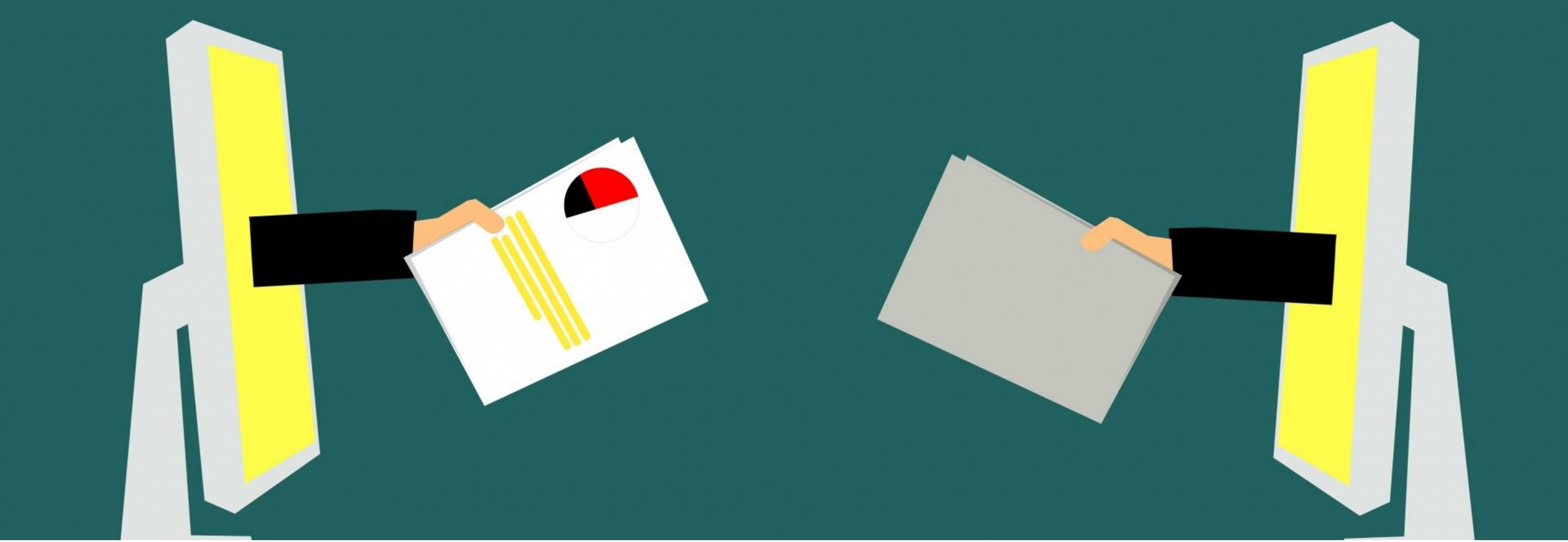


Group 1 Presentation

Jobs in Data

Are you financially curious about your future career in the Data Industry?

Our CSV

Relevant information for jobs in data including:

- *Job Titles*
- *Salary Details*
- *Experience Levels*
- *Work Year*
- *Etc..*

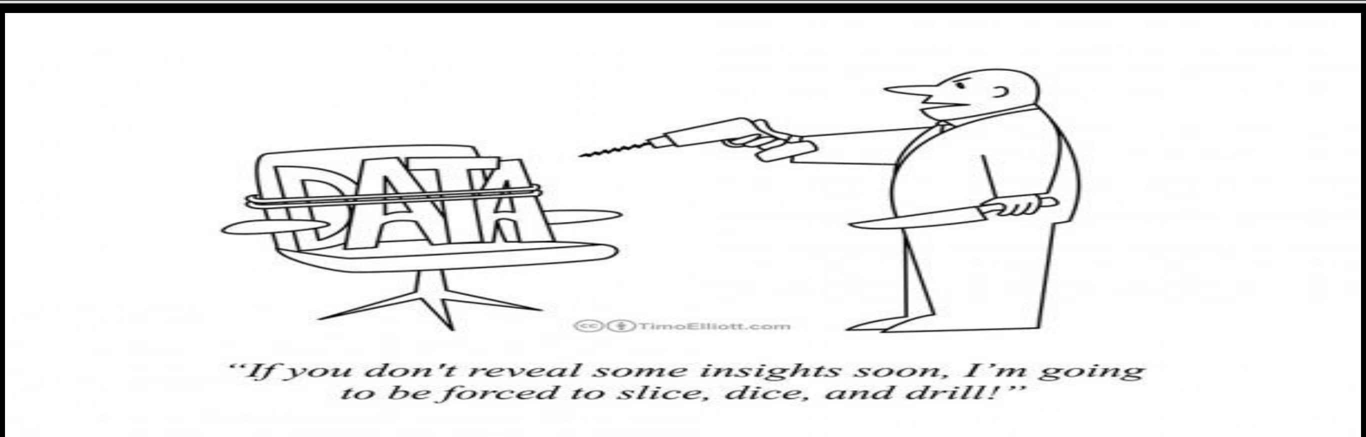
```
th
ath = r"C:\Users\Thomas\Desktop\Finished Projects\Project 7\Group-Proj

/ to Dataframe
= pd.read_csv(file_path)

Display
print(df.head())
```

```
 work_year                job_title                     job_category  \
      2023  Data DevOps Engineer                  Data Engineering
      2023        Data Architect  Data Architecture and Modeling
2     2023        Data Architect  Data Architecture and Modeling
3     2023        Data Scientist        Data Science and Research
4     2023        Data Scientist        Data Science and Research

 salary_currency  salary  salary_in_usd employee_residence experience_level  \
0            EUR   88000          95012            Germany        Mid-level
1            USD  186000         186000      United States           Senior
2            USD   81800          81800      United States           Senior
3            USD  212000         212000      United States           Senior
4            USD   93300          93300      United States           Senior

 employment_type work_setting company_location company_size
       Full-time       Hybrid          Germany            L
       Full-time    In-person    United States            M
       Full-time    In-person    United States            M
       Full-time    In-person    United States            M
       Full-time    In-person    United States            M
```

Let's explore this curiosity with some classmates!

