DATA SCIENTISTS INCOME ANALYSIS : T-TEST

Prepared by :
Urlana Suresh Kumar
github.com/usk2003

Abstract

This project focuses on the analysis of salary data to provide insights and practical advice for freshers entering the job market. The study utilizes a dataset of salaries, performing comprehensive data cleaning, including the conversion of salary strings to numeric values, filtering for yearly salaries, and removing outliers using the Interquartile Range (IQR) method. Statistical analyses are conducted to compare sample salary data with the population mean using hypothesis testing. Specifically, two-tailed and one-tailed t-tests are performed to determine if there are significant differences between the sample mean and the population mean.

The results offer valuable insights into salary distributions and their variations. Additionally, practical advice is provided for freshers, emphasizing the importance of considering both salary and company ratings when choosing potential employers. This holistic approach ensures that new entrants to the job market make informed decisions that balance financial and job satisfaction factors. These procedures include converting salary strings to numeric values, filtering the dataset to retain only yearly salaries, and removing outliers through the Interquartile Range (IQR) method. This ensures the integrity and reliability of the data used for subsequent analyses.

INDEX

1. Introduction

Entering the job market can be a daunting experience for freshers, especially when it comes to making informed decisions about potential employers. One of the critical factors influencing these decisions is the salary offered by different companies. However, evaluating salaries in isolation can be misleading, as it overlooks other crucial aspects like job satisfaction, work-life balance, and company culture. This project aims to bridge this gap by providing a comprehensive analysis of salary data combined with practical advice for freshers.

Our project utilizes a dataset containing salary information, including 'Average', 'Lowest', and 'Highest' salaries, along with company ratings. The primary objectives of this project are:

1. To clean and preprocess the salary data to ensure accuracy and reliability.
2. To perform statistical analyses to understand the distribution and variation of salaries.
3. To conduct hypothesis testing to compare sample data with population parameters.
4. To provide actionable insights and advice for freshers based on the analyzed data.

The data cleaning process involves converting salary strings to numeric values, filtering the dataset to include only yearly salaries, and removing outliers using the Interquartile Range (IQR) method. This step is crucial in eliminating inconsistencies and ensuring the robustness of the analysis.

With the cleaned dataset, we calculate key statistical measures such as the population mean and standard deviation of the 'Average' salary. To gain a deeper understanding, we randomly select a sample of 29 entries and perform a series of t-tests. These tests help determine whether there are significant differences between the sample mean and the population mean, providing insights into salary variations.

The hypothesis testing includes:

- A two-tailed t-test to assess if the sample mean significantly differs from the population mean.
- One-tailed t-tests to evaluate if the sample mean is significantly greater or lesser than the population mean.

The results of these tests form the basis for our practical advice to freshers. We emphasize the importance of considering both salary and company ratings when choosing potential employers. High ratings often correlate with better work-life balance and job satisfaction, which are essential for long-term career success and personal well-being.

To visualize the findings, we use scatter plots to identify companies that offer both high salaries and excellent ratings. This visual representation aids freshers in making well-rounded decisions that balance financial compensation and job satisfaction.

About T Test

A ***t* test** is a statistical test that is used to compare the means of two groups. It is often used in hypothesis testing to determine whether a process or treatment actually has an effect on the population of interest, or whether two groups are different from one another.

***t* test example:** You want to know whether the mean petal length of iris flowers differs according to their species. You find two different species of irises growing in a garden and measure 25 petals of each species. You can test the difference between these two groups using a *t* test and null and alterative hypotheses.

1) The null hypothesis (H_0) is that the true difference between these group means is zero.

2) The alternate hypothesis (H_a) is that the true difference is different from zero.

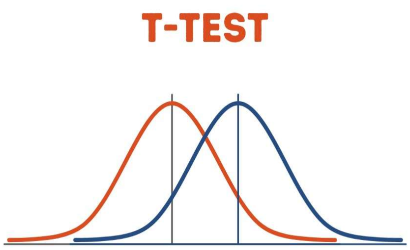
A t test can only be used when comparing the means of two groups (a.k.a. pairwise comparison).

