# Project Development

# Delivery Of Sprint-1

| | |
|---|---|
| Date | 03 October 2022 |
| Team ID | PNT2022TMID37447 |
| Project Name | Project - Corporate Employee Attrition Analytics |

## CODING & SOLUTIONING

## DATASET:

- Employee Attrition Analysis (Logistic Regression Model)

Employee Attrition Analysis (Logistic Regression Model)

https://www.kaggle.com/vjchoudhary7/hr-analytics-case-study

## DATA UNDERSTANDING:

The data received for the analysis can be divided into 4 broad categories -

•General Data – General data, acquired from HR

•Employee Survey Data – Data collected from yearly employee survey

•Manager Survey Data – Data collected from yearly manager survey

•Biometric Data – Daily in and out times for each employee, collected using biometric attendance machines

| General Data |
|---|
| Age |
| **Attrition (Yes/No)** |
| Department |
| Education Field |

| Manager Survey Data |
|---|
| Job Involvement |
| Performance Rating |

| Employee Survey Data |
|---|
| Environment Satisfaction |
| Job Satisfaction |
| Work Life Balance |

| Biometric Data |
|---|
| In Time |
| Out Time |

## UNDERSTANDING THE DATASET:

Let us try to understand each field of the data (general_data.csv)

Below are the values each column has. The column names are pretty self-explanatory.

1. AGE Numerical Value
2. ATTRITION Employee leaving the company (0=no, 1=yes)
3. BUSINESS TRAVEL (1=No Travel, 2=Travel Frequently, 3=Travel Rarely)
4. DEPARTMENT (1=HR, 2=R&D, 3=Sales)
5. DISTANCE FROM HOME Numerical Value - THE DISTANCE FROM WORK TO HOME
6. EDUCATION Numerical Value. (1 'Below College' 2 'College' 3 'Bachelor' 4 'Master' 5 'Doctor')
7. EDUCATION FIELD (1=HR, 2=LIFE SCIENCES, 3=MARKETING, 4=MEDICAL SCIENCES, 5=OTHERS, 6= TECHNICAL)
8. EMPLOYEE COUNT Numerical Value
9. EMPLOYEE ID Numerical Value
10. GENDER (1=FEMALE, 2=MALE)
11. JOB LEVEL Numerical Value
12. JOB ROLE (1=HR REP, 2=HR, 3=LAB TECHNICIAN, 4=MANAGER, 5= MANAGING DIRECTOR, 6= RESEARCH DIRECTOR, 7= RESEARCH SCIENTIST, 8=SALES EXECUTIVE, 9= SALES REPRESENTATIVE)
13. MARITAL STATUS (1=DIVORCED, 2=MARRIED, 3=SINGLE)
14. MONTHLY INCOME Numerical Value - MONTHLY SALARY
15. NUMCOMPANIES WORKED Numerical Value - NO. OF COMPANIES WORKED AT
16. OVER 18 (1=YES, 2=NO)
17. PERCENT SALARY HIKE Numerical Value - PERCENTAGE INCREASE IN SALARY
18. STANDARD HOURS Numerical Value - STANDARD HOURS
19. STOCK OPTIONS LEVEL Numerical Value - STOCK OPTIONS (Higher the number, the more stock option an employee has)
20. TOTAL WORKING YEARS Numerical Value - TOTAL YEARS WORKED
21. TRAINING TIMES LAST YEAR Numerical Value - HOURS SPENT TRAINING

22. YEARS AT COMPANY Numerical Value - TOTAL NUMBER OF YEARS AT THE COMPANY
23. YEARS SINCE LAST PROMOTION Numerical Value - LAST PROMOTION
24. YEARS WITH CURRENT MANAGER Numerical Value - YEARS SPENT WITH CURRENT MANAGER

b. Let us try to understand about each field of the data (employee_survey_data.csv)

1. Employee ID
2. Environment Satisfaction (1 'Low' 2 'Medium' 3 'High' 4 'Very High')
3. Job Satisfaction (1 'Low' 2 'Medium' 3 'High' 4 'Very High')
4. Work Life Balance (1 'Bad', 2 'Good', 3 'Better', 4 'Best')

c. Let us try to understand about each field of the data (manager_survey_data.csv)

1. Employee ID
2. Job Involvement (1 'Low' 2 'Medium' 3 'High' 4 'Very High')
3. Performance Rating (1 'Low', 2 'Good', 3 'Excellent', 4 'Outstanding')

## SOLUTION REQUIRED:

- To model the probability of attrition using a logistic regression
- Business Understanding
- Data Understanding – sources of the data, meaning of the data
- Data preparation & EDA
- Model Building
- Model Evaluation
- Data Visualization charts
- Dashboard Creation

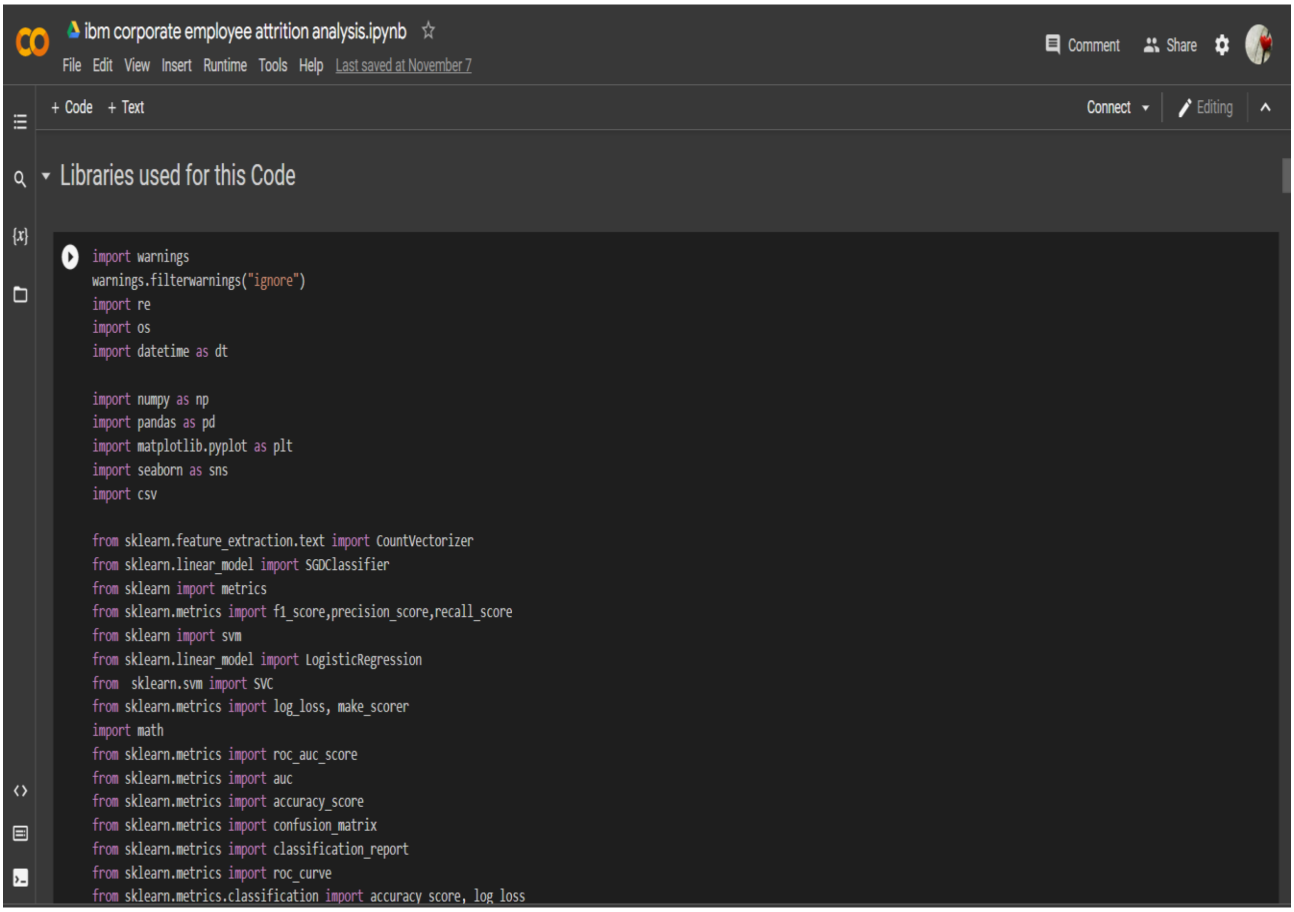
## METHODOLOGY USED:

