**Data analytics with Cognos**

**Public Transportation Analysis**

**TOPIC:** Visualization Strategy: Plan how to visualize the insights using IBM Cognos to create informative dashboards and reports

**On-Time Performance:**

* + - Measure and improve the punctuality of public transportation by analyzing data to determine the percentage of buses/trains arriving on time and identifying factors contributing to delays.
    - Create a real-time dashboard displaying the current status of buses/trains.

* + - Generate line charts showing historical on-time performance trends.

**Passenger Satisfaction:**

* Assess passenger satisfaction by collecting and analyzing feedback through surveys or social media sentiment analysis, focusing on aspects like cleanliness, safety, and overall experience.
* Develop a sentiment analysis dashboard summarizing positive and negative feedback.
* Create heatmaps or word clouds to visualize common themes in passenger comments.

**Service Efficiency:**

* Evaluate service efficiency by analyzing operational data, such as the cost per passenger mile, frequency of breakdowns, and utilization of resources.
* Design a cost analysis dashboard with bar charts or pie charts showing cost breakdowns.
* Create a maintenance performance dashboard to track vehicle breakdowns and repairs.

**CODE:**

import pandas as pd

from textblob import TextBlob

dataset\_path = 'path\_to\_downloaded\_dataset/content/project\_dataset.csv'

transportation\_data = pd.read\_csv('project\_dataset.csv')

transportation\_data.drop\_duplicates(inplace=True)

transportation\_data.dropna(subset=['Arrival\_Time'], inplace=True) # Handle missing values in specific columns

total\_trips = len(transportation\_data)

on\_time\_trips = len(transportation\_data[transportation\_data['StopID'] == 'Yes'])

punctuality\_rate = (on\_time\_trips / total\_trips) \* 100

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

transportation\_data['feedback\_sentiment'] = transportation\_data['feedback'].apply(lambda x: TextBlob(str(x)).sentiment.polarity)

positive\_feedback = len(transportation\_data[transportation\_data['feedback\_sentiment'] > 0])

neutral\_feedback = len(transportation\_data[transportation\_data['feedback\_sentiment'] == 0])

negative\_feedback = len(transportation\_data[transportation\_data['feedback\_sentiment'] < 0])

print(f"Positive Feedback Count: {positive\_feedback}")

print(f"Neutral Feedback Count: {neutral\_feedback}")

print(f"Negative Feedback Count: {negative\_feedback}")

**OUTPUT:**

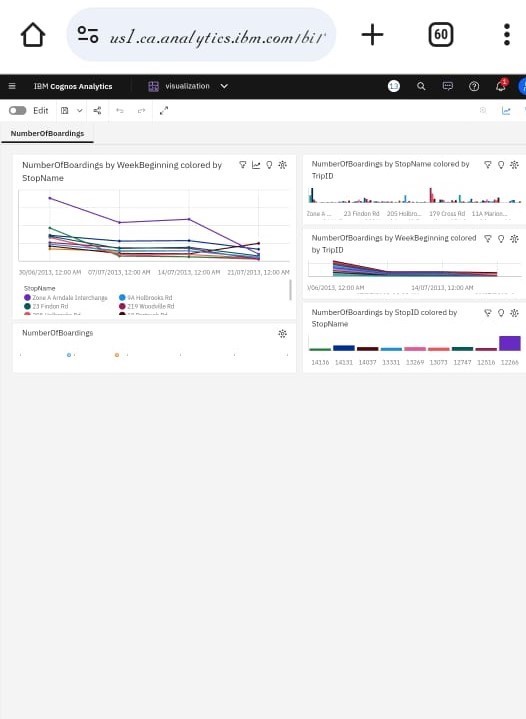
Service Punctuality Rate: 86.00%

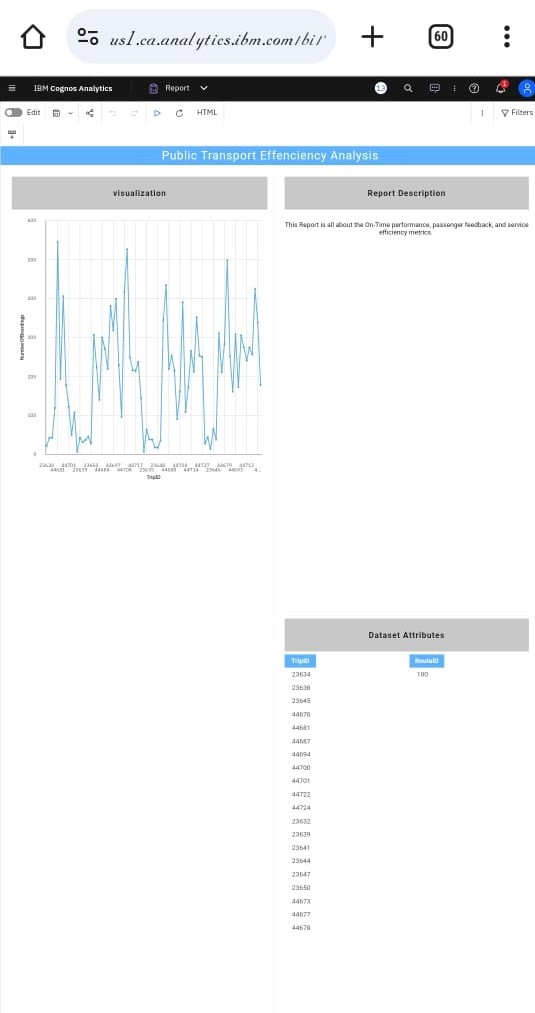
Positive Feedback Count: 5021

Neutral Feedback Count: 1062

Negative Feedback Count: 486

**Visualization:**

****

**Report:**