sustainability

MDPI

Article

Forecasting the Usage of Bike-Sharing Systems through Machine Learning Techniques to Foster Sustainable Urban Mobility

Jaume Torres, Enrique Jiménez-Meroño and Francesc Soriguera *

BIT—Barcelona Innovative Transportation, Universitat Politècnica de Catalunya-BarcelonaTech, 08034 Barcelona, Spain

* Correspondence: francesc.soriguera@upc.edu

Abstract: Bike-sharing systems can definitely contribute to the achievement of sustainable urban mobility. In spite of this potential, their planning and operation are not free of difficulties. The main operational problem of bike-sharing systems is the unbalanced distribution of bicycles over the service region, resulting in zones where bicycles are scarce and zones where bicycles accumulate. In order to provide an acceptable level of service, the operator needs to carry out repositioning movements, which are costly. Bike-sharing repositioning optimization solutions have been developed that rely on the estimation of the expected number of requests and returns at each location. Errors in this prediction are directly transferred to suboptimal repositioning solutions. For this reason, the development of methodologies able to accurately forecast bike-sharing usage is an issue of great concern. This paper deals with this problem using machine learning regression methods, which yield usage predictions from inputs such as historical usage and meteorological data. Three different machine learning regression techniques have been analyzed (i.e., random forest, gradient boosting, and artificial neural networks) and applied to a case study based on the New York City bike-sharing system. This paper describes the variables of the models and their calibration processes. Results are analyzed and compared in order to determine which one of the three techniques and under what conditions is the most adequate. Comparisons are not only made in terms of accuracy but also with respect to the applicability of the algorithms. Results indicate that, given the similar accuracy of all methods, the simpler calibration process of the random forest technique makes it advisable for most applications.

Keywords: sustainable urban mobility; sustainable transportation; shared-mobility; bike-sharing; station-based; demand prediction; machine learning regression

Citation: Torres, J.; Jiménez-Meroño, E.; Soriguera, F. Forecasting the Usage of Bike-Sharing Systems through Machine Learning Techniques to Foster Sustainable Urban Mobility. *Sustainability* **2024**, *16*, 6910. https://doi.org/10.3390/su16166910

Academic Editors: Giovanni Leonardi, Yusheng Ci, Lina Wu and Ming Wei

Received: 14 May 2024
Revised: 29 July 2024
Accepted: 8 August 2024
Published: 12 August 2024

1. Introduction and Background

The concept of bike-sharing consists of providing a fleet of bicycles for users to use to make trips without needing to own them. This shifts the focus to a mobility-as-a-service model. From the user's perspective, bike-sharing, as with other vehicle-sharing initiatives (e.g., car-sharing, motorbike-sharing, scooter-sharing), offers the advantages of low-cost on-demand transportation (i.e., flexibility, traveling when and where needed) in front of the traditional public transportation alternatives with predefined routes and schedules [1]. Politically and societally, bike-sharing systems represent a significant step towards sustainable urban mobility by balancing ecological, economic, and social interests [2]. Bicycling is therefore promoted for its benefits to sustainable mobility, public health, and mitigating traffic congestion [3]. In turn, as a non-motorized and low-emission transport mode, bicycling reduces energy consumption and carbon emissions, thereby enhancing the urban environment.

Despite the previous potential benefits, the operation of one-way bike sharing systems is usually affected by the imbalance in the spatial and temporal distribution of their demand. This can lead to a lack of available bikes or parking spots in certain areas at specific times [4]. To better meet the travel needs of people in various locations and ensure a balance between the supply and demand of shared bikes, bike-sharing operators invest significant effort in managing these bikes, enhancing dispatching efficiency, and promoting the sustainability of urban transport [5]. Repositioning operations are a fundamental part of vehicle-sharing operation management. Repositioning aims to relocate vehicles within the service region in order to achieve various objectives (e.g., recharge the batteries of electric vehicles, perform maintenance operations, or balance the system to maximize the demand served and improve vehicle availability). In the particular case of station-based bike-sharing systems, repositioning operations are carried out by vans or small trucks, which move groups of bikes between stations with the main objective of reducing the number of empty and full stations and therefore improving the level of service offered. Since this phenomenon is caused by a demand imbalance in the service region, these relocation operations are also commonly called rebalancing operations.

The efficient planning of repositioning operations is important. Repositioning is costly, and to a large extent, it can determine the success or failure of the vehicle-sharing system. Lack of repositioning, or its poor performance, results in the frustration of users when left without the possibility of picking up a bike at the origin of their trip because of empty stations or when unable to return it near their destination due to the lack of available parking slots. Given the importance of the problem, several authors have proposed repositioning strategies and algorithms able to maximize the service provided without incurring excessive repositioning costs. Examples are the works by [6–9]. Despite the fact that these works follow different approaches to solving the repositioning optimization problem, all of them have one thing in common: they rely on an accurate forecast of the expected usage (i.e., requests and returns) at the vehicle-sharing stations. This forecast is fundamental to all models because it is used to define the utility of each repositioning movement. Clearly, it would be more profitable to address stations with high demands that are likely to become full or empty during the operation period. On the contrary, it does not matter much if a station is empty or full when the expected demand during the following hours is negligible. Predicting the bicycle inventory level at stations is one of the key variables in the optimization of repositioning operations, either if it takes place while the system is in operation (i.e., "dynamic" repositioning) or only when the system is closed to users (i.e., "static" repositioning). Take as examples the works by [7,9], which use neural network techniques to update the predicted inventory levels of the system at different times during the repositioning optimization process.

The most recent literature regarding the short-term usage prediction of vehicle-sharing systems relies on advanced regression techniques based on machine learning methods, aiming to explore the generalization capabilities of the models. Works by [10,11] can be considered the starting points of this approach. Specifically, [10] opt for a random forest method for a station-based bike-sharing system, while [11] use long short-term memory neural networks for the case of a free-floating system. Several other authors have contributed with their own ad-hoc methodologies based on different machine learning applications, as in [12–18]. We refer to [19] for an excellent and up-to-date literature review on the application of machine learning techniques to big data from vehicle-sharing systems. Still, to the author's knowledge, there is no research work comparing the overall performance of different machine learning techniques for a particular type of vehicle-sharing system in a single case study, so the accuracy of predictions could be directly compared.

The present work partially fills this research gap and builds on these previous contributions by proposing three alternative machine learning algorithms, namely (i) random forest (RF), (ii) gradient boosting (GB), and (iii) artificial neural networks (ANN), to predict requests and returns at bike-sharing stations, especially in the short term. The RF method is based on assembling multiple decision trees during the training phase of the model. Each decision tree represents a non-linear model that infers a series of rules from the data in order to split it into branches and make predictions based on feature values. The RF method combines the predictions of several trees to improve its robustness and accuracy [20]. Similarly, the GB method is also an ensemble technique that combines weak learners (e.g., decision trees) to create a strong predictive model through iterative improvement [21]. The main difference with the RF is the sequential building of the model, with each new model correcting the errors of the previous ones. Finally, ANN consists of layers of interconnected nodes (neurons, as an analogy to the human brain), where each layer recognizes patterns of the data through training by iteratively updating their parameters to improve accuracy [18]. It must be noted that such a selection of three machine learning techniques responds to their common application in the referred literature, so that a comparison between them in a specific case study can provide valuable insights, as it is equivalently proposed in other fields of forecasting (e.g., prediction of traffic volumes in emergencies [22]). In spite of this, the application of other powerful machine learning techniques that are also usually used in regression and classification applications (e.g., Support Vector Regression), or alternative regression methods such as AutoRegressive Integrated Moving Average (ARIMA) could also be a possibility. This analysis is left as further research.

