untitled2

August 24, 2023

1 What is HR analytics?

Human resource analytics is an area in the field of analytics that refers to applying analytic processes to the human resource department of an organization in the hope of improving employee performance and therefore getting a better return on investment. HR analytics does not just deal with gathering data on employee efficiency. Instead, it aims to provide insight into each process by gathering data and then using it to make relevant decisions about how to improve these processes.

2 What is attrition in business?

Attrition in business describes a gradual but deliberate reduction of staff numbers that occurs as employees retire or resign and are not replaced. The term is also sometimes used to describe the loss of customers or clients as they mature beyond a product or company's target market without being replaced by a younger generation

3 How attrition affect a company?

A major problem in high employee attrition is its cost to an organization. Job postings, hiring processes, paperwork and new hire training are some of the common expenses of losing employees and replacing them. Additionally, regular employee turnover prohibits your organization from increasing its collective knowledge base and experience over time. This is especially concerning if your business is customer facing, as customers often prefer to interact with familiar people. Errors and issues are more likely if you constantly have new workers

4 What are Positive and Negative attrition

5 Positive attrition:

Positive attrition refers to staff turnover that actually benefits the organization. Think of an employee who is a poor performer, makes many errors, has difficulty working with others, delivers low quality customer service and/or uses sick leave and vacation time as the hours are earned. When the employee quits, the organization benefits because now the supervisor can replace the low performer employee with someone who is better for the organization

6 Negative attrition:

Negative attrition refers to the loss of an employee the organization would like to keep. Qualified and skilled employees leave for a variety of reasons, and it is often challenging to find an equally skilled replacement. Negative attrition, especially in industries with the highest turnover rates, is expensive. The organization must once again recruit, assess, hire and train a new employee, and until the position is filled, team productivity declines

7 Our Objectives:

1. • Study the HR employee attrition data to identify the patters and causes of attrition with respect to various parameters.
2. • Identify the important parameter and generate helpful insights from them.
3. • Build model to predict if the employee is unsatisfied and will resign or is satisfied and will stay.
4. • Compare the parameters of a satisfied and an unsatisfied employee to come up with idea of what can be improved.
5. • Identify future attrition early so that proper measures can be taken on time

To perform this analysis,we need to follow these general steps: we will be importing necessary libraries required for data preprocessing and visualisation,It provide us with a set of functions, classes, and tools that we can use to perform various data processing and visualization tasks. these libraries don't perform tasks automatically upon import, they significantly simplify the process of working with data and creating visual representations of that data.

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

Uploaded the csv file using google colab upload file statement to access it accordingly and further read the file from it

[2]:
```
from google.colab import files
uploaded = files.upload()
```

```
<IPython.core.display.HTML object>
```

```
Saving IBM.csv to IBM.csv
```

[3]:
```
data = pd.read_csv('IBM.csv')
```

The data.head() function displays the first few rows of the dataset to give a glimpse of its contents and then got the exact no. of rows and columns from data file.

[4]:
```
data.head()
```

[4]:
```
   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]
```

[5]:
```
print('Total number of rows:',data.shape[0], 'and columns:', data.shape[1])
```

```
Total number of rows: 1470 and columns: 35
```

[6]:
```
data.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
```

WE WILL BE DOING THE FEATURE SELECTION: Choose the relevant features (columns) that are likely to impact attrition. Common features might include age, job role, years of experience, salary, performance ratings, work-life balance, etc.

Their are 26 Numerical Variables and 9 Cateogarical variables according to the above info.

[7]:
```python
# Looking into Numerical Features
data.describe(include = 'int64')
```

[7]:
```
               Age    DailyRate  DistanceFromHome    Education  EmployeeCount  \
count  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]
```

[8]:
```python
# Looking into the Categorical Features
data.describe(include='object')
```

[8]:
```
       Attrition BusinessTravel              Department EducationField Gender  \
count       1470           1470                    1470           1470   1470
unique         2              3                       3              6      2
top           No  Travel_Rarely  Research & Development  Life Sciences   Male
freq        1233           1043                     961            606    882

                JobRole MaritalStatus Over18 OverTime
count              1470          1470   1470     1470
unique                9             3      1        2
top     Sales Executive       Married      Y       No
freq                326           673   1470     1054
```

[9]:
```python
#looking down to some employee features
data[['DailyRate','HourlyRate','MonthlyRate']].describe()
```

[9]:
```
         DailyRate   HourlyRate   MonthlyRate
count  1470.000000  1470.000000   1470.000000
mean    802.485714    65.891156  14313.103401
std     403.509100    20.329428   7117.786044
min     102.000000    30.000000   2094.000000
25%     465.000000    48.000000   8047.000000
50%     802.000000    66.000000  14235.500000
75%    1157.000000    83.750000  20461.500000
max    1499.000000   100.000000  26999.000000
```

