Citi Bike Usage in New York City

Big Data Systems (DS 5110)

August 11, 2021

Diana McSpadden (hdm5s)
Nick Daniello (njd9e)
David Fuentes (dmf4ns)
Eric Sarani (es5cj)
Abigail Bernhardt (aeb4rv)

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

1 ABSTRACT

New York City (NYC) offers residents and visitors a multitude of transportation options, but they must consider numerous factors when deciding which option to use. For this analysis we obtained data from Lyft's Citigroup-sponsored bike-share program, Citi Bike. By analyzing bike trips between 2018 and 2021, we discovered how changing conditions affect bike usage and how bike usage affects the city. Variables studied included bike-trip specifics (date-time, location, distance), weather, and real estate metrics (zip code, price, etc.). By employing K-Means, Graph Neural Network, Random Forest, Logistic and Linear Regression models, we observed:

1. Most Citi Bike trips start and end in the same neighborhood, indicating customers use Citi Bike for short trips, or to augment other public transportation methods for longer trips.
2. Most trips start and end below Central Park. These trips average the shortest distances, most likely due to the increased density of public transportation options in comparison to outer neighborhoods and boroughs.
3. Citi Bike strategies for distribution of bikes, i.e., "rebalancing", must consider weather conditions for optimal bike and docking availability throughout the Citi Bike network.
4. Holiday bike behavior differs; thus, Citi Bike must consider holidays when planning rebalancing.
5. The pandemic had a significant impact on bike behavior.
6. Exogenous factors, such as weather, often determine if a rider will use a bike, but once started, ride characteristics are relatively unchanged regardless of circumstances.

2 DATA AND METHODS

2.1 DATA

In recent years, shared services have become increasingly popular. "Shared service" is a business model allowing communities to leverage shared resources, resulting in lower cost goods or services for individuals. For this project, our group analyzed Lyft's Citigroup-sponsored New York City (NYC) bike-share program, Citi Bike. Citi Bike allows the NYC community and tourists a convenient and affordable way around town without the need to store and maintain a personal bike.

Our main dataset was publicly available: https://s3.amazonaws.com/tripdata/index.html.

The data were provided in zip files organized by month and year, ranging from June 2013 to June 2021. To work with less than 3GB of data, we focused on bike trips between August 2018 and April 2021. Each row represents a bike trip and each column a variable related to the trip. In addition to the Citi Bike dataset, we added a secondary NYC weather dataset and a tertiary NYC real-estate dataset. An example of the data showing the first four variables is shown below.

Table 2-1: First four variables in Citi Bike dataset

Borough	month_year	startStationId	startStationName	startStationLatitude
Brooklyn	2020-06	3419	Douglass St & 4 Ave	40.67927879999999
Brooklyn	2020-06	366	Clinton Ave & Myr...	40.693261

only showing top 2 rows

2.2 EXPLORATORY DATA ANALYSIS

2.2.1 Combining Datasets

- The Citi Bike data index is date/time of the bike trip, start location and end location.
- NYC weather data was joined by date/time of the bike trip.
- After engineering the zip code from station latitude and longitude, NYC real estate data were joined by borough of the start location to each Citi Bike trip[1].

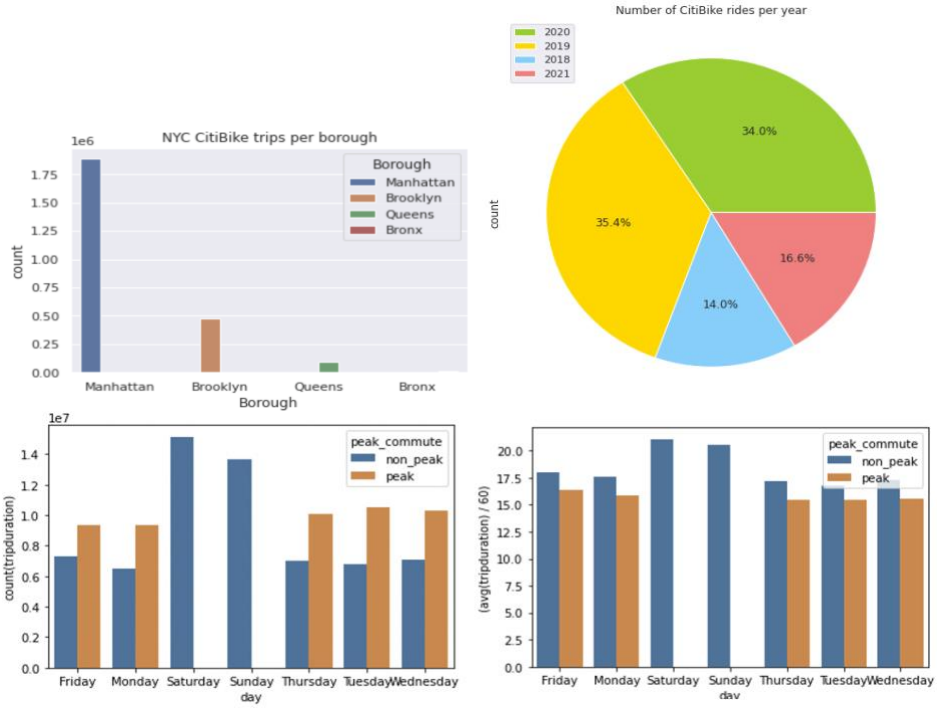
[1]See section "Zip Code and Borough Mapping."

