Naïve Bayes Classifiers and K means Clustering

Aditya Tiwari(2016csb1029)

Naïve Bayes Classifiers

The assignment was to classify a given mail as spam or not spam on the basis of words that constitute it using Naive Bayes Classifier Model. The model considered m-estimation in calculating posterior probabilities with m=|Vocab| and p= $\frac{1}{|Vocab|}$.

<u>Results:</u>

Accuracy on Training is **0.913**

Accuracy on Test is **0.899**

Top 5 words indicating Spam with their posterior probability:

1. enron: 0.0371195456851
2. a: 0.0245840841951
3. the: 0.0229999464732
4. corp: 0.021127783711
5. to: 0.0194442391169

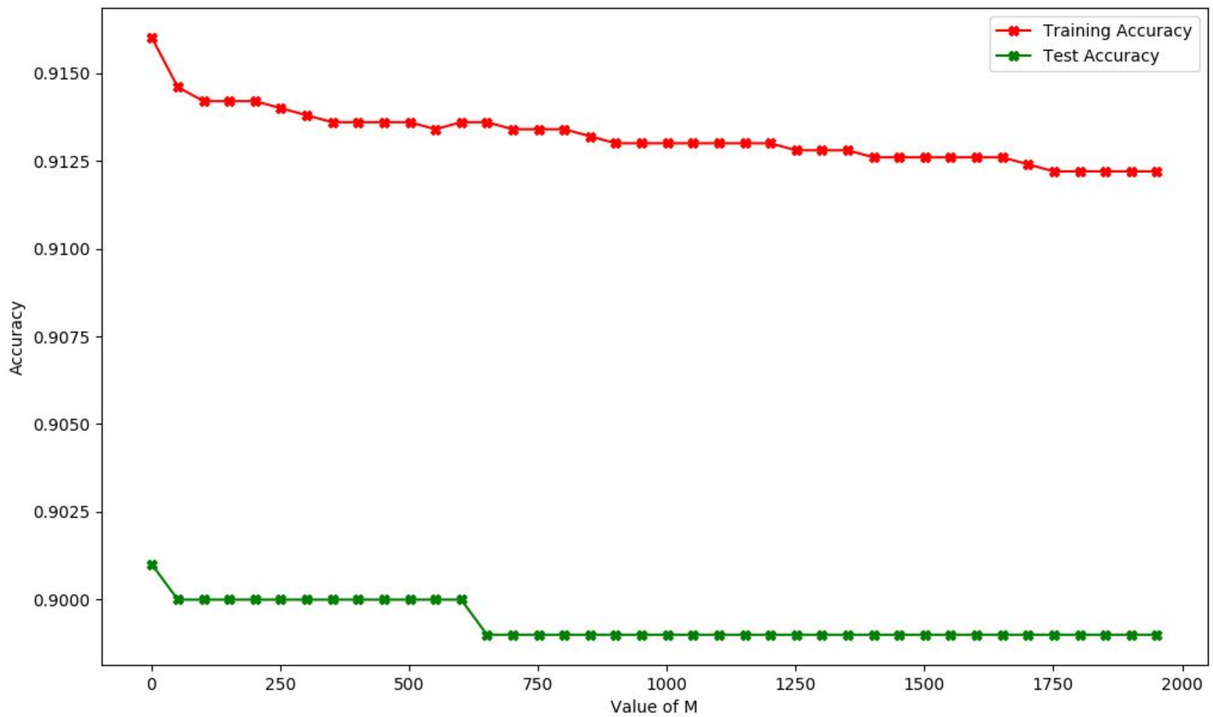
Top 5 words indicating Non-Spam with their posterior probability:

1. aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa: 0.0559609110405
2. enron: 0.0406659063735
3. the: 0.0368279212769
4. to: 0.0265208307094
5. a: 0.0188500364906

The Trouble with Predicting the class was that multiplying the Posterior probabilities of words lead to the the product being very close to zero which was throwing a math domain error. The Solution was to take log of the product i.e Sum of log of posterior

The Result of Varying m and observing Accuracy:

