CSGY 6513 - Optimizing Urban Transit: A Big Data Approach to MTA Subway Ridership and Service Analysis PROJECT REPORT

Team Members

Jay Daftari(jd5829)

Akshat Mishra (am5218)

Nikita Gupta (ng3230)

Link to PowerPoint:Power Point Link
Code Repository: Github Repository

Table of Contents

- SHAP Local Explanations

7. **Lessons Learned**

8. **Future Improvements**

    - Enhanced Predictive Modeling
    - Real-Time Data Integration
    - Accessibility Enhancements
    - User-Centric Dashboard Features
    - System Scalability

9. **Data Sources and Results**

    - Ridership & Station Data
    - Service & Operations Data
    - Performance & Feedback Data
    - Data Source URLs
    - Daily and Hourly Ridership Trends
    - Delays and Alerts Impact
    - ADA Accessibility Analysis
    - High-Traffic Stations Analysis
    - Rider Sentiment Overview
    - Rider Sentiment Themes
    - Operational Efficiency
    - Cleanliness and Maintenance
    - Communication and Alerts
    - Feature Importance Analysis Using SHAP
    - Model Overview
    - Key Features and Benefits
    - Performance Metrics
    - Applications
    - Forecast of Daily Subway Ridership
    - Residual Analysis of Model Predictions

10. MTA Ridership Explorer Dashboard

    - Dashboard Overview
    - Key Features
    - Practical Applications

1.Executive Summary

The New York City Subway system, operated by the Metropolitan Transportation Authority (MTA), serves as the backbone of the city's transportation network.
With over 8 million daily riders and 472 subway stations, managing congestion and service disruptions is a critical challenge.

This project leverages big data analytics to address the following:

1. Predicting and managing subway congestion in real time.
2. Understanding the impact of service disruptions on ridership.
3. Identifying gaps in accessibility for individuals with disabilities.
4. Enhancing customer satisfaction through feedback analysis and actionable insights.

The project integrates historical data, real-time ridership metrics, and sentiment analysis to propose data-driven solutions for a more reliable and efficient subway system.

Objectives:

- Characterize daily/weekly/hourly ridership trends.
- Quantify ridership drops due to service alerts.
- Map ADA-compliance and its ridership effects.
- Perform sentiment analysis on rider feedback.
- Build an XGBoost forecasting model with SHAP interpretability.
- Offer actionable recommendations for operations and infrastructure.

Technologies Used:

- **Data Processing:** PySpark
- **Modeling & Analysis:** Python, XGBoost, SHAP, scikit-learn
- **Visualization:** Matplotlib, GeoPandas
- **Version Control & CI/CD:** Git/GitHub

2. Code Execution Instructions

README URL: Full instructions and code structure are available at: github readme link

How to Run:

1. Clone the repository:
2. Download the data(Provided in Readme and in Data section)
3. Run all of the ipynb file, by setting the correct path of data. (Since data is too huge to upload it on github.)
4. Download dependency using code `pip install streamlit pandas plotly streamlit-folium folium prophet` for dashboard_streamlit.py
5. Run the dashboard_streamlit.py to see the live dashboard.

3. Technological Challenges

- **Scaling Data Processing:**
    - *Issue:* Initial prototyping with Pandas ran out of memory on 30 GB turnstile logs.
    - *Solution:* Switched to PySpark, sampling 5% for EDA and full runs on a 4‑node cluster.
    - *Lesson:* Early choice of scalable tools avoids late‑stage reengineering.
- **Model Interpretability Overhead:**
    - *Issue:* Computing SHAP values for ~10 million rows was prohibitively slow.
    - *Solution:* Employed stratified sampling for SHAP and parallelized computations.
    - *Lesson:* Interpretability tools need performance tuning at scale.

4. Changes in Technology:

Original Proposal	Actual Implementation	Rationale & Impact
Python 3, Pandas, Dask, NumPy, SciPy	PySpark	Pandas/Dask on single node ran out of memory on 30 GB+ datasets. PySpark on a 4‑node cluster scaled reliably.
PyTorch	XGBoost	Shifted from deep learning to gradient‑boosted trees for tabular forecasting—yielded higher R^2 (0.77) with far less training time.

scikit-learn	scikit-learn + SHAP	Retained scikit-learn for utility; added SHAP for model interpretability.
Matplotlib, Seaborn, Dash, Tableau	Matplotlib + GeoPandas	GeoPandas enabled richer geospatial maps; dropped Seaborn/Dash/Tableau to streamline dependencies and automate plot generation.

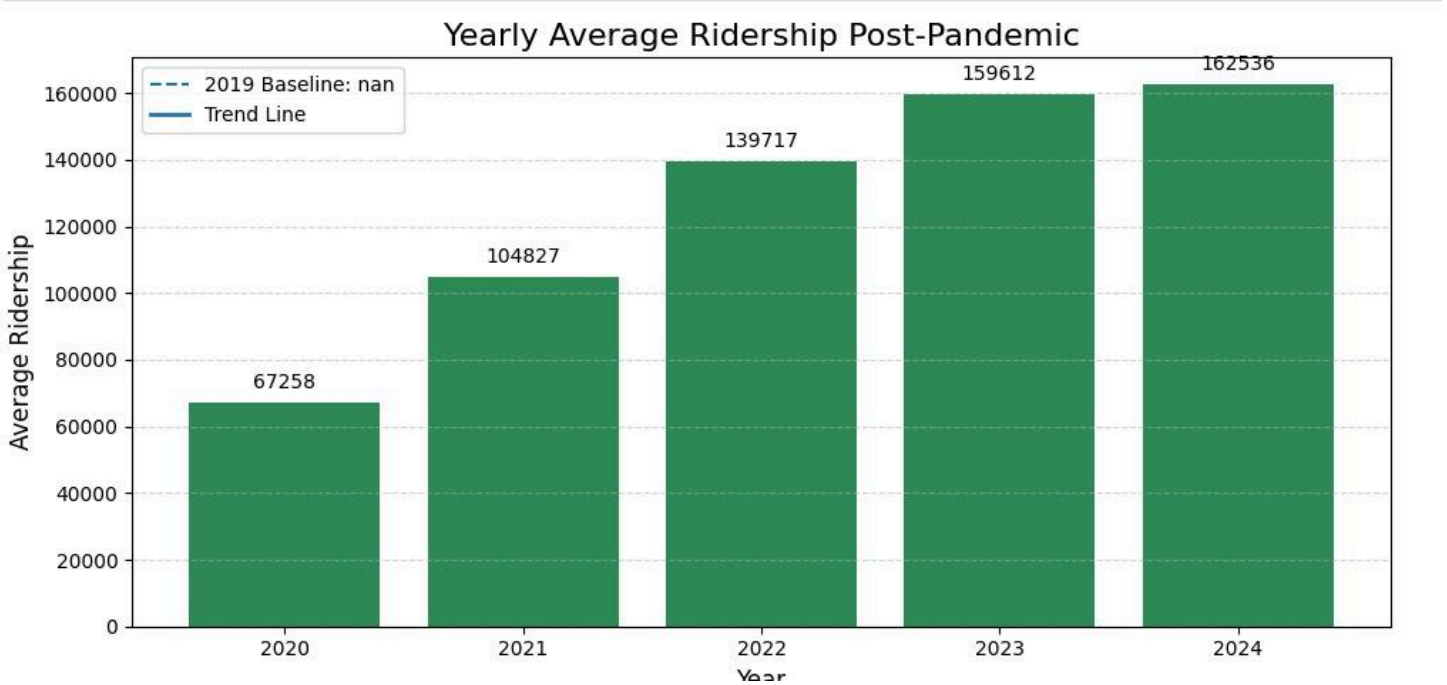
These changes were driven by the need for scalable data processing, faster model iteration, deeper interpretability, and reliable production-grade performance.

5. Uncovered Aspects from Presentations

During our live demo, time constraints prevented us from showing:

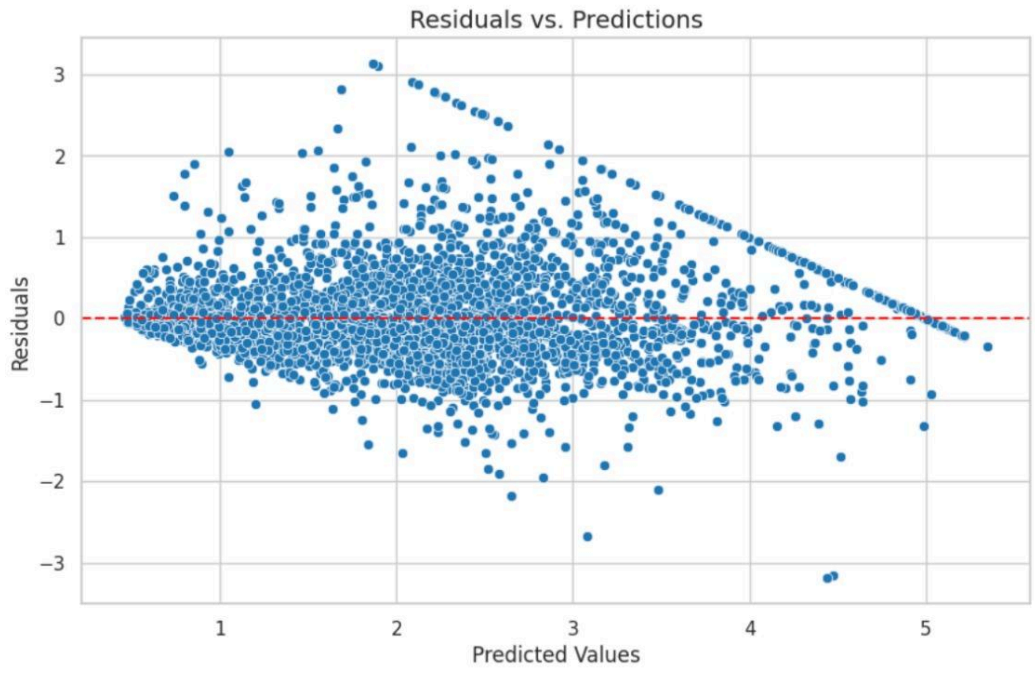
- Yearly Average Ridership Post-Pandemic:

  This bar plot shows the yearly average ridership after the pandemic, highlighting a significant increase in ridership in the following years.



