Predicting the Impact of Air Quality on Public Transportation Usage in Metropolitan Areas

SIADS 693: Milestone II

M. Cott, M. Giarmarco, M. Griffin

June 26, 2025

Introduction

Anthropogenic climate change, its potential impacts, and strategies for its mitigation dominate public policy discussions in the twenty-first century.[1] While there is copious research and literature on the role public transportation can play in reducing emissions and improving air quality,[2] there is much less research on the impact air quality has on the usage and adoption of public transportation. Our project looks to explore the relationship between air quality and public transportation usage in Chicago (CHI) and New York City (NYC), and specifically to understand the explanatory power of air quality metrics on public transportation usage prediction.

In addressing this problem, we sought to identify either a 'virtuous cycle' of improved air quality and public transportation adoption (thus improving air quality), or a 'downward spiral' of degraded air quality, and reduced adoption (thus exacerbating greenhouse gases). This would have substantial implications for public policy and could provide a clear signal to transportation stakeholders that interventions in this domain have reinforcing effects over time.

Our primary finding from the unsupervised learning workflow was that high level meta clusters emerged, corresponding to specific combinations of weather and air quality conditions. While we did not end up using these in the supervised learning workflow, they have high contextual relevance and are additional avenues for exploration. As for our supervised learning workflow, the primary finding was that we could not capture significant relationships between air quality and daily ridership using our methodology. While we were able to predict daily ridership per city/mode combination moderately effectively (especially for NYC), this was mostly explained by non-environmental factors and associated engineered features (proximity to holidays/weekends).

For this project we primarily explored the following models:

- Feature Engineering
    - Truncated SVD
- Unsupervised Learning
    - DBSCAN
    - Agglomerative Clustering
- Supervised Learning
    - Lasso Regression
    - Random Forest Regression
    - XGBoost Regression

Related Work

Air Pollution Hindering a Transit-Oriented City: Examining the Association of Particulate Matter Concentration with Public Transit Ridership and Road Traffic in Seoul, South Korea[3]

This study applied seemingly unrelated regression (SUR) models to assess PM concentration effects for both transit ridership and road traffic in Seoul from 2015 to 2018. The authors found that elevated pollution levels led to overall reductions in travel, with subway ridership dropping more than road traffic volumes. They conclude that poor air quality suppresses total mobility rather than triggering a shift from transit to private vehicles.

Air Pollution and Seasonality Effects on Mode Choice in China[4]

This study examined whether improved air quality increased the likelihood of choosing non motorized transport over motorized transport, using seasonal “revealed preference” data from Taiyuan, China. The authors used discrete mode choice models across summer and winter, and showed that individuals were more likely to bike or walk when air quality improved.

Our project differs from the above two studies in terms of locales and years studied, models used, air quality metrics included, and modes of transportation highlighted (no private or non motorized transportation modes considered).

Particulate Matter Concentration and Composition in the New York City Subway System[5]

This study collected measurements of PM2.5 levels across the NYC subway network, revealing significantly elevated concentrations in underground environments (and subway cars). It highlights the unique composition of subway air pollution, while not tackling impact on usage directly.

Our project differs from this study in terms of environments studied (indoor, underground vs outdoor, above ground) and focus of the overall analysis. While both efforts look at NYC and PM2.5 levels, this is where the similarities end.

Relationship to Milestone I

This project is a synthesis of topics explored by our team in Milestone I. Mark Griffin looked at air and weather conditions and how they affected baseball performance. This involved using data from the OpenMeteo APIs, but considered different timeframes and more locales. Matt Cott and Mickey Giarmarco investigated NYC public transportation ridership data during the COVID-19 pandemic. MTA datasets were used, but for a different time frame. Crucially, none of the EDA in this project was carried over from Milestone I. In order to reduce bias or code replication, Mark performed data acquisition for public transportation, and Mickey performed data acquisition for AQI and weather.

Data Sources

Air Quality & Weather

This project used two environmental datasets retrieved from the Open-Meteo API service, specifically targeting historical weather and air quality data. The datasets were accessed via two API endpoints (see Appendix for links). Both APIs are free, publicly available resources that return data in JSON format. The returned data was parsed and converted into structured pandas DataFrames for further cleaning and analysis.

The weather dataset included hourly observations that were aggregated to a daily level. Key variables included rainfall and snowfall totals and time boxed maximums, relative humidity, apparent temperature, wind speed, and wind direction. Similarly, the air quality dataset was composed of hourly readings for various pollutants and overall air quality metrics. These included U.S. AQI overall and for specific pollutants such as PM2.5, PM10, ozone, carbon monoxide, sulfur dioxide, and nitrogen dioxide.

Approximately 900 daily records were retrieved for each dataset per location, covering the period from 12/31/2022, to 6/1/2025. This was then reduced to the time period in question, 1/1/2023 - 12/31/2024.

Public Transportation

Public transportation data for this project was obtained from the respective cities' (Chicago and New York) Open Data Portals (see Appendix for links). The original data consisted of hourly ridership, available per city, per station/stop, and per mode (bus/train and bus/subway). This was gathered for 1/1/2023 - 12/31/2024. These web portals were used to generate CSVs of hourly data. These CSVs were then processed in Python to aggregate hourly ridership into daily ridership and originally organized by stop/station. This original combined transportation dataset consisted of just over 900,000 rows.

Final Dataset

These 2 data sources were combined to create the 'final' dataset with each row containing daily ridership and weather/air quality features. This 'final' dataset was used throughout Feature Engineering and Unsupervised Learning, but a reorganization and reduction of the dataset was required for Supervised Learning. That process and the factors involved will be elucidated in the Supervised Learning Discussion section.

Feature Engineering

Data Preprocessing and Cleaning Overview

We undertook a detailed preprocessing pipeline to prepare weather and air quality data for NYC and CHI. This process involved cleaning and dimensionality reduction steps, followed by strategic feature selection informed by redundancy and variance analysis.

Raw weather and air quality data were loaded from the Open-Meteo API. After loading, we found that the only missing values were from metrics that had not been recorded until the previous year (2024),

so we removed them from the dataset. The data from the API was provided at an hourly frequency, which we converted to daily frequency using resampling and aggregation techniques.

To better capture the non-linear effects of air quality on transit behavior, we binned the continuous AQI values into discrete severity categories using standard EPA defined thresholds. This binning transformed raw AQI measurements into categorical levels of pollution exposure. This subsequently allowed the model to more easily detect threshold effects, such as behavioral changes when air quality shifts from ‘moderate’ to ‘unhealthy’.

Redundancy Testing Across Locations

The Open Meteo API uses latitude-longitude combinations to determine which air quality/weather station sensors to use. To evaluate how many locations we needed to pull data from per city, we tested the presence of multiple latitude-longitude combinations within each urban area. This involved assessing whether each location provided unique information or if there was significant redundancy. We calculated a spatial ratio to determine whether the variance across locations was greater than the variance across days. The results showed high similarity between most locations, particularly in urban cores, indicating that including multiple sensors would introduce redundant signals into the model.

However, a notable exception was found in New York. The Hamptons and surrounding areas consistently exhibited distinct patterns, likely due to their geographic distance and socioeconomic difference from the city center. Although this outlier was documented, we ultimately chose to represent each city using a single, centrally located coordinate that was highly correlated with other sites. This approach reduced data dimensionality while preserving the key signal needed for predictive modeling.

Variance Testing & Feature Selection

We conducted statistical variance analysis on the cleaned dataset to identify which features demonstrated meaningful variation over time. Features with very low variance and those that remained effectively constant were flagged and removed. This step was critical to reducing noise, avoiding overfitting, and streamlining model training. The weather metrics included were the ones that changed meaningfully day to day and air quality measures that responded to environmental and seasonal conditions.

Dummy Variables & Feature Engineering

To incorporate categorical variables such as transit mode and unit ID, we applied one-hot encoding. However, since unit ID had high cardinality, we used Truncated Singular Value Decomposition to reduce the dimensionality of the sparse matrix. We selected 20 components after reviewing the explained variance ratio and model performance tradeoffs, attempting to balance compression with signal retention. While we believed this approach would yield results, the resulting components were not legible or able to be analyzed effectively. Therefore, after initially including them in our unsupervised learning models, they were dropped for supervised learning.

In addition to dimensionality reduction, we engineered lag features to capture temporal dependencies in the data. Specifically, we created lagged versions of min-max temperature and our binned AQI to allow the model to account for delayed effects on ridership. These lag features were selected based on domain knowledge and exploratory analysis, which suggested that certain

environmental impacts, such as extreme temperature or high particulate matter, might manifest with a delay.

To account for behavioral patterns in transit ridership, we introduced binary indicators for weekends and holidays. The weekend feature flags Saturdays and Sundays, recognizing that transit demand typically follows a different pattern on non-workdays.

We also created a holiday indicator based on the U.S. federal holiday calendar. To capture potential ridership effects in the days surrounding major holidays, such as reduced travel before or after events like Thanksgiving or Independence Day, we extended this indicator to include a 1 day window around each holiday. This "holiday window" feature helps the model recognize transitional travel behavior that often occurs outside the holiday itself, improving sensitivity to real-world ridership patterns.

Unsupervised Learning

Methods

Our unsupervised learning workflow was designed to explore latent structure in the daily transit ridership dataset, enriched with air quality and weather metrics. Our goal was to generate cluster-derived features that could later inform the supervised model predicting ridership behavior. Ultimately, we applied two different unsupervised methods, DBSCAN (Density-Based Spatial Clustering of Applications with Noise) on the feature engineered data, and Agglomerative Clustering on the DBSCAN derived centroids.

These methods were selected for their complementary strengths, with DBSCAN identifying fine-grained, density-based clusters in noisy data, and agglomerative clustering abstracting these clusters into broader, interpretable groups. This combination of methods enabled multi-scale pattern discovery suitable for downstream modeling.

While we initially explored the K-means model, it quickly proved inadequate in uncovering broad, related clusters. We had no reason to believe the data would cluster in equally sized, spherical groupings, but K-Means modelling was a quick and computationally lightweight way to explore the data.

Evaluation

We explored the following two representations:

- **Environmental Feature Set:** AQI, Weather
- **Truncated SVD Feature Set:** Stop-level indicators to reduce dimensionality (from > 1000 locations to 20 components)

These representations allowed for capturing of both global (air/weather) and local (stop-level) patterns in a manageable form. As our goal was to uncover clusters that represented human legible relationships in the data, we did not scale or transform the AQI/weather data before clustering.

