Predicting Bike Share Demand in Chattanooga, USA

Introduction

Bike sharing has become an increasingly popular mode of transportation in cities all over the world. These systems allow people to rent bikes on a short-term basis, often for a small fee, and use them to get around the city.While bike sharing has many benefits for cities, it also faces challenges such as operational issues due to the limitations of docks at stations. Cities can leave a lot of money on the table by not operating their bike systems effectively because potential customers will not be able to rent bikes if there are none available or no available docks to leave a bike after a ride. It is in the city's best interest to balance between keeping stations equipped and docks open without having to build dozens of docks at each location, which would be financially infeasible. Bike system administrators could be more proactive about the placement of bikes to avoid availability issues. For example, they could prompt users with incentives to leave their bike at a station that is running low on bikes, or their staff can take open up availability at docks of stations that are popular destination. These measures require timely, accurate predictions of bike inventory at stations.

The aim of this study is to forecast bike demand in the City of Chattanooga, specifically the change in bike inventory at every hour at popular bike dock stations. Chattanooga, Tennessee, USA launched its bike sharing system, called "Bike Chattanooga," in July 2012. Initially offering 30 stations and 300 rental bikes, Bike Chattanooga has subsequently grown to 42 stations and more than 400 bikes. Chattanooga's bike sharing program is particularly interesting because it was established in collaboration with Outdoor Chattanooga, an outdoor center that is a main attraction for tourists. This means that the city has a lot to gain in terms of tourist ridership by balancing demand at the stations near Outdoor Chattanooga and the partnership could ease with bike redistribution by taking advantage the center's staff on ground. For this reason, Bike Chattanooga is an ideal example of how improving operations from demand predictions could be very advantageous for cities.

Data

The primary data for this study is from the City of Chattanooga's Open Data Portal (City of Chattanooga). The dataset includes all the bike trips from the launch of the program until June 24, 2022, and specifies the start and end times, the start and end stations/locations, and the length of the trip. The analysis uses data starting from the end of 2020. Altogether there are 135,639 trips. The data was cleaned by dropping any trips lasting for less than a minute (8835 trips or 6.5%). These trips tended to end in the same station and were assumed to be mistakes or tests by the user. Trips that were missing an end station (11 trips) were also dropped from the dataset.

The data was resampled to hourly data and grouped by start station and end station. The difference between the number of trips starting and ending at a station every hour was calculated to find the "change in bike inventory", the variable of interest. Ideally, the change in bike inventory will be near 0, which means that the number of bikes being taken from the station is similar to the number of bikes being returned to the station that hour. Weather data was also gathered to use as a feature in modelling under the hypothesis that weather conditions would affect riders' decision to cycle. Specifically, the total precipitation in the 3 preceding hours was collected using the Meteostat API which hosts data provided by the National Oceanic and Atmospheric Administration.

All data collection, cleaning, feature engineering, analysis and modelling in this study was conducted in Python. The code and comprehensive results can be found in the accompanying Jupyter notebooks.

Exploratory Data Analysis

Figure 1 shows a map of the 41 bike stations that were operating throughout the duration of the dataset based on their number of starting trips. Most stations are in downtown Chattanooga and had about 2000 – 4000 trips. Station 1299 has been labelled because it was the starting point of the most trips by far with over 15,000 trips. Station 1299 is the station outside Outdoor Chattanooga, the attraction for adventure activities in the city.

Figure 2 below displays bar charts that show the number of trips starting and ending at each station. Station 1299 is not only the starting location of most trips but also the most popular ending station. Given that this station is driving much of the demand in the city, and it located at popular stop for tourists who are great potential customers, this study will focus on predicting the demand, specifically at Station 1299.

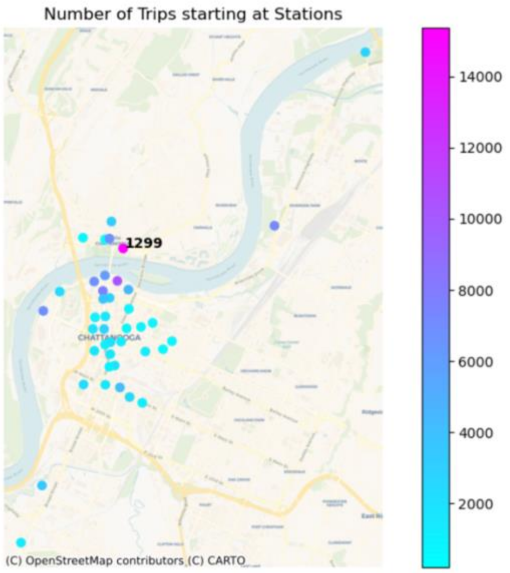

Number of Trips starting at Stations
1299
CHATTANOOGA
14000
12000
10000
8000
6000
4000
2000
(C) OpenStreetMap contributors (C) CARTO

