Mini Project I: Where do I fly next?

1. **Introduction**

The effective ways for collecting and interpreting flight data are required given the internet travel industry's explosive growth. An efficient technique to get flight information from numerous sources can be possible through process of web scraping, which involves pulling information from websites. In this project, I am trying to figure out the best possible flight deals for the user according to their preference. The data processing methods used to extract flight information from sample flight websites are thoroughly analyzed. The efficient collection and analysis of flight data from internet sources is the issue I attempt to solve. Accurate and real-time flight information is essential for a variety of tasks, including market research, pricing comparison, and trend forecasting. The difficulties encountered and the scientific techniques used to scrape flight information from a well-known flight website are covered. To travel, I want to find the greatest flight bargain available given the price, destination, departure date, and arrival date.

Websites like Kayak.com, Momondo.com, Booking.com, and others provides a large number of data with enormous bargains which sometimes confuse the user in getting the best deal. I use the data from these websites and data processing techniques to address this problem and help the user to make decision based on their concerns like average price, flight duration or transits. Web scraping, which is the process of extracting data from a website using the HTML structure of the website or through the APIs utilized by the website, will be used to collect the data from these websites. To acquire the greatest deal for their desired route and travel dates, travelers can compare flight prices across several airlines and travel sites. Scraped data can be used by companies in the travel sector to study pricing trends over time and assist them modify their rates in response to consumer demand and competitive pricing. Airlines and travel companies can examine well-traveled routes, client preferences, and seasonal variations in demand to improve route design and marketing tactics. This allows us to compare all the information at once and find the best bargain.

2. **Data Collection**

I collected the flights data from three websites namely:

- Kayak.com
- Momondo.com
- Booking.com

First, I set the following input for the process.
Departure Place: Helsinki
Destination: New York
Departure Date: 26th October 2023

The data collected includes airlines name, departure time, arrival time, layover time, and transit location and so on. More than 350 data have been collected after scraping all the three above mentioned websites and all the data are then saved in a flight.csv file.

3. **Data Processing**

To scrape a website the chrome driver and selenium stealth were used along with other libraries like Beautiful Soup with the Python programming language to send HTTP requests and parse HTML output, respectively. Specific HTML tags and classes were found after an analysis of the website's HTML structure, allowing for focused data extraction. To extract the data saved in a csv file is used using panda's library.

The collected data has to be cleaned before it can be used. The three data frames prepared were concatenated in a csv file. Since we had to compare the data from the three websites, it was important that the data would be on the same scale. In this process, HTML material was parsed using Beautiful Soup and the Python computer language, together with the request module for web requests.

To deal with discrepancies, missing numbers, and undesirable characters, the extracted data was cleaned. For this, the panda's library and regular expressions were used. For instance, I checked the dataset for the missing values and checked the data types for the columns. I removed the null values for airlines names to visualize the airlines names properly. I filtered the data with column airlines name to count the number of airlines.

4. **Data Analysis**

After data processing steps, now the extracted data are used for a comprehensive EDA by using appropriate visualization methods like count plot, bar diagram, and so on. These plots were used to visualize the collected data to show diverse characteristics, including pricing trends over time, price distribution across different airlines, and flight frequency on routes.
Let us discuss a few visualization methods done in this project. Their process and their result analysis.

1) Count Plot: Shows the number of flights for each airline.

Code for ploting Airlines Name

```
plt.figure(figsize=(10, 8))
sns.countplot(y=filtered_data['AirlinesName'],
order=filtered_data['Airlines Name'].value_counts().index)
plt.title('Number of Flights per Airline (More than 350 data
points)')
plt.xlabel('Count')
```

```
plt.ylabel('Airlines Name')
plt.show()
```