We used two standard clustering metrics:

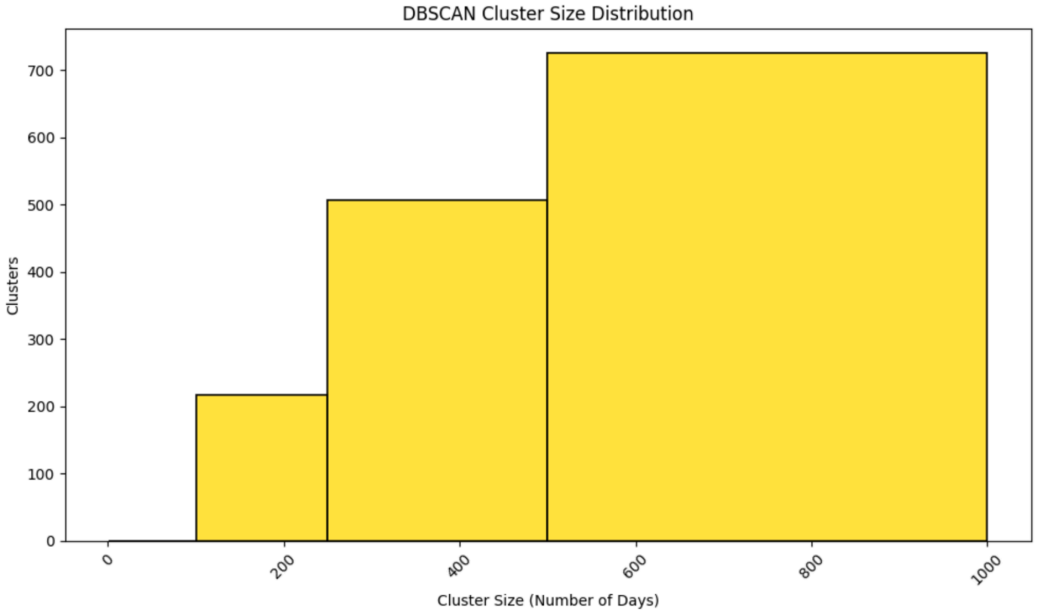
- **Silhouette Score:** Evaluates cohesion vs separation (-1 to +1)
- **Davies–Bouldin Index:** Measures cluster similarity (lower score is better)

DBSCAN

Our first unsupervised method was DBSCAN, chosen for its ability to identify clusters of arbitrary shape and to robustly handle noise in complex, real-world datasets. Given the irregular and heterogeneous nature of transit patterns and environmental variability across New York City and Chicago, DBSCAN offered a way to detect meaningful groupings without assuming spherical or evenly sized clusters.

We explored a range of parameters to tune DBSCAN's performance. The `eps` parameter, which controls neighborhood size, was tested across values of 0.5, 5.0, and 10, while `min_samples`, determining the minimum number of points to form a cluster, was varied between 3, 5, and 10. A grid search strategy was used to evaluate combinations of these parameters, using Silhouette Score as our evaluation criterion. This metric provided a proxy for intra-cluster cohesion and inter-cluster separation, key factors in assessing the strength of clustering in high-dimensional, noisy settings.

DBSCAN's application on both the prototype and full data sets yielded over 1,400 clusters, with most clusters spanning hundreds of days and the majority falling into the 500–1000 day range. Visualized in the histogram below, the DBSCAN output was dominated by large, heterogeneous groupings that lacked coherent description. While the algorithm captured spatial and temporal patterns, the resulting clusters lacked the interpretability needed for supervised modelling and cohesive data analysis. This motivated us to undertake a second layer of abstraction through meta-clustering, enabling us to distill DBSCAN's outputs into a smaller number of meaningful, categorical conditions.

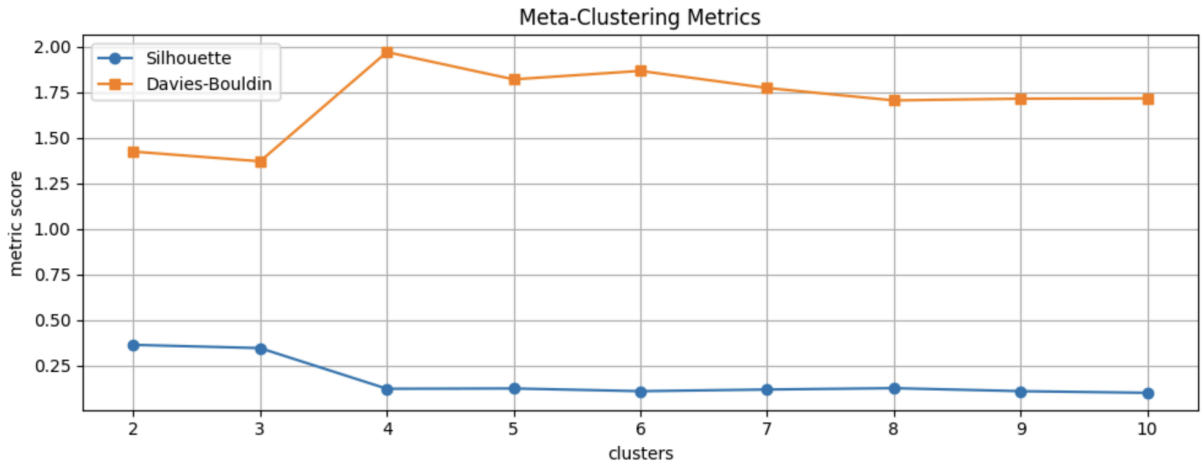


Agglomerative Clustering (Meta-Clustering)

To transform the granular DBSCAN output into a format suitable for supervised modeling, we applied Agglomerative Clustering as a secondary unsupervised method. This hierarchical clustering approach was used to group the DBSCAN-derived clusters into broader "meta-clusters," yielding a more interpretable and compact representation of environmental conditions. Furthermore, agglomerative clustering allows us to trace how clusters merge at different levels of granularity, supporting a cluster count selection that aligns with the above stated goals.

The input to the agglomerative model consisted of cluster-level centroids: each DBSCAN cluster was summarized by the mean of standardized environmental features, expressed as z-scores. This transformation allowed us to retain key contextual information while reducing dimensionality and noise. We explored different values for the `n_clusters` parameter, ranging from 2 to 10. To guide selection, we looked at both Silhouette Score (where higher values indicate better-defined clusters) and Davies–Bouldin Index (where lower values suggest more compact and well-separated clusters). This analysis revealed a clear plateau beyond 3 clusters, with diminishing returns in both score improvements and interpretability.

This 3 cluster solution represented an optimal balance between quantitative validity and interpretability. While the silhouette scores were modest (~0.35), this level of structure is acceptable in complex, real-world environmental data where overlapping patterns are expected. The summary table below provided insight into each meta-cluster's distinct environmental profile, highlighting contrasts in AQI levels, temperature, humidity, and wind patterns.



Meta-Cluster Naming and Interpretation

To interpret and name the meta-clusters, we examined the z-scored feature summaries, applying standardized thresholds to flag notable deviations from the mean. Features exceeding ±1.0 were considered strongly above or below average, while those in the ±0.5 to 1.0 range were treated as moderate signals. Related features (such as PM2.5, ozone, and NO_2) for air quality and (temperature and wind speed) weather were assessed in groups to form a legible narrative. This process yielded three interpretable categories:

- **Average Conditions:** No significant deviations across air quality or weather indicators
- **Heavy Air Pollution with Stagnant Air:** Broad elevation across pollutants, combined with low wind speed
- **Heavy Ozone Pollution with Heat Exposure:** Elevated ozone and temperature, with moderate particulate pollution

These names offer human interpretable summaries that reflect real-world conditions and support supervised model contextualization.

meta_cluster	cluster_name	AQI Max	PM2.5 Max	PM10 Max	Ozone Max	CO Max	SO_2 Max	NO_2 Max	Lagged AQI Max	Temp Max	Humidity Mean	Wind Speed Mean
0	Average Conditions	-0.35	-0.27	-0.28	-0.29	-0.20	-0.15	-0.10	-0.27	-0.18	-0.06	0.13
1	Heavy Air Pollution with Stagnant Air	1.01	1.79	1.82	0.13	1.96	1.53	1.19	0.49	0.25	0.13	-0.75
2	Heavy Ozone Pollution with Heat Exposure	2.29	0.90	0.97	2.53	0.12	0.04	-0.14	2.02	1.47	0.42	-0.49

A Note on Meta-Clusters Silhouette Score

The silhouette score for the meta-clusters was approximately (~0.35), which falls in the lower end of the "weak to moderate" interpretability range. While this might raise concerns in a tightly controlled or synthetic dataset, it's acceptable in this context given the complexity of our real-world environmental data. Weather and air quality metrics are continuous, noisy, and often correlated, making clustering inherently messy. Importantly, these clusters are not treated as ‘ground truth’ categories, but rather as abstracted environmental profiles to support downstream modeling.

Since these clusters were derived from DBSCAN centroids, which already represent aggregated local density structures, some imprecision in boundary definition is expected. The goal is not to produce sharp class labels, but to reveal recurring combinations of conditions that may influence transit behavior. That said, we avoided overstating these findings by proposing to use cluster types as categorical features in the supervised learning workflow, ensuring they did not overwhelm other predictive variables.

Evaluation Metric Table

Method	Clusters	Silhouette Score	Davies–Bouldin	Notes
DBSCAN	1453	0.75	N/A	Over-clusters data, useful for granularity
Agglomerative	3	0.35	1.425	Human interpretable
K-means	2	0.15	N/A	Limited structure captured

Discussion

During this unsupervised learning workflow, we learned the importance of cluster interpretability, specifically when it comes to a complex dataset such as ours. Many groupings were available and justified by raw metrics. However, without incorporating domain knowledge these clusters had an outsized impact when applied to the supervised learning workflow. When supervised learning was initially conducted with DBSCAN clusters, they dominated the feature impact evaluation. We were unable to parse this in an intelligent way, and this led us to go back and attempt the meta clustering described above. We ultimately did not use any clusters as features in our supervised learning workflow.

While these meta clusters were not included in supervised analysis, with additional time and more rigorous feature ablation in the unsupervised workflow, these have the potential to offer categorical insight when analyzing data subsets. Subset ridership prediction could be done not just by city and mode, but also by various clusters and categories derived from unsupervised learning.

Ethical Considerations

Ultimately, we did not encounter any ethical issues in our unsupervised learning workflow. However, this could have arisen if the clusters generated were derived using untransformed stop IDs. Analyzed geospatially, these clusters could have had strong relationships with neighborhoods exhibiting distinct socioeconomic characteristics. If ridership patterns arose only for affluent or poverty stricken areas, communicating those results with nuance and empathy would have been required.

Supervised Learning

Methods

Random Forest Regressor

Random Forest Regressor was selected for its robustness to multicollinearity, capacity to model non-linear relationships, and built-in feature importance estimation. To optimize model performance, we applied a grid search to tune hyperparameters. The grid search was conducted over a range of values for key parameters: `n_estimators`(number of trees), `max_depth` (tree depth), `min_samples_split`, and `min_samples_leaf`. Trying both shallow and deep trees helped to control overfitting, while varying `min_samples_split` and `min_samples_leaf` ensured the model doesn't overly rely on small sample splits, which can lead to high variance.

Lasso Regression

Lasso regression was chosen primarily for its simultaneous capability to predict outcomes and perform feature selection. This was particularly valuable given our dataset's complexity (more than 60 potential predictors), containing highly collinear and contextual features. Lasso's L1 regularization penalizes overly complex models, shrinking irrelevant coefficients toward zero, and thereby isolating the most influential weather and air quality features. This balance of interpretability and regularization made Lasso a natural fit for our analysis, allowing us to model transit ridership effectively without overfitting.

We conducted hyperparameter tuning using GridSearchCV, specifically targeting the regularization parameter `alpha`. Our search grid included 50 values logarithmically spaced between 0.0001 and 10. This broad range allowed exploration of various levels of regularization. Smaller alpha values permitted the model to retain more predictors, useful for subtle signals, while larger values enforced stronger shrinkage, which we believed could be beneficial if air quality features proved minimally predictive.

