Bike Sharing Demand

Mughundhan Chandrasekar

9/25/2017

1. About the Project

Bike sharing systems are a means of renting bicycles where the process of obtaining membership, rental, and bike return is automated via a network of kiosk locations throughout a city. Using these systems, people are able rent a bike from one location and return it to a different place on an as-needed basis.

The data generated by these systems makes them attractive for researchers because the **duration of travel, departure location, arrival location, and time elapsed** is explicitly recorded. Bike sharing systems therefore function as a sensor network, which can be used for studying mobility in a city. In this competition, participants were asked to ***combine historical usage patterns with weather data in order to forecast bike rental demand*** in the Capital Bikeshare program in Washington, D.C.

The project aims to Forecast the use of a city bikeshare system i.e. to ***predict the total count of bikes rented during each hour*** covered by the test set.

Kaggle Score: 0.40812 Ranking

```
##      Kaggle_Score Number_of_Participants Kaggle_Rank Among_Top_Percentile
## [1,]      0.40812                   3252         311           0.09563346
```

2. Hypotheses Generation

- **Hourly trend**: There must be high demand during office timings. Early morning and late evening can have different trend (cyclist) and low demand during 10:00 pm to 4:00 am.
- **Daily Trend**: Registered users demand more bike on weekdays as compared to weekend or holiday.
- **Rain**: The demand of bikes will be lower on a rainy day as compared to a sunny day. Similarly, higher humidity will cause to lower the demand and vice versa.
- **Temperature**: In India, temperature has negative correlation with bike demand. But, after looking at Washington???s temperature graph, I presume it may have positive correlation.
- **Pollution**: If the pollution level in a city starts soaring, people may start using Bike (it may be influenced by government / company policies or increased awareness).
- **Time**: Total demand should have higher contribution of registered user as compared to casual because registered user base would increase over time.

- **Traffic**: It can be positively correlated with Bike demand. Higher traffic may force people to use bike as compared to other road transport medium like car, taxi etc

3. About the Dataset

3.1. Independent Variables

1. **datetime**: date and hour in "mm/dd/yyyy hh:mm" format
2. **season**: Four categories-> 1 = spring, 2 = summer, 3 = fall, 4 = winter
3. **holiday**: whether the day is a holiday or not (1/0)
4. **workingday**: whether the day is neither a weekend nor holiday (1/0)
5. **weather**: Four Categories of weather 1-> Clear, Few clouds, Partly cloudy, Partly cloudy 2-> Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist 3-> Light Snow and Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds 4-> Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
6. **temp**: hourly temperature in Celsius
7. **atemp**: "feels like" temperature in Celsius
8. **humidity**: relative humidity
9. **windspeed**: wind speed

3.2. Dependent Variables

10. **registered**: number of registered user
11. **casual**: number of non-registered user
12. **count**: number of total rentals (registered + casual)

4. Creating an appropriate Environment

```
rm(list = ls())
setwd('/Users/Mughundhan/Analytics Vidhya/Rental Biking')
library(lubridate) # for csv files
library(leaflet)   # interactive maps
library(dplyr)     # for piping purpose %>%
#library(rCharts)   # route-map
#library(rMaps)     # route-map
library(data.table)# aggregate
library(ggplot2)   # barplot
library(mice)      # imputing with plausible data values (drawn from a 
distribution specifically designed for each missing datapoint)
#install.packages("rCharts", "rMaps", "data.table", "ggplot2", "mice")
#install.packages("rattle", dep=c("Suggests"))
library(rpart)     #Decision Tree Model
#library(rattle)    #Good visual plot for the decision tree model.
library(rpart.plot)
library(RColorBrewer)
library(MASS)      #Random Forest
library(randomForest)
library(corrplot) #Informative Correlation Plot

train <- read.csv("train.csv", header=T, na.strings=c("","NA")) #Empty spaces
```

```
to be replaced by NA
test <- read.csv("test.csv", header=T, na.strings=c("","NA"))
```

5. Basic Data Exploration

5.1. Combining both test and train dataset and Identify final structure.

Add or remove columns to adjust the structure of dataset in-order to facilitate the join.

```
test$registered=0
test$casual=0
test$count=0
fdata=rbind(train,test)
str(fdata)
```

```
## 'data.frame':    17379 obs. of  12 variables:
##  $ datetime  : Factor w/ 17379 levels "2011-01-01 00:00:00",..: 1 2 3 4 5
6 7 8 9 10 ...
##  $ season    : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ holiday   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ workingday: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ weather   : int  1 1 1 1 1 2 1 1 1 1 ...
##  $ temp      : num  9.84 9.02 9.02 9.84 9.84 ...
##  $ atemp     : num  14.4 13.6 13.6 14.4 14.4 ...
##  $ humidity  : int  81 80 80 75 75 75 80 86 75 76 ...
##  $ windspeed : num  0 0 0 0 0 ...
##  $ casual    : num  3 8 5 3 0 0 2 1 1 8 ...
##  $ registered: num  13 32 27 10 1 1 0 2 7 6 ...
##  $ count     : num  16 40 32 13 1 1 2 3 8 14 ...
```

