**PHASE:4**

**Public Transportation Efficiency Analysis**

**TEAM MEMBERS:**

NAME: Murali V

REG NO:721221104040

NAME: Ragul M

REG NO:721221104044

NAME: Aswin S

REG NO:721221104009

NAME: Arun S

REG NO:72122110400

NAME: Abishek S

REG NO:721221104004

**Table of Contents:**

1. Introduction

2. IBM Cognos Visualizations

2.1. On-Time Performance Dashboard

2.2. Passenger Feedback Dashboard

2.3. Service Efficiency Metrics Dashboard

3. Data Analysis with Code

3.1. Advanced Data Analysis with Python

3.2. Calculating Service Punctuality Rates

3.3. Sentiment Analysis on Passenger Feedback

4. Conclusion

**Dataset Link:** [**https://drive.google.com/drive/folders/1aM2ROjLiMaz6aXkKMvARcEmaeV6fYow?usp=drive\_link**](https://drive.google.com/drive/folders/1aM2ROjLiMaz6aXkKMvARcEmaeV6fYow?usp=drive_link)

**Source code link:**

[**https://drive.google.com/drive/folders/1lrQhLK6w6EaWQjJzIIwRUwSh0nwnbzf?usp=drive\_**](https://drive.google.com/drive/folders/1lrQhLK6w6EaWQjJzIIwRUwSh0nwnbzf?usp=drive_)

**link**

**1. Introduction**

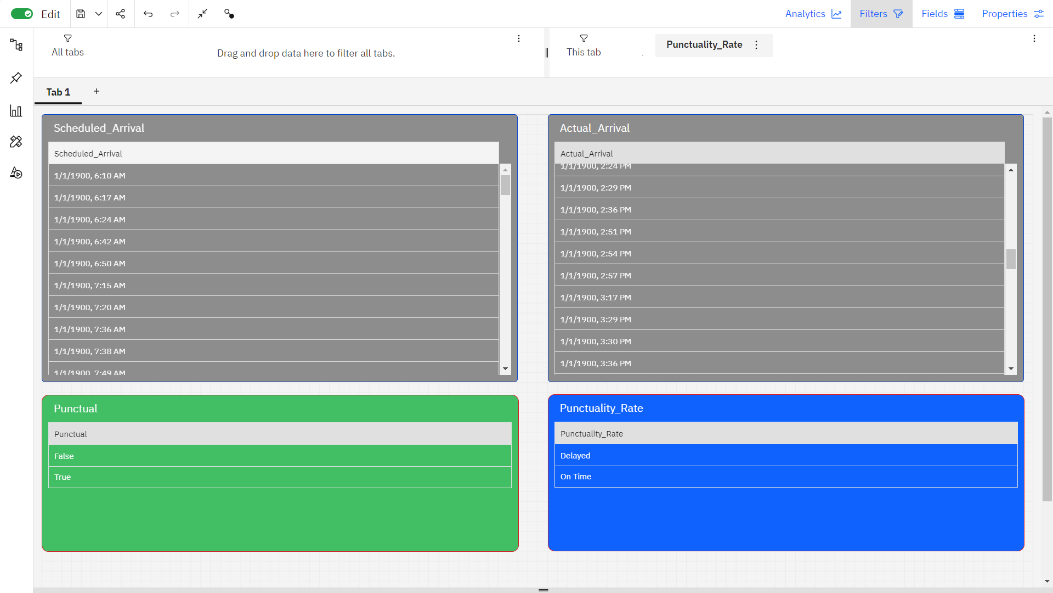
Briefly introduce the project, its purpose, and the role of IBM Cognos and code-based data analysis in achieving the project's objectives.

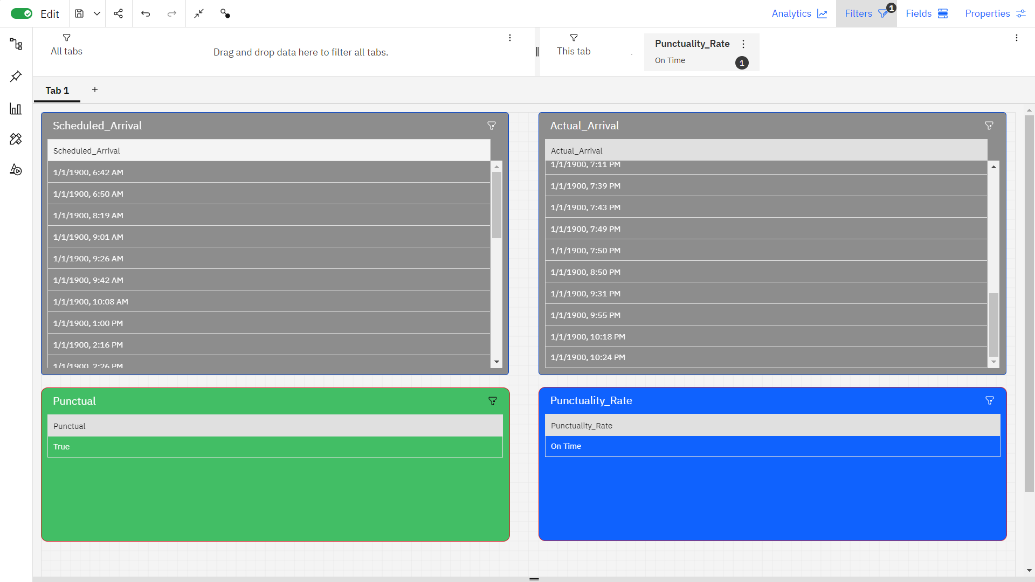
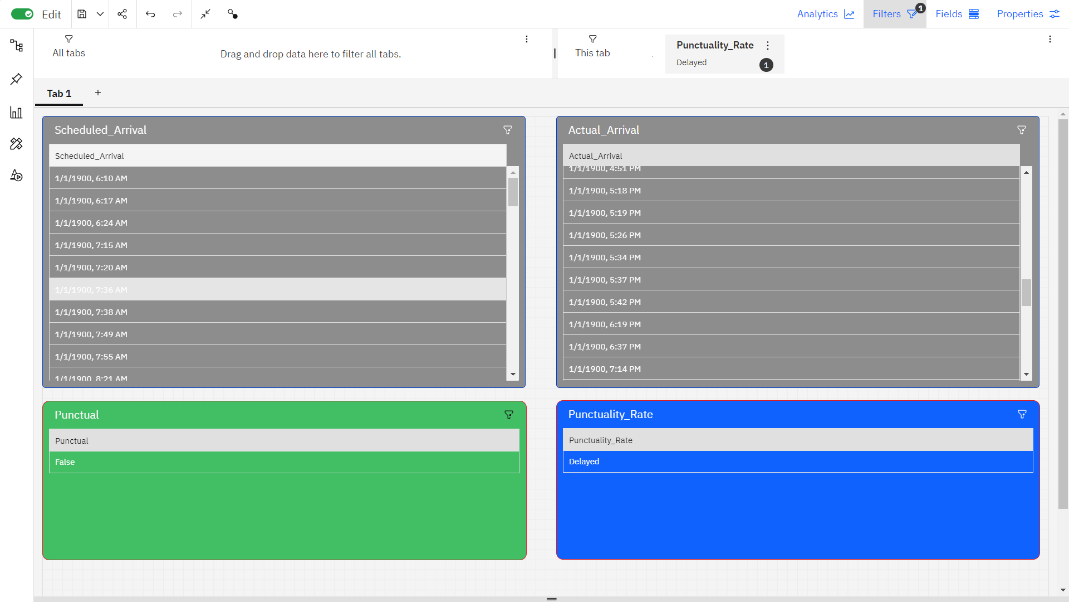
**2. IBM Cognos Visualizations**

