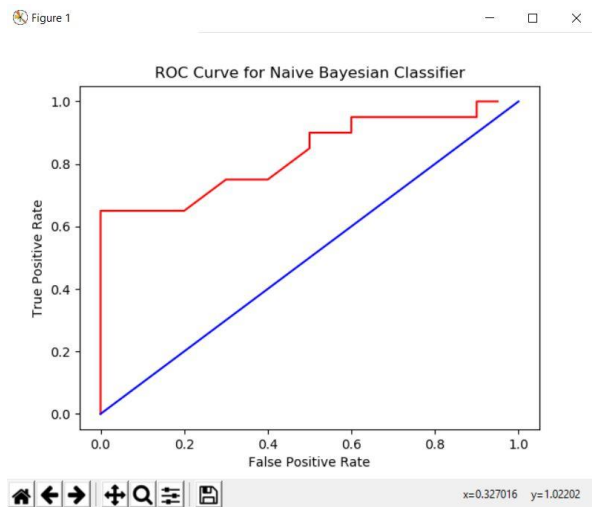


DATA SCIENCE ASSIGNMENT 1

Sahil Pundora 216092223

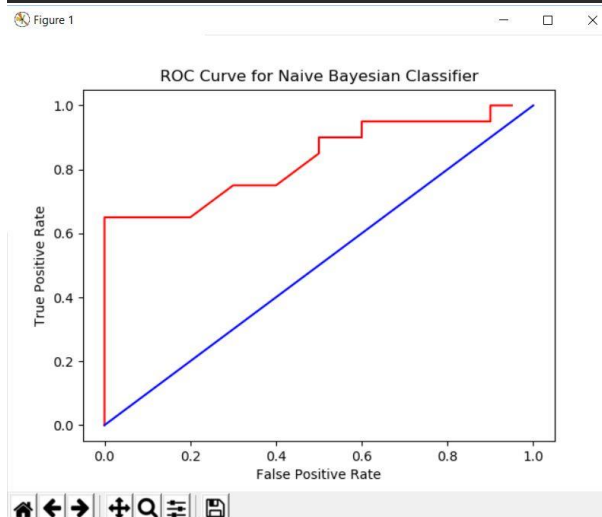
A - Output

```
"C:/Users/sahil/Documents/Business Analytics/Coursework - Fall Term/Python Codes/venv/Scripts/python.exe" "C:/Users/sahil/Documents/Business Analytics/Coursework - Fall Term/Python Codes/nb3.py"
WELCOME TO NAIVE BAYES CLASSIFIER
Inputting data from CSV.....
Following is the data-
['Obs', 'TestRes/Var1', 'Var2', 'Var3', 'Var4', 'Var5', 'Var6'] [[1.0, 0.0, 1.0, 0.0, 0.0, 1.0, 1.0], [2.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [3.0, 0.0, 2.0, 1.0, 1.0, 1.0, 1.0], [4.0, 0.0, 0.0, 1.0, 1.0, 1.0, 1.0], [5.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [6.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [7.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [8.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [9.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0], [10.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]]
Press Enter to continue...
Classifying the data based on Naive Bayes theorem
TestRes/Var1 is the Target Variable
Var 2-to-6 are Independent Variables (Features of the dataset)
Press Enter to continue...
CLASSIFICATION RESULTS
[Instance No., Target, Score]
[[1.0, 0.0, 0.015015], [2.0, 0.0, 0.2], [3.0, 0.0, 0.063903125000000002], [4.0, 0.0, 0.275], [5.0, 0.0, 0.2], [6.0, 0.0, 0.006971250000000001], [7.0, 0.0, 0.027885000000000004], [8.0, 0.0, 0.098312500000000002], [9.0, 0.0, 0.015015], [10.0, 0.0, 0.015015]]
Press Enter to continue...
Plotting ROC Curve.....
Press Enter to continue...
```