5.2. Identify Missing Values

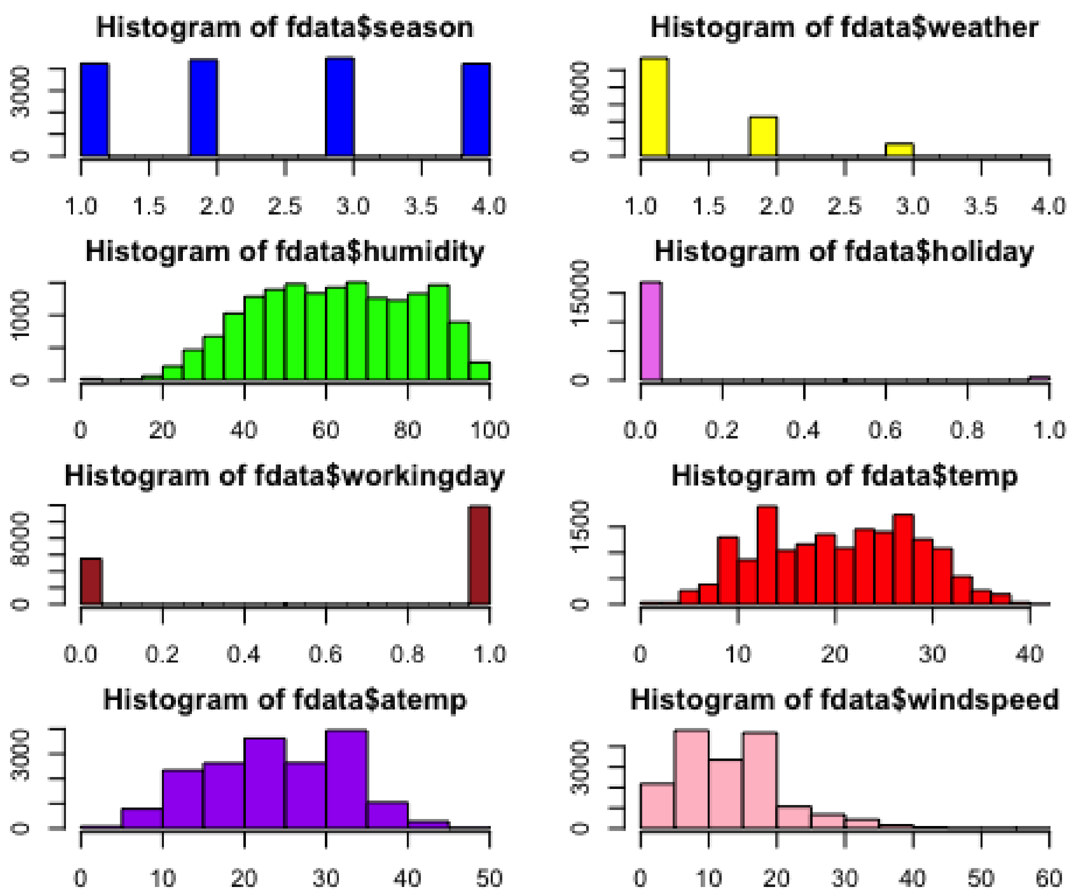
```
##   datetime     season    holiday workingday    weather       temp 
##          0          0          0          0          0          0 
##      atemp   humidity  windspeed     casual registered      count 
##          0          0          0          0          0          0

## 
##  FALSE 
## 208548
```

Observation: There are no missing values in the dataset

5.3. Understand Patterns

```
par(mfrow=c(4,2)) #Fill by rows: Row, Cols
par(mar = rep(2, 4)) #Setting Margins
hist(fdata$season, col="blue")
hist(fdata$weather, col="yellow")
hist(fdata$humidity, col="green")
hist(fdata$holiday, col="violet")
hist(fdata$workingday, col="brown")
hist(fdata$temp, col="red")
```

```
hist(fdata$atemp, col="purple")
hist(fdata$windspeed, col="pink")
```



Observation:

1. **Season** has four categories
2. **Weather-1** contributes the highest
3. Variables *temp, atemp, humidity and windspeed* looks naturally distributed.
4. Deeper look required in working day and holiday to understand the distribution

5.4. Identify the Proportion

```
prop.table(table(fdata$weather))

##
##            1            2            3            4
## 0.6567121238 0.2614649865 0.0816502676 0.0001726221

prop.table(table(fdata$holiday))

##
##          0          1
## 0.97122964 0.02877036

prop.table(table(fdata$workingday))
```

```
##
##         0         1
## 0.3172795 0.6827205
```

5.5. Type-Casting

```
fdata$season=as.factor(fdata$season)
fdata$weather=as.factor(fdata$weather)
fdata$holiday=as.factor(fdata$holiday)
fdata$workingday=as.factor(fdata$workingday)
```

