PROJECT 3: CLASSIFICATION

Ravi Teja Sunkara (50292191)
rsunkara@buffalo.edu

Logistic Regression

Logistic Regression has been implemented with Regularizer and as it is a multi-class classification, SoftMax has been used.

Effect of Learning Rate

When learning rate is too less, the function doesn't reach the global minima and when the learning rate is too high the function diverges. So, the value I chose for the learning rate is 0.1

The values of other parameters: Lambda = 1, x1 = 0, xn = 256

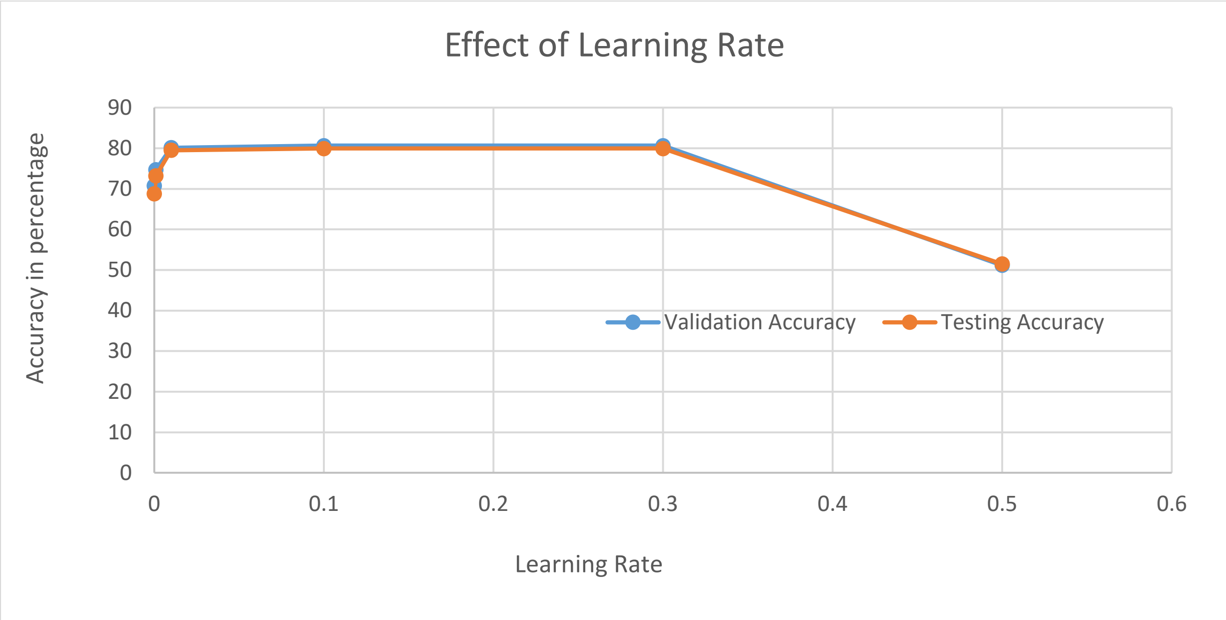

Fig 1: Graph showing the Effect of Learning Rate on the logistic Regression

Learning Rate	Validation Accuracy	Testing Accuracy
0.0001	70.74	68.78
0.001	74.72	73.22
0.01	80.14	79.5
0.1	80.62	79.92
0.3	80.62	79.92
0.5	51.16	51.46

Table 1: Table for different values of Learning Rate

Effect of Regularizer (Lambda)

As the regularizer value is increased, the validation and test accuracy are decreasing. So, the value of Lambda chosen for our Test data is 0.5

