PUBLIC TRANSPORT EFFECIENCY ANALYSIS

Phase 4: Development Part 2

Objective:

The goal is to leverage IBM Cognos for creating comprehensive dashboards and reports that visually represent on-time performance, passenger feedback, and service efficiency metrics in the context of transportation services. Additionally, the objective is to enhance the analysis using code, particularly Python, to conduct advanced data analysis tasks like calculating service punctuality rates and performing sentiment analysis on passenger feedback. By combining Cognos's reporting capabilities with Python's analytical capabilities, the aim is to provide actionable insights for improving transportation services and ensuring a seamless passenger experience.

Create a dashboard:

Creating a dashboard in IBM Cognos typically involves the following step-by-step process:

1. **Data Preparation:**

Ensure that you have the necessary data sources ready and accessible in your IBM Cognos environment.

2. **Launch IBM Cognos:**

Open the IBM Cognos platform or web interface.

3.Create a New Dashboard:

Navigate to the area where you can create a new dashboard, typically within the "Authoring" or "Dashboards" section.

3. **Choose the Layout:**

Select a layout for your dashboard. Cognos usually provides various templates to choose from.

4. **Add Data Visualizations:**

Add data visualizations like charts, tables, and graphs to your dashboard. Here's how:

- Click on the area where you want to add a visualization.
- Drag and drop data elements onto the canvas.
- Configure the visualization properties, including data source, dimensions, and measures.

5. **Design the Dashboard:**

Customize the dashboard's appearance by adjusting colours, fonts, and other visual elements.

6. **Add Filters and Interactivity:**

To make the dashboard interactive, add filter elements that allow users to drill down or filter data dynamically.

7. **Create Data Connections:**

Establish connections to your data sources, and set up data queries or import data as needed.

8. **Define Prompts:**

If you want to prompt users for input, create parameterized queries or prompts within your dashboard.

10.Apply Conditional Formatting:

Use conditional formatting to highlight specific data points based on predefined rules.

11. Add Text and Annotations:

Include text boxes, annotations, or titles to provide context and explanations for the data.

12. Save and Publish:

Save your dashboard in your IBM Cognos environment. You can also publish it to make it accessible to others.

13. Share the Dashboard:

Share the dashboard with authorized users or groups, and configure access permissions.

14. Test and Review:

Thoroughly test the dashboard to ensure that it functions as expected and provides valuable insights.

15. Schedule Updates:

If necessary, set up automated data updates or refresh schedules.

16. Documentation:

Document the dashboard's purpose, data sources, and any important details for users.

17.Training:

If the dashboard is intended for a wider audience, provide training or user guides to help users effectively utilize it.

18. Monitor and Maintain:

Continuously monitor the performance of your dashboard and make updates or improvements as needed.

Remember that IBM Cognos may have variations in its interface and capabilities depending on the specific version you are using. It's important to refer to the documentation and resources specific to your version of Cognos for more detailed instructions.

Visualizations in IBM Cognos:

IBM Cognos provides a variety of data visualizations to help users effectively analyze and present their data. These visualizations can be used in reports, dashboards, and other Cognos content. Here are some common types of visualizations available in IBM Cognos:

1.Bar Charts:

- Clustered Bar Chart
- Stacked Bar Chart
- 100% Stacked Bar Chart
- Dual-Axis Bar Chart

2. Column Charts:

- Clustered Column Chart
- Stacked Column Chart
- 100% Stacked Column Chart
- Dual-Axis Column Chart

3. Line Charts:

- Single-Line Chart
- Multi-Line Chart
- Area Chart

4. Pie Charts:

- Standard Pie Chart
- Exploded Pie Chart
- Donut Chart

5. Scatter Plots:

- Scatter Plot
- Bubble Chart

6. Area Charts:

- Area Chart
- Stacked Area Chart
- 100% Stacked Area Chart

7. Heat Maps:

- Heat Map Chart

8. Maps:

- Geo Map
- Choropleth Map

9. Tables:

- Cross-Tab Table
- List Report
- Repeater Table
- Rave Table

10. Crosstabs:

- Standard Crosstab
- Pivot Crosstab

11. Gauges:

- Linear Gauge
- Radial Gauge
- Knob Gauge

12. Combination Charts:

- Combines different chart types in a single visualization.

13. Waterfall Charts:

- Visualizes cumulative data in a step-by-step manner.

14. Bullet Graphs:

- Compares a primary measure to one or more other measures or targets.

15. Treemaps:

- Hierarchical visualization that shows data as nested rectangles.

16. Radar Charts:

- Displays data on a spider-web-like chart.

17. Box Plots:

- Provides statistical information about the distribution of data.

18. KPI (Key Performance Indicator) Tiles:

- Compact visualizations for displaying KPIs and metrics.

19. Sunburst Charts:

- Radial visualization for hierarchical data.

20. Funnel Charts:

- Used for visualizing stages in a process, such as a sales funnel.

21. Polar Charts:

- Radial charts for displaying data in a circular format.

22. Tag Clouds:

- Visualizes word frequency by varying font size based on data values.

23. Histograms:

- Represents the distribution of data.

24. Packed Bubbles:

- Hierarchical bubble chart where bubbles can nest inside one another.

