Pattern Recognition and Machine Learning

Lab Assignment 8

Feature Selection

Jatin Lohar

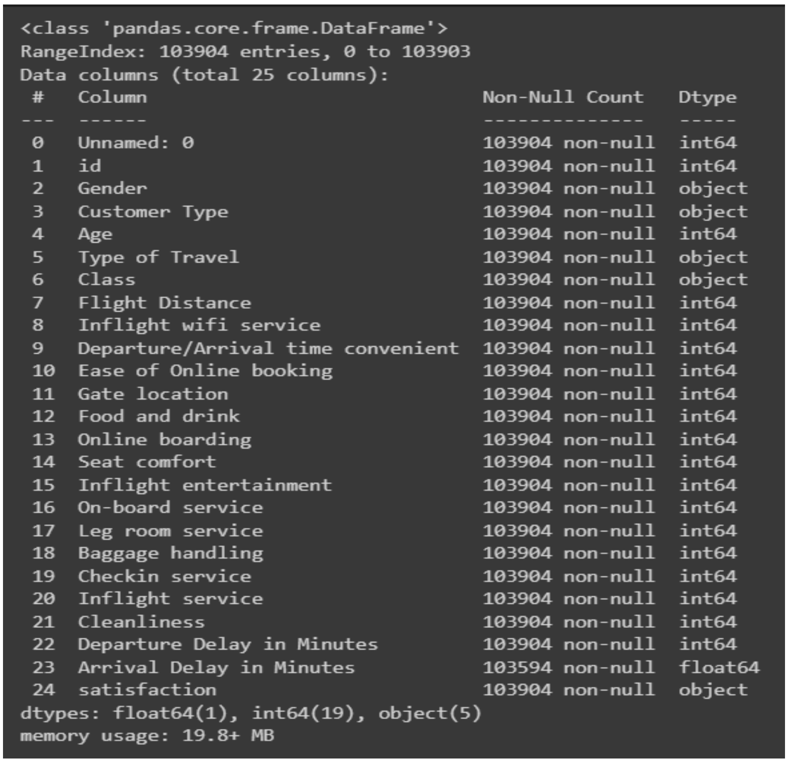
B21CS091

Question 1 (Sequential Feature Selection)

Firstly, imported the necessary libraries and also installed ***'mlxtend'*, *'SequentialFeatureSelector'*** so as to use in further questions.

