Report: Optimising NYC Taxi Operations

Include your visualisations, 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

1.1.1. Sample the data and combine the files

12 parquet files are given to us, representing the 12 months of 2023 NYC Taxi Data. Data processing for this entire dataset would require very high compute and time. Just a single parquet file has 30,41,714 rows and 19 columns. 12 such files would mean 3,65,00,568 rows. Hence we sample it such that it is a representative of the overall data. We iterate through each file and extract the date and hour values. For each day, each hour we sample a percentage of data (1.5%) to create a smaller dataset (~6,00,000 rows).

2. Data Cleaning

2.1. Fixing Columns

2.1.1. Fix the index

Due to the sampling, indexes are also sampled, hence we reset it.

2.1.2. Combine the two airport_fee columns

Since neither of the columns are entirely null, we cannot simply drop either of them. Instead, we compare the two columns and consider the non-zero (Non Null) entry for the final column. We also observe there are negative values for this column (-1.75) which could be some refund amount or adjustment or even a system error.

2.2. Handling Missing Values

2.2.1. Find the proportion of missing values in each column

The following columns have missing values:

Column	Proportion
passenger_count	0.033826
RatecodeID	0.033826
store_and_fwd_flag	0.033826
congestion_surcharge	0.033826

2.2.2. Handling missing values in passenger_count

Some of the possibilities:

1) Mean: Mean is prone to outliers and may result in decimal values
2) Median: Median can be used
3) Zero: Filling it with zero is not an option as a trip with 0 passengers is not feasible.
4) **Mode**: Replacing with mode seems appropriate for passenger count because it preserves realistic integer values and reflects the most common scenario.

2.2.3. Handle missing values in RatecodeID

Same approach as section 2.2.2 as this is also a categorical data and using Mode is the most appropriate approach.

2.2.4. Impute NaN in congestion_surcharge

Most likely there is no surcharge when NA. Hence we replace these with zeros.

2.3. Handling Outliers and Standardising Values

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

1) Removed entries where passenger count > 6 as this is unrealistic and very rare.
2) Use Interquartile Range (IQR) method to remove outliers in fare amount and trip distance.
3) Remove entries with payment method 0 as this is not valid

No standardizing is needed as analysing raw data gives a better idea about the trends and patterns.

3. Exploratory Data Analysis

3.1. General EDA: Finding Patterns and Trends

3.1.1. Classify variables into categorical and numerical

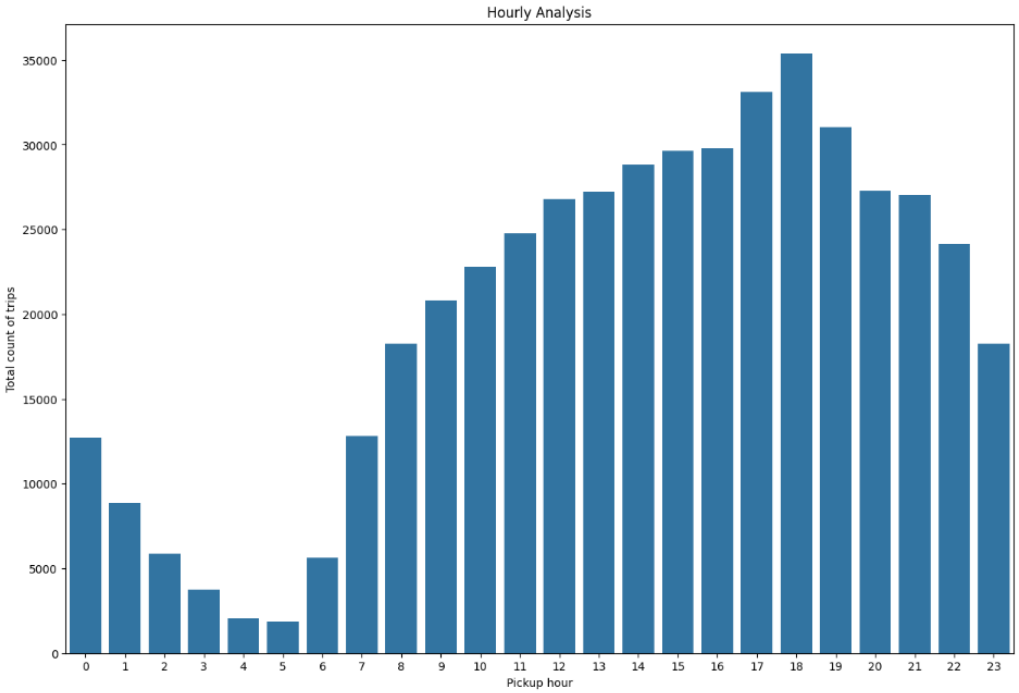
Numerical	Categorical
tpep_pickup_datetime tpep_dropoff_datetime	tpep_pickup_datetime tpep_dropoff_datetime
passenger_count	RatecodeID
trip_distance	payment_type
trip_duration	VendorID
pickup_hour	PULocationID
	DOLocationID

