Homework-2

Group IX (Rajeev Motwani)

Ashvin Khairnar

Dimple Bapna

Soumyajeet Patra

Balaji Kothandaraman Kalanidhi

206-941-8514(Ashvin Khairnar)

425-233-7801 (Dimple Bapna)

206-218-3144(Soumyajeet Patra)

206-915-5863 (Balaji Kothandaraman kalanidhi)

Percentage of Effort Contributed by Student 1: 25

Percentage of Effort Contributed by Student 2: 25

Percentage of Effort Contributed by Student 3: 25

Percentage of Effort Contributed by Student 4: 25

Signature of Student 1: Ashvin Khairnar

Signature of Student 2: Dimple Bapna

Signature of Student 3: Soumyajeet Patra

Signature of Student 4: Balaji Kothandaraman Kalanidhi

Submission Date: 03-14-2020

Home Work-2

```r
install.packages('corrplot')

## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-
library/3.6'
## (as 'lib' is unspecified)

install.packages('leaps')

## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-
library/3.6'
## (as 'lib' is unspecified)

install.packages('gvlma')

## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-
library/3.6'
## (as 'lib' is unspecified)

install.packages('MASS')

## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-
library/3.6'
## (as 'lib' is unspecified)

install.packages('effects')

## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-
library/3.6'
## (as 'lib' is unspecified)

install.packages('forecast')

## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-
library/3.6'
## (as 'lib' is unspecified)
```

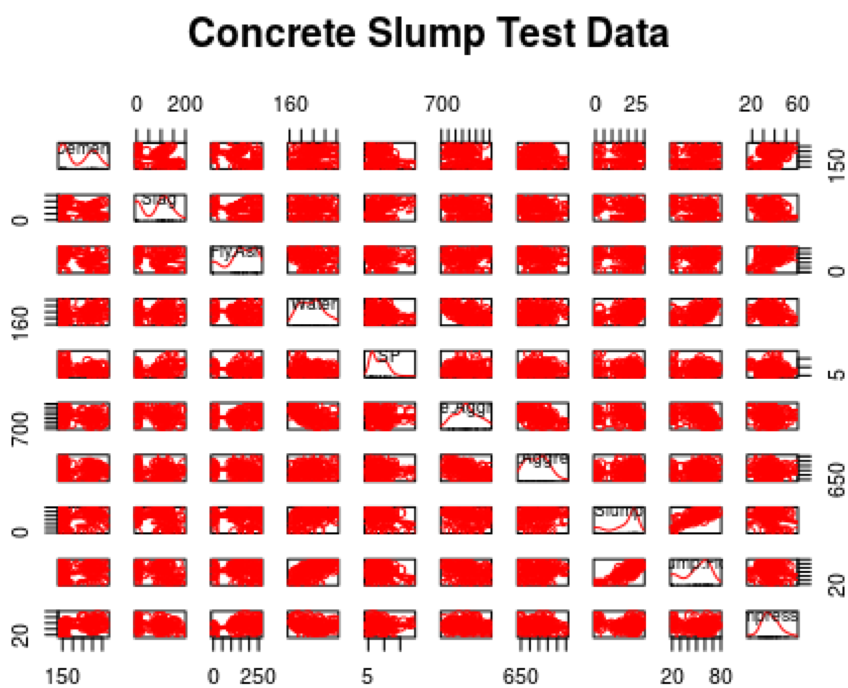
Problem 1

```
## Loading required package: carData

## corrplot 0.84 loaded

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

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

```
## The following object is masked from 'package:car':
##
##     recode

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

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

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

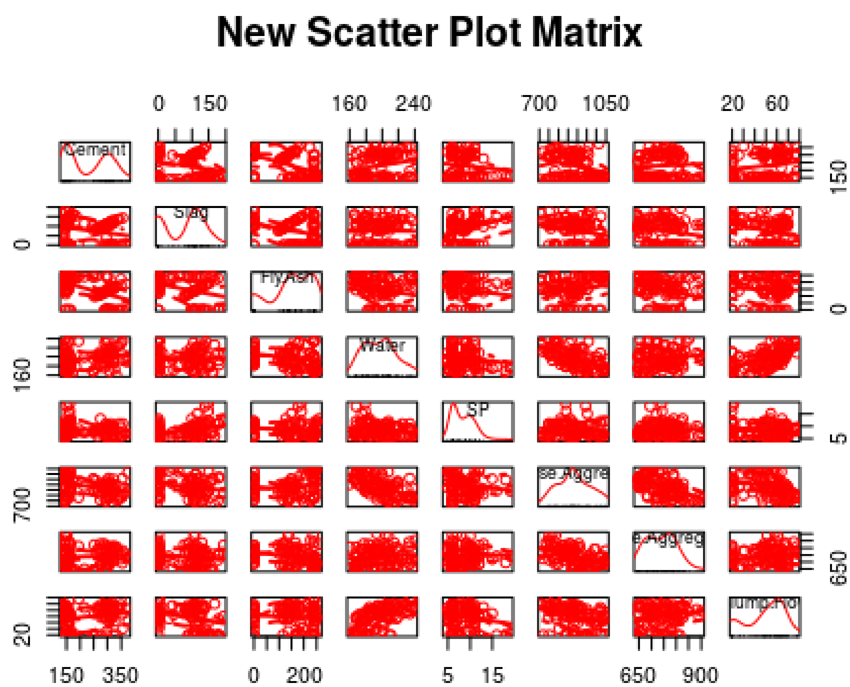
Question 1

```r
scatterplotMatrix(concreteSlump[,-1], main = "Concrete Slump Test Data", col
= 'Red')
```



We have 7 predictor variables i.e. Cement, Slag, Fly Ash, Water, SP, Coarse Aggregate and Fine Aggregate. Slump Flow will be our chosen response variable

```r
scatterplotMatrix(concreteSlump[, -c(1, 9, 11)], main = "New Scatter Plot
Matrix", col = 'Red')
```



Question 2

We will be using multiple linear regression and polynomial regression

```r
LinearRegression <- lm(`Slump Flow` ~ Cement + Slag + `Fly Ash` + Water + SP
+ `Coarse Aggregate` + `Fine Aggregate`, data = concreteSlump)
summary(LinearRegression)

##
## Call:
## lm(formula = `Slump Flow` ~ Cement + Slag + `Fly Ash` + Water +
##     SP + `Coarse Aggregate` + `Fine Aggregate`, data = concreteSlump)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -30.880 -10.428   1.815   9.601  22.953
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -252.87467  350.06649  -0.722   0.4718
## Cement                 0.05364    0.11236   0.477   0.6342
## Slag                  -0.00569    0.15638  -0.036   0.9710
## `Fly Ash`              0.06115    0.11402   0.536   0.5930
## Water                  0.73180    0.35282   2.074   0.0408 *
```

```
## SP                      0.29833    0.66263   0.450   0.6536
## `Coarse Aggregate`      0.07366    0.13510   0.545   0.5869
## `Fine Aggregate`        0.09402    0.14191   0.663   0.5092
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.84 on 95 degrees of freedom
## Multiple R-squared:  0.5022, Adjusted R-squared:  0.4656
## F-statistic: 13.69 on 7 and 95 DF,  p-value: 3.915e-12
```

```
PolynomialRegression <- lm(`Slump Flow` ~ (Cement + Slag + `Fly Ash` + Water
+ SP + `Coarse Aggregate` + `Fine Aggregate`)^2, data = concreteSlump)
summary(PolynomialRegression)
```

```
##
## Call:
## lm(formula = `Slump Flow` ~ (Cement + Slag + `Fly Ash` + Water +
##     SP + `Coarse Aggregate` + `Fine Aggregate`)^2, data = concreteSlump)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -23.8222  -6.0751   0.2499   4.7302  21.2758
##
## Coefficients:
##                                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          1.567e+03  1.277e+03   1.227 0.223715
## Cement                              -1.638e+00  1.387e+00  -1.181 0.241227
## Slag                                -5.560e+00  1.495e+00  -3.719 0.000386
## `Fly Ash`                           -3.498e+00  1.162e+00  -3.010 0.003568
## Water                               -6.165e+00  2.778e+00  -2.219 0.029543
## SP                                  -9.203e+01  1.474e+02  -0.624 0.534359
## `Coarse Aggregate`                  -5.943e-01  5.978e-01  -0.994 0.323325
## `Fine Aggregate`                    -9.309e-01  7.902e-01  -1.178 0.242545
## Cement:Slag                         -2.639e-04  5.594e-04  -0.472 0.638511
## Cement:`Fly Ash`                     3.774e-04  4.528e-04   0.834 0.407198
## Cement:Water                         4.472e-03  2.183e-03   2.049 0.044004
## Cement:SP                            4.826e-02  5.069e-02   0.952 0.344250
## Cement:`Coarse Aggregate`            5.554e-04  5.822e-04   0.954 0.343217
## Cement:`Fine Aggregate`              3.448e-04  6.659e-04   0.518 0.606098
## Slag:`Fly Ash`                       9.259e-04  4.603e-04   2.011 0.047927
## Slag:Water                           1.246e-02  2.541e-03   4.903 5.44e-06
## Slag:SP                              4.740e-02  7.788e-02   0.609 0.544640
## Slag:`Coarse Aggregate`              1.928e-03  5.389e-04   3.577 0.000618
## Slag:`Fine Aggregate`                1.972e-03  7.217e-04   2.732 0.007860
## `Fly Ash`:Water                      5.582e-03  1.770e-03   3.153 0.002331
## `Fly Ash`:SP                         4.320e-02  5.692e-02   0.759 0.450241
## `Fly Ash`:`Coarse Aggregate`         1.428e-03  4.753e-04   3.005 0.003624
## `Fly Ash`:`Fine Aggregate`           1.433e-03  5.691e-04   2.519 0.013940
## Water:SP                             5.024e-02  1.347e-01   0.373 0.710204
## Water:`Coarse Aggregate`             2.135e-03  1.191e-03   1.793 0.077110
```

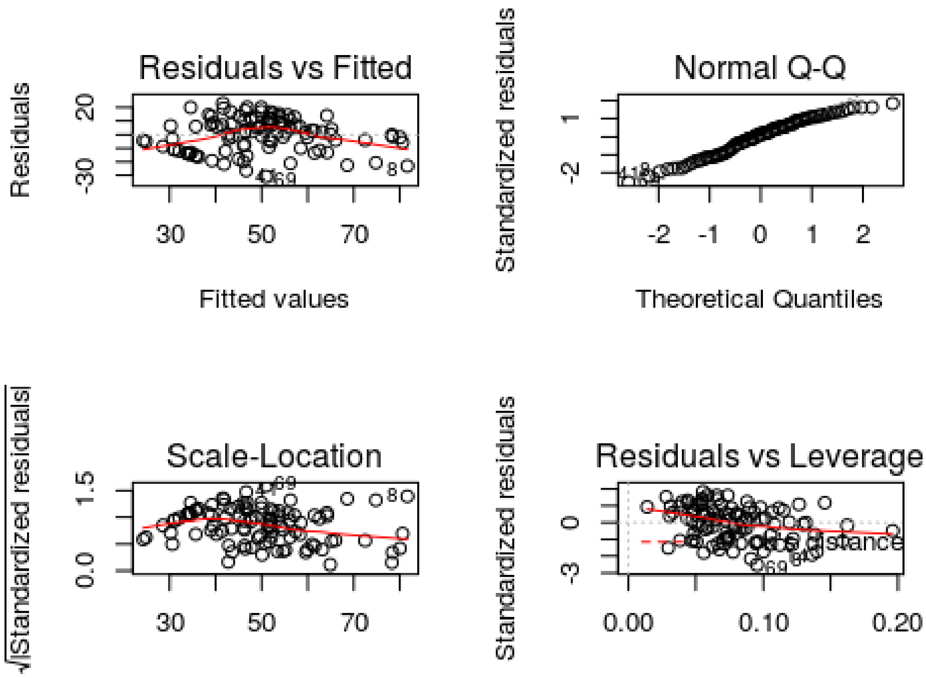
```
## Water:`Fine Aggregate`               4.104e-03  1.841e-03   2.229 0.028857
## SP:`Coarse Aggregate`                3.877e-02  5.893e-02   0.658 0.512625
## SP:`Fine Aggregate`                  3.905e-02  6.008e-02   0.650 0.517680
## `Coarse Aggregate`:`Fine Aggregate` -1.164e-04  4.208e-04  -0.276 0.782943
##
## (Intercept)
## Cement
## Slag                                ***
## `Fly Ash`                           **
## Water                               *
## SP
## `Coarse Aggregate`
## `Fine Aggregate`
## Cement:Slag
## Cement:`Fly Ash`
## Cement:Water                        *
## Cement:SP
## Cement:`Coarse Aggregate`
## Cement:`Fine Aggregate`
## Slag:`Fly Ash`                      *
## Slag:Water                          ***
## Slag:SP
## Slag:`Coarse Aggregate`             ***
## Slag:`Fine Aggregate`               **
## `Fly Ash`:Water                     **
## `Fly Ash`:SP
## `Fly Ash`:`Coarse Aggregate`        **
## `Fly Ash`:`Fine Aggregate`          *
## Water:SP
## Water:`Coarse Aggregate`            .
## Water:`Fine Aggregate`              *
## SP:`Coarse Aggregate`
## SP:`Fine Aggregate`
## `Coarse Aggregate`:`Fine Aggregate`
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.27 on 74 degrees of freedom
## Multiple R-squared:  0.7519, Adjusted R-squared:  0.658
## F-statistic:  8.01 on 28 and 74 DF,  p-value: 3.907e-13
```

Multiple Linear Regression will be our preffered model of choice as polynomial regression might result in overfitting

Question 3

Regression diagnostics with typical approach:

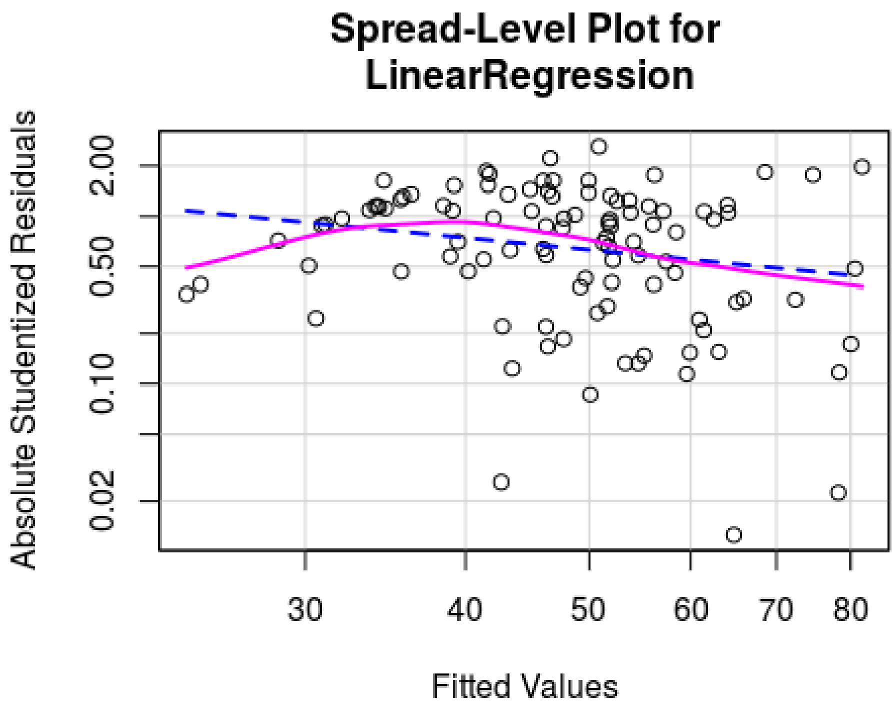
```r
par(mfrow = c(2, 2))
plot(LinearRegression)
```



In the top right graph we can see our normality assumption is satisfied

Homoscedasticity

```
spreadLevelPlot(LinearRegression)
```

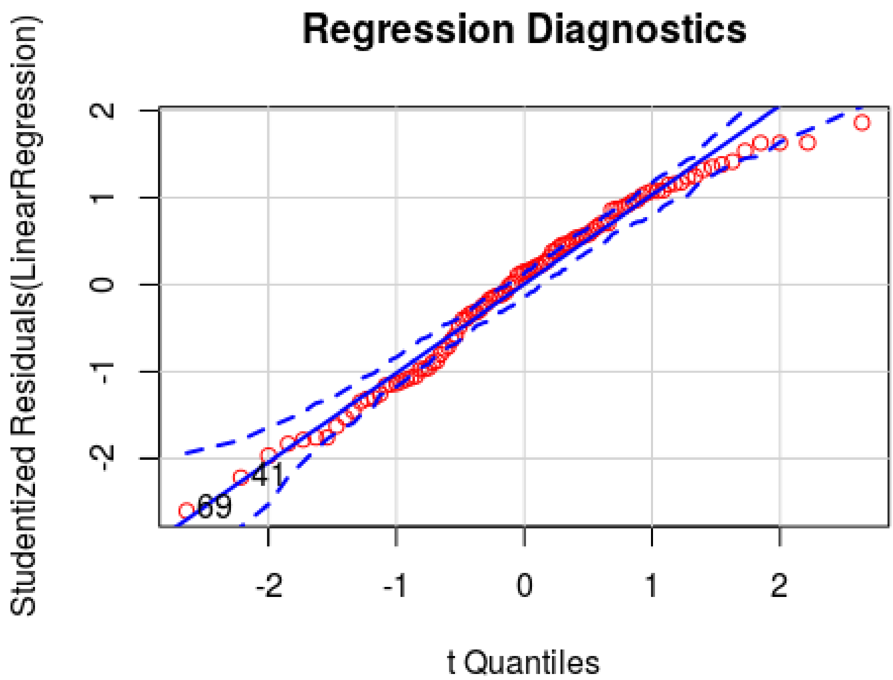



```
##
## Suggested power transformation:  1.743362
```

The abovegraph shows a random band around a horizontal line, so the homoscedasticity assumption is satisfied.

Regression diagnostics with enhanced approach:

```
qqPlot(LinearRegression, id.method = "identify", main = "Regression
Diagnostics", col.lines = 'blue', col = 'red')
```




```
## [1] 41 69
```

Points are within a line and are within the confidence bounds indicating satisfaction of normality assumption

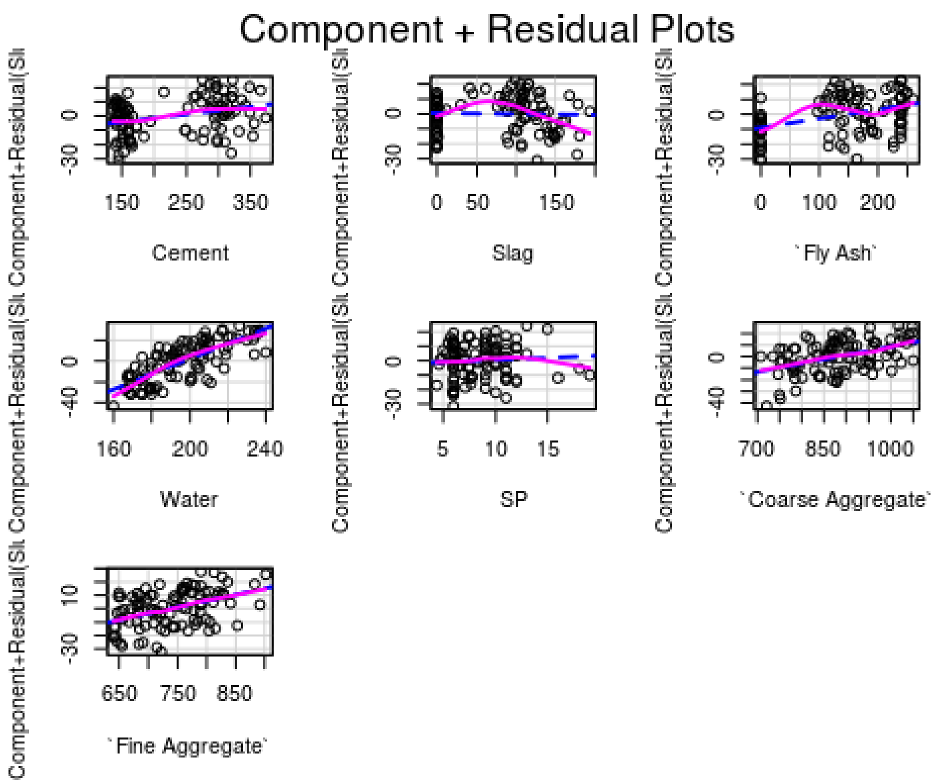
Independence

```
durbinWatsonTest(LinearRegression)

##  lag Autocorrelation D-W Statistic p-value
##    1     -0.01249995      2.009189   0.828
##  Alternative hypothesis: rho != 0
```

The non-significant p-value of 0.808 signifies no autocorrelation.

Linearity

```
crPlots(LinearRegression)
```



all graphs denote that the linearity assumption has been satisfied.

- Homoscedasticity

```
ncvTest(LinearRegression)

## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.2327094, Df = 1, p = 0.62952
```

There is no evidence of heteroscedasticity due to non significant p value . ## Question 4

- Outliers

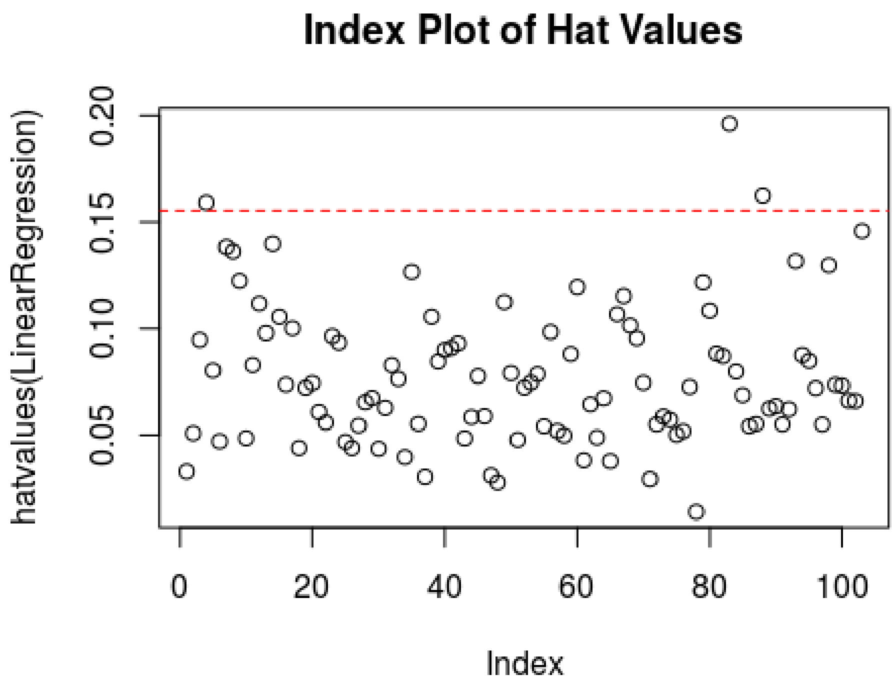
```
outlierTest(LinearRegression)

## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##     rstudent unadjusted p-value Bonferroni p
## 69 -2.603738           0.010717           NA
```

No outlier.

- High leverage points

```
hat.plot <- function(LinearRegression) {
  p <- length(coefficients(LinearRegression))
  n <- length(fitted(LinearRegression))
  plot(hatvalues(LinearRegression), main="Index Plot of Hat Values")
  abline(h = c(2, 3) * p / n, col = "red", lty = 2)
```

```
  identify(1:n, hatvalues(LinearRegression),
names(hatvalues(LinearRegression)))
  }
hat.plot(LinearRegression)
```

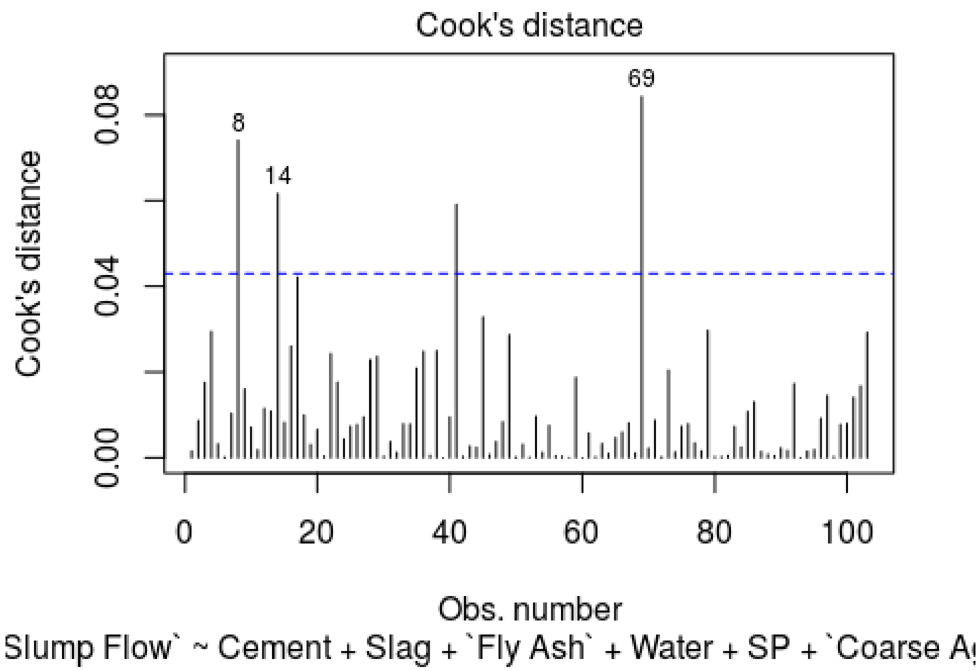



```
## integer(0)
```

A few observations are over the line

Influential observations

```
cutoff <- 4 / (nrow(concreteSlump) - length(LinearRegression$coefficients) -
2)
plot(LinearRegression, which = 4, cook.levels = cutoff)
abline(h = cutoff, lty = 2, col = "blue")
```



69, 8 and 14 are influential observations.

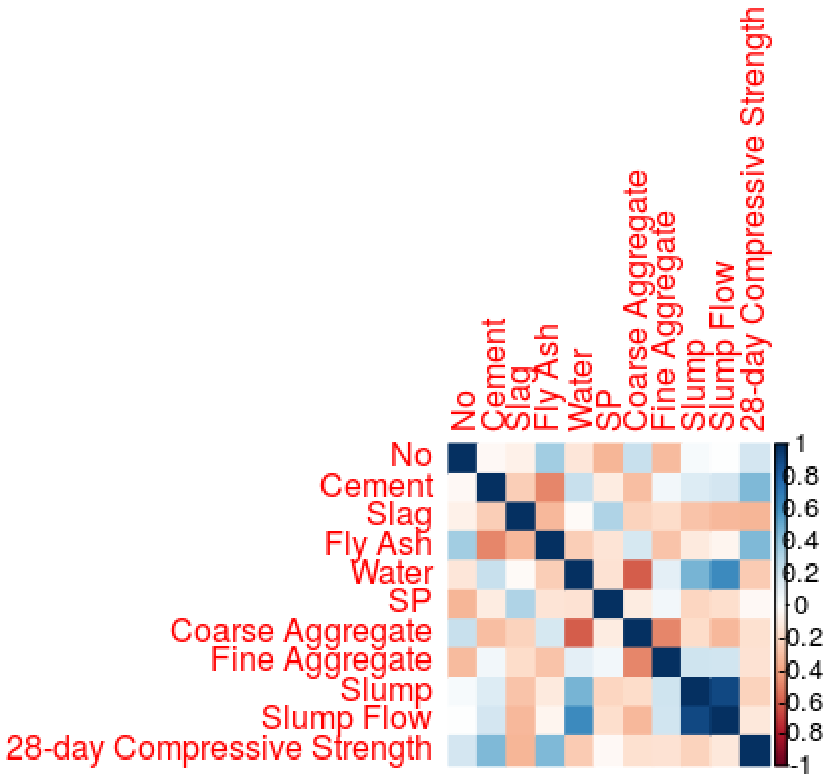
Corrective measures

Transforming variables

```
summary(powerTransform(concreteSlump$`Slump Flow`))

## bcPower Transformation to Normality
##                             Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## concreteSlump$`Slump Flow`    1.4678           1        0.9342       2.0015
##
## Likelihood ratio test that transformation parameter is equal to 0
##  (log transformation)
##                             LRT df       pval
## LR test, lambda = (0) 31.06187  1 2.4993e-08
##
## Likelihood ratio test that no transformation is needed
##                             LRT df     pval
## LR test, lambda = (1) 3.036391  1 0.081417
```

5:

```
data<-read_excel('Concrete Slump Test Data.xlsx')
M <- cor(data)
corrplot(M, method = "color")
```



After looking at correlation matrix, we can observe that, Cement and Fly Ash both have same affect on Compressive strength. correlations are very weak for slump w.r.t predictor variables. slump flow and slump are highly correlated. As water increase Slump and Flow increases.

let's try to fit the predictor for our responses.

