Machine Learning- COL774
Assignment 2

Ankesh Gupta
2015CS10435

Naive Bayes Classifier

Observations:

1. The test accuracy obtained on random prediction=12.50%. This is emperical averaged over 10 runs. This could well be guessed, as randomly, hitting a correct class during prediction has $probability = \frac{1}{8}$.
2. The test accuracy obtained on majority prediction=20.08%.
3. The test accuracy obtained by our Naive Bayes classifier=38.4%.
4. Our algorithm causes an $\approx$ 26% increase over random baseline and $\approx$ 18.5% increase over majority baseline.
5. Label 1 has highest value of diagonal entry. This means that this maximum correct predictions were made corresponding to this category.
6. Another category, that shines similar to Label 1 is Label 10, with 2^{nd} highest correct predictions.
7. Studying the matrix, we realise that predictions for classes {2,3,4} got heavily biased towards class 1. Similar is the situation with classes {7,8,9} which tilted towards 10.
8. The reason for this bias could be the ***initial class imbalance*** that our training data suffers from.
9. Again, the accuracy obtained was 38.4%. No significant gains were observed. This might be because initial data cleaning might have removed very much the noise in the data. The class imbalance still remains.
10. The bottleneck that algorithm is facing is loss of context so a negated good word is getting interpreted as good word and that is the root of the problem.
11. One of the feature engineering was to use combination of ***unigrams and bigrams*** along with data augmentation and thresholding. Duplicates of data were added to training data to reduce class imbalance.
12. Accuracy obtained using this method was 38.9%.
13. Another method tried was ***TF-IDF*** in which experiments were repeated with and counts of words were replaced by their weights. The model again landed up showing accuracy 37% accuracy when no augmentation was made, and 38% accuracy with single augmentation.

