CC5067NI-Smart Data Discovery

60% Individual Coursework

2023-24 Autumn

Student Name: Sikum Hangma Madi

London Met ID: 22085627

College ID: np01cp4s230077

Assignment Due Date: Monday, May 13, 2024

Assignment Submission Date: Monday, May 13, 2024

Word Count: 2490

I confirm that I understand my coursework needs to be submitted online via MySecondTeacher under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Table of Contents

Tables of Table

Table of Figures

1. Data Understanding

Data is an asset of the organization and business alike. Data understanding involves taking a closer look at the data available for mining. Data understanding involve the process of data accessing and exploring it using tables and graphics (IBM, 2024).

- **Data Collection**

  Gathering raw data from various accessible sources, including databases, spreadsheets, websites, forums, and individuals, is the initial stage in analysing data. The necessary data must be gathered for the project or firm. For this assignment, we will be using a spreadsheet as our data collection tool.

- **Data Description**

  To facilitate access and retrieval of the project's data, the acquired data must be described. Similar pieces of information are placed together, which defines the content and highlights the connections between it.

- **Data Exploration**

  Data Exploration refers to the initial step in data analysis in which data analysis use data visualization and statical techniques to describe the dataset characteristics in order to better understand the nature of the data (heavyAI, 2024).

- **Data Visualization**

  Data Visualization is the representation of data through visualization using common graphics such as charts, plots, graphs, and even animations. It helps make complex data much understandable (IBM, What is Data Visualization?, 2024)

- **Data Processing**

  Data Processing is the method of collecting data and translating it into useable information. It is essential for organizations to create better strategies and use the collected data. The useful data is stored and sent for data analysis.

  The dataset under study comprises 3755 data that records detail of employees within an organization. Each entry provides useful information of employment, such as work_year and experience_level, as well as employment_type, job_title, and so on. Additional salary, salary_currency, and salary_in_usd provide detailed information on wages, whereas employee_residence provides geographical distribution. Furthermore, details on remote work ratio, company_location, and company_size provide deep understanding into organizational structures.

  This dataset is an important resource for over seeing multiple aspects of employee characteristics and the overall organization, establishing informed decision-making and strategic planning.The objective of this analysis is to obtain a better understanding of the elements that influence the salaries of data scientists and discover any regularities or tendencies within the data.

	Column Name	**Description**	**Data Type**
1	Work_year	Provides the duration of employment in which an employee began their current position	INT
2	experience_level	Provides the level of professional experience of an employee.	VARCHAR

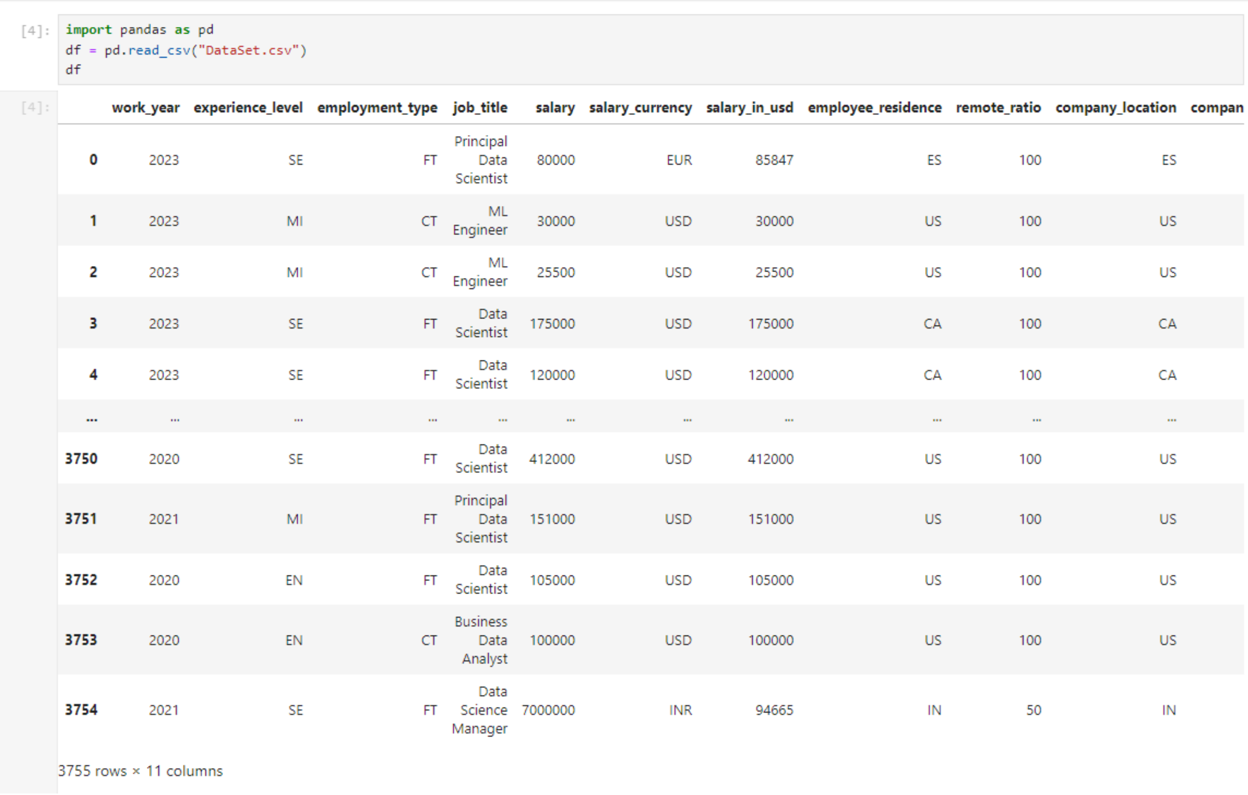
3	employment_type	Provides the type of employment status for each employee, classified as ‘FT(Full Time)’ and ‘CT(Contract)’,	VARCHAR
4	Job_title	Provides information about the roles of an employee in the company.	VARCHAR
5	Salary	Provides the monthly wages received by employees for their work in the company	INT
6	Salary_currency	Provides the currency in which salary are given for each employee according to their geographic location.	VARCHAR
7	Salary_in_usd	Provides the monthly wages received by an employee which is converted into United States Dollar(USD)	INT
8	Employe_residence	Provides the location where each employee resides	VARCHAR
9	Remote_ratio	Provides the proportion of employee working remotely in the company	INT
10	Company_location	Provides the locations where the companies are located	VARCHAR
11	Company_size	Categorizes companies based on their work size and employee count.	INT

Table 1 Data Understanding

2. Data Preparation

Data preparation is the process of preparing raw data so that it is suitable for further processing and analysis. Key steps include collecting, cleaning, and labeling raw data into a form suitable for machine learning (ML) algorithms and then exploring and visualizing the data (AWS, 2024).

2.1 Write a Python Program to load data into pandas DataFrame.



```
[4]: import pandas as pd
     df = pd.read_csv("DataSet.csv")
     df
```

[4]:

	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	compan
0	2023	SE	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	
1	2023	MI	CT	ML Engineer	30000	USD	30000	US	100	US	
2	2023	MI	CT	ML Engineer	25500	USD	25500	US	100	US	
3	2023	SE	FT	Data Scientist	175000	USD	175000	CA	100	CA	
4	2023	SE	FT	Data Scientist	120000	USD	120000	CA	100	CA	
...	...	...	...	...	...	...	...	...	...	...	
3750	2020	SE	FT	Data Scientist	412000	USD	412000	US	100	US	
3751	2021	MI	FT	Principal Data Scientist	151000	USD	151000	US	100	US	
3752	2020	EN	FT	Data Scientist	105000	USD	105000	US	100	US	
3753	2020	EN	CT	Business Data Analyst	100000	USD	100000	US	100	US	
3754	2021	SE	FT	Data Science Manager	7000000	INR	94665	IN	50	IN	

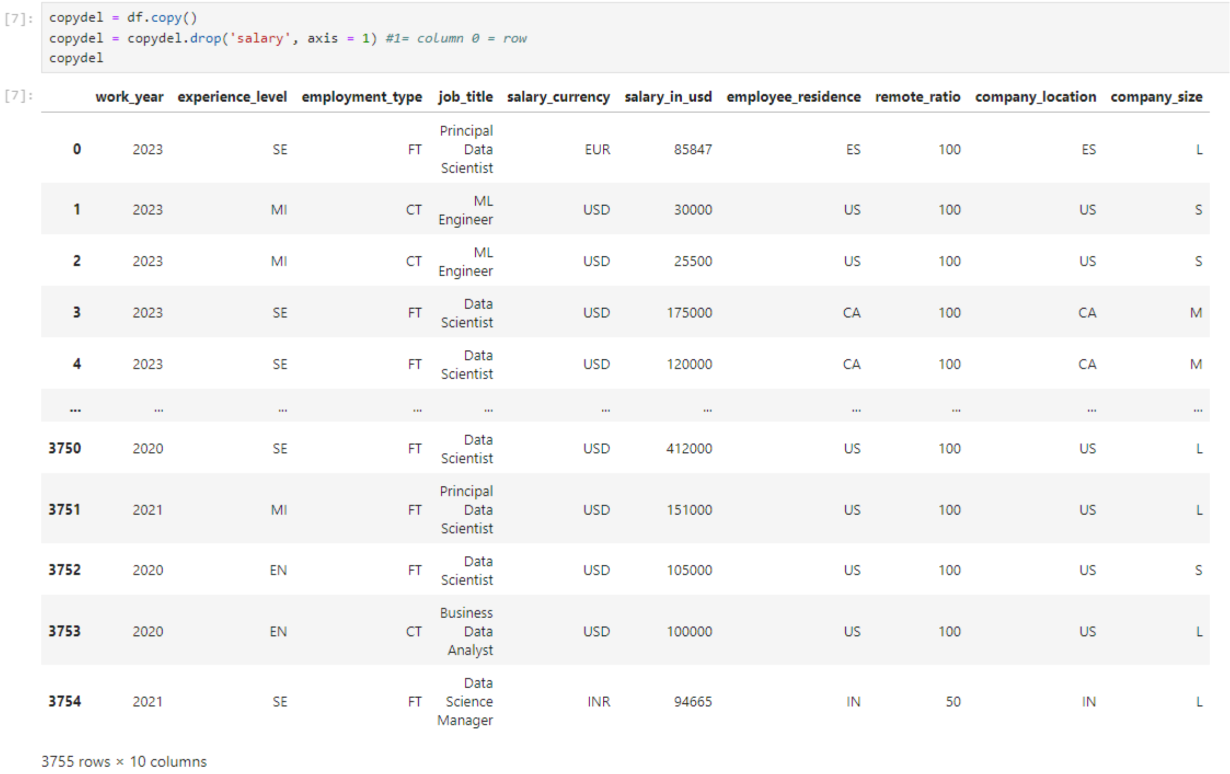
3755 rows × 11 columns

Figure 1 Python Program to load data into Pandas DataFrame

The above implemented code reads the CSV file named ds.csv from the directory. The loaded data is then loaded into pandas DataFrame as df. The csv file is read and then loaded from the CSV files in the director.