The implementation of the considered methods uses a time step for the prediction that varies from one to six hours, which yields adequate results in the case of dynamic repositioning. In addition, the prediction focuses on a minimum horizon of one day (i.e., the next day), as this is critical to planning the repositioning operations overnight in the static repositioning configuration. In spite of this, results are analyzed up to a horizon of 15 days. In all cases, the input data consists of historical trip data and variables related to the type of calendar day and meteorology, which are the most influential factors in the determination of bike-sharing usage [23–30]. Note that the usage data for the system is a biased approximation of its demand. Potential demand might be truncated at the station level (e.g., diverted to a nearby station or lost) when the station is empty of vehicles or parking spots. Refs. [31,32] analyze in detail such demand truncation at the station level. If the interest resides in the potential demand prediction, it is worth mentioning that in the present paper, the clustering of nearby stations in the prediction might attenuate such difference.

In summary, the main objectives of this research are twofold. First, to explore the model variables and establish an adequate treatment methodology that allows obtaining an adequate calibration of the models. Second, to compare the accuracy of the predictions obtained with the different methods. Previous research in [14] suggests that accuracy differences are small. In such a case, the most convenient method would be one with a simpler calibration process and easier implementation.

The rest of this paper is structured as follows: Sections 2 and 3 describe the methodology of the analysis. This includes the definition of the considered variables and their data treatment (Section 2), together with the description of the calibration process of the three machine learning methods considered (Section 3). Next, Section 4 introduces a case study based on the New York City station-based bike-sharing system and analyzes the results of the application of the three methods. Finally, this paper ends with the conclusions section and reference list.

2. Predictor Variables for the Inventory Level at Bike-Sharing Stations

2.1. Variables of the Model

All the proposed models will consider up to 13 variables. The first two are the dependent variables, namely the trips generated (i.e., requests) and the trips attracted (i.e., returns) at every station. These are the variables to forecast by using some other predictor variables. Predictors are classified as follows: (i) identifier of each station in the system (1 variable); (ii) time and calendar-related predictors (4); and (iii) weather-related predictors (4). Table 1 details all these variables used in the machine learning algorithms.

Table 1. Variables considered in all the models.

Class	Variable	Type	Encoding
Dependent	Requests (generated)	Continuous	Positive real
	Returns (attracted)	Continuous	Positive real
ID	Station ID	Identifier	Number of stations
Time & calendar related	Season	Categorical	Label 0–3
	Day of the week	Categorical	Label 0–6
	Hour of the day	Discrete	Ordinal 0–23
	Minute of the hour	Discrete	Ordinal 0–59
	Holiday	Binary	Label 0–1
Weather related	Temperature	Continuous	Ordinal 0–5
	Wind speed	Continuous	Ordinal 0–5
	Rain	Continuous	Ordinal 0–5
	Snow	Continuous	Ordinal 0–5

2.1.1. Dependent Variables: Requests and Returns at Bike-Sharing Stations

The proposed model predicts the number of requests (i.e., trips generated; picked-up bicycles) and the number of returns (i.e., trips finished; bikes left) for every station in the system and for a given time-step. The model uses historical data on these dependent variables to learn. Usage data is generally available for most of the systems operating worldwide, although it might be organized slightly differently, with more or less variables reported and with different time aggregations. Typically, the shorter time aggregations are of 1 min, and they usually do not go above 5 min, as this information is used to monitor and inform users about the real-time inventory level at stations. In any case, the time-step of the prediction must be larger than the time aggregations of this historical input data.

The preprocessing of the usage data consists of a few steps. First, a data cleaning process consisting of detecting and deleting extreme outliers and possible errors. Take as an example the common case of stations without demand throughout the day, which usually means that the station is closed or out of order. Second, an aggregation process to fulfill the desired time-step of the regression method. This only applies in the event that the selected time-step is larger than the time granularity of the available data. And third, a standardization process consisting of subtracting the mean and dividing by the standard deviation in order to obtain a standardized variable with zero mean and unit variance.

2.1.2. Time and Calendar-Related Predictors

Time predictors describe when the bike is requested from, or returned to, the station. All bike-sharing datasets include the minute, hour, day, month, and year of each request and return. The calendar variables (i.e., day, month, and year) are processed as follows:

- Day of the month: It is discretized into seven categories corresponding to the day of the week (i.e., Monday to Sunday), which are label encoded to 0–6.
- Month and year: They are aggregated into a single variable (i.e., season) and discretized into four categories (i.e., winter, spring, summer, autumn), which are label encoded to 0–3.

Such discretization of calendar variables responds to the typical usage behavior of bike-sharing systems, with different daily patterns and seasonality [19,33].

In addition, the "hour" and "minute" time variables are ordinally encoded (i.e., to 0–23 and 0–59) and aggregated into the time-step selected for the machine learning method. The selected time-step may range from the refreshing time of the monitored bike-sharing data (usually 1–5 min) to longer time-steps (e.g., hourly or daily predictions), which could provide higher accuracy in the predictions due to the increase in the sample size of the dependent variables and due to the reduction of the variance of the dependent variables over longer time periods.

Finally, the "holiday" variable is included in the database in order to take into account calendar bank holidays and encoded into a binary label 0–1.

2.1.3. Weather-Related Predictors

Four weather predictors are considered: temperature, wind speed, rain, and snow precipitation. Data were obtained from the meteorological registers in the city where the system is operating. Meteorological measurements can be considered continuous variables. However, user behavior is not sensitive to small variations of meteorological variables, so that discretization into different levels would better fit the purpose of predicting bike-sharing usage. For example, the decision to use or not use the system can be affected by whether it rains or not. But, under rainy conditions, users will not distinguish the nuance of a few millimeters more or less of rainfall. The perception of windy conditions could be similar.

In an attempt to adequately model this behavior, each meteorological variable is encoded into six ordinal labels (i.e., 0–5), corresponding to different intensity degrees. For precipitation variables (i.e., rain and snow), a binary transformation (i.e., whether it rains or not) could also be a possibility. In spite of this, it has been found that the approach with six categories yields better results. The reason for this has been identified as being related to the heterogeneity of precipitation conditions in time and space. Episodes with low precipitation levels and only in certain regions of the city can be classified as rainy conditions, but there are still many users who are not affected. Therefore, if these variables are treated as binary, the results are more prone to errors.

3. Machine Learning Methods to Predict the Inventory Level at Bike-Sharing Stations

Three regression methods have been considered and compared in the present work. These are: (i) random forest; (ii) gradient boosting; and (iii) artificial neural networks. In this section, the calibration process for each of them is presented in order of complexity. Random forest is the simplest method and is directly applicable with the calibration of only one parameter, while artificial neural networks is the most complex as they require a particular definition of the network structure of the model. The gradient boosting method lies in between, with two parameters that require calibration.

The proposed machine learning methods perform automated feature extraction to automatically learn important features from the data. From this, the outcome of the regression is obtained (i.e., the prediction of the continuous values of bicycle requests and returns). The objective function for the training and calibration of the methods consists of minimizing the mean absolute error (MAE) between the model outcome and the true realization, whose values are known in the training phase of the methods. The particular optimization procedure for the objective function is a characteristic of each method used. Finally, the performance of the methods is assessed by the accuracy of the prediction. Accuracy here is defined as the complementary of the mean absolute percentage error (MAPE) (i.e., $100 - MAPE$ [%]).

