Project Name - Data Analyst Jobs (ML _ FA _ DA projects) (Part 2)

Project Type - Data Analysis

Industry - Unified Mentor

Contribution - Individual

Member Name - Hare Krishana Mishra

Task - 2

Project Summary -

Project Description:

This project analyzes a dataset of over 2,000 job listings for Data Analyst positions collected from Glassdoor. It covers various attributes like salary estimates, location, company ratings, industry, job description, company size, and ownership type. The main goal is to uncover job market trends, evaluate salary ranges, and identify the factors that influence pay in the Data Analytics industry. It also includes predictive modeling to estimate salaries based on job and company features, enabling job seekers and recruiters to make informed decisions.

Objective:

- To analyze trends in Data Analyst job postings across different industries, sectors, and locations.
- To predict salary ranges based on job attributes like company rating, size, industry, and skills required.
- To provide actionable insights about company ratings, hiring patterns, and salary trends.
- To highlight top-paying sectors, industries, and locations for Data Analyst roles.

Key Project Details:

Dataset Source: Glassdoor job postings, >2,000 records, features like salary, location, company rating, job description, and ownership type.

Data Cleaning & Preprocessing: Removed duplicates, handled missing values, standardized column names, extracted salary ranges.

Exploratory Data Analysis (EDA):

Distribution of salaries, ratings, and company sizes.

Trends in job postings by industry, sector, and location.

Top industries and sectors hiring data analysts.

Feature Engineering:

Extracted technical skills (Python, Excel) from job descriptions.

Created Tech_Skills score.

Split location into City and State.

Visualization Insights:

Top 10 job titles by count.

Average salary by job title, company size, and sector.

Salary trends by location.

Model Development:

Trained a Random Forest Regressor to predict average salary.

Features: Rating, Tech_Skills, Size, Founded.

Key Findings:

Highest salaries often in California-based locations.

Top-paying sectors: Biotech & Pharmaceuticals, Real Estate, Art, Entertainment & Recreation.

Private companies dominate hiring.

Deployment: Model can be deployed with Streamlit or Flask for interactive predictions.

Let's Begin:-

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
import re
import plotly.graph_objects as go
import plotly.express as px
from plotly.subplots import make_subplots
warnings.filterwarnings('ignore')
```

In [2]:

```
data_analyst_jobs = pd.read_csv('/content/DataAnalyst.csv')
```

In [3]:

```
data_analyst_jobs = data_analyst_jobs.drop('Unnamed: 0',axis=1)
data_analyst_jobs = data_analyst_jobs.drop('Founded', axis=1)
data_analyst_jobs = data_analyst_jobs.drop('Competitors',axis=1)
print(f'Number of rows:{data_analyst_jobs.shape[0]};Number ofcolumns:{data_analyst_jobs.
```

```
Number of rows:2253;Number ofcolumns:13; No of missingvalues:1
```

Dataset Overview

In [4]:

```
data_analyst_jobs.head()
```

Out[4]:

	Job Title	Salary Estimate	Job Description	Rating	Company Name	Location	Headquarters
0	Data Analyst, Center on Immigration and Justic...	$37K–66K (Glassdoor est.)	Are you eager to roll up your sleeves and harn...	3.2	Vera Institute of Justice\n3.2	New York, NY	New York, NY
1	Quality Data Analyst	$37K–66K (Glassdoor est.)	Overview\n\nProvides analytical and technical ...	3.8	Visiting Nurse Service of New York\n3.8	New York, NY	New York, NY
2	Senior Data Analyst, Insights & Analytics Team...	$37K–66K (Glassdoor est.)	We're looking for a Senior Data Analyst who ha...	3.4	Squarespace\n3.4	New York, NY	New York, NY
3	Data Analyst	$37K–66K (Glassdoor est.)	Requisition NumberRR-0001939\nRemote:Yes\nWe c...	4.1	Celerity\n4.1	New York, NY	McLean, VA
4	Reporting Data Analyst	$37K–66K (Glassdoor est.)	ABOUT FANDUEL GROUP\n\nFanDuel Group is a worl...	3.9	FanDuel\n3.9	New York, NY	New York, NY

Quick view

In [5]:

```
data_analyst_jobs.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2253 entries, 0 to 2252
Data columns (total 13 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Job Title          2253 non-null   object 
 1   Salary Estimate    2253 non-null   object 
 2   Job Description    2253 non-null   object 
 3   Rating             2253 non-null   float64
 4   Company Name       2252 non-null   object 
 5   Location           2253 non-null   object 
 6   Headquarters       2253 non-null   object 
 7   Size               2253 non-null   object 
 8   Type of ownership  2253 non-null   object 
 9   Industry           2253 non-null   object 
 10  Sector             2253 non-null   object 
 11  Revenue            2253 non-null   object 
 12  Easy Apply         2253 non-null   object 
```

```
dtypes: float64(1), object(12)
memory usage: 228.9+ KB
```

Renaming Columns for Better Analysis

In [6]:

```
data_analyst_jobs.rename(columns={"Job Title": "job_title"},inplace=True)
```

In [7]:

```
data_analyst_jobs.rename(columns={"Salary Estimate":"salary_estimate"}, inplace=True)
data_analyst_jobs.rename(columns={"Job Description":"job_description"}, inplace=True)
data_analyst_jobs.rename(columns={"Company Name":"company_name"}, inplace=True)
data_analyst_jobs.rename(columns={"Location": "location"},inplace=True)
data_analyst_jobs.rename(columns={"Headquarters":"headquarters"}, inplace=True)
data_analyst_jobs.rename(columns={"Size": "size"},
inplace=True)
data_analyst_jobs.rename(columns={"Type of ownership":"type_of_ownership"}, inplace=True
data_analyst_jobs.rename(columns={"Industry": "industry"},inplace=True)
data_analyst_jobs.rename(columns={"Sector": "sector"},inplace=True)
data_analyst_jobs.rename(columns={"Revenue": "revenue"},inplace=True)
data_analyst_jobs.rename(columns={"Easy Apply": "easy_apply"},inplace=True)
```

In [8]:

```
data_analyst_jobs.head()
```

Out[8]:

	job_title	salary_estimate	job_description	Rating	company_name	location	headquart
0	Data Analyst, Center on Immigration and Justic...	$37K-66K (Glassdoor est.)	Are you eager to roll up your sleeves and harn...	3.2	Vera Institute of Justice\n3.2	New York, NY	New York,
1	Quality Data Analyst	$37K-66K (Glassdoor est.)	Overview\n\nProvides analytical and technical ...	3.8	Visiting Nurse Service of New York\n3.8	New York, NY	New York,
2	Senior Data Analyst, Insights & Analytics Team...	$37K-66K (Glassdoor est.)	We're looking for a Senior Data Analyst who ha...	3.4	Squarespace\n3.4	New York, NY	New York,
3	Data Analyst	$37K-66K (Glassdoor est.)	Requisition NumberRR-0001939\nRemote:Yes\nWe c...	4.1	Celerity\n4.1	New York, NY	McLean,
4	Reporting Data Analyst	$37K-66K (Glassdoor est.)	ABOUT FANDUEL GROUP\n\nFanDuel Group is a worl...	3.9	FanDuel\n3.9	New York, NY	New York,

Job Title

In [9]:

```
data_analyst_jobs['job_title'] =data_analyst_jobs['job_title'].replace(['Sr. Data Analys
```

In [10]:

```
data_analyst_jobs['job_title'] = data_analyst_jobs['job_title'].replace(['Data Analyst I
```

In [11]:

```
data_analyst_jobs['job_title'] =data_analyst_jobs['job_title'].replace(['Data Analyst II
```

In [12]:

