

# TRACKS OF THE NEW YORK CITY 2025 SUBWAY



NYU

**Optimizing Urban Transit: A  
Big Data Approach to MTA  
Subway Ridership and  
Service Analysis**

CSGY-6513 Spring 2025 Big Data : Prof. Amit Patel

Team Details: Akshat Mishra (am15111), Nikita  
Gupta (ng3230), Jay Daftari (jd5829)



# AGENDA

- INTRODUCTION & PROBLEM STATEMENT
- DATA SOURCES
- WHY BIG DATA?
- DATA PROCESSING
- PROJECT ARCHITECTURE
- EXPLORING RIDERSHIP
- RIDER SENTIMENT ANALYSIS
- RECOMMENDATIONS FOR IMPROVEMENT
- FEATURES
- XGBOOST RIDERSHIP PREDICTION
- HOW CAN WE IMPROVE THE ANALYSIS FURTHER

# INTRODUCTION AND PROBLEM STATEMENT

The Metropolitan Transportation Authority (MTA) is an integral part of New York City, serving as the backbone of daily commutes for millions of residents and visitors. MTA is essential to the city's economy and is what ties the city together.

As ridership has grown significantly since the pandemic, the demand for reliable, efficient, and accessible service has increased. This ever-growing demand brings its own set of challenges, from managing delays, maintaining service quality, and addressing rider satisfaction.



# DATA SOURCES

**Datasets Used:** (Millions of Rows ~ **35 GB** of size)

- **MTA Customer Feedback Data (2014–2019):** Contains feedback regarding service quality, complaints, and commendations.
- **MTA Daily Ridership Data (Post-2020):** Tracks daily ridership metrics for subways, buses, and other transit systems.
- **MTA Key Performance Indicators (2008–2021):** Evaluates performance metrics such as punctuality and safety.
- **MTA Monthly Ridership Traffic Data:** Aggregated monthly ridership data for trend analysis.
- **MTA Service Alerts Data (Post-2020):**
  - Logs alerts and disruptions in transit services.
- **MTA Subway Customer Journey Metrics:** Provides details on journey times, delays, and passenger metrics.
- **MTA Hourly Ridership Data (Post-July 2020):** Hourly granular ridership data for subway systems.
- **MTA Subway Major Incidents Data:** Logs major incidents affecting service.
- **MTA Subway Origin-Destination Data:** Tracks passenger movement between stations.
- **MTA Subway Stations:** Metadata about station attributes such as location, accessibility, and facilities.
- **MTA Subway Stations and Complexes:** Details about station complexes and associated stations.
- **MTA Subway Turnstile Usage Data:** Data on subway entry and exit counts.
- **MTA Subway and Bus Lost Time Data:** Metrics on lost time due to accidents and other disruptions.

## Sources:

- MTA Open Data Platform
- Official NYC Open Data API



## Some important KeyWords

- **Ridership:** Total number of riders that entered a subway complex via OMNY or MetroCard at the specific hour and for that specific fare type.
- **Date :** The date of travel (MM/DD/YYYY).
- **transit\_mode:** Distinguishes between the subway, Staten Island Railway, and the Roosevelt Island Tram
- **Agency:** This is the abbreviated code for an agency. Example: LIRR=Long Island Rail Road, MNR = Metro-North Railroad, NYCT = New York City Transit
- **station\_complex\_id:** A unique identifier for station complexes
- **station\_complex:** The subway complex where an entry swipe or tap took place. Large subway complexes, such as Times Square and Fulton Center, may contain multiple subway lines.
- **Alert ID:** Unique ID of events that can range from planned maintenance and construction activities to unexpected incidents such as accidents, track maintenance or planned work and more
- **ADA(Americans with Disabilities):** 0 if the station is not ADA-accessible, 1 if the station is fully accessible, 2 if the station is partially accessible.
- **Transfers:** Number of individuals who entered a subway complex via a free bus-to-subway, or free out-of-network transfer. This represents a subset of total ridership, meaning that these transfers are already included in the preceding ridership column. Transfers that take place within a subway complex (e.g., individuals transferring from the 2 to the 4 train within Atlantic Avenue) are not captured here.
- **distance to a central :** distance from central point (e.g., Times Square).

# WHY BIG DATA??

- **Volume** - Handling large datasets from multiple sources (daily, monthly, hourly ridership, service alerts, station data).
- **Variety** - Diverse data types: numerical ridership figures, textual service alerts, geospatial station locations.
- **Velocity** - Real-time data processing for timely insights and alerts.
- **Veracity** - Ensuring data quality and accuracy amidst vast and varied data sources.
- **Value** - Extracting actionable insights to drive strategic decisions.

## Loading:

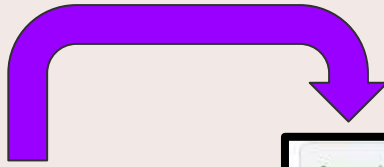
- Used **PySpark** for large dataset handling and sampling.
- Example: Sampled 5% of the "Hourly Ridership Data" for exploratory analysis.

**Cleaning:** Removed duplicates, handled missing values, standardized column names.

## Transformation:

- Created a geosphere for mapping latitudes and longitudes.
- Calculated and added the distance parameter calculating it from the geosphere.
- Derived new features like "Day of Week" and "Hour of Day" for trend analysis.

**Merging:** Integrated multiple datasets on keys such as station\_complex\_id, transit\_times, or agency.



```
import pandas as pd
import random
from pyspark.sql import SparkSession
from pyspark.sql.functions import rand

spark = SparkSession.builder \
    .appName("MTADataSampling") \
    .getOrCreate()

input_file = "/kaggle/input/mta-combined/MTA_Subway_Hourly_Ridership__Beginning_July_2020.csv"

sample_fraction = 0.01

# Read the CSV file using PySpark
try:
    # Load data into Spark DataFrame
    df = spark.read.csv(input_file, header=True, inferSchema=True)

    sampled_df = df.sample(withReplacement=False, fraction=sample_fraction, seed=42)

    sampled_df.coalesce(1).write.csv("sampled_output_temp", header=True, mode='overwrite')

    import shutil
    import glob
    import os

    part_file = glob.glob("sampled_output_temp/part-*.csv")[0]

    shutil.move(part_file, output_file)

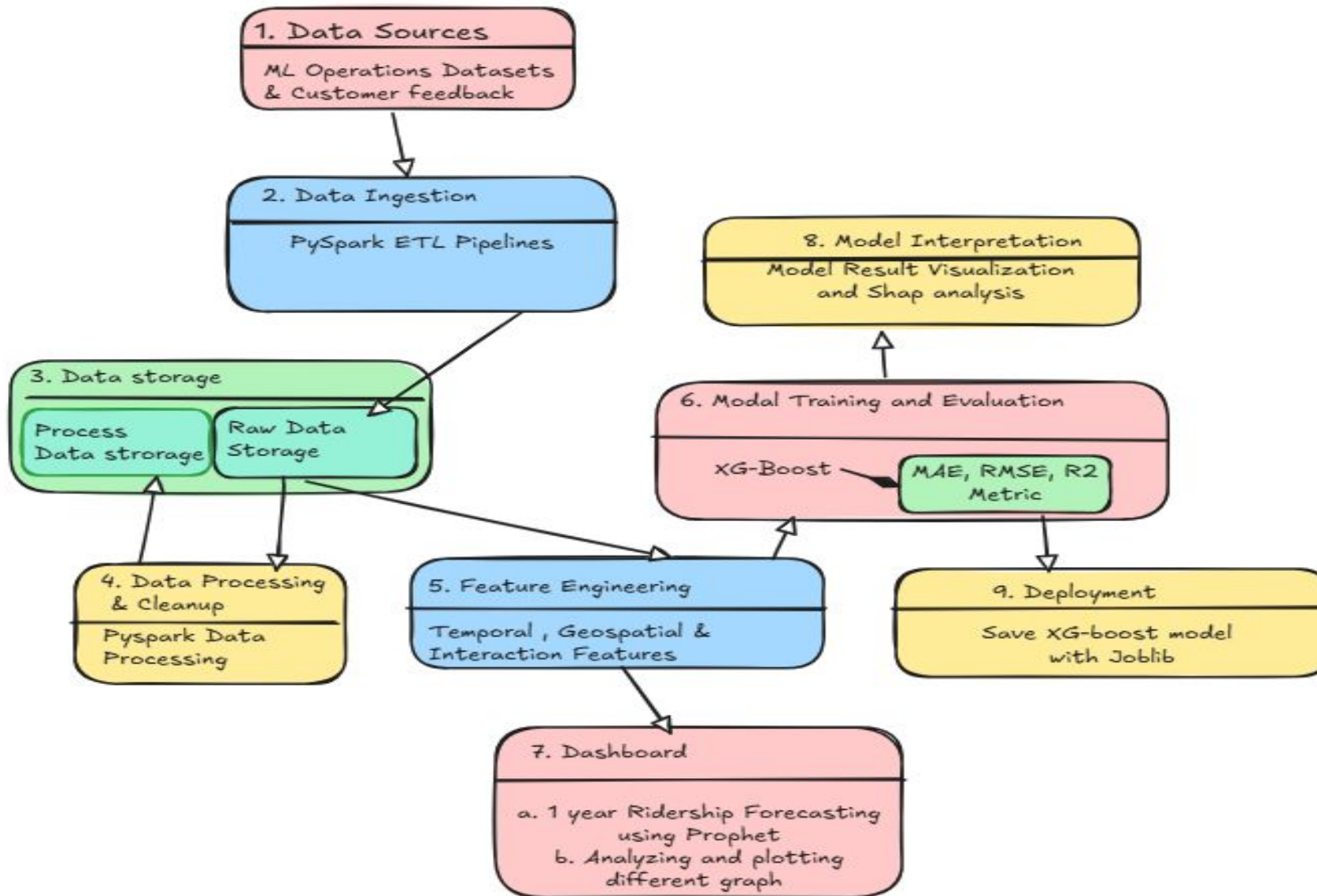
    shutil.rmtree("sampled_output_temp")

    print(f"Sampled data saved to {output_file}")

except Exception as e:
    print(f"An error occurred: {e}")

# Stop the Spark session
spark.stop()
```

