RN SHETTY TRUST®

RNS INSTITUTE OF TECHNOLOGY

Affiliated to VTU, Recognized by GOK, Approved by AICTE, New Delhi
(NAAC 'A+ Grade' Accredited, NBA Accredited (UG - CSE, ECE, ISE, EIE and EEE)
Channasandra, Dr. Vishnuvardhan Road, Bengaluru - 560 098
Ph:(080)28611880,28611881 URL: www.rnsit.ac.in

DEPARTMENT OF CSE (DATA SCIENCE)

Statistical Machine Learning for Data Science

[BAD702]

LAB MANUAL

(As per Visvesvaraya Technological University Course type - IPCC)

Compiled by

DEPARTMENT OF CSE (DATA SCIENCE)

R N S Institute of Technology

Bengaluru-98

Name: _____

USN: _____

RN SHETTY TRUST®

RNS INSTITUTE OF TECHNOLOGY

Affiliated to VTU, Recognized by GOK, Approved by AICTE, New Delhi
(NAAC 'A+ Grade' Accredited, NBA Accredited (UG - CSE, ECE, ISE, EIE and EEE)
Channasandra, Dr. Vishnuvardhan Road, Bengaluru - 560 098
Ph:(080)28611880,28611881 URL: www.rnsit.ac.in

Estd : 2001

DEPARTMENT OF CSE (DATA SCIENCE)

VISION OF THE DEPARTMENT

Empowering students to solve complex real-time computing problems involving high volume multi-dimensional data.

MISSION OF THE DEPARTMENT

- Provide quality education in both theoretical and applied Computer Science to solve real world problems.
- Conduct research to develop algorithms that solve complex problems involving multi-dimensional high-volume data through intelligent inferencing.
- Develop good linkages with industry and research organizations to expose students to global problems and find optimal solutions.
- Creating confident Graduates who can contribute to the nation through high levels of commitment following ethical practices and with integrity.

Disclaimer

Trademark

Edition: 2024- 25

Document Owner

The primary contact for questions regarding this document is:

Author(s):
1. Dr. Mahantesh K
2. Ms. Smitha B A

Department: **CSE (Data Science)**

Contact email ids: mahantesh.k@rnsit.ac.in

COURSE OUTCOMES

Course Outcomes: At the end of this course, students are able to:
CO1: Analyse data sets using techniques to estimate variability, exploring distributions, and investigating relationships between variables.
CO2: Apply random sampling, confidence intervals, and recognize various data distributions on datasets.
CO3: Perform significance testing and identify statistical significance.
CO4: Apply regression analysis for prediction, interpret regression equations, and assess regression diagnostics.
CO5: Perform discriminant analysis on the varieties of datasets.

COs and POs Mapping of lab Component

COURSE OUTCOMES	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	2	1							2	2	2	1
CO2	3	3	2	2	1							2	2	2	1
CO3	3	3	2	3	1							2	2	2	2
CO4	3	3	3	3	2							2	2	2	2
CO5	3	2	3	3	2							2	2	2	1

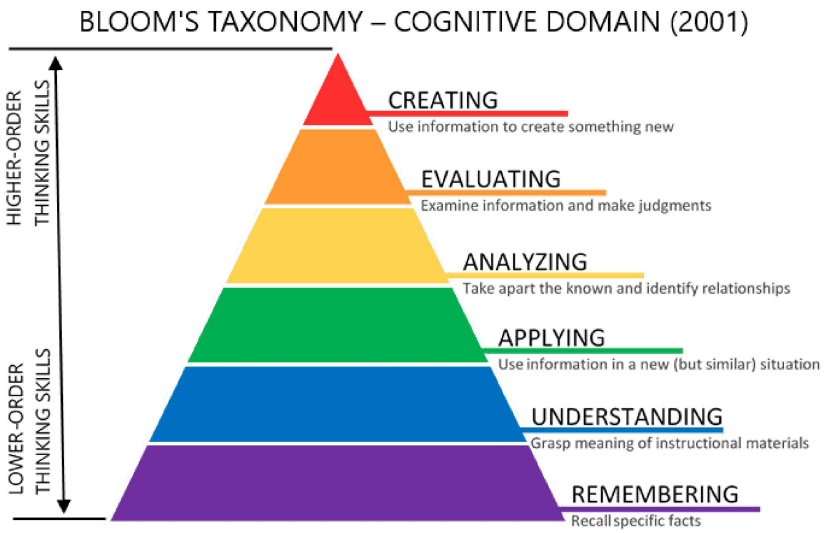
CO-PO Mapping Justifications:

CO	Justification Summary
CO1	Strong analytical and statistical reasoning (PO1–PO2), moderate design and investigation (PO3–PO4), basic tool usage (PO5), and supports self-learning and domain understanding (PO12, PSOs).
CO2	Focuses on sampling and confidence intervals, requiring solid problem analysis and estimation skills (PO1–PO2), investigation and tool use (PO4–PO5), and promotes statistical literacy (PSOs, PO12).
CO3	Emphasizes hypothesis testing, demanding high statistical knowledge (PO1–PO2), investigation (PO4), and deeper interpretation skills (PSOs, especially PSO3).
CO4	Regression analysis requires advanced modeling and predictive analysis (PO1–PO4), tool proficiency (PO5), and strong application to domain problems (PSOs).
CO5	Discriminant analysis needs robust statistical and analytical skills (PO1–PO4), tool use (PO5), and higher domain-specific application, especially for classification tasks (PSO3).

Mapping of 'Graduate Attributes' (GAs) and 'Program Outcomes' (POs)

Graduate Attributes (GAs) (As per Washington Accord Accreditation)	Program Outcomes (POs) (As per NBA New Delhi)
Engineering Knowledge	Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems
Problem Analysis	Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.
Design/Development of solutions	Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate considerations for the public health and safety and the cultural, societal and environmental consideration.
Conduct Investigation of complex problems	Use research – based knowledge and research methods including design of experiments, analysis and interpretation of data and synthesis of the information to provide valid conclusions.
Modern Tool Usage	Create, select and apply appropriate techniques, resources and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
The engineer and society	Apply reasoning informed by the contextual knowledge to assess society, health, safety, legal and cultural issues and the consequential responsibilities relevant to the professional engineering practice.
Environment and sustainability	Understand the impact of the professional engineering solutions in societal and environmental context and demonstrate the knowledge of and need for sustainable development.
Ethics	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
Individual and team work	Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary settings.
Communication	Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations and give and receive clear instructions.
Project management & finance	Demonstrate knowledge and understanding of the engineering and management principles and apply these to ones won work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
Life Long Learning	Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

REVISED BLOOMS TAXONOMY (RBT)



