**ATTRITION PROJECT**

**Import packages:**

import pandas as pd

import matplotlib.pyplot as plt

**Import data:**

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

**Cheacking whether null values are existing or not**:

dataset.isnull()

da=dataset.dropna()

da.shape

**Describe the data:**

describe=da[['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()



**Skwness of the data:**

da.skew()

Out[22]:

|  |  |  |
| --- | --- | --- |
| * Age | 0.413048 | |
| * DistanceFromHome | **0.955517** | |
| * Education | **-0.288977** | |
| * EmployeeCount | **0.00000** | |
| * EmployeeID | **-0.002335** | |
| * JobLevel | **1.021797** | |
| * MonthlyIncome | **1.367457** | |
| * NumCompaniesWorked | **1.029174** | |
| * PercentSalaryHike | **0.819510** | |
| * StandardHours | | **0.000000** |
| * StockOptionLevel | | **0.967263** |
| * TotalWorkingYears | | **1.115419** |
| * TrainingTimesLastYear | | **0.551818** |
| * YearsAtCompany | | **1.764619** |
| * YearsSinceLastPromotion | | **1.980992** |
| * YearsWithCurrManager | | **0.834277** |
|  | | |

da.kurt()

Out[23]:

|  |  |
| --- | --- |
| * Age | -0.409517 |
| * DistanceFromHome | **-0.230691** |
| * Education | **-0.565008** |
| * EmployeeCount | **0.000000** |
| * EmployeeID | **-1.198607** |
| * JobLevel | **0.388189** |
| * MonthlyIncome | **0.990836** |
| * NumCompaniesWorked | **0.014307** |
| * PercentSalaryHike | **-0.306951** |
| * StandardHours | **0.000000** |
| * StockOptionLevel | **0.356755** |
| * TotalWorkingYears | **0.909316** |
| * TrainingTimesLastYear | **0.494215** |
| * YearsAtCompany | **3.930726** |
| * YearsSinceLastPromotion | **3.592162** |
| * YearsWithCurrManager | **0.170703** |

**Infrence from analysis**:

**skwness analysis result**:

Here Age,DistanceFromHome,JobLevel,MonthlyIncome

NumCompaniesWorked,PercentSalaryHike,

StockOptionLevel,TotalWorkingYears,TrainingTimesLastYear,YearsAtCompany,YearsSinceLastPromotion,YearsWithCurrManager ,those variables are positiveskwness

Employee id and Education are negative skewness.

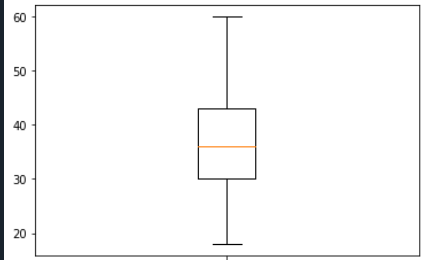
**kurtosis analysis result**:

Here,MonthlyIncome,NumCompaniesWorked,StandardHours,StockOptionLevel,TotalWorkingYears,TrainingTimesLastYear,YearsAtCompany,YearsSinceLastPromotion,YearsWithCurrManager,those variables peckness are high.(Leptokurtic)

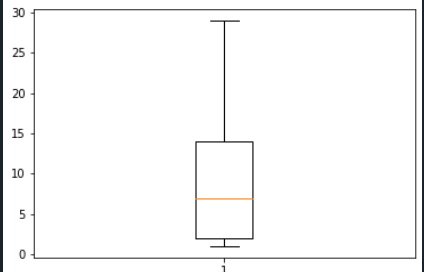
HereAge,DistanceFromHome,Education,Employeeid,PercentSalaryHike,those variables peakness are flat.(platykurtic)

**Box plot for analyis distribution of data:**

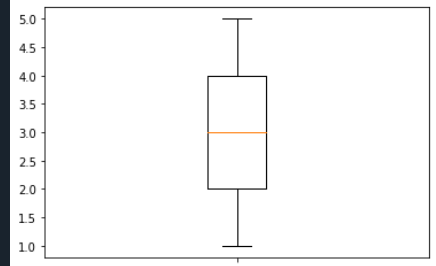
plt.boxplot(da.Age)



