AMOD-5460H – Data Science with Python

Assignment 4

Jugal Chauhan – 0755856

Introduction

The state of Data Science job roles has been consistently changing as the field keeps stemming multiple job roles in the industry. From ‘Director of Data Science’ to ‘Head of Data’ and ‘Data Visualization Analyst’ the creation of new job descriptions is imminent. In this regard, it might be difficult to track the salary and popularity trends of Data Science job titles. However, simple visualization techniques can provide details of the course of these job roles. In this assignment, a modest attempt of tracking the direction of jobs titles in the field of Data Science has been made using the pandas and matplotlib modules of python language. The visualizations help us understand what pattern the employees or countries are following, and the changes in salaries of job titles can be tracked. These visuals can assist corporate human resources to grasp and further design the job titles for their organization. Additionally, students can get the picture of what job description to follow based on the company locations and salaries offered.

Dataset description

The dataset is providing information about Job Title, Employment Type, Experience Level, Expertise Level, Salary and its Currency, Salary in USD, Employee residence, Company Size, and the Year. The dataset is obtained from Kaggle website and can be accessed using this link.

The dataset contains 11 attributes and 3300 rows and has zero instance of null or duplicated values.

	Job Title	Employment Type	Experience Level	Expertise Level	Salary	Salary Currency	Company Location	Salary in USD	Employee Residence	Company Size	Year
0	Data Engineer	Full-Time	Senior	Expert	210000	United States Dollar	United States	210000	United States	Medium	2023
1	Data Engineer	Full-Time	Senior	Expert	165000	United States Dollar	United States	165000	United States	Medium	2023
2	Data Engineer	Full-Time	Senior	Expert	185900	United States Dollar	United States	185900	United States	Medium	2023
3	Data Engineer	Full-Time	Senior	Expert	129300	United States Dollar	United States	129300	United States	Medium	2023
4	Data Scientist	Full-Time	Senior	Expert	140000	United States Dollar	United States	140000	United States	Medium	2023

Figure 1: Dataset View

Most Popular Job Roles in Data Science

To begin with, we will try to understand what job roles are currently most approved in the industry, that which have the greatest number of peoples employed. To visualize this, a bar plot is used that gives us a thorough picture of most popular job roles.