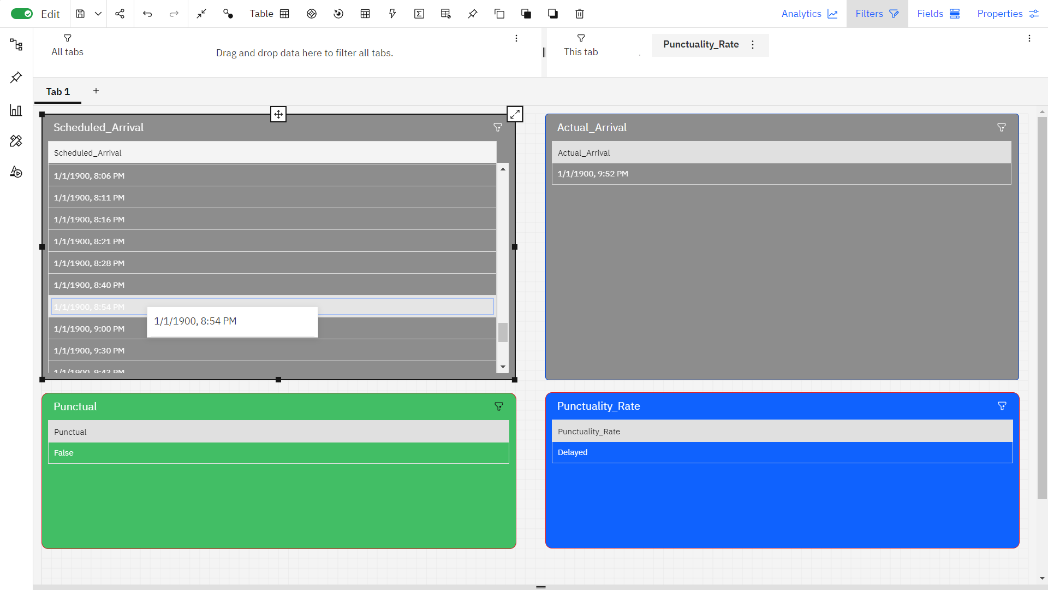
**2.1. On-Time Performance Dashboard**

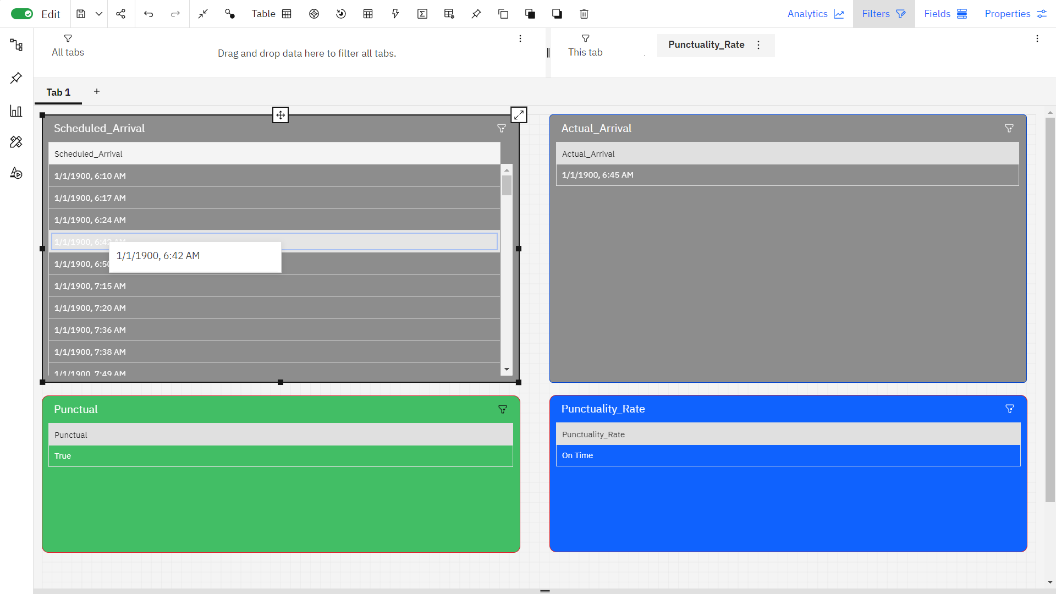
The On-Time Performance Dashboard is designed to provide a visual representation of the punctuality and reliability of a service.

**Screenshots:**

****

****

****

****

In the On-Time Performance Analysis Dashboard, the following key information is presented:

**1. Actual Arrival Time:** This is the time at which a service, such as a bus, train, or flight, actually arrives at its destination. It is recorded in real-time and provides an accurate indication of when the service reaches its final stop.

**2. Scheduled Arrival Time:** The scheduled arrival time represents the expected or planned arrival time for the service. It's the time passengers were informed to expect the service to arrive.

