Untitled12

June 9, 2024

```
[76]: #IMPORTING IMPORTANT LIBRARIES
      import pandas as pd
      import numpy as np
      from sklearn.tree import DecisionTreeRegressor
      import matplotlib.pyplot as plt
      import seaborn as sns
```

```
[77]: #LOADING THE DATASET
      df=pd.read_excel('Flyzy Flight Cancellation (4).xlsx')
```

```
[78]: #EXPLORING THE DATA
      df.head()
```

```
[78]:    Flight ID    Airline  Flight_Distance Origin_Airport Destination_Airport  \
      0    7319483  Airline D              475      Airport 3           Airport 2
      1    4791965  Airline E              538      Airport 5           Airport 4
      2    2991718  Airline C              565      Airport 1           Airport 2
      3    4220106  Airline E              658      Airport 5           Airport 3
      4    2263008  Airline E              566      Airport 2           Airport 2

         Scheduled_Departure_Time  Day_of_Week  Month Airplane_Type  Weather_Score  \
      0                         4            6      1        Type C       0.225122
      1                        12            1      6        Type B       0.060346
      2                        17            3      9        Type C       0.093920
      3                         1            1      8        Type B       0.656750
      4                        19            7     12        Type E       0.505211

         Previous_Flight_Delay_Minutes  Airline_Rating  Passenger_Load  \
      0                            5.0        2.151974        0.477202
      1                           68.0        1.600779        0.159718
      2                           18.0        4.406848        0.256803
      3                           13.0        0.998757        0.504077
      4                            4.0        3.806206        0.019638

         Flight_Cancelled
      0                 0
      1                 1
```

```
2                  0
3                  1
4                  0
```

[79]:

```
#Dropping null values
inputs=df.drop('Flight_Cancelled',axis='columns')
target=df['Flight_Cancelled']
```

[80]:

```
inputs
```

[80]:

```
      Flight ID    Airline  Flight_Distance Origin_Airport  \
0       7319483  Airline D              475      Airport 3   
1       4791965  Airline E              538      Airport 5   
2       2991718  Airline C              565      Airport 1   
3       4220106  Airline E              658      Airport 5   
4       2263008  Airline E              566      Airport 2   
…           …        …              …             …   
2995    1265781  Airline D              395      Airport 2   
2996    5440150  Airline E              547      Airport 1   
2997     779080  Airline C              461      Airport 1   
2998    4044431  Airline B              464      Airport 3   
2999    2806578  Airline A              369      Airport 1   

     Destination_Airport  Scheduled_Departure_Time  Day_of_Week  Month  \
0              Airport 2                         4            6      1   
1              Airport 4                        12            1      6   
2              Airport 2                        17            3      9   
3              Airport 3                         1            1      8   
4              Airport 2                        19            7     12   
…                    …                       …          …    …   
2995           Airport 3                         0            6      1   
2996           Airport 4                        22            4      7   
2997           Airport 3                         8            3      1   
2998           Airport 3                         5            5      3   
2999           Airport 2                         1            1     10   

     Airplane_Type  Weather_Score  Previous_Flight_Delay_Minutes  \
0           Type C       0.225122                        5.00000   
1           Type B       0.060346                       68.00000   
2           Type C       0.093920                       18.00000   
3           Type B       0.656750                       13.00000   
4           Type E       0.505211                        4.00000   
…             …            …                            …   
2995        Type B       0.190018                        1.00000   
2996        Type E       0.719271                       91.00000   
2997        Type B       0.458724                        3.00000   
2998        Type E       0.443373                       46.00000   
```

```
2999        Type A       0.704563                       18.66667

      Airline_Rating  Passenger_Load
0           2.151974        0.477202
1           1.600779        0.159718
2           4.406848        0.256803
3           0.998757        0.504077
4           3.806206        0.019638
…              …             …
2995        2.451216        0.283440
2996        0.027039        0.665294
2997        1.131315        0.991307
2998        0.968651        0.254808
2999        1.879411        0.532486

[3000 rows x 13 columns]
```

