**Problem Set 4 – Email Spam Classifier**

**Students – Manas Thakre, Charudatta Deshpande**

To create an email spam classifier, we employed three machine learning algorithms –

1. Naïve Bayes
2. Decision Tree
3. Supervised KNN

We will analyze steps involved, results from different methods and compare them.

Data is randomized and split into training and testing set. The test data size is set to 30%.

**Method 1 – Naïve Bayes**

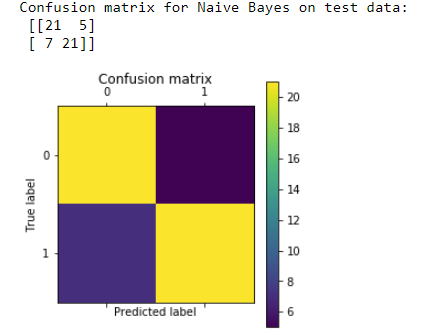
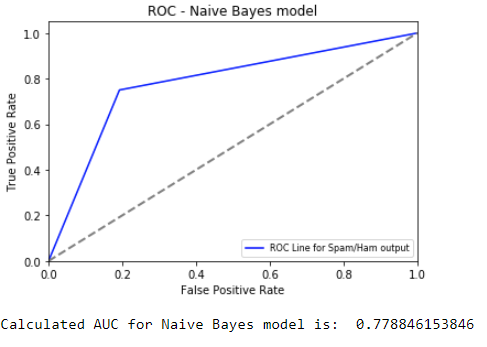
Step 1 – Create a model using a combination of text vectorizer function ‘CountVectorizer’ and multinomial Naïve Bayes function ‘MultinomialNB’.



Step 2 – Train the model using email text as predictor and category (Spam or Ham) as target. Use K-fold sampling with K = 6.

Step 3 – Predict the model using test dataset. Binarize the output, plot ROC curve, calculate AUC (Area under the curve) and print and plot confusion matrix.





**Method 2 – Decision Tree**

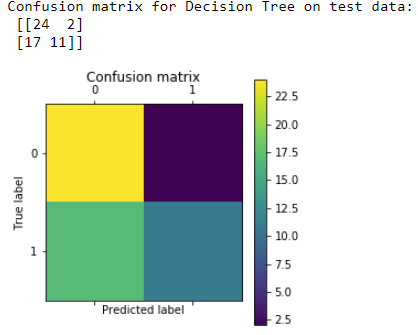
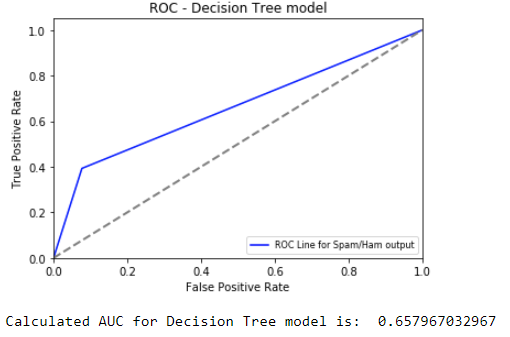
Step 1 – Create a model using a combination of text vectorizer function ‘CountVectorizer’ and Decision Tree function ‘tree.DecisionTreeClassifier’.



Step 2 – Train the model using email text as predictor and category (Spam or Ham) as target. Use K-fold sampling with K = 6.

Step 3 – Predict the model using test dataset. Binarize the output, plot ROC curve, calculate AUC (Area under the curve) and print and plot confusion matrix.