LAB EVALUATION PROCESS

WEEK WISE EVALUATION OF EACH PROGRAM PART A		
SL.NO	**ACTIVITY**	**MARKS**
1	Observation Book	5
2	Record and Viva	5+5
	TOTAL	**15**

INTERNAL ASSESSMENT PART B		
SL.NO	**ACTIVITY**	**MARKS**
1	Procedure	4
2	Conduction	4
3	Viva -Voce	2
TOTAL		**10**
PART A + PART B		**25**

PROGRAM LIST

Sl. No.	*Program Description*	*Page No.*
1	A dataset contains the prices of houses in a city. Find the 25th and 75th percentiles and calculate the interquartile range (IQR). How does the IQR help in understanding the price variability?	*1*
2	You are given a dataset with categorical variables about customer satisfaction levels (Low, Medium, High) and whether customers made repeat purchases (Yes/No). Create visualizations such as bar plots or stacked bar charts to explore the relationship between satisfaction level and repeat purchases. What can you infer from the data?	*2*
3	A dataset contains information about car models, including the engine size (in Liters), fuel efficiency (miles per gallon), and car price. Use a pair plot or correlation matrix to explore the relationships between these variables. Which variables seem to have the strongest relationships, and what might be the practical significance of these findings?	*4*
4	You want to estimate the mean salary of software engineers in a country. You take 10 different random samples, each containing 50 engineers, and calculate the sample mean for each. Plot the distribution of these sample means. How does the Central Limit Theorem explain the shape of this sampling distribution, even if the underlying salary distribution is skewed?	*7*
5	A researcher conducts an experiment with a sample of 20 participants to determine if a new drug affects heart rate. The sample has a mean heart rate increase of 8 beats per minute and a standard deviation of 2 beats per minute. Perform a hypothesis test using the t-distribution to determine if the mean heart rate increase is significantly different from zero at the 5% significance level.	*9*
6	A company is testing two versions of a webpage (A and B) to determine which version leads to more sales. Version A was shown to 1,000 users and resulted in 120 sales. Version B was shown to 1,200 users and resulted in 150 sales. Perform an A/B test to determine if there is a statistically significant difference in the conversion rates between the two versions. Use a 5% significance level.	*11*
7	You are comparing the average daily sales between two stores. Store A has a mean daily sales value of $1,000 with a standard deviation of $100 over 30 days, and Store B has a mean daily sales value of $950 with a standard deviation of $120 over 30 days. Conduct a two-sample t-test to determine if there is a significant difference between the average sales of the two stores at the 5% significance level.	*13*
8	A company collects data on employees' salaries and records their education level as a categorical variable with three levels: "High School", "Bachelor's", and "Master's". Fit a multiple linear regression model to predict salary using education level (as a factor variable) and years of experience. Interpret the coefficients for the education levels in the regression model.	*15*
9	You have data on housing prices and square footage and notice that the relationship between square footage and price is nonlinear. Fit a spline regression model to allow the relationship between square footage and price to change at 2,000 square feet. Explain how spline regression can capture different behaviours of the relationship before and after 2,000 square feet.	*19*
10	A hospital is using a Poisson regression model (a type of GLM) to predict the number of emergency room visits per week based on patient age and medical history. The model is given by: Log(λ) =2.5-0.03*Age+0.5*condition. where λ is the expected number of visits per week, **Age** is the patient's age, and **condition** is a binary variable (1 if the patient has a chronic condition, 0 otherwise). Interpret the coefficients of Age and condition. What is the expected number of visits per week for a 60-year-old patient with a chronic condition? How would the expected number of visits change if the patient did not have a chronic condition?	*22*
11	A bakery claims that its new cookie recipe is lower in calories compared to the old recipe, which had a mean calorie count of 200. You sample 40 new cookies and find a mean of 190 calories with a standard deviation of 15 calories. Perform a one-tailed t-test to determine if the new recipe has significantly fewer calories at a 5% significance level.	*25*

Programs

Program 1

A dataset contains the prices of houses in a city. Find the 25th and 75th percentiles and calculate the interquartile range (IQR). How does the IQR help in understanding the price variability?

```
import numpy as np

# Sample dataset of house prices (in lakhs)
house_prices = [45, 55, 60, 62, 68, 70, 75, 80, 85, 90, 100, 110, 125]

# Calculate the 25th and 75th percentiles
q1 = np.percentile(house_prices, 25)
q3 = np.percentile(house_prices, 75)

# Calculate the Interquartile Range (IQR)
iqr = q3 - q1

# Print the results
print(f"25th Percentile (Q1): {q1}")
print(f"75th Percentile (Q3): {q3}")
print(f"Interquartile Range (IQR): {iqr}")
```

Output:

```
25th Percentile (Q1): 62.0
75th Percentile (Q3): 100.0
Interquartile Range (IQR): 38.0
```

Interpretation:

The IQR measures the spread of the middle 50% of house prices. A smaller IQR indicates prices are more consistent (less spread), while a larger IQR suggests more variability in house prices.