2.2.2 Preprocessing Data: Removing Variables and Observations

- Removed *gender* from Citi Bike data; 4 million records contained unknown gender and 7 million records were null[2].
- Removed *birthyear* from Citi Bike data. The birth year values ranged from 1857 to 2005. Citi Bike launched in May 2013, which makes it more or less impossible for anyone born in the 1800s or early 1900s to be members.
- Removed *bikeid* from the Citi Bike data. 7 million observations were without *bikeid*, and we did not analyze Citi Bike behavior by individual bikes.
- Dropped 13,885 trips (observations) from Citi Bike data. These observations contained missing end location information, most likely caused by a user beginning a Citi Bike rental and immediately returning the bike. The start station name and coordinate points were recorded correctly, but the end-station name and coordinates were recorded as null.

After EDA[3], our dataset contained 49,380,335 records and 38 variables.

Figure 2-2: Simple EDA Plots



2.3 FEATURE ENGINEERING

2.3.1 Crow-Flies Distance

The Citi Bike dataset provided latitude and longitude metrics for every start and end bike station, which were angle measures. While possible to use latitude and longitude as features in our various models,

[2] Additionally, three times as many males than females were recorded with Citi Bike accounts. After exploring the discrepancy between male and female accounts, we learned that there has been a recent change to the Citi Bike enrollment process, where individuals are now asked to record their pronouns instead of gender.
[3] See BikeEDA.ipynb file.

it seemed more applicable to convert these measures to miles. To transform this feature, we used the haversine formula.

Given NYC streets often follow a grid pattern, we initially considered calculating the *startStationLatitude / Longitude* and *endStationLatitude / Longitude* station distance using 'Manhattan-distance'. However, much of our dataset fell outside the Manhattan Street grid of Midtown to Uptown. Because of this, we decided to use the straight distance "as the crow flies" for our distance measures. We feel this was sufficiently accurate for our purposes and any increase in crow-flies distance had an acceptable scalar increase in Manhattan-distance.

2.3.2 Categorical Bins

Due to the large quantity of data in this analysis, it was necessary to conduct numerical-to-categorical binning. This approach facilitated efficient modeling and eased the interpretation of excess granularity in quantitative values. For example, we binned ride times (HH:mm) to better capture how bicyclists typically broke up their day. Specifically, we did not think it was important to differentiate between using a Citi Bike at 8 AM as opposed to 9 AM on a weekday, but rather that the trip occurred during the peak morning commute time.

Aside from categorical binning, we employed standard feature cleansing to make our data more human legible. We employed splitting in Spark to transform date-time data to individual fields, like *month*, *dow* (day-of-week), etc. We mapped via simple dictionary lookups to transform these discrete quantitative data to categorical data, e.g., January instead of 1, February instead of 2, and so on. And later, these data were one-hot encoded prior to modeling.

2.3.3 Good/Bad Weather

A good/bad weather category, by trip, was created[4]. Good or bad weather is relative to a seasonally comfortable day; our "good" / "bad" label uses monthly low and high temperature thresholds and precipitation[5]. If a trip occurred in a time range with precipitation, it was assigned to the *weatherBadDurationIndex* variable. **Appendix A Table 1-1** displays monthly low and high temperatures resulting in a "bad" label for a trip.

2.3.4 Station Groupings

2.3.4.1 Zip Code and Borough Mapping

Considering individual bike stations in our models turned out to be unnecessary, particularly given there were greater than 1500 individual Citi Bike stations spread throughout NYC between August 2018 and June 2021. NYC is split into boroughs, namely Manhattan, The Bronx, Brooklyn, Queens, and Staten Island (although no start stations were in Staten Island). These boroughs provided a clear and obvious partition on which to perform analyses and run our models[6]. However, the Citi Bike dataset did not include borough, incorporating only latitude and longitude for each station. To overcome this, we built a Python script to map each station's latitude and longitude to its zip code via the GeoPy package[7]. Aside from allowing us to join (via Spark) each zip code to a borough, having zip code data as a feature variable led to new modeling opportunities, including the incorporation of real-estate data.

[4] See stationWeatherEffectIndex.ipynb file.

[5] Wind speed was considered, but extreme wind days are almost entirely captured by precipitation.

[6] Citi Bike has also spread farther out into New Jersey, but this is a recent development, so those stations are not incorporated in our analysis.

[7] See create_zips.ipynb file.

Table 2-2: Stations by Borough (left); Rides by Borough (right)

Borough	# Rides	% Total
Queens	1849398	3.74
Brooklyn	9574892	19.38
Manhattan	37688969	76.3
Bronx	280961	0.57

Borough	# Stations	% Total
Queens	179	11.72
Brooklyn	471	30.84
Manhattan	651	42.63
Bronx	226	14.8

2.3.4.2 Station Location K-Means

To better balance bike trips, we further partitioned boroughs into neighborhoods[8]. Rather than arbitrarily determining cut-off latitudes and longitudes to create neighborhoods manually, we used machine learning, running a K-Means algorithm by station location, and using latitude and longitute to create splits of Manhattan and Brooklyn. We reasoned that if this approach worked properly, a K-Means algorithm would return logical neighborhood groupings that could be confirmed by visualizing output data. Based on our understanding of NYC and initial ride and station distributions, grouping Manhattan and Brooklyn into 5 and 3 neighborhoods, respectively, while leaving Queens and the Bronx as their own individual neighborhoods made the most sense.

