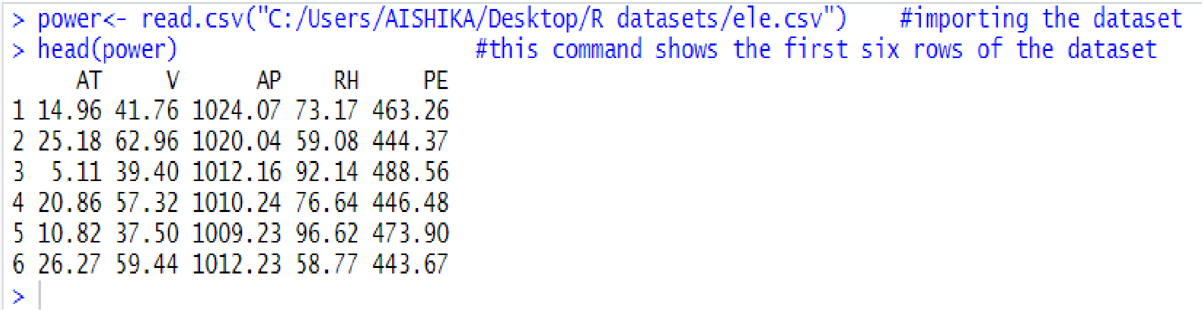
LINEAR REGRESSION

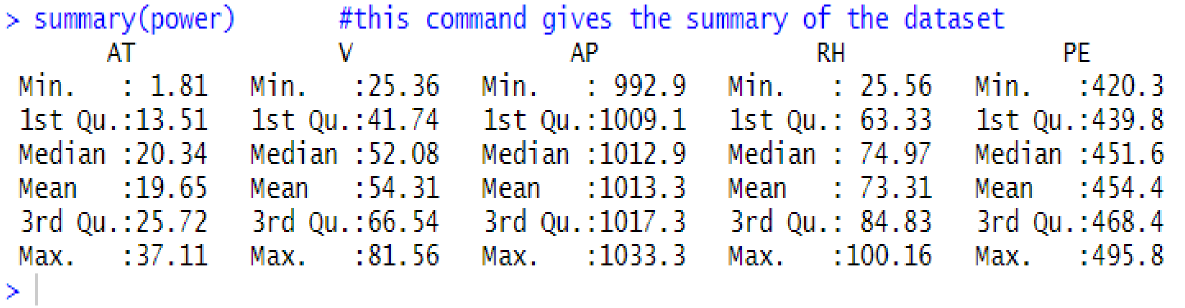
1. SELECTION OF DATASET:

The dataset used here is ‘Combined Cycle Power Plant Data Set’. The dataset is collected from a Combined Cycle Power Plant over 6 years (2006-2011), when the plant was set to work with full load. Here we want to predict the net hourly electrical energy output (EP) of the plant.

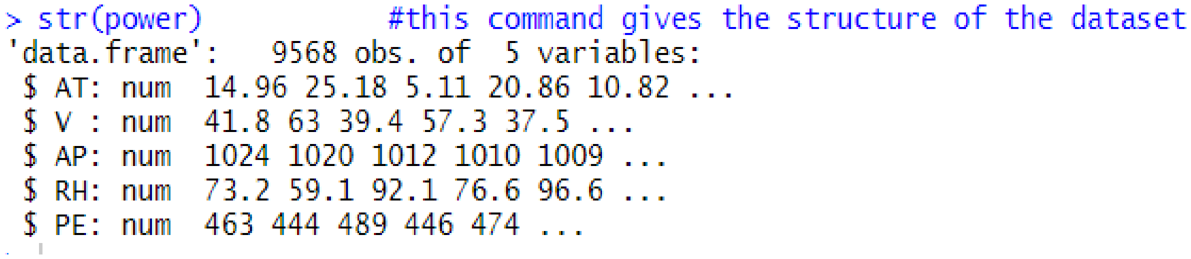
Dimension: 9568 rows and 5 columns
Attributes used: Temperature (AT), Ambient Pressure (AP), Relative Humidity (RH) and Exhaust Vacuum (V) and the net hourly electrical energy output (PE) of the plant.



```
> power<- read.csv("C:/Users/AISHIKA/Desktop/R datasets/ele.csv")    #importing the dataset
> head(power)                          #this command shows the first six rows of the dataset
     AT     V      AP    RH     PE
1 14.96 41.76 1024.07 73.17 463.26
2 25.18 62.96 1020.04 59.08 444.37
3  5.11 39.40 1012.16 92.14 488.56
4 20.86 57.32 1010.24 76.64 446.48
5 10.82 37.50 1009.23 96.62 473.90
6 26.27 59.44 1012.23 58.77 443.67
> 
```

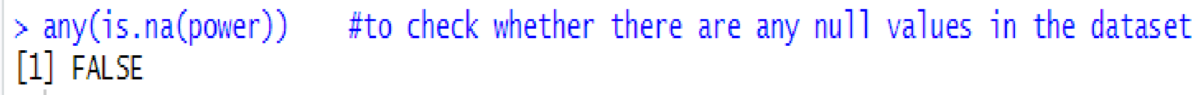


```
> summary(power)       #this command gives the summary of the dataset
       AT              V               AP               RH               PE       
 Min.   : 1.81   Min.   :25.36   Min.   : 992.9   Min.   : 25.56   Min.   :420.3  
 1st Qu.:13.51   1st Qu.:41.74   1st Qu.:1009.1   1st Qu.: 63.33   1st Qu.:439.8  
 Median :20.34   Median :52.08   Median :1012.9   Median : 74.97   Median :451.6  
 Mean   :19.65   Mean   :54.31   Mean   :1013.3   Mean   : 73.31   Mean   :454.4  
 3rd Qu.:25.72   3rd Qu.:66.54   3rd Qu.:1017.3   3rd Qu.: 84.83   3rd Qu.:468.4  
 Max.   :37.11   Max.   :81.56   Max.   :1033.3   Max.   :100.16   Max.   :495.8  
> 
```



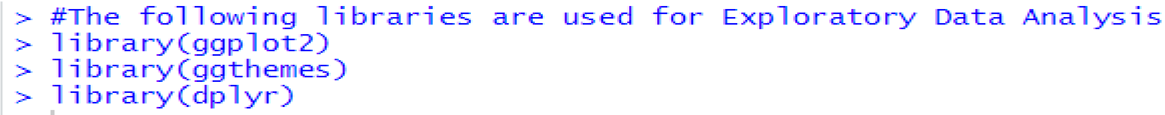
```
> str(power)           #this command gives the structure of the dataset
'data.frame':   9568 obs. of  5 variables:
 $ AT: num  14.96 25.18 5.11 20.86 10.82 ...
 $ V : num  41.8 63 39.4 57.3 37.5 ...
 $ AP: num  1024 1020 1012 1010 1009 ...
 $ RH: num  73.2 59.1 92.1 76.6 96.6 ...
 $ PE: num  463 444 489 446 474 ...
```

The structure of the data set gives a detailed description of the data types and the levels for the variables which have factors.

