Name: Hemesh Sawakar (C74)

Program: upGrad and IIITB Machine Learning & AI Program

Course: SQL and Statistics Essentials

# Report: Optimizing NYC Taxi Operations

Include your visualizations, analysis, results, insights, and outcomes. Explain your methodology and approach to the tasks. Add your conclusions to the sections.

## 1. Data Preparation

### 1.1. Loading the dataset

We started by suppressing warnings and importing the python libraries (numpy,pandas,matplotlib,seaborn). We ensured their versions are latest as recommended.

We read one month file and learned that there are around 3 million records which brought us to conclusion that it is huge and infeasible to computationally process for 12-month files. Hence, we need to sample the fractions of data before combining all months in a single file.

#### 1.1.1. Sample the data and combine the files

One way is to take a small percentage of entries for pickup in every hour of a date. So, for all the days in a month, we are iterating through the hours and selecting 5% values randomly from those. We used tpep_pickup_datetime for this. The date and hours were separated from the datetime values and then for each date, 5% are sampled from each hour. These samples are combined to empty dataframe at initialization and after each iteration we appended the sample to the dataframe.

Processing and combining 12 months data we got final size of data having around 1.9 million records and 22 columns. We exported the data into a parquet file for easy reading next time and performing further operations ahead.

# 2. Data Cleaning

## 2.1. Fixing Columns

### 2.1.1. Fix the index

The total records are 1896400 and indexes range from 0 to 1896399. So indexes are fine. But, as part of reading the data we created 2 extra columns (date,hour) which we have dropped in this step as they were unnecessary.

### 2.1.2. Combine the two airport_fee columns

Due to different naming convention two columns got created for airport fare (airport_fee, Airport_fee). We are combining them by taking their sum after replacing any null values in them. Final column is ‘airport_fees’ and we have dropped the original columns.

### 2.1.3. Fix columns with negative (monetary) values

There are no negative fare amounts nor RatecodeID. But we found

total_amount has negative values. It also revealed that mta_tax, improvement_surcharge, congestio_surcharge, airport_fees also negative values.

careful observation on other columns shows that column 'extra' also has some negative values.

tolls_amount and tip_amount do not have any negative values

We applied absolute function on

'extra', 'mta_tax', 'improvement_surcharge', 'congestion_surcharge', 'airport_fees', 'total_amount'

to turn negative values into positive

## 2.2. Handling Missing Values

### 2.2.1. Find the proportion of missing values in each column

passenger counts, RatecodeID, store_and_fwd_flag, congesion_surcharge each have 64,874 null values which is 3.5% of total records 1,896,400

### 2.2.2. Handling missing values in passenger_count

The missing values in passenger_count are replaced with median value in that column which is 1. We used median instead of mean because there could be extreme values in the column which may skew the analysis. Also we found 0 passenger_count which does not make sense from a point that fare collected is non zero but no passengers commuted in taxi. Hence we also replaced them with median.

### 2.2.3. Handle missing values in RatecodeID

since the RatecodeID is a catagorical column it is general practice to replace nulls in catagorical column with mode values and numeric values wih mean/median

### 2.2.4. Impute NaN in congestion_surcharge

Congestion_surcharge was imputed with median values

## 2.3. Handling Outliers and Standardising Values

### 2.3.1. Check outliers in payment type, trip distance and tip amount columns

summary statistics run on data show that

passenger max count of 9 seems unreal as taxis are 5 seater if sedan and 7 seater if SUVs

trip_distance, total_amount feel like outlier from difference in their max and 75th percentile

and also starter notebook guides to check

1. Entries where trip_distance is nearly 0 and fare_amount is more than 300
2. Entries where trip_distance and fare_amount are 0 but the pickup and dropoff zones are different (both distance and fare should not be zero for different zones)
3. Entries where trip_distance is more than 250 miles.
4. Entries where payment_type is 0 (there is no payment_type 0 defined in the data dictionary) (This is invalid because latest dictionary as of 2025 says payment type 0 is Flex fare trip)

We have removed the outlier cases from first 3 points and passenger count>6.

For obtaining clean and better dataset, we have run Interquartile range method to remove the outliers from all the fact (numeric) columns

['fare_amount','extra', 'mta_tax', 'tip_amount','tolls_amount', 'improvement_surcharge', 'total_amount','congestion_surcharge', 'airport_fees'].

From initial 1,896,399 records we are down 389,653 records at 1,506,747

We have also standardized the passenger_count and RatecodeID from float data type to integer

# 3. Exploratory Data Analysis

## 3.1. General EDA: Finding Patterns and Trends

### 3.1.1. Classify variables into categorical and numerical

catagorical=['VendorID','RatecodeID','store_and_fwd_flag','payment_type]

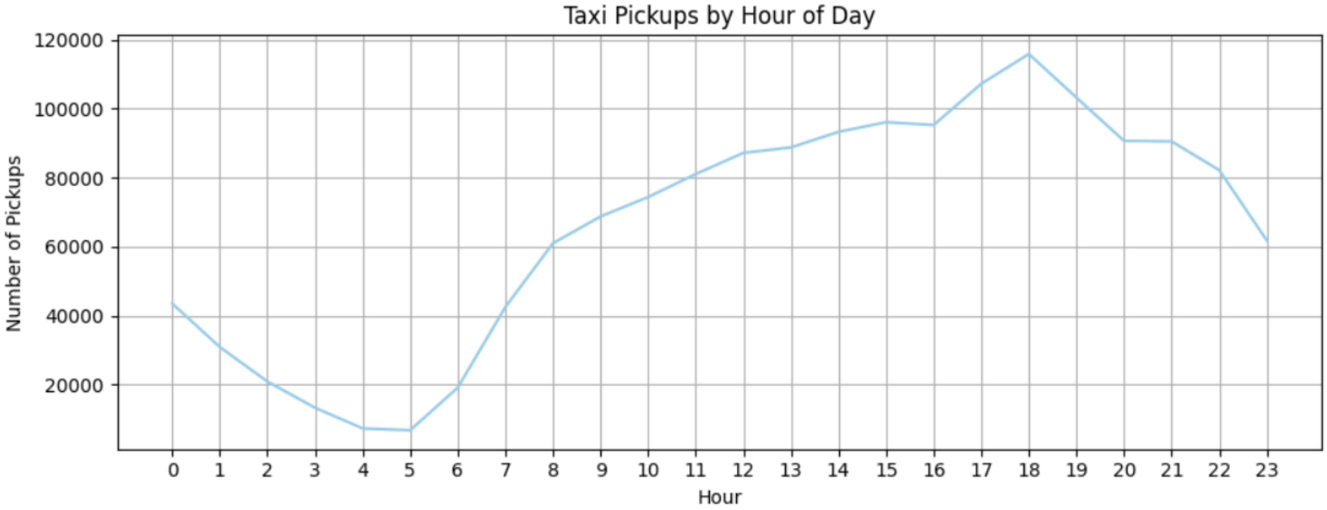
numerical= [ 'tpep_pickup_datetime', 'tpep_dropoff_datetime',

'PULocationID', 'DOLocationID', 'fare_amount', 'extra', 'mta_tax', 'tip_amount', 'tolls_amount', 'improvement_surcharge', 'total_amount', 'congestion_surcharge', 'airport_fees']

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

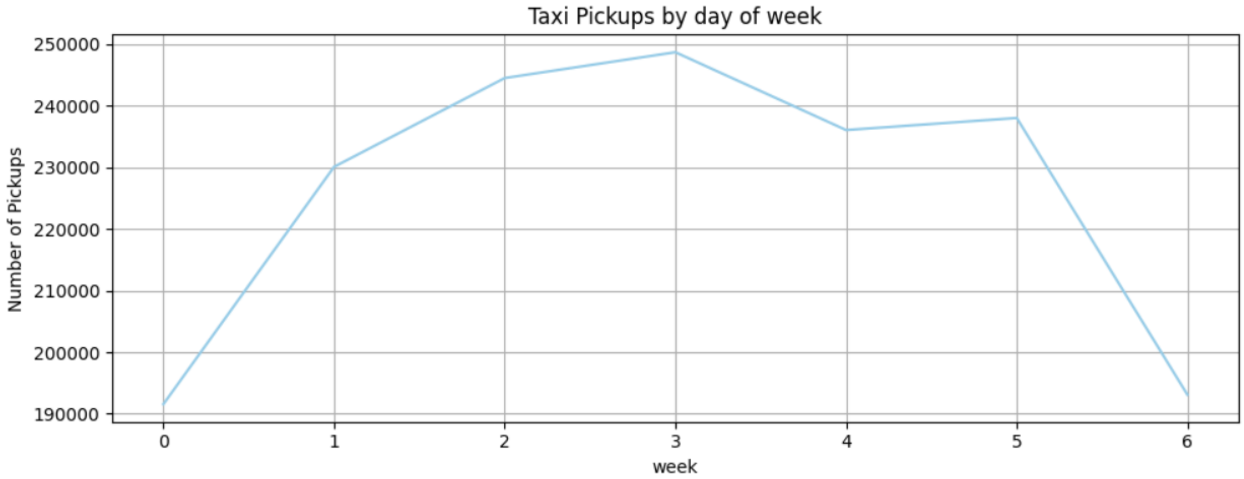
The taxi pickup are generally high from 8 am to 11pm.

Peak business hours are between 11 am to 9pm surging to the top at 6pm.



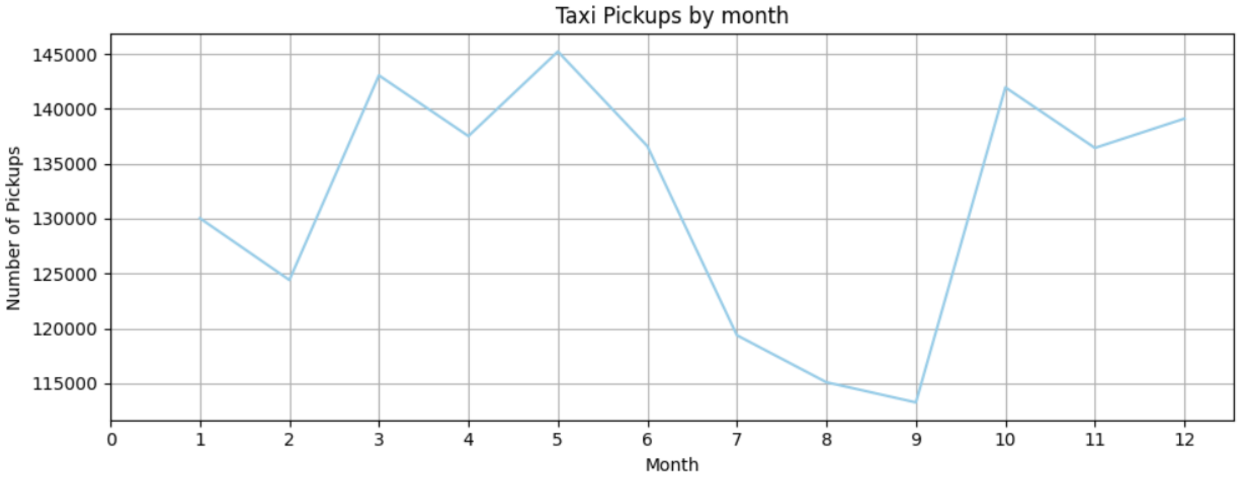
The taxi pickups are lowest on Sunday followed by Monday. Taxi pickup are trending high from Tuesday to Friday.

Wed, Thu, are almost identical showing highest business activity.



The taxi pickups are high during Mar,Apr,May,June and Oct,Nov,Dec. In May it was at peak.

