Name: Kaushik Kotian Roll no.:30 Div :D15B Batch:B

Aim:
a. To perform Data Visualization for the selected data set using Matplotlib and Seaborn
b. To perform Exploratory Data Analysis for the selected data set

Theory:
The process of data visualization and exploratory data analysis (EDA) is a critical step in understanding the underlying patterns, trends, and anomalies in data. This process not only aids in making informed decisions but also in communicating findings effectively. Below, we delve into the theory behind each of the tasks mentioned in the problem statement, using Matplotlib and Seaborn for visualization in Python.

- Bar Graphs compare different categories of data, useful for visualizing and comparing categorical data distributions.

- Contingency Tables display the frequency distribution of variables, helping identify relationships between categorical variables.

- Scatter Plots visualize relationships between two continuous variables, useful for spotting patterns and outliers.

- Box Plots graphically represent data distribution, highlighting the median, quartiles, and outliers, useful for comparing distributions across groups.

- Heatmaps visualize complex data matrices, such as correlations between variables, using color intensity to represent data values.

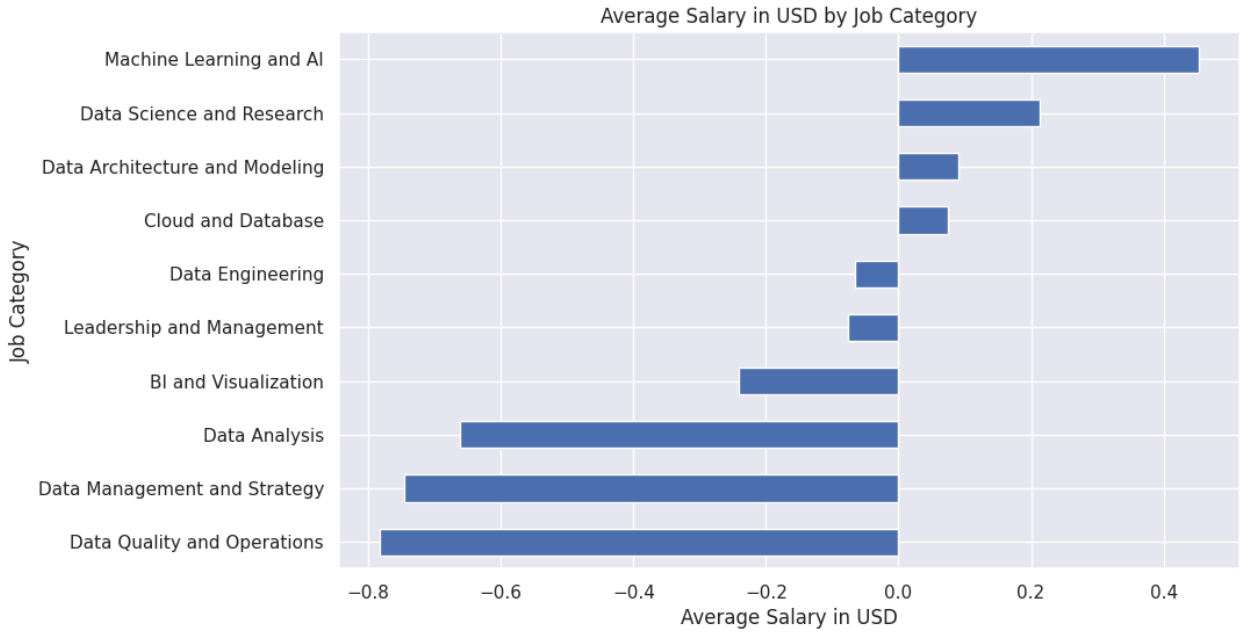
- Histograms depict the distribution of a numerical variable, showing the frequency of data points within specific ranges. A normalized histogram adjusts this to represent the proportion of observations, facilitating comparison of distribution shapes.
- Outlier Handling with IQR involves identifying data points that significantly deviate from the rest using the Interquartile Range (IQR) method. It's crucial for ensuring the robustness of statistical analyses.

1: Create Bar Graph and Contingency Table

Bar Graph: Let's visualize the average salary in USD by job category.