Thomas	What are the top 3 Job Titles per year?
Chai	What size companies pay the most? Which work setting pays the most?
Amy	What country has the highest average salary in data jobs?
Jessica	Is there a correlation between high salaries and job category?

Clean the Data




```python
# Check data shape
print("Original:", df.shape)

# Remove duplicates
df = df.drop_duplicates()

# Check the shape after removing duplicates
print("After duplicates drop:", df.shape)

# Check for duplicated rows
duplicated_rows = df[df.duplicated()]
print("Duplicated rows:")
print(duplicated_rows)
```

[11] ✓ 0.0s Python

```
Original: (5341, 12)
After duplicates drop: (5341, 12)
Duplicated rows:
Empty DataFrame
Columns: [work_year, job_title, job_category, salary_currency, salary, salary_in_usd, employee_residence, experience_level, employment_type, work_setting, company_locatio
Index: []
```

```python
#Look for missing values
df.isnull().sum()
```

[12] ✓ 0.0s Python

```
work_year             0
job_title             0
job_category          0
salary_currency       0
salary                0
salary_in_usd         0
employee_residence    0
experience_level      0
employment_type       0
work_setting          0
company_location      0
company_size          0
dtype: int64
```

```python
# Display statistics
columns_of_interest = ['work_year', 'job_title']
describe_stats = df[columns_of_interest].describe(include='all')

print(describe_stats)
```

[56]

```
          work_year      job_title
count   5341.000000           5341
unique          NaN            125
top             NaN  Data Engineer
freq            NaN           1100
mean    2022.682082            NaN
std        0.608026            NaN
min     2020.000000            NaN
25%     2022.000000            NaN
50%     2023.000000            NaN
75%     2023.000000            NaN
max     2023.000000            NaN
```

```python
# Find the top 3 most common job titles per year
top_job_titles_per_year = df.groupby(['work_year', 'job_title']).size().reset_index(name='occurrences')

# Print
for year, titles in top_job_titles_per_year.groupby('work_year'):
    print(f"\nYear: {year}")
    for index, row in titles.nlargest(3, 'occurrences').iterrows():
        title = row['job_title']
        count = row['occurrences']
        print(f"{title}: {count} occurrences")
```

[57]

```
...
Year: 2020
Data Scientist: 19 occurrences
Data Engineer: 11 occurrences
Data Analyst: 6 occurrences

Year: 2021
Data Engineer: 34 occurrences
Data Scientist: 33 occurrences
Data Analyst: 19 occurrences

Year: 2022
Data Engineer: 277 occurrences
Data Scientist: 244 occurrences
Data Analyst: 168 occurrences

Year: 2023
Data Engineer: 778 occurrences
Data Scientist: 743 occurrences
Data Analyst: 551 occurrences
```

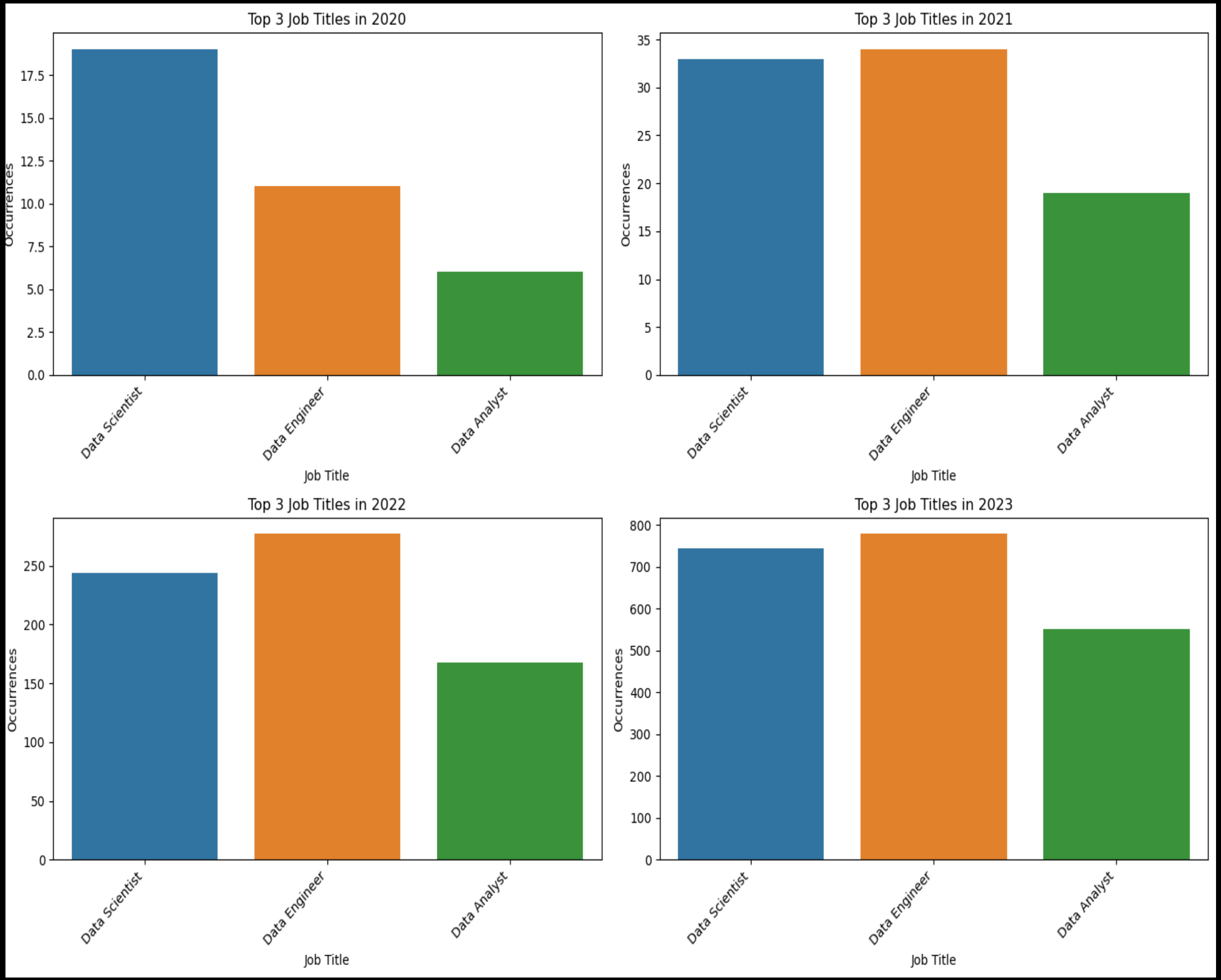
```python
# distribution of job titles
print(df['job_title'].value_counts())

# distribution of work years
print(df['work_year'].value_counts())
```

