EE219 Project 1

Classification Analysis

Winter 2018

Fangyao Liu 204945018

Zhechen Xu 805030074

**Content**

Introduction………………………………………………………………………………….. 1

Dataset……………………………………………………………………………………….. 1

Problem Statement ………………………………………………………………………….. 1

Question (a) Histogram of The Number of Documents Per Topic

Modeling Text Data and Feature Extraction………………………………………………… 2

Question (b) TFxIDF Vector Representation

Question (c) TFxICF and 10 Most significant Terms

Feature Selection…………………………………………………………………………….. 2

Question (d) Latent Semantic Indexing(LSI) Representation of TFxIDF

Vectors

Learning Algorithms………………………………………………………………………… 3

Question (e) Linear Support Vector Machines(SVM) Method

Question (f) Soft Margin SVM Method

Question (g) Naïve Bayes Algorithm

Question (h) Logistic Regression Classifier

Question (i) Logistic Regression Classifier with Regularization

Multiclass Classification…………………………………………………………………… 22

Question (j) Naïve Bayes classification and multiclass SVM classification

# **Introduction**

In this project, we implement several classifiers such as SVM, Naïve Bayesian and Logistic Regression to classify textual data based on the 20 Newsgroup Dataset, which had 11314 documents, partitioned evenly across 20 distinct categories.

# **Dataset**

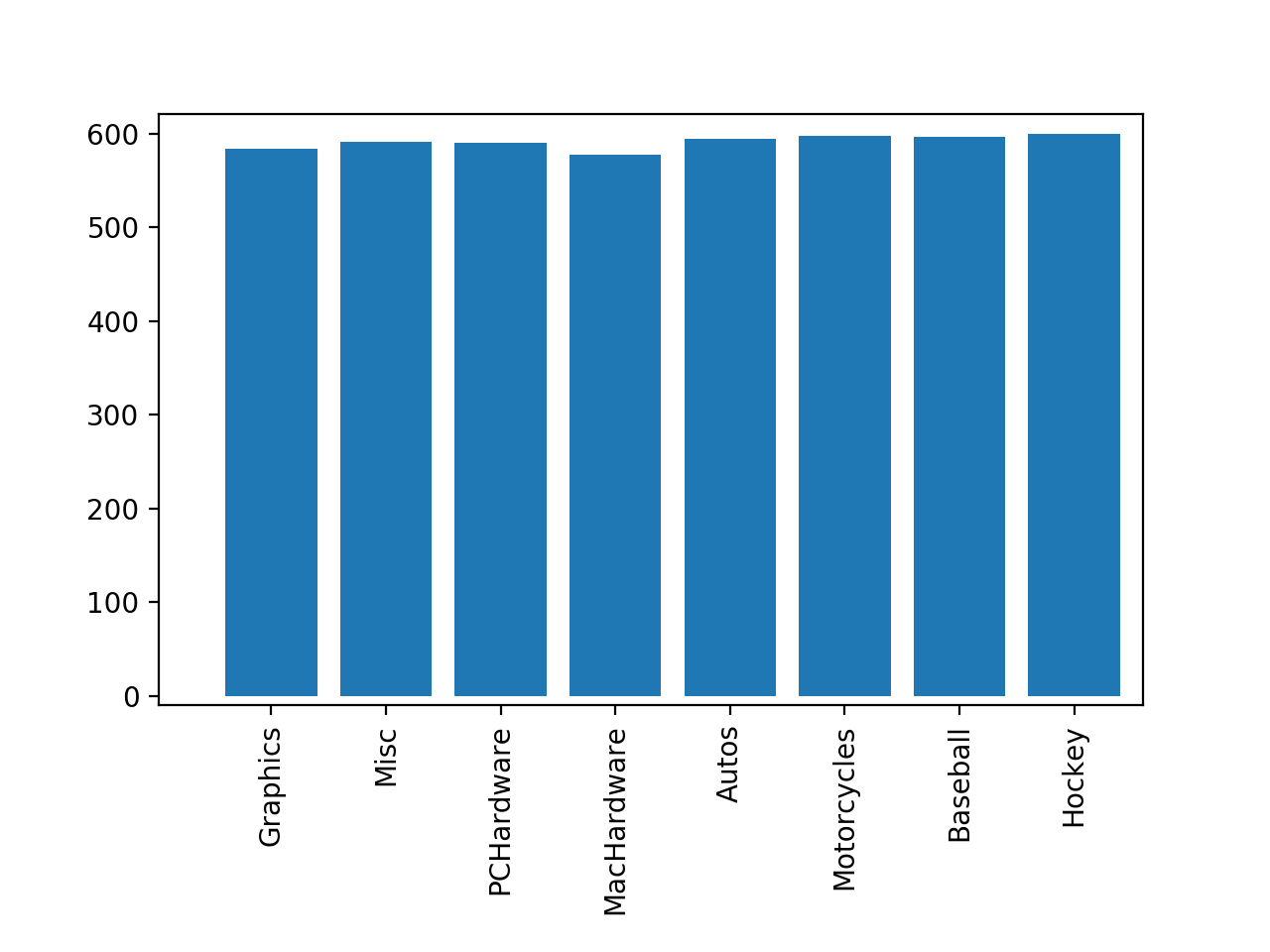
We load the 20 newsgroups dataset by using sklearn package module, which has 20 categories and each category is corresponding to a different topic.

# **Problem Statement**

We classify 20000 documents into two classes as followed

|  |  |
| --- | --- |
| Computer technology | Recreational activity |
| comp.graphics  comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware | rec.autos  rec.motorcycles  rec.sport.baseball  rec.sport.hockey |

## Question(a) Histogram of Number of Documents Per Topic



## figure 1 Histogram of Number of Documents Per Topic

As shown in figure 1, the numbers of documents per topic are evenly distributed.

**Modeling Text Data and Feature Extraction**

We use Term Frequency-Inverse Document Frequency (TFxIDF) metric to capture the most significant word to a document in a corpus.

**Question (b) TFxIDF Vector Representation**

We create a TFxIDF vector representations, tokenize the documents and exclude the stop words, punctuations, and different stems of a word.