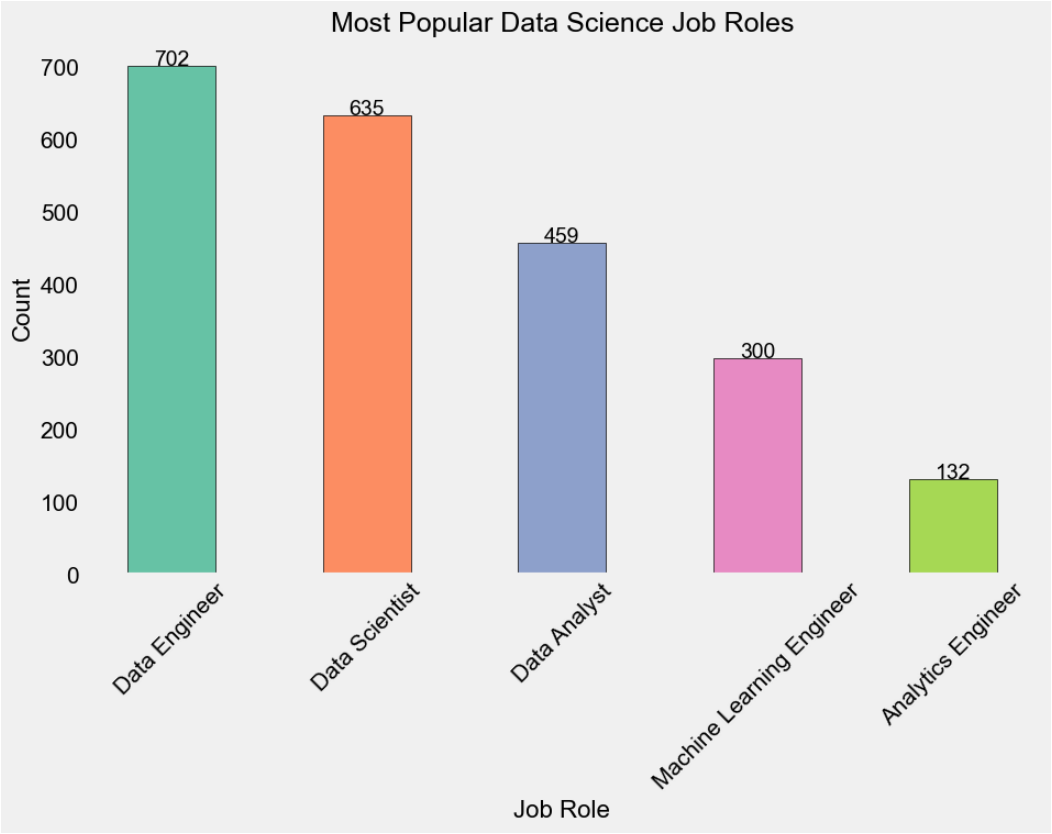

Figure 2: Most Popular Data Science Job Roles

The relevant findings from this plot are:

1. With 702 counts, the largest number of people are employed as ‘Data Engineer’ in the field of data science

2. Some popular job roles like ‘Business Analyst’ or ‘Financial Analyst’ do not make it to the top five popular job titles.

3. Data Scientist, often considered most popular and in demand job description, comes close second to the title of Data Engineer.

Experience level and Company size

It is vital to understand the level of experience of employees in job titles as it gives better understanding of the skill level of the employees. The experience levels are categorized as: Senior, Executive, Mid, and Entry. Furthermore, the job hiring and salaries also depend on the size of the company which are categorized as Small, Medium, and Large.

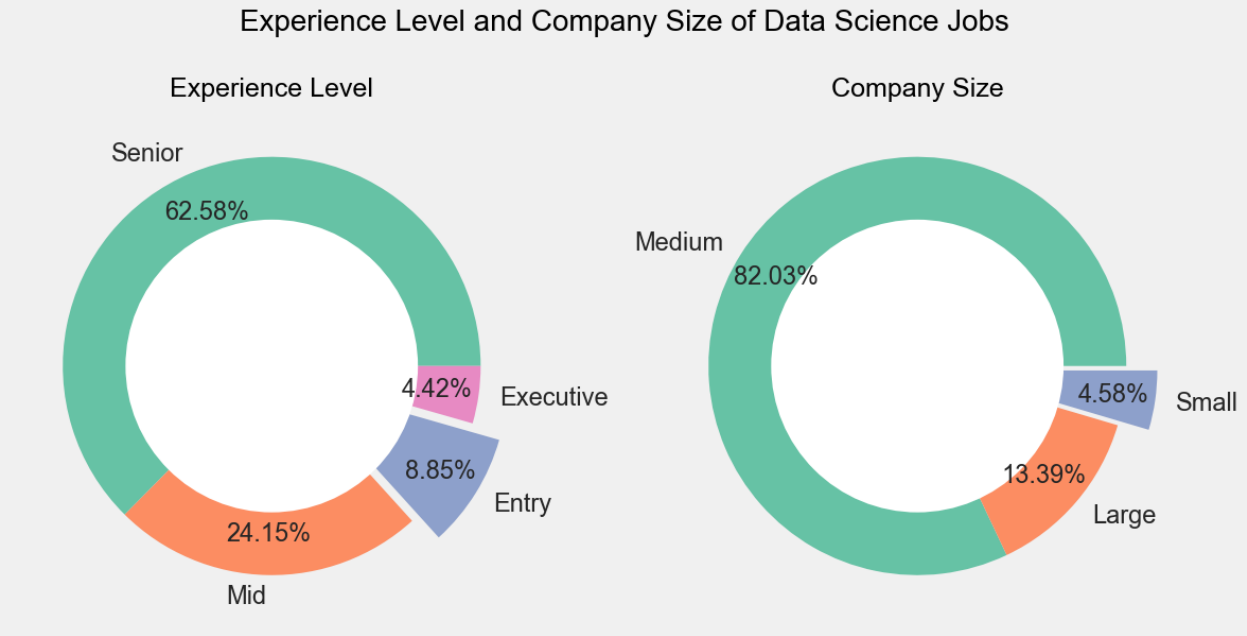

Figure 3: Experience level and Company size

The relevant findings from this plot are:

1. Out of 3300 different job titles, as low as 8.85% of skilled technicians are offered entry level jobs.

2. About 62% of employees have the senior experience level, which is more than double of Mid level employees.

3. When it comes to company size, medium sized company dominate the industry with 82%.

4. Smal sized companies only account for 4.5% in the industry, while large sized companies are about 13%.

Countries having most number of employees and companies

The data science job demographic is important when it comes to understand which country offers most opportunities of data science jobs. This can be better perceived through a grouped bar chart.

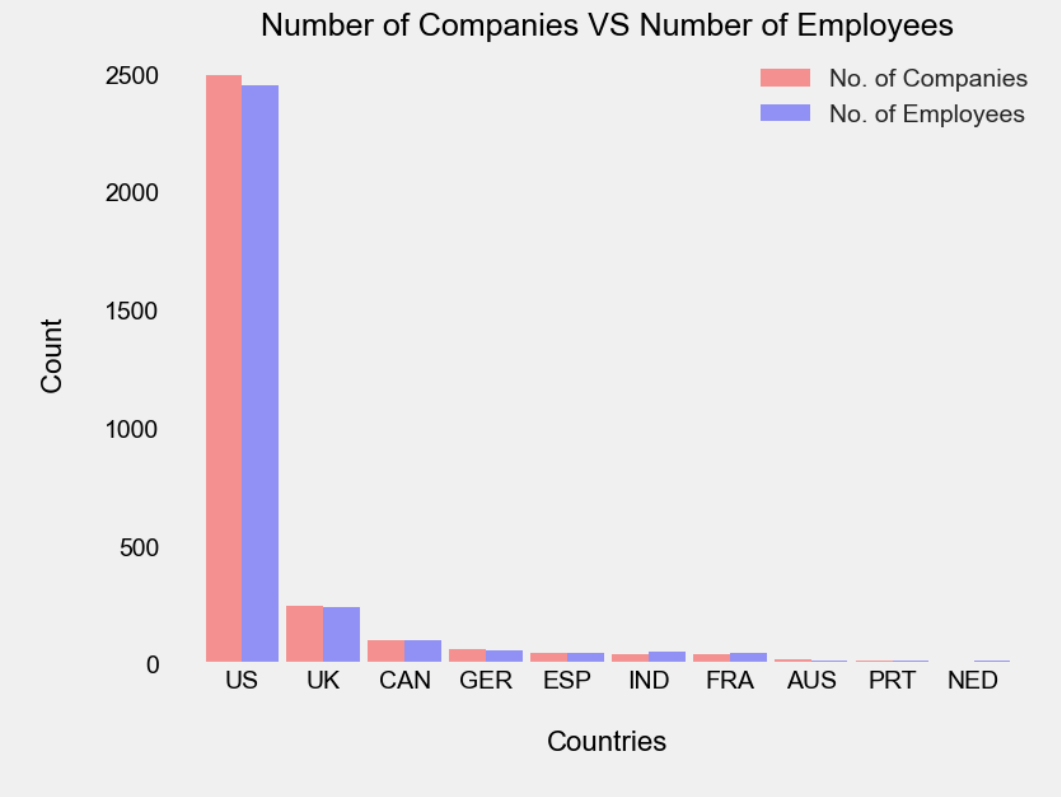

Figure 4: Countries with most companies and employees

The relevant findings from this plot are:

1. US has, with a huge margin, the greatest number of data science companies and equally high number of employees.

2. Most countries have almost equal ratio of companies vs employees.

3. India and France seem to be only two countries that have more employees than data science companies.

Salary Comparison of different job titles from 2020 to 2023

Finally, the salary comparison of most popular data science job titles is made, the average salaries are calculated for the years ranging from 2020 to 2023.

