STEP 1: IMPORTING LIBRARIES

We have imported various libraires in this project

- Pandas: for reading and saving dataset and also to perform different operations on data like grouping, apply (), map (), info () etc.

- Numpy: we have also used numpy library for working with arrays.

- Seaborn and matplotlib: These are used for making visualization. We have created different types of graphs for visualization purpose like countplot , histogram , KDE, barplot , boxplot etc.

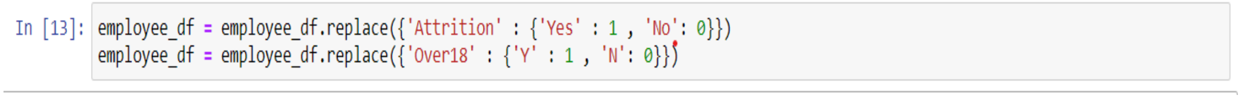
- Sklearn: This library provides many important functions for different models and different evaluation metrices.

STEP 2: DATA CLEANING

These are the different steps we have implemented in data cleaning.

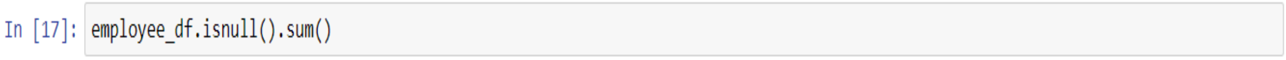
- Deleted the column “EmployeeID”, it will not make any sense to our analysis.

- Replacing “attrition” and “Over18” columns with integers.



```
In [13]: employee_df = employee_df.replace({'Attrition' : {'Yes' : 1 , 'No': 0}})
         employee_df = employee_df.replace({'Over18' : {'Y' : 1 , 'N': 0}})
```

- Dealing with missing value. We have three features in our dataset which have missing value NumCompaniesWorked , EnvironmentSatisfaction , WorkLifeBalance.

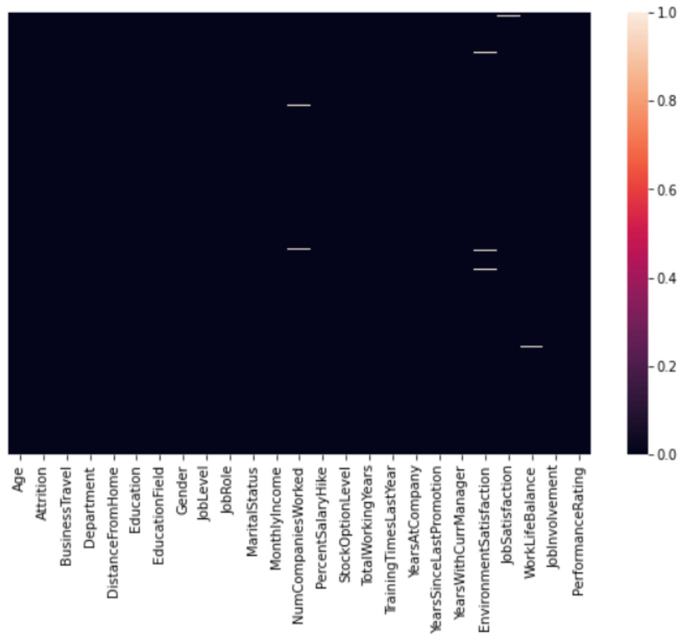


```
In [17]: employee_df.isnull().sum()
```

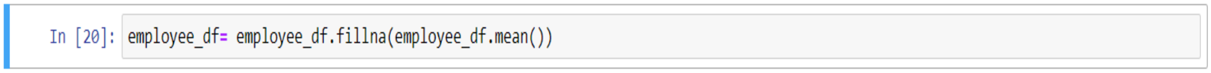
```
Age                         0
Attrition                   0
BusinessTravel              0
Department                  0
DistanceFromHome            0
Education                   0
EducationField              0
Gender                      0
JobLevel                    0
JobRole                     0
MaritalStatus               0
MonthlyIncome               0
NumCompaniesWorked         19
PercentSalaryHike           0
StockOptionLevel            0
TotalWorkingYears           9
TrainingTimesLastYear       0
YearsAtCompany              0
YearsSinceLastPromotion     0
YearsWithCurrManager        0
EnvironmentSatisfaction    25
JobSatisfaction            20
WorkLifeBalance            38
JobInvolvement              0
PerformanceRating           0
dtype: int64
```

In [28]:
```
plt.figure(figsize=(10,6))
sns.heatmap(employee_df.isnull(),yticklabels=False)
```

Out[28]:
```
<AxesSubplot:>
```

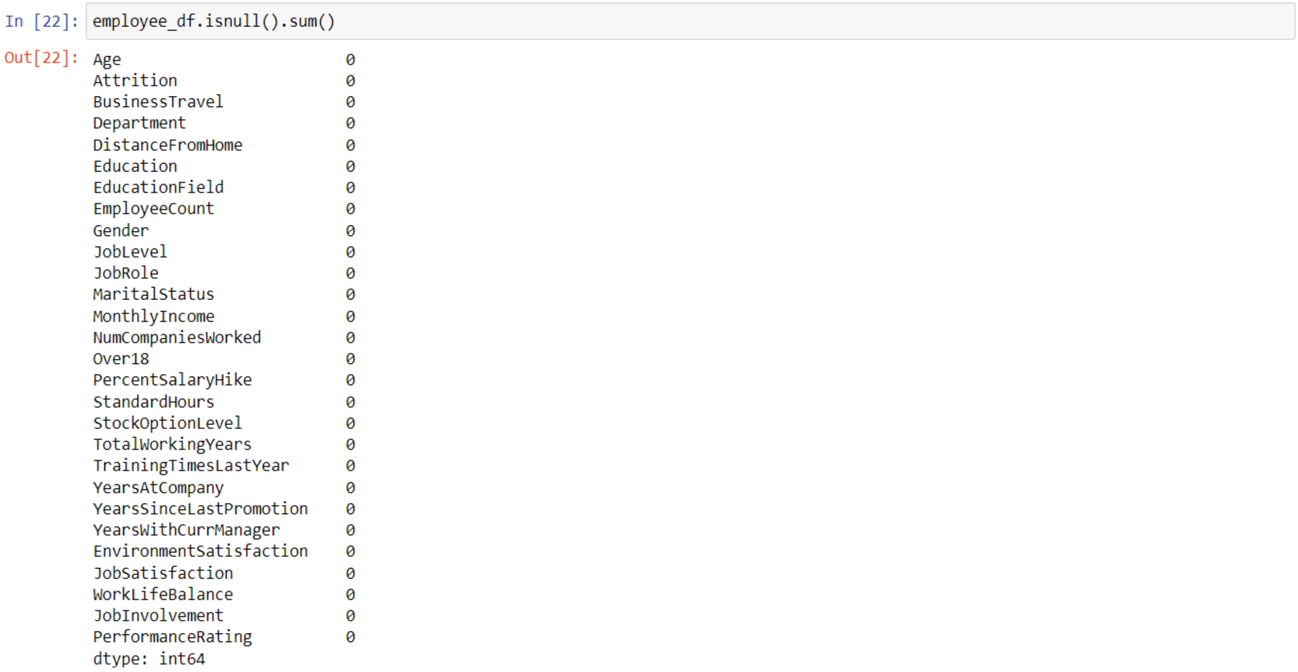


- Used fillna() method and replaced the missing values with mean of specific column.