The other parameter values: Learning Rate = 0.1, x1 = 0, xn = 256

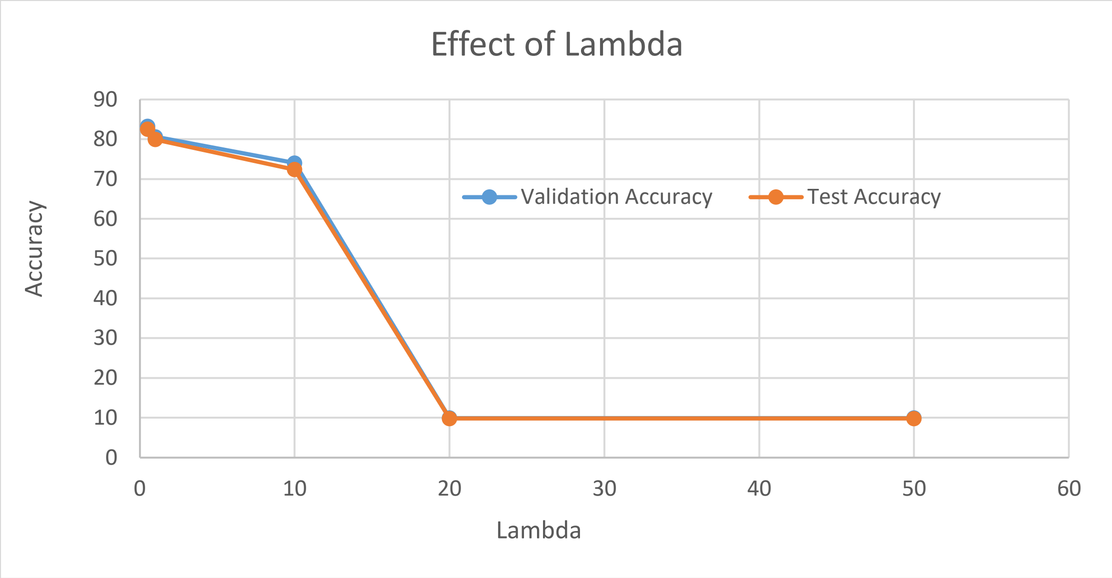

Fig 2: Graph showing the effect of Lambda on Accuracy

Lambda	Validation Accuracy	Test Accuracy
0.5	83.2	82.5
1	80.62	79.92
10	73.99	72.35
20	9.91	9.8
50	9.91	9.8

Table 2: Values of Accuracy for different Lambda

Effect of Batch Size

The lowest batch size has the lowest accuracy on the validation dataset. And for 256 samples, the validation accuracy is highest and so for testing data the Number of Samples is set to 256.

The other parameter values: Learning rate = 0.1, Lambda = 0.5

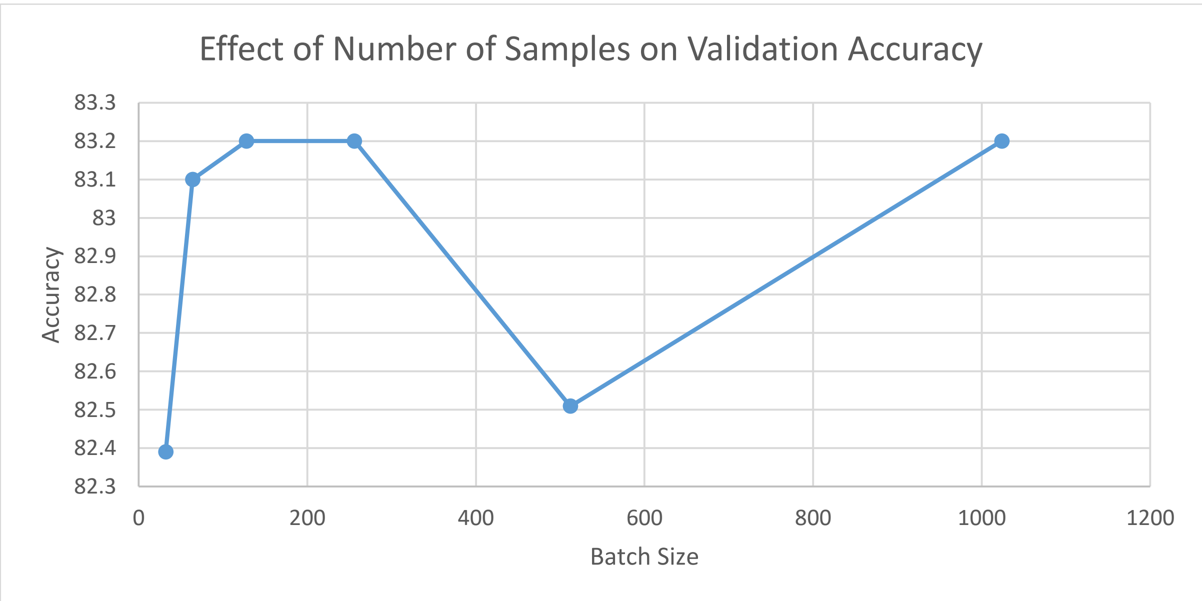

Fig 3: The effect of number of samples on validation accuracy

Batch Size	Validation Accuracy
32	82.39
64	83.1
128	83.2
256	83.2
512	82.509
1024	83.2