- Predictive modelling of attrition
- Recommending ways for company XYZ to decrease its level of attrition

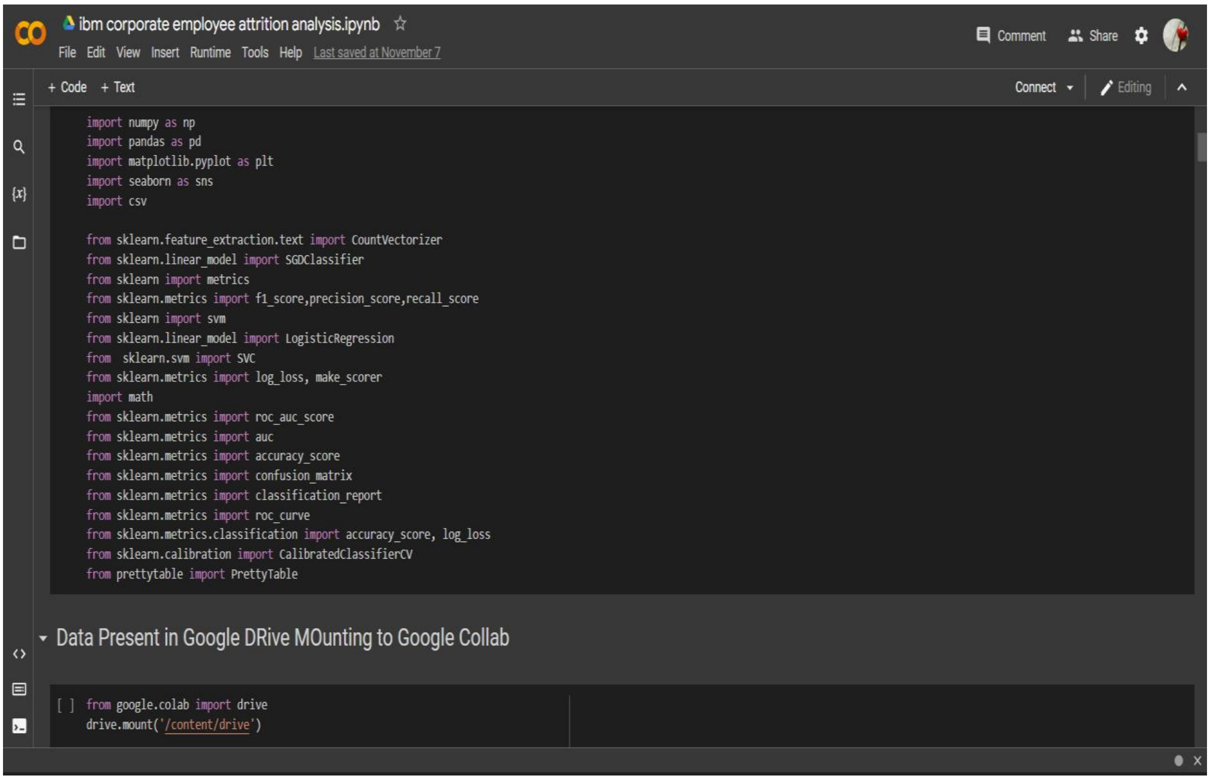
# TO MODEL THE PROBABILITY OF ATTRITION USING A LOGISTIC REGRESSION

## BUSINESS UNDERSTANDING, IMPORTING PACKAGES, UNDERSTANDING THE DATA AND EDA

## LIBRARIES USED FOR THIS CODE:




```
import warnings
warnings.filterwarnings("ignore")
import re
import os
import datetime as dt

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import csv

from sklearn.feature_extraction.text import CountVectorizer
from sklearn.linear_model import SGDClassifier
from sklearn import metrics
from sklearn.metrics import f1_score,precision_score,recall_score
from sklearn import svm
from sklearn.linear_model import LogisticRegression
from  sklearn.svm import SVC
from sklearn.metrics import log_loss, make_scorer
import math
from sklearn.metrics import roc_auc_score
from sklearn.metrics import auc
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
from sklearn.metrics import roc_curve
from sklearn.metrics.classification import accuracy_score, log_loss
```

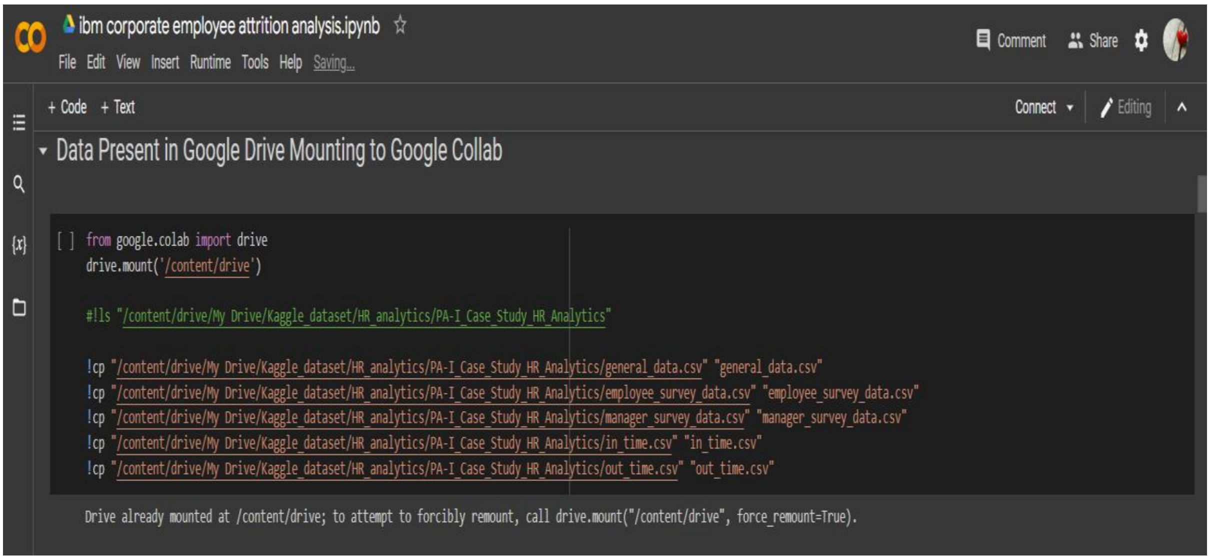



## CODING:

```
import warnings
warnings.filterwarnings("ignore")
import re
import os
import datetime as dt

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import csv
```

```
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.linear_model import SGDClassifier
from sklearn import metrics
from sklearn.metrics import f1_score,precision_score,recall_score
from sklearn import svm
from sklearn.linear_model import LogisticRegression
from  sklearn.svm import SVC
from sklearn.metrics import log_loss, make_scorer
import math
from sklearn.metrics import roc_auc_score
from sklearn.metrics import auc
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
from sklearn.metrics import roc_curve
from sklearn.metrics.classification import accuracy_score, log_loss
from sklearn.calibration import CalibratedClassifierCV
from prettytable import PrettyTable
```

