Tech Salaries Mini Project: Python Implementation

1. Project Overview

1.1 Project Title

- Tech Salaries Analysis: US and International Salary & Experience Landscape

1.2 Objective

- Analyze and compare tech salaries across the US and international markets.
- Explore the relationship between salaries, experience, education, and skills.
- Provide actionable insights for job seekers, employers, and policymakers.

1.3 Key Questions

- How do tech salaries in the US compared to international markets?
- What is the impact of experience, education, and skills on salaries?
- Which countries or cities offer the highest salaries for tech roles?
- How do remote work trends affect salary structures?

2. Python Tools and Libraries

2.1 Core Libraries

- **Programming Languages**: Python,
- **Data Manipulation**: `Pandas`, `NumPy`
- **Visualization**: `Matplotlib`, `Seaborn`, `Plotly`
- **Geospatial Analysis**: `Folium`

3. Data Collection

```
#Load the dataset
df = pd.read_csv("salaries_clean.csv")
print(df.head())
```

4. Data Cleaning

4.1 Handling Missing Values

```
missing_values = df.isnull().sum()
print("Missing values per column:\n", missing_values)
```

```
#Imputation for critical columns
df['annual_base_pay'].fillna(df['annual_base_pay'].median(), inplace=True)

# Imputation for `total_experience_years` and `employer_experience_years` with
median
df['total_experience_years'].fillna(df['total_experience_years'].median(),
inplace=True)
df['employer_experience_years'].fillna(df['employer_experience_years'].median(),
inplace=True)

# Imputation for `employer_name`
df['employer_name'].fillna('A stranger', inplace=True)
```

```
#Imputation for `employer_name`
df['employer_name'].fillna('A stranger', inplace=True)

# Imputation for categorical columns with fashion
df['location_state'].fillna(df['location_state'].mode()[0], inplace=True)
df['location_country'].fillna(df['location_country'].mode()[0], inplace=True)
df['location_latitude'].fillna(df['location_latitude'].mode()[0], inplace=True)
df['location_longitude'].fillna(df['location_longitude'].mode()[0], inplace=True)
df['job_title_rank'].fillna(df['job_title_rank'].mode()[0], inplace=True)
```

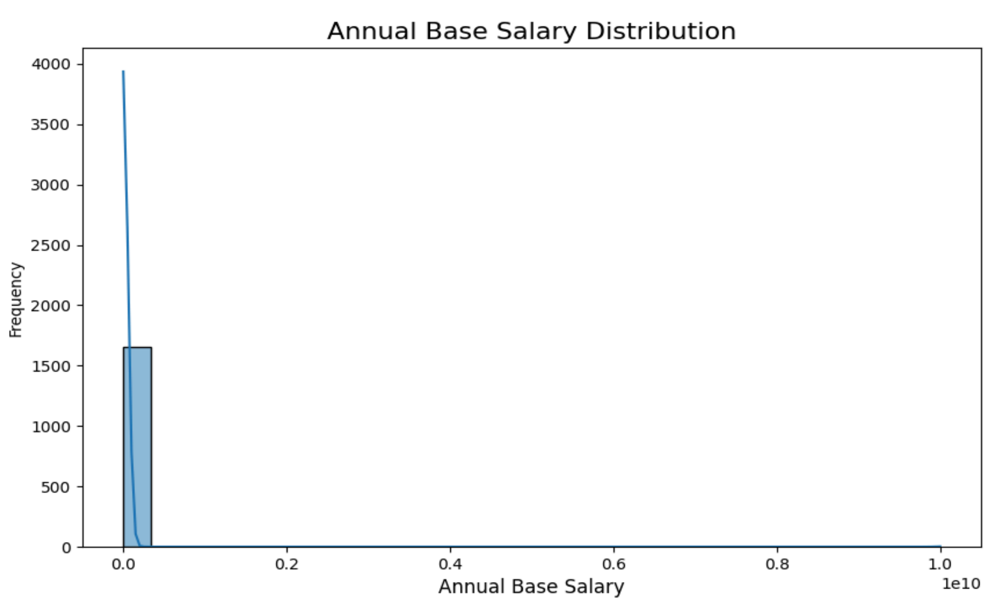
5. Exploratory Data Analysis (EDA)