[54] Python

```
... job_title
Data Engineer                  1100
Data Scientist                 1039
Data Analyst                    744
Machine Learning Engineer       518
Analytics Engineer              207
                               ...
Deep Learning Researcher          1
Analytics Engineering Manager     1
BI Data Engineer                  1
Power BI Developer                1
Marketing Data Engineer           1
Name: count, Length: 125, dtype: int64
work_year
2023    3980
2022    1095
2021     195
2020      71
Name: count, dtype: int64
```

Sort the information

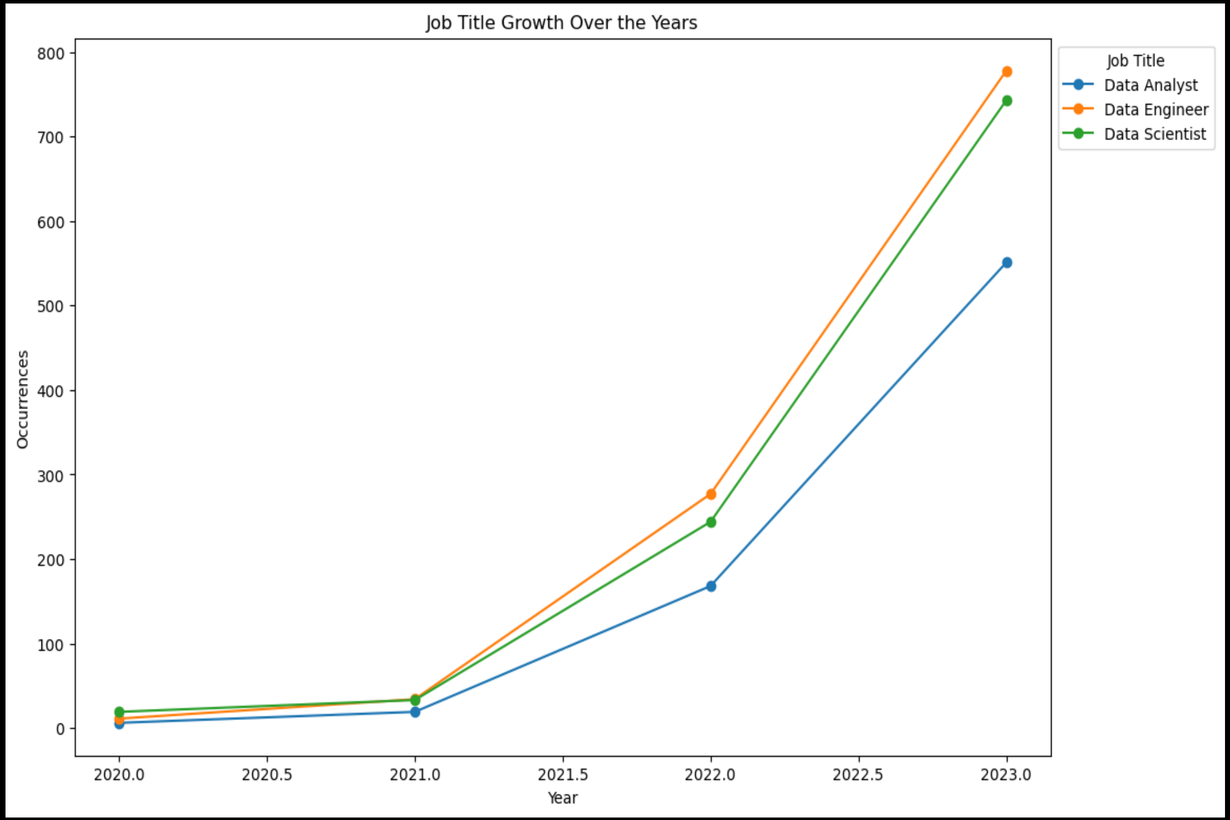
Bar Plot – Top 3 Job Titles by year




```python
# Find the top 3 most common job titles per year
top_job_titles_per_year = df.groupby('work_year')['job_title'].value_counts

# Get the unique job titles
unique_job_titles = top_job_titles_per_year.index.get_level_values('job_tit

# Create subplots for the bar graphs
fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(15, 10))

for ax, (year, data) in zip(axes.flatten(), top_job_titles_per_year.groupby
    sns.barplot(x=data.index.get_level_values('job_title'), y=data.values,
    ax.set_title(f'Top 3 Job Titles in {year}')
    ax.set_xlabel('Job Title')
    ax.set_ylabel('Occurrences')
    ax.set_xticklabels(ax.get_xticklabels(), rotation=45, ha='right')

# Adjust layout
plt.tight_layout()

plt.show()
```