```
# plot the most commmon types of jobs
to_plot = data_analyst_jobs.job_title.value_counts()[:5]
# ax = to_plot.plot(kind='bar',
color = sns.color_palette('Spectral')
to_plot
```

Out[12]:

job_title	count
Data Analyst	405
Senior Data Analyst	120
Junior Data Analyst	58
Business Data Analyst	28
Data Quality Analyst	17

dtype: int64

Salary Estimate and Trends

In [13]:

```
## Changing Salary column to int for better calculation
data_analyst_jobs[['MinSalary', 'MaxSalary']] =data_analyst_jobs['salary_estimate'].str.
```

In [14]:

```
data_analyst_jobs['MinSalary'] =pd.to_numeric(data_analyst_jobs['MinSalary'])
data_analyst_jobs['MaxSalary'] =pd.to_numeric(data_analyst_jobs['MaxSalary'])
```

changing format to float

In [15]:

```
data_analyst_jobs['MinSalary'] =data_analyst_jobs['MinSalary'].astype(float)
data_analyst_jobs['MaxSalary'] =data_analyst_jobs['MaxSalary'].astype(float)
data_analyst_jobs['average_salary'] =(data_analyst_jobs['MaxSalary'] +
data_analyst_jobs['MinSalary']) / 2
#drop salary estimate(unuseful column)
data_analyst_jobs.drop(['salary_estimate', 'MinSalary','MaxSalary'], axis=1, inplace=Tru
```

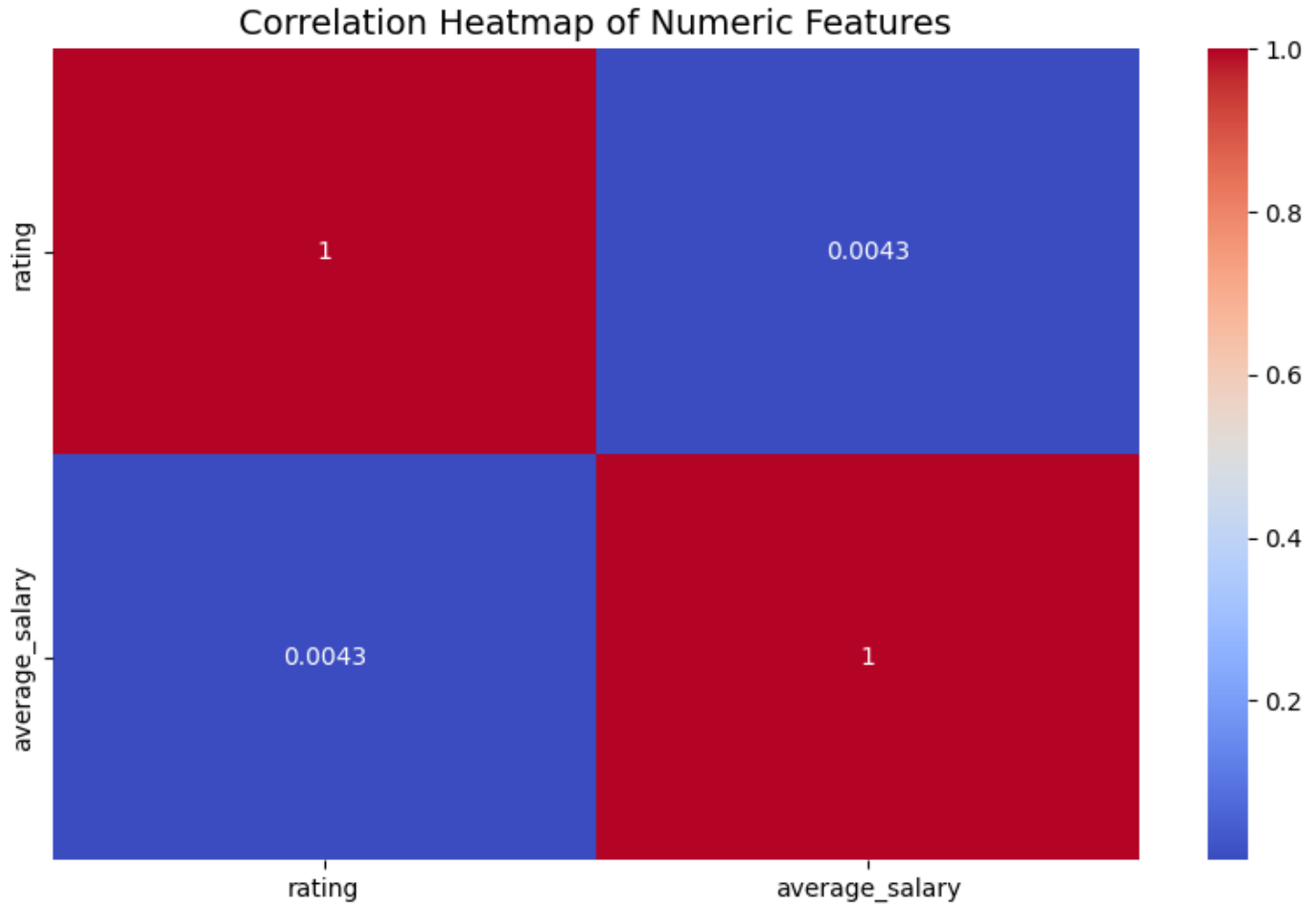
Graphs and Chart Visualizations

Correlation Heatmap of Numerical Features in Data Analyst Jobs Dataset

In [16]:

```
# Clean up column names (similar to your PDF renaming step)
data_analyst_jobs.columns = data_analyst_jobs.columns.str.strip().str.lower().str.replac
```

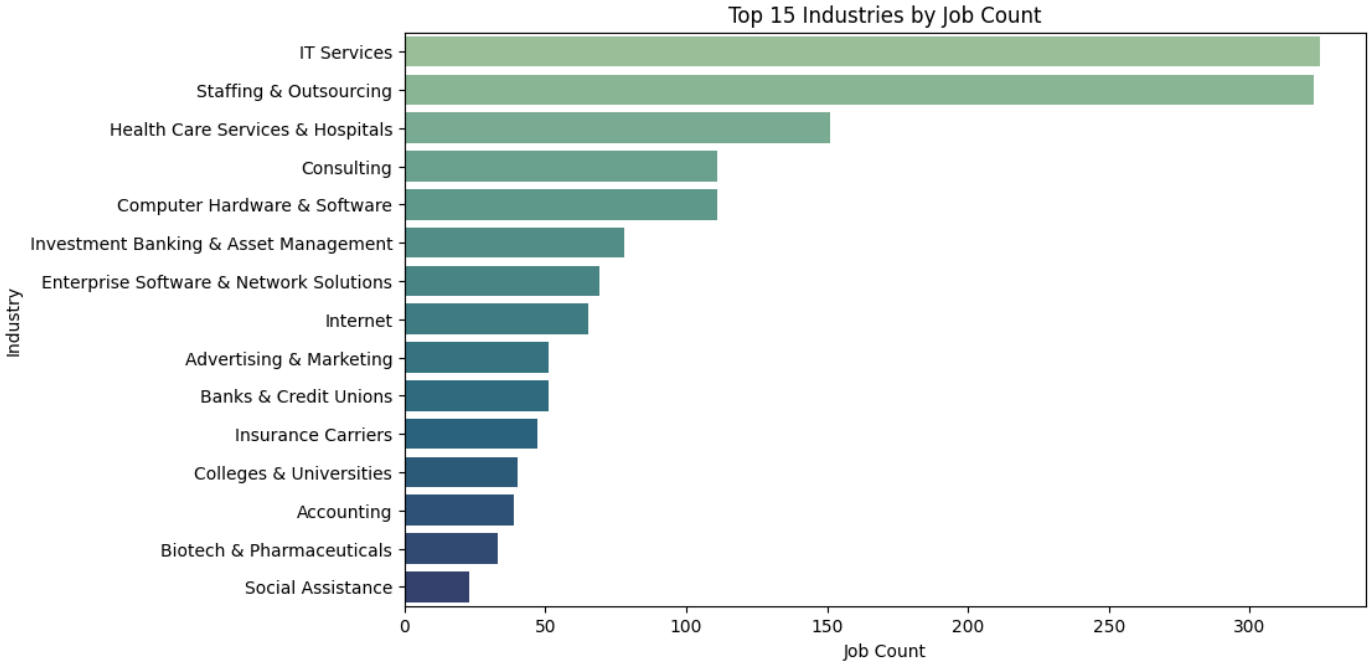
```
# --- 1. Correlation heatmap for numeric columns only ---
plt.figure(figsize=(10, 6))
sns.heatmap(data_analyst_jobs.select_dtypes(include='number').corr(), annot=True, cmap='
plt.title("Correlation Heatmap of Numeric Features", fontsize=14)
plt.show()
```



Top 15 Industries Hiring Data Analysts

In [17]:

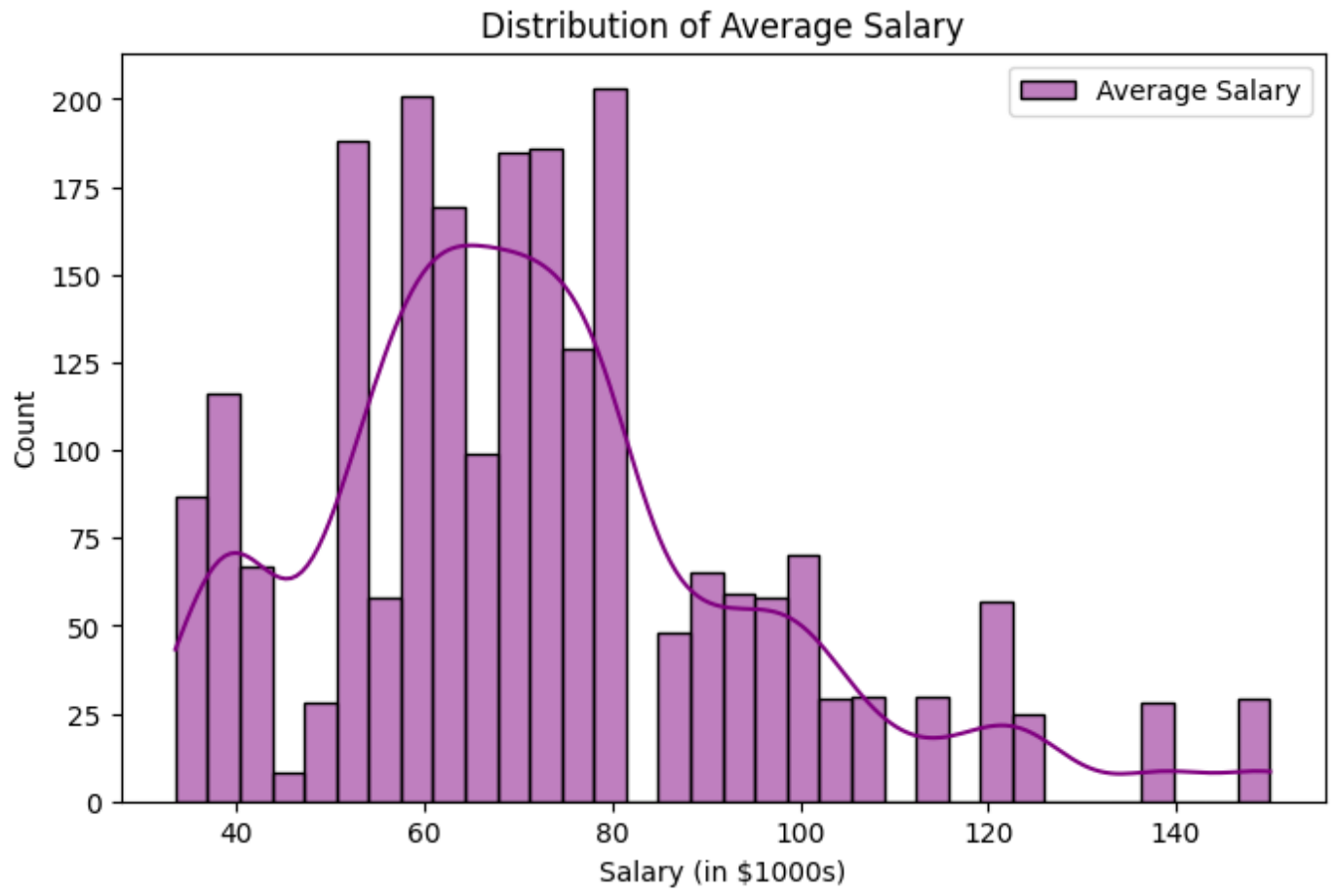
```
# --- 3. Top industries by job count ---
top_industries = data_analyst_jobs[data_analyst_jobs['industry'] != '-1']['industry'].va
plt.figure(figsize=(10, 6))
sns.barplot(y=top_industries.index, x=top_industries.values, palette='crest')
plt.title("Top 15 Industries by Job Count")
plt.xlabel("Job Count")
plt.ylabel("Industry")
plt.show()
```



Salary Distribution for Data Analyst Roles