3.1. Random Forest (RF)

One of the advantages of RF is that it is easy to implement. It is only needed to characterize the number of estimators (i.e., the number of decision trees randomly created) in order to obtain an accurate model. For the purpose of this work, an accuracy analysis has been conducted for different numbers of estimators. As observed in Figure 1, the marginal gain in accuracy decreases with the number of estimators. Accuracy strongly grows when adding a few estimators, but the marginal gain is null over 100 estimators. In this case, overestimating the number of estimators does not imply a deterioration of the results but only an increase in computational time. This increase in computational time is notorious when surpassing 100 estimators. In conclusion, 100 estimators are considered to be an adequate number for the RF method.

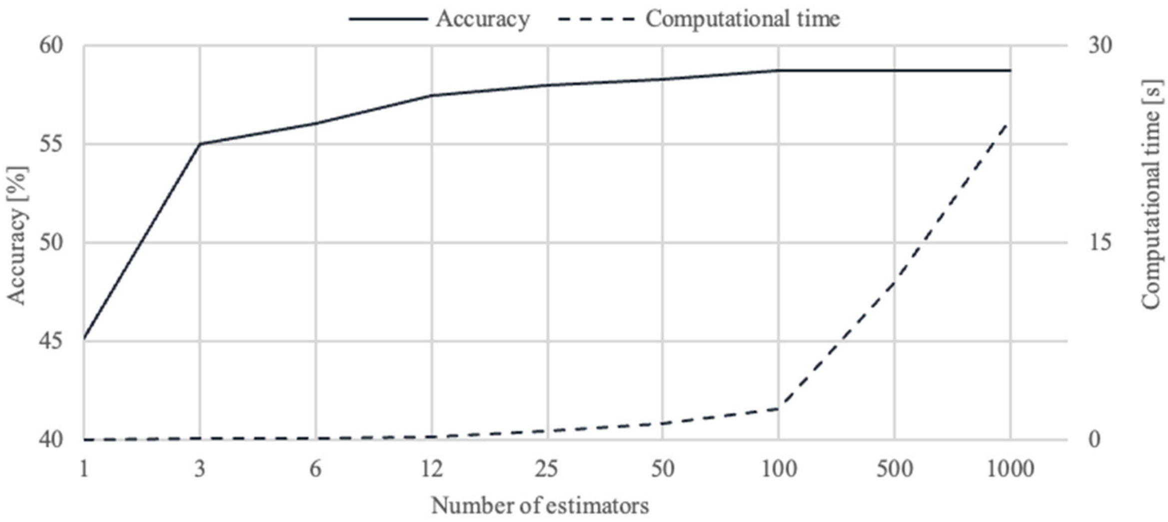

Figure 1. Accuracy analysis for parameter tuning in RF. (Note: Accuracy is defined as the complementary of the mean absolute percentage error).

3.2. Gradient Boosting (GB)

Unlike the RF technique, the GB method creates the predictors sequentially so that each one can learn from the errors in the previous iterations. This means that these algorithms are prone to overfitting if they are not properly controlled through regularization techniques. This is achieved by calibrating two parameters. The first one is again the maximum number of estimators (i.e., decision trees). Unlike the RF, in the GB, an overestimation of this parameter can be detrimental to the results due to the overfitting. The second calibration parameter is the learning rate, a multiplier between 0 and 1, which shrinks the update rule of the algorithm. Smaller values (i.e., towards 0) decrease the contribution of each weak learner in the ensemble. This requires building more estimators and, therefore, more time to finish training, but the final model would be less prone to overfitting. This means that the lower the learning rate, the greater the improvement in the model generalization capabilities.

A calibration analysis has been carried out to determine an adequate value for these parameters. Results are shown in Figure 2. The best accuracy has been achieved with 500 estimators and a learning rate of 0.2. Higher values lead to overfitting problems, and accuracy actually slightly decreases when the learning rate increases.

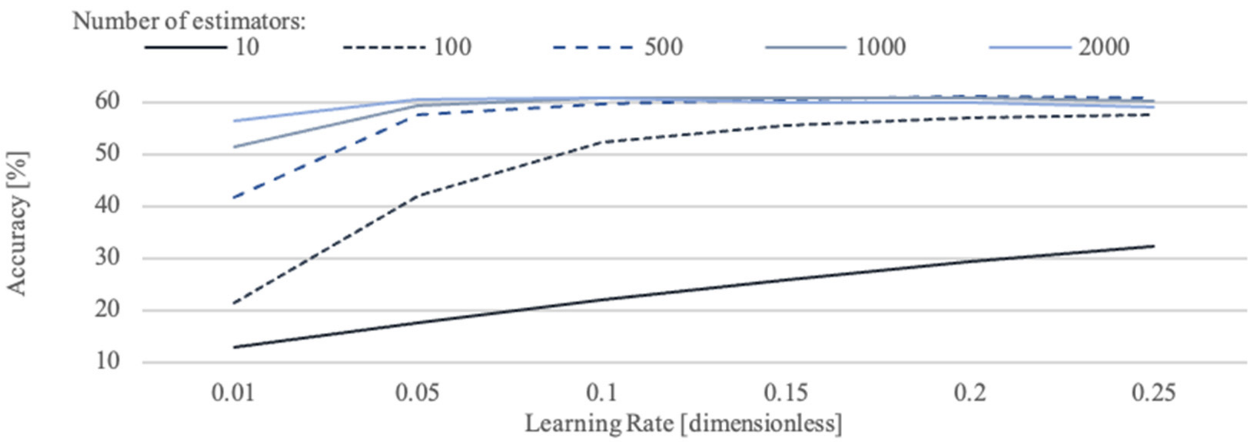

Figure 2. Accuracy analysis for parameter tuning in GB with different number of estimators. (Note: Accuracy is defined as the complementary of the mean absolute percentage error).

3.3. Artificial Neural Networks (ANN)

NN is a much more complex method in relation to the previous RF and GB methods because it is not only needed to calibrate a few parameters but also to build the structure of the algorithm itself. On the one hand, this partly softens the "black box" effect of previous methods since the analyst has more control over the structure of the algorithm and its parameters, which might yield improvements to obtain better accuracy. On the other hand, this increases the complexity of the model as it requires additional calibration efforts without eliminating the risk of overfitting.

In order to define and calibrate the ANN algorithm for the problem being analyzed, different layouts with a different number of hidden layers between inputs and outputs and with different activation functions (e.g., linear, tangent hyperbolic—tanh, rectified linear unit—relu) have been tested to see which one yields the best accuracy. For each proposed layout, the number of epochs (i.e., the number of times the algorithm processes the same data) has been monitored to reach an adequate trade-off between under- and overfitting.

Table 2 summarizes all the experiments, showing that layouts L-1 and L-2, which are based on a single layer, do not have enough training power to achieve acceptable accuracy. Two or three layers are required to predict the output quite accurately. Layouts L-3, L-4, L-5, and L-10 outperform the others in terms of accuracy, with L-3 being the most convenient for the analyzed problem when considering the resulting accuracy and the complexity of the resulting ANN layout.

Table 2. ANN structure for test and calibration.