XGBoost Regression

To maximize predictive accuracy, gradient boosted trees improved upon Random Forest regression's performance while worsening interpretability. With intensive feature analysis and dimension reduction already completed, our focus was solely prediction using our mixture of selected features. In similar fashion to our process with random forest regression, a grid search was structured to test key

parameters of tree structure including: `n_esimators` (number of trees), `max_depth` (tree depth), `learning_rate` (rate to fix errors from previous iteration) and `min_child_weight` (minimum number of samples needed for a new tree).

Evaluation

Metrics & Validation

Each of our models was measured using the following:

- **R^2:** to quantify the proportion of variance explained by the model
- **RMSE:** to penalize large prediction errors on the log-transformed scale
- **MAE:** to complement RMSE, MAE reflects the average magnitude of error. It is less sensitive to large deviations than RMSE.

Using both mean absolute error (MAE) and root mean squared error (RMSE) allowed for improved balance in evaluating outliers. The distribution of several of our features lent themselves to extreme weather events, impacting individual predictions. MAE is less sensitive to outliers while having the risk of underfitting to those extreme events, while RMSE is more sensitive, penalizing those larger errors and having the tradeoff of potentially overfitting to them.

Additionally, all hyperparameter tuning employed 5-fold time-series cross-validation. Given the chronological nature of our ridership data, standard randomized KFold risked temporal leakage and would allow the model to inadvertently access future data during training. Using a time aware cross validation method preserved the integrity of the data's structure and provided a realistic evaluation environment reflective of actual predictive tasks.

NOTE: Lasso was run on log transformed data, but the other two models were not. This was discovered very late in the project, and thus the Evaluation Table below reveals this discrepancy.

Random Forest

We evaluated model performance using R^2 and RMSE, averaged over 5-fold cross-validation for each city-mode combination. Chicago Bus achieved a Mean R^2 of -0.14 and a Mean RMSE of 545. The Train achieved a Mean R^2 of 0.49 and a Mean RMSE of 317. New York City Bus achieved a Mean R^2 of 0.78 and a Mean RMSE of 312. The Subway performed slightly better with a Mean R^2 of 0.79 and a Mean RMSE of 816. The combined model, which aggregated all modes across both cities, had a Mean R^2 of 0.75 and a Mean RMSE of 1671.

These evaluation scores reveal notable variation in model performance across city and mode. The negative R^2 for CHI Bus indicates poor generalization, possibly due to higher variability or noise in that mode's ridership. In contrast, the CHI Train model, while not as strong as NYC's, showed some consistent patterns and a generalizable signal. NYC models consistently performed better than CHI, with Subway and Bus both showing high R^2 and relatively low RMSE, indicating strong predictive performance.

The model operating on the combined data, with an R^2 of 0.76 and a high RMSE, indicates a capturing of broad trends across transportation systems. However, it struggled with precision, perhaps due to aggregation smoothing out mode or city specific dynamics. Final test scores echoed these patterns, as the test R^2 for CHI Bus was 0.25, while the NYC Bus and Subway were 0.78 and 0.68,

respectively. The combined model achieved a final test R^2 of 0.63, though with a higher test RMSE of 2,326.

Lasso

Our analysis revealed moderate predictive performance for the combined (full) dataset, but notably stronger performance when modeling ridership separately by city and mode. This suggests that transit ridership dynamics are locally distinct and that a single global model might obscure the patterns specific to particular city/mode combinations. Interestingly, although our combined model initially suggested several air quality variables as significant predictors, these findings were not consistently replicated within individual subset models. This discrepancy likely arises due to local differences in sensitivity or collinearity with more predictive variables like temperature. These findings highlight the challenges inherent in attributing causal significance within complex observational datasets, particularly when predictor variables exhibit strong correlations.

Despite achieving a strong overall fit within subsets, air quality variables consistently demonstrated minimal explanatory power. Even the most influential air quality feature in each subset explained ridership variations of just a few riders per standard deviation change, or produced negligible coefficients. This limited impact likely reflects both a genuine lack of substantial short term air quality influence on daily ridership patterns and potentially a methodological limitation. Lasso regression is fundamentally linear and additive, and consequently it cannot capture complex nonlinearities, threshold effects, or conditional interactions. This could manifest as air quality influences becoming meaningful only under certain environmental conditions. Given these potentially critical interactions, we were hopeful that subsequent analyses using nonlinear methods like XGBoost or Random Forest were likely to provide deeper insight into how air quality influences transit ridership.

	state	mode	top_aq_feature	impact
0	NYC	bus	us_aqi_nitrogen_dioxide_max	+ ~4 riders
1	NYC	subway	us_aqi_max	- ~1 riders
2	CHI	bus	us_aqi_sulphur_dioxide_mean	- ~0 riders
3	CHI	train	us_aqi_max_bin	- ~0 riders

XGBoost

Overfitting was a potential concern with gradient boosting, so the lambda term was also included in grid search to test the impact of L2 (Ridge) regularization, with a higher lambda term controlling for overfitting more aggressively. L1 (Lasso) regularization was also considered, but the grid search selected 0 (XGBoost's default value) in each instance, with impacts of less than 1 on the MAE when it was moved. The final hyperparameters tested were `subsample` and `colsample_bytree`, with both attempting to use fractions of the rows and columns respectively, to reduce overfitting and introduce back an amount of randomness to the data that is able to be selected for each tree. Differences in their values were also shown to have impacts of less than 1 on MAEs across any transportation mode, so they were removed from the grid search and left at default values.

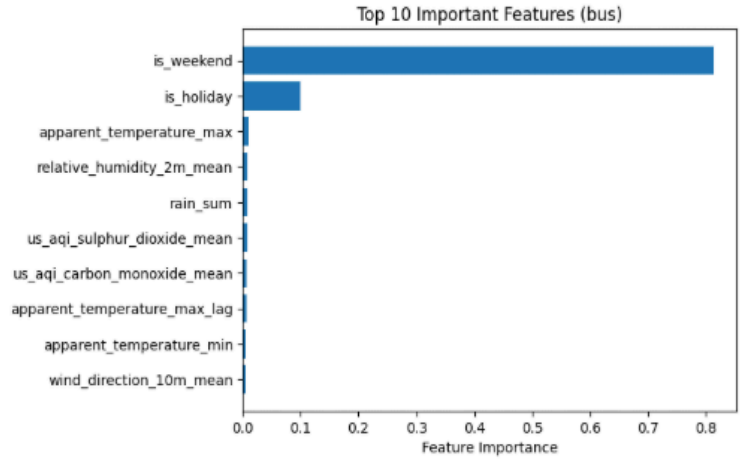
The other key consideration and potential tradeoff with the XGBoost model was computing time. To produce analyses across all of our transportation modes and splits, initial setups with grid search were taking over 40 minutes to run on just one subset of the data (CHI buses). Of the parameters being tuned and tested, `n_estimators` and `max_depth` were two we targeted that led us to capping the number of estimators to 300 (had originally tested up to 800), and the maximum depth to 5 (had originally tested up to 20 or none at all). Using the tree method `hist` also helped performance by binning values into histograms, rather than having the algorithm scan every unique value. With these improvements, we were able to reduce run time on each subset of the data to below 2 minutes, with many being below 1 minute. This sacrificed no more than 2% worse MAE or RMSE! This balanced our goals of speed vs accuracy and tuning control vs interpretability.

However, even with intensive hyperparameter tuning and testing, final results were extremely close to those of the random forest regressor. The moderate size of our aggregated dataset does not lend itself to the full advantage of gradient boosting relative to random forest, so improvements on accuracy metrics were relatively minor. Weekend and holiday indicator variables remained the most predictive, with temperature being relatively consistently represented across all transportation modes. The sum of rain in a given day was the next most predictive feature in NYC, while not showing up in the top 10 in CHI. On the other hand, CHI had PM2.5 air quality and total air quality as more predictive factors than they were in NYC.

All Models

Across all models, is_weekend was by far the most dominant feature (see chart below of NYC bus features), contributing between 30% and 80% of total feature importance depending on the mode and city. `is_holiday` also played a significant role, although generally less impactful than `is_weekend`. Other influential features include `apparent_temperature_max`, `apparent_temperature_min`, and their lagged versions, which were moderately important, reflecting seasonal or comfort-related ridership patterns. Variables like `us_aqi_pm2_5_mean`, `us_aqi_pm10_mean`, and `us_aqi_sulphur_dioxide_mean` showed low but non-trivial importance, particularly in the CHI models. The top features for each mode of transportation can be found in the Appendix.

```
]: randforr(NYC_df)
Best Parameters: {'max_depth': 10, 'min_samples_leaf': 2, 'min_samples_split': 25, 'n_estimators': 50}
R² scores (CV): [0.8927857927952702, 0.8748808744499593, 0.6059518758201989, 0.771008110218931, 0.786396078969822]
RMSE scores (CV): [248.46810046280652, 244.92705195163703, 464.43466172666075, 316.9971411478718, 288.5636347876195]
Mean R²: 0.7862 ± 0.1019
Mean RMSE: 312.68 ± 80.43
```




```
Test R²: 0.7779
```

Evaluation Metric Table (tables for each city/mode can be found in the Appendix)

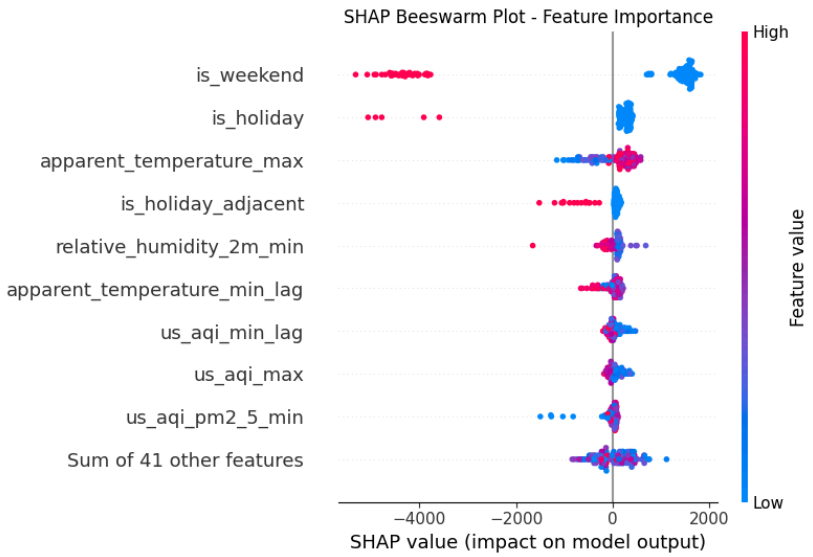
ALL CITIES, ALL MODES

Model	CV R^2 (mean)	CV R^2 (sd)	Test R^2	Test RMSE	Test MAE
XGBoost (All Features)	0.716	0.092	0.658	2224.8	1947.1
Random Forest	0.756	0.052	0.626	2326.7	1953.0
XGBoost (Reduced Features)	0.776	0.042	0.649	2254.4	1969.9
XGBoost (Holiday & Weekend Ablation)	-0.076	0.072	-0.054	3907.9	3505.4
XGBoost (AQI Only Ablation)	-0.056	0.056	-0.119	4026.0	3603.9
Lasso	0.412	0.073	0.206	1186926.5	755819.0

