



CC5067NI-Smart Data Discovery

60% Individual Coursework

2023-24 Spring

Student Name: Sumit Shrestha

London Met ID: 22085637

College ID: np01cp4s230046

Assignment Due Date: Monday, May 13, 2024

Assignment Submission Date: Sunday, May 12, 2024

Word Count: 2232

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 Content

1. Introduction	1
2. Data Preparation.....	4
2.1 Write a Python program to load data into Pandas Data Frame.	4
2.2 Write a Python program to remove unnecessary columns i.e., salary and salary currency.	5
2.3 Write a python program to remove the NaN missing values from updated dataframe.	6
2.4 Write a python program to check duplicate values in the data frame.....	7
2.5 Write a python program to see the unique values from all the columns in the dataframe.	8
2.6 Rename the experience level columns as below.	8
SE – Senior Level/Expert.....	8
MI – Medium Level/Intermediate.....	8
EN – Entry Level.....	8
3. Data Analysis	9
3.1 Write a Python program to show summary statistics of sum, mean, standard deviation, skewness, and kurtosis of any chosen variable.....	9
3.2 Write a Python program to calculate and show correlation of all variables.	10
4. Data Exploration	11
4.1 Write a python program to find out top 15 jobs. Make a bar graph of sales as well.	11
4.2 Which job has the highest salaries? Illustrate with bar graph.	11
4.3 Write a python program to find out salaries based on experience level. Illustrate it through bar graph.	12
4.4 Write a Python program to show histogram and box plot of any chosen different variables. Use proper labels in the graph.	13

Table of Figures

Figure 1 Importing Library	5
Figure 2Removing unnecessary columns.....	6
Figure 3removing NaN value.....	6
Figure 4Check Duplicate	7
Figure 5Program to see the unique values.....	8
Figure 6Renaming the names of experience level columns	9
Figure 7Python program to show summary statistics of sum, mean, standard deviation, skewness, and kurtosis of any chosen variable.	10
Figure 8Calculate and show correlation of all variables.	10
Figure 9Python program to find out top 15 jobs.....	11
Figure 10Job has the highest salaries.....	12
Figure 11Python program to find out salaries based on experience level.....	13
Figure 12Python program to show histogram.....	14

Table Of Content

Table 1Data Understanding 3

1. Introduction

Data science is the methods and techniques for extracting useful information and knowledge from data for personal growth or business growth. It is a multidisciplinary approach that combines principles and practices from the fields of mathematics, statistics, artificial intelligence, and computer engineering to analyze large amounts of data. This analysis helps data scientists to ask and answer questions like what happened, why it happened, what will happen, and what can be done with the results.

Data science is important because it combines tools, methods, and technology to generate meaning from data. Modern organizations are inundated with data; there is a proliferation of devices that can automatically collect and store information. Online systems and payment portals capture more data in the fields of e-commerce, medicine, finance, and every other aspect of human life. We have text, audio, video, and image data available in vast quantities.

The given CSV file is the data of the salaries of data scientist whole world. This assignment is assigned to year 2, second semester. Data science has become increasingly important in today's data-driven world, which is defined by an abundance of digital data coming from various sources, including social media, online transactions, sensor networks, and Internet of Things devices. Data science is used by modern companies in a wide range of industries, including e-commerce, healthcare, finance, and more, to get actionable insights, streamline operations, and gain a competitive edge in a market that is becoming more and more crowded.

This assignment is evidence of the increasing importance of data science in both academia and industry. It provides a hands-on opportunity for second-year students in their second semester to apply fundamental data science principles to real-world datasets. The complexities of data preparation, analysis, and visualization are navigated by students, who obtain vital expertise in utilizing.

2. Data Understanding

Data understanding is the fundamental step in our analysis, providing insights into the structure, quality, and characteristics of the dataset. This phase encompasses data collection, exploration, cleaning, and preprocessing setting the stage for subsequent analysis and modeling. The most important activities of data understanding are to identify potential data sources (transactional databases, spreadsheets, CSV, text files, web logs, web services, etc.). Data collection efforts focused on gathering information from multiple sources relevant to the project objectives. The first step in data understanding involves identifying and assessing potential data sources. Data collection efforts are focused on gathering information from multiple sources relevant to the project objectives. This involves systematically retrieving data from identified sources and consolidating it into a unified dataset for analysis. Once the data is collected, thorough exploration is conducted to understand its underlying structure and properties. Exploring data analysis techniques to understand its underlying structure and properties.

Exploratory data analysis techniques such as summary statistics, data visualization, and correlation analysis are employed to uncover patterns, trends, and potential issues within the dataset. Data cleaning and preprocessing are essential steps to address inconsistencies, errors, and missing values in the dataset. Techniques such as handling missing data, removing duplicates, correcting errors, and transforming variables are applied to ensure data integrity and quality (ScienceDirect, 2016). The main goal of data understanding is to gain general insights about the data that will potentially be helpful for further steps in the data analysis process, but data understanding should not be driven exclusively by the goals and methods to be applied in later steps.

Although these requirements should be kept in mind during data understanding, one should approach the data from a neutral point of view. Never trust any data as long as you have not carried out some simple plausibility checks. Methods for such plausibility checks will be discussed in this chapter. At the end of the data understanding phase, we know much better whether the assumptions we made during the project understanding phase concerning representativeness, informativeness, data quality, and the presence or absence of external factors are justified (Berthold, 2018).