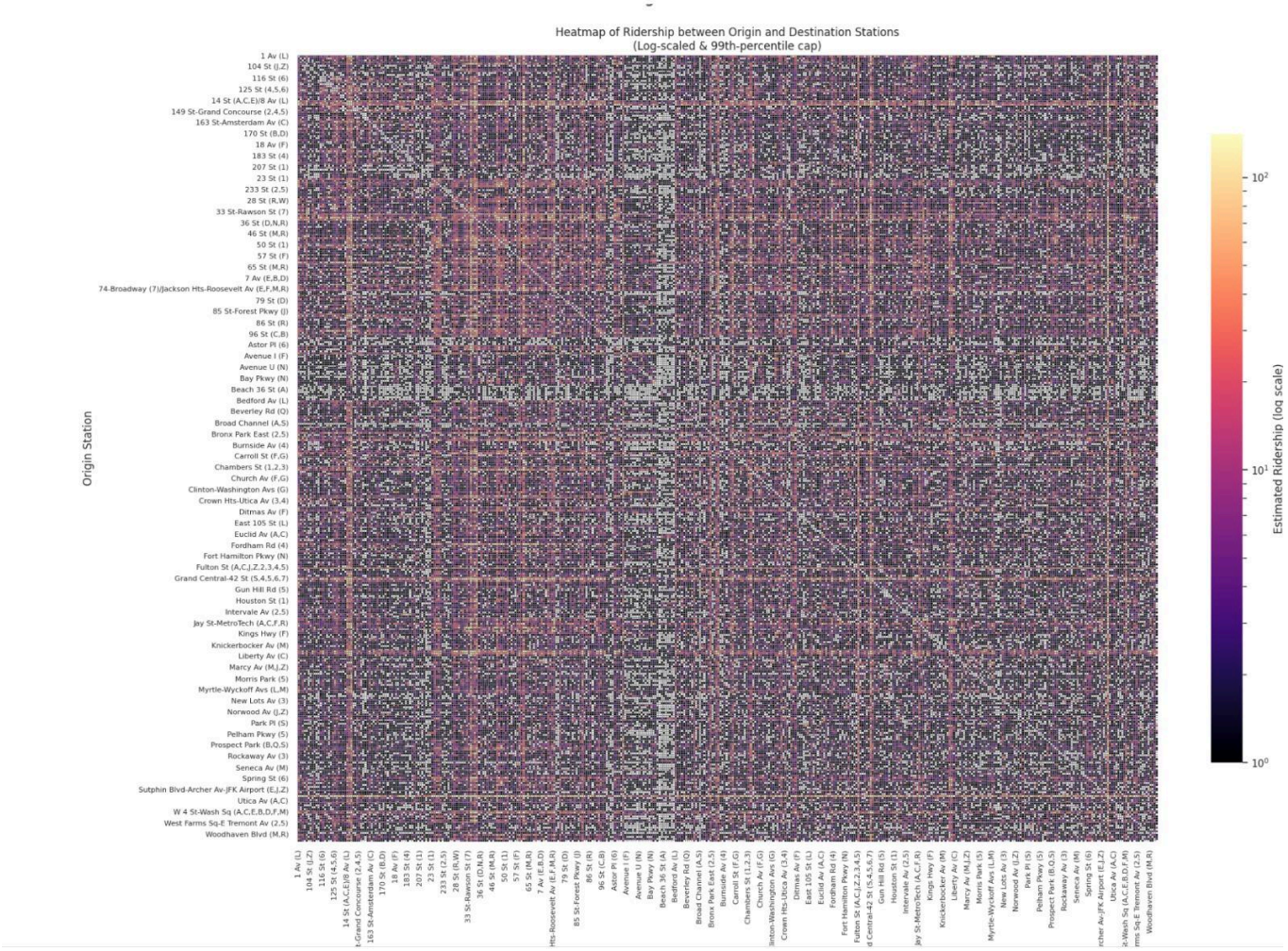
- **Residual vs Predictions graph:**

  This shows how far the forecast deviates from the actual number of riders. The model appears to be quite accurate, as the residuals are mostly centered around the x-axis, indicating that the prediction errors are close to zero.



- **Heatmap of Origin-Destination Relationships:**

This heatmap illustrates the aggregated ridership between origin and destination stations across the entire network. The values are log-scaled and capped at the 99th percentile to reduce the impact of extreme outliers.



- **SHAP Local Explanations:**

  In the presentation, we only showed the overall feature contributions for forecasting, without distinguishing between positive and negative correlations.

6. Lessons Learned

- **What Worked Well:**
  - Modular PySpark pipelines accelerated data experiments.

- SHAP explanations provided trust in model predictions.

- **Challenges & Overcoming Them:**
  - Hyperparameter tuning with large datasets was slow → used randomized search on subsamples.
  - Balancing EDA depth and presentation scope → focused on top use cases.
  - Coordinating a multi-member codebase → adopted strict branch-per-feature Git workflow.

- **Key Insight:**
  - Early investment in scalable infrastructure pays dividends when data volume grows.

7. Future Improvements

1. **Enhanced Predictive Modeling:**
   - Use deep learning models for congestion and delay predictions.
   - Incorporate external factors such as weather and events.
2. **Real-Time Integration:**
   - Fully integrate Apache Kafka for real-time data streaming.
   - Deploy predictive models on live data pipelines.
3. **Accessibility Improvements:**
   - Expand accessibility analysis to include real-time elevator and escalator outages.
   - Collaborate with MTA to prioritize upgrades for high-traffic, non-compliant stations.
4. **User-Centric Enhancements:**
   - Develop a rider-facing dashboard for real-time congestion and service updates.
   - Integrate sentiment feedback into actionable alerts.
5. **Scalability:**
   - Extend the system to analyze other MTA services, such as buses and commuter rails.

8. Data Sources and Results

Code Repository: Github Repository

📊 Ridership & Station Data

- **MTA Daily Ridership Data (Post-2020)**
  - Access daily ridership data for various MTA services from March 2020 to January 2025.
  - **View Dataset**

- **MTA Hourly Ridership Data (Post-July 2020)**
  - Explore hourly ridership data for MTA services starting July 2020.
  - **View Dataset**

- **MTA Subway Turnstile Usage Data**

    - Analyze subway turnstile usage data to understand station-level ridership patterns.
    - View Dataset

- **MTA Subway Stations**
    - Access information about all subway stations, including locations and service details.
    - View Dataset

- **MTA Subway Stations and Complexes**
    - Explore data on subway stations and their interconnected complexes.
    - View Dataset

🛠️ Service & Operations

- **MTA Service Alerts Data (Post-2020)**
    - Stay informed with service alerts data for MTA services from 2020 onward.
    - View Dataset

- **MTA Subway Major Incidents Data**
    - Review data on major subway incidents that affected service.
    - View Dataset

- **MTA Subway Origin-Destination Data**
    - Analyze origin-destination data to understand subway travel patterns.
    - View Dataset

- **MTA Subway and Bus Lost Time Data**
    - Examine data on lost time for subway and bus services.
    - View Dataset

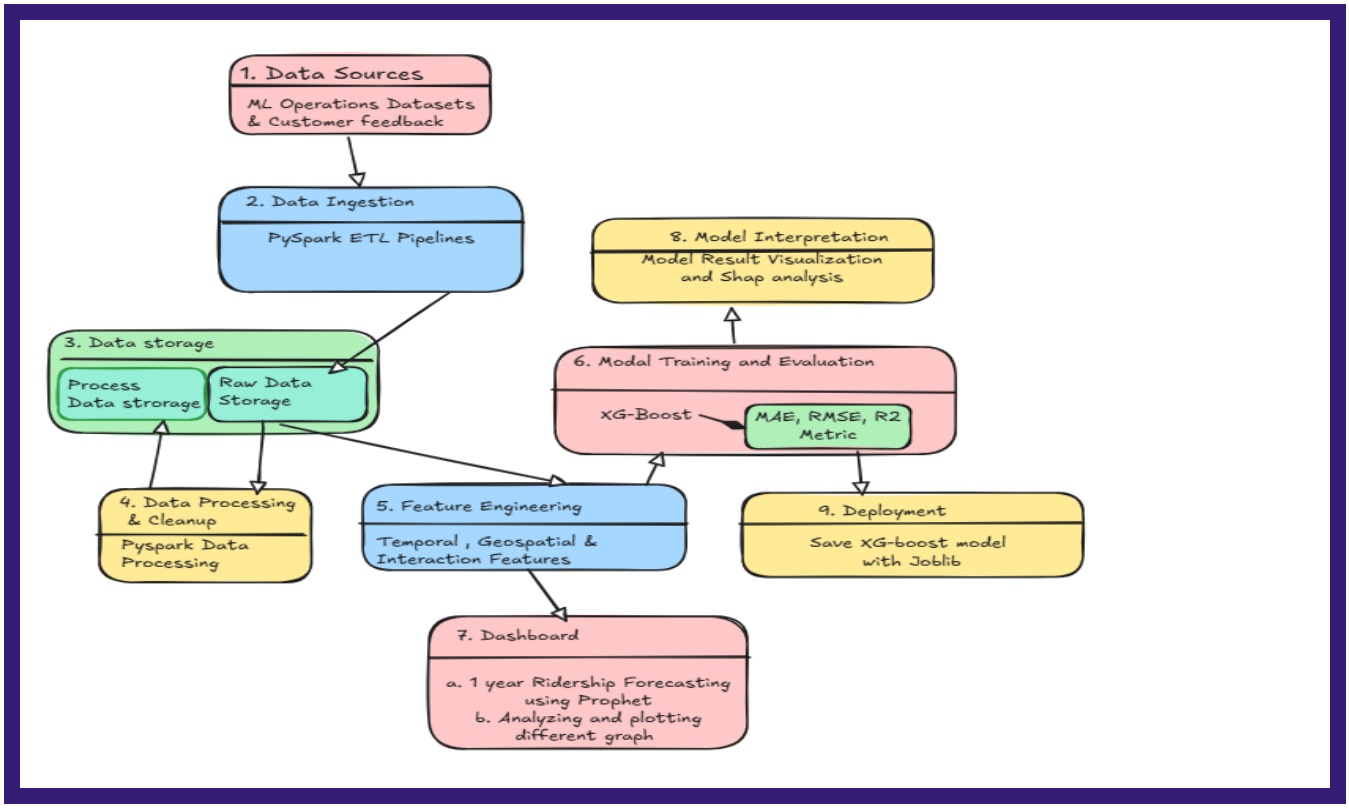
📈 Performance & Feedback

- **MTA Key Performance Indicators (2008–2021)**
    - Access key performance indicators for MTA services from 2008 to 2021.
    - View Dataset

- **MTA Subway Customer Journey Metrics**
    - Explore customer journey metrics for the subway system**.**
    - View Dataset

- **MTA Customer Feedback Data (2014–2019)**
    - Review customer feedback data, including complaints and commendations.
    - View Dataset

Architecture:



Exploring Ridership: Daily Trends

Analysis

Weekday ridership patterns reveal significant insights into commuter behavior. The data shows:

- **Peak Ridership on Weekdays:** Ridership is consistently higher on weekdays, with a notable peak on Wednesdays. This trend underscores the reliance on subway systems for regular workweek commuting.
- **Lower Ridership on Weekends:** As expected, weekend ridership is considerably lower, reflecting reduced demand from work-related travel and increased leisure-based activities.