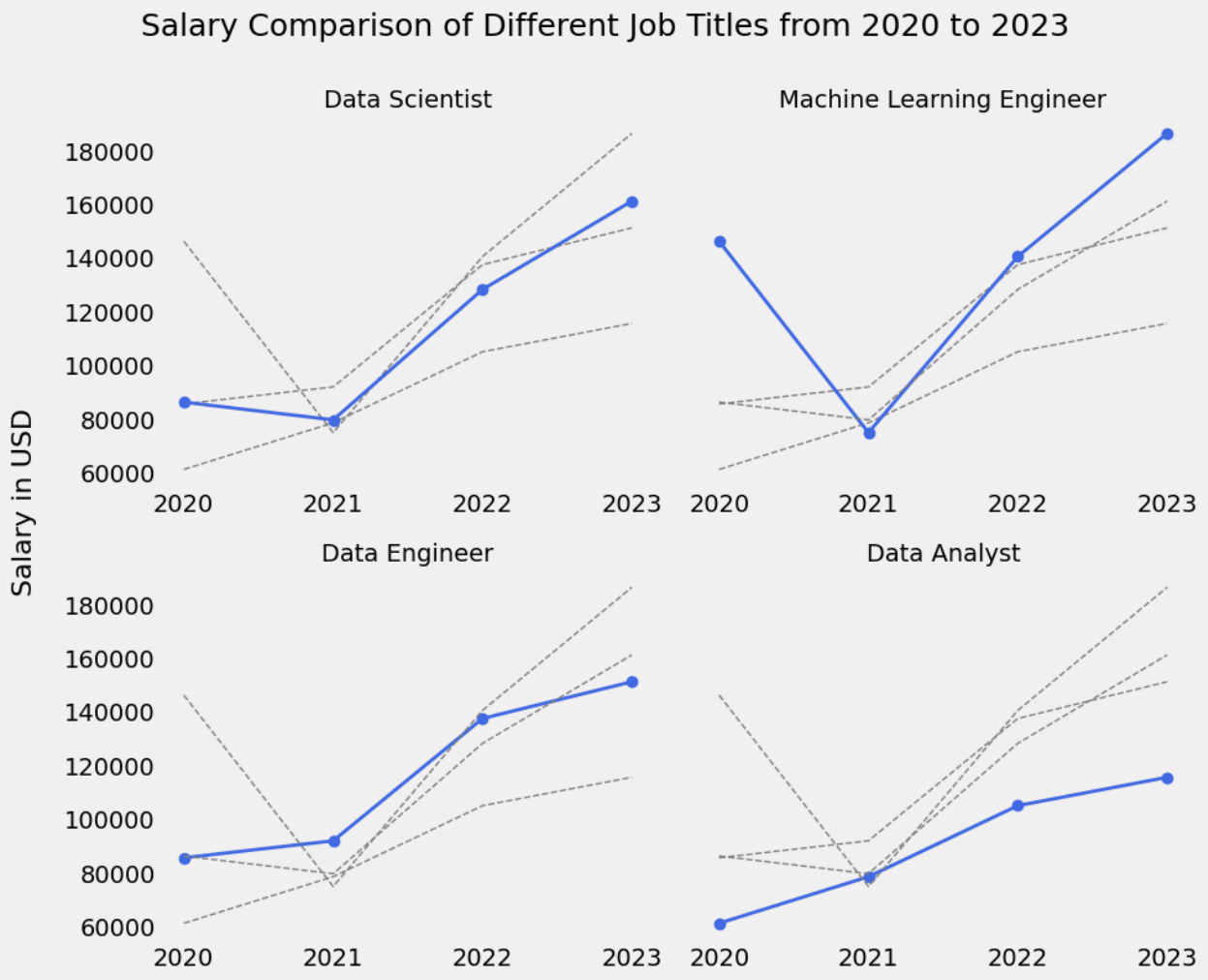

Figure 5: Salary comparison for different job titles

The relevant findings from this plot are:

1. Average salaries for Data Scientist, ML Engineer, Data Analyst, and Data Engineer all follow increasing trend from the year 2021.

2. A slight drop in salaries is observed for Data Scientist salary in 2021, and in case of ML engineer a severe drop in salary is observed in 2021.