Table 3: Values of Validation Accuracy for different Batch Sizes

Confusion Matrix of Validation Dataset

```
[[ 943    0    4    6    2    3   15    2   13    3]
 [   0 1026    5    6    1    7    3    2   11    3]
 [   7   16  852   12   19    2   21   22   28   11]
 [   5    6   20  896    1   41    7    3   38   13]
 [   0   11    5    0  899    2    5    3    7   51]
 [  17    9   13   54   14  733   22    5   30   18]
 [   6    8   12    1    9    9  914    0    8    0]
 [  14   21   18    4   13    1    0  984    2   33]
 [   5   38   17   31    4   27    6   10  846   25]
 [   7    9    8   18   34    5    1   37    6  836]]
```

Final Model Parameter Settings

Learning Rate = 0.1
Lambda = 0.0001
Batch Size = 256

SUPPORT VECTOR MACHINE (SVM)

The SVM model has been implemented using the scikit libraries ‘SVC’. ‘linear’ kernel was chosen and hyper-parameter was the ‘Penalty Term (C)’.

Effect of Penalty Term (C)

As the C value increases the model fits to the training data more and more appropriately and when the C value is very large, the model over-fits to the training data. Accordingly, the accuracy for the validation data decreases. The same can be seen in the graph and the table below. So, for the model the C is being set to 0.0001

