PREDICTION OF RENTAL DEMAND FOR A BIKE-SHARE PROGRAM

A Paper
Submitted to the Graduate Faculty
of the
North Dakota State University
of Agriculture and Applied Science

By

Om prakash Nekkanti

In Partial Fulfillment of the Requirements
for the Degree of
MASTER OF SCIENCE

Major Department:
Computer Science

March 2017

Fargo, North Dakota

North Dakota State University
Graduate School

Title

PREDICTION OF 2016 RENTAL DEMAND IN GREAT RIDES BIKE SHARE PROGRAM

By

Om prakash Nekkanti

The Supervisory Committee certifies that this ***disquisition*** complies with North Dakota State University's regulations and meets the accepted standards for the degree of

MASTER OF SCIENCE

SUPERVISORY COMMITTEE:

Dr. Kendall E. Nygard
Chair

Dr. Simone Ludwig

Dr. Ranjit Prasad Godavarthy

Approved:

3/27/2017
Date

Dr. Brian M. Slator
Department Chair

ABSTRACT

In recent years, bike-sharing programs have become more prevalent. Bicycle usage can be affected by different factors, such as nearby events, road closures, and on-campus traffic policies. The research presented here analyzed the effect of weather (average temperature, total daily precipitation, average wind speed, and weather outlook), day of the week, holiday/workday, month, and season on the use of the Great Rides Bike Share program in Fargo, North Dakota, U.S.A. This study also focused on predicting the 2016 rental demand for the Great Rides Bike Share program using Bayesian methods and decision trees. Further, the order of importance among the causal attributes was assessed. It was found that decision trees worked well to predict the 2016 demand.

ACKNOWLEDGEMENTS

I would like to express my gratitude to Dr. Kendall E. Nygard for his continued support with the paper. His critique has helped me to understand and to develop my skills for machine learning. I would also like to thank Dr. F. Adnan Akyuz and Dr. Curt Doetkott for their help with data. Finally, I would like to thank Dr. Simone Ludwig and Dr. Ranjit Prasad Godavarthy for their willingness to serve on my committee and to provide useful input.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

LIST OF ABBREVIATIONS

NOAA	National Oceanic and Atmospheric Administration
USNO	United States Naval Observatory
GBM	Generalized Boosted Models
GLMNet	Generalized Linear Models with Elastic Net Regularization
PCR	Principal Component Regression
SVR	Support Vector Regression
Ctree	Conditional Inference Trees

1. INTRODUCTION

Today, more than 500 cities in 49 countries host bike-sharing programs. Urban transport advisor Peter Midgley notes that "bike sharing has experienced the fastest growth of any mode of transport in the history of the planet" [1]. Modern bike-sharing systems have greatly reduced the theft and vandalism that hindered earlier programs by using easily identified specialty bicycles with unique parts that would have little value to a thief, by monitoring the cycles' locations with radio frequency or GPS, and by requiring credit-card payment or smart-card-based membership to check out bikes. With most systems, after paying a daily, weekly, monthly, or annual membership fee, riders can pick up a bicycle that is locked to a well-marked bike rack or electronic docking station for a short ride (typically an hour or less) at no additional cost and can return it to any station in the system. Riding longer than the program's specified amount of time generally incurs additional fees to maximize the number of available bikes.

Bike-sharing programs are becoming popular for the following reasons [2]:

- They decrease greenhouse gases and improve public health.
- They increase transit use due to the new bike transit trips, the improved connectivity to other modes of transit because of the first-mile/last-mile solution that bike-sharing helps solve, and the decreased number of personal vehicle trips.

Due to the increased popularity of these bike-sharing programs across the world, it is increasingly becoming important to analyze these systems from different perspectives. Figure 1 shows the growth of these bike-sharing programs over the last decade. In this paper, I focus on predicting the 2016 bike-rental demand for the Great Rides Bike Share system based in Fargo, North Dakota. Fargo's Great Rides is an 11-station, 101-bicycle seasonal system. In 2015, there were 143,000 trips and an average of 6-7 rides per bike per day, more usage per bike than in

New York; Washington, D.C.; or Paris [3]. The main reason for the program's success is the integration with student IDs; the Great Rides seasonal pass is included as part of the mandatory student-activity fees at North Dakota State University (NDSU).

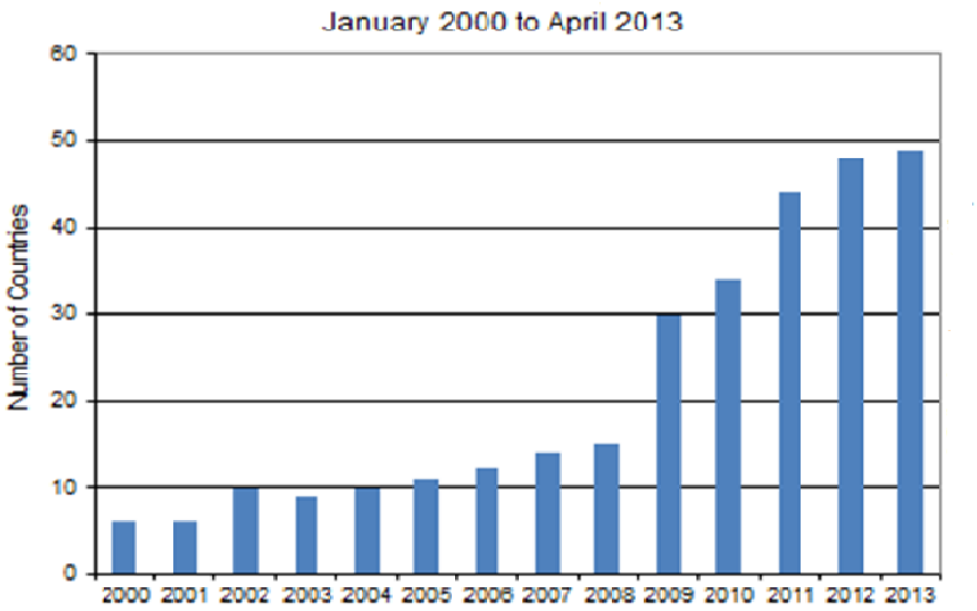

Figure 1. Number of countries with bike-sharing programs [1].

The goal of this analysis is to predict the total demand for the Great Rides Bike Share in 2016 and to compare the prediction with the actual demand for validation. This study involves finding a model which predicts the demand with a small error rate by combining the historical usage patterns with the related information about weather, workday/holiday, temperature, wind, precipitation, day of the week, season, and month. This study did not include details about each customer's usage pattern or each docking station's activity. This study helps program managers with system planning and with making informed decisions, such as when to perform maintenance.

Multiple tools, such as SQL Server, Excel, Weka, and SAS, were used during the study for data pre-processing, developing predictive models, and visualizing demand predictions with different models (Naïve Bayes, Bayesian Network, and C4.8).

In this paper, the second chapter reviews the attempts made by other authors in order to predict the demand for bike-sharing programs. The third chapter explains the data preprocessing and methodology that are used to handle the problem, the reasoning for choosing the Bayesian and decision-tree algorithms to build models, the evaluation method used to calculate the accuracy of the aforementioned models, and the implementation in Weka. The fourth chapter shows the accuracies and predictions of the statistical (Bayesian) logic-based (decision tree) models [4] as well as the relative importance of the demand prediction's attributes. Finally, the fifth chapter concludes the study with the results and outlines the different directions where this work can be extended and improved.

2. LITERATURE REVIEW

Almost all the studies were done used the Capital Bikeshare program's data set which is available on the Kaggle website [5]. Capital Bikeshare is the name of bike-sharing program in Washington, D.C. This bike-share dataset has 12 features, and they are similar to the Fargo Great Rides' features except for few, such as atemp (feels like temperature) and humidity. The Fargo Great Rides Bike Share dataset is explained, in detail, in the next chapter. Godavarthy et al. [6] have studied the operational aspect, travel behavior, and travel mode shift due to Great Rides Bike Share program in Fargo, North Dakota, and found that introduction of bike share program in the Fargo has increased the bicycling behavior for North Dakota State University (NDSU) students and Fargo residents. Further, significant number of NDSU students and Fargo residents are also willing to use bike share program during cold winters if available [7]. Several studies attempted to solve a similar problem, and this section briefly reviews those studies in comparison with the approach used for this paper.

In the study done by M. Alhusseini, two approaches were used with the hourly data for the Capital Bikeshare program [8]. With the first approach, demand was modeled as a numeric attribute and was predicted using support vector machines. For the second approach, demand was modeled as a categorical attribute with five ordinal class labels; the demand was predicted using SoftMax regression and support vector machines. In my study, demand was modeled as a categorical attribute, which is like Alhusseini's second approach except that there are only two class labels due to the limitations with feasible accuracy when using a small dataset. Here, Decision trees and Bayesian Classifiers are used for classification. Also, the demand in my study is calculated together for casual users and registered users, unlike in Alhusseini's study.

In the study done by J. Du et al. [9], hourly data for the Capital Bikeshare were used to predict the hourly demand, expressed as the number of rides. An important part of their study was that they generated different data subsets for training from the altered original data and that they applied ensemble methods along with other regression models. They used stacking, generalized boosted models (GBM) and Random Forest for the ensemble methods as well as generalized linear models with elastic net regularization (GLMNet), principal component regression (PCR), support vector regression (SVR), and conditional inference trees (Ctree) for the regression models. In this paper, demand is predicted as a category due to the small size of the data set. Unavailability of hourly data is the reason for small data set, which increases the dataset size exponentially.

In the study done by C. Lee et al., the feature set is modified such that every categorical attribute is binary (has two possible values). The rest of the attributes, such as numeric and categorical, with two possible values were used as is [10]. For categorical attributes with multiple possible values, a new feature was created for each possible label under it such that each label has a separate column. Lee et al. used a Poisson regression model, Neural Network, and Markov model to predict the number of rides. In this paper, demand was predicted as a categorical attribute, and daily data were used, unlike the hourly data in Lee's study. The reason for doing this is the unavailability of reliable hourly data for attributes such as humidity, precipitation, and temperature to prepare a data set. This unavailability of hourly data resulted in a daily data set which is smaller (236 training instances).

In the study done by W. Wang [11], the Citibike data set was used. Citibike is the bike-share program in New York. Unlike other studies, he prepared a dataset. His data preparation involved collecting information from three sources: one for historical weather data, one for

individual ride information, and one for official holidays in New York. He used multiple Linear regression, Neural Networks, Decision trees, and Random forests to predict the system's hourly demand. The main aspect of this study was that he focused on the "Random forest" ensemble method to improve accuracy. He created multiple trees by adding new independent attributes, improving the quality of each independent attribute, and transforming the dependent variable. In this paper, dataset preparation is done by collecting information from the national oceanic and atmospheric administration (NOAA) and the Great Rides Bike Share websites. Because the demand is predicted as a categorical attribute, important attributes for predicting demand were assessed.

3. METHODOLOGY

This chapter presents the data preparation and data pre-processing that were done in order to apply different machine-learning schemes, selection of learning schemes, slection of evaluation methods for the models and Implementation.

3.1. Data Preparation

In this study, the data sets were prepared by aggregating information from two websites: Great Rides Bike Share and the NOAA [3, 12]. The dataset that is available on the Great Rides website is for the 2015 season. Each ride has the following attributes:

- Checkout Station
- Return Station
- Checkout Date
- Return Date
- Checkout Time
- Return Time

Although these attributes do not reveal anything about the circumstances associated with any given ride, the attributes can be used to deduce the circumstances. Our goal was to predict the demand for a given day based on a set of attributes associated with that day. Therefore, rides were grouped by date. There were 236 unique dates in the 2015 bike-share data set. This grouping gives the number of rides that occurred for each date. This grouping was done by importing the raw data from the Great Rides website into SQL Server as a table. The following query (Figure 2) was used to group the rides according to the date.

```
SELECT convert(date,CheckoutDate,106)  as Date, count(CheckoutDate) as NumberofRides
FROM [Training].[dbo].['season-data-2015Copy1$'] group by CheckoutDate order by CheckoutDate
```

Figure 2. SQL query for grouping the 2015 bike-share records by date.

Data from the above query were used in combination with the following attributes for analysis. For each date in the query's result set, the corresponding attributes were collected. Daily data from the NOAA website were used because there were no reliable hourly data available for precipitation, temperature, and wind speed.

- **Workday/Holiday:** The reason for considering this attribute was that it helps to determine if people are using bike rides for their commute to work/school or for recreational purposes. Possible values for this column were workday and holiday. Workday was assigned to days from Monday through Friday, and Holiday was assigned to weekends and public holidays [13].
- **Day of the week:** This factor was useful when determining the use type. Also, it helped the model to capture usage variations during the weekdays. The main purpose of this attribute was to determine if class schedules had any effect on the demand. Possible values for this column were MWF (Monday, Wednesday, Friday), TR (Tuesday, Thursday) and SS (Saturday, Sunday). Each day was assigned to its corresponding group.
- **Season:** This attribute helped to capture the effect of seasonality on bike usage. Possible values for this column were Winter, Spring, Summer, and Fall. Each day was assigned to its season according to the date. Seasonal information was based on information from United States Naval Observatory (USNO) website [14].

- **Month:** This attribute gave more granularity for the season. Possible values for this column were 1 through 12.
- **Precipitation:** People tended to skip a bike ride because of precipitation. This attribute helped to determine the extent to which precipitation affects the rider's choice of riding or not. Historical precipitation data were collected from the NOAA website and entered manually. The NOAA website reported the total amount of precipitation for a given day. Each day was assigned a category according to the total amount of precipitation over the entire day (Table 1). Because the range of numeric values for precipitation was not large, discretization helped to better capture the change in the values [15].

Table 1. Precipitation.

Label	Precipitation in inches
No-Rain	0
Drizzle/light rain:	0.01-0.1
Moderate:	0.1 to 1.0
Heavy:	>= 1

Possible categorical values for precipitation

- **Average temperature:** Temperature plays a major rule in the bike usage. Its relationship with bike usage is not linear. Choosing this attribute helped to determine people's bike-usage preferences. Temperature data were collected from the NOAA website and entered manually [15]. The average temperature for the entire day was used for preparing the data. This attribute was discretized into 6 bins using 16.66 percentiles in a SAS program to make it a categorical variable. Table 2 shows the intervals used for discretization.

Table 2. Average temperature.

