CSC 424 Advanced Data Analysis

Final Project

BIKE SHARING DEMAND

By:

Akhil Kumar Ramasagaram
DePaul ID#: 1488438

Table of Contents.

I. OVERVIEW

Our group is analyzing the bike sharing system in Washington, DC. Core to this analysis, we will be predicting the total number of bike rentals by hour, throughout the day based on other variables. In our total data set there are a total of 17,379 rows (initially divided into 10,886 rows of training data and 6,493 rows of testing data) of hourly rental data spanning two years pulled from Kaggle. The variables in this datasets are listed below, in which *count* is the response variable.

datetime: hourly date + timestamp
season: 1 = spring, 2 = summer, 3 = fall, 4 = winter
holiday: whether the day is considered a holiday
workingday: whether the day is neither a weekend nor holiday
weather: 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 clouds, Light Rain + Scattered clouds 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
temp: temperature in Celsius
atemp: "feels like" temperature in Celsius
humidity: relative humidity
windspeed: wind speed
casual: number of non-registered user rentals initiated
registered: number of registered user rentals initiated
count: number of total rentals

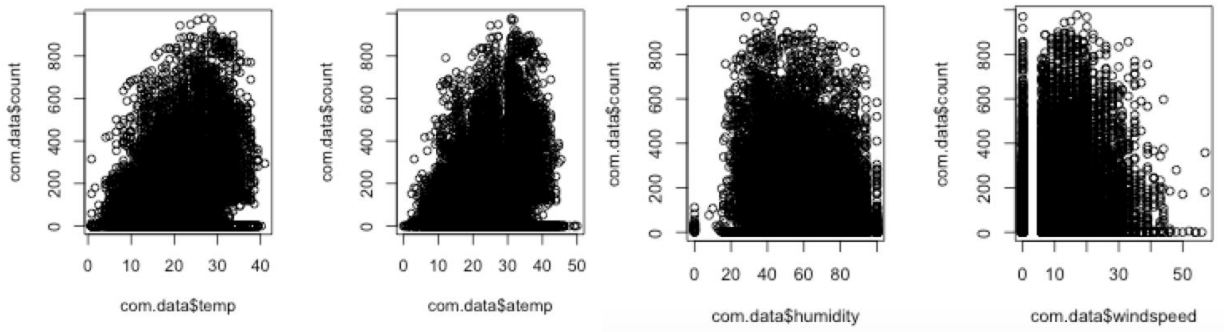
II. PREMILINARY DATA ANALYSIS:

The original Training Data has 12 variables *datetime, season, holiday, workingday, weather, temperature, air temperature, humidity, windspeed, casual, registered and count*. The response/dependent variable is *count*. Out of these 12 we can ignore *casual* and *registered*, because the sum of these makes up the *count*, and the Test Data doesn't contain these variables. In order to use the timestamp data in our model from *datetime*, we have to split the variable into several different variables like *hour, day, month, year,* etc. The variables *season, holiday, workingday,* and *weather* are of type integer, but it makes more sense to convert them into factors because they are categorical variables. To do all of this, we must first combine the training and testing data into one dataset, Com.data, then split up the timestamp into new variables *hour wday, month, and year,* and then convert *season, holiday, workingday*, and *weather* into factor variables. After making all these changes, our combined Training and Testing Datasets remains unchanged with 17,379 rows of hourly rental data across 13 variables (12 independent).

1. Does the Data Show a Non-linear Relationship? Explore the Distributions of Each Variable.

After plotting all of the independent numerical variables *(temp, atemp, humidity, windspeed)* against the response variable *count*, and then examining the graphs, it looks like only *temp* and *atemp* show a positive linear relationship with *count,* while *humidity* and *windspeed* show a non-linear relationship. This makes sense: as the temperature and "feels like" temperature rises, bike rentals go up, but as the humidity and/or wind speed goes up, bike rentals go up and down.

Graph 1-4: Independent/Dependent Scatterplot



From looking at the distributions of the data below, and after turning the *datetime* variable into multiple variables and transforming the integer variables that should be categorical into factor variables. From the *season* variable, we can see that there were 4,242 entries during the spring season, 4,409 during the summer season, 4,496 during the fall season, and 4,232 during the winter season. There were only 500 entries on holidays, but 16,879 on normal days. There were 11,865 entries on working days and 5,514 on non-working days (weekends+holidays). There were 11,413 entries on cloudy days, 4,544 on misty days, 1,419 on days with light snow, and only three on days with heavy snow. The average temperature (in degrees Celsius) is 20.38, the median is 20.5, the minimum is 0.82, and the max is 41. The average "feels like" temperature (*atemp*, in degrees Celsius) is 23.79, while the median is 24.24, the minimum is

zero, and the max is 50. The average humidity is 62.72, the median is 63, the minimum is zero, and the max is 100. The average windspeed is 12.74, the median is 13, the minimum is zero, and the max is 57.

Regarding time, the entries between each hour of the day were pretty evenly distributed, with the minimum being 697 between the hours of 3-4 AM and 4-5 AM, and the max being 730 between 4-5p.m. and 5-6p.m. There were 2,502 entries on Sunday, 2,479 on Monday, 2,453 on Tuesday, 2,475 on Wednesday, 2,471 on Thursday, 2,487 on Friday, and 2,512 on Saturday. There were 1,429 entries in January, 1,341 in February, 1,473 in March, 1,437 in April, 1,488 in May, 1,440 in June, 1,488 in July, 1,475 in August, 1,437 in September, 1,451 in October, 1,437 in November, and 1,483 in December. There were 8,645 entries in 2011 and 8,734 in 2012. The count, which is the number of total rentals during the time period, had an average of 120 with a median of 28, while the minimum was zero and max was 977.