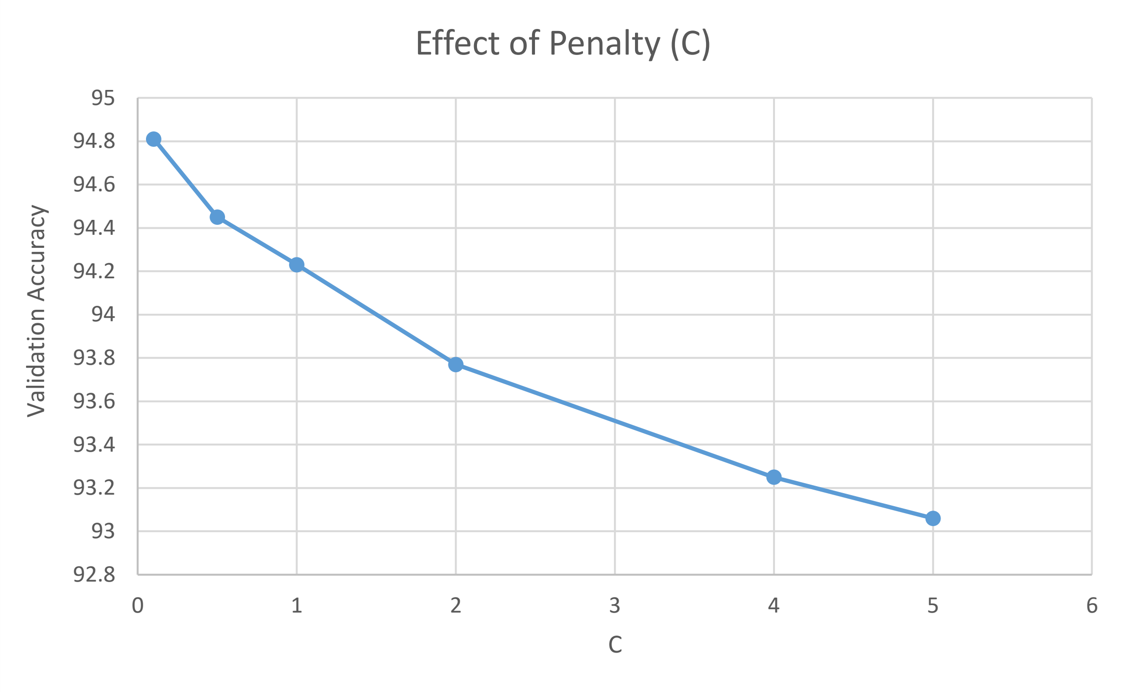

Fig 4: The effect of C on the Validation Accuracy

C	Validation Accuracy
0.1	94.81
0.5	94.45
1	94.23
2	93.77
4	93.25
5	93.06

Table 4: Values of validation accuracy for different values of C

Confusion Matrix of Validation Data

```
[[ 971    0    3    1    2    1    6    0    4    3]
 [   0 1051    1    5    1    0    0    1    4    1]
 [   3    3  945    7    7    2    6    6    8    3]
 [   4    2   12  961    1   28    1    3   14    4]
 [   1    8    5    0  939    0    5    2    1   22]
 [   8    3    9   37    2  823   18    3   10    2]
 [   4    1    6    0    4    5  945    0    2    0]
 [   2    3   13    6    6    2    0 1045    2   11]
 [   3   15    9   22    0   23    2    5  922    8]
 [   5    8    2   11   22    5    0   22    7  879]]
```

RANDOM FOREST

The model has been implemented using Scikit-learn library's RandomForestClassifier function. The quality of split has been measured using 'entropy'.

Effect of Number of Trees in the Forest

As the number of Trees increase, the Validation Accuracy is increasing. As a trade-off with time, the number of trees is set to 300.

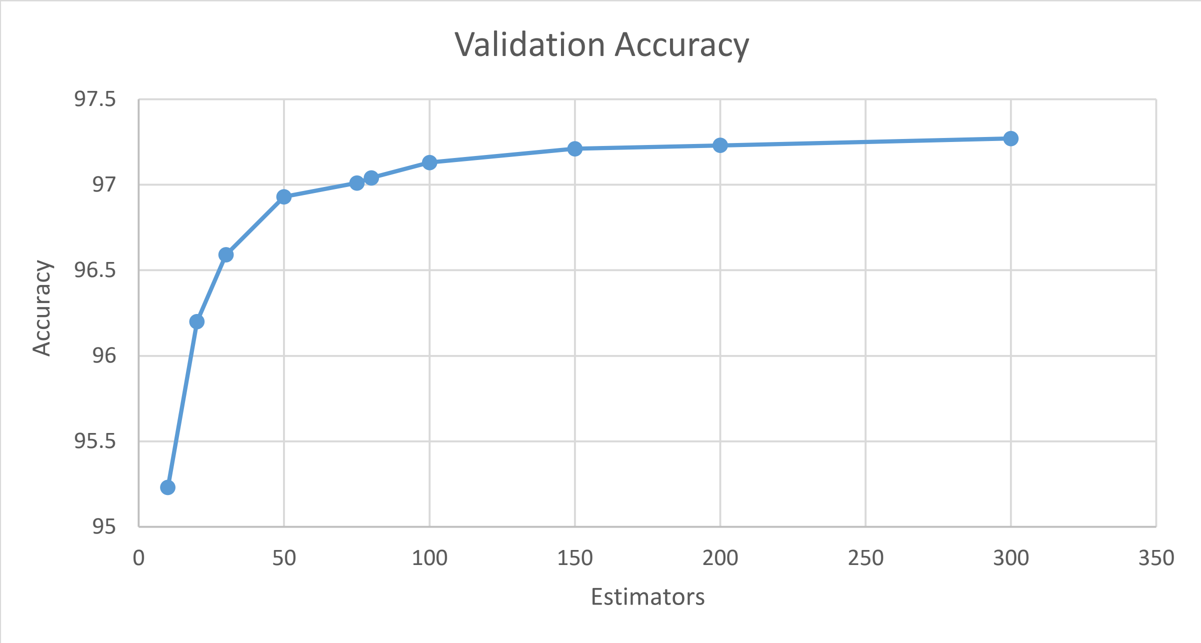

Fig 5: Effect of number of trees on Validation Accuracy

Estimators	Validation Accuracy
10	95.23
20	96.2
30	96.59
50	96.93
75	97.01
80	97.04
100	97.13
150	97.21
200	97.23
300	97.27

Table 5: Values of Validation Accuracy on Number of Estimators

Confusion Matrix of Validation Dataset

```
[[ 979    0    3    0    0    0    2    0    5    2]
 [    0 1052    5    1    2    1    1    0    2    0]
 [    1    1  966    1    3    0    1    9    5    3]
 [    2    0    4  996    0   11    0    4    9    4]
 [    0    5    0    0  956    0    2    1    1   18]
 [    2    1    4   15    2  871   13    1    4    2]
 [    2    0    0    0    2    2  957    0    4    0]
 [    0    5   10    2    4    0    0 1060    0    9]
 [    2    4    5    4    2    5    4    0  974    9]
```

```
[   4    3    1   12    8    5    0    8    4  916]]
```

NEURAL NETWORKS

A 2 hidden layer Neural Network has been implemented with ‘relu’ activation function for the hidden layers and ‘softmax’ activation function for the final/output layer. The optimizer used was ‘adam’

Effect of Batch Size

For batch size of 32, the accuracy is highest for both validation and testing datasets. So the batch size has been set to 32 for the model.