3. From the four job titles, highest average salary in 2023 is observed for ML engineer at approximately $180,000. In contrast, Data Analyst position has least average salary in 2023.

Improvements and Further Analysis.

The visualization presented in this report showcase the trend of current Data Science roles in the industry. This report only presents few of the important insights, and on further analysis, much detailed information can be extracted (such as distribution of experience level among different job titles etc.). Certain limitations include lack of detailed categorization of company size which does not give much details of what exactly would be considered a 'Medium/Large/Small sized company.' The code optimization for the generation of plots can be done by implementing loops instead of redundant codes. Finally, as always is the case, a larger sample size can provide even more consistent insights for the topic.

Appendix - Python Code

```
import os

from datetime import datetime

print(os. getcwd())

print(datetime.now())



# DATA SCIENCE JOBS SALARY ANALYSIS USING MATPLOTLIB



# Importing required libraries and loading dataset



import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import numpy as np
```

```
data = pd.read_csv("Latest_Data_Science_Salaries.csv")

print(data.info())




#--------------------------------VIZ 1----------------------------------#


# DISTRIBUTION OF SALARY OF DATA SCIENCE JOBS


# We will create a histogram indicating the distribution of salary in USD of data science jobs


# Setting figure size

fig, ax = plt.subplots(figsize = (8,6))


# Plotting the histogram

ax = plt.hist(data['Salary in USD'], bins=30, color = 'red', alpha=0.5, edgecolor='black')

plt.grid(False)


# Setting labels & title

plt.xlabel('Salary in USD')

plt.ylabel('Count')

plt.title('Distribution of Data Science Salary')

plt.show()
```

```
#----------------------------------VIZ 2----------------------------------#

# MOST POPULAR JOB ROLES IN DATA SCIENCE

# Now we will create a bar plot showcasing five most popular jobs in field of data science

plt.figure(figsize=(10,6)) # Setting the figure size
plt.style.use('fivethirtyeight') # Using 'fivethirtyeight' plot style
jobtitle = data['Job Title'].value_counts() # Storing required data in new variable

# addlabels function displays the count of jobs above the bar
def addlabels(x,y):
    for i in range(len(x)):
        plt.text(i, y[i], y[i], ha = 'center', fontsize=14, color='black')

# plt.bar will plot the bar plot for the jobtitle data
plt.bar(jobtitle.index[0:5], jobtitle.values[0:5],
        color=sns.color_palette("Set2"),
        width=0.45,
        edgecolor='black')

# Customizing ticks and labels & title
plt.xticks(rotation=45, fontsize=15.0, color='black')
plt.yticks(fontsize = 15.0, color='black')
```

```
addlabels(jobtitle.index[0:5], jobtitle.values[0:5])

plt.xlabel('Job Role', fontsize=16, color='black')

plt.ylabel('Count', fontsize=16, color='black')

plt.title('Most Popular Data Science Job Roles', fontsize = 18.0, color='black')


# We do not want to see the grid in our plot

plt.grid(False)


plt.show()



#----------------------------------VIZ 3-------------------------------------------#


# EXPERIENCE LEVEL & COMPANY SIZE IN DATA SCIENCE


# Let's use a pie chart to understand the percentage of experience levels and company size


# Preparing data for pie chart


exp_level = data['Experience Level'].value_counts()

comp_size = data['Company Size'].value_counts()


# Implementing style and setting figure size

plt.style.use('fivethirtyeight')

plt.figure(figsize=(12,10))
```

```
# Adding a title for the plot
plt.suptitle('Experience Level and Company Size of Data Science Jobs',
             y=0.8,ha='center',va='center',
            fontsize=20.0, color='black')

# Creating First subplot
plt.subplot(1,2,1)
