ML1000 FINANCE GROUP

<u>Assignment 3</u>

<u>Predicting Demand in a Bike Sharing System</u>

22nd March 2019

<u>ML1000 Finance Group</u>

Daniyal Shamim

Juan Calvillo

Mark Lewis

Salman Amin

Table of Contents

1.0 Introduction

Bike sharing is rapidly spreading across major metropolitan areas. They are cheap, environmentally friendly, and reduce congestion in cities that are plagued with long commute times. They offer a quick and convenient method of transportation and are gaining popularity with young commuters and local governments.

To make this more convenient, these services are powered by easy to use technology that makes it really easy for riders to rent bikes. Riders can instantly register and pick up bikes at several rental stations around the city and drop them off near their destination. Their popularity has also attracted venture capital and resulted in large investments in a number of successful start-ups such as Lime, which raised $335 million from a group of investors including ride-sharing firm Uber Technologies Inc and Alphabet Inc[1]. It is estimated that are more than 500 bike-share services around the world with more than 500,000 bicycles[2].

2.0 Business Background & Context

2.1 Background

Capital Bikeshare[3] operates the Bike Sharing program for Washington D.C., and we have been tasked with solving an important problem for Capital Bikeshare. The demand for bike rentals is quite volatile and Capital Bikeshare has been struggling to keep up with this demand. The company owns 500 bikes and a fleet of 10 trucks that transports bikes between different bike rental stations and the warehouse.

Large fluctuations in demand sometimes causes a lot of chaos for Capital Bikeshare. The warehouse gets extremely busy due to the traffic from the fleet of trucks, while there is a large imbalance of bikes at several stations. This also leads to confusion with the renters as they have problems in renting and dropping off the bikes they have rented, which in turn results in a heavy volume of calls for the customer service team.

2.2 Objective

Our objective is to accurately predict the demand of bikes on a given hour on a given day so that Capital Bikeshare is prepared in to manage the fluctuation in demand and respond accordingly. We will apply several data analysis techniques and machine learning algorithms to find out what factors drive demand for bike rentals and build a model to predict the demand based on those factors.

---

[1] Reuters: Uber, Alphabet invest in bike sharing service Lime (2018)

[2] Policy Institute: Bike-sharing programs hit the streets in over 500 cities worldwide. http://www.earth-policy.org/plan_b_updates/2013/update112 (2013)

[3] https://www.capitalbikeshare.com/

3.0 Data Set

For our analysis, we will use the Capital Bike Sharing (CBS)[4] data obtained from the UCI Machine Learning Repository. It contains 17,389 records, which consists of data spanning two years of bike rental information from 2011 to 2012. The dataset is rich with exogenous variables such a weather conditions, holidays, temperature, etc. This makes the research more robust and simply goes beyond the generic numeric variables.

Based on a simple check-in idea by Yexin et el[5], a user can rent a bike at a station near their origin and return (i.e. check in) it to a station close to their destination. Users are required to swipe an RFID card when checking out/in a bike. A record, consisting of the bike ID, timestamp and station ID, is generated for each card swipe. Faney et el[6], states that “these characteristics of data being generated by systems make it attractive for research. The duration of travel, departure and arrival position is explicitly recorded in these systems. This feature turns bike sharing system into a virtual sensor network that can be used for sensing mobility in the city. Hence, it is expected that most of important events in the city could be detected via monitoring this data.”

In addition to other research literature, we have also relied on a Kernel in Kaggle that has also used this dataset for analysis[7].

3.1 Data Dictionary

Table 3.1

Column Name	Type	Column Description
Instant	Integer	Unique ID of each Bike Rental transaction
Dteday	Date	Date of the Bike Rental
Season	Factor	1:spring, 2:summer, 3:fall, 4:winter
Yr	Integer	0: 2011, 1:2012
Mnth	Integer	month (1 to 12)
Hr	Integer	hour (0 to 23)
Holiday	Nominal	a 5-digit integral number uniquely assigned to each customer.
Weekday	Integer	Day of the week
Workingday	Integer	If day is neither weekend nor holiday is 1, otherwise is 0
Weathersit	Factor	1: Clear, Few clouds, Partly cloudy, Partly cloudy - 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist - 3: Light Snow, Light Rain + Thunderstorm + Scattered

[4] https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset
[5] Yexin Li, Yu Zheng , Huichu Zhang, Lei Chen (2016) Traffic Prediction in a Bike-Sharing System Yexin Li, Yu Zheng , Huichu Zhang, Lei Chen
[6] Fanaee-T, H. & Gama, J. Prog Artif Intell (2014) 2: 113. https://doi.org/10.1007/s13748-013-0040-3
[7] https://www.kaggle.com/janellchao/crisp-dm-data-mining-using-r/data

Column Name	Type	Column Description
		clouds, Light Rain + Scattered clouds - 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
Temp	Number	Normalized temperature in Celsius. The values are derived via (t-t_min)/(t_max-t_min), t_min=-8, t_max=+39 (only in hourly scale)
Atemp	Number	Normalized feeling temperature in Celsius. The values are derived via (t-t_min)/(t_max-t_min), t_min=-16, t_max=+50 (only in hourly scale)
Hum	Number	Normalized humidity. The values are divided to 100 (max)
Windspeed	Number	Normalized wind speed. The values are divided to 67 (max)
Casual	Integer	Count of casual users
Registered	Integer	Count of registered users
Cnt	Integer	Count of total rental bikes including both casual and registered

4.0 Data Exploration

We start by loading the dataset into RStudio and look at the first few records to see what it looks like:

```
> str(bike)
> head(bike)
```

```
  instant     dteday season yr mnth hr holiday weekday workingday weathersit temp  atemp  hum windspeed casu
1       1 2011-01-01      1  0    1  0       0       6          0          1 0.24 0.2879 0.81    0.0000
2       2 2011-01-01      1  0    1  1       0       6          0          1 0.22 0.2727 0.80    0.0000
3       3 2011-01-01      1  0    1  2       0       6          0          1 0.22 0.2727 0.80    0.0000
4       4 2011-01-01      1  0    1  3       0       6          0          1 0.24 0.2879 0.75    0.0000
5       5 2011-01-01      1  0    1  4       0       6          0          1 0.24 0.2879 0.75    0.0000
6       6 2011-01-01      1  0    1  5       0       6          0          2 0.24 0.2576 0.75    0.0896
  registered cnt
1         13  16
2         32  40
3         27  32
4         10  13
5          1   1
6          1   1
```

Next, we check to see if there are any blank records:

```
> sum(is.na(bike))
[1] 0
```