Hourly Trends

- Morning and Evening Rush Hours: The data highlights two distinct spikes during typical commuting hours:
    - Morning Peak (7 AM - 9 AM): Reflects the inbound commute to workplaces.
    - Evening Peak (5 PM - 7 PM): Represents the outbound commute back home.
- Off-Peak Hours: There is a noticeable dip in ridership during midday and late-night hours, indicating reduced service utilization during these times.

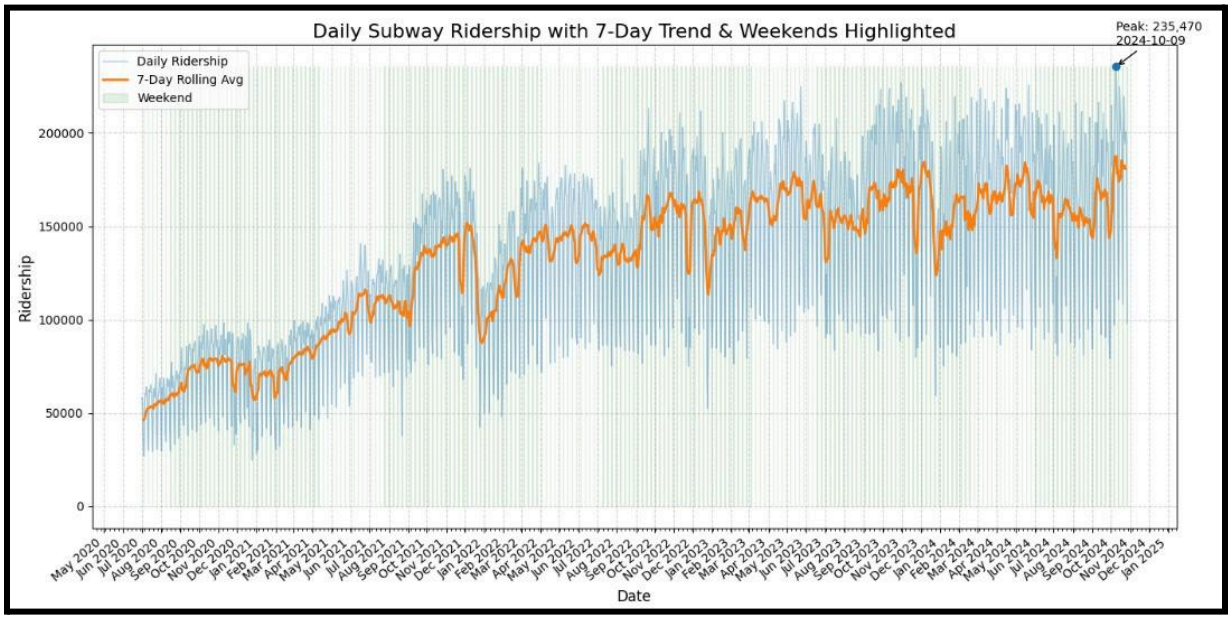
Implications

- **Operational Adjustments:** These trends can inform operational decisions, such as allocating additional resources during peak hours or optimizing schedules for off-peak periods.

- **Infrastructure Planning:** Understanding commuter behavior can aid in improving infrastructure, such as platform crowd management during rush hours.
- **Commuter-Focused Services:** Insights into weekday patterns allow for targeted improvements in reliability and communication, enhancing the overall commuter experience.

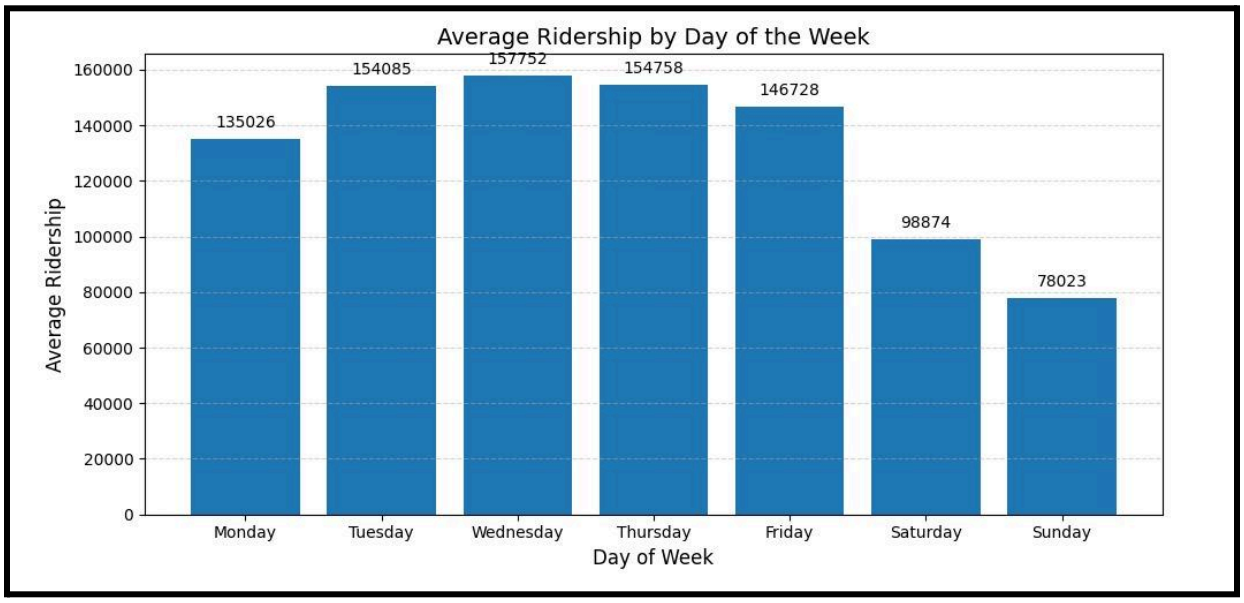
Visualizations

1. **Average Ridership by Day of the Week:**
    - A bar chart showcasing higher ridership on weekdays, with a clear peak on Wednesdays.
2. **Ridership by Hour of the Day:**
    - A detailed histogram illustrating hourly ridership patterns, emphasizing the morning and evening rush hours.

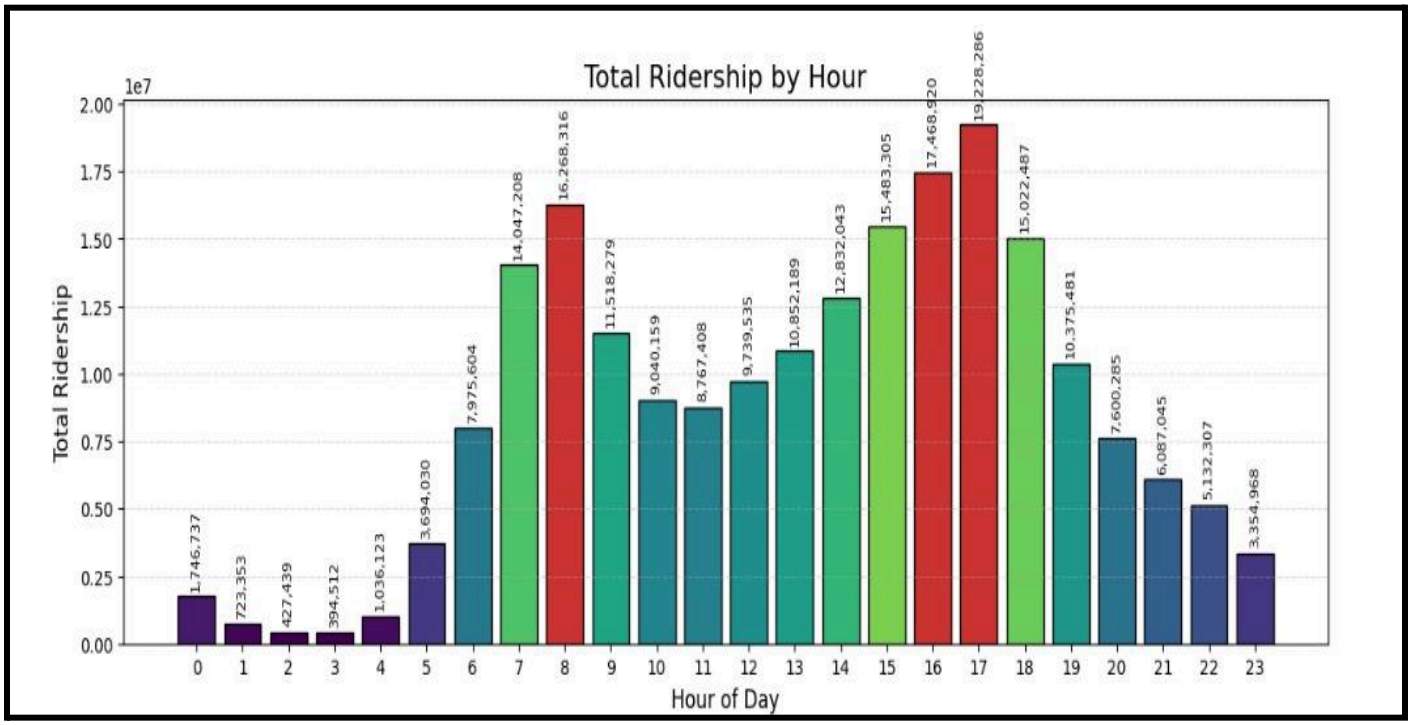


Visualizations

1. **Average Ridership by Day of the Week:**
    - A bar chart showcasing higher ridership on weekdays, with a clear peak on Wednesdays.

