

# y-kiran-21bai1718-week4-assignment

September 29, 2023

Perform Data preprocessing & Model Building on Employee Attrition Dataset 1.Data Collection. Please download the dataset from <https://www.kaggle.com/datasets/patelprashant/employee-attrition>

2.Data Preprocessing o Import the Libraries. o Importing the dataset. o Checking for Null Values. o Data Visualization. o Outlier Detection o Splitting Dependent and Independent variables o Perform Encoding o Feature Scaling. o Splitting Data into Train and Test

3.Model Building o Import the model building Libraries o Initializing the model o Training and testing the model o Evaluation of Model & Performance metrics o Save the Model

## 1 Data Collection

Collected Data from Kaggle - Employee Attrition Dataset

## 2 Data Preprocessing

### 2.1 Import the Libraries

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

### 2.2 Importing the DataSet

```
[2]: df=pd.read_csv(r"C:\Users\nitin\Desktop\assignkeents - \
↳submissions\Datasets\WA_Fn-UseC_-HR-Employee-Attrition.csv")
df.head()
```

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

DistanceFromHome Education EducationField EmployeeCount EmployeeNumber

0	1	2	Life Sciences	1	1 \
1	8	1	Life Sciences	1	2
2	2	2	Other	1	4
3	3	4	Life Sciences	1	5
4	2	1	Medical	1	7

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

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

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

[5 rows x 35 columns]

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1470 entries, 0 to 1469
Data columns (total 35 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Age                                   1470 non-null   int64
1   Attrition                           1470 non-null   object
2   BusinessTravel                      1470 non-null   object
3   DailyRate                           1470 non-null   int64
4   Department                          1470 non-null   object
5   DistanceFromHome                   1470 non-null   int64
6   Education                           1470 non-null   int64
7   EducationField                      1470 non-null   object
8   EmployeeCount                      1470 non-null   int64
9   EmployeeNumber                     1470 non-null   int64
10  EnvironmentSatisfaction             1470 non-null   int64
```

```

11 Gender                1470 non-null object
12 HourlyRate            1470 non-null int64
13 JobInvolvement        1470 non-null int64
14 JobLevel              1470 non-null int64
15 JobRole               1470 non-null object
16 JobSatisfaction       1470 non-null int64
17 MaritalStatus         1470 non-null object
18 MonthlyIncome         1470 non-null int64
19 MonthlyRate           1470 non-null int64
20 NumCompaniesWorked    1470 non-null int64
21 Over18                1470 non-null object
22 OverTime              1470 non-null object
23 PercentSalaryHike     1470 non-null int64
24 PerformanceRating     1470 non-null int64
25 RelationshipSatisfaction 1470 non-null int64
26 StandardHours        1470 non-null int64
27 StockOptionLevel      1470 non-null int64
28 TotalWorkingYears     1470 non-null int64
29 TrainingTimesLastYear 1470 non-null int64
30 WorkLifeBalance       1470 non-null int64
31 YearsAtCompany        1470 non-null int64
32 YearsInCurrentRole    1470 non-null int64
33 YearsSinceLastPromotion 1470 non-null int64
34 YearsWithCurrManager  1470 non-null int64
dtypes: int64(26), object(9)
memory usage: 402.1+ KB

```

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

```

[4]:
count    1470.000000  1470.000000  1470.000000  1470.000000  1470.000000  1470.0 \
mean      36.923810   802.485714     9.192517     2.912925         1.0
std        9.135373   403.509100     8.106864     1.024165         0.0
min       18.000000   102.000000     1.000000     1.000000         1.0
25%       30.000000   465.000000     2.000000     2.000000         1.0
50%       36.000000   802.000000     7.000000     3.000000         1.0
75%       43.000000  1157.000000    14.000000     4.000000         1.0
max       60.000000  1499.000000    29.000000     5.000000         1.0

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

```

max	2068.000000	4.000000	100.000000	4.000000
-----	-------------	----------	------------	----------

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

	StockOptionLevel	TotalWorkingYears	TrainingTimesLastYear
count	1470.000000	1470.000000	1470.000000 \
mean	0.793878	11.279592	2.799320
std	0.852077	7.780782	1.289271
min	0.000000	0.000000	0.000000
25%	0.000000	6.000000	2.000000
50%	1.000000	10.000000	3.000000
75%	1.000000	15.000000	3.000000
max	3.000000	40.000000	6.000000

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

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

[8 rows x 26 columns]

```
[5]: df.shape
```

```
[5]: (1470, 35)
```

## 2.3 Checking for null values

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

```
[6]: Age                False
     Attrition          False
     BusinessTravel     False
     DailyRate          False
     Department         False
     DistanceFromHome   False
     Education          False
     EducationField      False
     EmployeeCount      False
     EmployeeNumber     False
     EnvironmentSatisfaction False
     Gender             False
     HourlyRate         False
     JobInvolvement     False
     JobLevel           False
     JobRole            False
     JobSatisfaction    False
     MaritalStatus      False
     MonthlyIncome      False
     MonthlyRate        False
     NumCompaniesWorked False
     Over18             False
     OverTime           False
     PercentSalaryHike  False
     PerformanceRating  False
     RelationshipSatisfaction False
     StandardHours      False
     StockOptionLevel   False
     TotalWorkingYears  False
     TrainingTimesLastYear False
     WorkLifeBalance    False
     YearsAtCompany     False
     YearsInCurrentRole False
     YearsSinceLastPromotion False
     YearsWithCurrManager False
     dtype: bool
```

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

```
[7]: Age                0
     Attrition          0
     BusinessTravel     0
     DailyRate          0
```

Department	0
DistanceFromHome	0
Education	0
EducationField	0
EmployeeCount	0
EmployeeNumber	0
EnvironmentSatisfaction	0
Gender	0
HourlyRate	0
JobInvolvement	0
JobLevel	0
JobRole	0
JobSatisfaction	0
MaritalStatus	0
MonthlyIncome	0
MonthlyRate	0
NumCompaniesWorked	0
Over18	0
OverTime	0
PercentSalaryHike	0
PerformanceRating	0
RelationshipSatisfaction	0
StandardHours	0
StockOptionLevel	0
TotalWorkingYears	0
TrainingTimesLastYear	0
WorkLifeBalance	0
YearsAtCompany	0
YearsInCurrentRole	0
YearsSinceLastPromotion	0
YearsWithCurrManager	0

dtype: int64

```
[8]: print("Null percentage in columns : ")
for i in df.columns:
    c=df[i].count()
    n=df[i].isnull().sum()
    print(i," : ",(n/(n+c)) * 100)
```

Null percentage in columns :

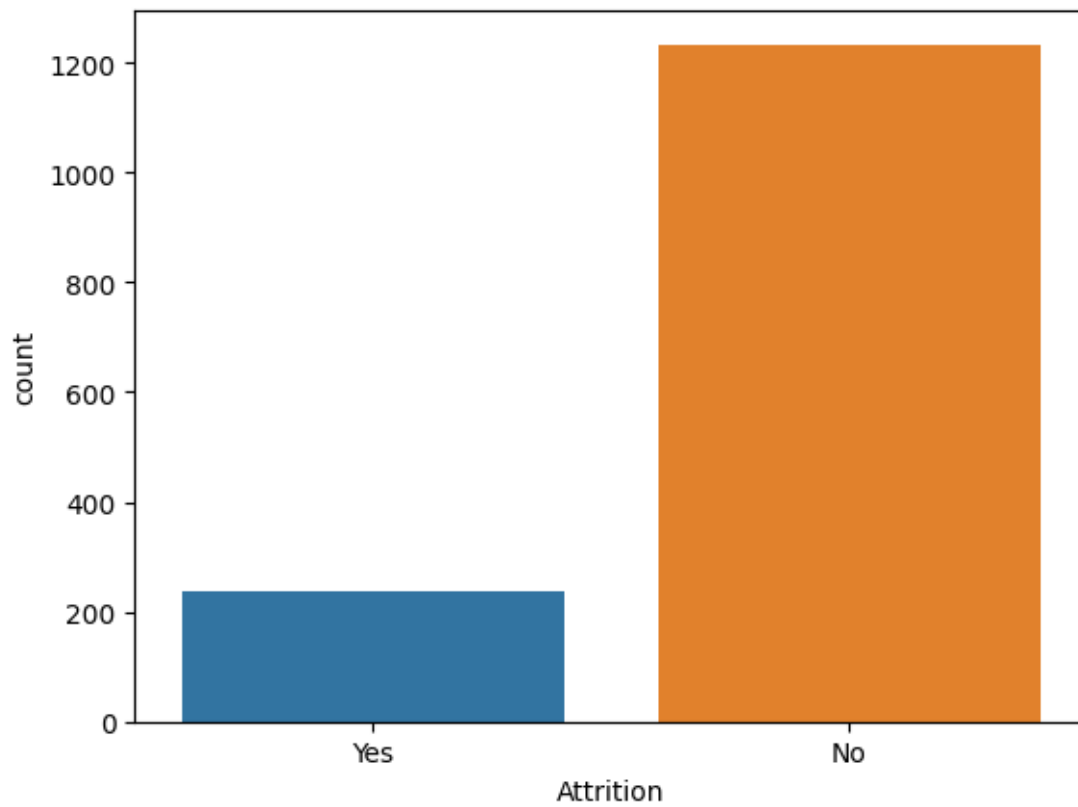
Age	: 0.0
Attrition	: 0.0
BusinessTravel	: 0.0
DailyRate	: 0.0
Department	: 0.0
DistanceFromHome	: 0.0
Education	: 0.0

EducationField : 0.0  
EmployeeCount : 0.0  
EmployeeNumber : 0.0  
EnvironmentSatisfaction : 0.0  
Gender : 0.0  
HourlyRate : 0.0  
JobInvolvement : 0.0  
JobLevel : 0.0  
JobRole : 0.0  
JobSatisfaction : 0.0  
MaritalStatus : 0.0  
MonthlyIncome : 0.0  
MonthlyRate : 0.0  
NumCompaniesWorked : 0.0  
Over18 : 0.0  
OverTime : 0.0  
PercentSalaryHike : 0.0  
PerformanceRating : 0.0  
RelationshipSatisfaction : 0.0  
StandardHours : 0.0  
StockOptionLevel : 0.0  
TotalWorkingYears : 0.0  
TrainingTimesLastYear : 0.0  
WorkLifeBalance : 0.0  
YearsAtCompany : 0.0  
YearsInCurrentRole : 0.0  
YearsSinceLastPromotion : 0.0  
YearsWithCurrManager : 0.0

## 2.4 Data Visualization

```
[9]: sns.countplot(x=df.Attrition,data=df)
```

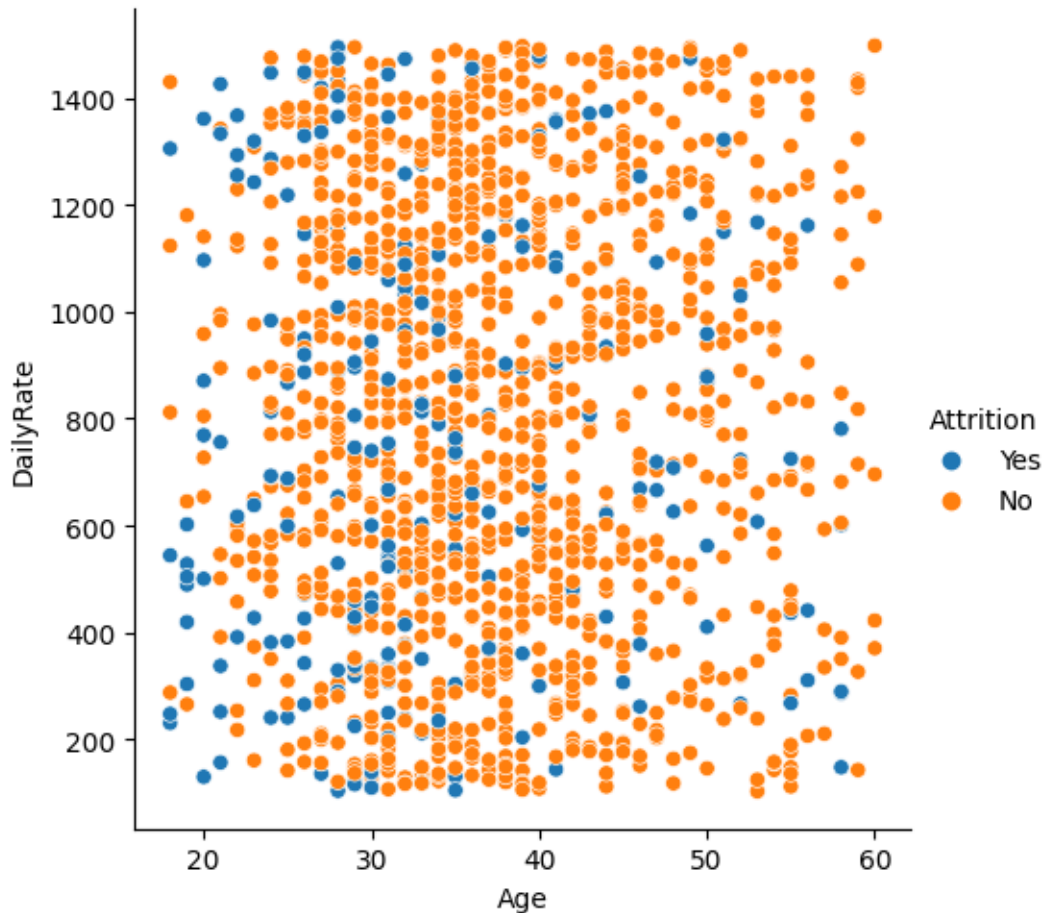
```
[9]: <Axes: xlabel='Attrition', ylabel='count'>
```



```
[10]: sns.relplot(x = 'Age', y = 'DailyRate', hue = 'Attrition', data = df)
```

```
[10]: <seaborn.axisgrid.FacetGrid at 0x21cce40bc90>
```





```
[11]: sns.distplot(df["TotalWorkingYears"])
```

C:\Users\nitin\AppData\Local\Temp\ipykernel\_25332\1374166729.py:1: UserWarning:

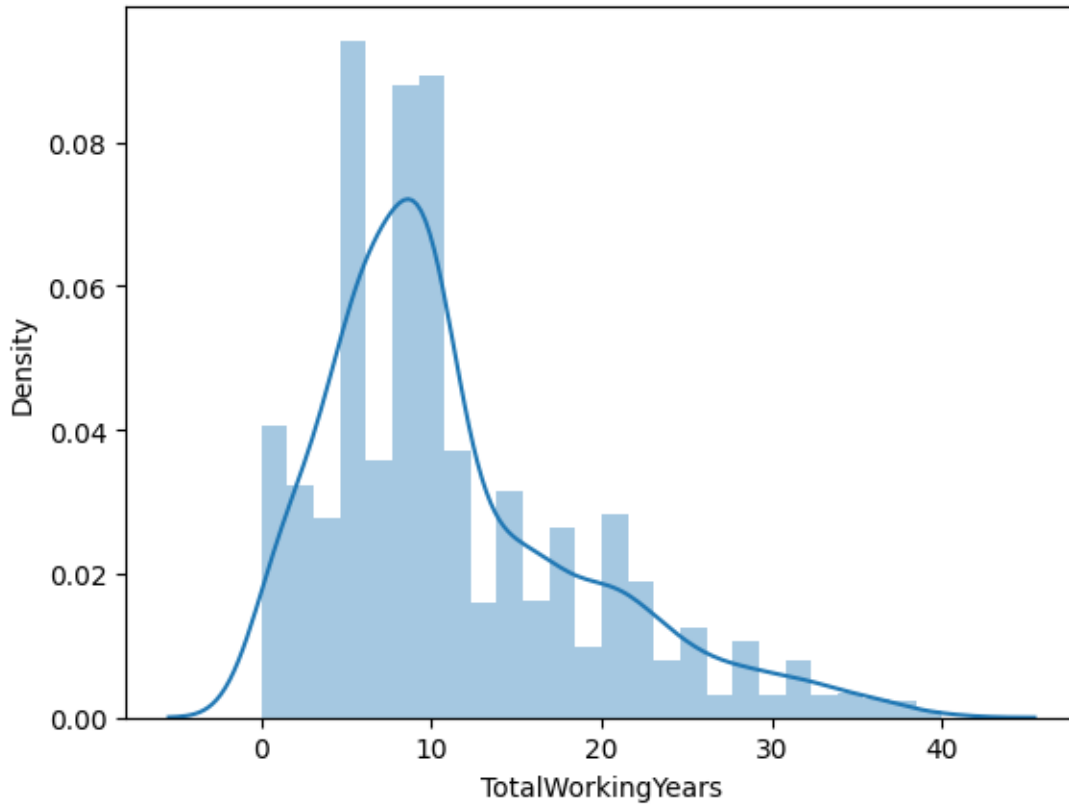
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

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

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

```
sns.distplot(df["TotalWorkingYears"])
```

```
[11]: <Axes: xlabel='TotalWorkingYears', ylabel='Density'>
```



```
[12]: # fig=plt.figure(figsize=(20,6))
# sns.pairplot(df)
```

```
[13]: corr=df.corr(numeric_only=True)
corr
```

```
[13]:
```

	Age	DailyRate	DistanceFromHome	Education
Age	1.000000	0.010661	-0.001686	0.208034 \
DailyRate	0.010661	1.000000	-0.004985	-0.016806
DistanceFromHome	-0.001686	-0.004985	1.000000	0.021042
Education	0.208034	-0.016806	0.021042	1.000000
EmployeeCount	NaN	NaN	NaN	NaN
EmployeeNumber	-0.010145	-0.050990	0.032916	0.042070
EnvironmentSatisfaction	0.010146	0.018355	-0.016075	-0.027128
HourlyRate	0.024287	0.023381	0.031131	0.016775
JobInvolvement	0.029820	0.046135	0.008783	0.042438
JobLevel	0.509604	0.002966	0.005303	0.101589
JobSatisfaction	-0.004892	0.030571	-0.003669	-0.011296
MonthlyIncome	0.497855	0.007707	-0.017014	0.094961
MonthlyRate	0.028051	-0.032182	0.027473	-0.026084
NumCompaniesWorked	0.299635	0.038153	-0.029251	0.126317

PercentSalaryHike	0.003634	0.022704	0.040235	-0.011111
PerformanceRating	0.001904	0.000473	0.027110	-0.024539
RelationshipSatisfaction	0.053535	0.007846	0.006557	-0.009118
StandardHours	NaN	NaN	NaN	NaN
StockOptionLevel	0.037510	0.042143	0.044872	0.018422
TotalWorkingYears	0.680381	0.014515	0.004628	0.148280
TrainingTimesLastYear	-0.019621	0.002453	-0.036942	-0.025100
WorkLifeBalance	-0.021490	-0.037848	-0.026556	0.009819
YearsAtCompany	0.311309	-0.034055	0.009508	0.069114
YearsInCurrentRole	0.212901	0.009932	0.018845	0.060236
YearsSinceLastPromotion	0.216513	-0.033229	0.010029	0.054254
YearsWithCurrManager	0.202089	-0.026363	0.014406	0.069065

	EmployeeCount	EmployeeNumber
Age	NaN	-0.010145 \
DailyRate	NaN	-0.050990
DistanceFromHome	NaN	0.032916
Education	NaN	0.042070
EmployeeCount	NaN	NaN
EmployeeNumber	NaN	1.000000
EnvironmentSatisfaction	NaN	0.017621
HourlyRate	NaN	0.035179
JobInvolvement	NaN	-0.006888
JobLevel	NaN	-0.018519
JobSatisfaction	NaN	-0.046247
MonthlyIncome	NaN	-0.014829
MonthlyRate	NaN	0.012648
NumCompaniesWorked	NaN	-0.001251
PercentSalaryHike	NaN	-0.012944
PerformanceRating	NaN	-0.020359
RelationshipSatisfaction	NaN	-0.069861
StandardHours	NaN	NaN
StockOptionLevel	NaN	0.062227
TotalWorkingYears	NaN	-0.014365
TrainingTimesLastYear	NaN	0.023603
WorkLifeBalance	NaN	0.010309
YearsAtCompany	NaN	-0.011240
YearsInCurrentRole	NaN	-0.008416
YearsSinceLastPromotion	NaN	-0.009019
YearsWithCurrManager	NaN	-0.009197

	EnvironmentSatisfaction	HourlyRate	JobInvolvement
Age	0.010146	0.024287	0.029820 \
DailyRate	0.018355	0.023381	0.046135
DistanceFromHome	-0.016075	0.031131	0.008783
Education	-0.027128	0.016775	0.042438
EmployeeCount	NaN	NaN	NaN

EmployeeNumber	0.017621	0.035179	-0.006888
EnvironmentSatisfaction	1.000000	-0.049857	-0.008278
HourlyRate	-0.049857	1.000000	0.042861
JobInvolvement	-0.008278	0.042861	1.000000
JobLevel	0.001212	-0.027853	-0.012630
JobSatisfaction	-0.006784	-0.071335	-0.021476
MonthlyIncome	-0.006259	-0.015794	-0.015271
MonthlyRate	0.037600	-0.015297	-0.016322
NumCompaniesWorked	0.012594	0.022157	0.015012
PercentSalaryHike	-0.031701	-0.009062	-0.017205
PerformanceRating	-0.029548	-0.002172	-0.029071
RelationshipSatisfaction	0.007665	0.001330	0.034297
StandardHours	NaN	NaN	NaN
StockOptionLevel	0.003432	0.050263	0.021523
TotalWorkingYears	-0.002693	-0.002334	-0.005533
TrainingTimesLastYear	-0.019359	-0.008548	-0.015338
WorkLifeBalance	0.027627	-0.004607	-0.014617
YearsAtCompany	0.001458	-0.019582	-0.021355
YearsInCurrentRole	0.018007	-0.024106	0.008717
YearsSinceLastPromotion	0.016194	-0.026716	-0.024184
YearsWithCurrManager	-0.004999	-0.020123	0.025976

	JobLevel	...	RelationshipSatisfaction
Age	0.509604	...	0.053535 \
DailyRate	0.002966	...	0.007846
DistanceFromHome	0.005303	...	0.006557
Education	0.101589	...	-0.009118
EmployeeCount	NaN	...	NaN
EmployeeNumber	-0.018519	...	-0.069861
EnvironmentSatisfaction	0.001212	...	0.007665
HourlyRate	-0.027853	...	0.001330
JobInvolvement	-0.012630	...	0.034297
JobLevel	1.000000	...	0.021642
JobSatisfaction	-0.001944	...	-0.012454
MonthlyIncome	0.950300	...	0.025873
MonthlyRate	0.039563	...	-0.004085
NumCompaniesWorked	0.142501	...	0.052733
PercentSalaryHike	-0.034730	...	-0.040490
PerformanceRating	-0.021222	...	-0.031351
RelationshipSatisfaction	0.021642	...	1.000000
StandardHours	NaN	...	NaN
StockOptionLevel	0.013984	...	-0.045952
TotalWorkingYears	0.782208	...	0.024054
TrainingTimesLastYear	-0.018191	...	0.002497
WorkLifeBalance	0.037818	...	0.019604
YearsAtCompany	0.534739	...	0.019367
YearsInCurrentRole	0.389447	...	-0.015123

YearsSinceLastPromotion	0.353885 ...	0.033493
YearsWithCurrManager	0.375281 ...	-0.000867

	StandardHours	StockOptionLevel	TotalWorkingYears
Age	NaN	0.037510	0.680381 \
DailyRate	NaN	0.042143	0.014515
DistanceFromHome	NaN	0.044872	0.004628
Education	NaN	0.018422	0.148280
EmployeeCount	NaN	NaN	NaN
EmployeeNumber	NaN	0.062227	-0.014365
EnvironmentSatisfaction	NaN	0.003432	-0.002693
HourlyRate	NaN	0.050263	-0.002334
JobInvolvement	NaN	0.021523	-0.005533
JobLevel	NaN	0.013984	0.782208
JobSatisfaction	NaN	0.010690	-0.020185
MonthlyIncome	NaN	0.005408	0.772893
MonthlyRate	NaN	-0.034323	0.026442
NumCompaniesWorked	NaN	0.030075	0.237639
PercentSalaryHike	NaN	0.007528	-0.020608
PerformanceRating	NaN	0.003506	0.006744
RelationshipSatisfaction	NaN	-0.045952	0.024054
StandardHours	NaN	NaN	NaN
StockOptionLevel	NaN	1.000000	0.010136
TotalWorkingYears	NaN	0.010136	1.000000
TrainingTimesLastYear	NaN	0.011274	-0.035662
WorkLifeBalance	NaN	0.004129	0.001008
YearsAtCompany	NaN	0.015058	0.628133
YearsInCurrentRole	NaN	0.050818	0.460365
YearsSinceLastPromotion	NaN	0.014352	0.404858
YearsWithCurrManager	NaN	0.024698	0.459188

	TrainingTimesLastYear	WorkLifeBalance
Age	-0.019621	-0.021490 \
DailyRate	0.002453	-0.037848
DistanceFromHome	-0.036942	-0.026556
Education	-0.025100	0.009819
EmployeeCount	NaN	NaN
EmployeeNumber	0.023603	0.010309
EnvironmentSatisfaction	-0.019359	0.027627
HourlyRate	-0.008548	-0.004607
JobInvolvement	-0.015338	-0.014617
JobLevel	-0.018191	0.037818
JobSatisfaction	-0.005779	-0.019459
MonthlyIncome	-0.021736	0.030683
MonthlyRate	0.001467	0.007963
NumCompaniesWorked	-0.066054	-0.008366
PercentSalaryHike	-0.005221	-0.003280

PerformanceRating	-0.015579	0.002572
RelationshipSatisfaction	0.002497	0.019604
StandardHours	NaN	NaN
StockOptionLevel	0.011274	0.004129
TotalWorkingYears	-0.035662	0.001008
TrainingTimesLastYear	1.000000	0.028072
WorkLifeBalance	0.028072	1.000000
YearsAtCompany	0.003569	0.012089
YearsInCurrentRole	-0.005738	0.049856
YearsSinceLastPromotion	-0.002067	0.008941
YearsWithCurrManager	-0.004096	0.002759

	YearsAtCompany	YearsInCurrentRole	
Age	0.311309	0.212901	\
DailyRate	-0.034055	0.009932	
DistanceFromHome	0.009508	0.018845	
Education	0.069114	0.060236	
EmployeeCount	NaN	NaN	
EmployeeNumber	-0.011240	-0.008416	
EnvironmentSatisfaction	0.001458	0.018007	
HourlyRate	-0.019582	-0.024106	
JobInvolvement	-0.021355	0.008717	
JobLevel	0.534739	0.389447	
JobSatisfaction	-0.003803	-0.002305	
MonthlyIncome	0.514285	0.363818	
MonthlyRate	-0.023655	-0.012815	
NumCompaniesWorked	-0.118421	-0.090754	
PercentSalaryHike	-0.035991	-0.001520	
PerformanceRating	0.003435	0.034986	
RelationshipSatisfaction	0.019367	-0.015123	
StandardHours	NaN	NaN	
StockOptionLevel	0.015058	0.050818	
TotalWorkingYears	0.628133	0.460365	
TrainingTimesLastYear	0.003569	-0.005738	
WorkLifeBalance	0.012089	0.049856	
YearsAtCompany	1.000000	0.758754	
YearsInCurrentRole	0.758754	1.000000	
YearsSinceLastPromotion	0.618409	0.548056	
YearsWithCurrManager	0.769212	0.714365	

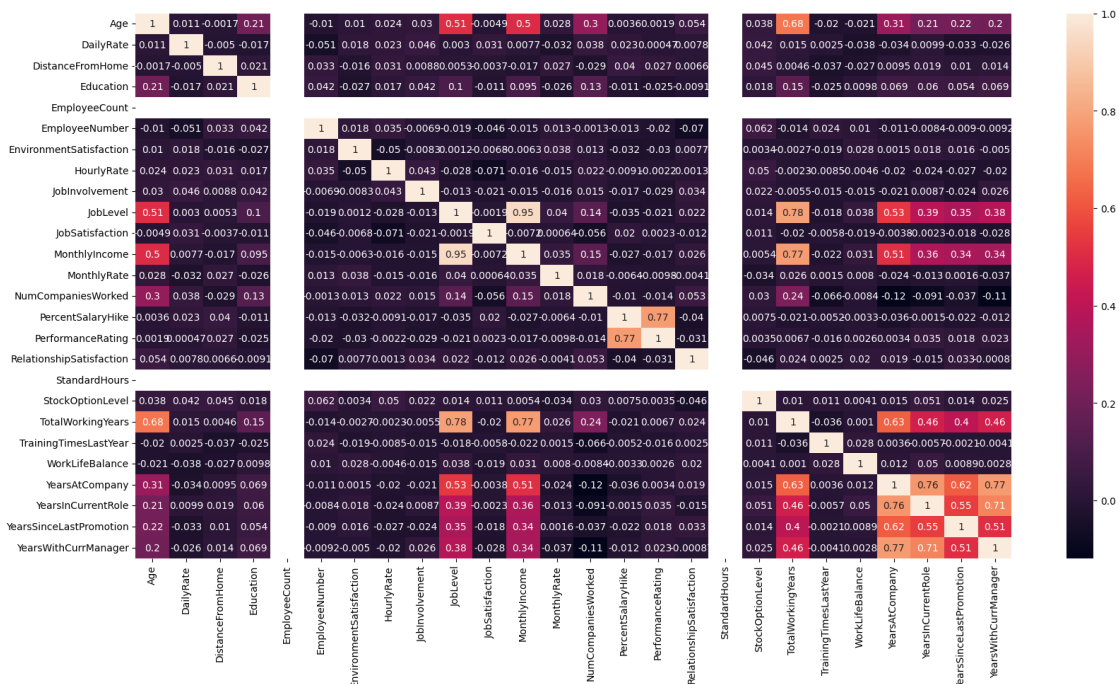
	YearsSinceLastPromotion	YearsWithCurrManager
Age	0.216513	0.202089
DailyRate	-0.033229	-0.026363
DistanceFromHome	0.010029	0.014406
Education	0.054254	0.069065
EmployeeCount	NaN	NaN
EmployeeNumber	-0.009019	-0.009197

EnvironmentSatisfaction	0.016194	-0.004999
HourlyRate	-0.026716	-0.020123
JobInvolvement	-0.024184	0.025976
JobLevel	0.353885	0.375281
JobSatisfaction	-0.018214	-0.027656
MonthlyIncome	0.344978	0.344079
MonthlyRate	0.001567	-0.036746
NumCompaniesWorked	-0.036814	-0.110319
PercentSalaryHike	-0.022154	-0.011985
PerformanceRating	0.017896	0.022827
RelationshipSatisfaction	0.033493	-0.000867
StandardHours	NaN	NaN
StockOptionLevel	0.014352	0.024698
TotalWorkingYears	0.404858	0.459188
TrainingTimesLastYear	-0.002067	-0.004096
WorkLifeBalance	0.008941	0.002759
YearsAtCompany	0.618409	0.769212
YearsInCurrentRole	0.548056	0.714365
YearsSinceLastPromotion	1.000000	0.510224
YearsWithCurrManager	0.510224	1.000000

[26 rows x 26 columns]

