

Forecasting Rental Demand for Bike-Sharing Systems

Wei Guo, Yi He, Yan Jin, Xiao Tan

Department of Industrial Systems and Engineering, University of Washington, Seattle, WA 98195, U.S.A.

Abstract: Bike-sharing programs offer an efficient and low-cost way to travel across the populated urban areas. To promote bicycle use in a bike-sharing system, a crucial component is to accurately forecast the potential demand of the bike rentals. In this paper, a Poisson-based regression model is first considered to estimate hourly bike rentals of the Capital Bikeshare system in Washington, D.C. Two state-of-the-art ensemble algorithms, random forest and gradient boosting machines (GBM), are also proposed as solutions for estimation. The results show that, as expected, random forest and GBMs outperform Poisson-based regression. Moreover, several characteristics of user behavior have been identified during the analysis of the fluctuations of bike usage over time. After identifying a set of predictor variables, a thorough comparison of the relationship between bike usage and the chosen predictors reveals the likely factors that affect travel decisions of different user groups. This is verified through the rank of features that are spontaneously selected by importance in the ensemble methods.

1. Introduction

As a healthy, sustainable and environmentally friendly form of public transportation, bike-sharing is becoming increasingly popular around the world in urban environments. There are currently over 500 bike-sharing programs worldwide with a total of more than 500,000 bikes in use [1]. In Seattle, the Pronto Cycle Share program was launched on October 13, 2014, featuring 500 bikes at 50 stations and rapidly gained popularity. These bike-sharing systems employ a distributed network of kiosks for registered or casual users to rent a bike from or return a rented bike to. A particular convenience for users of the program is that a bike need not be returned to the same kiosk from which it was rented, allowing for one-way bike rental. The kiosks collect a large assortment of bike usage data automatically, including member type, duration of travel, departure and arrival locations. These explicitly recorded data provide analysts with a means of estimating a city's non-automobile mobility.

With this attractive feature, an extensive interest has arisen within the research community in understanding how bike-sharing systems are used and what factors impact travel behavior. A common concern in bike-sharing systems is the unbalanced distribution of docking stations, i.e., certain stations have higher traffic flows than others. As a result, these stations will be quickly filled or emptied, especially during peak periods, preventing other users from picking up or delivering bikes. Therefore, optimizing the locations of the stations or finding the functional area in a city have become an endeavor for researchers.

Studies on this subject can be divided into two primary categories: prediction and clustering. The goal of prediction is to develop models which can estimate the occupancy of the stations over time while clustering concentrates on identifying dynamic patterns in bike-sharing usage by partitioning the stations into different concentrated groups with respect to their usage profiles. The prediction problem was first studied by Froehlich et al. [2]. Through analyzing the station data collected by Barcelona’s Bicing system, Froehlich et al. forecast the short-term availability of bikes at each station using Bayesian networks. The same problem is addressed by Kaltenbrunner et al. [3] with a time series model. Vogel et al. [4] leverage the power of data mining techniques in a more complicated time series analysis to predict daily and hourly rentals by taking weather patterns into account.

Meanwhile, research from other domains has also emerged towards enhancing accessibility from bike stations. From an operations research perspective, Lin et al. [5] address the problem of determining an adequate number and locations of bike stations as a facility location problem. A nonlinear mixed-integer model is accordingly formulated and tested with artificial bike demand data. Conrad et al. [6] present a mathematical modeling approach to schedule vehicle routes so as to balance the network, where Dantzig-Wolfe decomposition and Benders’ decomposition schemes are applied to large-scale instances. García-Palomares et al. [7] propose a Geographic Information System (GIS)-based method to calculate the spatial distribution of the potential demand for trips, locate stations using location-allocation models and determine station capacity.

However, our work differs from the aforementioned studies. In this paper, our objective is not balancing a dynamic bike-sharing system, rather we mainly focus on forecasting accurate bike rental demand for the given dates based on historical patterns and weather data. As a regression problem, several regression approaches are undertaken and compared in this study.

Before building the predictive model, an exploratory data analysis is firstly performed. Not only does it aid to the selection of the features for the ensuing regression, but also explores different user behaviors in the rental demand.

Since the observed data are counts, our first attempt is Poisson regression, yet over-dispersion is detected. Being a reasonable alternative, a negative binomial model is then employed to predict the rental counts. During the fitting process, principal component analysis is also taken into account since a reduction in dimensionality can serve as a regularization to prevent overfitting. Nevertheless, the estimation results are still undesirable, which leads us to seek more sophisticated techniques to capture the complexity of the count model. Based on our previous knowledge of machine learning, two prevailing methods, random forests and GBMs are chosen as candidate models, and the prediction results are greatly improved.

