```
# Rayidi Abhiram
# 21BCE9261
# VITAP MORNING SLOT
# ASSIGNMENT-4
# Performing Multiple models (Logistic Regression, Decision Tree Regressor, Random Forest Regressor) on Employee Attrition Dataset
```

Import neccessary libraries

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Import the dataset

```
df = pd.read_csv("/content/drive/MyDrive/Colab Datasets/WA_Fn-UseC_-HR-Employee-Attrition.csv")
```

```
df.head()
```

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Educati
0	41	Yes	Travel_Rarely	1102	Sales	1	
1	49	No	Travel_Frequently	279	Research & Development	8	
2	37	Yes	Travel_Rarely	1373	Research & Development	2	
3	33	No	Travel_Frequently	1392	Research & Development	3	
4	27	No	Travel_Rarely	591	Research & Development	2	

5 rows × 35 columns

```
df.shape
```

```
(1470, 35)
```

```
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
```

```
df.describe()
```

	Age	DailyRate	DistanceFromHome	Education	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	HourlyRate	J
count	1470.000000	1470.000000	1470.000000	1470.000000	1470.0	1470.000000	1470.000000	1470.000000	
mean	36.923810	802.485714	9.192517	2.912925	1.0	1024.865306	2.721769	65.891156	
std	9.135373	403.509100	8.106864	1.024165	0.0	602.024335	1.093082	20.329428	
min	18.000000	102.000000	1.000000	1.000000	1.0	1.000000	1.000000	30.000000	
25%	30.000000	465.000000	2.000000	2.000000	1.0	491.250000	2.000000	48.000000	
50%	36.000000	802.000000	7.000000	3.000000	1.0	1020.500000	3.000000	66.000000	
75%	43.000000	1157.000000	14.000000	4.000000	1.0	1555.750000	4.000000	83.750000	
max	60.000000	1499.000000	29.000000	5.000000	1.0	2068.000000	4.000000	100.000000	

8 rows × 26 columns

Handling null values

```
df.isnull().any()
# No null values found
```

```
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
```

```
df.isnull().sum()
# Since the dataset does not contain any null values, we can proceed to next step.
```

```
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
```

Data Visualization

```
df.corr()
```

```
<ipython-input-210-2f6f6606aa2c>:1: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future ve
  df.corr()
```

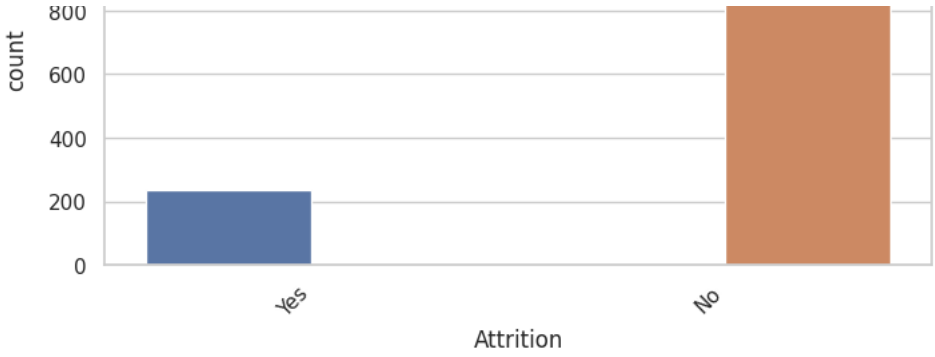
	Age	DailyRate	DistanceFromHome	Education	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	H
Age	1.000000	0.010661	-0.001686	0.208034	NaN	-0.010145	0.010146	
DailyRate	0.010661	1.000000	-0.004985	-0.016806	NaN	-0.050990	0.018355	
DistanceFromHome	-0.001686	-0.004985	1.000000	0.021042	NaN	0.032916	-0.016075	
Education	0.208034	-0.016806	0.021042	1.000000	NaN	0.042070	-0.027128	
EmployeeCount	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
EmployeeNumber	-0.010145	-0.050990	0.032916	0.042070	NaN	1.000000	0.017621	
EnvironmentSatisfaction	0.010146	0.018355	-0.016075	-0.027128	NaN	0.017621	1.000000	
HourlyRate	0.024287	0.023381	0.031131	0.016775	NaN	0.035179	-0.049857	
JobInvolvement	0.029820	0.046135	0.008783	0.042438	NaN	-0.006888	-0.008278	
JobLevel	0.509604	0.002966	0.005303	0.101589	NaN	-0.018519	0.001212	
JobSatisfaction	-0.004892	0.030571	-0.003669	-0.011296	NaN	-0.046247	-0.006784	
MonthlyIncome	0.497855	0.007707	-0.017014	0.094961	NaN	-0.014829	-0.006259	
MonthlyRate	0.028051	-0.032182	0.027473	-0.026084	NaN	0.012648	0.037600	
NumCompaniesWorked	0.299635	0.038153	-0.029251	0.126317	NaN	-0.001251	0.012594	
PercentSalaryHike	0.003634	0.022704	0.040235	-0.011111	NaN	-0.012944	-0.031701	
PerformanceRating	0.001904	0.000473	0.027110	-0.024539	NaN	-0.020359	-0.029548	
RelationshipSatisfaction	0.053535	0.007846	0.006557	-0.009118	NaN	-0.069861	0.007665	
StandardHours	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
StockOptionLevel	0.037510	0.042143	0.044872	0.018422	NaN	0.062227	0.003432	
TotalWorkingYears	0.680381	0.014515	0.004628	0.148280	NaN	-0.014365	-0.002693	
TrainingTimesLastYear	-0.019621	0.002453	-0.036942	-0.025100	NaN	0.023603	-0.019359	
WorkLifeBalance	-0.021490	-0.037848	-0.026556	0.009819	NaN	0.010309	0.027627	
YearsAtCompany	0.311309	-0.034055	0.009508	0.069114	NaN	-0.011240	0.001458	
YearsInCurrentRole	0.212901	0.009932	0.018845	0.060236	NaN	-0.008416	0.018007	
YearsSinceLastPromotion	0.216513	-0.033229	0.010029	0.054254	NaN	-0.009019	0.016194	
YearsWithCurrManager	0.202089	-0.026363	0.014406	0.069065	NaN	-0.009197	-0.004999	