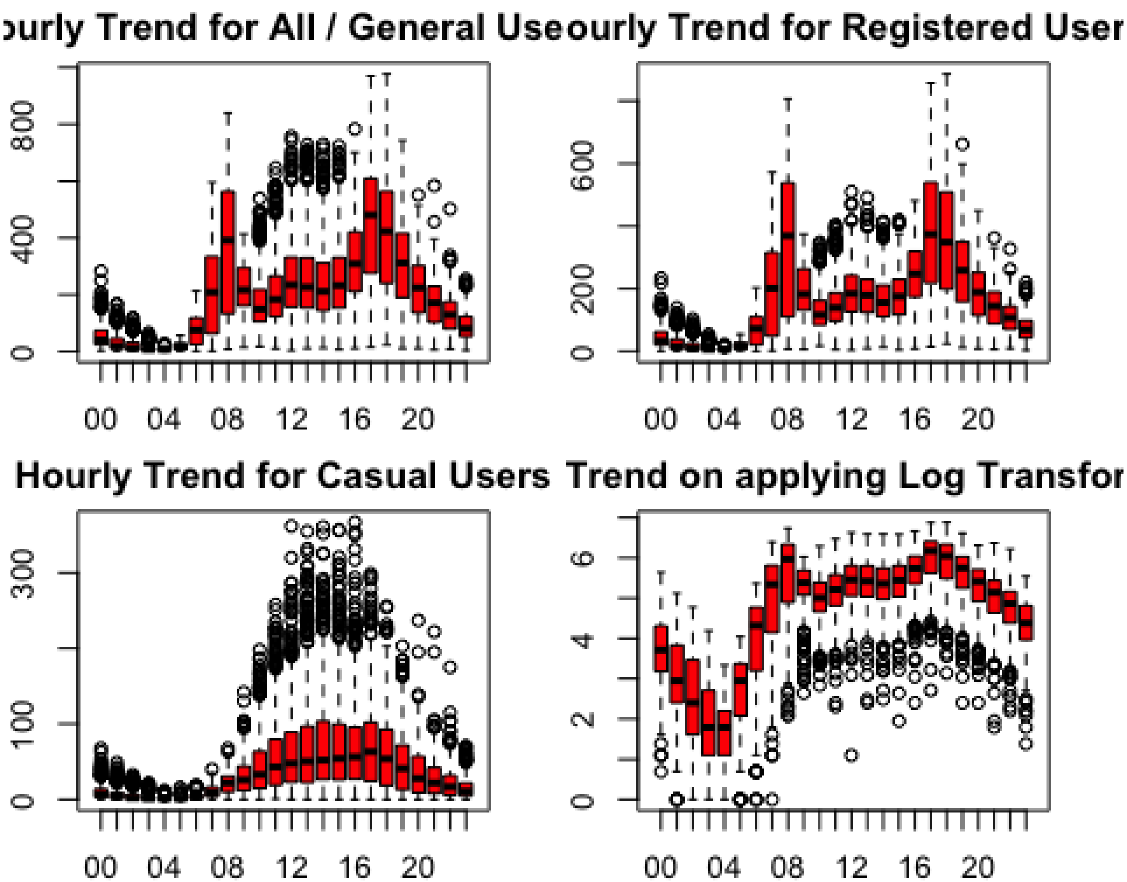
6. Multi-Variate Analysis

This can also be considered as Hypotheses Testing.

6.1. Hourly Trend - Bike Usage

Partitioning data as follows:

1. Train <- First 19 days of every month
2. Test <- Last 10-12 days of every month



Observation:

1. The General usage shall be classified into 3:

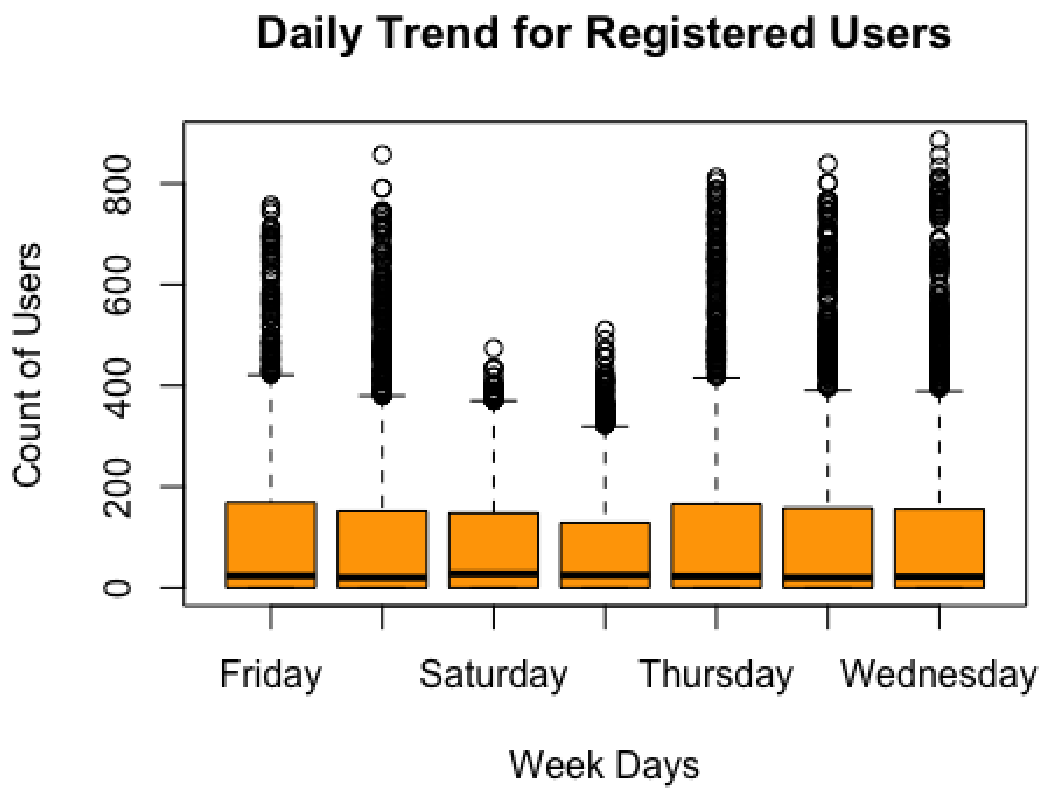
- High: 7-9 and 17-19 hours
- Medium: 10-16 hours
- Low: 0-6 and 20-24 hours

