Images from Outlier Dataset | Classifications Labels from the CNN classifier of the following Datasets | | | |
| CIFAR | MNIST  (0 to 9 digits) | USPS  (0to 9 digits) | At&t Face  (40 Faces) |
|  | Horse | 8 | 0 | Face 2 |
|  | Horse | 8 | 8 | Face 2 |
|  | Horse | 8 | 1 | Face 1 |
|  | Horse | 8 | 0 | Face 1 |
|  | Horse | 8 | 1 | Face 1 |
|  | Horse | 8 | 8 | Face 22 |
|  | Dog | 8 | 8 | Face 2 |
|  | Horse | 8 | 1 | Face 11 |
|  | Plane | 8 | 0 | Face 2 |
|  | Horse | 8 | 1 | Face 34 |

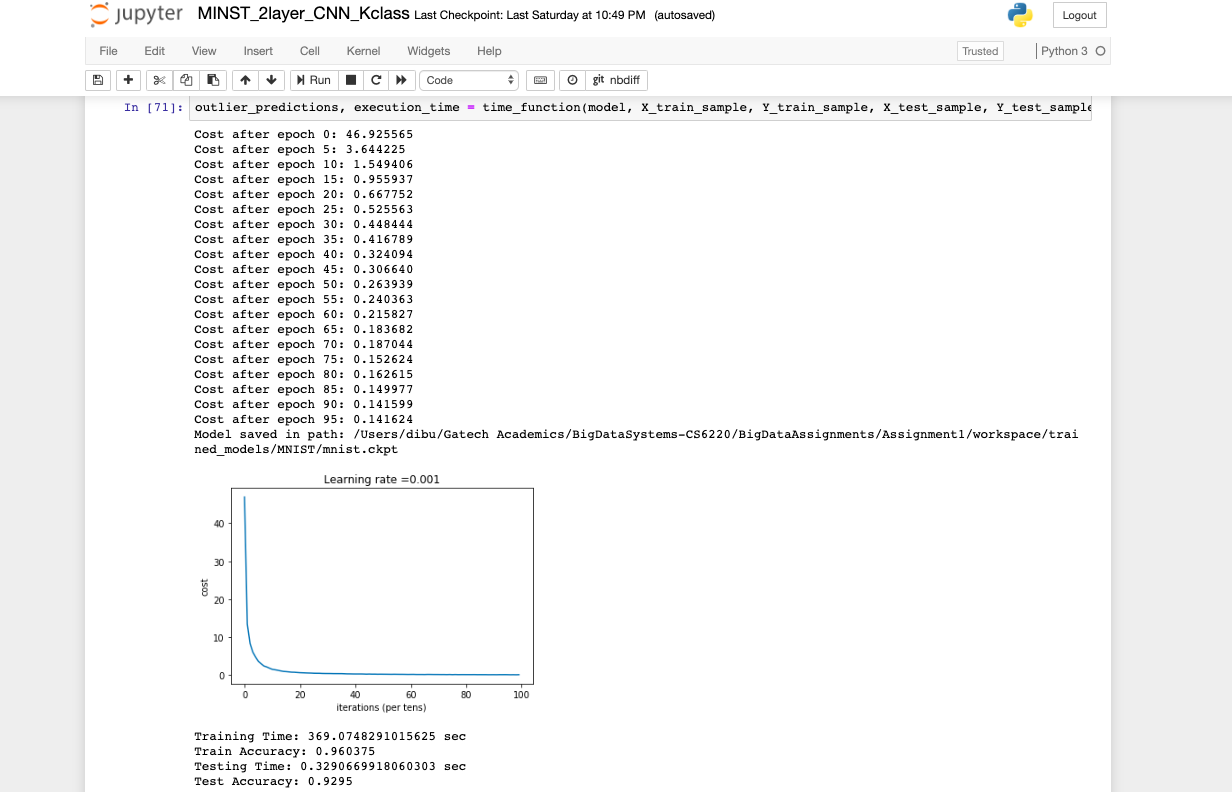
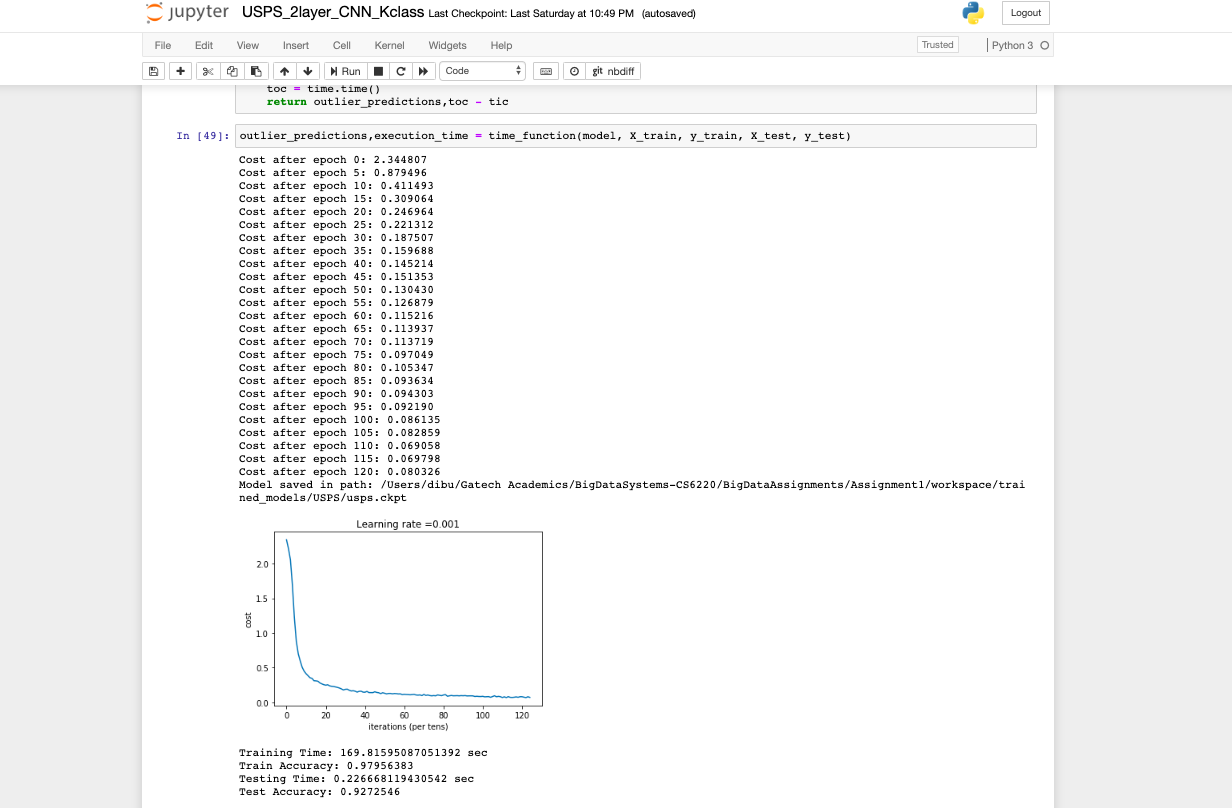
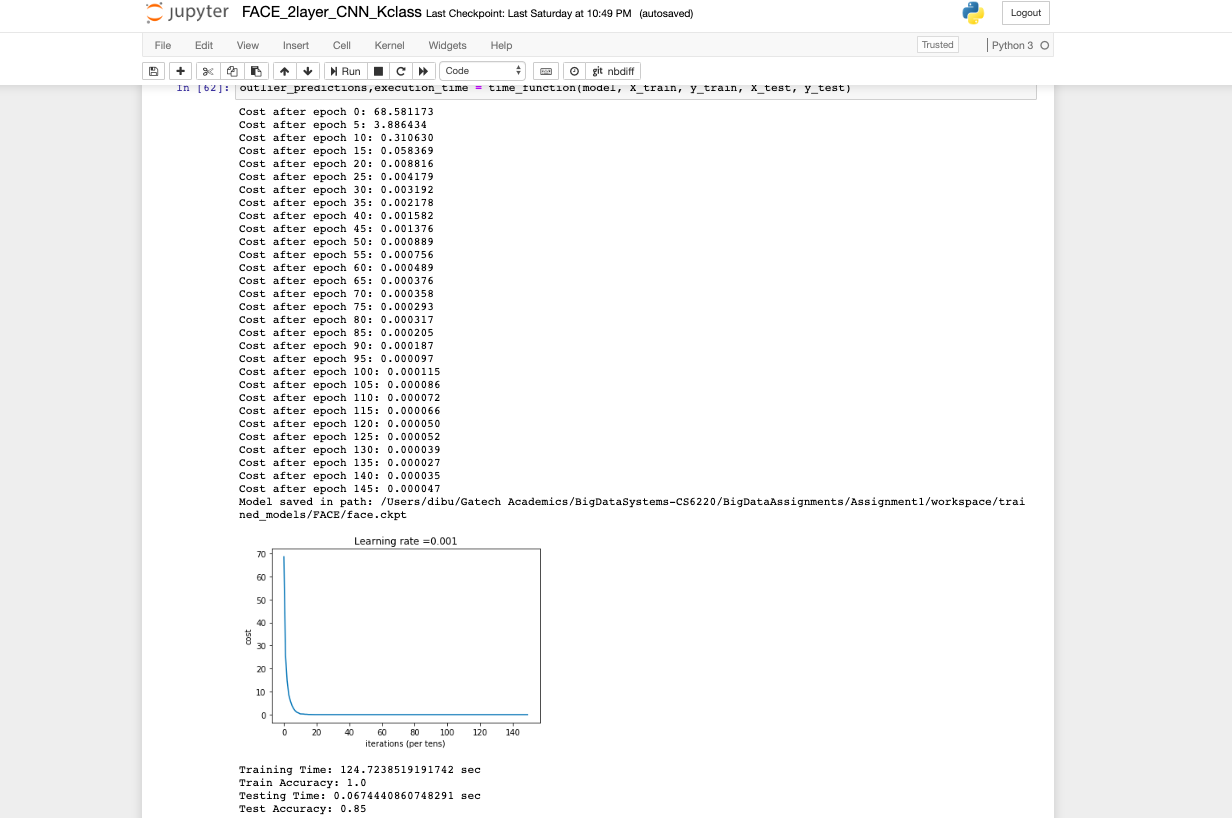
1. In this section I will present with the observations I made during the course of this experimentation.

* Firstly, training and the testing time is highly dependent on the size of the model trained. As we can clearly see CIFAR CNN with 10 layers is significantly larger in size and also takes significant amount of more time to train and test.
* Secondly, CIFAR data set requires a significantly a greater number of layers to train because of it has more complex features as compared to other datasets. So, as we increase the number of layers in CNN it is able to capture more complex relationships and patterns in an image,
* Thirdly, training accuracy is highly dependent on number of epochs, learning rate and size of train dataset. Choosing optimal values of learning rate, number of epochs is crucial to getting good training and testing accuracy. For this assignment I have manually tried to tweak these values to get optimal results. But this approach can be programmatically extended by searching the space for different values for #epochs and learning rates.

1. In this last section I am including some snapshots from the jupyter notebooks that I ran to train the CNN classifier for each dataset

**CIFAR Dataset:**

****

* **MNIST DATASET**
* **USPS DATASET:**
* **FACE DATASET**