SN	Column Name	Description	Data Type
1	work_year	Year of individual got enrolled.	Integer
2	experience_level	Level of experience an individual has.	String
3	employment_type	Duration of job an individual doing.	String
4	job_title	Type of job of an individual.	String
5	salary	Amount individual getting paid.	Integer
6	Salary_currency	Currency individual getting paid.	String
7	Salary_in_USD	US dollar Currency individual getting paid.	Integer
8	employee_residency	The location of the individual.	String
9	remote_ratio	Remote working ratio of an individual.	Integer
10	company_location	Location of the company an individual is working for.	String

Table 1 Data Understanding

3. Data Preparation

Data preparation is a critical phase in the data science workflow, where raw data is transformed, cleaned, and structured to make it suitable for analysis and modeling. This phase encompasses a series of activities aimed at ensuring that the data is accurate, complete, and formatted appropriately for the chosen analysis techniques. Effective data preparation lays the foundation for meaningful insights and accurate predictions. The most important activities of data preparation are preparing an analytics sandbox, a central repository environment separate from the production environment. (Data warehouse, data lake, and big data platform).

The repository should collect all kinds of data (summary-level aggregated data, structured data, raw data feeds, and unstructured text data from call logs or weblogs. To get data into the sandbox by performing a combination of extract, transform, and load activities. Learning or understanding the data to clarify what data is accessible Highlight gaps by identifying datasets that are useful but not accessible. Identify external datasets that might be useful to obtain through APIs, data sharing, or purchasing.

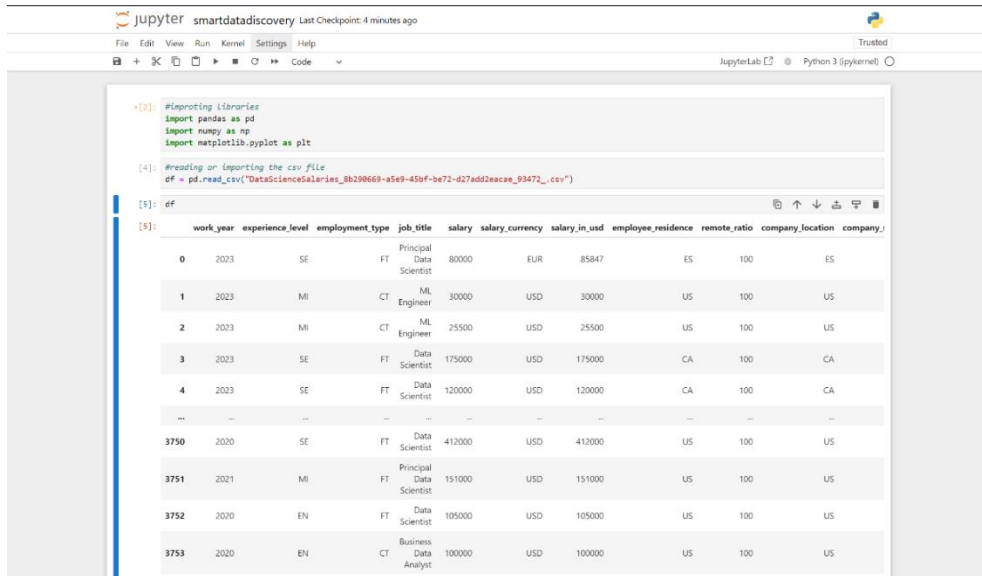
3.1 Write a Python program to load data into Pandas Data Frame.

Pandas Data Frame is a common task in data analysis and manipulation using Python. Panda is a powerful library that provides data structures and functions for working with structured data, such as tabular data, intuitively and efficiently.

```
import pandas as pd
```

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```

* [3]: #Importing Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

[4]: #Reading or Importing the csv file
df = pd.read_csv("DataScienceSalaries_Bh298669-a5e9-45bf-b672-d27add2eacae_93472_.csv")