In [18]:

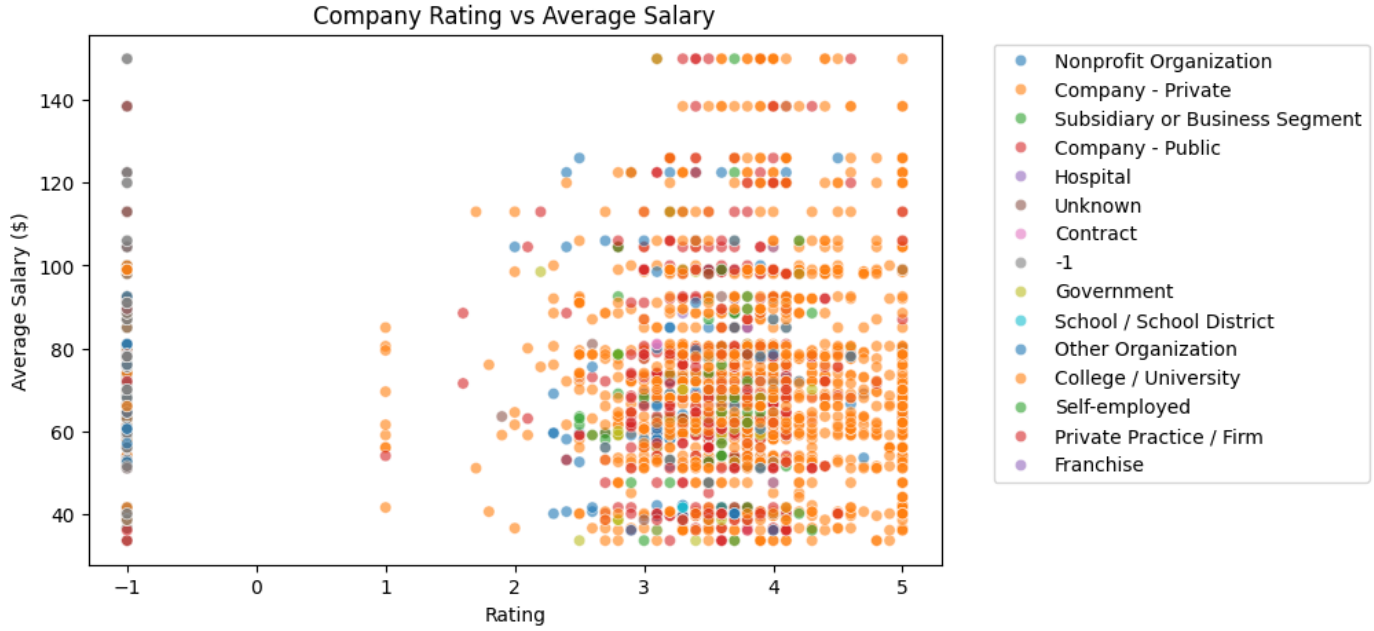
```
plt.figure(figsize=(8, 5))
sns.histplot(data_analyst_jobs['average_salary'], color='purple', label='Average Salary'
plt.legend()
plt.title('Distribution of Average Salary')
plt.xlabel('Salary (in $1000s)')
plt.show()
```



Relationship Between Company Rating and Average Salary by Ownership Type

In [19]:

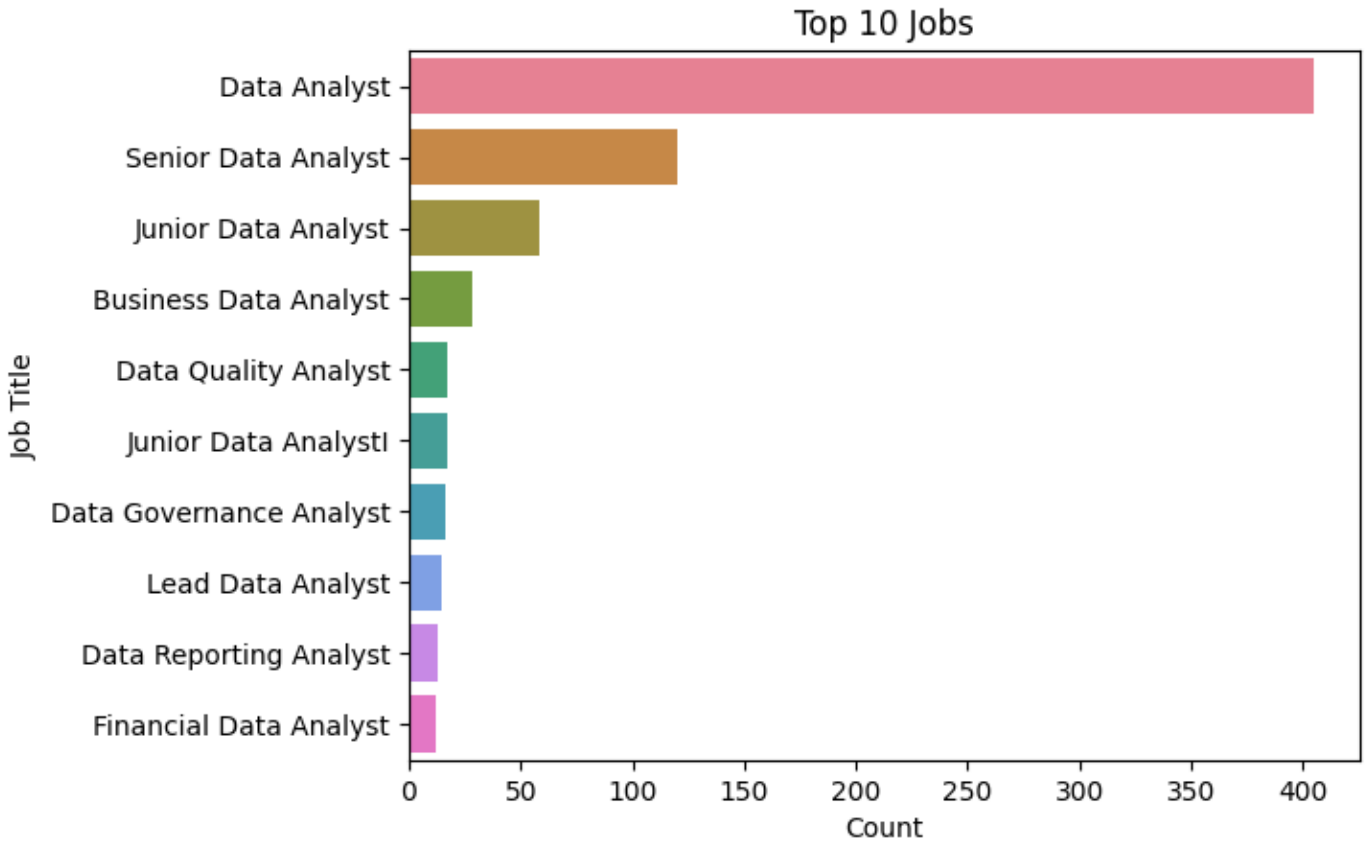
```
# --- 5. Company rating vs Average Salary scatter plot ---
plt.figure(figsize=(8, 5))
sns.scatterplot(data=data_analyst_jobs, x='rating', y='average_salary', alpha=0.6, hue='
plt.title('Company Rating vs Average Salary')
plt.xlabel('Rating')
plt.ylabel('Average Salary ($)')
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')
plt.show()
```



Top 10 Most Common Data Analyst Job Titles

In [20]:

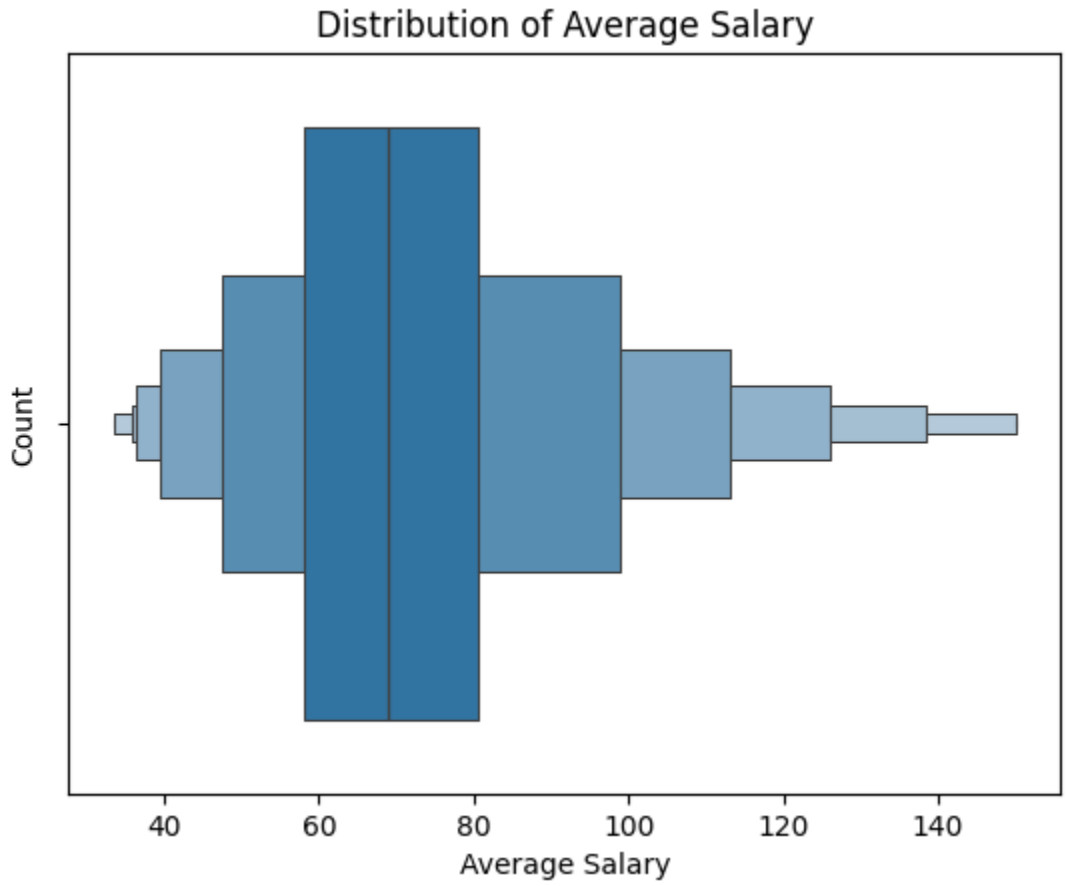
```
top_jobs = data_analyst_jobs['job_title'].value_counts().head(10)