The rest of the paper is organized as follows: Section 2 gives a detailed data description and a simple discussion of model evaluation, followed by a summary of the framework of the proposed methods. In

Section 3, a data exploratory analysis is conducted. Several determinants in the data set are identified. Concurrently, different user behaviors suggest that two models should be trained separately for the prediction of total counts. Estimation results from different models are presented and compared in Section 4. Finally, we conclude the paper with remarks and future research topics.

2. Methodology

In this section, we begin with the description of the data set and outline the Poisson-based regression models. In addition, we briefly review basic methodology and learning algorithms of random forests and GBMs. For a more comprehensive review, we refer the reader to [8] and [9].

2.1. Data description

The bike-share data set in our paper was provided by Hadi Fanaee-Tork [10] and hosted by the UCI machine learning repository. It contains 17,379 records of hourly bike rental counts of the Capital Bikeshare system in Washington, D.C. from 2011 to 2012. The corresponding input variables includes both time- and weather-related information. Values for the former subset are season, year, month, day, hour, whether or not the rental occurred on a holiday, and whether or not the rental occurred on a working day. The weather-related information consists of temperature, “feels-like” temperature, humidity, and wind speed on an hourly basis. A detailed description of the variables are shown below.

Table I: Characteristics of Variables

| Field | Data Type | Description |
|------------|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| dteday | date | date from 01/01/2011 to 12/31/2012 |
| season | categorical | 1 = spring, 2 = summer, 3 = fall, 4 = winter |
| yr | categorical | 0 = year 2011, 1 = year 2012 |
| mnth | categorical | month, 1-12 |
| weekday | categorical | day of the week; 1-6 = Monday-Saturday, 0 = Sunday |
| hr | categorical | hour, 0-23 |
| holiday | categorical | 0 = not a holiday, 1 = holiday |
| workingday | categorical | 0 = not a working day, 1 = working day |
| weathersit | categorical | 1 = clear, few clouds, partly cloudy, 2 = mist+cloudy, mist+broken clouds, mist+few clouds, mist, 3 = light snow, light rain+thunderstorm+scattered clouds, light rain+scattered clouds, 4 = heavy rain+ice pellets+thunderstorm+mist, snow+fog |
| temp | continuous | normalized temperature in Celsius; the values are divided by 41 (max) |
| atemp | continuous | normalized “feels like” temperature in Celsius; the values are divided by 50 (max) |
| hum | integer | normalized humidity; the values are divided by 100 (max) |
| windspeed | continuous | normalized wind speed; the values are divided by 67 (max) |
| casual | integer | number of non-registered user rentals |
| registered | integer | number of registered user rentals |
| cnt | integer | number of total rentals |

We randomly split the entire data set into a training set and a test set in 2:1 ratio, and employ the models trained on the training set to predict the number of bike rentals in the test set. We use ten-fold cross-validation to quantify the accuracy of our models. The training data is thereby arbitrarily partitioned into ten equal size subsets, where nine subsets are treated for training and one subset is retained for validation. The process is then repeated ten times with each of the ten subsets used exactly once as the validation data, and the output represents the average of all the runs.

2.2. Model evaluation

Root mean square error (RMSE), along with mean squared error (MSE), is one of the most frequently used criteria for regression problems. RMSE is defined by

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2} \quad (1)$$

where \hat{y}_i is the predicted value, y_i is the true value and n is the total number of observations in the data set. The reason why RMSE or MSE are favorable is not only that they integrate the variance and the bias of the estimator, but also that they penalize large errors severely due to the squaring process. In our problem, since we adopt Poisson-based regression model as our predictive model, we calculate \hat{y}_i and y_i on the log scale. Therefore, we adjust the evaluation metric to be root mean squared logarithmic error (RMSLE), as shown below,

$$\text{RMSLE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (\log(\hat{y}_i + 1) - \log(y_i + 1))^2} \quad (2)$$

where “+1” simply ensures the value applied to logarithmic function is always positive. In fact, the introduction of logarithm function balances the weight of small and large predictive errors for the test set. For example, suppose a test set has only one record with true value 1 and the predicted value 2, then $\text{RSME} = 1$ and $\text{RSMLE} = 0.41$. However, if the predicted value is 20, then $\text{RSME} = 19$ while $\text{RSMLE} = 2.35$. It can be seen that RSMLE is less affected by the large predictive errors than RMSE . Since large predictive errors often stem from outliers in the test set, RSMLE is more justified to measure the performance of the predictive models.

