DATA EXPLORATION USING NON-PARAMETRIC METHODS

```
#Aim: To explore and analyze salary trends and relationships across various
job titles, experience levels, and employment types in the dataset.

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

#Loading the dataset

data = pd.read_csv('data_science_salaries.csv')
data
```

```
                    job_title experience_level employment_type work_models
\
0                Data Engineer        Mid-level       Full-time      Remote
1                Data Engineer        Mid-level       Full-time      Remote
2               Data Scientist     Senior-level       Full-time      Remote
3               Data Scientist     Senior-level       Full-time      Remote
4                 BI Developer        Mid-level       Full-time     On-site
...                        ...              ...             ...         ...
6594        Staff Data Analyst      Entry-level        Contract      Hybrid
6595        Staff Data Analyst  Executive-level       Full-time     On-site
6596  Machine Learning Manager     Senior-level       Full-time      Hybrid
6597             Data Engineer        Mid-level       Full-time      Hybrid
6598            Data Scientist     Senior-level       Full-time     On-site

      work_year employee_residence  salary salary_currency  salary_in_usd  \
0          2024      United States  148100             USD         148100
1          2024      United States   98700             USD          98700
2          2024      United States  140032             USD         140032
3          2024      United States  100022             USD         100022
4          2024      United States  120000             USD         120000
...         ...                ...     ...             ...            ...
6594       2020             Canada   60000             CAD          44753
6595       2020            Nigeria   15000             USD          15000
6596       2020             Canada  157000             CAD         117104
6597       2020            Austria   65000             EUR          74130
6598       2020            Austria   80000             EUR          91237

     company_location company_size
0       United States       Medium
1       United States       Medium
2       United States       Medium
3       United States       Medium
4       United States       Medium
...               ...          ...
6594           Canada        Large
6595           Canada       Medium
6596           Canada        Large
6597          Austria        Large
6598          Austria        Small
```

```
[6599 rows x 11 columns]
```

```
#Basic information about the dataset like the number of samples, features, data types, etc.

print(data.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6599 entries, 0 to 6598
Data columns (total 11 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   job_title           6599 non-null   object
 1   experience_level    6599 non-null   object
 2   employment_type     6599 non-null   object
 3   work_models         6599 non-null   object
 4   work_year           6599 non-null   int64 
 5   employee_residence  6599 non-null   object
 6   salary              6599 non-null   int64 
 7   salary_currency     6599 non-null   object
 8   salary_in_usd       6599 non-null   int64 
 9   company_location    6599 non-null   object
 10  company_size        6599 non-null   object
dtypes: int64(3), object(8)
memory usage: 567.2+ KB
None
```

DATA EXPLORATION USING NON-PARAMETRIC METHODS

```
#First few rows to understand the structure and format of the data

print(data.head())
```

