Jalari Mallikarjuna

Uber Analysis

The Uber Analysis project examines ride data to uncover trends and optimize service efficiency

project_link

Agenda

1. Introduction
2. Key Concepts
3. Data Collection
4. Data Preprocessing
5. EDA
6. Vizualizations

Agenda

7 Insights and Findings

8 Conclusion

9 End

Introduction

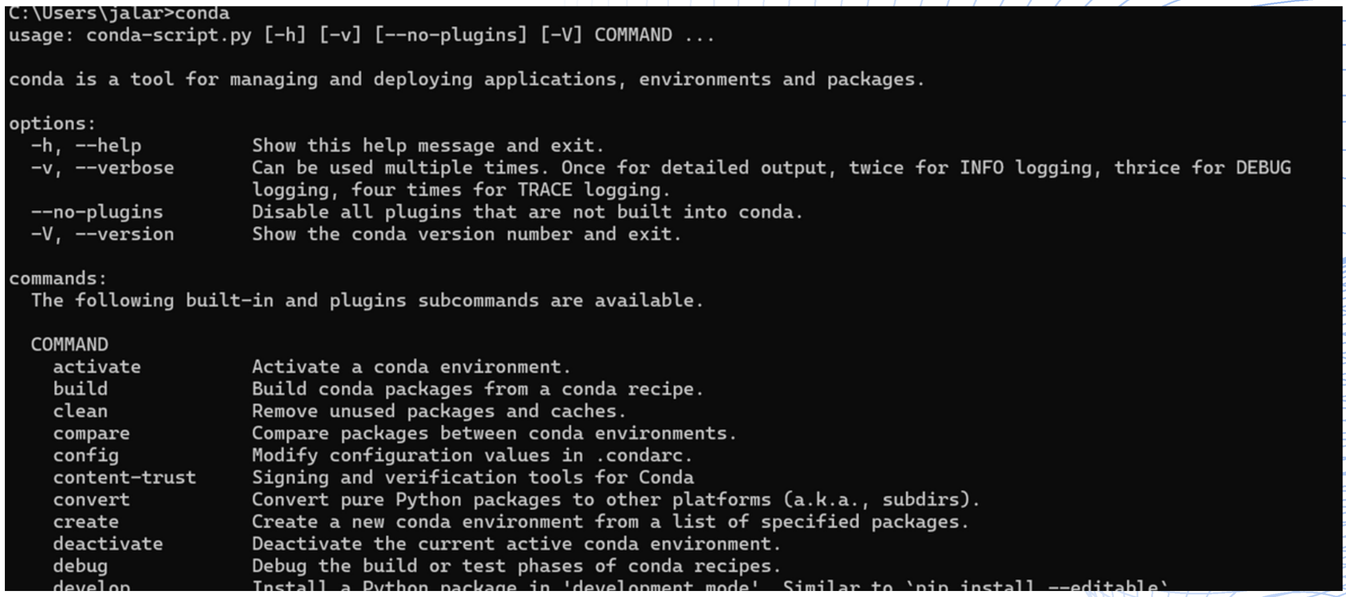
- Objective: Analyze Uber trip data to uncover trends in ride demand, peak hours, and popular locations.
- Focus Areas: Identify trip patterns, seasonal variations, and operational efficiency opportunities.
- Key Insights: Optimize driver allocation, improve service availability, and enhance user experience.
- Data Utilized: Includes trip timings, pickup/dropoff locations, trip durations, and ride status.
- Outcome: Provide actionable recommendations to guide Uber's strategic decisions and business improvements.
- Tools: Data analysis and visualization techniques to communicate findings effectively.