2.2 Write a python program to remove unnecessary columns i.e. salary and salary currency.

2.2.1 SALARY



```
[7]: copydel = df.copy()
     copydel = copydel.drop('salary', axis = 1) #1= column 0 = row
     copydel
```

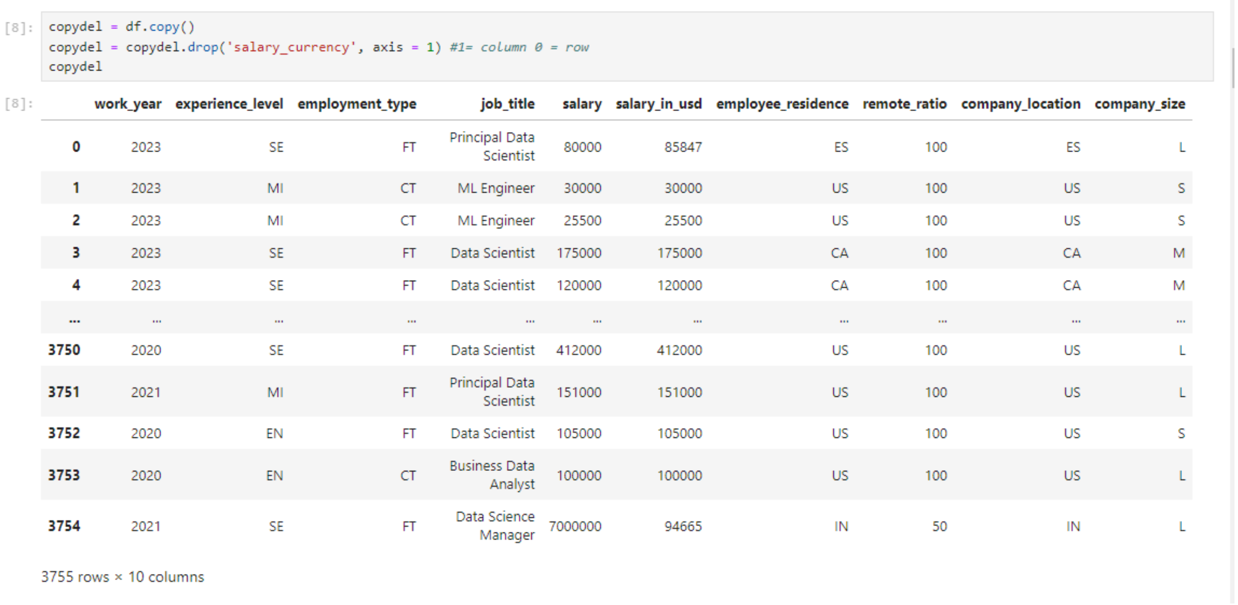
[7]:

	work_year	experience_level	employment_type	job_title	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	company_size
0	2023	SE	FT	Principal Data Scientist	EUR	85847	ES	100	ES	L
1	2023	MI	CT	ML Engineer	USD	30000	US	100	US	S
2	2023	MI	CT	ML Engineer	USD	25500	US	100	US	S
3	2023	SE	FT	Data Scientist	USD	175000	CA	100	CA	M
4	2023	SE	FT	Data Scientist	USD	120000	CA	100	CA	M
...	...	...	...	...	...	...	...	...	...	...
3750	2020	SE	FT	Data Scientist	USD	412000	US	100	US	L
3751	2021	MI	FT	Principal Data Scientist	USD	151000	US	100	US	L
3752	2020	EN	FT	Data Scientist	USD	105000	US	100	US	S
3753	2020	EN	CT	Business Data Analyst	USD	100000	US	100	US	L
3754	2021	SE	FT	Data Science Manager	INR	94665	IN	50	IN	L

3755 rows × 10 columns

Figure 2 Python Program, to remove Salary Column

2.2.2 SALARY CURRENCY



```
[8]: copydel = df.copy()
     copydel = copydel.drop('salary_currency', axis = 1) #1= column 0 = row
     copydel
```

[8]:

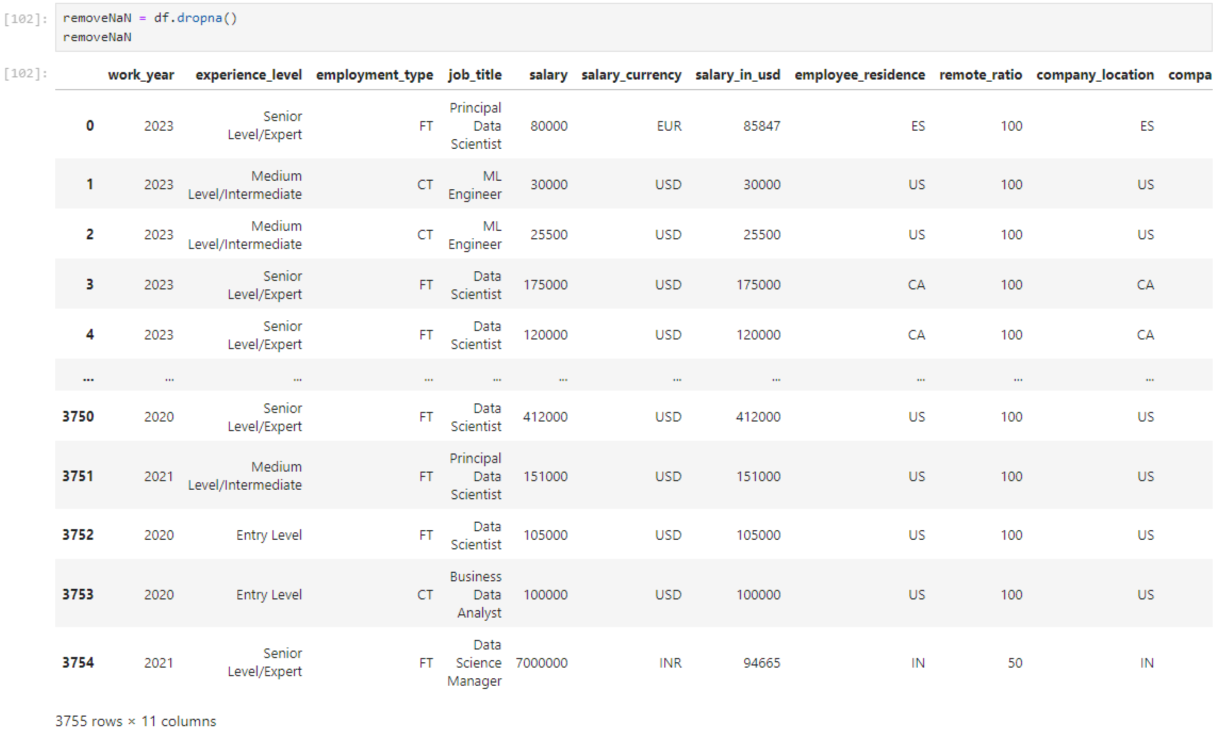
	work_year	experience_level	employment_type	job_title	salary	salary_in_usd	employee_residence	remote_ratio	company_location	company_size
0	2023	SE	FT	Principal Data Scientist	80000	85847	ES	100	ES	L
1	2023	MI	CT	ML Engineer	30000	30000	US	100	US	S
2	2023	MI	CT	ML Engineer	25500	25500	US	100	US	S
3	2023	SE	FT	Data Scientist	175000	175000	CA	100	CA	M
4	2023	SE	FT	Data Scientist	120000	120000	CA	100	CA	M
...	...	...	...	...	...	...	...	...	...	...
3750	2020	SE	FT	Data Scientist	412000	412000	US	100	US	L
3751	2021	MI	FT	Principal Data Scientist	151000	151000	US	100	US	L
3752	2020	EN	FT	Data Scientist	105000	105000	US	100	US	S
3753	2020	EN	CT	Business Data Analyst	100000	100000	US	100	US	L
3754	2021	SE	FT	Data Science Manager	7000000	94665	IN	50	IN	L

3755 rows × 10 columns

Figure 3 Python Program to remove Salary Currency

Here, df.drop() drops the columns salary and salary_currency as requested respectively. Then the dataset is printed again without the removed columns.

2.3 Write a python program to remove the NaN missing values from updated dataframe.



```
[102]: removeNaN = df.dropna()
       removeNaN
```

[102]:

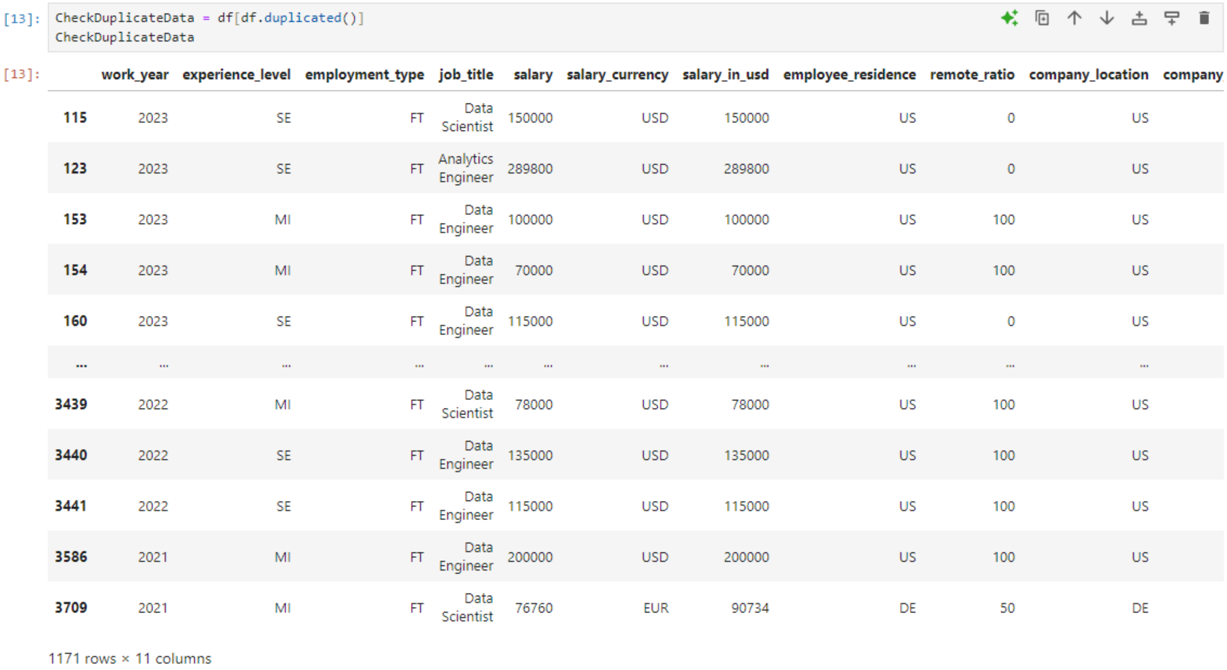
	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	compa
0	2023	Senior Level/Expert	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	
1	2023	Medium Level/Intermediate	CT	ML Engineer	30000	USD	30000	US	100	US	
2	2023	Medium Level/Intermediate	CT	ML Engineer	25500	USD	25500	US	100	US	
3	2023	Senior Level/Expert	FT	Data Scientist	175000	USD	175000	CA	100	CA	
4	2023	Senior Level/Expert	FT	Data Scientist	120000	USD	120000	CA	100	CA	
...	...	...	...	...	...	...	...	...	...	...	
3750	2020	Senior Level/Expert	FT	Data Scientist	412000	USD	412000	US	100	US	
3751	2021	Medium Level/Intermediate	FT	Principal Data Scientist	151000	USD	151000	US	100	US	
3752	2020	Entry Level	FT	Data Scientist	105000	USD	105000	US	100	US	
3753	2020	Entry Level	CT	Business Data Analyst	100000	USD	100000	US	100	US	
3754	2021	Senior Level/Expert	FT	Data Science Manager	7000000	INR	94665	IN	50	IN	

3755 rows × 11 columns

Figure 4 Python program to remove the NaN missing values form the update dataframes

df.dropna() is a pandas method to drop rows from a DataFrame that contains missing value or NaN(Not a Number). By default, dropna() removes rows containing any missing values.