Since there aren't any blank records, we dig in further and see an overview of all the variables:

```
> summary(bike)
    instant            dteday          season            yr              mnth              hr             holiday       
 Min.   :    1   2011-01-01:   24   Min.   :1.000   Min.   :0.0000   Min.   : 1.000   Min.   : 0.00   Min.   :0.00000  
 1st Qu.: 4346   2011-01-08:   24   1st Qu.:2.000   1st Qu.:0.0000   1st Qu.: 4.000   1st Qu.: 6.00   1st Qu.:0.00000  
 Median : 8690   2011-01-09:   24   Median :3.000   Median :1.0000   Median : 7.000   Median :12.00   Median :0.00000  
 Mean   : 8690   2011-01-10:   24   Mean   :2.502   Mean   :0.5026   Mean   : 6.538   Mean   :11.55   Mean   :0.02877  
 3rd Qu.:13034   2011-01-13:   24   3rd Qu.:3.000   3rd Qu.:1.0000   3rd Qu.:10.000   3rd Qu.:18.00   3rd Qu.:0.00000  
 Max.   :17379   2011-01-15:   24   Max.   :4.000   Max.   :1.0000   Max.   :12.000   Max.   :23.00   Max.   :1.00000  
                 (Other)   :17235                                                                                      
    weekday        workingday       weathersit         temp           atemp             hum           windspeed     
 Min.   :0.000   Min.   :0.0000   Min.   :1.000   Min.   :0.020   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000  
 1st Qu.:1.000   1st Qu.:0.0000   1st Qu.:1.000   1st Qu.:0.340   1st Qu.:0.3333   1st Qu.:0.4800   1st Qu.:0.1045  
 Median :3.000   Median :1.0000   Median :1.000   Median :0.500   Median :0.4848   Median :0.6300   Median :0.1940  
 Mean   :3.004   Mean   :0.6827   Mean   :1.425   Mean   :0.497   Mean   :0.4758   Mean   :0.6272   Mean   :0.1901  
 3rd Qu.:5.000   3rd Qu.:1.0000   3rd Qu.:2.000   3rd Qu.:0.660   3rd Qu.:0.6212   3rd Qu.:0.7800   3rd Qu.:0.2537  
 Max.   :6.000   Max.   :1.0000   Max.   :4.000   Max.   :1.000   Max.   :1.0000   Max.   :1.0000   Max.   :0.8507  

     casual         registered         cnt       
 Min.   :  0.00   Min.   :  0.0   Min.   :  1.0  
 1st Qu.:  4.00   1st Qu.: 34.0   1st Qu.: 40.0  
 Median : 17.00   Median :115.0   Median :142.0  
 Mean   : 35.68   Mean   :153.8   Mean   :189.5  
 3rd Qu.: 48.00   3rd Qu.:220.0   3rd Qu.:281.0  
 Max.   :367.00   Max.   :886.0   Max.   :977.0  
```

We make several important observations just by looking at the Summary data:

- The Dteday column appears to be all strings. We will convert the data in this column to a Date in the Data Preparation section
- The Season column is set as an Integer. We will convert it to a Factor and assign the appropriate labels
- The Weathersit column is set as an Integer and ranges from 1 to 4. Instead of the long descriptions given in the Data Dictionary, we will convert this into a Factor and assign some simple labels
- There are several variables that provide information about weather conditions. While the Weathersit variable categorizes the overall weather conditions in one of four classes, other variables such as Season, Temp, Atemp, Humidity and Windspeed also contain useful information. We will have to explore these variables and their relationships in more detail
- The time of the day and the day itself, whether it is a weekend, holiday, or weekday, may also contain some useful information. We will explore these variables in more detail as well
- The dataset only contains data for years 2011 and 2012, so the specific year of the record may not be as relevant for our analysis
- The number of registered users seems to be significantly higher than casual users.

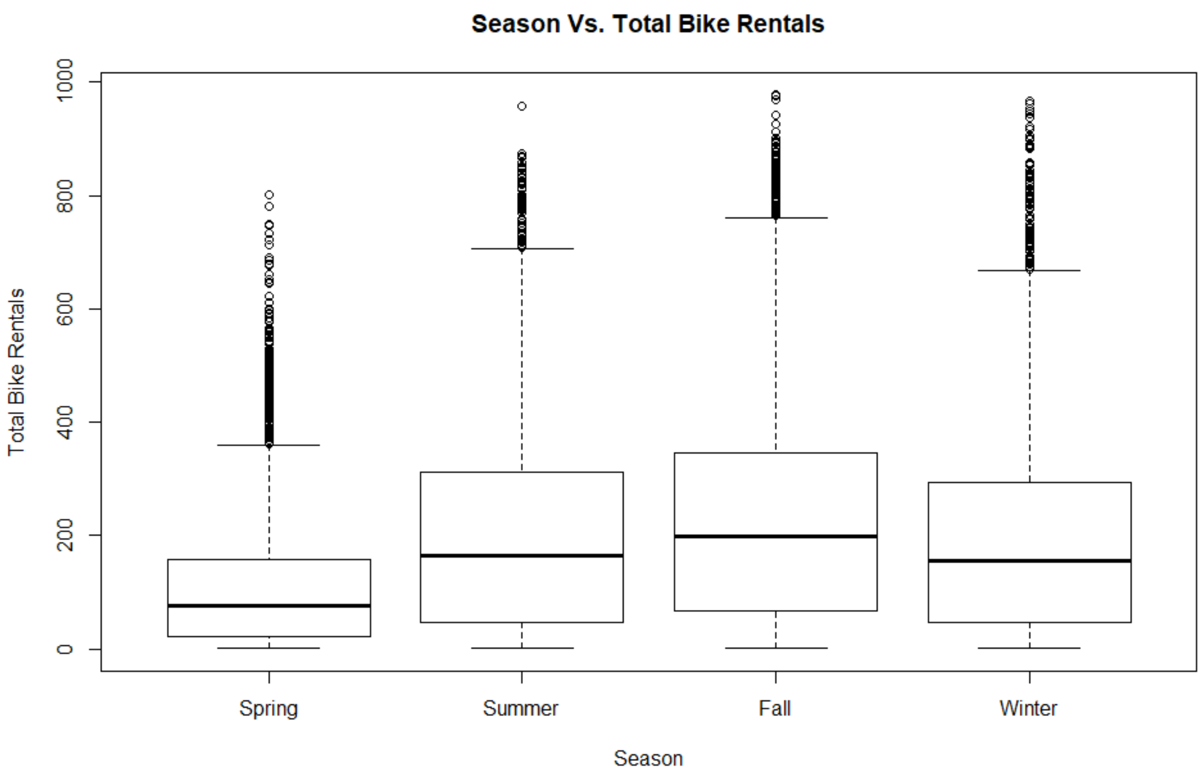
Equipped with some basic knowledge of our dataset, we will now analyze some of the variables more closely to see if we can notice any other trends.