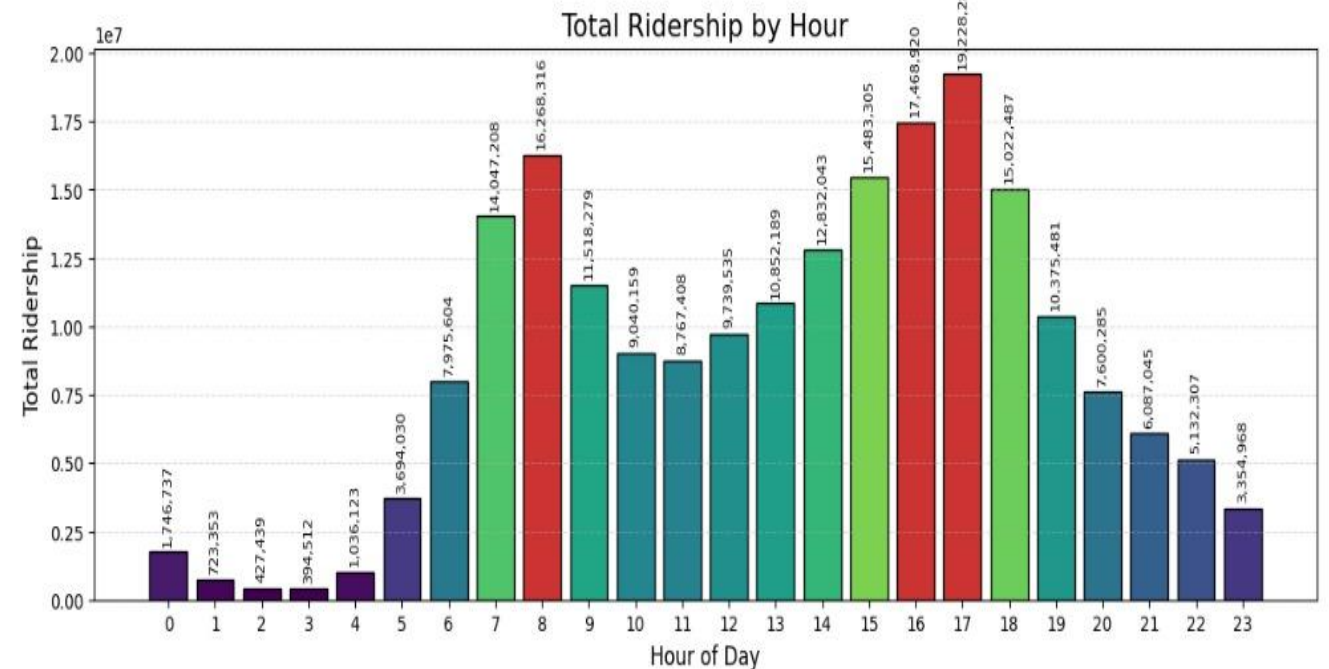
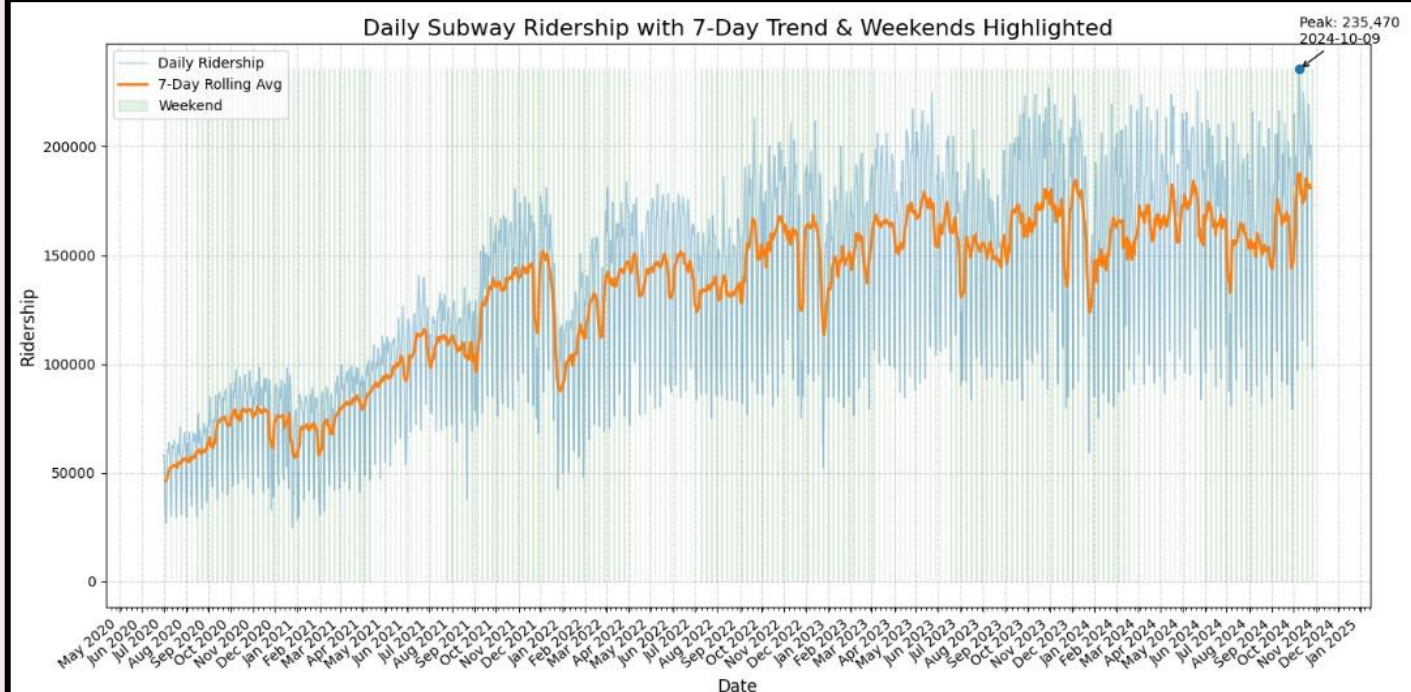
# ARCHITECTURE





