Citibike Demand Optimization

Dhruv Mojila
Stevens Institute of Technology
Hoboken, United States
dmojila@stevens.edu

Vrushali Khatane
Stevens Institute of Technology
Hoboken, United States
vkhatane@stevens.edu

Sreram Vasudev
Stevens Institute of Technology
Hoboken, United States
svasudev@stevens.edu

Vidhi Patel
Stevens Institute of Technology
Hoboken, United States
vpatel40@stevens.edu

***Abstract*— This project addresses the challenges of bike availability in New York City's CitiBike system, where users frequently encounter issues with docking and accessing bikes at various stations. The core objective is to optimize CitiBike demand across different docks to ensure a more reliable and efficient service. By developing a predictive model, this project aims to forecast bike availability dynamically, allowing for strategic bike redistributions and improved station management. This approach seeks to enhance user experience by minimizing the inconvenience of empty or full stations, thereby facilitating smoother transitions for users requesting bikes.**

I. INTRODUCTION

The CitiBike program, launched in New York City, represents a significant advancement in urban mobility as the largest bike-share system in the United States. Offering thousands of bicycles across hundreds of strategically placed stations, CitiBike provides a flexible and eco-friendly transportation alternative to city dwellers. Users can access the service through short-term passes or annual memberships, which offer unlimited use of bikes for predetermined intervals, making it a convenient choice for daily commutes and casual rides alike.

Despite its popularity and convenience, CitiBike users frequently face challenges related to station availability. After completing a ride, individuals often find no available docks to return their bikes, while those looking to start their journey may encounter empty stations, particularly near residential areas. These issues are indicative of deeper problems in the management and distribution of bike-sharing resources, highlighting the need for more effective operational strategies to ensure the system's reliability and enhance user satisfaction.

CitiBike also serves as a rich data source, providing valuable insights into urban transportation patterns. Each bike trip generates data, including start and end times, origin and destination stations, and trip duration. This wealth of information is pivotal for analyzing user behavior, identifying peak usage times, and understanding the spatial dynamics of bike movement throughout the city. By leveraging this data, planners and operators can better predict demand, manage station stock, and optimize the overall efficiency of the bike-sharing system.

CitiBike generates a vast amount of data that offers profound insights into urban transportation patterns. Each bike trip is tracked, logging start and end times, origin and destination stations, and trip duration. This data is critical for understanding user behavior, peak usage times, and the spatial dynamics of bike movement across the city.

Statistically, the system facilitates millions of rides each year. For example, in recent years, CitiBike reported over 20 million rides annually, with daily rides peaking during warmer months. Analysis of this data helps in predicting demand, identifying high-traffic areas, and optimizing bike and dock availability to improve user experience. Advanced predictive models and machine learning techniques are employed to forecast bike availability and demand patterns, facilitating dynamic rebalancing of bikes across the city to meet user needs efficiently.

The goal of this initiative is to solve the problem of unbalanced bike availability at various city bike stations, where some are overstocked with bikes while others are understocked. It is difficult to provide the city's citizens with effective transit services because of this imbalance. The goal is to predict future demand for bikes at certain stations by applying machine learning (ML) techniques. In order to guarantee that individuals have access to bikes when and where they need them, this predictive capability will enable improved bike allocation and distribution across city bike stations.

II. Our Solution

A. Describing the dataset

The dataset is provided from Citibike system data. It encompasses a rich and comprehensive set of data points that provides a detailed look at bike-sharing usage.

*Ride Identifier (ride_id)*is a unique identifier for each trip recorded. It's essential for tracking individual rides without revealing personal user information, ensuring user privacy. This ID helps in differentiating every single trip, even if other characteristics of the trip (like start time and station) might be similar.

Type of Bike (rideable_type) indicates whether the bike used was a classic pedal bike, an electric bike, or any other type. This classification can help analyze usage patterns based on bike preferences, which can be influenced by trip distances, rider preferences, or geographical topography of the route.