2.3. Poisson-based regression

For count data, Poisson regression model is commonly employed by statisticians. It is similar to logistic regression in that both are in the class of generalized linear models (GLM). Considering the characteristic of dependent variable ‘cnt’ (count of total rental bikes including both casual and registered counts), intuitively we considered to use Poisson regression model to predict. A random variable X is

said to follow a Poisson distribution with parameter μ if it takes integer values $x = 0, 1, 2, \dots$, with probability

$$P(X = x) = (e^{-\mu} \mu^x) / x!$$

for $\mu > 0$. The mean and variance of this distribution can be shown to be

$$E(X) = \text{var}(X) = \mu$$

The regression model for μ is therefore

$$\log(\mu) = \beta_0 + \beta_1 X_1$$

2.3.1. Negative Binomial regression

In negative binomial regression, since $\sigma^2 > \mu$, we can add an error term, ε , which is independently distributed. And expected value of ε is a gamma-distributed error term that accounts for over-dispersion. Then we get

$$\log(\mu) = \beta_0 + \beta_1 X_1 + \varepsilon$$

2.4. Improved model

Ensembles of regression trees have been the most effective general-purpose algorithms in practice in the last decade. The aim of ensemble methods is to combine the predictions of several base estimators built with a given learning algorithm in order to improve the predictive capability over a single estimator. There are two families of ensemble methods: *boosting methods* and *averaging methods*. Base estimators in boosting methods are built sequentially to reduce the bias of the combined estimator. The idea is to combine several weak models to generate a powerful ensemble. By contrast, the motivation of averaging methods is to build several estimators independently and then average their predictions. Usually, a combined estimator is better than any single base estimator due to the reduction of its variance.

Inspired by the considerable success of ensemble methods across a wide variety of applications, we apply GBM and random forest techniques, the typical examples of boosting methods and averaging methods respectively, for predictive analytics and evaluate the regression models.

2.4.1. Random forest

Random forests [11] is an ensemble learning technique which has most of its application in classification and regression. In regression, bagging or bootstrap aggregation can be used to reduce the variance of an estimated prediction. Compared to bagging, Random forests generally have better performance in accuracy and requires little tuning [11]. Random forests creates multitude of de-correlated trees and take the mean of the individual tree. It distinguishes with bagging in the way of selecting the leaves i.e. process of feature bagging, which correct the correlation of trees in original bagging.

The basic idea of Random forest is as follows [11]:

1. Draw a bootstrap from the training data.
2. Select m variables from the terminal nodes randomly and pick up the best variable. Split it into two nodes.
3. Repeat (1) and (2) until the minimum node size is reached. After the tree is created, we can calculate the mean of the individual trees to make a prediction.

2.4.2. Gradient boosting machine

The gradient boosting machine (GBM) is a popular machine learning technique for regression problem, which ensembles a bunch of weak learners, such as decision trees. GBM is boosting method, which differs from random forest that is based on bagging idea, and generally boosting method could provide more accurate results, while bagging method is easier to be deployed in distributed system since the individual learner is not relevant to others.

In GBM, the learning procedure consecutively fits new models to provide a more accurate estimate of the response variable. The principle idea behind this algorithm is to construct the new base-learners to be maximally correlated with the negative gradient of the loss function, associated with the whole ensemble [9]. GBM, on the other hand, has high flexibility so that it is compatible with various application through the choice of loss function. For example, choose L_2 norm for continuous response, binomial loss for categorical loss, etc, and loss function can be even customized by user.

The algorithm of GBM is a iterative procedure. At t -th iteration, the output is defined as

$$f_t = f_{t-1} + \rho_t h(x, \theta_t)$$

While ρ_t and θ_t are estimated again at each iteration through following simple example,

$$(\rho_t, \theta_t) = \arg \min \sum_{i=1}^N [-g_t(x_i) + \rho h(x_i, \theta)]^2$$

where $g_t(x)$ is the negative gradient along the observed data. Generally, the exact form of the derived algorithm with all the corresponding formulas will depend on the design of the choices of loss function and $h(x, \theta)$ [9].

3. Data exploration

Before we start modeling, we need to understand the correlation between the count variable and the input variables (features) as well as the correlation between the input variables, which can be achieved by

data visualization tools. These tools also allow us to detect the outliers and identify different behaviors between register users and casual users in rental demand.

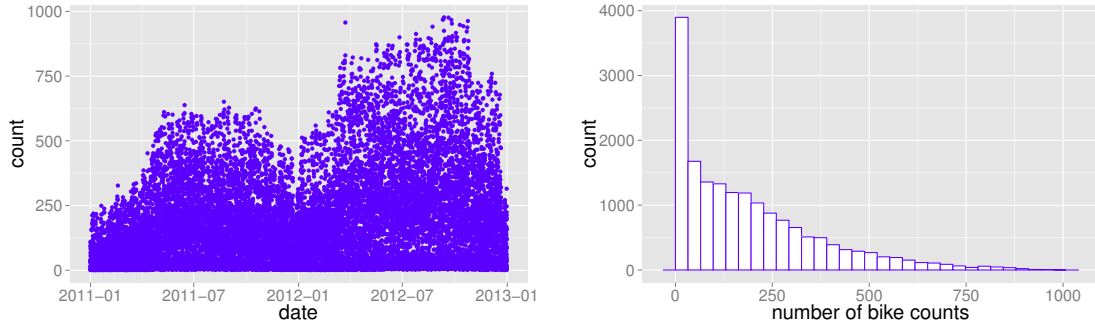


Figure 1: Distribution of total count variable.

