assignment-4

21BAI10418

```
[187]: #Import the Libraries.
       import numpy as np
       import pandas as pd
       import matplotlib.pyplot as plt
       import seaborn as sns
```

```
[188]: df=pd.read_csv("WA_Fn-UseC_-HR-Employee-Attrition.csv")
```

```
[189]: df.shape
```

```
[189]: (1470, 35)
```

```
[190]: df.head()
```

```
[190]:    Age Attrition     BusinessTravel  DailyRate              Department  \
       0   41       Yes      Travel_Rarely       1102                   Sales
       1   49        No  Travel_Frequently        279  Research & Development
       2   37       Yes      Travel_Rarely       1373  Research & Development
       3   33        No  Travel_Frequently       1392  Research & Development
       4   27        No      Travel_Rarely        591  Research & Development

          DistanceFromHome  Education EducationField  EmployeeCount  EmployeeNumber  \
       0                 1          2  Life Sciences              1               1
       1                 8          1  Life Sciences              1               2
       2                 2          2          Other              1               4
       3                 3          4  Life Sciences              1               5
       4                 2          1        Medical              1               7

          …  RelationshipSatisfaction StandardHours  StockOptionLevel  \
       0  …                         1            80                 0
       1  …                         4            80                 1
       2  …                         2            80                 0
       3  …                         3            80                 0
       4  …                         4            80                 1

          TotalWorkingYears  TrainingTimesLastYear WorkLifeBalance  YearsAtCompany  \
       0                  8                      0               1               6
```

```
1                  10                      3               3             10
2                   7                      3               3              0
3                   8                      3               3              8
4                   6                      3               3              2

  YearsInCurrentRole  YearsSinceLastPromotion  YearsWithCurrManager
0                  4                        0                     5
1                  7                        1                     7
2                  0                        0                     0
3                  7                        3                     0
4                  2                        2                     2

[5 rows x 35 columns]
```

[191]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1470 entries, 0 to 1469
Data columns (total 35 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Age                       1470 non-null   int64
 1   Attrition                 1470 non-null   object
 2   BusinessTravel            1470 non-null   object
 3   DailyRate                 1470 non-null   int64
 4   Department                1470 non-null   object
 5   DistanceFromHome          1470 non-null   int64
 6   Education                 1470 non-null   int64
 7   EducationField            1470 non-null   object
 8   EmployeeCount             1470 non-null   int64
 9   EmployeeNumber            1470 non-null   int64
 10  EnvironmentSatisfaction   1470 non-null   int64
 11  Gender                    1470 non-null   object
 12  HourlyRate                1470 non-null   int64
 13  JobInvolvement            1470 non-null   int64
 14  JobLevel                  1470 non-null   int64
 15  JobRole                   1470 non-null   object
 16  JobSatisfaction           1470 non-null   int64
 17  MaritalStatus             1470 non-null   object
 18  MonthlyIncome             1470 non-null   int64
 19  MonthlyRate               1470 non-null   int64
 20  NumCompaniesWorked        1470 non-null   int64
 21  Over18                    1470 non-null   object
 22  OverTime                  1470 non-null   object
 23  PercentSalaryHike         1470 non-null   int64
 24  PerformanceRating         1470 non-null   int64
 25  RelationshipSatisfaction  1470 non-null   int64
 26  StandardHours             1470 non-null   int64 
 27  StockOptionLevel          1470 non-null   int64 
 28  TotalWorkingYears         1470 non-null   int64 
 29  TrainingTimesLastYear     1470 non-null   int64 
 30  WorkLifeBalance           1470 non-null   int64 
 31  YearsAtCompany            1470 non-null   int64 
 32  YearsInCurrentRole        1470 non-null   int64 
 33  YearsSinceLastPromotion   1470 non-null   int64 
 34  YearsWithCurrManager      1470 non-null   int64 
dtypes: int64(26), object(9)
memory usage: 402.1+ KB
```

[192]:
```
df.describe()
```

[192]:
```
               Age    DailyRate  DistanceFromHome    Education  EmployeeCount  \
count  1470.000000  1470.000000       1470.000000  1470.000000         1470.0
mean     36.923810   802.485714          9.192517     2.912925            1.0
std       9.135373   403.509100          8.106864     1.024165            0.0
min      18.000000   102.000000          1.000000     1.000000            1.0
25%      30.000000   465.000000          2.000000     2.000000            1.0
50%      36.000000   802.000000          7.000000     3.000000            1.0
75%      43.000000  1157.000000         14.000000     4.000000            1.0
max      60.000000  1499.000000         29.000000     5.000000            1.0

       EmployeeNumber  EnvironmentSatisfaction   HourlyRate  JobInvolvement  \
count     1470.000000              1470.000000  1470.000000     1470.000000
mean      1024.865306                 2.721769    65.891156        2.729932
std        602.024335                 1.093082    20.329428        0.711561
min          1.000000                 1.000000    30.000000        1.000000
25%        491.250000                 2.000000    48.000000        2.000000
50%       1020.500000                 3.000000    66.000000        3.000000
75%       1555.750000                 4.000000    83.750000        3.000000
max       2068.000000                 4.000000   100.000000        4.000000

          JobLevel  …  RelationshipSatisfaction  StandardHours  \
count  1470.000000  …               1470.000000         1470.0
mean      2.063946  …                  2.712245           80.0
std       1.106940  …                  1.081209            0.0
min       1.000000  …                  1.000000           80.0
25%       1.000000  …                  2.000000           80.0
50%       2.000000  …                  3.000000           80.0
75%       3.000000  …                  4.000000           80.0
max       5.000000  …                  4.000000           80.0

       StockOptionLevel  TotalWorkingYears  TrainingTimesLastYear  \
count       1470.000000        1470.000000            1470.000000
mean           0.793878          11.279592               2.799320
```

```
std            0.852077           7.780782               1.289271
min            0.000000           0.000000               0.000000
25%            0.000000           6.000000               2.000000
50%            1.000000          10.000000               3.000000
75%            1.000000          15.000000               3.000000
max            3.000000          40.000000               6.000000

       WorkLifeBalance  YearsAtCompany  YearsInCurrentRole  \
count      1470.000000     1470.000000         1470.000000
mean          2.761224        7.008163            4.229252
std           0.706476        6.126525            3.623137
min           1.000000        0.000000            0.000000
25%           2.000000        3.000000            2.000000
50%           3.000000        5.000000            3.000000
75%           3.000000        9.000000            7.000000
max           4.000000       40.000000           18.000000

       YearsSinceLastPromotion  YearsWithCurrManager
count              1470.000000           1470.000000
mean                  2.187755              4.123129
std                   3.222430              3.568136
min                   0.000000              0.000000
25%                   0.000000              2.000000
50%                   1.000000              3.000000
75%                   3.000000              7.000000
max                  15.000000             17.000000

[8 rows x 26 columns]
```

1 Droping the Unnesscary Colums

[193]:
```
df.drop(columns=['EmployeeCount', 'Over18', 'StandardHours','EmployeeNumber'], inplace=True)
df.shape
```

[193]:
```
(1470, 31)
```

2 Checking and hanldeling null values

[194]:
```
df.isnull().any()
```

[194]:
```
Age                         False
Attrition                   False
BusinessTravel              False
DailyRate                   False
```

```
Department                  False
DistanceFromHome            False
Education                   False
EducationField              False
EnvironmentSatisfaction     False
Gender                      False
HourlyRate                  False
JobInvolvement              False
JobLevel                    False
JobRole                     False
JobSatisfaction             False
MaritalStatus               False
MonthlyIncome               False
MonthlyRate                 False
NumCompaniesWorked          False
OverTime                    False
PercentSalaryHike           False
PerformanceRating           False
RelationshipSatisfaction    False
StockOptionLevel            False
TotalWorkingYears           False
TrainingTimesLastYear       False
WorkLifeBalance             False
YearsAtCompany              False
YearsInCurrentRole          False
YearsSinceLastPromotion     False
YearsWithCurrManager        False
dtype: bool
```

[195]:

```
print(df.isna().sum())
'''Therfore There are no null values Present in the dataset'''
```

```
Age                         0
Attrition                   0
BusinessTravel              0
DailyRate                   0
Department                  0
DistanceFromHome            0
Education                   0
EducationField              0
EnvironmentSatisfaction     0
Gender                      0
HourlyRate                  0
JobInvolvement              0
JobLevel                    0
JobRole                     0
JobSatisfaction             0
```

```
MaritalStatus               0
MonthlyIncome               0
MonthlyRate                 0
NumCompaniesWorked          0
OverTime                    0
PercentSalaryHike           0
PerformanceRating           0
RelationshipSatisfaction    0
StockOptionLevel            0
TotalWorkingYears           0
TrainingTimesLastYear       0
WorkLifeBalance             0
YearsAtCompany              0
YearsInCurrentRole          0
YearsSinceLastPromotion     0
YearsWithCurrManager        0
dtype: int64
```

```
[195]: 'Therfore There are no null values Present in the dataset'
```

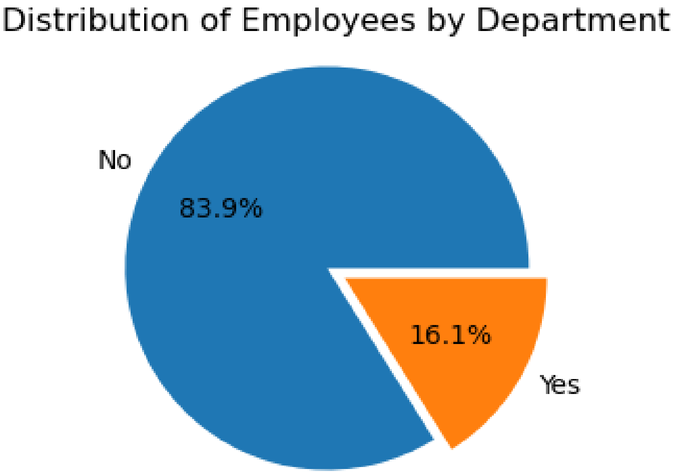
```
[196]: #check for Duplicates
       print(df.duplicated().sum())
       '''Therfore There are no duplicate values Present in the dataset'''
```

```
0
```

```
[196]: 'Therfore There are no duplicate values Present in the dataset'
```

3 Data Visualization

```
[197]: department_counts = df['Attrition'].value_counts()
       plt.figure(figsize=(4, 3))
       Explode=[0.1,0]
       plt.pie(department_counts, labels=department_counts.
         ↪index,explode=Explode,autopct='%1.1f%%')
       plt.title('Distribution of Employees by Department')
       plt.axis('equal')
       plt.show()
```



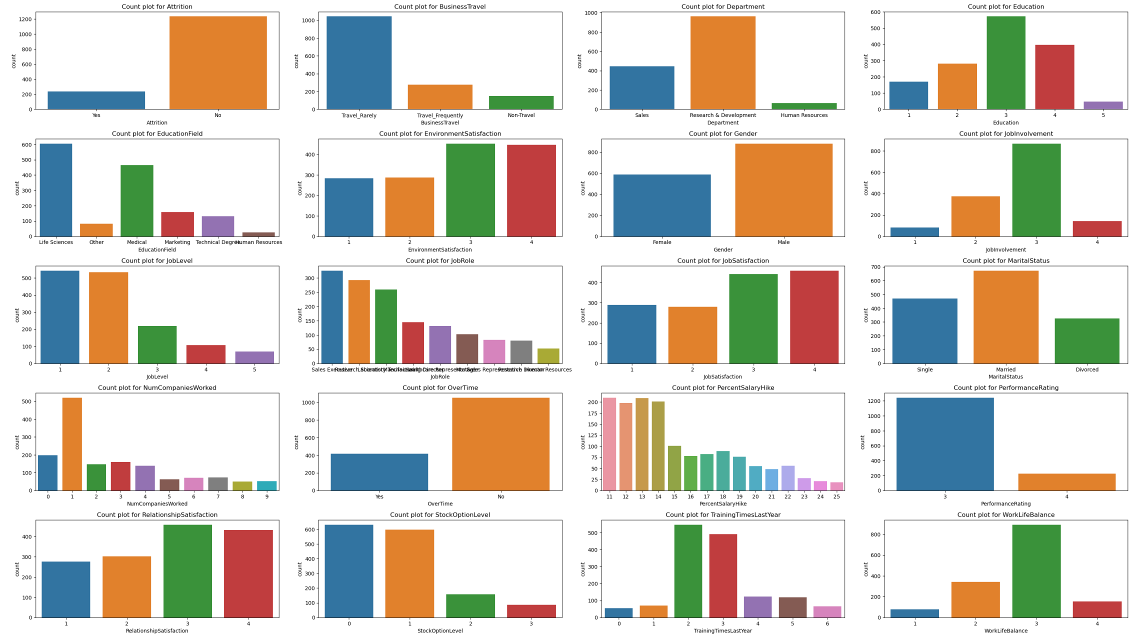
[198]:
```
df.columns
```

[198]:
```
Index(['Age', 'Attrition', 'BusinessTravel', 'DailyRate', 'Department',
       'DistanceFromHome', 'Education', 'EducationField',
       'EnvironmentSatisfaction', 'Gender', 'HourlyRate', 'JobInvolvement',
       'JobLevel', 'JobRole', 'JobSatisfaction', 'MaritalStatus',
       'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked', 'OverTime',
       'PercentSalaryHike', 'PerformanceRating', 'RelationshipSatisfaction',
       'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear',
       'WorkLifeBalance', 'YearsAtCompany', 'YearsInCurrentRole',
       'YearsSinceLastPromotion', 'YearsWithCurrManager'],
      dtype='object')
```

[199]:
```
columns=['Attrition', 'BusinessTravel', 'Department', 'Education',␣
 ↪'EducationField','EnvironmentSatisfaction',␣
 ↪'Gender','JobInvolvement','JobLevel', 'JobRole', 'JobSatisfaction',␣
 ↪'MaritalStatus', 'NumCompaniesWorked', 'OverTime','PercentSalaryHike',␣
 ↪'PerformanceRating', 'RelationshipSatisfaction','StockOptionLevel',␣
 ↪'TrainingTimesLastYear','WorkLifeBalance']
```

[200]:
```
#visualizing the categorical attributes

