Exploratory Data Analysis

New York City Yellow Taxi Data (2023)

A Comprehensive Report

Prepared by:

Ashad Ahmed
DevOps Engineer, Nokia

TABLE OF CONTENTS

1 Data Preparation

1.1 Load the dataset

1.1.1 Sampling and Combining

The original dataset comprised 12 monthly .parquet files, each containing approximately 3 million trip records, resulting in a total of over 36 million observations. While such a comprehensive dataset offers rich analytical potential, processing the full volume poses significant computational challenges, especially during exploratory data analysis (EDA) where interpretability and efficiency are crucial.

To ensure a balanced representation across all months while maintaining resource efficiency, a stratified sampling approach was adopted. Specifically, 1% of records were randomly selected from each monthly file and subsequently combined into a unified DataFrame. This yielded a manageable yet sufficiently diverse dataset that preserved temporal granularity and enabled effective trend analysis throughout the calendar year.

The final sampled file named "Hourly_Sampled_Data.parquet" has 375186 records which is sufficient and approximately in the range of recommended data size.

2 Data Cleaning

2.1 Fixing Columns

2.1.1 Fixing Index and Dropping Columns

After consolidating the sampled records from the 12 monthly .parquet files, the resulting dataset exhibited a non-sequential index due to the randomized sampling process. To ensure a clean and consistent structure suitable for analysis, the index was reset, assigning continuous integer-based indices to all rows.

During the column review, the store_and_fwd_flag column was removed, as it did not offer meaningful insights for our intended exploratory analysis.

2.1.2 Combining the Airport_fee and airport_fee columns

Additionally, the dataset contained two columns representing the same information, Airport_fee and airport_fee. To resolve this redundancy, all missing (NaN) values in Airport_fee were replaced using the corresponding values from airport_fee, following which the redundant airport_fee column was dropped.

2.1.3 Fix columns with negative (monetary) values

A data integrity check revealed a small number of records (16 in total, approximately 0.004% of the dataset) containing negative values in monetary fields such as extra, mta_tax, improvement_surcharge, total_amount, congestion_surcharge, and Airport_fee. Since negative amounts in these fields are not logically valid, these records were removed.

Furthermore, the VendorID column had 60 records where the value was 6, which is not compliant with the data documentation that specifies valid VendorIDs to be either 1 or 2. These anomalies were corrected by replacing the invalid values with the mode of the column, ensuring conformity with expected standards.

2.2 Handling Missing Values

2.2.1 Find the proportion of missing values in each column

	Non-Null Count	Null Count	Null Percentage
VendorID	375170	0	0.000000
tpep_pickup_datetime	375170	0	0.000000
tpep_dropoff_datetime	375170	0	0.000000
passenger_count	362415	12755	3.399792
trip_distance	375170	0	0.000000
RatecodeID	362415	12755	3.399792
PULocationID	375170	0	0.000000
DOLocationID	375170	0	0.000000
payment_type	375170	0	0.000000
fare_amount	375170	0	0.000000
extra	375170	0	0.000000
mta_tax	375170	0	0.000000
tip_amount	375170	0	0.000000
tolls_amount	375170	0	0.000000
improvement_surcharge	375170	0	0.000000
total_amount	375170	0	0.000000
congestion_surcharge	362415	12755	3.399792
date	375170	0	0.000000
hour	375170	0	0.000000
Airport_fee	362415	12755	3.399792

2.2.2 Handling missing values in passenger_count

The passenger_count column contained a small number of missing values as well as several records where the passenger count was recorded as zero. Since it is not practical for a passenger trip to occur with zero passengers, these values were treated as invalid. Rather than discarding these records, we opted to impute the missing and zero values using the mode of the column.

Using the mode is a reasonable choice in this context because passenger_count is a discrete, categorical-type variable (typically ranging from 1 to 6), and the mode represents the most common real-world scenario, usually single-passenger trips in urban taxi services. This approach ensures data consistency while preserving the maximum amount of valid trip records for further analysis.

2.2.3 Handle missing values in RatecodeID

The RatecodeID column had a small number of missing values as well as some entries with the value 99, which is not part of the standard codes defined in the dataset documentation. This value likely represents either an erroneous entry or a placeholder for an unknown rate code. To ensure consistency and avoid introducing bias through manual imputation, we replaced both the missing and the anomalous RatecodeID values with the mode of the column.

The mode was chosen as a suitable imputation method because RatecodeID is a categorical field with a limited number of predefined values, and the most frequently occurring code likely represents the default or standard rate used across most trips.

2.2.4 Impute NaN in congestion_surcharge

To address missing data in surcharge-related columns, we applied different imputation strategies based on the nature and distribution of each field.

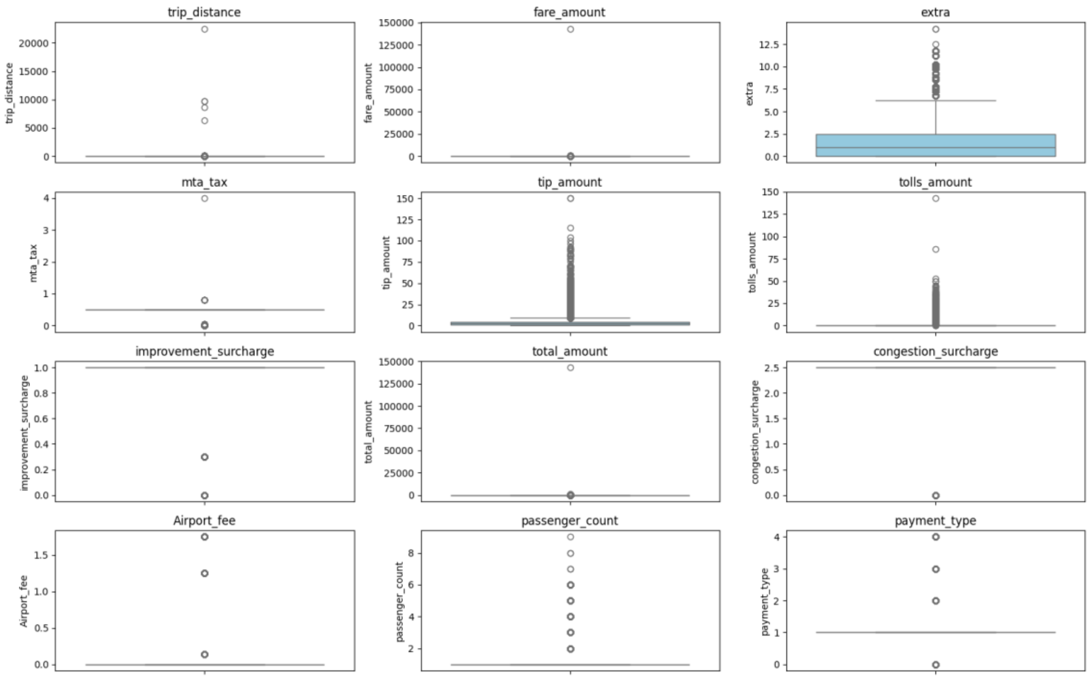
For the congestion_surcharge column, we filled missing values using the median. This approach helps avoid the influence of extreme values or outliers and preserves the central tendency of the distribution, which is especially relevant for surcharge values that can vary due to fluctuating traffic patterns.

The Airport_fee column had a significant number of missing values. Although the airport fee is generally a fixed charge, imputing a constant value (such as 0 or a predefined fee) for all missing entries could introduce uniformity that may not reflect the actual conditions. Instead, we used the mean value of the non-missing entries to fill the gaps. This provided a more balanced representation while accounting for some variability due to differing pickup/drop-off conditions around airport zones.