[81]:
```
target
```

[81]:
```
0       0
1       1
2       0
3       1
4       0
       ..
2995    1
2996    1
2997    0
2998    1
2999    1
Name: Flight_Cancelled, Length: 3000, dtype: int64
```

[82]:
```
df.tail()
```

[82]:
```
      Flight ID    Airline  Flight_Distance Origin_Airport  \
2995    1265781  Airline D              395      Airport 2
2996    5440150  Airline E              547      Airport 1
2997     779080  Airline C              461      Airport 1
2998    4044431  Airline B              464      Airport 3
2999    2806578  Airline A              369      Airport 1

     Destination_Airport  Scheduled_Departure_Time  Day_of_Week  Month  \
2995           Airport 3                         0            6      1
2996           Airport 4                        22            4      7
2997           Airport 3                         8            3      1
2998           Airport 3                         5            5      3
2999           Airport 2                         1            1     10
```

```
     Airplane_Type  Weather_Score  Previous_Flight_Delay_Minutes  \
2995        Type B       0.190018                        1.00000
2996        Type E       0.719271                       91.00000
2997        Type B       0.458724                        3.00000
2998        Type E       0.443373                       46.00000
2999        Type A       0.704563                       18.66667

      Airline_Rating  Passenger_Load  Flight_Cancelled
2995        2.451216        0.283440                 1
2996        0.027039        0.665294                 1
2997        1.131315        0.991307                 0
2998        0.968651        0.254808                 1
2999        1.879411        0.532486                 1
```

[83]:

```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000 entries, 0 to 2999
Data columns (total 14 columns):
 #   Column                         Non-Null Count  Dtype
---  ------                         --------------  -----
 0   Flight ID                      3000 non-null   int64
 1   Airline                        3000 non-null   object
 2   Flight_Distance                3000 non-null   int64
 3   Origin_Airport                 3000 non-null   object
 4   Destination_Airport            3000 non-null   object
 5   Scheduled_Departure_Time       3000 non-null   int64
 6   Day_of_Week                    3000 non-null   int64
 7   Month                          3000 non-null   int64
 8   Airplane_Type                  3000 non-null   object
 9   Weather_Score                  3000 non-null   float64
 10  Previous_Flight_Delay_Minutes  3000 non-null   float64
 11  Airline_Rating                 3000 non-null   float64
 12  Passenger_Load                 3000 non-null   float64
 13  Flight_Cancelled               3000 non-null   int64
dtypes: float64(4), int64(6), object(4)
memory usage: 328.2+ KB
```

[84]:

```python
#converting all null values to numeric
from sklearn.preprocessing import LabelEncoder
label_encoder=LabelEncoder()
```

[85]:

```python
df['Airline']=label_encoder.fit_transform(df['Airline'])
df['Origin_Airport']=label_encoder.fit_transform(df['Origin_Airport'])
df['Destination_Airport']=label_encoder.fit_transform(df['Destination_Airport'])
df['Airplane_Type']=label_encoder.fit_transform(df['Airplane_Type'])
```

[86]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000 entries, 0 to 2999
Data columns (total 14 columns):
 #   Column                         Non-Null Count  Dtype  
---  ------                         --------------  -----  
 0   Flight ID                      3000 non-null   int64  
 1   Airline                        3000 non-null   int64  
 2   Flight_Distance                3000 non-null   int64  
 3   Origin_Airport                 3000 non-null   int64  
 4   Destination_Airport            3000 non-null   int64  
 5   Scheduled_Departure_Time       3000 non-null   int64  
 6   Day_of_Week                    3000 non-null   int64  
 7   Month                          3000 non-null   int64  
 8   Airplane_Type                  3000 non-null   int64  
 9   Weather_Score                  3000 non-null   float64
 10  Previous_Flight_Delay_Minutes  3000 non-null   float64
 11  Airline_Rating                 3000 non-null   float64
 12  Passenger_Load                 3000 non-null   float64
 13  Flight_Cancelled               3000 non-null   int64  
dtypes: float64(4), int64(10)
memory usage: 328.2 KB
```

[87]:

```
df.isnull().sum()
```

[87]:

```
Flight ID                        0
Airline                          0
Flight_Distance                  0
Origin_Airport                   0
Destination_Airport              0
Scheduled_Departure_Time         0
Day_of_Week                      0
Month                            0
Airplane_Type                    0
Weather_Score                    0
Previous_Flight_Delay_Minutes    0
Airline_Rating                   0
Passenger_Load                   0
Flight_Cancelled                 0
dtype: int64
```

[88]:

```
df.describe()
```

[88]:

```
          Flight ID      Airline  Flight_Distance  Origin_Airport  \
count  3.000000e+03  3000.000000      3000.000000     3000.000000   
mean   4.997429e+06     1.567333       498.909333        1.631667   
```

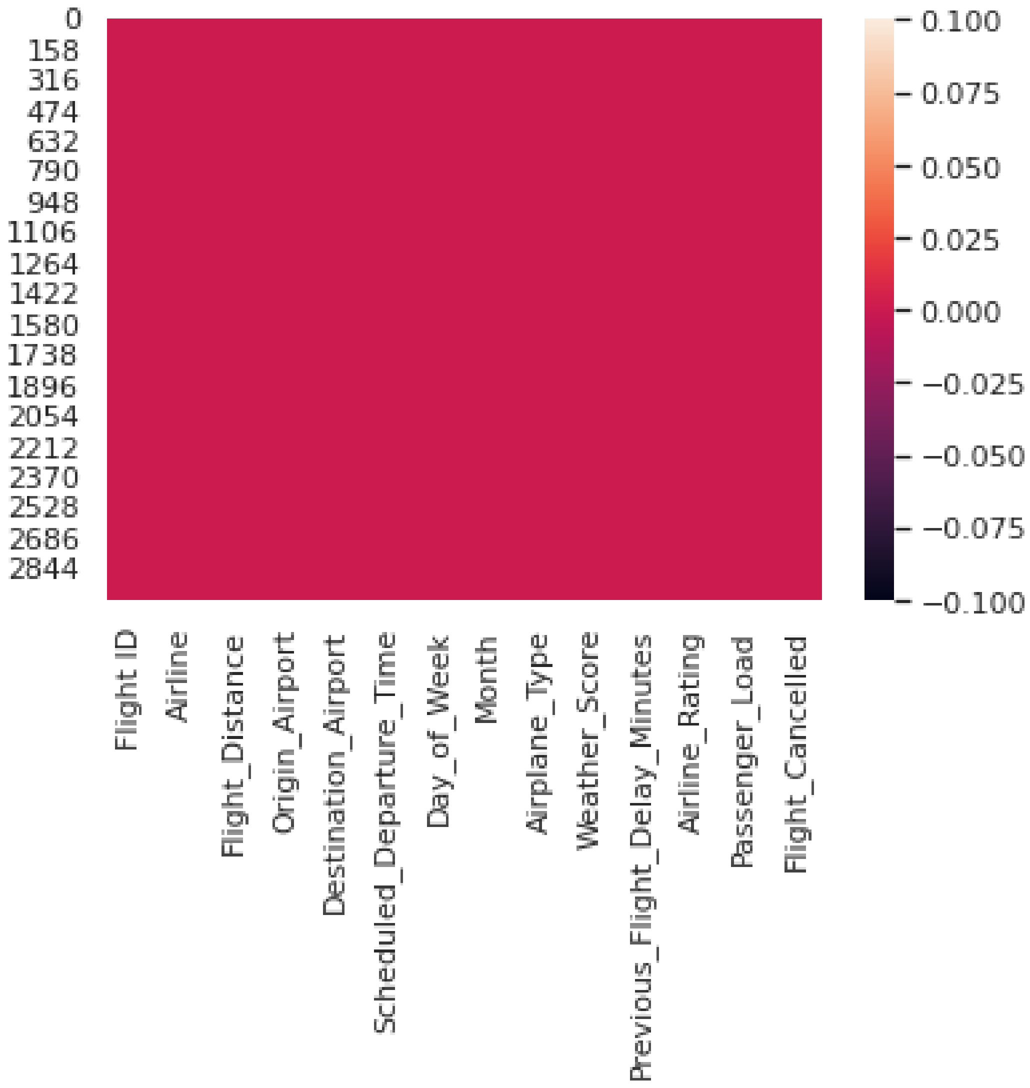
```
std    2.868139e+06     1.513350        98.892266        1.499805
min    3.681000e+03     0.000000       138.000000        0.000000
25%    2.520313e+06     0.000000       431.000000        0.000000
50%    5.073096e+06     1.000000       497.000000        1.000000
75%    7.462026e+06     3.000000       566.000000        3.000000
max    9.999011e+06     4.000000       864.000000        4.000000

