ATTRITION ASSIGNMENT DAY 12 STATS TESTS

STEP 1 = LAUNCHING :

import pandas as pd

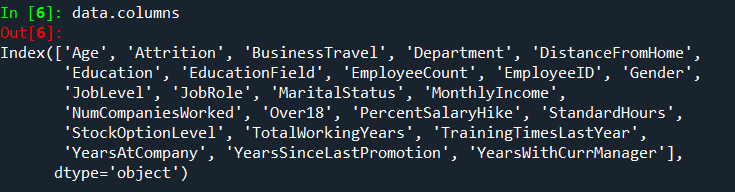
import numpy as n

import matpolib.pyplot as pl

data=pd.read\_csv(“general\_data.csv”)

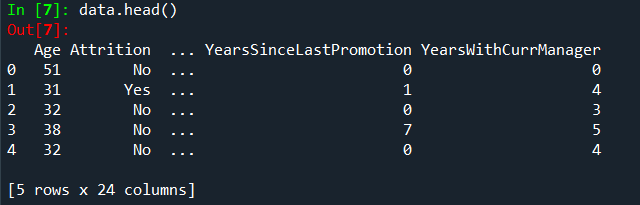
->To find column names .

data.columns



->To find the data of first 5 rows.

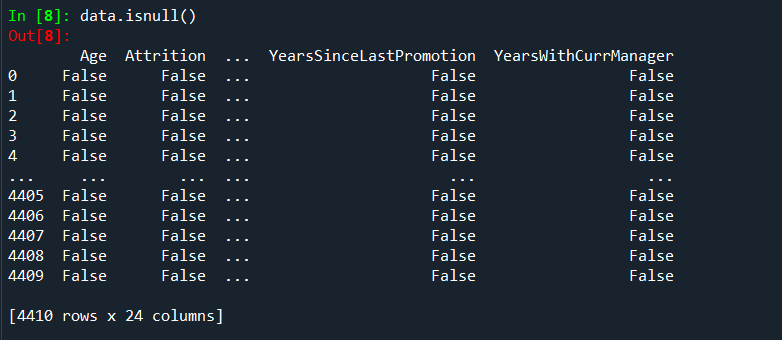
data.head( )



STEP 2 = DATA TREATMENT :

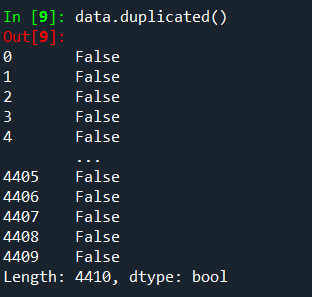
->To find out null values in the table.

data.isnull( )



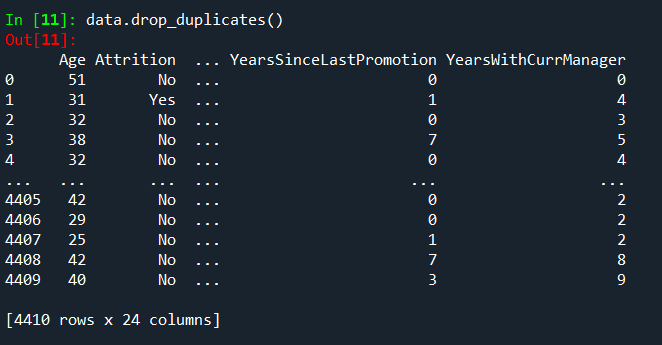
->To find out duplicated values of table.

data.duplicated( )



->To drop all duplicated values of the table.

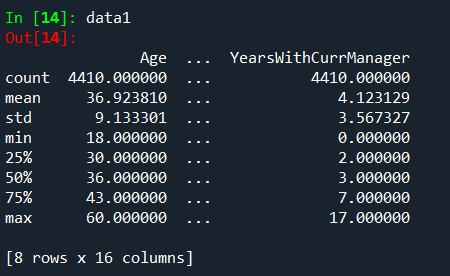
data.drop\_duplicates( )



STEP 3 = UNIVARIATE ANALYSIS :

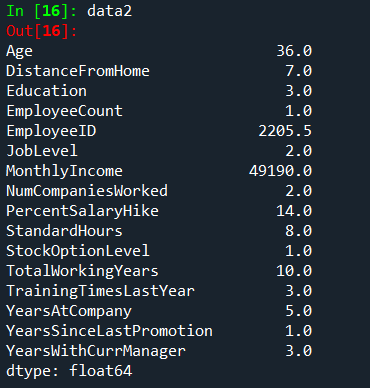
->To describe the whole table.

data1=data[['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome','Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gender','JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome', 'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours','StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear','YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager']].describe( )