2.4 Write a python program to check duplicates value in the dataframe.



```
[13]: CheckDuplicateData = df[df.duplicated()]
      CheckDuplicateData
```

[13]:

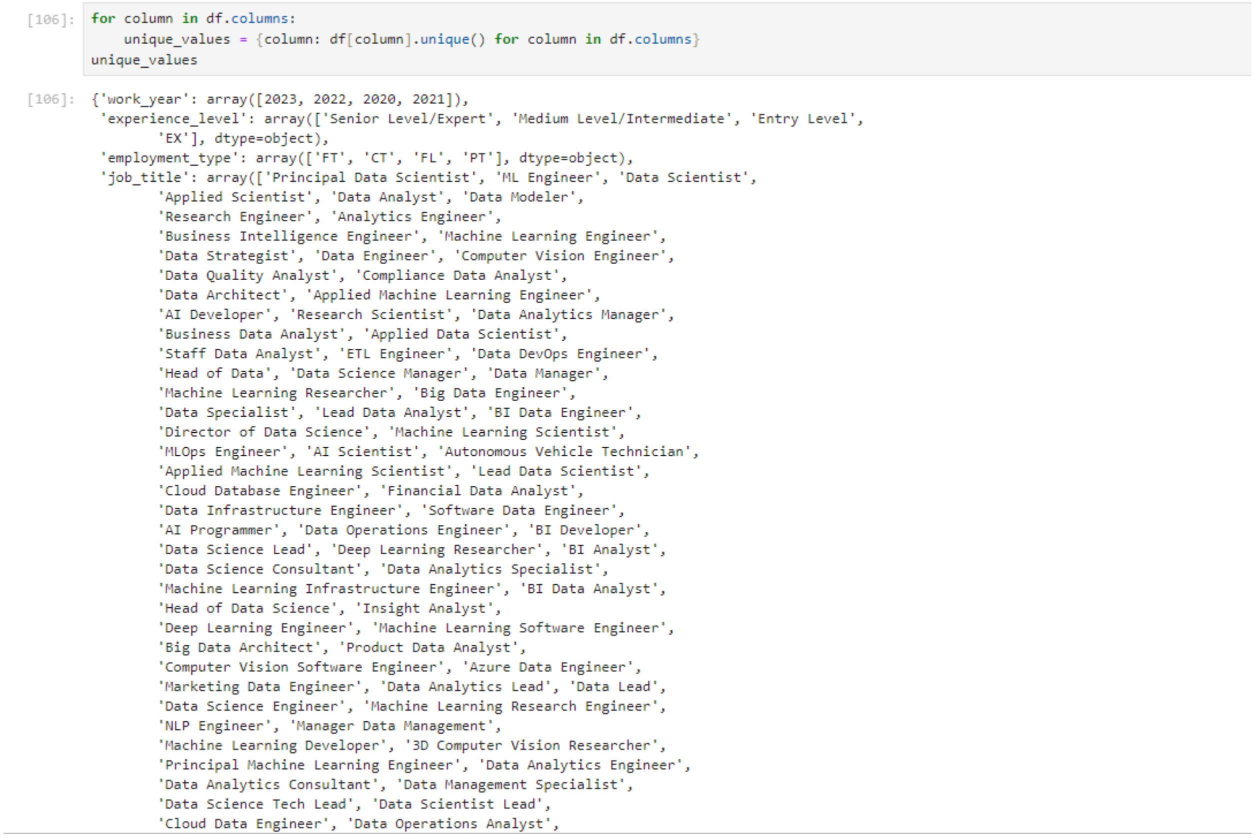
	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	company
115	2023	SE	FT	Data Scientist	150000	USD	150000	US	0	US	
123	2023	SE	FT	Analytics Engineer	289800	USD	289800	US	0	US	
153	2023	MI	FT	Data Engineer	100000	USD	100000	US	100	US	
154	2023	MI	FT	Data Engineer	70000	USD	70000	US	100	US	
160	2023	SE	FT	Data Engineer	115000	USD	115000	US	0	US	
...	...	...	...	...	...	...	...	...	...	...	
3439	2022	MI	FT	Data Scientist	78000	USD	78000	US	100	US	
3440	2022	SE	FT	Data Engineer	135000	USD	135000	US	100	US	
3441	2022	SE	FT	Data Engineer	115000	USD	115000	US	100	US	
3586	2021	MI	FT	Data Engineer	200000	USD	200000	US	100	US	
3709	2021	MI	FT	Data Scientist	76760	EUR	90734	DE	50	DE	

1171 rows × 11 columns

Figure 5 Python Program to check duplicate value in the dataframe

The duplicated() method in pandas is used to identify duplicate rows in a DataFrame. By default, this method considers all the columns when identifying duplicate rows.

2.5 Write a python program to see the unique values from all the columns in the dataframe.



```
[106]: for column in df.columns:
           unique_values = {column: df[column].unique() for column in df.columns}
       unique_values
```

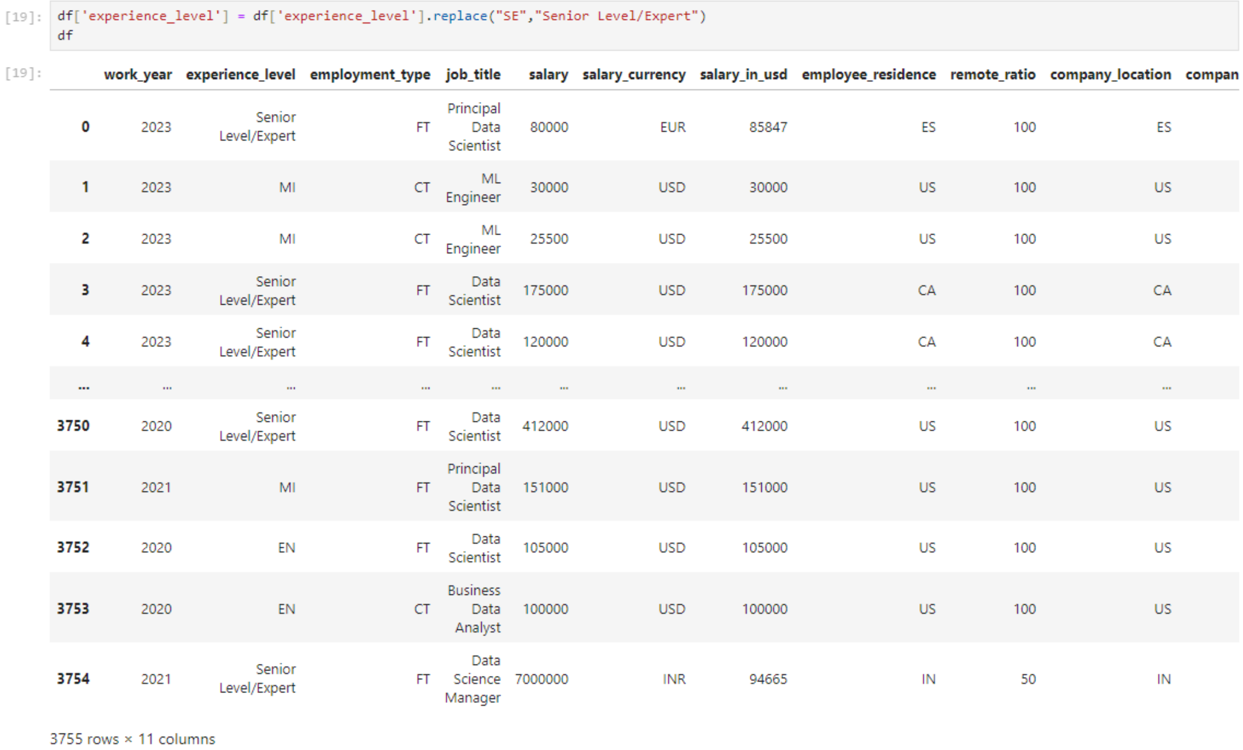
```
[106]: {'work_year': array([2023, 2022, 2020, 2021]),
        'experience_level': array(['Senior Level/Expert', 'Medium Level/Intermediate', 'Entry Level',
               'EX'], dtype=object),
        'employment_type': array(['FT', 'CT', 'FL', 'PT'], dtype=object),
        'job_title': array(['Principal Data Scientist', 'ML Engineer', 'Data Scientist',
               'Applied Scientist', 'Data Analyst', 'Data Modeler',
               'Research Engineer', 'Analytics Engineer',
               'Business Intelligence Engineer', 'Machine Learning Engineer',
               'Data Strategist', 'Data Engineer', 'Computer Vision Engineer',
               'Data Quality Analyst', 'Compliance Data Analyst',
               'Data Architect', 'Applied Machine Learning Engineer',
               'AI Developer', 'Research Scientist', 'Data Analytics Manager',
               'Business Data Analyst', 'Applied Data Scientist',
               'Staff Data Analyst', 'ETL Engineer', 'Data DevOps Engineer',
               'Head of Data', 'Data Science Manager', 'Data Manager',
               'Machine Learning Researcher', 'Big Data Engineer',
               'Data Specialist', 'Lead Data Analyst', 'BI Data Engineer',
               'Director of Data Science', 'Machine Learning Scientist',
               'MLOps Engineer', 'AI Scientist', 'Autonomous Vehicle Technician',
               'Applied Machine Learning Scientist', 'Lead Data Scientist',
               'Cloud Database Engineer', 'Financial Data Analyst',
               'Data Infrastructure Engineer', 'Software Data Engineer',
               'AI Programmer', 'Data Operations Engineer', 'BI Developer',
               'Data Science Lead', 'Deep Learning Researcher', 'BI Analyst',
               'Data Science Consultant', 'Data Analytics Specialist',
               'Machine Learning Infrastructure Engineer', 'BI Data Analyst',
               'Head of Data Science', 'Insight Analyst',
               'Deep Learning Engineer', 'Machine Learning Software Engineer',
               'Big Data Architect', 'Product Data Analyst',
               'Computer Vision Software Engineer', 'Azure Data Engineer',
               'Marketing Data Engineer', 'Data Analytics Lead', 'Data Lead',
               'Data Science Engineer', 'Machine Learning Research Engineer',
               'NLP Engineer', 'Manager Data Management',
               'Machine Learning Developer', '3D Computer Vision Researcher',
               'Principal Machine Learning Engineer', 'Data Analytics Engineer',
               'Data Analytics Consultant', 'Data Management Specialist',
               'Data Science Tech Lead', 'Data Scientist Lead',
               'Cloud Data Engineer', 'Data Operations Analyst',
```

Figure 6 python program to see the unique values from all the columns in the dataframe

Here, the code iterates over each column in the DataFrame. The df gets unique values. The dictionary is created here where the keys are column names, and the values are arrays of unique values for each column.