The K-Means algorithm resulted in the groupings in **Figure 2-2**. Notice that NYC's natural features, such as its rivers, bays, and Central Park, are made apparent by the lack of Citi Bike start stations.

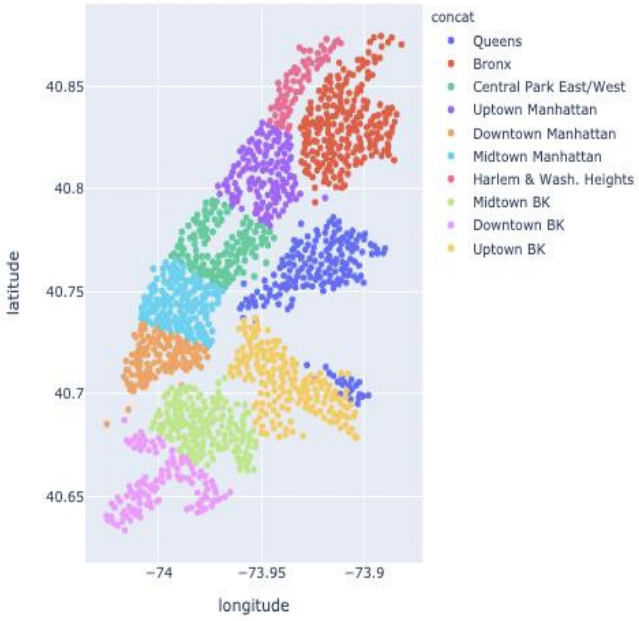


Figure 2-2: K-Means Neighborhood Grouping

Table 2-3: Rides by K-Means Neighborhood (left); Stations by K-Means Neighborhood (right)

start_neighborhood	# Rides	% Total
Midtown Manhattan	15825350	32.12
Downtown Manhattan	9991879	20.28
Central Park East...	8245275	16.73
Midtown BK	5233122	10.62
Uptown BK	4002562	8.12
Uptown Manhattan	3139583	6.37
Queens	1917315	3.89
Downtown BK	488525	0.99
Bronx	280608	0.57
Harlem & Wash. He...	146790	0.3

start_neighborhood	# Stations	% Total
Bronx	226	14.8
Uptown BK	183	11.98
Queens	179	11.72
Midtown BK	172	11.26
Midtown Manhattan	171	11.2
Uptown Manhattan	138	9.04
Downtown Manhattan	138	9.04
Central Park East...	137	8.97
Downtown BK	116	7.6
Harlem & Wash. He...	67	4.39

Though, still imbalanced by ride count, the neighborhoods were more even by the station count. Further, by visualizing the K-Means output, we saw that the neighborhoods make intuitive sense. We reasoned that trying to better balance the neighborhoods by number of rides would improperly misrepresent the data, leading to results that did not properly model real life.

[8] NYC has many neighborhoods, and the physical location of each -- and even the total number -- is often something of a contentious debate.

2.3.4.3 Bike Behavior K-Means

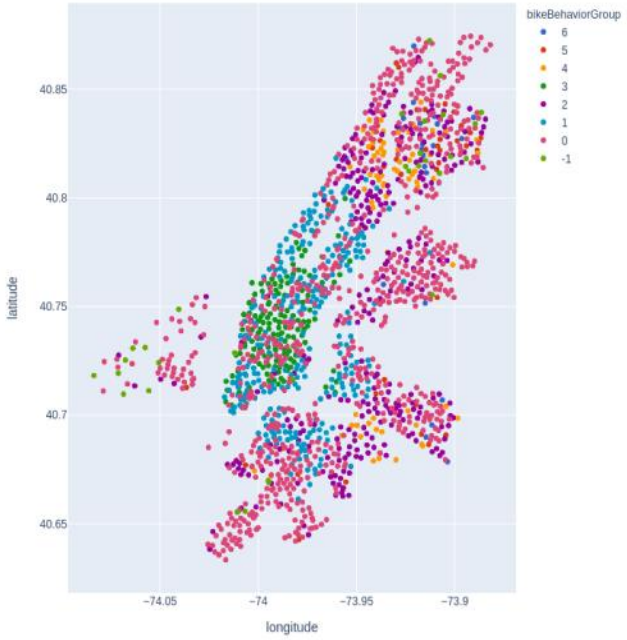
A categorization of Citi Bike stations by bike behavior helped model performance and interpretation. To create the bike behavior category, we employed a K-Means model using aggregate trip behavior by the various *endStation*[9] variables. The following aggregate features were calculated using Spark for trips by end station:

1. Avg and std *trip_duration*
2. Avg and std *crowDist* (crow-flies distance)
3. Avg and std *trip_duration* and *crowDist* for *weatherGoodDurationIndex* and *weatherBadDurationIndex* trips
4. Avg count of trips for *weatherGoodDurationIndex* and *weatherBadDurationIndex* trips
6. Avg count of trips for *PEAK* and *NON-PEAK* hours
7. Avg and std *trip_duration* and *crowDist* for *dow*
8. Avg count of trips for *dow*

The features above were MaxAbsScaled and used in a Spark K-Means pipeline. K's 3 - 25 were evaluated by silhouette score and distribution of station count by group. K = 19 was selected as a balance between silhouette score (0.46) and an even distribution of stations. Using K = 19, categories were further collapsed into 7 groups by assigning 32 ungrouped stations to an outlier behavior group with an id of -1. See **Appendix A: Table 1-2.**

- Below Central Park (green, group 3) has the shortest avg *trip_duration* and shortest *crowDist.*
- Groups 5 and 6 (red and dark blue) have, on average, longer trips indicating riders are coming from further away when arriving at these stations.
- The outlier group (-1) contains only stations at the edges of NYC's neighborhoods.

