P4DS Summative Assignment 2

Data Analysis Project

Exploring UFO Sightings: Geographical Analysis and Correlation Study

Student ID: 201804932

Email: od23aks@leeds.ac.uk

Project Plan

The Data (15 marks)

We use NUFORC(National UFO Reporting Center) data for this work. The National UFO Reporting Center is one of many sources used to compile reports of UFO activity throughout the last century and is available on Kaggle in two versions: 'scrubbed' and 'complete.' The first version contains about 80,000 items of data. The second type of data excludes entries with no location data and entries with no time data. Data "complete.csv" is the large file which size is approximately 15.35 MB. This file contains data about UFO sightings. This file has 11 columns which show different aspects of the sightings.

The column contain fields such as the date and time of the sighting, the city and state where it happened, an event description, the reported duration of the sighting, and geographical information, mainly latitude and longitude. An entry in the data entails the city and state where the observation was made and its date and time; a description of the occurrence and how long it took according to the report . Each entry in the dataset gives fields such as the place and state of the location where the sighting was done and the day and time of the sighting; a description of the occurrence and how long the reporting suggests the seeing took. From the previous notes, it could also be the case that some entries do not have correct or handle these location or time values. Furthermore, because the collection includes reports from the 20th century, some older data points may have lost information or become distorted with time. Thus, I think one should use caution when interpreting their findings, taking into account any potential biases or limitations in the data.

There good value in UFO data owing to its scientific, security, cultural and societal implications; thus, it becomes a subject of importance not only for scholars but also for policymakers and even the common man. Governments usually take keen interest in investigating UFO sighting reports as they consider it a matter related to national security and risks to flying safety. Having knowledge about what leads to UFO encounters help governments identify threats that come their way and act accordingly. On one hand, when sightings of UFOs generate curiosity among

large sections of people there are others who are left disturbed by such unexplained aerial phenomen. Hence analysis on this data would help satisfy the interest of those curious while at the same time reassuring those distressed by these events. Overall, this dataset is a valuable resource for studying the phenomenon of UFO encounters and researching associated concerns such as geographical distribution, temporal trends, and potential relationships with other variables. Researchers can get great insights into this fascinating and perplexing issue by using advanced analytical techniques and visualization tools.

Project Aim and Objectives (5 marks)

The main goal of this project is to explore and understand the patterns in UFO sightings data. I'm interested in identifying which countries report the most UFO sightings and figuring out why certain areas have a higher number of sightings. I'm curious to find out which countries have the most UFO sightings and why these countries have so many sightings. I also plan to find connections in the data, like how long the sighting lasted, what the UFO looked like, and where it was seen. The primary objective is to gain a thorough understanding of the UFO sightings dataset, using various tools to unearth intriguing insights. I intend to present these findings in a manner that is both easy to comprehend and relevant. I hope that these new discoveries will pique curiosity and encourage further research into the phenomenon of UFOs. So, let's begin by outlining the objectives we hope to accomplish with this project.

Specific Objective(s)

Objectives:

- **Objective 1:** Identify Countries with High UFO Sightings
- **Objective 2:** Analyze the dataset to determine the average values of the various data points that collectively represent a single UFO sighting
- **Objective 3:** Analyze Correlations and Relationships in the Dataset

System Design (5 marks)

Describe your code in terms of the following two sections.

Architecture

The architecture of of my code is as follows: There are multiple steps in data processing and analysis which brings out some knowledge from the dataset regarding UFO sightings. These include loading the dataset from a CSV file, preprocessing it – converting data types, dealing with missing values and cleaning up the data. After that we take this cleaned dataset to do correlation analysis and representation. Here we calculate correlation coefficients between numerical variables and create a heatmap of correlation matrix. The relationships among variables can be studied through scatter plots and box plots. Metaphorically speaking, the architecture simply goes straight from loading & pre-processing stage till correlation analysis & representation phase.

Processing Modules and Algorithms

1. Data Loading and Preprocessing: Parsing the datetime column to datetime objects: No specific model used, standard pandas datetime parsing. Converting duration (seconds)

and latitude columns to numerical data types: Pandas to_numeric function. Handling missing values and outliers: No specific model used, standard data cleaning techniques.

2. Correlation Analysis: Computing correlation coefficients between numerical variables: Pearson correlation method. No specific model, implemented using pandas DataFrame's corr function.
3. Visualization: Generating a heatmap to visualize the correlation matrix: Seaborn's heatmap function. Creating scatter plots to explore relationships between duration (seconds) and latitude: Matplotlib's scatter function. Constructing box plots to analyze the relationship between country and duration (seconds): Seaborn's boxplot function.

Program Code (25 marks)

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import numpy as np

import csv

# Open the original CSV file
with open("complete.csv", "r", encoding="utf-8") as original_file:
    # Open a new CSV file for writing
    with open("cleaned_data.csv", "w", newline="", encoding="utf-8")
as cleaned_file:
        # Create a CSV writer object
        writer = csv.writer(cleaned_file)

        # Create a CSV reader object for the original file
        reader = csv.reader(original_file)

        # Write the header row to the new file
        header = next(reader)
        writer.writerow(header)

        # Iterate over each row in the original file
        for row in reader:
            # Check if the row is valid (e.g., contains all required
fields)
            if len(row) == len(header):
                # Write the row to the new file
                writer.writerow(row)

print("New CSV file created with cleaned data.")
```

```
New CSV file created with cleaned data.
```

The CSV file originally produced some error when parsed due to misformatting, which sometimes would include extra fields in various lines. A new CSV file is created with corrected formatting issues or exclusion of problematic entries to address this concern. The code reads

data from the CSV module using efficient functionalities provided by the csv module, where data integrity will be ensured during cleaning of the misformatted fields in the file.