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

avg_salary_by_category = df.groupby('job_category')['salary_in_usd'].mean().sort_values()
avg_salary_by_category.plot(kind='barh', figsize=(10, 6))
plt.xlabel('Average Salary in USD')
plt.ylabel('Job Category')
plt.title('Average Salary in USD by Job Category')
plt.show()
```



Contingency Table: Let's create a contingency table between job_category and company_location.

```
contingency_table = pd.crosstab(df['job_category'], df['company_location'])
print(contingency_table)
```

company_location	Algeria	American Samoa	Andorra	Argentina
job_category				
BI and Visualization	0	0	0	0
Cloud and Database	0	0	0	0
Data Analysis	0	1	0	1
Data Architecture and Modeling	0	0	0	0
Data Engineering	0	0	0	4
Data Management and Strategy	0	0	0	0
Data Quality and Operations	0	0	0	0
Data Science and Research	1	0	1	0
Leadership and Management	0	0	0	0
Machine Learning and AI	0	0	0	0

company_location	Armenia	Australia	Austria	Bahamas	Belgium
job_category					
BI and Visualization	0	0	0	0	0
Cloud and Database	0	0	0	0	0
Data Analysis	0	6	0	0	0
Data Architecture and Modeling	0	0	0	0	0
Data Engineering	0	2	1	0	0
Data Management and Strategy	0	0	0	0	0
Data Quality and Operations	0	0	0	0	0
Data Science and Research	0	3	5	0	2
Leadership and Management	0	0	0	0	0
Machine Learning and AI	1	13	0	1	2

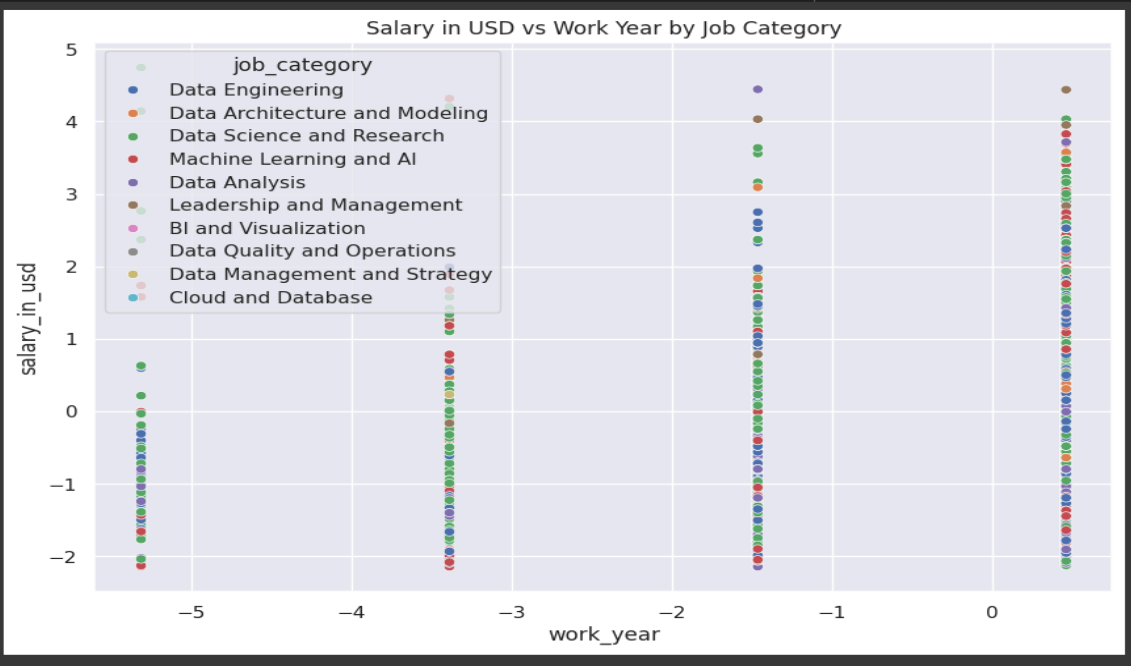
company_location	Bosnia and Herzegovina	...	South Korea
job_category		...	
BI and Visualization	0	...	0
Cloud and Database	0	...	0
Data Analysis	0	...	0
Data Architecture and Modeling	0	...	0
Data Engineering	0	...	0
Data Management and Strategy	0	...	0
Data Quality and Operations	0	...	0
Data Science and Research	0	...	0
Leadership and Management	0	...	0
Machine Learning and AI	1	...	2

company_location	Spain	Sweden	Switzerland	Thailand	Turkey
job_category					
BI and Visualization	0	0	0	0	1
Cloud and Database	0	0	0	0	0
Data Analysis	17	0	0	0	0
Data Architecture and Modeling	0	0	0	0	0
Data Engineering	26	2	0	0	1
Data Management and Strategy	0	0	0	0	0

2: Plot Scatter Plot, Box Plot, Heatmap

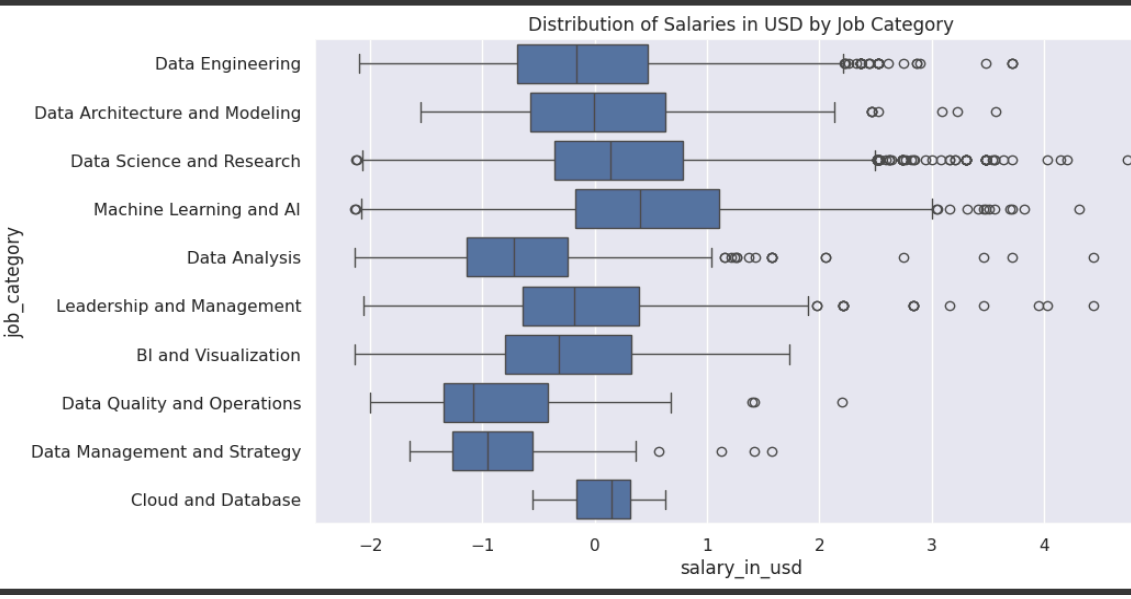
Scatter Plot: Visualize the relationship between work_year and salary_in_usd.

