1. Business Understanding

1. What is the business problem that you are trying to solve?

Solution: Predict employee attrition in the organization and identify the key factors influencing attrition. Also, segment the employees into different groups based on their characteristics and work behavior.”

1. What data do you need to answer the above problem?

Solution: EmployeeID: A unique identifier for each employee.

EmployeeLocation: The location where the employee works.

Age: Age of the employee.

Department: The department in which the employee works (e.g., Sales, HR, Engineering, Marketing, Finance).

Gender: Gender of the employee.

MaritalStatus: Marital status of the employee (e.g., Married, Single, Divorced).

Education: Level of education of the employee (e.g., High School, Bachelor, Master, PhD).

JobRole: Role or position of the employee (e.g., Manager, Team Leader, Engineer, Sales Executive, HR).

JobLevel: Level of the employee's job within the organization.

MonthlyIncome: Monthly income of the employee.

NumCompaniesWorked: Number of companies the employee has worked for previously.

TotalWorkingYears: Total number of years the employee has been employed.

TrainingTimesLastYear: Number of training sessions the employee attended last year.

YearsAtCompany: Number of years the employee has been with the current company.

YearsInCurrentRole: Number of years the employee has been in their current role.

YearsSinceLastPromotion: Number of years since the employee's last promotion.

YearsWithCurrManager: Number of years the employee has been with their current manager.

Attrition: Whether the employee has left the company (Yes/No).

PerformanceRating: Performance rating of the employee (e.g., Low, Good, Excellent).

JobSatisfaction: Level of job satisfaction reported by the employee (e.g., Low, Medium, High, Very High).

OverTime: Whether the employee works overtime (Yes/No)

1. What are the different sources of data?

Solution: HR Systems: Most of the employee-specific information such as EmployeeID, Age, Gender, MaritalStatus, Education, JobRole, JobLevel, MonthlyIncome, TotalWorkingYears, YearsAtCompany, YearsInCurrentRole,

YearsSinceLastPromotion, YearsWithCurrManager, Attrition, PerformanceRating, JobSatisfaction, and Overtime are typically stored in Human Resource Information Systems (HRIS) or Human Resource Management Systems (HRMS).

Employee Surveys: Data related to employee satisfaction, such as JobSatisfaction, can be collected through surveys conducted within the organization.

Training Records: Information about the number of training sessions attended by employees (TrainingTimesLastYear) can be obtained from training records maintained by the HR department or training management systems.

Payroll Systems: MonthlyIncome data can be sourced from payroll systems that manage salary and compensation information for employees.

Performance Management Systems: Performance ratings (PerformanceRating) can be retrieved from performance management systems where employee performance evaluations are recorded.

Employee Exit Interviews/Records: Attrition data (Attrition) can be collected from employee exit interviews or records maintained by the HR department on employees who have left the organization.

Previous Employment Records: Number of companies worked for (NumCompaniesWorked) could be obtained from previous employment records submitted by employees during the hiring process.

Organizational Records: Information such as EmployeeLocation and Department can be sourced from organizational records or employee profiles maintained by the HR department.

1. What kind of analytics task are you performing?

Solution:

Descriptive Analytics: Describing the characteristics of the employee population, such as age distribution, gender ratio, marital status distribution, etc. Summarizing the distribution of employees across departments, job roles, and education levels. Analyzing the average monthly income and other relevant statistics.

Exploratory Data Analysis (EDA): Exploring relationships between different features through visualizations and statistical analysis. Identifying patterns and trends in employee data. Detecting outliers and anomalies in the data.

Predictive Analytics: Predicting employee attrition (binary classification task) based on the provided features using machine learning algorithms such as logistic regression, decision trees, random forests, etc. Predicting performance ratings or job satisfaction levels using regression analysis. Forecasting future employee turnover rates based on historical data.

Feature Importance Analysis: Identifying key factors influencing attrition using feature importance techniques such as random forest feature importance, permutation importance, or SHAP values. Understanding which features have the most significant impact on employee performance ratings or job satisfaction.

Segmentation Analysis: Segmenting employees into different groups based on their characteristics and work behavior using clustering algorithms such as k-means clustering or hierarchical clustering. Analyzing differences between segments in terms of attrition rates, performance ratings, job satisfaction levels, etc. Identifying common characteristics and behaviors within each segment to tailor retention strategies or improve employee satisfaction. Predictive Modeling for Individual Features:

Building predictive models to estimate monthly income, total working years, years at the company, etc., based on other available features. This could involve regression analysis or other predictive modeling techniques.

2. Data Acquisition

For the problem identified, find an appropriate data set (Your data set must be unique with minimum 20 features and 10k rows) from any public data source.

2.1 Download the data directly

```
##---------Type the code below this line---------- ##
import os
import pandas as pd

os.environ['KAGGLE_USERNAME'] = 'poojaraninayak'
os.environ['KAGGLE_KEY'] = '033e84e15bc66681c67725e59e64e7d3'

!kaggle datasets download -d poojaraninayak/hr-data-set --unzip


df = pd.read_csv('HR_Dataset.csv')
```

```
'kaggle' is not recognized as an internal or external command,
operable program or batch file.
```

2.2 Code for converting the above downloaded data into a dataframe

```
##---------Type the code below this line---------- ##
import pandas as pd
import numpy as np


df = pd.read_csv('HR_Dataset.csv')
df_copy = df.copy()
df
```

```
      EmployeeID EmployeeLocation   Age   Department  Gender
MaritalStatus  \
0              1         Bangalore  25.0        Sales  Female
Single
1              2     Andrapradesh  45.0    Marketing    Male
Single
2              3     Andrapradesh  37.0           HR    Male
Married
3              4     Andrapradesh  35.0  Engineering  Female
Divorced
4              5            Orissa  51.0           HR  Female
Divorced
...          ...               ...   ...          ...     ...
...
9995        9996            Orissa  45.0  Engineering    Male
Single
9996        9997            Mumbai  42.0      Finance    Male
Divorced
9997        9998            Mumbai  42.0        Sales  Female
Single
9998        9999         Tamilnadu  35.0        Sales  Female
```

```
Divorced
9999       10000     Andrapradesh  53.0     Marketing  Female
Single

        Education           JobRole  JobLevel  MonthlyIncome in $  ...
\
0           Master           Manager         1              5505.0  ...

1     High School  Sales  Executive         4              8377.0  ...

2           Master  Sales  Executive         3             17854.0  ...

3         Bachelor           Manager         1             18881.0  ...

4         Bachelor          Engineer         1             15019.0  ...

...            ...               ...       ...                 ...  ...

9995      Bachelor           Manager         4              6992.0  ...

9996  High School          Engineer         3             17241.0  ...

9997      Bachelor           Manager         1              5015.0  ...

9998        Master           Manager         1              8168.0  ...

9999      Bachelor          Engineer         2              7144.0  ...


      TotalWorkingYears  TrainingTimesLastYear  YearsAtCompany  \
0                   8.0                    0.0             9.0
1                  15.0                    1.0            19.0
2                   3.0                    4.0            13.0
3                  29.0                    1.0             9.0
4                   1.0                    3.0            11.0
...                 ...                    ...             ...
9995               23.0                    0.0            16.0
9996               37.0                    3.0             5.0
9997               16.0                    3.0            19.0
9998               27.0                    3.0            14.0
9999               32.0                    4.0             4.0

      YearsInCurrentRole  YearsSinceLastPromotion
YearsWithCurrManager  \
0                    5.0                      1.0
2.0
1                    1.0                      1.0
5.0
2                    1.0                      1.0
8.0
3                    0.0                      3.0
```

```
8.0
4                    2.0                          3.0
9.0
...                  ...                          ...                        ..
.
9995                 1.0                          4.0
0.0
9996                 6.0                          1.0
0.0
9997                 1.0                          3.0
8.0
9998                 3.0                          4.0
4.0
9999                 2.0                          3.0
9.0

      Attrition  PerformanceRating  JobSatisfaction  OverTime
0            No               Good           Medium       Yes
1           Yes          Excellent              Low        No
2           Yes               Good           Medium       Yes
3           Yes                Low              Low        No
4            No               Good       Very High        No
...         ...                ...              ...       ...
9995         No          Excellent       Very High        No
9996         No          Excellent       Very High        No
9997        Yes                Low           Medium        No
9998         No               Good       Very High        No
9999        Yes                Low           Medium       Yes

[10000 rows x 21 columns]
```

2.3 Confirm the data has been correctly by displaying the first 5 and last 5 records.

```
##---------Type the code below this line------------------------------ ##
df.head(5)

   EmployeeID EmployeeLocation   Age   Department  Gender
MaritalStatus  \
0           1        Bangalore  25.0        Sales  Female
Single
1           2     Andrapradesh  45.0    Marketing    Male
Single
2           3     Andrapradesh  37.0           HR    Male
Married
3           4     Andrapradesh  35.0  Engineering  Female
Divorced
4           5           Orissa  51.0           HR  Female
Divorced
```

```
     Education           JobRole  JobLevel  MonthlyIncome in $   ...  \
0       Master           Manager         1              5505.0   ...
1  High School  Sales Executive         4              8377.0   ...
2       Master  Sales Executive         3             17854.0   ...
3     Bachelor           Manager         1             18881.0   ...
4     Bachelor          Engineer         1             15019.0   ...

   TotalWorkingYears  TrainingTimesLastYear   YearsAtCompany  \
0                8.0                    0.0              9.0
1               15.0                    1.0             19.0
2                3.0                    4.0             13.0
3               29.0                    1.0              9.0
4                1.0                    3.0             11.0

   YearsInCurrentRole  YearsSinceLastPromotion
YearsWithCurrManager  \
0                 5.0                      1.0                     2.0

1                 1.0                      1.0                     5.0

2                 1.0                      1.0                     8.0

3                 0.0                      3.0                     8.0

4                 2.0                      3.0                     9.0


   Attrition PerformanceRating JobSatisfaction OverTime
0         No              Good          Medium      Yes
1        Yes         Excellent             Low       No
2        Yes              Good          Medium      Yes
3        Yes               Low             Low       No
4         No              Good       Very High       No

[5 rows x 21 columns]
```

```
df.tail(5)
```

```
      EmployeeID EmployeeLocation   Age   Department  Gender
MaritalStatus  \
9995        9996           Orissa  45.0  Engineering    Male
Single
9996        9997           Mumbai  42.0      Finance    Male
Divorced
9997        9998           Mumbai  42.0        Sales  Female
Single
9998        9999        Tamilnadu  35.0        Sales  Female
Divorced
9999       10000     Andrapradesh  53.0    Marketing  Female
Single
```

```
        Education   JobRole  JobLevel  MonthlyIncome in $  ...  \
9995     Bachelor   Manager         4              6992.0  ...
9996  High School  Engineer         3             17241.0  ...
9997     Bachelor   Manager         1              5015.0  ...
9998       Master   Manager         1              8168.0  ...
9999     Bachelor  Engineer         2              7144.0  ...

      TotalWorkingYears  TrainingTimesLastYear  YearsAtCompany  \
9995               23.0                    0.0            16.0
9996               37.0                    3.0             5.0
9997               16.0                    3.0            19.0
9998               27.0                    3.0            14.0
9999               32.0                    4.0             4.0

      YearsInCurrentRole  YearsSinceLastPromotion
YearsWithCurrManager  \
9995                 1.0                      4.0
0.0
9996                 6.0                      1.0
0.0
9997                 1.0                      3.0
8.0
9998                 3.0                      4.0
4.0
9999                 2.0                      3.0
9.0

      Attrition PerformanceRating JobSatisfaction OverTime
9995         No         Excellent       Very High       No
9996         No         Excellent       Very High       No
9997        Yes               Low          Medium       No
9998         No              Good       Very High       No
9999        Yes               Low          Medium      Yes

[5 rows x 21 columns]
```

2.4 Display the column headings, statistical information, description and statistical summary of the data.

```
##---------Type the code below this line---------- ##

#df = pd.DataFrame(df)

# Display column headings
print("Column Headings:")
print(df.columns)
```

```
Column Headings:
Index(['EmployeeID', 'EmployeeLocation', 'Age', 'Department',
'Gender',
       'MaritalStatus', 'Education', 'JobRole', 'JobLevel',
       'MonthlyIncome in $', 'NumCompaniesWorked',
'TotalWorkingYears',
       'TrainingTimesLastYear', 'YearsAtCompany',
'YearsInCurrentRole',
       'YearsSinceLastPromotion', 'YearsWithCurrManager', 'Attrition',
       'PerformanceRating', 'JobSatisfaction', 'OverTime'],
      dtype='object')
```

```
# Display statistical information
print("\nStatistical Information:")
print(df.describe())
```

```
Statistical Information:
        EmployeeID           Age       JobLevel  MonthlyIncome in $  \
count  10000.00000  9996.000000  10000.000000          9984.000000
mean    5000.50000    39.450180      2.490700         10489.239583
std     2886.89568    11.463312      1.112222          5496.488046
min        1.00000    20.000000      1.000000          1001.000000
25%     2500.75000    30.000000      2.000000          5668.500000
50%     5000.50000    39.000000      2.000000         10427.000000
75%     7500.25000    49.000000      3.000000         15337.500000
max    10000.00000    59.000000      4.000000         19999.000000

       NumCompaniesWorked  TotalWorkingYears  TrainingTimesLastYear  \
count         9999.000000        9993.000000            9993.000000
mean             4.537954          19.455819               2.017712
std              2.859120          11.493448               1.405477
min              0.000000           0.000000               0.000000
25%              2.000000          10.000000               1.000000
50%              5.000000          20.000000               2.000000
75%              7.000000          29.000000               3.000000
max              9.000000          39.000000               4.000000

       YearsAtCompany  YearsInCurrentRole  YearsSinceLastPromotion  \
count     9998.000000         9998.000000              9998.000000
mean         9.562212            4.459792                 2.018604
std          5.775837            2.859788                 1.416424
min          0.000000            0.000000                 0.000000
25%          5.000000            2.000000                 1.000000
50%         10.000000            4.000000                 2.000000
75%         15.000000            7.000000                 3.000000
max         19.000000            9.000000                 4.000000

       YearsWithCurrManager
count           9999.000000
```

```
mean                 4.498450
std                  2.867092
min                  0.000000
25%                  2.000000
50%                  5.000000
75%                  7.000000
max                  9.000000
```

```
# Display description
print("\nDescription:")
df.info()
```

```
Description:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 21 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   EmployeeID                10000 non-null  int64
 1   EmployeeLocation          9999 non-null   object
 2   Age                       9996 non-null   float64
 3   Department                9999 non-null   object
 4   Gender                    9999 non-null   object
 5   MaritalStatus             9993 non-null   object
 6   Education                 9996 non-null   object
 7   JobRole                   9991 non-null   object
 8   JobLevel                  10000 non-null  int64
 9   MonthlyIncome in $        9984 non-null   float64
 10  NumCompaniesWorked        9999 non-null   float64
 11  TotalWorkingYears         9993 non-null   float64
 12  TrainingTimesLastYear     9993 non-null   float64
 13  YearsAtCompany            9998 non-null   float64
 14  YearsInCurrentRole        9998 non-null   float64
 15  YearsSinceLastPromotion   9998 non-null   float64
 16  YearsWithCurrManager      9999 non-null   float64
 17  Attrition                 10000 non-null  object
 18  PerformanceRating         9992 non-null   object
 19  JobSatisfaction           9998 non-null   object
 20  OverTime                  9998 non-null   object
dtypes: float64(9), int64(2), object(10)
memory usage: 1.6+ MB
```

```
# Display statistical summary
print("\nStatistical Summary:")
print(df.describe(include='all'))
```

```
Statistical Summary:
         EmployeeID EmployeeLocation           Age Department Gender  \
```

```
count    10000.00000              9999   9996.000000        9999    9999
unique           NaN                 5           NaN           5       3
top              NaN            Orissa           NaN          HR    Male
freq             NaN              2041           NaN        2047    5054
mean      5000.50000               NaN     39.450180         NaN     NaN
std       2886.89568               NaN     11.463312         NaN     NaN
min          1.00000               NaN     20.000000         NaN     NaN
25%       2500.75000               NaN     30.000000         NaN     NaN
50%       5000.50000               NaN     39.000000         NaN     NaN
75%       7500.25000               NaN     49.000000         NaN     NaN
max      10000.00000               NaN     59.000000         NaN     NaN

        MaritalStatus     Education JobRole       JobLevel  MonthlyIncome
in $  \
count            9993          9996    9991  10000.000000
9984.000000
unique              3             4       5           NaN
NaN
top           Married  High School      HR           NaN
NaN
freq             3389          2600    2060           NaN
NaN
mean              NaN           NaN     NaN      2.490700
10489.239583
std               NaN           NaN     NaN      1.112222
5496.488046
min               NaN           NaN     NaN      1.000000
1001.000000
25%               NaN           NaN     NaN      2.000000
5668.500000
50%               NaN           NaN     NaN      2.000000
10427.000000
75%               NaN           NaN     NaN      3.000000
15337.500000
max               NaN           NaN     NaN      4.000000
19999.000000

        ...  TotalWorkingYears  TrainingTimesLastYear  YearsAtCompany
\
count   ...        9993.000000            9993.000000     9998.000000

unique  ...                NaN                    NaN             NaN

top     ...                NaN                    NaN             NaN

freq    ...                NaN                    NaN             NaN

mean    ...          19.455819               2.017712        9.562212

std     ...          11.493448               1.405477        5.775837
```

```
min      ...            0.000000                0.000000         0.000000
25%      ...           10.000000                1.000000         5.000000
50%      ...           20.000000                2.000000        10.000000
75%      ...           29.000000                3.000000        15.000000
max      ...           39.000000                4.000000        19.000000


        YearsInCurrentRole  YearsSinceLastPromotion
YearsWithCurrManager  \
count          9998.000000              9998.000000
9999.000000
unique                 NaN                      NaN
NaN
top                    NaN                      NaN
NaN
freq                   NaN                      NaN
NaN
mean              4.459792                 2.018604
4.498450
std               2.859788                 1.416424
2.867092
min               0.000000                 0.000000
0.000000
25%               2.000000                 1.000000
2.000000
50%               4.000000                 2.000000
5.000000
75%               7.000000                 3.000000
7.000000
max               9.000000                 4.000000
9.000000

        Attrition  PerformanceRating  JobSatisfaction OverTime
count       10000               9992             9998     9998
unique          2                  3                4        2
top            No          Excellent           Medium      Yes
freq         7438               3341             2605     5019
mean          NaN                NaN              NaN      NaN
std           NaN                NaN              NaN      NaN
min           NaN                NaN              NaN      NaN
25%           NaN                NaN              NaN      NaN
50%           NaN                NaN              NaN      NaN
75%           NaN                NaN              NaN      NaN
max           NaN                NaN              NaN      NaN
```

```
[11 rows x 21 columns]
```

2.5 Write your observations from the above.

1. Size of the dataset What type of data attributes are there?
2. What type of data attributes are there?
3. Is there any null data that has to be cleaned?

```
# Determine the size of the dataset
dataset_size = df.shape

# Display the size of the dataset
print("Size of the dataset:", dataset_size)
```

```
Size of the dataset: (10000, 21)
```

1. What type of data attributes are there?

Categorical Attributes: EmployeeLocation Department Gender MaritalStatus Education JobRole Attrition PerformanceRating JobSatisfaction OverTime

Numerical Attributes: EmployeeID Age JobLevel MonthlyIncome NumCompaniesWorked TotalWorkingYears TrainingTimesLastYear YearsAtCompany YearsInCurrentRole YearsSinceLastPromotion YearsWithCurrManager

3. Is there any null data that has to be cleaned?

Yes, we have null values and that has to be cleaned

3. Data Preparation

3.1 Check for

- duplicate data
- missing data
- data inconsistencies

```
##---------Type the code below this line----------##
#Duplicate Data

duplicate_rows = df.duplicated().sum()
duplicate_rows
```

```
0
```

```
#Find missingdata
null_values = df.isnull().sum()

# Display the null values
print("Null values in the DataFrame:")
print(null_values)
```

```
Null values in the DataFrame:
EmployeeID                  0
EmployeeLocation            1
Age                         4
Department                  1
Gender                      1
MaritalStatus               7
Education                   4
JobRole                     9
JobLevel                    0
MonthlyIncome in $         16
NumCompaniesWorked          1
TotalWorkingYears           7
TrainingTimesLastYear       7
YearsAtCompany              2
YearsInCurrentRole          2
YearsSinceLastPromotion     2
YearsWithCurrManager        1
Attrition                   0
PerformanceRating           8
JobSatisfaction             2
OverTime                    2
dtype: int64
```

Data inconsistencies

Manual Inspection: Manually inspect the data by scanning through the dataset to identify any obvious inconsistencies or irregularities. Look for unexpected or unusual values, patterns, or outliers.

Summary Statistics: Calculate summary statistics such as mean, median, minimum, maximum, and standard deviation for numerical columns to identify outliers or values that fall outside the expected range. For categorical columns, examine frequency counts to identify uncommon or unexpected categories.

Data Profiling: Use data profiling techniques or tools to automatically generate statistics and visualizations that provide insights into the data distribution, missing values, unique values, and outliers.

Data Validation Rules: Define data validation rules or constraints based on domain knowledge or business requirements and check if the data adheres to these rules. For example, check if numerical values fall within permissible ranges, if categorical values match predefined categories, or if relationships between columns are consistent.

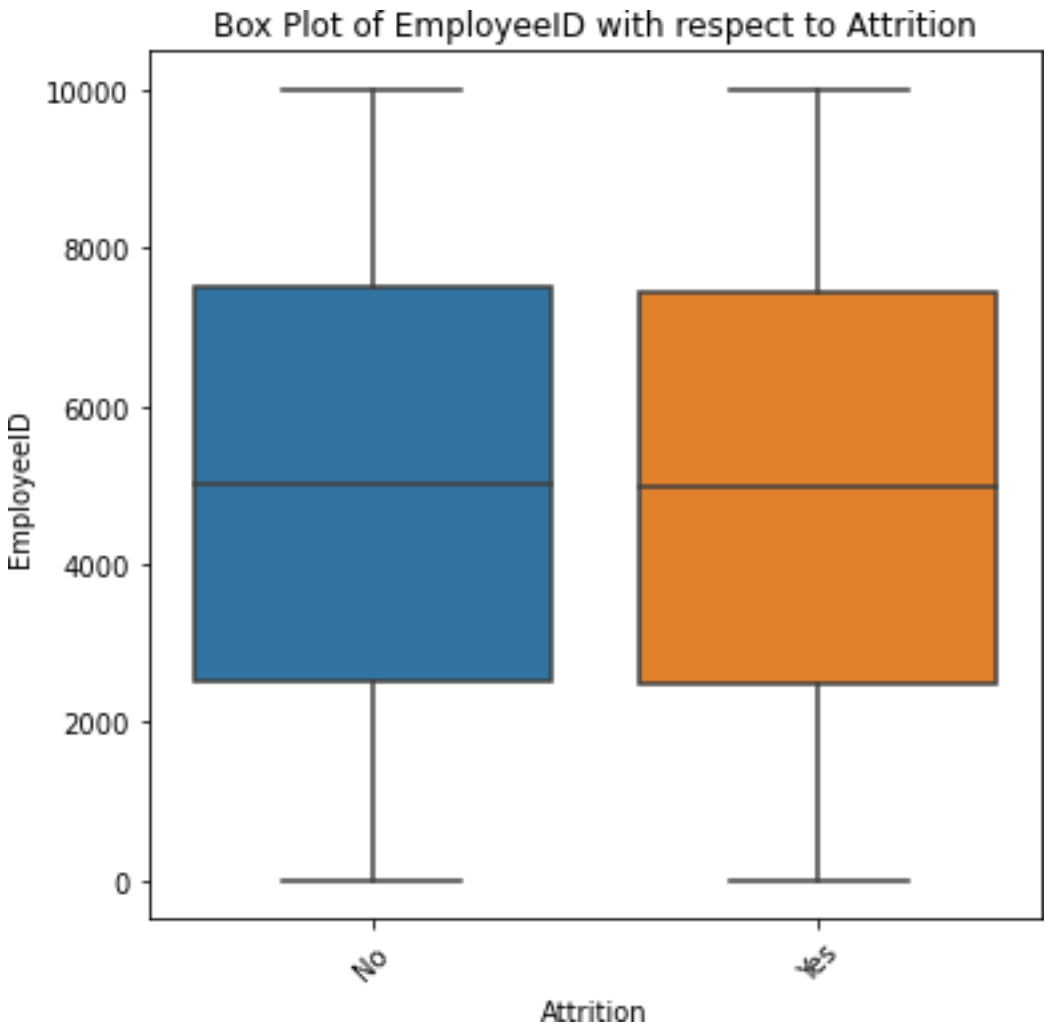
Cross-Column Validation: Perform cross-column validation to identify inconsistencies or discrepancies between related columns. For example, verify if the start date of an event is before the end date, if the total quantity matches the sum of individual quantities, or if the relationship between two columns is logically consistent.

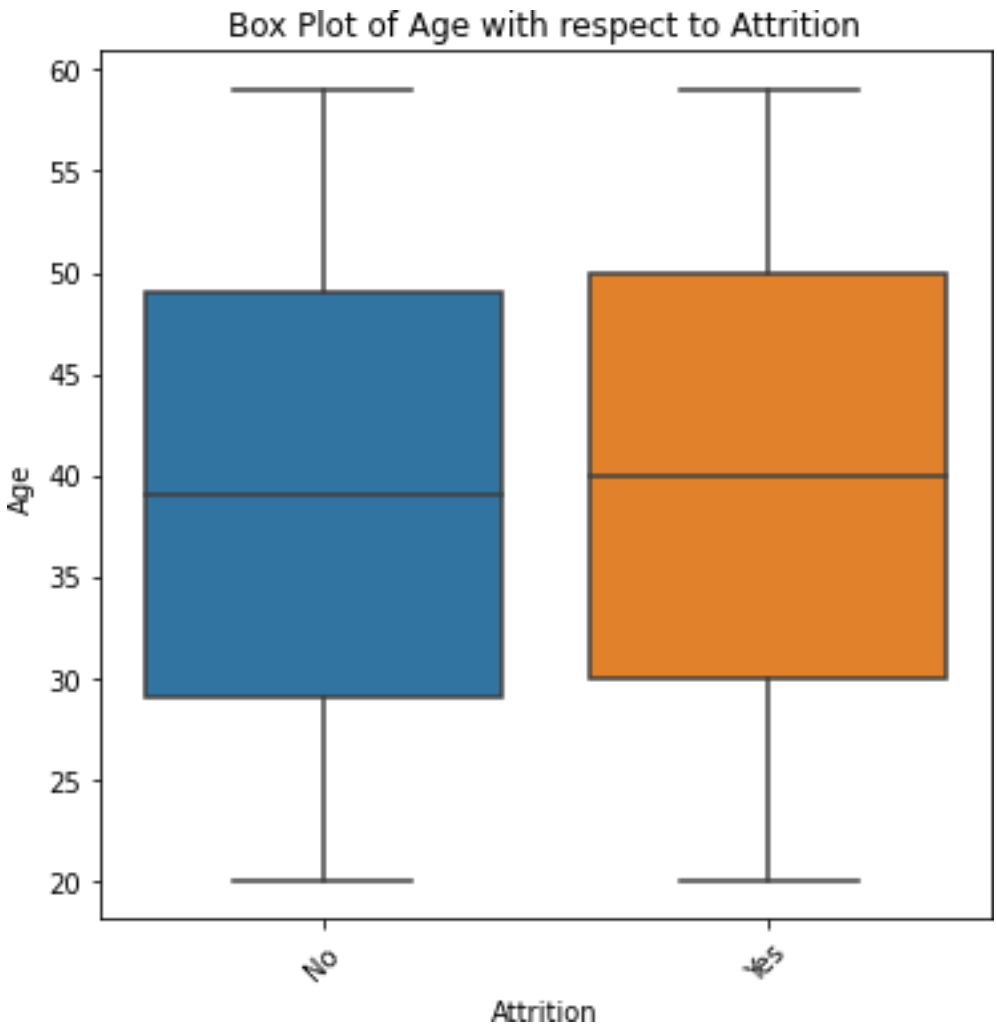
Domain Knowledge: Leverage domain knowledge or subject matter expertise to identify potential data inconsistencies based on the context of the data. Understand the business processes, data sources, and expected data behaviors to recognize anomalies or discrepancies.

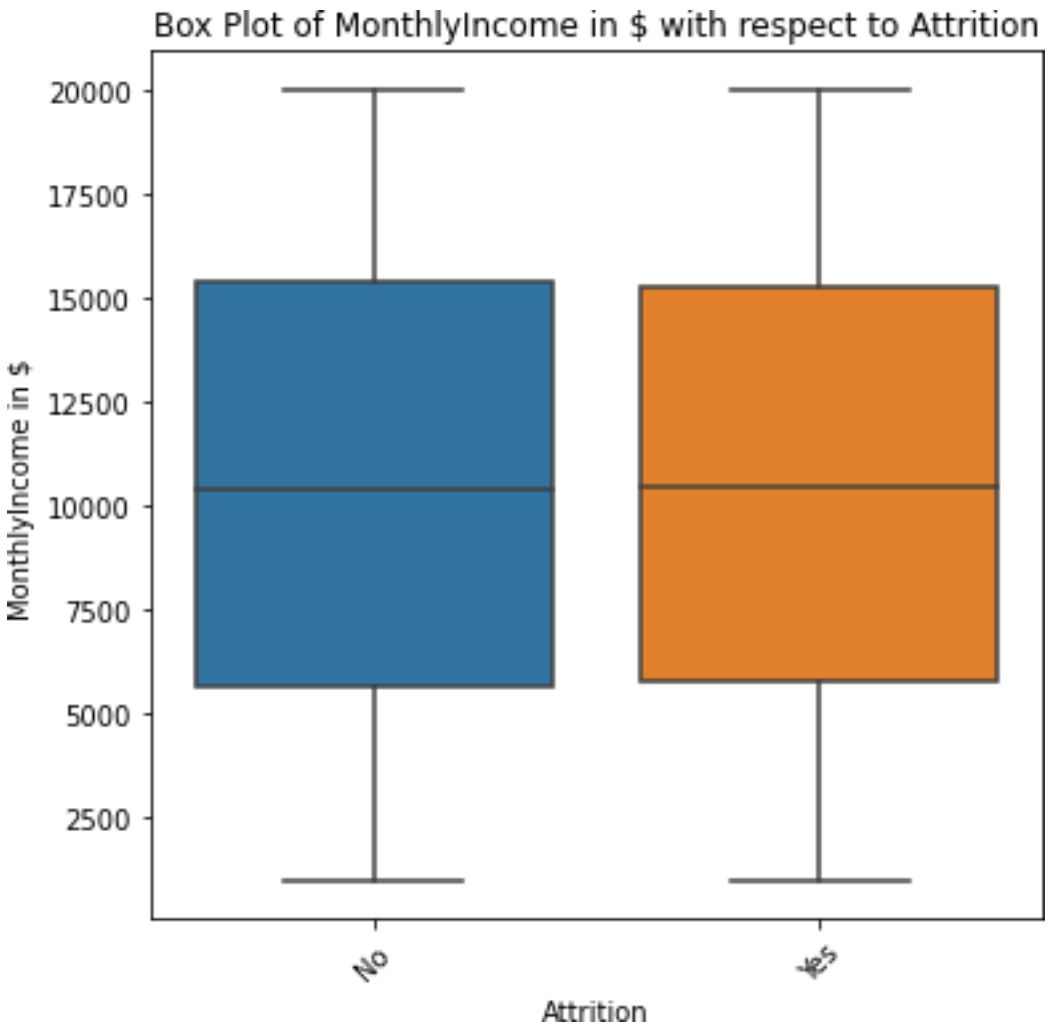
Data Quality Assessment: Conduct a data quality assessment to evaluate the accuracy, completeness, consistency, and timeliness of the data. Use data quality metrics and frameworks to systematically evaluate data inconsistencies and prioritize areas for improvement.