# Create subplots for count plots
plt.figure(figsize=(30, 20))  # Adjust the figure size

for i, column in enumerate(columns, start=1):
    plt.subplot(6, 4, i)  # Adjust the subplot layout (3 rows, 4 columns)
    sns.countplot(data=df, x=column)
    plt.title(f'Count plot for {column}')
```

```
# Adjust layout
plt.tight_layout()

# Show the plots
plt.show()
```



[201]:

```
#visualizing the continous distrubution attributes
dist_columns =['Age','MonthlyIncome', 'DailyRate', 'HourlyRate',␣
↪'MonthlyRate','TotalWorkingYears', 'YearsAtCompany', 'YearsInCurrentRole',␣
↪'YearsSinceLastPromotion', 'YearsWithCurrManager']
```

[202]:

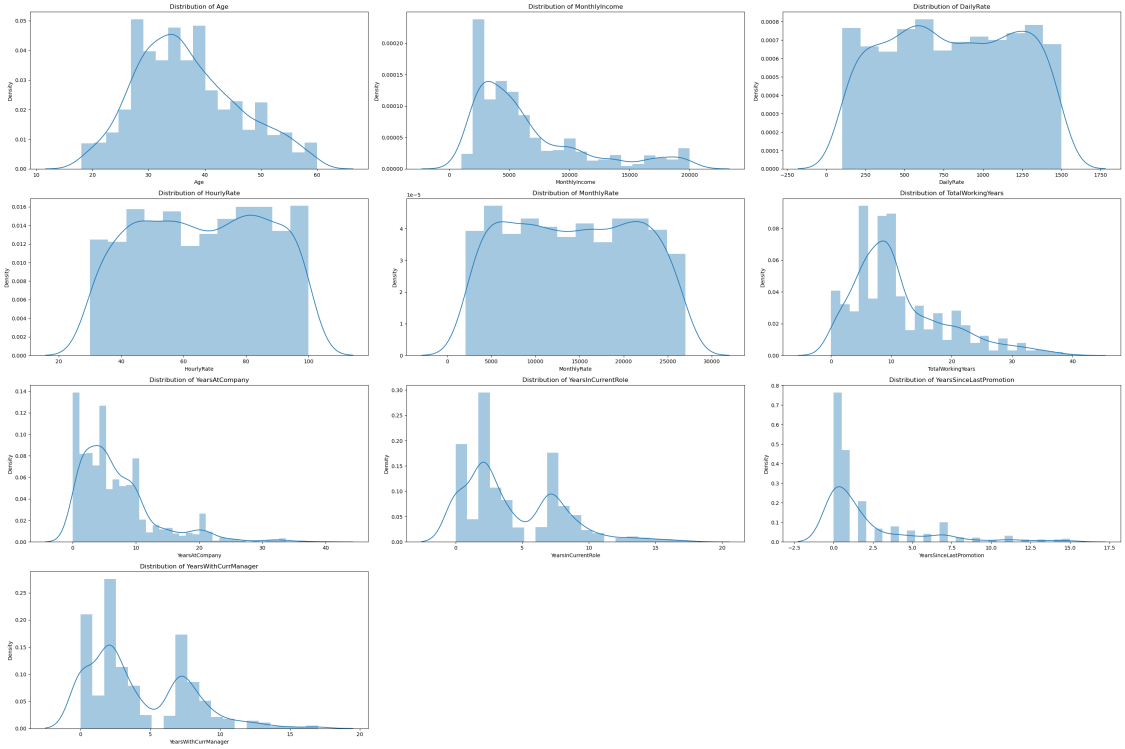
```
#visualizing the distribution of numerical attributes

plt.figure(figsize=(30, 20))
for i, column in enumerate(dist_columns, start=1):
    plt.subplot(4, 3, i)
    sns.distplot(df[column],kde=True)
    plt.title(f'Distribution of {column}')
plt.tight_layout()
plt.show()
```

```
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\1121139151.py:6: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).
```

```
For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df[column],kde=True)
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\1121139151.py:6: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df[column],kde=True)
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\1121139151.py:6: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df[column],kde=True)
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\1121139151.py:6: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df[column],kde=True)
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\1121139151.py:6: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751
```

```
  sns.distplot(df[column],kde=True)
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\1121139151.py:6: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df[column],kde=True)
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\1121139151.py:6: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df[column],kde=True)
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\1121139151.py:6: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df[column],kde=True)
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\1121139151.py:6: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df[column],kde=True)
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\1121139151.py:6: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.
```

```
Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(df[column],kde=True)
```



```
[203]: df.corr()
```

```
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\1134722465.py:1: FutureWarning:
The default value of numeric_only in DataFrame.corr is deprecated. In a future
version, it will default to False. Select only valid columns or specify the
value of numeric_only to silence this warning.
  df.corr()
```

```
[203]:                                  Age  DailyRate  DistanceFromHome  Education  \
        Age                         1.000000   0.010661         -0.001686   0.208034
        DailyRate                   0.010661   1.000000         -0.004985  -0.016806
        DistanceFromHome           -0.001686  -0.004985          1.000000   0.021042
        Education                   0.208034  -0.016806          0.021042   1.000000
        EnvironmentSatisfaction     0.010146   0.018355         -0.016075  -0.027128
        HourlyRate                  0.024287   0.023381          0.031131   0.016775
```

```
JobInvolvement            0.029820   0.046135          0.008783   0.042438
JobLevel                  0.509604   0.002966          0.005303   0.101589
JobSatisfaction          -0.004892   0.030571         -0.003669  -0.011296
MonthlyIncome             0.497855   0.007707         -0.017014   0.094961
MonthlyRate               0.028051  -0.032182          0.027473  -0.026084
NumCompaniesWorked        0.299635   0.038153         -0.029251   0.126317
PercentSalaryHike         0.003634   0.022704          0.040235  -0.011111
PerformanceRating         0.001904   0.000473          0.027110  -0.024539
RelationshipSatisfaction  0.053535   0.007846          0.006557  -0.009118
StockOptionLevel          0.037510   0.042143          0.044872   0.018422
TotalWorkingYears         0.680381   0.014515          0.004628   0.148280
TrainingTimesLastYear    -0.019621   0.002453         -0.036942  -0.025100
WorkLifeBalance          -0.021490  -0.037848         -0.026556   0.009819
YearsAtCompany            0.311309  -0.034055          0.009508   0.069114
YearsInCurrentRole        0.212901   0.009932          0.018845   0.060236
YearsSinceLastPromotion   0.216513  -0.033229          0.010029   0.054254
YearsWithCurrManager      0.202089  -0.026363          0.014406   0.069065

                          EnvironmentSatisfaction  HourlyRate  JobInvolvement  \
Age                                      0.010146    0.024287        0.029820
DailyRate                                0.018355    0.023381        0.046135
DistanceFromHome                        -0.016075    0.031131        0.008783
Education                               -0.027128    0.016775        0.042438
EnvironmentSatisfaction                  1.000000   -0.049857       -0.008278
HourlyRate                              -0.049857    1.000000        0.042861
JobInvolvement                          -0.008278    0.042861        1.000000
JobLevel                                 0.001212   -0.027853       -0.012630
JobSatisfaction                         -0.006784   -0.071335       -0.021476
MonthlyIncome                           -0.006259   -0.015794       -0.015271
MonthlyRate                              0.037600   -0.015297       -0.016322
NumCompaniesWorked                       0.012594    0.022157        0.015012
PercentSalaryHike                       -0.031701   -0.009062       -0.017205
PerformanceRating                       -0.029548   -0.002172       -0.029071
RelationshipSatisfaction                 0.007665    0.001330        0.034297
StockOptionLevel                         0.003432    0.050263        0.021523
TotalWorkingYears                       -0.002693   -0.002334       -0.005533
TrainingTimesLastYear                   -0.019359   -0.008548       -0.015338
WorkLifeBalance                          0.027627   -0.004607       -0.014617
YearsAtCompany                           0.001458   -0.019582       -0.021355
YearsInCurrentRole                       0.018007   -0.024106        0.008717
YearsSinceLastPromotion                  0.016194   -0.026716       -0.024184
YearsWithCurrManager                    -0.004999   -0.020123        0.025976

                          JobLevel  JobSatisfaction  MonthlyIncome  …  \
Age                       0.509604        -0.004892       0.497855  …
DailyRate                 0.002966         0.030571       0.007707  …
DistanceFromHome          0.005303        -0.003669      -0.017014  …
```

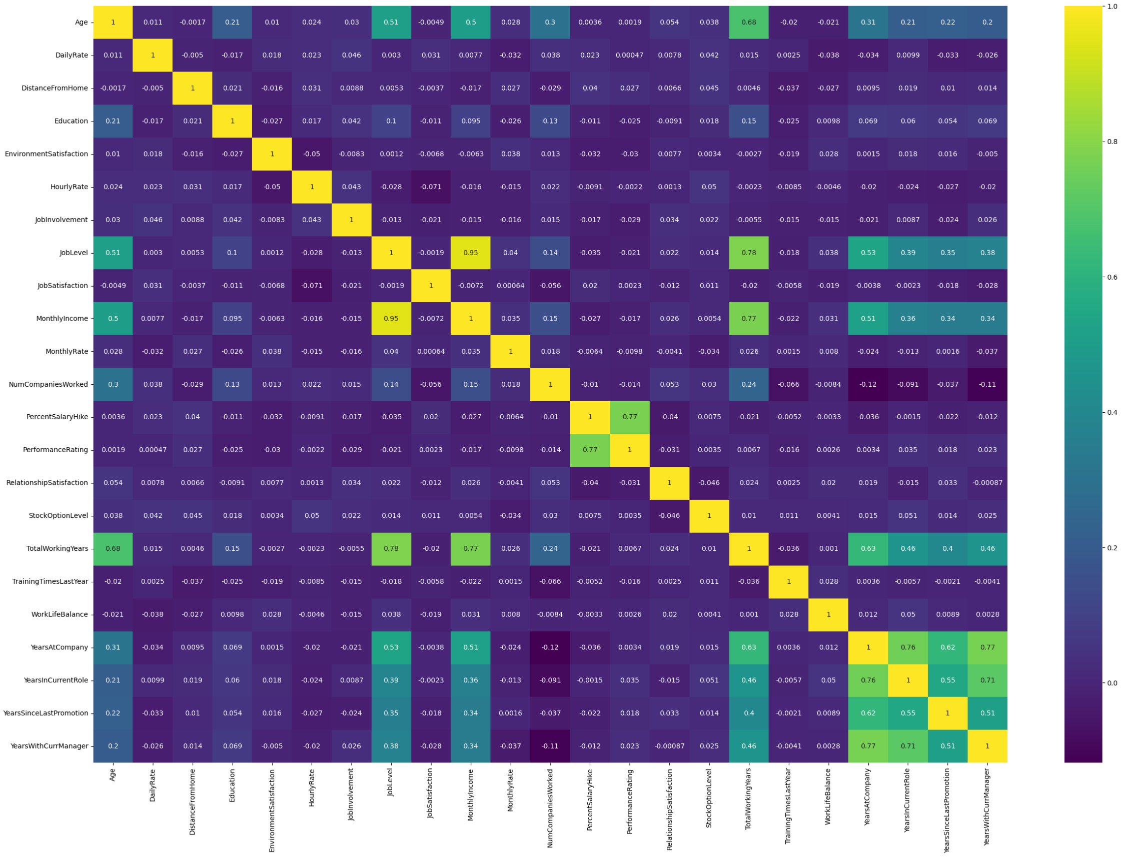
```
Education                 0.101589        -0.011296       0.094961  …
EnvironmentSatisfaction   0.001212        -0.006784      -0.006259  …
HourlyRate               -0.027853        -0.071335      -0.015794  …
JobInvolvement           -0.012630        -0.021476      -0.015271  …
JobLevel                  1.000000        -0.001944       0.950300  …
JobSatisfaction          -0.001944         1.000000      -0.007157  …
MonthlyIncome             0.950300        -0.007157       1.000000  …
MonthlyRate               0.039563         0.000644       0.034814  …
NumCompaniesWorked        0.142501        -0.055699       0.149515  …
PercentSalaryHike        -0.034730         0.020002      -0.027269  …
PerformanceRating        -0.021222         0.002297      -0.017120  …
RelationshipSatisfaction  0.021642        -0.012454       0.025873  …
StockOptionLevel          0.013984         0.010690       0.005408  …
TotalWorkingYears         0.782208        -0.020185       0.772893  …
TrainingTimesLastYear    -0.018191        -0.005779      -0.021736  …
WorkLifeBalance           0.037818        -0.019459       0.030683  …
YearsAtCompany            0.534739        -0.003803       0.514285  …
YearsInCurrentRole        0.389447        -0.002305       0.363818  …
YearsSinceLastPromotion   0.353885        -0.018214       0.344978  …
YearsWithCurrManager      0.375281        -0.027656       0.344079  …

