# Predicting Bike Rental

Navdeep Singh Sidana

3rd, Augest, 2019

Contents

1. Introduction
   1. Problem Statement
   2. Data
2. Methodology
   1. Exploratory Data Analysis
   2. Feature Engineering
      1. Feature Selection
   3. Prediction
      1. Multiple Linear Regression
      2. Support Vector Regression
      3. Random Forest Regression
3. Conclusion

Appendix A – Extra Figures

Appendix B – R Code

Appendix C- Daily Bike Rental Prediction and Code Files

1. Introduction

Bikes are the most preferred means of going around the city, Bikers can rent a Bike from one location and can return it to a different location where they are travelling on as-needed by another traveller who is looking to rent a Bike from that particular location. Travellers take these bikes on a ride during their Weekends for a Long Drive and Enjoy the Weather and the Experience. We know that many factors can play a role in the choice of a traveller for choosing to travel via Bike, instead of Car, Public Transport, etc. It can depend upon the Season, weather conditions and day of the week, these factors can play a role in the Demand of Bike rental on any particular Day.

* 1. Problem Statement: Forecasting the Bike Rental Demand can help the Vendors in allocating the estimated vehicles that might be rented on that particular Day. This will not only help the Bike Rental Companies, but will also benefit the Biker looking to rent a bike.
  2. Data – Our task is to build multiple linear Regression models which can predict the Bike rental count on daily basis depending upon the seasonal and the environmental setting.

Table 1.1 Bike Rental Sample Data(Columns 1-16)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |
| --- |
| instant dteday season yr mnth holiday weekday workingday weathersit temp atemp hum windspeed casual registered cnt  ----------------------------------------------------------------------------------------------------------------------------------------------------  1 1 2011-01-01 1 0 1 0 6 0 2 0.344167 0.363625 0.805833 0.1604460 331 654 985  2 2 2011-01-02 1 0 1 0 0 0 2 0.363478 0.353739 0.696087 0.2485390 131 670 801  3 3 2011-01-03 1 0 1 0 1 1 1 0.196364 0.189405 0.437273 0.2483090 120 1229 1349  4 4 2011-01-04 1 0 1 0 2 1 1 0.200000 0.212122 0.590435 0.1602960 108 1454 1562  5 5 2011-01-05 1 0 1 0 3 1 1 0.226957 0.229270 0.436957 0.1869000 82 1518 1600  6 6 2011-01-06 1 0 1 0 4 1 1 0.204348 0.233209 0.518261 0.0895652 88 1518 1606 |
|  |
| |  | | --- | |  | |

Based upon all the above factors we have to accurately predict the count of Bikes rented on Daily Basis.

# Chapter 2

1. **Methodology** 
   1. **Pre-Processing –** Any Predictive modelling requires that we look at the data before we start modelling. However, in data mining terms looking at the data refers to so much that just looking. Looking at data refers to exploring the data, cleaning the data as well as visualizing the data through graphical representations and plots. This is often called as **Exploratory Data Analysis**. To start this process we will first try and look at all the probability distribution of the variables. Most analysis like regression, require the data to be normally distributed. We can visualize that in a glance by looking at the probability distributions or probability density functions of the variable.

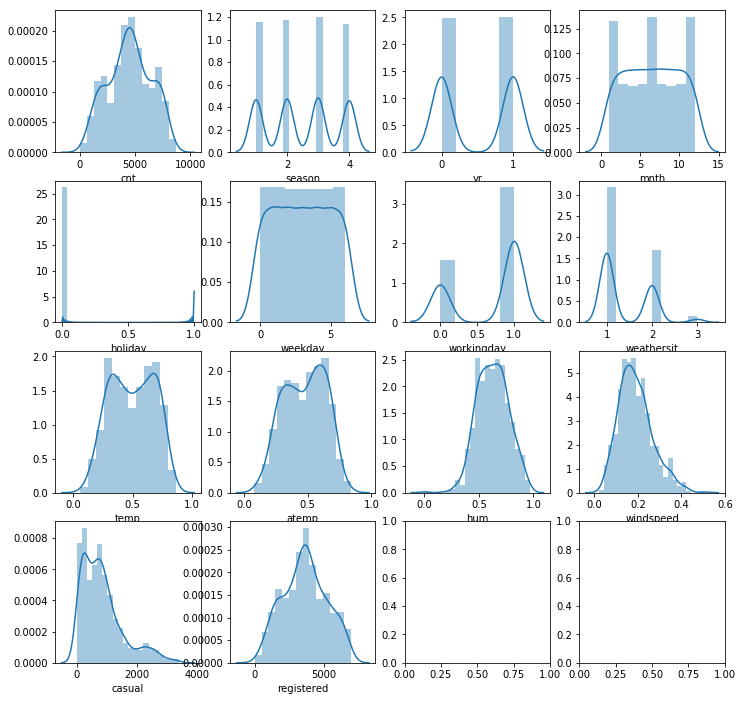
In Figure 2.1 we have plotted the probability density function of all the Seasonal & Environmental Factors we have in our data as the dependent Bike Rental Count Variable. The blue lines indicate Kernel Density Estimations (KDE) of the variable. While the pointed Blue Line represents the Normal Distribution.