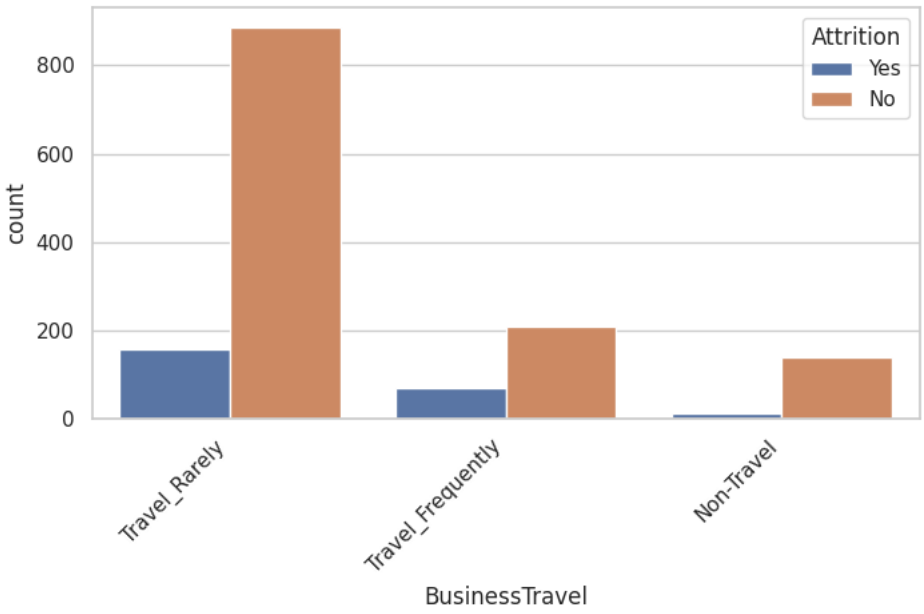
26 rows × 26 columns

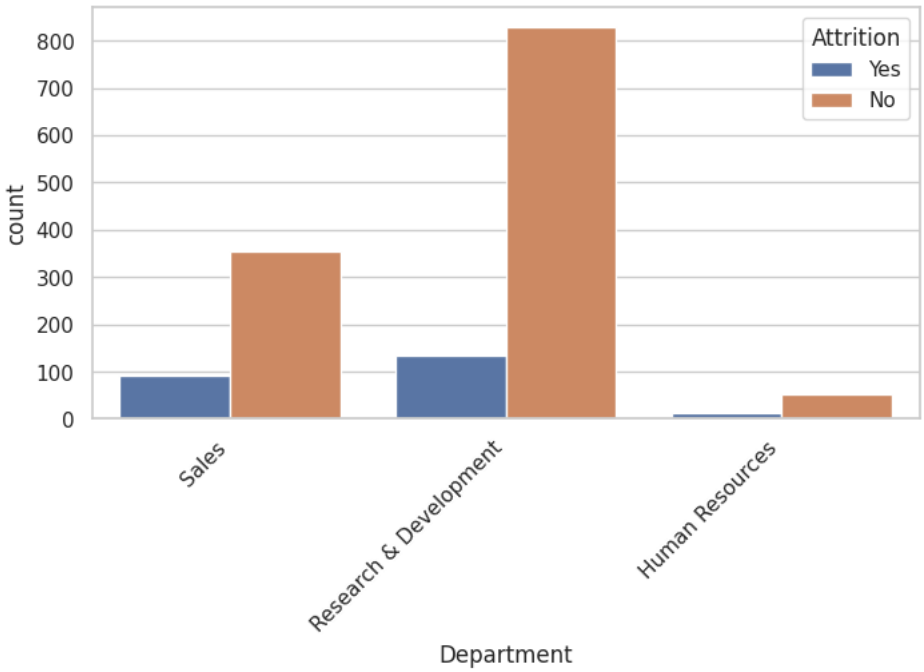
```
plt.figure(figsize=(25,16))
sns.heatmap(df.corr(),annot=True)
```

```
<ipython-input-211-1020bb8eeb59>:2: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future ve
  sns.heatmap(df.corr(),annot=True)
<Axes: >
```

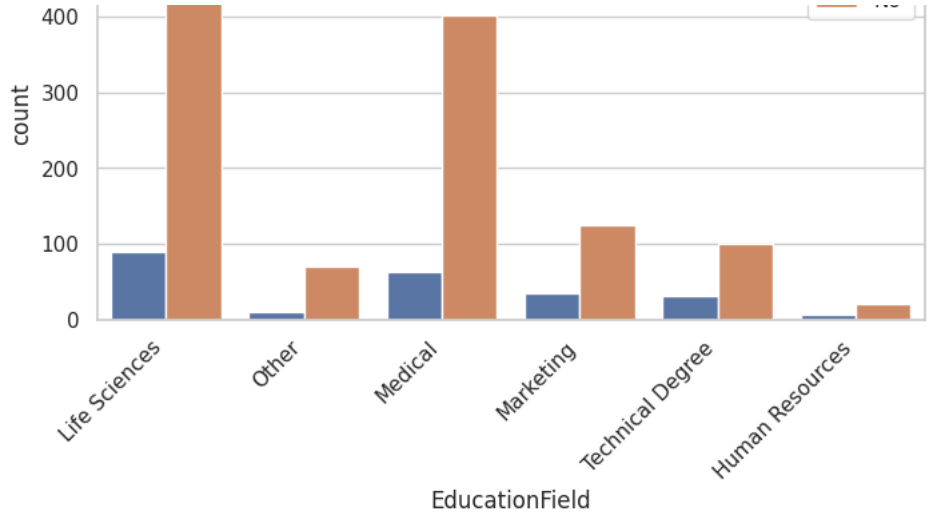

```
categorical_columns = df.select_dtypes(include=['object']).columns
for col in categorical_columns:
    plt.subplots(figsize=(8, 4))
    sns.countplot(data=df, x=col, hue="Attrition")
    plt.xticks(rotation=45, ha='right')

"""
Inference :
1) We can say that the emoployees who travel frequently are more likely to be
   Attrited followed by travels rarely followed by non travel.
   So people who not travel has less Attrtion.
2) Similarly there isn't much linear trend regarding Department and Attrition.
3) Intrestingly the count of Attrition is more in Research and Development
   but the number changes when comes to percentage.
4) Also the gender may not affect much Attrition.
5) Research Director Designation has less Attrition which is leading to security
   of role, Also we can find that more than 50% are Attrited in
   Sales Representative Designation.
6) Not so suprised by how the Bachelors has been most Attrited compared to
   Married and Divorced.
7) Not much progressed by the analytics of over18.
8) Major point founded is that people who do overtime has more possibility of
   Attrition.
"""
```

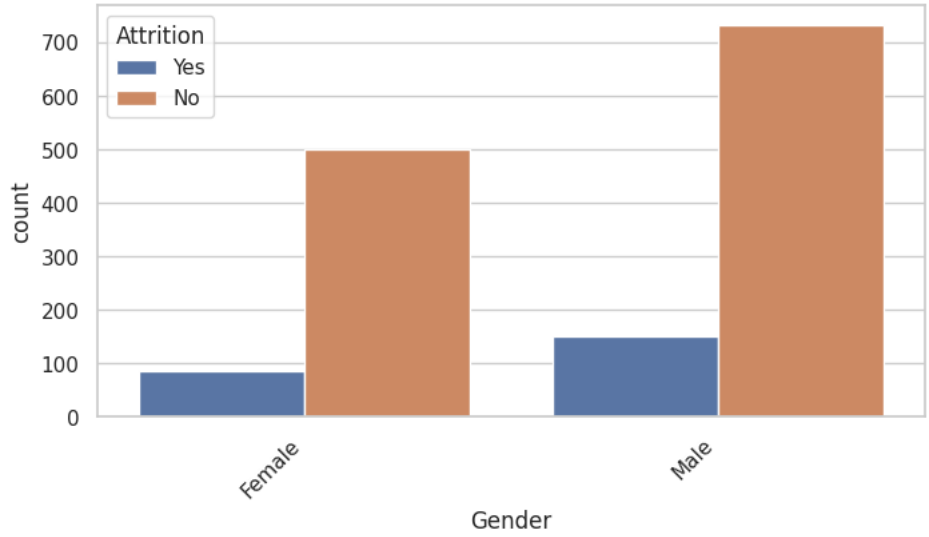

800
600
400
200
0
count
Yes
No
Attrition

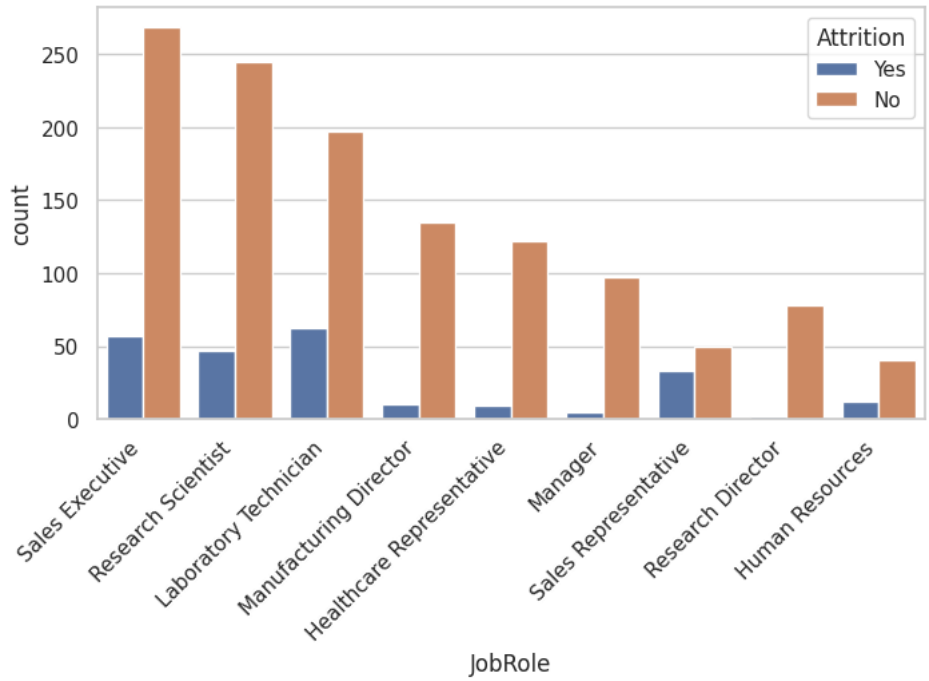

Attrition
Yes
No
800
600
400
200
0
count
Travel_Rarely
Travel_Frequently
Non-Travel
BusinessTravel

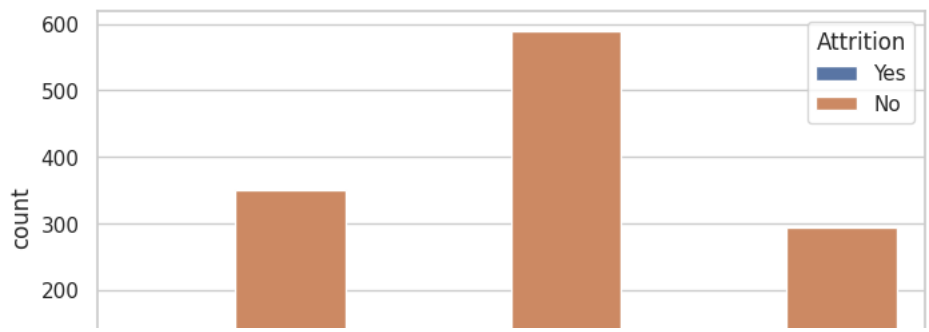

Attrition
Yes
No
800
700
600
500
400
300
200
100
0
count
Sales
Research & Development
Human Resources
Department