Figure 1

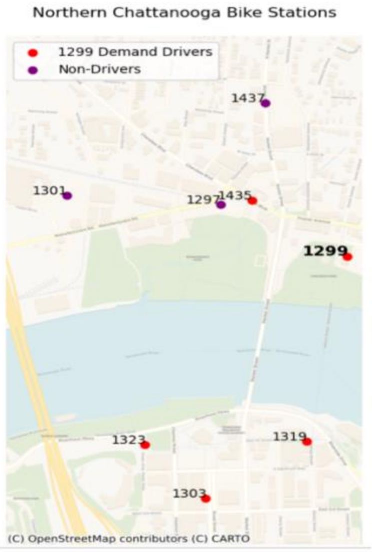

Northern Chattanooga Bike Stations
● 1299 Demand Drivers
● Non-Drivers
1437
1301
1297
1435
1299
1323
1319
1303
(C) OpenStreetMap contributors (C) CARTO

Figure 1B

Investigating Station 1299

The mean length of rides from Station 1299 was 44 minutes and the average was 36 minutes, suggesting that aggregating the data hourly would be suitable, so the study will focus on hourly data prediction. The calculation of the change in bike inventory (number of trips starting minus number of trips ending at station) showed that overall 65% of the time the change in inventory was zero which is ideal. However, when restricting the data to the most active hours of the day including the opening times of Outdoor Chattanooga (9am to 5pm), the change in inventory is negative 26.5% of the time. This could be alarming because recurrent negative inventory suggests unavailability of bikes at the docks which could frustrate potential riders which underscores the need to predict bike demand.

Figure 3 shows that activity in Station 1299 is driven by the same 5 stations: Station 1319, 1323, 1435, 1303 and 1299 itself. These stations were both the most popular destinations and origins for trips starting and ending at Station 1299. This information is useful in considering other variables to include in the models to predict the change in bike inventory.