4.1 Season and Month of the Year

First, lets look at the relationship between the Season and Total Bike Rentals, shown in Figure 4.1 below:

```
> plot(bike$season,bike$cnt,main="Season Vs. Total Bike Rentals",xlab="Season
",ylab="Total Bike Rentals")
```

Figure 4.1

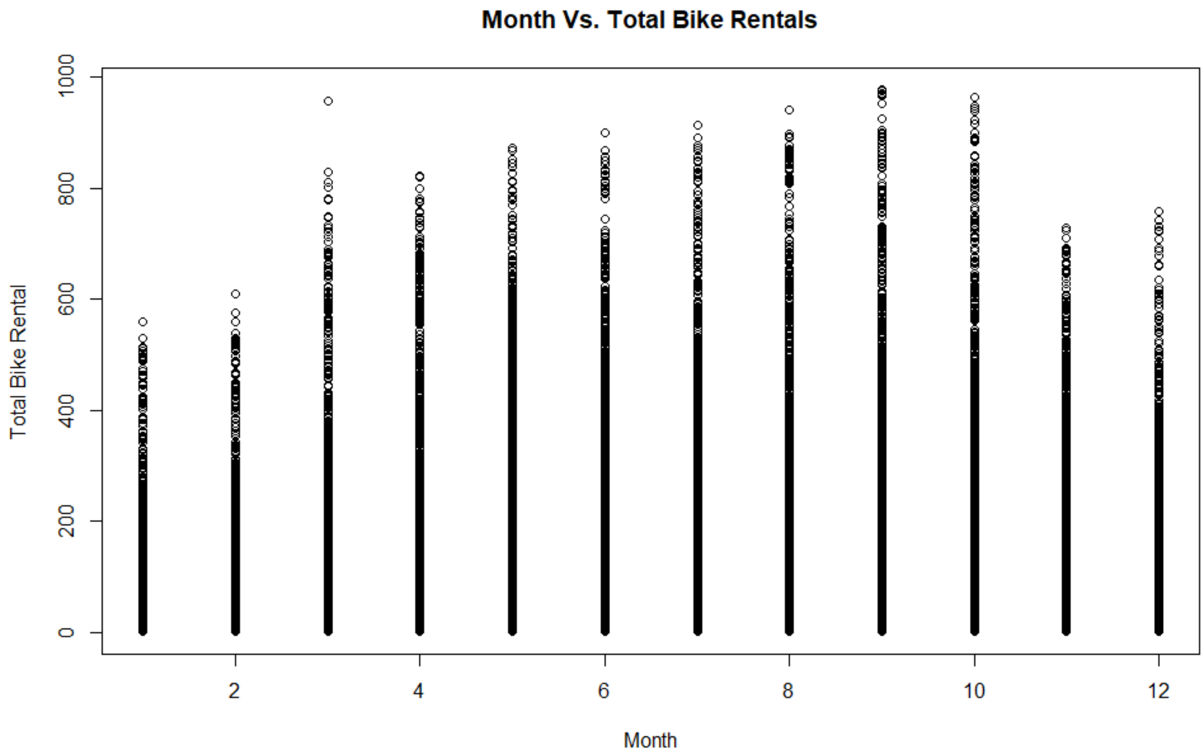


The box plot indicates that the number of bike rentals are the lowest in Spring and start increasing as the weather improves in the Summer. They peak in the Fall and start declining again as Winter approaches.

Since each of the Seasons span several months, lets see how the total number of bikes rented is distributed across several months.

```
> plot(bike$mnth,bike$cnt,main="Month Vs. Total Bike Rentals",xlab="Month",yl
ab="Total Bike Rental")
```

Figure 4.2



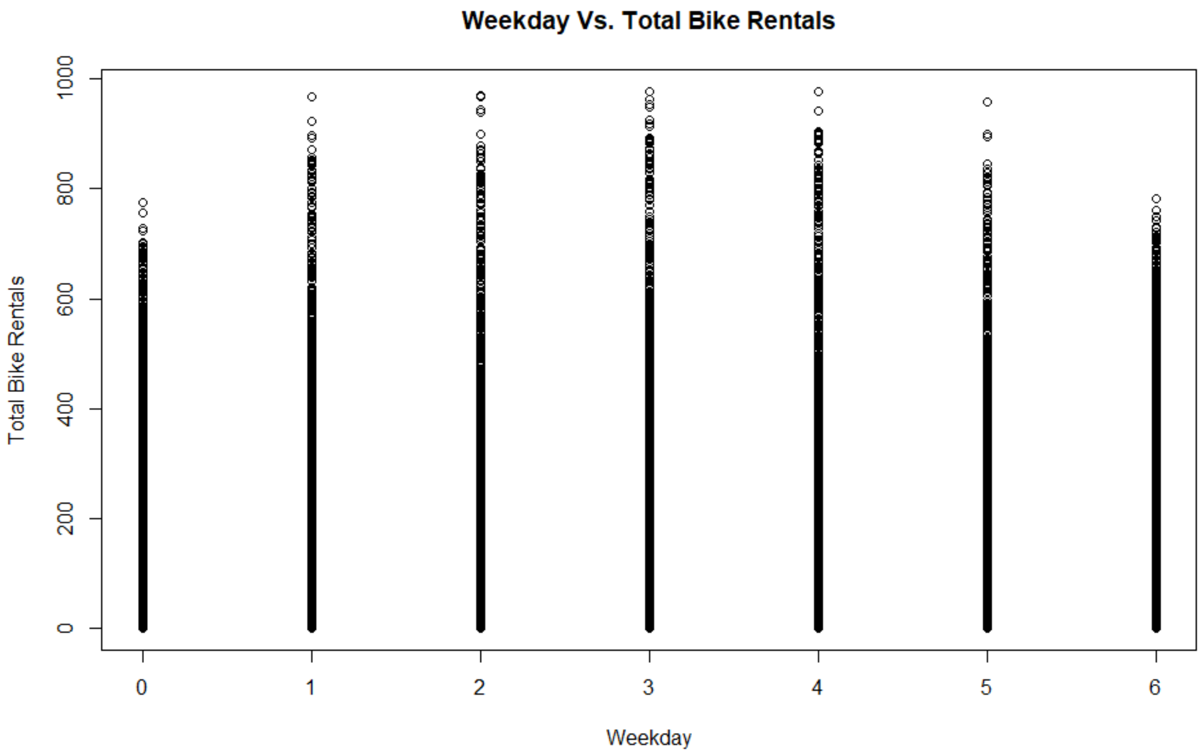
This plot is consistent with the previous one and provides a more granular view of the dataset. Bikes rented are lowest in January and February, which are usually the coldest part of the winter. As the weather improves, the number of bikes rented rises from March to the warmer months and peaks in the months of September and October. After that, it starts declining again as the cold weather comes back.