2.3 Handling Outliers

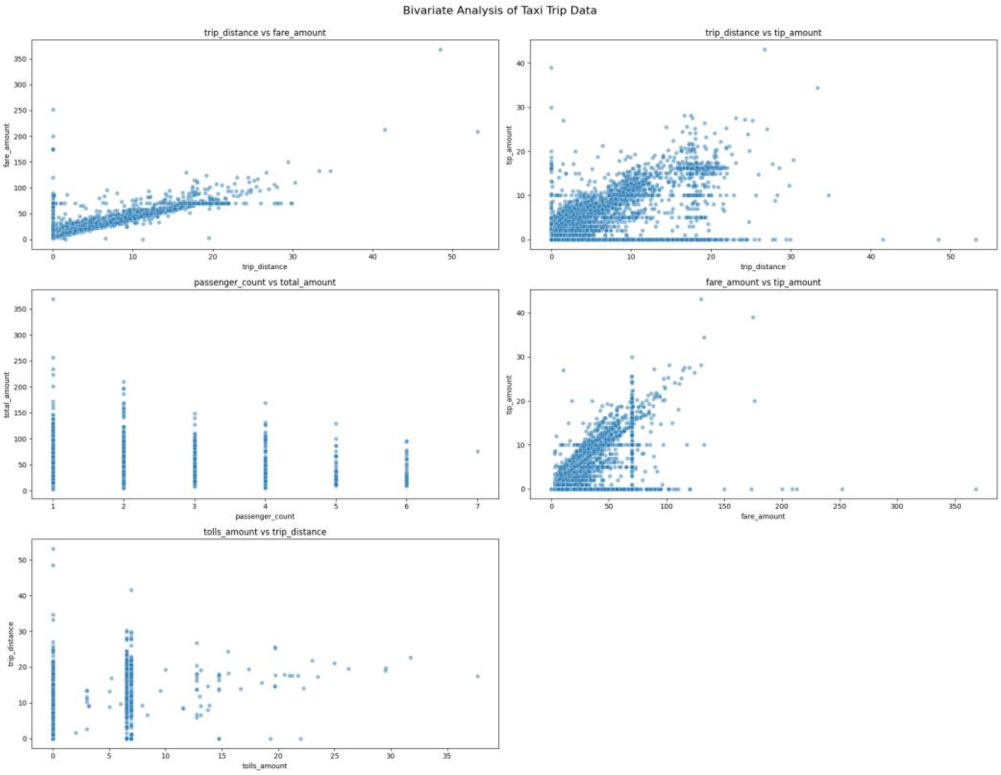
We did a detailed Univariate, Bivariate and Multivariate analysis of all the numerical columns. Below are some of the important plots that helped us discover hidden outliers.

A. Univariate Analysis:



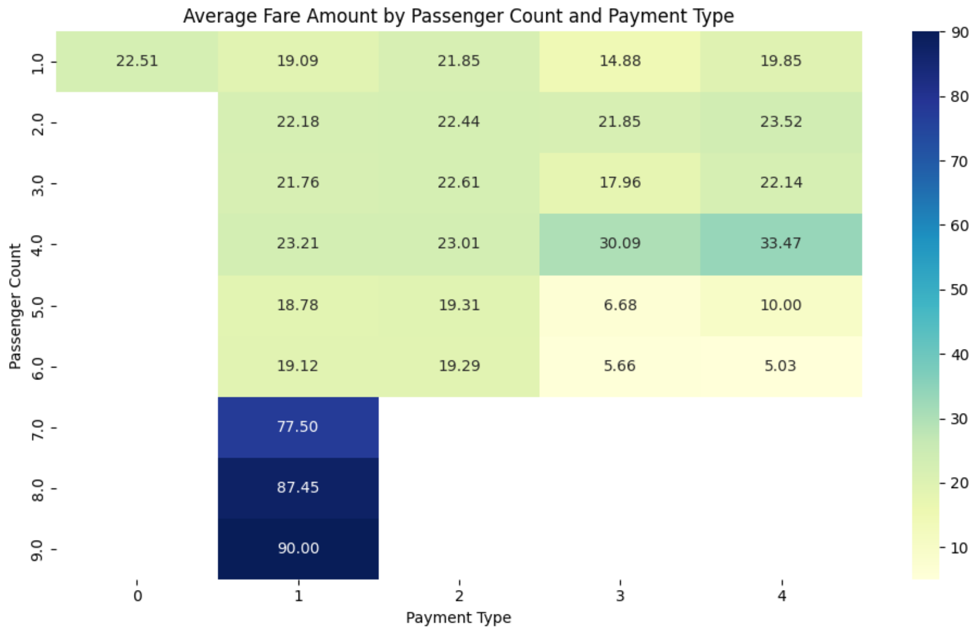
We can clearly observe outliers in trip distance, fare amount and even total amount affected by fare amount. Payment type 0 and negative mta_tax seem unusual. The univariate plots tell a lot about the spread of data, skewness and possible outliers.

B. Bivariate Analysis:



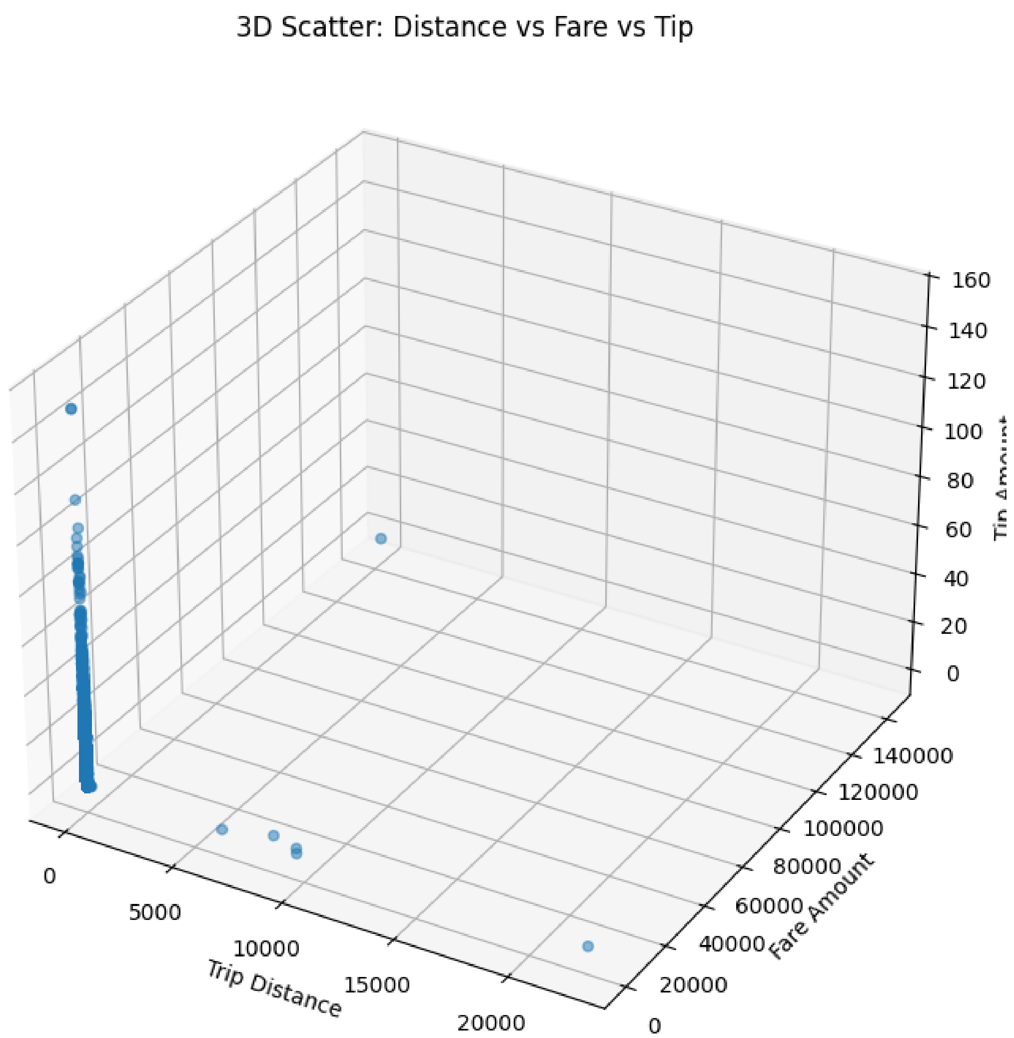
Looking at these plots we can clearly observe the few outliers in our data that need to be handles appropriately. We can also appreciate the correlation trip distance vs amount and trip distance vs tip amount have.

C. Multivariate Analysis:



The first thing we can learn from above plot is the passenger counts data needs to be fixed.

We also used a 3D Scatter of Distance vs Fare vs Tip for our analysis.



2.3.1 Finding, Interpreting and Handling Outliers

Outliers were addressed to ensure the accuracy of our analysis. Records with passenger_count greater than 6 were removed, as standard NYC taxis do not support more than six passengers. Similarly, 6 trips had trip_distance over 250 miles, extreme values that were excluded due to their improbability and potential to skew results.

Trips with unusually low distances (less than 0.1 miles) but extremely high fare_amount (above $300) were also dropped, as were cases where both distance and fare were zero despite different pickup and drop-off zones, suggesting data issues.

For entries where payment_type was incorrectly marked as 0, we used the presence of a tip_amount to infer and assign the correct type (credit card), since only card payments include tips.