                          PerformanceRating  RelationshipSatisfaction  \
Age                                0.001904                  0.053535
DailyRate                          0.000473                  0.007846
DistanceFromHome                   0.027110                  0.006557
Education                         -0.024539                 -0.009118
EnvironmentSatisfaction           -0.029548                  0.007665
HourlyRate                        -0.002172                  0.001330
JobInvolvement                    -0.029071                  0.034297
JobLevel                          -0.021222                  0.021642
JobSatisfaction                    0.002297                 -0.012454
MonthlyIncome                     -0.017120                  0.025873
MonthlyRate                       -0.009811                 -0.004085
NumCompaniesWorked                -0.014095                  0.052733
PercentSalaryHike                  0.773550                 -0.040490
PerformanceRating                  1.000000                 -0.031351
RelationshipSatisfaction          -0.031351                  1.000000
StockOptionLevel                   0.003506                 -0.045952
TotalWorkingYears                  0.006744                  0.024054
TrainingTimesLastYear             -0.015579                  0.002497
WorkLifeBalance                    0.002572                  0.019604
YearsAtCompany                     0.003435                  0.019367
YearsInCurrentRole                 0.034986                 -0.015123
YearsSinceLastPromotion            0.017896                  0.033493
YearsWithCurrManager               0.022827                 -0.000867

                          StockOptionLevel  TotalWorkingYears  \
Age                                0.037510           0.680381
DailyRate                          0.042143           0.014515
DistanceFromHome                   0.044872           0.004628
Education                          0.018422           0.148280
EnvironmentSatisfaction            0.003432          -0.002693
HourlyRate                         0.050263          -0.002334
JobInvolvement                     0.021523          -0.005533
JobLevel                           0.013984           0.782208
JobSatisfaction                    0.010690          -0.020185
MonthlyIncome                      0.005408           0.772893
MonthlyRate                       -0.034323           0.026442
NumCompaniesWorked                 0.030075           0.237639
PercentSalaryHike                  0.007528          -0.020608
PerformanceRating                  0.003506           0.006744
RelationshipSatisfaction          -0.045952           0.024054
StockOptionLevel                   1.000000           0.010136
TotalWorkingYears                  0.010136           1.000000
TrainingTimesLastYear              0.011274          -0.035662
WorkLifeBalance                    0.004129           0.001008
YearsAtCompany                     0.015058           0.628133
YearsInCurrentRole                 0.050818           0.460365
YearsSinceLastPromotion            0.014352           0.404858
YearsWithCurrManager               0.024698           0.459188

                          TrainingTimesLastYear  WorkLifeBalance  \
Age                                   -0.019621        -0.021490
DailyRate                              0.002453        -0.037848
DistanceFromHome                      -0.036942        -0.026556
Education                             -0.025100         0.009819
EnvironmentSatisfaction               -0.019359         0.027627
HourlyRate                            -0.008548        -0.004607
JobInvolvement                        -0.015338        -0.014617
JobLevel                              -0.018191         0.037818
JobSatisfaction                       -0.005779        -0.019459
MonthlyIncome                         -0.021736         0.030683
MonthlyRate                            0.001467         0.007963
NumCompaniesWorked                    -0.066054        -0.008366
PercentSalaryHike                     -0.005221        -0.003280
PerformanceRating                     -0.015579         0.002572
RelationshipSatisfaction               0.002497         0.019604
StockOptionLevel                       0.011274         0.004129
TotalWorkingYears                     -0.035662         0.001008
TrainingTimesLastYear                  1.000000         0.028072
WorkLifeBalance                        0.028072         1.000000
YearsAtCompany                         0.003569         0.012089
YearsInCurrentRole                    -0.005738         0.049856
YearsSinceLastPromotion               -0.002067         0.008941
```

```
YearsWithCurrManager                  -0.004096         0.002759

                          YearsAtCompany  YearsInCurrentRole  \
Age                             0.311309            0.212901
DailyRate                      -0.034055            0.009932
DistanceFromHome                0.009508            0.018845
Education                       0.069114            0.060236
EnvironmentSatisfaction         0.001458            0.018007
HourlyRate                     -0.019582           -0.024106
JobInvolvement                 -0.021355            0.008717
JobLevel                        0.534739            0.389447
JobSatisfaction                -0.003803           -0.002305
MonthlyIncome                   0.514285            0.363818
MonthlyRate                    -0.023655           -0.012815
NumCompaniesWorked             -0.118421           -0.090754
PercentSalaryHike              -0.035991           -0.001520
PerformanceRating               0.003435            0.034986
RelationshipSatisfaction        0.019367           -0.015123
StockOptionLevel                0.015058            0.050818
TotalWorkingYears               0.628133            0.460365
TrainingTimesLastYear           0.003569           -0.005738
WorkLifeBalance                 0.012089            0.049856
YearsAtCompany                  1.000000            0.758754
YearsInCurrentRole              0.758754            1.000000
YearsSinceLastPromotion         0.618409            0.548056
YearsWithCurrManager            0.769212            0.714365

                          YearsSinceLastPromotion  YearsWithCurrManager
Age                                      0.216513              0.202089
DailyRate                               -0.033229             -0.026363
DistanceFromHome                         0.010029              0.014406
Education                                0.054254              0.069065
EnvironmentSatisfaction                  0.016194             -0.004999
HourlyRate                              -0.026716             -0.020123
JobInvolvement                          -0.024184              0.025976
JobLevel                                 0.353885              0.375281
JobSatisfaction                         -0.018214             -0.027656
MonthlyIncome                            0.344978              0.344079
MonthlyRate                              0.001567             -0.036746
NumCompaniesWorked                      -0.036814             -0.110319
PercentSalaryHike                       -0.022154             -0.011985
PerformanceRating                        0.017896              0.022827
RelationshipSatisfaction                 0.033493             -0.000867
StockOptionLevel                         0.014352              0.024698
TotalWorkingYears                        0.404858              0.459188
TrainingTimesLastYear                   -0.002067             -0.004096
WorkLifeBalance                          0.008941              0.002759
```

```
YearsAtCompany                                 0.618409             0.769212
YearsInCurrentRole                             0.548056             0.714365
YearsSinceLastPromotion                        1.000000             0.510224
YearsWithCurrManager                           0.510224             1.000000

[23 rows x 23 columns]
```

[204]:
```
# Heat maps based on the corelation betwen all the varibles
plt.figure(figsize=(30, 20))
sns.heatmap(df.corr(),annot=True,cmap="viridis")
plt.show()
```

```
C:\Users\adhar\AppData\Local\Temp\ipykernel_1760\3278965269.py:3: FutureWarning:
The default value of numeric_only in DataFrame.corr is deprecated. In a future
version, it will default to False. Select only valid columns or specify the
value of numeric_only to silence this warning.
  sns.heatmap(df.corr(),annot=True,cmap="viridis")
```



[205]:
```
#encoding Attrition
from sklearn.preprocessing import LabelEncoder
```

```
le=LabelEncoder()
df["Attrition"]=le.fit_transform(df["Attrition"])
print(le.classes_)
```

```
['No' 'Yes']
```

[206]:
```
mapping=dict(zip(le.classes_,range(len(le.classes_))))
mapping
```

[206]:
```
{'No': 0, 'Yes': 1}
```

4 Splitting Dependent and Independent variables

[207]:
```
df.head()
```

[207]:
```
   Age  Attrition     BusinessTravel  DailyRate              Department  \
0   41          1      Travel_Rarely       1102                   Sales
1   49          0  Travel_Frequently        279  Research & Development
2   37          1      Travel_Rarely       1373  Research & Development
3   33          0  Travel_Frequently       1392  Research & Development
4   27          0      Travel_Rarely        591  Research & Development

   DistanceFromHome  Education EducationField  EnvironmentSatisfaction  \
0                 1          2  Life Sciences                        2
1                 8          1  Life Sciences                        3
2                 2          2          Other                        4
3                 3          4  Life Sciences                        4
4                 2          1        Medical                        1

   Gender  …  PerformanceRating  RelationshipSatisfaction  StockOptionLevel  \
0  Female  …                  3                         1                 0
1    Male  …                  4                         4                 1
2    Male  …                  3                         2                 0
3  Female  …                  3                         3                 0
4    Male  …                  3                         4                 1

  TotalWorkingYears  TrainingTimesLastYear WorkLifeBalance  YearsAtCompany  \
0                 8                      0               1               6
1                10                      3               3              10
2                 7                      3               3               0
3                 8                      3               3               8
4                 6                      3               3               2

   YearsInCurrentRole  YearsSinceLastPromotion YearsWithCurrManager
0                   4                        0                    5
1                   7                        1                    7
```

```
2                   0                        0                    0
3                   7                        3                    0
4                   2                        2                    2

[5 rows x 31 columns]
```

[208]:

```
X=df.drop(columns=["Attrition"],axis=1)
X.head()
```

[208]:

```
   Age     BusinessTravel  DailyRate              Department  \
0   41      Travel_Rarely       1102                   Sales
1   49  Travel_Frequently        279  Research & Development
2   37      Travel_Rarely       1373  Research & Development
3   33  Travel_Frequently       1392  Research & Development
4   27      Travel_Rarely        591  Research & Development

   DistanceFromHome  Education EducationField  EnvironmentSatisfaction  \
0                 1          2  Life Sciences                        2
1                 8          1  Life Sciences                        3
2                 2          2          Other                        4
3                 3          4  Life Sciences                        4
4                 2          1        Medical                        1

   Gender  HourlyRate  …  PerformanceRating  RelationshipSatisfaction  \
0  Female          94  …                  3                         1
1    Male          61  …                  4                         4
2    Male          92  …                  3                         2
3  Female          56  …                  3                         3
4    Male          40  …                  3                         4

  StockOptionLevel  TotalWorkingYears TrainingTimesLastYear  WorkLifeBalance  \
0                0                  8                     0                1
1                1                 10                     3                3
2                0                  7                     3                3
3                0                  8                     3                3
4                1                  6                     3                3

   YearsAtCompany  YearsInCurrentRole YearsSinceLastPromotion  \
0               6                   4                       0
1              10                   7                       1
2               0                   0                       0
3               8                   7                       3
4               2                   2                       2

   YearsWithCurrManager
0                     5
1                     7
```

```
2                      0
3                      0
4                      2

 [5 rows x 30 columns]
```

[209]:

```
print(X.shape,X.info())
print(type(X))
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1470 entries, 0 to 1469
Data columns (total 30 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   Age                       1470 non-null   int64
 1   BusinessTravel            1470 non-null   object
 2   DailyRate                 1470 non-null   int64
 3   Department                1470 non-null   object
 4   DistanceFromHome          1470 non-null   int64
 5   Education                 1470 non-null   int64
 6   EducationField            1470 non-null   object
 7   EnvironmentSatisfaction   1470 non-null   int64
 8   Gender                    1470 non-null   object
 9   HourlyRate                1470 non-null   int64
 10  JobInvolvement            1470 non-null   int64
 11  JobLevel                  1470 non-null   int64
 12  JobRole                   1470 non-null   object
 13  JobSatisfaction           1470 non-null   int64
 14  MaritalStatus             1470 non-null   object
 15  MonthlyIncome             1470 non-null   int64
 16  MonthlyRate               1470 non-null   int64
 17  NumCompaniesWorked        1470 non-null   int64
 18  OverTime                  1470 non-null   object
 19  PercentSalaryHike         1470 non-null   int64
 20  PerformanceRating         1470 non-null   int64
 21  RelationshipSatisfaction  1470 non-null   int64
 22  StockOptionLevel          1470 non-null   int64
 23  TotalWorkingYears         1470 non-null   int64
 24  TrainingTimesLastYear     1470 non-null   int64
 25  WorkLifeBalance           1470 non-null   int64
 26  YearsAtCompany            1470 non-null   int64
 27  YearsInCurrentRole        1470 non-null   int64
 28  YearsSinceLastPromotion   1470 non-null   int64
 29  YearsWithCurrManager      1470 non-null   int64
dtypes: int64(23), object(7)
memory usage: 344.7+ KB
(1470, 30) None
```