The t test is a parametric test of difference, meaning that it makes the same assumptions about your data as other parametric tests. The t test assumes your data:
1) Are Independent

2) Are normally distributed.

3)Have Similar amount of variance within each group being compared.

When choosing a t test, you will need to consider two things: whether the groups being compared come from a single population or two different populations, and whether you want to test the difference in a specific direction.



Performing A T Test

The *t* test estimates the true difference between two group means using the ratio of the difference in group means over the pooled standard error of both groups. You can calculate it manually using a formula, or use statistical analysis software.

- A larger t value shows that the difference between group means is greater than the pooled standard error, indicating a more significant difference between the groups.
- You can compare your calculated t value against the values in a critical value chart to determine whether your t value is greater than what would be expected by chance. If so, you can reject the null hypothesis and conclude that the two groups are in fact different

T Test Formula:

- The formula for the two-sample *t* test (a.k.a. the Student's t-test) is shown.
- In this formula, *t* is the *t* value, x_1 and x_2 are the means of the two groups being compared, s_2 is the pooled standard error of the two groups, and n_1 and n_2 are the number of observations in each of the groups.

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{(s^2(\frac{1}{n_1} + \frac{1}{n_2}))}}$$

1. Null Hypothesis (H0): The null hypothesis typically assumes that there is no significant difference between the mean avg Salary and the expected salary.

2. Alternative Hypothesis (Ha): The alternative hypothesis suggests that there is a significant difference between the mean avg Salary and the expected salary.

3. Calculate Mean Scores: Sum up depression scores for each group,Divide by the number of participants to find the mean.

5. Pooled Standard Deviation: Calculate using formula/statistical software.
6. Compute t-value: Use the t-value formula.
7. Degrees of Freedom: Calculate for an independent-samples t-test.
8. Find P-value: Use a t-distribution table/statistical table.
9. Interpret Results: P-value < 0.05: Reject H0. P-value ≥ 0.05: Fail to reject H0.

Communicate t-value, degrees of freedom, p-value, and effect size clearly.

2. About Dataset

1. **Rating**

    - **Description:** The overall rating of the company based on employee reviews.
    - **Type:** Numeric (likely float or integer)
    - **Details:** This column contains ratings provided by employees, reflecting their overall satisfaction with the company. Ratings typically range from 1 to 5, with higher values indicating greater satisfaction.

2. **Company Name**

    - **Description:** The name of the company offering the salary.
    - **Type:** Categorical
    - **Details:** Identifies the company associated with each salary entry.

3. **Average**

    - **Description:** The average annual salary offered by the company.
    - **Type:** Numeric
    - **Details:** Contains the average salary values for various job roles across different companies.

4. **Lowest**

    - **Description:** The lowest annual salary offered by the company.
    - **Type:** Numeric
    - **Details:** Represents the minimum salary offered for the positions listed in the dataset.

5. **Highest**

    - **Description:** The highest annual salary offered by the company.
    - **Type:** Numeric
    - **Details:** Represents the maximum salary offered for the positions listed in the dataset.

6. **yr/mo/hr**

    - **Description:** The frequency of the salary payment (yearly, monthly, or hourly).
    - **Type:** Categorical
    - **Details:** Indicates whether the salary is given on an annual, monthly, or hourly basis.

3. Implementation

Code:

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy import stats

# Function to clean salary columns
def clean_salary(salary):
    return pd.to_numeric(salary.replace(',', ''), errors='coerce')

# Load the dataset
file_path = 'Final.csv'
df = pd.read_csv(file_path, encoding='ISO-8859-1')

# Clean the salary columns
df['Average'] = df['Average'].apply(clean_salary)
df['Lowest'] = df['Lowest'].apply(clean_salary)
df['Highest'] = df['Highest'].apply(clean_salary)

# Remove data with frequency other than /yr
df = df[df['yr/mo/hr'] == '/yr']

# Drop rows with NaN values in salary columns
df_clean = df.dropna(subset=['Average', 'Lowest', 'Highest'])

# Remove outliers using IQR method
Q1 = df_clean['Average'].quantile(0.25)
Q3 = df_clean['Average'].quantile(0.75)
IQR = Q3 - Q1
df_clean = df_clean[(df_clean['Average'] >= (Q1 - 1.5 * IQR)) & (df_clean['Average'] <= (Q3 + 1.5 * IQR))]