An overall look at our variable of interest indicates that the total bike count fluctuates with time, but generally has an increasing trend. The histogram shows that the count variable basically follows a Poisson distribution, which is an essential premise of Poisson-based regression.

An analysis of correlation between the numeric attributes is also implemented. As can be seen in the heatmap, temperature is positively correlated with rental counts. Likewise, humidity is negatively correlated with all three dependent variables in this data set. Interestingly, holiday and weekday display virtually no correlation with bike demand. Also, we notice that season and month are highly correlated, and similarly temperature and “feels like” temperature. Thus we discard one of each pair of variables to avoid multicollinearity. Moreover, we find that the travel decision of registered users is less affected by the temperature and humidity compared to the casual users. Additionally, workingday is negatively correlated with the casual users whereas positively correlated with the registered users.

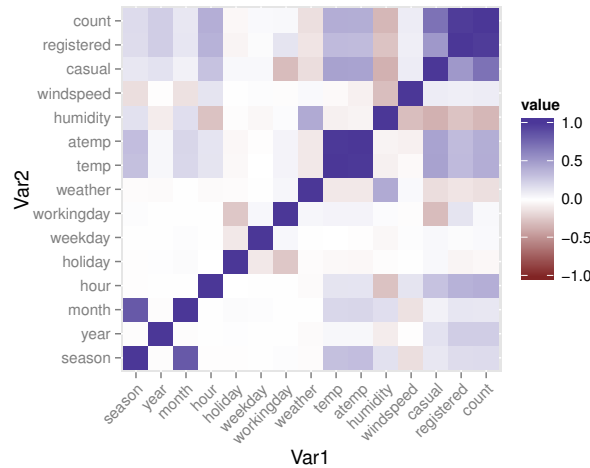


Figure 2: Correlation heatmap of all the variables.

3.1. User behavior analysis

The correlation analysis leads us to believe that registered and casual users might behave differently in the rental demand and the plots below just verify our conjecture.

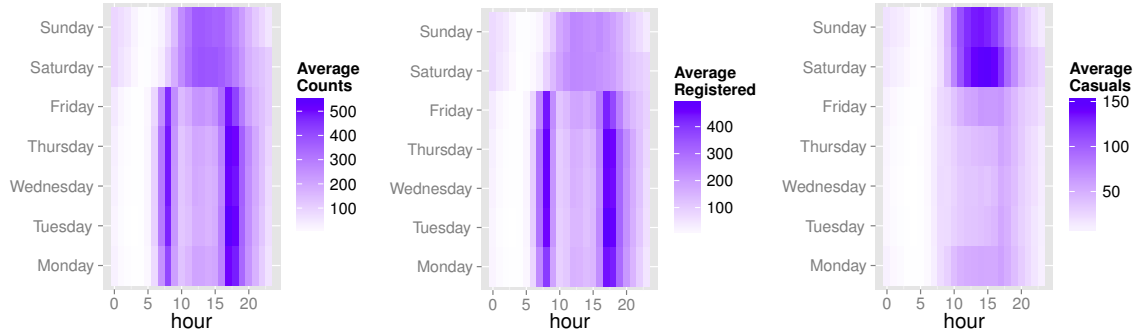


Figure 3: Average total counts, average registered counts, average casual counts vs Hour & Day of week.

It is intriguing to observe that the two user groups seem to come from separate distributions. The registered users appear to come from a bimodal distribution peaked at some point early morning around 8:00 am and some point late afternoon around 5:00 pm, both corresponding to typical occupational commutes. On the other hand, the casual users seem to come from a single peaked distribution. If we compare the bike rentals during the weekend, it can be seen that the casual users rent bikes more frequently than on weekdays, while there is a drastic drop in bikes rentals from the registered users.

These plots not only imply that hour and workingday are two critical determinants in users' travel behaviors, but also suggest that we should train two separate models, one for each of the 'casual' and 'registered' users, then combine them together for a more accurate prediction of the total counts.

4. Estimation results

4.1. Poisson Regression, NB model and PCA

4.1.1. Poisson Regression

Here we include all variables except date (intuitively single specific date is meaningless for the prediction, and actually we have tried to use date as one of the regressors, only finding it did not work, and at the same time, we have variables representing holiday and workday, so we decide to omit this variable) and also exclude casual count and registered count that are intermediate dependent variables. And also we factorize variable of season, year, month, holiday, weekday, workingday and weather.

