No Bike, Take A Hike

Modeling Bike Availability at CitiBike Station West 116th Street & Broadway

Samuel Braun, Brian Luna, and Kynnedy Smith

August 14, 2023

Abstract

In this report, we will conduct a principal component analysis of availability at Citi Bike stations near the campus of Columbia University.

Keywords: principal component analysis, linear algebra, statistics

1 Introduction

Bike sharing is cost-effective, convenient and environmentally friendly since there are no carbon emissions. A large number of commuters heavily and highly depend on this system as their mode of transport. One of the key challenges for users of bike-sharing systems however, including the Citi Bike system, is the unpredictability of bike availability at different stations and times. This issue is quite relevant for the Columbia community, many of who rely on Citi Bikes to commute between the University's Morningside Heights and Manhattanville campuses, as well as commute from either campus during the congestion of rush hour. The ability to predict bike availability can prove useful for the Columbia community by reducing the time they might spend waiting for an available bike, and therefore increasing the efficiency of the local Citi Bike system.

This report aims to address the challenge of unknown availability by leveraging Citi Bike's bike availability data to predict the availability of bikes at stations located between 110th Street and 123rd Street at any time of day and any given day of the week. By developing a predictive model based on historical bike usage data, we aim to draw conclusions on Citi Bike availability that can provide the community with information that can improve the efficiency and convenience of their commute.

2 Variables, Parameters, and Assumptions

In our analysis, we consider several variables and parameters. The primary variables are `docktime`, `stationid`, `act`, and `cnt`.

- `docktime`: The `docktime` variable represents the time of each event (i.e., when a bike is picked up or dropped off). It is a continuous variable measured in hours and minutes.
- `stationid`: The `stationid` variable represents the unique identifier for each bike station. It is a categorical variable with each category representing a different bike station.
- `act`: The `act` variable represents the action at each event (either bike pick-up or bike drop-off). It is a binary variable for which -1 indicates a bike was picked up and $+1$ indicates a bike was dropped off.
- `cnt`: The `cnt` variable represents the count of available bikes at a specific station at a given time calculated as the cumulative sum of the `act` values.

Identifying traffic patterns can generally provide insight and more broadly serve as a valuable tool in guiding the allocation of road congestion mitigation resources [1]. During periods of high demand, such as morning and evening rush hours, the bike-sharing system can effectively manage the availability of bicycles at popular stations to meet the needs of users.

We assume that `act` values accurately reflect the number of bikes picked up and dropped off at each station and that `docktime` values accurately represent the time of each event. We also assume that the PCA can adequately capture the main patterns and trends in the bike availability data. We make this assumption based on the premise that a large portion of the variance in the data can be explained by a few principal components. However, we aim to validate this assumption by examining the proportion of variance explained by the principal components.

3 Limitations of Data

While our dataset provides valuable insights into bike availability trends, it is important to acknowledge its limitations.

- **Accuracy of Data**: The data is assumed to be accurate, i.e., the `act` values correctly reflect the number of bikes picked up and dropped off at each station, and the `docktime` values correctly represent the time of each event. However, any inaccuracies present in the data collection process could affect the results of our analysis.
- **Missing Data**: If there are any missing or incomplete records in the data, our analysis could include possible bias. For instance, if certain times of day or certain stations are underrepresented in the data, our analysis might not fully capture the bike availability patterns during those times or at those stations.
- **Temporal Scope**: The data in our study is limited to data collected in September 2020. Therefore, the patterns and trends we identify in that month may not apply to other time periods. For example, bike availability patterns could vary by season, and our data might not capture these seasonal variations if it only covers a few months. Our data might further be impacted by the COVID-19 pandemic in ways that might not have occurred at any time after.

- **Geographical Scope**: The data is limited to specific bike stations around campuses. Therefore, the patterns and trends identified may not apply to other Citi Bike stations in New York City or other bike-sharing systems in different cities.
- **Limitations of PCA**: While PCA is a powerful tool for dimensionality reduction, it makes certain assumptions such as linearity, and large variance implies more structure. These assumptions might not hold true for all datasets. Additionally, PCA might not be the best method for data with non-linear structures.