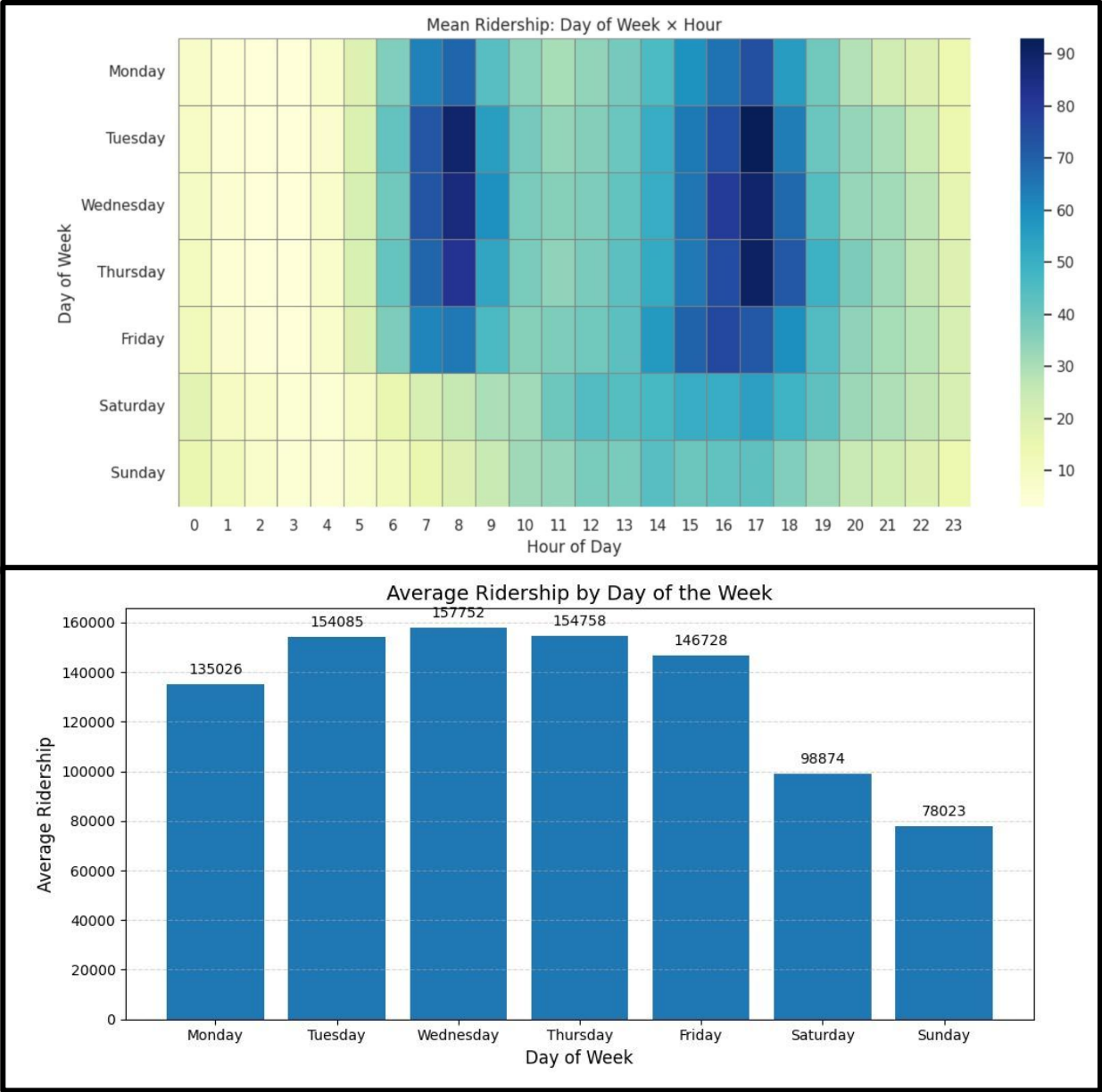
# EXPLORING RIDERSHIP: DAILY AND HOURLY TRENDS

- MTA ridership experienced about 140% growth from 2020 pandemic to 2024 November.
- Daily ridership has yet to return to pre-pandemic levels. Additionally, as more and more jobs demand return-to-office, the need to placate the growing demands of reliability becomes more important.



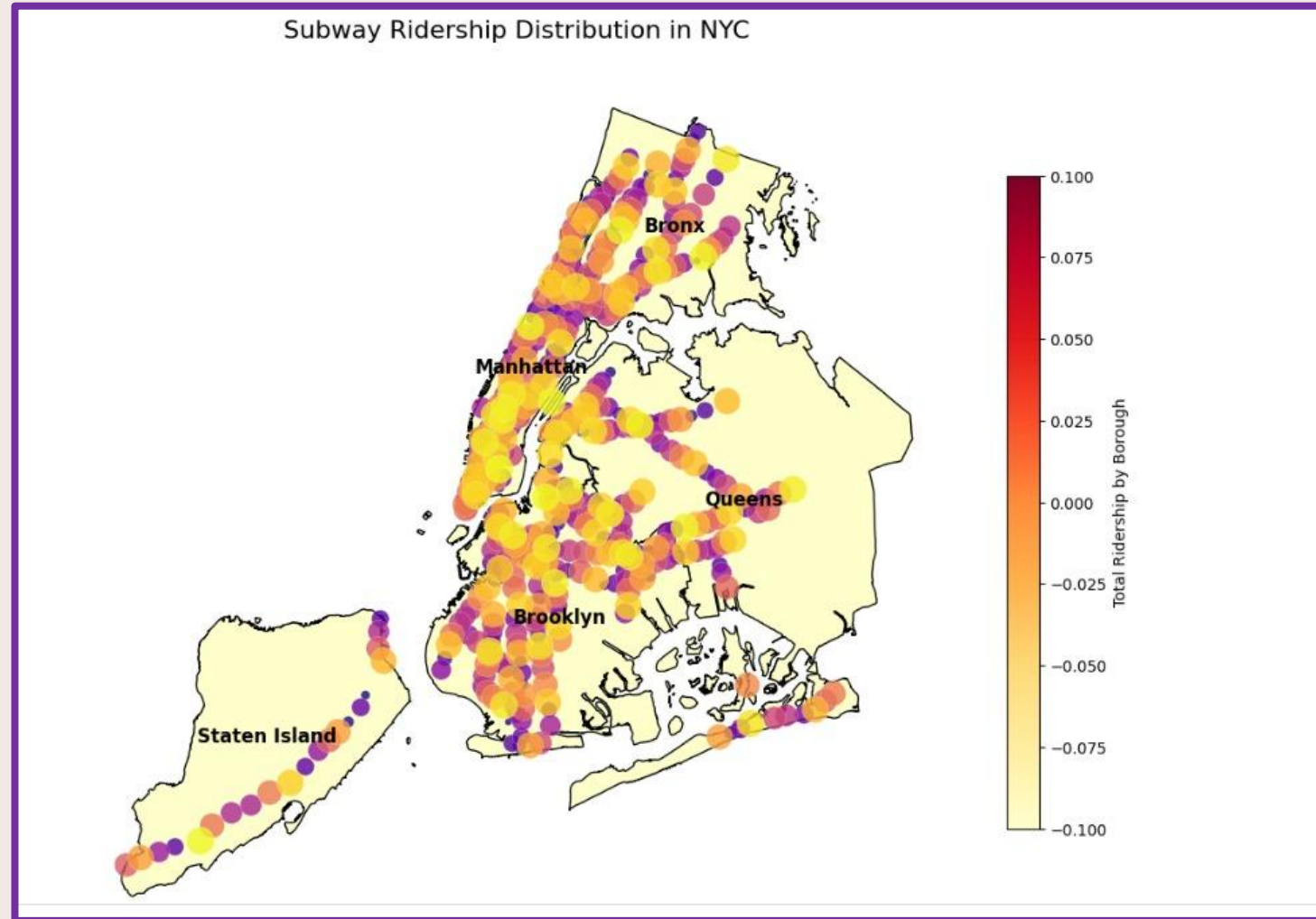
# EXPLORING RIDERSHIP: WEEKDAY PEAKS

- Ridership is the highest on weekdays, peaking on Wednesdays, reflecting typical work week commuting.
- This underscores the importance of on weekdays and hours when commuters travel from and to work as critical ridership periods, driven by work and business needs.

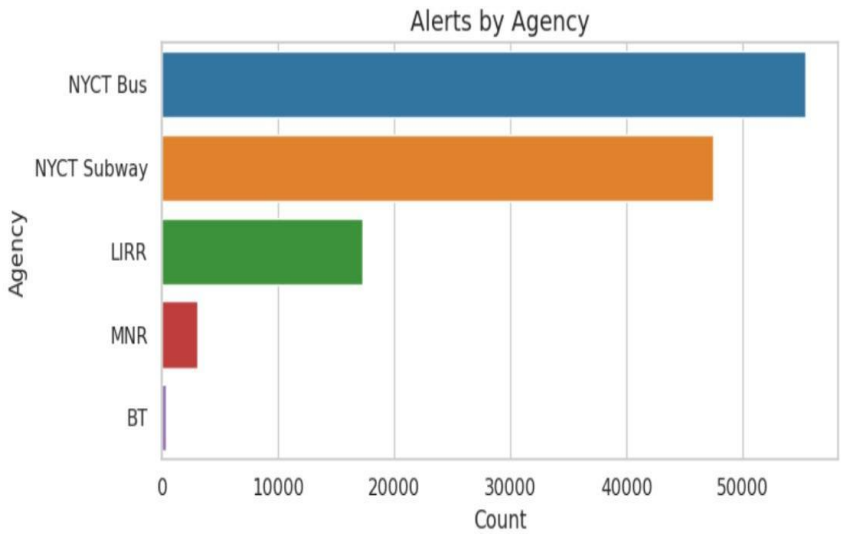
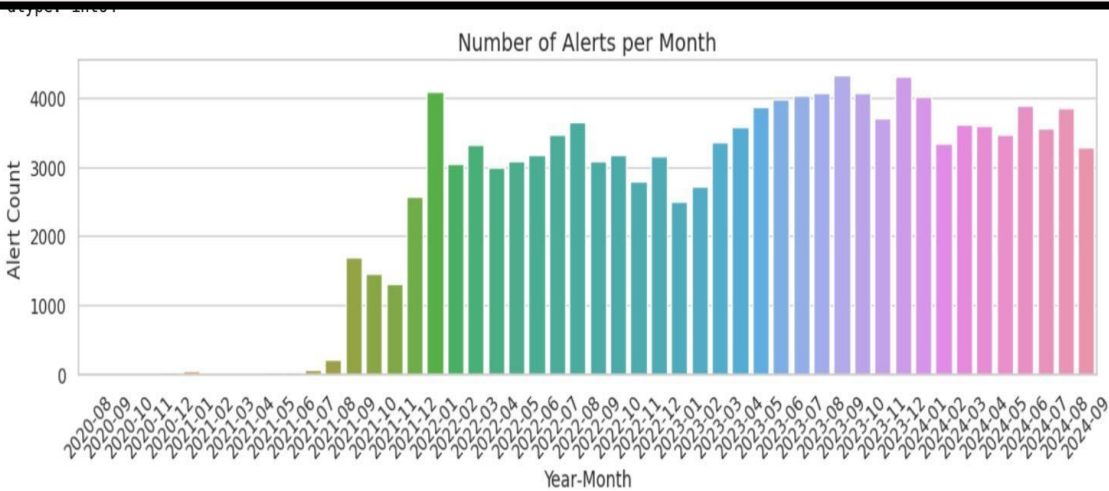