4.2 Time and Day of the Week

We shorten our times frame and analyze bike rental patterns on different days of the week. Again, we will start by visualising this variable:

```
> plot(bike$weekday,bike$cnt,main="Weekday Vs. Total Bike Rentals",xlab="Week
day",ylab="Total Bike Rentals")
```

Figure 4.3

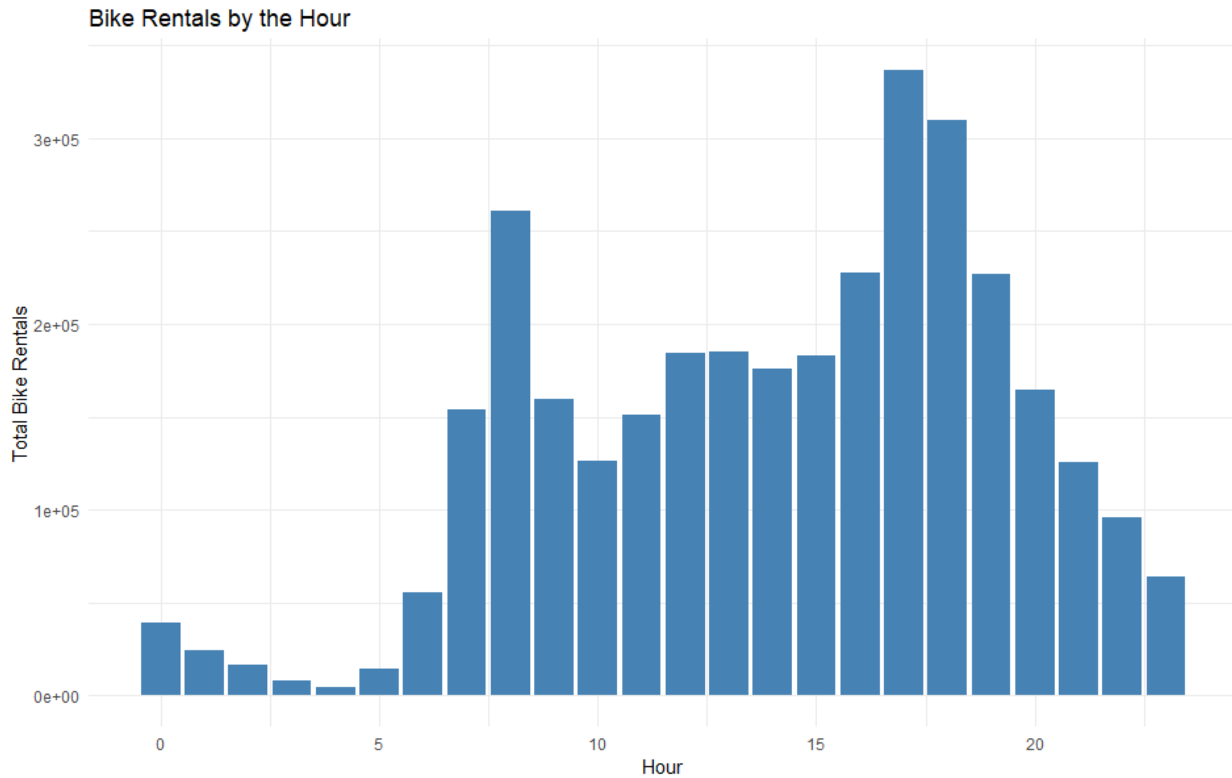


Fewer bike rental occur on days 0 and 6, which represent Sunday and Saturday, respectively. The number of rentals during the weekdays seem to be quite evenly spread. This could be because there are more people renting bikes to commute for work than those renting it for leisure on the weekends.

Let us now shorten the time frame even further and see how the bike rentals are spread out across the hours of the day:

```
> p=ggplot(data=bike, aes(x=hr, y=cnt,fill = hr)) +
+   geom_bar(stat="identity", fill="steelblue")+
+   theme_minimal() + ggtitle("Bike Rentals by the Hour") + xlab("Hour") + yl
ab("Total Bike Rentals")
> p
```

Figure 4.4



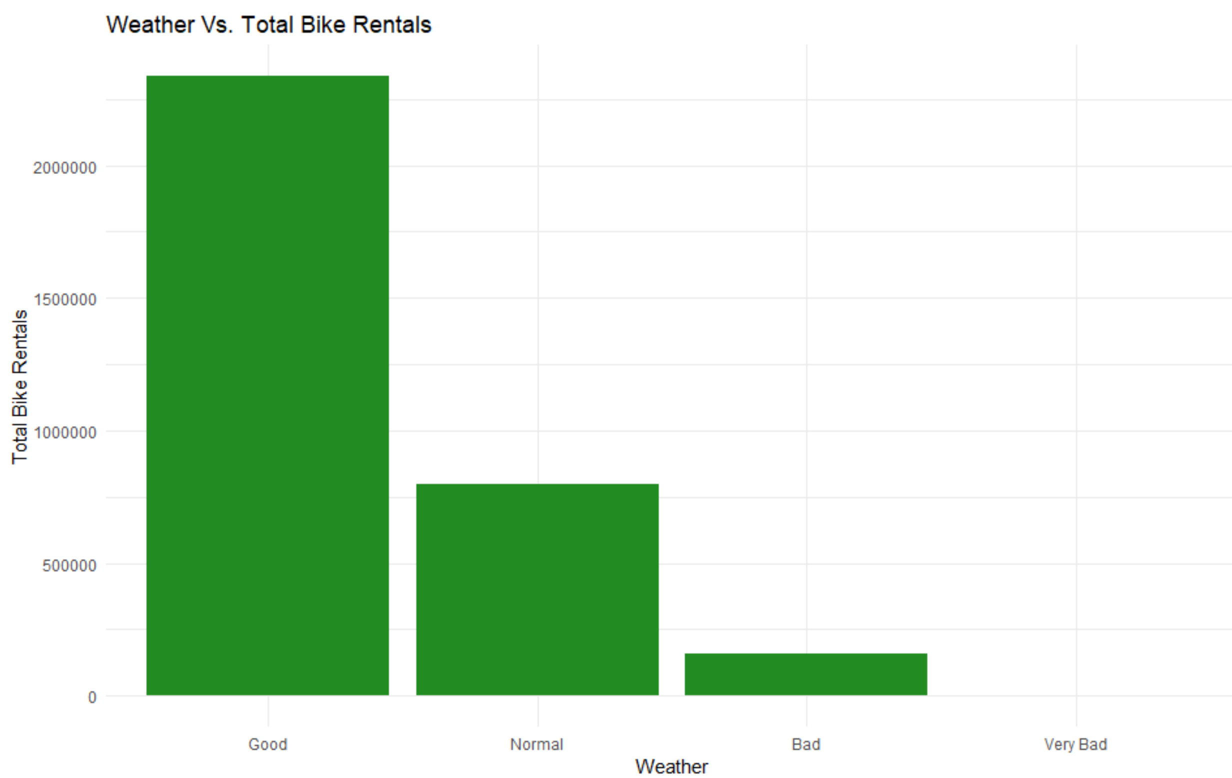
Bike Rentals seem to be lowest from midnight to 6 am. They are highest in the morning at 8 am and then again at 5 pm and 6 pm. Rentals between 8 am and 5 pm seem to be quite evenly distributed. This supports our assumption that the work commute may be a contributing factor of rental demand.