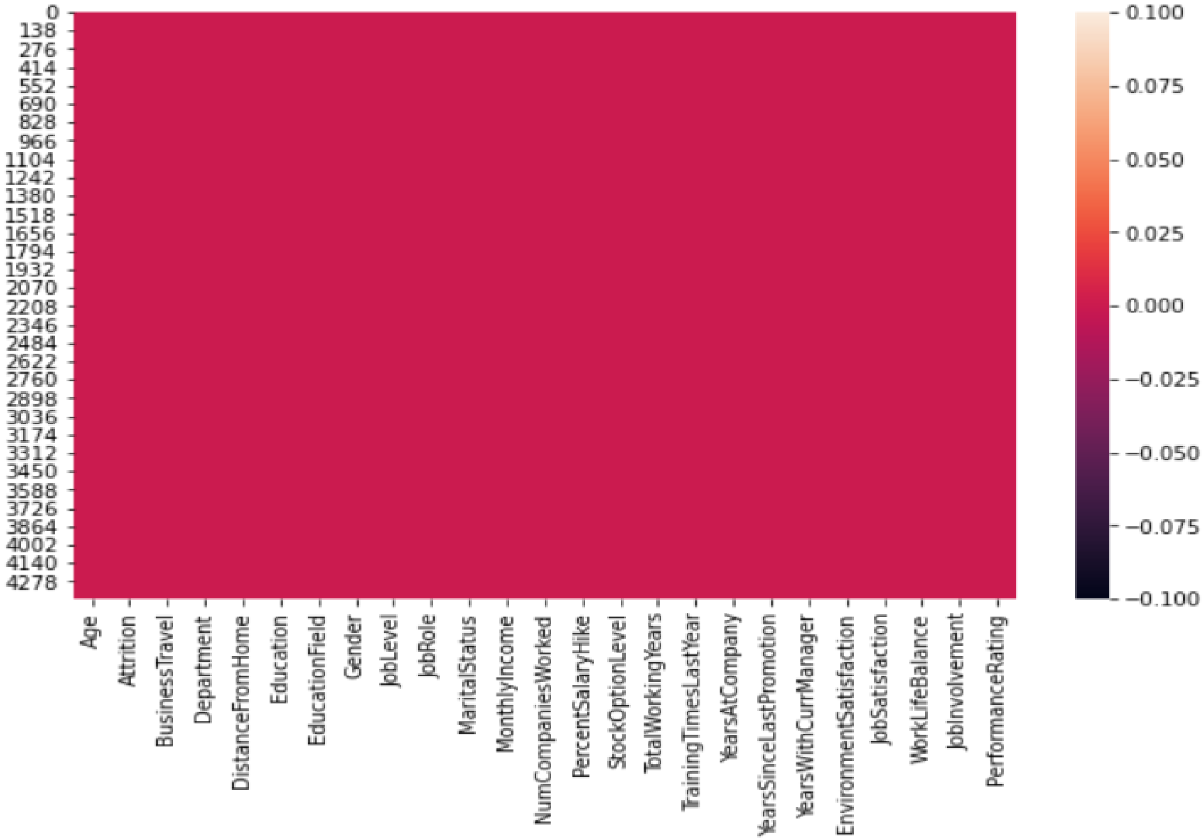


```
In [20]: employee_df= employee_df.fillna(employee_df.mean())
```

- Now all the missing values has been replaced.



```
In [22]: employee_df.isnull().sum()

Out[22]: Age                        0
         Attrition                  0
         BusinessTravel             0
         Department                 0
         DistanceFromHome           0
         Education                  0
         EducationField             0
         EmployeeCount              0
         Gender                     0
         JobLevel                   0
         JobRole                    0
         MaritalStatus              0
         MonthlyIncome              0
         NumCompaniesWorked         0
         Over18                     0
         PercentSalaryHike          0
         StandardHours              0
         StockOptionLevel           0
         TotalWorkingYears          0
         TrainingTimesLastYear      0
         YearsAtCompany             0
         YearsSinceLastPromotion    0
         YearsWithCurrManager       0
         EnvironmentSatisfaction    0
         JobSatisfaction            0
         WorkLifeBalance            0
         JobInvolvement             0
         PerformanceRating          0
         dtype: int64
```

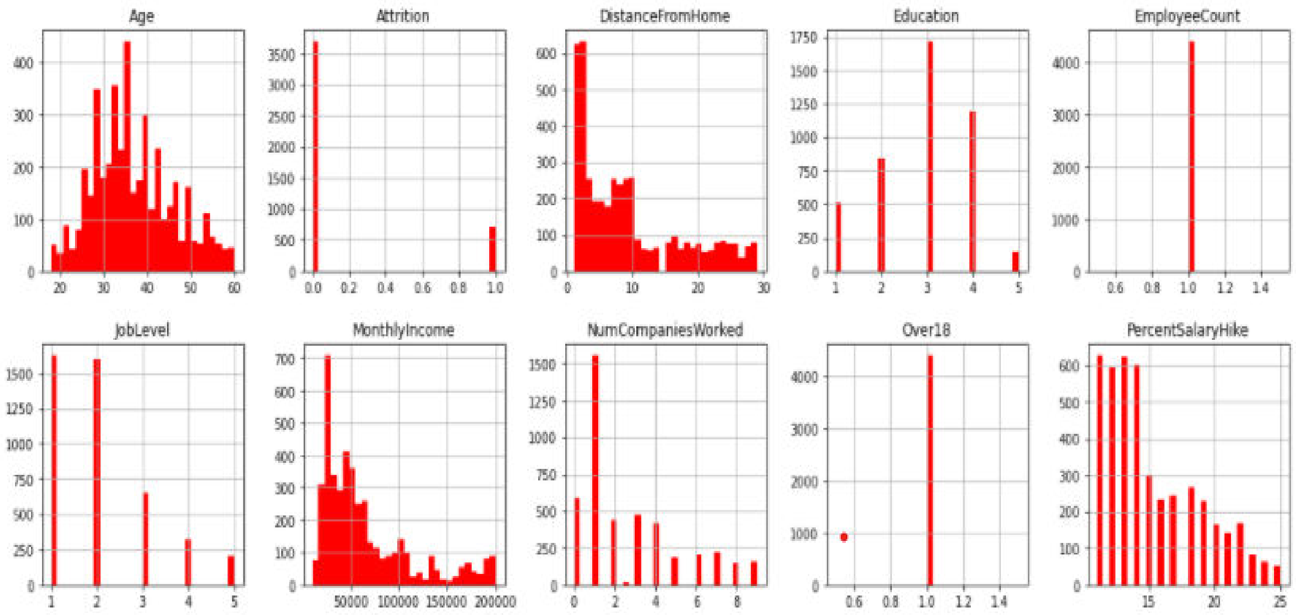


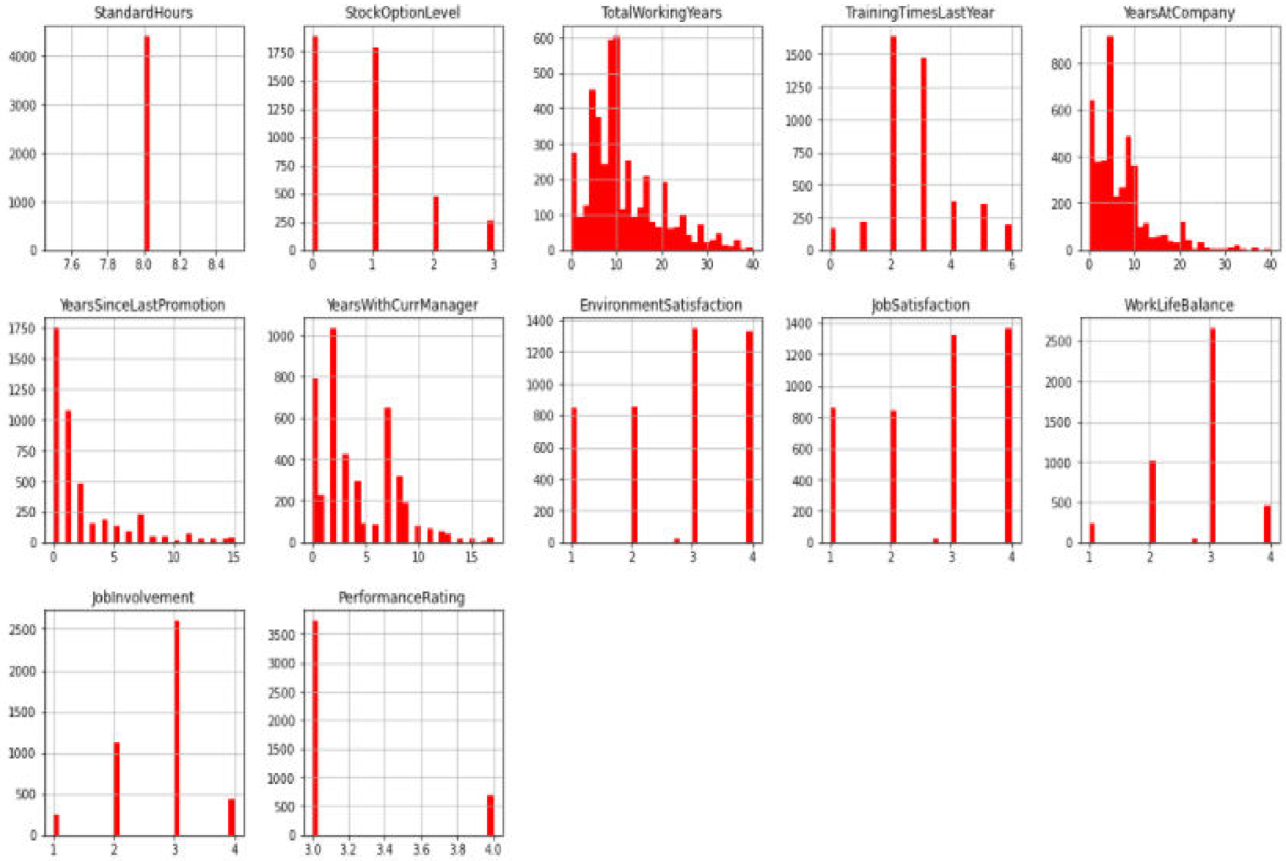
- The data is arranged and visualization is done.
- Various questions related to problem statement are answered.
- The visualization of correlation and heatmap is done
- Using Boxplot, the outliers are identified
- Graphical representation of various features v/s attrition rate is plotted.

