Assignment 1 *By Mohib Ali Khan 33370311*

Task A: Data Exploration and Auditing:

A1.

1. Can refer to the screenshot attached

2. As per the information we find at the end of the table, there are 3227 rows and 11 columns, meaning there are 11 different variables with a total of 3227 instances.

Code:

//salaries = pd.read_csv("salaries.csv")
salaries.info()//

A1. Dataset size

In [203]:
```
salaries = pd.read_csv("salaries.csv")
salaries.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3227 entries, 0 to 3226
Data columns (total 11 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   work_year           3227 non-null   int64 
 1   experience_level    3227 non-null   object
 2   employment_type     3227 non-null   object
 3   job_title           3227 non-null   object
 4   salary              3227 non-null   int64 
 5   salary_currency     3227 non-null   object
 6   salary_in_usd       3227 non-null   int64 
 7   employee_residence  3227 non-null   object
 8   remote_ratio        3227 non-null   int64 
 9   company_location    3227 non-null   object
 10  company_size        3227 non-null   object
dtypes: int64(4), object(7)
memory usage: 277.4+ KB
```

A2.

1. Can refer to the screenshot attached

Code:

//salaries.head(8)//

A2. Data auditing

```
In [7]: salaries.head(8)
```

Out[7]:

	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	com
0	2023	SE	FT	AI Scientist	1500000	ILS	427820	IL	0	IL	
1	2023	SE	FT	Machine Learning Engineer	216000	USD	216000	US	100	US	
2	2023	SE	FT	Machine Learning Engineer	184000	USD	184000	US	100	US	
3	2023	SE	FT	Data Engineer	180000	USD	180000	US	100	US	
4	2023	SE	FT	Data Engineer	165000	USD	165000	US	100	US	
5	2023	SE	FT	Data Scientist	185900	USD	185900	US	0	US	
6	2023	SE	FT	Data Scientist	129300	USD	129300	US	0	US	
7	2023	SE	FT	Data Engineer	145000	USD	145000	US	0	US	

Code:

//salaries.tail(12)//

```
In [8]: salaries.tail(12)
```

Out[8]:

	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location
3215	2020	MI	FT	Data Engineer	130800	USD	130800	ES	100	US
3216	2020	SE	FT	Machine Learning Engineer	40000	EUR	45618	HR	100	HR
3217	2021	SE	FT	Director of Data Science	168000	USD	168000	JP	0	JP
3218	2021	MI	FT	Data Scientist	160000	SGD	119059	SG	100	IL
3219	2021	MI	FT	Applied Machine Learning Scientist	423000	USD	423000	US	50	US
3220	2021	MI	FT	Data Engineer	24000	EUR	28369	MT	50	MT
3221	2021	SE	FT	Data Specialist	165000	USD	165000	US	100	US
3222	2020	SE	FT	Data Scientist	412000	USD	412000	US	100	US
3223	2021	MI	FT	Principal Data Scientist	151000	USD	151000	US	100	US
3224	2020	EN	FT	Data Scientist	105000	USD	105000	US	100	US
3225	2020	EN	CT	Business Data Analyst	100000	USD	100000	US	100	US

Code:

//salaries.sample(6)//

```
In [6]: salaries.sample(6)
```

Out[6]:

	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location
2688	2022	SE	FT	Data Scientist	260000	USD	260000	US	100	US
2948	2021	MI	FT	ML Engineer	7000000	JPY	63711	JP	50	JP
2213	2022	SE	FT	Data Engineer	160000	USD	160000	US	100	US
2539	2022	EX	FT	Data Science Manager	260500	USD	260500	US	0	US
1014	2023	MI	FT	Data Engineer	130000	USD	130000	US	0	US
893	2023	SE	FT	Applied Scientist	350000	USD	350000	US	0	US
2181	2022	SE	FT	ETL Developer	63000	USD	63000	US	100	US
2587	2022	SE	FT	Data Analyst	117000	USD	117000	US	100	US
3014	2021	MI	FT	Data Engineer	110000	PLN	28476	PL	100	PL
2102	2022	SE	FT	Data Scientist	225000	USD	225000	US	0	US
942	2023	SE	FT	Analytics Engineer	200000	USD	200000	US	100	US
405	2023	MI	FT	Data Engineer	85000	GBP	103202	GB	0	GB

A3.

1. Can refer to the screenshot attached

Code:

//salaries.dtypes//

A3. Data Types

```
In [11]: salaries.dtypes
```

```
Out[11]: work_year              int64
         experience_level      object
         employment_type       object
         job_title             object
         salary                 int64
         salary_currency       object
         salary_in_usd          int64
         employee_residence    object
         remote_ratio           int64
         company_location      object
         company_size          object
         dtype: object
```

A4.