Line Plot

- Showing yearly growth for the top 3 job titles




```python
# Line Plot - Job Growth over Years
plt.figure(figsize=(12, 8))
for job_title, data in top_job_titles_per_year.groupby(level=1):
    plt.plot(data.index.get_level_values('work_year'), data.values, marker='o', label=job_title)

plt.title('Job Title Growth Over the Years')
plt.xlabel('Year')
plt.ylabel('Occurrences')
plt.legend(title='Job Title', bbox_to_anchor=(1, 1))
plt.show()
```

Conclusion:

Top 3 job titles per year:

Data scientist

Data engineer

Data analyst

The analysis provides valuable insights into the distribution of job titles over the years, highlighting data engineer, scientist, and analyst roles

the cleaned dataset is now ready for more in-depth analyses, and the identified trends can help with strategic decisions in hiring and workforce planning

Question: What country has the highest average salary for jobs in data?

Step 1: Import dependencies

In [1]:

```
# libs and dependancies
import pandas as pd
from matplotlib import pyplot as plt
from scipy.stats import sem
import hvplot.pandas
```

Step 2: Read CSV file

In [2]:

```
# Read CSV into DF
job_data = pd.read_csv("../Resources/jobs_in_data.csv")
job_data.head()
```

Out[2]:

	work_year	job_title	job_category	salary_currency	salary	salary_in_usd	employee_residence	experience_level
0	2023	Data DevOps Engineer	Data Engineering	EUR	88000	95012	Germany	Mid-level
1	2023	Data Architect	Data Architecture and Modeling	USD	186000	186000	United States	Senior
2	2023	Data Architect	Data Architecture and Modeling	USD	81800	81800	United States	Senior
3	2023	Data Scientist	Data Science and Research	USD	212000	212000	United States	Senior
4	2023	Data Scientist	Data Science and Research	USD	93300	93300	United States	Senior

Step 3: Group data by country and find the average salary per group

```
# Group data by country and average salary
country_groupby = job_data.groupby("employee_residence")
country_avg_salary = round(country_groupby[['salary_in_usd']].mean(),2)
country_avg_salary
```

	salary_in_usd
employee_residence	
Algeria	100000.00
American Samoa	45555.00
Andorra	50745.00
Argentina	56444.44
Armenia	33500.00

Step 4: Reduce data by groups with 50+ data entries

```
# Count number of entries per country
country_resident_count = country_groupby[['salary_in_usd']].count()
country_resident_count = country_resident_count[(country_resident_count['salary_in_usd']>50)]
country_resident_count
```

	salary_in_usd
employee_residence	
Canada	224
France	54
Germany	66
Spain	117
United Kingdom	442
United States	8086

