Uber Trips Analysis

Vaishnavi Kataram, *Pace University*

Abstract— Data analytics has been instrumental in helping companies optimize and enhance their performance for decades. The use of data analytics and visualization offers numerous advantages, including the identification of emerging trends, the exploration of relationships and patterns within data, and the extraction of valuable insights from these patterns. In light of the current demands, a thorough examination of these concepts becomes essential for maximizing the benefits they provide. This project focuses on comprehending a specific dataset from Uber in New York City, serving as a crucial resource to understand the application of data analytics and visualization. The dataset is analysed using the 'Python' programming language, leveraging libraries such as pandas, matplotlib and seaborn. Projects of this nature facilitate the acquisition of knowledge about intricate operations in data visualization, enabling the recognition of patterns within the extensive dataset of this major organization and offering critical insights into previously undiscovered information.

Keywords— Uber, Data analytics, Data visualization, Python programming, pandas, matplotlib, seaborn.

1. INTRODUCTION

Uber has positioned itself as a leading company in offering modern transportation solutions. Essentially, Uber operates as a networking platform, and all its evolving activities can be simplified as facilitating the connection between supply and demand. The field of analytics is experiencing significant growth, and businesses are increasingly applying it to enhance their performance. This project is primarily focused on data visualization, specifically using the matplotlib and seaborn libraries. The aim is to expand our understanding of utilizing matplotlib and seaborn for interpreting data and developing insights into the characteristics of customers who use Uber for their trips.

The resolution to this matter involves grasping the concept of Customer Segmentation, also known as Market Segmentation. Visualize it as a game where a child categorizes balls or cubes based on their shapes or colors. In simpler terms, customer segmentation involves dividing customers and markets based on various criteria and characteristics. Uber utilizes historical data from the past 3 or 4 weeks to pinpoint areas in the city experiencing exceptionally high demand.

In the current competitive business landscape, leveraging data analysis is imperative for growth. Businesses need to generate data analysis reports and various analytical documents to serve as references for specific activities and endeavors, especially when making decisions for future company operations. By organizing information based on the intended activities, a structured data analysis can be developed. In the Uber data analysis Python project, we explored the creation of data visualizations. Notably, Uber stands out as the sole mobility company that assesses and publishes real-world sustainability data.

In this Python project, our aim is to analyze the dataset on Uber Pickups in New York City. This project primarily focuses on data visualization and aims to provide guidance on utilizing the matplotlib and seaborn libraries for interpreting data and gaining insights into the preferences of customers who use Uber for their trips. The key objectives include:

- Visualizing the growth of Uber in NYC
- Identifying patterns in the time series to characterize demand
- Estimating the value of the NYC market for Uber
- Uncovering additional insights regarding the service usage
- Making an effort to predict the growth in demand.

2. RELATED WORK

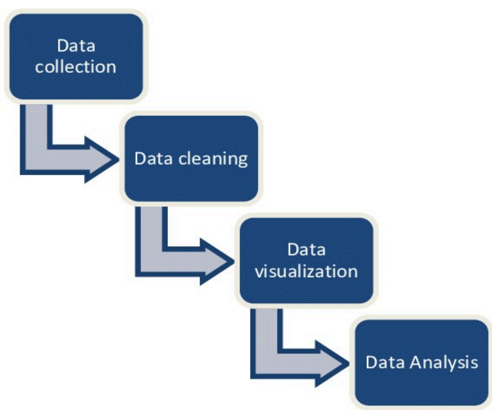
Two papers on relevant studies are pertinent to the same project—those focusing on case studies of Uber data analytics and visualization.

The research paper "Uber Related Data Analysis using Machine Learning" by Rishi Srinivas, B. Ankayarkanni, and R. Sathya Bama Krishna, published by IEEE, emphasizes on predicting ride prices before the journey starts, incorporating factors such as cab type, source, destination, total ride time, and cost. The paper's exploration of using visualization techniques, such as graphs and charts, to represent the relationship between different factors and ride costs has guided our approach to presenting insights. Visual aids not only help identify patterns and trends in the dataset but also enhance stakeholders' understanding of the intricate interactions influencing ride costs. Ultimately, the emphasis on visual insights in data analysis has been integral to our project, enabling clearer communication of the accuracy of our analysis model and highlighting areas for potential improvement, thereby facilitating better decision-making.