Layout	Parameter	Input	Hidden 1	Hidden 2	Hidden 3	Output	Epochs	Accuracy [%]
Layout 1	Neurons Act. Func.	n_inputs -	n_inputs tanh	- -	- -	1 relu	25	24.75
Layout 2	Neurons Act. Func.	n_inputs -	2 × n_inputs tanh	- -	- -	1 relu	25	25.24
Layout 3	Neurons Act. Func.	n_inputs -	n_inputs tanh	n_inputs/2 relu	- -	1 linear	50	61.52
Layout 4	Neurons Act. Func.	n_inputs -	n_inputs tanh	n_inputs relu	- -	1 linear	50	61.32
Layout 5	Neurons Act. Func.	n_inputs -	2 × n_inputs tanh	n_inputs relu	- -	1 linear	50	61.15
Layout 6	Neurons Act. Func.	n_inputs -	2 × n_inputs tanh	n_inputs/2 relu	- -	1 linear	50	61.05
Layout 7	neurons Act. Func.	n_inputs -	n_inputs tanh	n_inputs relu	n_inputs relu	1 linear	30	60.98
Layout 8	Neurons Act. Func.	n_inputs -	2 × n_inputs tanh	n_inputs relu	n_inputs/2 relu	1 linear	30	60.76
Layout 9	Neurons Act. Func.	n_inputs -	n_inputs tanh	n_inputs/2 relu	n_inputs/4 relu	1 linear	40	50.54
Layout 10	Neurons Act. Func.	n_inputs -	n_inputs tanh	2 × n_inputs relu	n_inputs relu	1 linear	30	61.54

Note: Accuracy is defined as the complementary of the mean absolute percentage error.

4. Citi Bike NYC Case Study

The previous machine learning regression methods have been applied to a case study based on the New York City bike-sharing program (i.e., Citi Bike NYC), as in [34]. Datasets were retrieved from the available historical database. The whole year 2018 data has been used for training the models, and data from 15–29 January 2019 has been used for testing the accuracy of the results. By the time of the analysis, Citi Bike NYC consisted of 706 stations and 12,000 bikes. Figure 3 shows the spatial configuration of the system, while Figure 4 illustrates the temporal variability of its usage. From Figure 3, it can be observed that the density and capacity of Citi Bike stations are larger in Manhattan and smaller in northwestern Brooklyn, which completes the service area of the Citi Bike system. In turn, it can be seen that trip durations are shorter in midtown Manhattan and larger in the periphery (i.e., lower and upper Manhattan). A similar trip duration distribution is observed in the Brooklyn neighborhood, with shorter trips in the central area of the neighborhood and longer trips in the periphery. This is a consequence of most of the trips having their origin or destination in the most popular areas of the city. Regarding the temporal distribution of trips, from Figure 4, it can be observed the typical weekday usage behavior, with morning and evening peak periods (i.e., at 8 h and 17–18 h), while weekends exhibit a unimodal usage distribution, with a maximum between 12–16 h. In turn, the season of the year has an impact on the overall daily usage, being larger in summer, fall, and spring (in this order) and significantly lower in winter. Such clear temporal variability reinforces the need to use time and calendar-related variables as predictors (see Table 1).

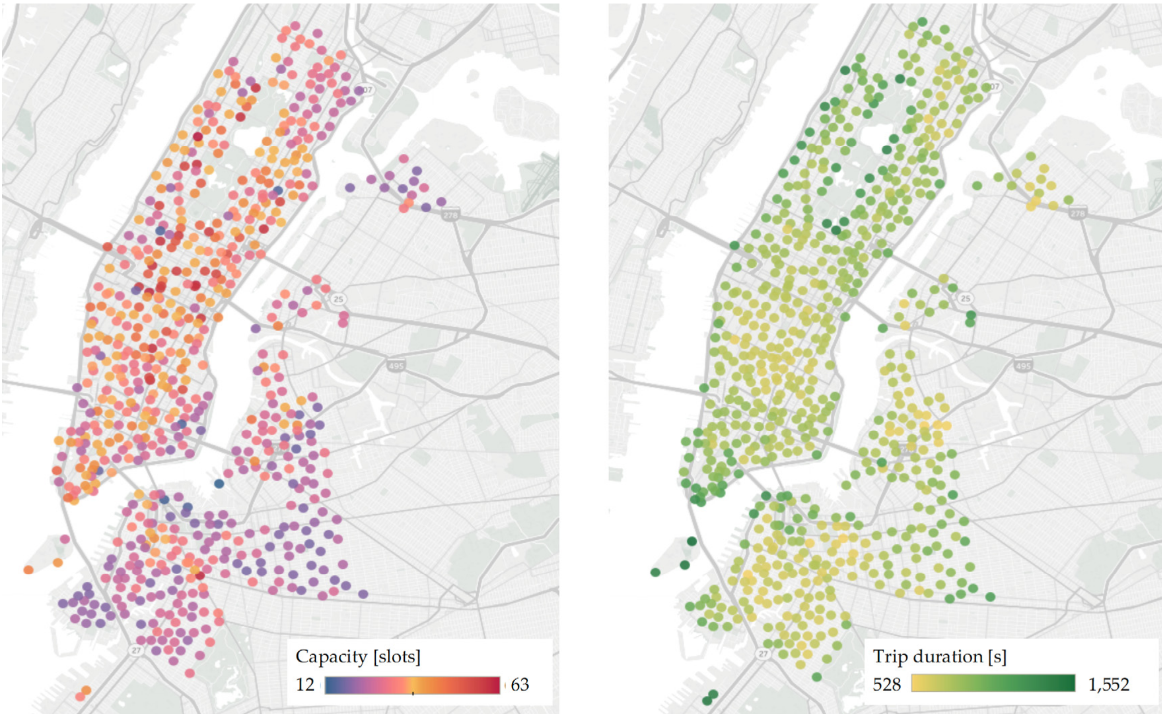

Figure 3. Stations' capacity (**left**) and originated trips' duration [s] (**right**) in the New York City bike-sharing program.

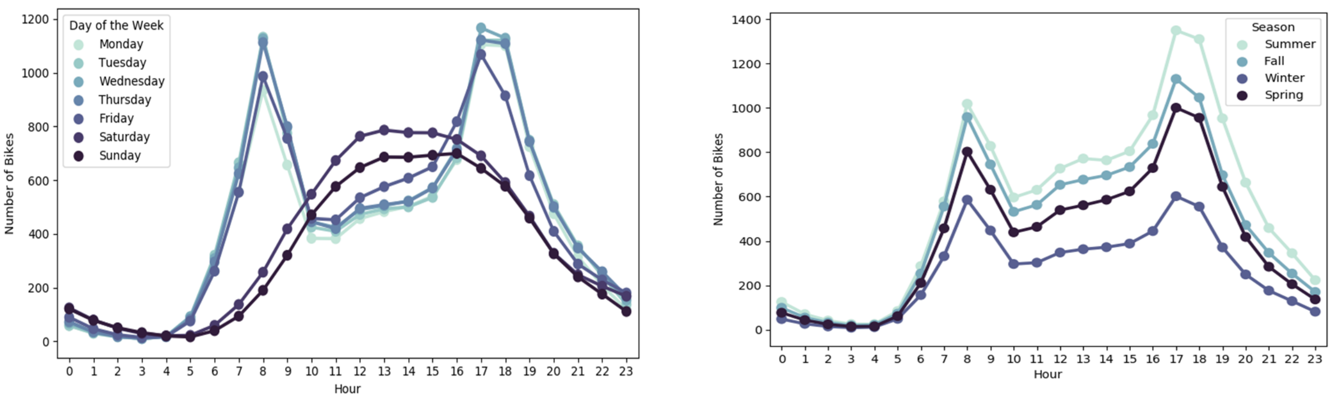

Figure 4. Average hourly usage shows the daily (**left**) and seasonal (**right**) variability of the usage in the New York City bike-sharing program.