July, Aug, Sep is where it is lower than the trend. At September it is lowest then increases in Oct again



### 3.1.3. Filter out the zero/negative values in fares, distance and tips

We had removed the negative values using abs() function so there are no negative values. We can confirm with summary statistics

There are 0 values in fare_amount, tip_amount and trip_distance

zero_fare_amount_count:  48

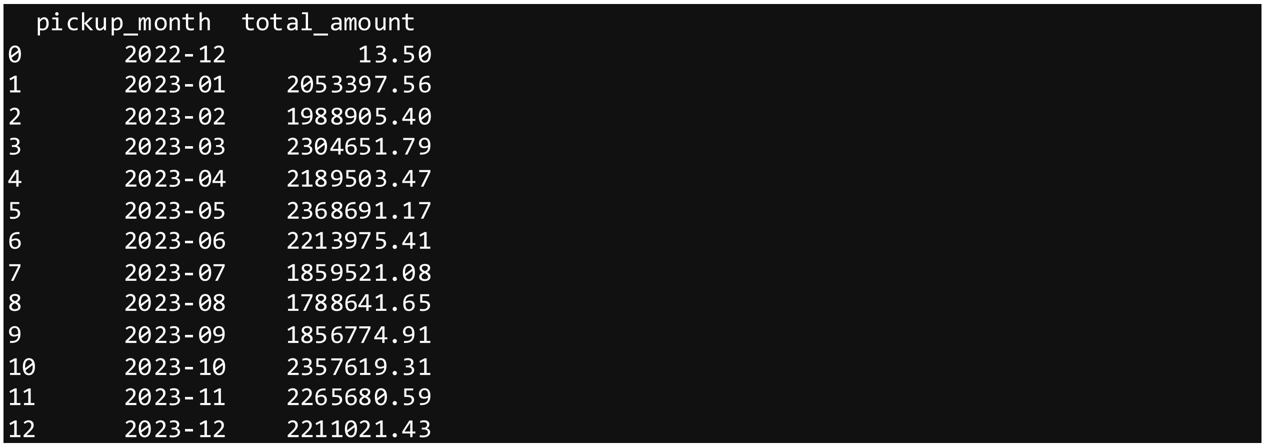
zero_tip_amount_count: 315157

zero_trip_distance_count: 3783

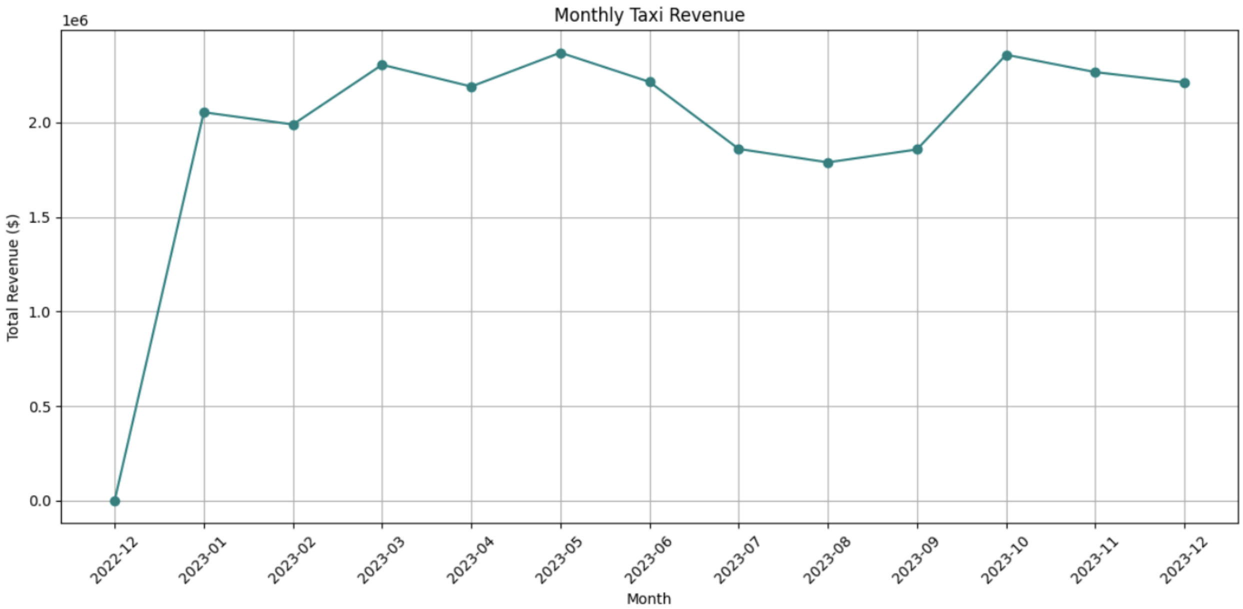
### 3.1.4. Analyse the monthly revenue trends

monthly revenue shows fluctuating trend but Mar'23 to May'23 was a good revenue grossing period.

It dropped during July to September but again came to peak high in Oct'23.

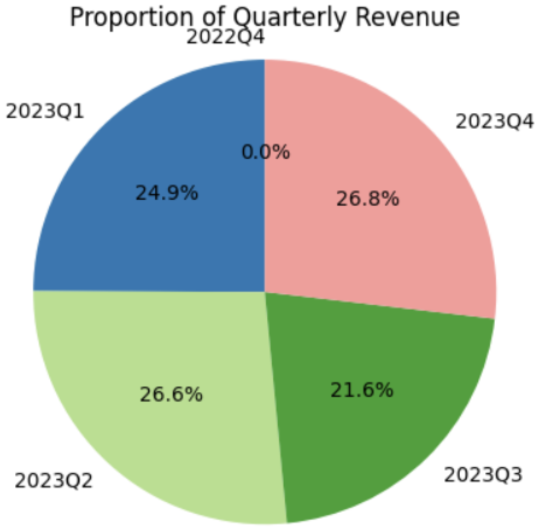


```
  pickup_month  total_amount
0       2022-12         13.50
1       2023-01    2053397.56
2       2023-02    1988905.40
3       2023-03    2304651.79
4       2023-04    2189503.47
5       2023-05    2368691.17
6       2023-06    2213975.41
7       2023-07    1859521.08
8       2023-08    1788641.65
9       2023-09    1856774.91
10      2023-10    2357619.31
11      2023-11    2265680.59
12      2023-12    2211021.43
```



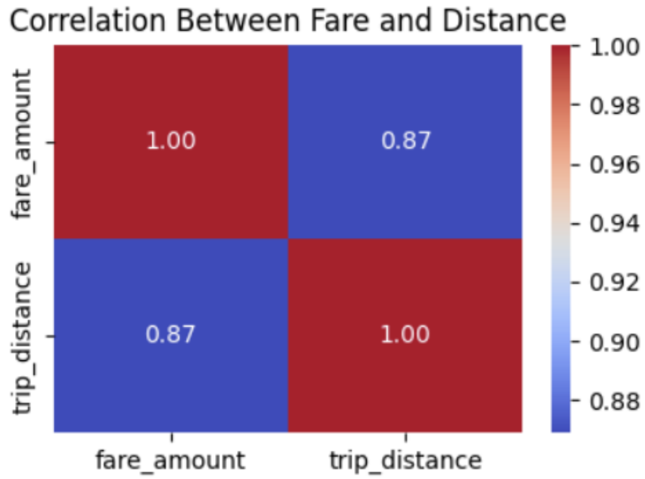
### 3.1.5. Find the proportion of each quarter’s revenue in the yearly revenue

In all of the quarters the revenue earned is proportional. 2nd and 4th quarters are highest revenue earned quarters.



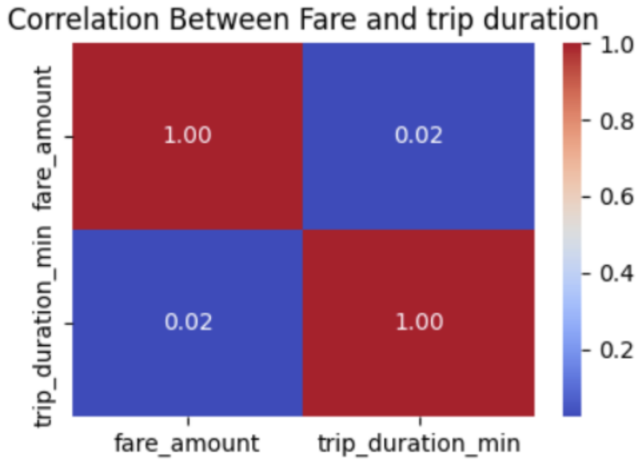
### 3.1.6. Analyse and visualise the relationship between distance and fare amount

We can see fare_amount and trip_distnace are positively very correlated. So, when trip_distance increases fare_amount increases.

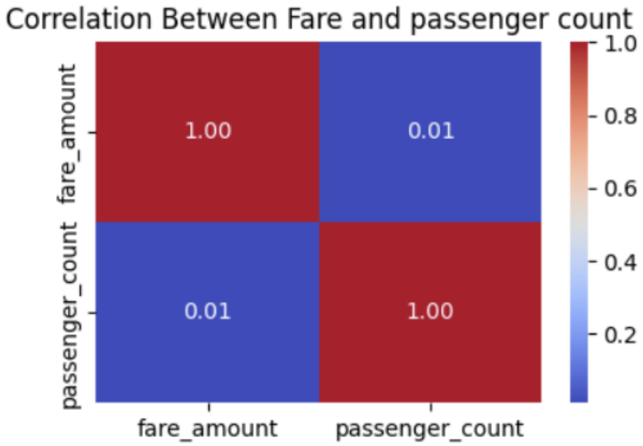


### 3.1.7. Analyse the relationship between fare/tips and trips/passengers

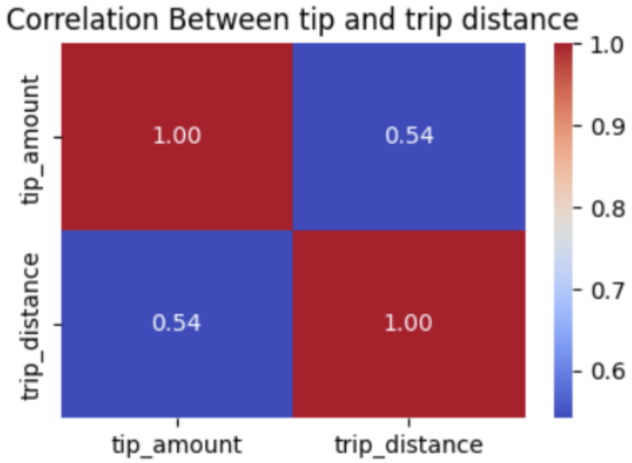
There is no correlation between taxi fare and trip duration (in minutes)



