Machine Learning Final Report Spring 2014

Capital BikeShare Data Set Analysis and Prediction

Shreya Ghosh, Paul Landes, Nianzu Ma, Pavan Reddy

Contents

1 Introduction

Bike sharing systems have made renting bikes efficient and quick with memberships, multiple bike locations and easy rental and return process. Through these systems, a user is able to easily rent a bike from a particular station and return back at another station. The biggest issue bike sharing systems face is due to adverse effects of harsh weather conditions on biking. This paper shows how different weather conditions would affect bike usage. The data used here has log of bikes rented in different seasons, under different weather conditions and on weekdays or holidays. We use different regression and classification algorithms to make relevant predictions for bike sharing systems, which would help them deal with extreme conditions like very few bikes were rented or all of the bikes were rented.

1.1 Instructional Resources

The class machine learning assigned book[5] was used to formulate the theory behind the analysis and machine learning tasks.

1.2 Data

In recent years bikes have turned out to be an efficient alternate mode of transportation in some cities, which lead to growth of bike-sharing systems. These systems are quite convenient for users, since the user do not have to worry about parking, they have options where you can become a member and often these systems let first 30-60 minutes of ride free. Currently there are over 500 bike-sharing programs around the world which is composed of over 500 thousands bicycles[1].

The data set being used in this paper was created from trip logs kept by Capital Bikeshare (CaBi) in Washington, DC, where Capital Bikeshare is currently the largest in the nation with over 1,200 bicycles at 140 stations. The data set put together by The University of California at Irvine has features like date, weather, weekdays, holidays and count of bikes rented. These features of the data make it apt for research such as finding patterns between different features and number of bikes rented. Feature selection was in part chosen based on data manipulation[3].

The data set was created from three different sources. Bike trip logs were collected from Capital Bikeshare, merged with weather data from Freemeteo and a holiday schedule provided by The District of Columbia. The data set being used in this paper has hourly as well as daily usage data over the course of two years (2011 - 2012).

2 Problem Definition

The data has hourly and daily usage. In this paper we develop an understanding about how humidity, temperature, rainfall, snow and other weather conditions affect the number

of bikes being rented. We also determine bike rental patterns of different kind of users[1]. This analysis would help bike sharing companies to predict what number to bikes they need to stock on any given day.

3 Feature Selection

We used various methods for data processing and choosing features to build the classification and regression models.

The majority of the data instances of the hourly bike rental count are in the range of 0 - 400 as shown in figure 1.

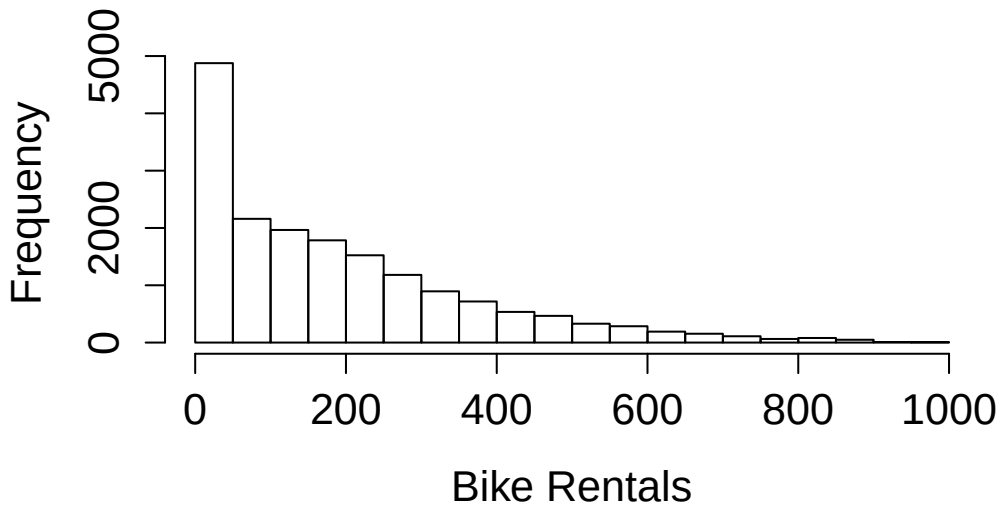

Figure 1: Bike Rentals Histogram

3.1 Feature Scaling

Feature Scaling is a method used for data processing to standardize the range of the feature variables. The range of the features are normalized to a value having unit variance. The values can standardized to either ranging from (0 to 1), this can be obtained by normalizing by the maximum value or ranging from (-1 to 1) which can be obtained by subtracting the value by the mean and then normalizing by the maximum value.

[1]Registered and non-registered users

For example, the temperature ranges from 0 to 40 deg C but it is standardized to a variable having unit variance ranging from (0 to 1). We used standardized features for variables temperature, feels like temperature, humidity and wind speeds. The consequence of using feature scaling is to make sure all the features have similar variances and ranges which inturn helps in faster convergence of gradient descent of the cost function for the regression models.

3.2 Dependent Variable Correlation matrix

	workingday	**temp**	**atemp**	**hum**	**windspeed**	**cnt**
workingday	1.00	0.06	0.05	0.02	-0.01	0.03
temp	0.06	1.00	0.99	-0.07	-0.02	0.40
atemp	0.05	0.99	1.00	-0.05	-0.06	0.40
hum	0.02	-0.07	-0.05	1.00	-0.29	-0.32
windspeed	-0.01	-0.02	-0.06	-0.29	1.00	0.09
casual	-0.30	0.46	0.45	-0.35	0.09	0.69
registered	0.13	0.34	0.33	-0.27	0.08	0.97
cnt	0.03	0.40	0.40	-0.32	0.09	1.00

Table 1: Features Correlation Matrix

Interpretation of table 1 are given below:

- 0 indicates no linear relationship.
- +1 indicates a perfect positive linear relationship: as one variable increases in its values, the other variable also increases in its values via an exact linear rule.
- -1 indicates a perfect negative linear relationship: as one variable increases in its values, the other variable decreases in its values via an exact linear rule.
- Values between 0 and 0.3 (0 and -0.3) indicate a weak positive (negative) linear relationship via a shaky linear rule.
- Values between 0.3 and 0.7 (0.3 and -0.7) indicate a moderate positive (negative) linear relationship via a fuzzy-firm linear rule.
- Values between 0.7 and 1.0 (-0.7 and -1.0) indicate a strong positive (negative) linear relationship via a firm linear rule.

According to the values in table 1 only temp (normalized temperature, value = 0.4), atemp (normalized feeling temperature, value = 0.4), and humidity (value = -0.32) has moderate positive-negative linear relationship with count (the total bike count of each hour).

This indicates that these features implies an important role in regression and classification algorithms. This information helps us to do feature selection when building models for prediction. Figure 2 shows the relationsip between temperature and bike rental count.

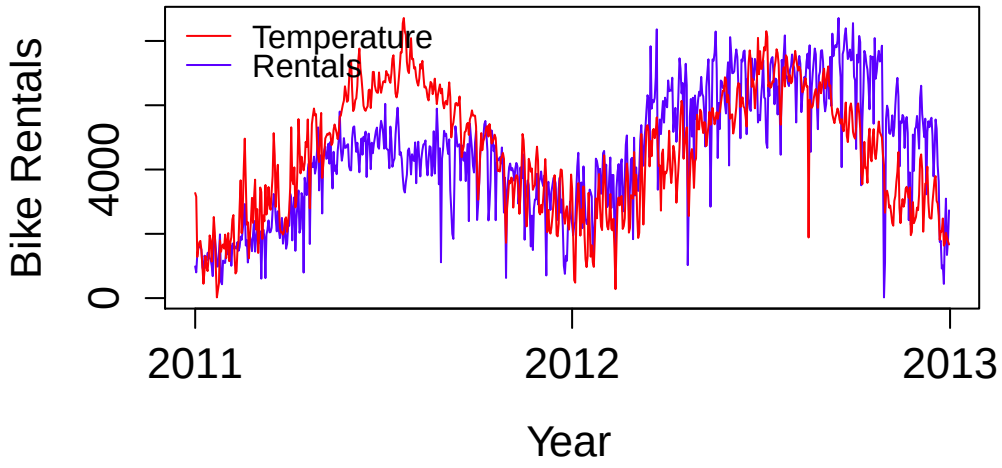

Figure 2: Temperature Bike Counts Correlation

Note that, the correlation coefficient of windspeed and count is only 0.09. This number indicates a very weak linear relationship correlation. This challenges the common sense of those living in colder climates as windspeed, and therefore comfort of the ride, would effect the willingness commute by bike. However, the average in Washington D.C. the value is 8.2 MPH, this can be described as a "gentle breeze" defined by the Beaufort[2] wind force scale. Therefore, both the correlation coefficient value and real windspeed data show that windspeed will not affect the rental numbers of the D.C. area.

Figure 3 shows the correlation between wind speeds and rentals, which resulted in 0.09.

3.3 Stepwise Regression Feature Selection

Stepwise regression is a method used to automate the process of choosing the feature variables used in the regression models. This method allows selecting a subset of the predictor variables from a larger set of predictor variables by performing a stepwise regression. The stepwise regression can be performed by forward selection, backward elimination or both. Forward selection is a method where the algorithm starts with no predictor variables and

[2]http://en.wikipedia.org/wiki/Beaufort_scale

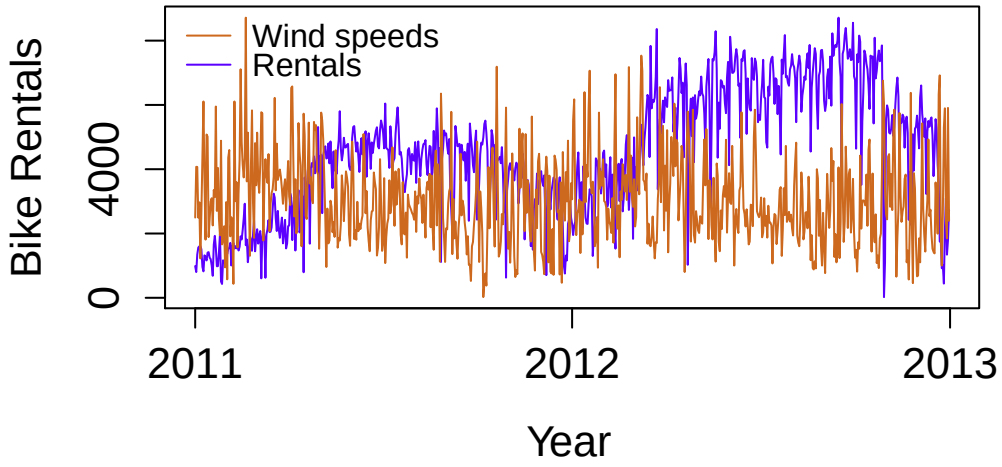

Figure 3: Correlation of Wind Speed and Rentals

adds variables iteratively, choosing variable based on which improves the model the most. Backward elimination is where the algorithm starts with a model that has all the predictor variables, and then deletes variables from the model that improves the model the most after its deletion, iteratively. We used both forward selection and backward elimination stepwise regression.

Using stepwise regression, we were able to ascertain that features such a hour23 (23rd hour of the day), season_winter (winter season) are eliminated from the model. The predictor variables such as hour23 and season_winter have very low correlation coefficients, -0.1171 and 0.02942 respectively.

4 Approach

Our approach involved linear regression, logistical regression, naïve Bayesian classification, SVM, and decision tree. Both linear regression and SVM was used for prediction while logistical regression, naïve Bayesian classification, SVM and decision tree was used for classification.

4.1 Regression

4.1.1 Linear Regression

We built multiple linear regression models for both daily and hourly data. We built multiple regression models based on different feature selections. Based on the data procession, predictor variable analysis, feature selection methods, we built multiple linear regression models, which takes the general form:

$$y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i = \mathbf{x}_i^T \beta + \varepsilon_i, \quad i = 1, \ldots, n \tag{1}$$

where p is the number of variables. Examples of values of x are weather (i.e. rain), temperature, number of rentals, etc. The β values are the learned parameters.

4.1.2 Non-linear Regression

Based on the analysis of the predictor variables, we build non linear regression models using nonlinear combination of the model parameters. We transformed predictor variables such as the temperature, feels like temperature to their respective square values. Predictor variables such as wind speed which have lower impact on the model, based on the correlation analysis of predictor variables, were transformed to square roots of the respective values. Using these non linear combination of the predictor variables, we built non linear regression models for the hourly dataset.

4.1.3 Support Vector Machines

Support Vector Machines (SVM) is a supervised machine learning method that can be used to build regression models. We used SVM to build regression models for the hourly dataset. We tuned the SVM models with a multitude of parameters such as kernel functions, degree of the polynomial, constant for regularization term which intuitively is a parameter that helps mitigate the effects of overfitting, γ the parameter that defines how far a training example's influence reaches etc. For kernel functions we experimented with SVM models that used different kernel functions such as linear, polynomial, radial and sigmoid functions.

4.2 Classification

For bike sharing companies, the cost of arranging bikes to different station to meet customers needs is a big part of the total operation cost. The prediction given by regression model could provide good reference for such companies. However, there are some extreme data instances that cannot be solved by regression model prediction. These case will make regression approach less reliable and will mislead companies decisions for arranging bikes for stations. If in some cases, the number of bike needed is extreme high, but regression model cannot give such prediction, then it would be a huge loss of income of the company and also harm the customers benefit. A reliable classification approach is need to detect such extreme cases and let companies to prepare for such cases.

We will discretized bike count as "low" and "high" to show the extreme case. But for different companies, the ability of the amount of bikes to prepare for stations at each hour is quite different. It means the definition of "low" and "high" of each bike sharing company is different. We will set the boundary of "high" as 600, 700, 800 respectively and set the boundary of "low" as 100 to analyze the performance of our classification model.

We used J48 decision tree, naive bayes, SVM and logistic regression models to classify hourly bike sharing data. All these models were used with combination of multiple features trying to analyze how these features affected prediction outcome. Using the correlation matrix we decided which combination of features to be used in classification models.

4.3 Evaluation

The results of the experiments gave a wide variety of success.

4.3.1 Training and Testing

We split the entire dataset into a training and test set in 70 : 30 percent ratio. The models were trained on the training set and we used these model to predict the total number of bike rentals in the testing set.

4.3.2 K Fold Cross Validation

We are using 10 fold cross validation to test accuracy of our models. It divides the data in 10 sets and treats 9 sets for training, 1 set for testing. This process is repeated 10 times and the output represents average of all the runs.

4.3.3 Regression

The regression model and predictions are evaluated using Coefficient of Determination or R^2 and Root Mean Squared Error (RMSE). R^2 is the measure of how well a model has fit the data. R^2 can be computed by

$$\begin{aligned} R^2 &\equiv 1 - \frac{SS_{res}}{SS_{tot}} \\ SS_{tot} &= \sum_i (y_i - \overline{y})^2 \end{aligned} \tag{2}$$

where $\overline{y}$ is the mean value; that is the total sum of squares and SS_{res} is the sum of squared residuals.

Root Mean Squared Error (RMSE) is a measure of the square root of the mean of the squared errors or squared deviations of the prediction values from the actual values. This value is an indicator to compare different models. A lower RMSE value indicates a better model. RMSE can also be used as a measure of how well a model fits the data and if the model is overfit or underfit. This is done by computing the RMSE of both the training set and the testing set. A model that is fit well will have comparable RMSE values. But, if the

training RMSE is significantly lower than the RMSE of testing set then it indicates a model that is overfit, in which case we can use methods like regularization or increase and decrease the values of regularization parameters accordingly.

4.3.4 Classification

Evaluation for J48 and logistic regression models is done based on accuracy, precision, recall and f-measure. These measures can be computed from truth table. Accuracy shows how close the model was in predicting right outcomes, precision is the fraction of data marked as relevant and relevant data by test $\frac{TP}{TP+FP}$, recall is the fraction of data marked as relevant and all data supposed as relevant $\frac{TP}{TP+FN}$ and f-measure is another measure to compute a models accuracy using precision and recall.

5 Experimental Results

5.1 Regression

5.1.1 Linear and Non-Linear Regression

The R-squared training data value obtained from the linear regression runs was 0.2522 and the coefficients detailed in table table 2. The RMSE of the predicted values is 207.856. These analysis comes from the hourly data.

The processed data mentioned in section 3 yielded an R-squared training data value of 0.646. The RMSE of the predicted values is 163.3292.

This test was repeated by square rooting humidity and windspeed, squaring atemp and yielded an R-squared value of 0.6442. These variables were chosen from the data analysis performed on the correlation matrix. The RMSE of the predicted values is 163.8344.

Parameter	Description	Coefficient
β_0	Intercept	123.7933
β_1	season	7.6740
β_2	mnth	-2.3836
β_3	holiday	-23.3311
β_4	weekday	1.3178
β_5	workingday	-0.5981
β_6	weathersit	1.6828
β_7	atemp	357.3604
β_8	hum	-225.3344
β_9	windspeed	36.6928

Table 2: Linear Regression Coefficients by Hour

5.1.2 SVM Regression

The stepwise regression given in section 3.3 did not provide good results. Instead the following parameters were used for the SVM model.

- discretized seasons
- discretized months
- discretized hours
- discretized weekdays
- workingday
- discretized weather (clear, cloudy, perception)
- atemp
- humidity
- windspeed

However, it did perform similar to linear regression as described in section 5.1.1 with a RMSE value of 163.1421.

The epsilon value used for all SVM regression runs was 0.1. The results for the cross fold validation for SVM are given in table 3.

Cost	Gamma	Kernel	Degree	Vector Count	RMSE
1000	0.0001	radial	N/A	8724	139.6982
10	0.01	sigmoid	N/A	12963	1427.655
10	0.01	polynomial	2	8807	133.0355
10	0.01	polynomial	3	8317	128.6375
10	0.01	polynomial	4	8345	129.9472

Table 3: SVM Results

The scatter plot of the SVM prediction with the bike rental counts in figure 4 graphically shows the error well.

5.2 Classification

The purpose of this project was to predict maximum number of bikes rented under and give situation. We first use regression models to achieve those predictions, but there were few outliers that were not covered by regression models. We then created few classification models to make accurate predictions for these outliers.

5.2.1 J48 Decision Trees

After using the four classification models discussed in this paper, we observed that J48 produced the best results. From the decision tree provided by J48 we made few conclusions about how the model was handling different features. It seemed like the model was most

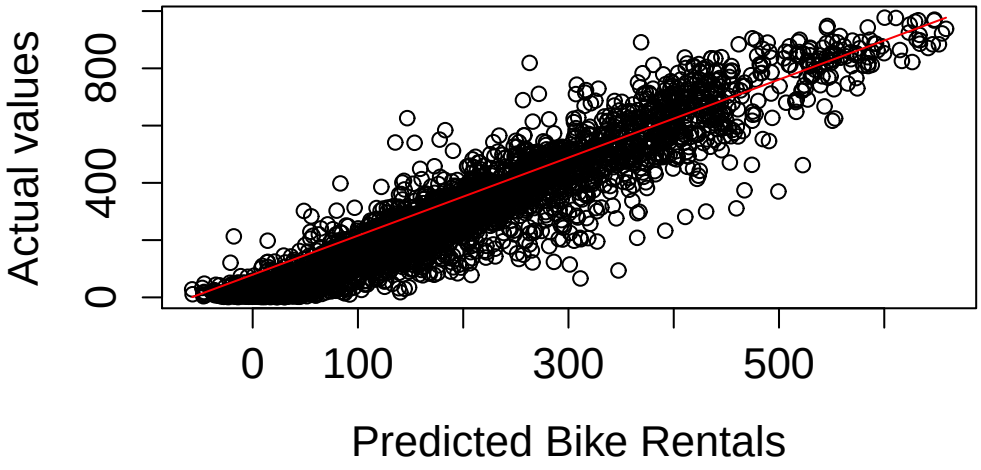

Figure 4: Actual Rentals vs. Predicated Values

dependent on time of the day and then affected by weather conditions or what day of the week it is. Also the feature "season" was given second preference to feature "weathersit". This feature mentions if there was rain, snow, thunder or humidity.

	TP Rate	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**Class**
	0.999	0.079	0.997	0.999	0.998	low
	0.921	0.001	0.976	0.921	0.948	high
Weighted Avg.	0.996	0.075	0.996	0.996	0.996	

Table 4: Decision Tree Results

The accuracy of the decision tree classification was 99.536%

5.2.2 Logistic Regression

Throughout our experiments it seemed like logistic regressions did not provide results as good as J48. We tried different feature sets, thresholds and evaluation models, but still J48 provided better results.

The accuracy of logistic regression classification was 99.60%.

Low	High	
2064	2	Low
7	82	High

Table 5: Decision Tree Confusion Matrix

	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
	0.998	0.047	0.998	0.998	0.998	low
	0.953	0.002	0.956	0.953	0.955	high
Weighted Avg.	0.996	0.045	0.996	0.996	0.996	

Table 6: Logistic Regression Results

5.2.3 SVM (SMO)

We used SVM for regression models as well as classification models. The regression model did not predict accurate classes for outliers, but when it was used for classification it produced results almost as good as J48. To get a better comparison between J48 and SVM we tried different threshold values and it seemed J48 performs better than SVM in most cases.

The accuracy of SVM classification was 99.6316%.

5.2.4 Naïve Bayes

One of the reasons why Naïve Bayes did not provide a better classification model is because we removed latent variables from out feature list

The accuracy of naïve bayes classification was 99.3451%.

In earlier trials J48 decision tree made better predictions than logistic regression. After coming across a set of features that gave us optimum outcome we tried tweaking classification model arguments. For J48 decision tree we tried changing confidence factor, value of seed and minimum number of object, but since the accuracy was already high making these changes to argument did not cause any significant change to the outcome. Similar results where seen with logistic regression model. J48 and SVM showed better results, but when we changed threshold of classes J48 produced better results.

5.3 Results Conclusion

The correlation coefficient of temperature, feeling temperature between count are both 0.40, which shows they have moderate positive linear relationship. Figure 5 illustrates the variation tendency of the positively correlated bike counts and temperature by hour.

Low	High	
6996	14	Low
15	307	High

Table 7: Logistic Regression Confusion Matrix

	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
0.999	0.069	0.997	0.999	0.998	0.61	low
0.931	0.001	0.979	0.931	0.955	0.906	high
Weighted Avg.	0.996	0.066	0.996	0.996	0.996	0.623

Table 8: SVM Classification By Hour

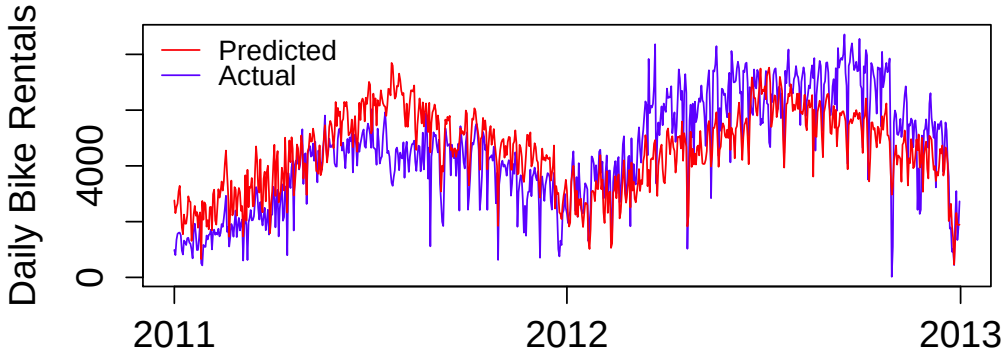

Figure 5: Predicted Values vs. Actual Values

5.4 Implementation

Data processing, was implemented using Java, that is predictor variable transformations from nominal to numerical/boolean variables, generation of new variables, discretizing the variables.

We used Java and Weka[4] API for building and evaluating classification models. We used Weka API for building J48 Decision trees, Naive Bayes , Logistic Regression and Support Vector Machines (SVM).

We implemented the regression models using R[6] such as Multivariate Linear Regression and Support Vector Machines including methods such as regularization and stepwise

Low	High	
2339	2	Low
7	95	High

Table 9: SVM Classification Confusion Matrix

	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
0.998	0.118	0.995	0.998	0.997	0.878	low
0.882	0.002	0.957	0.882	0.918	0.958	high
Weighted Avg.	0.993	0.113	0.993	0.993	0.993	0.881

Table 10: Naïve Bayes Classification By Hour

regression for model selection. MASS[7] and e1071[2] packages in R were used for the implementation.

6 Conclusion

After comparing results from regression and classification models, we observed few relations between features and nominal class. As discussed earlier windspeed does not affect prediction models, since Washington has average windspeed throughout the year. This assumption was later proved by correlation table. During step-wise regression the model removed "winter" since it did not have much effect on bike count class. This behavior was observed by classification models as well.

7 Future Work

As a progression to this project, we can add geo-location of each docking station, duration for each trip to our current data. Often bike sharing systems give offers where first 30-60 minutes of the ride is free, using this new data set we will be able to provide better free slots. Using geo-locations of docking stations we can alert bike sharing companies to restock bikes at a station.

References

[1] Uci machine learning repository: Bike sharing. http://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset#/.

Low	**High**	
2337	4	Low
12	90	High

Table 11: Naïve Bayes Classification By Hour

[2] Evgenia Dimitriadou, Kurt Hornik, Friedrich Leisch, David Meyer, and Andreas Weingessel. Misc functions of the department of statistics (e1071), tu wien. *R package*, pages 1–5, 2008.

[3] Kyle Gebhart and Robert B Noland. The impact of weather conditions on capital bikeshare trips. In *92nd Annual Meeting of the Transportation Research Board, Jan*, pages 13–17, 2013.

[4] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[5] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT Press, 2012.

[6] RDevelopment Core Team et al. R: A language and environment for statistical computing. *R foundation for Statistical Computing*, 2005.

[7] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, New York, fourth edition, 2002. ISBN 0-387-95457-0.