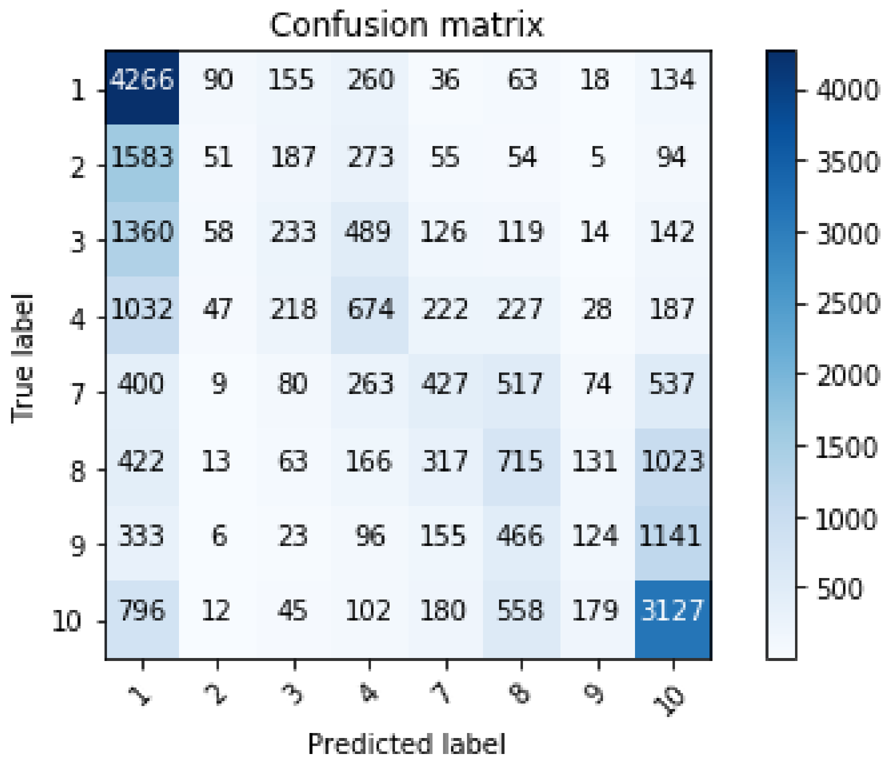


Figure 1: Confusion Matrix without Stemming

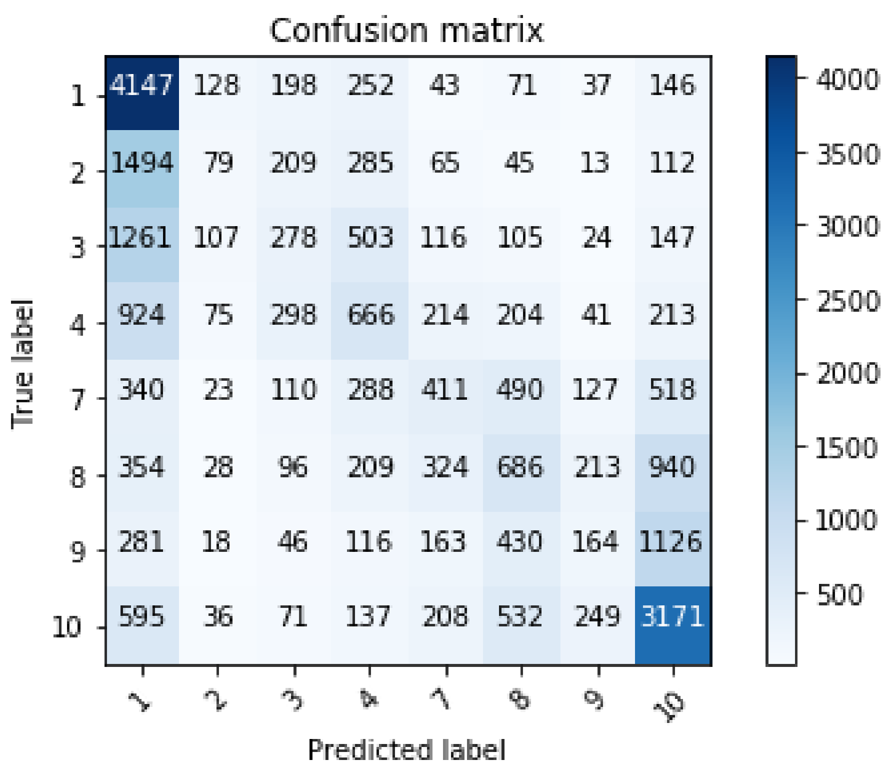


Figure 2: Confusion Matrix with Stemming and Stopwords removed

14. ***Bigram Model*** turns out better because it in a sense includes partial context of word. One may argur that this logic would be better for $n - grams$ for $n > 2$ but out input space grows exponentially, and hence the features become extremely sparse. If we had large data, probably $n - grams$ would out-perform.

Support Vector Machine(SVM)

Observations:

1. Test Accuracy obtained was = 92.54%. Train Accuracy was = 94.1%.
2. Accuracy in case of Linear Kernel = 92.78%. Accuracy with Gaussian Kernel = 97.23%.
3. Our implementation SVM compare fairly well with the inbuilt version. Their implementation attains a meagre 0.24% improvement compared with our's. That could be taken care if we change the number of iterations for a classifier's convergence. Current limit= 2000 iterations.
4. Best value of $C = 10$. Actually $C = 5$ and $C = 10$ both gives exact same results, but $C = 10$ ran a bit faster.
5. This value of C also outperforms on the test set. Here is a tabulated form of above findings.

C	Validation(%)	Test(%)
10^{-5}	71.59	72.11
0.001	71.59	72.11
1	97.355	97.23
5	97.455	97.29
10	97.455	97.29

6. On a subtle note, analyzing the above scenario we realize that increasing C is causing our classifier to classify more accurately.
7. When C is low, it indicates classifier that you are allowed to ***misclassify***, but the ones you classify correct much have large margin.
8. As we increase C, it indicates to the classifier that our focus is now to *classify more and more points accurately*, rather than separating the classes well.
9. Class 9 faces most difficulty in classification as it has most number of misclassifications.
10. On visualizing, we see noisy 9 is quite similar to 4 and 7 at times, and it is natural for our model to breakdown to such granularity.
11. Besides, 2 and 7 are the pair which confuses each other a lot.
12. Visualization also shows how 2 and 7 our inter twined in some scenario and it becomes difficult even for human vision to distinguish unambiguously.