There is no correlation between fare and passenger count

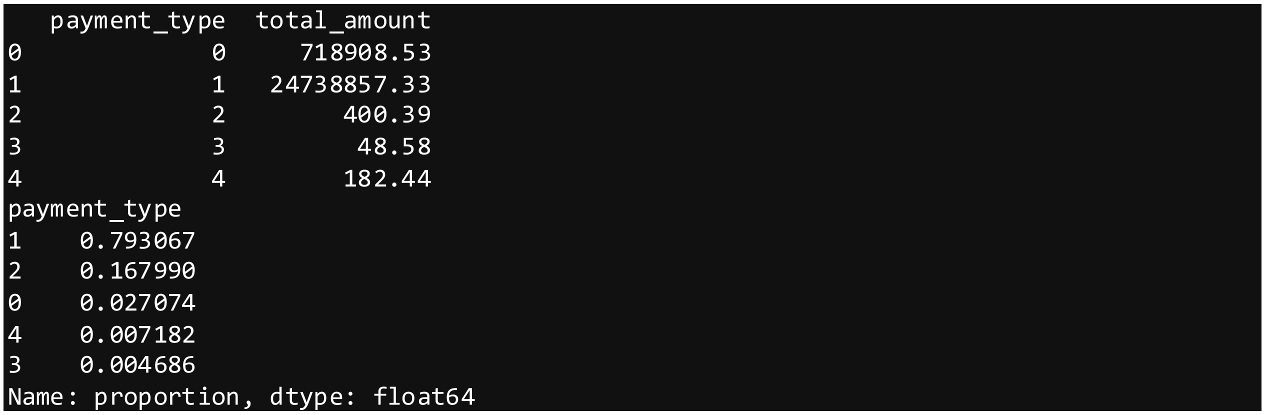


But there is mild positive correlation between tip amount and trip distance

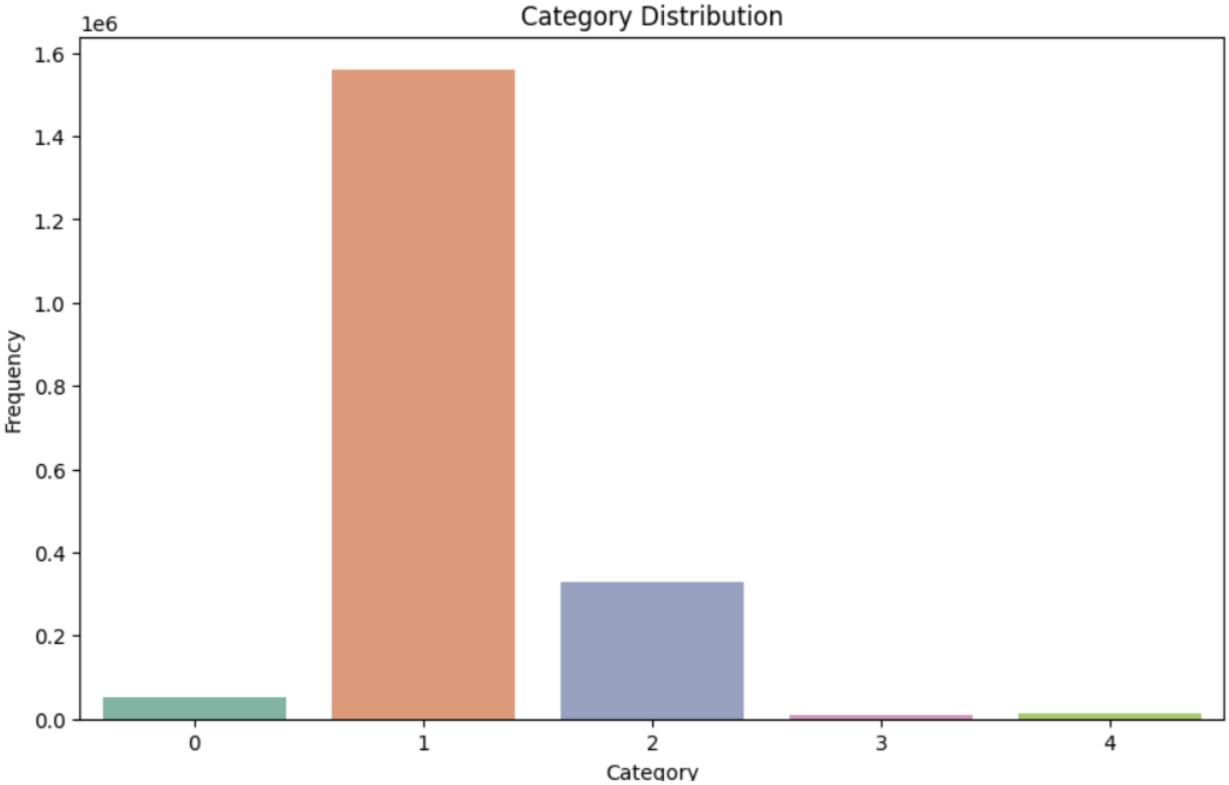


### 3.1.8. Analyse the distribution of different payment types

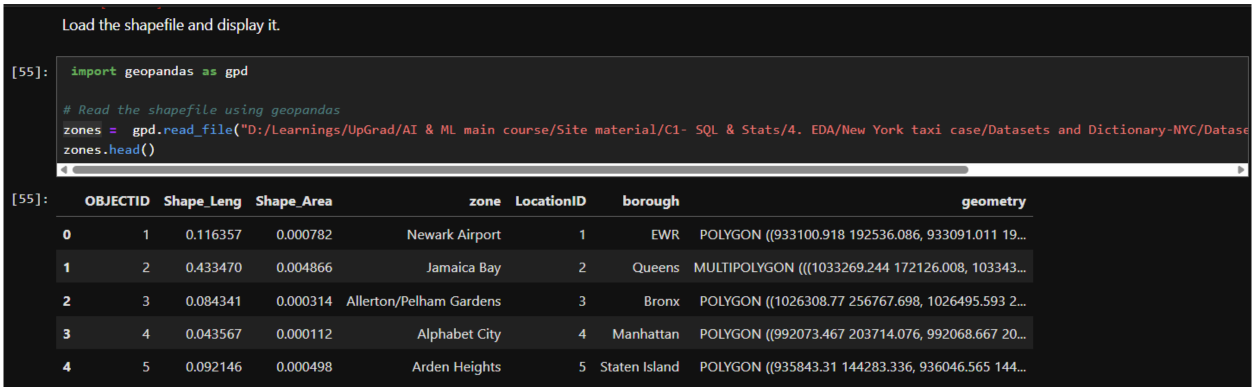
Nearly 80% payments are by credit card and 17% are by cash.



```
   payment_type  total_amount
0             0     718908.53
1             1   24738857.33
2             2        400.39
3             3         48.58
4             4        182.44
payment_type
1    0.793067
2    0.167990
0    0.027074
4    0.007182
3    0.004686
Name: proportion, dtype: float64
```



### 3.1.9 Load the taxi zones shapefile and display it



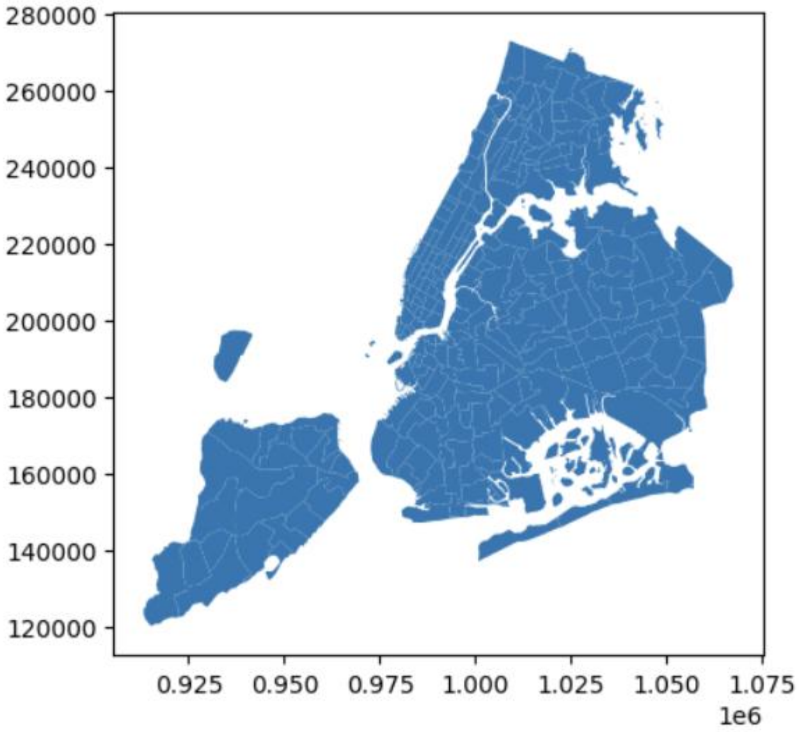
Load the shapefile and display it.

[55]:
```
import geopandas as gpd

# Read the shapefile using geopandas
zones =  gpd.read_file("D:/Learnings/UpGrad/AI & ML main course/Site material/C1- SQL & Stats/4. EDA/New York taxi case/Datasets and Dictionary-NYC/Datase
zones.head()
```

[55]:

| | 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... |

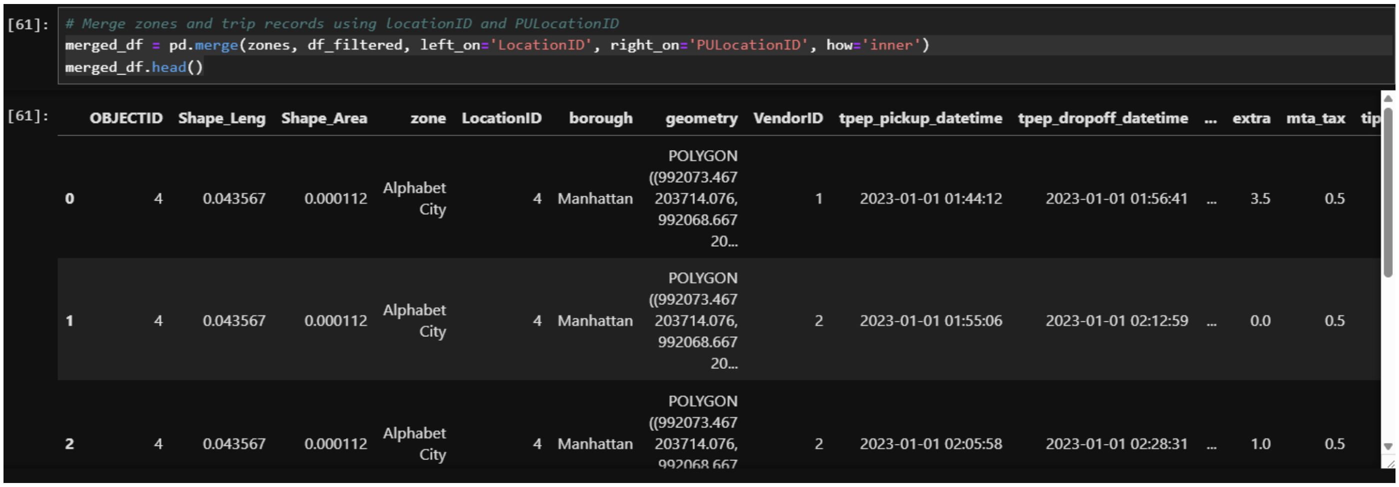


### 3.1.10 Merge the zone data with trips data

We merged zones and trip recods dataframes on LocationID and PULocationID with an inner join.

merged_df = pd.merge(zones, df_filtered, left_on='LocationID', right_on='PULocationID', how='inner')

merged_df.head()



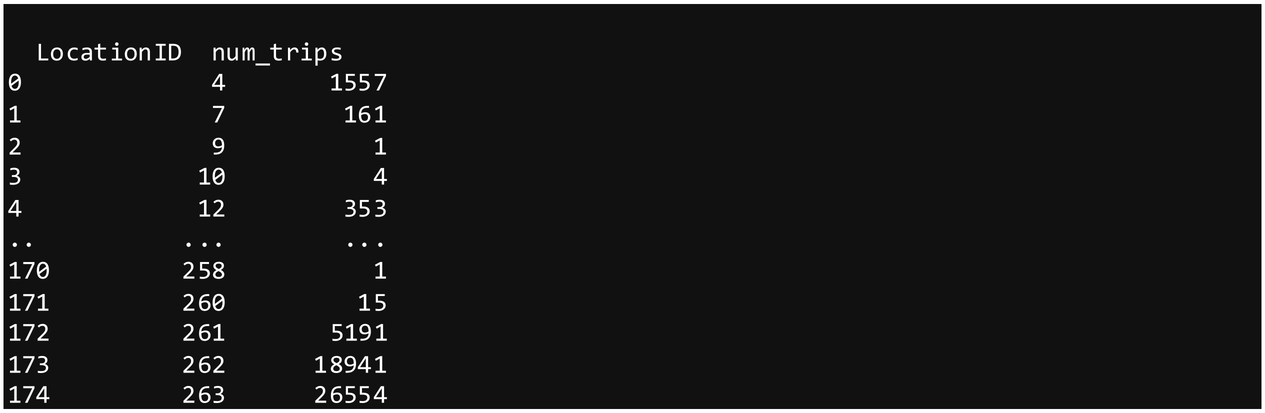
```
[61]: # Merge zones and trip records using locationID and PULocationID
merged_df = pd.merge(zones, df_filtered, left_on='LocationID', right_on='PULocationID', how='inner')
merged_df.head()
```

[61]:

| | OBJECTID | Shape_Leng | Shape_Area | zone | LocationID | borough | geometry | VendorID | tpep_pickup_datetime | tpep_dropoff_datetime | ... | extra | mta_tax | tip |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 4 | 0.043567 | 0.000112 | Alphabet City | 4 | Manhattan | POLYGON ((992073.467 203714.076, 992068.667 20... | 1 | 2023-01-01 01:44:12 | 2023-01-01 01:56:41 | ... | 3.5 | 0.5 | |
| **1** | 4 | 0.043567 | 0.000112 | Alphabet City | 4 | Manhattan | POLYGON ((992073.467 203714.076, 992068.667 20... | 2 | 2023-01-01 01:55:06 | 2023-01-01 02:12:59 | ... | 0.0 | 0.5 | |
| **2** | 4 | 0.043567 | 0.000112 | Alphabet City | 4 | Manhattan | POLYGON ((992073.467 203714.076, 992068.667 | 2 | 2023-01-01 02:05:58 | 2023-01-01 02:28:31 | ... | 1.0 | 0.5 | |

### 3.1.11 Find the number of trips for each zone/location ID

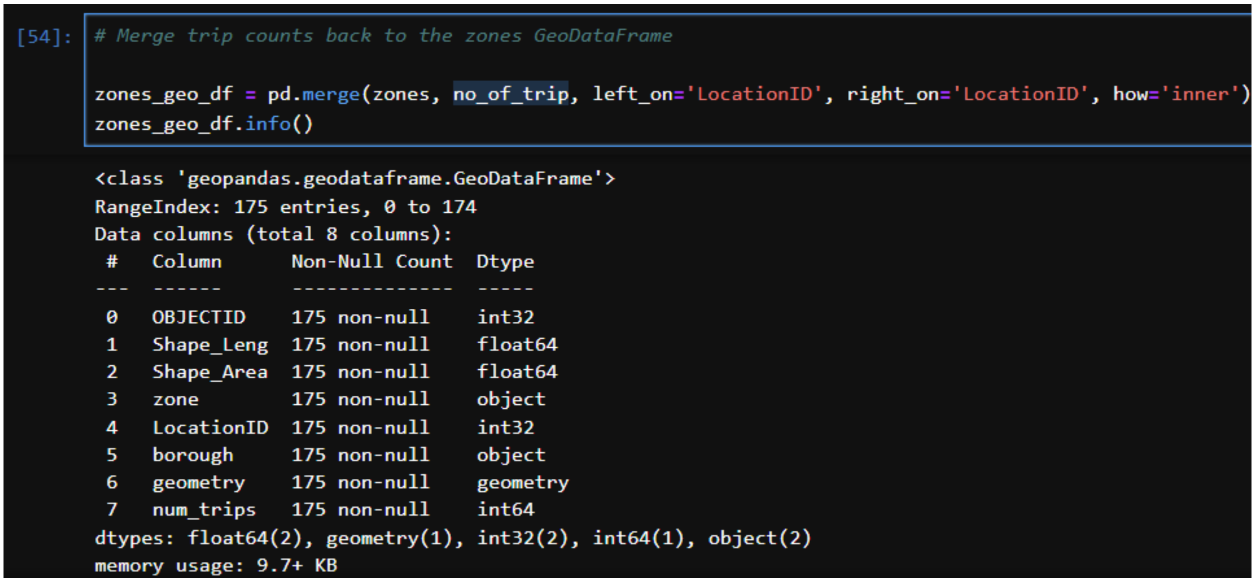
We grouped by LocationID and count number of trips. We store it in a variable and display.



```
  LocationID  num_trips
0           4       1557
1           7        161
2           9          1
3          10          4
4          12        353
..        ...        ...
170       258          1
171       260         15
172       261       5191
173       262      18941
174       263      26554
```

### 3.1.12 Add the number of trips for each zone to the zones dataframe

Here merged the zones dataframe with no_of_trips dataframe on ‘LocationID’ with inner join.

