# Z TEST

**One tailed test**

*# Import the necessary libraries*

**import** **numpy** **as** **np**

**import** **scipy.stats** **as** **stats**

*# Given information*

sample\_mean = 110

population\_mean = 100

population\_std = 15

sample\_size = 50

alpha = 0.05

*# compute the z-score*

z\_score = (sample\_mean-population\_mean)/(population\_std/np.sqrt(50))

print('Z-Score :',z\_score)

*# Approach 1: Using Critical Z-Score*

*# Critical Z-Score*

z\_critical = stats.norm.ppf(1-alpha)

print('Critical Z-Score :',z\_critical)

*# Hypothesis*

**if** z\_score > z\_critical:

print("Reject Null Hypothesis")

**else**:

print("Fail to Reject Null Hypothesis")

*# Approach 2: Using P-value*

*# P-Value : Probability of getting less than a Z-score*

p\_value = 1-stats.norm.cdf(z\_score)

print('p-value :',p\_value)

*# Hypothesis*

**if** p\_value < alpha:

print("Reject Null Hypothesis")

**else**:

print("Fail to Reject Null Hypothesis")

**Output**:

Z-Score : 4.714045207910317Critical Z-Score : 1.6448536269514722Reject Null Hypothesisp-value : 1.2142337364462463e-06Reject Null Hypothesis

Two tailed test

**import** **numpy** **as** **np**

**import** **scipy.stats** **as** **stats**

*# Group A (Offline Classes)*

n1 = 50

x1 = 75

s1 = 10

*# Group B (Online Classes)*

n2 = 60

x2 = 80

s2 = 12

*# Null Hypothesis = mu\_1-mu\_2 = 0*

*# Hypothesized difference (under the null hypothesis)*

D = 0

*# Set the significance level*

alpha = 0.05

*# Calculate the test statistic (z-score)*

z\_score = ((x1 - x2) - D) / np.sqrt((s1\*\*2 / n1) + (s2\*\*2 / n2))

print('Z-Score:', np.abs(z\_score))

*# Calculate the critical value*

z\_critical = stats.norm.ppf(1 - alpha/2)

print('Critical Z-Score:',z\_critical)

*# Compare the test statistic with the critical value*

**if** np.abs(z\_score) > z\_critical:

print("""Reject the null hypothesis.

There is a significant difference between the online and offline classes.""")

**else**:

print("""Fail to reject the null hypothesis.

There is not enough evidence to suggest a significant difference between the online and offline classes.""")

**T test**

**One sample**

import scipy.stats as stats

# Sample data

sample\_mean = 110

population\_mean = 100

sample\_std = 15 s

ample\_size = 50

alpha = 0.05

# Calculate the t-statistic

t\_statistic = (sample\_mean - population\_mean) / (sample\_std / np.sqrt(sample\_size))

print('t-statistic:', t\_statistic)

# Calculate the critical t-value

t\_critical = stats.t.ppf(1 - alpha, df=sample\_size - 1)

print('Critical t-value:', t\_critical)

# Hypothesis testing using critical value approach

if abs(t\_statistic) > t\_critical:

print("Reject the null hypothesis")

else:

print("Fail to reject the null hypothesis")

INDEPENDENT SAMPLE T TEST

import numpy as np

import scipy.stats as stats

# Sample data for two groups

group\_a = [10, 15, 12, 18, 20]

group\_b = [8, 12, 14, 16, 18]

# Calculate sample means and standard deviations

mean\_a = np.mean(group\_a)

mean\_b = np.mean(group\_b)

std\_a = np.std(group\_a, ddof=1)

std\_b = np.std(group\_b, ddof=1)

n\_a = len(group\_a)

n\_b = len(group\_b)

# Calculate the t-statistic using the given formula

t\_statistic = (mean\_a - mean\_b) / np.sqrt((std\_a\*\*2 / n\_a) + (std\_b\*\*2 / n\_b))

# Calculate degrees of freedom (using Welch-Satterthwaite approximation)

df = ((std\_a\*\*2 / n\_a + std\_b\*\*2 / n\_b)\*\*2) / ((std\_a\*\*2 / n\_a)\*\*2 / (n\_a - 1) + (std\_b\*\*2 / n\_b)\*\*2 / (n\_b - 1))

# Calculate the critical t-value

alpha = 0.05

t\_critical = stats.t.ppf(1 - alpha/2, df=df)

print("t-statistic:", t\_statistic)

print("t-critical:", t\_critical)

# Hypothesis testing using critical value approach

if abs(t\_statistic) > t\_critical:

print("Reject the null hypothesis. There is a significant difference between the two groups.")

else:

print("Fail to reject the null hypothesis. There is no significant difference between the two groups.")

PAIRED HYPOTHESIS  
import numpy as np

import scipy.stats as stats

# Sample data for paired samples

group\_a = [10, 15, 12, 18, 20]

group\_b = [8, 12, 14, 16, 18]

# Calculate the difference between paired observations

differences = np.array(group\_a) - np.array(group\_b)

# Calculate the mean and standard deviation of the differences

mean\_diff = np.mean(differences)

std\_diff = np.std(differences, ddof=1)

n = len(differences)

# Calculate the t-statistic

t\_statistic = (mean\_diff - 0) / (std\_diff / np.sqrt(n)) # Assuming null hypothesis: μd = 0

# Calculate degrees of freedom

df = n - 1

# Calculate the critical t-value (two-tailed test)

alpha = 0.05

t\_critical = stats.t.ppf(1 - alpha/2, df=df)

print("t-statistic:", t\_statistic)

print("Critical t-value:", t\_critical)

# Hypothesis testing

if abs(t\_statistic) > t\_critical:

print("Reject the null hypothesis. There is a significant difference between the paired samples.")

else:

print("Fail to reject the null hypothesis. There is no significant difference between the paired samples.")

ONE WAY ANOVA

import numpy as np

# Sample data for three groups

group\_a = [10, 15, 12, 18, 20]

group\_b = [8, 12, 14, 16, 18]

group\_c = [15, 20, 18, 22, 25]

# Combine the groups into a single list

data = group\_a + group\_b + group\_c

# Create a list of group labels

groups = ['A'] \* len(group\_a) + ['B'] \* len(group\_b) + ['C'] \* len(group\_c)

# Calculate the overall mean

overall\_mean = np.mean(data)

# Calculate the sum of squares between groups (SSB)

ssb = sum([len(group) \* (np.mean(group) - overall\_mean)\*\*2 for group in [group\_a, group\_b, group\_c]])

# Calculate the sum of squares within groups (SSW)

ssw = sum([sum((x - np.mean(group))\*\*2) for group in [group\_a, group\_b, group\_c] for x in group])

# Calculate degrees of freedom

df\_between = len(set(groups)) - 1 # Number of groups - 1

df\_within = len(data) - len(set(groups)) # Total number of observations - number of groups

# Calculate mean square between groups (MSB) and mean square within groups (MSW)

msb = ssb / df\_between

msw = ssw / df\_within

# Calculate the F-statistic

f\_statistic = msb / msw

# Calculate the p-value

p\_value = 1 - stats.f.cdf(f\_statistic, df\_between, df\_within)

print("F-statistic:", f\_statistic)

print("p-value:", p\_value)

# Interpret the results

alpha = 0.05 # Significance level

if p\_value < alpha:

print("Reject the null hypothesis. There is a significant difference between at least two groups.")

else:

print("Fail to reject the null hypothesis. There is no significant difference between the groups.")