4.3 Weather Conditions

We now turn our attention to the weather conditions and see how they impact the demand of bike rentals. Using the four categories of weather, we look at the distribution of total bike rentals against weather:

```
> p3=ggplot(data=bike, aes(x=weathersit, y=cnt,fill = weathersit)) +
+   geom_bar(stat="identity", fill="forestgreen")+
+   theme_minimal()+ ggtitle("Weather Vs. Total Bike Rentals") + xlab("Weathe
r") + ylab("Total Bike Rentals")
> p3
```

Figure 4.5



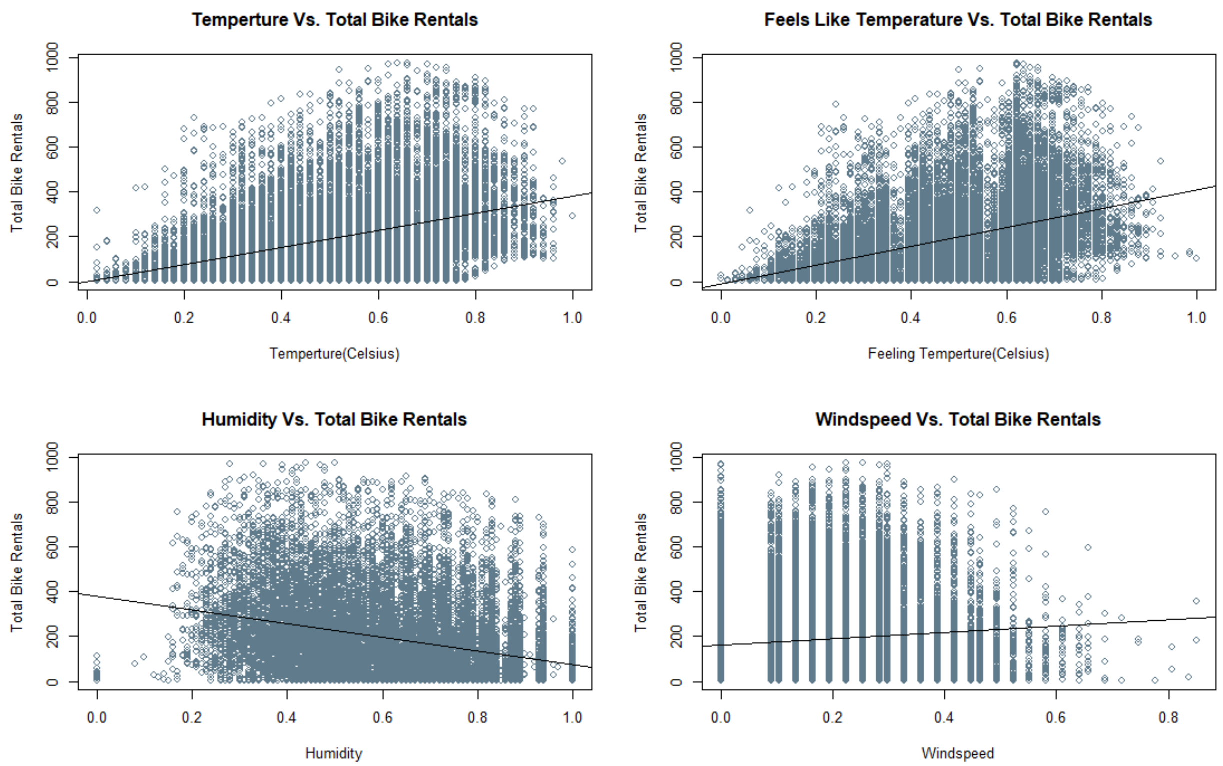
Generally, it is quite evident that Good weather conditions are the best for riding bikes. Normal weather is moderately good while there is very low demand in Bad weather and no demand at all when the weather is Very Bad.

Now we will take a closer look at other variables that also represent weather conditions and see if there are any other noticeable trends in our dataset. We plot each of Temperature, Feels like Temperature, Humidity, and Windspeed against Total Bike Rentals and plot the line of best fit to make the trend evident:

```
> par(mfrow=c(2,2))
> plot(bike$temp,bike$cnt, col="lightskyblue4",xlab="Temperture(Celsius)",yla
b="Total Bike Rentals",main = "Temperture Vs. Total Bike Rentals")
> abline(lm(bike$cnt~bike$temp))
```

```
> plot(bike$atemp,bike$cnt, col="lightskyblue4",xlab="Feeling Temperture(Cels
ius)",ylab="Total Bike Rentals",main = "Feels Like Temperature Vs. Total Bike
Rentals")
> abline(lm(bike$cnt~bike$atemp))
> plot(bike$hum,bike$cnt, col="lightskyblue4",xlab="Humidity",ylab="Total Bik
e Rentals",main = "Humidity Vs. Total Bike Rentals")
> abline(lm(bike$cnt~bike$hum))
> plot(bike$windspeed,bike$cnt,col="lightskyblue4",xlab="Windspeed",ylab="Tot
al Bike Rentals",main = "Windspeed Vs. Total Bike Rentals")
> abline(lm(bike$cnt~bike$windspeed))
```

Figure 4.6



The plots above assume a linear relationship and the results are somewhat mixed. The regression line for Temperature provides a fairly good approximate of the data, while Feels like Temperature does not seem to represent the data too accurately. Similarly, Humidity provides a fairly good representation of the data but Windspeed provides a rather inaccurate representation.

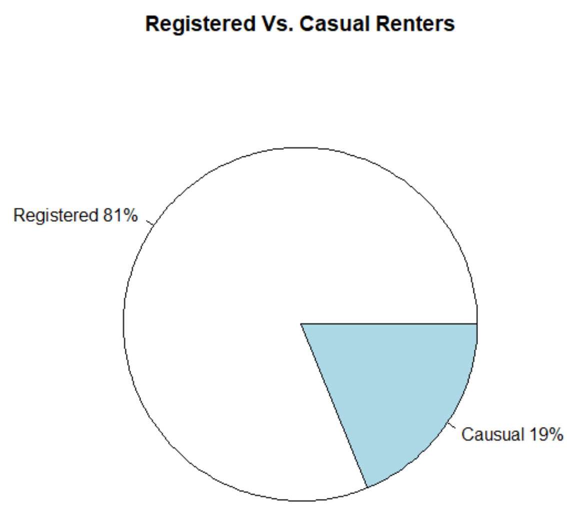
The results above indicate that there maybe a more complex relationship between these four variables and the total number of bikes rented, or no relationship at all. We will have to try different types of models to ensure we explore these relationships further.

4.4 Registered and Casual Renters

Finally, we turn our attention to the two types of users that rent bikes. As noted earlier, Registered renters are significantly higher than Casual Renters, so we simply plot a pie chart to look at the share of each type of renters:

```
> par(mfrow=c(1,1))
> slices = c(sum(bike$registered), sum(bike$casual))
> lbls = c("Registered","Causual")
> pct <- round(slices/sum(slices)*100)
> lbls <- paste(lbls, pct)
> lbls <- paste(lbls,"%",sep="")
> pie(slices, labels = lbls, main="Registered Vs. Casual Renters")
```

Figure 4.7



As expected, 81% of renters are Registered users while 19% are Casual users.

