BIDV PROJECT

THE NORTHCAP UNIVERSITY

Topic - Cab Services

Tableau

Name – Bhagyashri Pradhan

Class – 5TH SEM DS-B (DS-4)

Roll no- 21CSU313

Index

- Problem statement
- Sub problems
- Introduction (why it is required, what it is, difficulty to handle this)
- Objective
- Dataset
- Preprocessing
- Charts
- Dashboard

Problem statement: Cab drivers face several challenges in their day-to-day operations, ranging from navigating through traffic congestion to managing passenger requests effectively. To enhance the overall experience for both drivers and passengers, there is a need for innovative solutions that address the following key problems.

Sub problems

1)Route optimization

2) Fare optimization

3)Passenger allocation

4)Traffic management

5)Competitive market

Introduction

Providing Cab facilities with optimizing possible problems.

addressing the challenges faced by cab drivers is essential for improving the overall efficiency, safety, and sustainability of urban transportation systems. By optimizing routes, enhancing passenger satisfaction, ensuring safety and security, promoting transparency, taxi services can offer a more appealing and competitive option in a rapidly evolving transportation landscape. These improvements not only benefit drivers and passengers but also contribute to urban mobility, environmental sustainability, and economic growth, making it imperative to explore innovative, data-driven solutions to tackle these issues effectively.

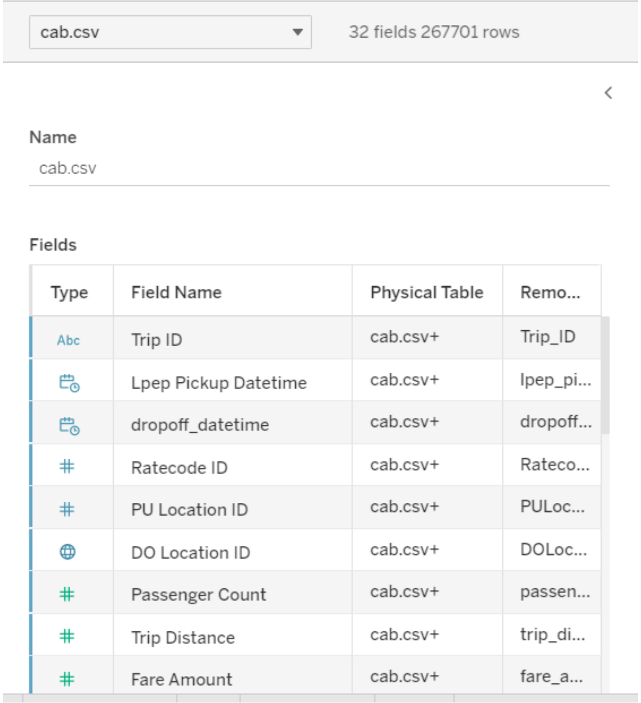
Objective:

The primary objective of addressing the challenges outlined in the given problem statement is to enhance the overall efficiency, Optimization, and quality of taxi services within urban environments. To achieve this, several key objectives have been identified:

- Route Optimization: Improve travel efficiency by developing algorithms that optimize taxi routes, reducing travel time and environmental impact.
- Passenger Allocation: Enhance the passenger experience by creating systems that efficiently match passengers with available taxis, reducing wait times and improving resource utilization.
- Traffic Management: Implement real-time traffic management solutions to help drivers navigate congested areas and optimize route choices for a smoother journey.
- Cab availability: refers to the accessibility of taxi services for commuters in urban areas. In today's fast-paced world, the convenience of hailing a cab or using a ride-sharing app has become indispensable.
- Fare optimization: Ensuring the pricing with calculating distance and giving the best fare Pricing.

Data Set

1) CAB



cab.csv ▾ 32 fields 267701 rows

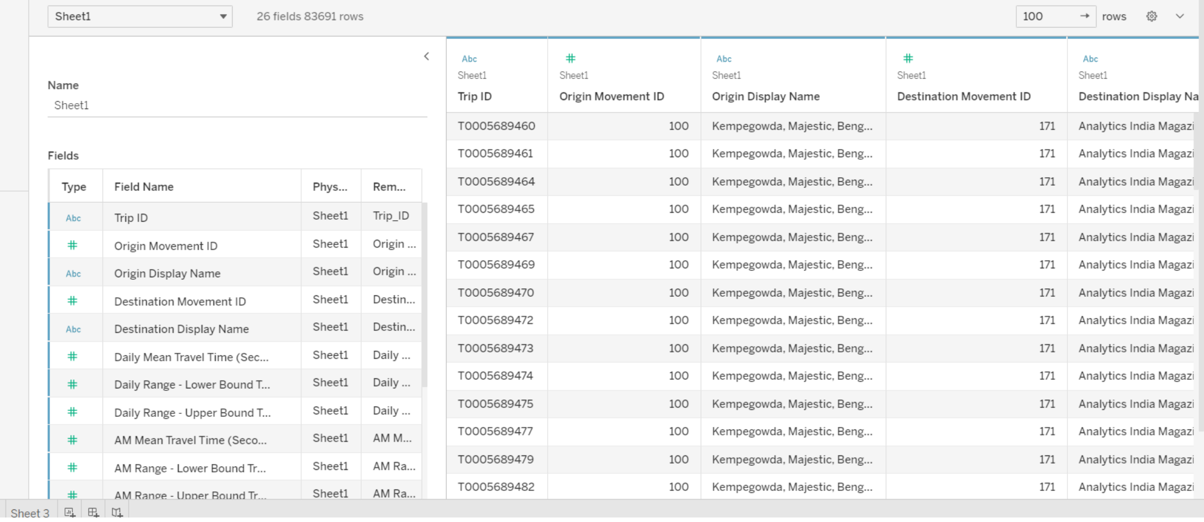
Name

cab.csv

Fields

Type	Field Name	Physical Table	Remo...
Abc	Trip ID	cab.csv+	Trip_ID
📅	Lpep Pickup Datetime	cab.csv+	lpep_pi...
📅	dropoff_datetime	cab.csv+	dropoff...
#	Ratecode ID	cab.csv+	Rateco...
#	PU Location ID	cab.csv+	PULoc...
🌐	DO Location ID	cab.csv+	DOLoc...
#	Passenger Count	cab.csv+	passen...
#	Trip Distance	cab.csv+	trip_di...
#	Fare Amount	cab.csv+	fare_a...

2) Traffic



Sheet1 ▾ 26 fields 83691 rows

100 → rows

Name

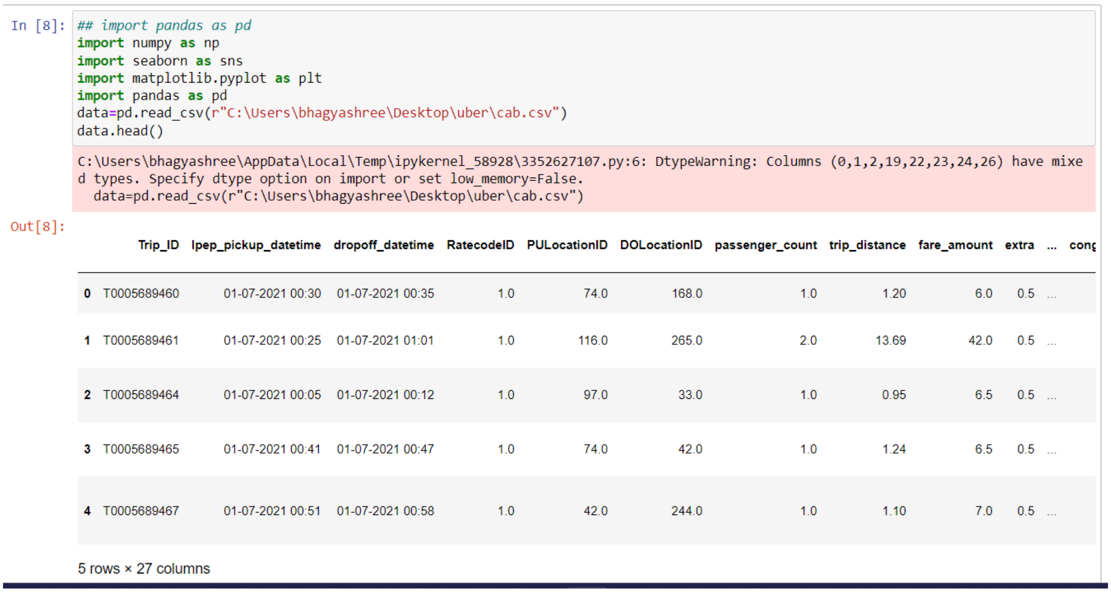
Sheet1

Fields

Type	Field Name	Phys...	Rem...
Abc	Trip ID	Sheet1	Trip_ID
#	Origin Movement ID	Sheet1	Origin ...
Abc	Origin Display Name	Sheet1	Origin ...
#	Destination Movement ID	Sheet1	Destin...
Abc	Destination Display Name	Sheet1	Destin...
#	Daily Mean Travel Time (Sec...	Sheet1	Daily ...
#	Daily Range - Lower Bound T...	Sheet1	Daily ...
#	Daily Range - Upper Bound T...	Sheet1	Daily ...
#	AM Mean Travel Time (Seco...	Sheet1	AM M...
#	AM Range - Lower Bound Tr...	Sheet1	AM Ra...
#	AM Range - Upper Bound Tr...	Sheet1	AM Ra...

Abc Sheet1 Trip ID	# Sheet1 Origin Movement ID	Abc Sheet1 Origin Display Name	# Sheet1 Destination Movement ID	Abc Sheet1 Destination Display Na
T0005689460	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689461	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689464	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689465	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689467	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689469	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689470	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689472	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689473	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689474	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689475	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689477	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689479	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi
T0005689482	100	Kempegowda, Majestic, Beng...	171	Analytics India Magazi

Preprocessing

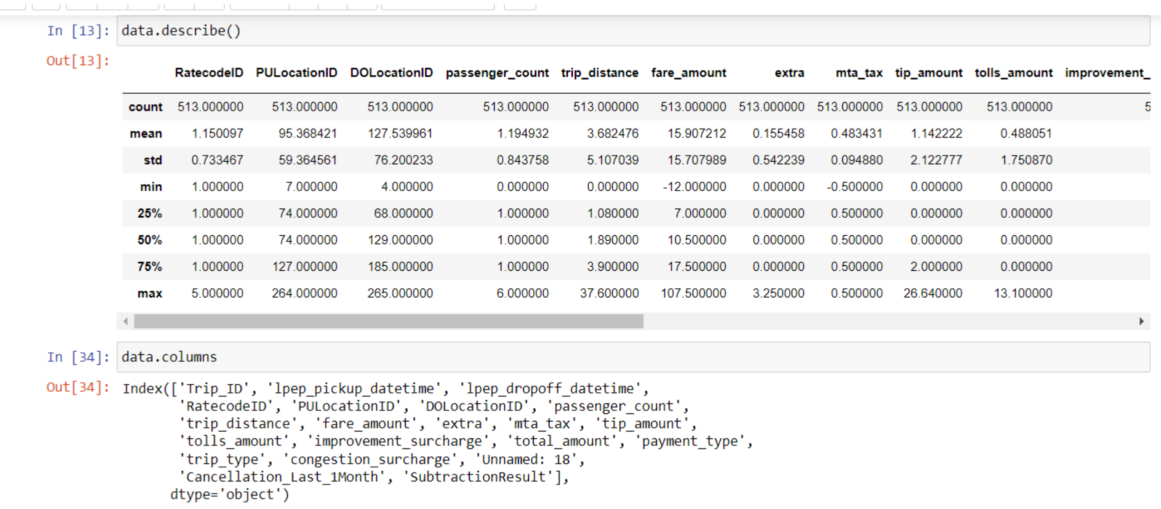


```
In [8]: ## import pandas as pd
        import numpy as np
        import seaborn as sns
        import matplotlib.pyplot as plt
        import pandas as pd
        data=pd.read_csv(r"C:\Users\bhagyashree\Desktop\uber\cab.csv")
        data.head()

C:\Users\bhagyashree\AppData\Local\Temp\ipykernel_58928\3352627107.py:6: DtypeWarning: Columns (0,1,2,19,22,23,24,26) have mixe
d types. Specify dtype option on import or set low_memory=False.
  data=pd.read_csv(r"C:\Users\bhagyashree\Desktop\uber\cab.csv")
```

Out[8]:

	Trip_ID	lpep_pickup_datetime	dropoff_datetime	RatecodeID	PULocationID	DOLocationID	passenger_count	trip_distance	fare_amount	extra	...	cong
0	T0005689460	01-07-2021 00:30	01-07-2021 00:35	1.0	74.0	168.0	1.0	1.20	6.0	0.5	...	
1	T0005689461	01-07-2021 00:25	01-07-2021 01:01	1.0	116.0	265.0	2.0	13.69	42.0	0.5	...	
2	T0005689464	01-07-2021 00:05	01-07-2021 00:12	1.0	97.0	33.0	1.0	0.95	6.5	0.5	...	
3	T0005689465	01-07-2021 00:41	01-07-2021 00:47	1.0	74.0	42.0	1.0	1.24	6.5	0.5	...	
4	T0005689467	01-07-2021 00:51	01-07-2021 00:58	1.0	42.0	244.0	1.0	1.10	7.0	0.5	...	