After the modeling we calculate the RMSLE to evaluate. Here we get $\text{RMSLE} = 1.734$. During doing the prediction using Poisson regression model, a warning message is shown. From related documentation we find possible reason of this warning: there might be some collinear covariates. But which pair(s) of

variables should count for this problem? From the scatter plot of variable pairs and common sense, it seems like the pairs of temperature and feeling temperature, season and month, holiday, weekday and workingday all might lead to multicollinearity, but we are not very sure by now. Also we check the assumption that the dependent variable 'cnt' follows Poisson distribution by calculating its mean and variance, which are 188.49 and 32978. Apparently there are over-dispersion problem in this case, then we have to utilize negative binomial model instead.

4.1.2. Original negative binomial regression and revisions

Here we still include all variables except date and casual and registered count. Also we calculate the fitted values for test set, in which we still get the same warning message as previous case. And this time $\text{RMSLE} = 1.119$. It seems like the NB model does improve the model fitting, but the problem of multicollinearity have not been addressed.

Afterward we try series of revisions on the original NB model to improve the fitting including omitting feeling temperature, temperature, windspeed, season, month, holiday, weekday or workingday, also try to omit various combinations of these regressors. When we omit holiday and weekday, only keeping workingday, the warning of collinearity is eliminated, from which we might conclude these three regressors should be highly correlated. All above attempts don not improve the model significantly, only getting RMSLE between 1.120 and 1.129. Then we must try other ways to improve.

4.1.3. Separated prediction of casual count and registered count

In this phase we try to predict casual count and registered count separately, afterward combine them together to predict the total count. Since in the way information provided by regressors might be utilized more than original negative binomial model, we expect more improvement will be available.

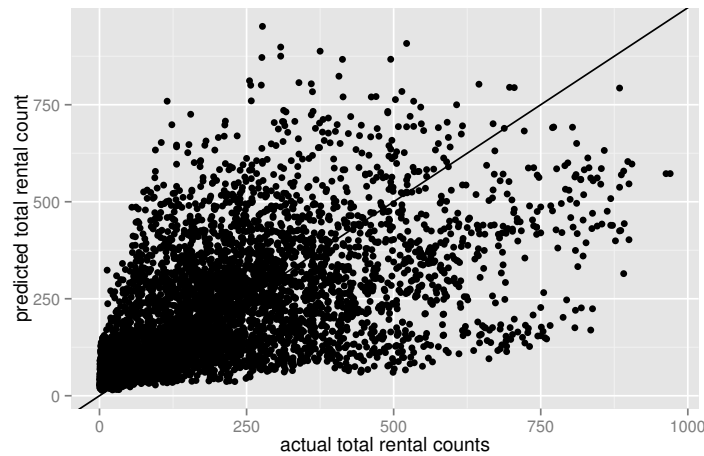


Figure 4: NB Result, Predicted Value vs. Actual Value

Here also we base on the Revised NB model-7 (omitted holiday and weekday, and only keep work-

ingday). And we get $\text{RMSLE} = 1.119$, which is almost same with the original NB model. Also we revise this separated-prediction model by omitting feeling temperature, then we get the best outcome using NB regression model, $\text{RMSLE} = 1.107$, which is the best fitting so far. We also try omitting temperature instead of feeling temperature and other ways to improve model fitting, but don not get satisfied outcome. And Figure 4 also shows the bad performance of prediction, since the black dots are not along the best fitted line $y = x$, especially when the actual total rental count is larger.

On the other hand, through Figure 5, it shows that the residual is relatively acceptable when the response value is small, while it is distributed sparsely when the response is big. From the best fitting of negative binomial model, especially the plot of the residual against the predicted total count, we can figure out there might be a condition of heteroscedasticity in our case. Then we should seek more appropriate model for prediction of total count.

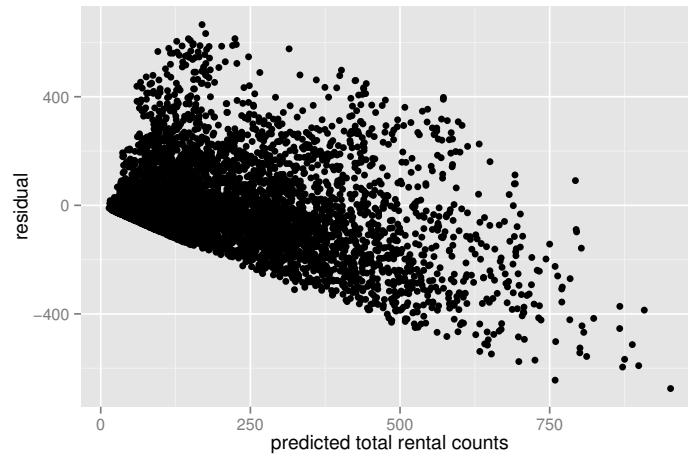


Figure 5: NB Residual Plot

4.1.4. PCA

At last, we try Principal Components Analysis (PCA) to reduce the number of predictors. According to the principle of choosing components, we choose 3 component, however, it only explains 55% variations. With these generated components, the RMSLE of NB model is 1.232, which is not improved comparing to the previous one. We also try non rotation and promax, there is no significant improvement.

