Assignment 2

Hrithik Maheshwari

March 9, 2020

1 Text Classification

1.1 Basic Naive Bayes (Part a)

In this part the training data was split on the spaces only and no punctuation and other things were removed. As required by the question it was trained from the first principle with Laplace Smoothing and used Lograthims to avoid underflows
The accuracy with the **Training Set** is : 86.2645625%
The accuracy with the **Test Set** is : 81.89415041782729%

1.2 Random/Majority Predection (Part b)

The accuracy with the **Random Prediction** was : 48.18941504178273%
The accuracy with the **Majority Prediction** was : 49.303621169916434%
Naive Bayes Algorithm almost give 30-35% more accuracy than Random and Majority Predection and hence its better to learn and predict then randomly guessing or predicting the one which comes more often

1.3 Confusion Matrix (Part c)

Class 4 Prediction has the highest value for having prediction of Actual Class 4.
It can be seen that Class 4 is predicted the correct most of the time. Also it can be seen that 81.11% (146/180) of the time Class 0 predicted is actually of Class 0 and 82.68% (148/179) Class 4 is predicted is actually of Class 4.

	Class 0(Predicted)	Class 4(Predicted)
Class 0(Actual)	146.0	31.0
Class 4(Actual)	34.0	148.0

1.4 Removing the noise in the data (Part d)

The stop words have been removed along with stemming. The sentence is converted to lower case and the tokens are split on the punctuation. Also the twitter handle names are removed

after matching the tokens with regular Expression.
The accuracy over **test set** is 83.28690807799443%
The confusion Matrix is as follows :

	Class 0(Predicted)	Class 4(Predicted)
Class 0(Actual)	148.0	29.0
Class 4(Actual)	31.0	151.0

The accuracy is increased from 81.89% to 83.28%. This hence prove that noise removal is better in this case for the results !

1.5 Feature Engineering (Part e)

In this I added two extra feature along with the bigram feature:

- **Bigram:** In this feature the dictionary was made by calculating all the bi features and threshold only those values whose count in all the documents of the single class is greater then 12. After than the probablity calculation was done from the Naive Bayes methord with the following formula given below
- **Trigram:** In this feature the dictionary was made by calculating all the tri features and threshold only those values whose count in all the documents of the single class is greater then 9. After than the probablity calculation was done from the Naive Bayes methord.
- **Quadgram:** In this feature the dictionary was made by calculating all the bi features and threshold only those values whose count in all the documents of the single class is greater then 7. After than the probablity calculation was done from the Naive Bayes methord.

P((single word,bi,tri,quad)/class) = P(single word/class)*P(bi/class)*P(tri/class)*P(quad/class)
P(single/Class) = (word count in class = "Class")/Total words in that class + len(Dictionary)
P(bi/Class) = (word count of bi features in class)/Total bi feature in class + len(Bi Dict)
P(tri/Class) = (word count of tri features in class)/Total tri feature in class + len(tri Dict)
P(quad/Class) = (count of quad features in class)/Allquad feature in class + len(quad Dict)
The accuracy over test set after this: 0.841225626740947
The confusion Matrix was as follows:

	Class 0(Predicted)	Class 4(Predicted)
Class 0(Actual)	148.0	29.0
Class 4(Actual)	28.0	154.0

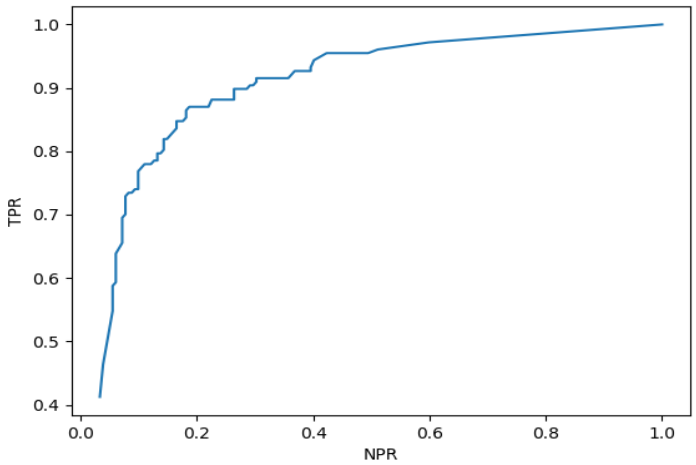
Observation: The class 4 prediction improved after implementing these features

1.6 Tf-IDF (Part f)

In this part implemented the Gaussian NB module and TfidfVectorizer for vectorizing the test and train data. But as the input features were very much and the array size tured out

to be 1600000 * 600000 and hence "Memory Error"
After that I divided the training set into 1600 batches each having 1000 entries and partially fitting the data using the Partial Fit Library. It took very much time as it was a very large feature vector array
The Accuracy from the test data comes out to be : 81.33 %

1.7 ROC Metric (Part g)



It can be seen that if the Negative Positive Rate is less than the True Posititve rate is also decreased and if it is increased than the TPR also gets increase. Our Treashold should be selected to maximize TPR*(1-NPR). It can be seen that the maxima will come near 0.5 which is ideal value for threshold