5.0 Data Preparation

Here is a list of all the Data Preparation tasks that we perform before the data set can be used by the models we choose:

1. Convert the dteday column to a Date format

   ```
   > bike$dteday=as.Date(bike$dteday)
   ```

2. Convert the Season column to a Factor and assign the appropriate labels

   ```
   > bike$season=factor(bike$season, labels = c("Spring", "Summer", "Fall"
   , "Winter"))
   > table(bike$season)

   Spring Summer   Fall Winter 
    4242   4409   4496   4232 
   ```

3. Convert the Weathersit column to a Factor and assign the appropriate labels

   ```
   > bike$weathersit=factor(bike$weather, labels = c("Good", "Normal", "Ba
   d", "Very Bad"))
   > table(bike$weathersit)

       Good   Normal      Bad Very Bad 
      11413     4544     1419        3 
   ```

6.0 Modeling and Evaluation

We use several types of models, both unsupervised and supervised, to further analyze the dataset at a more granular level and to come up with a model that provides the best prediction. The following subsections provide more details on the various types of models.

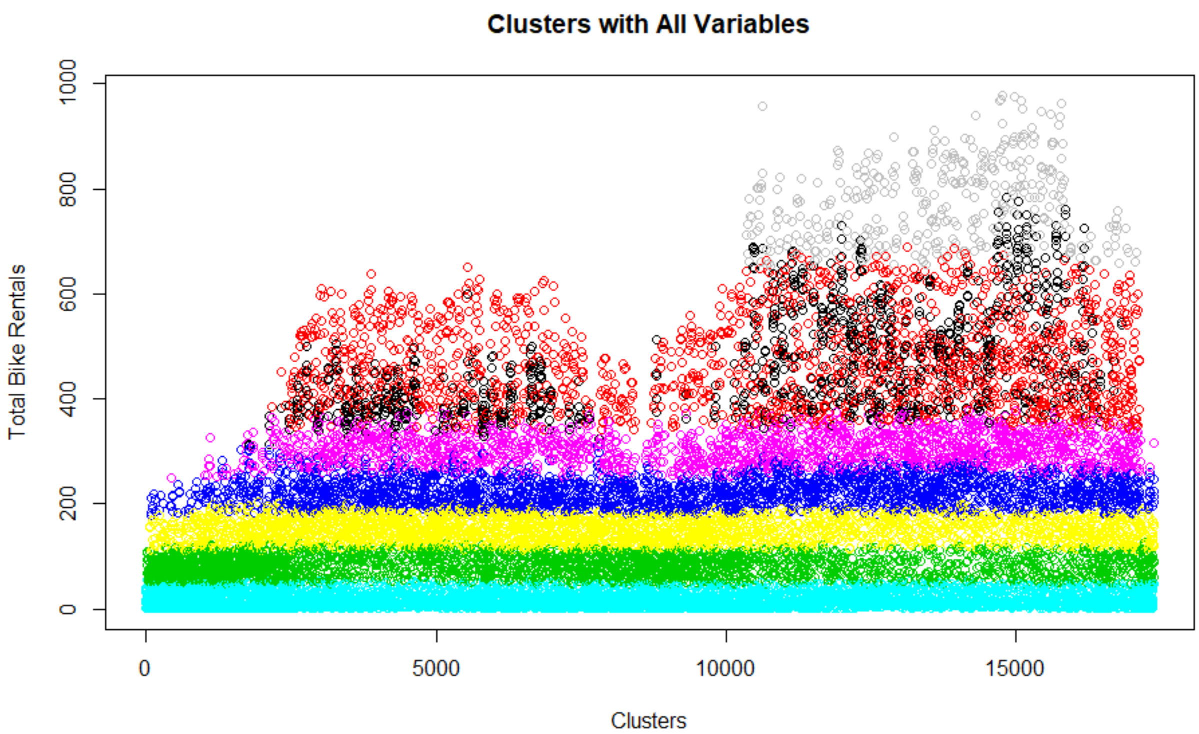
6.1 Unsupervised Learning

First, we run a generic KMeans model with all the variables to identify if there are any obvious clusters in our dataset

```
> model <- cascadeKM(data, 1, 10, iter = 10)
> model$results[2,]
 1 groups  2 groups  3 groups  4 groups  5 groups  6 groups  7 groups  8 grou
ps  9 groups 10 groups
       NA  31981.84  38524.86  42010.11  44808.37  45365.04  48067.11  49376.
97  49767.83  52255.84
> which.max(model$results[2,])
10 groups
       10
```

Kmeans with all variables gives us 10 clusters, as visually illustrated in Figure 4.8 below:

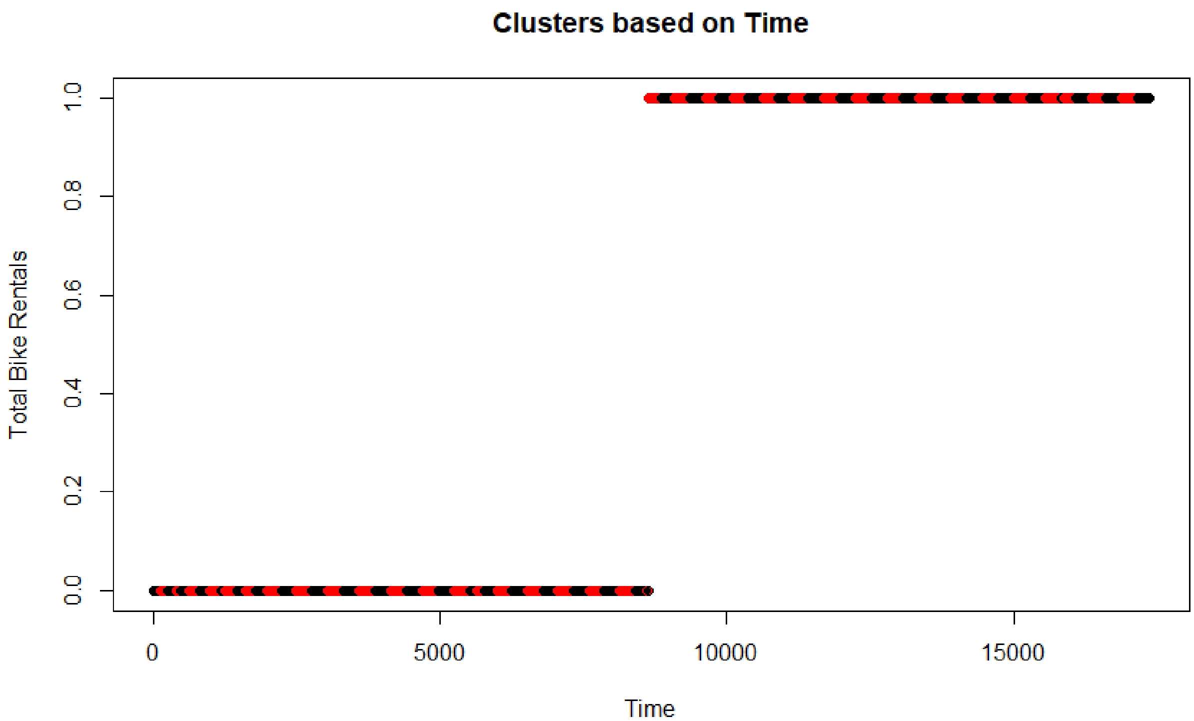
Figure 4.8



Although there seem to be some obvious clusters, we should take a closer look and try to dissect these clusters based on each model separately to see if we can identify any patterns.

