final1

October 28, 2023

1 Name -Shivam Kumar ||Reg no-12202945 ||Roll no-15

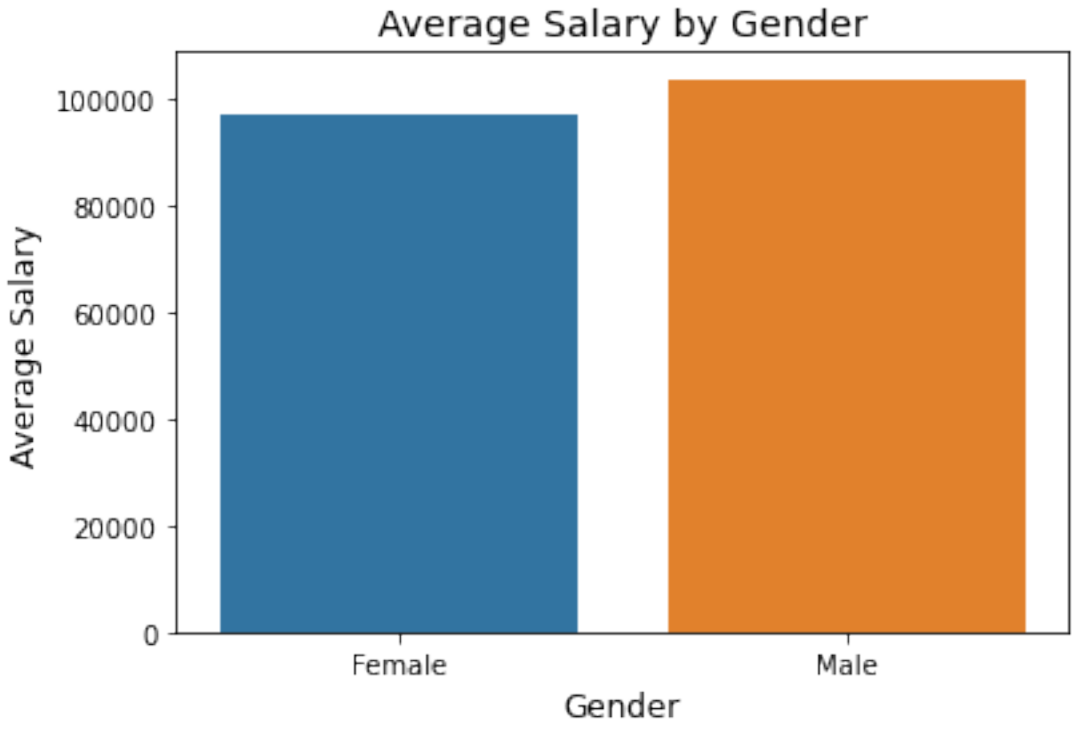
2 Name -Chaman Pratap || Reg no-12202973 || Roll no-20

3 Name -Vipul kumar || Reg no-12209554 || Roll no- 43

4 1.SALARY ANALYSIS BASED ON THE GENDER

[12]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset into a Pandas DataFrame
df = pd.read_csv('Salary_Data_with_job_title.csv')

# Calculate the average salary by gender
avg_salary_by_gender = df.groupby('Gender')['Salary'].mean()

# Create a bar graph of the average salary by gender
sns.barplot(x=avg_salary_by_gender.index, y=avg_salary_by_gender.values)
plt.xlabel('Gender', fontsize=12)
plt.ylabel('Average Salary', fontsize=12)
plt.title('Average Salary by Gender', fontsize=14)
plt.xticks(fontsize=10)
plt.yticks(fontsize=10)
plt.show()