[5]: df

```

	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	company_size
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	

Figure 1 Importing Library

3.2 Write a Python program to remove unnecessary columns i.e., salary and salary currency.

It is frequently required to remove unnecessary features from the dataset that have minimal bearing on the main study to achieve analytical clarity and efficiency. Analysts reduce the number of duplicate or unnecessary columns in the dataset such as salary and salary currency which reduces the possibility of dimensionality-related

complications and improves the readability of further analysis.

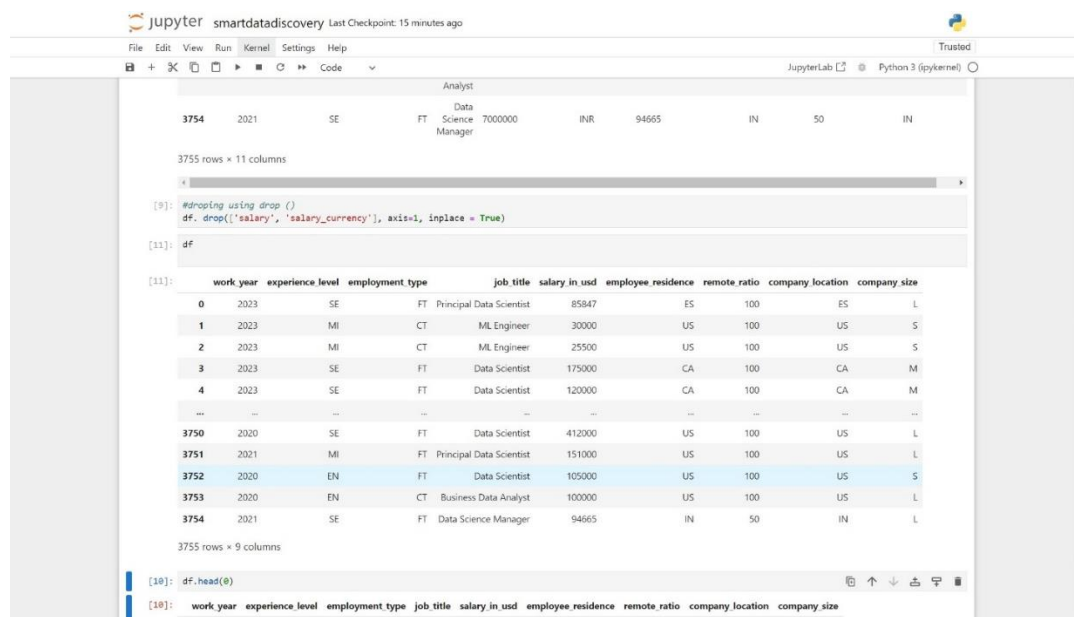


Figure 2 Removing unnecessary columns.

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

The existence of missing values, indicated as NaN (Not a Number), significantly compromises the dataset's dependability and integrity. To address this problem, analysts utilize methods to detect and remove NaN values, guaranteeing the accuracy and consistency of the dataset. Analysts can increase the accuracy and usefulness of their conclusions by carefully managing missing data, which strengthens the validity of later analyses.

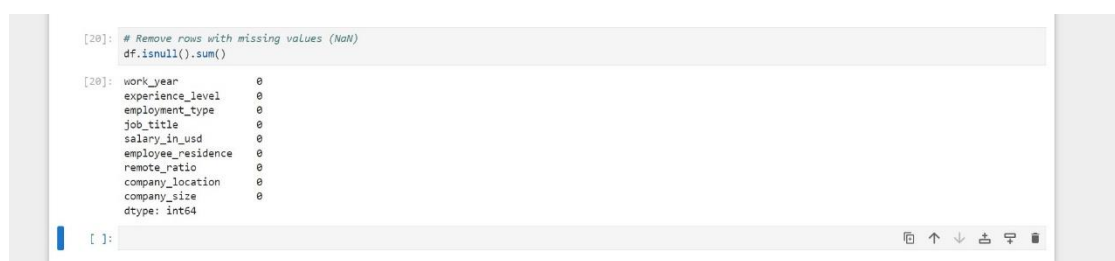
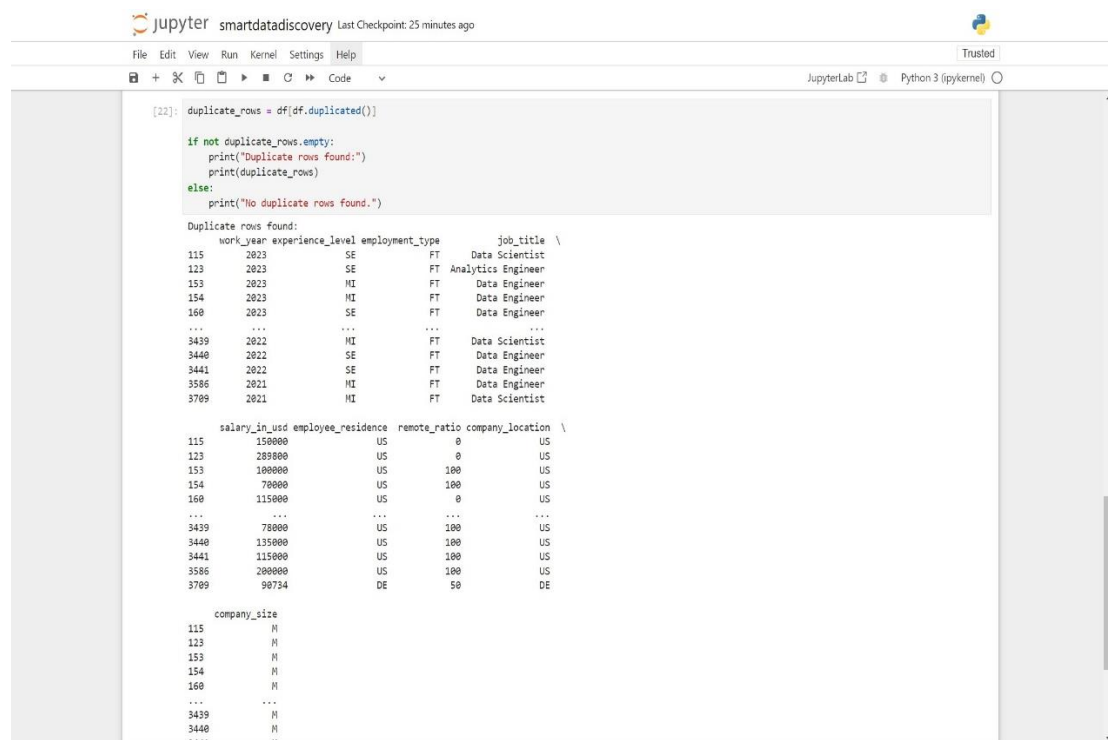


Figure 3 removing NaN value

3.4 Write a python program to check duplicate values in the data frame.

Within the dataset, duplicate items have the potential to distort analytical findings, resulting in incorrect conclusions and poor decision-making. To prevent this problem in the first place, analysts put processes in place to identify and eliminate duplicate values, protecting the integrity of the dataset and guaranteeing the accuracy of ensuing studies. Analysts create a more accurate and representative representation of the underlying data distribution by removing entries that are duplicates.