The paper "Data Analysis of Uber and Lyft Cab Services" by Shashank H., as published by IJIIRD, provides a robust and data-driven method of analyzing Uber trips. The emphasis on understanding demand trends and optimizing operational procedures aligns with our project's goal of gaining insights from Uber's vast datasets. The suggested combination of machine learning algorithms with logistic and linear regression provides a strong framework for cost prediction based on cab type, source, destination. The focus on analysis through visualization approaches has guided our efforts to represent intricate data relationships visually, enhancing our understanding and improving the overall results of our data analytics. This paper has served as a valuable guide, offering practical approaches and insights to enhance decision-making within the ride-sharing industry through effective data analytics.

3. METHOD AND MATERIAL

3.1 FLOWCHART



The diagram above represents the method flow of this project.

3.2 DATASET

The dataset under consideration comprises information related to Uber pickups in New York City, specifically for the month of September 2014, sourced from Kaggle. It encompasses a substantial amount of data, featuring over 1,000,000 pickups represented as rows.

	Date/Time	Lat	Lon	Base
0	2014-09-01 00:01:00	40.2201	-74.0021	B02512
1	2014-09-01 00:01:00	40.7500	-74.0027	B02512
2	2014-09-01 00:03:00	40.7559	-73.9864	B02512
3	2014-09-01 00:06:00	40.7450	-73.9889	B02512
4	2014-09-01 00:11:00	40.8145	-73.9444	B02512

The dataset is organized into four key columns, each providing distinct details:

1. Date/Time: The date and time of each Uber pickup. It serves as a temporal reference point for analyzing patterns and trends in the data.
2. Lat: The latitude coordinates corresponding to the location of each Uber pickup. Latitude values indicate the north-south position on the Earth's surface.
3. Long: The longitude coordinates associated with the pickup locations. Longitude values denote the east-west position on the Earth's surface.
4. Base: The base refers to the base or station identifier for each pickup. This information could be crucial for understanding the operational aspects of Uber's services.

3.3 PYTHON

Python is a versatile and widely-used programming language. Python harnesses the synergy of Pandas, Matplotlib, and Seaborn for efficient data analysis and visualization. Pandas offers robust data manipulation, while Matplotlib and Seaborn provide dynamic plotting tools, enhancing the presentation of data insights. Python's seamless integration of these libraries underscores its strength as a data-centric programming language, empowering users to explore and communicate complex datasets with ease. The trio's collaborative functionality solidifies Python's position as a preferred platform for data scientists and analysts seeking a comprehensive and streamlined approach to data tasks.

3.4 LIBRARIES USED

Pandas - a powerful data manipulation library in our project, efficiently handles and explores the extensive Uber trip dataset, ensuring seamless data analysis.
Matplotlib - a versatile plotting library, enables the creation of insightful visualizations, enhancing our ability to communicate patterns and trends in the data.
Seaborn - built on Matplotlib, further enhances our project by providing a high-level interface for aesthetically pleasing statistical graphics, elevating the overall quality of our visual presentations.

3.5 JUPYTER NOTEBOOK

Jupyter Notebook serves as a dynamic platform for our project, providing an interactive and collaborative environment for data analysis and visualization using Python. Its integrated support for Pandas, Matplotlib, and Seaborn makes it an ideal choice for exploring and presenting insights from the Uber trip dataset. Jupyter's combination of code cells and narrative text facilitates a seamless workflow, allowing us to document and share our analysis in a cohesive manner. The interactive nature of Jupyter Notebook enhances our ability to iteratively refine and visualize results, making it an invaluable tool for the dynamic requirements of our project.