Note: tpep_pickup_datetime and tpep_dropoff_datetime can be either numerical or categorical data based on usage. If we directly use it to calculate the duration, then it is numerical. But when we extract information like the day or month then it becomes categorical.

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

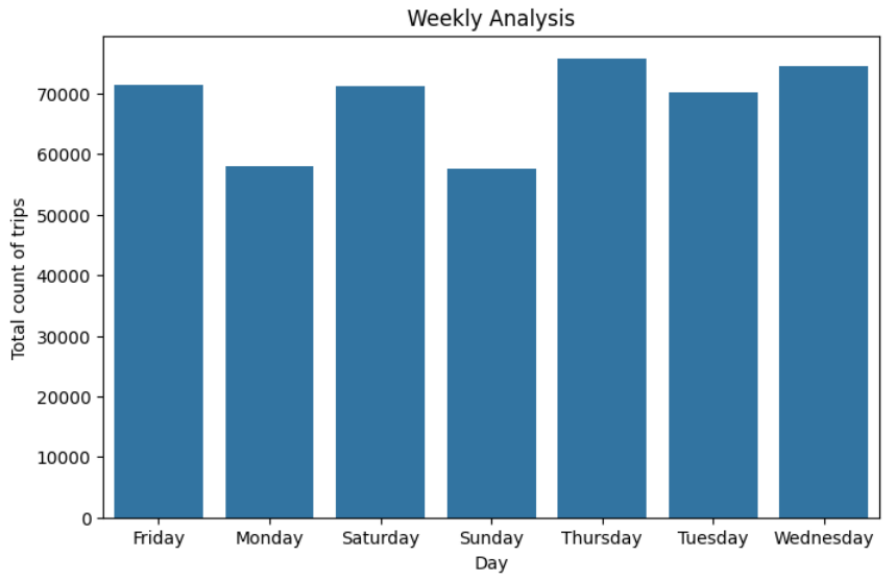
Hours:

Extract hour data from ‘tpep_pickup_datetime’. Get the count values by groupby ‘pickup_hour’



Analysis: We can see the number of trips are the least at night (3am-5am) and maximum in the evening (5pm-7pm). This peak could be because most people are returning from offices during this period.

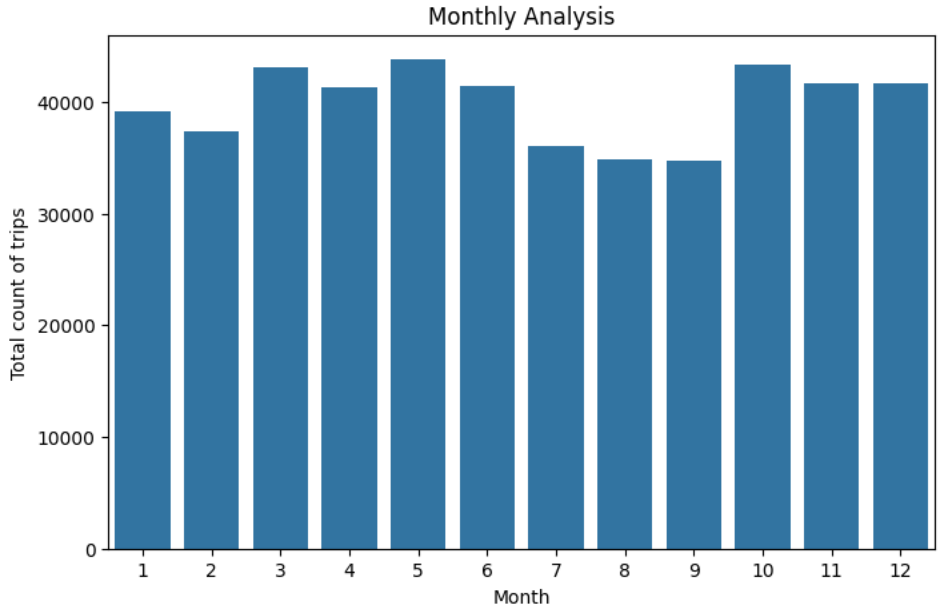
Days of the week:



Analysis: Variation of data is not very high. We observe slight dip in the number of trips on Sunday and Monday.

