**Logistic Regression on Attrition Data Set**

**Loading Data and Data Treatment:**

import pandas as pd

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

Attrition\_dataset.head(2)

Out[100]:

Age Attrition ... YearsSinceLastPromotion YearsWithCurrManager

0 51 No ... 0 0

1 31 Yes ... 1 4

[2 rows x 24 columns]

Attrition\_dataset.isnull().sum()

Out[102]:

Age 0

Attrition 0

BusinessTravel 0

Department 0

DistanceFromHome 0

Education 0

EducationField 0

EmployeeCount 0

EmployeeID 0

Gender 0

JobLevel 0

JobRole 0

MaritalStatus 0

MonthlyIncome 0

NumCompaniesWorked 19

Over18 0

PercentSalaryHike 0

StandardHours 0

StockOptionLevel 0

TotalWorkingYears 9

TrainingTimesLastYear 0

YearsAtCompany 0

YearsSinceLastPromotion 0

YearsWithCurrManager 0

dtype: int64

Attrition\_dataset = Attrition\_dataset.fillna(Attrition\_dataset.mean().round())

Attrition\_dataset.columns

Out[104]:

Index(['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHome',

'Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gender',

'JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome',

'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHours',

'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear',

'YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager'],

dtype='object')

Attrition\_dataset.dtypes

Out[105]:

Age int64

Attrition object

BusinessTravel object

Department object

DistanceFromHome int64

Education int64

EducationField object

EmployeeCount int64

EmployeeID int64

Gender object

JobLevel int64

JobRole object

MaritalStatus object

MonthlyIncome int64

NumCompaniesWorked float64

Over18 object

PercentSalaryHike int64

StandardHours int64

StockOptionLevel int64

TotalWorkingYears float64

TrainingTimesLastYear int64

YearsAtCompany int64

YearsSinceLastPromotion int64

YearsWithCurrManager int64

dtype: object

**Encoding Categorical Features:**

from sklearn import preprocessing

label\_encoder = preprocessing.LabelEncoder()

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

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

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

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

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

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

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

**Model Generation:**

X = Attrition\_dataset[['Age','BusinessTravel', 'Department', 'DistanceFromHome',

'Education', 'EducationField','Gender',

'JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome',

'NumCompaniesWorked','PercentSalaryHike',

'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear',

'YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManager']]

Y = Attrition\_dataset[['Attrition']]

import statsmodels.api as sm

X1 = sm.add\_constant(X)

Logistic = sm.Logit(Y, X1)

result = Logistic.fit()

Optimization terminated successfully.

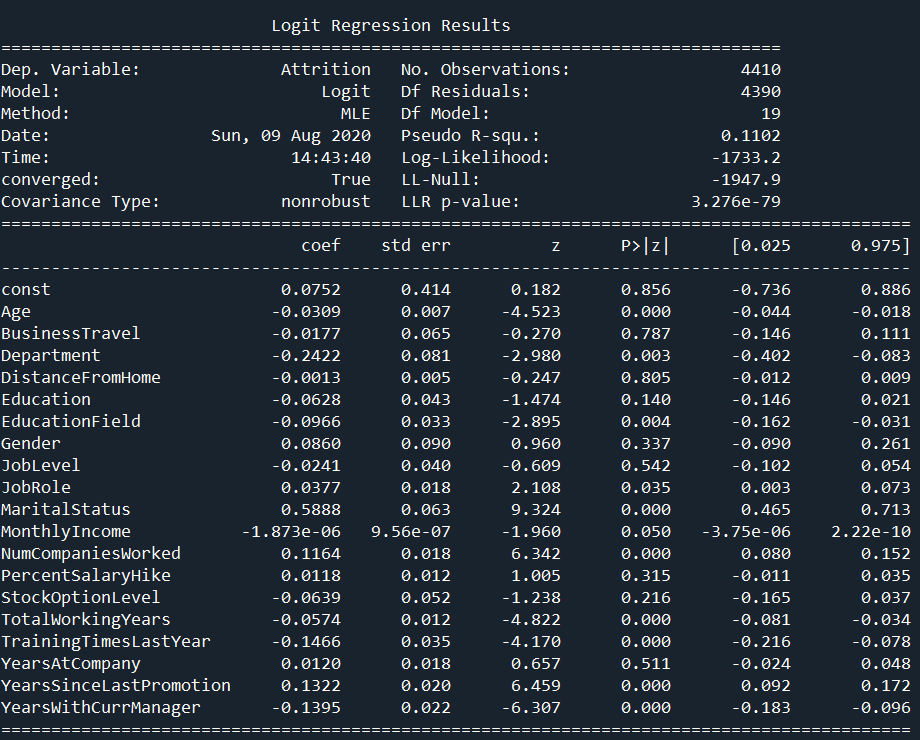
Current function value: 0.393008

Iterations 7

result.summary()

Out[122]:

<class 'statsmodels.iolib.summary.Summary'>



**Inference:**

Features **'Age', 'Department', 'EducationField','Gender', 'JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome', 'NumCompaniesWorked', 'TotalWorkingYears', 'TrainingTimesLastYear', 'YearsSinceLastPromotion', 'YearsWithCurrManager'** are significantly important for analysis the attrition in the company.

The probability for Attrition P(Y/N) is calculated as follows

P(Y/N) = 1/(1+e^(-k))

Where k = -0.0752 +(-0.0309)(Age) +(-0.0177)( 'BusinessTravel') +(-0.2422)( 'Department') +(-0.0013)( 'DistanceFromHome') +(-0.0628)( 'Education') +(-0.0966)( 'EducationField') +(0.0860)( 'Gender') +(-0.0241)( 'JobLevel') +(0.0377)( 'JobRole') +(0.5888)( 'MaritalStatus') +(-1.873e^(-06))( 'MonthlyIncome') +(0.1164)( 'NumCompaniesWorked') +(0.0118)( 'PercentSalaryHike') +(-0.0639)( 'StockOptionLevel') +(-0.0574)( 'TotalWorkingYears') +(-0.1466)( 'TrainingTimesLastYear') +(0.0120)( 'YearsAtCompany') +(0.1322)( 'YearsSinceLastPromotion') +(-0.1395)( 'YearsWithCurrManager')