Two time-steps have been defined for data aggregation and prediction. In general, the time-step is set to 1 h, which ensures a significant change in the stations' inventory levels and adequate granularity for the application of the results to the planning of repositioning operations. However, in stations with very low average demands (i.e., less than 2 requests or returns per hour), the time-step is increased to 3 h to obtain a significant amount of data. Table 3 shows the results obtained for regular and low demand stations. In turn, Figure 5 illustrates the prediction for a particular station.

Table 3. Usage prediction results for regular and low demand stations.

		Regular Stations (1 h Aggregation Period)						**Low Demand Stations [3] (3 h Aggregation Period)**					
		Random Forest (RF)		**Gradient Boosting (GB)**		**Neural Network (ANN)**		**Random Forest (RF)**		**Gradient Boosting (GB)**		**Neural Network (ANN)**	
		Ret. [1]	**Req. [1]**	**Ret.**	**Req.**	**Ret.**	**Req.**	**Ret.**	**Req.**	**Ret.**	**Req.**	**Ret.**	**Req.**
Working days	Avg. demand	9.3	8.9	9.3	3.4	3.1	3.4	3.4	3.1	3.4	3.1	3.4	3.1
	Avg. Error	3.6	2.8	3.5	1.8	1.7	1.8	1.8	1.7	1.8	1.7	1.8	1.6
	Accuracy [2] (%)	60.8	67.5	62.2	48.3	46.0	47.1	48.3	46.0	47.1	46.8	48.3	47.5
	Max. Error	38.2	18.7	37.6	11.1	8.5	9.7	11.1	8.5	9.7	6.9	10.1	6.5
Weekends & holiday	Avg. demand	4.3	4.2	4.3	2.6	3.0	2.6	2.6	3.0	2.6	3.0	2.6	3.0
	Avg. Error	2.7	2.5	3.7	2.1	1.6	2.0	2.1	1.6	2.0	1.7	2.1	1.8
	Accuracy (%)	38.6	39.9	15.1	19.7	45.2	21.7	19.7	45.2	21.7	42.1	18.8	40.9
	Max. Error	29.4	24.7	31.3	7.4	6.4	8.0	7.4	6.4	8.0	5.6	6.8	5.9
Rainy days	Avg. demand	5.7	5.7	5.7	3.4	3.1	3.4	3.4	3.1	3.4	3.1	3.4	3.1
	Avg. Error	3.3	2.6	3.7	1.6	1.4	1.5	1.6	1.4	1.5	1.4	1.5	1.4
	Accuracy (%)	42.0	54.0	34.3	53.2	55.9	56.2	53.2	55.9	56.2	55.1	55.9	56.0
	Max. Error	38.2	24.7	37.7	11.1	8.5	9.7	11.1	8.5	9.7	6.9	10.1	6.5
Peak Hours	Avg. demand	20.2	27.7	20.2	7.1	4.9	7.1	7.1	4.9	7.1	4.9	7.1	4.9
	Avg. Error	7.5	6.2	6.8	3.2	2.1	2.8	3.2	2.1	2.8	1.8	2.8	1.8
	Accuracy (%)	63.0	77.7	66.2	55.3	58.0	60.0	55.3	58.0	60.0	63.8	60.3	64.5
	Max. Error	25.2	18.7	23.5	11.1	8.5	10.1	11.1	8.5	10.1	6.5	9.7	6.5
Overall	Avg. demand	7.5	7.3	7.5	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1
	Avg. Error	3.3	2.8	3.6	1.9	1.6	1.8	1.9	1.6	1.8	1.7	1.8	1.7
	Accuracy (%)	56.3	61.8	52.6	40.6	46.7	40.9	40.6	46.7	40.9	45.8	41.0	45.7
	Max. Error	38.2	24.7	37.7	11.1	8.5	9.7	11.1	8.5	9.7	6.9	10.1	6.5

Note: [1] "Ret." stands for bike returns; "Req." stands for bike requests. [2] Accuracy is defined as the complementary of the mean absolute percentage error; [3] Low demand stations imply less than two requests or returns per hour.

As seen in Table 3, the accuracy increases when the average number of expected bike movements is higher. Stations with large demand, peak hours, and working days are easier to predict than stations with low demand, weekends, and rainy days, which suffer higher variability in relative terms. Figure 5 further illustrates these results. It shows how Monday 21 January 2019 (i.e., Dr. Martin Luther King, Jr. Day; bank holiday in New York City) is predicted with less accuracy. Also, the usage drop during the rainy morning of Thursday 24 January 2019 is not predicted by the model. Likewise, the model does not capture the abnormally high usage on sunny Sunday 27 January 2019. In addition, Table 3 shows that requests are usually predicted with more accuracy than returns. All three methods provide a similar level of accuracy. RF provides the best results in many cases but suffers an accuracy drop when the sample for training is small (e.g., holidays). GB behaves similarly to RF, with slightly worse results in general and suffering larger accuracy drops in the same contexts. Finally, ANN is less affected by the size of the training database and yields lower maximum errors. Given these results and taking into account the marginal gains of the different methods in particular contexts, the overall conclusion is that any of them would be a good option for predicting the inventory level at bike-sharing stations in order to use it as an input for relocation operation algorithms.