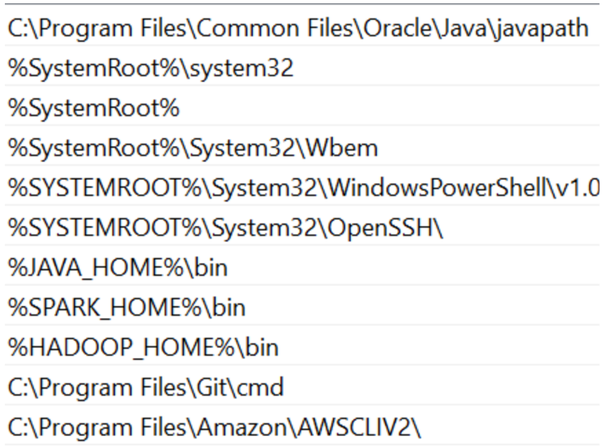
Project Needs

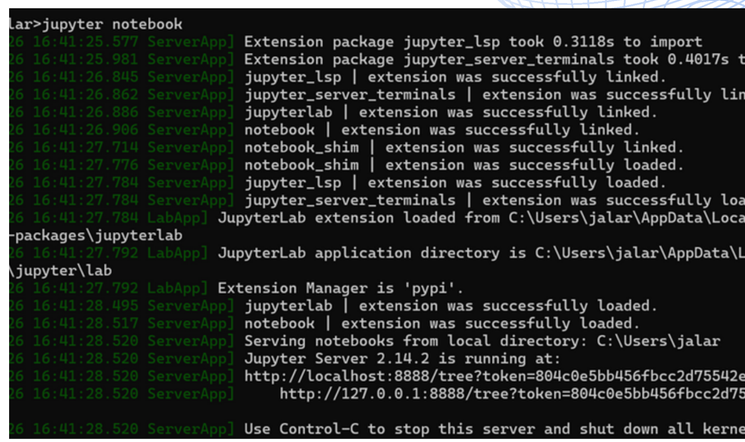
- Jupyter Notebook for interactive data analysis and visualization.
- Python as the primary language for data processing and algorithm implementation.
- Anaconda for managing packages, dependencies, and virtual environments.
- VSCode for coding and debugging with an efficient editor.
- GitHub for version control and project collaboration.

Environmental Variables

- **Package Installation**: Installing required libraries and dependencies for smooth project execution.
- **Environment Variables**: Configuring variables to ensure compatibility with system settings.
- **Compatibility Check**: Ensuring all tools, libraries, and dependencies are compatible with the project setup.
- **Efficient Execution**: Setting up the environment to facilitate seamless data analysis and algorithm execution.