2.6 Rename the experience level columns as below.

2.6.1 SE-Senior Level/Expert



```
[19]: df['experience_level'] = df['experience_level'].replace("SE","Senior Level/Expert")
      df
```

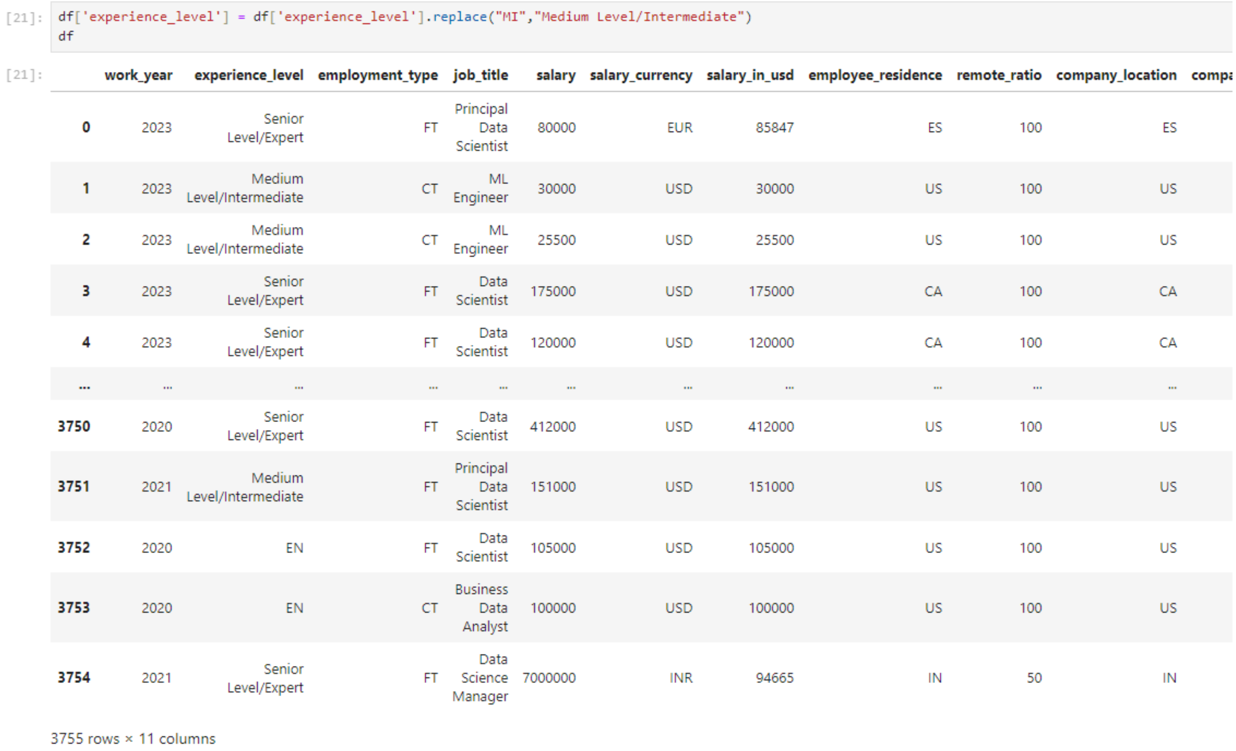
[19]:

	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	compan
0	2023	Senior Level/Expert	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	
1	2023	MI	CT	ML Engineer	30000	USD	30000	US	100	US	
2	2023	MI	CT	ML Engineer	25500	USD	25500	US	100	US	
3	2023	Senior Level/Expert	FT	Data Scientist	175000	USD	175000	CA	100	CA	
4	2023	Senior Level/Expert	FT	Data Scientist	120000	USD	120000	CA	100	CA	
...	...	...	...	...	...	...	...	...	...	...	
3750	2020	Senior Level/Expert	FT	Data Scientist	412000	USD	412000	US	100	US	
3751	2021	MI	FT	Principal Data Scientist	151000	USD	151000	US	100	US	
3752	2020	EN	FT	Data Scientist	105000	USD	105000	US	100	US	
3753	2020	EN	CT	Business Data Analyst	100000	USD	100000	US	100	US	
3754	2021	Senior Level/Expert	FT	Data Science Manager	7000000	INR	94665	IN	50	IN	

3755 rows × 11 columns

Figure 7 Renaming SE to Senior Level/Expert

2.6.2 MI-Medium Level/Intermediate



```
[21]: df['experience_level'] = df['experience_level'].replace("MI","Medium Level/Intermediate")
      df
```

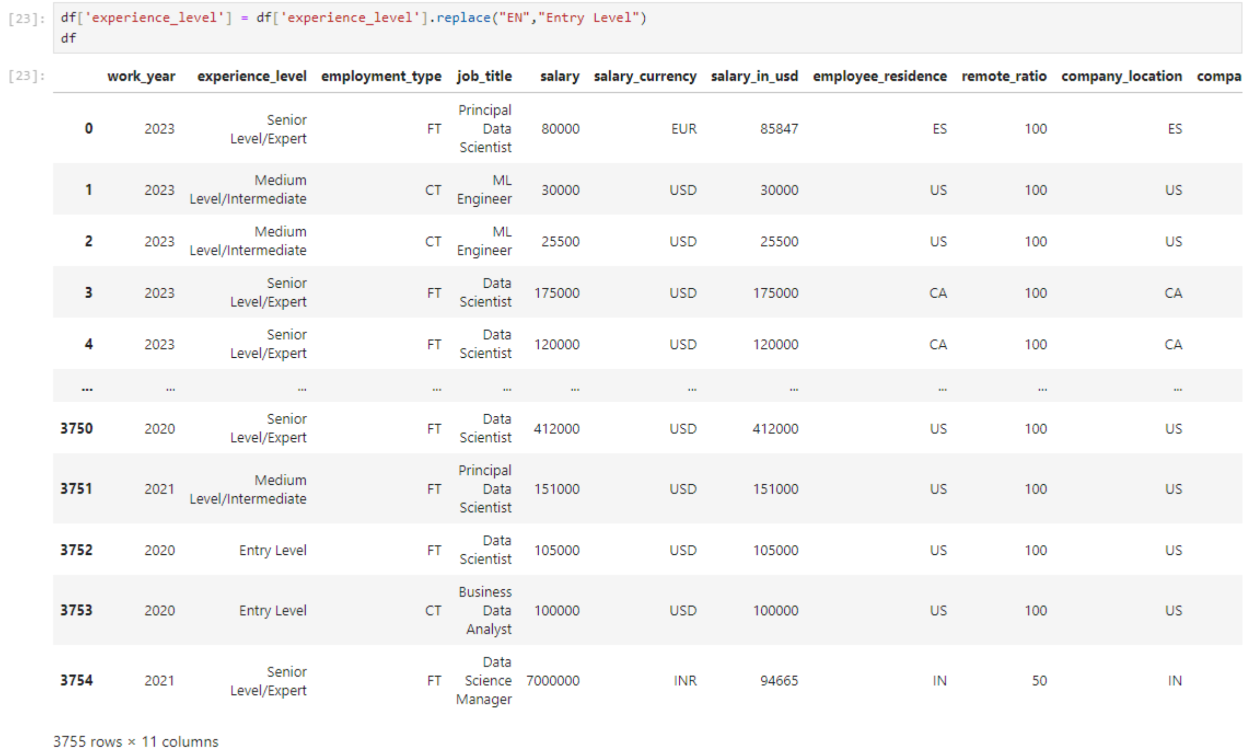
[21]:

	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	compa
0	2023	Senior Level/Expert	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	
1	2023	Medium Level/Intermediate	CT	ML Engineer	30000	USD	30000	US	100	US	
2	2023	Medium Level/Intermediate	CT	ML Engineer	25500	USD	25500	US	100	US	
3	2023	Senior Level/Expert	FT	Data Scientist	175000	USD	175000	CA	100	CA	
4	2023	Senior Level/Expert	FT	Data Scientist	120000	USD	120000	CA	100	CA	
...	...	...	...	...	...	...	...	...	...	...	
3750	2020	Senior Level/Expert	FT	Data Scientist	412000	USD	412000	US	100	US	
3751	2021	Medium Level/Intermediate	FT	Principal Data Scientist	151000	USD	151000	US	100	US	
3752	2020	EN	FT	Data Scientist	105000	USD	105000	US	100	US	
3753	2020	EN	CT	Business Data Analyst	100000	USD	100000	US	100	US	
3754	2021	Senior Level/Expert	FT	Data Science Manager	7000000	INR	94665	IN	50	IN	

3755 rows × 11 columns

Figure 8 Renaming ML to Medium Level/Intermediate

2.6.3 EN-Entry Level



```
[23]: df['experience_level'] = df['experience_level'].replace("EN","Entry Level")
      df
```

[23]:

	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	compa
0	2023	Senior Level/Expert	FT	Principal Data Scientist	80000	EUR	85847	ES	100	ES	
1	2023	Medium Level/Intermediate	CT	ML Engineer	30000	USD	30000	US	100	US	
2	2023	Medium Level/Intermediate	CT	ML Engineer	25500	USD	25500	US	100	US	
3	2023	Senior Level/Expert	FT	Data Scientist	175000	USD	175000	CA	100	CA	
4	2023	Senior Level/Expert	FT	Data Scientist	120000	USD	120000	CA	100	CA	
...	...	...	...	...	...	...	...	...	...	...	
3750	2020	Senior Level/Expert	FT	Data Scientist	412000	USD	412000	US	100	US	
3751	2021	Medium Level/Intermediate	FT	Principal Data Scientist	151000	USD	151000	US	100	US	
3752	2020	Entry Level	FT	Data Scientist	105000	USD	105000	US	100	US	
3753	2020	Entry Level	CT	Business Data Analyst	100000	USD	100000	US	100	US	
3754	2021	Senior Level/Expert	FT	Data Science Manager	7000000	INR	94665	IN	50	IN	

3755 rows × 11 columns

Figure 9 Renaming EN to Entry LevelFigure 9

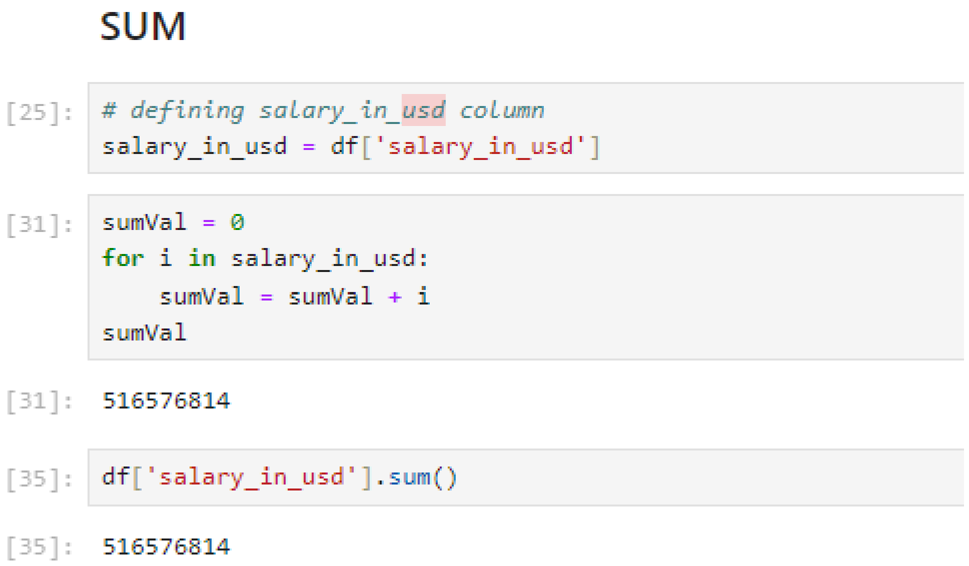
Here, the replace() method replaces the mentioned values or columns in a DataFrame. SE is replaced as Senior level/ Expert, MI as Medium Level/ Intermediate, EN as Entry Level, and EX as Executive Level in the column ‘experience_level’.