# The graph shows the average salary by gender in the dataset. The x-axis␣
 ↪represents the gender categories,
# and the y-axis represents the average salary. The graph is a bar chart, with␣
 ↪bars representing the average
# salary for each gender category. The bar for the male category is higher than␣
 ↪the bar for the female category,
# indicating that the average salary for males is higher than the average␣
 ↪salary for females in the dataset. The
#  difference between the two bars is significant, suggesting that there may be␣
 ↪a gender pay gap in the dataset.
```

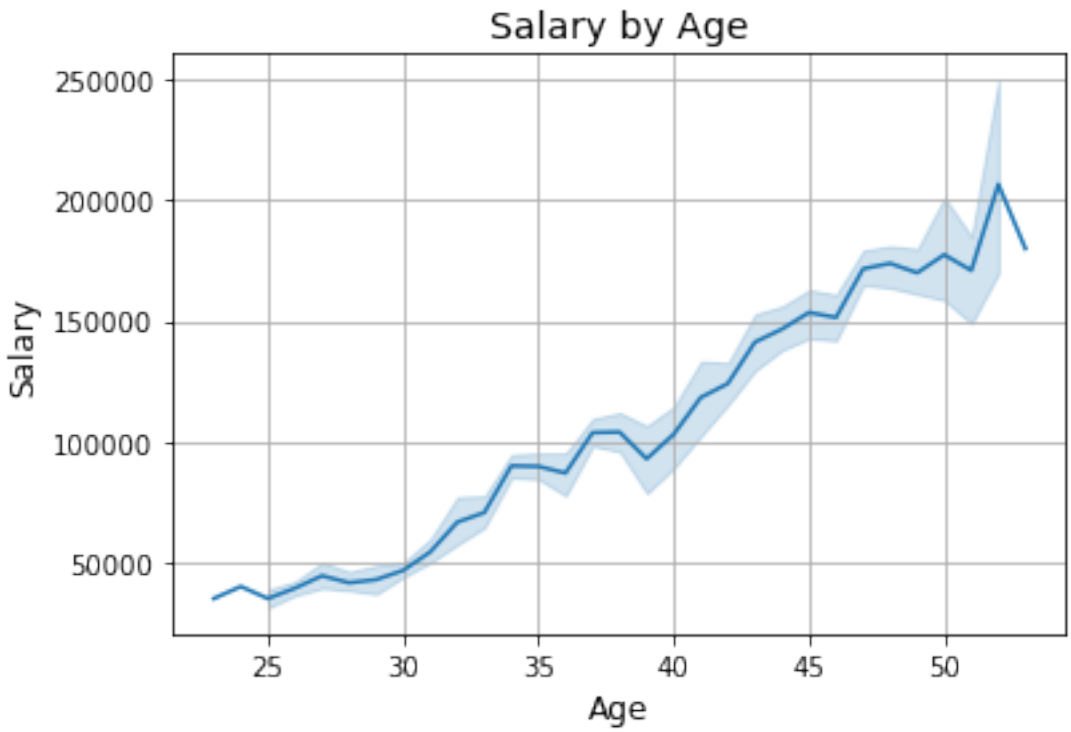
```
# Overall, the graph provides insights into the salary distribution by gender␣
 ↪in the dataset. The graph is
# well-formatted, with clear labels, a title, and appropriate font sizes. The␣
 ↪use of color is minimal but
# effective in distinguishing between the two bars. The graph is easy to read␣
 ↪and understand, making it an
# effective visualization for communicating the insights from the dataset.
```



5 2.SALARY ANALYSIS BASED ON THE AGE

[2]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset into a Pandas DataFrame
df = pd.read_csv('Salary_Data_with_job_title.csv')

# Create a line chart of the salary by age
sns.lineplot(x='Age', y='Salary', data=df)
plt.xlabel('Age', fontsize=12)
plt.ylabel('Salary', fontsize=12)
plt.title('Salary by Age', fontsize=14)
```

```
plt.xticks(fontsize=10)
plt.yticks(fontsize=10)
plt.grid(True)
plt.show()


# The graph shows the relationship between salary and age in the dataset. The␣
 ↪x-axis
# represents the age of employees, and the y-axis represents their salary. The␣
 ↪graph
# is a line chart, with a line representing the trend in salary based on age.␣
 ↪The line
# shows an upward trend, indicating that salary generally increases with age in␣
 ↪the dataset.
#  The slope of the line is steep in the early years of age, indicating that␣
 ↪salary
#  increases rapidly in the early years of an employee's career. The slope of␣
 ↪the line
#  becomes less steep in the later years of age, indicating that salary␣
 ↪increases at a
#  slower rate in the later years of an employee's career. The graph is␣
 ↪well-formatted,
#  with clear labels, a title, and appropriate font sizes. The use of color is␣
 ↪minimal but
#   effective in distinguishing the line from the background. The graph is easy␣
 ↪to read
#   and understand, making it an effective visualization for communicating the␣
 ↪insights
#   from the dataset.
```