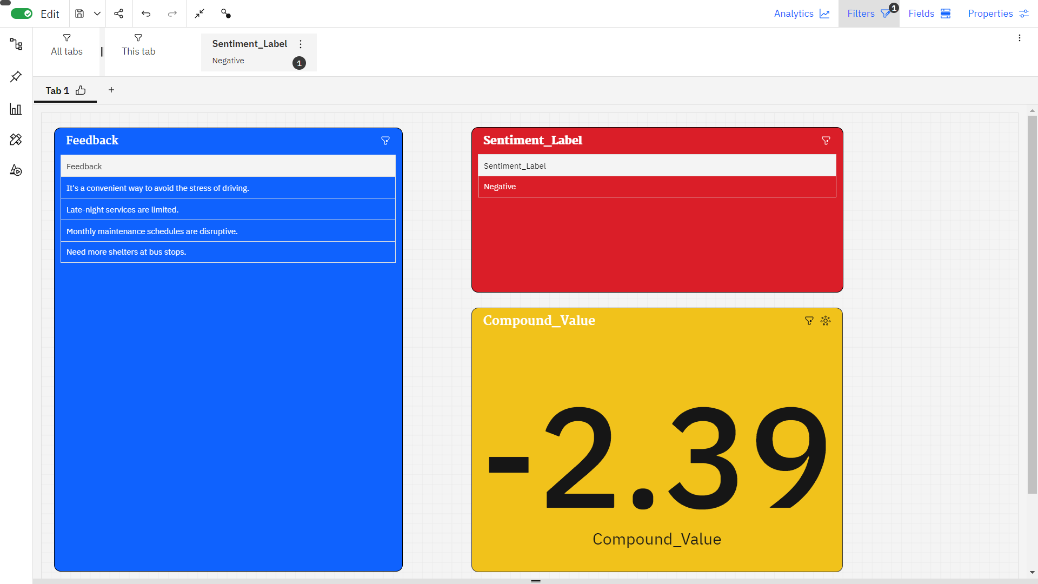
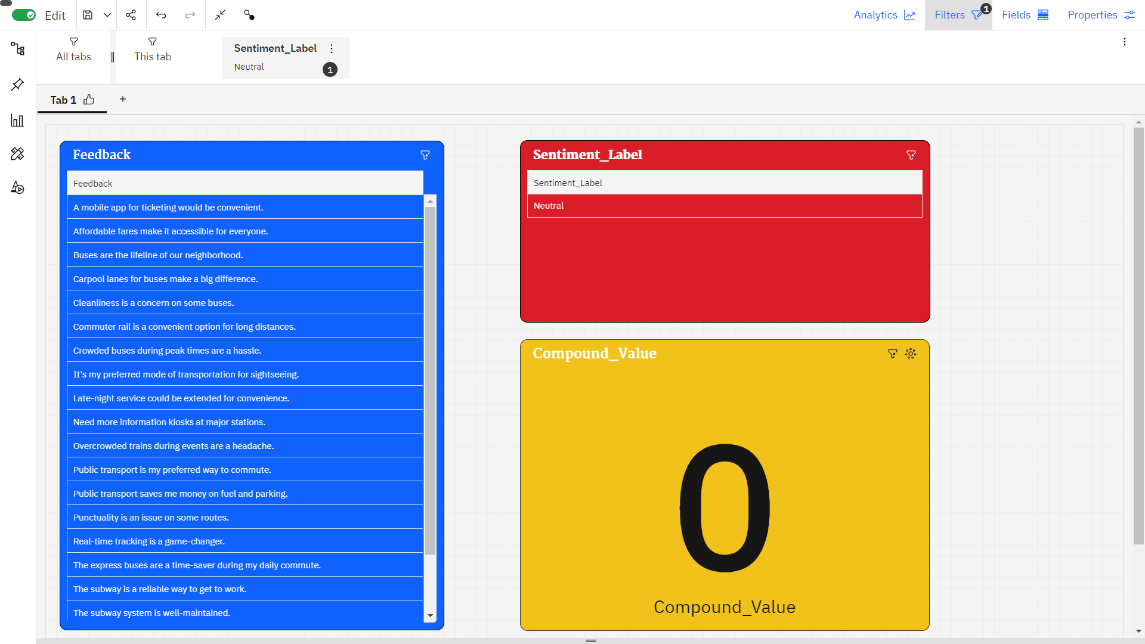
**3. Punctuality Rate:** The punctuality rate is a percentage that indicates how often the service arrives on time, i.e., within the acceptable time window defined for punctuality. It's calculated by comparing the actual arrival time to the scheduled arrival time.

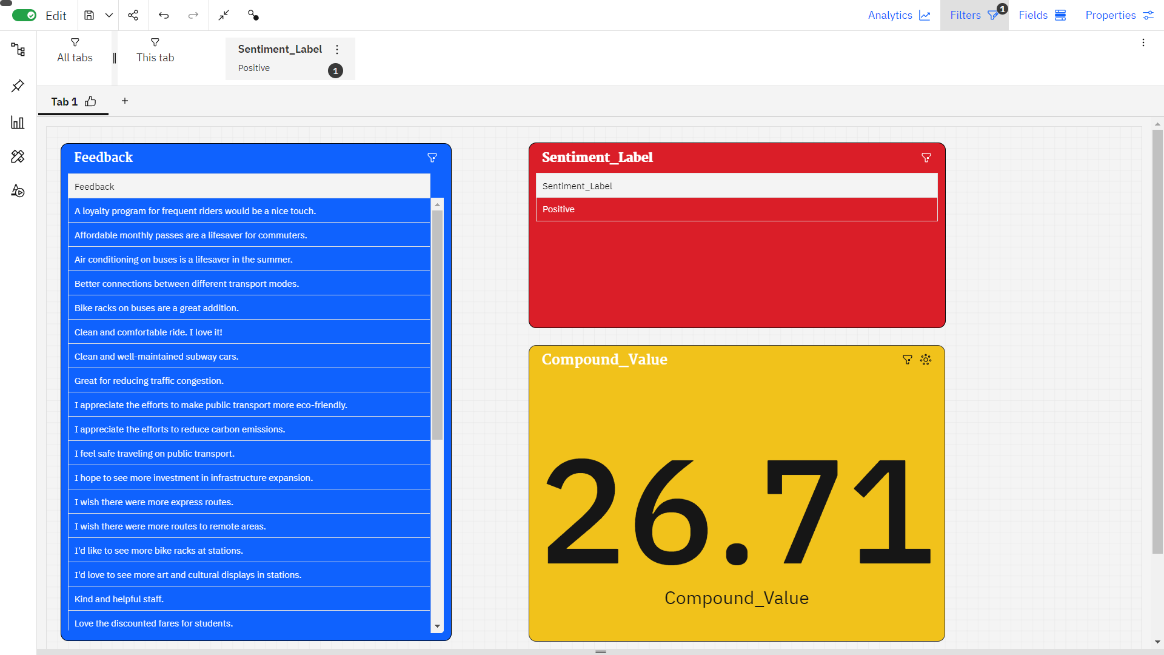
**4. Punctual or Delayed:** This is a categorical indicator that classifies whether the service's arrival was punctual or delayed. If the actual arrival time falls within the defined punctuality time window, it's labeled as "Punctual." If it's outside that time window, it's labeled as "Delayed."