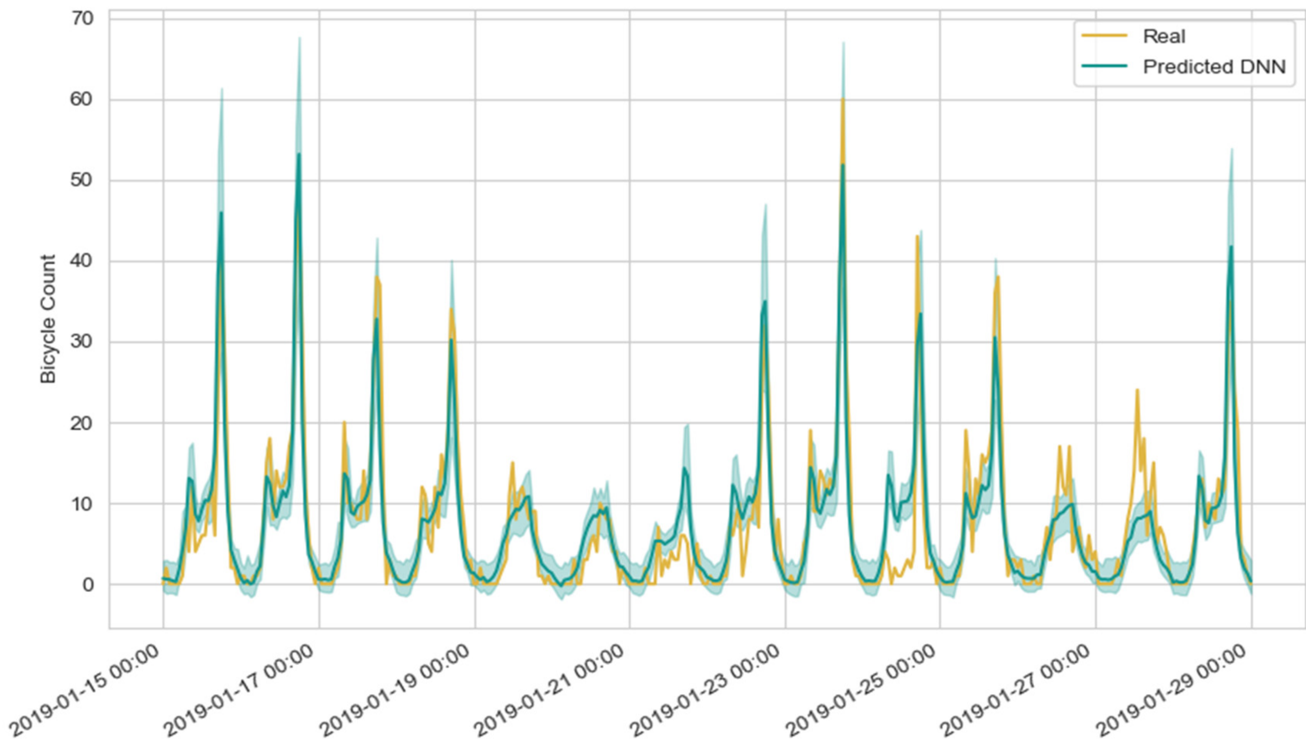

Figure 5. Prediction of requests at Station 402 (high demand) from 15 January 2019 to 29 January 2019 using ANN.

4.1. Effects of the Time Step Duration in the Prediction Accuracy

Overall, the accuracy of the predictions is approximately 60%, at best. This drops to 40% at stations with low demand (even though this error represents a few trips in absolute terms) and grows to 70% during peak periods. Recall that accuracy here is defined as the complementary of the mean average percentage error, meaning 100-MAPE [%]. In summary, the accuracy level achieved, with even the best calibration of the machine learning algorithms considered, is not very high.

The accuracy of the prediction grows (in relative terms) as the demand grows at stations, due to the reduction of the totally random statistical variability. This means that considering longer time-steps (i.e., longer temporal data aggregation) would yield better accuracy of the predictions. Figure 6 shows the accuracy improvement resulting from an increase in the time-step from the defaults (i.e., 1 h for regular stations and 3 h for low demand stations) and up to 6 h. In all cases, the accuracy improvement is significant.

The conclusion is that, regardless of the machine learning method used, the time-step for the prediction should be as large as the applicability of the results allows. For instance, in the case of static repositioning, where rebalancing operations take place only when the system is closed at night, the time-step should include all the daily trips in the system. In contrast, for dynamic repositioning while the system is in operation, the time-step should be a few hours, depending on the repositioning algorithm considered.

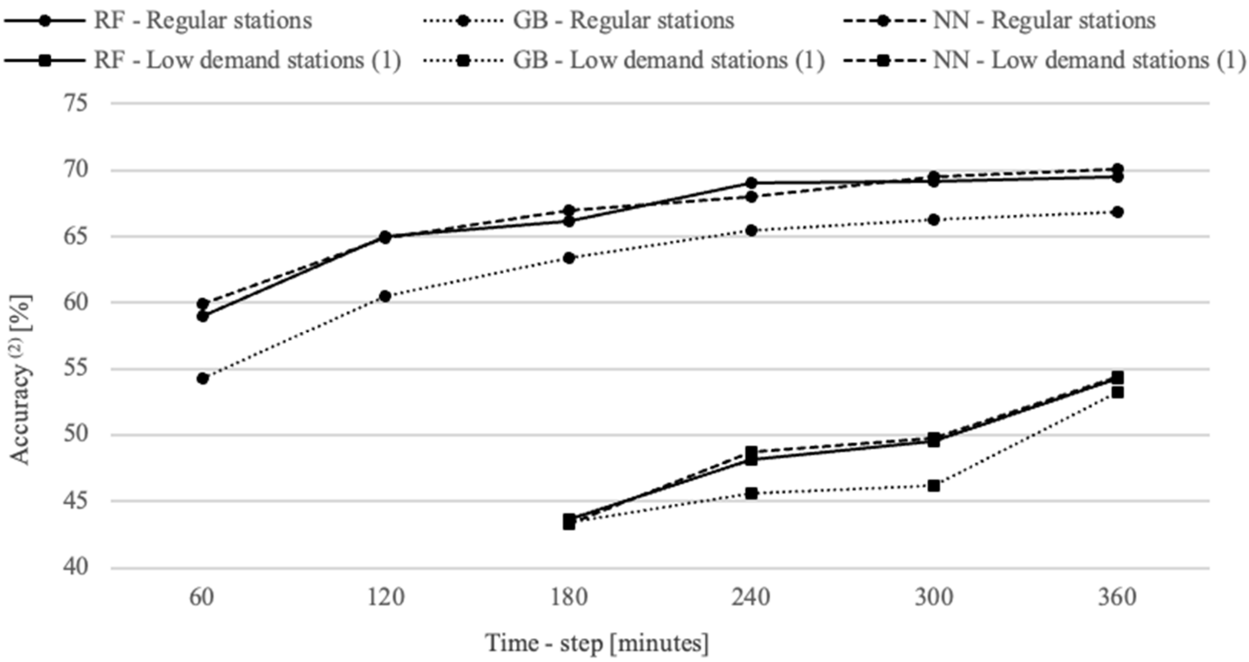

Figure 6. Effects of the duration of the time-step in the accuracy of the prediction at the station level. (Note: (1) Low demand stations imply less than 2 requests or returns per hour; (2) Accuracy is defined as the complementary of the mean absolute percentage error).

4.2. Effects of Spatial Clustering in the Prediction Accuracy

Besides the temporal usage data aggregation, spatial aggregation could also contribute to increasing the accuracy of usage predictions. This is not only due to a higher sample of usage data, which reduces its statistical variability, but also to the fact that, at the station level, usage demand has a significant random part not explained by the considered explanatory variables. Typically, the density of stations is large in bike-sharing systems, and users may choose one or another between nearby stations depending on their bicycle availability. This means that creating clusters of nearby stations with similar aggregate behavior may yield a more predictable aggregate number of requests and returns. This aggregate prediction would still be precious for the operating agency, as the repositioning planning could be performed at the cluster level and executed later on at the station level, as the repositioning team would have information on the real-time inventory of every station in the cluster.

The k-means clustering technique with Euclidean distance has been used to group similar stations in the proposed case study. The variables considered to compute the clustering "distance" between stations have been the UTM coordinates of the station location (i.e., to group nearby stations) and the overall number of requests and returns between 0–12 a.m. and 0–12 p.m. for the different types of days considered (i.e., weekdays, weekends, and holidays). These last variables intend to group stations with a similar aggregated demand pattern.

Results in Figure 7 show the effect of clustering a different number of similar stations on the accuracy of the aggregate prediction of their requests and returns. The accuracy improvements are below 10% in all cases. For regular stations, clusters of 8 stations are enough to achieve most of the improvement, while for low demand stations, clusters larger than 10 stations would be required to achieve some improvement.

In general, considering larger clusters is not an option due to their excessive geographical extension and being considered as a single unit in the repositioning operations framework. For other applications, where the spatial distribution of sharing vehicles is not relevant, predictions could be estimated at the city level (i.e., all the stations of the system together). Results in Figure 8 show that in such cases, the accuracy of the predictions can reach 80%.