The final number of terms (frequency 2) is 30383 and the final number of terms (frequency 5) is 12915.

**Question (c) TFxICF And 10 Most Significant Terms**

The top 10 significant terms for classes ‘comp.sys.ibm.pc.hardware’,‘comp.sys.mac.hardware’, ‘misc.forsale’, and ‘soc.religion.christian’.

|  |  |  |  |
| --- | --- | --- | --- |
| comp.sys.ibm.pc.hardware | comp.sys.mac.hardware | misc.forsale | soc.religion.christian |
| scsi | quadra | forsale | bible |
| ide | scsi | dos | christ |
| controller | centris | game | church |
| bios | nubus | condit | scripture |
| disk | simms | hiram | people |
| floppy | duo | obo | god |
| pc | fpu | cd | way |
| isa | monitor | hulk | jesus |
| bus | lc | printer | faith |
| drives | mac | pc | athos |

**Table 1 10 Most Significant Terms**

**Feature Selection:**

Due to the sparse feature and high dimensionality of TFxIDF vectors, we apply LSI decomposition to improve the performance and effectiveness of the algorithm.

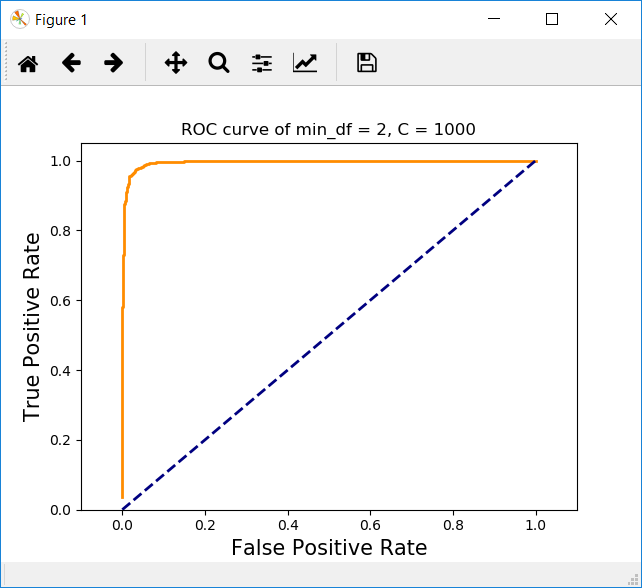
**Question (d) Latent Semantic Indexing (LSI) Representation Of TFxIDF Vectors**

We get the optimal lower dimensional matrix through using Latent Semantic Indexing and Non-Negative Matrix Factorization. The 50-dimensional vector was used to represent the TFxIDF matrices for following sections.

**Learning Algorithms**

**Question (e): Linear Support Vector Machines (SVM) Method**

In this question, we implement the SVM classifier to classify the documents into Computer Technology and Recreational Activity groups. We can tell the performance from the Receiver Operating Characteristic (ROC) curve, report the confusion matrix and the accuracy, recall and precision of the classifier.



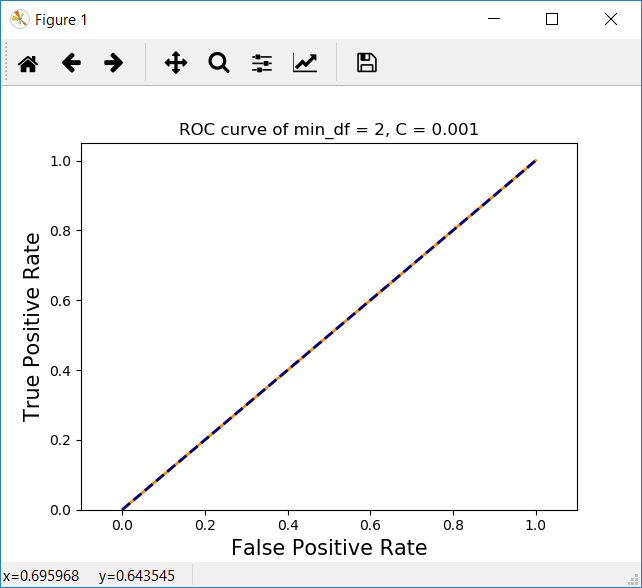
## figure 2 ROC for SVM, LSI (frequency = 2, c = 1000)

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9686 |
| Recall | 0.9780 |
| Precision | 0.9605 |

**Table 2 statics for SVM classifier (frequency = 2, c = 1000)**

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1496 | 64 |
| Actual Recreation | 35 | 1555 |

**Table 3 SVM Confusion Matrix (frequency = 2, c = 1000)**



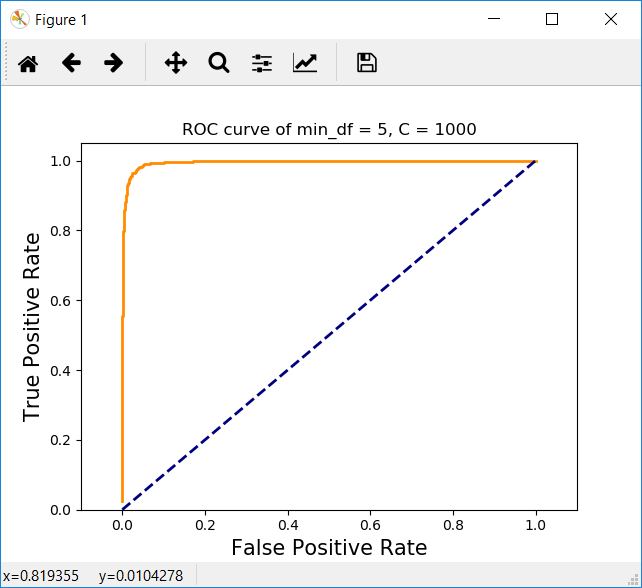
## figure 3 ROC for SVM, LSI (frequency = 2, c = 0.001)

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.5048 |
| Recall | 1.0 |
| Precision | 0.5048 |

## Table 4 statics for SVM classifier (frequency = 2, c = 0.001)

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 0 | 1560 |
| Actual Recreation | 0 | 1590 |