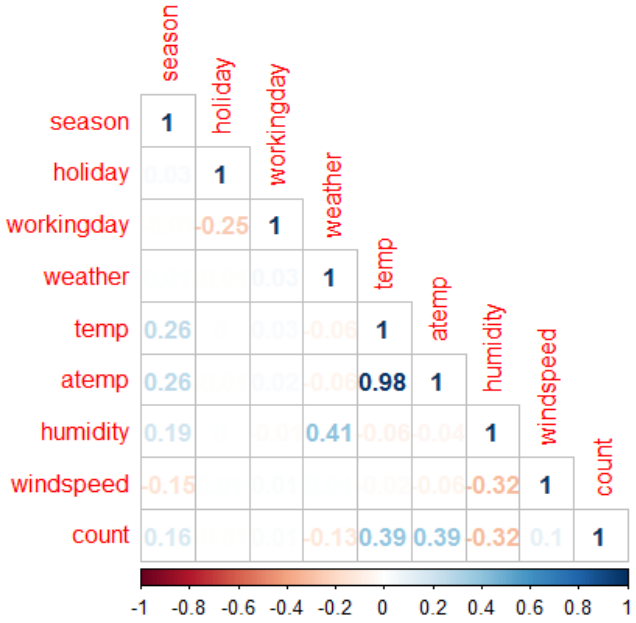
R Output 1: Summary Statistics

```
> summary(com.data$season)
   1    2    3    4
4242 4409 4496 4232
> summary(com.data$holiday)
    0     1
16879   500
> summary(com.data$workingday)
    0     1
 5514 11865
> summary(com.data$weather)
    1     2     3     4
11413  4544  1419     3
> summary(com.data$hour)
  0   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18  19  20  21  22
726 724 715 697 697 717 725 727 727 727 727 727 728 729 729 729 730 730 728 728 728 728 728
 23
728
> summary(com.data$wday)
   0    1    2    3    4    5    6
2502 2479 2453 2475 2471 2487 2512
> summary(com.data$month)
   0    1    2    3    4    5    6    7    8    9   10   11
1429 1341 1473 1437 1488 1440 1488 1475 1437 1451 1437 1483
> summary(com.data$year)
2011 2012
8645 8734
> summary(com.data$count)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      0       0      28     120     192     977
> summary(com.data$temp)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.82   13.94   20.50   20.38   27.06   41.00
> summary(com.data$atemp)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00   16.66   24.24   23.79   31.06   50.00
> summary(com.data$humidity)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00   48.00   63.00   62.72   78.00  100.00
> summary(com.data$windspeed)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   7.002  13.000  12.740  17.000  57.000
```

2. Should a Transformation of the Response Variable and/or the Predictors Be Useful? Assess the Normality Of the Variables

The data shows that only *temp* and *atemp* shows a linear relationship with *count* and the rest of the variables show a non-linear relationship. Out of the 12 variables, *casual* and *registered* are ignored because these sum to *count*, and the test set doesn't contain these variables.

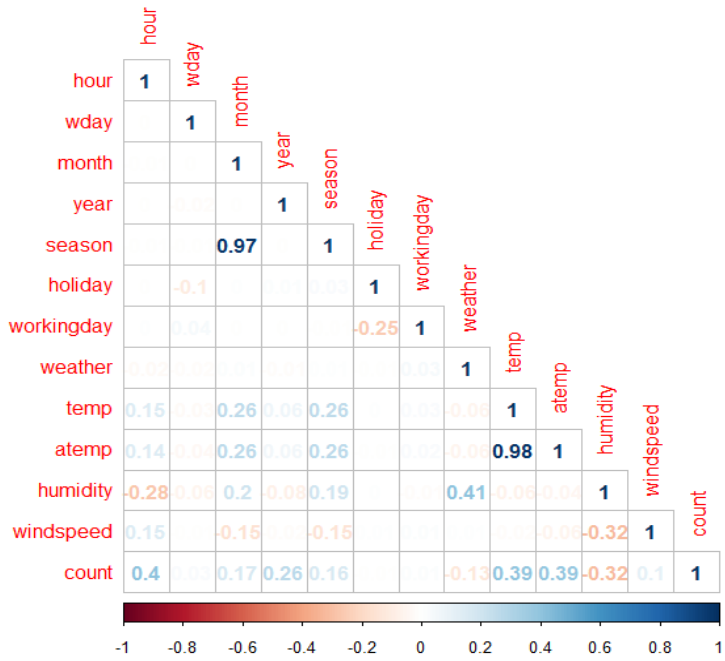
Chart 1: Correlation Table of the Initial Dataset



Upon initial analysis, the datetime variable has the timestamp in the format YYYY-MM-DD HH:MM:SS. It is of type factor with 10,886 levels. That's a lot of levels and most of the machine learning models can't handle that much levels. In order to use the timestamp data in our model we have to split it into several different variables like hour, day, weekend where the hour field would have data with 24 levels and weekend with two levels.

