# Forecasting the CitiBike Ridership in New York City

Dan Mao | Eva Yao | Siyong Liu | Xinlu Xu | Yu Wu | Ziheng Gong

CUSP, Tandon School of Engineering

Applied Data Science

CUSP-GX 6001 Spring 2022

Instructors: Stanislav Sobolevsky

3 May 2022

## Abstract

As bike-sharing systems are widely distributed in cities, system operators need to provide good management to ensure a balanced distribution of bicycles in cities. New York City's Citi Bike is the largest bike-sharing system in the US. Predicting the number of Citi Bikes from one station to another can help understand the transition model of bikes and then optimize bike storage. In this study, we adopt a new type of data source, event record, to analyze and address flow detection problems. Other factors that are driving Citi Bike demand, such as time, meteorology, and socio-economic statistics, are combined with the event records. We explore bike-sharing data using time series analysis and network analysis, and construct a gradient boosting tree model to better predict the number of trips and see how events influence trips.

## Introduction

Public bike-sharing systems are becoming an increasingly substantial element of transportation networks both in the United States and around the world. As the bike-sharing network expenses, the ability to predict the number of hourly users can allow the planner to manage the system in an efficient and cost-effective manner. The purpose of this research is to develop an accurate prediction model that estimates the demand for bike-sharing ridership. With this information, planners can give an accurate estimation of ridership, and therefore optimize the allocation of bike resources system-wide.

In the past years, several studies identified a variety of relationships between demographic, temporal, and weather variables and bike-sharing ridership. Following the trajectory of previous works, we select Citi Bike ridership records in New York City that happened in 2021 as our scope of the study. In the first part of our research, we explore and visualize the bike-sharing dataset in the form of network analysis and time-series analysis. In the second part, we estimate the hourly ridership at a station level using the decision tree, random forest, and gradient boosting tree model. We perform model selection based on the models' accuracy score. The gradient boosting tree model outperforms others and sufficiently predicts NYC bike-sharing demand on a station level with a 0.72 R-square score.

### Literature Review

The deployment of Citi Bike has changed the urban transportation landscape in New York City, and many other major cities around the world have already adopted similar bike-sharing systems. In recent years, there has also been a growing body of research on bike-sharing, such as analysis and forecasting of bike-sharing flow or usage.

Bike stations located in the same urban area usually have similar traffic patterns. The researchers proposed cluster-level prediction models to predict the total bike use in the clusters. Y. Li, Y.Zheng, H.Zhang, and L.Chen[1] proposed a two-part clustering algorithm, and then combined the GBRT model with a diversity-based reasoning model to predict the bike demand in each cluster. In [2], they proposed an adaptive transition constrained clustering algorithm (ATC). Both clustering algorithms can obtain clusters with more regular concessions and transfers. Similarly, X. Zhou[3] created a hierarchical clustering method using a community detection algorithm.

Unlike the static clustering model mentioned above, L. Chen, D. Zhang, L. Wang, D. Yang, X. Ma, S. Li, Z. Wu, G. Pan, T.M.T. Nguyen, and J. Jakubowicz[4] dynamically classifies adjacent sites into clusters according to context, including common context factors (such as weather and climate) and opportunistic context factors (such as social and traffic events). Although the cluster level prediction model has relatively high prediction accuracy, the prediction results can not directly guide the system manager to redistribute bicycles to solve the imbalance problem. Therefore, many researchers develop prediction models from the site level.

Station-level bike projection is more practical. Bike use at stations is usually treated as timeline data and then a linear regression model is built to predict it[5, 6]. In addition, other traditional statistical methods such as Kalman filtering and the ARIMA model are also widely used[7, 8]. To avoid the inequity and uncertainty that the traditional timeline model cannot reflect the change in bicycle flow, a research method based on machine learning is proposed to automatically learn the statistical legal relationship from the bicycle traffic data. For example, the SVR model was used in conjunction with random forest (RF) in [9], but [10] was developed on a projection algorithm based on a Bayesian network. In recent years, deep neural networks have been widely used in traffic prediction. Neurons such as CNN and RNA were also used to predict bike usage[11, 12].

Despite the good predictive performance, existing deep neural networks (e.g., RNA) are unable to sense the spatial dependence of flow models. To predict bicycle flow, the above method cannot cover the spatial correlation of bike flow, because it ignores the topology of the bike station network. To address these problems, traffic forecasting began using graphics-based methods[13, 14], which continue to be applied to traffic forecasting[15, 16, 17]. Graph rotation operations are very efficient for acyclic data, and researchers define rotation operators from the vertex domain[18, 19] and spectral domain[20, 21]. Detailed information on the development network (GCN) schedule can be found in [22]. By combining GCN and RNN, the researchers were able to solve the problem of spatial time and space dependence. However, there is a graphic design problem. In general, existing works in the bike station system are defined as units and different definitions are provided for the interconnection of stations marked as edge weights. For example, in [17], they modeled site relevance based on the geographical distance between sites. And D. Chai, L. Wang, and Q. Yang offered three alternative ways of building inter-station graphs: distance graph, interaction graph, and bike usage correlation graph. L. Lin, Z. He, and S. Peeta propose four more typical data matrices to quantify the correlation between stations, namely spatial distance, bike demand, average trip duration, and bike demand correlation. It is also important to solve the problem of graph generation in this work. Different from existing projects, in the existing engineering, a bike station can be assigned for the node, and use the actual relationship, such as geographic distance to measure edge, in our work, using standing between traffic on its own as a node, they cannot be connected to the physical network, such as road networks, or the standing relationship, make it more difficult to pattern formation.