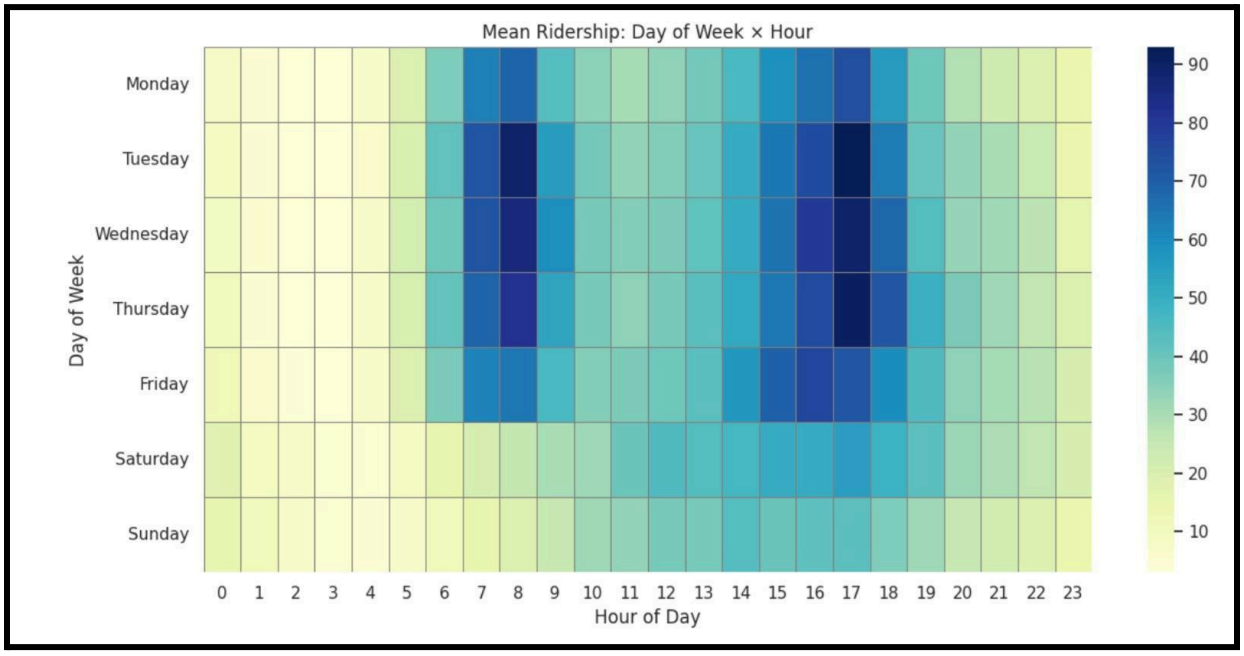


2. **Ridership by Hour of the Day:**
    - A detailed histogram illustrating hourly ridership patterns, emphasizing the morning and evening rush hours.



3. Mean Ridership Day of Week x hour:

- Drilling into weekdays, Wednesdays show the highest average usage—likely tied to mid-week business activities. Both morning (7–9 AM) and evening (5–7 PM) hours exhibit sharp spikes, underscoring the need for additional capacity during these windows.



Exploring Ridership: Delays and Alerts

Analysis

- **Prevalence of Alerts:**
  - Delays represent the most frequent type of service alert with **173,711 occurrences**, underscoring a consistent disruption in MTA services.
  - Other types of alerts, such as "Some-Delays" (**48,120 occurrences**) and "Buses-Detoured" (**16,901 occurrences**), reflect secondary operational issues.
- **Impact on Commuters:**
  - The likelihood of riders encountering delays is significantly high, affecting their trust and reliance on MTA services.
  - Alerts create noticeable impacts on ridership patterns, as visualized in the comparison of ridership levels "With Alerts" and "Without Alerts."

Key Observations:

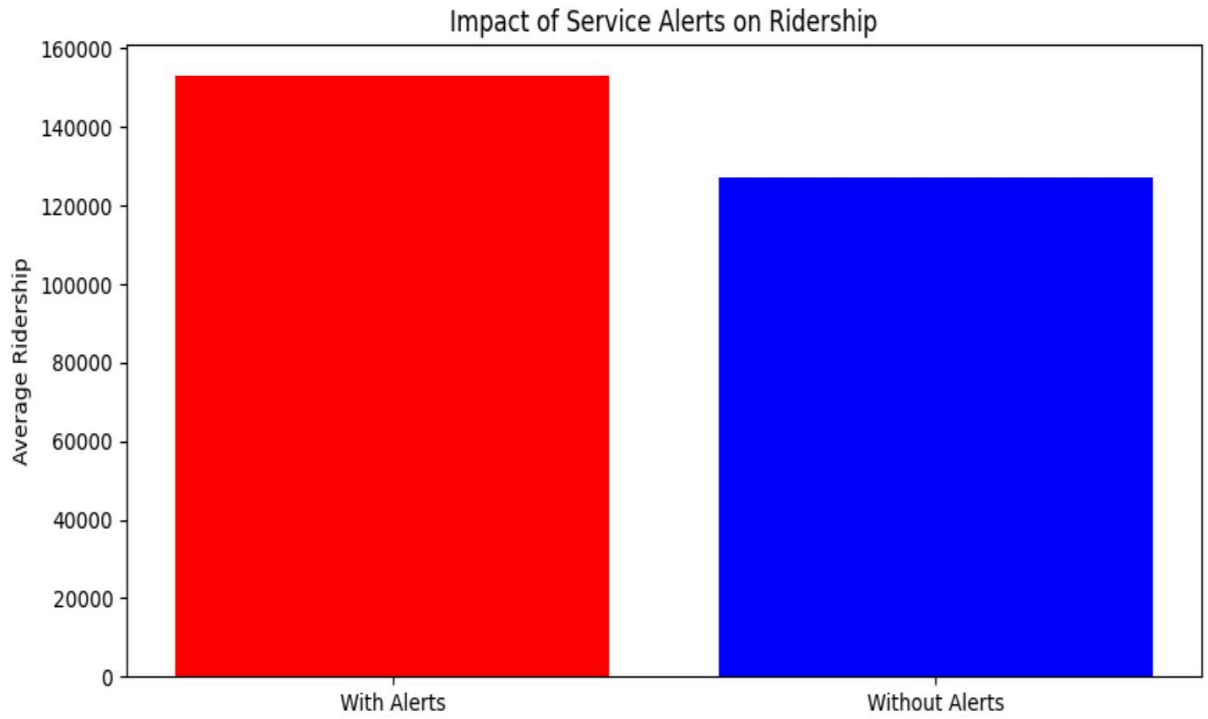
- **Infrastructure Challenges:**
  - The regularity of alerts points to underlying infrastructure or operational inefficiencies that need to be addressed.
- **Ridership Impact:**
  - The presence of alerts correlates with a noticeable reduction in ridership, indicating that service reliability directly influences commuter behavior.

Recommendations:

- **Root Cause Analysis:**
  - Investigate the primary causes of frequent delays and service disruptions, focusing on infrastructure, scheduling inefficiencies, and maintenance practices.
  - Proactive Communication
  - Enhance real-time communication with commuters regarding alerts to reduce dissatisfaction and provide alternative travel options when delays occur.
- **Maintenance Prioritization:**
  - Leverage predictive analytics to prioritize maintenance on routes or systems frequently associated with delays.

Visual Representation:

1. **Status Alerts Table:**
   - Provides a breakdown of the frequency of various service alerts, highlighting delays as the most recurrent issue.
2. **Impact on Ridership Bar Chart:**
   - Shows a clear contrast in average ridership between times with service alerts and times without, emphasizing the need for improved reliability.



ADA Accessibility Analysis

Overview:

Our analysis of ADA (Americans with Disabilities Act) compliance across subway stations highlights a significant gap in accessibility. Only **24.5%** of stations meet ADA standards, leaving **75.5%** of stations without essential features such as elevators, ramps, or other accommodations for individuals with disabilities.

Key Findings:

- **Inclusivity Challenges**: The lack of ADA-compliant stations creates a barrier for individuals with disabilities, affecting their independence and mobility within New York City.
- **Tourism and Diversity Impact**: As a city that prides itself on diversity and being a global tourist destination, limited ADA accessibility reduces potential ridership from tourists and individuals requiring accessible facilities.

Visual Insights:

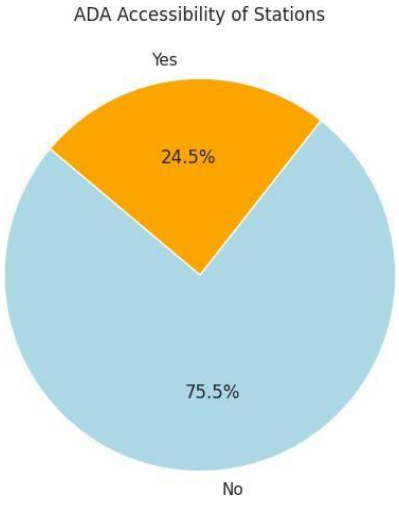
The pie chart provides a clear visualization of the distribution:

- **24.5% ADA Compliant**: These stations provide necessary features to accommodate individuals with disabilities.
- **75.5% Non-Compliant**: These stations lack sufficient accommodations, highlighting the need for targeted improvements.

Recommendations:

1. **Targeted Upgrades**:
    - Prioritize stations with high traffic or central locations for accessibility improvements.Install elevators, ramps, and visual/audio signage at key non-compliant stations.
2. **Government and Community Initiatives**:
    - Collaborate with local government and advocacy groups to secure funding for upgrades.
    - Engage communities to identify priority areas for accessibility enhancements.
3. **Awareness Campaigns**:
    - Educate the public on the importance of accessibility and ongoing efforts to bridge gaps.
    - Highlight upgraded stations in citywide campaigns to encourage ridership.

Conclusion:



Improving ADA compliance is not just about infrastructure; it is about ensuring equitable access for all New Yorkers and visitors, fostering inclusivity, and enhancing the overall usability of the MTA subway system.

Where Riders Are Concentrated

Overview:

Our analysis highlights the most heavily trafficked subway stations in New York City, focusing on peak hours. Stations in Manhattan, such as **Times Square-42nd St**, **Grand Central-42nd St**, and **34th St-Penn Station**, are major hubs for commuter activity. These locations serve as critical focal points for managing crowd control and optimizing service.

Key Findings:

1. **Peak Hour Patterns**:
    - Ridership peaks between **8 AM** and **6–7 PM**, coinciding with typical workday commuting hours.
    - Stations like **Times Square-42nd St** and **Grand Central-42nd St** see the highest concentration of commuters during these times.
2. **Localized Spikes**:
    - Stations such as **Rockefeller Center** and **Chambers St** exhibit notable spikes in ridership, which may reflect localized surges driven by nearby events, offices, or attractions.

Visual Insights:

1. **Ridership Heatmap**:
    - The heatmap illustrates the busiest stations by hour, showing consistent high ridership in Manhattan hubs.
    - **Times Square-42nd St** has the highest ridership during peak hours.
2. **Bar Chart Analysis**:
    - The bar chart ranks the **top 10 busiest stations** during peak hours, emphasizing the dominance of Manhattan stations in ridership volume.