Figure 2-3: Geographical groupings of end stations by the ride behavior[10]



[9] *endStation* variables were chosen because 109 stations were used as end stations but never as start stations.

[10] Additional findings of behavior grouping are detailed in k-means-BikeBehavior-UnBoroughed.ipynb.

2.4 METHODS

2.4.1 Destination Station Predictor – Log Reg vs Random Forest[11]

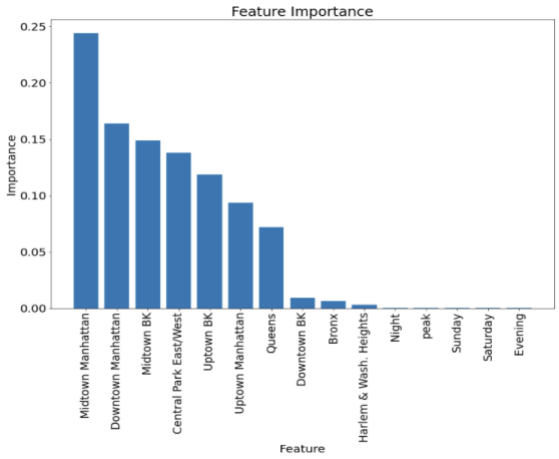
Each Citi Bike trip was a unique event starting and ending at a specific time and station. Different modeling techniques were used to predict a trip's terminal neighborhood from a combination of the starting point and variables associated with the time at which a trip began.

As mentioned in the feature-engineering section, there were over 1500 Citi Bike stations throughout NYC (and growing). It was nearly impossible to predict an end station based on any characteristics[12], and the results would have been too granular to be useful. To build a practical model, we decided to predict an end station's neighborhood. Neighborhoods were produced through the K-Means process described in our Feature Engineering section above.

Before modeling, we performed additional EDA to view the distribution of rides between neighborhoods; approximately 63.3% of rides started and ended in the same neighborhood.

We ran Random Forest (RF) and Logistic Regression (with 3-fold cross validation to tune parameters; more folds proved to be too computationally expensive) models to classify ending neighborhood based on feature variables *startStation*, *dow*, time-of-day bin, *peak/non-peak* commute flag, and *month*. Distinct testing and training splits were used in the modeling process. **Appendix A: Table 1-3** contains model performance stats, including precision and true-positive rates by *endStation* variables**.**

Figure 2-4: Importance by Feature Var., Random Forest. Non-neighborhood features have negligible importance (left); correct predictions from RF model - % Total Predictions v % Total Actual Data (right)



start_neighborhood	end_neighborhood	% Total	% Tot Act
Midtown Manhattan	Midtown Manhattan	20.61	20.61
Downtown Manhattan	Downtown Manhattan	11.7	11.7
Central Park East...	Central Park East...	9.9	9.91
Midtown BK	Midtown BK	7.76	7.76
Uptown BK	Uptown BK	5.54	5.55
Uptown Manhattan	Uptown Manhattan	3.99	3.98
Queens	Queens	2.86	2.86
Downtown BK	Midtown BK	0.53	0.53
Bronx	Bronx	0.29	0.43
Harlem & Wash. He...	Uptown Manhattan	0.02	0.1
Harlem & Wash. He...	Central Park East...	0.0	0.04
Harlem & Wash. He...	Midtown Manhattan	0.01	0.01
Bronx	Midtown Manhattan	0.0	0.01

The results of the models were similar. Each model struggled to predict end neighborhood if it differed from the starting neighborhood, likely due to how imbalanced the data were and the complexities inherent in multi-category prediction with numerous classes. However, the RF model was able to predict more inter-neighborhood rides than the logistic regression model with a slightly higher accuracy of 63.2% vs 62.9%. Both nearly identical to the intra-neighborhood percentage seen in the full dataset because the models predict these rides well but almost never identified inter-neighborhood rides.

We believe that most rides started and ended in the same neighborhood because of NYC's transportation infrastructure. NYC is a walkable city with many other transportation options, including nearly 500 subway stations. We posit that most riders are using multiple modes of transportation during any given trip, e.g., biking to a subway station in their starting neighborhood, taking a train to a new neighborhood, then biking from the terminal subway station to their destination in the new neighborhood.

These characteristics made prediction difficult, as seen in the model results, which typically predict that a ride will end in the starting neighborhood regardless of additional information.

[11] Jupyter Notebooks: RF_neighborhood_pred, log_reg_neighborhood_pred, LR_RF_model_analysis.ipynb

[12] With rider-specific information, which Citi Bike has access to, end station prediction may be possible.