Label	Temperature intervals in fahrenheit
Bin1	21-44
Bin2	44-54
Bin3	54-61
Bin4	61-67
Bin5	67-73
Bin6	73-85

Possible categorical values for temperature

- **Average wind speed:** Because wind makes it difficult to pedal a bike, we can use this attribute to help determine if people are really concerned about wind. Similar to temperature and precipitation, wind speed data were collected from the NOAA website [15]. This attribute was discretized into seven bins using the Beaufort Scale as shown in Table 3 [16].

Table 3. Average wind-speed categories.

Average wind speed in mph	Label
>=32	High wind
>=25	Strong breeze
>=19	Fresh breeze
>=13	Moderate breeze
>=8	Gentle breeze
>=4	Light breeze
>=1	Light air
< 1	Calm

- **Weather conditions:** This attribute helped capture the effect of weather outlook on rider's behavior. Data were collected from the NOAA website and organized in the Weather Conditions column. This column had a value from the list shown in Figure 3 for each day, depending on the weather outlook [15]. In order to capture a bicyclist's

perspective, these weather conditions were grouped according to Table 4. This categorization was based on assumptions about how a bicyclist perceives weather in relation to using a bicycle. These values were assigned to each day, depending on the day's actual weather outlook. Table 4 shows the discretization in detail.

```
SYMBOLS USED IN COLUMN 16

1 = FOG OR MIST
2 = FOG REDUCING VISIBILITY
    TO 1/4 MILE OR LESS
3 = THUNDER
4 = ICE PELLETS
5 = HAIL
6 = FREEZING RAIN OR DRIZZLE
7 = DUSTSTORM OR SANDSTORM:
    VSBY 1/2 MILE OR LESS
8 = SMOKE OR HAZE
9 = BLOWING SNOW
X = TORNADO
```

Figure 3. Weather-column values [15].

Table 4. Weather conditions.

Numbers	Label
1 (Fog) and 2 (Fog reducing visibility)	Sub-optimal
8 (Smoke)	Manageable
X (Tornado), 9 (Blowing snow), 7 (Dust storm), 6 (Drizzle), and 5 (Hail)	Impossible
3 (Thunder) and 4 (Ice pellets)	Inclement
Nothing	optimal

Possible categorical values for weather conditions

- **Number of rides:** This numerical attribute had values ranging from 1 to 1924. Initially, the column was divided into 10 different classes from 1 (lowest) to 10 (highest) using the 10^{th} percentile and 3 different classes from 1 (lowest) to 3 (highest) using the 33.3 percentile to predict with more granularity and to eliminate the class-imbalance problem. However, this grouping gave a maximum test accuracy of 33% and 67%, respectively. In an attempt to obtain better accuracy, the

classification labels were reduced to 2 labels (below average and above average); these labels allowed for a larger error margin in the model. The average number of rides for the 2015 season was 607. The number of rides column was separated into two categories using this average value.

Table 5. Number of rides.

Number of Rides	Label
Greater than 607	MorethanAverage
Less than 607	LessthanAverage

Possible categorical values for Number of rides

Deducing these factors was the crux of the data preparation. These attributes were crucial to apply the machine learning techniques because they helped the model learn what conditions were associated with high, moderate, and low demands. There were literature studies which focused on generating the feature set from the base set of attributes by using constructor functions that employed a predefined set of arithmetic and logic operators [17]. It is one approach that can be employed to extend feature engineering.

3.1.1. Discretization of Numerical Attributes

In the data-preparation process, numerical attributes, both outcome and predictors, are discretized for the following reasons:

- The number of examples is 236, which is very small for a machine-learning training data set [18]. Discretization helps to reduce the number of possible combinations across the attribute set, thus reducing the learning space.
- The number of attributes in the training set is 8, which is closer to 6, the accepted number of attributes in the literature for calling a dataset small [18].

Table 6 illustrates the records from the prepared data. The complete data are provided in Appendix A.

Table 6. Training set for machine-learning algorithms.

Workday/Holiday	Day of the week	Month	Season	Precipitation	Average temperature	Average wind speed	Weather conditions	Number of rides
Holiday	GroupSS	3	winter	No Rain	Group6	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	3	winter	Heavy Rain	Group6	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	3	Spring	Light Rain	Group6	Fresh breeze	Manageable	LessthanAverage
Holiday	GroupSS	3	Spring	Moderate Rain	Group6	Fresh breeze	Sub-optimal	MorethanAverage
Workday	GroupMWF	4	Spring	No Rain	Group5	Moderate breeze	optimal	MorethanAverage
Holiday	GroupSS	4	Spring	No Rain	Group6	Light breeze	optimal	LessthanAverage
Workday	GroupTR	4	Spring	Moderate Rain	Group6	Gentle breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	4	Spring	Heavy Rain	Group6	Gentle breeze	Sub-optimal	MorethanAverage
Workday	GroupMWF	4	Spring	No Rain	Group4	Fresh breeze	optimal	MorethanAverage
Workday	GroupTR	4	Spring	Heavy Rain	Group5	Gentle breeze	optimal	MorethanAverage

Only a few training examples are shown in the table.

Demand values for the bike-sharing system were expressed in two ways. For the first method, the number of rides was expressed as a proportion of the total number of rides possible per day for the bike-share system. The maximum possible utilization can be calculated using the following approach:

- The average time for a single ride was calculated using all 143,354 records in the 2015 bike-share data set. The result was 16 minutes.
- The total time available for each bike was 60 minutes times 18 hours (6 AM to 12AM) which equals 1080 minutes.
- The total number of possible rides for each bike was the total available time divided by the average ride time, which equals 67 (1080/16).
- The total number of rides possible for the bike-share program was the number of bikes in the system times the possible number of rides for a bike, **6,817 (**67*101).

With the second method, the number of rides was expressed as below or above average compared to the daily average of 607 rides in the 2015 season.

This study used the second method for following reasons:

- Demand prediction was handled as a classification problem.
- All the attributes, including outcome, were categorical in the training examples.
- Discriminant models built in this study predicted categories instead of actual numbers because the training examples were categorical.

The first method is useful when the problem is handled as a regression problem, i.e., when the outcome column is numerical.

3.2. Modeling Algorithms' Selection

The data set was modeled as a multi-class classification problem (i.e., The outcome variable "Number of rides" has multiple class labels for values.) in the beginning, and different multi-class classification algorithms were used upfront in the analysis. Because the accuracy for those models was small, granularity was reduced by decreasing the number of class labels to two.

We asserted that no learning algorithm can uniformly outperform other algorithms for all data sets. Therefore, our approach was to empirically calculate the accuracy of the candidate algorithms for the problem and to select the one that provides the highest accuracy [19]. The following algorithms were selected to build models with the 2015 bike-sharing dataset. The reasons for choosing them are explained below in the corresponding subheadings.

- Naïve Bayes
- Bayes Network
- C4.8

3.2.1. Simple Naïve Bayes

Naïve Bayes is a common statistical learning algorithm that is used for classification. Like other statistical approaches, Naïve Bayes assumes an underlying probability model that provides the probability that an instance belongs in each class, rather than a simple classification [4]. A Simple Naïve Bayes classifier is easy to implement, and its accuracy tends to be good. It assumes that all the attributes are equally important and statistically independent [20]. It calculates the prior probabilities for each class-attribute value and updates the values by calculating the posterior probabilities based on the examples. The reasons for selecting the Naïve Bayes classifier are as follows:

- The Naïve Bayes method needs a relatively small data set when compared to neural networks and Support Vector Machines. Because the bike-share data set is small, the method is suitable for this scenario [4].
- The Naïve Bayes approach works well when all the causal/predictor attributes and the dependent attribute are categorical[4, 21], which is the case for this study.
- The Naïve Bayes algorithm train very quickly because it requires only a single pass of the data either to count the discrete variables' frequencies or to compute the normal probability density function for continuous variables under normal assumptions[4].
- The Naïve Bayes method is transparent and can be easily grasped by users [4].

3.2.2. Bayesian Network

A Bayesian Network is a graphical model for concisely representing probability relationships among a set of variables [22]. A Bayesian Network structure is a directed acyclic graph with nodes that are in one-to-one correspondence with the features. The arcs represent casual influences among the features while the lack of possible arcs encodes conditional independencies. Figure 4 shows a sample Bayesian Network. Each node could have one to many parents, depending on the relationship between the attributes. Moreover, a feature node is conditionally independent from its non-descendants given its parents. In Figure 4, X_1 is conditionally independent from X_2 given X_3 if $P(X_1|X_2,X_3)=P(X_1|X_3)$ for all possible values of X_1,X_2,X_3.

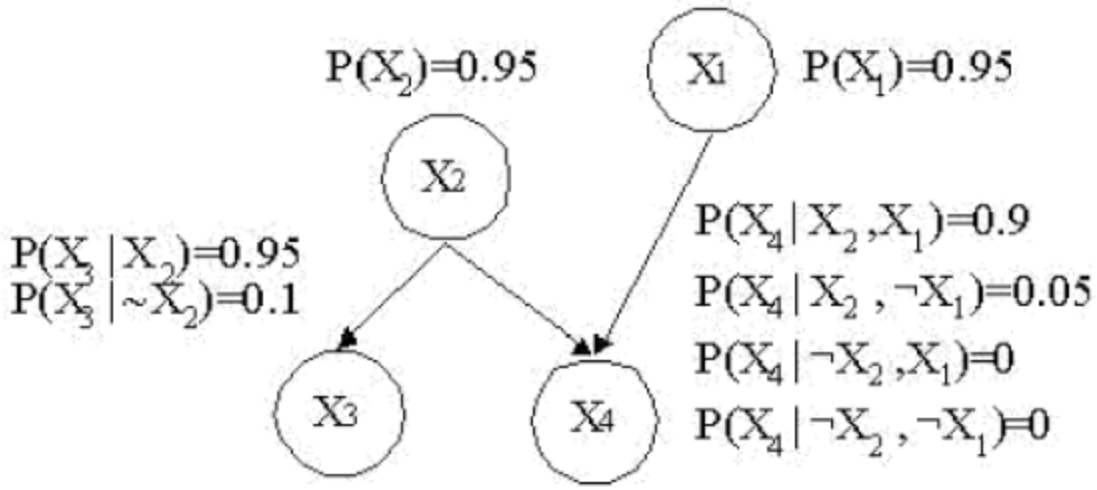

Figure 4. Structure of a Bayes network [22].

A Bayesian Network is constructed in two steps: 1) a function for evaluating a given network based on the data and 2) a method for searching through the space of possible networks. In this study, the K2 algorithm was used to construct the Bayesian Network. The Bayesian Network method was chosen for the following reasons:

- A Bayesian Network is suitable for data sets with fewer attributes[23]. Because there are eight attributes, this method was suitable for this study.
- A Bayesian Network works well when all the attributes in an instance are categorical and there are no missing values [23].
- A Bayesian Network can capture complex, conditional probability distributions for the class attribute better than the Naïve Bayes method, given the values of other casual attributes [22].

3.2.3. C4.8 Decision Tree

Decision trees are one of the logic-based techniques used for classification. Decision trees try to find the attributes' hierarchy based on the training data that show the importance of individual attributes when classifying a new instance [24].

Decision trees classify instances by sorting them based on feature values. Each node in a decision tree represents an instance feature to classify, and each branch represents a value that the node can assume. Instances are classified by starting at the root node and are sorted based on their feature values. Figure 5 is an example of a decision tree.

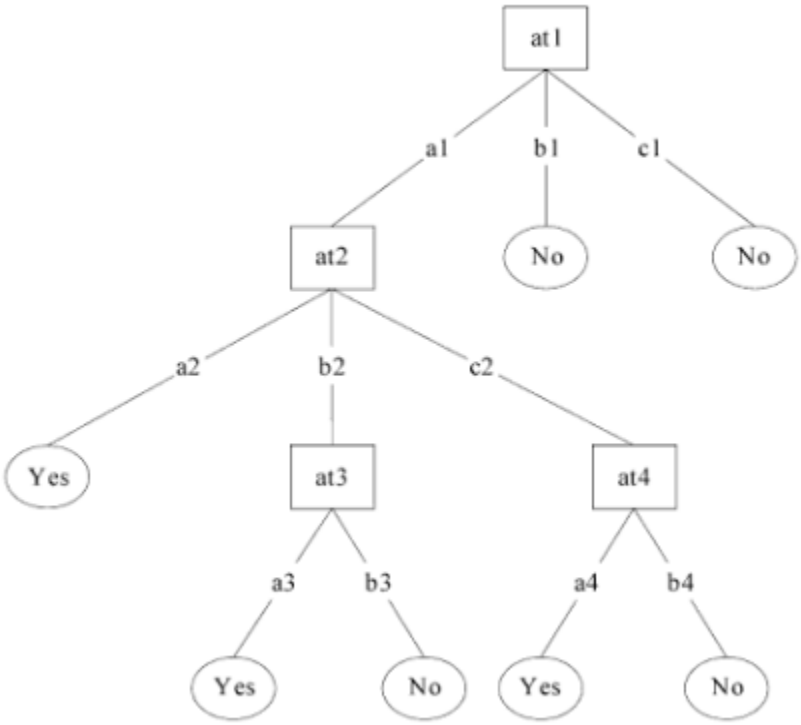

Figure 5. Decision tree [24].

Using the decision tree depicted in Figure 5 as an example, the instance (at1=a1, at2=b2, at3=a3, at4=b4) would sort to the nodes at1; at2; and, finally, at3, which would classify the instance as being positive (represented by the "Yes" values). The feature best divides the training data would be the root node of the tree. Constructing optimal binary decision trees is an NP-complete problem.

C4.8 is the flagship decision-tree algorithm to build trees; it is well known in the literature. It is the last open-source algorithm that was written by John Ross Quinlan [25]. All

later versions are proprietary. J48 is the name given to the implementation of C4.8 in the Weka data-mining tool. The following reasons are why it was chosen for this study.

- Decision trees are more suitable for classification problems with small data sets [26]. Because the bike-share data set for this study is small, it is excellent for the problem.
- Decision trees have a very good combination of error rate and speed when compared to other learning algorithms such as neural networks [27].
- Decision trees are a logic-based technique that supports a good understanding of the concept underlying the data (i.e., good transparency) [24].
- Logic-based methods, such as decision trees, tend to perform better when dealing with discrete/categorical variables [4].

