New York City Yellow Taxi Data

Objective

In this case study you will be learning exploratory data analysis (EDA) with the help of a dataset on yellow taxi rides in New York City. This will enable you to understand why EDA is an important step in the process of data science and machine learning.

Problem Statement

As an analyst at an upcoming taxi operation in NYC, you are tasked to use the 2023 taxi trip data to uncover insights that could help optimise taxi operations. The goal is to analyse patterns in the data that can inform strategic decisions to improve service efficiency, maximise revenue, and enhance passenger experience.

Tasks

You need to perform the following steps for successfully completing this assignment:

1. Data Loading
2. Data Cleaning
3. Exploratory Analysis: Bivariate and Multivariate
4. Creating Visualisations to Support the Analysis
5. Deriving Insights and Stating Conclusions

↳ 3 cells hidden

Data Understanding

The yellow taxi trip records include fields capturing pick-up and drop-off dates/times, pick-up and drop-off locations, trip distances, itemized fares, rate types, payment types, and driver-reported passenger counts.

The data is stored in Parquet format (*.parquet*). The dataset is from 2009 to 2024. However, for this assignment, we will only be using the data from 2023.

The data for each month is present in a different parquet file. You will get twelve files for each of the months in 2023.

The data was collected and provided to the NYC Taxi and Limousine Commission (TLC) by technology providers like vendors and taxi hailing apps.

You can find the link to the TLC trip records page here: https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page

↳ 4 cells hidden

1 Data Preparation

[5 marks]

Import Libraries

```
# Import warnings



# Import the libraries you will be using for analysis



# Recommended versions
# numpy version: 1.26.4
# pandas version: 2.2.2
# matplotlib version: 3.10.0
# seaborn version: 0.13.2

# Check versions
import warnings
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
# Recommended versions
# numpy version: 1.26.4
# pandas version: 2.2.2
# matplotlib version: 3.10.0
# seaborn version: 0.13.2

# Check versions
print("numpy version:", np.__version__)
print("pandas version:", pd.__version__)
```

```
print("matplotlib version:", plt.matplotlib.__version__)
print("seaborn version:", sns.__version__)
```

```
numpy version: 1.26.4
pandas version: 2.2.2
matplotlib version: 3.10.0
seaborn version: 0.13.2
```

1.1 Load the dataset

[5 marks]

[ ] ↳ 18 cells hidden

2 Data Cleaning

[30 marks]

[ ] ↳ 44 cells hidden

3 Exploratory Data Analysis

[90 marks]

[ ] ↳ 3 cells hidden

Here's how I would categorize these variables:

Categorical Variables

- `VendorID` : Categorical (identifies the provider/company)
- `RatecodeID` : Categorical (represents different rate types like standard, JFK, etc.)
- `PULocationID` : Categorical (taxi zone codes for pickup location)
- `DOLocationID` : Categorical (taxi zone codes for dropoff location)
- `payment_type` : Categorical (represents payment methods like credit card, cash, etc.)

Numerical Variables

- `passenger_count` : Numerical (discrete count)
- `trip_distance` : Numerical (continuous)

- `pickup_hour` : Numerical (can be treated as either numerical or categorical depending on analysis)
- `trip_duration` : Numerical (continuous)

Datetime Variables (special type)

- `tpep_pickup_datetime` : Datetime
- `tpep_dropoff_datetime` : Datetime

Monetary Variables

All the monetary parameters are numerical (continuous):

- `fare_amount` : Numerical
- `extra` : Numerical
- `mta_tax` : Numerical
- `tip_amount` : Numerical
- `tolls_amount` : Numerical
- `improvement_surcharge` : Numerical
- `total_amount` : Numerical
- `congestion_surcharge` : Numerical
- `airport_fee` : Numerical

Temporal Analysis

[ ] ↳ 4 cells hidden

Financial Analysis

[ ] ↳ 21 cells hidden

Geographical Analysis

[ ] ↳ 21 cells hidden

3.2 Detailed EDA: Insights and Strategies

[50 marks]

[ ] ↳ 45 cells hidden

4 Conclusion

[15 marks]

[ ] ↳ 7 cells hidden