## TO READ ALL THE CSV FILES:




## CODING:

```
from google.colab import drive

drive.mount('/content/drive')


#!ls "/content/drive/My Drive/Kaggle_dataset/HR_analytics/PA-I_Case_Study_HR_Analytics"


!cp "/content/drive/My Drive/Kaggle_dataset/HR_analytics/PA-I_Case_Study_HR_Analytics/general_data.csv" "general_data.csv"

!cp "/content/drive/My Drive/Kaggle_dataset/HR_analytics/PA-I_Case_Study_HR_Analytics/employee_survey_data.csv" "employee_survey_data.csv"

!cp "/content/drive/My Drive/Kaggle_dataset/HR_analytics/PA-I_Case_Study_HR_Analytics/manager_survey_data.csv" "manager_survey_data.csv"

!cp "/content/drive/My Drive/Kaggle_dataset/HR_analytics/PA-I_Case_Study_HR_Analytics/in_time.csv" "in_time.csv"

!cp "/content/drive/My Drive/Kaggle_dataset/HR_analytics/PA-I_Case_Study_HR_Analytics/out_time.csv" "out_time.csv"
```

## OUTPUT:

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

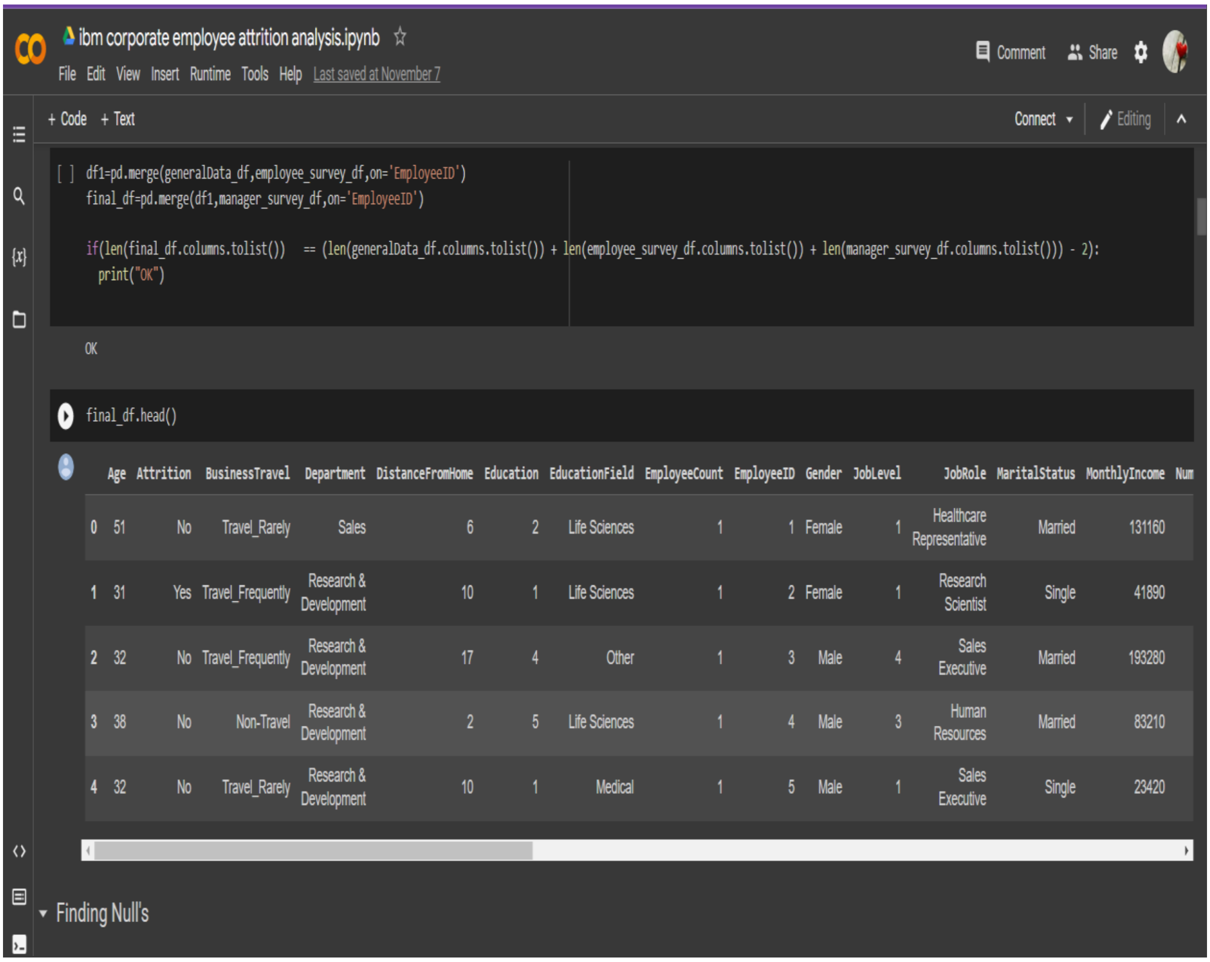
## CODING:


```
generalData_df     = pd.read_csv("general_data.csv")

employee_survey_df = pd.read_csv("employee_survey_data.csv")

manager_survey_df  = pd.read_csv("manager_survey_data.csv")

intime_df          = pd.read_csv("in_time.csv")

outtime_df         = pd.read_csv("out_time.csv")


print(generalData_df.shape ,  employee_survey_df.shape ,  manager_survey_df.shape)

print(employee_survey_df.columns.tolist())

print(manager_survey_df.columns.tolist())
```

## OUTPUT:

```
(4410, 24) (4410, 4) (4410, 3)
['EmployeeID', 'EnvironmentSatisfaction', 'JobSatisfaction',
'WorkLifeBalance']
['EmployeeID', 'JobInvolvement', 'PerformanceRating']
```