3.3. Evaluation of the Methods

This study used a tenfold, stratified cross-validation in Weka to evaluate the learned models' accuracy [28]. The difference between a regular n-fold cross validation and a stratified cross validation is that each one of the n-parts will have correct representation for class labels like the original data set in terms of proportions. By default, Weka uses stratified cross-validation. The following reasons are given for choosing stratified cross-validation over percentage split and regular n-fold cross-validation.

- When we randomly partition data into two parts (2/3rd for training and 1/3 for testing) using a percentage split. There is a chance to obtain more examples from one class type in the training set, and it could lead to a prediction error because the model will not train on other class types. Also, there is a chance to have all instances of one season in the training set and another season in the testing set.

- When we use regular n-fold cross-validation, it evens out the seasonality. For example, if we go with a tenfold cross-validation for 236 examples (8 months) from the 2015 bike-share data set, there is a chance that the other nine training parts will have all the records from 7 months and only few records from one month because the process is random for the worst-case scenario. Choosing a stratified cross-validation will ensure that each part is identical to others in the class representation.
- Another reason is that the data set is small, making it unusable for other model-validation methods.

3.4. Implementation

Weka 3.8 a well-known data-mining tool that is used to build models [28]. The weka is a bird that is endemic to New Zealand. The Weka workbench contains a collection of visualization tools and algorithms for data analysis and predictive modeling, together with graphical user interfaces, for easy access to this functionality. Weka supports several standard data-mining tasks, more specifically data preprocessing, clustering, classification, regression, visualization, and feature selection. All the Weka techniques are predicated on the assumption that the data are available as a single flat file or relation, where each data point is described by a fixed number of attributes (normally numeric or nominal attributes, but some other attribute types are also supported). Weka provides access to SQL databases using Java Database Connectivity and can process the result returned by a database query. It is not capable of multi-relational data mining, but there is separate software to convert a collection of linked database tables into a single table that is suitable for processing using Weka. Another important area that is currently not covered by the algorithms included with the Weka distribution is sequence modeling.

3.4.1. Configuration Used for the Naïve Bayes Model

The Weka tool has different classifiers, and they are grouped together depending upon their similarities. Naïve Bayes is available under the "bayes" group of classifiers in the Weka explorer. The Bayes group has all the classifiers that are based on Bayesian methods. Figure 6 shows the configuration used with the Weka tool for the Naïve Bayes method.

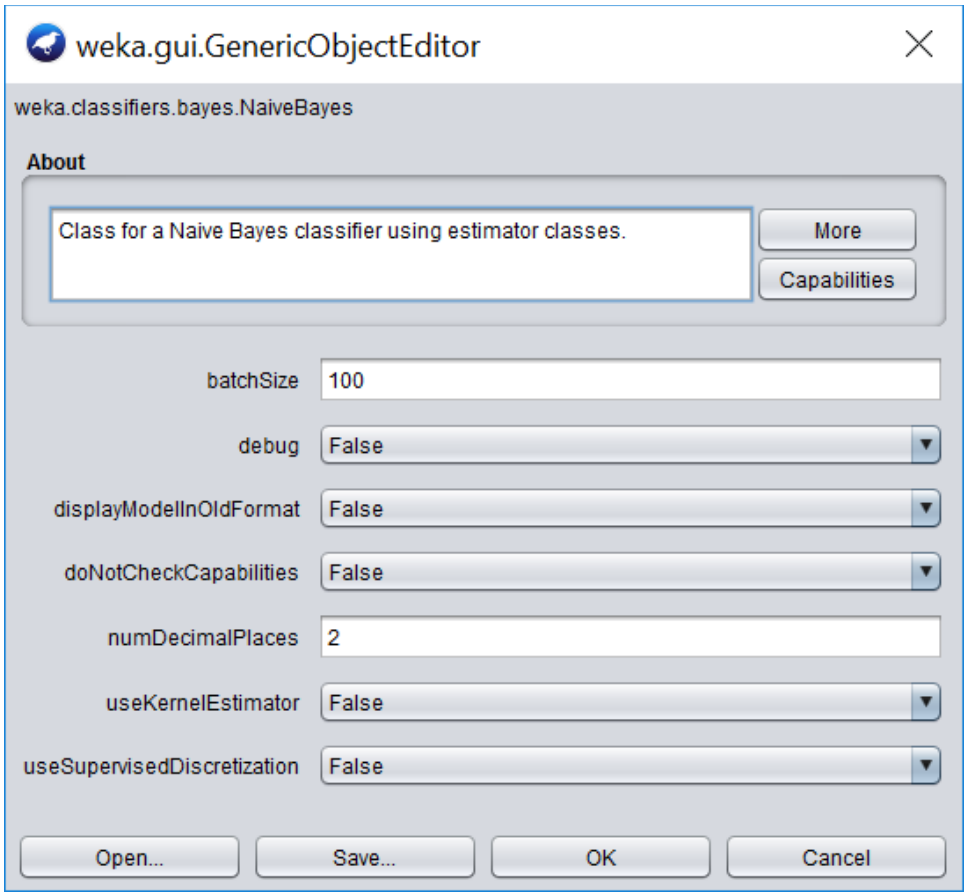

Figure 6. Naïve Bayes configuration panel.

Table 7 explains the purpose of different fields in the Naïve Bayes configuration panel that was shown in Figure 4.

Table 7. Naïve Bayes configuration panel.

Field	Use
batchSize	It allows to choose the preferred number of instances to process if batch prediction is being performed.
debug	It outputs additional information to the console if it is set to true. By default, it is False.
usekernelEstimator	It is utilized when a kernel estimator has to be used for numerical attributes rather than a normal distribution.
useSupervisedDiscretization	This filter is used when numerical attributes have to be converted to nominal attributes.

Fields in the Naïve Bayes configuration panel

3.4.2. Configuration Used for the Bayesian Network Model

Similar to Naïve Bayes, the Bayesian Network is available the “bayes” group of classifiers in Weka because it was based on probabilities. Figure 7 shows the configuration used for the Bayesian Network in the Weka tool.

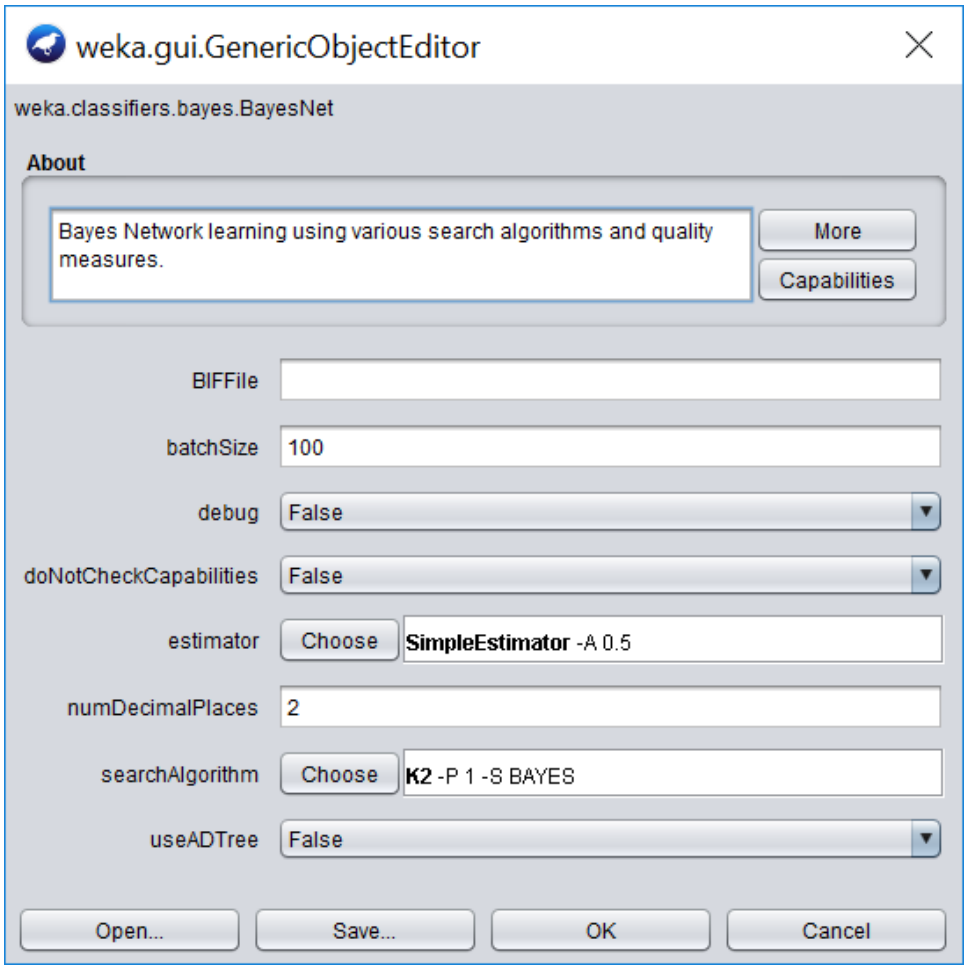

Figure 7. Bayesian network configuration panel.

Table 8 explains the purpose of the different fields for the Bayesian network configuration panel that is shown in Figure 5.

Table 8. Bayesian network configuration panel.

Field	Use
batchSize	It allows to choose the preferred number of instances to process if a batch prediction is being performed.
debug	It outputs additional information to the console if it is set to true. By default, it is False.
estimator	The SimpleEstimator is used for approximating the conditional probability tables of a Bayes network once the structure is learned.
searchAlgorithm	This Bayes network learning algorithm uses a hill-climbing algorithm that is restricted by an order on the variables.

Fields in the Bayesian network configuration panel

3.4.3. Configuration Used for the C4.8/J48 Model

J48 is the Weka name for algorithm C4.8. It is available under the "trees" group. Figure 8 shows the Weka configuration used for J48. Most fields are left to have the default values because they are associated with the C4.8 algorithm.

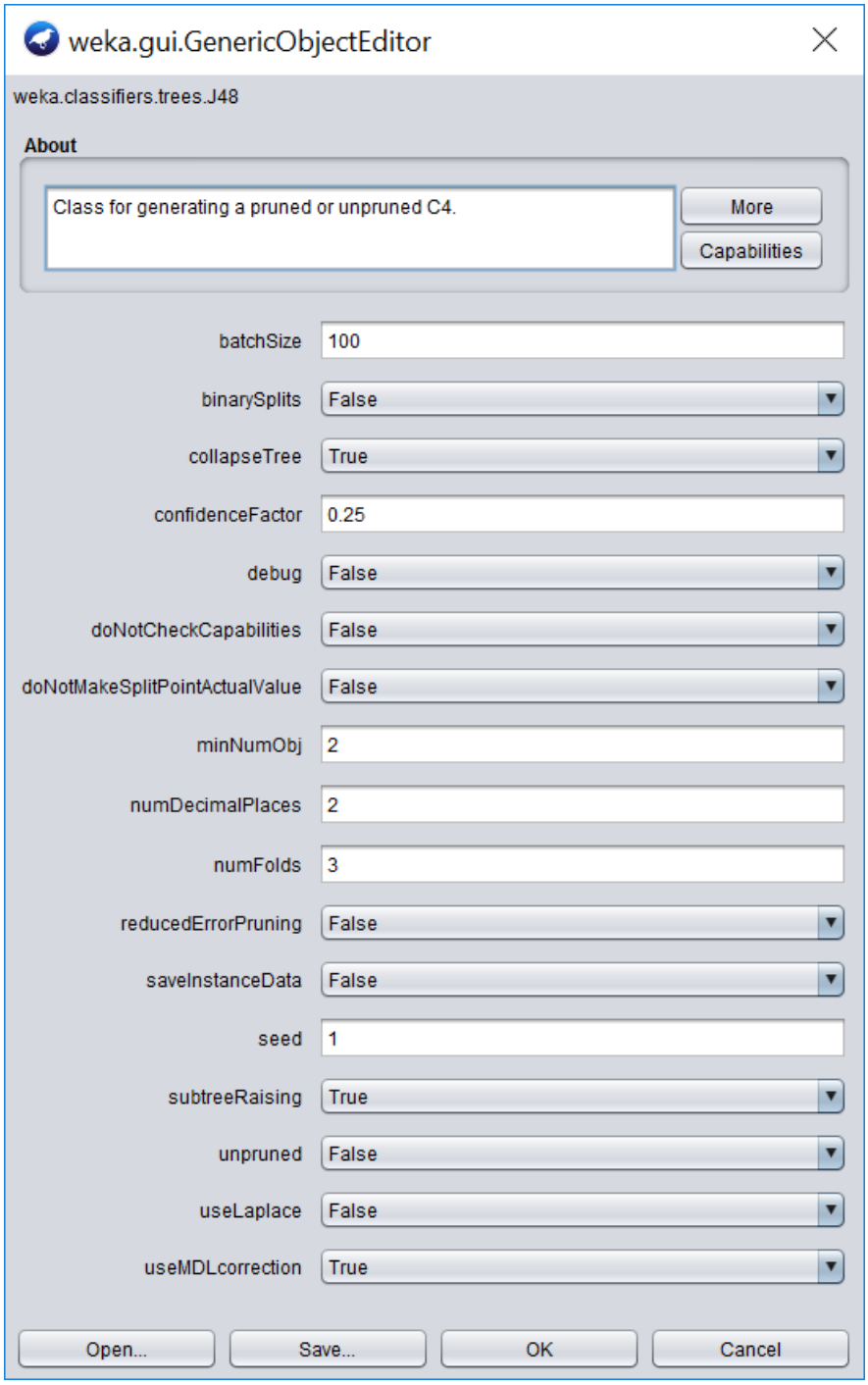

Figure 8. C4.8/J48 configuration panel.

Table 9 explains the purpose of different fields in the J48 configuration panel that was shown in Figure 6.

Table 9. C4.8 configuration panel.

Field	Use
batchSize	It allows to choose the preferred number of instances to process if a batch prediction is being performed.
binarySplits	It allows the use of binary splits on nominal attributes when building the trees.
Collapse tree	It allows removing the parts that do not reduce the training error.
Confidence factor	It is the amount of factor used for pruning.
minNumObj	Number of instances per leaf
Unpruned	It allows for pruning of the tree being built.

Fields in the C4.8 configuration panel

4. RESULTS

This chapter discusses, in detail, the accuracies of the different candidate models that were selected to capture the concept underlying the 2015 Great Rides Bike Share data set and to predict bike-rental demand for the 2016 bike-sharing season using these models. The chapter also covers the order of importance among the attributes that were used for predicting the demand with decision trees.