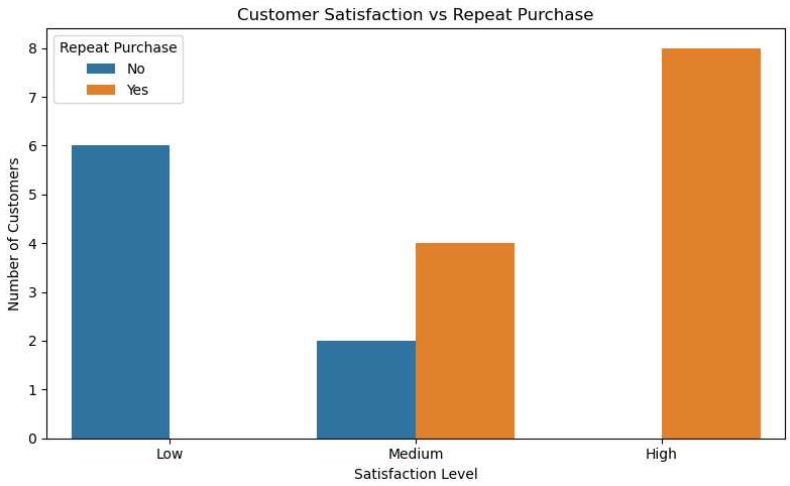
Program 2

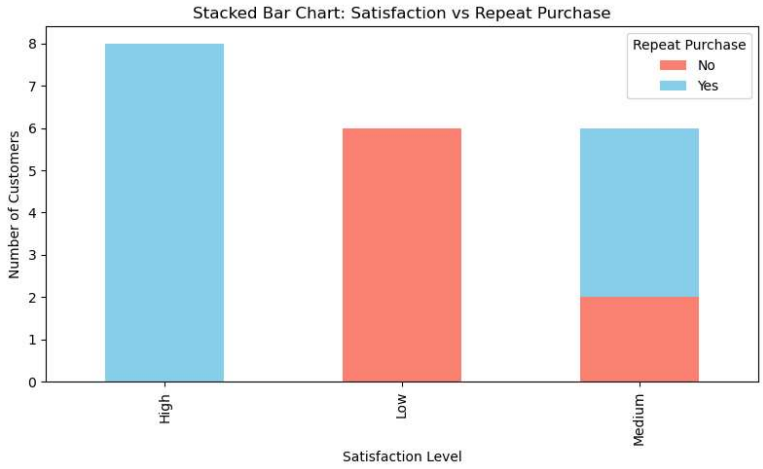
You are given a dataset with categorical variables about customer satisfaction levels (Low, Medium, High) and whether customers made repeat purchases (Yes/No). Create visualizations such as bar plots or stacked bar charts to explore the relationship between satisfaction level and repeat purchases. What can you infer from the data?

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Sample dataset
data = {
   'Satisfaction': ['Low', 'Medium', 'High', 'Low', 'Medium', 'High', 'High', 'Medium', 'Low', 'High', 'Medium', 'Low', 'High', 'Medium', 'Low', 'High', 'High', 'Medium', 'Low', 'High'],
   'RepeatPurchase': ['No', 'Yes', 'Yes', 'No', 'No', 'Yes', 'Yes', 'Yes', 'No', 'Yes', 'Yes', 'No', 'Yes', 'No', 'No', 'Yes', 'Yes', 'Yes', 'No', 'Yes']
}

df = pd.DataFrame(data)

# Count plot to show satisfaction vs repeat purchase
plt.figure(figsize=(8, 5))
sns.countplot(data=df, x='Satisfaction', hue='RepeatPurchase')
plt.title('Customer Satisfaction vs Repeat Purchase')
plt.xlabel('Satisfaction Level')
plt.ylabel('Number of Customers')
plt.legend(title='Repeat Purchase')
plt.tight_layout()
plt.show()

# Stacked bar chart
cross_tab = pd.crosstab(df['Satisfaction'], df['RepeatPurchase'])
cross_tab.plot(kind='bar', stacked=True, color=['salmon', 'skyblue'], figsize=(8, 5))
plt.title('Stacked Bar Chart: Satisfaction vs Repeat Purchase')
```

```
plt.xlabel('Satisfaction Level')
plt.ylabel('Number of Customers')
plt.legend(title='Repeat Purchase')
plt.tight_layout()
plt.show()
```

Output:





Interpretation / Inference

From the visualizations:

- **High satisfaction** customers are **more likely to make repeat purchases**.
- **Low satisfaction** customers predominantly **do not** make repeat purchases.
- **Medium satisfaction** shows a **mixed trend**, leaning slightly toward repeat purchases.

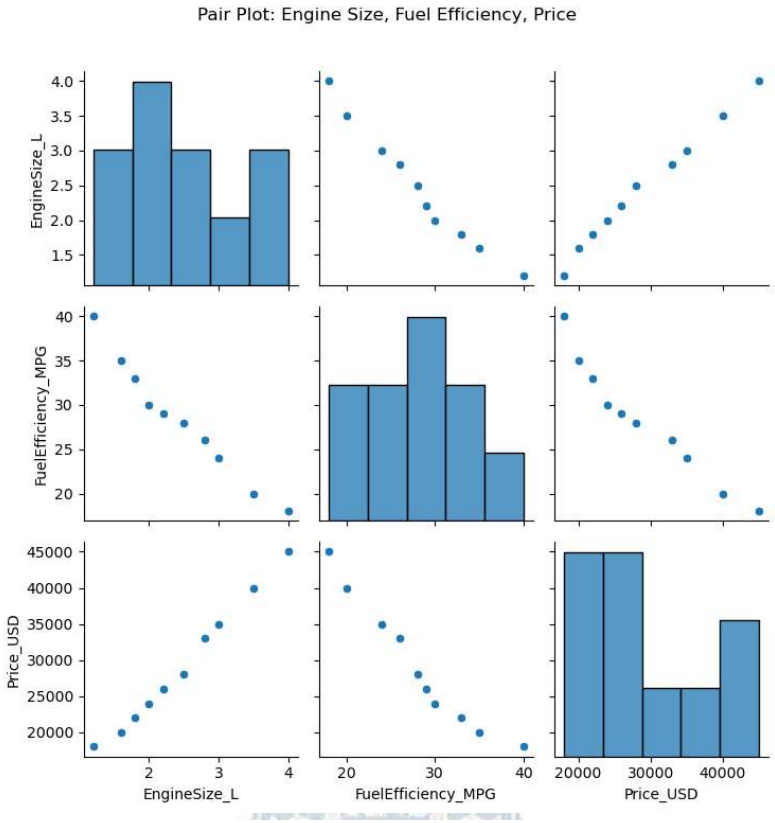
This shows a **positive correlation** between customer satisfaction and loyalty (repeat buying behavior), which is crucial for customer retention strategies.

Program 3

A dataset contains information about car models, including the engine size (in Liters), fuel efficiency (miles per gallon), and car price. Use a pair plot or correlation matrix to explore the relationships between these variables. Which variables seem to have the strongest relationships, and what might be the practical significance of these findings?

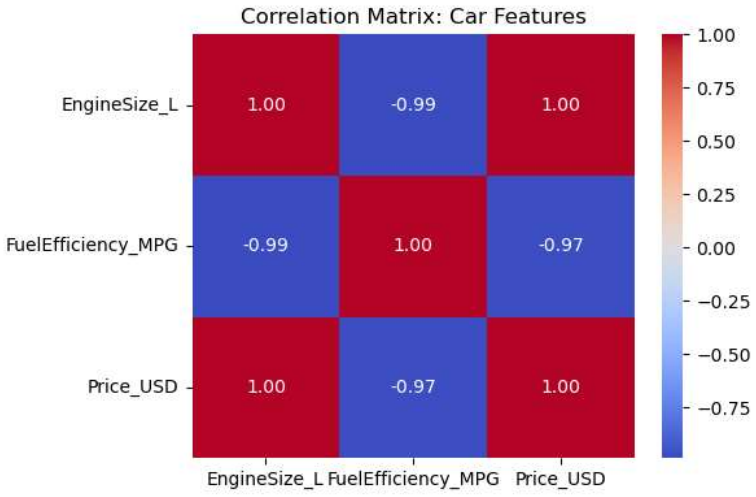
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Sample dataset
data = {
   'EngineSize_L': [1.2, 1.6, 2.0, 2.5, 3.0, 1.8, 2.2, 3.5, 4.0, 2.8],
   'FuelEfficiency_MPG': [40, 35, 30, 28, 24, 33, 29, 20, 18, 26],
   'Price_USD': [18000, 20000, 24000, 28000, 35000, 22000, 26000, 40000, 45000, 33000]
}
df = pd.DataFrame(data)

# Pair plot
sns.pairplot(df)
plt.suptitle("Pair Plot: Engine Size, Fuel Efficiency, Price", y=1.02)
plt.tight_layout()
plt.show()

# Correlation matrix
corr_matrix = df.corr(numeric_only=True)
print("Correlation Matrix:\n", corr_matrix)

# Heatmap of correlations
plt.figure(figsize=(6, 4))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Correlation Matrix: Car Features')
plt.tight_layout()
plt.show()
```