To start, we look at the clusters based on time. This is shown by Figure 4.9 below:

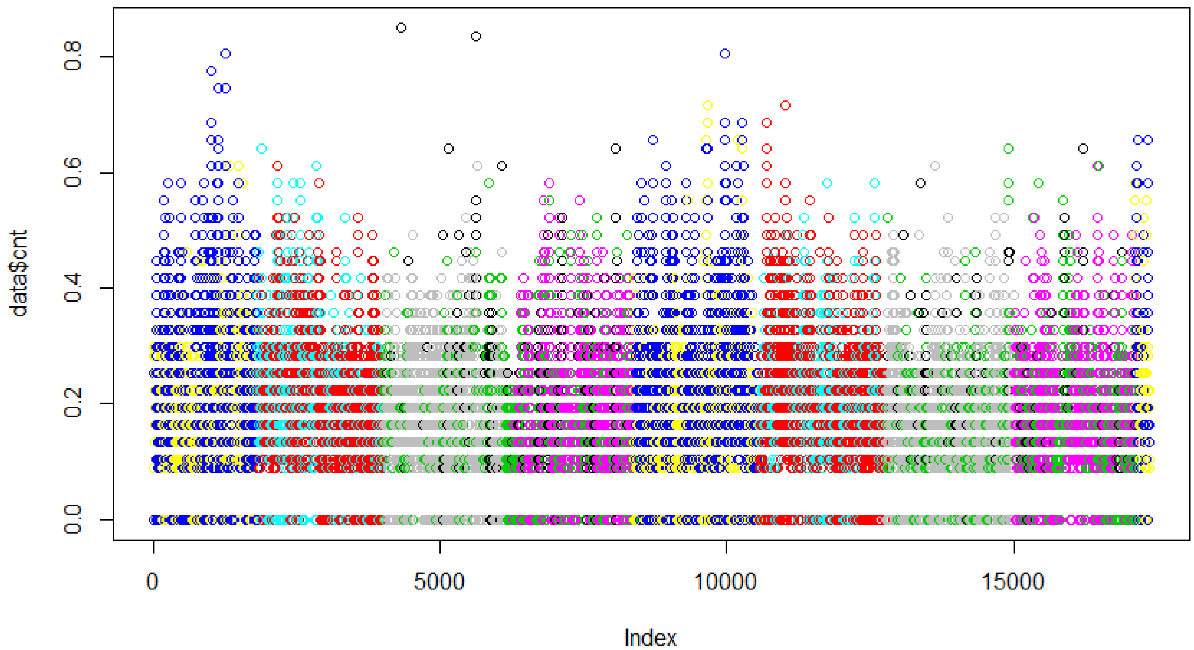
Figure 4.9



Although we do see two separate sets of clusters separated by wide vertical distance, we aren't a ble to discern any clear patterns. We confirm that by looking at all the variables that relate to ti-mespan and see that there is not much we can infer from the clusters, visually or otherwise.

Finally, we conduct clustering based on all the variables related to Weather, as illustrated by Fig-ure 4.10 below, and note that it produces 8 clusters.

Figure 4.10



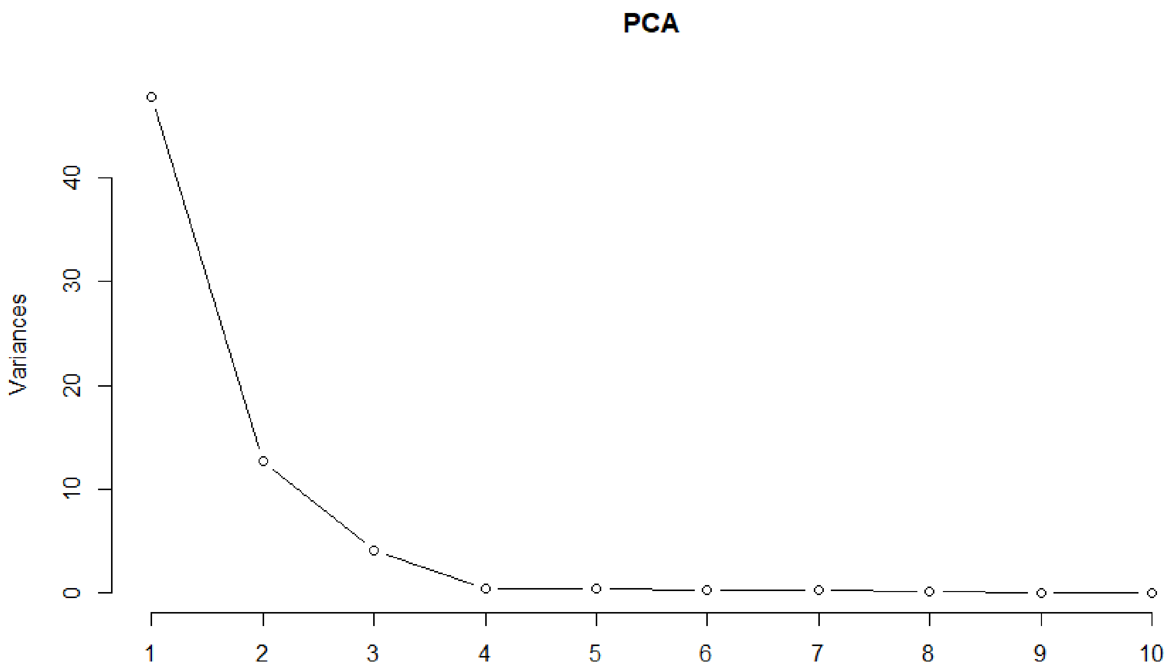
Finally, we conduct a Principal Component Analysis (PCA) to understand the interaction between the variables and the total number of bike rentals.

First, we run a correlation matrix and notice that none of the variables are highly correlated:

```
> pca <- prcomp(data)
> cor(data)
                 season           yr         mnth           hr      holiday      weekday   workingday   weathersit
season      1.000000000 -0.010742486  0.830385892 -0.006116901 -0.009584526 -0.002335350  0.013743102 -0.014523552
yr         -0.010742486  1.000000000 -0.010472929 -0.003867005  0.006691617 -0.004484851 -0.002196005 -0.019156853
mnth        0.830385892 -0.010472929  1.000000000 -0.005771909  0.018430325  0.010400061 -0.003476922  0.005399522
hr         -0.006116901 -0.003867005 -0.005771909  1.000000000  0.000479136 -0.003497739  0.002284998 -0.020202528
holiday    -0.009584526  0.006691617  0.018430325  0.000479136  1.000000000 -0.102087791 -0.252471370 -0.017036113
weekday    -0.002335350 -0.004484851  0.010400061 -0.003497739 -0.102087791  1.000000000  0.035955071  0.003310740
workingday  0.013743102 -0.002196005 -0.003476922  0.002284998 -0.252471370  0.035955071  1.000000000  0.044672224
weathersit -0.014523552 -0.019156853  0.005399522 -0.020202528 -0.017036113  0.003310740  0.044672224  1.000000000
temp        0.312025237  0.040913380  0.201691494  0.137603494 -0.027340477 -0.001794927  0.055390317 -0.102639936
atemp       0.319379811  0.039221595  0.208096131  0.133749965 -0.030972737 -0.008820945  0.054667235 -0.105563108
hum         0.150624745 -0.083546421  0.164411443 -0.276497828 -0.010588465 -0.037158268  0.015687512  0.418130329
windspeed  -0.149772751 -0.008739533 -0.135386323  0.137251568  0.003987632  0.011501545 -0.011829789  0.026225652
                   temp        atemp         hum    windspeed
season      0.312025237  0.319379811  0.15062475 -0.149772751
yr          0.040913380  0.039221595 -0.08354642 -0.008739533
mnth        0.201691494  0.208096131  0.16441144 -0.135386323
hr          0.137603494  0.133749965 -0.27649783  0.137251568
holiday    -0.027340477 -0.030972737 -0.01058846  0.003987632
weekday    -0.001794927 -0.008820945 -0.03715827  0.011501545
workingday  0.055390317  0.054667235  0.01568751 -0.011829789
weathersit -0.102639936 -0.105563108  0.41813033  0.026225652
temp        1.000000000  0.987672139 -0.06988139 -0.023125262
atemp       0.987672139  1.000000000 -0.05191770 -0.062336043
hum        -0.069881391 -0.051917696  1.00000000 -0.290104895
windspeed  -0.023125262 -0.062336043 -0.29010490  1.000000000
```