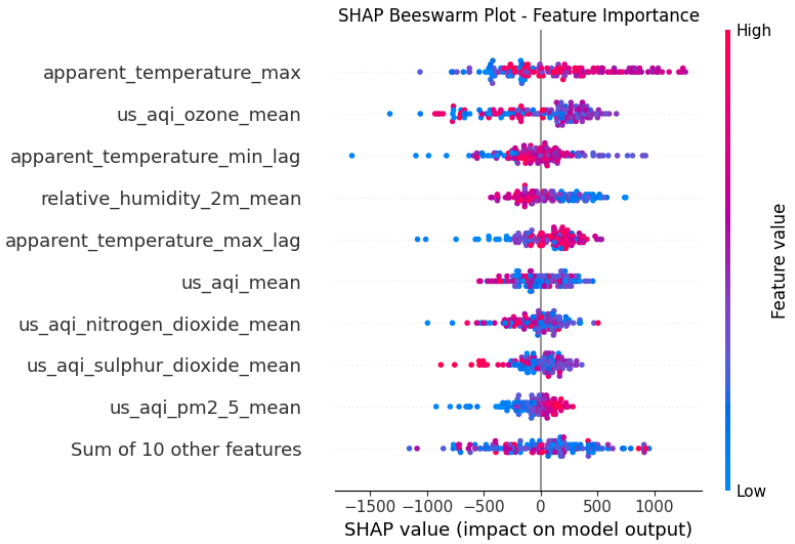
Feature Ablation & Failure Analysis

Feature Ablation (XGBoost)

We were able to further analyze our feature importance by using Shapely Additive exPlanations (SHAP) to examine the outputs of our XGBoost model. In our full dataset, the features `is_weekend` and `is_holiday` showed key importance through the SHAP values generated, allocating credit for the model's output among its input features. The beeswarm plot through the SHAP package is able to display these SHAP values for each feature across all predictions in the sample.



To perform an initial ablation analysis, the categorical features of `is_weekend`, `is_holiday`, and `is_holiday_adjacent` were removed to eliminate the only features not directly related to weather or air quality. This led to the maximum temperature of a day being the most important feature, but a relatively flat distribution of feature importance, and extremely poor performance overall. Somewhat surprisingly, rainfall and snowfall were not in the top 10 features here despite being commonly thought to impact public transportation ridership. There were few days of heavy snow in our dataset, and the ridership in those days did not stray far from expectation.



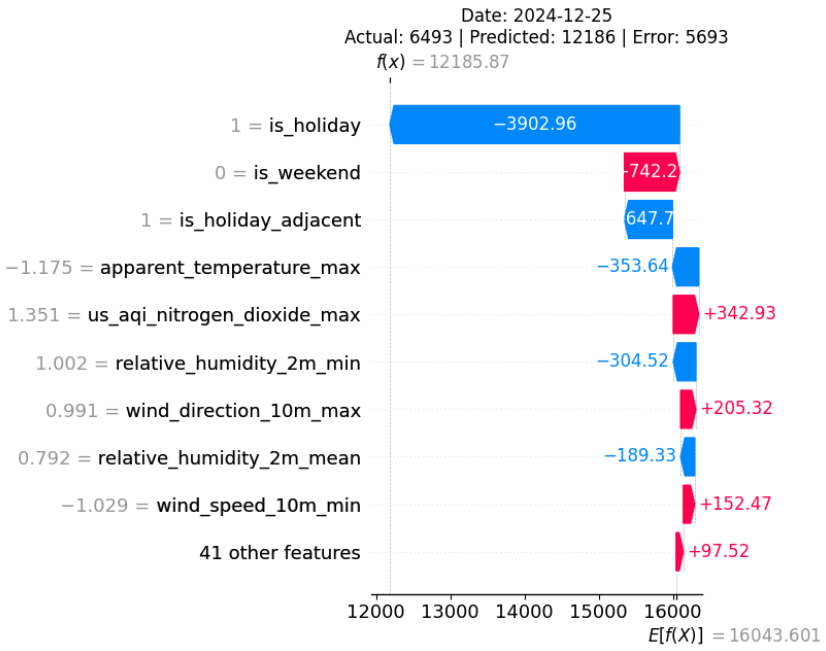
To test one final split, we wanted to strip out all temperature, wind, and precipitation data to solely evaluate the AQI factors. This was run on the entire dataset of all cities and transportations nodes. It had similar performance to the prior ablation, with even worse error metrics. These features were tracked for their minimum, maximum, and mean values throughout each day. What stood out in this ablation was that the ozone (O3) index was found to be the most impactful of the AQI metrics when the model was run with just the AQI features (see appendix for visualization).

Failure Analysis (XGBoost)

Shapely Additive exPlanations (SHAP) can also be used to explore individual predictions through visualizations such as waterfall plots. These are able to show an individual prediction and the exact impact of each feature on that prediction. To examine our model's failures, we analyzed our three largest raw errors of predicted daily ridership vs actual (across all modes).

Two of those three largest errors were on successive days: November 11th and November 12th , 2024 (see appendix for visuals). November 11th happens to be the federal holiday of Veterans Day, which was flagged by our binary holiday variable, yet is not a holiday that is acknowledged by most private businesses. That bore out in the data as on the 11th the actual ridership was 5,169 higher than our prediction. The SHAP values show that the holiday variable reduced our ridership projection by 5,054. Our error on November 12th was not as clearly explainable. Holiday adjacency led our model to lower its prediction, but given that the 11th was not a standard holiday that is an understandable error. The largest impact came from the AQI PM2.5 variable but the actual total ridership far exceeded the movement of any pattern of features, pointing to a level of random behavior on a high ridership day.

The other largest error was a logical day: Christmas. Holiday was again the feature most predicting the result and failed to measure the scale of holidays' importance.These events displayed the full spectrum of public behavior, with some treating the holiday as close to a full shutdown (Christmas) or close to a normal day (Veterans Day). This was a logical error that could only be corrected if we were to add additional learning to our holiday feature or add another variable further breaking down the holiday. The waterfall plot for 12/25/2024 is shown at the top of the following page, with the plots for 11/11/24 and 11/12/24 in the Appendix.



Discussion

As we began our supervised learning workflow, we used our ‘final’ dataset that had stop-level granularity for daily ridership. Unfortunately, computational constraints appeared immediately. After only a single run for each model, we made the decision to pivot to another organization of our data. We aggregated the stop-level data into `daily_ridership` for an entire city, for a given mode, for a given date. As the air quality and weather entries were derived from a single station for all daily readings, we were not required to transform that data. This had the further implication of removing both raw stop/station IDs, and the SVD transformed versions. This pivot allowed our dataset size to be reduced significantly, and allowed modelling to continue.

The most significant learning throughout this workflow was that air quality metrics did not impact ridership in a significant way in our analysis. This was quite surprising, but an unavoidable conclusion using our methodology. Furthermore, this highlighted for us why certain choices were made in a few of the studies we referenced when designing this project. They often focused on one metropolitan region, and for a longer period of time. Given our supervised model prediction results on the data subsets, we would make this choice as well were we to undertake a similar project in the future. Lastly, as air quality trends continue to accelerate along their current trajectory, and public recognition and attention increase, these metrics could eventually become more impactful to public transportation usage.

Ethical Considerations

While the purpose of this project was to highlight the impact air quality has on public transportation usage, we did not want to overstate this impact, or advocate for the behavior observed. As we did not discover a strong relationship between air quality and ridership, the ethical dimensions of this project diminished, but this may have been a function of flawed methodology and may not reflect real world conditions.

As noted in the introduction, mass public transportation adoption can substantially contribute to improvements in air quality and climate change mitigation. However, if public transportation is not sufficiently used due to existing air quality and weather conditions we risk blunting positive impacts. We seek to inform public policy stakeholders of reinforcing trends, allowing them to advocate for a 'virtuous cycle' of public transportation adoption and improved air quality, rather than a 'downward spiral' of private transportation utilization, reduced individual mobility, and degraded air quality.

Statement of Work

- **Matt Cott**
  - Project Design
  - Model Selection Background
  - Supervised Learning (XGBoost)
    - Failure Analysis
    - Feature Ablation
  - Project report
- **Mickey Giarmarco**
  - Project Design
  - Data Acquisition (Air Quality, Weather)
  - Feature Engineering (Truncated SVD, lag and holiday features)
  - Unsupervised Learning (Principal Component Analysis)
  - Supervised Learning (Random Forest)
  - Project Report
- **Mark Griffin**
  - Project Design
  - Data Acquisition (Public Transportation)
  - Unsupervised Learning (DBSCAN, Agglomerative Clustering)
  - Supervised Learning (Lasso)
  - Project Report

References

- https://websites.umich.edu/~kevynct/mads_ml_synthesis.pdf
- https://www.geeksforgeeks.org/machine-learning/shap-a-comprehensive-guide-to-shapley-additive-explanations/
- https://pypi.org/project/shap/
- https://www.datacamp.com/tutorial/tutorial-lasso-ridge-regression

Citations

1. https://css.umich.edu/publications/factsheets/climate-change/climate-change-policy-and-mitigation-factsheet
2. ScienceDirect search on 'air quality'
3. Air pollution hindering a transit-oriented city: Examining the association of particulate matter concentration with public transit ridership and road traffic in Seoul, South Korea - ScienceDirect
4. Air Pollution and Seasonality Effects on Mode Choice in China - Weibo Li, Maria Kamargianni, 2017
5. Particulate matter concentration and composition in the New York City subway system

Appendix

Original Data Sources

- https://historical-forecast-api.open-meteo.com/v1/forecast
- https://air-quality-api.open-meteo.com/v1/air-quality
- https://data.cityofchicago.org/Transportation/CTA-Ridership-Bus-Routes-Daily-Totals-by-Route/jyb9-n7fm/data_preview
- https://data.cityofchicago.org/Transportation/CTA-Ridership-L-Station-Entries-Daily-Totals/5neh-572f/about_data
- https://data.ny.gov/Transportation/MTA-Daily-Ridership-Data-2020-2025/vxuj-8kew/about_data

Final Model Results by City & Mode of Transportation

Model	City	Mode	CV R^2 (mean)	CV R^2 (std)	Test R^2	Test RMSE	Test MAE
Lasso	CHI	bus	0.673	0.026	0.669	76121.8	65179.5
Random Forest	CHI	bus	-0.145	0.446	0.251	635.0	541.7
XGBoost	CHI	bus	-0.063	0.409	0.252	634.6	556.9