[10]:
```python
# Counting missing values
pd.DataFrame({'Count':data.isnull().sum()})
```

[10]:
```
                          Count
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
```

[11]:
```
# Removing insignificant columns
data.drop(['EmployeeCount','Over18', 'StandardHours'],axis=1,inplace=True)
```

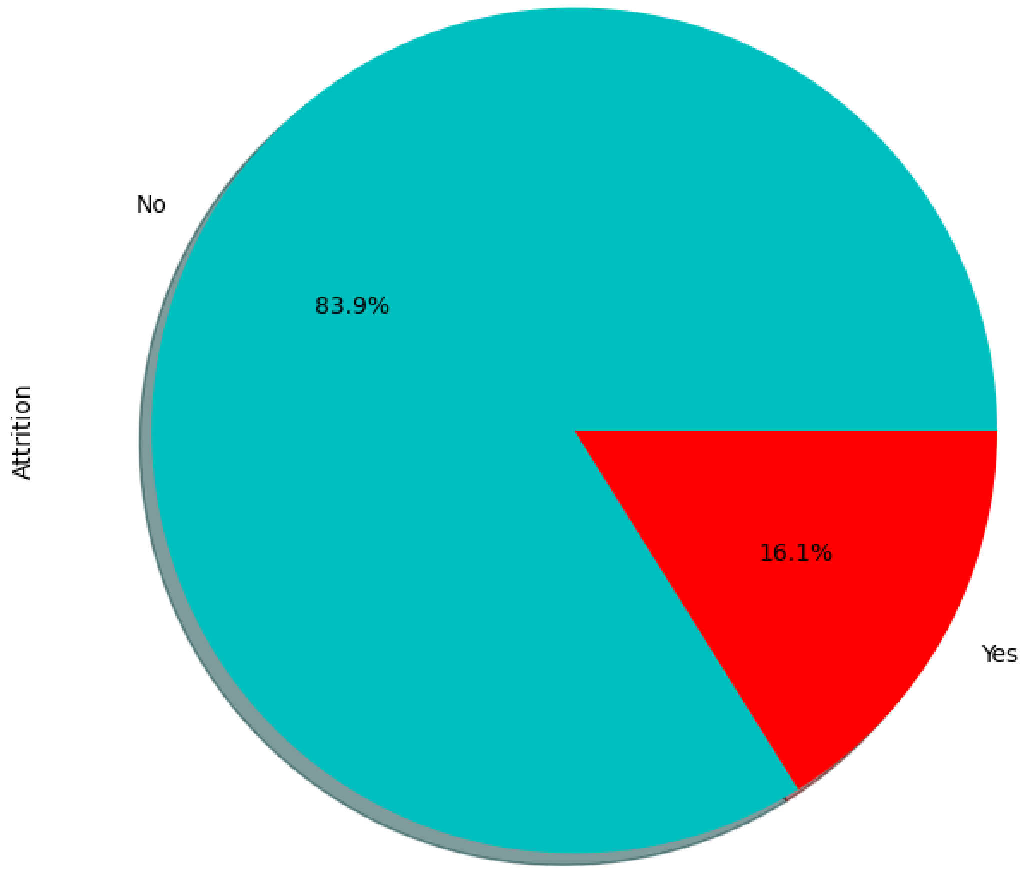
8 Feature Analysis

[12]:
```
data.hist(figsize=(28,24))
plt.show()
```

[13]:

```
plt.figure(figsize=(20,8))
data['Attrition'].value_counts().plot(kind='pie',autopct='%1.
 ↪1f%%',shadow=True,colors=['c','r'])
print(data['Attrition'].value_counts())
```

```
No      1233
Yes      237
Name: Attrition, dtype: int64
```



[14]:
```
#The numerical columns with high skewness
for i in data.select_dtypes(exclude='O'):
    if data[i].skew() > 0.9:
        print(i, ':', data[i].skew())
```

```
DistanceFromHome : 0.9581179956568269
JobLevel : 1.0254012829518246
MonthlyIncome : 1.3698166808390662
NumCompaniesWorked : 1.026471111968205
PerformanceRating : 1.921882702142603
StockOptionLevel : 0.9689803167738937
TotalWorkingYears : 1.1171718528128527
YearsAtCompany : 1.7645294543422085
YearsInCurrentRole : 0.9173631562908262
```

```
YearsSinceLastPromotion : 1.9842899833524859
```