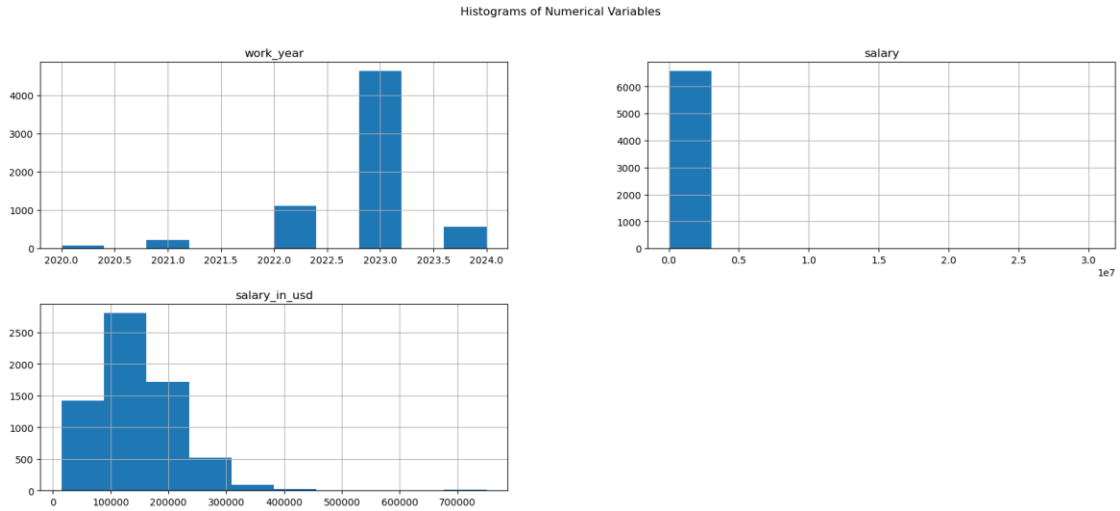
```
        job_title experience_level employment_type work_models  work_year  \
0   Data Engineer        Mid-level       Full-time      Remote       2024
1   Data Engineer        Mid-level       Full-time      Remote       2024
2  Data Scientist     Senior-level       Full-time      Remote       2024
3  Data Scientist     Senior-level       Full-time      Remote       2024
4    BI Developer        Mid-level       Full-time     On-site       2024

  employee_residence  salary salary_currency  salary_in_usd company_location
\
0      United States  148100             USD         148100    United States
1      United States   98700             USD          98700    United States
2      United States  140032             USD         140032    United States
3      United States  100022             USD         100022    United States
4      United States  120000             USD         120000    United States

  company_size
0       Medium
1       Medium
2       Medium
3       Medium
4       Medium
```

```
#Univariate Analysis:
#For numerical variables:
#Basic descriptive statistics

print(data.describe())
```

```
         work_year        salary  salary_in_usd
count  6599.000000  6.599000e+03    6599.000000
mean   2022.818457  1.792833e+05  145560.558569
std       0.674809  5.263722e+05   70946.838070
min    2020.000000  1.400000e+04   15000.000000
25%    2023.000000  9.600000e+04   95000.000000
50%    2023.000000  1.400000e+05  138666.000000
75%    2023.000000  1.875000e+05  185000.000000
max    2024.000000  3.040000e+07  750000.000000
```

DATA EXPLORATION USING NON-PARAMETRIC METHODS

```
#Visualization

data.hist(figsize=(20, 8))
plt.suptitle('Histograms of Numerical Variables')
plt.show()
```