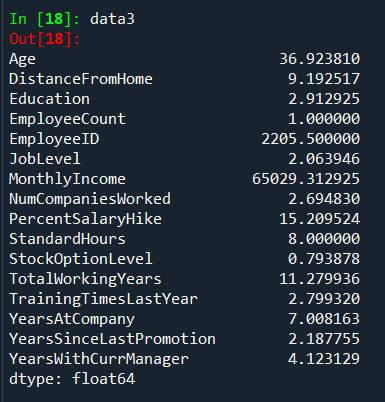
->To find out median of each column .

data2=data[['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome','Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gender','JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome', 'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours','StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear','YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager']].median( )



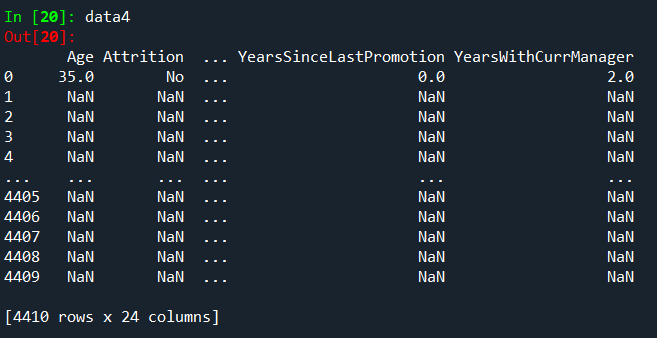
->To find out mean of each column.

data3=data[['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome','Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gender','JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome', 'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours','StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear','YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager']].mean( )



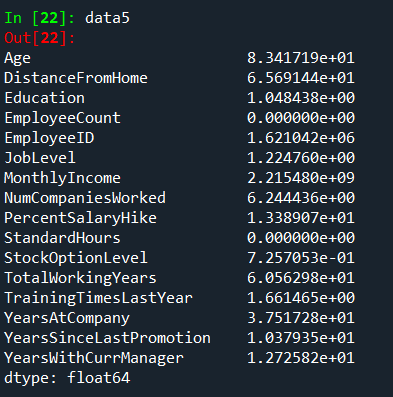
->To find out mode.

data4=data[['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome','Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gender','JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome', 'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours','StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear','YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager']].mode( )



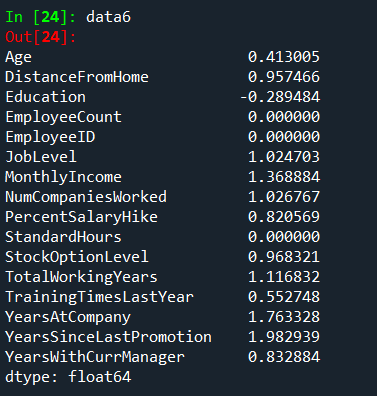
->To find variance of each column.

data5=data[['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome','Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gender','JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome', 'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours','StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear','YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager']].var( )



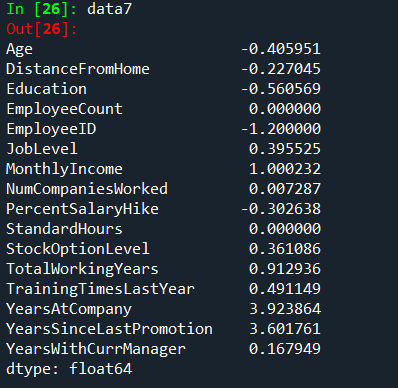
->To find skewness.

data6=data[['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome','Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gender','JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome', 'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours','StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear','YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager']].skew( )



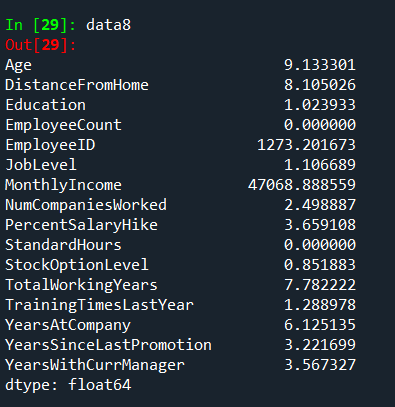
->To find out kurtosis.

data7=data[['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome','Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gender','JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome', 'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours','StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear','YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager']].kurt( )