### Data Resource and Processing

Our primary data sources are NYC Permitted event information[23], Citi Bike Trip records[24], NOAA daily temperature[25], and US Census Bureau socio-economic data[26] from January 2021 to December 2021. Besides, our study will use Google Place Searching API[27] to convert event location from street name to precise longitude and latitude.

First, the study obtains bike usage statistics from the Citi bike official website using request API. The dataset contains the start station id, end station id, station, station longitude, and trip time for each bike trip. Regarding the volume of raw data, the study will use the Dask package to assist in data exploration. At the beginning of January, 890 Citi Bike stations had one or more originating bike trips. The number grew to around 1400 by December 2021. The study only considers the initial 890 stations as valid data points to maintain consistency of the research scope.

Then the study collects all the events that start and end within 2021 from NYC Data Portal. Notice that the dataset does not come along with precise geolocation. Therefore, we use Google Place API to retrieve the longitude and latitude for each event. In this process, we used a multiprocessing module in python, which reduced the requesting time from 7 hours to 30 minutes. In the study, we defined all the Citi Bike stations that are within 150 meters of the event location as event-affiliated stations. We check the trip records that end at these stations if their trip time overlaps with the time when events start and merge the corresponding event information with trip records.

Finally, we aggregate the raw trip records by four values, end station name, event type, trip date, and trip hour. We calculate the number of trip counts. Then we obtain socioeconomic factors from the US Census Bureau and daily from NOAA and merge these features with aggregated records. Sample data and a description of our dataset are attached in Appendix A.

## Methodology

### TimeSeries Analysis

There are five key topics in time series analysis. The first is trend and seasonality analysis. This analysis is aiming to analyze if our dataset has a consistent upward or downward trend, or if it has periodic behavior with seasonal variations. The second topic is hypothesis testing, in this part, we want to test whether the trend is statistically significant. What’s more, forecasting is an important topic in time series analysis. Based on the past performance of the time series dataset, we plan to predict future observations and model time-adjacent observations. The fourth part is simulation modeling: based on our knowledge of the time-series patterns and the process behind them, estimate the probability of certain outcomes through multiple simulations of the process. Last but not least, in the time series analysis, we try to control our dataset to remain stable over time.

### Network Analysis

Network analysis is essential in data exploration. We can analyze our network from two aspects. The first part is getting an understanding of how each node can be affected by spatial distance. There are four important indicators: degree centrality, closeness centrality, betweenness centrality, and Pagerank centrality. Degree centrality helps us to understand how important the node is for the internal network topology. Closeness centrality measures how close is the node to all other nodes in network topology. Betweenness centrality suggests how many bridges the node has, in other words, if this node fails, how many other routes will get disrupted. Pagerank centrality represents the chances for the network random walker with teleport to be seen in the node at a random moment.

Another part focuses on the structural heterogeneity of the networks, which is named community detection. In this part, we use edge weights to compare with the average expected weight to analyze whether a set of nodes are strongly connected internally compared with external connections.

### Predictive Model

In our project, we used three types of tree models: decision tree, random forest, and gradient boosting to analyze our usage count of the Citi Bike.

The decision tree model is commonly used in data mining which is used to predict the value of a target variable based on input variables. This model is a classifier as well. Due to the simplicity of the decision tree, it has several serious disadvantages, such as overfitting, bias error, and variance error. Then we introduce the random forest model to fix these problems. When random forests as a classifier, it operates by constructing a multitude of decision trees at training time to do classification tasks, the output is the class selected by most trees. When it does regression tasks, the results are the mean or average predictions of the individual trees. Therefore, the random forest model generates a large number of decision trees to reduce the variance, and improve the accuracy of prediction.Gradient boosting is another powerful ensemble model but has differences from the random forest. Gradient boosting builds one tree at a time that learns from the previous iteration and combines results along the way while the random forest combines results at the end of the process.

## Data Analysis

### TimeSeries Analysis

In this part, we implement the method of time series analysis on our Citi Bike Ridership dataset. According to the characteristics of our dataset, we choose to implement control stability of the dataset, hypothesis testing, trend and seasonality analysis, and forecasting on our data.

After analyzing the data, we selected three factors: average temperature, the population of 16 years and over with earnings, and median earnings. We use these factors with Citi Bike ridership to conduct a time-series data analysis.

Figure 1 shows the ratio of the daily number of usage counts with the three factors. We also plotted the relationship line between the total number of daily usage counts over time as a reference. The green line represents how the number of total usage changes daily, the purple line represents the ratio of the number of users to the average temperature, the red line is the ratio of the number of daily usage to the population of 16 years and over with earnings, and the yellow line is the ratio of the number of users to the median earnings.