1. Can refer to the screenshot attached

Code:

//salaries['salary_in_usd'] = salaries['salary_in_usd'].apply(lambda x: x * 4.47)
salaries['salary_in_myr'] = salaries['salary_in_usd']
salaries//

A4. Conversion

```
In [53]: salaries['salary_in_usd'] = salaries['salary_in_usd'].apply(lambda x: x * 4.47)
```

```
In [54]: salaries['salary_in_myr'] = salaries['salary_in_usd']
```

```
In [55]: salaries
```

experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	company_size	salary_in_myr
SE	FT	AI Scientist	1500000	ILS	1912355.40	IL	0	IL	L	1912355.40
SE	FT	Machine Learning Engineer	216000	USD	965520.00	US	100	US	M	965520.00
SE	FT	Machine Learning Engineer	184000	USD	822480.00	US	100	US	M	822480.00
SE	FT	Data Engineer	180000	USD	804600.00	US	100	US	M	804600.00
SE	FT	Data Engineer	165000	USD	737550.00	US	100	US	M	737550.00
SE	FT	Data Scientist	185900	USD	830973.00	US	0	US	M	830973.00

A5.

1. If you refer to the screenshot attached firstly, we can observe that the mean remote ratio is about 48.280136 which tells us that most of the jobs had no amount of remote work done or partially remote work. Secondly, we can see that the max salary_in_myr is 2.011500e+06 which means 2.01 x 10^6 RM is the maximum salary paid. Thirdly, if you observe the work_year column we see that min and max tells us that the records are between the year 2020 to 2023.

Code:

//salaries.describe()//

A5. Descriptive Statistics

```
In [74]: salaries.describe()
```

Out[74]:

	work_year	salary	salary_in_usd	remote_ratio	salary_in_myr
count	3227.000000	3.227000e+03	3.227000e+03	3227.000000	3.227000e+03
mean	2022.273939	1.950125e+05	6.023338e+05	48.280136	6.023338e+05
std	0.693571	7.226896e+05	2.798106e+05	48.546623	2.798106e+05
min	2020.000000	6.000000e+03	2.294004e+04	0.000000	2.294004e+04
25%	2022.000000	9.500000e+04	4.128045e+05	0.000000	4.128045e+05
50%	2022.000000	1.350000e+05	5.812162e+05	50.000000	5.812162e+05
75%	2023.000000	1.796375e+05	7.703933e+05	100.000000	7.703933e+05

A6.

1. 85 unique job titles are recorded in the 'job_title' column.
2. Can refer to the screenshot attached for each different job title and their count.

Code:

//salaries['job_title'].nunique()
mode = {'job_title': 'count'}
jobs_df = salaries.groupby('job_title').agg(mode)
jobs_df.rename(
columns = {"job_title":"count"}, inplace= True)
pd.set_option('display.max_rows', None)
jobs_df
filter_df = salaries[salaries['job_title']=='Data Scientist']
job_count = len(filter_df)
total_count = len(salaries)
job_percent = ((job_count/ total_count) * 100)
job_percent//

A6. Exploring Job Titles

In [38]:
```
salaries['job_title'].nunique()
```
Out[38]: 85

In [151]:
```
mode = {'job_title': 'count'}
jobs_df = salaries.groupby('job_title').agg(mode)
jobs_df.rename(
columns = {"job_title":"count"}, inplace= True)
pd.set_option('display.max_rows', None)
jobs_df
```
Out[151]:

job_title	count
3D Computer Vision Researcher	4
AI Developer	5
AI Programmer	2
AI Scientist	16
Analytics Engineer	79
Applied Data Scientist	8
Applied Machine Learning Engineer	1
Applied Machine Learning Scientist	12
Applied Scientist	30
Autonomous Vehicle Technician	2

```
In [160]: filter_df = salaries[salaries['job_title']=='Data Scientist']
          job_count = len(filter_df)
          total_count = len(salaries)
          job_percent = ((job_count/ total_count) * 100)
          job_percent

Out[160]: 22.342733188720175
```

A7.

1. There are 70 different company locations recorded. Can refer to the screenshot attached for their name/code and counts.