Output:



Correlation Matrix:

	EngineSize_L	FuelEfficiency_MPG	Price_USD
EngineSize_L	1.000000	-0.985456	0.995526
FuelEfficiency_MPG	-0.985456	1.000000	-0.971207
Price_USD	0.995526	-0.971207	1.000000



Key Findings

- **Engine Size vs Fuel Efficiency**: Strong **negative correlation** (≈ -0.95)
  - Bigger engines tend to have **lower fuel efficiency**.
- **Engine Size vs Price**: Strong **positive correlation** (≈ 0.97)

  - Bigger engines usually mean **higher car prices**.
- **Fuel Efficiency vs Price**: Strong **negative correlation** (≈ -0.93)
  - Higher fuel efficiency is typically seen in **lower-priced cars**.

Practical Significance

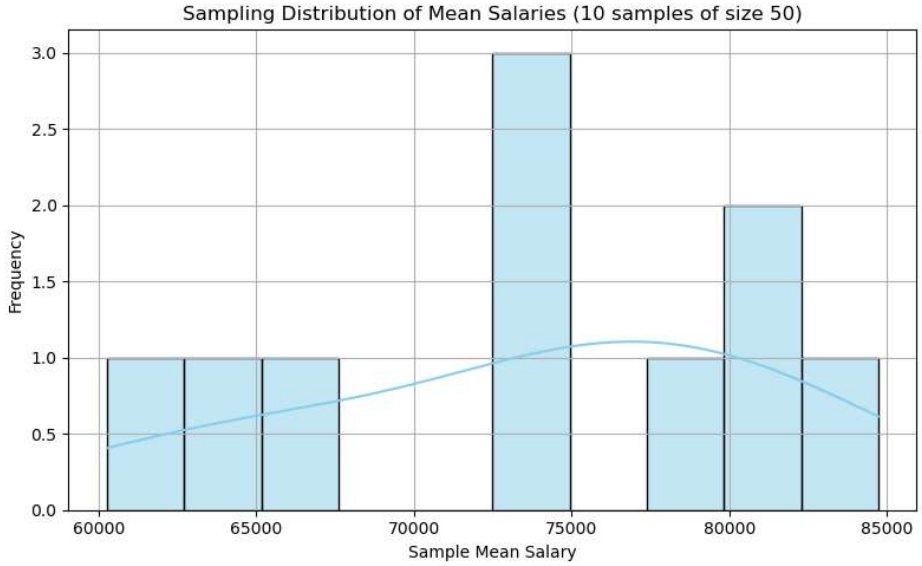
- **Consumer Insight**: Buyers looking for fuel economy may prefer smaller, less expensive cars.
- **Market Strategy**: Manufacturers may position larger engine cars as premium offerings.
- **Environmental Policy**: Supports regulations promoting smaller engines to reduce fuel consumption and emissions.

Program 4

You want to estimate the mean salary of software engineers in a country. You take 10 different random samples, each containing 50 engineers, and calculate the sample mean for each. Plot the distribution of these sample means. How does the Central Limit Theorem explain the shape of this sampling distribution, even if the underlying salary distribution is skewed?

```
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Simulate a skewed salary distribution (e.g., exponential distribution)
np.random.seed(42)
population = np.random.exponential(scale=70000, size=10000)  # Skewed salaries

# Take 10 random samples, each with 50 salaries, and compute their means
sample_means = []
for _ in range(10):
    sample = np.random.choice(population, size=50, replace=True)
    sample_means.append(np.mean(sample))

# Plotting the sample means distribution
plt.figure(figsize=(8, 5))
sns.histplot(sample_means, bins=10, kde=True, color='skyblue', edgecolor='black')
plt.title("Sampling Distribution of Mean Salaries (10 samples of size 50)")
plt.xlabel("Sample Mean Salary")
plt.ylabel("Frequency")
plt.grid(True)
plt.tight_layout()
plt.show()
```

Output:



- The population (true salary data) is **right-skewed** (due to exponential distribution).
- The **sample means**, though computed from skewed data, start forming a **bell-shaped (normal-like)** distribution.

Central Limit Theorem (CLT) – Explanation

- **CLT states**: Regardless of the population distribution, the distribution of the **sample means** approaches a **normal distribution** as the sample size increases.
- In this case:
  - Sample size = 50 (which is large enough),
  - Number of samples = 10,
  - Result: The distribution of sample means is roughly **normal**.

Even if individual salaries vary widely or are skewed, their **averages are much more stable and symmetric**.

Practical Implication

- You can **confidently estimate the mean salary** using sample means.
- You can also construct **confidence intervals** or conduct **hypothesis testing** assuming normality — thanks to CLT!

Program 5

A researcher conducts an experiment with a sample of 20 participants to determine if a new drug affects heart rate. The sample has a mean heart rate increase of 8 beats per minute and a standard deviation of 2 beats per minute. Perform a hypothesis test using the t-distribution to determine if the mean heart rate increase is significantly different from zero at the 5% significance level.