```
[22]: duplicate_rows = df[df.duplicated()]

if not duplicate_rows.empty:
    print("Duplicate rows found:")
    print(duplicate_rows)
else:
    print("No duplicate rows found.")
```

Duplicate rows found:

	work_year	experience_level	employment_type	job_title
115	2023	SE	FT	Data Scientist
123	2023	SE	FT	Analytics Engineer
153	2023	MI	FT	Data Engineer
154	2023	MI	FT	Data Engineer
160	2023	SE	FT	Data Engineer
...
3439	2022	MI	FT	Data Scientist
3440	2022	SE	FT	Data Engineer
3441	2022	SE	FT	Data Engineer
3586	2021	MI	FT	Data Engineer
3709	2021	MI	FT	Data Scientist

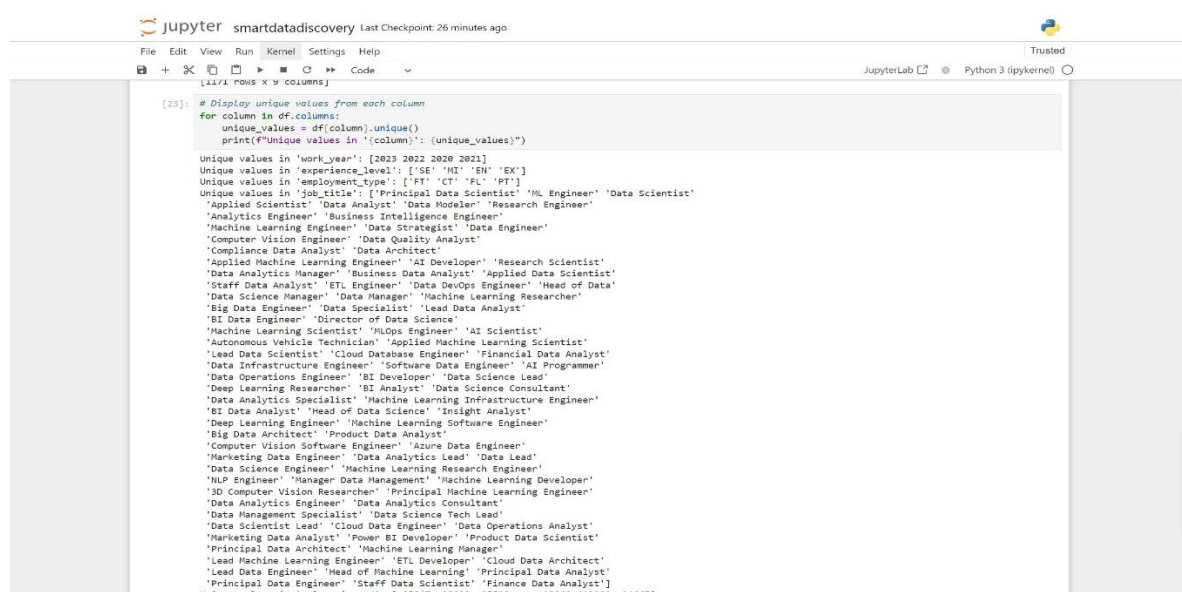
	salary_in_usd	employee_residence	remote_ratio	company_location
115	150000	US	0	US
123	289000	US	0	US
153	100000	US	100	US
154	70000	US	100	US
160	115000	US	0	US
...
3439	78000	US	100	US
3440	135000	US	100	US
3441	115000	US	100	US
3586	200000	US	100	US
3709	90734	DE	50	DE

	company_size
115	M
123	M
153	M
154	M
160	M
...	...
3439	M
3440	M
...	...

Figure 4Check Duplicate

2.5 Write a Python program to see the unique values from all the columns in the data frame.

Finding distinct values in each dataset column is a fundamental first step in exploring and comprehending the data. Analysts can obtain insights into the categorical distribution of the dataset by counting the unique values in each column. This allows them to detect both common categories and potential abnormalities. This makes it easier to comprehend the fundamental structure of the dataset and opens the door to more complex analysis and interpretations.



```

jupyter smartdatadiscovery Last Checkpoint: 26 minutes ago
File Edit View Run Kernel Settings Help
[12/1] POWS X Y columns
JupyterLab Python 3 (ipykernel)

[25]: # Display unique values from each column
for column in df.columns:
    unique_values = df[column].unique()
    print(f"Unique values in '{column}': {unique_values}")

Unique values in 'work_year': [2023 2022 2020 2021]
Unique values in 'experience_level': ['SE' 'MI' 'EN' 'EX']
Unique values in 'employment_type': ['FT' 'CT' 'FL' 'PT']
Unique values in 'job_title': ['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' 'ML Ops 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'
'MLP 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'
'Marketing Data Analyst' 'Power BI Developer' 'Product Data Scientist'
'Principal Data Architect' 'Machine Learning Manager'
'Lead Machine Learning Engineer' 'ETL Developer' 'Cloud Data Architect'
'Lead Data Engineer' 'Head of Machine Learning' 'Principal Data Analyst'
'Principal Data Engineer' 'Staff Data Scientist' 'Finance Data Analyst']

```

Figure 5 Program to see the unique values

2.6 Rename the experience level columns as below.

SE – Senior Level/Expert

MI – Medium Level/Intermediate

EN – Entry Level

EX – Executive Level

The semantics and importance of each characteristic in the dataset are clarified in large part by the column names. Analysts improve the accessibility and interpretability of the dataset by standardizing and elucidating column names. This promotes easy

communication and information sharing within multidisciplinary teams. Furthermore, analysts can more effectively communicate domain-specific information by renaming columns, which promotes a more nuanced comprehension of the dataset's complexities.

