Project Title	**Data Analyst Jobs**
Tools	ML, Python, SQL, Excel
Domain	Finance Analyst
Project Difficulties level	intermediate

Dataset : Dataset is available in the given link. You can download it at your convenience.

[Click here to download data set]

Amidst the pandemic many people lost their jobs, with this dataset it is possible to hone the job search so that more people in need can find employment.
This dataset was created by picklesueat and contains more than **2000 job listing for data analyst** positions, with features such as:

- Salary Estimate
- Location
- Company Rating
- Job Description
- and more.

How to use

- Find the best jobs by salary and company rating
- Explore skills required in job descriptions
- Predict salary based on industry, location, company revenue
- Your kernel can be featured here!
- Data Engineer Jobs
- Business Analyst Jobs
- Data Scientist Jobs
- More Datasets

Acknowledgements

If you use this dataset, please support the author.

License

Splash banner

Photo by Chris Liverani on Unsplash

Splash Icon

Icon by Eucalyp available on flaticon.com

Example: You can get the basic idea how you can create a project from here

Step 1: Problem Definition

Objective

- Analyze trends in data analyst job postings.
- Predict salary ranges for given job attributes.
- Provide insights into company ratings, locations, and industry trends.

Input Columns

- **Job Title**: Position name.
- **Salary Estimate**: Predicted/actual salary.
- **Job Description**: Text describing responsibilities.
- **Rating**: Employer rating.
- **Company Name**: Employer name.
- **Location**: Job location.
- **Headquarters**: Company HQ location.
- **Size, Founded, Type of ownership**: Company metadata.
- **Industry, Sector, Revenue, Competitors**: Market details.
- **Easy Apply**: Indicates if the job has a one-click application option.

Step 2: Data Collection

Assume data is in a CSV file named `data_analyst_jobs.csv`. Load the data and inspect.

Code:

python

```python
import pandas as pd

# Load the dataset
data = pd.read_csv("data_analyst_jobs.csv")

# Inspect the dataset
print(data.head())
print(data.info())
```

Step 3: Exploratory Data Analysis (EDA)

Step 3.1: Overview

- Check for duplicates.
- Understand column distributions.

Code:

python

```python
# Check for duplicates
print(f"Duplicate rows: {data.duplicated().sum()}")

# General statistics
print(data.describe(include='all'))
```

```
# Value counts for categorical columns
for col in ['Job Title', 'Type of ownership', 'Industry',
'Sector']:
    print(data[col].value_counts().head())
```

Step 3.2: Visualization

Use visualizations to explore data.

Code: Salary Distribution

python

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

# Salary distribution
plt.figure(figsize=(10, 6))
sns.histplot(data['Salary Estimate'], kde=True, bins=20)
plt.title("Salary Estimate Distribution")
plt.xlabel("Salary")
plt.show()
```

Code: Ratings by Industry

python

```python
plt.figure(figsize=(12, 6))
sns.boxplot(x='Industry', y='Rating', data=data)
plt.xticks(rotation=90)
plt.title("Company Ratings by Industry")
plt.show()
```

Step 4: Data Cleaning

Step 4.1: Handling Missing Values

- Fill missing values with appropriate techniques.
- Drop columns with excessive missing data.

Code:

python

```python
# Check missing values
print(data.isnull().sum())

# Fill missing numerical values
data['Rating'].fillna(data['Rating'].median(), inplace=True)

# Drop columns with > 30% missing data
threshold = len(data) * 0.3
data = data.dropna(thresh=threshold, axis=1)
```

```
# Forward-fill categorical values
categorical_cols = ['Company Name', 'Industry', 'Sector', 'Type of ownership']
data[categorical_cols] = data[categorical_cols].fillna(method='ffill')
```

Step 4.2: Standardizing Data

- Extract numerical values from text (e.g., `Salary Estimate`).

Code:

python

```
# Extract minimum salary
data['Min Salary'] = data['Salary Estimate'].str.extract(r'(\d+)').astype(float)

# Extract maximum salary
data['Max Salary'] = data['Salary Estimate'].str.extract(r'-\s*(\d+)').astype(float)

# Compute average salary
data['Avg Salary'] = (data['Min Salary'] + data['Max Salary']) / 2
```

```
# Drop old salary column
data.drop('Salary Estimate', axis=1, inplace=True)
```

Step 5: Feature Engineering

Step 5.1: Text Analysis

- Process `Job Description` for keywords (e.g., Python, Excel).

Code:

python

```
# Extract keywords from Job Description
data['Python'] = data['Job Description'].str.contains('Python',
case=False, na=False).astype(int)
data['Excel'] = data['Job Description'].str.contains('Excel',
case=False, na=False).astype(int)

# Create a tech skills score
data['Tech_Skills'] = data['Python'] + data['Excel']
```

Step 5.2: Location Splits

python

```
# Extract city and state from location
data['City'] = data['Location'].str.split(',', expand=True)[0]
```

```python
data['State'] = data['Location'].str.split(',', expand=True)[1]
```

Step 6: Statistics

Analyze relationships using correlation and significance tests.

Code:

python

```python
# Correlation matrix
plt.figure(figsize=(10, 8))
sns.heatmap(data.corr(), annot=True, cmap="coolwarm")
plt.title("Correlation Matrix")
plt.show()
```

Step 7: Model Development

Step 7.1: Data Splitting

Split into features and target:

python

```python
from sklearn.model_selection import train_test_split

# Define features and target
```

```
features = ['Rating', 'Tech_Skills', 'Size', 'Founded']
X = data[features]
y = data['Avg Salary']

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)
```

Step 7.2: Model Training

Use Random Forest Regressor to predict salaries.

python

```
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_absolute_error, r2_score

# Train model
model = RandomForestRegressor(n_estimators=100,
random_state=42)
model.fit(X_train, y_train)

# Predict
y_pred = model.predict(X_test)
```

```
# Evaluate
mae = mean_absolute_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f"MAE: {mae}, R2 Score: {r2}")
```

Step 8: Deployment

Deploy model using Streamlit or Flask.

Example:

python

```
import streamlit as st

st.title("Data Analyst Job Analysis")
st.write("Average Salary Prediction")

# User input
rating = st.slider("Company Rating", 1, 5, 3)
tech_skills = st.slider("Tech Skills Score", 0, 2, 1)
size = st.selectbox("Company Size", [0, 1, 2])
founded = st.number_input("Year Founded", min_value=1900,
max_value=2023, value=2000)

# Predict
```

```
prediction = model.predict([[rating, tech_skills, size,
founded]])
st.write(f"Predicted Salary: ${prediction[0]:,.2f}")
```

Sample Code and output

```
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('/kaggle/input/data-analyst-jobs/DataAnalyst.csv')
```

3 Dataset Overview

The overview is prepared to get the feel on data structure. It will also include a quick analysis on missing values, basic statistics and data manipulation.

The dataset consists of the following information

- **Job Title** :A name that describes someone's job or position.
- **Salary Estimate**: A display a range for annual base or hourly pay and are specific to Data Analytics Industry
- **Job Description**: The plain-language tool that explains the tasks, duties, function and responsibilities of a position
- **Rating** : Company Rating
- **Company Name**: The name of the company
- **Location**: The location where the job is available
- **Headquarters**: The headquarters of the company
- **Size**: The size of the employee
- **Type of Ownership**: Type of ownership whether it is public, private or non-profit
- **Industry**: Different industries where the job is available
- **Sector**: Sector where the job is available
- **Revenue**: Company earnings annually.
- **Easy Apply**: Easy Apply section
- **Observations**
- There are 2253 rows and 13 columns and 1 missing values.

(to see the details, please expand)

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 of columns:{data_analyst_jobs.shape[1]}; No of missing values:{sum(data_analyst_jobs.isna().sum())}')
```

```
Number of rows:2253;Number of columns:13; No of missing values:1
```