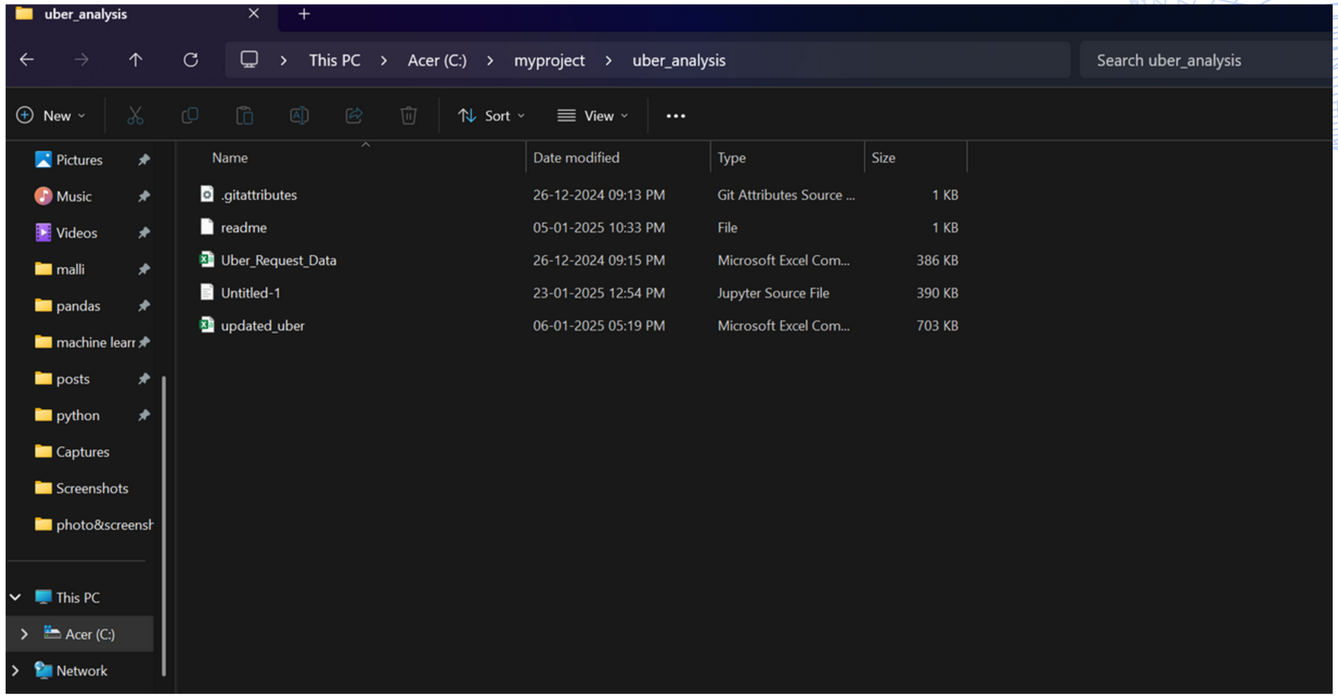


Project Setup

Create Environment

C:\myproject\uber_analysis

- **Local Storage**: Jupyter Notebook files are stored on the local desktop for easy access and management.
- **Organized Directory**: A dedicated directory is created to structure and organize notebooks systematically.
- **Global Python Script**: A global script (environment.py) is used to execute the analysis, containing essential imports, configurations, and reusable functions.
- **Efficiency**: This setup ensures that all project files are easily accessible and that Python code execution is streamlined.