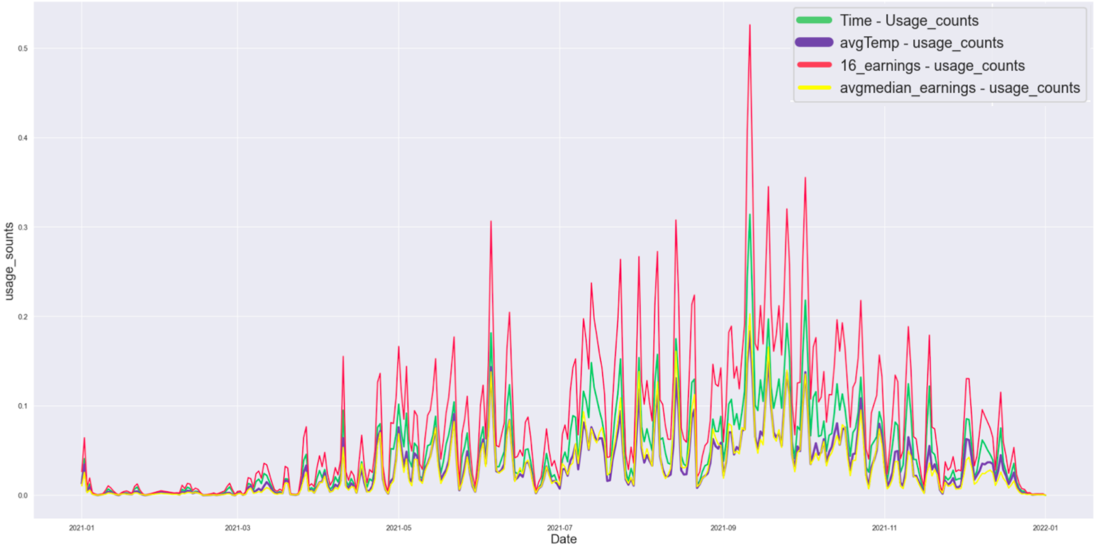


**Figure 1** Citi Bike Ridership Daily Ratios

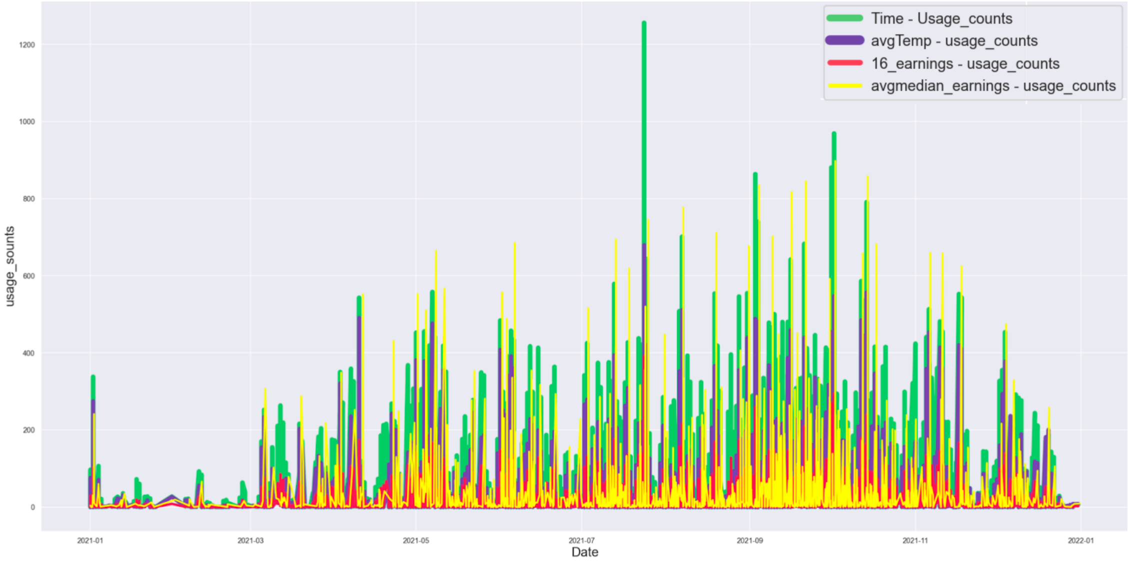


**Figure 2** Citi Bike Ridership Hourly Ratios

Figure 2 shows a more frequent fluctuation in the number of usages in hours. It can be seen from the time-series images that average temperature, average earnings over 16 years, and median earnings have similar effects on the number of users. We can also have a conclusion from two figures that environmental factors have little influ16ence on the number of users. Therefore, we can take a look at how the regional distribution will influence the usage counts.

Figure 3 is the forecast chart of the total number of daily usage counts. Here, we focus on the influence of time development on the number of users. The blue line is the trend of the training set, and the green line is the change in the number of users in the test set. Both lines are original data without model fitting. The yellow line is the training set's result after fitting an ARIMA model, and the red line is the prediction we made with the test set in the trained model. Here we use the ARIMA(14,2,14) model. It can be concluded from the model and Figure 3 that our data does not perform well in the time series model and the accuracy of prediction is low. In other words, the influence of time on the number of usage counts  is not so important. Therefore, we will conduct regional distribution research on Citi Bike data in the next part.