A detailed report of exploratory data analysis (EDA) has been shown below.

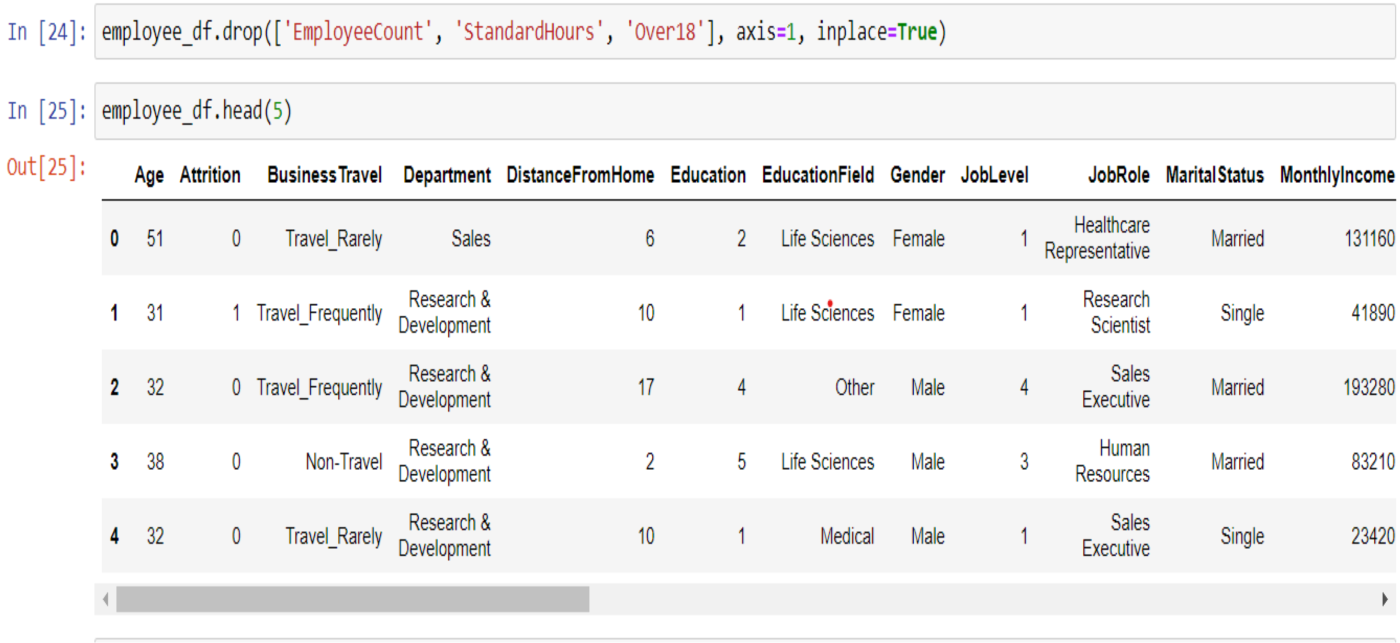
Exploratory data analysis process was done to gather a better understanding of the data that we had.

1. We can see from the below histogram that several features such as “MonthlyIncome” and “TotalWorkingYears” are tail heavy and also it makes sense to drop “EmployeeCount”, “Standardhours” and “Over18” since they do not change from one employee to the other.





- So dropping the “Employeecount” , “Standardhours” , and “ Over18”.



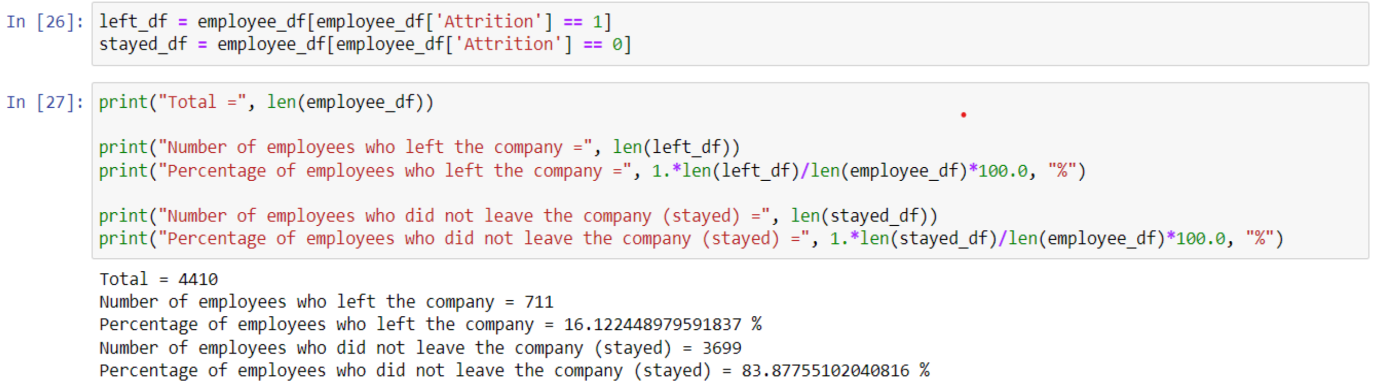
```
In [24]: employee_df.drop(['EmployeeCount', 'StandardHours', 'Over18'], axis=1, inplace=True)
```

```
In [25]: employee_df.head(5)
```

Out[25]:

	Age	Attrition	BusinessTravel	Department	DistanceFromHome	Education	EducationField	Gender	JobLevel	JobRole	MaritalStatus	MonthlyIncome
0	51	0	Travel_Rarely	Sales	6	2	Life Sciences	Female	1	Healthcare Representative	Married	131160
1	31	1	Travel_Frequently	Research & Development	10	1	Life Sciences	Female	1	Research Scientist	Single	41890
2	32	0	Travel_Frequently	Research & Development	17	4	Other	Male	4	Sales Executive	Married	193280
3	38	0	Non-Travel	Research & Development	2	5	Life Sciences	Male	3	Human Resources	Married	83210
4	32	0	Travel_Rarely	Research & Development	10	1	Medical	Male	1	Sales Executive	Single	23420

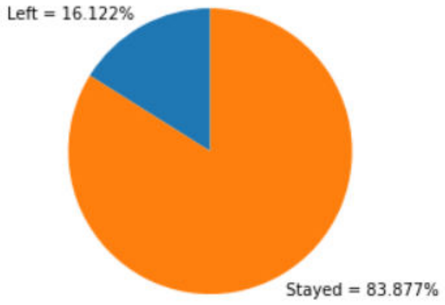
2. Let’s see how many employees left the company!