```
[14]: fig=plt.figure(figsize=(20,10))
sns.heatmap(corr,annot=True,)
```

[14]: <Axes: >



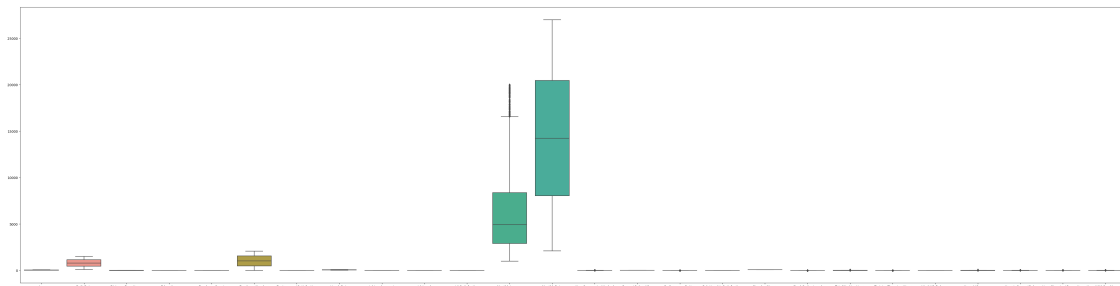
## 2.5 Outlier Detection

```
[15]: df.columns
```

```
[15]: Index(['Age', 'Attrition', 'BusinessTravel', 'DailyRate', 'Department',  
        'DistanceFromHome', 'Education', 'EducationField', 'EmployeeCount',  
        'EmployeeNumber', 'EnvironmentSatisfaction', 'Gender', 'HourlyRate',  
        'JobInvolvement', 'JobLevel', 'JobRole', 'JobSatisfaction',  
        'MaritalStatus', 'MonthlyIncome', 'MonthlyRate', 'NumCompaniesWorked',  
        'Over18', 'OverTime', 'PercentSalaryHike', 'PerformanceRating',  
        'RelationshipSatisfaction', 'StandardHours', 'StockOptionLevel',  
        'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance',  
        'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion',  
        'YearsWithCurrManager'],  
        dtype='object')
```