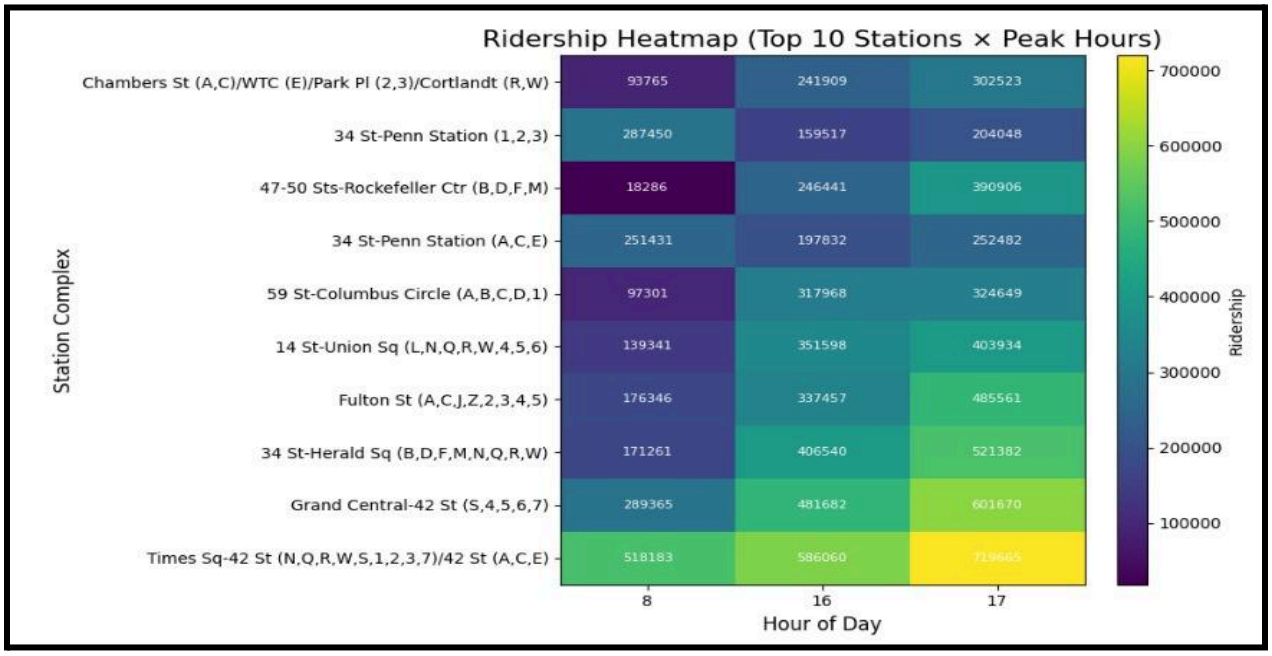
Recommendations:

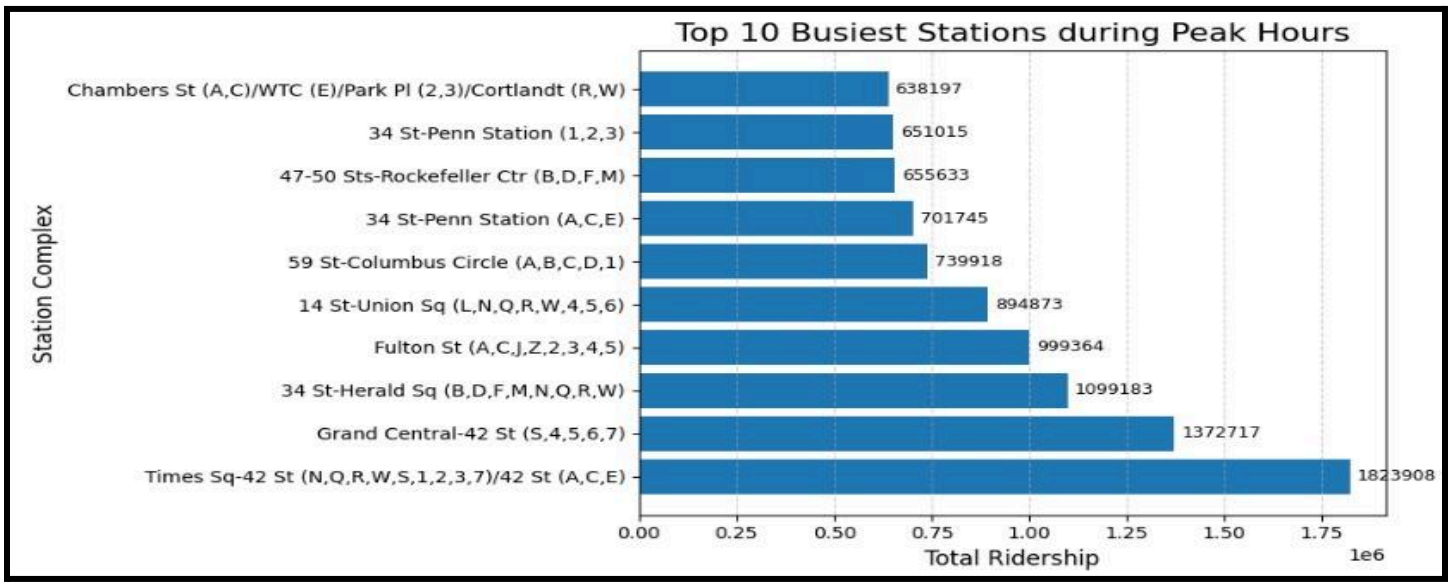
1. **Crowd Management**:
    - Deploy additional staff and implement crowd control measures, such as barriers and designated lanes, during peak hours.
    - Use digital signage to provide real-time updates and redirect commuters to less crowded stations.
2. **Service Optimization**:
    - Increase train frequency during peak hours at key stations to reduce wait times and improve commuter flow.
    - Enhance train scheduling based on hourly ridership patterns derived from the data.

3. **Strategic Resource Allocation**:
    - Allocate more resources to maintenance and cleaning at high-traffic stations like Times Square and Grand Central.
    - Ensure seamless accessibility features to handle high commuter volumes, including elevator maintenance and signage improvements.

Conclusion:

The data underscores the importance of Manhattan stations in the city's transit ecosystem. By leveraging these insights, the MTA can proactively address congestion issues, improve commuter experience, and optimize resource deployment at critical hubs.





<u>Rider Sentiment: Mostly Neutral, Room for Improvement</u>

Overview:

Rider sentiment analysis reveals that the majority of passengers express neutral feelings towards their experiences with the MTA. This presents a significant opportunity to enhance customer satisfaction and shift neutral opinions toward positive sentiment.

Key Findings:

1. Neutral Sentiment Dominates:
    - 71.5% of the feedback falls under neutral sentiment, suggesting a lack of strong positive or negative feelings.
    - Riders are neither dissatisfied nor impressed, indicating a steady but uninspiring service experience.
2. Positive Sentiment:
    - 27.3% of riders express positive feedback, reflecting satisfaction with certain aspects of the service, such as punctuality, cleanliness, or accessibility.
3. Negative Sentiment is Minimal:
    - Only 1.2% of the sentiment is negative, showing that outright dissatisfaction is relatively rare. Negative feedback may focus on delays, service disruptions, or lack of ADA accessibility.

Recommendations:

1. Enhancing Rider Experience:

- Improve areas highlighted in feedback, such as reducing delays, enhancing communication during disruptions, and providing better seating conditions.

Targeting Neutral Sentiment:

- Conduct focus groups or surveys to understand the needs of neutral riders and convert their opinions into positive sentiment.
- Offer incentives like free rides during off-peak hours or loyalty programs to improve the rider experience.

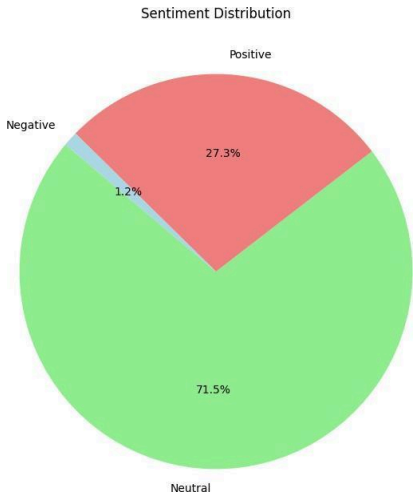
2. Addressing Negative Feedback:
   - Analyze negative feedback to identify recurring issues, such as delays or insufficient station amenities, and implement actionable solutions.

Diagram Insight:

The pie chart showcases the sentiment distribution, with the vast majority (71.5%) being neutral. This visual emphasizes the importance of targeting this group to foster positivity.

Conclusion:

The sentiment analysis highlights that while the MTA is not failing its riders, there is significant room for improvement to deliver a better commuting experience. By addressing feedback, particularly for neutral riders, the MTA can enhance customer loyalty and satisfaction.



Rider Sentiment: Themes in Feedback

Overview:

Sentiment analysis of feedback highlights distinct themes in positive and negative comments. This analysis provides a clearer picture of customer perceptions and areas for improvement.

Key Insights:

1. Positive Feedback Themes:
    - The positive word cloud showcases frequent terms like *Bus*, *Vehicle*, and *General*.
    - These terms suggest that riders appreciate the overall functionality and reliability of the services.
    - Positive mentions of stations, trains, and social interactions emphasize the effectiveness of basic transit operations.
2. Negative Feedback Themes:
    - The negative word cloud highlights terms like *Receptacles*, *Waste*, *Rude*, and *Improper Language*.
    - Negative feedback often pertains to issues of cleanliness, inadequate waste management, and poor customer service.
    - Operational inefficiencies and unprofessional interactions with staff are also recurring themes.

Recommendations:

1. Build on Positive Sentiment:
    - Enhance the rider experience by maintaining high standards in vehicle reliability and station amenities.
    - Celebrate positive feedback through public communication or marketing campaigns to reinforce strengths.
2. Address Negative Sentiment:Focus on waste management and cleanliness, with targeted efforts to improve receptacles at stations and on vehicles.
    - Implement customer service training programs to address issues related to rudeness and unprofessional behavior.
    - Regular audits to ensure operational efficiency and quicker response to complaints can mitigate dissatisfaction.

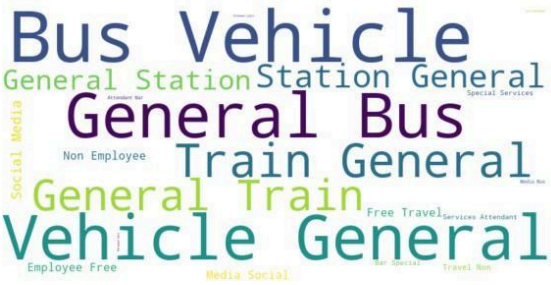
Visual Representation:

- Positive Feedback Word Cloud: Highlights the strong aspects of MTA services that resonate well with users.
- Negative Feedback Word Cloud: Points to specific areas that require attention to enhance customer satisfaction.

Conclusion:

Understanding these themes allows the MTA to sustain its strengths and strategically target weaknesses, ensuring continuous improvement in rider experiences.

Positive Feedback Word Cloud



Negative Feedback Word Cloud

Recommendations for Improvement

1. Operational Efficiency

- Focus Areas:
    - Reduce delays and inefficiencies during peak hours and weekdays, reflecting normal commuting patterns.
    - Pay special attention to the busiest stations such as Times Square-42nd St and Grand Central-42nd St, which see high congestion and demand.
- Proposed Actions:
    - Implement data-driven resource allocation to manage peak traffic efficiently.
    - Optimize train schedules and station staffing during high-traffic periods.
    - Regular audits of station infrastructure and service performance to ensure consistent operation.