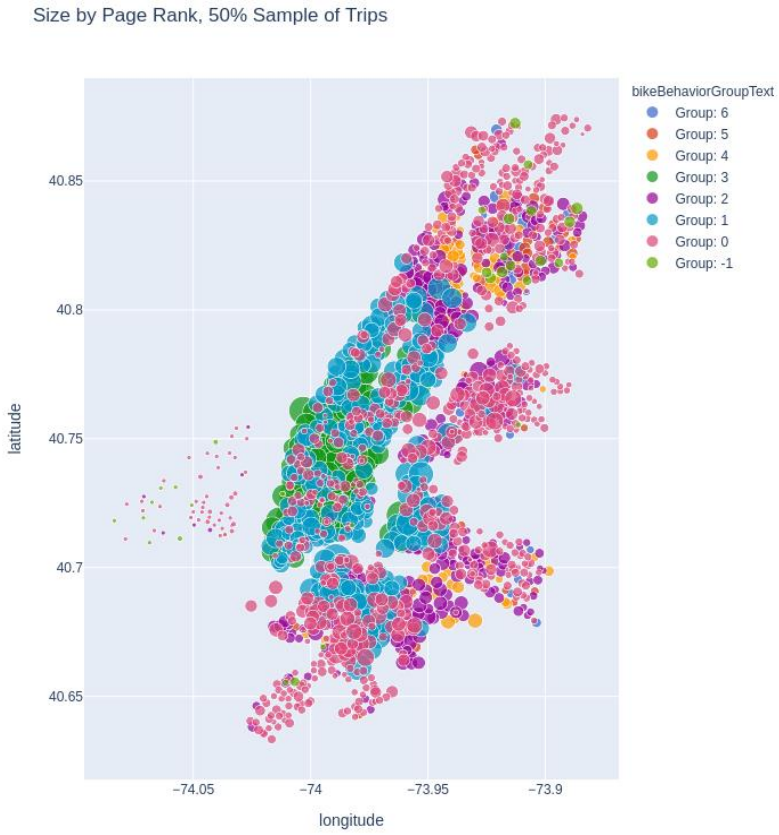
In general, Citi Bike data were a poor proxy for commuting data since the data was blind to intermediate commuting steps. It would have been far more interesting to predict a rider's end neighborhood given their *startStation* if we were able to train our models on data that included these intermediate commuting steps.

2.4.2 Graph Neural Network: Exploration of Station Rank

Rank quantified a station's criticality for the flow of bikes through NYC. A high-ranking station without enough bikes to rent or without adequate docking for bike return would impede the efficiency of Citi Bike's distribution system. When rental and return availability are not adequately balanced by riders moving bikes, Citi Bike "rebalances" by manually moving bikes between stations.[13] Knowledge of how different conditions alter optimal rebalancing strategies is of interest to Citi Bike.

A graph neural network (GNN) was suited to quantify the importance/rank of a station using the PageRank algorithm.[14] The figure below plotted end stations with size relative to rank and color by bike behavior group. Bike behavior groups 1 and 3 are clearly the most important for the efficient distribution of bikes throughout the stations. Unsurprisingly, central stations are high ranking.

Figure 2-5: Citi Bike stations: color by bike behavior group, sized by station rank.



Throughout the analysis, rank > 3 was selected as the threshold for a station to be considered a "most important" station. **Appendix A: Figure 1-3: Station rank distribution** supports the threshold.

[13] An example of the importance of rebalancing is that, in our data set, 109 Citi Bike stations were only used as end stations, never a starting station.

[14] From Wikipedia: "PageRank works by counting the number and quality of links to a page to determine a rough estimate of how important the website is. The underlying assumption is that more important websites are likely to receive more links from other websites "Facts about Google and Competition". Archived from the original on 4 November 2011. Retrieved 12 July 2014.

Analysis of station rank by weather conditions for a trip is described below, and **Appendix B: Station Rank and COVID-19** outlines additional interesting analysis of pre-COVID vs. COVID station rank.

2.4.3 Citi Bike Stations and Subway Stations

We hypothesized that the highest ranked stations would be associated with subway stations, especially outside of lower Manhattan. However, referencing the figure below, it did not appear to be true, except for Midtown, when looking at a zoomed-in plot that includes subway stations.

Figure 2-6: Size by station rank, including subway stations

2.4.4 Most Important Stations Good vs. Bad Weather

To facilitate Citi Bike's process for effective rebalancing based on weather conditions, we investigated rank for stations during good and bad weather. This analysis was based on a 50% sample of our Citi Bike data consisting of 17,043,596 good weather trips, and 7,739,887 bad weather trips. "Most Important" stations have a rank > 3.

For *weatherGoodDurationIndex* trips, 45 stations had a page rank > 3. 11 of these stations were uniquely high-ranking stations for good weather trips, which were located further from the center of NYC compared to bad weather. The max page rank for stations during good weather was 5.73. The top five good weather stations by rank were:

1. Front St & Washington St
2. 1 Ave & E 68 St
3. E 17 St & Broadway
4. West St & Chambers St
5. 134 Kent Ave & N 7 St

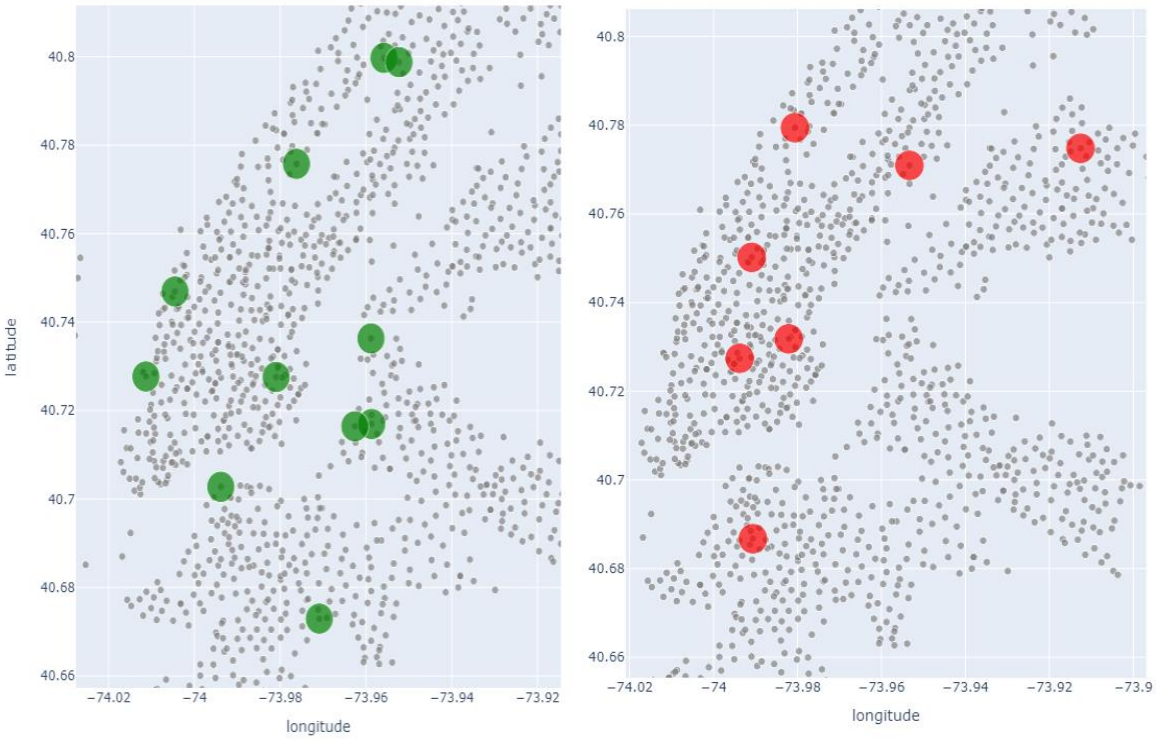
For *weatherBadDurationIndex* weather trips, 41 stations had a page rank > 3. 7 of these were uniquely high-ranked stations for bad weather trips. Unlike highly ranked good weather stations, no stations unique to bad weather trips were located on the border of Brooklyn and Queens. Generally, stations ranked highly for bad weather trips were located closer to the center of NYC than good weather stations. The max rank for bad weather stations was 5.498. The top five bad weather stations by rank were[15]:

1. 1 Ave & E 68 St
2. Front St & Washington St

[15] Analysis of the top 5 uniquely Most Important for bad weather is in Appendix D.

3. Pershing Square North
4. E 17 St & Broadway
5. W 21 St & 6 Ave

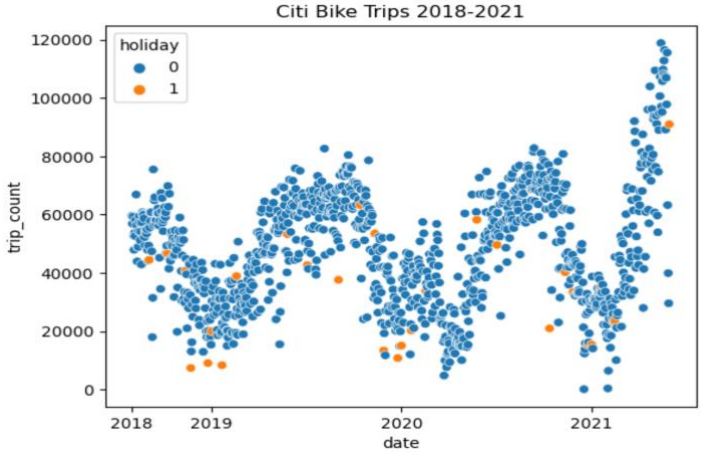
Figure 2-7: Rank > 3 stations unique to good weather trips (left); Rank > 3 stations unique to bad weather trips (right)



2.4.5 US Federal Holiday Prediction – Logistic Regression vs. Random Forest

A benchmark and competitor model were created to see if we could accurately predict holidays based on Citi Bike data. We initially hypothesized that riders rode less on holidays. Subsequent data exploration revealed this to be true, and our data for each model were grouped by daily rides.

Figure 2-8: Number of Citi Bike trips by date

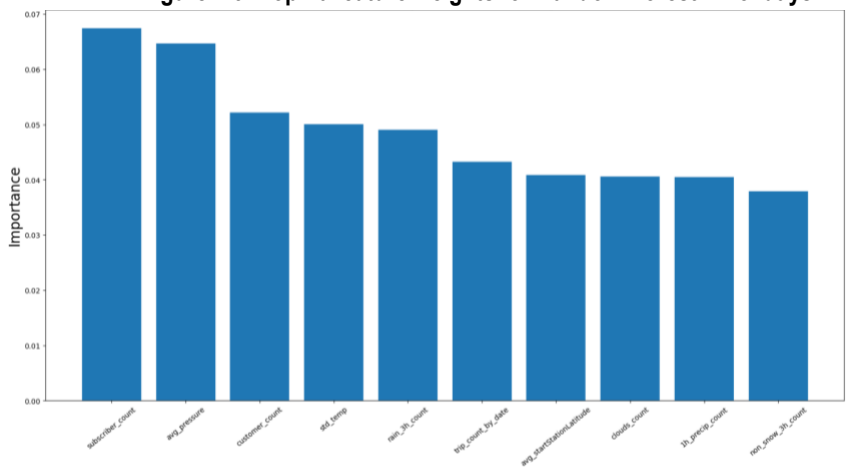


Using Citi Bike data to accurately predict a holiday, combined with our previous finding that fewer trips occur during/around the holidays, had potential to help Citi Bike rebalance bikes around these days.