       Destination_Airport  Scheduled_Departure_Time  Day_of_Week  \
count          3000.000000               3000.000000  3000.000000
mean              0.911667                 11.435000     3.963000
std               1.147012                  6.899298     2.016346
min               0.000000                  0.000000     1.000000
25%               0.000000                  6.000000     2.000000
50%               0.000000                 12.000000     4.000000
75%               2.000000                 17.000000     6.000000
max               3.000000                 23.000000     7.000000

             Month  Airplane_Type  Weather_Score  \
count  3000.000000    3000.000000    3000.000000
mean      6.381000       1.582000       0.524023
std       3.473979       1.515049       0.290694
min       1.000000       0.000000       0.000965
25%       3.000000       0.000000       0.278011
50%       6.000000       1.000000       0.522180
75%       9.000000       3.000000       0.776323
max      12.000000       4.000000       1.099246

       Previous_Flight_Delay_Minutes  Airline_Rating  Passenger_Load  \
count                    3000.000000     3000.000000     3000.000000
mean                       26.793383        2.317439        0.515885
std                        27.874733        1.430386        0.295634
min                         0.000000        0.000103        0.001039
25%                         7.000000        1.092902        0.265793
50%                        18.000000        2.126614        0.517175
75%                        38.000000        3.525746        0.770370
max                       259.000000        5.189038        1.123559

       Flight_Cancelled
count       3000.000000
mean           0.690667
std            0.462296
min            0.000000
25%            0.000000
50%            1.000000
75%            1.000000
max            1.000000
```

[89]:

```
#GETING A PLOT OF THE DATA
sns.heatmap(df.isnull())
```

[89]:

```
<AxesSubplot: >
```



[90]:

```
#Splitting the Dataset
from sklearn.model_selection import train_test_split
x=df.copy()
y=df['Flight_Cancelled']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
```

[91]:

```
from sklearn.tree import DecisionTreeClassifier
clf=DecisionTreeClassifier(max_depth=4)
```

```
clf=clf.fit(x_train, y_train)
```

[92]:
```
clf.get_params()
```

[92]:
```
{'ccp_alpha': 0.0,
 'class_weight': None,
 'criterion': 'gini',
 'max_depth': 4,
 'max_features': None,
 'max_leaf_nodes': None,
 'min_impurity_decrease': 0.0,
 'min_samples_leaf': 1,
 'min_samples_split': 2,
 'min_weight_fraction_leaf': 0.0,
 'random_state': None,
 'splitter': 'best'}
```

[93]:
```
model=tree.DecisionTreeClassifier()
```

[94]:
```
model.fit(df,target)
```

[94]:
```
DecisionTreeClassifier()
```

[97]:
```
model.score(df,target)
```

[97]:
```
1.0
```

[98]:
```
#Getting Predictions
x_test
```

[98]:
```
      Flight ID  Airline  Flight_Distance  Origin_Airport  \
803     5389318        4              320               3
346     5734151        2              479               0
2866    1243338        2              542               3
2576     386006        3              510               1
975     2638373        3              484               1
…         …      …              …             …
1100    3747269        4              646               2
2786    5974294        2              478               2
1428     675543        4              549               4
1991    5125324        0              656               3
1342    5017424        0              500               0