## Table 5 SVM Confusion Matrix (frequency = 2, c = 0.001)



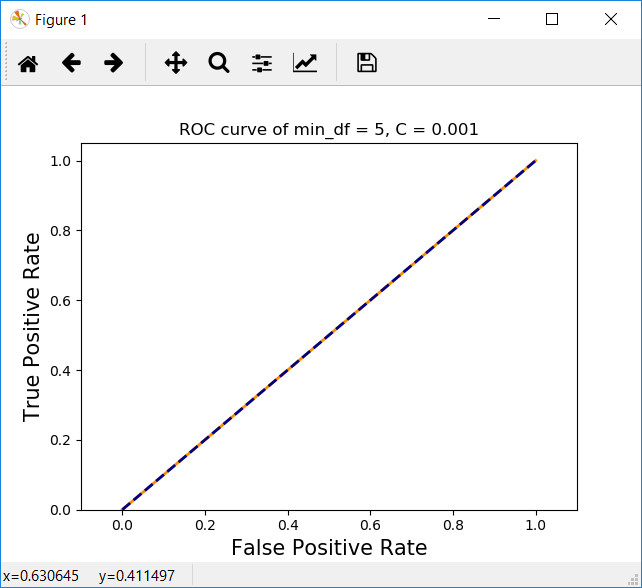
## figure 4 ROC for SVM, LSI (frequency = 5, c = 1000)

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9695 |
| Recall | 0.9761 |
| Precision | 0.9640 |

## Table 6 statics for SVM classifier (frequency = 5, c = 1000)

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1502 | 58 |
| Actual Recreation | 38 | 1552 |

**Table 7 SVM Confusion Matrix (frequency = 5, c = 1000)**



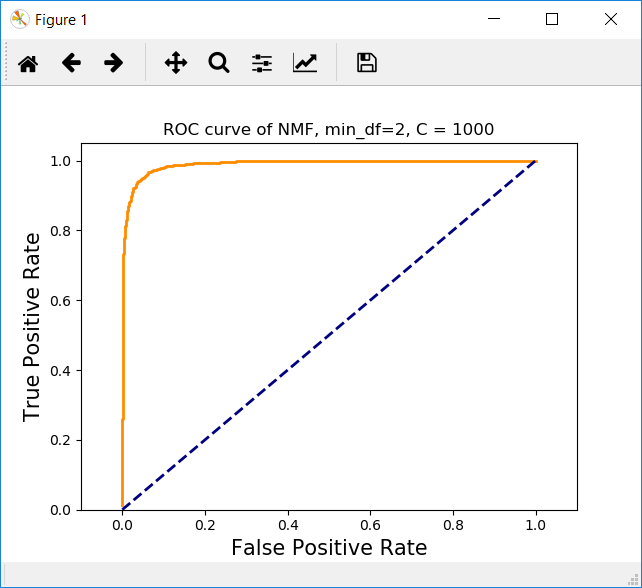
## figure 5 ROC for SVM, LSI (frequency = 5, c = 0.001)

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9695 |
| Recall | 0.9761 |
| Precision | 0.9640 |

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1502 | 58 |
| Actual Recreation | 38 | 1552 |

**Table 8 statics for SVM classifier (frequency = 5, c = 0. 001)**

**Table 9 SVM Confusion Matrix (frequency = 5, c = 0.001)**



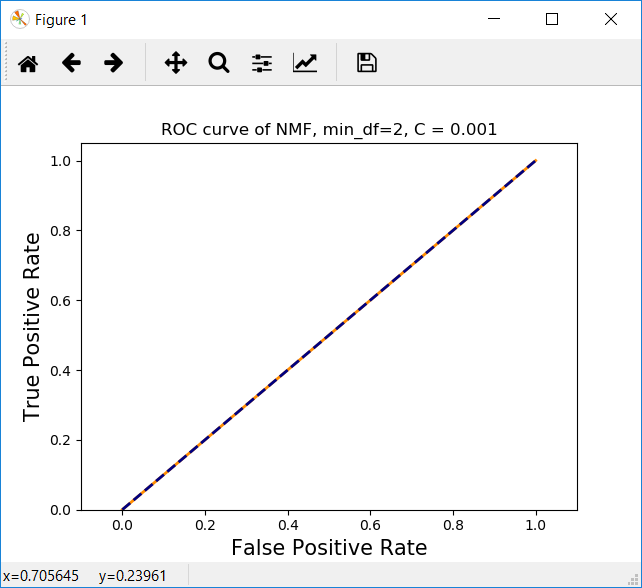
## figure 6 ROC for NMF (frequency = 2, c = 1000)

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9492 |
| Recall | 0.9679 |
| Precision | 0.9338 |

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1451 | 109 |
| Actual Recreation | 51 | 1539 |

**Table 10 statics for NMF (frequency = 2, c = 1000)**

**Table 11 NMF Confusion Matrix (frequency = 2, c = 1000)**



## figure 7 ROC for NMF (frequency = 2, c = 0.001)

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.5047 |
| Recall | 1.0 |
| Precision | 0.5047 |

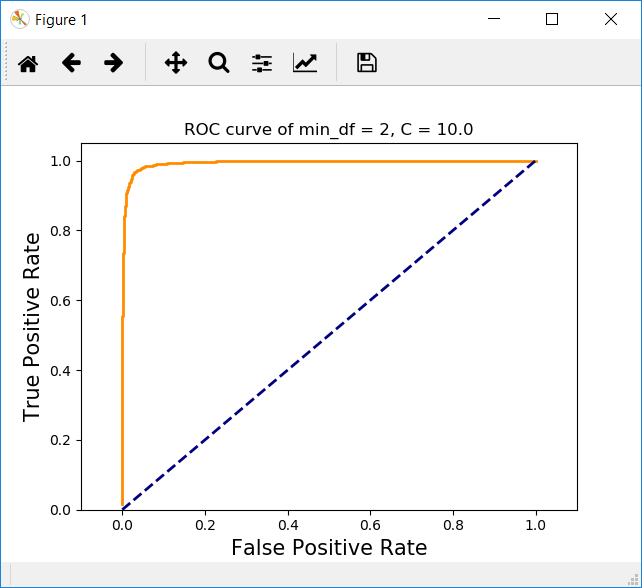
|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 0 | 1560 |
| Actual Recreation | 0 | 1590 |

**Table 12 statics for NMF (frequency = 2, c = 0.001)**

**Table 13 NMF Confusion Matrix (frequency = 2, c = 0.001)**

## Question(f) Cross-Validation

In this task, we use SVM Method to classify the documents with 5-fold cross-validation to find the optimal value of the parameter 𝛾 in the range {10−𝑘| − 3 ≤ 𝑘 ≤ 3, 𝑘 ∈ 𝑍}. And optimum c for different min\_df and factorization method are shown below.