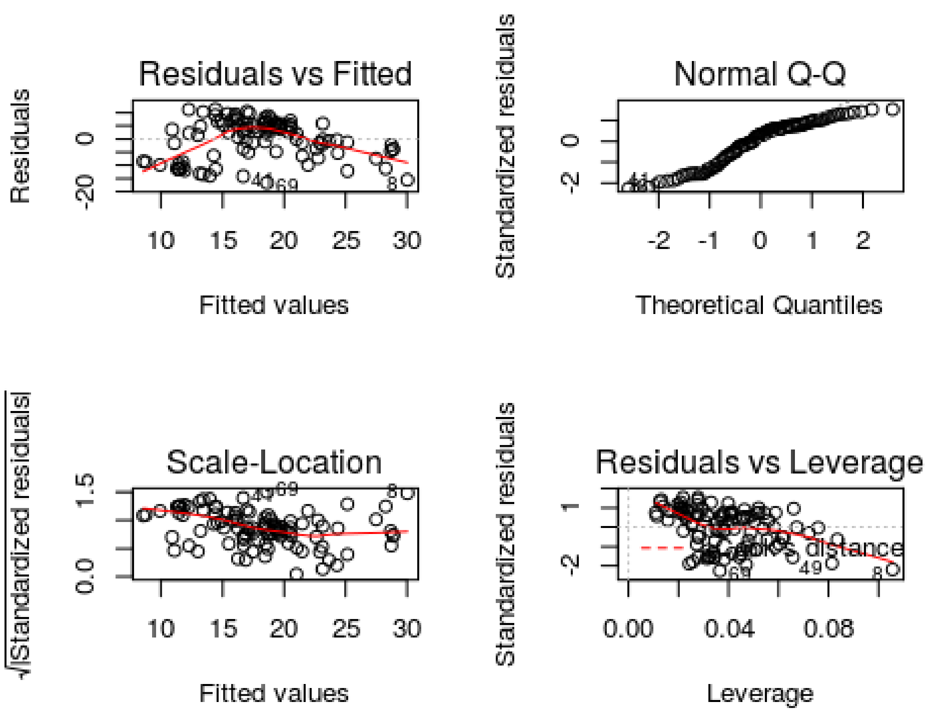
```
set.seed(0)
size <- floor(0.66*nrow(data))
train_ind <- sample(seq_len(nrow(data)), size = size)

train <- data[train_ind, ]
test <- data[-train_ind, ]
mean_squared_err <- function(data){
  return(mean(data^2))
}
```

For Slump

```
fit <- lm(Slump ~ Cement + Slag + Water,
     data = data)
summary(fit)

##
## Call:
## lm(formula = Slump ~ Cement + Slag + Water, data = data)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.618  -5.240   2.285   5.551  11.317
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.859118   7.400671  -2.413  0.01765 *
## Cement       -0.002705   0.009919  -0.273  0.78562
## Slag         -0.040169   0.012625  -3.182  0.00196 **
## Water         0.201157   0.037565   5.355 5.56e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.474 on 99 degrees of freedom
## Multiple R-squared:  0.292,  Adjusted R-squared:  0.2706
## F-statistic: 13.61 on 3 and 99 DF,  p-value: 1.668e-07
```

```
par(mfrow = c(2,2))
plot(fit)
```



After the initial Assessment we find all the predictors are very loosely correlated to Slump (all P-values > 0.05). So, the accuracy of the model is going to be very low. since, this model doesn't fit correctly for slump.

let's try to fit the model anyways.

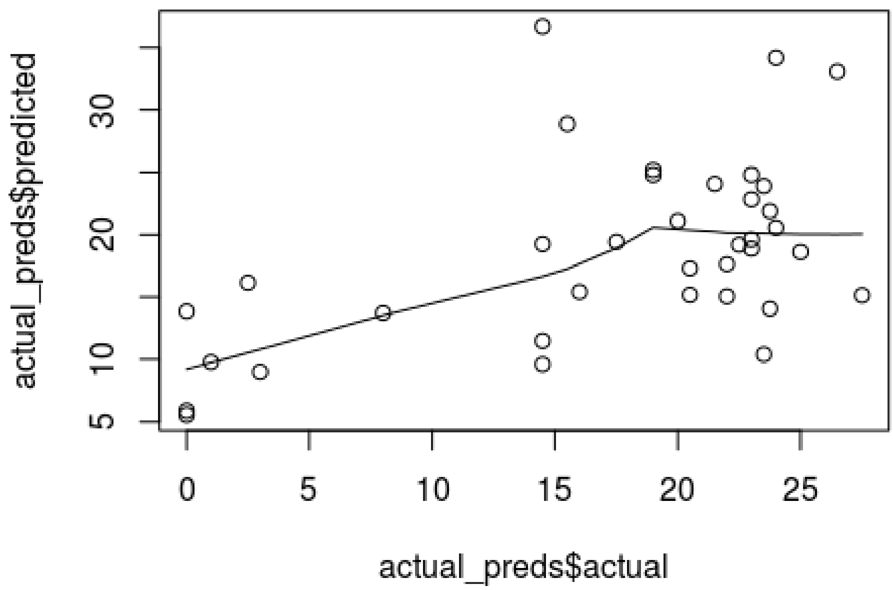
```r
# training phase
y_train_fit <-
  lm(Slump ~  Cement + Slag + `Fly Ash` + Water + SP + `Coarse Aggregate` +
`Fine Aggregate`,
     data = train)
summary(y_train_fit)
```

```
##
## Call:
## lm(formula = Slump ~ Cement + Slag + `Fly Ash` + Water + SP +
##     `Coarse Aggregate` + `Fine Aggregate`, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -16.001  -5.333   1.898   4.809  11.899
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)        -75.302884 265.935656  -0.283    0.778
## Cement              -0.004239   0.082964  -0.051    0.959
## Slag                -0.029912   0.117839  -0.254    0.801
## `Fly Ash`            0.006398   0.086257   0.074    0.941
## Water                0.276705   0.271448   1.019    0.312
## SP                  -0.221548   0.517411  -0.428    0.670
## `Coarse Aggregate`   0.015321   0.101931   0.150    0.881
## `Fine Aggregate`     0.040537   0.108338   0.374    0.710
##
## Residual standard error: 7.609 on 59 degrees of freedom
## Multiple R-squared:  0.3576, Adjusted R-squared:  0.2814
## F-statistic: 4.693 on 7 and 59 DF,  p-value: 0.0003111
```

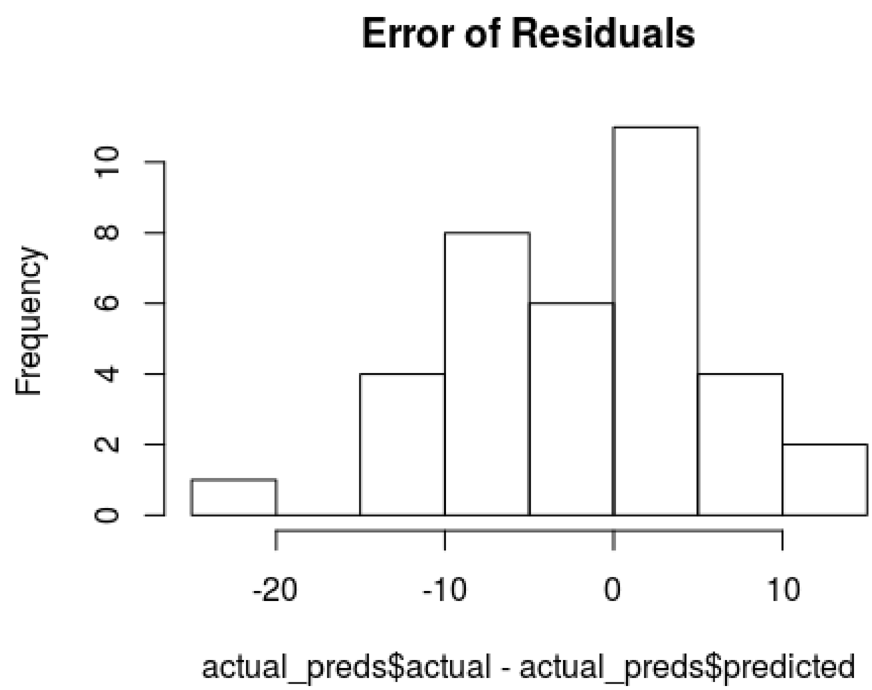
```r
# testing phase
y_pred <- predict(y_train_fit, test)


actual_preds <-
  data.frame(cbind(actual = test$Slump, predicted = y_pred))
cor(actual_preds$actual, actual_preds$predicted)
```

```
## [1] 0.5252733
```

```r
plot(actual_preds$actual, actual_preds$predicted)
lines(lowess(actual_preds$actual, actual_preds$predicted))
```




```r
accuracy(y_pred, test$Slump)

##                   ME     RMSE      MAE  MPE MAPE
## Test set -1.388559 7.762898 6.178486 -Inf  Inf

hist(actual_preds$actual - actual_preds$predicted,main='Error of Residuals')
```



The distribution is not normal, which show's that our predictions or fit were poor.

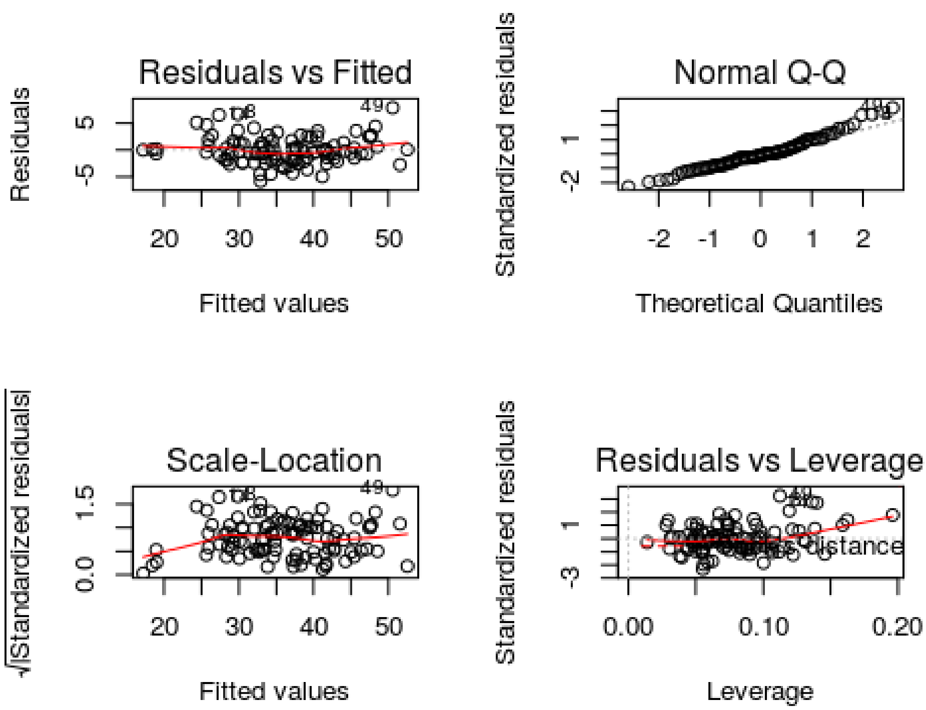
Here we see, that the correlation between $\hat{y}$ and y i.e (y predicted and y actual) after the testing is still very low (r = 0.5252). with R^2 value's to be 0.35 and Adjusted R^2 to be 0.28.

For 28-day Compressive Strength

```
fit <-
  lm(
    `28-day Compressive Strength` ~ Cement + Slag + `Fly Ash` + Water + SP +
`Coarse Aggregate` + `Fine Aggregate`,
    data = data
  )
summary(fit)

##
## Call:
## lm(formula = `28-day Compressive Strength` ~ Cement + Slag +
##     `Fly Ash` + Water + SP + `Coarse Aggregate` + `Fine Aggregate`,
##     data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.8411 -1.7063 -0.2831  1.2986  7.9424
##
## Coefficients:
```

```
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         139.78150   71.10128   1.966  0.05222 .
## Cement                0.06141    0.02282   2.691  0.00842 **
## Slag                 -0.02971    0.03176  -0.935  0.35200
## `Fly Ash`             0.05053    0.02316   2.182  0.03159 *
## Water                -0.23270    0.07166  -3.247  0.00161 **
## SP                    0.10315    0.13459   0.766  0.44532
## `Coarse Aggregate`   -0.05562    0.02744  -2.027  0.04546 *
## `Fine Aggregate`     -0.03908    0.02882  -1.356  0.17833
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.609 on 95 degrees of freedom
## Multiple R-squared:  0.8968, Adjusted R-squared:  0.8892
## F-statistic:   118 on 7 and 95 DF,  p-value: < 2.2e-16
```