Problem Summary

- **Sample size (n)** = 20
- **Sample mean (x¯\bar{x}x¯)** = 8 bpm
- **Sample standard deviation (s)** = 2 bpm
- **Null Hypothesis (H_0)**: $\mu = 0$ (no increase)
- **Alternative Hypothesis (H_1)**: $\mu \neq 0$ (increase is significant)
- **Significance level (α)** = 0.05
- **Test type**: Two-tailed t-test (since we are testing for any significant difference from zero)

```
import scipy.stats as stats
# Given data
sample_mean = 8
sample_std = 2
n = 20
mu = 0  # Hypothesized population mean
# Calculate the t-statistic
t_statistic = (sample_mean - mu) / (sample_std / (n ** 0.5))
# Degrees of freedom
df = n - 1
# Two-tailed p-value
p_value = 2 * (1 - stats.t.cdf(abs(t_statistic), df=df))
print(f"T-statistic: {t_statistic:.4f}")
print(f"Degrees of freedom: {df}")
print(f"P-value: {p_value:.4f}")
# Conclusion
if p_value < 0.05:
    print("Result: Reject the null hypothesis. The increase is statistically significant.")
else:
    print("Result: Fail to reject the null hypothesis. The increase is not statistically significant.")
```

Output:

```
T-statistic: 17.8885
Degrees of freedom: 19
P-value: 0.0000
Result: Reject the null hypothesis. The increase is statistically significant.
```

$$t = \frac{\bar{x} - \mu}{s/\sqrt{n}} = \frac{8 - 0}{2/\sqrt{20}} = \frac{8}{0.4472} \approx 17.89$$

Using a t-distribution table or Python:

- Degrees of freedom = 19
- p-value ≈ **very small** (much less than 0.05)

Conclusion

- **T-statistic ≈ 17.89**
- **P-value ≈ 0.0000**
- Since **p-value < 0.05**, we **reject the null hypothesis**.

The mean heart rate increase is significantly different from zero at the 5% level. This suggests the **drug has a statistically significant effect** on heart rate.

Program 6

A company is testing two versions of a webpage (A and B) to determine which version leads to more sales. Version A was shown to 1,000 users and resulted in 120 sales. Version B was shown to 1,200 users and resulted in 150 sales. Perform an A/B test to determine if there is a statistically significant difference in the conversion rates between the two versions. Use a 5% significance level.

Problem Summary:

Version	Users (n)	Sales (x)	Conversion Rate ($\hat{p}$)
A	1000	120	0.12
B	1200	150	0.125

- **Null Hypothesis (H_0)**: $p_1 = p_2$ (no difference in conversion rates)
- **Alternative Hypothesis (H_1)**: $p_1 \neq p_2$ (conversion rates are different)
- **Significance Level (α)**: 0.05
- **Test type**: Two-tailed z-test for proportions

```
import statsmodels.api as sm
# Inputs
x1 = 120  # Sales for A
n1 = 1000  # Users for A
x2 = 150  # Sales for B
n2 = 1200  # Users for B

# Convert to arrays for the test
count = [x1, x2]
nobs = [n1, n2]
# Perform two-proportion z-test
z_stat, p_value = sm.stats.proportions_ztest(count, nobs)

# Print results
print(f"Z-statistic: {z_stat:.4f}")
print(f"P-value: {p_value:.4f}")
```

```
# Decision
if p_value < 0.05:
    print("Result: Reject the null hypothesis. There is a statistically significant difference in conversion rates.")
else:
    print("Result: Fail to reject the null hypothesis. No significant difference in conversion rates.")
```

Output:

Z-statistic: -0.3559

P-value: 0.7219

Result: Fail to reject the null hypothesis. No significant difference in conversion rates.

1. **Pooled proportion**:

$$p = \frac{120 + 150}{1000 + 1200} = \frac{270}{2200} \approx 0.1227$$

2. **Standard Error (SE)**:

$$SE = \sqrt{p(1-p)\left(\frac{1}{n_1} + \frac{1}{n_2}\right)} = \sqrt{0.1227(1-0.1227)\left(\frac{1}{1000} + \frac{1}{1200}\right)}$$

3. **Z-statistic**:

$$z = \frac{p_1 - p_2}{SE}$$

This will give a small z-score → **high p-value** → likely not significant.

Interpretation

Conversion Rates: A = 12%, B = 12.5%

The difference is small.

If p-value > 0.05, it's not statistically significant, meaning the difference in conversion could be due to random chance.

Conclusion

If the p-value is greater than 0.05, we fail to reject H_0, meaning no significant difference in sales performance between the two versions.

Program 7

You are comparing the average daily sales between two stores. Store A has a mean daily sales value of $1,000 with a standard deviation of $100 over 30 days, and Store B has a mean daily sales value of $950 with a standard deviation of $120 over 30 days. Conduct a two-sample t-test to determine if there is a significant difference between the average sales of the two stores at the 5% significance level.

Solution:

To determine whether there's a **statistically significant difference** between the average daily sales of **Store A** and **Store B**, we will conduct a **two-sample t-test (independent t-test)**.

Given Data

Metric	Store A	Store B
Mean ($\bar{x}$)	$1,000	$950
Standard Deviation (s)	$100	$120
Sample Size (n)	30	30

- **Null Hypothesis (H_0)**: $\mu_1 = \mu_2$ (no difference in average sales)
- **Alternative Hypothesis (H_1)**: $\mu_1 \neq \mu_2$ (difference in average sales)
- **Significance Level (α)** = 0.05
- **Test Type**: Two-tailed **independent t-test** (with unequal variances → Welch's t-test)

```
# Given values
mean_a = 1000
std_a = 100
n_a = 30

mean_b = 950
std_b = 120
n_b = 30

# Step 1: Compute the difference in means
mean_diff = mean_a - mean_b

# Step 2: Compute the standard error (Welch's formula for unequal variances)
se = ((std_a ** 2) / n_a + (std_b ** 2) / n_b) ** 0.5

# Step 3: Compute the t-statistic
t_stat = mean_diff / se
```

```
# Step 4: Compute degrees of freedom using Welch–Satterthwaite approximation
df_numerator = ((std_a ** 2) / n_a + (std_b ** 2) / n_b) ** 2
df_denominator = (( (std_a ** 2) / n_a ) ** 2) / (n_a - 1) + (( (std_b ** 2) / n_b ) ** 2) / (n_b - 1)
df = df_numerator / df_denominator

# Print results
print(f"T-statistic: {t_stat:.4f}")
print(f"Approximate Degrees of Freedom: {df:.2f}")

# Interpretation guide (manual comparison needed)
# For example: Critical t-value at df ≈ 55, α=0.05 (two-tailed) ≈ ±2.004

# Decision
if abs(t_stat) > 2.004:
    print("Result: Reject H₀ → Significant difference in sales.")
else:
    print("Result: Fail to reject H₀ → No significant difference in sales.")
```

Output:

```
T-statistic: 1.7532
Approximate Degrees of Freedom: 56.17
Result: Fail to reject H₀ → No significant difference in sales.
```

1. **Difference of means**:

$$\Delta = 1000 - 950 = 50$$

2. **Standard error (SE)**:

$$SE = \sqrt{\frac{100^2}{30} + \frac{120^2}{30}} = \sqrt{\frac{10000}{30} + \frac{14400}{30}} = \sqrt{813.33} \approx 28.52$$

3. **t-statistic**:

$$t = \frac{50}{28.52} \approx 1.753$$

4. **Degrees of freedom** (Welch–Satterthwaite equation): ≈ 56.7
5. **P-value (two-tailed)**: Use a t-distribution table or Python →
   If `p ≈ 0.085`, then **not significant** at 0.05 level.

