

Logistic Regression on Attrition

February 17, 2022

```
In [1]: import pandas as pd
import seaborn as sb
import matplotlib.pyplot as plt
import missingno as mo

A=pd.read_csv("C:/Users/abhijeet/Documents/attrition.csv")

In [2]: CAT=[]
CON=[]
for i in A.columns:
    if(A[i].dtype=="object"):
        CAT.append(i)
    else:
        CON.append(i)

In [3]: CAT

Out[3]: ['BusinessTravel',
'Department',
'EducationField',
'Gender',
'JobRole',
'MaritalStatus',
'Over18',
'OverTime']

In [4]: CON

Out[4]: ['Attrition',
'Age',
'DailyRate',
'DistanceFromHome',
'Education',
'EmployeeCount',
'EmployeeNumber',
'EnvironmentSatisfaction',
'HourlyRate',
'JobInvolvement',
```

```

'JobLevel',
'JobSatisfaction',
'MonthlyIncome',
'MonthlyRate',
'NumCompaniesWorked',
'PercentSalaryHike',
'PerformanceRating',
'RelationshipSatisfaction',
'StandardHours',
'StockOptionLevel',
'TotalWorkingYears',
'TrainingTimesLastYear',
'WorkLifeBalance',
'YearsAtCompany',
'YearsInCurrentRole',
'YearsSinceLastPromotion',
'YearsWithCurrManager']

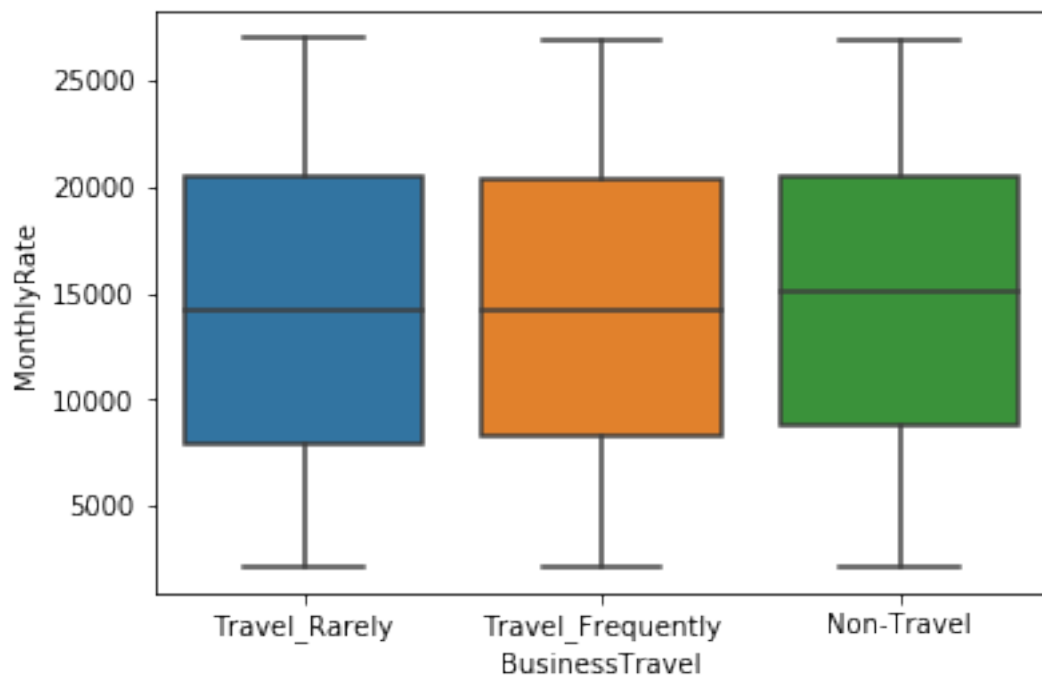
```