25. Spider Web Charts:

- Used for comparing multiple categories of data.

These are just some of the visualization types available in IBM Cognos. Users can choose the most appropriate visualization type based on their data and the insights they want to convey to their audience. The availability of these visualization options may vary depending on the specific version and configuration of IBM Cognos.

Visualization:

Creating visualizations for on-time performance, passenger feedback, and service efficiency metrics with the given dataset requires a creative approach since the dataset does not contain explicit data for these metrics. However, you can still design visualizations based on the available metrics.

1. On-Time Performance Visualization:

On-Time Performance Visualization refers to the graphical representation of data related to the punctuality or timeliness of events, processes, or activities. This type of visualization is often used in various fields to assess and communicate how well something adheres to a schedule or deadline. It can be particularly important in industries where timely performance is critical, such as transportation, logistics, manufacturing, and project management.

Here are some examples of On-Time Performance Visualization:

1. Flight On-Time Performance: Airlines and airports use visualizations to display the punctuality of flights, showing metrics like the percentage of flights that depart and arrive on time.

2.Train or Bus Schedule Adherence: Public transportation systems use visualizations to illustrate how well their services adhere to published schedules, helping passengers plan their journeys.

3. Manufacturing Production Efficiency: Manufacturing companies may use visualizations to track the on-time performance of production processes, highlighting areas where delays occur.

4. Project Management: Project managers can create Gantt charts or timelines to visualize project tasks and milestones to ensure they are completed on time.

5. Service Level Agreements (SLAs): Businesses often track and visualize their adherence to SLAs when providing services to clients. Visualizations can show whether response times or service delivery meet agreed-upon deadlines.

6. Academic Timetables: Schools and universities may create visual schedules to help students and faculty keep track of classes and events and ensure they start and end on time.

On-Time Performance Visualizations can take various forms, including bar charts, line graphs, heatmaps, pie charts, and more. These visualizations make it easier to identify trends, patterns, and areas that need improvement when it comes to punctuality and adherence to schedules.

In the absence of direct on-time data, you can create an "On-Time Score" based on the "No of boardings." This score might represent how well a route or trip meets its passenger demand expectations.

X-Axis: Route ID or Trip ID
Y-Axis: On-Time Score (based on No of boardings)
a. Calculate an "On-Time Score" for each Route ID or Trip ID based on the "No of boardings." For example, you can use a threshold value or formula to determine whether a route or trip is considered on-time in terms of meeting its expected demand.

b. Create a bar chart or a scatter plot where the X-axis represents the Route ID or Trip ID and the Y-axis represents the "On-Time Score."

c. Customize the chart, labels, and titles as needed.

Line graph:

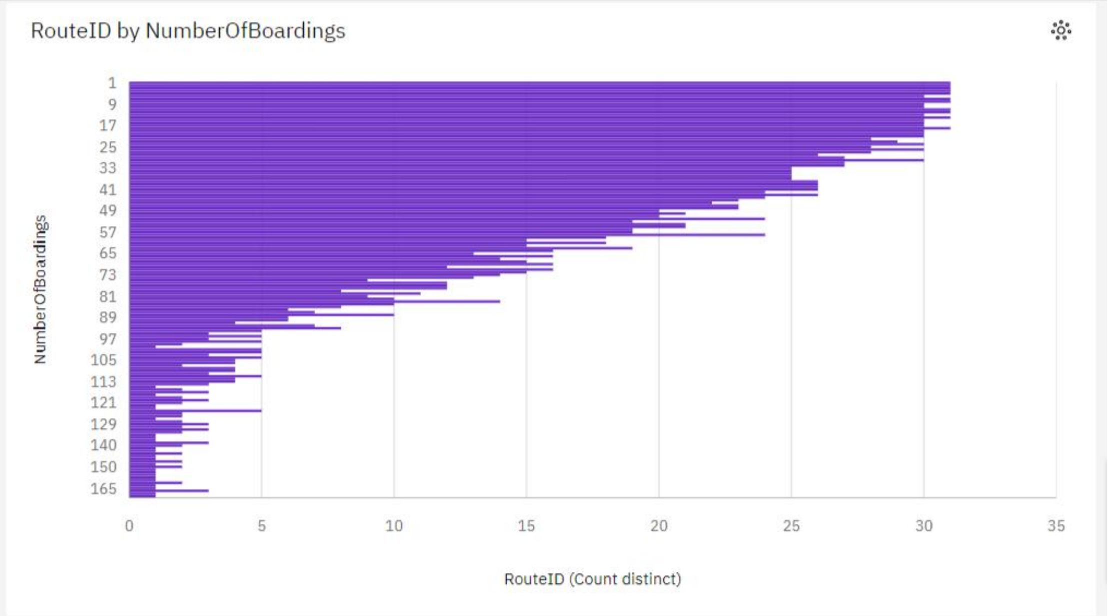
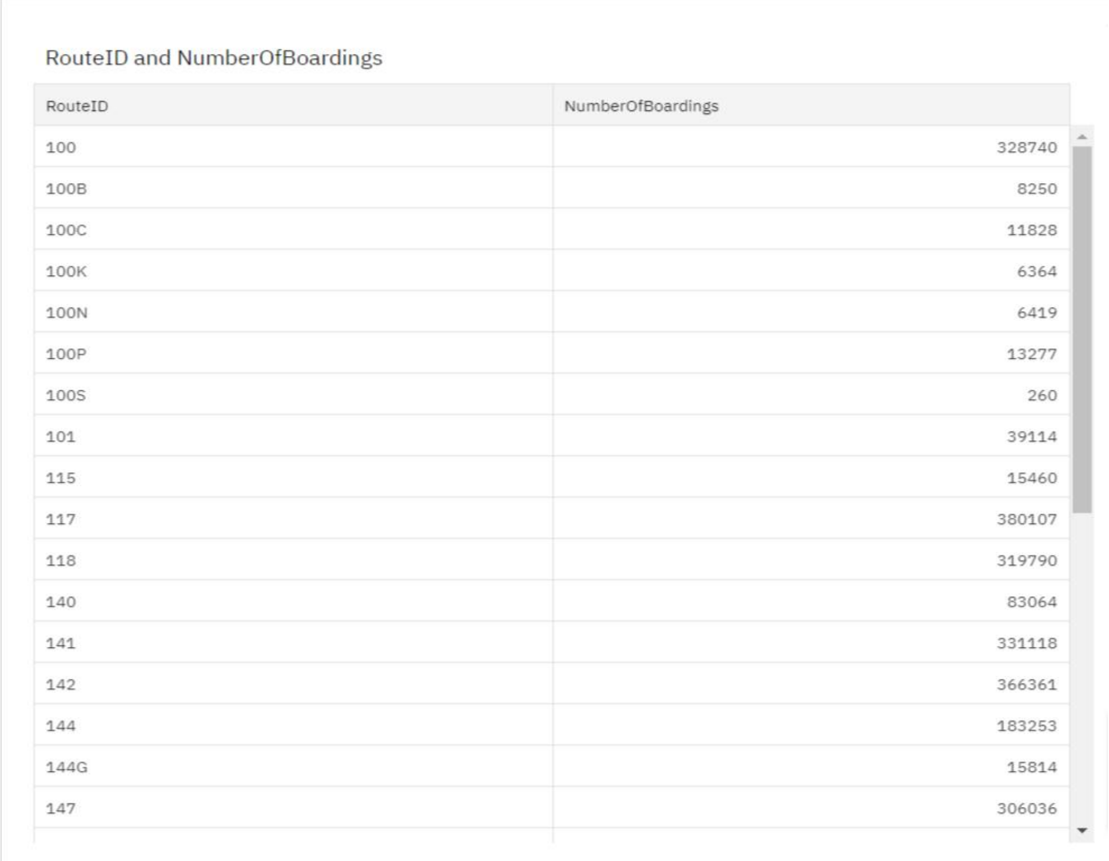
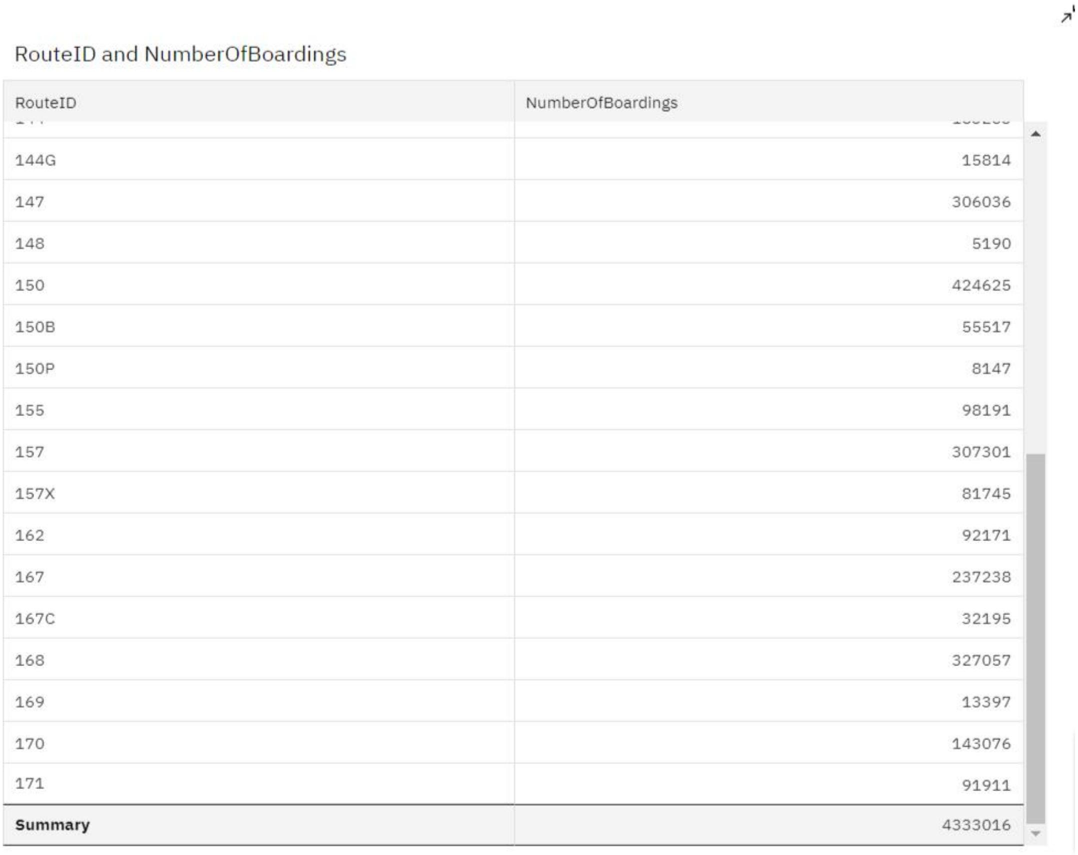