Project Setup

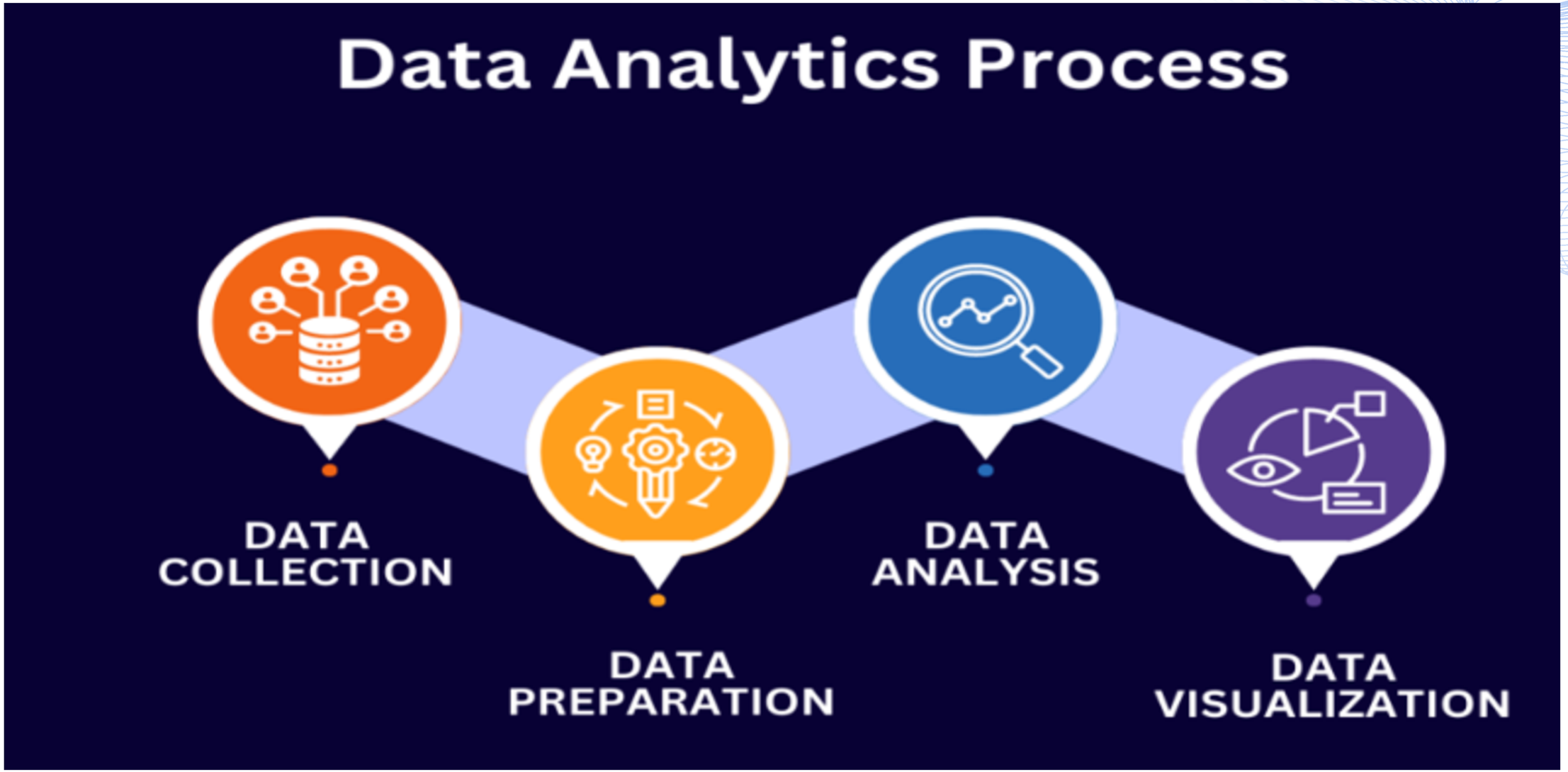
Clone GitHub Repository to VSCode

1.Create a GitHub repository.
2.Clone the repository to VSCode.
3.Set up Git for version control.

Project Requirements and Libraries

1.Required libraries include Matplotlib, Pandas, and NumPy etc .
2.Install libraries using pip or conda.
3.Verify library versions.

Data Analytics Process



Data Collection

- **Libraries Imported**: Used Pandas for data manipulation, NumPy for numerical operations, and Matplotlib for data visualization.
- **Data Source**: Dataset provided by Instrovate Technologies, containing detailed Uber ride request information.
- **Key Attributes**: Includes request ID, pickup/dropoff points, driver IDs, status, and timestamps.
- **Data Processing**: Utilized Pandas for loading, cleaning, and analyzing the data to generate actionable insights.

Data collection

```
import matplotlib.pyplot as plt
import seaborn as s
import warnings
warnings.filterwarnings("ignore")
```

[93] Python

```
uber=pd.read_csv("Uber_Request_Data.csv")
```

[94] Python

```
uber.head()
```

[95] Python

...

	Request id	Pickup point	Driver id	Status	Request timestamp	Drop timestamp
0	619	Airport	1.0	Trip Completed	11/7/2016 11:51	11/7/2016 13:00
1	867	Airport	1.0	Trip Completed	11/7/2016 17:57	11/7/2016 18:47
2	1807	City	1.0	Trip Completed	12/7/2016 9:17	12/7/2016 9:58
3	2532	Airport	1.0	Trip Completed	12/7/2016 21:08	12/7/2016 22:03
4	3112	City	1.0	Trip Completed	13-07-2016 08:33:16	13-07-2016 09:25:47

Data Preprocessing

- **Data Loading**: Imported the dataset into a Pandas DataFrame using pd.read_csv() for easy manipulation.
- **Handling Missing Values**: Dropped rows with missing data using dropna() to maintain data integrity.
- **Datetime Conversion:** Converted request_timestamp and drop_timestamp columns to datetime format using pd.to_datetime() for easier time-based analysis.
- Extracted Hour of the Day to analyze hourly trends.
- Derived Day of the Week to examine weekday vs. weekend patterns.
- Created a Month feature to observe seasonal trends.
- **Data Type Conversion**: Optimized data by converting columns like request_id and driver_id to string types.

Exploratory Data Analysis

Summary Statistics:

- Computed basic statistics to overview numeric features and identify outliers.

Request Status Distribution:

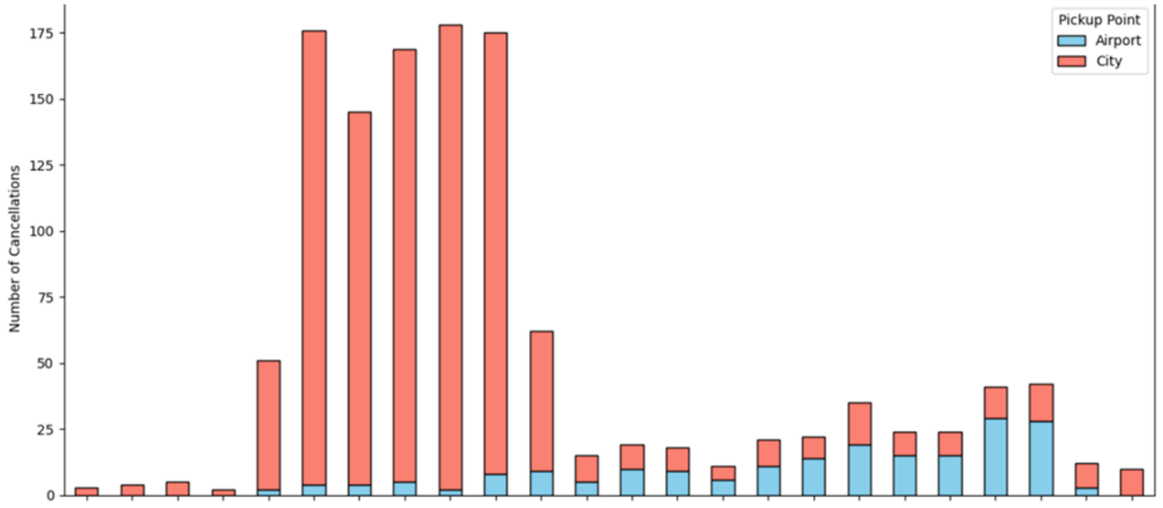
- Analyzed the status of ride requests (completed, canceled, no cars available) and visualized with a bar plot.

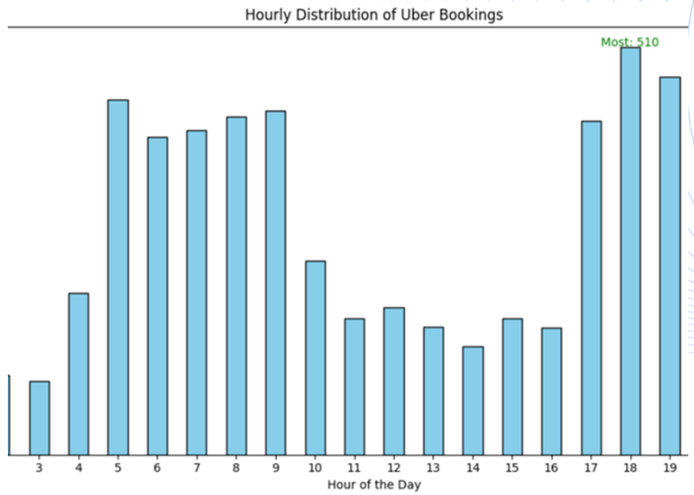
Request Timing Analysis:

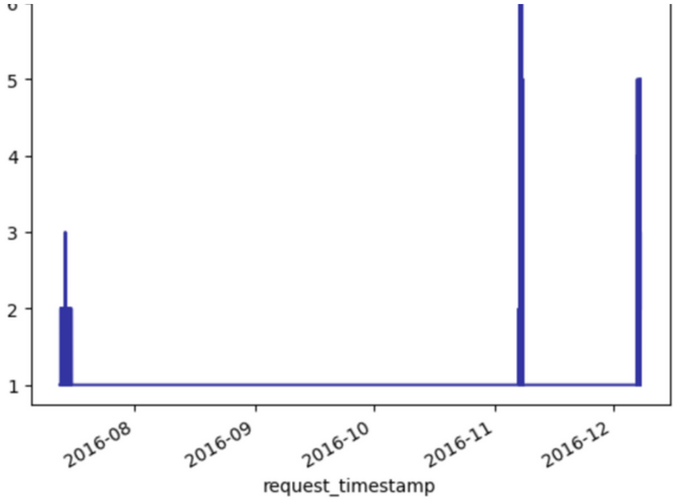
- Studied distribution of requests by time of day, weekdays, and months using histograms.