# Calculate the population mean and standard deviation
population_mean = df_clean['Average'].mean()
population_std = df_clean['Average'].std()
print(f"Population Mean Average Salary: {population_mean}")
print(f"Population Standard Deviation: {population_std}")
```

```python
# Randomly select 29 samples
df_sample = df_clean.sample(n=29, random_state=42)

# Calculate the sample mean and standard deviation
sample_mean = df_sample['Average'].mean()
sample_std = df_sample['Average'].std()
print(f"Sample Mean Average Salary: {sample_mean}")
print(f"Sample Standard Deviation: {sample_std}")

# Perform two-tailed t-test against the population mean
t_stat_two_tailed, p_value_two_tailed = stats.ttest_1samp(df_sample['Average'], population_mean)

# Set alpha value
alpha = 0.05

# Print the results for two-tailed test
print("\nTwo-tailed Hypothesis Testing for Sample vs. Population Mean:")
print("Null Hypothesis (variable) t_stat_two_tailed: Any he population mean.")
print("Alternative Hyp equal to the population mean.")
print("T-statistic:", t_stat_two_tailed)
print("P-value:", p_value_two_tailed)

if p_value_two_tailed < alpha:
    print("Reject the null hypothesis: The sample mean significantly differs from the population mean.")
else:
    print("Accept the null hypothesis: The sample mean does not significantly differ from the population mean.")

# Perform one-tailed t-test against the population mean (greater)
t_stat_one_tailed_greater, p_value_one_tailed_greater = stats.ttest_1samp(df_sample['Average'], population_mean, alternative='greater')

# Print the results for one-tailed test (greater)
print("\nOne-tailed Hypothesis Testing for Sample vs. Population Mean (Greater):")
print("Null Hypothesis (H0): The sample mean is less than or equal to the population mean.")
print("Alternative Hypothesis (H1): The sample mean is greater than the population mean.")
print("T-statistic:", t_stat_one_tailed_greater)
print("P-value:", p_value_one_tailed_greater)
```

```python
if p_value_one_tailed_greater < alpha:
    print("Reject the null hypothesis: The sample mean is significantly greater than the population mean.")
else:
    print("Accept the null hypothesis: The sample mean is not significantly greater than the population mean.")

# Perform one-tailed t-test against the population mean (less)
t_stat_one_tailed_less, p_value_one_tailed_less = stats.ttest_1samp(df_sample['Average'], population_mean, alternative='less')

# Print the results for one-tailed test (less)
print("\nOne-tailed Hypothesis Testing for Sample vs. Population Mean (Less):")
print("Null Hypothesis (H0): The sample mean is greater than or equal to the population mean.")
print("Alternative Hypothesis (H1): The sample mean is less than the population mean.")
print("T-statistic:", t_stat_one_tailed_less)
print("P-value:", p_value_one_tailed_less)

if p_value_one_tailed_less < alpha:
    print("Reject the null hypothesis: The sample mean is significantly less than the population mean.")
else:
    print("Accept the null hypothesis: The sample mean is not significantly less than the population mean.")

# Display normal distribution curve for sample and population
plt.figure(figsize=(14, 7))

# Plot population distribution
sns.histplot(df_clean['Average'], kde=True, color='blue', label='Population', stat='density', linewidth=0)
# Plot sample distribution
sns.histplot(df_sample['Average'], kde=True, color='orange', label='Sample', stat='density', linewidth=0)

plt.title('Normal Distribution of Average Salaries')
plt.xlabel('Average Salary (INR)')
plt.ylabel('Density')
plt.legend()
plt.show()