* 1. **Outlier Analysis-** We can clearly observe from the probability distributions that variables like Windspeed, Humidity, Holiday and casual are skewed. The skew in these distributions can be most likely explained by presence of Outliers and extreme values in the data. For Example, you see the skewness in the Distribution of Holidays and it’s for obvious reasons.

One of the other steps of pre-processing apart from checking for normality is the presence of outliers. In this case we use a classic approach of removing outliers, Tukey’s method. We visualize the outliers using boxplots.

In figure 2.3 we have plotted the boxplots of the 11 predictor variables with respect to each quality value ranging from 3 to 8. A lot of useful inferences can be made from these plots. Turns Out there are no Outliers in the data set, even the extreme values lies within the range.

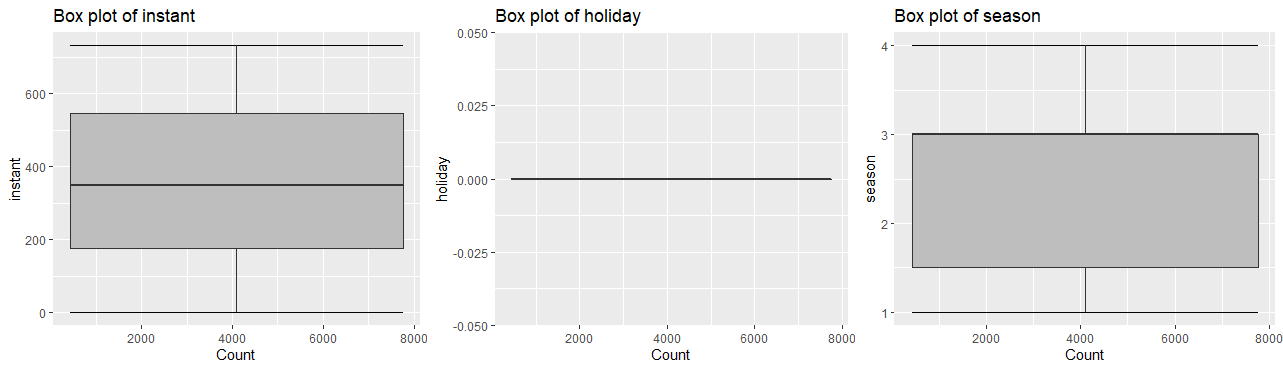
Figure 1.1 Distribution Plot of all Predictors

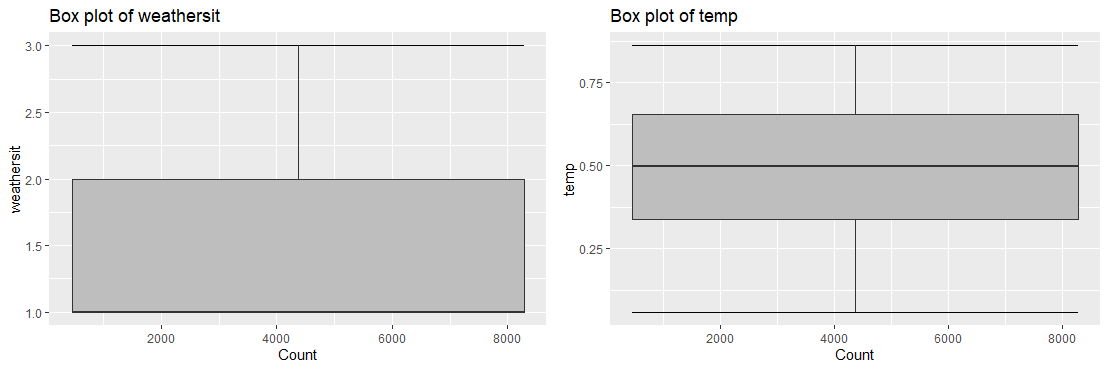
****

Other Useful inferences can also be drawn from these plots. For Example, if you compare the quality boxplots for each of the predictor variables. We can see that see 1st Month of the year is the lowest count 3rd Month which is march and then it gain starts to come down in the month of April. The Count of casual seems to vary in differentiabily over the period. We can further more interpret that the count on a working day remains consistent while varies over the weekends, Some very High while others pretty low. We can see all kind of diversity in Counts for Months.

**Boxplot of all predictors**

Figure 1.2 Box Plot of Predictors



1. Feature Selection

Before Performing any kind of modelling we need to assess the importance of each predictor in our analysis. There is a Possibility that many variables in our analysis are not important at all to the problem of class prediction There is a possibility that many variables in our analysis a are not important at all to the problem of class prediction. There are several methods of doing that. Below we have used Support Vector Regression, Mutiple Linear Regression and Random Forest Regression.