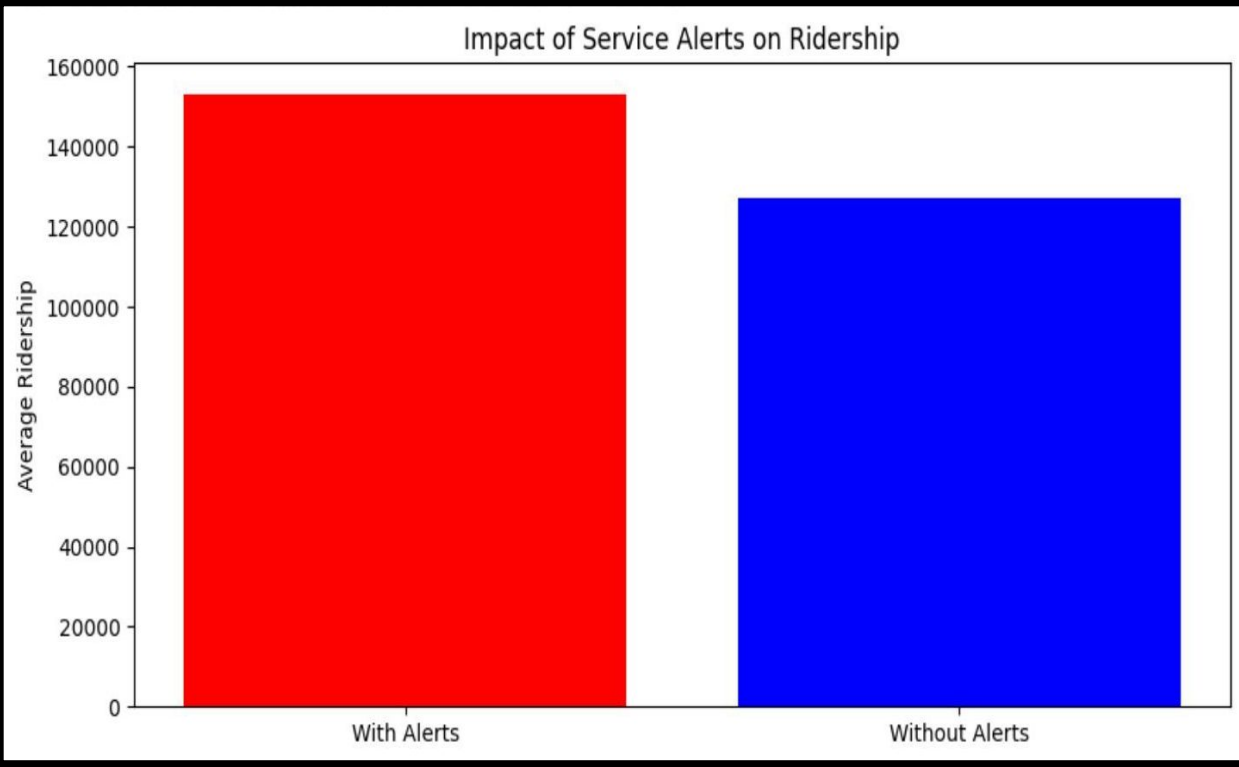
# Ridership Distribution



# EXPLORING RIDERSHIP: DELAYS AND ALERTS

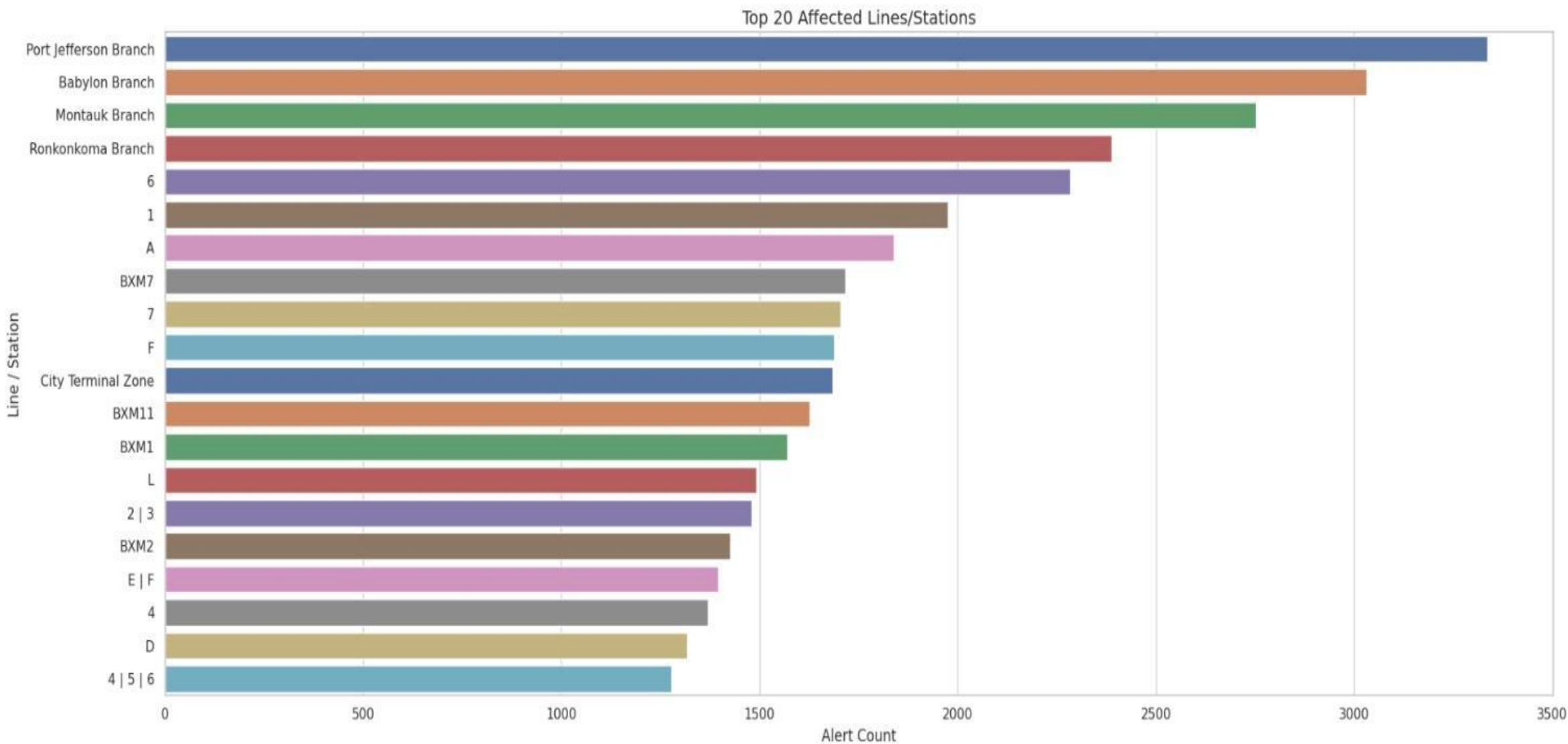


Status Alerts	Frequency
Delays	173,711
Some-Delays	48,120
Buses-Detoured	16,901
Weekday-Service	16,850
Essential-Service	8,219



