

## Assignment 2 Report

Name: Manish Tanwar

Entry Number: 2016CS10363

## 1 Naive Bayes:

## (a) Accuracy over test and training dataset:

- Accuracy over test dataset: **60.4 %**
- Accuracy over training dataset: **64.6 %**

## (b) Accuracy using Random and Majority Prediction:

- Accuracy using Random Prediction: **19.9 %**
- Accuracy using Majority Prediction: **44.0 %**

## (c) Confusion Matrix:

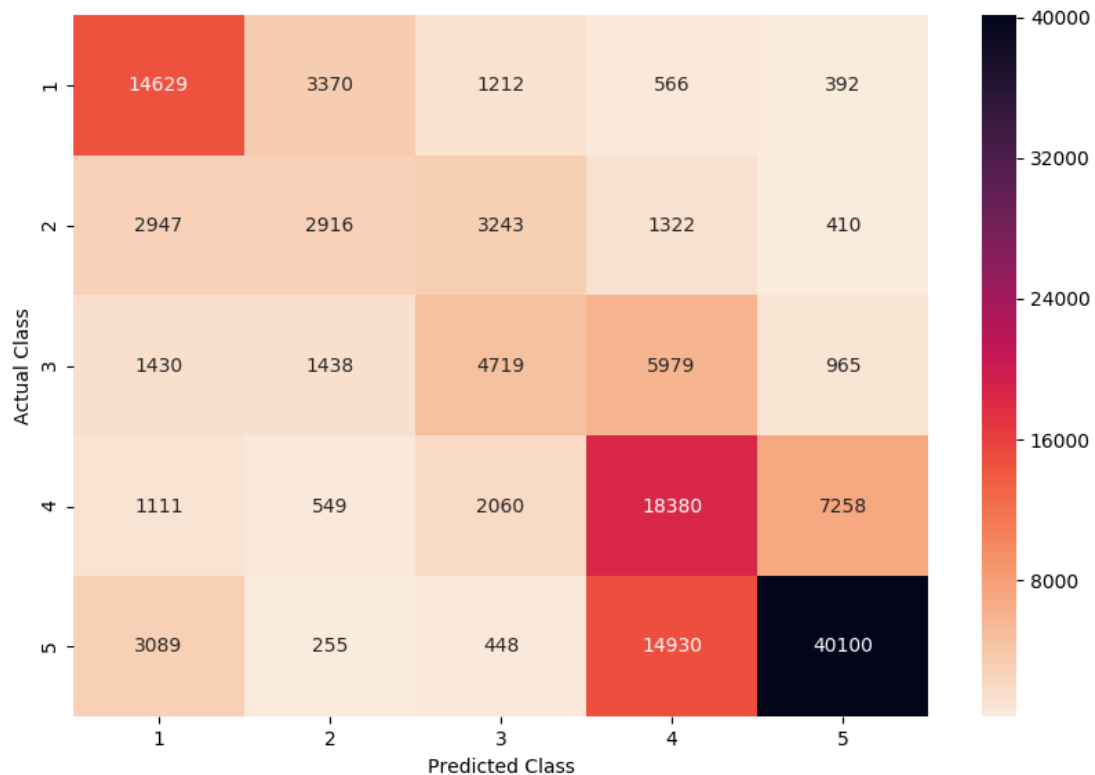


Figure 1: Confusion Matrix

- Diagonal has the most elements which represents the correct predictions.
- Confusion matrix also shows the correlation between adjacent ratings.(For example for actual class 5, it has most entries in 5 but it has second most entries in 4).
- The number of elements decreases as we move away from the diagonal which shows clear correlation.

**(d) Stemming and Stopword Removal:**

- Accuracy Obtained: **60.0 %**
- Accuracy does not change much on performing stemming and stopwords removal.

**(e) Feature Engineering:**

- Accuracy Obtained: **63.4 %**
- On adding bigram feature on top of our model, accuracy increases by **3-4%**.
- Using bigrams feature introduces dependencies among words which increases the accuracy.

**(f) F1-Score:**

- F1 Score : [0.7193 0.2362 0.2805 0.5231 0.7821]
- Less F1 score for class 2 and 3 shows less accuracy for these classes.
- Macro F1 Score : **0.5083**
- Macro F1 score is a better metric, especially in uneven class distribution model as it considers both precision and recall to compute the score.

