

assignment4-thridiva-1

September 28, 2023

0.1 Gajjala Thridiva Reddy

morning slot(7am-9am)

0.2 Importing the libraries

```
[1]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

0.3 Importing the Dataset

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

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

```
[3]:
```

	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]

```
[4]: df.shape
```

```
[4]: (1470, 35)
```

```
[5]: 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

```

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

```

[6]:
count      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]

```
[7]: corr=df.corr()
      corr
```

<ipython-input-7-7d5195e2bf4d>: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.

```
corr=df.corr()
```

```
[7]:
```

	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
EmployeeCount	NaN	NaN	NaN	NaN
EmployeeNumber	-0.010145	-0.050990	0.032916	0.042070
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
StandardHours	NaN	NaN	NaN	NaN
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

	EmployeeCount	EmployeeNumber \
Age	NaN	-0.010145
DailyRate	NaN	-0.050990
DistanceFromHome	NaN	0.032916
Education	NaN	0.042070
EmployeeCount	NaN	NaN
EmployeeNumber	NaN	1.000000
EnvironmentSatisfaction	NaN	0.017621
HourlyRate	NaN	0.035179
JobInvolvement	NaN	-0.006888
JobLevel	NaN	-0.018519
JobSatisfaction	NaN	-0.046247
MonthlyIncome	NaN	-0.014829
MonthlyRate	NaN	0.012648
NumCompaniesWorked	NaN	-0.001251
PercentSalaryHike	NaN	-0.012944
PerformanceRating	NaN	-0.020359
RelationshipSatisfaction	NaN	-0.069861
StandardHours	NaN	NaN
StockOptionLevel	NaN	0.062227
TotalWorkingYears	NaN	-0.014365
TrainingTimesLastYear	NaN	0.023603

WorkLifeBalance	NaN	0.010309
YearsAtCompany	NaN	-0.011240
YearsInCurrentRole	NaN	-0.008416
YearsSinceLastPromotion	NaN	-0.009019
YearsWithCurrManager	NaN	-0.009197

	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
EmployeeCount	NaN	NaN	NaN
EmployeeNumber	0.017621	0.035179	-0.006888
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
StandardHours	NaN	NaN	NaN
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 ...	RelationshipSatisfaction \
Age	0.509604 ...	0.053535
DailyRate	0.002966 ...	0.007846
DistanceFromHome	0.005303 ...	0.006557
Education	0.101589 ...	-0.009118
EmployeeCount	NaN ...	NaN
EmployeeNumber	-0.018519 ...	-0.069861
EnvironmentSatisfaction	0.001212 ...	0.007665
HourlyRate	-0.027853 ...	0.001330
JobInvolvement	-0.012630 ...	0.034297
JobLevel	1.000000 ...	0.021642
JobSatisfaction	-0.001944 ...	-0.012454
MonthlyIncome	0.950300 ...	0.025873

MonthlyRate	0.039563	...	-0.004085
NumCompaniesWorked	0.142501	...	0.052733
PercentSalaryHike	-0.034730	...	-0.040490
PerformanceRating	-0.021222	...	-0.031351
RelationshipSatisfaction	0.021642	...	1.000000
StandardHours	NaN	...	NaN
StockOptionLevel	0.013984	...	-0.045952
TotalWorkingYears	0.782208	...	0.024054
TrainingTimesLastYear	-0.018191	...	0.002497
WorkLifeBalance	0.037818	...	0.019604
YearsAtCompany	0.534739	...	0.019367
YearsInCurrentRole	0.389447	...	-0.015123
YearsSinceLastPromotion	0.353885	...	0.033493
YearsWithCurrManager	0.375281	...	-0.000867

	StandardHours	StockOptionLevel	TotalWorkingYears	\
Age	NaN	0.037510	0.680381	
DailyRate	NaN	0.042143	0.014515	
DistanceFromHome	NaN	0.044872	0.004628	
Education	NaN	0.018422	0.148280	
EmployeeCount	NaN	NaN	NaN	
EmployeeNumber	NaN	0.062227	-0.014365	
EnvironmentSatisfaction	NaN	0.003432	-0.002693	
HourlyRate	NaN	0.050263	-0.002334	
JobInvolvement	NaN	0.021523	-0.005533	
JobLevel	NaN	0.013984	0.782208	
JobSatisfaction	NaN	0.010690	-0.020185	
MonthlyIncome	NaN	0.005408	0.772893	
MonthlyRate	NaN	-0.034323	0.026442	
NumCompaniesWorked	NaN	0.030075	0.237639	
PercentSalaryHike	NaN	0.007528	-0.020608	
PerformanceRating	NaN	0.003506	0.006744	
RelationshipSatisfaction	NaN	-0.045952	0.024054	
StandardHours	NaN	NaN	NaN	
StockOptionLevel	NaN	1.000000	0.010136	
TotalWorkingYears	NaN	0.010136	1.000000	
TrainingTimesLastYear	NaN	0.011274	-0.035662	
WorkLifeBalance	NaN	0.004129	0.001008	
YearsAtCompany	NaN	0.015058	0.628133	
YearsInCurrentRole	NaN	0.050818	0.460365	
YearsSinceLastPromotion	NaN	0.014352	0.404858	
YearsWithCurrManager	NaN	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
EmployeeCount	NaN	NaN
EmployeeNumber	0.023603	0.010309
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
StandardHours	NaN	NaN
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
EmployeeCount	NaN	NaN
EmployeeNumber	-0.011240	-0.008416
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
StandardHours	NaN	NaN
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
EmployeeCount	NaN	NaN
EmployeeNumber	-0.009019	-0.009197
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
StandardHours	NaN	NaN
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

[26 rows x 26 columns]

0.4 Checking the null values

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

```
[8]: Age                False
      Attrition          False
      BusinessTravel     False
      DailyRate          False
      Department         False
      DistanceFromHome   False
      Education           False
      EducationField      False
```

EmployeeCount	False
EmployeeNumber	False
EnvironmentSatisfaction	False
Gender	False
HourlyRate	False
JobInvolvement	False
JobLevel	False
JobRole	False
JobSatisfaction	False
MaritalStatus	False
MonthlyIncome	False
MonthlyRate	False
NumCompaniesWorked	False
Over18	False
OverTime	False
PercentSalaryHike	False
PerformanceRating	False
RelationshipSatisfaction	False
StandardHours	False
StockOptionLevel	False
TotalWorkingYears	False
TrainingTimesLastYear	False
WorkLifeBalance	False
YearsAtCompany	False
YearsInCurrentRole	False
YearsSinceLastPromotion	False
YearsWithCurrManager	False
dtype:	bool

```
[9]: df.isnull().sum()
```

```
[9]: Age           0
Attrition         0
BusinessTravel    0
DailyRate        0
Department       0
DistanceFromHome  0
Education         0
EducationField    0
EmployeeCount     0
EmployeeNumber    0
EnvironmentSatisfaction  0
Gender           0
HourlyRate       0
JobInvolvement    0
JobLevel         0
JobRole          0
```

```

JobSatisfaction      0
MaritalStatus        0
MonthlyIncome        0
MonthlyRate          0
NumCompaniesWorked   0
Over18               0
OverTime             0
PercentSalaryHike    0
PerformanceRating    0
RelationshipSatisfaction 0
StandardHours        0
StockOptionLevel     0
TotalWorkingYears    0
TrainingTimesLastYear 0
WorkLifeBalance      0
YearsAtCompany       0
YearsInCurrentRole    0
YearsSinceLastPromotion 0
YearsWithCurrManager  0
dtype: int64

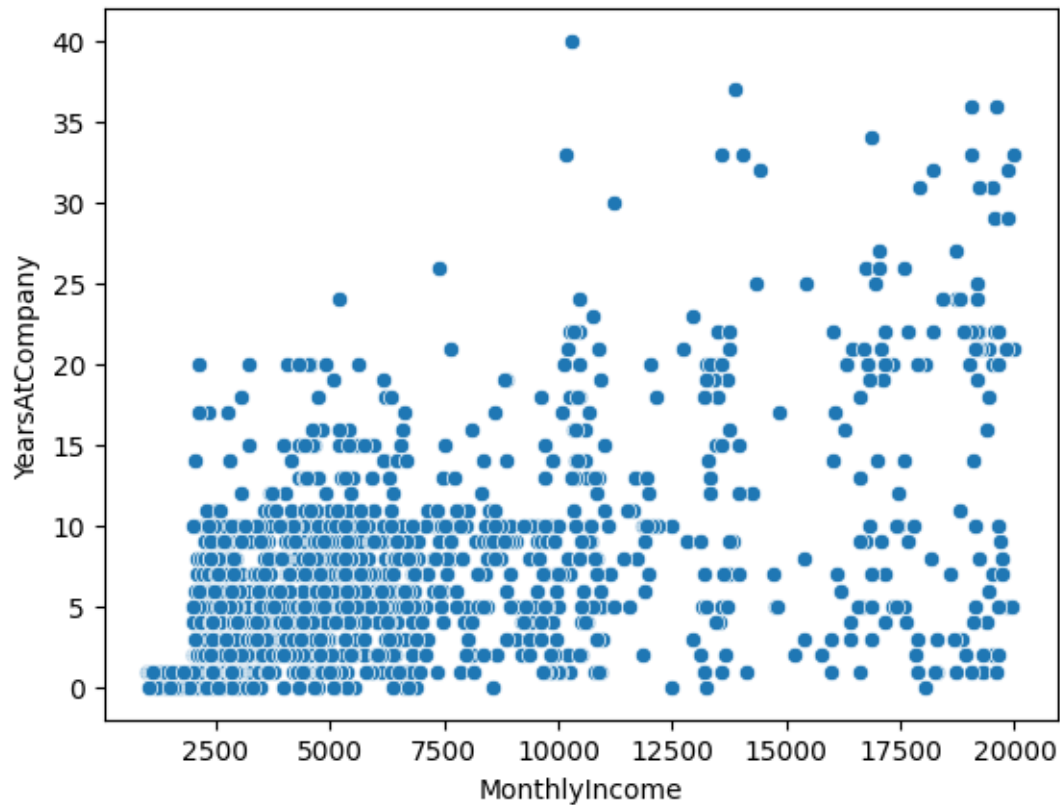
```

INFERENCE: Here we can see there is no null values

0.5 Data Visualization

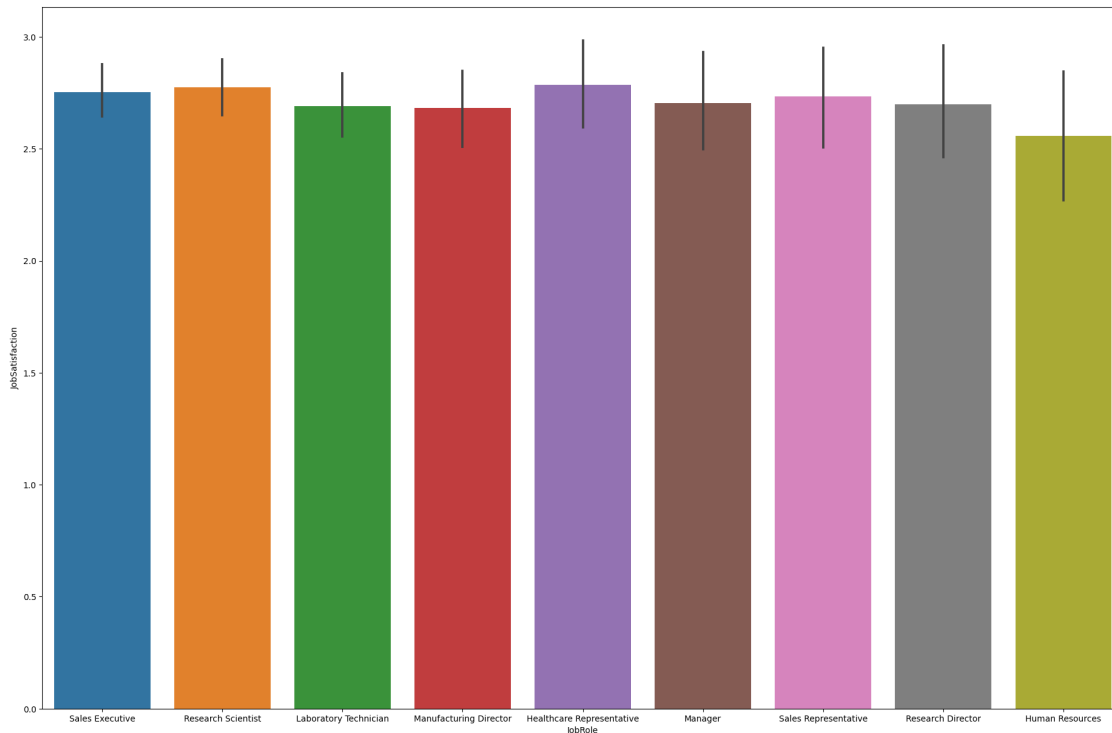
```
[10]: sns.scatterplot(data=df,x="MonthlyIncome",y="YearsAtCompany")
```

```
[10]: <Axes: xlabel='MonthlyIncome', ylabel='YearsAtCompany'>
```



INFERENCE: The above graph is plotted between the MonthlyIncome and YearsAtCompany, By this plot we can observe that when the monthly income is more YearsAtCompany of employee is more. and here we can see that highest is when monthly income is 10000