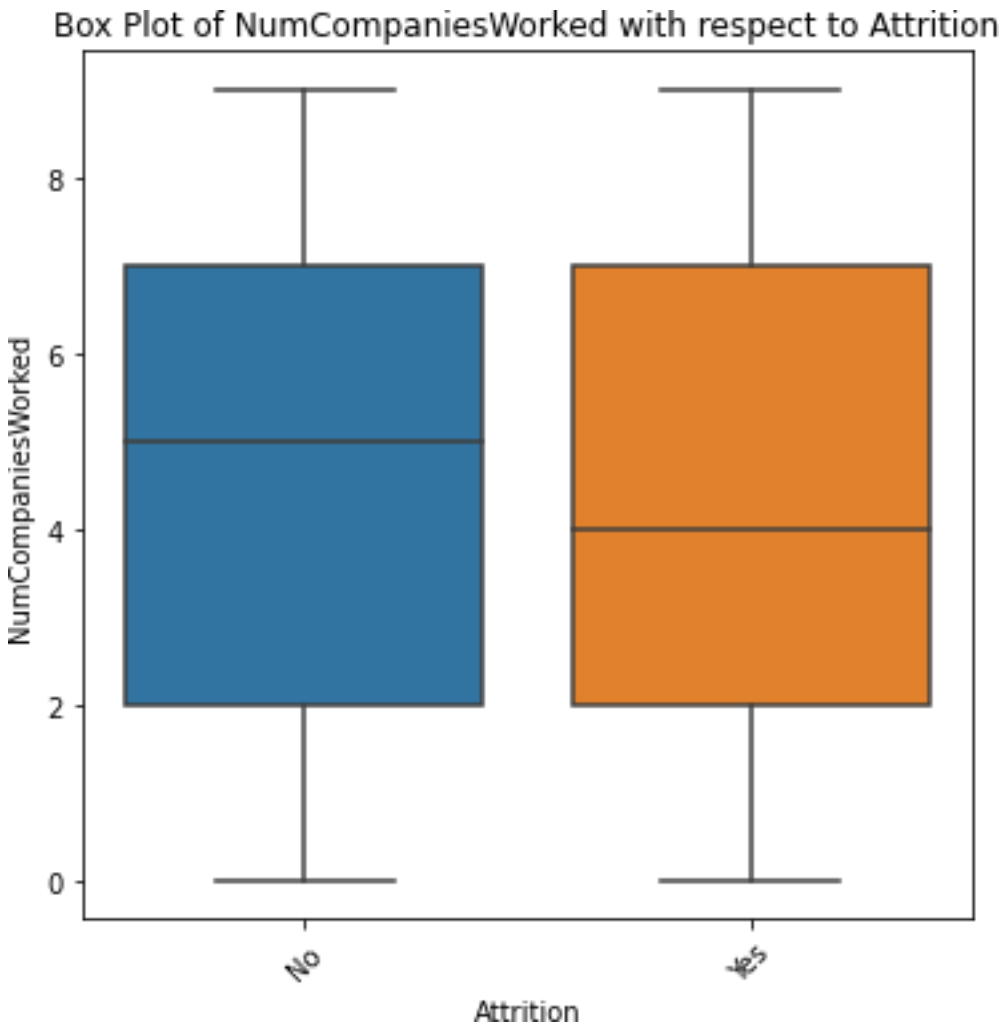
Data Visualization: Visualize the data using plots, charts, histograms, or heatmaps to identify patterns, trends, or anomalies that may indicate data inconsistencies. Explore relationships between variables visually to detect inconsistencies or unexpected patterns.

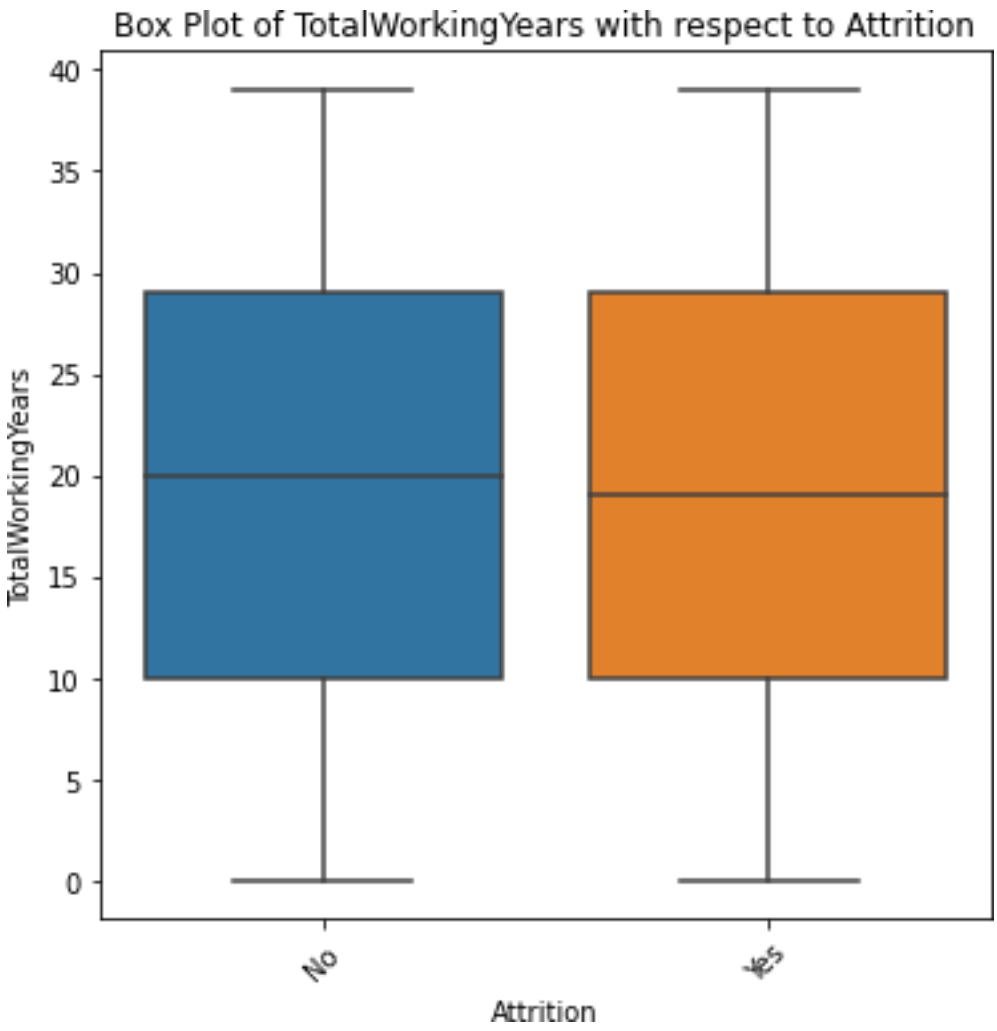
```
#Data Inconsistencies

import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd

# Assuming 'df' is your DataFrame containing the provided features

# Select only the quantitative parameters
 quantitative_features = ['EmployeeID', 'Age', 'MonthlyIncome in $',
                          'NumCompaniesWorked', 'TotalWorkingYears',
'TrainingTimesLastYear',
                          'YearsAtCompany', 'YearsInCurrentRole',
'YearsSinceLastPromotion',
                          'YearsWithCurrManager']

# Create box plots for each quantitative parameter with respect to
'Attrition'
for feature in quantitative_features:
    plt.figure(figsize=(6, 6))
    sns.boxplot(x='Attrition', y=feature, data=df)
    plt.title(f'Box Plot of {feature} with respect to Attrition')
    plt.xlabel('Attrition')
    plt.ylabel(feature)
    plt.xticks(rotation=45)  # Rotate x-axis labels for better
readability
    plt.show()
```

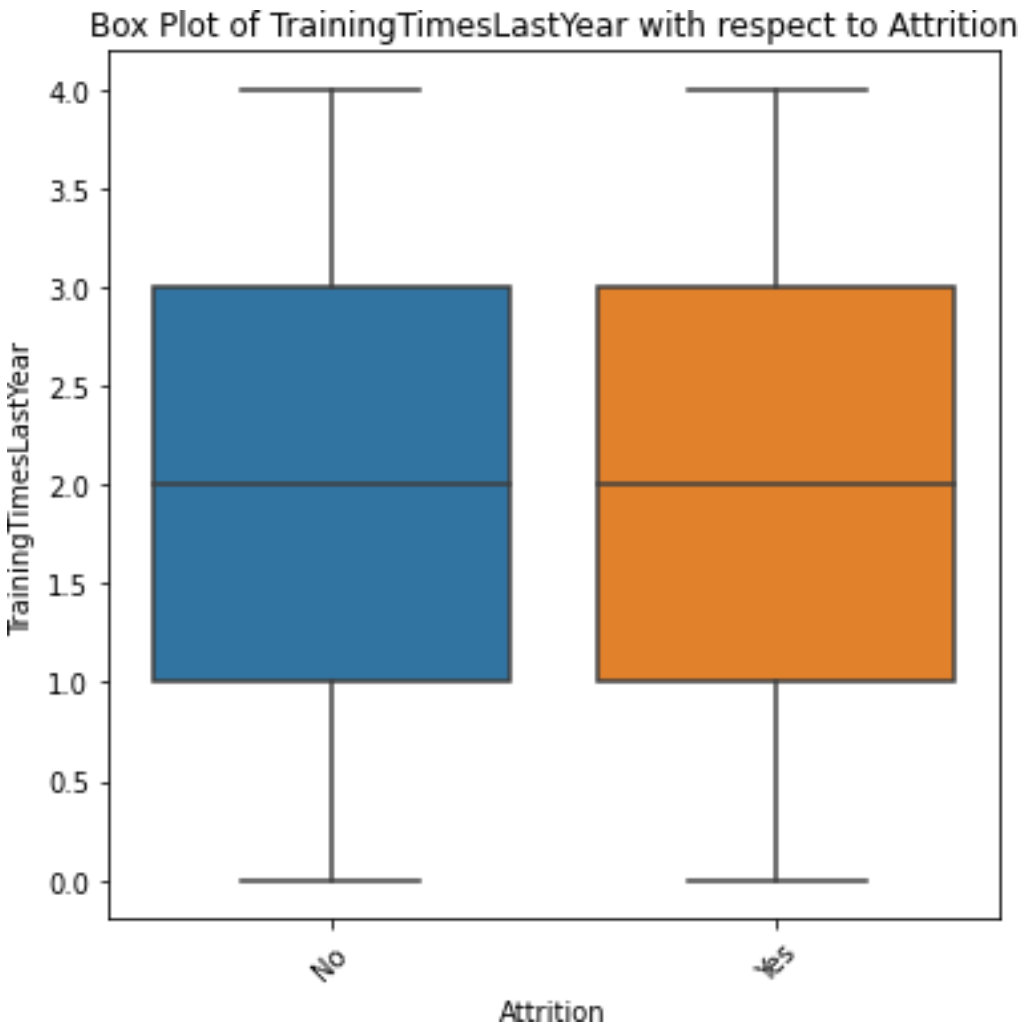

Box Plot of EmployeeID with respect to Attrition
10000
8000
6000
4000
2000
0
EmployeeID
No
Yes
Attrition

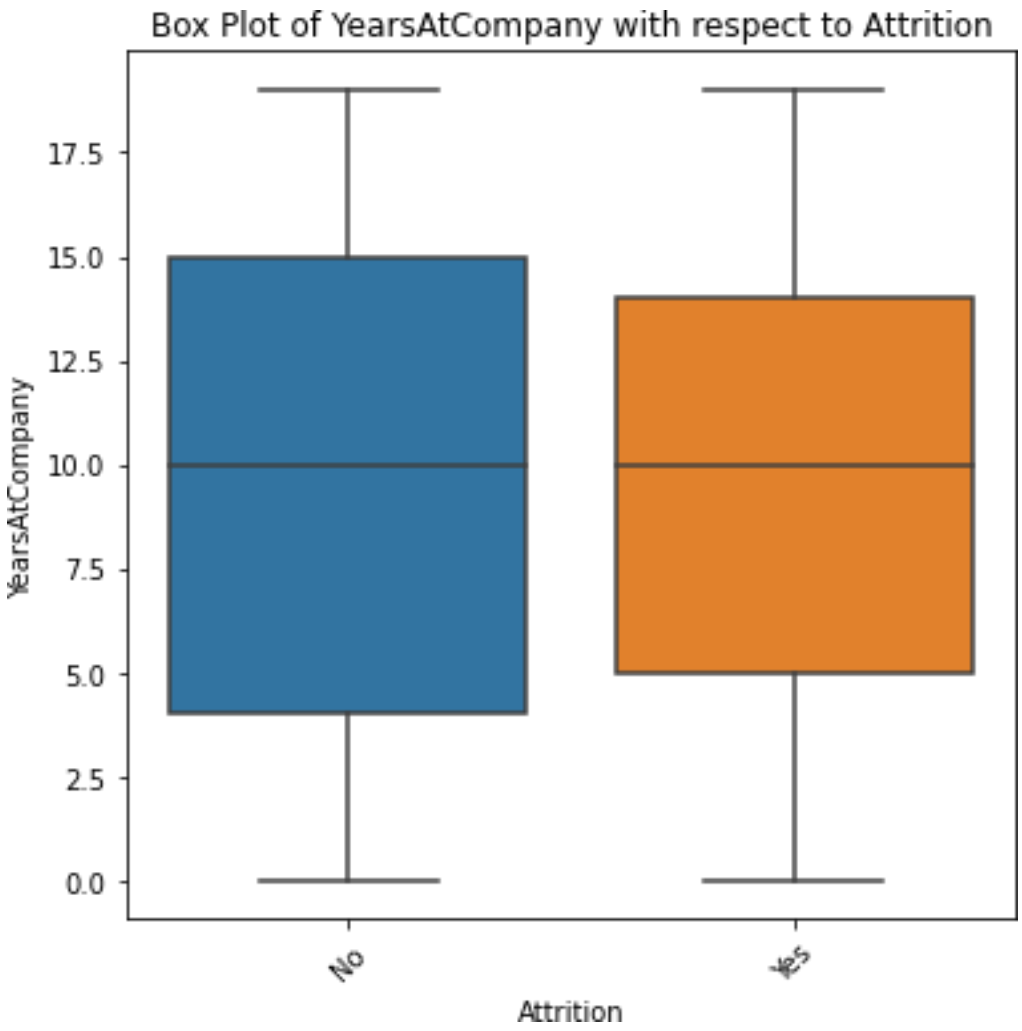

Box Plot of Age with respect to Attrition
60
55
50
45
40
35
30
25
20
Age
No
Yes
Attrition

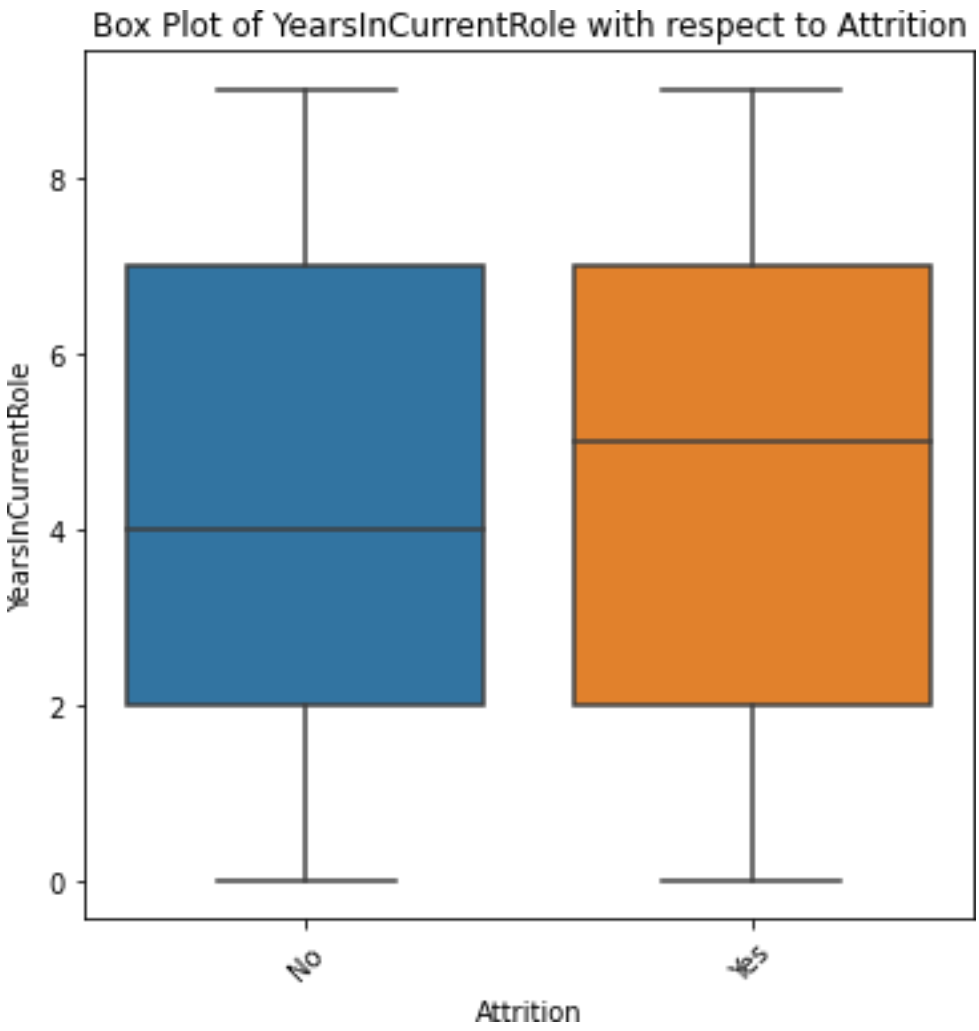

Box Plot of MonthlyIncome in $ with respect to Attrition
20000
17500
15000
12500
10000
7500
5000
2500
MonthlyIncome in $
No
Yes
Attrition

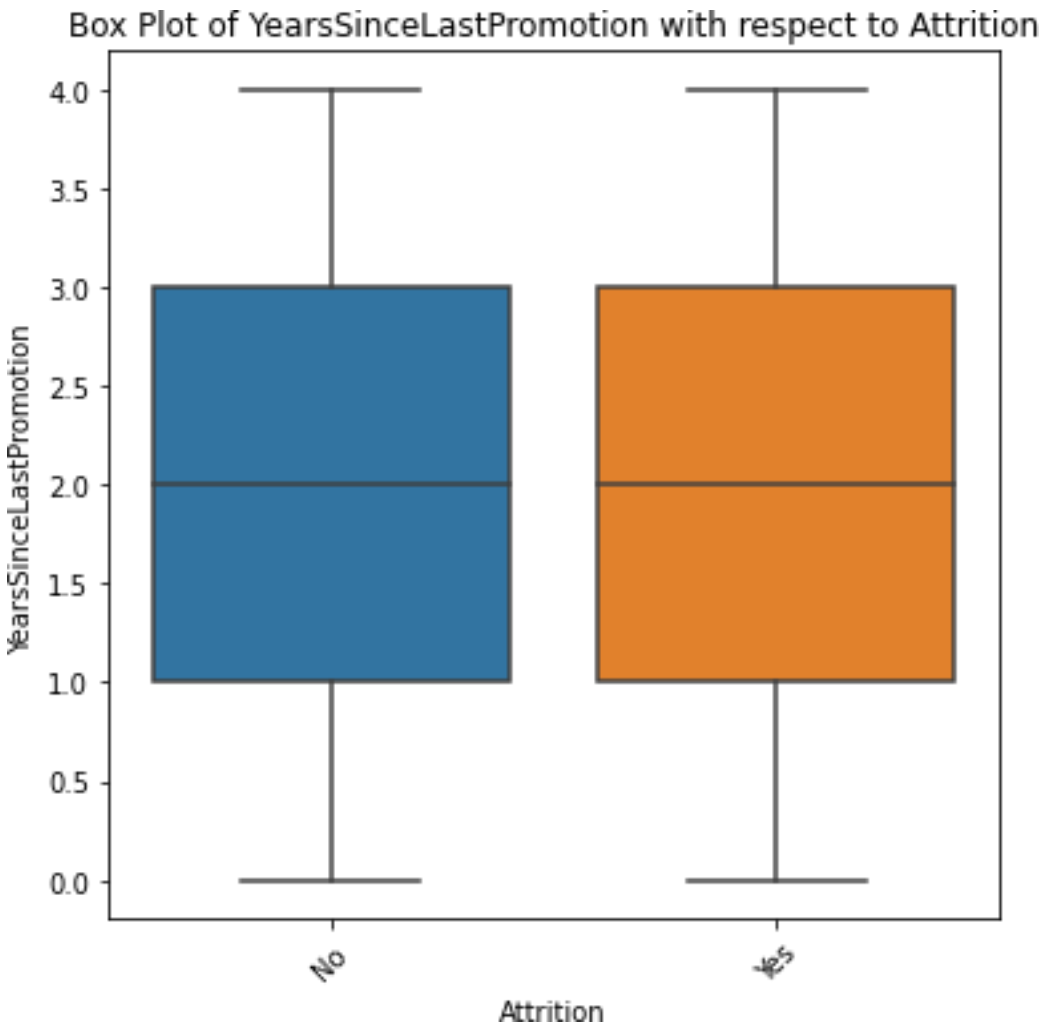

Box Plot of NumCompaniesWorked with respect to Attrition
NumCompaniesWorked
0
2
4
6
8
No
Yes
Attrition

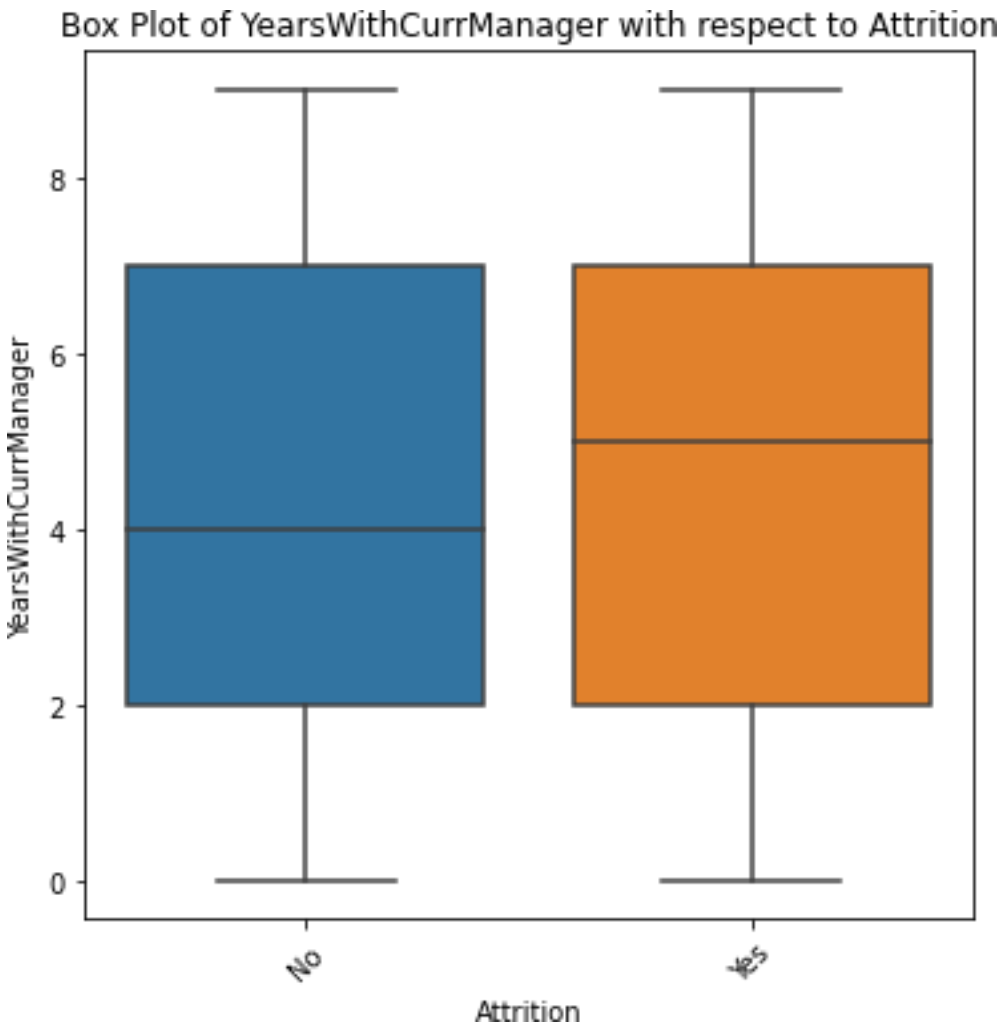

Box Plot of TotalWorkingYears with respect to Attrition
TotalWorkingYears
40
35
30
25
20
15
10
5
0
No
Yes
Attrition


Box Plot of TrainingTimesLastYear with respect to Attrition
TrainingTimesLastYear
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
No
Yes
Attrition


Box Plot of YearsAtCompany with respect to Attrition
YearsAtCompany
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
No
Yes
Attrition


Box Plot of YearsInCurrentRole with respect to Attrition
YearsInCurrentRole
0
2
4
6
8
No
Yes
Attrition


Box Plot of YearsSinceLastPromotion with respect to Attrition
YearsSinceLastPromotion
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
No
Yes
Attrition



3.2 Apply techiniques

- to remove duplicate data
- to impute or remove missing data
- to remove data inconsistencies

##---------Type the code below this line ----------##

To remove dulpicate data

We dont have any duplicate values

```
# To impute or remove missing data

#For numerical
#We shall now clean the null values
# Replace null values with mean of the column

# Calculate the mean of the column
```

```
mean_value = df['Age'].mean()
# Fill missing values with the mean
df['Age'].fillna(mean_value, inplace=True)

mean_value = df['MonthlyIncome in $'].mean()
df['MonthlyIncome in $'].fillna(mean_value, inplace=True)

mean_value = df['NumCompaniesWorked'].mean()
df['NumCompaniesWorked'].fillna(mean_value, inplace=True)

mean_value = df['TotalWorkingYears'].mean()
df['TotalWorkingYears'].fillna(mean_value, inplace=True)

mean_value = df['TrainingTimesLastYear'].mean()
df['TrainingTimesLastYear'].fillna(mean_value, inplace=True)

mean_value = df['YearsAtCompany'].mean()
df['YearsAtCompany'].fillna(mean_value, inplace=True)

mean_value = df['YearsInCurrentRole'].mean()
df['YearsInCurrentRole'].fillna(mean_value, inplace=True)

mean_value = df['YearsSinceLastPromotion'].mean()
df['YearsSinceLastPromotion'].fillna(mean_value, inplace=True)

mean_value = df['YearsSinceLastPromotion'].mean()
df['YearsSinceLastPromotion'].fillna(mean_value, inplace=True)

mean_value = df['YearsWithCurrManager'].mean()
df['YearsWithCurrManager'].fillna(mean_value, inplace=True)

#For categorical
# Forward fill missing values
df['EmployeeLocation'].fillna(method='ffill', inplace=True)

df['Department'].fillna(method='ffill', inplace=True)

df['Gender'].fillna(method='ffill', inplace=True)

df['MaritalStatus'].fillna(method='ffill', inplace=True)

df['Education'].fillna(method='ffill', inplace=True)

df['JobRole'].fillna(method='ffill', inplace=True)

df['PerformanceRating'].fillna(method='ffill', inplace=True)

df['JobSatisfaction'].fillna(method='ffill', inplace=True)

df['OverTime'].fillna(method='ffill', inplace=True)
```

```python
#checking if the missing values is filled
null_values = df.isnull().sum()

# Display the null values
print("Null values in the DataFrame:")
print(null_values)
```

```
Null values in the DataFrame:
EmployeeID                  0
EmployeeLocation            0
Age                         0
Department                  0
Gender                      0
MaritalStatus               0
Education                   0
JobRole                     0
JobLevel                    0
MonthlyIncome in $          0
NumCompaniesWorked          0
TotalWorkingYears           0
TrainingTimesLastYear       0
YearsAtCompany              0
YearsInCurrentRole          0
YearsSinceLastPromotion     0
YearsWithCurrManager        0
Attrition                   0
PerformanceRating           0
JobSatisfaction             0
OverTime                    0
dtype: int64
```

```python
#Data Inconsistencies

# Define a list of special characters to check for
special_characters = ['$' , '&' , '^', '#' , '@']

for column in df.columns:
    # Check if the column contains string values
    if df[column].dtype == 'object':
        # Iterate through each value in the column
        for value in df[column]:
            # Check if the value is a string (avoiding NaN values)
            if isinstance(value, str):
                # Check if the value contains any of the special
characters
                for char in special_characters:
                    if char in value:
                        print(f"Special character '{char}' found in
column '{column}': '{value}'")
```

```
                        break  # No need to check other characters
once one is found
```

3.3 Encode categorical data

```
##---------Type the code below this line----------##

data =
pd.get_dummies(df,columns=['EmployeeLocation','Department','Gender','M
aritalStatus','Education','JobRole','PerformanceRating','JobSatisfacti
on','OverTime'])
print(data.head())
```

```
   EmployeeID   Age  JobLevel  MonthlyIncome in $  NumCompaniesWorked
\
0           1  25.0         1              5505.0                 9.0

1           2  45.0         4              8377.0                 2.0

2           3  37.0         3             17854.0                 9.0

3           4  35.0         1             18881.0                 9.0

4           5  51.0         1             15019.0                 8.0


   TotalWorkingYears  TrainingTimesLastYear  YearsAtCompany  \
0                8.0                    0.0             9.0
1               15.0                    1.0            19.0
2                3.0                    4.0            13.0
3               29.0                    1.0             9.0
4                1.0                    3.0            11.0

   YearsInCurrentRole  YearsSinceLastPromotion  ...  JobRole_Team
Leader  \
0                 5.0                      1.0  ...
0
1                 1.0                      1.0  ...
0
2                 1.0                      1.0  ...
0
3                 0.0                      3.0  ...
0
4                 2.0                      3.0  ...
0

  PerformanceRating_Excellent  PerformanceRating_Good
PerformanceRating_Low  \
0                           0                       1
0
```

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

   JobSatisfaction_High  JobSatisfaction_Low
JobSatisfaction Medium  \
0                      0                     0                          1

1                      0                     1                          0

2                      0                     0                          1

3                      0                     1                          0

4                      0                     0                          0


    JobSatisfaction_Very High   OverTime_No   OverTime_Yes
0                           0             0              1
1                           0             1              0
2                           0             0              1
3                           0             1              0
4                           1             1              0