```
import pandas as pd

# Load the dataset
df = pd.read_csv("cleaned_data.csv")

# Display the first few rows of the dataframe
df.head()
```

```
<ipython-input-5-7726b8b2ce5f>:4: DtypeWarning: Columns (5,9) have
mixed types. Specify dtype option on import or set low_memory=False.
  df = pd.read_csv("cleaned_data.csv")
```


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 88679,\n  \"fields\":
[\n    {\n      \"column\": \"datetime\",\n      \"properties\": {\n
\"dtype\": \"string\",\n        \"num_unique_values\": 76159,\n
\"samples\": [\n          \"4/11/2014 21:20\",\n          \"11/30/2008
13:00\",\n          \"10/15/2004 14:00\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"city\",\n      \"properties\": {\n
\"dtype\": \"category\",\n        \"num_unique_values\": 22018,\n
\"samples\": [\n          \"frankfort\",\n          \"yakima indian
reservation\",\n          \"trinidad (location unspecified)\"\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"state\",\n      \"properties\":
{\n        \"dtype\": \"category\",\n        \"num_unique_values\":
68,\n        \"samples\": [\n          \"ak\",\n          \"ab\",\n
\"al\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"country\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 5,\n        \"samples\":
[\n          \"gb\",\n          \"de\",\n          \"ca\"\n        ],\
n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"shape\",\n      \"properties\":
{\n        \"dtype\": \"category\",\n        \"num_unique_values\":
29,\n        \"samples\": [\n          \"dome\",\n          \"egg\",\n
\"triangle\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"duration (seconds)\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 732,\n
\"samples\": [\n          8400.0,\n          \"35\",\n
\"1398\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"duration (hours/min)\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 9791,\n
\"samples\": [\n          \"ten mins still happening\",\n
\"~2.5\",\n          \"atleats an hour\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
```

```
n    },\n    {\n      \"column\": \"comments\",\n      \"properties\":
{\n        \"dtype\": \"string\",\n        \"num_unique_values\":
88283,\n        \"samples\": [\n          \"Two bright colored lights
came from the east and stood over Orwigsburg at 8:00. I was up at my
computer when my Mom yelled up to get do\",\n          \"Large orange
ball of light over London late 1990&#39s\",\n          \"Balls of fire
in sky\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"date posted\",\n      \"properties\": {\n        \"dtype\":
\"object\",\n        \"num_unique_values\": 317,\n        \"samples\":
[\n          \"10/15/2003\",\n          \"7/5/2013\",\n
\"2/24/2005\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"latitude\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 25428,\n
\"samples\": [\n          62.1091667,\n          \"34.9455556\",\n
38.2205556\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"longitude\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 41.421744257281766,\n        \"min\": -
176.6580556,\n        \"max\": 178.4419,\n
\"num_unique_values\": 20549,\n        \"samples\": [\n          -
80.9113889,\n          -122.2708333,\n          -88.8183333\
n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    }\n  ]\
n}","type":"dataframe","variable_name":"df"}
```


In this code block, i am just viewing the cleaned data with the help of pandas dataframe(df). I am viewing the first few entries usin df.head. The data and columns is already discussed in data section earlier.

```python
# Count the number of sightings by country
country_counts = df['country'].value_counts()

total_sightings = country_counts.sum()

country_counts_sorted = country_counts.sort_values(ascending=False)

# Calculate the percentage of sightings for each country
country_percentages = (country_counts_sorted / total_sightings) * 100

# Display the top countries with the highest number of sightings and
their percentages
top_countries = country_counts_sorted.head(5)
top_countries_percentages = country_percentages.head(5)

# Combine the top countries and their percentages into a single
DataFrame
top_countries_data = pd.DataFrame({'Number of Sightings':
```

```
top_countries, 'Percentage': top_countries_percentages})

# Display the combined DataFrame
print("Top Countries with the Highest Number of UFO Sightings:")
print(top_countries_data)
```

```
Top Countries with the Highest Number of UFO Sightings:
         Number of Sightings  Percentage
country
us                     70293   92.110229
ca                      3266    4.279687
gb                      2050    2.686270
au                       593    0.777053
de                       112    0.146762
```

The analysis showed the the quantity and proportion of UFO sightings by nation. The following are the top nations where UFO sightings have occurred:

1. United States (US)
2. Canada (CA)
3. Australia (AU)
4. Germany (DE)

```
import csv

# Initialize lists to store numerical data
duration_seconds = []
latitude = []
longitude = []

# Open the CSV file
with open("cleaned_data.csv", "r", encoding="utf-8") as file:
    # Create a CSV reader object
    reader = csv.reader(file)

    # Skip the header row
    next(reader)

    # Iterate over each row in the CSV file
    for row in reader:
        # Extract numerical data
        try:
            duration_sec = float(row[5])
            lat = float(row[9])
            lon = float(row[10])

            # Append numerical data to lists
            duration_seconds.append(duration_sec)
```

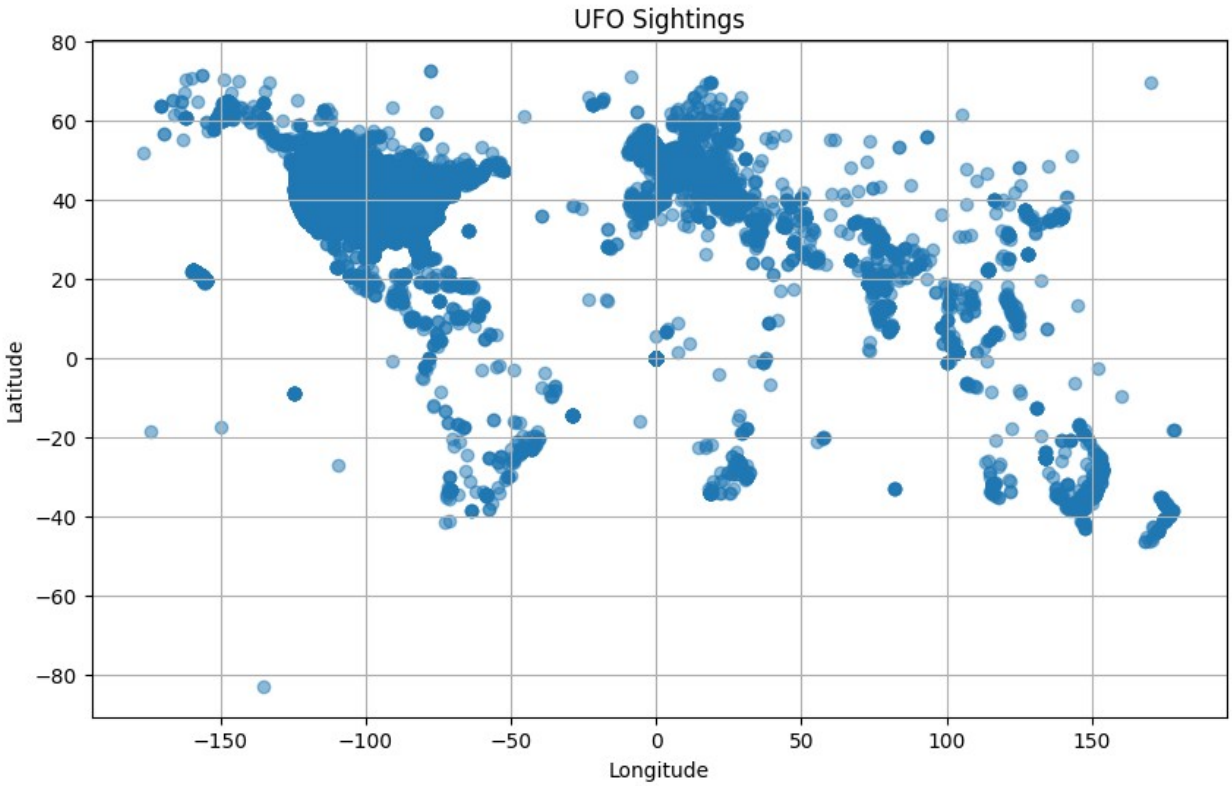
```
            latitude.append(lat)
            longitude.append(lon)
        except ValueError:
            continue  # Skip rows with non-numeric latitude or
longitude

# Calculate statistics
num_sightings = len(duration_seconds)
avg_duration_sec = sum(duration_seconds) / num_sightings
min_duration_sec = min(duration_seconds)
max_duration_sec = max(duration_seconds)
avg_latitude = sum(latitude) / num_sightings
avg_longitude = sum(longitude) / num_sightings

# Print statistics
print("Number of UFO sightings:", num_sightings)
print("Average duration of UFO sightings (seconds):",
avg_duration_sec)
print("Minimum duration of UFO sightings (seconds):",
min_duration_sec)
print("Maximum duration of UFO sightings (seconds):",
max_duration_sec)
print("Average latitude of UFO sightings:", avg_latitude)
print("Average longitude of UFO sightings:", avg_longitude)
```

```
Number of UFO sightings: 88673
Average duration of UFO sightings (seconds): 8392.012233498357
Minimum duration of UFO sightings (seconds): 0.0
Maximum duration of UFO sightings (seconds): 97836000.0
Average latitude of UFO sightings: 37.45350608523135
Average longitude of UFO sightings: -85.02192242181938
```