4.1. Accuracy of Models

Tenfold stratified cross-validation was used to calculate the models' accuracy. The following subheadings detail the confusion matrices and other metrics that corresponding to each model discussed in the Methodology chapter.

4.1.1. Naïve Bayes

The accuracy of this model on the 2015 Great Rides Bike Share data set (the training set) was 76.69% {(126+55/236)*100} using the confusion matrix presented in Table 10. Of the 145 less-than-average demand days, 127 days were correctly predicted by this model, and of the 91 more-than-average demand days, 55 days were correctly predicted by the model. The accuracy for the model was calculated by using the diagonal elements in Table 10 because they represent the correct predictions.

Table 10. Confusion matrix for Naïve Bayes

	LessthanAverage	MorethanAverage	Total
LessthanAverage	126	19	145
MorethanAverage	36	55	91

Table 11 shows the Naïve Bayes class-level accuracies using measures such as true positive rate, false positive rate, precision, recall, f-measure, area under ROC curves, and recall-precision curves. The following list gives the meaning for each measure:

- **True Positive Rate:** This measure tells us about the proportion of positive instances that are classified correctly. An optimal classifier has this measure approaching one.
- **False Positive Rate:** This measure tells us about the proportion of negative instances that are incorrectly classified as positives.
- **Precision:** This measure tells us about the proportion of instances that are truly of a class among the total instances that are predicted as that class.
- **Recall:** Like the true positive rate, this measure tells us about the proportion of instances that are classified as a given class divided by the actual total in that class.
- **F-measure:** It is a combined measure for precision and recall. It is equal to 2*recall*precision/(recall+precision).
- **ROC Curve:** The larger the area under the curve, the better the model is. The ROC area represents the area under the curve, and an ideal classifier will have values approaching one.
- **PRC Curve:** This curve is similar to lift charts and ROC curves. It is ideal to have this value close to one.

Table 11. Detailed accuracy by class for Naïve Bayes

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	PRC Area
LessthanAverage	0.869	0.396	0.778	0.869	0.821	0.781	0.816
MorethanAverage	0.604	0.131	0.743	0.604	0.667	0.781	0.687

4.1.2. Bayesian Network

The accuracy for this model on the 2015 Great Rides Bike Share data set (training set) was 80.93% {(128+63/236)*100} using the confusion matrix presented in Table 12. Of the 145 less-than-average demand days, 128 were predicted correctly by this model, and of the 91 more-

than-average demand days, 63 were predicted correctly by model. The model's accuracy was calculated by using the diagonal elements in Table 12 because they represent the correct predictions.

Table 12. Confusion matrix for the Bayesian network

	LessthanAverage	MorethanAverage	Total
LessthanAverage	128	17	145
MorethanAverage	28	63	91

Table 13 shows the Bayes Network class-level accuracies using measures such as true positive rate, false positive rate, precision, recall, f-measure, area under the ROC curves, and recall-precision curves.

Table 13. Detailed accuracy by class for the Bayesian network

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	PRC Area
LessthanAverage	0.883	0.308	0.821	0.883	0.850	0.851	0.904
MorethanAverage	0.692	0.117	0.788	0.692	0.737	0.851	0.750

4.1.3. C4.8 Decision Tree

This model's accuracy with the 2015 Great Rides Bike Share data set (training set) was 79.23% {(127+60/236)*100} using the confusion matrix presented in Table 14. Of the 145 less-than-average demand days, 127 were predicted correctly by this model, and of the 91 more-than-average demand days, 60 were predicted correctly by model. The model's accuracy was calculated by using the diagonal elements in Table 14.

Table 14. Confusion matrix for the C4.8 decision tree

	LessthanAverage	MorethanAverage	Total
LessthanAverage	127	18	145
MorethanAverage	31	60	91

Table 15 shows the C4.8 decision-tree class-level accuracies using measures such as true positive rate, false positive rate, precision, recall, f-measure, area under the ROC curves, and recall-precision curves.

Table 15. Detailed accuracy by class for the C4.8 decision tree

Class	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	PRC Area
LessthanAverage	0.876	0.341	0.804	0.876	0.838	0.785	0.798
MorethanAverage	0.659	0.124	0.769	0.659	0.710	0.785	0.688

4.2. Predicting the 2016 Bike-Rental Demand for the Great Rides Program

To predict the demand on a given day accurately, the values for all the casual attributes need to be fed into the model. Not all the attributes used by the model can be deduced by the calendar date for a given day because the set of attributes has few based on daily weather. Thus, at the time of this study, the only possible way to obtain values for the weather-based attributes was with the assumption that weather during the 2016 bike-sharing season was going to be similar to the weather during the 2015 bike-sharing season.

To access accuracy, the predictions made by these trained models were then compared with the actual rides that happened during the 2016 season. Because the predictions from these models were categorical (i.e., above average or below average) for a given day, the actual number of rides column for the 2016 bike-share data had to be discretized for comparison.

Similar to the 2015 bike-sharing data set, number of rides for each day in the 2016 season was calculated by using the SQL statement shown in Figure 9 with the 2016 bike-sharing data that were available online [29]. Result set from this query shows the number of rides for every date in the season.

```
SELECT convert(date,CheckoutDate,106)  as Date, count(CheckoutDate) as NumberofRides
  FROM [Training].[dbo].[season-data-2016] group by CheckoutDate order by CheckoutDate
```

Figure 9. SQL query for grouping the 2016 bike-share records by date.

Before comparison, the number of rides column in 2016 season was discretized using the average number of rides from the 2015 season: 607. The number of rides corresponding to every day in the 2016 season was compared with 607 and assigned a "LessthanAverage" label if they were less than 607 and a "GreaterthanAverage" label if they were greater than 607.

Table 16 shows the prediction accuracies of different models for 2016 bike-sharing season, and subsequent pages have the visualizations that help to understanding each model's predictions. The total number of operational days during the 2016 Great Rides season was 218. The accuracy for each model was calculated by dividing the number of days which were correctly predicted by total number of days in the season.

Table 16. Prediction accuracies for the models

Model	Accuracy in percentage
Naïve Bayes	70
Bayes Network	72
C4.8	73

4.2.1. Visualization for the Naïve Bayes Prediction

A needle plot was used to visualize the prediction errors. The graph in Figure 10was generated in a SAS program, and the code is provided as Appendix B. The graph has a time line on the x-axis, ranging from March 28, 2016, to October 31, 2016, and a class label on the y-axis. For a day on the x-axis, there is a bar either below the reference line or above the reference line. A bar is displayed below the reference line when the number of rides was below the daily

average of 607 during the 2015 season and above the reference line when the number of rides was above the daily average of 607 from the 2015 season.

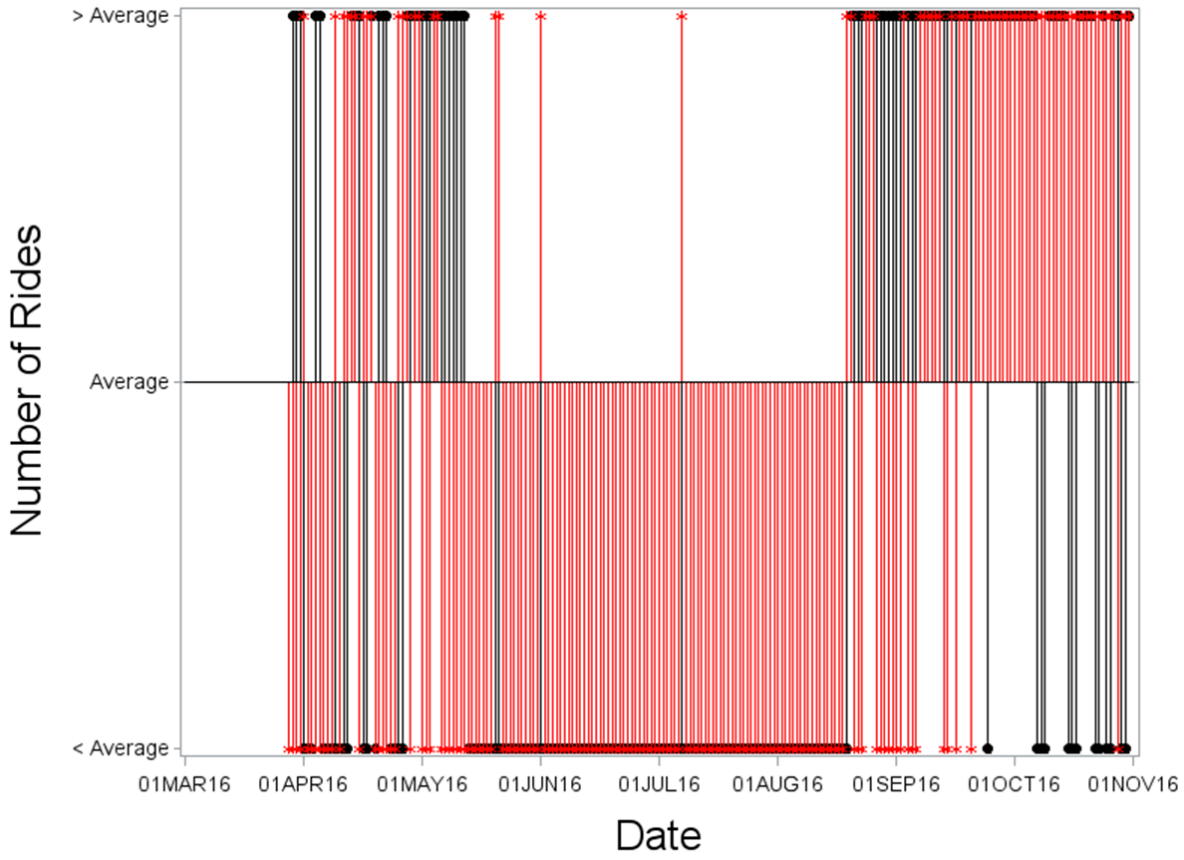

Figure 10. Naïve Bayes prediction.

In Figure 10 the red lines correspond to the Naïve Bayes prediction, and the black lines correspond to the actual 2016 data.

4.2.2. Visualization for the Bayes Network Prediction

Similar to Naïve Bayes, a needle plot was used to show the differences between the class labels predicted by the Bayesian network and the actual class labels from the 2016 bike-share data. If a day has a more-than-average number of rides, a bar is on top of the reference line, and if a day has a less-than-average number of rides, there is a bar at the bottom. Ideally, for a model,

both the red and black lines should coincide across the time line. The SAS code for the graph in Figure 11 is provided as Appendix C.

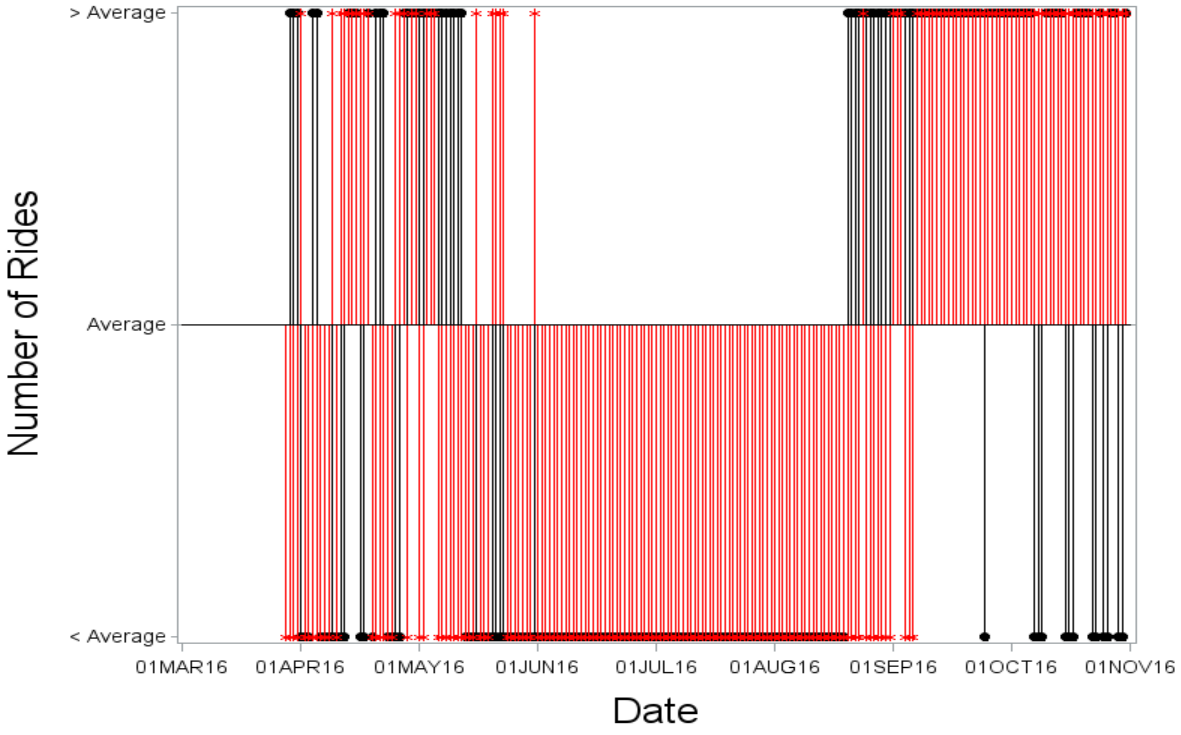

Figure 11. Bayesian network prediction.

4.2.3. Visualization for the C4.8 Tree Prediction

The graph in Figure 12 shows the prediction errors for the decision tree built by C4.8. This model has more accuracy when compared to other models. The SAS code for the graph is attached as Appendix D.