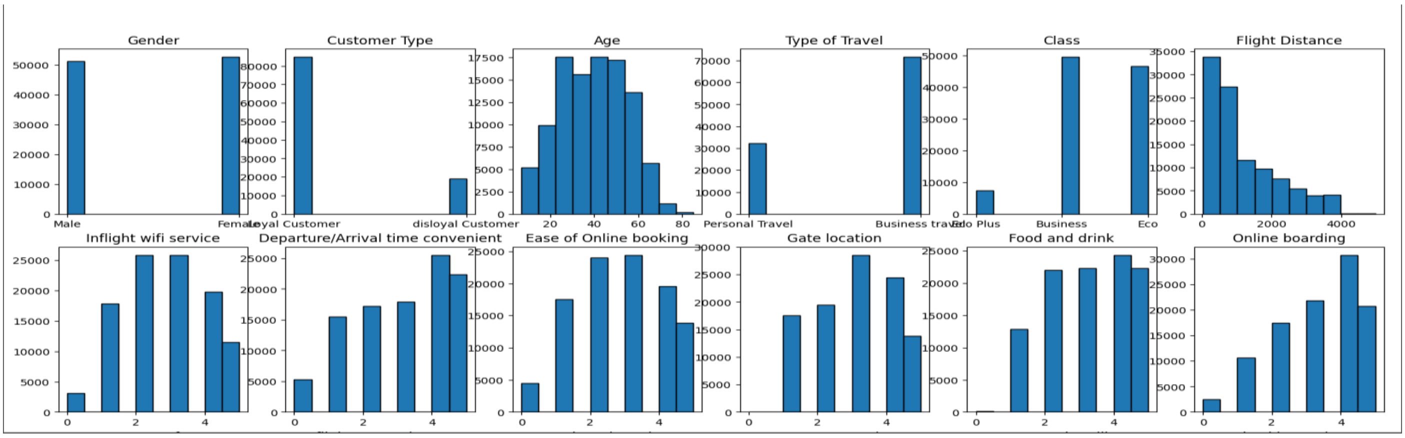
Part 1

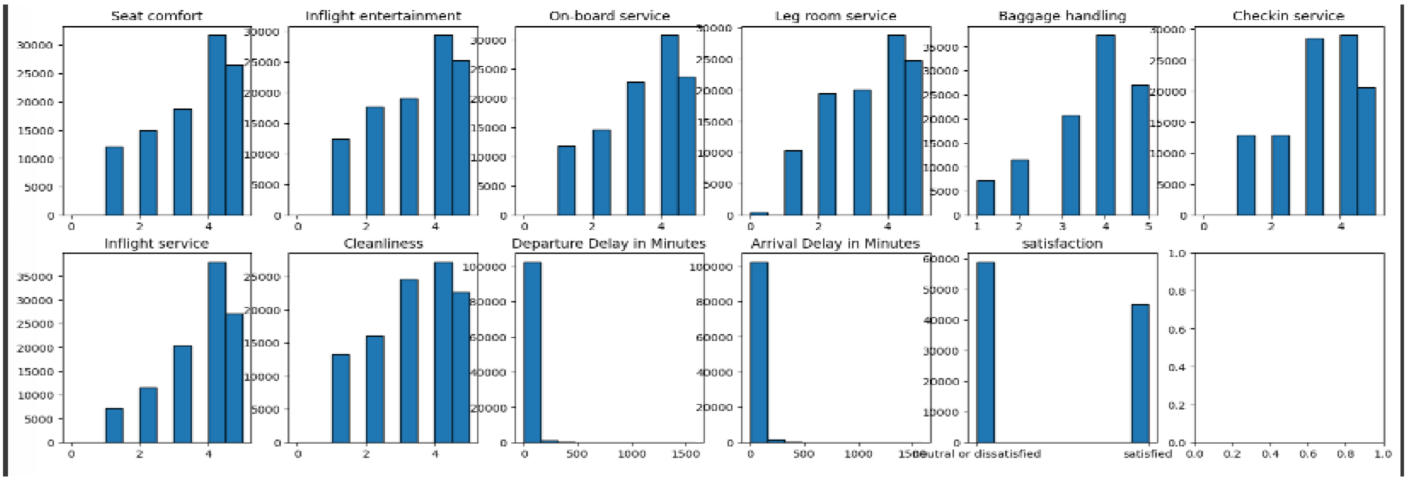
Imported the data using ***'pandas.read_csv'*** and got the necessary details of the data.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 103904 entries, 0 to 103903
Data columns (total 25 columns):
 #   Column                             Non-Null Count   Dtype
---  ------                             --------------   -----
 0   Unnamed: 0                         103904 non-null  int64
 1   id                                 103904 non-null  int64
 2   Gender                             103904 non-null  object
 3   Customer Type                      103904 non-null  object
 4   Age                                103904 non-null  int64
 5   Type of Travel                     103904 non-null  object
 6   Class                              103904 non-null  object
 7   Flight Distance                    103904 non-null  int64
 8   Inflight wifi service              103904 non-null  int64
 9   Departure/Arrival time convenient  103904 non-null  int64
 10  Ease of Online booking             103904 non-null  int64
 11  Gate location                      103904 non-null  int64
 12  Food and drink                     103904 non-null  int64
 13  Online boarding                    103904 non-null  int64
 14  Seat comfort                       103904 non-null  int64
 15  Inflight entertainment             103904 non-null  int64
 16  On-board service                   103904 non-null  int64
 17  Leg room service                   103904 non-null  int64
 18  Baggage handling                   103904 non-null  int64
 19  Checkin service                    103904 non-null  int64
 20  Inflight service                   103904 non-null  int64
 21  Cleanliness                        103904 non-null  int64
 22  Departure Delay in Minutes         103904 non-null  int64
 23  Arrival Delay in Minutes           103594 non-null  float64
 24  satisfaction                       103904 non-null  object
dtypes: float64(1), int64(19), object(5)
memory usage: 19.8+ MB
```

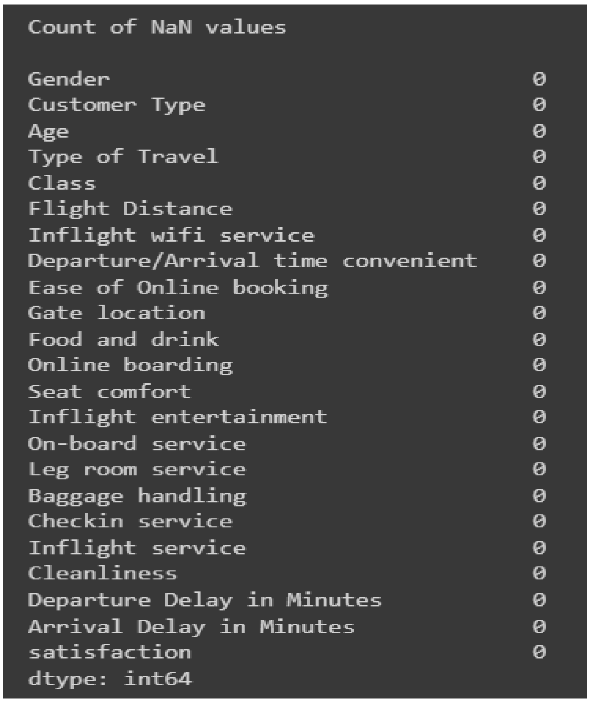
Analysed the Data by making histograms. And saw the following Distribution.





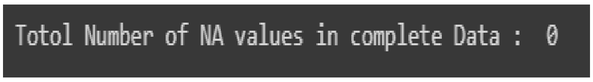
Found ***'Unnamed: 0', 'id'*** columns useless, so dropped them from the data.

Later checked for NAN values in the Dataset and found 310 NaN values in ***'Arrival Delay in Minutes'*** column. Then replaced it with the median of the columns.