3.1 Bike Rentals

summary(regressor)

Call:

svm(formula = cnt ~ ., data = train1, type = "eps-regression", kernel = "radial")

Parameters:

SVM-Type: eps-regression

SVM-Kernel: radial

cost: 1

gamma: 0.06666667

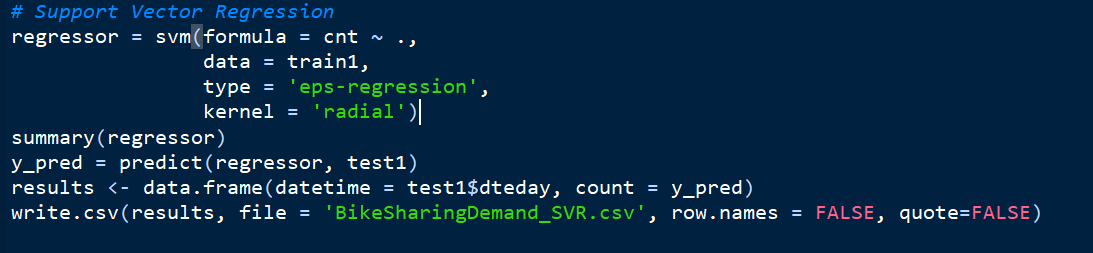
epsilon: 0.1

Number of Support Vectors: 488

We can clearly observe there is a linear relationship between number of people who registered and the count. Further more, There’s an inverse relationship with Weather Situation, which is understood.

1. Modelling
   1. Model Selection- In our early stages of analysis during pre-processing we have come to understand that Bike Rental Varies all the Months of the year depending upon the seasons, further more temperature, Humidity, Weather Condition seems to having an effect upon the Bike Rental Count
   2. The dependent variable we are dealing with is Ordinal, for regression can be done, because even though the Count variable has a numerical values, these value have an order associated with them. You always start your model building from the most simplest to more complex. Therefore, we use Multiple Linear Regression.
   3. Multiple Linear Regression
2. > # Now fitting Multiple Linear Regression to the Training Set.
3. > input <- train1[,c("weathersit", "temp", "windspeed", "weekday", "hum", "mnth", "season", "casual", "registered", "cnt")]
4. > print(head(input))
5. weathersit temp windspeed weekday hum mnth season casual registered cnt
6. 1 2 0.344167 0.1604460 6 0.805833 1 1 331 654 985
7. 2 2 0.363478 0.2485390 0 0.696087 1 1 131 670 801
8. 3 1 0.196364 0.2483090 1 0.437273 1 1 120 1229 1349
9. 4 1 0.200000 0.1602960 2 0.590435 1 1 108 1454 1562
10. 5 1 0.226957 0.1869000 3 0.436957 1 1 82 1518 1600
11. 6 1 0.204348 0.0895652 4 0.518261 1 1 88 1518 1606
12. > # Creating a relationship Model and Getting the coefficiants
13. > model <- lm(cnt~weathersit+temp+windspeed+weekday+season+casual+registered, data = input)
14. > print(model)
15. Call:
16. lm(formula = cnt ~ weathersit + temp + windspeed + weekday +
17. season + casual + registered, data = input)
18. Coefficients:
19. (Intercept) weathersit temp windspeed weekday season
20. 47.4377 -3.6233 156.1502 -174.6765 -3.9308 -8.8670
21. casual registered
22. 0.9611 0.9998
23. > summary(model)
24. Call:
25. lm(formula = cnt ~ weathersit + temp + windspeed + weekday +
26. season + casual + registered, data = input)
27. Residuals:
28. Min 1Q Median 3Q Max
29. -90.60 -42.29 -25.56 -6.01 1536.21
30. Coefficients:
31. Estimate Std. Error t value Pr(>|t|)
32. (Intercept) 4.744e+01 4.971e+01 0.954 0.3404
33. weathersit -3.623e+00 1.709e+01 -0.212 0.8321
34. temp 1.562e+02 6.681e+01 2.337 0.0198 \*
35. windspeed -1.747e+02 1.305e+02 -1.339 0.1812
36. weekday -3.931e+00 4.613e+00 -0.852 0.3945
37. season -8.867e+00 8.966e+00 -0.989 0.3232
38. casual 9.611e-01 2.538e-02 37.874 <2e-16 \*\*\*
39. registered 9.998e-01 7.278e-03 137.362 <2e-16 \*\*\*
40. ---
41. Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1
42. Residual standard error: 194.1 on 481 degrees of freedom
43. Multiple R-squared: 0.9891, Adjusted R-squared: 0.989
44. F-statistic: 6252 on 7 and 481 DF, p-value: < 2.2e-16
45. ## Residual standard error: 0.648 Adjusted R-squared: 0.3561
46. ## F-statistic: 81.35 on 11 and 1587 DF, p-value: < 2.2e-16

# Support Vector Regression



### **Chapter 3**

**Conclusion**

1. **Model Evaluation**

**Now that we have a few models for predicting the target variable, we need to decide which one to choose. There are several criteria that exist for evaluating and comparing models. We can compare the Models using any of the following criteria.**

1. **Predictive Performance**
2. **Interpretability**
3. **Computational Efficiency**

**As we are calculating Bike Rental Count, we need good interpretability. Therefore, we use Predictive Performance as the criteria to compare and Evaluate models.**

**Interpretability can be measured by the Methodology of Understanding a model can provide which can be measure by Metrics such as R Square, Adjusted R square, etc.**