3.1 Quick view

Below is the first 5 rows of `data analyst jobs` dataset:

In [4]:

```
data_analyst_jobs.head()
```

Out[4]:

	Job Title	Salary Estimat	Job Description	Rating	Company Name	Location	Headquarters	Size	Type of ownersh	Industry	Sector	Revenue	EasyA

		e							ip				pply
0	Data Analyst, Center on Immigration and Justic...	37K – 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	201 to 500 employees	Nonprofit Organization	Social Assistance	Non-Profit	100 to 100to500 million (USD)	True
1	Quality Data a	37 K	Overview\n\nProvides analytical	3.8	Visiting Nurse Servic	New York,	New York, NY	10000+ emplo	Nonprofit Org	Health Care	Health Care	2 t	-1

	Analyst	– 37K− 66K (Glassdoor est.)	and technical ...		e of New York\n3.8	NY		yees	anization	Services & Hospitals		o 2to 5 billion (USD)	
2	Senior Data Analyst, Insights & Analytics Tea	37K – 37K− 66K (Gl	We're looking for a Senior Data Analyst who ha...	3.4	Squarespace\n3.4	New York, NY	New York, NY	1001 to 5000 employees	Company - Private	Internet	Information Technology	Unknown / Non-Applicable	-1

	m...	assdoor est.)											
3	Data Analyst	37K – 37K−66K (Glassdoor est.)	Requisition NumberRR-0001939\nRemote:Yes\nWec...	4.1	Celerity\n4.1	New York, NY	McLean, VA	201 to 500 employees	Subsidiary or Business Segment	IT Services	Information Technology	50 to 50to100 million (USD)	-1
4	Report	37	ABOUT FANDUEL	3.	FanDu	New	New York	501 to	Compa	Sports	Arts, Ente	10	Tr

	ing Data Analyst	K – 37K–66K (Glassdoor est.)	GROUP\n\nFanDuel Group is a worl...	9	el\n3.9	York, NY	, NY	1000 employees	ny - Private	& Recreation	rtainment & Recreation	0 to 100to 500 million (USD)	ue

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
```

3.2 Data Manipulation

Some of the data in your dataset needed to be moved around in order to make it easier for you to analyze it. For example, you might want to rename some columns in your dataset. You also want to avoid duplicates or other redundancies on your dataset.

3.2.1 Renaming Columns for Better Analysis

The columns are renamed for easier analysis.

In [6]:

```
#renaming columns for better analysis


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

```python
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 [7]:

```python
data_analyst_jobs.head()
```

Out[7]:

	job_title	salary_estimate	job_description	Rating	company_name	location	headquarters	size	type_of_ownership	industry	sector	revenue	easy_apply
0	Data Analyst,	37 K	Are you eager to roll up your sleeves	3.2	Vera Institute of Justic	New Yor	New York,	201 to 500	Nonprofit Organization	Social Assistan	Non-Profit	100 t	True

	Center on Immigration and Justic...	– 37K – 66K (Glassdoor est.)	and harn...		e\n3.2	k, NY	NY	employees		ce		o 100to 500 million (USD)	
1	Quality Data Analyst	37K – 37K – 66K (Glassdo	Overview\n\nProvides analytical and technical ...	3.8	Visiting Nurse Service of New York\n3.8	New York, NY	New York, NY	10000+ employees	Nonprofit Organization	Health Care Services & Hospital	Health Care	2 to 2to 5 billion (U	-1

		or est.)								s		SD)	
2	Senior Data Analyst, Insights & Analytics Team...	37K – 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	1001 to 5000 employees	Company - Private	Internet	Information Technology	Unknown / Non-Applicable	-1
3	Data Analyst	37K	Requisition NumberRR-0001939\nRemot	4.1	Celerity\n4.1	New York	McLean, VA	201 to 500	Subsidiary or Business	IT Services	Information Technol	50t	-1

		– 37K – 66K (Glassdoor est.)	e:Yes\nWe c...			k, NY		employees	Segment		ogy	o 50to 100 million (USD)	
4	Reporting Data Analyst	37K – 37K – 66K (Glassdo	ABOUT FANDUEL GROUP\n\nFanDuel Group is a worl...	3.9	FanDuel\n3.9	New York, NY	New York, NY	501 to 1000 employees	Company - Private	Sports & Recreation	Arts, Entertainment & Recreation	100 to 1000to 5000	True

		or est.)										mil lio n (U SD)	

4 Job Title

The job title column on the dataset showed duplicated job titles that would inhibit a proper analysis. The names were replaced to avoid duplicates.

Observations:

The top 5 jobs in the data set are as follows.

- ***Data Analyst*** that has a value of 405
- ***Senior Data Analyst*** that has a value of 131
- ***Junior Data Analyst*** that has a value of 58
- ***Business Data Analyst*** that has a value of 28
- ***Data Quality Analyst*** that has a value of 17

(to see the details, please expand)

In [8]:

```
# replacing Job Titles to avoid duplicates


data_analyst_jobs['job_title'] =
```

```
data_analyst_jobs['job_title'].replace(['Sr. Data Analyst',
'sr. data analyst', 'Sr Data Analyst', 'sr data
analyst','senior data analyst', 'Senior Data Analyst', 'Data
Analyst III', 'data analyst iii', 'senior data analyst'],
                                            'Senior Data
Analyst', regex=True)
data_analyst_jobs['job_title'] =
data_analyst_jobs['job_title'].replace(['Data Analyst I', 'data
analyst i', 'Data Analyst Junior', 'data analyst junior',
                                             'Junior Data
Analyst', 'Junior Data AnalystI', 'Junior Data Analystl'],
'Junior Data Analyst', regex=True)
data_analyst_jobs['job_title'] =
data_analyst_jobs['job_title'].replace(['Data Analyst II',
'data analyst ii', 'Middle Data Analyst'],
                                            'Middle Data
Analyst', regex=True)
```

In [9]:

```
# 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[9]:

```
job_title
Data Analyst              405
Senior Data Analyst       131
Junior Data Analyst        58
Business Data Analyst      28
Data Quality Analyst       17

Name: count, dtype: int64
```

5 Salary Estimate and Trends

The salary estimation column is an item and needs to be converted into a float column for a better analysis.To change the column, extract the minimum and maximum salary, convert them to a float column and drop the columns that are not relevant.

In [10]:

```
## Changing Salary column to int for better calculation

data_analyst_jobs[['MinSalary', 'MaxSalary']] =
data_analyst_jobs['salary_estimate'].str.extract(r'\$(\d+)K-\$(
\d+)K')
```

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

In [11]:

```
# changing format to float

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=True)
```