These limitations should be taken into account when interpreting the results of our analysis.

4 Implementation

The implementation of our project involves several steps, each of which is crucial to the analysis of bike availability trends using principal component analysis (PCA).

1. **Data Collection**: We collected data on bike pick-ups and drop-offs from a public bike-sharing system. The data includes the time of each event (`docktime`), the station where each event occurred (`stationid`), and whether each event was a bike pick-up or drop-off (`act`).
2. **Data Preprocessing**: We preprocessed the data by calculating a running sum of the `act` column to generate the `cnt` column, which represents the running total of available bikes at each station at each time. This was done using the formula $c_i = \sum_{k=1}^{i} a_k$, where c_i is the `cnt` value at event i, and a_k is the `ac` value at event k.
3. **PCA**: We applied PCA to the preprocessed data to reduce its dimensionality. This involved transforming the original variables (`docktime`, `stationid`, `cnt`) into a set of uncorrelated principal components (PCs) that capture the maximum amount of variation in the data. Before applying PCA, the data was standardized using the `StandardScaler` function from the `sklearn.preprocessing` Python library. This ensures that each feature has a mean of 0 and a standard deviation of 1, which is a requirement for the optimal performance of many machine learning algorithms. PCA is performed twice: once with 2 principal components and once with 3 principal components.
4. **Visualization**: The transformed data is then visualized using a scatter plot. The colors of the points are determined by whether the day is a weekend or not. The `components_` attribute of the PCA objects in Python gives the loadings of each feature in the original dataset on the principal components. The `explained_variance_ratio_` attribute gives the proportion of the total variance in the dataset that is explained by each principal component.
5. **Insights and Decision Making**: Based on the results of the PCA and our visualizations, we derived insights about bike availability trends and made decisions relevant to the operation and management of the bike-sharing system.

The implementation was carried out using Python, with the help of libraries such as pandas [5] for data manipulation, scikit-learn [6] for PCA, and matplotlib [2] and seaborn [7] for data visualization.

5 Methodology

5.1 Data Sourcing

We sourced our data using a method first published by Cliff Kranish in his article "Estimating Bike Availability from NYC Bike Share Data." [4]. To summarize the author's process in order to clarify how we sourced our data for this report, trip records were transformed from New York City's Citi Bike system into an estimate of the number of bikes available at a given station throughout the day. The purpose of his original study was to measure the availability of bikes and empty docks as a success metric for any bike share system.

The author's first read the trip data file for September 2020, which was then split into two separate dataframes: one for bike pickups and the other for drop-offs. These dataframes were then merged and sorted by the ride start time, offering a view of how the number of bikes at a station fluctuated over time. To visualize bike flow at a station, a graph was plotted showing bike availability throughout the month at a single station near New York's Pennsylvania Station, using the running sum of the act column to show the increase and decrease in bikes at the station. To ensure the count didn't start from zero bikes, the lowest number reached by the cumulative sum was subtracted (so it's absolute value was added if it was negative) to the first cell before the recalculation, providing a more accurate starting point.

This methodology was used to compare different stations based on their surrounding environment, such as stations near business or residential areas. Kranish concluded that, even without direct data on bike and dock availability from Citi Bike, it is possible to estimate this information from the provided trip data files, thus reflecting the actual usage of the bike stations. We hope to further Kranish's approach by conducting a principal component analysis that is explained in the next section in order to draw more detailed conclusions about bike availability beyond simpler data analysis.

5.2 Principal Component Analysis

In our project, we utilize principal component analysis (PCA), a statistical procedure that employs an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components.

The cumulative sum (also known as a "running sum") is a sequence of partial sums of a given sequence. For a sequence of numbers $x_1, x_2, x_3, \ldots, x_n$, the cumulative sum can be defined as a second sequence of numbers $y_1, y_2, y_3, \ldots, y_n$ such that each y_i is defined as:

$$y_i = \sum_{j=1}^{i} x_j$$

In the context of our analysis, x_j corresponds to the `act` values and y_i corresponds to the `cnt` values. This formula is applied iteratively for each row in the dataframe to calculate the cumulative sum of the `act` column.