* 1. **R Square & Adjusted R Square**

1. > summary(model)
2. Call:
3. lm(formula = cnt ~ weathersit + temp + windspeed + weekday +
4. season + casual + registered, data = input)
5. Residuals:
6. Min 1Q Median 3Q Max
7. -1.296e-11 -1.795e-13 -4.980e-14 1.385e-13 1.339e-11
8. Coefficients:
9. Estimate Std. Error t value Pr(>|t|)
10. (Intercept) 2.526e-13 2.191e-13 1.153e+00 0.249
11. weathersit -3.975e-13 7.546e-14 -5.268e+00 1.99e-07 \*\*\*
12. temp -1.567e-12 2.759e-13 -5.679e+00 2.22e-08 \*\*\*
13. windspeed 2.252e-13 5.060e-13 4.450e-01 0.656
14. weekday 2.997e-14 1.911e-14 1.569e+00 0.117
15. season 7.533e-13 3.901e-14 1.931e+01 < 2e-16 \*\*\*
16. casual 1.000e+00 7.001e-17 1.428e+16 < 2e-16 \*\*\*
17. registered 1.000e+00 3.196e-17 3.129e+16 < 2e-16 \*\*\*
18. ---
19. Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1
20. Residual standard error: 8.901e-13 on 541 degrees of freedom
21. Multiple R-squared: 1, Adjusted R-squared: 1
22. F-statistic: 3.631e+32 on 7 and 541 DF, p-value: < 2.2e-16

**5.2**  **Model Selection**

We can see that Multiple Linear Regression perform comparatively on average and therefore we can select Multiple Linear Regression of the three models without any loss of information.