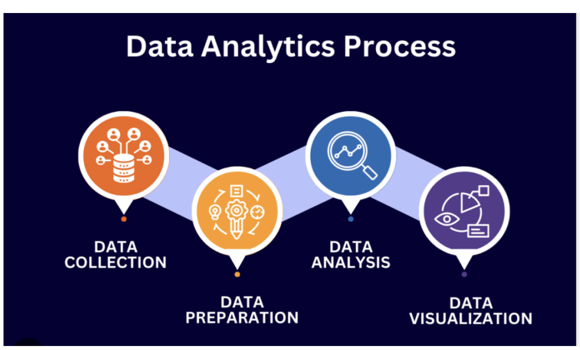
3.6 ALGORITHM

1. Importing Essential Packages
2. Loading the data into their respective variables
3. Processing and Transforming the data as needed
4. Visualizing the distribution of trips by the hour of the day
5. Illustrating data patterns through daily trip plots of the month
6. Analyzing patterns in weekday distribution of trips
7. Constructing a Heatmap representation for hour and weekday
8. Creating a scatter plot to analyze Uber trips based on geographical coordinates (Latitude and Longitude)
9. Extracting insights from all the generated visualizations

4. PROPOSED SYSTEM

We propose the development of a data visualization project using Matplotlib, Seaborn, and other Python libraries to analyze various parameters such as Trips by the hours in a day, Trips during weekdays and days in a month and Trips per base. The goal is to create visualizations for different timeframes and elucidate how time influences customer trips.

- Customers often express dissatisfaction with traditional cab companies due to high prices and long wait times, presenting an opportunity to tap into new and expansive markets.
- Identifying days with a higher number of active vehicles for each base can enhance operational efficiency.
- Exploring growing markets in suburban areas without taxi services can expand Uber's reach.
- Reducing Estimated Time of Arrival by increasing the number of Uber drivers can improve customer satisfaction, leading to increased revenue for the company and higher profits for drivers.
- Leveraging the data, we can identify the most frequented destinations, generating substantial airline revenues based on the booked trip count.



Data Collection:
Gather Uber trip data in New York City for September 2014, including date/time, latitude, longitude, and base information.

Data Preparation:
Clean and organize the dataset, addressing any missing or inconsistent values. Prepare the data for analysis by loading it into designated variables.

Data Analysis:
Analyze the dataset to unveil patterns such as hourly and weekly trip distributions, identify popular travel destinations, and assess the correlation between Uber activity and specific timeframes.

Data Visualization:
Utilize Python libraries, including Matplotlib and Seaborn, to create visualizations. Generate plots illustrating hourly and weekly trip patterns, create scatter plots for geographical analysis, and develop insightful visual representations to understand customer behavior and operational trends.

Insights and Reporting:
Collect insights derived from the visualizations, addressing aspects like customer preferences, peak hours, and potential growth markets. Present findings in a comprehensive report, enhancing understanding and informing strategic decisions for Uber's operations in New York City.

5. CODE EXPLANATION

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

def DataLoader(csv_file):
    try:
        data = pd.read_csv(csv_file)
        return data
    except FileNotFoundError:
        print(f"The file '{csv_file}' does not exist. Please provide a valid CSV file.")
    except Exception as e:
        print(f"Exception occurred while reading the file: {e}")

data = DataLoader("uber-dataset.csv")

if data is not None:
    data["Date/Time"] = data["Date/Time"].map(pd.to_datetime)
    print(data.head())
```

"DataLoader" function loads data from a CSV file using the Pandas library. The function takes a CSV file path as an argument, attempts to read the file, and returns the loaded data. If the file is not found, it prints an error message.

The code then calls this function to load Uber trip data from a file named "uber-dataset.csv." If the data is successfully loaded, it converts the "Date/Time" column to datetime format using Pandas' to_datetime function and prints the first few rows of the loaded data using the head() method.

```python
class DataProcessor:

    def __init__(self, data):
        self.data = data

    def add_day_column(self):
        self.data["Day"] = self.data["Date/Time"].apply(lambda x: x.day)

    def add_weekday_column(self):
        self.data["Weekday"] = self.data["Date/Time"].apply(lambda x: x.weekday())

    def add_hour_column(self):
        self.data["Hour"] = self.data["Date/Time"].apply(lambda x: x.hour)

    def process_data(self):
        self.add_day_column()
        self.add_weekday_column()
        self.add_hour_column()

try:
    if data is None:
        raise TypeError
    data_processor = DataProcessor(data)
    data_processor.process_data()
    print(data_processor.data.head())
except TypeError:
    print("Exception occured, data is None")
except Exception as e:
    print("Exception occured: ",e)
```

"DataProcessor" class has methods to process date and time information in the provided dataset. The class takes the data as an input during initialization. Three methods add_day_column, add_weekday_column, and add_hour_column are implemented to add new columns for day, weekday, and hour, respectively, based on the "Date/Time" column.

An instance of the DataProcessor class is created with the Uber trip data, and the process_data method is called to execute the data processing steps. The code then prints the first few rows of the processed data.

The try-except block ensures that if the provided data is None, a TypeError is raised, and an appropriate error message is displayed. Additionally, it catches and prints any other exceptions that might occur during the execution of the code.