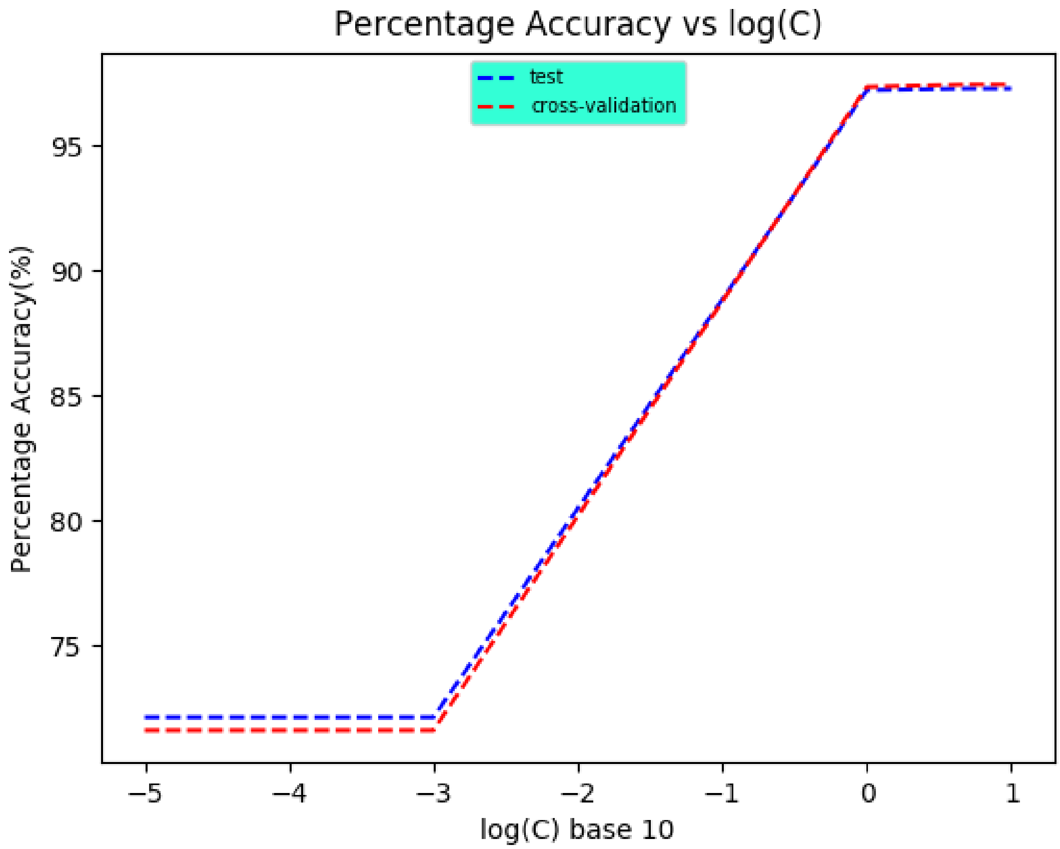
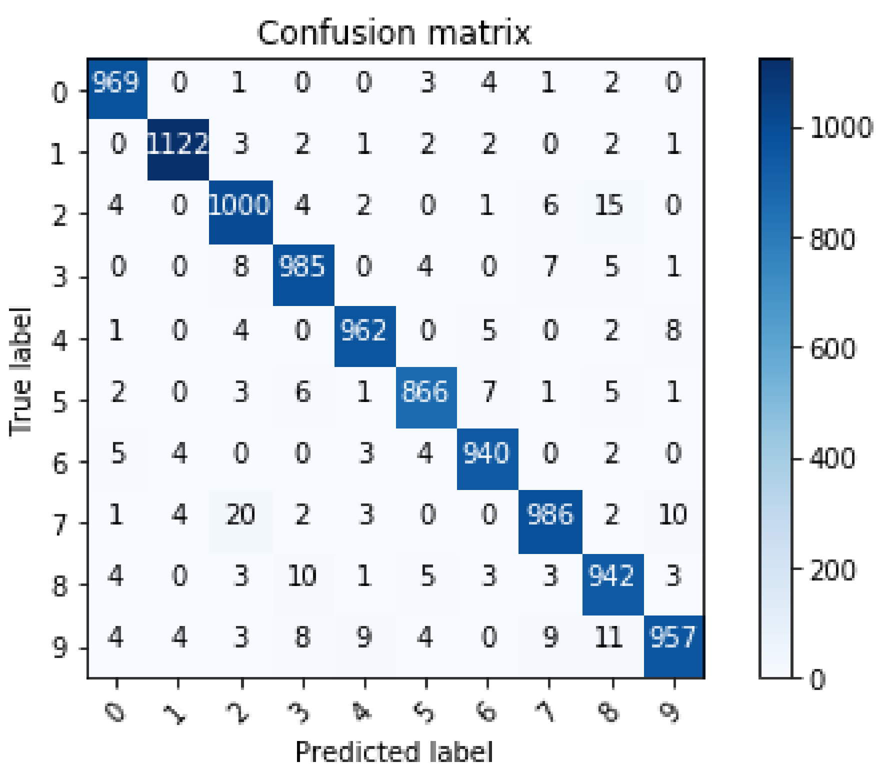


