Business Analytics Project- Forest Fire Data

Harsh Yadav (hy1217), Nitish Dabas(nd1292), Kush Shah (ks4437)

11/22/2017

Introduction

The data is about the forest fire that occured in a region. It involves 517 incidents of occurance of fire and records values of several factors on which the fire depends. Our task is to perform exploratory data analysis to gain insights about the data and observe the dependence of these variables on each other and how they supplement the occurance of forest fire.

Then, we have performed predictive analysis using the Tree and Random Forest algorithms to predict the area (in hectares) damaged because of the forest fire. The random forest model has been trained by dividing our target variable 'area' into several categories as described in the report below.

1. X - x-axis spatial coordinate within the Montesinho park map: 1 to 9
2. Y - y-axis spatial coordinate within the Montesinho park map: 2 to 9
3. month - month of the year: "jan" to "dec"
4. day - day of the week: "mon" to "sun"
5. FFMC - FFMC index from the FWI system: 18.7 to 96.20
6. DMC - DMC index from the FWI system: 1.1 to 291.3
7. DC - DC index from the FWI system: 7.9 to 860.6
8. ISI - ISI index from the FWI system: 0.0 to 56.10
9. temp - temperature in Celsius degrees: 2.2 to 33.30
10. RH - relative humidity in %: 15.0 to 100
11. wind - wind speed in km/h: 0.40 to 9.40
12. rain - outside rain in mm/m2 : 0.0 to 6.4
13. area - the burned area of the forest (in hectares): 0.00 to 1090.84

```r
fire <- read.csv("C:/Users/Harsh Yadav/Downloads/forestfires.csv", header=TRUE, stringsAsFactors=FALSE)
str(fire)
```

```
## 'data.frame':    517 obs. of  13 variables:
##  $ X    : int  7 7 7 8 8 8 8 8 8 7 ...
##  $ Y    : int  5 4 4 6 6 6 6 6 6 5 ...
##  $ month: chr  "mar" "oct" "oct" "mar" ...
##  $ day  : chr  "fri" "tue" "sat" "fri" ...
##  $ FFMC : num  86.2 90.6 90.6 91.7 89.3 92.3 92.3 91.5 91 92.5 ...
##  $ DMC  : num  26.2 35.4 43.7 33.3 51.3 ...
##  $ DC   : num  94.3 669.1 686.9 77.5 102.2 ...
##  $ ISI  : num  5.1 6.7 6.7 9 9.6 14.7 8.5 10.7 7 7.1 ...
##  $ temp : num  8.2 18 14.6 8.3 11.4 22.2 24.1 8 13.1 22.8 ...
##  $ RH   : int  51 33 33 97 99 29 27 86 63 40 ...
##  $ wind : num  6.7 0.9 1.3 4 1.8 5.4 3.1 2.2 5.4 4 ...
##  $ rain : num  0 0 0 0.2 0 0 0 0 0 0 ...
##  $ area : num  0 0 0 0 0 0 0 0 0 0 ...
```

The data has been stored in a dataframe named fire and the str function shows that there are 517 observations and 13 variables in our dataset. The description of each column is as follows:

```
sum(is.na(fire))
```

```
## [1] 0
```

```
nacols <- function(fire) {
    colnames(fire)[unlist(lapply(fire, function(x) any(is.na(x))))]
}
```

There are no NA values in our dataset.

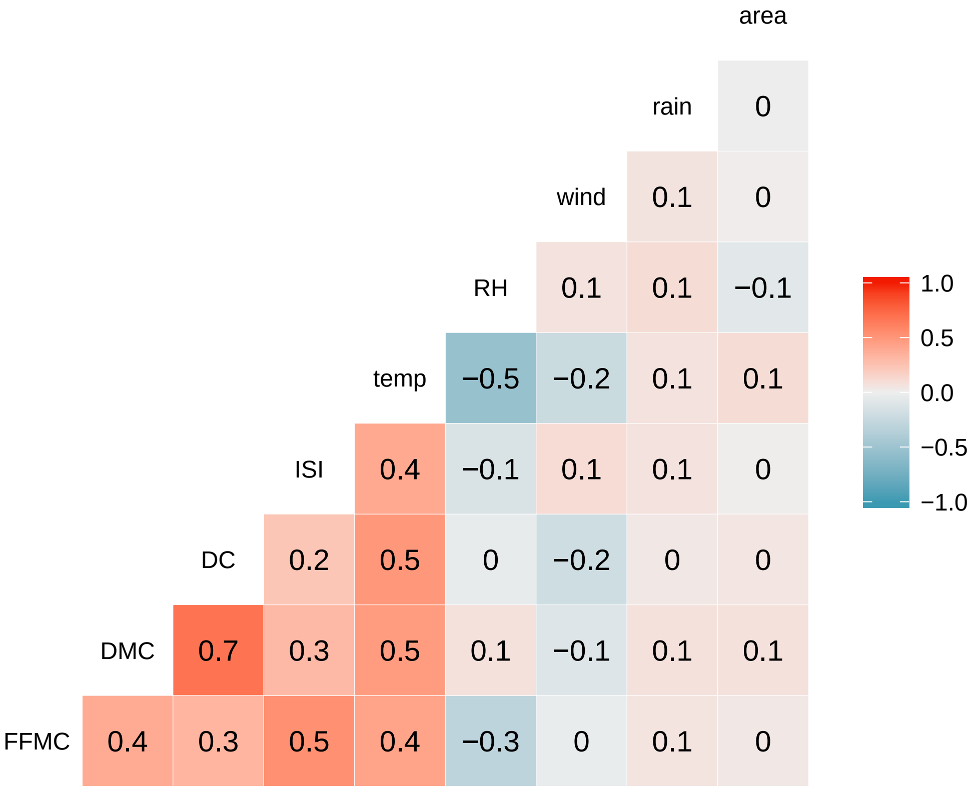
```
summary(fire)
```

```
##        X               Y           month               day           
##  Min.   :1.000   Min.   :2.0   Length:517         Length:517        
##  1st Qu.:3.000   1st Qu.:4.0   Class :character   Class :character  
##  Median :4.000   Median :4.0   Mode  :character   Mode  :character  
##  Mean   :4.669   Mean   :4.3                                        
##  3rd Qu.:7.000   3rd Qu.:5.0                                        
##  Max.   :9.000   Max.   :9.0                                        
##       FFMC            DMC              DC             ISI        
##  Min.   :18.70   Min.   :  1.1   Min.   :  7.9   Min.   : 0.000  
##  1st Qu.:90.20   1st Qu.: 68.6   1st Qu.:437.7   1st Qu.: 6.500  
##  Median :91.60   Median :108.3   Median :664.2   Median : 8.400  
##  Mean   :90.64   Mean   :110.9   Mean   :547.9   Mean   : 9.022  
##  3rd Qu.:92.90   3rd Qu.:142.4   3rd Qu.:713.9   3rd Qu.:10.800  
##  Max.   :96.20   Max.   :291.3   Max.   :860.6   Max.   :56.100  
##       temp             RH              wind            rain        
##  Min.   : 2.20   Min.   : 15.00   Min.   :0.400   Min.   :0.00000  
##  1st Qu.:15.50   1st Qu.: 33.00   1st Qu.:2.700   1st Qu.:0.00000  
##  Median :19.30   Median : 42.00   Median :4.000   Median :0.00000  
##  Mean   :18.89   Mean   : 44.29   Mean   :4.018   Mean   :0.02166  
##  3rd Qu.:22.80   3rd Qu.: 53.00   3rd Qu.:4.900   3rd Qu.:0.00000  
##  Max.   :33.30   Max.   :100.00   Max.   :9.400   Max.   :6.40000  
##       area        
##  Min.   :   0.00  
##  1st Qu.:   0.00  
##  Median :   0.52  
##  Mean   :  12.85  
##  3rd Qu.:   6.57  
##  Max.   :1090.84  
```

The above function represents the summary of each of the feature of our dataset representing their min, max, quartile values.

```
library(GGally)
ggcorr(fire[,3:13],label_size = 4, size = 3, palette = "RdBu", label = TRUE)
```

```
## Warning in ggcorr(fire[, 3:13], label_size = 4, size = 3, palette =
## "RdBu", : data in column(s) 'month', 'day' are not numeric and were ignored
```



The correlation represented above shows the relationship between different feartures of our dataset. We can see that DC and DMC values are most strongly positively correlated to each other and temperature is strongly negatively correlated to RH value.