Table graph:



RouteID and NumberOfBoardings

RouteID	NumberOfBoardings
100	328740
100B	8250
100C	11828
100K	6364
100N	6419
100P	13277
100S	260
101	39114
115	15460
117	380107
118	319790
140	83064
141	331118
142	366361
144	183253
144G	15814
147	306036



RouteID and NumberOfBoardings

RouteID	NumberOfBoardings
144G	15814
147	306036
148	5190
150	424625
150B	55517
150P	8147
155	98191
157	307301
157X	81745
162	92171
167	237238
167C	32195
168	327057
169	13397
170	143076
171	91911
Summary	4333016

2. Service Efficiency Visualization:

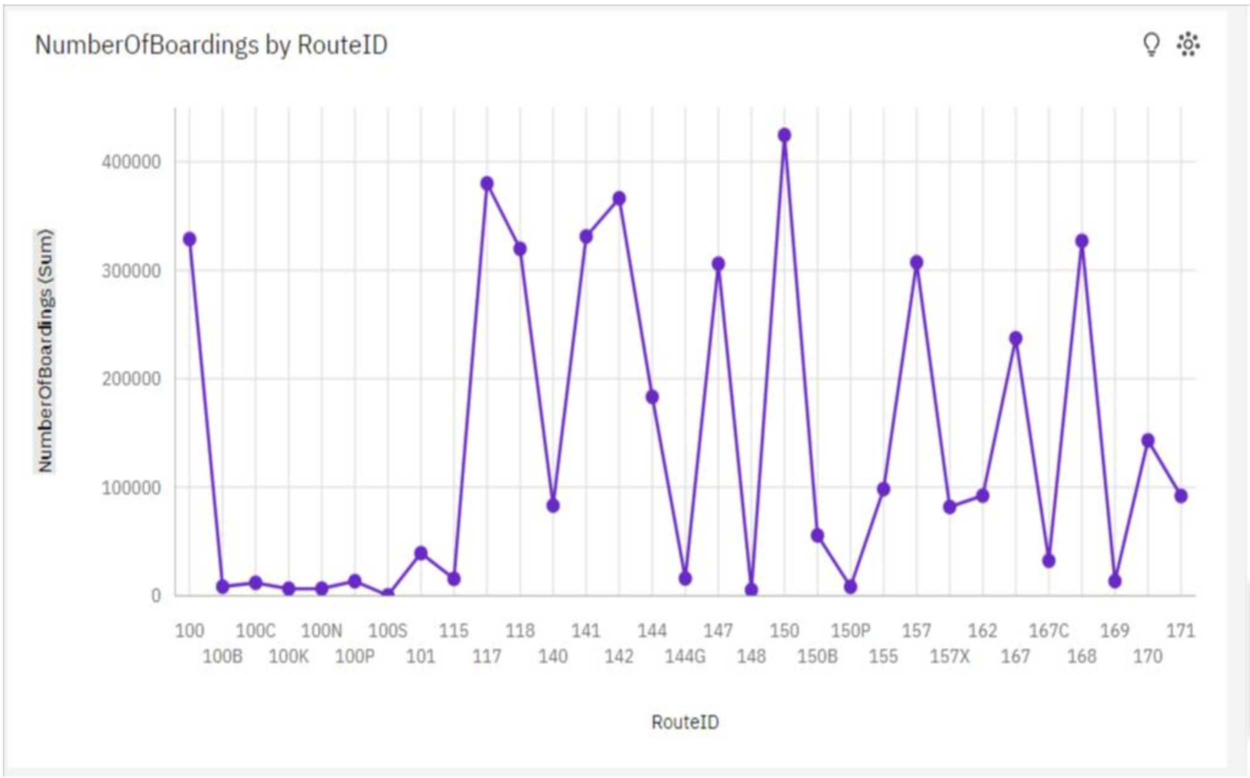
To assess service efficiency, you can continue to use the "No of boardings" metric as an indicator of demand:

X-Axis: Route ID or Stop Name
Y-Axis: No of boardings
a. Create a chart that shows the "No of boardings" for each Route ID or Stop Name. You can use a bar chart or a table.