```
[16]: fig=plt.figure(figsize=(60,15))  
sns.boxplot(df)
```

```
[16]: <Axes: >
```

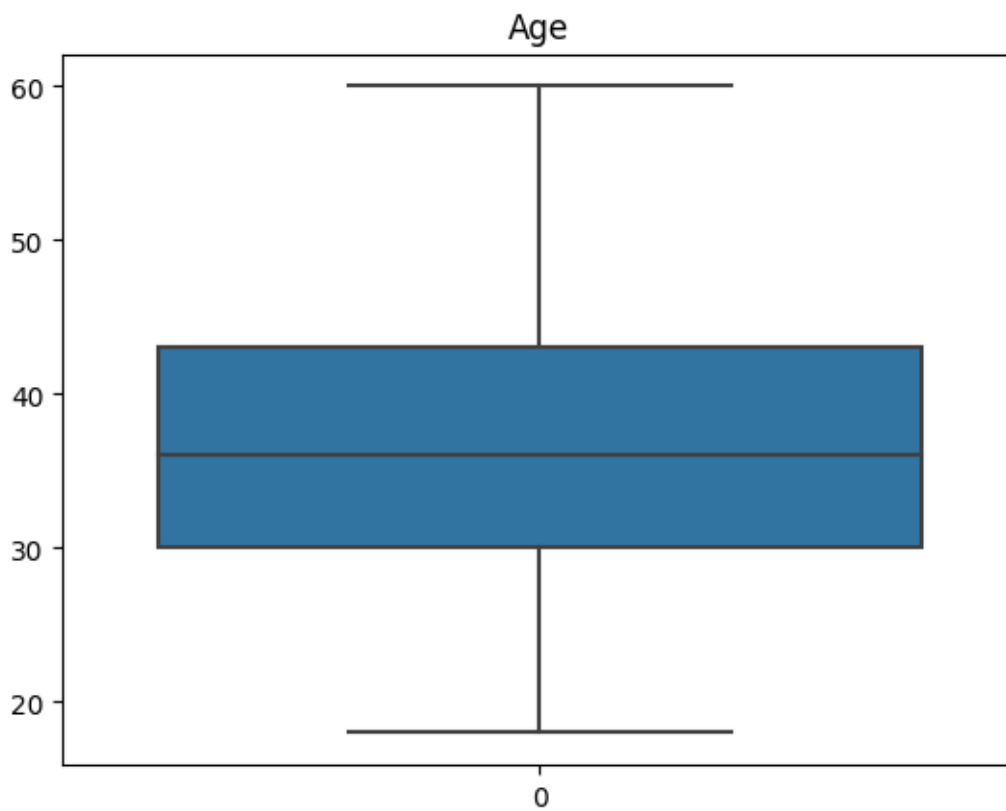


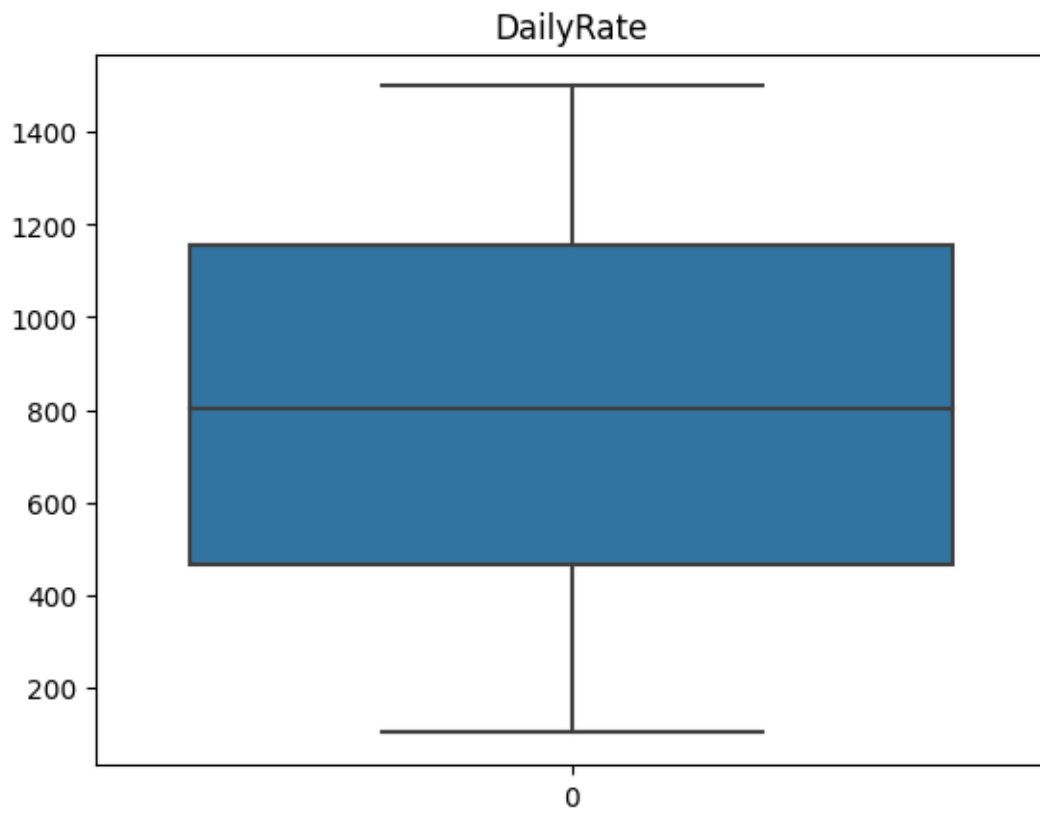
```
[17]: for i in df.columns:  
        if(df[i].dtype!=object):  
            plt.figure()  
            plt.title(i)  
            sns.boxplot(df[i])
```

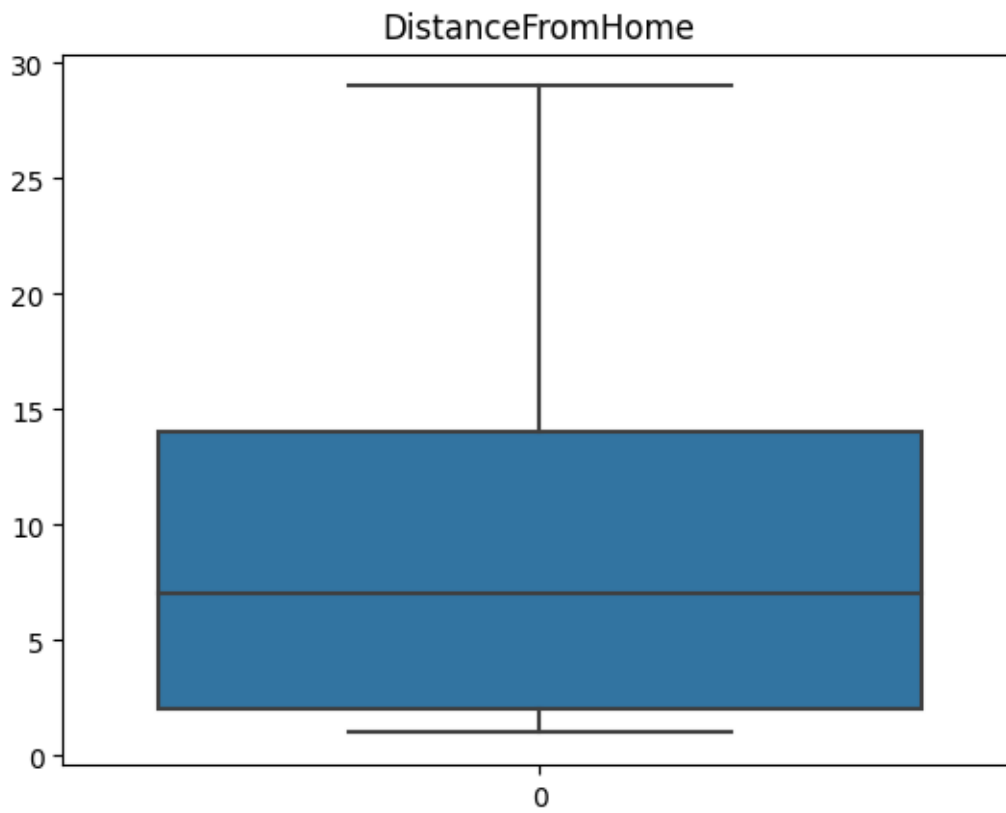
C:\Users\nitin\AppData\Local\Temp\ipykernel\_25332\2388892046.py:3:

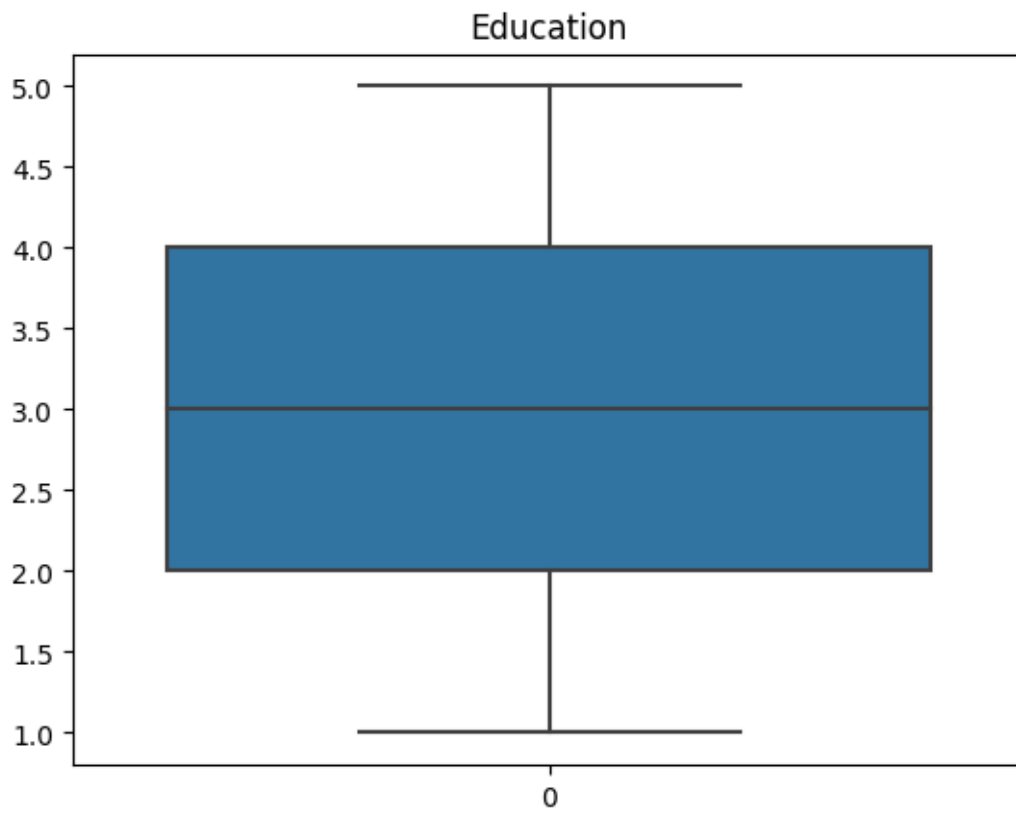
RuntimeWarning: More than 20 figures have been opened. Figures created through the pyplot interface (`matplotlib.pyplot.figure`) are retained until explicitly closed and may consume too much memory. (To control this warning, see the rcParam `figure.max_open_warning`). Consider using `matplotlib.pyplot.close`.\nplt.figure()

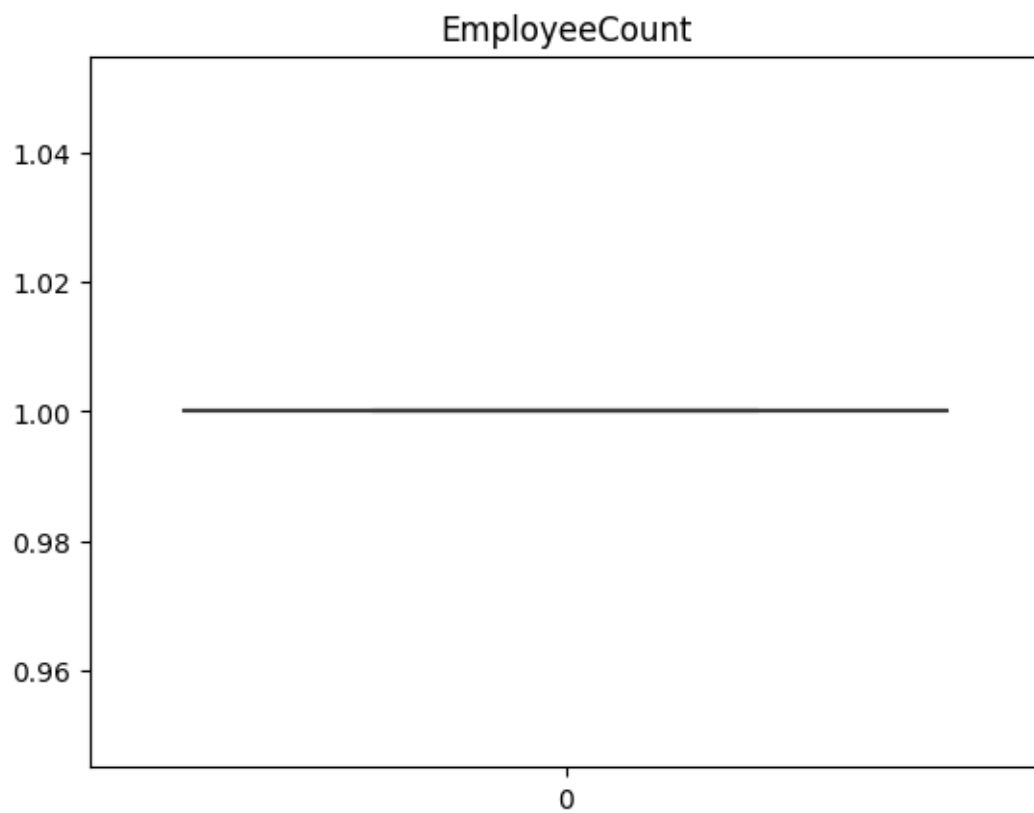


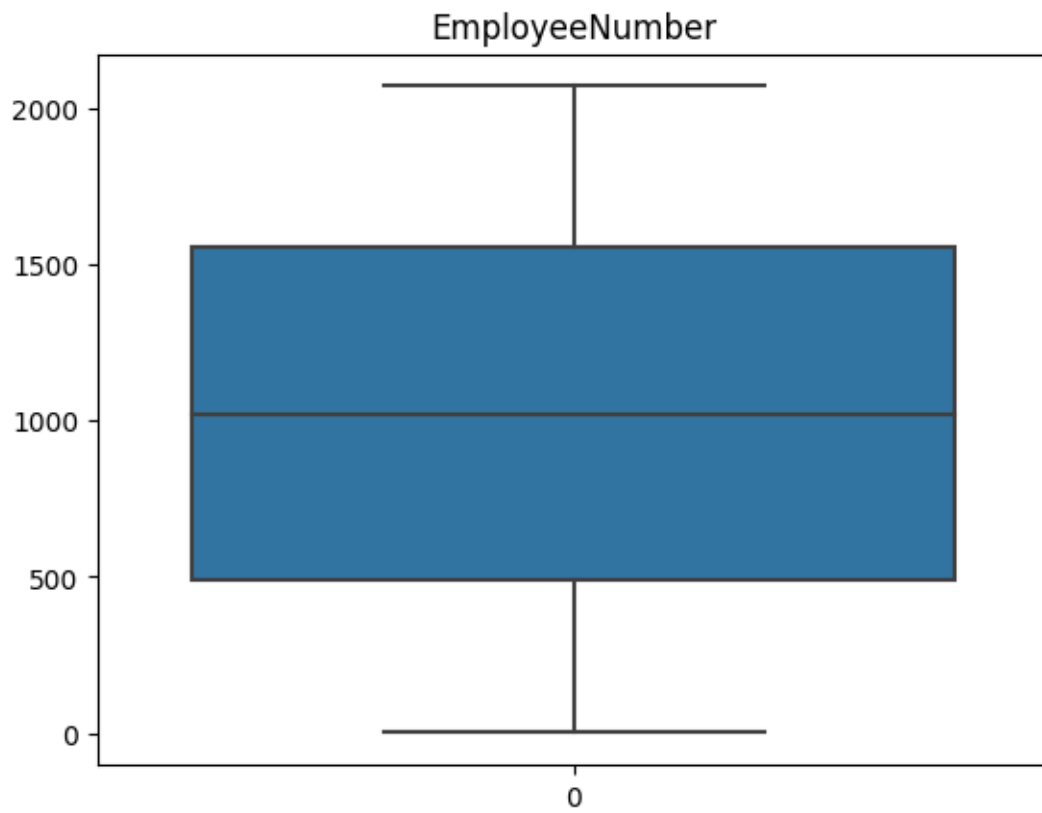


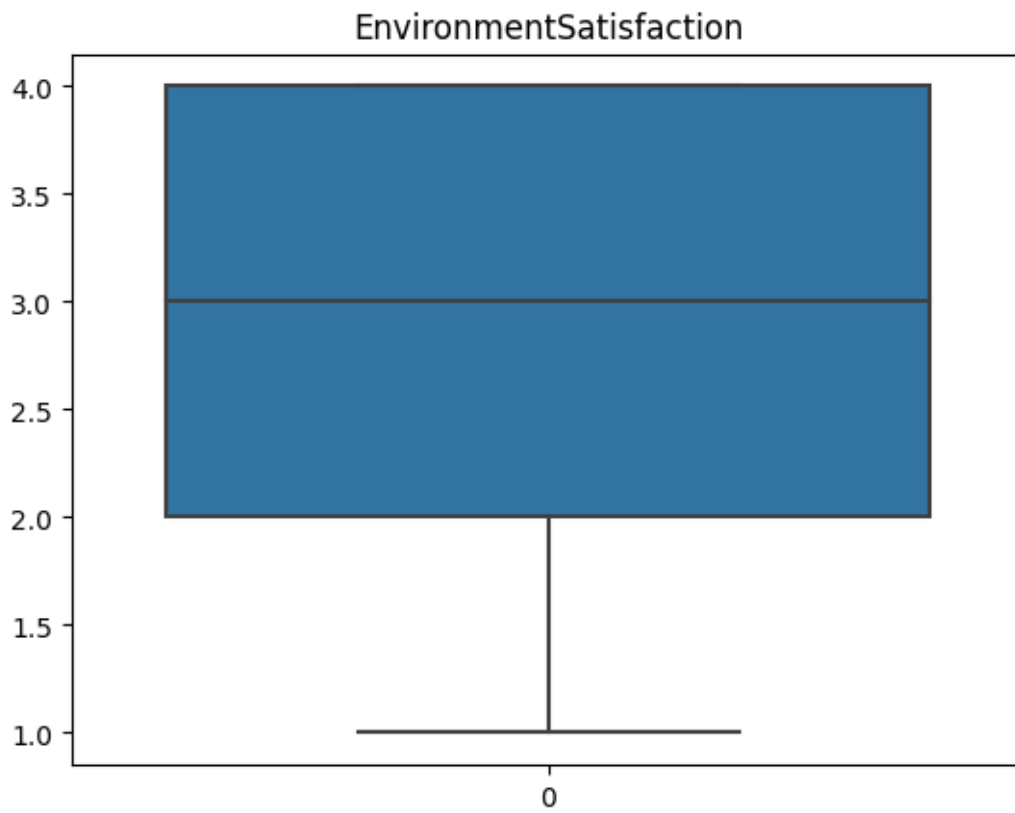


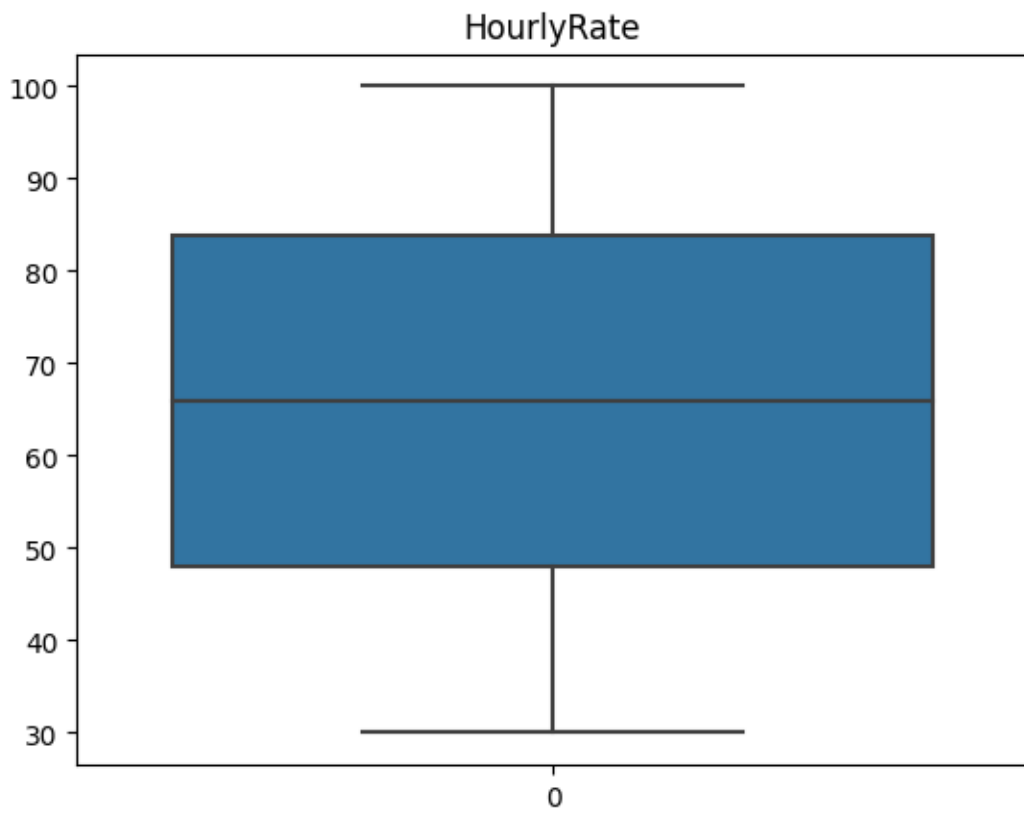




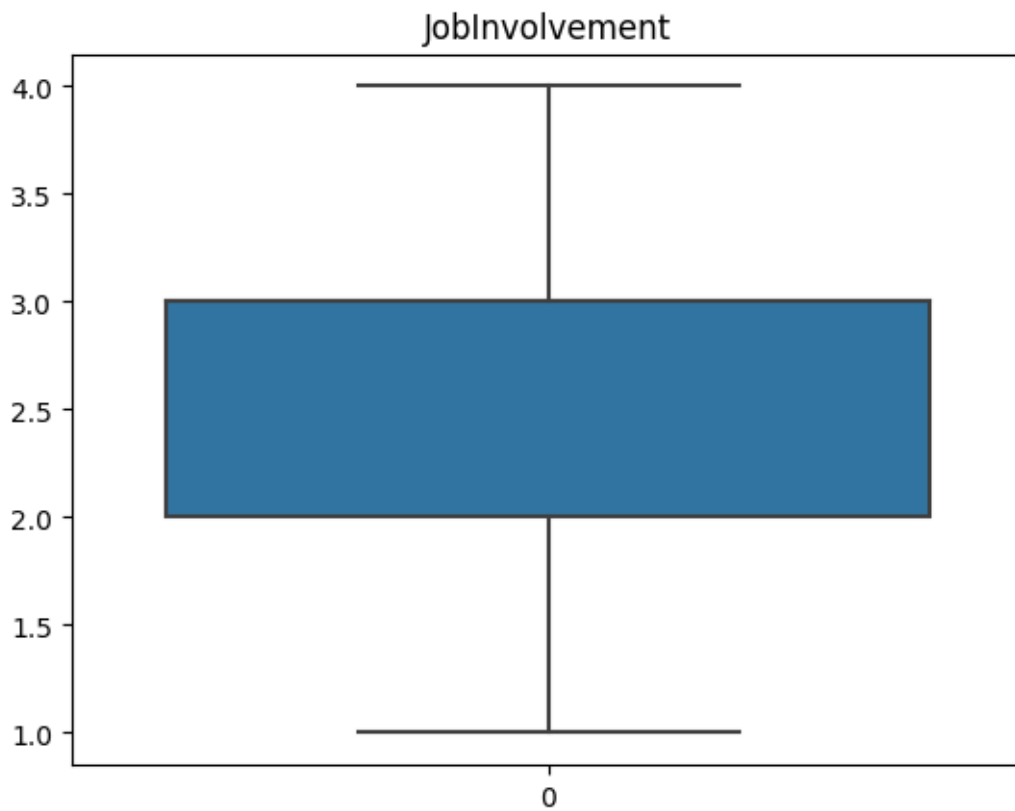


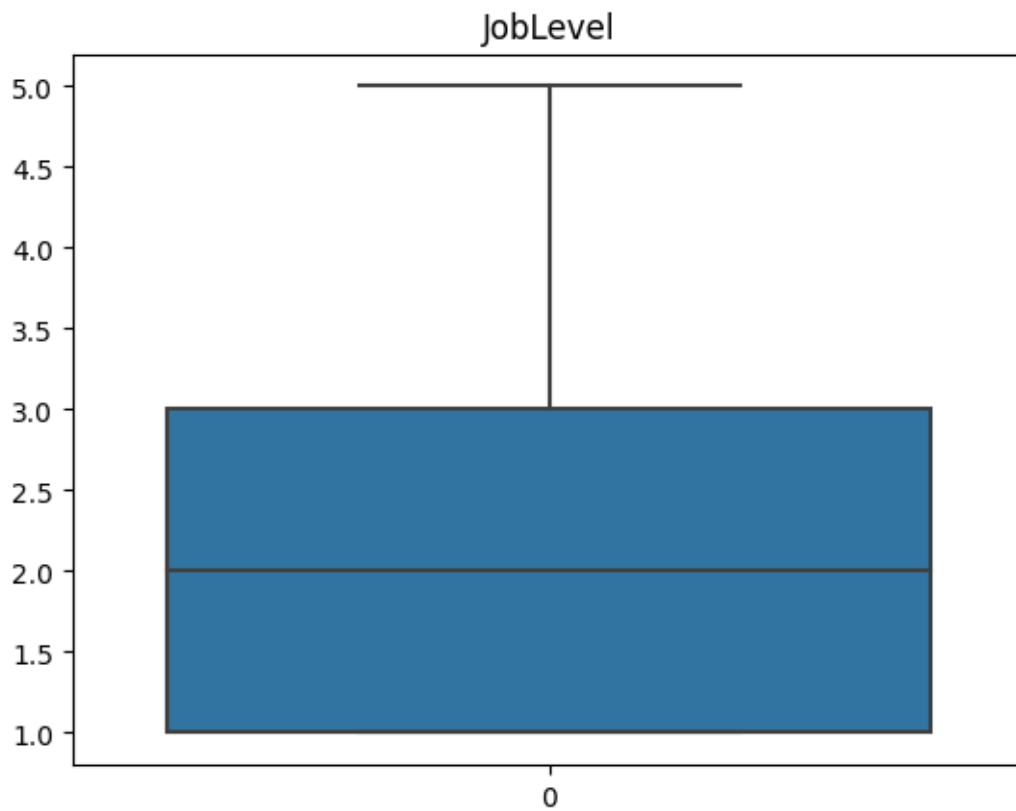


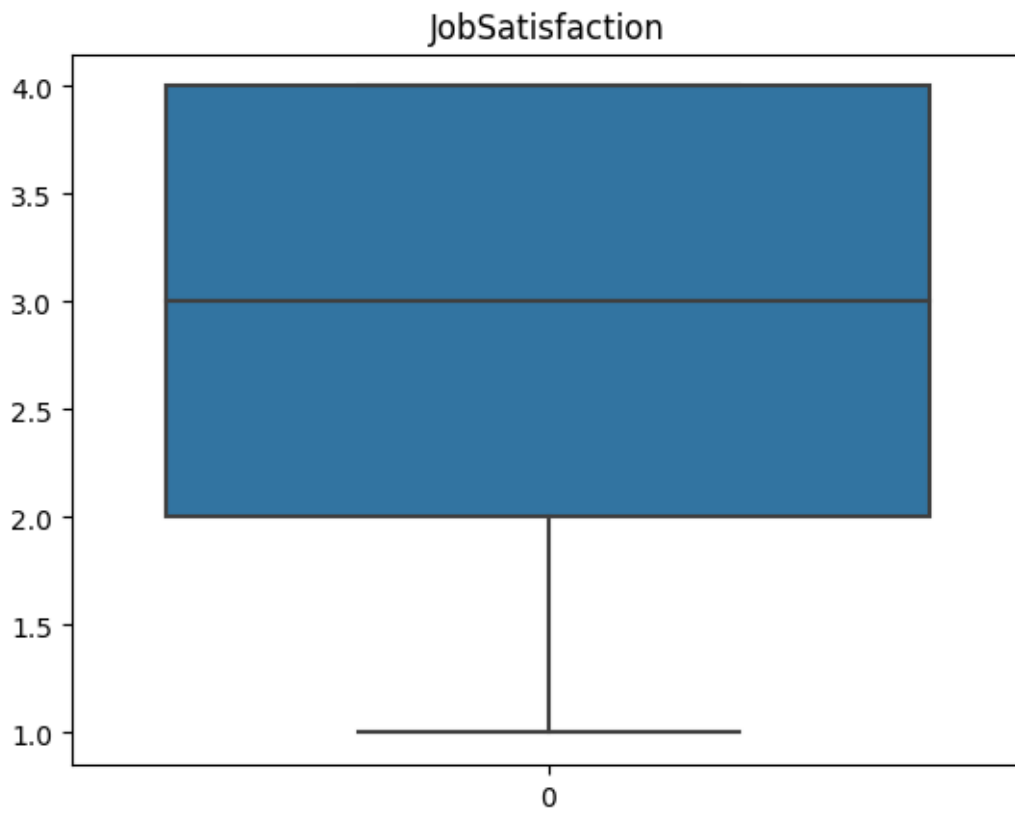


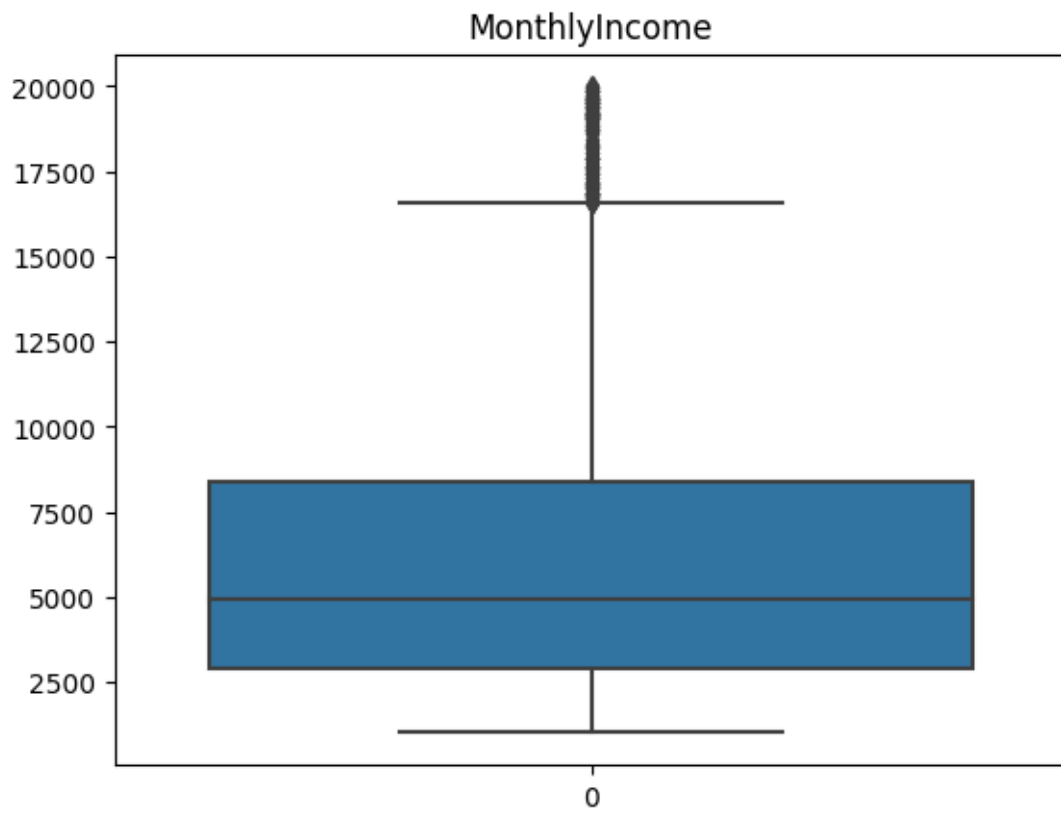


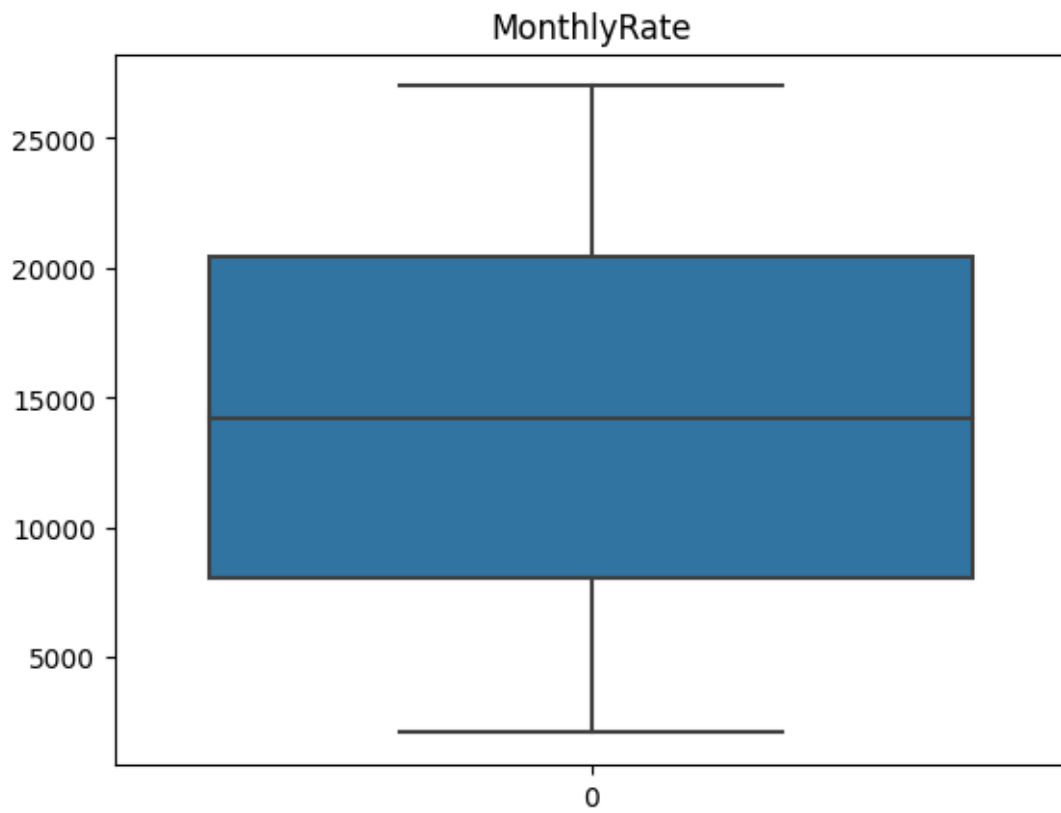


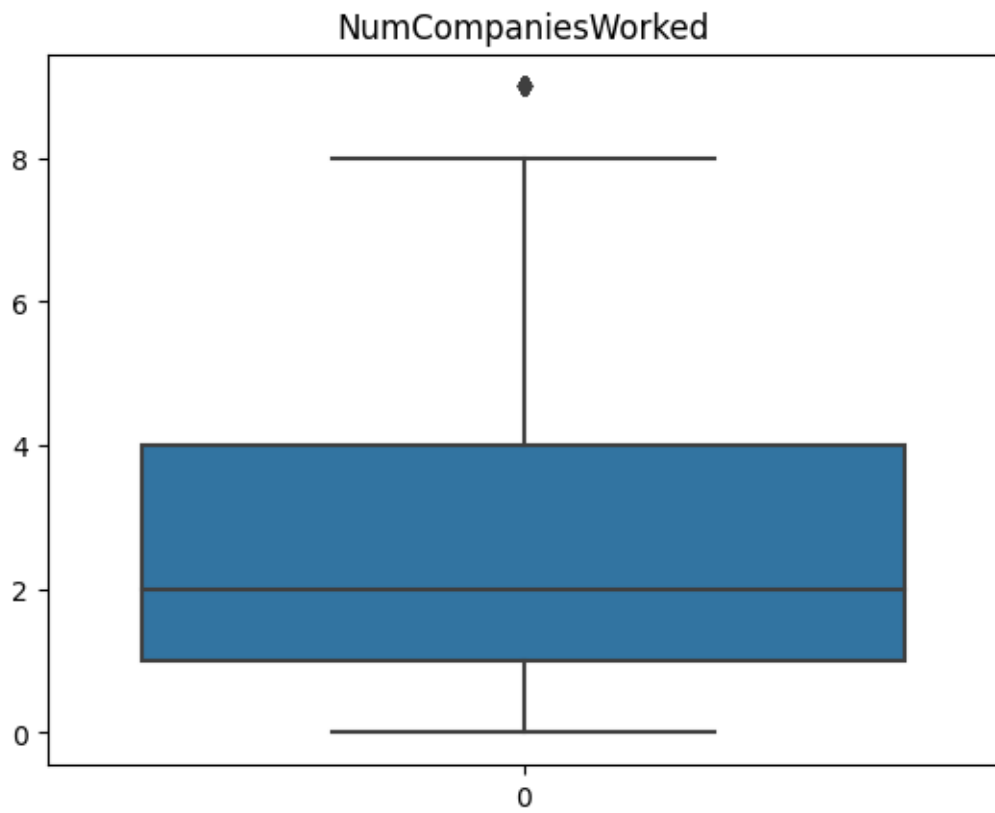


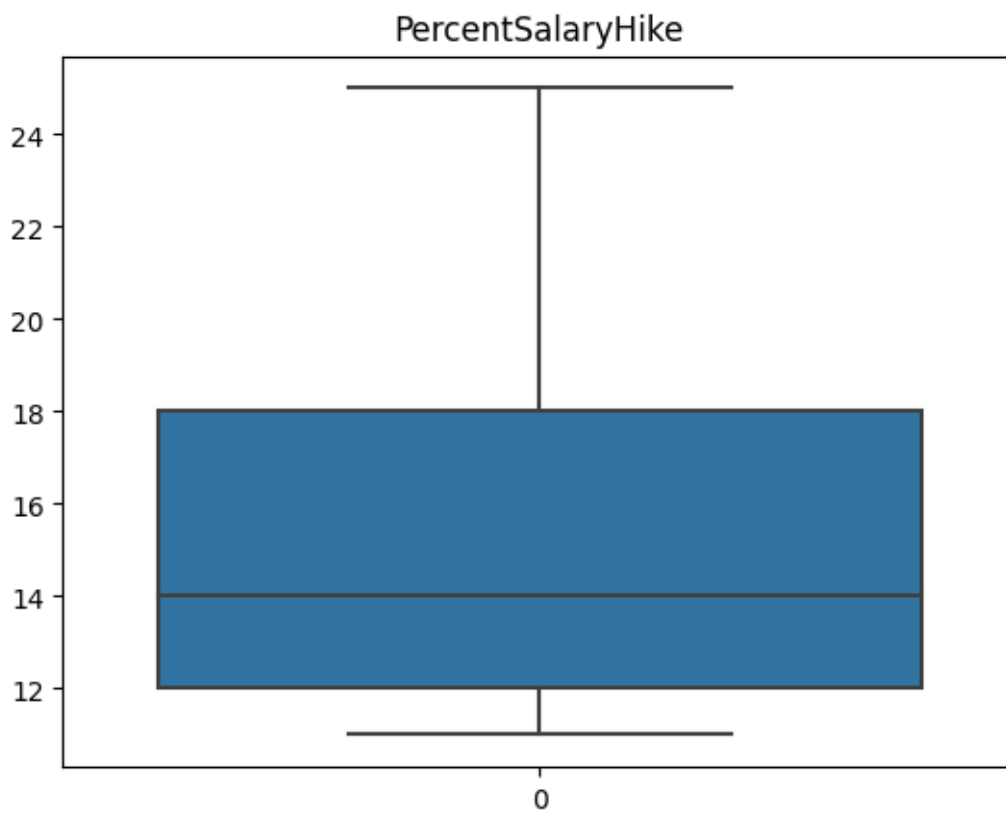