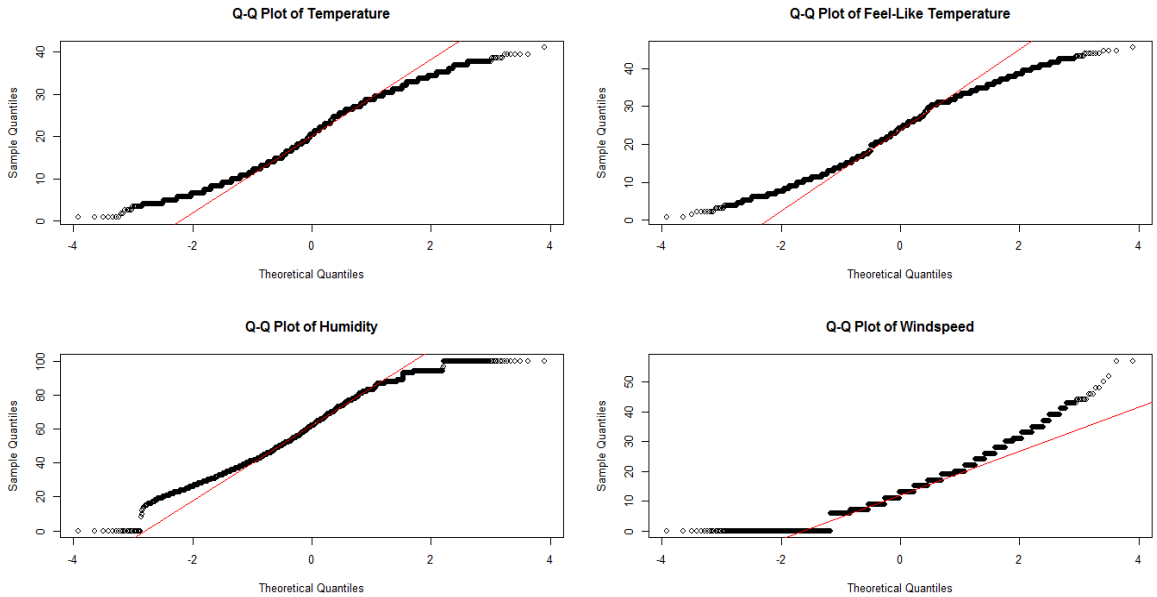
The variables *season, holiday, workingday* and *weather* are of type integer, but makes more sense to convert them into factor as they are categorical variables. The *weather* variable has four classes. Cloudy, Mist, Light Snow, and Heavy Snow. There is only one observation with Heavy Snow, treating it as an outlier we combine it with light snow.

Chart 2: Correlation Table after Transforming Data



After creating the *hour, wday, month* and *year* variables, all the variables have a positive relation with *count* except for the *wday.* Another path of analysis can be to create another variable called *weekend* where it has a binary value saying whether that day was a weekend or not. So transforming the predictor turned out to be useful.

Graphs 4-8: Normality Plot



There are points scattered away from the line, this is one way to tell if the data is normally distributed or not. Here the data is not normally distributed. Another way to find the normality of the data is by using a t-test. After performing t-test on each variable we found that the probability is very low almost near to zero. As our null hypothesis is that our data is normal and Alternative hypothesis is that our data is not normal. If the P-value is greater than 5 percent then we accept the null hypotheses. But as our p-value is zero the null hypothesis is rejected and alternative is accepted.

3. Explore the Correlation Scatterplots And Covariance for the Associations Between the Independent Variables.

Continuing the correlation assessment from the previous section, the correlation matrix between all the independent variables (Appendix A, Chart 3) demonstrates clear patterns amongst the variables. Specifically, due to the categorical nature of of same variables, datetime correlations are cyclical, and season and weather demonstrate seasonality. However, most important for this analysis is the numerical variables *temp, atemp, humidity* and *windspeed*. From this evaluation a clear linear relationship exists between *temp* and *atemp*.

In reviewing the covariance we show some predicted and unpredicted results. First, as expected the covariance between *temp* and *atemp* is very high, even higher than *temp* measured against itself. With regard to the unpredicted results, *humidity* is negatively associated with *temp* and *atemp*. The negative covariance between *windspeed* and the other variables is understandable as a " breeze" in either the summer or winter appears to reduce the temperature and more so the "feels like" temperature. As for unexepected results, the covariance between *temp* and *humidity*, particularly for the Washington, DC climate, was presumed to be positively correlated. For future analysis, I think a seasonal understanding of these variables should be undertaken to more fully understand this relationship.

As highligted in the correlation and covariance analysis above, as well as in the next section's evaluation of the VIF, dropping the *temp* or *atemp* variables may present the best option for continued analysis.