```
<class 'pandas.core.frame.DataFrame'>
```

[210]:

```
y=df["Attrition"]
y.head()
```

[210]:

```
0    1
1    0
2    1
3    0
4    0
Name: Attrition, dtype: int32
```

5 Lable Encoding

[211]:

```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
```

[212]:

```
X["BusinessTravel"]=le.fit_transform(X["BusinessTravel"])
print(le.classes_)
m1=dict(zip(le.classes_,range(len(le.classes_))))
print(m1)
X["Department"]=le.fit_transform(X["Department"])
print(le.classes_)
m2=dict(zip(le.classes_,range(len(le.classes_))))
print(m2)
X["EducationField"]=le.fit_transform(X["EducationField"])
print(le.classes_)
m3=dict(zip(le.classes_,range(len(le.classes_))))
print(m3)
X["Gender"]=le.fit_transform(X["Gender"])
print(le.classes_)
m4=dict(zip(le.classes_,range(len(le.classes_))))
print(m4)
X["JobRole"]=le.fit_transform(X["JobRole"])
print(le.classes_)
m5=dict(zip(le.classes_,range(len(le.classes_))))
print(m5)
X["MaritalStatus"]=le.fit_transform(X["MaritalStatus"])
print(le.classes_)
m6=dict(zip(le.classes_,range(len(le.classes_))))
print(m6)
X["OverTime"]=le.fit_transform(X["OverTime"])
print(le.classes_)
m6=dict(zip(le.classes_,range(len(le.classes_))))
print(m6)
```

```
['Non-Travel' 'Travel_Frequently' 'Travel_Rarely']
```

```
{'Non-Travel': 0, 'Travel_Frequently': 1, 'Travel_Rarely': 2}
['Human Resources' 'Research & Development' 'Sales']
{'Human Resources': 0, 'Research & Development': 1, 'Sales': 2}
['Human Resources' 'Life Sciences' 'Marketing' 'Medical' 'Other'
 'Technical Degree']
{'Human Resources': 0, 'Life Sciences': 1, 'Marketing': 2, 'Medical': 3,
'Other': 4, 'Technical Degree': 5}
['Female' 'Male']
{'Female': 0, 'Male': 1}
['Healthcare Representative' 'Human Resources' 'Laboratory Technician'
 'Manager' 'Manufacturing Director' 'Research Director'
 'Research Scientist' 'Sales Executive' 'Sales Representative']
{'Healthcare Representative': 0, 'Human Resources': 1, 'Laboratory Technician':
2, 'Manager': 3, 'Manufacturing Director': 4, 'Research Director': 5, 'Research
Scientist': 6, 'Sales Executive': 7, 'Sales Representative': 8}
['Divorced' 'Married' 'Single']
{'Divorced': 0, 'Married': 1, 'Single': 2}
['No' 'Yes']
{'No': 0, 'Yes': 1}
```

[213]:

```
X.head()
```

[213]:

```
   Age  BusinessTravel  DailyRate  Department  DistanceFromHome  Education  \
0   41               2       1102           2                 1          2
1   49               1        279           1                 8          1
2   37               2       1373           1                 2          2
3   33               1       1392           1                 3          4
4   27               2        591           1                 2          1

   EducationField  EnvironmentSatisfaction  Gender  HourlyRate  …  \
0               1                        2       0          94  …
1               1                        3       1          61  …
2               4                        4       1          92  …
3               1                        4       0          56  …
4               3                        1       1          40  …

   PerformanceRating  RelationshipSatisfaction  StockOptionLevel  \
0                  3                         1                 0
1                  4                         4                 1
2                  3                         2                 0
3                  3                         3                 0
4                  3                         4                 1

   TotalWorkingYears  TrainingTimesLastYear  WorkLifeBalance  YearsAtCompany  \
0                  8                      0                1               6
1                 10                      3                3              10
2                  7                      3                3               0
```

```
3                  8                      3                 3               8
4                  6                      3                 3               2

   YearsInCurrentRole  YearsSinceLastPromotion  YearsWithCurrManager
0                   4                        0                     5
1                   7                        1                     7
2                   0                        0                     0
3                   7                        3                     0
4                   2                        2                     2

[5 rows x 30 columns]
```

[214]:

```
y.head()
#all categories are encoded
```

[214]:

```
0    1
1    0
2    1
3    0
4    0
Name: Attrition, dtype: int32
```

6 Feature Scaling

[215]:

```
#min max scaling from  0 to 1
from sklearn.preprocessing import MinMaxScaler
ms=MinMaxScaler()
```

[216]:

```
X_Scaled=ms.fit_transform(X)
X_Scaled=pd.DataFrame(ms.fit_transform(X),columns=X.columns)
X_Scaled.head()
```

[216]:

```
        Age  BusinessTravel  DailyRate  Department  DistanceFromHome  \
0  0.547619             1.0   0.715820         1.0          0.000000
1  0.738095             0.5   0.126700         0.5          0.250000
2  0.452381             1.0   0.909807         0.5          0.035714
3  0.357143             0.5   0.923407         0.5          0.071429
4  0.214286             1.0   0.350036         0.5          0.035714

   Education  EducationField  EnvironmentSatisfaction  Gender  HourlyRate  \
0       0.25             0.2                 0.333333     0.0    0.914286
1       0.00             0.2                 0.666667     1.0    0.442857
2       0.25             0.8                 1.000000     1.0    0.885714
3       0.75             0.2                 1.000000     0.0    0.371429
4       0.00             0.6                 0.000000     1.0    0.142857
```

```
   …  PerformanceRating  RelationshipSatisfaction  StockOptionLevel  \
0  …                0.0                  0.000000          0.000000
1  …                1.0                  1.000000          0.333333
2  …                0.0                  0.333333          0.000000
3  …                0.0                  0.666667          0.000000
4  …                0.0                  1.000000          0.333333

   TotalWorkingYears  TrainingTimesLastYear  WorkLifeBalance  YearsAtCompany  \
0              0.200                    0.0         0.000000            0.15
1              0.250                    0.5         0.666667            0.25
2              0.175                    0.5         0.666667            0.00
3              0.200                    0.5         0.666667            0.20
4              0.150                    0.5         0.666667            0.05

   YearsInCurrentRole  YearsSinceLastPromotion  YearsWithCurrManager
0            0.222222                 0.000000              0.294118
1            0.388889                 0.066667              0.411765
2            0.000000                 0.000000              0.000000
3            0.388889                 0.200000              0.000000
4            0.111111                 0.133333              0.117647

[5 rows x 30 columns]
```

7 Splitting DataSet into Train and Test¶

[217]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(X_Scaled,y,test_size =0.
 ↪2,random_state =0)
```

[218]:
```
print(x_train.shape,x_test.shape,y_train.shape,y_test.shape)
```

```
(1176, 30) (294, 30) (1176,) (294,)
```

[219]:
```
x_train.head()
```

[219]:
```
           Age  BusinessTravel  DailyRate  Department  DistanceFromHome  \
1374  0.952381             1.0   0.360057         1.0          0.714286
1092  0.642857             1.0   0.607015         0.5          0.964286
768   0.523810             1.0   0.141732         1.0          0.892857
569   0.428571             0.0   0.953472         1.0          0.250000
911   0.166667             0.5   0.355762         1.0          0.821429

      Education  EducationField  EnvironmentSatisfaction  Gender  HourlyRate  \
1374       0.50             0.2                 1.000000     0.0    0.600000
1092       0.50             1.0                 1.000000     1.0    0.957143
768        0.50             0.4                 0.666667     1.0    0.628571
```

```
569        0.75             0.2                 0.000000     1.0    0.657143
911        0.00             0.2                 0.666667     1.0    0.614286

      …  PerformanceRating  RelationshipSatisfaction  StockOptionLevel  \
1374  …                0.0                  0.666667          0.333333
1092  …                1.0                  1.000000          0.333333
768   …                0.0                  0.333333          0.333333
569   …                0.0                  0.333333          0.000000
911   …                0.0                  1.000000          0.000000

      TotalWorkingYears  TrainingTimesLastYear  WorkLifeBalance  \
1374              0.725               0.333333         0.333333
1092              0.200               0.500000         0.666667
768               0.200               0.500000         0.333333
569               0.250               0.166667         0.666667
911               0.025               0.666667         0.666667

      YearsAtCompany  YearsInCurrentRole  YearsSinceLastPromotion  \
1374           0.025            0.000000                 0.000000
1092           0.125            0.222222                 0.000000
768            0.175            0.388889                 0.466667
569            0.250            0.388889                 0.000000
911            0.025            0.000000                 0.066667

      YearsWithCurrManager
1374              0.000000
1092              0.176471
768               0.294118
569               0.529412
911               0.000000

[5 rows x 30 columns]
```

8 LogisticRegression Model Building

[220]:
```
from sklearn.linear_model import LogisticRegression
model=LogisticRegression()
```

[221]:
```
model.fit(x_train,y_train)
```

[221]:
```
LogisticRegression()
```

[222]:
```
pred=model.predict(x_test)
```

[223]:
```
pred
```

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

```
[224]: y_test.head()
```

```
[224]: 442     0
       1091    0
       981     1
       785     0
       1332    1
       Name: Attrition, dtype: int32
```

9 Evaluation of classification model

```
[225]: #Accuracy score
       from sklearn.metrics import␣
        ↪accuracy_score,confusion_matrix,classification_report,roc_auc_score,roc_curve
```

```
[226]: accuracy_score(y_test,pred)
```

```
[226]: 0.8809523809523809
```

```
[227]: confusion_matrix(y_test,pred)
```

```
[227]: array([[242,   3],
              [ 32,  17]], dtype=int64)
```

```
[228]: pd.crosstab(y_test,pred)
```

```
[228]: col_0        0   1
       Attrition
       0          242   3
       1           32  17
```

[229]:

```
print(classification_report(y_test,pred))
```

```
              precision    recall  f1-score   support

           0       0.88      0.99      0.93       245
           1       0.85      0.35      0.49        49

    accuracy                           0.88       294
   macro avg       0.87      0.67      0.71       294
weighted avg       0.88      0.88      0.86       294
```

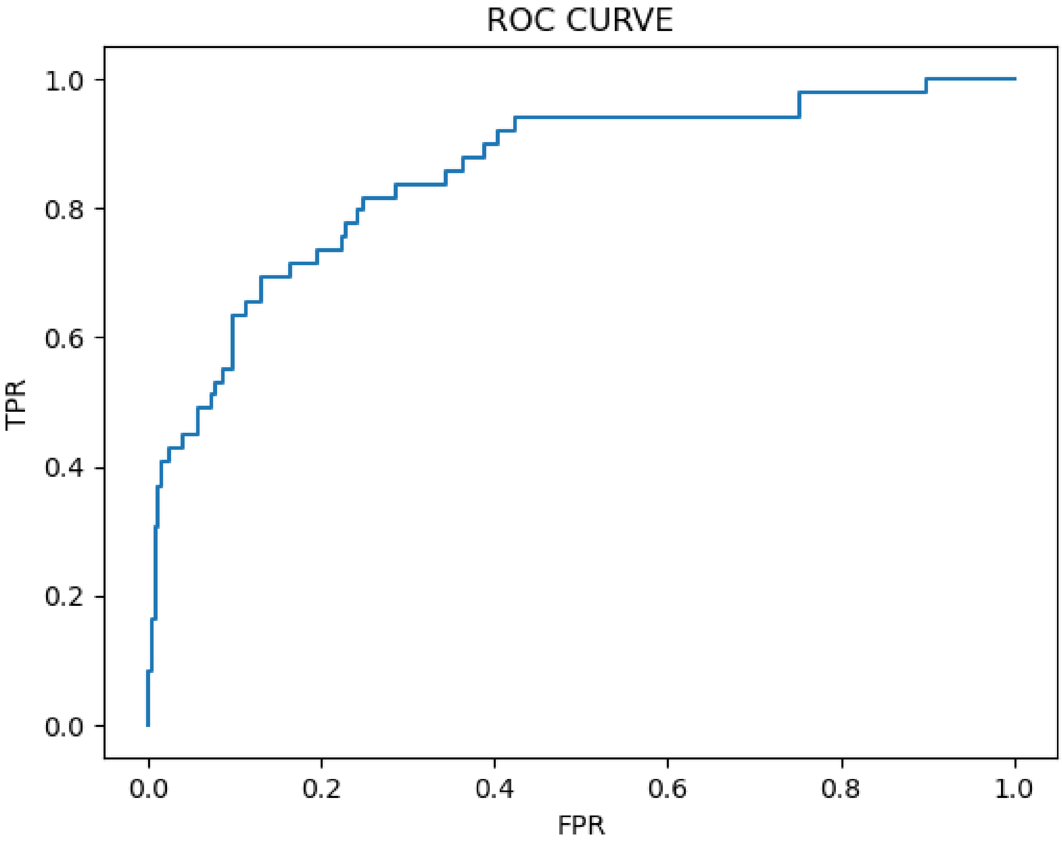
10 ROC-AUC curve

[230]:

```
probability=model.predict_proba(x_test)[:,1]
```

[231]:

```
# roc_curve
fpr,tpr,threshsholds = roc_curve(y_test,probability)
plt.plot(fpr,tpr)
plt.xlabel('FPR')
plt.ylabel('TPR')
plt.title('ROC CURVE')
plt.show()
```