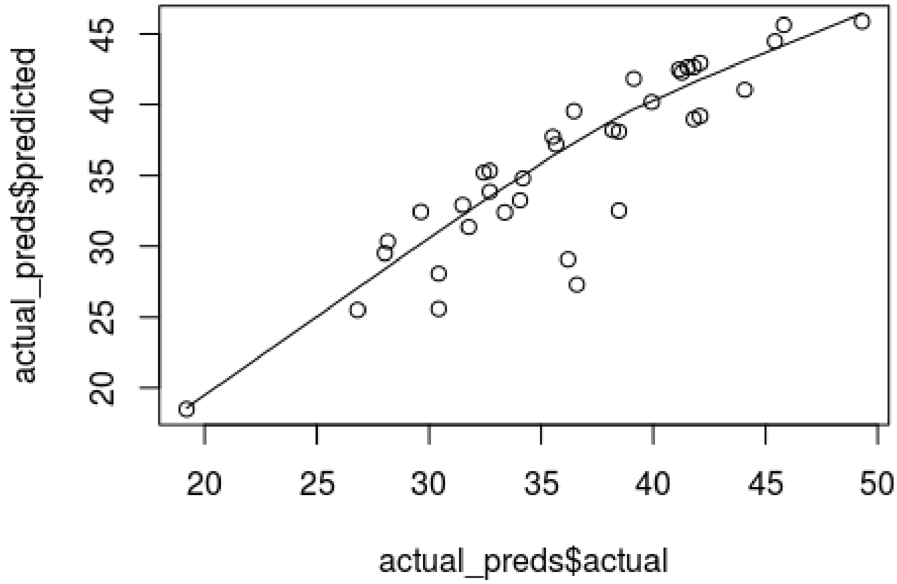
```r
par(mfrow = c(2,2))
plot(fit)
```



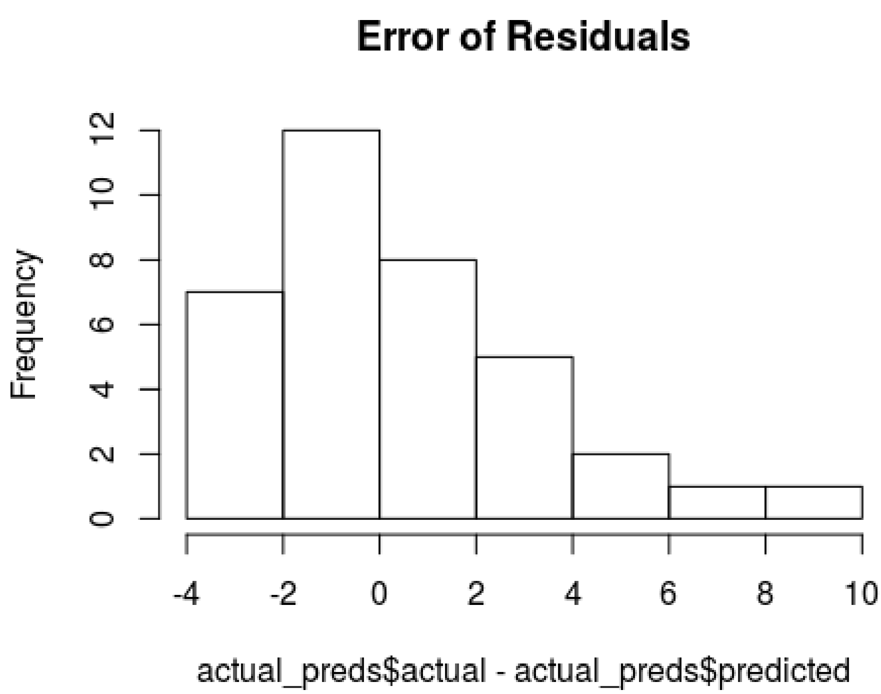
Our Initial Assessment here tells that, Cement, Water and `Coarse Aggregate` are highly correlated to Compressive Strength. As P-values are lower than 0.05. so on this basis we can model the predictions.

```r
# training phase
y_train_fit <-
  lm(
    `28-day Compressive Strength` ~ Cement + Slag + `Fly Ash` + Water  +
```

```r
`Coarse Aggregate` + `Fine Aggregate`,
    data = train
  )
summary(y_train_fit)

##
## Call:
## lm(formula = `28-day Compressive Strength` ~ Cement + Slag +
##     `Fly Ash` + Water + `Coarse Aggregate` + `Fine Aggregate`,
##     data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.5231 -1.2428 -0.2829  1.6071  7.2467
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        179.93063   64.41657   2.793  0.00699 **
## Cement               0.05634    0.02064   2.729  0.00833 **
## Slag                -0.03861    0.02987  -1.293  0.20106
## `Fly Ash`            0.03769    0.02119   1.779  0.08036 .
## Water               -0.28927    0.06537  -4.425 4.13e-05 ***
## `Coarse Aggregate`  -0.06817    0.02506  -2.720  0.00852 **
## `Fine Aggregate`    -0.05723    0.02713  -2.109  0.03910 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.6 on 60 degrees of freedom
## Multiple R-squared:  0.917,  Adjusted R-squared:  0.9087
## F-statistic: 110.5 on 6 and 60 DF,  p-value: < 2.2e-16

# testing phase
y_pred <- predict(y_train_fit, test)


actual_preds <-
  data.frame(cbind(
    actual = test$`28-day Compressive Strength`,
    predicted = y_pred
  ))
cor(actual_preds$actual, actual_preds$predicted)

## [1] 0.8958828

plot(actual_preds$actual, actual_preds$predicted)
lines(lowess(actual_preds$actual, actual_preds$predicted))
```




```
hist(actual_preds$actual - actual_preds$predicted,main='Error of Residuals')
```




```
accuracy(y_pred,test$`28-day Compressive Strength`)
```

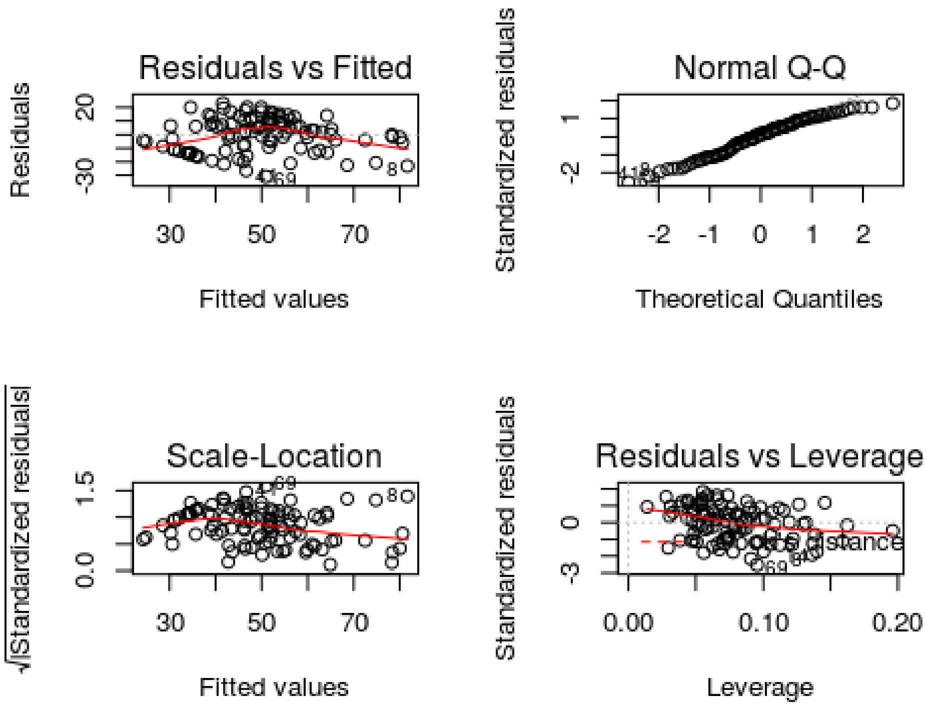
```
##                   ME     RMSE      MAE      MPE     MAPE
## Test set 0.4856471 2.916888 2.149264 1.181612 6.052467
```

Here we see, that the correlation between $\hat{y}$ and y i.e (y predicted and y actual) after the testing is still high (r = 0.89). with R^2 value's to be 0.91 and Adjusted R^2 to be 0.90.

The model fit's well for Compressive strength.

For Slump Flow

```
fit <-
  lm(
    `Slump Flow` ~ Cement + Slag + `Fly Ash` + Water + SP + `Coarse
Aggregate` + `Fine Aggregate`,
    data = data
  )
summary(fit)

##
## Call:
## lm(formula = `Slump Flow` ~ Cement + Slag + `Fly Ash` + Water +
##     SP + `Coarse Aggregate` + `Fine Aggregate`, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -30.880 -10.428   1.815   9.601  22.953
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -252.87467  350.06649  -0.722   0.4718
## Cement                 0.05364    0.11236   0.477   0.6342
## Slag                  -0.00569    0.15638  -0.036   0.9710
## `Fly Ash`              0.06115    0.11402   0.536   0.5930
## Water                  0.73180    0.35282   2.074   0.0408 *
## SP                     0.29833    0.66263   0.450   0.6536
## `Coarse Aggregate`     0.07366    0.13510   0.545   0.5869
## `Fine Aggregate`       0.09402    0.14191   0.663   0.5092
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.84 on 95 degrees of freedom
## Multiple R-squared:  0.5022, Adjusted R-squared:  0.4656
## F-statistic: 13.69 on 7 and 95 DF,  p-value: 3.915e-12

par(mfrow = c(2,2))
plot(fit)
```

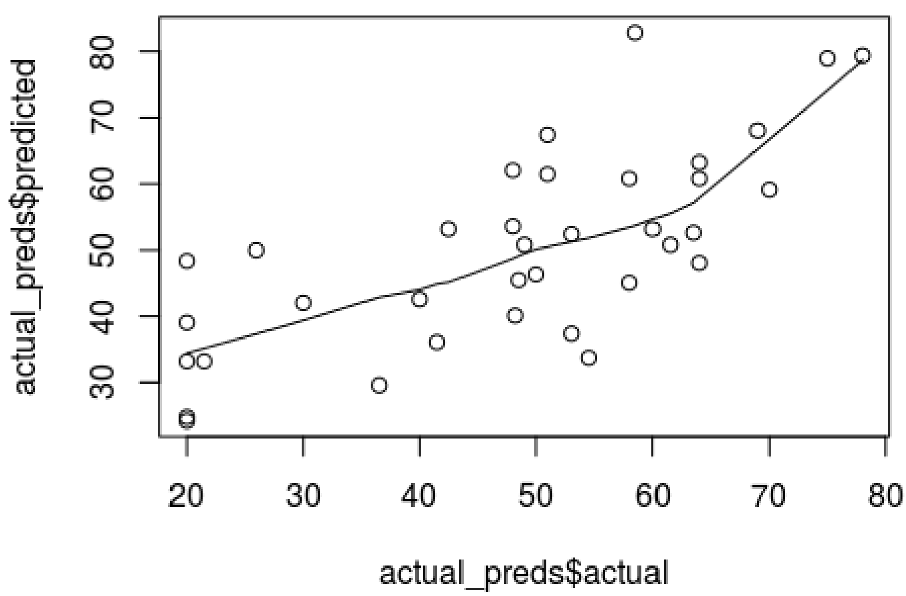


For Slump Flow, The initial Assessment tells only water is correlated to the output rest are not correlated.
let's train our model on this variables.

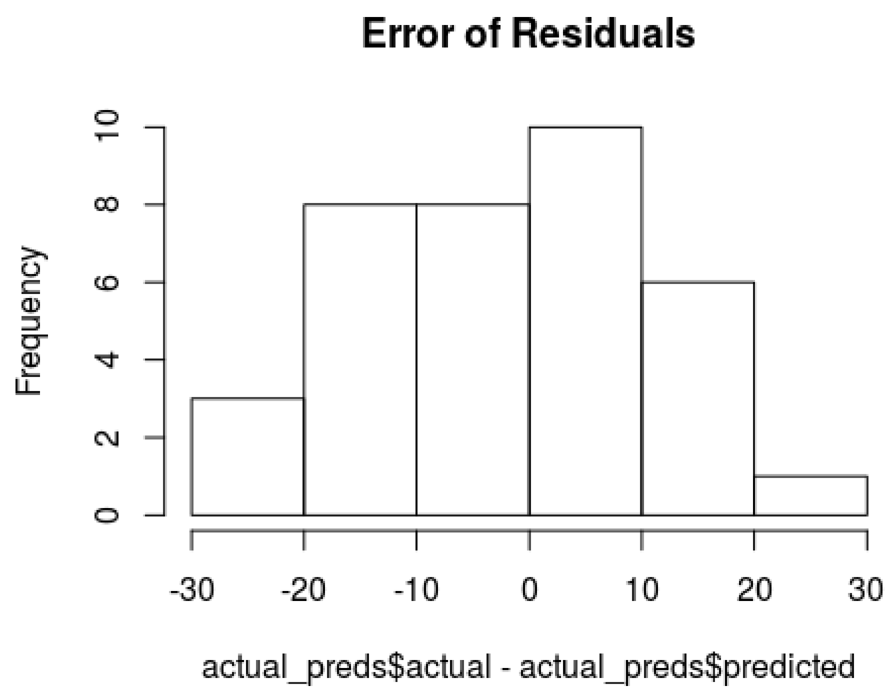
```
# training phase
y_train_fit <-
  lm(
    `Slump Flow` ~ Slag  + Water,
    data = train
  )
summary(y_train_fit)
```

```
##
## Call:
## lm(formula = `Slump Flow` ~ Slag + Water, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -34.083 -10.531   1.911   9.446  23.901
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -59.63077   15.61159  -3.820 0.000305 ***
## Slag         -0.09086    0.02757  -3.296 0.001603 **
## Water         0.59361    0.07915   7.499 2.46e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 12.98 on 64 degrees of freedom
## Multiple R-squared:  0.4956, Adjusted R-squared:  0.4798
## F-statistic: 31.44 on 2 and 64 DF,  p-value: 3.081e-10

# testing phase
y_pred <- predict(y_train_fit, test)


actual_preds <-
  data.frame(cbind(
    actual = test$`Slump Flow`,
    predicted = y_pred
  ))
cor(actual_preds$actual, actual_preds$predicted)

## [1] 0.7158847

plot(actual_preds$actual, actual_preds$predicted)
lines(lowess(actual_preds$actual, actual_preds$predicted))
```




```
hist(actual_preds$actual - actual_preds$predicted,main='Error of Residuals')
```



After training we find that the correlation between the test output and predicted outupts is 71%. Our model worked just fair enough.