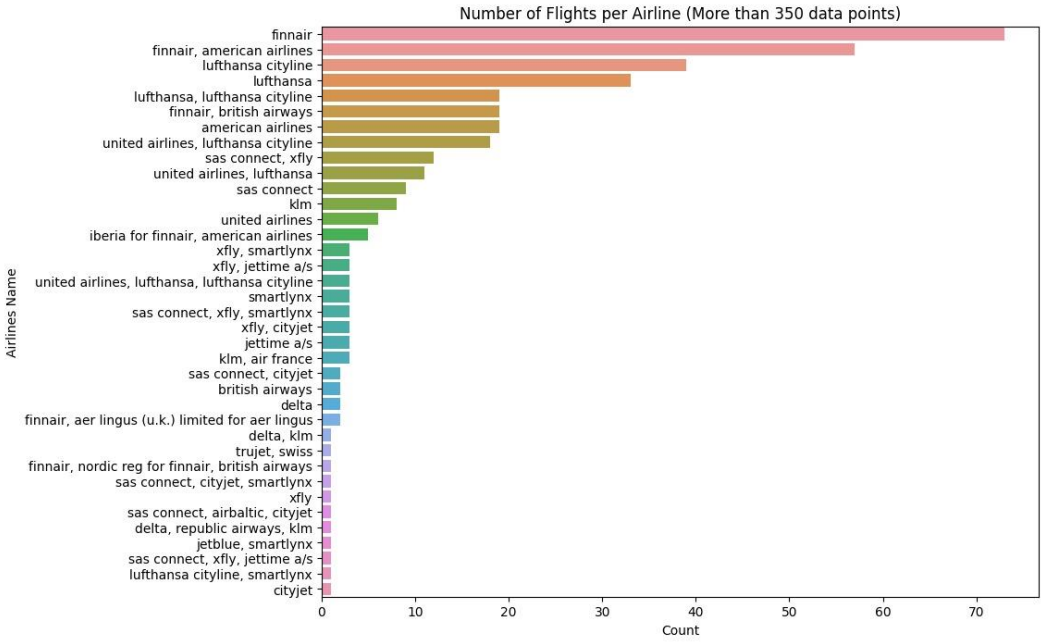

Figure 1: Number of flights per Airline

In the above figure, I use count plot visualization methods. The figure shows that Finnair has the highest number of flights followed by American airlines and Luthansa. On the other hand, airlines like cityjet, jetblue, and smartLynx have least number of flights.

2) Calculate average price for each airline

Here the average price is visualized based on airlines name using box plot.

```
# Calculate average price for each airline
plt.figure(figsize=(35, 8))
sns.barplot(x='AirlinesName',y='Price',data=df, palette='viridis')
plt.title('Average Price by Airline')
plt.xlabel('Airlines Name')
plt.ylabel('Average Price')
plt.xticks(rotation=35)
plt.show()
```

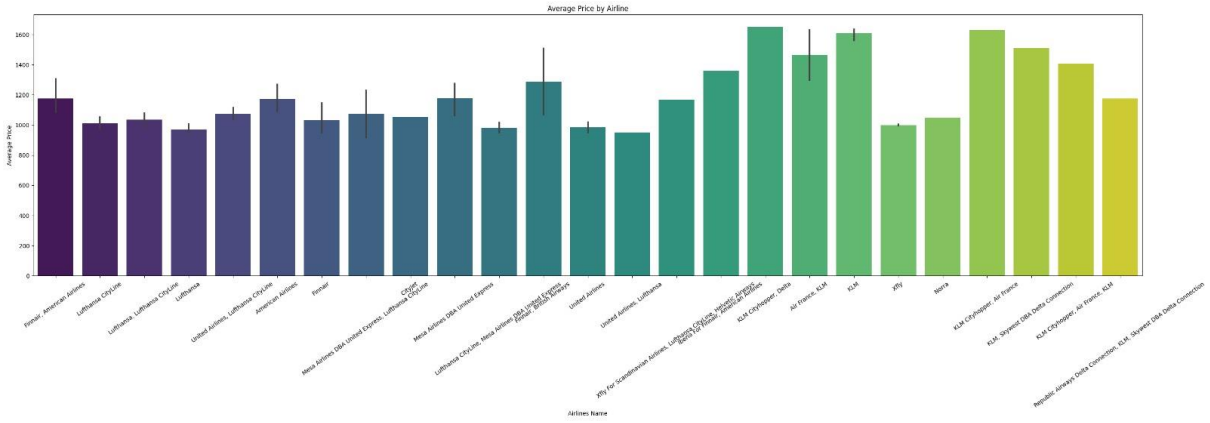

Figure 2: Average Price per Airlines

This plot demonstrate the average price for the flight ticket for airlines under observation. Here, United Airlines is found to be the cheapest option available. Whereas, KLM cityhopper, Delta, and Air France are found to be among the expensive ones.