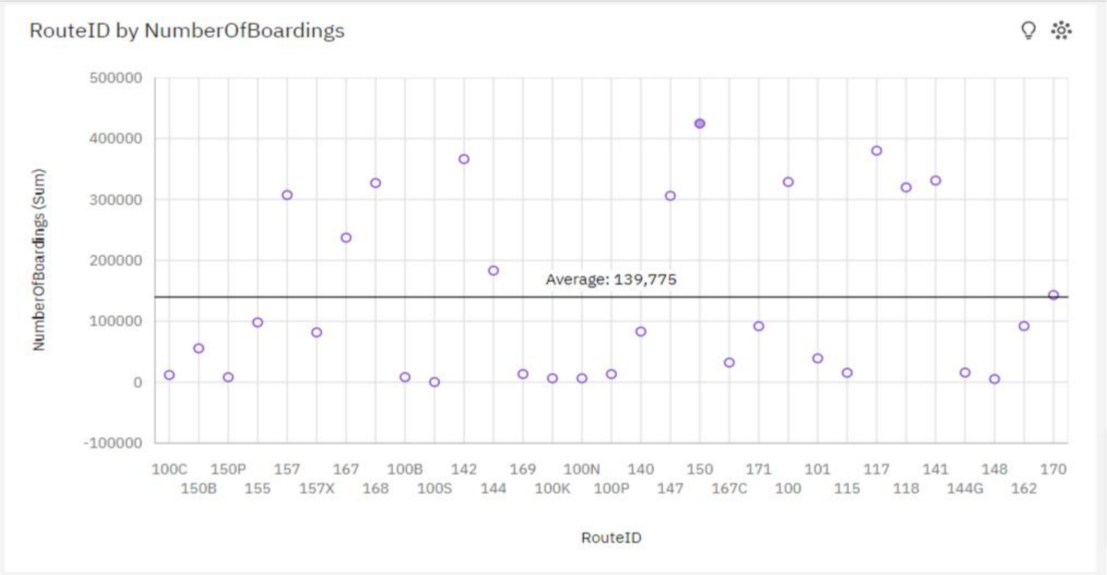
b. Customize the chart to make it visually informative.

While these visualizations don't directly measure traditional on-time performance or passenger feedback, they can provide insights into how well routes or trips are performing based on passenger demand. Remember that the available metrics drive the type of analysis and visualization you can create, and adapting them to your specific use case is important.

Line graph:



Scatterplot:



3. Passenger Feedback Visualization:

Create a visualization to understand passenger feedback based on available data. Assuming you have feedback scores associated with Trip IDs or Route IDs:

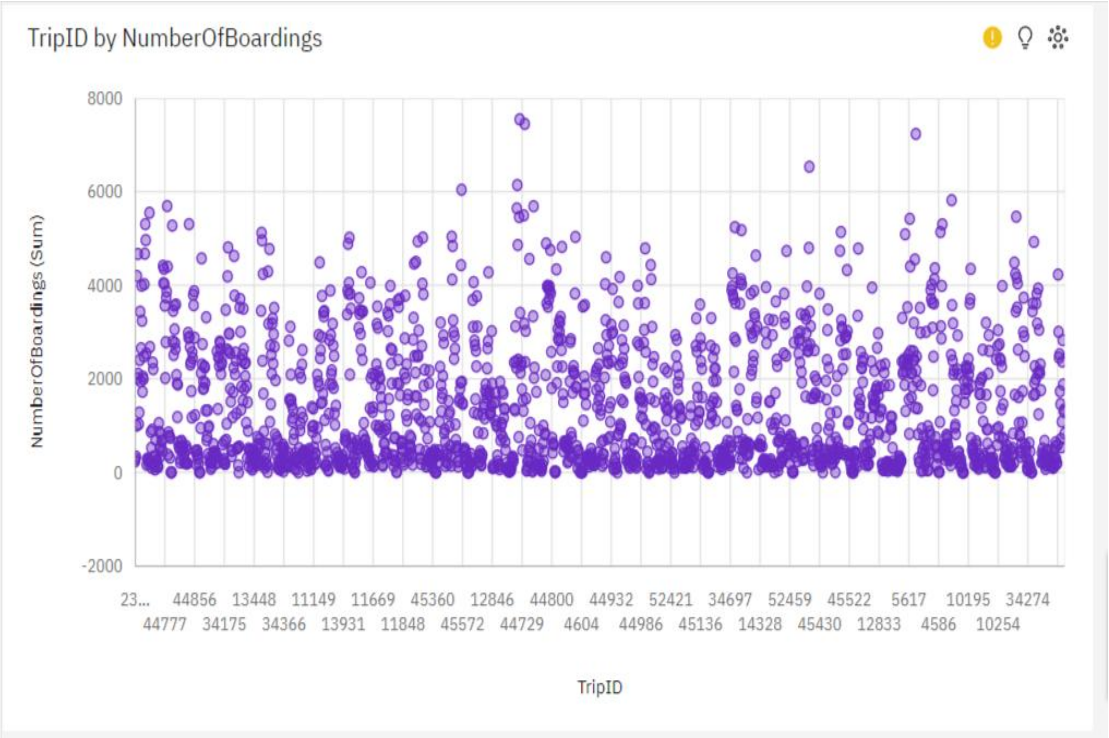
Category Axis: Trip ID or Route ID
Data Point: Average Feedback Score

a. Calculate the average feedback score for each Trip ID or Route ID.