```

unique values in 'company_size': ['L' 'S' 'M']

[26]: #Rename the experience level columns as below. SE - Senior Level/Expert MI - Medium Level/Intermediate EN - Entry Level EX - Executive Level
new_experience_level_mapping = {
    "SE": "Senior Level/Expert",
    "MI": "Medium Level/Intermediate",
    "EN": "Entry Level",
    "EX": "Executive Level"
}

df["experience_level"] = df["experience_level"].replace(new_experience_level_mapping)
df.head()

[26]:
work_year  experience_level  employment_type  job_title  salary_in_usd  employee_residence  remote_ratio  company_location  company_size
0      2023      Senior Level/Expert          FT  Principal Data Scientist      85847              ES          100              ES          L
1      2023  Medium Level/Intermediate          CT      ML Engineer      30000              US          100              US          S
2      2023  Medium Level/Intermediate          CT      ML Engineer      25500              US          100              US          S
3      2023      Senior Level/Expert          FT      Data Scientist     175000              CA          100              CA          M
4      2023      Senior Level/Expert          FT      Data Scientist     120000              CA          100              CA          M

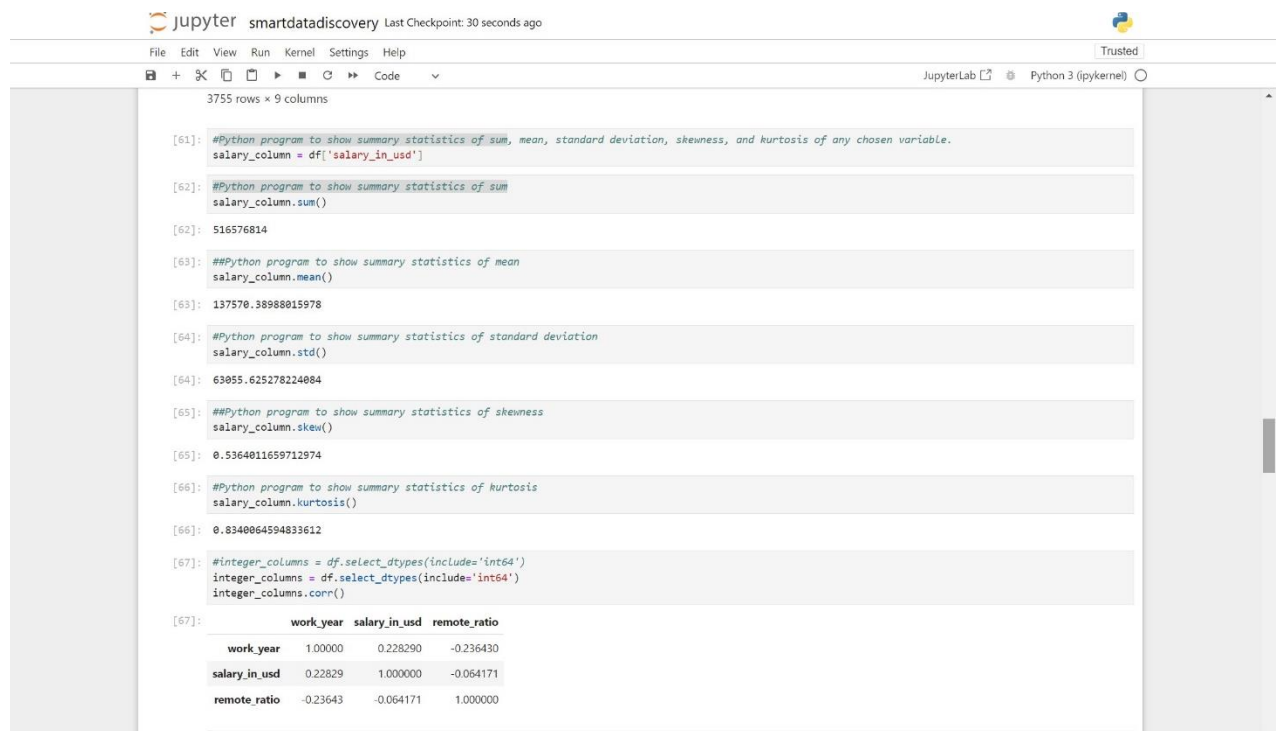
```

Figure 6 Renaming the names of experience-level columns

4. Data Analysis

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

A broad overview of the distributional properties of the dataset is provided by summary statistics, which shed light on shape, dispersion, and central tendencies. Analysts can identify trends, spot outliers, and create hypotheses for more research by calculating summary statistics like sum, mean, standard deviation, skewness, and kurtosis for selected variables. This gives them a thorough understanding of the dataset's underlying distributional properties.