## MERGING OF DATA:



ibm corporate employee attrition analysis.ipynb

File Edit View Insert Runtime Tools Help Last saved at November 7

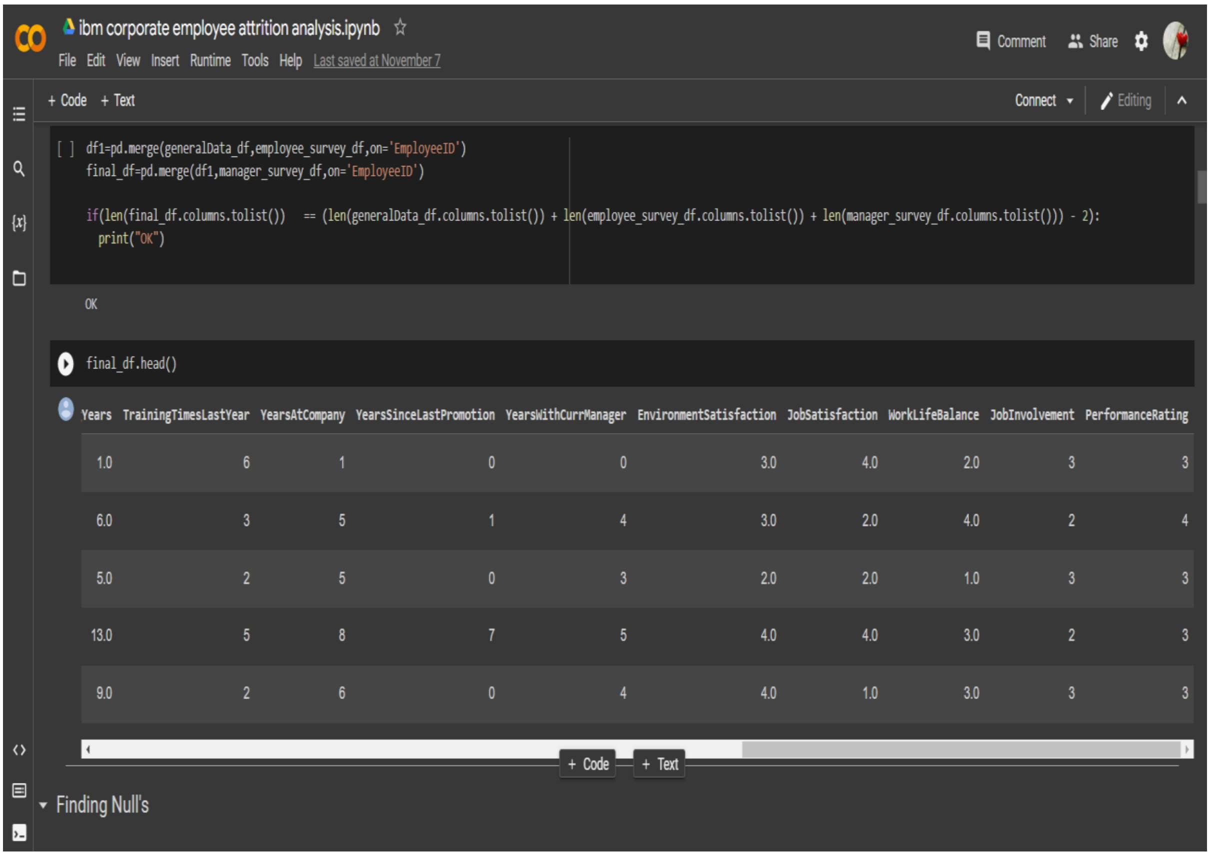
```
df1=pd.merge(generalData_df,employee_survey_df,on='EmployeeID')
final_df=pd.merge(df1,manager_survey_df,on='EmployeeID')

if(len(final_df.columns.tolist())  == (len(generalData_df.columns.tolist()) + len(employee_survey_df.columns.tolist()) + len(manager_survey_df.columns.tolist())) - 2):
  print("OK")
```

```
OK
```

```
final_df.head()
```

| | Age | Attrition | BusinessTravel | Department | DistanceFromHome | Education | EducationField | EmployeeCount | EmployeeID | Gender | JobLevel | JobRole | MaritalStatus | MonthlyIncome | Num |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 51 | No | Travel_Rarely | Sales | 6 | 2 | Life Sciences | 1 | 1 | Female | 1 | Healthcare Representative | Married | 131160 | |
| 1 | 31 | Yes | Travel_Frequently | Research & Development | 10 | 1 | Life Sciences | 1 | 2 | Female | 1 | Research Scientist | Single | 41890 | |
| 2 | 32 | No | Travel_Frequently | Research & Development | 17 | 4 | Other | 1 | 3 | Male | 4 | Sales Executive | Married | 193280 | |
| 3 | 38 | No | Non-Travel | Research & Development | 2 | 5 | Life Sciences | 1 | 4 | Male | 3 | Human Resources | Married | 83210 | |
| 4 | 32 | No | Travel_Rarely | Research & Development | 10 | 1 | Medical | 1 | 5 | Male | 1 | Sales Executive | Single | 23420 | |

Finding Null's




## CODING:

```
df1=pd.merge(generalData_df,employee_survey_df,on='EmployeeID')
final_df=pd.merge(df1,manager_survey_df,on='EmployeeID')


if(len(final_df.columns.tolist())   == (len(generalData_df.columns.tolist()) + len(employee_survey_df.columns.tolist()) + len(manager_survey_df.columns.tolist())) - 2):
  print("OK")
```

## OUTPUT:

OK

Age Attrition BusinessTravel Department DistanceFromHome Education
EducationField EmployeeCount EmployeeID Gender JobLevel JobRole
MaritalStatus MonthlyIncome NumCompaniesWorked Over18 PercentSalaryHike
StandardHours StockOptionLevel TotalWorkingYears TrainingTimesLastYear
YearsAtCompany YearsSinceLastPromotion YearsWithCurrManager
EnvironmentSatisfaction JobSatisfaction WorkLifeBalance JobInvolvement
PerformanceRating

0 51 No Travel_Rarely Sales 6 2 Life Sciences 1 1
Female 1 Healthcare Representative Married 131160 1.0 Y 11
8 0 1.0 6 1 0 0 3.0 4.0 2.0 3 3

1 31 Yes Travel_Frequently Research & Development 10 1 Life
Sciences 1 2 Female 1 Research Scientist Single 41890 0.0 Y
23 8 1 6.0 3 5 1 4 3.0 2.0 4.0 2
4

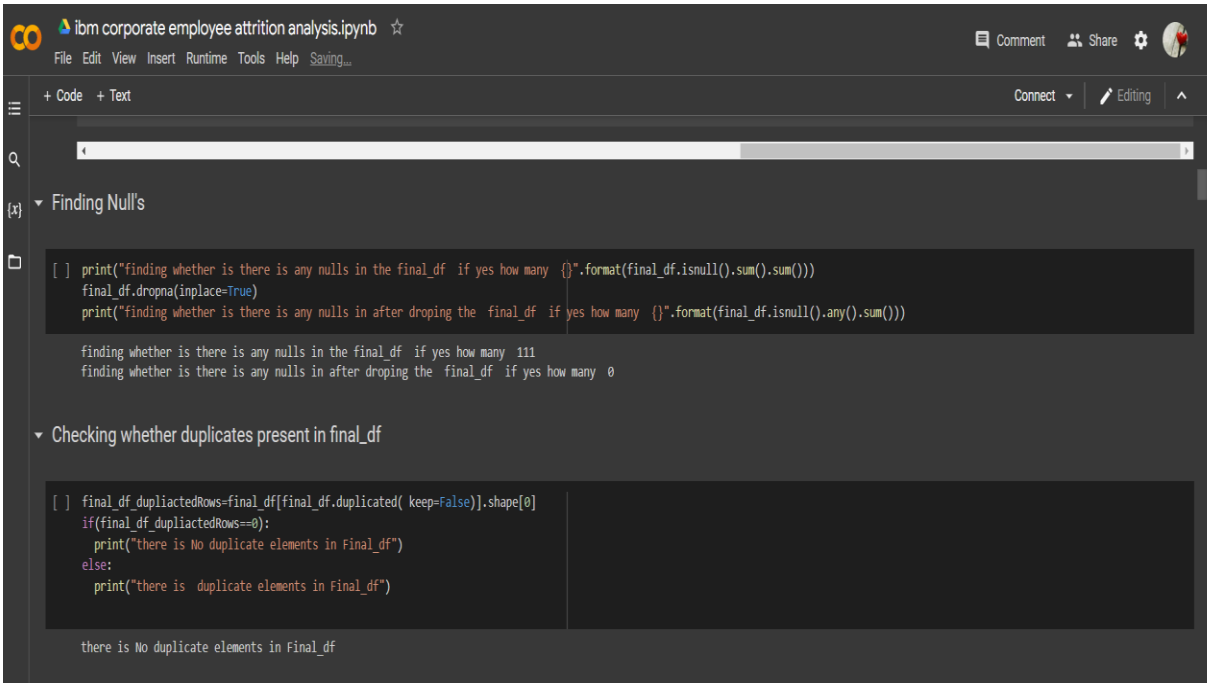
2 32 No Travel_Frequently Research & Development 17 4
Other 1 3 Male 4 Sales Executive Married 193280 1.0 Y
15 8 3 5.0 2 5 0 3 2.0 2.0 1.0 3
3