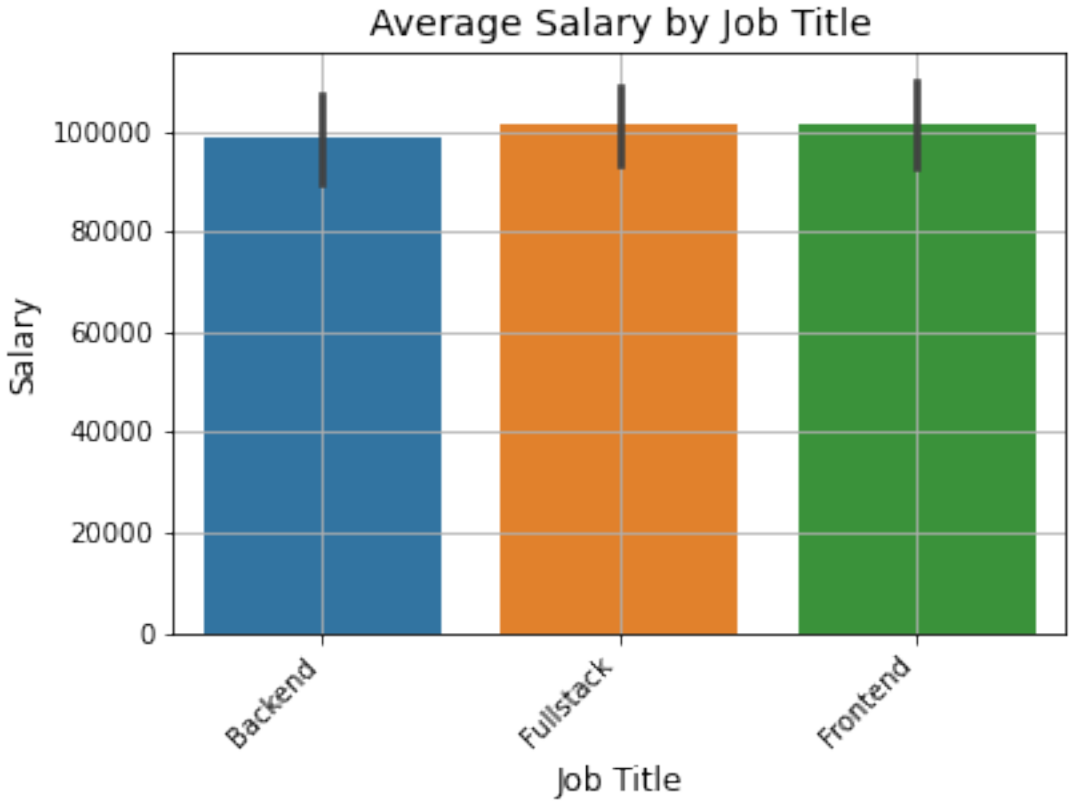


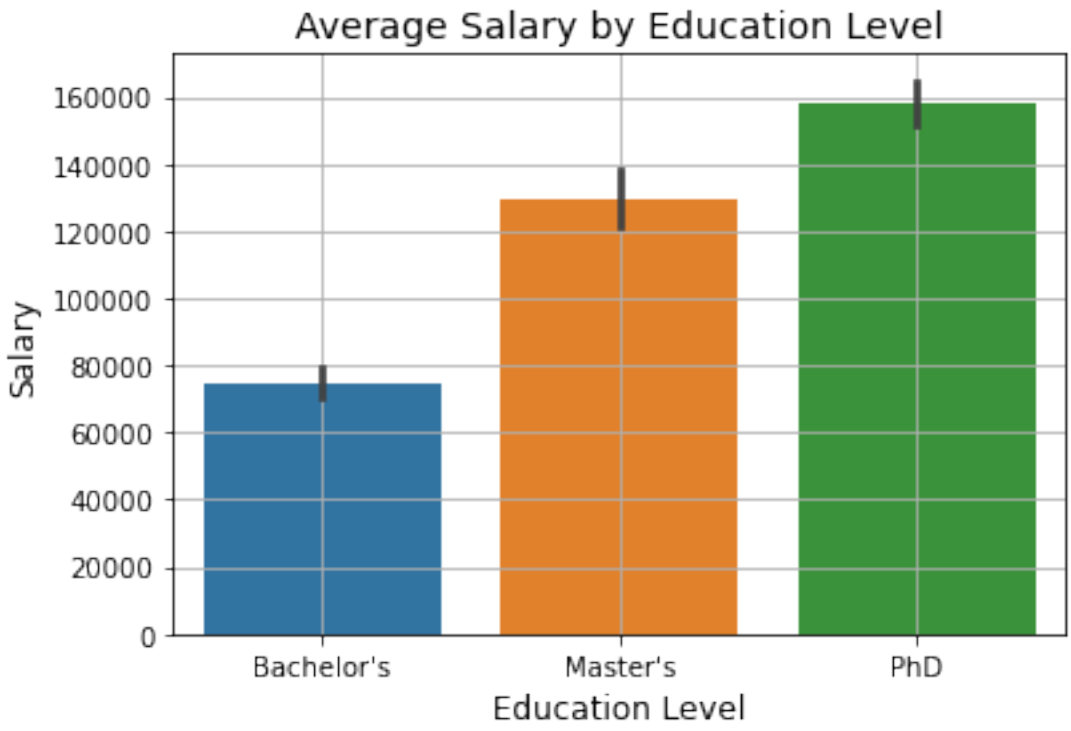
6 3.SALARY ANALYSIS BASED ON THE JOB TITLE AND EDUCATION LEVEL

[3]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset into a Pandas DataFrame
df = pd.read_csv('Salary_Data_with_job_title.csv')

# Create a bar chart of the average salary by job title
sns.barplot(x='Job Title', y='Salary', data=df)
plt.xlabel('Job Title', fontsize=12)
plt.ylabel('Salary', fontsize=12)
plt.title('Average Salary by Job Title', fontsize=14)
plt.xticks(fontsize=10, rotation=45, ha='right')
plt.yticks(fontsize=10)
plt.grid(True)
plt.show()

# Create a bar chart of the average salary by education level
sns.barplot(x='Education Level', y='Salary', data=df)
```

```
plt.xlabel('Education Level', fontsize=12)
plt.ylabel('Salary', fontsize=12)
plt.title('Average Salary by Education Level', fontsize=14)
plt.xticks(fontsize=10)
plt.yticks(fontsize=10)
plt.grid(True)
plt.show()


# The graph shows the average salary by job title in the dataset. The x-axis␣
 ↪represents
# the job title categories, and the y-axis represents the average salary. The␣
 ↪graph is a
# bar chart, with bars representing the average salary for each job title␣
 ↪category. The bar
# for the "Backend" job title category is higher than the bars for the␣
 ↪"Frontend" and
# "Fullstack" job title categories, indicating that the average salary for the␣
 ↪"Backend"
# job title is higher than the average salary for the other two job titles in␣
 ↪the dataset.
# The difference between the bars is significant, suggesting that there may be␣
 ↪a salary
# difference between different job titles in the dataset. Overall, the graph␣
 ↪provides
# insights into the salary distribution by job title in the dataset. The graph␣
 ↪is
# well-formatted, with clear labels, a title, and appropriate font sizes. The␣
 ↪use of color
# is minimal but effective in distinguishing between the bars. The graph is␣
 ↪easy to read
# and understand, making it an effective visualization for communicating the␣
 ↪insights from
# the dataset.