**(g) Training of Full Dataset:**

- Accuracy Obtained: **73.15%**
- F1 Score : [0.7645 0.5130 0.5901 0.6533 0.8336]
- Macro F1 Score : **0.6709**
- Increase in training data helps in better model learning and prediction.

## **2 Support Vector Machine:**

### **2.1 Binary Classification:**

**(a) CVXOPT with Linear Kernel:**

- Accuracy Obtained: **99.49%**
- Indices of Support Vectors, Weights( $w$ ) and bias( $b$ ) are submitted in file "linear\_results.txt"
- Number of Support Vectors: 134

**(b) CVXOPT with Gaussian Kernel:**

- Accuracy Obtained: **99.89%**
- Indices of Support Vectors are submitted in file "gaussian\_results.txt"
- Number of Support Vectors: 1386

**(c) LIBSVM Package with Linear and Gaussian Kernels:****Linear Kernel:**

- Accuracy Obtained: **99.49%**
- Indices of Support Vectors, Weights( $w$ ) and bias( $b$ ) are submitted in file "libsvm\_linear\_results.txt"
- Number of Support Vectors: 134

**Gaussian Kernel:**

- Accuracy Obtained: **99.89%**
- Indices of Support Vectors are submitted in file "libsvm\_gaussian\_results.txt"
- Number of Support Vectors: 1344

**Computational Cost Comparision:**

Training Time(in Sec)		
Kernel	CVXOPT	LIBSVM
Linear	74.82	4.59
Gaussian	35.84	8.96

- LIBSVM takes way less time than CVXOPT package.
- Using CVXOPT and LIBSVM gives almost same **number of support vectors**. Slight difference is due to floating point arithmetic comparison  $\alpha > 0$ , which is done using  $\alpha > eps$  ( $eps = 10^{-5}$ ).

**2.2 Multi-Class Classification:****(a) Using CVXOPT Package:**

- Accuracy over test dataset: **97.24 %**
- Accuracy over training dataset: **99.92 %**

**(b) Using LIBSVM Package:**

- Accuracy over test dataset: **97.23 %**
- Accuracy over training dataset: **99.92 %**