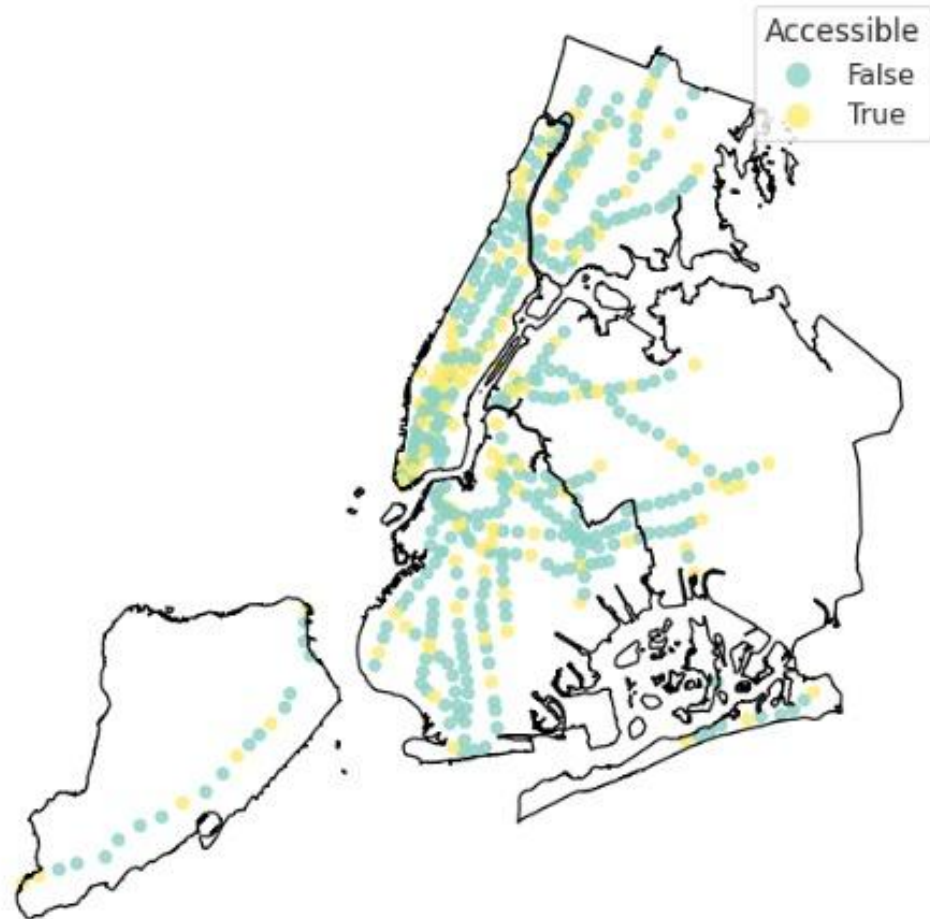


# Affected NYC Subway Stations

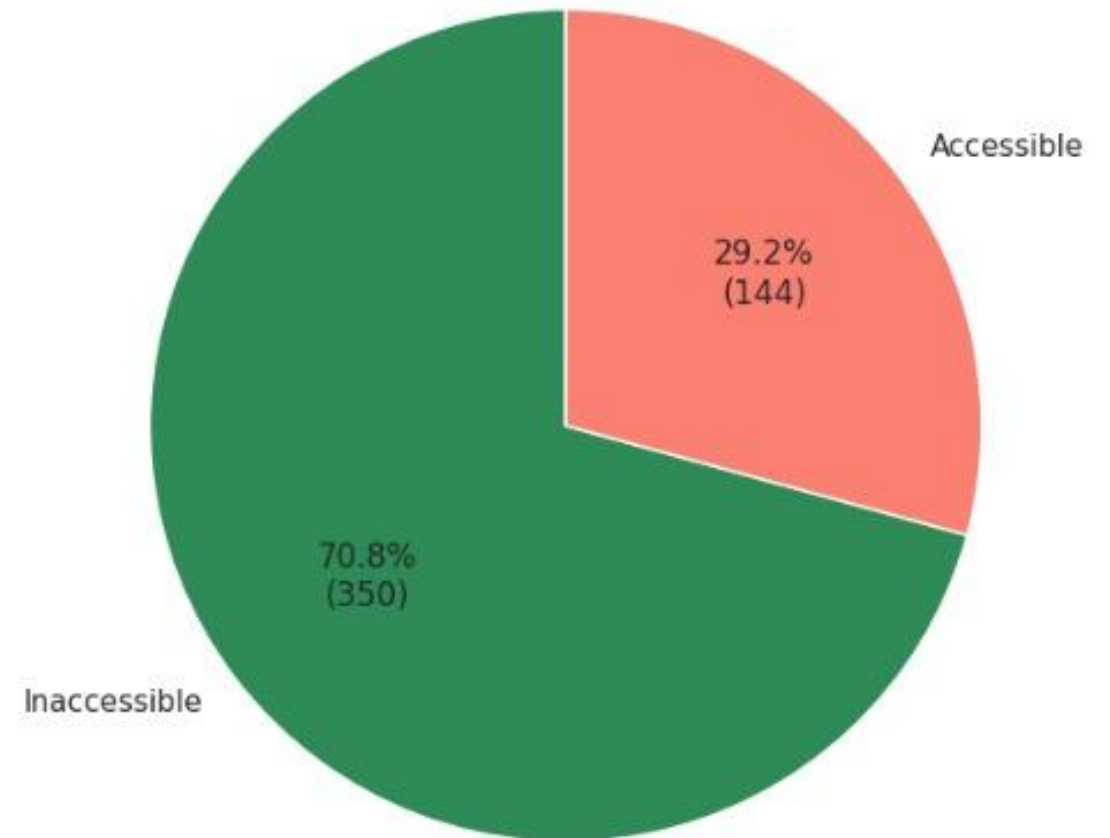


# EXPLORING RIDERSHIP: ADA ACCESSIBILITY

NYC Subway Stations: Accessible (yellow) vs Inaccessible (Teal Blue)