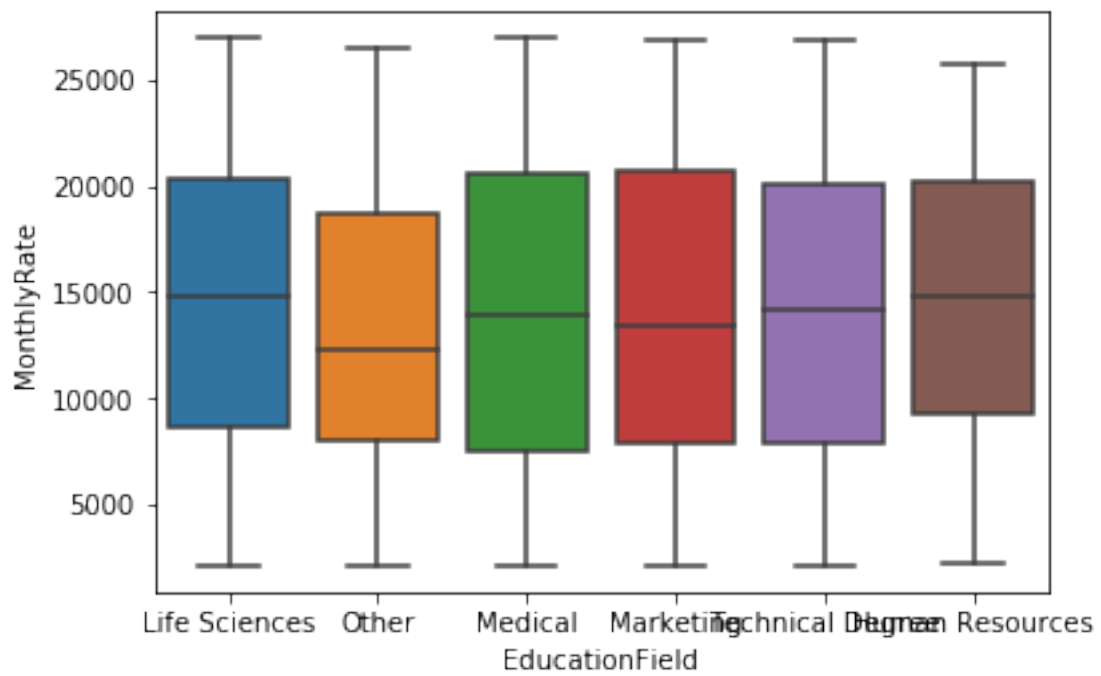
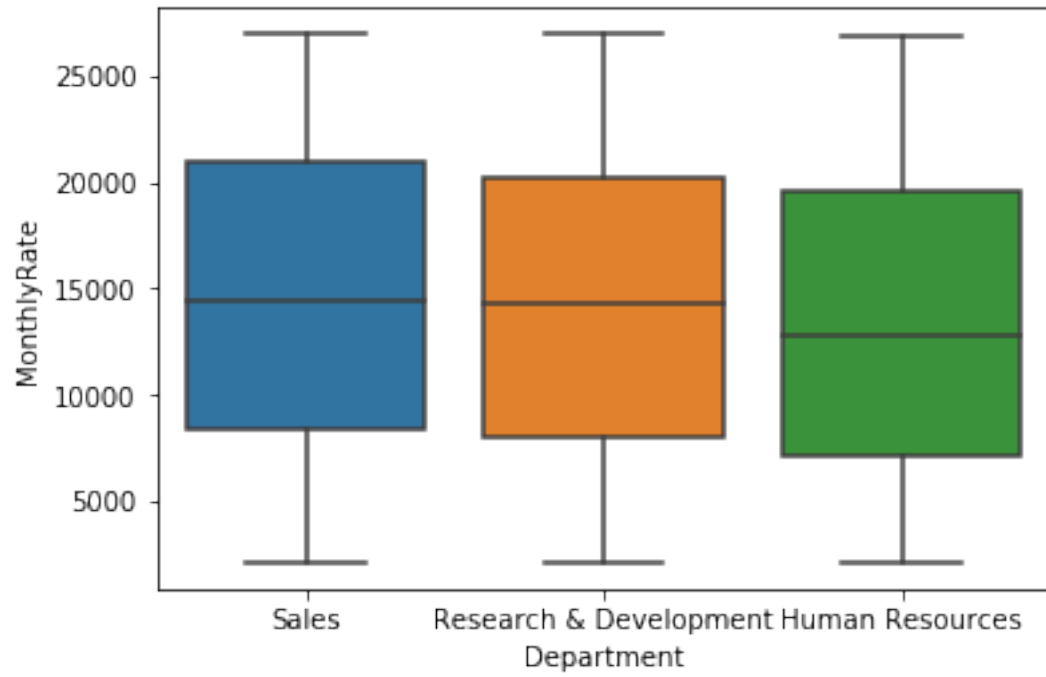
```

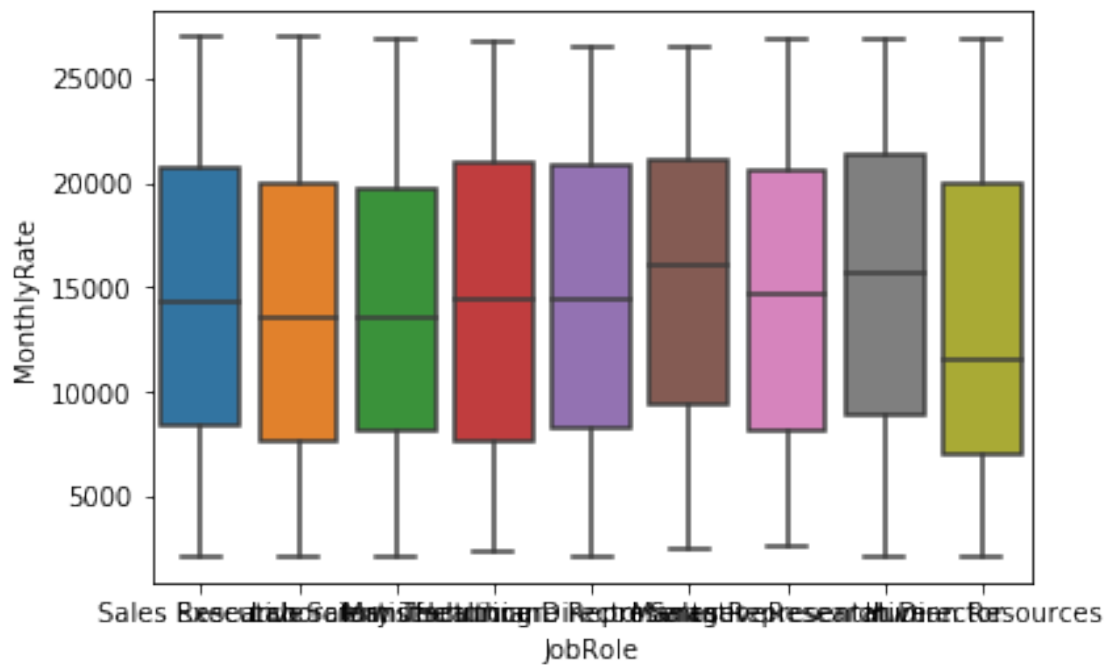