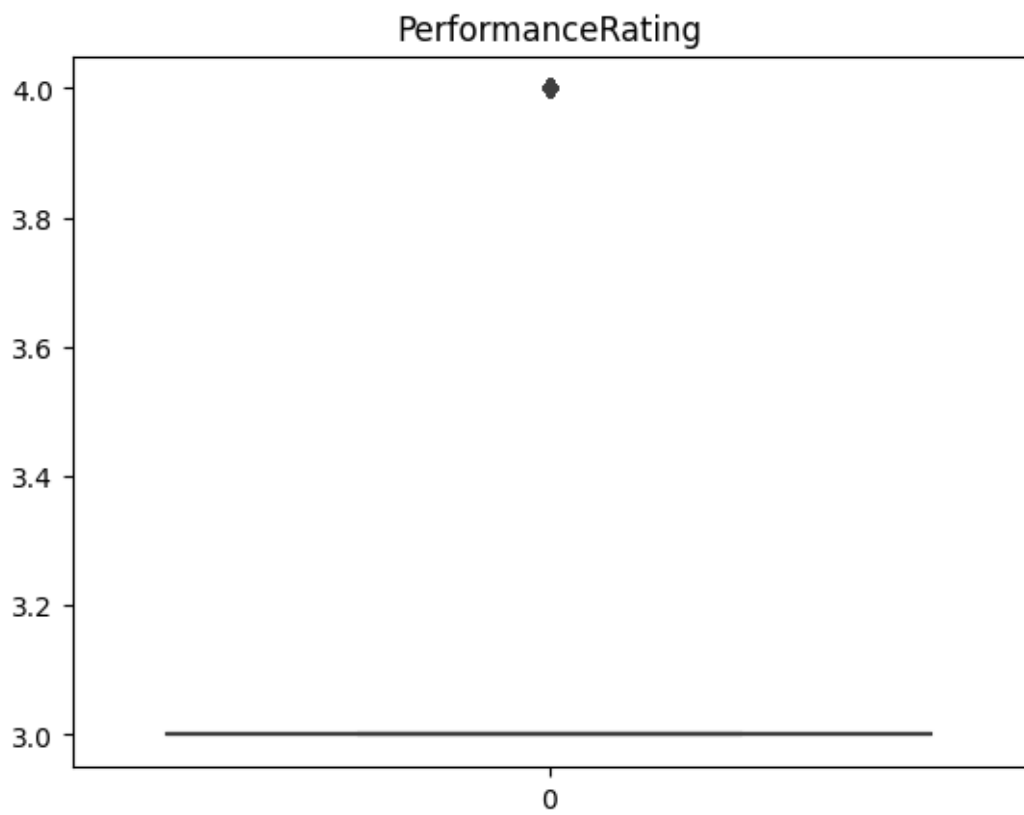




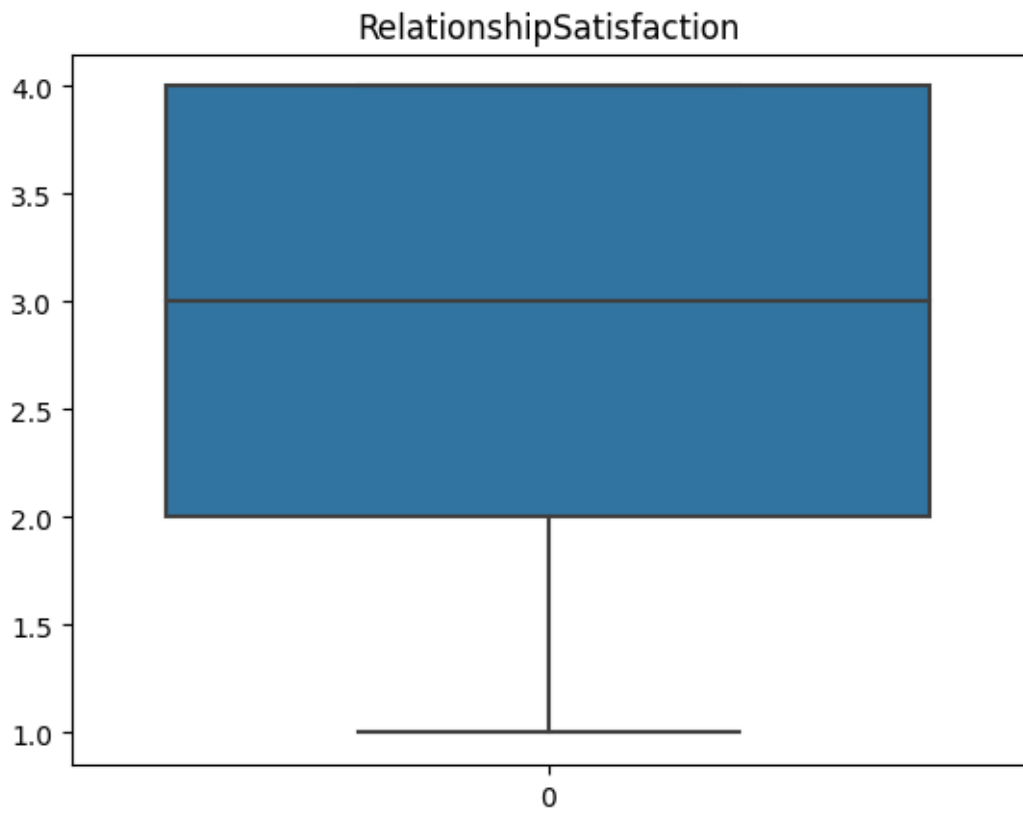


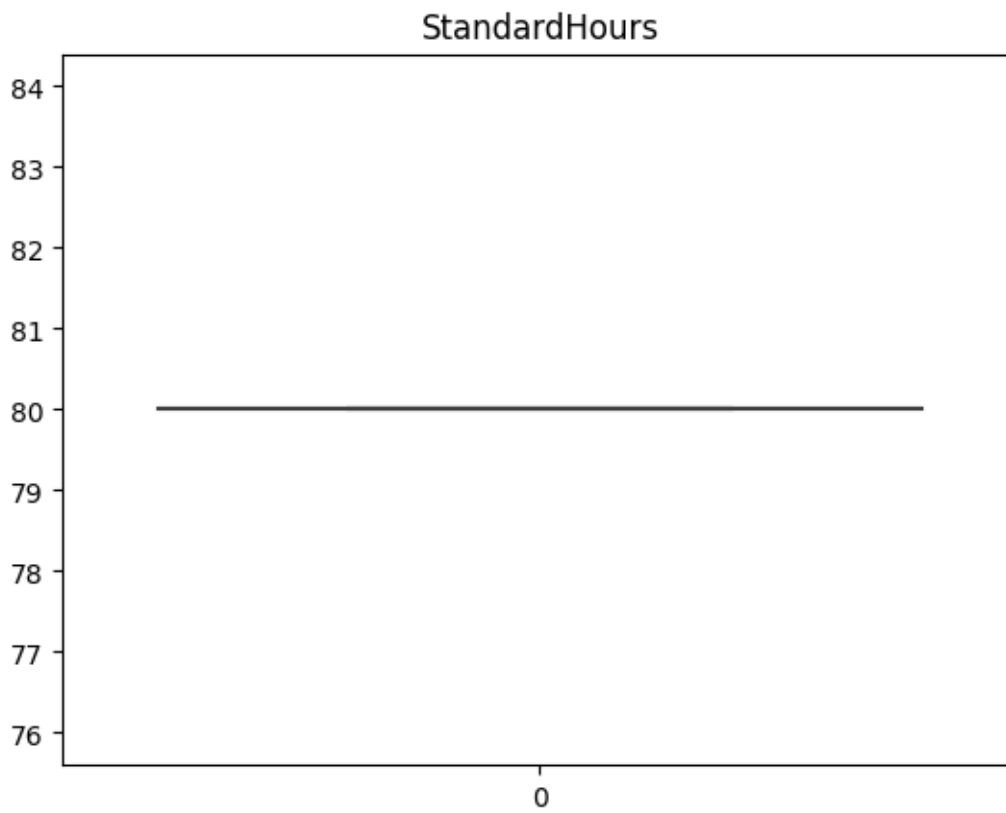


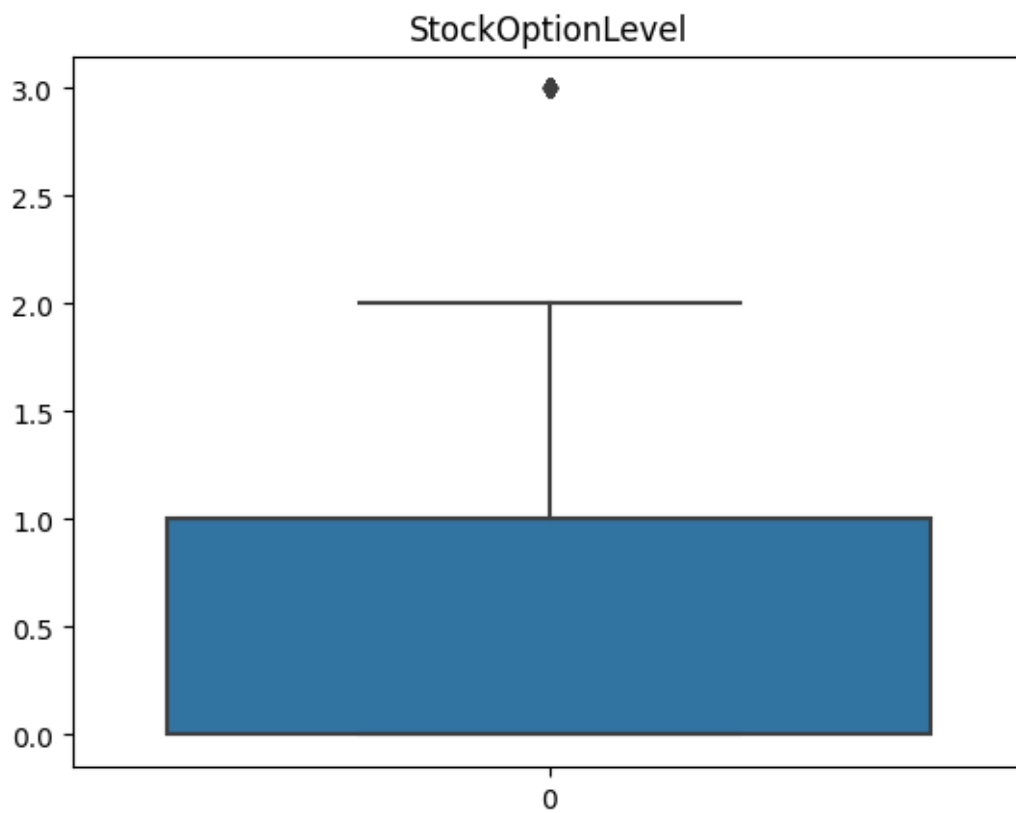


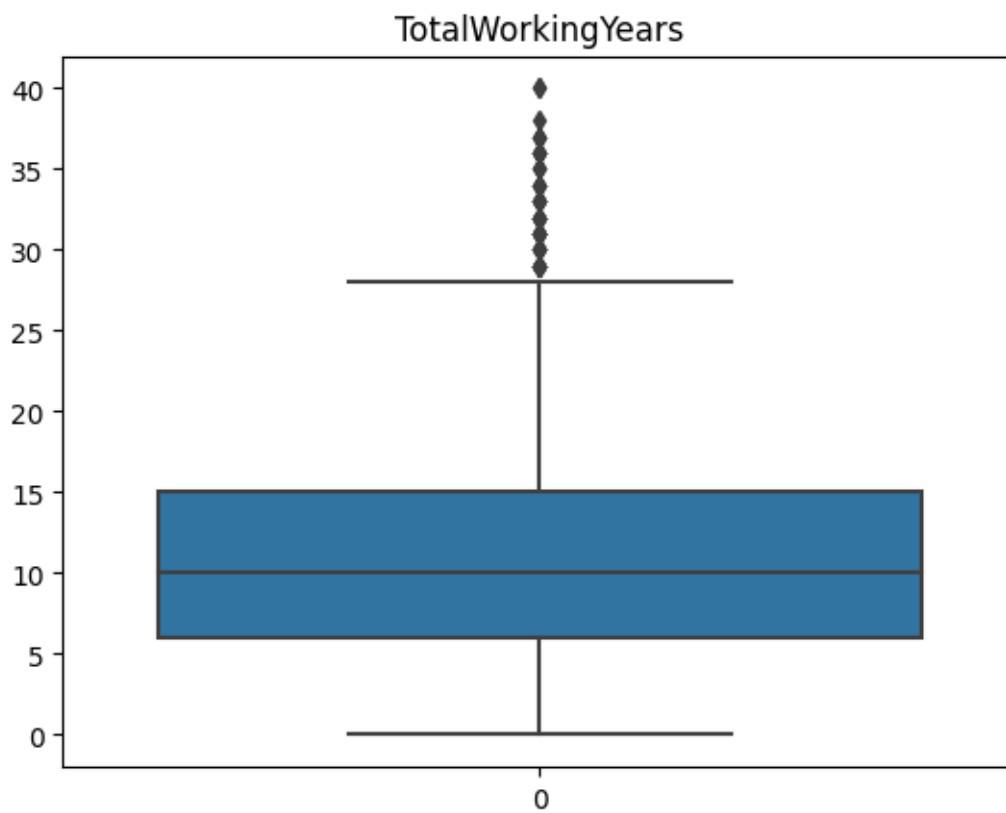


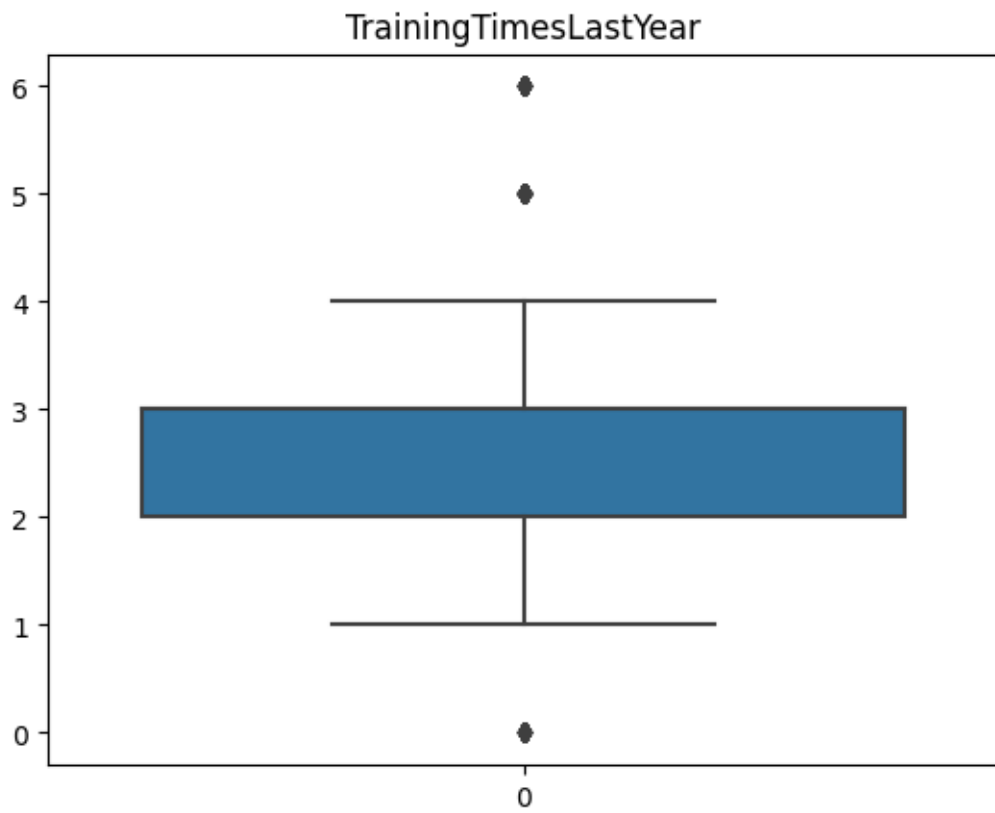


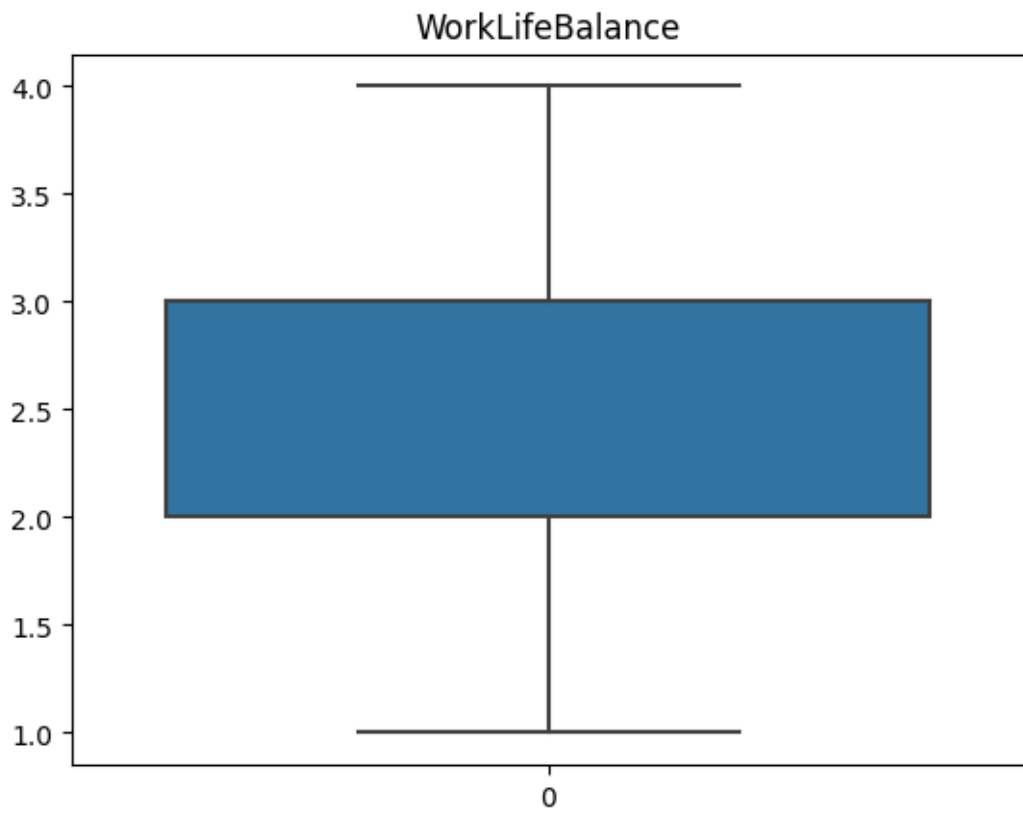


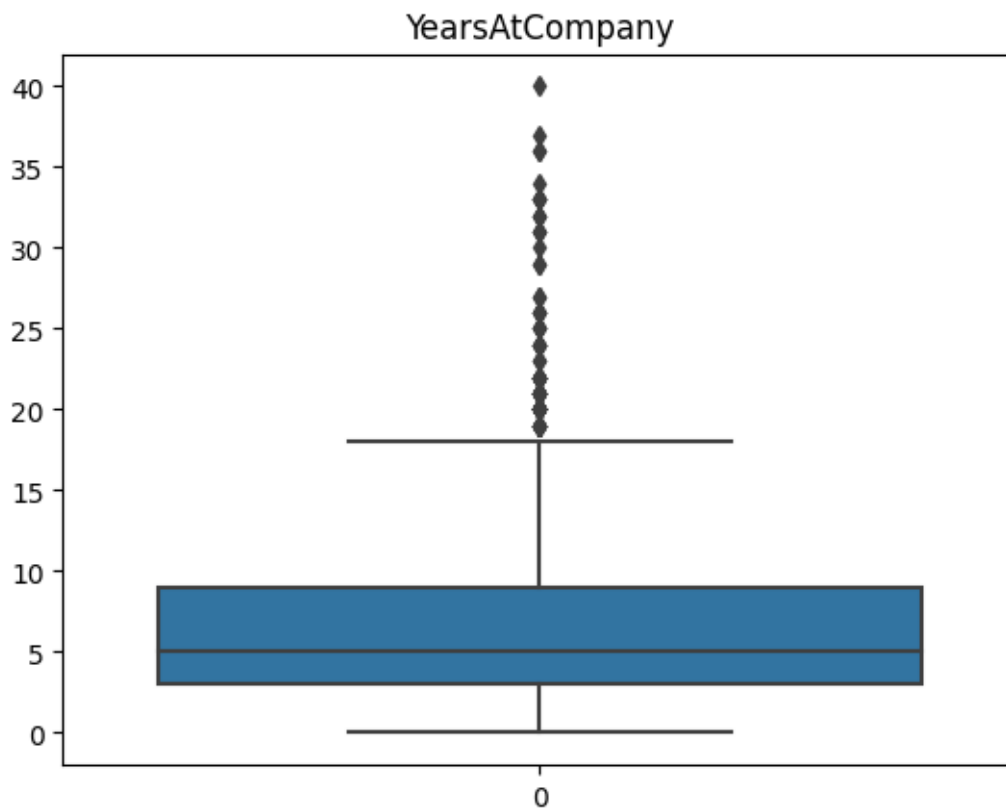


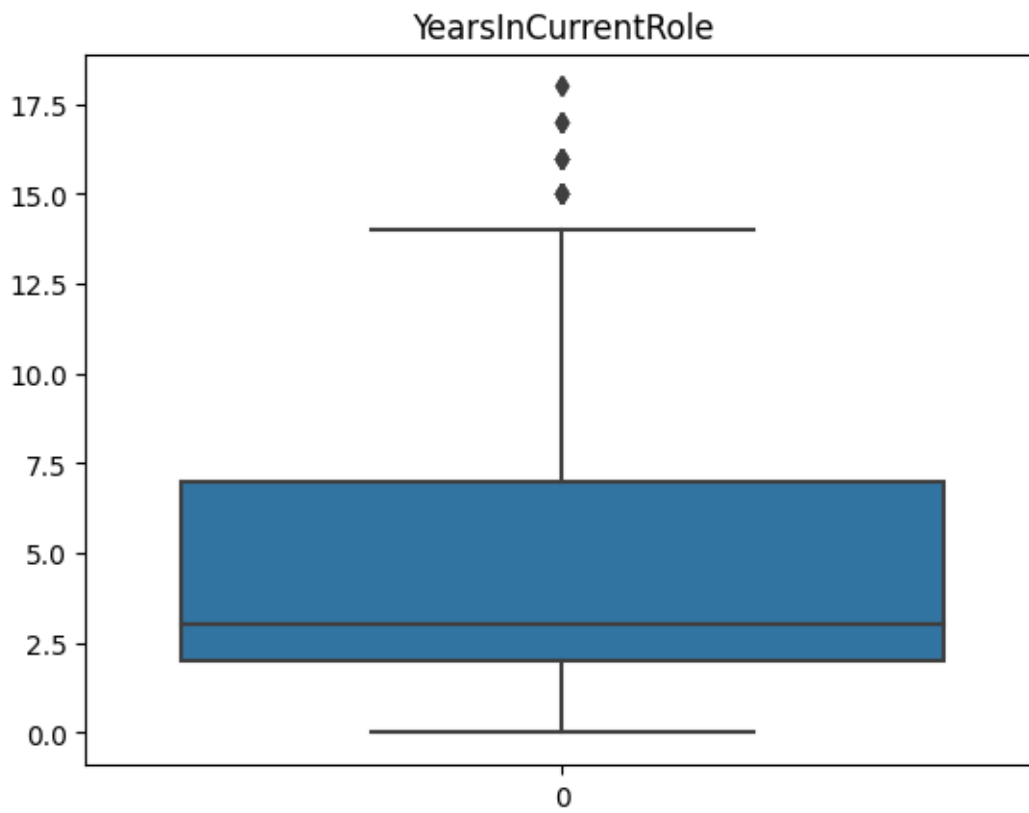




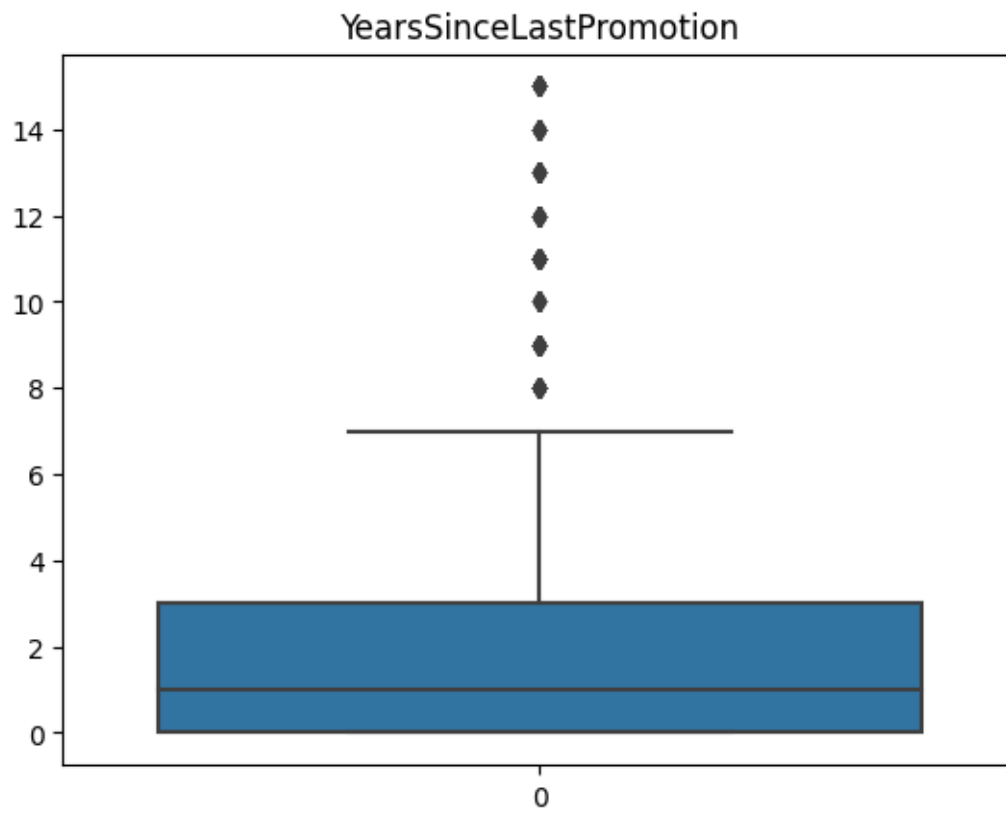


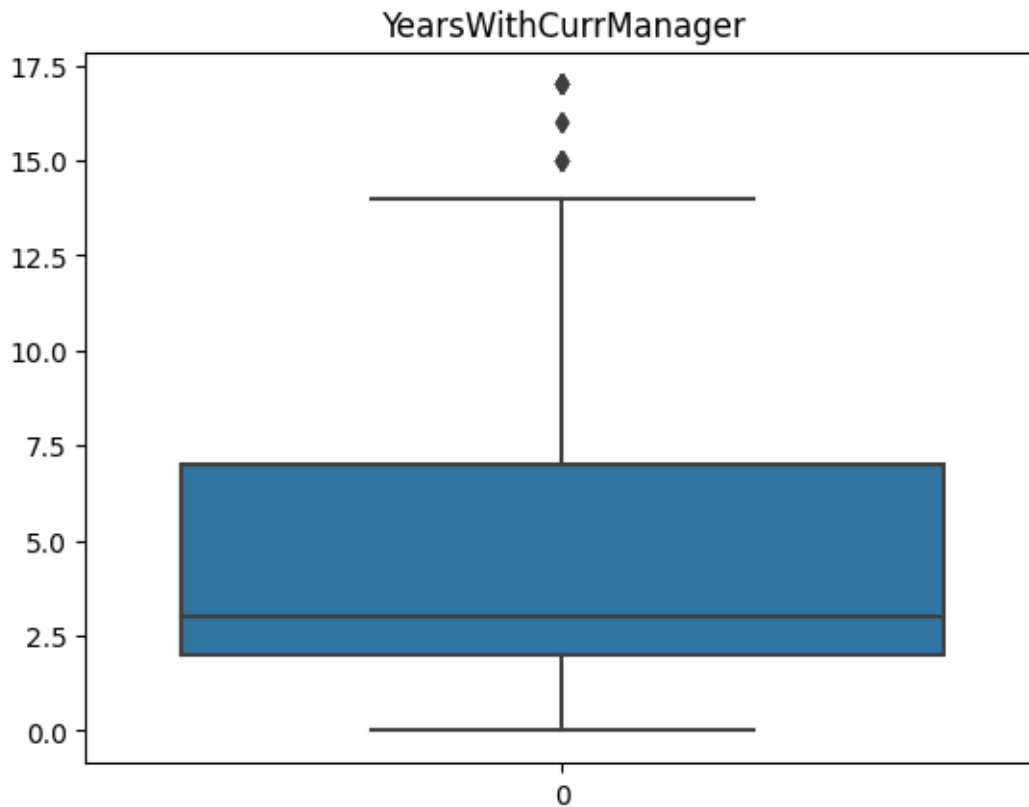












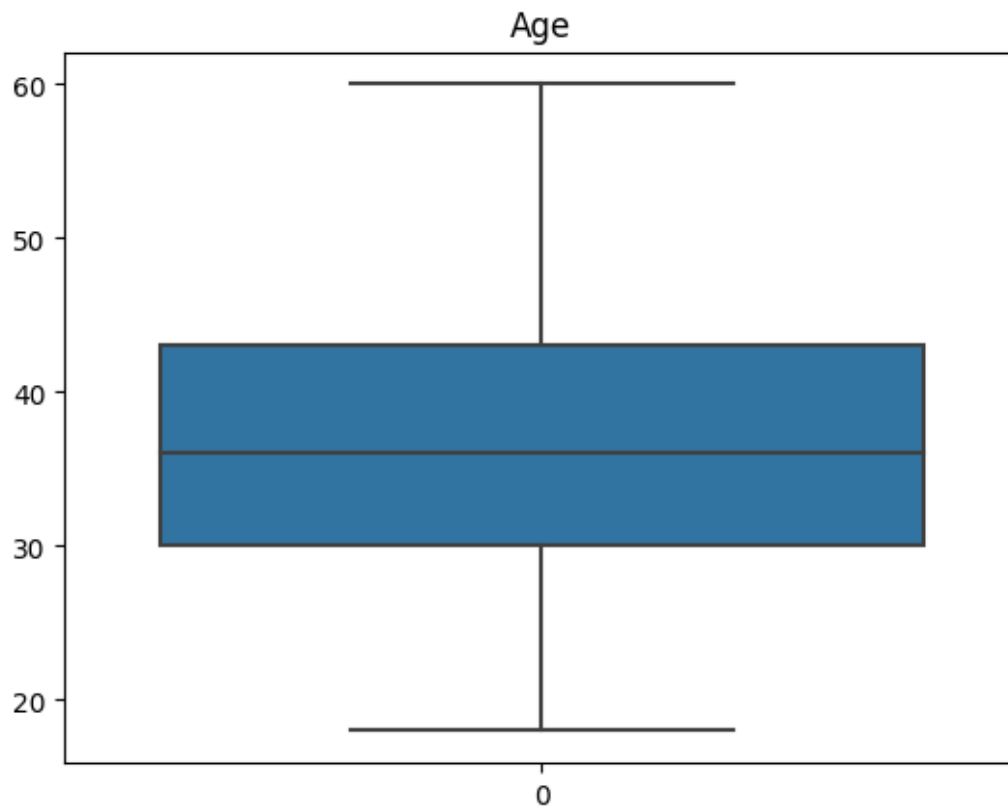
```
[18]: l=["MonthlyIncome","NumCompaniesWorked","StockOptionLevel","TotalWorkingYears","TrainingTimesL
↪ "YearsInCurrentRole", "YearsSinceLastPromotion",
      "YearsWithCurrManager"]
```

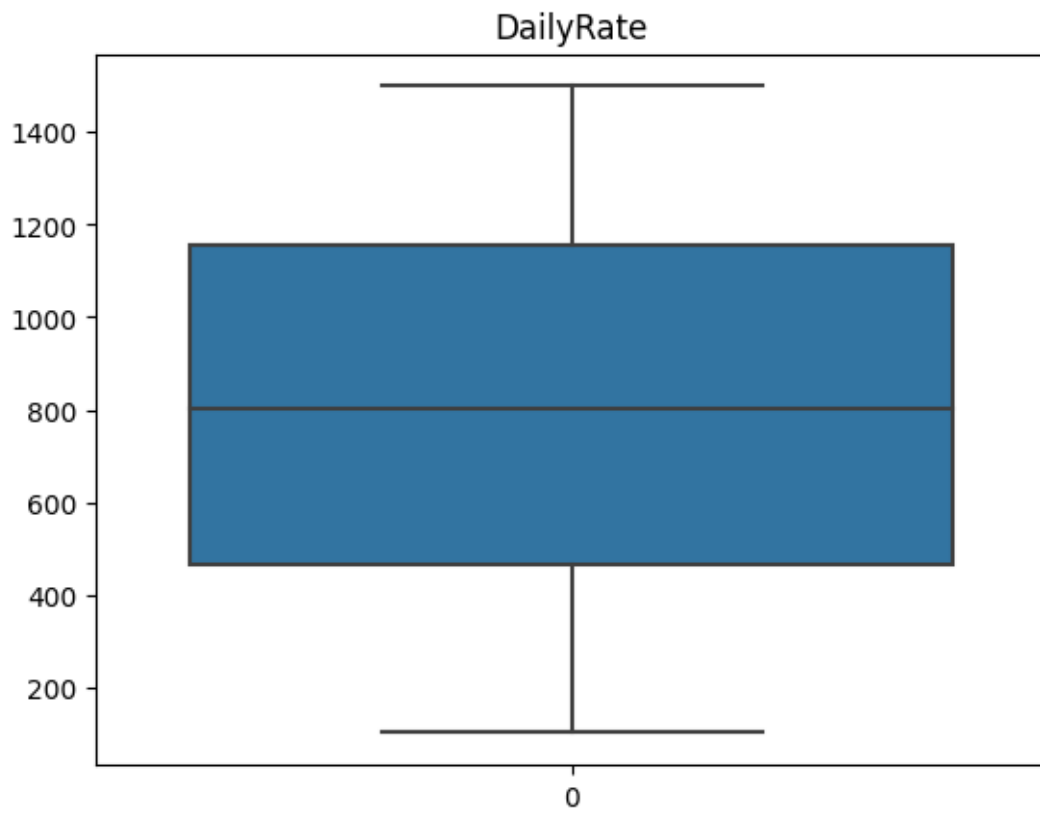
```
for i in l:
    q3=df[i].quantile(0.75)
    q1=df[i].quantile(0.25)
    IQR=q3-q1
    upper_limit=q3+(1.5*IQR)
    lower_limit=q1-(1.5*IQR)
    df[i]=np.where(df[i]>upper_limit,df[i].median(),df[i])
```

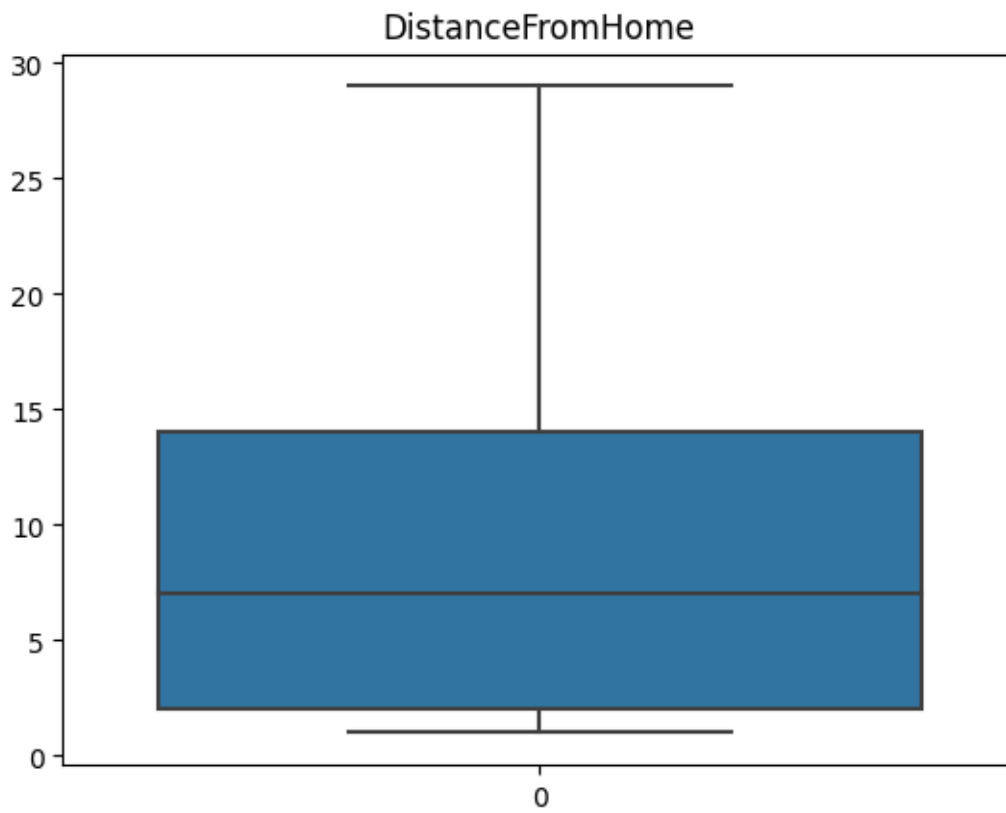
```
[19]: for i in df.columns:
        if(df[i].dtype!=object):
            plt.figure()
            plt.title(i)
            sns.boxplot(df[i])
```

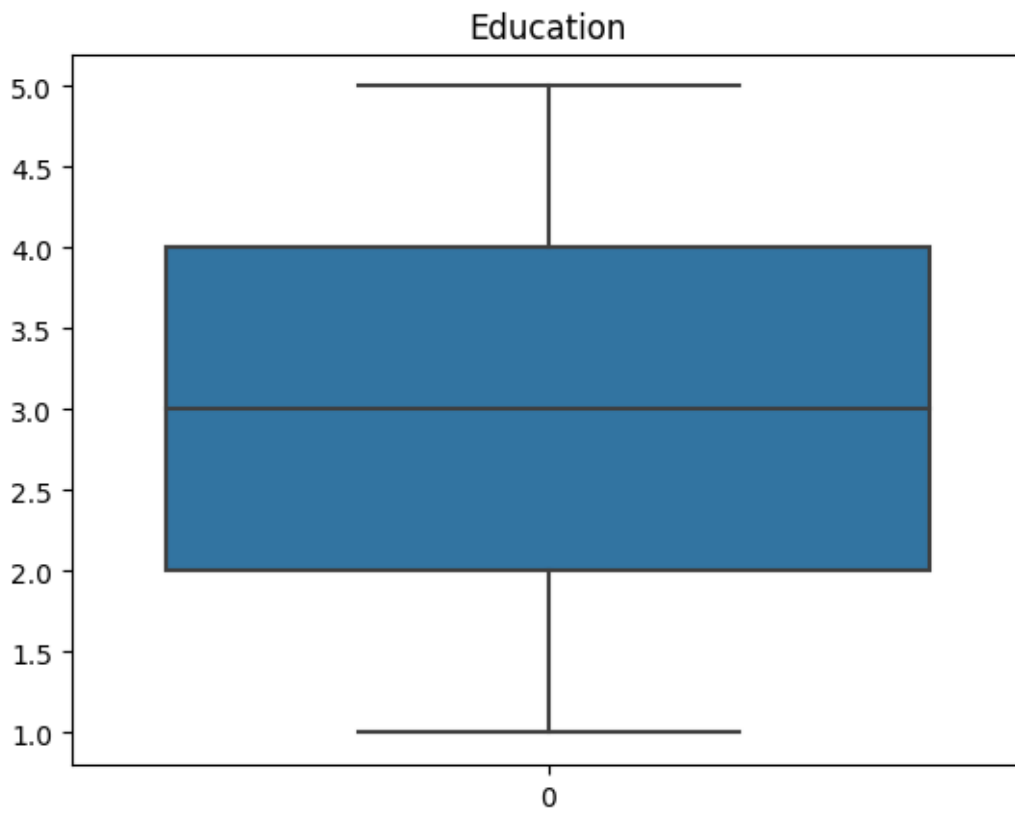
C:\Users\nitin\AppData\Local\Temp\ipykernel\_25332\2388892046.py:3:  
RuntimeWarning: More than 20 figures have been opened. Figures created through  