Driver Activity Analysis:

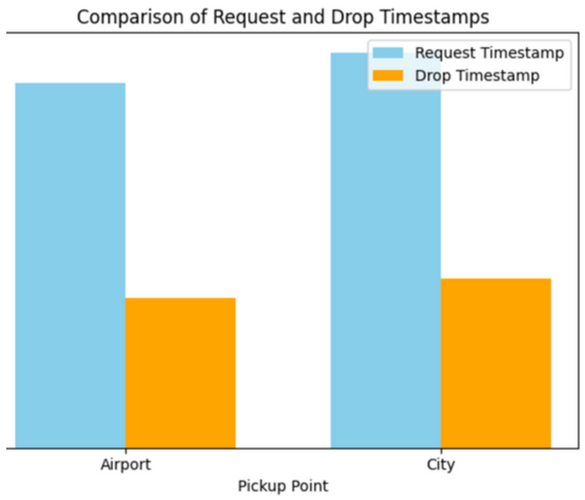
- Analyzed the frequency of requests per driver to identify any imbalances in driver workload.

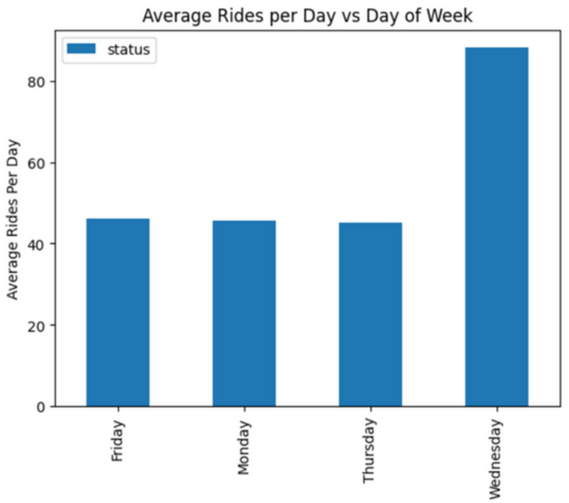


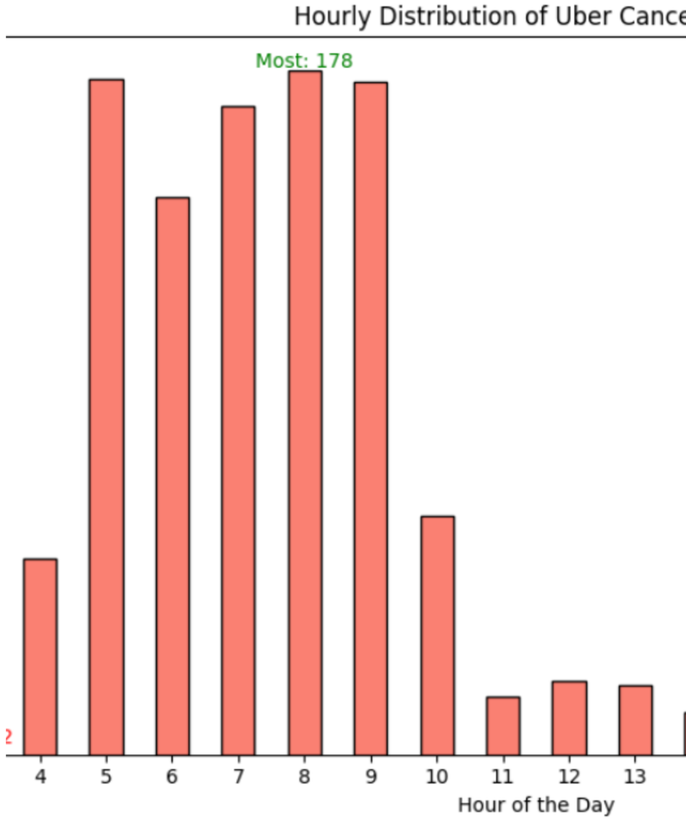




Visualization







Insights and Findings

01

Hourly Trends

Ride requests peaked during morning rush hours (7 AM–9 AM) and evening rush hours (5 PM–8 PM), while late-night demand (12 AM–5 AM) was significantly lower.

02

Request Status

A significant percentage of requests were canceled or marked as "No Cars Available," particularly during high-demand periods

03

Geospatial Trends

High-demand locations included city centers and airports, with dense clusters of pickups and drop-offs in these areas.

Conclusion

This project uncovered critical patterns in Uber's ride request data, including peak demand during morning and evening rush hours and higher daytime demand compared to nighttime. Wednesdays recorded the highest number of successful rides. Geospatial analysis identified city centers and airports as high-demand zones, vital for operational planning. Additionally, an imbalance in driver workload was observed, highlighting opportunities for optimizing resource allocation.


Thank you!