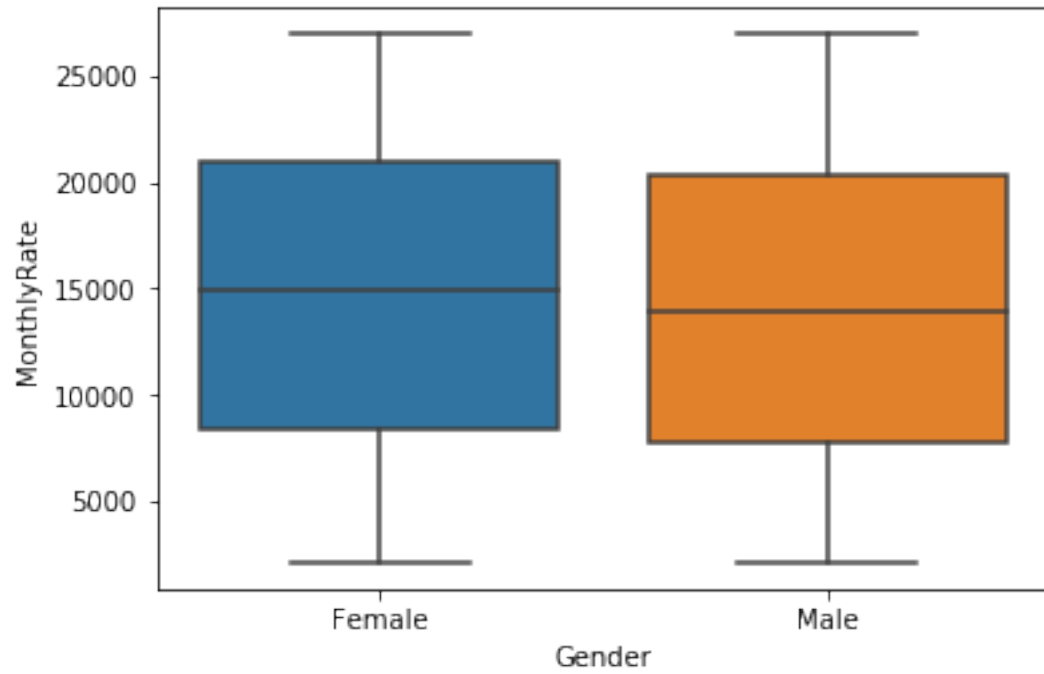
In [5]: import seaborn as sb
import matplotlib.pyplot as plt