3 Exploratory Data Analysis

3.1 General EDA: Finding Patterns and Trends

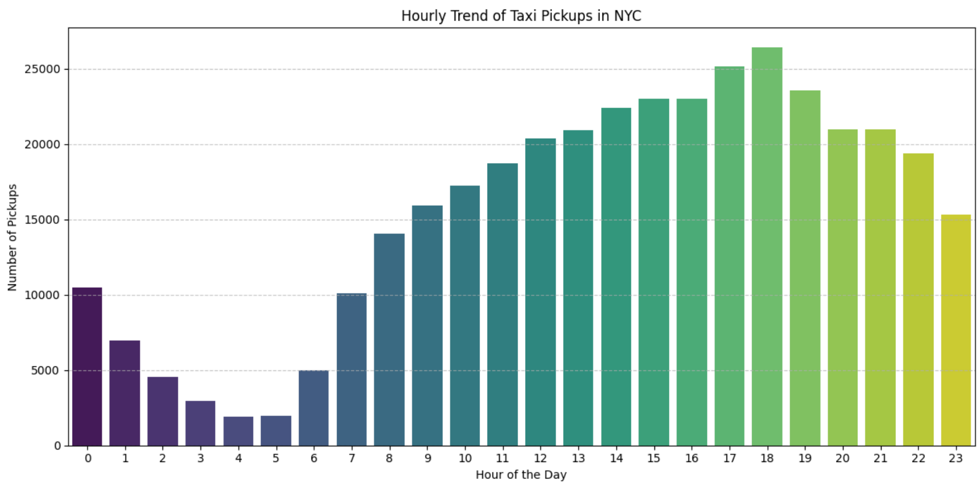
3.1.1 Categorise the varaibles into Numerical or Categorical

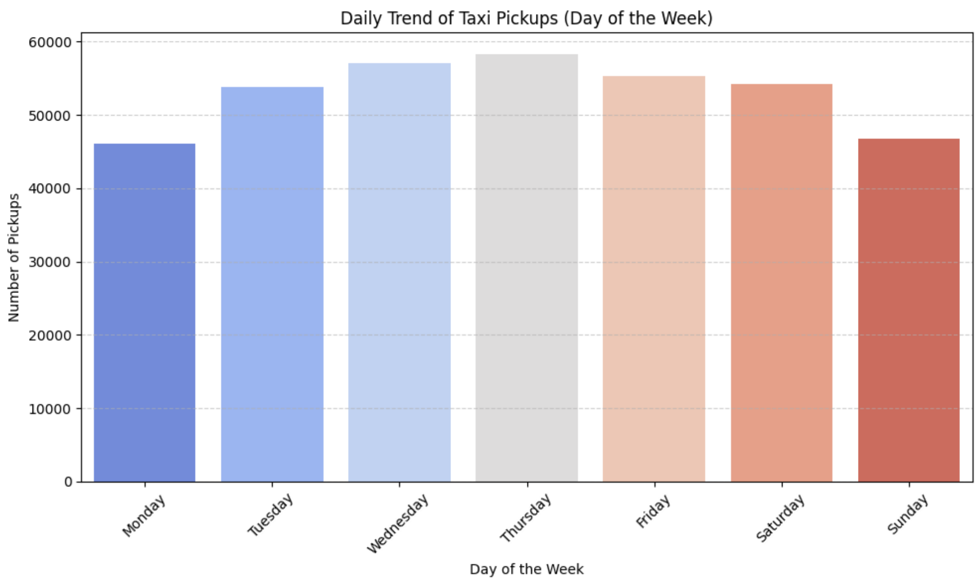
The following monetary parameters belong in the same category, is it categorical or numerical?

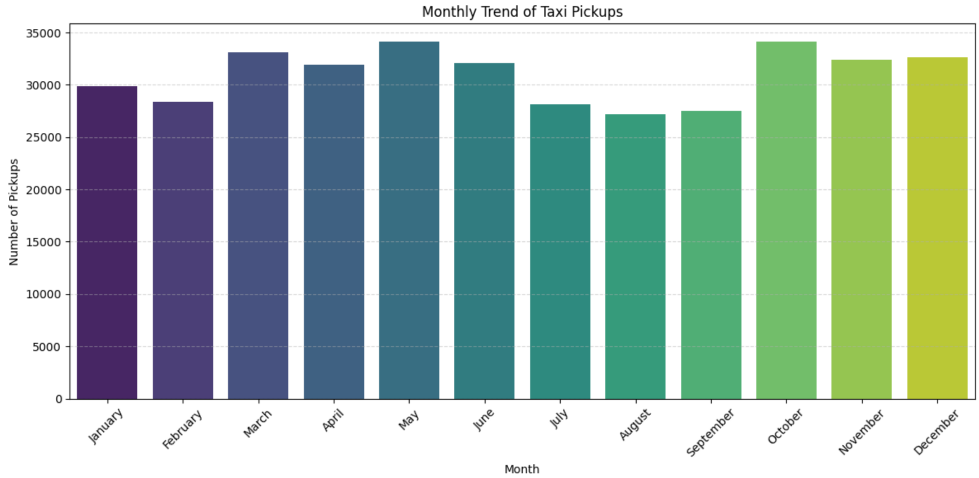
- `fare_amount`
- `extra`
- `mta_tax`
- `tip_amount`
- `tolls_amount`
- `improvement_surcharge`
- `total_amount`
- `congestion_surcharge`
- `airport_fee`

Answer: The monetary parameters listed above all are Numerical.

3.1.2 Analyse the distribution of taxi pickups by hours, days of the week, and months







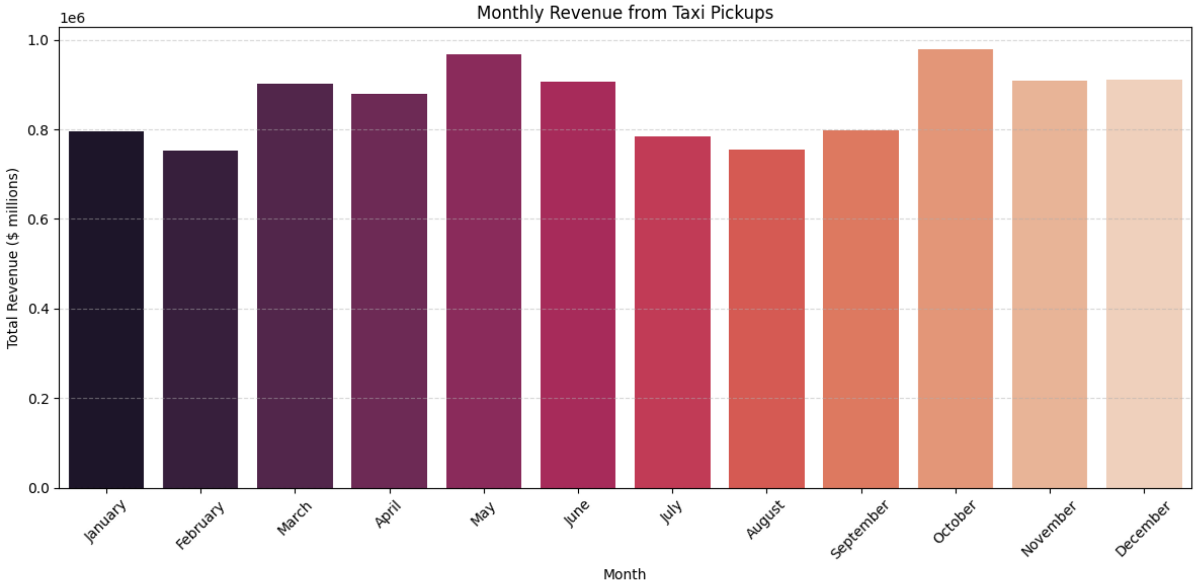
3.1.3 Filter out the zero values from the above columns.

As suggested, we've filtered out zero valued records from 'fare_amount' and 'total_amount' from our data as these had very few numbers of trips.

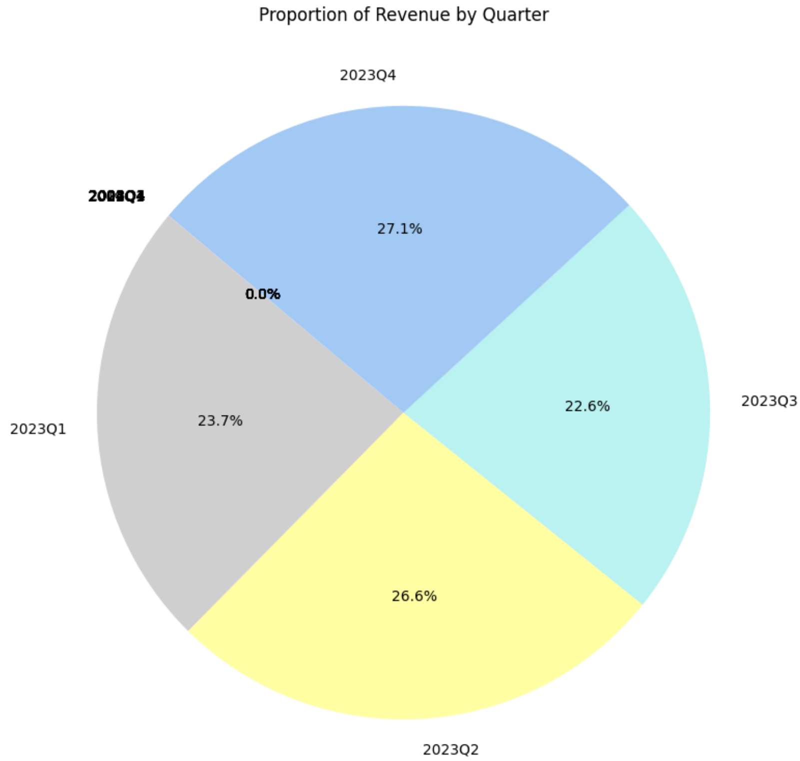
However, we did not remove the zero values from tip_amount as there were significant numbers of records. A person may decide to not tip the driver for personal reasons and hence it makes sense to keep these records.

