REPORT

-Abhimanyu Agarwal ,ECE 20875

Problem – 1

Results and Conclusions

Question1: Suppose the instructor of the course is convinced that the mean engagement of students who become knowledgeable in the material (i.e., the eng1 population) is 0.75. Formulate null and alternative hypotheses for a statistical test that seeks to challenge this belief. What type of test can be used?

Answer1:

```
H0: mean of the student who became knowledgeable is equal to 0.75

H1: mean of the students who become knowledgeable is not equal to 0.75
```

Question2: Carry out this statistical test using the eng1 sample. Report the sample size, the sample mean, the standard error, the standard score, and the p-value. Are the results significant at a level of 0.1? How about 0.05? How about 0.01?What (if anything) can we conclude?

Answer2:

The sample size is : 937

The sample mean is: 0.7430304110448239

The sample standard of deviation is: 0.12712605317614

The standard error is: 0.004153027288269652

The standard score is: -1.6781948375012814

The standard p-value is: 0.09330906925243751

Conclusions :

The sample is significant as 10% level of significance, we fail to reject H0.

Code Snippet:

```
eng1size = np.size(eng1data)
eng1avg = np.mean(eng1data)
eng1sd = np.std(eng1data, ddof=1)
eng1se = (eng1sd / (np.sqrt(eng1size)))
z_c = (eng1avg - 0.75) / eng1se
p_value = 2 * norm.cdf(-abs(z_c))
```

Question3: Determine the largest standard of error for which the test will be significant at a level of 0.05.What is the corresponding minimum sample size?

Answer3:

The largest standard of error is: 0.0035559780741642272

The corresponding minimum sample size is: 1279

Code Snippet:

```
p_value_new = 0.05
z_c_new = norm.ppf(p_value_new / 2)
SE_new = (eng1avg - 0.75) / z_c_new                    #Largest standard error
samplesize = (eng1sd / SE_new)                         #Sample size
size = m.ceil(samplesize **2)
```

Question4: Suppose the instructor is also convinced that the mean engagement is different between students who become knowledgeable(the eng1 population) and those who do not (the eng0 population). Formulate null and alternative hypotheses that seek to validate this belief. What type of test can be used ?

Answer4:

H0: the mean engagement is different between students who become knowledgeable and those who do not is equal.

H1: the mean engagement is different between students who become knowledgeable and those who do not is not equal.

Question5: Carry out this statistical test using the eng0 and eng1 samples. Report the sample sizes, the sample means, the standard error, the z-score, and the p-value. Are the results significant? What (if anything) can we conclude?

Answer5:

The sample mean for 'eng1data.txt' is: 0.7430304110448239

The sample mean for 'eng0data.txt' is : 0.638854507735914

The standard error is: 0.007065420910043284

The z-score value is: -14.588784540028351

The p-value is: 3.3104307168195455e-48

Conclusions: We reject the null hypothesis due to such a small p-value which approximates so close to zero.

Code Snippet:

```
#Are the results significant? What (if anyt
eng1avg = np.mean(eng1data)
eng0avg = np.mean(eng0data)
netavg = (eng1avg - eng0avg)

eng1sd = np.std(eng1data, ddof=1)
eng0sd = np.std(eng0data, ddof=1)

eng1var = (eng1sd ** 2)
eng1size = (np.size(eng1data))
eng1div = ((eng1var) / eng1size)

eng0var = (eng0sd ** 2)
eng0size = (np.size(eng0data))
eng0div = ((eng0var) / eng0size)

sum = eng0div + eng1div
final = (np.sqrt(sum))

z_c = (netavg / final)
p = 2 * norm.cdf(-abs(z_c))
```

Problem - 2

Results and Conclusions

Question1: Use the sample to construct a 95% confidence interval for the number of points by which the team wins on average. Report the sample mean, the standard error, the standard statistic, and the interval.