The data is represented as a matrix $\mathbf{X}$ where each row i represents an event and each column j represents a variable or attribute of the event. The entry $\mathbf{X}_{ij}$ represents the value of attribute

j for event i. The `cnt` column is a running sum of the `act` column. It can be represented as a vector $c = [c_1, c_2, ..., c_n]$ such that c_i is the cumulative sum of the `act` values up to event i:

$$c_i = \sum_{k=1}^{i} a_k$$

This equation means that the `cnt` value at event i is the sum of all `act` values from event 1 to event i. This gives us the estimated number of bikes available at the station at the time of event i.

Since it reduces the dimensionality of our data, employing a PCA makes it easier to visualize and interpret the data and reveal patterns and trends in the data that might be difficult to see in the high-dimensional space. In this project, we conduct a PCA to identify patterns in Citi Bike availability at different times.

The principle components are calculated from the data by finding the eigenvectors of the covariance matrix of the data [3]. Given an nxk data matrix $\mathbf{X}$ whose columns represent variables, and whose rows represent whole records of data, the covariance matrix can be calculated by first centralizing the data, i.e. subtracting the means from each column (name this new matrix B) as follows:

$$\mathbf{B} = \mathbf{X} - \mathbf{h}\left(\frac{1}{n}\mathbf{h}^t\mathbf{X}\right)$$

where h is an $nx1$ column vector of 1's. Then the covariance matrix, $\mathbf{C}$, is:

$$\mathbf{C} = Cov(\mathbf{X}) = \frac{1}{n-1}\mathbf{B}^t\mathbf{B}$$

The eigenvectors of this covariance matrix are the vectors, $\mathbf{v}$, such that:

$$\mathbf{Cv} = \lambda\mathbf{v}$$

for some number λ (the eigenvalue of $\mathbf{v}$). So the effect of $\mathbf{C}$ on eigenvector $\mathbf{v}$ is to change its magnitude but not its direction. There are infinitely many such eigenvectors since we can multiply $\mathbf{v}$ by any number to get another eigenvector, but only finitely many with length 1 (i.e. with $|\mathbf{v}| = \mathbf{v^t v} = 1$). If $\mathbf{C}$ is invertible (which it almost certainly will be if we an adequate amount of data) then it will have exactly k orthogonal eigenvectors. These eigenvectors can be found by diagonalizing $\mathbf{C}$, i.e. finding an orthonormal matrix $\mathbf{A}$, and a diagonal matrix D such that:

$$\mathbf{A^{-1}CA} = \mathbf{D}$$

Then the columns of $\mathbf{A}$ contain the eigenvectors, and the diagonal entries of D contain the corresponding eigenvalues. To calculate the proportion of variation in the original data that is accounted for by a particular eigenvector, we can look at the ratio of its eigenvalue to the sum of all the eigenvalues, i.e. the variance accounted for by the i^{th} eigenvector is:

$$prop_i = \frac{\lambda_i}{\sum_{j=1}^{k} \lambda_j}$$

where $\lambda_1 \ldots \lambda_k$ are the eigenvalues corresponding to eigenvectors $v_1 \ldots v_k$. Then we can order the eigenvectors by $prop_i$ values, and select the top few of them so that a reasonable amount of variation in the data is accounted for.

Once the eigenvectors have been calculated and a subset of them selected, we need to project the original data onto these eigenvectors in order to obtain a new data set, **P** of lower dimension:

$$\mathbf{P} = \mathbf{XW}$$

where **W** is a $k \times l$ matrix whose columns are the selected eigenvectors. The new data matrix **P** will have the same number of rows as **X**, but fewer columns (l instead of k).

6 Results & Analysis

First let's look at some general descriptive statistics before looking at the results of the PCA analysis at the end of this section.