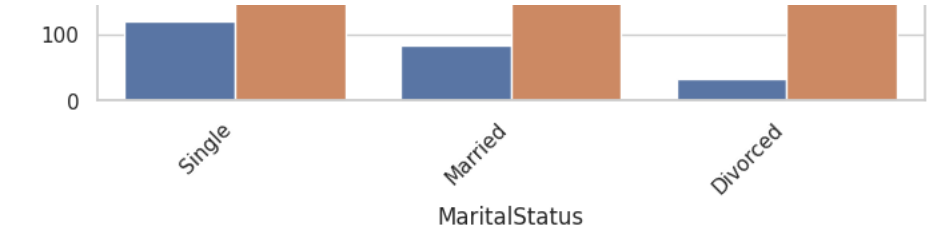

Attrition
Yes
No
500

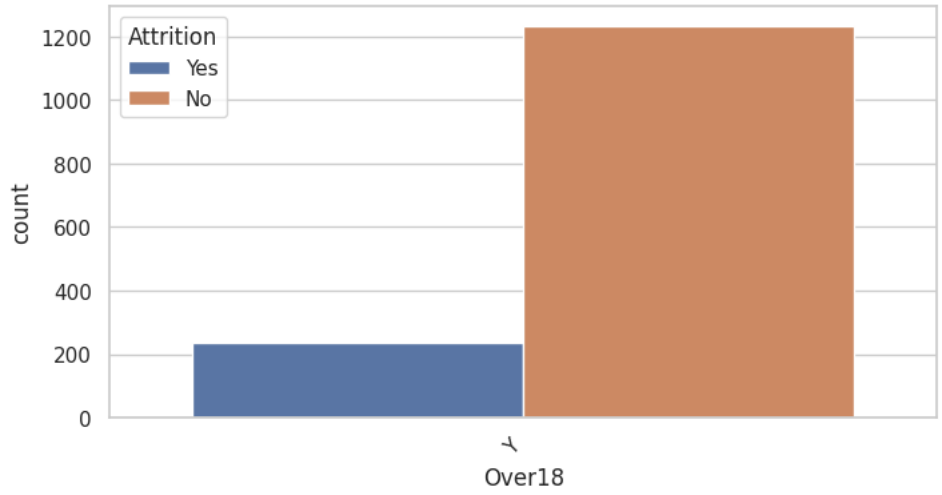

count
400
300
200
100
0
Life Sciences
Other
Medical
Marketing
Technical Degree
Human Resources
EducationField


Attrition
Yes
No
count
700
600
500
400
300
200
100
0
Female
Male
Gender


Attrition
Yes
No
count
250
200
150
100
50
0
Sales Executive
Research Scientist
Laboratory Technician
Manufacturing Director
Healthcare Representative
Manager
Sales Representative
Research Director
Human Resources
JobRole

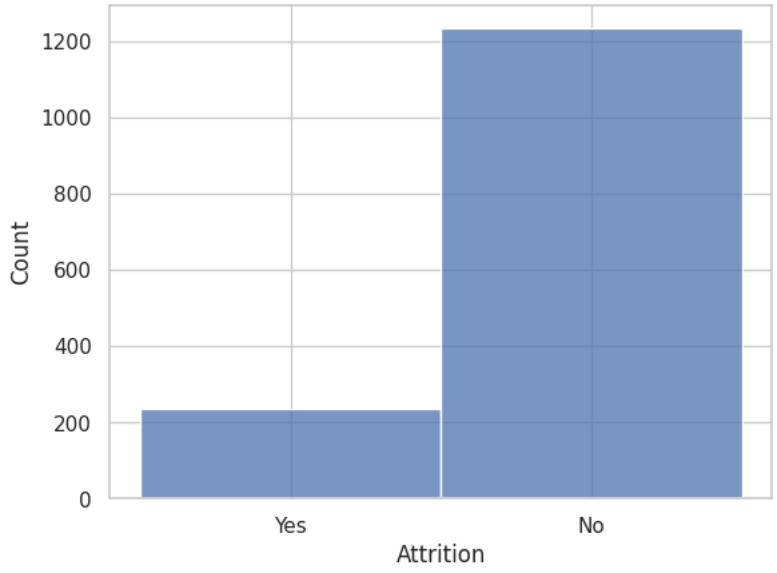

Attrition
Yes
No
count
600
500
400
300
200





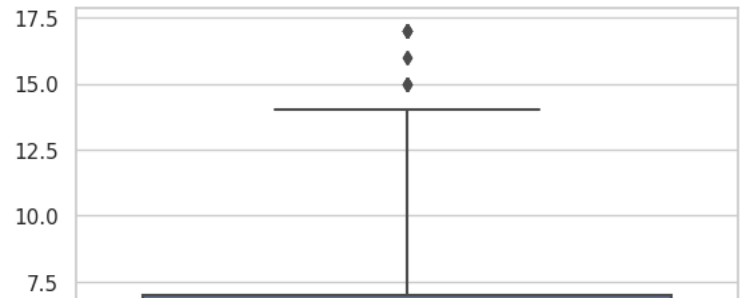

```
sns.histplot(df["Attrition"])
```

```
<Axes: xlabel='Attrition', ylabel='Count'>
```



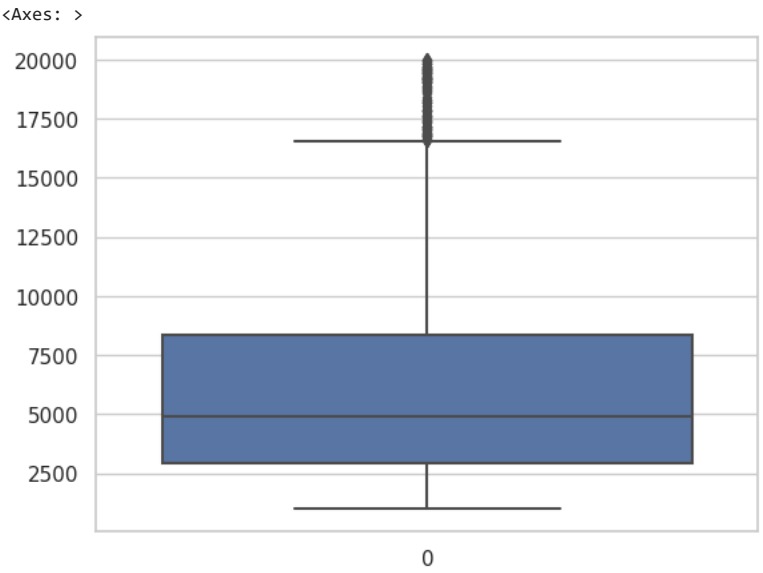

```
sns.boxplot(df.YearsWithCurrManager )
```

```
<Axes: >
```



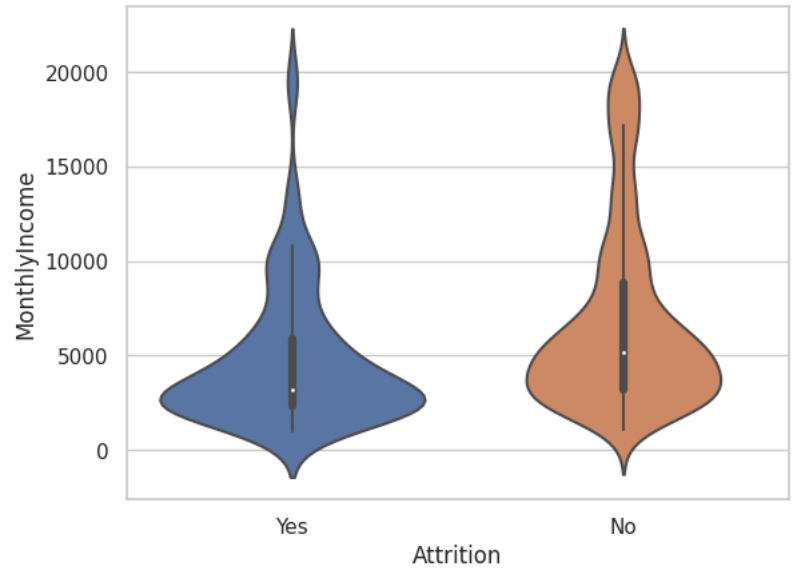

```
sns.boxplot(df.MonthlyIncome )
```

```
<Axes: >
```




```
sns.violinplot(x= "Attrition", y= "MonthlyIncome", data=df)
```

```
<Axes: xlabel='Attrition', ylabel='MonthlyIncome'>
```