In [26]:
```
left_df = employee_df[employee_df['Attrition'] == 1]
stayed_df = employee_df[employee_df['Attrition'] == 0]
```

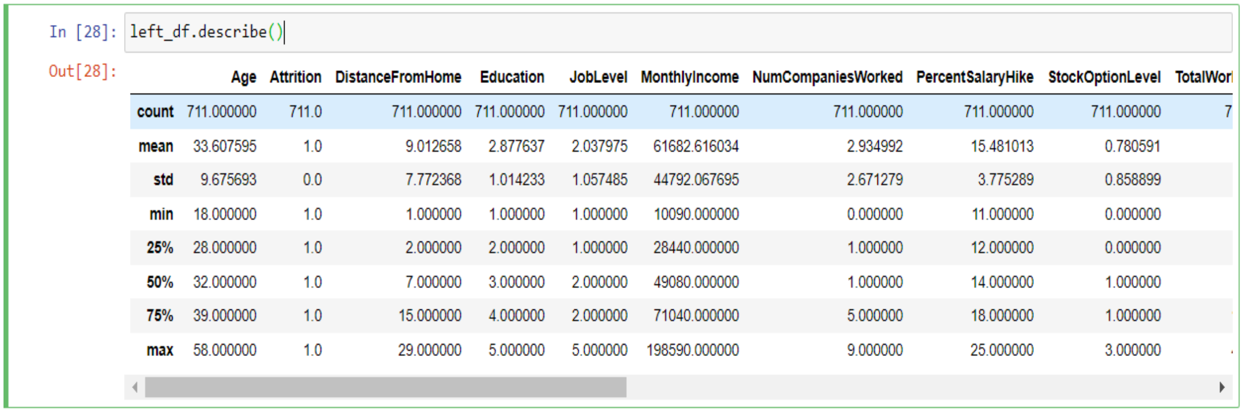
In [27]:
```
print("Total =", len(employee_df))

print("Number of employees who left the company =", len(left_df))
print("Percentage of employees who left the company =", 1.*len(left_df)/len(employee_df)*100.0, "%")

print("Number of employees who did not leave the company (stayed) =", len(stayed_df))
print("Percentage of employees who did not leave the company (stayed) =", 1.*len(stayed_df)/len(employee_df)*100.0, "%")
```

```
Total = 4410
Number of employees who left the company = 711
Percentage of employees who left the company = 16.122448979591837 %
Number of employees who did not leave the company (stayed) = 3699
Percentage of employees who did not leave the company (stayed) = 83.87755102040816 %
```



So, almost 16% employees left the organization.

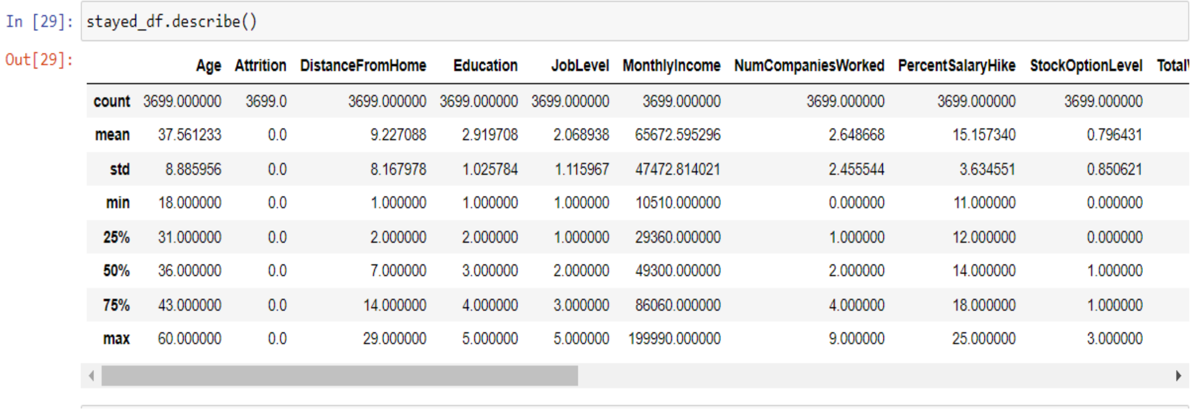
3. Let’s compare the mean and standard deviation of the employees who stayed and left.



In [28]:
```
left_df.describe()
```

Out[28]:

	Age	Attrition	DistanceFromHome	Education	JobLevel	MonthlyIncome	NumCompaniesWorked	PercentSalaryHike	StockOptionLevel	TotalWor
count	711.000000	711.0	711.000000	711.000000	711.000000	711.000000	711.000000	711.000000	711.000000	7
mean	33.607595	1.0	9.012658	2.877637	2.037975	61682.616034	2.934992	15.481013	0.780591	
std	9.675693	0.0	7.772368	1.014233	1.057485	44792.067695	2.671279	3.775289	0.858899	
min	18.000000	1.0	1.000000	1.000000	1.000000	10090.000000	0.000000	11.000000	0.000000	
25%	28.000000	1.0	2.000000	2.000000	1.000000	28440.000000	1.000000	12.000000	0.000000	
50%	32.000000	1.0	7.000000	3.000000	2.000000	49080.000000	1.000000	14.000000	1.000000	
75%	39.000000	1.0	15.000000	4.000000	2.000000	71040.000000	5.000000	18.000000	1.000000	
max	58.000000	1.0	29.000000	5.000000	5.000000	198590.000000	9.000000	25.000000	3.000000	



In [29]:
```
stayed_df.describe()
```

Out[29]:

	Age	Attrition	DistanceFromHome	Education	JobLevel	MonthlyIncome	NumCompaniesWorked	PercentSalaryHike	StockOptionLevel	Total
count	3699.000000	3699.0	3699.000000	3699.000000	3699.000000	3699.000000	3699.000000	3699.000000	3699.000000	
mean	37.561233	0.0	9.227088	2.919708	2.068938	65672.595296	2.648668	15.157340	0.796431	
std	8.885956	0.0	8.167978	1.025784	1.115967	47472.814021	2.455544	3.634551	0.850621	
min	18.000000	0.0	1.000000	1.000000	1.000000	10510.000000	0.000000	11.000000	0.000000	
25%	31.000000	0.0	2.000000	2.000000	1.000000	29360.000000	1.000000	12.000000	0.000000	
50%	36.000000	0.0	7.000000	3.000000	2.000000	49300.000000	2.000000	14.000000	1.000000	
75%	43.000000	0.0	14.000000	4.000000	3.000000	86060.000000	4.000000	18.000000	1.000000	
max	60.000000	0.0	29.000000	5.000000	5.000000	199990.000000	9.000000	25.000000	3.000000	

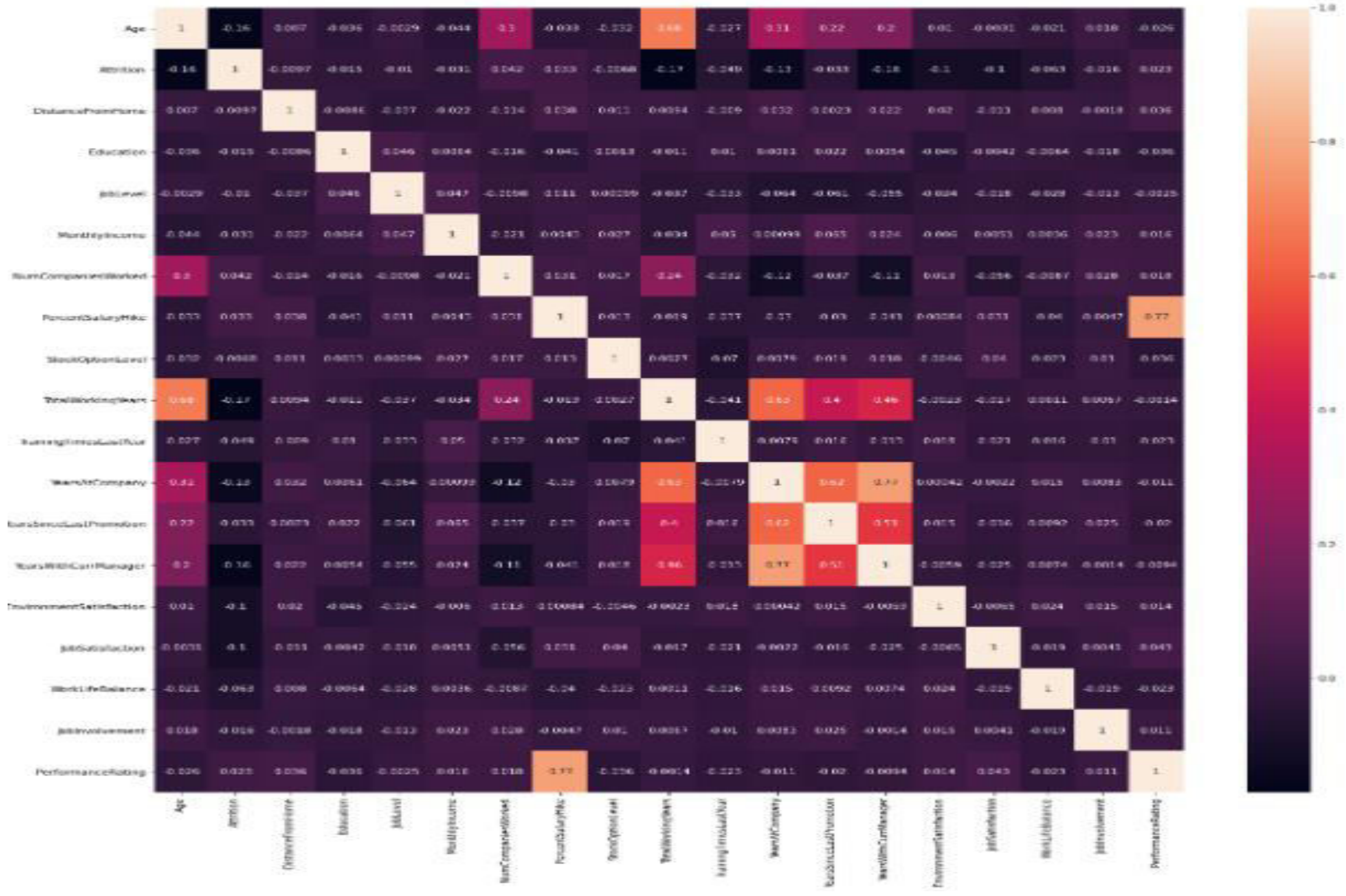
- 'age': mean age of the employees who stayed is higher compared to who left.
- 'DistanceFromHome': Employees who stayed live closer to home.
- 'EnvironmentSatisfaction' & 'JobSatisfaction': Employees who stayed are generally more satisifed with their jobs.
- 'StockOptionLevel': Employees who stayed tend to have higher stock option level.