Timestamps (started_at, ended_at) provide the exact times when a rider checked out a bike and when they returned it. Analyzing these timestamps can give insights into peak usage times, average ride duration, and temporal patterns in bike usage, which are crucial for operational planning, such as fleet redistribution and maintenance scheduling.

Station Information (start_station_name, end_station_name, start_station_id, end_station_id) attributes identify where each bike trip started and ended, allowing for detailed analysis of traffic flows between areas. Understanding the most popular routes and stations can help in strategic decisions related to station placement and capacity enhancements.

Geographical Coordinates (start_lat, start_lng, end_lat, end_lng) implicates The latitude and longitude of the start and end stations enable a geographic analysis of the rides. Mapping these can highlight high-density areas of bike usage and reveal geographical trends and challenges, such as higher demands in uphill regions or areas far from public transport hubs.

Member or Casual (member_casual) distinguishes whether the rider was a registered member of the Citi Bike program or a casual one-time user. This differentiation is vital for customer segmentation and marketing strategies. It helps understand user loyalty and can influence promotional campaigns aimed at converting casual riders to members.

B. Preparing the dataset

The conversion of the started_at and ended_at fields from string to datetime format is pivotal for any temporal analysis. Parsing these timestamps allows for more accurate manipulation and computation, such as measuring ride durations or extracting specific components like day or hour for time-based trends. This step is crucial for data integrity, enabling the alignment of time series data across various analyses. Utilizing Python's pandas library pd.to_datetime() function, which efficiently handles diverse datetime formats and standardizes them into a Python datetime object.

Dealing with missing data is essential to prevent biases and potential errors during analysis. This step involves identifying any gaps within key columns such as station names and coordinates, which are crucial for geographic and operational insights. Various strategies include imputing missing values using statistical methods (mean, median, or mode) for continuous variables, or by applying more complex imputation methods based on other data features. Alternatively, rows or entire columns with excessive missing data might need to be excluded to maintain the dataset's quality and reliability.

Data cleaning encompasses standardizing text entries, correcting outliers, and ensuring consistency across the dataset. For instance, station names should be uniform to avoid duplication due to text case differences or typographical errors. Geographic coordinates require verification against unusual or impossible values to ensure all data points are within logical boundaries. These steps are critical for maintaining a clean dataset which provides reliable insights and supports accurate predictive modeling.

Enhancing the dataset with additional features such as ride duration, distance traveled, and categorized ride times can significantly amplify the depth of analysis. Ride duration derived from the difference between 'started_at' and 'ended_at' timestamps offers insights into usage patterns. Distance can be calculated using the Haversine formula, which considers the curvature of the earth for accuracy in measuring the distance between two latitude-longitude points. Additionally, categorizing rides into time segments like morning

or evening helps in analyzing peak usage periods, essential for operational planning and customer engagement strategies.

C. Exploratory Data Analysis

For our Citibike demand forecasting project, Exploratory Data Analysis (EDA) was a crucial initial step that allowed us to gain valuable insights into the patterns and trends within the bike rental data. We began by visualizing the distribution of rentals over time, noting key trends such as peak usage hours, differences in weekday versus weekend usage, and seasonal variations. This helped us understand when and where the demand for bikes was highest, guiding our subsequent modeling efforts. Additionally, we examined the relationship between bike usage and external variables like weather conditions and local events. By plotting bike rentals against weather data, we observed clear trends where certain weather conditions, like mild temperatures and clear skies, significantly boosted bike usage, whereas rain and extreme temperatures led to a drop in rentals.

We also used correlation matrices and scatter plots to identify which variables had the strongest relationships with bike demand. This analysis not only confirmed expected relationships, such as higher rentals on sunny days, but also highlighted less obvious insights, such as the impact of local events on specific stations. For instance, stations near parks and other recreational areas showed a notable increase in rentals on weekends and public holidays. These insights were invaluable for feature selection in our machine learning model, ensuring that we included factors that directly impacted demand. By thoroughly understanding these relationships through EDA, we were better equipped to build a predictive model that accurately reflects the complexities of real-world bike rental patterns, thereby enhancing our forecasting capabilities.