Months:

```
month
5     43804
10    43379
3     43121
11    41705
12    41606
6     41376
4     41323
1     39115
2     37375
7     36092
8     34851
9     34681
Name: tpep_pickup_datetime, dtype: int64
```



Analysis: Variation of data is not very high. There is a slight dip in the number of trips from July-September.

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

3.1.4. Analyse the monthly revenue trends

Month total_amount

8 556708.84

7 576062.61

9 596046.73

2 613213.23

1 627761.29

4 675560.47

6 690296.27

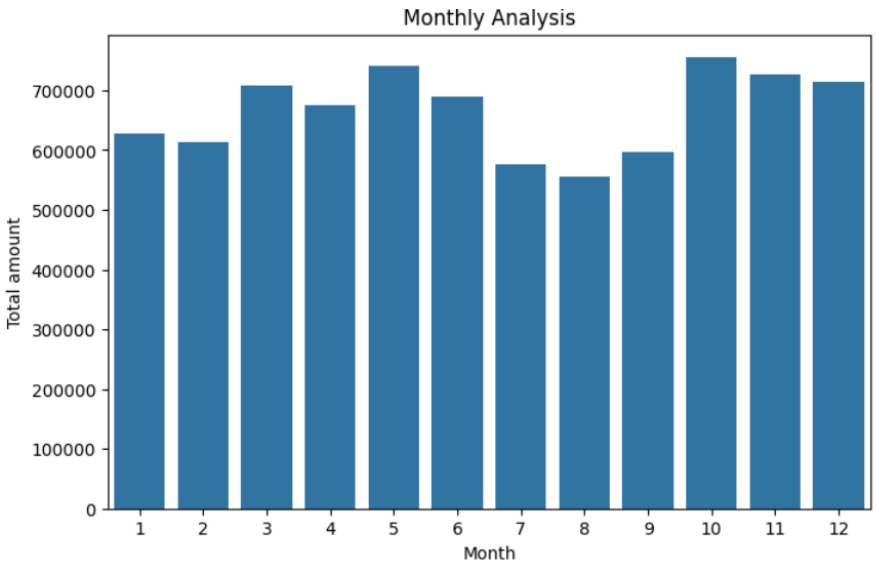
3 707736.00

12 714767.42

11 725929.81

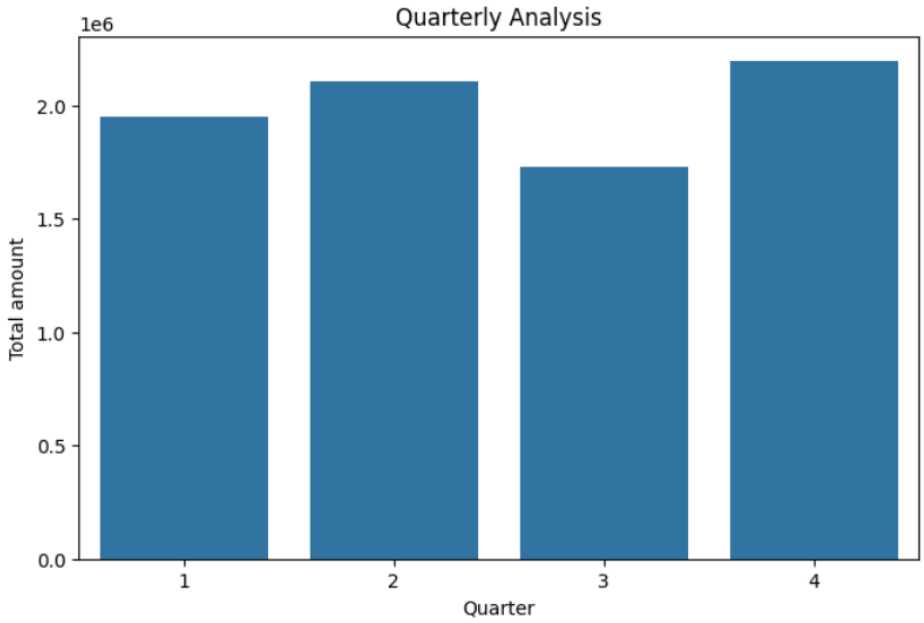
5 741919.14

10 755417.13



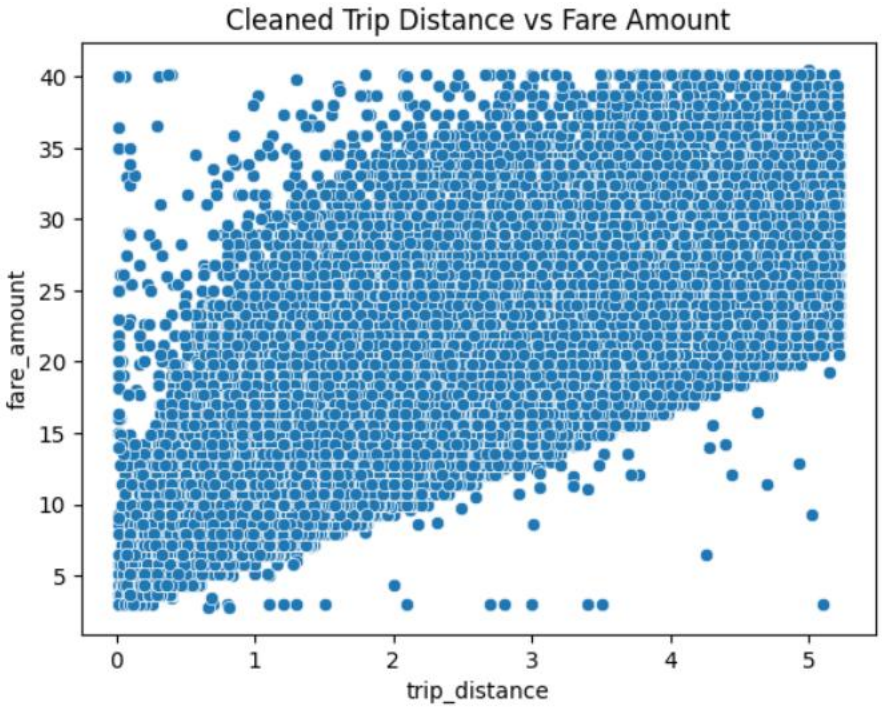
3.1.5. Find the proportion of each quarter's revenue in the yearly revenue

```
Proportions:  quarter
4    0.275153
2    0.264085
1    0.244156
3    0.216605
Name: total_amount, dtype: float64
```