4. Correlation

```
In [30]: correlations = employee_df.corr()
         f, ax = plt.subplots(figsize = (20, 20))
         sns.heatmap(correlations, annot = True)
```

From the correlation plot that we have shown in the next page, we can draw the following insights.

- Job level is strongly correlated with total working hours.
- Monthly income is strongly correlated with Job level.
- Monthly income is strongly correlated with total working hours.
- Age is strongly correlated with monthly income.

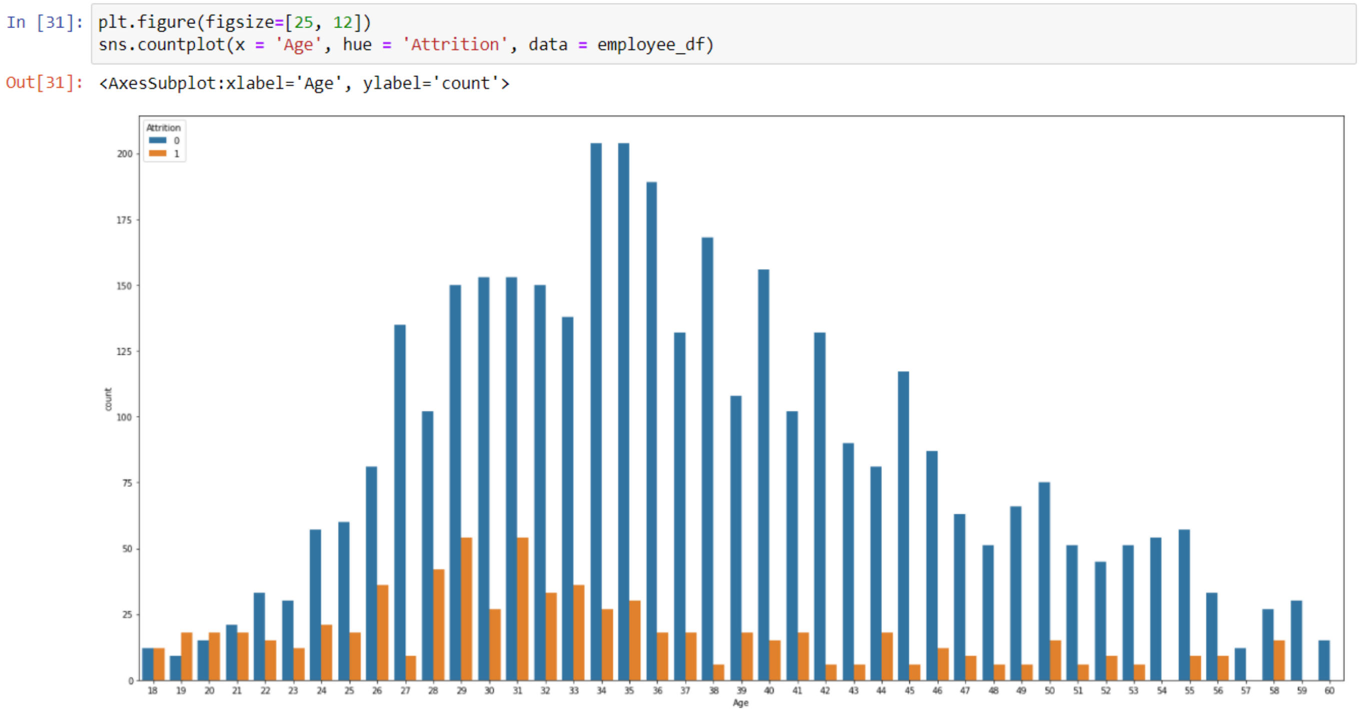


STEP 4: DATA VISUALIZATION

After digging more into the data, we got below findings.

1. Age vs Attrition analysis:

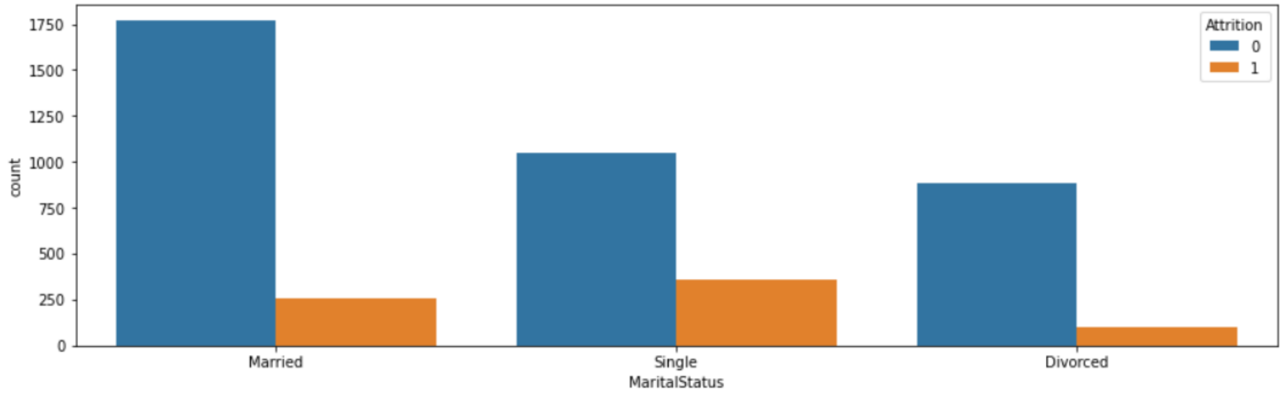
People of age of 29 and 31 years left the company more frequently. Although the number of employees in age group of 18 to 23 is less but the attrition rate is also high in this group. Also, as age increases the chances of leaving the company decreases.



```
In [31]: plt.figure(figsize=[25, 12])
         sns.countplot(x = 'Age', hue = 'Attrition', data = employee_df)

Out[31]: <AxesSubplot:xlabel='Age', ylabel='count'>
```

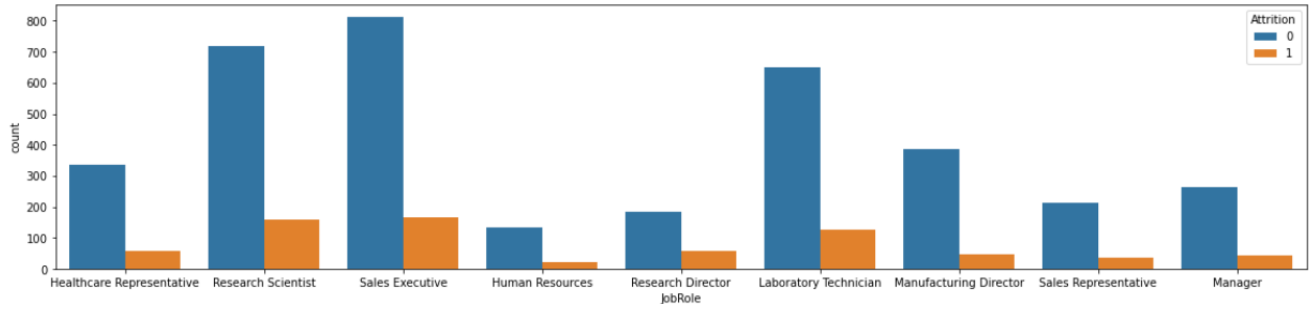
2. Marital Status vs Attrition:

Single employees tend to leave compared to married and divorced.