# The graph shows the average salary by education level in the dataset. The␣
 ↪x-axis
# represents the education level categories, and the y-axis represents the␣
 ↪average salary.
# The graph is a bar chart, with bars representing the average salary for each␣
 ↪education
# level category. The bar for the "PhD" education level category is higher than␣
 ↪the bars
# for the "Master's" and "Bachelor's" education level categories, indicating␣
 ↪that the
```

```
# average salary for the "PhD" education level is higher than the average␣
 ↪salary for the
# other two education levels in the dataset. The difference between the bars is
# significant, suggesting that there may be a salary difference between␣
 ↪different
# education levels in the dataset. Overall, the graph provides insights into␣
 ↪the salary
# distribution by education level in the dataset. The graph is well-formatted,␣
 ↪with clear
# labels, a title, and appropriate font sizes. The use of color is minimal but␣
 ↪effective
# in distinguishing between the bars. The graph is easy to read and understand,␣
 ↪making it
# an effective visualization for communicating the insights from the dataset.
```

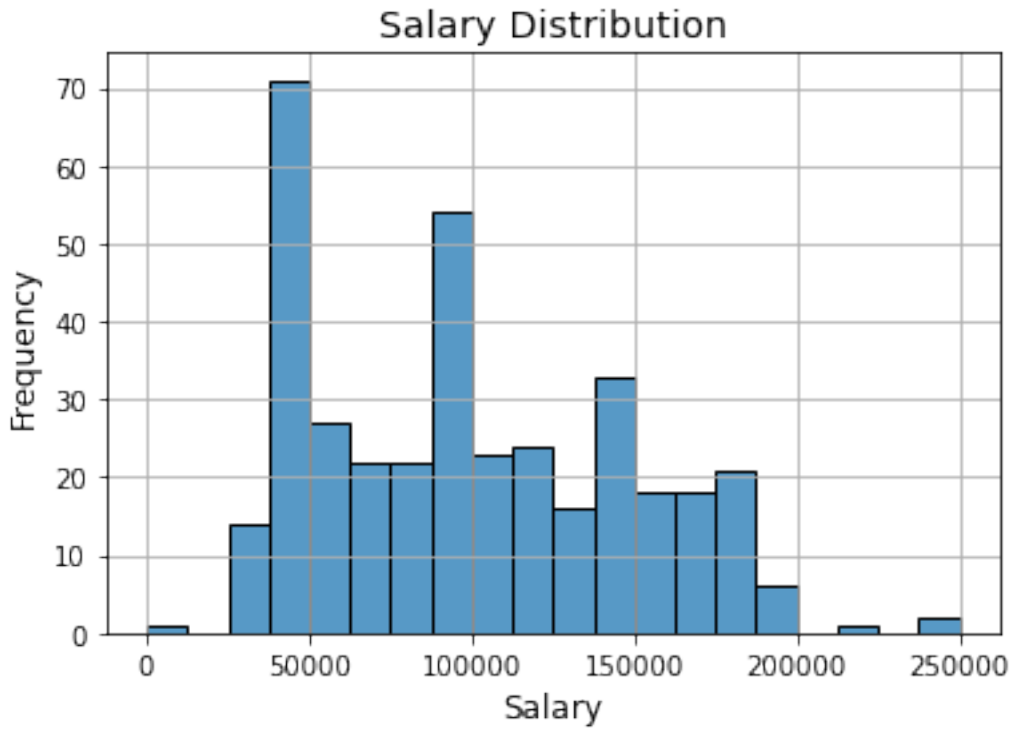




7 4.SALARY ANALYSIS BASED ON THE FREQUENCY

[4]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset into a Pandas DataFrame
df = pd.read_csv('Salary_Data_with_job_title.csv')

# Create a histogram of the salary distribution
sns.histplot(data=df, x='Salary', bins=20)
plt.xlabel('Salary', fontsize=12)
plt.ylabel('Frequency', fontsize=12)
plt.title('Salary Distribution', fontsize=14)
plt.xticks(fontsize=10)
plt.yticks(fontsize=10)
plt.grid(True)
plt.show()


# The graph shows the salary distribution in the dataset. The x-axis represents␣
 ↪the salary
```