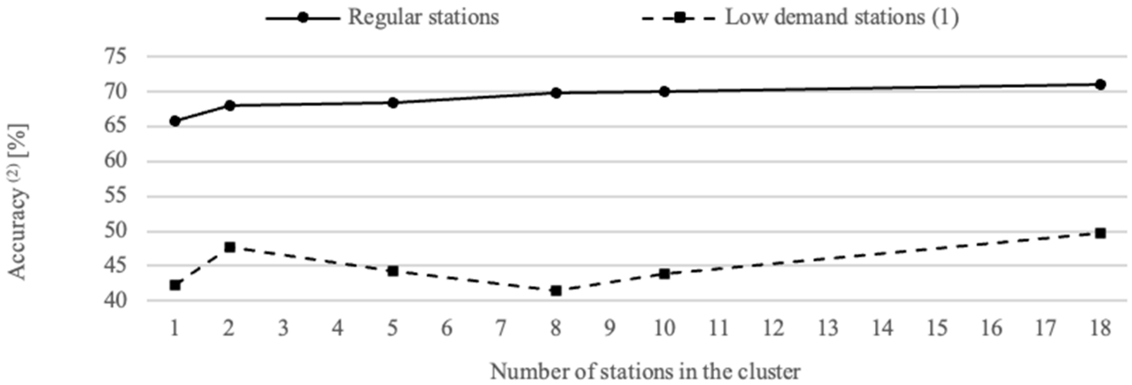

Figure 7. Effects of stations' clustering in the prediction accuracy. (Note: (1) Time-step is 3 h for both stations' types; (2) Accuracy is defined as the complementary of the mean absolute percentage error).

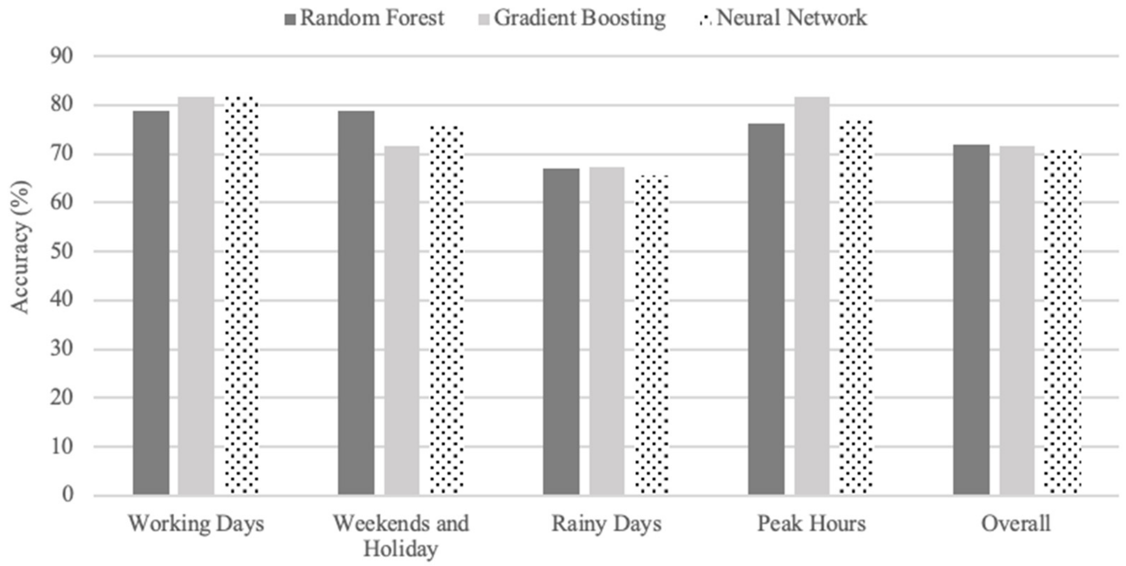

Figure 8. Accuracy of the prediction at the city level using different machine learning methods. (Note: Accuracy is defined as the complementary of the mean absolute percentage error).

5. Conclusions and Further Research

The prediction of the inventory level at bike-sharing stations is an important input, especially for the planning of repositioning operations. In this respect, the present paper fills an existing research gap by providing a comparison of demand forecasting methods for bike-sharing systems based on machine learning algorithms. Three methods have been analyzed: Random Forest (RF), Gradient Boosting (GB), and Neural Networks (ANN). All of them learn from historical usage data of bike-sharing systems and use calendar and meteorological variables as the explicative factors.

In order to test the feasibility and accuracy of the proposed methods, they are calibrated and applied to a case study using data from Citi Bike NYC. In this application, the time-step of the prediction algorithms (i.e., the time-aggregation of data) has been selected so that it yields significant changes in the number of bicycles at stations and provides an adequate response time for the repositioning operations. In the baseline analysis, a one-hour time-step was selected, although for very low demand stations, the time-step was extended to three hours to increase the significance of the results.

Results indicate that differences are small between the accuracy of the calibrated algorithms. In such a context, the simple Random Forest method is an advisable option when a quick, simple prediction is required. Having said that, Neural Networks use a Bayesian approach, and it is the only of the three methods analyzed that is able to provide confidence intervals on the prediction. If this is a requirement in the application of the method (i.e., in the repositioning optimization model considered), then ANN is the only feasible option and also the one that provides the best results in terms of accuracy.

The accuracy obtained for the predicted usage of bike-sharing stations with hourly time-steps is below 60%. Extending the time-step up to 6 h can improve the accuracy to approximately 70%. Similar accuracy improvements could be obtained by predicting the aggregate usage of clusters of 8–10 nearby stations. Such temporal or spatial aggregated predictions would still fulfill the requirements for being fed to most of the dynamic rebalancing optimization algorithms. If only static rebalancing at night is the objective, daily predictions would suffice. In such a case, prediction accuracy might reach 80%.

Improvements to the regression methods used could include a more adequate definition of the calendar variables (e.g., the position of a working day between holidays could be used instead of the day of the week) or a more refined discretization of the continuous variables. Also, the applicability of Support Vector Machine (SVN), a powerful machine learning technique for regression, should be explored. In turn, more specific neural networks (e.g., Long Short-Term Memory ANN, Convolutional ANN, Residual ANN, Transformer ANN, or graph-based ANN) could be applied and analyzed. Finally, the effect of the amount of training data should be analyzed. In the present case study, a whole year of data has been used to predict the next 15 days. It is possible that with much less data, the accuracy would not have been affected.

Author Contributions: Conceptualization, J.T., E.J.-M. and F.S.; methodology, J.T., E.J.-M. and F.S.; software, J.T.; validation, J.T.; formal analysis, J.T.; investigation, J.T., E.J.-M. and F.S.; resources, J.T. and F.S.; data curation, J.T.; writing—original draft preparation, J.T., E.J.-M. and F.S.; writing—review and editing, F.S.; visualization, J.T., E.J.-M. and F.S.; supervision, E.J.-M. and F.S.; project administration, F.S.; funding acquisition, F.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Spanish Ministry of Science and Innovation (*Ministerio de Ciencia e Innovación*) grant number BES-2017-081266 within the project TRA2016-79019-R.