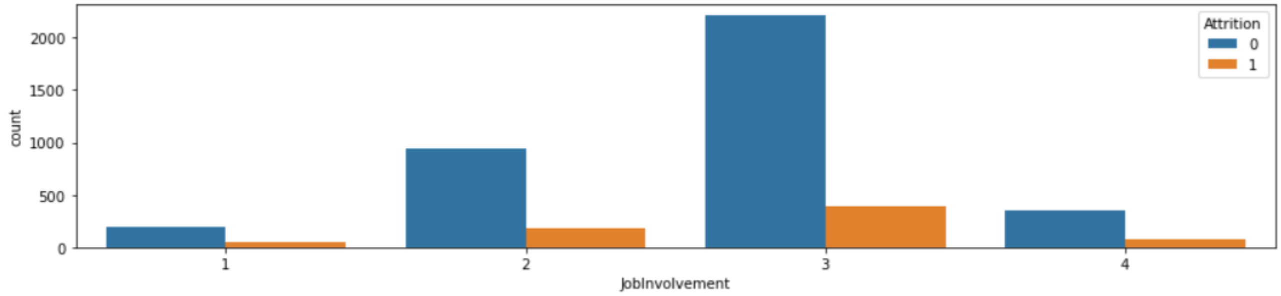
3. Job Role vs Attrition:

Sales Executive and Lab Technician tend to leave compared to any other job.



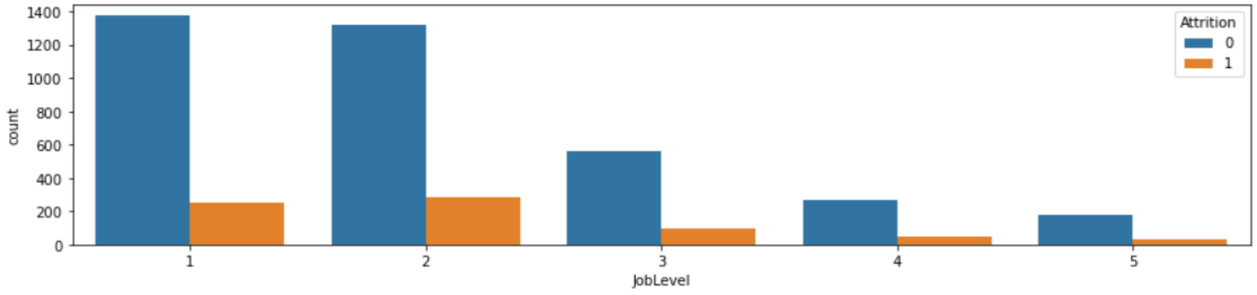
4. Job Involvement vs Attrition:

Less involved employees tend to leave the company.



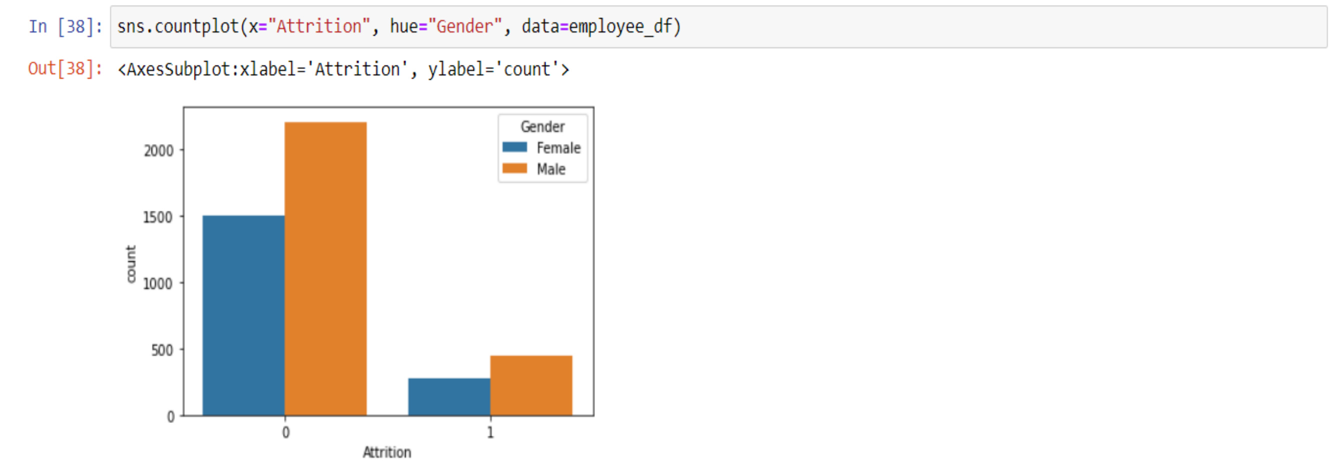
5. Experienced vs Attrition:

Less experienced (low job level i.e., JobLevel=1) tend to leave the company.



6. Gender vs Attrition:

Male tend to leave the company compared to Female.

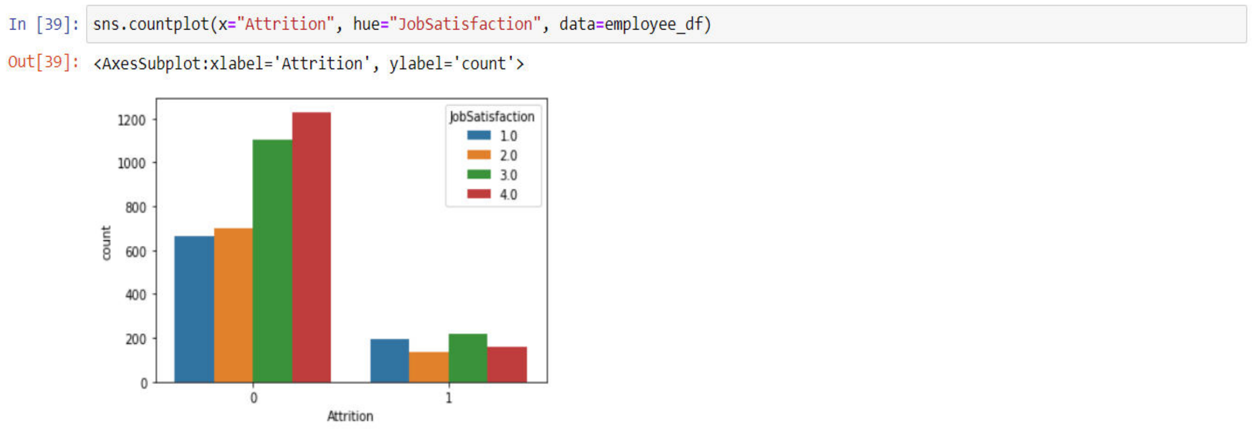


```
In [38]: sns.countplot(x="Attrition", hue="Gender", data=employee_df)
```

```
Out[38]: <AxesSubplot:xlabel='Attrition', ylabel='count'>
```

7. Job Satisfaction vs Attrition:

Employee who has lower Job Satisfaction level tend to leave the company compared to others.

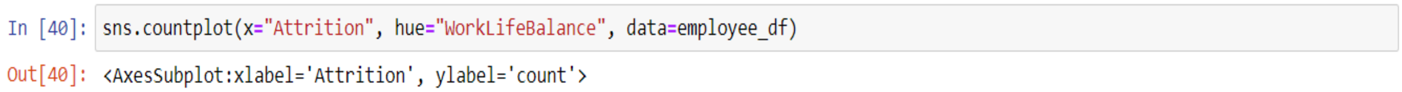


```
In [39]: sns.countplot(x="Attrition", hue="JobSatisfaction", data=employee_df)
```

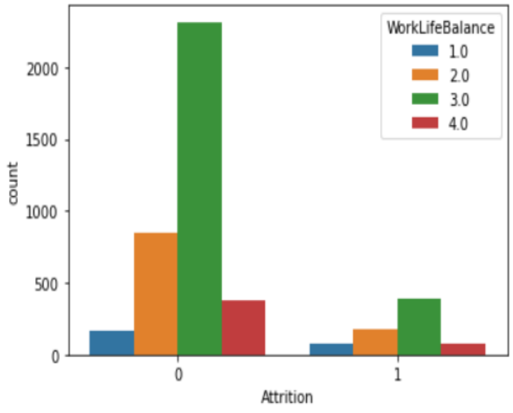
```
Out[39]: <AxesSubplot:xlabel='Attrition', ylabel='count'>
```

8. Work Life Balance vs Attrition:

People who have Bad Work Life Balance tend to leave the company compared to others.