In the above code i tried to represent the data of cleaned UFO sightings from a CSV file and does various calculations considering numerical fields. It creates some lists in order to store numbers like duration (in seconds), latitude, longitude and so on. Then it opens the CSV file and for each row, retrieves number from specific columns (index 5, 9, 10) and appends them into respective lists. In the try block, values are tried to be converted into float type that has drawn out from rows with catch any value error exceptions may occur when they are not numeric figures. After all rows have been processed through; sighting number of UFOs , average duration as well as minimum-maximum durations among other things such like average longitudes or latitudes are statistically calculated. Finally these stats will be displayed by printing them out on screen . Therefore this piece codes gives an idea about distribution of UFOs sightings based on numbers which have been cleaned up already .

```
import csv
import matplotlib.pyplot as plt

# Initialize lists to store latitude and longitude data
latitude = []
```

```
longitude = []

# Open the CSV file
with open("cleaned_data.csv", "r", encoding="utf-8") as file:
    # Create a CSV reader object
    reader = csv.reader(file)

    # Skip the header row
    next(reader)

    # Iterate over each row in the CSV file
    for row in reader:
        # Extract latitude and longitude data
        try:
            lat = float(row[9])
            lon = float(row[10])

            # Append latitude and longitude to lists
            latitude.append(lat)
            longitude.append(lon)
        except ValueError:
            continue  # Skip rows with non-numeric latitude or
longitude

# Plot the latitude and longitude data
plt.figure(figsize=(10, 6))
plt.scatter(longitude, latitude, marker='o', alpha=0.5)
plt.xlabel('Longitude')
plt.ylabel('Latitude')
plt.title('UFO Sightings')
plt.grid(True)
plt.show()
```



The above code reads this information and produces a scatter plot that demonstrates where these sightings are concentrated geographically. It doesn't simply read the data but goes through each row of the file one by one, extracting both the latitude and longitude values. These are stored in separate lists which are later used by Matplotlib for plotting on a scatter plot — longitude on the x-axis, latitude on the y-axis. You can get some valuable information from this plotted data: be it heavily concentrated regions or any peculiar patterns that might be visible across different geographic areas.

The plot resembles a global map since latitude and longitude coordinates are commonly used to illustrate locations on Earth's surface. Plotting latitude and longitude on the y- and x-axes, respectively, allows the sightings to be effectively mapped into a two-dimensional representation of the Earth's surface.

The analysis showed the the quantity and proportion of UFO sightings by nation. The following are the top nations where UFO sightings have occurred:

1. United States (US)
2. Canada (CA)
3. Australia (AU)
4. Germany (DE)

```
# Convert 'duration (seconds)' column to numeric
df['duration (seconds)'] = pd.to_numeric(df['duration (seconds)'],
errors='coerce')
```

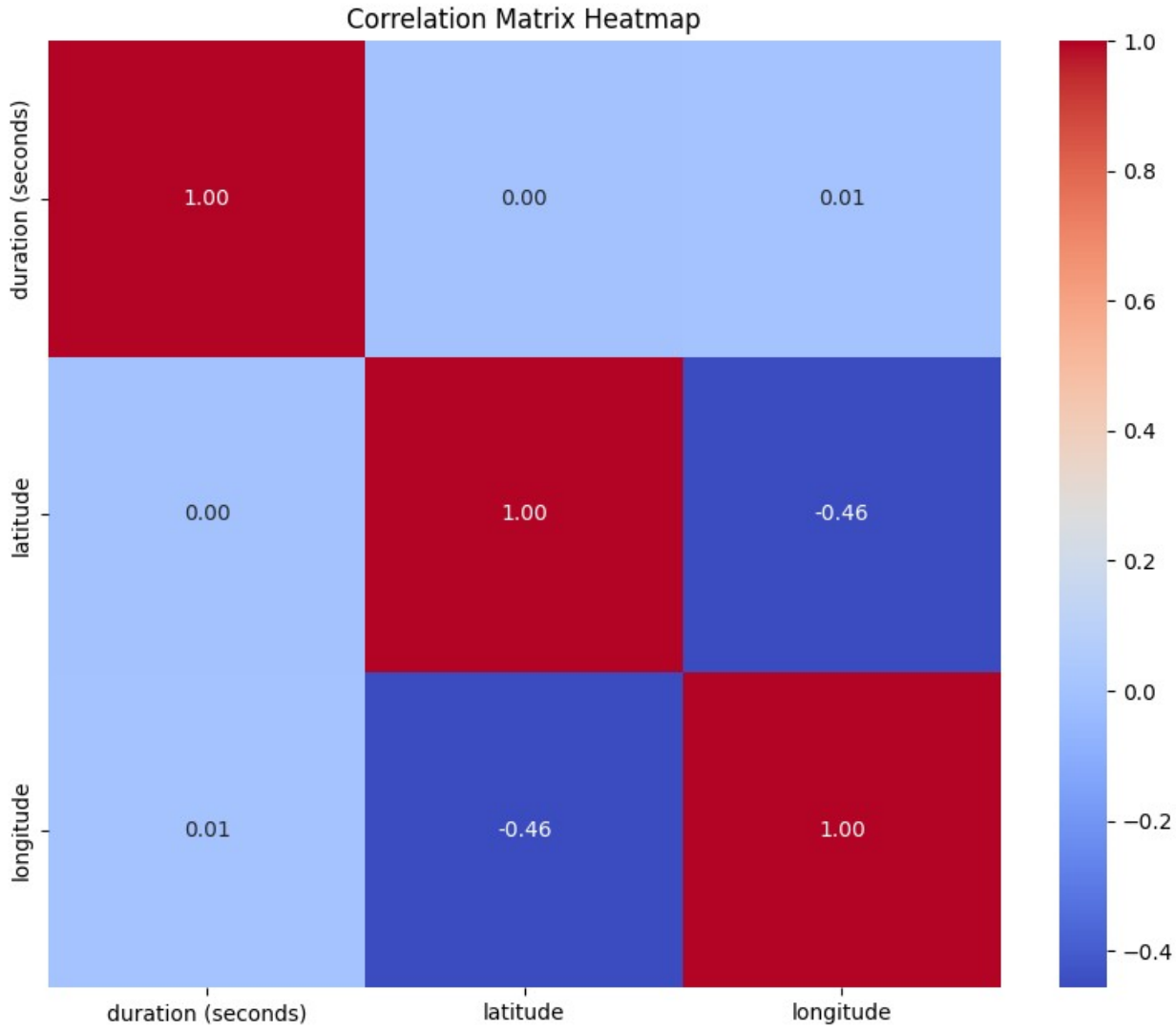
```
# Check the data types again
print(df.dtypes)
```