Chart 4: Covariance Matrix

	temp	**atemp**	**humidity**	**windspeed**
temp	62.327882			
atemp	66.999888	73.831242		
humidity	-10.643933	-8.606665	372.219209	
windspeed	-1.496484	-4.390393	-45.877373	67.187453

4. Collinearity among the Independent Variables

First, the variables *temp, atemp, humidity and windspeed* in order to place these different units into the same scale:

Chart 5: Normalization Matrix

season	holiday	workingday	weather	temp	atemp	humidity	windspeed	casual	registered	count
1	0	0	1	0.24	0.2879	0.81	0.0000	3	13	16
1	0	0	1	0.22	0.2727	0.80	0.0000	8	32	40
1	0	0	1	0.22	0.2727	0.80	0.0000	5	27	32
1	0	0	1	0.24	0.2879	0.75	0.0000	3	10	13
1	0	0	1	0.24	0.2879	0.75	0.0000	0	1	1
1	0	0	2	0.24	0.2576	0.75	0.0896	0	1	1
1	0	0	1	0.22	0.2727	0.80	0.0000	2	0	2
1	0	0	1	0.20	0.2576	0.86	0.0000	1	2	3
1	0	0	1	0.24	0.2879	0.75	0.0000	1	7	8
1	0	0	1	0.32	0.3485	0.76	0.0000	8	6	14
1	0	0	1	0.38	0.3939	0.76	0.2537	12	24	36

After normalization, a standard linear model is built in order to evaluate multicollinearity.

fit=lm(hour$count~season+holiday+workingday+weather+temp+atemp+humidity+windspeed+casual+registered,data=hour)

Then we examined if there is multicollinearity problem by calculating VIF statistics:

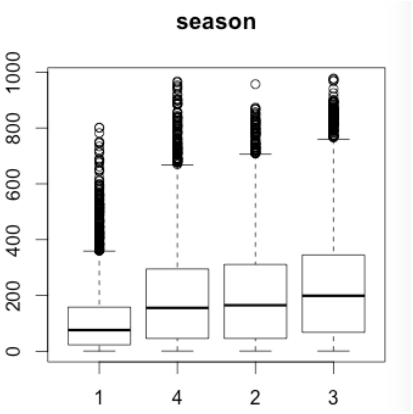
R Output 2: VIF Outputs

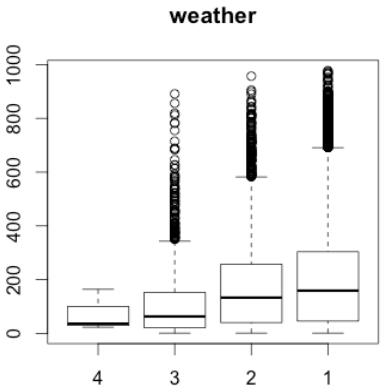
```
> vif(fit)
    season    holiday workingday    weather       temp      atemp   humidity  windspeed     casual registered
  1.184877   1.072383   1.402452   1.267038  43.724847  43.936653   1.611666   1.194671   2.162105   1.583293
```

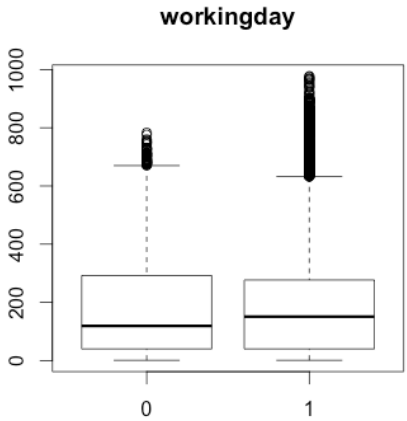
The VIF demonstrates that values for both *temp* and *atemp* are larger than ten. It suggests a multicollinearity problem of these two variables, which fits with the variable description because *temp* is the temperature in Celsius and *atemp* is the "feels like" temperature in Celsius.

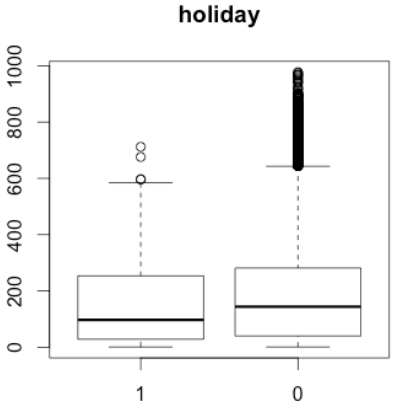
5. Box-Plots Exploring Qualitative Variables

Graphs 13-16: Qualitative Box-Plot





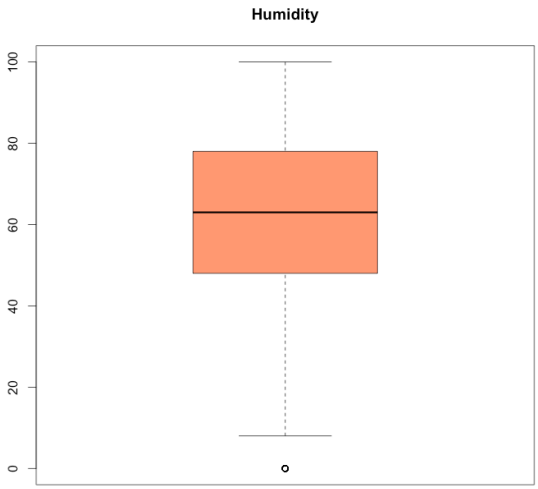




6. Explore the outliers in the dataset.

For an initial outlier analysis, we looked at each individual variable in the combined Test/Train dataset using boxplots and Tukey's hinges. It is also important to evaluate outliers in the context of the models created for our projects to see how they affect the models and whether they should be removed, especially if the outliers are not extreme.

The variables *temp* and *atemp* do not have any outliers when looking at boxplots. The humidity variable does have one outlier on the low end, as shown below.