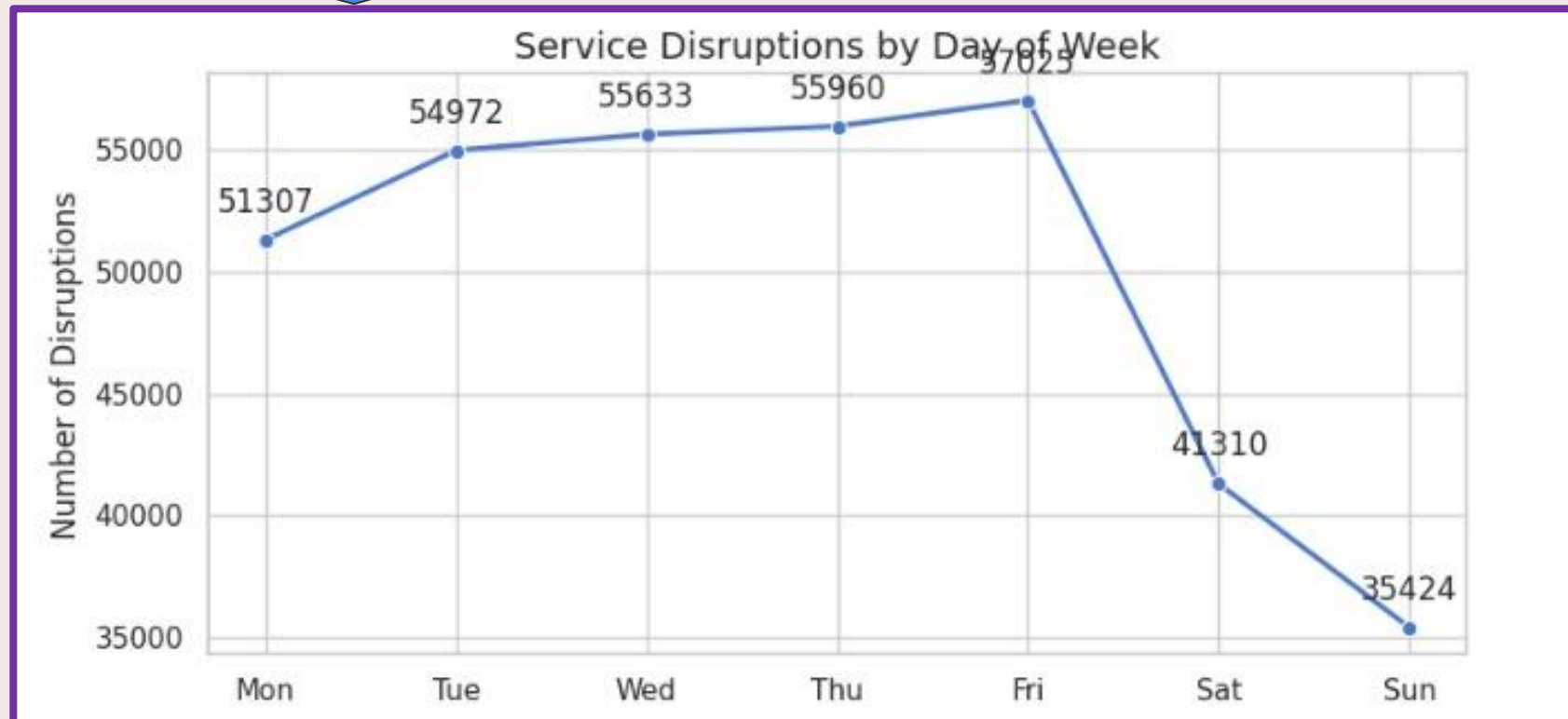


Station Accessibility Breakdown



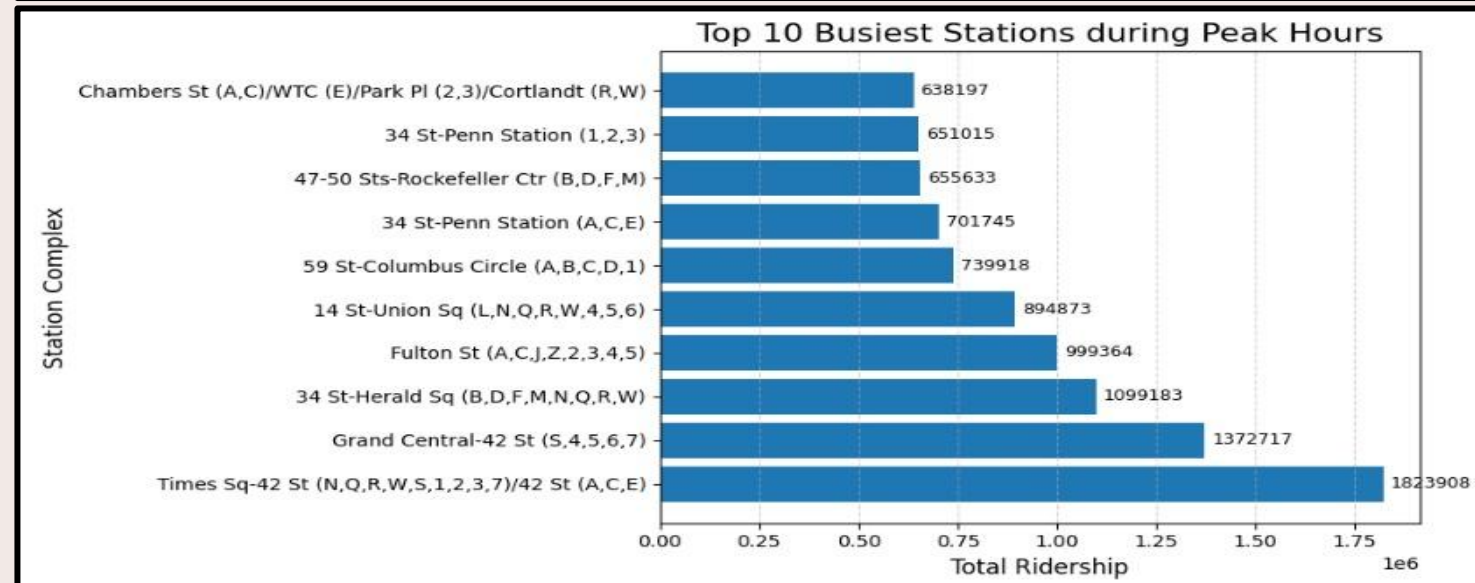
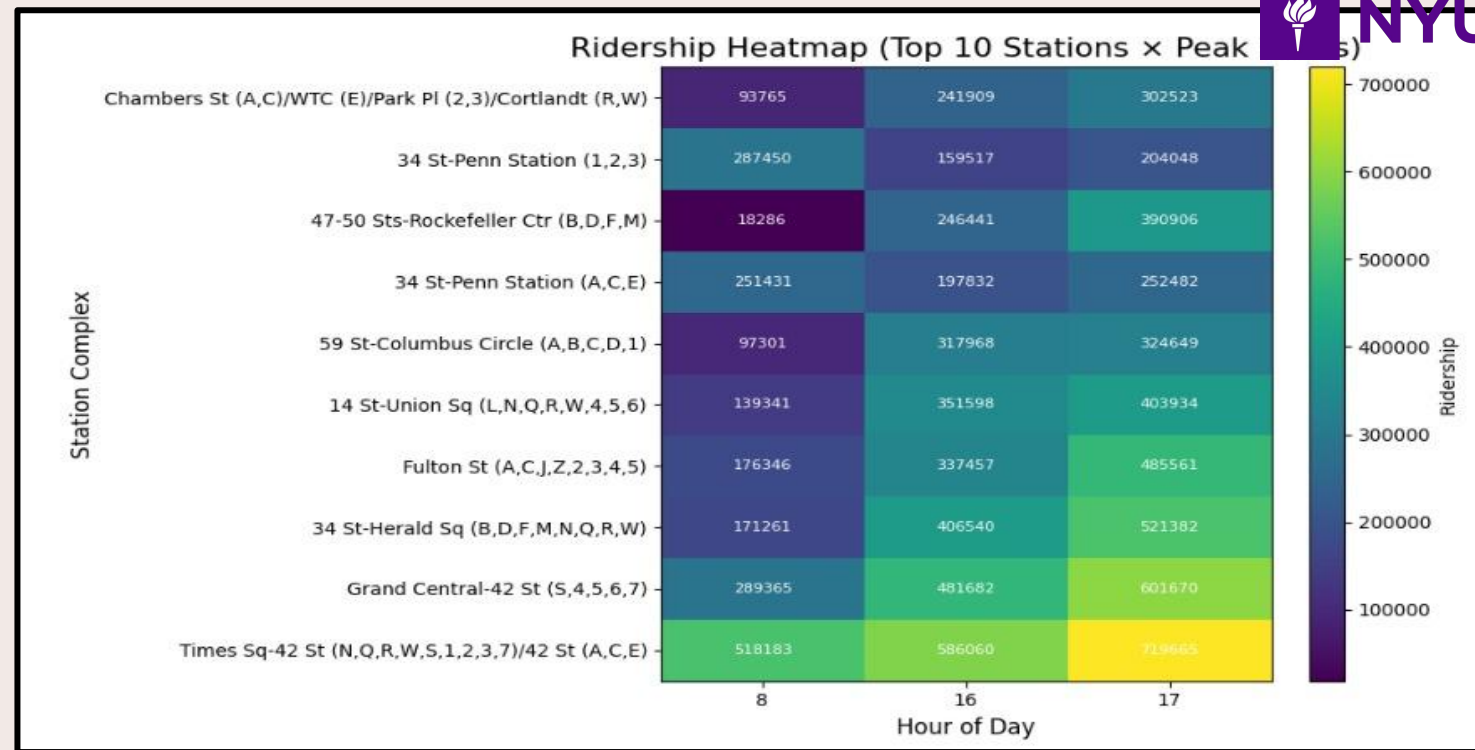
# Recommendations:

1. Prioritize elevator installs at the highest-traffic inaccessible stations.
2. Add ramps and tactile signage at medium-traffic locations.
3. Use spatial gaps in coverage to plan new accessible routes.



# EXPLORING RIDERSHIP: WHERE RIDERS ARE CONCENTRATED

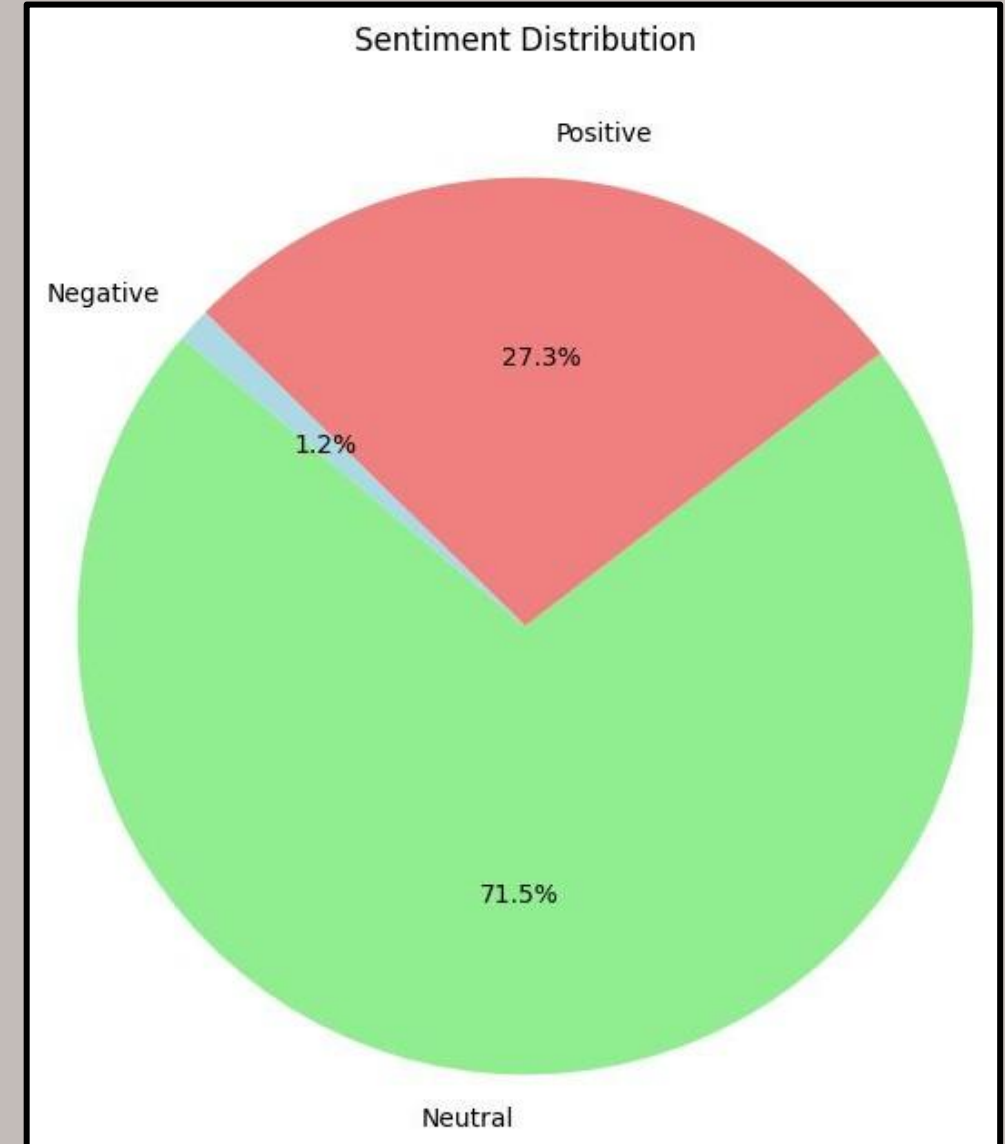
- Stations in Manhattan, like 42nd St and 34th St, remain critical focal points for managing crowd control and optimizing service during peak hours.
- Stations like Rockefeller Center and Chambers St have notable spikes in ridership, indicating there are localized surges that MTA can strategically allocate resources to