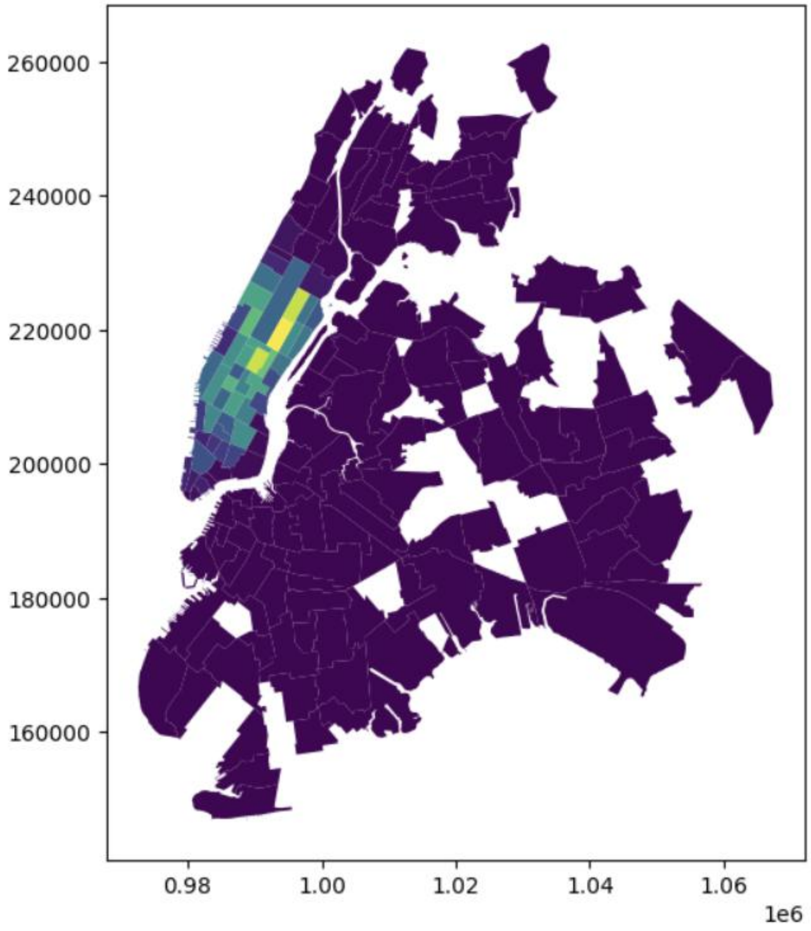


```
[54]: # Merge trip counts back to the zones GeoDataFrame

      zones_geo_df = pd.merge(zones, no_of_trip, left_on='LocationID', right_on='LocationID', how='inner')
      zones_geo_df.info()
```

```
<class 'geopandas.geodataframe.GeoDataFrame'>
RangeIndex: 175 entries, 0 to 174
Data columns (total 8 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   OBJECTID    175 non-null    int32
 1   Shape_Leng  175 non-null    float64
 2   Shape_Area  175 non-null    float64
 3   zone        175 non-null    object
 4   LocationID  175 non-null    int32
 5   borough     175 non-null    object
 6   geometry    175 non-null    geometry
 7   num_trips   175 non-null    int64
dtypes: float64(2), geometry(1), int32(2), int64(1), object(2)
memory usage: 9.7+ KB
```

### 3.1.13 Plot a map of the zones showing number of trips



### 3.1.14 Conclude with results

The greater number of trips are in the **Manhattan** region of New York.

In **Manhattan** Highest also they are greatest in the **Upper East Side South, Midtown Center, Upper East Side North zones**.

**Busy business hours** are 11am to 10 pm with **6pm** being the **busiest**.

**Busiest day** is **Thursday**.

**Busiest month** is **May**.

**Quarterly revenues** are almost identical for all quarters but **2nd** and **4th quarter** top the chart.

We found **strong correlation** between **distance and fare amount**.

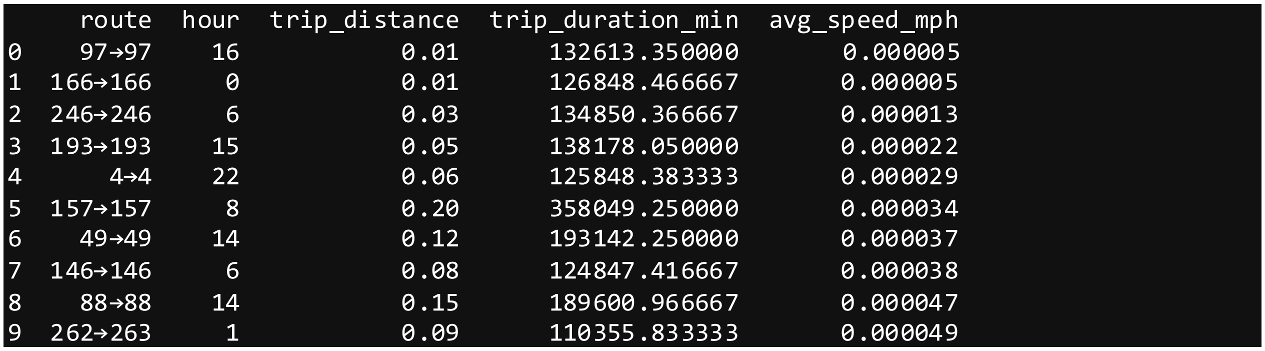
**Descent correlation** between **tip and trip distance**.

And **low to almost no correlation** for **fare-trip duration**, **fare-passenger count**.

## **3.2** Detailed EDA: Insights and Strategies

### 3.2.1 Identify slow routes by comparing average speeds on different routes

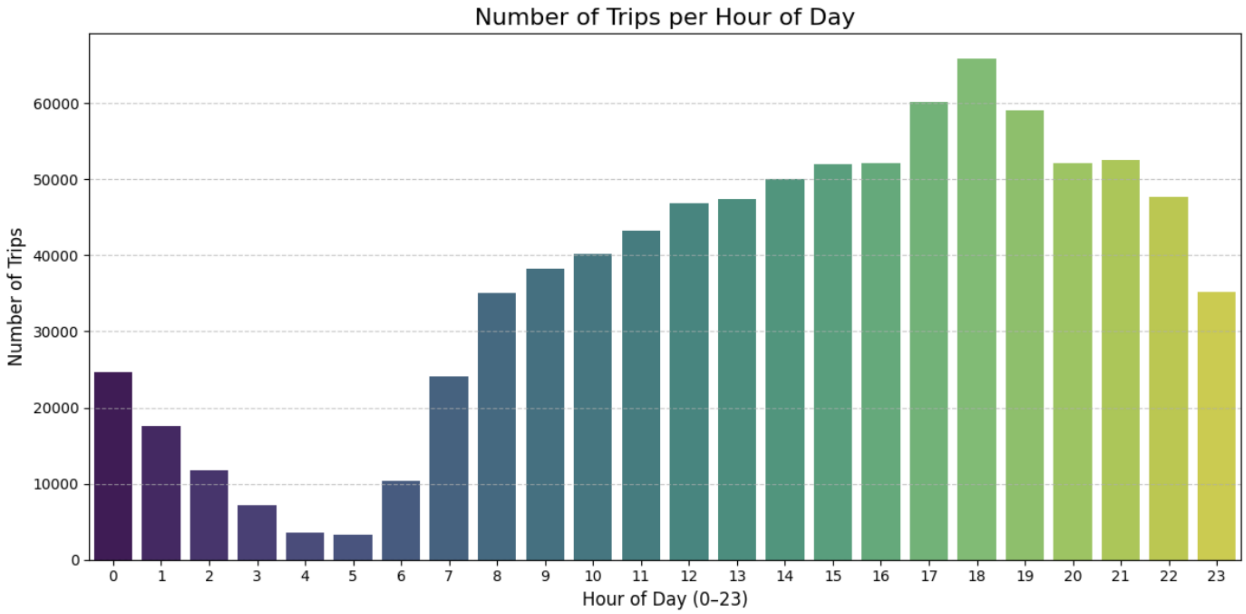
Below are top 10 slow routes**.** These seem absurd, must be error in capturing the pickup/drop time by car system or gps.



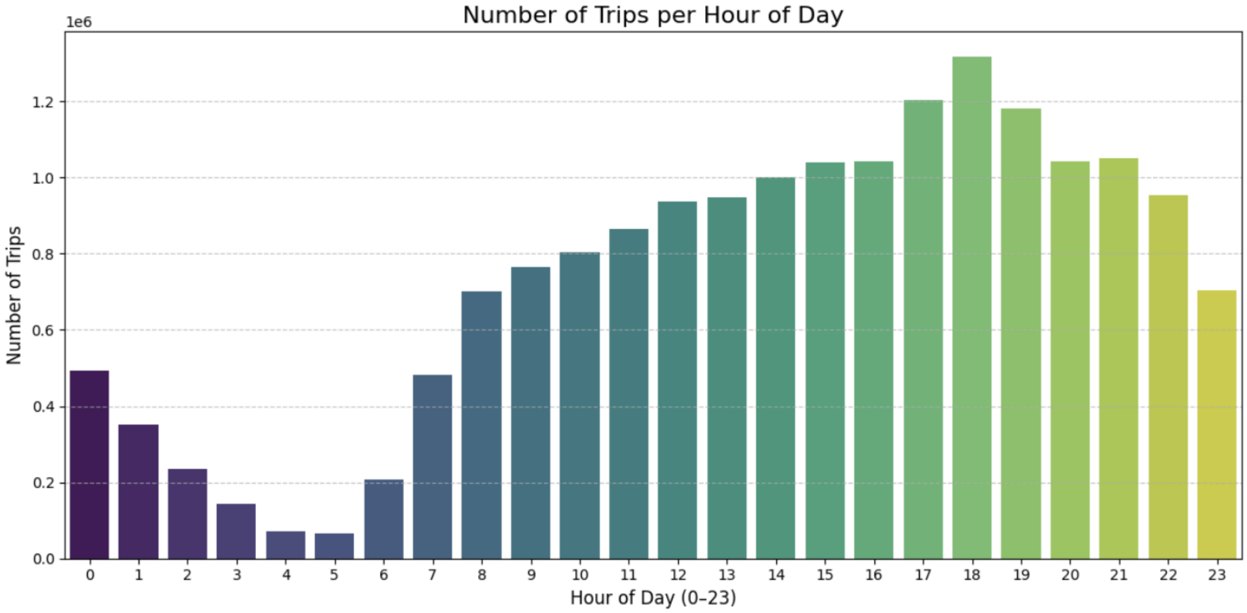
```
     route  hour  trip_distance  trip_duration_min  avg_speed_mph
0    97→97    16           0.01      132613.350000       0.000005
1  166→166     0           0.01      126848.466667       0.000005
2  246→246     6           0.03      134850.366667       0.000013
3  193→193    15           0.05      138178.050000       0.000022
4      4→4    22           0.06      125848.383333       0.000029
5  157→157     8           0.20      358049.250000       0.000034
6    49→49    14           0.12      193142.250000       0.000037
7  146→146     6           0.08      124847.416667       0.000038
8    88→88    14           0.15      189600.966667       0.000047
9  262→263     1           0.09      110355.833333       0.000049
```