# Create the bar plot with custom colors
sns.barplot(
    x=top_jobs.values,
    y=top_jobs.index,
    palette=sns.color_palette("husl", len(top_jobs))  # "husl" gives different bright co
)

plt.xlabel('Count')
plt.ylabel('Job Title')
plt.title('Top 10 Jobs')
plt.show()
```



Average Salary

In [21]:

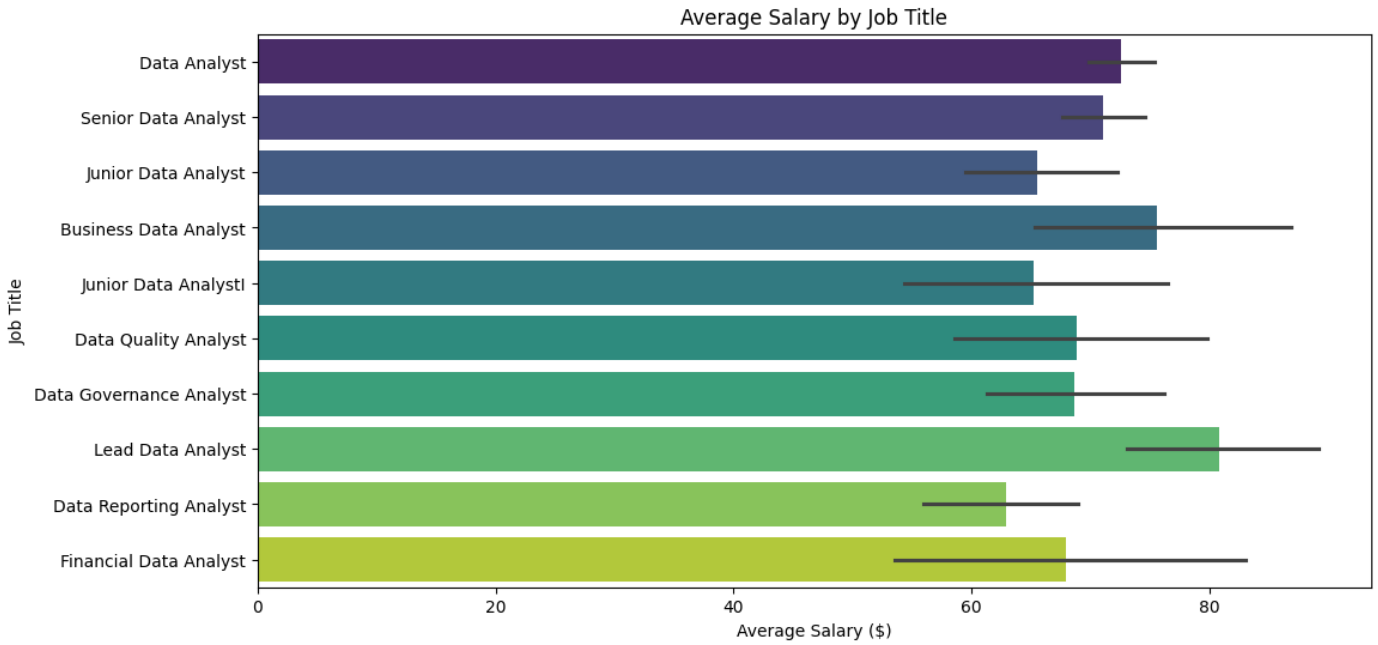
```
# Average Salary
sns.boxenplot(data=data_analyst_jobs, x='average_salary')
plt.xlabel('Average Salary')
plt.ylabel('Count')
plt.title('Distribution of Average Salary')
plt.show()
```



Top 10 Data Analyst Job Titles by Average Salary

In [22]:

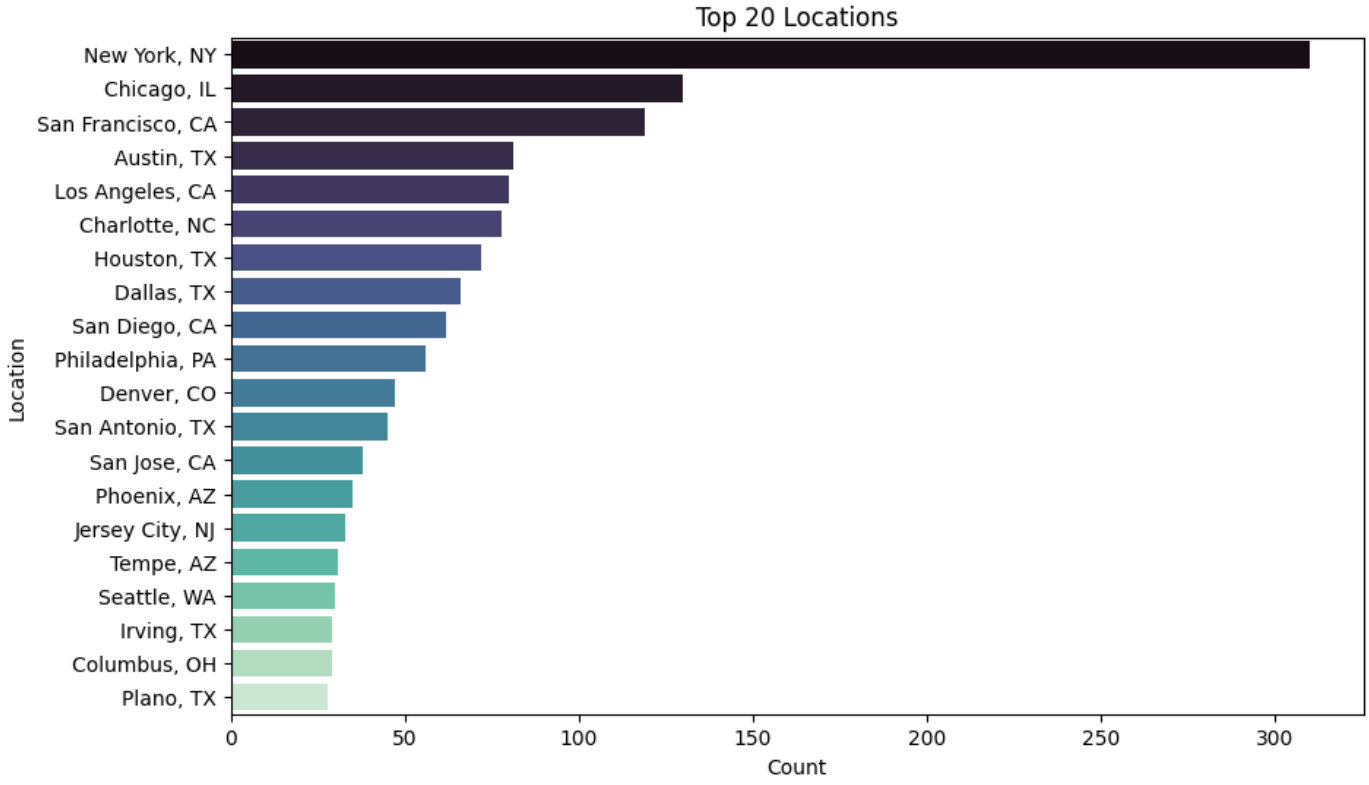
```
import seaborn as sns
import matplotlib.pyplot as plt