```
sns.scatterplot(data=df, x='work_year', y='salary_in_usd', hue='job_category')
plt.title('Salary in USD vs Work Year by Job Category')
plt.show()
```



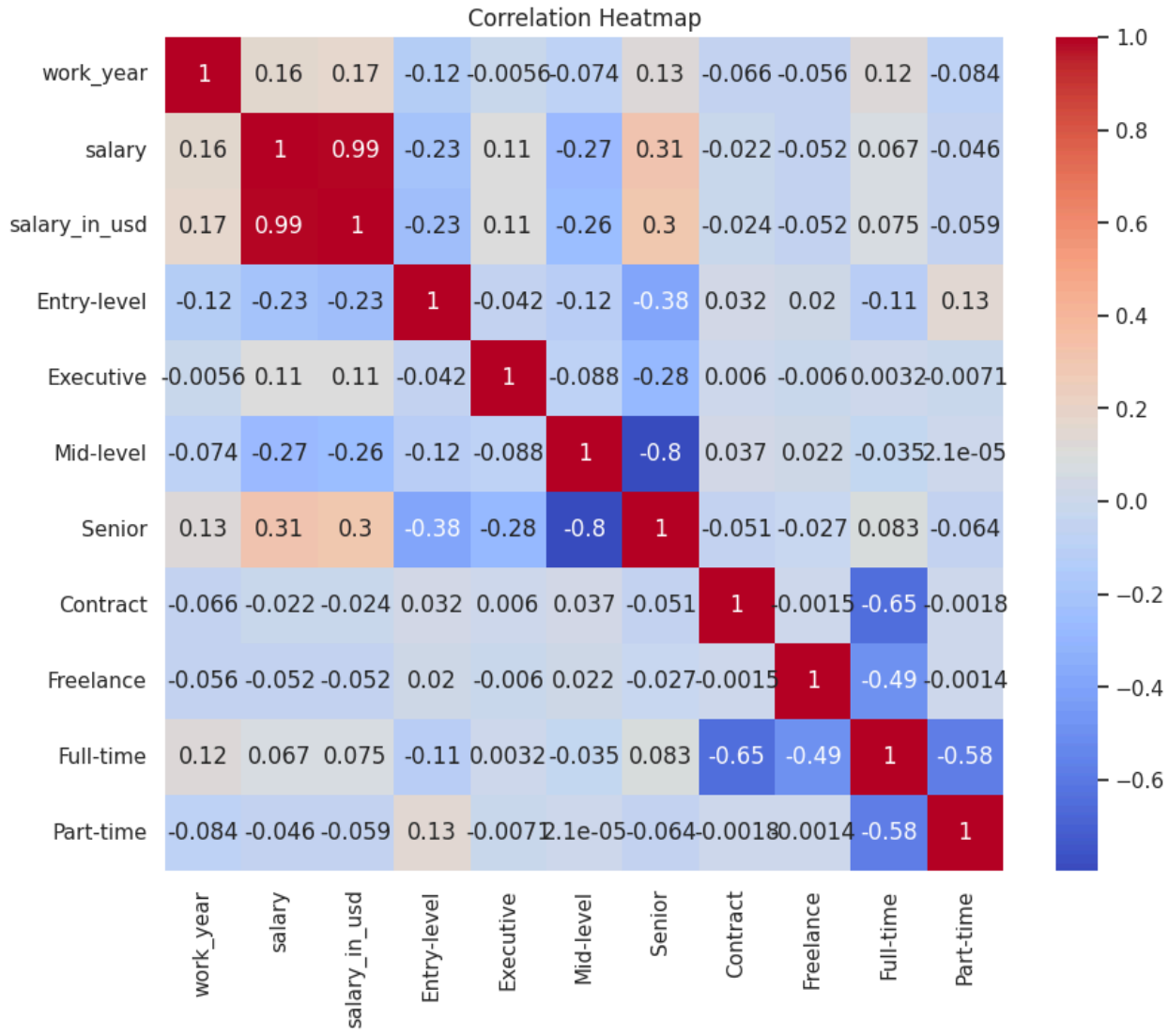
Box Plot: Distribution of salaries in USD by job category.