**Computational Cost Comparision (Training time):**

- CVXOPT : 472.36 sec
- LIBSVM : 237.51 sec

## (c) Confusion Matrix:

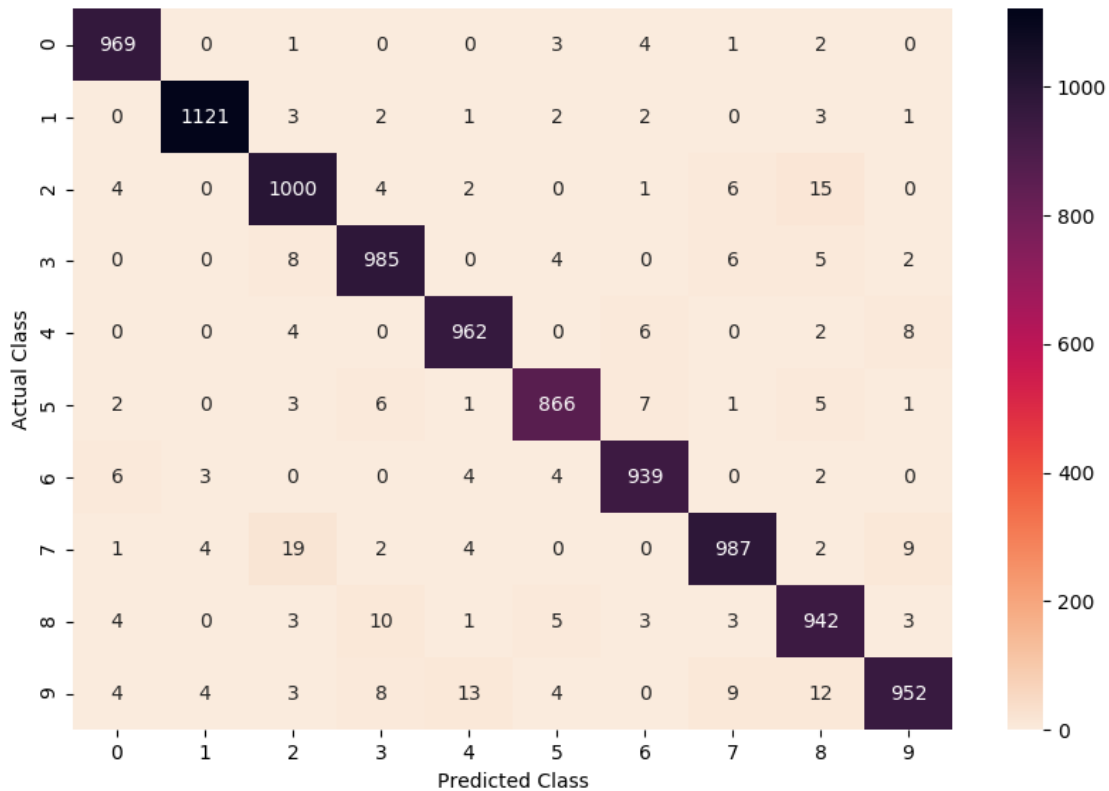


Figure 2: Confusion Matrix(drawn for part(b))

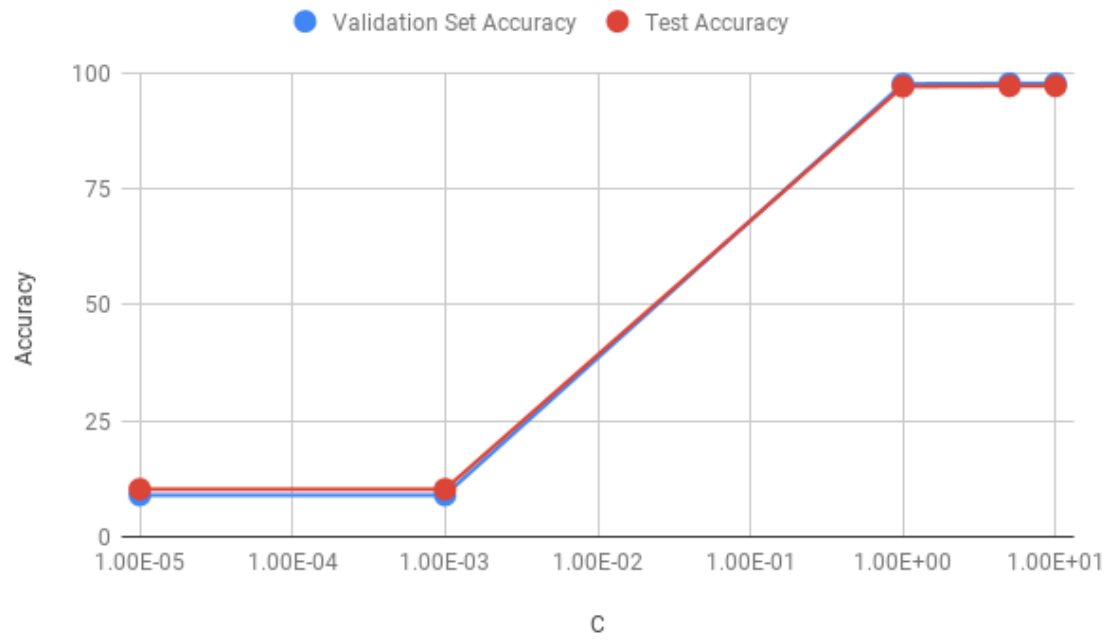
**Analysis:**

- Most often 2,7,8,9 are misclassified as 8,2,3,4 respectively.
- Misclassification of 7 as 2, 8 as 3 and 9 as 4 is due to their similarity in structural representation.

**(d) Validation:**

- We randomly select 10% data from training dataset as validation set to estimate the parameter  $C$ .
- $C$  is penalty parameter. Value of parameter  $C$  how much misclassification do we allow for finding the hyperplane.
- For  $C = 5$  and  $10$ , we obtain the best validation and test accuracy.
- For smaller  $C$ , we allow too much misclassification by imposing too low penalty which results in really low accuracy.

$C$	Validation Accuracy	Test Accuracy
$10^{-5}$	9	10.28
$10^{-3}$	9	10.28
1	97.7	97.12
5	97.8	97.24
10	97.8	97.24

Figure 3: Accuracy Vs Parameter  $C$