[5 rows x 46 columns]
```

3.4 Report

Mention and justify the method adopted

- to remove duplicate data, if present
- to impute or remove missing data, if present
- to remove data inconsistencies, if present

OR for textdata

- How many tokens after step 3?
- how may tokens after stop words filtering?

If the any of the above are not present, then also add in the report below.

##---------Type the code below this line ----------##

Duplicate Data:

There are so many ways methods to remove duplicate data. Our dataset did not have any duplicate data However most commonly used methods for duplicate values is drop_duplicates()

To impute or remove missing data:

For numerical: Common imputation methods include filling missing values with the mean, median, mode, or a constant value Here we used mean method to fill the missing data beacause of its simplicity and effectiveness

For categorical: Common imputation methods include replace missing categorical values with the mode (the most frequent value) of the respective column, forward filling, backward filling Considering our data has a ordered categorical data we have used forward filling

To remove data inconsistencies, if present: We have to check our dataset to identify inconsistencies, such as misspelled or inconsistent values, duplicate records, or data outliers. Here we have used a boxplot and also checked for special characters.

3.5 Identify the target variables.

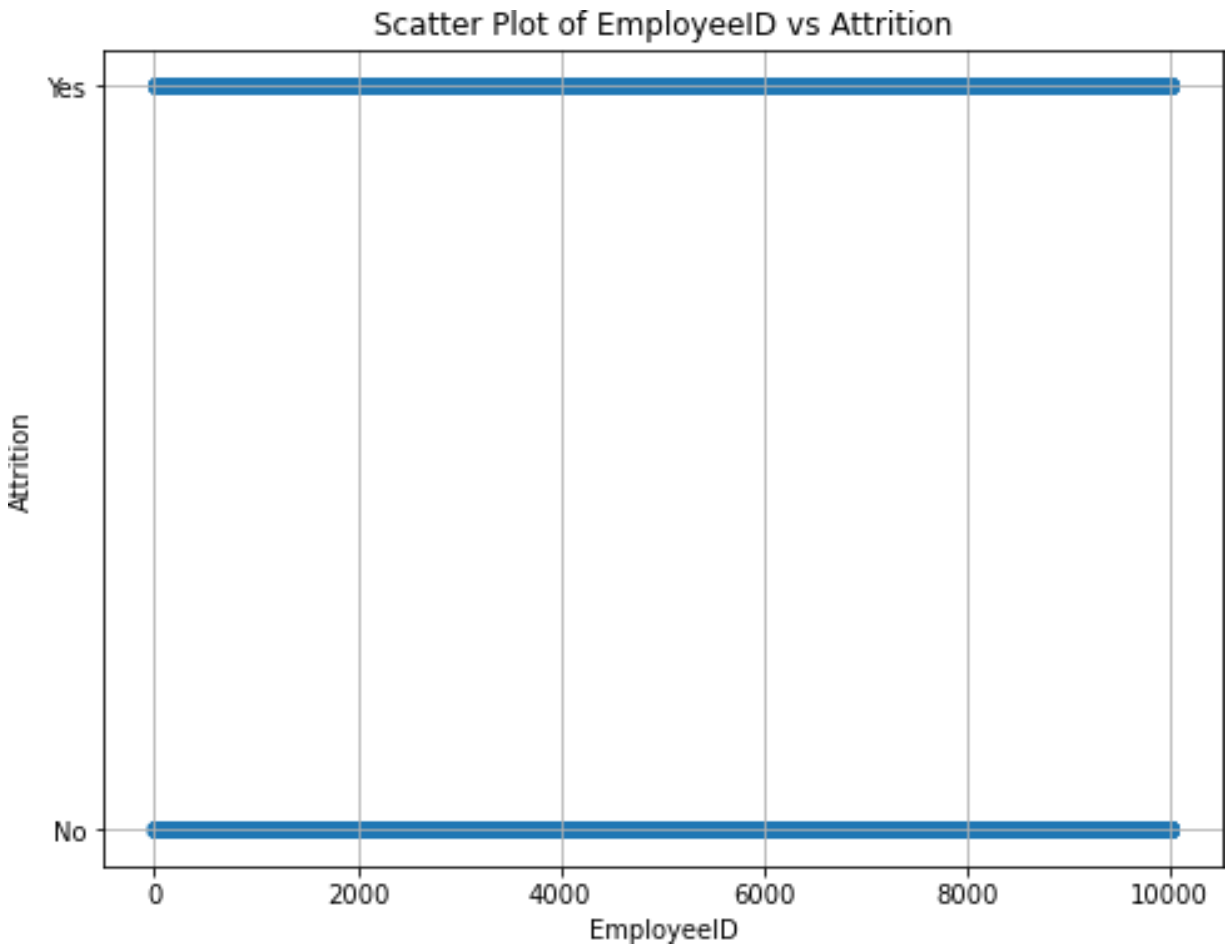
- Separate the data from the target such that the dataset is in the form of (X,y) or (Features, Label)
- Discretize / Encode the target variable or perform one-hot encoding on the target or any other as and if required.
- Report the observations

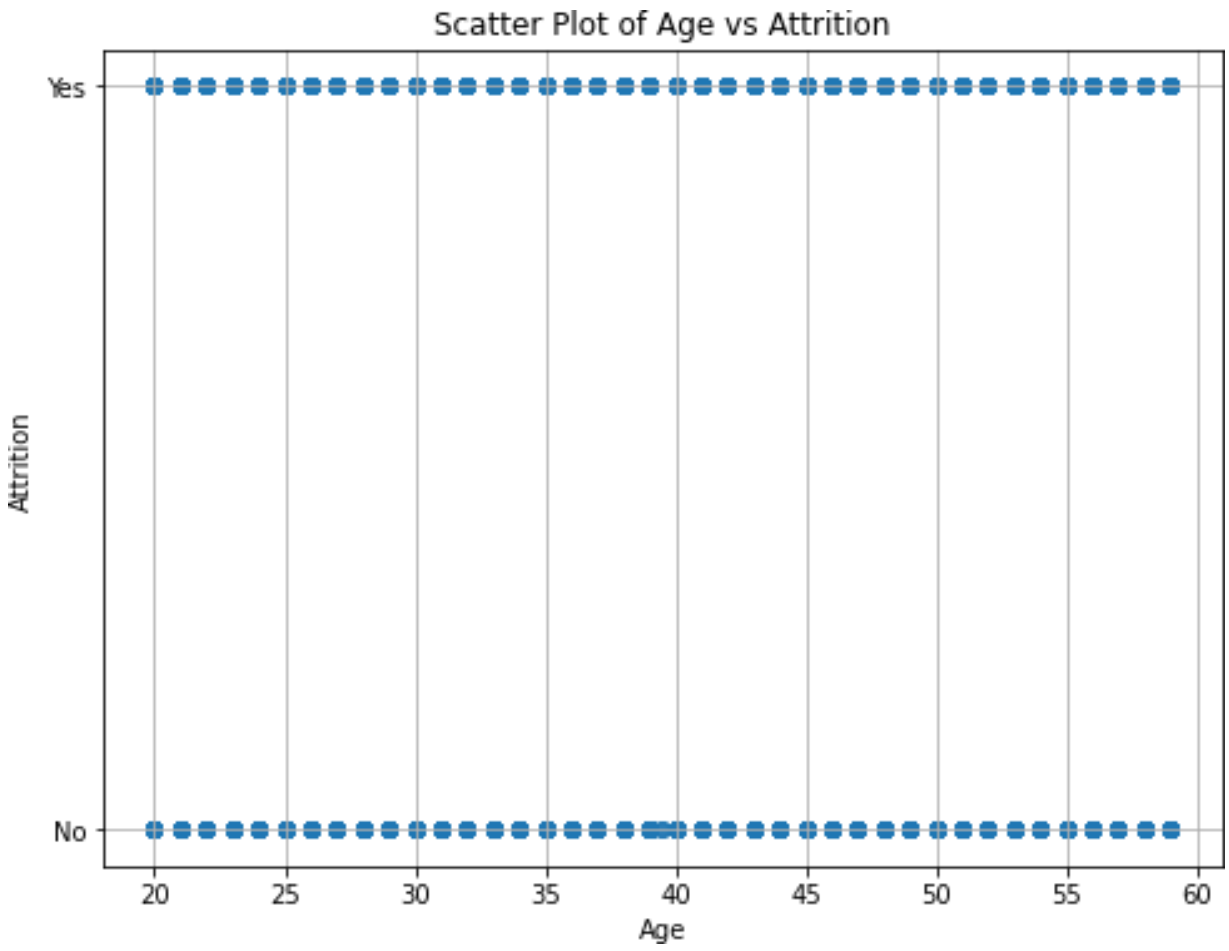
```
##---------Type the code below this line---------- ##

from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error


# Split the data into features Independent Variable/Features (X) and
Dependent/Target variable (y)
X = data.drop(['Attrition','EmployeeID'], axis=1)  # Features
y = data['Attrition']  # Target variable

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)
```

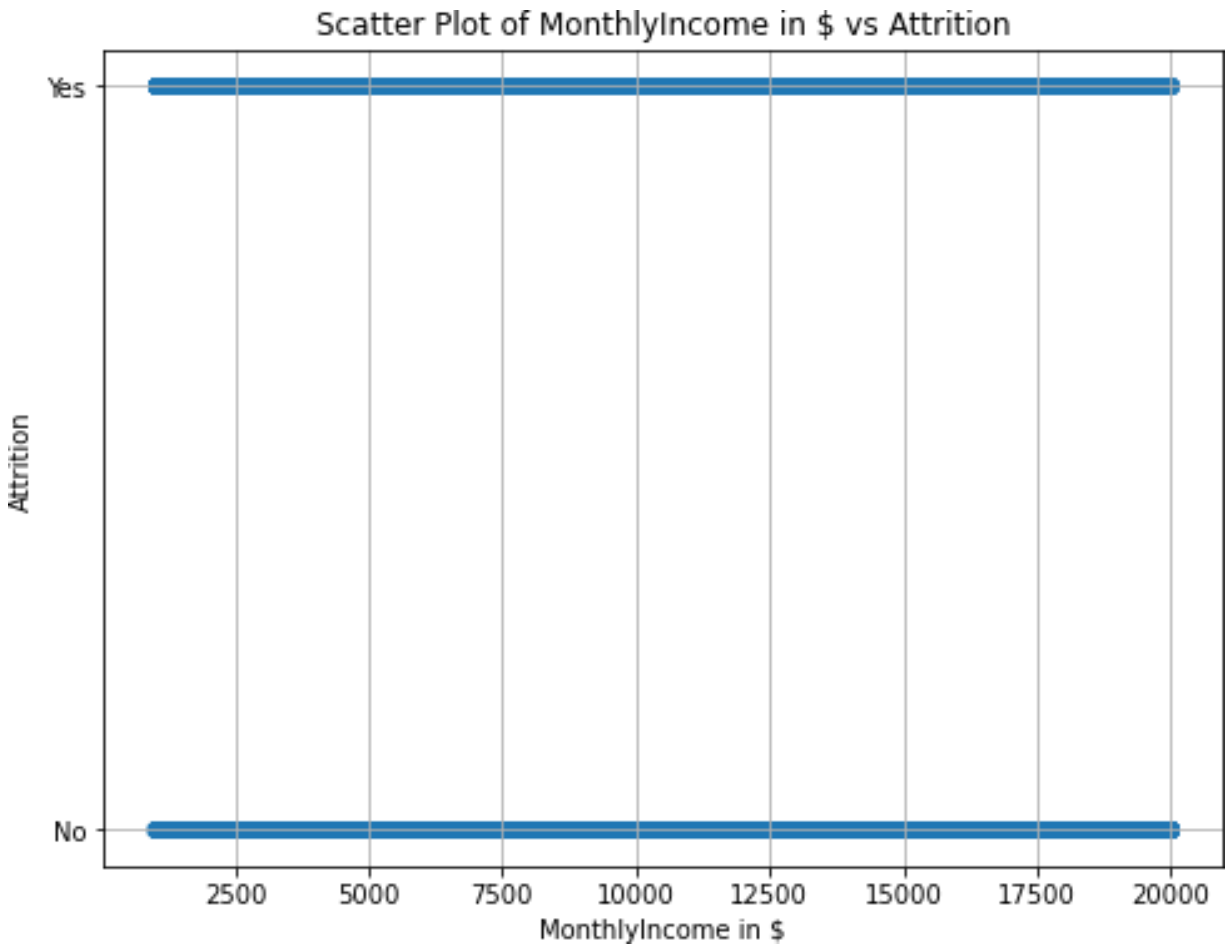
4. Data Exploration using various plots

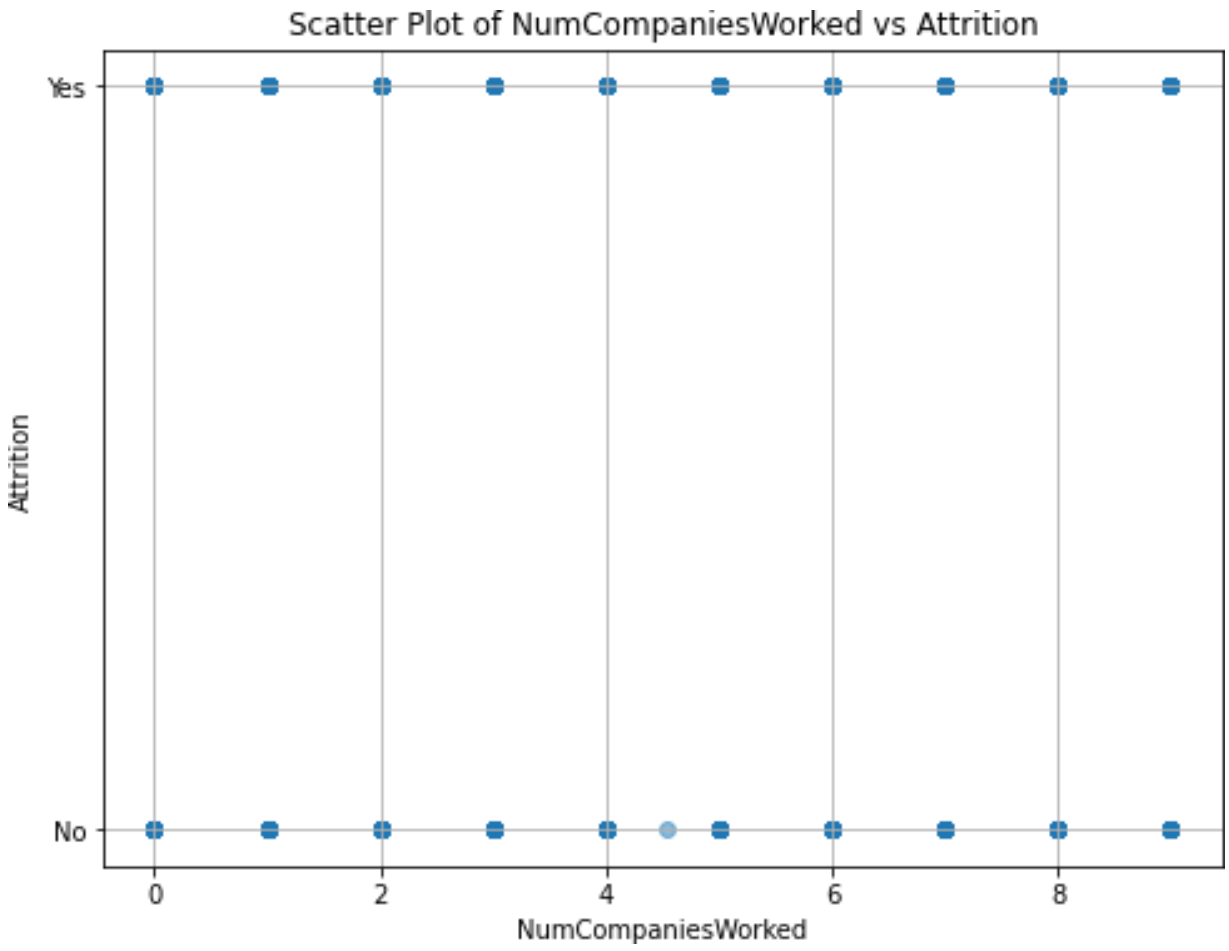
4.1 Scatter plot of each quantitative attribute with the target.

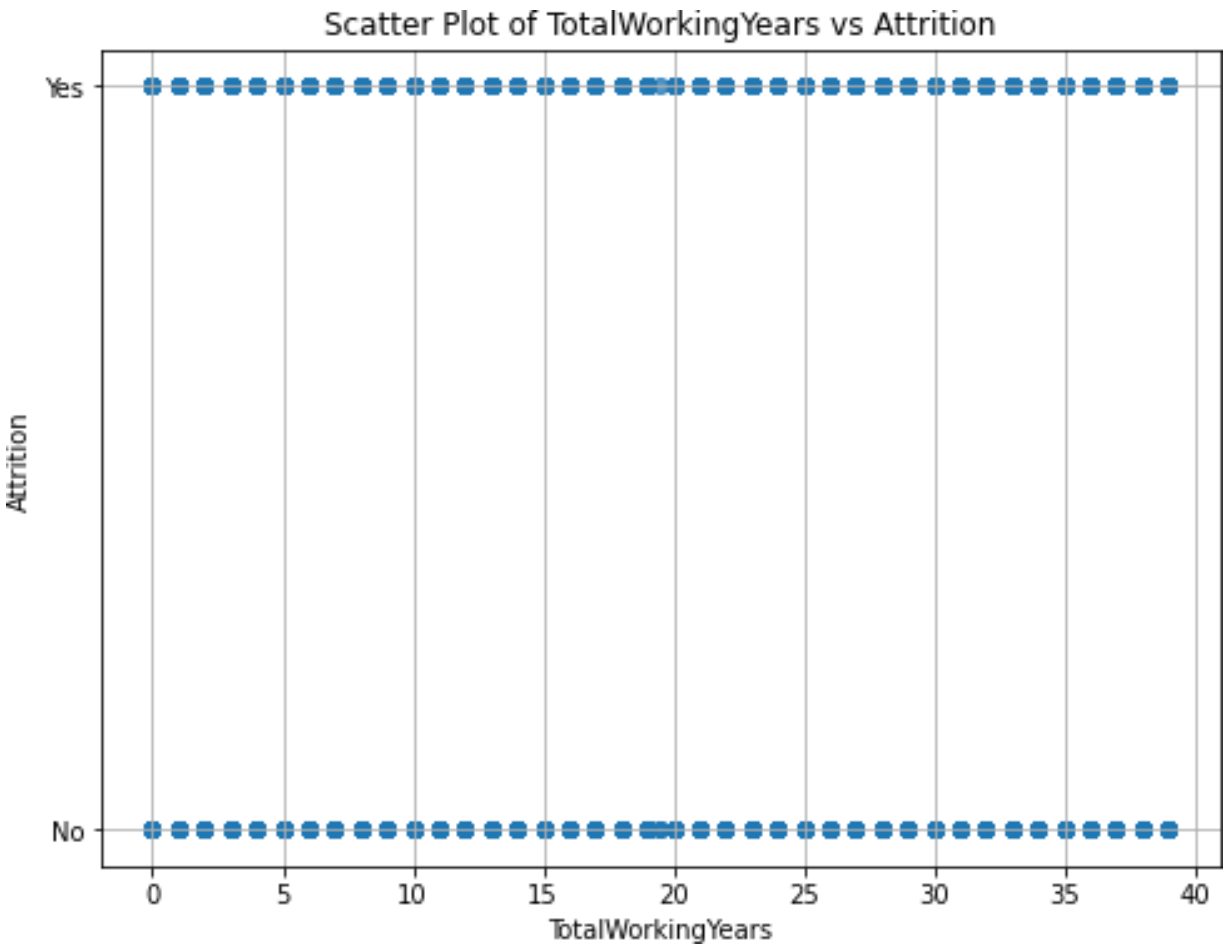
```
##---------Type the code below this line----------##

# Define the list of features for scatter plot
quantitative_features = ['EmployeeID', 'Age', 'MonthlyIncome in $',
'NumCompaniesWorked',
                         'TotalWorkingYears', 'TrainingTimesLastYear',
'YearsAtCompany',
                         'YearsInCurrentRole',
'YearsSinceLastPromotion', 'YearsWithCurrManager']

# Create scatter plots for each quantitative feature
for feature in quantitative_features:
    plt.figure(figsize=(8, 6))
    plt.scatter(df[feature], df['Attrition'], alpha=0.5)
    plt.title(f'Scatter Plot of {feature} vs Attrition')
    plt.xlabel(feature)
    plt.ylabel('Attrition')
    plt.grid(True)
    plt.show()
```

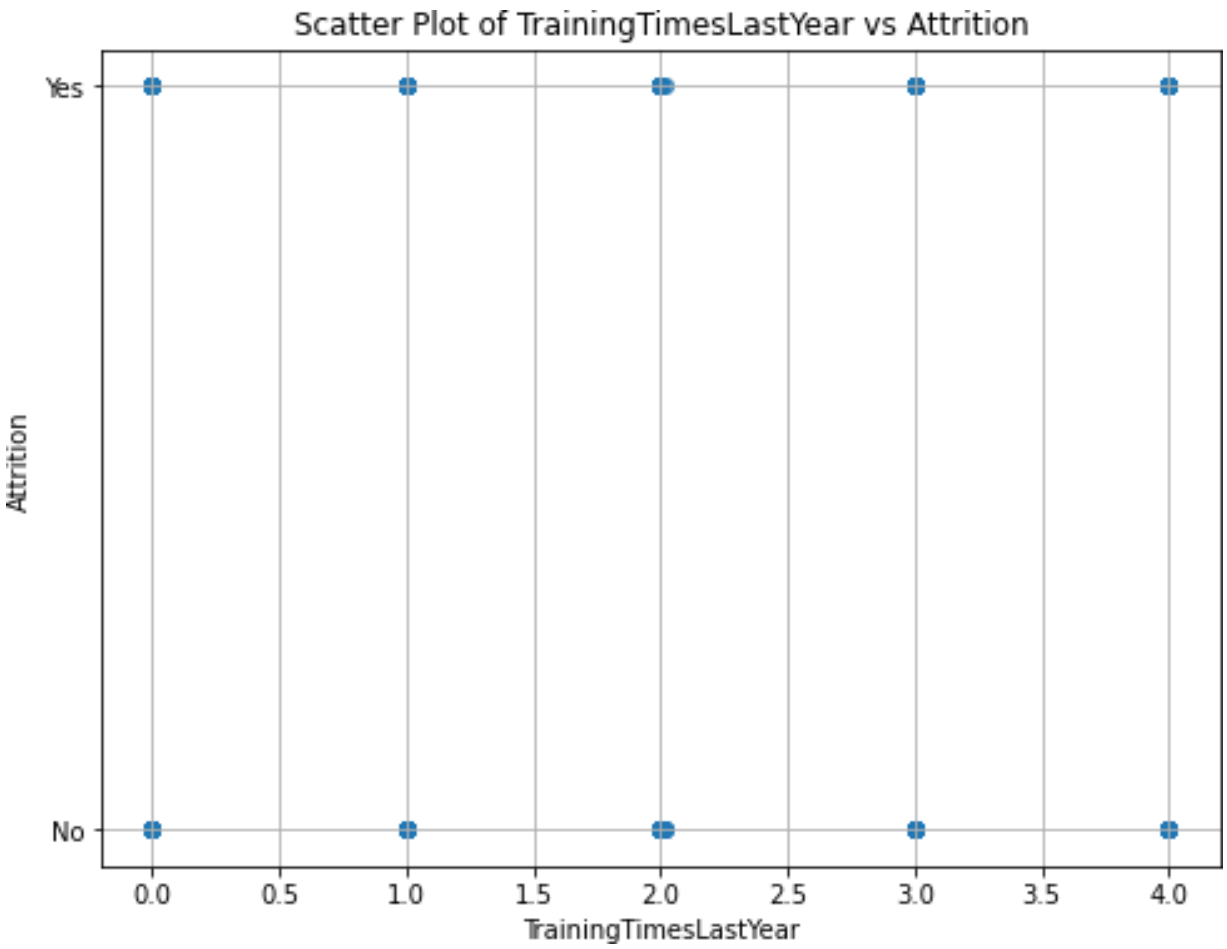

Scatter Plot of EmployeeID vs Attrition
Yes
No
Attrition
0
2000
4000
6000
8000
10000
EmployeeID

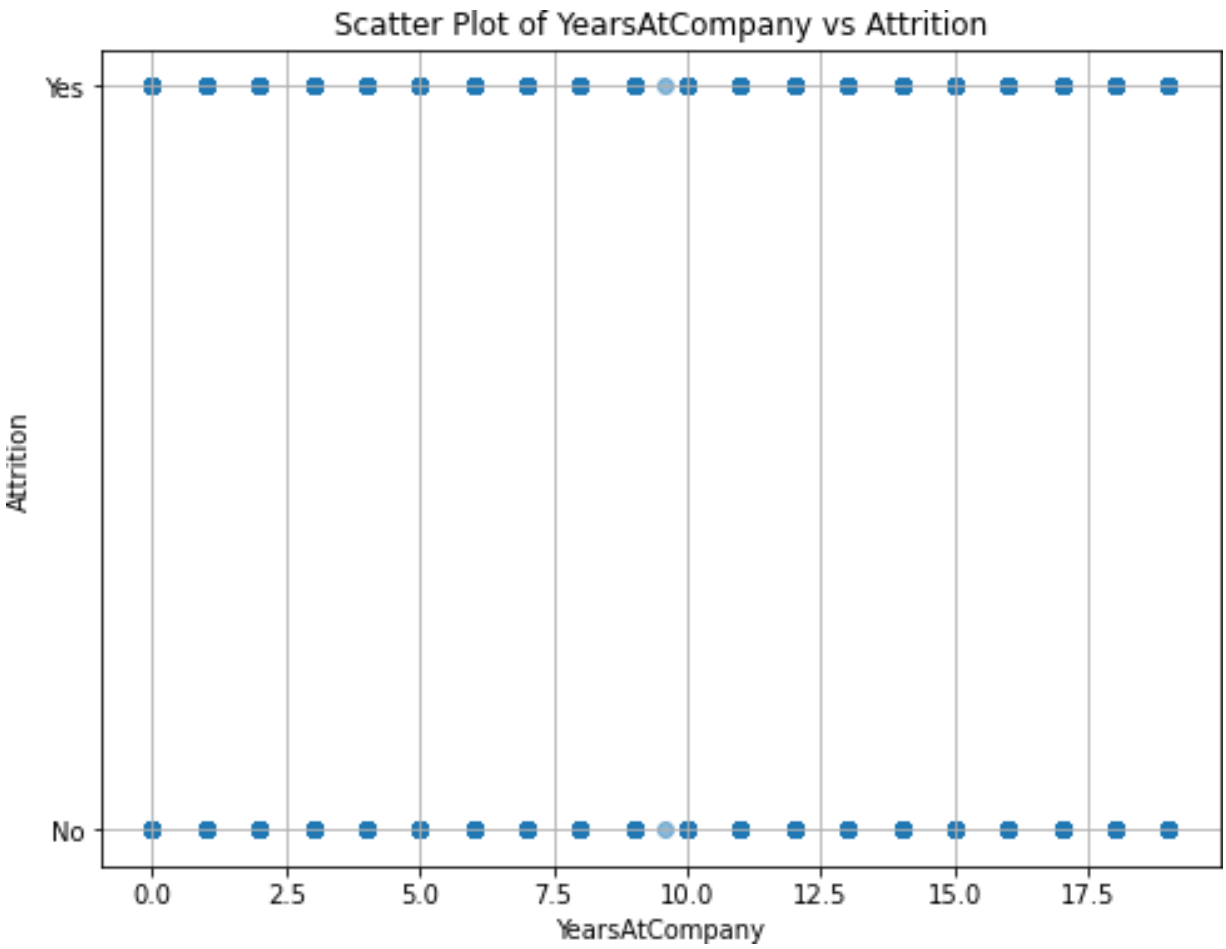

Scatter Plot of Age vs Attrition
Yes
No
Attrition
20
25
30
35
40
45
50
55
60
Age

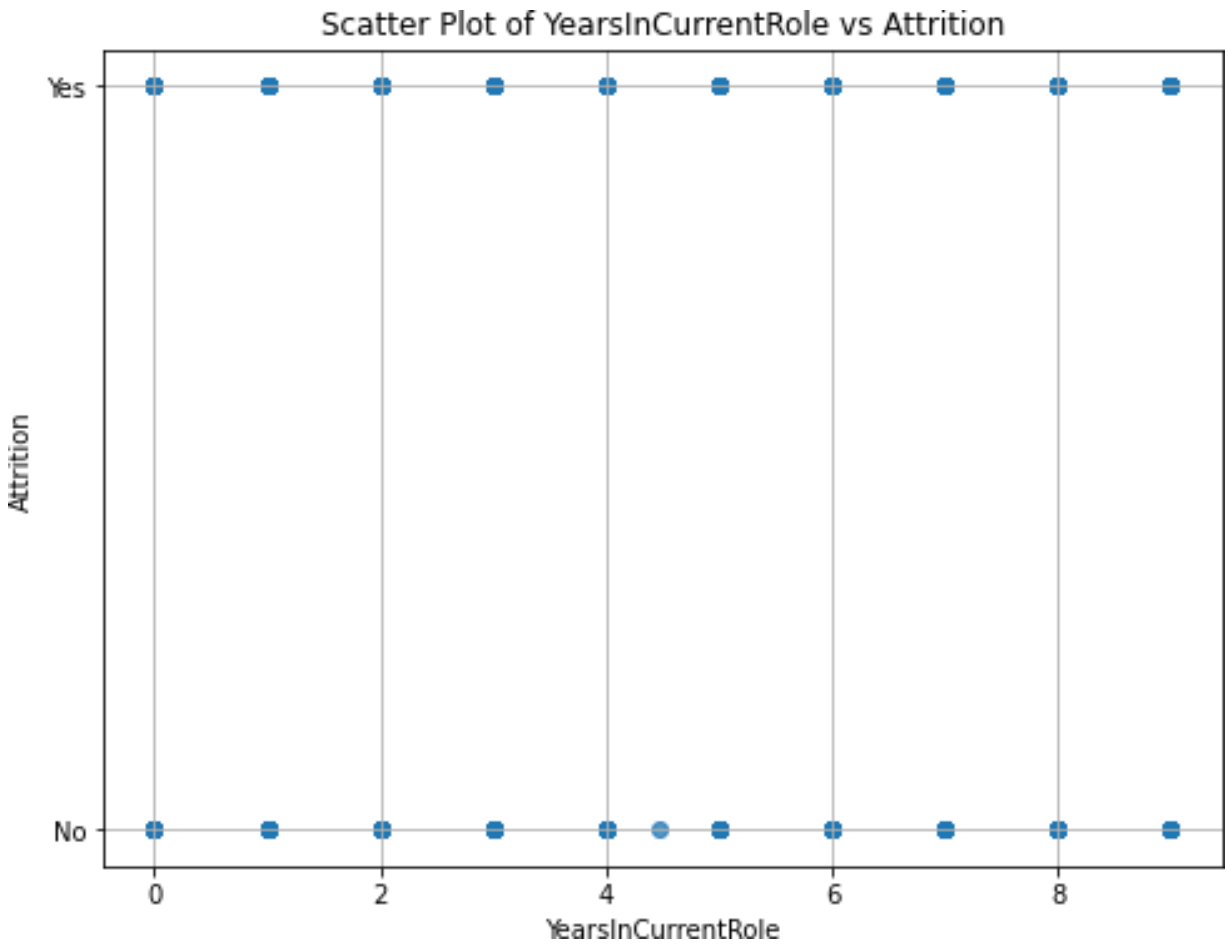

Scatter Plot of MonthlyIncome in $ vs Attrition
Yes
No
Attrition
2500
5000
7500
10000
12500
15000
17500
20000
MonthlyIncome in $

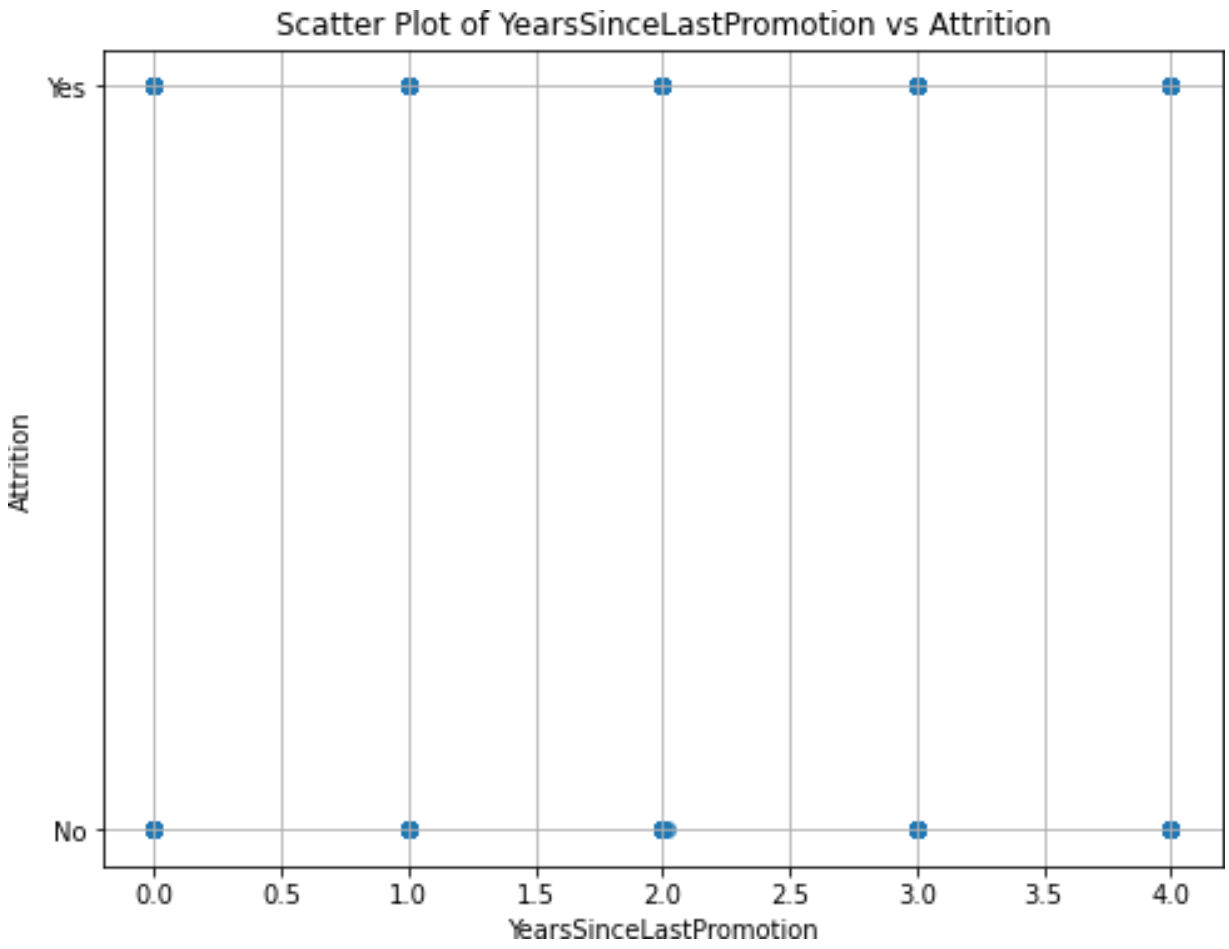

Scatter Plot of NumCompaniesWorked vs Attrition
Yes
No
Attrition
0
2
4
6
8
NumCompaniesWorked

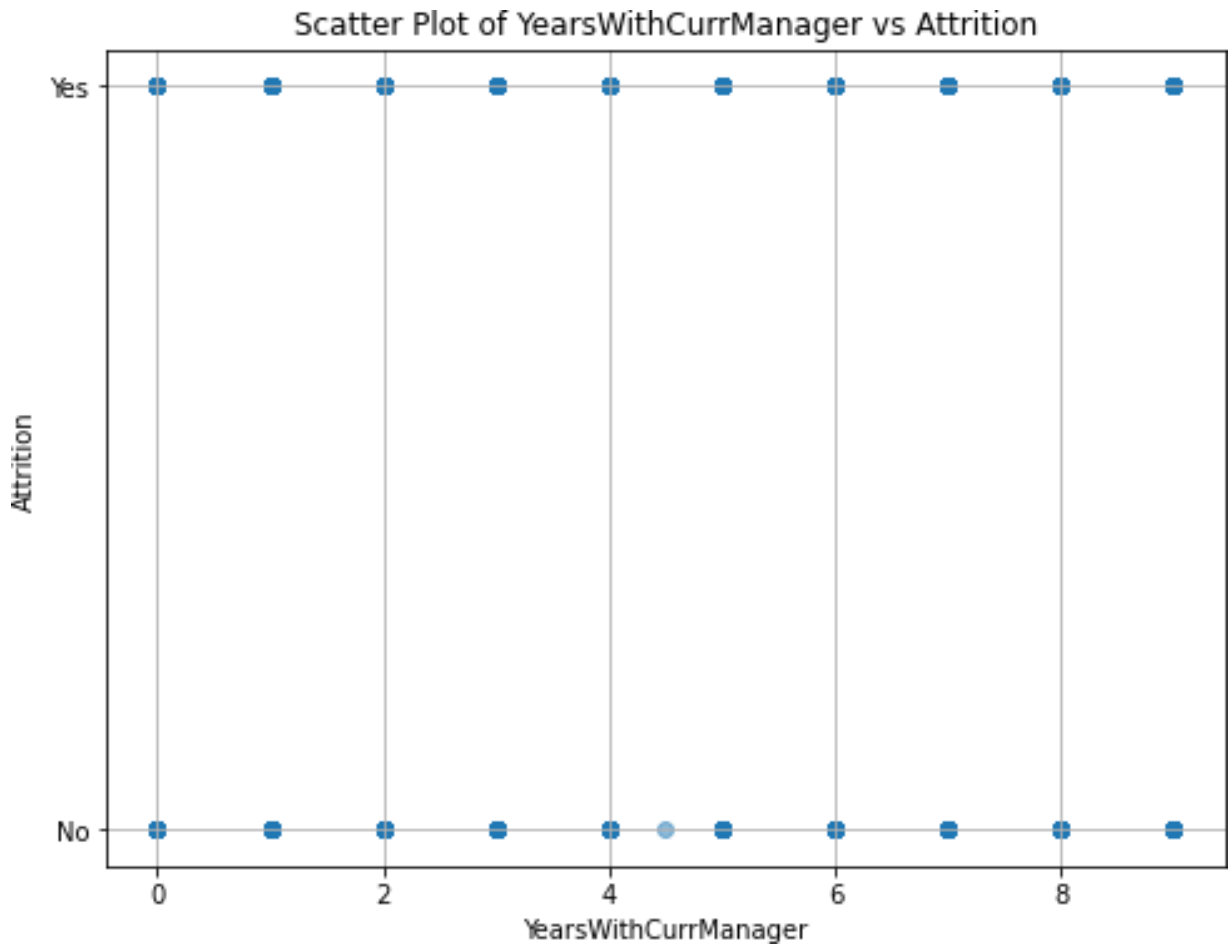

Scatter Plot of TotalWorkingYears vs Attrition
Yes
No
Attrition
0
5
10
15
20
25
30
35
40
TotalWorkingYears


Scatter Plot of TrainingTimesLastYear vs Attrition
Yes
No
Attrition
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
TrainingTimesLastYear


Scatter Plot of YearsAtCompany vs Attrition
Yes
No
Attrition
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
YearsAtCompany


Scatter Plot of YearsInCurrentRole vs Attrition
Yes
No
Attrition
0
2
4
6
8
YearsInCurrentRole


Scatter Plot of YearsSinceLastPromotion vs Attrition
Yes
No
Attrition
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
YearsSinceLastPromotion

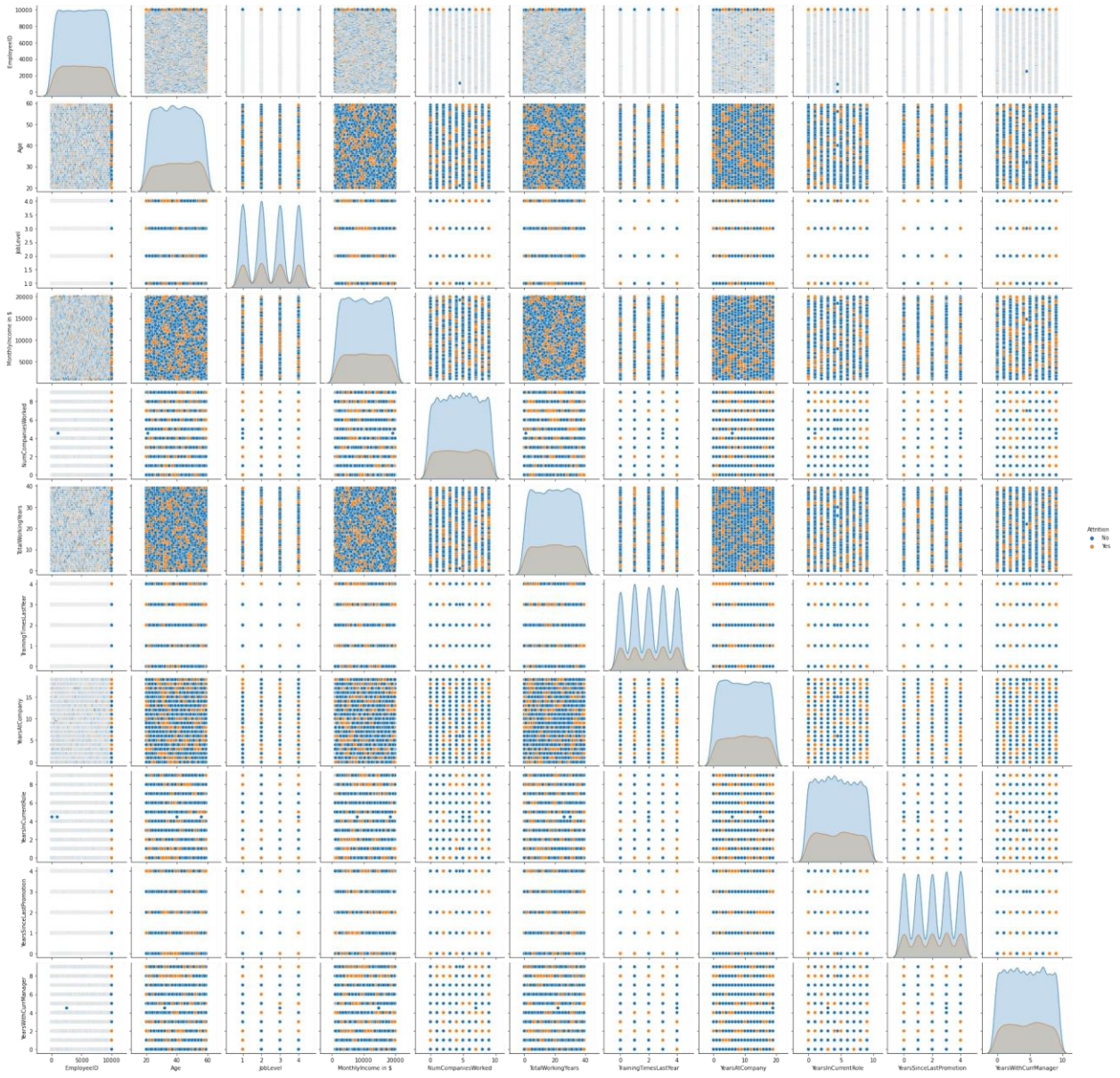


4.2 EDA using visuals

- Use (minimum) 2 plots (pair plot, heat map, correlation plot, regression plot...) to identify the optimal set of attributes that can be used for classification.
- Name them, explain why you think they can be helpful in the task and perform the plot as well. Unless proper justification for the choice of plots given, no credit will be awarded.

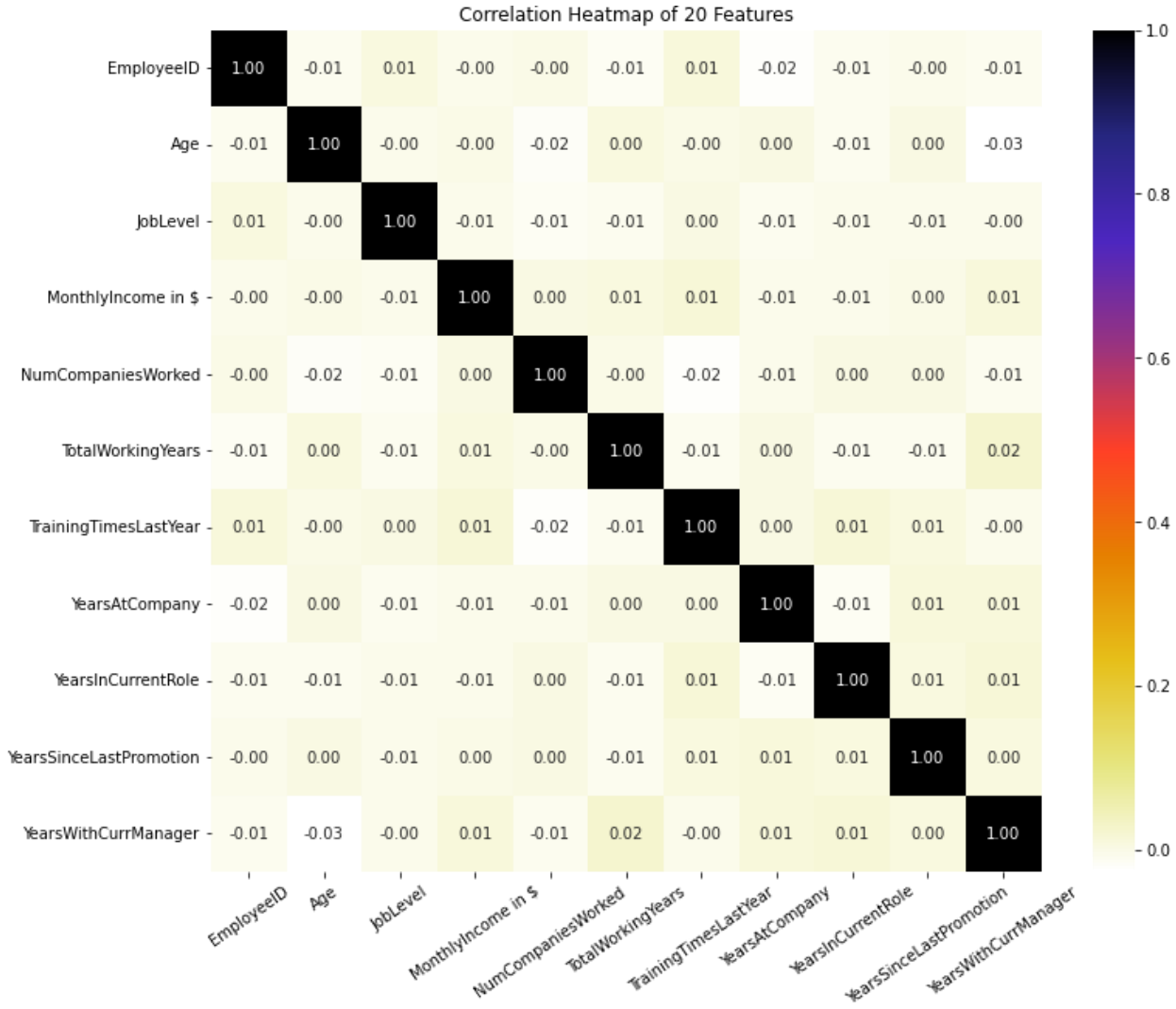
```
##---------Type the code below this line----------##