```
plt.figure(figsize=(10, 6))
sns.boxplot(data=df, x='salary_in_usd', y='job_category')
plt.title('Distribution of Salaries in USD by Job Category')
plt.show()
```



Heatmap: Correlation between numerical features.

```
[ ]  corr = df.select_dtypes(include=np.number).corr()
     plt.figure(figsize=(10, 8))
     sns.heatmap(corr, annot=True, cmap='coolwarm')
     plt.title('Correlation Heatmap')
     plt.show()
```

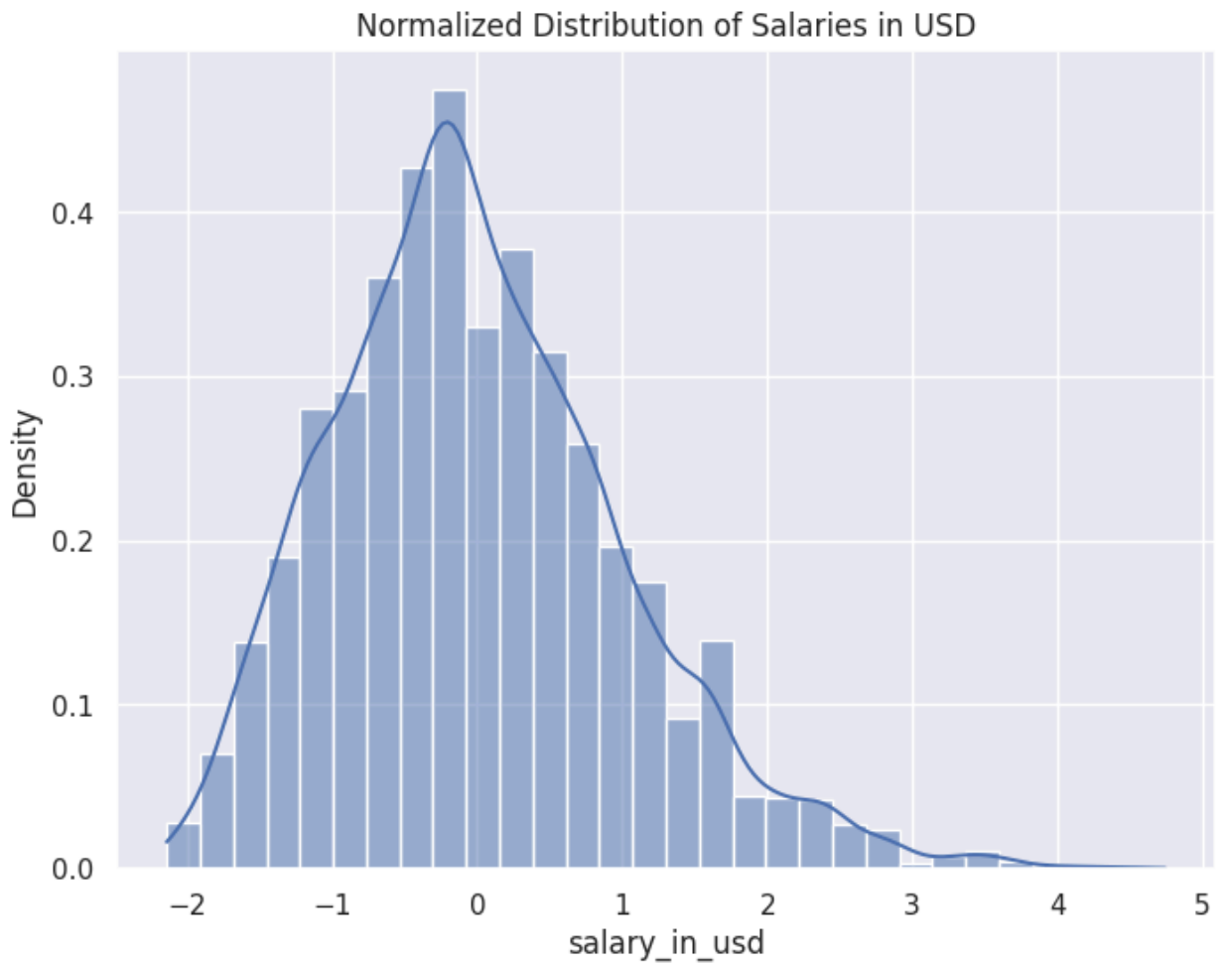


3: Create Histogram and Normalized Histogram
Histogram: Distribution of salaries in USD.

```
[ ]  plt.figure(figsize=(8, 6))
     sns.histplot(df['salary_in_usd'], bins=30, kde=True)
     plt.title('Distribution of Salaries in USD')
     plt.show()
```

Normalized Histogram: Let's normalize the histogram for salary_in_usd.