Metric	Value
Total Stations	11593
Total Bikes	2737881
Total Trips Taken	2737881
Average Trips Per Day	85559
Average Duration Per Trip	12.6 minutes
% of Trips Taken by Annual Members	84.3%

Fig 1. Summary of essential statistics

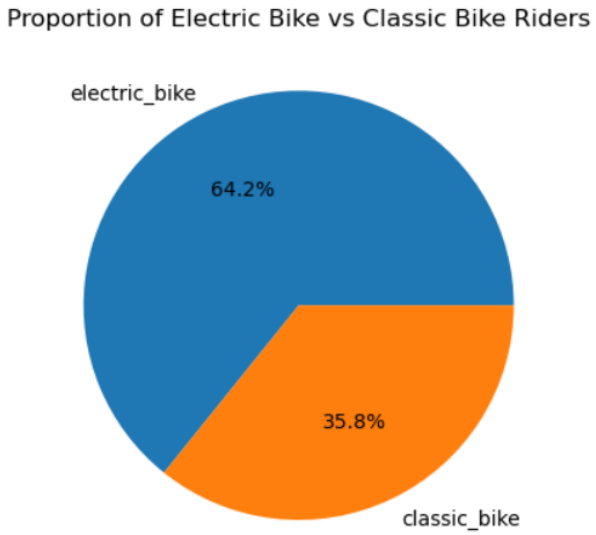

Fig 2. Proportion of Type of Bike Riders

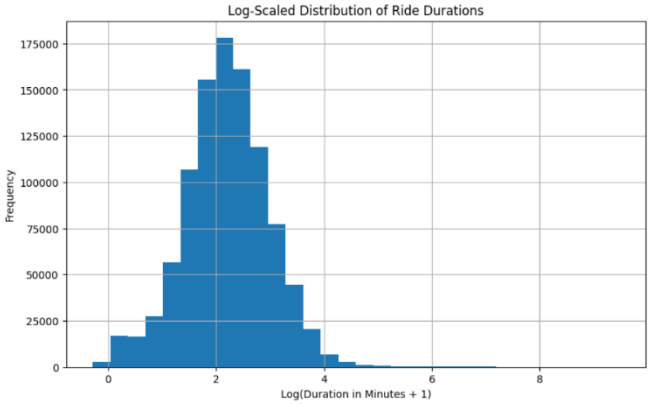

Fig 3. Log Scaled Distribution of Bike Rider

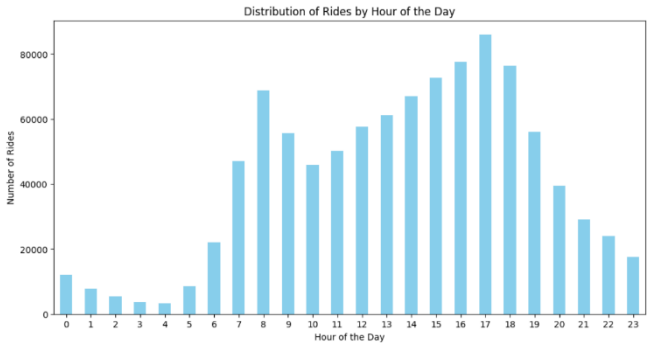

Fig 4. Distribution of Rides by Hour of the Day

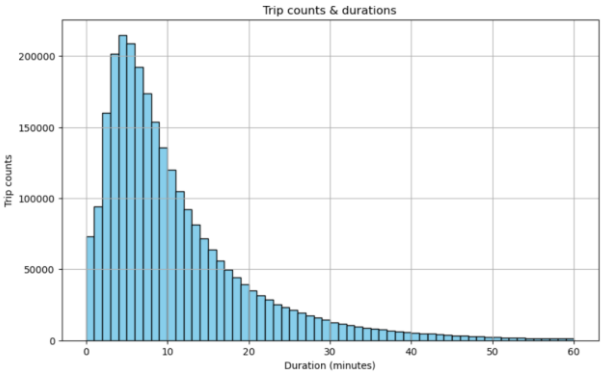

Fig 5. Summary of trip counts and durations

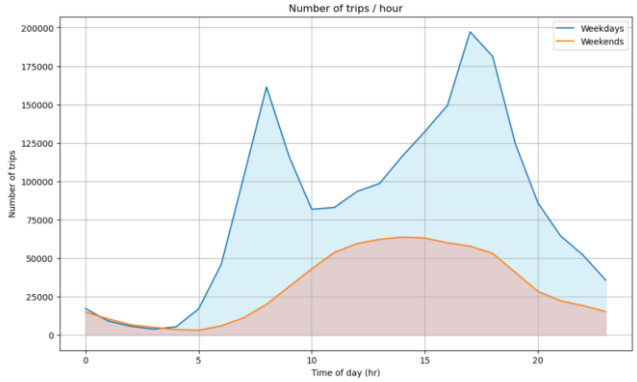

Fig 6. Distribution of rides on weekdays and weekends

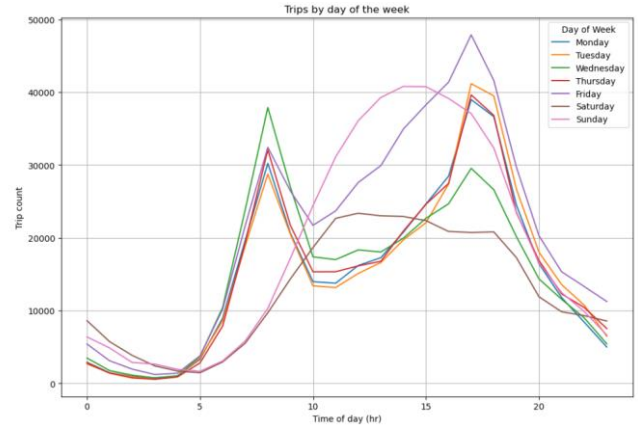

Fig 7. Trips by the day of the week

III. MACHINE LEARNING MODEL

In our Citibike demand forecasting project, we implemented a Random Forest regression, chosen for its robustness and accuracy in handling complex regression tasks. This model utilizes multiple decision trees to make predictions and averages these results to produce a final forecast. This method is particularly effective due to its ability to handle a wide range of input variables and its resistance to overfitting. Our training dataset includes diverse features such as the date and time of rentals, and day of the week, which are crucial for capturing the dynamic and nonlinear dependencies that influence bike rental patterns.

We evaluated the performance of our Random Forest model using several metrics, including Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R-squared. These metrics help us understand the model's accuracy and predictive power on unseen data. Additionally, to ensure the model's reliability and robustness, we employed cross-validation techniques that involved splitting the data into several subsets to validate the model's performance across different scenarios. We also optimized our model by tuning key hyperparameters, such as the number of decision trees in the forest and the depth of each tree, using GridSearchCV and RandomizedSearchCV. This not only enhanced the model's performance but also helped in achieving a delicate balance between bias and variance, ensuring that our forecasts are both accurate and generalizable.

A. Problem Formulation

To tackle the challenge of predicting bike demand within a bike-sharing system, we approached the issue as a regression problem, aiming to estimate both incoming and outgoing bike traffic hourly for each station. We meticulously pre-processed the data to enhance the robustness of our models. This preprocessing involved addressing missing values, converting categorical variables, and dividing the data into two separate datasets: one for outgoing traffic and another for incoming traffic at each station.

The outgoing traffic predicts the number of bikes that will depart from each station at a given date and hour, whereas the incoming model forecasts the number of bikes arriving at each station for similar temporal parameters. By comparing the predictions of these models for any given time and station, we can effectively determine the station’s demand dynamics. A higher predicted outgoing than incoming traffic indicates a demand for more bikes at that station, suggesting an opportunity for proactive bike rebalancing by the city bike handlers to optimize usability and service reliability.