import seaborn as sns
features = ['EmployeeID', 'EmployeeLocation', 'Age', 'Department',
'Gender',
            'MaritalStatus', 'Education', 'JobRole', 'JobLevel',
'MonthlyIncome in $',
            'NumCompaniesWorked', 'TotalWorkingYears',
'TrainingTimesLastYear',
            'YearsAtCompany', 'YearsInCurrentRole',
'YearsSinceLastPromotion',
```

```
            'YearsWithCurrManager', 'Attrition', 'PerformanceRating',
            'JobSatisfaction', 'OverTime']

# Select relevant columns from the DataFrame
selected_features = ['EmployeeID', 'EmployeeLocation', 'Age',
'Department', 'Gender',
                     'MaritalStatus', 'Education', 'JobRole',
'JobLevel', 'MonthlyIncome in $',
                     'NumCompaniesWorked', 'TotalWorkingYears',
'TrainingTimesLastYear',
                     'YearsAtCompany', 'YearsInCurrentRole',
'YearsSinceLastPromotion',
                     'YearsWithCurrManager', 'Attrition',
'PerformanceRating',
                     'JobSatisfaction', 'OverTime']

# Filter the DataFrame to include only the selected features
df_selected = df[selected_features]

# Create pair plot
sns.pairplot(df_selected, hue='Attrition', diag_kind='kde')
plt.show()
```

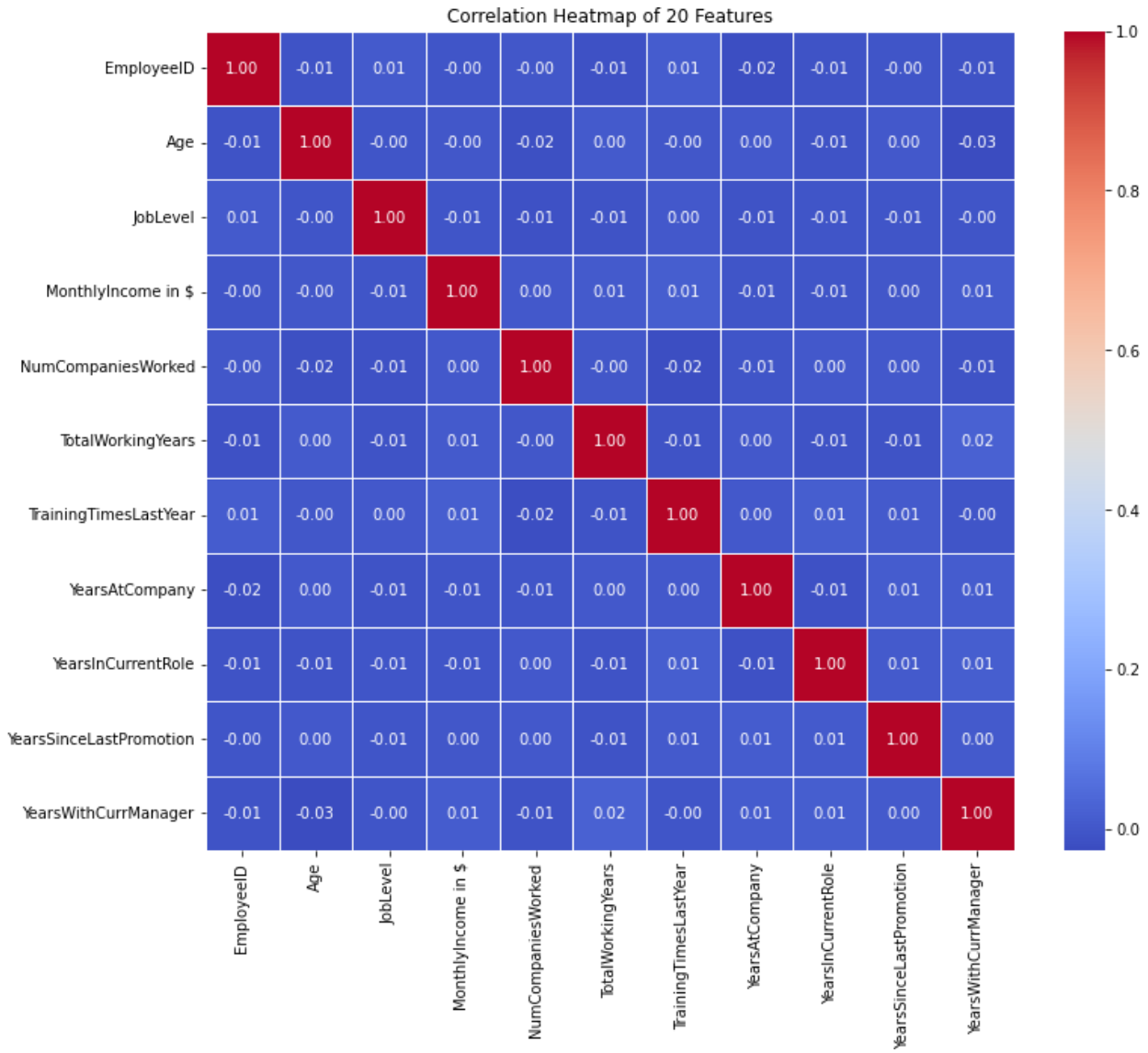



```
# Calculate the correlation matrix
corr_matrix = df.corr()

# Set up the matplotlib figure
plt.figure(figsize=(12, 10))

# Plot the heatmap
sns.heatmap(corr_matrix, annot=True, cmap='CMRmap_r', fmt=".2f")

# Add title and rotate the x-axis labels
plt.title('Correlation Heatmap of 20 Features')
plt.xticks(rotation=35)
```

```
# Show plot
plt.show()
```




```
# Calculate the correlation matrix
correlation_matrix = df.corr()

# Plot the correlation heatmap
plt.figure(figsize=(12, 10))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm',
fmt=".2f", linewidths=0.5)
plt.title('Correlation Heatmap of 20 Features')
plt.show()
```



5. Data Wrangling

5.1 Univariate Filters

Numerical and Categorical Data

- Identify top 5 significant features by evaluating each feature independently with respect to the target/other variable by exploring

1. Mutual Information (Information Gain)
2. Gini index
3. Gain Ratio
4. Chi-Squared test
5. Strenth of Association

```
##---------Type the code below this line----------##
#Mutual Information ( Information Gain)
from sklearn.feature_selection import mutual_info_classif

# Assuming 'df' is your DataFrame containing the 20 features and
'target_variable' is your target variable

# Drop non-numeric columns if present
df_numeric = df.select_dtypes(include=['number'])

# Calculate mutual information scores between features and target
variable
mi_scores = mutual_info_classif(df_numeric, df['Attrition'])

# Create a DataFrame to store feature names and their corresponding
mutual information scores
mi_df = pd.DataFrame({'Feature': df_numeric.columns, 'MI_Score':
mi_scores})

# Sort the DataFrame by mutual information score in descending order
mi_df_sorted = mi_df.sort_values(by='MI_Score', ascending=False)

# Print the top significant features
print("Top Significant Features based on Mutual Information:")
print(mi_df_sorted.head())
```

```
Top Significant Features based on Mutual Information:
                  Feature  MI_Score
8     YearsInCurrentRole   0.006672
1                    Age   0.004328
2               JobLevel   0.002850
10   YearsWithCurrManager   0.002775
4      NumCompaniesWorked   0.002258
```

```
#Gini Index
from sklearn.tree import DecisionTreeClassifier

# Assuming 'df' is your DataFrame containing the 20 features including
categorical data

# Perform one-hot encoding for categorical features
df_encoded = pd.get_dummies(df.drop('Attrition', axis=1))

# Split data into features and target variable
X = df_encoded
y = df['Attrition']
```

```
# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

# Train decision tree classifier
clf = DecisionTreeClassifier(random_state=42)
clf.fit(X_train, y_train)

# Extract feature importances based on Gini index
feature_importances = clf.feature_importances_

# Create DataFrame to store feature importances
feature_importance_df = pd.DataFrame({'Feature': X.columns,
'Gini_Importance': feature_importances})

# Sort features by Gini importance in descending order
top_significant_features =
feature_importance_df.sort_values(by='Gini_Importance',
ascending=False).head(10)  # Adjust '10' to select top features

# Display top significant features
print("Top significant features based on Gini importance:")
print(top_significant_features)
```

```
Top significant features based on Gini importance:
                    Feature  Gini_Importance
40     JobSatisfaction_Low         0.217830
41   JobSatisfaction_Medium         0.121787
0                EmployeeID         0.082464
3       MonthlyIncome in $          0.066196
7            YearsAtCompany         0.060375
1                       Age         0.057847
5         TotalWorkingYears         0.048541
10     YearsWithCurrManager         0.038793
4        NumCompaniesWorked         0.031452
8        YearsInCurrentRole         0.030160
```

5.2 Report observations

Write your observations from the results of each method. Clearly justify your choice of the method.

##---------Type the code below this line ----------##

The Gini Importance values represent the relative importance of each feature in predicting the target variable (Attrition in this case) according to a decision tree model. Job satisfaction (especially when low) appears to be the most significant predictor of attrition, followed by employee ID, monthly income, and tenure-related features such as Years at Company and Total

Working Years. Gini index is a measure of impurity frequently used in decision tree algorithms, particularly for binary classification problems. It helps decision tree algorithms to decide the best split at each node by quantifying the impurity of the data

Mutual Information measures the amount of information gained about the target variable from observing a particular feature. Features like Training Times Last Year, Years in Current Role, and Years with Current Manager have relatively higher MI scores, indicating they provide some information about attrition

6. Implement Machine Learning Techniques

Use any 2 ML tasks

1. Classification
2. Clustering
3. Association Analysis
4. Anomaly detection

You may use algorithms included in the course, e.g. Decision Tree, K-means etc. or an algorithm you learnt on your own with a brief explanation. A clear justification have to be given for why a certain algorithm was chosen to address your problem.

6.1 ML technique 1 + Justification

```
##---------Type the code below this line---------- ##
from sklearn import tree

#Top Feature based on Gini Index
top_features = ['MonthlyIncome in
$','TotalWorkingYears','Age','YearsAtCompany','YearsWithCurrManager','
YearsInCurrentRole','NumCompaniesWorked','YearsSinceLastPromotion','Tr
ainingTimesLastYear']


#ML Technique 1 - Classification
# Split data into features and target variable
X = df_encoded
y = df['Attrition']

# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=20)

# Train decision tree classifier
clf = DecisionTreeClassifier(random_state=42)
```

```
clf.fit(X_train, y_train)

model = DecisionTreeClassifier(splitter='best', criterion = 'gini')
model.fit(X_train,y_train)
```

```
DecisionTreeClassifier()
```

```
#Plot the decision tree
tree.plot_tree(model)
```

```
[Text(0.5063489520112979, 0.9807692307692307, 'X[41] <= 0.5\ngini =
0.379\nsamples = 7000\nvalue = [5220, 1780]'),
 Text(0.19311120311762855, 0.9423076923076923, 'X[40] <= 0.5\ngini =
0.275\nsamples = 5192\nvalue = [4337, 855]'),
 Text(0.05076466425538607, 0.9038461538461539, 'X[5] <= 0.5\ngini =
0.001\nsamples = 3470\nvalue = [3469, 1]'),
 Text(0.04950660481207737, 0.8653846153846154, 'X[0] <= 878.0\ngini =
0.022\nsamples = 91\nvalue = [90, 1]'),
 Text(0.04824854536876867, 0.8269230769230769, 'X[3] <= 13953.0\ngini
= 0.278\nsamples = 6\nvalue = [5, 1]'),
 Text(0.04699048592545998, 0.7884615384615384, 'gini = 0.0\nsamples =
5\nvalue = [5, 0]'),
 Text(0.04950660481207737, 0.7884615384615384, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.05076466425538607, 0.8269230769230769, 'gini = 0.0\nsamples =
85\nvalue = [85, 0]'),
 Text(0.05202272369869476, 0.8653846153846154, 'gini = 0.0\nsamples =
3379\nvalue = [3379, 0]'),
 Text(0.33545774197987105, 0.9038461538461539, 'X[28] <= 0.5\ngini =
0.5\nsamples = 1722\nvalue = [868, 854]'),
 Text(0.22814736004088693, 0.8653846153846154, 'X[3] <= 16176.5\ngini
= 0.498\nsamples = 1250\nvalue = [664, 586]'),
 Text(0.12441765607799969, 0.8269230769230769, 'X[8] <= 3.5\ngini =
0.5\nsamples = 1006\nvalue = [515, 491]'),
 Text(0.05202272369869476, 0.7884615384615384, 'X[1] <= 28.5\ngini =
0.489\nsamples = 414\nvalue = [237, 177]'),
 Text(0.018713634219216856, 0.75, 'X[38] <= 0.5\ngini = 0.491\nsamples
= 81\nvalue = [35, 46]'),
 Text(0.012266079572259789, 0.7115384615384616, 'X[4] <= 5.5\ngini =
0.452\nsamples = 55\nvalue = [19, 36]'),
 Text(0.009435445824815222, 0.6730769230769231, 'X[16] <= 0.5\ngini =
0.5\nsamples = 32\nvalue = [16, 16]'),
 Text(0.008177386381506526, 0.6346153846153846, 'X[6] <= 1.5\ngini =
0.495\nsamples = 29\nvalue = [16, 13]'),
 Text(0.005032237773234785, 0.5961538461538461, 'X[18] <= 0.5\ngini =
0.444\nsamples = 12\nvalue = [4, 8]'),
 Text(0.003774178329926089, 0.5576923076923077, 'X[5] <= 37.0\ngini =
0.32\nsamples = 10\nvalue = [2, 8]'),
 Text(0.0025161188866173927, 0.5192307692307693, 'X[19] <= 0.5\ngini =
0.198\nsamples = 9\nvalue = [1, 8]'),
```

```
 Text(0.0012580594433086963, 0.4807692307692308, 'gini = 0.0\nsamples
= 6\nvalue = [0, 6]'),
 Text(0.003774178329926089, 0.4807692307692308, 'X[9] <= 3.0\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.0025161188866173927, 0.4423076923076923, 'gini = 0.0\nsamples
= 2\nvalue = [0, 2]'),
 Text(0.005032237773234785, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.005032237773234785, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.006290297216543481, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.011322534989778267, 0.5961538461538461, 'X[8] <= 0.5\ngini =
0.415\nsamples = 17\nvalue = [12, 5]'),
 Text(0.008806416103160875, 0.5576923076923077, 'X[17] <= 0.5\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.007548356659852178, 0.5192307692307693, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.01006447554646957, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.01383865387639566, 0.5576923076923077, 'X[5] <= 0.5\ngini =
0.26\nsamples = 13\nvalue = [11, 2]'),
 Text(0.012580594433086963, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.015096713319704356, 0.5192307692307693, 'X[4] <= 0.5\ngini =
0.153\nsamples = 12\nvalue = [11, 1]'),
 Text(0.01383865387639566, 0.4807692307692308, 'X[7] <= 5.5\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.012580594433086963, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.015096713319704356, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.016354772763013052, 0.4807692307692308, 'gini = 0.0\nsamples =
10\nvalue = [10, 0]'),
 Text(0.01069350526812392, 0.6346153846153846, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.015096713319704356, 0.6730769230769231, 'X[0] <= 373.5\ngini =
0.227\nsamples = 23\nvalue = [3, 20]'),
 Text(0.01383865387639566, 0.6346153846153846, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.016354772763013052, 0.6346153846153846, 'X[7] <= 16.5\ngini =
0.091\nsamples = 21\nvalue = [1, 20]'),
 Text(0.015096713319704356, 0.5961538461538461, 'gini = 0.0\nsamples =
18\nvalue = [0, 18]'),
 Text(0.01761283220632175, 0.5961538461538461, 'X[13] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.016354772763013052, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.018870891649630444, 0.5576923076923077, 'gini = 0.0\nsamples =
```

```
1\nvalue = [1, 0]'),
 Text(0.025161188866173925, 0.7115384615384616, 'X[35] <= 0.5\ngini =
0.473\nsamples = 26\nvalue = [16, 10]'),
 Text(0.02390312942286523, 0.6730769230769231, 'X[7] <= 0.5\ngini =
0.397\nsamples = 22\nvalue = [16, 6]'),
 Text(0.022645069979556533, 0.6346153846153846, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.025161188866173925, 0.6346153846153846, 'X[7] <= 16.5\ngini =
0.32\nsamples = 20\nvalue = [16, 4]'),
 Text(0.022645069979556533, 0.5961538461538461, 'X[10] <= 8.0\ngini =
0.117\nsamples = 16\nvalue = [15, 1]'),
 Text(0.02138701053624784, 0.5576923076923077, 'gini = 0.0\nsamples =
14\nvalue = [14, 0]'),
 Text(0.02390312942286523, 0.5576923076923077, 'X[17] <= 0.5\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.022645069979556533, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.025161188866173925, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.02767730775279132, 0.5961538461538461, 'X[3] <= 5401.0\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.026419248309482623, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.028935367196100015, 0.5576923076923077, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.026419248309482623, 0.6730769230769231, 'gini = 0.0\nsamples =
4\nvalue = [0, 4]'),
 Text(0.08533181317817266, 0.75, 'X[7] <= 11.5\ngini = 0.477\nsamples
= 333\nvalue = [202, 131]'),
 Text(0.05759553388897625, 0.7115384615384616, 'X[5] <= 22.5\ngini =
0.495\nsamples = 196\nvalue = [108, 88]'),
 Text(0.04277402107249568, 0.6730769230769231, 'X[10] <= 2.5\ngini =
0.499\nsamples = 109\nvalue = [52, 57]'),
 Text(0.0339676049693348, 0.6346153846153846, 'X[19] <= 0.5\ngini =
0.428\nsamples = 29\nvalue = [20, 9]'),
 Text(0.032709545526026104, 0.5961538461538461, 'X[4] <= 7.5\ngini =
0.469\nsamples = 24\nvalue = [15, 9]'),
 Text(0.03145148608271741, 0.5576923076923077, 'X[5] <= 8.0\ngini =
0.408\nsamples = 21\nvalue = [15, 6]'),
 Text(0.030193426639408712, 0.5192307692307693, 'gini = 0.0\nsamples =
6\nvalue = [6, 0]'),
 Text(0.032709545526026104, 0.5192307692307693, 'X[5] <= 13.0\ngini =
0.48\nsamples = 15\nvalue = [9, 6]'),
 Text(0.030193426639408712, 0.4807692307692308, 'X[1] <= 34.0\ngini =
0.32\nsamples = 5\nvalue = [1, 4]'),
 Text(0.028935367196100015, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.03145148608271741, 0.4423076923076923, 'gini = 0.0\nsamples =
4\nvalue = [0, 4]'),
```