Figure 1 shows a time-series plot of bike availability across all stations during the month of September 2020. There were more bikes available in the first few days than at any other time, which indicates that fewer rides were being taken earlier in the month. There is a regular sequence of spikes along the time-series plot with a daily frequency. This can be explained by the fact that bike usage will be higher at certain times of the day than others, which can also be seen in figure 2 showing the hourly bike availability for the bike station at W 116 St & Broadway on September the 8^{th} (Tuesday) & 9^{th} (Wednesday). There we can see that there is a spike in

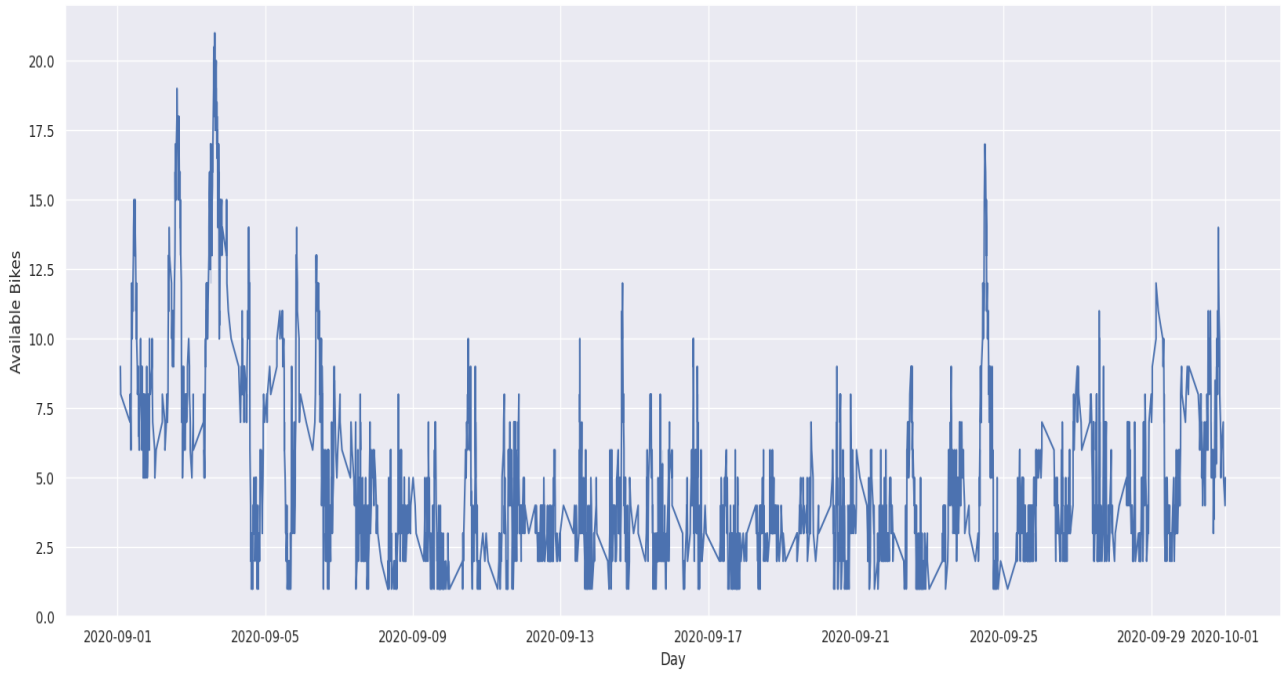

Figure 1: Bike availability across time, all stations

bike availability at around 9am, and another at around 3pm. A possible explanation for the 9am spike is bikes being replaced by the bike hire company in the morning.

Figure 3 shows the number of rides per day throughout September, with weekends colored red, and weekdays colored green. We can see that in general demand is slightly higher on Saturdays, but not much.