### 3.2.2 Calculate the hourly number of trips and identify the busy hours

The number of trips are highest at 6pm also high at 5 and 7 but in general high after the afternoon

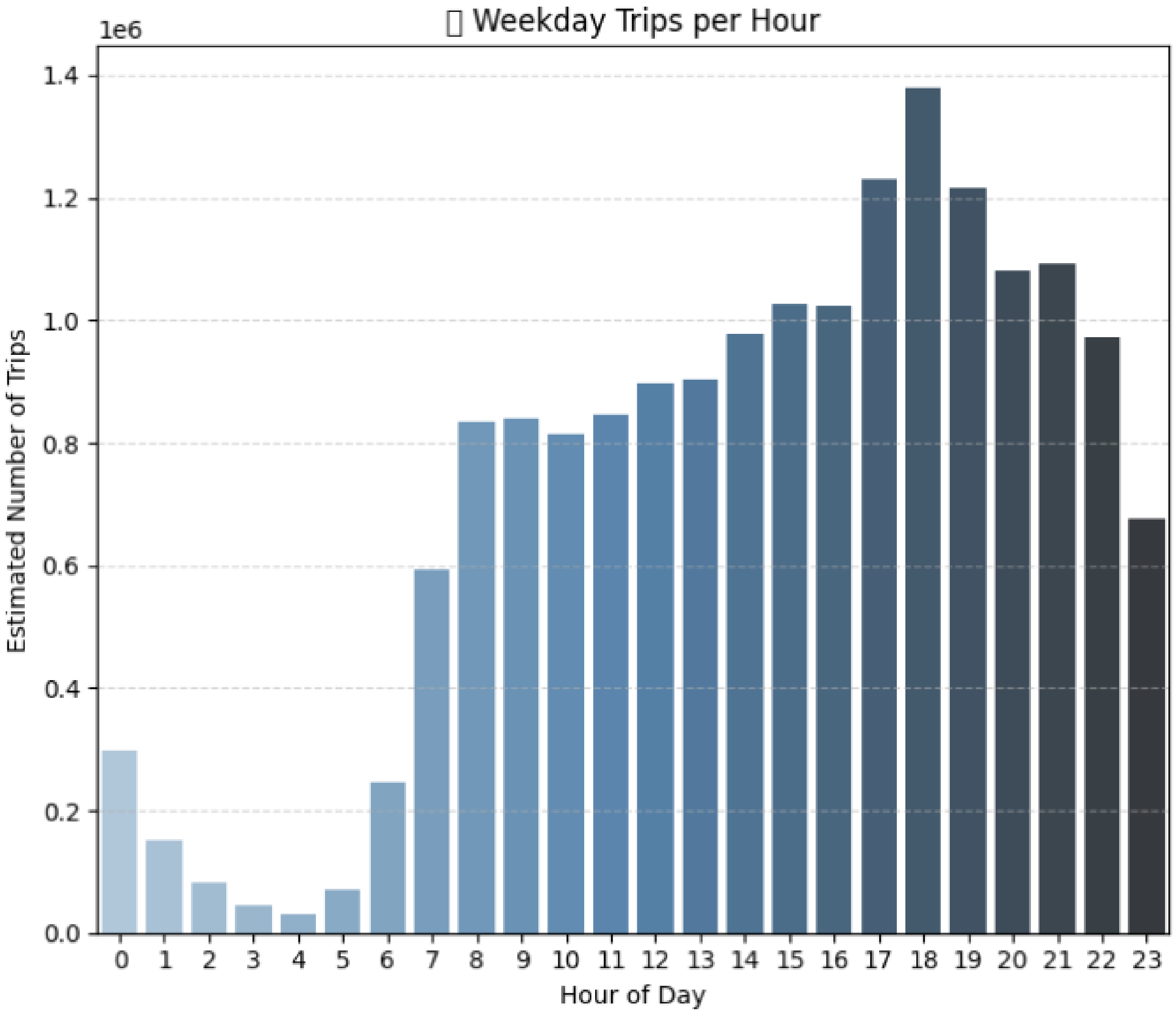


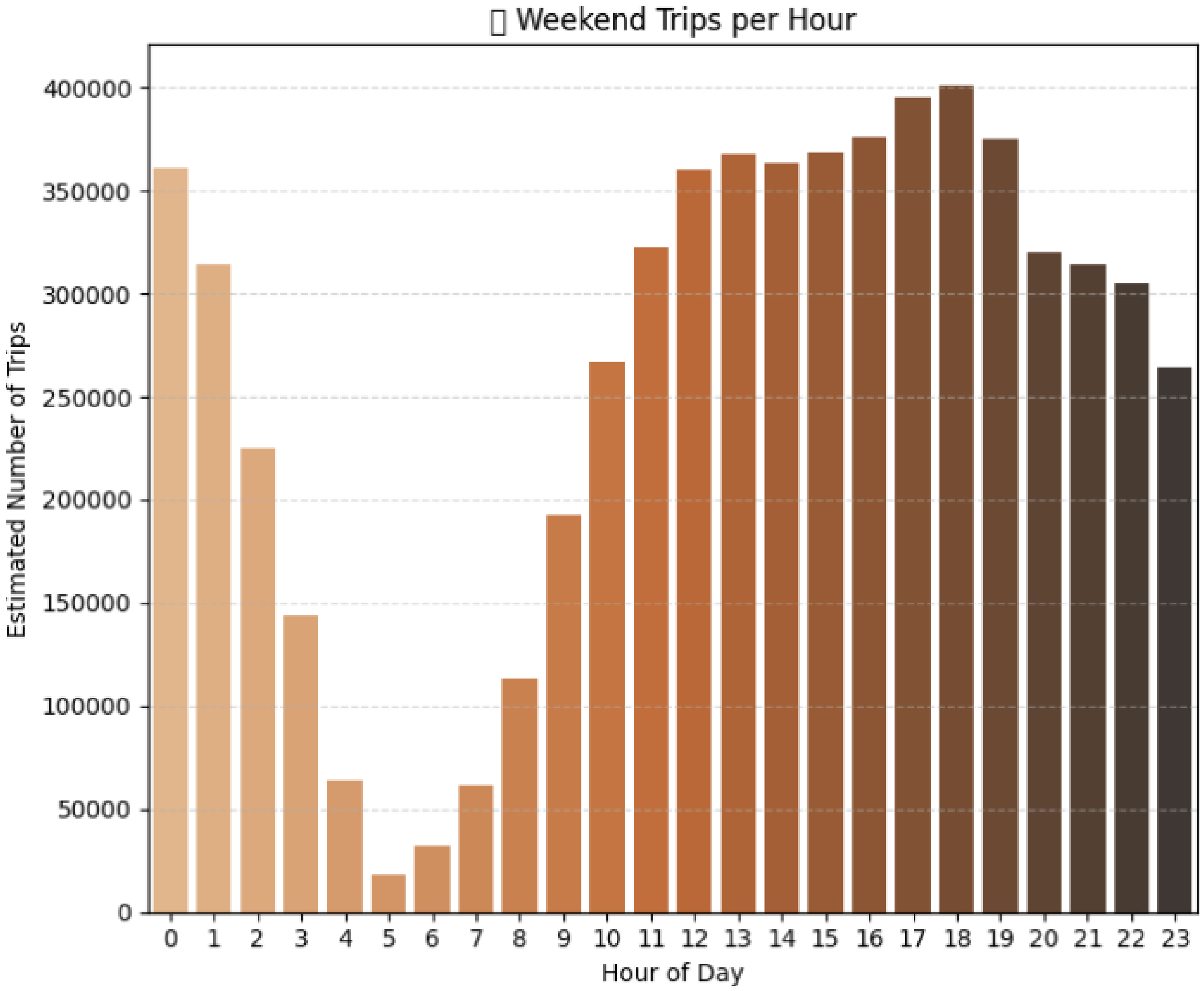
### 3.2.3 Scale up the number of trips from above to find the actual number of trips



### 3.2.4 Compare hourly traffic on weekdays and weekends

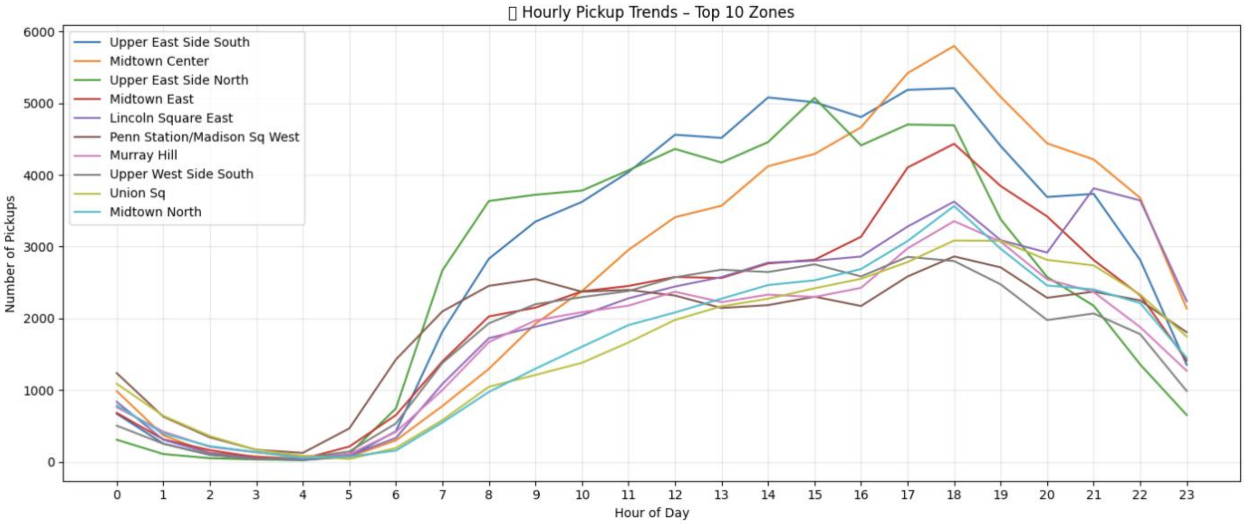
- Weekday trips are higher than weekend trips overall.
- Both show similar pattern for identical hours of day exception being on weekend between 12am to 4am also we can see higher trip counts compared to weekdays. The reason could be people relaxing and partying on weekends so staying up late and commuting to clubs, returning homes etc.
- On weekdays people staring their day at 6am to 8 am are higher than weekends. This might be that they want to start late so as to take a little rest from the week days work.


□ Weekday Trips per Hour
1e6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Estimated Number of Trips
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
Hour of Day

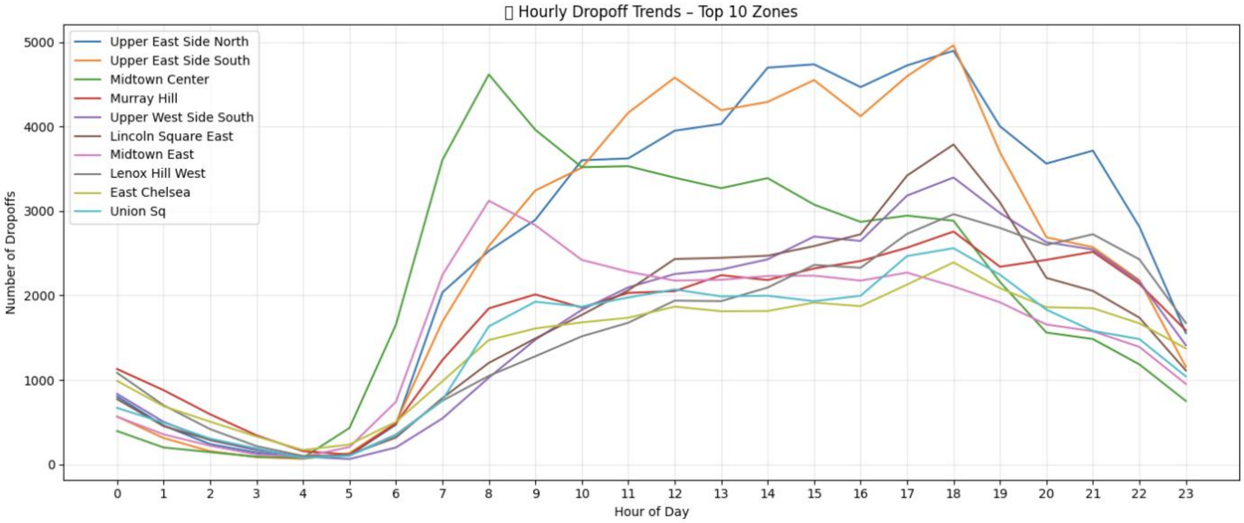

□ Weekend Trips per Hour
Estimated Number of Trips
400000
350000
300000
250000
200000
150000
100000
50000
0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
Hour of Day