2. The General Users' hourly trend is similar to the Registered Users' Hourly Trend
3. Existence of ***Natural Outliers*** to be treated with ***Logarathmic Transformations*** (taking the logarithm only works if the data is non-negative. Other transforms, such as ***arcsinh***, can be used to decrease data range if we have zero or negative values.)

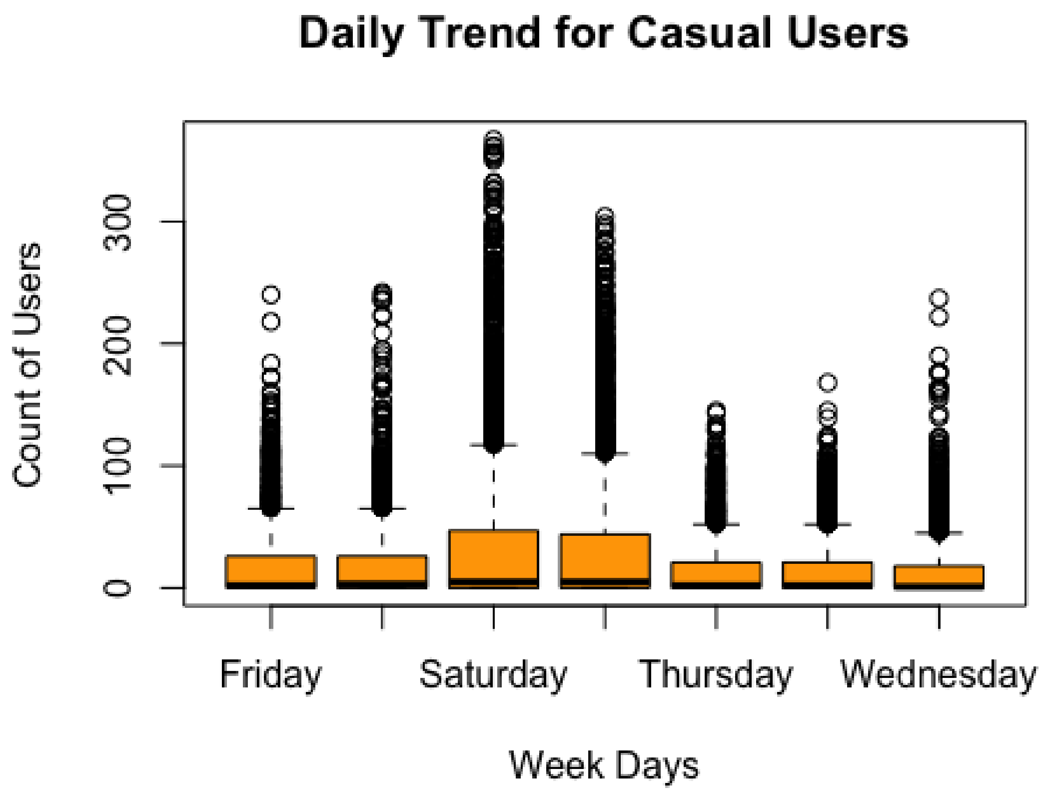
NOTE: Why Logarathmic Transformations?

1. It is generally a good idea to log transform data with values that range over several orders of magnitude. 2. Because Modeling techniques often have a difficult time with very wide data ranges
2. Because such data often comes from multiplicative processes, so log units are in some sense more natural.