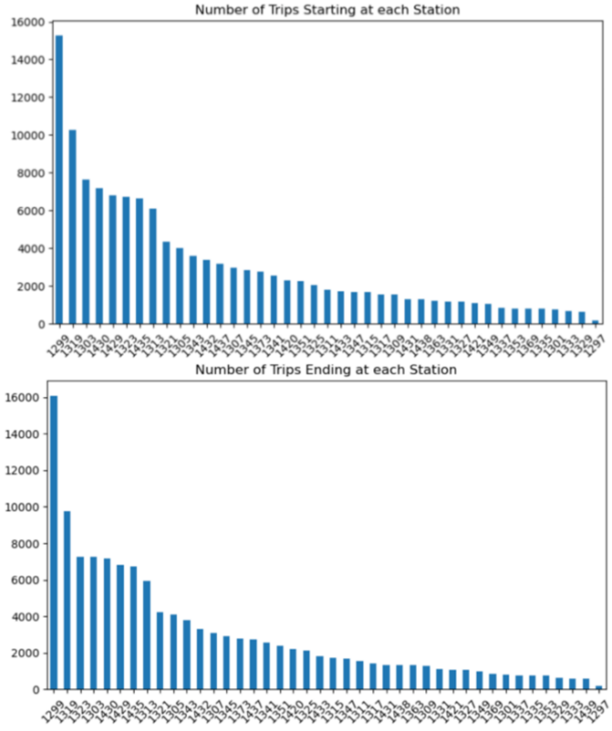


Figure 2

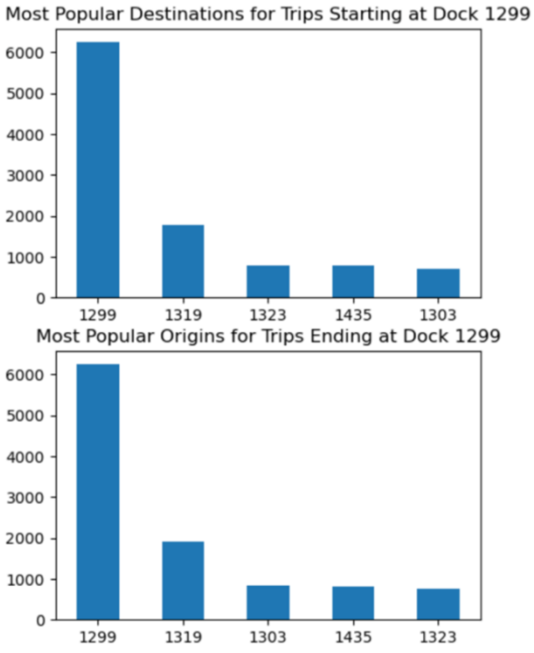


Figure 3

Figure 1B depicts a map zoomed in to Northern Chattanooga near the Tennessee River, showing all the stations near Station 1299. The map shows that the 4 other stations driving the activity at Station 1299 are nearby. However, it also suggests that there isn't a distance-based correlation between the stations and how they drive demand because several other stations are equally or even closer to Station 1299 and their activity is fairly unrelated to that of Station 1299. For this reason, a times series model that also includes data from other relevant stations (over a spatial one) seemed more suitable and was used in the analysis.

Statistical Exploratory Data Analysis

For the most part, the change in inventory data is within -14 and 14 except for a few outliers mainly on the 11th and 12th and November 2021. From a visual analysis, the series appears stationary. However, additional statistical tests are conducted to test for stationarity. The Augmented Dickey-Fuller (ADF) test was conducted to test for the presence of a unit root which is an indication of non-stationarity in a time series. If the test statistic is less than the critical value, then the null hypothesis is rejected, and the time series is considered stationary (Dickey, 1979). The result of the test was a test statistic of 22 with a p-value less than 0.0001, indicating that our data was stationary, and no differencing would be required.

Next, the autocorrelation and partial autocorrelation functions were used to measures the correlation between the time series and its lagged values. Figure 5 displays the corresponding correlograms.

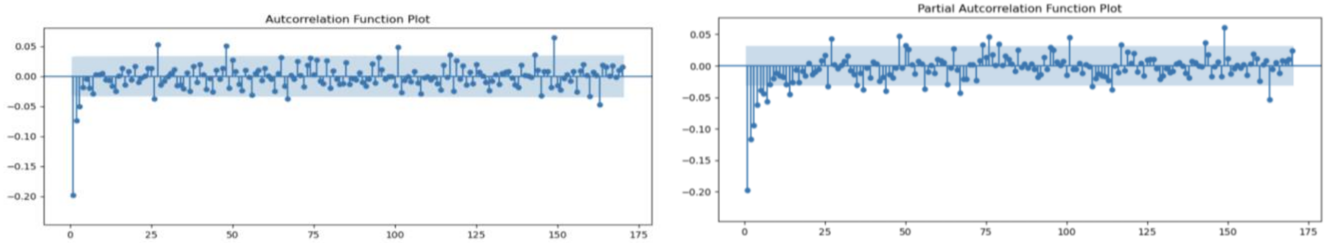


Figure 5

From the function plots, the first 7 lags are important in predicting the change in the bike inventory though the first 3 lags are the most useful. Additionally, there appears to be some seasonality as significant lags appear around every 24-26 hours which means the value at that hour in the previous days is also correlated with the current value.

Seasonal Autoregressive Integrated Moving Average (SARIMA)

Given the seasonal trends observed during the exploratory data analysis, a seasonal autoregressive integrated moving average (SARIMA) model was used.

SARIMA (p,d,q) (P,D,Q)s is defined as

$$Y(t) = c + \varphi 1y(t-1) + \dots + \varphi py(t-p) - \theta 1e(t-1) - \dots - \theta qe(t-q) + \varepsilon(t) + \Phi 1y(t-s) + \dots + \Phi Py(t-ps) - \Theta 1e(t-s) - \dots - \Theta Qe(t-qs)$$

where Y(t) is the value of the change in bike inventory at time t, $\varphi 1, \dots, \varphi p$ are the autoregressive (AR) parameters, y(t-1), ..., y(t-p) are the lagged values of the time series, $\theta 1, \dots, \theta q$ are the moving average (MA) parameters, e(t-1), ..., e(t-q) are the lagged values of the errors, $\varepsilon(t)$ is the error or noise at time t, $\Phi 1, \dots, \Phi P$ are the seasonal autoregressive (SAR) parameters, y(t-s), ..., y(t-ps) are the lagged values of the time series at seasonal intervals, $\Theta 1, \dots, \Theta Q$ are the seasonal moving average (SMA) parameters, e(t-s), ..., e(t-qs) are the lagged values of the errors at seasonal intervals and p, d, q, P, D, Q, and s are the orders of the AR, differencing, MA, SAR, seasonal differencing, and SMA components, respectively.