3 38 No Non-Travel Research & Development 2 5 Life
Sciences 1 4 Male 3 Human Resources Married 83210 3.0
Y 11 8 3 13.0 5 8 7 5 4.0 4.0 3.0
2 3

4 32 No Travel_Rarely Research & Development 10 1 Medical1
5 Male 1 Sales Executive Single 23420 4.0 Y 12 8 2
9.0 2 6 0 4 4.0 1.0 3.0 3

## FINDING NULL'S:

## CODING:




```
print("finding whether is there is any nulls in the final_df  if yes how many  {}".format(final_df.isnull().sum().sum()))
final_df.dropna(inplace=True)
print("finding whether is there is any nulls in after droping the  final_df  if yes how many  {}".format(final_df.isnull().any().sum()))
```

## OUTPUT:

finding whether is there is any nulls in the final_df  if yes how many  111
finding whether is there is any nulls in after droping the  final_df  if yes how many  0

finding whether is there is any nulls in the final_df  if yes how many  111

finding whether is there is any nulls in after droping the  final_df  if yes how many  0

## CHECKING WHETHER DUPLICATES PRESENT IN FINAL_DF

## CODING:


```
final_df_dupliactedRows=final_df[final_df.duplicated( keep=False)].shape[0]
if(final_df_dupliactedRows==0):
  print("there is No duplicate elements in Final_df")
else:
  print("there is  duplicate elements in Final_df")
```

## OUTPUT:

there is No duplicate elements in Final_df

EDA

We will cover how to visually analyse:

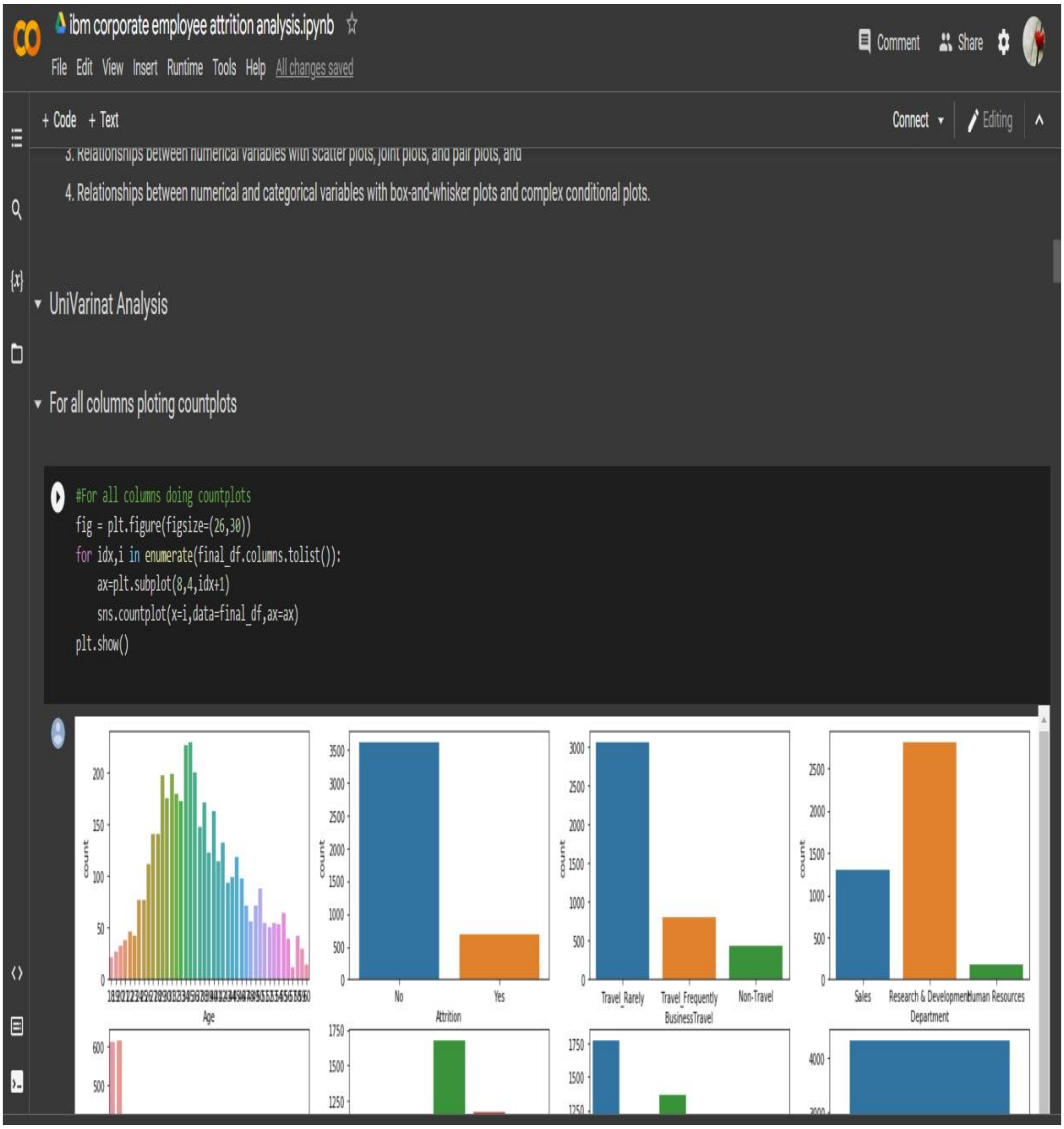
- Numerical variables with histograms,
- Categorical variables with count plots,
- Relationships between numerical variables with scatter plots, joint plots, and pair plots, and
- Relationships between numerical and categorical variables with box-and-whisker plots and complex conditional plots