the pyplot interface (`matplotlib.pyplot.figure`) are retained until explicitly  
closed and may consume too much memory. (To control this warning, see the

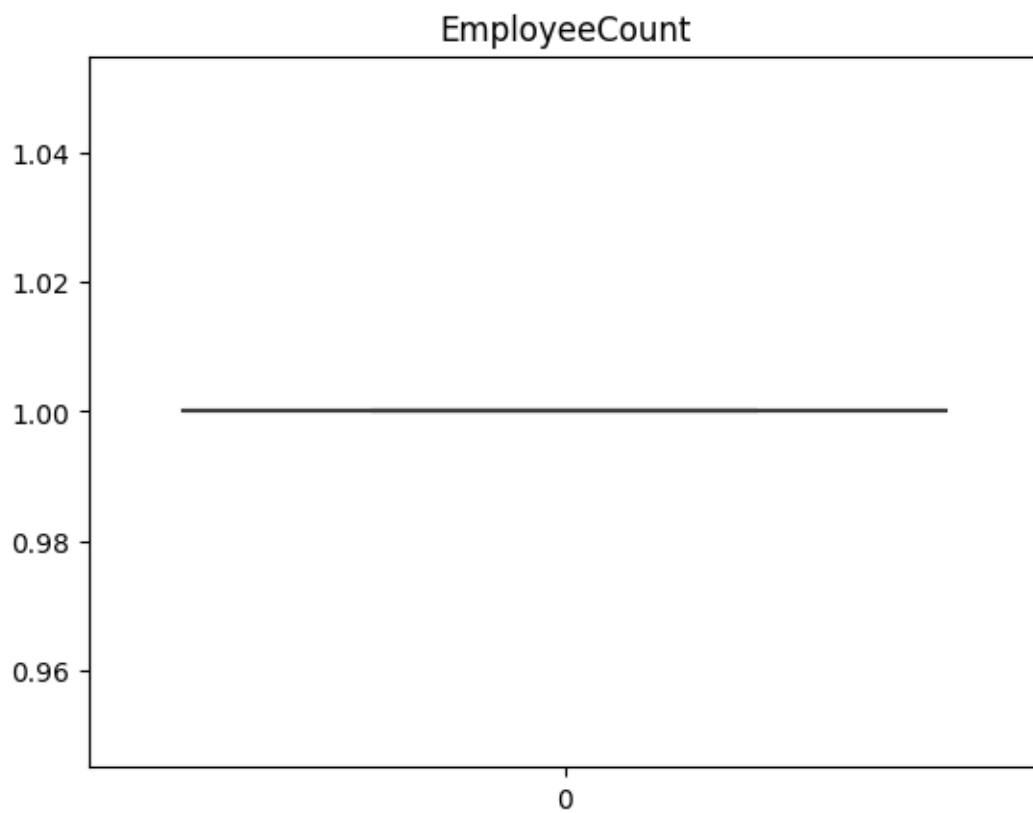
```
rcParam `figure.max_open_warning`). Consider using `matplotlib.pyplot.close()`.  
plt.figure()
```

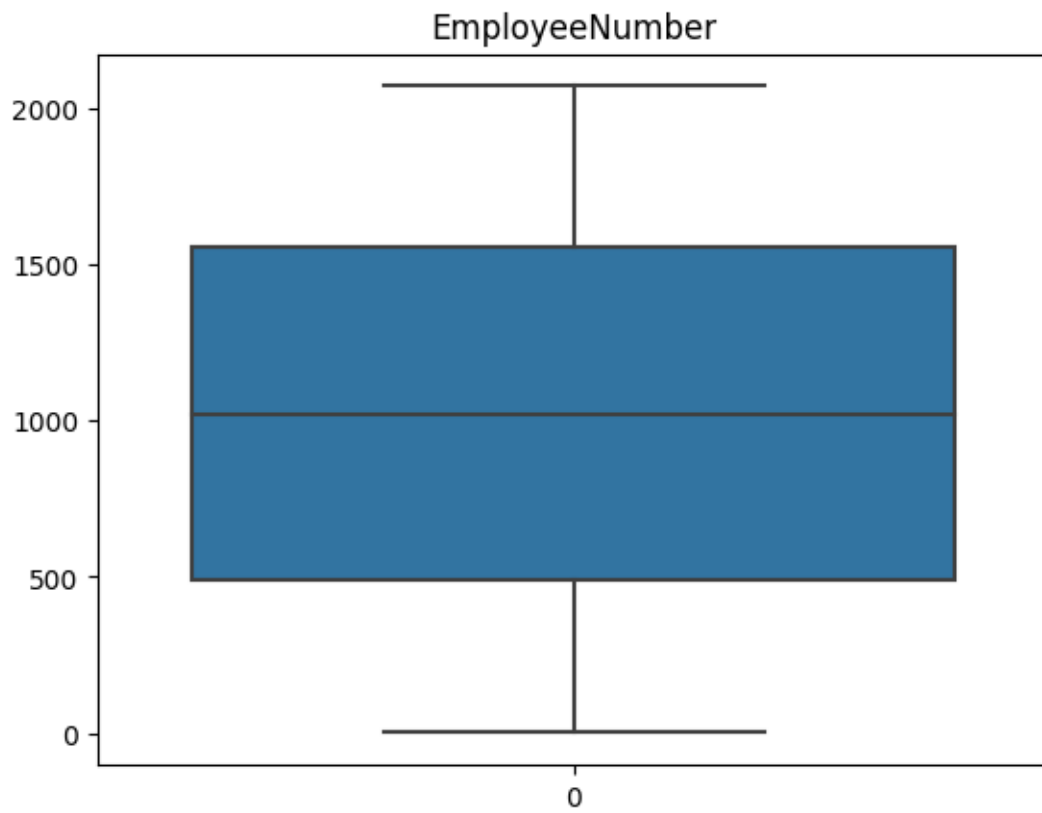




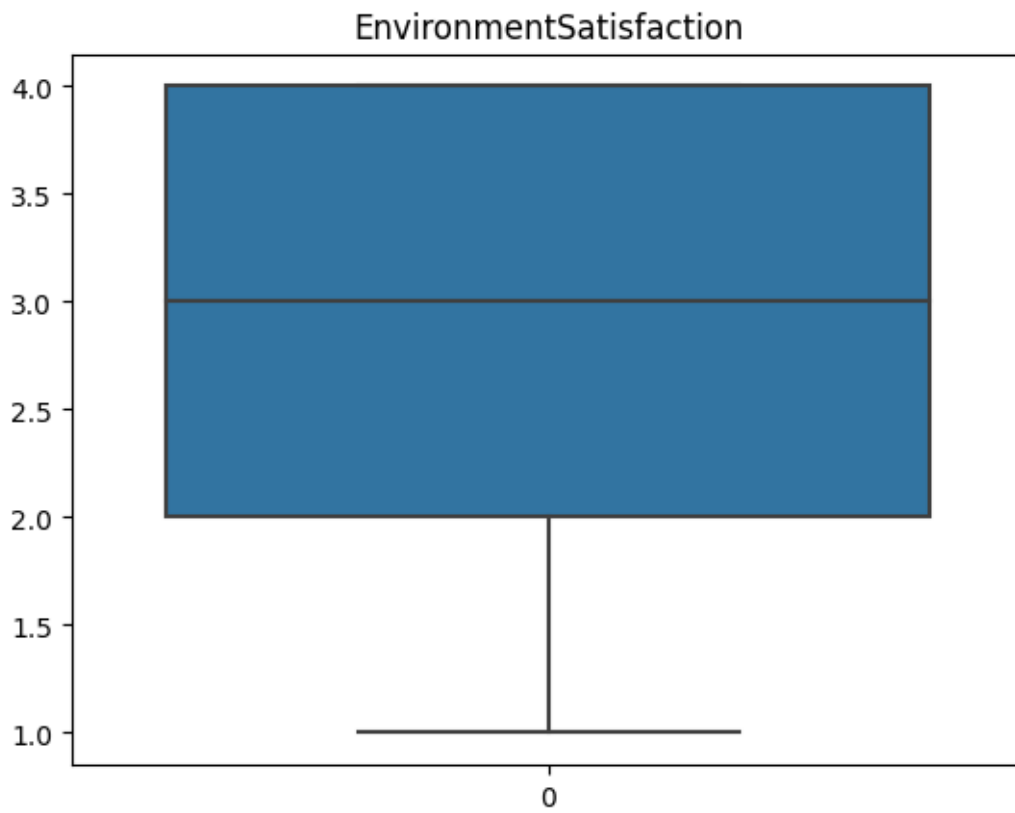


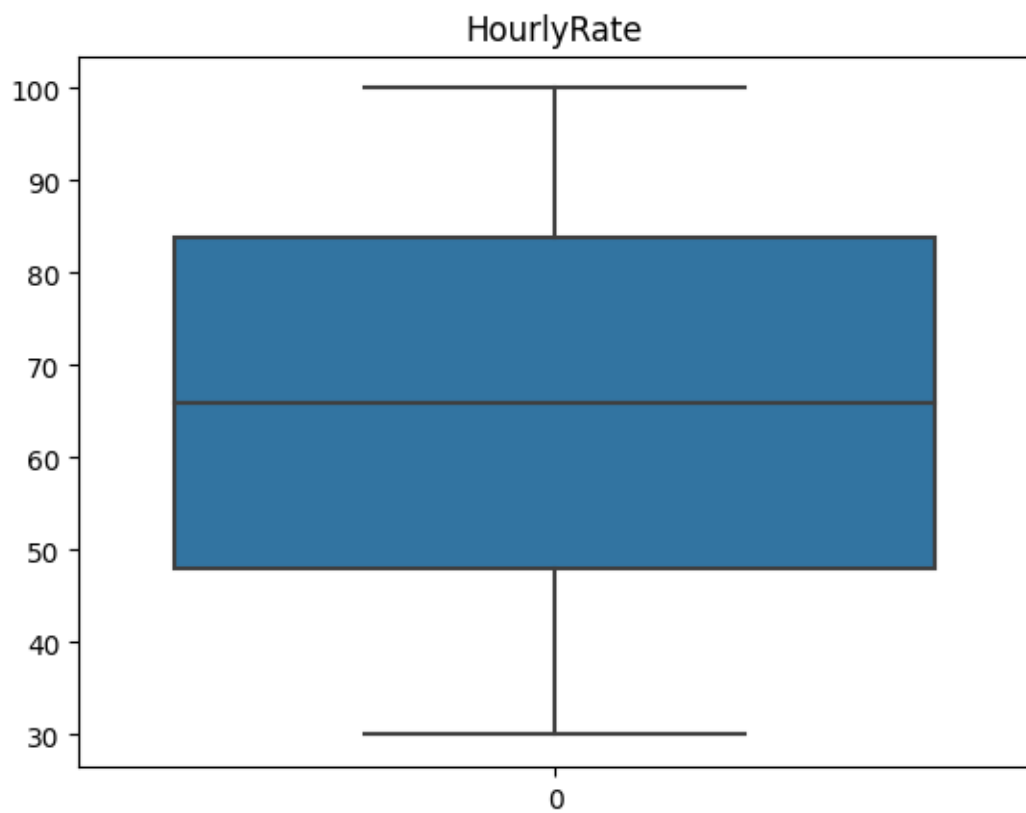


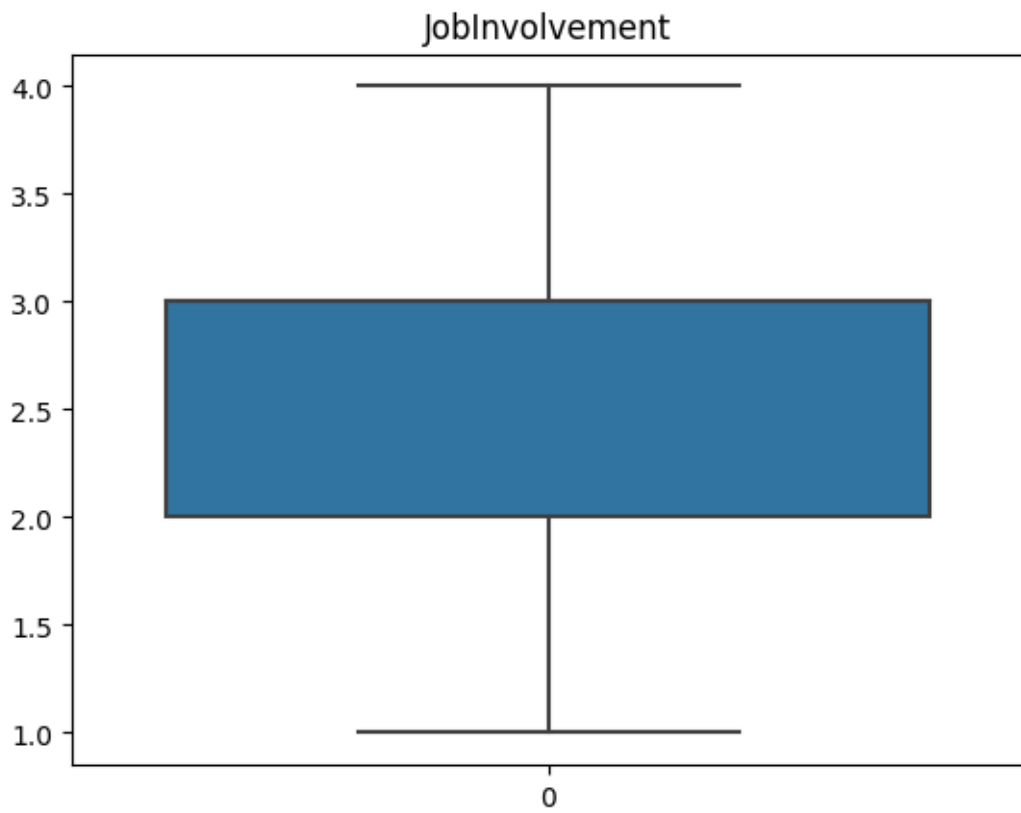


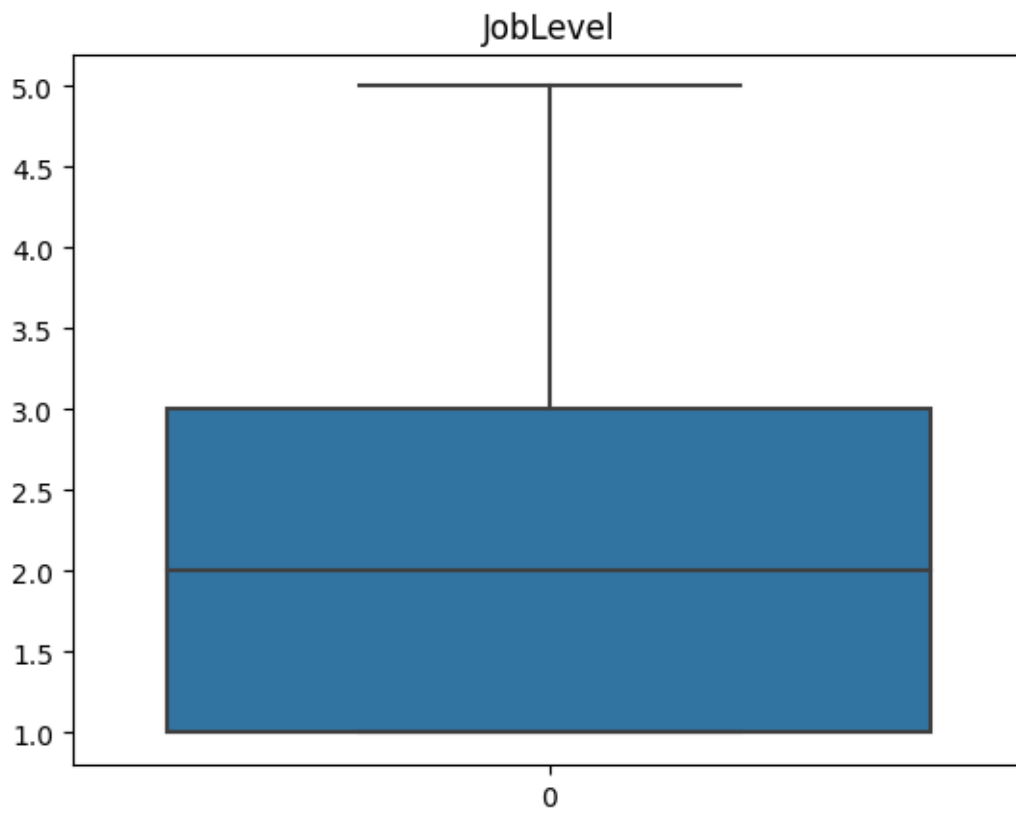


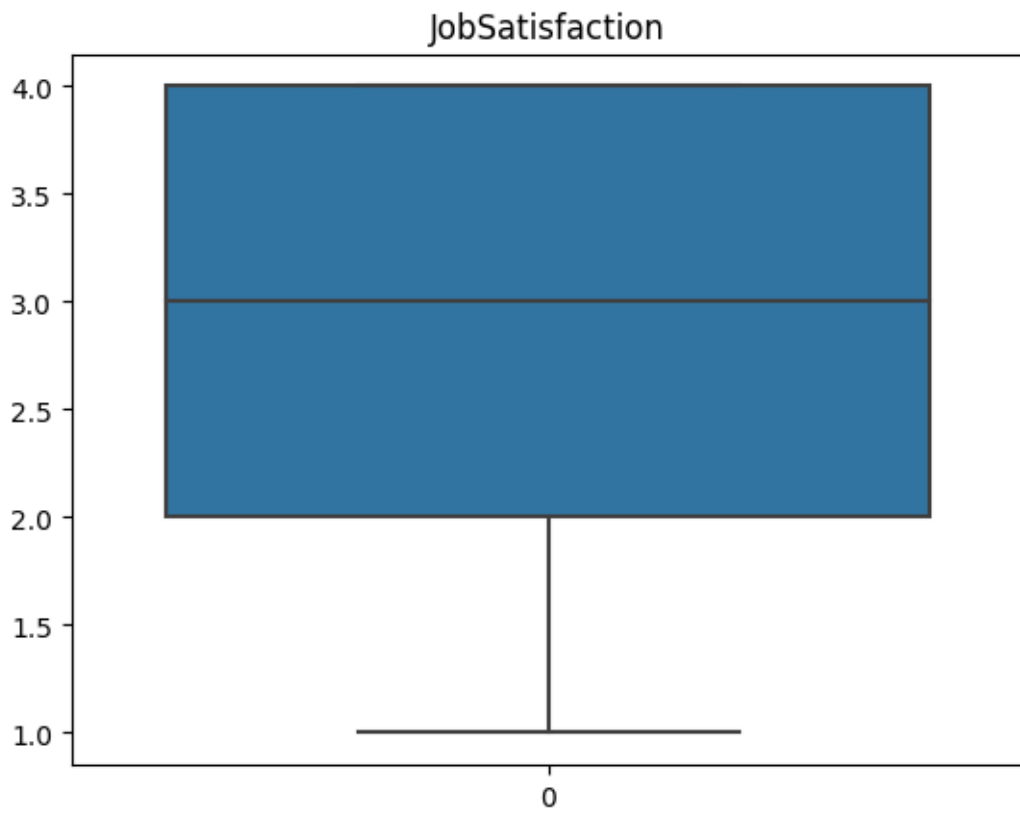


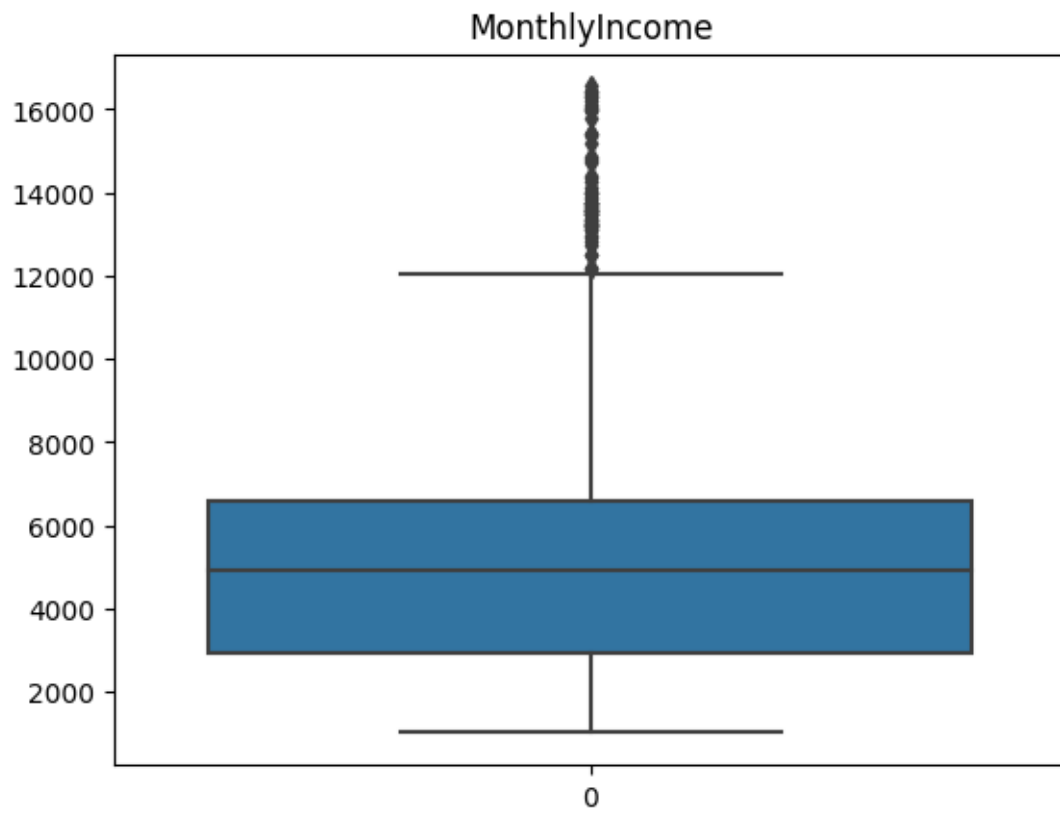


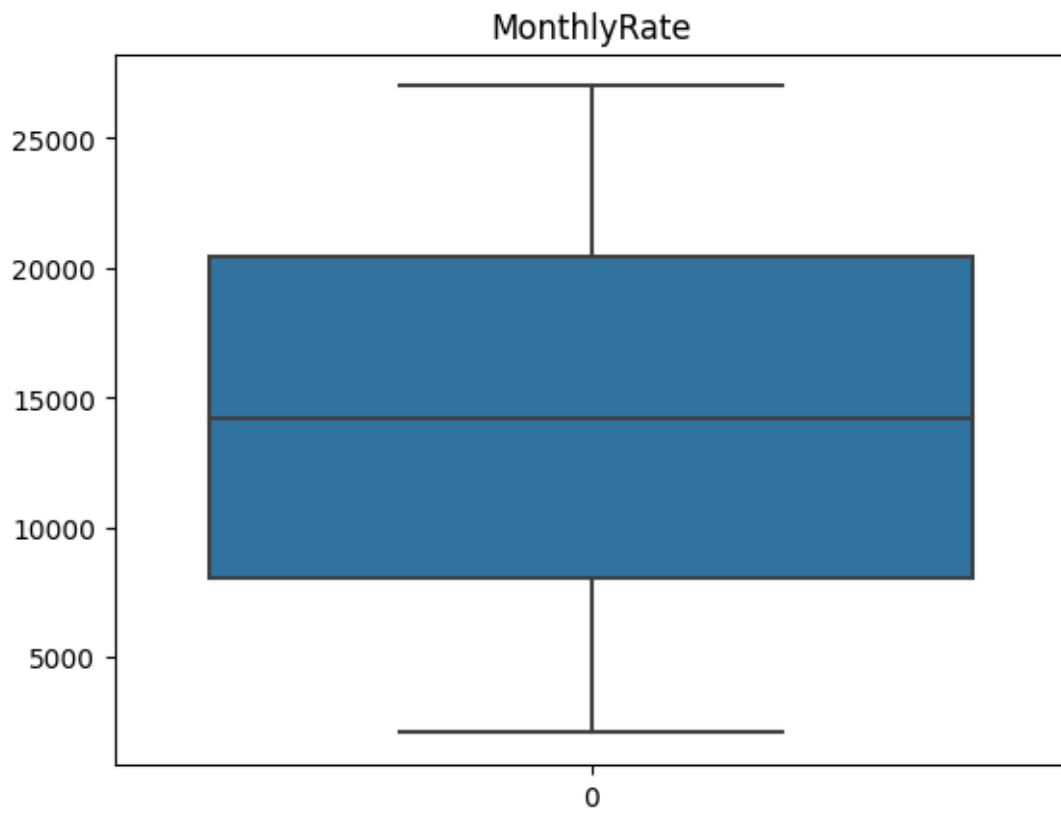


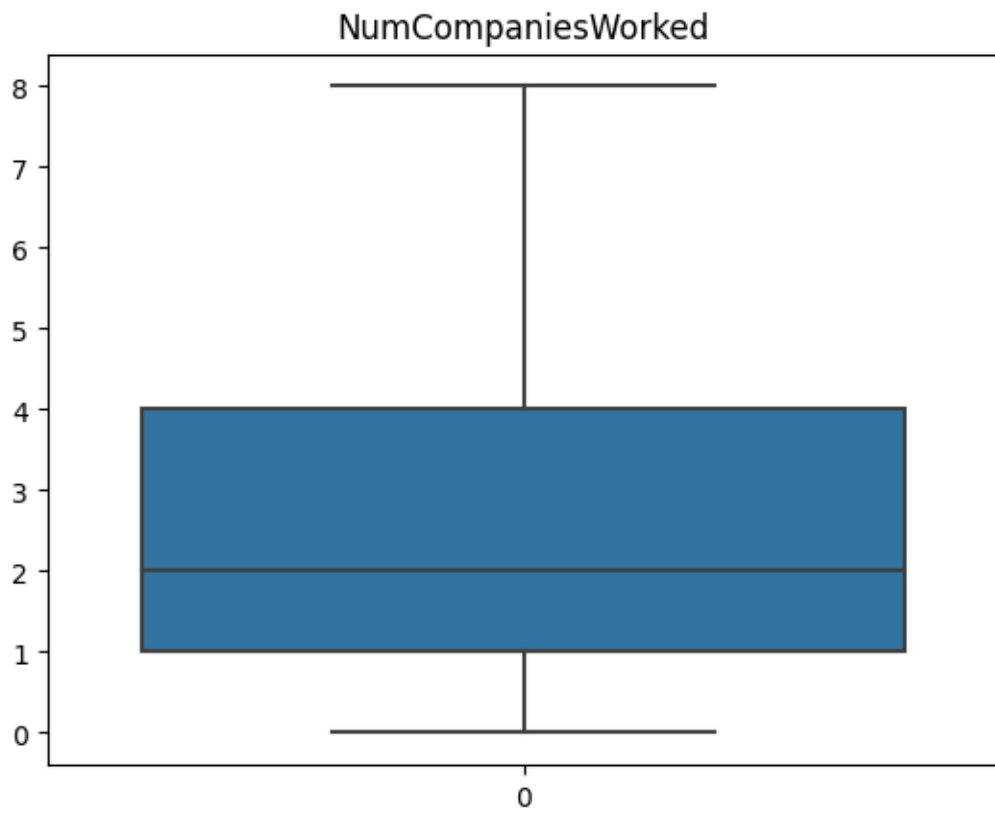




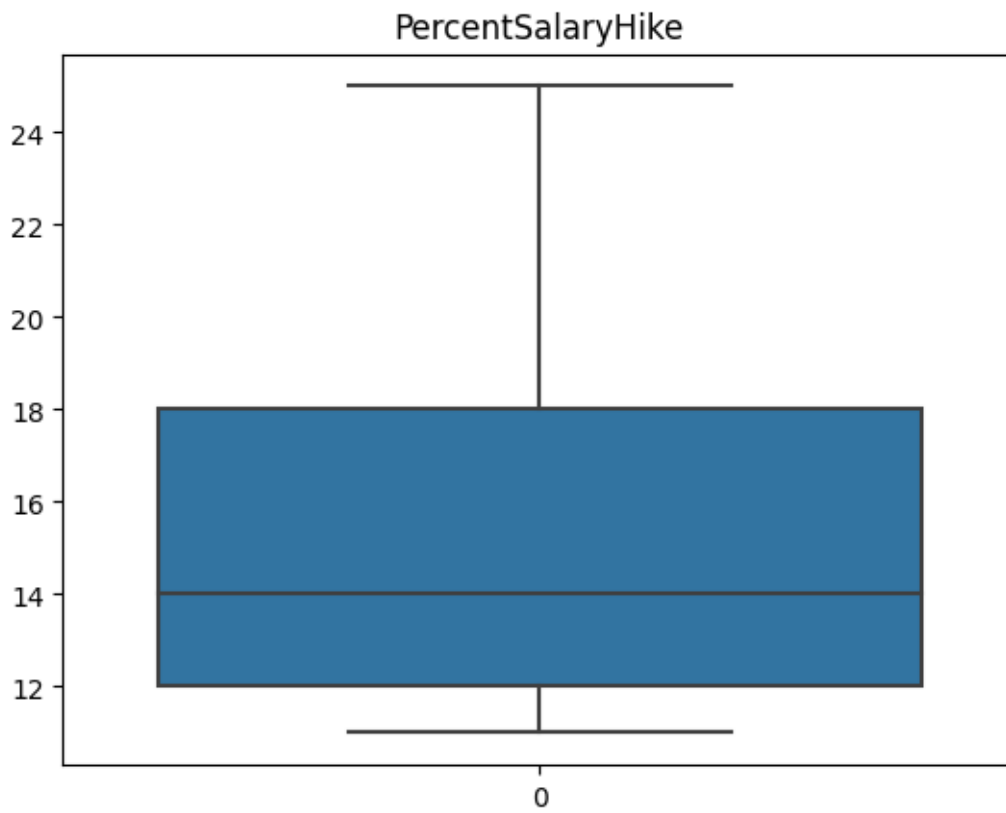


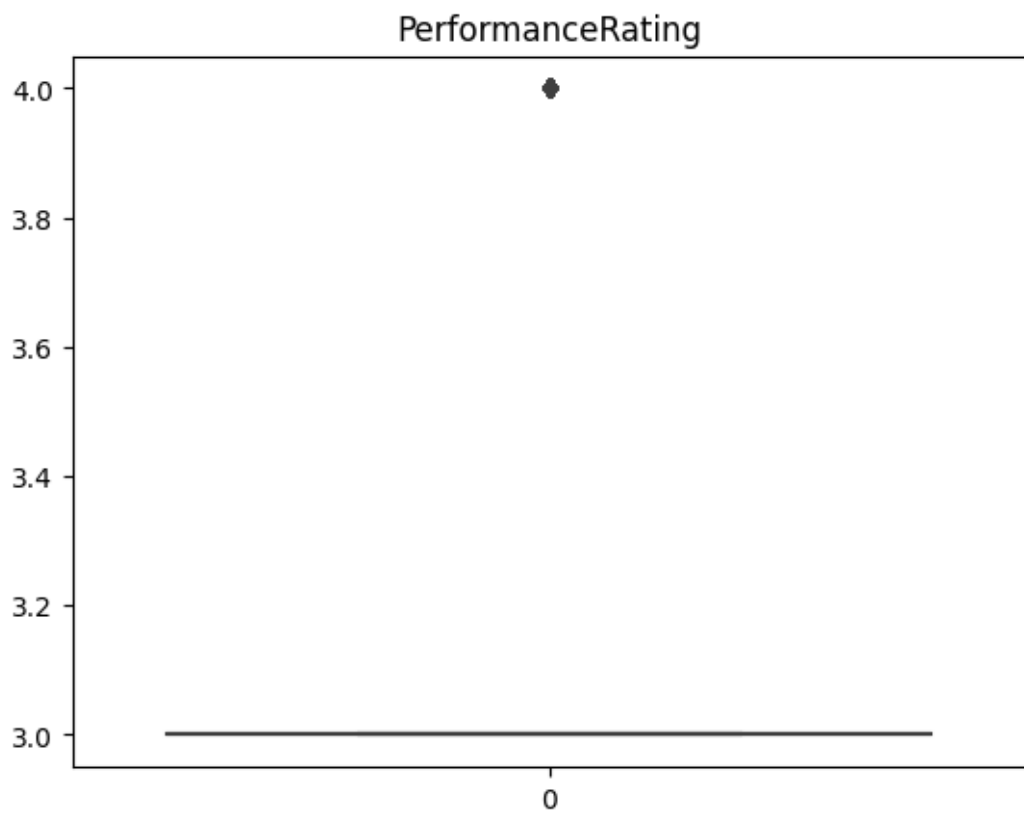


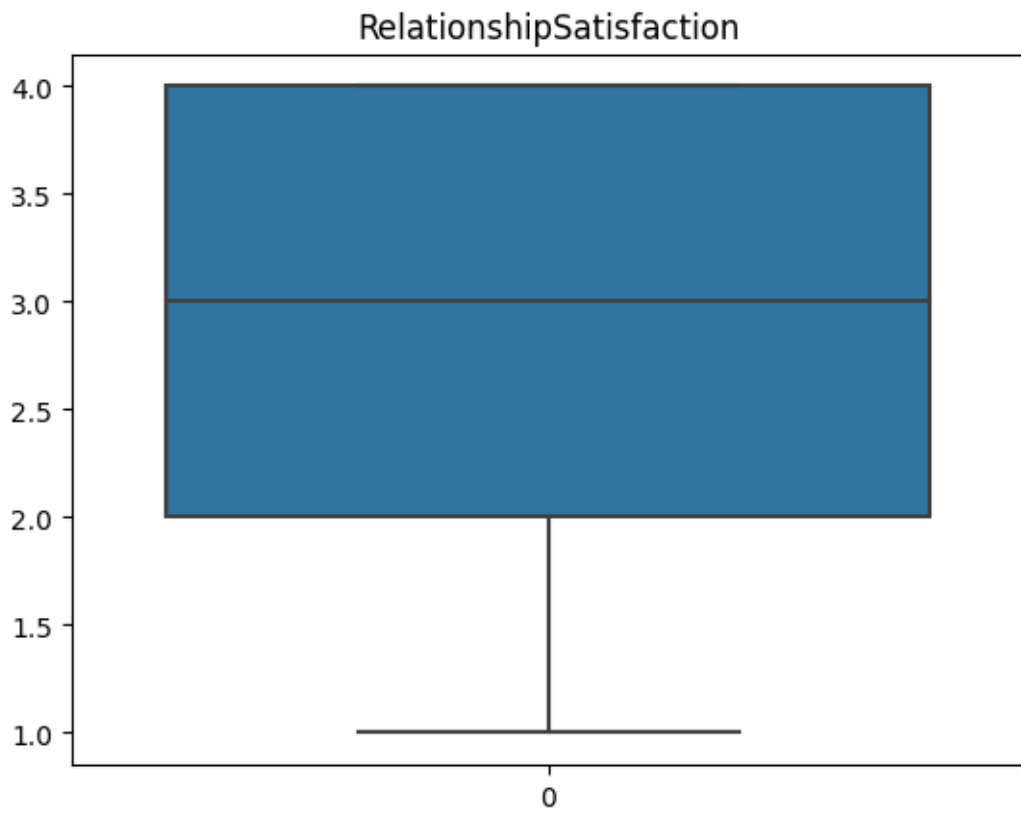


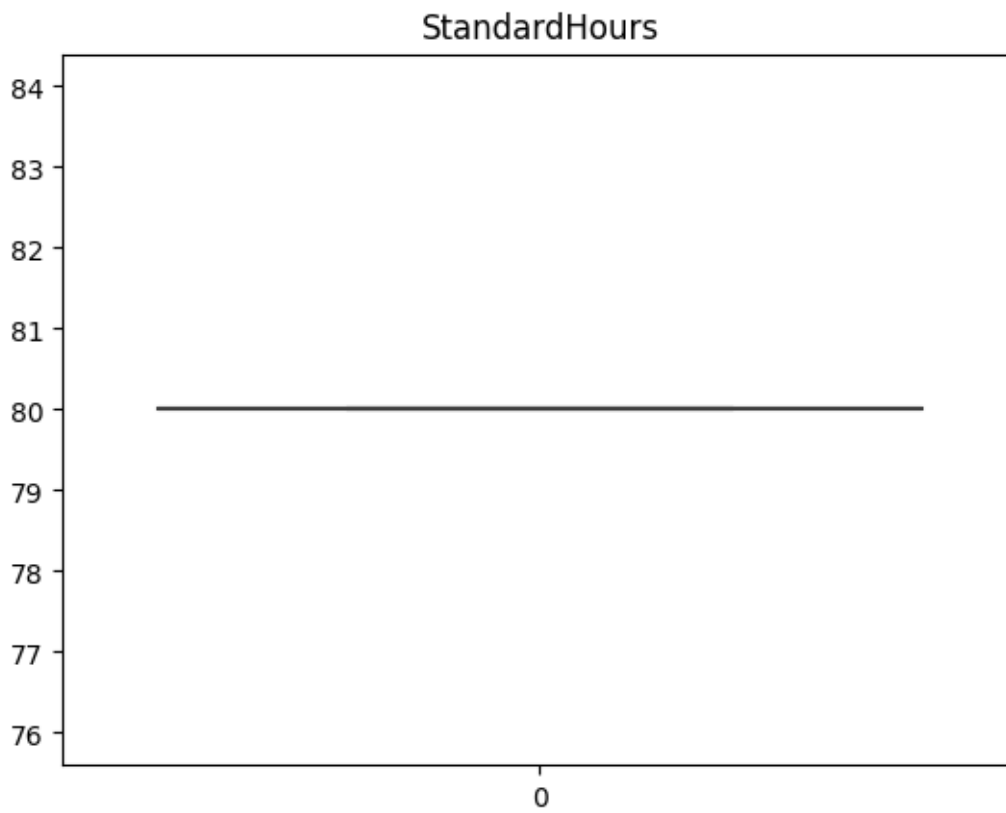


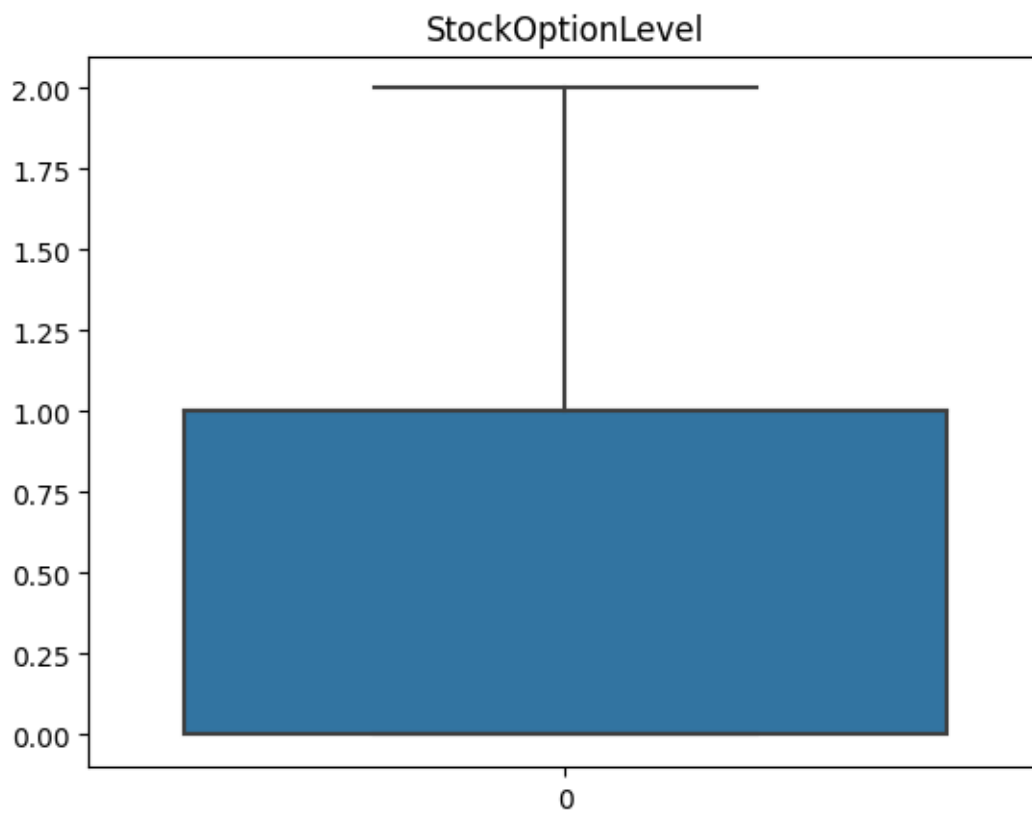


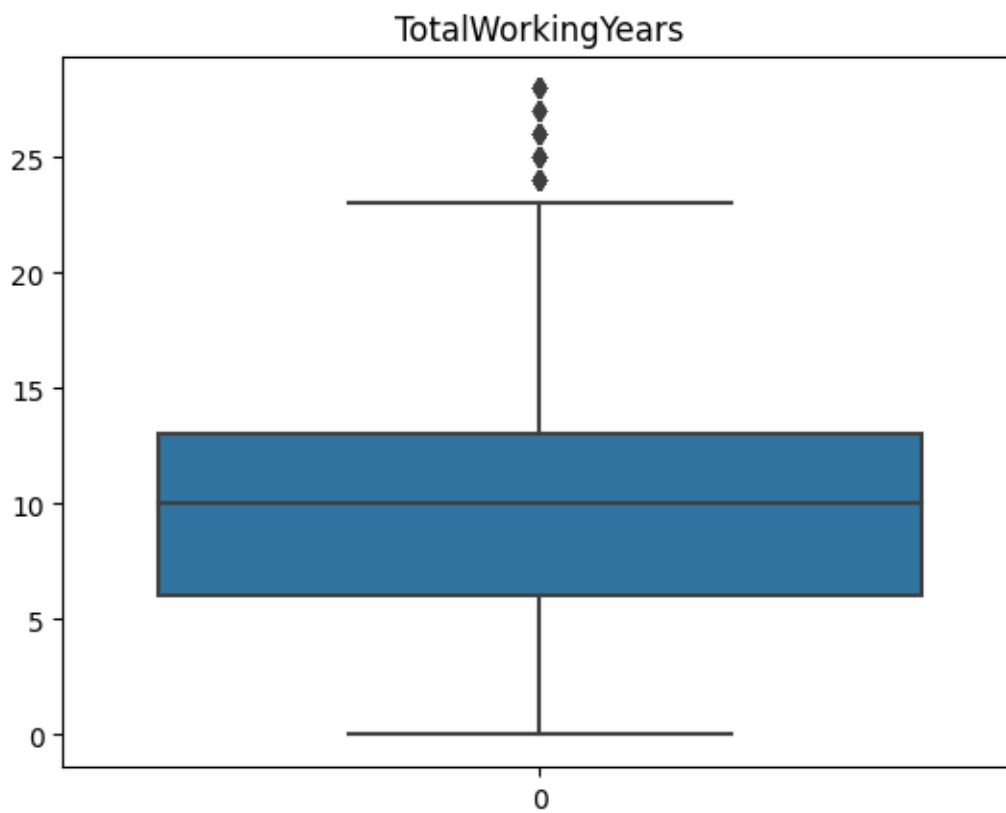


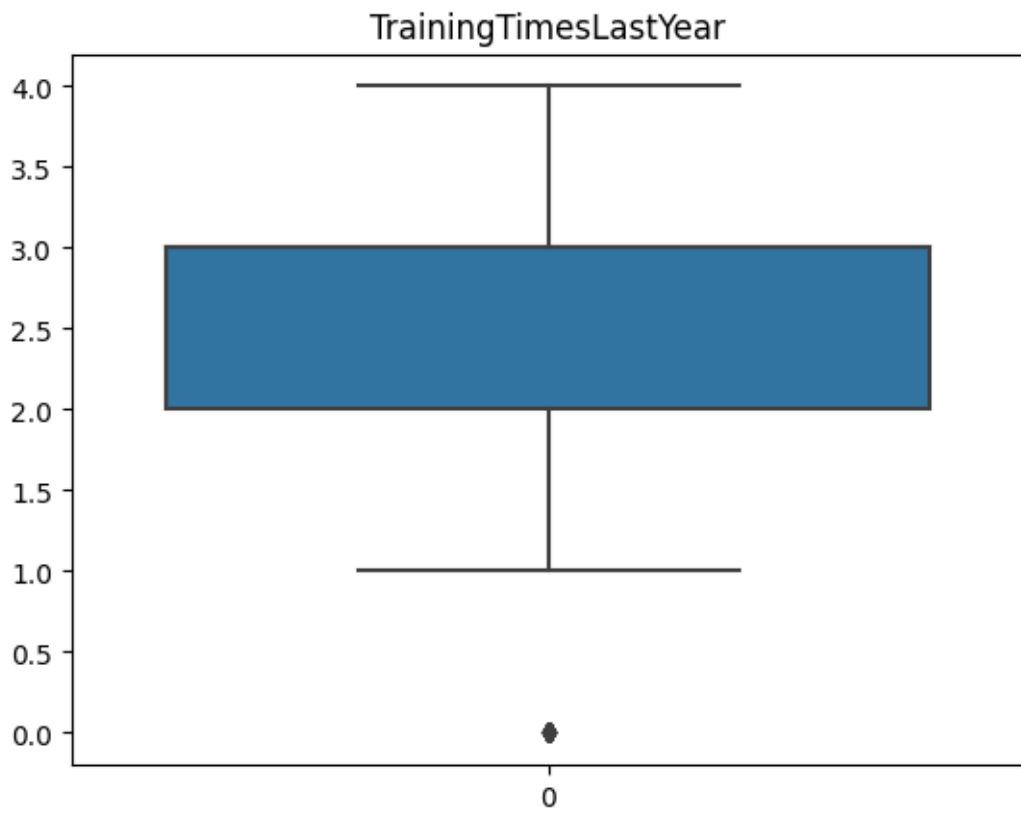


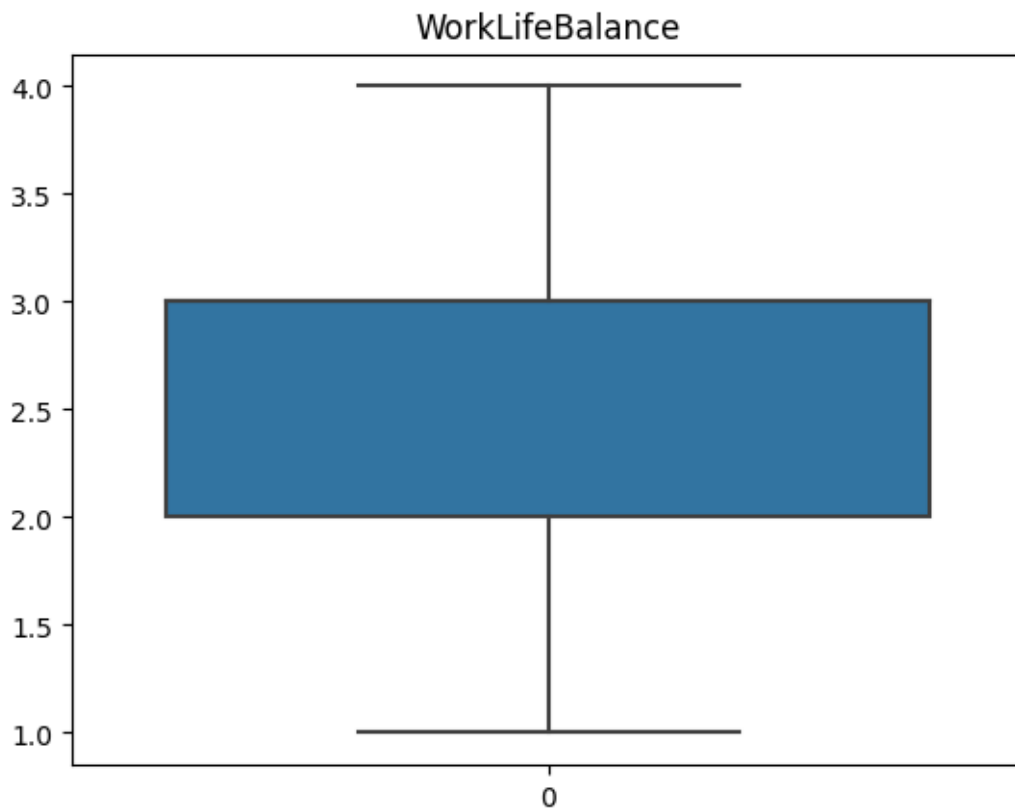




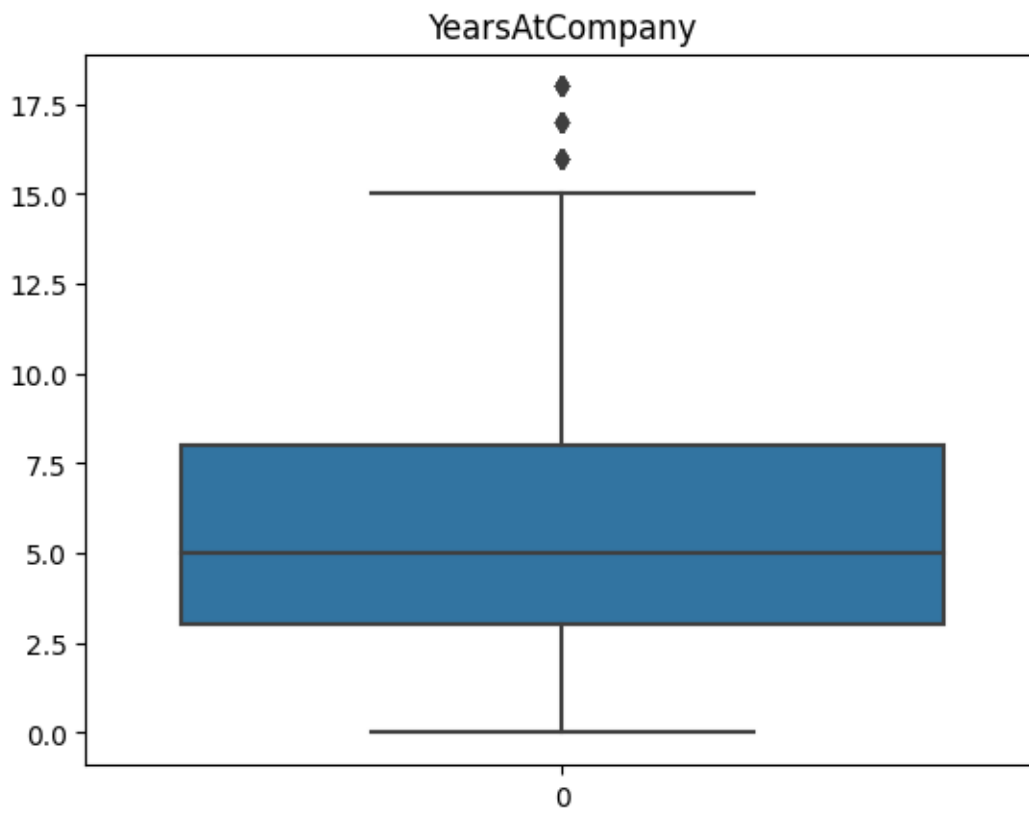


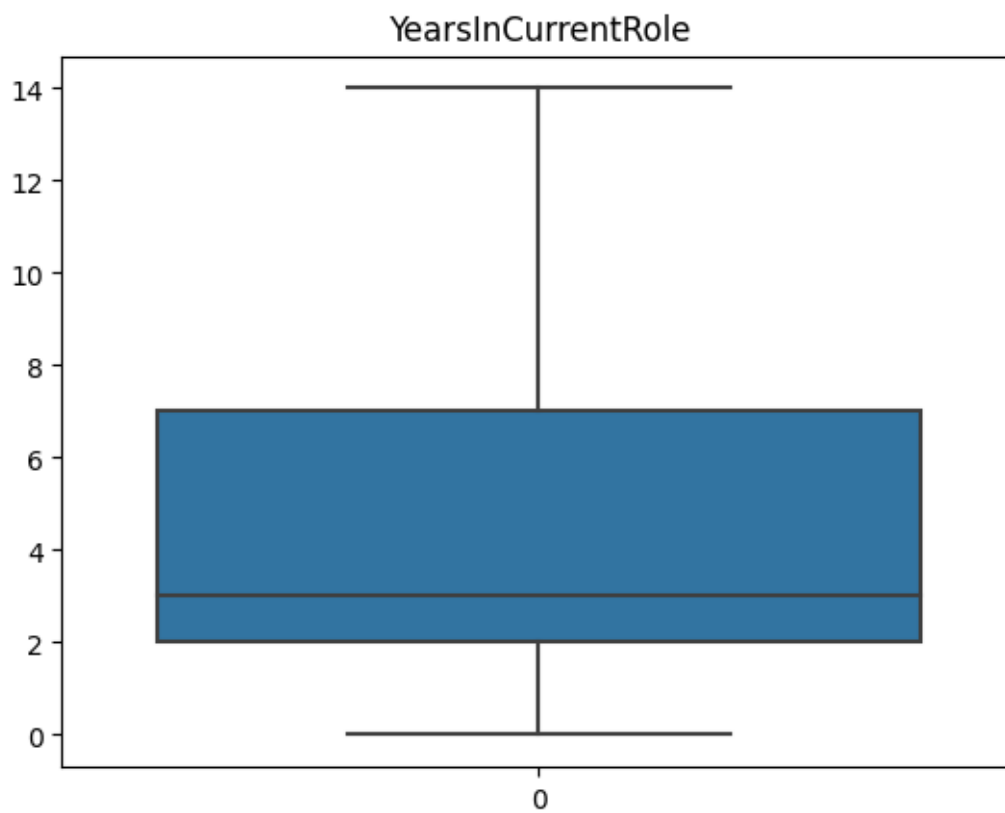


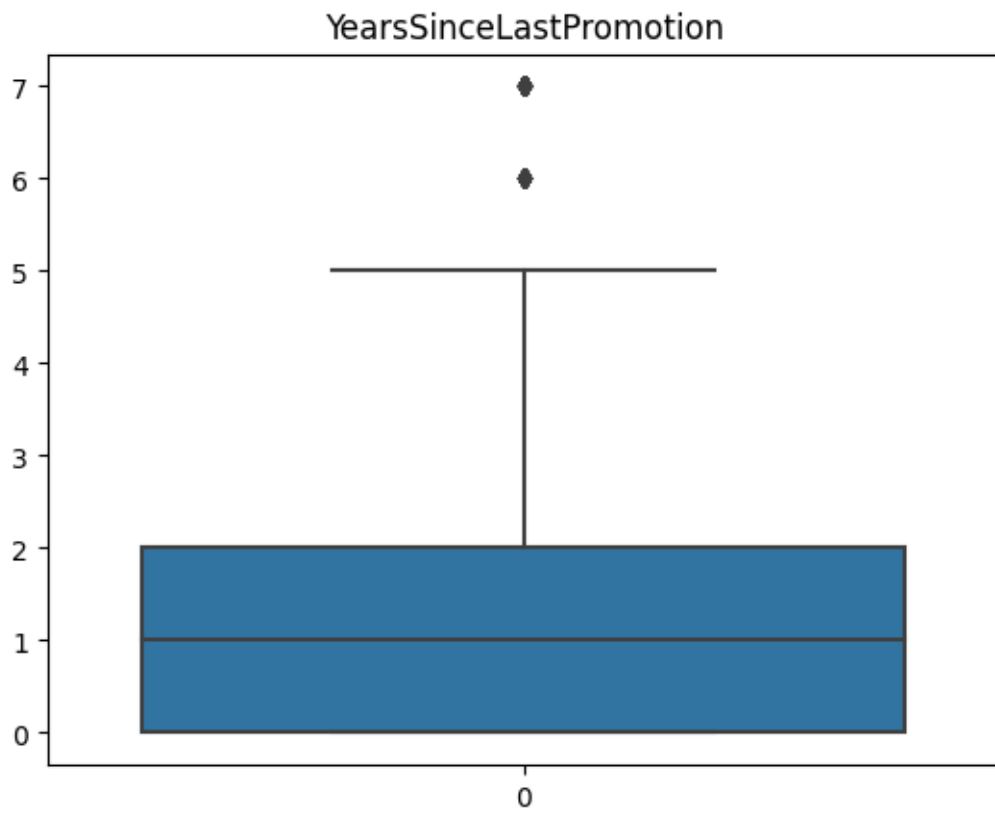


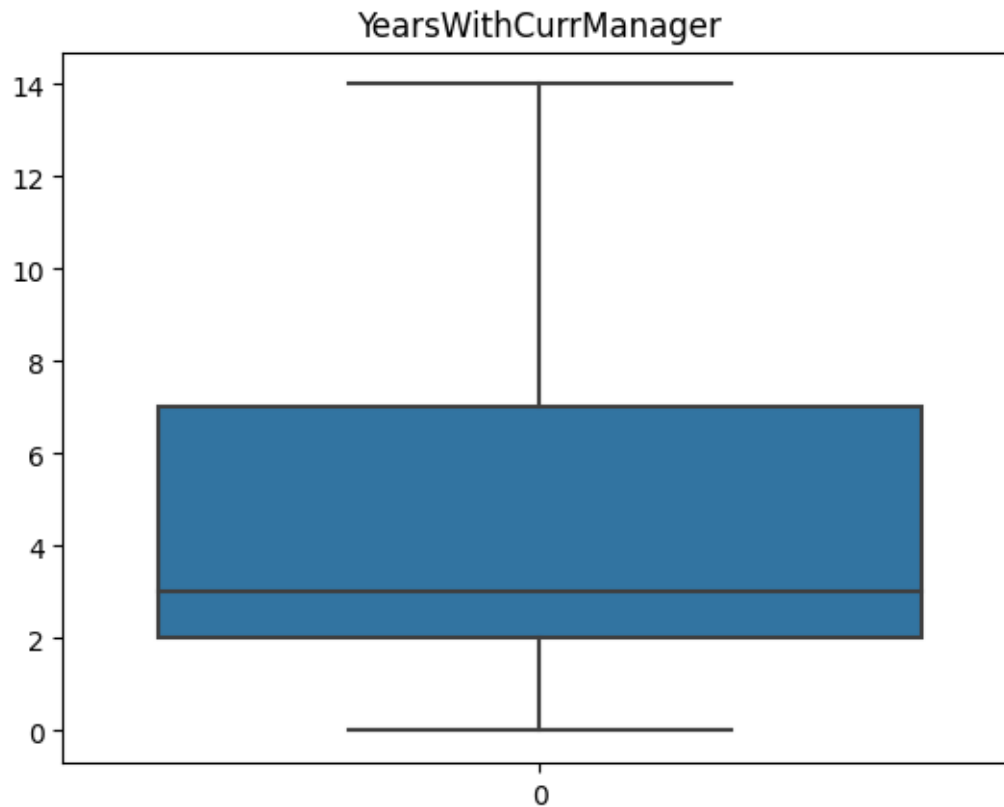












## 2.6 Splitting Dependent and Independent variables