We also removed zero values from trip_distance as they had zero fare as well as different zones which do not make sense and may corrupt our analysis.

3.1.4 Analyse the monthly revenue (`total_amount`) trend

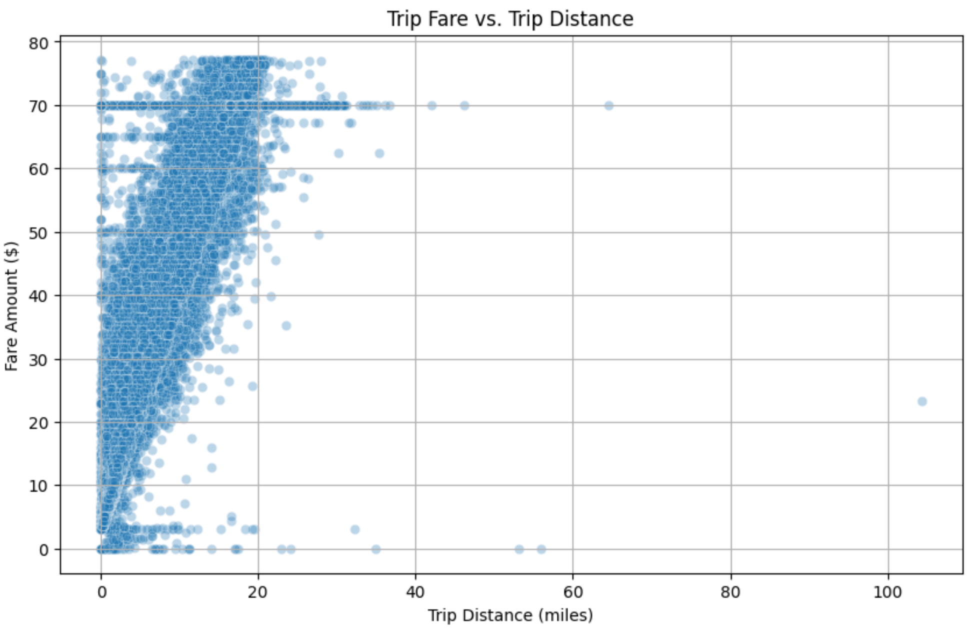


3.1.5 Show the proportion of each quarter of the year in the revenue



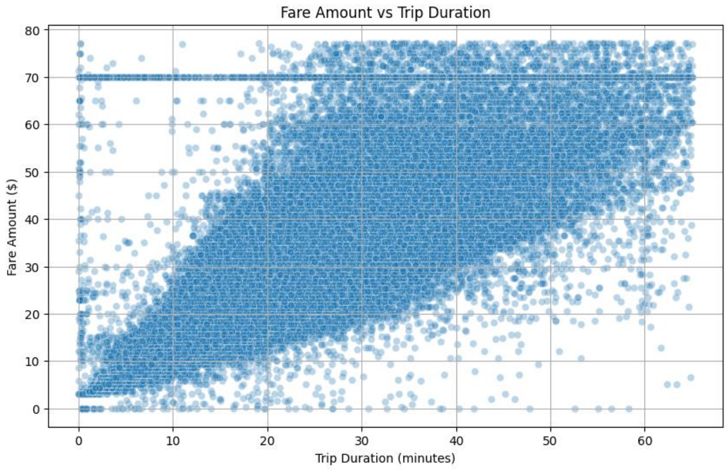
3.1.6 Visualise the relationship between `trip_distance` and `fare_amount` and find correlation

Note: We found a correlation value of 0.95 for trip_distance vs Trip fare

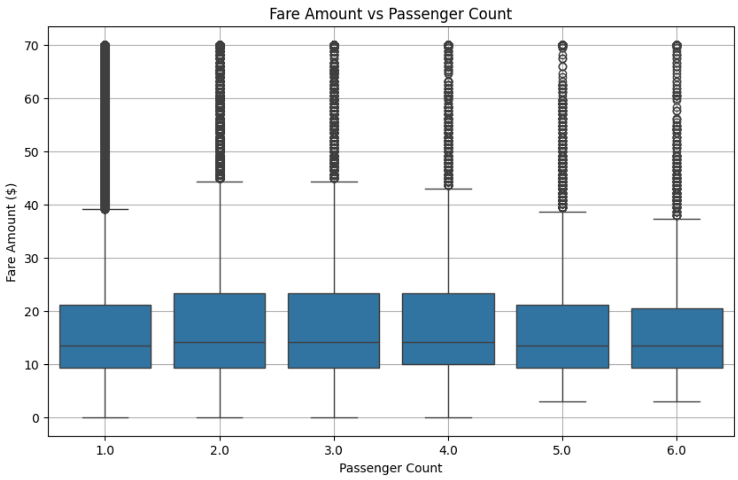


3.1.7 Find and visualise the correlation between Fare & Trip, Fare & Passenger count, Tip and Trip Distance

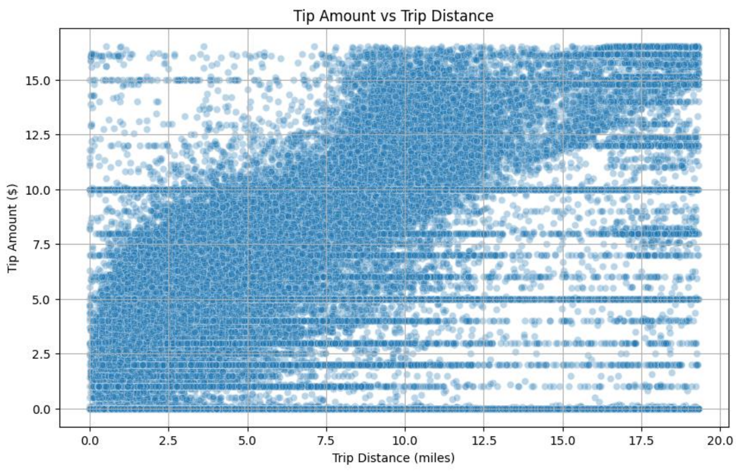
Note: We found a correlation value of 0.88 for Fare vs Trip Duration



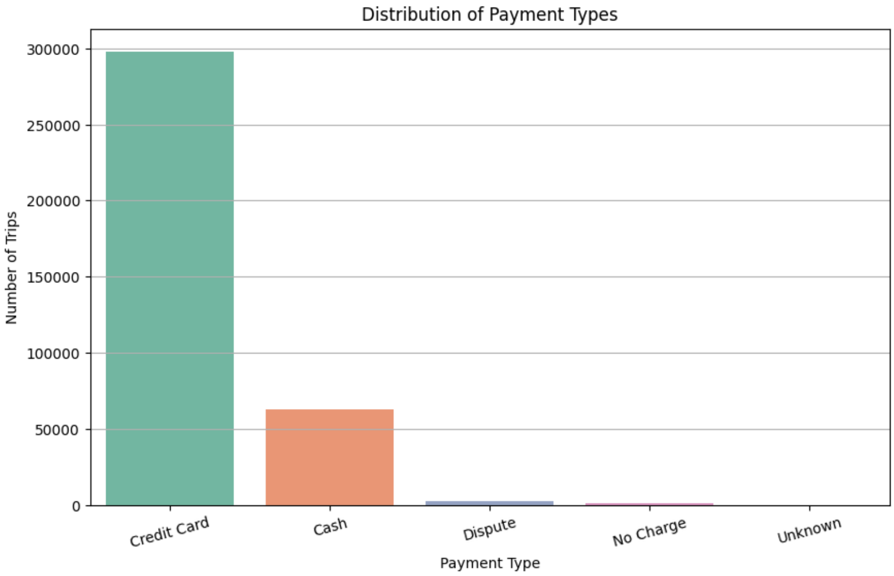
Note: We found a correlation value of 0.03 for Fare vs Passenger Count