```
 Text(0.0352256644126435, 0.4807692307692308, 'X[18] <= 0.5\ngini =
0.32\nsamples = 10\nvalue = [8, 2]'),
 Text(0.0339676049693348, 0.4423076923076923, 'gini = 0.0\nsamples =
7\nvalue = [7, 0]'),
 Text(0.036483723855952194, 0.4423076923076923, 'X[9] <= 1.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.0352256644126435, 0.40384615384615385, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.03774178329926089, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.0339676049693348, 0.5576923076923077, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.0352256644126435, 0.5961538461538461, 'gini = 0.0\nsamples =
5\nvalue = [5, 0]'),
 Text(0.05158043717565655, 0.6346153846153846, 'X[5] <= 0.5\ngini =
0.48\nsamples = 80\nvalue = [32, 48]'),
 Text(0.05032237773234785, 0.5961538461538461, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.052838496618965246, 0.5961538461538461, 'X[9] <= 1.5\ngini =
0.47\nsamples = 77\nvalue = [29, 48]'),
 Text(0.04654819940242176, 0.5576923076923077, 'X[18] <= 0.5\ngini =
0.367\nsamples = 33\nvalue = [8, 25]'),
 Text(0.04403208051580437, 0.5192307692307693, 'X[19] <= 0.5\ngini =
0.172\nsamples = 21\nvalue = [2, 19]'),
 Text(0.04277402107249568, 0.4807692307692308, 'X[7] <= 0.5\ngini =
0.095\nsamples = 20\nvalue = [1, 19]'),
 Text(0.04151596162918698, 0.4423076923076923, 'X[8] <= 2.5\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.04025790218587828, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.04277402107249568, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.04403208051580437, 0.4423076923076923, 'gini = 0.0\nsamples =
18\nvalue = [0, 18]'),
 Text(0.04529013995911307, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.049064318289039156, 0.5192307692307693, 'X[4] <= 6.5\ngini =
0.5\nsamples = 12\nvalue = [6, 6]'),
 Text(0.04780625884573046, 0.4807692307692308, 'X[7] <= 9.5\ngini =
0.375\nsamples = 8\nvalue = [2, 6]'),
 Text(0.04654819940242176, 0.4423076923076923, 'gini = 0.0\nsamples =
5\nvalue = [0, 5]'),
 Text(0.049064318289039156, 0.4423076923076923, 'X[10] <= 4.0\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.04780625884573046, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.05032237773234785, 0.40384615384615385, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.05032237773234785, 0.4807692307692308, 'gini = 0.0\nsamples =
```

```
4\nvalue = [4, 0]'),
 Text(0.05912879383550873, 0.5576923076923077, 'X[29] <= 0.5\ngini =
0.499\nsamples = 44\nvalue = [21, 23]'),
 Text(0.056612674948891335, 0.5192307692307693, 'X[0] <= 9272.5\ngini
= 0.482\nsamples = 32\nvalue = [19, 13]'),
 Text(0.05535461550558264, 0.4807692307692308, 'X[14] <= 0.5\ngini =
0.452\nsamples = 29\nvalue = [19, 10]'),
 Text(0.05409655606227394, 0.4423076923076923, 'X[1] <= 37.5\ngini =
0.496\nsamples = 22\nvalue = [12, 10]'),
 Text(0.052838496618965246, 0.40384615384615385, 'gini = 0.0\nsamples
= 3\nvalue = [0, 3]'),
 Text(0.05535461550558264, 0.40384615384615385, 'X[10] <= 5.5\ngini =
0.465\nsamples = 19\nvalue = [12, 7]'),
 Text(0.052838496618965246, 0.36538461538461536, 'X[2] <= 3.0\ngini =
0.219\nsamples = 8\nvalue = [7, 1]'),
 Text(0.05158043717565655, 0.3269230769230769, 'gini = 0.0\nsamples =
7\nvalue = [7, 0]'),
 Text(0.05409655606227394, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.05787073439220003, 0.36538461538461536, 'X[7] <= 5.0\ngini =
0.496\nsamples = 11\nvalue = [5, 6]'),
 Text(0.056612674948891335, 0.3269230769230769, 'X[3] <= 14672.0\ngini
= 0.245\nsamples = 7\nvalue = [1, 6]'),
 Text(0.05535461550558264, 0.28846153846153844, 'gini = 0.0\nsamples =
6\nvalue = [0, 6]'),
 Text(0.05787073439220003, 0.28846153846153844, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.05912879383550873, 0.3269230769230769, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.056612674948891335, 0.4423076923076923, 'gini = 0.0\nsamples =
7\nvalue = [7, 0]'),
 Text(0.05787073439220003, 0.4807692307692308, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.06164491272212612, 0.5192307692307693, 'X[0] <= 2797.0\ngini =
0.278\nsamples = 12\nvalue = [2, 10]'),
 Text(0.060386853278817425, 0.4807692307692308, 'X[0] <= 2070.5\ngini
= 0.5\nsamples = 4\nvalue = [2, 2]'),
 Text(0.05912879383550873, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.06164491272212612, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.06290297216543482, 0.4807692307692308, 'gini = 0.0\nsamples =
8\nvalue = [0, 8]'),
 Text(0.07241704670545683, 0.6730769230769231, 'X[10] <= 1.5\ngini =
0.459\nsamples = 87\nvalue = [56, 31]'),
 Text(0.06667715049536091, 0.6346153846153846, 'X[18] <= 0.5\ngini =
0.444\nsamples = 12\nvalue = [4, 8]'),
 Text(0.06541909105205221, 0.5961538461538461, 'X[4] <= 1.5\ngini =
0.198\nsamples = 9\nvalue = [1, 8]'),
```

```
 Text(0.06416103160874351, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.06667715049536091, 0.5576923076923077, 'gini = 0.0\nsamples =
8\nvalue = [0, 8]'),
 Text(0.0679352099386696, 0.5961538461538461, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.07815694291555277, 0.6346153846153846, 'X[44] <= 0.5\ngini =
0.425\nsamples = 75\nvalue = [52, 23]'),
 Text(0.070451328825287, 0.5961538461538461, 'X[1] <= 58.5\ngini =
0.272\nsamples = 37\nvalue = [31, 6]'),
 Text(0.0691932693819783, 0.5576923076923077, 'X[7] <= 1.5\ngini =
0.202\nsamples = 35\nvalue = [31, 4]'),
 Text(0.06667715049536091, 0.5192307692307693, 'X[8] <= 1.5\ngini =
0.48\nsamples = 5\nvalue = [2, 3]'),
 Text(0.06541909105205221, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.0679352099386696, 0.4807692307692308, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.07170938826859569, 0.5192307692307693, 'X[3] <= 1895.5\ngini =
0.064\nsamples = 30\nvalue = [29, 1]'),
 Text(0.070451328825287, 0.4807692307692308, 'X[0] <= 7233.0\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.0691932693819783, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.07170938826859569, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.07296744771190439, 0.4807692307692308, 'gini = 0.0\nsamples =
28\nvalue = [28, 0]'),
 Text(0.07170938826859569, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.08586255700581852, 0.5961538461538461, 'X[8] <= 1.5\ngini =
0.494\nsamples = 38\nvalue = [21, 17]'),
 Text(0.08114483409341092, 0.5576923076923077, 'X[4] <= 7.5\ngini =
0.397\nsamples = 22\nvalue = [16, 6]'),
 Text(0.07799968548513918, 0.5192307692307693, 'X[7] <= 10.0\ngini =
0.278\nsamples = 18\nvalue = [15, 3]'),
 Text(0.07548356659852178, 0.4807692307692308, 'X[15] <= 0.5\ngini =
0.124\nsamples = 15\nvalue = [14, 1]'),
 Text(0.07422550715521309, 0.4423076923076923, 'gini = 0.0\nsamples =
12\nvalue = [12, 0]'),
 Text(0.07674162604183048, 0.4423076923076923, 'X[24] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.07548356659852178, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.07799968548513918, 0.40384615384615385, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.08051580437175657, 0.4807692307692308, 'X[36] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.07925774492844787, 0.4423076923076923, 'gini = 0.0\nsamples =
```

```
2\nvalue = [0, 2]'),
 Text(0.08177386381506527, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.08428998270168266, 0.5192307692307693, 'X[25] <= 0.5\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.08303192325837395, 0.4807692307692308, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.08554804214499136, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.09058027991822613, 0.5576923076923077, 'X[0] <= 8220.0\ngini =
0.43\nsamples = 16\nvalue = [5, 11]'),
 Text(0.08932222047491745, 0.5192307692307693, 'X[5] <= 34.5\ngini =
0.337\nsamples = 14\nvalue = [3, 11]'),
 Text(0.08806416103160875, 0.4807692307692308, 'gini = 0.0\nsamples =
8\nvalue = [0, 8]'),
 Text(0.09058027991822613, 0.4807692307692308, 'X[7] <= 5.5\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.08932222047491745, 0.4423076923076923, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.09183833936153483, 0.4423076923076923, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.09183833936153483, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.11306809246736908, 0.7115384615384616, 'X[20] <= 0.5\ngini =
0.431\nsamples = 137\nvalue = [94, 43]'),
 Text(0.10724956754206637, 0.6730769230769231, 'X[44] <= 0.5\ngini =
0.469\nsamples = 104\nvalue = [65, 39]'),
 Text(0.1019028149080044, 0.6346153846153846, 'X[5] <= 30.5\ngini =
0.5\nsamples = 55\nvalue = [28, 27]'),
 Text(0.09687057713476961, 0.5961538461538461, 'X[3] <= 3132.0\ngini =
0.476\nsamples = 41\nvalue = [25, 16]'),
 Text(0.09561251769146092, 0.5576923076923077, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.09812863657807831, 0.5576923076923077, 'X[29] <= 0.5\ngini =
0.45\nsamples = 38\nvalue = [25, 13]'),
 Text(0.09435445824815222, 0.5192307692307693, 'X[8] <= 1.5\ngini =
0.32\nsamples = 25\nvalue = [20, 5]'),
 Text(0.09309639880484352, 0.4807692307692308, 'gini = 0.0\nsamples =
13\nvalue = [13, 0]'),
 Text(0.09561251769146092, 0.4807692307692308, 'X[1] <= 49.5\ngini =
0.486\nsamples = 12\nvalue = [7, 5]'),
 Text(0.09435445824815222, 0.4423076923076923, 'X[10] <= 0.5\ngini =
0.346\nsamples = 9\nvalue = [7, 2]'),
 Text(0.09309639880484352, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.09561251769146092, 0.40384615384615385, 'X[4] <= 1.0\ngini =
0.219\nsamples = 8\nvalue = [7, 1]'),
 Text(0.09435445824815222, 0.36538461538461536, 'X[26] <= 0.5\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
```

```
 Text(0.09309639880484352, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.09561251769146092, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.09687057713476961, 0.36538461538461536, 'gini = 0.0\nsamples =
6\nvalue = [6, 0]'),
 Text(0.09687057713476961, 0.4423076923076923, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.10190281490800044, 0.5192307692307693, 'X[2] <= 3.5\ngini =
0.473\nsamples = 13\nvalue = [5, 8]'),
 Text(0.10064475546469570, 0.4807692307692308, 'X[10] <= 8.5\ngini =
0.408\nsamples = 7\nvalue = [5, 2]'),
 Text(0.09938669602138701, 0.4423076923076923, 'gini = 0.0\nsamples =
5\nvalue = [5, 0]'),
 Text(0.10190281490800044, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.10316087435131310, 0.4807692307692308, 'gini = 0.0\nsamples =
6\nvalue = [0, 6]'),
 Text(0.10693505268123919, 0.5961538461538461, 'X[30] <= 0.5\ngini =
0.337\nsamples = 14\nvalue = [3, 11]'),
 Text(0.10567699323793049, 0.5576923076923077, 'X[11] <= 0.5\ngini =
0.153\nsamples = 12\nvalue = [1, 11]'),
 Text(0.10441893379462179, 0.5192307692307693, 'gini = 0.0\nsamples =
11\nvalue = [0, 11]'),
 Text(0.10693505268123919, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.10819311212454788, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.11259632017612832, 0.6346153846153846, 'X[0] <= 963.5\ngini =
0.37\nsamples = 49\nvalue = [37, 12]'),
 Text(0.11133826073281962, 0.5961538461538461, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.11385437961943702, 0.5961538461538461, 'X[7] <= 17.5\ngini =
0.335\nsamples = 47\nvalue = [37, 10]'),
 Text(0.11259632017612832, 0.5576923076923077, 'X[0] <= 9466.0\ngini =
0.415\nsamples = 34\nvalue = [24, 10]'),
 Text(0.11133826073281962, 0.5192307692307693, 'X[35] <= 0.5\ngini =
0.375\nsamples = 32\nvalue = [24, 8]'),
 Text(0.10693505268123919, 0.4807692307692308, 'X[1] <= 56.5\ngini =
0.227\nsamples = 23\nvalue = [20, 3]'),
 Text(0.10441893379462179, 0.4423076923076923, 'X[15] <= 0.5\ngini =
0.095\nsamples = 20\nvalue = [19, 1]'),
 Text(0.10316087435131310, 0.40384615384615385, 'gini = 0.0\nsamples =
18\nvalue = [18, 0]'),
 Text(0.10567699323793049, 0.40384615384615385, 'X[1] <= 36.0\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.10441893379462179, 0.36538461538461536, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.10693505268123919, 0.36538461538461536, 'gini = 0.0\nsamples =
```

```
1\nvalue = [1, 0]'),
 Text(0.10945117156785658, 0.4423076923076923, 'X[3] <= 11660.0\ngini
= 0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.10819311212454788, 0.40384615384615385, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.11070923101116528, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.11574146878440006, 0.4807692307692308, 'X[0] <= 6160.5\ngini =
0.494\nsamples = 9\nvalue = [4, 5]'),
 Text(0.11448340934109137, 0.4423076923076923, 'X[36] <= 0.5\ngini =
0.278\nsamples = 6\nvalue = [1, 5]'),
 Text(0.11322534989778267, 0.40384615384615385, 'gini = 0.0\nsamples =
5\nvalue = [0, 5]'),
 Text(0.11574146878440006, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.11699952822770876, 0.4423076923076923, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.11385437961943702, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.11511243906274571, 0.5576923076923077, 'gini = 0.0\nsamples =
13\nvalue = [13, 0]'),
 Text(0.1188866173926718, 0.6730769230769231, 'X[1] <= 57.0\ngini =
0.213\nsamples = 33\nvalue = [29, 4]'),
 Text(0.11762855794936311, 0.6346153846153846, 'X[3] <= 11865.5\ngini
= 0.121\nsamples = 31\nvalue = [29, 2]'),
 Text(0.11637049850605441, 0.5961538461538461, 'gini = 0.0\nsamples =
23\nvalue = [23, 0]'),
 Text(0.1188866173926718, 0.5961538461538461, 'X[25] <= 0.5\ngini =
0.375\nsamples = 8\nvalue = [6, 2]'),
 Text(0.11762855794936311, 0.5576923076923077, 'gini = 0.0\nsamples =
5\nvalue = [5, 0]'),
 Text(0.1201446768359805, 0.5576923076923077, 'X[10] <= 6.0\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.1188866173926718, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.1214027362792892, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.1201446768359805, 0.6346153846153846, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.19681258845730462, 0.7884615384615384, 'X[1] <= 37.5\ngini =
0.498\nsamples = 592\nvalue = [278, 314]'),
 Text(0.14716346909891492, 0.75, 'X[0] <= 9399.5\ngini = 0.498\
nsamples = 269\nvalue = [143, 126]'),
 Text(0.13416024532159143, 0.7115384615384616, 'X[10] <= 0.5\ngini =
0.5\nsamples = 253\nvalue = [129, 124]'),
 Text(0.12423337002673376, 0.6730769230769231, 'X[3] <= 2526.5\ngini =
0.375\nsamples = 24\nvalue = [18, 6]'),
 Text(0.12297531058342506, 0.6346153846153846, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
```

```
 Text(0.12549142947004246, 0.6346153846153846, 'X[0] <= 8288.0\ngini =
0.245\nsamples = 21\nvalue = [18, 3]'),
 Text(0.12423337002673376, 0.5961538461538461, 'X[3] <= 13416.0\ngini
= 0.1\nsamples = 19\nvalue = [18, 1]'),
 Text(0.12297531058342506, 0.5576923076923077, 'gini = 0.0\nsamples =
18\nvalue = [18, 0]'),
 Text(0.12549142947004246, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.12674948891335117, 0.5961538461538461, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.14408712061644913, 0.6730769230769231, 'X[7] <= 0.5\ngini =
0.5\nsamples = 229\nvalue = [111, 118]'),
 Text(0.13052366724327724, 0.6346153846153846, 'X[0] <= 7672.5\ngini =
0.245\nsamples = 14\nvalue = [12, 2]'),
 Text(0.12926560779996854, 0.5961538461538461, 'gini = 0.0\nsamples =
11\nvalue = [11, 0]'),
 Text(0.13178172668658594, 0.5961538461538461, 'X[24] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.13052366724327724, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.13303978612989464, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.157650573989621, 0.6346153846153846, 'X[10] <= 8.5\ngini =
0.497\nsamples = 215\nvalue = [99, 116]'),
 Text(0.145463123132568, 0.5961538461538461, 'X[24] <= 0.5\ngini =
0.49\nsamples = 186\nvalue = [80, 106]'),
 Text(0.13555590501651202, 0.5576923076923077, 'X[3] <= 9714.0\ngini =
0.499\nsamples = 139\nvalue = [67, 72]'),
 Text(0.12895109293914137, 0.5192307692307693, 'X[3] <= 6671.0\ngini =
0.488\nsamples = 85\nvalue = [49, 36]'),
 Text(0.12328982544425224, 0.4807692307692308, 'X[5] <= 20.5\ngini =
0.496\nsamples = 55\nvalue = [25, 30]'),
 Text(0.11951564711432615, 0.4423076923076923, 'X[3] <= 5981.0\ngini =
0.469\nsamples = 24\nvalue = [15, 9]'),
 Text(0.11825758767101746, 0.40384615384615385, 'X[12] <= 0.5\ngini =
0.375\nsamples = 20\nvalue = [15, 5]'),
 Text(0.11574146878440006, 0.36538461538461536, 'X[0] <= 257.0\ngini =
0.291\nsamples = 17\nvalue = [14, 3]'),
 Text(0.11448340934109137, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.11699952822770876, 0.3269230769230769, 'X[7] <= 15.5\ngini =
0.219\nsamples = 16\nvalue = [14, 2]'),
 Text(0.11574146878440006, 0.28846153846153844, 'X[0] <= 1301.5\ngini
= 0.124\nsamples = 15\nvalue = [14, 1]'),
 Text(0.11448340934109137, 0.25, 'X[7] <= 3.5\ngini = 0.444\nsamples =
3\nvalue = [2, 1]'),
 Text(0.11322534989778267, 0.21153846153846154, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.11574146878440006, 0.21153846153846154, 'gini = 0.0\nsamples =
```

```
2\nvalue = [2, 0]'),
 Text(0.11699952822770876, 0.25, 'gini = 0.0\nsamples = 12\nvalue =
[12, 0]'),
 Text(0.11825758767101746, 0.28846153846153844, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.12077370655763485, 0.36538461538461536, 'X[31] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.11951564711432615, 0.3269230769230769, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.12203176600094355, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.12077370655763485, 0.40384615384615385, 'gini = 0.0\nsamples =
4\nvalue = [0, 4]'),
 Text(0.12706400377417834, 0.4423076923076923, 'X[3] <= 2404.5\ngini =
0.437\nsamples = 31\nvalue = [10, 21]'),
 Text(0.12580594433086964, 0.40384615384615385, 'gini = 0.0\nsamples =
8\nvalue = [0, 8]'),
 Text(0.12832206321748701, 0.40384615384615385, 'X[15] <= 0.5\ngini =
0.491\nsamples = 23\nvalue = [10, 13]'),
 Text(0.12580594433086964, 0.36538461538461536, 'X[3] <= 2484.0\ngini
= 0.415\nsamples = 17\nvalue = [5, 12]'),
 Text(0.12454788488756094, 0.3269230769230769, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.12706400377417834, 0.3269230769230769, 'X[27] <= 0.5\ngini =
0.32\nsamples = 15\nvalue = [3, 12]'),
 Text(0.12580594433086964, 0.28846153846153844, 'gini = 0.0\nsamples =
9\nvalue = [0, 9]'),
 Text(0.12832206321748701, 0.28846153846153844, 'X[34] <= 0.5\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.12706400377417834, 0.25, 'X[20] <= 0.5\ngini = 0.375\nsamples
= 4\nvalue = [1, 3]'),
 Text(0.12580594433086964, 0.21153846153846154, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.12832206321748701, 0.21153846153846154, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.12958012266079572, 0.25, 'gini = 0.0\nsamples = 2\nvalue = [2,
0]'),
 Text(0.13083818210410442, 0.36538461538461536, 'X[10] <= 2.5\ngini =
0.278\nsamples = 6\nvalue = [5, 1]'),
 Text(0.12958012266079572, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.13209624154741312, 0.3269230769230769, 'gini = 0.0\nsamples =
5\nvalue = [5, 0]'),
 Text(0.13461236043403052, 0.4807692307692308, 'X[44] <= 0.5\ngini =
0.32\nsamples = 30\nvalue = [24, 6]'),
 Text(0.13335430099072182, 0.4423076923076923, 'gini = 0.0\nsamples =
13\nvalue = [13, 0]'),
 Text(0.1358704198773392, 0.4423076923076923, 'X[3] <= 7707.0\ngini =
0.457\nsamples = 17\nvalue = [11, 6]'),
 Text(0.13461236043403052, 0.40384615384615385, 'gini = 0.0\nsamples =
```

```
7\nvalue = [7, 0]'),
 Text(0.1371284793206479, 0.40384615384615385, 'X[2] <= 3.5\ngini =
0.48\nsamples = 10\nvalue = [4, 6]'),
 Text(0.1358704198773392, 0.36538461538461536, 'X[12] <= 0.5\ngini =
0.245\nsamples = 7\nvalue = [1, 6]'),
 Text(0.13461236043403052, 0.3269230769230769, 'gini = 0.0\nsamples =
6\nvalue = [0, 6]'),
 Text(0.1371284793206479, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.1383865387639566, 0.36538461538461536, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.1421607170938827, 0.5192307692307693, 'X[11] <= 0.5\ngini =
0.444\nsamples = 54\nvalue = [18, 36]'),
 Text(0.1396445982072653, 0.4807692307692308, 'X[7] <= 1.5\ngini =
0.375\nsamples = 44\nvalue = [11, 33]'),
 Text(0.1383865387639566, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.140902657650574, 0.4423076923076923, 'X[5] <= 12.5\ngini =
0.337\nsamples = 42\nvalue = [9, 33]'),
 Text(0.1396445982072653, 0.40384615384615385, 'gini = 0.0\nsamples =
15\nvalue = [0, 15]'),
 Text(0.1421607170938827, 0.40384615384615385, 'X[4] <= 1.5\ngini =
0.444\nsamples = 27\nvalue = [9, 18]'),
 Text(0.140902657650574, 0.36538461538461536, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.14341877653719137, 0.36538461538461536, 'X[0] <= 3330.0\ngini
= 0.375\nsamples = 24\nvalue = [6, 18]'),
 Text(0.140902657650574, 0.3269230769230769, 'X[5] <= 24.5\ngini =
0.469\nsamples = 8\nvalue = [5, 3]'),
 Text(0.1396445982072653, 0.28846153846153844, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.1421607170938827, 0.28846153846153844, 'X[37] <= 0.5\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.140902657650574, 0.25, 'gini = 0.0\nsamples = 3\nvalue = [0,
3]'),
 Text(0.14341877653719137, 0.25, 'gini = 0.0\nsamples = 1\nvalue = [1,
0]'),
 Text(0.14593489542380877, 0.3269230769230769, 'X[8] <= 8.5\ngini =
0.117\nsamples = 16\nvalue = [1, 15]'),
 Text(0.14467683598050007, 0.28846153846153844, 'gini = 0.0\nsamples =
15\nvalue = [0, 15]'),
 Text(0.14719295486711748, 0.28846153846153844, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.14467683598050007, 0.4807692307692308, 'X[36] <= 0.5\ngini =
0.42\nsamples = 10\nvalue = [7, 3]'),
 Text(0.14341877653719137, 0.4423076923076923, 'gini = 0.0\nsamples =
6\nvalue = [6, 0]'),
 Text(0.14593489542380877, 0.4423076923076923, 'X[10] <= 7.5\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
```

```
 Text(0.14467683598050007, 0.40384615384615385, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.14719295486711748, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.155370341248624, 0.5576923076923077, 'X[6] <= 2.5\ngini = 0.4\
nsamples = 47\nvalue = [13, 34]'),
 Text(0.15222519264035225, 0.5192307692307693, 'X[9] <= 3.5\ngini =
0.491\nsamples = 23\nvalue = [10, 13]'),
 Text(0.14970907375373485, 0.4807692307692308, 'X[0] <= 1419.0\ngini =
0.415\nsamples = 17\nvalue = [5, 12]'),
 Text(0.14845101431042618, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.15096713319704355, 0.4423076923076923, 'X[10] <= 4.5\ngini =
0.32\nsamples = 15\nvalue = [3, 12]'),
 Text(0.14970907375373485, 0.40384615384615385, 'X[5] <= 6.0\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.14845101431042618, 0.36538461538461536, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.15096713319704355, 0.36538461538461536, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.15222519264035225, 0.40384615384615385, 'gini = 0.0\nsamples =
9\nvalue = [0, 9]'),
 Text(0.15474131152696965, 0.4807692307692308, 'X[0] <= 534.5\ngini =
0.278\nsamples = 6\nvalue = [5, 1]'),
 Text(0.15348325208366095, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.15599937097027836, 0.4423076923076923, 'gini = 0.0\nsamples =
5\nvalue = [5, 0]'),
 Text(0.15851548985689573, 0.5192307692307693, 'X[7] <= 1.5\ngini =
0.219\nsamples = 24\nvalue = [3, 21]'),
 Text(0.15725743041358703, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.15977354930020443, 0.4807692307692308, 'X[36] <= 0.5\ngini =
0.159\nsamples = 23\nvalue = [2, 21]'),
 Text(0.15851548985689573, 0.4423076923076923, 'gini = 0.0\nsamples =
16\nvalue = [0, 16]'),
 Text(0.16103160874351313, 0.4423076923076923, 'X[7] <= 4.5\ngini =
0.408\nsamples = 7\nvalue = [2, 5]'),
 Text(0.15977354930020443, 0.40384615384615385, 'gini = 0.0\nsamples =
4\nvalue = [0, 4]'),
 Text(0.16228966818682183, 0.40384615384615385, 'X[35] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.16103160874351313, 0.36538461538461536, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.16354772763013053, 0.36538461538461536, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.169838024846674, 0.5961538461538461, 'X[5] <= 21.5\ngini =
0.452\nsamples = 29\nvalue = [19, 10]'),
 Text(0.1673219059600566, 0.5576923076923077, 'X[2] <= 1.5\ngini =
```

```
0.278\nsamples = 18\nvalue = [15, 3]'),
 Text(0.1660638465167479, 0.5192307692307693, 'X[4] <= 4.5\ngini =
0.48\nsamples = 5\nvalue = [2, 3]'),
 Text(0.1648057870734392, 0.4807692307692308, 'X[17] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.16354772763013053, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.1660638465167479, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.1673219059600566, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.1685799654033653, 0.5192307692307693, 'gini = 0.0\nsamples =
13\nvalue = [13, 0]'),
 Text(0.1723541437332914, 0.5576923076923077, 'X[1] <= 24.5\ngini =
0.463\nsamples = 11\nvalue = [4, 7]'),
 Text(0.17109608428998271, 0.5192307692307693, 'X[0] <= 2231.5\ngini =
0.32\nsamples = 5\nvalue = [4, 1]'),
 Text(0.169838024846674, 0.4807692307692308, 'gini = 0.0\nsamples = 1\
nvalue = [0, 1]'),
 Text(0.1723541437332914, 0.4807692307692308, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.1736122031766001, 0.5192307692307693, 'gini = 0.0\nsamples =
6\nvalue = [0, 6]'),
 Text(0.16016669287623841, 0.7115384615384616, 'X[16] <= 0.5\ngini =
0.219\nsamples = 16\nvalue = [14, 2]'),
 Text(0.1589086334329297, 0.6730769230769231, 'gini = 0.0\nsamples =
12\nvalue = [12, 0]'),
 Text(0.1614247523195471, 0.6730769230769231, 'X[5] <= 25.0\ngini =
0.5\nsamples = 4\nvalue = [2, 2]'),
 Text(0.16016669287623841, 0.6346153846153846, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.1626828117628558, 0.6346153846153846, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.2464617078156943, 0.75, 'X[6] <= 2.5\ngini = 0.487\nsamples =
323\nvalue = [135, 188]'),
 Text(0.22519264035225664, 0.7115384615384616, 'X[1] <= 57.5\ngini =
0.498\nsamples = 208\nvalue = [97, 111]'),
 Text(0.20820883786758923, 0.6730769230769231, 'X[0] <= 3012.0\ngini =
0.5\nsamples = 193\nvalue = [95, 98]'),
 Text(0.18949520364837238, 0.6346153846153846, 'X[7] <= 12.5\ngini =
0.468\nsamples = 59\nvalue = [37, 22]'),
 Text(0.18273313414058814, 0.5961538461538461, 'X[27] <= 0.5\ngini =
0.496\nsamples = 44\nvalue = [24, 20]'),
 Text(0.1773863815065262, 0.5576923076923077, 'X[10] <= 7.5\ngini =
0.42\nsamples = 20\nvalue = [6, 14]'),
 Text(0.1761283220632175, 0.5192307692307693, 'X[1] <= 49.0\ngini =
0.346\nsamples = 18\nvalue = [4, 14]'),
 Text(0.1748702626199088, 0.4807692307692308, 'gini = 0.0\nsamples =
9\nvalue = [0, 9]'),
```

```
 Text(0.1773863815065262, 0.4807692307692308, 'X[20] <= 0.5\ngini =
0.494\nsamples = 9\nvalue = [4, 5]'),
 Text(0.1761283220632175, 0.4423076923076923, 'X[8] <= 7.5\ngini =
0.444\nsamples = 6\nvalue = [4, 2]'),
 Text(0.1748702626199088, 0.40384615384615385, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.1773863815065262, 0.40384615384615385, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.1786444409498349, 0.4423076923076923, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.1786444409498349, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.1880798867746501, 0.5576923076923077, 'X[34] <= 0.5\ngini =
0.375\nsamples = 24\nvalue = [18, 6]'),
 Text(0.18493473816637837, 0.5192307692307693, 'X[9] <= 3.5\ngini =
0.266\nsamples = 19\nvalue = [16, 3]'),
 Text(0.18241861927976097, 0.4807692307692308, 'X[14] <= 0.5\ngini =
0.124\nsamples = 15\nvalue = [14, 1]'),
 Text(0.18116055983645227, 0.4423076923076923, 'gini = 0.0\nsamples =
13\nvalue = [13, 0]'),
 Text(0.18367667872306967, 0.4423076923076923, 'X[17] <= 0.5\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.18241861927976097, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.18493473816637837, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.18745085705299575, 0.4807692307692308, 'X[1] <= 49.5\ngini =
0.5\nsamples = 4\nvalue = [2, 2]'),
 Text(0.18619279760968704, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.18870891649630445, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.19122503538292185, 0.5192307692307693, 'X[14] <= 0.5\ngini =
0.48\nsamples = 5\nvalue = [2, 3]'),
 Text(0.18996697593961315, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.19248309482623055, 0.4807692307692308, 'X[22] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.19122503538292185, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.19374115426953922, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.19625727315615663, 0.5961538461538461, 'X[5] <= 38.5\ngini =
0.231\nsamples = 15\nvalue = [13, 2]'),
 Text(0.19499921371284792, 0.5576923076923077, 'X[1] <= 54.5\ngini =
0.133\nsamples = 14\nvalue = [13, 1]'),
 Text(0.19374115426953922, 0.5192307692307693, 'gini = 0.0\nsamples =
11\nvalue = [11, 0]'),
 Text(0.19625727315615663, 0.5192307692307693, 'X[44] <= 0.5\ngini =
```