```
Count of NaN values

Gender                               0
Customer Type                        0
Age                                  0
Type of Travel                       0
Class                                0
Flight Distance                      0
Inflight wifi service                0
Departure/Arrival time convenient    0
Ease of Online booking               0
Gate location                        0
Food and drink                       0
Online boarding                      0
Seat comfort                         0
Inflight entertainment               0
On-board service                     0
Leg room service                     0
Baggage handling                     0
Checkin service                      0
Inflight service                     0
Cleanliness                          0
Departure Delay in Minutes           0
Arrival Delay in Minutes             0
satisfaction                         0
dtype: int64
```

Before replacing



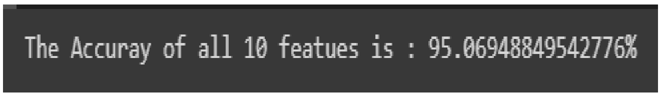
```
Totol Number of NA values in complete Data :  0
```

After replacing

Then converted the categorical columns into numerical values and added them into data. Then split the data into X and Y.

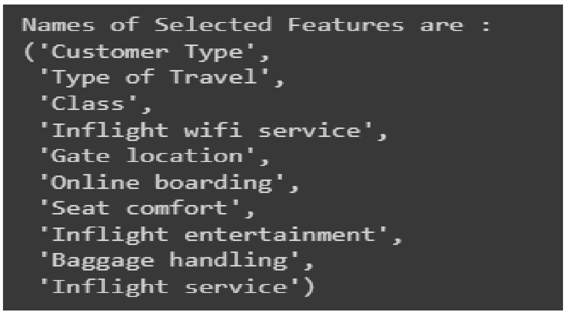
Part B

Imported ***'DecisionTreeClassifier'*** from sklearn.tree. Made a ***'SequentialFeatureSelector'*** with given properties. Fitted X and Y into the feature selector. Then printed the accuracy using k_score.



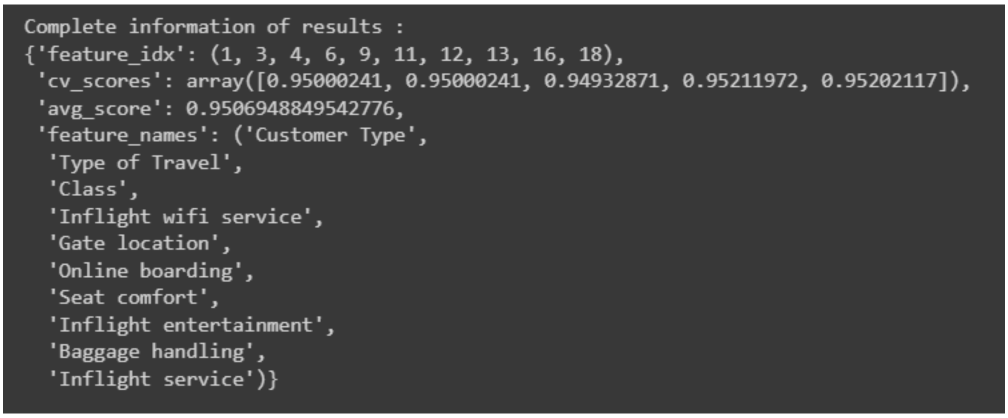
```
The Accuray of all 10 featues is : 95.06948849542776%
```

The Selected Features were…