```
class DataVisualizer:

    def __init__(self, data):
        self.data = data

    def set_seaborn_plot_size(self, figsize=(12, 10)):
        sns.set(rc={'figure.figsize': figsize})

    def plot_day_distribution(self):
        try:
            sns.histplot(self.data["Day"], bins=31, kde=True, color='blue')
            plt.title("Trips per Day")
            plt.xlabel("Day")
            plt.ylabel("Density")
            plt.xticks(range(31))
            plt.show()
        except Exception as e:
            print("Exception occurred while plotting day distribution: ",e)

    def plot_hour_distribution(self):
        try:
            sns.histplot(self.data["Hour"], bins=24, kde=True, color='red')
            plt.title("Trips per Hour")
            plt.xlabel("Hour")
            plt.ylabel("Density")
            plt.xticks(range(24))
            plt.show()
        except Exception as e:
            print("Exception occurred while plotting hour distribution: ",e)
```

```
    def plot_weekday_distribution(self):
        try:
            sns.histplot(self.data["Weekday"], bins=7, kde=True, color='green')
            plt.title("Trips per Week")
            plt.xlabel("Weekday")
            plt.ylabel("Density")
            plt.xticks(range(7), ['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun'])
            plt.show()
        except Exception as e:
            print("Exception occurred while plotting weekday distribution: ",e)

    def plot_trips_per_base(self):
        try:
            base_counts = self.data['Base'].value_counts()
            base_counts.plot(kind='bar', color='skyblue')
            plt.title("Trips per Base")
            plt.xlabel("Base")
            plt.ylabel("Number of Trips")
            plt.show()
        except Exception as e:
            print("Exception occurred while plotting trips per base: ",e)

try:
    if data is None:
        raise TypeError
    data_visualizer = DataVisualizer(data)
    data_visualizer.set_seaborn_plot_size()
except TypeError:
    print("Exception occured, data is None")
except Exception as e:
    print("Exception occured: ",e)
```

"DataVisualizer" class has methods to visualize the distribution of Uber trips based on different parameters using Seaborn and Matplotlib. The class takes the Uber trip data as input during initialization.

Methods in the class include:
set_seaborn_plot_size: Sets the size of Seaborn plots.
plot_day_distribution: Plots the distribution of trips per day.
plot_hour_distribution: Plots the distribution of trips per hour.
plot_weekday_distribution: Plots the distribution of trips per week.
plot_trips_per_base: Plots the number of trips per Uber base.

An instance of the DataVisualizer class is created with the Uber trip data, and the size of Seaborn plots is set. The try-except block ensures that if the provided data is None, a TypeError is raised, and an appropriate error message is displayed. Additionally, it catches and prints any other exceptions that might occur during the execution of the code. The class methods are then called to generate and display various visualizations of the Uber trip data.

```
def correlation_heatmap():
    df = data.groupby(["Weekday", "Hour"]).apply(lambda x: len(x))
    df = df.unstack()
    sns.heatmap(df, annot=False)
    plt.title('Correlation of Weekday and Hour')
    plt.show()

correlation_heatmap()
```

correlation_heatmap() function generates a heatmap to visualize the correlation between the 'Weekday' and 'Hour' columns in a DataFrame.

A heatmap in python's seaborn library is a graphical representation of a data matrix where values are represented as colors. It uses a color scale to visualize the intensity of values across two dimensions (rows and columns), making it easy to identify patterns, correlations, and variations in the data.