Note: We found a correlation value of 0.55 for Tip Amount vs Trip Distance

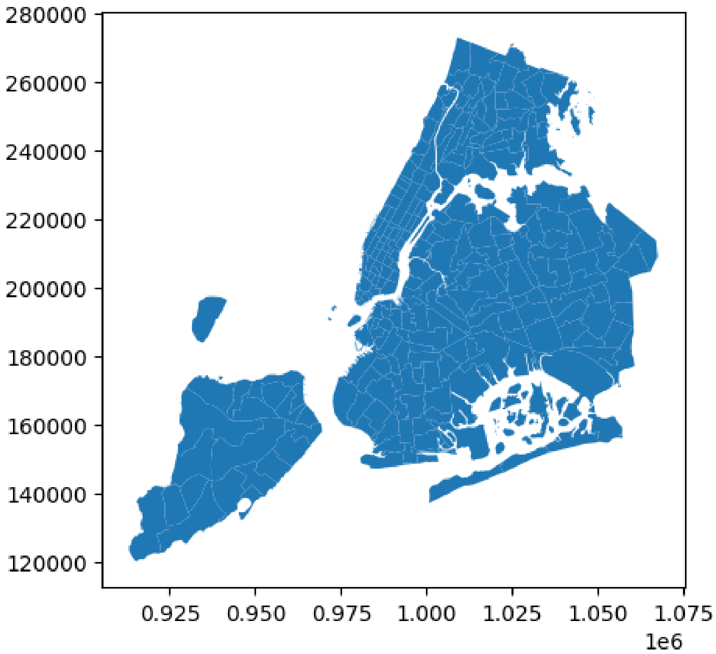


3.1.8 Analyse the distribution of different payment types



3.1.9 Load the shapefile and display it

	OBJECTID	Shape_Leng	Shape_Area	zone	LocationID	borough	geometry
0	1	0.116357	0.000782	Newark Airport	1	EWR	POLYGON ((933100.918 192536.086, 933091.011 19...
1	2	0.433470	0.004866	Jamaica Bay	2	Queens	MULTIPOLYGON (((1033269.244 172126.008, 103343...
2	3	0.084341	0.000314	Allerton/Pelham Gardens	3	Bronx	POLYGON ((1026308.77 256767.698, 1026495.593 2...
3	4	0.043567	0.000112	Alphabet City	4	Manhattan	POLYGON ((992073.467 203714.076, 992068.667 20...
4	5	0.092146	0.000498	Arden Heights	5	Staten Island	POLYGON ((935843.31 144283.336, 936046.565 144...



3.1.10 Merge zones & trip data using locationID and PULocationID

We merged the zones data and trip data using locationID and PULocationID as suggested.

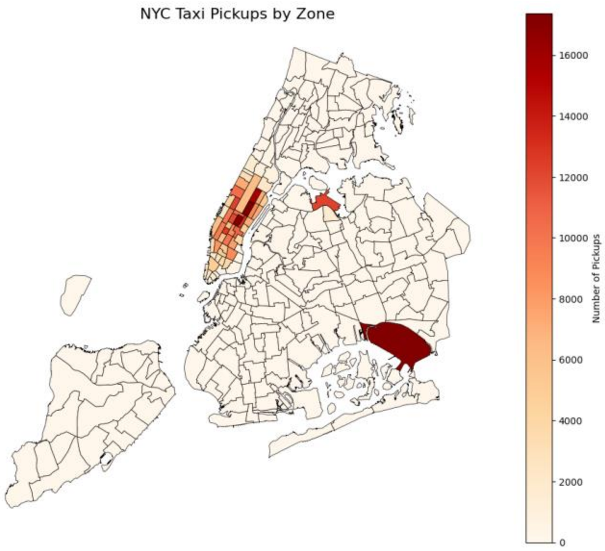
3.1.11 Group data by location IDs to find the total number of trips per location ID

	PULocationID	trip_count
212	237	17380
115	132	17306
143	161	17043
211	236	15631
144	162	13121

3.1.12 Add number of trips to the GeoDataFrame

	OBJECTID	Shape_Leng	Shape_Area	zone	LocationID	borough	geometry	PULocationID	trip_count
0	1	0.116357	0.000782	Newark Airport	1	EWR	POLYGON ((933100.918 192536.086, 933091.011 19...	1.0	4
1	2	0.433470	0.004866	Jamaica Bay	2	Queens	MULTIPOLYGON (((1033269.244 172126.008, 103343...	NaN	0
2	3	0.084341	0.000314	Allerton/Pelham Gardens	3	Bronx	POLYGON ((1026308.77 256767.698, 1026495.593 2...	3.0	9
3	4	0.043567	0.000112	Alphabet City	4	Manhattan	POLYGON ((992073.467 203714.076, 992068.667 20...	4.0	416
4	5	0.092146	0.000498	Arden Heights	5	Staten Island	POLYGON ((935843.31 144283.336, 936046.565 144...	5.0	2

3.1.13 Plot a color-coded map showing zone-wise trips



Findings and conclusions:

- Busiest Hour: Most trips happen between 5 PM and 6 PM.
- Busiest Day: Thursday sees the highest number of trips in a week.
- Busiest Months: May and October have the highest number of pickups.
- Revenue Quarters: Q2 (26.6%) and Q4 (27.1%) contribute the most to annual revenue.
- Distance vs Fare: Very strong positive correlation (0.95). Longer trips usually cost more.
- Trip Duration vs Fare: Also highly correlated (0.88). Longer time, Higher fare.
- Passenger Count vs Fare: Very weak correlation (0.03)
- Trip Distance vs Tip Amount: Moderate positive correlation (0.55) – longer trips tend to get higher tips.
- Payment Method: Credit cards are the most used payment option.
- High-Demand Zones: JFK Airport and Downtown Manhattan consistently see the most pickups and drop-offs.

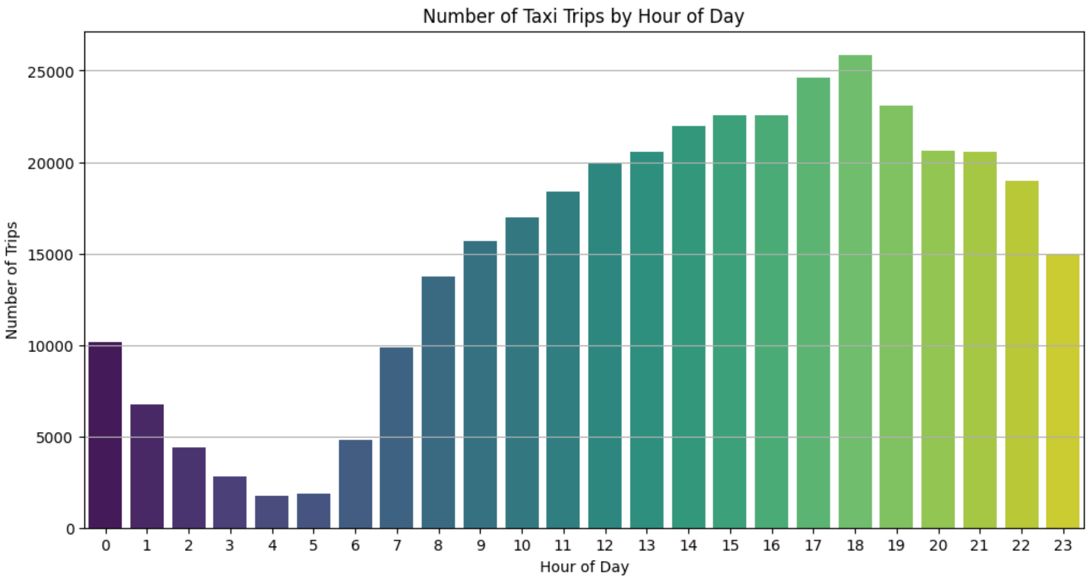
3.2 Detailed Analysis

3.2.1 Identify slow routes by averaging travel times between zones by hour

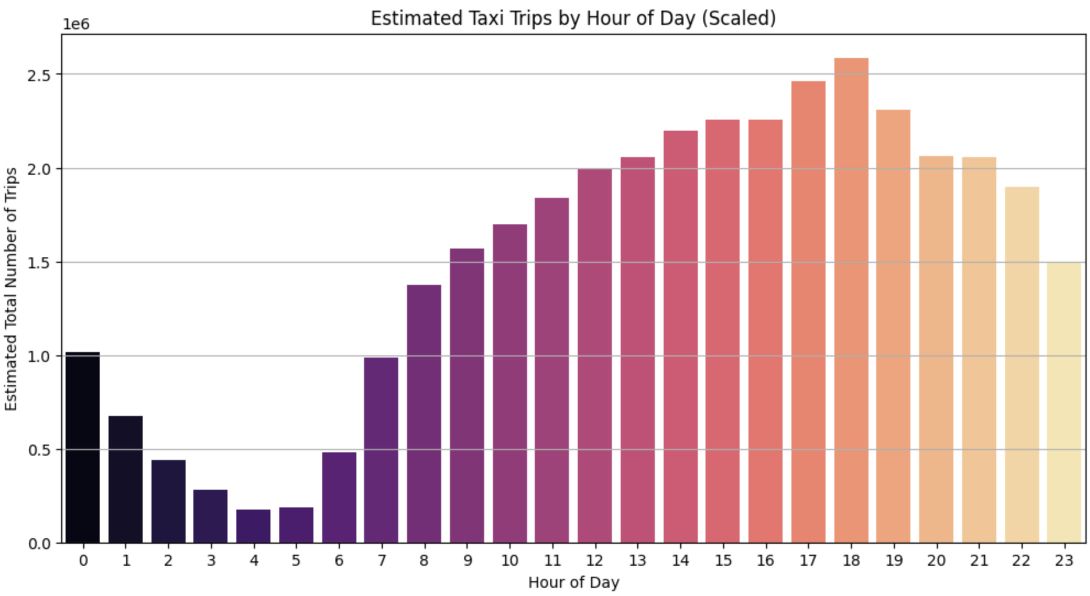
	PULocationID	DOLocationID	hour	trip_distance	trip_duration_hours	avg_speed_mph
51563	226	145	18	1.200000	45.165000	0.026569
66792	260	129	17	0.960000	23.560556	0.040746
19993	113	235	22	0.280000	5.820556	0.048105
6328	50	43	8	1.420000	23.855556	0.059525
35985	148	45	23	0.800000	12.065139	0.066307