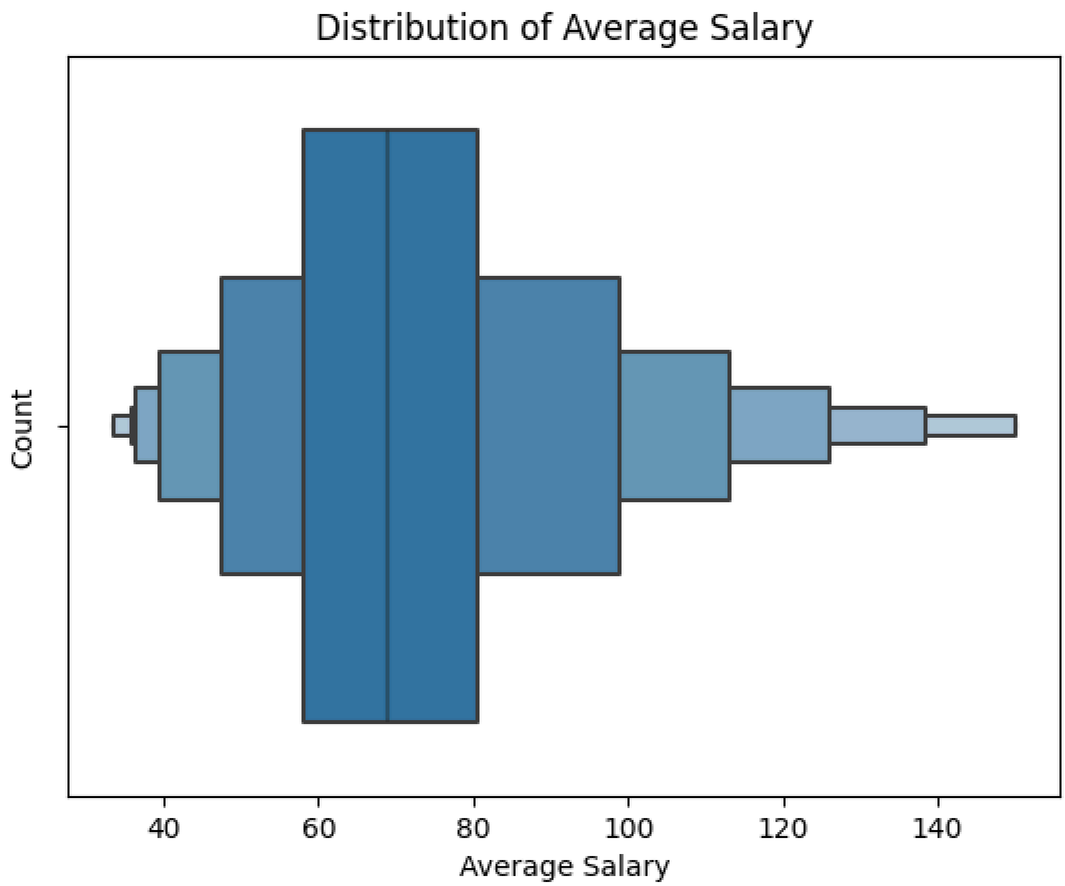
5.1 Average Salary

Observations:

The average salary for data analysts jobs is between 60K-80K annually wth a minimim of 40K and a maximum of 140K.

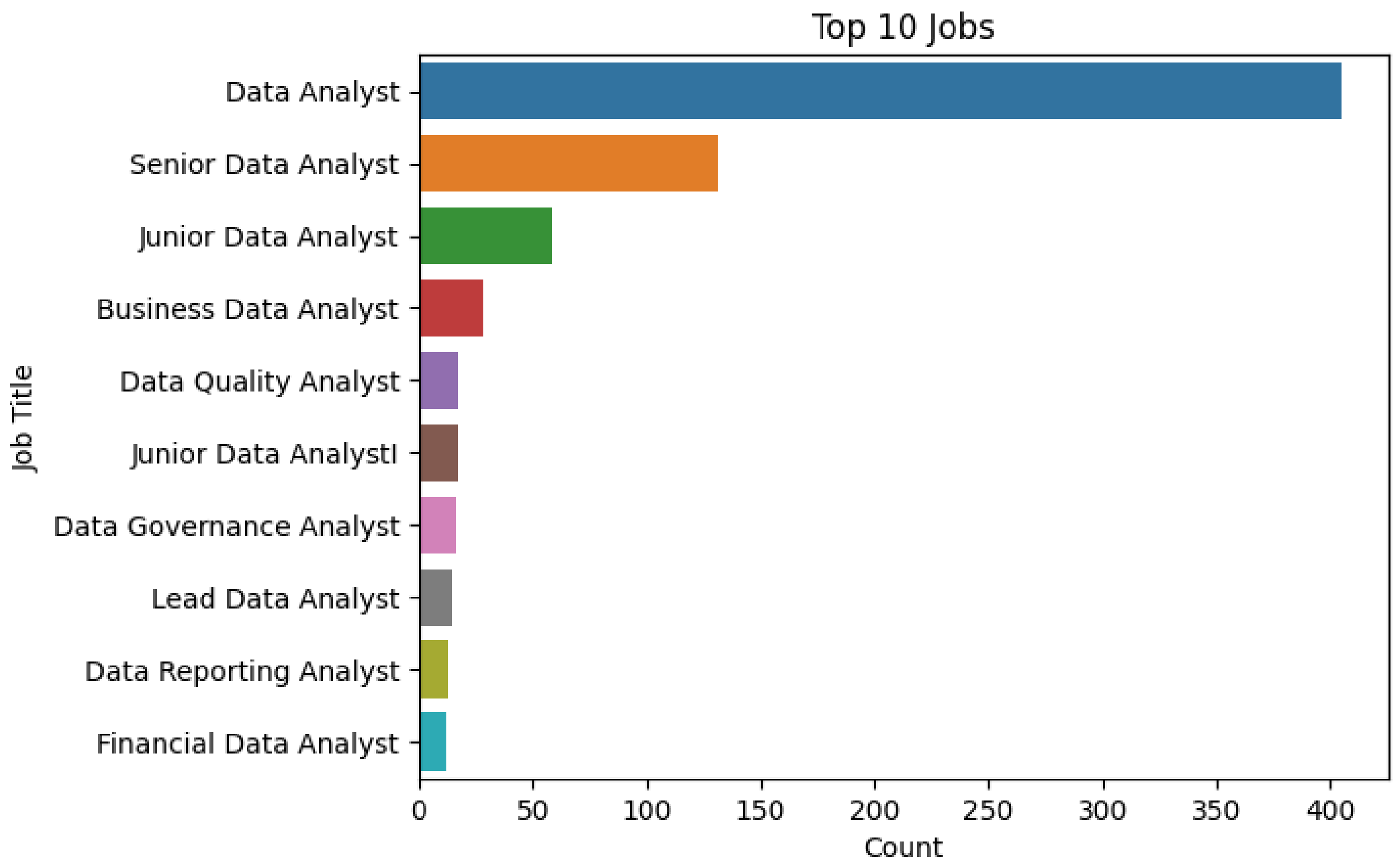
In [12]:

```
# 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()
```