```
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.19499921371284792, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.19751533259946533, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.19751533259946533, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.2269224720868061, 0.6346153846153846, 'X[19] <= 0.5\ngini =
0.491\nsamples = 134\nvalue = [58, 76]'),
 Text(0.21670073910992294, 0.5961538461538461, 'X[9] <= 2.5\ngini =
0.499\nsamples = 108\nvalue = [52, 56]'),
 Text(0.20710803585469414, 0.5576923076923077, 'X[34] <= 0.5\ngini =
0.463\nsamples = 55\nvalue = [20, 35]'),
 Text(0.20349111495518163, 0.5192307692307693, 'X[26] <= 0.5\ngini =
0.499\nsamples = 40\nvalue = [19, 21]'),
 Text(0.20003145148608273, 0.4807692307692308, 'X[8] <= 7.5\ngini =
0.477\nsamples = 28\nvalue = [17, 11]'),
 Text(0.19688630287781098, 0.4423076923076923, 'X[36] <= 0.5\ngini =
0.397\nsamples = 22\nvalue = [16, 6]'),
 Text(0.19437018399119357, 0.40384615384615385, 'X[5] <= 34.5\ngini =
0.219\nsamples = 16\nvalue = [14, 2]'),
 Text(0.1931121245478849, 0.36538461538461536, 'X[0] <= 3655.0\ngini =
0.124\nsamples = 15\nvalue = [14, 1]'),
 Text(0.1918540651045762, 0.3269230769230769, 'X[0] <= 3385.5\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.1905960056612675, 0.28846153846153844, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.1931121245478849, 0.28846153846153844, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.19437018399119357, 0.3269230769230769, 'gini = 0.0\nsamples =
13\nvalue = [13, 0]'),
 Text(0.19562824343450227, 0.36538461538461536, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.19940242176442838, 0.40384615384615385, 'X[4] <= 5.5\ngini =
0.444\nsamples = 6\nvalue = [2, 4]'),
 Text(0.19814436232111968, 0.36538461538461536, 'gini = 0.0\nsamples =
4\nvalue = [0, 4]'),
 Text(0.20066048120773708, 0.36538461538461536, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.20317660009435445, 0.4423076923076923, 'X[24] <= 0.5\ngini =
0.278\nsamples = 6\nvalue = [1, 5]'),
 Text(0.20191854065104575, 0.40384615384615385, 'gini = 0.0\nsamples =
5\nvalue = [0, 5]'),
 Text(0.20443465953766315, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.20695077842428056, 0.4807692307692308, 'X[1] <= 51.5\ngini =
0.278\nsamples = 12\nvalue = [2, 10]'),
 Text(0.20569271898097186, 0.4423076923076923, 'gini = 0.0\nsamples =
9\nvalue = [0, 9]'),
```

```
 Text(0.20820883786758923, 0.4423076923076923, 'X[8] <= 5.0\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.20695077842428056, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.20946689731089793, 0.40384615384615385, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.21072495675420663, 0.5192307692307693, 'X[5] <= 2.0\ngini =
0.124\nsamples = 15\nvalue = [1, 14]'),
 Text(0.20946689731089793, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.21198301619751533, 0.4807692307692308, 'gini = 0.0\nsamples =
14\nvalue = [0, 14]'),
 Text(0.22629344236515175, 0.5576923076923077, 'X[10] <= 6.5\ngini =
0.478\nsamples = 53\nvalue = [32, 21]'),
 Text(0.22173297688315774, 0.5192307692307693, 'X[24] <= 0.5\ngini =
0.499\nsamples = 40\nvalue = [21, 19]'),
 Text(0.21638622424909576, 0.4807692307692308, 'X[10] <= 0.5\ngini =
0.453\nsamples = 26\nvalue = [17, 9]'),
 Text(0.21324107564082403, 0.4423076923076923, 'X[14] <= 0.5\ngini =
0.32\nsamples = 5\nvalue = [1, 4]'),
 Text(0.21198301619751533, 0.40384615384615385, 'gini = 0.0\nsamples =
4\nvalue = [0, 4]'),
 Text(0.21449913508413274, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.2195313728573675, 0.4423076923076923, 'X[7] <= 15.5\ngini =
0.363\nsamples = 21\nvalue = [16, 5]'),
 Text(0.2170152539707501, 0.40384615384615385, 'X[25] <= 0.5\ngini =
0.208\nsamples = 17\nvalue = [15, 2]'),
 Text(0.2157571945274414, 0.36538461538461536, 'gini = 0.0\nsamples =
10\nvalue = [10, 0]'),
 Text(0.2182733134140588, 0.36538461538461536, 'X[10] <= 3.5\ngini =
0.408\nsamples = 7\nvalue = [5, 2]'),
 Text(0.2170152539707501, 0.3269230769230769, 'gini = 0.0\nsamples =
5\nvalue = [5, 0]'),
 Text(0.2195313728573675, 0.3269230769230769, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.22204749174398491, 0.40384615384615385, 'X[29] <= 0.5\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.2207894323006762, 0.36538461538461536, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.2233055511872936, 0.36538461538461536, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.2270797295172197, 0.4807692307692308, 'X[12] <= 0.5\ngini =
0.408\nsamples = 14\nvalue = [4, 10]'),
 Text(0.225821670073911, 0.4423076923076923, 'X[2] <= 3.5\ngini =
0.165\nsamples = 11\nvalue = [1, 10]'),
 Text(0.2245636106306023, 0.40384615384615385, 'gini = 0.0\nsamples =
10\nvalue = [0, 10]'),
 Text(0.2270797295172197, 0.40384615384615385, 'gini = 0.0\nsamples =
```

```
1\nvalue = [1, 0]'),
 Text(0.2283377889605284, 0.4423076923076923, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.23085390784714577, 0.5192307692307693, 'X[5] <= 3.5\ngini =
0.26\nsamples = 13\nvalue = [11, 2]'),
 Text(0.2295958484038371, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.23211196729045447, 0.4807692307692308, 'X[5] <= 32.5\ngini =
0.153\nsamples = 12\nvalue = [11, 1]'),
 Text(0.23085390784714577, 0.4423076923076923, 'gini = 0.0\nsamples =
10\nvalue = [10, 0]'),
 Text(0.23337002673376317, 0.4423076923076923, 'X[0] <= 4650.0\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.23211196729045447, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.23462808617707187, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.23714420506368927, 0.5961538461538461, 'X[9] <= 0.5\ngini =
0.355\nsamples = 26\nvalue = [6, 20]'),
 Text(0.23462808617707187, 0.5576923076923077, 'X[13] <= 0.5\ngini =
0.32\nsamples = 5\nvalue = [4, 1]'),
 Text(0.23337002673376317, 0.5192307692307693, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.23588614562038057, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.23966032395030665, 0.5576923076923077, 'X[7] <= 16.5\ngini =
0.172\nsamples = 21\nvalue = [2, 19]'),
 Text(0.23840226450699795, 0.5192307692307693, 'gini = 0.0\nsamples =
18\nvalue = [0, 18]'),
 Text(0.24091838339361535, 0.5192307692307693, 'X[5] <= 24.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.23966032395030665, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.24217644283692405, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.24217644283692405, 0.6730769230769231, 'X[4] <= 8.5\ngini =
0.231\nsamples = 15\nvalue = [2, 13]'),
 Text(0.24091838339361535, 0.6346153846153846, 'X[7] <= 1.0\ngini =
0.133\nsamples = 14\nvalue = [1, 13]'),
 Text(0.23966032395030665, 0.5961538461538461, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.24217644283692405, 0.5961538461538461, 'gini = 0.0\nsamples =
13\nvalue = [0, 13]'),
 Text(0.24343450228023275, 0.6346153846153846, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.2677307752791319, 0.7115384615384616, 'X[24] <= 0.5\ngini =
0.442\nsamples = 115\nvalue = [38, 77]'),
 Text(0.25900298789117787, 0.6730769230769231, 'X[8] <= 6.5\ngini =
0.361\nsamples = 76\nvalue = [18, 58]'),
```

```
 Text(0.25318446296587516, 0.6346153846153846, 'X[19] <= 0.5\ngini =
0.449\nsamples = 44\nvalue = [15, 29]'),
 Text(0.24909576977512188, 0.5961538461538461, 'X[10] <= 5.5\ngini =
0.375\nsamples = 36\nvalue = [9, 27]'),
 Text(0.24469256172354142, 0.5576923076923077, 'X[3] <= 2026.0\ngini =
0.159\nsamples = 23\nvalue = [2, 21]'),
 Text(0.24343450228023275, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.24595062116685013, 0.5192307692307693, 'X[0] <= 1122.5\ngini =
0.087\nsamples = 22\nvalue = [1, 21]'),
 Text(0.24469256172354142, 0.4807692307692308, 'X[43] <= 0.5\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.24343450228023275, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.24595062116685013, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.24720868061015883, 0.4807692307692308, 'gini = 0.0\nsamples =
20\nvalue = [0, 20]'),
 Text(0.25349897782670233, 0.5576923076923077, 'X[1] <= 51.0\ngini =
0.497\nsamples = 13\nvalue = [7, 6]'),
 Text(0.25098285894008493, 0.5192307692307693, 'X[4] <= 4.5\ngini =
0.375\nsamples = 8\nvalue = [6, 2]'),
 Text(0.24972479949677623, 0.4807692307692308, 'gini = 0.0\nsamples =
5\nvalue = [5, 0]'),
 Text(0.25224091838339363, 0.4807692307692308, 'X[1] <= 46.0\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.25098285894008493, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.25349897782670233, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.2560150967133197, 0.5192307692307693, 'X[0] <= 8447.5\ngini =
0.32\nsamples = 5\nvalue = [1, 4]'),
 Text(0.25475703727001103, 0.4807692307692308, 'gini = 0.0\nsamples =
4\nvalue = [0, 4]'),
 Text(0.2572731561566284, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.2572731561566284, 0.5961538461538461, 'X[4] <= 8.5\ngini =
0.375\nsamples = 8\nvalue = [6, 2]'),
 Text(0.2560150967133197, 0.5576923076923077, 'gini = 0.0\nsamples =
6\nvalue = [6, 0]'),
 Text(0.2585312155999371, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.2648215128164806, 0.6346153846153846, 'X[5] <= 34.5\ngini =
0.17\nsamples = 32\nvalue = [3, 29]'),
 Text(0.2623053939298632, 0.5961538461538461, 'X[0] <= 8765.0\ngini =
0.069\nsamples = 28\nvalue = [1, 27]'),
 Text(0.2610473344865545, 0.5576923076923077, 'gini = 0.0\nsamples =
26\nvalue = [0, 26]'),
 Text(0.2635634533731719, 0.5576923076923077, 'X[1] <= 46.5\ngini =
```

```
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.2623053939298632, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.2648215128164806, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.267337631703098, 0.5961538461538461, 'X[4] <= 2.0\ngini = 0.5\
nsamples = 4\nvalue = [2, 2]'),
 Text(0.2660795722597893, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.2685956911464067, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.276458562667086, 0.6730769230769231, 'X[7] <= 5.5\ngini = 0.5\
nsamples = 39\nvalue = [20, 19]'),
 Text(0.27111181003302404, 0.6346153846153846, 'X[14] <= 0.5\ngini =
0.337\nsamples = 14\nvalue = [11, 3]'),
 Text(0.2698537505897154, 0.5961538461538461, 'gini = 0.0\nsamples =
8\nvalue = [8, 0]'),
 Text(0.27236986947633274, 0.5961538461538461, 'X[5] <= 16.0\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.27111181003302404, 0.5576923076923077, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.27362792891964144, 0.5576923076923077, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.281805315301148, 0.6346153846153846, 'X[33] <= 0.5\ngini =
0.461\nsamples = 25\nvalue = [9, 16]'),
 Text(0.2805472558578393, 0.5961538461538461, 'X[13] <= 0.5\ngini =
0.423\nsamples = 23\nvalue = [7, 16]'),
 Text(0.27614404780625884, 0.5576923076923077, 'X[11] <= 0.5\ngini =
0.291\nsamples = 17\nvalue = [3, 14]'),
 Text(0.27362792891964144, 0.5192307692307693, 'X[32] <= 0.5\ngini =
0.133\nsamples = 14\nvalue = [1, 13]'),
 Text(0.27236986947633274, 0.4807692307692308, 'gini = 0.0\nsamples =
13\nvalue = [0, 13]'),
 Text(0.27488598836295014, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.27866016669287624, 0.5192307692307693, 'X[37] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.27740210724956754, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.27991822613618494, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.2849504639094197, 0.5576923076923077, 'X[1] <= 49.0\ngini =
0.444\nsamples = 6\nvalue = [4, 2]'),
 Text(0.28369240446611105, 0.5192307692307693, 'X[2] <= 1.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.28243434502280235, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.2849504639094197, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
```

```
 Text(0.2862085233527284, 0.5192307692307693, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.2830633747444567, 0.5961538461538461, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.3318770640037742, 0.8269230769230769, 'X[1] <= 54.5\ngini =
0.476\nsamples = 244\nvalue = [149, 95]'),
 Text(0.30709427582953297, 0.7884615384615384, 'X[1] <= 26.5\ngini =
0.485\nsamples = 210\nvalue = [123, 87]'),
 Text(0.2887246422393458, 0.75, 'X[7] <= 1.5\ngini = 0.381\nsamples =
43\nvalue = [32, 11]'),
 Text(0.2843214341877654, 0.7115384615384616, 'X[6] <= 2.5\ngini =
0.444\nsamples = 6\nvalue = [2, 4]'),
 Text(0.2830633747444567, 0.6730769230769231, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.28557949363107404, 0.6730769230769231, 'X[11] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.2843214341877654, 0.6346153846153846, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.28683755307438275, 0.6346153846153846, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.29312785029092625, 0.7115384615384616, 'X[9] <= 1.5\ngini =
0.307\nsamples = 37\nvalue = [30, 7]'),
 Text(0.29061173140430885, 0.6730769230769231, 'X[19] <= 0.5\ngini =
0.496\nsamples = 11\nvalue = [6, 5]'),
 Text(0.28935367196100015, 0.6346153846153846, 'X[2] <= 1.5\ngini =
0.375\nsamples = 8\nvalue = [6, 2]'),
 Text(0.28809561251769145, 0.5961538461538461, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.29061173140430885, 0.5961538461538461, 'X[35] <= 0.5\ngini =
0.245\nsamples = 7\nvalue = [6, 1]'),
 Text(0.28935367196100015, 0.5576923076923077, 'gini = 0.0\nsamples =
6\nvalue = [6, 0]'),
 Text(0.29186979084761755, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.29186979084761755, 0.6346153846153846, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.29564396917754365, 0.6730769230769231, 'X[11] <= 0.5\ngini =
0.142\nsamples = 26\nvalue = [24, 2]'),
 Text(0.29438590973423495, 0.6346153846153846, 'gini = 0.0\nsamples =
17\nvalue = [17, 0]'),
 Text(0.29690202862085235, 0.6346153846153846, 'X[8] <= 4.5\ngini =
0.346\nsamples = 9\nvalue = [7, 2]'),
 Text(0.29564396917754365, 0.5961538461538461, 'X[22] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.29438590973423495, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.29690202862085235, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.29816008806416106, 0.5961538461538461, 'gini = 0.0\nsamples =
```

```
6\nvalue = [6, 0]'),
 Text(0.3254639094197201, 0.75, 'X[18] <= 0.5\ngini = 0.496\nsamples =
167\nvalue = [91, 76]'),
 Text(0.3106227394244378, 0.7115384615384616, 'X[3] <= 16614.0\ngini =
0.5\nsamples = 130\nvalue = [65, 65]'),
 Text(0.3031923258373958, 0.6730769230769231, 'X[6] <= 0.5\ngini =
0.32\nsamples = 15\nvalue = [12, 3]'),
 Text(0.3019342663940871, 0.6346153846153846, 'X[19] <= 0.5\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.3006762069507784, 0.5961538461538461, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.3031923258373958, 0.5961538461538461, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.3044503852807045, 0.6346153846153846, 'gini = 0.0\nsamples =
11\nvalue = [11, 0]'),
 Text(0.3180531530114798, 0.6730769230769231, 'X[7] <= 7.5\ngini =
0.497\nsamples = 115\nvalue = [53, 62]'),
 Text(0.3102689102060072, 0.6346153846153846, 'X[0] <= 3103.0\ngini =
0.48\nsamples = 50\nvalue = [30, 20]'),
 Text(0.3057084447240132, 0.5961538461538461, 'X[2] <= 1.5\ngini =
0.231\nsamples = 15\nvalue = [13, 2]'),
 Text(0.3044503852807045, 0.5576923076923077, 'X[1] <= 42.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.3031923258373958, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.3057084447240132, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.3069665041673219, 0.5576923076923077, 'gini = 0.0\nsamples =
12\nvalue = [12, 0]'),
 Text(0.31482937568800123, 0.5961538461538461, 'X[25] <= 0.5\ngini =
0.5\nsamples = 35\nvalue = [17, 18]'),
 Text(0.31136971221890236, 0.5576923076923077, 'X[29] <= 0.5\ngini =
0.453\nsamples = 26\nvalue = [9, 17]'),
 Text(0.3082245636106306, 0.5192307692307693, 'X[35] <= 0.5\ngini =
0.291\nsamples = 17\nvalue = [3, 14]'),
 Text(0.3057084447240132, 0.4807692307692308, 'X[4] <= 1.0\ngini =
0.133\nsamples = 14\nvalue = [1, 13]'),
 Text(0.3044503852807045, 0.4423076923076923, 'X[44] <= 0.5\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.3031923258373958, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.3057084447240132, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.3069665041673219, 0.4423076923076923, 'gini = 0.0\nsamples =
12\nvalue = [0, 12]'),
 Text(0.310740682497248, 0.4807692307692308, 'X[8] <= 5.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.3094826230539393, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.3119987419405567, 0.4423076923076923, 'gini = 0.0\nsamples =
```

```
1\nvalue = [0, 1]'),
 Text(0.31451486082717406, 0.5192307692307693, 'X[5] <= 20.5\ngini =
0.444\nsamples = 9\nvalue = [6, 3]'),
 Text(0.3132568013838654, 0.4807692307692308, 'gini = 0.0\nsamples =
6\nvalue = [6, 0]'),
 Text(0.31577292027048276, 0.4807692307692308, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.31828903915710016, 0.5576923076923077, 'X[20] <= 0.5\ngini =
0.198\nsamples = 9\nvalue = [8, 1]'),
 Text(0.31703097971379146, 0.5192307692307693, 'gini = 0.0\nsamples =
8\nvalue = [8, 0]'),
 Text(0.31954709860040886, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.32583739581695237, 0.6346153846153846, 'X[9] <= 3.5\ngini =
0.457\nsamples = 65\nvalue = [23, 42]'),
 Text(0.32332127693033497, 0.5961538461538461, 'X[1] <= 31.5\ngini =
0.492\nsamples = 48\nvalue = [21, 27]'),
 Text(0.32206321748702627, 0.5576923076923077, 'gini = 0.0\nsamples =
8\nvalue = [0, 8]'),
 Text(0.32457933637364367, 0.5576923076923077, 'X[1] <= 50.5\ngini =
0.499\nsamples = 40\nvalue = [21, 19]'),
 Text(0.32332127693033497, 0.5192307692307693, 'X[5] <= 22.5\ngini =
0.48\nsamples = 35\nvalue = [21, 14]'),
 Text(0.31828903915710016, 0.4807692307692308, 'X[4] <= 1.5\ngini =
0.397\nsamples = 22\nvalue = [16, 6]'),
 Text(0.31703097971379146, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.31954709860040886, 0.4423076923076923, 'X[3] <= 16769.5\ngini
= 0.32\nsamples = 20\nvalue = [16, 4]'),
 Text(0.31828903915710016, 0.40384615384615385, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.32080515804371756, 0.40384615384615385, 'X[14] <= 0.5\ngini =
0.198\nsamples = 18\nvalue = [16, 2]'),
 Text(0.31954709860040886, 0.36538461538461536, 'gini = 0.0\nsamples =
14\nvalue = [14, 0]'),
 Text(0.32206321748702627, 0.36538461538461536, 'X[5] <= 17.5\ngini =
0.5\nsamples = 4\nvalue = [2, 2]'),
 Text(0.32080515804371756, 0.3269230769230769, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.32332127693033497, 0.3269230769230769, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.32835351470356977, 0.4807692307692308, 'X[10] <= 4.5\ngini =
0.473\nsamples = 13\nvalue = [5, 8]'),
 Text(0.32709545526026107, 0.4423076923076923, 'X[24] <= 0.5\ngini =
0.408\nsamples = 7\nvalue = [5, 2]'),
 Text(0.32583739581695237, 0.40384615384615385, 'X[37] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.32457933637364367, 0.36538461538461536, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
```

```
 Text(0.32709545526026107, 0.36538461538461536, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.32835351470356977, 0.40384615384615385, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.3296115741468784, 0.4423076923076923, 'gini = 0.0\nsamples =
6\nvalue = [0, 6]'),
 Text(0.32583739581695237, 0.5192307692307693, 'gini = 0.0\nsamples =
5\nvalue = [0, 5]'),
 Text(0.32835351470356977, 0.5961538461538461, 'X[0] <= 214.5\ngini =
0.208\nsamples = 17\nvalue = [2, 15]'),
 Text(0.32709545526026107, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.3296115741468784, 0.5576923076923077, 'X[5] <= 38.5\ngini =
0.117\nsamples = 16\nvalue = [1, 15]'),
 Text(0.32835351470356977, 0.5192307692307693, 'gini = 0.0\nsamples =
15\nvalue = [0, 15]'),
 Text(0.3308696335901871, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.3403050794150024, 0.7115384615384616, 'X[7] <= 16.5\ngini =
0.418\nsamples = 37\nvalue = [26, 11]'),
 Text(0.3371599308067306, 0.6730769230769231, 'X[24] <= 0.5\ngini =
0.32\nsamples = 30\nvalue = [24, 6]'),
 Text(0.3346438119201132, 0.6346153846153846, 'X[3] <= 17090.0\ngini =
0.172\nsamples = 21\nvalue = [19, 2]'),
 Text(0.3333857524768045, 0.5961538461538461, 'X[13] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.3321276930334958, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.3346438119201132, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.3359018713634219, 0.5961538461538461, 'gini = 0.0\nsamples =
18\nvalue = [18, 0]'),
 Text(0.339676049693348, 0.6346153846153846, 'X[30] <= 0.5\ngini =
0.494\nsamples = 9\nvalue = [5, 4]'),
 Text(0.3384179902500393, 0.5961538461538461, 'X[8] <= 5.0\ngini =
0.444\nsamples = 6\nvalue = [2, 4]'),
 Text(0.3371599308067306, 0.5576923076923077, 'X[6] <= 3.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.3359018713634219, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.3384179902500393, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.339676049693348, 0.5576923076923077, 'gini = 0.0\nsamples = 3\
nvalue = [0, 3]'),
 Text(0.3409341091366567, 0.5961538461538461, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.3434502280232741, 0.6730769230769231, 'X[31] <= 0.5\ngini =
0.408\nsamples = 7\nvalue = [2, 5]'),
 Text(0.34219216857996543, 0.6346153846153846, 'gini = 0.0\nsamples =
```

```
5\nvalue = [0, 5]'),
 Text(0.3447082874665828, 0.6346153846153846, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.356659852178015 44, 0.7884615384615384, 'X[16] <= 0.5\ngini =
0.36\nsamples = 34\nvalue = [26, 8]'),
 Text(0.352256644126435, 0.75, 'X[7] <= 12.0\ngini = 0.211\nsamples =
25\nvalue = [22, 3]'),
 Text(0.3497405252398176, 0.7115384615384616, 'X[7] <= 4.5\ngini =
0.091\nsamples = 21\nvalue = [20, 1]'),
 Text(0.3484824657965089, 0.6730769230769231, 'X[8] <= 5.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.3472244063532002, 0.6346153846153846, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.3497405252398176, 0.6346153846153846, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.3509985846831263, 0.6730769230769231, 'gini = 0.0\nsamples =
18\nvalue = [18, 0]'),
 Text(0.3547727630130524, 0.7115384615384616, 'X[7] <= 14.5\ngini =
0.5\nsamples = 4\nvalue = [2, 2]'),
 Text(0.3535147035697437, 0.6730769230769231, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.3560308224563611, 0.6730769230769231, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.361063060229595 83, 0.75, 'X[3] <= 18435.0\ngini = 0.494\
nsamples = 9\nvalue = [4, 5]'),
 Text(0.359805000786287 13, 0.7115384615384616, 'X[31] <= 0.5\ngini =
0.444\nsamples = 6\nvalue = [4, 2]'),
 Text(0.358546941342978 43, 0.6730769230769231, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.361063060229595 83, 0.6730769230769231, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.362321119672904 53, 0.7115384615384616, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.442768123918855 17, 0.8653846153846154, 'X[24] <= 0.5\ngini =
0.491\nsamples = 472\nvalue = [204, 268]'),
 Text(0.419425224091838 36, 0.8269230769230769, 'X[5] <= 27.5\ngini =
0.477\nsamples = 305\nvalue = [120, 185]'),
 Text(0.394126434974052 53, 0.7884615384615384, 'X[3] <= 8268.5\ngini =
0.491\nsamples = 215\nvalue = [93, 122]'),
 Text(0.372542852649787 7, 0.75, 'X[2] <= 1.5\ngini = 0.497\nsamples =
82\nvalue = [44, 38]'),
 Text(0.364837238559521 94, 0.7115384615384616, 'X[1] <= 48.5\ngini =
0.397\nsamples = 22\nvalue = [6, 16]'),
 Text(0.363579179116213 24, 0.6730769230769231, 'X[20] <= 0.5\ngini =
0.32\nsamples = 20\nvalue = [4, 16]'),
 Text(0.361063060229595 83, 0.6346153846153846, 'X[10] <= 8.0\ngini =
0.208\nsamples = 17\nvalue = [2, 15]'),
 Text(0.359805000786287 13, 0.5961538461538461, 'gini = 0.0\nsamples =
13\nvalue = [0, 13]'),
```

```
 Text(0.36232111967290453, 0.5961538461538461, 'X[4] <= 2.5\ngini =
0.5\nsamples = 4\nvalue = [2, 2]'),
 Text(0.36106306022959583, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.36357917911621324, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.36609529800283064, 0.6346153846153846, 'X[12] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.36483723855952194, 0.5961538461538461, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.36735335744613934, 0.5961538461538461, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.36609529800283064, 0.6730769230769231, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.38024846674005347, 0.7115384615384616, 'X[37] <= 0.5\ngini =
0.464\nsamples = 60\nvalue = [38, 22]'),
 Text(0.37678880327095454, 0.6730769230769231, 'X[10] <= 7.5\ngini =
0.496\nsamples = 44\nvalue = [24, 20]'),
 Text(0.37553074382764584, 0.6346153846153846, 'X[6] <= 1.5\ngini =
0.497\nsamples = 37\nvalue = [17, 20]'),
 Text(0.36986947633275674, 0.5961538461538461, 'X[25] <= 0.5\ngini =
0.472\nsamples = 21\nvalue = [13, 8]'),
 Text(0.36735335744613934, 0.5576923076923077, 'X[3] <= 8138.5\ngini =
0.165\nsamples = 11\nvalue = [10, 1]'),
 Text(0.36609529800283064, 0.5192307692307693, 'gini = 0.0\nsamples =
10\nvalue = [10, 0]'),
 Text(0.36861141688944804, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.3723855952193741, 0.5576923076923077, 'X[3] <= 4498.5\ngini =
0.42\nsamples = 10\nvalue = [3, 7]'),
 Text(0.37112753577606544, 0.5192307692307693, 'gini = 0.0\nsamples =
5\nvalue = [0, 5]'),
 Text(0.3736436546626828, 0.5192307692307693, 'X[3] <= 6418.0\ngini =
0.48\nsamples = 5\nvalue = [3, 2]'),
 Text(0.3723855952193741, 0.4807692307692308, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.3749017141059915, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.381192011322535, 0.5961538461538461, 'X[31] <= 0.5\ngini =
0.375\nsamples = 16\nvalue = [4, 12]'),
 Text(0.3799339518792263, 0.5576923076923077, 'X[8] <= 8.5\ngini =
0.245\nsamples = 14\nvalue = [2, 12]'),
 Text(0.3786758924359176, 0.5192307692307693, 'X[1] <= 55.5\ngini =
0.142\nsamples = 13\nvalue = [1, 12]'),
 Text(0.3774178329926089, 0.4807692307692308, 'gini = 0.0\nsamples =
11\nvalue = [0, 11]'),
 Text(0.3799339518792263, 0.4807692307692308, 'X[4] <= 2.5\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.3786758924359176, 0.4423076923076923, 'gini = 0.0\nsamples =
```

```
1\nvalue = [0, 1]'),
 Text(0.381192011322535, 0.4423076923076923, 'gini = 0.0\nsamples = 1\
nvalue = [1, 0]'),
 Text(0.381192011322535, 0.5192307692307693, 'gini = 0.0\nsamples = 1\
nvalue = [1, 0]'),
 Text(0.3824500707658437, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.37804686271426324, 0.6346153846153846, 'gini = 0.0\nsamples =
7\nvalue = [7, 0]'),
 Text(0.3837081302091524, 0.6730769230769231, 'X[3] <= 1621.5\ngini =
0.219\nsamples = 16\nvalue = [14, 2]'),
 Text(0.3824500707658437, 0.6346153846153846, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.3849661896524611, 0.6346153846153846, 'X[20] <= 0.5\ngini =
0.124\nsamples = 15\nvalue = [14, 1]'),
 Text(0.3837081302091524, 0.5961538461538461, 'gini = 0.0\nsamples =
14\nvalue = [14, 0]'),
 Text(0.3862242490957698, 0.5961538461538461, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.41571001729831736, 0.75, 'X[1] <= 49.5\ngini = 0.465\nsamples
= 133\nvalue = [49, 84]'),
 Text(0.40619594275829535, 0.7115384615384616, 'X[3] <= 14718.0\ngini
= 0.485\nsamples = 109\nvalue = [45, 64]'),
 Text(0.3972322692247209, 0.6730769230769231, 'X[0] <= 6145.5\ngini =
0.428\nsamples = 58\nvalue = [18, 40]'),
 Text(0.3931435760339676, 0.6346153846153846, 'X[5] <= 10.5\ngini =
0.489\nsamples = 40\nvalue = [17, 23]'),
 Text(0.38874036798238715, 0.5961538461538461, 'X[0] <= 1722.0\ngini =
0.465\nsamples = 19\nvalue = [12, 7]'),
 Text(0.3862242490957698, 0.5576923076923077, 'X[6] <= 2.0\ngini =
0.32\nsamples = 5\nvalue = [1, 4]'),
 Text(0.3849661896524611, 0.5192307692307693, 'gini = 0.0\nsamples =
4\nvalue = [0, 4]'),
 Text(0.38748230853907845, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.39125648686900455, 0.5576923076923077, 'X[3] <= 8416.0\ngini =
0.337\nsamples = 14\nvalue = [11, 3]'),
 Text(0.38999842742569585, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.39251454631231325, 0.5192307692307693, 'X[10] <= 3.5\ngini =
0.26\nsamples = 13\nvalue = [11, 2]'),
 Text(0.39125648686900455, 0.4807692307692308, 'X[32] <= 0.5\ngini =
0.48\nsamples = 5\nvalue = [3, 2]'),
 Text(0.38999842742569585, 0.4423076923076923, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.39251454631231325, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.39377260575562195, 0.4807692307692308, 'gini = 0.0\nsamples =
8\nvalue = [8, 0]'),
```

```
 Text(0.39754678408554806, 0.5961538461538461, 'X[4] <= 5.5\ngini =
0.363\nsamples = 21\nvalue = [5, 16]'),
 Text(0.39628872464223935, 0.5576923076923077, 'gini = 0.0\nsamples =
9\nvalue = [0, 9]'),
 Text(0.39880484352885676, 0.5576923076923077, 'X[7] <= 12.0\ngini =
0.486\nsamples = 12\nvalue = [5, 7]'),
 Text(0.39754678408554806, 0.5192307692307693, 'X[7] <= 6.5\ngini =
0.469\nsamples = 8\nvalue = [5, 3]'),
 Text(0.39628872464223935, 0.4807692307692308, 'X[44] <= 0.5\ngini =
0.48\nsamples = 5\nvalue = [2, 3]'),
 Text(0.39503066519893065, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.39754678408554806, 0.4423076923076923, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.39880484352885676, 0.4807692307692308, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.40006290297216546, 0.5192307692307693, 'gini = 0.0\nsamples =
4\nvalue = [0, 4]'),
 Text(0.40132096241547416, 0.6346153846153846, 'X[10] <= 8.5\ngini =
0.105\nsamples = 18\nvalue = [1, 17]'),
 Text(0.40006290297216546, 0.5961538461538461, 'gini = 0.0\nsamples =
16\nvalue = [0, 16]'),
 Text(0.4025790218587828, 0.5961538461538461, 'X[3] <= 10331.0\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.40132096241547416, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.4038370813020915, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.4151596162918698, 0.6730769230769231, 'X[10] <= 4.5\ngini =
0.498\nsamples = 51\nvalue = [27, 24]'),
 Text(0.410127378518635, 0.6346153846153846, 'X[0] <= 1714.5\ngini =
0.436\nsamples = 28\nvalue = [19, 9]'),
 Text(0.4076112596320176, 0.5961538461538461, 'X[7] <= 12.0\ngini =
0.444\nsamples = 9\nvalue = [3, 6]'),
 Text(0.4063532001887089, 0.5576923076923077, 'X[4] <= 6.0\ngini =
0.375\nsamples = 4\nvalue = [3, 1]'),
 Text(0.4050951407454002, 0.5192307692307693, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.4076112596320176, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.4088693190753263, 0.5576923076923077, 'gini = 0.0\nsamples =
5\nvalue = [0, 5]'),
 Text(0.4126434974052524, 0.5961538461538461, 'X[4] <= 3.5\ngini =
0.266\nsamples = 19\nvalue = [16, 3]'),
 Text(0.4113854379619437, 0.5576923076923077, 'X[8] <= 2.0\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.410127378518635, 0.5192307692307693, 'X[43] <= 0.5\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.4088693190753263, 0.4807692307692308, 'gini = 0.0\nsamples =
```