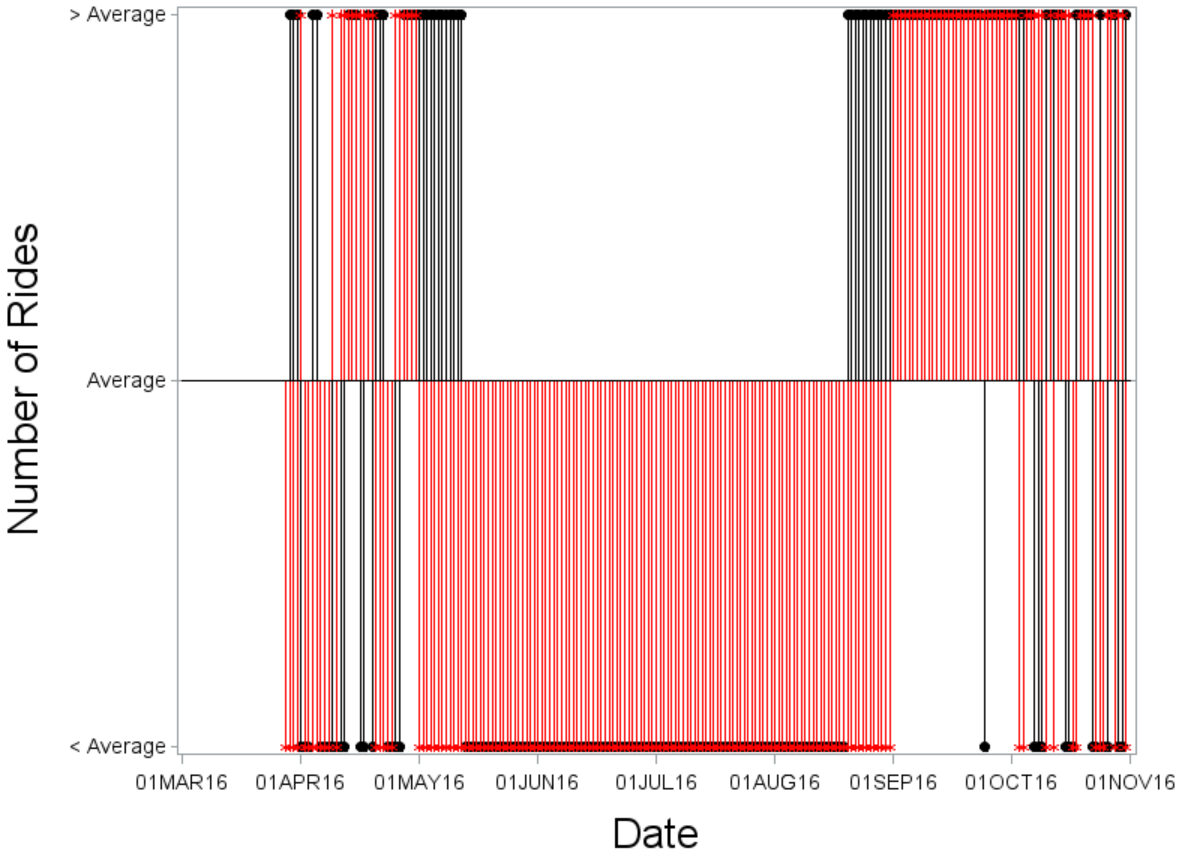

Figure 12. C4.8 decision-tree prediction.

4.3. Order of Importance Among the Causal Attributes

Because the decision tree that was built by the C4.8 algorithm has a higher accuracy for predicting demand, it was logical to use decision trees when determining the important attributes. To find the hierarchy among causal attributes, different trees were built by eliminating the attribute on root node in the original tree that was learned from the 2015 Great Rides data set.

Figure 13 shows the original tree that was built using the C4.8 algorithm. It was used to deduce the important attributes on which the model was relying to predict the demand for a given day. From the tree in Figure 13 we can make following inferences:

- Month was at the top of the decision structure, meaning that the model relied heavily on the month while predicting the demand. Thus, month is the first important attribute.
- The next attribute used by model was dependent upon the month. If month was greater than 9, the second attribute used by the model was weather conditions. If it was less than 5, the next attribute used by the model was temperature. In a way, this order translated to the season of the year.

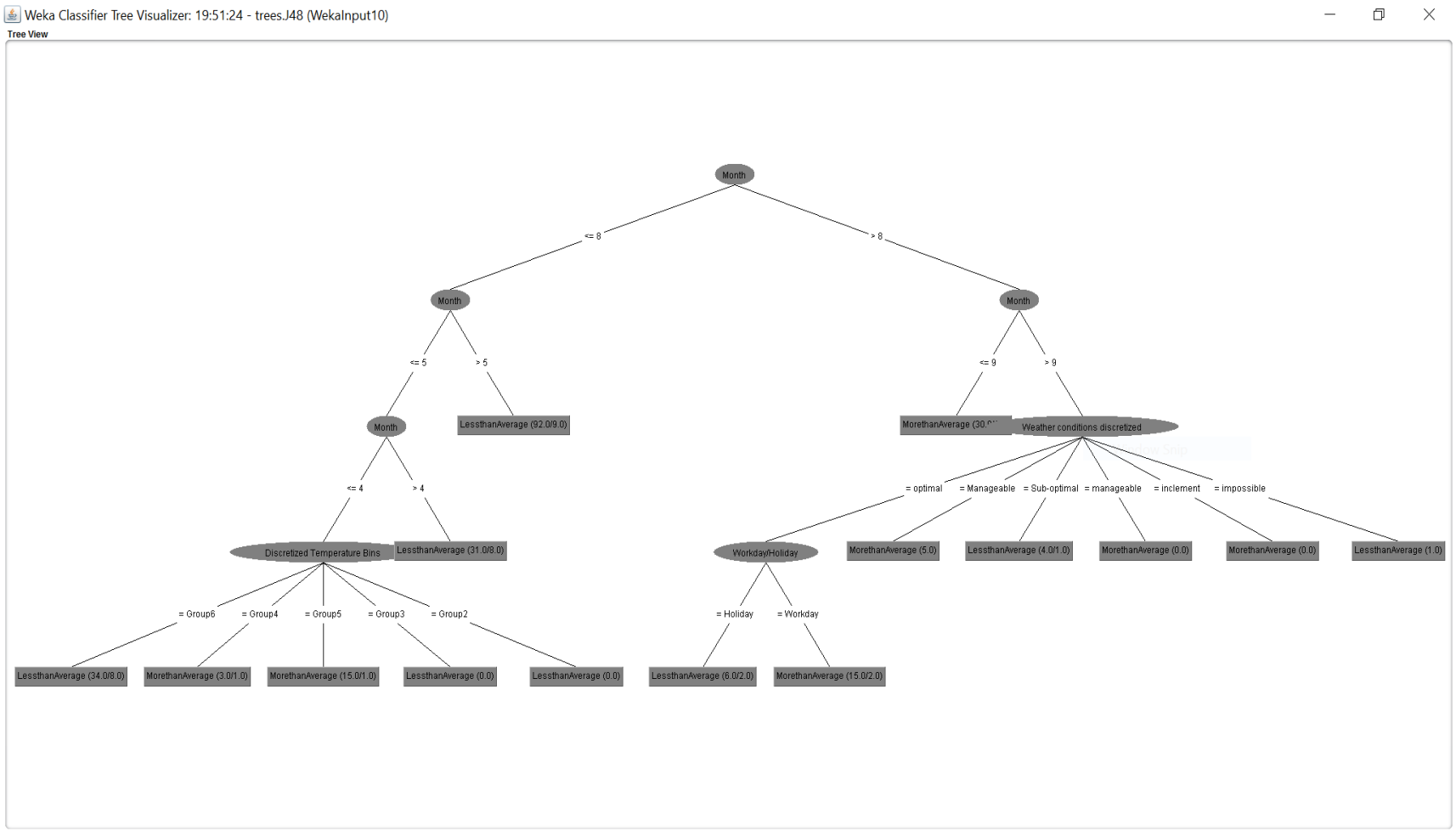

Figure 13. C4.8 decision tree learned from the 2015 season data set.

The second tree was built by removing the month attribute from 2015 Great Rides data set. Figure 14 shows the tree that was learned by using the C4.8 algorithm. From Figure 14, we can make the following inferences:

- While assessing the demand, the decision-tree model relied on the season attribute in the absence of month attribute. Thus, season was the second-most important attribute for the entire set of attributes.
- The day's average temperature was the next attribute on which the model relied.
- The final attribute that model used was the weather outlook.

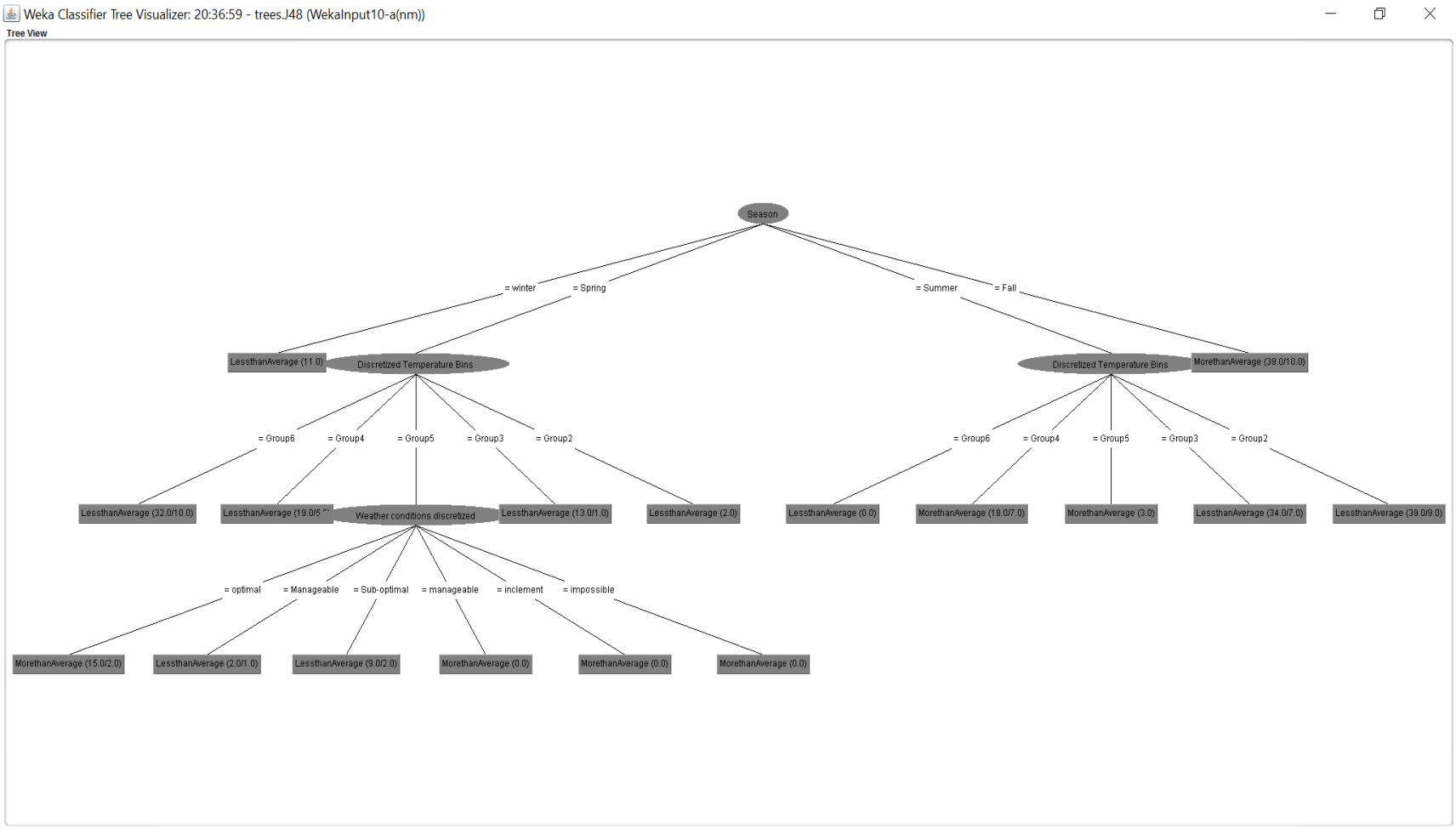

Figure 14. C4.8 decision tree learned from the 2015 season dataset without a month column.

The third tree was built by removing both the month and season attributes from 2015 Great Rides data set to find the third important attribute. From Figure 15, we can make the following inferences:

- The model primarily used average temperature in the absence of the month and season attributes while forecasting the demand. Thus, average temperature is the third-most important attribute for the set of attributes.
- The next attribute used by model was dependent upon the average-temperature class label. If average temperature was in group 5, it used the weather outlook for prediction, and if it was in group 4, it used precipitation and average wind speed for prediction.

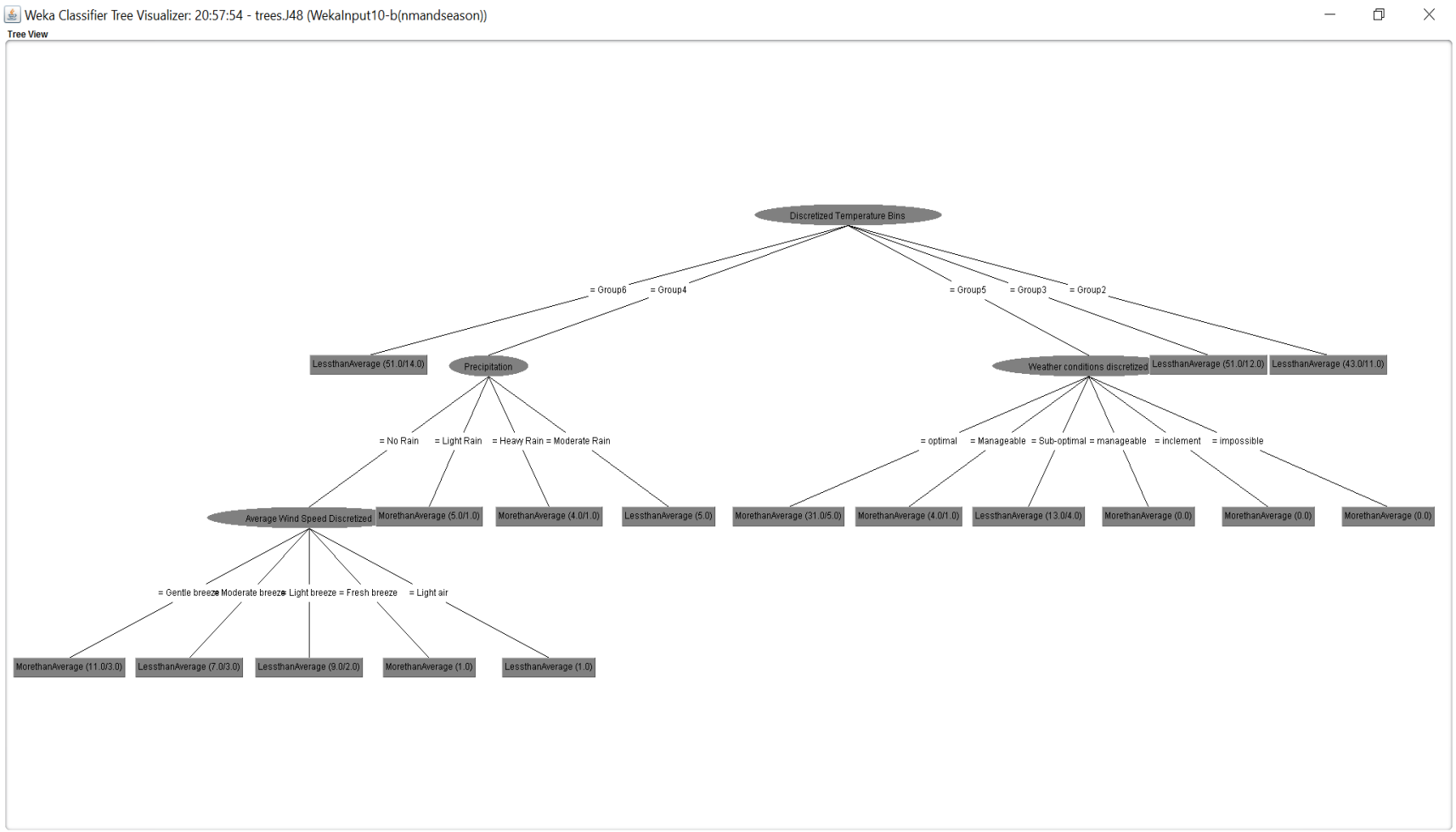

Figure 15. C4.8 decision tree learned from the 2015 season dataset without the month and season columns.