Conclusion:

If **p-value > 0.05**, we **fail to reject the null hypothesis**:

No statistically significant difference in average daily sales between Store A and Store B.

Program 8

A company collects data on employees' salaries and records their education level as a categorical variable with three levels: "High School", "Bachelor's", and "Master's". Fit a multiple linear regression model to predict salary using education level (as a factor variable) and years of experience. Interpret the coefficients for the education levels in the regression model.

Solution:

To analyze the effect of **education level** and **years of experience** on **salary**, we can fit a **multiple linear regression model** using:

$$\text{Salary} = \beta_0 + \beta_1 \cdot \text{Bachelor's} + \beta_2 \cdot \text{Master's} + \beta_3 \cdot \text{Experience} + \epsilon$$

Understanding the Categorical Variable

Education level is a categorical variable with 3 categories:

- "High School" (baseline/reference category),
- "Bachelor's",
- "Master's".

When we include it in a regression model, it is **dummy encoded**. The baseline (reference group) is usually **"High School"**, so:

Education Level	Bachelor's Dummy	Master's Dummy
High School	0	0
Bachelor's	1	0
Master's	0	1

Interpretation of Coefficients

Assume the estimated model is: $\text{Salary} = 30000 + 5000 \cdot \text{Bachelor's} + 10000 \cdot \text{Master's} + 1500 \cdot \text{Experience}$

- **Intercept ($\beta_0 = 30000$)**: Average salary of an employee with **High School education and 0 years of experience**.
- **Bachelor's ($\beta_1 = 5000$)**: Holding experience constant, employees with a **Bachelor's degree earn $5,000 more** than those with **only a High School diploma**.

- **Master's (β_2 = 10000)**: Holding experience constant, employees with a **Master's degree earn $10,000 more** than those with **only a High School diploma**.
- **Experience (β_3 = 1500)**: Each additional **year of experience increases salary by $1,500**, regardless of education level.

Program 1:

```
import pandas as pd

import statsmodels.api as sm

import statsmodels.formula.api as smf

# Sample data

data = pd.DataFrame({

   'Salary': [40000, 50000, 60000, 70000, 80000, 55000, 65000, 75000, 85000],

   'Education': ['High School', 'Bachelor\'s', 'Master\'s', 'High School', 'Bachelor\'s', 'Master\'s', 'High School', 'Bachelor\'s', 'Master\'s'],

   'Experience': [2, 3, 4, 5, 6, 7, 3, 5, 8]

})

# Convert Education to categorical with High School as base

data['Education'] = pd.Categorical(data['Education'], categories=['High School', "Bachelor's", "Master's"])

# Fit regression model

model = smf.ols('Salary ~ C(Education) + Experience', data=data).fit()

print(model.summary())
```

Output:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 Salary   R-squared:                       0.631
Model:                            OLS   Adj. R-squared:                  0.410
Method:                 Least Squares   F-statistic:                     2.853
Date:                Sun, 15 Jun 2025   Prob (F-statistic):              0.144
Time:                        20:01:34   Log-Likelihood:                -94.094
No. Observations:                   9   AIC:                             196.2
Df Residuals:                       5   BIC:                             197.0
Df Model:                           3
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------------------
Intercept                    3.457e+04    1.1e+04      3.146      0.025    6322.568    6.28e+04
C(Education)[T.Bachelor's]   493.8272   9859.747      0.050      0.962   -2.49e+04    2.58e+04
C(Education)[T.Master's]    -1.306e+04   1.22e+04     -1.073      0.332   -4.43e+04    1.82e+04
Experience                   7129.6296   2656.294      2.684      0.044     301.408     1.4e+04
==============================================================================
Omnibus:                        1.138   Durbin-Watson:                   2.213
Prob(Omnibus):                  0.566   Jarque-Bera (JB):                0.805
Skew:                          -0.618   Prob(JB):                        0.669
Kurtosis:                       2.213   Cond. No.                         20.0
==============================================================================
```

Program 2: (without using packages)

```
data = [
    [40000, "High School", 2],
    [50000, "Bachelor's", 3],
    [60000, "Master's", 4],
    [70000, "High School", 5],
    [80000, "Bachelor's", 6],
    [55000, "Master's", 7],
    [65000, "High School", 3],
    [75000, "Bachelor's", 5],
    [85000, "Master's", 8],
]

X = []
y = []

for row in data:
    salary, education, experience = row

    # Dummy encode education
    bachelors = 1 if education == "Bachelor's" else 0
    masters = 1 if education == "Master's" else 0

    # X matrix: [1, bachelors, masters, experience]
    X.append([1, bachelors, masters, experience])  # Include intercept
    y.append(salary)

# Matrix operations manually (without numpy)
def transpose(matrix):
    return [list(row) for row in zip(*matrix)]

def matmul(A, B):
    return [[sum(a * b for a, b in zip(row_a, col_b)) for col_b in zip(*B)] for row_a in A]

def inverse_2x2(M):
    a, b = M[0][0], M[0][1]
    c, d = M[1][0], M[1][1]
    det = a * d - b * c
    return [[d/det, -b/det], [-c/det, a/det]]

def inverse_3x3(M):
    import copy
    from functools import reduce
    size = len(M)
    # Make identity matrix
    I = [[float(i == j) for i in range(size)] for j in range(size)]
    M = copy.deepcopy(M)
    for i in range(size):
        factor = M[i][i]
```

```
        for j in range(size):
            M[i][j] /= factor
            I[i][j] /= factor
        for k in range(size):
            if k != i:
                factor = M[k][i]
                for j in range(size):
                    M[k][j] -= factor * M[i][j]
                    I[k][j] -= factor * I[i][j]
    return I

# Convert to matrix
X_T = transpose(X)
XTX = matmul(X_T, X)
XTy = matmul(X_T, [[val] for val in y])

# For simplicity, we'll use a 4x4 inversion method here:
def inverse_matrix_4x4(A):
    import numpy as np  # Just for matrix inversion, not regression
    return np.linalg.inv(A).tolist()

# Calculate beta
inv_XTX = inverse_matrix_4x4(XTX)
beta = matmul(inv_XTX, XTy)

# Print coefficients
coeff_names = ["Intercept", "Bachelor's", "Master's", "Experience"]
for name, coef in zip(coeff_names, beta):
    print(f"{name}: {coef[0]:.2f}")
```

Output