9 Exploratory Data Analysis (EDA):

Exploring the data to gain a preliminary understanding of its characteristics.It Create visualizations using libraries like matplotlib and seaborn to analyze trends, correlations, and distributions. This step can help to identify potential factors contributing to attrition.(Data Understanding,Data Quality Check, Pattern Identification, Feature Selection, Model Assumptions, Outlier Detection, Data Transformation, Effective Visualization, Hypothesis Generation, Validation of Assumptions, Enhanced Decision Making)

10 1. GENDER vs ATTRITION

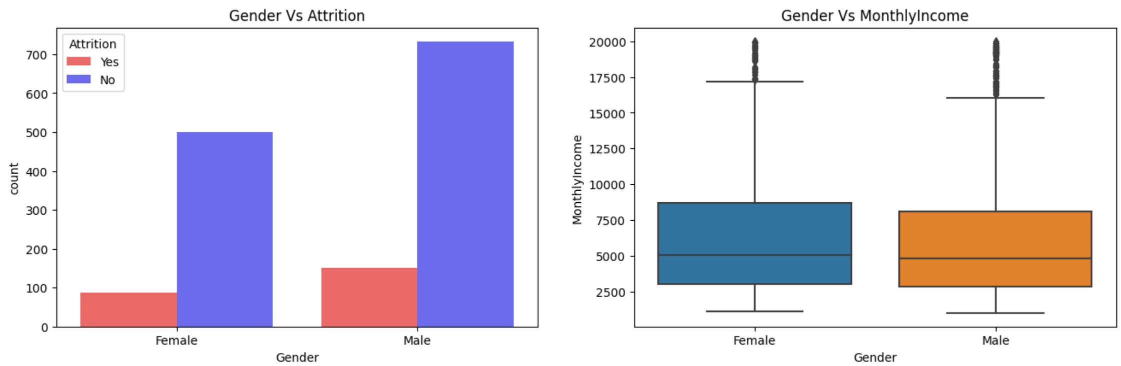
[15]:
```
#comparision with attrition
pd.crosstab(data['Attrition'],data['Gender'])
```

[15]:
```
Gender     Female  Male
Attrition
No            501   732
Yes            87   150
```

[16]:
```
#comparision with MonthlyIncome
pd.pivot_table␣
 ↪(data=data,index=['Gender'],values=['MonthlyIncome'],aggfunc='mean')
```

[16]:
```
        MonthlyIncome
Gender
Female    6686.566327
Male      6380.507937
```

[17]:
```
plt.figure(figsize=(16,10))
plt.subplot(221)
plt.title('Gender Vs Attrition')
sns.countplot(x=data['Gender'],hue=data['Attrition'],palette='seismic_r')
plt.subplot(222)
plt.title('Gender Vs MonthlyIncome')
sns.boxplot(x=data['Gender'],y=data['MonthlyIncome'])
plt.show()
```



Key Inferences from the above Gender vs Attrition • Males have a higher rate of attrition • Females are earning a little higher than male

11 2. DEPARTMENT vs ATTRITION

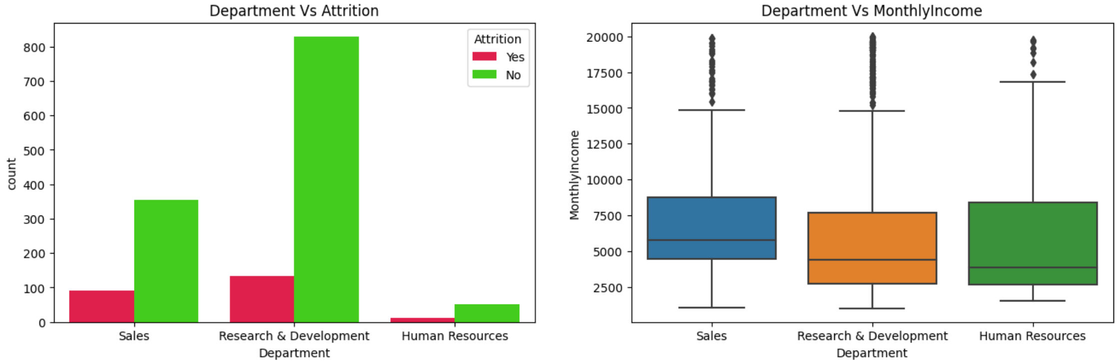
[18]:
```
#comparision with attrition
pd.crosstab(data['Attrition'],data['Department'])
```

[18]:
```
Department  Human Resources  Research & Development  Sales
Attrition
No                       51                     828    354
Yes                      12                     133     92
```