3.2.2 Plot trips per hour, Identify the busiest hour and display its trip count

For our sample data the busiest hour is 18:00 with 25,868 trips.



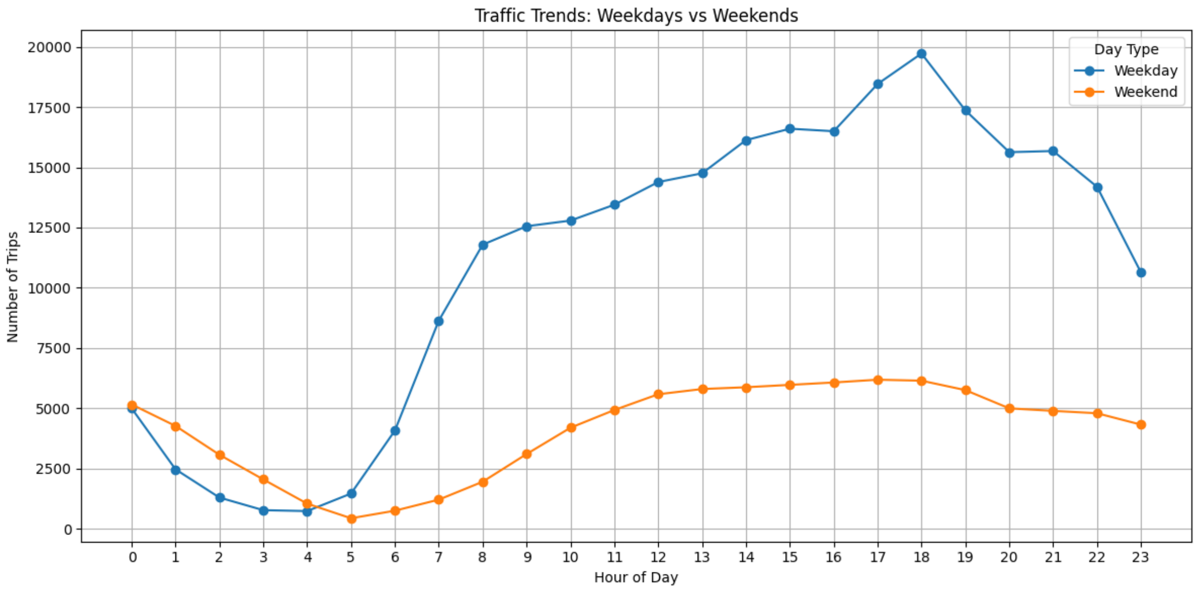
For the complete NYC data estimated busiest hour is 18:00 with approximately 2,586,800 trips.



3.2.3 Find the actual number of trips in the five busiest hours

```
Actual number of trips in the 5 busiest hours (scaled):
Hour 18:00 — 2,586,800 trips
Hour 17:00 — 2,464,500 trips
Hour 19:00 — 2,311,700 trips
Hour 15:00 — 2,257,400 trips
Hour 16:00 — 2,256,800 trips
```

3.2.4 Compare hourly traffic pattern on weekdays and weekend



3.2.5 Identify and visualize the top 10 zones with the highest hourly pickups and drop-offs

```
Top 10 Pickup Zones:
 PULocationID
237     17350
132     17194
161     17004
236     15602
162     13098
186     12489
138     12237
142     12146
230     11920
170     10799
dtype: int64
```

```
Top 10 Dropoff Zones:
 DOLocationID
236     16417
237     15587
161     14343
230     11123
170     10900
162     10465
142     10430
239     10230
141      9659
68       9376
dtype: int64
```

3.2.6 Find the top 10 and bottom 10 pickup/dropoff ratios

```
Top 10 Pickup/Dropoff Ratios:
 70      9.355030
132     4.474109
138     2.688860
23      2.333333
186     1.529952
43      1.387442
249     1.352592
114     1.350624
162     1.251601
161     1.185526
dtype: float64
```

```
Bottom 10 Pickup/Dropoff Ratios:
 192     0.025000
101     0.032258
178     0.040000
257     0.040000
31      0.045455
102     0.051948
64      0.052632
252     0.054054
120     0.055556
16      0.057143
dtype: float64
```

3.2.7 Find high pickup & dropoff traffic zones during 11PM to 5AM (Night)

```
Top 10 Night Pickup Zones:
 PULocationID
79      3107
132     2556
249     2551
48      2006
148     1932
114     1689
230     1660
186     1362
164     1208
138     1199
dtype: int64
```

```
Top 10 Night Dropoff Zones:
 DOLocationID
79      1694
48      1420
170     1260
107     1160
68      1148
141     1073
263     1025
249      926
229      887
236      879
dtype: int64
```

3.2.8 Find the revenue share for night and day hours

```
Nighttime Revenue Share: 12.00%
Daytime Revenue Share: 88.00%
```

3.2.9 Find fare per mile per passenger for different passenger count

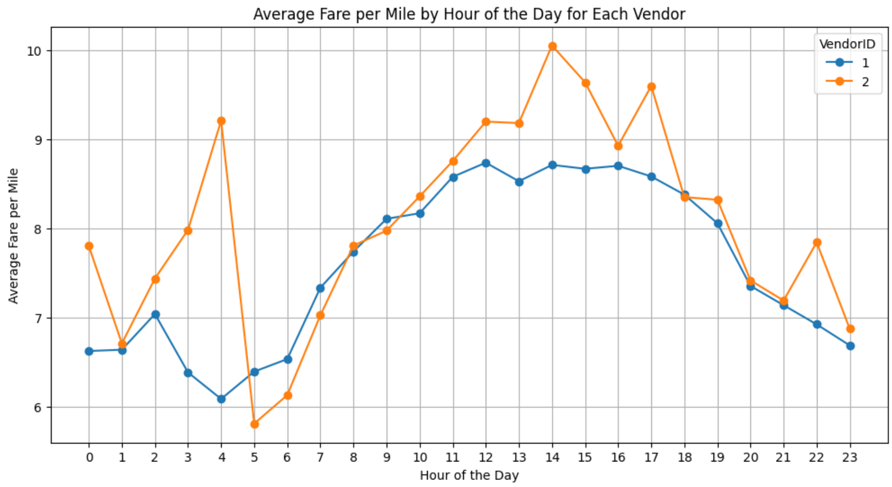
```
passenger_count
1.0     8.281668
2.0     4.187183
3.0     2.638525
4.0     2.206715
5.0     1.531042
6.0     1.278152
dtype: float64
```