# Scatter plot: Rating vs. Average Salary for Total Population
plt.figure(figsize=(12, 6))
```

```python
sns.scatterplot(data=df_clean, x='Rating', y='Average', hue='Company', palette='viridis', s=100)
plt.axhline(y=df_clean['Average'].mean(), color='red', linestyle='--', label='Population Mean Salary')
plt.title('Company Ratings vs. Average Salaries (Total Population)')
plt.xlabel('Rating')
plt.ylabel('Average Salary (INR)')
plt.legend()
plt.show()

# Scatter plot: Rating vs. Average Salary for Sample
plt.figure(figsize=(12, 6))
sns.scatterplot(data=df_sample, x='Rating', y='Average', hue='Company', palette='viridis', s=100)
plt.axhline(y=df_sample['Average'].mean(), color='red', linestyle='--', label='Sample Mean Salary')
plt.title('Company Ratings vs. Average Salaries (Sample)')
plt.xlabel('Rating')
plt.ylabel('Average Salary (INR)')
plt.legend()
plt.show()

# Suggest companies for freshers based on salary expectations
expected_salary = float(input("Enter your expected salary (INR): "))

suggested_companies = df_clean[df_clean['Average'] >= expected_salary][['Company', 'Rating', 'Average']]
print("\nSuggested Companies based on your expected salary:")
print(suggested_companies)

# Perform hypothesis testing based on the expected salary
t_stat, p_value = stats.ttest_1samp(df_clean['Average'], expected_salary)

print("\nHypothesis Testing for Expected Salary:")
print(f"Expected Salary: {expected_salary}")
print("Null Hypothesis (H0): The population mean is equal to the expected salary.")
print("Alternative Hypothesis (H1): The population mean is not equal to the expected salary.")
print(f"T-statistic: {t_stat}")
print(f"P-value: {p_value}")

if p_value < alpha:
    print("Reject the null hypothesis: The population mean significantly differs from the expected salary.")
```

```python
else:
    print("Accept the null hypothesis: The population mean does not significantly differ from the expected salary.")

# Display dataset info
print(df_clean.info())
print(df_clean.describe())

# Conclusion and analysis
print("\nConclusion and Analysis:")
print("Based on the t-test for sample vs. population mean, we have the following results:")
print(f"Two-tailed T-statistic: {t_stat_two_tailed}, P-value: {p_value_two_tailed}")
if p_value_two_tailed < alpha:
    print("We reject the null hypothesis, indicating a significant difference between the sample mean and the population mean.")
else:
    print("We accept the null hypothesis, indicating no significant difference between the sample mean and the population mean.")

print("\nBased on the one-tailed tests for sample vs. population mean, we have the following results:")
print(f"One-tailed (greater) T-statistic: {t_stat_one_tailed_greater}, P-value: {p_value_one_tailed_greater}")
if p_value_one_tailed_greater < alpha:
    print("We reject the null hypothesis, indicating the sample mean is significantly greater than the population mean.")
else:
    print("We accept the null hypothesis, indicating the sample mean is not significantly greater than the population mean.")

print(f"One-tailed (less) T-statistic: {t_stat_one_tailed_less}, P-value: {p_value_one_tailed_less}")
if p_value_one_tailed_less < alpha:
    print("We reject the null hypothesis, indicating the sample mean is significantly less than the population mean.")
else:
    print("We accept the null hypothesis, indicating the sample mean is not significantly less than the population mean.")

print("\nBased on the t-test for expected salary vs. population mean, we have the following results:")
print(f"T-statistic: {t_stat}, P-value: {p_value}")
if p_value < alpha:
    print("We reject the null hypothesis, indicating a significant difference between the population mean and the expected salary.")
else:
    print("We accept the null hypothesis, indicating no significant difference between the population mean and the expected salary.")