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

Correlation value between Fare amount and Trip distance is:
0.8612089576585235



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

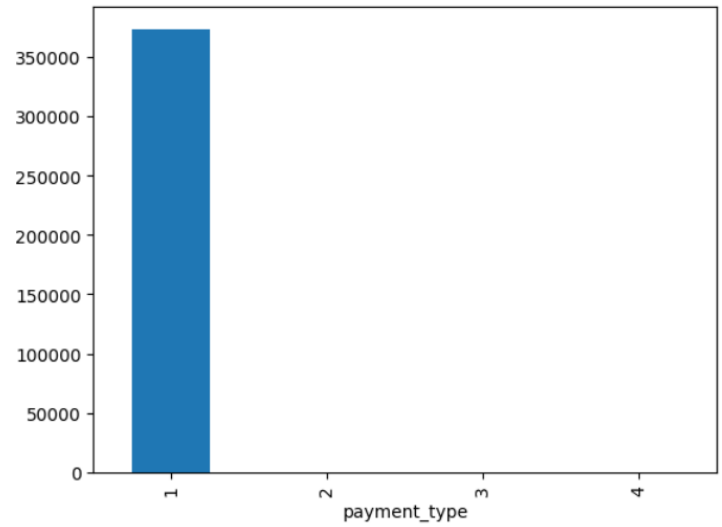
Correlation between Trip duration and Fare amount is:
0.9527897348634646

Correlation between Fare amount and Passenger count is:
0.01727958073589577

Correlation between Trip distance and Tip amount is:
0.5022403242600242

3.1.8. Analyse the distribution of different payment types

```
payment_type
1    373546
2         6
4         4
3         1
Name: payment_type, dtype: int64
```



3.1.9. Load the taxi zones 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 the zone data with trips data

Merged the data with left_on='PULocationID' (df), right_on='LocationID' (zone)

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

LocationID	LocationID
237	21143
161	19267
236	19219
162	15221
142	14309
...	...
196	1
219	1
218	1
197	1
1	1

