Bike-Sharing Data Analysis: Prediction of Daily Bike Rental Counts Based on Multiple Linear Regression

Final Project Report · MA 575 Fall 2021 · C3 · Team #2

Ali Taqi, Hsin-Chang Lin, Huiru Yang, Ryan Mahoney, Yulin Li

12/10/2021

In this project, the following question is to be answered: If we have the past history of bike rental counts as well as records of environmental and seasonal conditions, how and how well could we predict the bike rental counts in the future? In this project, such questions are approached by predictive modeling of daily bike rental counts from a 2011-2012 Bike Sharing dataset [1]. The daily bike rental counts are predicted with models based on Multiple Linear Regression (MLS) using the environmental and seasonal variables as predictors. The initial goal of this project is to train the model using only the 2011 data, and then validate the prediction power of the model on the 2012 data. Given the limited time span of available training data, issues are found in the validation process using the 2012 data; the impact of user base on the future predictions is brought to our attention. The initial models are then revisited and corrected to account for the effect of user base. The refined models are expected to have better prediction powers than the initial MLS models, but a full validation would require further availability of bike rental data.

1 Introduction

Bike sharing has become a world-wide phenomenon. Optimization of inventories and dynamic reallocation of bike-sharing resources are of growing interests from both a business and an environmental point of view. Both of these tasks require accurate predictions of bike rental behaviors at least on the daily level.

(further motivates & applications?)

In this project, we strive to answer the following question:

- If we have the past history of bike rental counts as well as records of environmental and seasonal conditions, how and how well could we predict the bike rental counts in the future?
- In particular, how and how well could we predict for the next whole year, and what about for the next few days?

Such questions are approached by predictive modeling of daily bike rental counts from a 2011-2012 Bike Sharing dataset [1]. The modeling approach is based on Multiple Linear Regression (MLS), and the daily bike rental counts are predicted using the environmental variables (e.g., weather conditions) and seasonal variables (e.g., holiday schedules) as predictors.

2 Background

The aim of this project is to achieve the best model(s) that can be obtained from past data for the use of predictions for the future, preferably predictions one year ahead. To validate the prediction power of models under this setting, the basic goal of this project is to train all models using only the 2011 data, and then test them on the 2012 data.

The response variable to be predicted is the **daily** bike rental count. In the dataset being studied [1], the following 3 types of bike rental counts are recorded:

1. the count of bike rentals by **casual** users
2. the count of bike rentals by **registered** users
3. the **total** count, which is the sum of casual count and registered count.

Two main types of predictors are included in the dataset, the environmental ones and the seasonal ones:

1. environmental variables

(Table 1: A sample of the data - variable names, meanings, units, sample values)

2. seasonal variables

(Table 2: A sample of the data - variable names, meanings, units, sample values)

3 Modeling & Analysis

3.1 Pre-processing

3.1.1 Type Conversion

To be noticed, the value of categorical variables indicates type labels and has very limited physical meaning in the magnitude of those values, which thus cannot be used in the same way as the numeric variables in MLS models. The categorical variables therefore needs to be recognized before the actual modeling process and to be carefully handled.

The below variables are interpreted as Boolean variables and are transformed into `logical`-type variables in `R`:

- `holiday` (holiday or not)
- `workingday` (working day or not)

The below variables are interpreted as categorical variables and are transformed into `factor`-type variables in `R`:

- `season` (season, from 1 to 4)
- `yr` (year, from 0 to 1)
- `mnth` (month, from 1 to 12)
- `weekday` (weekday, from 0 to 6)
- `weathersit` (weather type, from 1 to 4)

3.1.2 Value Conversion

The recorded values of `temp` (measured temperature), `atemp` (feeling temperature), `hum` (measured humidity) and `windspeed` (measured wind speed) in the data set being studied here are the normalized ones; all recorded values are the ones that have been divided by the maximum of measured values [1]. For example, the recorded values of `temp` (measured temperature) are obtained by dividing the original measured values by 41 (max) and are thus all less than or equal to 1.

In this project, these normalized records are scaled back to their original values for the sake of easier interpretations. For example, the recorded values of `temp` (measured temperature) are multiplied by 41 (max) in the pre-processing process, which recovers the original scale of temperatures in Celsius.

3.2 Variable Selection

3.2.1 Response Transformation

Notably, the behaviors of rental counts from different user types are considerably different.

1. **Patterns with weekdays** (see Figure 1,2): Over the time span of a week, the casual count usually reaches its minimum in the middle of a week (grey dots mostly) and its maximum on weekends (green dots mostly), while the registered count does the opposite.
2. **Patterns with temperatures** (see Figure 3): The casual count seems more linear in both the feeling and measured temperatures (`atemp` and `temp`), while the registered count seems to be (at least) quadratic.