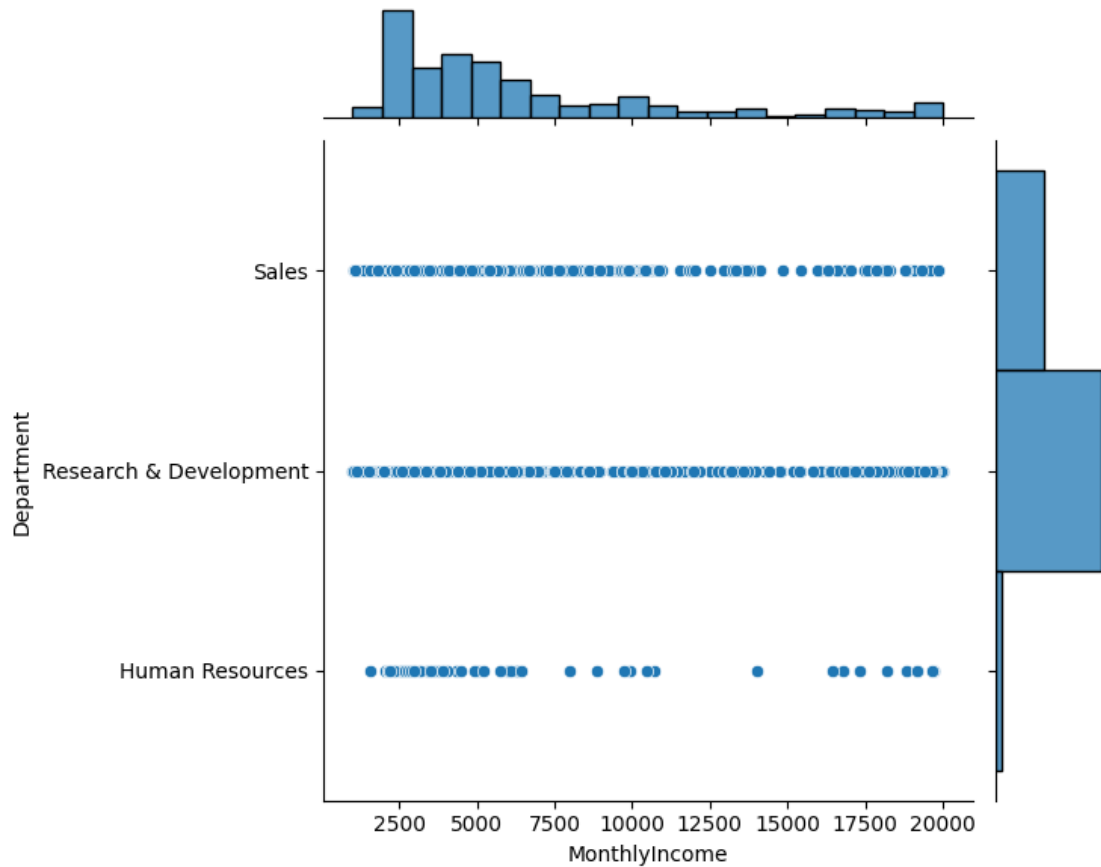
```
[11]: plt.figure(figsize=(23,15))
      sns.barplot(data=df,x="JobRole",y="JobSatisfaction")
      plt.show()
```



INFERENCE: The above graph is plotted between the Jobrole and Jobsatisfaction, by the above graph we can say the HealthcareRepresentative,salesExecutive,Research Scientist jobroles has approximately equal jobstatisfaction and lowest is Human Resources.

```
[12]: sns.jointplot(data=df,x="MonthlyIncome",y="Department")
```

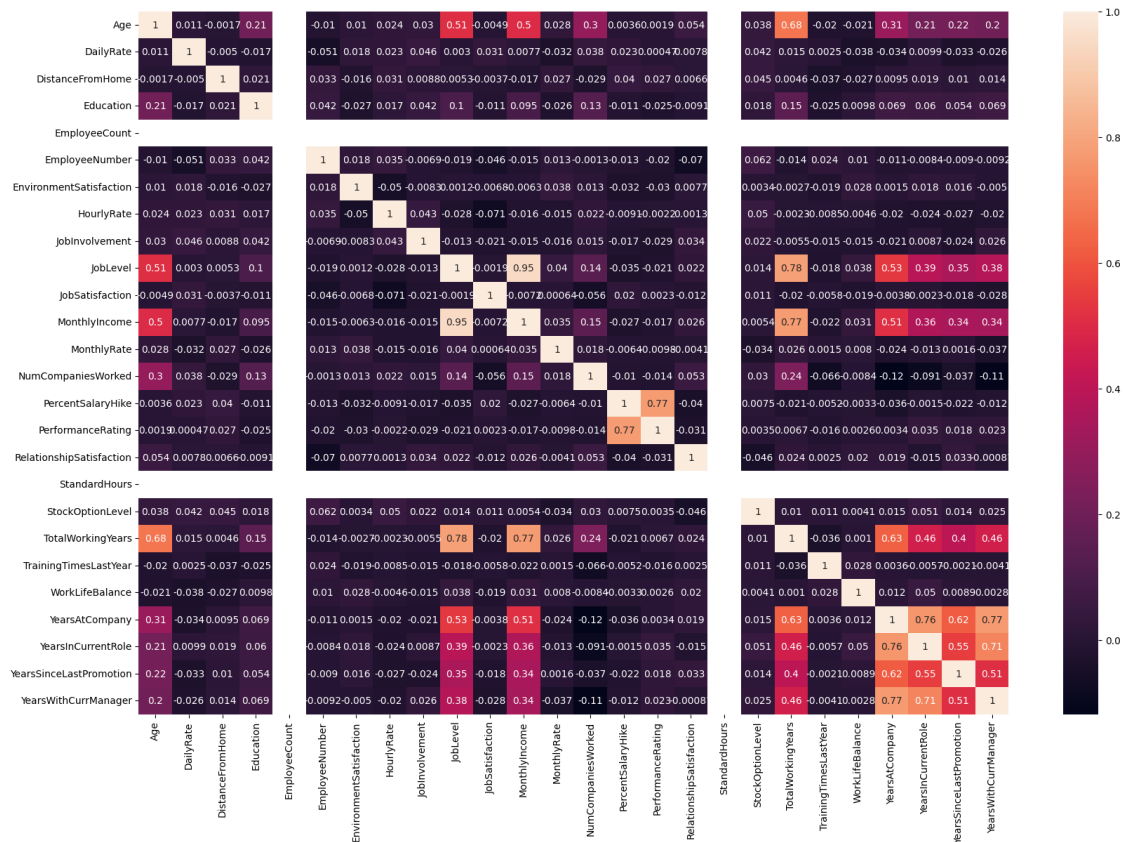
```
[12]: <seaborn.axisgrid.JointGrid at 0x7914f83ed450>
```



INFERENCE: By this plot we can say that the monthly income for Human Resources is not continuous and decreasing when compared to Research and Development & sales

INFERENCE: The above is plotted in Yearsatcompany and Job satisfaction which

```
[13]: plt.figure(figsize=(20,13))
sns.heatmap(corr,annot=True)
plt.show()
```



INFERENCE: We can say that these is the heat map of daataset which gives all the summary of the Data

```
[14]: sns.distplot(df["PercentSalaryHike"])
```

```
<ipython-input-14-3e10469594d7>:1: UserWarning:
```

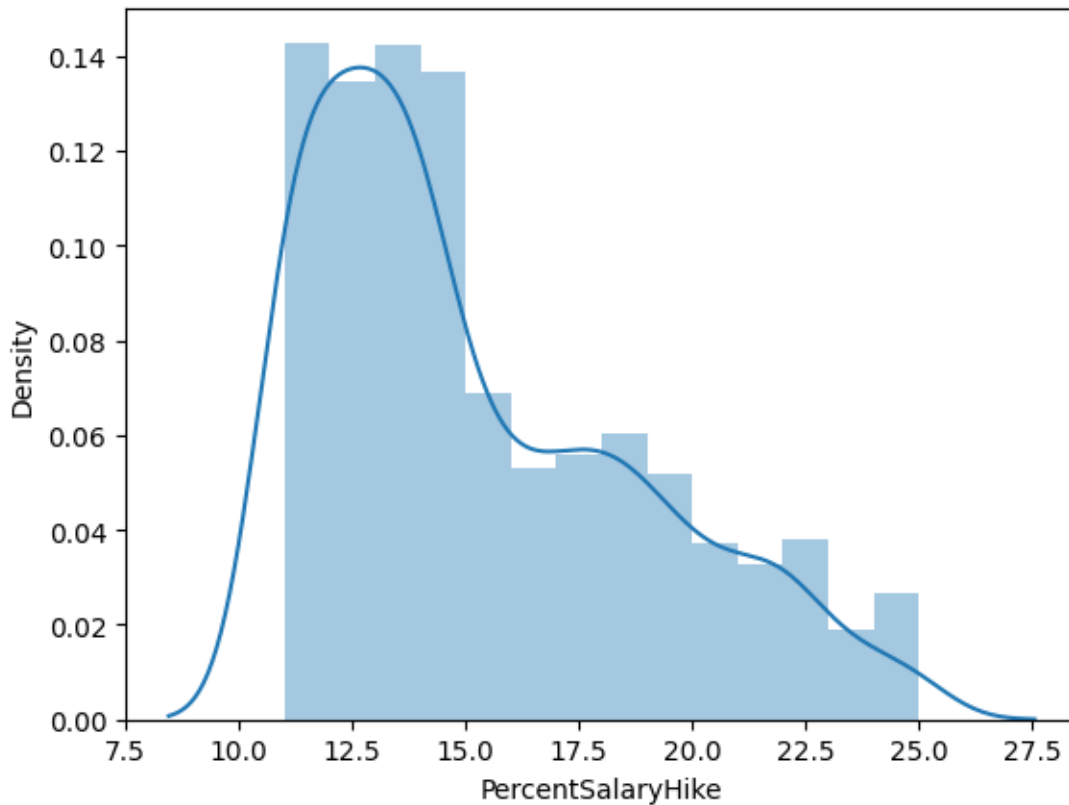
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

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["PercentSalaryHike"])
```

```
[14]: <Axes: xlabel='PercentSalaryHike', ylabel='Density'>
```



INFERENCE: Here is the distplot of the PercenatgeSalaryHike which gives us the relation between the PercenatgeSalaryHike abd Density By the above graph we can say that at 11 and 12.5 reached highest and started decreasing from 15

```
[15]: sns.distplot(df["Age"])
```

```
<ipython-input-15-cf0334540b62>:1: UserWarning:
```

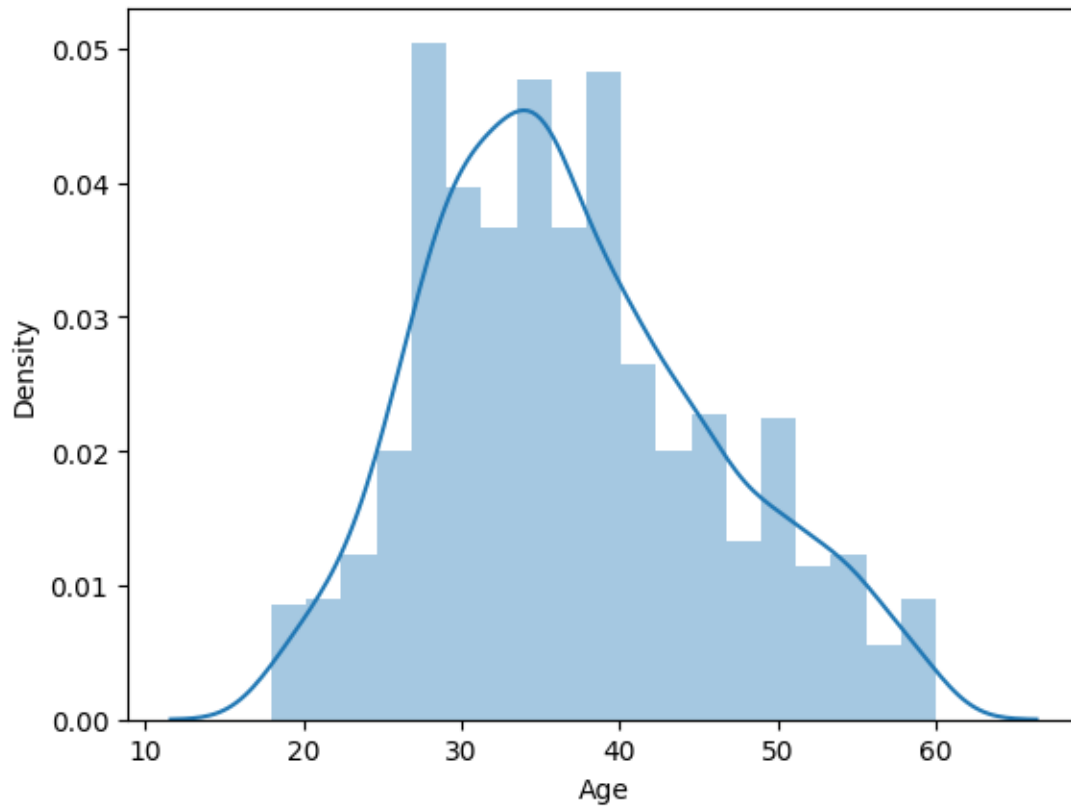
```
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.
```

```
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["Age"])
```

```
[15]: <Axes: xlabel='Age', ylabel='Density'>
```

INFERENCE: Here is the distplot of the Age by this plot we can say that at age 25 reached highest and keeps on decreasing.