## UNIVARIANT ANALYSIS

## FOR ALL COLUMNS PLOTTING COUNT PLOTS

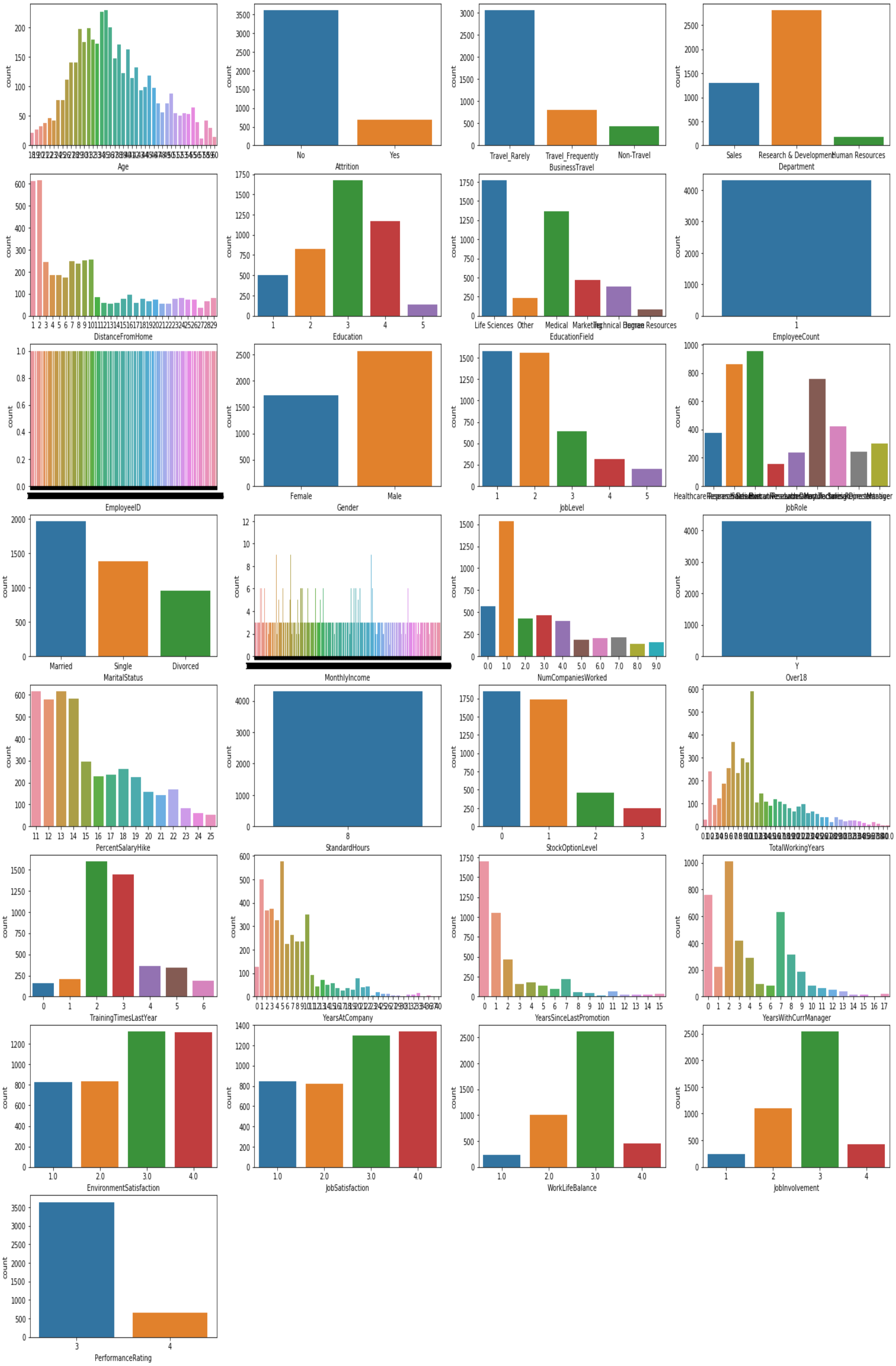
## CODING:

```
#For all columns doing countplots
fig = plt.figure(figsize=(26,30))
for idx,i in enumerate(final_df.columns.tolist()):
    ax=plt.subplot(8,4,idx+1)
    sns.countplot(x=i,data=final_df,ax=ax)
plt.show()
```