4.2. Random Forest and GBM

4.2.1. Analysis of Random Forest

Our problem of predicting the counts in the bicycle sharing system can be handled by the method of Random forests. In this application, we randomly divide the original data set into two sets. The training data set contains 11,844 samples and the test data contains 5,537 samples. By importing the package of `randomForest` in R, we could do the regression on our data sets. As we did in the previous work, we

separate the total counts into the counts of registered and counts of casual. By creating two models for registered and casual counts respectively, we eliminated the errors produced by the differences between casual group and registered group. After applying the `tuneRF()` method to search the optimal number of variables in each split and running the algorithm several times to determine the best parameters of the number of trees, we perform the tuned Random forests model on our test data set and evaluate the outcomes by RMSLE. The results are given in Table II.

The model for registered model has better performance than the model for casual. This might be caused by that the randomness in the casual rent is greater than in registered rent. We notice that the total RMSLE is less than both the Registered RMSLE and the Casual RMSLE. This might be caused by the biases in the registered model and the casual model.

We introduce Figure 6 to shows the predictions of the model against the actual values. We can see from the graph that the points are distributed along $y = x$ and there is no sign of the heteroscedasticity

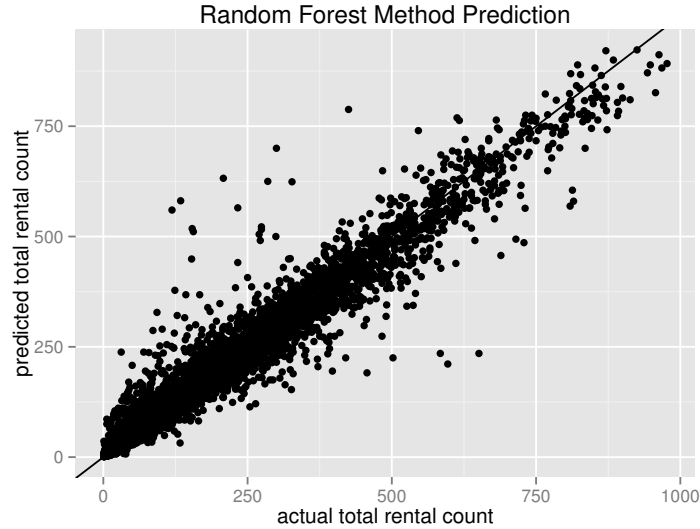


Figure 6: RF Result, Predicted Value vs. Actual Value

Figure 7 present the importance of the variables in our models. Both model have hours, years and workingday the most important variables while the variable of holiday is the least important variable. We also notice that the variables related to the weather (atemp, temp, windspeed) is less important in the registered model than the casual model. The reason may be that people who register bicycles in advance tend to pay less attention on the weather condition.

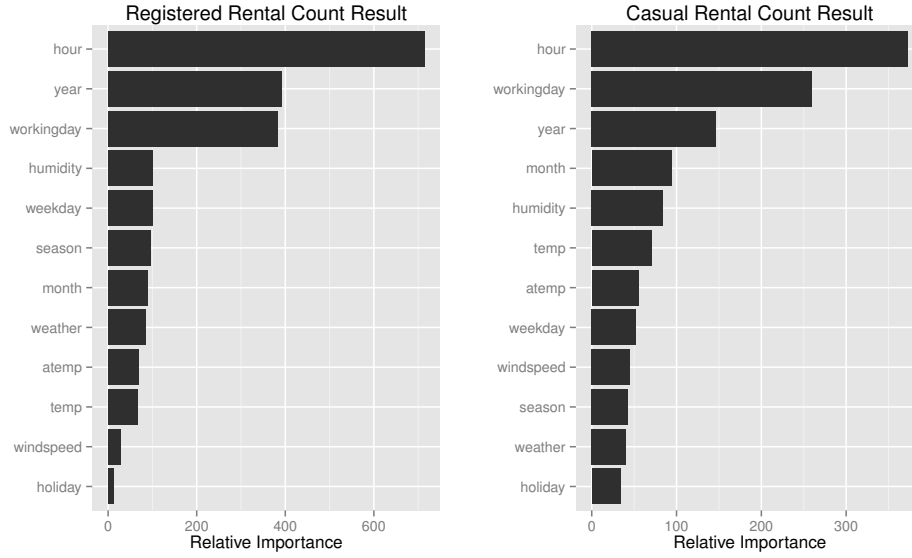


Figure 7: Relative Importance Comparison of Random Forest

Random forests have generally prominent performance compared to previous methods. The amount of trees in the model can affect the accuracy of the model. The model with 500 trees have the accuracy improved compared to 100 trees. But this does not mean that to increase the amount of trees will always be a good choice. The improvement of accuracy will have its limitation and we should always try to find the balance between fitness and computation complexity. On the other hand, the model with all the variables as predictors has better performance than the one with less variables. We could conclude that the Random forests method is reliable to avoid overfitting.