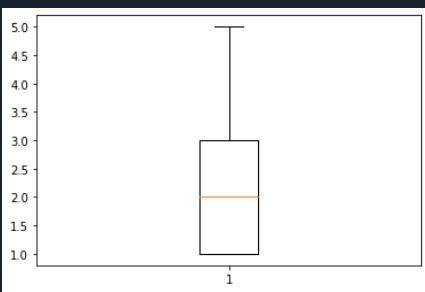
plt.boxplot(da.DistanceFromHome)



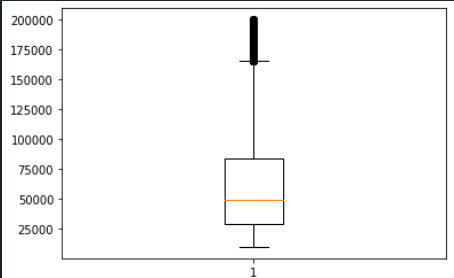
plt.boxplot(da.Education)



plt.boxplot(da.JobLevel)

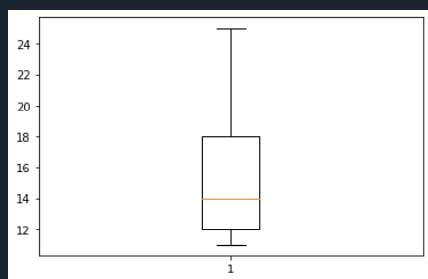


plt.boxplot(da.MonthlyIncome)

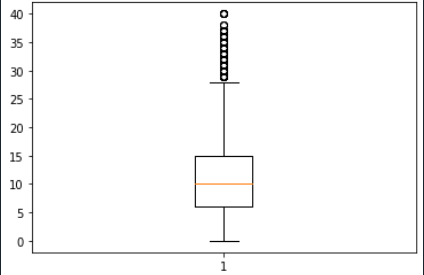


Here monthlyincome has few outlayers.

plt.boxplot(da.PercentSalaryHike)

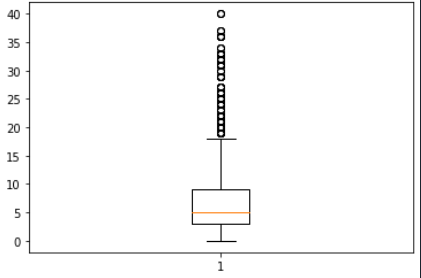


plt.boxplot(da.TotalWorkingYears)



Here totalworking years has few outlayers.

plt.boxplot(da.YearsAtCompany)



Here yearsatcompany has few outlayers.

**Infrence from box plots:**

Here Age,joblevel,Education,total working years are normoly distributed.