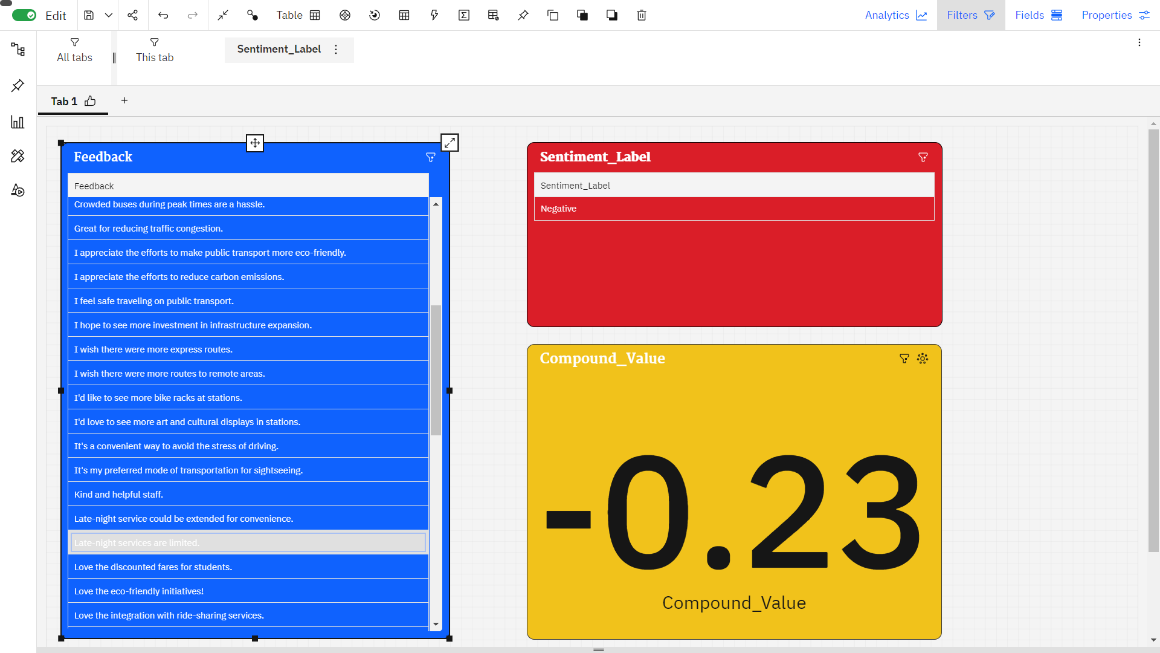
**2.2. Passenger Feedback Dashboard**

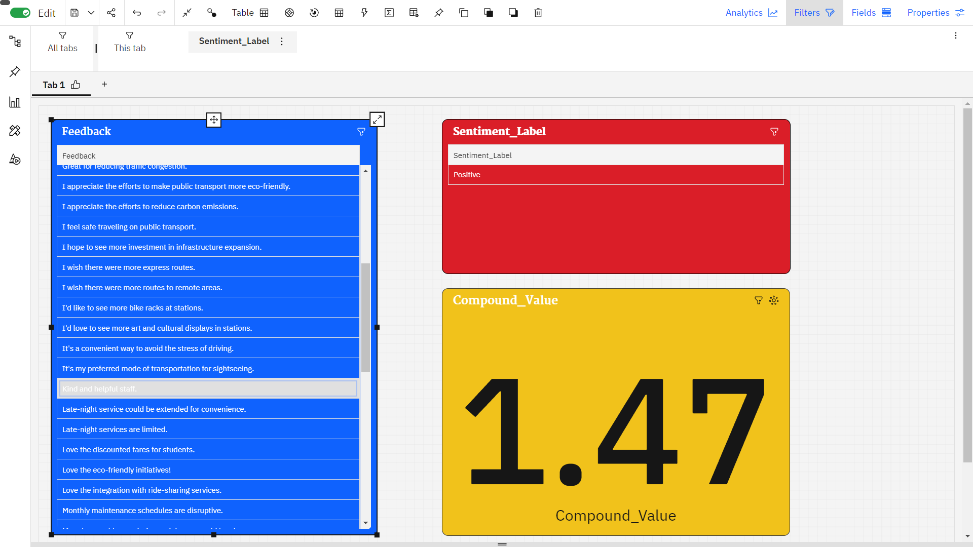
The Passenger Feedback Dashboard is designed to provide a comprehensive overview of customer sentiment and feedback for a service.

**Screenshots:**









In the Passenger Feedback Dashboard, feedback data collected from various sources is processed and analysed to provide valuable insights into passenger sentiment and satisfaction. Here's a brief explanation of key components:

**1. Feedback Data:** This is the raw data collected from passengers through surveys, social media, customer support interactions, app/website feedback, comment cards, online forums, and other channels. It includes comments, ratings, and other textual or numerical feedback provided by passengers.

**2. Sentiment Analysis**: Sentiment analysis is a natural language processing (NLP) technique used to classify the sentiment or emotional tone of the feedback data. It categorizes the feedback into three main sentiment categories:

**Positive:** Feedback with a positive tone, indicating satisfaction or happiness with the service.

**Negative:** Feedback with a negative tone, highlighting dissatisfaction, concerns, or issues.

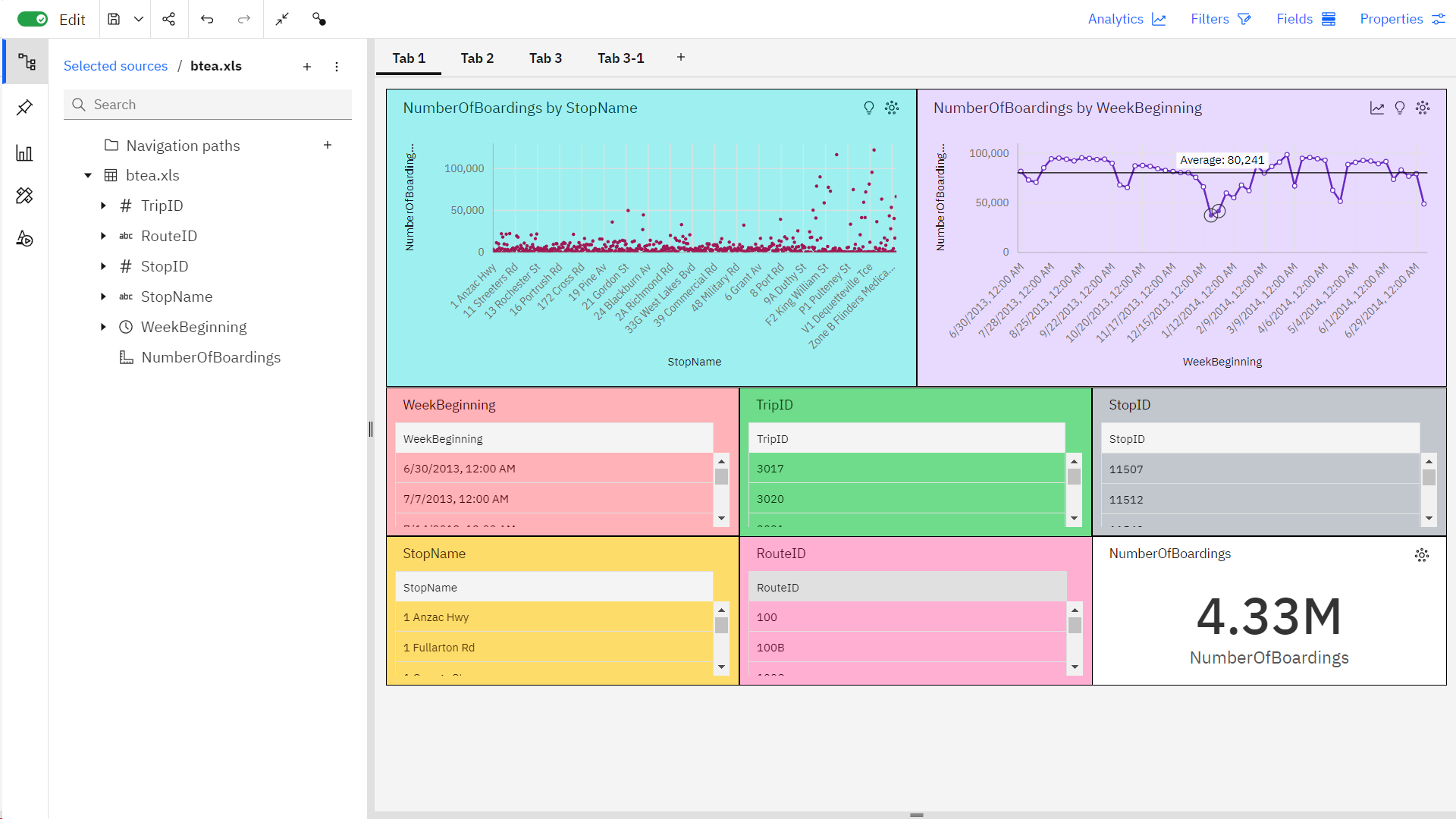
**Neutral:** Feedback that doesn't express strong positive or negative sentiment, often considered as factual or objective statements.