[19]:
```
#comparision with MonthlyIncome
pd.
 ↪pivot_table(data=data,index=['Department'],values=['MonthlyIncome'],aggfunc='mean')
```

[19]:
```
                        MonthlyIncome
Department
Human Resources           6654.507937
Research & Development    6281.252862
Sales                     6959.172646
```

[20]:
```
plt.figure(figsize=(16,10))
plt.subplot(221)
plt.title('Department Vs Attrition')
sns.countplot(x=data['Department'],hue=data['Attrition'],palette='prism_r')
plt.subplot(222)
plt.title('Department Vs MonthlyIncome')
sns.boxplot(x=data['Department'],y=data['MonthlyIncome'])
plt.show()
```



Key Inferences from the above Department vs Attrition • Sales Department has a higher rate of attrition • Sales employees are earning a little higher than other

12 3. JOB ROLE vs ATTRTION

[21]:
```
#comparision with attrition
pd.crosstab(data['Attrition'],data['JobRole'])
```

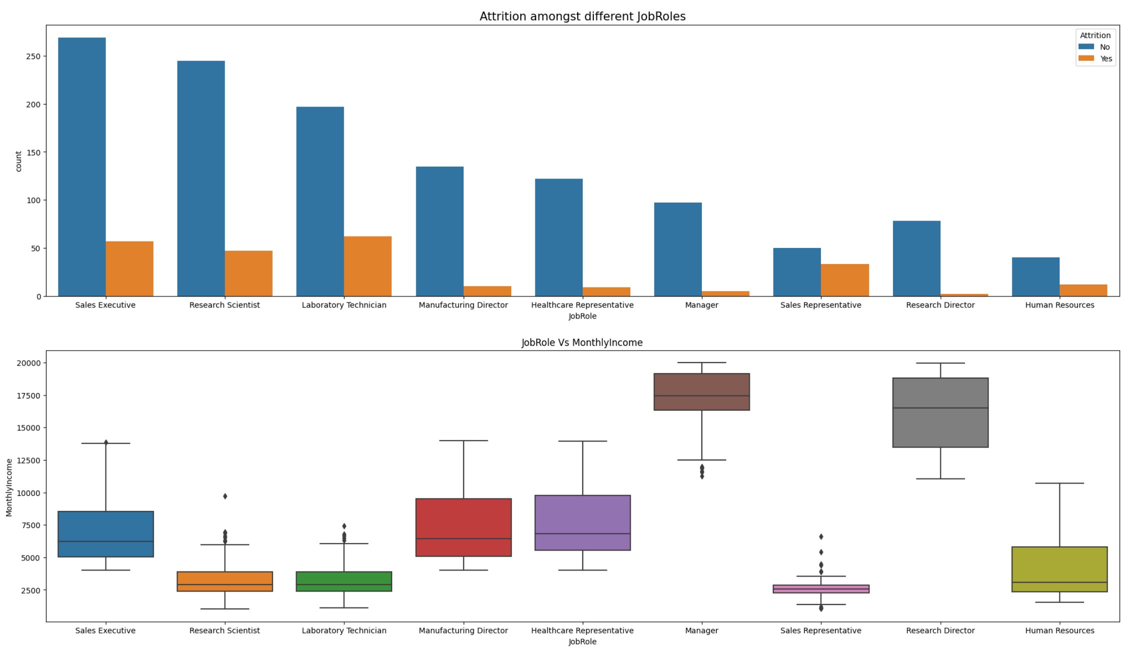
[21]:
```
JobRole    Healthcare Representative  Human Resources  Laboratory Technician  \
Attrition
No                               122               40                    197
Yes                                9               12                     62

JobRole    Manager  Manufacturing Director  Research Director  \
Attrition
No              97                     135                 78
Yes              5                      10                  2

JobRole    Research Scientist  Sales Executive  Sales Representative
Attrition
No                        245              269                    50
Yes                        47               57                    33
```

[22]:
```
#comparision with attrition
pd.crosstab(data['Attrition'],data['JobRole'])
```

[22]:
```
JobRole    Healthcare Representative  Human Resources  Laboratory Technician  \
Attrition
No                               122               40                    197
Yes                                9               12                     62

JobRole    Manager  Manufacturing Director  Research Director  \
Attrition
```

```
No                97                     135                 78
Yes                5                      10                  2

JobRole    Research Scientist  Sales Executive  Sales Representative
Attrition
No                        245              269                    50
Yes                        47               57                    33
```

[23]:
```
plt.figure(figsize=(25,14))
plt.subplot(211)
plt.title('JobRole Vs Attrition')
sns.countplot(x=data['JobRole'],hue=data['Attrition'].
 ↪sort_values(ascending=True))
plt.title('Attrition amongst different JobRoles',size=15)
plt.subplot(212)
plt.title('JobRole Vs MonthlyIncome')
sns.boxplot(x=data['JobRole'],y=data['MonthlyIncome'])
plt.show()
```



Key Inferences from JobRole vs Attrition • Sales Representative and Lab Technicians have a high attrition rate.