6

Selection of predictors

based on the model fitting, we can reduce the dimensions and take only the variables which gives maximum accuracy and with minimum dimensions.
For Slump,

```
# training phase
y_train_fit <-
  lm(
    `Slump` ~ Slag +  Water + SP,
    data = train
  )
summary(y_train_fit)

##
## Call:
## lm(formula = Slump ~ Slag + Water + SP, data = train)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -18.138  -5.253   3.492   5.225  12.089
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -20.87882   10.22967  -2.041   0.0454 *
## Slag         -0.03898    0.01644  -2.371   0.0208 *
## Water         0.22403    0.04722   4.744 1.24e-05 ***
## SP           -0.21711    0.36820  -0.590   0.5575
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.603 on 63 degrees of freedom
## Multiple R-squared:  0.3152, Adjusted R-squared:  0.2826
## F-statistic: 9.665 on 3 and 63 DF,  p-value: 2.457e-05
```

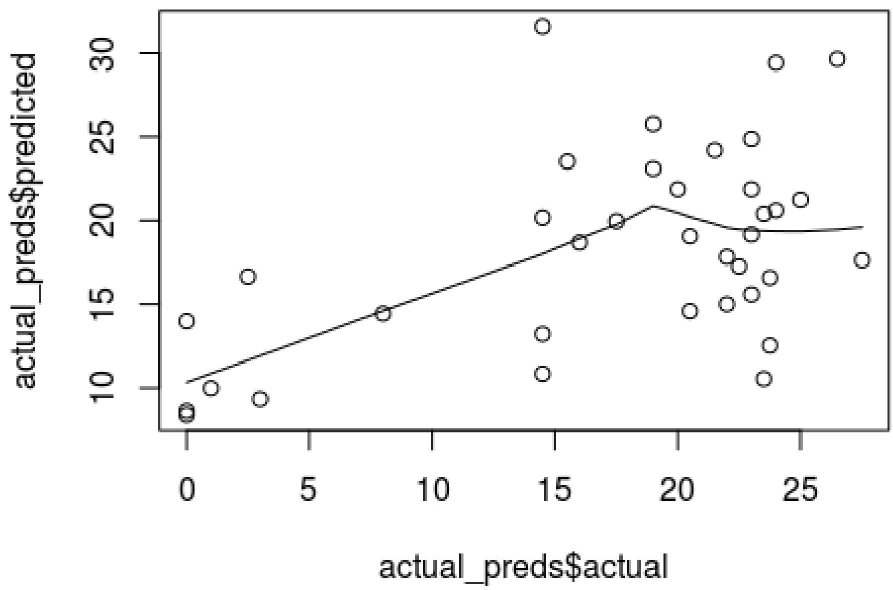
```
# testing phase
y_pred <- predict(y_train_fit, test)


actual_preds <-
  data.frame(cbind(
    actual = test$Slump,
    predicted = y_pred
  ))
cor(actual_preds$actual, actual_preds$predicted)
```

```
## [1] 0.522426
```

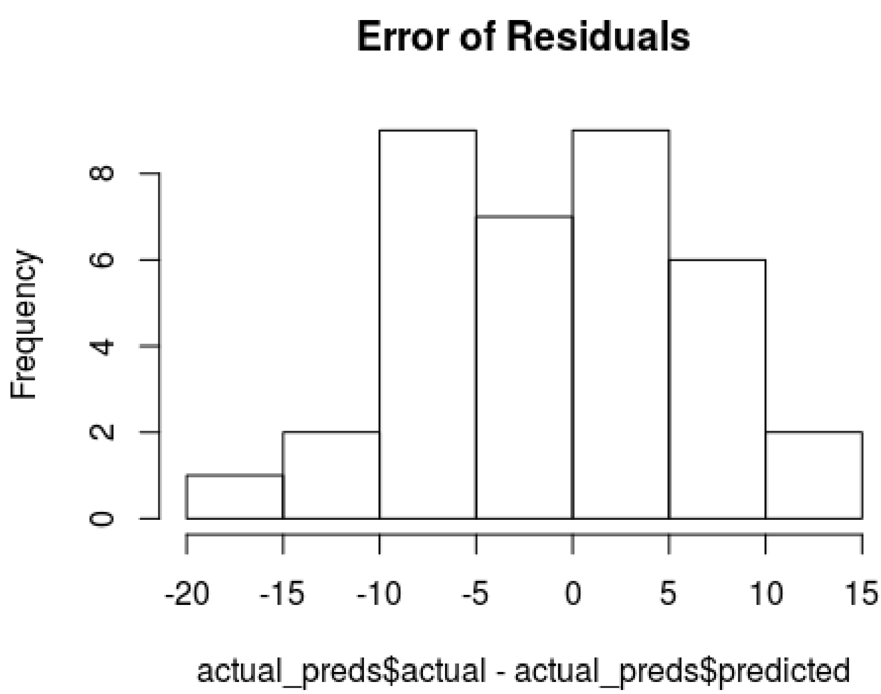
```
accuracy(y_pred, test$Slump)
```

```
##                   ME     RMSE      MAE  MPE MAPE
## Test set -1.001377 7.293248 6.144881 -Inf  Inf
```

```
plot(actual_preds$actual, actual_preds$predicted)
lines(lowess(actual_preds$actual, actual_preds$predicted))
```

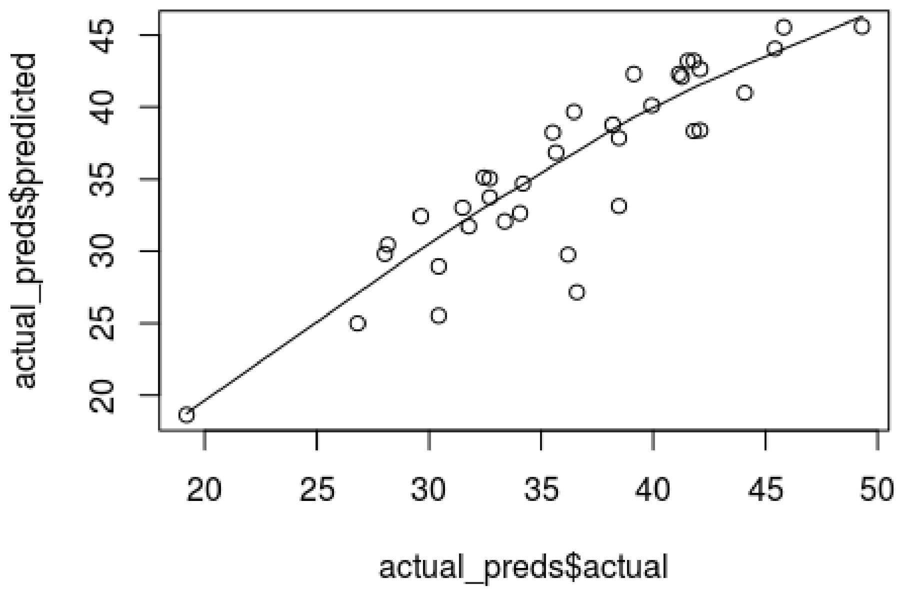



```r
hist(actual_preds$actual - actual_preds$predicted,main='Error of Residuals')
```

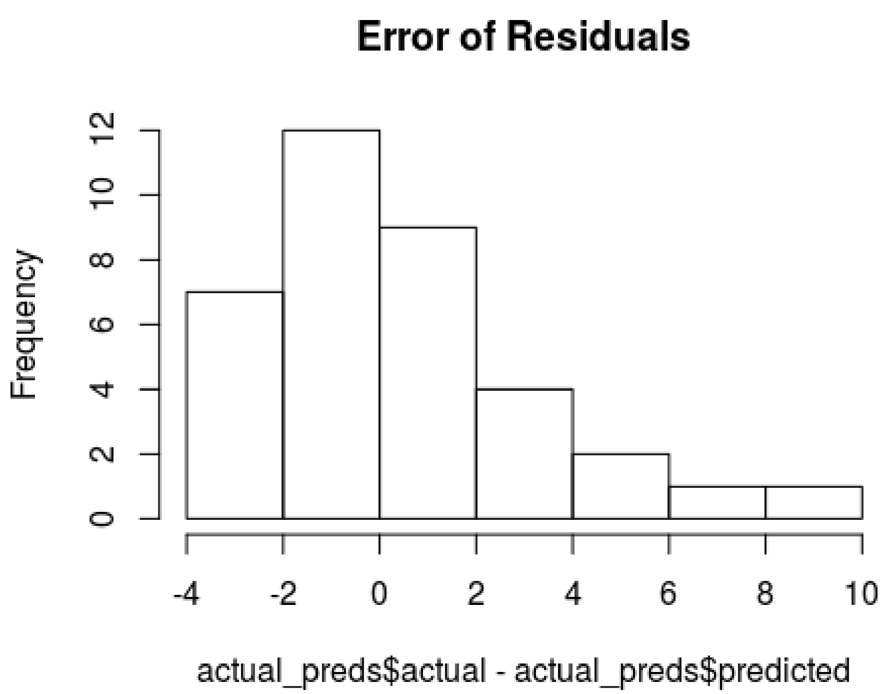


For Compressive Strength,

```r
# training phase
y_train_fit <-
  lm(`28-day Compressive Strength` ~ Cement  + `Fly Ash` + Water  + `Coarse
Aggregate` + `Fine Aggregate`,
     data = train)
summary(y_train_fit)

##
## Call:
## lm(formula = `28-day Compressive Strength` ~ Cement + `Fly Ash` +
##     Water + `Coarse Aggregate` + `Fine Aggregate`, data = train)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.954 -1.401 -0.259  1.189  7.250
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          98.677083  14.176468   6.961 2.71e-09 ***
## Cement                0.082211   0.005081  16.180  < 2e-16 ***
## `Fly Ash`             0.064370   0.004816  13.366  < 2e-16 ***
## Water                -0.210279   0.023355  -9.004 8.44e-13 ***
## `Coarse Aggregate`   -0.036683   0.005937  -6.179 5.85e-08 ***
## `Fine Aggregate`     -0.023541   0.007595  -3.100  0.00293 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.614 on 61 degrees of freedom
## Multiple R-squared:  0.9147, Adjusted R-squared:  0.9077
## F-statistic: 130.8 on 5 and 61 DF,  p-value: < 2.2e-16

# testing phase
y_pred <- predict(y_train_fit, test)


actual_preds <-
  data.frame(cbind(
    actual = test$`28-day Compressive Strength`,
    predicted = y_pred
  ))
cor(actual_preds$actual, actual_preds$predicted)

## [1] 0.8923297

accuracy(y_pred, test$`28-day Compressive Strength`)

##                 ME     RMSE      MAE      MPE     MAPE
## Test set 0.4850205 2.948567 2.236556 1.146666 6.255303

plot(actual_preds$actual, actual_preds$predicted)
lines(lowess(actual_preds$actual, actual_preds$predicted))
```



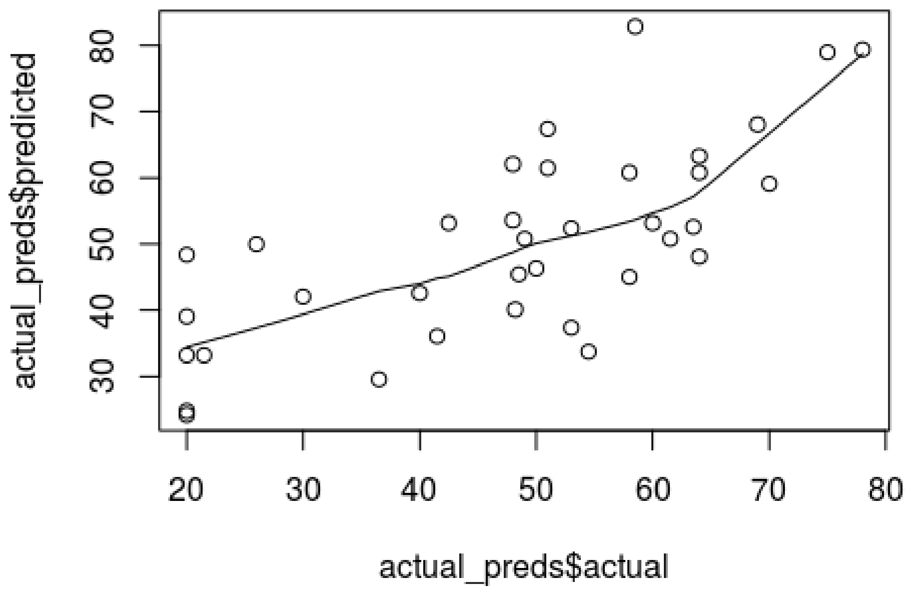

```
hist(actual_preds$actual - actual_preds$predicted,main='Error of Residuals')
```



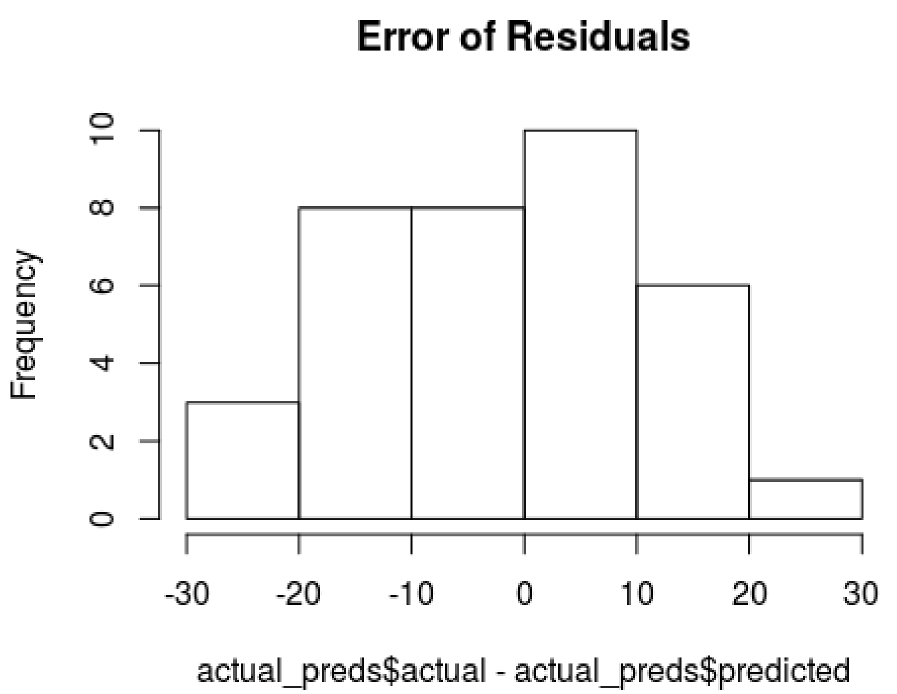
For Slump Flow, Only Water and Slag is the predictor Significantly correlated to Slump Flow.