DATA EXPLORATION USING NON-PARAMETRIC METHODS

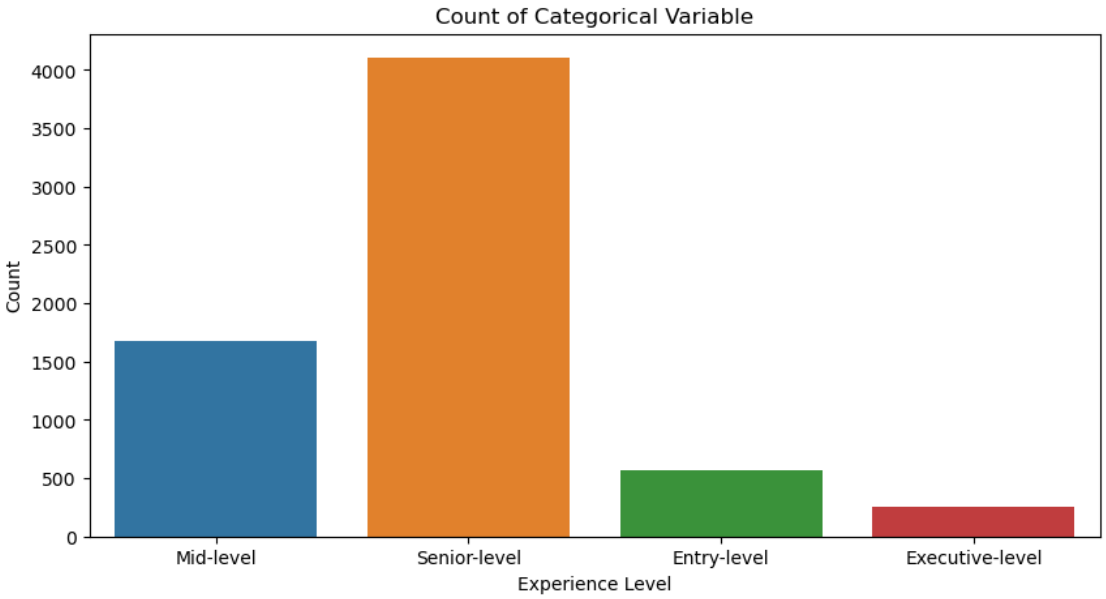
```python
#For categorical variables:
#Frequency tables showing counts and percentages

print("Frequency table for categorical variables:")
for column in data.select_dtypes(include=['object']):
    print(data[column].value_counts(normalize=True))
    print()
```

```
Frequency table for categorical variables:
job_title
Data Engineer                0.198060
Data Scientist               0.188362
Data Analyst                 0.137900
Machine Learning Engineer    0.095317
Analytics Engineer           0.037278
                               ...
Deep Learning Researcher     0.000152
Power BI Developer           0.000152
Marketing Data Scientist     0.000152
AI Product Manager           0.000152
Sales Data Analyst           0.000152
Name: proportion, Length: 132, dtype: float64

experience_level
Senior-level       0.622064
Mid-level          0.253826
Entry-level        0.085619
Executive-level    0.038491
Name: proportion, dtype: float64

employment_type
Full-time    0.992878
Contract     0.002879
Part-time    0.002425
Freelance    0.001818
Name: proportion, dtype: float64

work_models
On-site    0.577815
Remote     0.388089
Hybrid     0.034096
Name: proportion, dtype: float64

employee_residence
United States     0.803910
United Kingdom    0.060767
Canada            0.036521
Germany           0.010759
India             0.010608
                    ...
```

```
Georgia           0.000152
Israel            0.000152
Qatar             0.000152
Peru              0.000152
Honduras          0.000152
Name: proportion, Length: 87, dtype: float64

salary_currency
USD    0.883013
GBP    0.050614
EUR    0.044249
INR    0.007728
CAD    0.005910
AUD    0.001667
PLN    0.001061
SGD    0.000909
CHF    0.000758
JPY    0.000606
BRL    0.000606
DKK    0.000455
HUF    0.000455
TRY    0.000455
NOK    0.000303
THB    0.000303
CLP    0.000152
ILS    0.000152
HKD    0.000152
PHP    0.000152
ZAR    0.000152
MXN    0.000152
Name: proportion, dtype: float64

company_location
United States             0.811335
United Kingdom            0.061828
Canada                    0.036824
Germany                   0.011820
Spain                     0.009547
                            ...
Armenia                   0.000152
Bosnia and Herzegovina    0.000152
Qatar                     0.000152
Ecuador                   0.000152
Honduras                  0.000152
Name: proportion, Length: 75, dtype: float64

company_size
Medium    0.888013
Large     0.086225
Small     0.025761
```

```
Name: proportion, dtype: float64
```