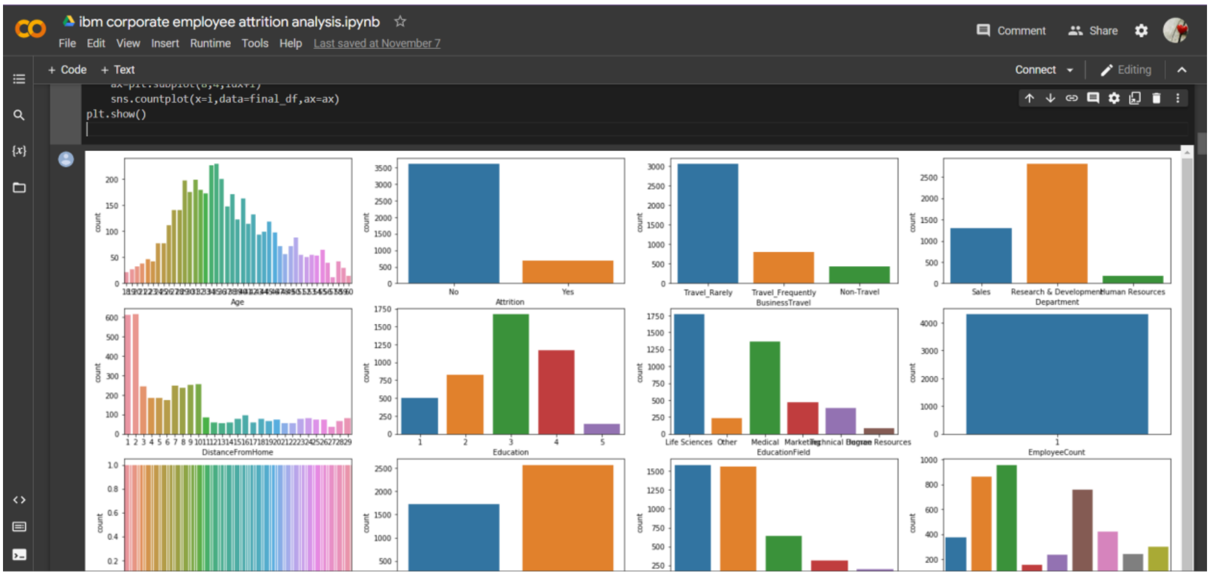


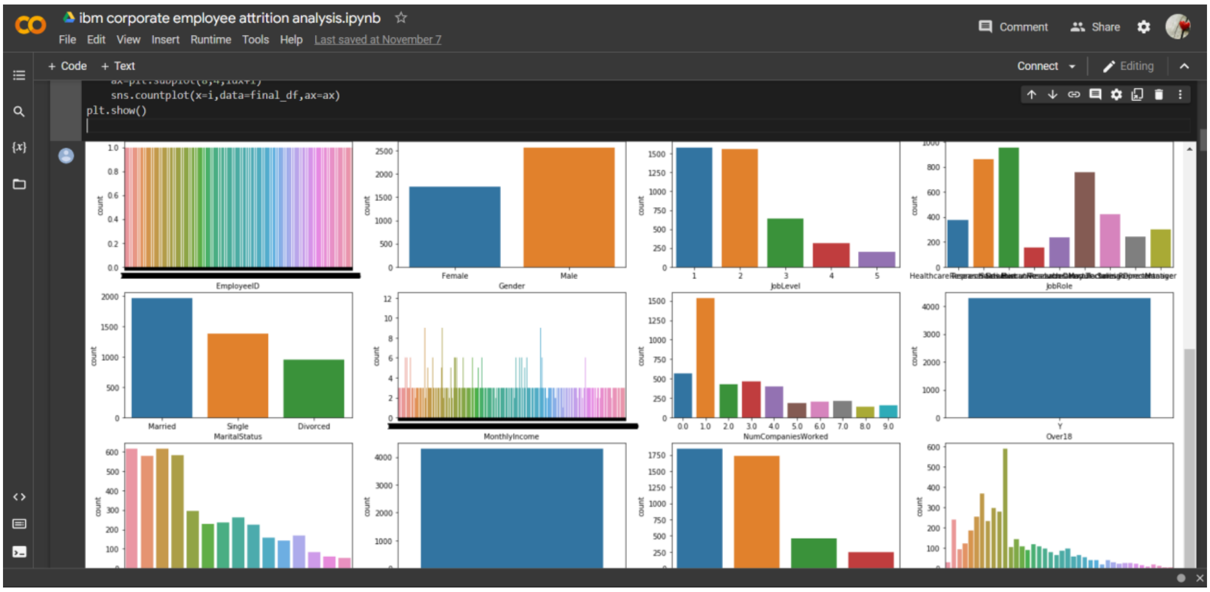

FOR NUMERICAL COLUMNS UNIVARIANT ANALYSIS IS DONE

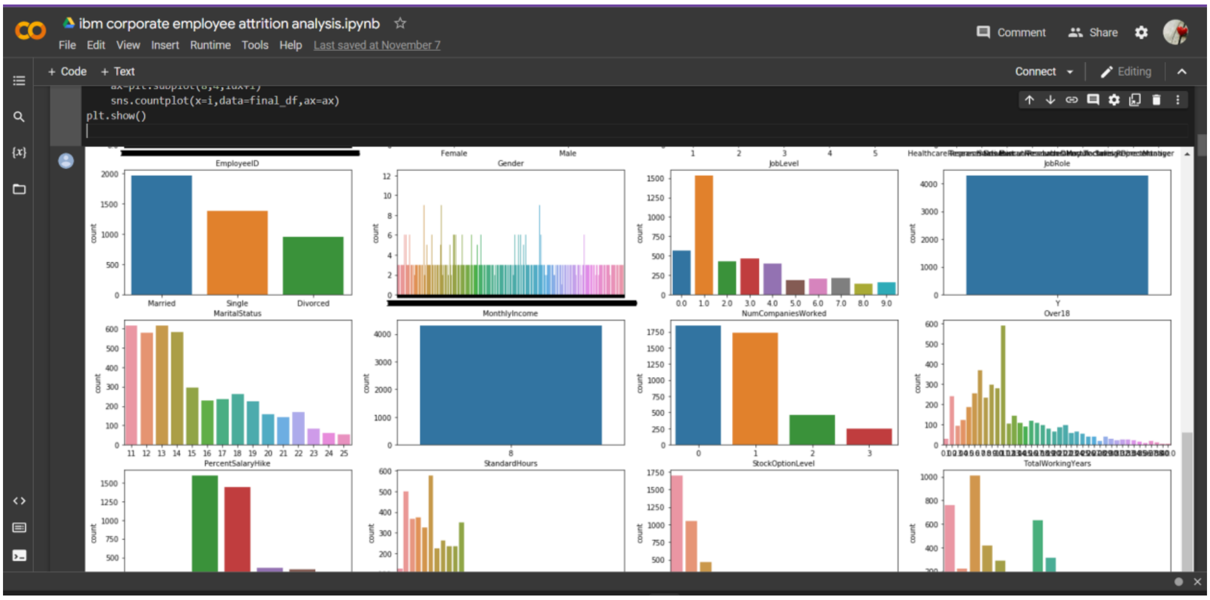
CODE



count
Age
No
Yes
Attrition
Travel_Rarely
Travel_Frequently
Non-Travel
BusinessTravel
Sales
Research & Development
Human Resources
Department
DistanceFromHome
Education
Life Sciences
Other
Medical
EducationField
EmployeeCount
EmployeeID
Female
Male
Gender
JobLevel
Healthcare
JobRole
Married
Single
Divorced
MaritalStatus
MonthlyIncome
NumCompaniesWorked
Y
Over18
PercentSalaryHike
StandardHours
StockOptionLevel
TotalWorkingYears
TrainingTimesLastYear
YearsAtCompany
YearsSinceLastPromotion
YearsWithCurrManager
EnvironmentSatisfaction
JobSatisfaction
WorkLifeBalance
JobInvolvement
PerformanceRating

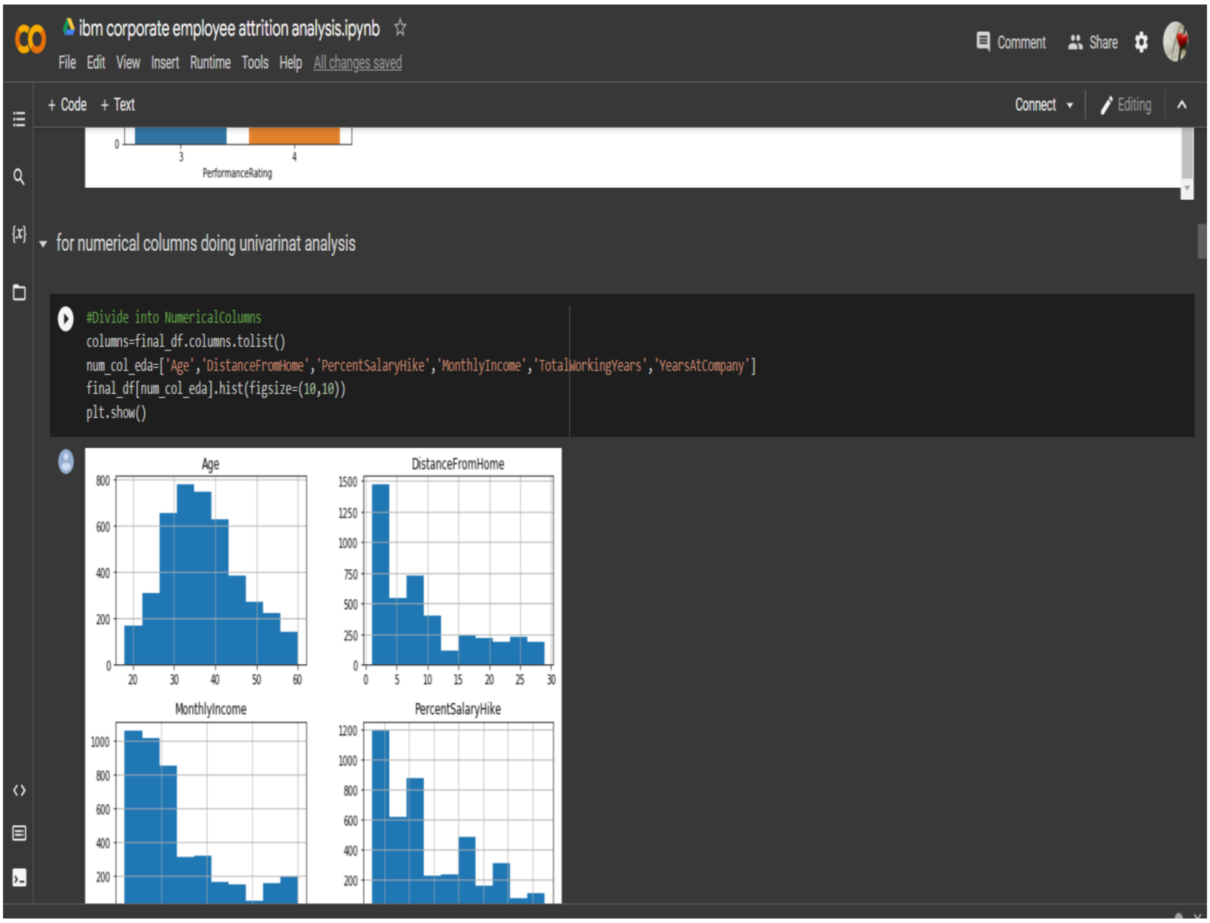

ibm corporate employee attrition analysis.ipynb
File Edit View Insert Runtime Tools Help Last saved at November 7
Comment Share
+ Code + Text
Connect Editing
sns.countplot(x=i,data=final_df,ax=ax)
plt.show()
Age
Attrition
BusinessTravel
Department
DistanceFromHome
Education
EducationField
EmployeeCount


ibm corporate employee attrition analysis.ipynb
File Edit View Insert Runtime Tools Help Last saved at November 7
Comment Share
+ Code + Text
Connect Editing
sns.countplot(x=i,data=final_df,ax=ax)
plt.show()
EmployeeID
Gender
JobLevel
JobRole
MaritalStatus
MonthlyIncome
NumCompaniesWorked
Over18


ibm corporate employee attrition analysis.ipynb
File Edit View Insert Runtime Tools Help Last saved at November 7
Comment Share
+ Code + Text
Connect Editing
sns.countplot(x=i,data=final_df,ax=ax)
plt.show()
MaritalStatus
MonthlyIncome
NumCompaniesWorked
Over18
PercentSalaryHike
StandardHours
StockOptionLevel
TotalWorkingYears