Splitting Dependent and Independent variables

```
x = df.drop(columns=['Attrition', 'DailyRate', 'StockOptionLevel', 'StandardHours', 'Over18'])
```

```
y = df['Attrition']
```

```
x.head()
```

	Age	BusinessTravel	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber	EnvironmentSatisfac
0	41	Travel_Rarely	Sales	1	2	Life Sciences	1	1	
1	49	Travel_Frequently	Research & Development	8	1	Life Sciences	1	2	
2	37	Travel_Rarely	Research & Development	2	2	Other	1	4	
3	33	Travel_Frequently	Research & Development	3	4	Life Sciences	1	5	
4	27	Travel_Rarely	Research &	2	1	Medical	1	7	

```
y.head()
```

```
0    Yes
1     No
2    Yes
3     No
4     No
Name: Attrition, dtype: object
```

Encoding

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


x["Gender"] = le.fit_transform(x["Gender"])
x["BusinessTravel"] = le.fit_transform(x["BusinessTravel"])
x['Department'] = le.fit_transform(x['Department'])
x['EducationField'] = le.fit_transform(x['EducationField'])
x['JobRole'] = le.fit_transform(x['JobRole'])
x['MaritalStatus'] = le.fit_transform(x['MaritalStatus'])
x['OverTime'] = le.fit_transform(x['OverTime'])
y = le.fit_transform(y)


x.shape
```

```
(1470, 30)
```

Feature Scaling

```
from sklearn.preprocessing import StandardScaler

sc = StandardScaler()

x_scaled = pd.DataFrame(sc.fit_transform(x), columns = x.columns)


x_scaled
```

	Age	BusinessTravel	Department	DistanceFromHome	Education	EducationField	EmployeeCount	EmployeeNumber	EnvironmentS
0	0.446350	0.590048	1.401512	-1.010909	-0.891688	-0.937414	0.0	-1.701283	
1	1.322365	-0.913194	-0.493817	-0.147150	-1.868426	-0.937414	0.0	-1.699621	
2	0.008343	0.590048	-0.493817	-0.887515	-0.891688	1.316673	0.0	-1.696298	
3	-0.429664	-0.913194	-0.493817	-0.764121	1.061787	-0.937414	0.0	-1.694636	
4	-1.086676	0.590048	-0.493817	-0.887515	-1.868426	0.565311	0.0	-1.691313	
...	...	...	...	...	...	...	...	...	
1465	-0.101159	-0.913194	-0.493817	1.703764	-0.891688	0.565311	0.0	1.721670	
1466	0.227347	0.590048	-0.493817	-0.393938	-1.868426	0.565311	0.0	1.723332	
1467	-1.086676	0.590048	-0.493817	-0.640727	0.085049	-0.937414	0.0	1.726655	
1468	1.322365	-0.913194	1.401512	-0.887515	0.085049	0.565311	0.0	1.728317	
1469	-0.320163	0.590048	-0.493817	-0.147150	0.085049	0.565311	0.0	1.733302	

1470 rows × 30 columns

Splitting Testing and Training data

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test= train_test_split(x_scaled, y, test_size=0.2, random_state = 0)
```

```
x_train.shape, x_test.shape, y_train.shape, y_test.shape
```

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

Model Building

Logistic Regression

```
from sklearn.linear_model import LogisticRegression
lo = LogisticRegression()
```

```
lo.fit(x_train, y_train)
```

```
▾ LogisticRegression
LogisticRegression()
```

```
y_pred= lo.predict(x_test)
```

Evaluation Metrics

```
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report,roc_auc_score,roc_curve
accuracy_score(y_test, y_pred)
```

```
0.8843537414965986
```

```
confusion_matrix(y_test,y_pred)
```

```
array([[240,   5],
       [ 29,  20]])
```

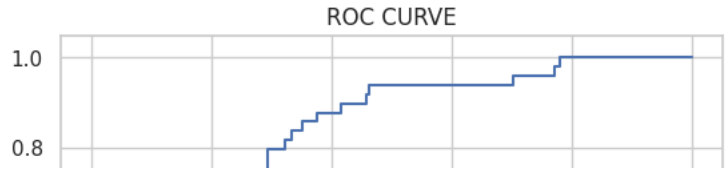
```
print(classification_report(y_test,y_pred))
```

```
                  precision    recall  f1-score   support

               0       0.89      0.98      0.93       245
               1       0.80      0.41      0.54        49

        accuracy                           0.88       294
       macro avg       0.85      0.69      0.74       294
    weighted avg       0.88      0.88      0.87       294
```

```
# ROC-AUC Curve
probability = lo.predict_proba(x_test)[:,1]
```

```
fpr, tpr, thresholds = roc_curve(y_test, probability)
```

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



Decision Tree Regressor

```
from sklearn.tree import DecisionTreeClassifier
dt_model = DecisionTreeClassifier()
```

```
dt_model.fit(x_train, y_train)
```

```
DecisionTreeClassifier()
```

```
dt_pred = dt_model.predict(x_test)
```

Evaluation Metrics

```
accuracy_score(y_test, dt_pred)
```

```
0.7312925170068028
```

```
confusion_matrix(y_test, dt_pred)
```

```
array([[201,  44],
       [ 35,  14]])
```

```
pd.crosstab(y_test, dt_pred)
```

col_0	0	1
row_0		
0	201	44
1	35	14

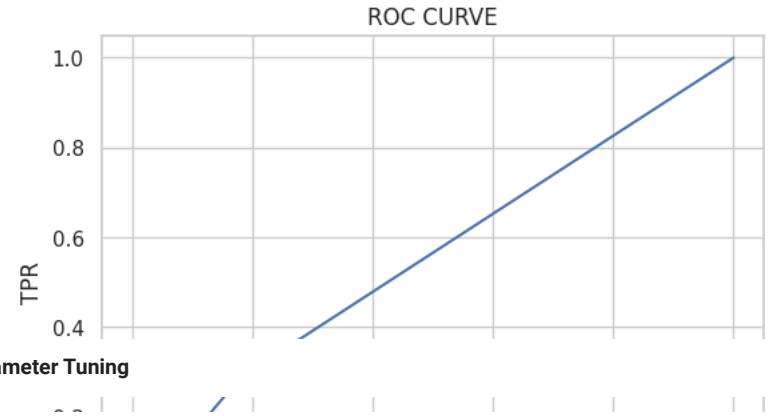
```
print(classification_report(y_test,y_pred))
```

```
                  precision    recall  f1-score   support

               0       0.89      0.98      0.93       245
               1       0.80      0.41      0.54        49

        accuracy                           0.88       294
       macro avg       0.85      0.69      0.74       294
    weighted avg       0.88      0.88      0.87       294
```

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

```
# ROC CURVE
fpr, tpr, thresholds = roc_curve(y_test, probability)
```