```
Names of Selected Features are :
('Customer Type',
 'Type of Travel',
 'Class',
 'Inflight wifi service',
 'Gate location',
 'Online boarding',
 'Seat comfort',
 'Inflight entertainment',
 'Baggage handling',
 'Inflight service')
```

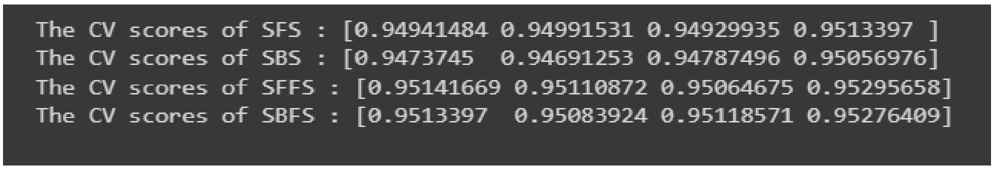
Complete details of the 10 features selected was :



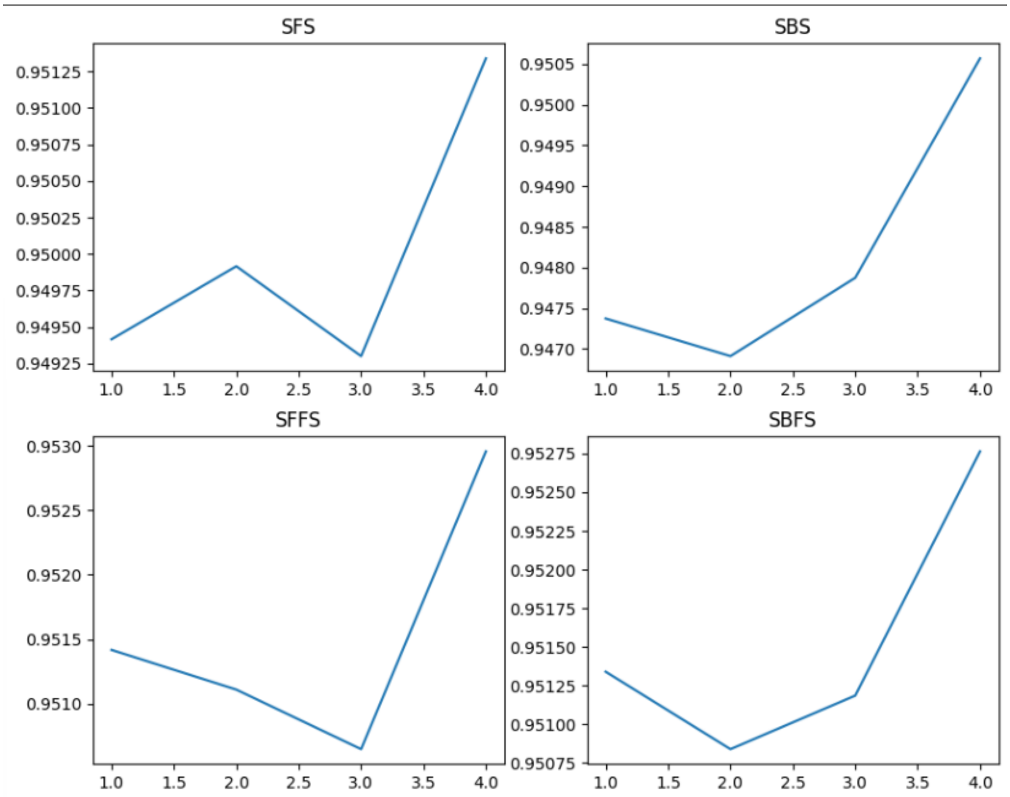
```
Complete information of results :
{'feature_idx': (1, 3, 4, 6, 9, 11, 12, 13, 16, 18),
 'cv_scores': array([0.95000241, 0.95000241, 0.94932871, 0.95211972, 0.95202117]),
 'avg_score': 0.9506948849542776,
 'feature_names': ('Customer Type',
  'Type of Travel',
  'Class',
  'Inflight wifi service',
  'Gate location',
  'Online boarding',
  'Seat comfort',
  'Inflight entertainment',
  'Baggage handling',
  'Inflight service')}
```

Part 3

Made feature Selector with the specifications given in the question. Fitted X and Y in the feature selectors. The Cross Validation Scores of the features selectors, given cv = 4 were :

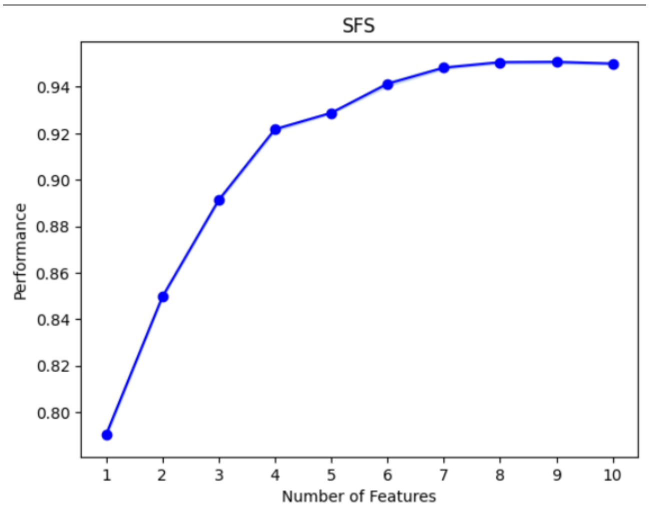


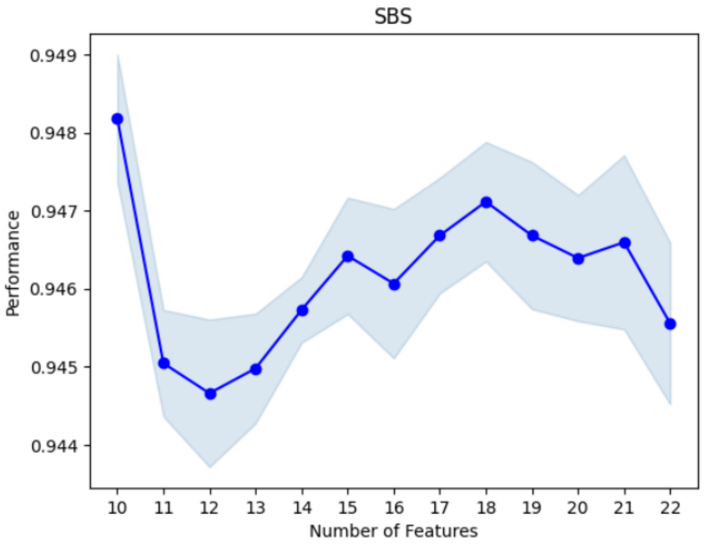
```
 The CV scores of SFS : [0.94941484 0.94991531 0.94929935 0.9513397 ]
 The CV scores of SBS : [0.9473745  0.94691253 0.94787496 0.95056976]
 The CV scores of SFFS : [0.95141669 0.95110872 0.95064675 0.95295658]
 The CV scores of SBFS : [0.9513397  0.95083924 0.95118571 0.95276409]
```

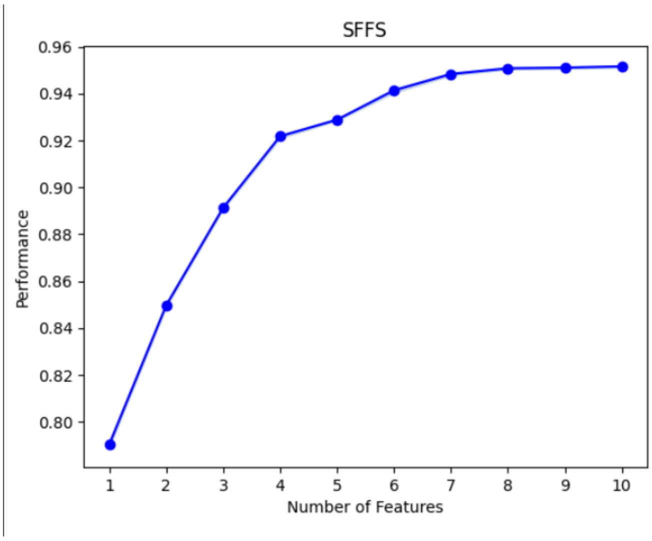


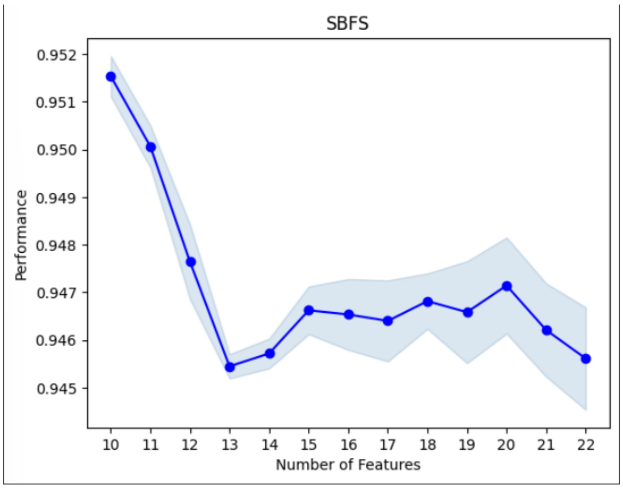
Part 4

Visualizing the output from the feature selection :









Part 5, 6

Imported ***'train_test_split', 'SVC', 'DecisionTreeClassifier'*** so as to use in this part. Made functions to calculate the confusion matrix, accuracy from confusion matrix (since we have just 2 classes, so there will bw just 2 X 2 confusion matrix.

Later made functions to find the entropy for given predictions and hence the ***'Information Gain'*** .

Function for finding the Euclidean Distance or in other words the root MSE of the model. Returning negative since higher value signifies poor predictions.

Function for finding City Block Distance, that is the absolute difference between the prediction and actual value. Returning negative since higher value signifies poor predictions.

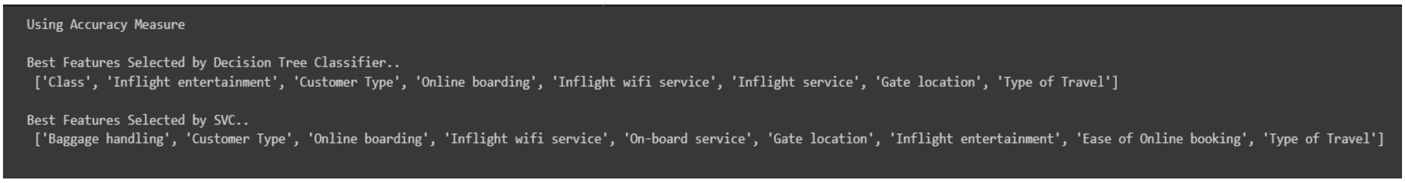
Function for finding Angular Separation between the predicted and the actual data. Returned the Angular distance.

Mainly, made the function for Bi-Directional Feature Set Generation. Made a set of best features, that will later be returned. Finding the performance of the every feature along with the best features and finding the feature with best accuracy. Later moving backwards, we will remove the feature in the best feature that performs worst by removing that feature and then checking the accuracy. Removing that feature from best features. Continuing the process until there are no features remaining. Later returning the best features in the form of a list.

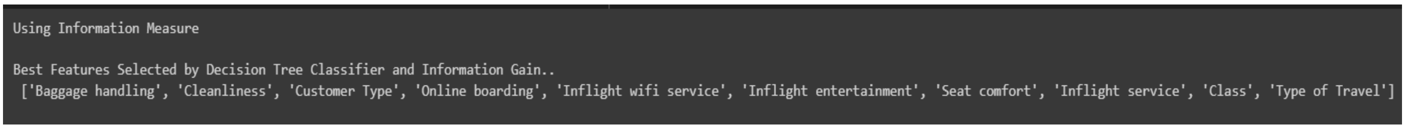
Later since the SVC was taking a lot of time to procces and train of 1 lakh data points. So, to reduce the time I considered 10,000 random points with equal number of data poimts of each class and used it for SVC model.

Part 7

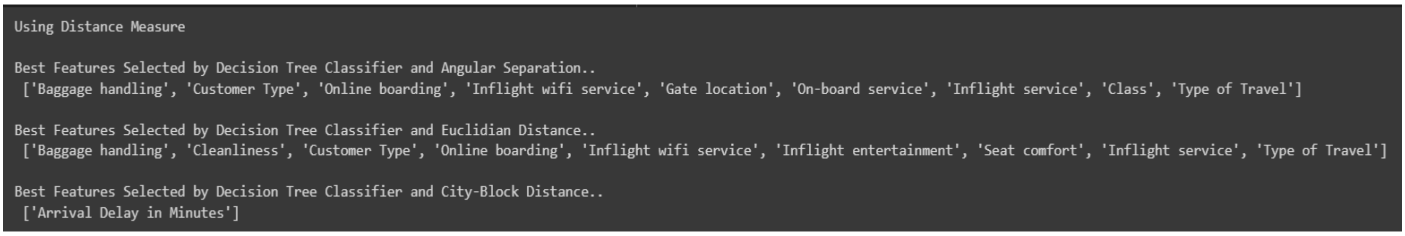
Applied the Feature selector with given measures and got the following features selected.