4.2.2. Analysis of GBM

To prove the high accuracy of GBM, we apply other popular machine learning techniques and compare their obtained performances. The chosen methods are the Poisson regression, negative binomial model and random forest. The optimal hyper-parameters for random forest models are chosen by the 10 fold cross validation applied to the grid search. On the other hand, not only the total rental count RMSLE is computed, the registered RMSLE and causal RMSLE are computed as well for sake of better interpretation. The algorithm accuracy comparisons are given in Table II.

Table II: Machine Learning Algorithm Accuracy Comparison

| Method | Registered RMSLE | Causal RMSLE | Total RMSLE |
|--------------------|------------------|--------------|-------------|
| Poisson Regression | 1.1723 | 1.0504 | 1.1603 |
| Negative Binomial | 1.1246 | 0.9925 | 1.1200 |
| Random Forest | 0.3299 | 0.5281 | 0.3289 |
| Gradient Boosting | 0.3995 | 0.5767 | 0.4034 |

Figure 8 shows the predictions of our model against the actual values. We can see that the model appears to capture the general trends in the data. Especially no matter high or low predicted count values, the prediction accuracy is always similar because they are all along the $y = x$ benchmark line.

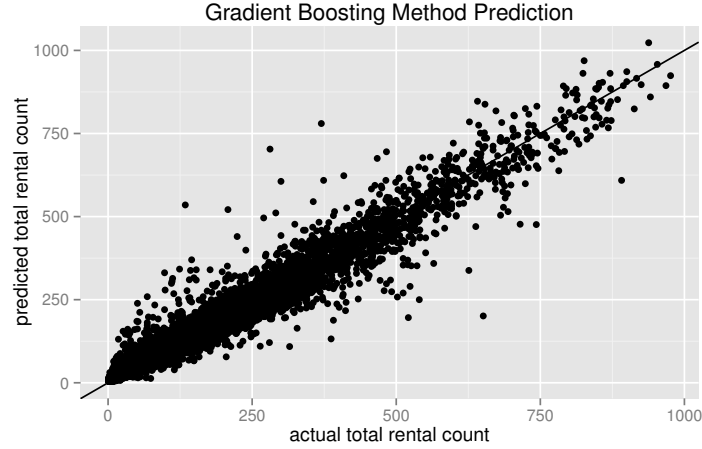


Figure 8: GBM Result, Predicted Value vs. Actual Value

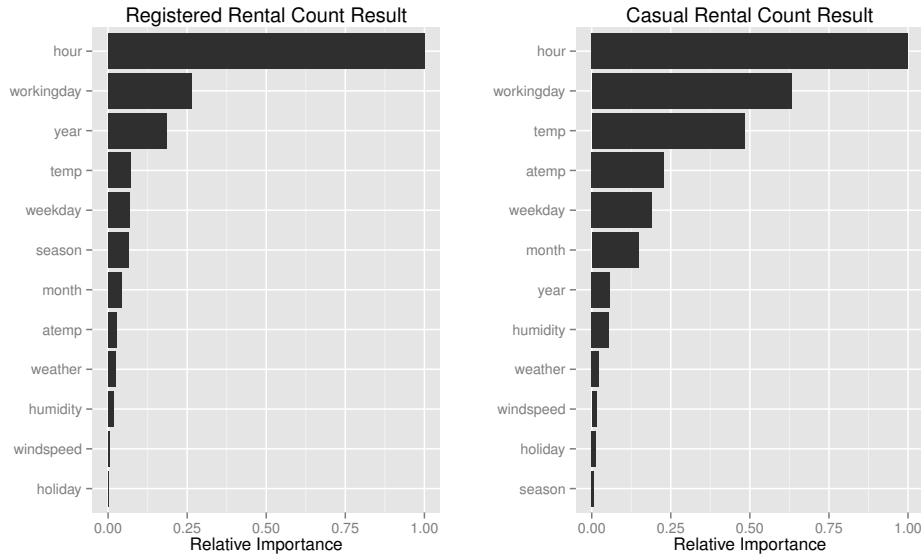


Figure 9: Relative Importance Comparison of GBM

To investigate the those significant variables that contribute much to this model, figure 9 is shown as follows. Hour and workingday are top 2 important on the prediction in both registered count model and causal count model. However, there are interesting differences when checking other features between these two models. For registered model, feature year is more important than temp that is the third

significant variable in causal model. This makes sense, since when we make a travel plan and book a vehicle, it is less likely to consider the temperature, in comparison, casually we would rent a vehicle depending on the temperature at that time. The effect of the other predictors appears to be low in comparison and may offer some room for feature selection or transformation for model improvement.