3. Data Analysis

Data analysis is a comprehensive method of inspecting, cleansing, transforming, and modelling data to discover useful information, draw conclusions, and support decision-making. It is a multifaceted process involving various techniques and methodologies to interpret data from various sources in different formats, both structured and unstructured (Nehma, 2024).

3.1 Write a Python Program to show summary statistics of sum, mean, standard deviation, skewness and kurtosis of any chosen variable.

3.1.1 SUM

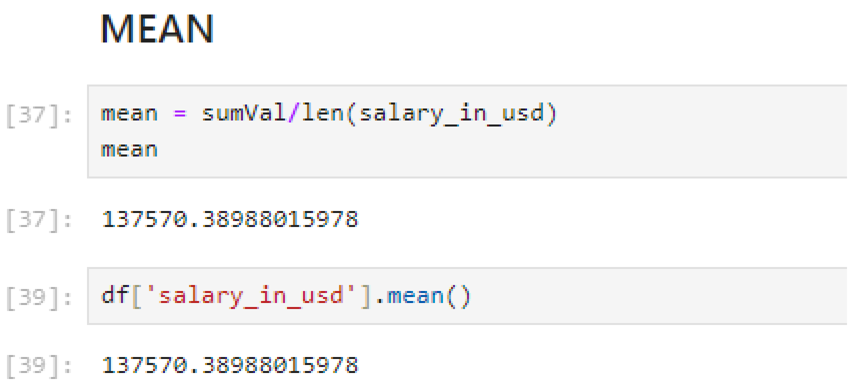


SUM

```
[25]: # defining salary_in_usd column
      salary_in_usd = df['salary_in_usd']

[31]: sumVal = 0
      for i in salary_in_usd:
          sumVal = sumVal + i
      sumVal

[31]: 516576814

[35]: df['salary_in_usd'].sum()

[35]: 516576814
```

Figure 10 Statistics of SUM

3.1.2 MEAN

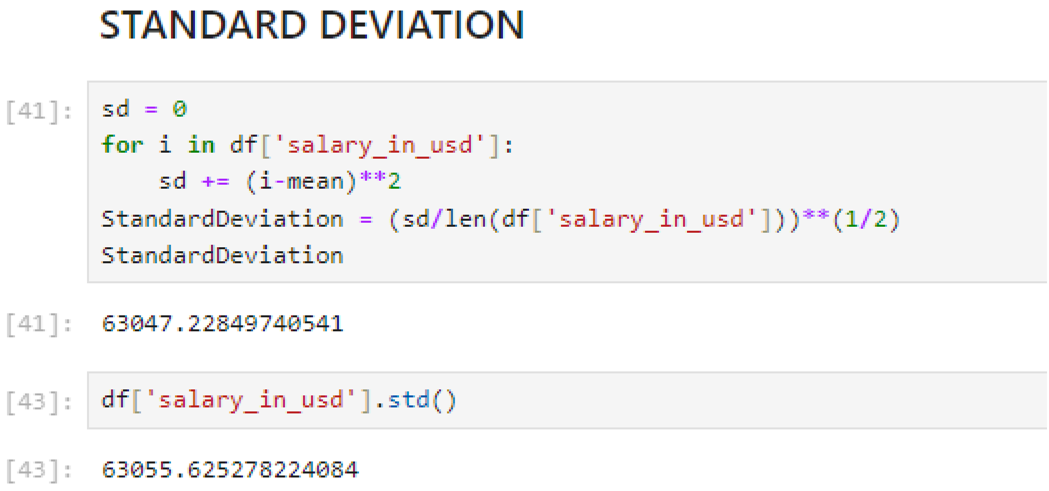


MEAN

```
[37]: mean = sumVal/len(salary_in_usd)
      mean

[37]: 137570.38988015978

[39]: df['salary_in_usd'].mean()

[39]: 137570.38988015978
```

Figure 11 Statistics of MEAN

3.1.3 Standard Deviation

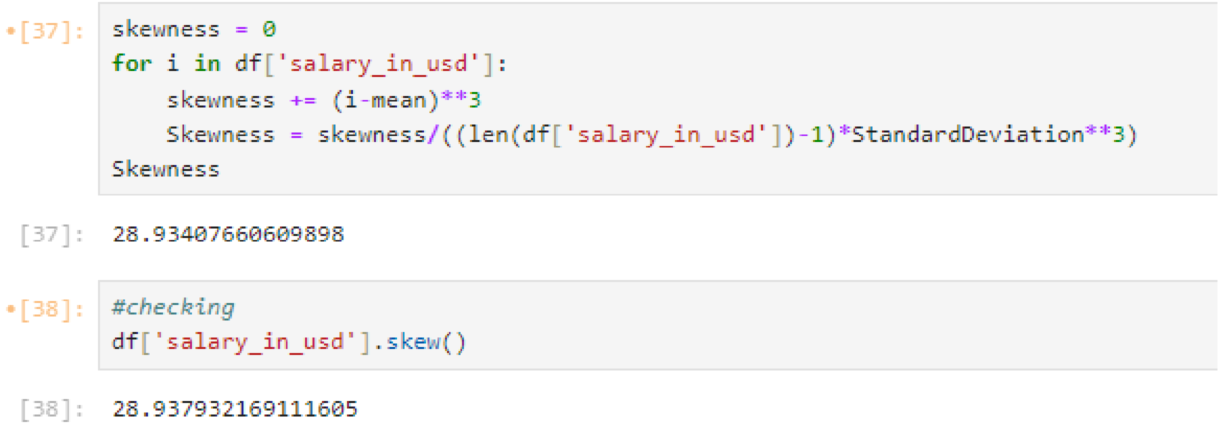


STANDARD DEVIATION

```
[41]: sd = 0
      for i in df['salary_in_usd']:
          sd += (i-mean)**2
      StandardDeviation = (sd/len(df['salary_in_usd']))**(1/2)
      StandardDeviation

[41]: 63047.22849740541

[43]: df['salary_in_usd'].std()

[43]: 63055.625278224084
```

Figure 12 Statistics of Standard Deviation

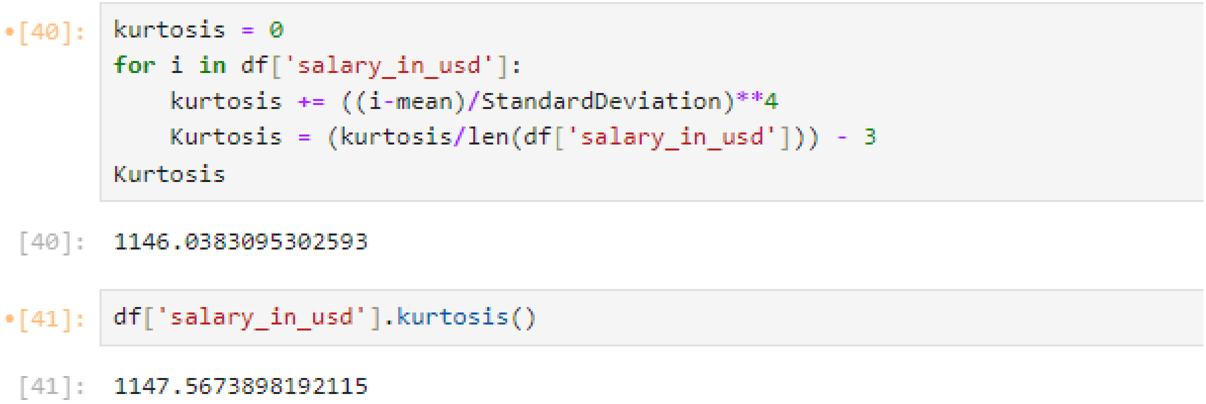
3.1.4 Skewness



```
[37]: skewness = 0
      for i in df['salary_in_usd']:
          skewness += (i-mean)**3
          Skewness = skewness/((len(df['salary_in_usd'])-1)*StandardDeviation**3)
      Skewness

[37]: 28.93407660609898

[38]: #checking
      df['salary_in_usd'].skew()

[38]: 28.937932169111605
```