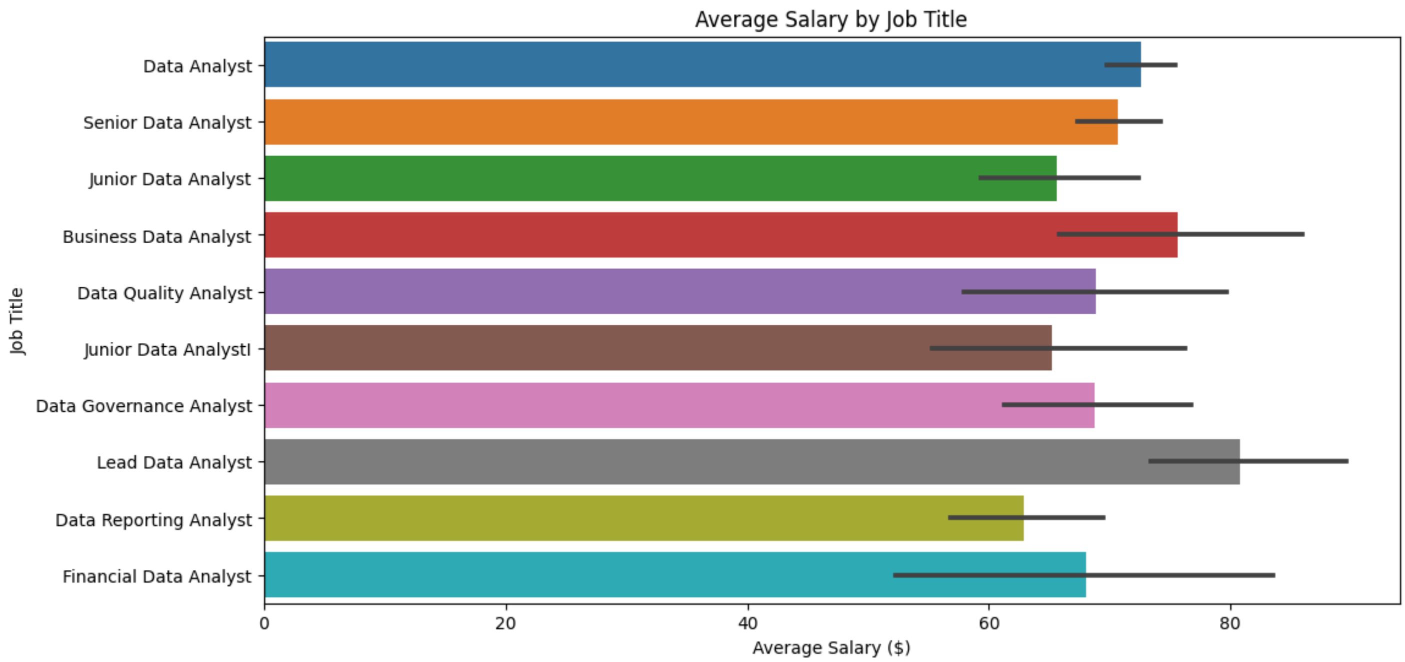
In [13]:

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

sns.barplot(x=top_jobs.values, y=top_jobs.index)

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




```
In [14]:
# Salary and Job Title

data_analyst_jobs_sorted =
data_analyst_jobs.sort_values(by='average_salary',
ascending=False)

plt.figure(figsize=(12, 6))
sns.barplot(x='average_salary', y='job_title',
data=data_analyst_jobs_sorted, orient='h',
order=data_analyst_jobs_sorted['job_title'].value_counts().head
(10).index)
```

```
plt.xlabel('Average Salary ($)')

plt.ylabel('Job Title')

plt.title('Average Salary by Job Title')

plt.show()
```



5.1.1 Average Salary by Job Title

1. Data Analyst
2. Senior Data Analyst
3. Junior Data analyst
4. Business Data Analyst
5. Data Quality Analyst
6. Junior Data Analyst

7. Data Governance Analyst
8. Lead Data Analyst
9. Data Reporting Analyst
10. Financial Analyst

Observations

The dataset shows that there is a massive demand for Data Analysts in the industry. There is a huge gap in job availability between the positions of Data Analyst and Senior Data Analyst, which are the two most sought-after positions in the industry.When it comes to salary, Data Analysts are paid on an average between 60,000-80,000 per year. The dataset also shows that the highest-paying job in the industry is Lead Data Analyst, which pays above 80,000 per year but lacks job availability.