13 4. Years In Current Role vs Attrition

[24]:
```
#comparision with attrition
pd.crosstab(data['Attrition'],data['YearsInCurrentRole'])
```

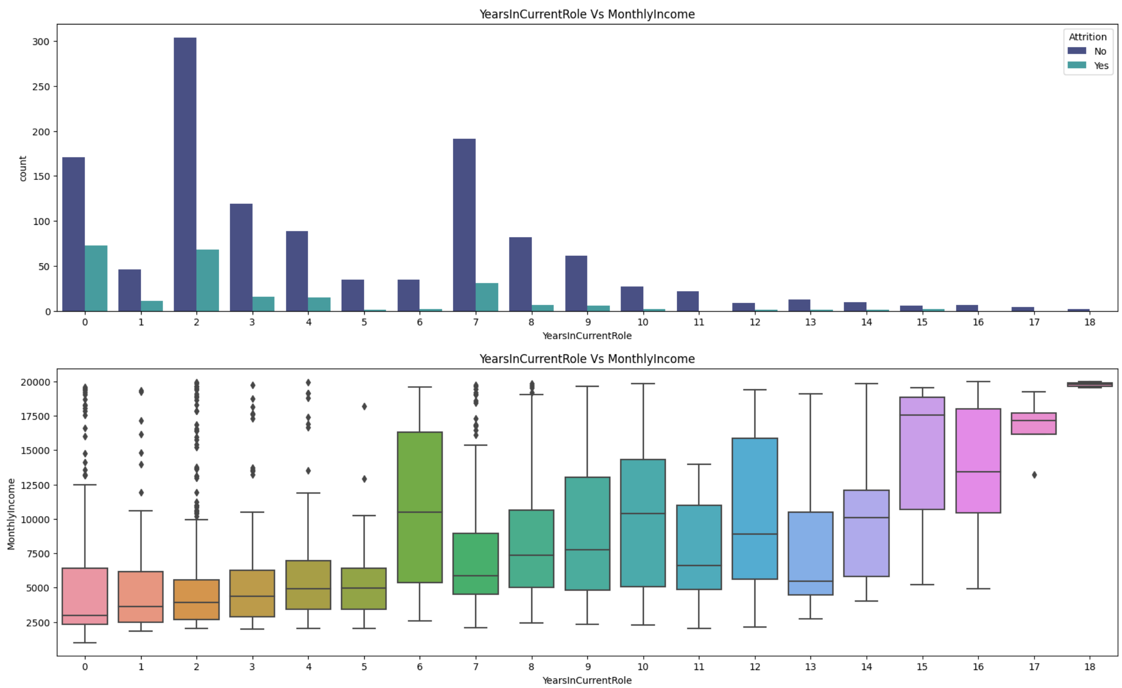
[24]:
```
YearsInCurrentRole   0   1    2    3   4   5   6    7   8   9   10  11  12  \
Attrition
No                 171  46  304  119  89  35  35  191  82  61  27  22   9
Yes                 73  11   68   16  15   1   2   31   7   6   2   0   1

YearsInCurrentRole  13  14  15  16  17  18
Attrition
No                  13  10   6   7   4   2
Yes                  1   1   2   0   0   0
```

[25]:
```
#comparision with MonthlyIncome
pd.
 ↪pivot_table(data=data,index=['YearsInCurrentRole'],values=['MonthlyIncome'],aggfunc='mean')
```

[25]:
```
                    MonthlyIncome
YearsInCurrentRole
0                     5082.487705
1                     5416.298246
2                     5179.615591
3                     5522.644444
4                     6153.701923
5                     5502.333333
6                    10585.945946
7                     7237.351351
8                     8563.808989
9                     9177.776119
10                   10348.448276
11                    7892.727273
12                   10338.400000
13                    8301.142857
14                   10066.818182
15                   14874.125000
16                   13591.285714
17                   16700.000000
18                   19768.000000
```

[26]:
```
plt.figure(figsize=(20,12))
plt.subplot(211)
plt.title('YearsInCurrentRole Vs MonthlyIncome')
sns.countplot(x=data['YearsInCurrentRole'],hue=data['Attrition'].
 ↪sort_values(ascending=True),palette='mako')
plt.subplot(212)
```

```
plt.title('YearsInCurrentRole Vs MonthlyIncome')
sns.boxplot(x=data['YearsInCurrentRole'],y=data['MonthlyIncome'])
plt.show()
```



Key Inferences from YearsInCurrentRole vs Attrition: • Employees with 7,8 and 9 YearsInCurrentRole contribute to 21.5% of the total attrition rate in the organisation • Employee with 6 years in Current Role is earning more than an employee carrying 14 years in Current Role • Need to come up with better stock options for people with more than 6+ years in Current Role as attrition seems to increase gradually with a drop in monthly income.