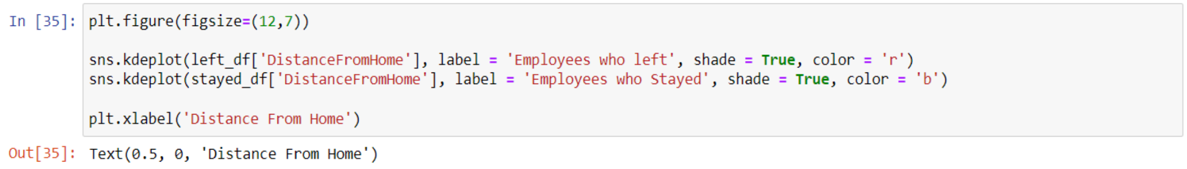
```
In [40]: sns.countplot(x="Attrition", hue="WorkLifeBalance", data=employee_df)

Out[40]: <AxesSubplot:xlabel='Attrition', ylabel='count'>
```



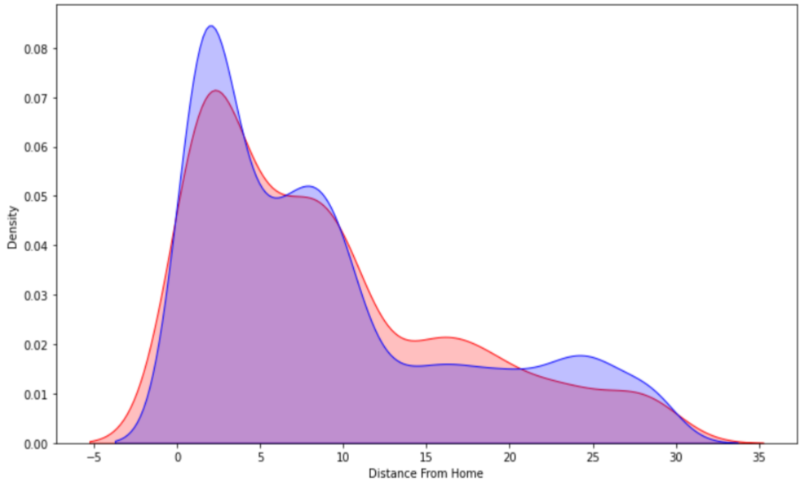
Now using KDE (Kernel Density Estimate), it is used for visualizing the Probability Density of a continuous variable and it describes the probability density at different values in a continuous variable.

9. Distance from Home vs Attrition:

People staying far (more than 10km) from office more likely to leave company. We can notice the red line is above the blue line after 10 in the x-axis i.e. Distance from Home.

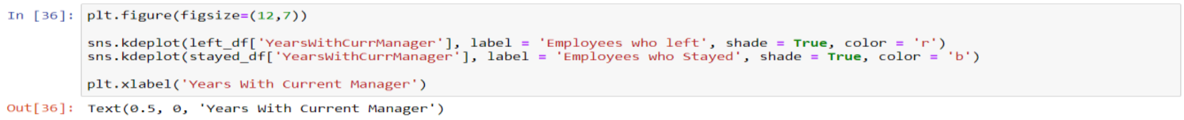


```
In [35]: plt.figure(figsize=(12,7))

         sns.kdeplot(left_df['DistanceFromHome'], label = 'Employees who left', shade = True, color = 'r')
         sns.kdeplot(stayed_df['DistanceFromHome'], label = 'Employees who Stayed', shade = True, color = 'b')

         plt.xlabel('Distance From Home')

Out[35]: Text(0.5, 0, 'Distance From Home')
```

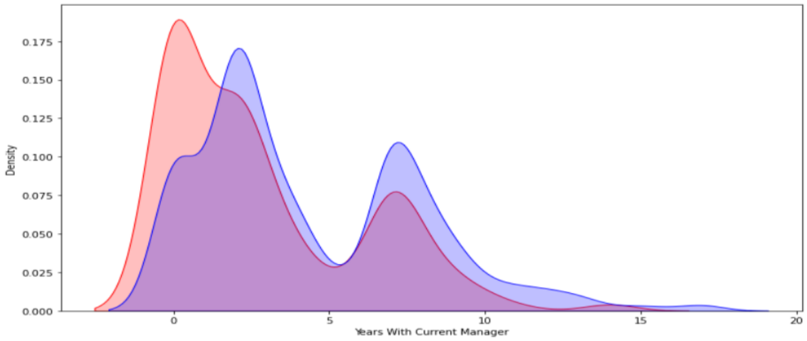


10. Years with Current manager vs Attrition:

Employee with small span of time with Current manager are more likely to leave the company. We can notice the red line is above the blue line at the starting of x-axis i.e., Years with Current manager. However, as we increase the number of years, the blue line tends to supersede the red line, which means that as we go beyond 4 to 15 years, the number of employees who actually tend to stay is more than the number of employees who actually leaves the company.

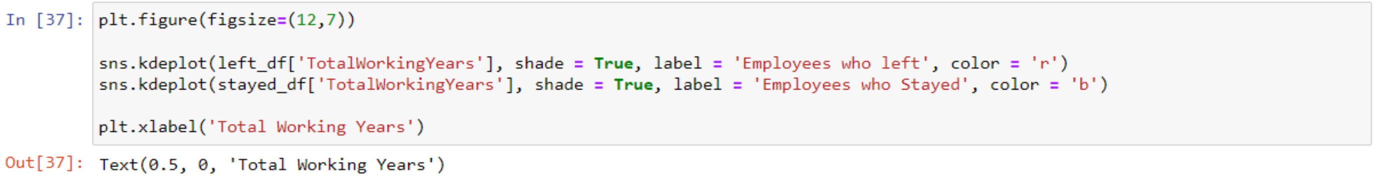


```
In [36]: plt.figure(figsize=(12,7))

         sns.kdeplot(left_df['YearsWithCurrManager'], label = 'Employees who left', shade = True, color = 'r')
         sns.kdeplot(stayed_df['YearsWithCurrManager'], label = 'Employees who Stayed', shade = True, color = 'b')

         plt.xlabel('Years With Current Manager')
Out[36]: Text(0.5, 0, 'Years With Current Manager')
```

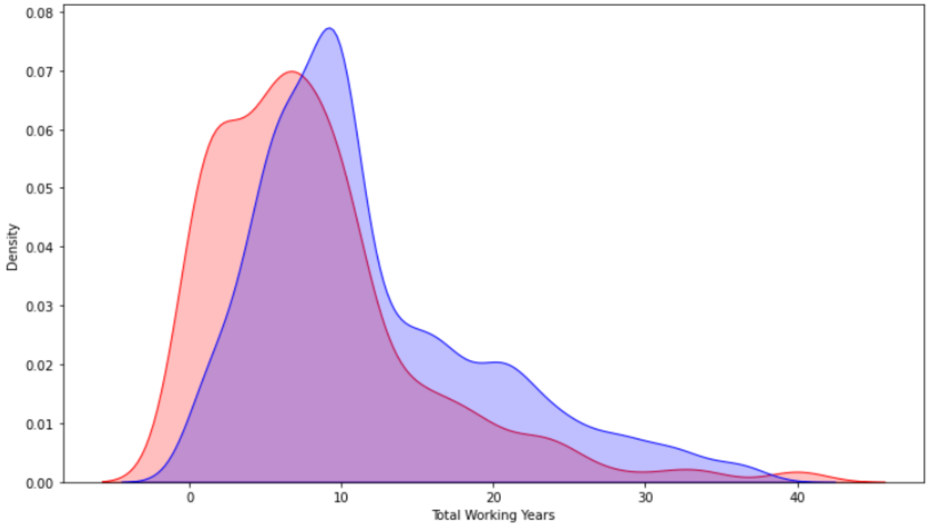


11. Total Working Years vs Attrition:

Employees with a smaller number of years (0 to 6 years) with the company tend to leave the company. We can notice the red line is above blue line at the starting of x-axis i.e., Total Working Years. However, as we go beyond 6 years, we will find that the blue line tends to supersede which means the employees tend to stay as we increase the total working years.