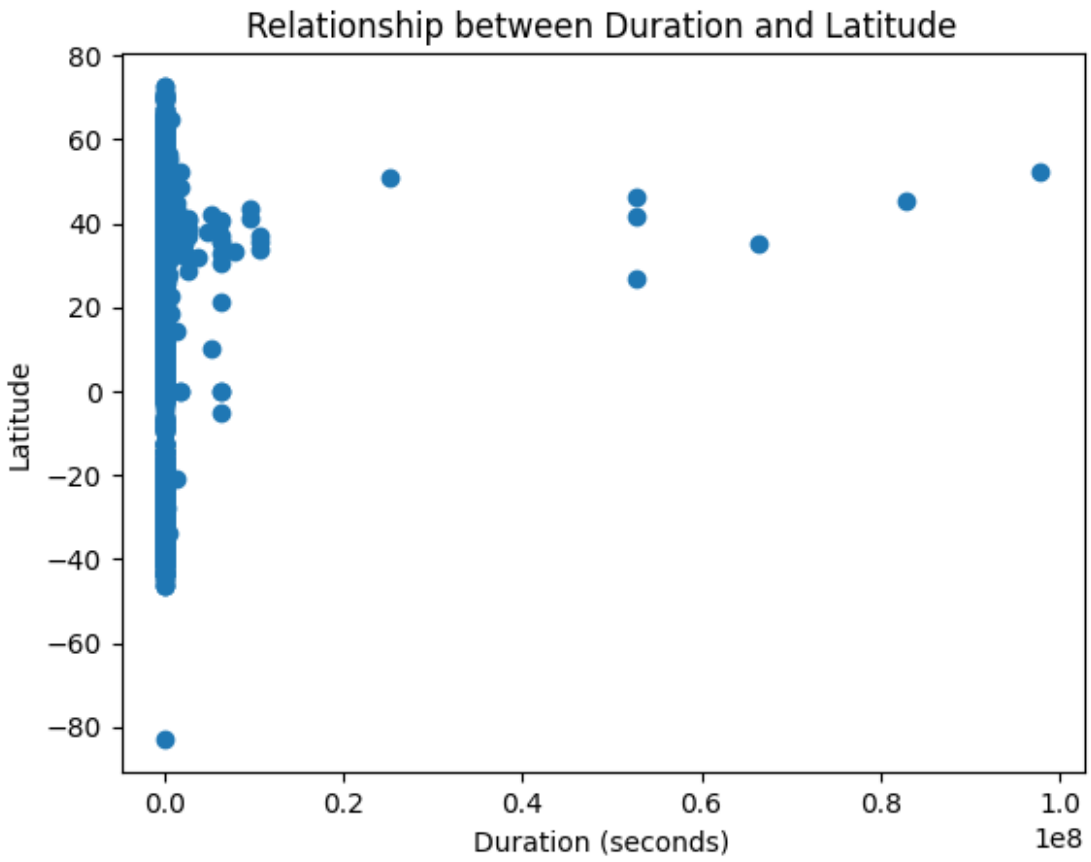
```
datetime                  object
city                      object
state                     object
country                   object
shape                     object
duration (seconds)       float64
duration (hours/min)      object
comments                  object
date posted               object
latitude                  object
longitude                float64
dtype: object
```