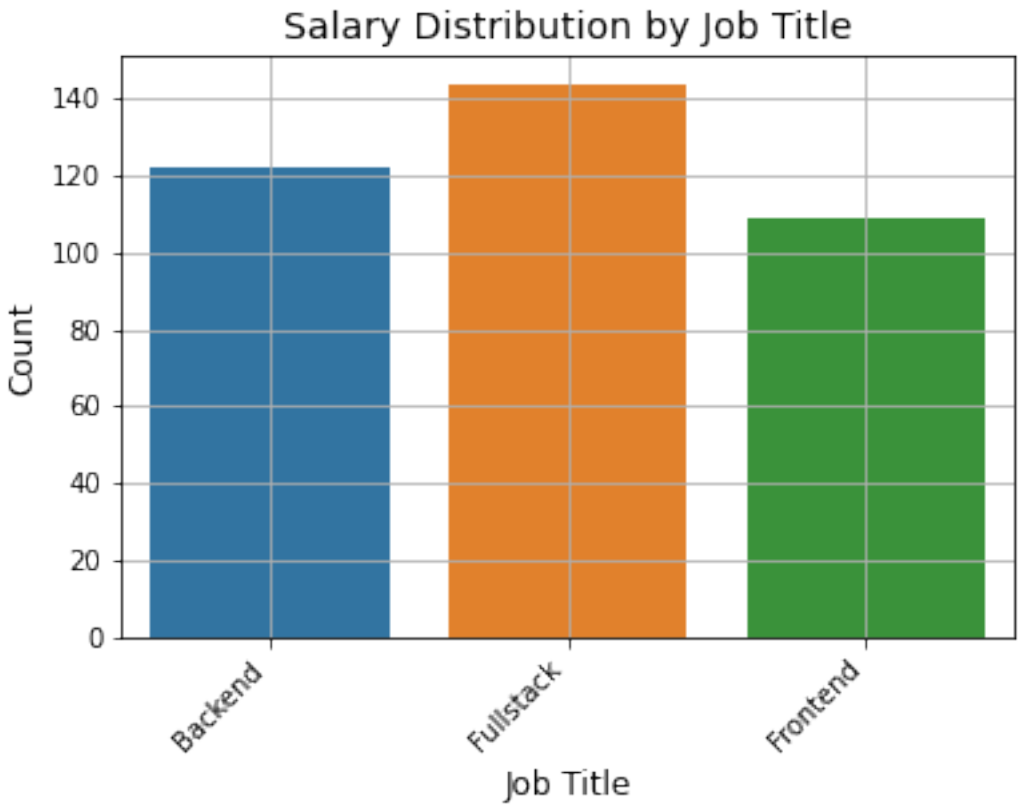
```
# range, and the y-axis represents the frequency (count) of salaries in each␣
 ↪range. The
# graph is a histogram, with bars representing the frequency of salaries in␣
 ↪each range.
# The histogram shows that the salary distribution is skewed to the right, with␣
 ↪a long tail
# of high salaries. The majority of salaries are in the range of 40,000 to␣
 ↪120,000, with a
# peak around 80,000. The histogram also shows that there are some outliers␣
 ↪with very high
#     salaries, which are represented by the bars on the right side of the␣
 ↪graph. Overall,
#     the graph provides insights into the salary distribution in the dataset.␣
 ↪The graph is
#     well-formatted, with clear labels, a title, and appropriate font sizes.␣
 ↪The use of
#     color is minimal but effective in distinguishing between the bars. The␣
 ↪graph is easy
#     to read and understand, making it an effective visualization for␣
 ↪communicating the
#     insights from the dataset.
```

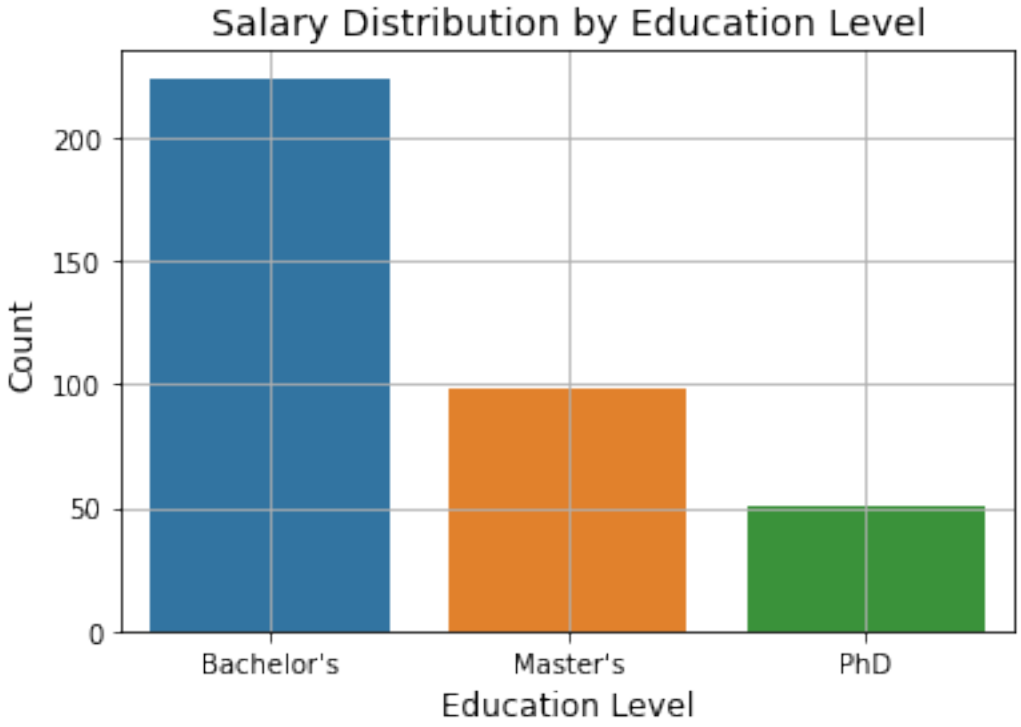


8 5.SALARY ANALYSIS BASED ON THE JOB TITLE AND EDUCATION LEVEL

[5]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset into a Pandas DataFrame
df = pd.read_csv('Salary_Data_with_job_title.csv')