B. Optimization and Training Procedure

- For both the incoming and outgoing traffic, we chose the RandomForestRegressor, recognizing its ability to handle non-linear relationships and its strong resistance to overfitting under default settings. The RandomForest was selected for its natural aptitude in capturing complex interactions and dependencies within the historical data

of bike usage, negating the need for extensive preprocessing of the input features.

- To optimize the RandomForest models, we utilized RandomizedSearchCV, an optimization tool that conducts a randomized search across hyperparameters. This technique strikes an effective balance between exploring various options and exploiting the best findings, making it particularly valuable in expansive hyperparameter spaces by swiftly identifying optimal parameters. Key hyperparameters we focused on optimizing included:

- n_estimators: The number of trees in the forest.

- max_depth: The maximum depth of each tree.

- min_samples_split: The minimum number of samples required to split an internal node.

- min_samples_leaf: The minimum number of samples required to be at a leaf node.

- We implemented RandomizedSearchCV with a cross-validation approach to ensure thorough and overfitting-resistant hyperparameter tuning. The optimal parameters unearthed through this process were subsequently employed to retrain the final models, thereby enhancing their accuracy and generalizability on new data.

- This optimization approach not only improved the performance of the models but also shed light on the importance of various features and the fundamental dynamics governing the bike-sharing system. By understanding which features most significantly influence the models' predictions, decision-makers gain critical insights. These insights enable more strategic planning and informed decisions regarding the management of the bike fleet. Consequently, this deeper understanding facilitates the implementation of more effective strategies for managing the distribution and availability of bikes, ensuring that the system operates more efficiently and meets user needs more effectively.

IV. RESULTS AND CONCLUSION

Random Forest Model		
Metrics	Incoming Traffic	Outgoing Traffic
MSE	1.0616	1.0918
MAE	0.6069	0.6320
R^2	0.2710	0.1975
Accuracy	0.6705	0.6552

The results from the Random Forest models provide valuable insights into predictive capabilities for bike availability in New York City's CitiBike system. The initial models for both incoming and outgoing bike demands showed that there is potential to predict bike availability with a reasonable degree of accuracy. Specifically, the 'RandomForestRegressor - incoming' achieved an R^2 score of 0.271, indicating that approximately 27% of the variance in bike demand can be explained by the model. Similarly, the 'RandomForestRegressor - outgoing' had a slightly lower R^2 score of 0.198.

Hyperparameter Fine Tuned Random Forest Model		
Metrics	Incoming Traffic	Outgoing Traffic
MSE	0.9359	0.9654
MAE	0.6257	0.6427
R^2	0.3573	0.2903
Accuracy	0.6910	0.6837

Upon hyperparameter tuning, improvements were observed in the model's performance. The tuned 'RandomForestRegressor' demonstrated a better fit with an R^2 score of 0.357 and an accuracy of approximately 69.10% for incoming bike R^2

score of 0.290 and an accuracy of approximately 68.37% for outgoing bikes. These metrics suggest that the model can reasonably predict periods of high and low bike demand, which is crucial for managing bike redistribution effectively.

The findings underscore the potential of machine learning techniques, such as Random Forest, to enhance operational efficiencies in urban bike-sharing systems. By continuing to refine these models and integrating real-time data, CitiBike can further improve station management and user satisfaction by reducing the occurrences of empty or full stations and ensuring bikes are available when and where they are needed most.

-

V. REFERENCES

- Citi Bike System Data | Citi Bike NYC | Citi Bike NYC : https://citibikenyc.com/system-data
- Wang, W. (2016). Forecasting Bike Rental Demand Using New York Citi Bike Data.
- O'Mahony, E., & Shmoys, D. (2015, February). Data analysis and optimization for (citi) bike sharing. In Proceedings of the AAAI conference on artificial intelligence (Vol. 29, No. 1)
- Citi Bike Data Analysis — Project | by Azeem Halim | Medium
- https://businesscode.medium.com/python-practice-analyzing-new-york-city-bike-sharing-data-using-pandas-3badfd6ab026
- https://nycdatascience.com/blog/student-works/capstone/solving-the-citibike-station-rebalancing-issue-with-python-machine-learning/