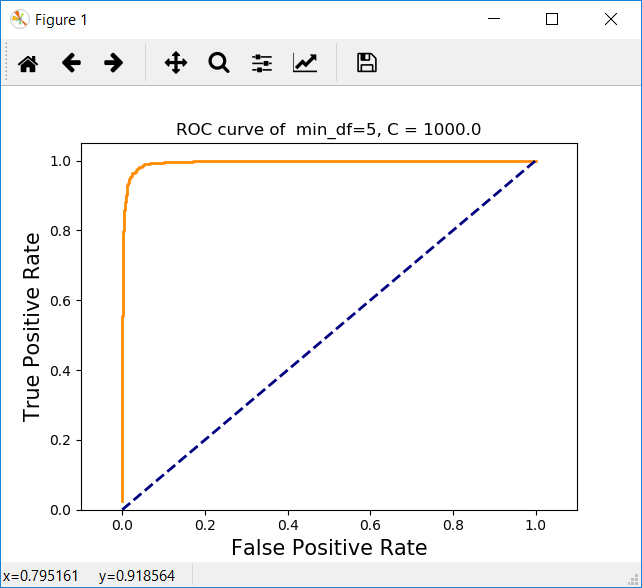
**figure 8 ROC for SVM, LSI(frequency = 2, c = 10.0)**

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9651 |
| Recall | 0.9774 |
| Precision | 0.9545 |

**Table 14 statics for SVM classifier with cross-validation (frequency = 2, c = 10.0)**

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1486 | 74 |
| Actual Recreation | 36 | 1554 |

**Table 15 SVM Confusion Matrix with cross-validation (frequency = 2, c = 10.0)**



**figure 9 ROC for SVM, LSI(frequency = 5, c = 1000)**

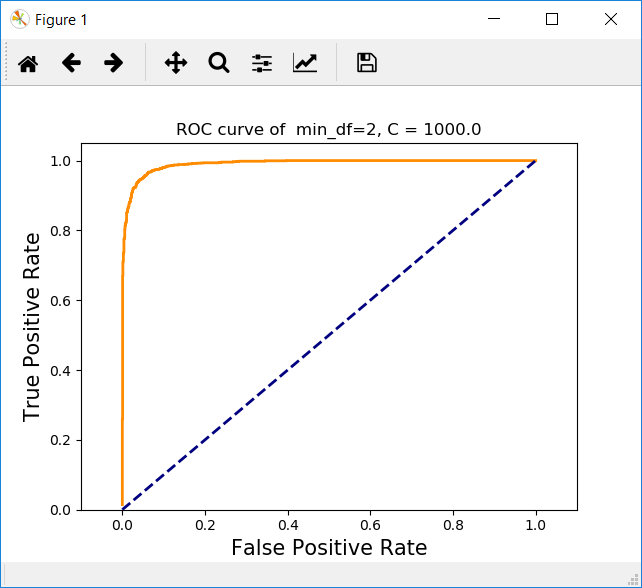
|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9695 |
| Recall | 0.9761 |
| Precision | 0.9640 |

**Table 16 statics for SVM classifier with cross-validation (frequency = 5, c = 1000)**

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1502 | 58 |
| Actual Recreation | 38 | 1552 |

**Table 17 SVM Confusion Matrix with cross-validation (frequency = 5, c = 1000)**

For NMF and min\_df = 2, we can get optimum when c = 1000:



**figure 10 ROC for NMF (frequency = 2, c = 1000)**

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9492 |
| Recall | 0.9679 |
| Precision | 0.9338 |

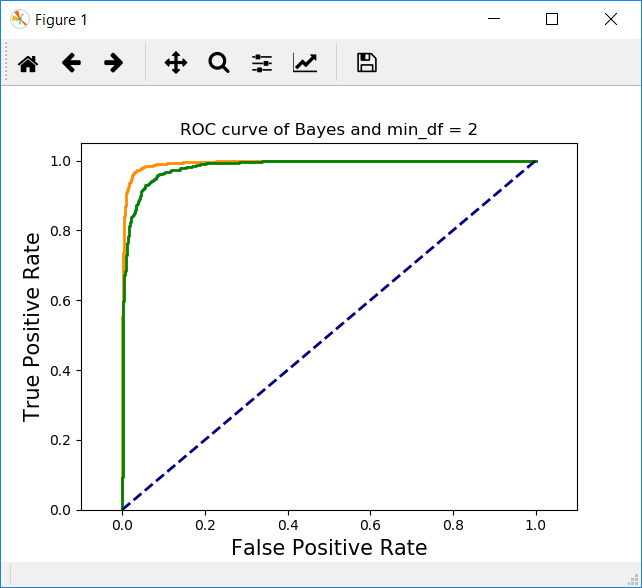
**Table 18 statics for NMF (frequency = 2, c = 1000)**

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1451 | 109 |
| Actual Recreation | 51 | 1539 |

**Table 19 NMF Confusion Matrix (frequency = 2, c = 1000)**

**Question(g) Naïve Bayes Algorithm**

Naïve Bayes Algorithm is used in this task. The algorithm estimates the maximum likelihood probability of a class given a document with feature set x, using Bayes rule, based upon the assumption that given the class, the features are statistically independent.