Fine Fuel Moisture Code - FFMC

This is a numerical rating of the moisture content of surface litter and other cured fine fuels. It shows the relative ease of ignition and flammability of fine fuels. The moisture content of fine fuels is very sensitive to the weather. Even a day of rain, or of fine and windy weather, will significantly affect the FFMC rating. The system uses a time lag of two-thirds of a day to accurately measure the moisture content in fine fuels. The FFMC rating is on a scale of 0 to 99. Any figure above 70 is high, and above 90 is extreme.

```
range(fire$FFMC)
```

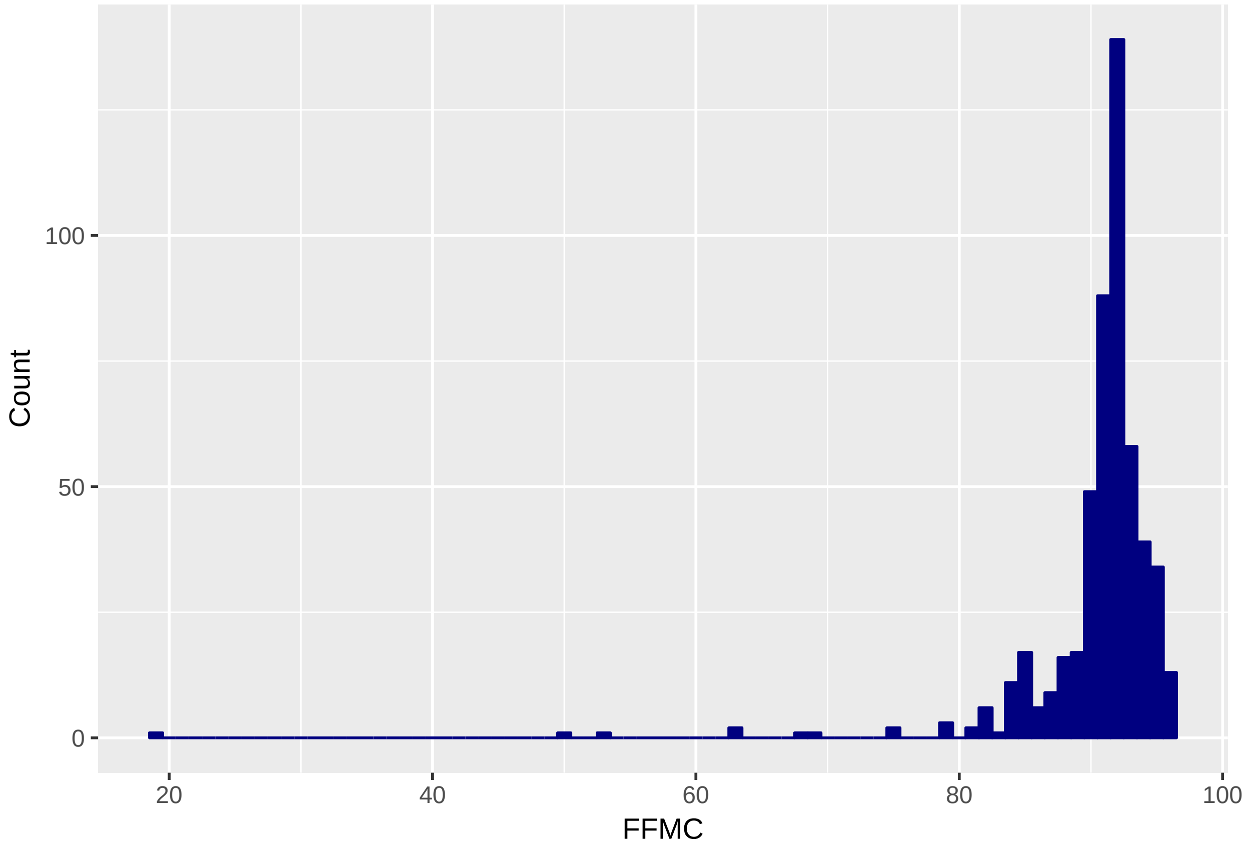
```
## [1] 18.7 96.2
```

Here we can see that our dataset has values of FFMC between 18.7 to 96.2 which is between the range of 0 to 99 as defined above for the FFMC index.

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.2
```

```
ggplot(fire, aes(x=fire$FFMC)) +
    geom_histogram(binwidth=1, colour="navy", fill="navy")  +
    labs(x="FFMC",y="Count")
```



```
median(fire$FFMC)
```

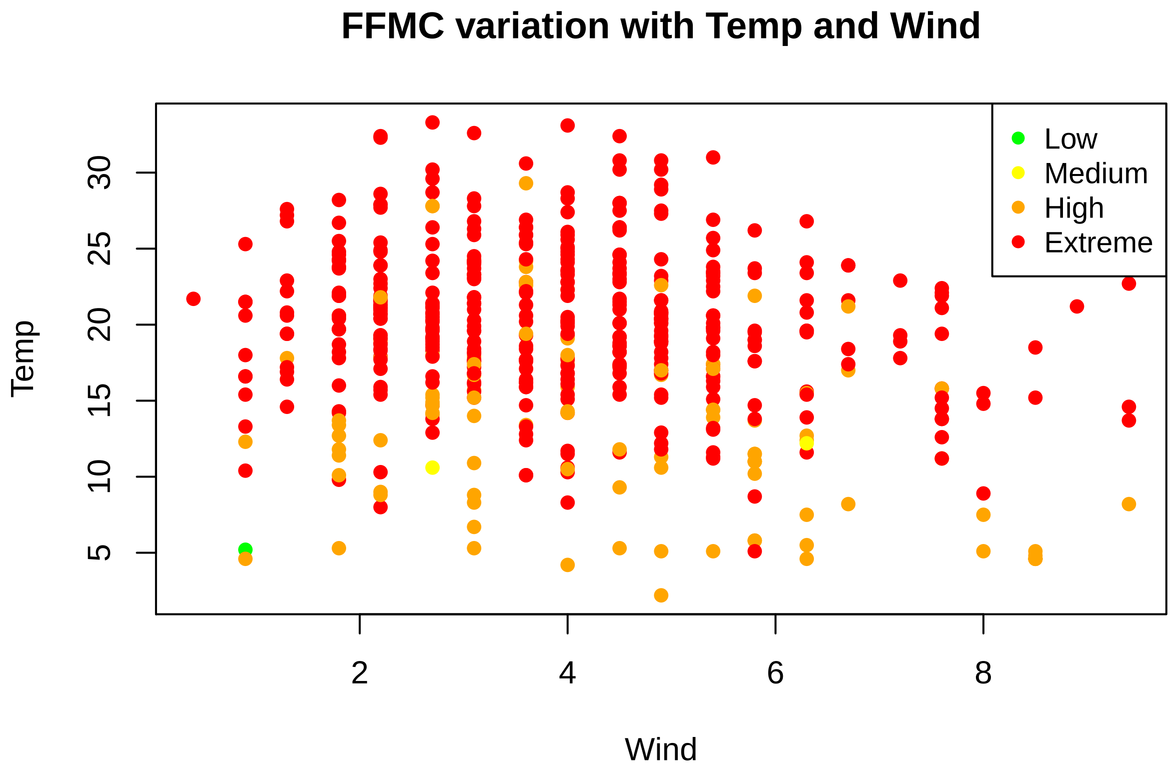
```
## [1] 91.6
```

The first table shows that the FFMC index has values between

This histogram shows that most of the data in our dataset involves the FFMC index that is above 70 (fire usually begins from this point) which comes in the high category and thus is quite dangerous for the fire to occur. The median value is 91.6. Also, we can see that a lot of values are near 90 and above(extreme), which indicates that there are high chances of forest fire to occur.

FFMC is affected by temperature, rain, wind and relative humidity since it represents the top-most layer of fuels.