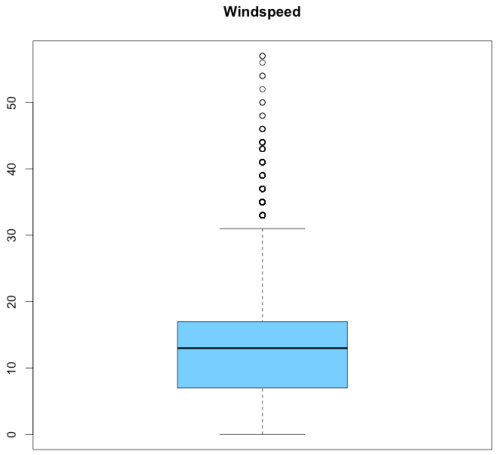


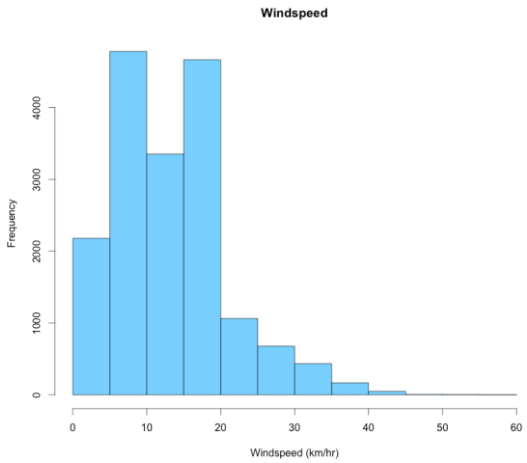
What looks like a single point below the lowest hinge, around a humidity level of zero, is actually 22 points covering an entire day of data (3/10/11). A humidity index of zero for the entire day in DC seems unlikely, especially since the preceding day had a humidity index of 93 and the following day had an index of 100. This indicates some sort

of instrument malfunction, and I would recommend removing this day from the data or replacing these values with the average humidity value from the rest of the data, since it seems to be equipment error.

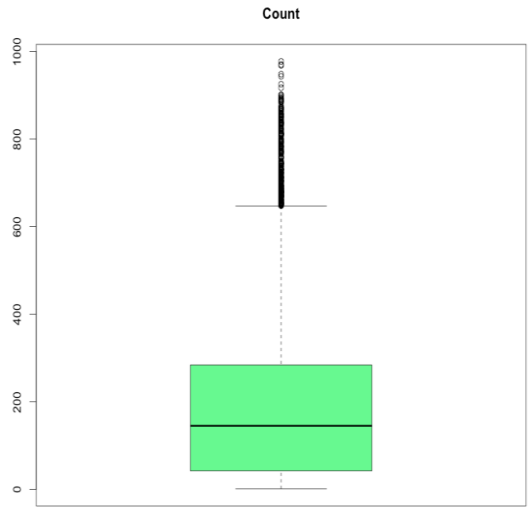
As mentioned in Section 2, a value of "4" for the *weather* variable indicating heavy precipitation is somewhat rare. There is one observation in the training data set (1/9/12 18:00) and two in the test data set (1/26/11 16:00 and 1/21/12 1:00) with this value. Since there are only three data points with this value in a combined data set of 17,379 rows, combining these values with class "3" for light precipitation seems like a reasonable way to deal with the outliers.

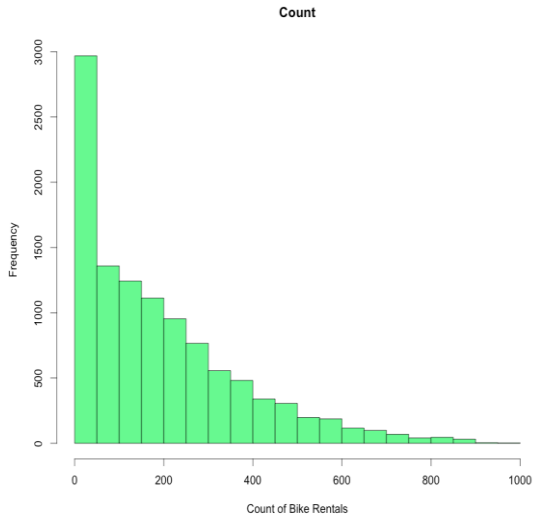
The variable *windspeed* appears to have many outliers. The median is 13 and the third quartile 17, while the max is 57, for context. The top hinge on the boxplot falls around 31. Based on a chart comparing windspeeds in different units, it seems these are km/hr. At 31 km/hr, surface waves form on water and small trees sway. At 57, large trees are swaying. These windspeeds are not unreasonable over the time period in this dataset. The distribution of the windspeed data is skewed and non-normal, but attempting log/square root transformations of the data does not reduce the number of outliers or have much effect on the normality of this variable. We will discuss how to handle this variable as a group.





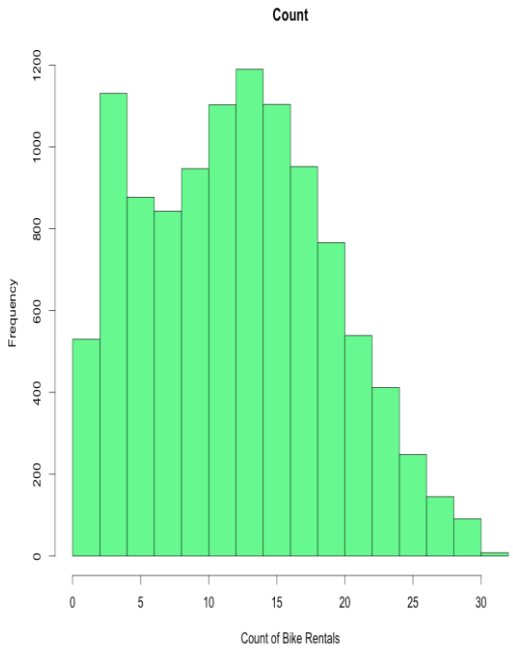
Lastly and perhaps most importantly, the count variable is non-normal and has a skewed distribution. There are outliers present, but this is not unexpected on the high end of the distribution considering most hours will have less rentals.