(to see the details, please expand)

5.2 Salary Trends by Location

In [15]:

```
#salary trends by location
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 [16]:

```
fig = px.bar(top_10, x='average_salary', y='location',
orientation='h', title='Salary Trends by Location', color =
"location")
fig.update_layout(xaxis_title='AVG Salary (USD)',
yaxis_title='Location', showlegend = False)

fig.show()
```

020406080100120140Glenview, ILElk Grove Village, ILNorthfield, ILBerkeley, CAWhittier, CAPico Rivera, CALos Gatos, CAMarin City, CADaly City, CANewark, CA

Salary Trends by LocationAVG Salary (USD)Location

5.2.1 Top Locations Based on Average Salary

Top 5 Locations and Headquarters

- New York, NY
- Chicago, IL
- San Franciso,CA
- Austin,TX
- Los Angeles CA

Top 5 Locations by Salary

- Newark, CA
- Daly City, CA
- Marin City, CA
- Los, Gatos, CA,
- Pico Rivera, CA

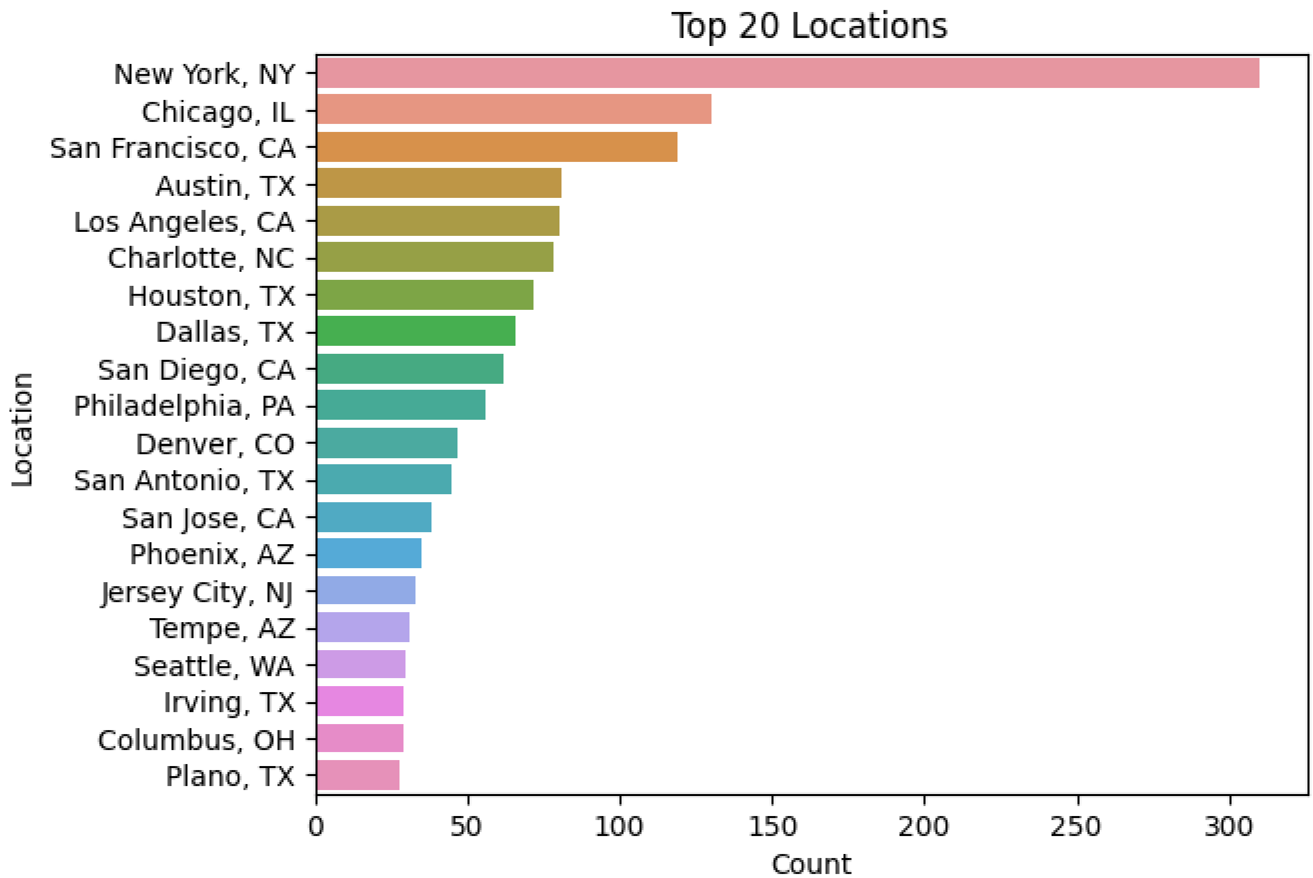
Observations

The dataset showed that the top locations and headquarters are the same. The job openings in New York is significantly higher compare to the job in Chicago.

Looking at the salary correlation the top 5 locations that has a higher salary are all located in California.

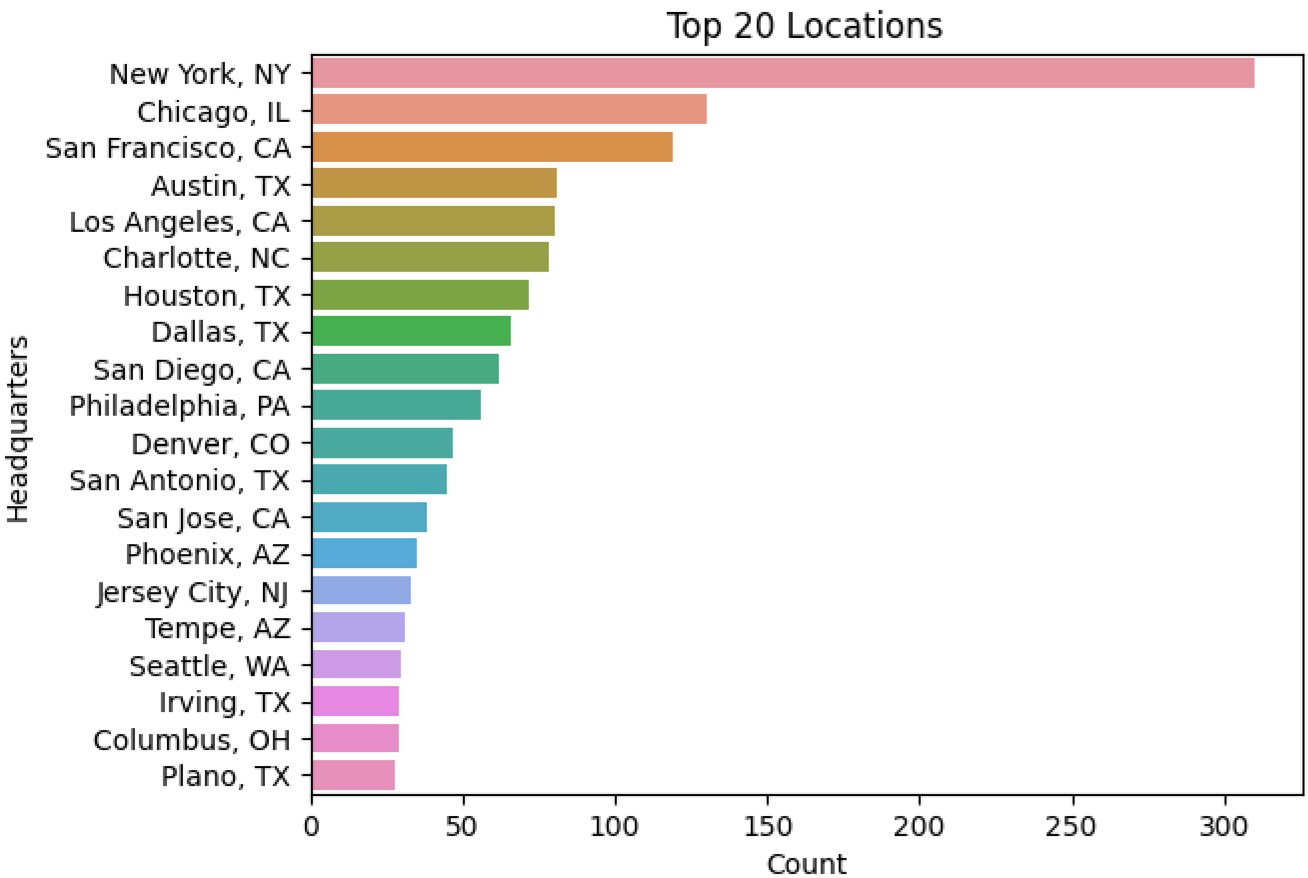
In [17]:

```
# Top work locations among interviewed

top_locations =
data_analyst_jobs['location'].value_counts().head(20)

sns.barplot(x=top_locations.values, y=top_locations.index)

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



In [18]:

```
top_headquarters =
data_analyst_jobs['headquarters'].value_counts().head(20)


sns.barplot(x=top_locations.values, y=top_locations.index)


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



6 Company

These are the focus areas of the analysis.

- 6.1 Average Salary by Company Size
- 6.2 Company Rating
- 6.3 Type of Ownership

6.1 Average Salary by Company Size

The company that has a biggest size which is around 5001-10000 employees has the smallest count. The company that has the highest count has around 51-200 employees. The smallest company size has a count of 350 and it's the same as the company that has around 1000 - 5000 employees. There is no significant difference between the company size and average salary in the dataset.

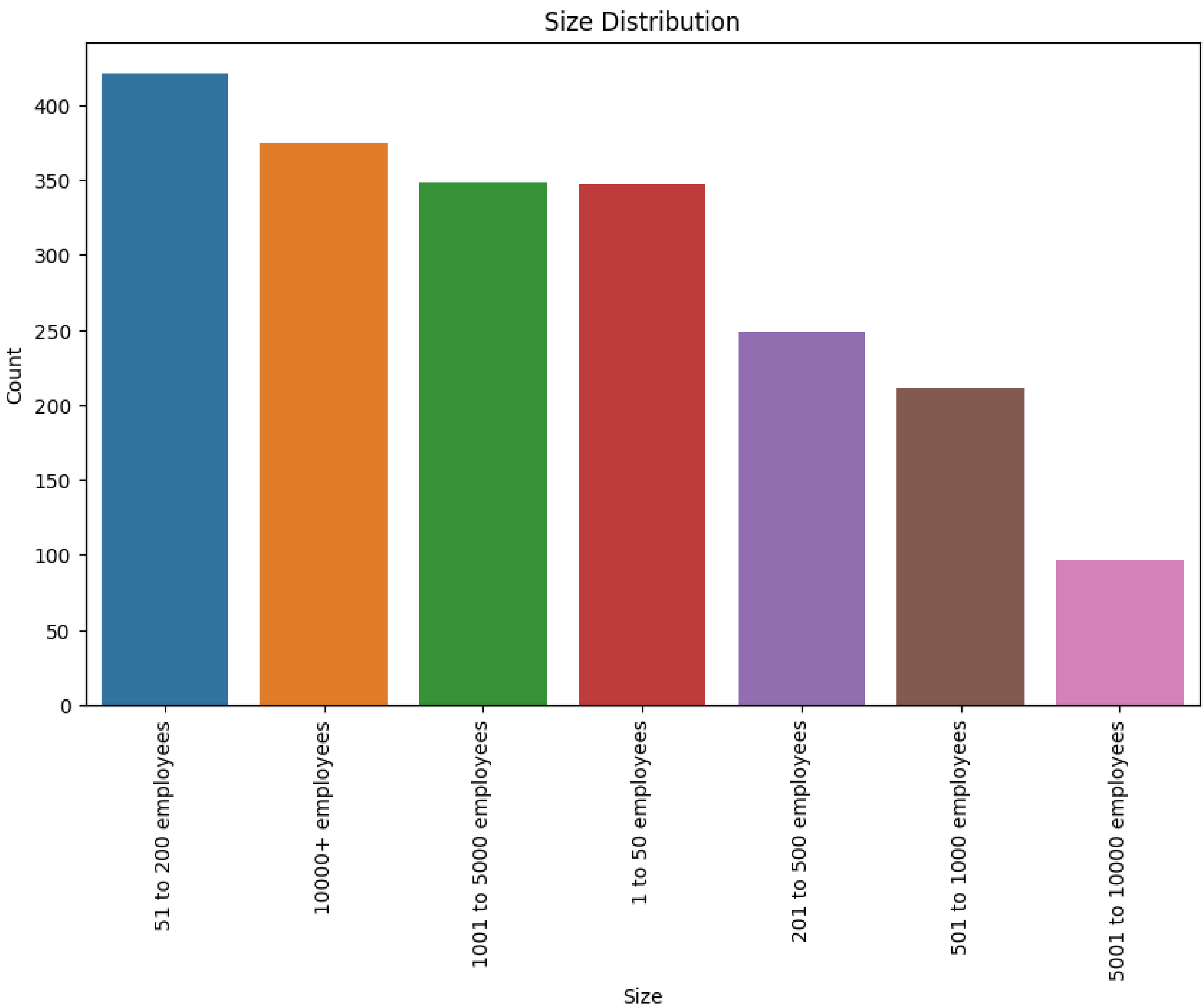
Observations

Based on the data, there are not a lot of companies that has 5000-10000 employees in the Data Analytics Industry. On the other hand the company that has more that 10000 employees has more than 350 which falls on 2nd place. The company size that has the most value counted is the company that has 51-200

In [19]:

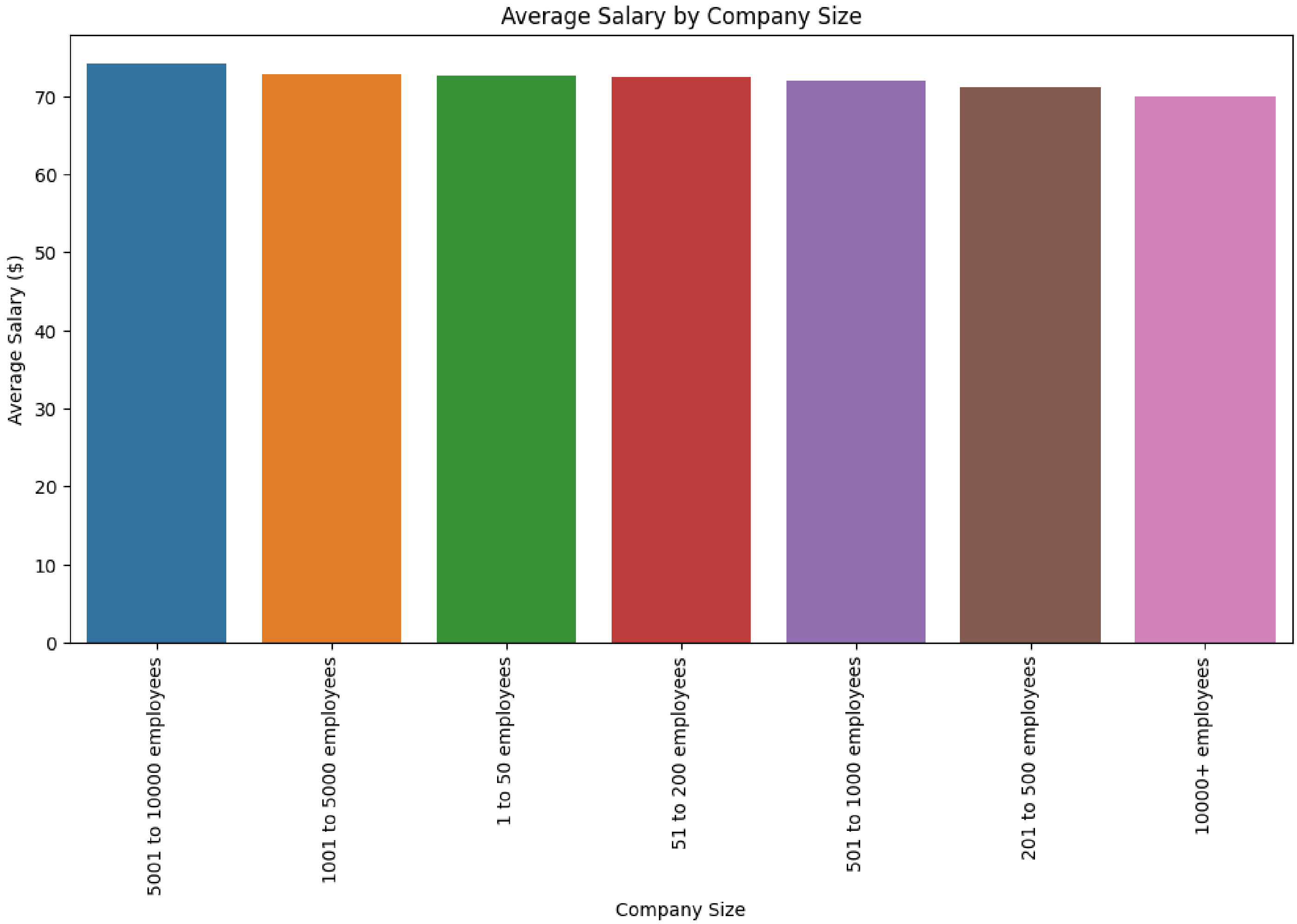
```
# Companies by Amount of Employees


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

```
plt.title('Size Distribution')

plt.xticks(rotation=90)

plt.show()
```




```
In [20]:

# Salary by Company Size
```

```
data_analyst_jobs_filtered =
data_analyst_jobs[(data_analyst_jobs['size'] != '-1') &
(data_analyst_jobs['size'] != 'Unknown')]
data_analyst_jobs_sizeXsalary =
data_analyst_jobs_filtered.groupby('size')['average_salary'].me
an().reset_index()

# Sort the DataFrame by 'AverageSalary' in descending order
data_analyst_jobs_sizeXsalary =
data_analyst_jobs_sizeXsalary.sort_values(by='average_salary',
ascending=False)

# Plot the bar chart
plt.figure(figsize=(12, 6))
sns.barplot(x='size', y='average_salary',
data=data_analyst_jobs_sizeXsalary)
plt.xlabel('Company Size')
plt.ylabel('Average Salary ($)')
plt.title('Average Salary by Company Size')
plt.xticks(rotation=90)

plt.show()
```