5 rows × 27 columns

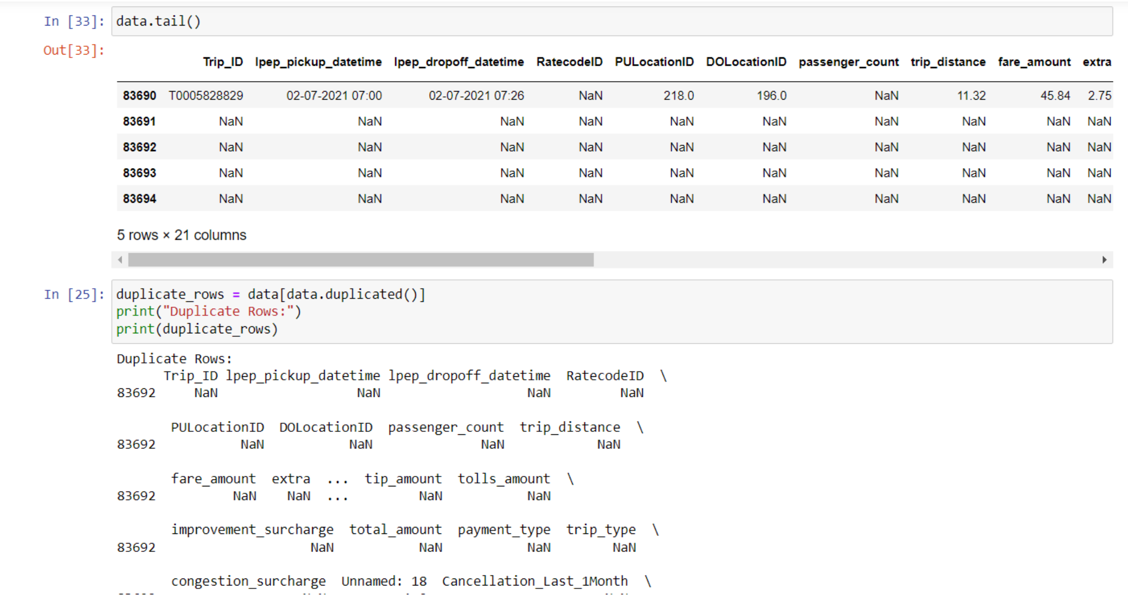


```
In [13]: data.describe()
```

Out[13]:

	RatecodeID	PULocationID	DOLocationID	passenger_count	trip_distance	fare_amount	extra	mta_tax	tip_amount	tolls_amount	improvement_
count	513.000000	513.000000	513.000000	513.000000	513.000000	513.000000	513.000000	513.000000	513.000000	513.000000	5
mean	1.150097	95.368421	127.539961	1.194932	3.682476	15.907212	0.155458	0.483431	1.142222	0.488051	
std	0.733467	59.364561	76.200233	0.843758	5.107039	15.707989	0.542239	0.094880	2.122777	1.750870	
min	1.000000	7.000000	4.000000	0.000000	0.000000	-12.000000	0.000000	-0.500000	0.000000	0.000000	
25%	1.000000	74.000000	68.000000	1.000000	1.080000	7.000000	0.000000	0.500000	0.000000	0.000000	
50%	1.000000	74.000000	129.000000	1.000000	1.890000	10.500000	0.000000	0.500000	0.000000	0.000000	
75%	1.000000	127.000000	185.000000	1.000000	3.900000	17.500000	0.000000	0.500000	2.000000	0.000000	
max	5.000000	264.000000	265.000000	6.000000	37.600000	107.500000	3.250000	0.500000	26.640000	13.100000	

```
In [34]: data.columns

Out[34]: Index(['Trip_ID', 'lpep_pickup_datetime', 'lpep_dropoff_datetime',
                'RatecodeID', 'PULocationID', 'DOLocationID', 'passenger_count',
                'trip_distance', 'fare_amount', 'extra', 'mta_tax', 'tip_amount',
                'tolls_amount', 'improvement_surcharge', 'total_amount', 'payment_type',
                'trip_type', 'congestion_surcharge', 'Unnamed: 18',
                'Cancellation_Last_1Month', 'SubtractionResult'],
               dtype='object')
```



In [33]:
```
data.tail()
```

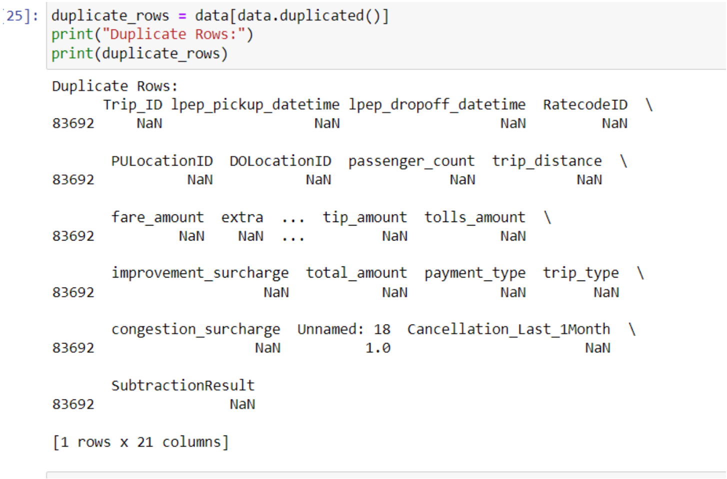
Out[33]:

	Trip_ID	lpep_pickup_datetime	lpep_dropoff_datetime	RatecodeID	PULocationID	DOLocationID	passenger_count	trip_distance	fare_amount	extra
83690	T0005828829	02-07-2021 07:00	02-07-2021 07:26	NaN	218.0	196.0	NaN	11.32	45.84	2.75
83691	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
83692	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
83693	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
83694	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

5 rows × 21 columns

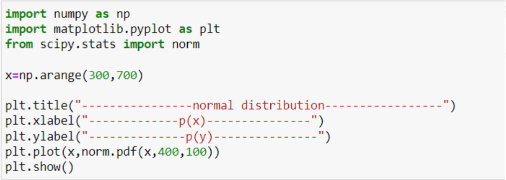
In [25]:
```
duplicate_rows = data[data.duplicated()]
print("Duplicate Rows:")
print(duplicate_rows)
```

```
Duplicate Rows:
      Trip_ID lpep_pickup_datetime lpep_dropoff_datetime  RatecodeID  \
83692     NaN                  NaN                   NaN         NaN

       PULocationID  DOLocationID  passenger_count  trip_distance  \
83692           NaN           NaN              NaN            NaN

       fare_amount  extra  ...  tip_amount  tolls_amount  \
83692          NaN    NaN  ...         NaN           NaN

       improvement_surcharge  total_amount  payment_type  trip_type  \
83692                    NaN           NaN           NaN        NaN

       congestion_surcharge  Unnamed: 18  Cancellation_Last_1Month  \
```

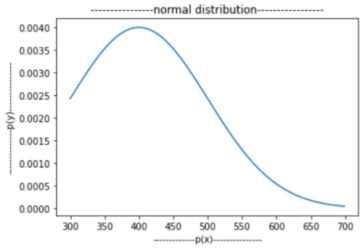


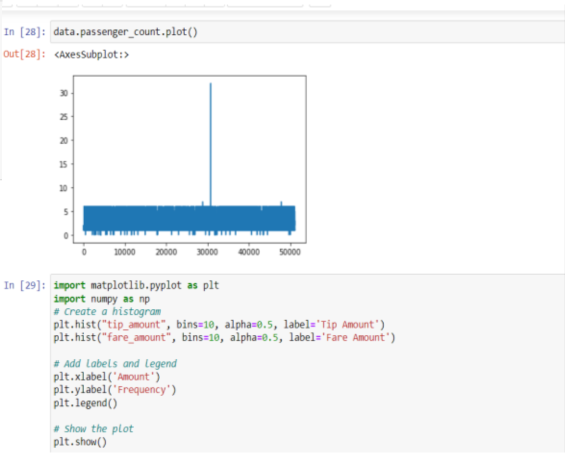
25]:
```
duplicate_rows = data[data.duplicated()]
print("Duplicate Rows:")
print(duplicate_rows)
```

```
Duplicate Rows:
      Trip_ID lpep_pickup_datetime lpep_dropoff_datetime  RatecodeID  \
83692     NaN                  NaN                   NaN         NaN

       PULocationID  DOLocationID  passenger_count  trip_distance  \
83692           NaN           NaN              NaN            NaN

       fare_amount  extra  ...  tip_amount  tolls_amount  \
83692          NaN    NaN  ...         NaN           NaN

       improvement_surcharge  total_amount  payment_type  trip_type  \
83692                    NaN           NaN           NaN        NaN

       congestion_surcharge  Unnamed: 18  Cancellation_Last_1Month  \
83692                   NaN          1.0                       NaN

       SubtractionResult
83692                NaN

[1 rows x 21 columns]
```



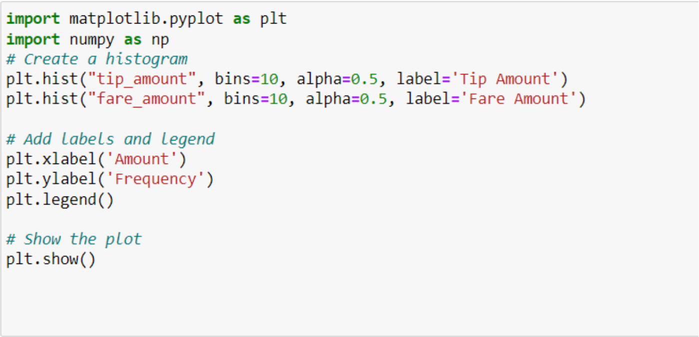
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm

x=np.arange(300,700)

plt.title("----------------normal distribution----------------")
plt.xlabel("-------------p(x)---------------")
plt.ylabel("-------------p(y)---------------")
plt.plot(x,norm.pdf(x,400,100))
plt.show()
```



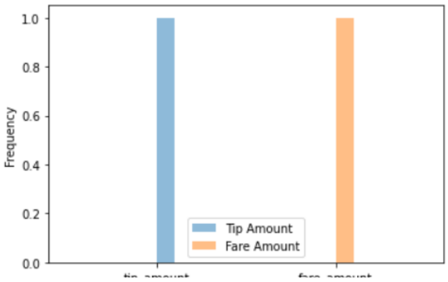


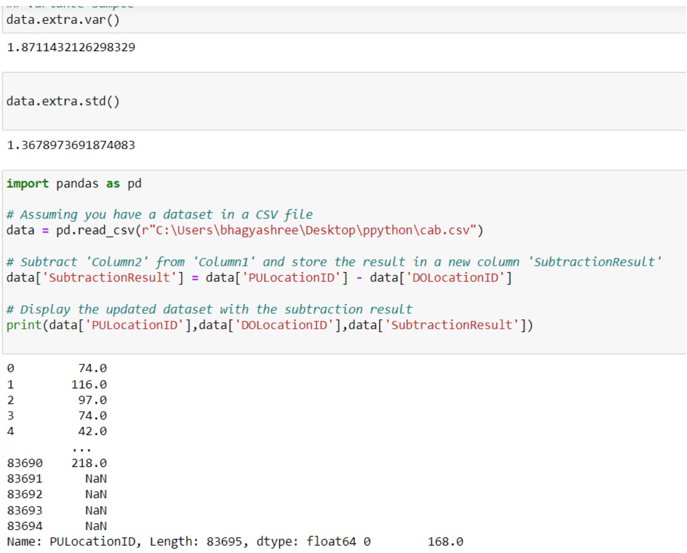
```
In [28]: data.passenger_count.plot()

Out[28]: <AxesSubplot:>
```

```
In [29]: import matplotlib.pyplot as plt
         import numpy as np
         # Create a histogram
         plt.hist("tip_amount", bins=10, alpha=0.5, label='Tip Amount')
         plt.hist("fare_amount", bins=10, alpha=0.5, label='Fare Amount')

         # Add labels and legend
         plt.xlabel('Amount')
         plt.ylabel('Frequency')
         plt.legend()

         # Show the plot
         plt.show()
```



```
import matplotlib.pyplot as plt
import numpy as np
# Create a histogram
plt.hist("tip_amount", bins=10, alpha=0.5, label='Tip Amount')
plt.hist("fare_amount", bins=10, alpha=0.5, label='Fare Amount')