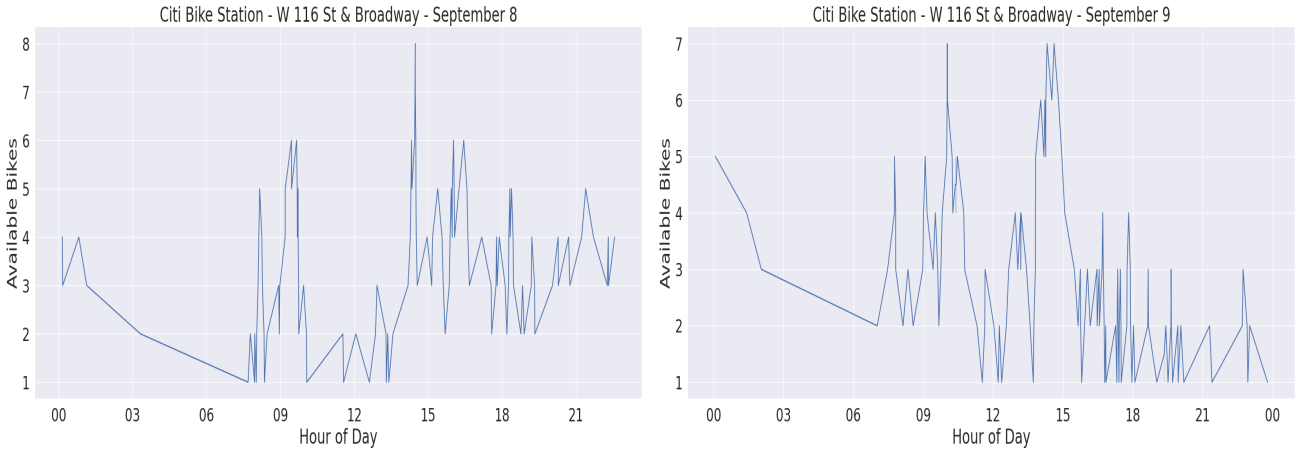

Figure 2: Bike availability at W 116 St & Broadway on September 8^{th}

Figure 4 shows the number of bike rides that ended at the West 116th Street & Broadway station, for each hour of the day, accumulated over the whole of September 2022. We can see that most rides ended around 5pm on weekdays, and midday and 5pm on weekends, with almost none overnight.

For the principle components analysis, we used the data from the West 116th Street & Broadway station. This data was reshaped to create a separate variable for each hour containing the number of bikes available at the station in that hour, with a separate row for each day of the month of September 2020. A principle component analysis was then carried out on these 24 variables.

It was found that the first component explained 59.7% of the variance, the second explained 18.5%, and the third explained just 8.6%.

The loadings of the hour variables onto these components are shown in table 1 in the appendix. We can see that the 1^{st} component has fairly similar loadings for all variables, but slightly lower values for hours 10 to 16, whereas the 2^{nd} component has negative loadings between midnight & 8am, very low positive loadings between 7pm & 11pm, and high loadings between 2pm & 5pm. This suggests that the 1^{st} component is measuring the overall level of bike usage on a particular day, and the 2^{nd} component is measuring the difference between normally busy and quiet hours at the bike station on that day, i.e. if there is a big difference between the bike usage of these times on a particular day, then the value of the 2^{nd} component will be high.

A scatter plot of the data projected onto the first and second principle components is shown in figure 5, with data points corresponding to weekends colored red, and data points corresponding to weekdays colored blue. We can see that most of the data points lie in the lower left quadrant, with just 3 in the upper right quadrant. These 3 data points represent days when there was both high bike usage, and also a large difference between the busy hours and quiet hours. It can also been seen that the red points (corresponding to weekends) generally have lower values along the 2^{nd} component than the blue points, which indicates that there is less variation between busy hours and quiet hours at the weekends.