2 Fashion MNIST Article Classification

Binary Classifier

a) Basic SVM

After Calculating all the P,Q,G,H,a,B and putting them in the solvers Equation. We will get the alphas for the solution.
After getting alpha .... w can be easily calculated with the formula Sigma over i of alpha(i)*yi*xi and it will be a column vector of size (Number of Features,1)
b can be calculated with the formula given in the notes easily
Value of b -0.49049909
The given classifier trains for the class 5 and 6 and divides them.
The Test Set Accuracy comes out to be : 99.6% (996/1000)
The validation Set Accuracy comes out to be : 99.8% (499/500)
The Training Time was: 83.81172704696655 sec
The Testing time was: 0.9777848720550537 sec

b) Kernel SVM

After calculating all the P,Q,G,H,A,B and putting them in the solver like above we get the value of all the alphas. But this time we cannot calculate w but we can calculate b. For any prediction we have to calculate the whole expression.

$$\Sigma \alpha_i y_i < x_i, x > +b$$

And similarly the b can becalculatedd using its formula given in the Notes.
The value of b turns out to be : -0.12245082
The accuracy over Validation set turns out to be : 99.6% (498/1000)
The accuracy over Test Set turns out to be : 100.0% (1000/1000)
Observation : The accuracy is increased in Test Set while decreased in Validation Set (But Ideally it should have increased in both the cases)
The Training Time was : 63.944714069366455 sec
The Testing time was : 1.1221630573272705 sec

c) Sckit Learning

The accuracy with Linear Model on test set is : 99.6 % (996/1000)
The accuracy with Gaussian Model on test set is : 100 % (1000/1000)
The accuracy with Linear Model on validation set is : 99.8 % (499/500)
The accuracy with Gaussian Model on validation set is : 99.6 % (498/1000)
The training time with linear model is : 8.864399194717407 sec
The testing time with linear model is : 1.9983370304107666 sec
The training time with Gaussian model is : 14.55728530883789 sec
The testing time with Gaussian model is : 3.303057909011841 sec
The value of b in case of Linear Model : -0.49029701
The value of b in case of Gaussian Model : -0.12264118
The value of b in both the cases came out to be same as it should be the case.
The computation time in this is very less as compared to the normal fitting method.

Multi-Class Classification:

a) SVM Kernel Multi classifier

The Training time is : 2774.7111637592316
Accuracy on Test set is : 85.06% (4253.0/5000)
Accuracy on validation set is : 85.04% (2126.0/2500)
Confusion Matrix is as follows

Predicted →	T-shirt	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	Ankle boot
T-shirt)	405	0	7	7	0	0	71	0	10	0
Trouser	0	484	6	2	0	0	7	0	1	0
Pullover	0	0	414	4	26	0	43	0	13	0
Dress	18	10	2	412	6	0	42	0	10	0
Coat	0	1	56	14	365	0	52	0	12	0
Sandal	1	0	0	0	0	436	0	7	44	12
Shirt	56	1	57	6	20	0	345	0	15	0
Sneaker	0	0	0	0	0	50	0	412	4	34
Bag	1	0	1	0	0	1	3	0	494	0
Ankle boot	0	0	0	0	0	5	0	6	3	486

b) SVM Sckit Multi Classifier

The Training time is : 616.1668138504028 sec
Accuracy on Test set is : 88.08% (4404.0/5000)
Accuracy on validation set is : 87.92 % (2198.0/2500)
Confusion Matrix is as follows :

Predicted →	T-shirt	Trouser	Pullover	Dress	Coat	Sandal	Shirt	Sneaker	Bag	Ankle boot
T-shirt	433	0	5	11	3	0	38	0	10	0
Trouser	1	482	4	9	0	0	4	0	0	0
Pullover	5	0	411	7	37	0	32	0	8	0
Dress	12	0	3	457	9	0	14	0	5	0
Coat	3	1	41	13	399	0	38	0	5	0
Sandal	0	0	0	0	0	473	0	16	5	6
Shirt	80	0	55	9	34	0	315	0	7	0
Sneaker	0	0	0	0	0	14	0	471	1	14
Bag	1	0	1	1	2	2	2	2	489	0
Ankle boot	0	0	0	0	0	11	0	14	1	474

2.0.1 c) Confusion Matrix:

The matrix for both part are above
It can be seen that when it is actually Tshirt than 71 times it has predicted it to be a Shirt. Pullover in actual is many a times misinterpreted as Coat and similarly cat is 56 times misinterpreted as a Pullover
Sneaker are 50 times misinterpreted as Sandals
It can be seen that Similiar Dresses (Like Tshirt and Shirt, Sandal and Sneaker....etc) are mainly misclassified most times coz they are generally both Upperware, or Lowerware or Footwear.

2.0.2 d) K-Fold Validation:

In this part the training data was divided into k parts and k-1 parts were used for train and the last was used for validation and it was done with k hypothesis each having diffrent validation set.
The accuracy with gamma 0.00001 on validation set : 55.6 %
The accuracy with gamma 0.001 on validation set : 82.4222%
The acccuracy with gamma 1 on validation set : 13.77 %
The accuracy with gamma 5 on validation set : 9.4 %
The accuracy with gamma 10 on validation set : 2.4 %
The accuracy with gamma 0.00001 on test set : 67.58%
The accuracy with gamma 0.001 on test set : 82.34%
The acccuracy with gamma 1 on test set : 22.42%
The accuracy with gamma 5 on test set : 10.16%
The accuracy with gamma 10 on test set : 3.4%
The value of C as 0.001 gives the best accuracy in Training as well as Test Set.