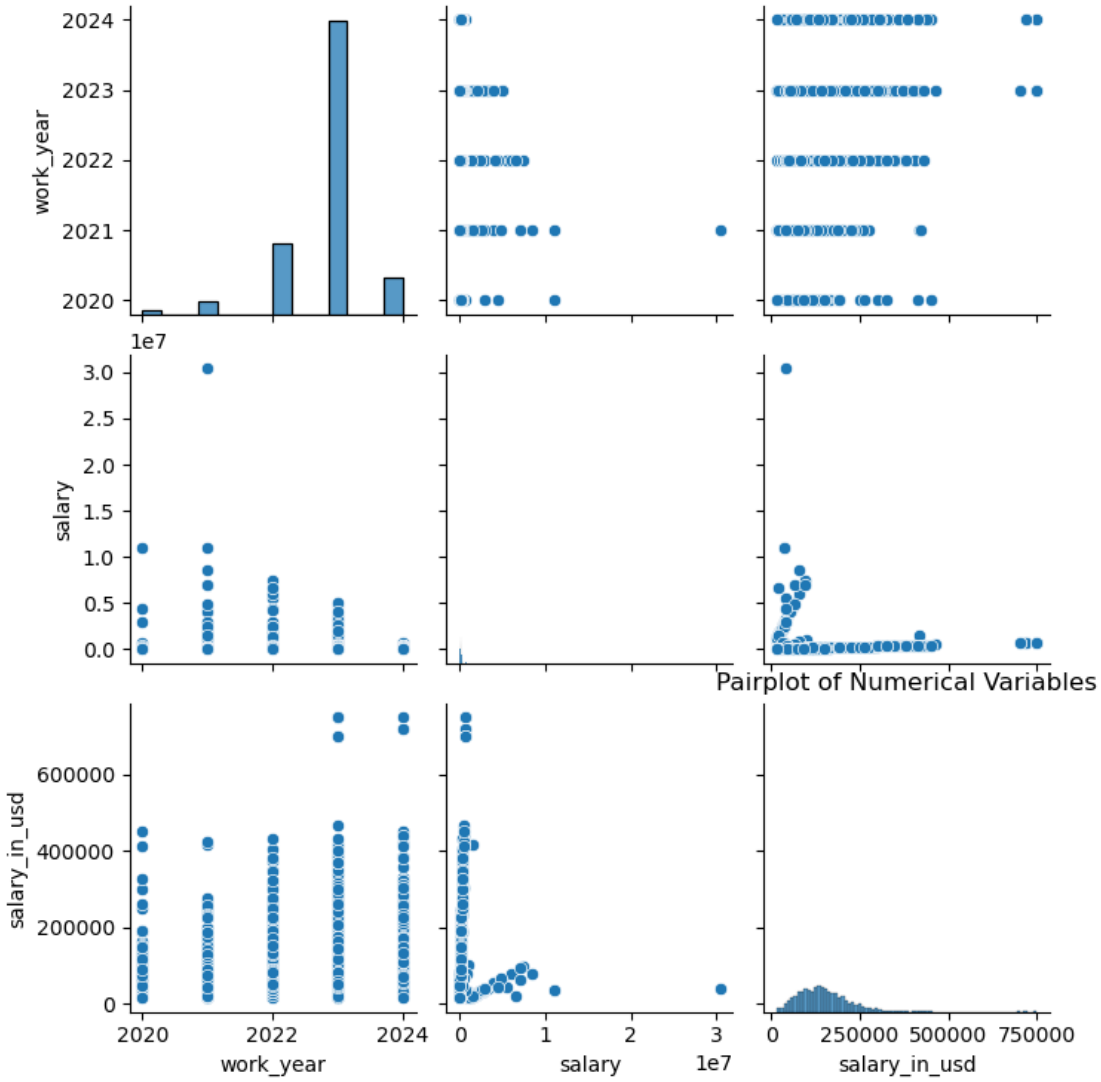
```
#Visualization using bar plots

import seaborn as sns
plt.figure(figsize=(10, 5))
sns.countplot(data=data, x='experience_level')
plt.title('Count of Categorical Variable')
plt.xlabel('Experience Level')
plt.ylabel('Count')
plt.show()
```



DATA EXPLORATION USING NON-PARAMETRIC METHODS

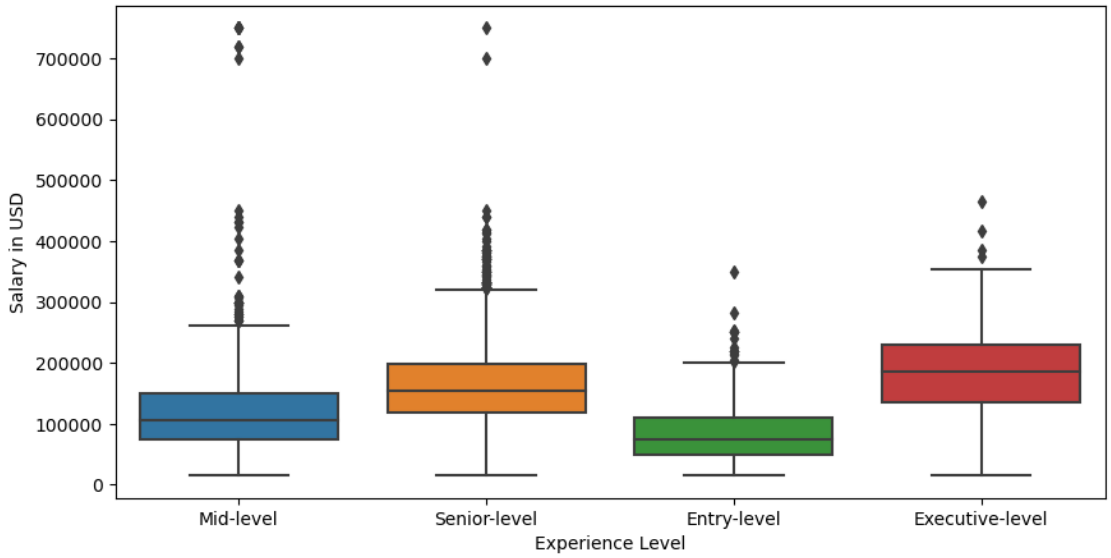
```
#Bivariate Analysis:
#Relationships between pairs of numerical variables using pair plots

sns.pairplot(data)
plt.title('Pairplot of Numerical Variables')
plt.show()
```

```
C:\Users\hp\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning:
The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```



DATA EXPLORATION USING NON-PARAMETRIC METHODS