### Appendix A- Extra Figures

Figure 3.1 Quality Boxplots for all the Predictor Variables

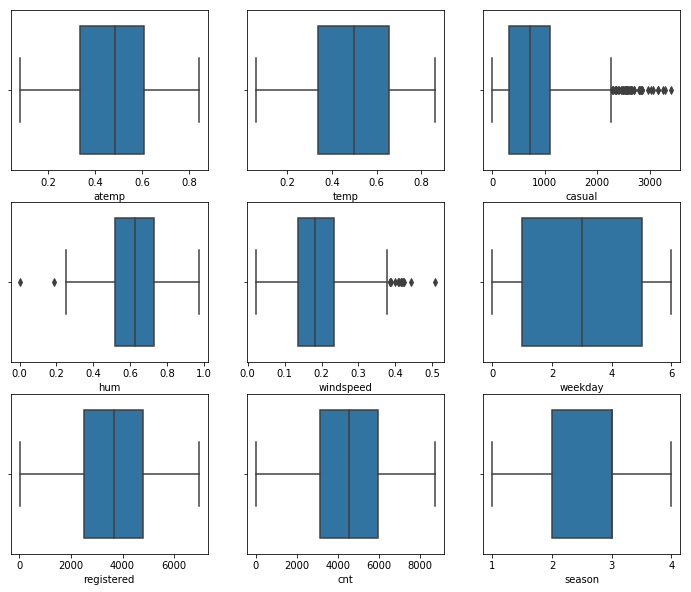


Figure 3.2 Quality Boxplots for Multiple Predictors

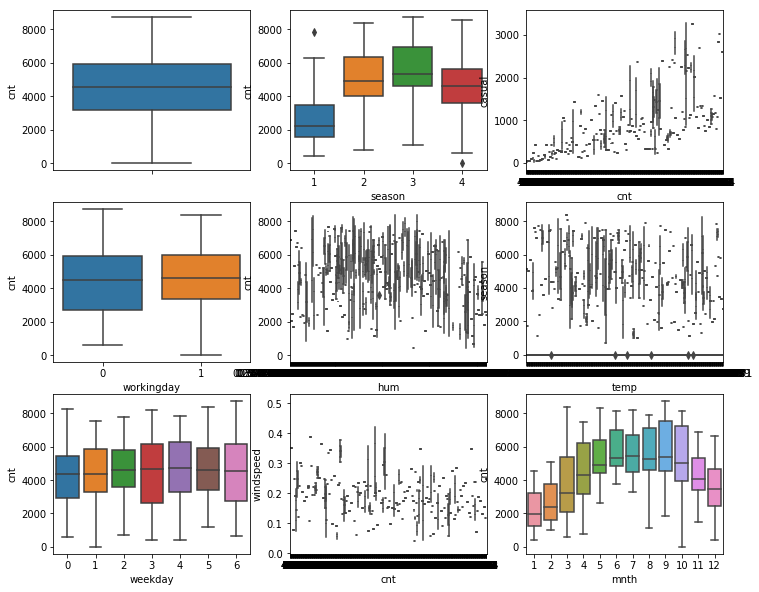


Figure 3.3 Correlation Matrix

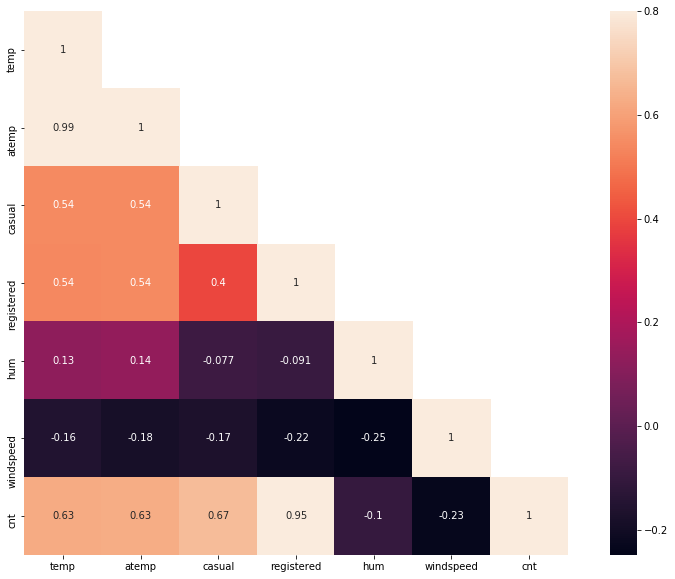


Figure 3.4 Subplots for Temperature, Humidity & Windspeed

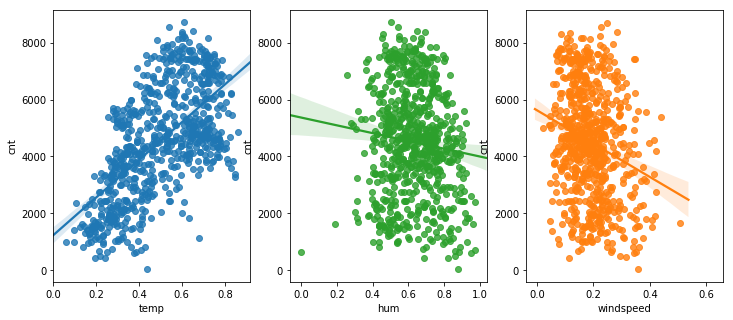
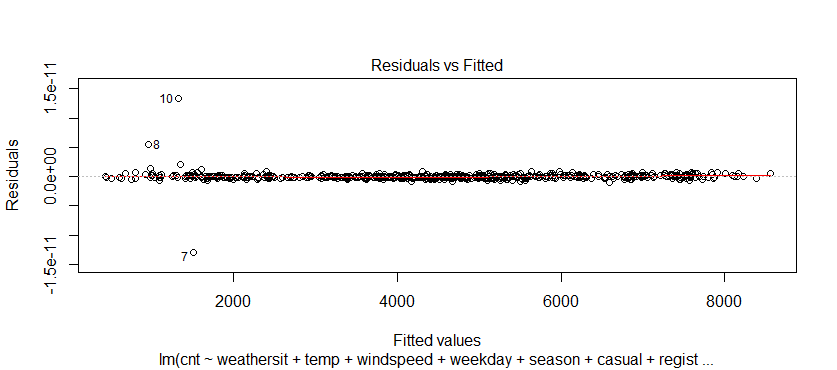
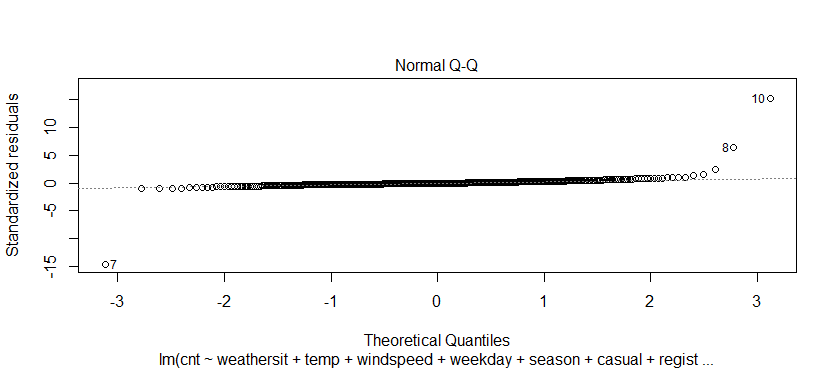