Figure 3: Accuracy for CrossValidation and Test



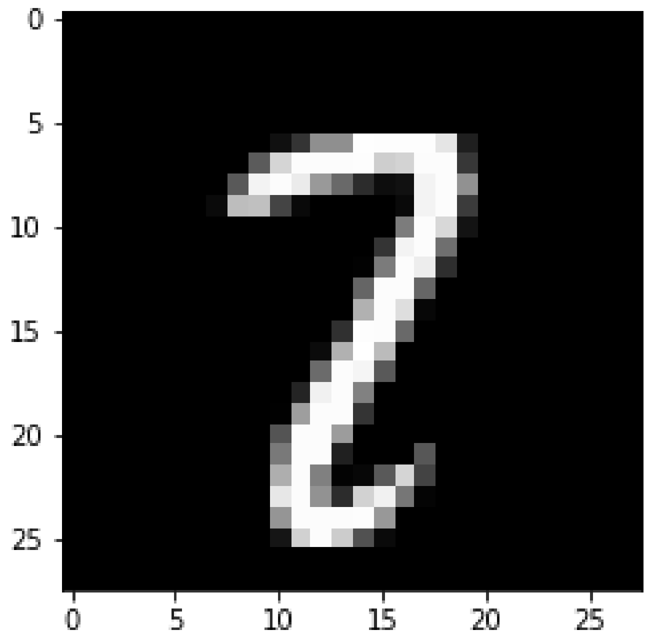


Figure 4: Correct-2 Pred-7

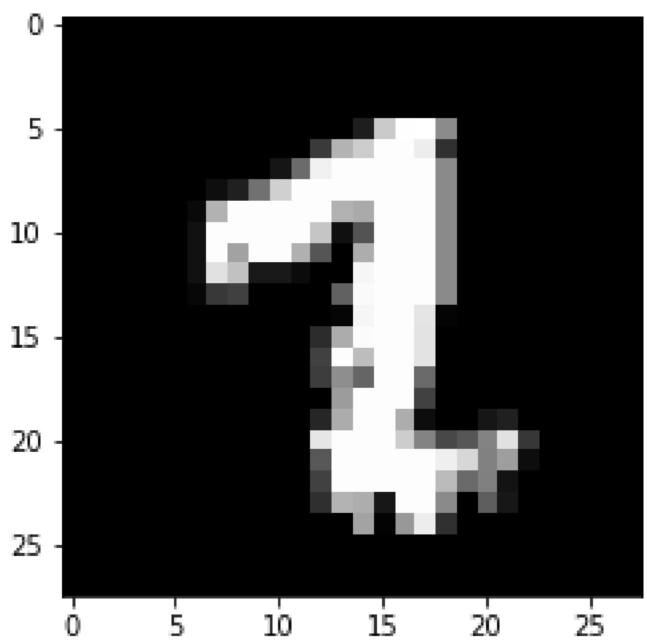


Figure 5: Correct-2 Pred-7

