# Statistics Part 2

## Assignment Questions

PW SKILLS

# Statistics Part 2

1. What is hypothesis testing in statistics?
2. What is the null hypothesis, and how does it differ from the alternative hypothesis?
3. What is the significance level in hypothesis testing, and why is it important?
4. What does a P-value represent in hypothesis testing?
5. How do you interpret the P-value in hypothesis testing?
6. What are Type 1 and Type 2 errors in hypothesis testing?
7. What is the difference between a one-tailed and a two-tailed test in hypothesis testing?
8. What is the Z-test, and when is it used in hypothesis testing?
9. How do you calculate the Z-score, and what does it represent in hypothesis testing?
10. What is the T-distribution, and when should it be used instead of the normal distribution?
11. What is the difference between a Z-test and a T-test?
12. What is the T-test, and how is it used in hypothesis testing?
13. What is the relationship between Z-test and T-test in hypothesis testing?
14. What is a confidence interval, and how is it used to interpret statistical results?
15. What is the margin of error, and how does it affect the confidence interval?
16. How is Bayes' Theorem used in statistics, and what is its significance?
17. What is the Chi-square distribution, and when is it used?
18. What is the Chi-square goodness of fit test, and how is it applied?
19. What is the F-distribution, and when is it used in hypothesis testing?
20. What is an ANOVA test, and what are its assumptions?
21. What are the different types of ANOVA tests?
22. What is the F-test, and how does it relate to hypothesis testing?

# Practical Part - 1

1. Write a Python program to generate a random variable and display its value.
2. Generate a discrete uniform distribution using Python and plot the probability mass function (PMF).
3. Write a Python function to calculate the probability distribution function (PDF) of a Bernoulli distribution.
4. Write a Python script to simulate a binomial distribution with n=10 and p=0.5, then plot its histogram.
5. Create a Poisson distribution and visualize it using Python.
6. Write a Python program to calculate and plot the cumulative distribution function (CDF) of a discrete uniform distribution.
7. Generate a continuous uniform distribution using NumPy and visualize it.
8. Simulate data from a normal distribution and plot its histogram.
9. Write a Python function to calculate Z-scores from a dataset and plot them.
10. Implement the Central Limit Theorem (CLT) using Python for a non-normal distribution.

11. Simulate multiple samples from a normal distribution and verify the Central Limit Theorem.
12. Write a Python function to calculate and plot the standard normal distribution (mean = 0, std = 1).
13. Generate random variables and calculate their corresponding probabilities using the binomial distribution.
14. Write a Python program to calculate the Z-score for a given data point and compare it to a standard normal distribution.
15. Implement hypothesis testing using Z-statistics for a sample dataset.
16. Create a confidence interval for a dataset using Python and interpret the result.
17. Generate data from a normal distribution, then calculate and interpret the confidence interval for its mean.
18. Write a Python script to calculate and visualize the probability density function (PDF) of a normal distribution.
19. Use Python to calculate and interpret the cumulative distribution function (CDF) of a Poisson distribution.
20. Simulate a random variable using a continuous uniform distribution and calculate its expected value.
21. Write a Python program to compare the standard deviations of two datasets and visualize the difference.
22. Calculate the range and interquartile range (IQR) of a dataset generated from a normal distribution.
23. Implement Z-score normalization on a dataset and visualize its transformation.
24. Write a Python function to calculate the skewness and kurtosis of a dataset generated from a normal distribution.

# Practical Part - 2

1. Write a Python program to perform a Z-test for comparing a sample mean to a known population mean and interpret the results.
2. Simulate random data to perform hypothesis testing and calculate the corresponding P-value using Python.
3. Implement a one-sample Z-test using Python to compare the sample mean with the population mean.
4. Perform a two-tailed Z-test using Python and visualize the decision region on a plot.
5. Create a Python function that calculates and visualizes Type 1 and Type 2 errors during hypothesis testing.
6. Write a Python program to perform an independent T-test and interpret the results.
7. Perform a paired sample T-test using Python and visualize the comparison results.
8. Simulate data and perform both Z-test and T-test, then compare the results using Python.
9. Write a Python function to calculate the confidence interval for a sample mean and explain its significance.
10. Write a Python program to calculate the margin of error for a given confidence level using sample data.
11. Implement a Bayesian inference method using Bayes' Theorem in Python and explain the process.
12. Perform a Chi-square test for independence between two categorical variables in Python.
13. Write a Python program to calculate the expected frequencies for a Chi-square test based on observed data.
14. Perform a goodness-of-fit test using Python to compare the observed data to an expected distribution.