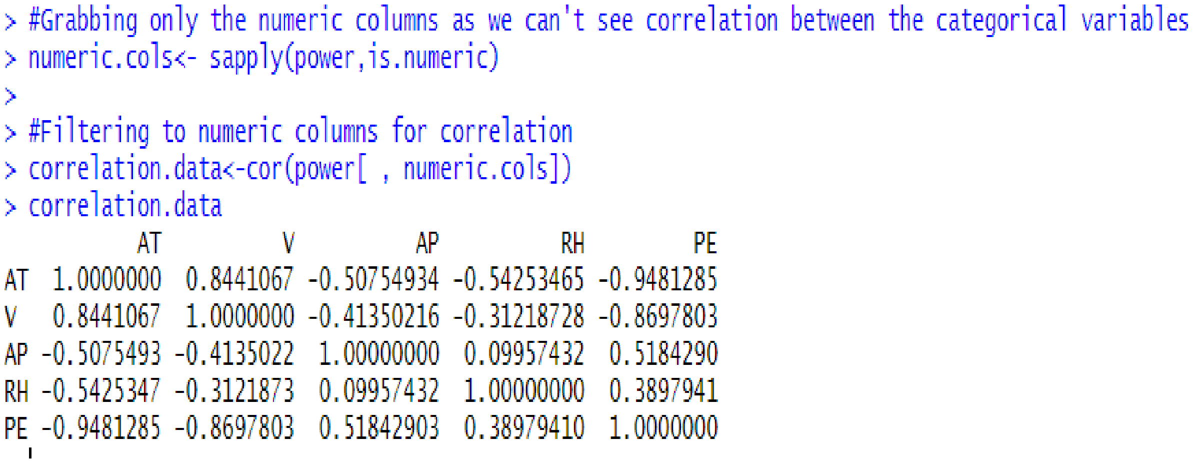


```
> any(is.na(power))    #to check whether there are any null values in the dataset
[1] FALSE
```

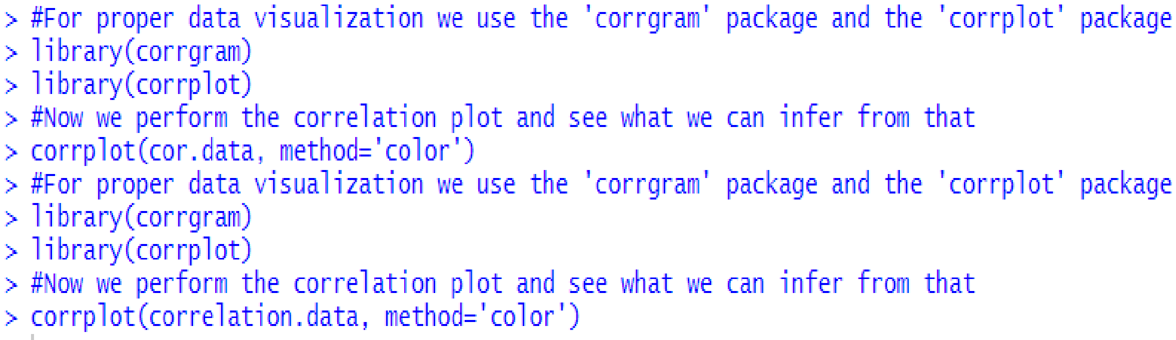
2. CORRELATION ANALYSIS OF THE DATASET:



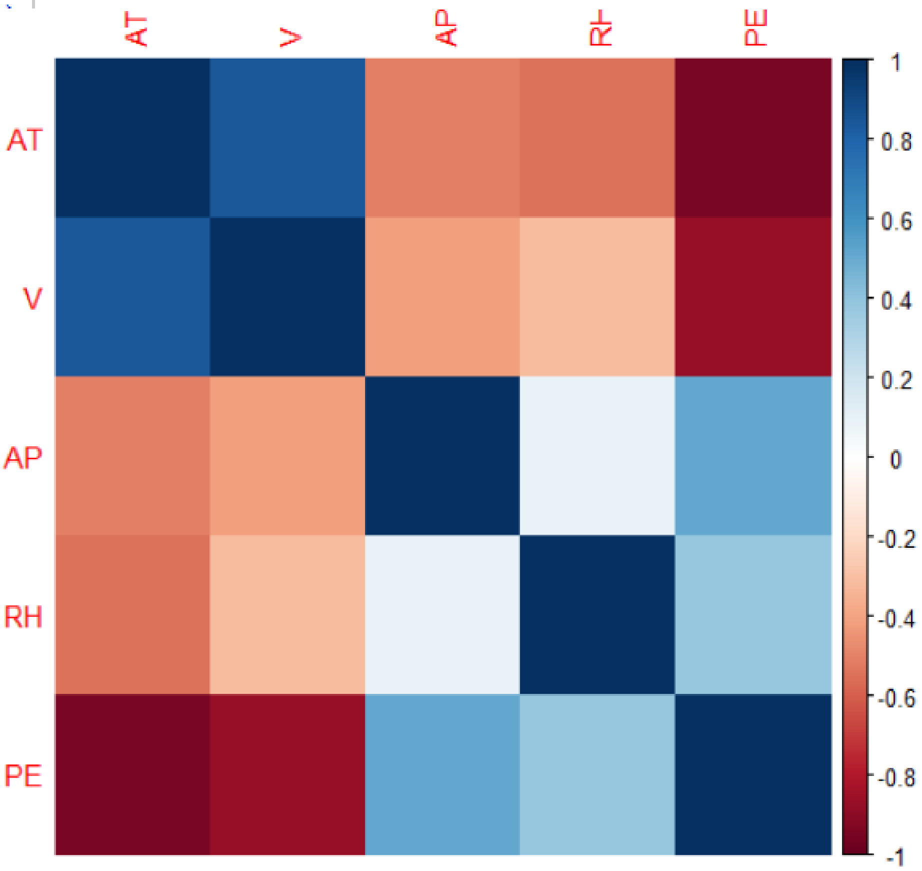
```
> #The following libraries are used for Exploratory Data Analysis
> library(ggplot2)
> library(ggthemes)
> library(dplyr)
```