Figure 13 Statistics for Skewness

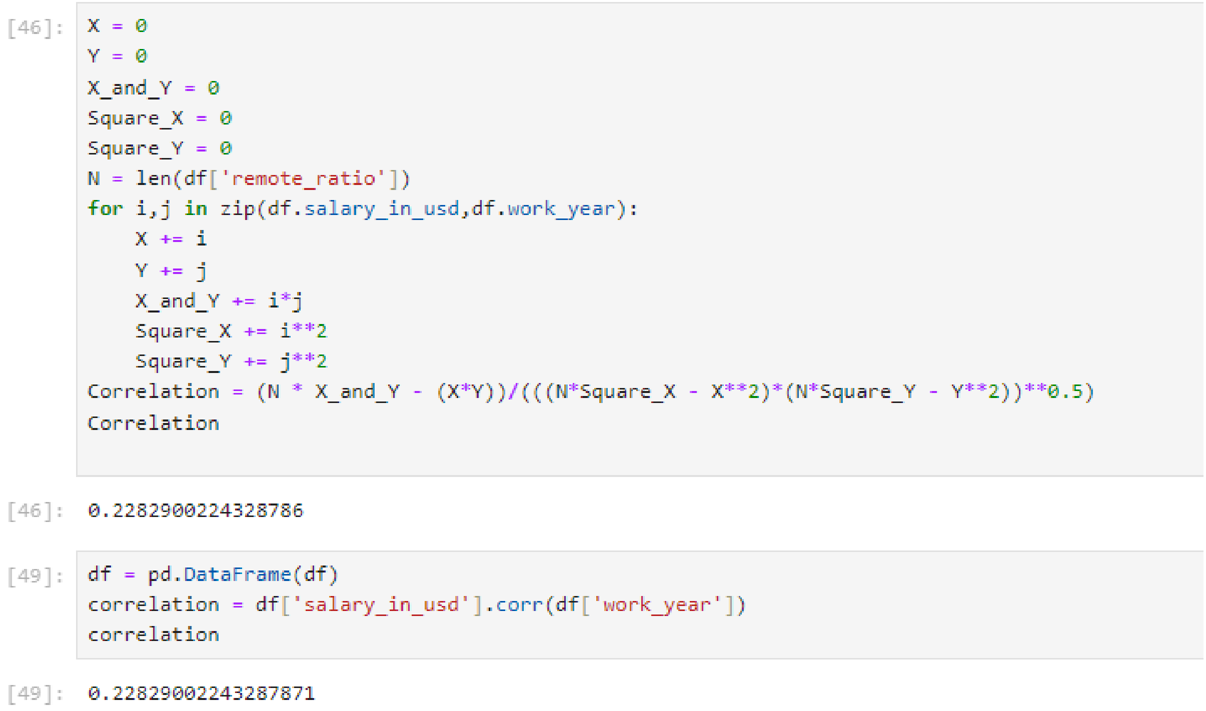
3.1.5 Kurtosis



```
[40]: kurtosis = 0
      for i in df['salary_in_usd']:
          kurtosis += ((i-mean)/StandardDeviation)**4
          Kurtosis = (kurtosis/len(df['salary_in_usd'])) - 3
      Kurtosis

[40]: 1146.0383095302593

[41]: df['salary_in_usd'].kurtosis()

[41]: 1147.5673898192115
```

Figure 14 Statistics for Kurtosis

3.2 Write a Python program to calculate and show correlation of all variables.



```
[46]: X = 0
      Y = 0
      X_and_Y = 0
      Square_X = 0
      Square_Y = 0
      N = len(df['remote_ratio'])
      for i,j in zip(df.salary_in_usd,df.work_year):
          X += i
          Y += j
          X_and_Y += i*j
          Square_X += i**2
          Square_Y += j**2
      Correlation = (N * X_and_Y - (X*Y))/(((N*Square_X - X**2)*(N*Square_Y - Y**2))**0.5)
      Correlation
```

```
[46]: 0.2282900224328786
```

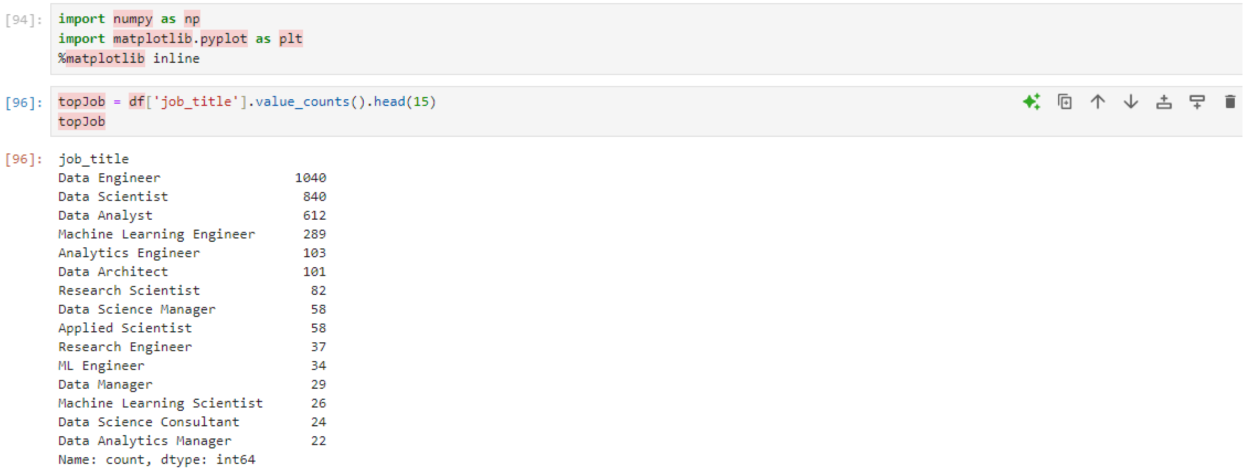
```
[49]: df = pd.DataFrame(df)
      correlation = df['salary_in_usd'].corr(df['work_year'])
      correlation
```

```
[49]: 0.22829002243287871
```

Figure 15 Python Program to calculate and show correlation of all the variables

4. Data Exploration

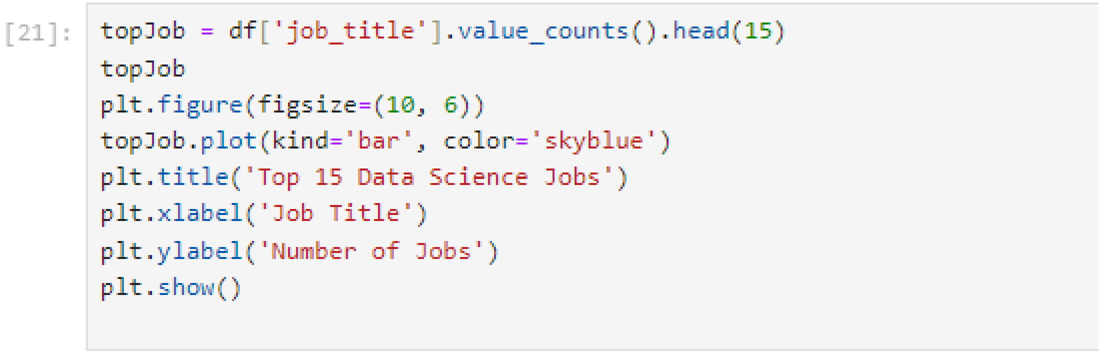
4.1 Write a python program to find out top 15 jobs. Make a bar graph of sales as well.



```
[94]: import numpy as np
      import matplotlib.pyplot as plt
      %matplotlib inline

[96]: topJob = df['job_title'].value_counts().head(15)
      topJob

[96]: job_title
      Data Engineer                 1040
      Data Scientist                 840
      Data Analyst                   612
      Machine Learning Engineer      289
      Analytics Engineer             103
      Data Architect                 101
      Research Scientist              82
      Data Science Manager            58
      Applied Scientist               58
      Research Engineer               37
      ML Engineer                     34
      Data Manager                    29
      Machine Learning Scientist      26
      Data Science Consultant         24
      Data Analytics Manager          22
      Name: count, dtype: int64
```

Figure 16 Top 15 jobs

The value_counts() method in pandas count the occurrences of each job title in the data frame. Head(15) displays the data frame with only the first 15 rows of the top 15 jobs.



```
[21]: topJob = df['job_title'].value_counts().head(15)
      topJob
      plt.figure(figsize=(10, 6))
      topJob.plot(kind='bar', color='skyblue')
      plt.title('Top 15 Data Science Jobs')
      plt.xlabel('Job Title')
      plt.ylabel('Number of Jobs')
      plt.show()
```

Figure 17 Python Program for selecting top 15 jobs

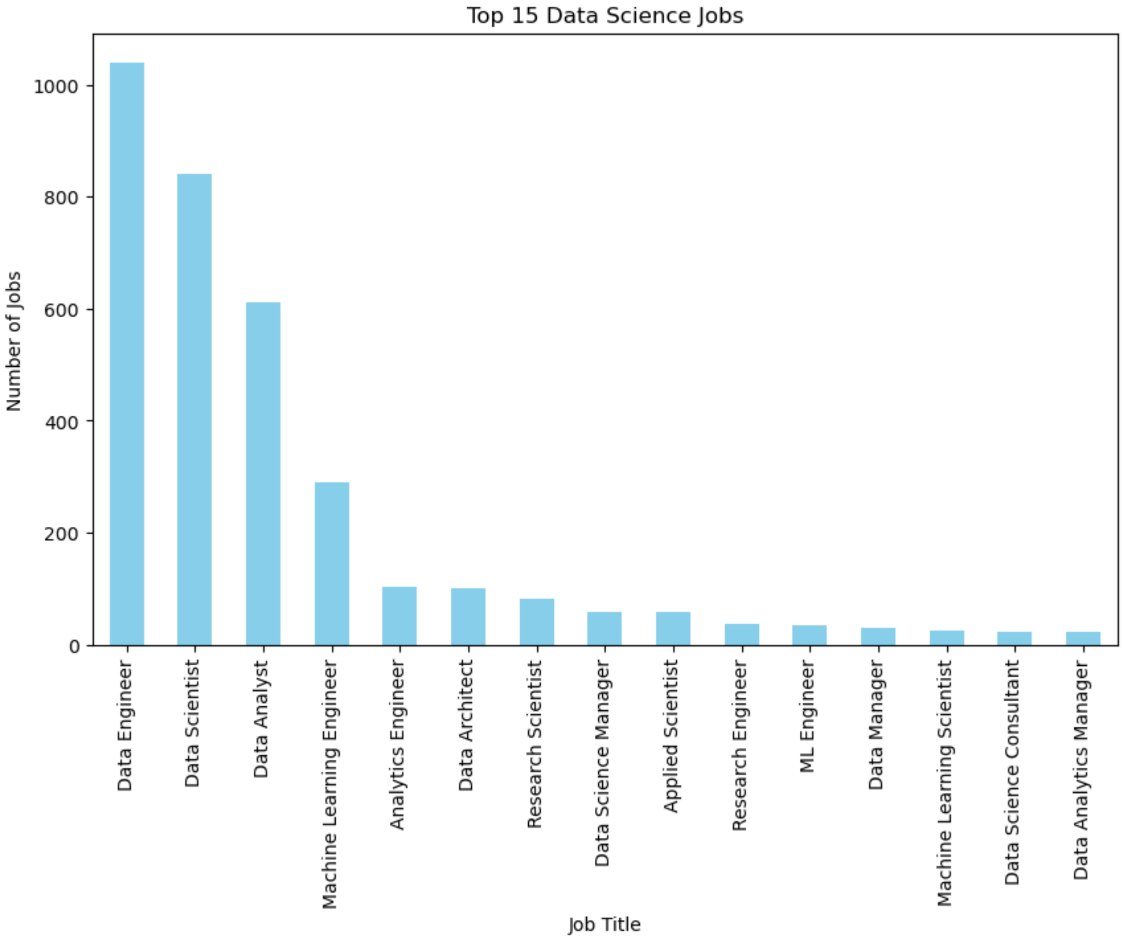
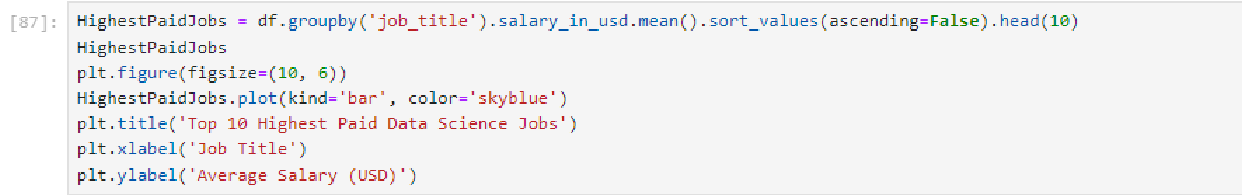


Figure 18 Graph of top 15 jobs

After importing matplotlib, plt.bar() function helps create a bar graph. Plt.title() function puts title on the top of the bar graph. Plt.xlabel() puts the title on the x-axis. Plt.ylabel() puts the label on the y-axis of the bar graph. Plt.xticks here rotates the tick labels by 45 degrees on the right. Finally, plt.show() illustrates the entire bar graph.