for i in CAT:

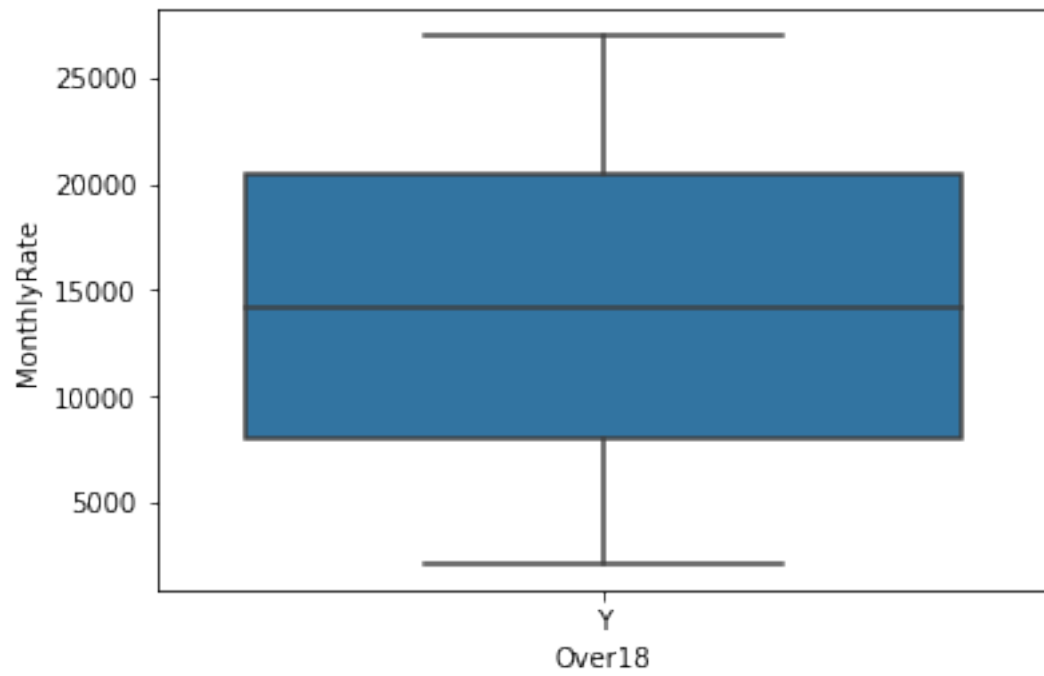
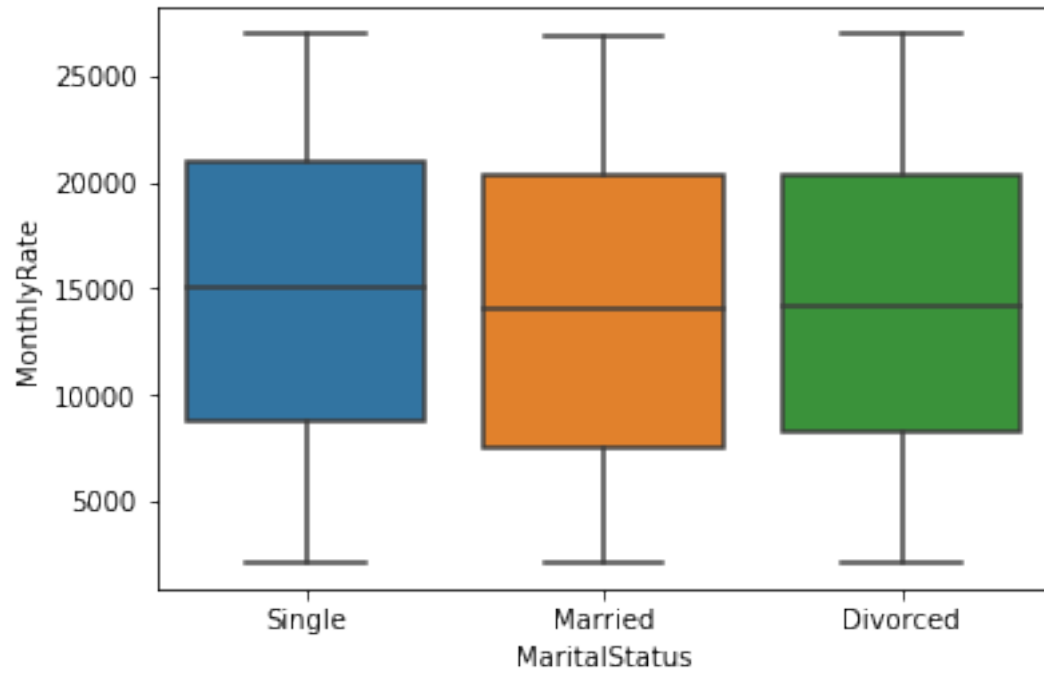