3.2.10 Find average fare per mile by hours of the day and by days of the week

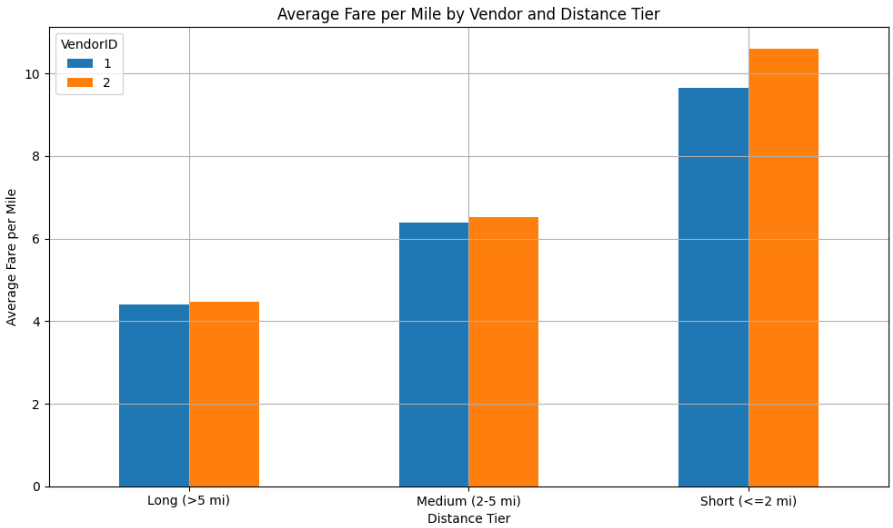
```
Average Fare per Mile by Hour of the Day:
pickup_hour
0     7.538711
1     6.695578
2     7.347154
3     7.627664
4     8.502352
5     5.973379
6     6.250145
7     7.116149
8     7.787959
9     8.011790
10    8.304450
11    8.705015
12    9.070738
13    8.994731
14    9.689186
15    9.373740
16    8.864649
17    9.326592
18    8.357537
19    8.255382
20    7.401506
21    7.178672
22    7.635153
23    6.831026
```

```
Average Fare per Mile by Day of the Week:
pickup_day
Friday       8.122037
Monday       8.094935
Saturday     8.439474
Sunday       7.585538
Thursday     8.540521
Tuesday      8.480576
Wednesday    8.538880
```

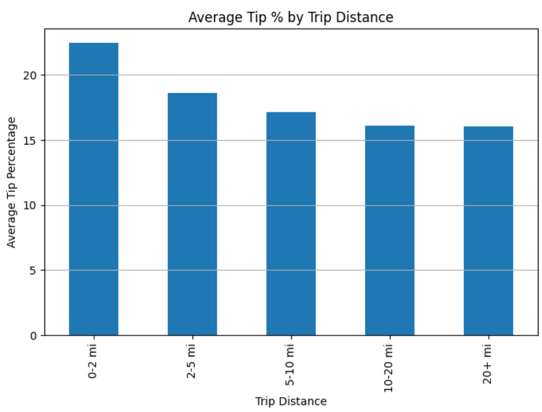
3.2.11 Analyze hourly average fare per mile by vendor

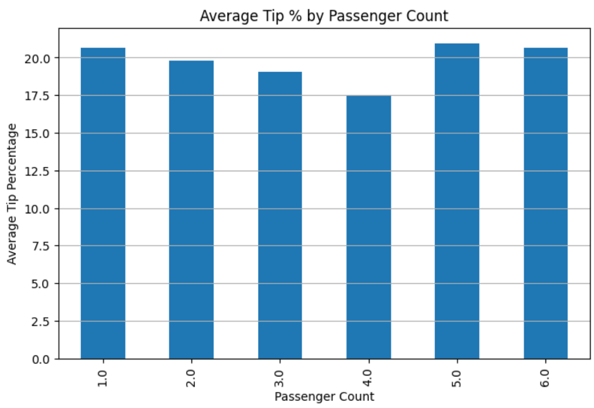


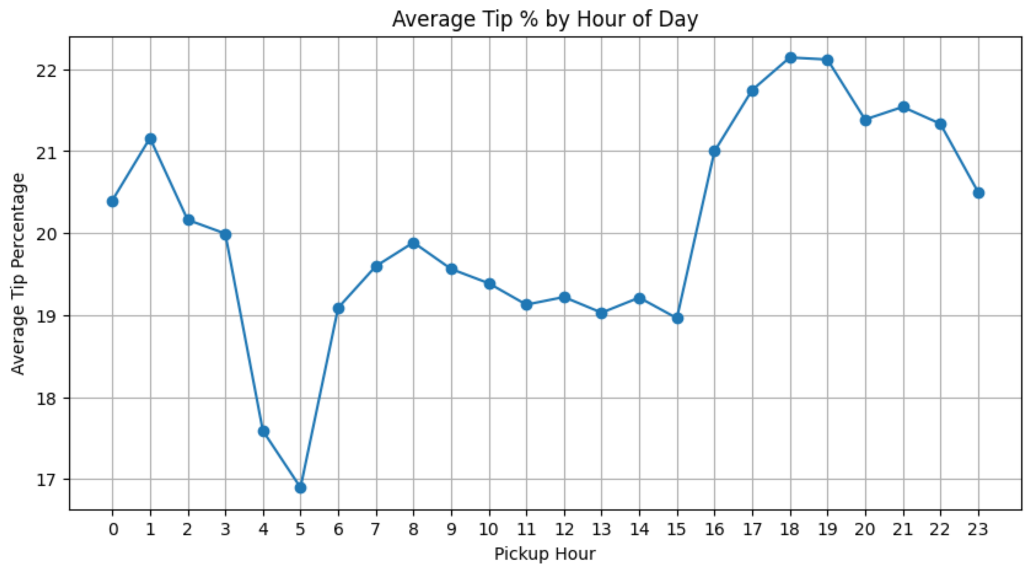
3.2.12 Compare vendors' fare per mile in tiers



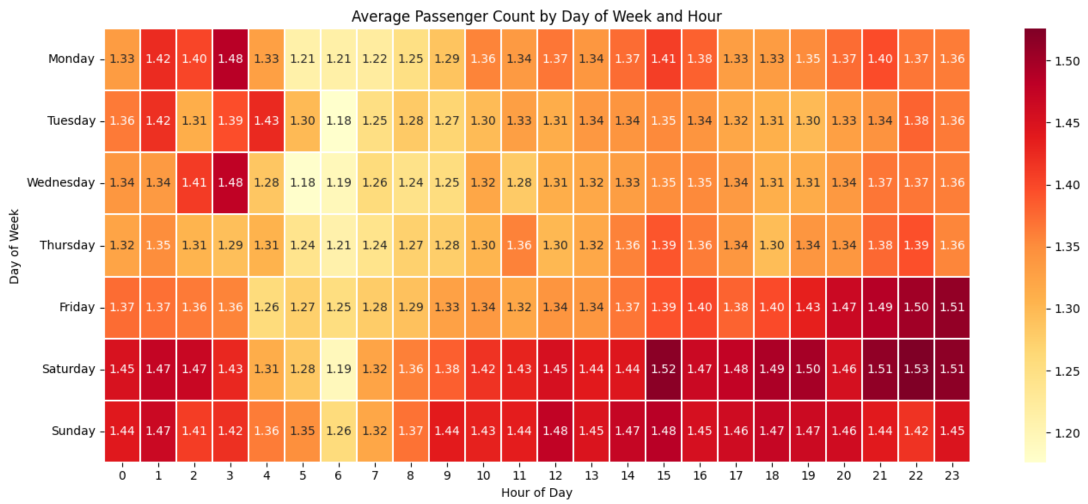
3.2.13 Analyze average tip % by distance, passenger count, and pickup time