**figure 11 ROC for Naïve Bayes Classifier, LSI (frequency = 2)**

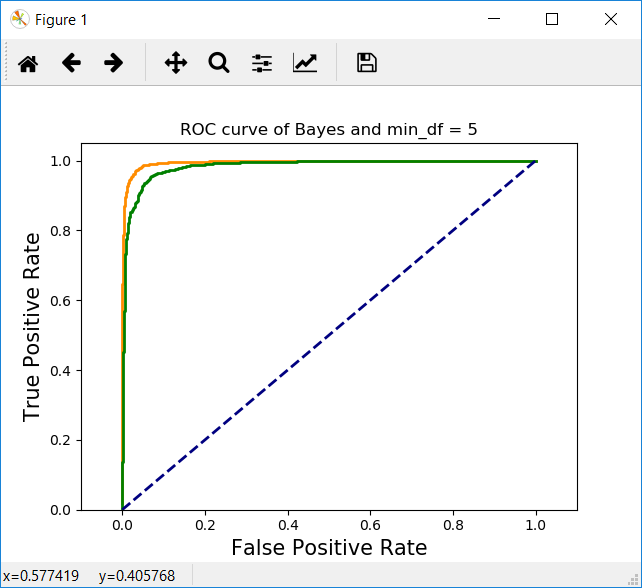
In figure 11, yellow line is ROC for SVM and green line is ROC for Naïve Bayes.

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9111 |
| Recall | 0.9830 |
| Precision | 0.8607 |

**Table 20 statics for Naïve Bayes Classifier (frequency = 2)**

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1307 | 253 |
| Actual Recreation | 27 | 1563 |

**Table 21 confusion matrix for Naïve Bayes Classifier (frequency = 2)**

****

**figure 12 ROC for Naïve Bayes Classifier, LSI (frequency = 5)**

In figure 12, yellow line is ROC for SVM and green line is ROC for Naïve Bayes.

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9247 |
| Recall | 0.9767 |
| Precision | 0.8859 |

**Table 22 statics for Naïve Bayes Classifier (frequency = 5)**

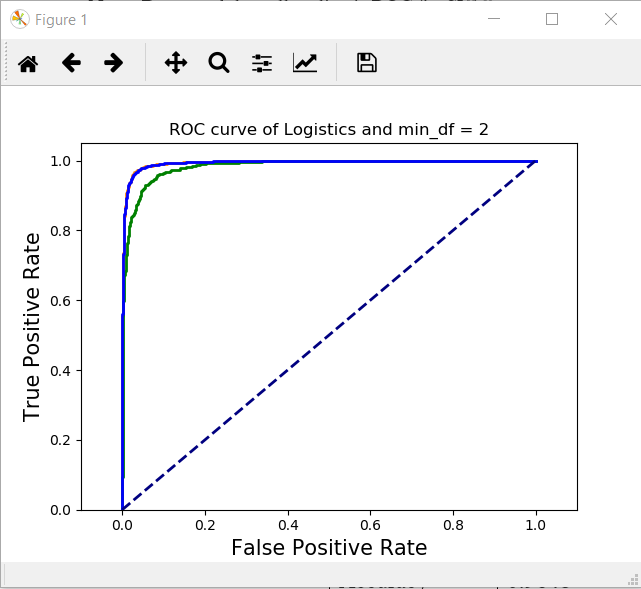
|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1360 | 200 |
| Actual Recreation | 37 | 1553 |

**Table 23 confusion matrix for Naïve Bayes Classifier (frequency = 5)**

## Question (h) Logistic Regression Classifier

In this section, we use logistics regression classifier to repeat previous steps and then plot the ROC.

In figure 13, 14 15, blue line is ROC for logistics classifier, the green line is ROC for Naïve Bayes and the blue line is ROC for SVM.



**figure 13 ROC for Logistic Classifier, LSI (frequency = 2)**

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9650 |
| Recall | 0.9780 |
| Precision | 0.9540 |

**Table 24 statics for Logistic Classifier (frequency = 2)**

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1485 | 75 |
| Actual Recreation | 35 | 1555 |

**Table 25 confusion matrix for Logistic Classifier (frequency = 2)**

## C:\Users\FANGYA~1\AppData\Local\Temp\1517362804(1).png

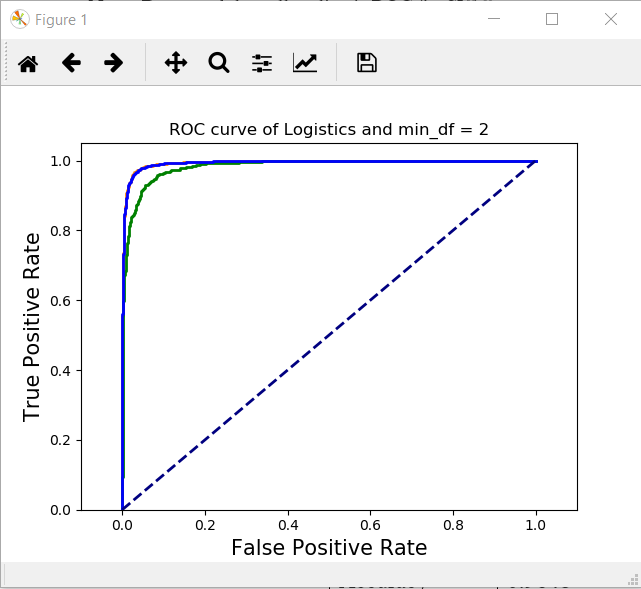
**figure 14 ROC for Logistic Classifier, LSI (frequency = 5)**

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9648 |
| Recall | 0.9792 |
| Precision | 0.9523 |

**Table 26 statics Logistic Classifier (frequency = 5)**

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1482 | 78 |
| Actual Recreation | 33 | 1557 |

**Table 27 confusion matrix Logistic Classifier (frequency = 5)**



**figure 15 ROC for NMF logistic regression (frequency = 2)**

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9098 |
| Recall | 0.9377 |
| Precision | 0.8896 |

**Table 28 statics for NMF logistic regression (frequency = 2)**

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1375 | 185 |
| Actual Recreation | 99 | 1491 |