    sb.boxplot(A[i],A['MonthlyRate'])
plt.show()

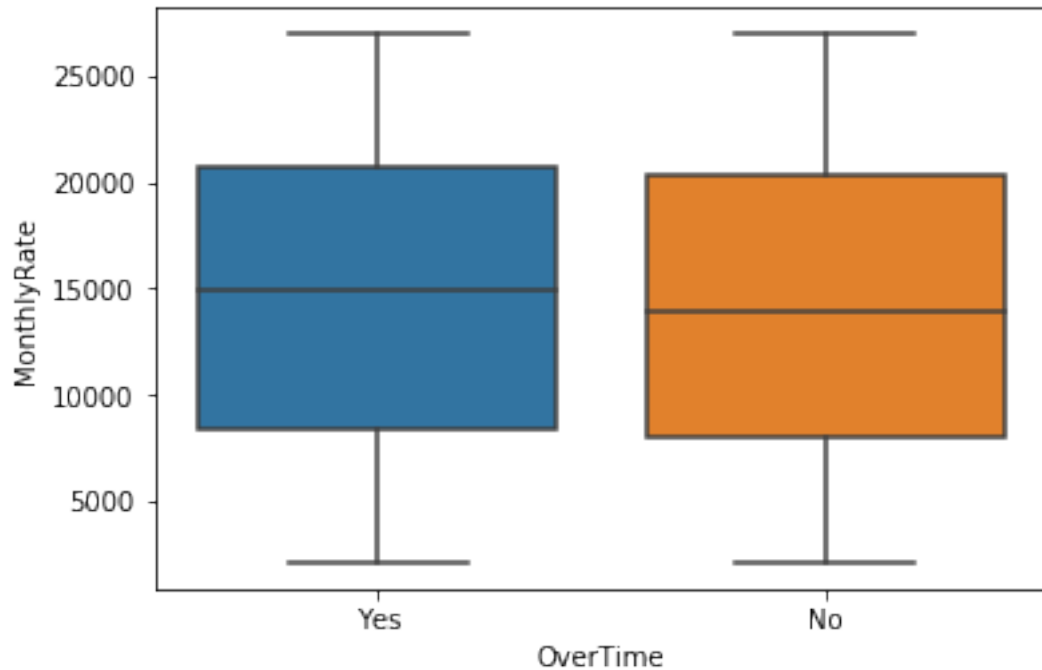
```