2. Cleanliness and Maintenance

- Focus Areas:
    - Invest in robust waste management systems and general station maintenance to improve commuter satisfaction.
    - Address frequent complaints about cleanliness and enforce rules against unauthorized activities like merchandise sales.
- Proposed Actions:
    - Increase the frequency of cleaning services during operational hours.
    - Install more waste receptacles in high-traffic areas and develop clear signage for waste disposal.
    - Launch community awareness programs to encourage commuter responsibility regarding station upkeep.

3. Communication and Alerts

- Focus Areas:
    - Enhance real-time communication regarding delays and disruptions to reduce rider frustration and improve satisfaction.
- Proposed Actions:
    - Leverage mobile apps and digital signage at stations to deliver real-time updates on delays, alternate routes, and expected service restoration times.
    - Use automated SMS or email alerts for registered commuters during significant disruptions.
    - Provide clear and timely updates on both the MTA website and social media platforms to keep riders informed.

Impact of Recommendations:

1. Operational Efficiency: Minimizing delays will directly improve commuter satisfaction, reduce crowding, and increase service reliability.
2. Cleanliness: Enhanced cleanliness will promote a better user experience and foster greater public trust in the transit system.
3. Communication: Proactive and transparent communication will alleviate commuter frustration during disruptions, maintaining rider loyalty and trust.

<u>Feature Analysis Using SHAP Analysis</u>

Overview:

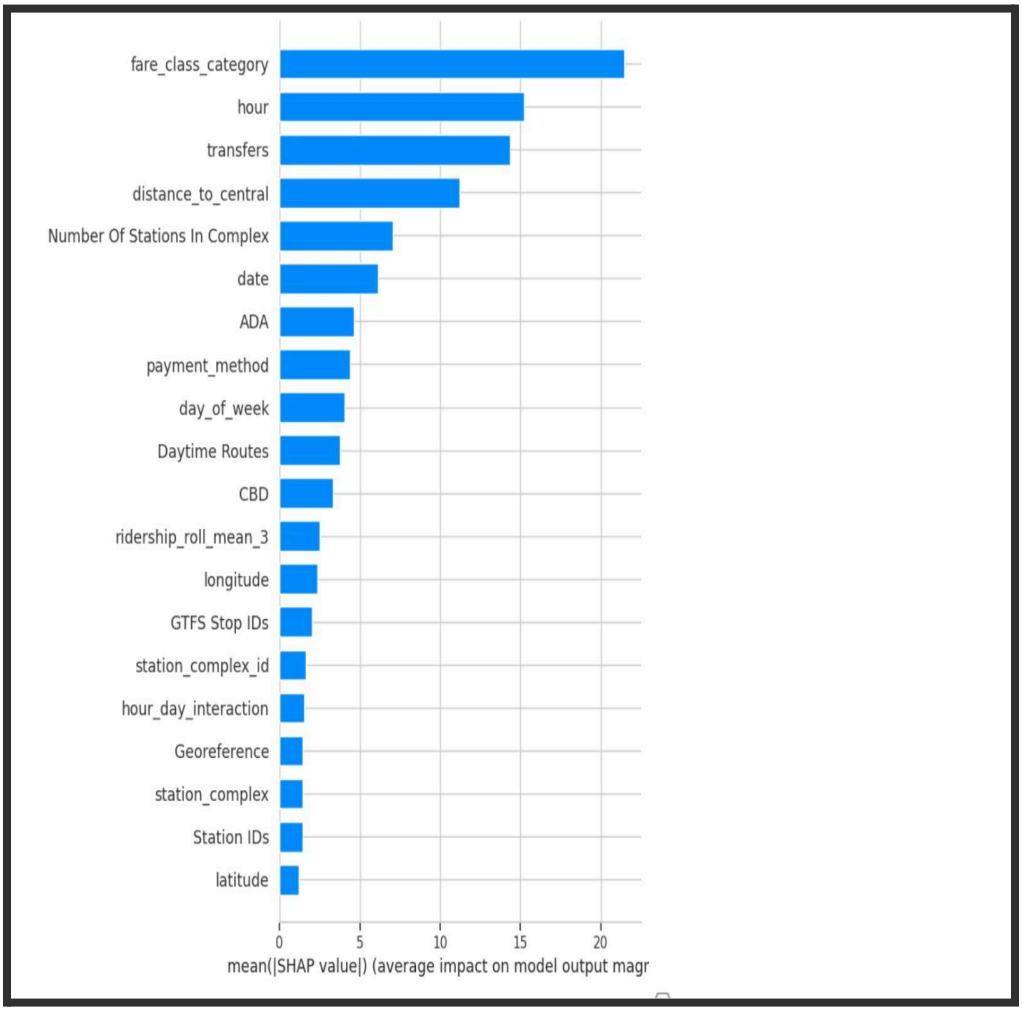
SHAP (SHapley Additive exPlanations) values were utilized to explain the feature importance in the predictive model. The bar chart on the right showcases the mean absolute SHAP values for each feature, ranked in descending order. These values provide insights into the average contribution of each feature to the model's predictions, offering interpretability to a typically "black-box" model like **XGBoost**.

Key Insights:

1. **Hour**:
    - **Impact**: The time of day significantly influences ridership patterns and model predictions.
    - **Reasoning**: Hourly variations align with peak commuting times, such as morning and evening rush hours, driving substantial ridership changes.
2. **Transfers**:
    - **Impact**: The number of transfers a commuter makes has a notable contribution to the model.

- **Reasoning**: More transfers often indicate longer, more complex journeys, which may correlate with higher ridership.

3. **Distance to Central Stations**:
   - **Impact**: Proximity to key hubs such as Times Square or Grand Central strongly affects ridership.
   - **Reasoning**: Stations closer to central business districts experience more significant commuter traffic.
4. Alert Date and ADA Compliance:
   - **Impact**: Alerts about disruptions and ADA (Americans with Disabilities Act) compliance moderately influence predictions.
   - **Reasoning**: Service alerts deter some commuters, and ADA accessibility impacts the inclusiveness of stations.
5. **Minimal Impact Features**:
   - Features like **Stop Name**, **Station IDs**, and **Geofence** showed negligible contributions to the model's output, indicating their limited relevance in predicting ridership trends.



Importance of SHAP:

- **Transparency**: SHAP values enhance the interpretability of XGBoost by explaining how each feature contributes to predictions.
- **Actionable Insights**: By identifying key drivers like **hour** and **transfers**, transit authorities can better understand ridership behaviors and optimize services accordingly.

Applications:

- Use feature insights to refine service schedules, focusing on peak hours.
- Implement targeted improvements in highly impactful areas, such as central stations and transfer efficiency.
- Leverage SHAP analysis to validate and refine predictive models for better decision-making.

XGBoost Ridership Prediction

Overview:

The project employed the XGBoost machine learning model to forecast daily ridership for the MTA subway system. By leveraging advanced predictive capabilities, the model demonstrated a high level of accuracy, reflected by an R^2 score of 0.7727, showcasing its robustness in capturing complex ridership patterns.

Key Features:

1. Big Data Integration:
    - Integrated data sources included:
        - Ridership statistics: Capturing historical trends.
        - Service alerts: Accounting for disruptions and delays.
        - Customer feedback: Highlighting sentiment and pain points.
        - Operational metrics: Evaluating system-level efficiency.
    - Unified multiple datasets to provide a comprehensive understanding of influencing factors.
2. Model Design:
    - Objective: To uncover complex patterns and provide actionable forecasts for ridership.
    - Scalable Processing: Utilized efficient large-scale data handling and preprocessing techniques to train the model on extensive datasets.
    - Feature Engineering: Derived critical features such as transfer counts, proximity to central stations, and time-of-day effects to enhance predictive power.
3. Forecasting Benefits:

- Optimized train schedules to align with predicted ridership demand.
- Facilitated strategic resource allocation for busy stations and times.
- Enhanced service reliability by enabling data-driven decisions for future planning.

Performance Metrics:

- R^2 Score: 0.7727
  - Indicates a strong fit between the predicted and actual ridership values.
  - Validates the model's capability to generalize well on unseen data.

Applications:

- Operational Planning: Forecasted ridership data can guide adjustments to train frequency and staff deployment during peak times.
- Policy Development: Insights derived from predictions can inform infrastructure investments and service improvement policies.
- Customer Experience: Anticipating ridership surges allows for preemptive measures to reduce congestion and enhance commuter satisfaction.

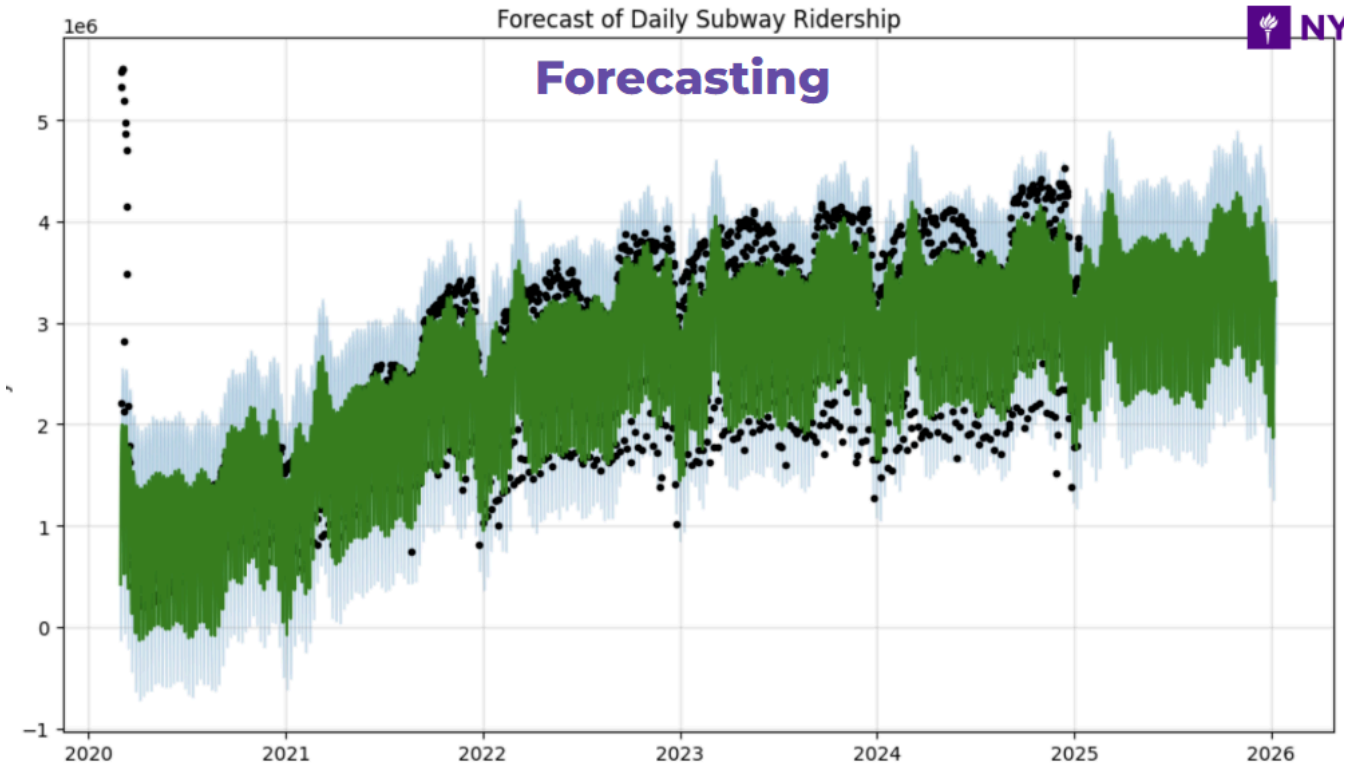
```
from sklearn.metrics import r2_score

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