6.2. Daily Trend - Bike Usage



Daily Trend for Registered Users
Count of Users
0
200
400
600
800
Friday
Saturday
Thursday
Wednesday
Week Days

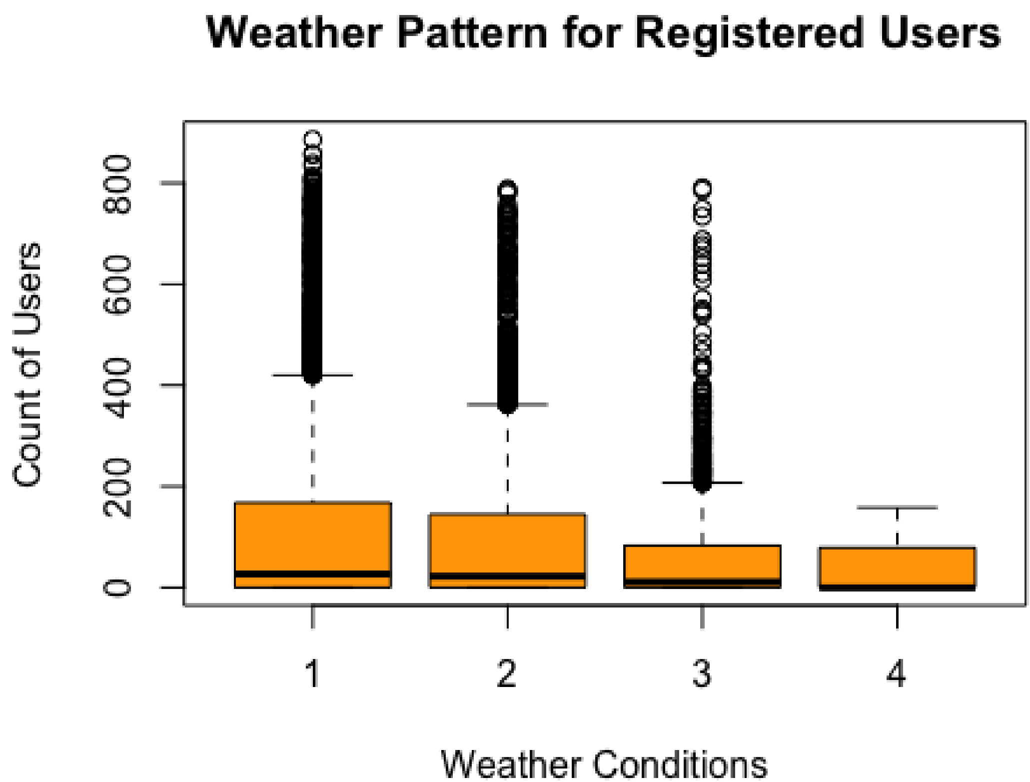


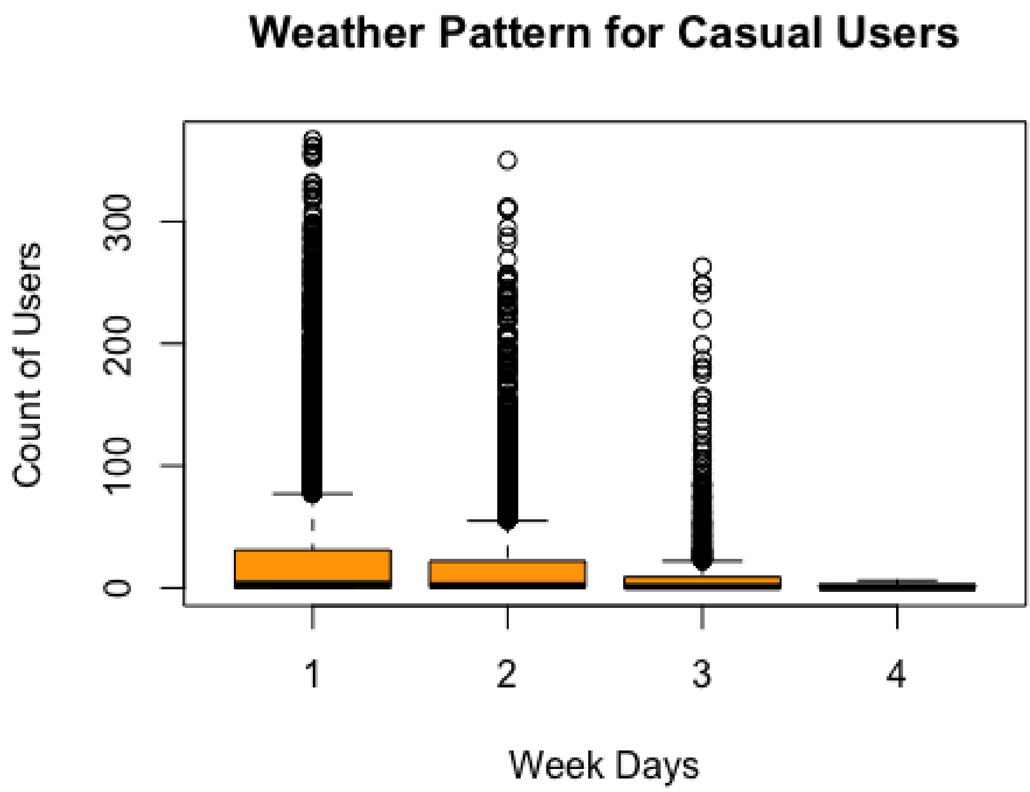
Observation:

1. Demand for bikes by registered users are high on weekdays
2. Demand for bikes by casual users are high on weekends

6.3. Weather Patterns - Bike Usage

1. Weather 1: No Clouds
2. Weather 2: Partly Cloudy
3. Weather 3: Represents light rain
4. Weather 4: Represents heavy rain