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



Parameter Tuning

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

```
[Text(0.322849025974026, 0.9722222222222222, 'x[23] <= -1.257\ngini = 0.269\nsamples = 1176\nvalue = [988, 188]'),
 Text(0.07142857142857142, 0.9166666666666666, 'x[15] <= 0.552\ngini = 0.5\nsamples = 78\nvalue = [39, 39]'),
 Text(0.04220779220779221, 0.8611111111111112, 'x[3] <= 0.902\ngini = 0.426\nsamples = 39\nvalue = [27, 12]'),
 Text(0.025974025974025976, 0.8055555555555556, 'x[14] <= -1.114\ngini = 0.312\nsamples = 31\nvalue = [25, 6]'),
 Text(0.012987012987012988, 0.75, 'x[16] <= -0.937\ngini = 0.49\nsamples = 7\nvalue = [3, 4]'),
 Text(0.006493506493506494, 0.6944444444444444, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
 Text(0.01948051948051948, 0.6944444444444444, 'x[17] <= -0.838\ngini = 0.375\nsamples = 4\nvalue = [3, 1]'),
 Text(0.012987012987012988, 0.6388888888888888, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.025974025974025976, 0.6388888888888888, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
 Text(0.03896103896103896, 0.75, 'x[18] <= -0.878\ngini = 0.153\nsamples = 24\nvalue = [22, 2]'),
 Text(0.032467532467532464, 0.6944444444444444, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.045454545454545456, 0.6944444444444444, 'x[8] <= -1.118\ngini = 0.083\nsamples = 23\nvalue = [22, 1]'),
 Text(0.03896103896103896, 0.6388888888888888, 'x[21] <= 0.96\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.032467532467532464, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.045454545454545456, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.05194805194805195, 0.6388888888888888, 'gini = 0.0\nsamples = 21\nvalue = [21, 0]'),
 Text(0.05844155844155844, 0.8055555555555556, 'x[7] <= -0.378\ngini = 0.375\nsamples = 8\nvalue = [2, 6]'),
 Text(0.05194805194805195, 0.75, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
 Text(0.06493506493506493, 0.75, 'gini = 0.0\nsamples = 6\nvalue = [0, 6]'),
 Text(0.10064935064935066, 0.8611111111111112, 'x[10] <= -0.511\ngini = 0.426\nsamples = 39\nvalue = [12, 27]'),
 Text(0.08441558441558442, 0.8055555555555556, 'x[16] <= -0.763\ngini = 0.133\nsamples = 14\nvalue = [1, 13]'),
 Text(0.07792207792207792, 0.75, 'gini = 0.0\nsamples = 13\nvalue = [0, 13]'),
 Text(0.09090909090909091, 0.75, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.11688311688311688, 0.8055555555555556, 'x[7] <= -1.341\ngini = 0.493\nsamples = 25\nvalue = [11, 14]'),
 Text(0.1038961038961039, 0.75, 'x[19] <= 0.482\ngini = 0.278\nsamples = 6\nvalue = [5, 1]'),
 Text(0.09740259740259741, 0.6944444444444444, 'gini = 0.0\nsamples = 5\nvalue = [5, 0]'),
 Text(0.11038961038961038, 0.6944444444444444, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.12987012987012986, 0.75, 'x[14] <= -0.207\ngini = 0.432\nsamples = 19\nvalue = [6, 13]'),
 Text(0.12337662337662338, 0.6944444444444444, 'gini = 0.0\nsamples = 7\nvalue = [0, 7]'),
 Text(0.13636363636363635, 0.6944444444444444, 'x[5] <= -0.186\ngini = 0.5\nsamples = 12\nvalue = [6, 6]'),
 Text(0.12337662337662338, 0.6388888888888888, 'x[11] <= -1.729\ngini = 0.278\nsamples = 6\nvalue = [5, 1]'),
 Text(0.11688311688311688, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.12987012987012986, 0.5833333333333334, 'gini = 0.0\nsamples = 5\nvalue = [5, 0]'),
 Text(0.14935064935064934, 0.6388888888888888, 'x[7] <= -0.845\ngini = 0.278\nsamples = 6\nvalue = [1, 5]'),
 Text(0.14285714285714285, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.15584415584415584, 0.5833333333333334, 'gini = 0.0\nsamples = 5\nvalue = [0, 5]'),
 Text(0.5742694805194806, 0.9166666666666666, 'x[19] <= 0.482\ngini = 0.235\nsamples = 1098\nvalue = [949, 149]'),
 Text(0.3234070616883117, 0.8611111111111112, 'x[25] <= -1.786\ngini = 0.162\nsamples = 798\nvalue = [727, 71]'),
 Text(0.16883116883116883, 0.8055555555555556, 'x[7] <= -0.173\ngini = 0.38\nsamples = 47\nvalue = [35, 12]'),
 Text(0.15584415584415584, 0.75, 'x[15] <= 0.552\ngini = 0.1\nsamples = 19\nvalue = [18, 1]'),
 Text(0.14935064935064934, 0.6944444444444444, 'gini = 0.0\nsamples = 18\nvalue = [18, 0]'),
 Text(0.16233766233766234, 0.6944444444444444, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.18181818181818182, 0.75, 'x[16] <= -0.789\ngini = 0.477\nsamples = 28\nvalue = [17, 11]'),
 Text(0.17532467532467533, 0.6944444444444444, 'gini = 0.0\nsamples = 4\nvalue = [0, 4]'),
 Text(0.18831168831168832, 0.6944444444444444, 'x[7] <= 0.099\ngini = 0.413\nsamples = 24\nvalue = [17, 7]'),
 Text(0.18181818181818182, 0.6388888888888888, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
 Text(0.19480519480519481, 0.6388888888888888, 'x[29] <= 0.386\ngini = 0.351\nsamples = 22\nvalue = [17, 5]'),
 Text(0.18181818181818182, 0.5833333333333334, 'x[24] <= 0.544\ngini = 0.133\nsamples = 14\nvalue = [13, 1]'),
 Text(0.17532467532467533, 0.5277777777777778, 'gini = 0.0\nsamples = 12\nvalue = [12, 0]'),
 Text(0.18831168831168832, 0.5277777777777778, 'x[5] <= 0.565\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.18181818181818182, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.19480519480519481, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.2077922077922078, 0.5833333333333334, 'x[18] <= -0.878\ngini = 0.5\nsamples = 8\nvalue = [4, 4]'),
 Text(0.2012987012987013, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
 Text(0.21428571428571427, 0.5277777777777778, 'x[8] <= -0.203\ngini = 0.444\nsamples = 6\nvalue = [2, 4]'),
 Text(0.2077922077922078, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.22077922077922077, 0.4722222222222222, 'x[23] <= 1.893\ngini = 0.32\nsamples = 5\nvalue = [1, 4]'),
 Text(0.21428571428571427, 0.4166666666666667, 'gini = 0.0\nsamples = 4\nvalue = [0, 4]'),
 Text(0.22727272727272727, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
 Text(0.47798295454545453, 0.8055555555555556, 'x[23] <= 3.564\ngini = 0.145\nsamples = 751\nvalue = [692, 59]'),
 Text(0.47148944805194803, 0.75, 'x[26] <= -0.41\ngini = 0.143\nsamples = 750\nvalue = [692, 58]'),
 Text(0.3449675324675325, 0.6944444444444444, 'x[8] <= -1.118\ngini = 0.218\nsamples = 257\nvalue = [225, 32]'),
 Text(0.30357142857142855, 0.6388888888888888, 'x[29] <= -0.455\ngini = 0.355\nsamples = 65\nvalue = [50, 15]'),
 Text(0.2824675324675325, 0.5833333333333334, 'x[29] <= -1.016\ngini = 0.303\nsamples = 59\nvalue = [48, 11]'),
 Text(0.2597402597402597, 0.5277777777777778, 'x[11] <= -0.323\ngini = 0.463\nsamples = 22\nvalue = [14, 8]'),
 Text(0.24675324675324675, 0.4722222222222222, 'x[10] <= -1.151\ngini = 0.198\nsamples = 9\nvalue = [8, 1]'),
 Text(0.24025974025974026, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.2532467532467532, 0.4166666666666667, 'gini = 0.0\nsamples = 8\nvalue = [8, 0]'),
 Text(0.2727272727272727, 0.4722222222222222, 'x[10] <= -0.388\ngini = 0.497\nsamples = 13\nvalue = [6, 7]'),
 Text(0.2662337662337662, 0.4166666666666667, 'gini = 0.0\nsamples = 4\nvalue = [4, 0]'),
 Text(0.2792207792207792, 0.4166666666666667, 'x[3] <= -0.024\ngini = 0.346\nsamples = 9\nvalue = [2, 7]'),
 Text(0.2727272727272727, 0.3611111111111111, 'x[23] <= -0.679\ngini = 0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.2662337662337662, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
 Text(0.2792207792207792, 0.3055555555555556, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
 Text(0.2857142857142857, 0.3611111111111111, 'gini = 0.0\nsamples = 6\nvalue = [0, 6]'),
 Text(0.3051948051948052, 0.5277777777777778, 'x[14] <= -1.114\ngini = 0.149\nsamples = 37\nvalue = [34, 3]'),
 Text(0.2987012987012987, 0.4722222222222222, 'x[25] <= -0.37\ngini = 0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.2922077922077922, 0.4166666666666667, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
 Text(0.3051948051948052, 0.4166666666666667, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
 Text(0.3116883116883117, 0.4722222222222222, 'gini = 0.0\nsamples = 31\nvalue = [31, 0]'),
 Text(0.3246753246753247, 0.5833333333333334, 'x[7] <= -1.479\ngini = 0.444\nsamples = 6\nvalue = [2, 4]'),
 Text(0.3181818181818182, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
 Text(0.33116883116883117, 0.5277777777777778, 'gini = 0.0\nsamples = 4\nvalue = [0, 4]'),
 Text(0.38636363636363635, 0.6388888888888888, 'x[0] <= -0.594\ngini = 0.161\nsamples = 192\nvalue = [175, 17]'),
 Text(0.35064935064935066, 0.5833333333333334, 'x[5] <= -1.313\ngini = 0.294\nsamples = 67\nvalue = [55, 12]'),
 Text(0.34415584415584416, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
 Text(0.35714285714285715, 0.5277777777777778, 'x[25] <= -0.37\ngini = 0.26\nsamples = 65\nvalue = [55, 10]'),
 Text(0.32792207792207795, 0.4722222222222222, 'x[5] <= 0.19\ngini = 0.469\nsamples = 16\nvalue = [10, 6]'),
 Text(0.32142857142857145, 0.4166666666666667, 'gini = 0.0\nsamples = 7\nvalue = [7, 0]'),
 Text(0.3344155844155844, 0.4166666666666667, 'x[8] <= 0.712\ngini = 0.444\nsamples = 9\nvalue = [3, 6]'),
```

```
Text(0.32792207792207795, 0.3611111111111111, 'gini = 0.0\nsamples = 5\nvalue = [0, 5]'),
Text(0.3409090909090909, 0.3611111111111111, 'x[11] <= -0.323\ngini = 0.375\nsamples = 4\nvalue = [3, 1]'),