Step 5: Merge reduced data with average salary data

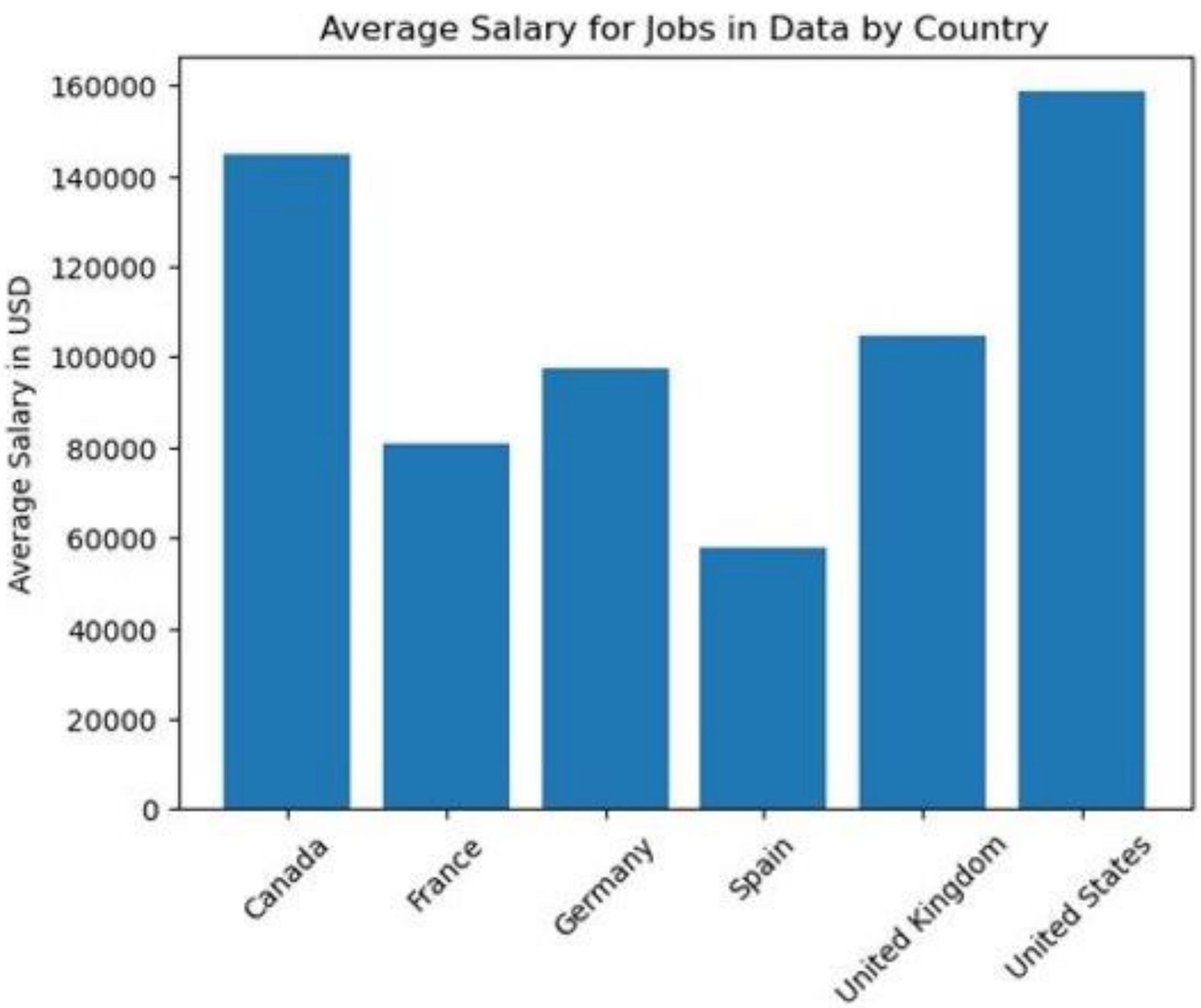
```
country_count_avg_salary = pd.merge(country_avg_salary, country_resident_count, on='employee_residence')
country_count_avg_salary = country_count_avg_salary.rename(columns=
                                                  {'salary_in_usd_x': 'Salary (USD)',
                                                   'salary_in_usd_y': 'Number of residents surveyed'})
country_count_avg_salary = country_count_avg_salary.reset_index()
country_count_avg_salary
```

	employee_residence	Salary (USD)	Number of residents surveyed
0	Canada	144743.01	224
1	France	80700.78	54
2	Germany	97640.64	66
3	Spain	58084.94	117
4	United Kingdom	104920.30	442
5	United States	158586.13	8086

Step 6: Plot the data

```
# Plot the data
x_value = country_count_avg_salary['employee_residence']
y_value = country_count_avg_salary['Salary (USD)']

plt.bar(x_value, y_value)
plt.ylabel('Average Salary in USD')
plt.title('Average Salary for Jobs in Data by Country')
plt.xticks(rotation=45)
plt.show()

# Save figure

plt.savefig('../Resources/country_avg_salary.png')
```

Answer: the United States



Question: What size companies pay the most?

```python
# Group by size of the company, experience level
grouped_by_company_size_experience_level_df = reduced_df_by_country[['salary_in_usd', 'company_size','experience_level']].groupby(['company_size','experience_level'])
# Take average pay of employees by company size and also experience level
mean_df = grouped_by_company_size_experience_level_df.mean()
# Reformatting salary_in_usd to make sure we only upto cents precision
mean_df
```

:

company_size	experience_level	salary_in_usd
L	Entry-level	103209.306122
	Executive	242048.444444
	Mid-level	145119.885417
	Senior	172673.888060
M	Entry-level	104379.806569
	Executive	192918.004292
	Mid-level	128905.978774
	Senior	165855.301396
S	Entry-level	83746.200000
	Executive	249000.000000
	Mid-level	105881.238095
	Senior	127318.181818