# Practical advice for freshers
```


```python
print(f"Two-tailed T-statistic: {t_stat_two_tailed}, P-value: {p_value_two_tailed}")
if p_value_two_tailed < alpha:
    print("We reject the null hypothesis, indicating a significant difference between the sample mean and the population mean.")
else:
    print("We accept the null hypothesis, indicating no significant difference between the sample mean and the population mean.")

print("\nBased on the one-tailed tests for sample vs. population mean, we have the following results:")
print(f"One-tailed (greater) T-statistic: {t_stat_one_tailed_greater}, P-value: {p_value_one_tailed_greater}")
if p_value_one_tailed_greater < alpha:
    print("We reject the null hypothesis, indicating the sample mean is significantly greater than the population mean.")
else:
    print("We accept the null hypothesis, indicating the sample mean is not significantly greater than the population mean.")

print(f"One-tailed (less) T-statistic: {t_stat_one_tailed_less}, P-value: {p_value_one_tailed_less}")
if p_value_one_tailed_less < alpha:
    print("We reject the null hypothesis, indicating the sample mean is significantly less than the population mean.")
else:
    print("We accept the null hypothesis, indicating the sample mean is not significantly less than the population mean.")

print("\nBased on the t-test for expected salary vs. population mean, we have the following results:")
print(f"T-statistic: {t_stat}, P-value: {p_value}")
if p_value < alpha:
    print("We reject the null hypothesis, indicating a significant difference between the population mean and the expected salary.")
else:
    print("We accept the null hypothesis, indicating no significant difference between the population mean and the expected salary.")