Code:

//salaries['company_location'].nunique()
mode1 = {'company_location': 'count'}
location_df = salaries.groupby('company_location').agg(mode)
location_df.rename(
columns = {"job_title":"count"}, inplace= True)
location_df//

A7. Exploring location of Companies

```
In [37]: salaries['company_location'].nunique()

Out[37]: 70
```

```
In [155]: mode1 = {'company_location': 'count'}
          location_df = salaries.groupby('company_location').agg(mode)
          location_df.rename(
          columns = {"job_title":"count"}, inplace= True)
          location_df
```

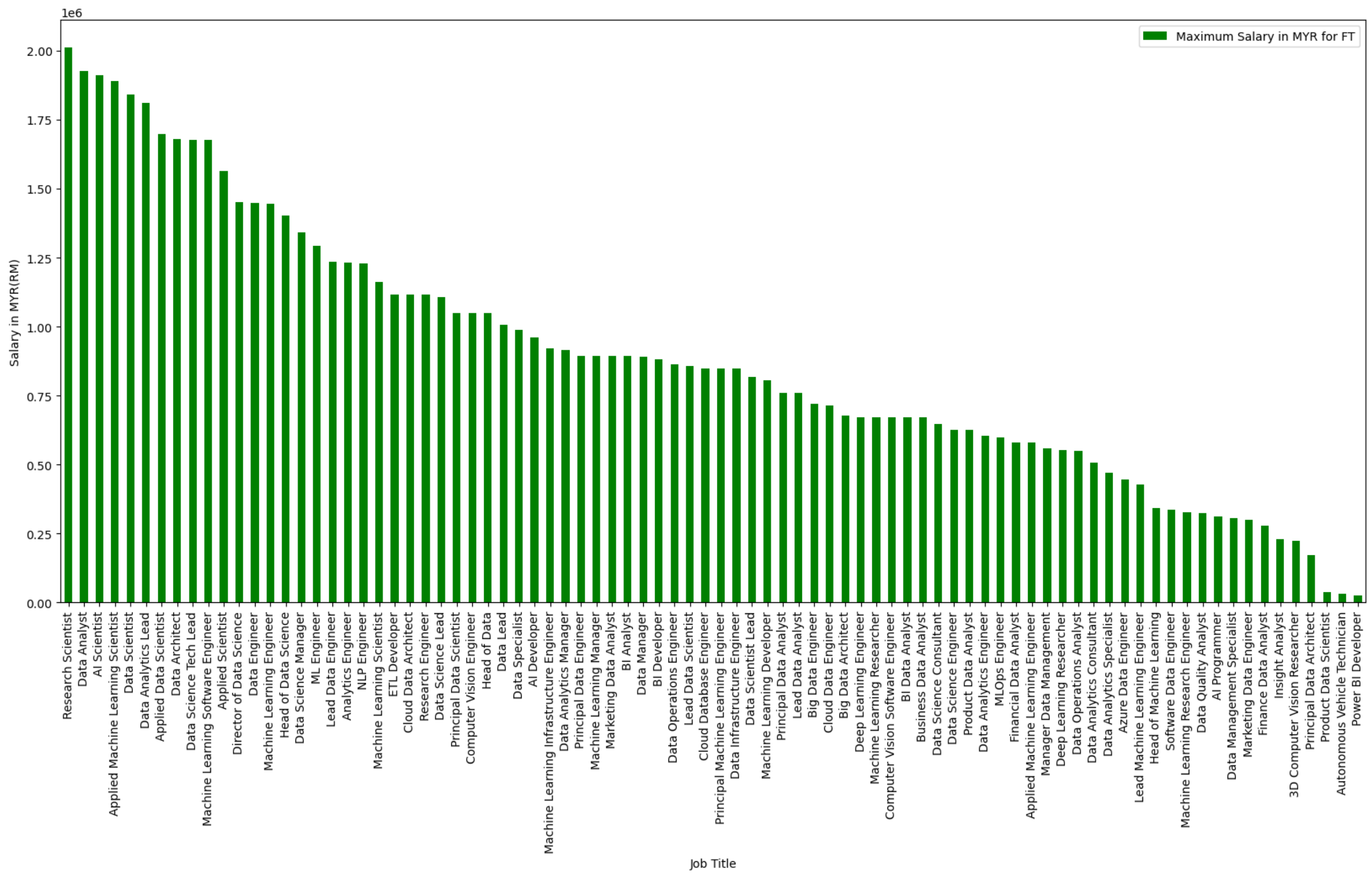
PR	4
PT	14
RO	2
RU	3
SE	2
SG	6
SI	4
SK	1
TH	3
TR	5
UA	1
US	2575
VN	1

Task B: Group Level Analysis and Visualisation:

B1.1

Code:

```
//ft_df = salaries.loc[salaries['employment_type'] == 'FT']
total_salary = ft_df.groupby('job_title')['salary_in_myr'].max()
salary_sorted = total_salary.sort_values(ascending=False)
#Plots the bar graph
ax = salary_sorted.plot(kind='bar', figsize=(20, 9),color='green')
ax.set_xlabel('Job Title')
ax.set_ylabel('Salary in MYR(RM)')
ax.legend(["Maximum Salary in MYR for FT"])
plt.show()//
```