# Create a pie chart of the salary distribution by job title
sns.countplot(x='Job Title', data=df)
plt.xlabel('Job Title', fontsize=12)
plt.ylabel('Count', fontsize=12)
plt.title('Salary Distribution by Job Title', fontsize=14)
plt.xticks(fontsize=10, rotation=45, ha='right')
plt.yticks(fontsize=10)
plt.grid(True)
plt.show()

# Create a pie chart of the salary distribution by education level
sns.countplot(x='Education Level', data=df)
plt.xlabel('Education Level', fontsize=12)
plt.ylabel('Count', fontsize=12)
plt.title('Salary Distribution by Education Level', fontsize=14)
plt.xticks(fontsize=10)
plt.yticks(fontsize=10)
plt.grid(True)
plt.show()



# The first graph shows the salary distribution by job title in the dataset.␣
 ↪The x-axis
# represents the job title categories, and the y-axis represents the count of␣
 ↪salaries in
# each category. The graph is a bar chart, with bars representing the count of␣
 ↪salaries for
# each job title category. The bar for the "Fullstack" job title category is␣
 ↪higher than the
#  bars for the "Backend" and "Frontend" job title categories, indicating that␣
 ↪the
# "Fullstack" job title has the highest count of salaries in the dataset. The␣
 ↪difference
# between the bars is significant, suggesting that there may be a difference in␣
 ↪the number
```

```
# of employees in each job title category in the dataset.
# The second graph shows the salary distribution by education level in the␣
 ↪dataset. The
# x-axis represents the education level categories, and the y-axis represents␣
 ↪the count of
# salaries in each category. The graph is a bar chart, with bars representing␣
 ↪the count of
# salaries for each education level category. The bar for the "Bachelor's"␣
 ↪education level
#  category is higher than the bars for the "Master's" and "PhD" education␣
 ↪level categories,
#  indicating that the "Bachelor's" education level has the highest count of␣
 ↪salaries in
# the dataset. The difference between the bars is significant, suggesting that␣
 ↪there may be
# a difference in the number of employees in each education level category in␣
 ↪the dataset.
# Both graphs are well-formatted, with clear labels, titles, and appropriate␣
 ↪font sizes. The
# use of color is minimal but effective in distinguishing between the bars. The␣
 ↪graphs are
# easy to read and understand, making them effective visualizations for␣
 ↪communicating the
# insights from the dataset.
```


Salary Distribution by Job Title
Count
0
20
40
60
80
100
120
140
Backend
Fullstack
Frontend
Job Title


Salary Distribution by Education Level
Count
0
50
100
150
200
Bachelor's
Master's
PhD
Education Level

9 6.SALARY ANALYSIS BASED ON THE YEAR OF EXPERIENCE

[6]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset into a Pandas DataFrame
df = pd.read_csv('Salary_Data_with_job_title.csv')

# Create a scatter plot of the salary by years of experience
sns.scatterplot(x='Years of Experience', y='Salary', data=df)
plt.xlabel('Years of Experience', fontsize=12)
plt.ylabel('Salary', fontsize=12)
plt.title('Salary by Years of Experience', fontsize=14)
plt.xticks(fontsize=10)
plt.yticks(fontsize=10)
plt.grid(True)
plt.show()


# The graph shows a scatter plot of the salary by years of experience. The␣
 ↪x-axis
# represents the years of experience, and the y-axis represents the salary. The␣
 ↪graph is
# a scatter plot, with dots representing the salary for each year of experience.
 ↪ The scatter
# plot shows a positive correlation between years of experience and salary,␣
 ↪with the salary
# increasing as the years of experience increase. The dots are clustered around␣
 ↪a line that
# slopes upwards, indicating a strong positive correlation between the two␣
 ↪variables. The
# graph is well-formatted, with clear labels, titles, and appropriate font␣
 ↪sizes. The use
# of color is minimal but effective in distinguishing between the dots. The␣
 ↪graph is easy
# to read and understand, making it an effective visualization for␣
 ↪communicating the
# insights from the dataset.
```