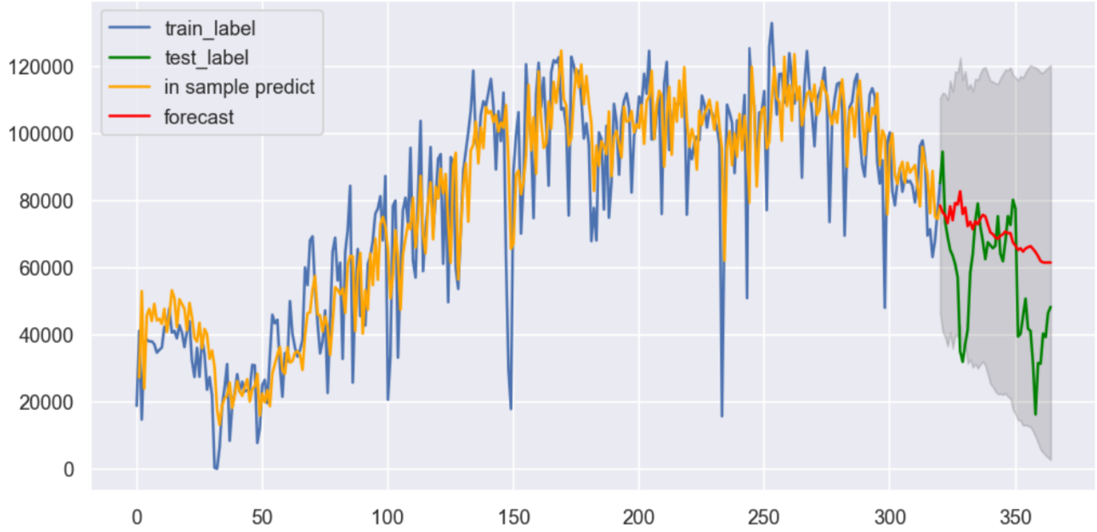


**Figure 3** Citi Bike Ridership Daily Usage Counts Forecasting

### Network Analysis

In this part, we focus on how regional distribution influences the usage count by the event days in the Citi Bike network. We analyzed four centralities and made a table to show our results. We also draw two plots to show the networks of Citi Bike and the results of community detection to get a further understanding of how regionality can affect the usage count of Citi Bike by events.

From table 1, we select the top 5 stations in each centrality and it is obvious that stations at 6th Ave & W 33rd St and Front St & Washington St are the most important stations. Reflected in the NYC map, we observe that these two stations are located near the Empire State Building and

Dumbo-Manhattan Bridge View, which are the most famous tourist attractions. And other stations are mostly located near the Empire State Building.

**Table 1** Top 5 Stations of Four Centralities

| rank | Degree Centrality | Closeness Centrality | Betweenness Centrality | PageRank Centrality |
|---|---|---|---|---|
| 1 | 6th Ave & W 33rd St | 6th Ave & W 33rd St | Front St & Washington St | 6th Ave & W 33rd St |
| 2 | Broadway & W 29th St | Front St & Washington St | Webster Ave & E Fordham Rd | Front St & Washington St |
| 3 | 6th Ave & W 21st St | 6th Ave & W 21st St | Melrose Ave & E 150th St | Melrose Ave & E 150th St |
| 4 | 6th Ave & W 34th St | Broadway & W 36th St | 6th Ave & W 33rd St | 6th Ave & W 34th St |
| 5 | Front St & Washington St | 6th Ave & W 34th St | Nicholas Ave & W 126th St | 6th Ave & W 21st St |

From figure 4 since our data is captured by station level, we take stations from the dataset as the nodes, then generate edges from each unique pair of usage counts. What's more, we define the weight of stations by the number of their edges. In figure 4, stations in Manhattan and North Brooklyn have a higher volume of usage and a higher degree than other parts of NYC. These stations are expected to need more dispatch of bikes during event days. We will do further analysis in the next part.

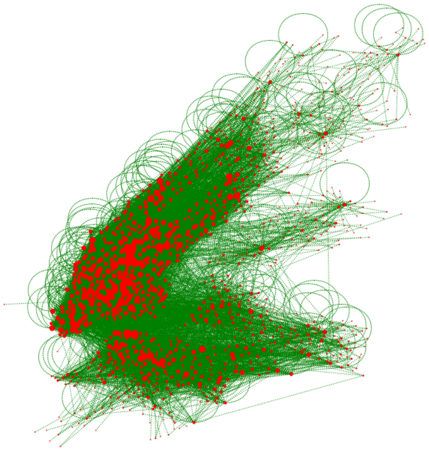


**Figure 4** Citi Bike Network

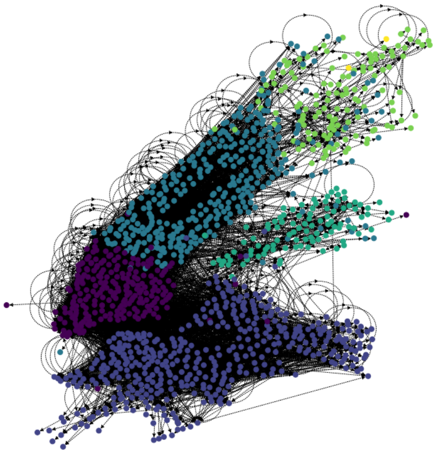


**Figure 5** Citi Bike Network with Partitioning

Figure 5 is the community partition result of Citi Bike usage counts by event days, in this plot, we set the max_communities parameter to be zero to allow the flexible number of communities. From this plot, we could also get some information based on the existing boroughs, Brooklyn, Queens, and the Bronx show distinct communities with usage counts as a whole, whereas Manhattan is divided into two communities: upper Manhattan and lower Manhattan. Stations in the same community tend to be influenced by the same events because they might have similar routes. This partitioning helps us to identify the stations that also need extra care when an event happens in the community.

**Predictive Model**

From community detection, we learn that the usage counts of Citi Bike are more concentrated in Manhattan and north of Brooklyn. Therefore, in this part, we use the decision tree, random forest, and gradient boosting to classify the Citi Bike usage counts with seven factors including trip hour, delta time, distance, daily temperature, population over 16 years with earnings, median earnings, and median ages. The parameters defined in our model and the R2 scores we finally reached for each model showed in table 2. What's more, the plot of the full decision tree and the plot of the random forest model with the first 3 estimators are put in appendix B as references for our results.