14 5. TotalWorkingYears vs Attrition

[27]:
```
#comparision with attrition
pd.crosstab(data['Attrition'],data['TotalWorkingYears'])
```

[27]:

TotalWorkingYears	0	1	2	3	4	5	6	7	8	9	…	30	31	32	\
Attrition											…				
No	6	41	22	33	51	72	103	63	87	86	…	7	8	9	
Yes	5	40	9	9	12	16	22	18	16	10	…	0	1	0	

TotalWorkingYears	33	34	35	36	37	38	40
Attrition							
No	6	4	3	6	4	1	0
Yes	1	1	0	0	0	0	2

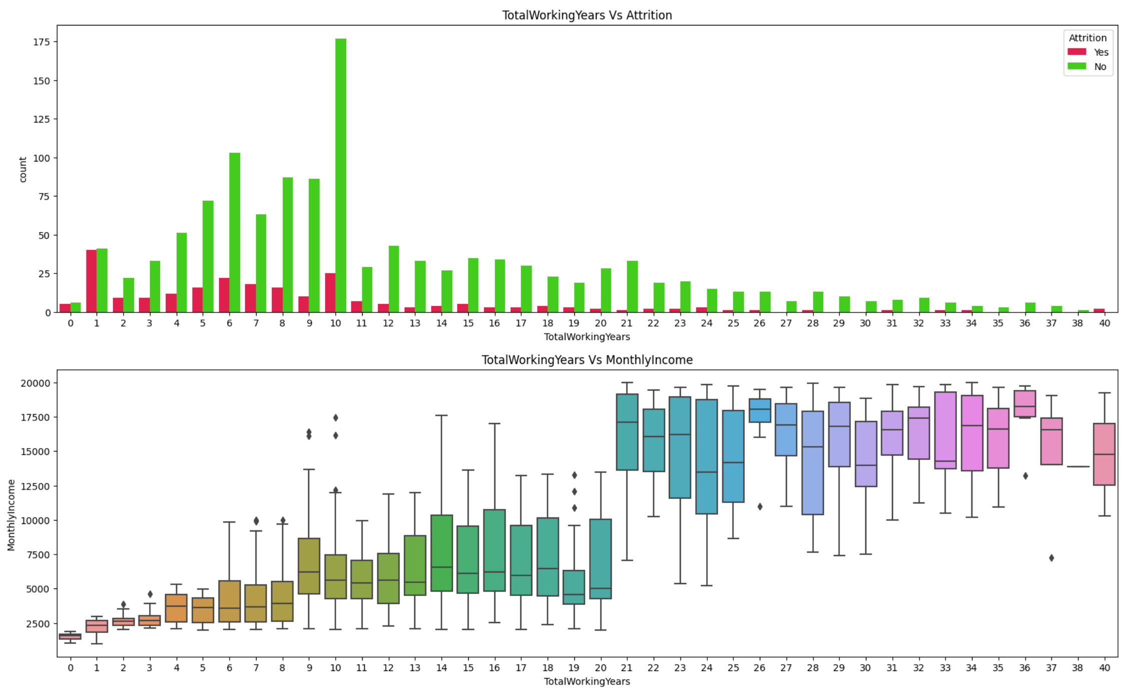
```
[2 rows x 40 columns]
```

[28]:
```
#comparision with MonthlyIncome
pd.
↪pivot_table(data=data,index=['TotalWorkingYears'],values=['MonthlyIncome'],aggfunc='mean').
↪sort_values(by='TotalWorkingYears')
```

[28]:
```
                   MonthlyIncome
TotalWorkingYears
0                    1523.636364
1                    2208.827160
2                    2650.193548
3                    2781.047619
4                    3614.428571
5                    3476.659091
6                    4215.256000
7                    4171.308642
8                    4209.252427
9                    6623.406250
10                   6019.767327
11                   5669.333333
12                   6020.583333
13                   6254.916667
14                   7362.258065
15                   7227.700000
16                   8189.810811
17                   6563.121212
18                   6844.000000
19                   5597.363636
20                   6431.400000
21                  16264.882353
22                  15696.190476
23                  15020.818182
24                  14117.722222
25                  14586.071429
26                  17554.071429
27                  16259.714286
28                  14253.857143
29                  15613.500000
30                  14208.857143
31                  16064.111111
32                  16362.333333
33                  15812.000000
34                  15927.800000
35                  15722.666667
36                  17740.333333
```

```
37                    14857.750000
38                    13872.000000
40                    14779.000000
```