```
fire$FFMC_CAT <- cut(fire$FFMC,
                      breaks = c(-Inf, 30, 60, 90, Inf),
                      labels = c("Low", "Medium", "High", "Extreme"),
                      right = FALSE)

mycolors = c('green','yellow','orange','red')
plot(fire$wind, fire$temp, pch = 16, col = mycolors[fire$FFMC_CAT],
     main="FFMC variation with Temp and Wind",xlab="Wind", ylab="Temp")
legend("topright", legend=levels(fire$FFMC_CAT), pch=16, cex=0.9, col=unique(mycolors))
```



The above scatter plot shows how FFMC index varies with variations in wind and temperature values. FFMC index caregories are mentioned in the legend of the graph. It can be observed that both of them directly impacts the FFMC index but if the temp is above 15 and the wind range is above 8 then there are high to extreme chances of fire.

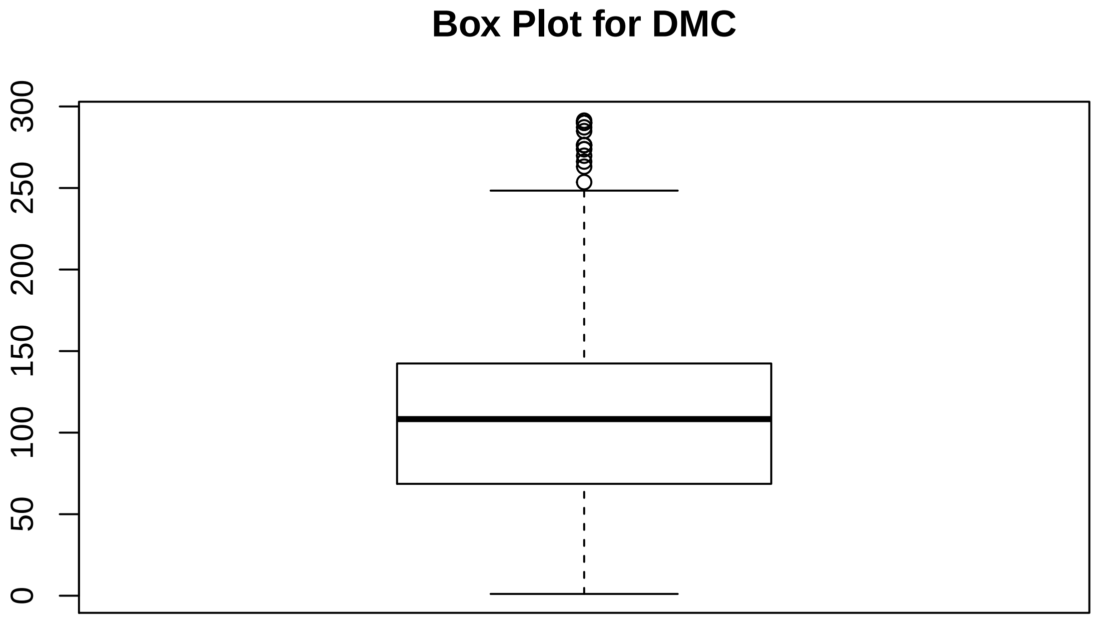
Duff Moisture Code - DMC

DMC is a numerical rating of the average moisture content of loosely compacted organic layers of moderate depth. The code indicates the depth that fire will burn in moderate duff layers and medium size woody material. Duff layers take longer than surface fuels to dry out but weather conditions over the past couple of weeks will significantly affect the DMC. The system applies a time lag of 12 days to calculate the DMC. A DMC rating of more than 30 is dry, and above 40 indicates that intensive burning will occur in the duff and medium fuels. Burning off operations should not be carried out when the DMC rating is above 40.

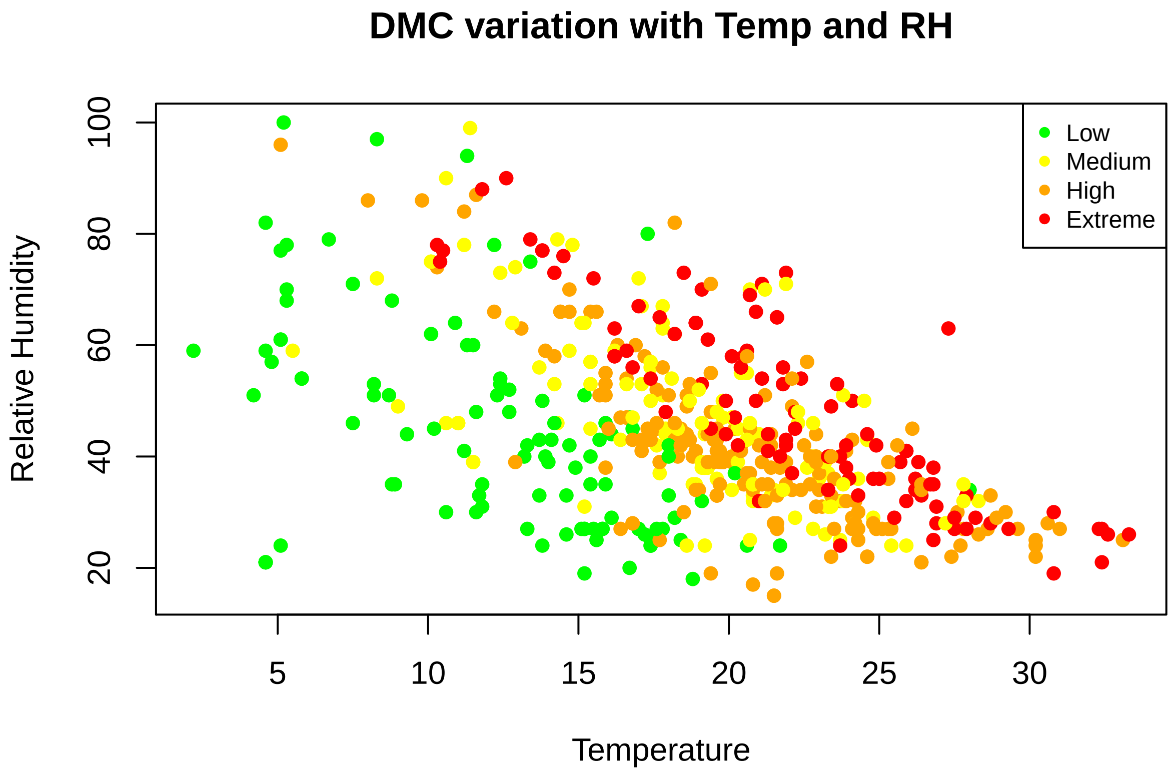
```r
median(fire$DMC)
```

```
## [1] 108.3
```

```r
boxplot(fire$DMC, main="Box Plot for DMC")
```



```
fire$DMC_CAT <- cut(fire$DMC,
                       breaks = c(-Inf, 50, 100, 150, Inf),
                       labels = c("Low", "Medium", "High", "Extreme"),
                       right = FALSE)

mycolors = c('green','yellow','orange','red')
plot(fire$temp, fire$RH, pch = 16, col = mycolors[fire$DMC_CAT],
     main="DMC variation with Temp and RH",xlab="Temperature", ylab="Relative Humidity")
legend("topright", legend=levels(fire$DMC_CAT), pch=16, cex=0.75, col=unique(mycolors))
```



The Boxplot for DMC index indicates that most of the values lie between 60 to 150 which are above 40 and thus, we can infer that there is very high chance of fire occuring due to such high values of DMC as described in the description above.

DMC is affected by rain, temperature and relative humidity since it represents the layer which is 2-4 inches deep where wind will have no effect. The second plot represents the variation of DMC index with Temperature and Relative Humidity and we can observe that for high temperature and lower values of Temperature the DMC index values are in the extreme range.

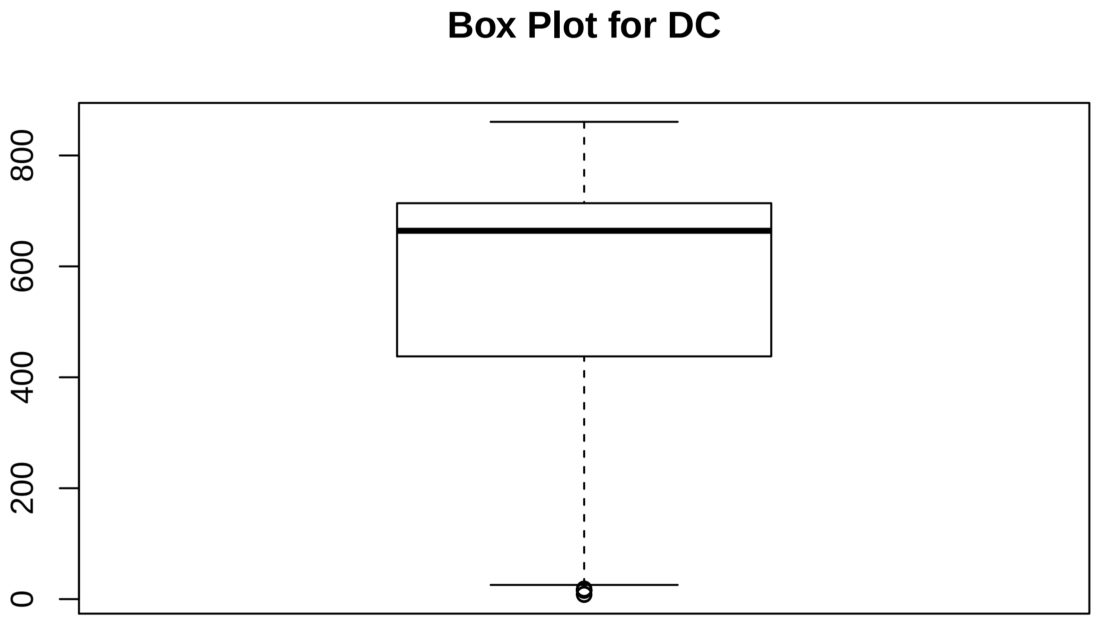
Drought Code - DC

The DC is a numerical rating of the moisture content of deep, compact, organic layers. It is a useful indicator of seasonal drought and shows the likelihood of fire involving the deep duff layers and large logs. A long period of dry weather (the system uses 52 days) is needed to dry out these fuels and affect the Drought Code. A DC rating of 200 is high, and 300 or more is extreme indicating that fire will involve deep sub-surface and heavy fuels. Burning off should not be permitted when the DC rating is above 300.

