COL 774: Assignment 2

Submitted By - Mahima Manik (Entry number - 2017MCS2093)

**Link to drive** - https://drive.google.com/open?id=12WxWrzyDxYt7mXTOynSx12wtndJwj6x6

Q.1

1. Time for training on the training dataset, i.e. forming vocabulary, calculating the parameters etc on dataset of size 25,000 examples = 6 seconds

Length of Vocabulary = 74629 words

Time for calculating accuracy on the test data = 20 seconds

Accuracy on Training data = 67.636 %

Accuracy on Testing data = 38.636 %

Steps Taken for preprocessing:

Regex was used to remove any punctuation marks and replacing it with blank character.

1. Randomly assigning the class labels

Training Accuracy = 12.5 %

Testing Accuracy = 12.376 %

Maximum occuring rating = 1

Training accuracy from max prediction Accuracy = 20.4 %

Testing accuracy from max prediction Accuracy = 20.088 %

Our algorithm gives around 20% improvement over the random/majority baseline.

1. Confusion Matrix for the Training Data:

[4798, 1042, 809, 702, 280, 275, 170, 321]

[10, 932, 8, 11, 4, 4, 4, 2]

[53, 59, 1294, 50, 21, 14, 7, 6]

[79, 82, 86, 1637, 52, 30, 17, 14]

[15, 26, 46, 48, 1407, 51, 47, 70]

[52, 38, 55, 72, 187, 1830, 167, 174]

[10, 5, 5, 8, 23, 22, 951, 37]

[83, 100, 117, 168, 522, 783, 900, 4108]

Confusion Matrix for the Testing Data:

[4244, 1575, 1331, 988, 351, 364, 281, 652]

[75, 59, 61, 44, 7, 9, 3, 11]

[156, 177, 228, 247, 76, 53, 21, 40]

[215, 235, 419, 554, 213, 122, 57, 69]

[40, 42, 112, 193, 320, 237, 108, 131]

[70, 64, 155, 304, 603, 749, 505, 559]

[14, 9, 19, 31, 54, 90, 72, 117]

[208, 141, 216, 274, 683, 1226, 1297, 3420]

1. Preprocessing Time = 2 seconds

Length of Vocabulary = 59845 words

Accuracy on Training data = 69.676 %

Accuracy on Testing data = 38.864 %

Time for calculating accuracy on the test data = 10 seconds

Time for calculating accuracy on the training data = 10 seconds

1. By treating two consecutive word as bigrams (building on part a):

Length of Vocabulary: 1379976

Preprocessing time: 23.7497360706

Training: Accuracy 90.424 %

Testing Accuracy 37.548 %

Prediction time: 23.6302101612

By treating two consecutive word as bigrams (building on part d):

Length of Vocabulary: 1570193

Preprocessing time: 17.7397072315

Training Accuracy 99.608 %

Testing Accuracy 38.328 %

Prediction time: 9.75174713135

Q. 2

1. Done with part b itself.
2. Training Accuracy: 86.305 %

Testing Accuracy: 87.1 %

Total Time 24.8217132092

1. Linear Model

Training Accuracy = 98.785% (19757/20000)

Testing Accuracy = 92.76% (9276/10000)

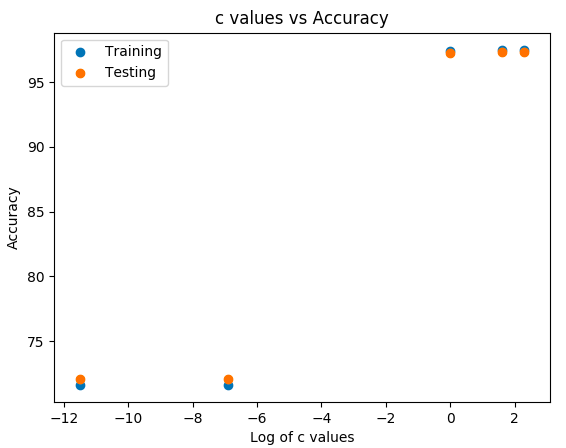
Gaussian Model

Training Accuracy = 99.92% (19984/20000)

Testing Accuracy = 97.22% (9722/10000)

We noticed that accuracy increased significantly as compared to Pegasos implementation in both linear and gaussian model. Pegasos solves the primal form of the SVM objective function whereas LIBSVM uses SMO algorithm which is used to solve the dual SVM objective function. This can be argued that in Pegasos, we put the convergence criteria on the number of iterations, which we took small number in the range 10-15. If we increase the number of iterations to very high number, then the dual and primal will give the same accuracies.

1. Best training set accuracy is obtained for c=5, i.e. 97.515%. Yes, testing accuracy is also maximum for c=5, i.e. 97.29% (also same for c=10).



|  |  |  |
| --- | --- | --- |
| C | Training Accuracy | Testing Accuracy |
| 0.00001 | 71.49% | 72.1% |
| 0.001 | 71.52% | 72.1% |
| 1 | 97.475% | 97.23% |
| 5 | 97.515% | 97.29% |
| 10 | 97.51% | 97.29% |

We observe that on very small values of C, the accuracy decreases. The accuracy at c = 5 and c = 10 are pretty close for both training and testing.

E. Actual values are taken along columns and predicted values are taken along rows.

0: [[969, 0, 4, 0, 1, 2, 5, 1, 4, 4],

1: [0, 1122, 0, 0, 0, 0, 4, 4, 0, 4],

2: [1, 3, 1000, 8, 4, 3, 0, 20, 3, 3],

3: [0, 2, 4, 985, 0, 6, 0, 2, 10, 8],

4: [0, 1, 2, 0, 962, 1, 3, 3, 1, 9],

5: [3, 2, 0, 4, 0, 866, 4, 0, 5, 4],

6: [4, 2, 1, 0, 5, 7, 940, 0, 3, 0],

7: [1, 0, 6, 7, 0, 1, 0, 986, 3, 9],

8: [2, 2, 15, 5, 2, 5, 2, 2, 942, 11],

9:[0, 1, 0, 1, 8, 1, 0, 10, 3, 957]]

The class which is most difficult to classify is 9 because its percentage of misclassified examples are higher than any other class label.

3 mis-classified examples are

Calculated Actual

0 4

4 7

8 9