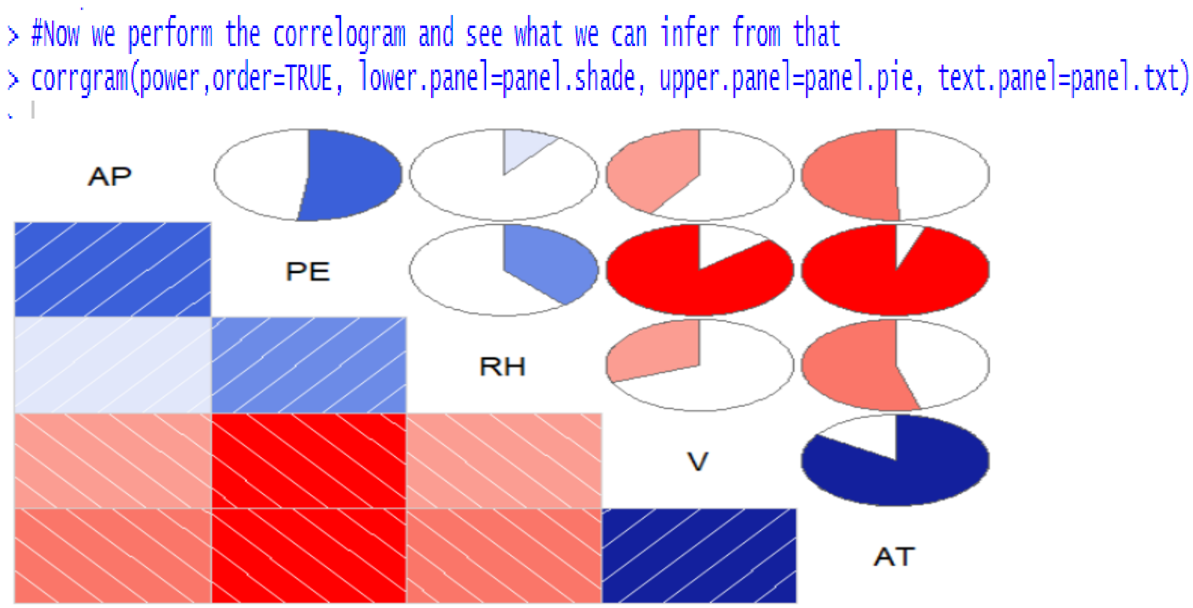
```
> #Grabbing only the numeric columns as we can't see correlation between the categorical variables
> numeric.cols<- sapply(power,is.numeric)
>
> #Filtering to numeric columns for correlation
> correlation.data<-cor(power[ , numeric.cols])
> correlation.data
           AT          V          AP          RH         PE
AT  1.0000000  0.8441067 -0.50754934 -0.54253465 -0.9481285
V   0.8441067  1.0000000 -0.41350216 -0.31218728 -0.8697803
AP -0.5075493 -0.4135022  1.00000000  0.09957432  0.5184290
RH -0.5425347 -0.3121873  0.09957432  1.00000000  0.3897941
PE -0.9481285 -0.8697803  0.51842903  0.38979410  1.0000000
```



```
> #For proper data visualization we use the 'corrgram' package and the 'corrplot' package
> library(corrgram)
> library(corrplot)
> #Now we perform the correlation plot and see what we can infer from that
> corrplot(cor.data, method='color')
> #For proper data visualization we use the 'corrgram' package and the 'corrplot' package
> library(corrgram)
> library(corrplot)
> #Now we perform the correlation plot and see what we can infer from that
> corrplot(correlation.data, method='color')
```