```r
median(fire$DC)
```

```
## [1] 664.2
```

```r
library(ggplot2)
boxplot(fire$DC, main="Box Plot for DC")
```

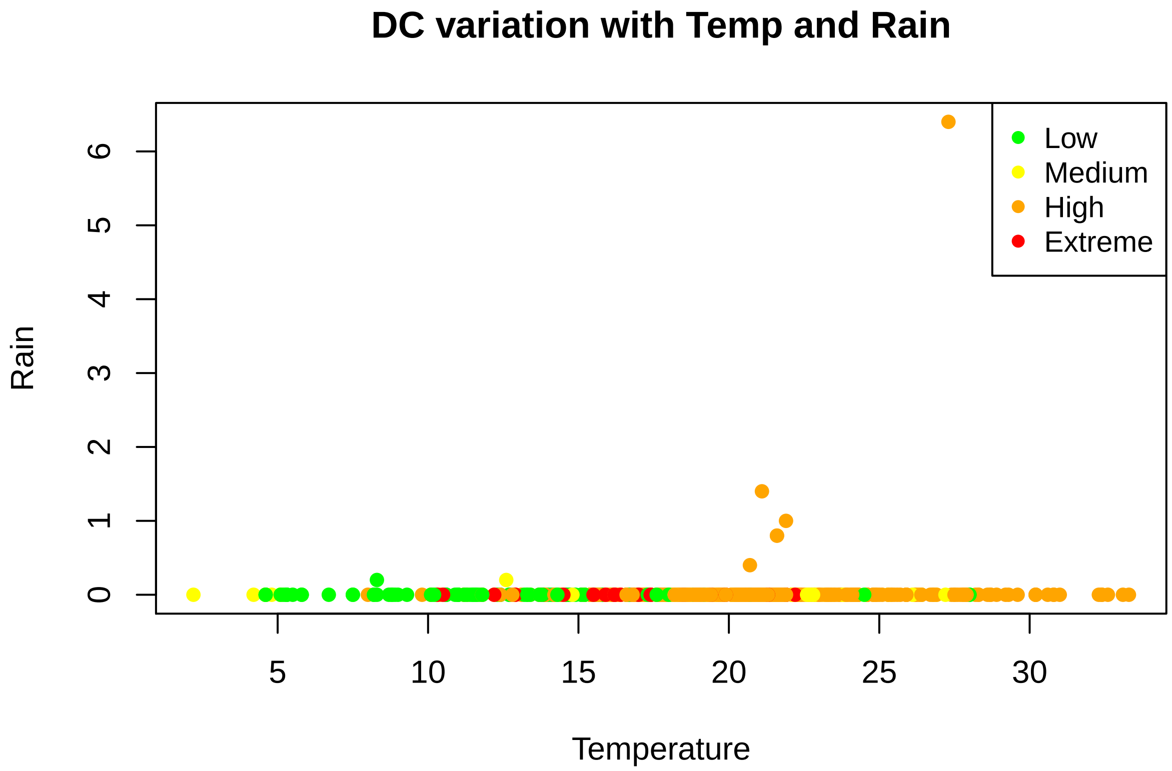


The above boxplot for the DC index shows that most of the values lie between 400 to 700 which are extreme conditions for fire to occur and in these situations burning off is not allowed as they may facilitate forest fires. Further, there exists outliers which are shown in the boxplot with a circular shape.

DC is affected only by rain and temperature because of its depth. Although we need a 24-hr rainfall of greater than 0.11 inches to consider it for DC calculations.

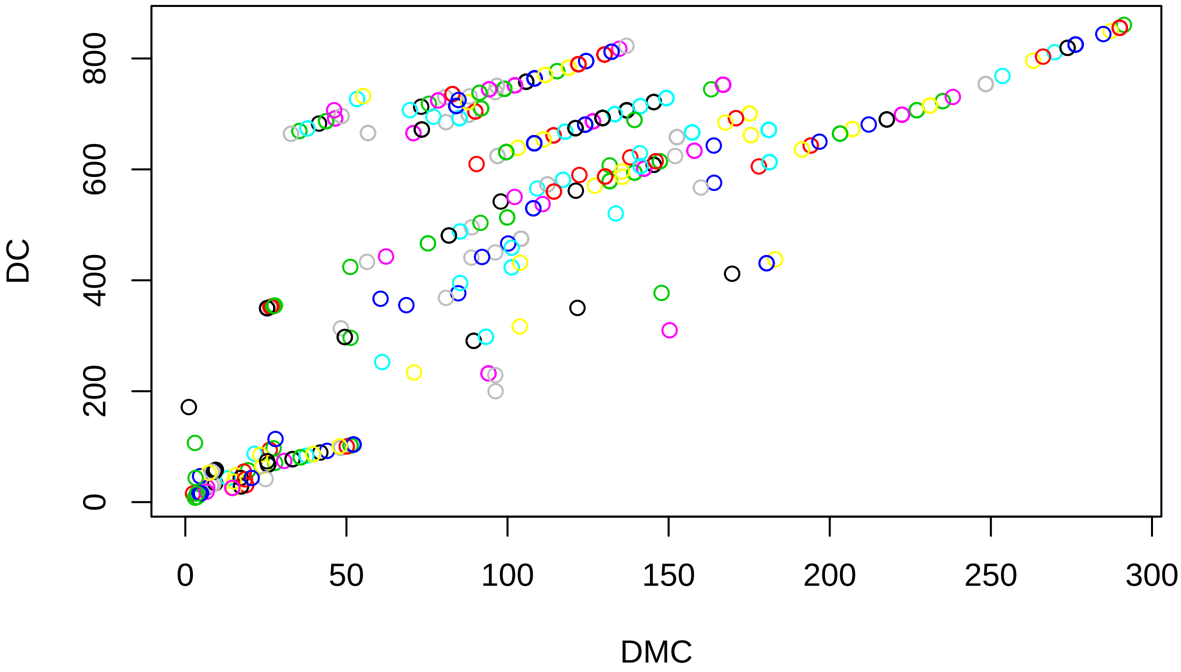
From the histogram we can view that most of the values of DC index are very high and thus there are extreme chnces for the fire to occur.

```
fire$DC_CAT <- cut(fire$DC,
                       breaks = c(-Inf, 300, 500, 800, Inf),
                       labels = c("Low", "Medium", "High", "Extreme"),
                       right = FALSE)

mycolors = c('green','yellow','orange','red')
plot(fire$temp, fire$rain, pch = 16, col = mycolors[fire$DC_CAT],
     main="DC variation with Temp and Rain",xlab="Temperature", ylab="Rain")
legend("topright", legend=levels(fire$DC_CAT), pch=16, cex=0.9, col=unique(mycolors))
```



This plot represents the variation of DC with Temperature and Rain, but sinse most of the values in the rain column in our dataset are 0, so we are able to make very less observations from this graph. We can observe that for high temperatures and no rain the value of DC index increases and goes upto the high and extreme values as the temperature increases.