### 3.2.5 Identify the top 10 zones with high hourly pickups and drops

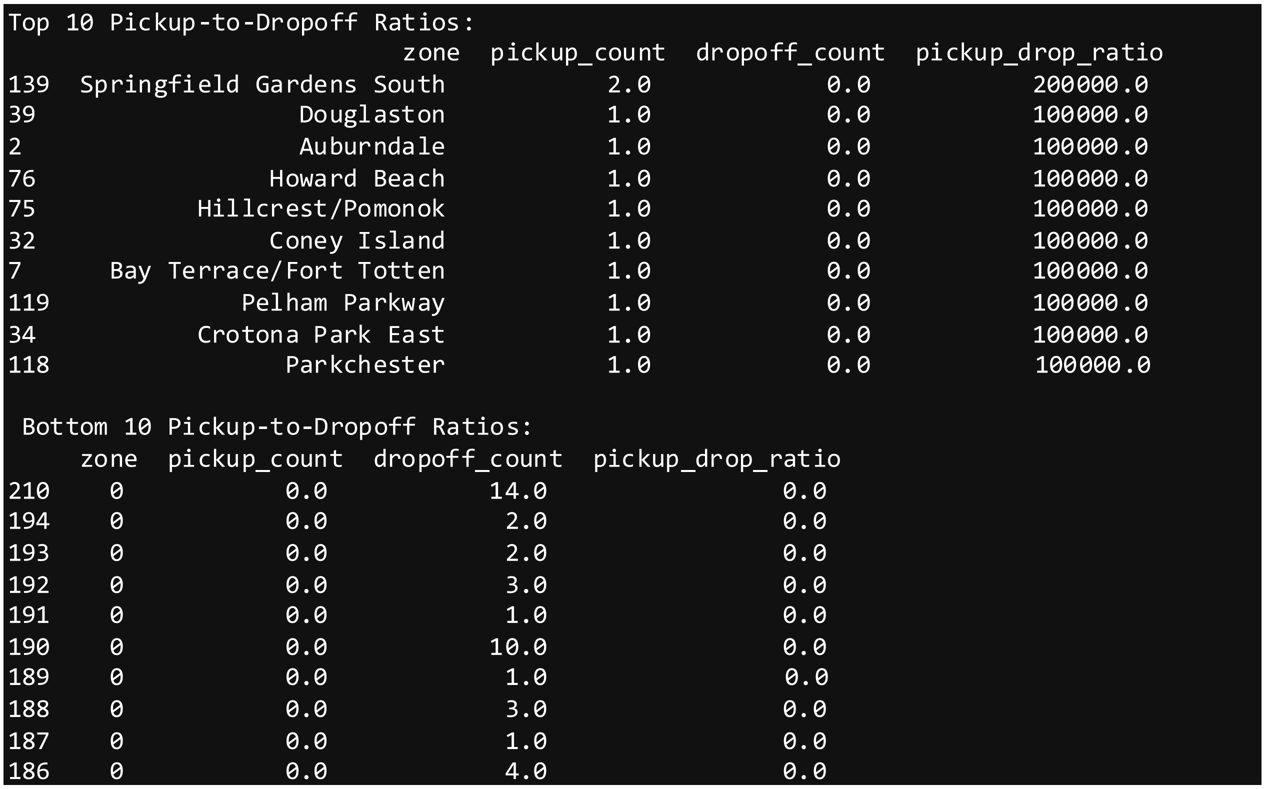
Top 10 zones with high hourly pickup



Top 10 zones with high hourly dropoffs



### 3.2.6 Find the ratio of pickups and dropoffs in each zone



```
Top 10 Pickup-to-Dropoff Ratios:
                           zone  pickup_count  dropoff_count  pickup_drop_ratio
139  Springfield Gardens South           2.0            0.0           200000.0
39                  Douglaston           1.0            0.0           100000.0
2                   Auburndale           1.0            0.0           100000.0
76                Howard Beach           1.0            0.0           100000.0
75           Hillcrest/Pomonok           1.0            0.0           100000.0
32                Coney Island           1.0            0.0           100000.0
7      Bay Terrace/Fort Totten           1.0            0.0           100000.0
119             Pelham Parkway           1.0            0.0           100000.0
34           Crotona Park East           1.0            0.0           100000.0
118                Parkchester           1.0            0.0           100000.0

 Bottom 10 Pickup-to-Dropoff Ratios:
     zone  pickup_count  dropoff_count  pickup_drop_ratio
210    0           0.0           14.0                0.0
194    0           0.0            2.0                0.0
193    0           0.0            2.0                0.0
192    0           0.0            3.0                0.0
191    0           0.0            1.0                0.0
190    0           0.0           10.0                0.0
189    0           0.0            1.0                0.0
188    0           0.0            3.0                0.0
187    0           0.0            1.0                0.0
186    0           0.0            4.0                0.0
```

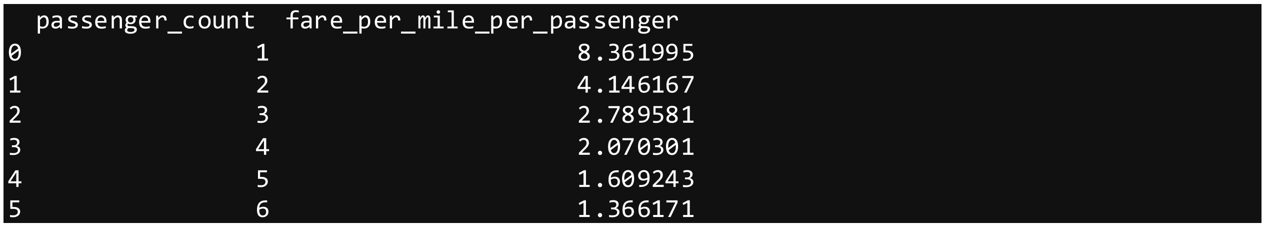
### 3.2.7 Identify the top zones with high traffic during night hours

### 3.2.8 Find the revenue share for nighttime and daytime hours

### 3.2.9 For the different passenger counts, find the average fare per mile per passenger

Lower the passenger count more fare per mile per passenger.

higher the passenger count lesser fare per mile per passenger.



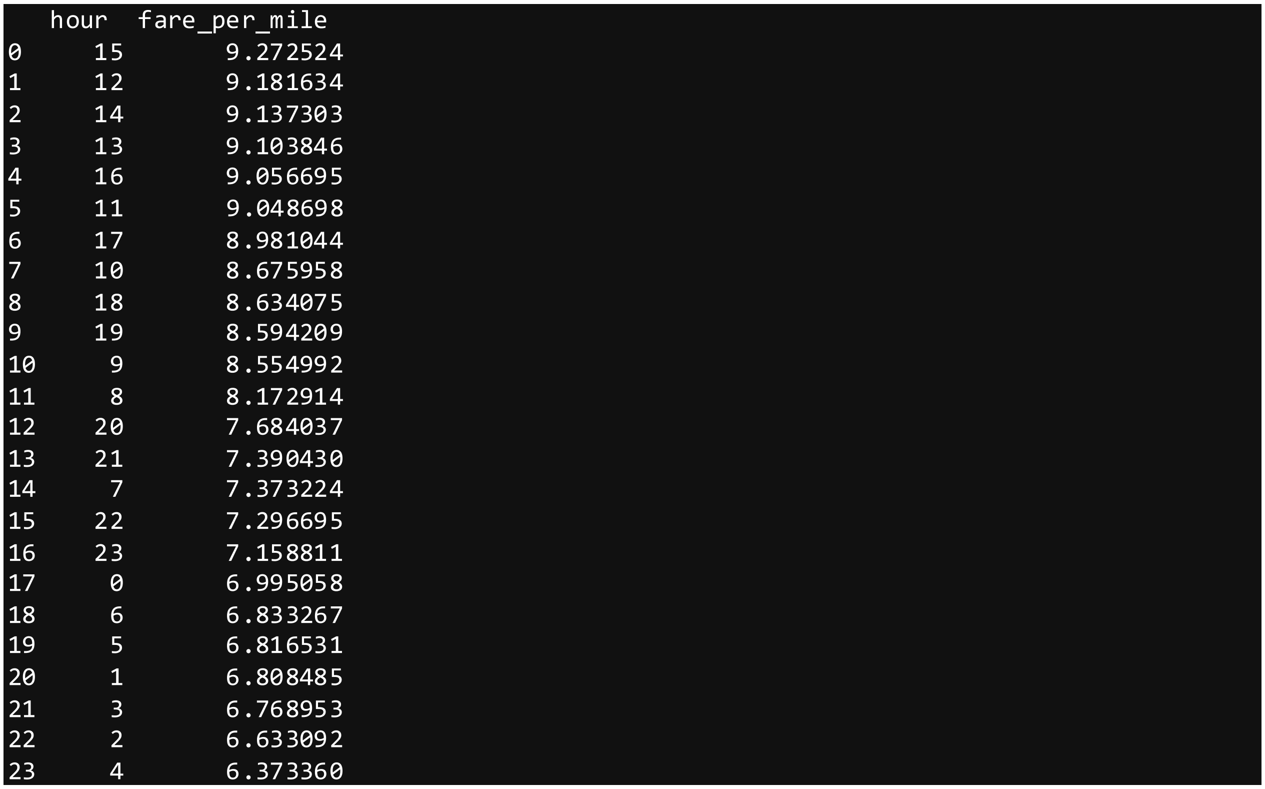
```
  passenger_count  fare_per_mile_per_passenger
0               1                     8.361995
1               2                     4.146167
2               3                     2.789581
3               4                     2.070301
4               5                     1.609243
5               6                     1.366171
```

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

At 3pm the fare per mile is greatest and at 4am it is least.

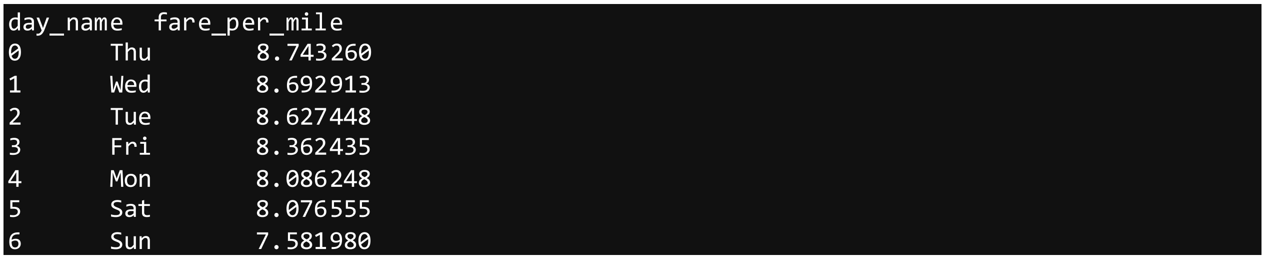
On thursday the fare per mile is greatest and on Sunday it is least.

**Fare per mile by hour:**



```
   hour  fare_per_mile
0     15       9.272524
1     12       9.181634
2     14       9.137303
3     13       9.103846
4     16       9.056695
5     11       9.048698
6     17       8.981044
7     10       8.675958
8     18       8.634075
9     19       8.594209
10     9       8.554992
11     8       8.172914
12    20       7.684037
13    21       7.390430
14     7       7.373224
15    22       7.296695
16    23       7.158811
17     0       6.995058
18     6       6.833267
19     5       6.816531
20     1       6.808485
21     3       6.768953
22     2       6.633092
23     4       6.373360
```