5.1 Exploratory Data Analysis (EDA)

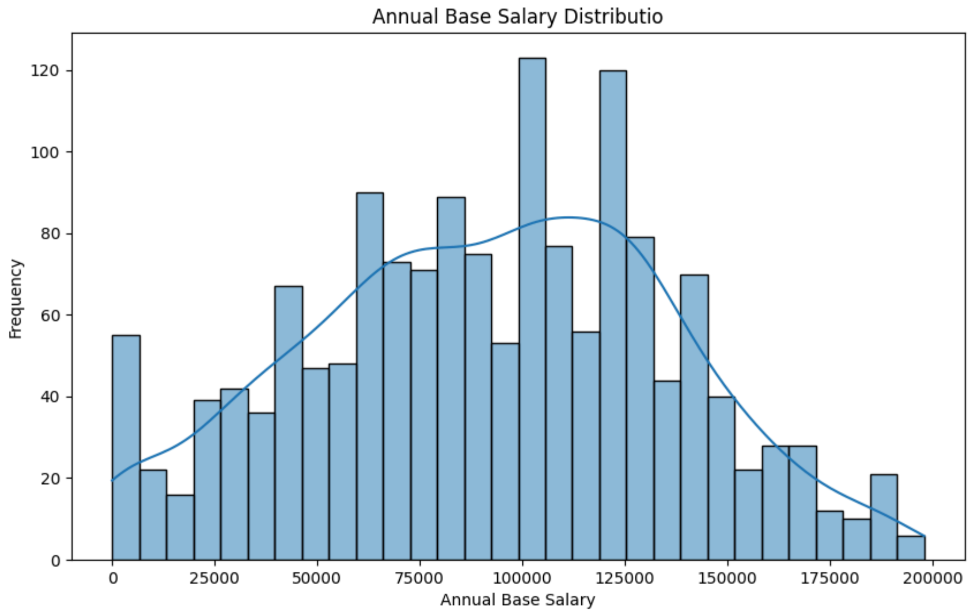
- Summary statistics (mean, median, standard deviation).
- Distribution of salaries by role, location, and experience.
- Heatmaps and correlation matrices.