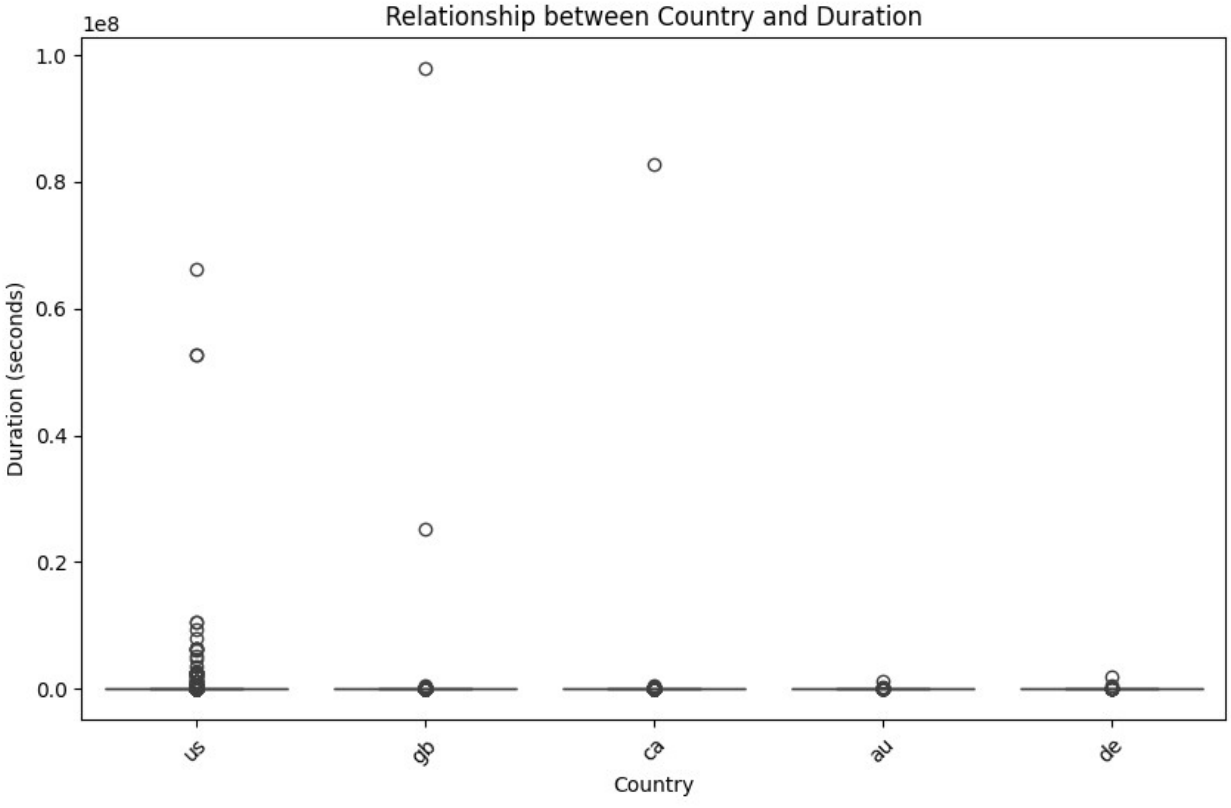
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load the dataset
df = pd.read_csv("cleaned_data.csv")

# Convert datetime column to datetime object with specified format
df['datetime'] = pd.to_datetime(df['datetime'], format='%m/%d/%Y %H:
%M', errors='coerce')

# Clean 'duration (seconds)' column: extract numeric values and
convert to float, handle errors
df['duration (seconds)'] = pd.to_numeric(df['duration (seconds)'],
errors='coerce')

# Clean 'latitude' column: convert to numeric and handle errors
df['latitude'] = pd.to_numeric(df['latitude'], errors='coerce')

# Clean 'longitude' column: convert to numeric and handle errors
df['longitude'] = pd.to_numeric(df['longitude'], errors='coerce')

# Drop rows with NaN values in 'duration (seconds)', 'latitude', or
'longitude' columns
df.dropna(subset=['duration (seconds)', 'latitude', 'longitude'],
inplace=True)

df.drop(columns=['state', 'city', 'comments', 'shape'], inplace=True)

# Exclude non-numeric columns from correlation analysis
numeric_columns = df.select_dtypes(include=['float64',
'int64']).columns
numeric_df = df[numeric_columns]
```

```
# Check for Correlation
correlation_matrix = numeric_df.corr()

# Visualize Correlation Matrix Heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm',
fmt=".2f")
plt.title('Correlation Matrix Heatmap')
plt.show()

# Explore Relationships using Scatter Plot
plt.scatter(df['duration (seconds)'], df['latitude'])
plt.xlabel('Duration (seconds)')
plt.ylabel('Latitude')
plt.title('Relationship between Duration and Latitude')
plt.show()

# Explore Relationships using Box Plot
plt.figure(figsize=(10, 6))
sns.boxplot(x='country', y='duration (seconds)', data=df)
plt.xlabel('Country')
plt.ylabel('Duration (seconds)')
plt.title('Relationship between Country and Duration')
plt.xticks(rotation=45)
plt.show()
```