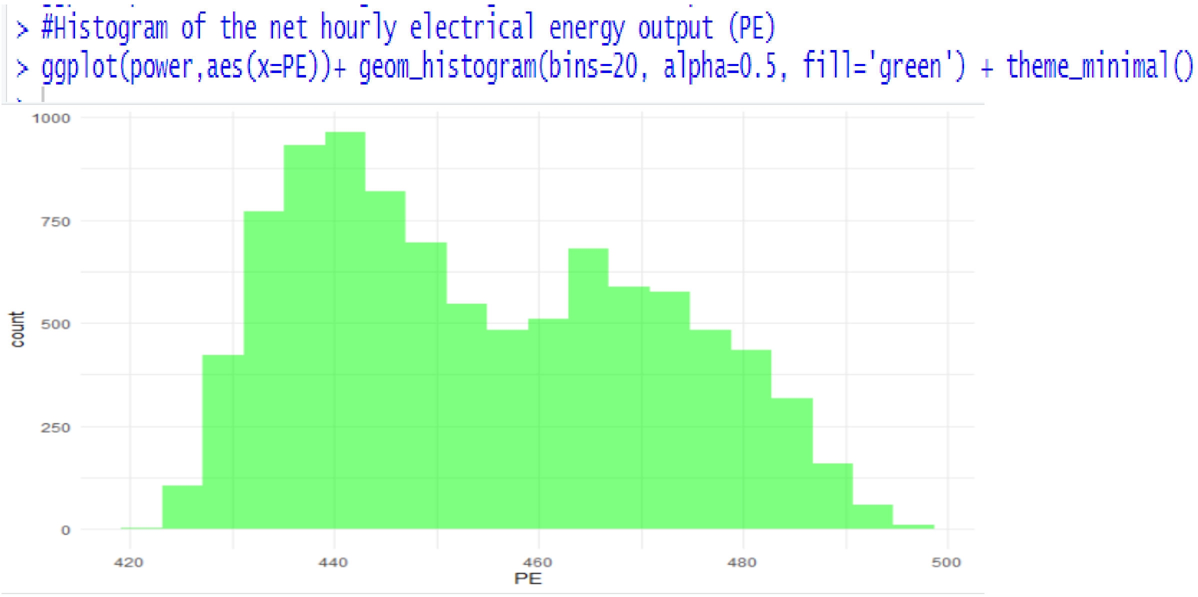


We can see from the above correlation plot that white indicates zero correlation, the shades from white to blue indicates positive correlation and from white to red indicates negative correlation. Positive correlation indicates that as the value of one variable increases, the value of the other variable also increases, i.e., directly proportional, while Negative correlation indicates that as the value of one variable increases the value of the other variable will decrease, i.e., indirectly proportional. We can see that V and AT, AP and PE are highly positively correlated to each other. Even there is positive correlation between RH and PE. There is high negative correlation between PE and AT, PE and V**.** We are going to predict the net hourly electrical energy output (PE) of the plant so it is the dependent variable. Here Temperature (AT), Ambient Pressure (AP), Relative Humidity (RH) and Exhaust Vacuum (V) are the independent variables.



```
> #Now we perform the correlogram and see what we can infer from that
> corrgram(power,order=TRUE, lower.panel=panel.shade, upper.panel=panel.pie, text.panel=panel.txt)
```

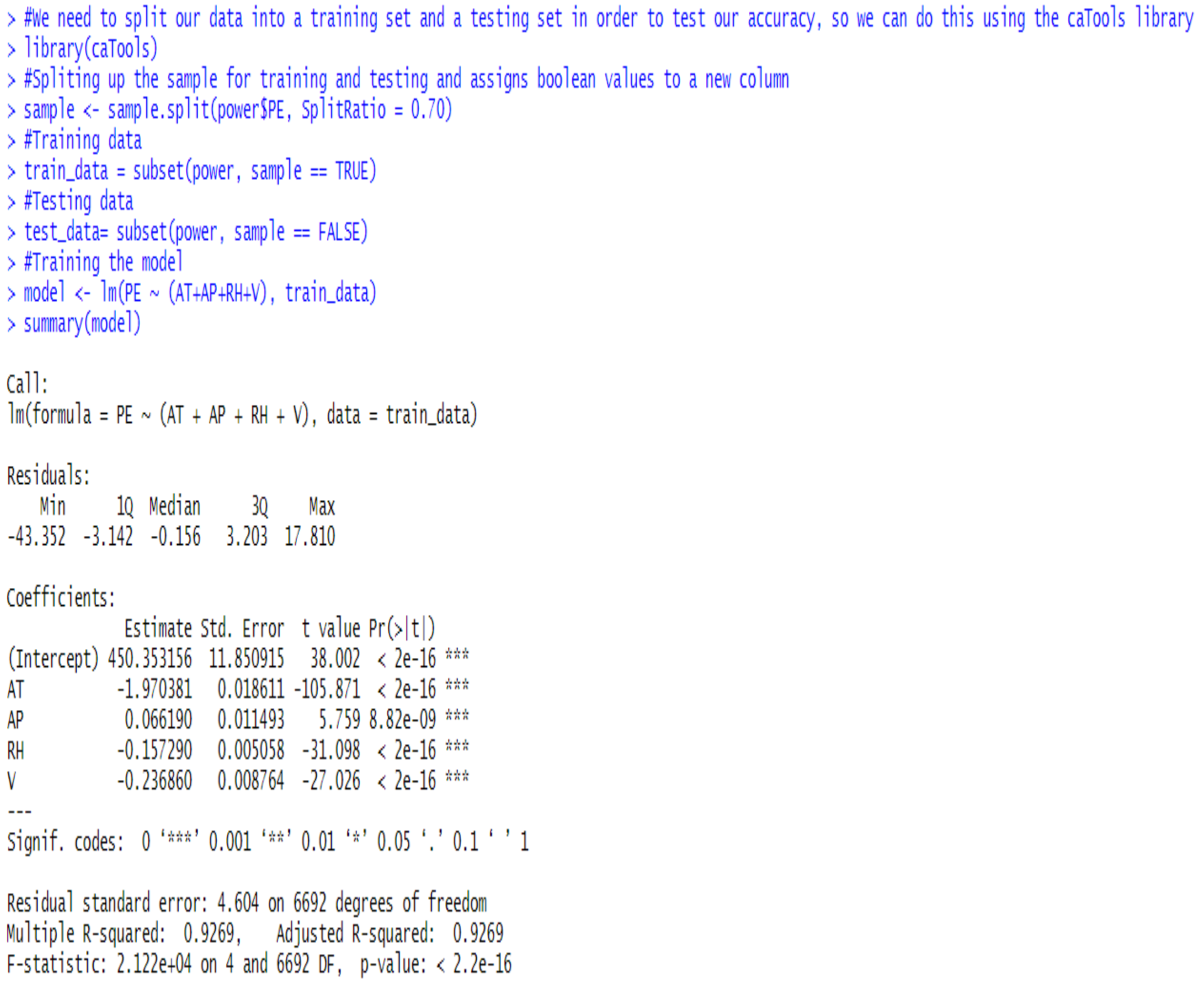
In R, correlograms are implemented using the ‘corrgram’ function. It shows the graph of the correlation matrix and is very useful for highlighting the most correlated values. The results of this plot can be interpreted in the same way as ‘corrplot’. Here blue is positive correlation and pink is negative correlation.



```
> #Histogram of the net hourly electrical energy output (PE)
> ggplot(power,aes(x=PE))+ geom_histogram(bins=20, alpha=0.5, fill='green') + theme_minimal()
```

As we have to predict PE so we draw a histogram of it. From the graph we can see that the high amount of PE is below the mean of the Ca and even there are more values beyond the mean.