:
```python
mean_df.plot(kind='bar', figsize=(10,6), ylabel='Salary in USD')
```

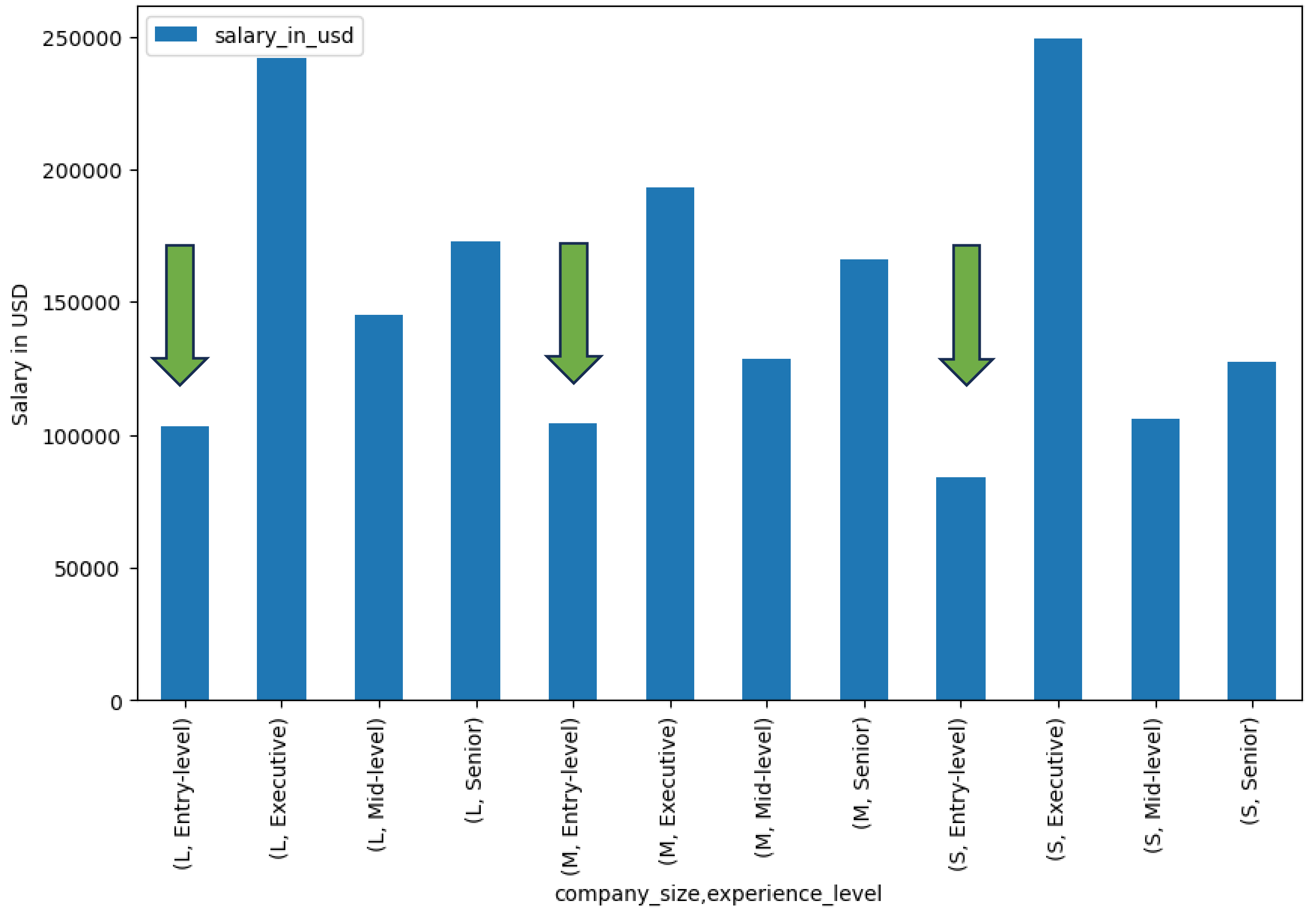
:
```
<Axes: xlabel='company_size,experience_level', ylabel='Salary in USD'>
```

Question: Which work setting pays the most?

```
grouped_by_work_setting = reduced_df_by_country[['salary_in_usd','work_setting', 'experience_level']].groupby(['work_setting', 'experience_level'])
average_salary_by_work_setting = grouped_by_work_setting.mean()
average_salary_by_work_setting
```

work_setting	experience_level	salary_in_usd
Hybrid	Entry-level	93243.047619
	Mid-level	163500.000000
	Senior	183454.545455
In-person	Entry-level	112381.229592
	Executive	184305.482517
	Mid-level	133667.806660
	Senior	168922.274801
Remote	Entry-level	90058.904762
	Executive	211346.087379
	Mid-level	120441.013453
	Senior	161276.978591

Question: What size companies pay the most?



Question: Which work setting pays the most?

Is there a correlation between High Salaries and Job Category?

Getting the analysis started

[2]:
```
# libs and dependancies
import pandas as pd
from matplotlib import pyplot as plt
from scipy.stats in
import hvplot.panda
import numpy as np
```

[3]:
```
# Read CSV into DF
job_data = pd.read_csv("../Resources/jobs_in_data.csv")
job_data.head()
```

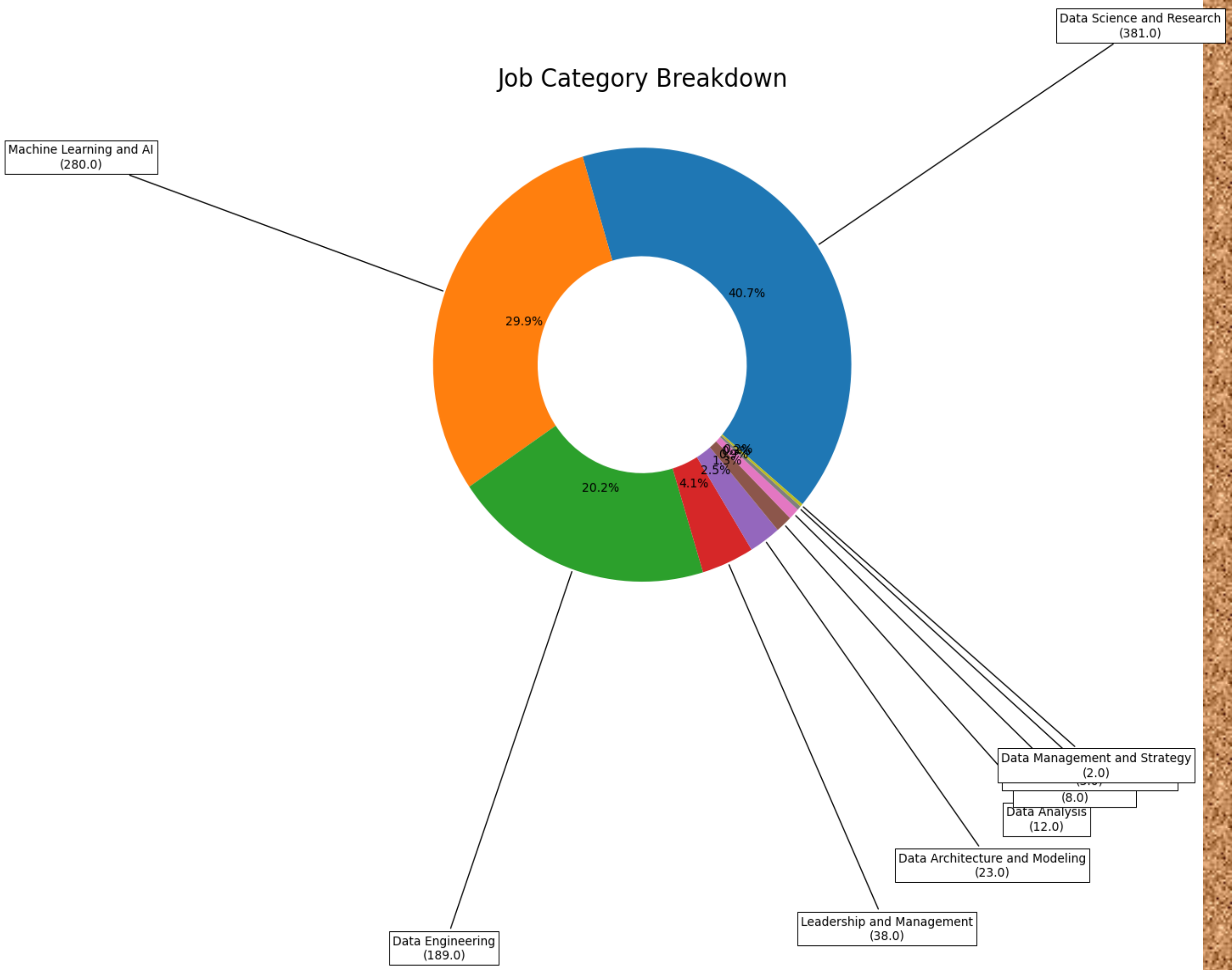
[3]:

	work_year	job_title	job_category	salary_currency	salary	salary_in_usd	employee_residence	experience_level	employment_type	work_setting	company_location	company_size
0	2023	Data DevOps Engineer	Data Engineering	EUR	88000	95012	Germany	Mid-level	Full-time	Hybrid	Germany	L
1	2023	Data Architect	Data Architecture and Modeling	USD	186000	186000	United States	Senior	Full-time	In-person	United States	M
2	2023	Data Architect	Data Architecture and Modeling	USD	81800	81800	United States	Senior	Full-time	In-person	United States	M
3	2023	Data Scientist	Data Science and Research	USD	212000	212000	United States	Senior	Full-time	In-person	United States	M
4	2023	Data Scientist	Data Science and Research	USD	93300	93300	United States	Senior	Full-time	In-person	United States	M

[25]:
```
job_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9355 entries, 0 to 9354
Data columns (total 12 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   work_year          9355 non-null   int64
 1   job_title          9355 non-null   object
 2   job_category       9355 non-null   object
 3   salary_currency    9355 non-null   object
 4   salary             9355 non-null   int64
 5   salary_in_usd      9355 non-null   int64
 6   employee_residence 9355 non-null   object
 7   experience_level   9355 non-null   object
                                        object
                                        ject
```

Top 10 % of Salaries



90^{th} Percentile = $233,800

10% of Data = 935 entries

```
salary90 = job_data["salary_in_usd"].quantile(0.9)
salary90
```

233800.00000000017

```
high_salary_filter_s = job_data["salary_in_usd"] >= salary90
high_salary_df = job_data.loc[high_salary_filter_s, ['job_category', 'salary_in_usd']]
high_salary_df
```

	job_category	salary_in_usd
17	Data Science and Research	300000
18	Data Science and Research	234000
25	Machine Learning and AI	266500
29	Machine Learning and AI	273400
39	Data Engineering	247300
...	...	...
9287	Data Science and Research	416000
9304	Data Science and Research	325000
9336	Data Science and Research	235000
9348	Machine Learning and AI	423000
9351	Data Science and Research	412000

936 rows × 2 columns

	Job Categories	Salaries	%
0	Data Science and Research	381	0.407051
1	Machine Learning and AI	280	0.299145
2	Data Engineering	189	0.201923
3	Leadership and Management	38	0.040598
4	Data Architecture and Modeling	23	0.024573
5	Data Analysis	12	0.012821
6	BI and Visualization	8	0.008547
7	Data Quality and Operations	3	0.003205
8	Data Management and Strategy	2	0.002137