5. CONCLUSION

The aim of this project was to forecast bike-rental demand of Fargo's Great Rides program for the 2016 season. There are many factors that will affect bicycle users' behavior, for example, an event at the Fargo Dome, a marathon near campus, road closures near campus, or a short-term policy related to on-campus traffic. Because it is impossible to consider all factors in one study, this project focused on the bike-share program's daily demand prediction that was based on the available attributes for weather conditions (outlook), average temperature, average wind speed, total daily precipitation, workday/holiday, day of the week, month, and season.

The decision-tree model built by J48 worked well to predict the demand for the 2016 season, and the accuracy attained by this decision-tree structure was 73%. Accuracies for the Bayes Network and Naive Bayes were 72% and 70%, respectively. Therefore, we can conclude that decision trees capture the structural pattern in Great Rides Bike Share program better than other models. The order of importance among the causal attributes was month, season, average temperature, weather conditions, precipitation, and average wind speed. Different models were built with J48 to understand the order of importance among attributes. J48 tree used only six from the total set of attributes that were employed to predict demand.

5.1. Future Work

There are multiple ways to improve and extend the study. The following list provides some of the different directions in which the study could be extended.

- Hourly weather data can be obtained from reliable sources for the 2015 and 2016 bike-sharing season and combined to have a bigger training set which is granular enough to make hourly predictions.

- Station-level demand prediction can be done by incorporating station-specific attributes in the set of independent attributes with separate training sets for each station.
- Attributes can be modeled to predict the demand between different origin-destination pairs that were created from different bike-docking stations.
- Models based on neural networks and support vector machines (SVM) can be used to build a model from the data by keeping the features continuous/numerical. SVM and neural networks (sub-symbolic or numeric learning) perform better when dealing with continuous features and multiple dimensions [30, 31].
- An ensemble of classifiers can be generated in the following ways, and their predictions can be combined using voting and weighted voting [32].
    - The bagging technique [33] and the boosting technique [34] can be used to create an ensemble of classifiers with a single learning method for achieving more accuracy. To do that, responsiveness to changes in the training data by the classifier is important [33].
    - Different learning methods can be employed with the same training set in order to create an ensemble of classifiers.
    - An ensemble of classifiers can be created with a single learning method by using a set of feature subsets that are generated by using random selection. This technique is called the random subspace method [35].
- Another way to improve classification accuracy is to use hybrid techniques. For example, hybrid trees, such as the NBTree, contain the General Naïve Bayes classifier on the leaf nodes and regular, univariate splits on the internal nodes [36].

REFERENCES

[1] J. Larsen. (April 25, 2013). *Plan B Updates - 112: Bike-Sharing Programs Hit the Streets in Over 500 Cities Worldwide*. Available: http://www.earth-policy.org/plan_b_updates/2013/update112

[2] P. DeMaio, "Bike-sharing: History, Impacts, Models of Provision, and Future," *Journal of Public Transportation,* vol. 12, no. 4, p. 3, 2009.

[3] A. Corbin. (May 19th, 2016). *Why the country's best bike share might be in Fargo - Better Bike Share*. Available: http://betterbikeshare.org/2016/05/19/countrys-best-bike-share-might-fargo/

[4] S. B. KOTSIANTIS, "Supervised Machine Learning: A Review of Classification Techniques.," in *Emerging Artificial Intelligence Applications in Computer Engineering: Real World AI systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*, vol. 160, K. k. llias Maglogiannis, Manolis Wallace, John Soldatos, Ed. 1 ed. (Frontiers in Artificial Intelligence and Applications, Amsterdam: IOS Press, 2007, pp. 3-24.

[5] *Kaggle Bike Sharing Demand*. Available: https://www.kaggle.com/c/bike-sharing-demand/data

[6] R. Godavarthy, J. Mattson, and A. Taleqani, "Evaluation Study of Bike Share Program in Fargo, ND," Small Urban and Rural Livability Center2017.

[7] R. P. Godavarthy and A. Rahim Taleqani, "Winter bikesharing in US: User willingness, and operator's challenges and best practices," *Sustainable Cities and Society*, vol. 30, pp. 254-262, 4// 2017.

[8] M. Alhusseini. (2014). *Prediction of Bike Sharing Systems for Casual and Registered Users* [online]. Available: http://cs229.stanford.edu/proj2014/Mahmood%20Alhusseini,Prediction%20of%20Bike%20Sharing%20Demand%20for%20Casual%20and%20Registered%20Users.pdf

[9] J. Du, R. He, and Z. Zhechev. *Forecasting Bike Rental Demand* [online]. Available: http://cs229.stanford.edu/proj2014/Jimmy%20Du,%20Rolland%20He,%20Zhivko%20Zhechev,%20Forecasting%20Bike%20Rental%20Demand.pdf

[10] C. Lee, D. Wang, and A. Wong. *Forecasting Utilization in City Bike-Share Program*. Available: http://cs229.stanford.edu/proj2014/Christina%20Lee,%20David%20Wang,%20Adeline%20Wong,%20Forecasting%20Utilization%20in%20City%20Bike-Share%20Program.pdf

[11] W. Wang, " Forecasting Bike Rental Demand Using New York Citi Bike Data," M.S. thesis, Computer Science Department, Dublin Institute Of Technology, 2016.

[12] (2/13/2017). *National Oceanic and Atmospheric Administration*. Available: http://www.noaa.gov/

[13] (2/13/2017). *Holiday Schedules for 2014 and 2015*. Available: http://dchr.dc.gov/page/holiday-schedules-2014-and-2015

[14] (2/13/2017). *Earth's Seasons*. Available: http://aa.usno.navy.mil/data/docs/EarthSeasons.php

[15] N. W. S. F. Office. (2/13/2017). *Observed Weather Reports*. Available: http://w2.weather.gov/climate/index.php?wfo=fgf

[16] "Beaufort Scale," in "Fact Sheet," Available: http://www.metoffice.gov.uk/media/pdf/4/4/Fact_Sheet_No._6_-_Beaufort_Scale.pdf, Accessed on: 2/13/2017.

[17] S. Markovitch and D. Rosenstein, "Feature Generation Using General Constructor Functions," *Machine Learning,* vol. 49, no. 1, pp. 59-98, 2002.

[18] O. Chandrakar and J. R. Saini, "Empirical Study to Suggest Optimal Classification Techniques for Given Dataset," in *Computational Intelligence & Communication Technology (CICT), 2015 IEEE International Conference on*, 2015, pp. 30-35.

[19] P. W. Eklund and A. Hoang, "A performance survey of public domain supervised machine learning algorithms," *Australian Journal of Intelligent Information Systems. v9 i1,* pp. 1-47, 2002.

[20] I. H. Witten, E. Frank, and M. A. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, 3rd ed. Burlington, MA: Morgan Kaufmann, 2011.

[21] Y. Yang and G. I. Webb, "On Why Discretization Works for Naive-Bayes Classifiers," in *AI 2003: Advances in Artificial Intelligence: 16th Australian Conference on AI, Perth, Australia, December 3-5, 2003. Proceedings*, T. D. Gedeon and L. C. C. Fung, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 440-452.

[22] F. V. Jensen, *Introduction to Bayesian networks*. University of Aalborg, Institute for Electronic Systems, Department of Mathematics and Computer Science, 1994.

[23] D. Heckerman, "A Tutorial on Learning with Bayesian Networks," in *Learning in Graphical Models*, M. I. Jordan, Ed. Dordrecht: Springer Netherlands, 1998, pp. 301-354.

[24] S. K. Murthy, "Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey," *Data Min. Knowl. Discov.,* vol. 2, no. 4, pp. 345-389, 1998.

[25] J. R. Quinlan, *C4.5: Programs for Machine Learning*. San Mateo, California: Morgan Kaufmann, 1993.

[26] M. V. Datla, "Bench Marking of Classification Algorithms: Decision Trees and Random Forests - A Case Study using R," (in English), *2015 International Conference on Trends in Automation, Communications and Computing Technology (I-Tact-15)*, Proceedings Paper p. 7, 2015.

[27] T.-S. Lim, W.-Y. Loh, and Y.-S. Shih, "A Comparison of Prediction Accuracy, Complexity, and Training Time of Thirty-Three Old and New Classification Algorithms," *Machine Learning*, journal article vol. 40, no. 3, pp. 203-228, 2000.

[28] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10-18, 2009.

[29] (2/13/2017). *Great Rides Bike Share*. Available: https://greatrides.bcycle.com/

[30] G. P. Zhang, "Neural networks for classification: a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 30, no. 4, pp. 451-462, 2000.

[31] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18-28, 1998.

[32] F. Roli, G. Giacinto, and G. Vernazza, "Methods for designing multiple classifier systems," in *International Workshop on Multiple Classifier Systems*, 2001, pp. 78-87: Springer.

[33] L. Breiman, "Bagging predictors," *Machine Learning,* journal article vol. 24, no. 2, pp. 123-140, 1996.

[34] Y. Freund and R. E. Schapire, "A desicion-theoretic generalization of on-line learning and an application to boosting," in *Computational Learning Theory: Second European Conference, EuroCOLT '95 Barcelona, Spain, March 13–15, 1995 Proceedings*, P. Vitányi, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1995, pp. 23-37.

[35] H. Tin Kam, "The random subspace method for constructing decision forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 20, no. 8, pp. 832-844, 1998.

[36] L. Wang, S. Yuan, L. Li, and H. Li, "Improving the Performance of Decision Tree: A Hybrid Approach," in *Conceptual Modeling – ER 2004: 23rd International Conference on Conceptual Modeling, Shanghai, China, November 8-12, 2004. Proceedings*, P. Atzeni, W. Chu, H. Lu, S. Zhou, and T.-W. Ling, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 327-335.

APPENDIX A. 2015 GREAT RIDES TRAINING DATASET

Workday/ Holiday	Day of the Week	Month	Season	Precipitation	Discretized Temperature Bins	Average Wind Speed Discretized	Weather conditions discretized	Number of Rides(Binary)
Holiday	GroupSS	3	winter	No Rain	Group6	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	3	winter	No Rain	Group6	Gentle breeze	optimal	LessthanAverage
Workday	GroupTR	3	winter	No Rain	Group6	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	3	winter	No Rain	Group6	Gentle breeze	Manageable	LessthanAverage
Holiday	GroupSS	3	winter	No Rain	Group6	Gentle breeze	Sub-optimal	LessthanAverage
Holiday	GroupSS	3	winter	No Rain	Group4	Moderate breeze	optimal	LessthanAverage
Workday	GroupMWF	3	winter	Light Rain	Group6	Moderate breeze	optimal	LessthanAverage
Workday	GroupTR	3	winter	No Rain	Group6	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	3	winter	Heavy Rain	Group6	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupTR	3	winter	Heavy Rain	Group6	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	3	winter	Heavy Rain	Group6	Gentle breeze	Manageable	LessthanAverage
Holiday	GroupSS	3	Spring	No Rain	Group6	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	3	Spring	Heavy Rain	Group6	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	3	Spring	Heavy Rain	Group6	Moderate breeze	optimal	LessthanAverage
Workday	GroupTR	3	Spring	Heavy Rain	Group6	Fresh breeze	Manageable	LessthanAverage
Workday	GroupMWF	3	Spring	Light Rain	Group6	Fresh breeze	Manageable	LessthanAverage
Workday	GroupTR	3	Spring	Heavy Rain	Group6	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	3	Spring	Heavy Rain	Group6	Light breeze	optimal	LessthanAverage
Holiday	GroupSS	3	Spring	Heavy Rain	Group6	Fresh breeze	optimal	MorethanAverage
Holiday	GroupSS	3	Spring	Moderate Rain	Group6	Fresh breeze	Sub-optimal	MorethanAverage
Workday	GroupMWF	3	Spring	No Rain	Group6	Light breeze	optimal	MorethanAverage
Workday	GroupTR	3	Spring	No Rain	Group6	Light breeze	optimal	MorethanAverage
Workday	GroupMWF	4	Spring	No Rain	Group5	Moderate breeze	optimal	MorethanAverage
Workday	GroupTR	4	Spring	No Rain	Group6	Fresh breeze	optimal	LessthanAverage
Workday	GroupMWF	4	Spring	No Rain	Group6	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	4	Spring	No Rain	Group6	Light breeze	optimal	LessthanAverage
Holiday	GroupSS	4	Spring	No Rain	Group6	Moderate breeze	optimal	LessthanAverage
Workday	GroupMWF	4	Spring	No Rain	Group6	Moderate breeze	optimal	LessthanAverage
Workday	GroupTR	4	Spring	Moderate Rain	Group6	Gentle breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	4	Spring	No Rain	Group6	Gentle breeze	optimal	MorethanAverage