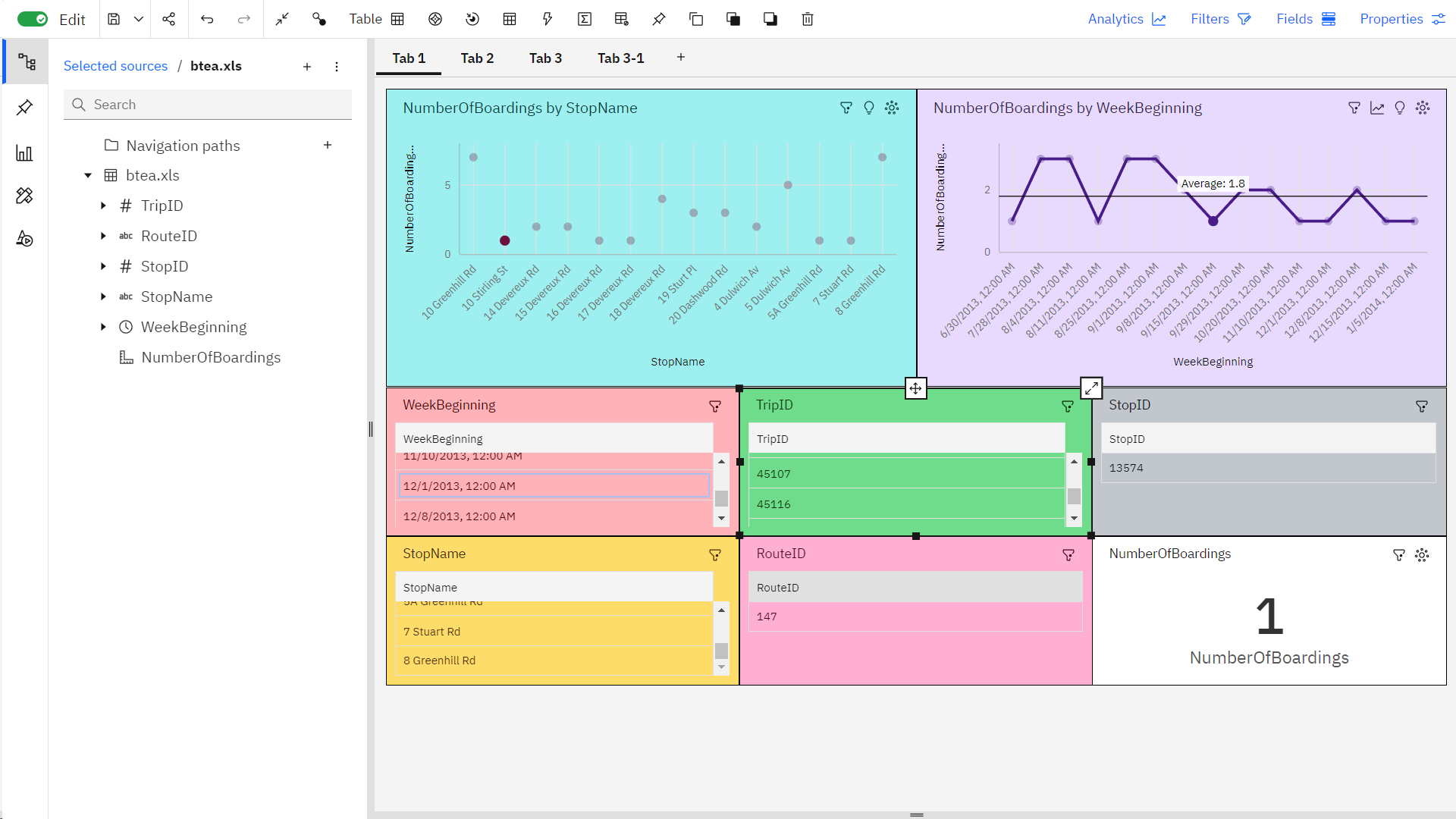
**3. Compound Value** :The compound value is a single numerical score that represents the overall sentiment or polarity of a piece of feedback. It's often calculated using sentiment analysis libraries and algorithms. The compound value takes into account the entire context of the text and assigns a score that ranges from -1 (most negative) to 1 (most positive). A score around 0 suggests a neutral sentiment.

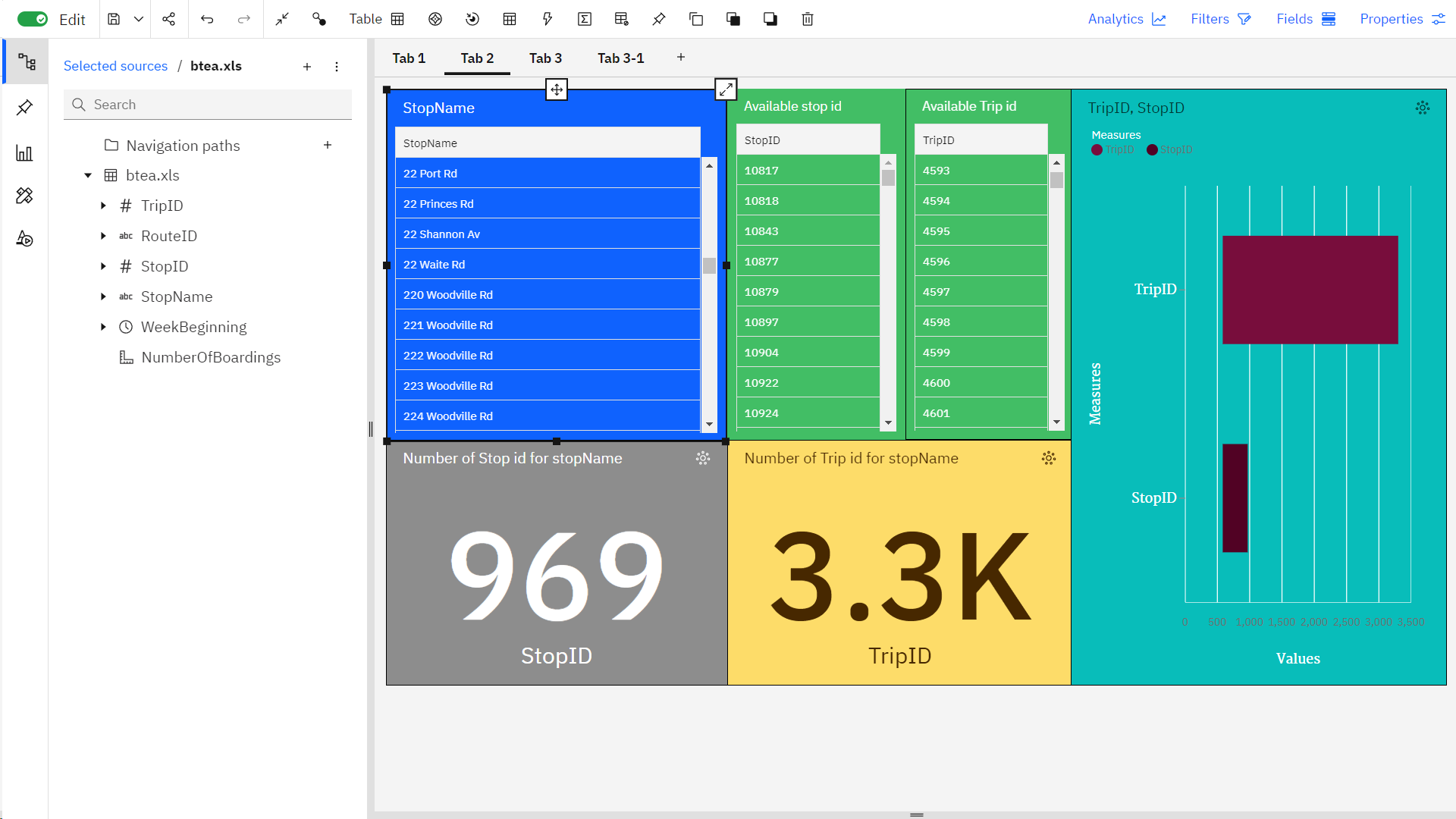
For instance, if a passenger provides feedback like, "The service was excellent, but the wait times were frustrating," sentiment analysis would classify "excellent" as positive, "frustrating" as negative, and the overall compound value would reflect the mixed sentiment of the feedback.