Text(0.3344155844155844, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.3474025974025974, 0.3055555555555556, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
Text(0.38636363636363635, 0.4722222222222222, 'x[8] <= -0.203\ngini = 0.15\nsamples = 49\nvalue = [45, 4]'),
Text(0.37337662337662336, 0.4166666666666667, 'x[16] <= -0.715\ngini = 0.444\nsamples = 6\nvalue = [4, 2]'),
Text(0.36688311688311687, 0.3611111111111111, 'x[14] <= -1.114\ngini = 0.444\nsamples = 3\nvalue = [1, 2]'),
Text(0.3603896103896104, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.37337662337662336, 0.3055555555555556, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.37987012987012986, 0.3611111111111111, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
Text(0.39935064935064934, 0.4166666666666667, 'x[4] <= 1.55\ngini = 0.089\nsamples = 43\nvalue = [41, 2]'),
Text(0.39285714285714285, 0.3611111111111111, 'x[20] <= -1.014\ngini = 0.046\nsamples = 42\nvalue = [41, 1]'),
Text(0.38636363636363635, 0.3055555555555556, 'x[3] <= -0.456\ngini = 0.444\nsamples = 3\nvalue = [2, 1]'),
Text(0.37987012987012986, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.39285714285714285, 0.25, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.39935064935064934, 0.3055555555555556, 'gini = 0.0\nsamples = 39\nvalue = [39, 0]'),
Text(0.40584415584415584, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.42207792207792205, 0.5833333333333334, 'x[7] <= -1.627\ngini = 0.077\nsamples = 125\nvalue = [120, 5]'),
Text(0.40584415584415584, 0.5277777777777778, 'x[13] <= -0.186\ngini = 0.5\nsamples = 4\nvalue = [2, 2]'),
Text(0.39935064935064934, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.41233766233766234, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.4383116883116883, 0.5277777777777778, 'x[17] <= 1.671\ngini = 0.048\nsamples = 121\nvalue = [118, 3]'),
Text(0.4253246753246753, 0.4722222222222222, 'x[13] <= 0.83\ngini = 0.033\nsamples = 118\nvalue = [116, 2]'),
Text(0.41883116883116883, 0.4166666666666667, 'gini = 0.0\nsamples = 95\nvalue = [95, 0]'),
Text(0.4318181818181818, 0.4166666666666667, 'x[11] <= 1.083\ngini = 0.159\nsamples = 23\nvalue = [21, 2]'),
Text(0.4253246753246753, 0.3611111111111111, 'x[25] <= -0.37\ngini = 0.087\nsamples = 22\nvalue = [21, 1]'),
Text(0.41883116883116883, 0.3055555555555556, 'x[14] <= 0.7\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
Text(0.41233766233766234, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.4253246753246753, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.4318181818181818, 0.3055555555555556, 'gini = 0.0\nsamples = 20\nvalue = [20, 0]'),
Text(0.4383116883116883, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.4512987012987013, 0.4722222222222222, 'x[27] <= -0.892\ngini = 0.444\nsamples = 3\nvalue = [2, 1]'),
Text(0.4448051948051948, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.4577922077922078, 0.4166666666666667, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.5980113636363636, 0.6944444444444444, 'x[26] <= 3.999\ngini = 0.1\nsamples = 493\nvalue = [467, 26]'),
Text(0.5596590909090909, 0.6388888888888888, 'x[14] <= -0.207\ngini = 0.094\nsamples = 486\nvalue = [462, 24]'),
Text(0.502435064935065, 0.5833333333333334, 'x[13] <= 1.236\ngini = 0.154\nsamples = 191\nvalue = [175, 16]'),
Text(0.49594155844155846, 0.5277777777777778, 'x[17] <= -0.035\ngini = 0.145\nsamples = 190\nvalue = [175, 15]'),
Text(0.4788961038961039, 0.4722222222222222, 'x[17] <= -0.073\ngini = 0.221\nsamples = 95\nvalue = [83, 12]'),
Text(0.4724025974025974, 0.4166666666666667, 'x[29] <= 2.629\ngini = 0.207\nsamples = 94\nvalue = [83, 11]'),
Text(0.4659090909090909, 0.3611111111111111, 'x[4] <= -0.403\ngini = 0.192\nsamples = 93\nvalue = [83, 10]'),
Text(0.4448051948051948, 0.3055555555555556, 'x[5] <= 1.692\ngini = 0.363\nsamples = 21\nvalue = [16, 5]'),
Text(0.4383116883116883, 0.25, 'x[16] <= 0.501\ngini = 0.266\nsamples = 19\nvalue = [16, 3]'),
Text(0.4253246753246753, 0.19444444444444445, 'x[16] <= -0.597\ngini = 0.117\nsamples = 16\nvalue = [15, 1]'),
Text(0.41883116883116883, 0.1388888888888889, 'x[3] <= 1.334\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
Text(0.41233766233766234, 0.08333333333333333, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.4253246753246753, 0.08333333333333333, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.4318181818181818, 0.1388888888888889, 'gini = 0.0\nsamples = 14\nvalue = [14, 0]'),
Text(0.4512987012987013, 0.19444444444444445, 'x[26] <= 2.448\ngini = 0.444\nsamples = 3\nvalue = [1, 2]'),
Text(0.4448051948051948, 0.1388888888888889, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.4577922077922078, 0.1388888888888889, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.4512987012987013, 0.25, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.487012987012987, 0.3055555555555556, 'x[27] <= -0.477\ngini = 0.129\nsamples = 72\nvalue = [67, 5]'),
Text(0.4707792207792208, 0.25, 'x[7] <= 0.633\ngini = 0.444\nsamples = 6\nvalue = [4, 2]'),
Text(0.4642857142857143, 0.19444444444444445, 'gini = 0.0\nsamples = 4\nvalue = [4, 0]'),
Text(0.4772727272727273, 0.19444444444444445, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.5032467532467533, 0.25, 'x[10] <= 1.654\ngini = 0.087\nsamples = 66\nvalue = [63, 3]'),
Text(0.4902597402597403, 0.19444444444444445, 'x[24] <= 0.544\ngini = 0.061\nsamples = 64\nvalue = [62, 2]'),
Text(0.4837662337662338, 0.1388888888888889, 'gini = 0.0\nsamples = 51\nvalue = [51, 0]'),
Text(0.4967532467532468, 0.1388888888888889, 'x[8] <= -0.203\ngini = 0.26\nsamples = 13\nvalue = [11, 2]'),
Text(0.4902597402597403, 0.08333333333333333, 'x[27] <= 0.351\ngini = 0.5\nsamples = 4\nvalue = [2, 2]'),
Text(0.4837662337662338, 0.027777777777777776, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.4967532467532468, 0.027777777777777776, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.5032467532467533, 0.08333333333333333, 'gini = 0.0\nsamples = 9\nvalue = [9, 0]'),
Text(0.5162337662337663, 0.19444444444444445, 'x[9] <= -0.204\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
Text(0.5097402597402597, 0.1388888888888889, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.5227272727272727, 0.1388888888888889, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.4788961038961039, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.48538961038961037, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.512987012987013, 0.4722222222222222, 'x[18] <= 0.724\ngini = 0.061\nsamples = 95\nvalue = [92, 3]'),
Text(0.5064935064935064, 0.4166666666666667, 'gini = 0.0\nsamples = 76\nvalue = [76, 0]'),
Text(0.5194805194805194, 0.4166666666666667, 'x[7] <= -1.149\ngini = 0.266\nsamples = 19\nvalue = [16, 3]'),
Text(0.5064935064935064, 0.3611111111111111, 'x[5] <= 0.19\ngini = 0.444\nsamples = 3\nvalue = [1, 2]'),
Text(0.5, 0.3055555555555556, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.512987012987013, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.5324675324675324, 0.3611111111111111, 'x[27] <= 2.007\ngini = 0.117\nsamples = 16\nvalue = [15, 1]'),
Text(0.525974025974026, 0.3055555555555556, 'gini = 0.0\nsamples = 14\nvalue = [14, 0]'),
Text(0.538961038961039, 0.3055555555555556, 'x[29] <= 0.106\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
Text(0.5324675324675324, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.5454545454545454, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.5089285714285714, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6168831168831169, 0.5833333333333334, 'x[20] <= -1.014\ngini = 0.053\nsamples = 295\nvalue = [287, 8]'),
Text(0.5941558441558441, 0.5277777777777778, 'x[28] <= 2.58\ngini = 0.159\nsamples = 46\nvalue = [42, 4]'),
Text(0.5876623376623377, 0.4722222222222222, 'x[23] <= 2.085\ngini = 0.124\nsamples = 45\nvalue = [42, 3]'),
Text(0.5714285714285714, 0.4166666666666667, 'x[11] <= -1.729\ngini = 0.089\nsamples = 43\nvalue = [41, 2]'),
Text(0.5584415584415584, 0.3611111111111111, 'x[26] <= 0.733\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
Text(0.551948051948052, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.564935064935065, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.5844155844155844, 0.3611111111111111, 'x[13] <= -1.609\ngini = 0.048\nsamples = 41\nvalue = [40, 1]'),
Text(0.577922077922078, 0.3055555555555556, 'x[8] <= -1.118\ngini = 0.375\nsamples = 4\nvalue = [3, 1]'),
Text(0.5714285714285714, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
```

```
Text(0.5714285714285714, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.5844155844155844, 0.25, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
Text(0.5909090909090909, 0.3055555555555556, 'gini = 0.0\nsamples = 37\nvalue = [37, 0]'),
Text(0.6038961038961039, 0.4166666666666667, 'x[0] <= 1.541\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
Text(0.5974025974025974, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6103896103896104, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.6006493506493507, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6396103896103896, 0.5277777777777778, 'x[16] <= -0.943\ngini = 0.032\nsamples = 249\nvalue = [245, 4]'),