```
Using Accuracy Measure

Best Features Selected by Decision Tree Classifier..
 ['Class', 'Inflight entertainment', 'Customer Type', 'Online boarding', 'Inflight wifi service', 'Inflight service', 'Gate location', 'Type of Travel']

Best Features Selected by SVC..
 ['Baggage handling', 'Customer Type', 'Online boarding', 'Inflight wifi service', 'On-board service', 'Gate location', 'Inflight entertainment', 'Ease of Online booking', 'Type of Travel']
```

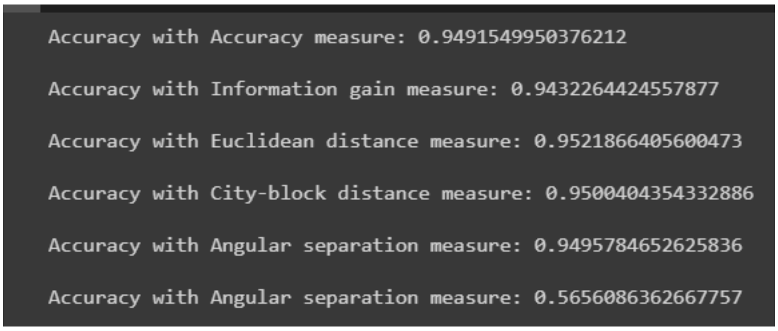