```
<ipython-input-10-cb43d4c3e858>:6: DtypeWarning: Columns (5,9) have
mixed types. Specify dtype option on import or set low_memory=False.
  df = pd.read_csv("cleaned_data.csv")
```


Correlation Matrix Heatmap
duration (seconds)
latitude
longitude
1.00
0.00
0.01
0.00
1.00
-0.46
0.01
-0.46
1.00
duration (seconds)
latitude
longitude
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4


Relationship between Duration and Latitude
Latitude
80
60
40
20
0
−20
−40
−60
−80
0.0
0.2
0.4
0.6
0.8
1.0
1e8
Duration (seconds)


Relationship between Country and Duration
1e8
Duration (seconds)
1.0
0.8
0.6
0.4
0.2
0.0
us
gb
ca
au
de
Country

The above Python script takes a distinctive approach — leveraging Pandas and Seaborn as well as Matplotlib — to unravel findings from a UFO sightings dataset. It first loads data stored within a CSV file, then conducts a pre-cleaning phase on the dataset before computing the correlation matrix that will expose hidden relationships among numerical variables. The resulting visual is a heatmap revealing these associations; subsequently more specific details about select relationships are explored through scatter plots and box plots superimposed onto it. This combined visualization strategy can bring forth hitherto unseen patterns or peculiar trends lying latent within our UFO sighting data set— thereby fostering greater insight generation and possibly even facilitating certain aspects of further analysis.

Project Outcome (10 + 10 marks)

Overview of Results

We realized some interesting facts from our study of the dataset on UFO sightings. The first thing we noticed is that most of these incidents take place in North America, particularly the United States which has more cases reported than Canada and United Kingdom combined. This implies that there might be a certain geographical component to witnessing unidentified flying objects; some areas are more likely to experience them than others. I also attempted to construct an image of an average UFO sighting in the form of average values of a significant number of characteristics found in the dataset. In addition, various factors were considered by us during correlation analysis where no relationship between duration of a sighting and its probability was found though correlations between latitude and longitude of where it occurred with country were established thereby underscoring geographical significance in this field. Visualization tools like scatter plots and heatmaps were used to better understand the data. These graphical representations enable us see trends or patterns easily.

Objective 1: Identify Countries with High UFO Sightings

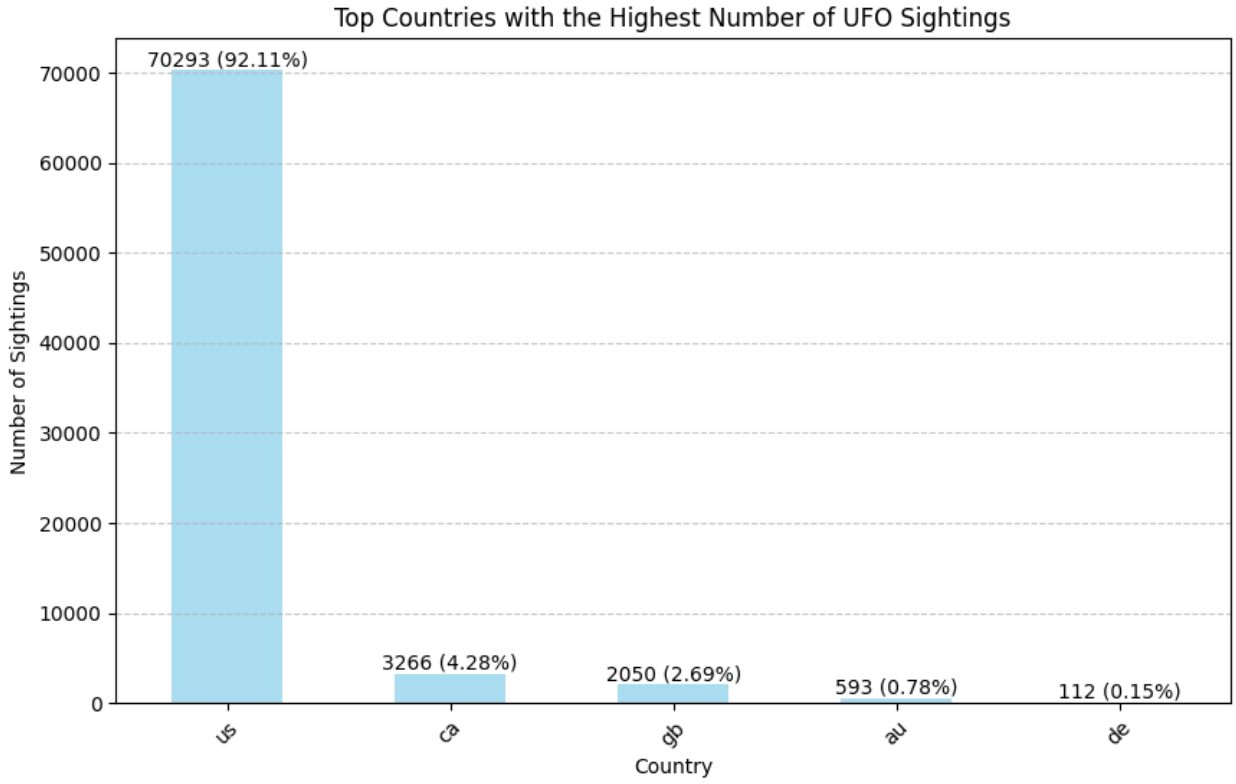
Explanation of Results

Following my analysis, I was able to identify and name the countries with the most numbers of the UFO sightings. The United States falls among the first three hundred highest-ranking in the percentage of UFO sightings found in the document at 92.11% . It means that more UFO encounters were spotted in this country more than other counties. The remaining countries that reported the most UFO cases around the globe are Canada, the United Kingdom, and Australia and Germany, arranged in less descending order. The data file percentages of UFO sightings were 4.28%, 2.69%, 0.78%, and 0.15%, respectively, from which they experienced the cases. I have learned that while the United States has high case percentages, other countries have also had their shares. Thus, we can conclude that UFO fixation is a global phenomenon that continues to capture people's minds.

Visualisation

The following bar chart gives a vivid representation of the top countries with highest UFO sightings

```
# Plot the distribution of sightings by country
plt.figure(figsize=(10, 6))
ax = top_countries.plot(kind='bar', color='skyblue', alpha=0.7)
plt.title('Top Countries with the Highest Number of UFO Sightings')
plt.xlabel('Country')
plt.ylabel('Number of Sightings')
plt.xticks(rotation=45)
plt.grid(axis='y', linestyle='--', alpha=0.7)

# Add percentages to the bar plot
for i, (num_sightings, percentage) in enumerate(zip(top_countries,
top_countries_percentages)):
    plt.text(i, num_sightings, f'{num_sightings} ({percentage:.2f}%)',
ha='center', va='bottom')