Data Availability Statement: The usage data for the Citi Bike NYC bike-sharing program is available at: https://citibikenyc.com/system-data (accessed 13 May 2024). Weather data for New York City is available through the U.S. National Weather Service: https://www.weather.gov (accessed 13 May 2024).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jin, H.; Jin, F.; Wang, J.; Sun, W.; Dong, L. Competition and cooperation between shared bicycles and public transit: A case study of Beijing. *Sustainability* **2019**, *11*, 1323. [CrossRef]
2. Zheng, F.; Gu, F.; Zhang, W.; Guo, J. Is bicycle sharing an environmental practice? Evidence from a life cycle assessment based on behavioral surveys. *Sustainability* **2019**, *11*, 1550. [CrossRef]
3. Okraszewska, R.; Romanowska, A.; Wołek, M.; Oskarbski, J.; Birr, K.; Jamroz, K. Integration of a multilevel transport system model into sustainable urban mobility planning. *Sustainability* **2018**, *10*, 479. [CrossRef]
4. Fishman, E. Bikeshare: A review of recent literature. *Transp. Rev.* **2015**, *36*, 92–113. [CrossRef]
5. Lei, Y.; Zhang, J.; Ren, Z. A study on bicycle-sharing dispatching station site selection and planning based on multivariate data. *Sustainability* **2023**, *15*, 13112. [CrossRef]
6. Schuijbroek, J.; Hampshire, R.C.; Van Hoeve, W.J. Inventory rebalancing and vehicle routing in bike sharing systems. *Eur. J. Oper. Res.* **2017**, *257*, 992–1004. [CrossRef]
7. Caggiani, L.; Camporeale, R.; Ottomanelli, M.; Szeto, W.Y. A modeling framework for the dynamic management of free-floating bike-sharing systems. *Transp. Res. Part C Emerg. Technol.* **2018**, *87*, 159–182. [CrossRef]
8. Lei, C.; Ouyang, Y. Continuous approximation for demand balancing in solving large-scale one-commodity pickup and delivery problems. *Transp. Res. Part B Methodol.* **2018**, *109*, 90–109. [CrossRef]

9. Zhang, J.; Meng, M.; Wong, Y.D.; Ieromonachou, P.; Wang, D.Z. A data-driven dynamic repositioning model in bicycle-sharing systems. *Int. J. Prod. Econ.* **2021**, *231*, 107909. [CrossRef]
10. Yang, Z.; Hu, J.; Shu, Y.; Cheng, P.; Chen, J.; Moscibroda, T. Mobility modeling and prediction in bike-sharing systems. In Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services, Singapore, 26–30 June 2016; pp. 165–178.
11. Xu, C.; Ji, J.; Liu, P. The station-free sharing bike demand forecasting with a deep learning approach and large-scale datasets. *Transp. Res. Part C Emerg. Technol.* **2018**, *95*, 47–60. [CrossRef]
12. Lin, L.; He, Z.; Peeta, S. Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach. *Transp. Res. Part C Emerg. Technol.* **2018**, *97*, 258–276. [CrossRef]
13. Guo, R.; Jiang, Z.; Huang, J.; Tao, J.; Wang, C.; Li, J.; Chen, L. BikeNet: Accurate bike demand prediction using graph neural networks for station rebalancing. In Proceedings of the 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation, Leicester, UK, 19–23 August 2019; pp. 686–693.
14. Boufidis, N.; Nikiforiadis, A.; Chrysostomou, K.; Aifadopoulou, G. Development of a station-level demand prediction and visualization tool to support bike-sharing systems' operators. *Transp. Res. Procedia* **2020**, *47*, 51–58. [CrossRef]
15. Sathishkumar, V.E.; Cho, Y. Season wise bike sharing demand analysis using random forest algorithm. *Comput. Intell.* **2020**, *40*, e12287.
16. Yang, Y.; Heppenstall, A.; Turner, A.; Comber, A. Using graph structural information about flows to enhance short-term demand prediction in bike-sharing systems. *Comput. Environ. Urban Syst.* **2020**, *83*, 101521. [CrossRef]
17. Li, X.; Xu, Y.; Chen, Q.; Wang, L.; Zhang, X.; Shi, W. Short-term forecast of bicycle usage in bike sharing systems: A spatial-temporal memory network. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 10923–10934. [CrossRef]
18. Li, X.; Xu, Y.; Zhang, X.; Shi, W.; Yue, Y.; Li, Q. Improving short-term bike sharing demand forecast through an irregular convolutional neural network. *Transp. Res. Part C Emerg. Technol.* **2023**, *147*, 103984. [CrossRef]
19. Abouelela, M.; Lyu, C.; Antoniou, C. Exploring the potentials of open-source big data and machine learning in shared mobility fleet utilization prediction. *Data Sci. Transp.* **2023**, *5*, 5. [CrossRef]
20. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
21. Ding, C.; Wang, D.; Ma, X.; Li, H. Predicting short-term subway ridership and prioritizing its influential factors using gradient boosting decision trees. *Sustainability* **2016**, *8*, 1100. [CrossRef]
22. Gao, X.; Zhou, J.; Ci, Y.; Wu, L. An improved Prophet emergency traffic-flow prediction model. *Proc. Inst. Civ. Eng. Transp.* 2024; *in press*. [CrossRef]
23. Kim, D.; Shin, H.; Im, H.; Park, J. Factors influencing travel behaviors in bikesharing. In Proceedings of the Transportation Research Board 91st Annual Meeting, Washington, DC, USA, 22–26 January 2012.
24. El-Assi, W.; Mahmoud, M.S.; Habib, K.N. Effects of built environment and weather on bike sharing demand: A station level analysis of commercial bike sharing in Toronto. *Transportation* **2017**, *44*, 589–613. [CrossRef]
25. Mattson, J.; Godavarthy, R. Bike share in Fargo, North Dakota: Keys to success and factors affecting ridership. *Sustain. Cities Soc.* **2017**, *34*, 174–182. [CrossRef]
26. Yoon, T.; Cherry, C.R.; Jones, L.R. One-way and round-trip carsharing: A stated preference experiment in Beijing. *Transp. Res. Part D Transp. Environ.* **2017**, *53*, 102–114. [CrossRef]
27. Lin, P.; Weng, J.; Liang, Q.; Alivanistos, D.; Ma, S. Impact of weather conditions and built environment on public bikesharing trips in Beijing. *Netw. Spat. Econ.* **2018**, *20*, 1–17. [CrossRef]
28. Shen, Y.; Zhang, X.; Zhao, J. Understanding the usage of dockless bike sharing in Singapore. *Int. J. Sustain. Transp.* **2018**, *12*, 686–700. [CrossRef]
29. Durán-Rodas, D.; Chaniotakis, E.; Wulfhorst, G.; Antoniou, C. Open source data-driven method to identify most influencing spatiotemporal factors. An example of station-based bike sharing. In *Mapping the Travel Behavior Genome*; Elsevier: Amsterdam, The Netherlands, 2020; pp. 503–526.
30. Wessel, J. Using weather forecasts to forecast whether bikes are used. *Transp. Res. Part A Policy Pract.* **2020**, *138*, 537–559. [CrossRef]
31. Gammelli, D.; Peled, I.; Rodrigues, F.; Pacino, D.; Kurtaran, H.A.; Pereira, F.C. Estimating latent demand of shared mobility through censored Gaussian processes. *arXiv* **2020**, arXiv:2001.07402. [CrossRef]
32. Wang, J.; Miwa, T.; Morikawa, T. A demand truncation and migration poisson model for real demand inference in free-floating bike-sharing system. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 10525–10536. [CrossRef]
33. Younes, H.; Zou, Z.; Wu, J.; Baiocchi, G. Comparing the temporal determinants of dockless scooter-share and station-based bike-share in Washington, DC. *Transp. Res. Part A Policy Pract.* **2020**, *134*, 308–320. [CrossRef]
34. Cantelmo, G.; Kucharski, R.; Antoniou, C. Low-dimensional model for bike-sharing demand forecasting that explicitly accounts for weather data. *Transp. Res. Rec.* **2020**, *2674*, 132–144. [CrossRef]