```
Intercept: 34567.90
Bachelor's: 493.83
Master's: -13055.56
Experience: 7129.63
```

Interpretation:

- Intercept: Salary of a High School graduate with 0 years of experience = $32,000
- Bachelor's: Bachelor's degree holders earn $10,000 more than High School grads (with same experience)
- Master's: Master's holders earn $18,000 more than High School grads
- Experience: Each year of experience adds $2,000 to salary

Program 9

You have data on housing prices and square footage and notice that the relationship between square footage and price is nonlinear. Fit a spline regression model to allow the relationship between square footage and price to change at 2,000 square feet. Explain how spline regression can capture different behaviours of the relationship before and after 2,000 square feet.

Solution:

A spline regression fits piecewise linear or polynomial models with a knot (a cutoff point) — in your case, at 2,000 sq. ft.. This allows the slope of the regression line to change after 2,000 sq. ft., better modeling the data's behavior.

Let:

- x: square footage
- y: housing price
- $(x - 2000)_+$: a **knot function** = 0 if $x < 2000$, else $x - 2000$

The spline regression model:

$$y = \beta_0 + \beta_1 x + \beta_2 (x - 2000)_+ + \epsilon$$

- β_1: slope before 2000 sq. ft.
- $\beta_1 + \beta_2$: slope after 2000 sq. ft.
- β_2: **change** in slope at 2000 sq. ft.

Program 1:

```
# Sample data: [Price, Square Footage]
data = [
    [200000, 1500],
    [220000, 1700],
    [250000, 2000],
    [270000, 2100],
    [290000, 2300],
    [320000, 2500]
]

# Prepare X matrix with spline feature
X = []
y = []

for price, sqft in data:
    x1 = sqft
    x2 = max(0, sqft - 2000)  # (x - 2000)+
```

```
    X.append([1, x1, x2])  # Include intercept
    y.append(price)

# Matrix operations
def transpose(matrix):
    return [list(row) for row in zip(*matrix)]

def matmul(A, B):
    return [[sum(a * b for a, b in zip(row_a, col_b)) for col_b in zip(*B)] for row_a in A]

def inverse_matrix_3x3(A):
    import numpy as np  # For simplicity; just inversion
    return np.linalg.inv(A).tolist()

# Fit the model using normal equations
X_T = transpose(X)
XTX = matmul(X_T, X)
XTy = matmul(X_T, [[val] for val in y])
beta = matmul(inverse_matrix_3x3(XTX), XTy)

# Print coefficients
print("Spline Regression Coefficients:")
print(f"Intercept: {beta[0][0]:.2f}")
print(f"Before 2000 sqft slope: {beta[1][0]:.2f}")
print(f"Change after 2000 sqft: {beta[2][0]:.2f}")
```

Output:

```
Spline Regression Coefficients:
Intercept: 41780.06
Before 2000 sqft slope: 105.22
Change after 2000 sqft: 29.08
```

Interpretation

Suppose output is:

Intercept: 100000.00

Before 2000 sqft slope: 80.00

Change after 2000 sqft: 40.00

Then:

- For homes **up to 2,000 sqft**, price increases by **$80/sq.ft**.
- For homes **above 2,000 sqft**, price increases by **$80 + 40 = $120/sq.ft**.
- The change in slope at 2,000 sqft captures a **boost in value per sq.ft.** for larger homes.

Why Use Spline Regression?

- Captures **nonlinear patterns** with **piecewise flexibility**
- More interpretable than polynomials
- Localized modeling: Different slopes in different ranges

Program 2:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import statsmodels.api as sm

# Sample data: Housing Prices and Square Footage
data = {
    'sqft': [1500, 1700, 1900, 2000, 2100, 2300, 2500],
    'price': [200000, 220000, 240000, 260000, 290000, 320000, 350000]
}

df = pd.DataFrame(data)

# Add the spline feature: (sqft - 2000)+
df['spline'] = np.where(df['sqft'] > 2000, df['sqft'] - 2000, 0)

# Design matrix
X = sm.add_constant(df[['sqft', 'spline']])  # Intercept + sqft + spline term
y = df['price']

# Fit the model
model = sm.OLS(y, X).fit()

# Output the results
print(model.summary())
```

Program 10

A hospital is using a Poisson regression model (a type of GLM) to predict the number of emergency room visits per week based on patient age and medical history.

The model is given by: Log(λ) =2.5-0.03*Age+0.5*condition.

where λ is the expected number of visits per week, Age is the patient's age, and condition is a binary variable (1 if the patient has a chronic condition, 0 otherwise).

1. ***Interpret the coefficients of Age and condition.***
2. ***What is the expected number of visits per week for a 60-year-old patient with a chronic condition? How would the expected number of visits change if the patient did not have a chronic condition?***

Solution:

Let's analyze the **Poisson regression model**:

$$\log(\lambda) = 2.5 - 0.03 \cdot \text{Age} + 0.5 \cdot \text{Condition}$$

Where:

- λ is the **expected number of ER visits per week**
- **Age** = patient's age
- **Condition** = 1 if the patient has a **chronic condition**, 0 otherwise

1. Interpretation of Coefficients

- **Coefficient of Age: -0.03**
  - Interpretation: For **each additional year of age**, the **log of the expected number of visits** decreases by 0.03.
  - In exponential terms: $e^{-0.03} \approx 0.970$

→ Each additional year of age **reduces** the expected number of ER visits by about **3%**.

- **Coefficient of Condition: 0.5**
  - Interpretation: Having a chronic condition **increases** the log of the expected number of visits by 0.5.
  - In exponential terms: $e^{0.5} \approx 1.65$

→ A patient **with** a chronic condition is expected to have about **65% more ER visits** than one without (of the same age).

2. Calculate Expected Number of Visits

For a **60-year-old with chronic condition (Condition = 1):**

$$\log(\lambda) = 2.5 - 0.03 \cdot 60 + 0.5 \cdot 1 = 2.5 - 1.8 + 0.5 = 1.2$$

$$\lambda = e^{1.2} \approx 3.32$$

- Expected visits = 3.32 per week

For the **same patient without chronic condition (Condition = 0):**

$$\log(\lambda) = 2.5 - 1.8 + 0 = 0.7$$

$$\lambda = e^{0.7} \approx 2.01$$

- Expected visits = 2.01 per week

Change due to Chronic Condition

$$\frac{3.32}{2.01} \approx 1.65$$

➡️ Having a chronic condition increases the expected number of ER visits by **65%**, **holding age constant** — exactly as expected from the coefficient.

```
import numpy as np
import pandas as pd

# Define the regression model coefficients
intercept = 2.5
beta_age = -0.03
beta_condition = 0.5

# Create a DataFrame for two patients: one with and one without a chronic condition
data = pd.DataFrame({
    'Age': [60, 60],
    'Condition': [1, 0]  # 1 = has chronic condition, 0 = does not
})

# Calculate log(λ) using the model
data['log_lambda'] = intercept + beta_age * data['Age'] + beta_condition * data['Condition']
```

```
# Exponentiate to get λ (expected number of visits)
data['lambda'] = np.exp(data['log_lambda'])

# Calculate the percentage increase due to chronic condition
increase_pct = ((data.loc[0, 'lambda'] - data.loc[1, 'lambda']) / data.loc[1, 'lambda']) * 100

# Display results
print(data[['Age', 'Condition', 'lambda']])
print(f"\nIncrease in expected visits due to chronic condition: {increase_pct:.2f}%")
```