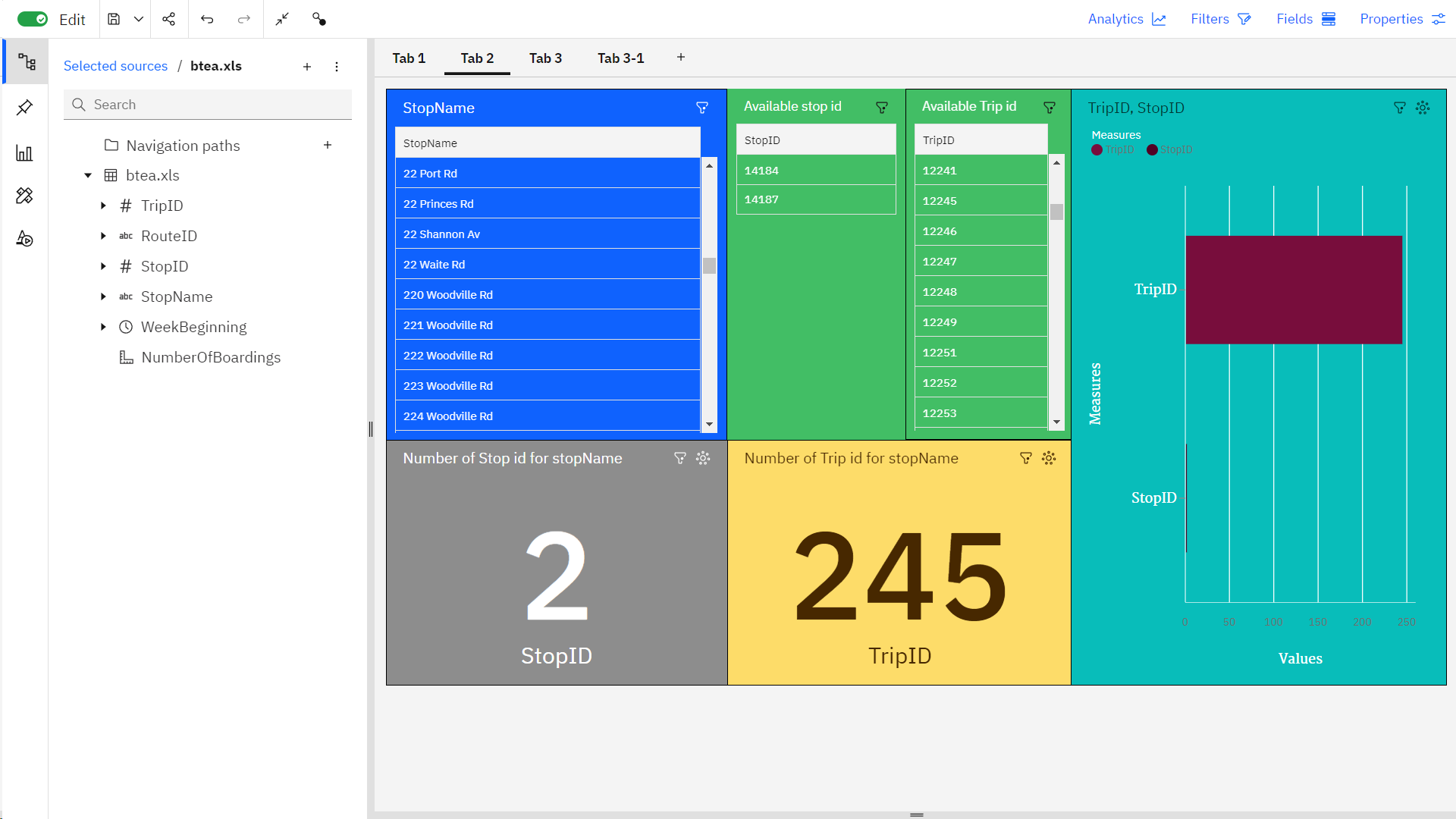
**2.3. Service Efficiency Metrics Dashboard**

The Service Efficiency Metrics Dashboard is designed to provide a comprehensive view of the efficiency and operational performance of a service, such as a transportation system.