Workday/ Holiday	Day of the Week	Month	Season	Precipitation	Discretized Temperature Bins	Average Wind Speed Discretized	Weather conditions discretized	Number of Rides(Binary)
Workday	GroupTR	4	Spring	No Rain	Group5	Moderate breeze	optimal	MorethanAverage
Workday	GroupMWF	4	Spring	Heavy Rain	Group6	Gentle breeze	Sub-optimal	MorethanAverage
Holiday	GroupSS	4	Spring	No Rain	Group5	Moderate breeze	Manageable	MorethanAverage
Holiday	GroupSS	4	Spring	No Rain	Group5	Moderate breeze	optimal	MorethanAverage
Workday	GroupMWF	4	Spring	No Rain	Group5	Moderate breeze	optimal	MorethanAverage
Workday	GroupTR	4	Spring	No Rain	Group5	Moderate breeze	optimal	MorethanAverage
Workday	GroupMWF	4	Spring	No Rain	Group4	Fresh breeze	optimal	MorethanAverage
Workday	GroupTR	4	Spring	Heavy Rain	Group5	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	4	Spring	No Rain	Group5	Gentle breeze	optimal	MorethanAverage
Holiday	GroupSS	4	Spring	No Rain	Group5	Gentle breeze	optimal	MorethanAverage
Holiday	GroupSS	4	Spring	Moderate Rain	Group5	Moderate breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	4	Spring	Light Rain	Group6	Fresh breeze	Sub-optimal	LessthanAverage
Workday	GroupTR	4	Spring	Heavy Rain	Group6	Fresh breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	4	Spring	No Rain	Group6	Moderate breeze	optimal	MorethanAverage
Workday	GroupTR	4	Spring	Light Rain	Group6	Light breeze	optimal	MorethanAverage
Workday	GroupMWF	4	Spring	Moderate Rain	Group6	Moderate breeze	optimal	LessthanAverage
Holiday	GroupSS	4	Spring	No Rain	Group5	Gentle breeze	optimal	MorethanAverage
Holiday	GroupSS	4	Spring	No Rain	Group5	Light breeze	optimal	MorethanAverage
Workday	GroupMWF	4	Spring	Heavy Rain	Group5	Gentle breeze	optimal	MorethanAverage
Workday	GroupTR	4	Spring	Moderate Rain	Group5	Gentle breeze	Sub-optimal	MorethanAverage
Workday	GroupMWF	4	Spring	No Rain	Group5	Light breeze	optimal	MorethanAverage
Workday	GroupTR	4	Spring	Light Rain	Group4	Moderate breeze	optimal	MorethanAverage
Workday	GroupMWF	5	Spring	Light Rain	Group4	Light breeze	Sub-optimal	MorethanAverage
Holiday	GroupSS	5	Spring	Light Rain	Group3	Moderate breeze	optimal	MorethanAverage
Holiday	GroupSS	5	Spring	No Rain	Group4	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	5	Spring	No Rain	Group5	Light breeze	optimal	MorethanAverage
Workday	GroupTR	5	Spring	No Rain	Group4	Moderate breeze	optimal	MorethanAverage
Workday	GroupMWF	5	Spring	Moderate Rain	Group4	Moderate breeze	Manageable	LessthanAverage
Workday	GroupTR	5	Spring	Moderate Rain	Group5	Moderate breeze	Sub-optimal	MorethanAverage
Workday	GroupMWF	5	Spring	No Rain	Group6	Moderate breeze	optimal	LessthanAverage

Workday/ Holiday	Day of the Week	Month	Season	Precipitation	Discretized Temperature Bins	Average Wind Speed Discretized	Weather conditions discretized	Number of Rides(Binary)
Holiday	GroupSS	5	Spring	No Rain	Group6	Moderate breeze	Manageable	MorethanAverage
Holiday	GroupSS	5	Spring	Moderate Rain	Group6	Moderate breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	5	Spring	Moderate Rain	Group6	Fresh breeze	Sub-optimal	LessthanAverage
Workday	GroupTR	5	Spring	Light Rain	Group6	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	5	Spring	Moderate Rain	Group5	Moderate breeze	Sub-optimal	LessthanAverage
Workday	GroupTR	5	Spring	Moderate Rain	Group5	Moderate breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	5	Spring	Light Rain	Group5	Light breeze	Sub-optimal	LessthanAverage
Holiday	GroupSS	5	Spring	No Rain	Group4	Moderate breeze	optimal	LessthanAverage
Holiday	GroupSS	5	Spring	Heavy Rain	Group5	Moderate breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	5	Spring	Moderate Rain	Group6	Fresh breeze	Sub-optimal	LessthanAverage
Workday	GroupTR	5	Spring	No Rain	Group6	Light breeze	manageable	LessthanAverage
Workday	GroupMWF	5	Spring	No Rain	Group5	Light breeze	optimal	LessthanAverage
Workday	GroupTR	5	Spring	No Rain	Group5	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	5	Spring	No Rain	Group4	Light breeze	optimal	LessthanAverage
Holiday	GroupSS	5	Spring	No Rain	Group4	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	5	Spring	No Rain	Group4	Light breeze	optimal	LessthanAverage
Holiday	GroupMWF	5	Spring	Moderate Rain	Group4	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupTR	5	Spring	No Rain	Group4	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	5	Spring	No Rain	Group3	Light breeze	optimal	LessthanAverage
Workday	GroupTR	5	Spring	Moderate Rain	Group2	Gentle breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	5	Spring	Moderate Rain	Group5	Fresh breeze	Sub-optimal	LessthanAverage
Holiday	GroupSS	5	Spring	No Rain	Group6	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	5	Spring	No Rain	Group5	Gentle breeze	Manageable	LessthanAverage
Workday	GroupMWF	6	Spring	No Rain	Group4	Moderate breeze	optimal	LessthanAverage
Workday	GroupTR	6	Spring	Heavy Rain	Group3	Moderate breeze	optimal	LessthanAverage
Workday	GroupMWF	6	Spring	No Rain	Group4	Gentle breeze	Manageable	LessthanAverage
Workday	GroupTR	6	Spring	No Rain	Group3	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	6	Spring	No Rain	Group3	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	6	Spring	Moderate Rain	Group3	Gentle breeze	Sub-optimal	LessthanAverage
Holiday	GroupSS	6	Spring	Heavy Rain	Group3	Gentle breeze	Sub-optimal	LessthanAverage

Workday/ Holiday	Day of the Week	Month	Season	Precipitation	Discretized Temperature Bins	Average Wind Speed Discretized	Weather conditions discretized	Number of Rides(Binary)
Workday	GroupMWF	6	Spring	Light Rain	Group3	Light breeze	optimal	LessthanAverage
Workday	GroupTR	6	Spring	Light Rain	Group2	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	6	Spring	No Rain	Group4	Light breeze	Manageable	LessthanAverage
Workday	GroupTR	6	Spring	No Rain	Group3	Light breeze	optimal	LessthanAverage
Workday	GroupMWF	6	Spring	No Rain	Group3	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	6	Spring	No Rain	Group3	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	6	Spring	Moderate Rain	Group4	Gentle breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	6	Spring	No Rain	Group4	Gentle breeze	optimal	LessthanAverage
Workday	GroupTR	6	Spring	Moderate Rain	Group5	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	6	Spring	No Rain	Group4	Light breeze	optimal	LessthanAverage
Workday	GroupTR	6	Spring	Heavy Rain	Group4	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	6	Spring	Light Rain	Group3	Moderate breeze	Sub-optimal	LessthanAverage
Holiday	GroupSS	6	Spring	Moderate Rain	Group3	Gentle breeze	Sub-optimal	LessthanAverage
Holiday	GroupSS	6	Summer	Moderate Rain	Group3	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	6	Summer	Moderate Rain	Group3	Gentle breeze	inclement	LessthanAverage
Workday	GroupTR	6	Summer	No Rain	Group3	Light breeze	optimal	LessthanAverage
Workday	GroupMWF	6	Summer	Moderate Rain	Group3	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupTR	6	Summer	No Rain	Group3	Light air	Sub-optimal	LessthanAverage
Workday	GroupMWF	6	Summer	No Rain	Group3	Light breeze	Sub-optimal	LessthanAverage
Holiday	GroupSS	6	Summer	Light Rain	Group2	Gentle breeze	inclement	LessthanAverage
Holiday	GroupSS	6	Summer	No Rain	Group3	Gentle breeze	Manageable	LessthanAverage
Workday	GroupMWF	6	Summer	No Rain	Group3	Gentle breeze	Manageable	LessthanAverage
Workday	GroupTR	6	Summer	No Rain	Group3	Light air	Manageable	LessthanAverage
Workday	GroupMWF	7	Summer	No Rain	Group3	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupTR	7	Summer	No Rain	Group3	Gentle breeze	optimal	LessthanAverage
Holiday	GroupMWF	7	Summer	No Rain	Group2	Light breeze	Manageable	LessthanAverage
Holiday	GroupSS	7	Summer	No Rain	Group2	Gentle breeze	Manageable	LessthanAverage
Holiday	GroupSS	7	Summer	No Rain	Group2	Gentle breeze	Manageable	LessthanAverage
Workday	GroupMWF	7	Summer	No Rain	Group4	Moderate breeze	Manageable	LessthanAverage
Workday	GroupTR	7	Summer	No Rain	Group4	Light breeze	optimal	LessthanAverage

Workday/ Holiday	Day of the Week	Month	Season	Precipitation	Discretized Temperature Bins	Average Wind Speed Discretized	Weather conditions discretized	Number of Rides(Binary)
Workday	GroupMWF	7	Summer	Moderate Rain	Group4	Gentle breeze	Sub-optimal	LessthanAverage
Workday	GroupTR	7	Summer	No Rain	Group3	Light breeze	Manageable	LessthanAverage
Workday	GroupMWF	7	Summer	No Rain	Group2	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	7	Summer	No Rain	Group2	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	7	Summer	Heavy Rain	Group2	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	7	Summer	Light Rain	Group2	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupTR	7	Summer	Heavy Rain	Group2	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	7	Summer	Light Rain	Group2	Light breeze	optimal	LessthanAverage
Workday	GroupTR	7	Summer	Moderate Rain	Group2	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	7	Summer	Moderate Rain	Group2	Light breeze	inclement	LessthanAverage
Holiday	GroupSS	7	Summer	No Rain	Group3	Moderate breeze	Sub-optimal	LessthanAverage
Holiday	GroupSS	7	Summer	No Rain	Group2	Moderate breeze	optimal	LessthanAverage
Workday	GroupMWF	7	Summer	No Rain	Group3	Gentle breeze	optimal	LessthanAverage
Workday	GroupTR	7	Summer	No Rain	Group3	Light breeze	optimal	LessthanAverage
Workday	GroupMWF	7	Summer	Moderate Rain	Group2	Gentle breeze	Sub-optimal	LessthanAverage
Workday	GroupTR	7	Summer	Light Rain	Group2	Moderate breeze	optimal	LessthanAverage
Workday	GroupMWF	7	Summer	Heavy Rain	Group2	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	7	Summer	Light Rain	Group2	Gentle breeze	Manageable	LessthanAverage
Holiday	GroupSS	7	Summer	No Rain	Group2	Light breeze	Manageable	LessthanAverage
Workday	GroupMWF	7	Summer	No Rain	Group2	Gentle breeze	optimal	LessthanAverage
Workday	GroupTR	7	Summer	Moderate Rain	Group2	Fresh breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	7	Summer	No Rain	Group2	Fresh breeze	optimal	LessthanAverage
Workday	GroupTR	7	Summer	No Rain	Group3	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	7	Summer	No Rain	Group3	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	8	Summer	No Rain	Group2	Light breeze	optimal	LessthanAverage
Holiday	GroupSS	8	Summer	Heavy Rain	Group3	Gentle breeze	Manageable	LessthanAverage
Workday	GroupMWF	8	Summer	No Rain	Group3	Light breeze	optimal	LessthanAverage
Workday	GroupTR	8	Summer	No Rain	Group3	Light breeze	optimal	LessthanAverage
Workday	GroupMWF	8	Summer	Heavy Rain	Group3	Gentle breeze	optimal	LessthanAverage
Workday	GroupTR	8	Summer	Light Rain	Group3	Light breeze	optimal	LessthanAverage

Workday/ Holiday	Day of the Week	Month	Season	Precipitation	Discretized Temperature Bins	Average Wind Speed Discretized	Weather conditions discretized	Number of Rides(Binary)
Workday	GroupMWF	8	Summer	Moderate Rain	Group2	Light breeze	inclement	LessthanAverage
Holiday	GroupSS	8	Summer	Moderate Rain	Group2	Light breeze	Sub-optimal	LessthanAverage
Holiday	GroupSS	8	Summer	Heavy Rain	Group3	Light breeze	optimal	LessthanAverage
Workday	GroupMWF	8	Summer	No Rain	Group3	Light breeze	optimal	LessthanAverage
Workday	GroupTR	8	Summer	No Rain	Group3	Light breeze	optimal	LessthanAverage
Workday	GroupMWF	8	Summer	No Rain	Group2	Moderate breeze	optimal	LessthanAverage
Workday	GroupTR	8	Summer	No Rain	Group2	Light breeze	optimal	LessthanAverage
Workday	GroupMWF	8	Summer	No Rain	Group2	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	8	Summer	Light Rain	Group2	Moderate breeze	Manageable	LessthanAverage
Holiday	GroupSS	8	Summer	Light Rain	Group3	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	8	Summer	No Rain	Group4	Light air	optimal	LessthanAverage
Workday	GroupTR	8	Summer	Moderate Rain	Group4	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	8	Summer	Light Rain	Group4	Gentle breeze	optimal	LessthanAverage
Workday	GroupTR	8	Summer	No Rain	Group4	Light breeze	optimal	LessthanAverage
Workday	GroupMWF	8	Summer	No Rain	Group2	Moderate breeze	optimal	LessthanAverage
Holiday	GroupSS	8	Summer	Moderate Rain	Group3	Moderate breeze	Manageable	LessthanAverage
Holiday	GroupSS	8	Summer	Light Rain	Group4	Fresh breeze	Sub-optimal	MorethanAverage
Workday	GroupMWF	8	Summer	No Rain	Group4	Moderate breeze	optimal	MorethanAverage
Workday	GroupTR	8	Summer	No Rain	Group4	Light breeze	optimal	MorethanAverage
Workday	GroupMWF	8	Summer	No Rain	Group4	Light breeze	optimal	MorethanAverage
Workday	GroupTR	8	Summer	No Rain	Group3	Light breeze	optimal	MorethanAverage
Workday	GroupMWF	8	Summer	No Rain	Group2	Moderate breeze	Sub-optimal	MorethanAverage
Holiday	GroupSS	8	Summer	No Rain	Group2	Gentle breeze	optimal	MorethanAverage
Holiday	GroupSS	8	Summer	No Rain	Group2	Moderate breeze	optimal	MorethanAverage
Workday	GroupMWF	8	Summer	No Rain	Group3	Gentle breeze	Manageable	MorethanAverage
Workday	GroupTR	9	Summer	No Rain	Group2	Light breeze	Sub-optimal	MorethanAverage
Workday	GroupMWF	9	Summer	No Rain	Group2	Gentle breeze	Sub-optimal	MorethanAverage
Workday	GroupTR	9	Summer	No Rain	Group2	Moderate breeze	Manageable	MorethanAverage
Workday	GroupMWF	9	Summer	Light Rain	Group2	Gentle breeze	Sub-optimal	MorethanAverage
Holiday	GroupSS	9	Summer	Moderate Rain	Group2	Light breeze	Sub-optimal	LessthanAverage