11 Decision Tree Model Building

```
[232]: from sklearn.tree import DecisionTreeClassifier
       dtc=DecisionTreeClassifier()
```

```
[233]: dtc.fit(x_train,y_train)
```

```
[233]: DecisionTreeClassifier()
```

```
[234]: pred=dtc.predict(x_test)
       pred
```

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

```
[235]: #Accuracy score
       from sklearn.metrics import␣
        ↪accuracy_score,confusion_matrix,classification_report,roc_auc_score,roc_curve
       accuracy_score(y_test,pred)
```

```
[235]: 0.7517006802721088
```

```
[236]: confusion_matrix(y_test,pred)
```

```
[236]: array([[204,  41],
              [ 32,  17]], dtype=int64)
```

```
[237]: pd.crosstab(y_test,pred)
```

```
[237]: col_0        0   1
       Attrition
       0          204  41
       1           32  17
```

```
[238]: print(classification_report(y_test,pred))
```

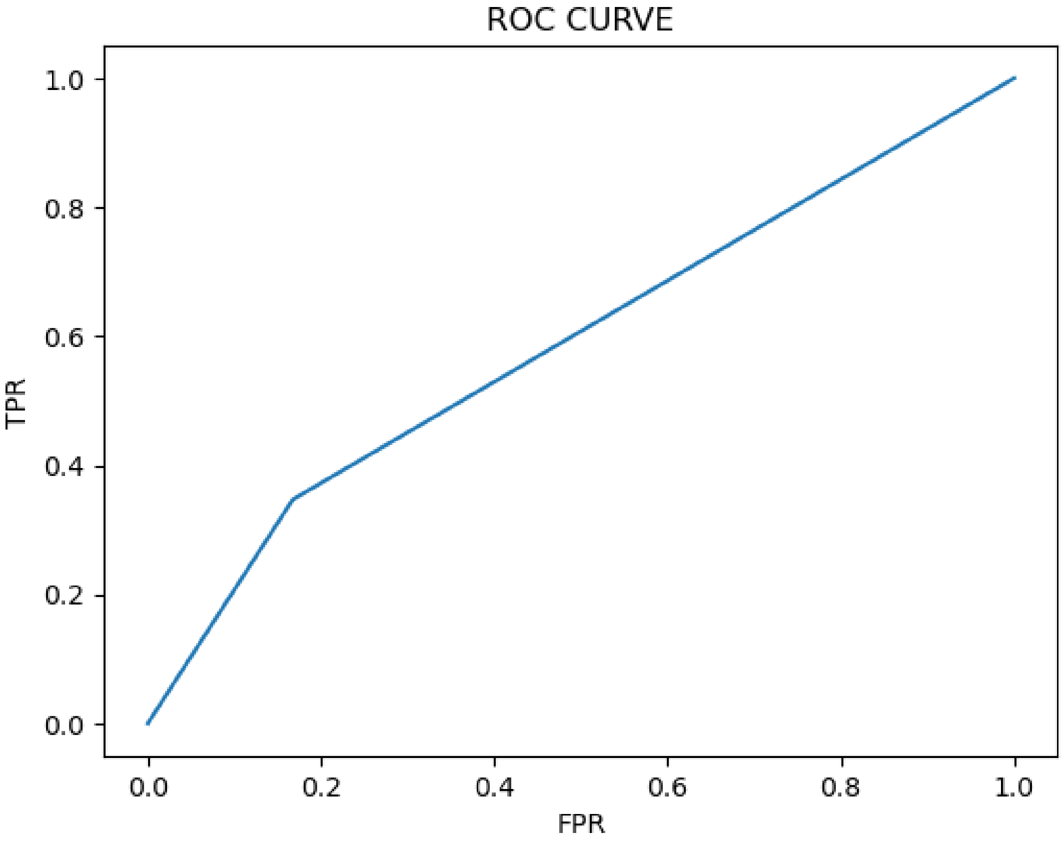
```
                  precision    recall  f1-score   support

               0       0.86      0.83      0.85       245
               1       0.29      0.35      0.32        49

        accuracy                           0.75       294
       macro avg       0.58      0.59      0.58       294
    weighted avg       0.77      0.75      0.76       294
```

```
[239]: probability=dtc.predict_proba(x_test)[:,1]
       probability
```

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

[240]:

```
# roc_curve
fpr,tpr,threshsholds = roc_curve(y_test,probability)
plt.plot(fpr,tpr)
plt.xlabel('FPR')
plt.ylabel('TPR')
plt.title('ROC CURVE')
plt.show()
```



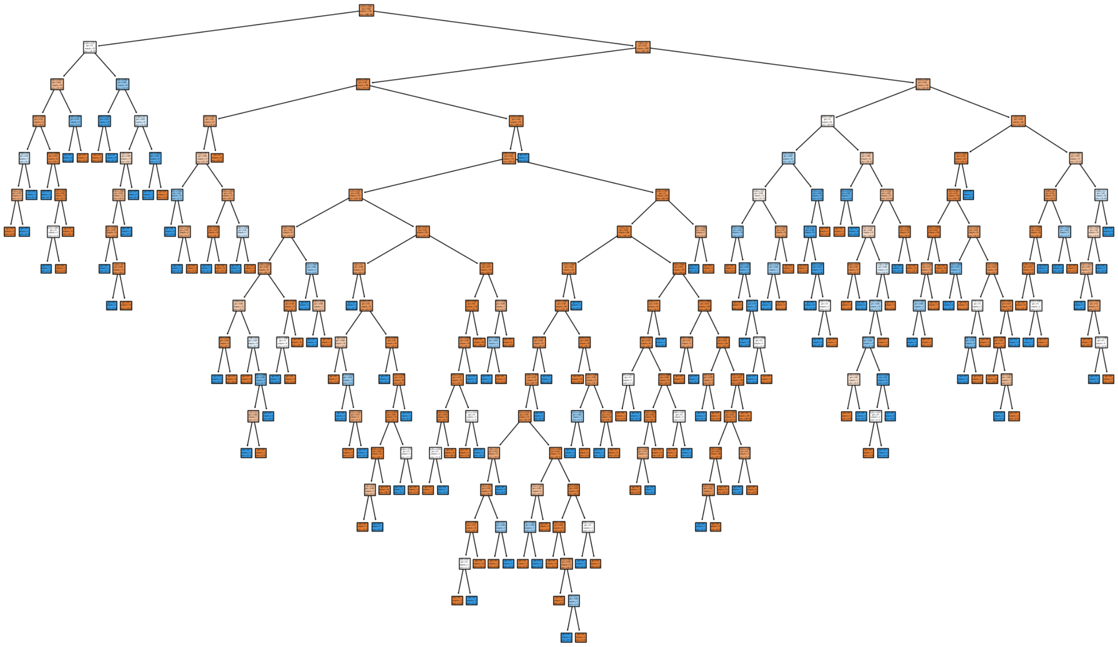
12 Hyperparameter tuning

[241]:

```
from sklearn import tree
plt.figure(figsize=(25,15))
tree.plot_tree(dtc,filled=True)
```