3. **Data Analysis with Code**

**3.1. Advanced Data Analysis with Python**

**# Importing data and preprocessing**

import pandas as pd

import numpy as np

import sklearn

from sklearn.preprocessing import StandardScaler

import matplotlib.pyplot as plt # Add this import statement

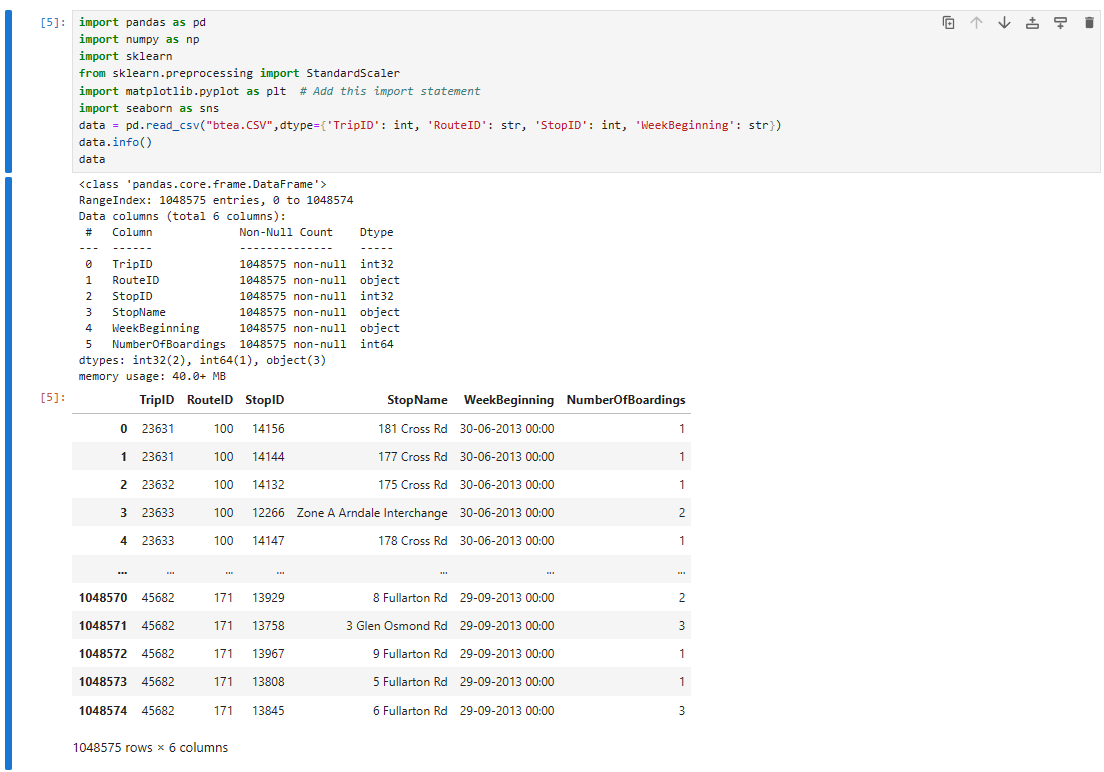
import seaborn as sns

data = pd.read\_csv("btea.CSV",dtype={'TripID': int, 'RouteID': str, 'StopID': int, 'WeekBeginning': str})

data.info()

data

**#output**



**#code**

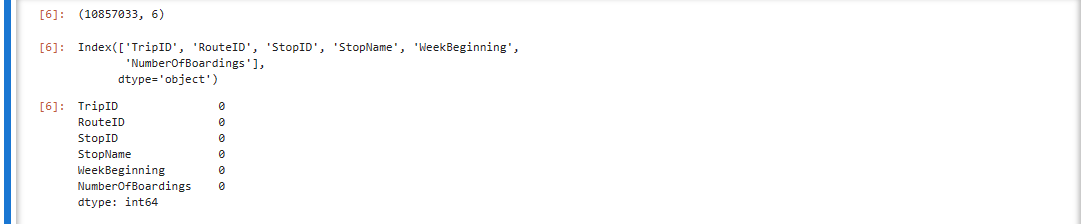
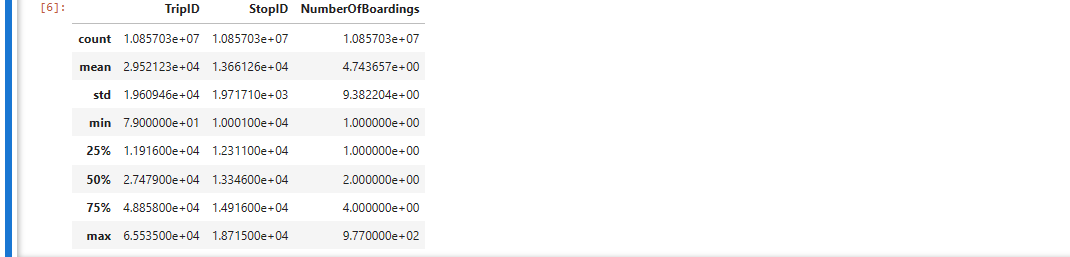
data.shape

data.columns

pd.isnull(data).sum()

data.describe()

**#output**

**#Data visualization**

plt.figure(figsize=(8, 6))

sns.histplot(data['TripID'], bins=20, kde=True, color='skyblue', label='TripID')

sns.histplot(data['StopID'], bins=20, kde=True, color='salmon', label='StopID')

plt.title('Distribution of TripID and StopID')

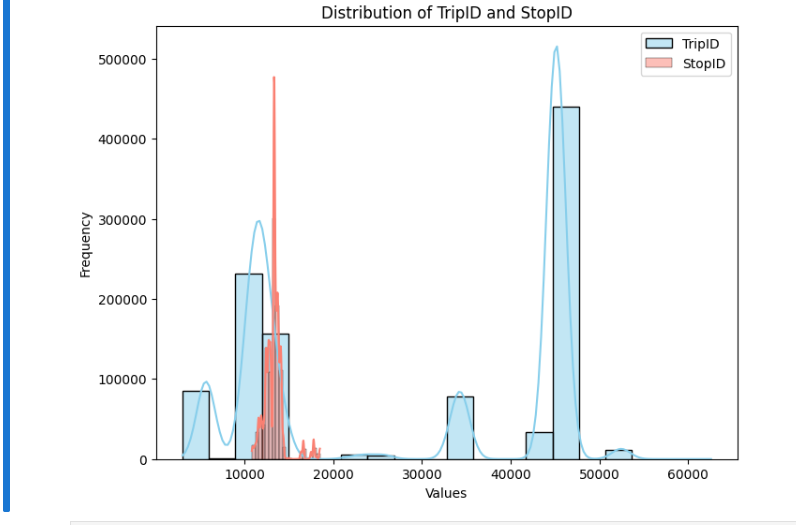
plt.xlabel('Values')

plt.ylabel('Frequency')

plt.legend()

plt.show()

**#output**



**#Correlation matrix**

numerical\_data = data.select\_dtypes(include=[np.number])

corr\_matrix = numerical\_data.corr()