```
[20]: df.  
      ↪drop(columns=['EmployeeCount', 'StandardHours', 'EmployeeNumber', 'Over18'], axis=1, inplace=True)
```

```
[21]: df.shape
```

```
[21]: (1470, 31)
```

```
[22]: df.head(5)
```

```
[22]:
```

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

	DistanceFromHome	Education	EducationField	EnvironmentSatisfaction
0	1	2	Life Sciences	2 \

1	8	1	Life Sciences	3
2	2	2	Other	4
3	3	4	Life Sciences	4
4	2	1	Medical	1

	Gender	...	PerformanceRating	RelationshipSatisfaction	StockOptionLevel
0	Female	...	3	1	0.0 \
1	Male	...	4	4	1.0
2	Male	...	3	2	0.0
3	Female	...	3	3	0.0
4	Male	...	3	4	1.0

	TotalWorkingYears	TrainingTimesLastYear	WorkLifeBalance	YearsAtCompany
0	8.0	0.0	1	6.0 \
1	10.0	3.0	3	10.0
2	7.0	3.0	3	0.0
3	8.0	3.0	3	8.0
4	6.0	3.0	3	2.0

	YearsInCurrentRole	YearsSinceLastPromotion	YearsWithCurrManager
0	4.0	0.0	5.0
1	7.0	1.0	7.0
2	0.0	0.0	0.0
3	7.0	3.0	0.0
4	2.0	2.0	2.0

[5 rows x 31 columns]

```
[23]: x=df.drop("Attrition",axis=1)
      y=df.iloc[:,1:2]
```

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

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

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

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

	StockOptionLevel	TotalWorkingYears	TrainingTimesLastYear	WorkLifeBalance	
0	0.0	8.0	0.0	1	\
1	1.0	10.0	3.0	3	
2	0.0	7.0	3.0	3	
3	0.0	8.0	3.0	3	
4	1.0	6.0	3.0	3	

	YearsAtCompany	YearsInCurrentRole	YearsSinceLastPromotion	
0	6.0	4.0	0.0	\
1	10.0	7.0	1.0	
2	0.0	0.0	0.0	
3	8.0	7.0	3.0	
4	2.0	2.0	2.0	

	YearsWithCurrManager
0	5.0
1	7.0
2	0.0
3	0.0
4	2.0

[5 rows x 30 columns]

```
[25]: y.head()
```

```
[25]: Attrition
0      Yes
1      No
2      Yes
3      No
4      No
```

```
[26]: y=np.squeeze(y)
y.head()
```

```
[26]: 0      Yes
1      No
2      Yes
3      No
4      No
```

Name: Attrition, dtype: object

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

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

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

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

## 2.7 Perform Encoding

```
[29]: from sklearn.preprocessing import LabelEncoder  
le=LabelEncoder()  
y_encoded=pd.Series(le.fit_transform(y))
```

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

```
<class 'pandas.core.frame.DataFrame'>  
RangeIndex: 1470 entries, 0 to 1469  
Data columns (total 30 columns):  
#   Column                                Non-Null Count  Dtype  
---  -  
0   Age                                   1470 non-null   int64  
1   BusinessTravel                       1470 non-null   object  
2   DailyRate                           1470 non-null   int64  
3   Department                          1470 non-null   object  
4   DistanceFromHome                    1470 non-null   int64  
5   Education                           1470 non-null   int64  
6   EducationField                      1470 non-null   object  
7   EnvironmentSatisfaction             1470 non-null   int64  
8   Gender                              1470 non-null   object  
9   HourlyRate                          1470 non-null   int64  
10  JobInvolvement                      1470 non-null   int64  
11  JobLevel                            1470 non-null   int64  
12  JobRole                             1470 non-null   object  
13  JobSatisfaction                     1470 non-null   int64  
14  MaritalStatus                      1470 non-null   object  
15  MonthlyIncome                      1470 non-null   float64  
16  MonthlyRate                        1470 non-null   int64  
17  NumCompaniesWorked                 1470 non-null   float64  
18  OverTime                           1470 non-null   object  
19  PercentSalaryHike                  1470 non-null   int64  
20  PerformanceRating                  1470 non-null   int64  
21  RelationshipSatisfaction            1470 non-null   int64  
22  StockOptionLevel                   1470 non-null   float64  
23  TotalWorkingYears                  1470 non-null   float64
```

```

24 TrainingTimesLastYear      1470 non-null    float64
25 WorkLifeBalance            1470 non-null    int64
26 YearsAtCompany             1470 non-null    float64
27 YearsInCurrentRole         1470 non-null    float64
28 YearsSinceLastPromotion    1470 non-null    float64
29 YearsWithCurrManager       1470 non-null    float64
dtypes: float64(9), int64(14), object(7)
memory usage: 344.7+ KB

```

```

[31]: Business_Travel1=pd.get_dummies(df["BusinessTravel"],drop_first=True).
      ↪astype(int)
      Department1=pd.get_dummies(df["Department"],drop_first=True).astype(int)
      EducationField1=pd.get_dummies(df["EducationField"],drop_first=True).astype(int)
      Gender1=pd.get_dummies(df["Gender"],drop_first=True).astype(int)
      JobRole1=pd.get_dummies(df["JobRole"],drop_first=True).astype(int)
      MaritalStatus1=pd.get_dummies(df["MaritalStatus"],drop_first=True).astype(int)
      OverTime1=pd.get_dummies(df["OverTime"],drop_first=True).astype(int)

```

```

[32]: x=pd.concat([x,Business_Travel1],axis=1)
      x=pd.concat([x,Department1],axis=1)
      x=pd.concat([x,EducationField1],axis=1)
      x=pd.concat([x,Gender1],axis=1)
      x=pd.concat([x,JobRole1],axis=1)
      x=pd.concat([x,MaritalStatus1],axis=1)
      x=pd.concat([x,OverTime1],axis=1)

```

```

[33]: x.drop(['BusinessTravel', 'Department', 'EducationField','Gender', 'JobRole',
      ↪'MaritalStatus', 'OverTime'],axis = 1, inplace = True)

```

```

[34]: x.head()

```

```

[34]:   Age  DailyRate  DistanceFromHome  Education  EnvironmentSatisfaction
0    41        1102                 1          2                      2 \
1    49         279                 8          1                      3
2    37        1373                 2          2                      4
3    33        1392                 3          4                      4
4    27         591                 2          1                      1

   HourlyRate  JobInvolvement  JobLevel  JobSatisfaction  MonthlyIncome  ...
0          94              3          2              4        5993.0  ... \
1          61              2          2              2        5130.0  ...
2          92              2          1              3        2090.0  ...
3          56              3          1              3        2909.0  ...
4          40              3          1              2        3468.0  ...