# Add labels and legend
plt.xlabel('Amount')
plt.ylabel('Frequency')
plt.legend()

# Show the plot
plt.show()
```





```
## variance sample
data.extra.var()
```

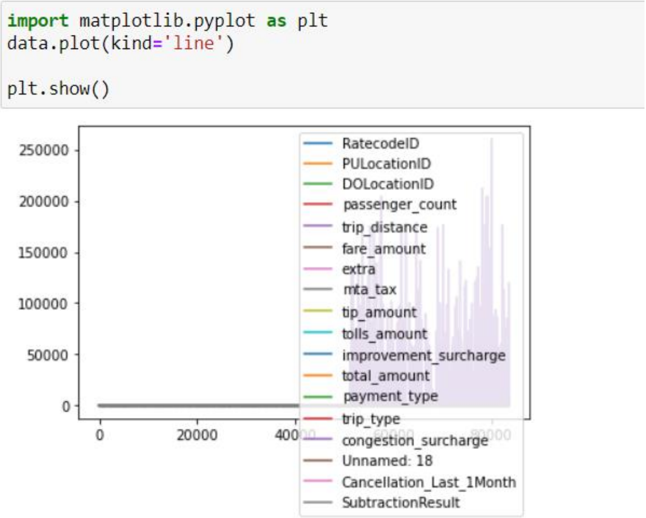
```
1.8711432126298329
```

```
data.extra.std()
```

```
1.3678973691874083
```

```
import pandas as pd

# Assuming you have a dataset in a CSV file
data = pd.read_csv(r"C:\Users\bhagyashree\Desktop\ppython\cab.csv")

# Subtract 'Column2' from 'Column1' and store the result in a new column 'SubtractionResult'
data['SubtractionResult'] = data['PULocationID'] - data['DOLocationID']

# Display the updated dataset with the subtraction result
print(data['PULocationID'],data['DOLocationID'],data['SubtractionResult'])
```

```
0         74.0
1        116.0
2         97.0
3         74.0
4         42.0
         ...  
83690    218.0
83691      NaN
83692      NaN
83693      NaN
83694      NaN
Name: PULocationID, Length: 83695, dtype: float64 0        168.0
```

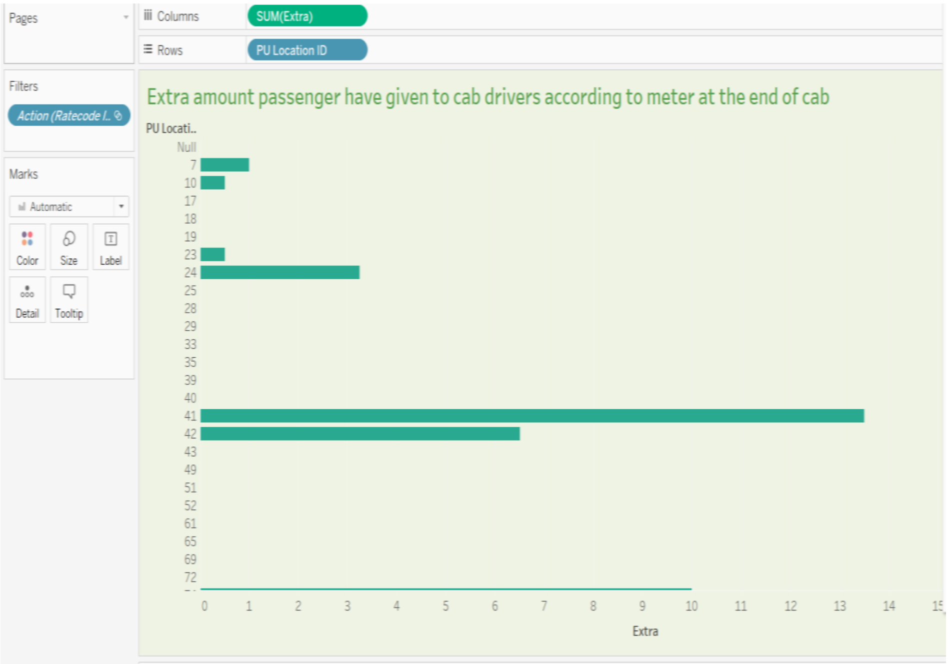


```
import matplotlib.pyplot as plt
data.plot(kind='line')