```
#Relationships between numerical and categorical variables using box plots

plt.figure(figsize=(10, 5))
sns.boxplot(data=data, x='experience_level', y='salary_in_usd')
plt.xlabel('Experience Level')
plt.ylabel('Salary in USD')
plt.show()
```




```
# Replace string values with float values
data['experience_level'] = data['experience_level'].replace({'Mid-level':
1.0, 'Senior-level': 2.0, 'Entry-level': 3.0, 'Executive-level': 4.0})
data['employment_type'] = data['employment_type'].replace({'Full-time':
1.0,'Part-time':2.0,'Contract':3.0,'Freelance':4.0})

# Calculate correlation matrix
correlation_matrix = data[['experience_level', 'salary_in_usd']].corr()

print("Correlation Matrix:", correlation_matrix)
```

```
Correlation Matrix:                  experience_level  salary_in_usd
experience_level          1.000000        0.099199
salary_in_usd             0.099199        1.000000
```

DATA EXPLORATION USING NON-PARAMETRIC METHODS

```
# Non-parametric methods:

# Spearman rank correlation
from scipy.stats import spearmanr
spearman_coefficient, p_value = spearmanr(data['experience_level'],
data['salary_in_usd'])

print("Spearman correlation coefficient:", spearman_coefficient)
print("P-value:", p_value)
```

```
Spearman correlation coefficient: 0.12370654101664764
P-value: 6.367191758829943e-24
```

```
# Mann-Whitney U test
from scipy.stats import mannwhitneyu
statistic,p_value=mannwhitneyu(data['experience_level'],data['salary_in_usd']
)
print("Mann-Whitney U statistic:",statistic)
print("p_value:",p_value)
```