```
plot(DC ~ DMC, fire, col=fire$DMC)
```



From the above scatter plot we can see that for high values of DC index the value of the DMC index is also high. As both DC and DMC index depend on moisture content of organic layers, therefore this kind of relationship exists betwen them where if one increases then other also increases. For higher values of DMC and DC index there are more extreme chances of fire to occur.

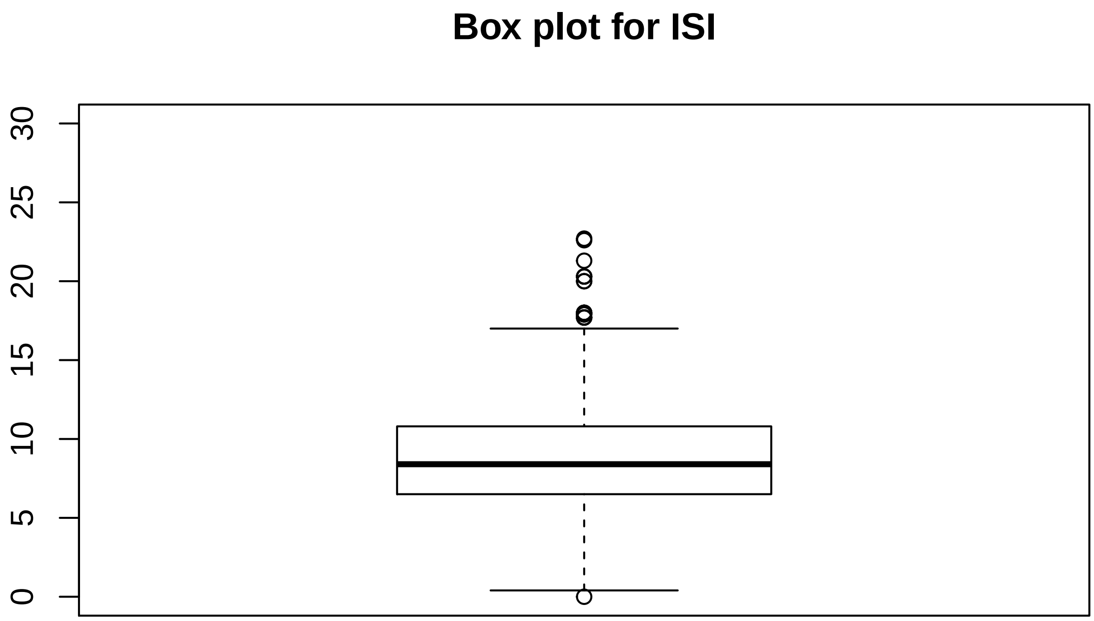
Initial Spread Index - ISI

This indicates the rate fire will spread in its early stages. It is calculated from the FFMC rating and the wind factor. The open-ended ISI scale starts at zero and a rating of 10 indicates high rate of spread shortly after ignition. A rating of 16 or more indicates extremely rapid rate of spread.

```
mean(fire$ISI)
```

```
## [1] 9.021663
```

```r
boxplot(fire$ISI, main="Box plot for ISI", ylim=c(0,30))
```



From the boxplot of the ISI index we can see that most of the values of this index are from 5 to 10, where 10 indicates high rate of spread shortly after ignition. Also, there are values above 10, which indicate severe spread of fire.

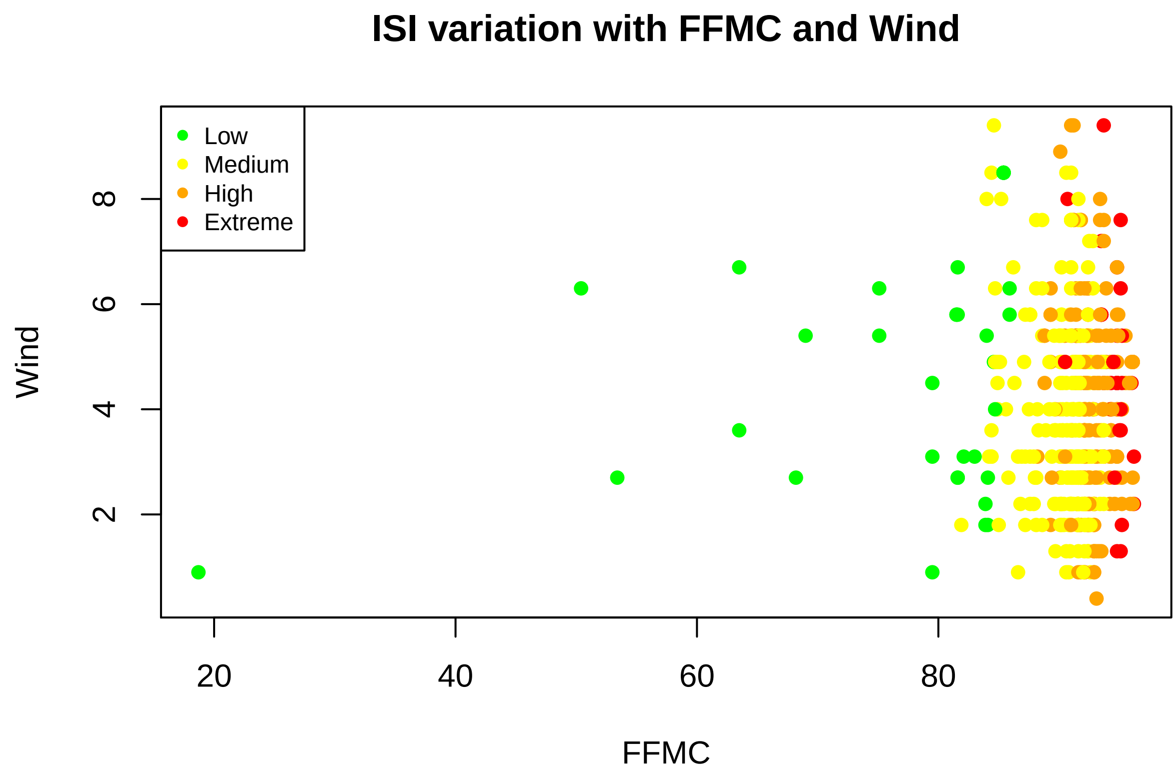
```r
## Calculated the correlation to get an idea about how my graph will show up.
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following object is masked from 'package:GGally':
##
##     nasa

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(tidyr)
```

```
## Warning: package 'tidyr' was built under R version 3.4.2
```

```
fire$ISI_Cat <- cut(fire$ISI,
                    breaks = c(-Inf, 3, 9, 16, Inf),
                    labels = c("Low", "Medium", "High", "Extreme"),
                    right = FALSE)

mycolors = c('green','yellow','orange','red')
plot(fire$FFMC, fire$wind, pch = 16, col = mycolors[fire$ISI_Cat],
     main="ISI variation with FFMC and Wind",xlab="FFMC", ylab="Wind")
legend("topleft", legend=levels(fire$ISI_Cat), pch=16,cex=0.75, col=unique(mycolors))
```



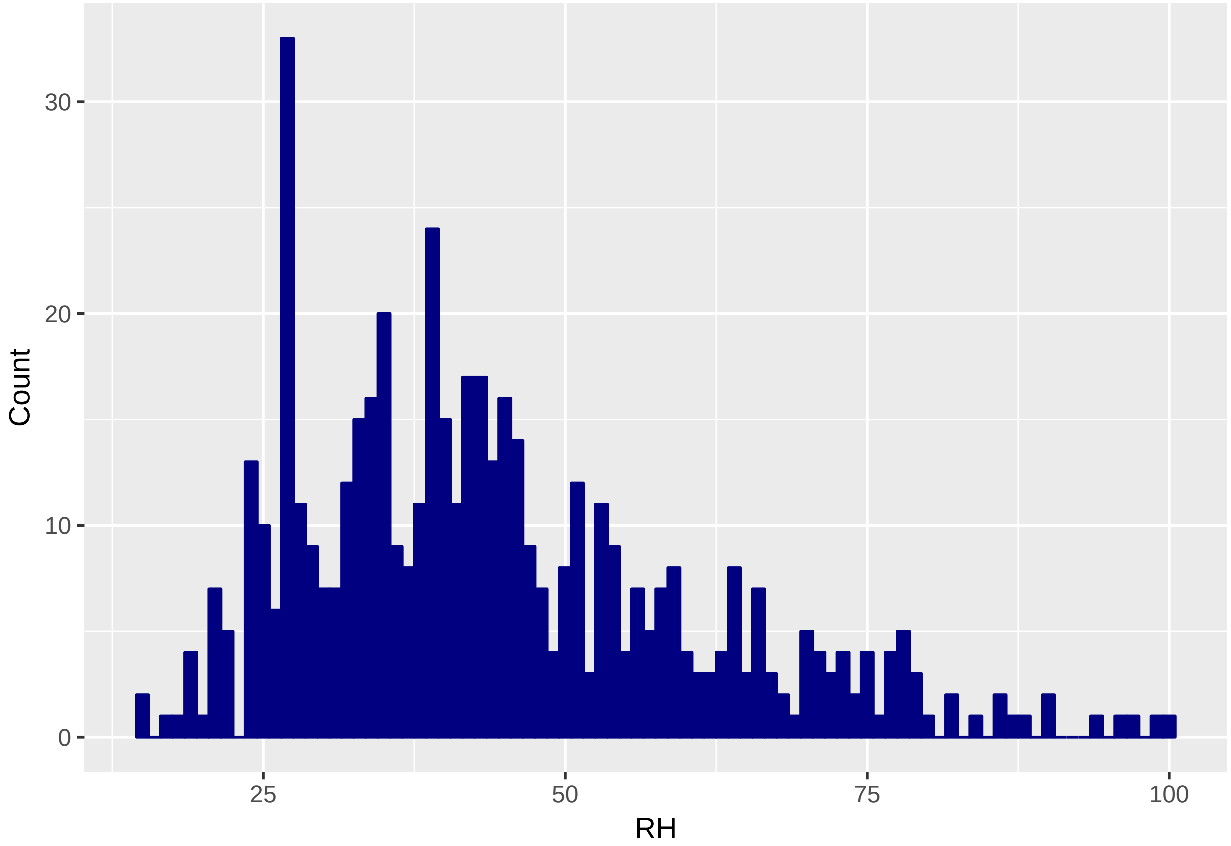
From the above graph we infer that for higher values of FFMC index the values of ISI increase. Thus, for extreme values of FFMC we obtain extreme values of ISI index which show that there are high chances of fire and also there are high chances that the fire will spread rapidly.

Further, it can be seen in the graph that for the given dataset, wind is not contributing much to ISI index since we can see that even at lower values of wind index we have higher values of ISI, it can be corroborated by correlation calculation between these factors.

Relative humidity - RH

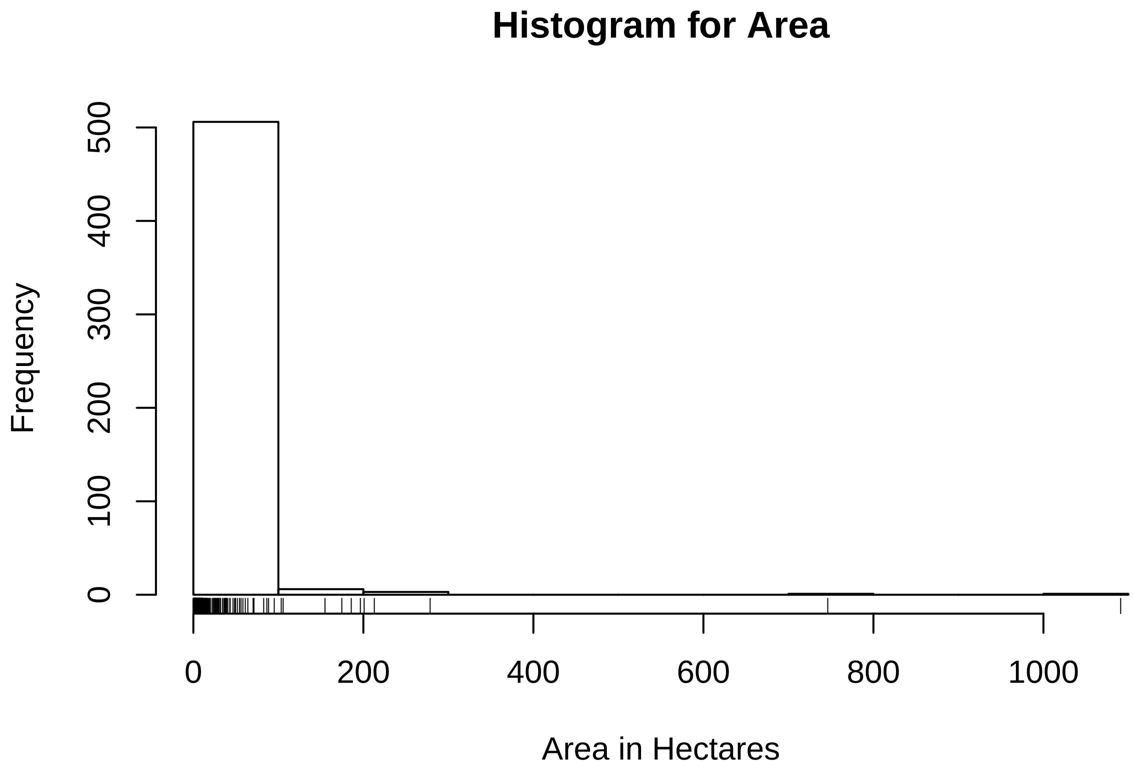
Relative humidity (RH) is the ratio of the partial pressure of water vapor to the equilibrium vapor pressure of water at a given temperature. Relative humidity depends on temperature and the pressure of the system of interest. It requires less water vapor to attain high relative humidity at low temperatures; more water vapor is required to attain high relative humidity in warm or hot air.

```r
#boxplot(fire$RH)

ggplot(fire, aes(x=fire$RH)) +
    geom_histogram(binwidth=1, colour="navy", fill="navy")  +
    labs(x="RH",y="Count")
```



This plot represents that most of the Relative Humidity values lie between 25 to 50, which are ideal RH values. The values near 100 represent very high humidity level.

```
hist(fire$area, main="Histogram for Area", xlab="Area in Hectares")
rug(fire$area)
```



The above histogram for the area that was affected by fire represents that most of the fire that occured was very less and did not spread much. But, in some cases we can see that the fire occured covering major areas.

Predictive Analysis

Regression Tree Algorithm

```
fire_t <- read.csv("C:/Users/Harsh Yadav/Downloads/forestfires.csv", header=TRUE, stringsAsFactors=FALS
```

Here we have loaded the dataset from the csv file to a dataframe named fire_t, which we are going to use for our predictive analysis using regression tree algorithm.

```
library(plyr)
```

```
## ------------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
## --------------------------------------------------------------------------

##
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
fire_t$day <- revalue(fire_t$day, c("sun"=1, "mon"=2, "tue"=3, "wed"=4, "thu"=5, "fri"=6, "sat"=7))
fire_t$month <- revalue(fire_t$month, c("jan"=1, "feb"=2, "mar"=3, "apr"=4, "may"=5, "jun"=6, "jul"=7,

set.seed(415)

data <- sample(2,nrow(fire_t),replace=TRUE,prob=c(0.7,0.3))
trainData <- fire_t[data==1,]
testData <- fire_t[data==2,]
```

Then we have used the plyr library to convert the day column in our dataframe from string to numerical values and converting the month column from string to numerical values from 1 to 12. We have done this in order to avaoid any coersions that introduce NA's in our model.

Further we have set the seed to 415, and then dive=ided our data into training and testing datasets named trainData and testData respectively. We have performed training of our model using the trainData dataset and will be testing it using the testData dataset.