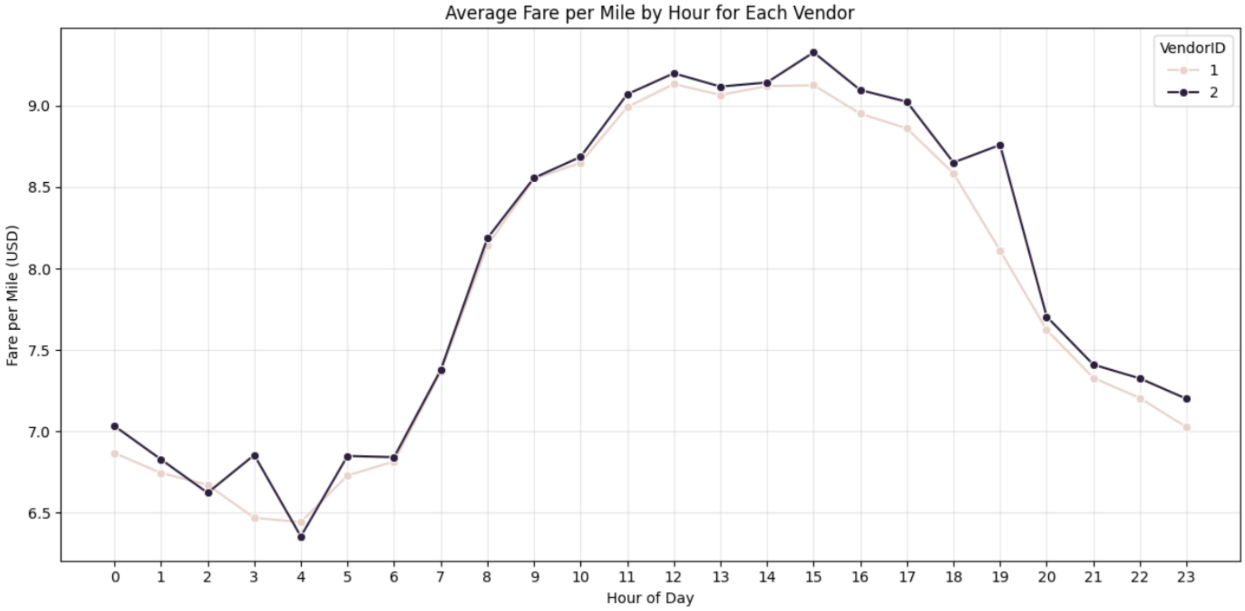
**Fare per mile by day:**



```
day_name  fare_per_mile
0      Thu       8.743260
1      Wed       8.692913
2      Tue       8.627448
3      Fri       8.362435
4      Mon       8.086248
5      Sat       8.076555
6      Sun       7.581980
```

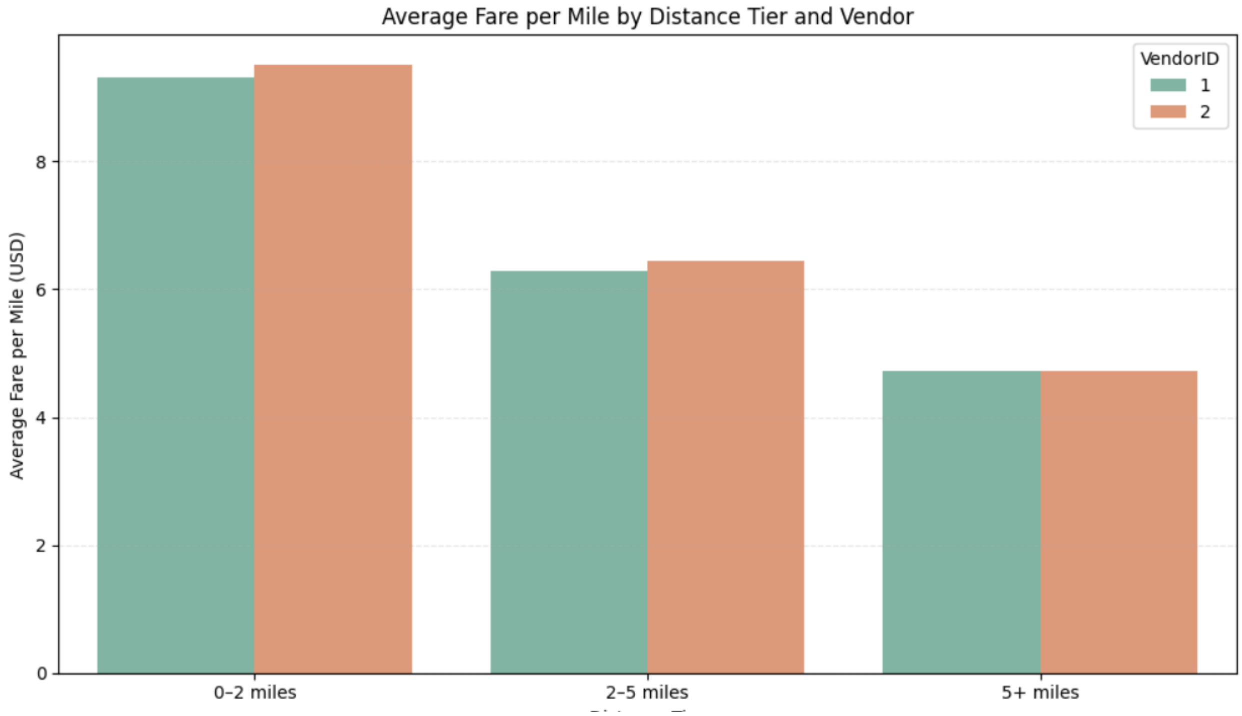
### 3.2.11  Analyse the average fare per mile for the different vendors

Vendor Curb Mobility LLC has high average fare than Creative Mobile Technologies for almost all hours except 6 to 10 am where it is same



### 3.2.12  Compare the fare rates of different vendors in a distance-tiered fashion

Vendor Curb Mobility LLC has high average fare per mile than Creative Mobile Technologies for 0-2 miles, 2-5 miles but for 5+ miles it is same.



### 3.2.13 Analyse the tip percentages

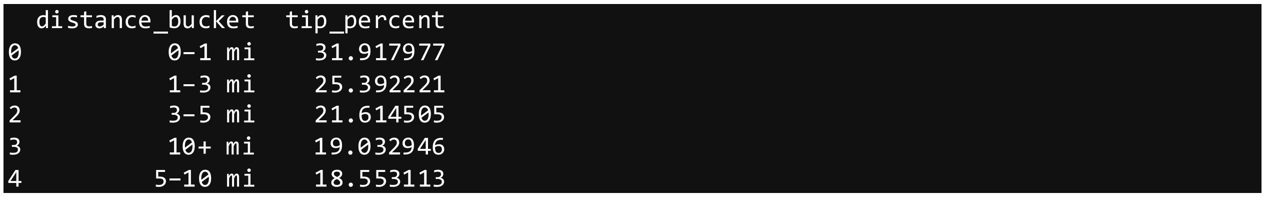
For 0-1 mile receives better share of tips. For higher distance journeys the tips get decreasing.

passenger count does not much have any relation to tips share received.

In busiest hours generally get good share of tips. 6pm is busiest we knew which got highest tips share.

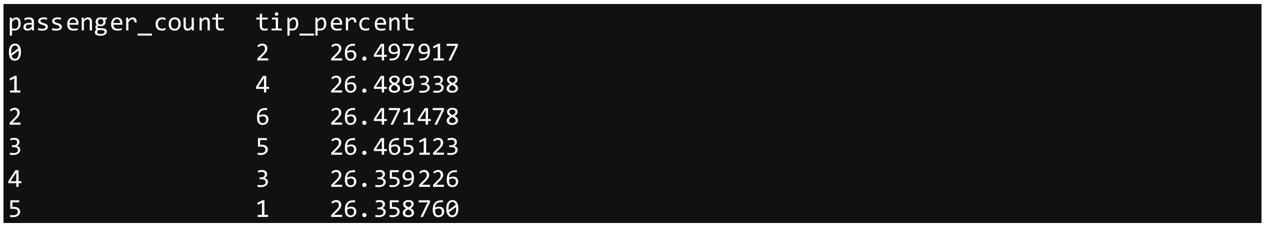
But in all hours the tip share is similar.

**Tip percentage based on distance bucket-**



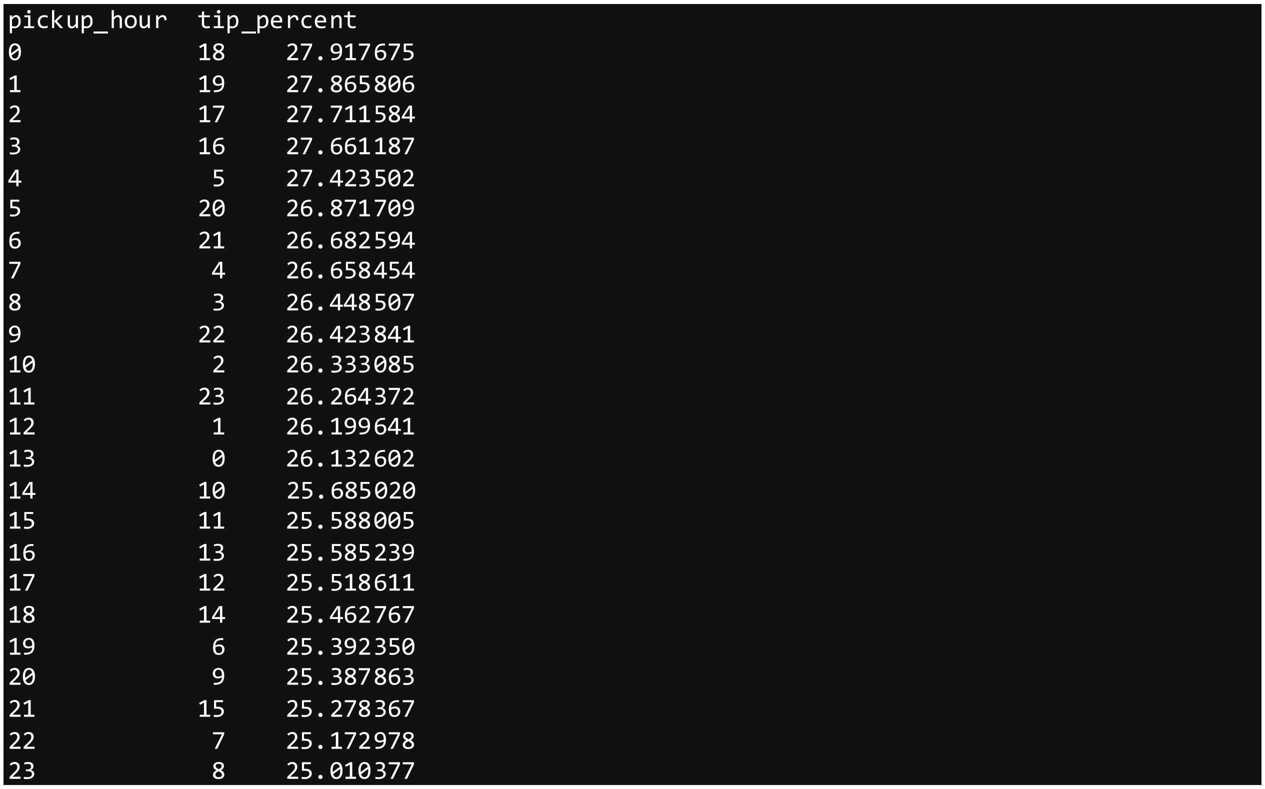
```
  distance_bucket  tip_percent
0          0-1 mi    31.917977
1          1-3 mi    25.392221
2          3-5 mi    21.614505
3         10+ mi    19.032946
4         5-10 mi    18.553113
```

**Tip percentage based on passenger count-**



```
passenger_count  tip_percent
0                2    26.497917
1                4    26.489338
2                6    26.471478
3                5    26.465123
4                3    26.359226
5                1    26.358760
```

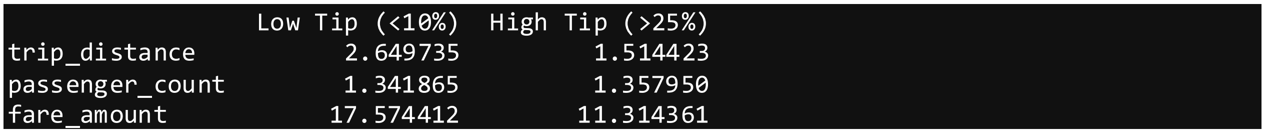
**Tip percentage based on pickup hour-**



```
pickup_hour  tip_percent
0            18    27.917675
1            19    27.865806
2            17    27.711584
3            16    27.661187
4             5    27.423502
5            20    26.871709
6            21    26.682594
7             4    26.658454
8             3    26.448507
9            22    26.423841
10            2    26.333085
11           23    26.264372
12            1    26.199641
13            0    26.132602
14           10    25.685020
15           11    25.588005
16           13    25.585239
17           12    25.518611
18           14    25.462767
19            6    25.392350
20            9    25.387863
21           15    25.278367
22            7    25.172978
23            8    25.010377
```

### 3.2.14 Analyse the trends in passenger count

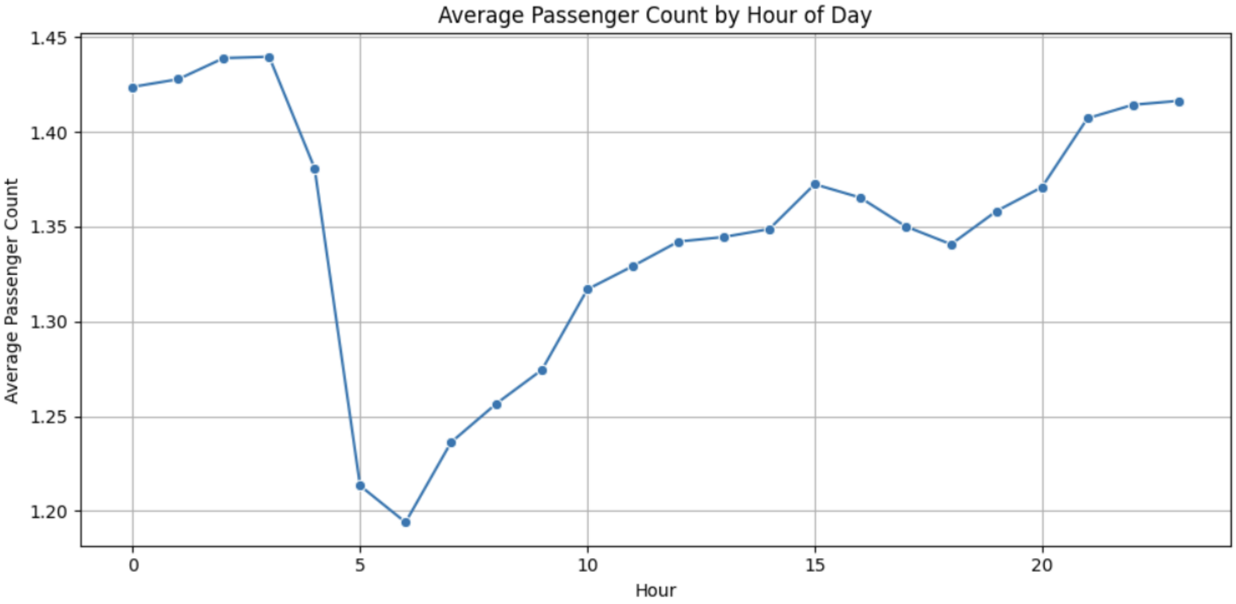
Low trip distance fetch higher share of tip on fare amount. Passenger count doesn’t influence tip share.