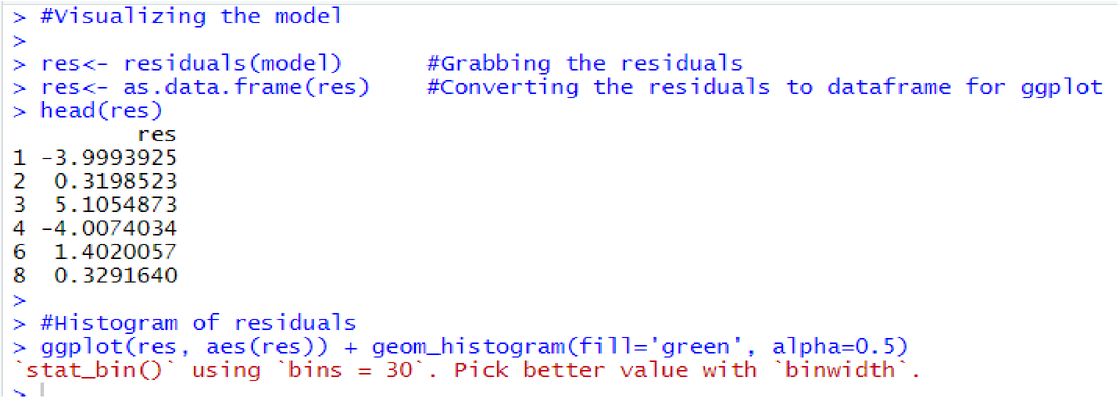
3. MULTIPLE LINEAR REGRESSION:



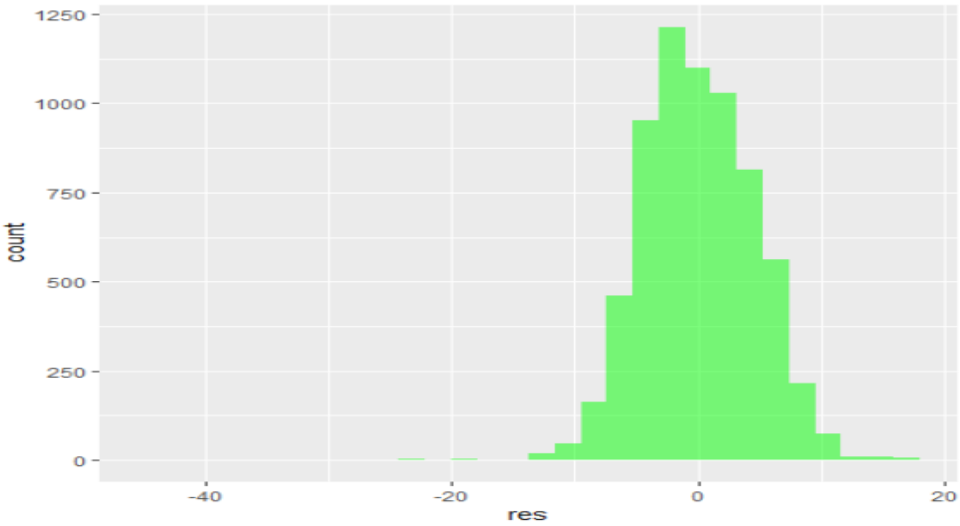
```
> #We need to split our data into a training set and a testing set in order to test our accuracy, so we can do this using the caTools library
> library(caTools)
> #Spliting up the sample for training and testing and assigns boolean values to a new column
> sample <- sample.split(power$PE, SplitRatio = 0.70)
> #Training data
> train_data = subset(power, sample == TRUE)
> #Testing data
> test_data= subset(power, sample == FALSE)
> #Training the model
> model <- lm(PE ~ (AT+AP+RH+V), train_data)
> summary(model)

Call:
lm(formula = PE ~ (AT + AP + RH + V), data = train_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-43.352  -3.142  -0.156   3.203  17.810 

Coefficients:
              Estimate Std. Error  t value Pr(>|t|)    
(Intercept) 450.353156  11.850915   38.002  < 2e-16 ***
AT           -1.970381   0.018611 -105.871  < 2e-16 ***
AP            0.066190   0.011493    5.759 8.82e-09 ***
RH           -0.157290   0.005058  -31.098  < 2e-16 ***
V            -0.236860   0.008764  -27.026  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.604 on 6692 degrees of freedom
Multiple R-squared:  0.9269,	Adjusted R-squared:  0.9269 
F-statistic: 2.122e+04 on 4 and 6692 DF,  p-value: < 2.2e-16
```

The residuals are the difference between the actual values of the variable we are predicting and predicted values from our regression.
The stars are for significance levels, with the number of asterisks displayed according to the p-value computed. *** indicates high significance. In this case, *** indicates that there is high significance between AT,AP,RH,V with PE.
The estimated coefficient is the value of slope calculated by the regression.
Standard Error of the Coefficient Estimate is the measure of the variability in the estimate for the coefficient.
t-value shows how many standard deviations the coefficient is far away from zero. Further it is away from zero, stronger the relationship between the variables.
p-value shows whether the overall model is significant or not. Coefficients having p-value less than the level of significance are said to be statistically significant. In this case we can say that the variables AT,AP,RH,V are statistically significant for the prediction of PE.
R-squared is an overall measure of the strength of association and adjusted R-squared gives a more proper estimate of the R-squared value for the population. Its value shows that 92.69 % of the variance in the net hourly electrical energy output (PE) of the plant can be predicted from

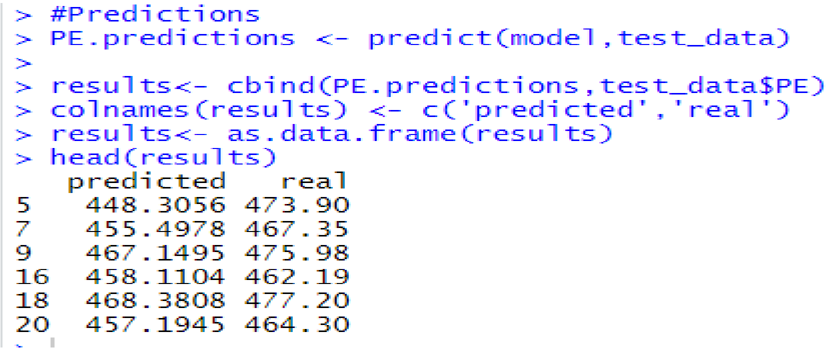
the variables Temperature (AT), Ambient Pressure (AP), Relative Humidity (RH) and Exhaust Vacuum (V).