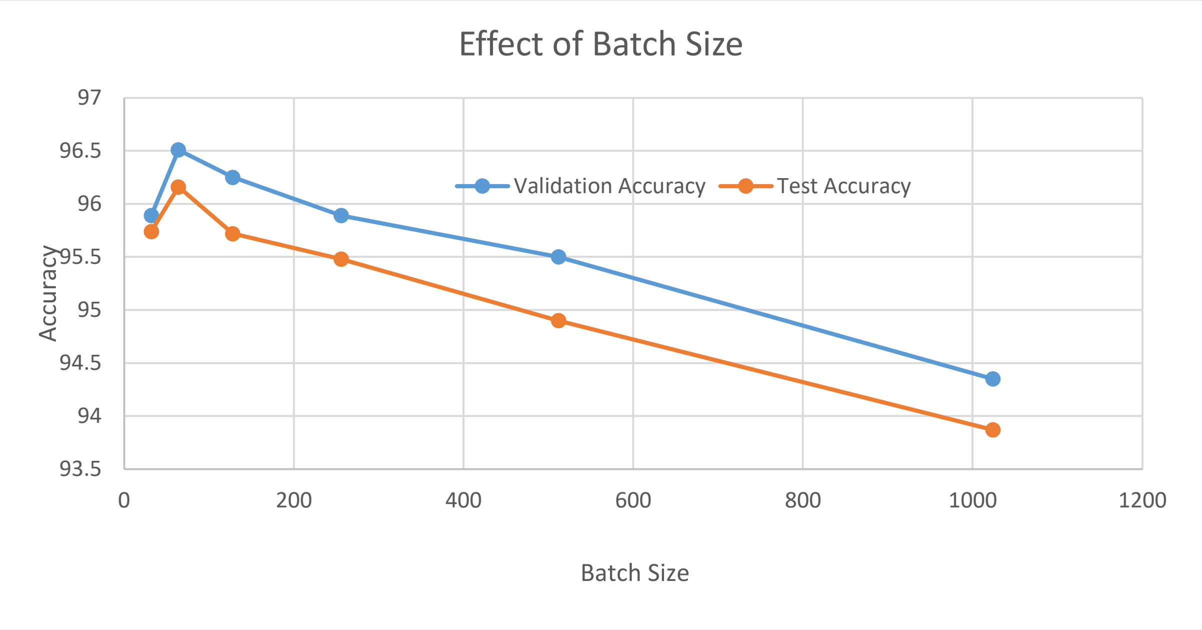

Fig 6: Effect of Batch size

Batch Size	Validation Accuracy	Test Accuracy
32	95.89	95.74
64	96.509	96.16
128	96.25	95.72
256	95.89	95.48
512	95.5	94.9
1024	94.35	93.87

Table 6: The values of Validation and Testing Accuracy for different Batch Sizes

Effect of Number of Epochs

As the epochs increase, the validation accuracy is increasing. So, the final number of epochs has been set to 100.

