CSE474/574: Introduction to Machine Learning(Fall 2017)
Project 3: Classification

Name: Prachi Jayakumar Patil
Person Number: **50247309**
UBIT Name: **ppatil3**

Objective: Implementation and evaluation of classification algorithms- logistic regression, single hidden layer neural network, convolutional neural network by training on MNIST dataset, tuning the hyper parameters and finally testing it on the USPS dataset.

1) Logistic Regression:

Hyper parameter Tuning:

Different Optimizers were used for optimization:

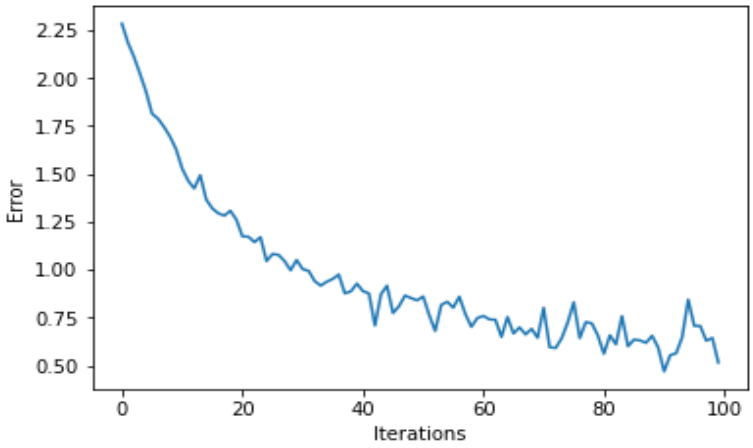
Optimisers	MNIST dataset	USPS data set
AdamOptimizer	0.8987	0.2506
GradientDescentOptimizer	0.9185	0.32
AdagradOptimizer	0.8583	0.284
FtrlOptimizer	0.8941	0.2987

Varying learning Rate:

Learning Rate	MNIST Dataset	USPS Dataset
0.1	0.9099	0.3246
0.5	0.9182	0.3146
0.9	0.9064	0.3046
0.01	0.8717	0.298

GradientDescentOptimizer is used for optimization that implements the gradient descent algorithm with learning rate 0.5.

Plot of Error Vs iterations for the training set:



2) <u>Single Hidden Layer Neural Network (Multi-Layer Perceptron):</u>

Hyper parameter Tuning:

1) Use of Optimizers:

Optimizers	MNIST Dataset	USPS Dataset
AdamOptimizer	0.9494	0.37133
GradientDescentOptimizer	0.6324	0.22
AdagradOptimizer	0.6159	0.174
FtrlOptimizer	0.9246	0.344

2) Learning Rate:

Learning Rate	MNIST Dataset	USPS Dataset
0.1	0.8182	0.2786
0.01	0.9504	0.382
0.05	0.8248	0.2961
0.001	0.854	0.3006

Different layers can be used:

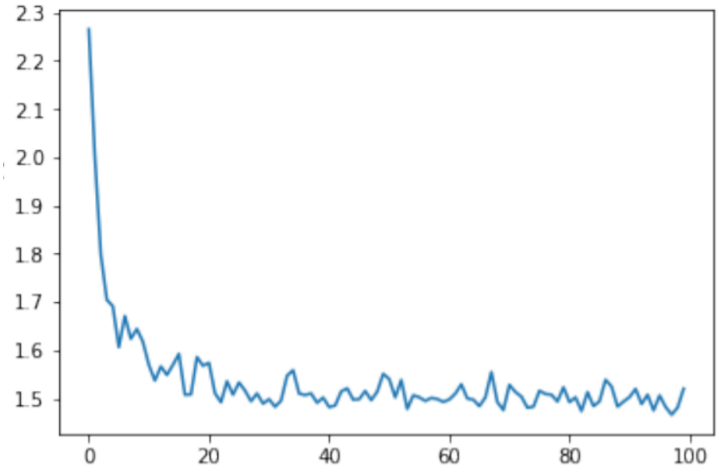
1) Convolutional Layer
2) Relu Layer
3) Pooling
4) Sigmoid layer
5) Softplus layer

Varying the number of features in the first layer:

Number of features	MNIST Accuracy	USPS Accuracy
30	0.9503	0.3467
40	0.9588	0.3953
20	0.9504	0.382

AdamOptimizer is used with learning rate with learning rate of 0.01 and RELU (rectified linear unit) activation layer is used.