We can model and reduce the noise in the model

```
# training phase
y_train_fit <-
  lm(`Slump Flow` ~ Slag + Water,
     data = train)
summary(y_train_fit)

##
## Call:
## lm(formula = `Slump Flow` ~ Slag + Water, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -34.083 -10.531   1.911   9.446  23.901
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -59.63077   15.61159  -3.820 0.000305 ***
## Slag         -0.09086    0.02757  -3.296 0.001603 **
## Water         0.59361    0.07915   7.499 2.46e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.98 on 64 degrees of freedom
## Multiple R-squared:  0.4956, Adjusted R-squared:  0.4798
## F-statistic: 31.44 on 2 and 64 DF,  p-value: 3.081e-10

# testing phase
y_pred <- predict(y_train_fit, test)


actual_preds <-
  data.frame(cbind(actual = test$`Slump Flow`,
                   predicted = y_pred))
cor(actual_preds$actual, actual_preds$predicted)

## [1] 0.7158847

plot(actual_preds$actual, actual_preds$predicted)
lines(lowess(actual_preds$actual, actual_preds$predicted))
```




```
hist(actual_preds$actual - actual_preds$predicted, main='Error of Residuals')
```




```
accuracy(y_pred,test$`Slump Flow`)
```

```
##                 ME     RMSE      MAE       MPE     MAPE
## Test set -2.047652 12.08842 9.683817 -13.01586 26.78364
```

The best model is:

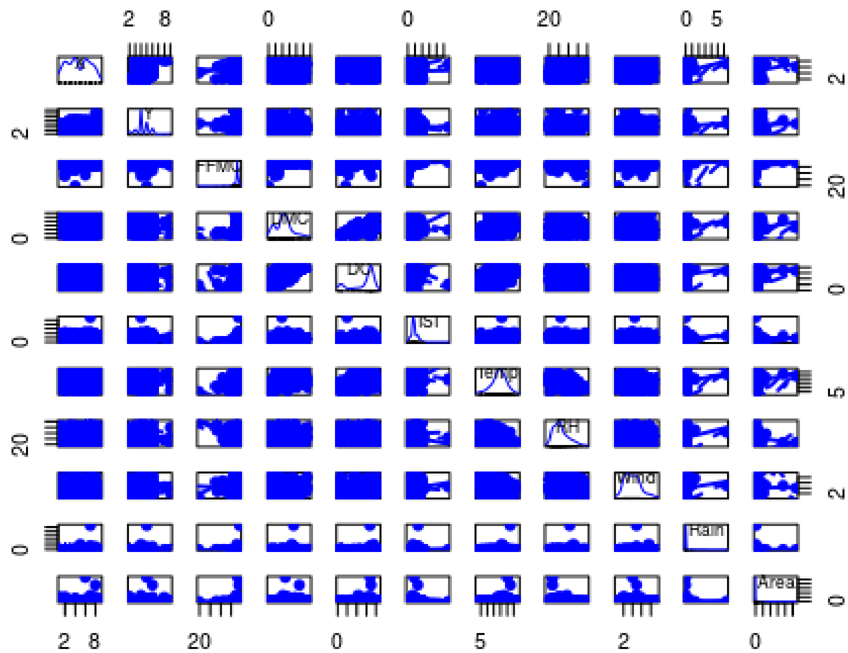
SlumpFlow = -59.63 + 0.5936 x Water + -0.0908 x Slag CompressiveStrength = 98.67 + 0.08 x Cement + 0.064 x `Fly Ash` -0.210 x Water - 0.036 x `Coarse Aggregate` -0.023541 x `Fine Aggregate` Slump = -20.87 + -0.038 x Slag + 0.22 x Water - 0.21 x SP

Problem-2

```
##### Loading the data

forest_fires<-read_xlsx('Forest Fires Data.xlsx')

### Let us convert the character to numerical values

forest_fires$Month <- as.numeric(as.factor(forest_fires$Month))
forest_fires$Day <- as.numeric(as.factor(forest_fires$Day))

head(forest_fires)

## # A tibble: 6 x 13
##       X     Y Month   Day  FFMC   DMC    DC   ISI  Temp    RH  Wind  Rain
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     7     5     8     1  86.2  26.2  94.3   5.1   8.2    51   6.7   0
## 2     7     4    11     6  90.6  35.4 669.    6.7  18      33   0.9   0
## 3     7     4    11     3  90.6  43.7 687.    6.7  14.6    33   1.3   0
## 4     8     6     8     1  91.7  33.3  77.5   9     8.3    97   4     0.2
## 5     8     6     8     4  89.3  51.3 102.    9.6  11.4    99   1.8   0
## 6     8     6     2     4  92.3  85.3 488    14.7  22.2    29   5.4   0
## # … with 1 more variable: Area <dbl>

glimpse(forest_fires)

## Observations: 517
## Variables: 13
## $ X     <dbl> 7, 7, 7, 8, 8, 8, 8, 8, 8, 7, 7, 7, 6, 6, 6, 6, 5, 8, 6, 6…
## $ Y     <dbl> 5, 4, 4, 6, 6, 6, 6, 6, 6, 5, 5, 5, 5, 5, 5, 5, 5, 5, 4, 4…
## $ Month <dbl> 8, 11, 11, 8, 8, 2, 2, 2, 12, 12, 12, 12, 2, 12, 12, 12, 8…
## $ Day   <dbl> 1, 6, 3, 1, 4, 4, 2, 2, 6, 3, 3, 3, 1, 2, 7, 1, 3, 2, 7, 3…
## $ FFMC  <dbl> 86.2, 90.6, 90.6, 91.7, 89.3, 92.3, 92.3, 91.5, 91.0, 92.5…
## $ DMC   <dbl> 26.2, 35.4, 43.7, 33.3, 51.3, 85.3, 88.9, 145.4, 129.5, 88…
## $ DC    <dbl> 94.3, 669.1, 686.9, 77.5, 102.2, 488.0, 495.6, 608.2, 692.…
## $ ISI   <dbl> 5.1, 6.7, 6.7, 9.0, 9.6, 14.7, 8.5, 10.7, 7.0, 7.1, 7.1, 2…
## $ Temp  <dbl> 8.2, 18.0, 14.6, 8.3, 11.4, 22.2, 24.1, 8.0, 13.1, 22.8, 1…
## $ RH    <dbl> 51, 33, 33, 97, 99, 29, 27, 86, 63, 40, 51, 38, 72, 42, 21…
## $ Wind  <dbl> 6.7, 0.9, 1.3, 4.0, 1.8, 5.4, 3.1, 2.2, 5.4, 4.0, 7.2, 4.0…
## $ Rain  <dbl> 0.0, 0.0, 0.0, 0.2, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0…
## $ Area  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0…
```

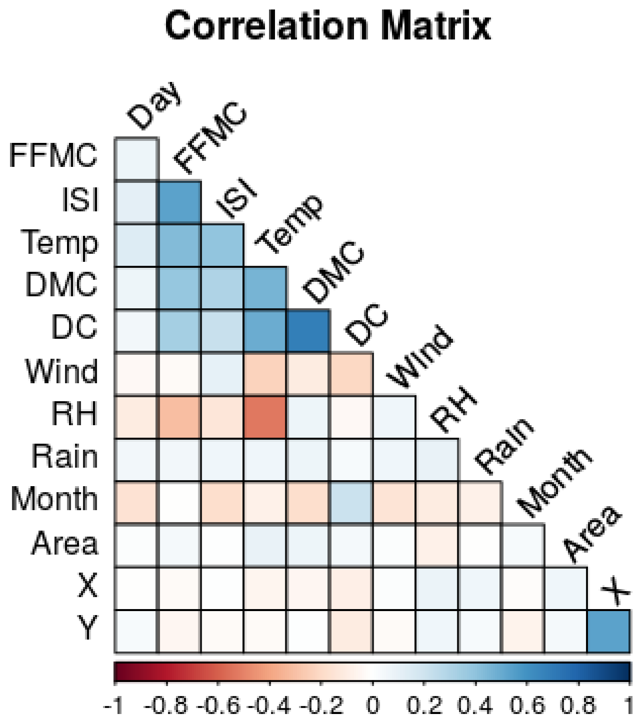
1. Create a scatterplot matrix of “Forest Fire Data” and select an initial set of predictor variables

```
##### Creating a scatter plot matrix
scatter<-forest_fires[,!(colnames(forest_fires)==c('Month','Day'))]
scatter%>%scatterplotMatrix(pch=19)
```



Scatter plot matrix helps us to find linear realationship between variables.But it is very diffcult to interpret from the plot so let us use correlation function to view

```
par(mfrow=c(1,1))
M <- cor(forest_fires)
corrplot(M, method="color", outline = TRUE,type="lower",order = "hclust",
         tl.col="black", tl.srt=45, diag=FALSE,tl.cex = 1,mar=c(0,0,3,0),
         title="Correlation Matrix")
```



The plot indicates postive correlation betweeen ISI,temp,dc and dcm. The correlation of variables will give proper interpretation.

```
cor(scatter[,])

##                X            Y        FFMC          DMC          DC
## X     1.000000000  0.539548171 -0.02103927 -0.048384178 -0.08591612
## Y     0.539548171  1.000000000 -0.04630755  0.007781561 -0.10117777
## FFMC -0.021039272 -0.046307546  1.00000000  0.382618800  0.33051180
## DMC  -0.048384178  0.007781561  0.38261880  1.000000000  0.68219161
## DC   -0.085916123 -0.101177767  0.33051180  0.682191612  1.00000000
## ISI   0.006209941 -0.024487992  0.53180493  0.305127835  0.22915417
## Temp -0.051258262 -0.024103084  0.43153226  0.469593844  0.49620805
## RH    0.085223194  0.062220731 -0.30099542  0.073794941 -0.03919165
## Wind  0.018797818 -0.020340852 -0.02848481 -0.105342253 -0.20346569
## Rain  0.065387168  0.033234103  0.05670153  0.074789982  0.03586086
## Area  0.063385299  0.044873225  0.04012200  0.072994296  0.04938323
##              ISI        Temp          RH        Wind        Rain
## X     0.006209941 -0.05125826  0.08522319  0.01879782  0.065387168
## Y    -0.024487992 -0.02410308  0.06222073 -0.02034085  0.033234103
## FFMC  0.531804931  0.43153226 -0.30099542 -0.02848481  0.056701533
## DMC   0.305127835  0.46959384  0.07379494 -0.10534225  0.074789982
## DC    0.229154169  0.49620805 -0.03919165 -0.20346569  0.035860862
## ISI   1.000000000  0.39428710 -0.13251718  0.10682589  0.067668190
## Temp  0.394287104  1.00000000 -0.52739034 -0.22711622  0.069490547
## RH   -0.132517177 -0.52739034  1.00000000  0.06941007  0.099751223
## Wind  0.106825888 -0.22711622  0.06941007  1.00000000  0.061118880
```

```
## Rain  0.067668190  0.06949055  0.09975122  0.06111888  1.000000000
## Area  0.008257688  0.09784411 -0.07551856  0.01231728 -0.007365729
##              Area
## X      0.063385299
## Y      0.044873225
## FFMC   0.040122004
## DMC    0.072994296
## DC     0.049383225
## ISI    0.008257688
## Temp   0.097844107
## RH    -0.075518563
## Wind   0.012317277
## Rain  -0.007365729
## Area   1.000000000
```

According to the description document there are 12 predictor variables and one response variables. 00.68 of DMC indicates it is having linear correlation with DC, so removing it might help in reducing multi-colinearity.

```
summary(forest_fires)
```

```
##        X               Y            Month            Day       
##  Min.   :1.000   Min.   :2.0   Min.   : 1.000   Min.   :1.000  
##  1st Qu.:3.000   1st Qu.:4.0   1st Qu.: 2.000   1st Qu.:2.000  
##  Median :4.000   Median :4.0   Median : 7.000   Median :4.000  
##  Mean   :4.669   Mean   :4.3   Mean   : 6.758   Mean   :3.737  
##  3rd Qu.:7.000   3rd Qu.:5.0   3rd Qu.:12.000   3rd Qu.:5.000  
##  Max.   :9.000   Max.   :9.0   Max.   :12.000   Max.   :7.000  
##       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  
```

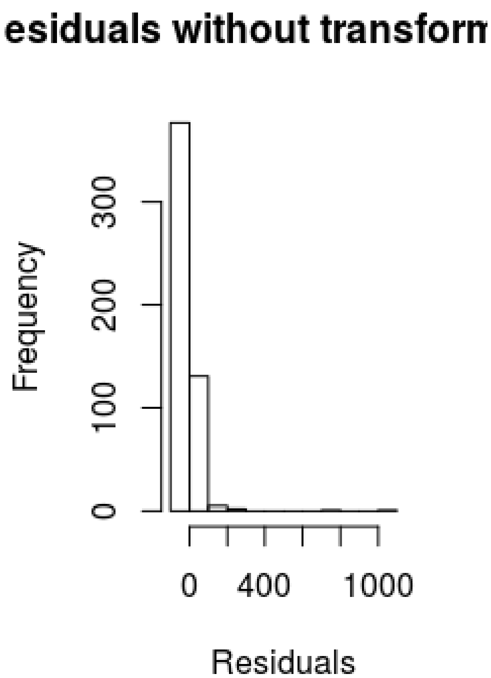
2. Build a few potential regression models using “Forest Fire Data”

Building some potential regression models, Let us first try without transformation

```
fit<-lm(`Area`~.,data=forest_fires)
summary(fit)

##
## Call:
## lm(formula = Area ~ ., data = forest_fires)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
##  -41.44  -16.47   -8.63    0.63 1061.73
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -12.97338   63.51663  -0.204    0.838
## X             1.88124    1.44965   1.298    0.195
## Y             0.52680    2.73989   0.192    0.848
## Month         0.97328    0.77592   1.254    0.210
## Day           0.49953    1.49700   0.334    0.739
## FFMC         -0.10740    0.66367  -0.162    0.872
## DMC           0.10980    0.07205   1.524    0.128
## DC           -0.01463    0.01878  -0.779    0.437
## ISI          -0.61081    0.77907  -0.784    0.433
## Temp          0.98013    0.80213   1.222    0.222
## RH           -0.18492    0.24027  -0.770    0.442
## Wind          1.78229    1.68034   1.061    0.289
## Rain         -3.25171    9.69858  -0.335    0.738
##
## Residual standard error: 63.61 on 504 degrees of freedom
## Multiple R-squared:  0.02472,    Adjusted R-squared:  0.001503
## F-statistic: 1.065 on 12 and 504 DF,  p-value: 0.3881
```

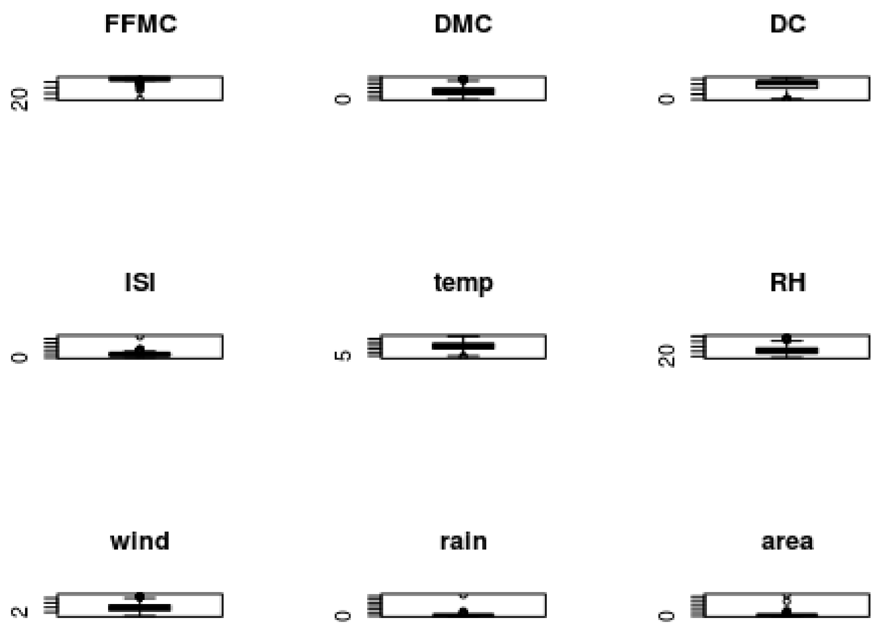
The R-square of 0.02472 indicates that the model does not perform well and adjusted R squared indicates that the model does not fit well.

```
par(mfrow=c(1,2))
hist(fit$residuals, main = "Residuals without transformation", xlab =
'Residuals')
```