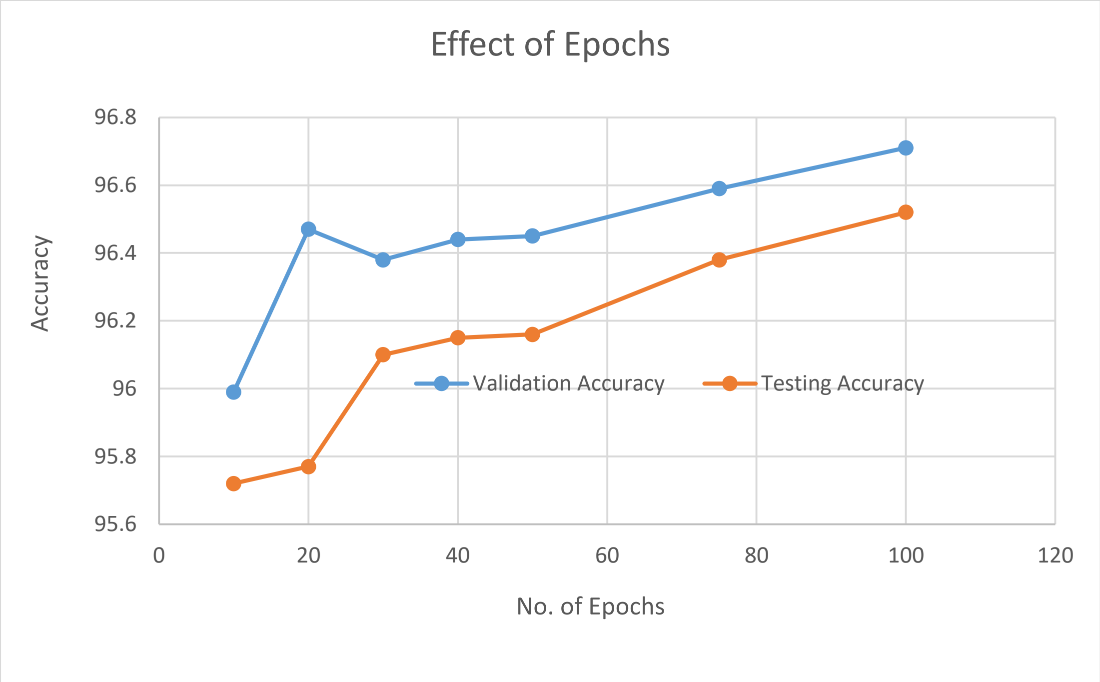

Fig 7: Effect of Epochs on Validation Accuray

epochs	Validation Accuracy	Testing Accuracy
10	95.99	95.72
20	96.47	95.77
30	96.38	96.1
40	96.44	96.15
50	96.45	96.16
75	96.59	96.38
100	96.71	96.52

Table 7: Values of Accuracy for different number of epochs

Effect of Number of Hidden Layers

As the number of hidden layer units increase, the validation set accuracy is increasing.

The other parameter values: Dropout = 0.2, epochs = 50, Batch Size = 128

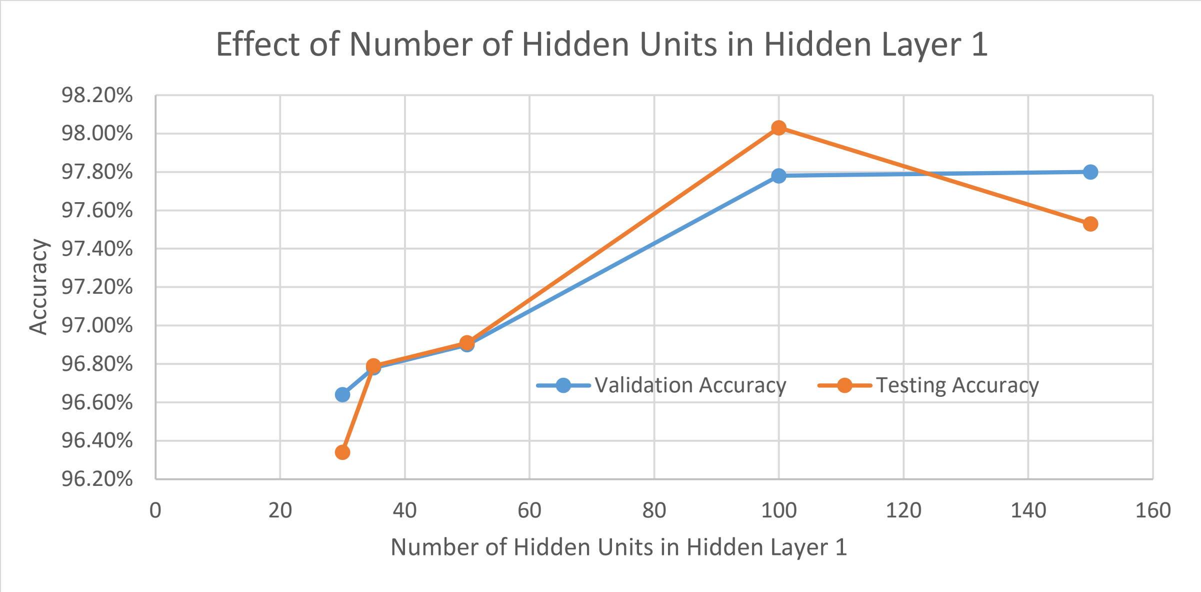

Fig 8: Effect of Number of Hidden Units in Hidden Layer 1

Hidden Layer 1	Validation Accuracy	Testing Accuracy
30	96.64%	96.34%
35	96.78%	96.79%
50	96.90%	96.91%
100	97.78%	98.03%
150	97.80%	97.53%