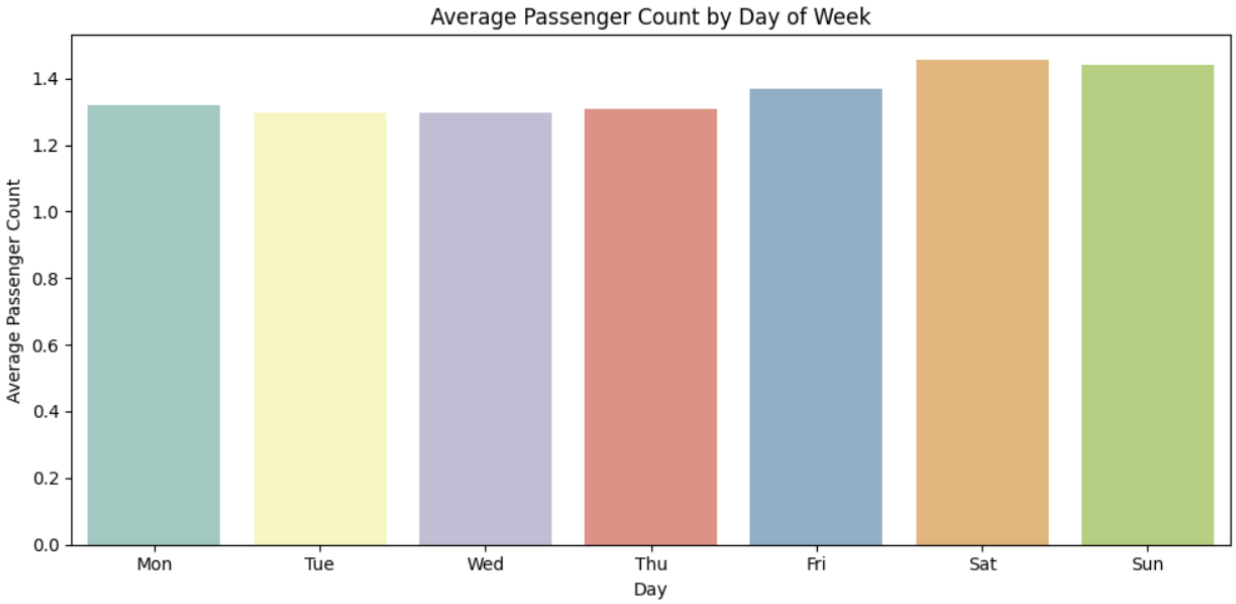
```
                 Low Tip (<10%)  High Tip (>25%)
trip_distance          2.649735         1.514423
passenger_count        1.341865         1.357950
fare_amount           17.574412        11.314361
```

### 3.2.15 Analyse the variation of passenger counts across zones

Average passenger count drops at 4-6am but starts increasing after that



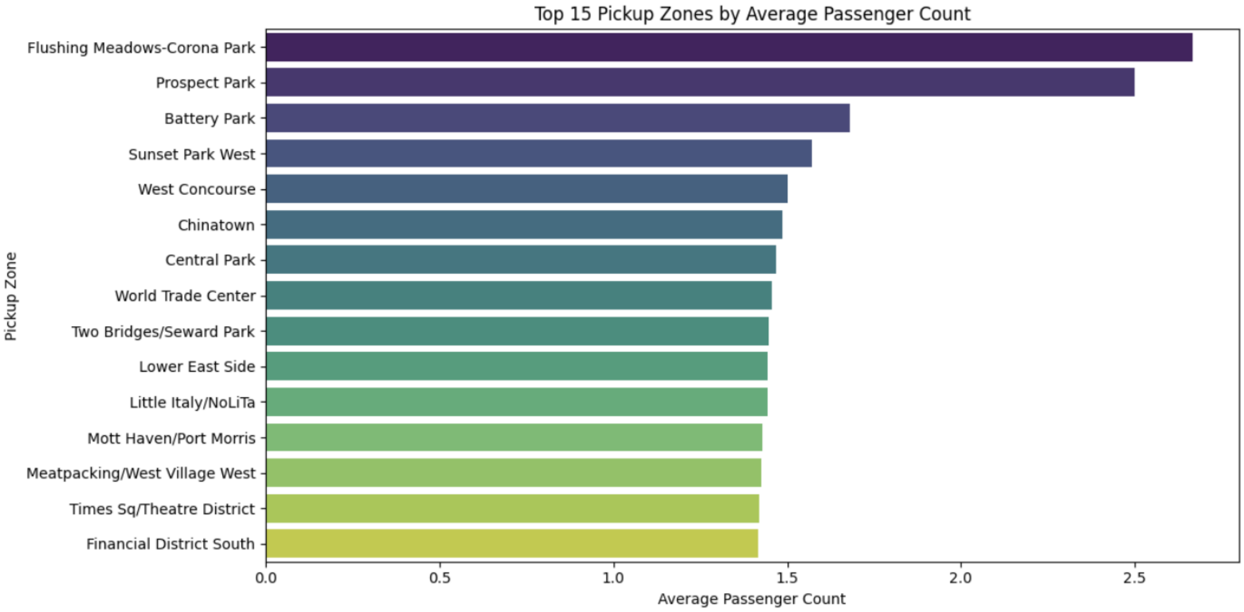
Average passenger count is high on Fri Sat Sun. People must be socializing as weekend comes closer.



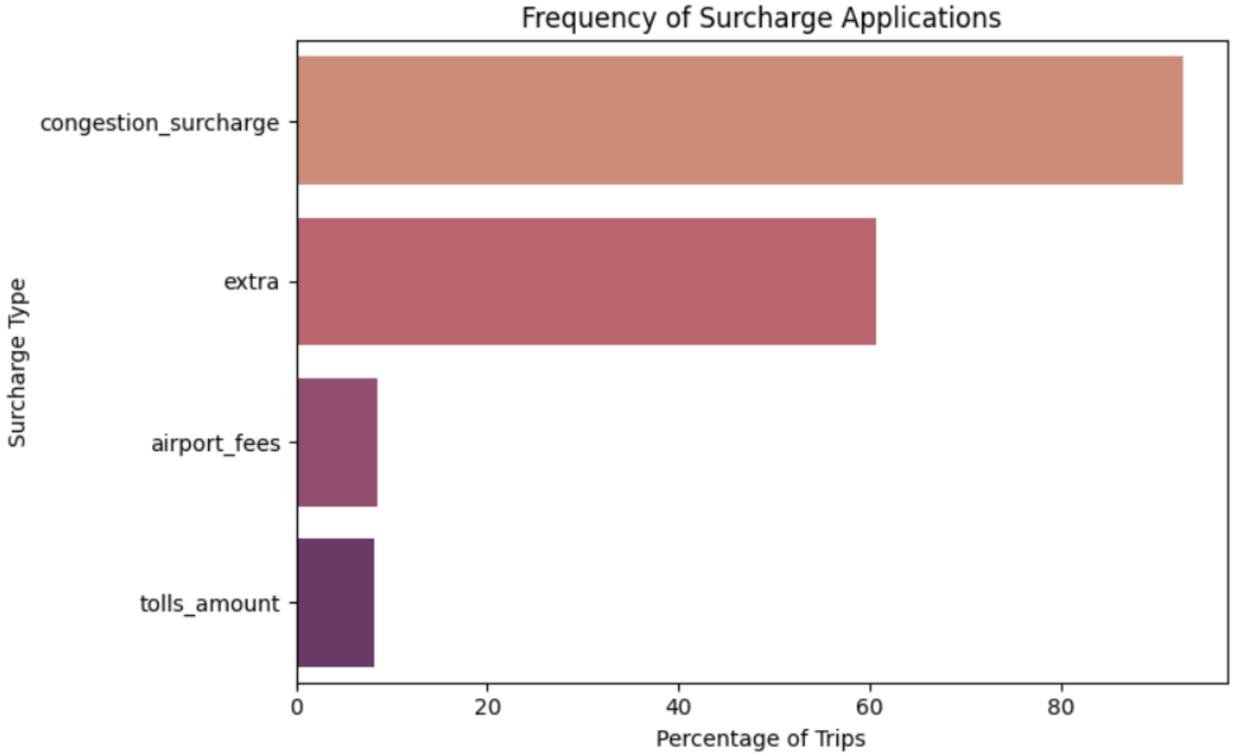
### 3.2.16 Analyse the pickup/dropoff zones or times when extra charges are applied more frequently.

Flushing meadows corona park and prospect park shown highest avg passenger count when it comes to top pickup zones.

These are beloved destinations for couples seeking romantic and memorable outings.



Congestion surcharge is applied in 80% of trips, followed by miscellaneous and extra surcharges that are applied 60 % of trips



# 4 Conclusions

## 4.2 Final Insights and Recommendations

### 4.2.1 Recommendations to optimize routing and dispatching based on demand patterns and operational inefficiencies.

- The slow routes are those where pickup and drop locations are same but there is absurdity which might be because of improper pickup and drop time of taxi system.

- The fare per mile per passenger is high for single passenger and decreases for higher count of passenger so During office hours the taxi availability should be high to make the business.

- Busy business hours are 11am to 10 pm with 6pm being the busiest. It is when people return home from their workplaces.

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

- Weekday trips are higher than weekend trips overall.
  Both show similar pattern for identical hours of day exception being on weekend between 12am to 4am also we can see higher trip counts compared to weekdays.

- The reason could be people relaxing and partying on weekends so staying up late and commuting to clubs, returning homes etc. so it makes sense to deploy taxis in these weekend late hours.

- People staring their day at 6am to 8 am on weekdays are higher than weekends.
  On weekends it might be that they people to start late so as to take a little rest from the week days work. So, it again makes sense to deploy taxis from early hours on weekdays but few hours late on weekends.

- Upper east side north, Upper east side south, Midtown center are the top 3 zones showing highest number of passenger pickup and drops.

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

- Deploy high taxi fleets in Manhattan.

- The fare per mile per passenger is high for single passenger and decreases for higher count of passenger so we can consider averaging the price such that we give some discount to single travelers and charge high for multiple passengers commuting in same taxi.

- At 3pm the fare per mile is greatest and at 4am it is least. So, we can levy night travel charge

- On Thursdays the fare per mile is greatest and on Sundays it is least.

- Vendor Curb Mobility LLC has high average fare than Creative Mobile Technologies for almost all hours except 6 to 10 am where it is same. Vendor Curb Mobility LLC has high average fare per mile than Creative Mobile Technologies for 0-2 miles, 2-5 miles but for 5+ miles it is same. So set pricing strategy as close to these vendors but slightly discounted to maximize revenue and remain competitive.

- Deploy more taxi fleet because we saw for shorter distance receives more tips because customer satisfaction could be higher for quick trip completions.

- Average passenger count drops at 4-6am but starts increasing after that so maintain high availability before 4pm and after 6pm.

- Average passenger count is high on Friday and weekends so ensure high availability, could be because people find time and love to socialize on those days.

- Flushing meadows corona park and prospect park shown highest avg passenger count when it comes to top pick up zones. These are beloved destinations for couples seeking romantic and memorable outings. so, maintain high availability around these parks as well.