```
def scatter_plot():
    fig, ax = plt.subplots(figsize=(12, 8))
    sc = ax.scatter(
        x=data['Lon'],
        y=data['Lat'],
        alpha=0.4,
        s=data['Day'],
        c='indigo',
        label='Uber Trips'
    )
    fig.colorbar(sc, ax=ax)
    ax.set_title("Uber Trips Analysis")
    ax.set_xlabel('Longitude')
    ax.set_ylabel('Latitude')
    ax.legend()
    plt.show()

scatter_plot()
```

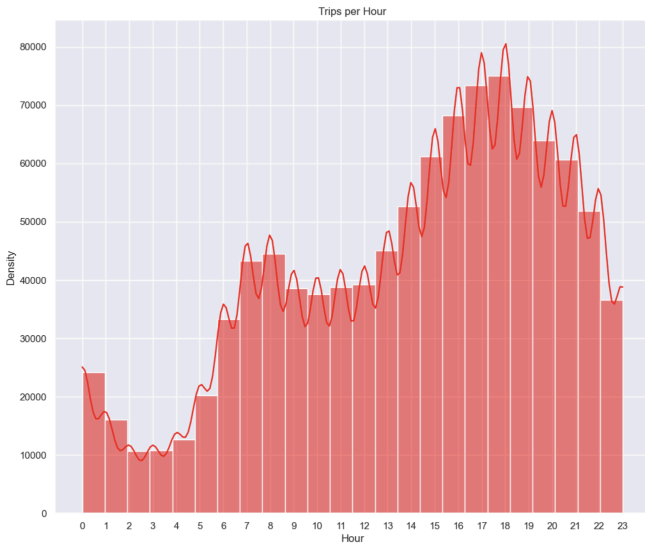
scatter_plot() function creates a scatter plot to analyze Uber trips based on geographical coordinates of Latitude and Longitude.

A scatter plot is a type of data visualization that displays individual data points on a two-dimensional graph. It is used to observe relationships or patterns between two continuous variables, with each point representing a unique data observation, and the horizontal and vertical axes representing the respective variables. Scatter plots are valuable for identifying trends, correlations, and outliers in the data.

6. RESULT

During the process, we have gathered numerous data points, visualized by graphs and charts that provide us with valuable insights.

1. Trips by the hours in a day:



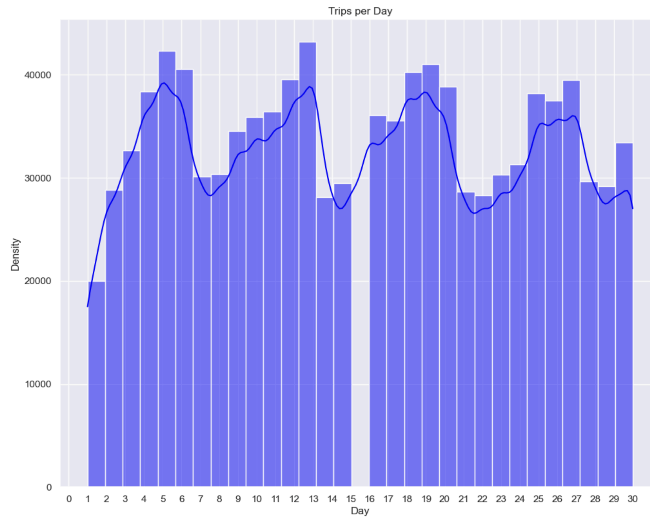
The graph shows the number of Uber trips taken each hour overall on an average day during September 2014.

Here are some insights that can be drawn from the graph:

- There are two clear peaks in the number of Uber trips taken, one in the morning around 7:00 AM and another in the evening around 6:00 PM. This suggests that Uber is most popular for commuting to and from work.
- There is a smaller peak in the number of trips taken around 10:00 AM and 3:00 PM. This could be due to people taking Uber for errands or other appointments during the day.
- The number of Uber trips taken is relatively low between 10:00 PM and 5:00 AM. This suggests that Uber is less popular for late-night and early-morning travel.

Overall, the graph suggests that Uber is most popular for commuting to and from work, but it is also used for other purposes throughout the day.

2. Trips during every day of the month:

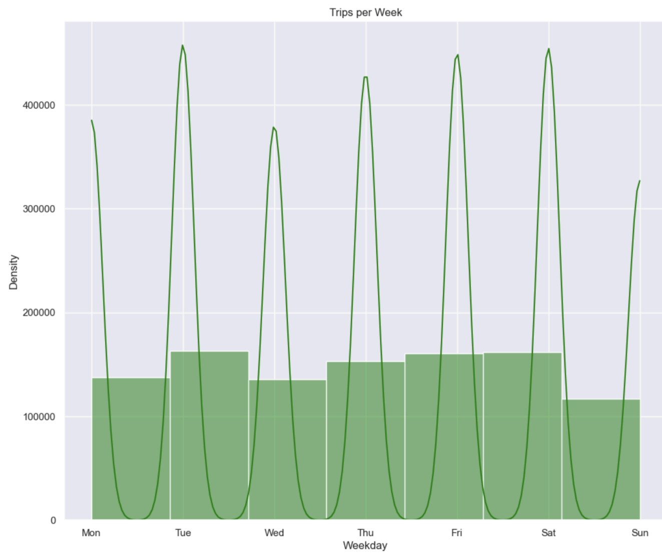


The graph shows a clear upward trend in the number of trips throughout the month, with a few dips and plateaus. Here are some insights that can be drawn from the graph:

- Increasing demand: The overall upward trend suggests that Uber was becoming increasingly popular in September 2014. This could be due to a number of factors, such as increased awareness of the service, lower prices, or a growing need for convenient transportation.
- Weekend effect: There appear to be dips in the number of trips on weekends (around days 7, 14, 21, and 28). This suggests that Uber usage is lower on Saturdays and Sundays, perhaps because people are more likely to be at home or engaged in activities that don't require transportation.
- Mid-week peak: There is a noticeable peak in the number of trips around day 18 or 19. This could be due to a number of factors, such as a specific event that happened in the middle of the week, or simply a typical mid-week increase in business travel or social activities.

Overall, the graph suggests that the service was becoming increasingly popular.

3. Trips taking place during weekdays:

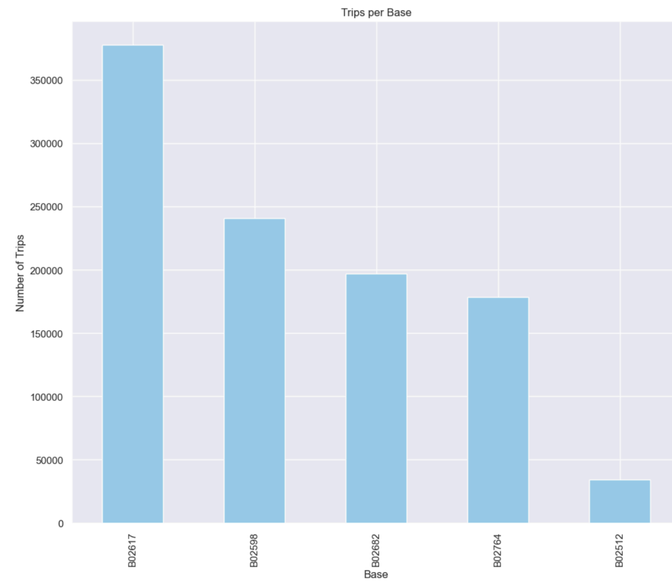