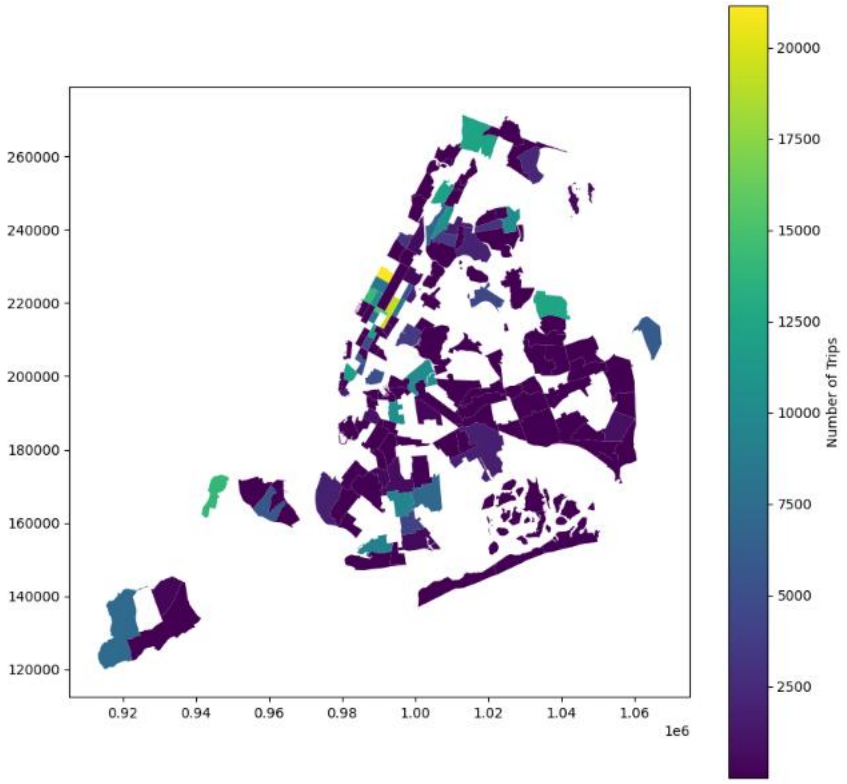
148 rows × 1 columns

dtype: int64

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

	OBJECTID	Shape_Leng	Shape_Area	zone	LocationID	borough	geometry	trip_count
0	1	0.116357	0.000782	Newark Airport	1	EWR	POLYGON ((933100.918 192536.086, 933091.011 19...	NaN
1	2	0.433470	0.004866	Jamaica Bay	2	Queens	MULTIPOLYGON (((1033269.244 172126.008, 103343...	1.0
2	3	0.084341	0.000314	Allerton/Pelham Gardens	3	Bronx	POLYGON ((1026308.77 256767.698, 1026495.593 2...	NaN
3	4	0.043567	0.000112	Alphabet City	4	Manhattan	POLYGON ((992073.467 203714.076, 992068.667 20...	NaN
4	5	0.092146	0.000498	Arden Heights	5	Staten Island	POLYGON ((935843.31 144283.336, 936046.565 144...	386.0
5	6	0.150491	0.000606	Arrochar/Fort Wadsworth	6	Staten Island	POLYGON ((966568.747 158679.855, 966615.256 15...	NaN
6	7	0.107417	0.000390	Astoria	7	Queens	POLYGON ((1010804.218 218919.641, 1011049.165 ...	NaN
7	8	0.027591	0.000027	Astoria Park	8	Queens	POLYGON ((1005482.276 221686.466, 1005304.898 ...	68.0
8	9	0.099784	0.000338	Auburndale	9	Queens	POLYGON ((1043803.993 216615.925, 1043849.708 ...	NaN
9	10	0.099839	0.000436	Baisley Park	10	Queens	POLYGON ((1044355.072 190734.321, 1044612.122 ...	1.0

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



3.1.14. Conclude with results

- Busiest hours, days and months:
    - Evenings are usually the busiest (5-7 PM) could be due to office commute
    - Thursday-Saturday have higher number of trips could be driven by both commuting and leisure activity.
    - October-December months have higher demand could be due to holiday season
- Trends in revenue collected
    - **Correlation Analysis:**
        - Fare ↔ Trip Distance: **0.861** (strong).
        - Fare ↔ Trip Duration: **0.953** (very strong).
        - Passenger Count ↔ Fare: **0.017** (negligible).
        - Tip Amount ↔ Trip Distance: **0.502** (moderate).