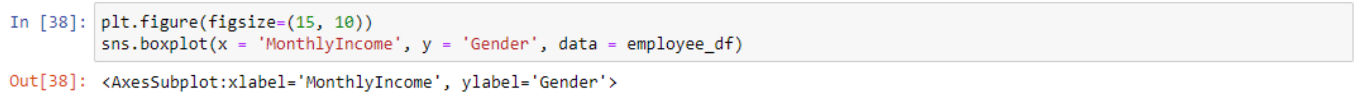


```
In [37]: plt.figure(figsize=(12,7))

         sns.kdeplot(left_df['TotalWorkingYears'], shade = True, label = 'Employees who left', color = 'r')
         sns.kdeplot(stayed_df['TotalWorkingYears'], shade = True, label = 'Employees who Stayed', color = 'b')

         plt.xlabel('Total Working Years')
Out[37]: Text(0.5, 0, 'Total Working Years')
```

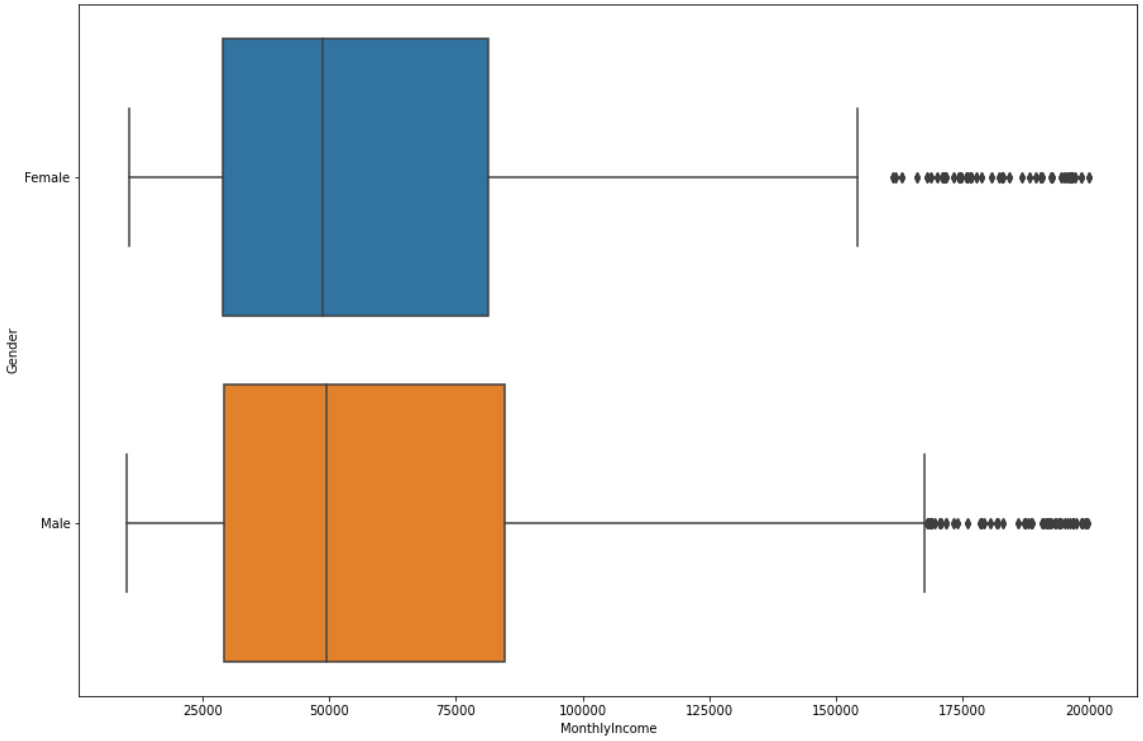


12. Gender vs Monthly Income:

We can see that the average salary is almost quite comparable between male and female, that’s actually a great thing. Gender pay equality is actually critical and very important thing for any company.



```
In [38]: plt.figure(figsize=(15, 10))
         sns.boxplot(x = 'MonthlyIncome', y = 'Gender', data = employee_df)

Out[38]: <AxesSubplot:xlabel='MonthlyIncome', ylabel='Gender'>
```



Using Logistic Regression as the analytical technique to model the probability of attrition:

Step 1

First, determine the binary separation. In order to do so, we first determine the best fitted line by following the linear regression steps.

Step 2

The regression line we get from Linear Regression is highly susceptible to outliers. Thus, it will not do a good job in classifying two classes.

Since the line obtained from the leading and regression is susceptible to outliers. Does in order to have the good probability we feed predicted values to the sigmoid function and get it converted to probability.

The equation of sigmoid:

$$S(x) = \frac{1}{1+e^{-x}}$$

Step 3

And finally, the output from this sigmoid function gets converted into 0 or1 based on the threshold value we have provided. And we get the binary classification.

Conclusion Result Summary is as below:

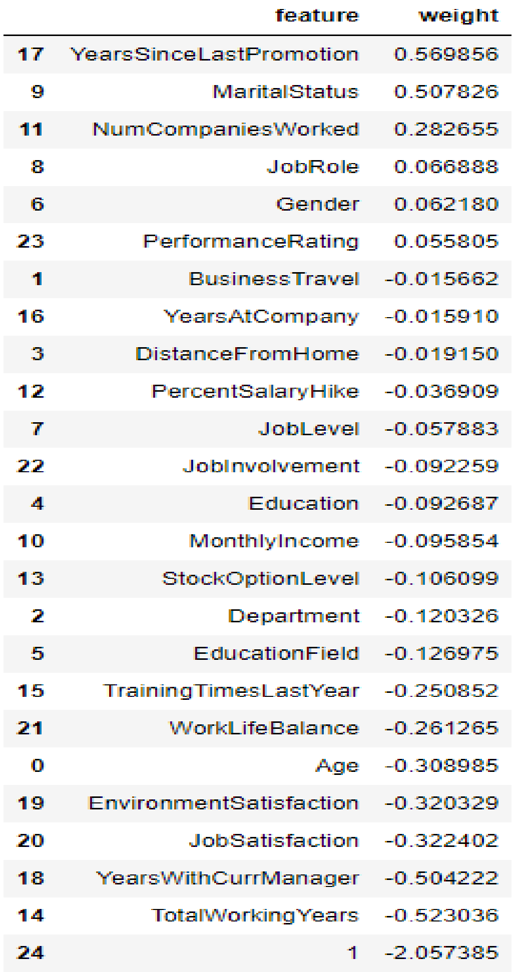
Experiment 1: Scaled data only

Support Vector Machine 66.36 Decision Tree 84.497 Linear Discriminant Analysis 85.041 KNearest Neighbors 85.857 Gaussian Naivey Bayes 83.046 **Logistic Regression: 84.95**

So, after applying the logistic regression model, we get following:

```
Accuracy 84.95013599274705 %
```

So, the model is able to predict attrition rate with 84.95% accuracy.



	feature	weight
17	YearsSinceLastPromotion	0.569856
9	MaritalStatus	0.507826
11	NumCompaniesWorked	0.282655
8	JobRole	0.066888
6	Gender	0.062180
23	PerformanceRating	0.055805
1	BusinessTravel	-0.015662
16	YearsAtCompany	-0.015910
3	DistanceFromHome	-0.019150
12	PercentSalaryHike	-0.036909
7	JobLevel	-0.057883
22	JobInvolvement	-0.092259
4	Education	-0.092687
10	MonthlyIncome	-0.095854
13	StockOptionLevel	-0.106099
2	Department	-0.120326
5	EducationField	-0.126975
15	TrainingTimesLastYear	-0.250852
21	WorkLifeBalance	-0.261265
0	Age	-0.308985
19	EnvironmentSatisfaction	-0.320329
20	JobSatisfaction	-0.322402
18	YearsWithCurrManager	-0.504222
14	TotalWorkingYears	-0.523036
24	1	-2.057385

Conclusion

It is evident from the model that the major factor contributing to the increase in attrition rate are linked to increase years since last promotion, marital status, number of companies worked.

Also, it can be seen that with increase in following factor will actually result in decrease in attrition rate: age, Environment satisfaction, job satisfaction, Years with current manager and Total working years.

Suggestion

Team management should focus on streamlining the promotion and look for years since last promotion of an employee as it delays can increase the attrition rate.

Before, employing the company should look for number of companies the current candidate has worked with. And preferably, the lower the better.

Also, the company should consider that the employee work under the same manager for a long period of time to get proper mentorship, as it is seen that it increases the number of years employees served in a company.

It is also seen that more the work experience lesser will be the chance of employee leaving the company and it should also be considered during the hiring criteria of a candidate.

The company should also focus on increasing the job satisfaction level, have an arrangement for experiential sharing and peer acknowledgement to increase the job satisfaction level.

----------------x--------------