[241]:

```
[Text(0.32688674428104575, 0.9722222222222222, 'x[23] <= 0.038\ngini =
0.269\nsamples = 1176\nvalue = [988, 188]'),
 Text(0.0784313725490196, 0.9166666666666666, 'x[14] <= 0.75\ngini =
0.5\nsamples = 78\nvalue = [39, 39]'),
 Text(0.049019607843137254, 0.8611111111111112, 'x[4] <= 0.554\ngini =
0.426\nsamples = 39\nvalue = [27, 12]'),
 Text(0.032679738562091505, 0.8055555555555556, 'x[13] <= 0.167\ngini =
0.312\nsamples = 31\nvalue = [25, 6]'),
 Text(0.0196078431372549, 0.75, 'x[18] <= 0.5\ngini = 0.49\nsamples = 7\nvalue =
[3, 4]'),
 Text(0.013071895424836602, 0.6944444444444444, 'x[14] <= 0.25\ngini =
0.375\nsamples = 4\nvalue = [3, 1]'),
 Text(0.006535947712418301, 0.6388888888888888, 'gini = 0.0\nsamples = 3\nvalue
= [3, 0]'),
 Text(0.0196078431372549, 0.6388888888888888, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.026143790849673203, 0.6944444444444444, 'gini = 0.0\nsamples = 3\nvalue
= [0, 3]'),
 Text(0.0457516339869281, 0.75, 'x[17] <= 0.056\ngini = 0.153\nsamples =
24\nvalue = [22, 2]'),
 Text(0.0392156862745098, 0.6944444444444444, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.05228758169934641, 0.6944444444444444, 'x[7] <= 0.167\ngini =
0.083\nsamples = 23\nvalue = [22, 1]'),
 Text(0.0457516339869281, 0.6388888888888888, 'x[13] <= 0.667\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.0392156862745098, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.05228758169934641, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.058823529411764705, 0.6388888888888888, 'gini = 0.0\nsamples = 21\nvalue
= [21, 0]'),
 Text(0.06535947712418301, 0.8055555555555556, 'x[19] <= 0.679\ngini =
0.375\nsamples = 8\nvalue = [2, 6]'),
 Text(0.058823529411764705, 0.75, 'gini = 0.0\nsamples = 6\nvalue = [0, 6]'),
 Text(0.0718954248366013, 0.75, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
 Text(0.10784313725490197, 0.8611111111111112, 'x[9] <= 0.364\ngini =
0.426\nsamples = 39\nvalue = [12, 27]'),
 Text(0.0915032679738562, 0.8055555555555556, 'x[25] <= 0.167\ngini =
0.133\nsamples = 14\nvalue = [1, 13]'),
 Text(0.08496732026143791, 0.75, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.09803921568627451, 0.75, 'gini = 0.0\nsamples = 13\nvalue = [0, 13]'),
 Text(0.12418300653594772, 0.8055555555555556, 'x[18] <= 0.5\ngini =
0.493\nsamples = 25\nvalue = [11, 14]'),
 Text(0.1111111111111111, 0.75, 'x[6] <= 0.7\ngini = 0.484\nsamples = 17\nvalue
= [10, 7]'),
 Text(0.10457516339869281, 0.6944444444444444, 'x[16] <= 0.892\ngini =
0.408\nsamples = 14\nvalue = [10, 4]'),
 Text(0.09803921568627451, 0.6388888888888888, 'x[2] <= 0.106\ngini =
0.278\nsamples = 12\nvalue = [10, 2]'),
 Text(0.0915032679738562, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.10457516339869281, 0.5833333333333334, 'x[25] <= 0.5\ngini =
0.165\nsamples = 11\nvalue = [10, 1]'),
 Text(0.09803921568627451, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.1111111111111111, 0.5277777777777778, 'gini = 0.0\nsamples = 10\nvalue =
[10, 0]'),
 Text(0.1111111111111111, 0.6388888888888888, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
 Text(0.11764705882352941, 0.6944444444444444, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
 Text(0.13725490196078433, 0.75, 'x[4] <= 0.839\ngini = 0.219\nsamples =
8\nvalue = [1, 7]'),
 Text(0.13071895424836602, 0.6944444444444444, 'gini = 0.0\nsamples = 7\nvalue =
[0, 7]'),
 Text(0.1437908496732026, 0.6944444444444444, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.575342116013072, 0.9166666666666666, 'x[18] <= 0.5\ngini =
0.235\nsamples = 1098\nvalue = [949, 149]'),
 Text(0.32388684640522875, 0.8611111111111112, 'x[25] <= 0.167\ngini =
0.162\nsamples = 798\nvalue = [727, 71]'),
 Text(0.18627450980392157, 0.8055555555555556, 'x[2] <= 0.747\ngini =
0.38\nsamples = 47\nvalue = [35, 12]'),
 Text(0.17973856209150327, 0.75, 'x[10] <= 0.5\ngini = 0.463\nsamples =
33\nvalue = [21, 12]'),
 Text(0.1568627450980392, 0.6944444444444444, 'x[4] <= 0.446\ngini =
0.42\nsamples = 10\nvalue = [3, 7]'),
 Text(0.1503267973856209, 0.6388888888888888, 'gini = 0.0\nsamples = 6\nvalue =
[0, 6]'),
 Text(0.16339869281045752, 0.6388888888888888, 'x[13] <= 0.167\ngini =
0.375\nsamples = 4\nvalue = [3, 1]'),
 Text(0.1568627450980392, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.16993464052287582, 0.5833333333333334, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
 Text(0.20261437908496732, 0.6944444444444444, 'x[24] <= 0.583\ngini =
0.34\nsamples = 23\nvalue = [18, 5]'),
 Text(0.1895424836601307, 0.6388888888888888, 'x[9] <= 0.107\ngini =
0.117\nsamples = 16\nvalue = [15, 1]'),
 Text(0.1830065359477124, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.19607843137254902, 0.5833333333333334, 'gini = 0.0\nsamples = 15\nvalue
= [15, 0]'),
 Text(0.21568627450980393, 0.6388888888888888, 'x[26] <= 0.287\ngini =
0.49\nsamples = 7\nvalue = [3, 4]'),
 Text(0.20915032679738563, 0.5833333333333334, 'gini = 0.0\nsamples = 4\nvalue =
[0, 4]'),
 Text(0.2222222222222222, 0.5833333333333334, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
 Text(0.19281045751633988, 0.75, 'gini = 0.0\nsamples = 14\nvalue = [14, 0]'),
 Text(0.461499183006536, 0.8055555555555556, 'x[23] <= 0.975\ngini =
0.145\nsamples = 751\nvalue = [692, 59]'),
 Text(0.45496323529411764, 0.75, 'x[26] <= 0.113\ngini = 0.143\nsamples =
750\nvalue = [692, 58]'),
 Text(0.31699346405228757, 0.6944444444444444, 'x[7] <= 0.167\ngini =
0.218\nsamples = 257\nvalue = [225, 32]'),
 Text(0.2565359477124183, 0.6388888888888888, 'x[29] <= 0.147\ngini =
0.355\nsamples = 65\nvalue = [50, 15]'),
 Text(0.23529411764705882, 0.5833333333333334, 'x[29] <= 0.029\ngini =
0.303\nsamples = 59\nvalue = [48, 11]'),
 Text(0.21241830065359477, 0.5277777777777778, 'x[10] <= 0.5\ngini =
0.463\nsamples = 22\nvalue = [14, 8]'),
 Text(0.19934640522875818, 0.4722222222222222, 'x[9] <= 0.179\ngini =
0.198\nsamples = 9\nvalue = [8, 1]'),
 Text(0.19281045751633988, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.20588235294117646, 0.4166666666666667, 'gini = 0.0\nsamples = 8\nvalue =
[8, 0]'),
 Text(0.22549019607843138, 0.4722222222222222, 'x[9] <= 0.4\ngini =
0.497\nsamples = 13\nvalue = [6, 7]'),
 Text(0.21895424836601307, 0.4166666666666667, 'gini = 0.0\nsamples = 4\nvalue =
[4, 0]'),
 Text(0.23202614379084968, 0.4166666666666667, 'x[4] <= 0.286\ngini =
0.346\nsamples = 9\nvalue = [2, 7]'),
 Text(0.22549019607843138, 0.3611111111111111, 'x[5] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.21895424836601307, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.23202614379084968, 0.3055555555555556, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
 Text(0.238562091503268, 0.3611111111111111, 'gini = 0.0\nsamples = 6\nvalue =
[0, 6]'),
 Text(0.2581699346405229, 0.5277777777777778, 'x[13] <= 0.167\ngini =
0.149\nsamples = 37\nvalue = [34, 3]'),
 Text(0.25163398692810457, 0.4722222222222222, 'x[26] <= 0.088\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.24509803921568626, 0.4166666666666667, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
 Text(0.2581699346405229, 0.4166666666666667, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
 Text(0.2647058823529412, 0.4722222222222222, 'gini = 0.0\nsamples = 31\nvalue =
[31, 0]'),
 Text(0.2777777777777778, 0.5833333333333334, 'x[21] <= 0.667\ngini =
0.444\nsamples = 6\nvalue = [2, 4]'),
 Text(0.27124183006535946, 0.5277777777777778, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
 Text(0.28431372549019607, 0.5277777777777778, 'x[28] <= 0.033\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.2777777777777778, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.2908496732026144, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
 Text(0.37745098039215685, 0.6388888888888888, 'x[0] <= 0.321\ngini =
0.161\nsamples = 192\nvalue = [175, 17]'),
 Text(0.3202614379084967, 0.5833333333333334, 'x[6] <= 0.1\ngini =
0.294\nsamples = 67\nvalue = [55, 12]'),
 Text(0.3137254901960784, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
 Text(0.32679738562091504, 0.5277777777777778, 'x[25] <= 0.5\ngini =
0.26\nsamples = 65\nvalue = [55, 10]'),
 Text(0.30392156862745096, 0.4722222222222222, 'x[6] <= 0.5\ngini =
0.469\nsamples = 16\nvalue = [10, 6]'),
 Text(0.2973856209150327, 0.4166666666666667, 'gini = 0.0\nsamples = 7\nvalue =
[7, 0]'),
 Text(0.3104575163398693, 0.4166666666666667, 'x[7] <= 0.833\ngini =
0.444\nsamples = 9\nvalue = [3, 6]'),
 Text(0.30392156862745096, 0.3611111111111111, 'gini = 0.0\nsamples = 5\nvalue =
[0, 5]'),
 Text(0.31699346405228757, 0.3611111111111111, 'x[11] <= 0.125\ngini =
0.375\nsamples = 4\nvalue = [3, 1]'),
 Text(0.3104575163398693, 0.3055555555555556, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
 Text(0.3235294117647059, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.34967320261437906, 0.4722222222222222, 'x[2] <= 0.037\ngini =
0.15\nsamples = 49\nvalue = [45, 4]'),
 Text(0.3431372549019608, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.3562091503267974, 0.4166666666666667, 'x[2] <= 0.938\ngini =
0.117\nsamples = 48\nvalue = [45, 3]'),
 Text(0.34967320261437906, 0.3611111111111111, 'x[5] <= 0.875\ngini =
0.081\nsamples = 47\nvalue = [45, 2]'),
 Text(0.3366013071895425, 0.3055555555555556, 'x[10] <= 0.167\ngini =
0.043\nsamples = 45\nvalue = [44, 1]'),
 Text(0.3300653594771242, 0.25, 'x[3] <= 0.75\ngini = 0.444\nsamples = 3\nvalue
= [2, 1]'),
 Text(0.3235294117647059, 0.19444444444444445, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
 Text(0.3366013071895425, 0.19444444444444445, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.3431372549019608, 0.25, 'gini = 0.0\nsamples = 42\nvalue = [42, 0]'),
 Text(0.3627450980392157, 0.3055555555555556, 'x[14] <= 0.75\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.3562091503267974, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.369281045751634, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.3627450980392157, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.434640522875817, 0.5833333333333334, 'x[6] <= 0.9\ngini = 0.077\nsamples
= 125\nvalue = [120, 5]'),
 Text(0.4215686274509804, 0.5277777777777778, 'x[0] <= 0.393\ngini =
0.05\nsamples = 118\nvalue = [115, 3]'),
 Text(0.4150326797385621, 0.4722222222222222, 'x[2] <= 0.956\ngini =
0.185\nsamples = 29\nvalue = [26, 3]'),
 Text(0.4084967320261438, 0.4166666666666667, 'x[29] <= 0.147\ngini =
0.133\nsamples = 28\nvalue = [26, 2]'),
 Text(0.3954248366013072, 0.3611111111111111, 'x[10] <= 0.167\ngini =
0.074\nsamples = 26\nvalue = [25, 1]'),
 Text(0.3888888888888889, 0.3055555555555556, 'x[19] <= 0.214\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.38235294117647056, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.3954248366013072, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.4019607843137255, 0.3055555555555556, 'gini = 0.0\nsamples = 24\nvalue =
[24, 0]'),
 Text(0.4215686274509804, 0.3611111111111111, 'x[17] <= 0.111\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.4150326797385621, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.42810457516339867, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.4215686274509804, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.42810457516339867, 0.4722222222222222, 'gini = 0.0\nsamples = 89\nvalue
= [89, 0]'),
 Text(0.4477124183006536, 0.5277777777777778, 'x[2] <= 0.594\ngini =
0.408\nsamples = 7\nvalue = [5, 2]'),
 Text(0.4411764705882353, 0.4722222222222222, 'x[27] <= 0.056\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.434640522875817, 0.4166666666666667, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
 Text(0.4477124183006536, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.4542483660130719, 0.4722222222222222, 'gini = 0.0\nsamples = 4\nvalue =
[4, 0]'),
 Text(0.5929330065359477, 0.6944444444444444, 'x[26] <= 0.787\ngini =
0.1\nsamples = 493\nvalue = [467, 26]'),
 Text(0.5584150326797386, 0.6388888888888888, 'x[13] <= 0.5\ngini =
0.094\nsamples = 486\nvalue = [462, 24]'),
 Text(0.5089869281045751, 0.5833333333333334, 'x[12] <= 0.938\ngini =
0.154\nsamples = 191\nvalue = [175, 16]'),
 Text(0.5024509803921569, 0.5277777777777778, 'x[16] <= 0.481\ngini =
0.145\nsamples = 190\nvalue = [175, 15]'),
 Text(0.4820261437908497, 0.4722222222222222, 'x[16] <= 0.47\ngini =
0.221\nsamples = 95\nvalue = [83, 12]'),
 Text(0.47549019607843135, 0.4166666666666667, 'x[29] <= 0.794\ngini =
0.207\nsamples = 94\nvalue = [83, 11]'),
 Text(0.46895424836601307, 0.3611111111111111, 'x[5] <= 0.375\ngini =
0.192\nsamples = 93\nvalue = [83, 10]'),
 Text(0.4411764705882353, 0.3055555555555556, 'x[6] <= 0.9\ngini =
0.363\nsamples = 21\nvalue = [16, 5]'),
 Text(0.434640522875817, 0.25, 'x[15] <= 0.413\ngini = 0.266\nsamples =
19\nvalue = [16, 3]'),
 Text(0.4215686274509804, 0.19444444444444445, 'x[17] <= 0.056\ngini =
0.117\nsamples = 16\nvalue = [15, 1]'),
 Text(0.4150326797385621, 0.1388888888888889, 'x[6] <= 0.4\ngini = 0.5\nsamples
= 2\nvalue = [1, 1]'),
 Text(0.4084967320261438, 0.08333333333333333, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.4215686274509804, 0.08333333333333333, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.42810457516339867, 0.1388888888888889, 'gini = 0.0\nsamples = 14\nvalue
= [14, 0]'),
 Text(0.4477124183006536, 0.19444444444444445, 'x[6] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.4411764705882353, 0.1388888888888889, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.4542483660130719, 0.1388888888888889, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
 Text(0.4477124183006536, 0.25, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
 Text(0.49673202614379086, 0.3055555555555556, 'x[27] <= 0.139\ngini =
0.129\nsamples = 72\nvalue = [67, 5]'),
 Text(0.4803921568627451, 0.25, 'x[15] <= 0.244\ngini = 0.444\nsamples =
6\nvalue = [4, 2]'),
 Text(0.4738562091503268, 0.19444444444444445, 'x[9] <= 0.686\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.4673202614379085, 0.1388888888888889, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.4803921568627451, 0.1388888888888889, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
 Text(0.4869281045751634, 0.19444444444444445, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
 Text(0.5130718954248366, 0.25, 'x[2] <= 0.958\ngini = 0.087\nsamples =
66\nvalue = [63, 3]'),
 Text(0.5, 0.19444444444444445, 'x[24] <= 0.583\ngini = 0.061\nsamples =
64\nvalue = [62, 2]'),
 Text(0.4934640522875817, 0.1388888888888889, 'gini = 0.0\nsamples = 52\nvalue =
[52, 0]'),
 Text(0.5065359477124183, 0.1388888888888889, 'x[12] <= 0.812\ngini =
0.278\nsamples = 12\nvalue = [10, 2]'),
 Text(0.5, 0.08333333333333333, 'gini = 0.0\nsamples = 9\nvalue = [9, 0]'),
 Text(0.5130718954248366, 0.08333333333333333, 'x[7] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.5065359477124183, 0.027777777777777776, 'gini = 0.0\nsamples = 2\nvalue
= [0, 2]'),
 Text(0.5196078431372549, 0.027777777777777776, 'gini = 0.0\nsamples = 1\nvalue