```
[16]: sns.relplot(data=df,x="TotalWorkingYears",y="JobRole")
```

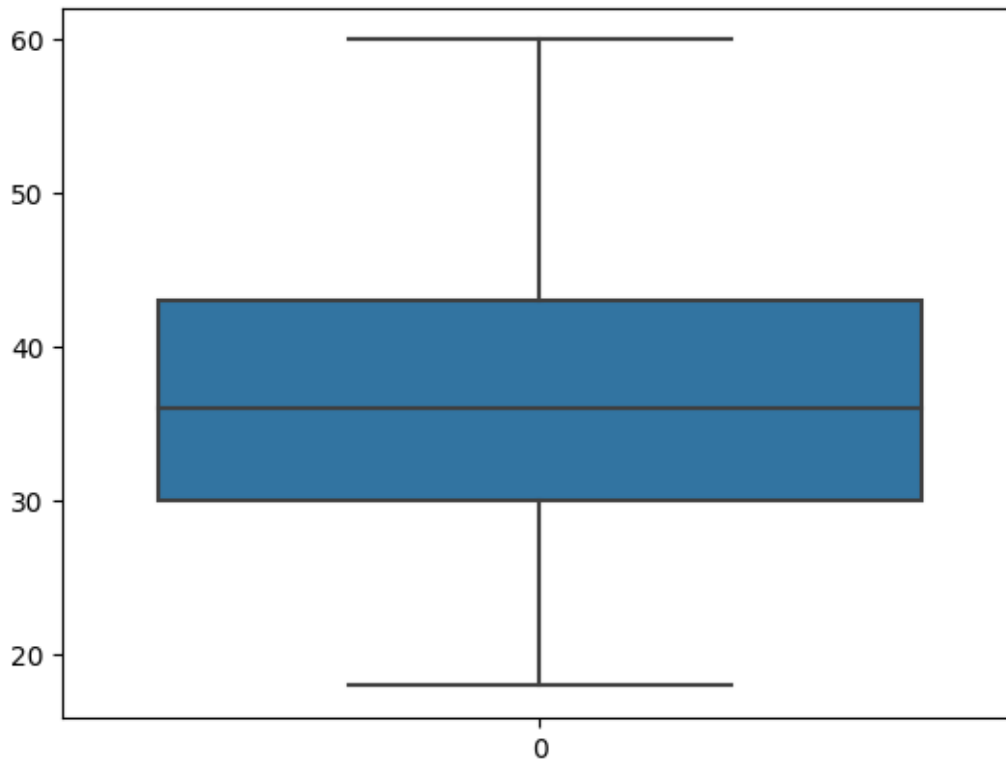
```
[16]: <seaborn.axisgrid.FacetGrid at 0x7914f5d1b1f0>
```



INFERENCE: By this relplot we can say that in Research Scientist and Laboratory Technician the totalWorkingYears are fullest and less working hours is Manager role

```
[17]: sns.boxplot(df["Age"])
```

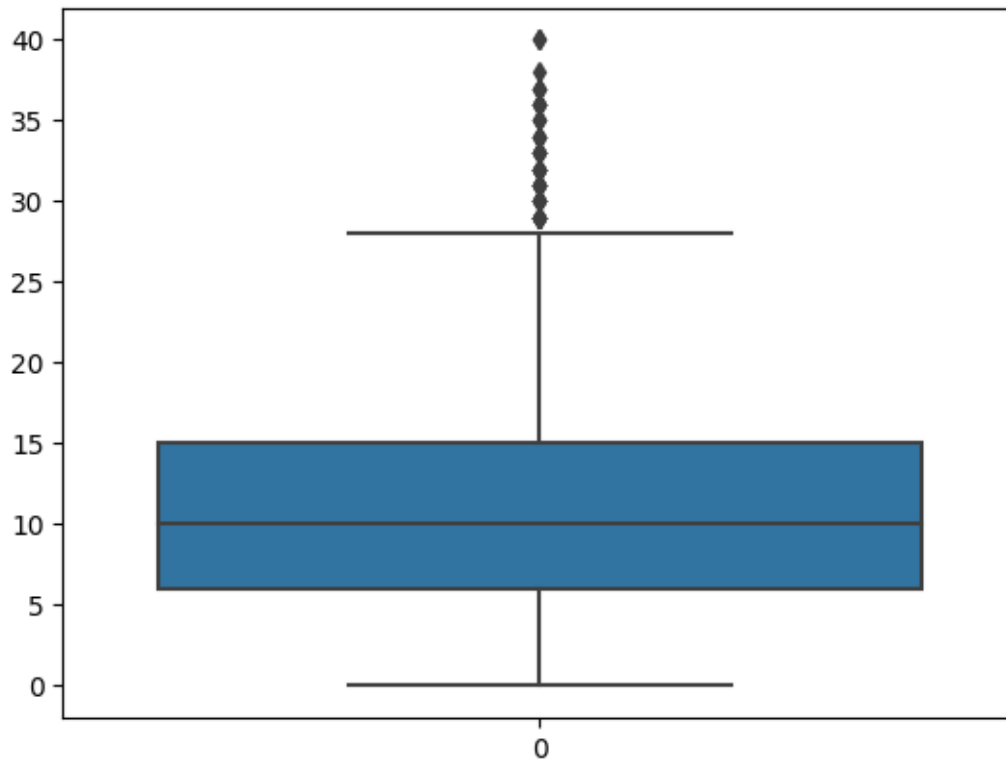
```
[17]: <Axes: >
```



0.6 Checking Outliers

```
[18]: sns.boxplot(df["TotalWorkingYears"])
```

```
[18]: <Axes: >
```



INFERENCE: INFERENCE: Here We can see there are some outliers, We have to follow the outliers removing techniques and remove them

1. Inter Quatile Range(IQR Method)

2. Z-Score Method

3. Percentile Method

we can remove the outliers by using the any one of those methods. Outlier removal by replacement with median

```
[19]: q1=df.TotalWorkingYears.quantile(0.25)
      q3=df.TotalWorkingYears.quantile(0.75)
```

```
[20]: q1
```

```
[20]: 6.0
```

```
[21]: q3
```

```
[21]: 15.0
```

```
[22]: IQR=q3-q1  
IQR
```

```
[22]: 9.0
```

```
[23]: upper_limit=q1+1.5*IQR
```

```
[24]: upper_limit
```

```
[24]: 19.5
```

```
[25]: median_TotalWorkingYears=df.TotalWorkingYears.median()  
median_TotalWorkingYears
```

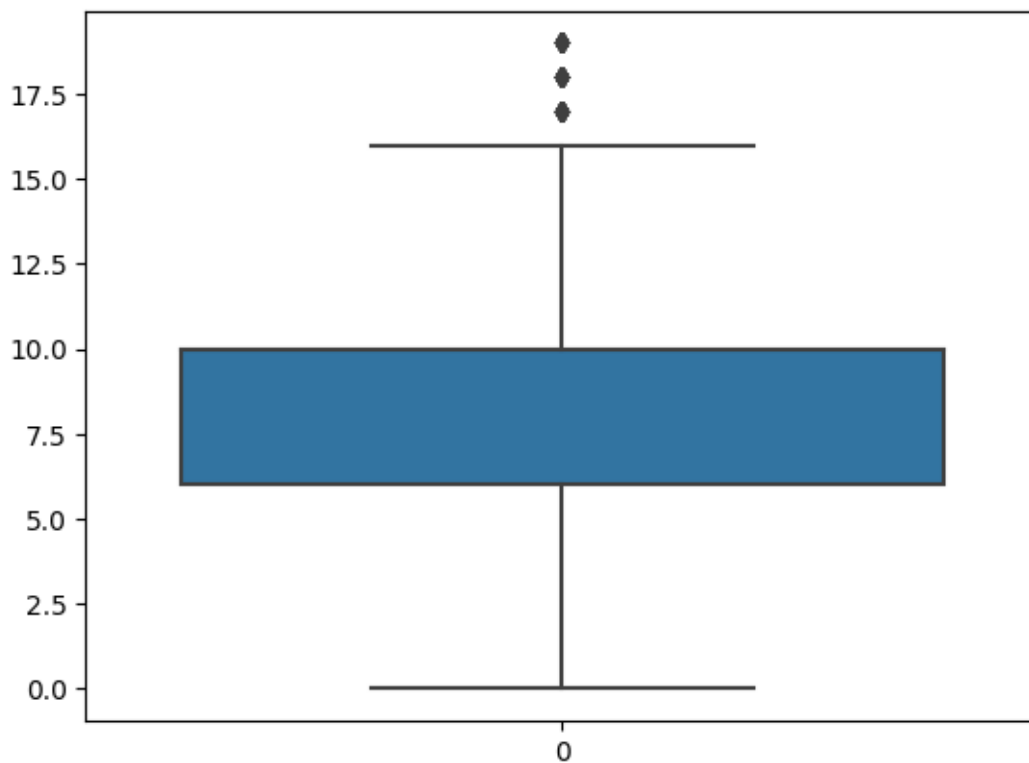
```
[25]: 10.0
```

Handling the Outliers

```
[26]: df["TotalWorkingYears"]=np.  
      ↪where(df["TotalWorkingYears"]>upper_limit,median_TotalWorkingYears,df["TotalWorkingYears"])
```

```
[27]: sns.boxplot(df["TotalWorkingYears"])
```

```
[27]: <Axes: >
```



0.7 spiliting the Dependent and Independent values

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

```
[28]:   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           0           1           6
1          10.0           3           3          10
2           7.0           3           3           0
3           8.0           3           3           8
4           6.0           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]
```

```
[29]: x_a=df.iloc[:,0:1]
      x_a
```

[29]: Age

```
0    41
1    49
2    37
3    33
4    27
...
1465 36
1466 39
1467 27
1468 49
1469 34
```

[1470 rows x 1 columns]

```
[30]: x=df.iloc[:,2:36]
x
```

```
[30]:      BusinessTravel  DailyRate      Department  DistanceFromHome  \
0      Travel_Rarely      1102      Sales      1
1  Travel_Frequently      279  Research & Development      8
2      Travel_Rarely     1373  Research & Development      2
3  Travel_Frequently     1392  Research & Development      3
4      Travel_Rarely      591  Research & Development      2
...
1465  Travel_Frequently      884  Research & Development     23
1466      Travel_Rarely      613  Research & Development      6
1467      Travel_Rarely      155  Research & Development      4
1468  Travel_Frequently     1023      Sales      2
1469      Travel_Rarely      628  Research & Development      8
```

```
      Education  EducationField  EmployeeCount  EmployeeNumber  \
0            2  Life Sciences            1            1
1            1  Life Sciences            1            2
2            2      Other            1            4
3            4  Life Sciences            1            5
4            1      Medical            1            7
...
1465      2      Medical            1          2061
1466      1      Medical            1          2062
1467      3  Life Sciences            1          2064
1468      3      Medical            1          2065
1469      3      Medical            1          2068
```

```
      EnvironmentSatisfaction  Gender  ...  RelationshipSatisfaction  \
0            2  Female  ...      1
1            3  Male  ...      4
```

2	4	Male	...	2
3	4	Female	...	3
4	1	Male	...	4
...
1465	3	Male	...	3
1466	4	Male	...	1
1467	2	Male	...	2
1468	4	Male	...	4
1469	2	Male	...	1

	StandardHours	StockOptionLevel	TotalWorkingYears	\
0	80	0	8.0	
1	80	1	10.0	
2	80	0	7.0	
3	80	0	8.0	
4	80	1	6.0	
...	
1465	80	1	17.0	
1466	80	1	9.0	
1467	80	1	6.0	
1468	80	0	17.0	
1469	80	0	6.0	

	TrainingTimesLastYear	WorkLifeBalance	YearsAtCompany	\
0	0	1	6	
1	3	3	10	
2	3	3	0	
3	3	3	8	
4	3	3	2	
...	
1465	3	3	5	
1466	5	3	7	
1467	0	3	6	
1468	3	2	9	
1469	3	4	4	

	YearsInCurrentRole	YearsSinceLastPromotion	YearsWithCurrManager
0	4	0	5
1	7	1	7
2	0	0	0
3	7	3	0
4	2	2	2
...
1465	2	0	3
1466	7	1	7
1467	2	0	3
1468	6	0	8


```
[1470 rows x 33 columns]
```

```
[31]:
```

	BusinessTravel	DailyRate	Department	DistanceFromHome	\
0	Travel_Rarely	1102	Sales		1
1	Travel_Frequently	279	Research & Development		8
2	Travel_Rarely	1373	Research & Development		2
3	Travel_Frequently	1392	Research & Development		3
4	Travel_Rarely	591	Research & Development		2
...	
1465	Travel_Frequently	884	Research & Development		23
1466	Travel_Rarely	613	Research & Development		6
1467	Travel_Rarely	155	Research & Development		4
1468	Travel_Frequently	1023	Sales		2
1469	Travel_Rarely	628	Research & Development		8

	Education	EducationField	EmployeeCount	EmployeeNumber \
0	2	Life Sciences	1	1
1	1	Life Sciences	1	2
2	2	Other	1	4
3	4	Life Sciences	1	5
4	1	Medical	1	7
...
1465	2	Medical	1	2061
1466	1	Medical	1	2062
1467	3	Life Sciences	1	2064
1468	3	Medical	1	2065
1469	3	Medical	1	2068

	EnvironmentSatisfaction	Gender	...	StandardHours	StockOptionLevel	\
0		2	Female	...	80	0
1		3	Male	...	80	1
2		4	Male	...	80	0
3		4	Female	...	80	0
4		1	Male	...	80	1
...	
1465		3	Male	...	80	1
1466		4	Male	...	80	1
1467		2	Male	...	80	1
1468		4	Male	...	80	0
1469		2	Male	...	80	0

25

0	8.0	0	1	6
1	10.0	3	3	10
2	7.0	3	3	0
3	8.0	3	3	8
4	6.0	3	3	2
...
1465	17.0	3	3	5
1466	9.0	5	3	7
1467	6.0	0	3	6
1468	17.0	3	2	9
1469	6.0	3	4	4

	YearsInCurrentRole	YearsSinceLastPromotion	YearsWithCurrManager	Age
0	4	0	5	41
1	7	1	7	49
2	0	0	0	37
3	7	3	0	33
4	2	2	2	27
...
1465	2	0	3	36
1466	7	1	7	39
1467	2	0	3	27
1468	6	0	8	49
1469	3	1	2	34

[1470 rows x 34 columns]

```
[32]: y=df["Attrition"]
      y
```

```
[32]: 0      Yes
      1      No
      2      Yes
      3      No
      4      No
      ...
      1465    No
      1466    No
      1467    No
      1468    No
      1469    No
      Name: Attrition, Length: 1470, dtype: object
```

```
[33]: type(y)
```

```
[33]: pandas.core.series.Series
```

```
[34]: type(x)
```

```
[34]: pandas.core.frame.DataFrame
```

0.8 Encoding

```
[35]: from sklearn.preprocessing import LabelEncoder  
le=LabelEncoder()
```

```
[36]: x["Gender"]=le.fit_transform(x["Gender"])  
x["Gender"]
```

```
[36]: 0      0  
1      1  
2      1  
3      0  
4      1  
...  
1465   1  
1466   1  
1467   1  
1468   1  
1469   1  
Name: Gender, Length: 1470, dtype: int64
```

```
[37]: x["Department"]=le.fit_transform(x["Department"])  
x["Department"]
```

```
[37]: 0      2  
1      1  
2      1  
3      1  
4      1  
...  
1465   1  
1466   1  
1467   1  
1468   2  
1469   1  
Name: Department, Length: 1470, dtype: int64
```

```
[38]: x["EducationField"]=le.fit_transform(x["EducationField"])  
x["EducationField"]
```

```
[38]: 0      1  
1      1  
2      4
```

```

3      1
4      3
..
1465   3
1466   3
1467   1
1468   3
1469   3
Name: EducationField, Length: 1470, dtype: int64

```

```
[39]: x["BusinessTravel"]=le.fit_transform(x["BusinessTravel"])
      x["BusinessTravel"]
```

```

[39]: 0      2
      1      1
      2      2
      3      1
      4      2
      ..
1465   1
1466   2
1467   2
1468   1
1469   2
Name: BusinessTravel, Length: 1470, dtype: int64

```

```
[40]: y=pd.DataFrame(y,columns=["Attrition"])
```

```
[41]: y["Attrition"]=le.fit_transform(y["Attrition"])
      y["Attrition"]
```

```

[41]: 0      1
      1      0
      2      1
      3      0
      4      0
      ..
1465   0
1466   0
1467   0
1468   0
1469   0
Name: Attrition, Length: 1470, dtype: int64

```