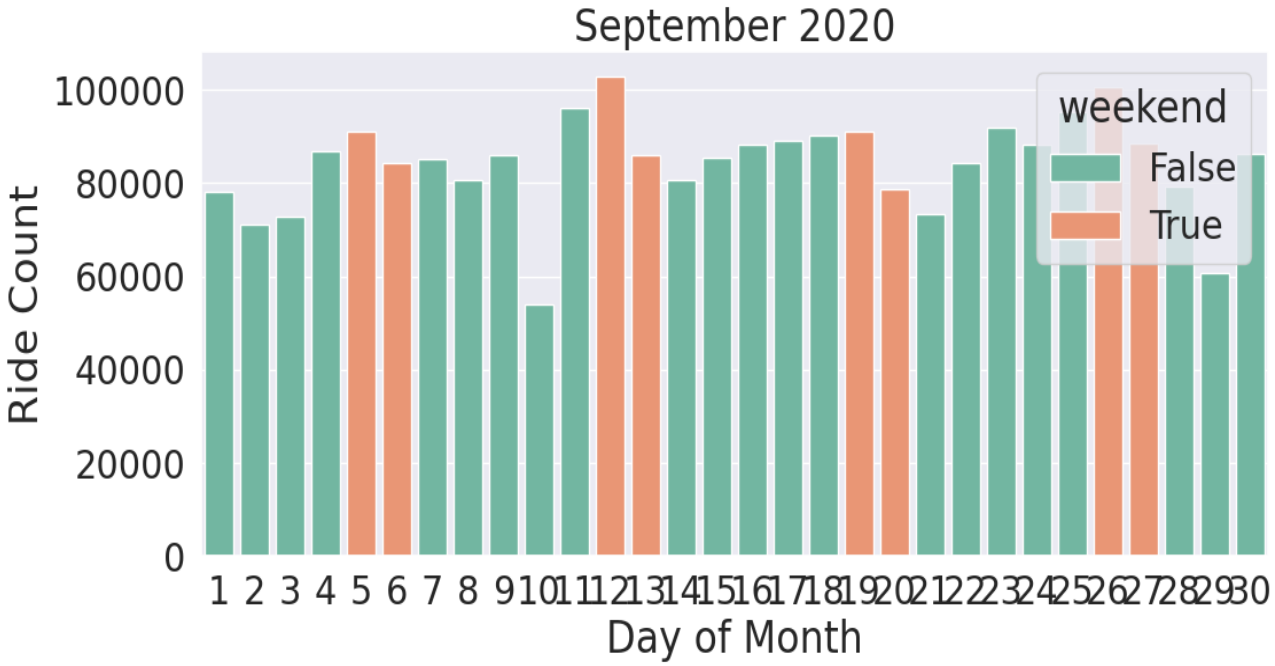

Figure 3: Ride counts per day

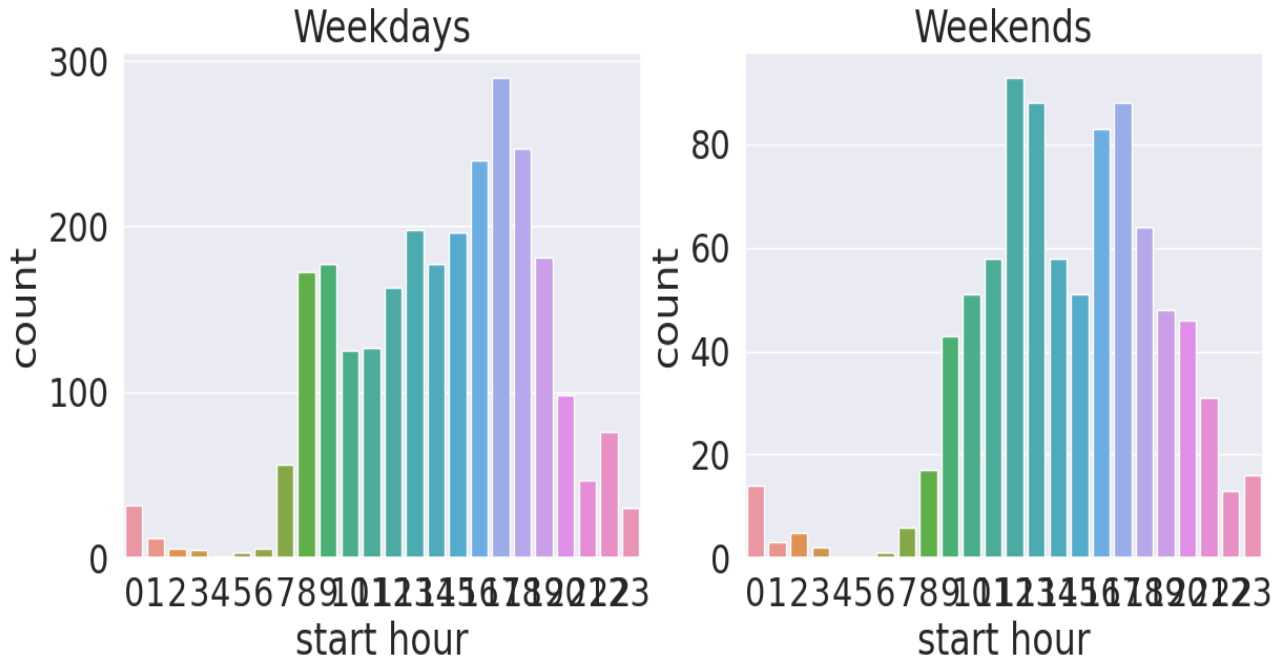

Figure 4: Ride counts per hour; weekdays vs weekends

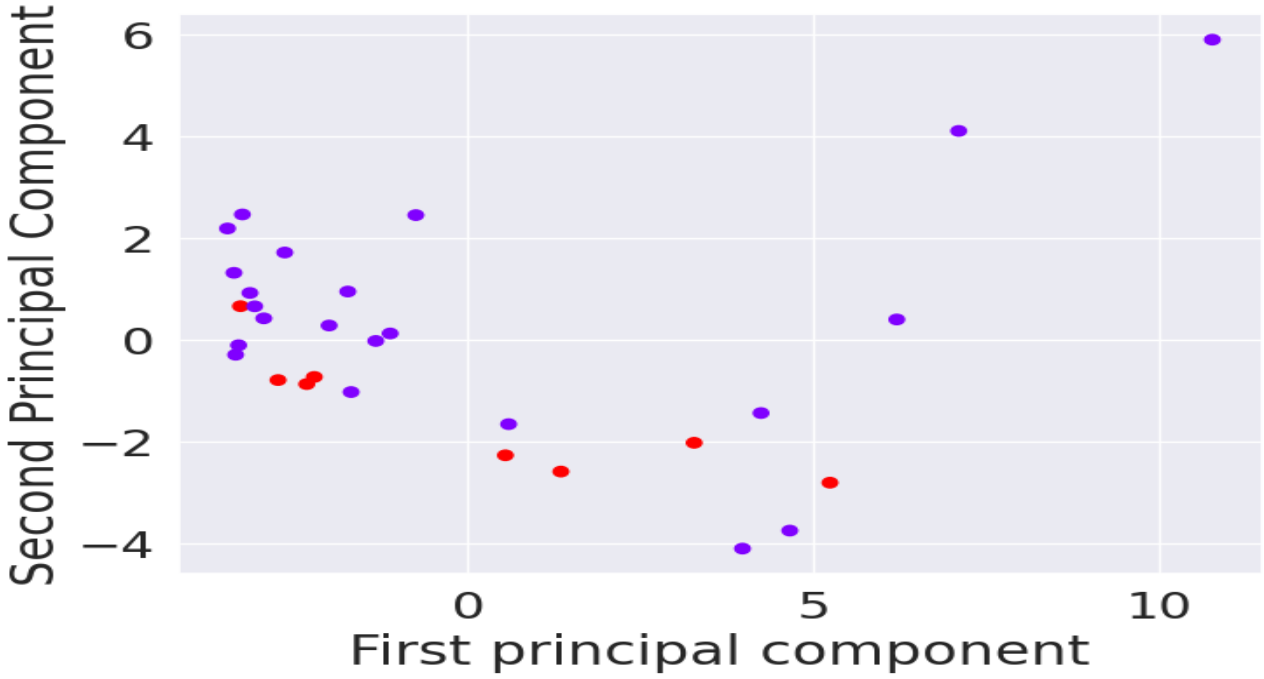

Figure 5: Principle components scatter plot

7 Conclusion

Ultimately in our analysis as documented in this report, we have successfully analyzed Citi Bike availability trends through a principal component analysis. We collected and pre-processed data from publicly available data published by Citi Bike, applied PCA to reduce the data's dimensionality, and visualized the results to identify patterns and trends in Citi Bike availability.

The PCA results showed that the first two principal components explained about 78.2% of the variance in the data, and the first three principal components explained about 86.8% of the variance. This indicates that our PCA was successful in reducing the dimensionality of the Citi Bike dataset while retaining most essential information.

Based on the results of the PCA and our visualizations, we have derived insights about Citi Bike availability trends at the West 116th Street & Broadway Citi Bike station relevant (and, we hope, helpful) to users. Specifically, we noticed a different pattern of usage between weekdays and weekends identified by the different values of the main principle components on weekdays compared to weekends. These insights could be used to assist frequent and new Citi Bike users planning to visit this station.