plt.show()
```

CHARTS

1) <u>Fare optimization</u>

Bar graph



Columns: SUM(Extra)

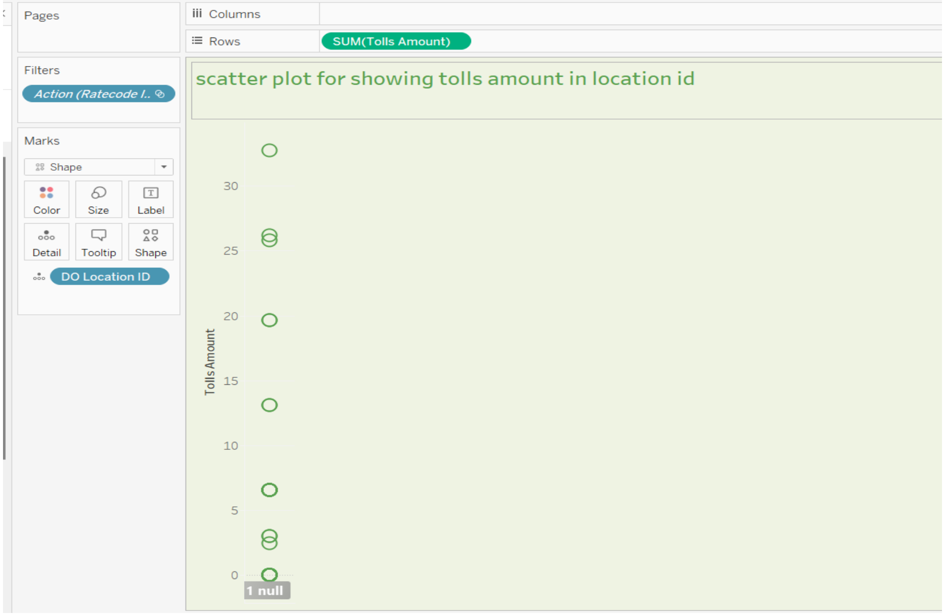
Rows: PU Location ID

Filters: Action (Ratecode I..

Marks: Automatic

Extra amount passenger have given to cab drivers according to meter at the end of cab

Scatter plot



Columns:

Rows: SUM(Tolls Amount)

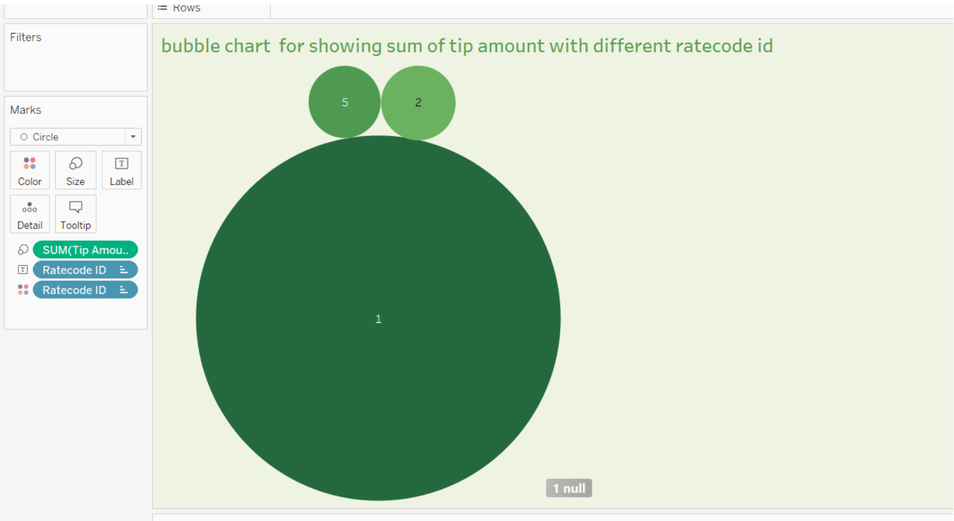
Filters: Action (Ratecode I..

Marks: Shape

Detail: DO Location ID

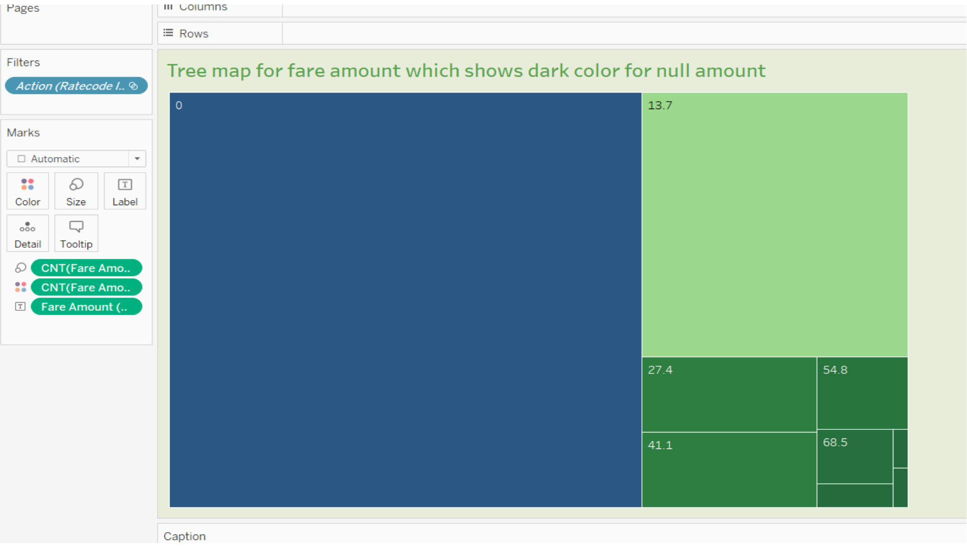
scatter plot for showing tolls amount in location id

Bubble chart



bubble chart for showing sum of tip amount with different ratecode id

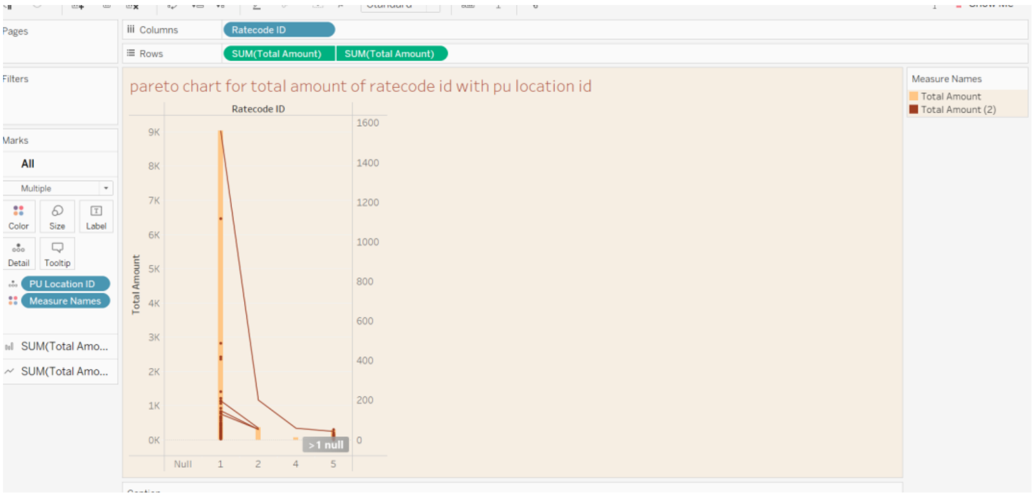
Tree map



Tree map for fare amount which shows dark color for null amount

2)Route Optimization

Pareto chart



Columns: Ratecode ID

Rows: SUM(Total Amount) SUM(Total Amount)

pareto chart for total amount of ratecode id with pu location id

Marks: All, Multiple

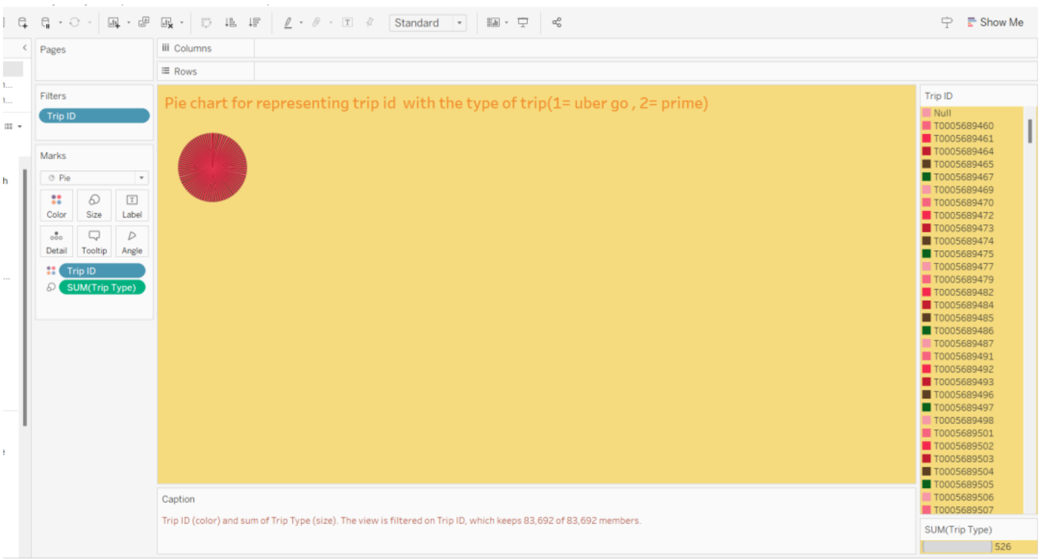
PU Location ID

Measure Names

SUM(Total Amo...

SUM(Total Amo...

Pie chart



Pie chart for representing trip id with the type of trip(1= uber go , 2= prime)

Filters: Trip ID

Marks: Pie

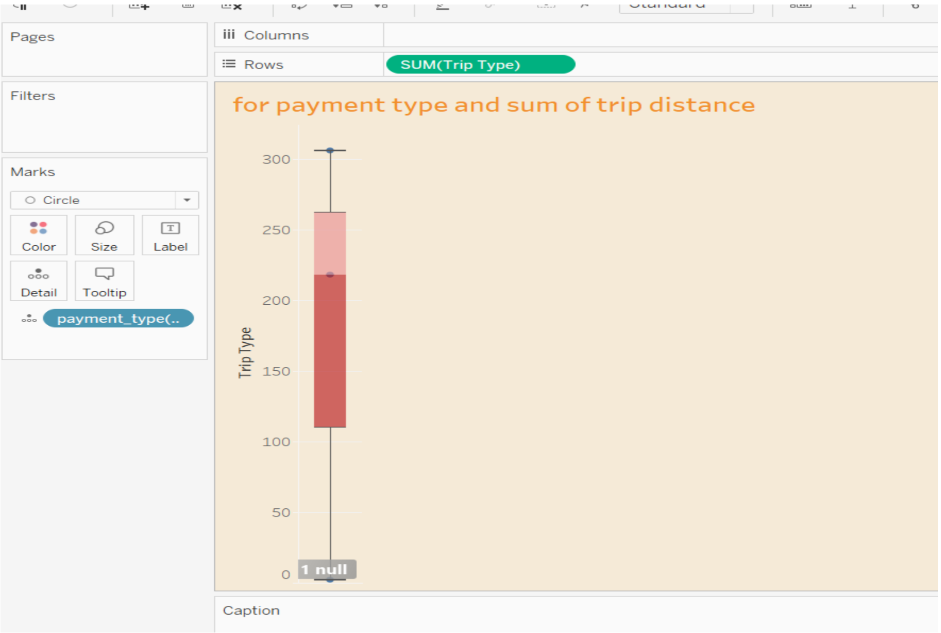
Trip ID

SUM(Trip Type)

Caption

Trip ID (color) and sum of Trip Type (size). The view is filtered on Trip ID, which keeps 83,692 of 83,692 members.

Box plot



Columns

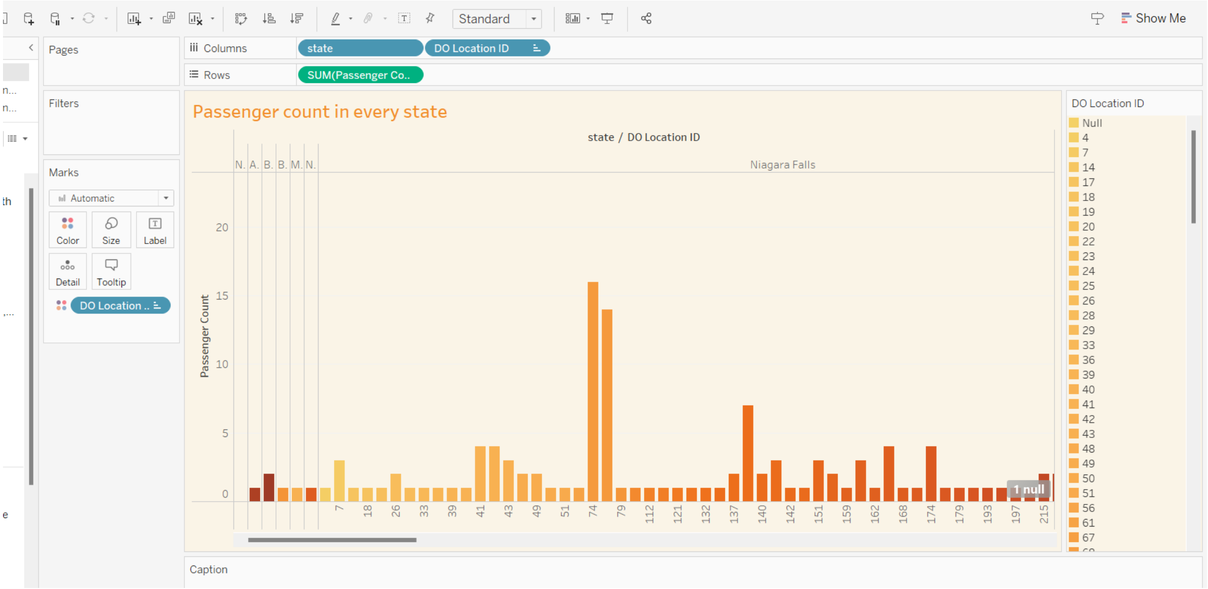
Rows: SUM(Trip Type)

Marks: Circle

Detail: payment_type(..

3)Passenger Optimization

Stacked bar chart



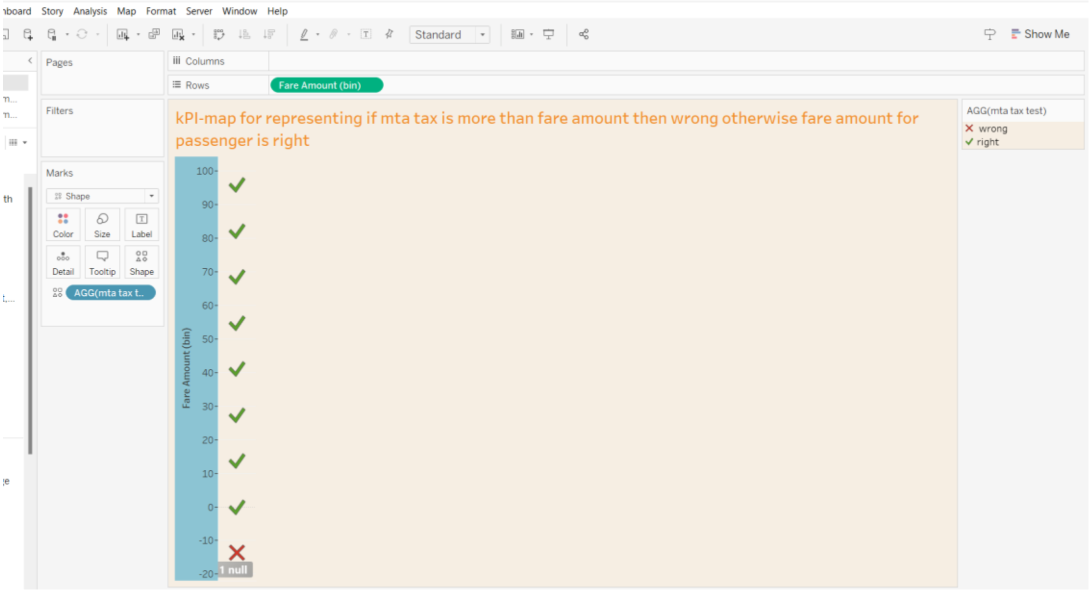
Columns: state, DO Location ID

Rows: SUM(Passenger Co..

Marks: Automatic

Color: DO Location ..

KPI



Columns

Rows: Fare Amount (bin)

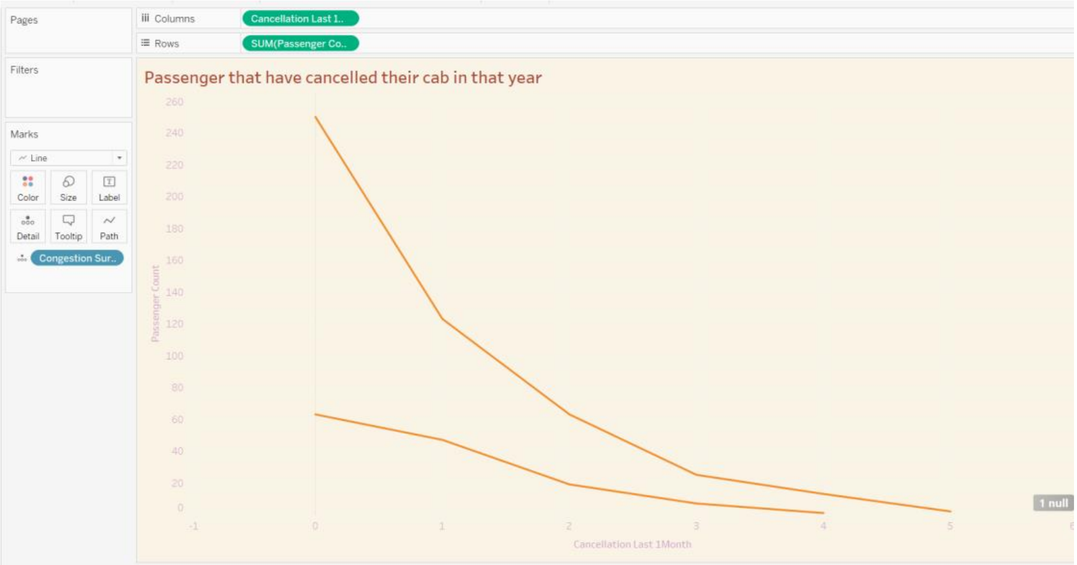
kPI-map for representing if mta tax is more than fare amount then wrong otherwise fare amount for passenger is right

AGG(mta tax test)