## RIDER SENTIMENT: MOSTLY NEUTRAL, ROOM FOR

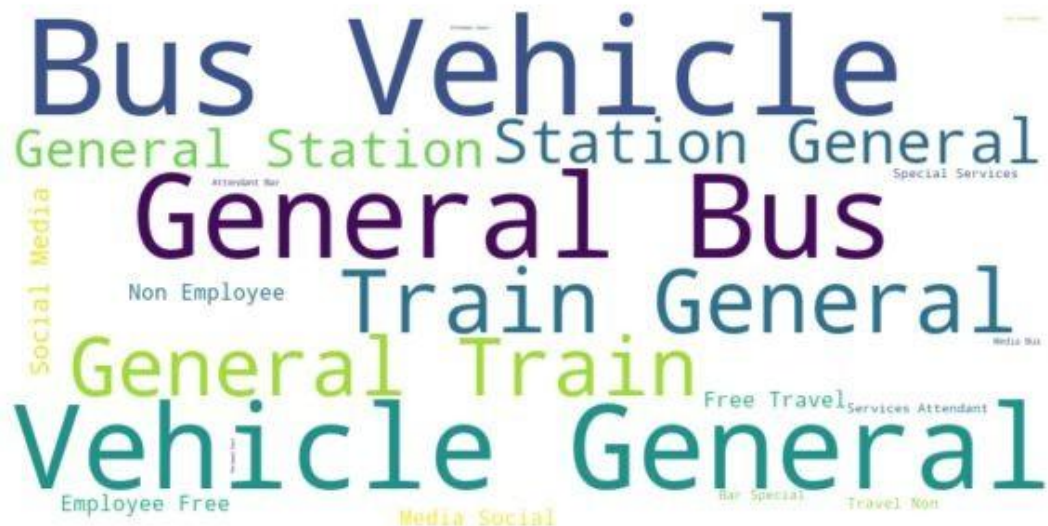
- Most riders express a neutral sentiment, indicating that while they are not dissatisfied, their experiences are not exceeding expectations.
- Negative sentiment at 1.2% shows that outright dissatisfaction is relatively rare.
- However, the overwhelming neutrality highlights opportunities to improve aspects of MTA to move more neutral riders toward positive experiences.



## RIDER SENTIMENT: THEMES IN FEEDBACK

- Within the positive word cloud, “Bus”, “Vehicle”, and “General” dominate could suggest riders feel that MTA do provide basic service reliability and functionality.
- Negative feedback is concentrated on issues of cleanliness, customer service, and operational inefficiencies, pointing to areas that MTA put more attention to improve.

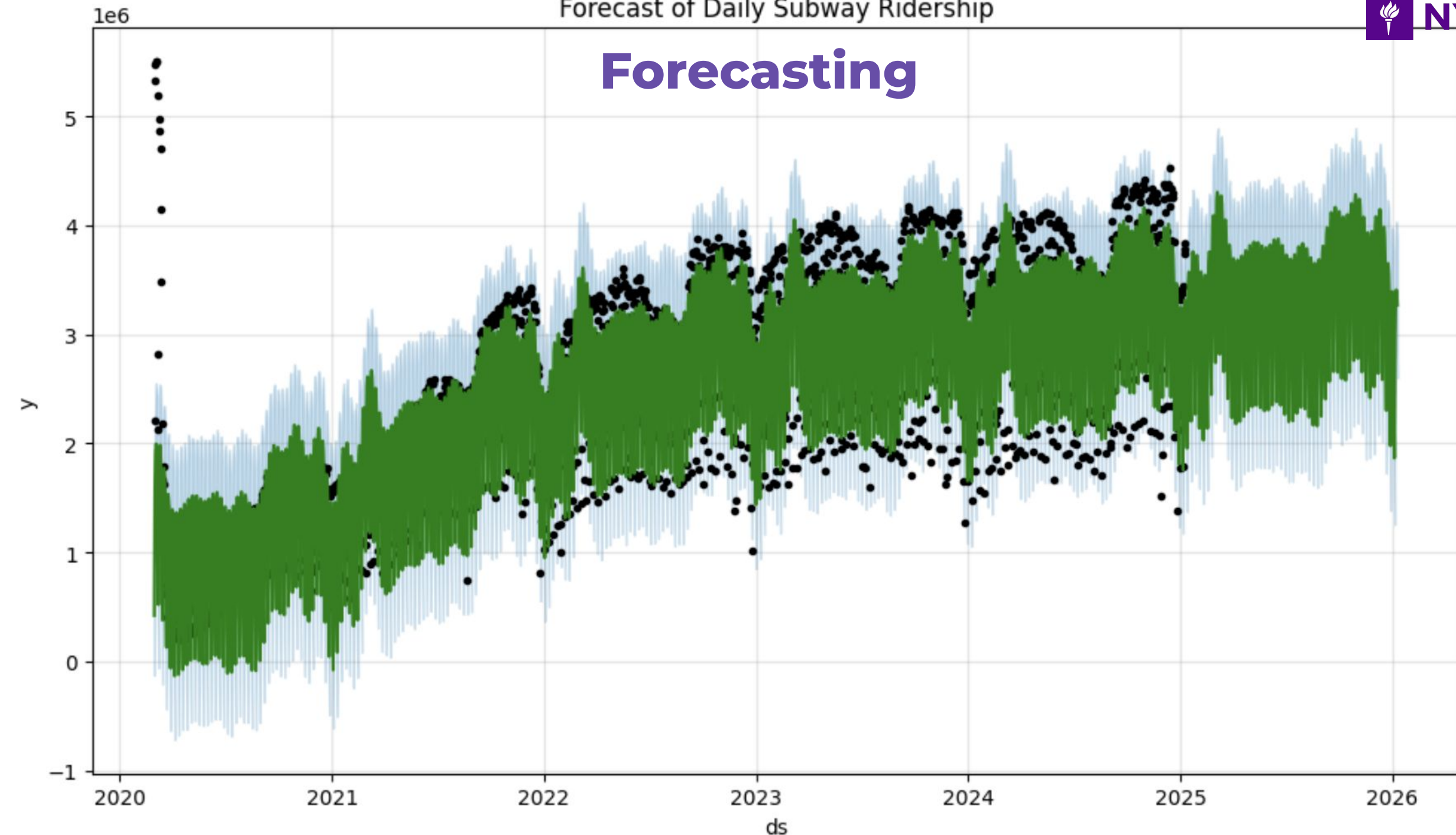
Positive Feedback Word Cloud



Negative Feedback Word Cloud



# Forecasting



# XGBOOST RIDERSHIP PREDICTION

Leveraged the XGBoost machine learning model to forecast daily estimated ridership for the MTA subway system with high accuracy.

Integrated big data sources such as ridership statistics, service alerts, customer feedback, and operational metrics to capture influencing factors.

Designed the model to identify complex patterns and provide actionable predictions using efficient large-scale data processing.