```
[42]: x["OverTime"]=le.fit_transform(x["OverTime"])
      x["OverTime"]
```

```
[42]: 0      1
      1      0
      2      1
      3      1
      4      0
      ..
     1465    0
     1466    0
     1467    1
     1468    0
     1469    0
      Name: OverTime, Length: 1470, dtype: int64
```

```
[43]: x["Over18"]=le.fit_transform(x["Over18"])
      x["Over18"]
```

```
[43]: 0      0
      1      0
      2      0
      3      0
      4      0
      ..
     1465    0
     1466    0
     1467    0
     1468    0
     1469    0
      Name: Over18, Length: 1470, dtype: int64
```

```
[44]: x["MaritalStatus"]=le.fit_transform(x["MaritalStatus"])
      x["MaritalStatus"]
```

```
[44]: 0      2
      1      1
      2      2
      3      1
      4      1
      ..
     1465    1
     1466    1
     1467    1
     1468    1
     1469    1
      Name: MaritalStatus, Length: 1470, dtype: int64
```

```
[45]: x["JobRole"]=le.fit_transform(x["JobRole"])
      x["JobRole"]
```

```
[45]: 0      7
      1      6
      2      2
      3      6
      4      2
      ..
     1465    2
     1466    0
     1467    4
     1468    7
     1469    2
      Name: JobRole, Length: 1470, dtype: int64
```

```
[46]: x.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1470 entries, 0 to 1469
Data columns (total 34 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   BusinessTravel                       1470 non-null   int64
1   DailyRate                           1470 non-null   int64
2   Department                          1470 non-null   int64
3   DistanceFromHome                    1470 non-null   int64
4   Education                           1470 non-null   int64
5   EducationField                      1470 non-null   int64
6   EmployeeCount                      1470 non-null   int64
7   EmployeeNumber                      1470 non-null   int64
8   EnvironmentSatisfaction              1470 non-null   int64
9   Gender                              1470 non-null   int64
10  HourlyRate                          1470 non-null   int64
11  JobInvolvement                      1470 non-null   int64
12  JobLevel                            1470 non-null   int64
13  JobRole                             1470 non-null   int64
14  JobSatisfaction                     1470 non-null   int64
15  MaritalStatus                      1470 non-null   int64
16  MonthlyIncome                      1470 non-null   int64
17  MonthlyRate                        1470 non-null   int64
18  NumCompaniesWorked                  1470 non-null   int64
19  Over18                             1470 non-null   int64
20  OverTime                           1470 non-null   int64
21  PercentSalaryHike                   1470 non-null   int64
22  PerformanceRating                   1470 non-null   int64
23  RelationshipSatisfaction             1470 non-null   int64
24  StandardHours                      1470 non-null   int64
25  StockOptionLevel                   1470 non-null   int64
26  TotalWorkingYears                   1470 non-null   float64
```

```

27 TrainingTimesLastYear    1470 non-null    int64
28 WorkLifeBalance          1470 non-null    int64
29 YearsAtCompany           1470 non-null    int64
30 YearsInCurrentRole       1470 non-null    int64
31 YearsSinceLastPromotion  1470 non-null    int64
32 YearsWithCurrManager     1470 non-null    int64
33 Age                      1470 non-null    int64
dtypes: float64(1), int64(33)
memory usage: 390.6 KB

```

```
[47]: type(y)
```

```
[47]: pandas.core.frame.DataFrame
```

```
[48]: y=y["Attrition"]
y
```

```

[48]: 0      1
      1      0
      2      1
      3      0
      4      0
      ..
     1465    0
     1466    0
     1467    0
     1468    0
     1469    0
Name: Attrition, Length: 1470, dtype: int64

```

```
[49]: type(y)
```

```
[49]: pandas.core.series.Series
```

```
[50]: x.head()
```

```

[50]:   BusinessTravel  DailyRate  Department  DistanceFromHome  Education  \
0              2      1102           2             1          2
1              1       279           1             8          1
2              2      1373           1             2          2
3              1      1392           1             3          4
4              2       591           1             2          1

      EducationField  EmployeeCount  EmployeeNumber  EnvironmentSatisfaction  \
0              1              1              1              2
1              1              1              2              3
2              4              1              4              4

```

3		1	1	5	4
4		3	1	7	1

	Gender	...	StandardHours	StockOptionLevel	TotalWorkingYears	\
0	0	...	80	0	8.0	
1	1	...	80	1	10.0	
2	1	...	80	0	7.0	
3	0	...	80	0	8.0	
4	1	...	80	1	6.0	

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

	YearsSinceLastPromotion	YearsWithCurrManager	Age
0	0	5	41
1	1	7	49
2	0	0	37
3	3	0	33
4	2	2	27

[5 rows x 34 columns]

[51]: x.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1470 entries, 0 to 1469
Data columns (total 34 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   BusinessTravel                       1470 non-null   int64
1   DailyRate                           1470 non-null   int64
2   Department                           1470 non-null   int64
3   DistanceFromHome                    1470 non-null   int64
4   Education                            1470 non-null   int64
5   EducationField                       1470 non-null   int64
6   EmployeeCount                       1470 non-null   int64
7   EmployeeNumber                      1470 non-null   int64
8   EnvironmentSatisfaction              1470 non-null   int64
9   Gender                              1470 non-null   int64
10  HourlyRate                           1470 non-null   int64
11  JobInvolvement                      1470 non-null   int64
12  JobLevel                            1470 non-null   int64
13  JobRole                             1470 non-null   int64
```



```

14  JobSatisfaction          1470 non-null  int64
15  MaritalStatus           1470 non-null  int64
16  MonthlyIncome           1470 non-null  int64
17  MonthlyRate             1470 non-null  int64
18  NumCompaniesWorked      1470 non-null  int64
19  Over18                  1470 non-null  int64
20  OverTime                1470 non-null  int64
21  PercentSalaryHike       1470 non-null  int64
22  PerformanceRating       1470 non-null  int64
23  RelationshipSatisfaction 1470 non-null  int64
24  StandardHours           1470 non-null  int64
25  StockOptionLevel        1470 non-null  int64
26  TotalWorkingYears       1470 non-null  float64
27  TrainingTimesLastYear   1470 non-null  int64
28  WorkLifeBalance         1470 non-null  int64
29  YearsAtCompany          1470 non-null  int64
30  YearsInCurrentRole      1470 non-null  int64
31  YearsSinceLastPromotion 1470 non-null  int64
32  YearsWithCurrManager    1470 non-null  int64
33  Age                    1470 non-null  int64
dtypes: float64(1), int64(33)
memory usage: 390.6 KB

```

0.9 Feature Scaling

```

[52]: from sklearn.preprocessing import MinMaxScaler
      ms=MinMaxScaler()
      x_scaled=pd.DataFrame(ms.fit_transform(x),columns=x.columns)

```

```

[53]: x_scaled.head()

```

```

[53]:   BusinessTravel  DailyRate  Department  DistanceFromHome  Education  \
0             1.0    0.715820           1.0           0.000000         0.25
1             0.5    0.126700           0.5           0.250000         0.00
2             1.0    0.909807           0.5           0.035714         0.25
3             0.5    0.923407           0.5           0.071429         0.75
4             1.0    0.350036           0.5           0.035714         0.00

      EducationField  EmployeeCount  EmployeeNumber  EnvironmentSatisfaction  \
0             0.2             0.0           0.000000             0.333333
1             0.2             0.0           0.000484             0.666667
2             0.8             0.0           0.001451             1.000000
3             0.2             0.0           0.001935             1.000000
4             0.6             0.0           0.002903             0.000000

      Gender  ...  StandardHours  StockOptionLevel  TotalWorkingYears  \
0      0.0  ...             0.0           0.000000           0.421053

```

1	1.0	...	0.0	0.333333	0.526316
2	1.0	...	0.0	0.000000	0.368421
3	0.0	...	0.0	0.000000	0.421053
4	1.0	...	0.0	0.333333	0.315789

	TrainingTimesLastYear	WorkLifeBalance	YearsAtCompany	YearsInCurrentRole	\
0	0.0	0.000000	0.15	0.222222	
1	0.5	0.666667	0.25	0.388889	
2	0.5	0.666667	0.00	0.000000	
3	0.5	0.666667	0.20	0.388889	
4	0.5	0.666667	0.05	0.111111	

	YearsSinceLastPromotion	YearsWithCurrManager	Age
0	0.000000	0.294118	0.547619
1	0.066667	0.411765	0.738095
2	0.000000	0.000000	0.452381
3	0.200000	0.000000	0.357143
4	0.133333	0.117647	0.214286

[5 rows x 34 columns]

0.10 Train test split

```
[54]: from sklearn.model_selection import train_test_split
```

```
[55]: x_train,x_test,y_train,y_test=train_test_split(x_scaled,y,test_size=0.
↪2,random_state=0)
```

```
[56]: x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

```
[56]: ((1176, 34), (294, 34), (1176,), (294,))
```

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

```
[57]:
```

	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	\
1374	1.0	0.360057	1.0	0.714286	0.50	
1092	1.0	0.607015	0.5	0.964286	0.50	
768	1.0	0.141732	1.0	0.892857	0.50	
569	0.0	0.953472	1.0	0.250000	0.75	
911	0.5	0.355762	1.0	0.821429	0.00	

	EducationField	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	\
1374	0.2	0.0	0.937107	1.000000	
1092	1.0	0.0	0.747460	1.000000	
768	0.4	0.0	0.515239	0.666667	
569	0.2	0.0	0.381229	0.000000	
911	0.2	0.0	0.615385	0.666667	

	Gender	...	StandardHours	StockOptionLevel	TotalWorkingYears	\
1374	0.0	...	0.0	0.333333	0.526316	
1092	1.0	...	0.0	0.333333	0.421053	
768	1.0	...	0.0	0.333333	0.421053	
569	1.0	...	0.0	0.000000	0.526316	
911	1.0	...	0.0	0.000000	0.052632	

	TrainingTimesLastYear	WorkLifeBalance	YearsAtCompany	\
1374	0.333333	0.333333	0.025	
1092	0.500000	0.666667	0.125	
768	0.500000	0.333333	0.175	
569	0.166667	0.666667	0.250	
911	0.666667	0.666667	0.025	

	YearsInCurrentRole	YearsSinceLastPromotion	YearsWithCurrManager	\
1374	0.000000	0.000000	0.000000	
1092	0.222222	0.000000	0.176471	
768	0.388889	0.466667	0.294118	
569	0.388889	0.000000	0.529412	
911	0.000000	0.066667	0.000000	

	Age
1374	0.952381
1092	0.642857
768	0.523810
569	0.428571
911	0.166667

[5 rows x 34 columns]

```
[58]: x_test.head()
```

	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	\
442	0.0	0.381532	1.0	0.321429	0.75	
1091	1.0	0.338583	0.5	0.857143	0.50	
981	0.5	0.400859	1.0	0.607143	0.75	
785	1.0	0.994989	0.5	0.678571	0.75	
1332	0.5	0.255548	0.5	0.821429	0.25	

	EducationField	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	\
442	0.6	0.0	0.285922	0.333333	
1091	0.2	0.0	0.746976	1.000000	
981	0.4	0.0	0.667150	1.000000	
785	1.0	0.0	0.527818	0.000000	
1332	0.2	0.0	0.903241	1.000000	

	Gender	...	StandardHours	StockOptionLevel	TotalWorkingYears	\
442	1.0	...	0.0	0.000000	0.526316	
1091	1.0	...	0.0	0.000000	0.263158	
981	0.0	...	0.0	0.333333	0.263158	
785	1.0	...	0.0	0.333333	0.736842	
1332	1.0	...	0.0	0.000000	0.052632	

	TrainingTimesLastYear	WorkLifeBalance	YearsAtCompany	\
442	0.500000	0.333333	0.250	
1091	0.333333	0.666667	0.125	
981	0.000000	0.333333	0.100	
785	1.000000	0.666667	0.275	
1332	0.500000	0.333333	0.025	

	YearsInCurrentRole	YearsSinceLastPromotion	YearsWithCurrManager	\
442	0.166667	0.600000	0.411765	
1091	0.166667	0.000000	0.117647	
981	0.111111	0.200000	0.117647	
785	0.555556	0.733333	0.058824	
1332	0.000000	0.066667	0.000000	

	Age
442	0.428571
1091	0.357143
981	0.404762
785	0.523810
1332	0.261905

[5 rows x 34 columns]

```
[59]: y_test.head
```

```
[59]: <bound method NDFrame.head of 442      0
1091      0
981       1
785       0
1332      1
..
1439      0
481       0
124       1
198       0
1229      0
Name: Attrition, Length: 294, dtype: int64>
```

0.11 Model Building

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

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

```
[61]: LogisticRegression()
```

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

```
[63]: pred
```

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

```
[64]: y_test
```

```
[64]: 442      0
      1091    0
      981     1
      785     0
      1332    1
      ..
     1439     0
      481     0
      124     1
      198     0
     1229     0
      Name: Attrition, Length: 294, dtype: int64
```

```
[65]: x.head()
```

```
[65]:   BusinessTravel  DailyRate  Department  DistanceFromHome  Education \
0                2        1102           2                  1          2
```

1	1	279	1	8	1
2	2	1373	1	2	2
3	1	1392	1	3	4
4	2	591	1	2	1

	EducationField	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	\
0	1	1	1		2
1	1	1	2		3
2	4	1	4		4
3	1	1	5		4
4	3	1	7		1

	Gender	...	StandardHours	StockOptionLevel	TotalWorkingYears	\
0	0	...	80	0	8.0	
1	1	...	80	1	10.0	
2	1	...	80	0	7.0	
3	0	...	80	0	8.0	
4	1	...	80	1	6.0	

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

	YearsSinceLastPromotion	YearsWithCurrManager	Age
0	0	5	41
1	1	7	49
2	0	0	37
3	3	0	33
4	2	2	27

[5 rows x 34 columns]

0.12 Evaluation of classification model

```
[66]: from sklearn.metrics import
      ↪ accuracy_score, confusion_matrix, classification_report, roc_auc_score, roc_curve, precision_score
```

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

```
[67]: 0.8775510204081632
```

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

```
[68]: array([[241,  4],
           [ 32, 17]])
```

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

```
[69]: col_0      0    1
Attrition
0         241    4
1          32   17
```

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

	precision	recall	f1-score	support
0	0.88	0.98	0.93	245
1	0.81	0.35	0.49	49
accuracy			0.88	294
macro avg	0.85	0.67	0.71	294
weighted avg	0.87	0.88	0.86	294