explode = [0,0,0.15,0] # Variable for explode parameter of pie()

# Function to display Experience level pie chart
plt.pie(exp_level, labels = exp_level.index, autopct = '%1.2f%%',
        pctdistance = 0.8, explode = explode, colors = sns.color_palette("Set2"),
        textprops={'fontsize':17})

# Adding title to the first subplot
plt.title('Experience Level',fontsize=18.0, color='black')

# Creating a white coloured circle to create a donut chart
hole = plt.Circle((0,0), 0.70, facecolor = 'white')
plt.gcf().gca().add_artist(hole)

# Creating second subplot
plt.subplot(1,2,2)
explode = [0,0,0.15]
```

```
# Function to display Company size pie chart
plt.pie(comp_size, labels = comp_size.index, autopct = '%1.2f%%',
        pctdistance = 0.8, explode = explode, colors = sns.color_palette("Set2"),
        textprops={'fontsize':17})

# Adding title to second subplot
plt.title('Company Size',fontsize=18.0, color='black')

# Converting to donut chart
hole = plt.Circle((0,0), 0.70, facecolor = 'white')
plt.gcf().gca().add_artist(hole)

plt.tight_layout()
plt.show()

#-----------------------------------VIZ 4-------------------------------------------#

# COUNTRIES HAVING MOST NUMBER OF DATA SCIENCE COMPANIES AND EMPLOYEES

# We will use a grouped bar chart

# Preparing data for grouped bar chart
location = pd.DataFrame(data['Company Location'].value_counts()) # New variable to store the location data
```

```
location['Employee Location'] = data['Employee Residence'].value_counts() # Creating new column that stores employee location

location = location.rename(columns = {'count':'Number of Companies',

                                      'Employee Location':'Number of Employees'}) # Renaming the columns



location = location.reset_index() # Resetting the index

sorter = location['Company Location'][0:10].value_counts().index # Variable to store Country names for x-axis labels

x = np.arange(len(sorter)) # creating an array of length of sorter for iterating the xticks labels



# Plotting the bar chart



# Setting style and figure size

plt.style.use('fivethirtyeight')

fig, ax = plt.subplots(1, figsize=(8,6))



# Using plt.bar() to create the plot

width = 0.45



# Bar chart indicating countries with most number of companies

ax.bar(x - width/2, location['Number of Companies'][0:10], width, label='No. of Companies',

    color = 'red', alpha=0.4)



# Bar chart indicating countries with most number of employees

ax.bar(x + width/2, location['Number of Employees'][0:10], width, label='No. of Employees',

    color = 'blue', alpha=0.4)
```

```
# Setting the x-axis labels using sorter variable created earlier
plt.xticks([i for i in range(len(sorter))],
           ['US', 'UK', 'CAN', 'GER', 'ESP', 'IND','FRA','AUS','PRT','NED'],
          color='black')
plt.yticks(color='black')

# Setting labels and title & legend
plt.title('Countries with most employees and companies', fontsize=18, color='black')
plt.xlabel('\nCountries\n', fontsize=16, color='black')
plt.ylabel('\n Count \n', fontsize=16, color='black')
plt.grid(False)
ax.legend(fontsize=14)

plt.show()



#----------------------------------------VIZ 5-----------------------------------------------#



# SALARY COMAPRISON OF DIFFERENT JOB TITLES FROM 2020 TO 2023

# For this, we are going to create a subplot of four line charts

# Preparing the data for the subplot
```

```
salary = data[['Job Title', 'Salary in USD', 'Year' ]] # Creating a subset of original data

salary = salary.groupby(['Job Title', 'Year']).mean() # Groupby average Salary per Job Title

salary = salary.reset_index() # Resetting the index

salary.sort_values(by = 'Salary in USD', ascending=False) #Sorting the df based on highest average salary

list1 = ['Data Engineer',
        'Data Scientist',
        'Data Analyst',