5.1 Visualization

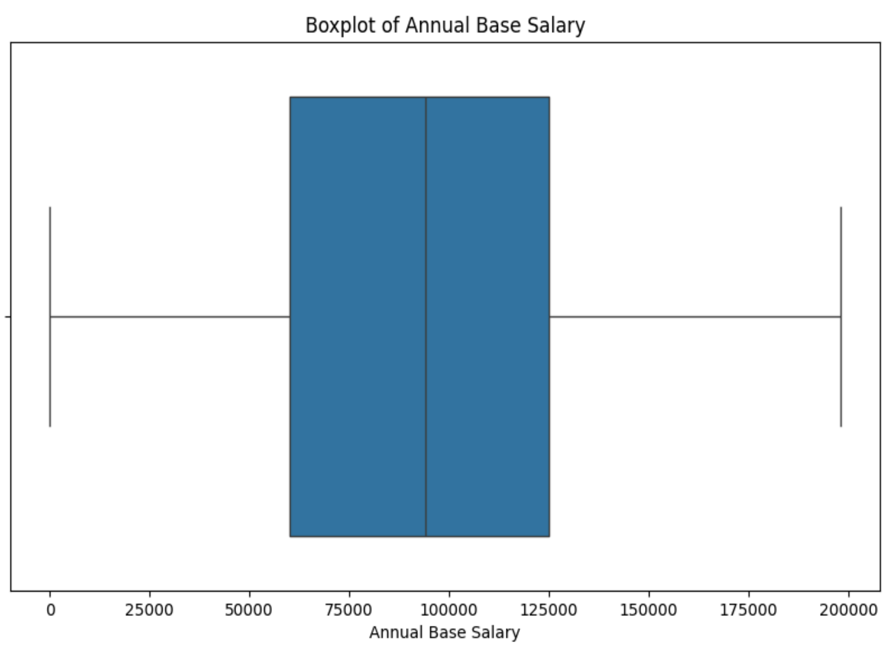
- Annual Base Salary Distribution



- Annual Base Salary Distribution

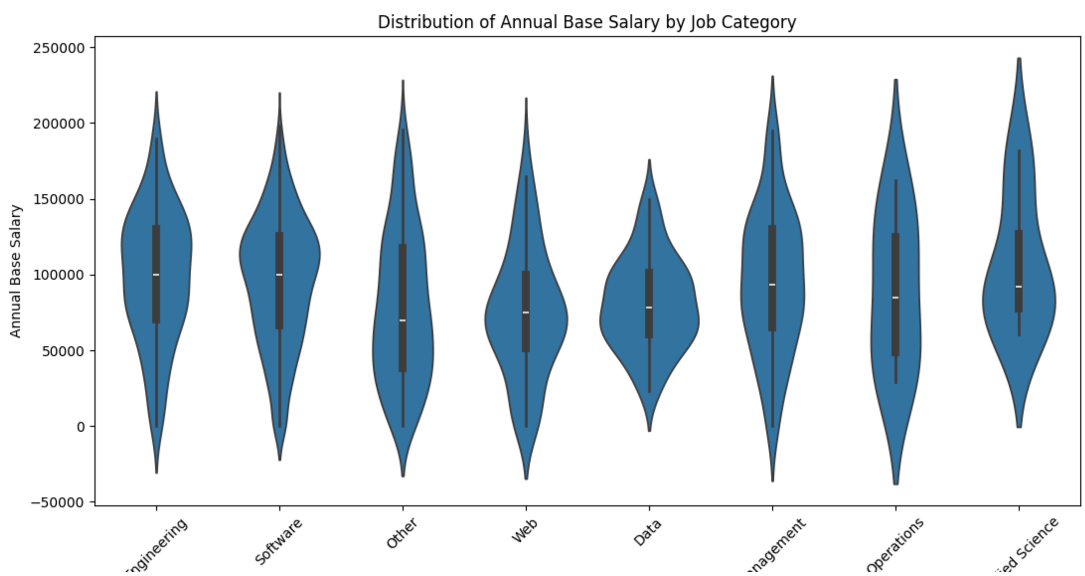


- Boxplot of Annual Base Salary



- Annual Base Salary vs Year of Experience

- Distribution of Annual Base by job category



5.2 Summary Statistics

```
print(df['salary'].describe())
```

5.3 Uni Variate Visualization #1: Salary Histogram

```
plt.figure(figsize=(10, 6))
sns.histplot(df['annual_base_pay'], bins=30, kde=True)
plt.title('Annual Base Salary Distribution',fontsize=16)
plt.xlabel('Annual Base Salary',fontsize=12)
plt.ylabel('Frequency')
plt.show()
```

```
Uni variate Visualization #1: Salary Histogram
plt.figure(figsize=(10, 6))
sns.histplot(df2['annual_base_pay'], bins=30, kde=True)
plt.title('Annual Base Salary Distributio')
plt.xlabel('Annual Base Salary')
plt.ylabel('Frequency')
plt.show()
```

6. Visualization

6.1 Interactive Salary Map

```
data = {
   'location_country': ['US', 'CA', 'OM', 'SE', 'JE'],
   'location_latitude': [37.77,36.36,41.47,38,43.1],
   'location_longitude': [-122.41,-94.2,-81.67,-97,-89.5],
   'annual_base_pay': [120000, 80000, 25000, 90000, 85000]  # Average salaries in
USD
}

# Create a DataFrame
df = pd.DataFrame(data)

# Create a base map centered at a global view
m = folium.Map(location=[20, 0], zoom_start=2)

# Add markers for each country
for idx, row in df.iterrows():
   folium.Marker(
       location=[row['location_latitude'], row['location_longitude']],  #
Latitude and Longitude
       popup=f"{row['location_country']}: ${row['annual_base_pay']:,.2f}",  #
Popup text
       tooltip=row['location_country']  # Tooltip on hover
   ).add_to(m)

# Save the map to an HTML file
m.save('global_salary_map.html')
```

7. Deliverables

1. **Cleaned Dataset**: `cleaned_tech_salaries.csv`
2. **Visualizations**: Interactive dashboards and charts.
3. **Final Report**: Insights and recommendations.

8. Deliverables

1. **Cleaned Dataset**: `salaries_clean.csv`
2. **Visualizations**: Interactive dashboards and charts.
3. **Final Report**: Insights and recommendations

9. Project Timeline

Task	Timeline
Data Collection	Day’s1–2
Data Cleaning	Day’s 2
EDA and Visualization	Day’s 3
Final Report	Day’s 2