```
[ ]  plt.figure(figsize=(8, 6))
     sns.histplot(df['salary_in_usd'], bins=30, kde=True, stat="density")
     plt.title('Normalized Distribution of Salaries in USD')
     plt.show()
```



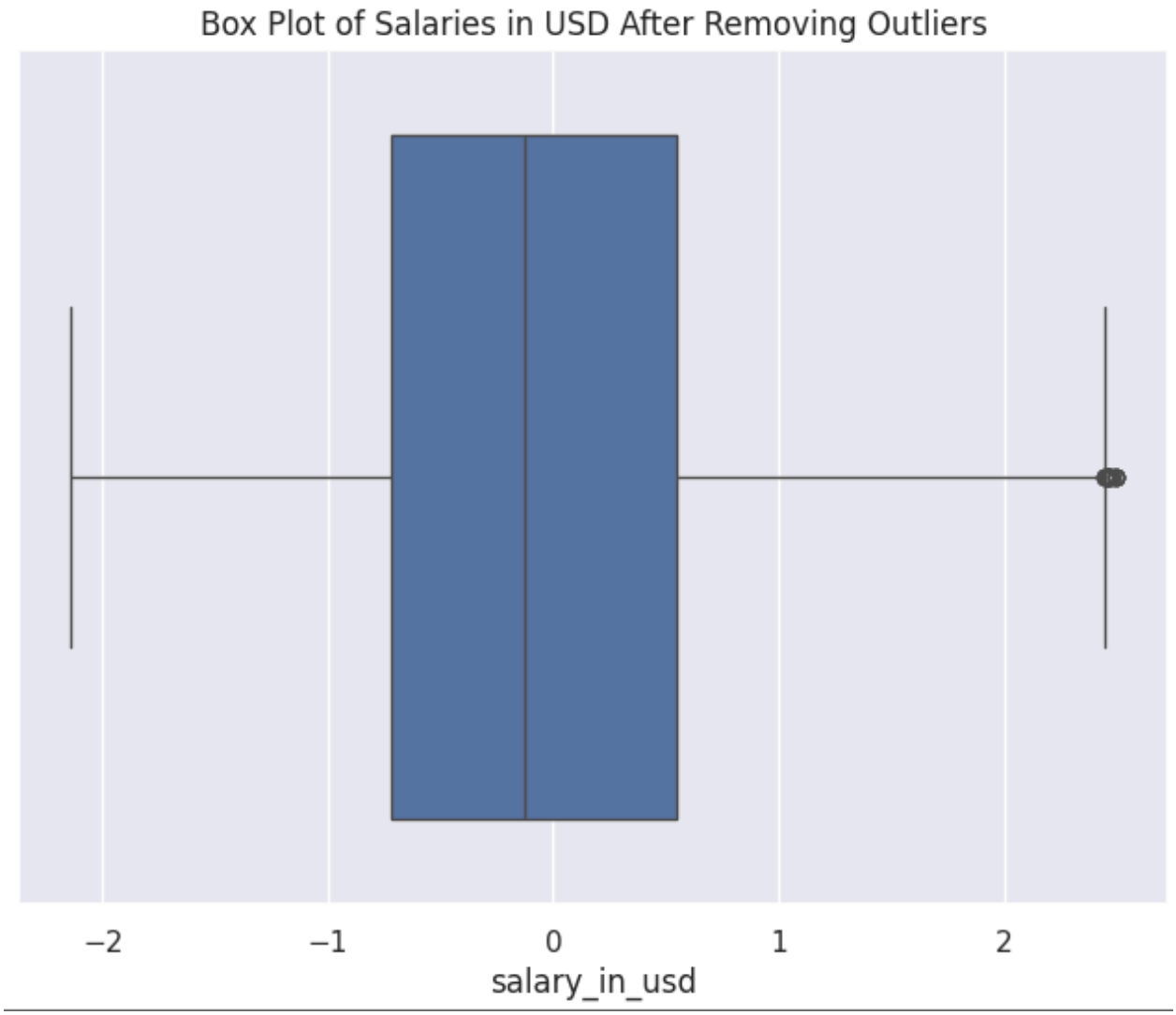
4: Handle Outlier Using Box Plot and Interquartile Range

To handle outliers in salary_in_usd:

```
Q1 = df['salary_in_usd'].quantile(0.25)
Q3 = df['salary_in_usd'].quantile(0.75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

df_filtered = df[(df['salary_in_usd'] >= lower_bound) & (df['salary_in_usd'] <= upper_bound)]

plt.figure(figsize=(8, 6))
sns.boxplot(data=df_filtered, x='salary_in_usd')
plt.title('Box Plot of Salaries in USD After Removing Outliers')
plt.show()
```



Conclusion: Data visualization and exploratory data analysis using tools like Matplotlib and Seaborn enable data scientists to uncover insights from data, communicate findings, and make data-driven decisions. Through various types of plots and analytical techniques, one can explore data distributions, relationships, and patterns, setting a solid foundation for further analysis, hypothesis testing, and predictive modeling.