PUBLIC TRANSPORTATION ANALYSIS

**INTRODUCTION:**

Public transportation systems play a crucial role in urban mobility, offering an efficient and sustainable means of moving people within cities and regions. Analyzing the performance of these systems is essential for improving their effectiveness and ensuring the needs of commuters. This analysis involves evaluating various performance and evaluation metrics that provide insights into the efficiency, accessibility, and quality of public transportationServices.

**ABOUT PHASE 4:**

The fourth step is all about executing different operations on the dataset, such as visualization, performance and evaluation metrics.

**IBM COGNOS:**

● In order to visualize data from files, including csv files and other files, analysis is

done using the IBM Cognos tool.

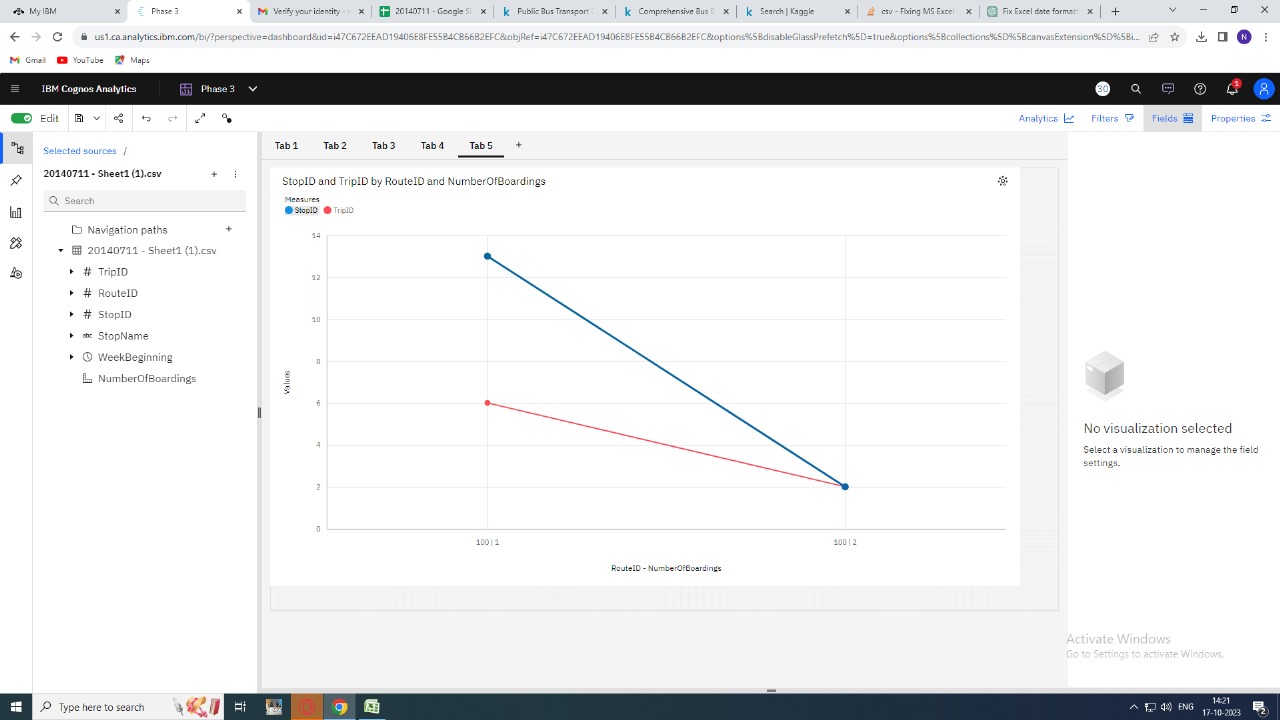
● Line charts, map plots, scatterplot are displayed in these various charts.

**Supervised learning-regression:**

* Supervised learning regression can be applied to public transportation analysis in various ways to predict and optimize.