Several variations of the SARIMA model were considered. From the ACF and PACF plots, an AR order of 4 and MA order of 3 would be considered as well as an AR model using all the first 7 lags. The model variations used in the study included SARIMA(4,0,0)(1,1,0)$_{24,}$ SARIMA(0,0,3)(0,1,1)$_{24,}$ SARIMA(7,0,0)(1,1,0)$_{24,}$ SARIMA(4,0,0)(1,1,1)$_{24,}$ SARIMA(4,0,1)(1,0,1)$_{24.}$ Models including both high-order AR and MA orders were attempted but not documented as these failed to converge. All models had all coefficients as statistically significant at the 5% level.

The models were trained with data from Jan 1st to May 28th and used predictions were made for the next 24 hours using the normal forecasting of the model as well as a rolling prediction. For every new prediction in the rolling prediction approach, the model was trained again with data including the additional ground-truth data for the previous hour before the prediction was made. This scenario is more representative of the real-world scenario where the bike sharing operators would have information on the true demand in the previous hour to make their forecast and decisions.

Table 1 below shows the resulting RMSE values of each of the SARIMA models rolling predictions.

SARIMA Model Variant	RMSE
SARIMA(4,0,0)(1,1,0)$_{24}$	3.6902
SARIMA(0,0,3)(0,1,1)$_{24}$	3.6195
SARIMA(7,0,0)(1,1,0)$_{24,}$	3.6956
SARIMA(4,0,0)(1,1,1)$_{24}$	3.6165
SARIMA(4,0,1)(1,0,1)$_{24.}$	3.6209

Table 1: RMSE for SARIMA model variants

The RMSEs values of the models are comparable and are quite large. However, plotting the predictions shows a different story. Figures 6A to 6E show the plots of the predictions (orange) and the true values (blue). The MA seasonal order appears to be what is driving the prediction values. Figures 6A and 6C whose predictions are almost identical both have seasonal MA orders of 0, while Figures 6B, 6D and 6E which have seasonal MA orders of 1 have similar predictions. Another thing noted from corresponding RMSEs and plots is that although SARIMA(4,0,0)(1,1,0)$_{24}$ (Figures 6A) and SARIMA(7,0,0)(1,1,0)$_{24}$ (Figures 6B) both follow the pattern of the true values better than the other models, their RMSEs are higher because predicting a value of 0 for the whole time series overall averages to lower residuals since the change in bike inventory is centered around 0.