```

[61]: #Python program to show summary statistics of sum, mean, standard deviation, skewness, and kurtosis of any chosen variable.
salary_column = df['salary_in_usd']

[62]: #Python program to show summary statistics of sum
salary_column.sum()

[62]: 516576814

[63]: ##Python program to show summary statistics of mean
salary_column.mean()

[63]: 137578.38988015978

[64]: #Python program to show summary statistics of standard deviation
salary_column.std()

[64]: 63055.625278224084

[65]: ##Python program to show summary statistics of skewness
salary_column.skew()

[65]: 0.5364011659712974

[66]: #Python program to show summary statistics of kurtosis
salary_column.kurtosis()

[66]: 0.8340064594833612

[67]: #integer_columns = df.select_dtypes(include='int64')
integer_columns = df.select_dtypes(include='int64')
integer_columns.corr()

[67]:
      work_year  salary_in_usd  remote_ratio
work_year      1.000000      0.228290      -0.236430
salary_in_usd    0.22829      1.000000      -0.064171
remote_ratio    -0.23643     -0.064171      1.000000

```

Figure 7 Python program to show summary statistics of sum, mean, standard deviation, skewness, and kurtosis of any chosen variable.

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

The foundation of exploratory data analysis is correlation analysis, which helps analysts find connections and dependencies among the variables in the dataset. Analysts detect patterns of association by calculating correlation coefficients between all variables and identifying pairings of variables that show co-occurrence or reciprocal influence. This makes it easier to find predictive associations and provides guidance for later modeling efforts that result in data-driven strategies and interventions.



```

[89]: #Python program to calculate and show the correlation of all variables.
integer_columns = df.select_dtypes(include='int64')
integer_columns = df.select_dtypes(include='int64')
integer_columns.corr()

[89]:
      work_year  salary_in_usd  remote_ratio
work_year      1.000000      0.228290      -0.236430
salary_in_usd    0.22829      1.000000      -0.064171
remote_ratio    -0.23643     -0.064171      1.000000