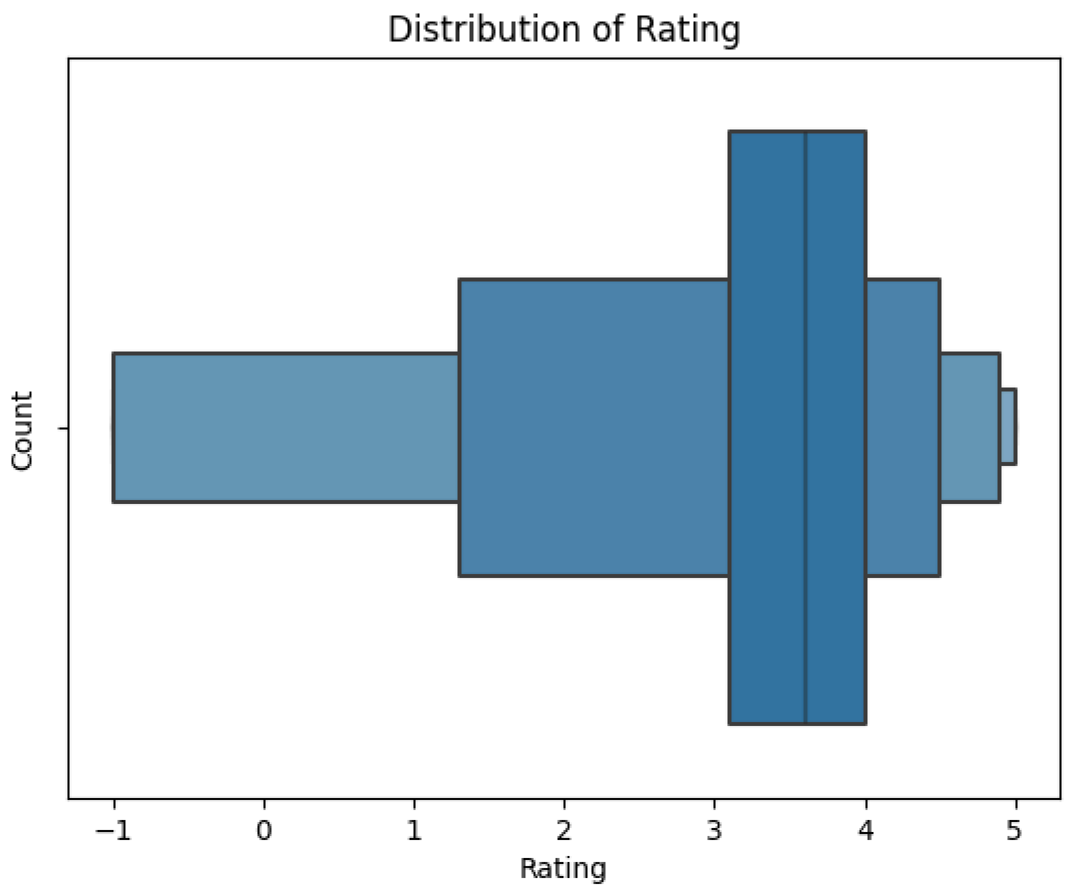
6.2. Company Rating

The rating shows that the rating is between 3.0- 4.0 meaning that there is a data analyst jobs rating is fairly average.

In [21]:

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

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

plt.show()
```



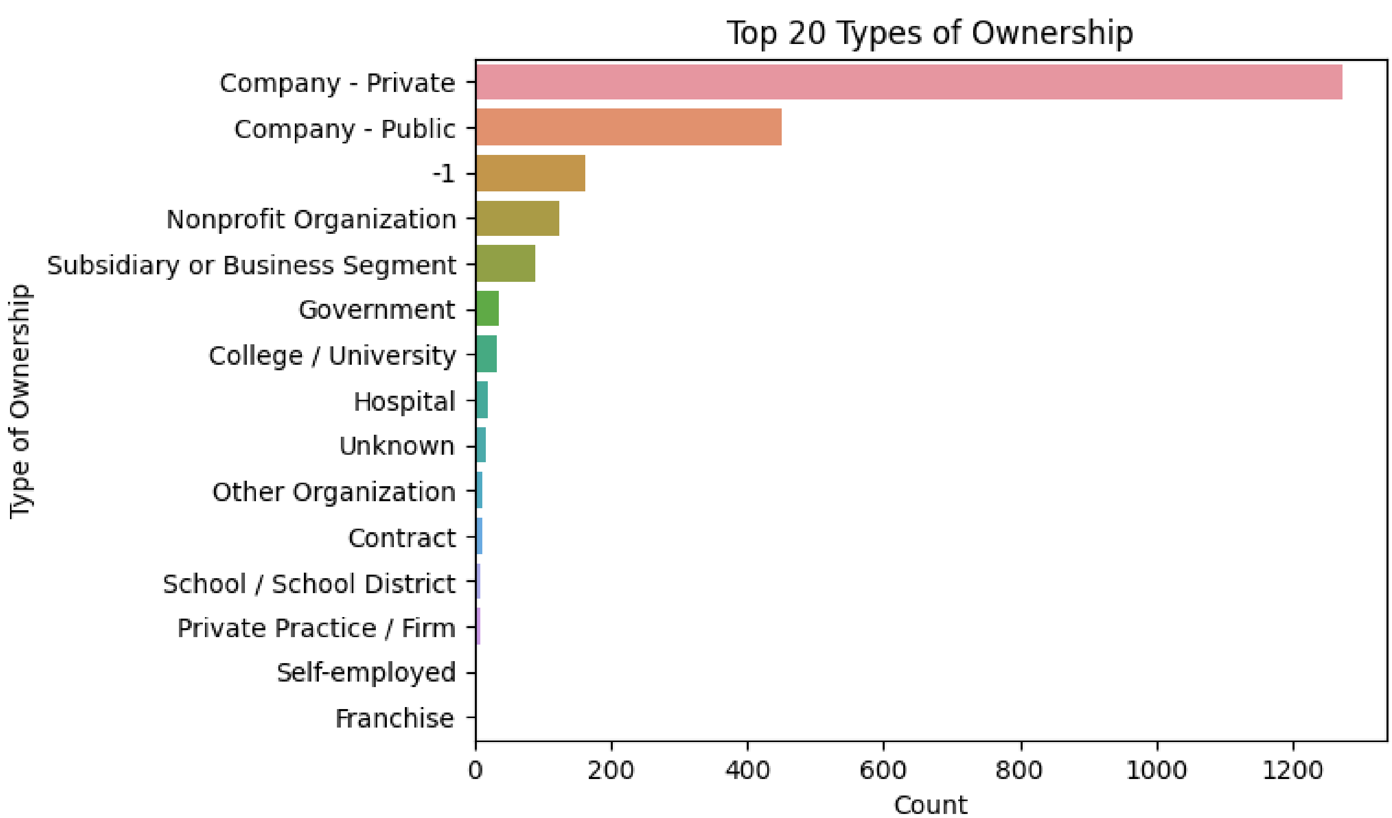
6.3 Type of Ownership

A significant amount of data falls on the Private sector, followed by public sector.

In [22]:

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

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

sns.barplot(x=TOP.values, y=TOP.index)

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



7 Sector

7.1 Top Sectors

This dataset shows two sets of data. One is the top sector distribution and the other one is in correlation with the Average Salary.

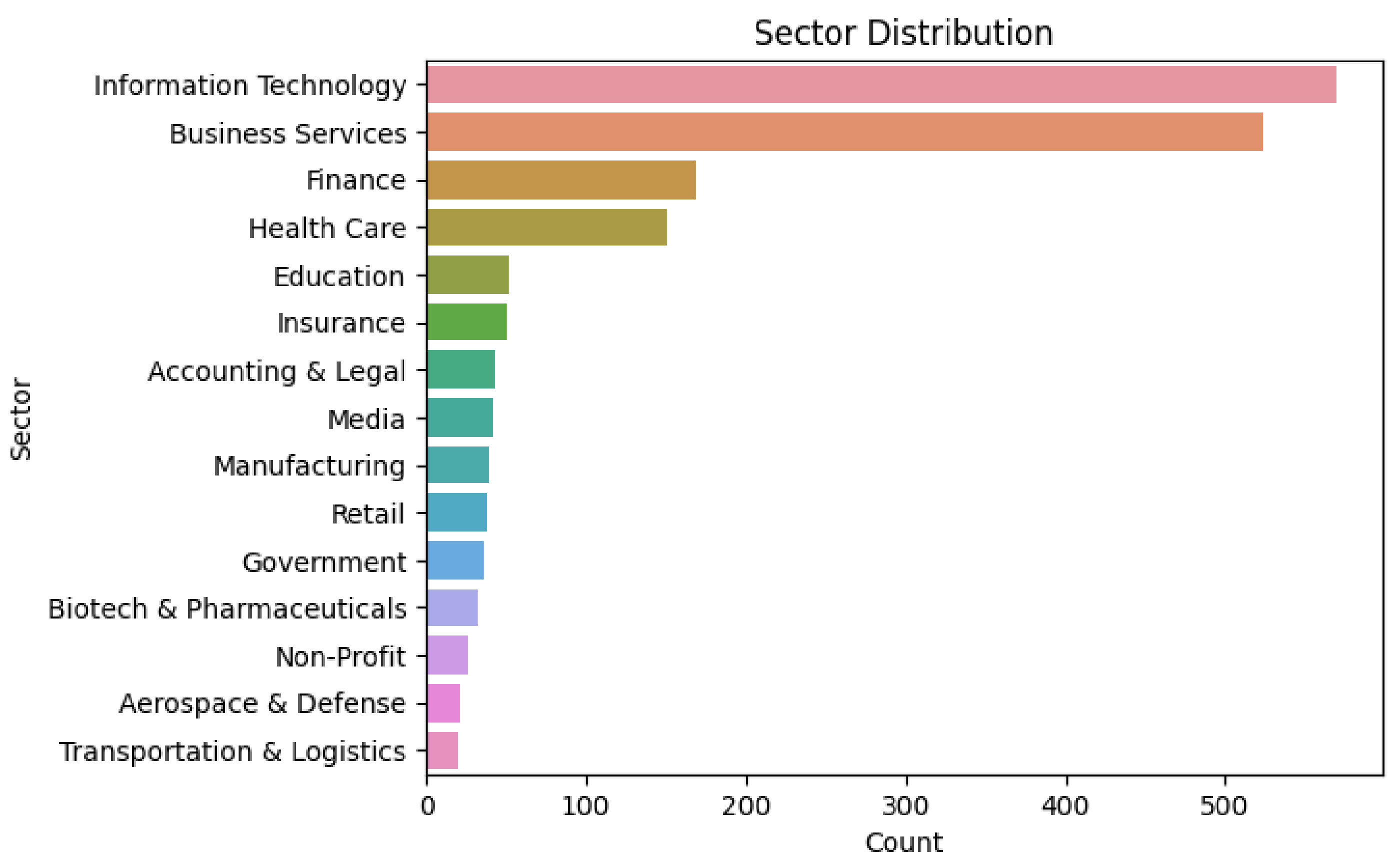
Top 5 Sectors Distribution Where Data Analyst Jobs are available

1. Information Technology
2. Business Services
3. Finance
4. Health Care
5. Education

In [23]:

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

sns.barplot(x=data_analyst_jobs_sector.values,
y=data_analyst_jobs_sector.index)
plt.xlabel('Count')
plt.ylabel('Sector')
```

```python
plt.title('Sector Distribution')
plt.show()
```