The graph shows the number of Uber trips taken each day of the week's average during September 2014. Here are some insights that can be drawn from it:

- Weekday surge: There is a clear peak in the number of trips taken on weekdays, particularly Monday through Thursday. This suggests that Uber is most popular for commuting to and from work or school.
- Weekend dip: The number of trips taken drops significantly on weekends, especially on Sundays. This suggests that people are less likely to use Uber for leisure activities or errands compared to commuting.
- Friday exception: Interestingly, the number of trips taken on Fridays is closer to the weekday peak than the weekend dip. This could be due to people using Uber for social activities or late-night outings on Fridays.
- Consistent trend: The overall pattern of the graph remains consistent throughout the month, with weekdays consistently having higher trip counts than weekends. This suggests that the weekday commuting trend is a reliable pattern for Uber usage in September 2014.

Overall, the graph suggests that Uber usage in September 2014 was heavily driven by commuting during the workweek. While there is some Uber activity on weekends, it is significantly lower than during the week. This aligns with the typical commuting patterns of most urban populations.

4. Number of Trips by bases:

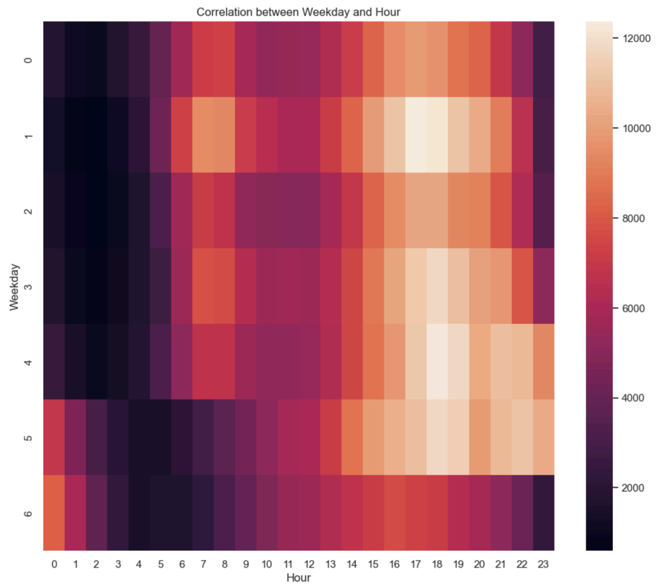


The graph shows the number of Uber trips taken per base during September 2014. Here are some insights that can be draw from it:

- Base B02617 had the most trips by a significant margin. This base likely had a larger service area or was located in a more densely populated area with higher demand for Uber services.
- There is a significant gap between the number of trips at base B02617 and the other bases. This suggests that the distribution of Uber trips is uneven across different bases, with some bases being much more popular than others.
- The other bases (B02598, B02764, B02512, and B02682) have a more similar number of trips. These bases may be more evenly distributed across the city or located in areas with similar demand for Uber services.

Overall, the graph suggests that there is a significant variation in the popularity of different Uber bases in September 2014. Base B02617 was the clear leader in terms of trip volume, while the other bases had a more similar distribution of trips.

5. Correlation Heatmap between Weekday and hour:



The graph shows the correlation between weekday and hour data for Uber trips during September 2014. Here are some insights that can be drawn from it:

- Strong positive correlation during morning and evening rush hours: The heatmap shows a strong positive correlation (yellow and orange areas) between weekdays and hours around 7:00 AM and 6:00 PM. This indicates that there is a significant increase in Uber trips during these times on weekdays, likely due to commuting to and from work.
- Weaker correlation during midday and late night: The heatmap shows a weaker positive correlation (light orange and yellow areas) between weekdays and hours around midday (11:00 AM to 3:00 PM) and late night (10:00 PM to 4:00 AM). This suggests that there is still some Uber activity during these times on weekdays, but it is not as concentrated as during rush hours.