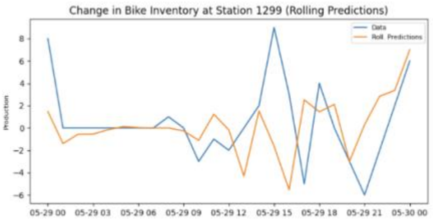


Figure 6A: SARIMA(4,0,0)(1,1,0)$_{24}$

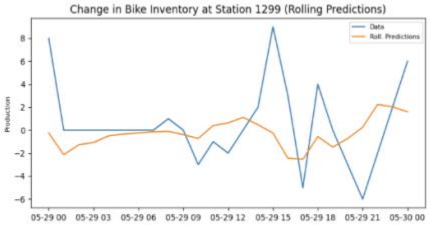


Figure 6B: SARIMA(0,0,3)(0,1,1)24

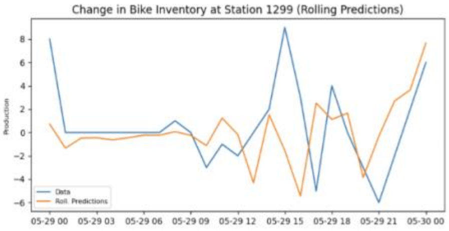


Figure 6C: SARIMA(7,0,0)(1,1,0)$_{24}$

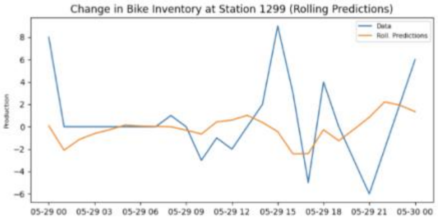


Figure 6D: SARIMA(4,0,0)(1,1,1)$_{24}$

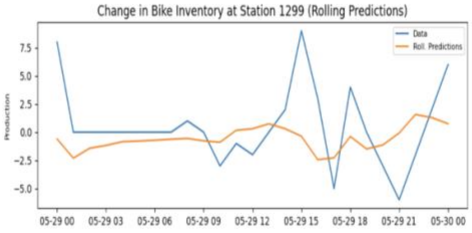


Figure 6E: SARIMA(4,0,0)(1,0,1)$_{24}$

Vector Autoregressive Model (VAR)

In order to improve the accuracy of the model, features from nearby stations that were drivers of the activity around Station 1299 were included using a vector autoregressive model. The VAR model is a multivariate extension of the autoregressive model that assumes that the values of all the variables in the model are jointly determined by their own past values and the past values of the other variables in the model.

The VAR model is defined as:

$$Y(t) = c + A_1Y_{t-1} + A_2Y_{t-2} + ... + A_nY_{t-n} + \varepsilon_t$$

where Y(t) is a vector of n endogenous variables at time t, c is a constant vector, A_1, A_2, ..., A_n are n × n matrices of coefficients, ε_t is a vector of error terms with mean zero and constant covariance matrix Σ, and p is the number of lags used in the model (Lütkepohl, H., 2005).

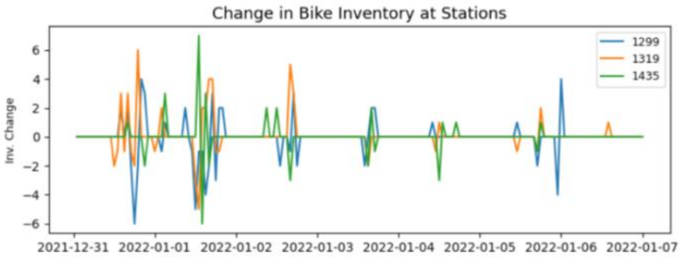


Figure 7

The time series for the change in inventory at Stations 1319 and 1435 as well as the precipitation for the past 3 hours were attempted to be included in the model. Figure 7 shows the values for Station 1299 and Stations 1319 and 1435 for the week following December 31st, showing that the values follow similar patterns.

The Adfuller test was run on the 3 time series and the results rejected the null hypothesis, meaning that all time series were stationary.Next the granger causality test was conducted to test whether the lagged values of the time series help predict the current value of the change in bike inventory at Station 1299 (Lütkepohl, H., 2005). The test was conducted for 8 lags. Below are the resulting p-values in Table 2. The results show only that the precipitation lags correlations were not statistically significant. For this reason, precipitation time series was not included the VAR model.

Time Series	Lag 1	Lag 2	Lag 3	Lag 4	Lag 5	Lag 6	Lag 7	Lag 8
Precipitation	0.531	0.613	0.605	0.680	0.700	0.801	0.854	0.892
Station 1319	<0.0001	<0.0001	<0.0001	0.001	<0.0001	<0.0001	<0.0001	<0.0001
Station 1435	0.006	<0.0001	<0.0001	<0.0001	<0.0001	0.001	0.002	0.001

Table 2: P-values for the Granger Causality Test

The select order function (which shows the orders with the lowest AIC and BIC values) was used to choose the order for the model. A lag of 5 was chosen as this had the lowest AIC value. All the coefficients of the resulting model were statistically significant.

The model was used to make rolling predictions for the next 48 hours. The result was an RMSE of 3.85 which was higher than those of the SARIMA models. Figure 8 shows the plot of the predictions of the VAR model. The model underperformed in predicting bike inventory. This might be as a result of the lack of the daily seasonal component that seems to play a main role in forecasting in the SARIMA models.