✗ wrong

✓ right

Line chart



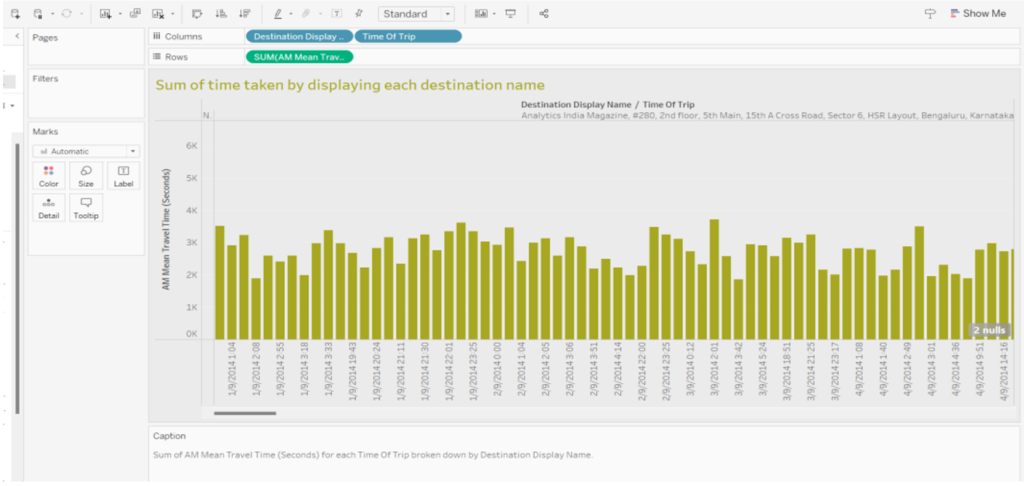
Columns: Cancellation Last 1..

Rows: SUM(Passenger Co..

Passenger that have cancelled their cab in that year

4)Traffic optimization

Histogram



Columns: Destination Display .. | Time Of Trip

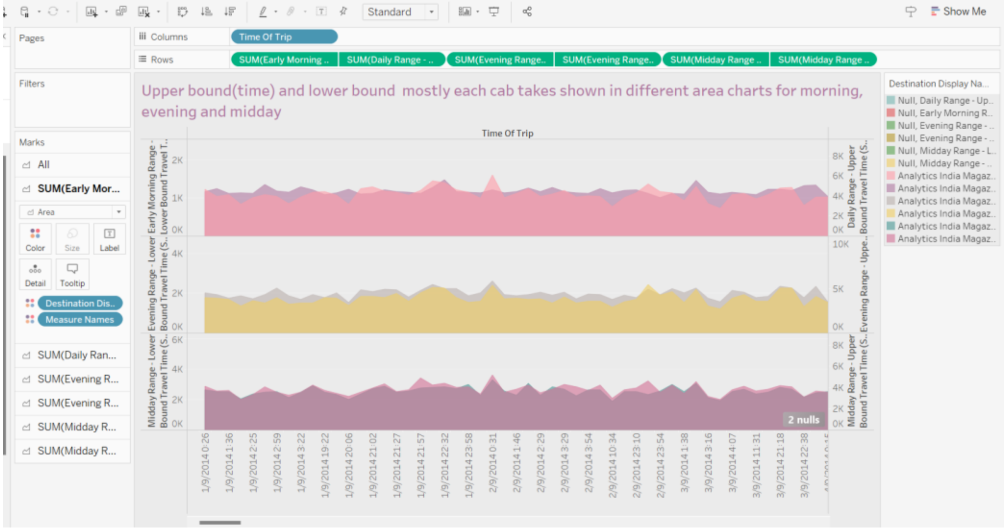
Rows: SUM(AM Mean Trav..

Sum of time taken by displaying each destination name

Caption

Sum of AM Mean Travel Time (Seconds) for each Time Of Trip broken down by Destination Display Name.

Area chart

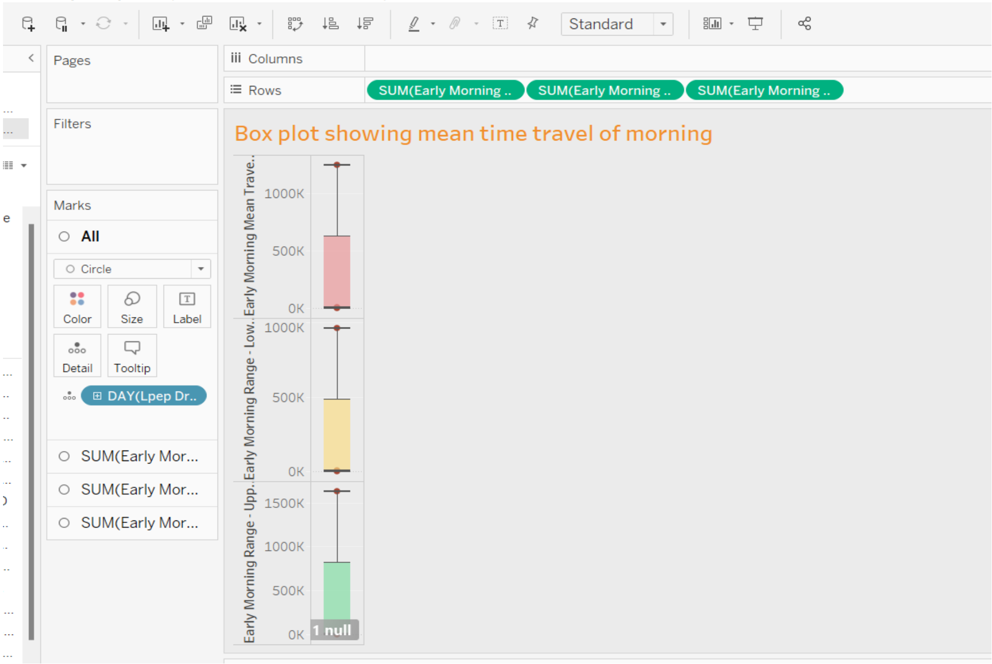


Columns: Time Of Trip

Rows: SUM(Early Morning .. | SUM(Daily Range - .. | SUM(Evening Range.. | SUM(Evening Range.. | SUM(Midday Range .. | SUM(Midday Range ..

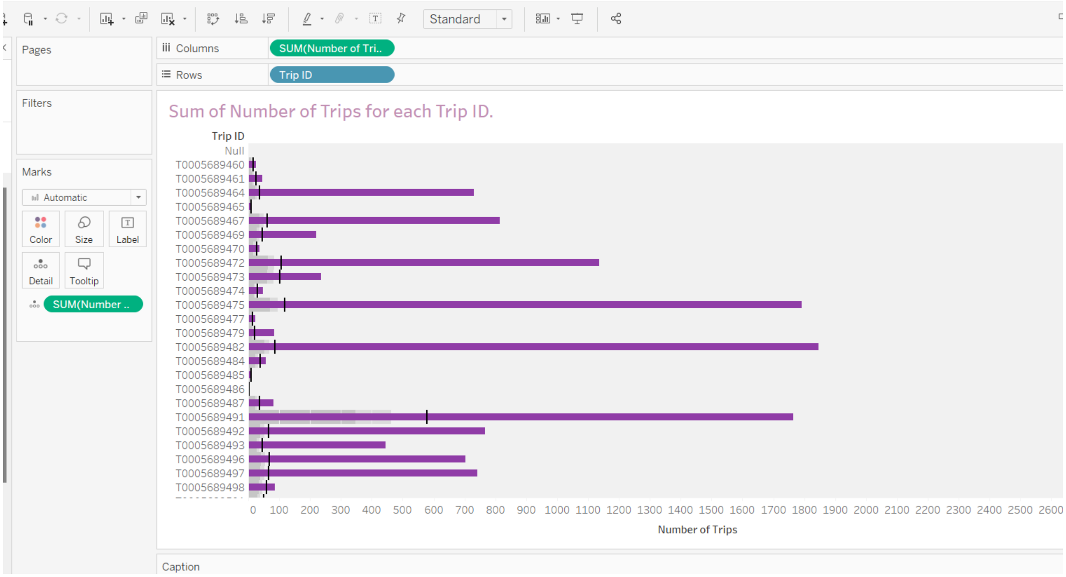
Upper bound(time) and lower bound mostly each cab takes shown in different area charts for morning, evening and midday

Box plot

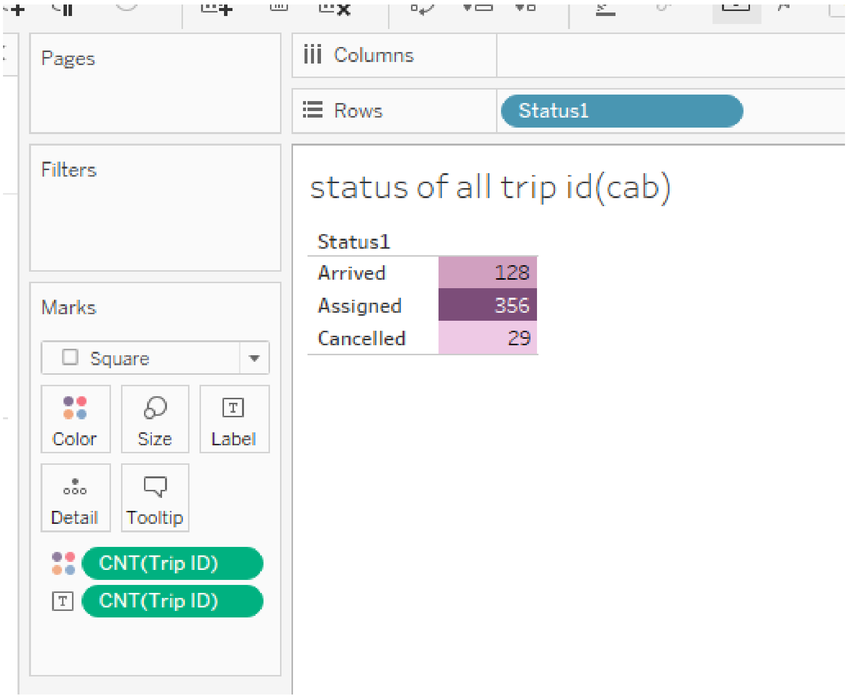


5) Cab availability

Bullet graph



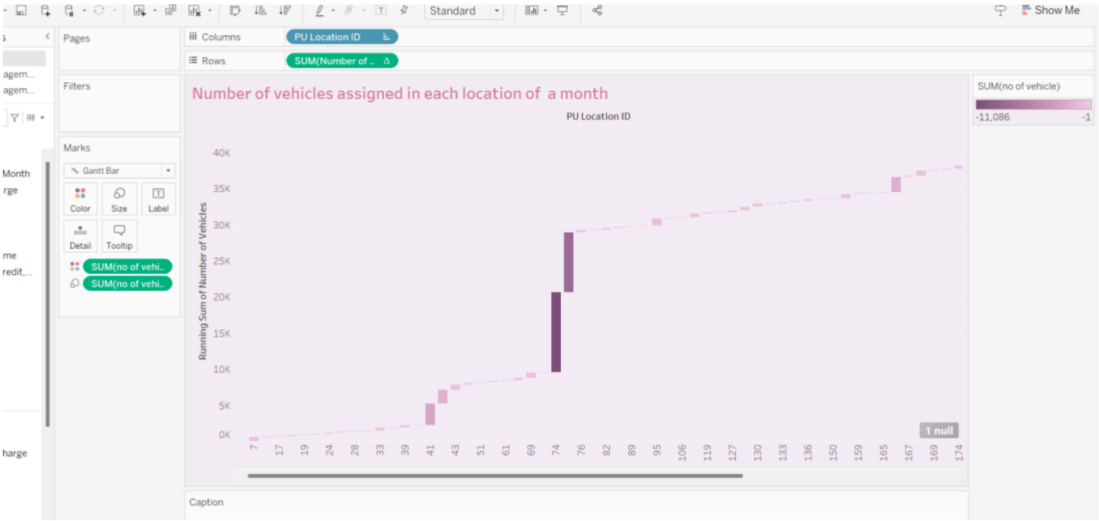
Text chart



status of all trip id(cab)

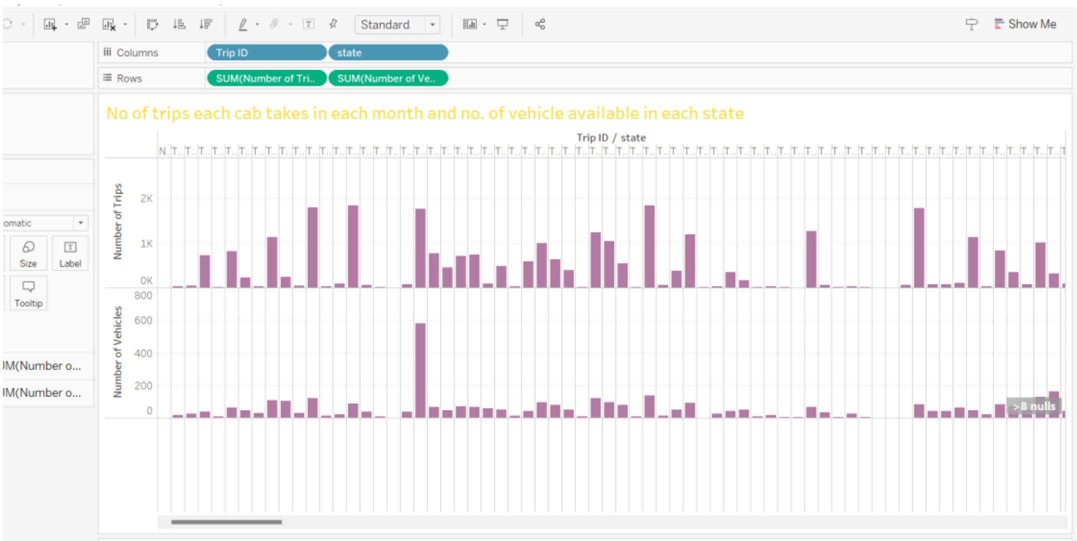
Status1	
Arrived	128
Assigned	356
Cancelled	29

Waterfall chart



Number of vehicles assigned in each location of a month

Bar graph

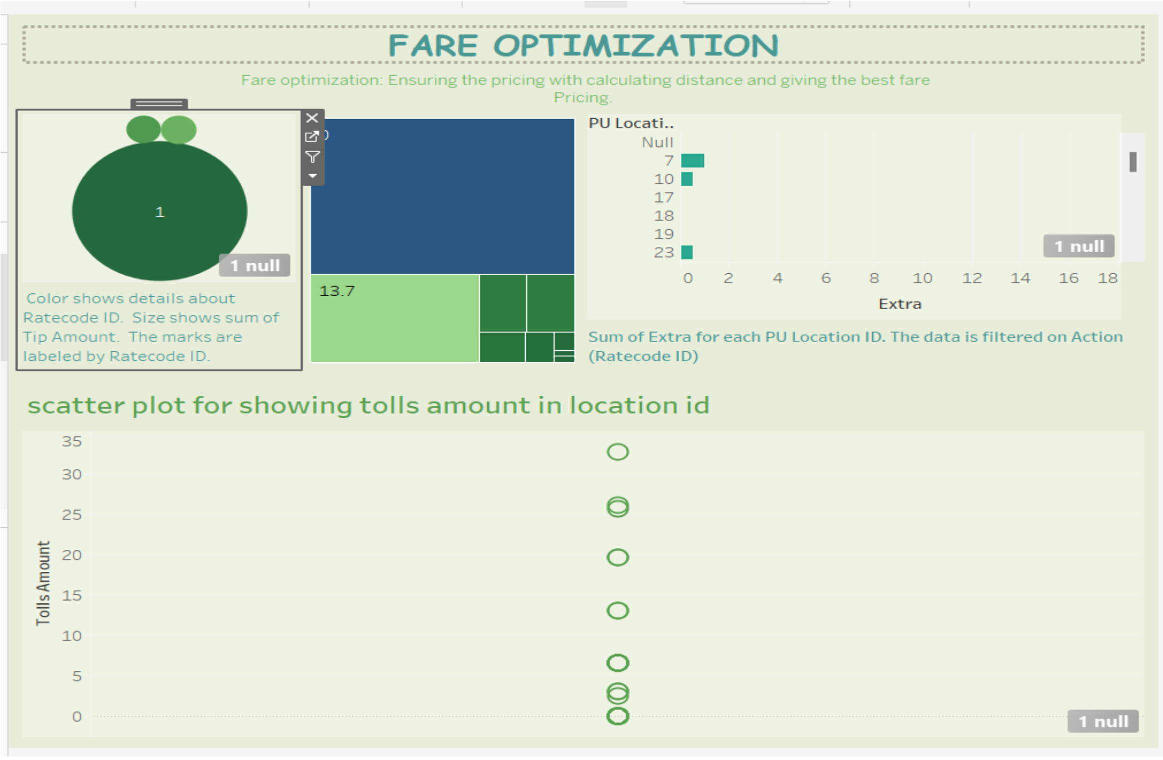


Columns: Trip ID, state

Rows: SUM(Number of Tri.., SUM(Number of Ve..

No of trips each cab takes in each month and no. of vehicle available in each state

Dashboards



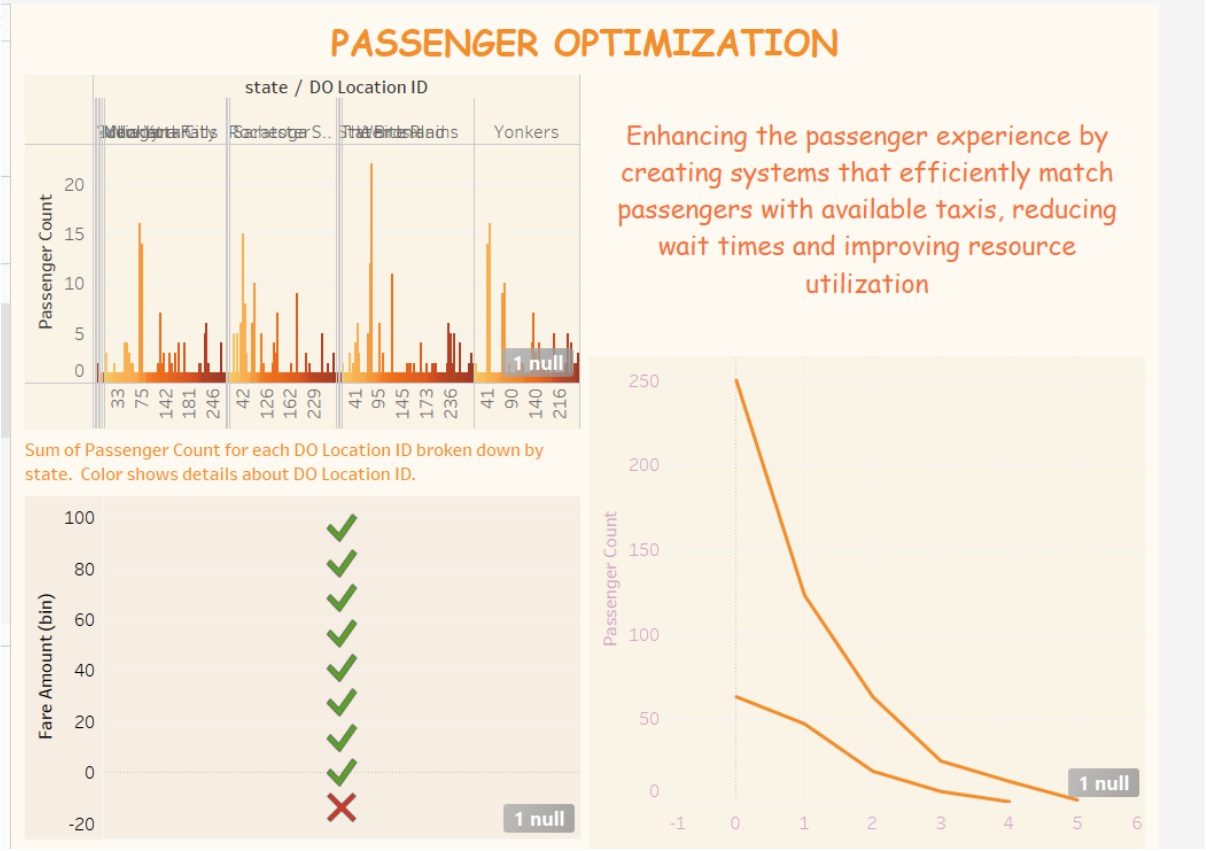
FARE OPTIMIZATION

Fare optimization: Ensuring the pricing with calculating distance and giving the best fare Pricing.

Color shows details about Ratecode ID. Size shows sum of Tip Amount. The marks are labeled by Ratecode ID.

Sum of Extra for each PU Location ID. The data is filtered on Action (Ratecode ID)

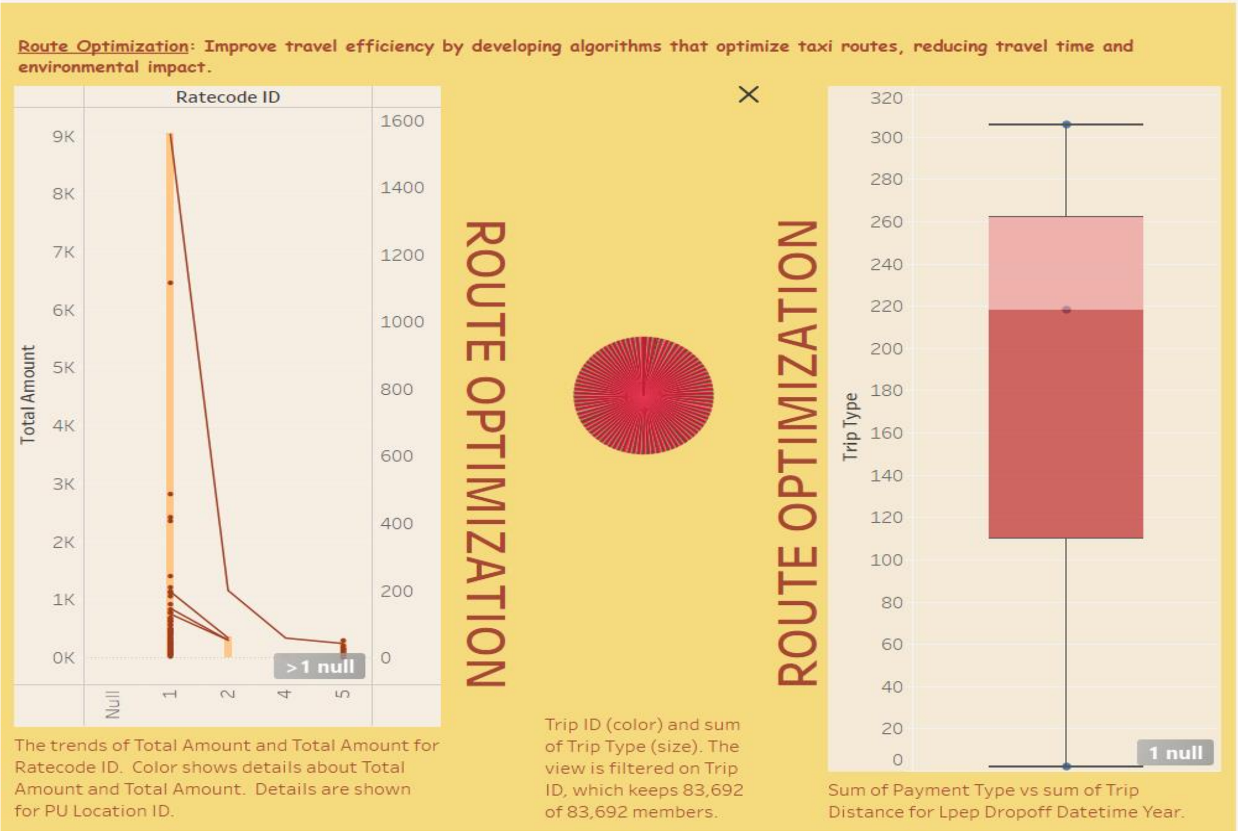
scatter plot for showing tolls amount in location id



