MACHINE LEARNING

Assignment 5

(skariyat-reddyak-vaismuku)

Nearest Neighbor

The program was run over a training data set of size 943.

Training Data Set size	**Accurately Predicted**				**Program accuracy**	**Execution Time (in minutes)**
	0	**90**	**180**	**270**		
0-5000	100	130	111	145	51.537%	2.90
0-10000	98	113	111	144	51.537%	5.80
0-20000	105	142	98	151	52.598%	11.42
0-36976	109	141	104	155	53.976%	21.09
Random 5000	130	145	126	149	58.324%	4.32

From the above it can be seen that the program accuracy increases marginally and the executio time increases drastically.

The random 5000, is a analysis where I considered 5000 continuous training data set within the range of 0-36976 (total number of training data sets) and there was an increase in the accuracy to 58.32% with an execution time of 4.32.

Classification Comparison:

Images that were predicted correctly:

Image	**Actual Rotation and Predicted**
test/9412628410.jpg	270
test/9391166393.jpg	180
test/9387948188.jpg	0
test/9197794520.jpg	90

Images that were predicted wrongly:

Image	Actual Rotation	Predicted
test/8383407552.jpg	180	270
test/81827804.jpg	0	90
test/7958500172.jpg	180	0
test/7847764.jpg	270	180

Adaboost:

The program was run on varied dataset size and varied number of decision stumps. The running time of the algorithm is < 5 min for the given inputs. The observations are listed below

Test data set size: 943

Training data set size: 36267

Number of decision stumps	Accuracy
5	50.15%
10	52.17%
20	56.62%
30	63.62%
40	61.93%
60	60.76%
70	64.05%
120	63.41%

As it can be seen above that the accuracy has increased gradually as the number of decision stumps were increased. On an average, accuracy of 62-63% was achieved using 70 decision stumps

Training set size: 18000

Number of decision stumps	Accuracy
20	58.20%
40	59%
80	62%
120	61.71%

Training set size: 9000

Number of decision stumps	Accuracy
20	55.35%
40	61.05
80	59.50%
120	60.97%

The above results were obtained after considering lesser training exemplars. Although it did not have a great impact on accuracy, we could achieve consistent accuracy with more training examples and more decision stumps

Classification Comparison:

Correctly classified images:

Image	Actual Rotation and Predicted
test/3933431055.jpg	270
test/3753608249.jpg	270
test/535419256.jpg	180
test/3601430993.jpg	0

Images that were predicted wrongly:

Image	Actual Rotation	Predicted
test/5762892453.jpg	90	270
test/9151723839.jpg	90	0
test/9151723839.jpg	90	0
test/27260178.jpg	180	0

Neural Network:

This program runs on different sets of hidden neurons we have considered 2 neurons. The other parameters that affected the network were, epoch and learning rate. Different values on these three parameters were tested and the running time and accuracy were analyzed. But we couldn't check for higher values for the number of hidden nodes as they consumed a lot of time. We have implemented the network with a sigmoid function as the activation function. We tried to implement the same with hyperbolic tangent as well but we did not observe a significant difference in the accuracy. The accuracy for the classification was 25.89%. This was checking over the entire and subsets of the test data.

The observations are listed below

Test data set size: 943

Training data set size: 36267

Classification Comparison:

The below results are based on training the network with the given data and also the subset of the given input.

Correctly classified images:

Image	Actual Rotation and Predicted
test/10161556064.jpg	270
test/10351347465.jpg	270
test/10353444674.jpg	270
test/10008707066.jpg	0

Images that were predicted wrongly:

Image	Actual Rotation	Predicted
test/10931472764.jpg	0	270
test/11057679623.jpg	0	270
test/11185093106.jpg	90	270
test/10161556064.jpg	270	0