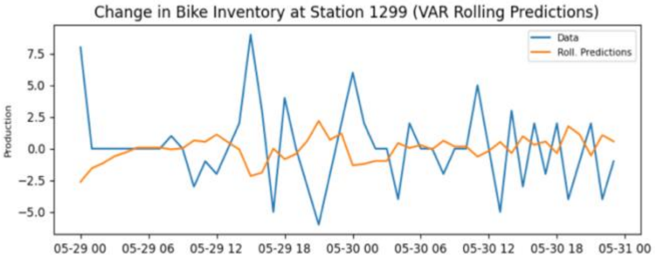


Figure 8

Random Forest Model

To provide more flexibility than the traditional statistical time series model, a random forest model was implemented. Random forest is a machine learning algorithm that uses an ensemble method to combine multiple decision trees to create robust prediction models (Breiman, 2001). One of the key advantages of the random forest model is its ability to handle noisy and high-dimensional data, allowing for the inclusion of additional features in the model than previously.

Two variations of the random forest model were run. Both models included the 5 lags for Station 1299, Station 1319, and Station 1435, a categorical variable for the time of the day (either early morning, morning, mid-noon, afternoon, evening, night) and a categorical variable whether the day was a national holiday or not. Additionally, the first model had a categorical variable for the day of the week while the second model replaced this feature for a binary variable of whether it was a weekday or weekend. Hyperparameter tuning was conducted using grid search cross-validation. For the first model the best parameters were a max depth of 9 and 100 decision trees, while for the second they were 9 and 200 respectively.

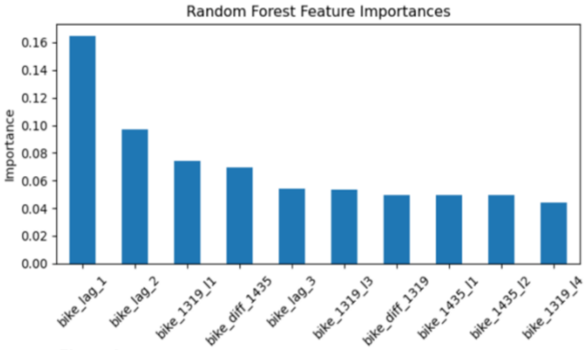


Figure 9

The models were used to predict the hourly change in bike inventory at Station 1299 for a week. Figure 9 shows the 10 most important features for the models. The lag values for the inventory at the stations were the most important predictors.

Both models had similar performance with RMSE values of 2.652 and 2.651 respectively. So far these are the smallest RMSE values found. However, Figures 10 and 11 shows that these models don’t necessarily follow the pattern well and in general are heavily underpredicting the magnitude of the true values. The residuals were plotted to examine if there were any trends, but the residual values are centered around 0 as shown in the figures.