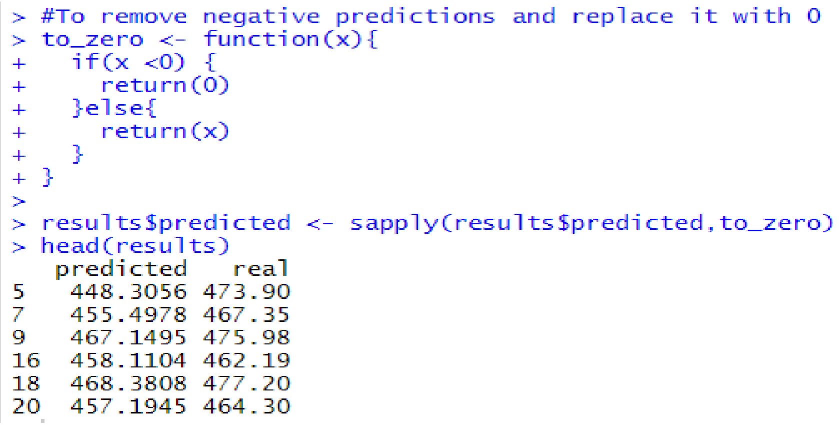
```
> #Visualizing the model
>
> res<- residuals(model)      #Grabbing the residuals
> res<- as.data.frame(res)    #Converting the residuals to dataframe for ggplot
> head(res)
         res
1 -3.9993925
2  0.3198523
3  5.1054873
4 -4.0074034
6  1.4020057
8  0.3291640
>
> #Histogram of residuals
> ggplot(res, aes(res)) + geom_histogram(fill='green', alpha=0.5)
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



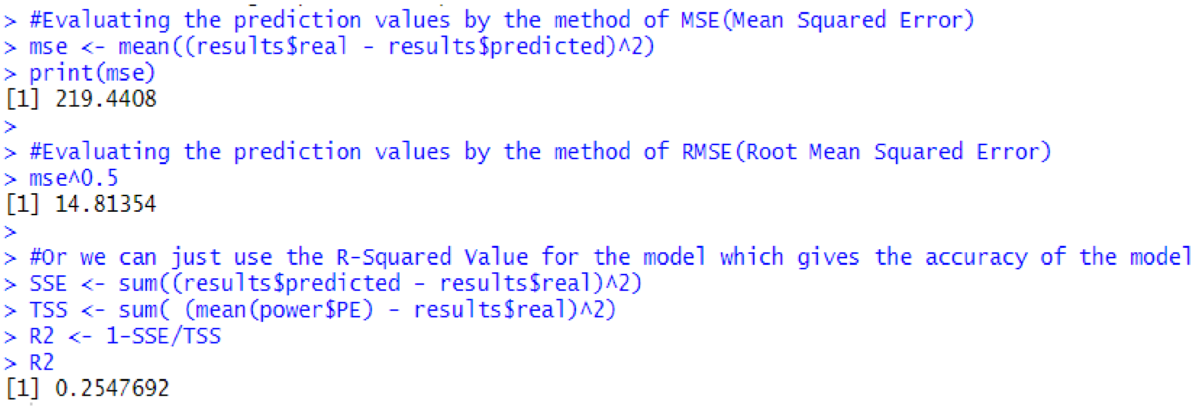
#We have to remove the negative values from here.



```
> #Predictions
> PE.predictions <- predict(model,test_data)
>
> results<- cbind(PE.predictions,test_data$PE)
> colnames(results) <- c('predicted','real')
> results<- as.data.frame(results)
> head(results)
   predicted   real
5   448.3056 473.90
7   455.4978 467.35
9   467.1495 475.98
16  458.1104 462.19
18  468.3808 477.20
20  457.1945 464.30
```



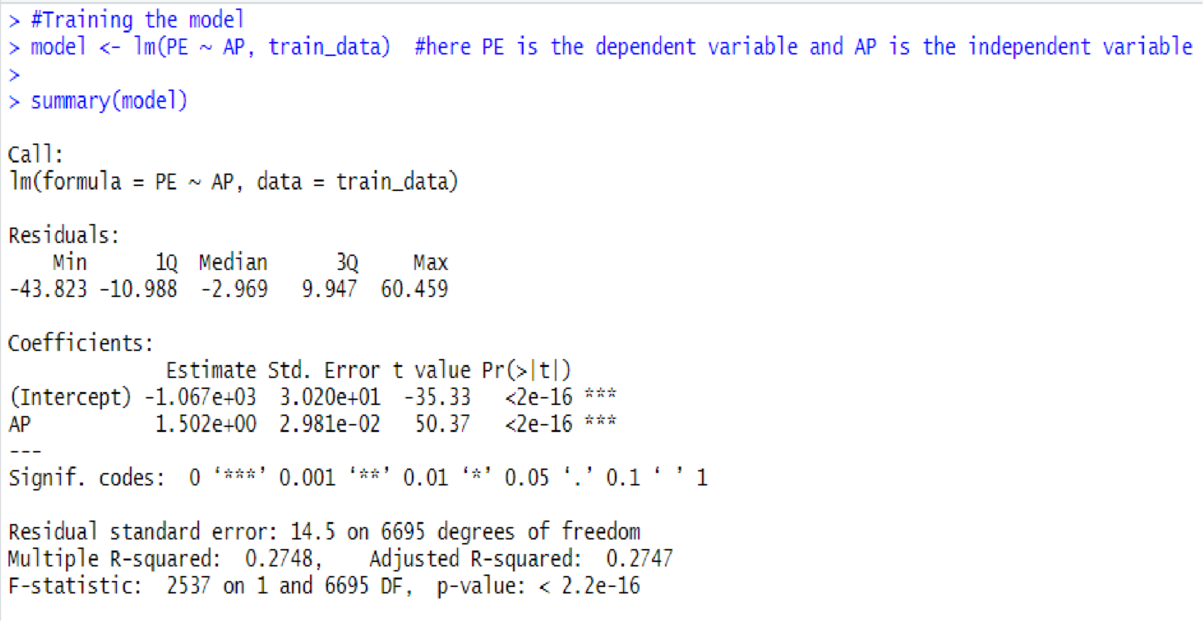
```
> #To remove negative predictions and replace it with 0
> to_zero <- function(x){
+   if(x <0) {
+     return(0)
+   }else{
+     return(x)
+   }
+ }
>
> results$predicted <- sapply(results$predicted,to_zero)
> head(results)
   predicted   real
5   448.3056 473.90
7   455.4978 467.35
9   467.1495 475.98
16  458.1104 462.19
18  468.3808 477.20
20  457.1945 464.30
```