Weather Pattern for Registered Users
Count of Users
0
200
400
600
800
1
2
3
4
Weather Conditions

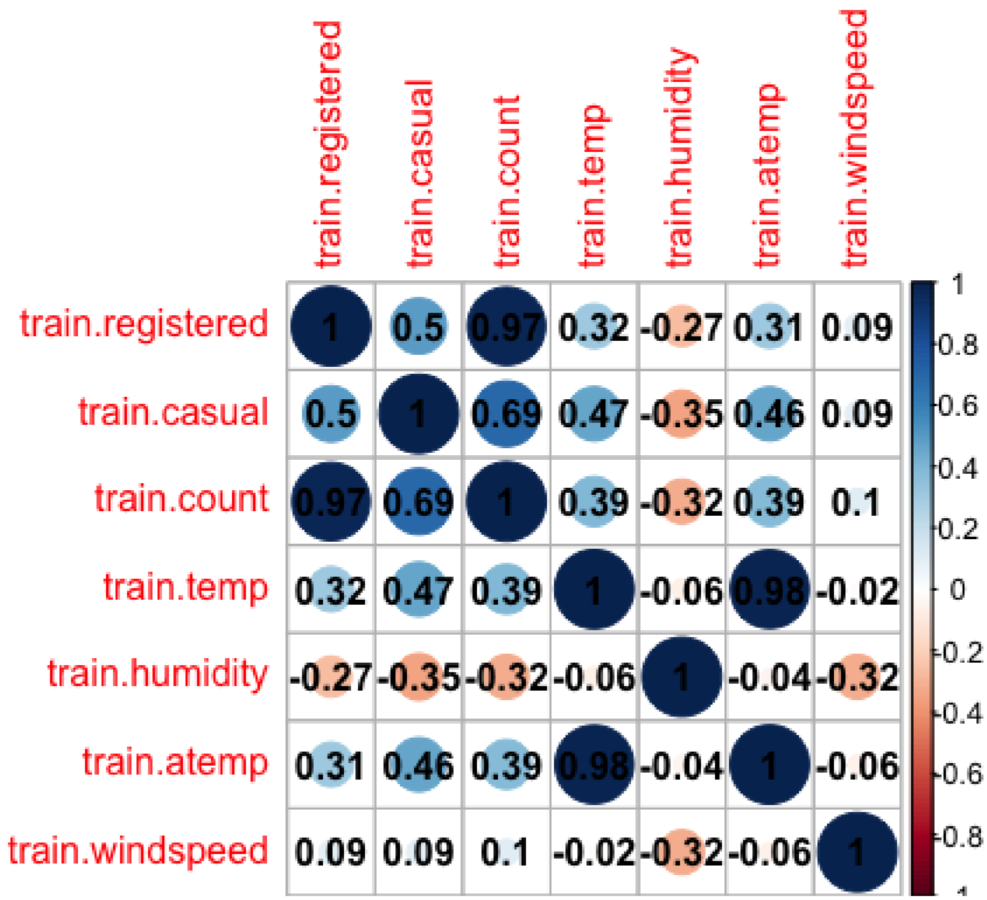


Observation:

1. Demand for bikes by all users are very low on rainy days
2. **Better Weather ~ High Demand**: Demand for bikes by all users is inversely proportional to the rain.

6.4. Temperature, Windspeed and Humidity - Bike Usage

These are continuous variables so we can look at the correlation factor to validate hypothesis.



Observation:

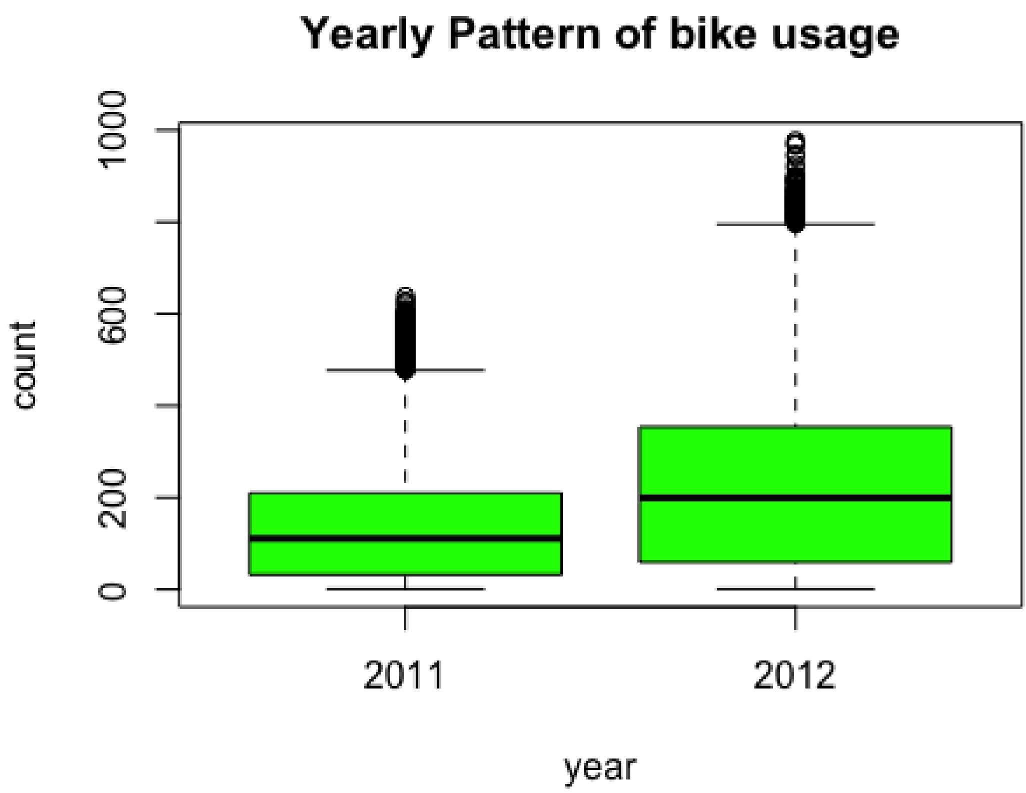
1. Highly Correlated:

- Registered Users and General Users
- Actual Temp or Temp and Casual Users
- Humidity and Users (Negative Correlation)

2. Poorly Correlated:

- Windspeed

6.5. Time - Bike Usage

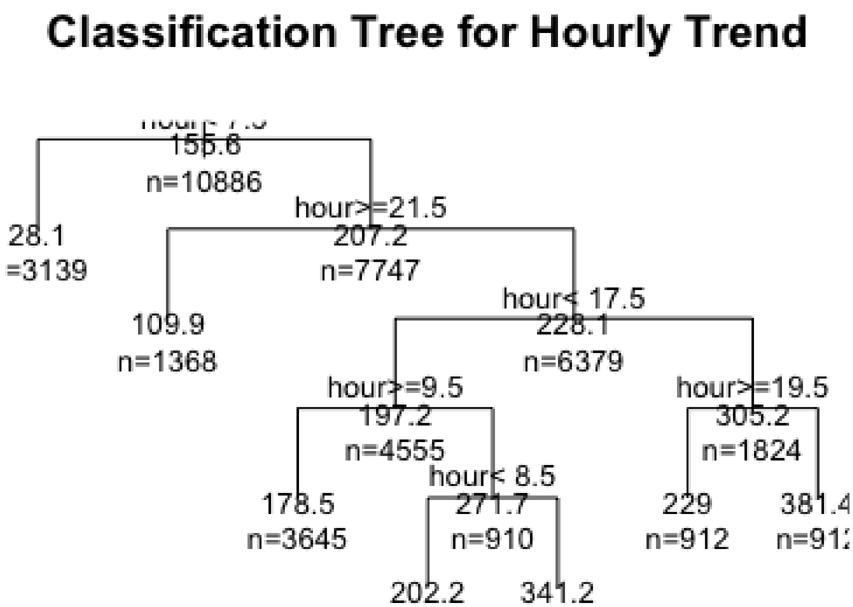


Observation:

2012 has higher bike demand than 2011.