        'Machine Learning Engineer'] # Generating list of required Job Titles


salary1 = salary[salary['Job Title'].isin(list1)] # Creating a subset of required Job Titles

salary1.sort_values(by='Job Title') # Sorting by Job Titles


# Now we will create a variable for each of the five job title for our subplots

dat_anal = salary1[salary1['Job Title'] == 'Data Analyst']

dat_anal = dat_anal.reset_index()

dat_anal = dat_anal.drop('index',axis=1)


dat_eng = salary1[salary1['Job Title'] == 'Data Engineer']

dat_eng = dat_eng.reset_index()

dat_eng = dat_eng.drop('index',axis=1)


ml_eng = salary1[salary1['Job Title'] == 'Machine Learning Engineer']

ml_eng = ml_eng.reset_index()

ml_eng = ml_eng.drop('index',axis=1)
```

```
dat_sci = salary1[salary1['Job Title'] == 'Data Scientist']

dat_sci = dat_sci.reset_index()

dat_sci = dat_sci.drop('index',axis=1)


# Creating Subplots


#List for xticks labels

years = [2020, 2021, 2022, 2023]


# Plot style, figure size & subplot title

plt.style.use('fivethirtyeight')

fig, ax = plt.subplots(2, 2, sharey=True, figsize=(10,8), )

plt.suptitle('Salary Comparison of Different Job Titles from 2020 to 2023',

        fontsize=18, color='black', ha='center', va='center')

plt.grid(visible=None)


# Subplot no.1 - line graph highlighting average salary of data scientist

ax[0,0].plot('Year','Salary in USD', data = dat_anal, linestyle = 'dashed', c='grey', linewidth=1)

ax[0,0].plot('Year','Salary in USD', data = dat_eng, linestyle = 'dashed', c='grey', linewidth=1)

ax[0,0].plot('Year','Salary in USD', data = ml_eng, linestyle = 'dashed', c='grey', linewidth=1)

ax[0,0].plot('Year','Salary in USD', data = dat_sci, linestyle = 'solid', c='royalblue', linewidth=2, marker='o')

ax[0,0].set_title('Data Scientist', fontsize=14, color='black')

ax[0,0].set_xticks(years)

ax[0,0].grid(False)
```

```
# Subplot no.2 - line graph highlighting average salary of ML Engineer
ax[0,1].plot('Year','Salary in USD', data = dat_anal, linestyle = 'dashed', c='grey', linewidth=1)
ax[0,1].plot('Year','Salary in USD', data = dat_eng, linestyle = 'dashed', c='grey', linewidth=1)
ax[0,1].plot('Year','Salary in USD', data = ml_eng, linestyle = 'solid', c='royalblue', linewidth=2, marker='o')
ax[0,1].plot('Year','Salary in USD', data = dat_sci, linestyle = 'dashed', c='grey', linewidth=1)
ax[0,1].set_title('Machine Learning Engineer', fontsize=14, color='black')
ax[0,1].set_xticks(years)
ax[0,1].grid(False)

# Subplot no.3 - line graph highlighting average salary of data engineer
ax[1,0].plot('Year','Salary in USD', data = dat_anal, linestyle = 'dashed', c='grey', linewidth=1)
ax[1,0].plot('Year','Salary in USD', data = dat_eng, linestyle = 'solid', c='royalblue', linewidth=2, marker='o')
ax[1,0].plot('Year','Salary in USD', data = ml_eng, linestyle = 'dashed', c='grey', linewidth=1)
ax[1,0].plot('Year','Salary in USD', data = dat_sci, linestyle = 'dashed', c='grey', linewidth=1)
ax[1,0].set_title('Data Engineer', fontsize=14, color='black')
ax[1,0].set_xticks(years)
ax[1,0].grid(False)

# Subplot no.4 - line graph highlighting average salary of data analyst
ax[1,1].plot('Year','Salary in USD', data = dat_anal, linestyle = 'solid', c='royalblue', linewidth=2, marker='o')
ax[1,1].plot('Year','Salary in USD', data = dat_eng, linestyle = 'dashed', c='grey', linewidth=1)
ax[1,1].plot('Year','Salary in USD', data = ml_eng, linestyle = 'dashed', c='grey', linewidth=1)
ax[1,1].plot('Year','Salary in USD', data = dat_sci, linestyle = 'dashed', c='grey', linewidth=1)
ax[1,1].set_title('Data Analyst', fontsize=14, color='black')
ax[1,1].set_xticks(years)
```

```
ax[1,1].grid(False)



# Setting a common Y-axis label

fig.supylabel('Salary in USD', fontsize=16, color='black')

fig.tight_layout()

plt.show()

#---------------------------------------THANK YOU!-----------------------------------------------#
```