```
> #Evaluating the prediction values by the method of MSE(Mean Squared Error)
> mse <- mean((results$real - results$predicted)^2)
> print(mse)
[1] 219.4408
>
> #Evaluating the prediction values by the method of RMSE(Root Mean Squared Error)
> mse^0.5
[1] 14.81354
>
> #Or we can just use the R-Squared Value for the model which gives the accuracy of the model
> SSE <- sum((results$predicted - results$real)^2)
> TSS <- sum( (mean(power$PE) - results$real)^2)
> R2 <- 1-SSE/TSS
> R2
[1] 0.2547692
```

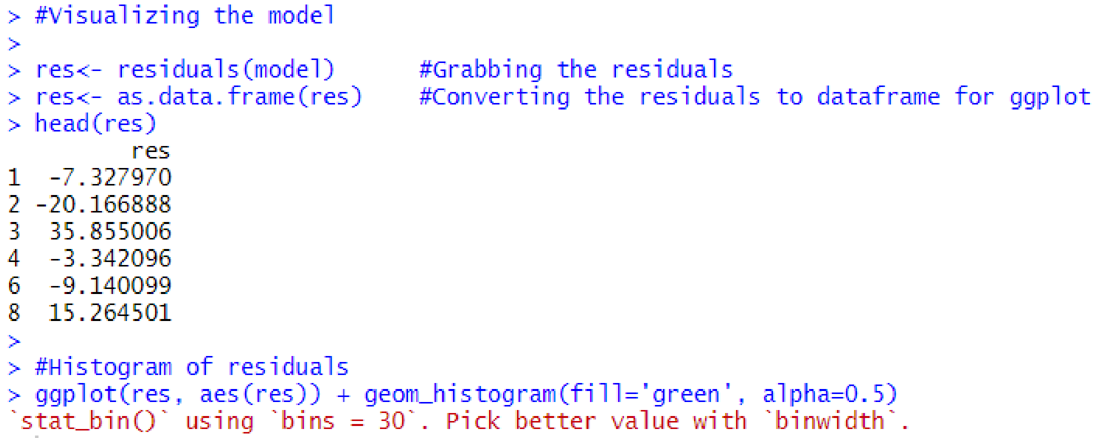
R-Squared (Coefficient of Determination) - This value lies between 0 and 1, and the higher it is, the better the model fits the data set.
Our main aim is to find those variables who give the lowest RMSE value and the highest R-Squared value.
The R-Squared for the training set is 25.47%. It means that the model can explain more than 25.477% of the variation.

4. SIMPLE LINEAR REGRESSION:

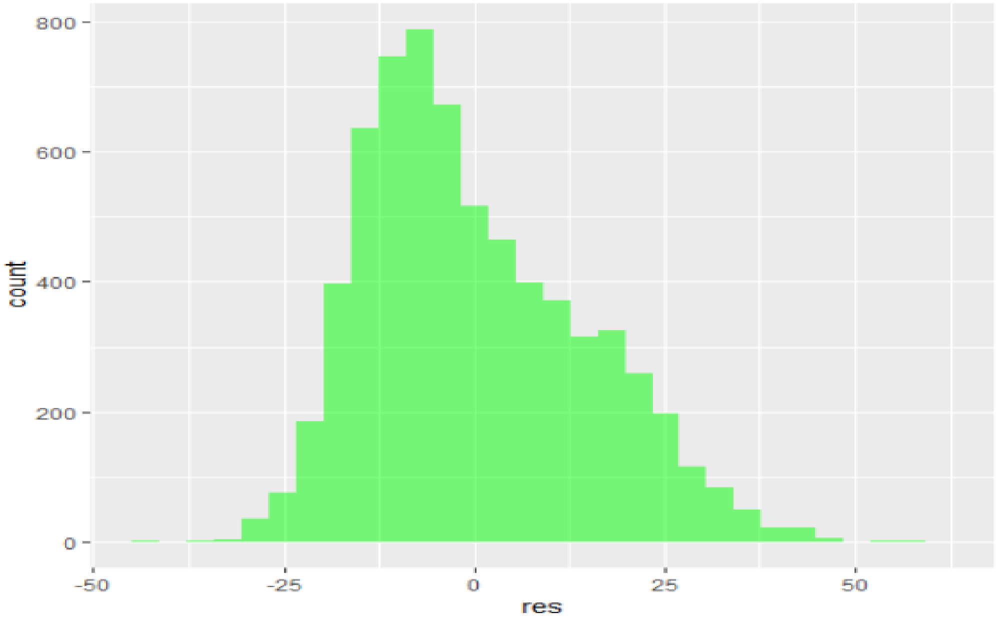


```
> #Training the model
> model <- lm(PE ~ AP, train_data)  #here PE is the dependent variable and AP is the independent variable
>
> summary(model)