```
Using Information Measure

Best Features Selected by Decision Tree Classifier and Information Gain..
 ['Baggage handling', 'Cleanliness', 'Customer Type', 'Online boarding', 'Inflight wifi service', 'Inflight entertainment', 'Seat comfort', 'Inflight service', 'Class', 'Type of Travel']
```

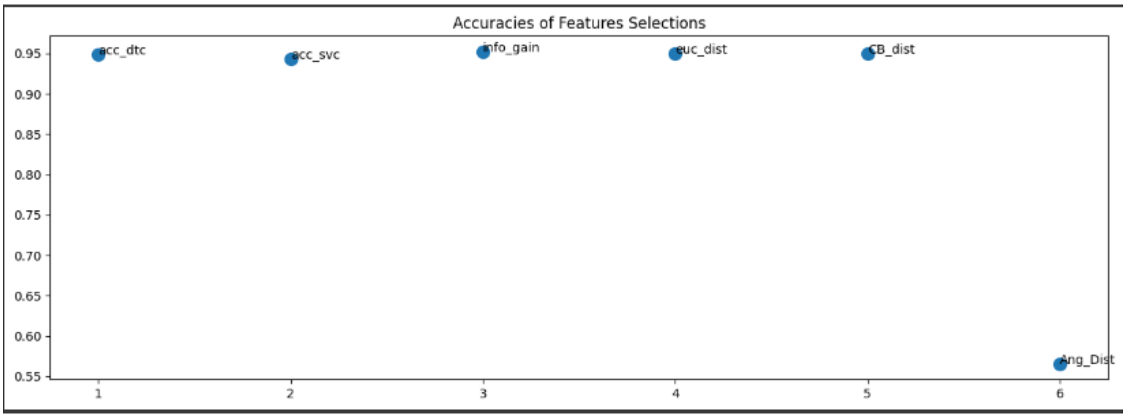


```
Using Distance Measure

Best Features Selected by Decision Tree Classifier and Angular Separation..
 ['Baggage handling', 'Customer Type', 'Online boarding', 'Inflight wifi service', 'Gate location', 'On-board service', 'Inflight service', 'Class', 'Type of Travel']

Best Features Selected by Decision Tree Classifier and Euclidian Distance..
 ['Baggage handling', 'Cleanliness', 'Customer Type', 'Online boarding', 'Inflight wifi service', 'Inflight entertainment', 'Seat comfort', 'Inflight service', 'Type of Travel']

Best Features Selected by Decision Tree Classifier and City-Block Distance..
 ['Arrival Delay in Minutes']
```

Also, checked the accuracy of the models using ***'cross_val_score'*** .



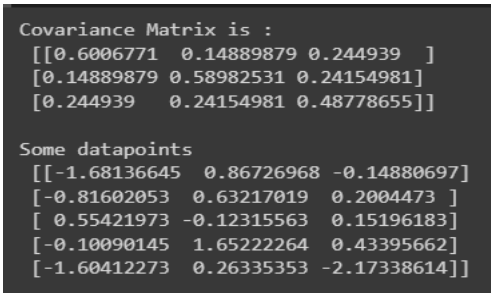
```
Accuracy with Accuracy measure: 0.9491549950376212

Accuracy with Information gain measure: 0.9432264424557877

Accuracy with Euclidean distance measure: 0.9521866405600473

Accuracy with City-block distance measure: 0.9500404354332886

Accuracy with Angular separation measure: 0.9495784652625836

Accuracy with Angular separation measure: 0.5656086362667757
```



Question 2

Part 1

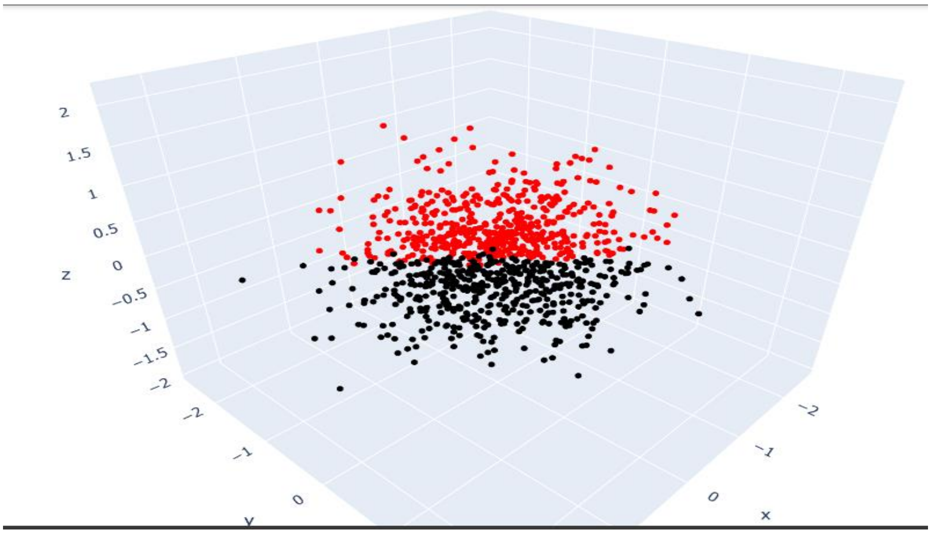
Made a 2D numpy array with given entries. And the Dataset using ***'np.random.multivariate_normal'*** with mean as zero matrix as given covariance matrix.



```
Covariance Matrix is :
 [[0.6006771  0.14889879 0.244939  ]
 [0.14889879 0.58982531 0.24154981]
 [0.244939   0.24154981 0.48778655]]