```
XGBoost R²: 0.7727
```

Forecast of Daily Subway Ridership:

This shows a 1 year Forecast using prophet.



Distribution Analysis of Residuals

Overview:

Residual analysis is a critical component in evaluating the performance of predictive models. For the XGBoost ridership prediction model, residual analysis helped confirm that the model is well-fitted and unbiased.

Key Insights:

1. **Normal Distribution of Residuals**:
    - The residuals exhibit a **normal distribution**, centered around zero.
    - This indicates that the **errors are unbiased** and randomly distributed.
    - Suggests that the model is capturing most of the variation in the data accurately without systematic error.
2. **Lack of Skewness or Outliers**:
    - The absence of major **skewness** confirms that the model predictions are balanced across the range of observed data.
    - No significant **outliers** are visible, ensuring the model's stability and robustness across varying scenarios.

Visual Analysis:

- **Distribution of Residuals**: A histogram with a smooth curve overlays shows the residuals' normality.
- **Residuals vs. Predictions**: Scatter plots highlight that residuals are evenly distributed, further validating the lack of bias in the model.

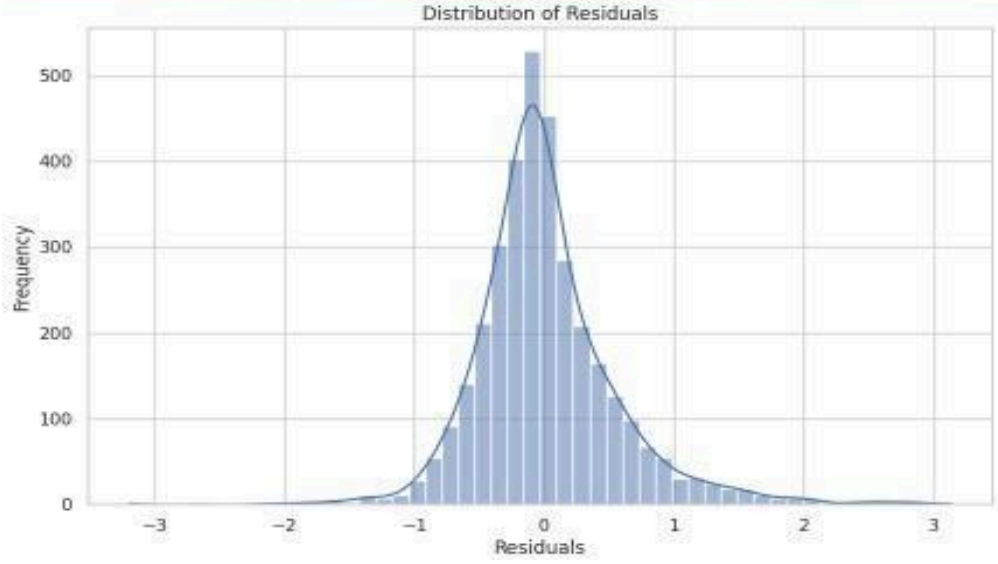
```
import matplotlib.pyplot as plt
import seaborn as sns