The Residuals are skewed towards we should transform it.

```
par(mfrow=c(3,3))
boxplot(forest_fires$FFMC, main='FFMC') #outliers
boxplot(forest_fires$DMC, main ='DMC') # outliers
boxplot(forest_fires$DC, main='DC') # some outliers
boxplot(forest_fires$ISI,main='ISI') # outliers
boxplot(forest_fires$Temp, main='temp')
boxplot(forest_fires$RH,main="RH") # outliers
boxplot(forest_fires$Wind, main='wind') #
boxplot(forest_fires$Rain, main='rain')  # heavy outliers...high variability
in data
boxplot(forest_fires$Area, main='area') # heavy outliers..high variability in
data
```



Let us try with transformed data.

```
transform<-function(x){
  y=(x+1)
  return (log(y))
}
```

Removing Skeweness in data

```
forest_fires$Area<-transform(forest_fires$Area)
forest_fires$ISI<-transform(forest_fires$ISI)
head(forest_fires)

## # A tibble: 6 x 13
##       X     Y Month   Day  FFMC   DMC    DC   ISI  Temp    RH  Wind  Rain
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     7     5     8     1  86.2  26.2  94.3  1.81   8.2    51   6.7   0
## 2     7     4    11     6  90.6  35.4 669.   2.04  18      33   0.9   0
## 3     7     4    11     3  90.6  43.7 687.   2.04  14.6    33   1.3   0
## 4     8     6     8     1  91.7  33.3  77.5  2.30   8.3    97   4     0.2
## 5     8     6     8     4  89.3  51.3 102.   2.36  11.4    99   1.8   0
## 6     8     6     2     4  92.3  85.3 488    2.75  22.2    29   5.4   0
## # … with 1 more variable: Area <dbl>
```

Fitting with transformed data

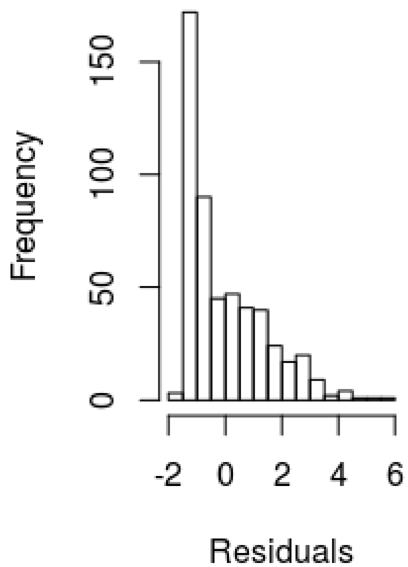
```r
transform_fit<-lm(`Area`~Month + Day + FFMC + DMC + DC + ISI + Temp + RH +
Wind,data=forest_fires)
summary(transform_fit)
```

```
##
## Call:
## lm(formula = Area ~ Month + Day + FFMC + DMC + DC + ISI + Temp +
##     RH + Wind, data = forest_fires)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7252 -1.1004 -0.6167  0.8437  5.6605
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.985e-01  1.506e+00  -0.331   0.7408
## Month        1.866e-02  1.694e-02   1.101   0.2714
## Day          2.278e-02  3.273e-02   0.696   0.4866
## FFMC         1.696e-02  1.845e-02   0.920   0.3582
## DMC          1.843e-03  1.570e-03   1.174   0.2409
## DC           7.657e-05  4.079e-04   0.188   0.8512
## ISI         -3.069e-01  2.164e-01  -1.418   0.1567
## Temp         6.378e-03  1.734e-02   0.368   0.7131
## RH          -3.625e-03  5.189e-03  -0.699   0.4851
## Wind         8.233e-02  3.669e-02   2.244   0.0253 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.395 on 507 degrees of freedom
## Multiple R-squared:  0.02271,    Adjusted R-squared:  0.005364
## F-statistic: 1.309 on 9 and 507 DF,  p-value: 0.2291
```

The transformed data has accuracy by 0.0226, this model is better than the previous model. F statistics of 1.309 and p values indicates that the predictor variables are not significant for predicting the response variable.

```r
par(mfrow=c(1,2))
hist(transform_fit$residuals, main = "Residuals with transformation", xlab =
'Residuals')
```

Residuals with transforma



This indicates transformation has helped in much better distribution of residuals.

Let us try fitting the model with polynomial regression.

```
poly_fit<-lm(Area~ X+Y+Month + Day + FFMC + DMC^2 + DC^2 + ISI^3 + Temp + RH
+ Wind,data=forest_fires)
summary(poly_fit)

##
## Call:
## lm(formula = Area ~ X + Y + Month + Day + FFMC + DMC^2 + DC^2 +
##     ISI^3 + Temp + RH + Wind, data = forest_fires)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7022 -1.0851 -0.5820  0.8949  5.5861
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.6944656  1.5261806  -0.455   0.6493
## X            0.0402015  0.0317124   1.268   0.2055
## Y            0.0154926  0.0600066   0.258   0.7964
## Month        0.0179267  0.0169367   1.058   0.2904
## Day          0.0222735  0.0327264   0.681   0.4964
## FFMC         0.0168540  0.0184450   0.914   0.3613
## DMC          0.0018312  0.0015775   1.161   0.2463
## DC           0.0001406  0.0004117   0.342   0.7328
```

```
## ISI         -0.3128065  0.2162224  -1.447   0.1486
## Temp         0.0049953  0.0173447   0.288   0.7735
## RH          -0.0044725  0.0052091  -0.859   0.3910
## Wind         0.0826686  0.0366958   2.253   0.0247 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.394 on 505 degrees of freedom
## Multiple R-squared:  0.02819,    Adjusted R-squared:  0.007017
## F-statistic: 1.332 on 11 and 505 DF,  p-value: 0.2032
```

```r
forestdata<-forest_fires
# Create interactive terms for Fire index
forestdata$FFMC.DMC <- forestdata$FFMC*forestdata$DMC
forestdata$FFMC.DC <-forestdata$FFMC*forestdata$DC
forestdata$FFMC.ISI <-forestdata$FFMC*forestdata$ISI
forestdata$DMC.DC<-forestdata$DMC*forestdata$DC
forestdata$DMC.ISI<-forestdata$DMC*forestdata$ISI
forestdata$DC.ISI<-forestdata$DC*forestdata$ISI

# Create interactive terms for Weather
forestdata$Wind.Temp<-(forestdata$Wind)*(forestdata$Temp)
forestdata$Temp.RH<-(forestdata$Temp)*(forestdata$RH)
forestdata$Wind.RH<-(forestdata$Wind)*(forestdata$RH)


interact_fit<-lm(`Area`~ .,data=forestdata)

summary(interact_fit)
```

```
##
## Call:
## lm(formula = Area ~ ., data = forestdata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6786 -1.0755 -0.5249  0.8736  5.5424
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.387e+00  2.971e+00  -0.803   0.4221
## X            3.635e-02  3.202e-02   1.135   0.2569
## Y            7.275e-03  6.050e-02   0.120   0.9043
## Month        3.140e-02  1.872e-02   1.677   0.0942 .
## Day          2.503e-02  3.335e-02   0.751   0.4533
## FFMC         9.303e-03  3.440e-02   0.270   0.7869
## DMC         -6.164e-02  4.824e-02  -1.278   0.2019
## DC           3.184e-03  7.662e-03   0.416   0.6779
## ISI          3.257e+00  2.793e+00   1.166   0.2441
## Temp         7.525e-02  4.450e-02   1.691   0.0915 .
```

```
## RH            1.120e-02  1.696e-02   0.661   0.5092
## Wind          3.796e-01  1.733e-01   2.190   0.0290 *
## Rain          1.025e-01  2.207e-01   0.465   0.6425
## FFMC.DMC      7.559e-04  6.082e-04   1.243   0.2145
## FFMC.DC      -3.064e-05  1.048e-04  -0.292   0.7702
## FFMC.ISI     -3.742e-02  2.817e-02  -1.329   0.1846
## DMC.DC       -5.875e-06  7.125e-06  -0.825   0.4100
## DMC.ISI      -4.183e-04  5.184e-03  -0.081   0.9357
## DC.ISI        8.839e-06  1.230e-03   0.007   0.9943
## Wind.Temp    -1.208e-02  6.264e-03  -1.928   0.0544 .
## Temp.RH      -3.101e-04  6.930e-04  -0.448   0.6547
## Wind.RH      -1.941e-03  2.207e-03  -0.880   0.3794
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.396 on 495 degrees of freedom
## Multiple R-squared:  0.04448,    Adjusted R-squared:  0.003939
## F-statistic: 1.097 on 21 and 495 DF,  p-value: 0.3467
```

The polynomial regression accuracy of 0.04448 and the F-statistics and P-value indicates that the overall fit is not significant.

Let us try with interaction fit

```
interaction_fit<-lm(`Area`~ Month + Day + (FFMC + DMC + DC + ISI + Temp + RH
+ Wind)^2 ,data=forest_fires)

summary(interaction_fit)

##
## Call:
## lm(formula = Area ~ Month + Day + (FFMC + DMC + DC + ISI + Temp +
##     RH + Wind)^2, data = forest_fires)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7524 -1.0751 -0.4743  0.8095  5.6000
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.630e+00  2.150e+01   0.169    0.866
## Month        3.754e-02  1.944e-02   1.931    0.054 .
## Day          2.242e-02  3.388e-02   0.662    0.509
## FFMC        -4.128e-02  2.765e-01  -0.149    0.881
## DMC         -3.057e-02  6.892e-02  -0.444    0.658
## DC           2.879e-03  1.128e-02   0.255    0.799
## ISI          2.153e+00  3.331e+00   0.646    0.518
## Temp        -1.169e-01  8.422e-01  -0.139    0.890
## RH          -4.035e-02  1.792e-01  -0.225    0.822
## Wind         9.932e-04  1.065e+00   0.001    0.999
```

```
## FFMC:DMC     4.495e-04  8.526e-04   0.527    0.598
## FFMC:DC     -2.096e-05  1.360e-04  -0.154    0.878
## FFMC:ISI    -2.858e-02  3.512e-02  -0.814    0.416
## FFMC:Temp    8.948e-04  1.072e-02   0.083    0.934
## FFMC:RH      5.678e-04  2.323e-03   0.244    0.807
## FFMC:Wind    2.665e-03  1.328e-02   0.201    0.841
## DMC:DC      -5.426e-06  8.109e-06  -0.669    0.504
## DMC:ISI     -4.317e-04  6.384e-03  -0.068    0.946
## DMC:Temp     7.306e-05  4.994e-04   0.146    0.884
## DMC:RH      -5.560e-05  1.508e-04  -0.369    0.712
## DMC:Wind    -4.373e-04  1.031e-03  -0.424    0.672
## DC:ISI      -7.179e-04  1.354e-03  -0.530    0.596
## DC:Temp      2.579e-05  1.152e-04   0.224    0.823
## DC:RH       -8.084e-06  3.044e-05  -0.266    0.791
## DC:Wind      2.077e-04  2.316e-04   0.897    0.370
## ISI:Temp     3.193e-02  7.784e-02   0.410    0.682
## ISI:RH      -5.395e-04  1.993e-02  -0.027    0.978
## ISI:Wind     3.272e-02  1.350e-01   0.242    0.809
## Temp:RH      2.463e-04  8.111e-04   0.304    0.761
## Temp:Wind   -1.382e-02  9.836e-03  -1.405    0.161
## RH:Wind     -1.186e-03  2.735e-03  -0.434    0.665
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.403 on 486 degrees of freedom
## Multiple R-squared:  0.05208,    Adjusted R-squared:  -0.00643
## F-statistic: 0.8901 on 30 and 486 DF,  p-value: 0.6371
```

The interaction fit has accuracy of 0.05208 percent. Eventhough the F-statistics and p values suggest that the interaction are not significant.

```
forest_fires$FFMC.DMC <- forest_fires$FFMC*forest_fires$DMC
forest_fires$FFMC.DC <-forest_fires$FFMC*forest_fires$DC
forest_fires$FFMC.ISI <-forest_fires$FFMC*forest_fires$ISI
forest_fires$DC.ISI<-forest_fires$DC*forest_fires$ISI
forest_fires$RH_sq<-(forest_fires$RH)^2

mod <- lm(formula = Area ~ X + Y + Month + DMC + DC + FFMC.DMC +
     FFMC.DC + FFMC.ISI + DC.ISI + RH + RH_sq, data = forest_fires)

summary(mod)
```