[29]:
```
plt.figure(figsize=(20,12))
plt.subplot(211)
plt.title('TotalWorkingYears Vs Attrition')
sns.
 ↪countplot(x=data['TotalWorkingYears'],hue=data['Attrition'],palette='prism_r')
plt.subplot(212)
plt.title('TotalWorkingYears Vs MonthlyIncome')
sns.boxplot(x=data['TotalWorkingYears'],y=data['MonthlyIncome'])
plt.show()
```



Key Inferences from TotalWorkingYears vs Attrition • An innovative structure needs to be implemented for employees with 1 year of experience as it is majorly contributing to the attrition % • Seems like the organisation has benefits in terms of income for people with 20+ years of experience • Why people with 6 years of experience earning the same as employees with 19 years of experience ? • Why employees with 21 years of work experience earning as much as an employee with 40 years of experience?

15 Statistical Analysis

We can perform an attrition analysis using statistics to gain insights, Statistical analysis in attrition refers to the use of various statistical techniques and methods to analyze and understand the factors, trends, and patterns associated with employee turnover or attrition within a human resources context. The goal of statistical analysis in attrition is to extract meaningful insights from data in order to make informed decisions about employee retention strategies, workforce planning, and organizational improvements.

- H0=Feature = Attrition
- H1=Feature!= Attrition to formulate a hypothesis with Gender ,
- h0=mu male = mu female
- h1=mu male!= mu female to formulate a hypothesis with Department ,
- h0=mu Sales = mu Research & Development= mu Human Resources
- h1=mu Sales = mu Research & Development= mu Human Resources

[30]:
```
from scipy.stats import chi2_contingency,chisquare,f_oneway
```

16 1. Statistical analysis for categorical data types, Chisquare is performed

Chi-Square Test: The Chi-Square Test is used to determine whether there is a significant association between two categorical variables. It assesses whether the observed distribution of data in a contingency table (cross-tabulation of two categorical variables) differs significantly from what would be expected under a null hypothesis of no association.

1. Chi-Square Test for Independence: Tests whether two categorical variables are independent of each other.
2. Chi-Square Test of Goodness of Fit: Tests whether an observed frequency distribution fits an expected theoretical distribution. ex: Testing whether there's a significant association between job satisfaction levels and attrition rates.

[31]:
```
cat_cols = list(data.describe(include = "O").columns)
print(cat_cols)
```

```
['Attrition', 'BusinessTravel', 'Department', 'EducationField', 'Gender',
'JobRole', 'MaritalStatus', 'OverTime']
```

[32]:
```
chi_stat = []
p_value = []

for i in cat_cols:
    chi_res = chi2_contingency(np.array(pd.crosstab(data[i],
 ↪data['Attrition'])))
    chi_stat.append(chi_res[0])
    p_value.append(chi_res[1])

chi_square = pd.DataFrame([chi_stat, p_value])
chi_square = chi_square.T
col = ['Chi Square Value', 'P-Value']
chi_square.columns = col
chi_square.index = cat_cols
```

[33]:
```
chi_square
```

[33]:

	Chi Square Value	P-Value
Attrition	1462.614554	0.000000e+00
BusinessTravel	24.182414	5.608614e-06
Department	10.796007	4.525607e-03
EducationField	16.024674	6.773980e-03
Gender	1.116967	2.905724e-01
JobRole	86.190254	2.752482e-15
MaritalStatus	46.163677	9.455511e-11
OverTime	87.564294	8.158424e-21

[34]:
```
#Obtaining categorical feature with P-value<0.05, means these features are␣
↪dependent and have correlation with target variable
chi_square[chi_square["P-Value"]<0.05]
```

[34]:

	Chi Square Value	P-Value
Attrition	1462.614554	0.000000e+00
BusinessTravel	24.182414	5.608614e-06
Department	10.796007	4.525607e-03
EducationField	16.024674	6.773980e-03
JobRole	86.190254	2.752482e-15
MaritalStatus	46.163677	9.455511e-11
OverTime	87.564294	8.158424e-21

[35]:
```
features_p = list(chi_square[chi_square["P-Value"]<0.05].index)
print("Significant categorical Features:\n",features_p)
```

```
Significant categorical Features:
 ['Attrition', 'BusinessTravel', 'Department', 'EducationField', 'JobRole',
'MaritalStatus', 'OverTime']
```

17 2. statistical analysis for numerical data types , ANOVA Test is performed

Analysis of Variance (ANOVA): ANOVA is used to compare the means of two or more groups to determine whether there are statistically significant differences among them. It's particularly useful when you have more than two groups to compare. ANOVA assesses whether the observed variance between group means is greater than what would be expected due to random chance.

