

**Experiment No: 4**

**Aim:** Implementation of Statistical Hypothesis Test using Scipy and Scikit-learn.

**Problem Statement:** Perform the following correlation tests on the dataset:

1. Pearson's Correlation Coefficient
2. Spearman's Rank Correlation
3. Kendall's Rank Correlation
4. Chi-Squared Test

**Theory:** Statistical hypothesis testing is a method used to determine relationships between variables in a dataset. The tests we perform are:

- **Pearson's Correlation Coefficient:** Measures the linear relationship between two continuous variables.
- Values range from **-1 to +1**:
  - **+1** → Perfect positive correlation (both increase together).
  - **-1** → Perfect negative correlation (one increases, the other decreases).
  - **0** → No correlation.

**Formula:**

$$r = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum (X_i - \bar{X})^2 \sum (Y_i - \bar{Y})^2}}$$

Where:

- $r$  = Pearson correlation coefficient
- $X_i, Y_i$  = Individual data points
- $\bar{X}, \bar{Y}$  = Means of  $X$  and  $Y$
- **Spearman's Rank Correlation:** Measures the monotonic relationship between variables using rank-based analysis.
- Values range from **-1 to +1**
- **Formula:**

$$r_s = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

Where:

- $r_s$  = Spearman's rank correlation coefficient
- $d_i$  = Difference between ranks of corresponding  $X$  and  $Y$  values
- $n$  = Number of data points
- **Kendall's Rank Correlation:** Measures the strength and direction of association between two variables.
- **Formula:**

$$\tau = \frac{(C - D)}{\frac{1}{2}n(n - 1)}$$

Where:

- $\tau$  = Kendall's correlation coefficient
- $C$  = Number of concordant pairs
- $D$  = Number of discordant pairs
- $n$  = Number of data points
- **Chi-Squared Test:** Used to test the independence between categorical variables.

**Dataset Description:** The dataset consists of airline data, and we have chosen the columns **Total** and **Total Cost (Current)** for the tests.

- **Formula:**

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

Where:

- $O_i$  = Observed count in each category
- $E_i$  = Expected count assuming independence

### Code Implementation:

**Step 1:** Load and Preprocess Data

```
[1] import pandas as pd
import numpy as np
import scipy.stats as stats

url = "processed1_fleet_data.csv"
df = pd.read_csv(url)

# Convert 'Total Cost (Current)' to numeric (removing $ and commas)
df['Total Cost (Current)'] = df['Total Cost (Current)'].replace(['$', ','], '', regex=True).astype(float)

# Selecting only relevant columns
df = df[['Total', 'Total Cost (Current)']].dropna()

print("Dataset Loaded and Cleaned Successfully.")
```

Dataset Loaded and Cleaned Successfully.

df.head()

	Total	Total Cost (Current)
0	4.0	90.0
1	8.0	0.0
2	41.0	3724.0
3	9.0	0.0
4	8.0	919.0

## Step 2: Pearson's Correlation Coefficient

```
[3] # Pearson's Correlation
pearson_corr, pearson_p = stats.pearsonr(df['Total'], df['Total Cost (Current)'])

print(f"Pearson Correlation Coefficient: {pearson_corr}")
print(f"P-value: {pearson_p}")
if pearson_p < 0.05:
    print("Reject the null hypothesis: Significant correlation exists.")
else:
    print("Fail to reject the null hypothesis: No significant correlation.")
```

Pearson Correlation Coefficient: 0.698548778087841  
P-value: 5.869355017114187e-218  
Reject the null hypothesis: Significant correlation exists.

## Step 3: Spearman's Rank Correlation

```
# Spearman's Rank Correlation
spearman_corr, spearman_p = stats.spearmanr(df['Total'], df['Total Cost (Current)'])

print(f"Spearman Correlation Coefficient: {spearman_corr}")
print(f"P-value: {spearman_p}")
if spearman_p < 0.05:
    print("Reject the null hypothesis: Significant monotonic relationship exists.")
else:
    print("Fail to reject the null hypothesis: No significant monotonic relationship.")
```

Spearman Correlation Coefficient: 0.5762818619372673  
P-value: 3.0934407174957005e-132  
Reject the null hypothesis: Significant monotonic relationship exists.

## Step 4: Kendall's Rank Correlation

```
[5] # Kendall's Rank Correlation
kendall_corr, kendall_p = stats.kendalltau(df['Total'], df['Total Cost (Current)'])

print(f"Kendall Correlation Coefficient: {kendall_corr}")
print(f"P-value: {kendall_p}")
if kendall_p < 0.05:
    print("Reject the null hypothesis: Significant association exists.")
else:
    print("Fail to reject the null hypothesis: No significant association.")
```

Kendall Correlation Coefficient: 0.444156533846385  
P-value: 7.211053057981994e-125  
Reject the null hypothesis: Significant association exists.

## Step 5: Chi-Squared Test

```
# Chi-Squared Test
chi2, chi_p = stats.chisquare(df['Total Cost (Current)'])

print(f"Chi-Squared Test Statistic: {chi2}")
print(f"P-value: {chi_p}")
if chi_p < 0.05:
    print("Reject the null hypothesis: Significant dependency exists.")
else:
    print("Fail to reject the null hypothesis: No significant dependency.")
```

Chi-Squared Test Statistic: 12004188.907641038  
P-value: 0.0  
Reject the null hypothesis: Significant dependency exists.

## Conclusion:

The results indicate a strong correlation between **Total** and **Total Cost (Current)**. Pearson's correlation coefficient of **0.6985** suggests a strong linear relationship, while Spearman's (**0.5763**) and Kendall's (**0.4442**) show a significant monotonic association. The Chi-Squared test also confirms a significant dependency. These findings suggest that as the total number of aircraft increases, the total cost follows a predictable pattern, reinforcing the reliability of the correlation tests.