= [1, 0]'),
 Text(0.5261437908496732, 0.19444444444444445, 'x[28] <= 0.567\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.5196078431372549, 0.1388888888888889, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.5326797385620915, 0.1388888888888889, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.4820261437908497, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.48856209150326796, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.5228758169934641, 0.4722222222222222, 'x[17] <= 0.5\ngini =
0.061\nsamples = 95\nvalue = [92, 3]'),
 Text(0.5163398692810458, 0.4166666666666667, 'gini = 0.0\nsamples = 76\nvalue =
[76, 0]'),
 Text(0.5294117647058824, 0.4166666666666667, 'x[29] <= 0.088\ngini =
0.266\nsamples = 19\nvalue = [16, 3]'),
 Text(0.5163398692810458, 0.3611111111111111, 'x[21] <= 0.833\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.5098039215686274, 0.3055555555555556, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
 Text(0.5228758169934641, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.5424836601307189, 0.3611111111111111, 'x[15] <= 0.108\ngini =
0.117\nsamples = 16\nvalue = [15, 1]'),
 Text(0.5359477124183006, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.5490196078431373, 0.3055555555555556, 'gini = 0.0\nsamples = 15\nvalue =
[15, 0]'),
 Text(0.5155228758169934, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.6078431372549019, 0.5833333333333334, 'x[19] <= 0.036\ngini =
0.053\nsamples = 295\nvalue = [287, 8]'),
 Text(0.5849673202614379, 0.5277777777777778, 'x[28] <= 0.7\ngini =
0.159\nsamples = 46\nvalue = [42, 4]'),
 Text(0.5784313725490197, 0.4722222222222222, 'x[10] <= 0.167\ngini =
0.124\nsamples = 45\nvalue = [42, 3]'),
 Text(0.5620915032679739, 0.4166666666666667, 'x[5] <= 0.625\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.5555555555555556, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.5686274509803921, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.5947712418300654, 0.4166666666666667, 'x[23] <= 0.688\ngini =
0.089\nsamples = 43\nvalue = [41, 2]'),
 Text(0.5816993464052288, 0.3611111111111111, 'x[12] <= 0.062\ngini =
0.048\nsamples = 41\nvalue = [40, 1]'),
 Text(0.5751633986928104, 0.3055555555555556, 'x[0] <= 0.595\ngini =
0.375\nsamples = 4\nvalue = [3, 1]'),
 Text(0.5686274509803921, 0.25, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
 Text(0.5816993464052288, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.5882352941176471, 0.3055555555555556, 'gini = 0.0\nsamples = 37\nvalue =
[37, 0]'),
 Text(0.6078431372549019, 0.3611111111111111, 'x[28] <= 0.167\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.6013071895424836, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.6143790849673203, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.5915032679738562, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.630718954248366, 0.5277777777777778, 'x[15] <= 0.056\ngini =
0.032\nsamples = 249\nvalue = [245, 4]'),
 Text(0.6143790849673203, 0.4722222222222222, 'x[15] <= 0.054\ngini =
0.32\nsamples = 5\nvalue = [4, 1]'),
 Text(0.6078431372549019, 0.4166666666666667, 'gini = 0.0\nsamples = 4\nvalue =
[4, 0]'),
 Text(0.6209150326797386, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.6470588235294118, 0.4722222222222222, 'x[2] <= 0.015\ngini =
0.024\nsamples = 244\nvalue = [241, 3]'),
 Text(0.6339869281045751, 0.4166666666666667, 'x[8] <= 0.5\ngini =
0.278\nsamples = 6\nvalue = [5, 1]'),
 Text(0.6274509803921569, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.6405228758169934, 0.3611111111111111, 'gini = 0.0\nsamples = 5\nvalue =
[5, 0]'),
 Text(0.6601307189542484, 0.4166666666666667, 'x[21] <= 0.167\ngini =
0.017\nsamples = 238\nvalue = [236, 2]'),
 Text(0.6535947712418301, 0.3611111111111111, 'x[25] <= 0.833\ngini =
0.073\nsamples = 53\nvalue = [51, 2]'),
 Text(0.6405228758169934, 0.3055555555555556, 'x[29] <= 0.088\ngini =
0.041\nsamples = 48\nvalue = [47, 1]'),
 Text(0.6339869281045751, 0.25, 'x[12] <= 0.312\ngini = 0.245\nsamples =
7\nvalue = [6, 1]'),
 Text(0.6274509803921569, 0.19444444444444445, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.6405228758169934, 0.19444444444444445, 'gini = 0.0\nsamples = 6\nvalue =
[6, 0]'),
 Text(0.6470588235294118, 0.25, 'gini = 0.0\nsamples = 41\nvalue = [41, 0]'),
 Text(0.6666666666666666, 0.3055555555555556, 'x[15] <= 0.38\ngini =
0.32\nsamples = 5\nvalue = [4, 1]'),
 Text(0.6601307189542484, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.673202614379085, 0.25, 'gini = 0.0\nsamples = 4\nvalue = [4, 0]'),
 Text(0.6666666666666666, 0.3611111111111111, 'gini = 0.0\nsamples = 185\nvalue
= [185, 0]'),
 Text(0.6274509803921569, 0.6388888888888888, 'x[2] <= 0.366\ngini =
0.408\nsamples = 7\nvalue = [5, 2]'),
 Text(0.6209150326797386, 0.5833333333333334, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
 Text(0.6339869281045751, 0.5833333333333334, 'gini = 0.0\nsamples = 5\nvalue =
[5, 0]'),
 Text(0.46803513071895425, 0.75, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.826797385620915, 0.8611111111111112, 'x[15] <= 0.157\ngini =
0.385\nsamples = 300\nvalue = [222, 78]'),
 Text(0.7410130718954249, 0.8055555555555556, 'x[22] <= 0.167\ngini =
0.5\nsamples = 96\nvalue = [49, 47]'),
 Text(0.7058823529411765, 0.75, 'x[4] <= 0.161\ngini = 0.459\nsamples =
42\nvalue = [15, 27]'),
 Text(0.6797385620915033, 0.6944444444444444, 'x[16] <= 0.41\ngini =
0.499\nsamples = 23\nvalue = [12, 11]'),
 Text(0.6601307189542484, 0.6388888888888888, 'x[15] <= 0.061\ngini =
0.426\nsamples = 13\nvalue = [4, 9]'),
 Text(0.6535947712418301, 0.5833333333333334, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
 Text(0.6666666666666666, 0.5833333333333334, 'x[24] <= 0.25\ngini =
0.298\nsamples = 11\nvalue = [2, 9]'),
 Text(0.6601307189542484, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.673202614379085, 0.5277777777777778, 'x[20] <= 0.5\ngini = 0.18\nsamples
= 10\nvalue = [1, 9]'),
 Text(0.6666666666666666, 0.4722222222222222, 'gini = 0.0\nsamples = 8\nvalue =
[0, 8]'),
 Text(0.6797385620915033, 0.4722222222222222, 'x[15] <= 0.083\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.673202614379085, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.6862745098039216, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.6993464052287581, 0.6388888888888888, 'x[7] <= 0.167\ngini =
0.32\nsamples = 10\nvalue = [8, 2]'),
 Text(0.6928104575163399, 0.5833333333333334, 'x[15] <= 0.137\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.6862745098039216, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
 Text(0.6993464052287581, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.7058823529411765, 0.5833333333333334, 'gini = 0.0\nsamples = 7\nvalue =
[7, 0]'),
 Text(0.7320261437908496, 0.6944444444444444, 'x[23] <= 0.35\ngini =
0.266\nsamples = 19\nvalue = [3, 16]'),
 Text(0.7254901960784313, 0.6388888888888888, 'x[9] <= 0.2\ngini =
0.198\nsamples = 18\nvalue = [2, 16]'),
 Text(0.7189542483660131, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.7320261437908496, 0.5833333333333334, 'x[27] <= 0.306\ngini =
0.111\nsamples = 17\nvalue = [1, 16]'),
 Text(0.7254901960784313, 0.5277777777777778, 'gini = 0.0\nsamples = 15\nvalue =
[0, 15]'),
 Text(0.738562091503268, 0.5277777777777778, 'x[26] <= 0.188\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.7320261437908496, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.7450980392156863, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.738562091503268, 0.6388888888888888, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.7761437908496732, 0.75, 'x[0] <= 0.202\ngini = 0.466\nsamples =
54\nvalue = [34, 20]'),
 Text(0.7581699346405228, 0.6944444444444444, 'x[0] <= 0.107\ngini =
0.245\nsamples = 7\nvalue = [1, 6]'),
 Text(0.7516339869281046, 0.6388888888888888, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.7647058823529411, 0.6388888888888888, 'gini = 0.0\nsamples = 6\nvalue =
[0, 6]'),
 Text(0.7941176470588235, 0.6944444444444444, 'x[2] <= 0.622\ngini =
0.418\nsamples = 47\nvalue = [33, 14]'),
 Text(0.7777777777777778, 0.6388888888888888, 'x[2] <= 0.145\ngini =
0.482\nsamples = 32\nvalue = [19, 13]'),
 Text(0.7647058823529411, 0.5833333333333334, 'x[26] <= 0.237\ngini =
0.18\nsamples = 10\nvalue = [9, 1]'),
 Text(0.7581699346405228, 0.5277777777777778, 'gini = 0.0\nsamples = 9\nvalue =
[9, 0]'),
 Text(0.7712418300653595, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.7908496732026143, 0.5833333333333334, 'x[16] <= 0.87\ngini =
0.496\nsamples = 22\nvalue = [10, 12]'),
 Text(0.7843137254901961, 0.5277777777777778, 'x[25] <= 0.833\ngini =
0.465\nsamples = 19\nvalue = [7, 12]'),
 Text(0.7777777777777778, 0.4722222222222222, 'x[17] <= 0.167\ngini =
0.415\nsamples = 17\nvalue = [5, 12]'),
 Text(0.7647058823529411, 0.4166666666666667, 'x[19] <= 0.321\ngini =
0.49\nsamples = 7\nvalue = [4, 3]'),
 Text(0.7581699346405228, 0.3611111111111111, 'gini = 0.0\nsamples = 4\nvalue =
[4, 0]'),
 Text(0.7712418300653595, 0.3611111111111111, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
 Text(0.7908496732026143, 0.4166666666666667, 'x[16] <= 0.235\ngini =
0.18\nsamples = 10\nvalue = [1, 9]'),
 Text(0.7843137254901961, 0.3611111111111111, 'x[24] <= 0.417\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.7777777777777778, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.7908496732026143, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.7973856209150327, 0.3611111111111111, 'gini = 0.0\nsamples = 8\nvalue =
[0, 8]'),
 Text(0.7908496732026143, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
 Text(0.7973856209150327, 0.5277777777777778, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
 Text(0.8104575163398693, 0.6388888888888888, 'x[9] <= 0.064\ngini =
0.124\nsamples = 15\nvalue = [14, 1]'),
 Text(0.803921568627451, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.8169934640522876, 0.5833333333333334, 'gini = 0.0\nsamples = 14\nvalue =
[14, 0]'),
 Text(0.9125816993464052, 0.8055555555555556, 'x[14] <= 0.75\ngini =
0.258\nsamples = 204\nvalue = [173, 31]'),
 Text(0.8611111111111112, 0.75, 'x[15] <= 0.992\ngini = 0.138\nsamples =
147\nvalue = [136, 11]'),
 Text(0.8545751633986928, 0.6944444444444444, 'x[4] <= 0.482\ngini =
0.128\nsamples = 146\nvalue = [136, 10]'),
 Text(0.8366013071895425, 0.6388888888888888, 'x[26] <= 0.063\ngini =
0.038\nsamples = 104\nvalue = [102, 2]'),
 Text(0.8300653594771242, 0.5833333333333334, 'x[9] <= 0.193\ngini =
0.32\nsamples = 10\nvalue = [8, 2]'),
 Text(0.8235294117647058, 0.5277777777777778, 'x[11] <= 0.625\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.8169934640522876, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
 Text(0.8300653594771242, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.8366013071895425, 0.5277777777777778, 'gini = 0.0\nsamples = 7\nvalue =
[7, 0]'),
 Text(0.8431372549019608, 0.5833333333333334, 'gini = 0.0\nsamples = 94\nvalue =
[94, 0]'),
 Text(0.8725490196078431, 0.6388888888888888, 'x[7] <= 0.167\ngini =
0.308\nsamples = 42\nvalue = [34, 8]'),
 Text(0.8562091503267973, 0.5833333333333334, 'x[25] <= 0.833\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.8496732026143791, 0.5277777777777778, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
 Text(0.8627450980392157, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.8888888888888888, 0.5833333333333334, 'x[0] <= 0.393\ngini =
0.229\nsamples = 38\nvalue = [33, 5]'),
 Text(0.8758169934640523, 0.5277777777777778, 'x[7] <= 0.5\ngini = 0.5\nsamples
= 6\nvalue = [3, 3]'),
 Text(0.869281045751634, 0.4722222222222222, 'x[4] <= 0.875\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.8627450980392157, 0.4166666666666667, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
 Text(0.8758169934640523, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.8823529411764706, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
 Text(0.9019607843137255, 0.5277777777777778, 'x[24] <= 0.917\ngini =
0.117\nsamples = 32\nvalue = [30, 2]'),
 Text(0.8954248366013072, 0.4722222222222222, 'x[12] <= 0.812\ngini =
0.062\nsamples = 31\nvalue = [30, 1]'),
 Text(0.8888888888888888, 0.4166666666666667, 'gini = 0.0\nsamples = 28\nvalue =
[28, 0]'),
 Text(0.9019607843137255, 0.4166666666666667, 'x[2] <= 0.561\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.8954248366013072, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.9084967320261438, 0.3611111111111111, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
 Text(0.9084967320261438, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.8676470588235294, 0.6944444444444444, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.9640522875816994, 0.75, 'x[12] <= 0.812\ngini = 0.456\nsamples =
57\nvalue = [37, 20]'),
 Text(0.9411764705882353, 0.6944444444444444, 'x[28] <= 0.4\ngini =
0.238\nsamples = 29\nvalue = [25, 4]'),
 Text(0.9281045751633987, 0.6388888888888888, 'x[9] <= 0.964\ngini =
0.142\nsamples = 26\nvalue = [24, 2]'),
 Text(0.9215686274509803, 0.5833333333333334, 'x[19] <= 0.75\ngini =
0.077\nsamples = 25\nvalue = [24, 1]'),
 Text(0.9150326797385621, 0.5277777777777778, 'gini = 0.0\nsamples = 23\nvalue =
[23, 0]'),
 Text(0.9281045751633987, 0.5277777777777778, 'x[23] <= 0.263\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.9215686274509803, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.934640522875817, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.934640522875817, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
 Text(0.954248366013072, 0.6388888888888888, 'x[28] <= 0.933\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.9477124183006536, 0.5833333333333334, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
 Text(0.9607843137254902, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
 Text(0.9869281045751634, 0.6944444444444444, 'x[28] <= 0.1\ngini =
0.49\nsamples = 28\nvalue = [12, 16]'),
 Text(0.9803921568627451, 0.6388888888888888, 'x[10] <= 0.833\ngini =
0.48\nsamples = 20\nvalue = [12, 8]'),
 Text(0.9738562091503268, 0.5833333333333334, 'x[26] <= 0.013\ngini =
0.415\nsamples = 17\nvalue = [12, 5]'),
 Text(0.9673202614379085, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
 Text(0.9803921568627451, 0.5277777777777778, 'x[16] <= 0.505\ngini =
0.32\nsamples = 15\nvalue = [12, 3]'),
 Text(0.9738562091503268, 0.4722222222222222, 'gini = 0.0\nsamples = 9\nvalue =
[9, 0]'),
 Text(0.9869281045751634, 0.4722222222222222, 'x[16] <= 0.706\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.9803921568627451, 0.4166666666666667, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
 Text(0.9934640522875817, 0.4166666666666667, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
 Text(0.9869281045751634, 0.5833333333333334, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
 Text(0.9934640522875817, 0.6388888888888888, 'gini = 0.0\nsamples = 8\nvalue =
[0, 8]')]
```