In [6]: A

```
Out[6]:
```

	Attrition	Age	BusinessTravel	DailyRate	Department \
0	1	41	Travel_Rarely	1102	Sales
1	0	49	Travel_Frequently	279	Research & Development
2	1	37	Travel_Rarely	1373	Research & Development
3	0	33	Travel_Frequently	1392	Research & Development
4	0	27	Travel_Rarely	591	Research & Development
...
1465	0	36	Travel_Frequently	884	Research & Development
1466	0	39	Travel_Rarely	613	Research & Development
1467	0	27	Travel_Rarely	155	Research & Development
1468	0	49	Travel_Frequently	1023	Sales
1469	0	34	Travel_Rarely	628	Research & Development

	DistanceFromHome	Education	EducationField	EmployeeCount \
0	1	2	Life Sciences	1
1	8	1	Life Sciences	1
2	2	2	Other	1
3	3	4	Life Sciences	1
4	2	1	Medical	1
...
1465	23	2	Medical	1
1466	6	1	Medical	1
1467	4	3	Life Sciences	1

1468	2	3	Medical	1
1469	8	3	Medical	1

	EmployeeNumber	...	RelationshipSatisfaction	StandardHours	\
0	1	...	1	80	
1	2	...	4	80	
2	4	...	2	80	
3	5	...	3	80	
4	7	...	4	80	
...	
1465	2061	...	3	80	
1466	2062	...	1	80	
1467	2064	...	2	80	
1468	2065	...	4	80	
1469	2068	...	1	80	

	StockOptionLevel	TotalWorkingYears	TrainingTimesLastYear	\
0	0	8	0	
1	1	10	3	
2	0	7	3	
3	0	8	3	
4	1	6	3	
...	
1465	1	17	3	
1466	1	9	5	
1467	1	6	0	
1468	0	17	3	
1469	0	6	3	

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

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

...
1465	0	3
1466	1	7
1467	0	3
1468	0	8
1469	1	2

[1470 rows x 35 columns]

```
In [7]: X=A[["Age","DailyRate","DistanceFromHome","Education","EmployeeCount","EmployeeNumber"]
Y=A[["Attrition"]]
```

```
from sklearn.model_selection import train_test_split
xtrain,xtest,ytrain,ytest=train_test_split(X,Y,test_size=0.2,random_state=30)

from sklearn.linear_model import LogisticRegression
lr=LogisticRegression()
model=lr.fit(xtrain,ytrain)
pred=model.predict(xtest)
ytest['predicted']=pred
```

C:\Users\abhijeet\Anaconda3\lib\site-packages\sklearn\utils\validation.py:578: DataConversionWarning: y = column_or_1d(y, warn=True)
C:\Users\abhijeet\Anaconda3\lib\site-packages\ipykernel_launcher.py:11: SettingWithCopyWarning: A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/index.html
This is added back by InteractiveShellApp.init_path()

```
In [8]: ytest
```

```
Out[8]:
```

	Attrition	predicted
461	0	0
640	0	0
509	0	0
788	0	0
950	0	0
...
1036	1	0
1339	1	0
956	0	0
575	0	0
1182	0	0

[294 rows x 2 columns]

```
In [9]: from sklearn.metrics import accuracy_score,confusion_matrix
accuracy_score(ytest['Attrition'],pred)
```


Out[9]: 0.8503401360544217