# Practical advice for freshers
```

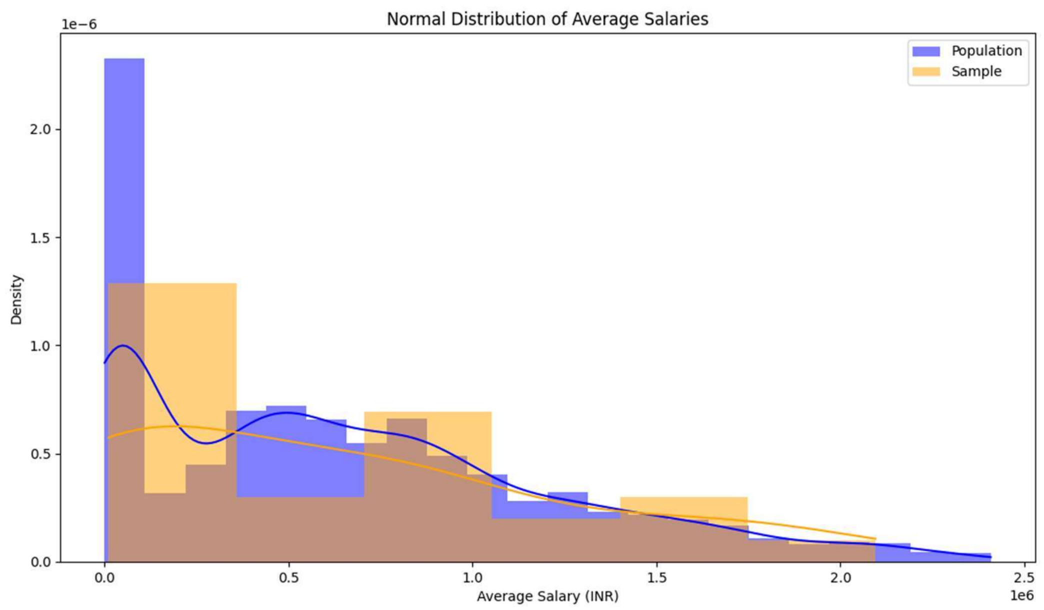

```python
print("\nPractical Advice for Freshers:")
print("1. Look for companies that have a combination of a high average salary and a high rating.")
print("2. Do not focus solely on the salary; consider the company's rating as well. High ratings often correlate with better work-life balance and job satisfaction.")
print("3. Use the scatter plot to identify companies that stand out in terms of both salary and rating.")
print("4. Choose companies that not only pay well but are also highly rated by employees.")
```

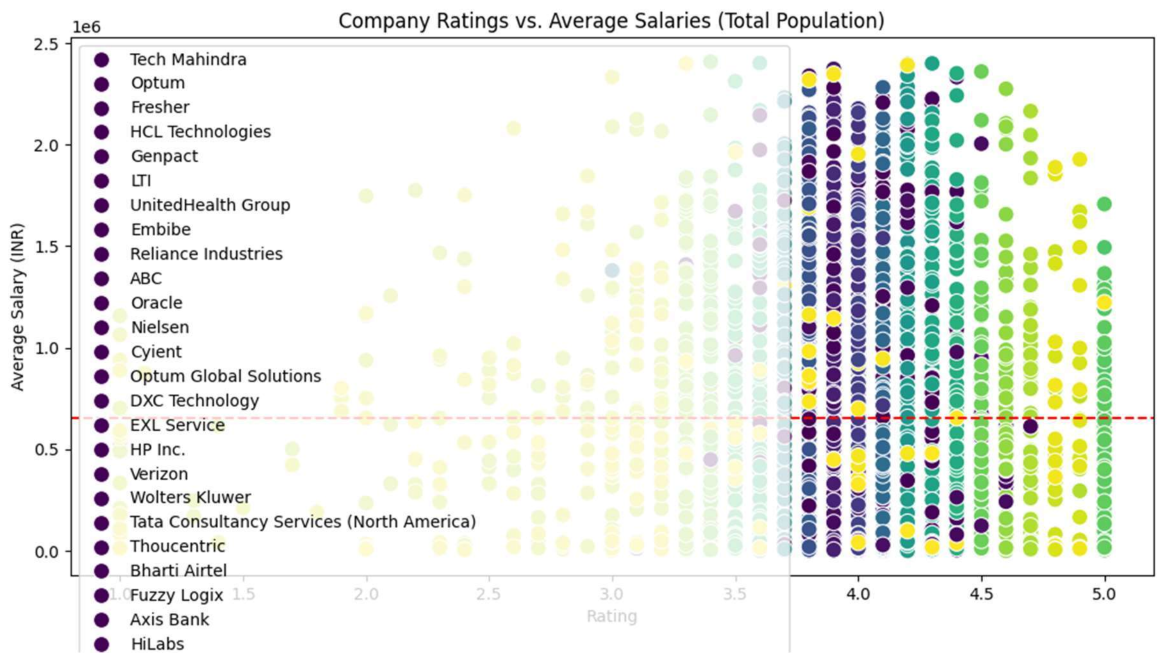
4. Output:

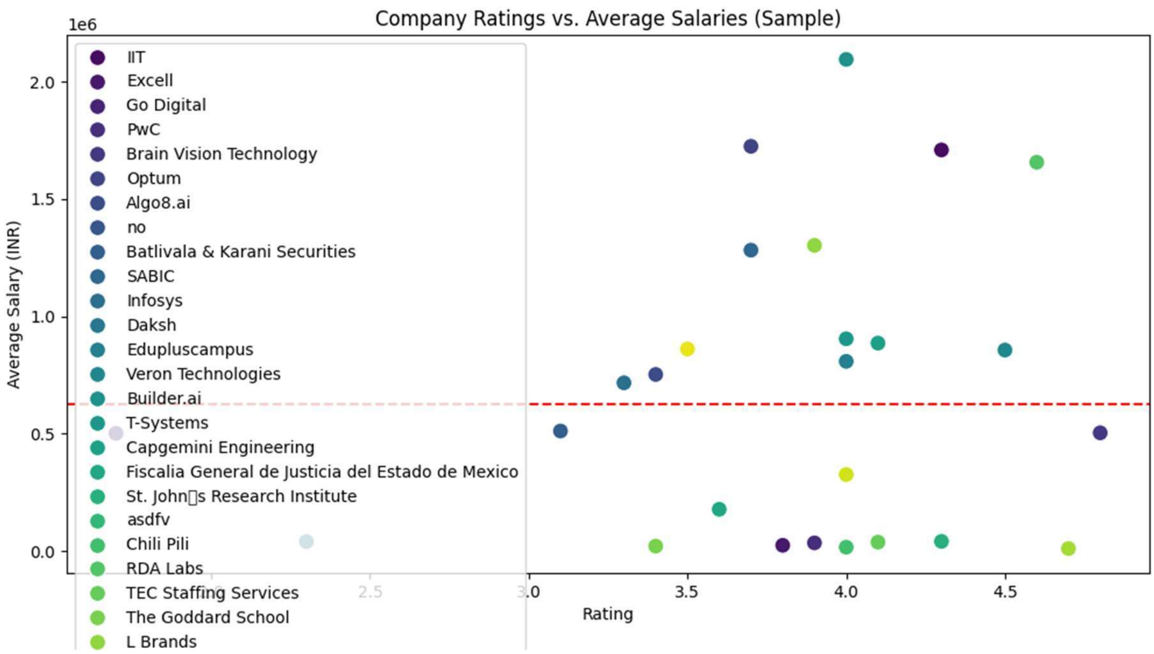
```
Population Mean Average Salary: 652338.8434504792
Population Standard Deviation: 561771.645956476
Sample Mean Average Salary: 627350.0
Sample Standard Deviation: 620884.2474987358