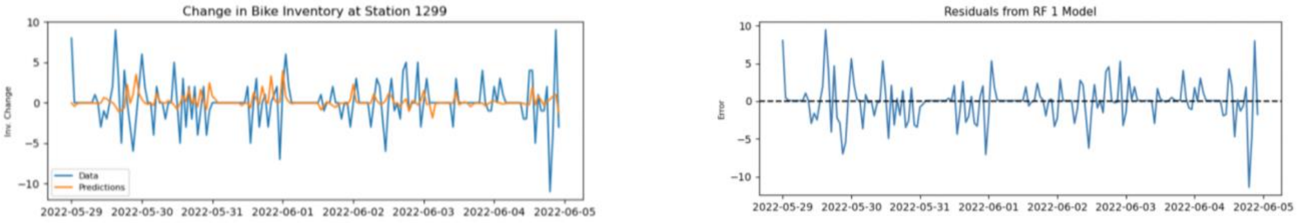


Figure 10: Predictions and Reisduals for First Random Forest Model

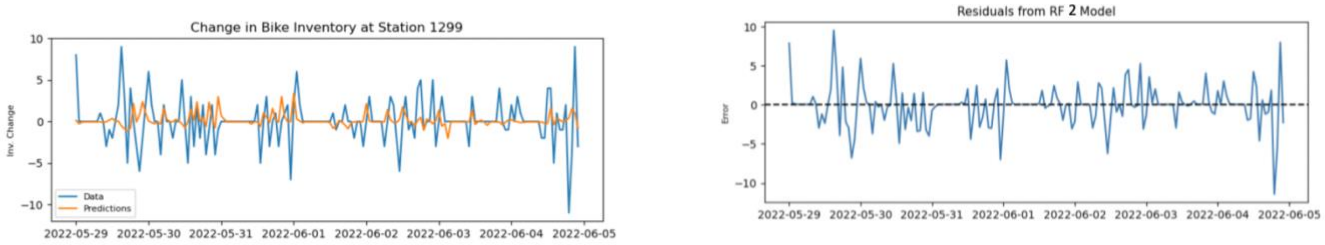
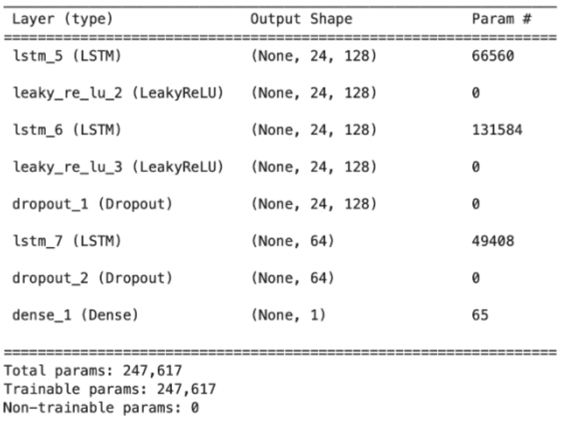


Figure 11: Predictions and Residuals for Second Random Forest Model

Recurrent Neural Network: Long Short-Term Memory (LSTM)

The final model attempted was LSTM, a type of recurrent neural network (RNN) that is particularly useful for time series prediction because it can learn long-term dependencies in the input sequence given its ability overcome the vanishing gradient problem, and can adapt to changing patterns and trends over time (Hochreiter, S., & Schmidhuber, J., 1997).

Table 3 shows the summary of the LSTM model used for predictions. It includes 3 LSTM layers as well as a dropout layer to reduce overfitting and a final dense layer. Like in previous models, the model was trained on the hourly values form Jan 1 2022 to May 15 2022 while the data day from May 15 to 29 (14 days) was used for validation and the week following May 29 was used for testing.



```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_5 (LSTM)               (None, 24, 128)           66560

 leaky_re_lu_2 (LeakyReLU)   (None, 24, 128)           0

 lstm_6 (LSTM)               (None, 24, 128)           131584

 leaky_re_lu_3 (LeakyReLU)   (None, 24, 128)           0

 dropout_1 (Dropout)         (None, 24, 128)           0

 lstm_7 (LSTM)               (None, 64)                49408

 dropout_2 (Dropout)         (None, 64)                0

 dense_1 (Dense)             (None, 1)                 65

=================================================================
Total params: 247,617
Trainable params: 247,617
Non-trainable params: 0
_________________________________________________________________
```

Table 3: Summary of LSTM model

The model only uses the previous values from Station 1299. Two versions of this model was run, one using 4 lag values (and a batch size of 16) and one using 24 lag values (and a batch size of 8) to hopefully capture the daily seasonality. The adam optimizer, mean absolute error metric, and learning rate of 0.0001 was used. The number of epochs was determined by a callback stopping the training if the validation loss did not improve after 3 epochs.

The results of the two univariate models were similar with the 24-lag model and 4-lag model having minimum validation loss values of 5.092 and 5.270 and RMSE values of 2.235 and 2.303 respectively. Figure 12 shows the plots of the predictions of the models. The LSTM did not perform as well as expected. Similar to the random forest model, it underpredicts the magnitude of the values. Note that additional models were run using scaled values but these models performed worse and their results can be found in the jupyter notebooks).

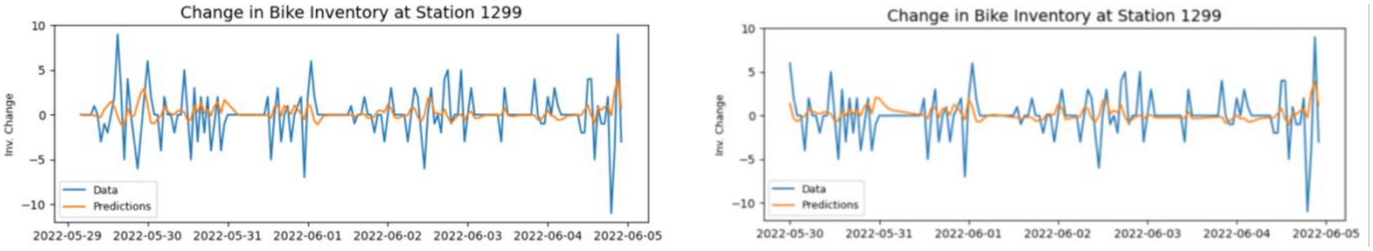


Figure 12: Predictions for 4-Lag LSTM (left) and 24-lag LSTM (right)