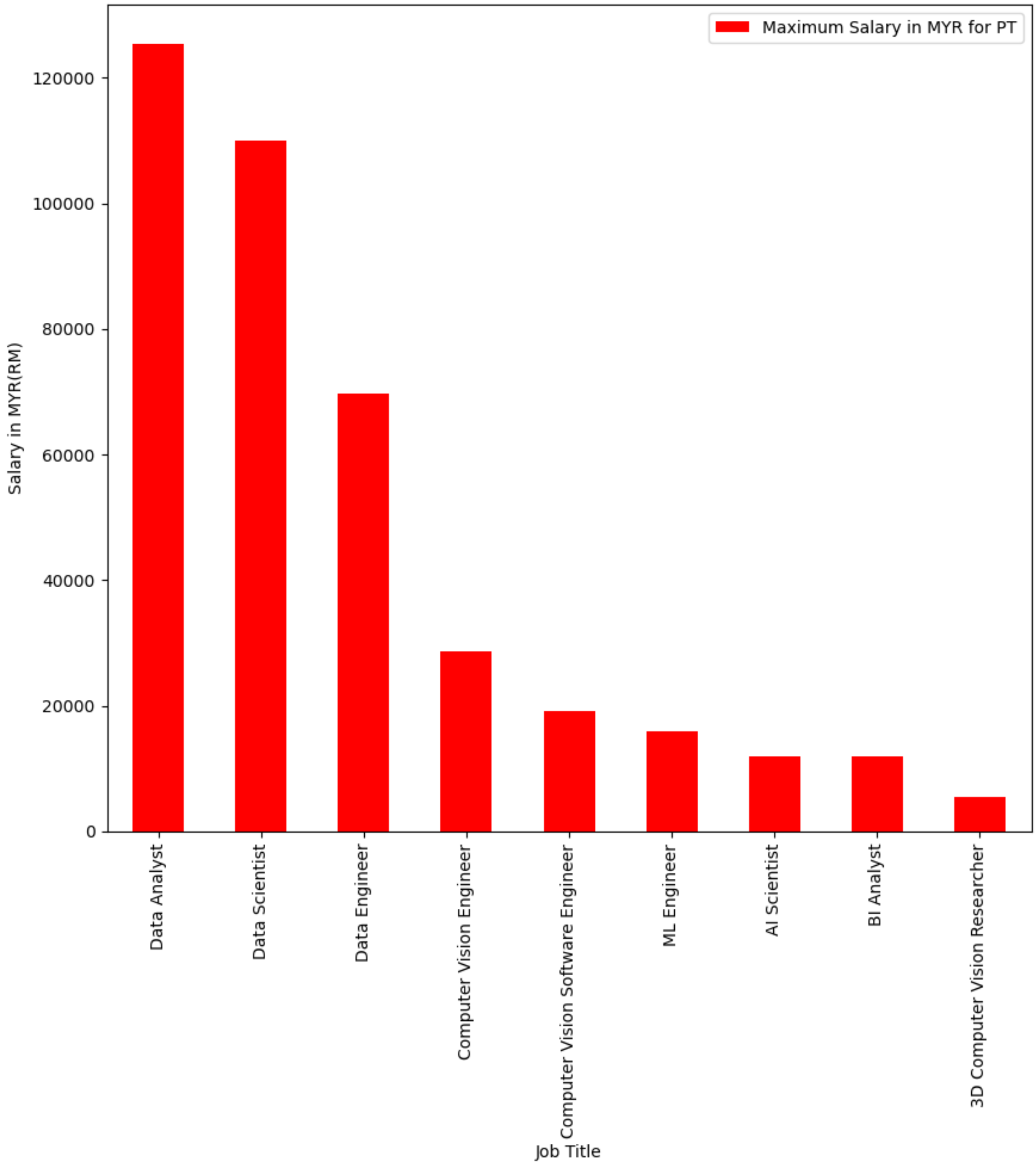
The job with the highest full-time (FT) employment type salary is Research Scientist,

We have used a bar graph to analyze this data since a bar graph would easily let us distinguish between each bar segment which job title is being paid the highest salary, hence if we observe our X-axis we can find the job titles and on the Y axis we can find the salary for each job title in MYR, therefore, the highest bar on the graph is for Research Scientist proving to be the highest paid full-time job.