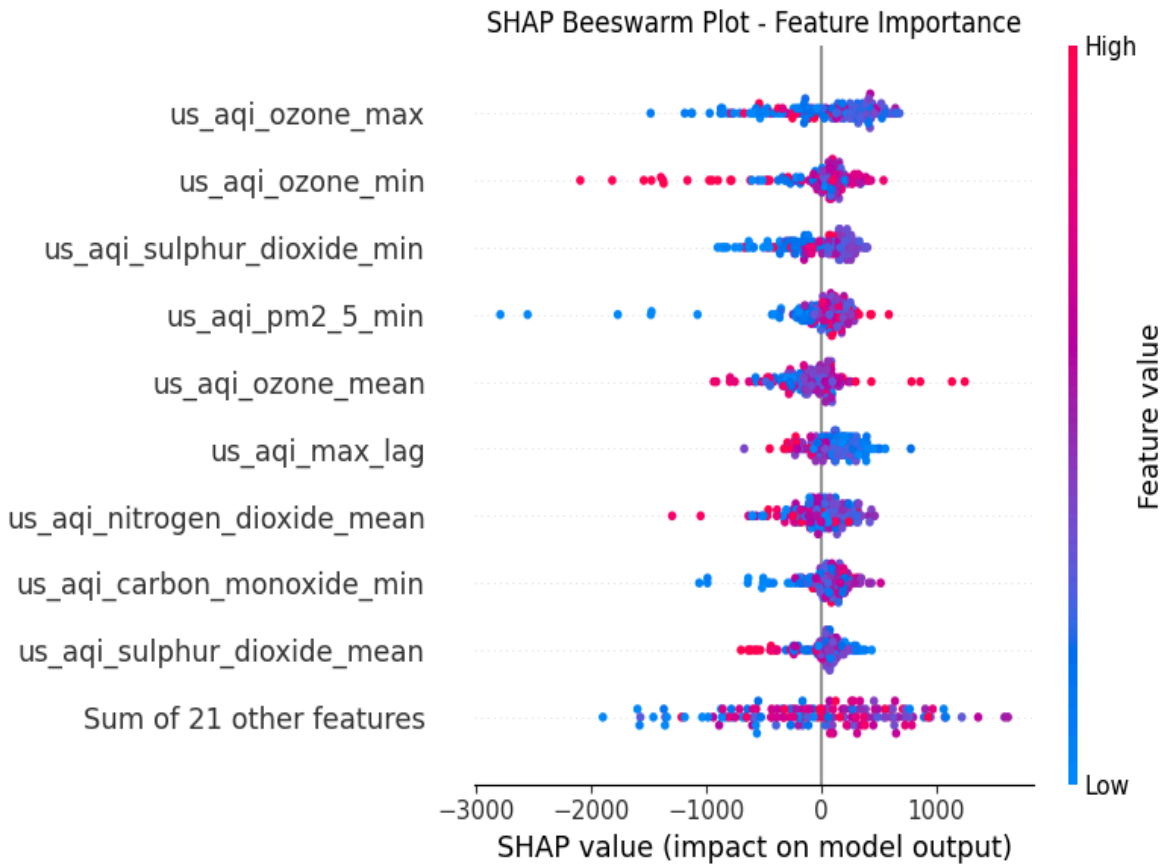
Model	City	Mode	CV R^2 (mean)	CV R^2 (std)	Test R^2	Test RMSE	Test MAE
Lasso	CHI	train	0.615	0.082	0.518	52244.5	41803.6
Random Forest	CHI	train	0.496	0.148	0.583	337.5	266.2
XGBoost	CHI	train	0.503	0.162	0.612	325.6	259.4

Model	City	Mode	CV R^2 (mean)	CV R^2 (std)	Test R^2	Test RMSE	Test MAE
Lasso	NYC	bus	0.773	0.096	0.752	151071.3	130709.1
Random Forest	NYC	bus	0.786	0.102	0.778	356.3	295.4
XGBoost	NYC	bus	0.769	0.071	0.781	354.0	298.6

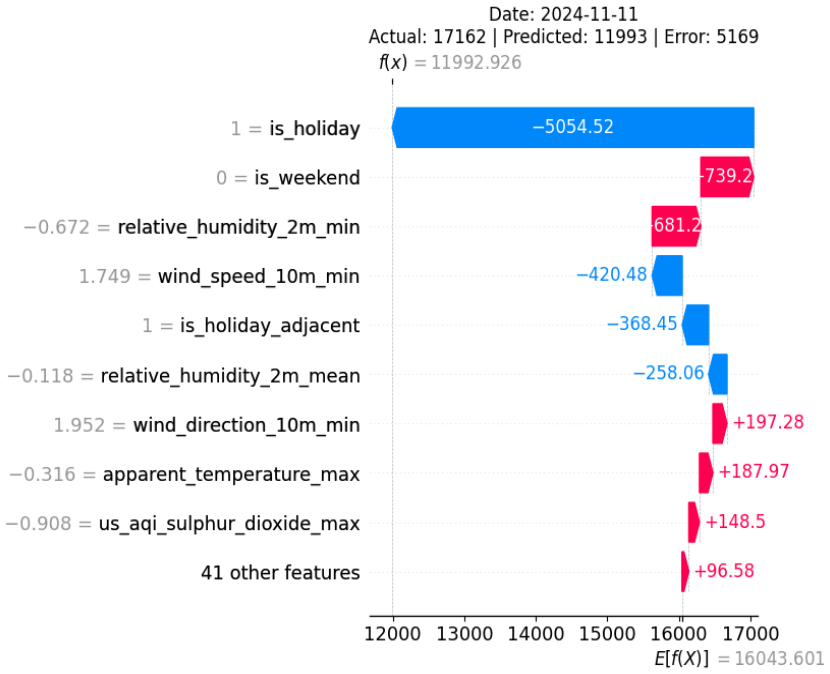
Model	City	Mode	CV R^2 (mean)	CV R^2 (std)	Test R^2	Test RMSE	Test MAE
Lasso	NYC	subway	0.785	0.092	0.647	503014.1	429736.3
Random Forest	NYC	subway	0.800	0.034	0.675	1125.0	928.0
XGBoost	NYC	subway	0.809	0.027	0.692	1095.5	924.6

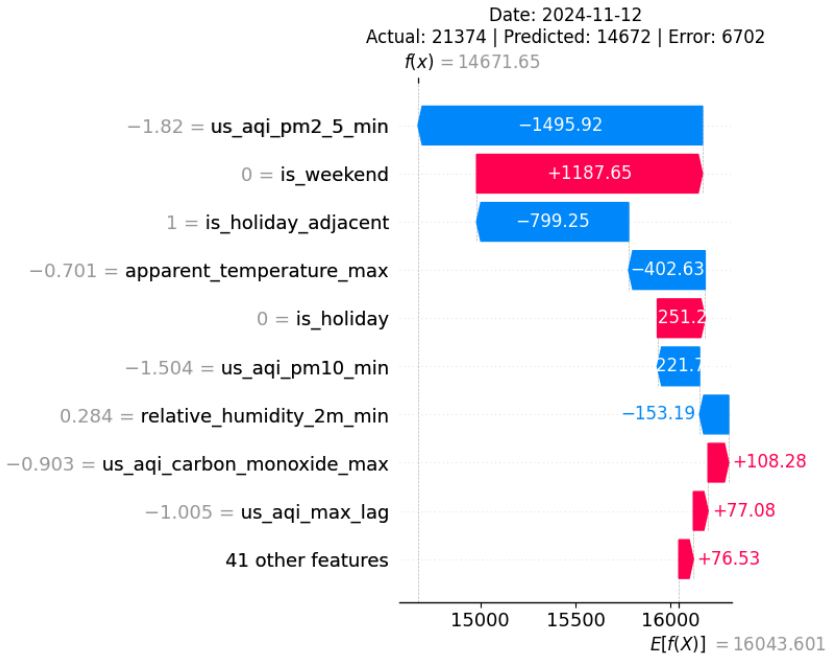
Additional XGBoost SHAP Visuals

AQI-only Ablated Data Feature Importance:



November 11th, 2024 and November 12th, 2025 Error Analysis:

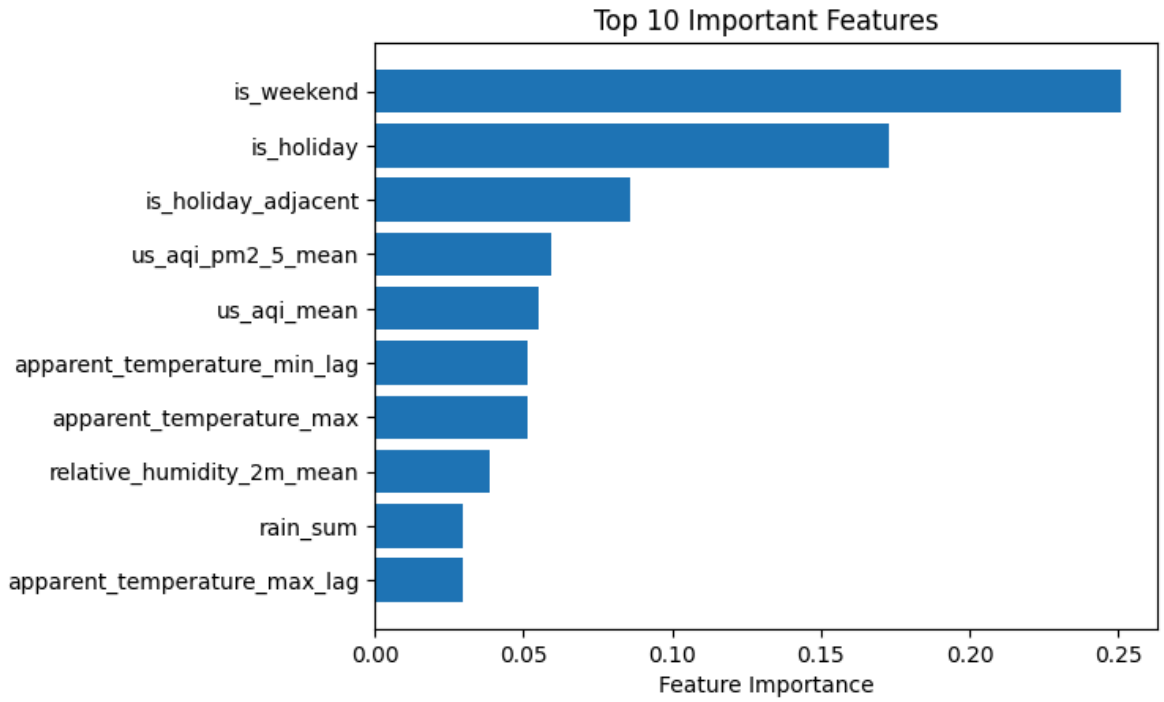




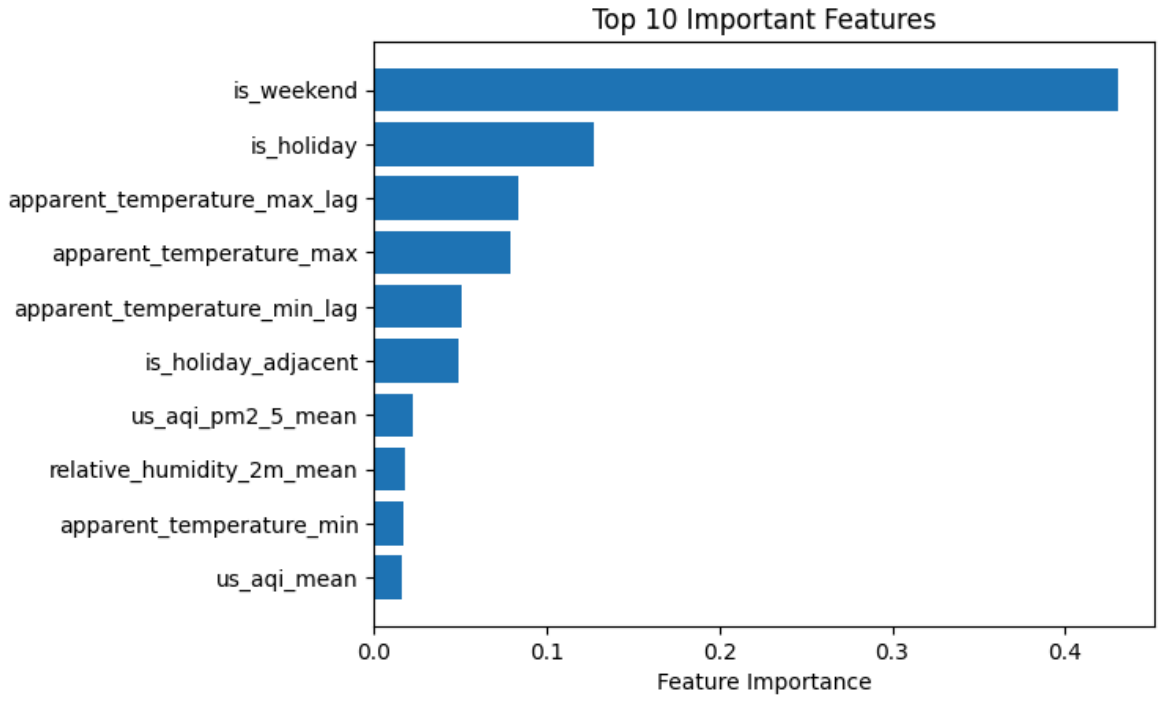
Note that for the waterfall plots and their analysis of prediction errors, the E[f(x)] represents the average prediction of all days - 1.60 million daily riders with the waterfall showing the model's movement from that base expectation to the final prediction (e.g. 1.47 million riders on November 12th).