Enabled the MTA to optimize train schedules, allocate resources, and enhance service reliability based on forecasted demand.

```
from sklearn.metrics import r2_score

# Evaluate R2 Score
r2 = r2_score(y_test_cv, y_pred_cv)
print(f"XGBoost R2: {r2:.4f}")
```

XGBoost R<sup>2</sup>: 0.7727



# DISTRIBUTION:

- A normal distribution of residuals indicates that the errors are unbiased and randomly distributed, suggesting the model is well-fitted.
- There are no major skewness or outliers visible in the distribution.

```
import seaborn as sns
```

```
# Calculate Residuals
```

```
residuals = y_test_cv - y_pred_cv
```

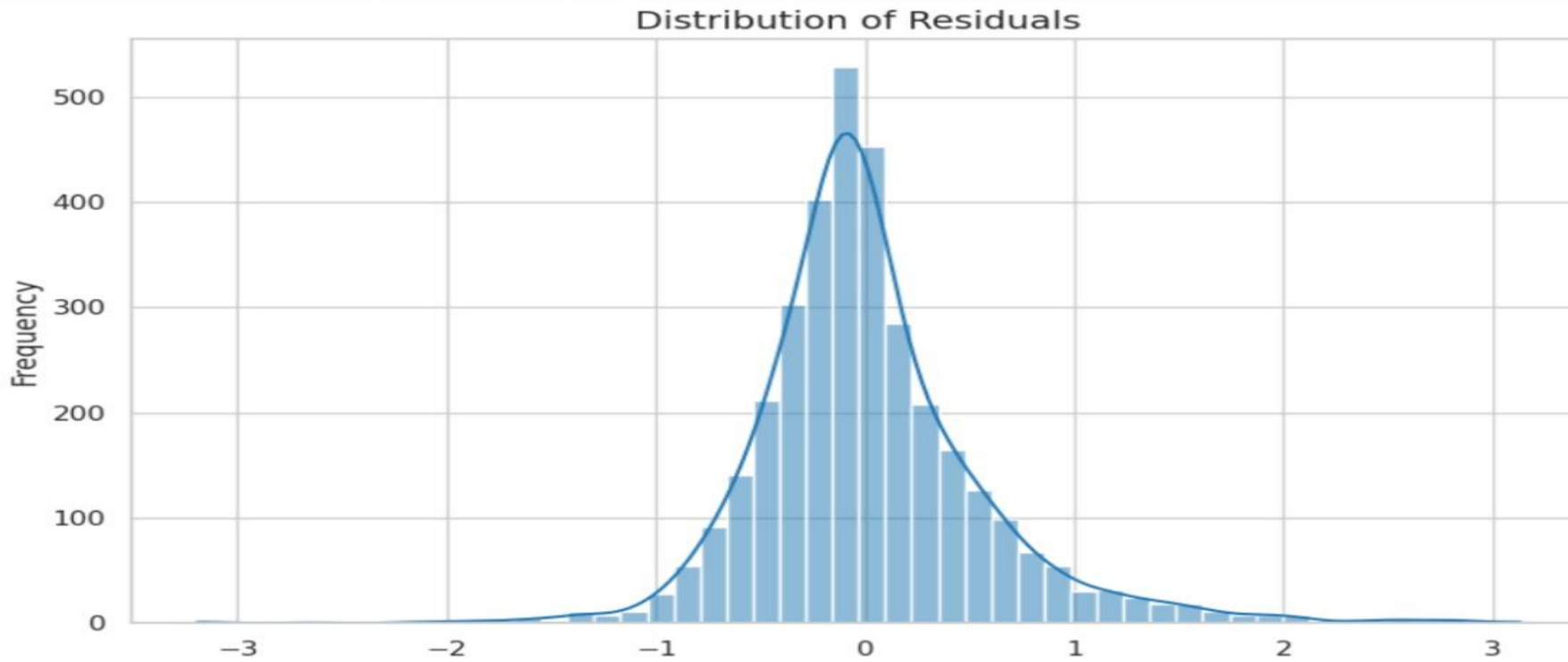
```
# Plot Residual Distribution
```

```
plt.figure(figsize=(10,6))
sns.histplot(residuals, bins=50, kde=True)
plt.title('Distribution of Residuals')
plt.xlabel('Residuals')
plt.ylabel('Frequency')
plt.show()
```

```
# Residuals vs Predictions Plot
```

```
plt.figure(figsize=(10,6))
sns.scatterplot(x=y_pred_cv, y=residuals)
plt.axhline(0, color='red', linestyle='--')
plt.title('Residuals vs. Predictions')
plt.xlabel('Predicted Values')
plt.ylabel('Residuals')
plt.show()
```

use\_inf\_as\_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.



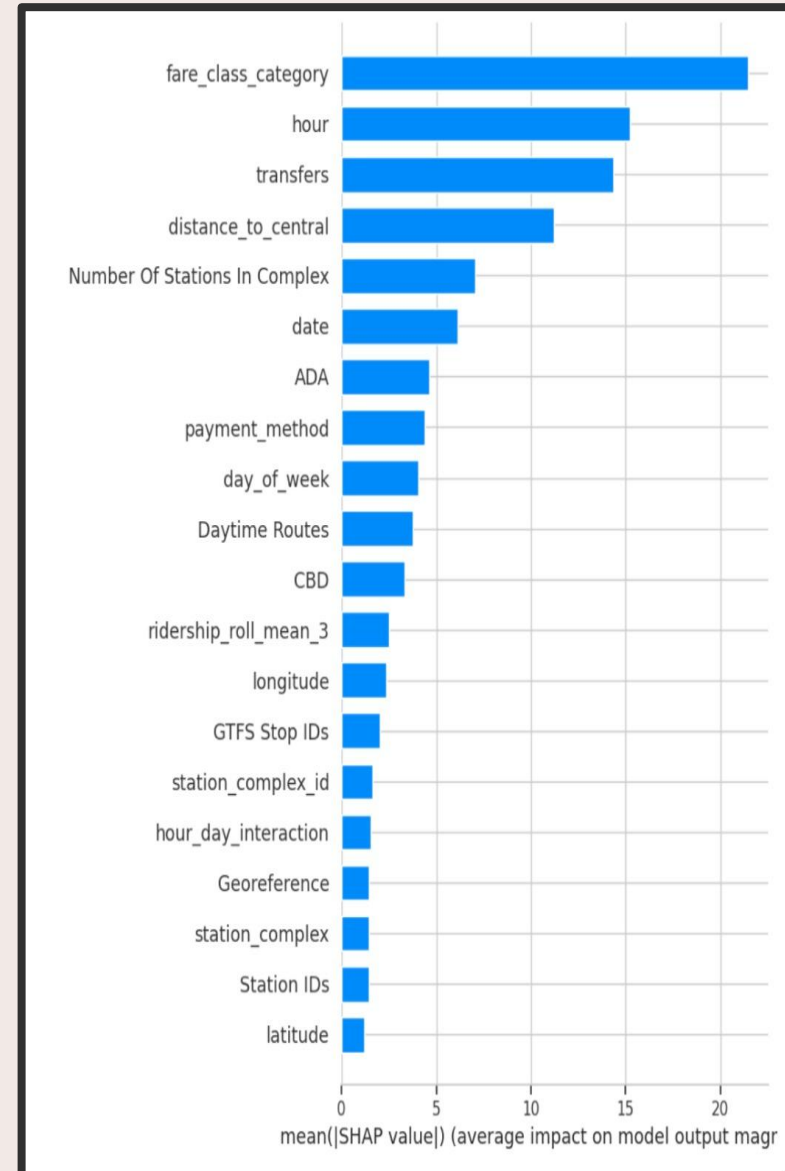
# FEATURE ANALYSIS USING SHAP ANALYSIS

The bar chart presents the mean absolute SHAP values for each feature, ranked in descending order. This shows the average contribution of each feature to the model's output.

**hour:** The time of day also significantly affects the model's decision.  
**transfers:** The number of transfers contributes substantially to the predictions.

Features such as Georeference, Stop Name, and Station IDs have minimal impact on the model's output.

By quantifying the importance of features, SHAP values provide transparency to a typically "black-box" model like XGBoost.



# RECOMMENDATIONS FOR IMPROVEMENT

## Operational Efficiency

Focus on reducing delays and efficiencies during peak hours and weekdays that reflect normal commuting times, especially in the busiest stations such as Times Square-42<sup>nd</sup> St and Grand Central-42<sup>nd</sup> St

## Cleanliness and Maintenance

Invest more into waste management and general maintenance to improve daily commutes; enforce basic etiquette and rules to discourage unauthorized activities such as merchandise sales

## Communication and Alerts

Improve real-time communication about delays and disruptions by leveraging mobile apps, digital signages to keep riders informed and reduce frustration

## Future Course of Actions:

- Incorporating real-time data streams for dynamic forecasting directly through an API.
- Sampling a larger set of data and training the model on a much larger training dataset.
- Extending analysis to include additional factors like weather, events, and economic indicators.
- Implement Reinforcement Learning by analyzing their predictions and connecting it to the MTA data stream API.



**THANK YOU!**

---