Next, we look at how do the variables explain the variation in the total number of bikes rented. This is illustrated by Figure 4.11 below. We note that the first three variables are most important in determining the total number of bikes rented. These variables are the hour, season, and month, respectively.

Figure 4.11



6.2 Supervised Learning

We apply several supervised learning model in an effort to come with the best model to make our prediction. Due to the extensive details that are part of the output of these models, the results have been included in the R Markdown. Here are the models that were applied in order:

- Random Forest
- Linear Regression
- Regression Tree
- Support Vector Machines
- Neural Network
- Random Forest with K-Fold Validation

6.3 Evaluation

After using K-Fold Cross Validation for all our models, we get the following results on the test set:

Table 6.1

Model	Accuracy
Random Forest	0.9778838
Linear Regression	0.5814736
Regression Tree	0.6393053
Support Vector Machines	0.5767638
Neural Network	NA
Random Forest with K-Fold	0.9838384

Note that we do not get reliable results from the Neural Network and the accuracy of the SVM Model is significantly lower than the other models, so we remove them further consideration.

In order to do a more thorough evaluation and see if we can further optimize our models, we explore if our clusters can add to the accuracy of our models and give us a better result. We add the clusters as variables to the dataset and run all the models again. The results from the revised set of models are listed below:

Table 6.2

Model	Accuracy
Random Forest	0.9778838
Linear Regression	0.5814736
Regression Tree	0.6393053
Random Forest with K-Fold	0.9838384
Random Forest 2.0	0.9715803
Linear Regression 2.0	0.5845314
Regression Tree 2.0	0.6393053
Random Forest with K-Fold	0.9836316

The results from adding the clusters are quite mixed. To ensure that they do not provide any useful information at all, we run the Random Forest with K-Fold validation just on the clusters alone and note that we get an accuracy of 0.5056317. Although this is quite close to accuracy of the Linear Regression model, we conclude that the clusters themselves are not important and exclude them.

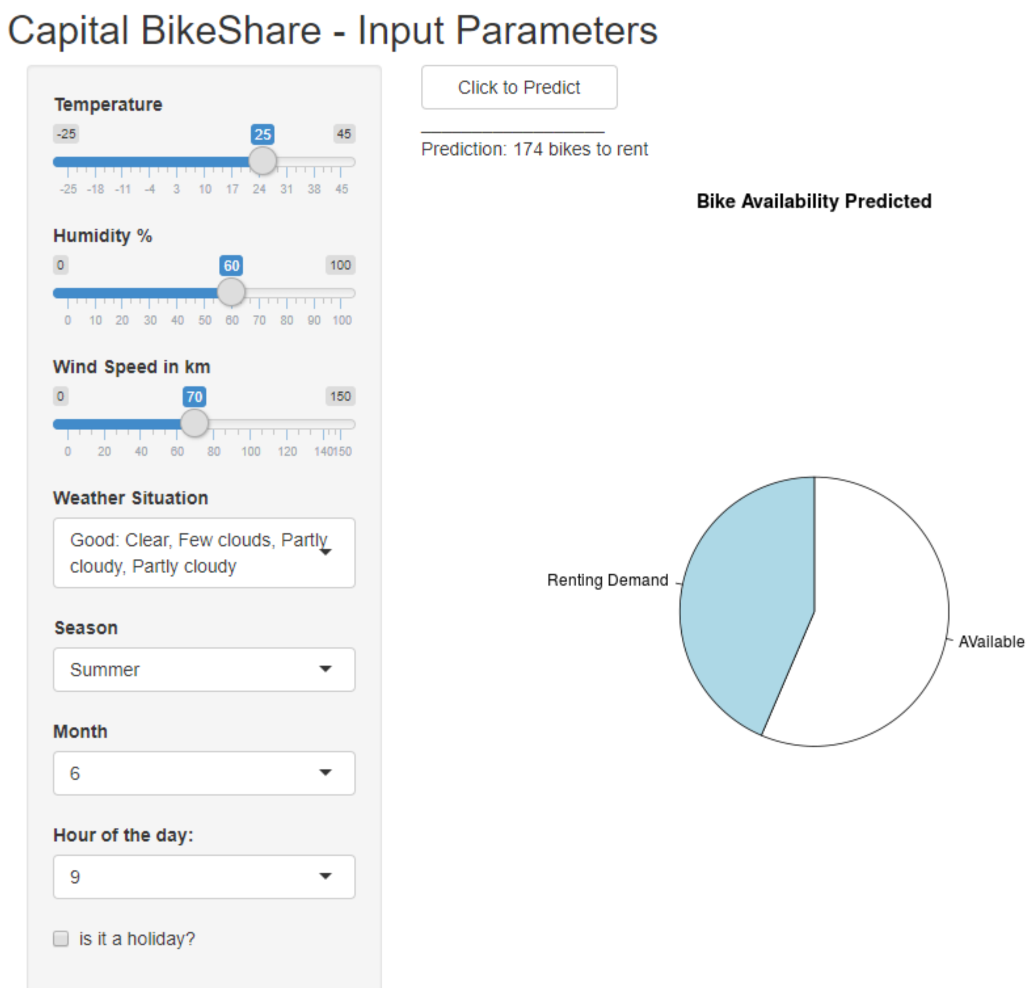
In conclusion, we choose the Random Forest with K-Fold Validation as our preferred prediction model.

7.0 Model Deployment

We have turned out prediction model into a Shiny App for Capital BikeShare so they can predict and better manage the demand for bike rentals by the hour. This Shiny App can be found at the link below and is shown in Figure 10.1

Shiny App: https://tenochinc.shinyapps.io/BikeShareAssignment3/

Table 10.1



The GitHub Repository for this Assignment can be found at:
https://github.com/markglewis/MLBikeSharing