[90]: topJobs = df["job_title"].value_counts().head(15)

```

Figure 8 Calculate and show correlation of all variables.

5. Data Exploration

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

The dataset's top 15 jobs are identified, offering an overview of the occupational landscape and illuminating popular job categories and employment patterns. Through the use of a bar graph to illustrate the distribution of job categories, analysts are able to determine the occupational makeup of the dataset and identify the most common job titles along with their respective frequencies.

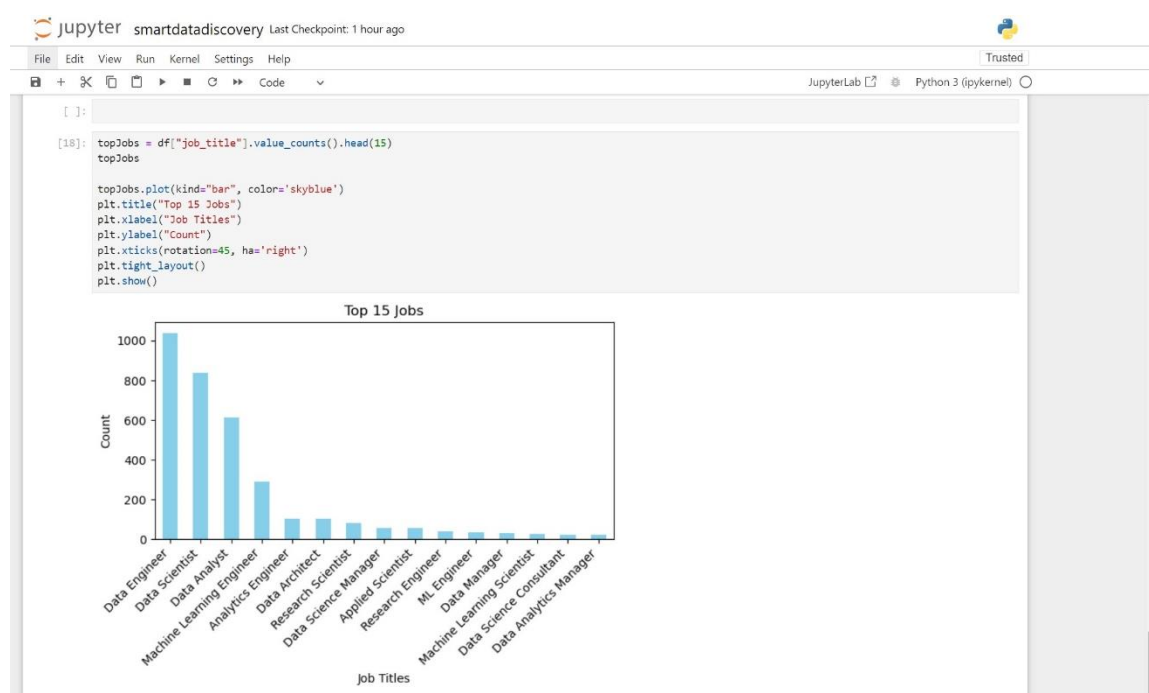


Figure 9 Python program to find out top 15 jobs

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

Finding the highest paid job in the dataset provides an insight into the compensation landscape by highlighting high-paying jobs and in-demand skill sets. Analysts discover outliers and abnormalities in the wage distributions across various job titles by displaying the data as a bar graph, which helps them uncover highly sought-after occupations that come with premium compensation packages.

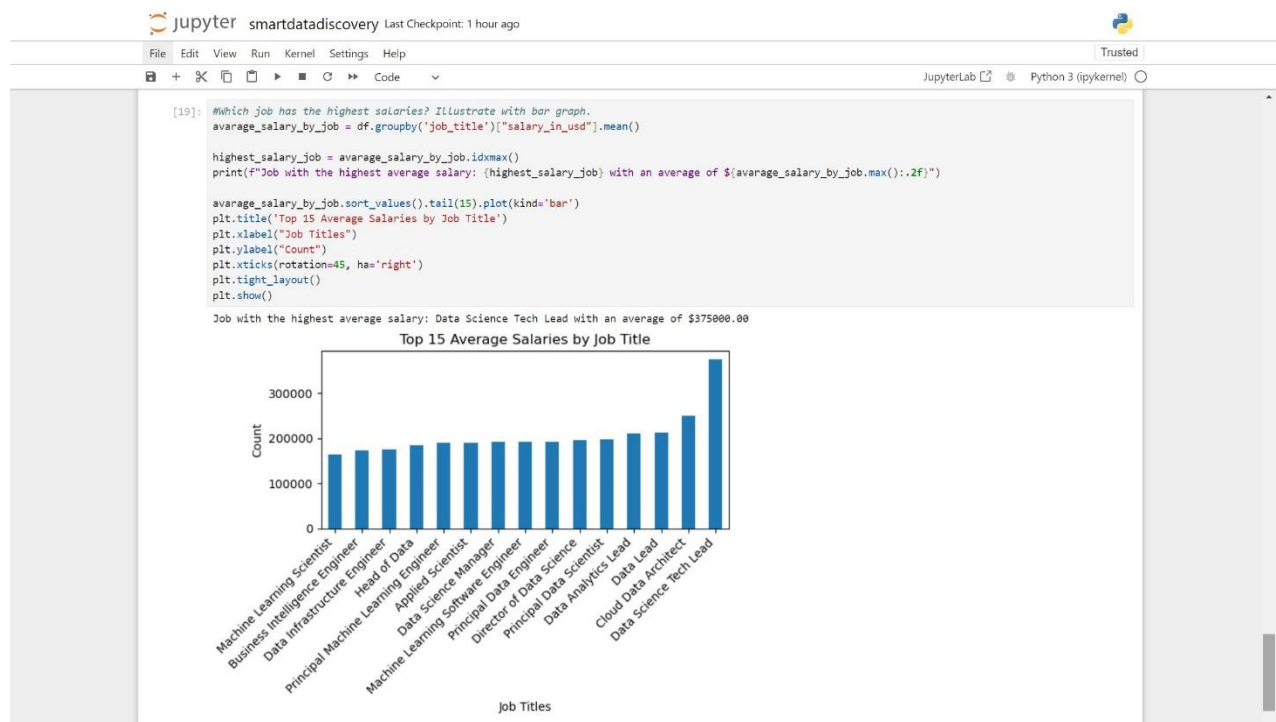


Figure 10 Job has the highest salaries

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

Experience level-based salary analysis reveals complex patterns of compensation at various career stages and sheds light on the relationship between tenure and compensation. Through the use of a bar graph to display the wage distributions across different experience levels, analysts are able to discover trends and inequalities, as well as entry-level positions, mid-career milestones, and senior leadership roles.