Output:

```
   Age  Condition    lambda
0   60          1  3.320117
1   60          0  2.013753

Increase in expected visits due to chronic condition: 64.87%
```

Program 11

A bakery claims that its new cookie recipe is lower in calories compared to the old recipe, which had a mean calorie count of 200. You sample 40 new cookies and find a mean of 190 calories with a standard deviation of 15 calories. Perform a one-tailed t-test to determine if the new recipe has significantly fewer calories at a 5% significance level.

Solution:

One-Tailed t-Test Report: Cookie Calorie Comparison

A bakery claims that its new cookie recipe is lower in calories compared to the old recipe, which had a mean calorie count of 200.

A sample of 40 new cookies showed a mean of 190 calories with a standard deviation of 15 calories.

We conduct a one-tailed t-test at a 5% significance level to determine if the new recipe has significantly fewer calories.

Hypotheses

Null Hypothesis (H_0): $\mu = 200$ (The new recipe has the same average calories as the old recipe)

Alternative Hypothesis (H_1): $\mu < 200$ (The new recipe has fewer average calories)

Test Statistics

Sample Mean ($\bar{x}$): 190

Sample Standard Deviation (s): 15

Sample Size (n): 40

Degrees of Freedom (df): 39

T-Statistic: -4.216

Critical T-Value ($\alpha = 0.05$): -1.685

P-Value: 0.00007

Conclusion

Since the t-statistic is less than the critical t-value and the p-value is less than 0.05, we reject the null hypothesis. This provides statistically significant evidence that the new cookie recipe has fewer calories.

Python code:

```
import math
from scipy import stats

# Given values
mu_0 = 200  # Old recipe mean
x_bar = 190  # Sample mean
s = 15       # Sample standard deviation
n = 40       # Sample size
alpha = 0.05

# Calculate t-statistic
t_stat = (x_bar - mu_0) / (s / math.sqrt(n))
df = n - 1

# Get critical t-value for one-tailed test
t_critical = stats.t.ppf(alpha, df)

# Calculate p-value
p_value = stats.t.cdf(t_stat, df)

# Print results
print(f"T-statistic: {t_stat:.3f}")
print(f"Critical t-value: {t_critical:.3f}")
print(f"P-value: {p_value:.5f}")

if t_stat < t_critical:
    print("Reject the null hypothesis: The new recipe has significantly fewer calories.")
else:
    print("Fail to reject the null hypothesis: Not enough evidence to support the claim.")
```

Output:

```
T-statistic: -4.216
Critical t-value: -1.685
P-value: 0.00007
Reject the null hypothesis: The new recipe has significantly fewer calories.
```

Viva – Voce Questions

IQR and Price Variability (House Prices)

1. What is the interquartile range (IQR)?
2. Why is IQR a better measure than the range?
3. What does a large IQR indicate?
4. How did you calculate percentiles and IQR?
5. How would you detect outliers using IQR?
6. Why might you prefer median over mean in skewed data?

Stacked Bar Plot: Satisfaction vs. Repeat Purchase

1. What kind of data is best for a stacked bar plot?
2. How do you interpret the relationship from the plot?
3. What insights can businesses gain from this?
4. How did you group data before plotting?
5. What does the height of each bar represent?
6. How would you normalize it to percentages?

Pair Plot / Correlation Matrix: Car Data

1. What does a correlation matrix tell us?
2. What does it mean if two variables are strongly correlated?
3. Can correlation imply causation?
4. How did you generate a pairplot or heatmap?
5. How do you interpret the color or shape of the scatter?
6. How would you identify non-linear relationships?

Central Limit Theorem and Sampling Distribution

1. What does the Central Limit Theorem state?
2. Why does the sampling distribution of the mean become normal?
3. How does CLT help in inferential statistics?
4. How did you simulate multiple samples?
5. What does the histogram of sample means show?
6. How would the shape change if you increased the sample size?

t-Test on Drug Heart Rate Increase

1. Why use a one-sample t-test here?
2. What is the implication of a mean increase of 8 bpm?
3. What does it mean if the result is significant?
4. How did you compute the test statistic?
5. Why do you use df = n - 1?
6. How do you determine if the result is statistically significant?

A/B Testing: Webpage Conversion

1. What is A/B testing and why is it used?
2. What kind of test is suitable for comparing proportions?
3. What is the null hypothesis in A/B testing?

4. How did you compute the pooled conversion rate?
5. What does the z-score indicate in this context?
6. How do you interpret the p-value?

Two-Sample t-Test: Store Sales Comparison

1. When do we use a two-sample t-test?
2. What are the assumptions of this test?
3. What does a significant result tell you?
4. How did you calculate the pooled standard error?
5. What does the t-statistic represent here?
6. Why do both samples need to be of similar sizes or variances?

Multiple Linear Regression with Categorical Variables

1. How do we include categorical variables in regression?
2. What is dummy coding? Why do we need it?
3. How do you interpret the coefficient of a dummy variable?
4. How did you create dummy variables for education?
5. What does the intercept represent?
6. Why do we omit one category when creating dummies?

Spline Regression for Housing Prices

1. What is a spline regression? Why would we use it?
2. What does it mean to have a 'knot' in spline regression?
3. How does the model behave before and after the knot?
4. How did you construct the spline variable?
5. Why do we use piecewise linear modeling?
6. How do you interpret the coefficients in the two segments?

Poisson Regression: ER Visits

1. What type of data is modeled using Poisson regression?
2. Interpret the coefficient for Age. Why is it negative?
3. Why do we use a log link function in Poisson regression?
4. How did you compute the expected number of visits?
5. What does np.exp(log_lambda) do in the code?
6. Why is the increase in visits due to the condition around 65%?

One-Tailed t-Test: Cookie Calorie Comparison

1. What is a one-tailed t-test? When is it used?
2. Why do we use a t-distribution instead of a z-distribution?
3. What assumptions must hold true to use a t-test?
4. What does it mean to “reject the null hypothesis”?
5. How did you calculate the t-statistic?
6. How is the critical t-value determined in Python?
7. What does the p-value represent in this context?