- Negative correlation on weekends: The heatmap shows a negative correlation (blue areas) between weekdays and hours throughout the weekend (Saturday and Sunday). This indicates that there is a decrease in Uber trips on weekends compared to weekdays.
- Specific weekday patterns: The heatmap also shows some specific patterns for individual weekdays. For example, Tuesdays and Wednesdays appear to have slightly higher correlations with morning rush hours than Mondays and Thursdays. Fridays appear to have a slightly higher correlation with evening rush hours compared to other weekdays.

Overall, the graph suggests that Uber usage in September 2014 was heavily influenced by the workweek, with strong correlations between weekdays and commuting hours. Weekend usage was lower, and there were some variations in usage patterns between different weekdays.

6. Scatter Plot Visualization of geographic locations:

The graph is a scatter plot, where each point represents an Uber trip and its position is determined by the pickup location's latitude and longitude.

Here are some insights that can be drawn from the graph:

1. Concentration of trips in specific areas:
   - The data points are not uniformly distributed across the map. Instead, they are clustered in several areas, indicating a higher density of Uber trips in those locations.
   - Some of the most notable clusters are:
     - Around 40.7°N latitude and -73.9°W longitude, which is likely to be Midtown Manhattan.
     - Around 40.6°N latitude and -74.0°W longitude, which could be Downtown Brooklyn or the Financial District.
     - Around 40.7°N latitude and -73.8°W longitude, which might be near Central Park.
2. Spatial patterns:
   - There appears to be a north-south trend in the distribution of Uber trips, with more trips concentrated in the southern part of the map (around -74.0°W longitude) compared to the northern part (around -73.5°W longitude). This could be due to factors like population density or the location of major attractions and business districts.
   - There is also a westward trend, with more trips concentrated towards the left side of the map (around -74.0°W longitude) compared to the right side (around -73.5°W longitude). This could be due to factors like the location of major transportation hubs or one-way streets that make it easier to travel in certain directions.

Overall, the graph provides a valuable snapshot of Uber trip locations in September 2014 and suggests that Uber usage was concentrated in certain areas and followed some spatial trends.

7. CONCLUSION

The Uber data analysis project using Python showcased the creation of insightful data visualizations. Leveraging Matplotlib and Seaborn libraries, we effectively plotted diverse visualizations across different timeframes throughout the month. By exploring patterns and distributions, we gained valuable insights into the impact of time and location on customer trips.

This analysis not only provided a comprehensive understanding of Uber's operational trends but also highlighted the significance of temporal and spatial factors in influencing customer behavior. The visualizations produced through Python libraries served as powerful tools to draw meaningful conclusions about the dynamics of Uber trips in New York City.

8. FUTURE SCOPE