   Laboratory Technician  Manager  Manufacturing Director  Research Director
0                      0          0                      0                  0 \

```



```

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

    Research Scientist  Sales Executive  Sales Representative  Married  Single \
0          0          1          0          0          1
1          1          0          0          1          0
2          0          0          0          0          1
3          1          0          0          1          0
4          0          0          0          1          0

    Yes
0     1
1     0
2     1
3     1
4     0

[5 rows x 44 columns]

```

```
[35]: y_encoded.head()
```

```

[35]: 0     1
      1     0
      2     1
      3     0
      4     0
      dtype: int32

```

## 2.8 Feature Scaling.

```
[36]: from sklearn.preprocessing import StandardScaler
      ss=StandardScaler()
      x_scaled=pd.DataFrame(ss.fit_transform(x),columns=x.columns)
```

```
[37]: x_scaled.head()
```

```

[37]:      Age  DailyRate  DistanceFromHome  Education  EnvironmentSatisfaction \
0  0.446350  0.742527      -1.010909  -0.891688      -0.660531
1  1.322365 -1.297775      -0.147150  -1.868426       0.254625
2  0.008343  1.414363      -0.887515  -0.891688       1.169781
3 -0.429664  1.461466      -0.764121   1.061787       1.169781
4 -1.086676 -0.524295      -0.887515  -1.868426      -1.575686

      HourlyRate  JobInvolvement  JobLevel  JobSatisfaction  MonthlyIncome  ... \
0    1.383138      0.379672 -0.057788      1.153254      0.167809  ...

```

1	-0.240677	-1.026167	-0.057788	-0.660853	-0.102624	...
2	1.284725	-1.026167	-0.961486	0.246200	-1.055253	...
3	-0.486709	0.379672	-0.961486	0.246200	-0.798607	...
4	-1.274014	0.379672	-0.961486	-0.660853	-0.623436	...

	Laboratory Technician	Manager	Manufacturing Director	Research Director	
0	-0.462464	-0.273059	-0.330808	-0.239904	\
1	-0.462464	-0.273059	-0.330808	-0.239904	
2	2.162331	-0.273059	-0.330808	-0.239904	
3	-0.462464	-0.273059	-0.330808	-0.239904	
4	2.162331	-0.273059	-0.330808	-0.239904	

	Research Scientist	Sales Executive	Sales Representative	Married	
0	-0.497873	1.873287	-0.244625	-0.918921	\
1	2.008543	-0.533821	-0.244625	1.088232	
2	-0.497873	-0.533821	-0.244625	-0.918921	
3	2.008543	-0.533821	-0.244625	1.088232	
4	-0.497873	-0.533821	-0.244625	1.088232	

	Single	Yes
0	1.458650	1.591746
1	-0.685565	-0.628241
2	1.458650	1.591746
3	-0.685565	1.591746
4	-0.685565	-0.628241

[5 rows x 44 columns]

## 2.9 Splitting Data into Train and Test

```
[38]: from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x_scaled,y_encoded,test_size=0.
↪2,random_state=0)
```

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

```
(1176, 44) (294, 44) (1176,) (294,)
```

## 3 Preprocessing Done

### 4 ->Model Building - Logistic regression

#### 4.1 Import the model building Libraries

```
[40]: from sklearn.linear_model import LogisticRegression
```

## 4.2 Initializing the model

```
[41]: lr=LogisticRegression()
```

## 4.3 Training the Model

```
[42]: lr.fit(x_train,y_train)
```

```
[42]: LogisticRegression()
```

## 4.4 Testing the Model

```
[43]: y_pred=lr.predict(x_test)
```

```
[44]: pd.DataFrame({"Actual_values":y_test,"Predicted_values":y_pred})
```

```
[44]:
```

	Actual_values	Predicted_values
442	0	0
1091	0	0
981	1	1
785	0	0
1332	1	1
...	...	...
1439	0	0
481	0	0
124	1	1
198	0	0
1229	0	0

```
[294 rows x 2 columns]
```

## 4.5 Evaluation of Model & Performance metrics

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

```
[46]: print("Accuracy of model :",accuracy_score(y_test,y_pred))
```

```
Accuracy of model : 0.8843537414965986
```

```
[47]: confusion_matrix(y_test,y_pred)
```

```
[47]: array([[240,   5],  
        [ 29,  20]], dtype=int64)
```

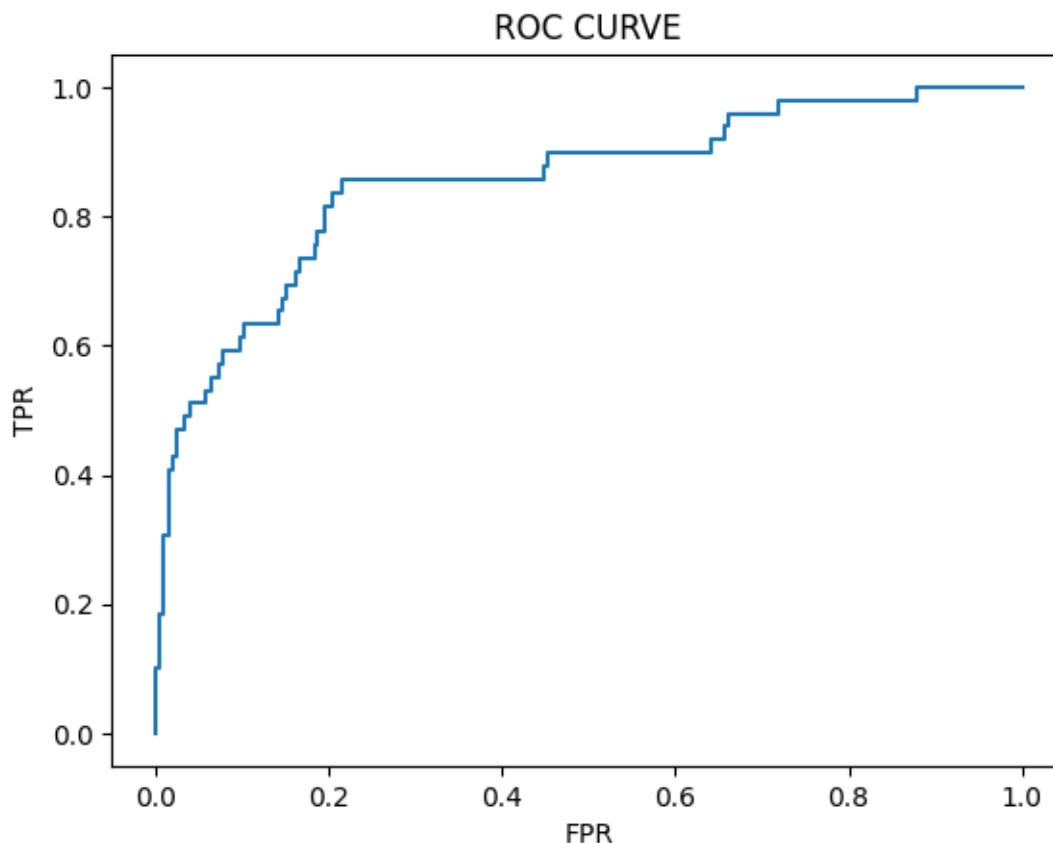
```
[48]: pd.crosstab(y_test,y_pred)
```

```
[48]: col_0    0    1
      row_0
      0    240    5
      1     29   20
```

```
[49]: print(classification_report(y_test,y_pred))
```

	precision	recall	f1-score	support
0	0.89	0.98	0.93	245
1	0.80	0.41	0.54	49
accuracy			0.88	294
macro avg	0.85	0.69	0.74	294
weighted avg	0.88	0.88	0.87	294

```
[50]: #ROC-AUC Curve
probability=lr.predict_proba(x_test)[: ,1]
fpr,tpr,threshholds = roc_curve(y_test,probability)
plt.plot(fpr,tpr)
plt.xlabel('FPR')
plt.ylabel('TPR')
plt.title('ROC CURVE')
plt.show()
```



```
[51]: threshsholds
```

```
[51]: array([1.92926459e+00, 9.29264586e-01, 8.24540258e-01, 8.19719126e-01,
7.05864328e-01, 7.01705053e-01, 6.17766349e-01, 5.95134749e-01,
5.60745910e-01, 5.26613211e-01, 4.82199936e-01, 4.81745804e-01,
4.74771100e-01, 4.36986830e-01, 4.33933781e-01, 4.22776305e-01,
4.07031472e-01, 3.81195557e-01, 3.72640404e-01, 3.56480271e-01,
3.53555059e-01, 3.41067921e-01, 3.40185812e-01, 3.30713733e-01,
3.28068832e-01, 3.11941526e-01, 3.11145218e-01, 3.04942832e-01,
3.03211961e-01, 2.76465104e-01, 2.76106724e-01, 2.68938573e-01,
2.62950012e-01, 2.57892042e-01, 2.57227758e-01, 2.31938463e-01,
2.30362095e-01, 2.25980078e-01, 2.25440213e-01, 2.05879790e-01,
2.02882152e-01, 2.02880868e-01, 2.01832278e-01, 2.01165755e-01,
1.94087843e-01, 1.88025837e-01, 1.87851863e-01, 1.85196453e-01,
1.75513007e-01, 6.79076553e-02, 6.69401546e-02, 6.42453954e-02,
6.41505789e-02, 3.50408809e-02, 3.47843524e-02, 3.31473257e-02,
3.27886549e-02, 3.24329006e-02, 3.23346779e-02, 2.41453604e-02,
2.38783881e-02, 1.10436282e-02, 1.07701367e-02, 7.51852648e-04])
```

## 5 -> Model Building - Decision Tree

### 5.1 Import the Model Building Libraries

```
[52]: from sklearn.tree import DecisionTreeClassifier
      from sklearn.model_selection import GridSearchCV
```

### 5.2 Initializing the Model

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

### 5.3 Hyper parametering and Training of Model

```
[54]: parameters=[{
    'criterion':['gini','Entropy'],
    'splitter':['best','random'],
    'max_depth':[1,2,3,4,5],
    'max_features':['auto','sqrt','log2'],
    'random_state':[0,42],
}]

griddtc=GridSearchCV(dtc,param_grid=parameters,cv=5,scoring='accuracy')
```

### 5.4 Training the Model

```
[55]: griddtc.fit(x_train,y_train)
```

```
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
```

```
warnings.warn(
```

```
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
```

```
warnings.warn(
```

```
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
```

```
warnings.warn(
```

```
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
```

```
warnings.warn(
```



```

explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-

```





```

warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.

```





```

explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
    warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-

```



```

warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.

```





C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\model\_selection\\_validation.py:378: FitFailedWarning: 300 fits failed out of a total of 600.  
The score on these train-test partitions for these parameters will be set to nan.  
If these failures are not expected, you can try to debug them by setting error\_score='raise'.

Below are more details about the failures:

```
-----
300 fits failed with the following error:
Traceback (most recent call last):
  File "C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\model_selection\_validation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File "C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\tree\_classes.py", line 889, in fit
    super().fit(
  File "C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\tree\_classes.py", line 177, in fit
    self._validate_params()
  File "C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\base.py", line 600, in _validate_params
    validate_parameter_constraints(
  File "C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\utils\_param_validation.py", line 97, in
validate_parameter_constraints
    raise InvalidParameterError(
sklearn.utils._param_validation.InvalidParameterError: The 'criterion' parameter
of DecisionTreeClassifier must be a str among {'gini', 'log_loss', 'entropy'}.
Got 'Entropy' instead.
```

```
warnings.warn(some_fits_failed_message, FitFailedWarning)
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\model_selection\_search.py:952: UserWarning: One or more of the
test scores are non-finite: [0.84013704 0.84013704 0.84013704 0.84013704
0.84013704 0.84013704 0.84013704 0.84013704 0.84013704 0.84013704
0.84013704 0.84013704 0.83758384 0.84013704 0.84013704 0.84013704
0.83758384 0.84013704 0.84013704 0.84013704 0.84013704 0.84013704
0.83502705 0.83164082 0.84014064 0.83929318 0.83502705 0.83164082
0.84014064 0.83929318 0.84096646 0.84183916 0.84185359 0.84269383
0.82908042 0.8358745 0.83758384 0.84438154 0.82908042 0.8358745
0.83758384 0.84438154 0.82822214 0.83077533 0.83929318 0.84184638
0.83079697 0.83248107 0.81634331 0.84268662 0.83079697 0.83248107
0.81634331 0.84268662 0.83588172 0.84013343 0.83164443 0.84012982
nan nan nan nan nan nan
nan nan nan nan nan nan
```

```

nan      nan      nan      nan      nan      nan
nan      nan      nan      nan      nan      nan
nan      nan      nan      nan      nan      nan
nan      nan      nan      nan      nan      nan
nan      nan      nan      nan      nan      nan
nan      nan      nan      nan      nan      nan
nan      nan      nan      nan      nan      nan
nan      nan      nan      nan      nan      nan

```

```

warnings.warn(
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\tree\_classes.py:269: FutureWarning: `max_features='auto'` has
been deprecated in 1.1 and will be removed in 1.3. To keep the past behaviour,
explicitly set `max_features='sqrt'`.
warnings.warn(

```

```

[55]: GridSearchCV(cv=5, estimator=DecisionTreeClassifier(),
        param_grid=[{'criterion': ['gini', 'Entropy'],
                      'max_depth': [1, 2, 3, 4, 5],
                      'max_features': ['auto', 'sqrt', 'log2'],
                      'random_state': [0, 42],
                      'splitter': ['best', 'random']}],
        scoring='accuracy')

```

```

[56]: griddtc.best_params_

```

```

[56]: {'criterion': 'gini',
      'max_depth': 4,
      'max_features': 'auto',
      'random_state': 42,
      'splitter': 'random'}

```

## 5.5 Testing the Model

```

[57]: y_pred1=griddtc.predict(x_test)

```

```

[58]: pd.DataFrame({"Actual_values":y_test,"Predicted_values":y_pred1})

```

```

[58]:
Actual_values Predicted_values
442          0          0
1091         0          0
981          1          0
785          0          0
1332         1          1
...         ...         ...
1439         0          0
481          0          0
124          1          0

```

```
198          0          0
1229         0          0
```

```
[294 rows x 2 columns]
```

## 5.6 Evaluation of Model & Performance metrics

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

```
[60]: print("Accuracy of model :", accuracy_score(y_test, y_pred1))
```

```
Accuracy of model : 0.8367346938775511
```

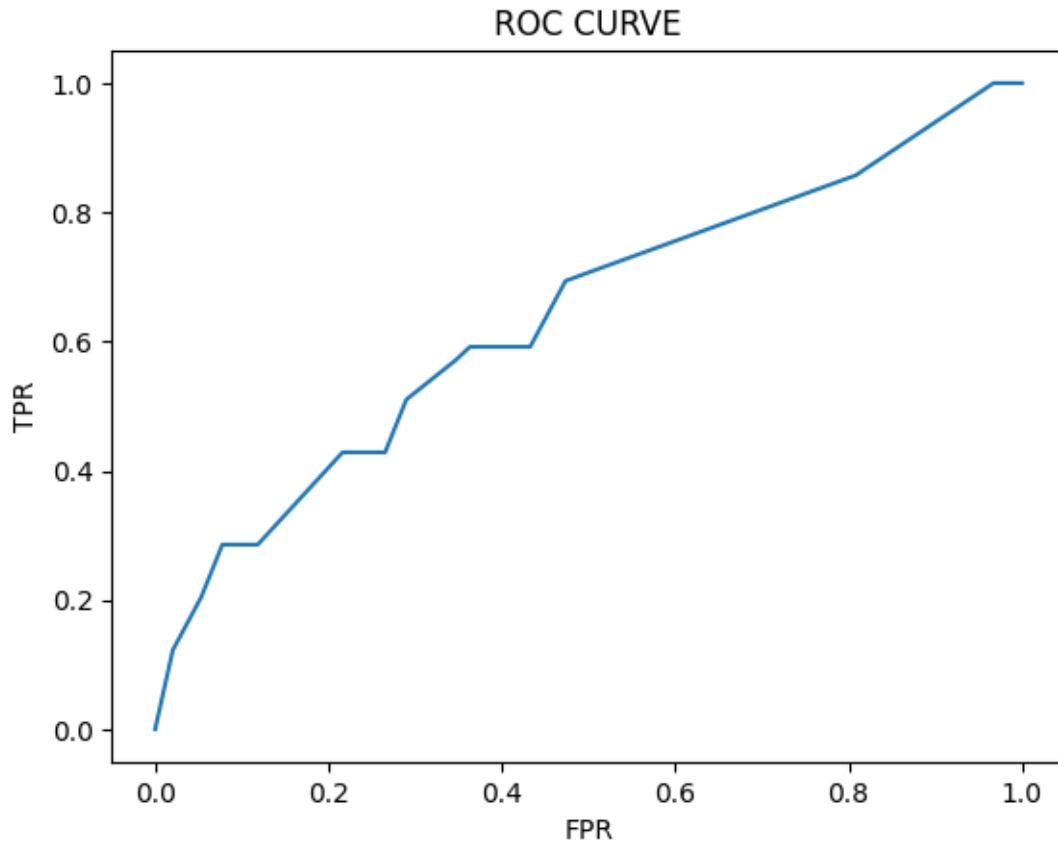
```
[61]: confusion_matrix(y_test, y_pred1)
```

```
[61]: array([[240,  5],
           [ 43,  6]], dtype=int64)
```

```
[62]: print(classification_report(y_test, y_pred1))
```

	precision	recall	f1-score	support
0	0.85	0.98	0.91	245
1	0.55	0.12	0.20	49
accuracy			0.84	294
macro avg	0.70	0.55	0.55	294
weighted avg	0.80	0.84	0.79	294

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



```
[64]: threshsholds
```

```
[64]: array([1.67647059, 0.67647059, 0.39583333, 0.37037037, 0.33823529,
          0.23577236, 0.2          , 0.18867925, 0.18604651, 0.16666667,
          0.15517241, 0.08474576, 0.07715134, 0.05936073, 0.          ])
```

```
[65]: #Tree Visualization using basic Decision Tree
dtc.fit(x_train,y_train)
from sklearn import tree
plt.figure(figsize=(25,15))
tree.plot_tree(dtc,filled=True)
```

```
[65]: [Text(0.33287432553956836, 0.96875, 'x[16] <= -1.397\ngini = 0.269\nsamples =
1176\nvalue = [988, 188]'),
Text(0.09892086330935251, 0.90625, 'x[42] <= 0.387\ngini = 0.5\nsamples =
78\nvalue = [39, 39]'),
Text(0.0539568345323741, 0.84375, 'x[2] <= 0.902\ngini = 0.426\nsamples =
39\nvalue = [27, 12]'),
Text(0.03597122302158273, 0.78125, 'x[23] <= 0.797\ngini = 0.312\nsamples =
31\nvalue = [25, 6]'),
```

```

Text(0.02158273381294964, 0.71875, 'x[8] <= -1.114\ngini = 0.198\nsamples =
27\nvalue = [24, 3]'),
Text(0.014388489208633094, 0.65625, 'x[43] <= 0.482\ngini = 0.5\nsamples =
6\nvalue = [3, 3]'),
Text(0.007194244604316547, 0.59375, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
Text(0.02158273381294964, 0.59375, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
Text(0.02877697841726619, 0.65625, 'gini = 0.0\nsamples = 21\nvalue = [21,
0]'),
Text(0.050359712230215826, 0.71875, 'x[1] <= 1.414\ngini = 0.375\nsamples =
4\nvalue = [1, 3]'),
Text(0.04316546762589928, 0.65625, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
Text(0.05755395683453238, 0.65625, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.07194244604316546, 0.78125, 'x[12] <= 1.446\ngini = 0.375\nsamples =
8\nvalue = [2, 6]'),
Text(0.06474820143884892, 0.71875, 'gini = 0.0\nsamples = 6\nvalue = [0, 6]'),
Text(0.07913669064748201, 0.71875, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.14388489208633093, 0.84375, 'x[38] <= 0.755\ngini = 0.426\nsamples =
39\nvalue = [12, 27]'),
Text(0.11510791366906475, 0.78125, 'x[29] <= 0.397\ngini = 0.26\nsamples =
26\nvalue = [4, 22]'),
Text(0.09352517985611511, 0.71875, 'x[5] <= 1.482\ngini = 0.095\nsamples =
20\nvalue = [1, 19]'),
Text(0.08633093525179857, 0.65625, 'gini = 0.0\nsamples = 18\nvalue = [0,
18]'),
Text(0.10071942446043165, 0.65625, 'x[16] <= -1.556\ngini = 0.5\nsamples =
2\nvalue = [1, 1]'),
Text(0.09352517985611511, 0.59375, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.1079136690647482, 0.59375, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.1366906474820144, 0.71875, 'x[2] <= 0.038\ngini = 0.5\nsamples =
6\nvalue = [3, 3]'),
Text(0.12949640287769784, 0.65625, 'x[12] <= -0.467\ngini = 0.375\nsamples =
4\nvalue = [1, 3]'),
Text(0.1223021582733813, 0.59375, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
Text(0.1366906474820144, 0.59375, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.14388489208633093, 0.65625, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.17266187050359713, 0.78125, 'x[1] <= 0.712\ngini = 0.473\nsamples =
13\nvalue = [8, 5]'),
Text(0.16546762589928057, 0.71875, 'x[10] <= 1.103\ngini = 0.32\nsamples =
10\nvalue = [8, 2]'),
Text(0.15827338129496402, 0.65625, 'gini = 0.0\nsamples = 8\nvalue = [8, 0]'),
Text(0.17266187050359713, 0.65625, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.17985611510791366, 0.71875, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
Text(0.5668277877697842, 0.90625, 'x[43] <= 0.482\ngini = 0.235\nsamples =
1098\nvalue = [949, 149]'),
Text(0.31935701438848924, 0.84375, 'x[18] <= -1.786\ngini = 0.162\nsamples =
798\nvalue = [727, 71]'),
Text(0.20863309352517986, 0.78125, 'x[17] <= 1.161\ngini = 0.38\nsamples =

```

```

47\nvalue = [35, 12]'),
  Text(0.2014388489208633, 0.71875, 'x[6] <= -0.323\ngini = 0.325\nsamples =
44\nvalue = [35, 9]'),
  Text(0.18705035971223022, 0.65625, 'x[1] <= 0.852\ngini = 0.498\nsamples =
15\nvalue = [8, 7]'),
  Text(0.17985611510791366, 0.59375, 'x[2] <= 0.532\ngini = 0.42\nsamples =
10\nvalue = [3, 7]'),
  Text(0.17266187050359713, 0.53125, 'gini = 0.0\nsamples = 6\nvalue = [0, 6]'),
  Text(0.18705035971223022, 0.53125, 'x[8] <= -1.114\ngini = 0.375\nsamples =
4\nvalue = [3, 1]'),
  Text(0.17985611510791366, 0.46875, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
  Text(0.19424460431654678, 0.46875, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
  Text(0.19424460431654678, 0.59375, 'gini = 0.0\nsamples = 5\nvalue = [5, 0]'),
  Text(0.2158273381294964, 0.65625, 'x[0] <= -0.539\ngini = 0.128\nsamples =
29\nvalue = [27, 2]'),
  Text(0.20863309352517986, 0.59375, 'x[42] <= 0.387\ngini = 0.408\nsamples =
7\nvalue = [5, 2]'),
  Text(0.2014388489208633, 0.53125, 'gini = 0.0\nsamples = 4\nvalue = [4, 0]'),
  Text(0.2158273381294964, 0.53125, 'x[9] <= -0.648\ngini = 0.444\nsamples =
3\nvalue = [1, 2]'),
  Text(0.20863309352517986, 0.46875, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
  Text(0.22302158273381295, 0.46875, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
  Text(0.22302158273381295, 0.59375, 'gini = 0.0\nsamples = 22\nvalue = [22,
0]'),
  Text(0.2158273381294964, 0.71875, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
  Text(0.4300809352517986, 0.78125, 'x[19] <= -0.309\ngini = 0.145\nsamples =
751\nvalue = [692, 59]'),
  Text(0.3444244604316547, 0.71875, 'x[4] <= -1.118\ngini = 0.218\nsamples =
257\nvalue = [225, 32]'),
  Text(0.30755395683453235, 0.65625, 'x[22] <= -0.445\ngini = 0.355\nsamples =
65\nvalue = [50, 15]'),
  Text(0.27697841726618705, 0.59375, 'x[22] <= -1.039\ngini = 0.303\nsamples =
59\nvalue = [48, 11]'),
  Text(0.2517985611510791, 0.53125, 'x[6] <= -0.323\ngini = 0.463\nsamples =
22\nvalue = [14, 8]'),
  Text(0.23741007194244604, 0.46875, 'x[5] <= -1.151\ngini = 0.198\nsamples =
9\nvalue = [8, 1]'),
  Text(0.2302158273381295, 0.40625, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
  Text(0.2446043165467626, 0.40625, 'gini = 0.0\nsamples = 8\nvalue = [8, 0]'),
  Text(0.26618705035971224, 0.46875, 'x[5] <= -0.388\ngini = 0.497\nsamples =
13\nvalue = [6, 7]'),
  Text(0.2589928057553957, 0.40625, 'gini = 0.0\nsamples = 4\nvalue = [4, 0]'),
  Text(0.2733812949640288, 0.40625, 'x[2] <= -0.024\ngini = 0.346\nsamples =
9\nvalue = [2, 7]'),
  Text(0.26618705035971224, 0.34375, 'x[14] <= -0.659\ngini = 0.444\nsamples =
3\nvalue = [2, 1]'),
  Text(0.2589928057553957, 0.28125, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),

```

```

Text(0.2733812949640288, 0.28125, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.2805755395683453, 0.34375, 'gini = 0.0\nsamples = 6\nvalue = [0, 6]'),
Text(0.302158273381295, 0.53125, 'x[8] <= -1.114\ngini = 0.149\nsamples =
37\nvalue = [34, 3]'),
Text(0.2949640287769784, 0.46875, 'x[18] <= -0.37\ngini = 0.5\nsamples =
6\nvalue = [3, 3]'),
Text(0.28776978417266186, 0.40625, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
Text(0.302158273381295, 0.40625, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
Text(0.30935251798561153, 0.46875, 'gini = 0.0\nsamples = 31\nvalue = [31,
0]'),
Text(0.3381294964028777, 0.59375, 'x[32] <= -0.204\ngini = 0.444\nsamples =
6\nvalue = [2, 4]'),
Text(0.33093525179856115, 0.53125, 'x[6] <= -1.729\ngini = 0.444\nsamples =
3\nvalue = [2, 1]'),
Text(0.3237410071942446, 0.46875, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.3381294964028777, 0.46875, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.34532374100719426, 0.53125, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
Text(0.381294964028777, 0.65625, 'x[27] <= 0.178\ngini = 0.161\nsamples =
192\nvalue = [175, 17]'),
Text(0.37410071942446044, 0.59375, 'x[18] <= -0.37\ngini = 0.24\nsamples =
122\nvalue = [105, 17]'),
Text(0.3597122302158273, 0.53125, 'x[5] <= 0.399\ngini = 0.463\nsamples =
22\nvalue = [14, 8]'),
Text(0.35251798561151076, 0.46875, 'x[0] <= -0.156\ngini = 0.444\nsamples =
12\nvalue = [4, 8]'),
Text(0.34532374100719426, 0.40625, 'x[2] <= -0.949\ngini = 0.198\nsamples =
9\nvalue = [1, 8]'),
Text(0.3381294964028777, 0.34375, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.35251798561151076, 0.34375, 'gini = 0.0\nsamples = 8\nvalue = [0, 8]'),
Text(0.3597122302158273, 0.40625, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
Text(0.3669064748201439, 0.46875, 'gini = 0.0\nsamples = 10\nvalue = [10, 0]'),
Text(0.38848920863309355, 0.53125, 'x[1] <= -1.711\ngini = 0.164\nsamples =
100\nvalue = [91, 9]'),
Text(0.381294964028777, 0.46875, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.39568345323741005, 0.46875, 'x[9] <= -0.745\ngini = 0.149\nsamples =
99\nvalue = [91, 8]'),
Text(0.381294964028777, 0.40625, 'x[3] <= 1.55\ngini = 0.283\nsamples =
41\nvalue = [34, 7]'),
Text(0.37410071942446044, 0.34375, 'x[16] <= 1.219\ngini = 0.224\nsamples =
39\nvalue = [34, 5]'),
Text(0.3669064748201439, 0.28125, 'x[9] <= -0.766\ngini = 0.188\nsamples =
38\nvalue = [34, 4]'),
Text(0.3597122302158273, 0.21875, 'x[0] <= 1.706\ngini = 0.149\nsamples =
37\nvalue = [34, 3]'),
Text(0.35251798561151076, 0.15625, 'x[9] <= -0.848\ngini = 0.105\nsamples =
36\nvalue = [34, 2]'),
Text(0.34532374100719426, 0.09375, 'gini = 0.0\nsamples = 29\nvalue = [29,

```