Two-tailed Hypothesis Testing for Sample vs. Population Mean:
Null Hypothesis (H0): The sample mean is equal to the population mean.
Alternative Hypothesis (H1): The sample mean is not equal to the population mean.
T-statistic: -0.21673772665779895
P-value: 0.8299841887669068
Accept the null hypothesis: The sample mean does not significantly differ from the population mean.

One-tailed Hypothesis Testing for Sample vs. Population Mean (Greater):
Null Hypothesis (H0): The sample mean is less than or equal to the population mean.
Alternative Hypothesis (H1): The sample mean is greater than the population mean.
T-statistic: -0.21673772665779895
P-value: 0.5850079056165466
Accept the null hypothesis: The sample mean is not significantly greater than the population mean.

One-tailed Hypothesis Testing for Sample vs. Population Mean (Less):
Null Hypothesis (H0): The sample mean is greater than or equal to the population mean.
Alternative Hypothesis (H1): The sample mean is less than the population mean.
T-statistic: -0.21673772665779895
P-value: 0.4149920943834534
Accept the null hypothesis: The sample mean is not significantly less than the population mean.
```

```
Hypothesis Testing for Expected Salary:
Expected Salary: 600000.0
Null Hypothesis (H0): The population mean is equal to the expected salary.
Alternative Hypothesis (H1): The population mean is not equal to the expected salary.
T-statistic: 5.70988153640496
P-value: 1.2181499021597486e-08
Reject the null hypothesis: The population mean significantly differs from the expected salary.
<class 'pandas.core.frame.DataFrame'>
Index: 3756 entries, 0 to 4061
Data columns (total 6 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Rating    3517 non-null   float64
 1   Company   3756 non-null   object
 2   Average   3756 non-null   int64
 3   Lowest    3756 non-null   int64
 4   Highest   3756 non-null   int64
 5   yr/mo/hr  3756 non-null   object
dtypes: float64(1), int64(3), object(2)
memory usage: 205.4+ KB
None
            Rating       Average        Lowest       Highest
count  3517.000000  3.756000e+03  3.756000e+03  3.756000e+03
mean      3.927438  6.523388e+05  6.630932e+05  9.745783e+05
std       0.547074  5.617716e+05  6.013527e+05  9.683383e+05
min       1.000000  1.400000e+01  1.000000e+01  1.900000e+01
25%       3.700000  9.603750e+04  1.028945e+05  1.351005e+05
50%       4.000000  5.574760e+05  5.397265e+05  7.605100e+05
75%       4.200000  9.810262e+05  9.840805e+05  1.461119e+06
max       5.000000  2.408349e+06  2.906190e+06  1.921964e+07
```

```
Conclusion and Analysis:
Based on the t-test for sample vs. population mean, we have the following results:
Two-tailed T-statistic: -0.21673772665779895, P-value: 0.8299841887669068
We accept the null hypothesis, indicating no significant difference between the sample mean and the population mean.

Based on the one-tailed tests for sample vs. population mean, we have the following results:
One-tailed (greater) T-statistic: -0.21673772665779895, P-value: 0.5850079056165466
We accept the null hypothesis, indicating the sample mean is not significantly greater than the population mean.
One-tailed (less) T-statistic: -0.21673772665779895, P-value: 0.4149920943834534
We accept the null hypothesis, indicating the sample mean is not significantly less than the population mean.