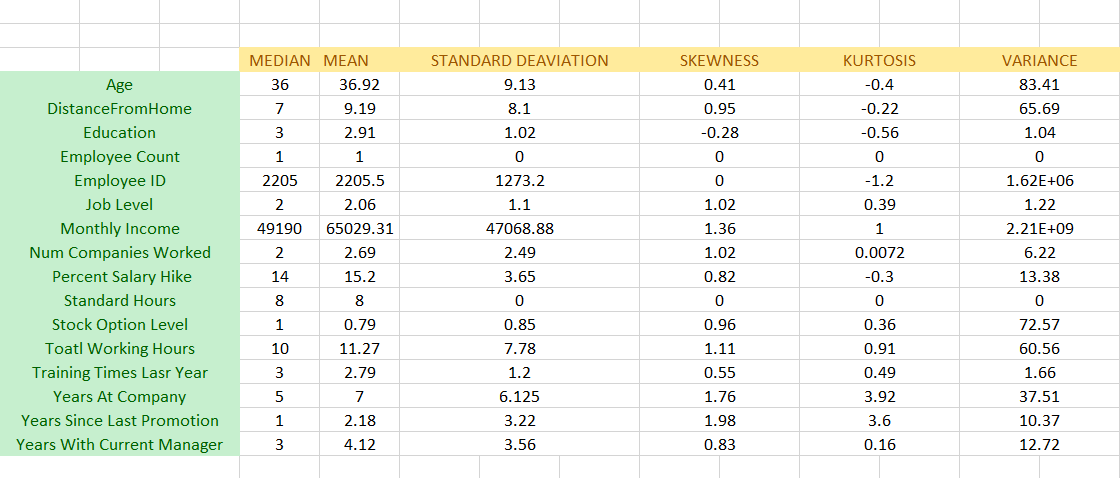


->To find standard deviation .

data8=data[['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome','Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gender','JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome', 'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours','StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear','YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager']].std( )



INFERENCE :



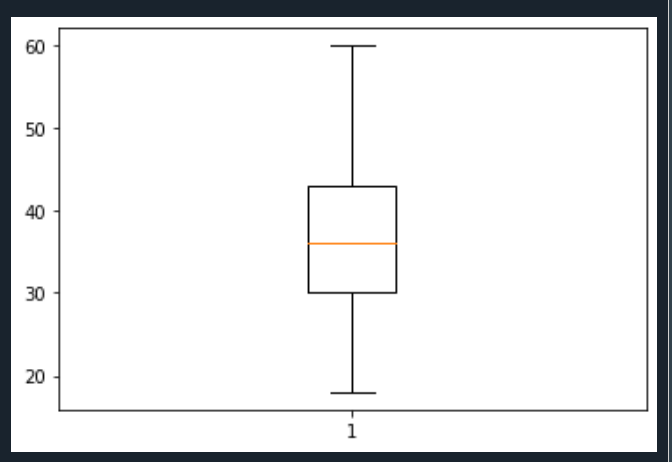
->All the above variables show positive skewness; while Age &Mean\_distance\_from\_home are leptokurtic and all other variables are platykurtic.

OUTLIERS :

There’s no regression found while plotting Age, MonthlyIncome, TotalWorkingYears , YearsAtCompany, etc., on a scatter plot.

box\_plot=data.Age

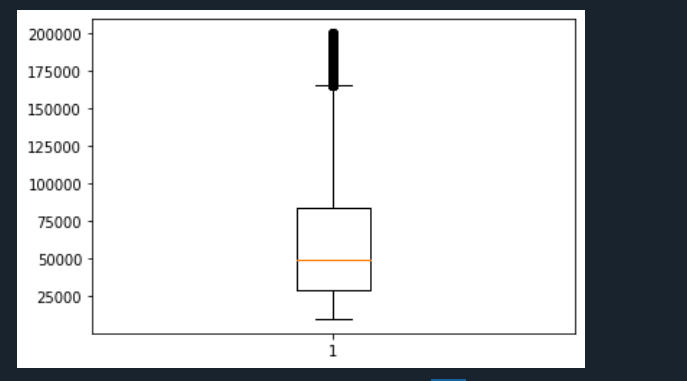
pl.boxplot(box\_plot)



Age is normally distributed without any outliers

box\_plot=data.MonthlyIncome

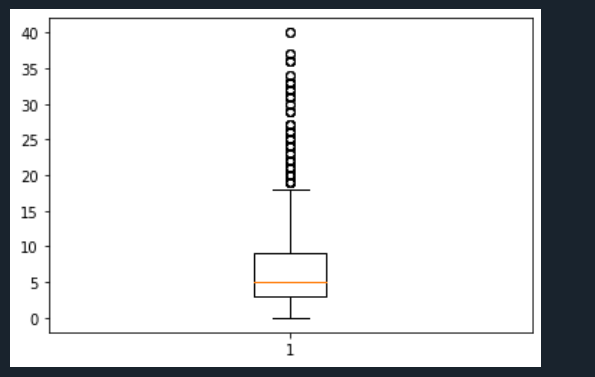
pl.boxplot(box\_plot)



Monthly Income is Right skewed with several outliers

box\_plot=data.YearsAtCompany

pl.boxplot(box\_plot)



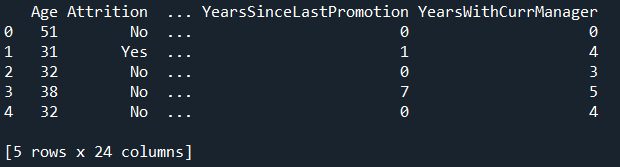
Years at company is also Right Skewed with several outliers observed.