7.2 Average Salary by Sector

Top Sectors in Correlation with Average Salary

1. Biotech & Pharmaceuticals
2. Real Estate
3. Art, Entertainment & Recreation
4. Accounting & Legal
5. Information Technology

Observations

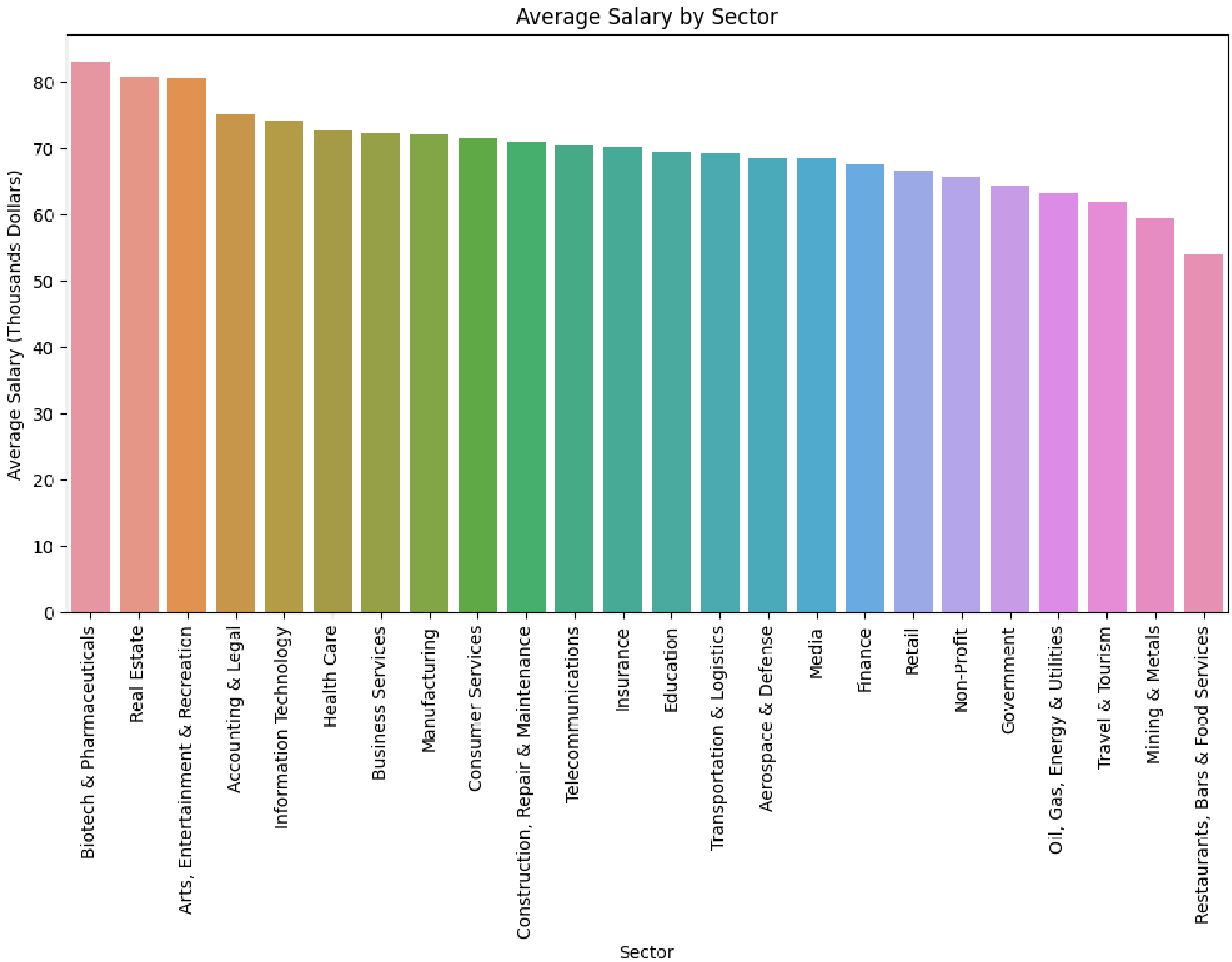
Information Technology and Business Services dominated the sector distribution. On the contrary, in correlation with the average salary, the information technology only fell at the 5th place where the average salary is between 70K-75K annually. Biotech & Pharmaceuticals showed that this sector is the highest paying sector which pays more than 80K annually.

The graph showed very distinct difference betweent the sector distribution and average salary by sector.

In [24]:

```
# Salary by Sector

average_salary_by_sector =
data_analyst_jobs[data_analyst_jobs['sector'] !=
'-1'].groupby('sector')['average_salary'].mean().reset_index()

average_salary_by_sector =
average_salary_by_sector.sort_values(by='average_salary',
ascending=False)

plt.figure(figsize=(12, 6))
sns.barplot(x='sector', y='average_salary',
data=average_salary_by_sector)
plt.xticks(rotation=90)
plt.xlabel('Sector')
plt.ylabel('Average Salary (Thousands Dollars)')
```

```
plt.title('Average Salary by Sector')

plt.show()
```



[Reference link]