Call:
lm(formula = PE ~ AP, data = train_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-43.823 -10.988  -2.969   9.947  60.459 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.067e+03  3.020e+01  -35.33   <2e-16 ***
AP           1.502e+00  2.981e-02   50.37   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 14.5 on 6695 degrees of freedom
Multiple R-squared:  0.2748,	Adjusted R-squared:  0.2747 
F-statistic:  2537 on 1 and 6695 DF,  p-value: < 2.2e-16
```

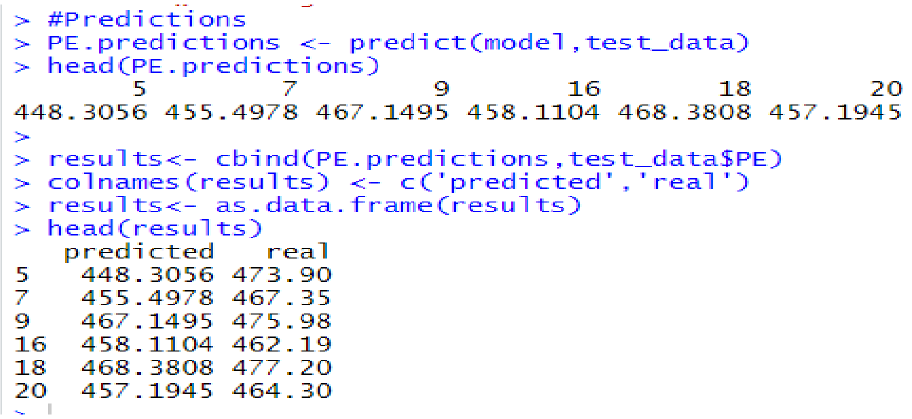
The residuals are the difference between the actual values of the variable we are predicting and predicted values from our regression.
The stars are for significance levels, with the number of asterisks displayed according to the p-value computed. *** indicates high significance. In this case, *** indicates that there is high significance between PE and AP.
The estimated coefficient is the value of slope calculated by the regression.
Standard Error of the Coefficient Estimate is the measure of the variability in the estimate for the coefficient.
t-value shows how many standard deviations the coefficient is far away from zero. Further it is away from zero, stronger the relationship between the variables.
p-value shows whether the overall model is significant or not. Coefficients having p-value less than the level of significance are said to be statistically significant. In this case we can say that the variable AP is statistically significant for the prediction of PE.
R-squared is an overall measure of the strength of association and adjusted R-squared gives a more proper estimate of the R-squared value for the population. Its value shows that 27.47 % of the variance in the net hourly electrical energy output (PE) of the plant can be predicted from Ambient Pressure (AP).



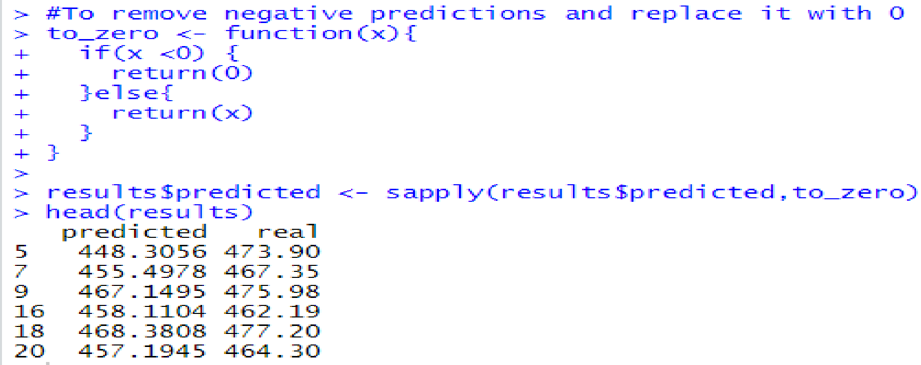
```
> #Visualizing the model
>
> res<- residuals(model)      #Grabbing the residuals
> res<- as.data.frame(res)    #Converting the residuals to dataframe for ggplot
> head(res)
         res
1  -7.327970
2 -20.166888
3  35.855006
4  -3.342096
6  -9.140099
8  15.264501
>
> #Histogram of residuals
> ggplot(res, aes(res)) + geom_histogram(fill='green', alpha=0.5)
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



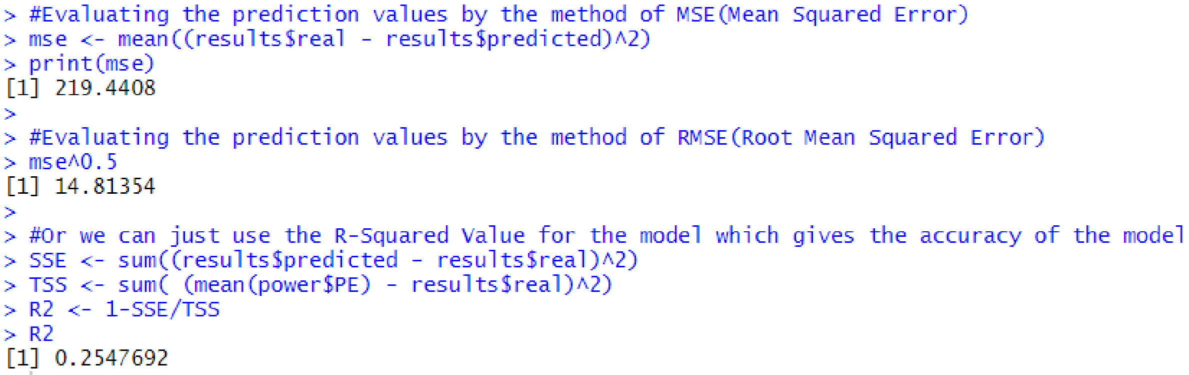
We have to remove the negative values from here.



```
> #Predictions
> PE.predictions <- predict(model,test_data)
> head(PE.predictions)
       5        7        9       16       18       20 
448.3056 455.4978 467.1495 458.1104 468.3808 457.1945 
>
> results<- cbind(PE.predictions,test_data$PE)
> colnames(results) <- c('predicted','real')
> results<- as.data.frame(results)
> head(results)
   predicted   real
5   448.3056 473.90
7   455.4978 467.35
9   467.1495 475.98
16  458.1104 462.19
18  468.3808 477.20
20  457.1945 464.30
```



```
> #To remove negative predictions and replace it with 0
> to_zero <- function(x){
+   if(x <0) {
+     return(0)
+   }else{
+     return(x)
+   }
+ }
>
> results$predicted <- sapply(results$predicted,to_zero)
> head(results)
   predicted   real
5   448.3056 473.90
7   455.4978 467.35
9   467.1495 475.98
16  458.1104 462.19
18  468.3808 477.20
20  457.1945 464.30
```



```
> #Evaluating the prediction values by the method of MSE(Mean Squared Error)
> mse <- mean((results$real - results$predicted)^2)
> print(mse)
[1] 219.4408
>
> #Evaluating the prediction values by the method of RMSE(Root Mean Squared Error)
> mse^0.5
[1] 14.81354
>
> #Or we can just use the R-Squared Value for the model which gives the accuracy of the model
> SSE <- sum((results$predicted - results$real)^2)
> TSS <- sum( (mean(power$PE) - results$real)^2)
> R2 <- 1-SSE/TSS
> R2
[1] 0.2547692
```

R-Squared (Coefficient of Determination) - This value lies between 0 and 1, and the higher it is, the better the model fits the data set.

Our main aim is to find those variables who give the lowest RMSE value and the highest R-Squared value.

The R-Squared for the training set is 25.47%. It means that the model can explain more than 25.47% of the variation.