Error Vs Iterations for MLP single hidden layer:



3) Convolutional Neural Network:

CNN has multiple layers in which each layer is a series of operations that we are applying to the data. The layers are applied are in the following order:

1) 2d Convolutional layer
2) Max pooling
3) 2d Convolutional layer
4) Max pooling
5) Fully connected layer
6) Dropout

A deeper neural network can be achieved by repeating the above layers.

Output of the code:

1. Logistic Regression:

```
Extracting ./MNIST_Logistic\train-images-idx3-ubyte.gz
Extracting ./MNIST_Logistic\train-labels-idx1-ubyte.gz
Extracting ./MNIST_Logistic\t10k-images-idx3-ubyte.gz
Extracting ./MNIST_Logistic\t10k-labels-idx1-ubyte.gz
Accuracy: 0.9182
USPS Test Accuracy: 0.3146
```

2. MLP with Single Hidden Layer Neural Network:

```
Extracting /tmp/tensorflow/mnist/input_data\train-images-idx3-ubyte.gz
Extracting /tmp/tensorflow/mnist/input_data\train-labels-idx1-ubyte.gz
Extracting /tmp/tensorflow/mnist/input_data\t10k-images-idx3-ubyte.gz
Extracting /tmp/tensorflow/mnist/input_data\t10k-labels-idx1-ubyte.gz
Optimization Finished!
Accuracy: 0.9504
time: 147.86796924675582
USPS Test Accuracy: 0.382
```

3. **Convolutional Neural Network:**

```
Extracting ./MNIST\train-images-idx3-ubyte.gz
Extracting ./MNIST\train-labels-idx1-ubyte.gz
Extracting ./MNIST\t10k-images-idx3-ubyte.gz
Extracting ./MNIST\t10k-labels-idx1-ubyte.gz
Saving graph to: C:\Users\prach\AppData\Local\Temp\tmp6074iaxm
step 0, training accuracy 0.02
step 100, training accuracy 0.84
step 200, training accuracy 0.88
step 300, training accuracy 0.96
step 400, training accuracy 0.94
step 500, training accuracy 0.92
step 600, training accuracy 0.98
step 700, training accuracy 1
step 800, training accuracy 1
step 900, training accuracy 0.96
step 1000, training accuracy 0.98
step 1100, training accuracy 0.94
step 1200, training accuracy 0.96
step 1300, training accuracy 0.98
step 1400, training accuracy 0.96
step 1500, training accuracy 1
step 1600, training accuracy 0.98
step 1700, training accuracy 0.98
step 1800, training accuracy 0.96
step 1900, training accuracy 0.96
test accuracy for mnist 0.9762
USPS Test Accuracy: 0.518
```

Results:

Models	Accuracy on MNIST dataset	Testing Accuracy on USPS dataset
Logistic Regression	0.9182	0.3146
MLP with 1 hidden layer	0.9504	0.382
Convolutional Neural Network	0.9762	0.518

No Free Lunch Theorem: No free lunch theorem states that any two optimization algorithms are equivalent when their performance is averaged across all possible problems. No machine learning algorithm is universally any better than any other.

For our example, when the models are trained on the MNIST data set and then tested on USPS data set, we can see that the train accuracy of the MNIST dataset is far greater than the test accuracy of USPS dataset.

Similarly, if we train the model on USPS dataset and test accuracy for MNIST dataset, the accuracy would be low for MNIST dataset.

Thus the two algorithms are not the same. The results prove the ‘No Lunch Theorem’.