With P constant as p= $\frac{1}{|Vocab|}$

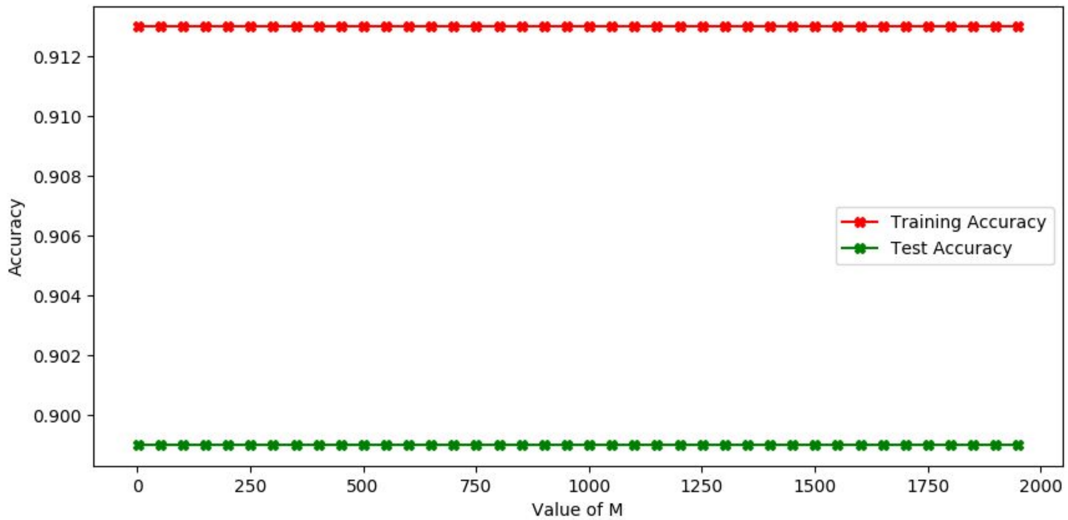


Observe Accuracy decreases as we increase m

We make the assumption with large m that for a given word, if its frequency is high, then only it affects spam decision else it won't.

And for words which aren't present in training, large M would mean that this word's effect is very low as M is in denominator.

With MxP constant i.e P= p= $\frac{1}{|M|}$



The m-estimation considers the words that were in test but not in training with some probability so as their posterior product doesn't goes to zero. If indeed M==0 means the given word since wasn't in the training the product would lead to zero, as we increase M the Accuracy decreases. I believe this is because Since we don't know anything about the current word that is not in training, its posterior of being equal to M

With P varying as MxP=1, the accuracy remains constant

<u>Beating The the Naive Basis Classifier:</u>

The model doesn't considers the frequency of the words when calculating the probability, it just takes in presence,So just increasing the frequency of same word in a mail wouldn't change its spamicity probability. However if One increased the number of words with High non-spam Posterior, NBC would consider it as non-spam even though it may be spam

This shortcoming of the model is because it relies on the Bayesian probability to calculate if it is a spam or not.

K-Means Clustering

The assignment was to label given 20x20 images of handwritten digits to actual digits.

Dataset:MNIST handwritten digits' dataset

***Library used sklearn**

For 5 Clusters, Accuracy is 0.4338

Label in Cluster	Label Assigned
1. 0	9
2. 1	0
3. 2	2
4. 3	6
5. 4	5

Here Label in cluster is the random(Varies after each run) label assigned by k means and Label Assigned is the Actual Label that was most frequent in random Labels.

E.g if we considered all the instances that were classified as 0 by the k means, and take the most frequent actual label of those instances, 9 would come out.