PASSENGER OPTIMIZATION

Enhancing the passenger experience by creating systems that efficiently match passengers with available taxis, reducing wait times and improving resource utilization

Sum of Passenger Count for each DO Location ID broken down by state. Color shows details about DO Location ID.

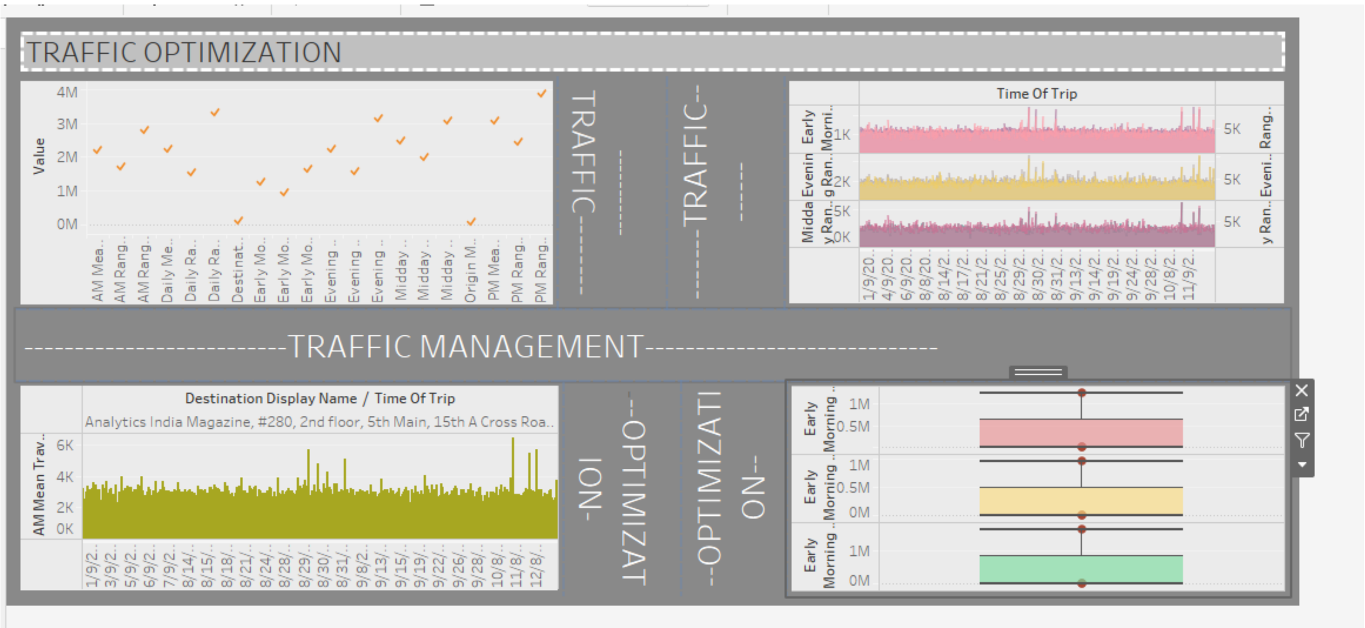


Route Optimization: Improve travel efficiency by developing algorithms that optimize taxi routes, reducing travel time and environmental impact.

The trends of Total Amount and Total Amount for Ratecode ID. Color shows details about Total Amount and Total Amount. Details are shown for PU Location ID.

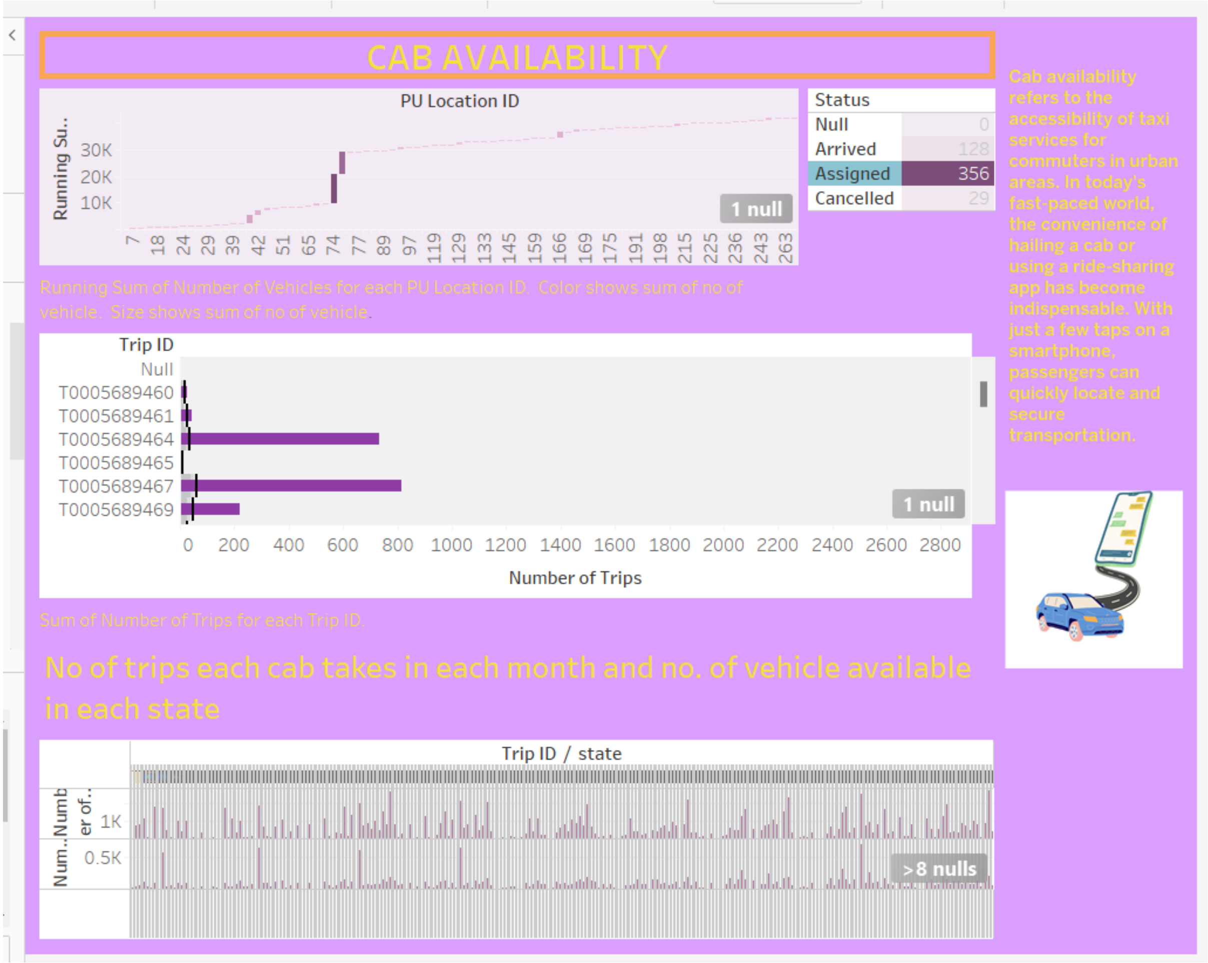
Trip ID (color) and sum of Trip Type (size). The view is filtered on Trip ID, which keeps 83,692 of 83,692 members.

Sum of Payment Type vs sum of Trip Distance for Lpep Dropoff Datetime Year.



TRAFFIC OPTIMIZATION

---------------------------TRAFFIC MANAGEMENT-----------------------------



CAB AVAILABILITY

Status	
Null	0
Arrived	128
Assigned	356
Cancelled	29

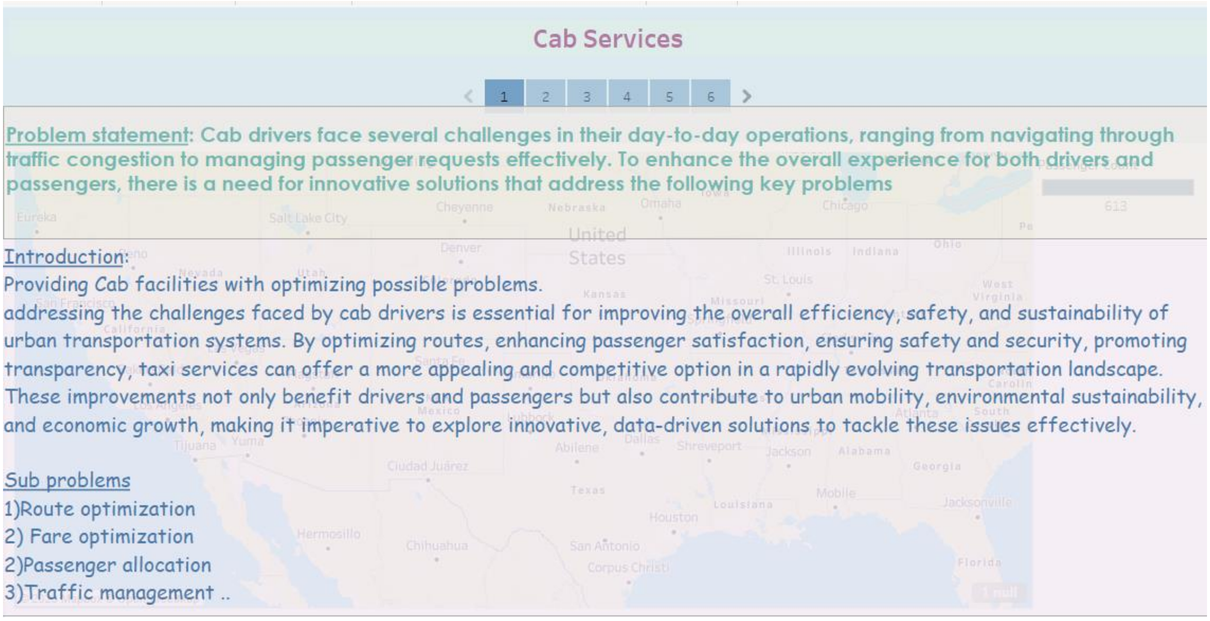
Running Sum of Number of Vehicles for each PU Location ID. Color shows sum of no of vehicle. Size shows sum of no of vehicle.

Sum of Number of Trips for each Trip ID.

Cab availability refers to the accessibility of taxi services for commuters in urban areas. In today's fast-paced world, the convenience of hailing a cab or using a ride-sharing app has become indispensable. With just a few taps on a smartphone, passengers can quickly locate and secure transportation.

No of trips each cab takes in each month and no. of vehicle available in each state

STORY

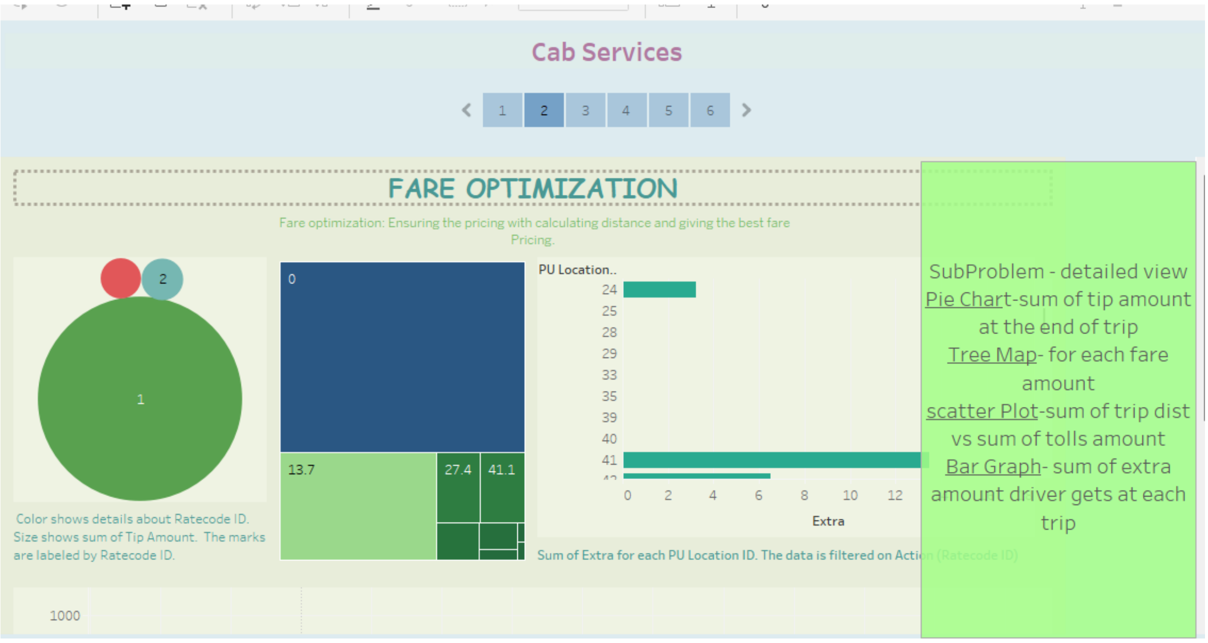


Cab Services

Problem statement: Cab drivers face several challenges in their day-to-day operations, ranging from navigating through traffic congestion to managing passenger requests effectively. To enhance the overall experience for both drivers and passengers, there is a need for innovative solutions that address the following key problems