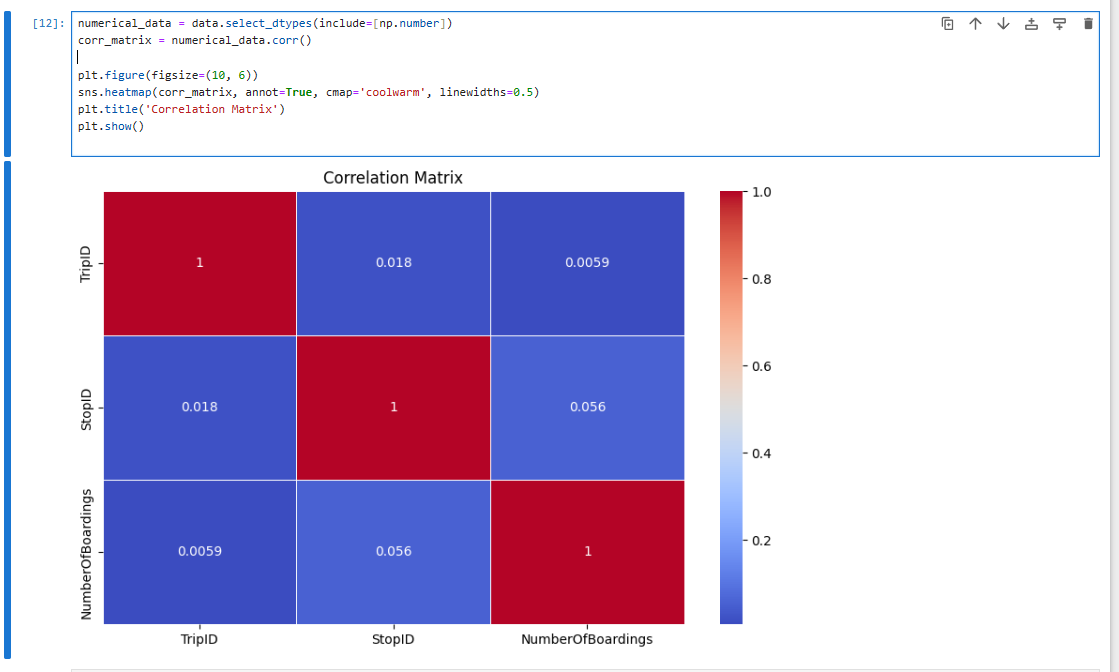
plt.figure(figsize=(10, 6))

sns.heatmap(corr\_matrix, annot=True, cmap='coolwarm', linewidths=0.5)

plt.title('Correlation Matrix')

plt.show()

**#output**



**3.2. Calculating Service Punctuality Rates**

**#Data processing and calculate punctuality Rate**

import pandas as pd

data = pd.read\_csv('arrival\_times.csv')

data['Scheduled\_Arrival'] = pd.to\_datetime(data['Scheduled\_Arrival'], format='%H:%M')

data['Actual\_Arrival'] = pd.to\_datetime(data['Actual\_Arrival'], format='%H:%M')

threshold = pd.Timedelta(minutes=20)

data['Punctual'] = (data['Actual\_Arrival'] - data['Scheduled\_Arrival']) <= threshold

data['Punctuality\_Rate'] = data['Punctual'].apply(lambda x: 'On Time' if x else 'Delayed')

data.to\_csv('punctuality.csv', index=False)

**#Calculating punctuality rate**

on\_time\_rate = (data['Punctual'].sum() / len(data)) \* 100

print(f'On-Time Performance Rate: {on\_time\_rate:.2f}%')

**#output**

****

**#Data Visualization**

import pandas as pd

import matplotlib.pyplot as plt

data = pd.read\_csv('punctuality.csv')

punctuality\_counts = data['Punctuality\_Rate'].value\_counts()

plt.figure(figsize=(2, 6))

plt.bar(punctuality\_counts.index, punctuality\_counts.values, color=['red', 'green'])

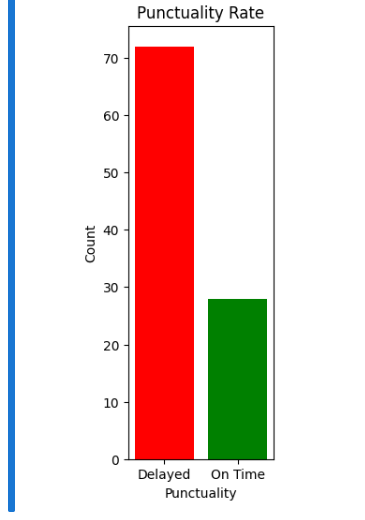
plt.title('Punctuality Rate')

plt.xlabel('Punctuality')

plt.ylabel('Count')

plt.show()

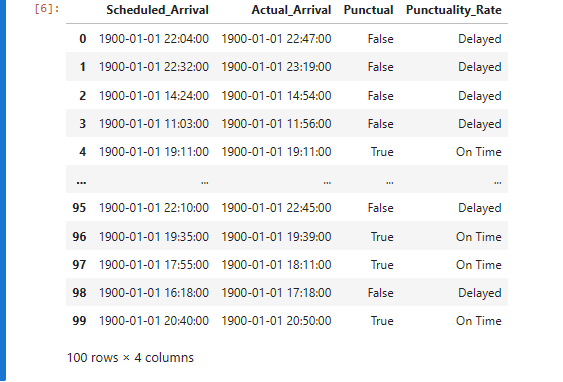
**#output**

****

data = pd.read\_csv('punctuality.csv')

data

**#output**

****

**3.3. Sentiment Analysis on Passenger Feedback**

import pandas as pd

from nltk.sentiment.vader import SentimentIntensityAnalyzer

data = pd.read\_csv('feedback.csv')

sid = SentimentIntensityAnalyzer()

sentiments = []

for comment in data['Feedback']:

sentiment = sid.polarity\_scores(comment)

sentiments.append(sentiment)

sentiment\_labels = []

compound\_values = [] # To store the compound values

for sentiment in sentiments:

compound = sentiment['compound']

compound\_values.append(compound) # Append the compound value

if compound >= 0.05:

sentiment\_labels.append('Positive')

elif compound <= -0.05:

sentiment\_labels.append('Negative')

else:

sentiment\_labels.append('Neutral')

# Print results

for i in range(len(data['Feedback'])):

print(f"Feedback: {data['Feedback'][i]}")

print(f"Sentiment: {sentiment\_labels[i]}")

print(f"Compound Value: {compound\_values[i]}\n")

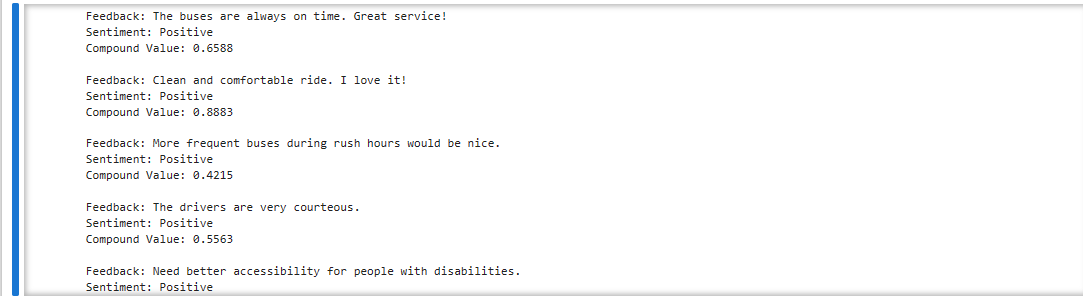
data['Sentiment\_Label'] = sentiment\_labels

data['Compound\_Value'] = compound\_values

# Save the DataFrame to a new CSV file

data.to\_csv('feedback\_with\_sentiment.csv', index=False)

**#output**

****

**4. Conclusion**

In summary, the project successfully leveraged IBM Cognos and advanced data analysis, primarily using Python, to enhance the monitoring and optimization of a service, such as a transportation system, or customer service operation.