        - Tip % ↔ Trip Characteristics: No meaningful correlation.
    - **Day vs Night Revenue:**
        - Daytime (6 AM–6 PM): **$16,996,466** (~89% share).
        - Nighttime (6 PM–6 AM): **$2,052,374** (~11% share).
- Trends in quarterly revenue
    - **Highest Quarter:** Q4
    - **Lowest Quarter:** Q1 — impacted by February slump
    - **Most Stable Quarter:** Q2 — consistent month-to-month performance
- How fare depends on trip distance, trip duration and passenger counts
    - Trip distance: Correlation: 0.861 → strong positive relationship.
    - Trip duration: Correlation: 0.953 → extremely strong relationship.
    - Passenger count: Correlation: 0.017 → negligible relationship. Number of passengers has almost no impact on fare amount — fare structure is per trip, not per passenger.
- How tip amount depends on trip distance
    - Correlation: 0.502 → moderate positive relationship.
    - Tipping behavior depends heavily on rider generosity, service quality, and payment method, so distance alone doesn't explain all variation.
- Busiest zones
    - Top five areas are: Upper East Side South and North, Midtown Center and East, Lincoln Square East.

3.2. Detailed EDA: Insights and Strategies

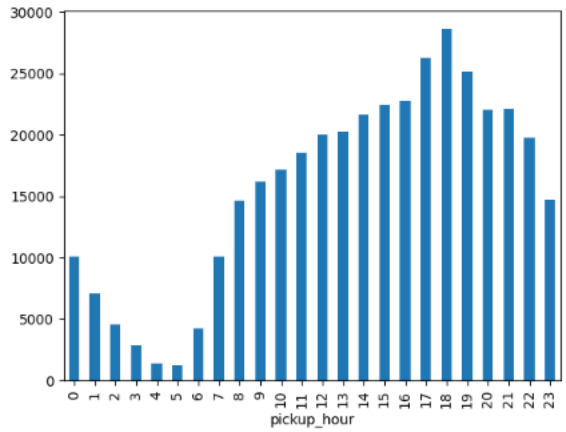
3.2.1. Identify slow routes by comparing average speeds on different routes

The slowest speed for every hour is diplayed in the below table

	pickup_hour	speed	PULocationID	DOLocationID
0	0	1.714286	186	186
1	1	0.127919	48	184
2	2	0.141176	255	256
3	3	1.384615	163	163
4	4	1.925134	141	75
5	5	4.142259	43	162
6	6	1.988950	48	48
7	7	1.419355	48	48
8	8	0.867470	233	233
9	9	0.692308	138	138
10	10	0.277858	186	170
11	11	0.088452	68	68
12	12	0.347490	229	229
13	13	0.511364	161	161
14	14	0.025955	264	264
15	15	0.129964	236	236
16	16	0.367347	43	43
17	17	0.157895	13	264
18	18	0.675422	163	162
19	19	0.025844	186	142
20	20	0.135338	163	48
21	21	0.611111	161	161
22	22	0.227848	164	164
23	23	1.344764	164	230

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

```
pickup_hour
18    28645
17    26282
19    25114
16    22797
15    22424
21    22141
20    22039
14    21600
13    20235
12    20011
22    19737
11    18526
10    17122
9     16144
23    14726
8     14626
0     10099
7     10067
1      7049
2      4542
6      4184
3      2857
4      1383
5      1207
Name: pickup_hour, dtype: int64
<Axes: xlabel='pickup_hour'>
```



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

```
pickup_hour
0      673266.00
1      469933.00
2      302800.00
3      190466.00
4       92200.00
5       80466.00
6      278933.00
7      671133.00
8      975066.00
9     1076266.00
10    1141466.00
11    1235066.00
12    1334066.00
13    1349000.00
14    1440000.00
15    1494933.00
16    1519800.00
17    1752133.00
18    1909666.00
19    1674266.00
20    1469266.00
21    1476066.00
22    1315800.00
23     981733.00
Name: pickup_hour, dtype: object
```

3.2.4. Compare hourly traffic on weekdays and weekends

is_weekend	False	True
pickup_hour		
0	916.4	2758.0
1	464.2	2360.0
2	235.8	1681.5
3	133.2	1095.5
4	85.8	477.0
5	193.2	120.5
6	732.8	260.0
7	1820.8	481.0
8	2579.2	865.0
9	2624.0	1512.0
10	2586.8	2094.0
11	2710.0	2487.5
12	2863.8	2846.0
13	2887.6	2898.5
14	3156.6	2908.0
15	3318.2	2916.0
16	3363.2	2990.0
17	3999.6	3142.0
18	4445.4	3208.5
19	3845.0	2944.5
20	3396.0	2529.5
21	3434.4	2484.5
22	2999.4	2370.0
23	2096.8	2121.0