Text(0.6233766233766234, 0.4722222222222222, 'x[25] <= -0.37\ngini = 0.32\nsamples = 5\nvalue = [4, 1]'),
Text(0.6168831168831169, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6298701298701299, 0.4166666666666667, 'gini = 0.0\nsamples = 4\nvalue = [4, 0]'),
Text(0.6558441558441559, 0.4722222222222222, 'x[17] <= 1.198\ngini = 0.024\nsamples = 244\nvalue = [241, 3]'),
Text(0.6428571428571429, 0.4166666666666667, 'x[0] <= 1.268\ngini = 0.01\nsamples = 208\nvalue = [207, 1]'),
Text(0.6363636363636364, 0.3611111111111111, 'gini = 0.0\nsamples = 185\nvalue = [185, 0]'),
Text(0.6493506493506493, 0.3611111111111111, 'x[23] <= -0.036\ngini = 0.083\nsamples = 23\nvalue = [22, 1]'),
Text(0.6428571428571429, 0.3055555555555556, 'x[4] <= 0.573\ngini = 0.5\nsamples = 2\nvalue = [1, 1]'),
Text(0.6363636363636364, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6493506493506493, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.6558441558441559, 0.3055555555555556, 'gini = 0.0\nsamples = 21\nvalue = [21, 0]'),
Text(0.6688311688311688, 0.4166666666666667, 'x[17] <= 1.207\ngini = 0.105\nsamples = 36\nvalue = [34, 2]'),
Text(0.6623376623376623, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6753246753246753, 0.3611111111111111, 'x[10] <= -1.372\ngini = 0.056\nsamples = 35\nvalue = [34, 1]'),
Text(0.6688311688311688, 0.3055555555555556, 'x[26] <= 0.162\ngini = 0.444\nsamples = 3\nvalue = [2, 1]'),
Text(0.6623376623376623, 0.25, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.6753246753246753, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6818181818181818, 0.3055555555555556, 'gini = 0.0\nsamples = 32\nvalue = [32, 0]'),
Text(0.6363636363636364, 0.6388888888888888, 'x[9] <= -0.204\ngini = 0.408\nsamples = 7\nvalue = [5, 2]'),
Text(0.6298701298701299, 0.5833333333333334, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.6428571428571429, 0.5833333333333334, 'gini = 0.0\nsamples = 5\nvalue = [5, 0]'),
Text(0.484476461038961, 0.75, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.8251318993506493, 0.8611111111111112, 'x[16] <= -0.533\ngini = 0.385\nsamples = 300\nvalue = [222, 78]'),
Text(0.737114448051948, 0.8055555555555556, 'x[11] <= 1.083\ngini = 0.5\nsamples = 96\nvalue = [49, 47]'),
Text(0.7132711038961039, 0.75, 'x[8] <= -1.118\ngini = 0.498\nsamples = 86\nvalue = [40, 46]'),
Text(0.685064935064935, 0.6944444444444444, 'x[10] <= -1.102\ngini = 0.245\nsamples = 14\nvalue = [2, 12]'),
Text(0.6785714285714286, 0.6388888888888888, 'x[18] <= 0.924\ngini = 0.444\nsamples = 3\nvalue = [2, 1]'),
Text(0.672077922077922, 0.5833333333333334, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.685064935064935, 0.5833333333333334, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6915584415584416, 0.6388888888888888, 'gini = 0.0\nsamples = 11\nvalue = [0, 11]'),
Text(0.7414772727272727, 0.6944444444444444, 'x[0] <= -1.141\ngini = 0.498\nsamples = 72\nvalue = [38, 34]'),
Text(0.7045454545454546, 0.6388888888888888, 'x[28] <= -0.214\ngini = 0.32\nsamples = 10\nvalue = [2, 8]'),
Text(0.698051948051948, 0.5833333333333334, 'gini = 0.0\nsamples = 7\nvalue = [0, 7]'),
Text(0.711038961038961, 0.5833333333333334, 'x[13] <= -0.186\ngini = 0.444\nsamples = 3\nvalue = [2, 1]'),
Text(0.7045454545454546, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.7175324675324676, 0.5277777777777778, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.7784090909090909, 0.6388888888888888, 'x[20] <= 1.719\ngini = 0.487\nsamples = 62\nvalue = [36, 26]'),
Text(0.7516233766233766, 0.5833333333333334, 'x[28] <= 1.183\ngini = 0.46\nsamples = 53\nvalue = [34, 19]'),
Text(0.7305194805194806, 0.5277777777777778, 'x[18] <= 0.724\ngini = 0.423\nsamples = 46\nvalue = [32, 14]'),
Text(0.7077922077922078, 0.4722222222222222, 'x[7] <= -1.093\ngini = 0.334\nsamples = 33\nvalue = [26, 7]'),
Text(0.6948051948051948, 0.4166666666666667, 'x[23] <= -0.807\ngini = 0.49\nsamples = 7\nvalue = [3, 4]'),
Text(0.6883116883116883, 0.3611111111111111, 'gini = 0.0\nsamples = 4\nvalue = [0, 4]'),
Text(0.7012987012987013, 0.3611111111111111, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
Text(0.7207792207792207, 0.4166666666666667, 'x[20] <= -1.014\ngini = 0.204\nsamples = 26\nvalue = [23, 3]'),
Text(0.7142857142857143, 0.3611111111111111, 'x[3] <= -0.456\ngini = 0.49\nsamples = 7\nvalue = [4, 3]'),
Text(0.7077922077922078, 0.3055555555555556, 'x[7] <= 0.471\ngini = 0.32\nsamples = 5\nvalue = [4, 1]'),
Text(0.7012987012987013, 0.25, 'gini = 0.0\nsamples = 4\nvalue = [4, 0]'),
Text(0.7142857142857143, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.7207792207792207, 0.3055555555555556, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.7272727272727273, 0.3611111111111111, 'gini = 0.0\nsamples = 19\nvalue = [19, 0]'),
Text(0.7532467532467533, 0.4722222222222222, 'x[4] <= 0.573\ngini = 0.497\nsamples = 13\nvalue = [6, 7]'),
Text(0.7467532467532467, 0.4166666666666667, 'x[28] <= 0.097\ngini = 0.346\nsamples = 9\nvalue = [2, 7]'),
Text(0.7402597402597403, 0.3611111111111111, 'x[7] <= 1.443\ngini = 0.219\nsamples = 8\nvalue = [1, 7]'),
Text(0.7337662337662337, 0.3055555555555556, 'gini = 0.0\nsamples = 7\nvalue = [0, 7]'),
Text(0.7467532467532467, 0.3055555555555556, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.7532467532467533, 0.3611111111111111, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.7597402597402597, 0.4166666666666667, 'gini = 0.0\nsamples = 4\nvalue = [4, 0]'),
Text(0.7727272727272727, 0.5277777777777778, 'x[14] <= -1.114\ngini = 0.408\nsamples = 7\nvalue = [2, 5]'),
Text(0.7662337662337663, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.7792207792207793, 0.4722222222222222, 'gini = 0.0\nsamples = 5\nvalue = [0, 5]'),
Text(0.8051948051948052, 0.5833333333333334, 'x[2] <= 0.454\ngini = 0.346\nsamples = 9\nvalue = [2, 7]'),
Text(0.7987012987012987, 0.5277777777777778, 'x[16] <= -0.548\ngini = 0.219\nsamples = 8\nvalue = [1, 7]'),
Text(0.7922077922077922, 0.4722222222222222, 'gini = 0.0\nsamples = 7\nvalue = [0, 7]'),
Text(0.8051948051948052, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.8116883116883117, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.7609577922077922, 0.75, 'x[10] <= -0.364\ngini = 0.18\nsamples = 10\nvalue = [9, 1]'),
Text(0.7544642857142857, 0.6944444444444444, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.7674512987012987, 0.6944444444444444, 'gini = 0.0\nsamples = 9\nvalue = [9, 0]'),
Text(0.9131493506493507, 0.8055555555555556, 'x[15] <= 0.552\ngini = 0.258\nsamples = 204\nvalue = [173, 31]'),
Text(0.862012987012987, 0.75, 'x[16] <= 2.837\ngini = 0.138\nsamples = 147\nvalue = [136, 11]'),
Text(0.8555194805194806, 0.6944444444444444, 'x[3] <= 0.655\ngini = 0.128\nsamples = 146\nvalue = [136, 10]'),
Text(0.8376623376623377, 0.6388888888888888, 'x[26] <= -0.736\ngini = 0.038\nsamples = 104\nvalue = [102, 2]'),
Text(0.8311688311688312, 0.5833333333333334, 'x[10] <= -1.102\ngini = 0.32\nsamples = 10\nvalue = [8, 2]'),
Text(0.8246753246753247, 0.5277777777777778, 'x[0] <= 0.501\ngini = 0.444\nsamples = 3\nvalue = [1, 2]'),
Text(0.8181818181818182, 0.4722222222222222, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.8311688311688312, 0.4722222222222222, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.8376623376623377, 0.5277777777777778, 'gini = 0.0\nsamples = 7\nvalue = [7, 0]'),
Text(0.8441558441558441, 0.5833333333333334, 'gini = 0.0\nsamples = 94\nvalue = [94, 0]'),
Text(0.8733766233766234, 0.6388888888888888, 'x[8] <= -1.118\ngini = 0.308\nsamples = 42\nvalue = [34, 8]'),
Text(0.8571428571428571, 0.5833333333333334, 'x[17] <= -1.039\ngini = 0.375\nsamples = 4\nvalue = [1, 3]'),
Text(0.8506493506493507, 0.5277777777777778, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.8636363636363636, 0.5277777777777778, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
Text(0.8896103896103896, 0.5833333333333334, 'x[0] <= -0.265\ngini = 0.229\nsamples = 38\nvalue = [33, 5]'),
```

```
        Text(0.8766233766233766, 0.5277777777777778, 'x[8] <= -0.203\ngini = 0.5\nsamples = 6\nvalue = [3, 3]'),
        Text(0.8701298701298701, 0.4722222222222222, 'x[27] <= 1.593\ngini = 0.375\nsamples = 4\nvalue = [1, 3]'),
        Text(0.8636363636363636, 0.4166666666666667, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
        Text(0.8766233766233766, 0.4166666666666667, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]')]
```

```
# Prepruning
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']

}
```

```
        Text(0.9220779220779222, 0.9583333333333334, 'x[10] <= -0.969\ngini = 0.444\nsamples = 3\nvalue = [1, 2]'),
```

```
dtc_cv=GridSearchCV(estimator=dt_model,param_grid=parameter,cv=5,scoring="accuracy")
```

```
        Text(0.9350649350649351, 0.9583333333333334, 'gini = 0.0\nsamples = 23\nvalue = [23, 0]')]
```

```
dtc_cv.fit(x_train,y_train) # fitting the data
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/tree/_classes.py:269: FutureWarning: `max_features='auto'` has been deprecated in 1
  warnings.warn(
```
(repeated 30 times)

```
dtc_cv.best_params_
```

```
{'criterion': 'entropy',
 'max_depth': 3,
 'max_features': 'auto',
 'splitter': 'best'}
```

```
pred=dtc_cv.predict(x_test)
```