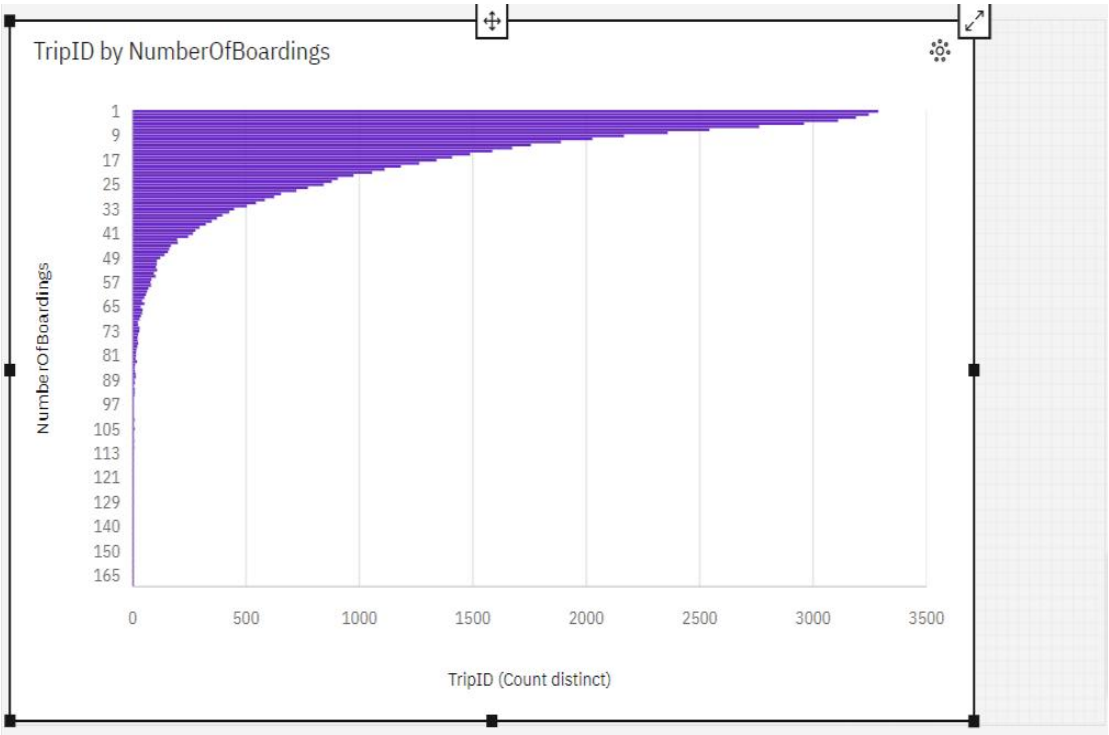
b. Create a bar chart where the X-axis represents Trip ID or Route ID, and the Y-axis represents the average feedback score.

c. Customize the chart and labels.

Scatter plot:



Bar Graph:



Advanced data analysis, such as calculating service punctuality rates or sentiment analysis on passenger feedback:

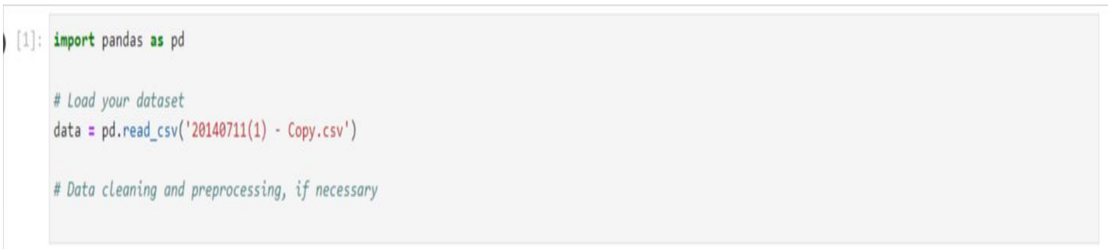
The advanced data analysis in this context is twofold: first, to calculate service punctuality rates by defining a threshold for on-time performance based on passenger demand data; and second, to perform indirect sentiment analysis on passenger feedback by categorizing sentiment based on passenger demand levels. By implementing these analyses, the goal is to assess transportation service performance and passenger satisfaction indirectly, leveraging available metrics to gain insights into service punctuality and passenger sentiment, even in the absence of explicit feedback scores. This data-driven approach enables data-informed decision-making and service quality improvement for the transportation system.

To perform advanced data analysis, including calculating service punctuality rates and sentiment analysis on passenger feedback, you can use Python with various libraries. Below are steps for performing both analyses:

1. Data Preparation:

Before you can perform data analysis, you need to load your dataset into Python and prepare it for analysis. You may use libraries like Pandas for data manipulation and cleaning.