```
3\nvalue = [0, 3]'),
 Text(0.4113854379619437, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.4126434974052524, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.4139015568485611, 0.5576923076923077, 'gini = 0.0\nsamples =
13\nvalue = [13, 0]'),
 Text(0.42019185406510456, 0.6346153846153846, 'X[8] <= 2.5\ngini =
0.454\nsamples = 23\nvalue = [8, 15]'),
 Text(0.41767573517848716, 0.5961538461538461, 'X[9] <= 3.5\ngini =
0.469\nsamples = 8\nvalue = [5, 3]'),
 Text(0.41641767573517846, 0.5576923076923077, 'X[26] <= 0.5\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.4151596162918698, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.41767573517848716, 0.5192307692307693, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.41893379462179586, 0.5576923076923077, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.42270797295172197, 0.5961538461538461, 'X[21] <= 0.5\ngini =
0.32\nsamples = 15\nvalue = [3, 12]'),
 Text(0.42144991350841327, 0.5576923076923077, 'gini = 0.0\nsamples =
9\nvalue = [0, 9]'),
 Text(0.42396603239503067, 0.5576923076923077, 'X[5] <= 16.5\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.42270797295172197, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.42522409183833937, 0.5192307692307693, 'X[3] <= 15729.5\ngini
= 0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.42396603239503067, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.42648215128164807, 0.4807692307692308, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.42522409183833937, 0.7115384615384616, 'X[4] <= 3.5\ngini =
0.278\nsamples = 24\nvalue = [4, 20]'),
 Text(0.42396603239503067, 0.6730769230769231, 'X[0] <= 4749.0\ngini =
0.494\nsamples = 9\nvalue = [4, 5]'),
 Text(0.42270797295172197, 0.6346153846153846, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.42522409183833937, 0.6346153846153846, 'gini = 0.0\nsamples =
5\nvalue = [0, 5]'),
 Text(0.42648215128164807, 0.6730769230769231, 'gini = 0.0\nsamples =
15\nvalue = [0, 15]'),
 Text(0.4447240132096242, 0.7884615384615384, 'X[10] <= 8.5\ngini =
0.42\nsamples = 90\nvalue = [27, 63]'),
 Text(0.4434659537663155, 0.75, 'X[3] <= 18976.5\ngini = 0.444\
nsamples = 81\nvalue = [27, 54]'),
 Text(0.439062745714735, 0.7115384615384616, 'X[3] <= 2233.5\ngini =
0.422\nsamples = 76\nvalue = [23, 53]'),
```

```
 Text(0.4340305079415002, 0.6730769230769231, 'X[0] <= 4277.5\ngini =
0.444\nsamples = 6\nvalue = [4, 2]'),
 Text(0.4327724484981915, 0.6346153846153846, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.4352885673848089, 0.6346153846153846, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.44409498348796983, 0.6730769230769231, 'X[44] <= 0.5\ngini =
0.396\nsamples = 70\nvalue = [19, 51]'),
 Text(0.4378046862714263, 0.6346153846153846, 'X[3] <= 4939.5\ngini =
0.467\nsamples = 43\nvalue = [16, 27]'),
 Text(0.4365466268281176, 0.5961538461538461, 'gini = 0.0\nsamples =
10\nvalue = [0, 10]'),
 Text(0.439062745714735, 0.5961538461538461, 'X[35] <= 0.5\ngini =
0.5\nsamples = 33\nvalue = [16, 17]'),
 Text(0.4340305079415002, 0.5576923076923077, 'X[7] <= 13.5\ngini =
0.469\nsamples = 24\nvalue = [9, 15]'),
 Text(0.4302563296115742, 0.5192307692307693, 'X[7] <= 0.5\ngini =
0.26\nsamples = 13\nvalue = [2, 11]'),
 Text(0.42899827016826547, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.4315143890548828, 0.4807692307692308, 'X[0] <= 2460.0\ngini =
0.153\nsamples = 12\nvalue = [1, 11]'),
 Text(0.4302563296115742, 0.4423076923076923, 'X[37] <= 0.5\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.42899827016826547, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.4315143890548828, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.4327724484981915, 0.4423076923076923, 'gini = 0.0\nsamples =
10\nvalue = [0, 10]'),
 Text(0.4378046862714263, 0.5192307692307693, 'X[1] <= 51.5\ngini =
0.463\nsamples = 11\nvalue = [7, 4]'),
 Text(0.4365466268281176, 0.4807692307692308, 'X[16] <= 0.5\ngini =
0.219\nsamples = 8\nvalue = [7, 1]'),
 Text(0.4352885673848089, 0.4423076923076923, 'gini = 0.0\nsamples =
7\nvalue = [7, 0]'),
 Text(0.4378046862714263, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.439062745714735, 0.4807692307692308, 'gini = 0.0\nsamples = 3\
nvalue = [0, 3]'),
 Text(0.44409498348796983, 0.5576923076923077, 'X[4] <= 7.5\ngini =
0.346\nsamples = 9\nvalue = [7, 2]'),
 Text(0.44283692404466113, 0.5192307692307693, 'X[11] <= 0.5\ngini =
0.219\nsamples = 8\nvalue = [7, 1]'),
 Text(0.4415788646013524, 0.4807692307692308, 'gini = 0.0\nsamples =
7\nvalue = [7, 0]'),
 Text(0.44409498348796983, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.4453530429312785, 0.5192307692307693, 'gini = 0.0\nsamples =
```

```
1\nvalue = [0, 1]'),
 Text(0.4503852807045133, 0.6346153846153846, 'X[19] <= 0.5\ngini =
0.198\nsamples = 27\nvalue = [3, 24]'),
 Text(0.4478691618178959, 0.5961538461538461, 'X[15] <= 0.5\ngini =
0.083\nsamples = 23\nvalue = [1, 22]'),
 Text(0.4466111023745872, 0.5576923076923077, 'gini = 0.0\nsamples =
20\nvalue = [0, 20]'),
 Text(0.4491272212612046, 0.5576923076923077, 'X[9] <= 2.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.4478691618178959, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.4503852807045133, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.4529013995911307, 0.5961538461538461, 'X[5] <= 35.0\ngini =
0.5\nsamples = 4\nvalue = [2, 2]'),
 Text(0.451643340147822, 0.5576923076923077, 'gini = 0.0\nsamples = 2\
nvalue = [2, 0]'),
 Text(0.4541594590344394, 0.5576923076923077, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.4478691618178959, 0.7115384615384616, 'X[4] <= 5.5\ngini =
0.32\nsamples = 5\nvalue = [4, 1]'),
 Text(0.4466111023745872, 0.6730769230769231, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.4491272212612046, 0.6730769230769231, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.4459820726529328, 0.75, 'gini = 0.0\nsamples = 9\nvalue = [0,
9]'),
 Text(0.466111023745872, 0.8269230769230769, 'X[7] <= 8.5\ngini = 0.5\
nsamples = 167\nvalue = [84, 83]'),
 Text(0.45730460764271114, 0.7884615384615384, 'X[0] <= 1186.5\ngini =
0.465\nsamples = 79\nvalue = [29, 50]'),
 Text(0.4541594590344394, 0.75, 'X[0] <= 373.5\ngini = 0.32\nsamples =
10\nvalue = [8, 2]'),
 Text(0.4529013995911307, 0.7115384615384616, 'X[43] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.451643340147822, 0.6730769230769231, 'gini = 0.0\nsamples = 1\
nvalue = [1, 0]'),
 Text(0.4541594590344394, 0.6730769230769231, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.4554175184777481, 0.7115384615384616, 'gini = 0.0\nsamples =
7\nvalue = [7, 0]'),
 Text(0.46044975625098283, 0.75, 'X[1] <= 24.5\ngini = 0.423\nsamples
= 69\nvalue = [21, 48]'),
 Text(0.4579336373643655, 0.7115384615384616, 'X[10] <= 3.5\ngini =
0.375\nsamples = 8\nvalue = [6, 2]'),
 Text(0.4566755779210568, 0.6730769230769231, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.4591916968076742, 0.6730769230769231, 'gini = 0.0\nsamples =
6\nvalue = [6, 0]'),
```

```
 Text(0.46296587513760024, 0.7115384615384616, 'X[17] <= 0.5\ngini =
0.371\nsamples = 61\nvalue = [15, 46]'),
 Text(0.46170781569429153, 0.6730769230769231, 'X[1] <= 57.5\ngini =
0.425\nsamples = 49\nvalue = [15, 34]'),
 Text(0.46044975625098283, 0.6346153846153846, 'X[35] <= 0.5\ngini =
0.4\nsamples = 47\nvalue = [13, 34]'),
 Text(0.4591916968076742, 0.5961538461538461, 'X[10] <= 6.5\ngini =
0.456\nsamples = 37\nvalue = [13, 24]'),
 Text(0.4566755779210568, 0.5576923076923077, 'X[5] <= 33.5\ngini =
0.32\nsamples = 25\nvalue = [5, 20]'),
 Text(0.4554175184777481, 0.5192307692307693, 'X[3] <= 18936.5\ngini =
0.227\nsamples = 23\nvalue = [3, 20]'),
 Text(0.4541594590344394, 0.4807692307692308, 'X[20] <= 0.5\ngini =
0.165\nsamples = 22\nvalue = [2, 20]'),
 Text(0.4529013995911307, 0.4423076923076923, 'gini = 0.0\nsamples =
16\nvalue = [0, 16]'),
 Text(0.4554175184777481, 0.4423076923076923, 'X[3] <= 9885.0\ngini =
0.444\nsamples = 6\nvalue = [2, 4]'),
 Text(0.4541594590344394, 0.40384615384615385, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.4566755779210568, 0.40384615384615385, 'gini = 0.0\nsamples =
4\nvalue = [0, 4]'),
 Text(0.4566755779210568, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.4579336373643655, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.46170781569429153, 0.5576923076923077, 'X[3] <= 13051.0\ngini
= 0.444\nsamples = 12\nvalue = [8, 4]'),
 Text(0.46044975625098283, 0.5192307692307693, 'X[7] <= 7.5\ngini =
0.32\nsamples = 10\nvalue = [8, 2]'),
 Text(0.4591916968076742, 0.4807692307692308, 'gini = 0.0\nsamples =
8\nvalue = [8, 0]'),
 Text(0.46170781569429153, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.46296587513760024, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.46170781569429153, 0.5961538461538461, 'gini = 0.0\nsamples =
10\nvalue = [0, 10]'),
 Text(0.46296587513760024, 0.6346153846153846, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.46422393458090894, 0.6730769230769231, 'gini = 0.0\nsamples =
12\nvalue = [0, 12]'),
 Text(0.47491743984903284, 0.7884615384615384, 'X[0] <= 687.0\ngini =
0.469\nsamples = 88\nvalue = [55, 33]'),
 Text(0.4736593804057242, 0.75, 'gini = 0.0\nsamples = 6\nvalue = [0,
6]'),
 Text(0.47617549929234154, 0.75, 'X[1] <= 26.5\ngini = 0.442\nsamples
= 82\nvalue = [55, 27]'),
 Text(0.47177229124076114, 0.7115384615384616, 'X[1] <= 22.5\ngini =
```

```
0.444\nsamples = 9\nvalue = [3, 6]'),
 Text(0.47051423179745244, 0.6730769230769231, 'X[5] <= 20.0\ngini =
0.375\nsamples = 4\nvalue = [3, 1]'),
 Text(0.46925617235414374, 0.6346153846153846, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.47177229124076114, 0.6346153846153846, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.47303035068406984, 0.6730769230769231, 'gini = 0.0\nsamples =
5\nvalue = [0, 5]'),
 Text(0.480578707343922, 0.7115384615384616, 'X[0] <= 9743.0\ngini =
0.41\nsamples = 73\nvalue = [52, 21]'),
 Text(0.4793206479006133, 0.6730769230769231, 'X[7] <= 18.5\ngini =
0.392\nsamples = 71\nvalue = [52, 19]'),
 Text(0.47428841012737855, 0.6346153846153846, 'X[21] <= 0.5\ngini =
0.346\nsamples = 63\nvalue = [49, 14]'),
 Text(0.46925617235414374, 0.5961538461538461, 'X[1] <= 57.0\ngini =
0.239\nsamples = 36\nvalue = [31, 5]'),
 Text(0.46799811291083504, 0.5576923076923077, 'X[4] <= 0.5\ngini =
0.202\nsamples = 35\nvalue = [31, 4]'),
 Text(0.46548199402421764, 0.5192307692307693, 'X[9] <= 2.0\ngini =
0.5\nsamples = 4\nvalue = [2, 2]'),
 Text(0.46422393458090894, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.46674005346752634, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.47051423179745244, 0.5192307692307693, 'X[2] <= 3.5\ngini =
0.121\nsamples = 31\nvalue = [29, 2]'),
 Text(0.46925617235414374, 0.4807692307692308, 'gini = 0.0\nsamples =
23\nvalue = [23, 0]'),
 Text(0.47177229124076114, 0.4807692307692308, 'X[6] <= 2.5\ngini =
0.375\nsamples = 8\nvalue = [6, 2]'),
 Text(0.47051423179745244, 0.4423076923076923, 'gini = 0.0\nsamples =
6\nvalue = [6, 0]'),
 Text(0.47303035068406984, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.47051423179745244, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.4793206479006133, 0.5961538461538461, 'X[6] <= 1.5\ngini =
0.444\nsamples = 27\nvalue = [18, 9]'),
 Text(0.4768045290139959, 0.5576923076923077, 'X[44] <= 0.5\ngini =
0.463\nsamples = 11\nvalue = [4, 7]'),
 Text(0.4755464695706872, 0.5192307692307693, 'X[8] <= 8.0\ngini =
0.444\nsamples = 6\nvalue = [4, 2]'),
 Text(0.47428841012737855, 0.4807692307692308, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.4768045290139959, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.4780625884573046, 0.5192307692307693, 'gini = 0.0\nsamples =
5\nvalue = [0, 5]'),
 Text(0.4818367667872307, 0.5576923076923077, 'X[15] <= 0.5\ngini =
```

```
0.219\nsamples = 16\nvalue = [14, 2]'),
 Text(0.480578707343922, 0.5192307692307693, 'X[3] <= 3318.5\ngini =
0.124\nsamples = 15\nvalue = [14, 1]'),
 Text(0.4793206479006133, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.4818367667872307, 0.4807692307692308, 'gini = 0.0\nsamples =
14\nvalue = [14, 0]'),
 Text(0.4830948262305394, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.4843528856738481, 0.6346153846153846, 'X[1] <= 40.0\ngini =
0.469\nsamples = 8\nvalue = [3, 5]'),
 Text(0.4830948262305394, 0.5961538461538461, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.4856109451171568, 0.5961538461538461, 'X[10] <= 1.5\ngini =
0.278\nsamples = 6\nvalue = [1, 5]'),
 Text(0.4843528856738481, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.4868690045604655, 0.5576923076923077, 'gini = 0.0\nsamples =
5\nvalue = [0, 5]'),
 Text(0.4818367667872307, 0.6730769230769231, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.8195867009049673, 0.9423076923076923, 'X[1] <= 47.5\ngini =
0.5\nsamples = 1808\nvalue = [883, 925]'),
 Text(0.7413649315315891, 0.9038461538461539, 'X[3] <= 19913.5\ngini =
0.5\nsamples = 1277\nvalue = [650, 627]'),
 Text(0.6709670072682419, 0.8653846153846154, 'X[34] <= 0.5\ngini =
0.5\nsamples = 1269\nvalue = [649, 620]'),
 Text(0.5339453370714735, 0.8269230769230769, 'X[1] <= 21.5\ngini =
0.499\nsamples = 1023\nvalue = [539, 484]'),
 Text(0.4925302720553546, 0.7884615384615384, 'X[5] <= 38.5\ngini =
0.448\nsamples = 68\nvalue = [45, 23]'),
 Text(0.4912722126120459, 0.75, 'X[4] <= 2.5\ngini = 0.426\nsamples =
65\nvalue = [45, 20]'),
 Text(0.4868690045604655, 0.7115384615384616, 'X[29] <= 0.5\ngini =
0.124\nsamples = 15\nvalue = [14, 1]'),
 Text(0.4856109451171568, 0.6730769230769231, 'gini = 0.0\nsamples =
12\nvalue = [12, 0]'),
 Text(0.4881270640037742, 0.6730769230769231, 'X[18] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.4868690045604655, 0.6346153846153846, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.48938512344708285, 0.6346153846153846, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.49567542066362635, 0.7115384615384616, 'X[20] <= 0.5\ngini =
0.471\nsamples = 50\nvalue = [31, 19]'),
 Text(0.49315930177700895, 0.6730769230769231, 'X[0] <= 1158.0\ngini =
0.497\nsamples = 39\nvalue = [21, 18]'),
 Text(0.49190124233370025, 0.6346153846153846, 'gini = 0.0\nsamples =
6\nvalue = [6, 0]'),
```

```
 Text(0.49441736122031765, 0.6346153846153846, 'X[6] <= 1.5\ngini =
0.496\nsamples = 33\nvalue = [15, 18]'),
 Text(0.4912722126120459, 0.5961538461538461, 'X[30] <= 0.5\ngini =
0.278\nsamples = 12\nvalue = [2, 10]'),
 Text(0.4900141531687372, 0.5576923076923077, 'X[16] <= 0.5\ngini =
0.165\nsamples = 11\nvalue = [1, 10]'),
 Text(0.48875609372542855, 0.5192307692307693, 'gini = 0.0\nsamples =
10\nvalue = [0, 10]'),
 Text(0.4912722126120459, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.4925302720553546, 0.5576923076923077, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.4975625098285894, 0.5961538461538461, 'X[8] <= 4.5\ngini =
0.472\nsamples = 21\nvalue = [13, 8]'),
 Text(0.495046390941972, 0.5576923076923077, 'X[0] <= 9465.0\ngini =
0.18\nsamples = 10\nvalue = [9, 1]'),
 Text(0.4937883314986633, 0.5192307692307693, 'gini = 0.0\nsamples =
9\nvalue = [9, 0]'),
 Text(0.4963044503852807, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.5000786287152068, 0.5576923076923077, 'X[27] <= 0.5\ngini =
0.463\nsamples = 11\nvalue = [4, 7]'),
 Text(0.4988205692718981, 0.5192307692307693, 'gini = 0.0\nsamples =
7\nvalue = [0, 7]'),
 Text(0.5013366881585155, 0.5192307692307693, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.49819153955024376, 0.6730769230769231, 'X[33] <= 0.5\ngini =
0.165\nsamples = 11\nvalue = [10, 1]'),
 Text(0.49693348010693505, 0.6346153846153846, 'gini = 0.0\nsamples =
10\nvalue = [10, 0]'),
 Text(0.49944959899355246, 0.6346153846153846, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.4937883314986633, 0.75, 'gini = 0.0\nsamples = 3\nvalue = [0,
3]'),
 Text(0.5753604020875924, 0.7884615384615384, 'X[0] <= 244.5\ngini =
0.499\nsamples = 955\nvalue = [494, 461]'),
 Text(0.5032237773234786, 0.75, 'X[3] <= 11160.5\ngini = 0.426\
nsamples = 26\nvalue = [8, 18]'),
 Text(0.5019657178801699, 0.7115384615384616, 'gini = 0.0\nsamples =
13\nvalue = [0, 13]'),
 Text(0.5044818367667873, 0.7115384615384616, 'X[3] <= 16109.0\ngini =
0.473\nsamples = 13\nvalue = [8, 5]'),
 Text(0.5032237773234786, 0.6730769230769231, 'X[1] <= 45.0\ngini =
0.198\nsamples = 9\nvalue = [8, 1]'),
 Text(0.5019657178801699, 0.6346153846153846, 'gini = 0.0\nsamples =
8\nvalue = [8, 0]'),
 Text(0.5044818367667873, 0.6346153846153846, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.505739896210096, 0.6730769230769231, 'gini = 0.0\nsamples = 4\
```

```
nvalue = [0, 4]'),
 Text(0.6474970268517063, 0.75, 'X[38] <= 0.5\ngini = 0.499\nsamples =
929\nvalue = [486, 443]'),
 Text(0.567118208444724, 0.7115384615384616, 'X[6] <= 0.5\ngini = 0.5\
nsamples = 615\nvalue = [306, 309]'),
 Text(0.517141059915081, 0.6730769230769231, 'X[3] <= 4184.5\ngini =
0.475\nsamples = 129\nvalue = [79, 50]'),
 Text(0.5069979556534047, 0.6346153846153846, 'X[3] <= 1367.0\ngini =
0.153\nsamples = 24\nvalue = [22, 2]'),
 Text(0.505739896210096, 0.5961538461538461, 'gini = 0.0\nsamples = 1\
nvalue = [0, 1]'),
 Text(0.5082560150967134, 0.5961538461538461, 'X[0] <= 1806.0\ngini =
0.083\nsamples = 23\nvalue = [22, 1]'),
 Text(0.5069979556534047, 0.5576923076923077, 'X[27] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.505739896210096, 0.5192307692307693, 'gini = 0.0\nsamples = 1\
nvalue = [0, 1]'),
 Text(0.5082560150967134, 0.5192307692307693, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.5095140745400221, 0.5576923076923077, 'gini = 0.0\nsamples =
20\nvalue = [20, 0]'),
 Text(0.5272841641767574, 0.6346153846153846, 'X[5] <= 18.5\ngini =
0.496\nsamples = 105\nvalue = [57, 48]'),
 Text(0.516747916339047, 0.5961538461538461, 'X[5] <= 1.5\ngini =
0.484\nsamples = 51\nvalue = [21, 30]'),
 Text(0.5120301934266394, 0.5576923076923077, 'X[8] <= 1.5\ngini =
0.32\nsamples = 10\nvalue = [8, 2]'),
 Text(0.5107721339833308, 0.5192307692307693, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.5132882528699481, 0.5192307692307693, 'X[7] <= 17.0\ngini =
0.198\nsamples = 9\nvalue = [8, 1]'),
 Text(0.5120301934266394, 0.4807692307692308, 'gini = 0.0\nsamples =
8\nvalue = [8, 0]'),
 Text(0.5145463123132568, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.5214656392514546, 0.5576923076923077, 'X[1] <= 26.5\ngini =
0.433\nsamples = 41\nvalue = [13, 28]'),
 Text(0.5183204906431829, 0.5192307692307693, 'X[31] <= 0.5\ngini =
0.42\nsamples = 10\nvalue = [7, 3]'),
 Text(0.5170624311998742, 0.4807692307692308, 'X[28] <= 0.5\ngini =
0.219\nsamples = 8\nvalue = [7, 1]'),
 Text(0.5158043717565655, 0.4423076923076923, 'gini = 0.0\nsamples =
7\nvalue = [7, 0]'),
 Text(0.5183204906431829, 0.4423076923076923, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.5195785500864916, 0.4807692307692308, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.5246107878597264, 0.5192307692307693, 'X[1] <= 35.5\ngini =
0.312\nsamples = 31\nvalue = [6, 25]'),
```

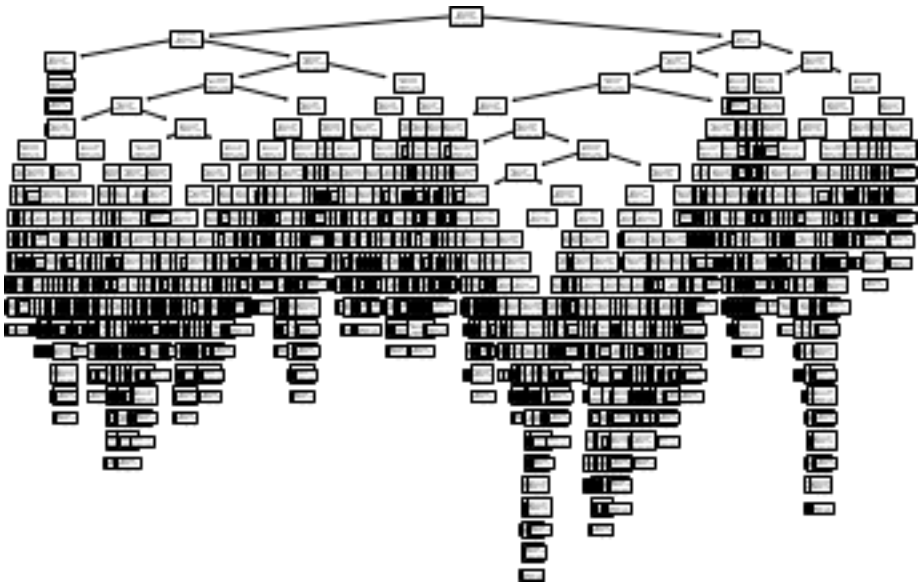
```
 Text(0.522094668973109, 0.4807692307692308, 'X[1] <= 34.5\ngini =
0.459\nsamples = 14\nvalue = [5, 9]'),
 Text(0.5208366095298003, 0.4423076923076923, 'X[7] <= 2.5\ngini =
0.375\nsamples = 12\nvalue = [3, 9]'),
 Text(0.5183204906431829, 0.40384615384615385, 'X[1] <= 28.0\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.5170624311998742, 0.36538461538461536, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.5195785500864916, 0.36538461538461536, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.5233527284164177, 0.40384615384615385, 'X[32] <= 0.5\ngini =
0.198\nsamples = 9\nvalue = [1, 8]'),
 Text(0.522094668973109, 0.36538461538461536, 'gini = 0.0\nsamples =
7\nvalue = [0, 7]'),
 Text(0.5246107878597264, 0.36538461538461536, 'X[19] <= 0.5\ngini =
0.5\nsamples = 2\nvalue = [1, 1]'),
 Text(0.5233527284164177, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.5258688473030351, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.5233527284164177, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.5271269067463438, 0.4807692307692308, 'X[31] <= 0.5\ngini =
0.111\nsamples = 17\nvalue = [1, 16]'),
 Text(0.5258688473030351, 0.4423076923076923, 'gini = 0.0\nsamples =
14\nvalue = [0, 14]'),
 Text(0.5283849661896525, 0.4423076923076923, 'X[5] <= 13.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.5271269067463438, 0.40384615384615385, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.5296430256329612, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.5378204120144677, 0.5961538461538461, 'X[7] <= 11.5\ngini =
0.444\nsamples = 54\nvalue = [36, 18]'),
 Text(0.534675263406196, 0.5576923076923077, 'X[1] <= 29.5\ngini =
0.298\nsamples = 33\nvalue = [27, 6]'),
 Text(0.5334172039628873, 0.5192307692307693, 'X[4] <= 7.5\ngini =
0.48\nsamples = 15\nvalue = [9, 6]'),
 Text(0.5321591445195786, 0.4807692307692308, 'X[0] <= 5924.0\ngini =
0.375\nsamples = 12\nvalue = [9, 3]'),
 Text(0.5309010850762699, 0.4423076923076923, 'gini = 0.0\nsamples =
8\nvalue = [8, 0]'),
 Text(0.5334172039628873, 0.4423076923076923, 'X[0] <= 8683.0\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.5321591445195786, 0.40384615384615385, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.534675263406196, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.534675263406196, 0.4807692307692308, 'gini = 0.0\nsamples = 3\
```

```
nvalue = [0, 3]'),
 Text(0.5359333228495047, 0.5192307692307693, 'gini = 0.0\nsamples =
18\nvalue = [18, 0]'),
 Text(0.5409655606227394, 0.5576923076923077, 'X[8] <= 1.5\ngini =
0.49\nsamples = 21\nvalue = [9, 12]'),
 Text(0.5384494417361221, 0.5192307692307693, 'X[32] <= 0.5\ngini =
0.245\nsamples = 7\nvalue = [6, 1]'),
 Text(0.5371913822928134, 0.4807692307692308, 'gini = 0.0\nsamples =
6\nvalue = [6, 0]'),
 Text(0.5397075011794308, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.5434816795093568, 0.5192307692307693, 'X[8] <= 6.5\ngini =
0.337\nsamples = 14\nvalue = [3, 11]'),
 Text(0.5422236200660481, 0.4807692307692308, 'gini = 0.0\nsamples =
8\nvalue = [0, 8]'),
 Text(0.5447397389526655, 0.4807692307692308, 'X[5] <= 30.5\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.5434816795093568, 0.4423076923076923, 'X[19] <= 0.5\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.5422236200660481, 0.40384615384615385, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.5447397389526655, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.5459977983959742, 0.4423076923076923, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.617095356974367, 0.6730769230769231, 'X[10] <= 4.5\ngini =
0.498\nsamples = 486\nvalue = [227, 259]'),
 Text(0.5909291948419563, 0.6346153846153846, 'X[7] <= 13.5\ngini =
0.498\nsamples = 257\nvalue = [136, 121]'),
 Text(0.554873014624941, 0.5961538461538461, 'X[0] <= 826.0\ngini =
0.498\nsamples = 168\nvalue = [79, 89]'),
 Text(0.5472558578392829, 0.5576923076923077, 'X[1] <= 40.5\ngini =
0.375\nsamples = 12\nvalue = [9, 3]'),
 Text(0.5459977983959742, 0.5192307692307693, 'gini = 0.0\nsamples =
8\nvalue = [8, 0]'),
 Text(0.5485139172825916, 0.5192307692307693, 'X[6] <= 1.5\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.5472558578392829, 0.4807692307692308, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.5497719767259003, 0.4807692307692308, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.5624901714105992, 0.5576923076923077, 'X[0] <= 3469.5\ngini =
0.495\nsamples = 156\nvalue = [70, 86]'),
 Text(0.5535461550558264, 0.5192307692307693, 'X[3] <= 2788.0\ngini =
0.411\nsamples = 38\nvalue = [11, 27]'),
 Text(0.5522880956125177, 0.4807692307692308, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.5548042144991351, 0.4807692307692308, 'X[2] <= 1.5\ngini =
0.353\nsamples = 35\nvalue = [8, 27]'),
```

```
 Text(0.5516590658908633, 0.4423076923076923, 'X[4] <= 6.5\ngini =
0.49\nsamples = 7\nvalue = [4, 3]'),
 Text(0.5504010064475546, 0.40384615384615385, 'gini = 0.0\nsamples =
4\nvalue = [4, 0]'),
 Text(0.552917125334172, 0.40384615384615385, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.5579493631074068, 0.4423076923076923, 'X[4] <= 6.5\ngini =
0.245\nsamples = 28\nvalue = [4, 24]'),
 Text(0.5554332442207894, 0.40384615384615385, 'X[8] <= 0.5\ngini =
0.087\nsamples = 22\nvalue = [1, 21]'),
 Text(0.5541751847774807, 0.36538461538461536, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.5566913036640981, 0.36538461538461536, 'gini = 0.0\nsamples =
21\nvalue = [0, 21]'),
 Text(0.5604654819940242, 0.40384615384615385, 'X[22] <= 0.5\ngini =
0.5\nsamples = 6\nvalue = [3, 3]'),
 Text(0.5592074225507155, 0.36538461538461536, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.5617235414373329, 0.36538461538461536, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.5714341877653719, 0.5192307692307693, 'X[4] <= 0.5\ngini =
0.5\nsamples = 118\nvalue = [59, 59]'),
 Text(0.5629816008806416, 0.4807692307692308, 'X[7] <= 10.0\ngini =
0.26\nsamples = 13\nvalue = [11, 2]'),
 Text(0.5617235414373329, 0.4423076923076923, 'gini = 0.0\nsamples =
10\nvalue = [10, 0]'),
 Text(0.5642396603239503, 0.4423076923076923, 'X[44] <= 0.5\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.5629816008806416, 0.40384615384615385, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.565497719767259, 0.40384615384615385, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.5798867746501022, 0.4807692307692308, 'X[3] <= 5994.0\ngini =
0.496\nsamples = 105\nvalue = [48, 57]'),
 Text(0.5705299575404937, 0.4423076923076923, 'X[27] <= 0.5\ngini =
0.355\nsamples = 26\nvalue = [6, 20]'),
 Text(0.5680138386538764, 0.40384615384615385, 'X[0] <= 3952.5\ngini =
0.188\nsamples = 19\nvalue = [2, 17]'),
 Text(0.5667557792105677, 0.36538461538461536, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.5692718980971851, 0.36538461538461536, 'X[13] <= 0.5\ngini =
0.105\nsamples = 18\nvalue = [1, 17]'),
 Text(0.5680138386538764, 0.3269230769230769, 'gini = 0.0\nsamples =
14\nvalue = [0, 14]'),
 Text(0.5705299575404937, 0.3269230769230769, 'X[7] <= 4.0\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.5692718980971851, 0.28846153846153844, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.5717880169838024, 0.28846153846153844, 'gini = 0.0\nsamples =
```