Figure 3.5 Plotting our Multiple Linear Regression Model

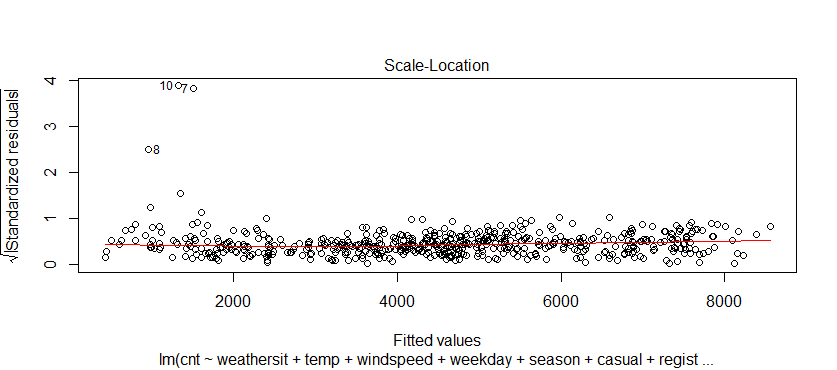
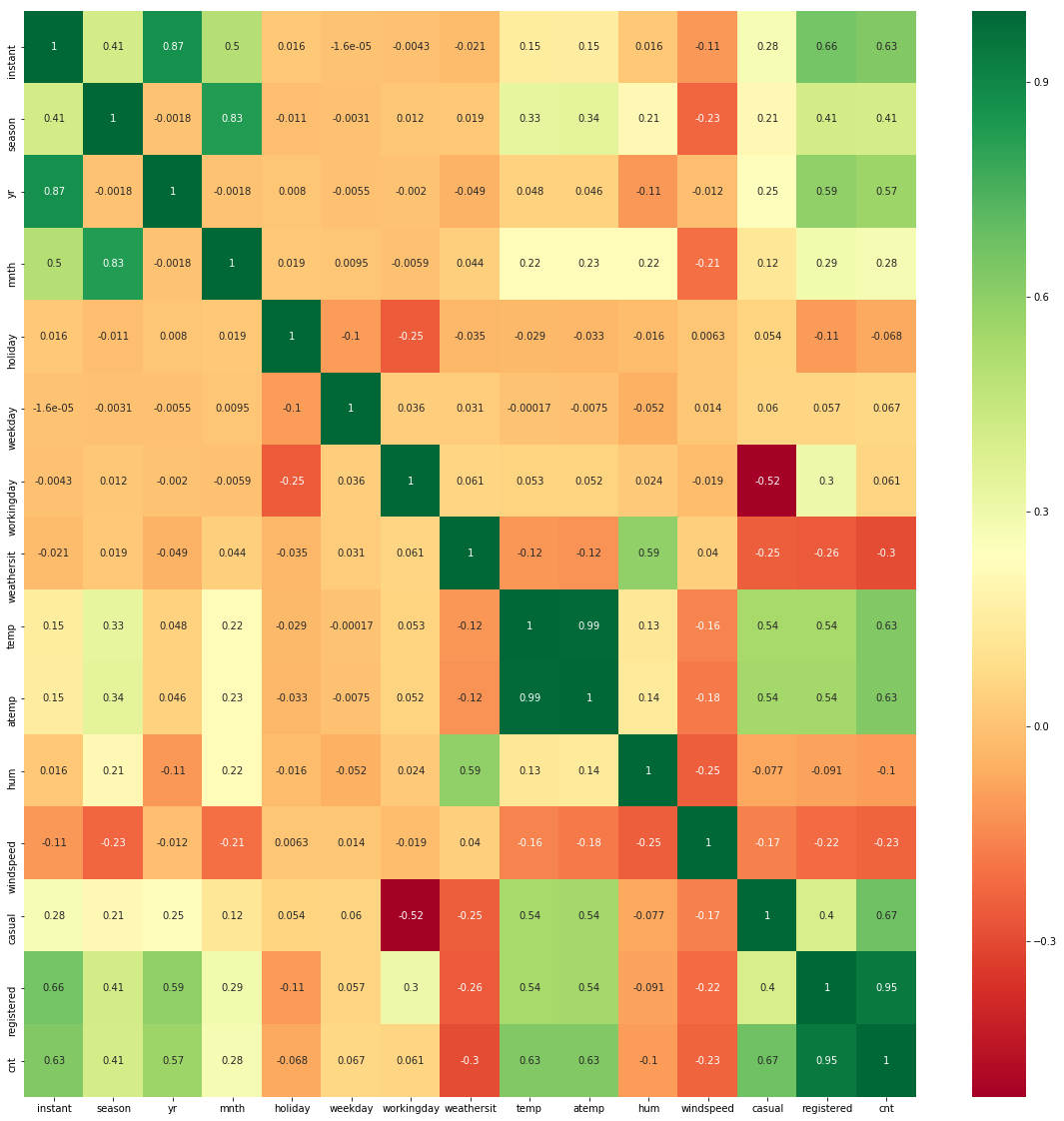


Figure 3.6 – Heat Map of all Predictor Values



# **Appendix B – R Code**

Exploratory Data Analysis- Plotting The Count of Rental Along with the Temperature(Figure 3.7)

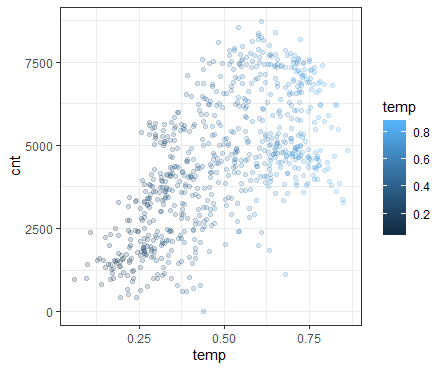


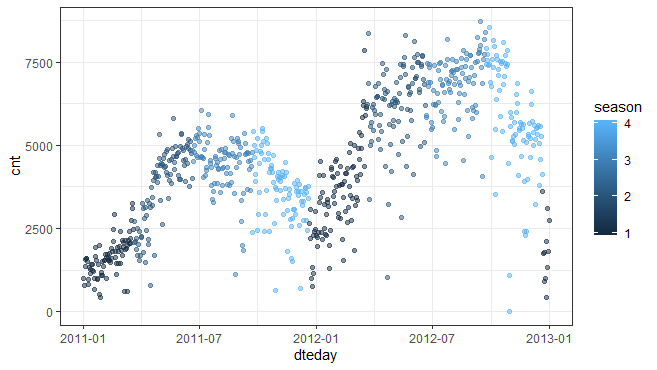
Figure 3.6- Plotting The Count of Rental Along with the Temperature