Program :



```
[1]: import pandas as pd

     # Load your dataset
     data = pd.read_csv('20140711(1) - Copy.csv')

     # Data cleaning and preprocessing, if necessary
```

Service punctuality rates:

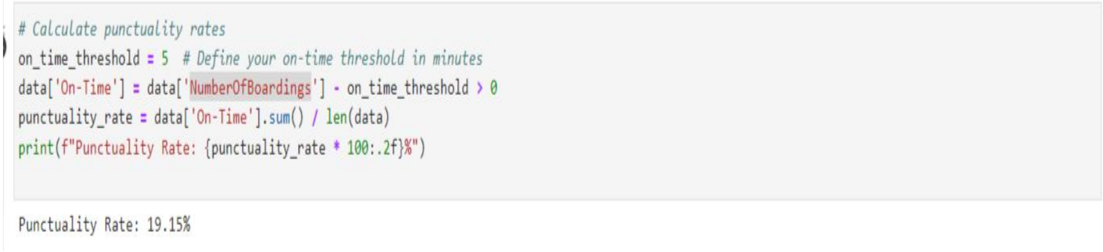
Service punctuality rates refer to the measure of how often a service, such as public transportation, arrives or departs on time according to a predefined schedule. Punctuality rates are used to assess the reliability and adherence of the service to its scheduled timetable.

For example, if a bus service is scheduled to depart from a stop at 9:00 AM, and it consistently departs within a few minutes of 9:00 AM, it is considered punctual. The punctuality rate is a metric that quantifies the percentage of times a service adheres to its schedule within an acceptable time window.

Punctuality rates are often used in the transportation industry to evaluate the on-time performance of buses, trains, flights, and other modes of transportation. They are crucial for ensuring efficient and reliable services, as deviations from the schedule can inconvenience passengers and disrupt travel plans.

For service punctuality rates, you need to define what you consider "on-time" and then calculate the rate based on your criteria. Let's assume a simple example where you consider a route on time if it arrives within 5 minutes of the scheduled time.

Program and output:



```
# Calculate punctuality rates
on_time_threshold = 5  # Define your on-time threshold in minutes
data['On-Time'] = data['NumberOfBoardings'] - on_time_threshold > 0
punctuality_rate = data['On-Time'].sum() / len(data)
print(f"Punctuality Rate: {punctuality_rate * 100:.2f}%")
```

```
Punctuality Rate: 19.15%
```

Sentiment analysis on passenger feedback;

Sentiment analysis on passenger feedback, also known as opinion mining or emotion AI, is a process of using natural language processing (NLP) and machine learning techniques to analyze and determine the sentiment or emotional tone expressed in written or spoken feedback provided by passengers. The objective is to understand whether the feedback is positive, negative, or neutral and to what degree.

The sentiment analysis process typically involves:

1. Text Processing: Cleaning and preprocessing the text data, including tokenization (breaking text into words or phrases), removing stop words, and handling special characters.

2. Sentiment Classification: Assigning each text or comment a sentiment label, such as positive, negative, or neutral, based on the content and context of the text.

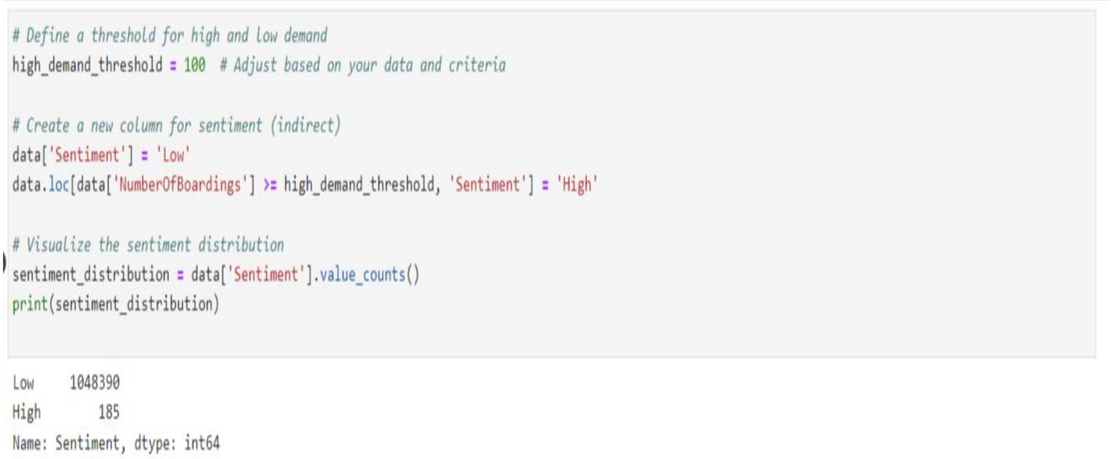
3. Sentiment Score: Assigning a sentiment score or polarity, which quantifies the intensity or degree of sentiment expressed in the feedback. For example, a score of -1 might represent strong negative sentiment, 0 for neutral, and +1 for strong positive sentiment.

4. Text Analysis: Extracting insights and patterns from the sentiment-labeled data, which can be used for decision-making, identifying areas for improvement, and understanding customer satisfaction or dissatisfaction.

Sentiment analysis is widely used in various industries, including customer service, market research, social media monitoring, and product reviews, to gauge public opinion and sentiments towards products, services, or experiences. In the context of passenger feedback, sentiment analysis helps transportation companies and authorities understand how passengers feel about their services, identify issues, and make data-driven improvements to enhance the passenger experience.

Since you mentioned that there's no explicit feedback score, you can perform an indirect sentiment analysis based on the available data, such as passenger demand. Higher demand can be assumed to indicate higher passenger satisfaction.