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

```
              precision    recall  f1-score   support

           0       0.83      1.00      0.91       245
           1       0.00      0.00      0.00        49

    accuracy                           0.83       294
   macro avg       0.42      0.50      0.45       294
weighted avg       0.69      0.83      0.76       294
```

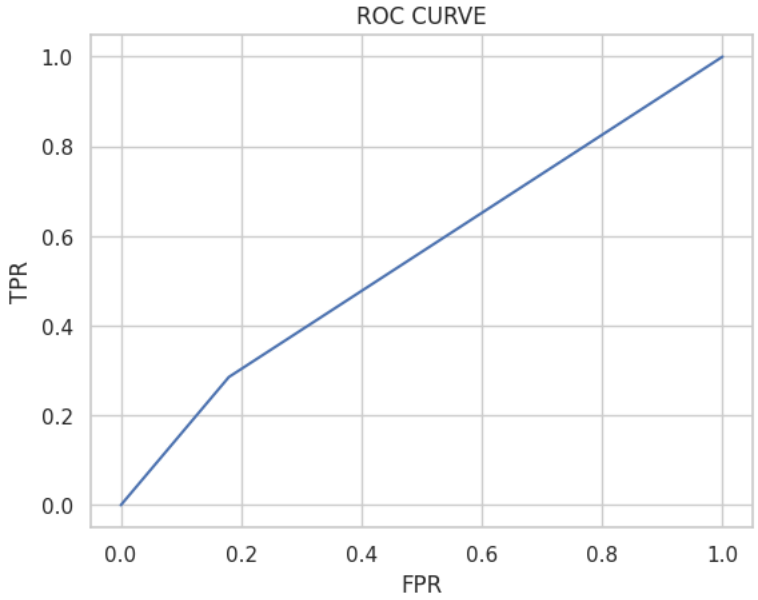
```
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.py:1344: UndefinedMetricWarning: Precision and F-score are
  _warn_prf(average, modifier, msg_start, len(result))
```
(repeated 3 times)

```
accuracy_score(y_test,pred)
```

```
0.8333333333333334
```

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

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



Random Forest Regressor

```
from sklearn.ensemble import RandomForestClassifier
rfc=RandomForestClassifier()
```

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

```
rfc_cv= GridSearchCV(rfc,param_grid=forest_params,cv=10,scoring="accuracy")
```

```
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.
```

```
pred_rf=rfc_cv.predict(x_test)
```

Evaluation Metrics

```
50 fits failed with the following error:
```

```
accuracy_score(y_test,pred_rf)
```

```
0.8469387755102041
```

```
    self._validate_params()
```

```
confusion_matrix(y_test,pred_rf)
```

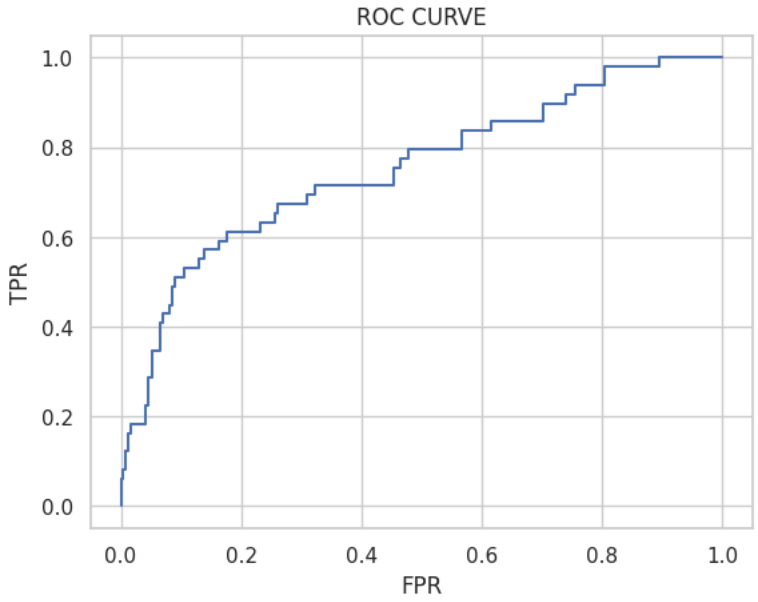
```
array([[240,   5],
       [ 40,   9]])
```

```
print(classification_report(y_test,pred_rf))
```

```
              precision    recall  f1-score   support

           0       0.86      0.98      0.91       245
           1       0.64      0.18      0.29        49

    accuracy                           0.85       294
   macro avg       0.75      0.58      0.60       294
weighted avg       0.82      0.85      0.81       294
```

```
 0.85796031 0.85541069 0.85287556 0.85712009]
```

```
probability = rfc_cv.predict_proba(x_test)[:,1]
fpr, tpr, thresholds = roc_curve(y_test, probability)
plt.plot(fpr, tpr)
plt.xlabel('FPR')
plt.ylabel('TPR')
plt.title('ROC CURVE')
plt.show()
```




```
''' THE END '''
```

```
' THE END '
```

```
''' Thank You '''
# Rayidi Abhiram
# 21bce9261
```

```
' Thank You '
```