B1.2

Code:

```
//#filtering the data
pt_df = salaries.loc[salaries['employment_type'] == 'PT']
total_salary = pt_df.groupby('job_title')['salary_in_myr'].max()
salary_sorted = total_salary.sort_values(ascending=False)
# Plots the bar graph
ax = salary_sorted.plot(kind='bar', figsize=(20, 9), color='red')
ax.set_xlabel('Job Title')
ax.set_ylabel('Salary in MYR(RM)')
ax.legend(["Maximum Salary in MYR for PT"])
plt.show()//
```

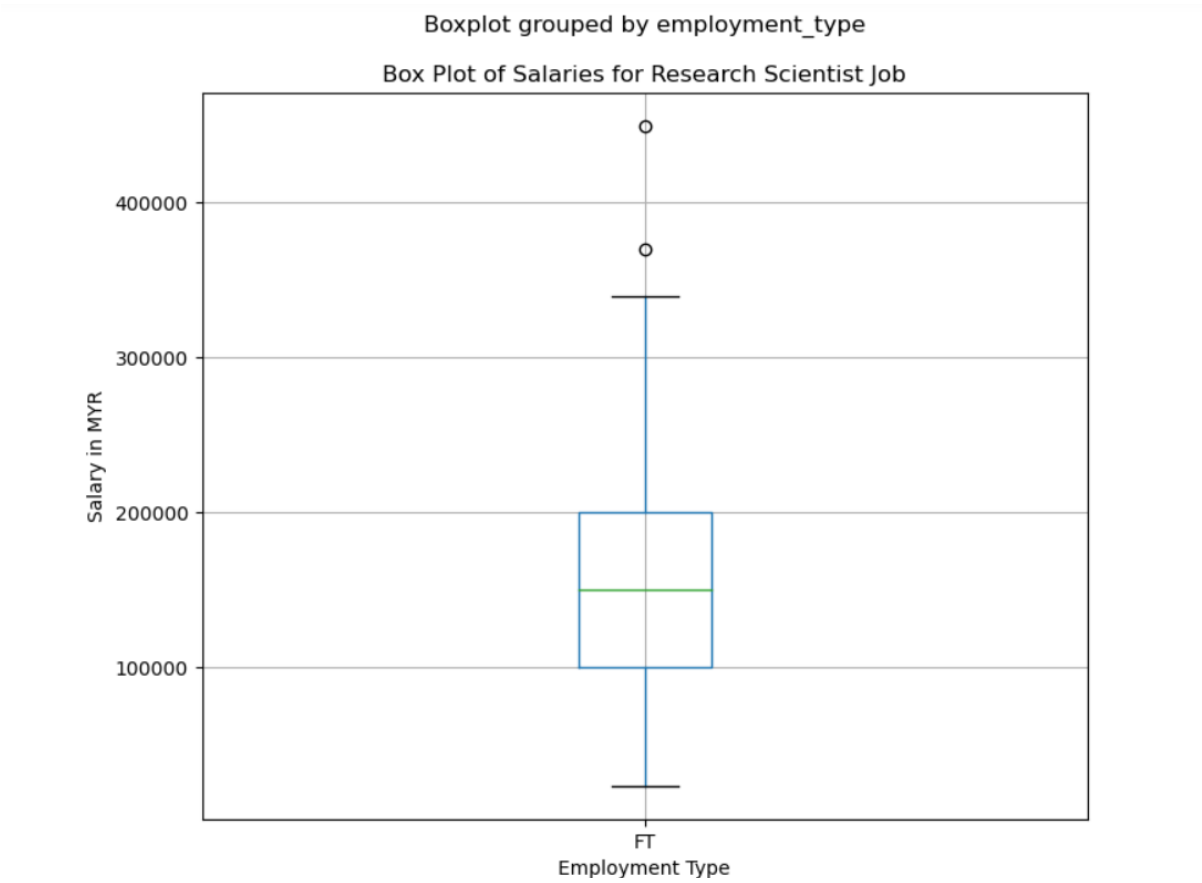


The job with the highest part-time (PT) employment type salary is Data Analyst, We have used a bar graph to analyze this data since a bar graph would easily let us distinguish between each bar segment which job title is being paid the highest salary, hence if we observe our X-axis we can find the job titles and on the Y axis we can find the salary for each job title in MYR, therefore, the highest bar on the graph is for Data Analyst proving to be the highest paid part-time job. We also observe that they are paid less than full-time job highest salary and also we do not find as many jobs as part-time employment type.

B1.3

Code:

```
//job_df = salaries.loc[salaries['job_title'] == 'Research Scientist]
# create a box plot of salaries by employment type
ax = job_df.boxplot(column='salary_in_myr', by='employment_type', figsize=(8, 7))
# set axis labels and title
ax.set_xlabel('Employment Type')
ax.set_ylabel('Salary in MYR')
ax.set_title('Box Plot of Salaries for Research Scientist Job')
plt.show()//
```



Since the highest paid salary job for full-time (FT) employment type was Research Scientist hence we tried plotting a bar graph to visualize the data related to it but we would just get the maximum and minimum salary for one employment type moreover, it does not look a good representation of data hence I used box plot because we just have one type of employment type associated with Research Scientist job therefore, box plot provides us with various other information regarding the job such as the median salary, range of the salary, the highest salary and minimum salary for this job.