4.2 Which job has the highest salaries? Illustrate with bar graph.



```
[87]: HighestPaidJobs = df.groupby('job_title').salary_in_usd.mean().sort_values(ascending=False).head(10)
      HighestPaidJobs
      plt.figure(figsize=(10, 6))
      HighestPaidJobs.plot(kind='bar', color='skyblue')
      plt.title('Top 10 Highest Paid Data Science Jobs')
      plt.xlabel('Job Title')
      plt.ylabel('Average Salary (USD)')
```

Figure 19 Python Program for selecting jobs with highest salaries

The dataframe is grouped by the employees' job_title and their salary_in_usd. The mean() function results out the mean of the job along with their salaries. Both the title and their mean salaries are then listed below.

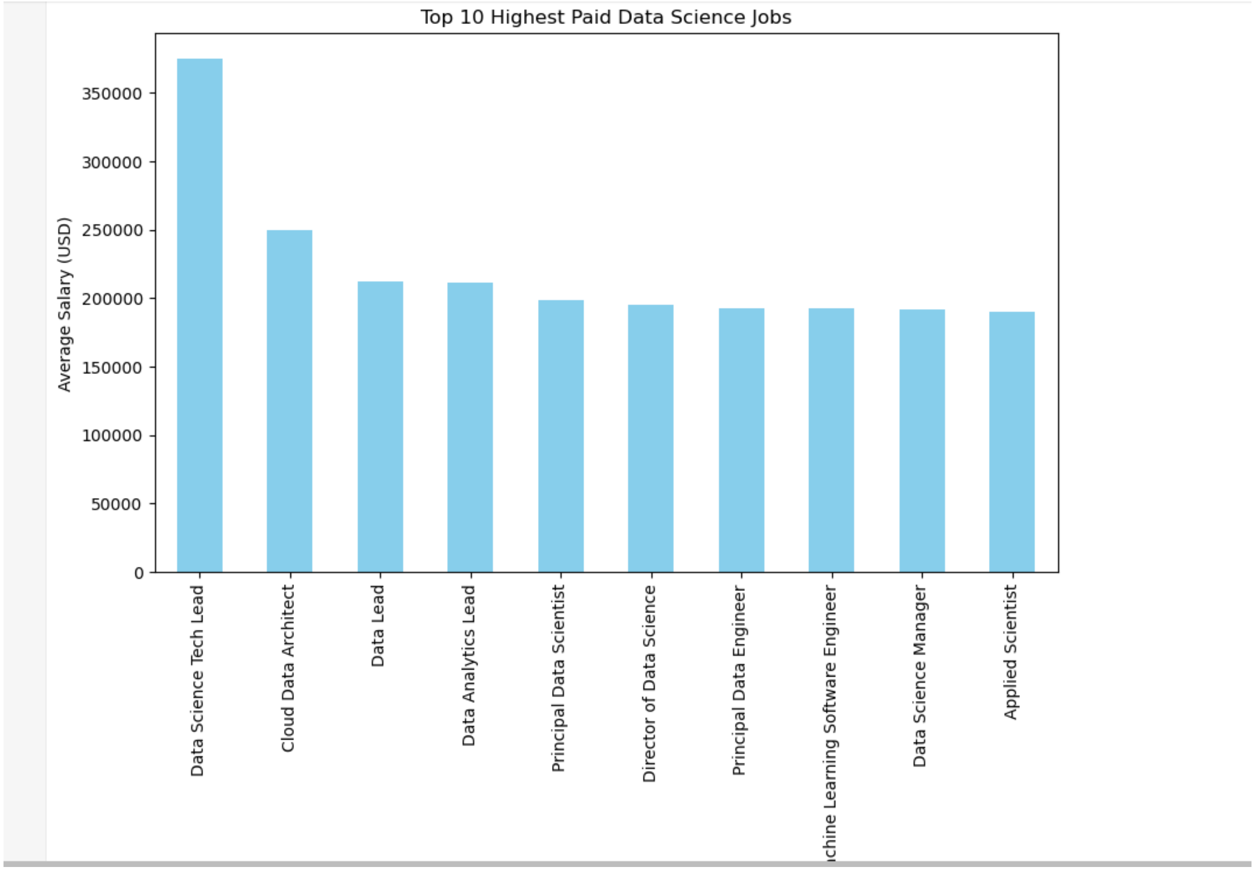
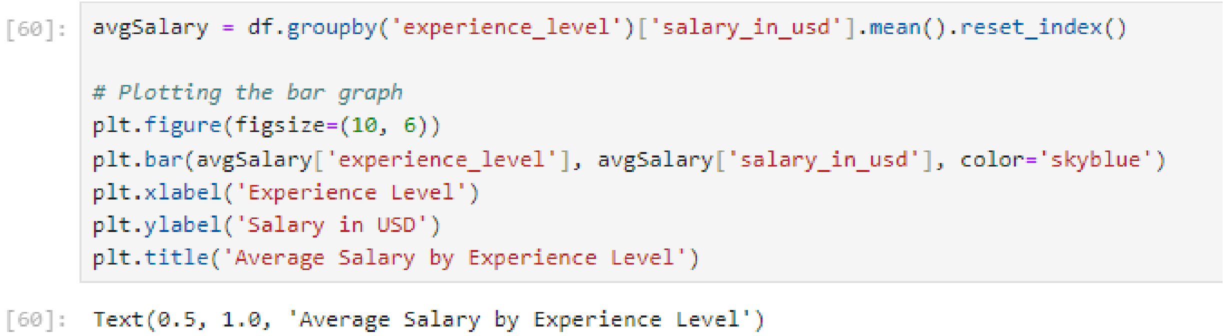


Figure 20 Graph of Top 10 highest paid data science job

The bar graph is titled Job with highest salary with x-axis label as Job Title and y-axis title named as Average Salary(USD). The ticks is rotated 90 degrees to avoid texts over lapping on each other. Finally, the bar graph is displayed with plt.show().

4.3 Write a python program to fins out salaries based on experience level. Illustrate it through bar graph.



```
[60]: avgSalary = df.groupby('experience_level')['salary_in_usd'].mean().reset_index()

      # Plotting the bar graph
      plt.figure(figsize=(10, 6))
      plt.bar(avgSalary['experience_level'], avgSalary['salary_in_usd'], color='skyblue')
      plt.xlabel('Experience Level')
      plt.ylabel('Salary in USD')
      plt.title('Average Salary by Experience Level')

[60]: Text(0.5, 1.0, 'Average Salary by Experience Level')
```

Figure 21 python program to find out salaries based on experience level

Here, in this code, experience_level and salary is grouped using groupby and the mean of the salary is also calculated. Then the experience level along with the mean of salary is listed below.

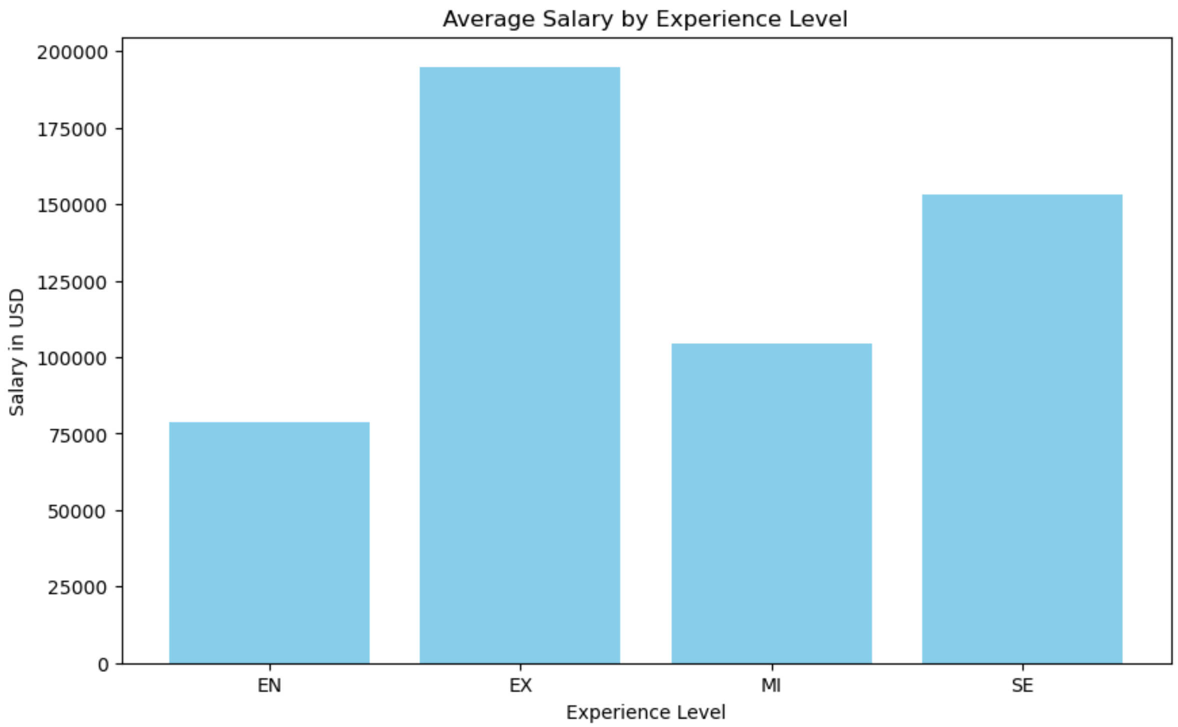
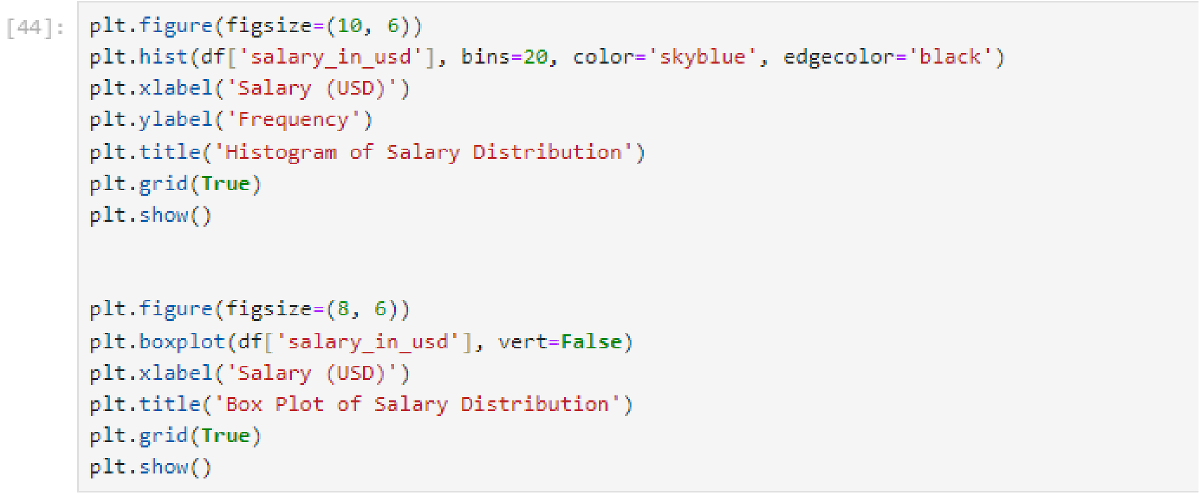


Figure 22 Graph of Average salary by experience level

The plt.bar() allows to form bar graph with ES as both index and value. Plt.title() adss title on the top of the bar graph. Plt.xlabel labels a title on the xx-axis as Experience level. The ticks on the x-axis are tilted 45 degrees on right. Plt.ylabel() puts a title on the y-axis as Average Salary. Finally plt.show() finally visualises the bar graph with the information.

