Week 4 Lab : Hypothesis Testing

Import all the relevant libraries and functions

[1]

0s

import numpy as np  
from scipy import stats  
import math as mt  
import random as rnd

Calculating critical z value in Python

[2]

from scipy.stats import norm  
  
def critical\_z\_value(p):  
    norm\_dist = norm(loc=0.0, scale=1.0)  
    left\_tail\_area = (1.0 - p) / 2.0  
    upper\_area = 1.0 - ((1.0 - p) / 2.0)  
    return norm\_dist.ppf(left\_tail\_area), norm\_dist.ppf(upper\_area)  
  
print(critical\_z\_value(p=.95))  
# (-1.959963984540054, 1.959963984540054)

Calculating a confidence interval in Python

[3]

from math import sqrt  
from scipy.stats import norm  
  
  
def critical\_z\_value(p):  
    norm\_dist = norm(loc=0.0, scale=1.0)  
    left\_tail\_area = (1.0 - p) / 2.0  
    upper\_area = 1.0 - ((1.0 - p) / 2.0)  
    return norm\_dist.ppf(left\_tail\_area), norm\_dist.ppf(upper\_area)  
  
  
def confidence\_interval(p, sample\_mean, sample\_std, n):  
    # Sample size must be greater than 30  
  
    lower, upper = critical\_z\_value(p)  
    lower\_ci = lower \* (sample\_std / sqrt(n))  
    upper\_ci = upper \* (sample\_std / sqrt(n))  
  
    return sample\_mean + lower\_ci, sample\_mean + upper\_ci  
  
print(confidence\_interval(p=.95, sample\_mean=64.408, sample\_std=2.05, n=31))  
# (63.68635915701992, 65.12964084298008)

Past studies have shown that the mean recovery time for a cold is 18 days, with a standard deviation of 1.5 days, and follows a normal distribution.

This means there is approximately 95% chance of recovery taking between 15 and 21 days. Calculating the probability of recovery between 15 and 21 days

[ ]

from scipy.stats import norm  
  
# Cold has 18 day mean recovery, 1.5 std dev  
mean = 18  
std\_dev = 1.5  
  
# 95% probability recovery time takes between 15 and 21 days.  
x = norm.cdf(21, mean, std\_dev) - norm.cdf(15, mean, std\_dev)  
  
print(x) # 0.9544997361036416

The area up to that 16-day mark is our p-value, which is .0912, and we calculate it in Python. Calculating the one-tailed p-value

[ ]

from scipy.stats import norm

# Cold has 18 day mean recovery, 1.5 std dev

mean = 18

std\_dev = 1.5

# Probability of 16 or less days

p\_value = norm.cdf(16, mean, std\_dev)

print(p\_value) # 0.09121121972586788

Calculating a range for a statistical significance of 5%, for two tailed test.

[ ]

from scipy.stats import norm

# Cold has 18 day mean recovery, 1.5 std dev

mean = 18

std\_dev = 1.5

# What x-value has 2.5% of area behind it?

x1 = norm.ppf(.025, mean, std\_dev)

# What x-value has 97.5% of area behind it

x2 = norm.ppf(.975, mean, std\_dev)

print(x1) # 15.060054023189918

print(x2) # 20.93994597681008

Solve the following question using the codes given above:

Jeffrey, as an eight-year old, established a mean time of 16.43 seconds for swimming the 25-yard freestyle, with a standard deviation of 0.8 seconds. His dad, Frank, thought that Jeffrey could swim the 25-yard freestyle faster using goggles. Frank bought Jeffrey a new pair of expensive goggles and timed Jeffrey for 15 25-yard freestyle swims. For the 15 swims, Jeffrey's mean time was 16 seconds. Frank thought that the goggles helped Jeffrey to swim faster than the 16.43 seconds. Conduct a hypothesis test using a preset α = 0.05. Assume that the swim times for the 25-yard freestyle are normal.