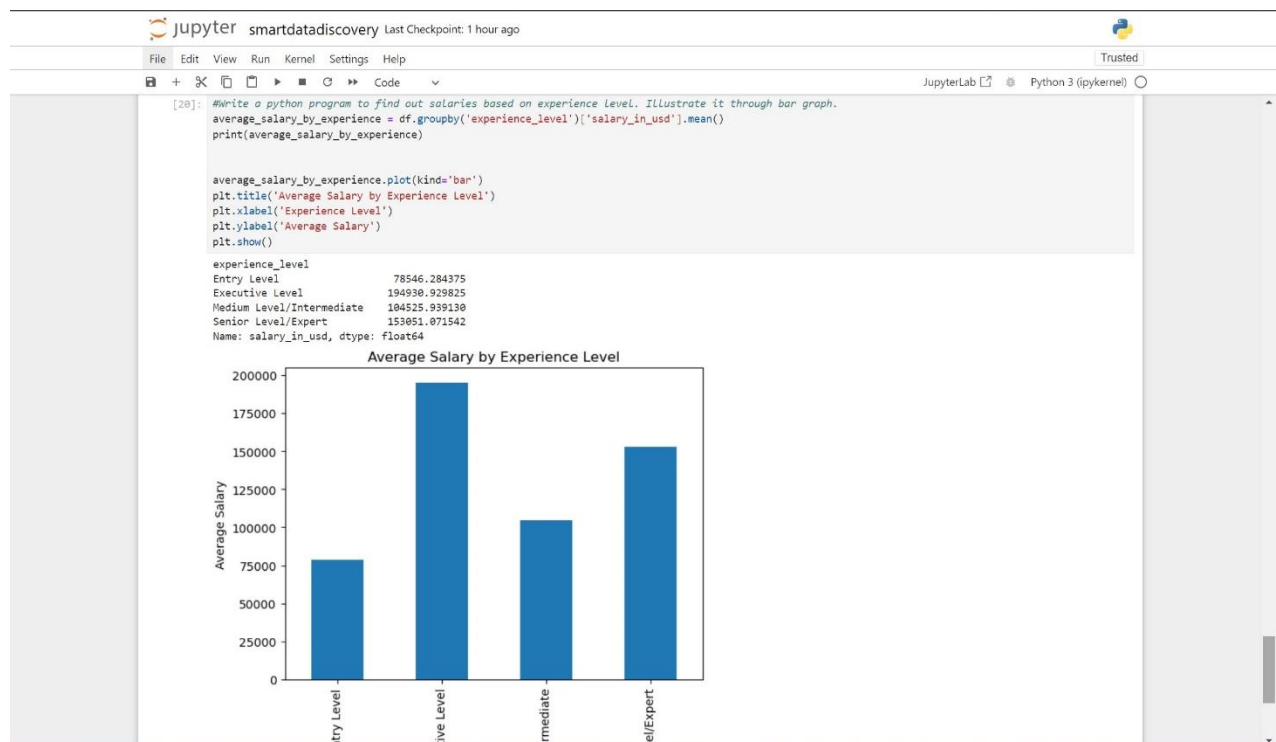


Figure 11 Python program to find out salaries based on experience level

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

Strong methods for displaying the distributional properties of numerical variables inside the dataset are histograms and box plots. Plotting histograms and box plots for selected data helps analysts identify outliers, anomalies, and underlying trends by providing insights into the variables' form, central tendencies, and dispersion. Additionally, analysts ensure that the graphical representations are useful as communication tools by adding appropriate labels and annotations to improve the graphical representations' interpretability and clarity.

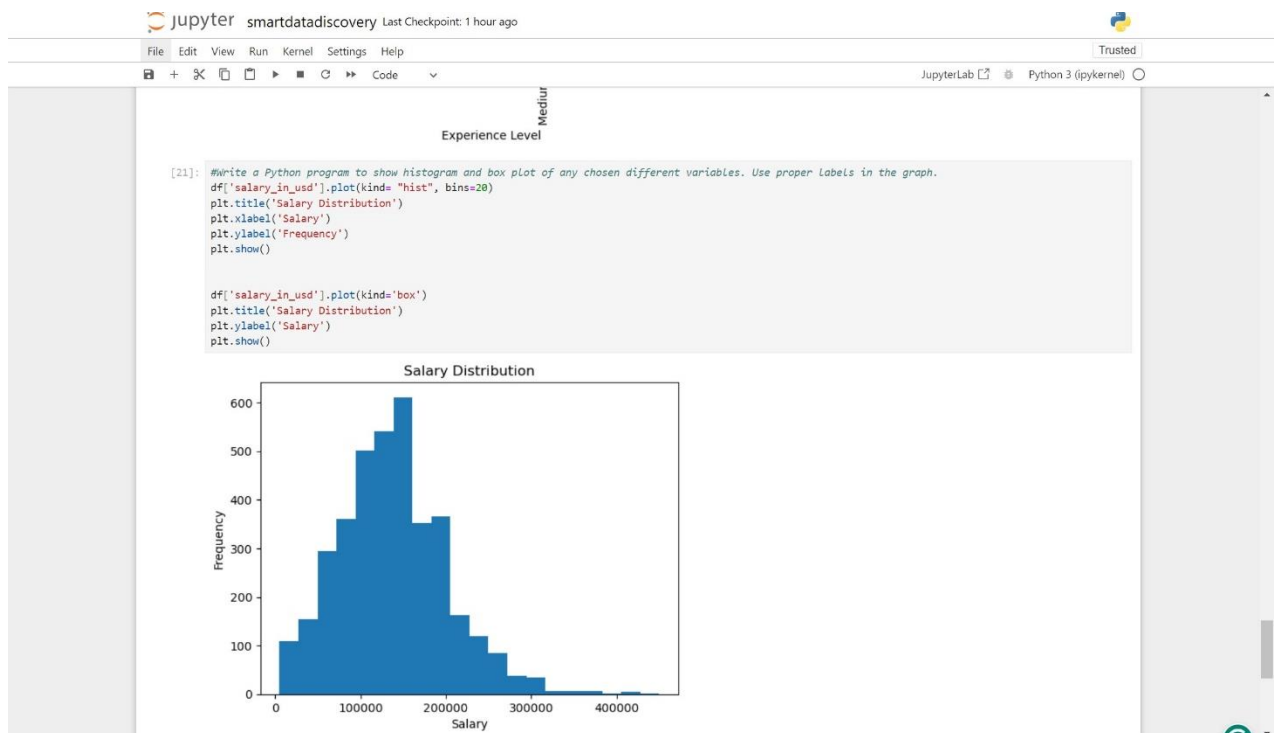


Figure 12 Python program to show histogram

6. Conclusion

In conclusion, this assignment has provided us with valuable insights into various aspects of data manipulation, analysis, and retrieval from diverse data sources. Through hands-on experience with Python programming and libraries like Pandas and Matplotlib, we have honed our skills in handling real-world datasets effectively. By delving into tasks such as data preparation, analysis, and visualization, we have gained a deeper understanding of the intricacies involved in working with data. Moving forward, the knowledge and expertise gained from this assignment will serve as a solid foundation for tackling more complex data science challenges and driving informed decision-making in the future.

References

Berthold, P. D. (2018). *Data Understanding*. Chicago: Springer.

ScienceDirect. (2016). Data Understanding. *ScienceDirect*, 1.