For 10 Clusters, Accuracy is 0.5562

Label in Cluster	Label Assigned
1. 0	2
2. 1	0
3. 2	0
4. 3	3
5. 4	9
6. 5	6
7. 6	7
8. 7	1
9. 8	5
10. 9	9

We Can see that there was only one cluster that was assigned actual 0 in k=5 where has here there are two clusters assigned 0 in k =10

For 15 Clusters, Accuracy is 0.673

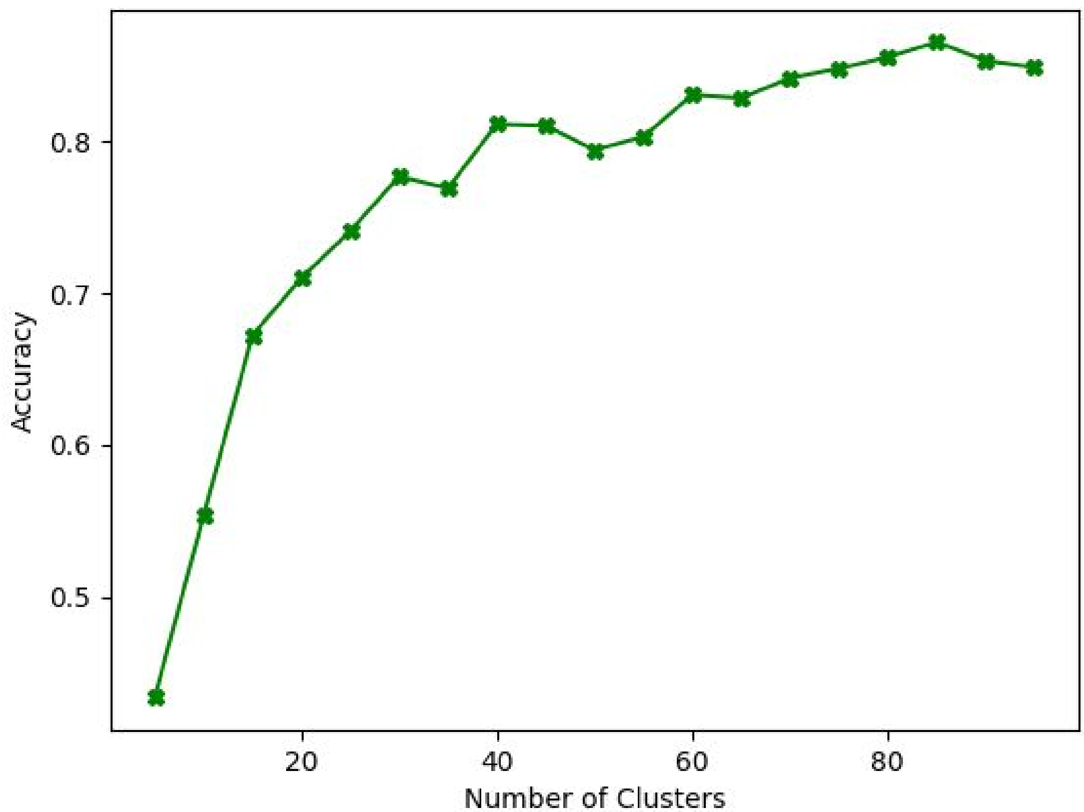
Label in Cluster	Label Assigned
1. 0	0
2. 1	4
3. 2	2
4. 3	7
5. 4	9

6. 5 2
7. 6 0
8. 7 3
9. 8 6
10. 9 1
11. 10 9
12. 11 5
13. 12 3
14. 13 6
15. 14 5

We Can see that there was only one cluster that was assigned actual 2 in k=10 where has here there are two clusters assigned 2

Similarly for 6 which comes two times. Since the label is from 0, 6 is for 7(THE DIGIT) and we can see that there are two 6 in k=15 and one 6 in k=10

So as K increases, the instances become more and more sparse and actual labels come out to be more frequent .



See the Above Graph.

It clearly shows increase in accuracy with increase in clusters which does risk overfitting

I think labels given in label.txt were 1-10 because then 6 would correspond to 7 and 0 to 1.

We can see in k=5 there were one 0(ACTUAL 1) and one 6(ACTUAL 7) in assigned label

And in k=10, there are two 0(ACTUAL 1) and one 6 (ACTUAL 7), so decreasing the clusters could have merged one of the cluster of 0(ACTUAL 1) with other 0(ACTUAL 1) itself or 6(ACTUAL 7).