Workday/ Holiday	Day of the Week	Month	Season	Precipitation	Discretized Temperature Bins	Average Wind Speed Discretized	Weather conditions discretized	Number of Rides(Binary)
Holiday	GroupSS	9	Summer	Moderate Rain	Group3	Light breeze	Sub-optimal	MorethanAverage
Holiday	GroupMWF	9	Summer	Heavy Rain	Group4	Light breeze	optimal	MorethanAverage
Workday	GroupTR	9	Summer	No Rain	Group4	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	9	Summer	Heavy Rain	Group4	Light breeze	optimal	MorethanAverage
Workday	GroupTR	9	Summer	No Rain	Group5	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	9	Summer	No Rain	Group5	Light breeze	optimal	MorethanAverage
Holiday	GroupSS	9	Summer	No Rain	Group4	Gentle breeze	optimal	MorethanAverage
Holiday	GroupSS	9	Summer	No Rain	Group3	Moderate breeze	optimal	MorethanAverage
Workday	GroupMWF	9	Summer	No Rain	Group3	Gentle breeze	optimal	MorethanAverage
Workday	GroupTR	9	Summer	No Rain	Group2	Moderate breeze	optimal	MorethanAverage
Workday	GroupMWF	9	Summer	No Rain	Group3	Gentle breeze	optimal	MorethanAverage
Workday	GroupTR	9	Summer	Light Rain	Group4	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	9	Summer	Light Rain	Group5	Light breeze	optimal	MorethanAverage
Holiday	GroupSS	9	Summer	No Rain	Group4	Gentle breeze	optimal	MorethanAverage
Holiday	GroupSS	9	Summer	No Rain	Group3	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	9	Summer	No Rain	Group2	Moderate breeze	optimal	MorethanAverage
Workday	GroupTR	9	Summer	Heavy Rain	Group4	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	9	Fall	Moderate Rain	Group5	Light breeze	Sub-optimal	MorethanAverage
Workday	GroupTR	9	Fall	No Rain	Group3	Light air	Manageable	MorethanAverage
Workday	GroupMWF	9	Fall	No Rain	Group3	Gentle breeze	Manageable	MorethanAverage
Holiday	GroupSS	9	Fall	No Rain	Group2	Moderate breeze	optimal	MorethanAverage
Holiday	GroupSS	9	Fall	No Rain	Group3	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	9	Fall	No Rain	Group5	Gentle breeze	optimal	MorethanAverage
Workday	GroupTR	9	Fall	No Rain	Group5	Light breeze	optimal	MorethanAverage
Workday	GroupMWF	9	Fall	No Rain	Group4	Moderate breeze	optimal	MorethanAverage
Workday	GroupTR	10	Fall	No Rain	Group4	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	10	Fall	No Rain	Group5	Gentle breeze	optimal	MorethanAverage
Holiday	GroupSS	10	Fall	No Rain	Group5	Gentle breeze	optimal	MorethanAverage
Holiday	GroupSS	10	Fall	Light Rain	Group5	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	10	Fall	No Rain	Group4	Gentle breeze	optimal	MorethanAverage

Workday/ Holiday	Day of the Week	Month	Season	Precipitation	Discretized Temperature Bins	Average Wind Speed Discretized	Weather conditions discretized	Number of Rides(Binary)
Workday	GroupTR	10	Fall	No Rain	Group5	Light breeze	Manageable	MorethanAverage
Workday	GroupMWF	10	Fall	Light Rain	Group5	Gentle breeze	optimal	MorethanAverage
Workday	GroupTR	10	Fall	No Rain	Group5	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	10	Fall	No Rain	Group5	Light breeze	optimal	MorethanAverage
Holiday	GroupSS	10	Fall	No Rain	Group3	Gentle breeze	optimal	MorethanAverage
Holiday	GroupSS	10	Fall	Heavy Rain	Group2	Gentle breeze	Manageable	MorethanAverage
Holiday	GroupMWF	10	Fall	Light Rain	Group5	Fresh breeze	Sub-optimal	MorethanAverage
Workday	GroupTR	10	Fall	No Rain	Group6	Light breeze	optimal	MorethanAverage
Workday	GroupMWF	10	Fall	No Rain	Group5	Gentle breeze	optimal	MorethanAverage
Workday	GroupTR	10	Fall	No Rain	Group6	Moderate breeze	optimal	MorethanAverage
Workday	GroupMWF	10	Fall	No Rain	Group6	Light breeze	optimal	MorethanAverage
Holiday	GroupSS	10	Fall	No Rain	Group6	Light air	optimal	LessthanAverage
Holiday	GroupSS	10	Fall	No Rain	Group5	Moderate breeze	optimal	LessthanAverage
Workday	GroupMWF	10	Fall	No Rain	Group4	Gentle breeze	Manageable	MorethanAverage
Workday	GroupTR	10	Fall	No Rain	Group4	Gentle breeze	optimal	MorethanAverage
Workday	GroupMWF	10	Fall	No Rain	Group5	Moderate breeze	optimal	MorethanAverage
Workday	GroupTR	10	Fall	No Rain	Group5	Light breeze	optimal	MorethanAverage
Workday	GroupMWF	10	Fall	Moderate Rain	Group5	Moderate breeze	Sub-optimal	LessthanAverage
Holiday	GroupSS	10	Fall	No Rain	Group5	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	10	Fall	No Rain	Group6	Light breeze	Sub-optimal	LessthanAverage
Workday	GroupMWF	10	Fall	Heavy Rain	Group6	Gentle breeze	Manageable	MorethanAverage
Workday	GroupTR	10	Fall	Heavy Rain	Group5	Moderate breeze	Manageable	MorethanAverage
Workday	GroupMWF	10	Fall	Moderate Rain	Group6	Fresh breeze	impossible	LessthanAverage
Workday	GroupTR	10	Fall	No Rain	Group6	Gentle breeze	optimal	LessthanAverage
Workday	GroupMWF	10	Fall	No Rain	Group6	Gentle breeze	optimal	LessthanAverage
Holiday	GroupSS	10	Fall	Light Rain	Group5	Fresh breeze	Sub-optimal	LessthanAverage

Training data set.

APPENDIX B. SAS CODE FOR THE NAÏVE BAYES PREDICTION GRAPH

```
****************************************
************************;
/*   om.sas

Directory: C:\Statistics\userproj\Nekkanti, Om\
Purpose:  RCBD ANOVA using MIXED

Input Data File ------------------------------------------
---
 om.txt - tab delimited text file copied from
Excel

Input Variables ------------------------------------------
----
 Number of Rides
 Unique dates
 Workday/Weekend
 MWF
 TR
 Precipitation
 Average Temperature in F
 Average Wind Speed Discretized
 Weather conditions discretized

Created Variables ----------------------------------------
------

*/
****************************************
************************;
options ls=132 ps=100 formchar="|----|+|---
+=|-/\<>*";

dm "log;clear;output;clear;"; *** Clear old log
and output. ***;
*ods html close;  *** Clear old results. ***;
*ods html;   *** Restart Results Viewer. ***;

title1 'Distribution Check - Om Nekkanti';

data raw;
 infile 'NaiveBayes_2016.txt'
       firstobs=2 dlm='09'x dsd missover;
 input  Date : mmddyy10.  NumRides : $16.
Prediction : $16.;
   if _n_=1 then NumRides=0;
   if NumRides='LessthanAverage' then NR=-1;
    else if NumRides='MorethanAverage' then
NR=1;
   if Prediction='LessthanAverage' then PR=-1;
    else if Prediction='MorethanAverage' then
PR=1;
 ;;;;

proc print;
 format Date date7.;
 title2 'Verify Data';
 run;

proc format;
```

```
 value RA -1='< Average'
          0='Average'
          1='> Average';
 run;
*ods rtf file='omplt.rtf';
ods graphics on;


proc gplot;
 plot NR*Date=1
     PR*Date=2 / overlay  vref=0
                vaxis=axis1 haxis=axis2;
 axis1   order=(-1 to 1 by 1)
        value=(c=black h=1.3)
        minor=none
        label=(c=black h=2.5 a=90 "Number of
Rides");
 axis2   order=('01mar16'd to '01nov16'd by
month)
        label=(c=black h=2.5 'Date')
        value=(c=black h=1.3)
        minor=none;
 *      offset=(1 cm, 1 cm);
 symbol1 v=dot i=needle c=black;
 symbol2 v=star i=needle c=red;
 format NR RA. Date date7.;
* title1 h=3 'Riders Over Time';
 title1 h=1 'NaiveBayes_2016';
 run;
ods rtf close;
```

APPENDIX C. SAS CODE FOR THE BAYESIAN NETWORK PREDICTION GRAPH

```
****************************************
************************;
/*   om.sas

Directory: C:\Statistics\userproj\Nekkanti, Om\

Purpose:  RCBD ANOVA using MIXED


Input Data File ------------------------------------------
---

 om.txt - tab delimited text file copied from
Excel


Input Variables ------------------------------------------
----

 Number of Rides

 Unique dates

 Workday/Weekend

 MWF

 TR

 Precipitation

 Average Temperature in F

 Average Wind Speed Discretized

 Weather conditions discretized


Created Variables ----------------------------------------
------

*/
****************************************
************************;
options ls=132 ps=100 formchar="|----|+|---
+=|-/\<>*";

dm "log;clear;output;clear;"; *** Clear old log
and output. ***;

*ods html close;  *** Clear old results. ***;

*ods html;   *** Restart Results Viewer. ***;


title1 'Distribution Check - Om Nekkanti';


data raw;

 infile 'BayesNet_2016.txt'

       firstobs=2 dlm='09'x dsd missover;

 input  Date : mmddyy10.  NumRides : $16.
Prediction : $16.;

   if _n_=1 then NumRides=0;

   if NumRides='LessthanAverage' then NR=-1;

    else if NumRides='MorethanAverage' then
NR=1;

   if Prediction='LessthanAverage' then PR=-1;

    else if Prediction='MorethanAverage' then
PR=1;

 ;;;;


proc print;

 format Date date7.;

 title2 'Verify Data';

 run;


proc format;

 value RA -1='< Average'

       0='Average'
```

```
          1='> Average';
 run;
*ods rtf file='omplt.rtf';
ods graphics on;


proc gplot;
 plot NR*Date=1
     PR*Date=2 / overlay  vref=0
               vaxis=axis1 haxis=axis2;
 axis1   order=(-1 to 1 by 1)
        value=(c=black h=1.3)
        minor=none
        label=(c=black h=2.5 a=90 "Number of
Rides");
 axis2   order=('01mar16'd to '01nov16'd by
month)
        label=(c=black h=2.5 'Date')
        value=(c=black h=1.3)
        minor=none;
 *      offset=(1 cm, 1 cm);
 symbol1 v=dot i=needle c=black;
 symbol2 v=star i=needle c=red;
 format NR RA. Date date7.;
* title1 h=3 'Riders Over Time';
 title1 h=1 'BayesNet_2016';
 run;
ods rtf close;
```

APPENDIX D. SAS CODE FOR THE C4.8 PREDICTION GRAPH

```
****************************************
************************;

/*   om.sas

Directory: C:\Statistics\userproj\Nekkanti, Om\

Purpose:  RCBD ANOVA using MIXED


Input Data File ----------------------------------------------
---

 om.txt - tab delimited text file copied from
Excel


Input Variables ----------------------------------------------
----

 Number of Rides

 Unique dates

 Workday/Weekend

 MWF

 TR

 Precipitation

 Average Temperature in F

 Average Wind Speed Discretized

 Weather conditions discretized


Created Variables --------------------------------------------
------


*/

****************************************
************************;
```

```
options ls=132 ps=100 formchar="|----|+|---
+=|-/\<>*";


dm "log;clear;output;clear;"; *** Clear old log
and output. ***;

*ods html close;  *** Clear old results. ***;

*ods html;   *** Restart Results Viewer. ***;


title1 'Distribution Check - Om Nekkanti';


data raw;

 infile 'j48_2016.txt'

       firstobs=2 dlm='09'x dsd missover;

 input  Date : mmddyy10.  NumRides : $16.
Prediction : $16.;

   if _n_=1 then NumRides=0;

   if NumRides='LessthanAverage' then NR=-1;

    else if NumRides='MorethanAverage' then
NR=1;

   if Prediction='LessthanAverage' then PR=-1;

    else if Prediction='MorethanAverage' then
PR=1;

 ;;;;


proc print;

 format Date date7.;

 title2 'Verify Data';

 run;


proc format;
```

```
 value RA -1='< Average'
         0='Average'
         1='> Average';
 run;
*ods rtf file='omplt.rtf';
ods graphics on;


proc gplot;
 plot NR*Date=1
     PR*Date=2 / overlay  vref=0
              vaxis=axis1 haxis=axis2;
 axis1   order=(-1 to 1 by 1)
        value=(c=black h=1.3)
        minor=none
        label=(c=black h=2.5 a=90 "Number of
Rides");
 axis2   order=('01mar16'd to '01nov16'd by
month)
        label=(c=black h=2.5 'Date')
        value=(c=black h=1.3)
        minor=none;
 *      offset=(1 cm, 1 cm);
 symbol1 v=dot i=needle c=black;
 symbol2 v=star i=needle c=red;
 format NR RA. Date date7.;
* title1 h=3 'Riders Over Time';
 title1 h=1 'j48_2016';
 run;
ods rtf close;
```