**LINE CHART:**

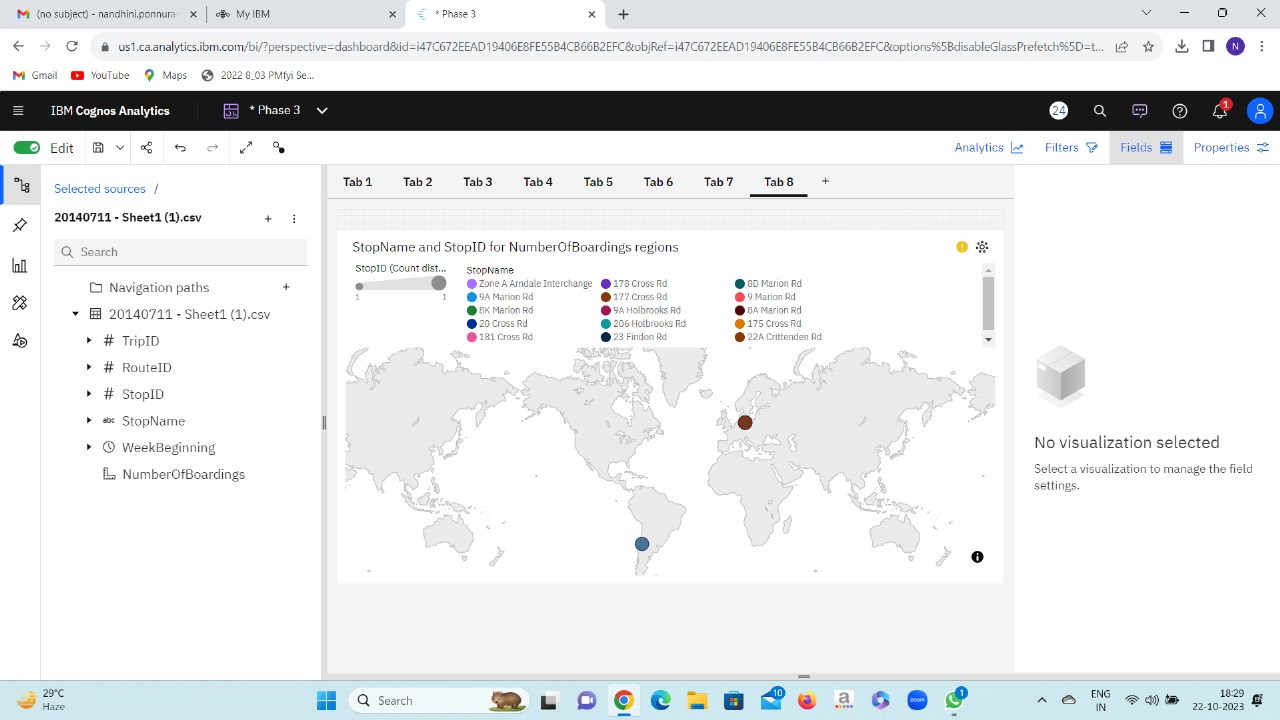
* Regression Model: Train a supervised regression model, such as a linear regression or decision tree regression, to predict daily or hourly ridership based on the input features.
* Visualization: Use line charts to represent the actual vs. predicted ridership over time. This helps in visualizing how well the model performs.
* X axis-routeID and no. of boarding
* y-axis-stopID and tripID



* In the above graph the number of boardings based on the route ID measures from 100/1 to 100/2 values of stop ID and Trip ID