The 10 features for the Logistic Regression model were chosen to capture the diversity of our dataset and included *usertype*, number of trips, *avg_median_sales_price*, *crowDist* of the Citi Bike trip, *trip_duration*, *time_bin*, whether it was a *peak* trip, whether there was *precip*, *wind_speed*, *humidity*, and what the weather *feels_like*. We aggregated on counts for the categorical predictors and both averages and standard deviations for the numeric predictors, culminating to 22 total predictors. Features were scaled using StandardScaler. After down sampling to control for overfitting, tuning consisted of 10-fold cross validation with ParamGridBuilder using a BinaryClassificationEvaluator. Ridge Regression parameters were preferred with a regularization parameter or 0.01 and a model iteration of 5. Model accuracy and AUROC were both 0.727.

A similar process was performed via RF. All features were used (besides time and *zipcodes*) and scaled using StandardScaler. After aggregation, features totaled 63. Tuning consisted of 5-fold cross validation with ParamGridBuilder using BinaryClassificationEvaluator to evaluate model performance. The optimal metrics of this model contained several trees equal to 30, and a maximum tree depth of 15. RF was slightly more predictive than the base model with AUROC of 0.707 and a model accuracy of 0.773. The top 10 features were extracted and visualized, shown below.

Figure 2-9: Top 10 feature weights for Random Forest – Holidays



Additionally, we conducted analysis on the count of trips per day. We regressed the variable *count* against the most important weather variables from the analysis above as well as additional weather variables, to see the discrepancy between the models. While the results of the Linear Regression (tuned with 10-fold cross validation) came out with a low R^2value of 0.44, the model did show what we had expected, even with low accuracy. The *avg_rain_3h and avg_feels_like* variables were the most important weather events that predicted the number of bike trips taken each day; precipitation and temperature held high feature weights in the holiday classification model.

2.4.6 Housing Sale Price Prediction – Linear Regression vs. Random Forest

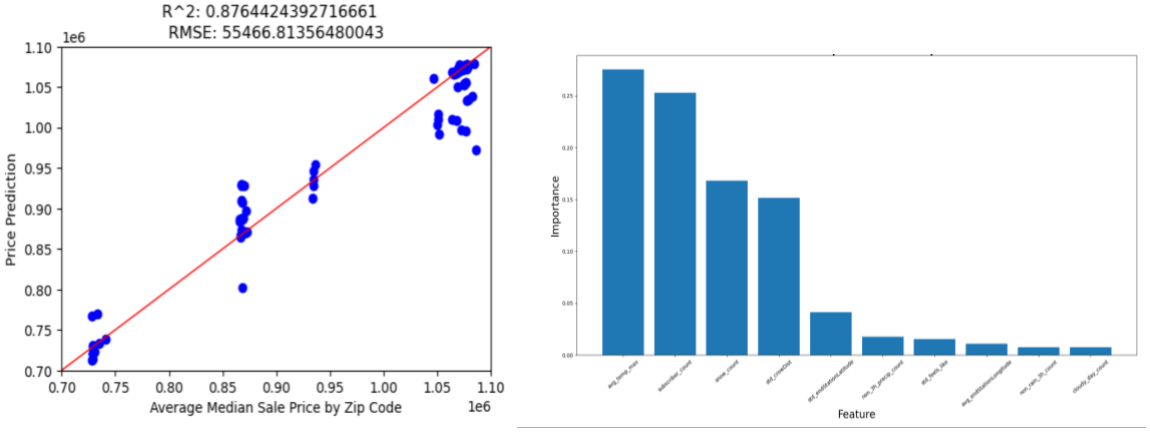
To predict median housing sales price in the NYC area, we modeled both a base Linear Regression model (consisting of 9 selected predictors and 1 engineered predictor) and a challenger RF model (consisting of all features). The data was grouped by *zipcodes* because the housing data were represented by borough zip. For both models, categorical variables were aggregated based on the frequency of occurrence per zip code, while numeric features were aggregated using averages and standard deviations. Features were scaled using StandardScaler.

For the base model, 10 features were selected because they best represented the diversity of the dataset. These features included ratio of sales to rent inventory (which was a featured engineered variable), *crowDist*, *feels_like* temperature, *humidity*, *wind_speed*, trip_*duration*, presence of clouds, *peak/non-peak* trip, *precip / no_precip* during trip, and time of day. After aggregating, we had 23 predictors. Tuning consisted of 10-fold cross validation with ParamGridBuilder using RegressionEvaluator to evaluate the model performance. Lasso Regression was preferred, removing the count of the non-precip days. Number

of trips and the standard deviation of the *crowDis*t had the highest coefficients. The optimal linear regression model had a R^2 of 0.537, a regularization parameter of 0.5 and a model iteration of 10.

For RF, all features were used (besides time and date) and scaled using StandardScaler, then aggregated to a total of 63 variables. Tuning consisted of 5-fold cross validation with ParamGridBuilder using RegressionEvaluator to evaluate model performance. The optimal RF regression model contained several trees of 10 and a max depth of 25. The R^2 of the model was 0.876, an improvement over our base model.