B2.1

Code:

//largest_three = salaries['company_location'].value_counts().nlargest(3)

largest_three//

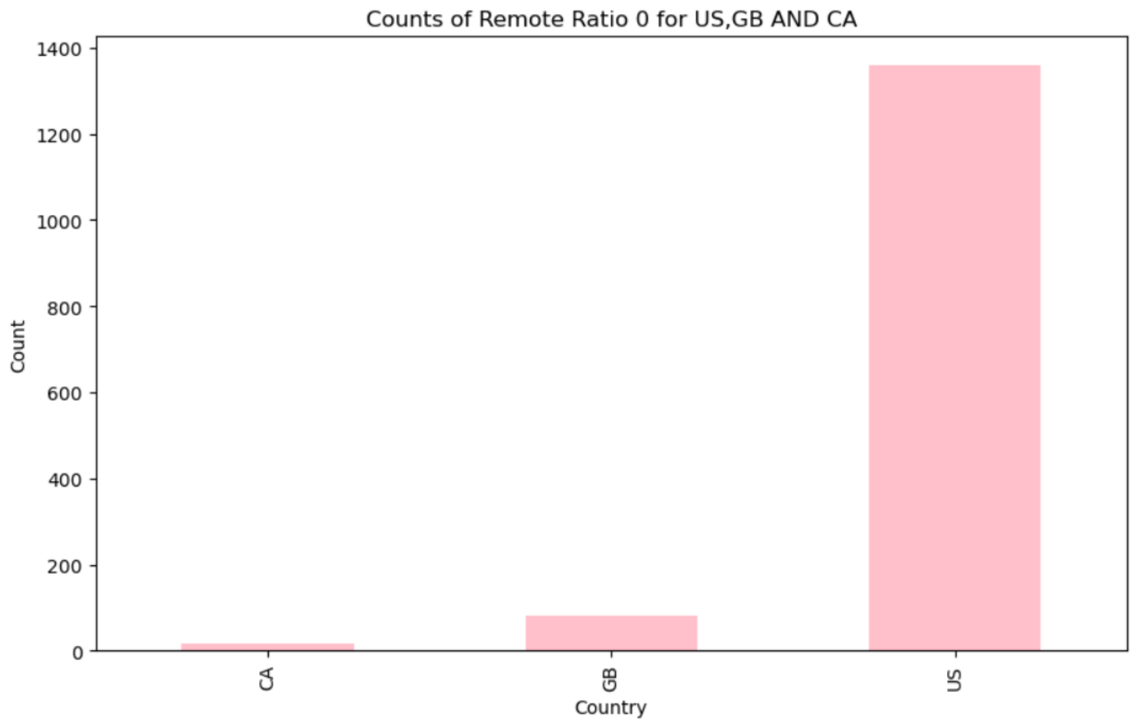
B2. Investigating Remote Ratio

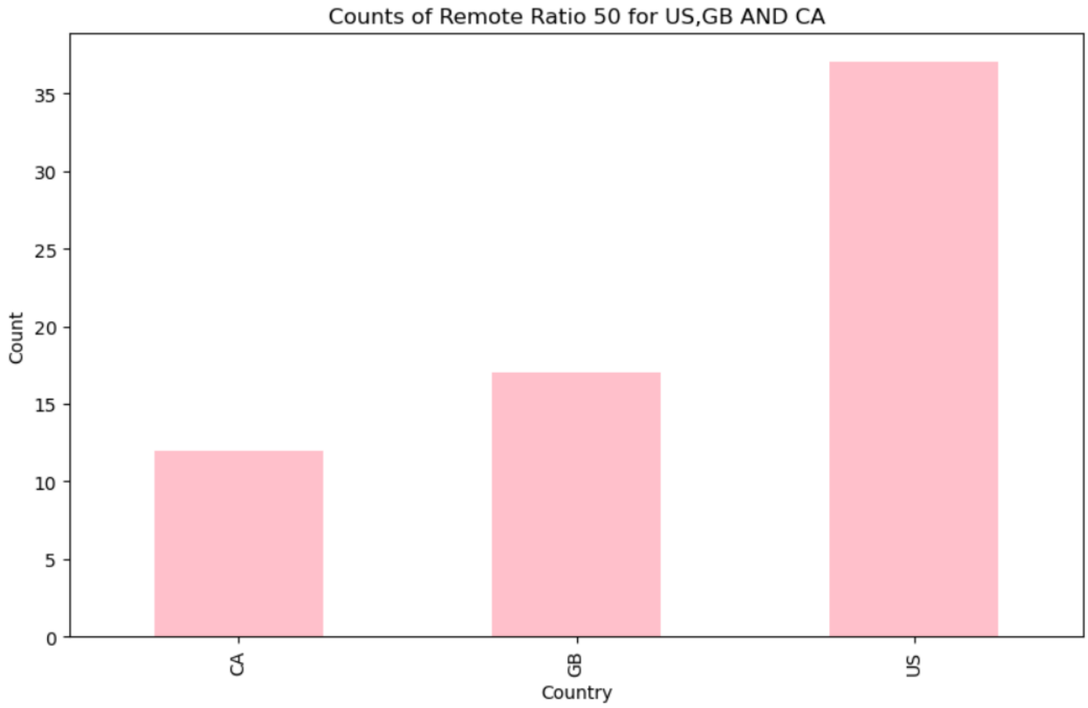
```
In [181]: largest_three = salaries['company_location'].value_counts().nlargest(3)

          largest_three

Out[181]: US    2575
          GB     159
          CA      69
          Name: company_location, dtype: int64
```