4.4 Write a Python program to show histogram and Box plot of any chosen different variables. Use proper labels in the graph.

A histogram is a graphical representation of data points organized into user-specified ranges. Similar in appearance to a bar graph, the histogram condenses a data series into an easily interpreted visual by taking many data points and grouping them into logical ranges or bins (Chen, 2024).



```
[44]: plt.figure(figsize=(10, 6))
      plt.hist(df['salary_in_usd'], bins=20, color='skyblue', edgecolor='black')
      plt.xlabel('Salary (USD)')
      plt.ylabel('Frequency')
      plt.title('Histogram of Salary Distribution')
      plt.grid(True)
      plt.show()


      plt.figure(figsize=(8, 6))
      plt.boxplot(df['salary_in_usd'], vert=False)
      plt.xlabel('Salary (USD)')
      plt.title('Box Plot of Salary Distribution')
      plt.grid(True)
      plt.show()
```

Figure 23 Python program to show histogram and box plot of salary_in_usd

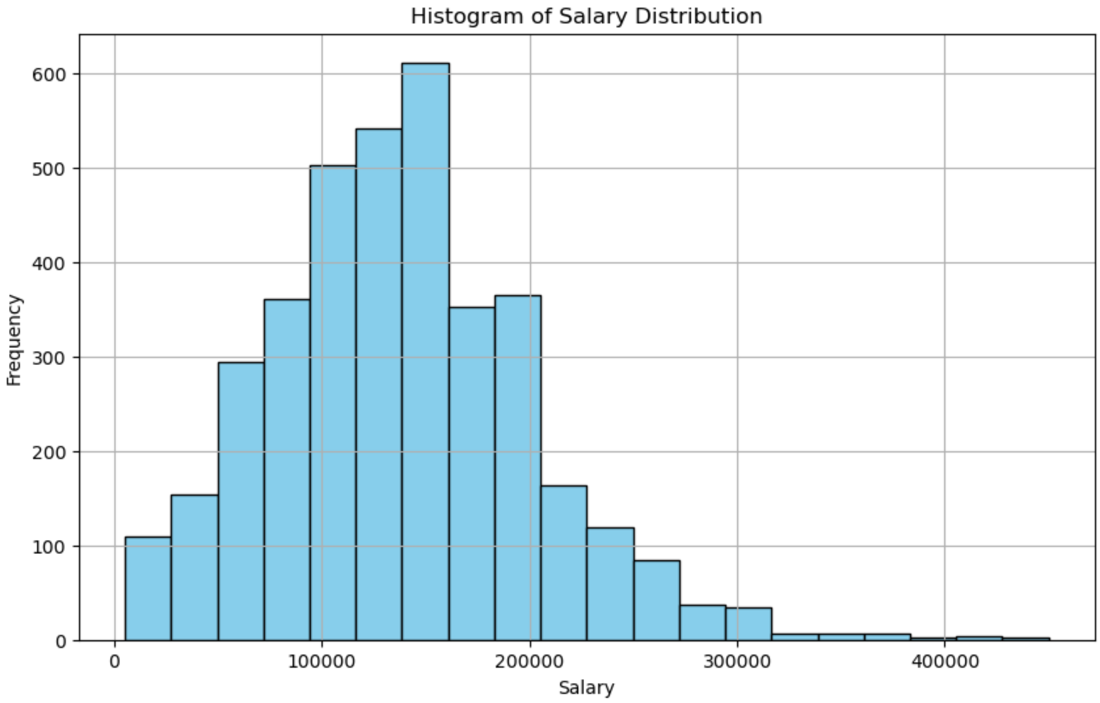


Figure 24 Histogram of Salary distribution

In the above figure, the histogram of 'salary_in_usd' column is created of the df dataset. The edge colour is set to black for the border among the conjoined bars.

The title of the histogram is History of salary_in_usd while x-axis is salary_in_usd and in y-axis as Frequency. The frequency represents the occurrence of data. Finally, plt.show() displays the histogram on the screen.

A box and whisker plot—also called a box plot—displays the five-number summary of a set of data. The five-number summary is the minimum, first quartile, median, third quartile, and maximum (Academy, 2024).

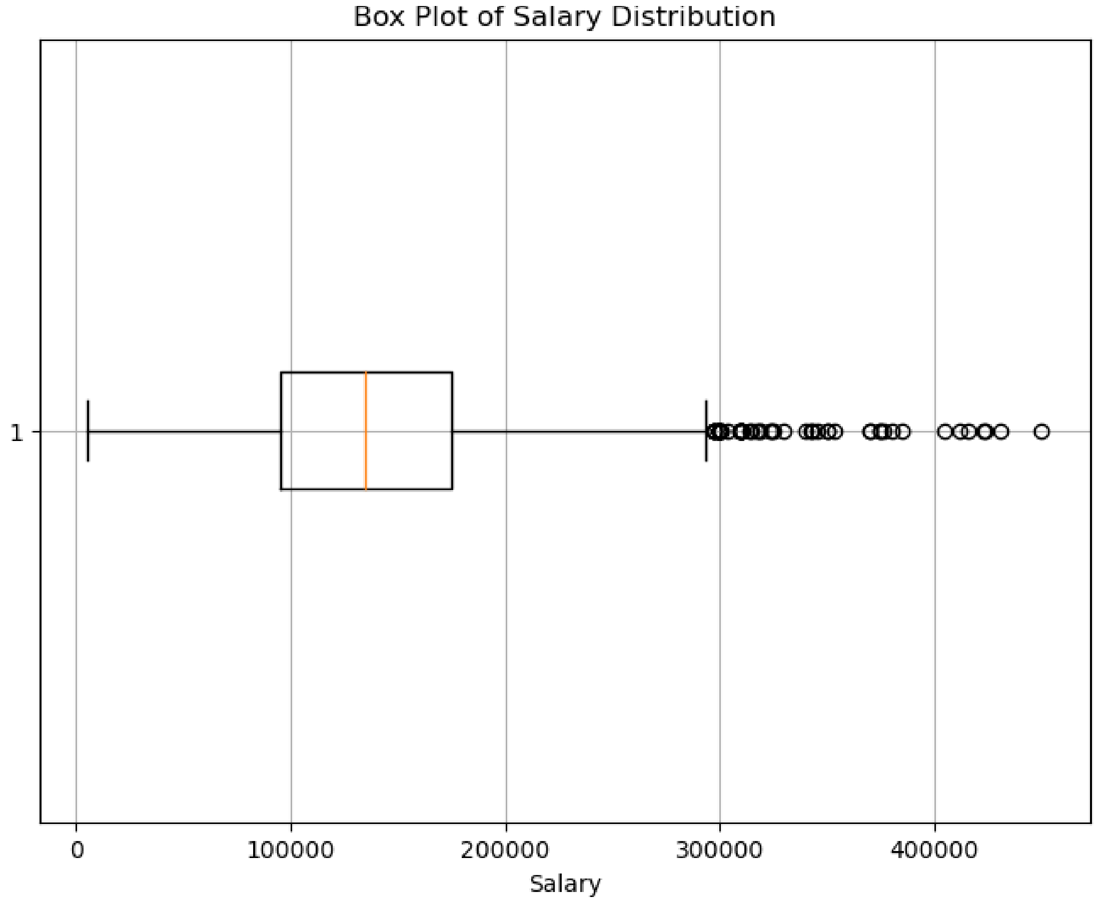


Figure 25 Box Plot of Salary Distribution

5. Conclusion

We were able to learn more about data processing, analysis, and exploration, respectively, thanks to this individual coursework. Understanding, analyzing, and exploring data about workers in the data science field and their pay was part of the training. It was necessary for us to read the CVS file into a pandas data frame, eliminate NaN values, look for unique values, and determine whether the variables were correlated. Additionally, the computation of data in the forms of mean, total, standard deviation, skewness, and kurtosis when selecting any variables was necessary. In addition, a bar graph representing the top 15 jobs, the highest paying job, experience-level-based pay, a histogram, and a box plot of any selected variables were displayed.

We had several challenges finishing our coursework, but the lecturers and tutorial teachers gave us the proper guidance and support. In order to solve an issue or in case I needed assistance finishing my coursework, I also read a few websites. While coding was fun, it was also extremely difficult. My enthusiasm for cracking the codes increased when I discovered how little I actually understood about Python as a programming language. We learned more about data interpretation, analysis, processing, and exploration thanks to this module.

In conclusion, the coursework provided us with the chance to analyze and resolve real-world issues. The knowledge and skills I've gained from completing this coursework will definitely come in handy for my future .

6. Reference

Academy, K. (2024, may 10). *Khan Academy*. Retrieved from khan Academy: https://www.khanacademy.org/math/statistics-probability/summarizing-quantitative-data/box-whisker-plots/a/box-plot-review

AWS. (2024, May 10). *What is Data Preparation?* Retrieved from AWS: https://aws.amazon.com/what-is/data-preparation/

Chen, J. (2024, may 10). *investopedia*. Retrieved from How a Histogram Works to Display Data: https://www.investopedia.com/terms/h/histogram.asp

heavyAI. (2024, MAY 12). *Data Exploration*. Retrieved from HeavyAI: https://www.heavy.ai/learn/data-exploration

IBM. (2024, 05 12). *Data Understanding Overview*. Retrieved from IBM: https://www.ibm.com/docs/en/spss-modeler/saas?topic=understanding-data-overview

IBM. (2024, May 10). *What is Data Visualization?* Retrieved from IBM: https://www.ibm.com/topics/data-visualization

Nehma, A. (2024, may 10). *What is Data Analysis?* Retrieved from datacamp: https://www.datacamp.com/blog/what-is-data-analysis-expert-guide?utm_source=google&utm_medium=paid_search&utm_campaignid=19589720824&utm_adgroupid=152984013294&utm_device=c&utm_keyword=&utm_matchtype=&utm_network=g&utm_adpostion=&utm_creative=698229374851