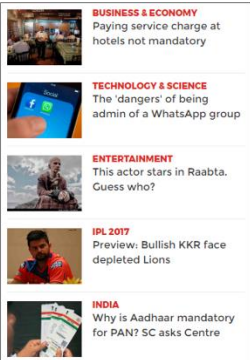
C - Output

```
"C:/Users/sahil/Documents/Business Analytics/Coursework - Fall Term/Python Codes/venv/Scripts/python.exe" "C:/Users/sahil/Documents/Business Analytics/Coursework - Fall Term/Python Codes/nb3.py"
WELCOME TO NAIVE BAYES CLASSIFIER
Inputting data from CSV.....
Creating Table and storing in the DataBase.....
Following is the data-
['Obs', 'TestRes/Var1', 'Var2', 'Var3', 'Var4', 'Var5', 'Var6'] [[1.0, 1, 0, 0, 1, 1], [2.0, 0, 0, 0, 0, 0], [3.0, 2, 1, 1, 1, 1], [4.0, 0, 1, 0, 0, 1], [5.0, 2, 1, 0, 0, 0], [6.0, 0, 0, 0, 0, 0], [7.0, 0, 0, 0, 0, 0], [8.0, 0, 0, 0, 0, 0], [9.0, 0, 0, 0, 0, 0], [10.0, 0, 0, 0, 0, 0]]
Press Enter to continue...
Classifying the data based on Naive Bayes theorem
TestRes/Var1 is the Target Variable
Var 2-to-6 are Independent Variables (Features of the dataset)
Press Enter to continue...
CLASSIFICATION RESULTS
[Instance No., Target, Score]
[[1.0, 0.015015], [2.0, 0.2], [3.0, 0.063903125000000002], [4.0, 0.275], [5.0, 0.2], [6.0, 0.006971250000000001], [7.0, 0.027885000000000004], [8.0, 0.098312500000000002], [9.0, 0.015015], [10.0, 0.015015]]
Press Enter to continue...
Plotting ROC Curve.....
Press Enter to continue...
!End of Program!
Press Enter to Exit (paradox eh! :p)
```




D – Advantages of Naïve Bayes Classifier and When to use it over Deep Learning Networks

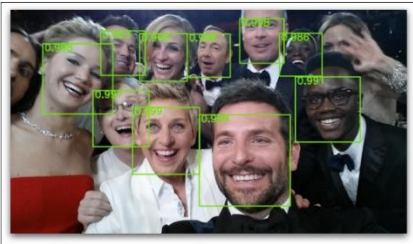
Categorizing News




Email Spam Detection




Face Recognition




Sentiment Analysis




Medical Diagnosis



Digit Recognition



Weather Prediction



“Categorizing news, email spam detection, face recognition, sentiment analysis, medical diagnosis, digit recognition and weather prediction are just few of the popular use cases of Naive Bayes algorithm “

Naïve Bayes vs Deep Learning

In order to use Naïve Bayes we assume that your features are independent of one another. When this assumption holds, it is a very strong classifier that requires very little data to work. Where as in deep learning such as Recurrent Neural Networks, the networks read your data sequentially, while keeping a "memory" of what they have read previously (these are really useful when dealing with text because of the correlation words have between them).

The two models (NB and RNN) differ greatly in the way they attempt to perform this classification:

- NB belongs to a category of models called **generative**. This means that during training (the procedure where the algorithm learns to classify), NB tries to find out how the data was *generated* in the first place. It essentially tries to figure out the underlying distribution that produced the examples you input to the model.
- On the other hand, RNN is a **discriminative** model. It tries to figure out what the differences are between your positive and negative examples, in order to perform the classification.

The main Assumption for NB - All the features used are mutually independent, **which is mostly not true in real world**. It will work best when you manually handcraft the features in such a way that there is negligible dependency between the features. Each feature contributes independently to the probability of the class. Whereas RNN are more

Another advantage of Naïve Bayes is that it has been shown to perform surprisingly well with very small amounts of training data that most other classifiers (including Deep learning ANNs).

One more reason - When computational resources are limited (low) we use NB Classifier. RNNs are starting to find applications over the past decade because of their need for high computational resources. RNNs most of the time are trained on dedicated GPUs (which compute a lot faster than CPUs).

Should we have split our data into training and validation sets?

Since the data is very small ~40 records, there is not much sense to split the data into test and training sets. Also, since the output is based on the probabilities, it does not make difference if the data is randomized as well.