Table 8: Values of accuracy

Effect of Dropout

For 0.1 dropout the Validation and Testing Accuracies are highest.

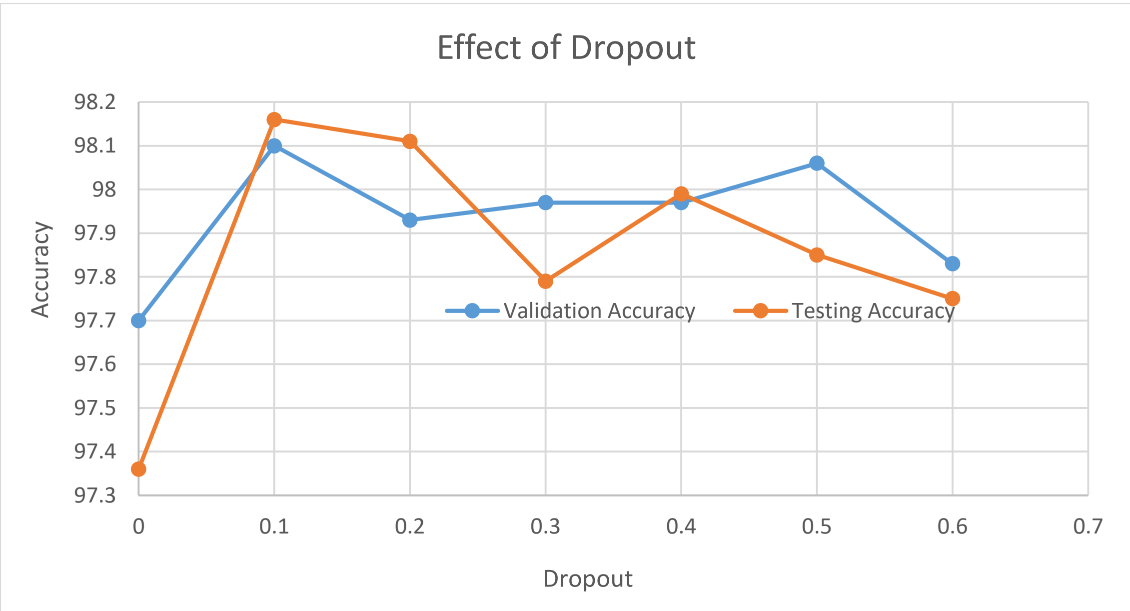

Fig 9: Effect of Dropout on Accuracy

Dropout	Validation Accuracy	Testing Accuracy
0	97.7	97.36
0.1	98.1	98.16
0.2	97.93	98.11
0.3	97.97	97.79
0.4	97.97	97.99
0.5	98.06	97.85
0.6	97.83	97.75

Table 9: Values of Accuracy for different Dropouts

Confusion Matrix of Validation Dataset

```
[[ 983    0    1    1    1    0    0    1    1    3]
 [   0 1057    1    2    0    0    1    1    2    0]
 [   2    0  974    6    0    1    1    3    3    0]
 [   0    0    3 1015    0    6    0    1    3    2]
 [   0    6    1    0  960    0    1    1    1   13]
 [   2    0    1   18    0  872    9    4    4    5]
 [   4    0    0    0    1    3  959    0    0    0]
 [   0    3    5    4    1    0    0 1071    1    5]
 [   3    5    2    3    4    4    2    1  980    5]
 [   2    3    0    7   15    3    0   10    4  917]]
```

ENSEMBLE CLASSIFIER

A combined classifier has been created in which majority voting has been implemented (i.e., the highest number of times a particular digit has been classified into) and in-case of a tie, the highest value will be taken.