plt.show()
```



Objective 2: Analyze the dataset to determine the average values of the various data points that collectively represent a single UFO sighting

Explanation of Results

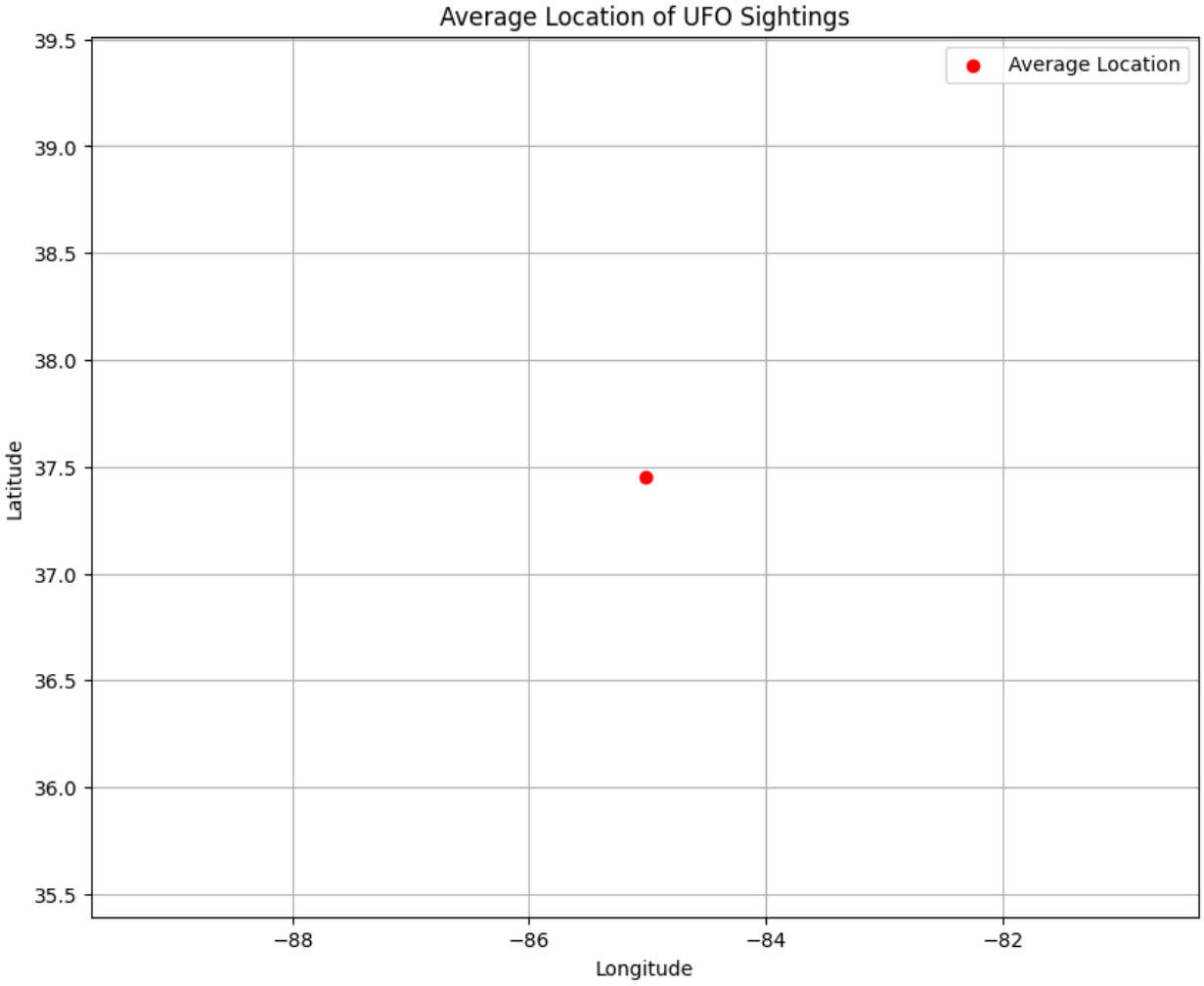
To get a general understanding of UFO sightings, the goal of our analysis is to assert the average characteristics under multiple data points. Total of 88,673 sightings were recorded. The average duration of a sighting estimated was 8,392 seconds or 140 minutes. However a point to be noted that, this estimate hides large disparities from a split-second sighting to lasting of over a day. We also differentiate averages under the geographic coordinates. The average latitude is close to 37.45, and the average longitude is near -85.02 . The data give a picture of where the sightings are most prevalent.

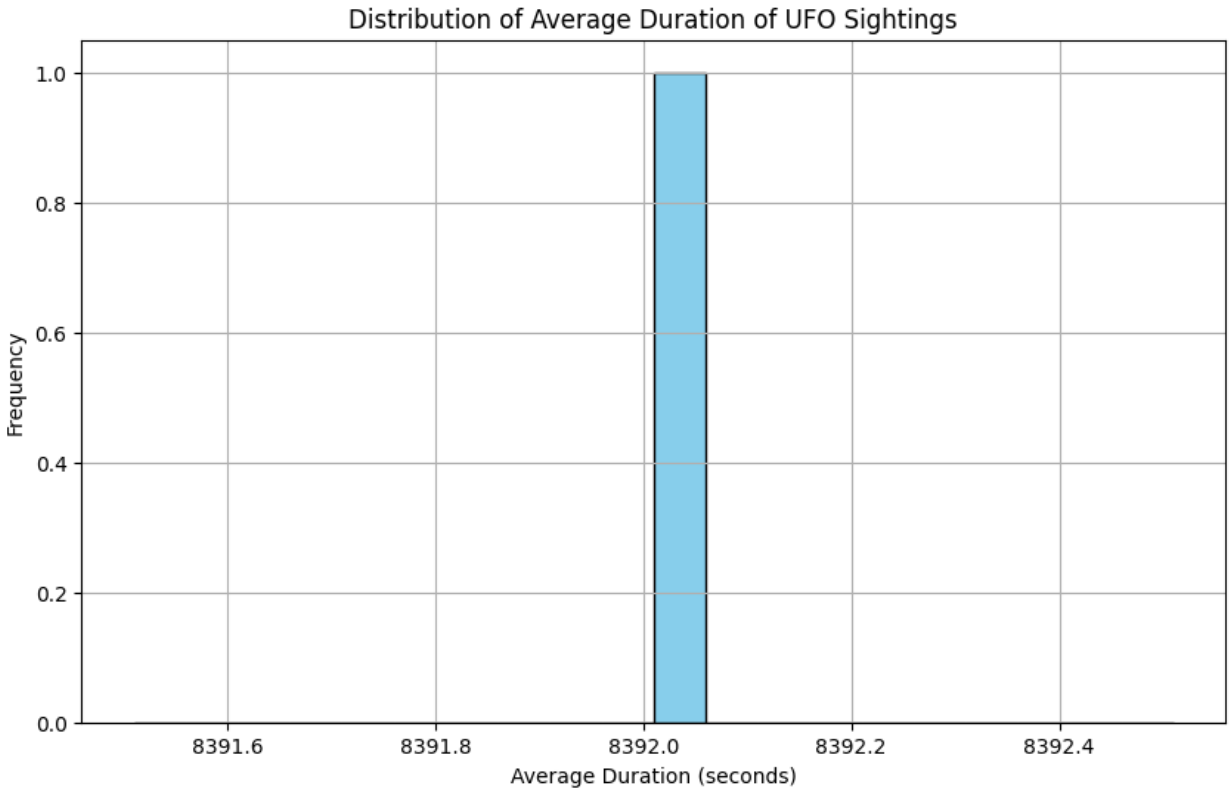
Summing up, I attempted to construct an image of an average UFO sighting in the form of average values of a significant number of characteristics found in the dataset. According to the data, an UFO sighting usually occur in USA and last a little over 8392 seconds or 140 minutes, although both brief glimpses and extremely long events are possible. As for the location, the most probable seeming latitudes are 37.45 and the longitude is about -85.02. Of course, there are still hundreds of variations, so there is no reason to assert the reliability of any of the above: these examples were used for a general overview of the fields. It underlines the wide range of variations in sightings

Visualisation

```
import matplotlib.pyplot as plt

# Average latitude and longitude of UFO sightings
avg_latitude = 37.4535
avg_longitude = -85.0219

# Plot the average location on a map
plt.figure(figsize=(10, 8))
plt.scatter(avg_longitude, avg_latitude, color='red', marker='o',
label='Average Location')
plt.xlabel('Longitude')
plt.ylabel('Latitude')
plt.title('Average Location of UFO Sightings')
plt.legend()
plt.grid(True)
plt.show()

# Average duration of UFO sightings
avg_duration = 8392.01

# Plot histogram of average duration
plt.figure(figsize=(10, 6))
plt.hist(avg_duration, bins=20, color='skyblue', edgecolor='black')
```