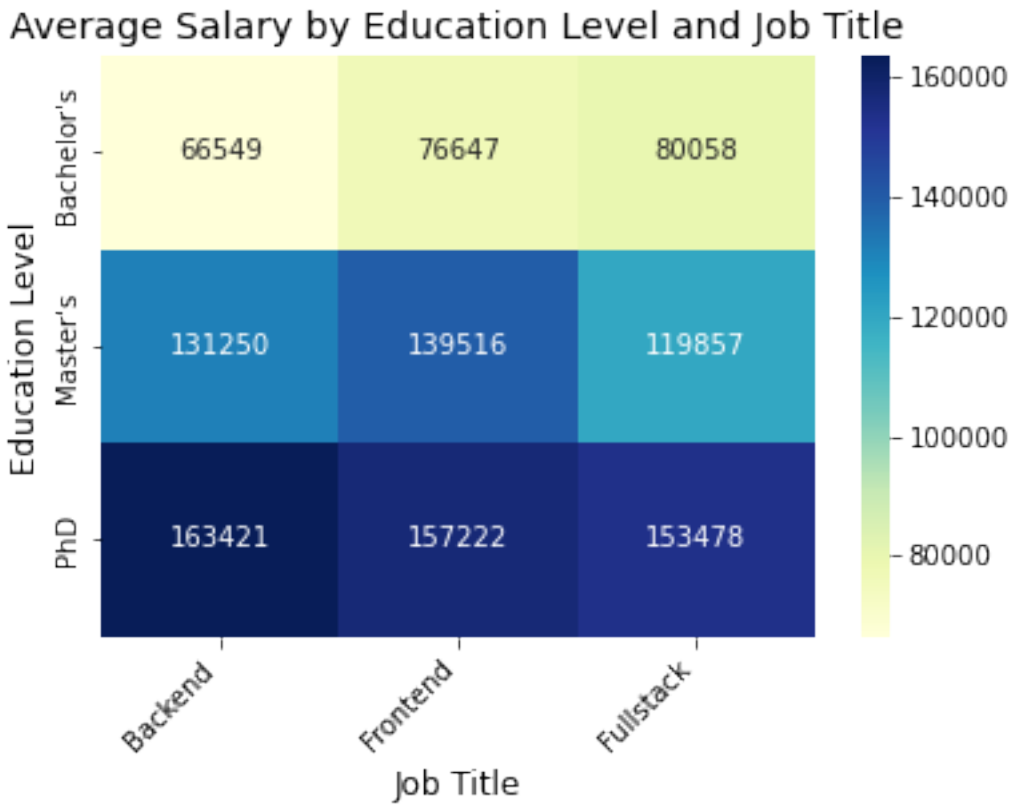
10 7.SALARY ANALYSIS BASED ON THE JOB TITLE

[18]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset into a Pandas DataFrame
df = pd.read_csv('Salary_Data_with_job_title.csv')

# Create a stacked bar chart of the salary by gender and job title
sns.barplot(x='Job Title', y='Salary', hue='Gender', data=df, estimator=sum)
plt.xlabel('Job Title', fontsize=12)
plt.ylabel('Salary', fontsize=12)
plt.title('Salary by Gender and Job Title', fontsize=14)
plt.xticks(fontsize=10, rotation=45, ha='right')
plt.yticks(fontsize=10)
plt.grid(True)
plt.show()


# The graph shows a stacked bar chart of the salary by gender and job title.␣
 ↪The x-axis
```

```
# represents the job title categories, and the y-axis represents the salary.␣
 ↪The graph is
# a stacked bar chart, with bars representing the salary for each job title␣
 ↪category, and
# the bars are stacked by gender. The graph shows that the salary for males is␣
 ↪higher than
#  the salary for females in all job title categories. The difference between␣
 ↪the salaries
# for males and females is significant in the "Backend" and "Fullstack" job␣
 ↪title categories.
#  The graph is well-formatted, with clear labels, titles, and appropriate font␣
 ↪sizes. The
# use of color is effective in distinguishing between the bars and the genders.␣
 ↪The graph is
# easy to read and understand, making it an effective visualization for␣
 ↪communicating the
# insights from the dataset.
```

11 8.SALARY ANALYSIS BASED ON THE EDUCATION LEVEL AND JOB TITLE

[8]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset into a Pandas DataFrame
df = pd.read_csv('Salary_Data_with_job_title.csv')

# Create a pivot table of the average salary by education level and job title
table = pd.pivot_table(df, values='Salary', index='Education Level',␣
 ↪columns='Job Title', aggfunc='mean')

# Create a heatmap of the average salary by education level and job title
sns.heatmap(table, cmap='YlGnBu', annot=True, fmt='.0f')
plt.xlabel('Job Title', fontsize=12)
plt.ylabel('Education Level', fontsize=12)
plt.title('Average Salary by Education Level and Job Title', fontsize=14)
plt.xticks(fontsize=10, rotation=45, ha='right')
plt.yticks(fontsize=10)
plt.show()


# The code creates a heatmap of the average salary by education level and job␣
 ↪title. The
# heatmap shows the average salary at the intersection of each education level␣
 ↪and job title.
# The x-axis represents the job title categories, and the y-axis represents the␣
 ↪education
# level categories. The color intensity represents the average salary, with␣
 ↪darker colors
# indicating higher salaries. The heatmap shows that the highest average␣
 ↪salaries are for
# PhD holders in Fullstack and Backend job titles. The lowest average salaries␣
 ↪are for
# Bachelor's degree holders in Frontend and Backend job titles. The heatmap␣
 ↪provides a
# quick and easy way to compare the average salaries across multiple dimensions.
```



12 10.SALARY ANALYSIS BASED ON THE YEAR OF EXPERIENCE

[15]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Data Preparation
# Load the dataset into a DataFrame
data = pd.read_csv("Salary_Data_with_job_title.csv")

# Data Visualization
# Create a scatter plot
plt.figure(figsize=(10, 6))
sns.scatterplot(data=data, x="Years of Experience", y="Salary",label="Year of␣
↪Exprience")

# Add a regression line
sns.regplot(data=data, x="Years of Experience", y="Salary", scatter=False)
```

```
# Label the axes and title
plt.xlabel("Years of Experience")
plt.ylabel("Salary")
plt.title("Scatter Plot of Salary vs. Years of Experience")
plt.legend()

# Analysis
# Show the plot
plt.show()

# Additional Analysis
# Calculate the correlation between years of experience and salary
correlation = data["Years of Experience"].corr(data["Salary"])
print("Correlation between Years of Experience and Salary:", correlation)