```
[71]: precision_score(y_test,pred)
```

```
[71]: 0.8095238095238095
```

```
[72]: recall_score(y_test,pred)
```

```
[72]: 0.3469387755102041
```

```
[73]: f1_score(y_test,pred)
```

```
[73]: 0.4857142857142857
```

0.13 Roc-AUC curve

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

```
[75]: probability
```

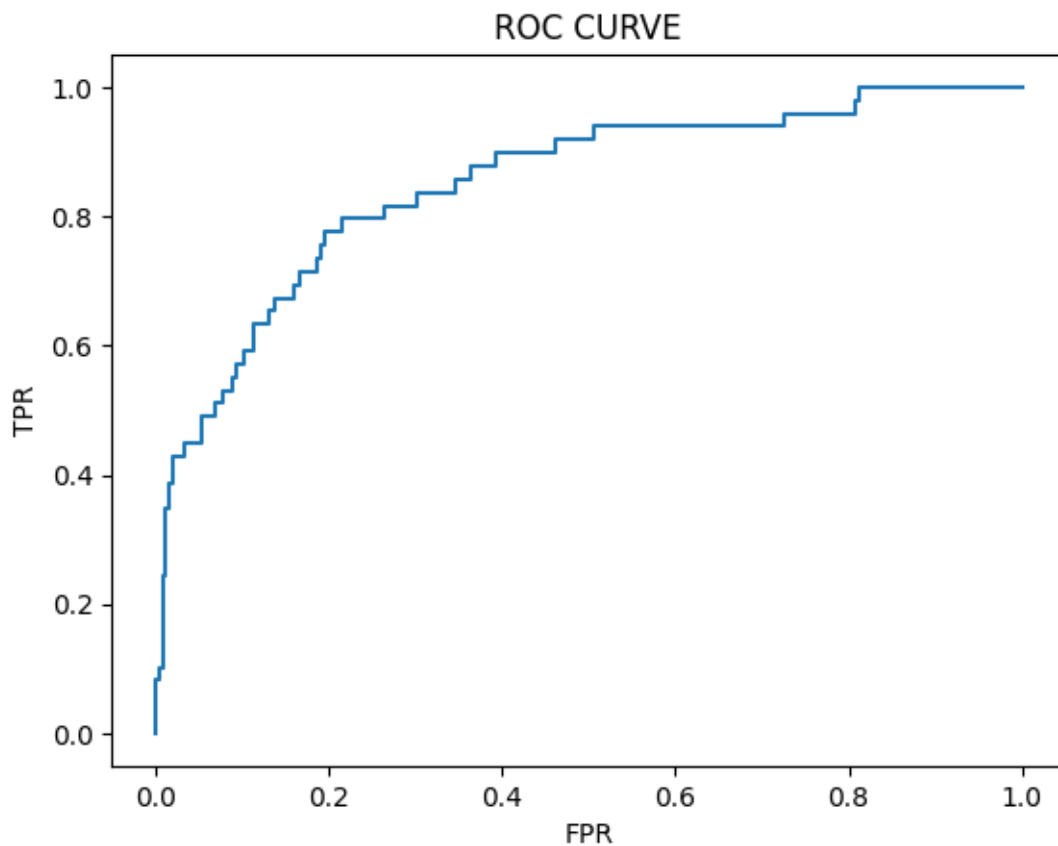
```
[75]: array([0.13634171, 0.21730734, 0.32705798, 0.07229488, 0.69580859,
         0.07469031, 0.52806558, 0.06049266, 0.00469311, 0.45731938,
         0.07113615, 0.34238116, 0.01949836, 0.65867841, 0.26438557,
         0.02266477, 0.10328484, 0.17788267, 0.04662848, 0.22767307,
         0.26499456, 0.02585468, 0.05826535, 0.04936706, 0.62425265,
         0.3811397 , 0.06897875, 0.03668439, 0.70326728, 0.04199525,
         0.01247891, 0.02948852, 0.09115453, 0.29318934, 0.07262632,
```

0.03558831, 0.09629858, 0.07244283, 0.03571409, 0.05878241,
0.07364352, 0.0188282 , 0.02720822, 0.00904886, 0.02209405,
0.48207768, 0.43271036, 0.00300348, 0.77869574, 0.49593185,
0.12067987, 0.53493629, 0.07523132, 0.26436232, 0.58508973,
0.14611472, 0.00982121, 0.19102835, 0.02813896, 0.13829777,
0.02004465, 0.21884089, 0.16133843, 0.03744237, 0.38860311,
0.01904085, 0.27086226, 0.12864115, 0.10377653, 0.11348021,
0.13713512, 0.32405164, 0.0777917 , 0.06799059, 0.1395426 ,
0.05615134, 0.04560527, 0.10899224, 0.22543052, 0.02348306,
0.01174092, 0.02267063, 0.15509414, 0.02643478, 0.02902701,
0.0816505 , 0.01319954, 0.03869732, 0.03024483, 0.14523475,
0.31122146, 0.11764803, 0.23371056, 0.16089808, 0.0166192 ,
0.16752508, 0.32427538, 0.23598049, 0.0780204 , 0.04504903,
0.2195145 , 0.62811547, 0.46526123, 0.01189559, 0.11475962,
0.02301548, 0.05056511, 0.1800559 , 0.04585039, 0.12980208,
0.08549239, 0.05339469, 0.01622171, 0.15036074, 0.06530978,
0.03110814, 0.03962745, 0.06393414, 0.01267552, 0.01047392,
0.18227658, 0.04954899, 0.06972452, 0.84682292, 0.03291138,
0.05031183, 0.00954385, 0.13951059, 0.15290101, 0.05059972,
0.02185443, 0.19879527, 0.54109252, 0.36110931, 0.07669664,
0.44763267, 0.60383617, 0.14170826, 0.06122373, 0.25945567,
0.10341778, 0.06715161, 0.09592746, 0.14972405, 0.16973424,
0.02451576, 0.18696568, 0.00539543, 0.0717976 , 0.17211153,
0.05855477, 0.17498729, 0.04982642, 0.18787672, 0.06503608,
0.02449723, 0.09214349, 0.08640433, 0.01263918, 0.01042935,
0.46140865, 0.00838725, 0.14949622, 0.8606726 , 0.09155982,
0.24144847, 0.15383096, 0.14231074, 0.02687429, 0.00839354,
0.05282875, 0.08095507, 0.06051501, 0.11531185, 0.02367573,
0.15294204, 0.09714092, 0.07719304, 0.04948916, 0.11305646,
0.02183542, 0.10174812, 0.00679619, 0.69932138, 0.04834459,
0.04394425, 0.38510719, 0.04825585, 0.71425972, 0.1026592 ,
0.40520202, 0.42541053, 0.28267166, 0.0414084 , 0.08790207,
0.14770658, 0.04593363, 0.02293972, 0.31029533, 0.04545581,
0.17147185, 0.18574883, 0.6707931 , 0.053669 , 0.25204964,
0.03577654, 0.50773946, 0.00522282, 0.13486096, 0.02498297,
0.0781506 , 0.18098033, 0.05948306, 0.10317474, 0.15384842,
0.01173852, 0.03057253, 0.08668822, 0.03399546, 0.16652657,
0.10962351, 0.25462399, 0.7566408 , 0.20403422, 0.41050669,
0.01725859, 0.06120808, 0.21191691, 0.36398358, 0.0330151 ,
0.04515306, 0.33439171, 0.05746819, 0.02120461, 0.13616467,
0.22966153, 0.28794163, 0.00501471, 0.08896455, 0.0220678 ,
0.14377617, 0.29175519, 0.02065761, 0.18801695, 0.04216821,
0.03240882, 0.36376573, 0.32954972, 0.03556164, 0.11253375,
0.35968396, 0.32355734, 0.79735174, 0.05995279, 0.21611516,
0.04121398, 0.0097675 , 0.70274038, 0.4305992 , 0.3694506 ,
0.34870072, 0.02756869, 0.16587715, 0.06083099, 0.06789623,
0.10114009, 0.0073245 , 0.2554153 , 0.31376004, 0.0570623 ,


```
0.10149719, 0.01199016, 0.16094349, 0.03458018, 0.02190261,  
0.02850696, 0.07070712, 0.29512032, 0.30422644, 0.22252621,  
0.28047327, 0.01704129, 0.16276334, 0.09071037, 0.05167575,  
0.19589614, 0.00507932, 0.24843434, 0.00439087, 0.02115045,  
0.22489408, 0.69393407, 0.05302359, 0.14080714])
```

```
[76]: # roc_curve  
fpr,tpr,threshsholds = roc_curve(y_test,probability)
```

```
[77]: plt.plot(fpr,tpr)  
plt.xlabel('FPR')  
plt.ylabel('TPR')  
plt.title('ROC CURVE')  
plt.show()
```



INFERENCE: We can say that above plotted graph is the Roc Auc curve which gives the plot between the FPR and TPR.

0.14 Decision Tree Classifier(Model Building)

```
[78]: from sklearn.tree import DecisionTreeClassifier
```

```
[79]: dtc=DecisionTreeClassifier()
```

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

```
[80]: DecisionTreeClassifier()
```

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

```
[82]: pred
```

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

```
[83]: y_test
```

```
[83]: 442      0
      1091    0
      981     1
      785     0
      1332    1
      ..
     1439     0
      481     0
      124     1
      198     0
     1229     0
      Name: Attrition, Length: 294, dtype: int64
```

0.15 Evaluation of classification model

```
[84]: from sklearn.metrics import  
      accuracy_score, classification_report, confusion_matrix, roc_auc_score, roc_curve, precision_score
```

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

```
[85]: 0.7721088435374149
```

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

```
[86]: array([[209,  36],  
         [ 31,  18]])
```

```
[87]: precision_score(y_test, pred)
```

```
[87]: 0.3333333333333333
```

```
[88]: recall_score(y_test, pred)
```

```
[88]: 0.3673469387755102
```

```
[89]: f1_score(y_test, pred)
```

```
[89]: 0.34951456310679613
```

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

	precision	recall	f1-score	support
0	0.87	0.85	0.86	245
1	0.33	0.37	0.35	49
accuracy			0.77	294
macro avg	0.60	0.61	0.61	294
weighted avg	0.78	0.77	0.78	294

0.16 Roc Auc curve

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

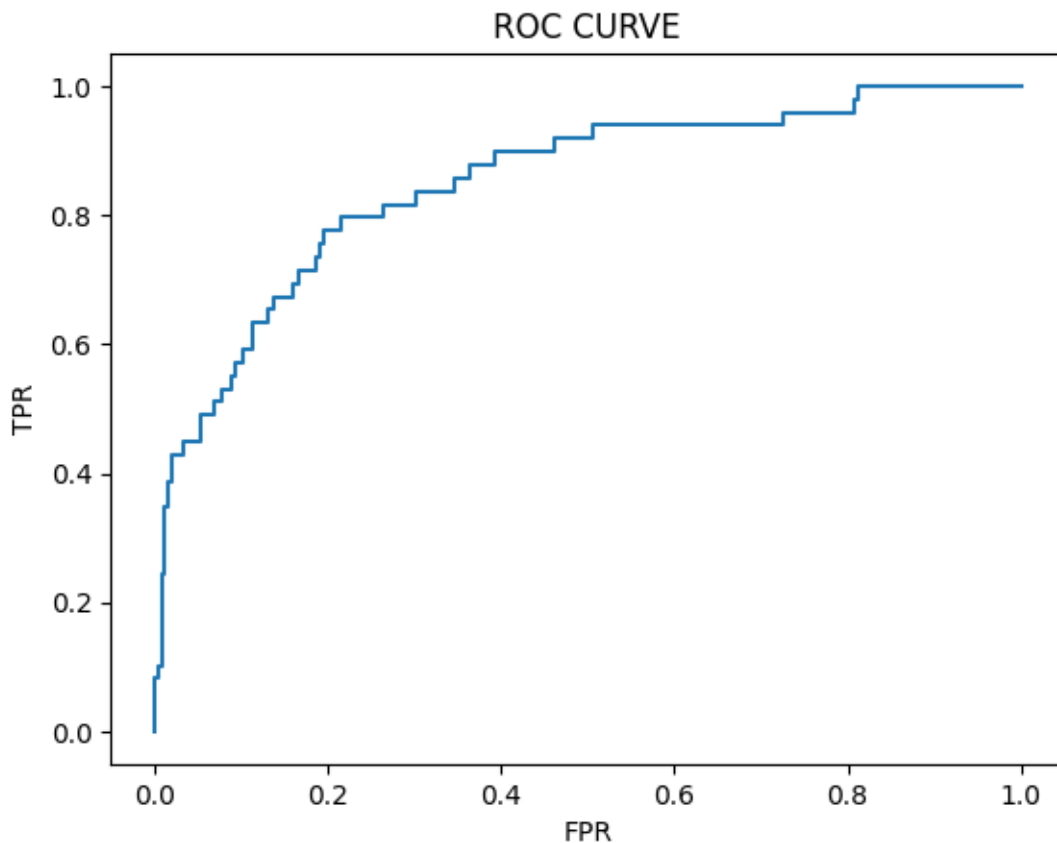
```
[91]: array([0.13634171, 0.21730734, 0.32705798, 0.07229488, 0.69580859,  
          0.07469031, 0.52806558, 0.06049266, 0.00469311, 0.45731938,  
          0.07113615, 0.34238116, 0.01949836, 0.65867841, 0.26438557,  
          0.02266477, 0.10328484, 0.17788267, 0.04662848, 0.22767307,  
          0.26499456, 0.02585468, 0.05826535, 0.04936706, 0.62425265,
```

0.3811397 , 0.06897875, 0.03668439, 0.70326728, 0.04199525,
 0.01247891, 0.02948852, 0.09115453, 0.29318934, 0.07262632,
 0.03558831, 0.09629858, 0.07244283, 0.03571409, 0.05878241,
 0.07364352, 0.0188282 , 0.02720822, 0.00904886, 0.02209405,
 0.48207768, 0.43271036, 0.00300348, 0.77869574, 0.49593185,
 0.12067987, 0.53493629, 0.07523132, 0.26436232, 0.58508973,
 0.14611472, 0.00982121, 0.19102835, 0.02813896, 0.13829777,
 0.02004465, 0.21884089, 0.16133843, 0.03744237, 0.38860311,
 0.01904085, 0.27086226, 0.12864115, 0.10377653, 0.11348021,
 0.13713512, 0.32405164, 0.0777917 , 0.06799059, 0.1395426 ,
 0.05615134, 0.04560527, 0.10899224, 0.22543052, 0.02348306,
 0.01174092, 0.02267063, 0.15509414, 0.02643478, 0.02902701,
 0.0816505 , 0.01319954, 0.03869732, 0.03024483, 0.14523475,
 0.31122146, 0.11764803, 0.23371056, 0.16089808, 0.0166192 ,
 0.16752508, 0.32427538, 0.23598049, 0.0780204 , 0.04504903,
 0.2195145 , 0.62811547, 0.46526123, 0.01189559, 0.11475962,
 0.02301548, 0.05056511, 0.1800559 , 0.04585039, 0.12980208,
 0.08549239, 0.05339469, 0.01622171, 0.15036074, 0.06530978,
 0.03110814, 0.03962745, 0.06393414, 0.01267552, 0.01047392,
 0.18227658, 0.04954899, 0.06972452, 0.84682292, 0.03291138,
 0.05031183, 0.00954385, 0.13951059, 0.15290101, 0.05059972,
 0.02185443, 0.19879527, 0.54109252, 0.36110931, 0.07669664,
 0.44763267, 0.60383617, 0.14170826, 0.06122373, 0.25945567,
 0.10341778, 0.06715161, 0.09592746, 0.14972405, 0.16973424,
 0.02451576, 0.18696568, 0.00539543, 0.0717976 , 0.17211153,
 0.05855477, 0.17498729, 0.04982642, 0.18787672, 0.06503608,
 0.02449723, 0.09214349, 0.08640433, 0.01263918, 0.01042935,
 0.46140865, 0.00838725, 0.14949622, 0.8606726 , 0.09155982,
 0.24144847, 0.15383096, 0.14231074, 0.02687429, 0.00839354,
 0.05282875, 0.08095507, 0.06051501, 0.11531185, 0.02367573,
 0.15294204, 0.09714092, 0.07719304, 0.04948916, 0.11305646,
 0.02183542, 0.10174812, 0.00679619, 0.69932138, 0.04834459,
 0.04394425, 0.38510719, 0.04825585, 0.71425972, 0.1026592 ,
 0.40520202, 0.42541053, 0.28267166, 0.0414084 , 0.08790207,
 0.14770658, 0.04593363, 0.02293972, 0.31029533, 0.04545581,
 0.17147185, 0.18574883, 0.6707931 , 0.053669 , 0.25204964,
 0.03577654, 0.50773946, 0.00522282, 0.13486096, 0.02498297,
 0.0781506 , 0.18098033, 0.05948306, 0.10317474, 0.15384842,
 0.01173852, 0.03057253, 0.08668822, 0.03399546, 0.16652657,
 0.10962351, 0.25462399, 0.7566408 , 0.20403422, 0.41050669,
 0.01725859, 0.06120808, 0.21191691, 0.36398358, 0.0330151 ,
 0.04515306, 0.33439171, 0.05746819, 0.02120461, 0.13616467,
 0.22966153, 0.28794163, 0.00501471, 0.08896455, 0.0220678 ,
 0.14377617, 0.29175519, 0.02065761, 0.18801695, 0.04216821,
 0.03240882, 0.36376573, 0.32954972, 0.03556164, 0.11253375,
 0.35968396, 0.32355734, 0.79735174, 0.05995279, 0.21611516,
 0.04121398, 0.0097675 , 0.70274038, 0.4305992 , 0.3694506 ,