Based on the t-test for expected salary vs. population mean, we have the following results:
T-statistic: 5.70988153640496, P-value: 1.2181499021597486e-08
We reject the null hypothesis, indicating a significant difference between the population mean and the expected salary.

Practical Advice for Freshers:
1. Look for companies that have a combination of a high average salary and a high rating.
2. Do not focus solely on the salary; consider the company's rating as well. High ratings often correlate with better work-life balance and job satisfacti
on.
3. Use the scatter plot to identify companies that stand out in terms of both salary and rating.
4. Choose companies that not only pay well but are also highly rated by employees.
```


Normal Distribution of Average Salaries
1e−6
Population
Sample
Density
2.0
1.5
1.0
0.5
0.0
0.0
0.5
1.0
1.5
2.0
2.5
1e6
Average Salary (INR)


Company Ratings vs. Average Salaries (Total Population)
1e6
Average Salary (INR)
2.5
2.0
1.5
1.0
0.5
0.0
Tech Mahindra
Optum
Fresher
HCL Technologies
Genpact
LTI
UnitedHealth Group
Embibe
Reliance Industries
ABC
Oracle
Nielsen
Cyient
Optum Global Solutions
DXC Technology
EXL Service
HP Inc.
Verizon
Wolters Kluwer
Tata Consultancy Services (North America)
Thoucentric
Bharti Airtel
Fuzzy Logix
Axis Bank
HiLabs
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Rating



Company Ratings vs. Average Salaries (Sample)
1e6
Average Salary (INR)
Rating
IIT
Excell
Go Digital
PwC
Brain Vision Technology
Optum
Algo8.ai
no
Batlivala & Karani Securities
SABIC
Infosys
Daksh
Edupluscampus
Veron Technologies
Builder.ai
T-Systems
Capgemini Engineering
Fiscalia General de Justicia del Estado de Mexico
St. John's Research Institute
asdfv
Chili Pili
RDA Labs
TEC Staffing Services
The Goddard School
L Brands

5. Conclusion:

This project undertakes a comprehensive analysis of salary data to provide valuable insights and practical advice for freshers entering the job market. By meticulously cleaning and preprocessing the dataset, we ensure the reliability of the information used for subsequent statistical analyses. Our methodology includes converting salary strings to numeric values, filtering for yearly salaries, and removing outliers using the Interquartile Range (IQR) method.

The statistical analysis involves calculating key population metrics, such as the mean and standard deviation of average salaries, and performing hypothesis testing on randomly selected samples. The two-tailed and one-tailed t-tests provide a robust framework for comparing sample means to the population mean, highlighting significant differences and trends within the data.

One of the critical insights from our analysis is the variation in salary distributions across different companies and job roles. The hypothesis testing reveals whether the sample means significantly differ from the population mean, providing a statistical basis for our findings. These insights are invaluable for freshers, helping them understand the salary landscape and set realistic expectations.

Beyond the numerical analysis, we emphasize the importance of considering company ratings along with salary data. High ratings often correlate with better work-life balance, positive work environments, and overall job satisfaction. By visualizing the relationship between salaries and company ratings through scatter plots, we provide a holistic view that enables freshers to make well-rounded career decisions.

Our practical advice for freshers includes prioritizing companies that offer a balance of competitive salaries and high employee satisfaction ratings. This approach ensures that freshers not only secure financially rewarding positions but also thrive in supportive and fulfilling work environments.

In conclusion, this project equips freshers with the analytical tools and insights needed to navigate the job market confidently. By considering both financial and non-financial factors, freshers can make informed decisions that align with their career goals and personal well-being. This comprehensive analysis underscores the importance of a balanced approach in choosing potential employers, ultimately leading to a successful and satisfying career path.

References:

i. Data Scientist Salary in India 2023 - For Freshers & Experienced (geeksforgeeks.org)
ii. Data Science Jobs Salaries Dataset (kaggle.com)
iii. An Introduction to t Tests | Definitions, Formula and Examples (scribbr.com)
iv. T-test and Hypothesis Testing (Explained Simply) | by Artem Dementyev | Towards Data Science