Some datapoints
 [[-1.68136645  0.86726968 -0.14880697]
 [-0.81602053  0.63217019  0.2004473 ]
 [ 0.55421973 -0.12315563  0.15196183]
 [-0.10090145  1.65222264  0.43395662]
 [-1.60412273  0.26335353 -2.17338614]]
```

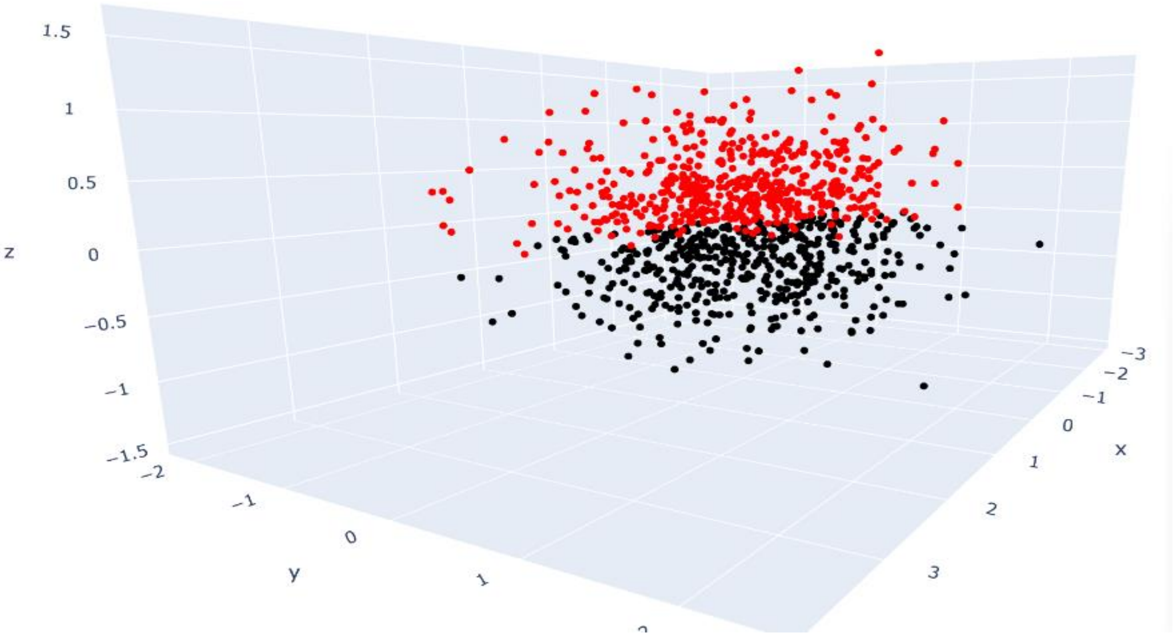
Made the 'v' vector with given values and the did a dot product of it with X and stored it in a temporary memory. Then using ***'np.where'*** threshold the data to made the output Y.

Made 3 D plot to visualize the Datapoints. And gave it colour according to the Y values.



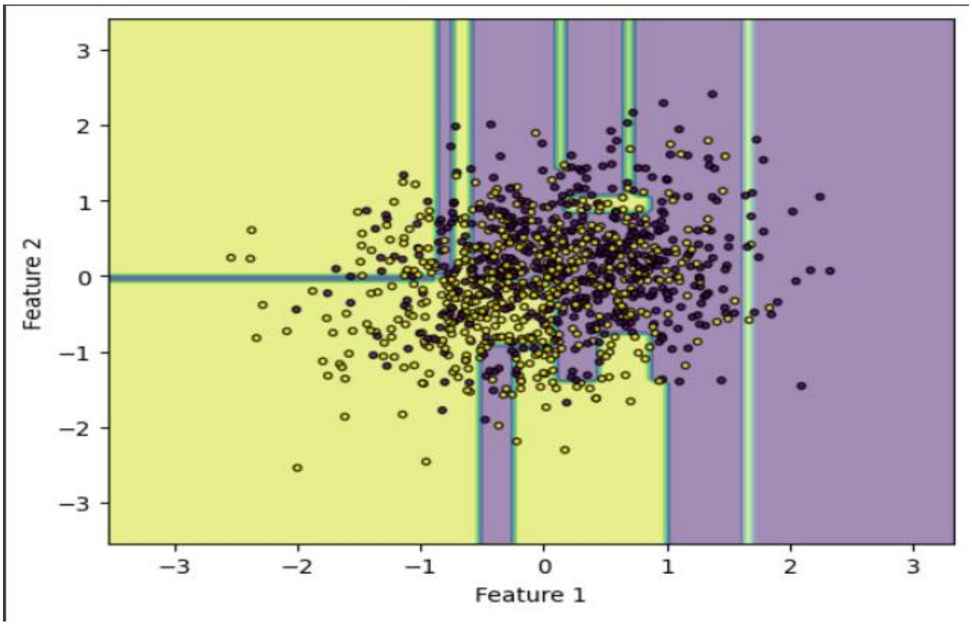
Part 2

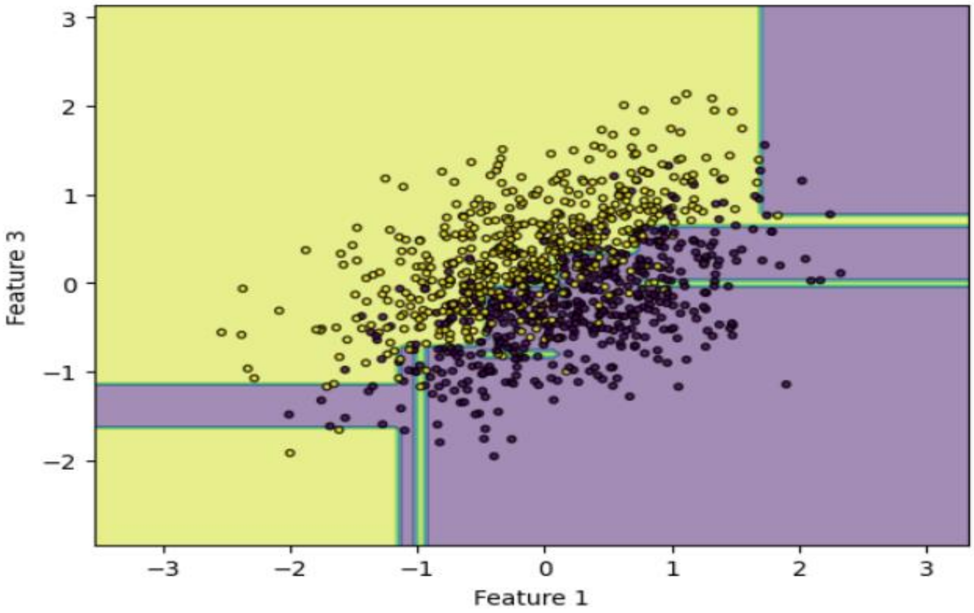
Applying PCA for n = 3, on the dataset and then transformed the data accordingly. Made a 3D the transformed data as well.

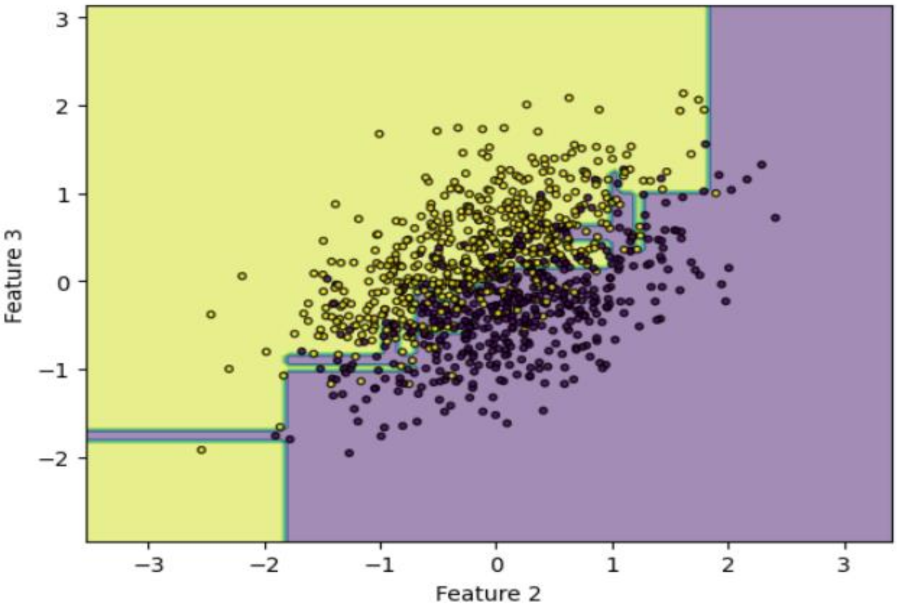


Part 3

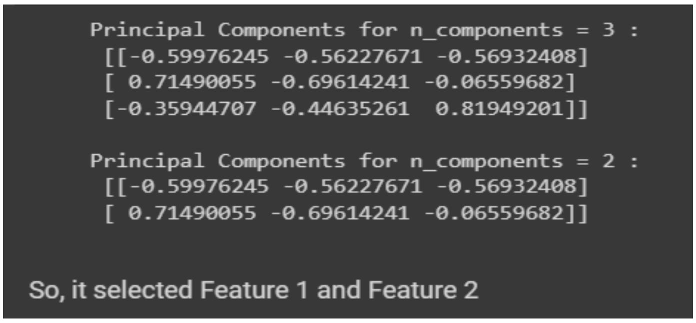
Made a function to return the subsets of given length. Then plotted the decision boundary considering just the two features and using the model as DecisionTreeClassifier. Got following Decision Boundaries.







Checked for the features selected by PCA, and found that it selected Feature 1 and Feature 2 which is the worst as can be seen in the decision boundaries.

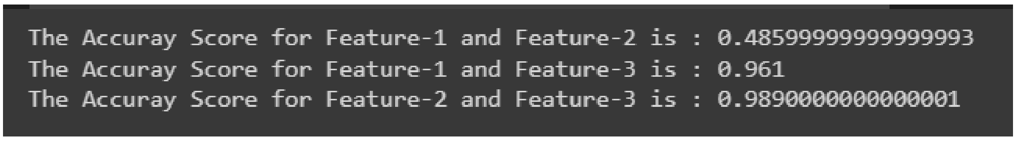


