DTSC 620 - STATISTICS FOR DATA SCIENCE
PROJECT ASSIGNMENT 1

BAHADURI PRACHITI JAGDISH
NYIT ID: 1317686
Manhattan Campus (M01)

Project Report

Project objective: To use the spam.csv dataset and perform classification with the help of 2 classifiers i.e Random Forest classifier and Decision tree classifier.
Project requirements: Well written report and compatible code
About the data: There are totally 57 features which constitute the number of times a certain word or character occurred in an email message with a total of 4601 instances. This data has to be classified into 2 classes viz 'ham' or 'spam'.

	make	address	all	3d	our	over	remove	internet	order	mail	...	semicol	paren	bracket	bang	dollar	pound	cap_avg	cap_long	cap_total	Class
0	0.00	0.00	0.29	0.0	0.00	0.00	0.00	0.00	0.00	0.00	...	0.000	0.178	0.0	0.044	0.000	0.00	1.666	10	180	ham
1	0.46	0.00	0.00	0.0	0.00	0.00	0.00	0.00	0.00	0.00	...	0.000	0.125	0.0	0.000	0.000	0.00	1.510	10	74	ham
2	0.00	0.00	0.00	0.0	0.00	0.00	0.00	0.00	0.00	0.00	...	0.000	0.000	0.0	0.000	0.000	0.00	1.718	11	55	ham
3	0.33	0.44	0.37	0.0	0.14	0.11	0.00	0.07	0.97	1.16	...	0.006	0.159	0.0	0.069	0.221	0.11	3.426	72	819	spam
4	0.00	2.08	0.00	0.0	3.12	0.00	1.04	0.00	0.00	0.00	...	0.000	0.000	0.0	0.263	0.000	0.00	1.428	4	20	spam
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
4596	0.00	0.00	0.53	0.0	0.00	0.53	0.00	0.00	0.00	0.53	...	0.000	0.101	0.0	0.000	0.000	0.00	1.857	16	52	ham
4597	0.00	0.00	0.00	0.0	0.00	0.00	0.00	0.00	0.00	0.00	...	0.000	0.443	0.0	0.221	0.665	0.00	3.812	15	61	spam
4598	0.00	0.00	0.00	0.0	0.00	0.00	0.00	0.00	0.00	0.00	...	0.000	0.000	0.0	0.000	0.000	0.00	1.000	1	3	ham
4599	0.00	0.00	0.00	0.0	0.00	0.00	0.00	0.00	0.00	0.00	...	0.000	0.218	0.0	0.218	0.000	0.00	1.687	10	27	ham
4600	0.13	0.26	0.52	0.0	0.26	0.00	0.13	0.00	0.00	0.39	...	0.000	0.000	0.0	0.366	0.000	0.04	7.138	149	1235	spam

4601 rows × 58 columns

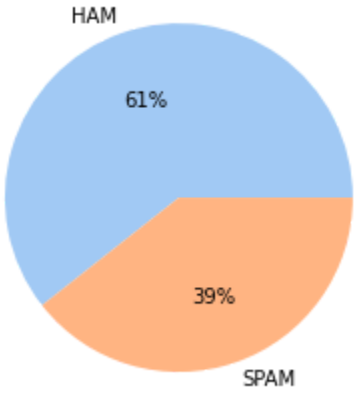
Libraries imported:
For this project, multiple libraries are imported. The **pandas, numpy** is included for the normal dataframe operations. The **warnings** library has been imported to avoid any kind of warnings that might occur while running a snippet of code. As for the data visualizations is concerned, **matplotlib and seaborn** libraries are used. The **sklearn** library has been used to split the dataset into train and test data, to import the classifiers, and to perform some metrics on the test data.

Exploratory Data Analysis:
Exploratory Data Analysis or in short EDA refers to the process of exploring the data to discover various patterns, to look out for any kind of anomalies, to check whether there are any null values, and to perform different statistical and visual interpretations between different features provided in the dataset.

This dataset has no null values which is found out with the help of **isna()** function. The **describe()** function is giving the count, mean, standard deviation, minimum value, maximum value, 25%, 50% and 75% of each particular feature. The **value_count()** function gives the total number of ham and spam word or character.

Data Visualization:
Data Visualization is basically graphically representation of information or data. It includes different kinds of plots for different kinds of data. There can be Scatter plot, Line plot, Histogram, Dendrogram, Bar plot, Heat map and so on. There are certain libraries that are used to create these plots.

This project makes use of the most common libraries such as **Matplotlib and Seaborn**. The pie chart is depicting the portion of ham and spam words or characters as shown below.



We can also see that in the above pie chart there is data imbalance. But in this project I am handling imbalance dataset problem.

Splitting of dataset:
The dataset will be split into train and test data respectively using the **train_test_split** function with a test size of 0.78 for 1000 training instances.

Training the dataset using the 2 classifiers:

Random Forest classifiers:

This is where the training of individual models happen parallely by a random subset of data. According to the problem statement, for this project three **base learners or estimators** will be considered i.e **100, 500 and 1000** and their accuracies will be compared with respect to the number of features to be considered i.e **max_features** which is either auto or sqrt**.**

n_estimators	auto	ham (auto)	spam (auto)	sqrt	ham (sqrt)	spam (sqrt)
100	93%	96%	90%	93%	96%	90%
500	94%	96%	90%	94%	96%	90%
1000	93%	96%	90%	94%	96%	90%

Above is the table which shows a comparison of the accuracies when the max_features is set to auto and sqrt for number of estimators being 100, 500 and 1000 respectively.

Decision tree classifier:

The decision tree classifier is a flowchart like structure which consists of multiple components. Each internal node represents a test on a particular feature, each branch of the node becomes the outcome of the test and each leaf node of the tree becomes the decision of the decision or class label assigned after computing all the features in the given dataset.

```
---------DECISION TREE CLASSIFIER-------------
CONFUSION MATRIX:
 [[1936  206]
 [ 215 1232]]
TOTAL ACCURACY SCORE:88%
Accuracy of Ham class is 90%
Accuracy of Spam class is 85%
```

Splitting the dataset:

The dataset will be split into train and test data respectively using the **train_test_split** function with a test size of 0.22 for 3601 training instances.

Training the dataset using the 2 classifiers:

Random Forest classifiers:

n_estimators	auto	ham (auto)	spam (auto)	sqrt	ham (sqrt)	spam (sqrt)
100	95%	97%	93%	96%	97%	94%
500	96%	97%	94%	95%	97%	94%
1000	96%	97%	94%	95%	97%	94%

Decision tree classifier:

```
---------DECISION TREE CLASSIFIER-------------
CONFUSION MATRIX:
 [[575  37]
 [ 40 361]]
TOTAL ACCURACY SCORE:92%
Accuracy of Ham class is 94%
Accuracy of Spam class is 90%
```

Conclusion:

From this project, in case of Random Forest classifier we can conclude that the accuracy was higher when the number of trees was considered as 1000 with a per class accuracy for ham and spam being 96% and 90% respectively when the max_features was set to sqrt and a total accuracy score of 94%. It is highly similar when you round to 2 decimals.

In case of Decision tree classifier, the total accuracy score is 88% and for each class accuracy with a score of 90% and 85% for ham and spam respectively.