**EX.NO: 04**

**Implementation of Z-Test – using Titanic case study**

**AIM:**

To write a program using Implementation of Z-Test -using Titanic case study

**Algorithm:**

Step 1: Evaluate the data distribution.

Step 2: Formulate Hypothesis statement symbolically

Step 3: Define the level of significance (alpha)

Step 4: Calculate Z test statistic or Z score.

Step 5: Derive P-value for the Z score calculated.

Step 6: Make decision:

Step 6.1: P-Value <= alpha, then we reject H0.

Step 6.2: If P-Value > alpha, Fail to reject H0.

**4a****) Overview Of Data Set**

titanic\_train\_data.describe()

**4b) Titanic Train Data**

temp= titanic\_train\_data[titanic\_train\_data['Age'].notna ()].Age

sns.distplot(temp)

**4c) Survived Data**

survived\_data = titanic\_train\_data(titanic\_train\_data['Survived']==1]

survived\_data[survived\_data['Age'].notna ()].head()

**4d****) Sample List**

sample\_list = []

for i in range (60):

sample\_list\_temp = np. random. Choice (survived\_data[survived\_data['Age'] .notna()].Age,60) .mean()

sample\_list.append(sample\_list\_temp)

sns.distplot (sample\_list)

**4e****. # Z test**

from statsmodels.stats.weightstats import ztest

from statsmodels.stats.weightstats import zconfint

#Formulating the hypothesis statement,

#HØ: Average age of passengers in Titanic is less than 28:µ0 <=28

#HA: New research claims mean age is greater than 28: µl > 28

mean\_hØ= 28

#calculation using Library

ztest Score, p\_value= ztest (sample\_list, value = mean\_hØ, alternative='larger")

print('p\_value’,p\_value)

print('ztest Score’,ztest Score)

if (p\_value < 0.05): #The smaller the p-value, the stronger the evidence to reject the HØ

print ()

print("Reject HØ with", (1-p\_value) \*100, '% level of confidence');

print ()

else:

print("Fail to reject HØ");

print()

**4f)** **survived male data**

sns.distplot (survived\_male\_data[survived\_male\_data.Age.notna ()1.Age)

**4g****)**  **Sample List male Using For Range (60)**

sample\_list\_male = []

for i in range (60):

sample\_list\_temp = np . random. Choice ( survived\_male\_data[survived\_male\_data.Age.notna()]. Age,60). Mean ()

sample\_list\_male.append(sample\_list\_temp)

sns.distplot (sample\_list\_male)

**4h) survived\_female\_data**

sns.distplot (survived\_female\_data[survived\_female\_data.Age.notna()]. Age)

**4i****) Sample List Female Using For Range (60)**

sample\_list\_female = [ ]

for i in range(60):

sample\_list\_temp = np . random . choice ( survived\_female\_data[survived\_female\_date.Age . notna ()] . Age,60). Mean()

sample\_list\_female.append(sample\_list\_temp)

sns.distplot (sample\_list\_female)

**4j****) # Z test**

from statsmodels.stats.weightstats import ztest

from statsmodels.stats.weightstats import zconfint

#Formulating the hypothesis statement,

#HO:No difference in mean age of male & female passengers who survived: µ

\_male = µ\_female

#HA:There is dprifference in mean age of male & fenale passengers who survived male< > µ\_female

#calculation using library

Ztest\_Score, p\_value = ztest(x1=sample\_list\_male, x2 sample\_list\_female, value = 0, alternatives two-sided"]

print('p\_value,p\_value)

print('ztest Score',ztest Score)

if (p\_value < 0.05): #The smaller the p-value, the stronger the evidence to reject the HØ

print()

print("Reject HØ with", (1-p\_value)\*100,' %level of confidence’);

print()

else:

print("Fail to reject HØ ");

print()