```
0.34870072, 0.02756869, 0.16587715, 0.06083099, 0.06789623,  
0.10114009, 0.0073245 , 0.2554153 , 0.31376004, 0.0570623 ,  
0.10149719, 0.01199016, 0.16094349, 0.03458018, 0.02190261,  
0.02850696, 0.07070712, 0.29512032, 0.30422644, 0.22252621,  
0.28047327, 0.01704129, 0.16276334, 0.09071037, 0.05167575,  
0.19589614, 0.00507932, 0.24843434, 0.00439087, 0.02115045,  
0.22489408, 0.69393407, 0.05302359, 0.14080714])
```

```
[92]: fpr, tpr, threshholds = roc_curve(y_test, probability)
```

```
[93]: plt.plot(fpr, tpr)  
plt.xlabel('FPR')  
plt.ylabel('TPR')  
plt.title('ROC CURVE')  
plt.show()
```



INFERENCE: We can say that above plotted graph is the Roc Auc curve which gives the plot between the FPR and TPR.

0.17 Hyper Parameter Tuning

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

```
[94]: [Text(0.32811104910714284, 0.9722222222222222, 'x[26] <= 0.079\ngini =
0.269\nsamples = 1176\nvalue = [988, 188]'),
Text(0.08571428571428572, 0.9166666666666666, 'x[15] <= 0.75\ngini =
0.5\nsamples = 78\nvalue = [39, 39]'),
Text(0.05357142857142857, 0.8611111111111112, 'x[3] <= 0.554\ngini =
0.426\nsamples = 39\nvalue = [27, 12]'),
Text(0.03571428571428571, 0.8055555555555556, 'x[14] <= 0.167\ngini =
0.312\nsamples = 31\nvalue = [25, 6]'),
Text(0.02142857142857143, 0.75, 'x[20] <= 0.5\ngini = 0.49\nsamples = 7\nvalue
= [3, 4]'),
Text(0.014285714285714285, 0.6944444444444444, 'x[16] <= 0.056\ngini =
0.375\nsamples = 4\nvalue = [3, 1]'),
Text(0.007142857142857143, 0.6388888888888888, 'gini = 0.0\nsamples = 1\nvalue
= [0, 1]'),
Text(0.02142857142857143, 0.6388888888888888, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
Text(0.02857142857142857, 0.6944444444444444, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
Text(0.05, 0.75, 'x[18] <= 0.056\ngini = 0.153\nsamples = 24\nvalue = [22,
2]'),
Text(0.04285714285714286, 0.6944444444444444, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.05714285714285714, 0.6944444444444444, 'x[8] <= 0.167\ngini =
0.083\nsamples = 23\nvalue = [22, 1]'),
Text(0.05, 0.6388888888888888, 'x[5] <= 0.4\ngini = 0.5\nsamples = 2\nvalue =
[1, 1]'),
Text(0.04285714285714286, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.05714285714285714, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.06428571428571428, 0.6388888888888888, 'gini = 0.0\nsamples = 21\nvalue
= [21, 0]'),
Text(0.07142857142857142, 0.8055555555555556, 'x[21] <= 0.679\ngini =
0.375\nsamples = 8\nvalue = [2, 6]'),
Text(0.06428571428571428, 0.75, 'gini = 0.0\nsamples = 6\nvalue = [0, 6]'),
Text(0.07857142857142857, 0.75, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.11785714285714285, 0.8611111111111112, 'x[10] <= 0.364\ngini =
0.426\nsamples = 39\nvalue = [12, 27]'),
Text(0.1, 0.8055555555555556, 'x[28] <= 0.167\ngini = 0.133\nsamples =
14\nvalue = [1, 13]'),
Text(0.09285714285714286, 0.75, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
```

```

Text(0.10714285714285714, 0.75, 'gini = 0.0\nsamples = 13\nvalue = [0, 13]'),
Text(0.1357142857142857, 0.8055555555555556, 'x[7] <= 0.105\ngini =
0.493\nsamples = 25\nvalue = [11, 14]'),
Text(0.12142857142857143, 0.75, 'x[21] <= 0.464\ngini = 0.278\nsamples =
6\nvalue = [5, 1]'),
Text(0.11428571428571428, 0.6944444444444444, 'gini = 0.0\nsamples = 5\nvalue =
[5, 0]'),
Text(0.12857142857142856, 0.6944444444444444, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.15, 0.75, 'x[14] <= 0.5\ngini = 0.432\nsamples = 19\nvalue = [6, 13]'),
Text(0.14285714285714285, 0.6944444444444444, 'gini = 0.0\nsamples = 7\nvalue =
[0, 7]'),
Text(0.15714285714285714, 0.6944444444444444, 'x[5] <= 0.4\ngini = 0.5\nsamples
= 12\nvalue = [6, 6]'),
Text(0.14285714285714285, 0.6388888888888888, 'x[31] <= 0.033\ngini =
0.278\nsamples = 6\nvalue = [5, 1]'),
Text(0.1357142857142857, 0.5833333333333334, 'gini = 0.0\nsamples = 5\nvalue =
[5, 0]'),
Text(0.15, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.17142857142857143, 0.6388888888888888, 'x[7] <= 0.249\ngini =
0.278\nsamples = 6\nvalue = [1, 5]'),
Text(0.16428571428571428, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.17857142857142858, 0.5833333333333334, 'gini = 0.0\nsamples = 5\nvalue =
[0, 5]'),
Text(0.5705078125, 0.9166666666666666, 'x[20] <= 0.5\ngini = 0.235\nsamples =
1098\nvalue = [949, 149]'),
Text(0.32717633928571427, 0.8611111111111112, 'x[28] <= 0.167\ngini =
0.162\nsamples = 798\nvalue = [727, 71]'),
Text(0.19285714285714287, 0.8055555555555556, 'x[7] <= 0.445\ngini =
0.38\nsamples = 47\nvalue = [35, 12]'),
Text(0.17857142857142858, 0.75, 'x[15] <= 0.75\ngini = 0.1\nsamples = 19\nvalue
= [18, 1]'),
Text(0.17142857142857143, 0.6944444444444444, 'gini = 0.0\nsamples = 18\nvalue
= [18, 0]'),
Text(0.18571428571428572, 0.6944444444444444, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.20714285714285716, 0.75, 'x[16] <= 0.094\ngini = 0.477\nsamples =
28\nvalue = [17, 11]'),
Text(0.2, 0.6944444444444444, 'gini = 0.0\nsamples = 4\nvalue = [0, 4]'),
Text(0.21428571428571427, 0.6944444444444444, 'x[31] <= 0.6\ngini =
0.413\nsamples = 24\nvalue = [17, 7]'),
Text(0.20714285714285716, 0.6388888888888888, 'x[10] <= 0.486\ngini =
0.351\nsamples = 22\nvalue = [17, 5]'),
Text(0.2, 0.5833333333333334, 'x[23] <= 0.5\ngini = 0.496\nsamples = 11\nvalue
= [6, 5]'),
Text(0.19285714285714287, 0.5277777777777778, 'x[16] <= 0.417\ngini =

```

```

0.408\nsamples = 7\nvalue = [2, 5]'),
Text(0.18571428571428572, 0.4722222222222222, 'gini = 0.0\nsamples = 5\nvalue =
[0, 5]'),
Text(0.2, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.20714285714285716, 0.5277777777777778, 'gini = 0.0\nsamples = 4\nvalue =
[4, 0]'),
Text(0.21428571428571427, 0.5833333333333334, 'gini = 0.0\nsamples = 11\nvalue
= [11, 0]'),
Text(0.22142857142857142, 0.6388888888888888, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
Text(0.4614955357142857, 0.8055555555555556, 'x[29] <= 0.963\ngini =
0.145\nsamples = 751\nvalue = [692, 59]'),
Text(0.4543526785714286, 0.75, 'x[29] <= 0.113\ngini = 0.143\nsamples =
750\nvalue = [692, 58]'),
Text(0.32276785714285716, 0.6944444444444444, 'x[8] <= 0.167\ngini =
0.218\nsamples = 257\nvalue = [225, 32]'),
Text(0.2767857142857143, 0.6388888888888888, 'x[32] <= 0.147\ngini =
0.355\nsamples = 65\nvalue = [50, 15]'),
Text(0.25357142857142856, 0.5833333333333334, 'x[32] <= 0.029\ngini =
0.303\nsamples = 59\nvalue = [48, 11]'),
Text(0.22857142857142856, 0.5277777777777778, 'x[11] <= 0.5\ngini =
0.463\nsamples = 22\nvalue = [14, 8]'),
Text(0.21428571428571427, 0.4722222222222222, 'x[10] <= 0.179\ngini =
0.198\nsamples = 9\nvalue = [8, 1]'),
Text(0.20714285714285716, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.22142857142857142, 0.4166666666666667, 'gini = 0.0\nsamples = 8\nvalue =
[8, 0]'),
Text(0.24285714285714285, 0.4722222222222222, 'x[10] <= 0.4\ngini =
0.497\nsamples = 13\nvalue = [6, 7]'),
Text(0.2357142857142857, 0.4166666666666667, 'gini = 0.0\nsamples = 4\nvalue =
[4, 0]'),
Text(0.25, 0.4166666666666667, 'x[3] <= 0.286\ngini = 0.346\nsamples = 9\nvalue
= [2, 7]'),
Text(0.24285714285714285, 0.3611111111111111, 'x[1] <= 0.369\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
Text(0.2357142857142857, 0.3055555555555556, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
Text(0.25, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.2571428571428571, 0.3611111111111111, 'gini = 0.0\nsamples = 6\nvalue =
[0, 6]'),
Text(0.2785714285714286, 0.5277777777777778, 'x[14] <= 0.167\ngini =
0.149\nsamples = 37\nvalue = [34, 3]'),
Text(0.2714285714285714, 0.4722222222222222, 'x[29] <= 0.088\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
Text(0.2642857142857143, 0.4166666666666667, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),

```