[242]:
```python
from sklearn.model_selection import GridSearchCV
parameter={
 'criterion':['gini','entropy'],
  'splitter':['best','random'],
  'max_depth':[1,2,3,4,5],
  'max_features':['auto', 'sqrt', 'log2']

}
```

[243]:
```python
grid_search=GridSearchCV(estimator=dtc,param_grid=parameter,cv=5,scoring="accuracy")
```

[244]:
```python
grid_search.fit(x_train,y_train)
```

```
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
```

```
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
```

```
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
```

```
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
```

```
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
```

```
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
```

```
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
```

```
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
```

```
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
```

```
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
C:\ProgramData\anaconda3\lib\site-packages\sklearn\tree\_classes.py:269:
FutureWarning: `max_features='auto'` has been deprecated in 1.1 and will be
removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
  warnings.warn(
```

```
[244]: GridSearchCV(cv=5, estimator=DecisionTreeClassifier(),
                    param_grid={'criterion': ['gini', 'entropy'],
                                'max_depth': [1, 2, 3, 4, 5],
                                'max_features': ['auto', 'sqrt', 'log2'],
                                'splitter': ['best', 'random']},
                    scoring='accuracy')
```

```
[245]: grid_search.best_params_
```

```
[245]: {'criterion': 'entropy',
        'max_depth': 3,
        'max_features': 'log2',
        'splitter': 'best'}
```

```
[246]: dtc_cv=DecisionTreeClassifier(criterion= 'entropy',
        max_depth=3,
        max_features='sqrt',
        splitter='best')
       dtc_cv.fit(x_train,y_train)
```

```
[246]: DecisionTreeClassifier(criterion='entropy', max_depth=3, max_features='sqrt')
```

```
[247]: pred=dtc_cv.predict(x_test)
       print(classification_report(y_test,pred))
```

```
              precision    recall  f1-score   support

           0       0.85      0.97      0.90       245
           1       0.43      0.12      0.19        49

    accuracy                           0.83       294
   macro avg       0.64      0.54      0.55       294
weighted avg       0.78      0.83      0.78       294
```

13 we can see the significant improvement in the accuracy after hyperparameter tuning

14 Random Forest Model Bulding

[249]:
```
from sklearn.ensemble import RandomForestClassifier
rfc=RandomForestClassifier()
```

[251]:
```
forest_params = [{'max_depth': list(range(10, 15)), 'max_features':␣
 ↪list(range(0,14))}]
```

[252]:
```
rfc_cv= GridSearchCV(rfc,param_grid=forest_params,cv=10,scoring="accuracy")
```

[253]:
```
rfc_cv.fit(x_train,y_train)
```

```
C:\ProgramData\anaconda3\lib\site-
packages\sklearn\model_selection\_validation.py:378: FitFailedWarning:
50 fits failed out of a total of 700.
The score on these train-test partitions for these parameters will be set to
nan.
If these failures are not expected, you can try to debug them by setting
error_score='raise'.

Below are more details about the failures:
--------------------------------------------------------------------------------
50 fits failed with the following error:
Traceback (most recent call last):
  File "C:\ProgramData\anaconda3\lib\site-
packages\sklearn\model_selection\_validation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File "C:\ProgramData\anaconda3\lib\site-packages\sklearn\ensemble\_forest.py",
line 340, in fit
    self._validate_params()
  File "C:\ProgramData\anaconda3\lib\site-packages\sklearn\base.py", line 581,
in _validate_params
    validate_parameter_constraints(
  File "C:\ProgramData\anaconda3\lib\site-
packages\sklearn\utils\_param_validation.py", line 97, in
validate_parameter_constraints
    raise InvalidParameterError(
sklearn.utils._param_validation.InvalidParameterError: The 'max_features'
parameter of RandomForestClassifier must be an int in the range [1, inf), a
float in the range (0.0, 1.0], a str among {'auto' (deprecated), 'log2', 'sqrt'}
or None. Got 0 instead.

  warnings.warn(some_fits_failed_message, FitFailedWarning)
C:\ProgramData\anaconda3\lib\site-
```

```
packages\sklearn\model_selection\_search.py:952: UserWarning: One or more of the
test scores are non-finite: [       nan 0.84950022 0.85631609 0.86137911
0.85967695 0.86140808
 0.86224105 0.85966971 0.86221932 0.86053165 0.85882225 0.86135738
 0.86053165 0.86306678        nan 0.85118065 0.85715631 0.86224105
 0.86224105 0.8596842  0.86393597 0.85628712 0.8622483  0.85882225
 0.85626539 0.85542518 0.85457048 0.85967695        nan 0.85035492
 0.85630885 0.85545415 0.86308127 0.85796755 0.86138635 0.86223381
 0.85373026 0.85797479 0.85883674 0.86307403 0.85540345 0.85626539
        nan 0.84863103 0.85542518 0.85883674 0.86224105 0.85798928
 0.86053165 0.86475445 0.86224105 0.85966247 0.86140084 0.86137187
 0.85882225 0.85625815        nan 0.84865276 0.85545415 0.85884398
 0.85882949 0.85967695 0.86054614 0.86223381 0.86308127 0.85798204
 0.86308127 0.85881501 0.85796031 0.86223381]
  warnings.warn(
```

```
[253]: GridSearchCV(cv=10, estimator=RandomForestClassifier(),
                    param_grid=[{'max_depth': [10, 11, 12, 13, 14],
                                 'max_features': [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11,
                                                  12, 13]}],
                    scoring='accuracy')
```

```
[254]: pred=rfc_cv.predict(x_test)
       print(classification_report(y_test,pred))
       print(rfc_cv.best_params_)
```

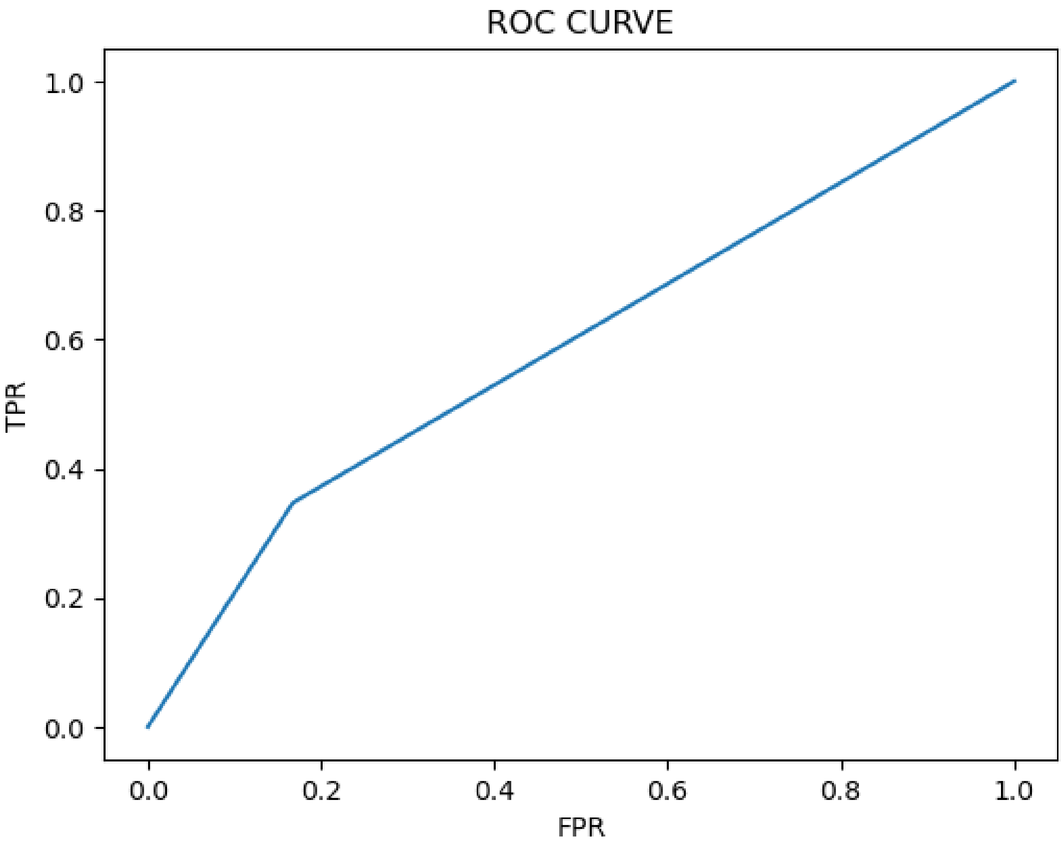
```
              precision    recall  f1-score   support

           0       0.86      0.98      0.92       245
           1       0.69      0.18      0.29        49

    accuracy                           0.85       294
   macro avg       0.77      0.58      0.60       294
weighted avg       0.83      0.85      0.81       294

{'max_depth': 13, 'max_features': 7}
```

```
[256]: probability=dtc.predict_proba(x_test)[:,1]
       probability
```

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

[257]:

```
# roc_curve
fpr,tpr,threshsholds = roc_curve(y_test,probability)
plt.plot(fpr,tpr)
plt.xlabel('FPR')
plt.ylabel('TPR')
plt.title('ROC CURVE')
plt.show()
```



[ ]: