# PREDICTION OF BIKE RENTAL COUNT

## MONIKA WADHWANI

15 November 2019

# Contents

# Chapter 1: Introduction

## 1.     Problem Statement

The aim of this project is to predict the count of bike rentals based on the seasonal and environmental settings. By predicting the count, it would be possible to help accommodate in managing the number of bikes required on a daily basis, and being prepared for high demand of bikes during peak periods.

## 2.     Data

The goal is to build regression models which will predict the number of bikes used based on the environmental and season behaviour. Given below is a sample of the data set that we are using to predict the number of bikes:

Table 1.1: Bike Count Sample Data (Columns: 1-9)

| instant | dteday | season | yr | mnth | holiday | weekday | workingday | weathersit |
|---|---|---|---|---|---|---|---|---|
| 1 | 2011 01 01 | 1 | 0 | 1 | 0 | 6 | 0 | 2 |
| 2 | 2011 01 02 | 1 | 0 | 1 | 0 | 0 | 0 | 2 |
| 3 | 2011 01 03 | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 4 | 2011 01 04 | 1 | 0 | 1 | 0 | 2 | 1 | 1 |
| 5 | 2011 01 05 | 1 | 0 | 1 | 0 | 3 | 1 | 1 |

Table 1.2: Bike Count Sample Data (Columns: 10-16)

| temp | atemp | hum | windspeed | casual | registered | cnt |
|---|---|---|---|---|---|---|
| 0.344170 | 0.363625 | 0.805833 | 0.1604460 | 331 | 654 | 985 |
| 0.363478 | 0.353739 | 0.696087 | 0.2485390 | 131 | 670 | 801 |
| 0.196364 | 0.189405 | 0.437273 | 0.2483090 | 120 | 1229 | 1349 |
| 0.2000000 | 0.2121220 | 0.590435 | 0.1602960 | 108 | 1454 | 1562 |
| 0.2269570 | 0.2292700 | 0.436957 | 0.1869000 | 82 | 1518 | 1600 |

As you can see in the table below we have the following 13 variables, using which we have to correctly predict the count of bikes:

| Sl. No | Variables |
|---|---|
| 1 | Instant |
| 2 | Dteday |
| 3 | Season |
| 4 | Yr |
| 5 | Month |
| 6 | Holiday |
| 7 | Weekday |
| 8 | Workingday |
| 9 | Weathersit |
| 10 | Temp |
| 11 | Atemp |
| 12 | Hum |
| 13 | windspeed |

Table 1.3: Predictor variables

# Chapter 2: Methodology

## 1.     Pre-Processing

A predictive model requires that we look at the data before we start to create a model. However, in data mining, looking at data refers to exploring the data, cleaning the data as well as visualising the data through graphs and plots. This is known as Exploratory Data Analysis.

## 2.     Distribution of continuous variables

It can be observed from the below histograms is that temperature and feel temperature are normally distributed, where as the variables windspeed and humidity are slightly skewed.
The skewness is likely because of the presence of outliers and extreme data in those variables.
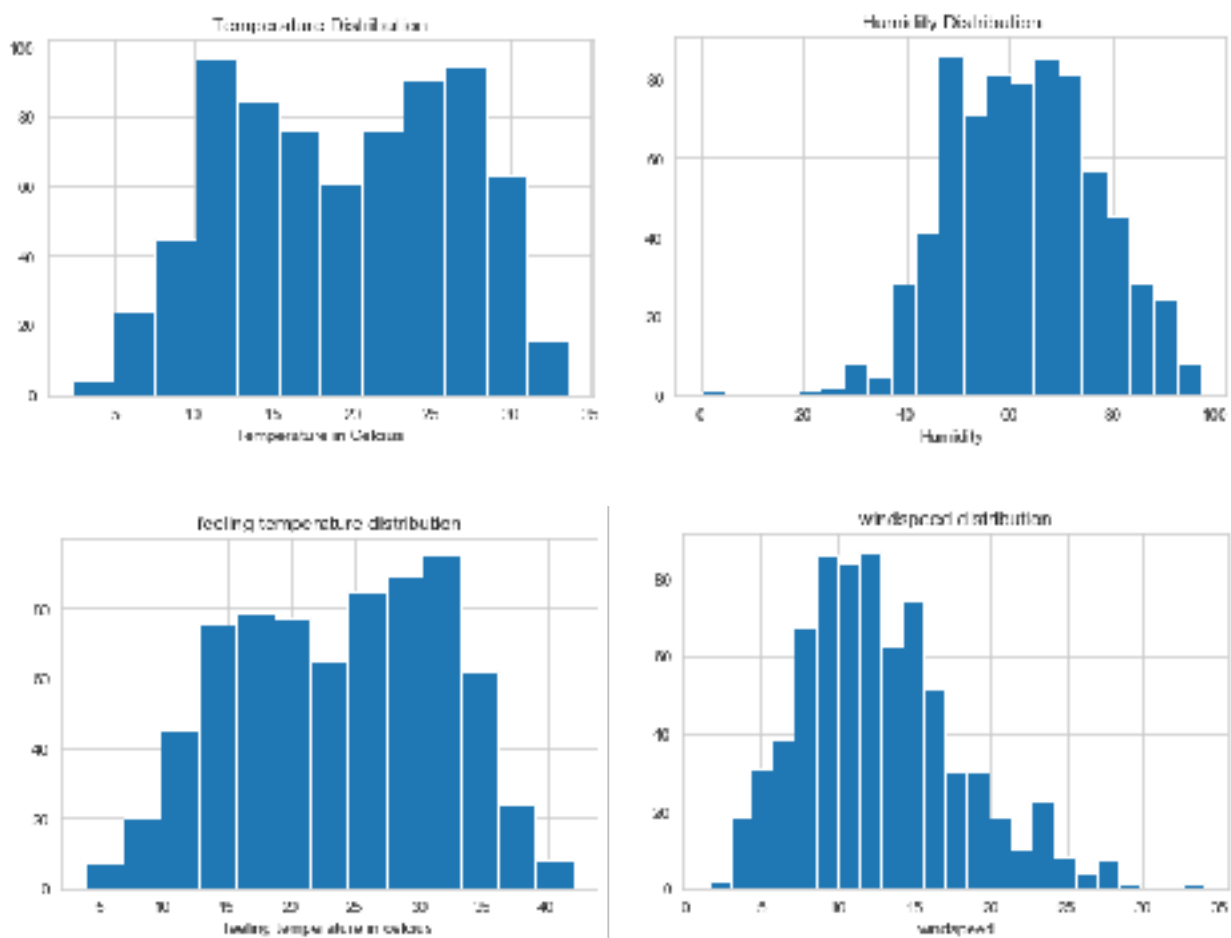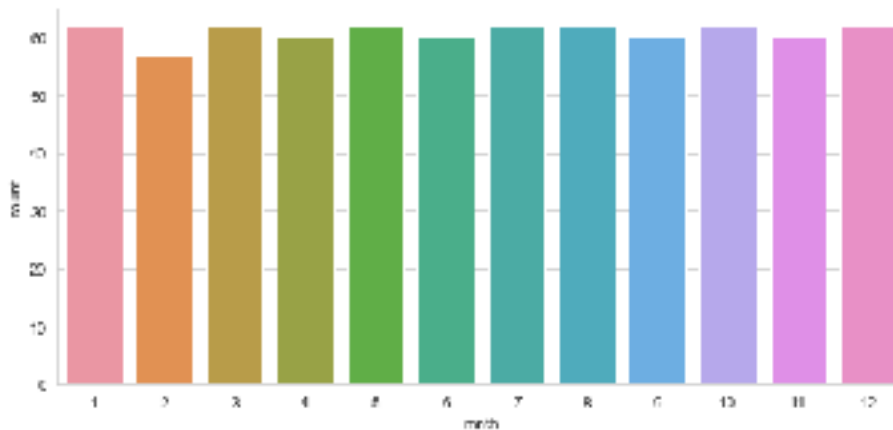


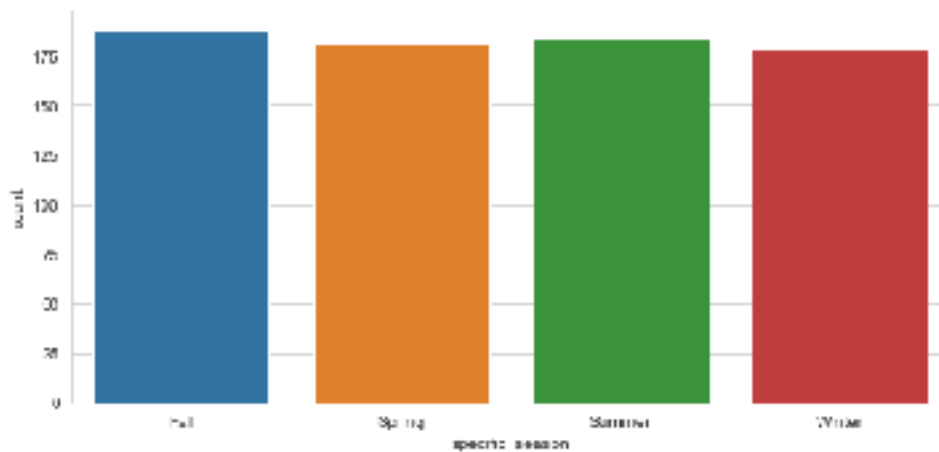Fig 2.1: Distribution of continuous variables using Histograms

## 3.    Distribution of categorical variables

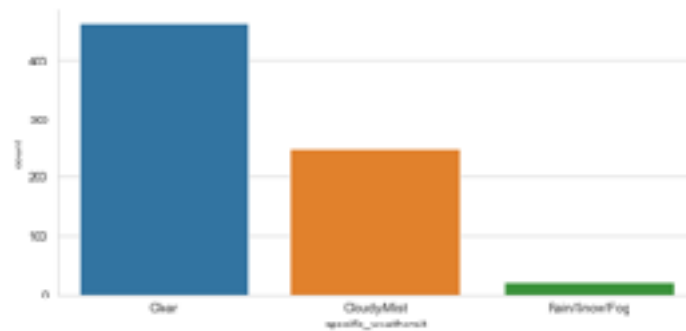The distribution of categorical variables is as shown in the below figure:

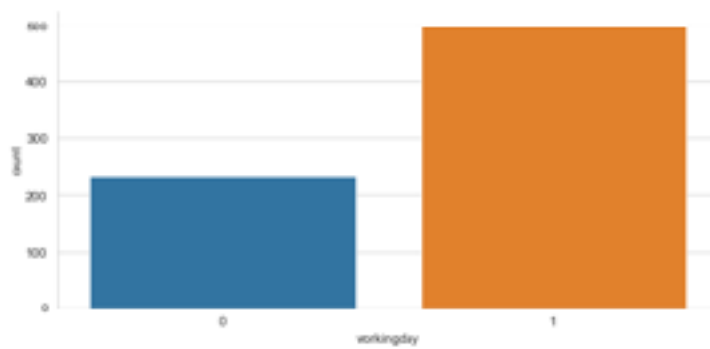Bar graph for distribution of Months on x-axis and bike rental count on y-axis



Bar graph for distribution of season on x-axis and bike rental count on y-axis

Bar graph for distribution of weatherit



Bar graph for distribution of working day and weekend/holiday



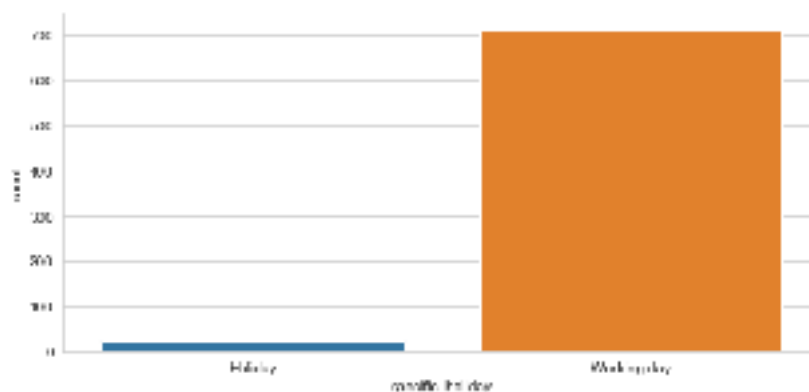Bar graph for distribution of holiday and working day



Fig 2.2: Distribution of categorical variables using bar plots

## Relationship of Continuous variables against bike count

The below figure shows the relationship between continuous variables and the target variable using scatter plot. It can be observed that there exists a linear positive relationship between the variables temperature and feel temperature with the bike rental count. There also exists a negative linear relationship between the variable's humidity and windspeed with the bike rental count.



Fig

2.3:

Scatter plot for continuous variables

## 5. :   Detection of outliers:

Outliers are detected using boxplots. Below figure illustrates the boxplots for all the continuous variables.
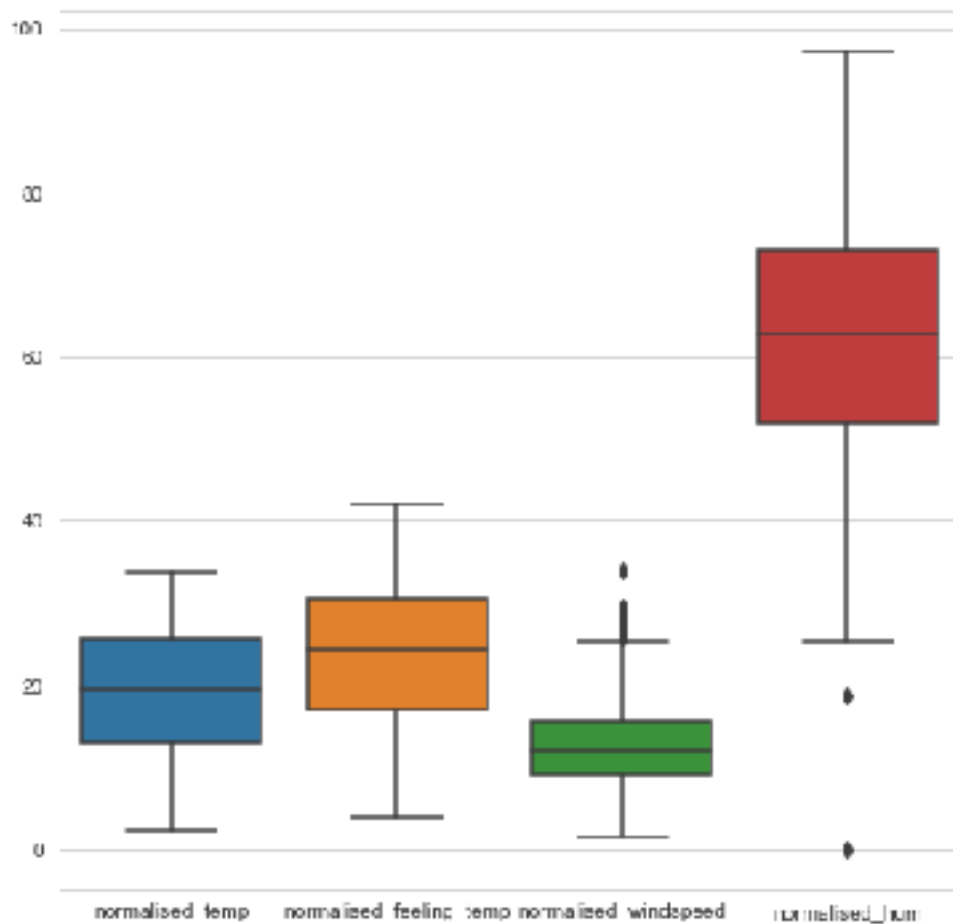


Fig 2.4: Boxplot of continuous variables

Outliers can be removed using the Boxplot stats method, wherein the Inter Quartile Range (IQR) is calculated and the minimum and maximum value are calculated for the variables. Any value ranging outside the minimum and maximum value are discarded. The boxplot of the continuous variables after removing the outliers is shown in the below figure:

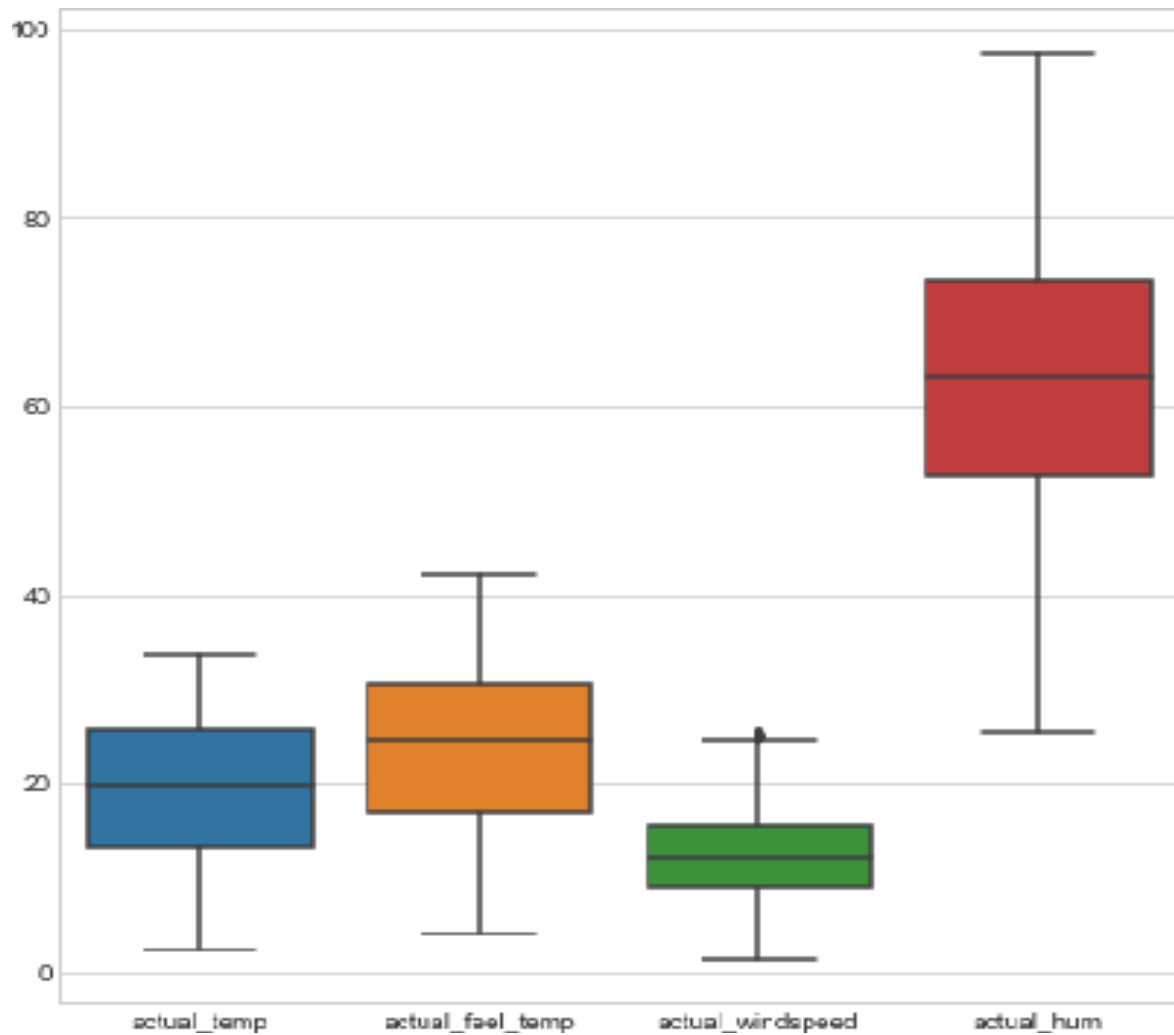Fig 2.5: Boxplot of continuous variables after removal of outliers

It can be observed from the distribution of Windspeed and humidity after removal of outliers, is that data is not skewed as much as before the removal of outliers. The figure shown below illustrates the distribution of continuous variables using histograms.
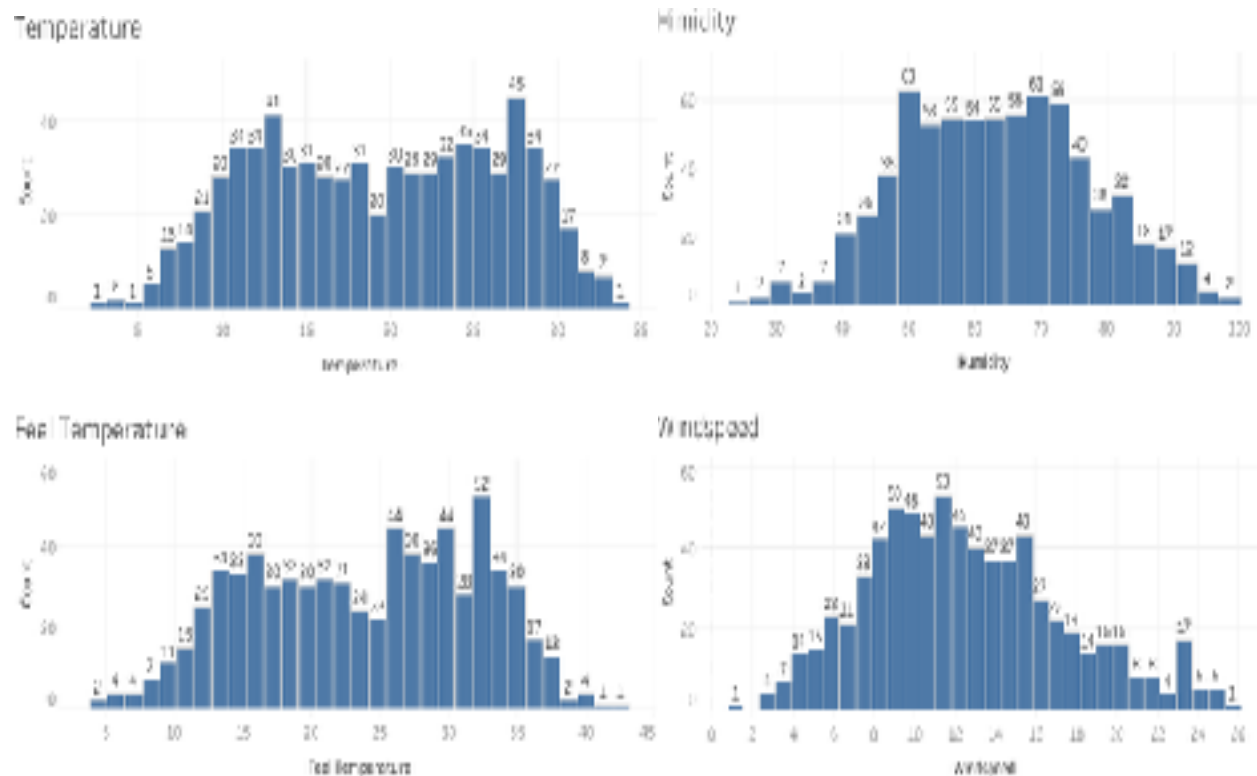
Fig 2.6: Distribution of numerical data using histograms after removal of outliers

## 6. : Feature Selection

Feature Selection reduces the complexity of a model and makes it easier to interpret. It also reduces overfitting. Features are selected based on their scores in various statistical tests for their correlation with the outcome variable.
Correlation plot is used to find out if there is any multicollinearity between variables. The highly collinear variables are dropped and then the model is executed.
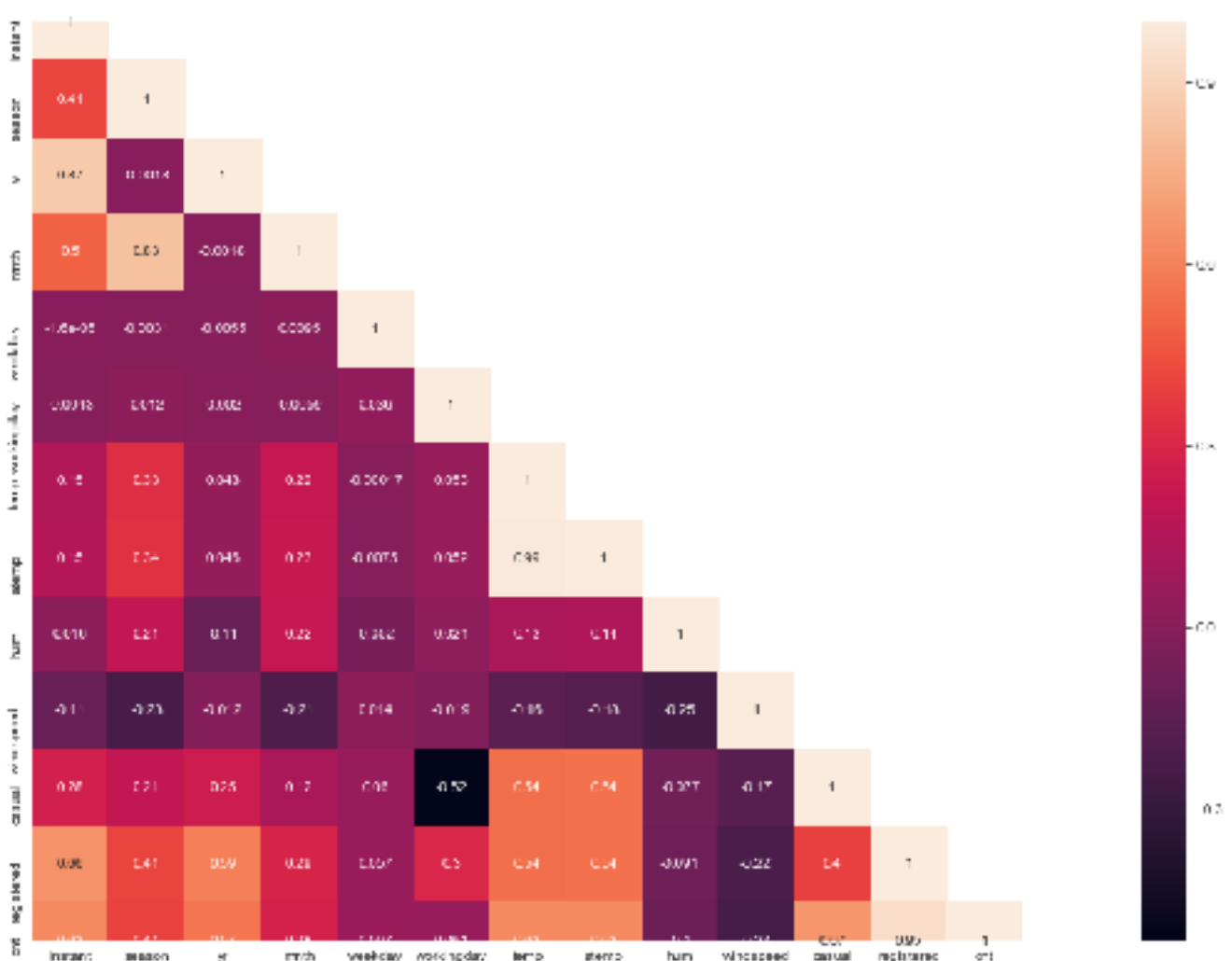


Fig 2.7: Correlation plot of all the variables

# Chapter 3: Modelling

## 1.    Model Selection

The dependent variable in our model is a continuous variable i.e., Count of bike rentals. Hence the models that we choose are Linear Regression, Decision Tree and Random Forest. The error metric chosen for the problem statement is Mean Absolute Error (MAE).

## 2.    Multiple Linear Regression

Multiple linear regression is the most common form of linear regression analysis. Multiple linear regression is used to explain the relationship between one continuous dependent variable and two or more independent variables. The independent variables can be continuous or categorical.

```
Call:
lm(formula = cnt ~ ., data = train)

Residuals:
    Min      1Q  Median      3Q     Max
 4014.3   341.8    77.7   467.5  2900.0

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    1521.86     271.45   5.606 3.28e-08 ***
season2         795.42     209.72   3.793 0.000166 ***
season3         960.31     252.49   3.803 0.000159 ***
season4        1639.81     207.96   7.885 1.72e-14 ***
yr1            2051.30      68.44  29.974  < 2e-16 ***
mnth2           195.05     171.97   1.134 0.257211
mnth3           554.12     195.04   2.841 0.004664 **
mnth4           533.72     286.19   1.865 0.062728 .
mnth5           885.32     309.63   2.859 0.004409 **
mnth6           636.14     325.81   1.953 0.051389 .
mnth7            24.72     363.78   0.068 0.945838
mnth8           246.58     357.38   0.690 0.490514
mnth9           920.80     401.95   2.971 0.003107 **
mnth10          495.87     279.68   1.773 0.076789 .
mnth11          160.50     265.88   0.604 0.546423
mnth12          162.47     210.48   0.772 0.440512
weekday1        536.15     212.60   2.522 0.011957 *
weekday2        467.51     234.45   1.994 0.046642 *
weekday3        363.07     234.88   1.546 0.122799
weekday4        357.59     234.41   1.526 0.127708
weekday5        338.47     233.02   1.452 0.146996
weekday6        427.46     126.44   4.385 0.000718 ***
workingday1     738.50     200.48   4.686 0.000357 ***
weathersit2     450.08      88.45   5.088 4.98e-07 ***
weathersit3    1960.75     215.77   9.087  < 2e-16 ***
temp           4473.94     493.07   8.953  < 2e-16 ***
hum            1500.11     333.95   4.492 8.62e-06 ***
windspeed      2748.98     504.15   5.453 7.53e-08 ***

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 797.8 on 546 degrees of freedom
Multiple R-squared:  0.845,     Adjusted R-squared:  0.8473
F-statistic: 110.2 on 27 and 546 DF,  p-value: < 2.2e-16
```

As you can see the Adjusted R-squared value, we can explain 83.73% of the data using our multiple linear regression model. By looking at the F-statistic and

combined p-value we can reject the null hypothesis that target variable does not depend on any of the predictor variables. This model explains the data very well and is considered to be good.

Even after removing the non-significant variables, the accuracy, Adjusted R-squared and F- statistic do not change by much, hence the accuracy of this model is chosen to be final.
Mean Absolute Error (MAE) is calculated and found to be 494.
MAPE of this multiple linear regression model is 12.17%. Hence the accuracy of this model is 87.83%. This model performs very well for this test data.

### 3. Decision Tree:

A decision tree can be used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions.
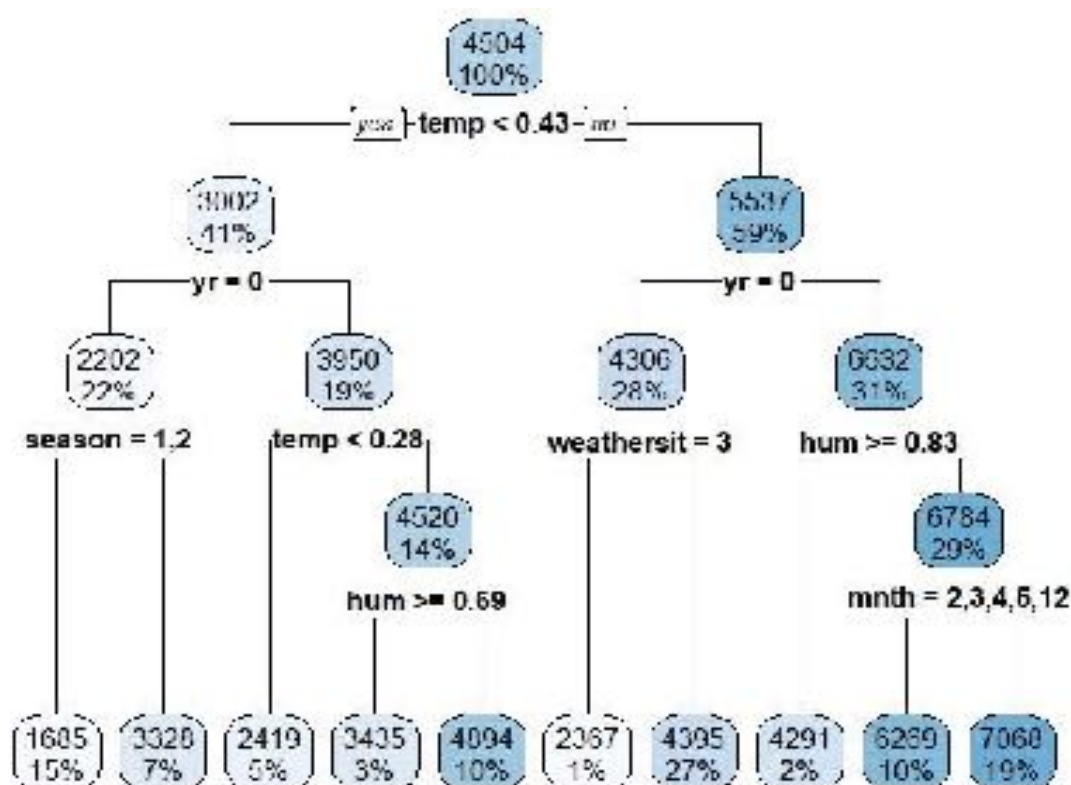


Using decision tree, we can predict the value of bike count. MAE for this model is 684. The MAPE for this decision tree is 17.47%. Hence the accuracy for this model is 82.53%.

### 4. Random Forest:

Using Classification for prediction analysis in this case is not normal, though it can be done. The number of decision trees used for prediction in the forest is 500.

MAE for this model is

392. Using random forest, the MAPE was found to be 10.68%. Hence the accuracy is 89.32%.

# Chapter 4: Conclusion

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

In our case of Bike count prediction Data, Interpretability and Computation Efficiency, do not hold much significance. Therefore, we will use Predictive performance as the criteria to compare and evaluate models.

Predictive performance can be measured by comparing Predictions of the models with real values of the target variables, and calculating some average error measure.

### 4.1    Mean Absolute Error (MAE)

MAE is one of the error measures used to calculate the predictive performance of the model. We will apply this measure to our models that we have generated in the previous section.

*MAE <- function (actual, pred)*
*{*
*  print(mean (abs (actual - pred)))*
*}*

**Linear Regression Model: MAE =**
**494 Decision Tree: MAE = 684.**
**Random Forest: MAE = 392**

Based on the above error metrics, Random Forest is the better model for our analysis. Hence Random Forest is chosen as the model for prediction of bike rental count.
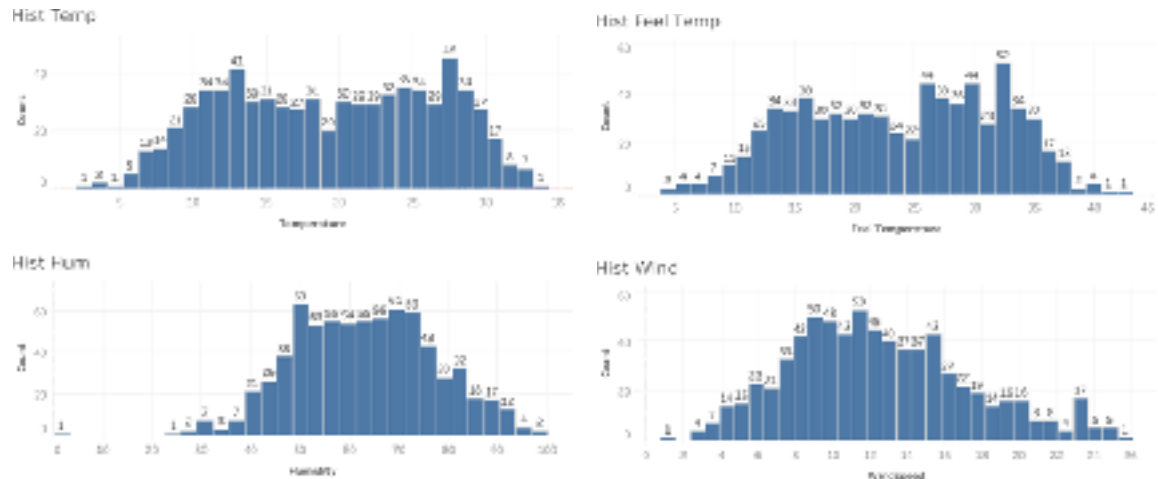
# Chapter 5: Appendix

## 5.1    Figures



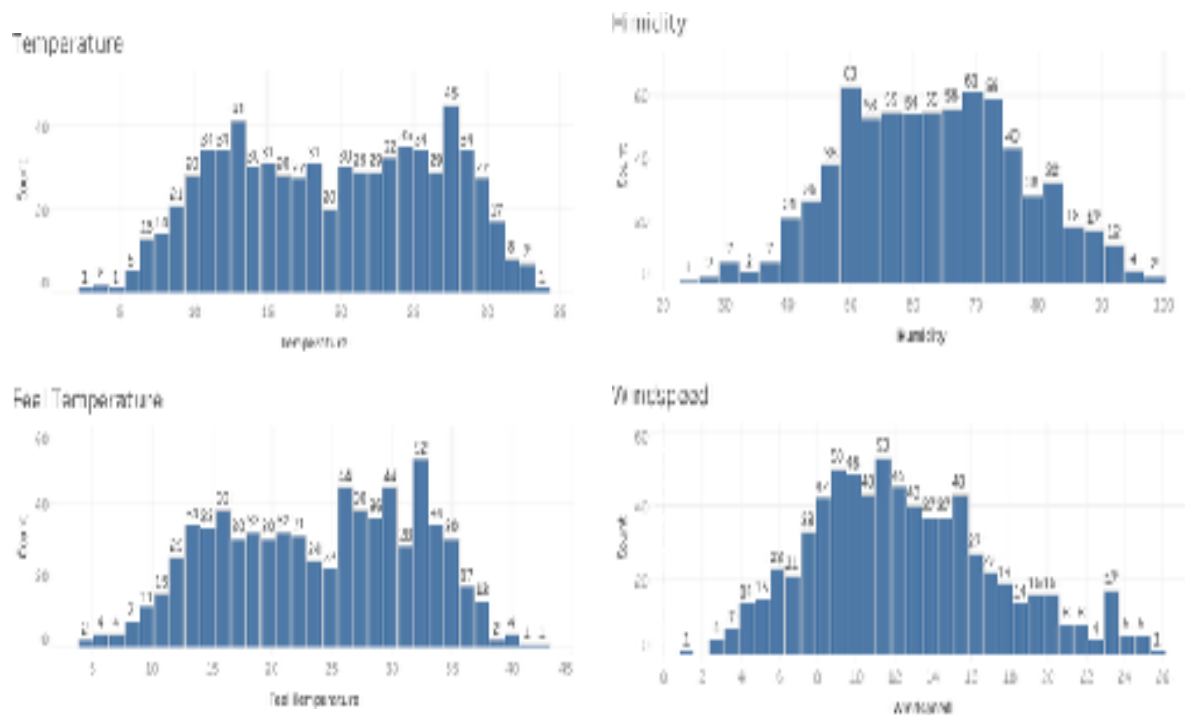**Fig 2.1: Distribution of continuous variables using Histograms**
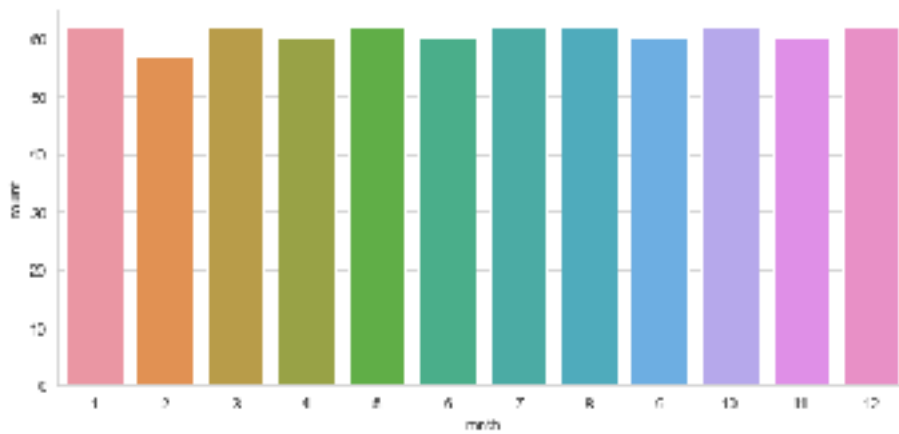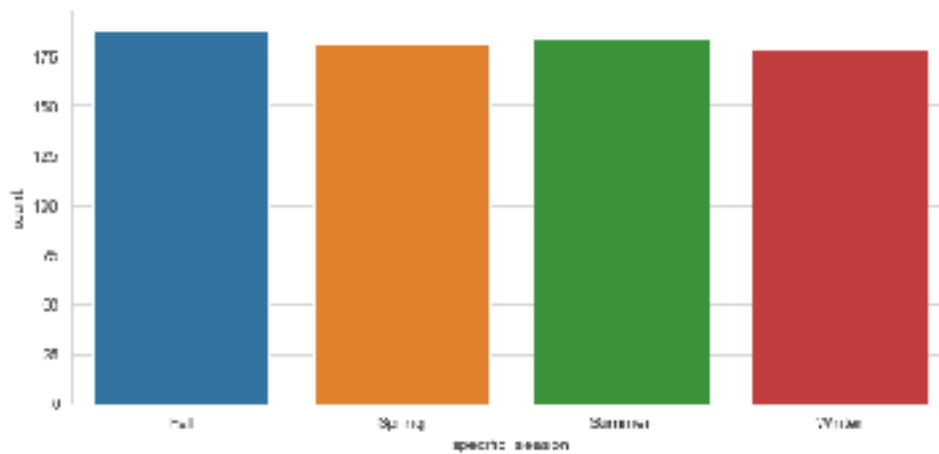


**Fig 2.6: Distribution of numerical data using histograms after removal of outliers**
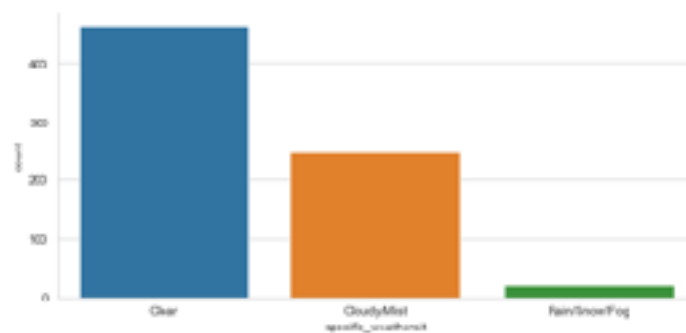
Bar graph for distribution of Months on x-axis and bike rental count on y-axis
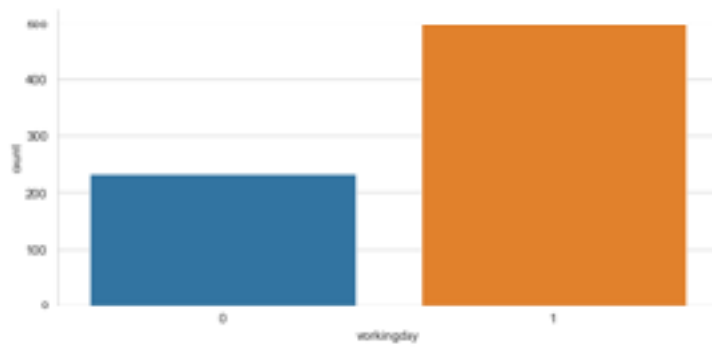


Bar graph for distribution of season on x-axis and bike rental count on y-axis



Bar graph for distribution of weatherit

Bar graph for distribution of working day and weekend/holiday



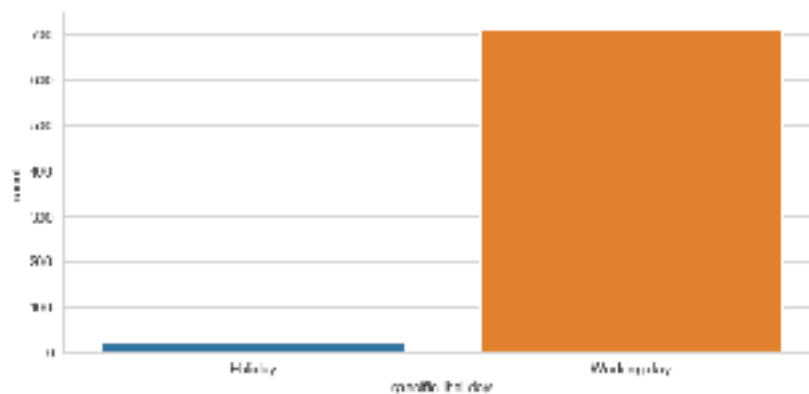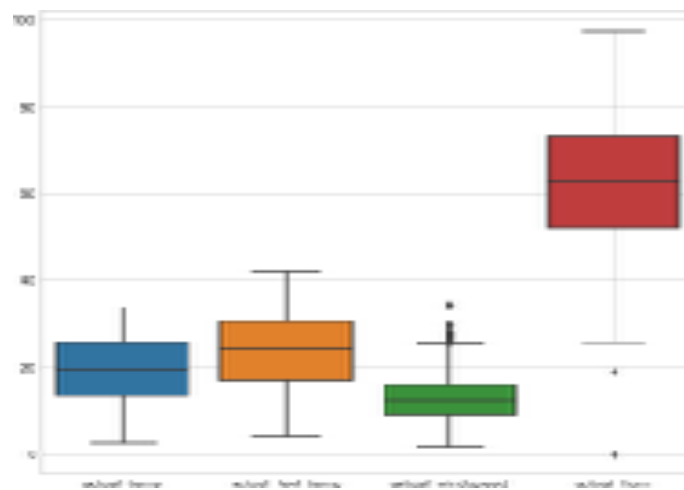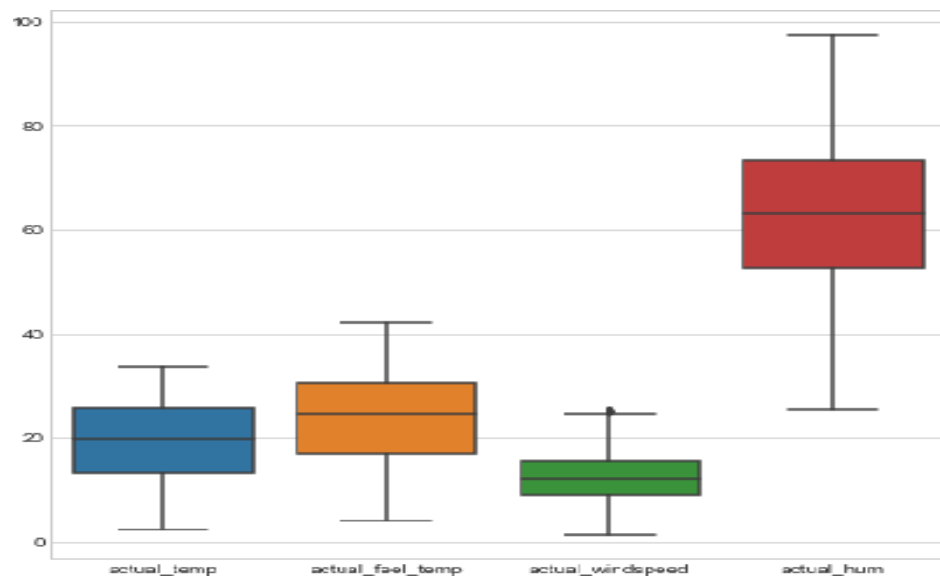Bar graph for distribution of holiday and working day



Fig 2.2: Distribution of categorical variables using bar plots

**Fig 2.3: Scatter plot for continuous variables**

**Fig 2.4: Boxplot of continuous variables**

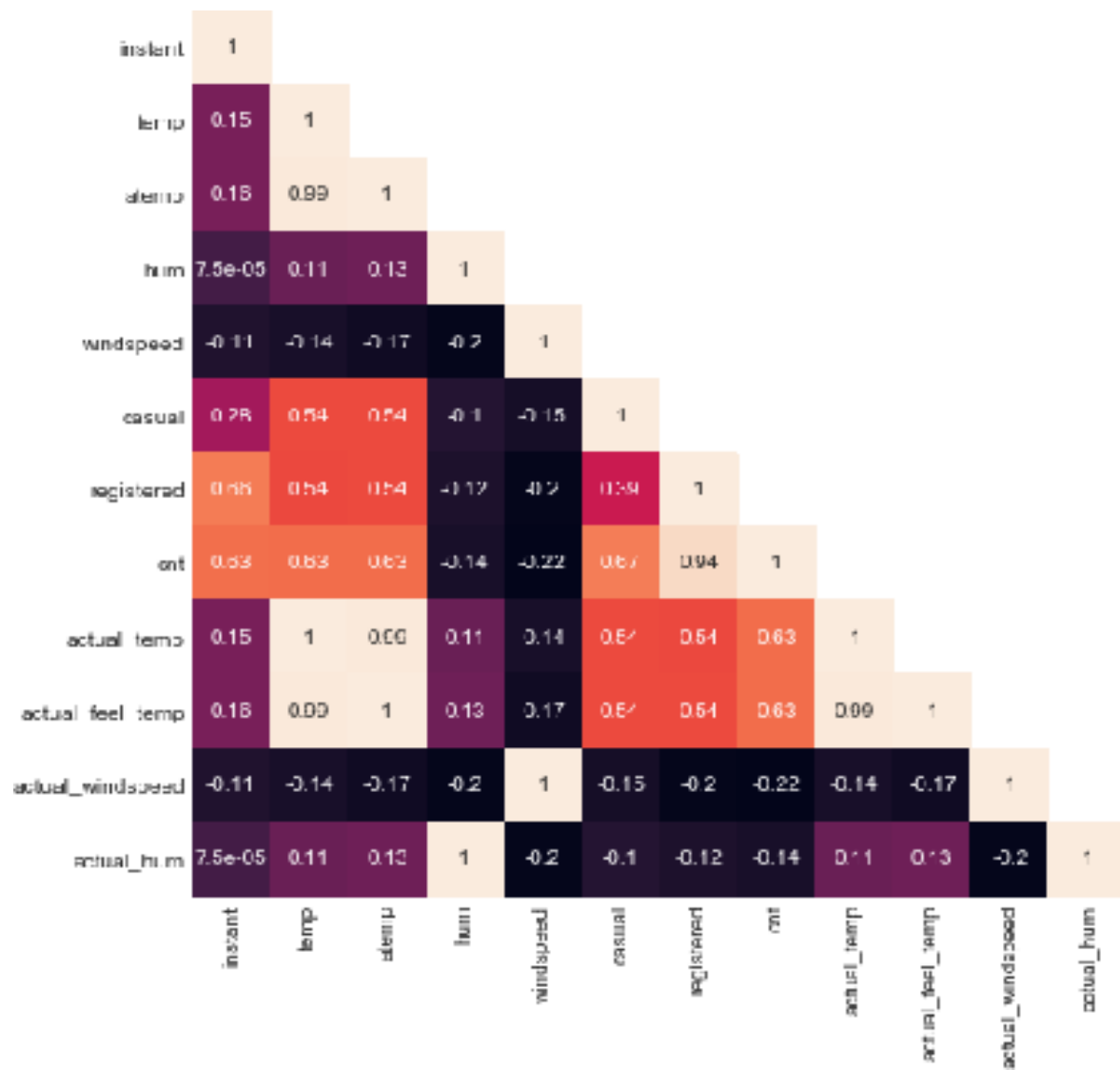**Fig 2.5: Boxplot of continuous variables after removal of outliers**