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

```
Top 10 pickup zones:  PULocationID
237    21141
161    19267
236    19219
162    15219
142    14309
186    14095
170    12469
239    12335
234    12114
163    12001
Name: PULocationID, dtype: int64
Top 10 dropoff zones:  DOLocationID
236    20106
237    18942
161    15872
170    12497
142    12390
239    12258
141    11772
162    11692
234    10377
68     10348
Name: DOLocationID, dtype: int64
```

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

```
Ten highest Pickup Ratios PULocationID
237    0.056596
161    0.051579
236    0.051451
162    0.040742
142    0.038306
186    0.037733
170    0.033380
239    0.033022
234    0.032430
163    0.032128
Name: PULocationID, dtype: float64
Ten highest Dropoff Ratios DOLocationID
236    0.053825
237    0.050709
161    0.042491
170    0.033455
142    0.033169
239    0.032816
141    0.031515
162    0.031300
234    0.027780
68     0.027702
Name: DOLocationID, dtype: float64
```

```
Ten lowest Pickup Ratios PULocationID
1      0.000003
39     0.000003
47     0.000003
56     0.000003
196    0.000003
29     0.000003
258    0.000003
28     0.000003
11     0.000003
9      0.000003
Name: PULocationID, dtype: float64
Ten lowest Dropoff Ratios DOLocationID
1      0.000003
248    0.000003
38     0.000003
153    0.000003
101    0.000003
64     0.000003
235    0.000003
136    0.000003
167    0.000003
177    0.000003
Name: DOLocationID, dtype: float64
```

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

```
Top 10 Pickup Zones:  PULocationID
79     3628
249    2986
148    2254
48     2128
114    2005
230    1562
186    1420
164    1368
107    1317
234    1264
Name: PULocationID, dtype: int64
Top 10 Dropoff Zones:  DOLocationID
79     2177
48     1659
170    1588
107    1560
141    1409
68     1387
249    1333
263    1243
236    1143
229    1136
Name: DOLocationID, dtype: int64
```

3.2.8. Find the revenue share for nighttime and daytime hours

Night revenue share: 2052373.57

Daytime revenue share: 16996466.1

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

passenger_count	per_passenger
1.0	8.505131
2.0	4.253611
3.0	2.915761
4.0	2.268665
5.0	1.615320
6.0	1.369388

dtype: float64

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

By hour of the day:

```
pickup_hour
14    9.357426
12    9.273978
15    9.269571
11    9.160660
13    9.147435
17    9.090602
10    9.080922
16    9.012908
19    8.930382
9     8.827730
18    8.734189
8     8.276615
2     8.037398
20    7.829105
22    7.558143
21    7.504805
7     7.383306
6     7.274267
23    7.136492
1     7.123208
3     7.049280
0     7.013985
5     6.712204
4     6.553744
Name: fare_per_mile, dtype: float64
```

By day:

```
day
Thursday     8.909659
Wednesday    8.852795
Tuesday      8.809311
Friday       8.367368
Saturday     8.208880
Monday       8.197912
Sunday       8.045313
Name: fare_per_mile, dtype: float64
```