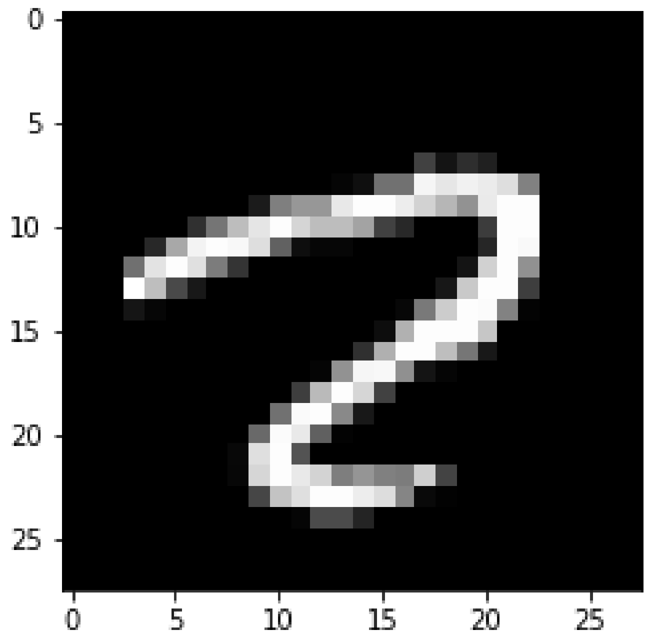


Figure 6: Correct-2 Pred-7

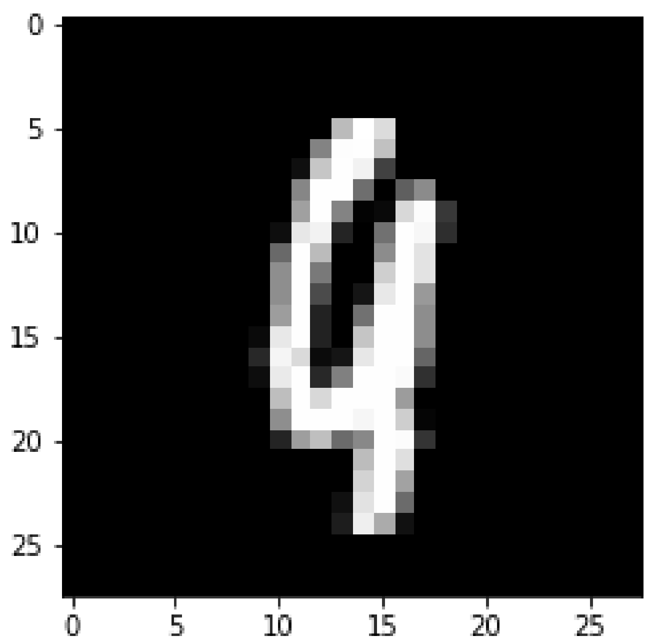


Figure 7: Correct-9 Pred-4

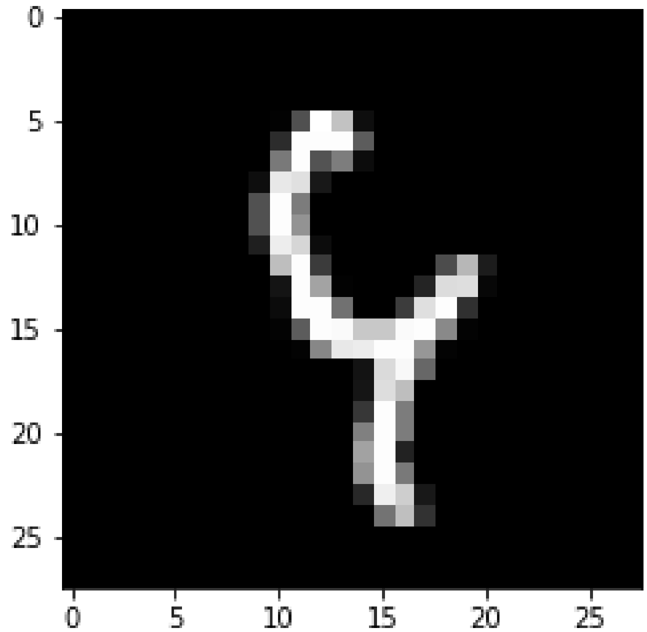


Figure 8: Correct-9 Pred-4