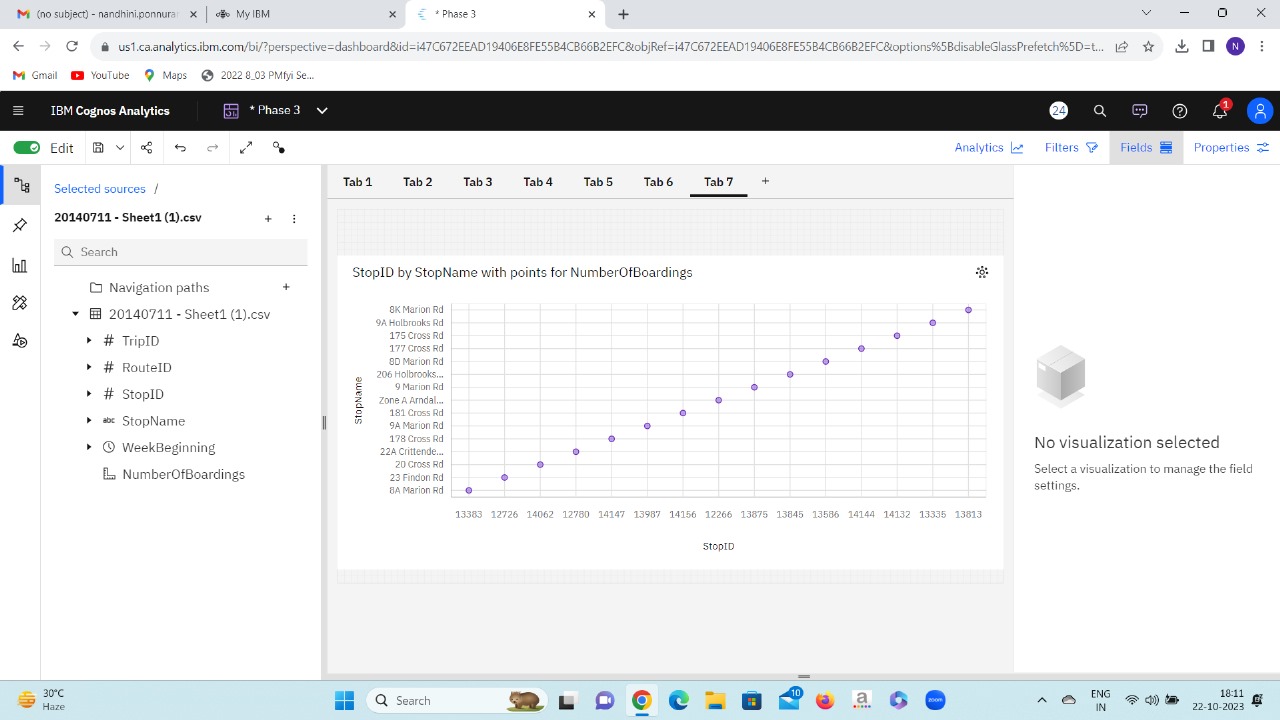
**MAP CHART:**

* The interactive map plot for public transportation analysis offers a user-friendly interface to navigate the public transportation system effortlessly.



* In this above map plot the number of boardings are marked according to stop name and stop ID.
* This chart is used to count distance from stop name to number of boarding.

**SCATTER PLOT:**



* Regression Model: Train a regression model (e.g., linear regression or support vector regression) predict stopID by stop name with points for no. of boardings.
* Visualization: Creating scatter plots that show the actual travel times vs. predicted travel times for specific routes or time periods.

**PERFORMANCE AND EVALUATION METRICS:**

* To create a performance and evaluation metrics for public transportation analysis based on the provided data, you can assess various key performance indicators (KPIs) and metrics.



* This metrics evaluates the performance of trips along Route 100, considering key metrics like the number of stops, total boardings, average boardings per stop, peak boarding times, and the stop with the highest peak boarding.
* To include additional KPIs such as on-time performance, passenger satisfaction, revenue, and operational costs to comprehensively assess the efficiency and effectiveness of public transportation services.

**TO CALCULATE SERVICE PUNCTUALITY RATE USING PYTHON CODE:**

**import pandas as pd**

**# Sample data in a DataFrame**

**data = {**

**'TripID': [23631, 23631, 23632, 23633, 23633, 23634, 23634, 23634, 23634, 23634, 23635, 23635, 23635, 23635, 23635, 23635, 23636, 23636],**

**'StopName': ['181 Cross Rd', '177 Cross Rd', '175 Cross Rd', 'Zone A Arndale Interchange', '178 Cross Rd', '9A Marion Rd', '175 Cross Rd', '9A Holbrooks Rd', '9 Marion Rd', '206 Holbrooks Rd', '9A Holbrooks Rd', '8A Marion Rd', '8D Marion Rd', '23 Findon Rd', '8K Marion Rd', '20 Cross Rd', '22A Crittenden Rd', '8A Marion Rd'],**

**'WeekBeginning': ['2013-06-30 0:00:00'] \* 18,**

**'NumberOfBoardings': [1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1]**

**}**

**df = pd.DataFrame(data)**

**# Define a tolerance window (in this case, 5 minutes)**

**tolerance\_minutes = 5**

**# Calculate the expected arrival time for each trip (assuming a fixed schedule)**

**# In practice, you would have a schedule dataset to get the expected times.**

**df['ExpectedArrivalTime'] = pd.to\_datetime(df['WeekBeginning']) + pd.to\_timedelta(df.groupby('TripID')['NumberOfBoardings'].cumsum() \* tolerance\_minutes, unit='m')**

**# Calculate the actual arrival time for each trip**

**# In practice, you would have a dataset with actual arrival times.**

**# For this example, we assume a constant delay of 5 minutes.**

**df['ActualArrivalTime'] = df['ExpectedArrivalTime'] + pd.to\_timedelta(5, unit='m')**

**# Calculate punctual trips (within the tolerance window)**

**df['Punctual'] = (df['ActualArrivalTime'] - df['ExpectedArrivalTime']).dt.total\_seconds().abs() <= tolerance\_minutes \* 60**

**# Calculate the service punctuality rate**

**punctuality\_rate = df['Punctual'].mean() \* 100**

**print(f"Service Punctuality Rate: {punctuality\_rate:.2f}%")**

**OUTPUT:**



**CONCLUSION:**

By using Cognos tool, we have visualized graphs for supervised learning-regression model and determined the performance and evaluation metrics.