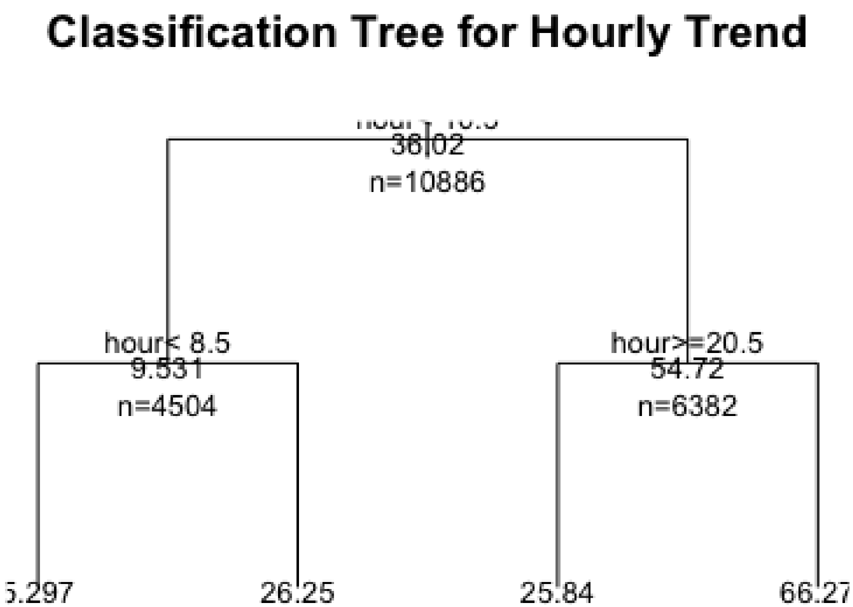
7. Feature Engineering

7.1. Hour Bins

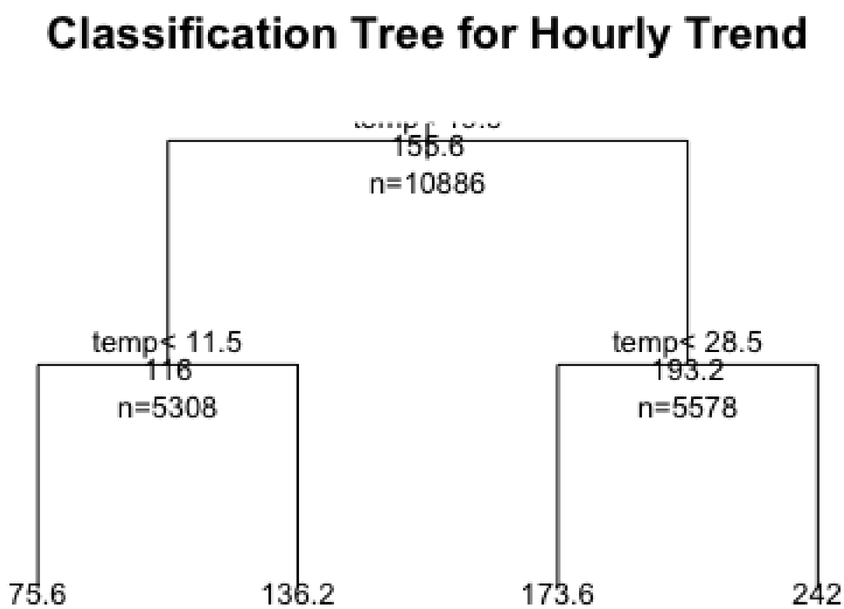


Making use of the splits and converting it into hourly bins for registered users

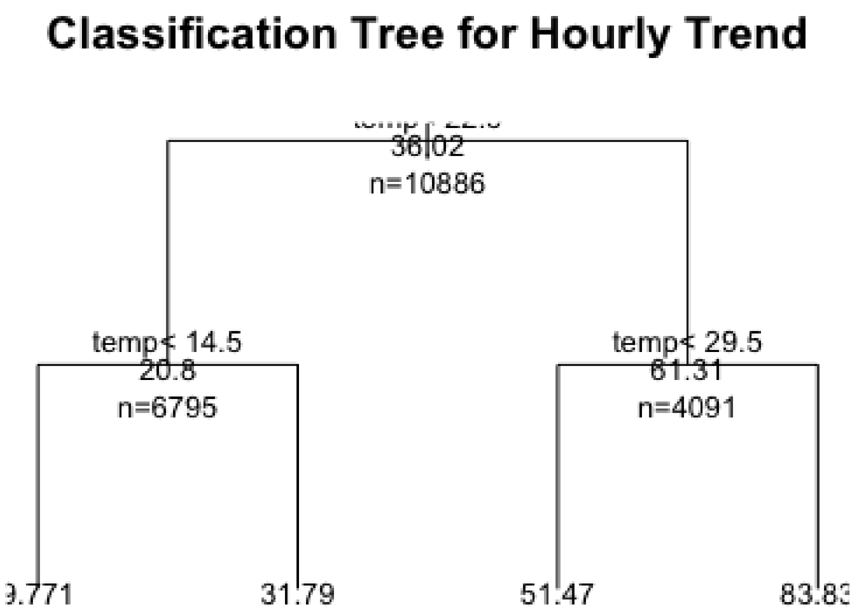
Making use of the splits and converting it into hourly bins for casual users



Making use of the splits and converting it into temperature wise bins



Making use of the splits and converting it into temperature wise bins for Casual Users



Making use of the splits and converting it into yearly-monthly bins

Creating 8 bins (quarterly) for two years

```
##
##         01  02  03  04  05  06  07  08  09  10  11  12
##   2011 688 649 730 719 744 720 744 731 717 743 719 741
##   2012 741 692 743 718 744 720 744 744 720 708 718 742

##
##    1    5
## 8645 8734
```

Making use of the splits and converting it into Day-Type bins

Variable having categories like ???weekday???, ???weekend??? and ???holiday???.

```
##
##     holiday     weekend working day
##         500        5014       11865
```

Making use of the splits and converting it into Weekend bins

Separate variable for weekend (0/1)

8. Model Building

Before executing the random forest model code, I have followed following steps:

1. Convert discrete variables into factor (weather, season, hour, holiday, working day, month, day)
2. As we know that dependent variables have natural outliers so we will predict ***log of dependent variables***.
3. Predict bike demand registered and casual users separately. Here we have added 1 to deal with zero values in the casual and registered columns:

- y1=log(casual+1) and
- y2=log(registered+1)

8.1. Predicting the log of registered users

8.2. Predicting the log of Casual users

8.3. Re-transforming the predicted variables and then writing the output of count to the submission file

9. Submission Format

```
##           datetime count
## 1 20-01-11 0:00:00     8
## 2 20-01-11 1:00:00     5
## 3 20-01-11 2:00:00     3
## 4 20-01-11 3:00:00     3
## 5 20-01-11 4:00:00     3
## 6 20-01-11 5:00:00     5

##              datetime count
## 1 2011-01-20 00:00:00     8
## 2 2011-01-20 01:00:00     5
## 3 2011-01-20 02:00:00     3
## 4 2011-01-20 03:00:00     3
## 5 2011-01-20 04:00:00     3
## 6 2011-01-20 05:00:00     5
```