We therefore expect that the registered counts and casual counts will follow different distributions and should thus be predicted by separate models. Furthermore, for the casual count, avoiding unnecessary higher other terms has the benefit of more stable computations and model structures. The prediction of total counts will then be obtained by adding the predicted registered counts and predicted casual counts together.

3.2.2 Predictor Selection

Given the predictive nature of modeling in the current problem setting, the predicted response is of greater interests than the actual value of the parameter estimates, as opposed to that in an inference task. This, to some degree, relaxes the constraint forbidding colinearity in the predictors, since colinearity will only lead to instability in the parameter estimates but not in the predictions; however, we should still seek to minimize colinearity at least in our beginning model, which would lead to clearer model structures as well as better interpretability of model statistics at the early stage of modeling, which could provide us clearer directions in the improvement process that follows.

With the above considerations in mind, the predictors in the beginning model are selected following the 2-step approach below:

1. The scatter plot matrix for the whole set of variables are plotted for the 2011 training dataset, and all predictors that seem to be significant, i.e., predictors with which the response variable (daily rental count, `cnt`) exhibits a notable visual pattern, are selected.
2. From the selected predictors above, all the highly correlated predictors are removed. Within a group of correlated predictors, only the one that has the largest correlation coefficient with the response variable as well as having the strongest causal relation with the response (in the intuitive sense) will be kept.

It is important that the investigation is done for all

predictors for the sake of minimal loss of information. Note that in practice, the whole set of predictors is divided into two groups, environmental and seasonal, and plotted separately, for better readability of the large scatter plot matrices. The separation is justified by the fact that most environmental variables, such as weathers, are expected to be independent of the seasonal variables, such as weekdays and holiday schedules.

At last, the above process leaves us with a small subset of the very core predictors for our beginning model: `weathersit`, `atemp` and `weekday`.

3.3 Initial Modeling

In the model building and selection process, we start from the simplest models, which have the minimal number of predictors all in the additive form, as the beginning models.

Beginning Models

1. For **total count**:

$$\text{cnt} \sim \text{wkngday} + \text{weathersit} + \text{atemp} + \text{atemp}^2$$

2. For **registered count**:

$$\text{reg} \sim \text{wkngday} + \text{weathersit} + \text{atemp} + \text{atemp}^2$$

3. For **casual count**:

$$\text{cas} \sim \text{wkngday} + \text{weathersit} + \text{atemp}$$

(Table 3, 4, 5: eval tables for total, registered and casual)

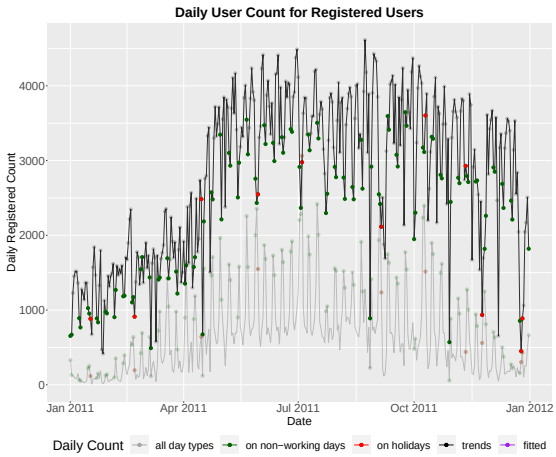
Model	rmse	n-rmse	% Error	cv-rmse
2011 cas	309.97	0.56	72.87	314.86
2011 reg	584.55	0.52	25.37	594.36
2011 tot	722.15	0.52	25.44	734.18

Table 1: Diagnostics for Model 1

In this process, the model statistics such as p-values are not relied on as much, because the colinearity issues worsen, which might weaken the significance of

Final Models

3.4 Diagonostic Analysis



interpretation for the final model as well as residual diagnostics

3.5 Validation and Problemshooting

3.6 Refined Model

3.6.1 Prediction of the Yearly Growth Ratio

The modeling is based on the assumption that the growth trend will remain the same in the future years as that in the year of 2011. Note that this is NOT saying that the user base is supposed to remain unchanged throughout the entire year; the fact that the same scaling factor works at all points in the entire year is due to the fact that the MLS model in the later part of the year, e.g., in fall and winter, are already trained to compensate for rental count growth due to user growth using the environmental and seasonal variables.

3.6.2 Prediction without the Yearly Growth Ratio

4 Prediction

4.1 Unadjusted Model

4.2 Refined Model

5 Discussion

Models for both long-term and short-term predictions are included.

To be noticed, at least one more year's data is needed for a final validation of the refined model, which is not available for the moment. This is to be left for the future work.

Time series

6 Appendix

6.1 Preprocessing

6.1.1 Type Conversion

(codes here)

6.1.2 Value Conversion

(codes here)

6.2 Variable Selection

6.2.1 Predictors Selection

6.2.2 Predictors Selection

6.2.3 Response Transformation

6.3 Initial Modeling

6.3.1 Beginning Model

6.3.2 Final Model

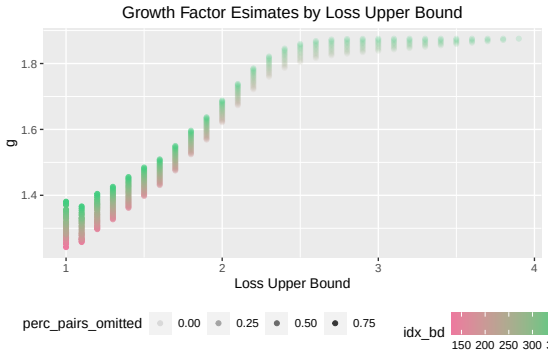
6.4 Diagonostic Analysis

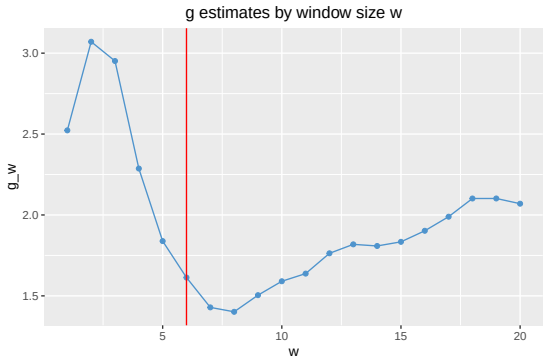
6.5 Validation and Problemshooting

6.6 Refined Model

6.6.1 Prediction of the Yearly Growth Ratio

```
##   idx_bd loss_bd        g    n
## 1    130     1.0 1.242667  521
## 2    138     1.0 1.252098  506
## 3    138     1.1 1.258080 3198
## 4    146     1.0 1.256074  497
## 5    146     1.1 1.260398 3149
## 6    146     1.2 1.297205 8044
```





6.6.2 Prediction without the Yearly Growth Ratio

6.7 Growth Rate Estimation using Environmental Loss Functions

Suppose we have two days (observations) d and d'. Consider the reduced dataset where the observations are solely the vector of our continous variables, in our case, we have three: `atemp`, `hum`, and `windspeed`. Also, let's use `holiday` to focus only on non-holidays. We can adjust for environmental factors but comparing two "environmentally equivalent" days without accounting for `holiday` may very well throw off our estimates of $\hat{g}$. Before we continue, note that we normalize the variables so the loss function is unit-agnostic. So, $d = (x_1, x_2, x_3)$ and $d' = (x'_1, x'_2, x'_3)$ where x_i represents the normalized value of predictor i.

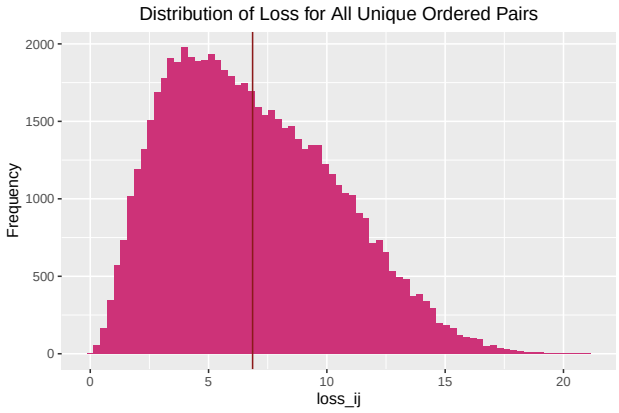
Furthermore, let $\vec{\alpha} = (\alpha_1, \alpha_2, \alpha_3)$ be a weight vector, more heavily weighing variables deemed more important in determining "environmental similarity". We deem this to be `atemp`. Now, consider the following loss function, which is simply the weighted difference of our normalized predictor variables.

$$\mathcal{L}_d(d') = |\vec{\delta}| \text{ where } \delta_i = \alpha_i(x_i - x'_i)$$

Now, suppose we enumerate all possible unique pairs of days, which we denote $\Pi = \{\pi_{ij} = (d_i, d_j) \mid i < j\}$. Note that we use the lower-triangle scheme, where we choose $i < j$ so that $i - j < 0$. This will prove useful later on since we want day i to preceed day j; this corresponds to the unique pairs found in the indices of a "lower-triangular matrix". In any case, we compute $\mathcal{L}(\pi_{ij})$ for every pair and obtain the set (which we call `df_loss`) $\mathcal{L}(\Pi) = \{l_{ij} = \mathcal{L}(\pi_{ij}) \mid i < j\}$.

Now, here is the magic! We are not interested in taking pairs π_{ij} which are not very environmentally similar. After all, our quest is to estimate growth between days where we in theory expect a similar amount

of users by marginalizing environmental factors. So, there comes a question, which pairs do we discount as days too disparate to be considered environmentally equivalent? Well, our good foresight in normalizing the continuous variables and weighing them properly (to the best of our abilities) means that we can consider the distribution of l_{ij}, which we see below:



A quick note: it should not be surprising that we obtain a distribution that resembles a χ^2 distribution. After all, since we assume our predictors are normal, their differences will be too; since our loss is essentially the norm of a multivariate normal, this explains our distribution shape! As such, since it is part of the exponential family, we expect the distribution to be approximtely normal, roughly speaking. So, since we have done a good job by normalizing our variables, a valid strategy to discount pairs/include them is to use a cutoff anywhere below the mean of our distribution μ. We could take for instance, $\mu - 1\sigma$ to be our cutoff for our upper loss bound B.

So, obtain our selected set of pairs of days, cutting off the maximum loss at the selected upper bound B. That way, we obtain the selected pair set by filtering Π with respect to the chosen loss upper bound B:

$$f_{loss}(\Pi, B) = \tilde{\Pi}_B = \{\pi_{ij} \mid l_{ij} = \mathcal{L}(\pi_{ij}) \leq B\}$$

Next, we also filter for what we call an "index difference bound". This is much less critical than the loss bound, as it is only impactful for low values. Namely, the idea is simple: we don't want to pick days too close to each other. We know days close to each other will have similar environmental conditions, so to avoid pesky autocorrelation, we impose a bound on how close the days are. Let $\rho := \min(i - j)$. In our lower-triangle matrix analogy (where we imagine indices as positions of matrix entries), ρ denotes which diagonal band we are considering. In any case, by imposing a lower bound on $i - j$, to pass the index difference filter, every pair must be atleast ρ days apart, or in other words, be at least ρ diagonal bands close to the main diagonal. In practice, we find low values of ρ to produce the most impactful filters, this makes sense, owing to the autocorrelation observed in days close to each other mentioned above.

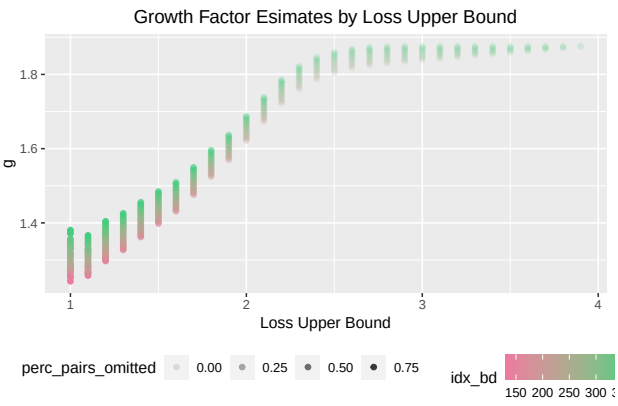
$$f_{idx\_diff}(\Pi, \rho) = \tilde{\Pi}_\rho = \{\pi_{ij} \mid i - j > \rho\}$$

Finally... we compute the estimate derived from the filtered pair set $\tilde{\Pi}$ as follows. Let c_i denote the count of bike users in day i, and c_j those in day j. **Since we have, in theory, marginalized out the environmental factors (and holidays), we may obtain a reasonable estimate for g by taking the observed growth observed between day i and day j, then taking the average over all the pairs.** Since by construction, we used lower-triangular indices, this means day j is always after day i. As such, we can imagine each pair, which has marginalized environmental factors, to contain true information about user-base growth since day j is roughly equivalent to day i in environmental factors and is in the future of day i! In other words, take the set of estimates for every $\pi_{ij} \in \tilde{\Pi}_{B,\rho}$ and $\hat{\gamma}_{ij} := \frac{c_j}{c_i}$:

$$\mathbb{E}[\{\gamma_{ij}\}] := \hat{g}$$

6.8 Estimate Performance over Bound Paramater Space

```
##   idx_bd loss_bd        g    n
## 1    130     1.0 1.242667  521
## 2    138     1.0 1.252098  506
## 3    138     1.1 1.258080 3198
## 4    146     1.0 1.256074  497
## 5    146     1.1 1.260398 3149
## 6    146     1.2 1.297205 8044
```



6.9 Window Technique