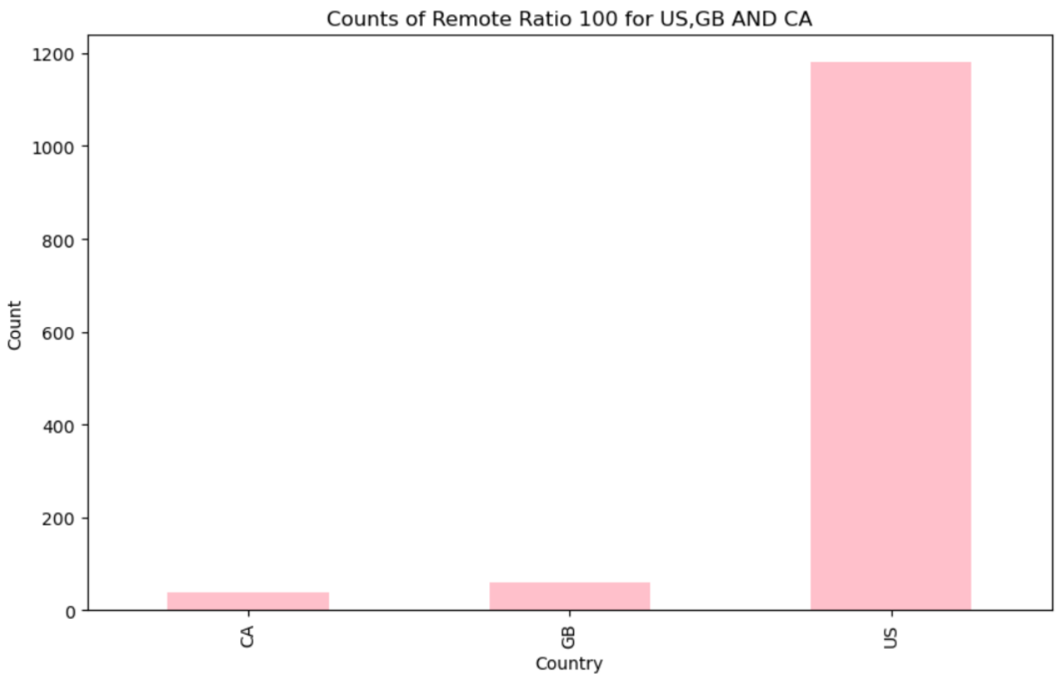
B2.2

```
top_countries = ['US', 'GB', 'CA']
filtered_data = salaries[salaries['company_location'].isin(top_countries)]
# group the data by country and remote_ratio
grouped_data = filtered_data.groupby(['company_location',
'remote_ratio']).size().unstack(fill_value=0)
# create the bar chart for each remote ratio
for ratio in [0, 50, 100]:
 # get the data for the current remote ratio
   ratio_data = grouped_data[ratio]
    # create a bar chart for the current remote ratio
   ax = ratio_data.plot(kind='bar', figsize=(10, 6), color='pink')
  # set the axis labels and title
   ax.set_xlabel('Country')
   ax.set_ylabel('Count')
```

```
ax.set_title(f"Counts of Remote Ratio {ratio} for Top Three Countries")
# display the chart
plt.show()
```



