Name:Bhushan Malpani Roll no:33 Div:D15C

Exp 4

Aim: Implementation of Statistical Hypothesis Test using Scipy and Sci-kit learn.

Theory and Output:

1.Loading dataset:

Data loading is the first step in data analysis. The dataset is stored in a CSV file and read using `pandas.read_csv()`.
The first few rows are displayed to understand the dataset structure

```
import pandas as pd
import scipy.stats as stats
```

```
df = pd.read_csv('/content/employee_data.csv')
```

```
df.head()
```

	Employee_ID	Age	Experience_Years	Monthly_Salary	Performance_Score	Hours_Worked_Week	Projects_Completed
0	1	50	25	104252	89	38	10
1	2	36	22	64749	92	48	2
2	3	29	8	129680	61	45	14
3	4	42	11	41907	93	37	2
4	5	40	0	43777	85	47	13

Name:Bhushan Malpani Roll no:33 Div:D15C

2.Pearson’s Correlation Coefficient:

Pearson’s Correlation Coefficient (denoted as **r**) measures the **linear** relationship between two continuous variables.
Values range from **-1 to +1**:

- **+1**: Perfect positive correlation
- **0**: No correlation
- **-1**: Perfect negative correlation

The formula for Pearson’s Correlation Coefficient is:

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

```
pearson_corr, pearson_p = stats.pearsonr(df['Age'], df['Monthly_Salary'])

print(f"Pearson’s Correlation Coefficient: {pearson_corr}")
print(f"P-value: {pearson_p}")
```

```
Pearson’s Correlation Coefficient: 0.04287327221666302
P-value: 0.4239519272951198
```

Name:Bhushan Malpani Roll no:33 Div:D15C

3.Spearman's Rank Correlation

- Spearman's Rank Correlation (denoted as ρ, rho) measures the monotonic relationship between two variables.
- It does not require normally distributed data.
- If ranks of two variables are related, it indicates correlation.
- The formula is:

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

```
spearman_corr, spearman_p = stats.spearmanr(df['Experience_Years'], df['Performance_Score'])

print(f"Spearman's Rank Correlation: {spearman_corr}")
print(f"P-value: {spearman_p}")
```

```
Spearman's Rank Correlation: 0.02681458037717826
P-value: 0.6171101462207367
```

Name:Bhushan Malpani Roll no:33 Div:D15C

4.Kendall's Rank Correlation

Theory:

- Kendall's Tau (τ) measures the **ordinal association** between two variables.
- It counts **concordant** and **discordant** pairs:
  - **Concordant pairs**: If one variable increases, the other also increases.
  - **Discordant pairs**: One increases while the other decreases.
- The formula is:

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

```
kendall_corr, kendall_p = stats.kendalltau(df['Hours_Worked_Week'], df['Projects_Completed'])

print(f"Kendall's Rank Correlation: {kendall_corr}")
print(f"P-value: {kendall_p}")
```

```
Kendall's Rank Correlation: -0.013818340859064245
P-value: 0.7135602814495787
```

Name:Bhushan Malpani Roll no:33 Div:D15C

5. Chi-Squared Test

- The **Chi-Squared Test** is used for **categorical data** to check if two variables are independent.
- It compares **observed** and **expected** frequencies.
- The formula is:

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

```python
df['Experience_Category'] = pd.cut(df['Experience_Years'], bins=[0, 5, 10, 20, 30], labels=['0-5', '6-10', '11-20', '21-30'])
df['Performance_Category'] = pd.cut(df['Performance_Score'], bins=[0, 50, 70, 90, 100], labels=['Low', 'Medium', 'High', 'Very High'])

contingency_table = pd.crosstab(df['Experience_Category'], df['Performance_Category'])

chi2_stat, p_val, dof, expected = stats.chi2_contingency(contingency_table)

print(f"Chi-Squared Statistic: {chi2_stat}")
print(f"P-value: {p_val}")
print(f"Degrees of Freedom: {dof}")
print("Expected Frequencies Table:")
print(expected)
```

```
Chi-Squared Statistic: 11.420158901810995
P-value: 0.24800442199136485
Degrees of Freedom: 9
Expected Frequencies Table:
[[ 0.96629213 15.78277154 19.16479401  7.08614232]
 [ 0.8988764  14.68164794 17.82771536  6.5917603 ]
 [ 2.04494382 33.40074906 40.55805243 14.99625468]
 [ 2.08988764 34.13483146 41.4494382  15.3258427 ]]
```

Name:Bhushan Malpani Roll no:33 Div:D15C

Conclusion

1. **Pearson's Correlation**: Measures **linear relationship** between numerical variables. If **p < 0.05**, the correlation is significant.
2. **Spearman's Correlation**: Checks for **monotonic relationship**. If **p < 0.05**, variables move together in a ranked order.
3. **Kendall's Correlation**: Identifies **ordinal association**. A small **p-value** means a strong relationship.
4. **Chi-Square Test**: Determines **independence of categorical variables**. If **p < 0.05**, variables are dependent; otherwise, they are independent.

Final Summary:

- If **p < 0.05**, the test indicates a significant relationship.
- If **p > 0.05**, no strong relationship exists.

These tests help understand **associations** in the dataset for data-driven decisions.