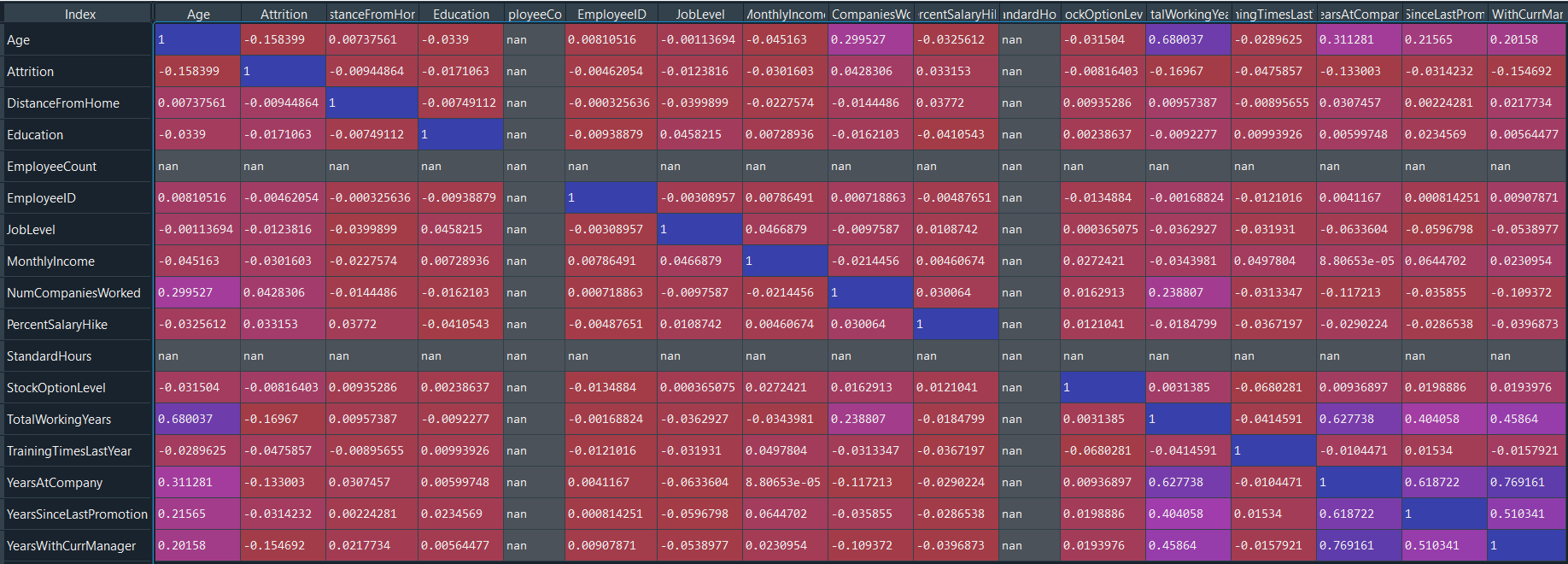
**Convert to catagorical varible as bineary**:

from sklearn import preprocessing

label\_encoder=preprocessing.LabelEncoder()

da["Attrition"]=label\_encoder.fit\_transform(da["Attrition"])

da1=da.corr()



According to the correlation analysis we conclude that YearsAtCompany,YearsWithCurrManager,

TotalWorkingYears are highly correlated with Attrition compare to other variables.

**1.correlation and hypothesis of Attrition and distance from home.**

#H0=there is no significance Attrition and distance from home.

#H1=there is significance Attrition and distance from home.

p= pearsonr(da.Attrition,da.DistanceFromHome)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.009448638515156243, 0.5317715668019558)

Accept the hypothesis

negative correlation

**2.correlation and hypothesis of Attrition and Education:**

#H0=there is no significance Attrition and Education.

#H1=there is significance Attrition and Education.

p= pearsonr(da.Attrition,da.Education)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.017106307050278727, 0.25757539308157945)

Accept the hypothesis

negative correlation

# 3.correlation and hypothesis of Attrition and JobLevel

#H0=there is no significance Attrition and JobLevel.

#H1=there is significance Attrition and JobLevel.

p= pearsonr(da.Attrition,da.JobLevel)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.012381569720790865, 0.4125489150380087)

Accept the hypothesis

negative correlation

4.correlation and hypothesis of Attrition and MonthlyIncome.

#H0=there is no significance Attrition and MonthlyIncome..

#H1=there is significance Attrition and MonthlyIncome..

p= pearsonr(da.Attrition,da.MonthlyIncome)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.030160293808460668, 0.045890862744719166)

Reject the hypothesis

negative correlation

5.correlation and hypothesis of Attrition and NumCompaniesWorked.

#H0=there is no significance Attrition and NumCompaniesWorked.

#H1=there is significance Attrition and NumCompaniesWorked.

p= pearsonr(da.Attrition,da.NumCompaniesWorked)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(0.042830567244720875, 0.004572057121620842)

Reject the hypothesis

positive correalation

6.correlation and hypothesis of Attrition and precentsalaryhike.

#H0=there is no significance Attrition and precentsalaryhike.

#H1=there is significance Attrition and precentsalaryhike.

p= pearsonr(da.Attrition,da.PercentSalaryHike)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(0.03315303713546665, 0.028192446935106235)

Reject the hypothesis

positive correalation

7.correlation and hypothesis of Attrition and StockOptionLevel.

#H0=there is no significance Attrition and StockOptionLevel.

#H1=there is significance Attrition and StockOptionLevel.

p= pearsonr(da.Attrition,da.StockOptionLevel)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.008164026684984324, 0.588999635831226)

Accept the hypothesis

negative correlation

8.correlation and hypothesis of Attrition and TotalWorkingYears.