```
library(tree)
```

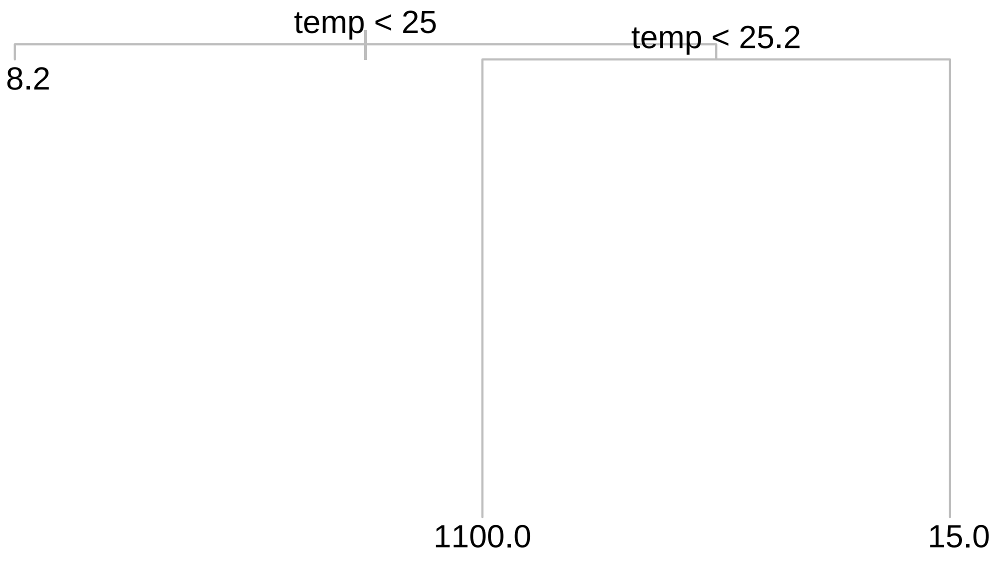
```
## Warning: package 'tree' was built under R version 3.4.2
```

```
pstree <- tree(area ~., data=trainData, mincut=1)
pstree
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
## 1) root 360 1407000   11.970
##   2) temp < 25 317   188200    8.208 *
##   3) temp > 25 43 1182000   39.660
##     6) temp < 25.2 1       0 1091.000 *
##     7) temp > 25.2 42   50480   14.640 *
```

We have now used the tree tree library to train the modlel named pstree which is our regression tree model with a mincut of 1. Then we have shown the traind model. The model shows that the feature named temp is the most important feature. It represents the temperature when the fire has occured or not. From this tree we can esily see that it is not a good model for making predictions of occurance of forest fires as it just depends on one feature which is the temperature and does not depend on other features like FFMC index, humidity, rain, ISI, etc, which also play an important role in the occurance of forest fires. So, this model is not a good prediction-tive model. Apart from this it has high deviance values which will again not help us in predicting good results of our target variable area.

```r
plot(pstree, col=8)
text(pstree, digits=2)
```



Here we have represented the above tree and the vertical lines represent the deviance in our obtained results. The tree represents high deviance and only depends on one feature. So we will not consider it for our further analysis of making prediction of fire areas.

```
library(tree)
pstree <- tree(area ~., data=trainData, mincut=5)
pstree
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
## 1) root 360 1407000  11.970
##   2) temp < 25 317  188200   8.208 *
##   3) temp > 25 43 1182000  39.660
##     6) temp < 25.8 5  948500 219.800 *
##     7) temp > 25.8 38   49730  15.960 *
```

Here we have used a different value of mincut to see if we have any improvements in our results. But, we have obtained higher values of deviance for the value of mincut as 5. So we consider not using this model.

```
set.seed(2)
cvpst <- cv.tree(pstree, K=10)
cvpst$size
```

```
## [1] 3 1
```

```
cvpst$dev
```

```
## [1] 1695060 1557663
```

Now using cross validation to determine the best case for our analysis using the tree algorithm we can see that the least deviance occurs for the value of micut as 1 as we saw above. This is not a good result, so we now plan to use other predictive algorithm to for our predictive analysis.

Random Forest Algorithm

```
fire_rf <- read.csv("C:/Users/Harsh Yadav/Downloads/forestfires.csv", header=TRUE, stringsAsFactors=FALS
```

Here we have again loaded our dataset into a dataset named fire_rf for our predictive modelling using the Random forest algorithm.

```
library(plyr)
fire_rf$area_category <- cut(fire_rf$area, breaks=c(-1, 10, 25, 50, 100, 150, 200, 250, Inf), labels=c(
fire_rf$day <- revalue(fire_rf$day, c("sun"=1, "mon"=2, "tue"=3, "wed"=4, "thu"=5, "fri"=6, "sat"=7))
fire_rf$month <- revalue(fire_rf$month, c("jan"=1, "feb"=2, "mar"=3, "apr"=4, "may"=5, "jun"=6, "jul"=7
```

Then we have used the plyr library to first define categories for our target variable area and have named it as area_category, so that we can use random forest on categorical data. We have divided the variable area into 8 discrete categories that are as follows:

1. Areas from 0 to 10 hectares: Almost negligible fire
2. Areas from 10 to 25 hectares: Small fire
3. Areas from 25 to 50 hectares: Moderate fire
4. Areas from 50 to 100 hectares: Lagre fire
5. Areas from 100 to 150 hectares: Big fire
6. Areas from 150 to 200 hectares: Huge fire
7. Areas from 200 to 250 hectares: Extreme fire
8. Areas above 250 hectares: Uncontrollable fire

We have assigned a number to each category for our predictive analysis from 1 to 8 respectively.

Then we convert the day column in our dataframe from string to numerical values and converting the month column from string to numerical values from 1 to 12. We have done this in order to avaoid any coersions that introduce NA's in our model.

```
set.seed(415)
data <- sample(2,nrow(fire_rf),replace=TRUE,prob=c(0.7,0.3))
trainData <- fire_rf[data==1,]
testData <- fire_rf[data==2,]
```

Further we have set the seed to 415, and then dive=ided our data into training and testing datasets named trainData and testData respectively. We have performed training of our model using the trainData dataset and will be testing it using the testData dataset.

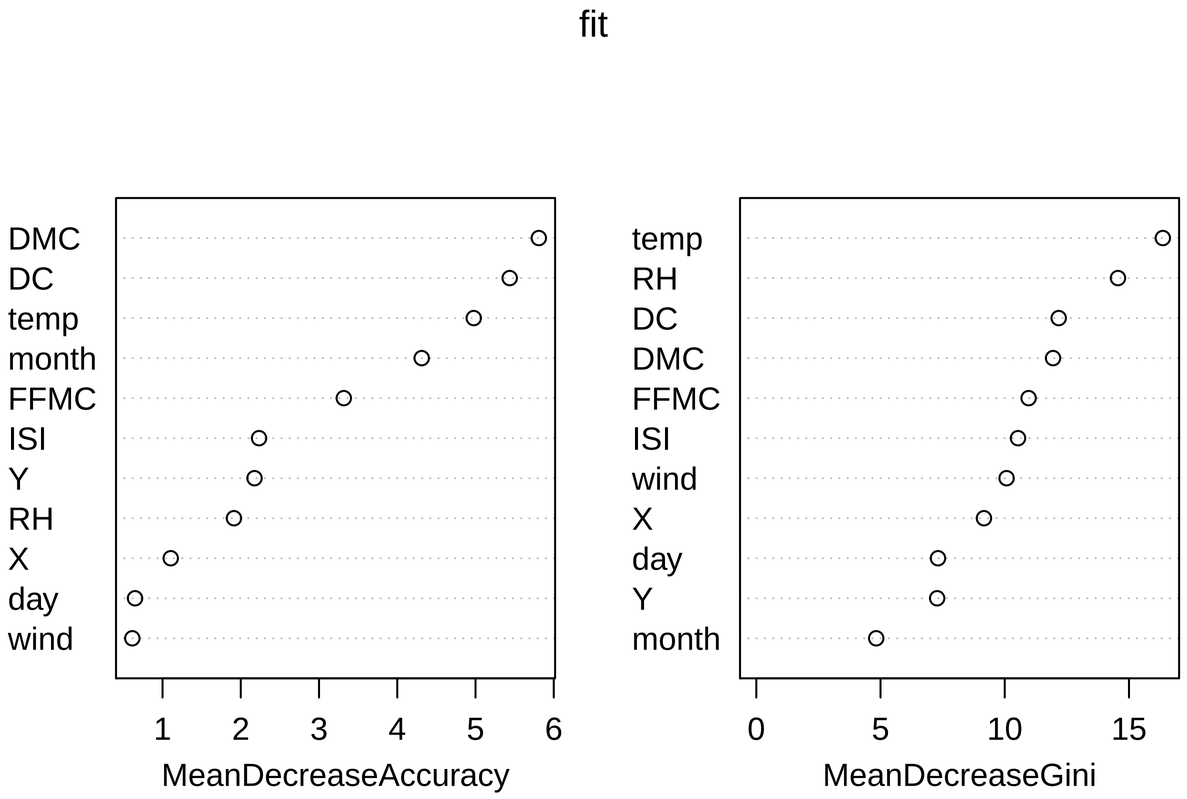
```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 3.4.2

## randomForest 4.6-12

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:dplyr':
##
##     combine

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
fit <- randomForest(area_category ~temp+ISI+DMC+Y+month+FFMC+DC+X+wind+day+RH+wind, data=trainData, imp
```

Here, we have created a model named fit, which will be trained using the Random Forest Algorithm on our categorical target variable named area_category based on all the predictive variables. The data used for training is the trainData dataset that is defined above. We have used 200 trees for our random forest algorithm analysis.