Putting Data to the Test

- Box Plot
- ANOVA

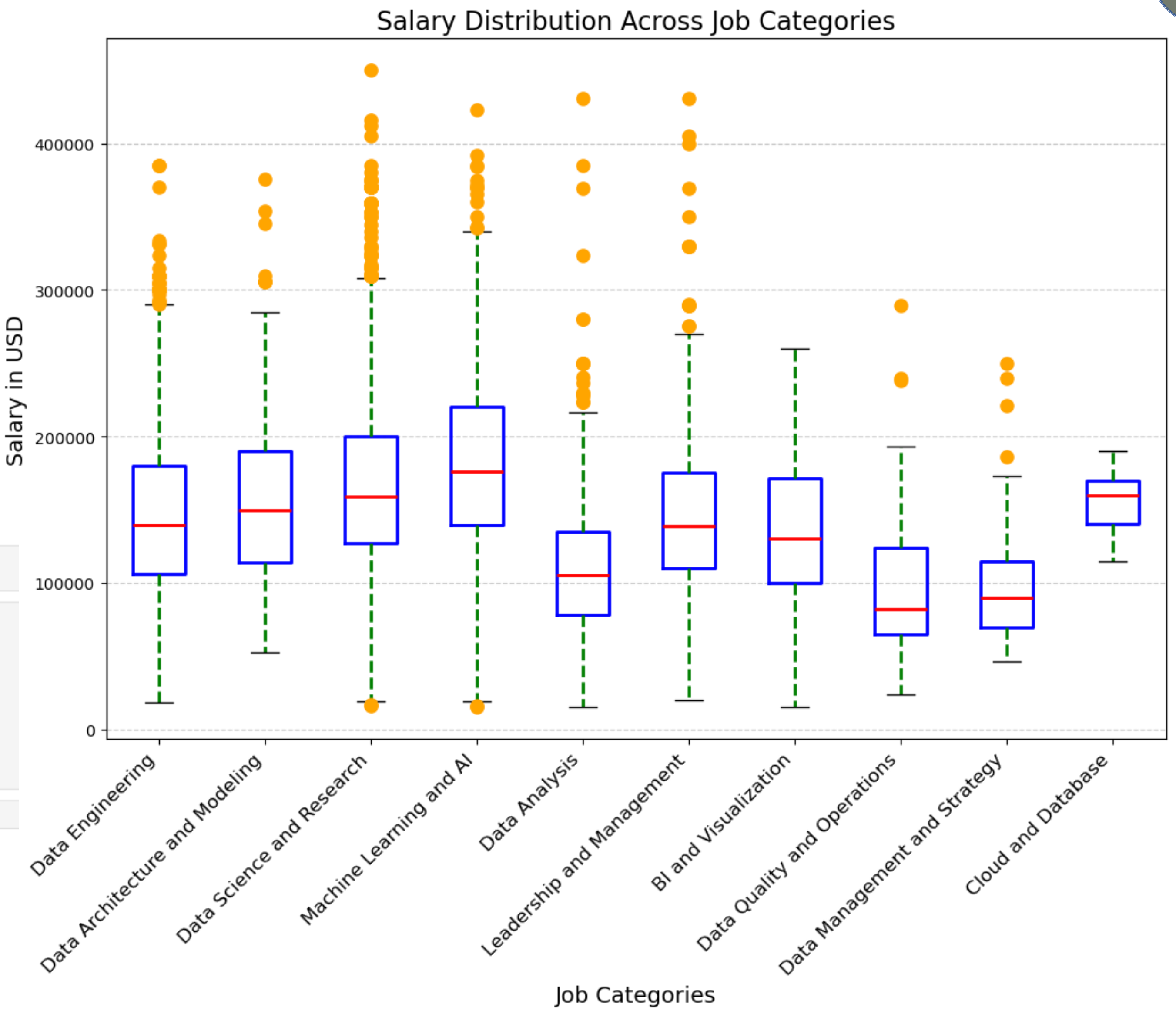
stat test ANOVA

```
import warnings
warnings.filterwarnings('ignore')
```

```
group0 = job_data[job_data['job_category'] == job_categories[0]]['salary_in_usd']
group1 = job_data[job_data['job_category'] == job_categories[1]]['salary_in_usd']
group2 = job_data[job_data['job_category'] == job_categories[2]]['salary_in_usd']
group3 = job_data[job_data['job_category'] == job_categories[3]]['salary_in_usd']
group4 = job_data[job_data['job_category'] == job_categories[4]]['salary_in_usd']
group5 = job_data[job_data['job_category'] == job_categories[5]]['salary_in_usd']
group6 = job_data[job_data['job_category'] == job_categories[6]]['salary_in_usd']
group7 = job_data[job_data['job_category'] == job_categories[7]]['salary_in_usd']
group8 = job_data[job_data['job_category'] == job_categories[8]]['salary_in_usd']
group9 = job_data[job_data['job_category'] == job_categories[9]]['salary_in_usd']
```

```
stats.f_oneway(group0, group1, group2, group3, group4, group5, group6, group7, group8, group9)
```

```
F_onewayResult(statistic=148.14691404434498, pvalue=9.32697394139812e-263)
```



F_onewayResult(statistic=148.14691404434498, pvalue=9.32697394139812e-263)

Valid Findings

- Pvalue is far less than 0.05 therefore there is no correlation between salaries and job categories in this data.

Discrepancies

- CSV data only has three years, these years were during a Global Pandemic which had unprecedented changes in the economy.
- International salaries were converted to USD but do not share the same economic characteristics as the US.

Conclusion

- Data used has limitations:
  - Entries
  - Years
  - Uneven distribution across categories (pictured below)
  - Salary analysis has to be compartmentalized by country further reducing the amount of data

```
job_data['job_category'].value_counts()
```

```
job_category
Data Science and Research        3014
Data Engineering                 2260
Data Analysis                    1457
Machine Learning and AI          1428
Leadership and Management         503
BI and Visualization              313
Data Architecture and Modeling    259
Data Management and Strategy       61
Data Quality and Operations        55
Cloud and Database                  5
Name: count, dtype: int64
```

Conclusion

- The country with the highest average salary for data jobs is the US.
- For entry-level roles, large/medium sized companies are paying the highest salary.
- Hybrid work setting is paying the most for mid and senior level roles. For entry level roles, in-person work setting is paying the most.
- The most common job categories are: Data Science and Research(32.2%), Data Engineering(24.2%), and Data Analysis(15.5%)
- The most common job titles are: Data Engineer(23.5%), Data Scientist(21.3%), and Data Analyst(14.8%)