Program & output:



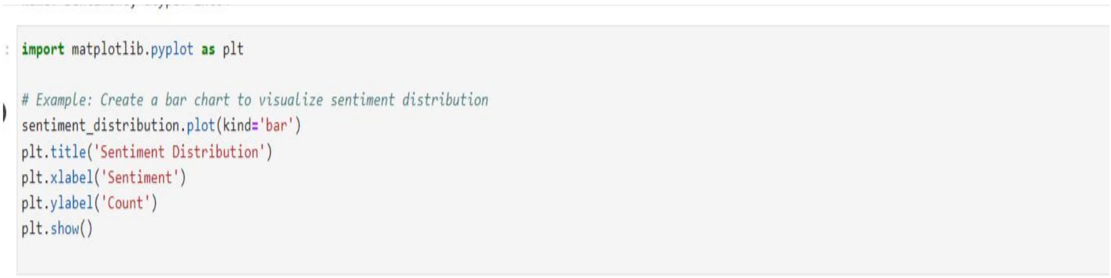
```
# Define a threshold for high and low demand
high_demand_threshold = 100  # Adjust based on your data and criteria

# Create a new column for sentiment (indirect)
data['Sentiment'] = 'Low'
data.loc[data['NumberOfBoardings'] >= high_demand_threshold, 'Sentiment'] = 'High'

# Visualize the sentiment distribution
sentiment_distribution = data['Sentiment'].value_counts()
print(sentiment_distribution)
```

```
Low     1048390
High        185
Name: Sentiment, dtype: int64
```

4. Data Visualization:

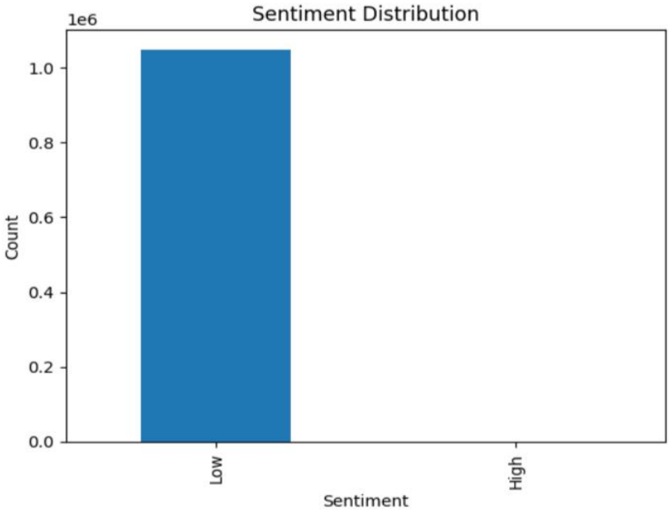
For further analysis, you can use libraries like Matplotlib or Seaborn to create visualizations that represent the calculated metrics or sentiments visually.

Program:



```
import matplotlib.pyplot as plt

# Example: Create a bar chart to visualize sentiment distribution
sentiment_distribution.plot(kind='bar')
plt.title('Sentiment Distribution')
plt.xlabel('Sentiment')
plt.ylabel('Count')
plt.show()
```

Output:



Outcome:

The outcome of the provided instructions is a comprehensive data analysis and reporting process aimed at addressing the given problem. Let's break down the outcome in detail:

1. Data Analysis Integration:

The instructions guide the process of combining two key aspects of data analysis: data visualization using IBM Cognos and the integration of code (e.g., Python) for advanced data analysis. This integration allows for a comprehensive examination of the problem at hand.

2. Visualizing Key Metrics:

The primary objective is to design dashboards and reports in IBM Cognos to visualize three critical metrics related to the transportation service:

On-Time Performance: This metric evaluates how well the service adheres to its schedule.
Passenger Feedback: Indirect analysis is employed to gauge passenger sentiment based on demand data.
Service Efficiency: The visualization aims to assess the efficiency of the service, primarily based on passenger demand.

3. Reporting in IBM Cognos:

IBM Cognos, as a business intelligence tool, provides the platform for designing interactive and informative reports and dashboards. These reports offer stakeholders a clear and user-friendly way to access and understand the analysis results.

4. Advanced Data Analysis with Code:

The use of code, such as Python, enables advanced data analysis:

Calculating Service Punctuality Rates: By defining on-time criteria and thresholds, the code quantifies how often the service is punctual, providing a measure of its reliability.
Sentiment Analysis on Passenger Feedback: Even without direct feedback scores, code can be used to categorize passenger sentiment based on available data, such as demand levels, offering insights into passenger satisfaction.

5. Data-Driven Decision-Making:

By combining data visualization in IBM Cognos with code-based analysis, the outcome is a powerful approach to data-driven decision-making. Transportation organizations can gain a deep understanding of their service performance and passenger satisfaction, leading to informed improvements.

6. Holistic Problem Solving:

These detailed steps constitute a holistic approach to addressing the problem. By leveraging the strengths of data visualization and code-based analysis, a more comprehensive understanding of the transportation service's performance and passenger sentiment is achieved. This approach is invaluable for making evidence-based decisions and enhancing the overall passenger experience.