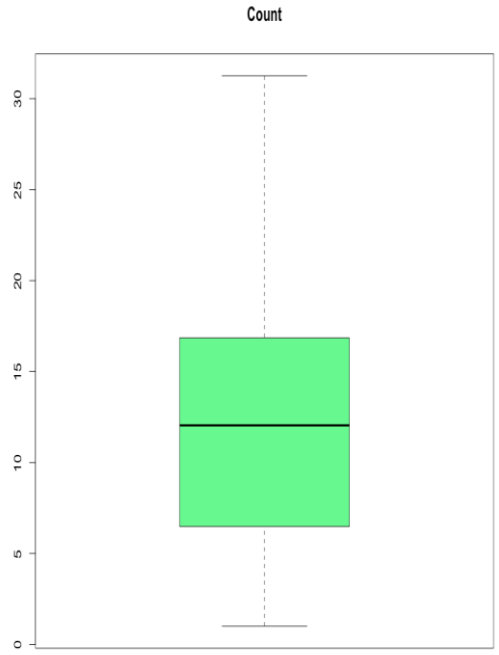




A simple square root transformation is beneficial for this variable, removing the outliers.

```
> c <- sqrt(com$count)
> hist(c)
> boxplot(c)
```





III. MODEL

We are using the bike sharing dataset from Kaggle, where the outcome variable count is of type integer. We also have casual and registered variable and the sum of these two variables is count. We are going to implement the Canonical Correlation Analysis (CCA) to understand the relationship between our dependent and independent variables.

The reason why choose CCA is, it measures the relationship between two set of variables. Now in our case we have two dependent variables count & registered. If you are familiar with Multiple Regression (MR), CCA in general does multiple regression on both sides. Here we are interested in the correlations between the linear combinations created for both set of variables.

The first set called **U** contains all the predictors like “season”, ”yr”, ”month”, ”hour”, “ daypart” , ”weekday”, ”workingday”, ”holiday”, ”weather”, ”temp”, ”humidity”, ”windspeed”. The second set called **V** contains our two new dependent variables “casual”, “registered”.

We can look at the correlations within and between the two sets of variables using the **matcor()** function from **CCA** package in **R**. We can check the within class correlation using the $Xcor or $Ycor, between class correlation can be found using $XYcor.

CODE:

```
matcor(U,V)
```

The canonincal correlation of our variables can found using the **cc()**.

CODE:

```
> Con_1 <- cc(U,V)
> Con_1$cor
[1] 0.6931028 0.5079348
```

We only got two values because, the number of canonical variate pairs we can have is equal to the number of variables in the smaller set. Below are the raw canonical coefficients

```
$xcoef
                  [,1]         [,2]
casual     -0.016198535  0.017058633
registered -0.002117715 -0.007364651

$ycoef
                  [,1]         [,2]
season     -0.095637499 -0.207949926
yr         -0.457046612 -0.683114331
mnth        0.001267027 -0.001009635
hr         -0.054464092 -0.062317604
holiday     0.314715634 -0.239315014
weekday    -0.021763643  0.012508335
workingday  0.705358574 -1.752986942
weathersit -0.010672105  0.168594087
temp       -3.051464447  1.092233725
hum         1.970356026 -0.547657592
windspeed  -0.022725918 -0.349967202
daypart     0.183253828  0.262619817
```

Interpretation of the above table is like, a one unit increase in seasons leads to .095 decrease in the first canonical variate of set 2 when all of the other variables are held constant. Next, we'll

use **comput()** to compute the loadings of the variables on the canonical dimensions. These loadings are correlations between variables and the canonical variates.

```
                [,1]       [,2]
casual     -0.9610562  0.2763530
registered -0.7251516 -0.6885893

$corr.Y.xscores
                [,1]         [,2]
season     -0.15185001 -0.093105373
yr         -0.19534665 -0.162692598
mnth       -0.09386709 -0.078719060
hr         -0.36048447 -0.163718544
holiday    -0.01003320  0.079323061
weekday    -0.03304998  0.003468479
workingday  0.19729785 -0.402847721
weathersit  0.16067247  0.006467763
temp       -0.47457462  0.012747326
hum         0.36496488  0.013474318
windspeed  -0.09849570 -0.015824435
daypart     0.07137279  0.108863569

$corr.X.yscores
                [,1]       [,2]
casual     -0.6661108  0.1403693
registered -0.5026046 -0.3497585

$corr.Y.yscores
                [,1]         [,2]
season     -0.21908728 -0.183301804
yr         -0.28184368 -0.320302103
mnth       -0.13543025 -0.154978658
hr         -0.52010245 -0.322321940
holiday    -0.01447577  0.156167789
weekday    -0.04768410  0.006828591
workingday  0.28465885 -0.793109052
weathersit  0.23181621  0.012733450
temp       -0.68471028  0.025096381
hum         0.52656672  0.026527651
windspeed  -0.14210836 -0.031154459
daypart     0.10297577  0.214325854
```


Canonical loadings for set 1

Cross loadings for set 1

Canonical loadings for set 2

Cross loadings for set 2

This table presents the canonical loading and cross loading of both the sets. The top 3 most significant attributes in set 2 for first variate are temperature, humidity and hour. In the second variate attributes workingday, hour and year are most significant.