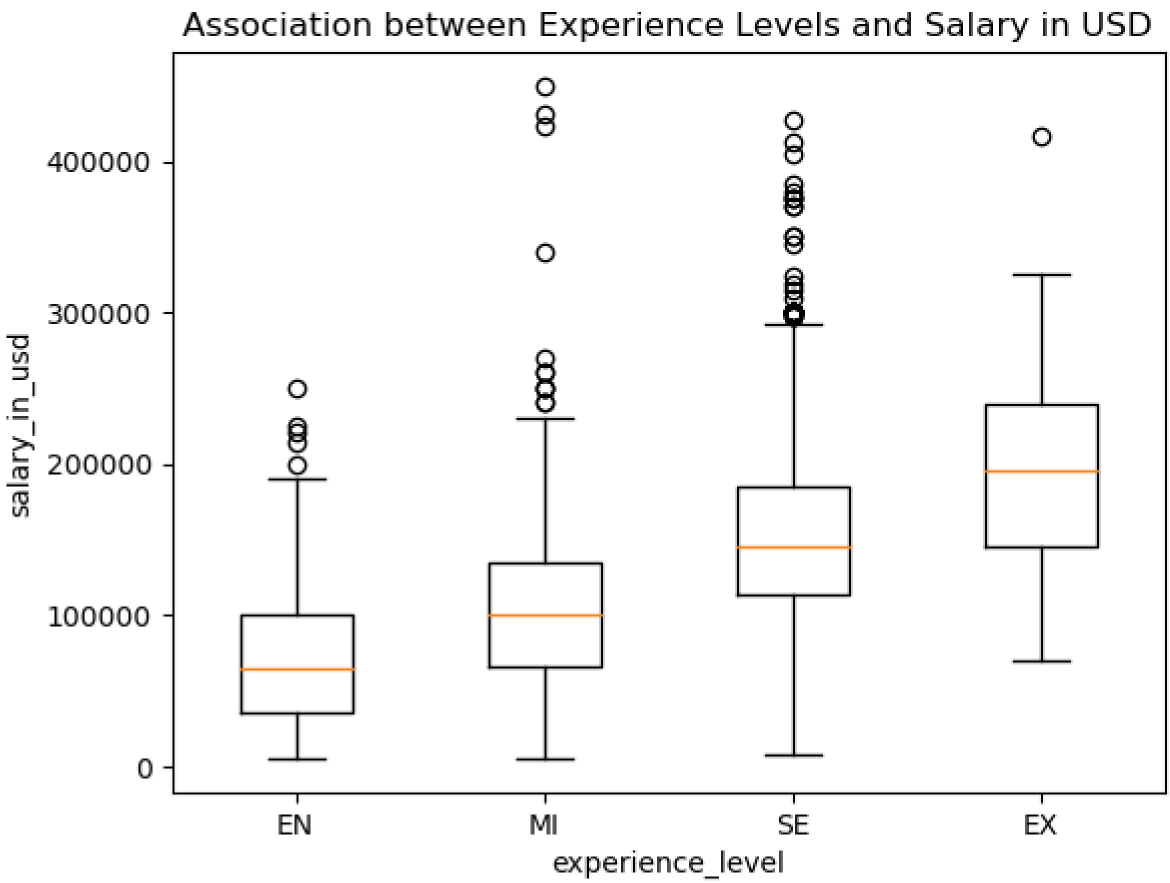

Counts of Remote Ratio 50 for US,GB AND CA
Count
35
30
25
20
15
10
5
0
CA
GB
US
Country


Counts of Remote Ratio 100 for US,GB AND CA
Count
1200
1000
800
600
400
200
0
CA
GB
US
Country

I have used a bar graph to represent this data since we had to visualize the remote ratio for each country hence in my opinion a clearer and better representation would be with counts of each top three company_locations for each remote ratio. We see that the United States (US) tops in every remote ratio category while in between Canada (CA) and Great Britain (GB), we observe that Great Britain has more jobs in every category. The major thing to note is that Canada's highest count is in partially remote jobs and in the other two categories it seems to be low.

B3.1

```
//salaries3 = pd.read_csv("salaries.csv")
#the order I want my experience level to be in
level_order = ['EN', 'MI', 'SE', 'EX']
# Create a box plot using matplotlib
plt.boxplot([salaries3[salaries3['experience_level'] == 'EN']['salary_in_usd'],
             salaries3[salaries3['experience_level'] == 'MI']['salary_in_usd'],
             salaries3[salaries3['experience_level'] == 'SE']['salary_in_usd'],
             salaries3[salaries3['experience_level'] == 'EX']['salary_in_usd']],
            labels=level_order)
# Add axis labels and a title
plt.xlabel('experience_level')
plt.ylabel('salary_in_usd')
plt.title('Association between Experience Levels and Salary in USD')//
```



I have used a box plot to represent this data since we had to visualize the experience_level relation with salary, after seeing this visualization I conclude that there is an association between experience level and salary because if you start from the left side of the X-axis to right you will notice that the experience level increase and with you must notice that the median for each of the box plot for the respective experience levels increase as the experience level increases or we can say the median or interquartile range of each experience level is higher than the previous one.