#H0=there is no significance Attrition and TotalWorkingYears.

#H1=there is significance Attrition and TotalWorkingYears.

p= pearsonr(da.Attrition,da.TotalWorkingYears)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.16966991684723917, 1.1645434967091854e-29)

Reject the hypothesis

negative correlation

9.correlation and hypothesis of Attrition and TrainingTimesLastYear.

#H0=there is no significance Attrition and TrainingTimesLastYear .

#H1=there is significance Attrition and TrainingTimesLastYear.

p= pearsonr(da.Attrition,da.TrainingTimesLastYear)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.04758573693081737, 0.0016276603635477602)

Reject the hypothesis

negative correlation

10.correlation and hypothesis of Attrition and YearsAtCompany.

#H0=there is no significance Attrition and TrainingTimesLastYear .

#H1=there is significance Attrition and TrainingTimesLastYear.

p= pearsonr(da.Attrition,da.YearsAtCompany)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.13300261842521538, 9.476118084840815e-19)

Reject the hypothesis

negative correlation

11.correlation and hypothesis of Attrition and YearsSinceLastPromotion.

#H0=there is no significance Attrition and YearsSinceLastPromotion. .

#H1=there is significance Attrition and YearsSinceLastPromotion.

p= pearsonr(da.Attrition,da.YearsSinceLastPromotion)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.03142315056330995, 0.03752293607395154)

Reject the hypothesis

negative correlation

12.correlation and hypothesis of Attrition and YearsWithCurrManager.

#H0=there is no significance Attrition and YearsWithCurrManager.

#H1=there is significance Attrition and YearsWithCurrManager.

p= pearsonr(da.Attrition,da.YearsWithCurrManager)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.15469153690287274, 7.105369646771178e-25)

Reject the hypothesis

negative correlation

**Convert all catagorical variable to binary and numeric:**

da["Department"]=label\_encoder.fit\_transform(da["Department"])

da["BusinessTravel"]=label\_encoder.fit\_transform(da["BusinessTravel"])

da["EducationField"]=label\_encoder.fit\_transform(da["EducationField"])

da["Gender"]=label\_encoder.fit\_transform(da["Gender"])

da["JobRole"]=label\_encoder.fit\_transform(da["JobRole"])

da["MaritalStatus"]=label\_encoder.fit\_transform(da["MaritalStatus"])

da["EducationField"]=label\_encoder.fit\_transform(da["EducationField"])

13.correlation and hypothesis of Attrition and Department.

#H0=there is no significance Attrition and Department.

#H1=there is significance Attrition and Department.

p= pearsonr(da.Attrition,da.Department)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.044623751607530314, 0.0031310203124893013)

Reject the hypothesis

negative correlation

14.correlation and hypothesis of Attrition and BusinessTravel.

#H0=there is no significance Attrition and BusinessTravel.

#H1=there is significance Attrition and BusinessTravel.

p= pearsonr(da.Attrition,da.BusinessTravel)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(0.000824797217422824, 0.9564704697198688)

Accept the hypothesis

positive correalation

15.correlation and hypothesis of Attrition and EducationField.

#H0=there is no significance Attrition and EducationField.

#H1=there is significance Attrition and EducationField.

p= pearsonr(da.Attrition,da.EducationField)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.05709894322407044, 0.00015571886683318366)

Reject the hypothesis

negative correlation

16.correlation and hypothesis of Attrition and Gender.

#H0=there is no significance Attrition and Gender.

#H1=there is significance Attrition and Gender.

p= pearsonr(da.Attrition,da.Gender)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(0.01839642519444095, 0.22339966086831062)

Accept the hypothesis

positive correalation

17.correlation and hypothesis of Attrition and JobRole.

#H0=there is no significance Attrition and JobRole.

#H1=there is significance Attrition and JobRole.

p= pearsonr(da.Attrition,da.JobRole)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(0.02670795258906898, 0.07709624727519869)

Accept the hypothesis

positive correalation

18.correlation and hypothesis of Attrition and MaritalStatus.

#H0=there is no significance Attrition and MaritalStatus.