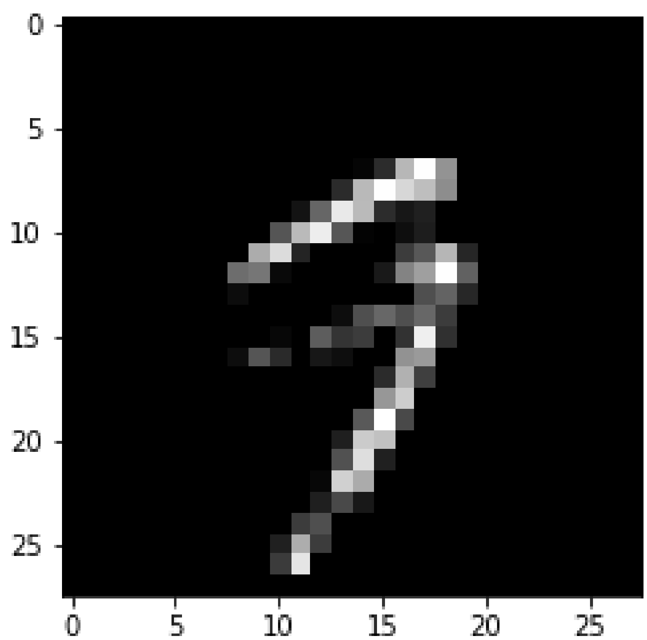


Figure 9: Correct-9 Pred-7

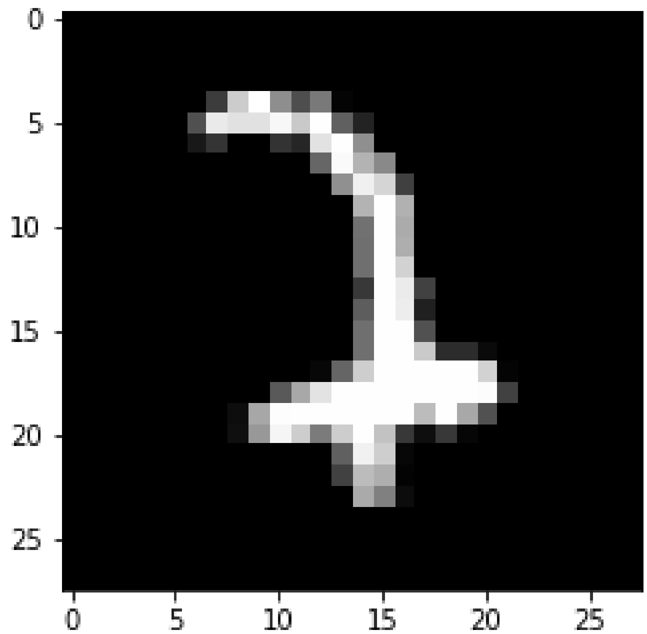


Figure 10: Correct-7 Pred-2

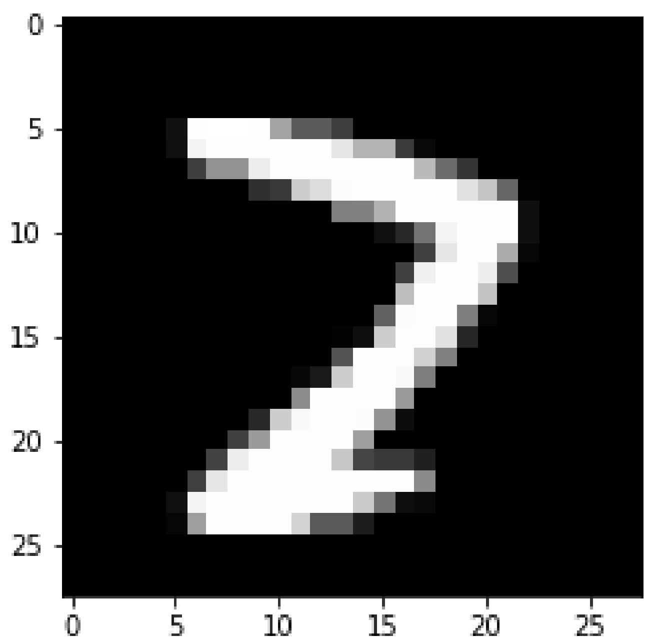


Figure 11: Correct-7 Pred-2