```
Mann-Whitney U statistic: 0.0
p_value: 0.0
```

```
# Wilcoxon signed-rank test
from scipy.stats import wilcoxon
wilcoxon,p_value=wilcoxon(data['experience_level'],data['salary_in_usd'])
print("wilcoxon statistic:",wilcoxon)
print("p_value:",p_value)
```

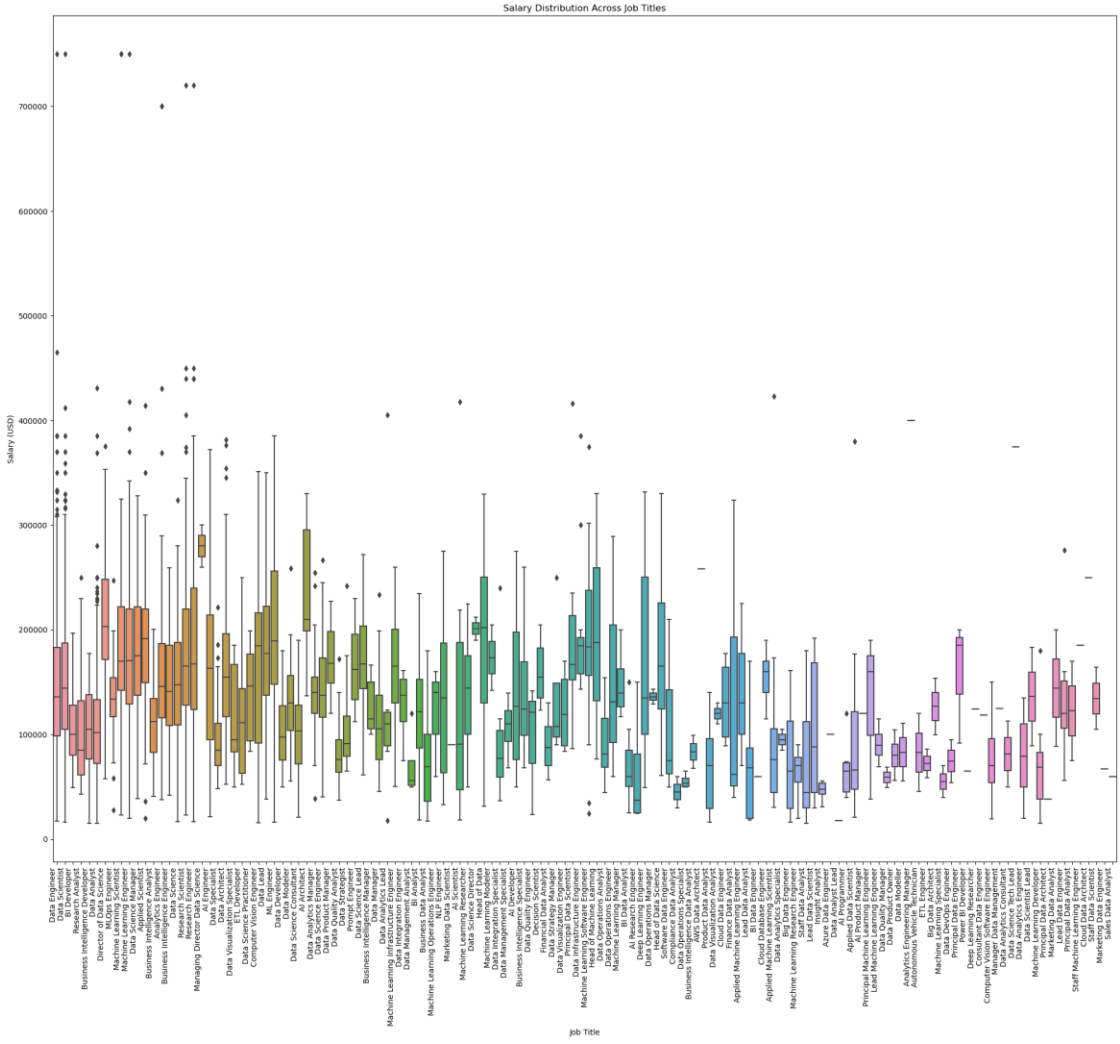
```
wilcoxon statistic: 0.0
p_value: 0.0
```

```
# Friedman Test
from scipy.stats import friedmanchisquare
friedmanchisquare,p_value=friedmanchisquare(data['experience_level'],data['sa
lary_in_usd'],data['employment_type'])
print("friedman test statistic:",friedmanchisquare)
print("p_value:",p_value)
```

```
friedman test statistic: 12489.04948253557
p_value: 0.0
```

DATA EXPLORATION USING NON-PARAMETRIC METHODS

```
#Visualization

plt.figure(figsize=(25, 20))
sns.boxplot(x='job_title', y='salary_in_usd', data=data)
plt.title('Salary Distribution Across Job Titles')
plt.xlabel('Job Title')
plt.ylabel('Salary (USD)')
plt.xticks(rotation=90, ha='right')  # Rotate x-axis labels for better visibility
plt.show()
```



DATA EXPLORATION USING NON-PARAMETRIC METHODS

*#**Conclusion:** The analysis reveals diverse salary distributions across job titles and experience levels, with significant variations in compensation based on employment type and experience level.*

#Mann-Whitney U Test: The Mann-Whitney U test indicates a significant difference between experience levels and salary, suggesting that the salary distributions differ across various experience levels.

#Wilcoxon Signed-Rank Test: The Wilcoxon signed-rank test demonstrates a significant difference between experience levels and salary, indicating variations in compensation based on experience level.

#Friedman Test: The Friedman test reveals a significant difference among experience levels, salary, and employment types, indicating that at least one of the variables significantly affects the others in the dataset.