Final Project [Naïve Bayes Classifier]

By – Ankit Chaudhary, Harshvardhan Sawal

Approach: We have implemented Naïve Bayes Classifier using Gaussian.

As Gaussian is probabilistic in nature so we have converted Categorical data into Binary data (Numeric Data).

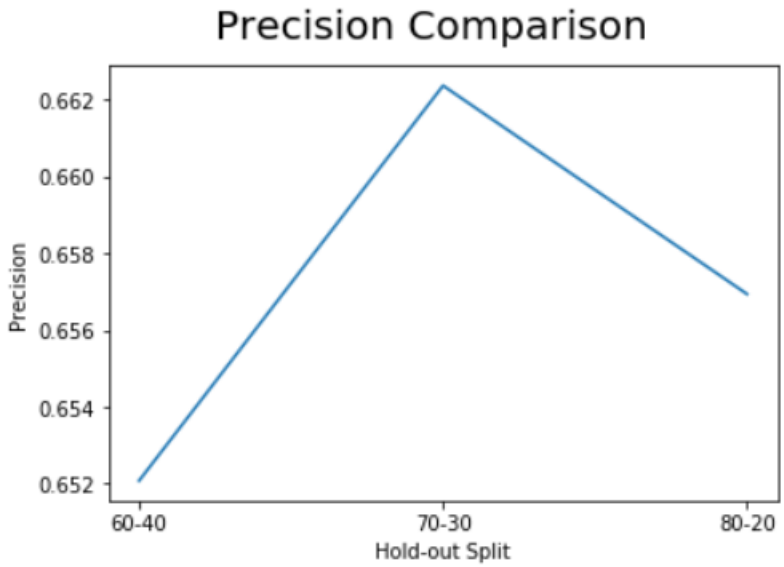
APIs Used:

pandas: used to read dataset
matplotlib.pyplot: used to create graphs
sklearn: **a.)** *model_selection*: used to split dataset into test and train data
b.) *metrics*: used to create confusion matrix and finding accuracy score
c.) *naïve_bayes*: used to implement GaussianNB

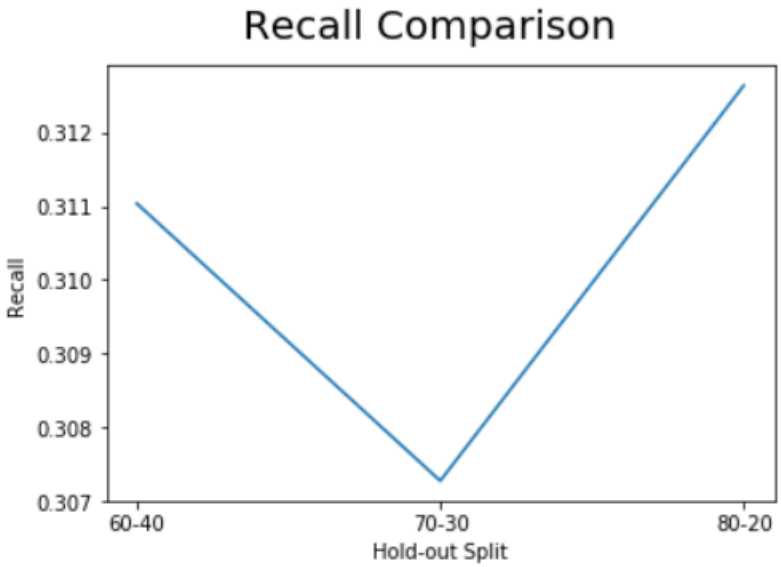
Outputs:

```
Splits 60-40 :

Accuracy :  0.7945948712698981
Recall :  0.31103536429484446
Precision :  0.6520768200089325


Splits 70-30 :

Accuracy :  0.7968334129529789
Recall :  0.3072753209700428
Precision :  0.6623616236162362


Splits 80-20 :

Accuracy :  0.7994677039615109
Recall :  0.3126356925749023
Precision :  0.656934306569343
```

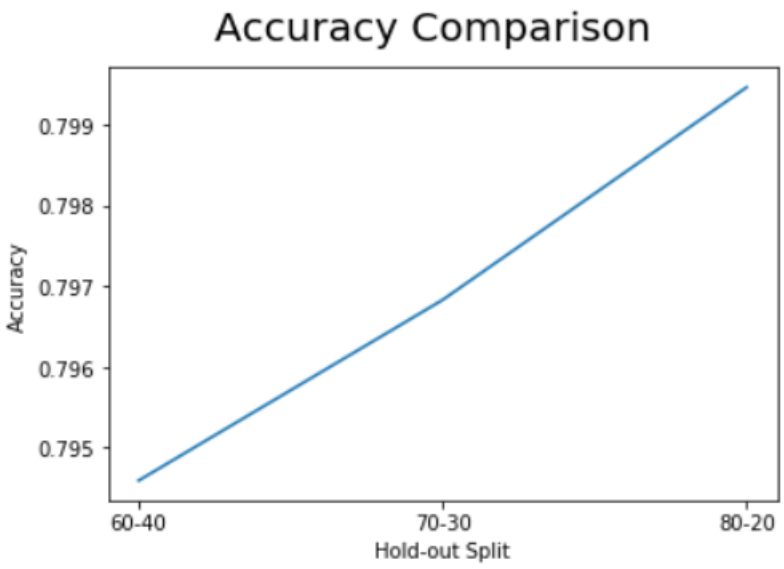
1. Y- axis: precision, x axis: hold out split



2. Y- axis: recall, x axis: hold out split



3. Y- axis: accuracy, x axis: hold out split



References:

[1.] : https://pandas.pydata.org/pandas-docs/stable/generated/pandas.read_csv.html
[2.] : https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html
[3.] : https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html#sklearn.naive_bayes.GaussianNB.predict
[4.] : https://scikit-learn.org/stable/auto_examples/model_selection/plot_confusion_matrix.html#sphx-glr-auto-examples-model-selection-plot-confusion-matrix-py
[5.] : https://matplotlib.org/api/_as_gen/matplotlib.pyplot.plot.html