**Fig 2.7: Correlation plot of all the variables**

# Chapter 6: R code

###########EXPLORE USING GRAPHS############

#CHECK THE DISTRIBUTION OF CATEGORICAL DATA USING BAR GRAPH

```
BAR1 = GGPLOT(DATA = DAY, AES(X = ACTUAL_SEASON)) + GEOM_BAR() + GGTITLE("COUNT OF
SEASON")
BAR2 = GGPLOT(DATA = DAY, AES(X = ACTUAL_WEATHERSIT)) + GEOM_BAR() + GGTITLE("COUNT OF
WEATHER")
BAR3 = GGPLOT(DATA = DAY, AES(X = ACTUAL_HOLIDAY)) + GEOM_BAR() +
GGTITLE("COUNT OF HOLIDAY") BAR4 = GGPLOT(DATA = DAY, AES(X = WORKINGDAY)) +
GEOM_BAR() + GGTITLE("COUNT OF WORKING DAY")
GRIDEXTRA::GRID.ARRANGE(BAR1,BAR2,BAR3,BAR4,NCOL
```

=2) #CHECK THE DISTRIBUTION OF NUMERICAL DATA

USING HISTOGRAM

```
HIST1 = GGPLOT(DATA = DAY, AES(X =ACTUAL_TEMP)) + GGTITLE("DISTRIBUTION OF TEMPERATURE") +
GEOM_HISTOGRAM(BINS = 25)
HIST2 = GGPLOT(DATA = DAY, AES(X =ACTUAL_HUM)) + GGTITLE("DISTRIBUTION OF HUMIDITY") +
GEOM_HISTOGRAM(BINS = 25)
HIST3 = GGPLOT(DATA = DAY, AES(X =ACTUAL_FEEL_TEMP)) + GGTITLE("DISTRIBUTION OF
FEEL TEMPERATURE") + GEOM_HISTOGRAM(BINS = 25)
HIST4 = GGPLOT(DATA = DAY, AES(X =ACTUAL_WINDSPEED)) + GGTITLE("DISTRIBUTION OF WINDSPEED")
+
GEOM_HISTOGRAM(BINS = 25)
GRIDEXTRA::GRID.ARRANGE(HIST1,HIST2,HIST3,HIST4,NCO
```

L=2) #CHECK THE DISTRIBUTION OF NUMERICAL DATA

USING SCATTERPLOT

```
SCAT1 = GGPLOT(DATA = DAY, AES(X =ACTUAL_TEMP, Y = CNT)) + GGTITLE("DISTRIBUTION OF
TEMPERATURE") + GEOM_POINT() + XLAB("TEMPERATURE") + YLAB("BIKE COUNT")
SCAT2 = GGPLOT(DATA = DAY, AES(X =ACTUAL_HUM, Y = CNT)) + GGTITLE("DISTRIBUTION OF HUMIDITY")
+
GEOM_POINT(COLOR="RED") + XLAB("HUMIDITY") + YLAB("BIKE COUNT")
SCAT3 = GGPLOT(DATA = DAY, AES(X =ACTUAL_FEEL_TEMP, Y = CNT)) +
GGTITLE("DISTRIBUTION OF FEEL TEMPERATURE") + GEOM_POINT() + XLAB("FEEL
TEMPERATURE") + YLAB("BIKE COUNT")
SCAT4 = GGPLOT(DATA = DAY, AES(X =ACTUAL_WINDSPEED, Y = CNT)) + GGTITLE("DISTRIBUTION OF
WINDSPEED") + GEOM_POINT(COLOR="RED") + XLAB("WINDSPEED") + YLAB("BIKE COUNT")
GRIDEXTRA::GRID.ARRANGE(SCAT1,SCAT2,SCAT3,SCAT4,N
```

COL=2) #CHECK FOR OUTLIERS IN DATA USING BOXPLOT

```
CNAMES =
COLNAMES(DAY[,C("ACTUAL_TEMP","ACTUAL_FEEL_TEMP","ACTUAL_WINDSPEED","ACTUA
```

```r
L_HUM")]) for (i in 1:length(cnames))
{
assign(paste0("gn",i), ggplot(aes_string(y = cnames[i]), data = day)+
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]) + ggtitle(paste("Box plot
for",cnames[i])))
}
gridExtra::grid.arrange(gn1,gn3,gn2,gn4,ncol=2)
```

#Remove outliers in Windspeed

```r
val = day[,19][day[,19] %in% boxplot.stats(day[,
19])$out] day = day[which(!day[,19] %in% val),]
```

#Check for multicollinearity using VIF

```r
df =
day[,c("instant","temp","atemp","hum","windspeed")]
vifcor(df)
```

#Check for collinearity using corelation graph

```r
corrgram(day, order = F, upper.panel=panel.pie, text.panel=panel.txt, main =
"Correlation Plot")
```

#Remove the unwanted

```r
variables day <- subset(day,

select = -
c(instant,dteday,atemp,casual,registered,actual_temp,actual_feel_temp,actual_windspeed,a
c tual_hum,actual_season,actual_yr,actual_holiday,actual_weathersit))
```

#########DECISION TREE#######################

 #Divide the data into train and test

```r
set.seed(123)
train_index = sample(1:nrow(day), 0.8 *
nrow(day)) train = day[train_index,]
test = day[-train_index,]
```

#rpart for regression

```r
dt_model = rpart(cnt ~ ., data = train, method =

"anova") #Predict the test cases

dt_predictions = predict(dt_model,

test[,-11]) #Create dataframe for actual and
```

PREDICTED VALUES

```r
DF = DATA.FRAME("ACTUAL"=TEST[,11], "PRED"=DT_PREDICTIONS)
HEAD(DF)
```

#CALCULATE MAPE

```r
REGR.EVAL(TRUES = TEST[,11], PREDS = DT_PREDICTIONS, STATS = C("MAE","MSE","RMSE","MAPE"))
```

##########RANDOM FOREST################

#TRAIN THE DATA USING RANDOM FOREST

```r
RF_MODEL = RANDOMFOREST(CNT~., DATA = TRAIN,

NTREE = 500) #PREDICT THE TEST CASES

RF_PREDICTIONS = PREDICT(RF_MODEL,

TEST[,-11]) #CREATE DATAFRAME FOR ACTUAL AND
```

PREDICTED VALUES

```r
DF = CBIND(DF,RF_PREDICTIONS)
HEAD(DF)
```

#CALCULATE MAPE

```r
REGR.EVAL(TRUES = TEST[,11], PREDS = RF_PREDICTIONS, STATS = C("MAE","MSE","RMSE","MAPE"))
```

###########LINEAR REGRESSION###########

#TRAIN THE DATA USING LINEAR REGRESSION

```r
LR_MODEL = LM(FORMULA = CNT~., DATA =

TRAIN) #CHECK THE SUMMARY OF THE MODEL

SUMMARY(LR_MODEL)
```

#PREDICT THE TEST CASES

```r
LR_PREDICTIONS = PREDICT(LR_MODEL,

TEST[,-11]) #CREATE DATAFRAME FOR ACTUAL AND
```

PREDICTED VALUES

```r
DF = CBIND(DF,LR_PREDICTIONS)
HEAD(DF)
```

#CALCULATE MAPE

```r
REGR.EVAL(TRUES = TEST[,11], PREDS = LR_PREDICTIONS, STATS =
```

```r
c("MAE","MSE","RMSE","MAPE")) #Predict a sample data
predict(lr_model,test[2,])
```