#H1=there is significance Attrition and MaritalStatus.

p= pearsonr(da.Attrition,da.MaritalStatus)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(0.16008094285320495, 1.522271982981261e-26)

Reject the hypothesis

positive correalation

19.correlation and hypothesis of Attrition and EducationField.

#H0=there is no significance Attrition and EducationField.

#H1=there is significance Attrition and EducationField.

p= pearsonr(da.Attrition,da.EducationField)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.05709894322407044, 0.00015571886683318366)

Reject the hypothesis

negative correlation

20.correlation and hypothesis of Attrition and Age

#H0=there is no significance Attrition and Age.

#H1=there is significance Attrition and Age.

p= pearsonr(da.Attrition,da.Age)

print(p)

if p[1]>=0.05:

print("Accept the hypothesis")

if p[1]<0.05:

print("Reject the hypothesis")

if p[0]==0:

print("there is no correlation")

if p[0]>0:

print("positive correalation")

if p[0]<0:

print("negative correlation")

(-0.15839867954096706, 5.1265982193780794e-26)

Reject the hypothesis

negative correlation

**Infreance from analysis:**

Here monthly income,no of companies worked,parcent salary hike,totel working years,training times years,years at company,years since lastpromotion,yearswithcurrmanager,Department,Educationfield,maritial status,education field,Age.this all variables are siginificance with Attrition.

**statistical testing:**

# From skwness of our dataset we conclude that which variables are normaly distibuted and not.

**Non parametric test:**

#here monthlyincome is not normoly distibuted so we have to use non-parametric test.

#mannwhitneyu test

#H0=there is no significance between Attrition of yes persons monthlyincome and Attrition of no persons monthlyincome.

#H1=there is significance between Attrition of yes persons monthlyincome and Attrition of no persons monthlyincome.

from scipy.stats import mannwhitneyu

stats,p= mannwhitneyu(da[da['Attrition']==0].MonthlyIncome,da[da['Attrition']==1].MonthlyIncome)

print(stats,p)

if p>=0.05:

print("Accept the hypothesis")

if p<0.05:

print("Reject the hypothesis")

1249573.5 0.06508807631576838

Accept the hypothesis

#here NumCompaniesWorked is not normoly distibuted so we have to use non-parametric test.

#mannwhitneyu test

#H0=there is no significance between Attrition of yes persons NumCompaniesWorked and Attrition of no persons NumCompaniesWorked.

#H1=there is significance between Attrition of yes persons NumCompaniesWorked and Attrition of no persons NumCompaniesWorked.

from scipy.stats import mannwhitneyu

stats,p= mannwhitneyu(da[da['Attrition']==1].NumCompaniesWorked,da[da['Attrition']==0].NumCompaniesWorked)

print(stats,p)

if p>=0.05:

print("Accept the hypothesis")

if p<0.05:

print("Reject the hypothesis")

1238814.5 0.02793197853866981

Reject the hypothesis

#here PercentSalaryHike is not normoly distibuted so we have to use non-parametric test.

#mannwhitneyu test

#H0=there is no significance between Attrition of yes persons PercentSalaryHike and Attrition of no persons PercentSalaryHike.

#H1=there is significance between Attrition of yes persons PercentSalaryHike and Attrition of no persons PercentSalaryHike.

from scipy.stats import mannwhitneyu

stats,p= mannwhitneyu(da[da['Attrition']==1].PercentSalaryHike,da[da['Attrition']==0].PercentSalaryHike)

print(stats,p)

if p>=0.05:

print("Accept the hypothesis")

if p<0.05:

print("Reject the hypothesis")

1231873.5 0.017810794960084964

Reject the hypothesis

#here TotalWorkingYears is not normoly distibuted so we have to use non-parametric test.

#mannwhitneyu test

#H0=there is no significance between Attrition of yes persons TotalWorkingYears and Attrition of no persons TotalWorkingYears.