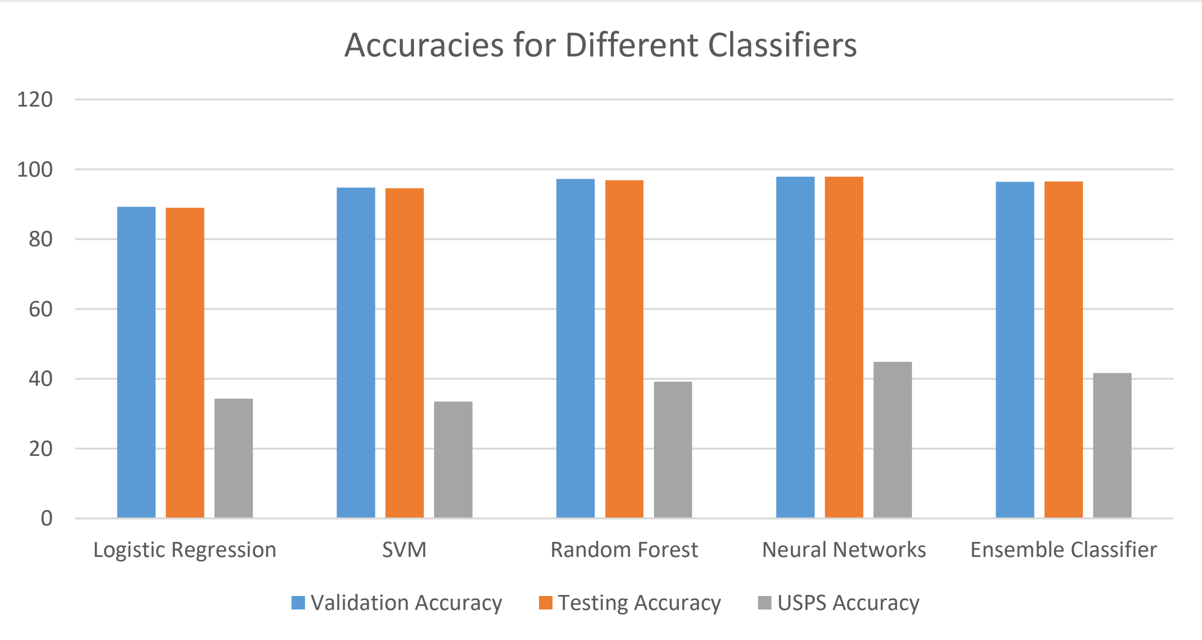

Fig 10: Accuracies for Different Classifiers

Model	Validation Accuracy	Testing Accuracy	USPS Accuracy
Logistic Regression	89.29	89.05	34.28
SVM	94.62	94.46	33.34
Random Forest	97.27	96.94	39.14
Neural Networks	97.88	97.96	44.87
Ensemble Classifier	96.82	95.78	38.49

Table 10: Values of Accuracies for Different Classifiers

Confusion Matrix of Validation Data

```
[[ 977    0    3    0    0    0    2    2    5    2]
 [    0 1053    2    1    1    2    2    0    2    1]
 [    0    0  947    2    5    1    6   12    7   10]
 [    1    1    4  965    2   23    3    5   14   12]
 [    0    6    0    0  948    0    2    2    0   25]
 [    3    1    4   14    1  851   17    4   10   10]
 [    3    0    0    0    1    2  957    1    3    0]
 [    0    5    5    2    2    0    0 1064    0   12]
 [    2    5    7    6    0    5    3    0  965   16]
 [    3    3    1   10    8    4    0   10    2  920]]
```

QUESTIONS AND ANSWERS

1. We test the MNIST trained models on two different test sets: the test set from MNIST and a test set from the USPS data set. Do your results support the “No Free Lunch” theorem?

A. Yes, the results do prove the ‘No Free Lunch’ theorem. The model that performs well on certain dataset doesn’t work on all the other datsets. Our models have been trained on the MNIST dataset and for the USPS dataset, the model predictions are very poor and the accuracy is around 35%.

B. Observe the confusion matrix of each classifier and describe the relative strengths/weaknesses of each classifier. Which classifier has the overall best performance?

A. Random Forest is performing best among all the classifiers.

C. Combine the results of the individual classifiers using a classifier combination method such as majority voting. Is the overall combined performance better than that of any individual classifier?

A. The combined classifier is performing (measured using accuracy) better than Logistic and SVM but the performance is less when compared to Random Forest and Neural Networks.