The full decision tree results plot in appendix B shows one of the results of the decision tree. As we know the decision tree has a high variance and is sensitive to outliers. Therefore we evaluate the changing of the R2 score when increasing the max depth and minimum increase impurity. We show the influence of the max depth limitation in figure 6. Since the decision tree would overfit if the depth is higher than 14, we try the max depth of 14 in our research.

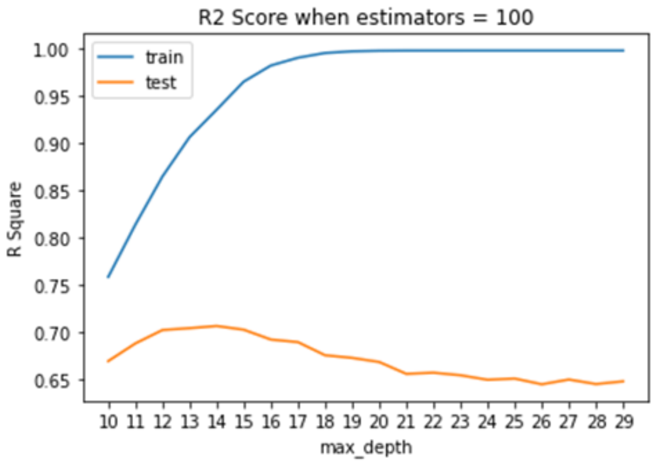


**Figure 6** Decision Tree R2 Score with Different Max Depth

As we know in theory, the random forest model will perform more reliably than the decision model. Therefore we test how the R2 score of the model will be changed when depth is increased in the random forest model. From figure 7, although the test score of R2 score is hard to improve when depth is more than 24 and impurity more than 1e-4, the model still has a high R2 score on the test dataset and did not go overfit. In this case, without manually testing the correct depth limitation, we apply a grid search to find the best max depth for the random forest model.

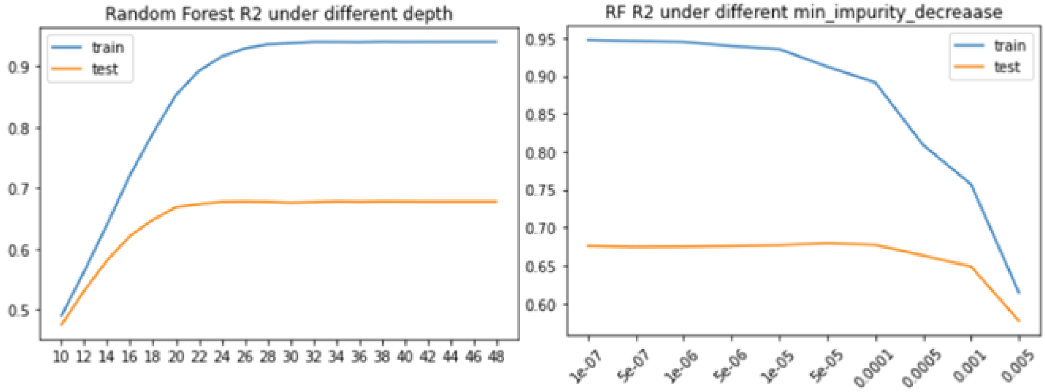


**Figure 7** R2 Score with Max Depth Increasing and Minimum Impurity Decreasing

We implement both regression and classification in the gradient boosting model. The regression model performs much better than the classification model. The R2 score of the test dataset in the regression model is around 0.72 without grid search. However, the running time of this model is significantly longer than the random forest model. Additionally, the ensemble learning method we use here is AdaBoost, which uses an exponential loss function. In order to reduce the time spent running the model, we only apply the max depth equal to 10 in this model.

From the following table 2, we could be told that the average R2 score of the three models is around 0.67. The decision tree model's R2 score is around 0.62, which is the lowest among the three models. In the random forest model, by applying cross-validation and grid search, and after running 450 fittings in this model, we got a higher R2 score of 0.69. The gradient boosting model has the highest R2 score of 0.72. Compared with the decision tree model which has a max depth of 14, and the random forest model which has a max depth of 28, the gradient boost model achieves the best performance with a lower depth.

**Table 2** Parameter Values of Three Models

| Model | Decision Tree | Random Forest | Gradient Boosting |
|---|---|---|---|
| R2 score | 0.620379 | 0.692235 | 0.720143 |
| max_depth | 14 | 28 | 10 |
| min_impurity_decrease | 1.89e-4 | 1e-05 | 1e-7 |
| n_estimators | N/A | 100 | 500 |
| min_sample_split | 2 | 2 | 2 |

Meanwhile, we also tried a two-layer neural network model as a possible method to predict the influence of events. We choose the categorical types of events, the nearest stations, trip dates, and the exact hours in a day when the events happen as predictors to fit the model. For the implementation of the neural network, we use the linear activation function and mean square error as the loss function to construct the network, and use cross-validation to validate the fitted model. In 100 epochs that we record, the mean squared error curves on both training and testing datasets are shown in Figure 8, respectively. And the accuracy curves on both training and testing datasets are shown in Figure 9. Although the accuracy of the testing dataset is generally higher than that of the training dataset, it is only around 0.30, which is quite humble. The problem can be resolved by adding more layers, which are not implemented here due to high time consumption.