**Table 29 confusion matrix for NMF logistic regression (frequency = 2)**

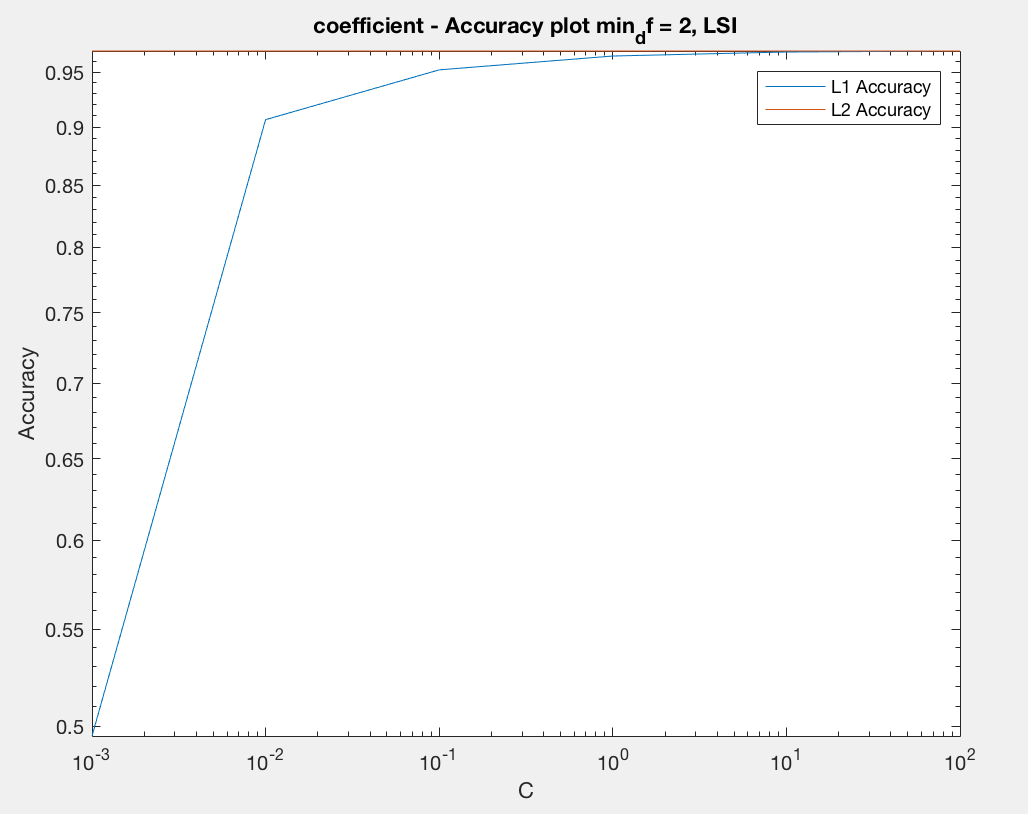
## Question (i) Logistic Regression Classifier with Regularization

In this section, We implement both 𝑙-1 and 𝑙-2 norm regularizations and use the logistic regression classifier for previous task. And optimum c for different min\_df and factorization method are shown below.

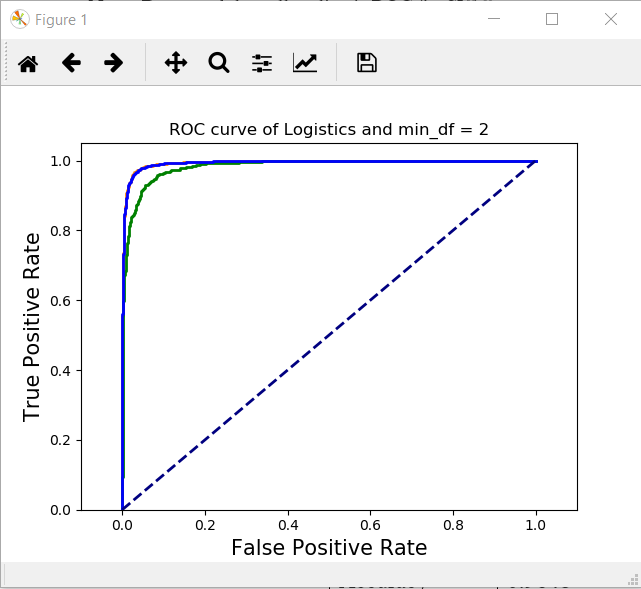
**For min\_df = 2, LSI**

|  |  |  |
| --- | --- | --- |
| C | L1 penalty accuracy | L2 penalty accuracy |
| 0.001 | 0.495238095238 | 0.969841269841 |
| 0.01 | 0. 906666666667 | 0.969841269841 |
| 0.1 | 0. 952063492063 | 0.969841269841 |
| 1 | 0. 965079365079 | 0.969841269841 |
| 10 | 0. 968888888889 | 0. 969841269841 |
| 100 | 0. 969841269841 | 0. 969841269841 |

**Table 30 statics for different coefficients**

****

**figure 16 min\_df = 2 penalty accuracy plot, LSI**



**figure 17 ROC for Logistic Regression Classifier with Regularization, LSI(frequency = 2)**

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9695 |
| Recall | 0.9811 |
| Precision | 0.9594 |