      Destination_Airport  Scheduled_Departure_Time  Day_of_Week  Month  \
803                     0                        10            1      5
346                     0                        11            3      6
2866                    0                         4            3      6
```

```
2576                     3                        18           1      7
975                      0                         8           2     10
…                   …                        …          …    …
1100                     0                         8           4     10
2786                     1                         0           3      6
1428                     1                         7           3      8
1991                     2                         3           6      8
1342                     0                         2           1      1

      Airplane_Type  Weather_Score  Previous_Flight_Delay_Minutes  \
803               0       0.409747                       6.000000
346               3       0.478720                       4.000000
2866              0       0.795747                      14.666670
2576              0       0.640035                       1.000000
975               0       0.125203                       2.000000
…             …            …                             …
1100              0       0.712677                      19.000000
2786              2       0.781934                      20.666670
1428              4       0.110287                      70.000000
1991              1       0.055725                      16.000000
1342              0       0.859967                       4.444444

      Airline_Rating  Passenger_Load  Flight_Cancelled
803         4.900937        0.426122                 1
346         2.034179        0.667554                 0
2866        3.856363        0.849552                 1
2576        1.606535        0.242138                 0
975         1.951069        0.813329                 0
…               …             …               …
1100        1.134719        0.464033                 0
2786        4.835026        0.741595                 1
1428        1.012879        0.948749                 1
1991        1.598190        0.329716                 0
1342        0.971944        1.023917                 1

[900 rows x 14 columns]
```

[99]:
```python
predictions=clf.predict(x_test)
predictions
```

[99]:
```
array([1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0, 0, 0, 1,
       1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1,
       1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1,
       1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1,
       0, 1, 0, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1,
       0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1,
       0, 1, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1,
       1, 1, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 1, 0, 0,
       1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1,
       0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 1, 1,
       1, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1,
       1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1,
       1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 0, 1, 1, 0, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1,
       1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 0,
       0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0, 1, 1, 1, 0, 1,
       0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1,
       1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1,
       1, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1,
       0, 0, 0, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 0,
       0, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0, 0, 1, 1,
       1, 1, 1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1,
       1, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 1, 0,
       1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 0, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 0, 1,
       1, 1, 1, 0, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1,
       1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 0,
       0, 1, 0, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1,
       0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1,
       1, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1,
       1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0,
       0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0, 1,
       1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 0,
       1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1,
       1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0,
       1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 1,
       1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1,
       0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 0, 1, 1, 1, 0, 1, 1, 0, 1])
```

[100]:
```
from sklearn.metrics import accuracy_score
accuracy_score(y_test,predictions)
```

[100]:
```
1.0
```

[101]:
```
from sklearn.metrics import confusion_matrix
confusion_matrix(y_test,predictions,labels=[0,1])
```

[101]:
```
array([[291,   0],
       [  0, 609]])
```

```
[102]: feature_names= x.columns
       feature_names
```

```
[102]: Index(['Flight ID', 'Airline', 'Flight_Distance', 'Origin_Airport',
              'Destination_Airport', 'Scheduled_Departure_Time', 'Day_of_Week',
              'Month', 'Airplane_Type', 'Weather_Score',
              'Previous_Flight_Delay_Minutes', 'Airline_Rating', 'Passenger_Load',
              'Flight_Cancelled'],
             dtype='object')
```

```
[103]: clf.feature_importances_
```

```
[103]: array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1.])
```

```
[104]: #Importing Libraries again
       from sklearn.tree import DecisionTreeClassifier,plot_tree
       from sklearn.datasets import load_iris
       import matplotlib.pyplot as plt
```

```
[105]: #Model Comparison
       iris=load_iris()
       x,y=iris.data,iris.target
```

```
[106]: feature_names= iris.feature_names
```

```
[ ]: plt.figure(figsize=(25,30))
     plot_tree(clf,feature_names=feature_names,filled=True,rounded=True)
     plt.show()
```

```
[ ]:
```