The data presents an opportunity for training a machine learning model and constructing an intelligent AI-driven predictive system. Such a model could autonomously relay insights to authorities or drivers regarding high-activity areas and passenger counts in specific regions. This extensive dataset offers the potential to delve into passenger behavior, unlocking valuable knowledge for future applications and enhancements in the transportation sector.

9. DISCUSSION

Unveiling Insights in the Uber Trips Data Landscape:

Our analysis of Uber trip data in New York City has unveiled compelling insights into the intricacies of ride-sharing dynamics. The temporal exploration of hourly and monthly trip distributions exposed distinct patterns, highlighting peak hours and seasonal variations. These revelations offer a valuable roadmap for optimizing service operations, ensuring responsiveness during periods of heightened demand.

Delving deeper into the geographical landscape, our analysis identified untapped markets in suburban areas, signaling potential growth opportunities for Uber. The examination of Uber bases shed light on the distribution of active vehicles, paving the way for strategic planning to efficiently meet demand. These findings underscore the project's contribution to not only understanding past performance but also laying a foundation for future advancements in the ride-sharing industry.

The prospect of employing machine learning models for predictive insights emerges as a key takeaway. Integrating such models could enable a proactive system, notifying authorities and drivers about areas with high trip density and passenger counts. This forward-looking approach aligns with the evolving paradigm of smart transportation systems, offering a glimpse into the future of data-driven decision-making in the realm of urban mobility.

10. ACKNOWLEDGEMENT

The authors would like to express sincere gratitude to Mr.Daniel Evans for his invaluable guidance and assistance throughout the CS661 Python Programming course at Pace University. The Data Analysis and Visualizations were computed using Jupyter Notebook.

11. REFERENCES

[1] Uber Engineering. "An Efficient Real-time Taxi Dispatching Algorithm." https://www.uber.com/en-US/blog/engineering/

[2] Uber Movement. "Explore Traffic and Mobility Trends." https://www.uber.com/us/en/community/supporting-cities/data/

[3] Kaggle Datasets - Uber Pickups in New York City: https://www.kaggle.com/datasets/fivethirtyeight/uber-pickups-in-new-york-city

[4] Towards Data Science - Uber Analysis and Visualization: https://towardsdatascience.com/tagged/uber

[5] Pandas Documentation. "Pandas: Powerful data structures for data analysis, time series, and statistics." - https://pandas.pydata.org/pandas-docs/stable/index.html

[6] Matplotlib Documentation. "Matplotlib: Visualization with Python." - https://matplotlib.org/stable/contents.html

[7] Seaborn Documentation. "Seaborn: Statistical Data Visualization." - https://seaborn.pydata.org/index.html

[8] Rishi Srinivas, B. Ankayarkanni, and R. Sathya Bama Krishna: Uber Related Data Analysis using Machine Learning, published by Institute of Electrical and Electronics Engineers(IEEE).
https://ieeexplore.ieee.org/document/9432347

[9] Shashank H: Data Analysis of Uber and Lyft Cab Services, published by International Journal of Interdisciplinary Innovative Research & Development (IJIIRD).
https://ijiird.com/wp-content/uploads/050144.pdf