**Method 3 – Supervised KNN**

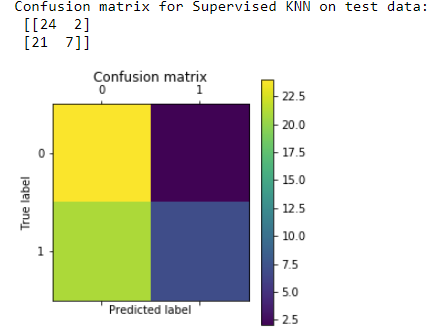
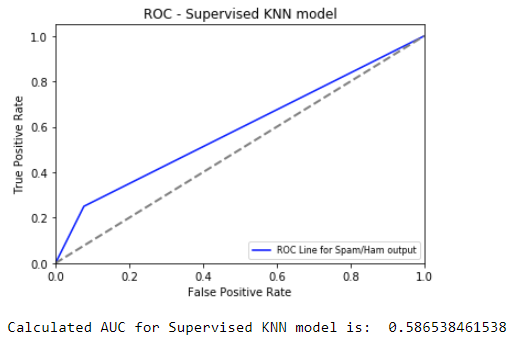
Step 1 – Create a model using a combination of text vectorizer function ‘CountVectorizer’ and K Nearest Neighbors function ‘KNeighborsClassifier’.



Step 2 – Train the model using email text as predictor and category (Spam or Ham) as target. Use K-fold sampling with K = 6.

Step 3 – Predict the model using test dataset. Binarize the output, plot ROC curve, calculate AUC (Area under the curve) and print and plot confusion matrix.





**Algorithm Comparison**

First step of the comparison is to analyze the ROC curves and confusion matrices plotted above. Then we can summarize the methods as below –

|  |  |  |
| --- | --- | --- |
| Naïve Bayes | Decision Tree | Supervised KNN |
| Accuracy = 77.78%. This is the most accurate method of the three. This is also equal to the calculated AUC. | Accuracy = 65.8%. This is the second most accurate method of the three. This is also equal to the calculated AUC. | Accuracy = 58.65%. This is the least accurate method of the three. This is also equal to the calculated AUC. |
| The ROC curve lies above the 50% or chance line. This indicates that this model performs better than chance. | The ROC curve lies above the 50% or chance line. This indicates that this model performs better than chance. | The ROC curve lies above the 50% or chance line. This indicates that this model performs better than chance. |
| This algorithm is based on Bayes’ probability theorem. Bayes theorem calculates posterior probability P(c|x) from P(c), P(x) and P(x|c) where c is class or target, and x is the predictor. It assumes independence between predictors. | Decision Tree works by breaking the data down into smaller and smaller subsets and developing a tree with decision nodes and leaf nodes. Most commonly used decision tree algorithm is ID3 by J. R. Quinlan which employs a top-down, greedy search. ID3 uses Entropy and Information Gain to build a decision tree. | KNN works by identifying K number of nearest data points to the target. It can employ various methods to calculate the distance and provide appropriate weight to the distances. It then analyzes features of the identified neighbors and assigns a class to the target based on the analysis. |
| Pros – Very simple, and works well with large amount of data, faster performance, and provides very accurate results if assumption of independence indeed holds. | Pros – Simple algorithm, robust with noise in the data, handles both continuous and discrete data, works well with small number of data points and features. | Pros – Simple algorithm, robust with noise, makes no assumptions about data, can be used for both classification and regression. |
| Cons – Assumption of independence seldom holds true in real life, less accurate with less data points | Cons – can grow exponentially with increase in number of features, needs additional techniques like pruning to keep the tree from exploding in size, may need additional improvements like bagging, boosting, stacking etc. | Cons – very sensitive to value of K, stores all data so it is computationally expensive, slower performance, can assign more weight to irrelevant features. |
| In this experiment, there is only one predictor, so assumption of independence holds true. As expected, Naïve Bayes provided the most accurate results | In this experiment, there is only one predictor. Number of data points are adequate. Decision tree performs reasonably well in these conditions. It is still weaker than Naïve Bayes. | In this experiment, there is only one predictor. Number of data points are adequate. But due to the nature of the experiment, text-based predictor can yield unsatisfactory results for KNN because of the subjectivity involved with text data. It would have worked better with more robust categorical data, like presence of multimedia, link etc. |

**Conclusion**

Based on above comparison, we determine that Naïve Bayes provides most accurate results of the three methods used. When assumption of independence holds, Naïve Bayes outperforms many other classification algorithms. We also note that the accuracy of the model can be improved by including additional predictors available in the data. This was not done for this experiment but can be taken as an optional improvement exercise.