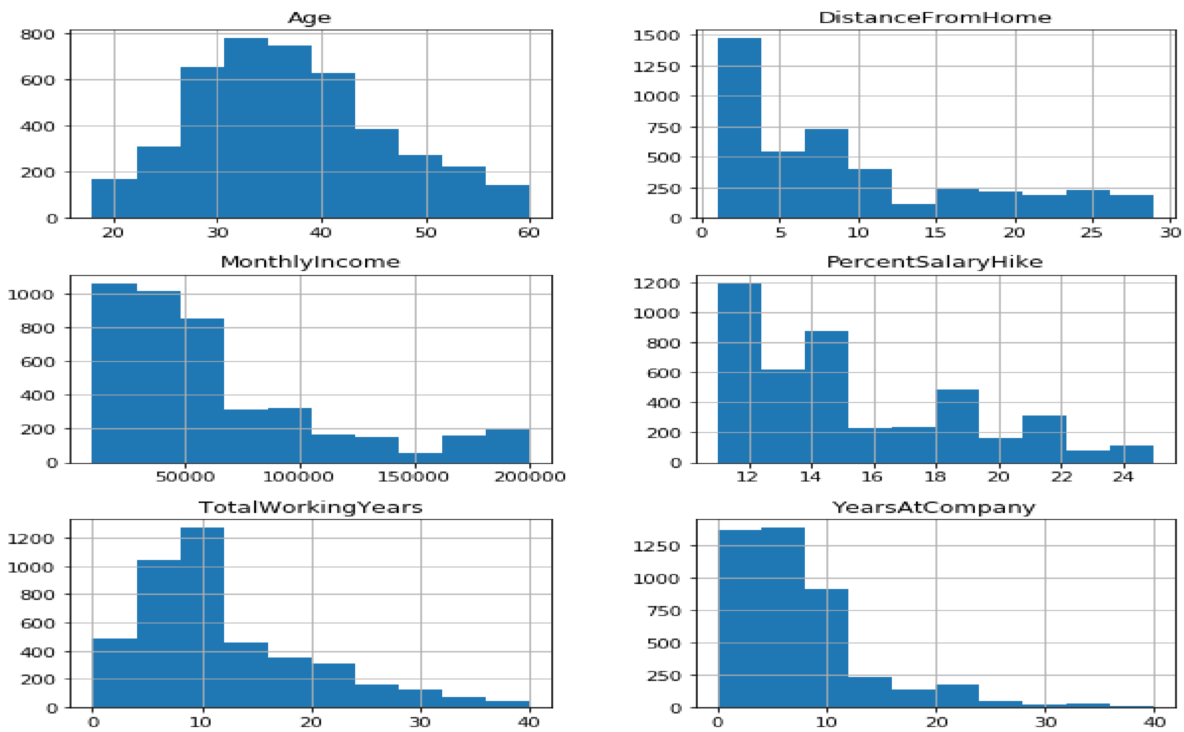
## FOR NUMERICAL COLUMNS DOING UNIVARINAT ANALYSIS

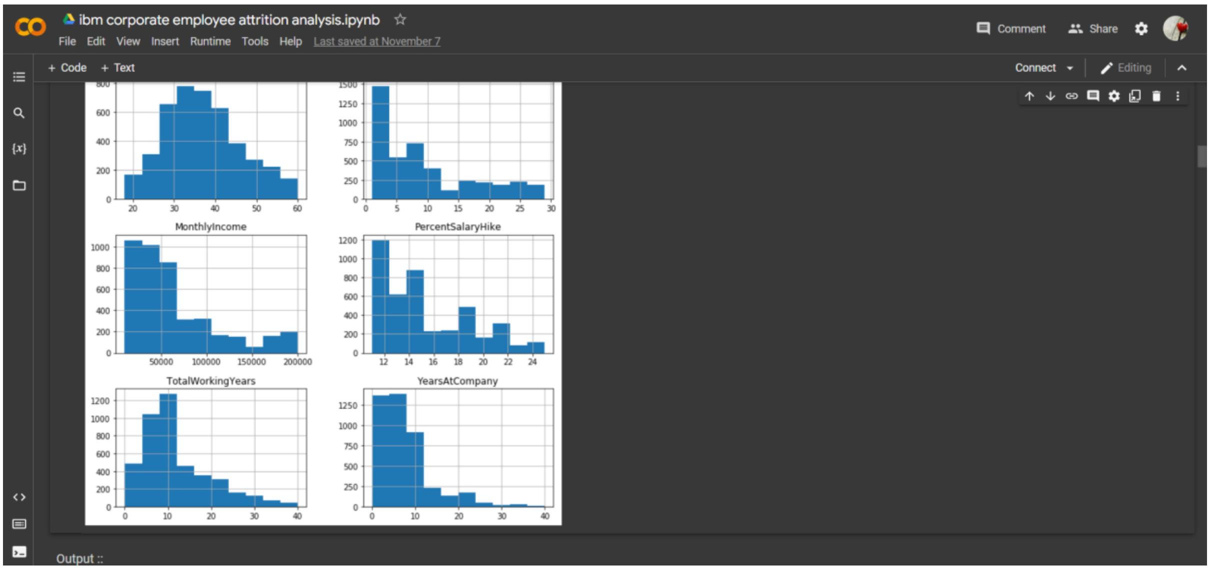
## CODING:




```
#Divide into NumericalColumns
columns=final_df.columns.tolist()
num_col_eda=['Age','DistanceFromHome','PercentSalaryHike','MonthlyIncome',
'TotalWorkingYears','YearsAtCompany']
final_df[num_col_eda].hist(figsize=(10,10))
plt.show()
```

## OUTPUT:




# INFERENCES

Key Observation from Above Plot are

- Except Age most of the Columns are in Skew Distribution form
- Age Feature Distribution is almost Normal Distribution
- As logistic regression does not require independent variables to be normal distributed .so i am not changing distribution of features which are skewed into the normal Distribution

## INSIGHTS

- **Attrition: Whether the employee left in the previous year or not**

1. Employee who left in the previous year are 14% of population (1375) i.e. 192 who believe Environment Satisfaction is High in org. in org.

2. Employee who left in the previous year are 15% of population (856) i.e. 129 who believe Environment Satisfaction is Medium in org.

3. Employee who left in the previous year are 13% of population (1334) i.e. 173 who believe Environment Satisfaction is Very High in org.

4. Employee who left in the previous year are 25% of population (845) i.e. 211 who believe Environment Satisfaction is Low in organization.

- People who left in the previous year & believe Environment Satisfaction is Low in org were 30% of population who left in the previous year. Second by People who left in the previous year & believe Environment Satisfaction is High in organization