5. Conclusion

In this project, we try to predict the total number of rental vehicles based on a bunch of environmental and date variables. To make better prediction, instead of predicting the total count, we consider the registered count and casual count as our intermediate response variables. And several different prediction models are applied, they are Poisson regression, negative binomial model, random forest and gradient boosting machine.

For count model, we firstly introduce the Poisson regression method and discover that the model does not fit the data basically. The Negative binomial method has improved the performance compared to the Poisson regression method, but the accuracy is still not acceptable. We attain the best result by the NB method with the total RMSLE of 1.107. With the conclusions from our models, we could claim that the primitive regression model is not capable to handle such a complicated problem in real life. The real-life data set usually has many variables which are intertwined with each other. The Poisson regression and Negative binomial method have too many constraints on the data which cannot be satisfied by the real-life data.

For Random forests model, we still separately build the models based on different dependent variables. From the process of tuning the algorithm, we notice that the performance of the algorithm will be affected by the number of trees and number of variables in each split. The tuned model has a significant improvement compared to the count models with the total RMSLE of 0.3289. From the result produced by the Random forests, we notice that the predictors related to time have generally greater impact on the results than the predictors about weather. Especially, different hours in the day will significantly influence the demand forecasting. In other word, the demand in the peak hours will decide how well the bike-sharing system would perform. The more specific future research on the demand trends based on hours is required if we want to make the perfect balance on the system. In addition, we also notice that the predictor, year, have a clear impact on the model even though this variable is a binary random variable. It clearly shows the trends that the bike-sharing system received much more popularity in the year of 2012 than 2011. The Random forests and Gradient boosting methods have different approaches to predict the data with the Poisson regression and Negative binomial methods. They have less assumptions about the data and could ?learn? from the data. The performance has significantly improved by these learning methods. In our problem of forecasting demands in the bike-sharing system, the Random

forest and Gradient boosting methods could be applied to produce the required results. The research led us to construct the models with relative high accuracy in the forecasting problem. However, future study is needed to figure out how each predictor contributes to the forecasting and how to optimize the bike-sharing system in detail.

In the next step, time series analysis would be incorporated into this analysis. Since our dataset is a time series dataset, we could utilize the time series information to improve the precision of the prediction. For example, we could add the difference between values at consecutive days into the set of regressors, and then fit the models as we already use in the previous. The result should be better.

References

- [1] J. Larsen, “Bike-sharing programs hit the streets in over 500 cities worldwide,” *Earth Policy Institute*, vol. 25, 2013.
- [2] J. Froehlich, J. Neumann, and N. Oliver, “Sensing and predicting the pulse of the city through shared bicycling,” in *IJCAI*, vol. 9, 2009, pp. 1420–1426.
- [3] A. Kaltenbrunner, R. Meza, J. Grivolla, J. Codina, and R. Banchs, “Urban cycles and mobility patterns: Exploring and predicting trends in a bicycle-based public transport system,” *Pervasive and Mobile Computing*, vol. 6, no. 4, pp. 455–466, 2010.
- [4] P. Vogel and D. C. Mattfeld, “Strategic and operational planning of bike-sharing systems by data mining—a case study,” in *Computational Logistics*. Springer, 2011, pp. 127–141.
- [5] J.-R. Lin and T.-H. Yang, “Strategic design of public bicycle sharing systems with service level constraints,” *Transportation research part E: logistics and transportation review*, vol. 47, no. 2, pp. 284–294, 2011.
- [6] C. Contardo, C. Morency, and L.-M. Rousseau, *Balancing a dynamic public bike-sharing system*. CIRRELT, 2012, vol. 4.
- [7] J. C. García-Palomares, J. Gutiérrez, and M. Latorre, “Optimizing the location of stations in bike-sharing programs: a gis approach,” *Applied Geography*, vol. 35, no. 1, pp. 235–246, 2012.
- [8] A. Criminisi, J. Shotton, and E. Konukoglu, “Decision forests for classification, regression, density estimation, manifold learning and semi-supervised learning,” *Microsoft Research Cambridge, Tech. Rep. MSRTR-2011-114*, vol. 5, no. 6, p. 12, 2011.
- [9] A. Natekin and A. Knoll, “Gradient boosting machines, a tutorial,” *Frontiers in Neurorobotics*, vol. 7, no. December, 2013.
- [10] H. Fanaee-T and J. Gama, “Event labeling combining ensemble detectors and background knowledge,” *Progress in Artificial Intelligence*, vol. 2, no. 2-3, pp. 113–127, 2014.
- [11] L. Breiman, “Random forests,” *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.