**4>STATISTICAL TEST ( MANN-WHITNEY)**

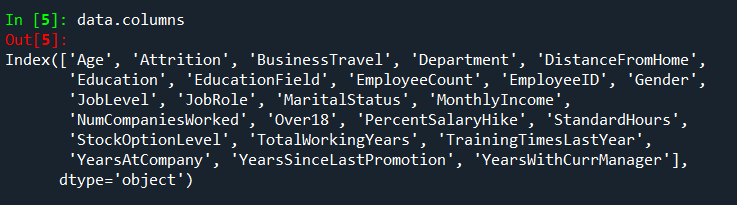
import pandas as pd

data=pd.read\_csv("general\_data.csv")

data.head()



data.columns



**MANN WHTNEY TEST**

import pandas as pd

df=pd.read\_csv('general\_data.csv')

dummy=pd.get\_dummies(df['Attrition'])

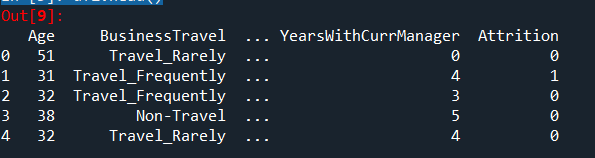
df2=pd.concat((df,dummy),axis=1)

df2=df2.drop(['Attrition'],axis=1)

df2=df2.drop(['No'],axis=1)

df2=df2.rename(columns={"Yes":"Attrition"})

df2.head()



**ATTRITION VS DISTANCE FROM HOME.**

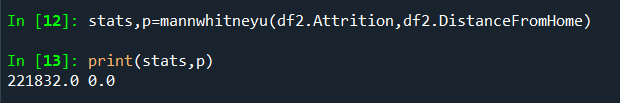
H0 = There is no significant difference between attrition yes and no for distance from home

HA= There is significant difference between attrition yes and no for distance from home

from scipy.stats import mannwhitneyu

stats,p=mannwhitneyu(df2.Attrition,df2.DistanceFromHome)

print(stats,p)



As the P value of 0.0 is < 0.05, the H0 is rejected and HA is accepted.

So there is difference in attrition and distance from home.

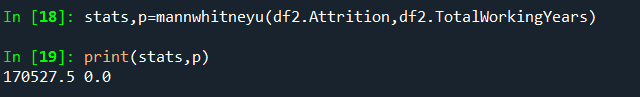
**ATTRITION VS TOTAL WORKING YEARS**

H0 = There is no significant difference between attrition yes and no for total working years.

HA= There is significant difference between attrition yes and no for total working years.

stats,p=mannwhitneyu(df2.Attrition,df2.TotalWorkingYears)

print(stats,p)



As the P value of 0.0 is < 0.05, the H0 is rejected and Ha is accepted.

So there is difference in attrition and total working years.

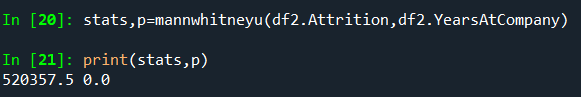
**ATTRITION VS YEARS AT COMPANY**

H0 = There is no significant difference between attrition yes and no for years at company.

HA= There is significant difference between attrition yes and no for years at company.

stats,p=mannwhitneyu(df2.Attrition,df2.YearsAtCompany)

print(stats,p)



As the P value of 0.0 is < 0.05, the H0 is rejected and Ha is accepted.

So there is difference in attrition and years at company.

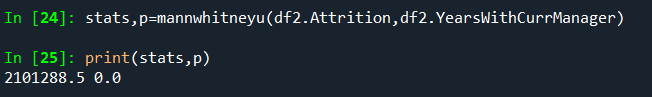
**ATTRITION VS YEARS WITH CURRENT MANAGER**

H0 = There is no significant difference between attrition yes and no for years with current manager.

HA= There is significant difference between attrition yes and no for years with current manager.

stats,p=mannwhitneyu(df2.Attrition,df2.YearsWithCurrManager)

print(stats,p)



As the P value of 0.0 is < 0.05, the H0 is rejected and HA is accepted.

So there is difference in attrition and years with current manager.