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

VendorID	fare_per_mile
2	8.571576
1	8.354928

dtype: float64

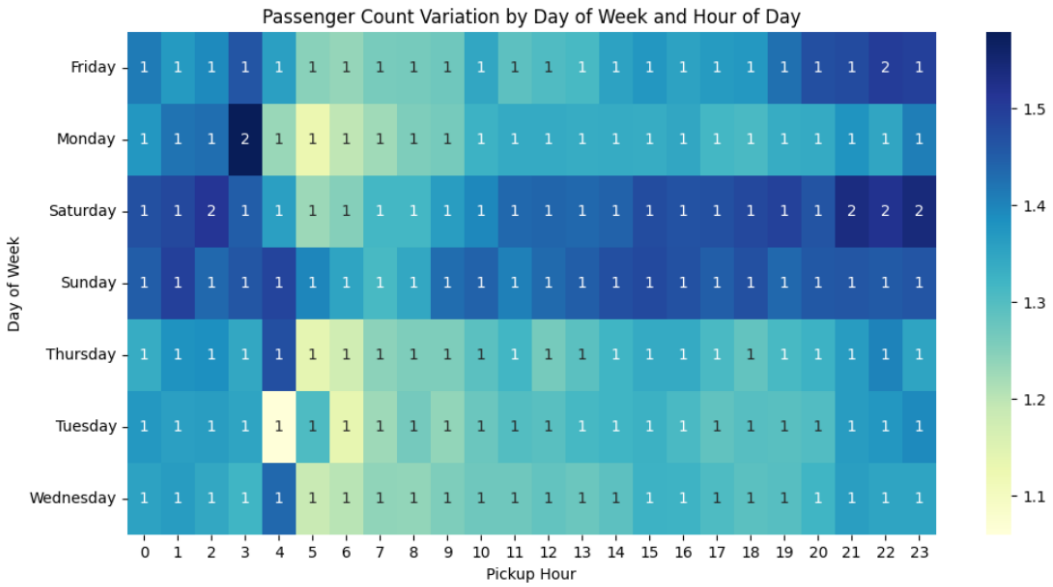
3.2.12. Compare the fare rates of different vendors in a distance-tiered fashion

distance_tier	0-2	2-5	5+
VendorID			
1	9.322905	6.344036	5.220549
2	9.695725	6.505858	5.297038

3.2.13. Analyse the tip percentages

There is no correlation between the tip percentage with either the distance or passenger count.

3.2.14. Analyse the trends in passenger count



3.2.15. Analyse the variation of passenger counts across zones

PULocationID	passenger_count
178	6.0
47	5.0
191	3.0
28	3.0
82	2.8
...	...
212	1.0
215	1.0
216	1.0
217	1.0
1	1.0

150 rows × 1 columns

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

```
Highest congestion surcharge pickup locations:  PULocationID
237    21133
161    19252
236    19175
162    15208
142    14299
Name: PULocationID, dtype: int64


Highest congestion surcharge dropoff locations:  DOLocationID
236    19983
237    18932
161    15855
170    12489
142    12383
Name: DOLocationID, dtype: int64


Highest congestion surcharge day and hour:  day        pickup_hour
Thursday   18             4729
Wednesday  18             4689
Tuesday    18             4527
Friday     18             4241
Thursday   17             4206
Name: pickup_hour, dtype: int64
```

```
Highest improvement surcharge pickup locations:  PULocationID
237    21143
161    19267
236    19219
162    15221
142    14309
Name: PULocationID, dtype: int64
```

```
Highest improvement surcharge dropoff locations:  DOLocationID
236    20107
237    18944
161    15872
170    12499
142    12391
Name: DOLocationID, dtype: int64
```

```
Highest improvement surcharge day and hour:  day        pickup_hour
Thursday   18             4818
Wednesday  18             4769
Tuesday    18             4600
Friday     18             4320
Thursday   17             4282
Name: pickup_hour, dtype: int64
```

4. Conclusions

4.1. Final Insights and Recommendations

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

Prioritize Peak Demand Hours (5–7 PM)

- Deploy additional cabs during **5–7 PM**, when demand reaches its peak.

Route Optimization for Slow Corridors

- Hours and locations with lowest average cab speeds can be routed to faster alternative streets during those periods to reduce trip times and enable more daily trips per vehicle.

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.

Reduce Idle Mileage through Predictive Positioning

- Assign more drivers to cover both commuter-heavy zones and leisure/nightlife areas during peak hours. Top five areas are: Upper East Side South and North, Midtown Center and East, Lincoln Square East.

Prioritize High demand zones

- Use hourly, weekly pickup–drop-off patterns to position idle cabs closer to anticipated next-trip zones.
  Example: Trips ending in high-demand neighbouring zones should be followed by repositioning rather than waiting in low-demand drop-off areas.

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

Peak period premiums

- Introduce a **fare increase** during **5–7 PM weekday peaks** and in high-demand zones such as Midtown, airports. These hours show strong willingness to pay due to commuting urgency.

Off-Peak Discounts to Stimulate Demand

- Offer **fare reductions** during **3–5 AM** and low-demand days (Sunday and Monday), especially for airport trips. Small discounts can help improve utilization without undercutting base revenue.

Cab sharing scheme

- Many trips share similar origin-destination pairs (especially airport, nightlife, and business districts). Cab-sharing could **reduce per-person fares**, making the service more attractive, particularly for cost-sensitive users.
- Monitor competitive pricing to ensure rates remain attractive against ride-hailing services.