The above correlations are between observed variables and canonical variables which are known as the canonical loadings. These canonical variates are actually a type of latent variable. Next we will perform significance test to find whether the canonical covariates are really significant to consider for the model.

```
> N = dim(U)[1]
> p = dim(U)[2]
> q = dim(V)[2]
> p.asym(rho = Con_1$cor, N, p, q, tstat = "Wilks")
Wilks' Lambda, using F-approximation (Rao's F):
              stat    approx df1   df2 p.value
1 to 2:  0.8222557 148.75939  24 34730       0
2 to 2:  0.9932079  10.79626  11 17366       0
```

In the above table, the first test of the canonical dimension test whether both the dimensions are significant and the next dimension tests for whether the dimension 2 itself is significant to the model. The tables indicates that both the canonicals dimensions are statistically significant at 0.05 significance level.

```
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Standardized canonical coefficients for DEPENDENT variables
          Function No.

Variable                  1               2

casual                .79860          -.84115
register              .32063          1.11467

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Standardized canonical coefficients for COVARIATES
          CAN. VAR.

COVARIATE                 1               2

dteday               -.24832          -.12745
season                .10546           .22995
yr                    .44482           .45254
mnth                  .11984           .06722
hr                    .37669           .43089
holiday              -.05267           .03998
weekday               .04367          -.02508
workingd             -.32816           .81595
weathers              .00661          -.10790
temp                  .58755          -.21038
hum                  -.37969           .10592
windspee              .00301           .04293
daypart              -.20440          -.29290
```

This table presents the standardised canonical coefficients across both sets of variables. In set 1 the first canonical dimension is mostly influenced by casual (.79) and the second canonical dimension is influenced by registered. In set 2 the first canonical dimension is influenced by temperature (0.58) and the second dimension is influenced by the working day (0.81)

Eigenvalues and Canonical Correlations

Root No.	Eigenvalue	Pct.	Cum. Pct.	Canon Cor.	Sq. Cor
1	.92472	72.67303	72.67303	.69314	.48044
2	.34772	27.32697	100.00000	.50794	.25801

The above table shows the eigenvalues for both the canonical dimensions and if we take a look at the percentage of variance covered by the particular canonical dimension. It looks like the first dimension captured the most variability which is 72.67 % and the second dimension captures only 27.32 %. So we will use only the first dimension for further use.

IV. EXECUTIVE SUMMARY

- There are two meaningful explanatory variables here, temperature and workingday. These variables made more sense when we try to model two different models for causal & registered instead of modelling with Count. For this purpose we used Canonical Correlation method to find which sets of variable are highly correlated to two target variables.

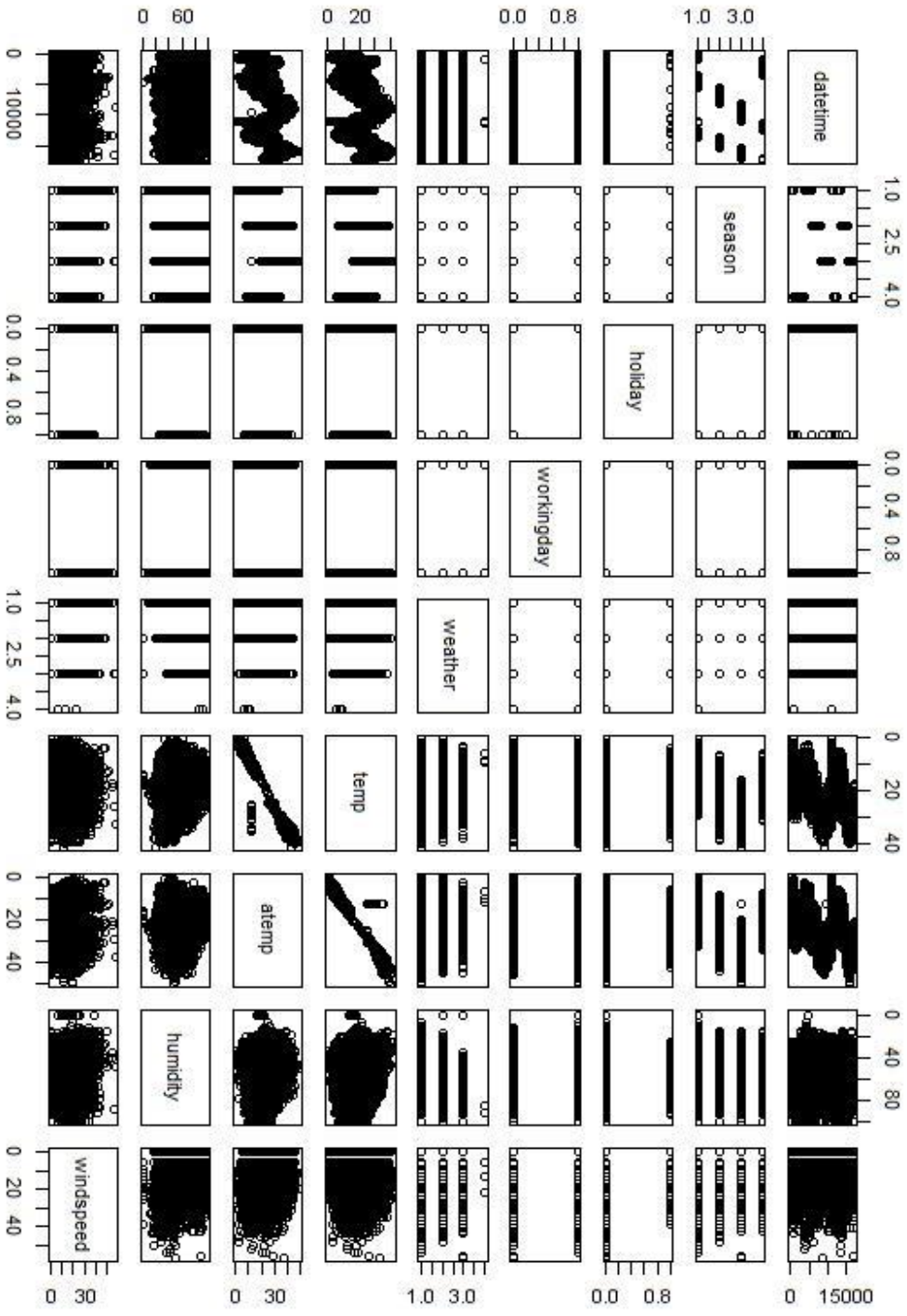
- The major finding of our analysis is that **casual** rentals are more influenced by the temperature, and **registered** rentals are more influenced by the workingday. May be people who work must have registered for the bike sharing program as a means of transit between their home and office.