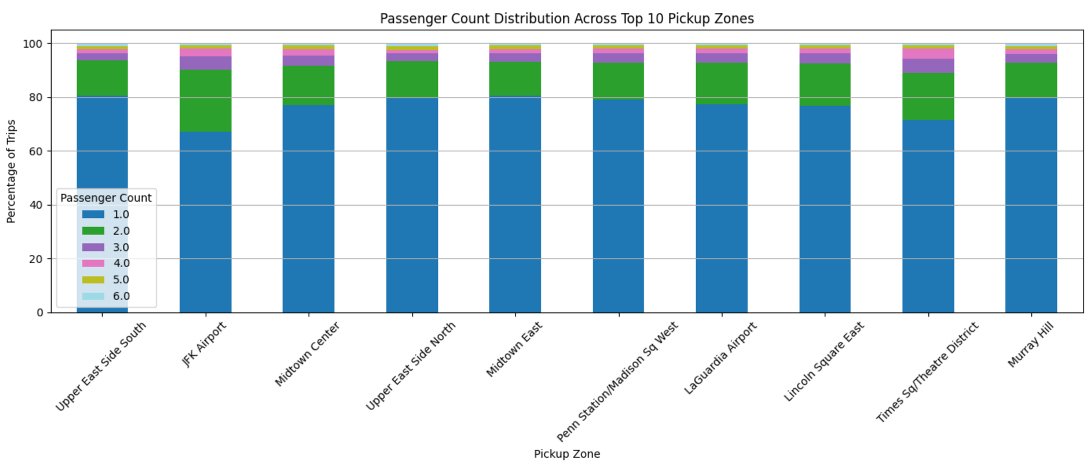




3.2.14 Analyse passenger count variation across hours and week days

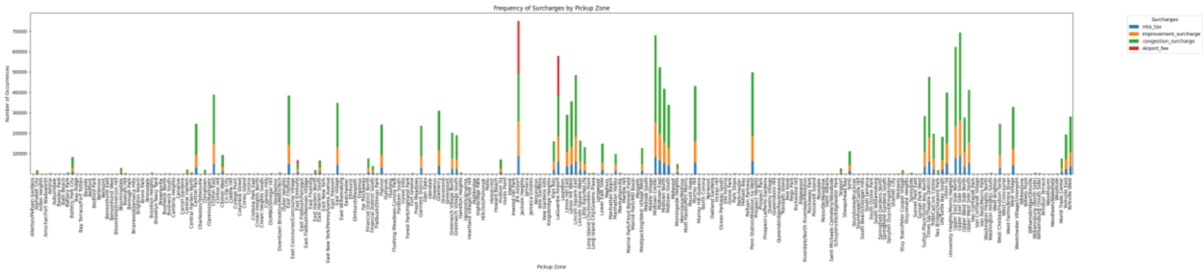


3.2.15 Analyse the passenger counts variation across zones

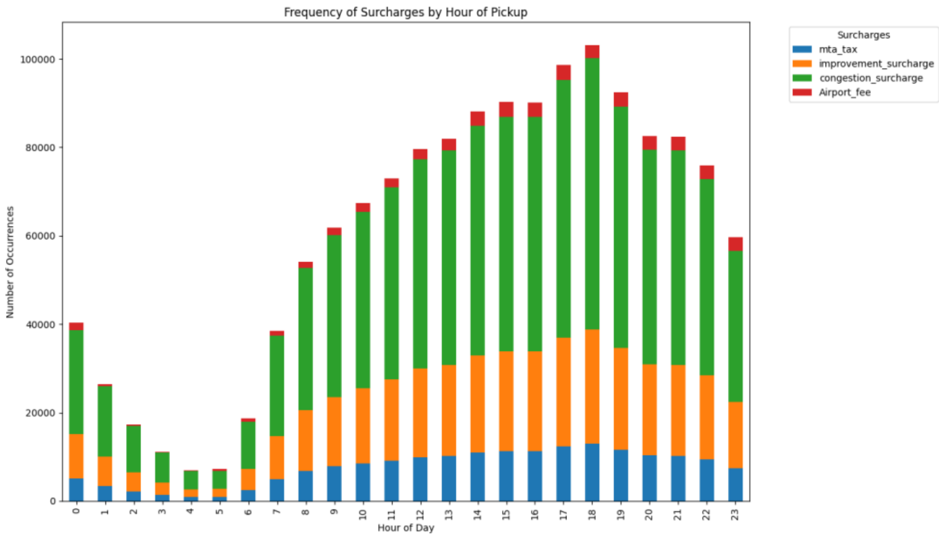


3.2.16 Analyse the pickup/dropoff zones or times when extra charges are applied more frequently

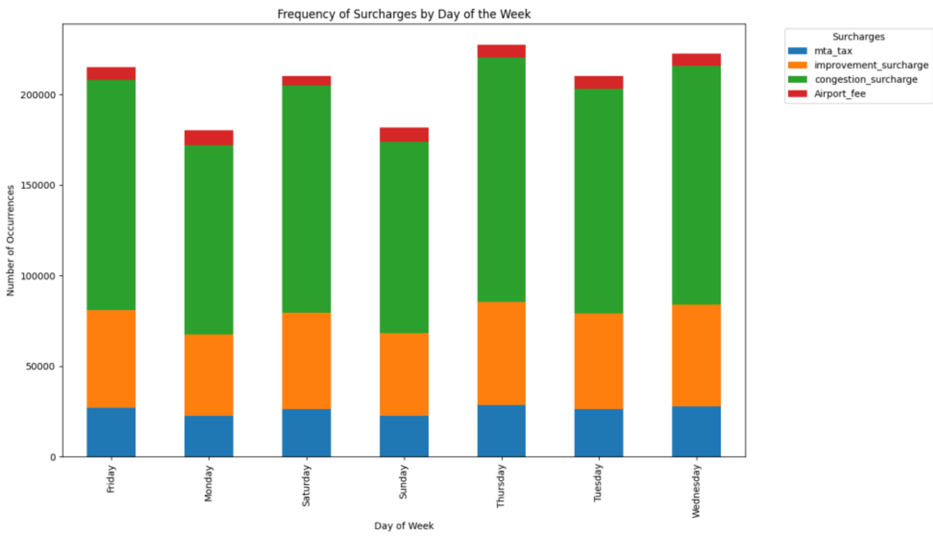
Below plot gives a detailed view all the zones and the frequency of surcharges applied in those zones.



Below chart shows the frequency of surcharges by hour of pickup.



Below chart shows the frequency of surcharges by day of the week.



4 Findings and Conclusions

4.1 Final Insights and Recommendations

The NYC Yellow Cab dataset provides a wealth of information that, when properly analysed, can lead to actionable strategies for improving operational efficiency, optimizing fleet positioning, and enhancing pricing models.

Based on our exploratory data analysis and supported by insights from our analysis, the following conclusions are drawn:

4.1.1 Recommendations to optimize routing and dispatching based on demand patterns and operational inefficiencies

Our analysis revealed that:

- **Morning demand peaks between 7 AM–9 AM**, on weekdays and **evening peaks around 6 PM**, with the highest concentration of pickups in business and transit-heavy areas such as Midtown Manhattan and JFK Airport.
- **Average trip speeds** drop significantly in certain zone-to-zone routes during peak hours. For example, trips from Zone 132 to Zone 236 at 8 AM average just **4.0 mph**, indicating heavy congestion and possible route inefficiencies.
- **Idle Time:** Some zones exhibit repeated low trip frequencies during certain hours, indicating underutilization of resources.
- Some outliers in the data showed trip speeds exceeding **100 mph**, which were identified and removed as likely timestamp anomalies.

Recommendation:

- Implement time-aware routing: Use historical traffic and trip duration data to recommend the most efficient zone-to-zone routes at different times of day.
- Dynamic dispatching: Allocate drivers toward high-yield routes like JFK–Manhattan during peak hours and away from zones with congestion bottlenecks.
- Stagger driver shifts: Ensure higher cab availability before expected spikes (e.g., rush hours), possibly using predictive analysis.
- Integrate real-time traffic + trip data into dispatch apps for adaptive routing.
- Tailoring dispatch and positioning strategies based on both the day of the week and time of day can increase trip volume and improve customer satisfaction. For instance, weekend evenings see a spike in trips likely due to social or leisure activities, where passengers are more relaxed and tend to tip better. In contrast, weekday morning commuters are less likely to tip, suggesting that ride experience optimization (like minimal wait times or efficient routing) should be prioritized during those hours.

4.1.2 Suggestions on strategically positioning cabs across different zones to make best use of insights uncovered by analysing trip trends across time, days and months

By combining hourly, daily, and zone-level pickup patterns, we observed:

- **Weekday rush hours** see demand concentrated in commercial hubs, whereas **weekend evenings** shift demand toward nightlife-heavy zones such as East Village and Williamsburg.
- Our **heatmap analysis** showed that Monday to Friday mornings in residential zones had consistently higher average passenger counts, while weekend nights showed higher group travel activity.
- **Zone Imbalances:** Certain zones repeatedly show high drop-off but low pickup volume (e.g., upper residential zones), suggesting poor repositioning.
- Passenger count trends also suggest a need for more **larger-capacity vehicles** (e.g., minivans) in airport and nightlife zones where group travel is common.

Recommendation:

- Develop a zone-time positioning grid: Place more taxis in nightlife areas post 8 PM on weekends and in commercial hubs during weekday mornings.
- Encourage repositioning after drop-offs: Use incentives to nudge drivers to shift toward high-pickup zones instead of waiting in low-demand areas.
- Deploy larger-capacity vehicles in zones with higher average passenger counts (e.g., airport terminals, nightlife areas).
- Display heatmaps to drivers through an app interface showing real-time and forecasted pickup zones.

4.1.3 Propose data-driven adjustments to the pricing strategy to maximize revenue while maintaining competitive rates with other vendors

Fare and tip-related findings from our data include:

- **Short trips (≤2 miles)** had the highest **fare per mile**, often exceeding $5/mile, while longer trips showed significantly lower unit costs.
- **Tip percentages were highest (avg. ~18%)** during mid-range trips and weekday evenings, and were notably lower for trips with passenger_count = 1 and for very short distances.
- Outliers such as **$300+ fare on sub-0.1 mile trips** were removed as anomalies.

- **Payment type 0** (undefined) was fixed using tip amount logic, records with tips were correctly reclassified as **credit card payments**.

Recommendation:

- Adopt a tiered fare model that offers incentives for mid-range rides while maintaining base profitability on short trips.
- Consider implementing tip-based loyalty bonuses for high-tipping customers or informative prompts for tipping ranges during card payments, similar to Uber's dynamic tipping model.
- Integrate time-based pricing strategies, offering off-peak discounts during underused hours to drive more traffic.

Through careful sampling, cleaning, and analysis of patterns across time, location, and rider behaviour, we found that the NYC taxi data holds valuable insights for real-world decision-making. When combined with broader industry findings, this data can help improve routing, pricing, and overall taxi operations in a smart and practical way.