Figure 2-10: Price Prediction vs Actual Sale Price (left); Top 10 Feature Importance for RF - Price by zip



Looking at RMSE, the prediction of the average sales price by the RF regression model is off by $55,467. The standard deviation of the average sales price data is around $159,400; our RF RMSE is about ⅓ this value, which indicates that, while there is room for improvement, we considered it a good model. Although more data points exist at the greater house values, the figures above shows that the RF model is more accurate at predicting lower median housing prices.

Feature weights were extracted and visualized to reveal the top 10 most important features for the RF Regression Model (**Figure 2-11**). The average maximum temperature, subscriber count, snow count, and standard deviation of crow distance were the strongest predictors. While difficult to interpret, there appears to be underlying predictiveness of the housing sales price with higher temperatures and increased Citi Bike subscribers[16]. This shows Citi Bike can use real estate data as additional indicators when determining where to place stations as they contain latent factors describing riders and their behaviors.

2.4.7 Distance Predictor – Random Forest Vs. Gradient Boosted Trees Vs. Linear Regression

Finally, we wanted to see if we could predict the distance travelled by a biker given the available data. We compared Linear Regression, RF, and GBT.

The predicted variable was *crowDist*. The categorical data were indexed and one-hot encoded. Upon transformation, feature count increased from 19 to 180 features, primarily due to the OHE expanded *zipcodes* feature. While this increased the cardinality of our dataset, we wanted to increase the predictive capacity of our models by including *zipcodes* and seeing if they introduced uniquely outsized influence on predicted distance. In the models, Spark pipelines were constructed to leverage the efficiencies allowed by the Spark environment. See features in **Appendix A: Table 1-4.**

The models all performed similarly, with a RMSE ranging from 0.94 to 0.96. The models did not perform well in its overall explanatory power of the variability in distance travel. All three models had a measured R^2 under 0.19. This is somewhat expected when thinking through what variables you would want captured when trying to predict how far someone would be willing to travel on a bike. We have almost no data associated with the user themself; Is the user a triathlete? Does the user bike every day? What are some other vital health statistics of the user?

[16] Indicating you should sell your home in the warmer weather months.

The Citi Bike data did provide some takeaways when analyzing distance travelled. We found the most important features from the GBT and RF models to be night, temperature, and the year (*2020),* Subscriber vs. Customer, respectively. **See Appendix A: Tables 1-4 & 1-5.**

3 RESULTS AND CONCLUSIONS

- There are regional differences in bike behavior in NYC that help contextualize city-wide cycling-as-transportation. Understanding regional and conditional variations in behavior can improve Citi Bike operations and inform strategies to increase bicycle use.
- Our dataset was insufficient for predicting trip start and end neighborhoods. Both the RF champion and Logistic Regression challenger models fail to predict inter-neighborhood rides accurately. However, most rides start and end in the same neighborhoods, so inter-neighborhood rides are anomalous - rides from Neighborhood A to B never represent more than about 6% of total rides in the full dataset for A ≠ B. Though we hoped these data could be used to predict start-to-finish commutes, our data were insufficient; we posit that most riders are biking intermediate point-to-point trips rather than directly to their destination. If the data tracked person behavior across ***all*** public transportation rather than bike behavior, the data would be better balanced and likely better for prediction.
- *year* is the most important feature for predicted distance. Each of the three contending models in our distance study identified the year 2020 as a significant predictor. Users travelled roughly 0.09 miles further in 2020. This was explained by the pandemic's notable effect on transportation (i.e., a disinclination to ride public transportation during a pandemic would influence a user to bike a longer distance). *user-type* is also a notable feature in predicting distance. Subscribers travelled 0.07 miles less than one-time users of Citi Bike, and attraction-based stations have longer distance predictions (e.g., Central Park and The Bronx Zoo). Unsurprisingly, *temperature* has a positive impact on distance travelled.
- Station rank is affected by *GOOD* vs. *BAD* weather. Specifically, stations closer to the center of lower Manhattan are more important for station balancing during bad weather. Additionally, *precip* and *feels_like* are the most important variables affecting the count of bike trips by day.
- Prior to down sampling, holidays represented only ~2% of the data, indicating possible overfitting issues and contributing to model difficulty due to lack of data. After down sampling, both the base and challenger model favor features relating to temperature, *subscriber count*, *trips per day*, distance traveled, and precipitation. Users travel less on holidays, and temperature is important for determining holidays. Understanding characteristics of holiday bike trips can help Citi Bike with bike rebalancing initiatives.
- The predictiveness of the RF regressor model is high with a R^2 value of 0.876 and RMSE that was roughly ⅓ of the standard deviation of the average housing sales price. Warmer temperatures, *subscriber count, sd. of crow-flies distance,* and end-station statistics are the most important feature weights. The model is more accurate at lower sales prices. Citi Bike can use this analysis to determine where to place additional stations as they contain latent factors describing riders and their behaviors.

3.1 SUMMARY

We gained notable insights into user behavior through our analyses. While this can help Citi Bike's decisioning, we recognize that further data collection (namely user-specific and intermediary-transport data) will offer high return when predicting user behavior. User-specific analysis will improve goodness-of-fit and complement insights gleaned from available data, particularly if Lyft can cross-sell car rides through the Lyft app with Citi Bike rides. With more data, we would begin with user-specific analysis to tie the whole project together.