- For this analysis I have not transformed the target variables, they are skewed towards right. Even after log transformation the data is not normal, so we have find another way for transformation.

- There are other variables like season, weather & holiday which have less correlation to the target variables but intuitively they look like important variables at first. So, may be if we find a method to combine these features and build a new feature will help improve the model.

APPENDIX A: ADDITIONAL CHARTS AND GRAPHS

Chart 3: VARIABLE CORRELATION MATRIX



APPENDIX B: R CODE

i. OVERVIEW

Creating and Cleaning the Data Set

```
train <- read.csv("train.csv", header = T)
test <- read.csv("test.csv", header = T)
train$casual <- NULL
train$registered <- NULL
test$count <- 0
train$type <- TRUE
test$type <- FALSE
#combining both datasets for transformation
com.data <- rbind(train,test)
com.data$datetime <- strptime(com.data$datetime, format="%Y-%m-%d %H:%M:%S")
com.data$hour <- as.factor(com.data$datetime$hour)
com.data$wday <- as.factor(com.data$datetime$wday)
com.data$month <- as.factor(com.data$datetime$mon)
com.data$year <- as.factor(com.data$datetime$year + 1900)
# converting integer variables into factor
com.data$season <- as.factor(com.data$season)
com.data$holiday <- as.factor(com.data$holiday)
com.data$workingday <- as.factor(com.data$workingday)
com.data$weather <- as.factor(com.data$weather)
com.data$datetime <- NULL
# spliting the train and test from the transformed data
com.data$datetime <- strptime(com.data$datetime, format="%Y-%m-%d %H:%M:%S")
com.data$hour <- as.factor(com.data$datetime$hour)
com.data$wday <- as.factor(com.data$datetime$wday)
com.data$month <- as.factor(com.data$datetime$mon)
com.data$year <- as.factor(com.data$datetime$year + 1900)
# converting integer variables into factor
com.data$season <- as.factor(com.data$season)
com.data$holiday <- as.factor(com.data$holiday)
com.data$workingday <- as.factor(com.data$workingday)
com.data$weather <- as.factor(com.data$weather)
com.data$datetime <- NULL
train <- subset(com.data, type == TRUE)
test <- subset(com.data, type == FALSE)

# checking for normality
par(mfrow = c(2,2))
```

```
qqnorm(train$temp,main = "Q-Q Plot of Temperature")
qqline(train$temp, col = 2)
qqnorm(train$atemp,main = "Q-Q Plot of Feel-Like Temperature")
qqline(train$atemp, col = 2)
qqnorm(train$humidity,main = "Q-Q Plot of Humidity")
qqline(train$humidity, col = 2)
qqnorm(train$windspeed,main = "Q-Q Plot of Windspeed")
qqline(train$windspeed, col = 2)
```

II. Preliminary Analysis, Section 5:

Weather:

```
> boxplot(hour$count~hour$weather, main="weather")
> byMedian = with(hour, reorder(weather, count, median))
> boxplot(count ~ byMedian, data=hour,main="weather")
```

Season:

```
> boxplot(hour$count~hour$season, main="season")
> byMedian = with(hour, reorder(season, count, median))
> boxplot(count ~ byMedian, data=hour,main="season")
```

Holiday:

```
> boxplot(hour$count~hour$holiday, main="holiday")
> byMedian = with(hour, reorder(holiday, count, median))
> boxplot(count ~ byMedian, data=hour,main="holiday")
```

Workingday:

```
> boxplot(hour$count~hour$workingday, main="workingday")
> byMedian = with(hour, reorder(workingday, count, median))
> boxplot(count ~ byMedian, data=hour,main="workingday")
```