Hoping to improve the accuracy, a multivariate LSTM (including the time series from Station 1319 and Station 1435) was also run. The multivariate model was run with 5 lags to match the random forest model and VAR models after observations of little improvement using the 24-lag univariate model. All other parameters were the same used for the univariate 4-lag LSTM.

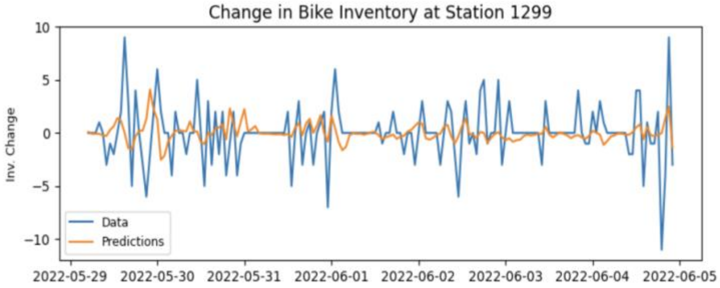


Figure 13: Predictions for 5-Lag Multivariate LSTM

The minimum validation loss of this model was 5.261 and it had an RMSE of 2.290 which was comparable to that of the univariate model and as seen in Figure 13, the prediction plot was very similar to that of the univariate LSTM models.

Overall, while LSTM had the lowest RMSE values, its results were still lower than expectations as they were a minimal improvement to the previous models.

Discussion and Conclusion

The aim of this study was to forecast the hourly change in bike inventory within the Chattanooga bike sharing system. The analysis focused on Station 1299, the most popular station within the network which is at a tourist attraction and has much potential for revenue increase from improvement. Four models were attempted for the predictions: seasonal autoregressive moving average (SARIMA) model, Vector autoregressive model (VAR), random forest, and Long Short-Term Memory (LSTM) Recurrent neural networks.

While for the most part the RMSE values reduced the complexity of the model, overall, the improvements to the model were minimal and the accuracy of all the models were not very high as the models repeatedly underpredicted the absolute magnitude of the true values, forecasting values near 0 for most of the hours. One of the reasons for this phenomenon might be the fact that a majority of the values in the change in bike inventory is 0. This is actually an ideal situation since it means that most of the time the number of bikes being picked up and dropped of at the station are equal. Consequently, the bike sharing system might generally be already self-balancing to some extent.

Nevertheless, future work could attempt other approaches to improve predictions. For example, given that the true values seem to alternate between times where the bike inventory change is stagnant at 0 and others with large fluctuations, a Generalized Autoregressive Conditional Heteroscedasticity (GARCH) model could be useful because it models the variance of a time series as a function of its past values and past errors (Engle, 1982). Alternatively, the modeling could be changed from a regression problem to a classification problem trying to classify periods of high negative values, high positive values and zero values and the analysis could attempt forecasting with different data aggregation levels such as every 20 mins or every 4 hours. Furthermore each of the models could be possibly be improved with additional data such as integrating seasonal data with the vector autoregressive model, or adding the time series of the other stations driving the activity at Station 1299 (since these models only included the first 2 stations), or trying out other recurrent neural networks such as the Gated Recurrent Unit (GRU) model.

Works Cited

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5-32.

City of Chattanooga. (2022). Bike Chattanooga Trip Data. ChattaData. https://www.chattadata.org/Recreation/Bike-Chattanooga-Trip-Data/tdrg-39c4. Accessed Jan 30 2023.

Dickey, D. A., & Fuller, W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. Journal of the American statistical Association, 74(366a), 427-431.

Engle, R. F. (1982). "Autoregressive Conditional Heteroscedasticity with Estimates of the Variance of United Kingdom Inflation". Econometrica. 50 (4): 987–1008.

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural Computation, 9(8), 1735-1780. doi: 10.1162/neco.1997.9.8.1735.

Lütkepohl, H. (2005). New Introduction to Multiple Time Series Analysis. Springer.