# 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.
```



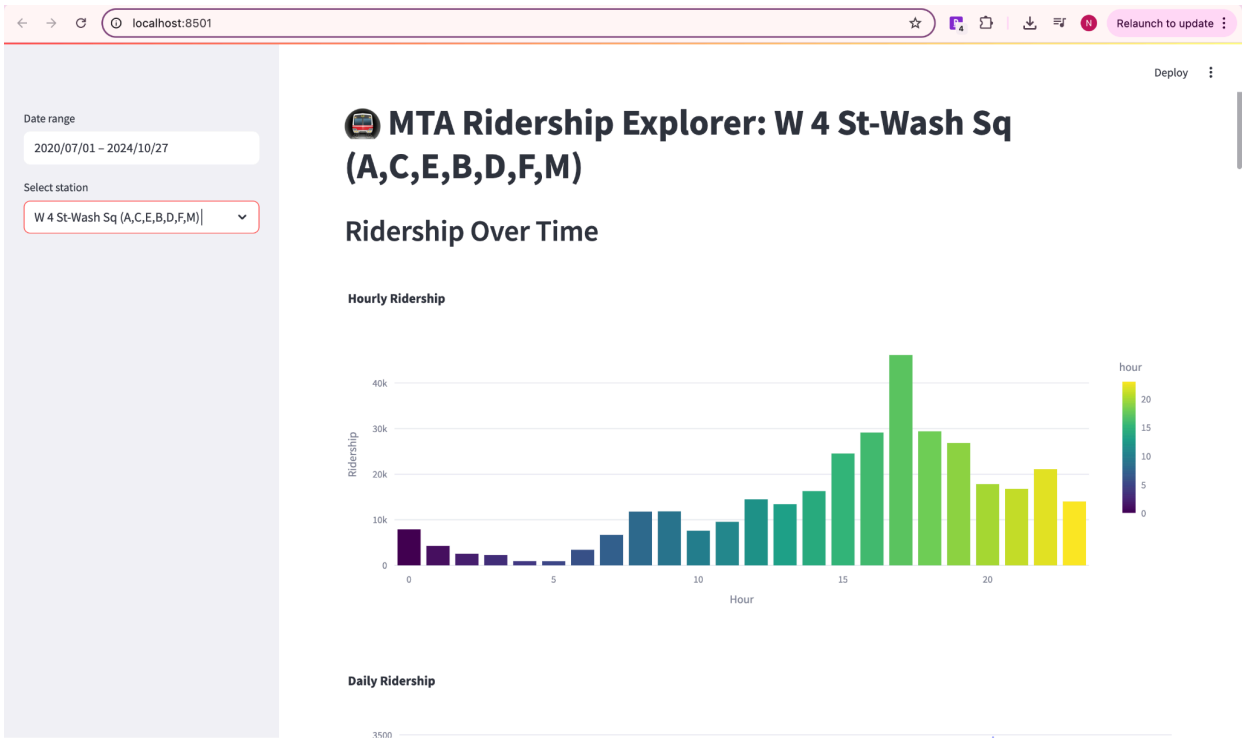
Conclusion:

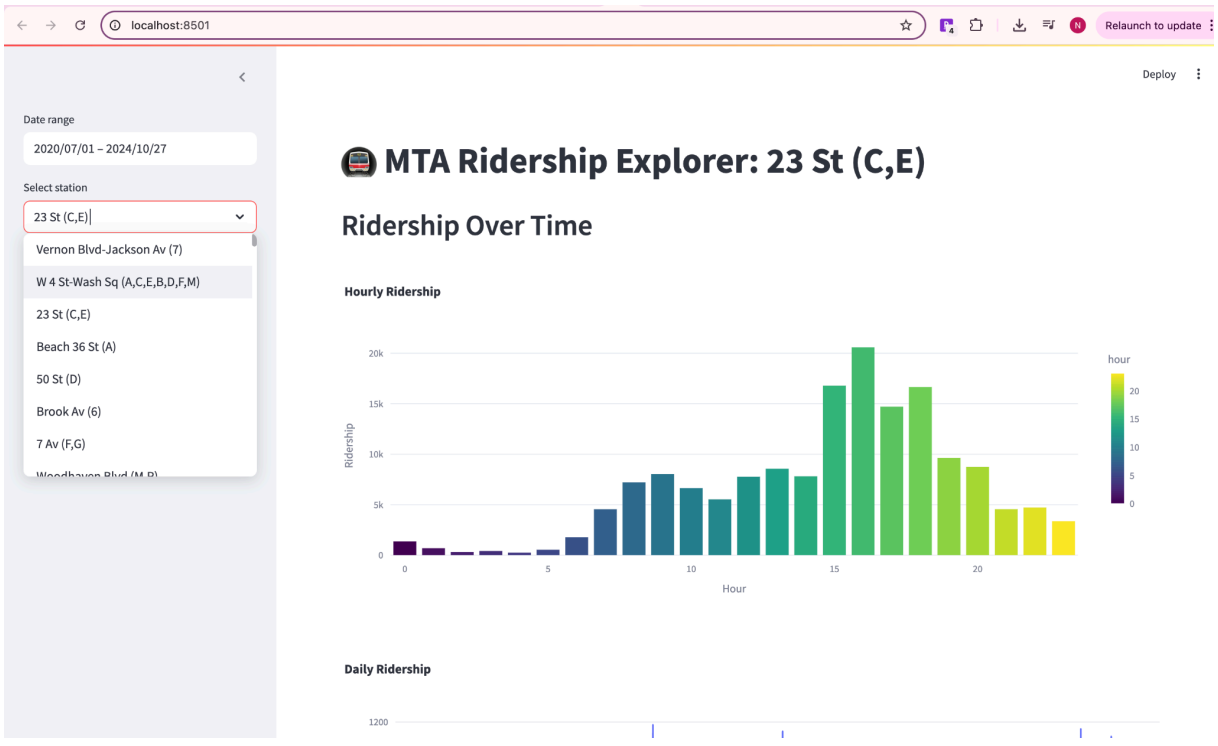
The normal distribution of residuals and the absence of skewness or outliers reaffirm the reliability and generalization capability of the XGBoost ridership prediction model. This analysis validates the model's effectiveness in predicting daily ridership patterns accurately.

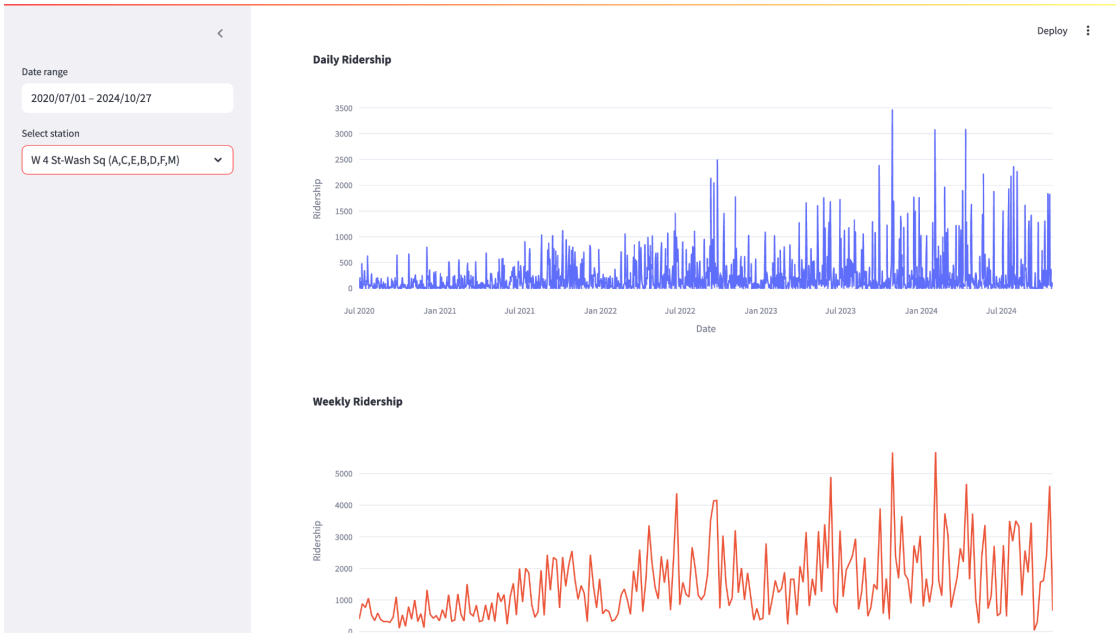
MTA Ridership Explorer Dashboard

Overview: A minimal Streamlit dashboard for exploring NYC MTA station ridership data. Key features include:

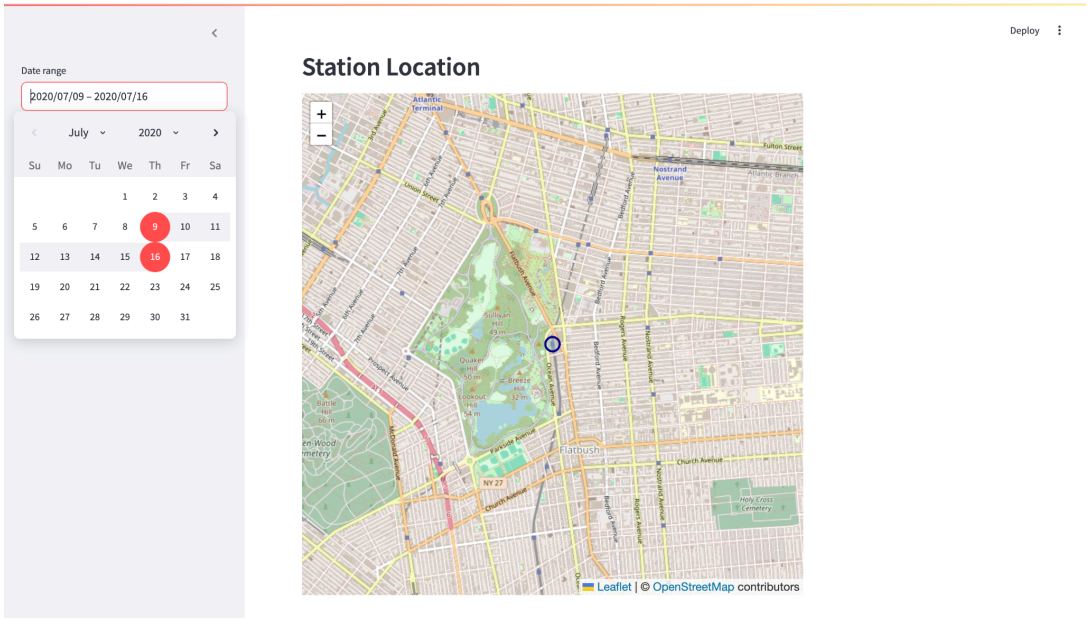
- **Time series views:** hourly, daily and weekly ridership line charts



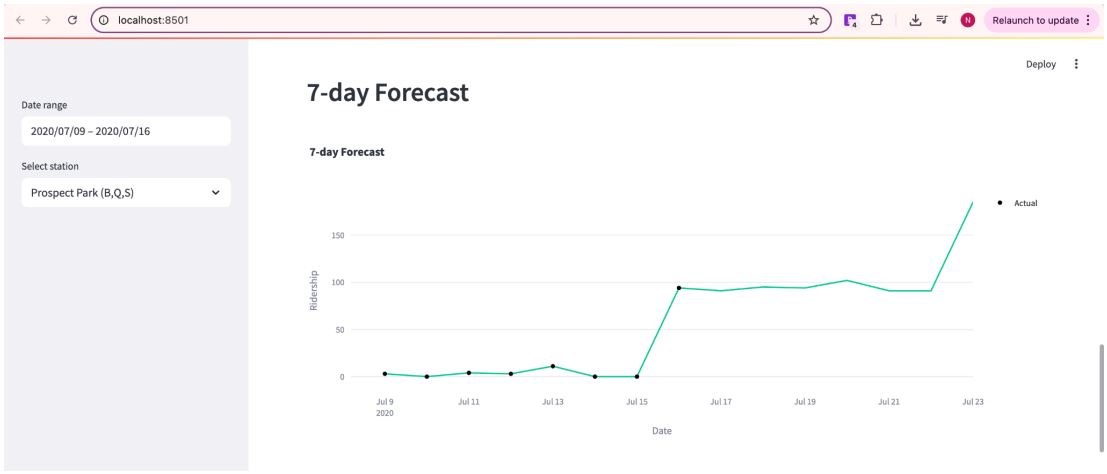




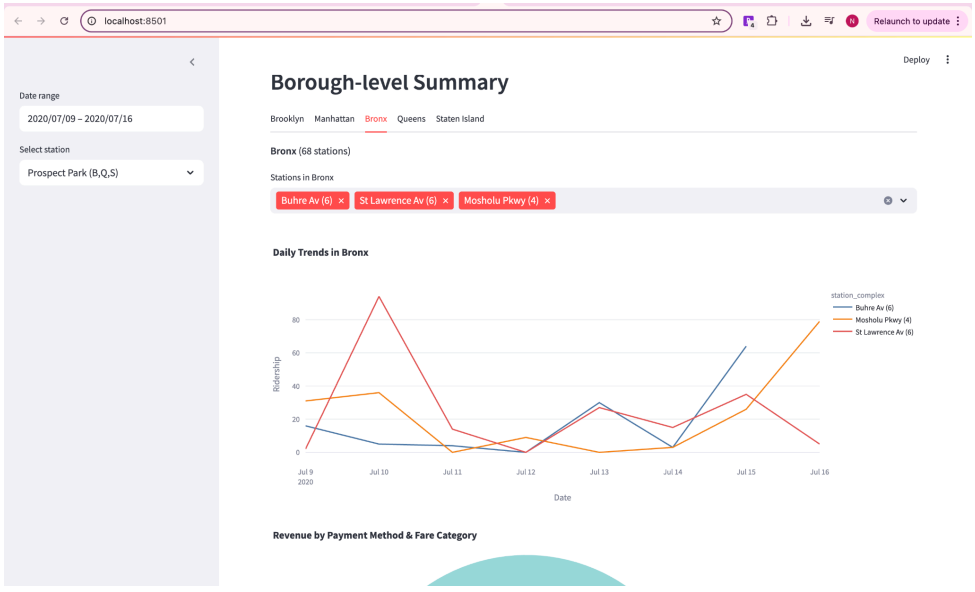
- **Interactive map:** visualize station location in Folium



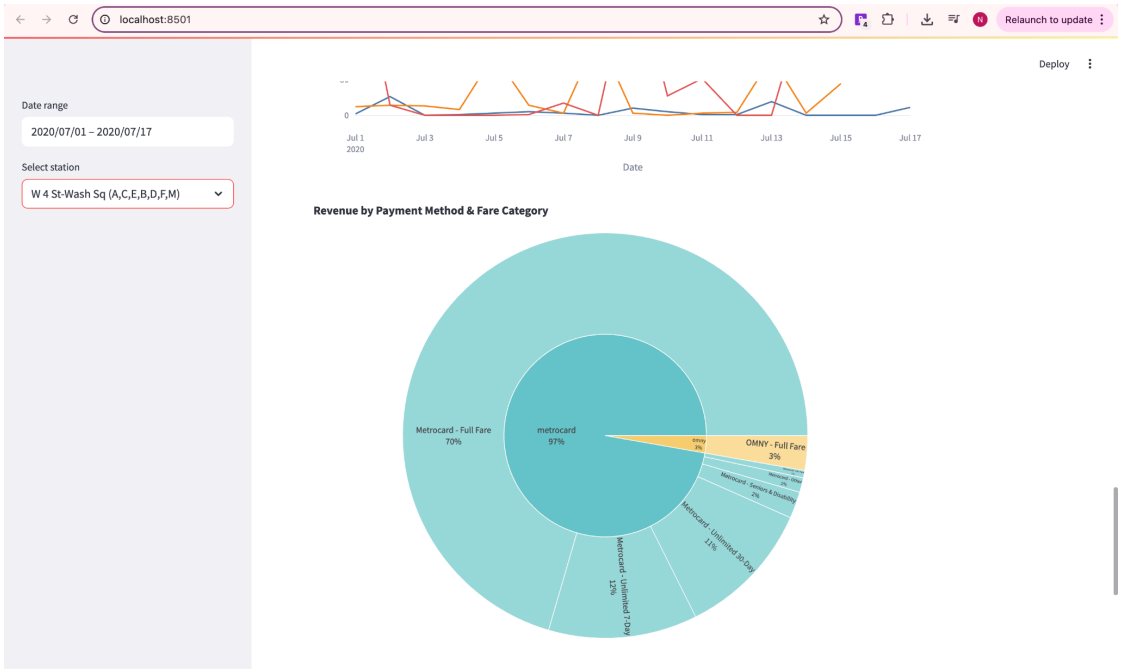
- **Short-term forecasts:** 7-day Prophet model with uncertainty intervals



- **Borough comparisons:** multi-select stations, compare trends side by side



- **Revenue breakdown:** sunburst chart of payment method vs. fare category



In just a few clicks, this dashboard transforms raw MTA ridership logs into actionable insights , from pinpointing when and where subway usage peaks, to forecasting short-term demand, to uncovering which payment options drive the most revenue. By blending interactive time-series plots, geospatial maps, and categorical breakdowns, it empowers transit planners and data teams to:

- Monitor real-world performance and quickly spot anomalies
- Compare station trends at both hourly and seasonal scale
- Anticipate demand and allocate resources more efficiently
- Evaluate fare strategies and revenue impacts across passenger segments

Whether the user is investigating service disruptions, planning maintenance windows, or optimizing fare promotions, this tool provides a flexible, data-driven lens on NYC's vast transit network—and lays the groundwork for even richer analyses (e.g., weather integration, real-time alerts, or predictive maintenance).