Introduction:
Providing Cab facilities with optimizing possible problems.
addressing the challenges faced by cab drivers is essential for improving the overall efficiency, safety, and sustainability of urban transportation systems. By optimizing routes, enhancing passenger satisfaction, ensuring safety and security, promoting transparency, taxi services can offer a more appealing and competitive option in a rapidly evolving transportation landscape. These improvements not only benefit drivers and passengers but also contribute to urban mobility, environmental sustainability, and economic growth, making it imperative to explore innovative, data-driven solutions to tackle these issues effectively.

Sub problems
1)Route optimization
2) Fare optimization
2)Passenger allocation
3)Traffic management ..



Cab Services

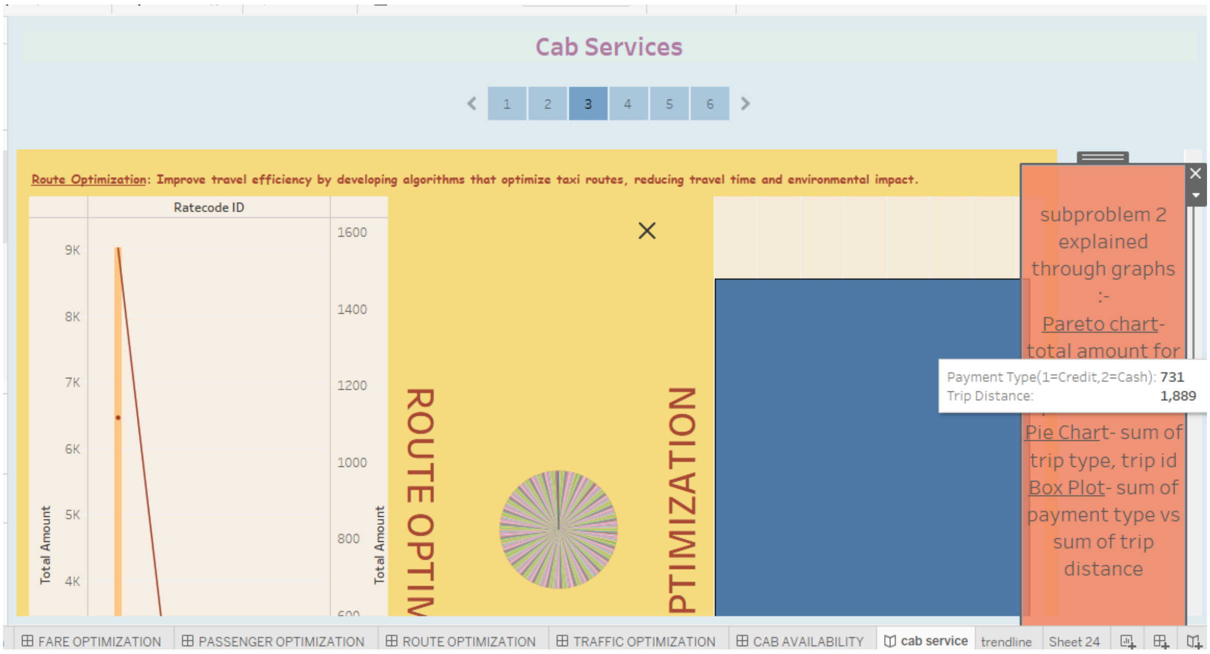
FARE OPTIMIZATION

Fare optimization: Ensuring the pricing with calculating distance and giving the best fare Pricing.

Color shows details about Ratecode ID. Size shows sum of Tip Amount. The marks are labeled by Ratecode ID.

Sum of Extra for each PU Location ID. The data is filtered on Acti

SubProblem - detailed view
Pie Chart-sum of tip amount at the end of trip
Tree Map- for each fare amount
scatter Plot-sum of trip dist vs sum of tolls amount
Bar Graph- sum of extra amount driver gets at each trip

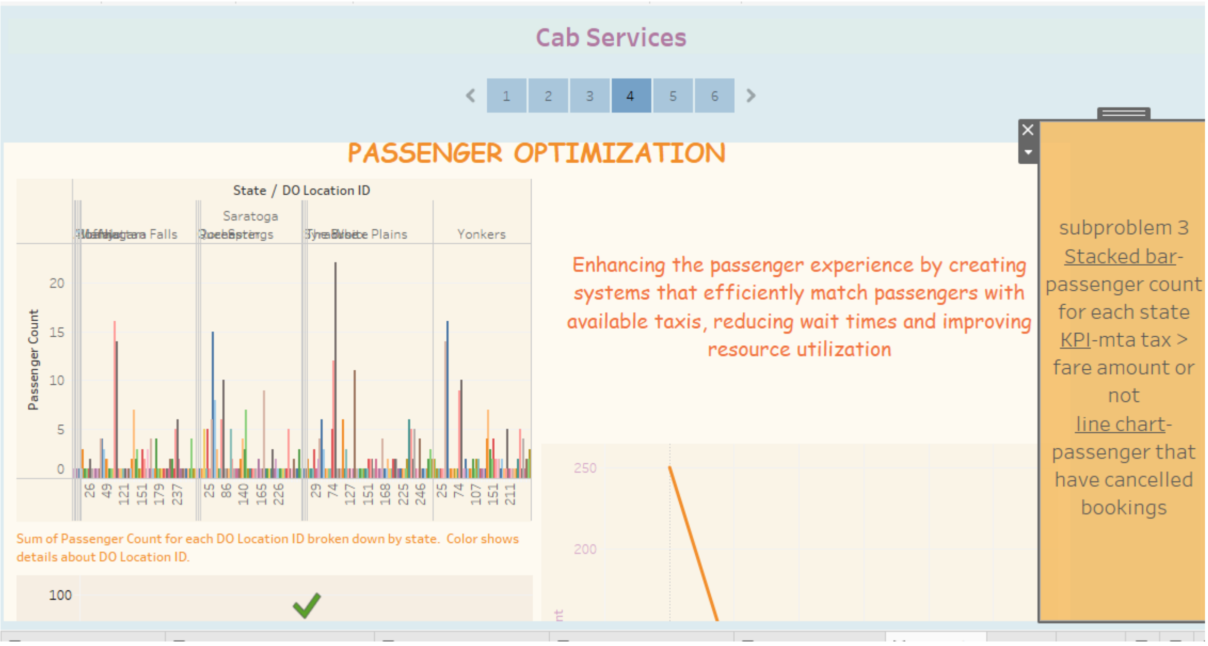


Cab Services

Route Optimization: Improve travel efficiency by developing algorithms that optimize taxi routes, reducing travel time and environmental impact.

subproblem 2 explained through graphs :-
Pareto chart- total amount for
Pie Chart- sum of trip type, trip id
Box Plot- sum of payment type vs sum of trip distance