```
1\nvalue = [1, 0]'),
 Text(0.5730460764271111, 0.40384615384615385, 'X[8] <= 5.5\ngini =
0.49\nsamples = 7\nvalue = [4, 3]'),
 Text(0.5717880169838024, 0.36538461538461536, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.5743041358704198, 0.36538461538461536, 'X[17] <= 0.5\ngini =
0.375\nsamples = 4\nvalue = [1, 3]'),
 Text(0.5730460764271111, 0.3269230769230769, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.5755621953137285, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.5892435917597106, 0.4423076923076923, 'X[11] <= 0.5\ngini =
0.498\nsamples = 79\nvalue = [42, 37]'),
 Text(0.5827960371127536, 0.40384615384615385, 'X[1] <= 26.5\ngini =
0.482\nsamples = 64\nvalue = [38, 26]'),
 Text(0.5793363736436546, 0.36538461538461536, 'X[0] <= 5693.0\ngini =
0.408\nsamples = 7\nvalue = [2, 5]'),
 Text(0.5780783142003459, 0.3269230769230769, 'gini = 0.0\nsamples =
5\nvalue = [0, 5]'),
 Text(0.5805944330869633, 0.3269230769230769, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.5862557005818525, 0.36538461538461536, 'X[5] <= 3.5\ngini =
0.465\nsamples = 57\nvalue = [36, 21]'),
 Text(0.5831105519735807, 0.3269230769230769, 'X[31] <= 0.5\ngini =
0.32\nsamples = 5\nvalue = [1, 4]'),
 Text(0.581852492530272, 0.28846153846153844, 'gini = 0.0\nsamples =
4\nvalue = [0, 4]'),
 Text(0.5843686114168894, 0.28846153846153844, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.5894008491901243, 0.3269230769230769, 'X[4] <= 7.5\ngini =
0.44\nsamples = 52\nvalue = [35, 17]'),
 Text(0.5868847303035069, 0.28846153846153844, 'X[26] <= 0.5\ngini =
0.397\nsamples = 44\nvalue = [32, 12]'),
 Text(0.5856266708601982, 0.25, 'X[1] <= 45.5\ngini = 0.463\nsamples =
33\nvalue = [21, 12]'),
 Text(0.5843686114168894, 0.21153846153846154, 'X[44] <= 0.5\ngini =
0.42\nsamples = 30\nvalue = [21, 9]'),
 Text(0.581852492530272, 0.17307692307692307, 'X[4] <= 5.5\ngini =
0.492\nsamples = 16\nvalue = [9, 7]'),
 Text(0.5805944330869633, 0.1346153846153846, 'X[5] <= 24.0\ngini =
0.426\nsamples = 13\nvalue = [9, 4]'),
 Text(0.5793363736436546, 0.09615384615384616, 'gini = 0.0\nsamples =
7\nvalue = [7, 0]'),
 Text(0.581852492530272, 0.09615384615384616, 'X[10] <= 0.5\ngini =
0.444\nsamples = 6\nvalue = [2, 4]'),
 Text(0.5805944330869633, 0.057692307692307696, 'X[4] <= 2.0\ngini =
0.444\nsamples = 3\nvalue = [2, 1]'),
 Text(0.5793363736436546, 0.019230769230769232, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
```

```
 Text(0.581852492530272, 0.019230769230769232, 'gini = 0.0\nsamples =
2\nvalue = [2, 0]'),
 Text(0.5831105519735807, 0.057692307692307696, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.5831105519735807, 0.1346153846153846, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.5868847303035069, 0.17307692307692307, 'X[9] <= 2.5\ngini =
0.245\nsamples = 14\nvalue = [12, 2]'),
 Text(0.5856266708601982, 0.1346153846153846, 'gini = 0.0\nsamples =
11\nvalue = [11, 0]'),
 Text(0.5881427897468156, 0.1346153846153846, 'X[5] <= 8.0\ngini =
0.444\nsamples = 3\nvalue = [1, 2]'),
 Text(0.5868847303035069, 0.09615384615384616, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.5894008491901243, 0.09615384615384616, 'gini = 0.0\nsamples =
2\nvalue = [0, 2]'),
 Text(0.5868847303035069, 0.21153846153846154, 'gini = 0.0\nsamples =
3\nvalue = [0, 3]'),
 Text(0.5881427897468156, 0.25, 'gini = 0.0\nsamples = 11\nvalue =
[11, 0]'),
 Text(0.5919169680767417, 0.28846153846153844, 'X[10] <= 2.5\ngini =
0.469\nsamples = 8\nvalue = [3, 5]'),
 Text(0.590658908633433, 0.25, 'X[7] <= 12.0\ngini = 0.375\nsamples =
4\nvalue = [3, 1]'),
 Text(0.5894008491901243, 0.21153846153846154, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.5919169680767417, 0.21153846153846154, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.5931750275200504, 0.25, 'gini = 0.0\nsamples = 4\nvalue = [0,
4]'),
 Text(0.5956911464066678, 0.40384615384615385, 'X[28] <= 0.5\ngini =
0.391\nsamples = 15\nvalue = [4, 11]'),
 Text(0.5931750275200504, 0.36538461538461536, 'X[1] <= 46.5\ngini =
0.165\nsamples = 11\nvalue = [1, 10]'),
 Text(0.5919169680767417, 0.3269230769230769, 'gini = 0.0\nsamples =
10\nvalue = [0, 10]'),
 Text(0.5944330869633591, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [1, 0]'),
 Text(0.5982072652932852, 0.36538461538461536, 'X[10] <= 1.5\ngini =
0.375\nsamples = 4\nvalue = [3, 1]'),
 Text(0.5969492058499765, 0.3269230769230769, 'gini = 0.0\nsamples =
1\nvalue = [0, 1]'),
 Text(0.5994653247365938, 0.3269230769230769, 'gini = 0.0\nsamples =
3\nvalue = [3, 0]'),
 Text(0.6269853750589716, 0.5961538461538461, 'X[5] <= 33.0\ngini =
0.461\nsamples = 89\nvalue = [57, 32]'),
 ...]
```

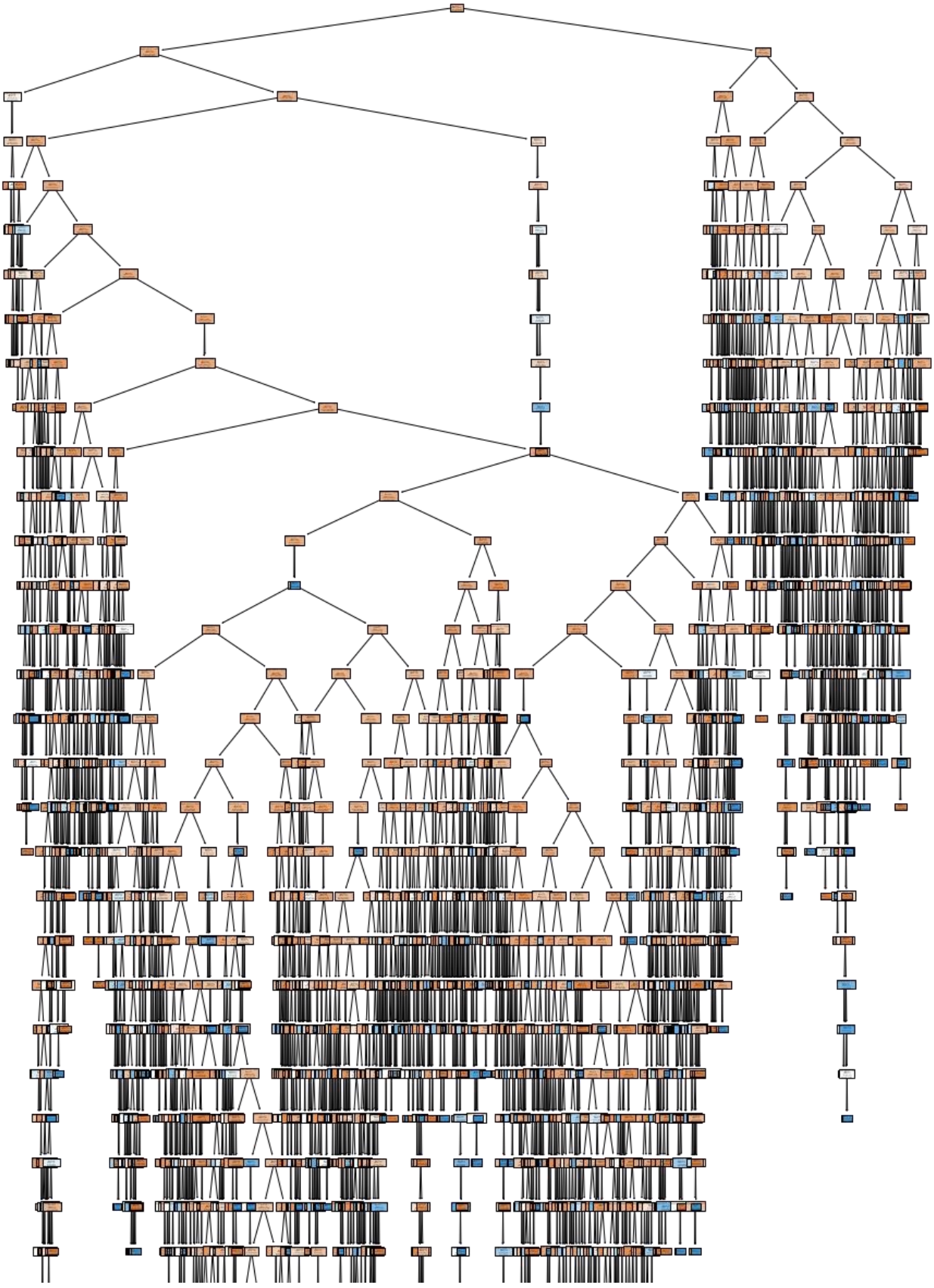


```
!pip install pydotplus
```

```
Collecting pydotplus
  Downloading pydotplus-2.0.2.tar.gz (278 kB)
Requirement already satisfied: pyparsing>=2.0.1 in c:\users\nayakp\
anaconda3\lib\site-packages (from pydotplus) (3.0.4)
Building wheels for collected packages: pydotplus
  Building wheel for pydotplus (setup.py): started
  Building wheel for pydotplus (setup.py): finished with status 'done'
  Created wheel for pydotplus: filename=pydotplus-2.0.2-py3-none-
any.whl size=24575
sha256=f7cd5fd823bcc626490a0f6959bae95d9f446f1e05deb4f55a66a8b950dccab
5
  Stored in directory: c:\users\nayakp\appdata\local\pip\cache\wheels\
89\e5\de\6966007cf223872eedfbebbe0e074534e72e9128c8fd4b55eb
Successfully built pydotplus
Installing collected packages: pydotplus
Successfully installed pydotplus-2.0.2
```

```
from sklearn.tree import DecisionTreeClassifier, plot_tree
import matplotlib.pyplot as plt

# Assuming 'X' contains your features and 'y' contains the target
variable
# Selecting the top features
#top_features = ['MonthlyIncome in $', 'TotalWorkingYears', 'Age',
'YearsAtCompany',
                #'YearsWithCurrManager', 'YearsInCurrentRole',
'NumCompaniesWorked',
                ##YearsSinceLastPromotion', 'TrainingTimesLastYear']
X_top_features = X[['MonthlyIncome in $', 'TotalWorkingYears', 'Age',
```

```
'YearsAtCompany',
                     'YearsWithCurrManager', 'YearsInCurrentRole',
'NumCompaniesWorked',
                     'YearsSinceLastPromotion',
'TrainingTimesLastYear']]
y = df['Attrition']

# Initialize and fit the decision tree classifier
clf = DecisionTreeClassifier(random_state=42)
clf.fit(X_top_features, y)

# Plot the decision tree
plt.figure(figsize=(15, 30))
plot_tree(clf, feature_names=X_top_features.columns, class_names=['No
Attrition', 'Attrition'], filled=True)
plt.show()
```



```
#Predicting on test data
preds = model.predict(X_test) # predicting on test data set
pd.Series(preds).value_counts() # getting the count of each category
```

```
No      2204
Yes      796
dtype: int64
```

```
pd.crosstab(y_test,preds) # getting the 2 way table to understand the
correct and wrong predictions
```

```
col_0         No  Yes
Attrition
No          1810  408
Yes          394  388
```

```
# Accuracy
np.mean(preds==y_test)
```

```
0.7326666666666667
```

```
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, preds)

# plt.figure(figsize=(8, 6))
# sns.heatmap(cm, annot=True, fmt="d", xticklabels=k_means.classes_,
yticklabels=k_means.classes_)
# plt.xlabel("Predicted")
# plt.ylabel("Actual")
# plt.title(f"Confusion Matrix\nAccuracy: {accuracy:.2f}")
# plt.show()

precision = precision_score(y_test, preds, average='weighted')
f1 = f1_score(y_test, y_preds, average='weighted')
recall = recall_score(y_test, preds, average='weighted')
print(f'Precision: {precision:.2f}')
print(f'F1 Score: {f1:.2f}')
print(recall)
```

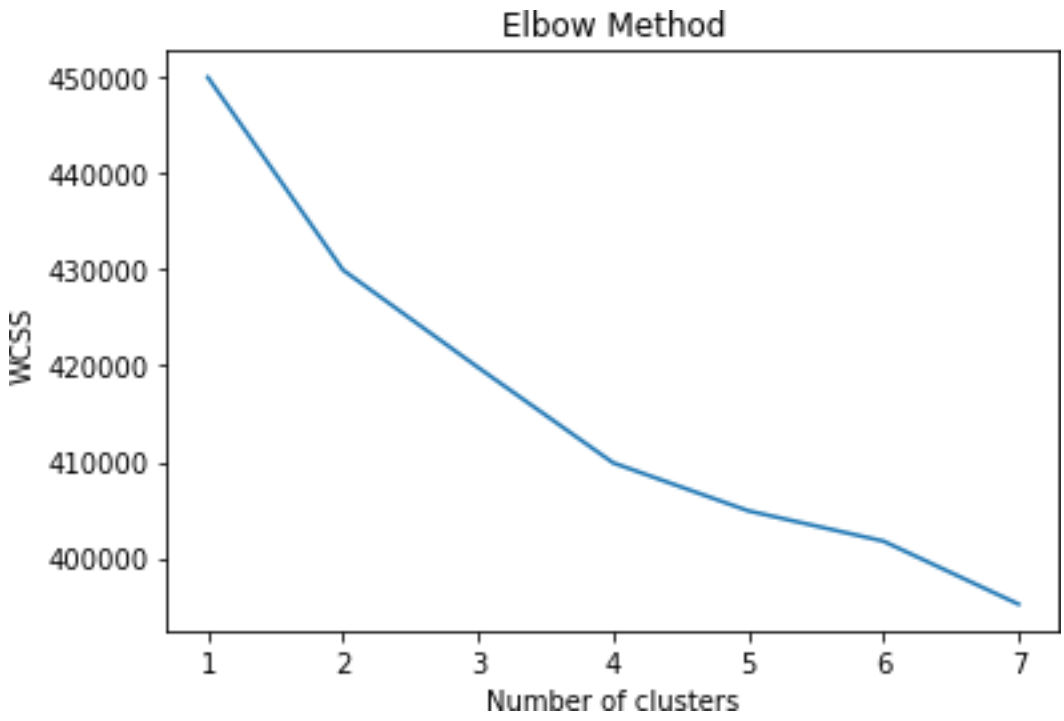
Justification on using Decision Tree Decision trees often provide good performance in terms of classification accuracy, especially when the dataset contains clear decision boundaries and interactions between features. Decision trees are relatively scalable and can handle large datasets efficiently, making them suitable for analyzing employee attrition in organizations of varying sizes

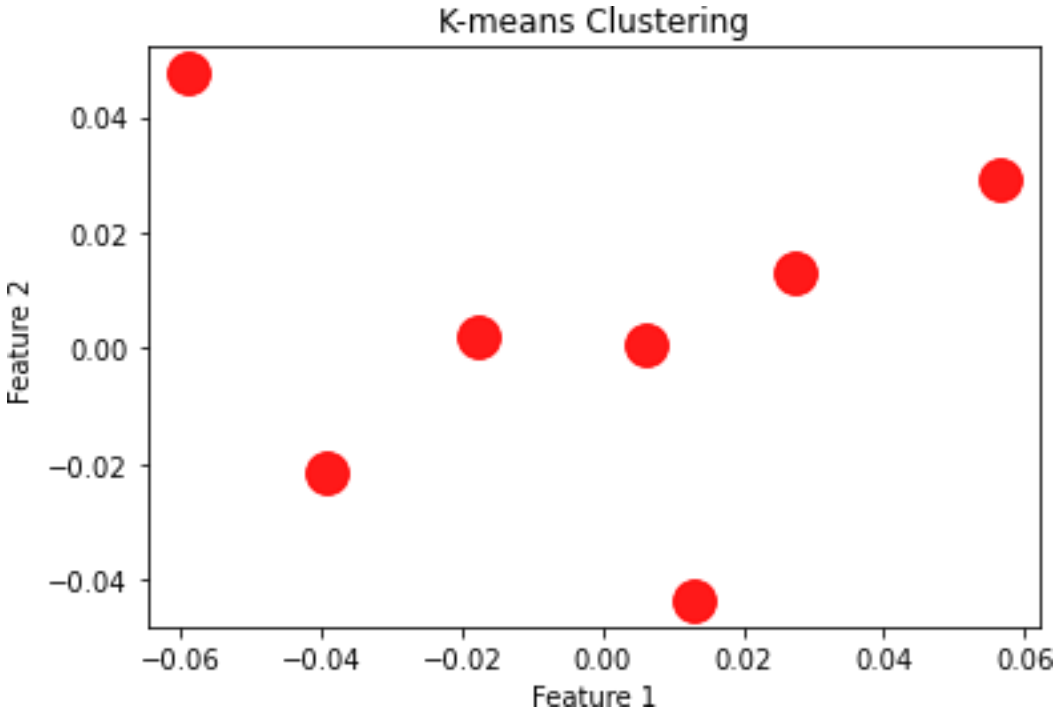
Precision: 0.74 F1 Score: 0.74 0.7346666666666668

6.2 ML technique 2 + Justification

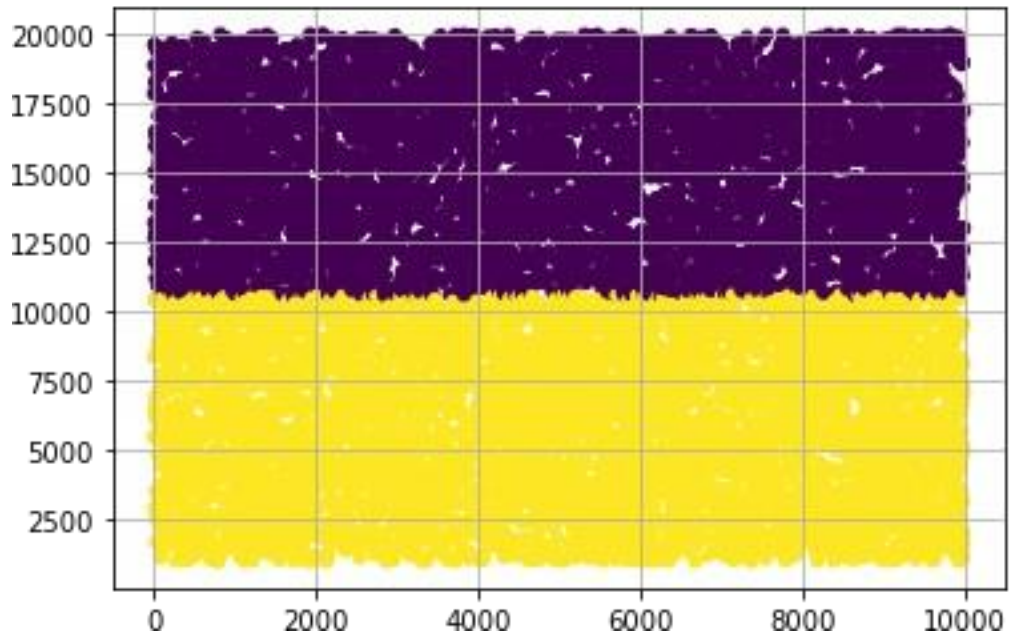
```
##---------Type the code below this line---------- ##
```

```
# Clustering

# Normalization function
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
scaler = StandardScaler()
scaled_df = scaler.fit_transform(df_encoded)

wcss = []
for i in range(1, 8):
    kmeans = KMeans(n_clusters=i,random_state=0)
    kmeans.fit(scaled_df)
    wcss.append(kmeans.inertia_)

plt.plot(range(1, 8), wcss)
plt.title('Elbow Method')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')
plt.show()

# Plotting the centroids of the clusters
centers = kmeans.cluster_centers_
plt.scatter(centers[:, 0], centers[:, 1], c='red', s=250, alpha=0.9)
plt.title('K-means Clustering')
plt.xlabel('Feature 1')
plt.ylabel('Feature 2')
plt.show()

# Visualize the clusters
fig = plt.figure(0)
plt.grid(True)
plt.scatter(X.iloc[:, 0],X.iloc[:, 1], c=y_kmeans, s=20,
cmap='viridis')
```






```
<matplotlib.collections.PathCollection at 0x29e2d297a60>
```




```
#Build Cluster algorithm
from sklearn.cluster import KMeans
import sklearn.metrics as metrics
clusters_new = KMeans(4, random_state=42) # No. of Clusters(4) chosen
based on the elbow curve
clusters_new.fit(scaled_df)
```

```
KMeans(n_clusters=4, random_state=42)
```

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

X_train, X_test,y_train,y_test =
train_test_split(X,y,test_size=0.20,random_state=42)
k_means = KMeans(4, random_state=42)
k_means.fit(X_train)
print(k_means.labels_)
print(y_test)
```

```
[2 2 3 ... 2 3 1]
6252    Yes
4684     No
1731     No
4742    Yes
4521     No
       ...
6412     No
8285     No
7853    Yes
1095     No
```

```
6929     No
Name: Attrition, Length: 2000, dtype: object
```

```
#bool_list = list(map(bool,X_test))

y_pred = k_means.predict(X_test)
bool_list = list(map(bool,y_pred))
y_test = y_test.map({"Yes":True,"No":False})
score = metrics.accuracy_score(y_test,bool_list)
print('Accuracy:{0:f}'.format(score))
```

```
Accuracy:0.339000
```

y_pred = model_rf.predict(X_test)

cm = confusion_matrix(y_test, y_pred)

plt.figure(figsize=(8, 6)) sns.heatmap(cm, annot=True, fmt="d", xticklabels=model_rf.classes_, yticklabels=model_rf.classes_) plt.xlabel("Predicted") plt.ylabel("Actual") plt.title(f"Confusion Matrix\nAccuracy: {accuracy:.2f}") plt.show()

precision = precision_score(y_test, y_pred, average='weighted') f1 = f1_score(y_test, y_pred, average='weighted') recall = recall_score(y_test, y_pred, average='weighted') print(f'Precision: {precision:.2f}') print(f'F1 Score: {f1:.2f}') print(recall)

```
#Assign clusters to the data set
df_encoded['clusterid_new'] = clusters_new.labels_
df['clusterid'] = clusters_new.labels_

df_encoded.groupby('clusterid_new').agg(['mean']).reset_index()
```

```
  clusterid_new    EmployeeID         Age  JobLevel MonthlyIncome in
$  \
                         mean        mean      mean                 mean

0              0   4975.990303   39.375717  2.491717        10758.736048

1              1   4947.785153   39.324607  2.470542        10482.710792

2              2   5018.536469   39.458529  2.485452        10247.773295

3              3   5061.074899   39.645749  2.515789        10471.206170


  NumCompaniesWorked TotalWorkingYears TrainingTimesLastYear
YearsAtCompany  \
                mean               mean                    mean
mean
0           4.645874          19.137338                1.995563
```

```
9.629899
1           4.562058          19.372528                2.029479
9.550496
2           4.469111          19.418277                2.038682
9.628139
3           4.474899          19.898934                2.006478
9.439499

  YearsInCurrentRole  ... Attrition_Yes PerformanceRating_Excellent  \
                mean  ...          mean                         mean
0           4.438384  ...      0.263030                     0.336566
1           4.431053  ...      0.252946                     0.330322
2           4.463133  ...      0.245118                     0.328816
3           4.507474  ...      0.263968                     0.340891

  PerformanceRating_Good PerformanceRating_Low JobSatisfaction_High  \
                    mean                 mean                  mean
0               0.329293             0.334141              0.254141
1               0.340141             0.329537              0.255302
2               0.336788             0.334396              0.226784
3               0.326316             0.332794              0.240081

  JobSatisfaction_Low JobSatisfaction_Medium JobSatisfaction_Very High
\
                 mean                    mean                       mean

0            0.249697                0.254141                   0.242020

1            0.246661                0.250982                   0.247054

2            0.249502                0.265046                   0.258669

3            0.234413                0.272065                   0.253441


  OverTime_No OverTime_Yes
         mean          mean
0         0.0           1.0
1         0.0           1.0
2         1.0           0.0
3         1.0           0.0

[4 rows x 49 columns]

df

      EmployeeID EmployeeLocation   Age   Department  Gender
MaritalStatus  \
0              1        Bangalore  25.0        Sales  Female
Single
1              2     Andrapradesh  45.0    Marketing    Male
```

```
Single
2               3     Andrapradesh  37.0           HR    Male
Married
3               4     Andrapradesh  35.0  Engineering  Female
Divorced
4               5           Orissa  51.0           HR  Female
Divorced
...           ...              ...   ...          ...     ...
...
9995         9996           Orissa  45.0  Engineering    Male
Single
9996         9997           Mumbai  42.0      Finance    Male
Divorced
9997         9998           Mumbai  42.0        Sales  Female
Single
9998         9999        Tamilnadu  35.0        Sales  Female
Divorced
9999        10000     Andrapradesh  53.0    Marketing  Female
Single

        Education          JobRole  JobLevel  MonthlyIncome in $  ...
\
0          Master          Manager         1              5505.0  ...

1     High School  Sales Executive         4              8377.0  ...

2          Master  Sales Executive         3             17854.0  ...

3        Bachelor          Manager         1             18881.0  ...

4        Bachelor         Engineer         1             15019.0  ...

...           ...              ...       ...                 ...  ...

9995     Bachelor          Manager         4              6992.0  ...

9996  High School         Engineer         3             17241.0  ...

9997     Bachelor          Manager         1              5015.0  ...

9998       Master          Manager         1              8168.0  ...

9999     Bachelor         Engineer         2              7144.0  ...


      YearsAtCompany  YearsInCurrentRole  YearsSinceLastPromotion  \
0                9.0                 5.0                      1.0
1               19.0                 1.0                      1.0
2               13.0                 1.0                      1.0
3                9.0                 0.0                      3.0
```

```
4                11.0                  2.0                        3.0
...               ...                  ...                        ...
9995             16.0                  1.0                        4.0
9996              5.0                  6.0                        1.0
9997             19.0                  1.0                        3.0
9998             14.0                  3.0                        4.0
9999              4.0                  2.0                        3.0

      YearsWithCurrManager  Attrition  PerformanceRating
JobSatisfaction  \
0                      2.0         No               Good
Medium
1                      5.0        Yes          Excellent
Low
2                      8.0        Yes               Good
Medium
3                      8.0        Yes                Low
Low
4                      9.0         No               Good         Very
High
...                    ...        ...                ...
...
9995                   0.0         No          Excellent         Very
High
9996                   0.0         No          Excellent         Very
High
9997                   8.0        Yes                Low
Medium
9998                   4.0         No               Good         Very
High
9999                   9.0        Yes                Low
Medium

     OverTime  clusterid  Cluster
0         Yes          0        1
1          No          2        1
2         Yes          1        0
3          No          3        0
4          No          3        0
...       ...        ...      ...
9995       No          2        1
9996       No          2        0
9997       No          3        1
9998       No          3        1
9999      Yes          0        1

[10000 rows x 23 columns]
```

```
y_pred = k_means.predict(X_test)
from sklearn.metrics import confusion_matrix
```

```
cm = confusion_matrix(y_test, y_pred)

# plt.figure(figsize=(8, 6))
# sns.heatmap(cm, annot=True, fmt="d", xticklabels=k_means.classes_,
yticklabels=k_means.classes_)
# plt.xlabel("Predicted")
# plt.ylabel("Actual")
# plt.title(f"Confusion Matrix\nAccuracy: {accuracy:.2f}")
# plt.show()

precision = precision_score(y_test, y_pred, average='weighted')
f1 = f1_score(y_test, y_pred, average='weighted')
recall = recall_score(y_test, y_pred, average='weighted')
print(f'Precision: {precision:.2f}')
print(f'F1 Score: {f1:.2f}')
print(recall)
```

Precision: 0.62 F1 Score: 0.36 0.2595 c:\Users\nayakp\Anaconda3\lib\site-packages\sklearn\metrics_classification.py:1318: UndefinedMetricWarning: Recall is ill-defined and being set to 0.0 in labels with no true samples. Use `zero_division` parameter to control this behavior. _warn_prf(average, modifier, msg_start, len(result))

Justification on using K-means clustering K-means clustering can help identify distinct groups or segments of employees based on their characteristics and work behavior. This can be valuable for understanding the different types of employees within the organization and tailoring strategies to address their needs

7. Conclusion

Compare the performance of the ML techniques used.

Derive values for performance study metrics like accuracy, precision, recall, F1 Score, AUC-ROC etc to compare the ML algos and plot them. A proper comparison based on different metrics should be done and not just accuracy alone, only then the comparison becomes authentic. You may use Confusion matrix, classification report, Word cloud etc as per the requirement of your application/problem.

Comparing both the models that we have used Classification model seems to be the best

8. Solution

Data Collection and Preprocessing: Gather data on employee attributes such as age, department, gender, job role, education, job level, etc., along with historical attrition data.

Preprocess the data by handling missing values, encoding categorical variables, and scaling numerical features if necessary.

Exploratory Data Analysis (EDA): Perform EDA to gain insights into the dataset. Analyze the distribution of features, correlations between variables, and identify potential patterns or trends related to attrition.

Feature Selection: Select relevant features that are likely to have a significant impact on attrition. This can be done using techniques such as feature importance analysis, correlation analysis, or domain knowledge.

Model Building: Build predictive models to forecast employee attrition. Experiment with different machine learning algorithms such as decision trees, random forests, logistic regression, etc. Train the models on historical data and evaluate their performance using appropriate metrics such as accuracy, precision, recall, F1 score, and AUC-ROC.

Model Evaluation and Validation: Validate the performance of the trained models using unseen data (e.g., a holdout test set or cross-validation). Ensure that the models generalize well to new data and are robust enough to handle different scenarios.

Interpretation and Insights: Interpret the results of the predictive models to understand the key factors driving attrition within the organization. Identify actionable insights and recommendations for HR policies, strategies, and interventions to mitigate attrition rates.

Deployment and Monitoring: Deploy the trained models into production for real-time prediction of employee attrition. Continuously monitor and update the models as new data becomes available and evaluate their performance over time.

During the process of solving the business problem, several challenges, observations, and decisions may arise. Some of these include:

Data Quality Issues: Dealing with missing values, outliers, and inconsistencies in the dataset. Feature Engineering: Selecting relevant features and transforming them appropriately to improve model performance. Model Selection: Choosing the most suitable machine learning algorithms and hyperparameters for the task at hand. Interpretability: Ensuring that the predictive models are interpretable and provide actionable insights to stakeholders. Ethical Considerations: Addressing potential biases in the data or models that may impact decision-making processes. Communication: Effectively communicating the findings and recommendations to stakeholders, including HR personnel, management, and other relevant parties.

Overall, solving the business problem of employee attrition requires a combination of data analysis, machine learning techniques, domain knowledge, and effective communication to drive positive outcomes for the organization.