**Table 31 statics for logistic regression (frequency = 2)**

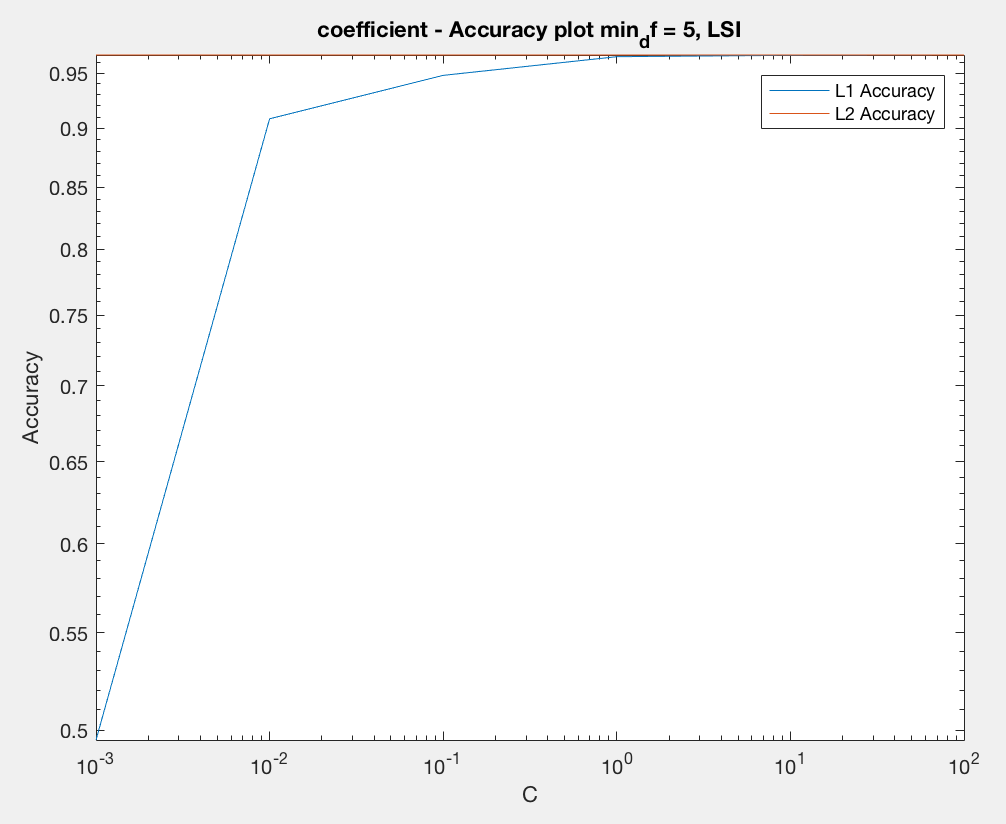
|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1494 | 66 |
| Actual Recreation | 30 | 1560 |

**Table 32 confusion matrix for NMF logistic regression (frequency = 2)**

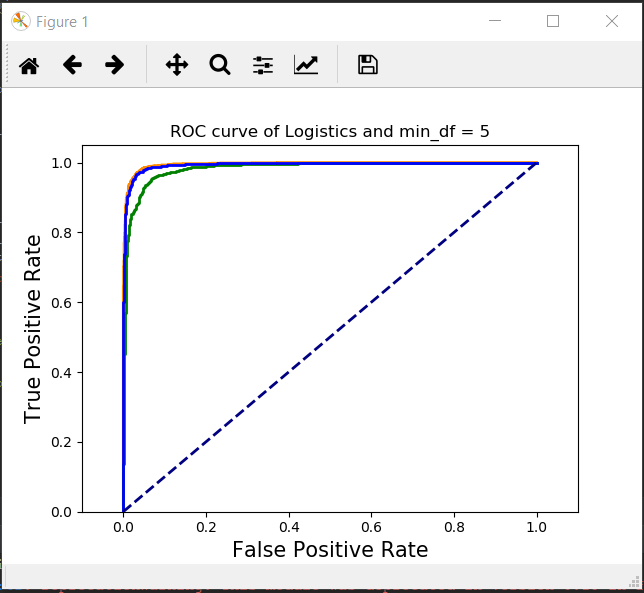
**For min\_df = 5, LSI**

|  |  |  |
| --- | --- | --- |
| C | L1 penalty accuracy | L2 penalty accuracy |
| 0.001 | 0. 495238095238 | 0. 966984126984 |
| 0.01 | 0. 908571428571 | 0. 966984126984 |
| 0.1 | 0. 947936507937 | 0. 966984126984 |
| 1 | 0. 965396825397 | 0. 966984126984 |
| 10 | 0. 966666666667 | 0. 966984126984 |
| 100 | 0. 966984126984 | 0. 966984126984 |

**Table 33 statics for different coefficients**

****

**figure 18 min\_df = 5 penalty accuracy plot, LSI**



**figure 19 ROC for Logistic Regression Classifier with Regularization, LSI(frequency = 5)**

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9679 |
| Recall | 0.9792 |
| Precision | 0.9582 |

**Table 34 statics for logistic regression with Regularization (frequency = 5)**

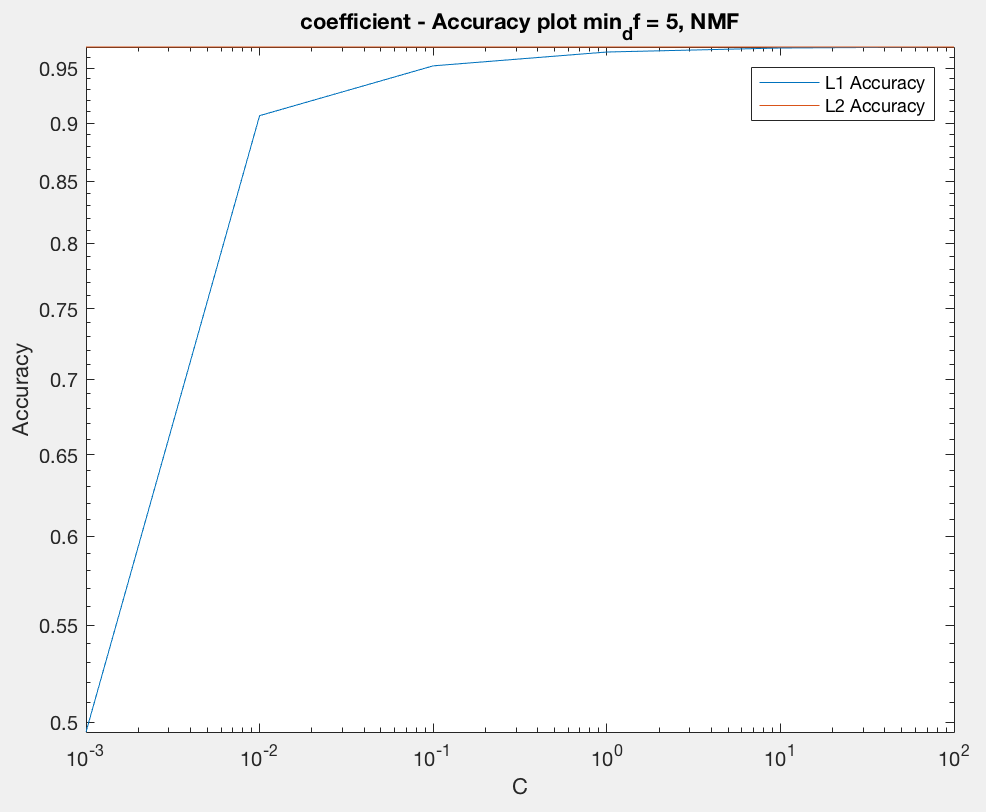
|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1492 | 68 |
| Actual Recreation | 33 | 1557 |