XGBoost Regression Feature Importance by City and Mode

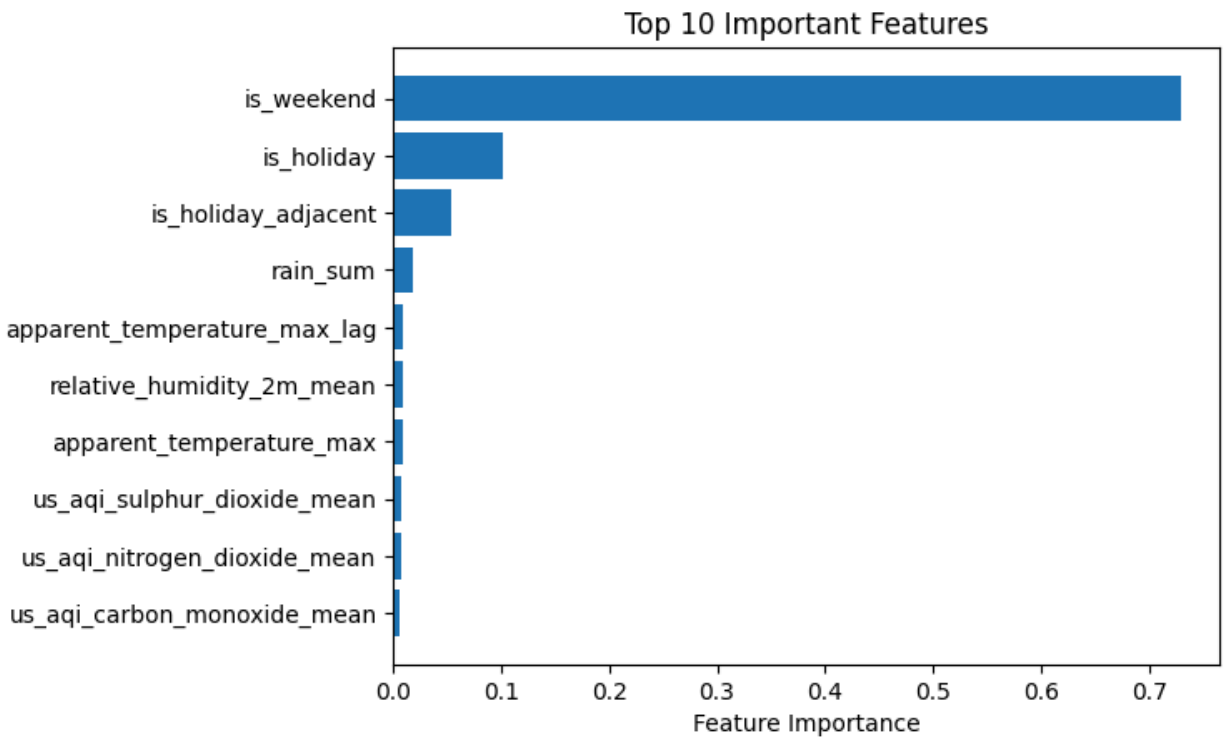
Chicago bus:



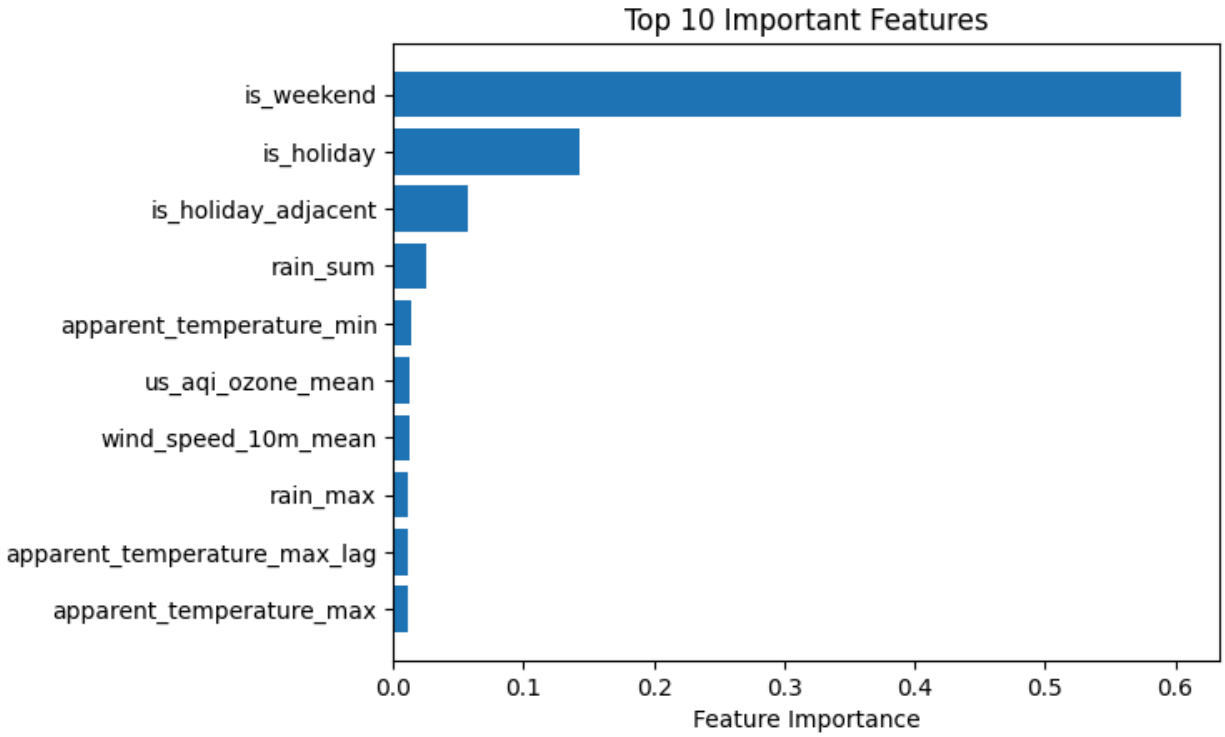
Chicago train:



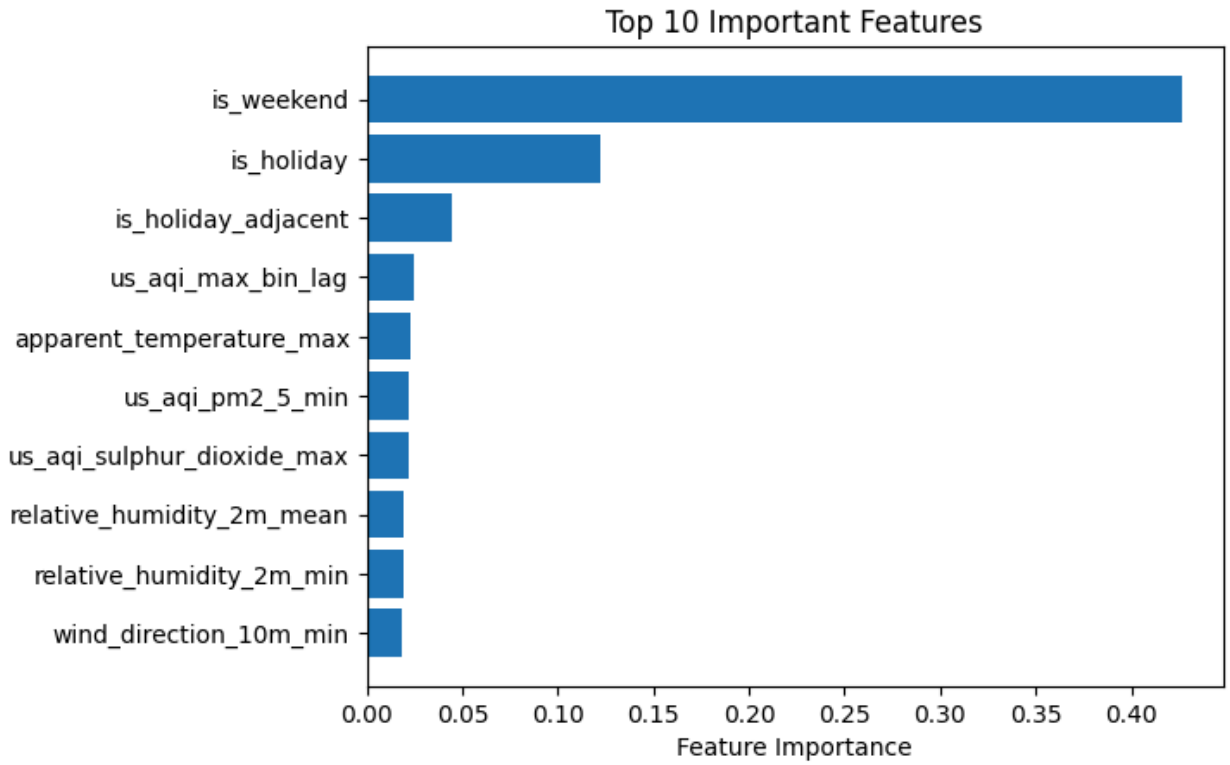
New York bus:



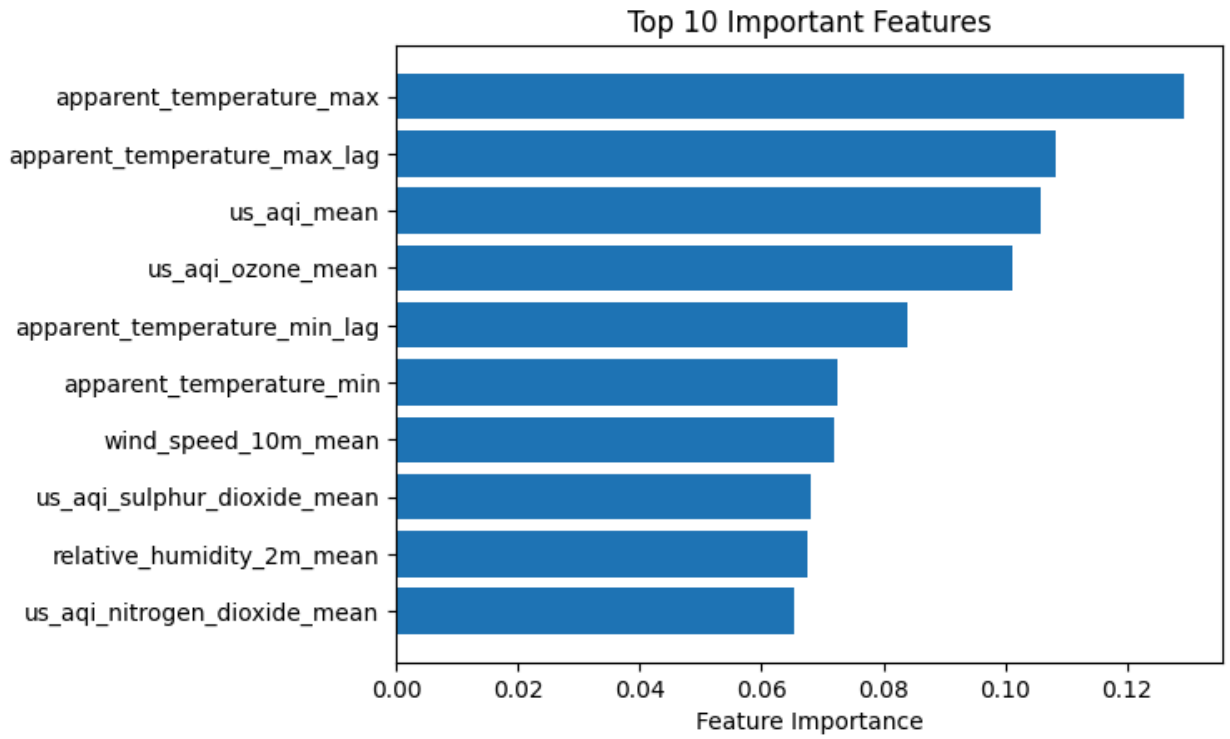
New York subway:



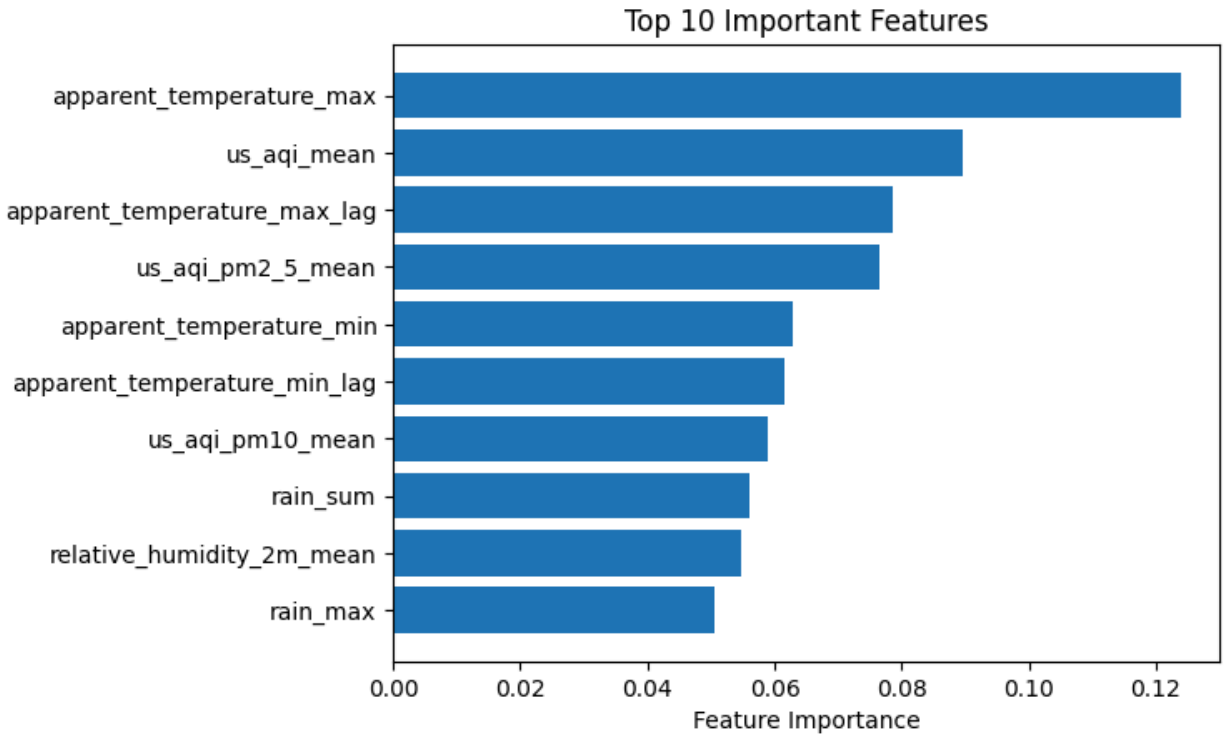
Combined dataset:



Combined dataset w/ Ablation:

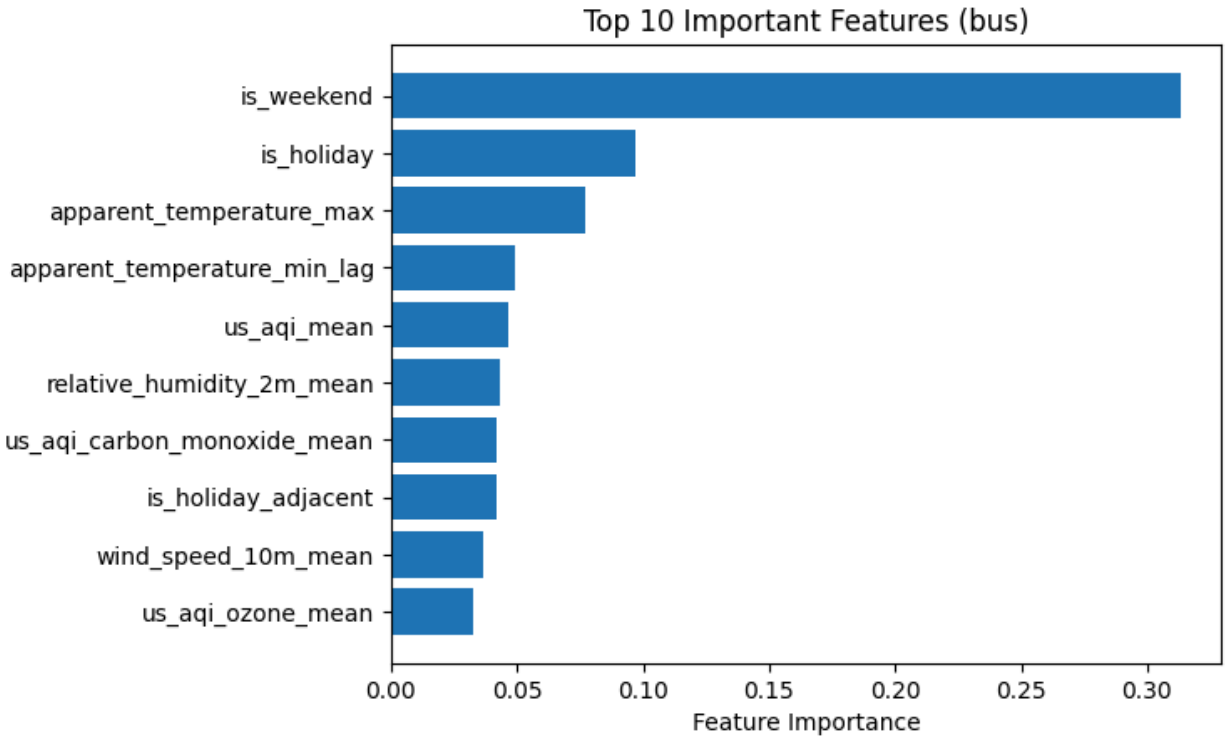


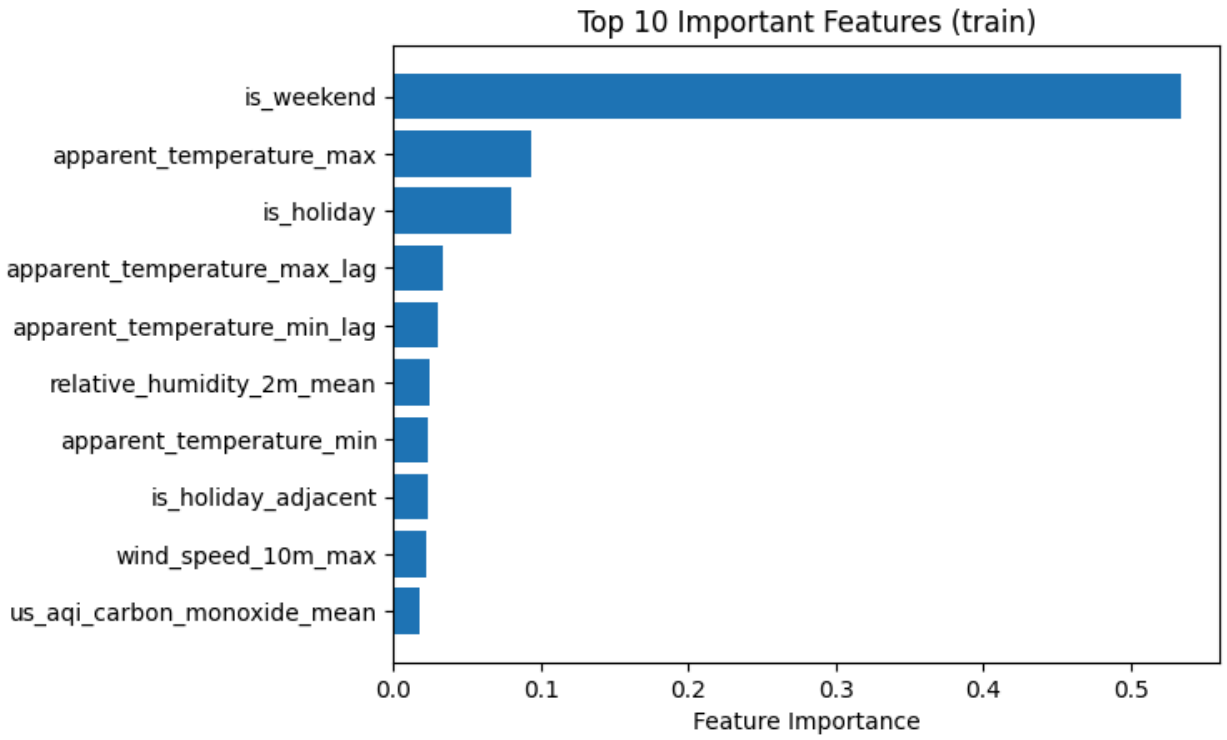
Combined dataset w/ Ablation to just AQI features:



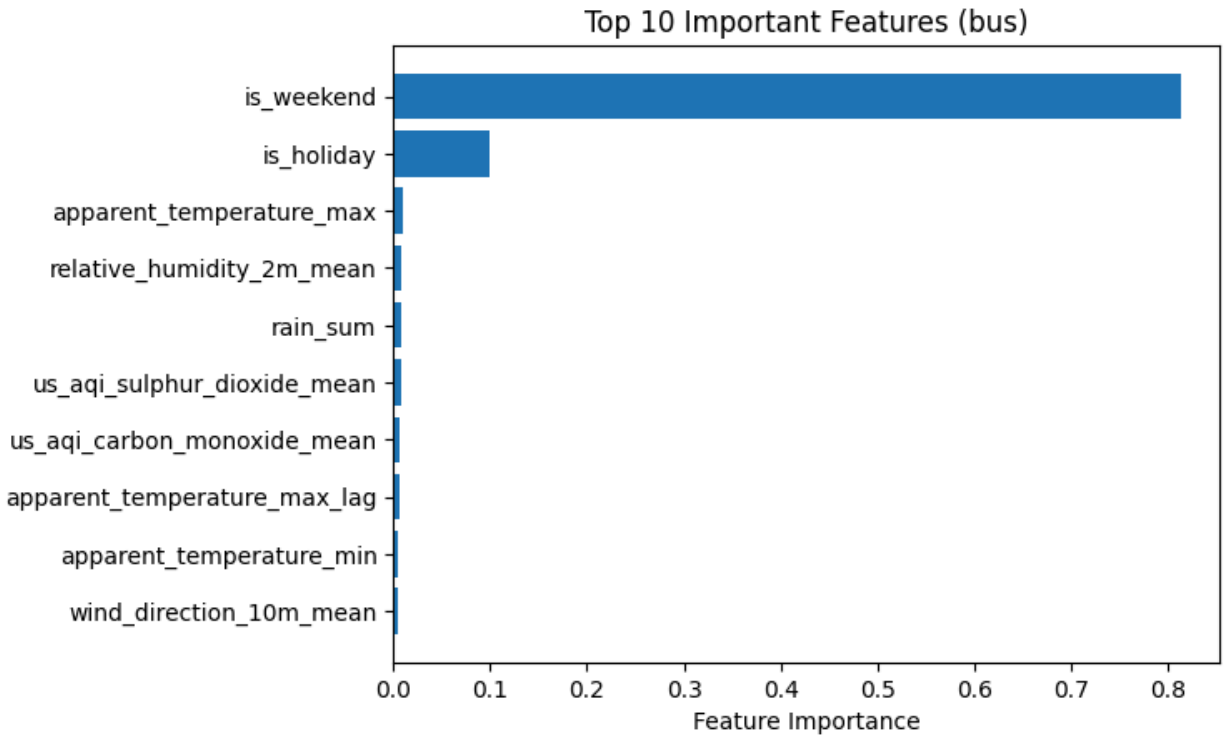
Random Forest Regression Feature Importance by City and Mode

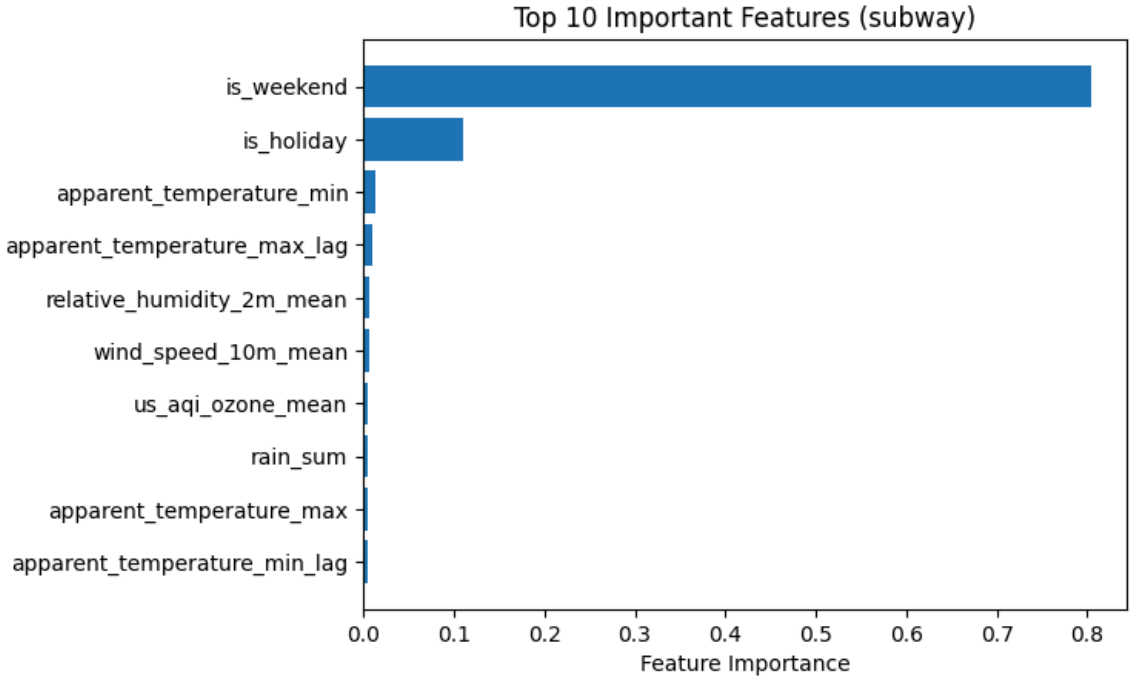
Chicago:



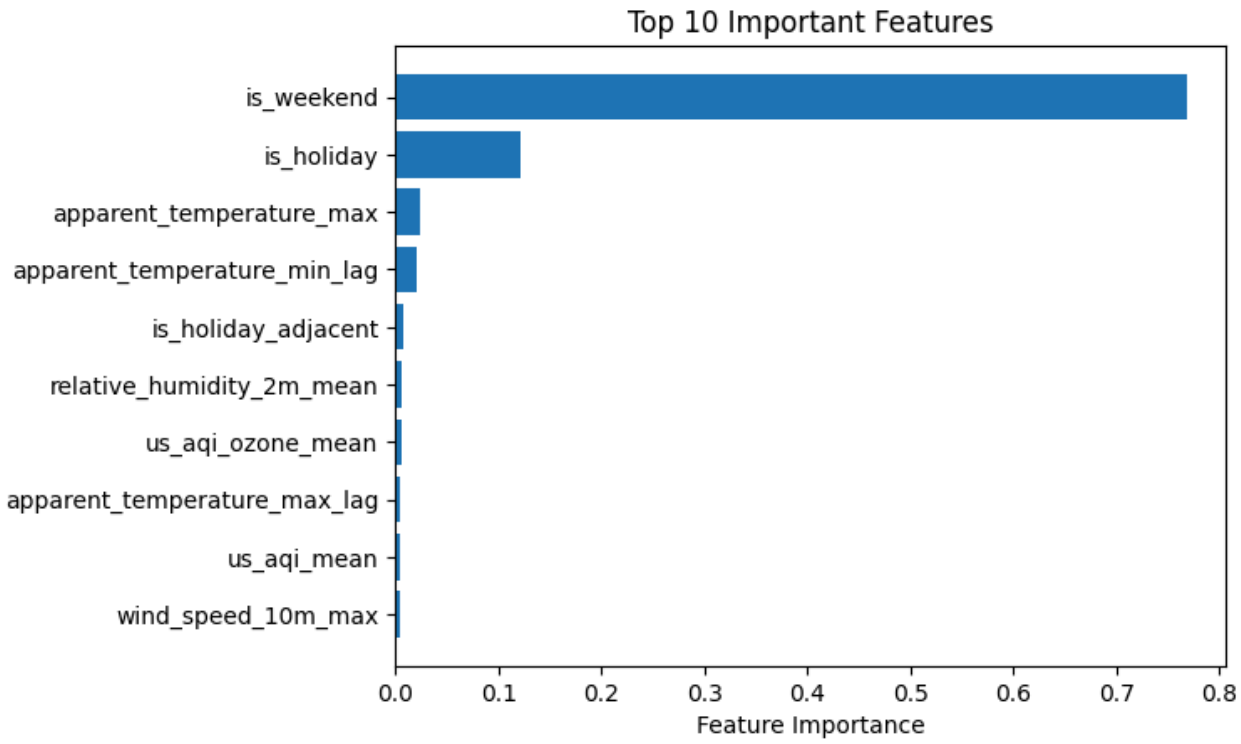


New York:


Top 10 Important Features (bus)
is_weekend
is_holiday
apparent_temperature_max
relative_humidity_2m_mean
rain_sum
us_aqi_sulphur_dioxide_mean
us_aqi_carbon_monoxide_mean
apparent_temperature_max_lag
apparent_temperature_min
wind_direction_10m_mean
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
Feature Importance


Top 10 Important Features (subway)
is_weekend
is_holiday
apparent_temperature_min
apparent_temperature_max_lag
relative_humidity_2m_mean
wind_speed_10m_mean
us_aqi_ozone_mean
rain_sum
apparent_temperature_max
apparent_temperature_min_lag
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
Feature Importance

Combined dataset:



Feature List (‘Final’ Dataset)

- 'date',
- 'state',
- 'mode',
- 'daily_ridership',
- 'rain_sum',
- 'rain_max',
- 'snowfall_sum',
- 'snowfall_max',
- 'relative_humidity_2m_min',
- 'relative_humidity_2m_max',
- 'relative_humidity_2m_mean',
- 'apparent_temperature_min',
- 'apparent_temperature_max',
- 'apparent_temperature_min_lag',
- 'apparent_temperature_max_lag',
- 'wind_speed_10m_min',
- 'wind_speed_10m_max',
- 'wind_speed_10m_mean',
- 'wind_direction_10m_min',
- 'wind_direction_10m_max',
- 'wind_direction_10m_mean',
- 'us_aqi_pm2_5_min',
- 'us_aqi_pm2_5_max',
- 'us_aqi_pm2_5_mean',
- 'us_aqi_pm10_min',
- 'us_aqi_pm10_max',
- 'us_aqi_pm10_mean',
- 'us_aqi_min',
- 'us_aqi_max',
- 'us_aqi_mean',
- 'us_aqi_min_bin',
- 'us_aqi_min_bin_lag',
- 'us_aqi_min_lag',
- 'us_aqi_max_bin',
- 'us_aqi_max_bin_lag',
- 'us_aqi_max_lag',
- 'us_aqi_mean_bin',
- 'us_aqi_mean_bin_lag',
- 'us_aqi_mean_lag',
- 'us_aqi_nitrogen_dioxide_min',
- 'us_aqi_nitrogen_dioxide_max',
- 'us_aqi_nitrogen_dioxide_mean',
- 'us_aqi_carbon_monoxide_min',
- 'us_aqi_carbon_monoxide_max',
- 'us_aqi_carbon_monoxide_mean',
- 'us_aqi_ozone_min',
- 'us_aqi_ozone_max',
- 'us_aqi_ozone_mean',
- 'us_aqi_sulphur_dioxide_min',
- 'us_aqi_sulphur_dioxide_max',
- 'us_aqi_sulphur_dioxide_mean',
- 'is_weekend',
- 'is_holiday',
- 'is_holiday_adjacent’