Answer1:

The sample mean is: 7.363636

The standard error is: 5.0762776757504415

```
The standard statistic is: 1.8124611228107335

The interval is: (-1.8369195722533433,16.56419229952607)
```

Code Snippet:

```python
pointsize = len(data)
pointavg = np.mean(data)
pointsd =  np.std(data, ddof = 1)
confidence = 0.95
pointerror = pointsd / (pointsize ** 0.5)
pvalue = (1 - ((1-confidence) / 2))
t_c = t.ppf(pvalue,pointsize-1)
C_L = pointavg - ((t_c * pointsd) / (pointsize ** 0.5))
C_U = pointavg + ((t_c * pointsd) / (pointsize ** 0.5))
confidenceinterval = (C_L,C_U)

print(' ')
print('The sample mean is: {}'.format(pointavg))
print('The standard error is: {} '.format(pointerror))
print('The standard statistic is: {} '.format(t_c))
print('The interval is: {} '.format(confidenceinterval))
```

Question2: Repeat part 1 for a 90% confidence interval. Compare your results.

Answer2:

The sample mean is: 7.3636

The point error is: 5.0762776757504415

The standard statistic is: -abs(1.8124611228107335)

The interval is: (-1.8369195722533433,16.56419229952607)

The t_c (Standard Statistic) has a relatively lower value which makes the interval precise as our confidence level decreases.

As t value decreases, the lower bound of our confidence level increase and upper bound decreases.

The interval is smaller than the first one that was, 95%

Code Snippet:

```
pointavg = np.mean(data)
pointsd =  np.std(data, ddof = 1)
confidence = 0.90
pointerror = pointsd / (pointsize ** 0.5)
pvalue = (1 - ((1-confidence) / 2))
t_c = t.ppf(pvalue,pointsize-1)
C_L = pointavg - ((t_c * pointsd) / (pointsize ** 0.5))
C_U = pointavg + ((t_c * pointsd) / (pointsize ** 0.5))
confidenceinterval = (C_L,C_U)
```

Question3: Repeat part 1 if you are told that the population standard deviation is 16.836. Compare your results.

Answer3:

Since standard deviation is given, we use the z-test.

The sample mean is: 7.3636

The standard error is: 5.0762776757504415

The standard statistic is : -abs(1.959963984540054)

The confidence- interval is: (-2.585621007795268,17.312893735067995)

The results are similar because the standard deviation in both the cases, that is, part 2.1 and 2.3 are the same.

The interval is smaller than 95% interval and larger than 90% interval.

Code Snippet:

```
pointavg = np.mean(data)
pointsd = 16.836
confidence = 0.95
pointerror = pointsd / (pointsize ** 0.5)
pvalue = (1 - ((1-confidence) / 2))
z_c =norm.ppf(pvalue)
C_L = pointavg - ((z_c * pointsd) / (pointsize ** 0.5))
C_U = pointavg + ((z_c * pointsd) / (pointsize ** 0.5))
confidenceinterval = (C_L,C_U)
```

Question4: With what level of confidence can we say that the team is expected to win on average?(Hint: The interval must be strictly positive.)

Answer4:

The average is: 7.3636

Mathematically,

We apply the limiting condition (Minimum value being 0 in the limiting case)
7.36 – Z_value * (5.07626278) = 0

Z_value = 1.4498

We choose the z-distribution since the population mean is 0 and the distribution is about the mean)

p = z_value * (cdf(1.4498))

p = 0.926

(1-((1 – C))) / 2 = 0.926 [Formula applied: 1-2p]

1-((1-C)/2) = p

-((1-C)/2) = p – 1

(1-C)/2 = 1 – p

1-C = 2 - 2p

-C = 1 - 2p

C = 2p - 1

C = 0.8531

The above logic has been applied to code by me to solve the problem using Python.

The confidence level should be around 85.3%

The interval is: (0.0001469226257935219, 14.727125804646933)

Code Snippet:

```
pointsize = len(data)
pointavg = np.mean(data)
pointsd =  np.std(data, ddof = 1)
pointerror = pointsd / (pointsize ** 0.5)
pointz_new = (pointavg) / pointerror
pvalue = norm.cdf(pointz_new)
C = 2 * pvalue - 1

C_L_new = pointavg - ((pointz_new * pointerror))
C_U_new= pointavg + ((pointz_new * pointerror))
confidenceinterval = (C_L_new,C_U_new)
print(' ')
print('The mean is {}'. format(pointavg))
print('The level of confidence is {}'.format(C*100))
print('The interval is {}'.format(confidenceinterval))
```