```

Text(0.2785714285714286, 0.4166666666666667, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
Text(0.2857142857142857, 0.4722222222222222, 'gini = 0.0\nsamples = 31\nvalue =
[31, 0]'),
Text(0.3, 0.5833333333333334, 'x[7] <= 0.065\ngini = 0.444\nsamples = 6\nvalue
= [2, 4]'),
Text(0.29285714285714287, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
Text(0.30714285714285716, 0.5277777777777778, 'gini = 0.0\nsamples = 4\nvalue =
[0, 4]'),
Text(0.36875, 0.6388888888888888, 'x[33] <= 0.321\ngini = 0.161\nsamples =
192\nvalue = [175, 17]'),
Text(0.32857142857142857, 0.5833333333333334, 'x[5] <= 0.1\ngini =
0.294\nsamples = 67\nvalue = [55, 12]'),
Text(0.32142857142857145, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
Text(0.3357142857142857, 0.5277777777777778, 'x[28] <= 0.5\ngini =
0.26\nsamples = 65\nvalue = [55, 10]'),
Text(0.3107142857142857, 0.4722222222222222, 'x[5] <= 0.5\ngini =
0.469\nsamples = 16\nvalue = [10, 6]'),
Text(0.30357142857142855, 0.4166666666666667, 'gini = 0.0\nsamples = 7\nvalue =
[7, 0]'),
Text(0.31785714285714284, 0.4166666666666667, 'x[8] <= 0.833\ngini =
0.444\nsamples = 9\nvalue = [3, 6]'),
Text(0.3107142857142857, 0.3611111111111111, 'gini = 0.0\nsamples = 5\nvalue =
[0, 5]'),
Text(0.325, 0.3611111111111111, 'x[10] <= 0.129\ngini = 0.375\nsamples =
4\nvalue = [3, 1]'),
Text(0.31785714285714284, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.33214285714285713, 0.3055555555555556, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
Text(0.3607142857142857, 0.4722222222222222, 'x[1] <= 0.037\ngini =
0.15\nsamples = 49\nvalue = [45, 4]'),
Text(0.3535714285714286, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.3678571428571429, 0.4166666666666667, 'x[1] <= 0.938\ngini =
0.117\nsamples = 48\nvalue = [45, 3]'),
Text(0.3607142857142857, 0.3611111111111111, 'x[4] <= 0.875\ngini =
0.081\nsamples = 47\nvalue = [45, 2]'),
Text(0.3464285714285714, 0.3055555555555556, 'x[11] <= 0.167\ngini =
0.043\nsamples = 45\nvalue = [44, 1]'),
Text(0.3392857142857143, 0.25, 'x[9] <= 0.5\ngini = 0.444\nsamples = 3\nvalue =
[2, 1]'),
Text(0.33214285714285713, 0.19444444444444445, 'gini = 0.0\nsamples = 1\nvalue
= [0, 1]'),
Text(0.3464285714285714, 0.19444444444444445, 'gini = 0.0\nsamples = 2\nvalue =

```

```

[2, 0]'),
Text(0.3535714285714286, 0.25, 'gini = 0.0\nsamples = 42\nvalue = [42, 0]'),
Text(0.375, 0.3055555555555556, 'x[16] <= 0.119\ngini = 0.5\nsamples = 2\nvalue
= [1, 1]'),
Text(0.3678571428571429, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.3821428571428571, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.375, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.4089285714285714, 0.5833333333333334, 'x[7] <= 0.022\ngini =
0.077\nsamples = 125\nvalue = [120, 5]'),
Text(0.3892857142857143, 0.5277777777777778, 'x[13] <= 0.5\ngini = 0.5\nsamples
= 4\nvalue = [2, 2]'),
Text(0.3821428571428571, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
Text(0.3964285714285714, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
Text(0.42857142857142855, 0.5277777777777778, 'x[17] <= 0.968\ngini =
0.048\nsamples = 121\nvalue = [118, 3]'),
Text(0.4107142857142857, 0.4722222222222222, 'x[1] <= 0.98\ngini =
0.033\nsamples = 118\nvalue = [116, 2]'),
Text(0.3964285714285714, 0.4166666666666667, 'x[13] <= 0.938\ngini =
0.017\nsamples = 114\nvalue = [113, 1]'),
Text(0.3892857142857143, 0.3611111111111111, 'gini = 0.0\nsamples = 107\nvalue
= [107, 0]'),
Text(0.4035714285714286, 0.3611111111111111, 'x[11] <= 0.167\ngini =
0.245\nsamples = 7\nvalue = [6, 1]'),
Text(0.3964285714285714, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.4107142857142857, 0.3055555555555556, 'gini = 0.0\nsamples = 6\nvalue =
[6, 0]'),
Text(0.425, 0.4166666666666667, 'x[29] <= 0.088\ngini = 0.375\nsamples =
4\nvalue = [3, 1]'),
Text(0.41785714285714287, 0.3611111111111111, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
Text(0.43214285714285716, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.44642857142857145, 0.4722222222222222, 'x[21] <= 0.714\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
Text(0.4392857142857143, 0.4166666666666667, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
Text(0.45357142857142857, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.5859375, 0.6944444444444444, 'x[29] <= 0.787\ngini = 0.1\nsamples =
493\nvalue = [467, 26]'),
Text(0.5504464285714286, 0.6388888888888888, 'x[14] <= 0.5\ngini =
0.094\nsamples = 486\nvalue = [462, 24]'),
Text(0.5008928571428571, 0.5833333333333334, 'x[13] <= 0.938\ngini =
0.154\nsamples = 191\nvalue = [175, 16]'),

```

```

Text(0.49375, 0.5277777777777778, 'x[17] <= 0.481\ngini = 0.145\nsamples =
190\nvalue = [175, 15]'),
Text(0.475, 0.4722222222222222, 'x[17] <= 0.47\ngini = 0.221\nsamples =
95\nvalue = [83, 12]'),
Text(0.46785714285714286, 0.4166666666666667, 'x[32] <= 0.794\ngini =
0.207\nsamples = 94\nvalue = [83, 11]'),
Text(0.4607142857142857, 0.3611111111111111, 'x[4] <= 0.375\ngini =
0.192\nsamples = 93\nvalue = [83, 10]'),
Text(0.4375, 0.3055555555555556, 'x[5] <= 0.9\ngini = 0.363\nsamples =
21\nvalue = [16, 5]'),
Text(0.4303571428571429, 0.25, 'x[16] <= 0.413\ngini = 0.266\nsamples =
19\nvalue = [16, 3]'),
Text(0.4160714285714286, 0.19444444444444445, 'x[18] <= 0.056\ngini =
0.117\nsamples = 16\nvalue = [15, 1]'),
Text(0.4089285714285714, 0.1388888888888889, 'x[0] <= 0.75\ngini = 0.5\nsamples
= 2\nvalue = [1, 1]'),
Text(0.4017857142857143, 0.08333333333333333, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.4160714285714286, 0.08333333333333333, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.4232142857142857, 0.1388888888888889, 'gini = 0.0\nsamples = 14\nvalue =
[14, 0]'),
Text(0.4446428571428571, 0.19444444444444445, 'x[21] <= 0.893\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
Text(0.4375, 0.1388888888888889, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.4517857142857143, 0.1388888888888889, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.4446428571428571, 0.25, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.48392857142857143, 0.3055555555555556, 'x[30] <= 0.139\ngini =
0.129\nsamples = 72\nvalue = [67, 5]'),
Text(0.4660714285714286, 0.25, 'x[7] <= 0.68\ngini = 0.444\nsamples = 6\nvalue
= [4, 2]'),
Text(0.4589285714285714, 0.19444444444444445, 'gini = 0.0\nsamples = 4\nvalue =
[4, 0]'),
Text(0.4732142857142857, 0.19444444444444445, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
Text(0.5017857142857143, 0.25, 'x[10] <= 0.993\ngini = 0.087\nsamples =
66\nvalue = [63, 3]'),
Text(0.4875, 0.19444444444444445, 'x[27] <= 0.583\ngini = 0.061\nsamples =
64\nvalue = [62, 2]'),
Text(0.48035714285714287, 0.1388888888888889, 'gini = 0.0\nsamples = 51\nvalue
= [51, 0]'),
Text(0.49464285714285716, 0.1388888888888889, 'x[13] <= 0.812\ngini =
0.26\nsamples = 13\nvalue = [11, 2]'),
Text(0.4875, 0.08333333333333333, 'gini = 0.0\nsamples = 9\nvalue = [9, 0]'),
Text(0.5017857142857143, 0.08333333333333333, 'x[1] <= 0.525\ngini =
0.5\nsamples = 4\nvalue = [2, 2]'),

```

```

Text(0.49464285714285716, 0.027777777777777776, 'gini = 0.0\nsamples = 2\nvalue
= [0, 2]'),
Text(0.5089285714285714, 0.027777777777777776, 'gini = 0.0\nsamples = 2\nvalue
= [2, 0]'),
Text(0.5160714285714286, 0.19444444444444445, 'x[3] <= 0.018\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
Text(0.5089285714285714, 0.13888888888888889, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.5232142857142857, 0.13888888888888889, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.475, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.48214285714285715, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.5125, 0.4722222222222222, 'x[18] <= 0.5\ngini = 0.061\nsamples =
95\nvalue = [92, 3]'),
Text(0.5053571428571428, 0.4166666666666667, 'gini = 0.0\nsamples = 76\nvalue =
[76, 0]'),
Text(0.5196428571428572, 0.4166666666666667, 'x[32] <= 0.088\ngini =
0.266\nsamples = 19\nvalue = [16, 3]'),
Text(0.5053571428571428, 0.3611111111111111, 'x[5] <= 0.3\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
Text(0.4982142857142857, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.5125, 0.3055555555555556, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.5339285714285714, 0.3611111111111111, 'x[16] <= 0.108\ngini =
0.117\nsamples = 16\nvalue = [15, 1]'),
Text(0.5267857142857143, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.5410714285714285, 0.3055555555555556, 'gini = 0.0\nsamples = 15\nvalue =
[15, 0]'),
Text(0.5080357142857143, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.6, 0.5833333333333334, 'x[21] <= 0.036\ngini = 0.053\nsamples =
295\nvalue = [287, 8]'),
Text(0.5767857142857142, 0.5277777777777778, 'x[31] <= 0.7\ngini =
0.159\nsamples = 46\nvalue = [42, 4]'),
Text(0.5696428571428571, 0.4722222222222222, 'x[11] <= 0.167\ngini =
0.124\nsamples = 45\nvalue = [42, 3]'),
Text(0.5553571428571429, 0.4166666666666667, 'x[18] <= 0.056\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
Text(0.5482142857142858, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.5625, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.5839285714285715, 0.4166666666666667, 'x[33] <= 0.595\ngini =
0.089\nsamples = 43\nvalue = [41, 2]'),
Text(0.5767857142857142, 0.3611111111111111, 'gini = 0.0\nsamples = 33\nvalue =
[33, 0]'),

```

```

Text(0.5910714285714286, 0.3611111111111111, 'x[8] <= 0.333\ngini =
0.32\nsamples = 10\nvalue = [8, 2]'),
Text(0.5839285714285715, 0.3055555555555556, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
Text(0.5982142857142857, 0.3055555555555556, 'gini = 0.0\nsamples = 8\nvalue =
[8, 0]'),
Text(0.5839285714285715, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.6232142857142857, 0.5277777777777778, 'x[16] <= 0.056\ngini =
0.032\nsamples = 249\nvalue = [245, 4]'),
Text(0.6053571428571428, 0.4722222222222222, 'x[18] <= 0.056\ngini =
0.32\nsamples = 5\nvalue = [4, 1]'),
Text(0.5982142857142857, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.6125, 0.4166666666666667, 'gini = 0.0\nsamples = 4\nvalue = [4, 0]'),
Text(0.6410714285714286, 0.4722222222222222, 'x[1] <= 0.015\ngini =
0.024\nsamples = 244\nvalue = [241, 3]'),
Text(0.6267857142857143, 0.4166666666666667, 'x[17] <= 0.715\ngini =
0.278\nsamples = 6\nvalue = [5, 1]'),
Text(0.6196428571428572, 0.3611111111111111, 'gini = 0.0\nsamples = 5\nvalue =
[5, 0]'),
Text(0.6339285714285714, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.6553571428571429, 0.4166666666666667, 'x[23] <= 0.167\ngini =
0.017\nsamples = 238\nvalue = [236, 2]'),
Text(0.6482142857142857, 0.3611111111111111, 'x[28] <= 0.833\ngini =
0.073\nsamples = 53\nvalue = [51, 2]'),
Text(0.6339285714285714, 0.3055555555555556, 'x[32] <= 0.088\ngini =
0.041\nsamples = 48\nvalue = [47, 1]'),
Text(0.6267857142857143, 0.25, 'x[13] <= 0.312\ngini = 0.245\nsamples =
7\nvalue = [6, 1]'),
Text(0.6196428571428572, 0.19444444444444445, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.6339285714285714, 0.19444444444444445, 'gini = 0.0\nsamples = 6\nvalue =
[6, 0]'),
Text(0.6410714285714286, 0.25, 'gini = 0.0\nsamples = 41\nvalue = [41, 0]'),
Text(0.6625, 0.3055555555555556, 'x[31] <= 0.367\ngini = 0.32\nsamples =
5\nvalue = [4, 1]'),
Text(0.6553571428571429, 0.25, 'gini = 0.0\nsamples = 4\nvalue = [4, 0]'),
Text(0.6696428571428571, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6625, 0.3611111111111111, 'gini = 0.0\nsamples = 185\nvalue = [185,
0]'),
Text(0.6214285714285714, 0.6388888888888888, 'x[9] <= 0.5\ngini =
0.408\nsamples = 7\nvalue = [5, 2]'),
Text(0.6142857142857143, 0.5833333333333334, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
Text(0.6285714285714286, 0.5833333333333334, 'gini = 0.0\nsamples = 5\nvalue =

```

```

[5, 0]'),
Text(0.46863839285714287, 0.75, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.8138392857142858, 0.8611111111111112, 'x[16] <= 0.157\ngini =
0.385\nsamples = 300\nvalue = [222, 78]'),
Text(0.7232142857142857, 0.8055555555555556, 'x[25] <= 0.167\ngini =
0.5\nsamples = 96\nvalue = [49, 47]'),
Text(0.6892857142857143, 0.75, 'x[3] <= 0.161\ngini = 0.459\nsamples =
42\nvalue = [15, 27]'),
Text(0.6642857142857143, 0.6944444444444444, 'x[7] <= 0.415\ngini =
0.499\nsamples = 23\nvalue = [12, 11]'),
Text(0.65, 0.6388888888888888, 'x[17] <= 0.561\ngini = 0.355\nsamples =
13\nvalue = [3, 10]'),
Text(0.6428571428571429, 0.5833333333333334, 'gini = 0.0\nsamples = 8\nvalue =
[0, 8]'),
Text(0.6571428571428571, 0.5833333333333334, 'x[16] <= 0.108\ngini =
0.48\nsamples = 5\nvalue = [3, 2]'),
Text(0.65, 0.5277777777777778, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
Text(0.6642857142857143, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
Text(0.6785714285714286, 0.6388888888888888, 'x[28] <= 0.833\ngini =
0.18\nsamples = 10\nvalue = [9, 1]'),
Text(0.6714285714285714, 0.5833333333333334, 'gini = 0.0\nsamples = 8\nvalue =
[8, 0]'),
Text(0.6857142857142857, 0.5833333333333334, 'x[10] <= 0.457\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
Text(0.6785714285714286, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.6928571428571428, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.7142857142857143, 0.6944444444444444, 'x[12] <= 0.125\ngini =
0.266\nsamples = 19\nvalue = [3, 16]'),
Text(0.7071428571428572, 0.6388888888888888, 'x[10] <= 0.2\ngini =
0.198\nsamples = 18\nvalue = [2, 16]'),
Text(0.7, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.7142857142857143, 0.5833333333333334, 'x[31] <= 0.433\ngini =
0.111\nsamples = 17\nvalue = [1, 16]'),
Text(0.7071428571428572, 0.5277777777777778, 'gini = 0.0\nsamples = 15\nvalue =
[0, 15]'),
Text(0.7214285714285714, 0.5277777777777778, 'x[15] <= 0.75\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
Text(0.7142857142857143, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.7285714285714285, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
Text(0.7214285714285714, 0.6388888888888888, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.7571428571428571, 0.75, 'x[33] <= 0.202\ngini = 0.466\nsamples =

```