```
varImpPlot(fit)
```


fit
DMC
DC
temp
month
FFMC
ISI
Y
RH
X
day
wind
1
2
3
4
5
6
MeanDecreaseAccuracy
temp
RH
DC
DMC
FFMC
ISI
wind
X
day
Y
month
0
5
10
15
MeanDecreaseGini

Here we can see the plot for our trained Random Forest model named fit. We can see the Mean Decrease Accuracy and Mean Decrease Gini for each of our predictor variables. From the plot we can observe that temperature is the most important feature followed by RH and DC index for computing our Gini value for the model. We can also see the accuracy value depends most on DMC followed by the DC index. Most of the variables are not that important in determining our accuracy for this case, so even if we drop the features like wind, day, X, RH, etc, they they would not affect our accuracy a lot.

The mean decrease in Gini coefficient is a measure of how each variable contributes to the homogeneity of the nodes and leaves in the resulting random forest. Each time a particular variable is used to split a node, the Gini coefficient for the child nodes are calculated and compared to that of the original node. SO we can see that temperature is again most imporant in computing the Gini value.

```
Pred<-predict(fit,newdata=testData)
conf <- table(Pred, testData$area_category)
```

Here we have stored the predicted results obtained from our trained model which we ran on the testData. The predicted values are stored in variable named Pred. Then, we have defined a table named conf that stores the actual vales and the Predicted values in a table, so that we can determone the accuracy and other parameters of our model.

```
library(e1071)
```

```
## Warning: package 'e1071' was built under R version 3.4.2
```

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.4.2
```

```
## Loading required package: lattice
```

```
confusionMatrix(conf)
```

```
## Confusion Matrix and Statistics
##
##
## Pred   1   2   3   4   5   6   7   8
##    1 125  11   7   2   0   3   1   1
##    2   3   1   0   1   0   0   0   0
##    3   1   0   1   0   0   0   0   0
##    4   0   0   0   0   0   0   0   0
##    5   0   0   0   0   0   0   0   0
##    6   0   0   0   0   0   0   0   0
##    7   0   0   0   0   0   0   0   0
##    8   0   0   0   0   0   0   0   0
## 
## Overall Statistics
##                                           
##                Accuracy : 0.8089          
##                  95% CI : (0.7386, 0.8672)
##     No Information Rate : 0.8217          
##     P-Value [Acc > NIR] : 0.7045          
##                                           
##                   Kappa : 0.0982          
##  Mcnemar's Test P-Value : NA              
## 
## Statistics by Class:
## 
##                      Class: 1 Class: 2 Class: 3 Class: 4 Class: 5 Class: 6
## Sensitivity            0.9690 0.083333 0.125000  0.00000       NA  0.00000
## Specificity            0.1071 0.972414 0.993289  1.00000        1  1.00000
## Pos Pred Value         0.8333 0.200000 0.500000      NaN       NA      NaN
## Neg Pred Value         0.4286 0.927632 0.954839  0.98089       NA  0.98089
## Prevalence             0.8217 0.076433 0.050955  0.01911        0  0.01911
## Detection Rate         0.7962 0.006369 0.006369  0.00000        0  0.00000
## Detection Prevalence   0.9554 0.031847 0.012739  0.00000        0  0.00000
## Balanced Accuracy      0.5381 0.527874 0.559144  0.50000       NA  0.50000
##                      Class: 7 Class: 8
## Sensitivity          0.000000 0.000000
## Specificity          1.000000 1.000000
## Pos Pred Value            NaN      NaN
## Neg Pred Value       0.993631 0.993631
## Prevalence           0.006369 0.006369
## Detection Rate       0.000000 0.000000
## Detection Prevalence 0.000000 0.000000
## Balanced Accuracy    0.500000 0.500000
```

Here are the results that have been obtained from our random forest algorithm. We have used the confusion matrix to obtain the results. We can observe that we have obtained an accuracy of 80.89 %, so we are quite accurately able to predict the extent of fire that would occur given the data of rain, temperature, FFMC, coordinates (X and Y), ISI, humidity, etc.

Additionally we can see the calsses to which the predicted results belong to. So, we can observe that maximum results belong to the 1st class, that predicted almost negligible fire. Further, the sensitivity, specificity and other parameters for all the classes have been shown. We have taken 95% confidence interval for our analysis and most of our results are within it.

The no information error rate is the error rate when the input and output are independent. So, for our case the value id very low, which indicates good modelling of our data.

The p-value tells us the probability of the under the null hypothesis. A small p-value indicates strong evidence against the null hypothesis, so we reject the null hypothesis.

The Kappa value is a metric that compares an Observed Accuracy with an Expected Accuracy (random chance). The kappa statistic is used not only to evaluate a single classifier, but also to evaluate classifiers amongst themselves. In essence, the kappa statistic is a measure of how closely the instances classified by the machine learning classifier matched the data labeled as ground truth, controlling for the accuracy of a random classifier as measured by the expected accuracy.

```r
library(randomForest)
fit11 <- randomForest(area_category ~DMC+DC+temp, data=trainData, importance=TRUE, ntree=100)

Pred1<-predict(fit11,newdata=testData)
conf1 <- table(Pred1, testData$area_category)

library(e1071)
library(caret)

confusionMatrix(conf1)
```

```
## Confusion Matrix and Statistics
##
##
## Pred1   1   2   3   4   5   6   7   8
##     1 123  11   7   2   0   3   1   1
##     2   6   1   0   1   0   0   0   0
##     3   0   0   1   0   0   0   0   0
##     4   0   0   0   0   0   0   0   0
##     5   0   0   0   0   0   0   0   0
##     6   0   0   0   0   0   0   0   0
##     7   0   0   0   0   0   0   0   0
##     8   0   0   0   0   0   0   0   0
## 
## Overall Statistics
##                                           
##                 Accuracy : 0.7962         
##                   95% CI : (0.7246, 0.8562)
##      No Information Rate : 0.8217         
##      P-Value [Acc > NIR] : 0.8267         
##                                           
##                    Kappa : 0.0787         
##   Mcnemar's Test P-Value : NA             
## 
## Statistics by Class:
## 
##                      Class: 1 Class: 2 Class: 3 Class: 4 Class: 5 Class: 6
## Sensitivity            0.9535 0.083333 0.125000  0.00000       NA  0.00000
## Specificity            0.1071 0.951724 1.000000  1.00000        1  1.00000
## Pos Pred Value         0.8311 0.125000 1.000000      NaN       NA      NaN
## Neg Pred Value         0.3333 0.926174 0.955128  0.98089       NA  0.98089
## Prevalence             0.8217 0.076433 0.050955  0.01911        0  0.01911
## Detection Rate         0.7834 0.006369 0.006369  0.00000        0  0.00000
## Detection Prevalence   0.9427 0.050955 0.006369  0.00000        0  0.00000
## Balanced Accuracy      0.5303 0.517529 0.562500  0.50000       NA  0.50000
##                      Class: 7 Class: 8
## Sensitivity          0.000000 0.000000
## Specificity          1.000000 1.000000
## Pos Pred Value            NaN      NaN
## Neg Pred Value       0.993631 0.993631
## Prevalence           0.006369 0.006369
## Detection Rate       0.000000 0.000000
## Detection Prevalence 0.000000 0.000000
## Balanced Accuracy    0.500000 0.500000
```

As we saw from our feature importance plot that most of the variables do not contribute much in predicting the area in our test data, so we have removed most of the features and have performed the predictive analysis just using 3 variables which are the DMC, DC indexes and the temperature. SO in this case we can observe that our accuracy did not decrease much and is still 79.6 %, which indicates a good model for aor prediction of area on the test data. But, at the same time this represents that our data does not have important features that can help us in determining the area more accurately.

So for improvement in this case, we can consider having more features which could be more important and help in determining the area more accurately. Further, if we have more data then we can train a better model which will be more accurate and would consider more possibilities of occurance of fire.

We can see that our model might be good for predicting low intensity fires, as we have more data of those. But, our model might not perform well on incidents that involve high internsity fires and cover larger areas, as we do not have data in our dataset that involves such cases. So, we in future we can consider to add such incidents in our dataset, to make a much reliable and accurate predictive model.

Summary

So, in summary we can infer from our data that forest fores depend mainly on the DMC index, DC index and the temperature. These are the parameters that play the most crucial role in determining whether fore would occur or not and if it occurs how much area will it cover. These insight are determined just from this data and may vary if we hve more data and consider more features for our analysis.