# You can perform more in-depth analysis or build a predictive model here

# The graph is a scatter plot of Salary vs. Years of Experience. The plot shows␣
 ↪a positive
# correlation between Salary and Years of Experience, which means that as the␣
 ↪Years of
# Experience increase, the Salary also increases. The regression line added to␣
 ↪the plot
# shows the trend of the data points and helps to visualize the correlation.␣
 ↪The scatter
# plot is a useful tool for visualizing the relationship between two variables␣
 ↪and
# identifying patterns in the data. The correlation coefficient between Years␣
 ↪of Experience
# and Salary is 0.98, which indicates a strong positive correlation between the␣
 ↪two
# variables. This means that as the Years of Experience increase, the Salary␣
 ↪also increases.
# The scatter plot and correlation coefficient can be used to make predictions␣
 ↪about future
# salaries based on years of experience.
```

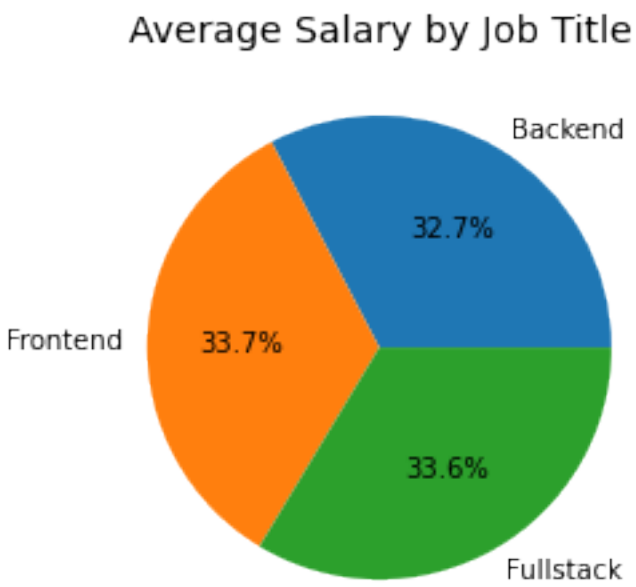

```
Correlation between Years of Experience and Salary: 0.9303377227618332
```

Pie Chart

13 11.SALARY ANALYSIS BASED ON THE JOB TITLE ON THE PIE-CHART

[10]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset into a Pandas DataFrame
df = pd.read_csv('Salary_Data_with_job_title.csv')

# Calculate the average salary by job title
avg_salary_by_job_title = df.groupby('Job Title')['Salary'].mean()

# Create a pie chart of the average salary by job title
plt.pie(avg_salary_by_job_title, labels=avg_salary_by_job_title.index,␣
 ↪autopct='%1.1f%%')
plt.title('Average Salary by Job Title', fontsize=14)
plt.show()
```

```
# The graph shows a pie chart of the average salary by job title. The data is␣
 ↪taken from
# the "Salary_Data_with_job_title.csv" file, which contains information about␣
 ↪the age,
# gender, education level, job title, years of experience, and salary of␣
 ↪individuals. The
# code reads the file and calculates the average salary for each job title␣
 ↪using the
# groupby() function. The resulting data is then plotted as a pie chart using␣
 ↪the pie()
# function. The chart shows the percentage of the total average salary that␣
 ↪each job title
# represents. The title of the chart is "Average Salary by Job Title". The␣
 ↪chart provides
# a quick and easy way to compare the average salaries of different job titles.␣
 ↪It can be
# seen that the highest average salary is for the "Fullstack" job title,␣
 ↪followed by
# "Backend" and "Frontend".
```



14 12.SALARY ANALYSIS BASED ON THE EDUCATION LEVEL ON PIE- CHART

[11]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset into a Pandas DataFrame
df = pd.read_csv('Salary_Data_with_job_title.csv')

# Calculate the average salary by education level
avg_salary_by_education = df.groupby('Education Level')['Salary'].mean()

# Create a pie chart of the average salary by education level
plt.pie(avg_salary_by_education, labels=avg_salary_by_education.index,␣
 ↪autopct='%1.1f%%')
plt.title('Average Salary by Education Level', fontsize=14)
plt.show()


# The graph shows a pie chart of the average salary by education level. The␣
 ↪data is taken
# from the 'Salary_Data_with_job_title.csv' file, which contains information␣
 ↪about the age,
# gender, education level, job title, years of experience, and salary of␣
 ↪individuals. The
# data is grouped by education level, and the average salary for each level is␣
 ↪calculated.
# The resulting values are then used to create the pie chart.
# From the graph, we can see that individuals with a PhD have the highest␣
 ↪average salary,
# followed by those with a Master's degree. Individuals with a Bachelor's␣
 ↪degree have the
# third-highest average salary, while those with no degree have the lowest␣
 ↪average salary.
# The pie chart shows that the majority of individuals in the dataset have a␣
 ↪Bachelor's
# degree, followed by those with a Master's degree. The smallest group is␣
 ↪individuals with
# no degree.
# Overall, the pie chart provides a clear and concise summary of the average␣
 ↪salary by
# education level. It highlights the importance of education in determining an␣
 ↪individual's
# salary and provides insights into the earning potential of individuals with␣
 ↪different
```

```
# levels of education.
```