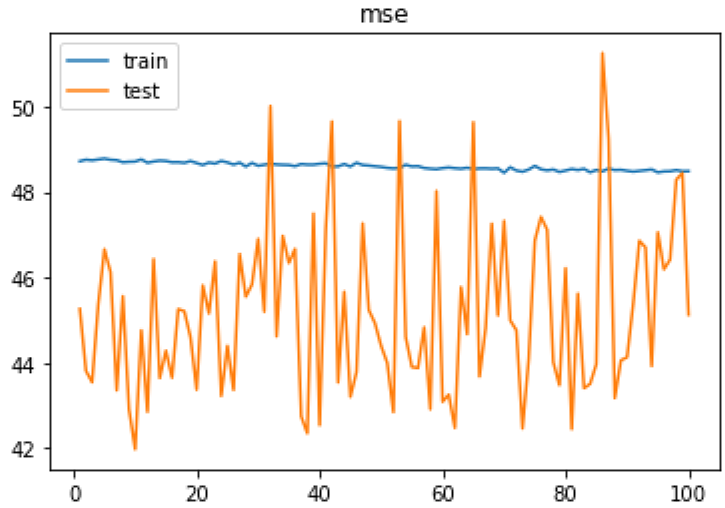


**Figure 8** Mean square error on two-layer Neural Network for 100 epochs

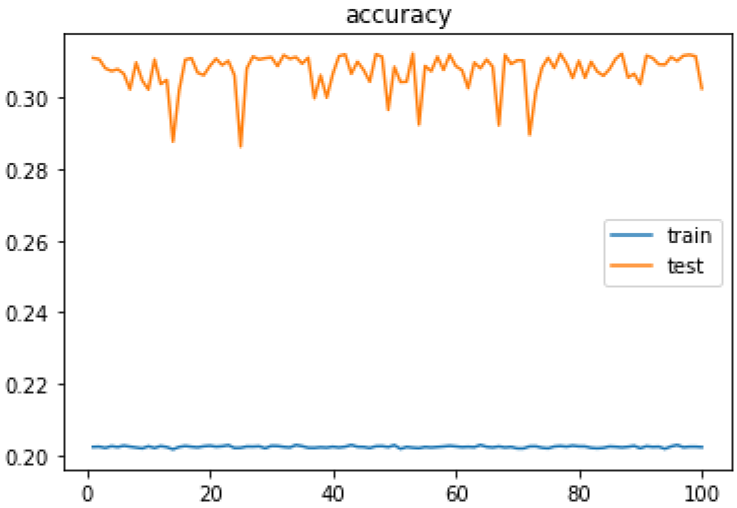


**Figure 9** Accuracy on two-layer Neural Network for 100 epochs

## Discussion

Until now, we have finished the time series analysis, network analysis, and predictive analysis of the Citi Bike dataset. The results above show that bike usage relates to the event, weather, population, and earnings. However, since time and experience are limited. We can only extract limited information from the events. For example, it is hard to evaluate the relation between event period and high usage time window because different types of events may have various delays. Second, we have room for improvement on the humble result in the neural network for this experiment. The high time complexity of such a large model also hinders fine-tuning the parameters and layers. In the next step, we plan to test it in the predictive model. Last but not least, the events' impact is specific to each mode of transportation. So we want to not only analyze the bike data but also the usage of taxis, buses, and private cars in future research.

## Conclusion

From all the analyses we have obtained so far, the number of Citi Bike usages is not highly correlated with any of the environmental factors or factors such as time, and is more likely to be caused by random human behavior.

However, what can be concluded from our research analysis is that the number of Citi Bike usages can be fitted in the time series model. It also reflects the cluster distribution of usages on a map by using the community partition model. The distribution model of this dataset can be regressed using machine learning methods or be classified into clusters. Also, the number of Citi Bike usages has seasonal fluctuation. For example, in the time series analysis, we learn that the bicycle usage count is based on a 12-hour cycle for hourly data analysis and a 7-day cycle for daily data analysis. It also changes with the seasons. The number of bicycle users in winter is significantly lower than the number of users in summer. Based on the network analysis, we know that there is a large demand for bicycles near popular tourist sites in New York. There are two major stations in the Bronx that play an important role in the overall bicycle-sharing network as well. Brooklyn, Bronx, and Queens show strong community patterns in the ridership network, while the pattern in Manhattan differentiates between the middle and lower parts. In the decision tree model, random forest model, and gradient boosting model, we find that more bike-sharing stations can be set up in some popular spots in the city to meet the demand during the daytime, which coordinates with the conclusions that are drawn in our other analyses. Therefore, for this paper, our findings are consistent.

## Reference

[1] Y. Li, Y.Zheng, H.Zhang, and L.Chen, Traffic prediction in a bike-sharing system, in SIGSPATIAL, 2015, pp.33, ACM.

[2] Y. Li and Y. Zheng, Citywide Bike Usage Prediction in a Bike-Sharing System, IEEE Trans.Knowl.Data Eng., 32(6):1079-1091, 2020.

[3] X. Zhou, Understanding Spatiotemporal Patterns of Biking Behavior by Analyzing Massive Bike Sharing Data in Chicago, PLOS ONE, 10(10), 2015.