**Table 35 confusion matrix for logistic regression with Regularization (frequency = 5)**

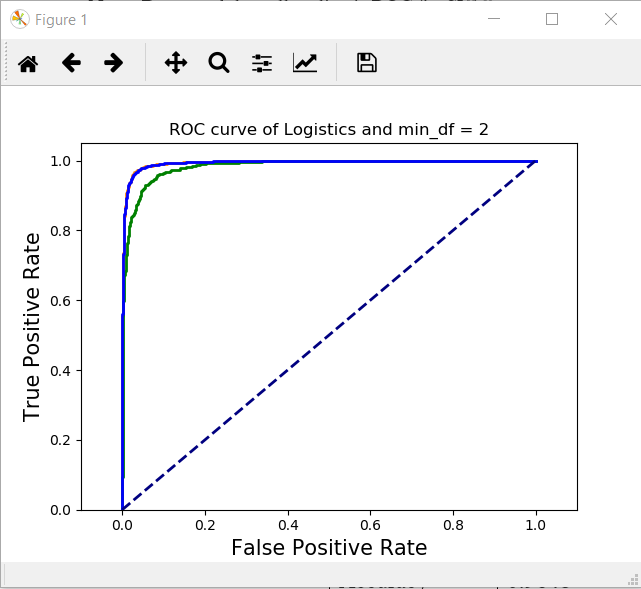
For min\_df = 2, NMF

|  |  |  |
| --- | --- | --- |
| C | L1 penalty accuracy | L2 penalty accuracy |
| 0.001 | 0. 495238095238 | 0. 969841269841 |
| 0.01 | 0. 906666666667 | 0. 969841269841 |
| 0.1 | 0. 952063492063 | 0. 969841269841 |
| 1 | 0. 965079365079 | 0. 969841269841 |
| 10 | 0. 968888888889 | 0. 969841269841 |
| 100 | 0. 969841269841 | 0. 969841269841 |

**Table 36 statics for different coefficients**

****

**figure 20 min\_df = 5 penalty accuracy plot, NMF**



**figure 21 ROC for Logistic Regression Classifier with Regularization, NMF(frequency = 2)**

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.9098 |
| Recall | 0.9377 |
| Precision | 0.8896 |

**Table 37 statics for logistic regression with Regularization (frequency = 2)**

|  |  |  |
| --- | --- | --- |
|  | Predicted Computer Tech | Predicted Recreation |
| Actual Computer Tech | 1375 | 185 |
| Actual Recreation | 99 | 1491 |

**Table 38 confusion matrix for logistic regression with Regularization (frequency = 2)**

From plot above we can find out that as the value of c increases, the accuracy gets higher, value of coefficients increase, number of nonzero coefficients increased.

Compared to penalty = l1, accuracy becomes higher for l2 penalty. Also all the coefficients of hyperplane are nonzero for l2 penalty.

We can find out that the result can be very different as coefficient varies. We must try all coefficients to get the optimum, because we cannot guarantee that one value can always provide the best result.

**Multiclass Classification**

## Question (j) Naïve Bayes classification and multiclass SVM classification

In this part, we build classifiers on the documents belonging to the classes mentioned in part b, which are ‘comp.sys.ibm.pc.hardware’, ‘comp.sys.mac.hardware’, ‘soc.religion.christian’ and ‘misc.forsale’.

1. Confusion matrix and statics for Multiclass NaiveBayes NMF & min\_df = 2:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Predict 1 | Predict 2 | Predict 3 | Predict 4 |
| Actual 1 | 337 | 20 | 31 | 4 |
| Actual 2 | 101 | 221 | 55 | 8 |
| Actual 3 | 55 | 18 | 307 | 10 |
| Actual 4 | 5 | 0 | 0 | 393 |

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.8038 |
| Recall | 0.8038 |
| Precision | 0.8149 |

1. Confusion matrix and statics for one vs one svm, LSI

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Predict 1 | Predict 2 | Predict 3 | Predict 4 |
| Actual 1 | 328 | 43 | 21 | 0 |
| Actual 2 | 39 | 322 | 23 | 1 |
| Actual 3 | 19 | 16 | 352 | 3 |
| Actual 4 | 4 | 2 | 0 | 392 |

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.8907 |
| Recall | 0.8907 |
| Precision | 0.8907 |

1. Confusion matrix and statics for one vs rest svm, LSI

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Predict 1 | Predict 2 | Predict 3 | Predict 4 |
| Actual 1 | 328 | 43 | 21 | 0 |
| Actual 2 | 39 | 322 | 23 | 1 |
| Actual 3 | 19 | 16 | 352 | 3 |
| Actual 4 | 4 | 2 | 0 | 392 |

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.8907 |
| Recall | 0.8440 |
| Precision | 0.8907 |

1. Confusion matrix and statics for one vs one svm, NMF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Predict 1 | Predict 2 | Predict 3 | Predict 4 |
| Actual 1 | 332 | 34 | 26 | 0 |
| Actual 2 | 80 | 277 | 27 | 1 |
| Actual 3 | 40 | 15 | 334 | 1 |
| Actual 4 | 11 | 4 | 5 | 392 |

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.8441 |
| Recall | 0.8441 |
| Precision | 0.8514 |

1. Confusion matrix and statics for one vs rest svm, NMF

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Predict 1 | Predict 2 | Predict 3 | Predict 4 |
| Actual 1 | 332 | 34 | 26 | 0 |
| Actual 2 | 80 | 277 | 27 | 1 |
| Actual 3 | 40 | 15 | 334 | 1 |
| Actual 4 | 11 | 4 | 5 | 378 |

|  |  |
| --- | --- |
| Statistic | Result |
| Accuracy | 0.8441 |
| Recall | 0.8441 |
| Precision | 0.8514 |