3) Number of transits stop permitted to each airline

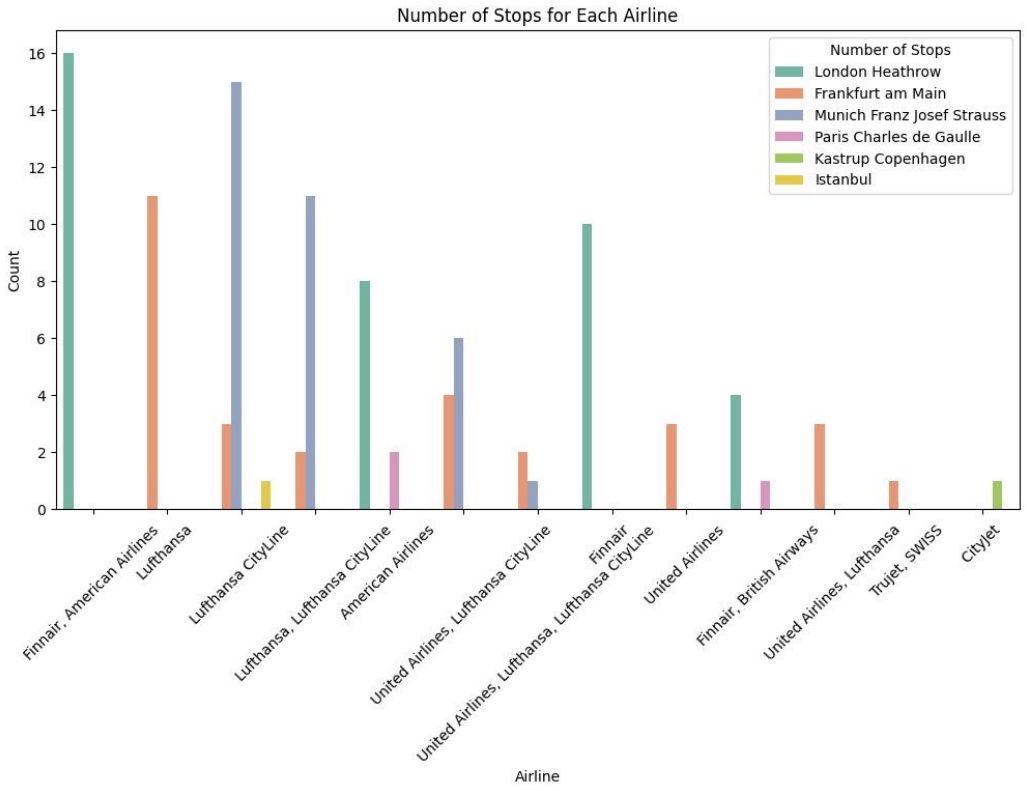

Figure 3: Number of transits stops for each airline

Here, I am trying to show the numbers of transit for each airline. This plot helps you to know the permitted transit location for each airline.

Normally, not all transit locations are given a permit for every airline. So, this visualization helps us to understand the figure of each airline and their permit transit location. Where highest number of permits in this route is given to Finnair and American airlines followed by Lufthansa Airlines.

```
# Create a countplot for the number of stops for each airline
      plt.figure(figsize=(12, 6))
      sns.countplot(x='Airlines Name', hue='Transit Airport', data=df,
      palette='Set2')
      # 'Set2' is a color palette, you can choose another one if you
      prefer
      plt.title('Number of Stops for Each Airline')
      plt.xlabel('Airline')
      plt.ylabel('Count')
      plt.legend(title='Number of Stops', loc='upper right')
      plt.xticks(rotation=45)
      plt.show()
```

5. **Conclusion**

In conclusion, web scraping a flight website has shown to be a fruitful endeavor, offering numerous insights and data-driven advantages for various stakeholders. I collected data and conducted in-depth analysis to learn vital information about flight prices, routes, airlines, and customer price preferences. The dynamic nature of this website made scraping it difficult for me since data on dynamic webpages is always changing. The dynamic websites listed above depend on user input to produce results. The utilization of several data types in numerical data was also the most challenging aspect. This problem was resolved by data processing processes that converted the data types into the format I needed.

The user was requested to provide their selections based on pricing, length of layover, and airlines when I tried to create a user interaction. The user preferences were printed with the outcome as per the preference. The enormous dynamic site caused the programming to run into numerous faults. To correct those issues, such as data types, null values, and others, I employ every parameter that is available. I am learning more about the difficulties dynamic websites face thanks to this initiative. I learned about various problems one could encounter when web scraping and how to fix them. The exploratory data analysis compares different visualization techniques on actual data from the real world.