[4] L. Chen, D. Zhang, L. Wang, D. Yang, X. Ma, S. Li, Z. Wu, G. Pan, T.M.T. Nguyen, and J. Jakubowicz, Dynamic cluster-based over-demand prediction in bike-sharing systems in UbiComp, 2016, pp. 841-852, ACM.

[5] M. Zeng, T. Yu, X. Wang, V. Su, L.T. Nguyen, and O.J. Mengshoel, Improving Demand Prediction in Bike Sharing System by Learning Global Features in KDD, 2016.

[6] Y. Feng and S. Wang, A forecast for bicycle rental demand based on random forests and multiple linear regression in ICIS, 2017, pp. 101-105, IEEE/ACIS.

[7] C. Gallop, C. Tse, and J. Zhao, A Seasonal Autoregressive Model Of Vancouver Bicycle Traffic Using Weather Variables in TRB 91st Annual Meeting, 2012.

[8] S. Lee and D.B. Fambro, Application of Sub-set Autoregressive Integrated Moving Average Model for Short-Term Freeway Traffic Volume Forecasting, Trans.Res.Record., 1678(1):179-188, 1999.

[9] Y.C. Shiao, W.H. Chung, and R.C. Chen, Using SVM and Random forest for different features selection in predicting bike rental amount in iCAST, 2018, IEEE.

[10] J.E.Froehlich, J.Neumann, and N.Oliver, Sensing and predicting the pulse of the city through shared bicycling in IJCAI, 2009, pp. 1420-1426, ACM.

[11] C.Thirumalai and R.Koppuravuri, Bike Sharing Prediction using Deep Neural Networks, JOIV., 1(3):83, 2017.

[12] P. Chen, H. Hsieh, K. Su, X.K. Sigalingging, Y. Chen and J.S. Leu, Predicting station level demand in a bike-sharing system using recurrent neural networks, IET Intell.Transp.Syst., 14(6):554-561, 2020.

[13] X. Wang, C. Chen, Y. Min, J. He, B. Yang and Y. Zhang, Efficient Metropolitan Traffic Prediction Based on Graph Recurrent Neural Network arXiv preprint arXiv:1811.00740, 2018.

[14] C. Song, Y. Lin, S. Guo and H. Wan, Spatial-Temporal Synchronous Graph Convolutional Networks: A New Framework for Spatial-Temporal Network Data Forecasting in AAAI, 2020, pp. 914-921.

[15] D. Chai, L. Wang, and Q. Yang, Bike flow prediction with multi-graph convolutional networks, in SIGSPA- TIAL, 2018, pp. 397-400, ACM.

[16] L. Lin, Z. He, and S. Peeta, Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach, Transport.Res.C-Emer., 97: 258-276,2018.

[17] R. Guo, Z. Jiang, J. Huang, J. Tao and L. Chen, BikeNet: Accurate Bike Demand Prediction Using Graph Neural Networks for Station Rebalancing, 2019.

[18] F. Monti, D. Boscaini, J. Masci, E. Rodola`, J. Svoboda and M.M. Bronstein, Geometric deep learning on graphs and manifolds using mixture model CNNs in CVPR, 2017, pp. 5115-5124.

[19] M. Niepert, M. Ahmed and K. Kutzkov, Learning Convolutional Neural Networks for Graphs in ICML, 2016, pp. 2014-2023, ACM.

[20] J. Bruna, W. Zaremba, A. Szlam and Y. LeCun, Spectral Networks and Locally Connected Networks on Graphs in ICLR, 2014.

[21] M. Defferrard, X. Bresson and P. Vandergheynst, Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering in NIPS, 2016.

[22] T.N. Kipf and M. Welling, Semi-Supervised Classification with Graph Convolutional Networks in ICLR, 2017.

[23] City Government, NYC Permitted Event Information - Historical. *NYC OPEN DATA*. Available At: https://data.cityofnewyork.us/City-Government/NYC-Permitted-Event-Information-Historical/bkfu-528j [Accessed April 20, 2022]

[24] Index of bucket "tripdata". *AmazonAWS*. Available at: https://s3.amazonaws.com/tripdata/index.html [Accessed April 20, 2022]

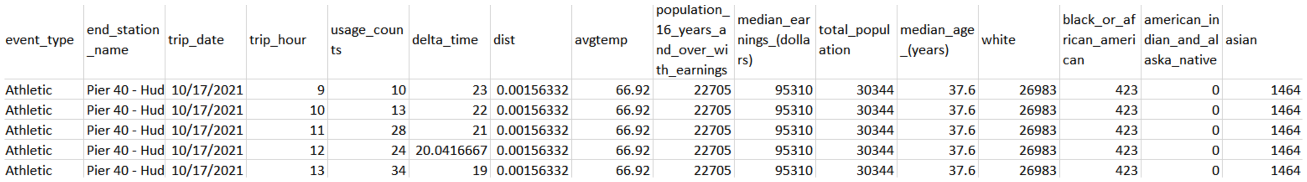
[25] Anon, ACS DEMOGRAPHIC AND HOUSING ESTIMATES. *Explore census data*. Available at: https://data.census.gov/cedsci/table?q=All+5-digit+ZIP+Code+Tabulation+Areas+fully%2Fpartially+contained+within+New+York+city%2C+New+York&tid=ACSDP5Y2019.DP05 [Accessed December 10, 2021].