# Exploratory Data Analysis

# Plotting the Count of Rental along with the Temprature

ggplot(df, aes(temp, cnt)) +

geom\_point(aes(color=temp),alpha=0.2) + theme\_bw()



(Figure 3.8) -Plotting Against Each Day

# Plotting count against Day

ggplot(df, aes(dteday, cnt)) +

geom\_point(aes(color=season),alpha=0.5) + theme\_bw()

# Plotting count against Season

ggplot(df, aes(season, cnt)) +

geom\_boxplot(aes(color=season),alpha=0.5) + theme\_bw()

**Complete R File**

library(ggplot2) # Visualisation

library(dplyr) # Data Wrangling

library(e1071) # Prediction: SVR

library(randomForest) # Prediction: Random Forest

df = read.csv('F:/Downloads/day.csv')

x = c("ggplot2", "corrgram", "DMwR", "caret", "randomForest", "unbalanced", "C50", "dummies", "e1071", "Information", "knitr", "xtable", "psych",

"MASS", "rpart", "gbm", "ROSE", 'sampling', 'DataCombine', 'inTrees')

# install.packages("caTools")

# install.packages("ie2misc")

library(ie2misc)

library(tidyverse)

library(rJava)

library(caTools)

# install.packages(x)

lapply(x, require, character.only = TRUE)

tail(df,250)

rm(x)

df$dteday = as.POSIXct(df$dteday, format="%Y-%m-%d")

str(df)

summary(df)

# Checking for Missing Data

sapply(df, function(x) sum(is.na(x)))

# Exploratory Data Analysis

# Plotting the Count of Rental along with the Temprature

ggplot(df, aes(temp, cnt)) +

geom\_point(aes(color=temp),alpha=0.2) + theme\_bw()

# Plotting count against Day

ggplot(df, aes(dteday, cnt)) +

geom\_point(aes(color=season),alpha=0.5) + theme\_bw()

# Plotting count against Season

ggplot(df, aes(season, cnt)) +

geom\_boxplot(aes(color=season),alpha=0.5) + theme\_bw()

# Outlier Analysis

numeric\_index = sapply(df,is.numeric)

numeric\_index

numeric\_data = df[,numeric\_index]

cnames = colnames(numeric\_data)

for(i in 1:length(cnames))

{

assign(paste0("gn",i), ggplot(aes\_string(y = (cnames[i]), x = "cnt"), data = subset(df))+

stat\_boxplot(geom = "errorbar", width = 0.5) +

geom\_boxplot(outlier.colour="red", fill = "grey" ,outlier.shape=18,

outlier.size=1, notch=FALSE) +

theme(legend.position="bottom")+

labs(y=cnames[i],x="Count")+

ggtitle(paste("Box plot of",cnames[i])))

}

# Plot the Graph

gridExtra::grid.arrange(gn1,gn5,gn2,ncol=3)

gridExtra::grid.arrange(gn6,gn7,ncol=2)

gridExtra::grid.arrange(gn8,gn9,ncol=2)

# We will now be applying the Multiple Linear Regression

# For Fitting our regression models first we have to split the data into Training and Testing

sample = sample.split(df,SplitRatio = 0.75) # splits the data in the ratio mentioned in SplitRatio. After splitting marks these rows as logical TRUE and the the remaining are marked as logical FALSE

train1 = subset(df,sample == TRUE)

test1 = subset(df, sample == FALSE)

train1

test1

# Check for Missing Value once,

sapply(train1, function(x) sum(is.na(x)))

# Now fitting Multiple Linear Regression to the Training Set.

input <- train1[,c("weathersit", "temp", "windspeed", "weekday", "hum", "mnth", "season", "casual", "registered", "cnt")]

print(head(input))

# Creating a relationship Model and Getting the coefficiants

model <- lm(cnt~weathersit+temp+windspeed+weekday+season+casual+registered, data = input)

print(model)

summary(model)

lrm.pred.red <- predict(model, input)

lrm.pred.red <- mean(abs(lrm.pred.red))

lrm.pred.red

# Cor weathersit and Temp

cor(train1$weathersit, train1$windspeed, method = "pearson")

# Ask the confidence intervals for the model coefficients

confint(model, conf.level=0.95)

# Lets check the regression diagontic plots for this model

plot(model)

regressor = lm (formula= cnt~ . , data = train1)

confint(regressor, conf.level=0.95)

plot(regressor)

# Choosing the best model in AIC in a stepwise Algorithm

# The Step( function iteratively removes insignificant features from the model

regressor = step(regressor)

y\_pred = predict(regressor, test1)

y\_pred

results <- data.frame(datetime = test1$dteday, count = y\_pred)

write.csv(results, file = 'BikeSharingDemand\_MLR.csv', row.names = FALSE, quote=FALSE)

lrm.pred.red <- predict(regressor, train1)

lrm.pred.red <- mean(abs(lrm.pred.red))

lrm.pred.red

# Support Vector Regression

regressor = svm(formula = cnt ~ .,

data = train1,

type = 'eps-regression',

kernel = 'radial')

summary(regressor)

y\_pred = predict(regressor, test1)

results <- data.frame(datetime = test1$dteday, count = y\_pred)

write.csv(results, file = 'BikeSharingDemand\_SVR.csv', row.names = FALSE, quote=FALSE)

summary(result)

plot(regressor)