```
plt.xlabel('Average Duration (seconds)')
plt.ylabel('Frequency')
plt.title('Distribution of Average Duration of UFO Sightings')
plt.grid(True)
plt.show()
```





Objective 3: Analyze Correlations and Relationships in the Dataset

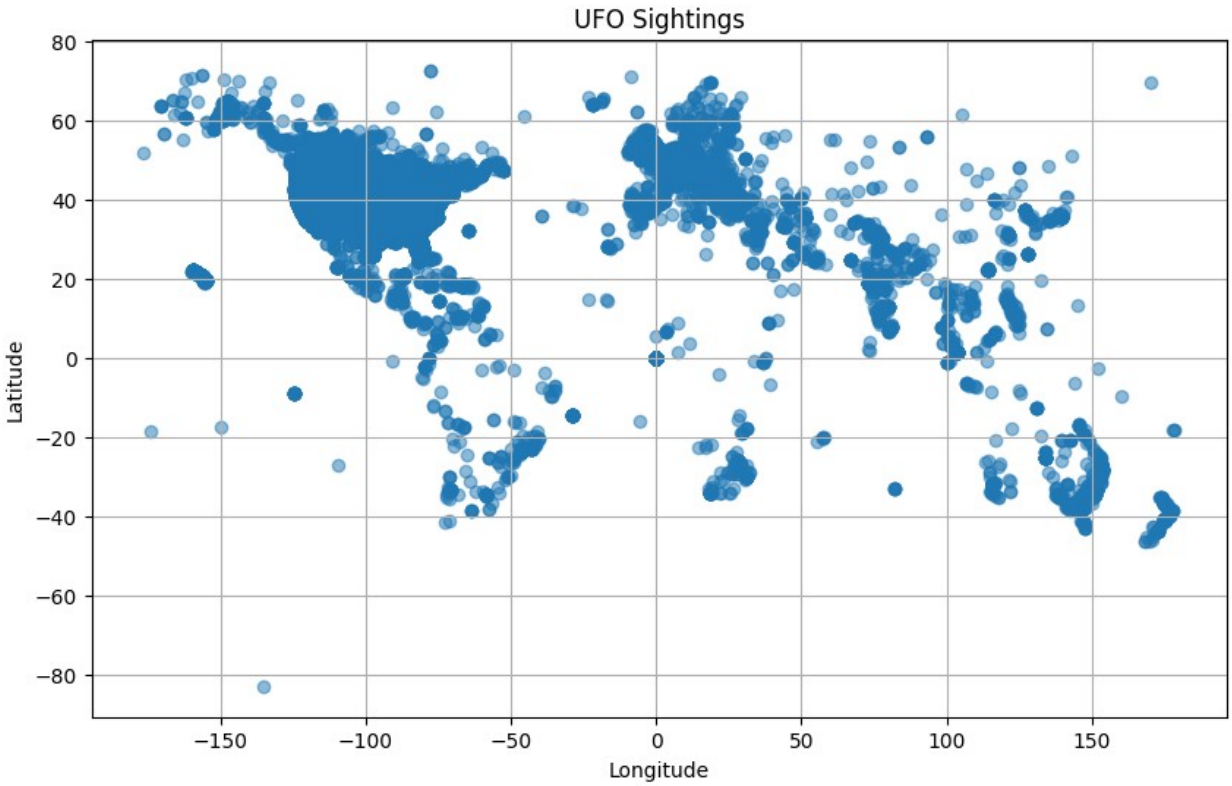
Explanation of Results

The correlation analysis of the dataset has provided some insightful results on the relationships between various variables. As noted, no meaningful correlation between the probability of seeing UFOs and the time they spend visible to the human eye. On the other hand, the country of origin, measured using latitude and longitude, exhibited correlations with the sightings . Specifically, larger numbers of sightings were registered in certain countries and regions, and the Western countries seemed to score highly on this variable. In fact, North America was a hotspot with the most sightings reported, as depicted in the plot. On the contrary, regions with higher altitude like the Tibetan Plateau seemed to have reported least sighting expression a possible correlation between altitude and UFO sightings. Duration of the UFO presence also seemed constant in the different locations studied. In conclusion, the data from 196 countries provided a valuable insights into the prevalence of UFO sightings. The frequent expression of UFO sightings in western and North American regions has several factors to cause it but the reasons for this is uncertain.

Visualisation

```
import csv
import matplotlib.pyplot as plt
```

```
# Initialize lists to store latitude and longitude data
latitude = []
longitude = []

# Open the CSV file
with open("cleaned_data.csv", "r", encoding="utf-8") as file:
    # Create a CSV reader object
    reader = csv.reader(file)

    # Skip the header row
    next(reader)

    # Iterate over each row in the CSV file
    for row in reader:
        # Extract latitude and longitude data
        try:
            lat = float(row[9])
            lon = float(row[10])

            # Append latitude and longitude to lists
            latitude.append(lat)
            longitude.append(lon)
        except ValueError:
            continue  # Skip rows with non-numeric latitude or
longitude

# Plot the latitude and longitude data
plt.figure(figsize=(10, 6))
plt.scatter(longitude, latitude, marker='o', alpha=0.5)
plt.xlabel('Longitude')
plt.ylabel('Latitude')
plt.title('UFO Sightings')
plt.grid(True)
plt.show()
```



Conclusion and presentation (10 marks)

Achievements

Overall, the conducted analysis of UFO sightings allowed learning some meaningful patterns and trends of these events. First, it was the determination of how often such sightings occur in some geographic regions, and certain locations like western territories, specifically North America, have higher rates, as revealed by correlation analysis. The meaningful associations of location aspects, such as latitude, longitude, with the frequency of sightings have been established. Second, the determination of average values of different variables became instrumental in recognizing what configurations such sightings tend to have.

Limitations

Apart from the invaluable insights, there are several limitations to the current analysis that the author should be aware of. The dataset could not cover all the possible UFO sightings since it mentioned only the reported ones, which are widely known for their biases and errors. Also the sightings were heavily USA centric Furthermore, as the dataset covered only sightings reported by a/to a single organization, it is possible to have UFOs' activity unregistered by it. Finally, the dataset may contain inaccuracies and missing data that may affect the current analysis. Also, as the collection includes reports from the 20th century, some older data points may have lost information or become distorted with time. Thus, I think one should use caution when interpreting their findings, taking into account any potential biases or limitations in the data.

Future Work

Future research may include diversification the sources of data. An even more detailed examination of the trends of UFO sightings can form part of further investigation. It might be interesting to consider the social and cultural elements that drive UFO reporting as well because they could possibly shed more light on the matter as why non western part of the world don't report UFO sightings as much. More research might look into the chronological and regional trends of UFO encounters in greater depth. ML engineers and data scientists can use the dataset in AI or machine learning engineering to strengthen the UFO research.

Video Presentation

I have submitted a video with my voiceover, providing a concise explanation of my project's design, key findings, successful aspects, and any challenges encountered.