1.One-Way ANOVA: Used when there's one independent variable (factor) and multiple levels or groups. 2. Two-Way ANOVA: Used when there are two independent variables, examining their individual and interactive effects. ex: Analyzing whether there are significant differences in average salaries across different job roles in a company.

[36]:
```
num_cols = list(data.describe(include='number').columns)
```

```python
# Print numerical columns one by one on separate lines
for col in num_cols:
    print(col)
```

```
Age
DailyRate
DistanceFromHome
Education
EmployeeNumber
EnvironmentSatisfaction
HourlyRate
JobInvolvement
JobLevel
JobSatisfaction
MonthlyIncome
MonthlyRate
NumCompaniesWorked
PercentSalaryHike
PerformanceRating
RelationshipSatisfaction
StockOptionLevel
TotalWorkingYears
TrainingTimesLastYear
WorkLifeBalance
YearsAtCompany
YearsInCurrentRole
YearsSinceLastPromotion
YearsWithCurrManager
```

[37]:

```python
f_stat = []
p_val = []

for i in num_cols:
    atr_0 = data[data['Attrition'] == "No"][i]
    atr_1 = data[data['Attrition'] == "Yes"][i]
    a = f_oneway(atr_0, atr_1)
    f_stat.append(a[0])
    p_val.append(a[1])

anova = pd.DataFrame([f_stat, p_val])
anova = anova.T
cols = ['F-STAT', 'P-VALUE']
anova.columns = cols
anova.index = num_cols
```

[38]:

```python
anova
```

[38]:

	F-STAT	P-VALUE
Age	38.175887	8.356308e-10
DailyRate	4.726640	2.985816e-02
DistanceFromHome	8.968277	2.793060e-03
Education	1.446308	2.293152e-01
EmployeeNumber	0.164255	6.853276e-01
EnvironmentSatisfaction	15.855209	7.172339e-05
HourlyRate	0.068796	7.931348e-01
JobInvolvement	25.241985	5.677065e-07
JobLevel	43.215344	6.795385e-11
JobSatisfaction	15.890004	7.043067e-05
MonthlyIncome	38.488819	7.147364e-10
MonthlyRate	0.337916	5.611236e-01
NumCompaniesWorked	2.782287	9.552526e-02
PercentSalaryHike	0.266728	6.056128e-01
PerformanceRating	0.012250	9.118840e-01
RelationshipSatisfaction	3.095576	7.871363e-02
StockOptionLevel	28.140501	1.301015e-07
TotalWorkingYears	44.252491	4.061878e-11
TrainingTimesLastYear	5.211646	2.257850e-02
WorkLifeBalance	6.026116	1.421105e-02
YearsAtCompany	27.001624	2.318872e-07
YearsInCurrentRole	38.838303	6.003186e-10
YearsSinceLastPromotion	1.602218	2.057900e-01
YearsWithCurrManager	36.712311	1.736987e-09

[39]:

```
features_p_n = list(anova[anova["P-VALUE"]<0.05].index)
print("Significant numerical Features:\n",features_p_n)
```

```
Significant numerical Features:
 ['Age', 'DailyRate', 'DistanceFromHome', 'EnvironmentSatisfaction',
'JobInvolvement', 'JobLevel', 'JobSatisfaction', 'MonthlyIncome',
'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear',
'WorkLifeBalance', 'YearsAtCompany', 'YearsInCurrentRole',
'YearsWithCurrManager']
```

Key Inference of Statistical Analysis There are 20 Features having a correlation with the Target Variable. These are:

‘Attrition’, ‘BusinessTravel’, ‘Department’, ‘EducationField’, ‘JobRole’, ‘MaritalStatus’, ‘OverTime’, ‘Age’, ‘DailyRate’, ‘DistanceFromHome’, ‘EnvironmentSatisfaction’, ‘JobInvolvement’, ‘JobLevel’, ‘JobSatisfaction’, ‘MonthlyIncome’, ‘StockOptionLevel’, ‘TotalWorkingYears’, ‘TrainingTimesLastYear’, ‘WorkLifeBalance’, ‘YearsAtCompany’, ‘YearsInCurrentRole’, ’YearsWithCurrManage