[26] Anon, EARNINGS IN THE PAST 12 MONTHS (IN 2019 INFLATION-ADJUSTED DOLLARS). *Explore census data*. Available at: https://data.census.gov/cedsci/table?q=All+5-digit+ZIP+Code+Tabulation+Areas+fully%2Fpartially+contained+within+New+York+city%2C+New+York&t=Income+and+Poverty&tid=ACSST5Y2019.S2001 [Accessed December 10, 2021].

[27] Place Search. *Google Maps Platform.* Available at: https://developers.google.com/maps/documentation/places/web-service/search [Accessed April 20, 2022]

## Appendix A. Sample Data and Data Description

### Sample Data



| event_type | end_station_name | trip_date | trip_hour | usage_counts | delta_time | dist | avgtemp | population_16_years_and_over_with_earnings | median_earnings_(dollars) | total_population | median_age_(years) | white | black_or_african_american | american_indian_and_alaska_native | asian |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Athletic | Pier 40 - Hud | 10/17/2021 | 9 | 10 | 23 | 0.00156332 | 66.92 | 22705 | 95310 | 30344 | 37.6 | 26983 | 423 | 0 | 1464 |
| Athletic | Pier 40 - Hud | 10/17/2021 | 10 | 13 | 22 | 0.00156332 | 66.92 | 22705 | 95310 | 30344 | 37.6 | 26983 | 423 | 0 | 1464 |
| Athletic | Pier 40 - Hud | 10/17/2021 | 11 | 28 | 21 | 0.00156332 | 66.92 | 22705 | 95310 | 30344 | 37.6 | 26983 | 423 | 0 | 1464 |
| Athletic | Pier 40 - Hud | 10/17/2021 | 12 | 24 | 20.0416667 | 0.00156332 | 66.92 | 22705 | 95310 | 30344 | 37.6 | 26983 | 423 | 0 | 1464 |
| Athletic | Pier 40 - Hud | 10/17/2021 | 13 | 34 | 19 | 0.00156332 | 66.92 | 22705 | 95310 | 30344 | 37.6 | 26983 | 423 | 0 | 1464 |

### Data Description

| Column Name | Description |
|---|---|
| event_type | This is the type of event. |
| end_station_name | This is the end station of trips. |
| trip_date | This is the end date of the trips. |
| trip_hour | This is the end hour of trips. |
| usage_counts | This is the total count of ridership at end_station_name, trip_date, trip_hour, related to event_type. |
| delta_time | This is the time difference between the end hour of trips and the start time of events. |
| dist | This is the straight line distance between the end station and the event location. |
| avgtemp | This is the daily temperature. |
| population_16_years_and_over_with_earnings | This is the population with earnings in the zip code area where end_station_name is located. |
| median_earnings_(dollars) | This is the median_earning in the zip code area where end_station_name is located. |
| total_population | This is the total population in the zip code area where end_station_name is located. |
| median_age_(years) | This is the median age in the zip code area where end_station_name is located. |
| white | This is the white population in the zip code area where end_station_name is located. |
| black_or_african_american | This is the African American population in the zip code area where end_station_name is located. |

| | |
|---|---|
| american_indian_and_alaska_native | This is the American Indian and Alaska Native population in the zip code area where end_station_name is located. |
| asian | This is the Asian population in the zip code area where end_station_name is located. |

## Appendix B. Predictive Model Plots

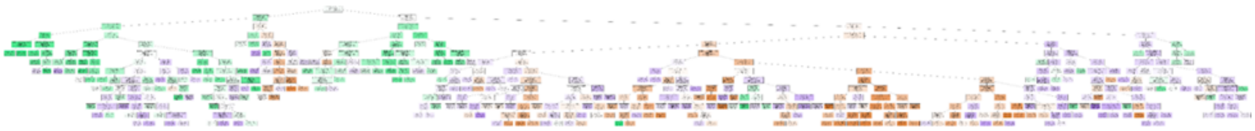


**Figure 1** The Full Decision Tree Model Classification

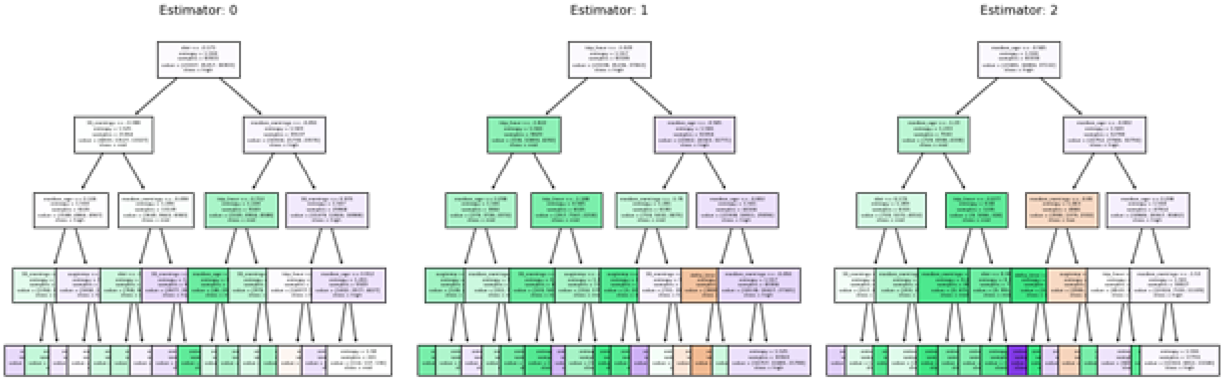


**Figure 2** Random Forest Model with the First Three Estimators