# Sort data by salary
data_analyst_jobs_sorted = data_analyst_jobs.sort_values(by='average_salary', ascending=

# Top 10 job titles by frequency
top_10_jobs = data_analyst_jobs_sorted['job_title'].value_counts().head(10).index

plt.figure(figsize=(12, 6))
sns.barplot(
    x='average_salary',
    y='job_title',
    data=data_analyst_jobs_sorted,
    orient='h',
    order=top_10_jobs,
    palette=sns.color_palette("viridis", len(top_10_jobs))  # nice gradient palette
)

plt.xlabel('Average Salary ($)')
plt.ylabel('Job Title')
plt.title('Average Salary by Job Title')
plt.show()
```



Top Locations Based on Average Salary

In [23]:

```
import seaborn as sns
import matplotlib.pyplot as plt

# Top work locations
top_locations = data_analyst_jobs['location'].value_counts().head(20)

plt.figure(figsize=(10, 6))
sns.barplot(
    x=top_locations.values,
    y=top_locations.index,
    palette=sns.color_palette("mako", len(top_locations))  # gradient colors
)

plt.xlabel('Count')
plt.ylabel('Location')
plt.title('Top 20 Locations')
plt.show()
```



Salary Trends by Location

In [24]:

```
job_location =data_analyst_jobs.groupby('location')["average_salary"].mean().reset_index
top_10 = job_location.sort_values(by = "average_salary",ascending=False).head(10)
```

In [25]:

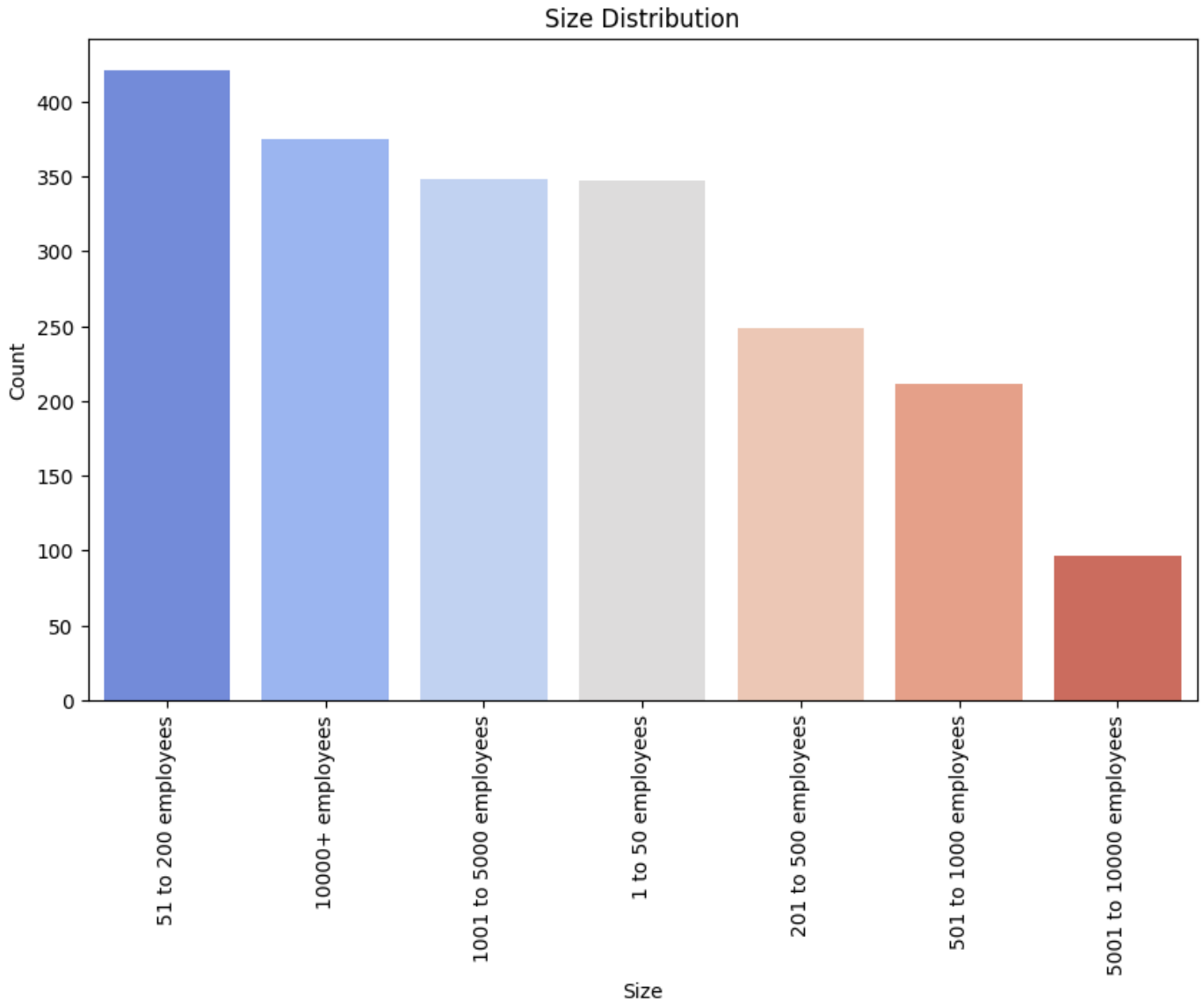
```
fig = px.bar(top_10, x='average_salary', y='location',orientation='h', title='Salary Tre
fig.update_layout(xaxis_title='AVG Salary (USD)',yaxis_title='Location', showlegend = Fa
fig.show()
```

Distribution of Company Sizes for Data Analyst Roles

In [26]:

```
import seaborn as sns
import matplotlib.pyplot as plt

filtered_size = data_analyst_jobs[
    (data_analyst_jobs['size'] != '-1') &
    (data_analyst_jobs['size'] != 'Unknown')
]

data_analyst_jobs_size = filtered_size['size'].value_counts().head(20)

plt.figure(figsize=(10, 6))
sns.barplot(
    x=data_analyst_jobs_size.index,
    y=data_analyst_jobs_size.values,
    palette=sns.color_palette("coolwarm", len(data_analyst_jobs_size))  # gradient
)

plt.xlabel('Size')
plt.ylabel('Count')
```

```
plt.title('Size Distribution')
plt.xticks(rotation=90)
plt.show()
```



Average Salary by Company Size

In [27]:

```
data_analyst_jobs_filtered = data_analyst_jobs[(data_analyst_jobs['size'] != '-1') & (da
data_analyst_jobs_sizeXsalary =data_analyst_jobs_filtered.groupby('size')['average_salar
```

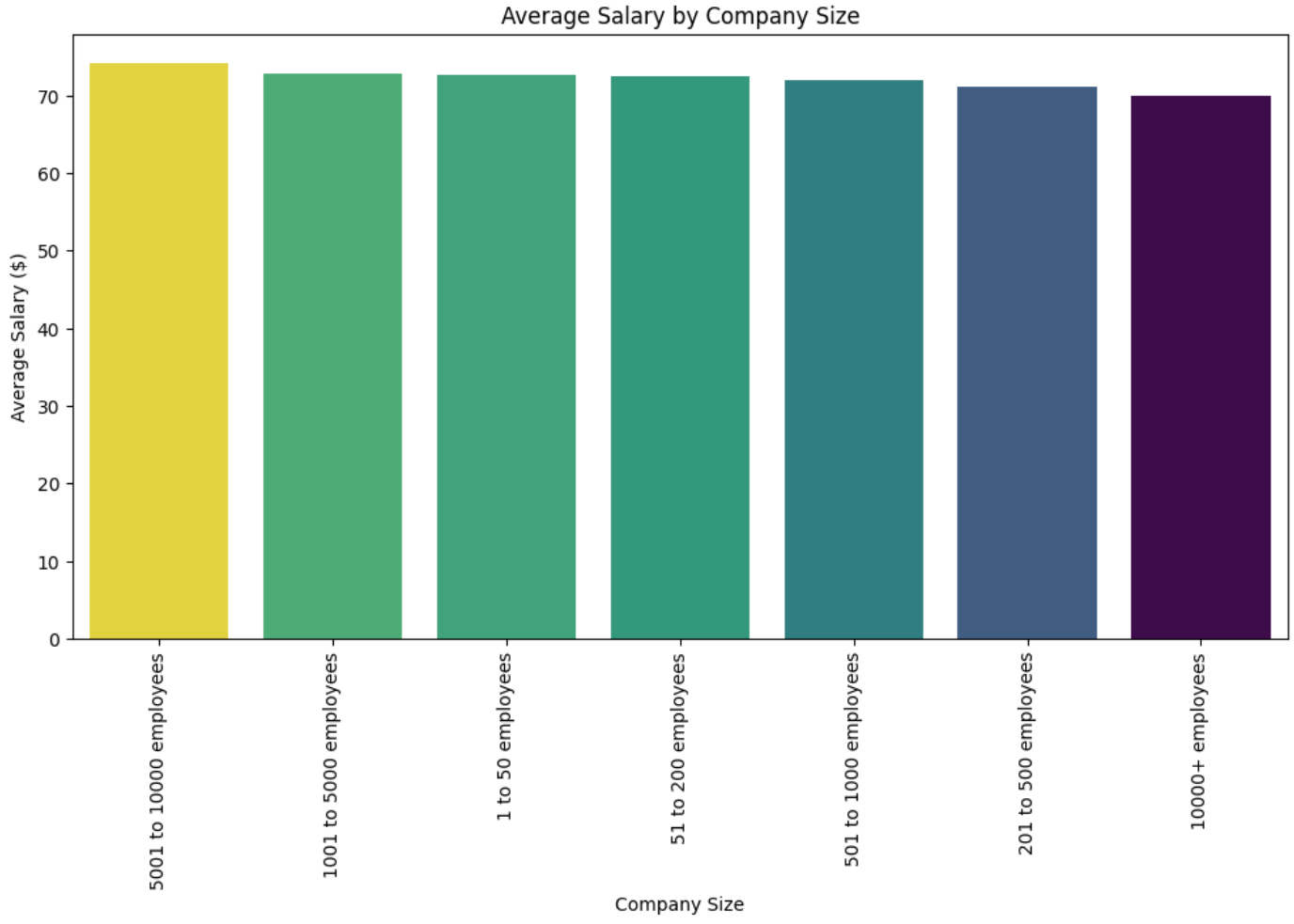
In [28]:

```
# Sort the DataFrame by 'AverageSalary' in descending order
data_analyst_jobs_sizeXsalary =data_analyst_jobs_sizeXsalary.sort_values(by='average_sal
```

In [29]:

```
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

# Normalize salary values to a 0-1 range for color mapping
norm = plt.Normalize(
    data_analyst_jobs_sizeXsalary['average_salary'].min(),
    data_analyst_jobs_sizeXsalary['average_salary'].max()
```

```
)

# Choose a colormap (e.g., viridis, plasma, coolwarm)
colors = plt.cm.viridis(norm(data_analyst_jobs_sizeXsalary['average_salary']))

plt.figure(figsize=(12, 6))
sns.barplot(
    x='size',
    y='average_salary',
    data=data_analyst_jobs_sizeXsalary,
    palette=colors
)

plt.xlabel('Company Size')
plt.ylabel('Average Salary ($)')
plt.title('Average Salary by Company Size')
plt.xticks(rotation=90)
plt.show()
```

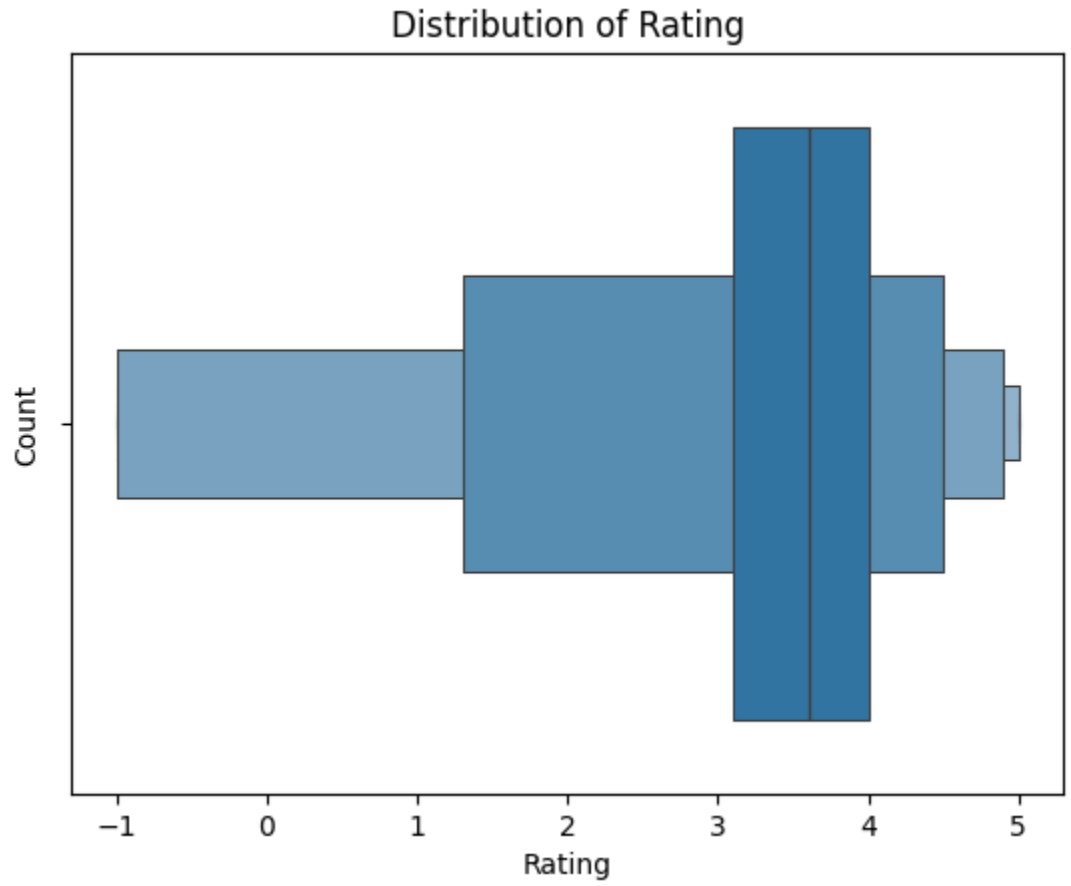


Distribution of Company Rating

In [37]:

```
import matplotlib.pyplot as plt
import seaborn as sns

sns.boxenplot(data=data_analyst_jobs, x='rating')
plt.xlabel('Rating')
plt.ylabel('Count')
```

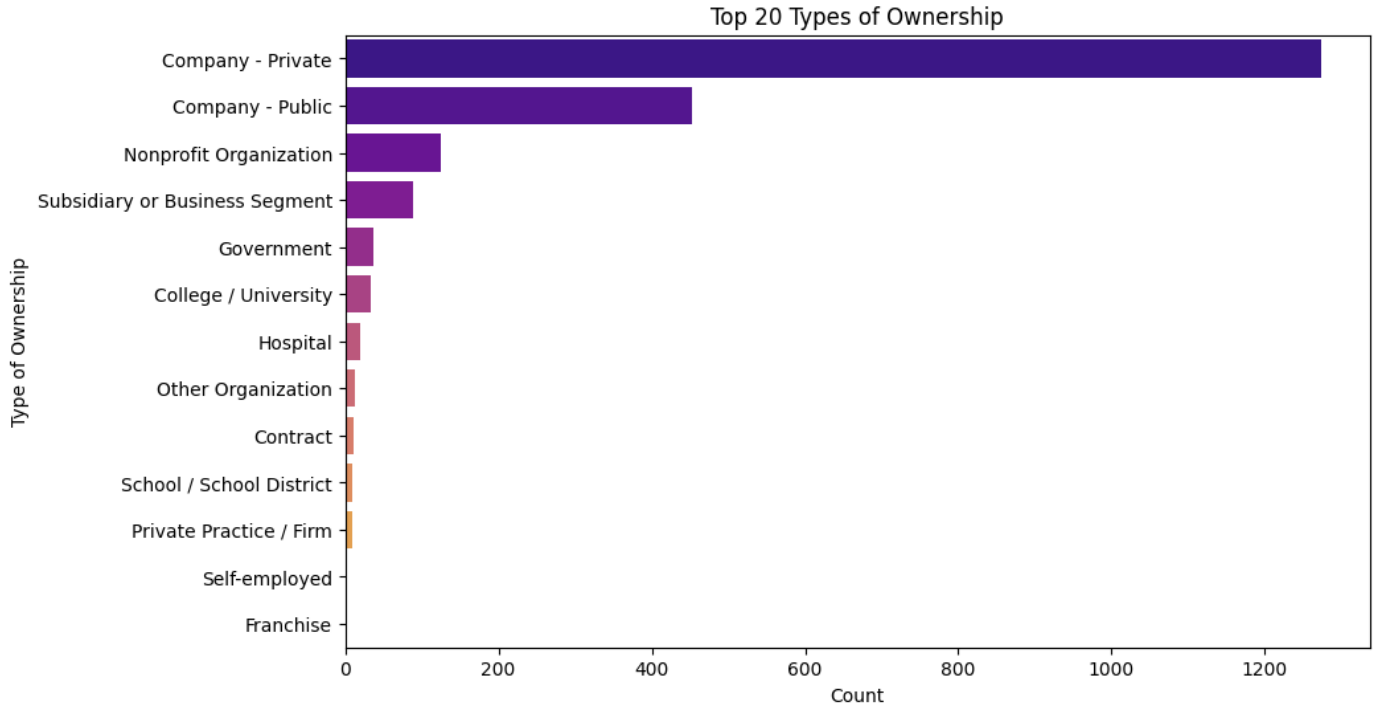
```
plt.title('Distribution of Rating')
plt.show()
```



Type of Ownership

In [32]:

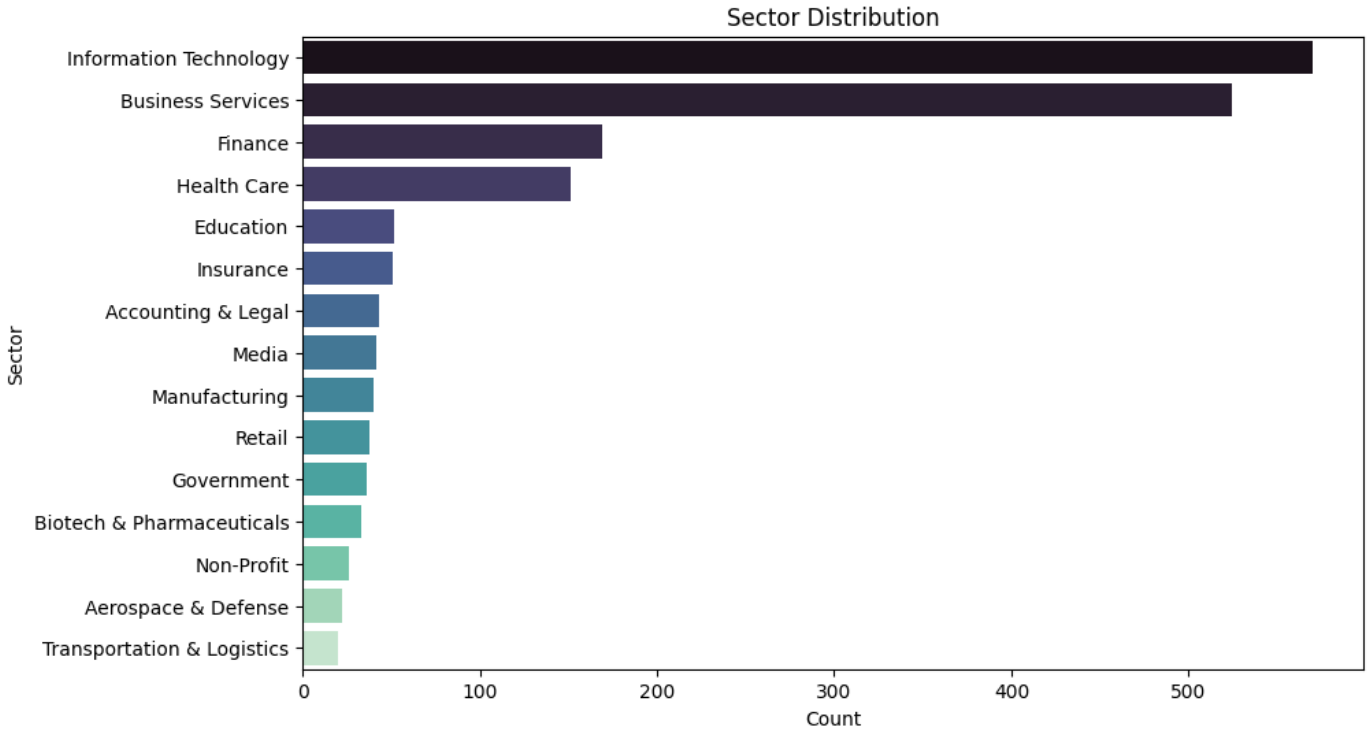
```
import seaborn as sns
import matplotlib.pyplot as plt

TOP = data_analyst_jobs[
    (data_analyst_jobs['type_of_ownership'] != '-1') &
    (data_analyst_jobs['type_of_ownership'] != 'Unknown')
]

TOP = TOP['type_of_ownership'].value_counts().head(20)

plt.figure(figsize=(10, 6))
sns.barplot(
    x=TOP.values,
    y=TOP.index,
    palette=sns.color_palette("plasma", len(TOP))  # colorful gradient
)

plt.xlabel('Count')
plt.ylabel('Type of Ownership')
plt.title('Top 20 Types of Ownership')
plt.show()
```



Top 15 Sectors Employing Data Analysts

In [33]:

```
data_analyst_jobs_sector = data_analyst_jobs[
    data_analyst_jobs['sector'] != '-1'
]['sector'].value_counts().head(15)

plt.figure(figsize=(10, 6))
sns.barplot(
    x=data_analyst_jobs_sector.values,
    y=data_analyst_jobs_sector.index,
    palette=sns.color_palette("mako", len(data_analyst_jobs_sector))  # gradient
)

plt.xlabel('Count')
plt.ylabel('Sector')
plt.title('Sector Distribution')
plt.show()
```



Average Salary by Sector

In [34]:

```
average_salary_by_sector =data_analyst_jobs[data_analyst_jobs['sector'] !='-1'].groupby(
average_salary_by_sector =average_salary_by_sector.sort_values(by='average_salary',ascen
```

In [35]:

```
# Normalize salary values for color mapping
norm = plt.Normalize(
    average_salary_by_sector['average_salary'].min(),
    average_salary_by_sector['average_salary'].max()
)
colors = plt.cm.viridis(norm(average_salary_by_sector['average_salary']))

plt.figure(figsize=(12, 6))
sns.barplot(
    x='sector',
    y='average_salary',
    data=average_salary_by_sector,
    palette=colors
)

plt.xticks(rotation=90)
plt.xlabel('Sector')
plt.ylabel('Average Salary (Thousands Dollars)')
plt.title('Average Salary by Sector')
plt.show()
```