```
Principal Components for n_components = 3 :
 [[-0.59976245 -0.56227671 -0.56932408]
 [ 0.71490055 -0.69614241 -0.06559682]
 [-0.35944707 -0.44635261  0.81949201]]

Principal Components for n_components = 2 :
 [[-0.59976245 -0.56227671 -0.56932408]
 [ 0.71490055 -0.69614241 -0.06559682]]
```

So, it selected Feature 1 and Feature 2

The reason behind it PCA does not matter about the class of the data points. It just selects the feature in which the spread is maximum. Since Feature 1 and Feature 2 has maximum spread, so they are selected as principal components.

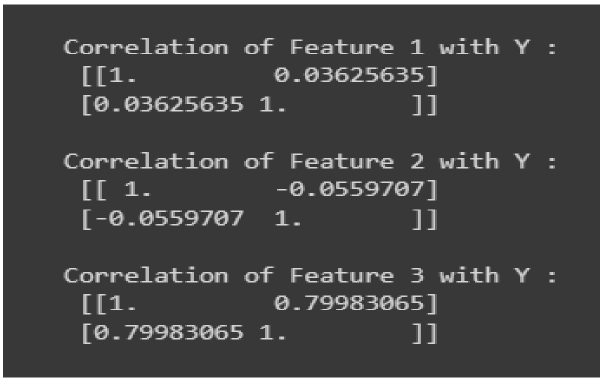
Doing Accuracy Testing of the Data, using KFold split. And got the following results.



```
The Accuray Score for Feature-1 and Feature-2 is : 0.48599999999999993
The Accuray Score for Feature-1 and Feature-3 is : 0.961
The Accuray Score for Feature-2 and Feature-3 is : 0.9890000000000001
```

As expected Feature 1 and Feature 2 performs the worst, no matter it was selected by the PCA.

Also, we can check the correlation of the Features with the output Y.



```
Correlation of Feature 1 with Y :
 [[1.         0.03625635]
 [0.03625635 1.        ]]

Correlation of Feature 2 with Y :
 [[ 1.        -0.0559707]
 [-0.0559707  1.       ]]

Correlation of Feature 3 with Y :
 [[1.         0.79983065]
 [0.79983065 1.        ]]
```

We can clearly see that Feature 1 and Feature 2 have least correlation with the output Y. Hence the combination of Feature 1 and Feature 2 is worst. At the same time, Feature 3 and Feature 2 are most correlated to the output, so gave best results as seen in the accuracy test.