# Random Forest Regression

regressor = randomForest(x = train1[,-which(names(train1)=="cnt")],

y = train1$cnt)

# Predicting a new result with Random Forest Regression

y\_pred = predict(regressor, test1)

results <- data.frame(datetime = test1$dteday, count = y\_pred)

# Write the results to a csv file

write.csv(results, file = 'BikeSharingDemand\_RandomForest.csv', row.names = FALSE, quote=FALSE)

# Mean Absolute error

lrm.pred.red <- predict(regressor, train1)

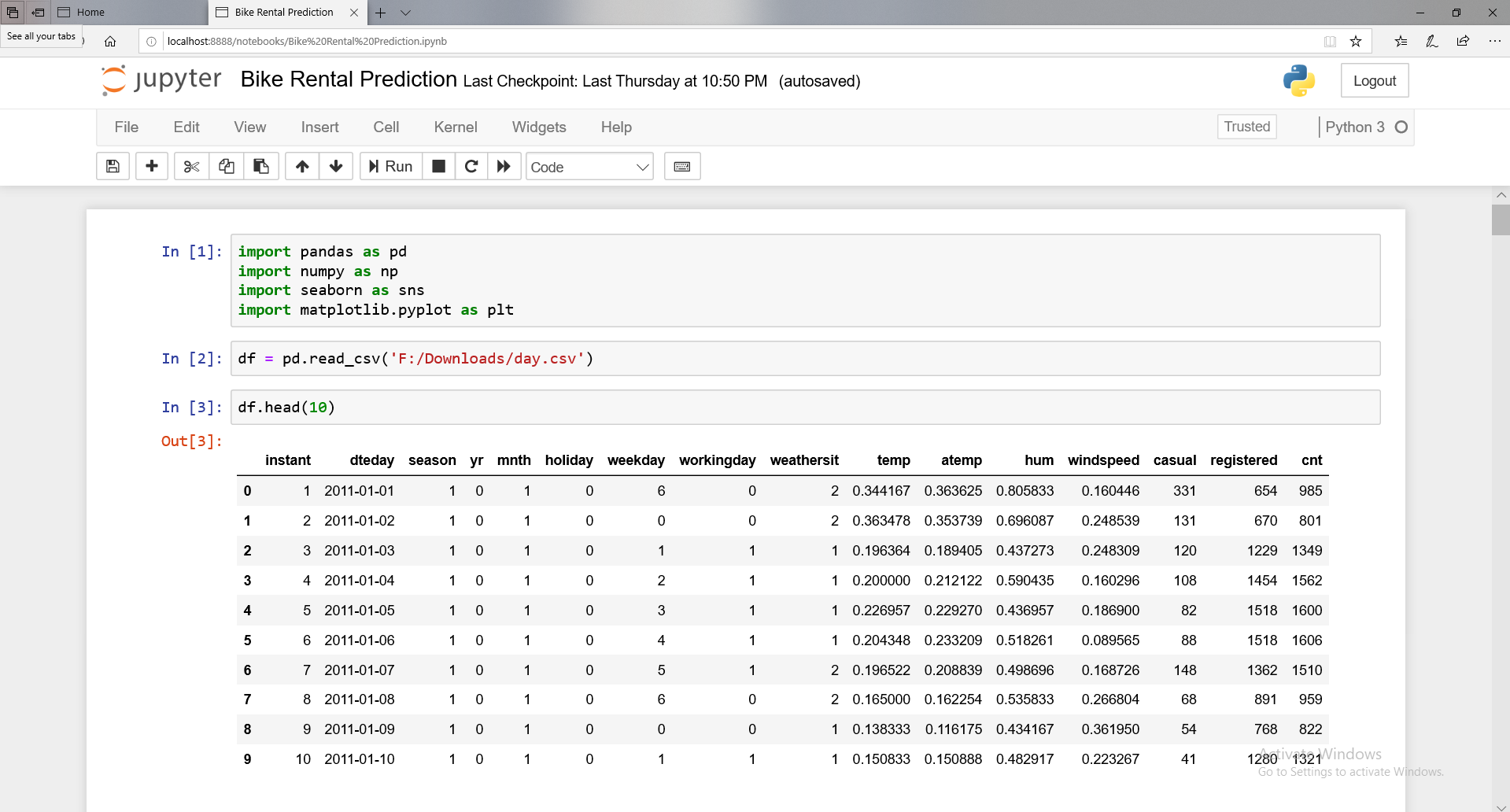
lrm.pred.red <- mean(abs(lrm.pred.red))

lrm.pred.red

plot(regressor)

summary(regressor)

# **Appendix C- Daily Bike Rental Count Prediction & Code Files**



Prediction of Bike Rental Count on Daily Basis Until 29th December, 2012 files are attached as above.