Additionally, in our general analysis of the Citi Bike data, we noticed that there were spikes in bike availability around 9 am and 2 pm. There was also a general decrease in bike availability around the 12 o'clock hour. While more historical data is needed to officially declare these periods as the official high and low peak times for the W116 St and Broadway Station, these insights could be useful for residents of Morningside Heights to keep in mind as they plan their bike commute times.

For future work, we would like to incorporate additional Citi Bike data unrestricted to the month of September 2020 to further improve our analysis. We also aim to explore other dimensionality reduction techniques and machine learning models to predict Citi Bike availability trends.

References

1. Sijia Feng, Hao Chen, Chun Du, Jun Li, and Ning Jing, *A hierarchical demand prediction method with station clustering for bike sharing system*, 2018 IEEE Third International Conference on Data Science in Cyberspace (DSC), 2018, pp. 829–836.

2. J. D. Hunter, *Matplotlib: A 2d graphics environment*, Computing in Science & Engineering **9** (2007), no. 3, 90–95.

3. I. T. Jolliffe, *Principal component analysis*, 2 ed., Springer Series in Statistics, Springer, New York, NY, 2002.

4. C. Kranish, *Estimating bike availability from nyc bike share data*, https://towardsdatascience.com/estimating-bike-availability-from-nyc-bike-share-data-7cfc4655d5f6, 2021.

5. The pandas development team, *pandas-dev/pandas: Pandas*, February 2020.

6. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, *Scikit-learn: Machine learning in Python*, Journal of Machine Learning Research **12** (2011), 2825–2830.

7. Michael L. Waskom, *seaborn: statistical data visualization*, Journal of Open Source Software **6** (2021), no. 60, 3021.

A Appendix

A.1 Appendix A

Hour variable	1^{st} component	2^{nd} component
00	0.17828568	-0.29433958
01	0.21742063	-0.23212498
02	0.22168395	-0.23368424
03	0.21902643	-0.24866532
04	0.21902643	-0.24866532
05	0.22277863	-0.23533972
06	0.2175461	-0.24145613
07	0.22738561	-0.18968912
08	0.21348704	-0.13349219
09	0.20623868	0.06619549
10	0.18966184	0.10572868
11	0.17302565	0.18683662
12	0.17662243	0.16789499
13	0.17743088	0.21309499
14	0.17571206	0.29547523
15	0.16913432	0.32518034
16	0.16370019	0.30648773
17	0.21293503	0.23034153
18	0.20527668	0.1623154
19	0.2064054	0.0953454
20	0.21094807	0.05250133
21	0.22931277	0.06891911
22	0.23178349	0.08111332
23	0.20814895	0.00938313

Table 1: Loadings of the first two principle components

A.2 Appendix B

See the code for our project here: view Colab notebook