#H1=there is significance between Attrition of yes persons TotalWorkingYears and Attrition of no persons TotalWorkingYears.

from scipy.stats import mannwhitneyu

stats,p= mannwhitneyu(da[da['Attrition']==1].TotalWorkingYears,da[da['Attrition']==0].TotalWorkingYears)

print(stats,p)

if p>=0.05:

print("Accept the hypothesis")

if p<0.05:

print("Reject the hypothesis")

895173.5 2.741211827689903e-39

Reject the hypothesis

#here TrainingTimesLastYear is not normoly distibuted so we have to use non-parametric test.

#mannwhitneyu test

#H0=there is no significance between Attrition of yes persons TrainingTimesLastYear and Attrition of no persons TrainingTimesLastYear.

#H1=there is significance between Attrition of yes persons TrainingTimesLastYear and Attrition of no persons TrainingTimesLastYear.

from scipy.stats import mannwhitneyu

stats,p= mannwhitneyu(da[da['Attrition']==1].TrainingTimesLastYear,da[da['Attrition']==0].TrainingTimesLastYear)

print(stats,p)

if p>=0.05:

print("Accept the hypothesis")

if p<0.05:

print("Reject the hypothesis")

1225582.0 0.008107344081224082

Reject the hypothesis

#here YearsAtCompany is not normoly distibuted so we have to use non-parametric test.

#mannwhitneyu test

#H0=there is no significance between Attrition of yes persons YearsAtCompany and Attrition of no persons YearsAtCompany.

#H1=there is significance between Attrition of yes persons YearsAtCompany and Attrition of no persons YearsAtCompany.

from scipy.stats import mannwhitneyu

stats,p= mannwhitneyu(da[da['Attrition']==1].YearsAtCompany,da[da['Attrition']==0].YearsAtCompany)

print(stats,p)

if p>=0.05:

print("Accept the hypothesis")

if p<0.05:

print("Reject the hypothesis")

912579.0 3.3433144809752036e-36

Reject the hypothesis

#here YearsSinceLastPromotion is not normoly distibuted so we have to use non-parametric test.

#mannwhitneyu test

#H0=there is no significance between Attrition of yes persons YearsSinceLastPromotion and Attrition of no persons YearsSinceLastPromotion.

#H1=there is significance between Attrition of yes persons YearsSinceLastPromotion and Attrition of no persons YearsSinceLastPromotion.

from scipy.stats import mannwhitneyu

stats,p= mannwhitneyu(da[da['Attrition']==1].YearsSinceLastPromotion,da[da['Attrition']==0].YearsSinceLastPromotion)

print(stats,p)

if p>=0.05:

print("Accept the hypothesis")

if p<0.05:

print("Reject the hypothesis")

1196606.0 0.00037904698157957496

Reject the hypothesis

#here YearsWithCurrManager is not normoly distibuted so we have to use non-parametric test.

#mannwhitneyu test

#H0=there is no significance between Attrition of yes persons YearsWithCurrManager and Attrition of no persons YearsWithCurrManager.

#H1=there is significance between Attrition of yes persons YearsWithCurrManager and Attrition of no persons YearsWithCurrManager.

from scipy.stats import mannwhitneyu

stats,p= mannwhitneyu(da[da['Attrition']==1].YearsWithCurrManager,da[da['Attrition']==0].YearsWithCurrManager)

print(stats,p)

if p>=0.05:

print("Accept the hypothesis")

if p<0.05:

print("Reject the hypothesis")

945958.5 5.420302388722274e-31

Reject the hypothesis.

Parametric test:

#Two sample independent separete t test.

#H0=there is no significance between Attrition of yes persons age mean and Attrition of no persons age mean.

#H1=there is significance between Attrition of yes persons age mean and Attrition of no persons age mean.

from scipy.stats import ttest\_ind

stats,p=ttest\_ind(da[da['Attrition']==1].Age,da[da['Attrition']==0].Age)

print(stats,p)

if p>=0.05:

print("Accept the hypothesis")

if p<0.05:

print("Reject the hypothesis")

-10.617111568458819 5.126598219406314e-26

Reject the hypothesis

Conclusion:

Here no of companies worked,parcent salary hike,totel working years,training times years,years at company,years since lastpromotion,yearswithcurrmanager,Age,this all varibles are significance(Attrition yes variables and attrition no variables)

Here monthly income not significance. (Attrition yes variables and attrition no variables)

So xyz company should be focus on monthly income of attrition no persons.