#Confidence interval, value shifts the confidence interval so it is centered at x1 mean-value

lower, upper = zconfint (x1=sample\_list\_male, value = 0,alpha=0.05, alternative = ‘two-sided")

print("with 95% considence interval we can say thta, Passenger Mean age is between, lower,"and", upper)

lower, upper = zconfint(x1=sample\_list\_female, value = 0,alpha 0.05, alternative="two-sided")

print("With 95% considence interval we can say thta, Passenger Mean age is between", lower, "and", upper)

**4k) #z test**

from statsmodels.stats.weightstats import ztest

from statsmodels.stats.weightstats import zconfint

#Formulating the hypothesis statement,

#HO: μ\_male≤ μ\_female

#H1: μ\_male> μ\_female

#calculation using library

ztest\_score,p\_value=ztest(x1=sample\_list\_male,x2=sample\_list\_female,value=0, alternative=’larger’)

print('p\_value,p\_value)

print('ztest Score',ztest Score)

if (p\_value < 0.05): #The smaller the p-value, the stronger the evidence to reject the HO

print()

print("Reject HØ with", (1-p\_value)\*100,’%level of confidence');

print()

else:

print("Fail to reject HØ’’);

print()

#Confidence interval, value shifts the confidence interval so it is centered at x1\_ mean – value

lower, upper = zconfint(x1=sample\_list\_male, value = 0,alpha=0.05, alternative='two-sided')

print("With 95% considence interval we can say thta, Passenger Mean age is between", lower,"and", upper)

lower, upper = zconfint (x1=sample\_list\_female, value = 0,alpha=0.05, alternatives=’ two-sided’)

print("with 95% considence interval we can say thta, Passenger Hean age is between", lower, "and", upper)

**4L)One Proportion Z-Test**

from statsmodels.stats.weightstats import ztest

from statsmodels.stats.weightstats import zconfint

from statsmodels.stats.proportion import proportions\_ztest

from statsmodels.stats.proportion import proportion\_confint

#HØ: p <= Ø.5Ø and H1: p>Ø.5Ø

p = Ø.5Ø

ztestScore,p\_value=proportions\_ztest(count=153,nobs=survived\_data.Age\_range.count(),value=p, alternative ‘large’

print("Below is the calculation using Library")

print('p\_value,p\_value)

print('ztest Score’,ztest Score)

if (p\_value < 0.05): #The smaller the p-value, the stronger the evidence to reject the HØ

print("Reject HØ with", (1-p\_value)\*100, '%level of confidence');

print()

else:

print("Fail to reject HØ ");

print()

#z =(p-pØ) / pØ (1-pØ)/n

z\_score= ((153/290)-0.5)/(math.sqrt((0.5+0.5)/714))

print("z\_score,z\_score)

lower,upper=proportion\_confint(count=153,nobs=survived\_data.Age\_range.count(), method="normal)

print (lower \*100, "% to”, upper\*100, “% of passengers who survived lie in the range of 20-40 age ")

**4m) # One Proportion Z-Test**

from statsmodels.stats.weightstats import ztest

from statsmodels.stats.weightstats import zconfint

from statsmodels.stats.proportion import proportions\_ztest

from statsmodels.stats.proportion import proportion\_confint

#H Ø: p <= Ø.5 Ø

#H1: p> Ø.5 Ø

P = Ø.5 Ø

Ztest\_ Score,p\_value= proportions\_ztest (count=385, nobs=titanic\_train\_data. Age\_range.count(),value=p,alternative=’large

print("Below is the calculation using Library")

print('p\_value.p\_value)

print( ztest Scoreztest Score)

if (p\_value < 0.05): #The smaller the p-value, the stronger the evidence to reject the H Ø

print("Reject H Ø with’’. (1-p\_value)\*100,’% level of confidence");

print()

else:

print("Fail to reject H Ø ");

print()

#z = (p-p Ø) / p Ø (1-p Ø)/n

z score = (0.539-0.5)/(math.sqrt((0.5\*0.5)/714))

print("score ,z\_score)

Lower, upper = proportion\_confint (count=385, nobs =titanic\_train\_data.Age\_range.count(), method="normal")

print (lower\*100, “% to, upper\*100,’’% of passengers lie in range of 20-40 age ")

**Output;**

**Results ;**