```

54\nvalue = [34, 20]'),
  Text(0.7428571428571429, 0.6944444444444444, 'x[33] <= 0.107\ngini =
0.245\nsamples = 7\nvalue = [1, 6]'),
  Text(0.7357142857142858, 0.6388888888888888, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
  Text(0.75, 0.6388888888888888, 'gini = 0.0\nsamples = 6\nvalue = [0, 6]'),
  Text(0.7714285714285715, 0.6944444444444444, 'x[26] <= 0.921\ngini =
0.418\nsamples = 47\nvalue = [33, 14]'),
  Text(0.7642857142857142, 0.6388888888888888, 'x[1] <= 0.523\ngini =
0.375\nsamples = 44\nvalue = [33, 11]'),
  Text(0.75, 0.5833333333333334, 'x[16] <= 0.079\ngini = 0.48\nsamples =
25\nvalue = [15, 10]'),
  Text(0.7428571428571429, 0.5277777777777778, 'gini = 0.0\nsamples = 5\nvalue =
[0, 5]'),
  Text(0.7571428571428571, 0.5277777777777778, 'x[1] <= 0.145\ngini =
0.375\nsamples = 20\nvalue = [15, 5]'),
  Text(0.75, 0.4722222222222222, 'gini = 0.0\nsamples = 9\nvalue = [9, 0]'),
  Text(0.7642857142857142, 0.4722222222222222, 'x[16] <= 0.122\ngini =
0.496\nsamples = 11\nvalue = [6, 5]'),
  Text(0.7571428571428571, 0.4166666666666667, 'x[18] <= 0.556\ngini =
0.245\nsamples = 7\nvalue = [6, 1]'),
  Text(0.75, 0.3611111111111111, 'gini = 0.0\nsamples = 6\nvalue = [6, 0]'),
  Text(0.7642857142857142, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
  Text(0.7714285714285715, 0.4166666666666667, 'gini = 0.0\nsamples = 4\nvalue =
[0, 4]'),
  Text(0.7785714285714286, 0.5833333333333334, 'x[10] <= 0.764\ngini =
0.1\nsamples = 19\nvalue = [18, 1]'),
  Text(0.7714285714285715, 0.5277777777777778, 'gini = 0.0\nsamples = 17\nvalue =
[17, 0]'),
  Text(0.7857142857142857, 0.5277777777777778, 'x[17] <= 0.538\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
  Text(0.7785714285714286, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
  Text(0.7928571428571428, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
  Text(0.7785714285714286, 0.6388888888888888, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
  Text(0.9044642857142857, 0.8055555555555556, 'x[15] <= 0.75\ngini =
0.258\nsamples = 204\nvalue = [173, 31]'),
  Text(0.8553571428571428, 0.75, 'x[16] <= 0.992\ngini = 0.138\nsamples =
147\nvalue = [136, 11]'),
  Text(0.8482142857142857, 0.6944444444444444, 'x[3] <= 0.482\ngini =
0.128\nsamples = 146\nvalue = [136, 10]'),
  Text(0.8285714285714286, 0.6388888888888888, 'x[29] <= 0.063\ngini =
0.038\nsamples = 104\nvalue = [102, 2]'),
  Text(0.8214285714285714, 0.5833333333333334, 'x[10] <= 0.193\ngini =

```

```

0.32\nsamples = 10\nvalue = [8, 2]'),
  Text(0.8142857142857143, 0.5277777777777778, 'x[10] <= 0.079\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
  Text(0.8071428571428572, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
  Text(0.8214285714285714, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
  Text(0.8285714285714286, 0.5277777777777778, 'gini = 0.0\nsamples = 7\nvalue =
[7, 0]'),
  Text(0.8357142857142857, 0.5833333333333334, 'gini = 0.0\nsamples = 94\nvalue =
[94, 0]'),
  Text(0.8678571428571429, 0.6388888888888888, 'x[8] <= 0.167\ngini =
0.308\nsamples = 42\nvalue = [34, 8]'),
  Text(0.85, 0.5833333333333334, 'x[1] <= 0.736\ngini = 0.375\nsamples = 4\nvalue
= [1, 3]'),
  Text(0.8428571428571429, 0.5277777777777778, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
  Text(0.8571428571428571, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
  Text(0.8857142857142857, 0.5833333333333334, 'x[33] <= 0.393\ngini =
0.229\nsamples = 38\nvalue = [33, 5]'),
  Text(0.8714285714285714, 0.5277777777777778, 'x[16] <= 0.35\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
  Text(0.8642857142857143, 0.4722222222222222, 'x[7] <= 0.157\ngini =
0.375\nsamples = 4\nvalue = [3, 1]'),
  Text(0.8571428571428571, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
  Text(0.8714285714285714, 0.4166666666666667, 'gini = 0.0\nsamples = 3\nvalue =
[3, 0]'),
  Text(0.8785714285714286, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
  Text(0.9, 0.5277777777777778, 'x[7] <= 0.992\ngini = 0.117\nsamples = 32\nvalue
= [30, 2]'),
  Text(0.8928571428571429, 0.4722222222222222, 'x[27] <= 0.917\ngini =
0.062\nsamples = 31\nvalue = [30, 1]'),
  Text(0.8857142857142857, 0.4166666666666667, 'gini = 0.0\nsamples = 30\nvalue =
[30, 0]'),
  Text(0.9, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
  Text(0.9071428571428571, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue =
[0, 1]'),
  Text(0.8625, 0.6944444444444444, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
  Text(0.9535714285714286, 0.75, 'x[13] <= 0.812\ngini = 0.456\nsamples =
57\nvalue = [37, 20]'),
  Text(0.9214285714285714, 0.6944444444444444, 'x[7] <= 0.071\ngini =
0.238\nsamples = 29\nvalue = [25, 4]'),
  Text(0.9071428571428571, 0.6388888888888888, 'x[8] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),

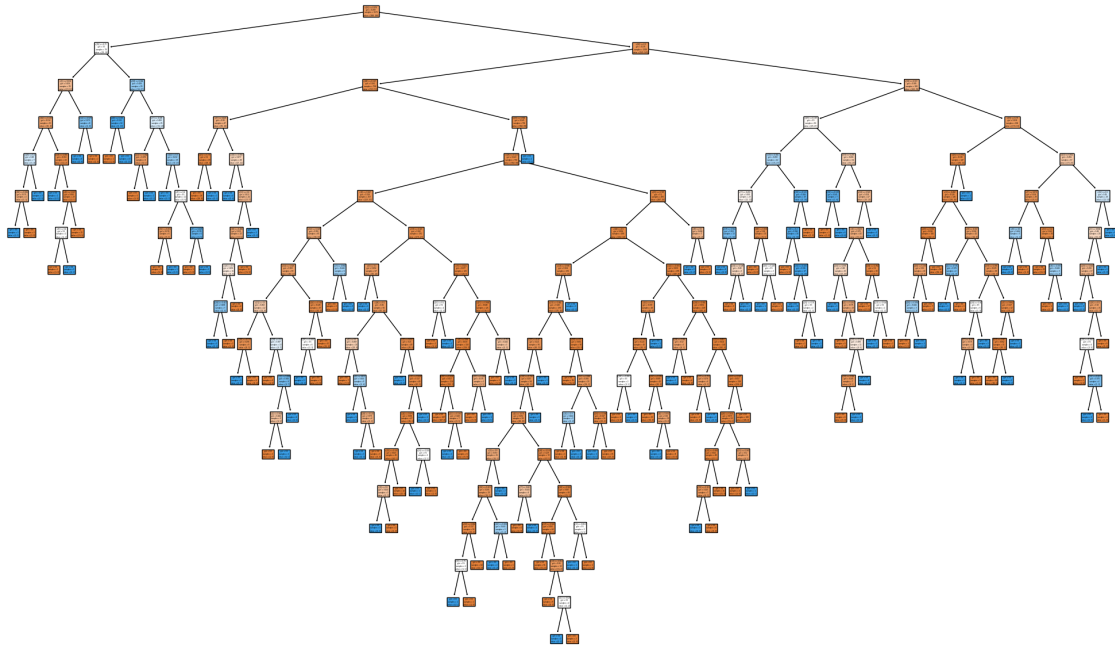
```



```

Text(0.9, 0.5833333333333334, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.9142857142857143, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.9357142857142857, 0.6388888888888888, 'x[31] <= 0.4\ngini =
0.142\nsamples = 26\nvalue = [24, 2]'),
Text(0.9285714285714286, 0.5833333333333334, 'gini = 0.0\nsamples = 23\nvalue =
[23, 0]'),
Text(0.9428571428571428, 0.5833333333333334, 'x[8] <= 0.333\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
Text(0.9357142857142857, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.95, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.9857142857142858, 0.6944444444444444, 'x[31] <= 0.1\ngini =
0.49\nsamples = 28\nvalue = [12, 16]'),
Text(0.9785714285714285, 0.6388888888888888, 'x[3] <= 0.804\ngini =
0.48\nsamples = 20\nvalue = [12, 8]'),
Text(0.9714285714285714, 0.5833333333333334, 'x[29] <= 0.013\ngini =
0.415\nsamples = 17\nvalue = [12, 5]'),
Text(0.9642857142857143, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue =
[0, 2]'),
Text(0.9785714285714285, 0.5277777777777778, 'x[23] <= 0.5\ngini =
0.32\nsamples = 15\nvalue = [12, 3]'),
Text(0.9714285714285714, 0.4722222222222222, 'x[33] <= 0.286\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
Text(0.9642857142857143, 0.4166666666666667, 'gini = 0.0\nsamples = 2\nvalue =
[2, 0]'),
Text(0.9785714285714285, 0.4166666666666667, 'x[31] <= 0.033\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
Text(0.9714285714285714, 0.3611111111111111, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
Text(0.9857142857142858, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue =
[1, 0]'),
Text(0.9857142857142858, 0.4722222222222222, 'gini = 0.0\nsamples = 9\nvalue =
[9, 0]'),
Text(0.9857142857142858, 0.5833333333333334, 'gini = 0.0\nsamples = 3\nvalue =
[0, 3]'),
Text(0.9928571428571429, 0.6388888888888888, 'gini = 0.0\nsamples = 8\nvalue =
[0, 8]')

```



0.18 GridSearchCv

```
[95]: 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']
}
```

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

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

```
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
```

```

/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(

```

[illegible]


```

warnings.warn(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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'`.

```

```

removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
    warnings.warn(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(

```

```

/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(

```



```

warnings.warn(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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'`.

```

```

removed in 1.3. To keep the past behaviour, explicitly set
`max_features='sqrt'`.
    warnings.warn(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(

```

```

/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(
/usr/local/lib/python3.10/dist-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(

```

```

[97]: 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')

```

```

[98]: grid_search.best_params_

```

```

[98]: {'criterion': 'entropy',
      'max_depth': 3,
      'max_features': 'auto',
      'splitter': 'random'}

```

```

[99]: grid_search.best_score_

```

```

[99]: 0.8486404615939417

```

```

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

```

```

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

```

```

[101]: pred=dtc_cv.predict(x_test)

```

```

[102]: pred

```

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

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

	precision	recall	f1-score	support
0	0.84	0.97	0.90	245
1	0.30	0.06	0.10	49
accuracy			0.82	294
macro avg	0.57	0.52	0.50	294
weighted avg	0.75	0.82	0.77	294

0.19 Random Forest

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

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

```
[105]: [{'max_depth': [10, 11, 12, 13, 14],
'max_features': [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]}]
```

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

```
[106]: 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')
```

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

```
/usr/local/lib/python3.10/dist-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 "/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_validation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File "/usr/local/lib/python3.10/dist-packages/sklearn/ensemble/_forest.py", line 340, in fit
    self._validate_params()
  File "/usr/local/lib/python3.10/dist-packages/sklearn/base.py", line 600, in _validate_params
    validate_parameter_constraints(
  File "/usr/local/lib/python3.10/dist-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 {'log2', 'auto' (deprecated), 'sqrt'}
or None. Got 0 instead.
```

```
warnings.warn(some_fits_failed_message, FitFailedWarning)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_search.py:952:
UserWarning: One or more of the test scores are non-finite: [      nan
0.8478053  0.85289005 0.8528828  0.85800377 0.8605389
0.85884398 0.85796755 0.85882949 0.85967695 0.85712734 0.86220484
0.85881501 0.86392148      nan 0.84779082 0.85885122 0.85798928
0.86138635 0.86309576 0.85885122 0.86050992 0.85798204 0.85883674
0.85625815 0.86137187 0.85967695 0.86307403      nan 0.85205708
0.85544691 0.85714907 0.85798204 0.85882949 0.8596842  0.86137911
0.8596842  0.85881501 0.85883674 0.85969144 0.85796755 0.85798928
      nan 0.85121686 0.85969868 0.85798204 0.8596842  0.85883674
0.86137187 0.86308127 0.86137187 0.86136462 0.85625815 0.86477618
0.86222657 0.86220484      nan 0.84694336 0.85459221 0.85543242
0.8605389  0.85884398 0.86308127 0.86222657 0.85797479 0.85798204
0.85796755 0.85798204 0.85457048 0.86564537]
```

```
warnings.warn(
```

```
[107]: 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')
```

```
[108]: pred=rfc_cv.predict(x_test)
```

```
[109]: pred
```

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

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

	precision	recall	f1-score	support
0	0.86	0.99	0.92	245
1	0.73	0.16	0.27	49
accuracy			0.85	294
macro avg	0.79	0.58	0.59	294
weighted avg	0.83	0.85	0.81	294

```
[111]: rfc_cv.best_params_
```

```
[111]: {'max_depth': 14, 'max_features': 13}
```

```
[112]: rfc_cv.best_score_
```

```
[112]: 0.8656453715775749
```