Payment Type(1=Credit,2=Cash): 731
Trip Distance: 1,889



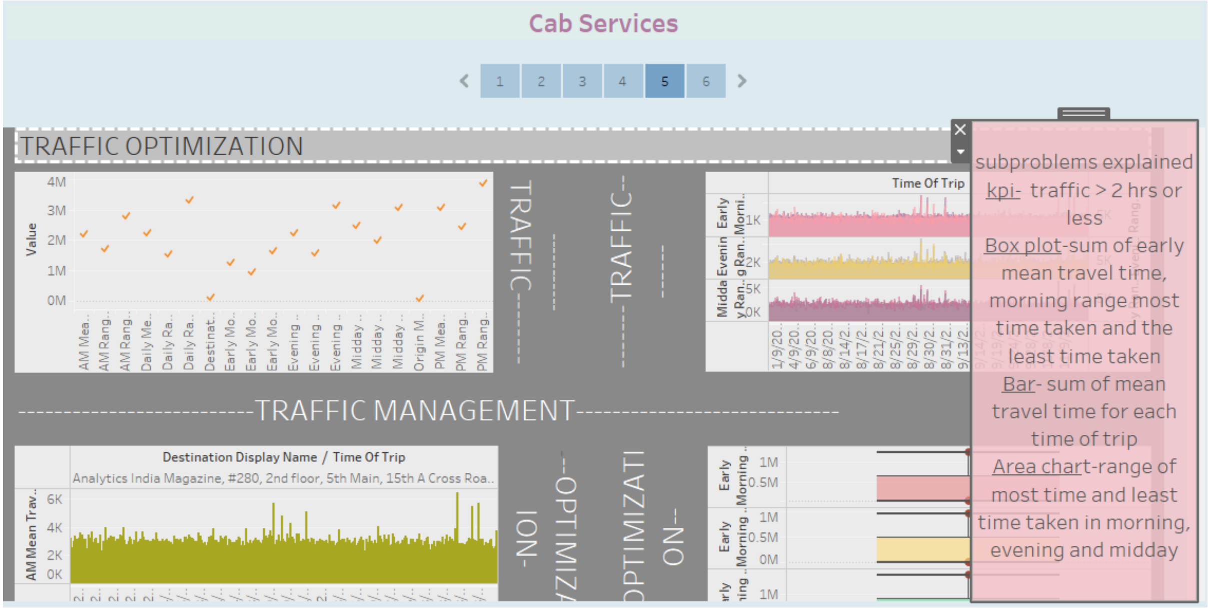
Cab Services

PASSENGER OPTIMIZATION

Sum of Passenger Count for each DO Location ID broken down by state. Color shows details about DO Location ID.

Enhancing the passenger experience by creating systems that efficiently match passengers with available taxis, reducing wait times and improving resource utilization

subproblem 3
Stacked bar- passenger count for each state
KPI-mta tax > fare amount or not
line chart- passenger that have cancelled bookings



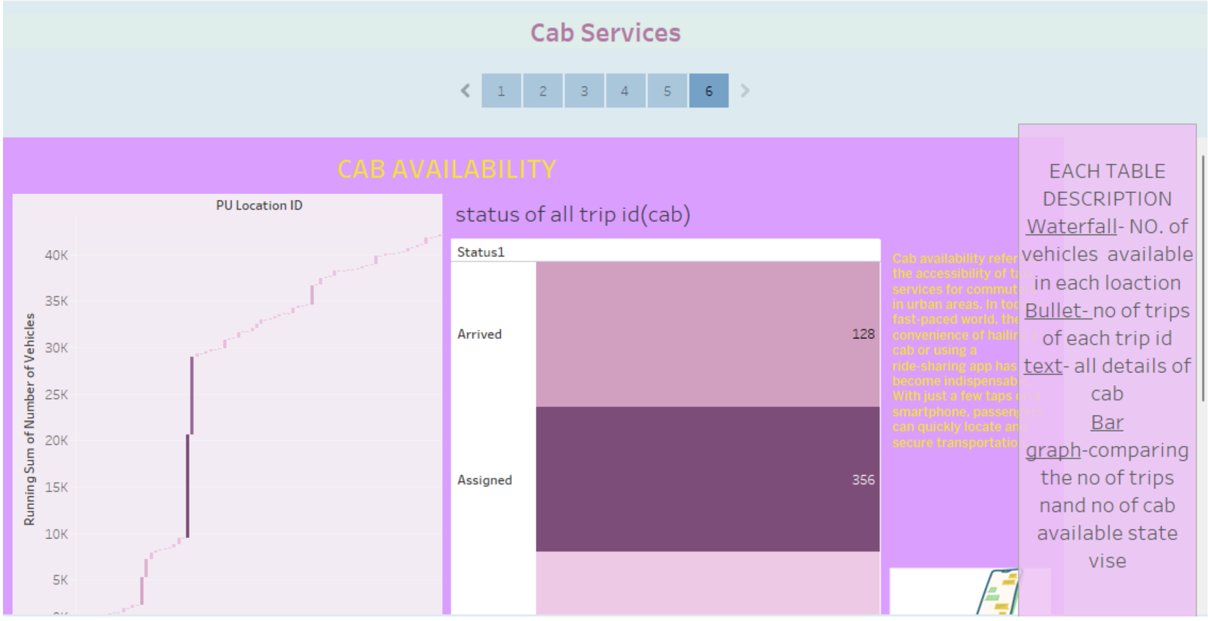
subproblems explained

kpi- traffic > 2 hrs or less

Box plot-sum of early mean travel time, morning range most time taken and the least time taken

Bar- sum of mean travel time for each time of trip

Area chart-range of most time and least time taken in morning, evening and midday



Cab availability refers to the accessibility of taxi services for commuters in urban areas. In today's fast-paced world, the convenience of hailing a cab or using a ride-sharing app has become indispensable. With just a few taps on a smartphone, passengers can quickly locate and secure transportation.

EACH TABLE DESCRIPTION

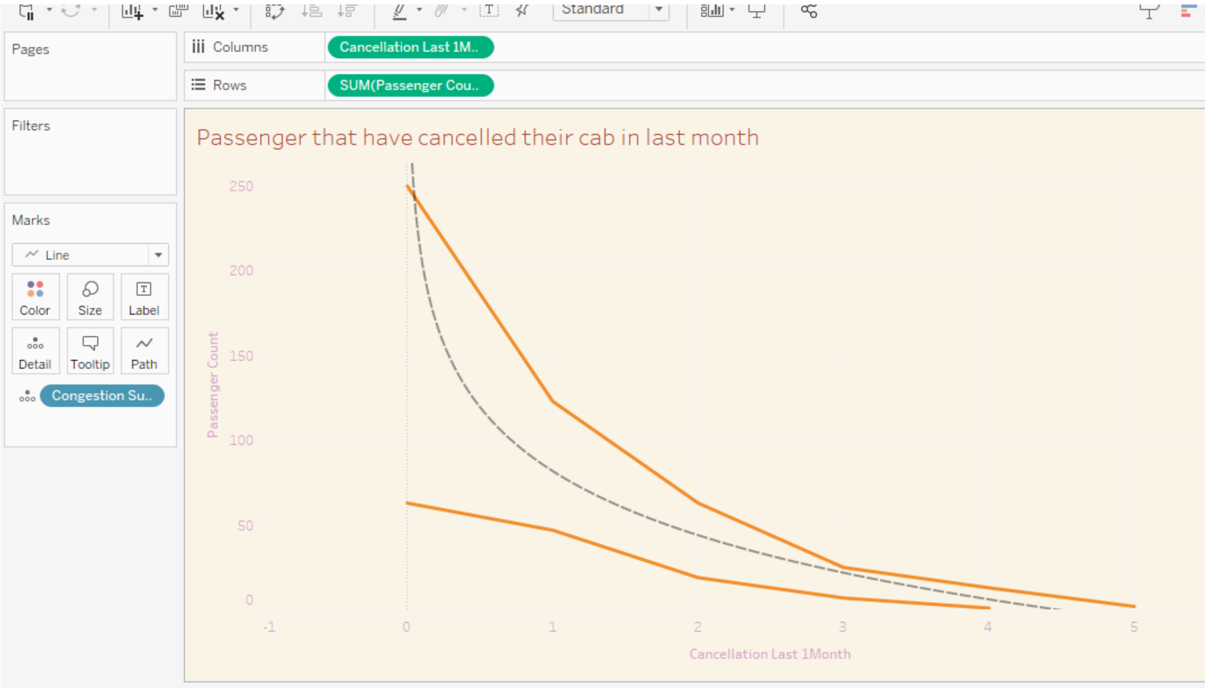
Waterfall- NO. of vehicles available in each loaction

Bullet- no of trips of each trip id

text- all details of cab

Bar graph-comparing the no of trips nand no of cab available state vise

Trendline


Columns: Cancellation Last 1M..

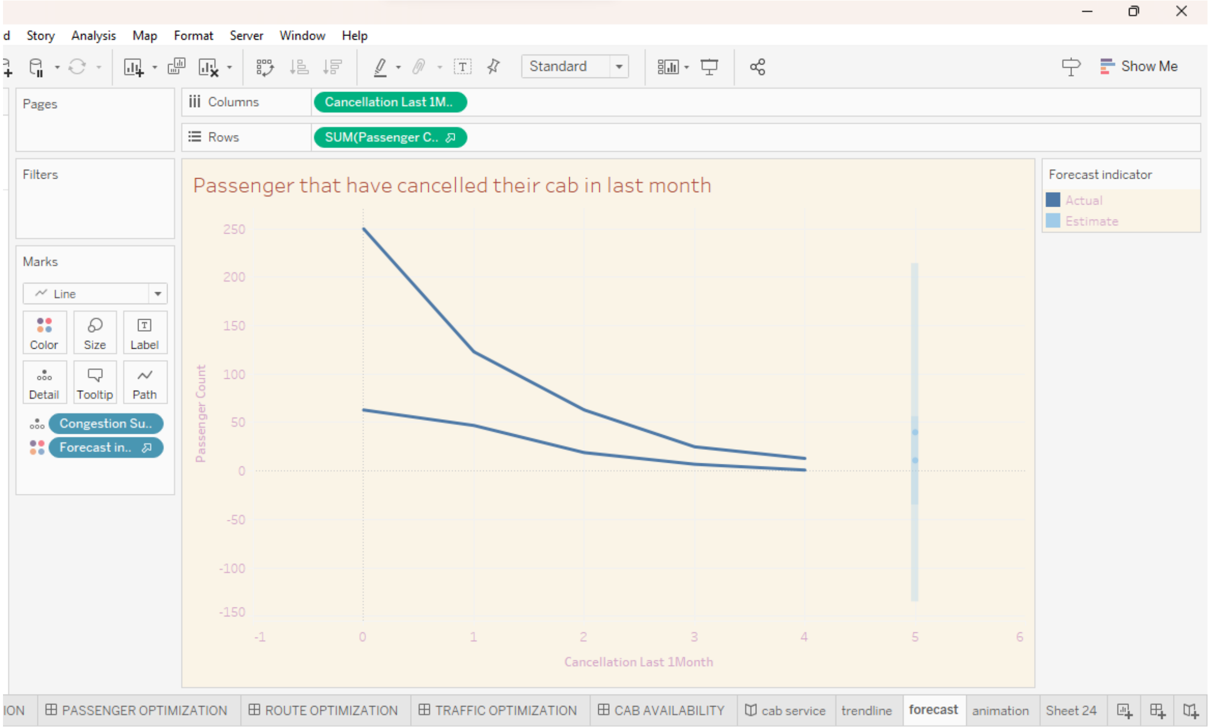
Rows: SUM(Passenger Cou..

Marks: Line

Detail: Congestion Su..

Passenger that have cancelled their cab in last month

Forecasting


Columns: Cancellation Last 1M..

Rows: SUM(Passenger C..

Marks: Line

Detail: Congestion Su..

Color: Forecast in..

Passenger that have cancelled their cab in last month

Forecast indicator

- Actual
- Estimate