```

0]'),
Text(0.3597122302158273, 0.09375, 'x[12] <= -0.467\ngini = 0.408\nsamples =
7\nvalue = [5, 2]'),
Text(0.35251798561151076, 0.03125, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.3669064748201439, 0.03125, 'gini = 0.0\nsamples = 5\nvalue = [5, 0]'),
Text(0.3669064748201439, 0.15625, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.37410071942446044, 0.21875, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.381294964028777, 0.28125, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.38848920863309355, 0.34375, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.41007194244604317, 0.40625, 'x[1] <= 1.626\ngini = 0.034\nsamples =
58\nvalue = [57, 1]'),
Text(0.4028776978417266, 0.34375, 'gini = 0.0\nsamples = 57\nvalue = [57, 0]'),
Text(0.4172661870503597, 0.34375, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.38848920863309355, 0.59375, 'gini = 0.0\nsamples = 70\nvalue = [70,
0]'),
Text(0.5157374100719424, 0.71875, 'x[9] <= 0.385\ngini = 0.103\nsamples =
494\nvalue = [467, 27]'),
Text(0.48201438848920863, 0.65625, 'x[20] <= 2.837\ngini = 0.056\nsamples =
345\nvalue = [335, 10]'),
Text(0.4748201438848921, 0.59375, 'x[22] <= 2.822\ngini = 0.051\nsamples =
344\nvalue = [335, 9]'),
Text(0.45323741007194246, 0.53125, 'x[5] <= 0.227\ngini = 0.046\nsamples =
342\nvalue = [334, 8]'),
Text(0.4316546762589928, 0.46875, 'x[11] <= 1.854\ngini = 0.01\nsamples =
202\nvalue = [201, 1]'),
Text(0.4244604316546763, 0.40625, 'gini = 0.0\nsamples = 184\nvalue = [184,
0]'),
Text(0.43884892086330934, 0.40625, 'x[7] <= -0.51\ngini = 0.105\nsamples =
18\nvalue = [17, 1]'),
Text(0.4316546762589928, 0.34375, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.4460431654676259, 0.34375, 'gini = 0.0\nsamples = 17\nvalue = [17, 0]'),
Text(0.4748201438848921, 0.46875, 'x[40] <= 1.922\ngini = 0.095\nsamples =
140\nvalue = [133, 7]'),
Text(0.4676258992805755, 0.40625, 'x[20] <= 0.137\ngini = 0.083\nsamples =
139\nvalue = [133, 6]'),
Text(0.460431654676259, 0.34375, 'x[23] <= 0.797\ngini = 0.161\nsamples =
68\nvalue = [62, 6]'),
Text(0.4316546762589928, 0.28125, 'x[14] <= -1.122\ngini = 0.098\nsamples =
58\nvalue = [55, 3]'),
Text(0.41007194244604317, 0.21875, 'x[1] <= 0.186\ngini = 0.346\nsamples =
9\nvalue = [7, 2]'),
Text(0.4028776978417266, 0.15625, 'gini = 0.0\nsamples = 6\nvalue = [6, 0]'),
Text(0.4172661870503597, 0.15625, 'x[4] <= -0.203\ngini = 0.444\nsamples =
3\nvalue = [1, 2]'),
Text(0.41007194244604317, 0.09375, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.4244604316546763, 0.09375, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.45323741007194246, 0.21875, 'x[9] <= -1.063\ngini = 0.04\nsamples =

```



```

49\nvalue = [48, 1]'),
  Text(0.4460431654676259, 0.15625, 'x[0] <= -0.594\ngini = 0.444\nsamples =
3\nvalue = [2, 1]'),
  Text(0.43884892086330934, 0.09375, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
  Text(0.45323741007194246, 0.09375, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
  Text(0.460431654676259, 0.15625, 'gini = 0.0\nsamples = 46\nvalue = [46, 0]'),
  Text(0.4892086330935252, 0.28125, 'x[8] <= -0.207\ngini = 0.42\nsamples =
10\nvalue = [7, 3]'),
  Text(0.48201438848920863, 0.21875, 'x[32] <= -0.204\ngini = 0.375\nsamples =
4\nvalue = [1, 3]'),
  Text(0.4748201438848921, 0.15625, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
  Text(0.4892086330935252, 0.15625, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
  Text(0.49640287769784175, 0.21875, 'gini = 0.0\nsamples = 6\nvalue = [6, 0]'),
  Text(0.4748201438848921, 0.34375, 'gini = 0.0\nsamples = 71\nvalue = [71, 0]'),
  Text(0.48201438848920863, 0.40625, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
  Text(0.49640287769784175, 0.53125, 'x[32] <= -0.204\ngini = 0.5\nsamples =
2\nvalue = [1, 1]'),
  Text(0.4892086330935252, 0.46875, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
  Text(0.5035971223021583, 0.46875, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
  Text(0.4892086330935252, 0.59375, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
  Text(0.5494604316546763, 0.65625, 'x[9] <= 0.391\ngini = 0.202\nsamples =
149\nvalue = [132, 17]'),
  Text(0.5422661870503597, 0.59375, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
  Text(0.5566546762589928, 0.59375, 'x[5] <= 1.654\ngini = 0.193\nsamples =
148\nvalue = [132, 16]'),
  Text(0.5494604316546763, 0.53125, 'x[1] <= -1.621\ngini = 0.183\nsamples =
147\nvalue = [132, 15]'),
  Text(0.5179856115107914, 0.46875, 'x[11] <= -0.432\ngini = 0.49\nsamples =
7\nvalue = [4, 3]'),
  Text(0.5107913669064749, 0.40625, 'x[9] <= 1.996\ngini = 0.375\nsamples =
4\nvalue = [1, 3]'),
  Text(0.5035971223021583, 0.34375, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
  Text(0.5179856115107914, 0.34375, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
  Text(0.5251798561151079, 0.40625, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
  Text(0.5809352517985612, 0.46875, 'x[41] <= 0.085\ngini = 0.157\nsamples =
140\nvalue = [128, 12]'),
  Text(0.5503597122302158, 0.40625, 'x[16] <= 2.726\ngini = 0.07\nsamples =
82\nvalue = [79, 3]'),
  Text(0.5323741007194245, 0.34375, 'x[1] <= -1.47\ngini = 0.049\nsamples =
80\nvalue = [78, 2]'),
  Text(0.5179856115107914, 0.28125, 'x[4] <= -0.203\ngini = 0.5\nsamples =
2\nvalue = [1, 1]'),
  Text(0.5107913669064749, 0.21875, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
  Text(0.5251798561151079, 0.21875, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
  Text(0.5467625899280576, 0.28125, 'x[2] <= 1.765\ngini = 0.025\nsamples =
78\nvalue = [77, 1]'),
  Text(0.539568345323741, 0.21875, 'gini = 0.0\nsamples = 76\nvalue = [76, 0]'),

```

```

Text(0.5539568345323741, 0.21875, 'x[12] <= 0.079\ngini = 0.5\nsamples =
2\nvalue = [1, 1]'),
Text(0.5467625899280576, 0.15625, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.5611510791366906, 0.15625, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.5683453237410072, 0.34375, 'x[10] <= -0.723\ngini = 0.5\nsamples =
2\nvalue = [1, 1]'),
Text(0.5611510791366906, 0.28125, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.5755395683453237, 0.28125, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6115107913669064, 0.40625, 'x[9] <= 1.606\ngini = 0.262\nsamples =
58\nvalue = [49, 9]'),
Text(0.60431654676259, 0.34375, 'x[0] <= 0.611\ngini = 0.375\nsamples =
36\nvalue = [27, 9]'),
Text(0.5899280575539568, 0.28125, 'x[20] <= 2.087\ngini = 0.211\nsamples =
25\nvalue = [22, 3]'),
Text(0.5827338129496403, 0.21875, 'x[10] <= 1.615\ngini = 0.153\nsamples =
24\nvalue = [22, 2]'),
Text(0.5755395683453237, 0.15625, 'x[17] <= 1.161\ngini = 0.083\nsamples =
23\nvalue = [22, 1]'),
Text(0.5683453237410072, 0.09375, 'gini = 0.0\nsamples = 21\nvalue = [21, 0]'),
Text(0.5827338129496403, 0.09375, 'x[5] <= -0.068\ngini = 0.5\nsamples =
2\nvalue = [1, 1]'),
Text(0.5755395683453237, 0.03125, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.5899280575539568, 0.03125, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.5899280575539568, 0.15625, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.5971223021582733, 0.21875, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6187050359712231, 0.28125, 'x[11] <= 0.025\ngini = 0.496\nsamples =
11\nvalue = [5, 6]'),
Text(0.6115107913669064, 0.21875, 'gini = 0.0\nsamples = 4\nvalue = [0, 4]'),
Text(0.6258992805755396, 0.21875, 'x[13] <= 0.96\ngini = 0.408\nsamples =
7\nvalue = [5, 2]'),
Text(0.6187050359712231, 0.15625, 'gini = 0.0\nsamples = 4\nvalue = [4, 0]'),
Text(0.6330935251798561, 0.15625, 'x[0] <= 0.939\ngini = 0.444\nsamples =
3\nvalue = [1, 2]'),
Text(0.6258992805755396, 0.09375, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.6402877697841727, 0.09375, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.6187050359712231, 0.34375, 'gini = 0.0\nsamples = 22\nvalue = [22, 0]'),
Text(0.5638489208633094, 0.53125, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.8142985611510791, 0.84375, 'x[9] <= -0.458\ngini = 0.385\nsamples =
300\nvalue = [222, 78]'),
Text(0.7095323741007195, 0.78125, 'x[15] <= -0.271\ngini = 0.5\nsamples =
96\nvalue = [49, 47]'),
Text(0.6726618705035972, 0.71875, 'x[2] <= -0.456\ngini = 0.459\nsamples =
42\nvalue = [15, 27]'),
Text(0.6474820143884892, 0.65625, 'x[10] <= -0.283\ngini = 0.499\nsamples =
23\nvalue = [12, 11]'),
Text(0.6330935251798561, 0.59375, 'x[9] <= -1.032\ngini = 0.426\nsamples =
13\nvalue = [4, 9]'),

```

```

Text(0.6258992805755396, 0.53125, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.6402877697841727, 0.53125, 'x[17] <= -1.173\ngini = 0.298\nsamples =
11\nvalue = [2, 9]'),
Text(0.6330935251798561, 0.46875, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.6474820143884892, 0.46875, 'x[0] <= -1.087\ngini = 0.18\nsamples =
10\nvalue = [1, 9]'),
Text(0.6402877697841727, 0.40625, 'x[24] <= -0.462\ngini = 0.5\nsamples =
2\nvalue = [1, 1]'),
Text(0.6330935251798561, 0.34375, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.6474820143884892, 0.34375, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.6546762589928058, 0.40625, 'gini = 0.0\nsamples = 8\nvalue = [0, 8]'),
Text(0.6618705035971223, 0.59375, 'x[17] <= 1.161\ngini = 0.32\nsamples =
10\nvalue = [8, 2]'),
Text(0.6546762589928058, 0.53125, 'gini = 0.0\nsamples = 8\nvalue = [8, 0]'),
Text(0.6690647482014388, 0.53125, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.697841726618705, 0.65625, 'x[7] <= -0.51\ngini = 0.266\nsamples =
19\nvalue = [3, 16]'),
Text(0.6906474820143885, 0.59375, 'x[5] <= -1.077\ngini = 0.198\nsamples =
18\nvalue = [2, 16]'),
Text(0.6834532374100719, 0.53125, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.697841726618705, 0.53125, 'x[20] <= 0.437\ngini = 0.111\nsamples =
17\nvalue = [1, 16]'),
Text(0.6906474820143885, 0.46875, 'gini = 0.0\nsamples = 15\nvalue = [0, 15]'),
Text(0.7050359712230215, 0.46875, 'x[16] <= -0.446\ngini = 0.5\nsamples =
2\nvalue = [1, 1]'),
Text(0.697841726618705, 0.40625, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.7122302158273381, 0.40625, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.7050359712230215, 0.59375, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.7464028776978417, 0.71875, 'x[0] <= -1.141\ngini = 0.466\nsamples =
54\nvalue = [34, 20]'),
Text(0.7266187050359713, 0.65625, 'x[0] <= -1.579\ngini = 0.245\nsamples =
7\nvalue = [1, 6]'),
Text(0.7194244604316546, 0.59375, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.7338129496402878, 0.59375, 'gini = 0.0\nsamples = 6\nvalue = [0, 6]'),
Text(0.7661870503597122, 0.65625, 'x[1] <= 0.419\ngini = 0.418\nsamples =
47\nvalue = [33, 14]'),
Text(0.7482014388489209, 0.59375, 'x[1] <= -1.236\ngini = 0.482\nsamples =
32\nvalue = [19, 13]'),
Text(0.7338129496402878, 0.53125, 'x[1] <= -1.655\ngini = 0.18\nsamples =
10\nvalue = [9, 1]'),
Text(0.7266187050359713, 0.46875, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.7410071942446043, 0.46875, 'gini = 0.0\nsamples = 9\nvalue = [9, 0]'),
Text(0.762589928057554, 0.53125, 'x[10] <= 1.329\ngini = 0.496\nsamples =
22\nvalue = [10, 12]'),
Text(0.7553956834532374, 0.46875, 'x[27] <= 0.178\ngini = 0.465\nsamples =
19\nvalue = [7, 12]'),
Text(0.7410071942446043, 0.40625, 'x[8] <= 0.7\ngini = 0.298\nsamples =

```

```

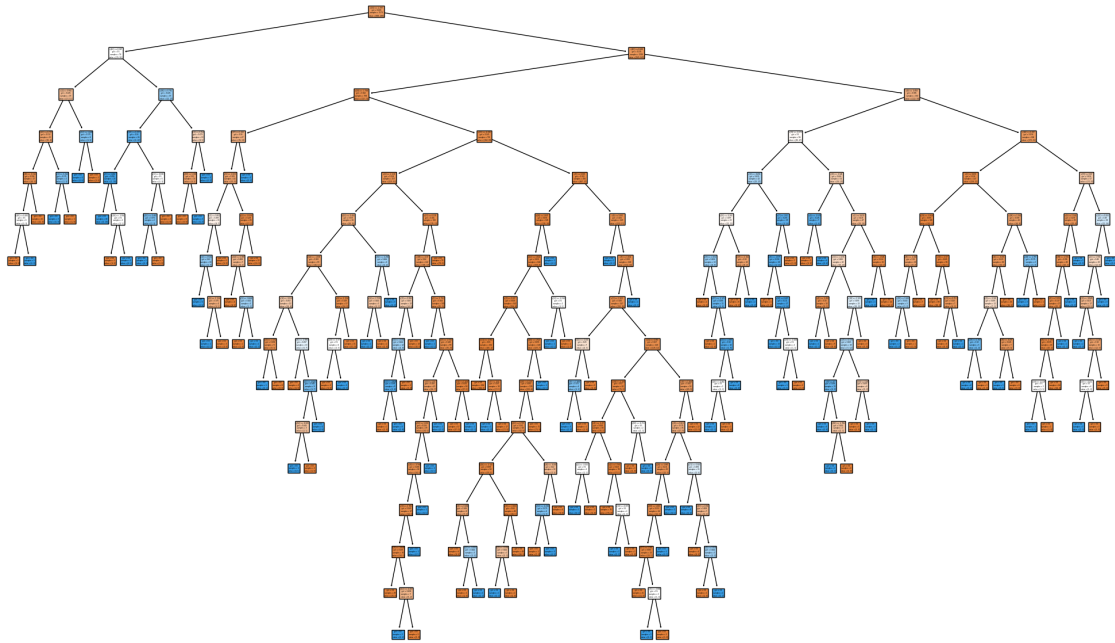
11\nvalue = [2, 9]'),
Text(0.7338129496402878, 0.34375, 'gini = 0.0\nsamples = 8\nvalue = [0, 8]'),
Text(0.7482014388489209, 0.34375, 'x[18] <= -1.786\ngini = 0.444\nsamples =
3\nvalue = [2, 1]'),
Text(0.7410071942446043, 0.28125, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.7553956834532374, 0.28125, 'gini = 0.0\nsamples = 2\nvalue = [2, 0]'),
Text(0.7697841726618705, 0.40625, 'x[34] <= 0.85\ngini = 0.469\nsamples =
8\nvalue = [5, 3]'),
Text(0.762589928057554, 0.34375, 'gini = 0.0\nsamples = 5\nvalue = [5, 0]'),
Text(0.7769784172661871, 0.34375, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
Text(0.7697841726618705, 0.46875, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
Text(0.7841726618705036, 0.59375, 'x[5] <= -1.545\ngini = 0.124\nsamples =
15\nvalue = [14, 1]'),
Text(0.7769784172661871, 0.53125, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.7913669064748201, 0.53125, 'gini = 0.0\nsamples = 14\nvalue = [14, 0]'),
Text(0.9190647482014388, 0.78125, 'x[42] <= 0.387\ngini = 0.258\nsamples =
204\nvalue = [173, 31]'),
Text(0.8669064748201439, 0.71875, 'x[2] <= 0.655\ngini = 0.138\nsamples =
147\nvalue = [136, 11]'),
Text(0.8273381294964028, 0.65625, 'x[19] <= -0.824\ngini = 0.056\nsamples =
105\nvalue = [102, 3]'),
Text(0.8129496402877698, 0.59375, 'x[5] <= -1.102\ngini = 0.32\nsamples =
10\nvalue = [8, 2]'),
Text(0.8057553956834532, 0.53125, 'x[35] <= 1.695\ngini = 0.444\nsamples =
3\nvalue = [1, 2]'),
Text(0.7985611510791367, 0.46875, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.8129496402877698, 0.46875, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.8201438848920863, 0.53125, 'gini = 0.0\nsamples = 7\nvalue = [7, 0]'),
Text(0.841726618705036, 0.59375, 'x[0] <= 1.87\ngini = 0.021\nsamples =
95\nvalue = [94, 1]'),
Text(0.8345323741007195, 0.53125, 'gini = 0.0\nsamples = 88\nvalue = [88, 0]'),
Text(0.8489208633093526, 0.53125, 'x[11] <= 0.711\ngini = 0.245\nsamples =
7\nvalue = [6, 1]'),
Text(0.841726618705036, 0.46875, 'gini = 0.0\nsamples = 6\nvalue = [6, 0]'),
Text(0.8561151079136691, 0.46875, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.9064748201438849, 0.65625, 'x[28] <= 1.262\ngini = 0.308\nsamples =
42\nvalue = [34, 8]'),
Text(0.8920863309352518, 0.59375, 'x[4] <= -0.203\ngini = 0.229\nsamples =
38\nvalue = [33, 5]'),
Text(0.8848920863309353, 0.53125, 'x[0] <= -0.101\ngini = 0.486\nsamples =
12\nvalue = [7, 5]'),
Text(0.8705035971223022, 0.46875, 'x[19] <= 1.367\ngini = 0.32\nsamples =
5\nvalue = [1, 4]'),
Text(0.8633093525179856, 0.40625, 'gini = 0.0\nsamples = 4\nvalue = [0, 4]'),
Text(0.8776978417266187, 0.40625, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.8992805755395683, 0.46875, 'x[8] <= -0.207\ngini = 0.245\nsamples =
7\nvalue = [6, 1]'),

```

```

Text(0.8920863309352518, 0.40625, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.9064748201438849, 0.40625, 'gini = 0.0\nsamples = 6\nvalue = [6, 0]'),
Text(0.8992805755395683, 0.53125, 'gini = 0.0\nsamples = 26\nvalue = [26, 0]'),
Text(0.920863309352518, 0.59375, 'x[20] <= 1.787\ngini = 0.375\nsamples =
4\nvalue = [1, 3]'),
Text(0.9136690647482014, 0.53125, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
Text(0.9280575539568345, 0.53125, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.9712230215827338, 0.71875, 'x[39] <= 0.67\ngini = 0.456\nsamples =
57\nvalue = [37, 20]'),
Text(0.9568345323741008, 0.65625, 'x[0] <= 2.144\ngini = 0.238\nsamples =
29\nvalue = [25, 4]'),
Text(0.9496402877697842, 0.59375, 'x[30] <= 1.936\ngini = 0.191\nsamples =
28\nvalue = [25, 3]'),
Text(0.9424460431654677, 0.53125, 'x[5] <= 1.555\ngini = 0.137\nsamples =
27\nvalue = [25, 2]'),
Text(0.935251798561151, 0.46875, 'x[6] <= -1.729\ngini = 0.074\nsamples =
26\nvalue = [25, 1]'),
Text(0.9280575539568345, 0.40625, 'x[16] <= -0.128\ngini = 0.5\nsamples =
2\nvalue = [1, 1]'),
Text(0.920863309352518, 0.34375, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.935251798561151, 0.34375, 'gini = 0.0\nsamples = 1\nvalue = [1, 0]'),
Text(0.9424460431654677, 0.40625, 'gini = 0.0\nsamples = 24\nvalue = [24, 0]'),
Text(0.9496402877697842, 0.46875, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.9568345323741008, 0.53125, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.9640287769784173, 0.59375, 'gini = 0.0\nsamples = 1\nvalue = [0, 1]'),
Text(0.9856115107913669, 0.65625, 'x[21] <= 0.026\ngini = 0.49\nsamples =
28\nvalue = [12, 16]'),
Text(0.9784172661870504, 0.59375, 'x[2] <= 1.765\ngini = 0.48\nsamples =
20\nvalue = [12, 8]'),
Text(0.9712230215827338, 0.53125, 'x[2] <= -0.949\ngini = 0.415\nsamples =
17\nvalue = [12, 5]'),
Text(0.9640287769784173, 0.46875, 'gini = 0.0\nsamples = 2\nvalue = [0, 2]'),
Text(0.9784172661870504, 0.46875, 'x[4] <= -1.118\ngini = 0.32\nsamples =
15\nvalue = [12, 3]'),
Text(0.9712230215827338, 0.40625, 'x[0] <= -0.211\ngini = 0.5\nsamples =
6\nvalue = [3, 3]'),
Text(0.9640287769784173, 0.34375, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
Text(0.9784172661870504, 0.34375, 'gini = 0.0\nsamples = 3\nvalue = [3, 0]'),
Text(0.9856115107913669, 0.40625, 'gini = 0.0\nsamples = 9\nvalue = [9, 0]'),
Text(0.9856115107913669, 0.53125, 'gini = 0.0\nsamples = 3\nvalue = [0, 3]'),
Text(0.9928057553956835, 0.59375, 'gini = 0.0\nsamples = 8\nvalue = [0, 8]')

```



## 6 → Model Building - Random Forest

### 6.1 Import the Model Building Libraries

```
[66]: from sklearn.ensemble import RandomForestClassifier
```

### 6.2 Initializing the Model

```
[67]: rfc=RandomForestClassifier()
```

### 6.3 Hyper parametering and Training of Model

```
[68]: from sklearn.model_selection import GridSearchCV
parameters=[{
    'max_depth': list(range(10, 15)),
    'max_features': list(range(0,14))
}]
gridrfc=GridSearchCV(rfc,param_grid=parameters,cv=5,scoring='accuracy')
```

### 6.4 Training the Model

```
[69]: gridrfc.fit(x_train,y_train)
```

```
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\model_selection\_validation.py:378: FitFailedWarning:
25 fits failed out of a total of 350.
The score on these train-test partitions for these parameters will be set to
nan.
If these failures are not expected, you can try to debug them by setting
error_score='raise'.
```

Below are more details about the failures:

```
-----
25 fits failed with the following error:
Traceback (most recent call last):
  File "C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\model_selection\_validation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File "C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\ensemble\_forest.py", line 340, in fit
    self._validate_params()
  File "C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\base.py", line 600, in _validate_params
    validate_parameter_constraints(
  File "C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\utils\_param_validation.py", line 97, in
validate_parameter_constraints
    raise InvalidParameterError(
sklearn.utils._param_validation.InvalidParameterError: The 'max_features'
parameter of RandomForestClassifier must be an int in the range [1, inf), a
float in the range (0.0, 1.0], a str among {'sqrt', 'log2', 'auto' (deprecated)}
or None. Got 0 instead.
```

```
warnings.warn(some_fits_failed_message, FitFailedWarning)
C:\Users\nitin\AppData\Local\Programs\Python\Python311\Lib\site-
packages\sklearn\model_selection\_search.py:952: UserWarning: One or more of the
test scores are non-finite: [      nan 0.85118644 0.85033177 0.85288136
0.85543455 0.85457988
0.85628201 0.85373963 0.85457988 0.85968265 0.85373603 0.85458348
0.85968265 0.85882798      nan 0.84948792 0.84948431 0.85543455
0.85288496 0.85544537 0.85457988 0.85373603 0.85798053 0.85457627
0.8588352  0.85883159 0.8588352  0.85882798      nan 0.84864407
0.85544176 0.85288136 0.85118283 0.8537216  0.85543815 0.85372521
0.85628201 0.85543455 0.85712946 0.85713307 0.85458348 0.85456906
      nan 0.85118283 0.85118644 0.85288136 0.85628561 0.85117923
0.8562784  0.85542373 0.85883159 0.86223224 0.86053732 0.85374324
0.85628922 0.85543815      nan 0.85033538 0.8520339  0.85458348
0.85458348 0.85288857 0.85457988 0.85882798 0.8562784  0.85713307
0.85628561 0.85458709 0.85798053 0.85287414]
warnings.warn(
```

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

```
[70]: gridrfc.best_params_
```

```
[70]: {'max_depth': 13, 'max_features': 9}
```

## 6.5 Testing the Model

```
[71]: y_pred2=gridrfc.predict(x_test)
```

```
[72]: pd.DataFrame({"Actual_values":y_test,"Predicted_values":y_pred2})
```

```
[72]:
```

	Actual_values	Predicted_values
442	0	0
1091	0	0
981	1	0
785	0	0
1332	1	1
...	...	...
1439	0	0
481	0	0
124	1	0
198	0	0
1229	0	0

```
[294 rows x 2 columns]
```

## 6.6 Evaluation of Model & Performance metrics

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

```
[74]: print("Accuracy of model :",accuracy_score(y_test,y_pred2))
```

```
Accuracy of model : 0.8571428571428571
```

```
[75]: confusion_matrix(y_test,y_pred2)
```

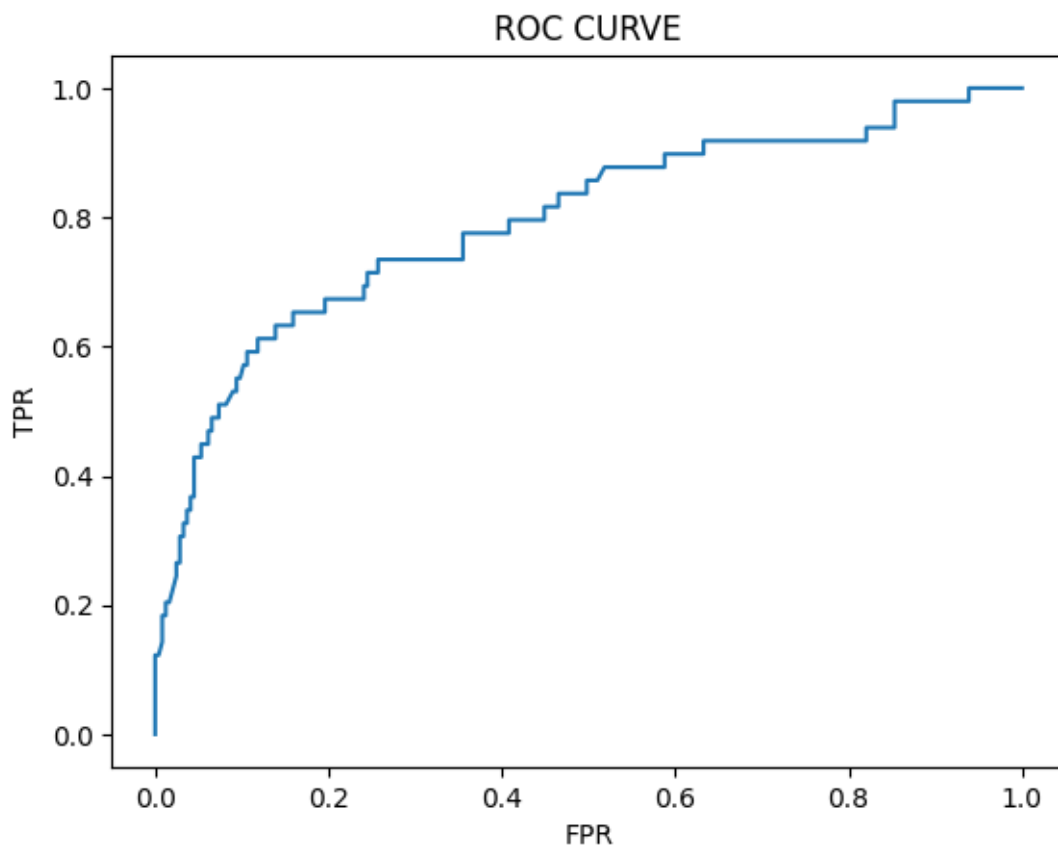
```
[75]: array([[243,  2],
          [ 40,  9]], dtype=int64)
```

```
[76]: print(classification_report(y_test,y_pred2))
```



	precision	recall	f1-score	support
0	0.86	0.99	0.92	245
1	0.82	0.18	0.30	49
accuracy			0.86	294
macro avg	0.84	0.59	0.61	294
weighted avg	0.85	0.86	0.82	294

```
[77]: #ROC-AUC Curve
probability=gridrfc.predict_proba(x_test)[:,-1]
fpr,tpr,threshholds = roc_curve(y_test,probability)
plt.plot(fpr,tpr)
plt.xlabel('FPR')
plt.ylabel('TPR')
plt.title('ROC CURVE')
plt.show()
```



```
[78]: threshholds
```

```
[78]: array([1.81      , 0.81      , 0.61333333, 0.59333333, 0.56      ,
0.511      , 0.48      , 0.47477584, 0.47      , 0.44      ,
0.43179845, 0.42      , 0.4       , 0.39796512, 0.38416667,
0.38133333, 0.37816667, 0.37      , 0.36993997, 0.36333333,
0.35      , 0.34090909, 0.34      , 0.33      , 0.32333333,
0.31538351, 0.31243478, 0.29331432, 0.29111111, 0.29046512,
0.29      , 0.28833333, 0.285      , 0.28      , 0.27076923,
0.26008264, 0.26      , 0.25042116, 0.25036364, 0.25      ,
0.24546512, 0.245      , 0.23      , 0.22924242, 0.21269231,
0.2104      , 0.21031746, 0.21      , 0.1904      , 0.19      ,
0.18363462, 0.18070764, 0.18046512, 0.18043478, 0.18027027,
0.18007634, 0.18      , 0.15237961, 0.152      , 0.15008264,
0.15      , 0.14008264, 0.14      , 0.1378847 , 0.13515152,
0.13089527, 0.13031746, 0.1301      , 0.13      , 0.12701638,
0.12111111, 0.120725 , 0.11324341, 0.11101549, 0.11020408,
0.11      , 0.10001165, 0.1       , 0.0997619 , 0.09573344,
0.09020408, 0.09      , 0.0845323 , 0.08215161, 0.08048462,
0.08      , 0.0754001 , 0.075      , 0.07008264, 0.07      ,
0.05364794, 0.05291435, 0.0475      , 0.04453597, 0.02454054,
0.02343333, 0.          ])
```