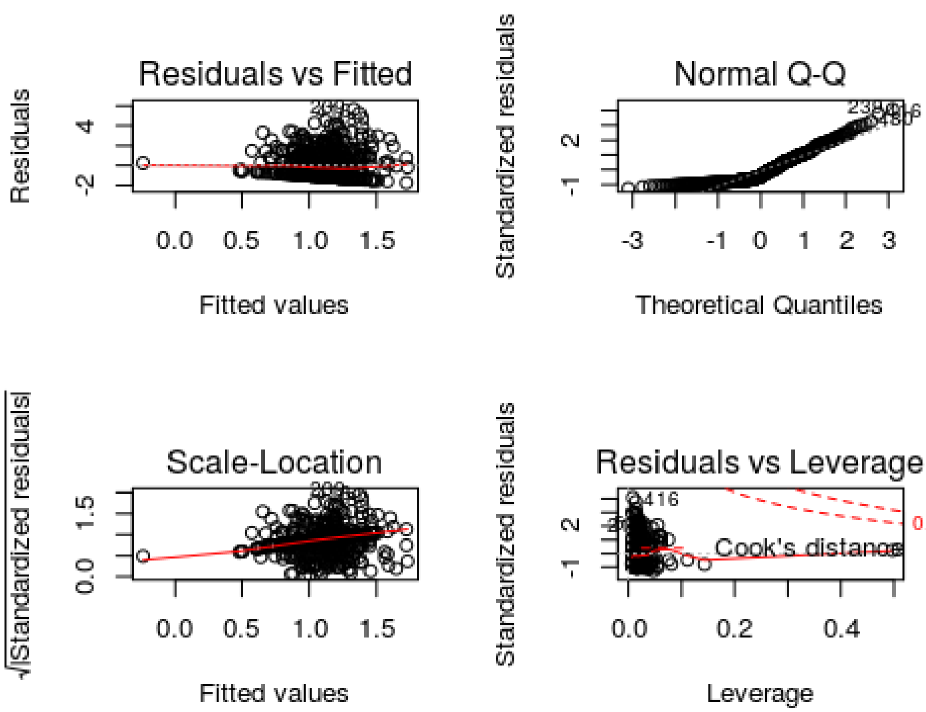
```
##
## Call:
## lm(formula = Area ~ X + Y + Month + DMC + DC + FFMC.DMC + FFMC.DC +
##     FFMC.ISI + DC.ISI + RH + RH_sq, data = forest_fires)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.6753 -1.0582 -0.6549  0.9066  5.6349
```

```
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.674e-01  7.179e-01   1.069    0.286
## X            4.387e-02  3.185e-02   1.378    0.169
## Y            3.237e-03  6.026e-02   0.054    0.957
## Month        1.788e-02  1.723e-02   1.038    0.300
## DMC         -4.161e-02  3.434e-02  -1.212    0.226
## DC           2.570e-03  4.724e-03   0.544    0.587
## FFMC.DMC     4.766e-04  3.765e-04   1.266    0.206
## FFMC.DC     -1.882e-05  5.580e-05  -0.337    0.736
## FFMC.ISI    -9.157e-04  2.878e-03  -0.318    0.750
## DC.ISI      -3.931e-04  6.262e-04  -0.628    0.530
## RH           3.247e-03  1.916e-02   0.169    0.865
## RH_sq       -7.639e-05  1.834e-04  -0.416    0.677
##
## Residual standard error: 1.399 on 505 degrees of freedom
## Multiple R-squared:  0.02101,    Adjusted R-squared:  -0.000315
## F-statistic: 0.9852 on 11 and 505 DF,  p-value: 0.4586
```

3. Perform regression diagnostics using both typical approach and enhanced approach

a) Typical approach

```r
par(mfrow=c(2,2))
plot(transform_fit)
```



The QQ plot indicates that the dependent variables are not normally distributed for given predictor variables with some outliers and it skewed in middle because of large amount of zeros in our data.

The Residuals vs fitted plot indicates that there is no systematic relationship, the staright horizontal line proves that, so there is linear relationship between Response and Predictor Variable.

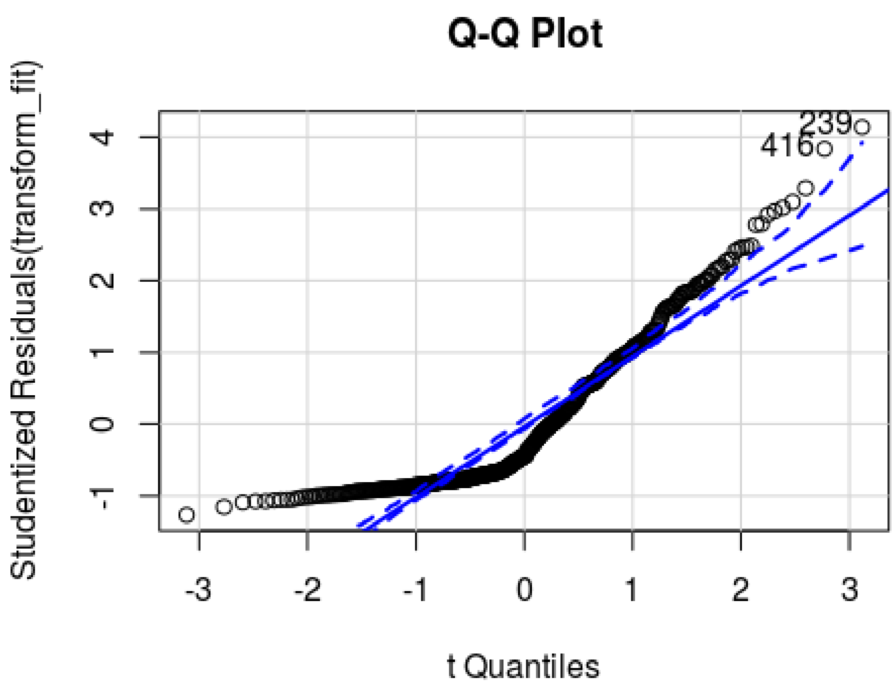
The scale-Location plot gives that there is homoscedasticity(constant variance).

The residual vs Leverage plot indicates some outliers and observation has some high leverage values indicating unusual combination of predictor values.

b) Enhanced Approach

1.Normality

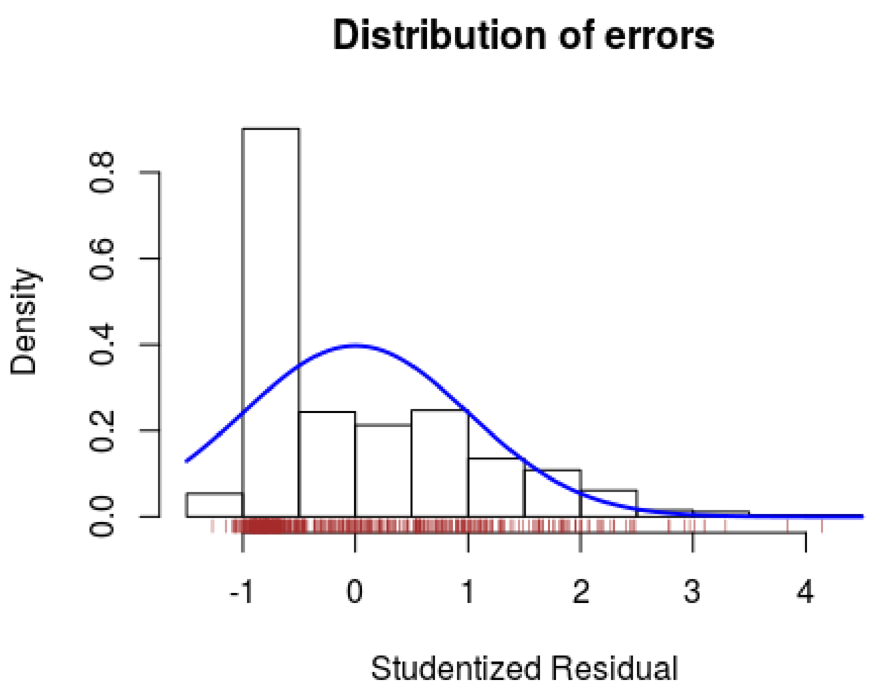
```
qqPlot(transform_fit, labels = row.names(forest_fires), id.method =
"identify", simulate = TRUE, main = "Q-Q Plot")
```




```
## [1] 239 416
```

The QQ plot indicates that the fit does not satisfy the condition of normality.Due to the large sample size, even small deviations from normality will be picked up as significant so normality will be assessed with plots. There also seems to be a heavy skew to the residuals which are not normally distributed.This appears to be due to the large number of 0's in the dataset. When these 0's are removed, we can see the residuals become more normally distributed.Although this means we lose a large chunk of the data cases, this is needed in order to correctly use the lm model.

```
residual<-function(fit,nbreaks=10){
  z<-rstudent(fit)
  hist(z,breaks=nbreaks,freq=FALSE,
  xlab='Studentized Residual',
  main='Distribution of errors')
  rug(jitter(z),col='brown')
  curve(dnorm(x,mean=mean(z),sd=sd(z)),add=T,col='blue',lwd=2)

}

residual(transform_fit)
```



This residual plot indicates that the distribution of residuals are not normaly distributed which confirms with qq plot

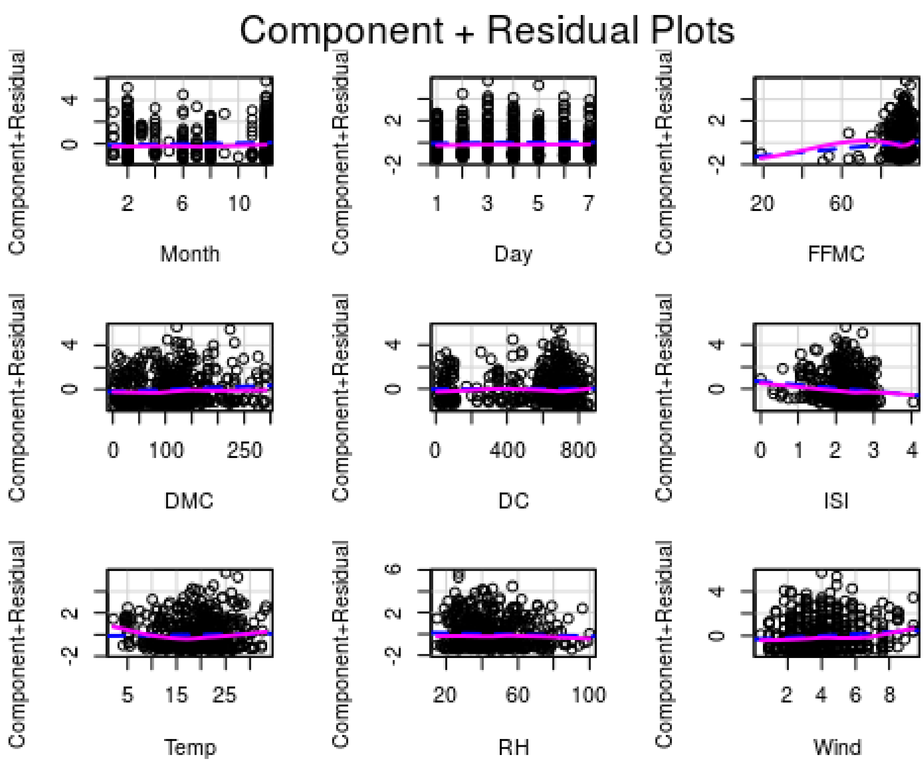
2.Independence of errors

```
durbinWatsonTest(transform_fit)

##  lag Autocorrelation D-W Statistic p-value
##    1       0.5374377     0.9223002       0
##  Alternative hypothesis: rho != 0
```

The Durbin Watson Test indicates that there is some correlation in errors and indicates some correlation in predictor variables.

3. Linearity

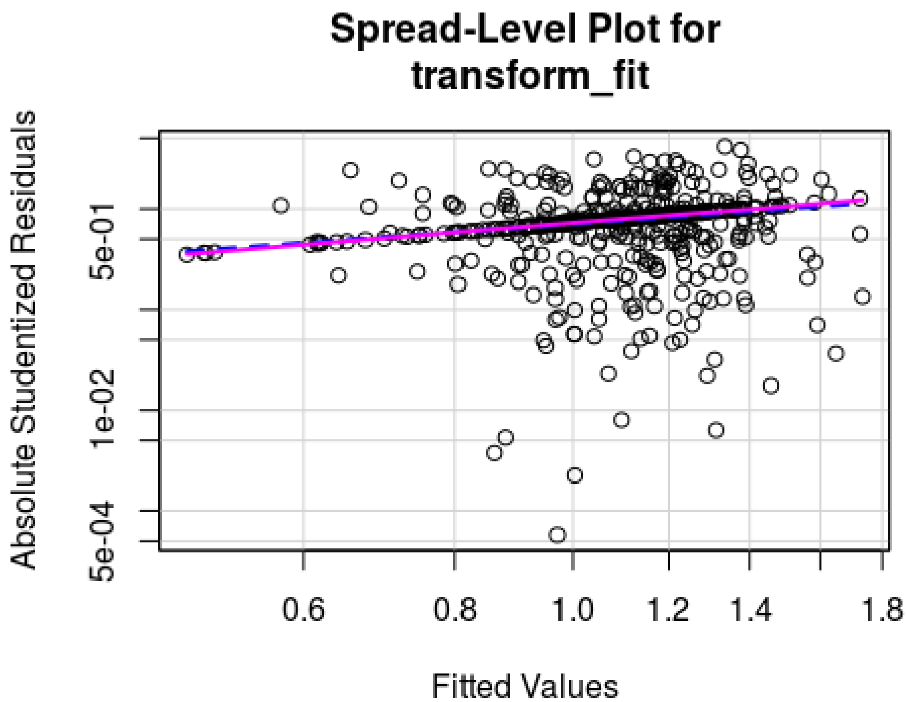
```
crPlots(transform_fit)
```



The plot indicates that there is no systematic relationship, Hence there is linearity between the predictor variables and response variable and this indicates that the property of linearity is satisfied.

4. Homoscedasticity

```
ncvTest(transform_fit)

## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 12.87837, Df = 1, p = 0.0003324

spreadLevelPlot(transform_fit)

## Warning in spreadLevelPlot.lm(transform_fit):
## 1 negative fitted value removed
```




```
##
## Suggested power transformation:  0.1451258
```

The ncv Test indicates that there is constant variance and the horizontal line proves the same.So Homoscedascity is satisfied.

Checking for multicolinearity

```
sqrt(vif(transform_fit))>2

## Month   Day  FFMC   DMC    DC   ISI  Temp    RH  Wind
## FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

This indicates that there is no multicolinearity in independent variables.

d) Identify unusual observations and take corrective measures

1) Outliers

```
outlierTest(transform_fit)

##     rstudent unadjusted p-value Bonferroni p
## 239 4.140671         4.0571e-05     0.020975
```

Removing outliers(according outlier test the 239 row is an outlier)

```
new_forest_fires<-forest_fires[-c(239),]
head(new_forest_fires)
```

```
## # A tibble: 6 x 18
##       X     Y Month   Day  FFMC   DMC    DC   ISI  Temp    RH  Wind  Rain
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     7     5     8     1  86.2  26.2  94.3  1.81   8.2    51   6.7   0
## 2     7     4    11     6  90.6  35.4 669.   2.04  18      33   0.9   0
## 3     7     4    11     3  90.6  43.7 687.   2.04  14.6    33   1.3   0
## 4     8     6     8     1  91.7  33.3  77.5  2.30   8.3    97   4     0.2
## 5     8     6     8     4  89.3  51.3 102.   2.36  11.4    99   1.8   0
## 6     8     6     2     4  92.3  85.3 488    2.75  22.2    29   5.4   0
## # … with 6 more variables: Area <dbl>, FFMC.DMC <dbl>, FFMC.DC <dbl>,
## #   FFMC.ISI <dbl>, DC.ISI <dbl>, RH_sq <dbl>
```

```
removing_outlier<-lm(Area~.,data=new_forest_fires)
summary(removing_outlier)
```

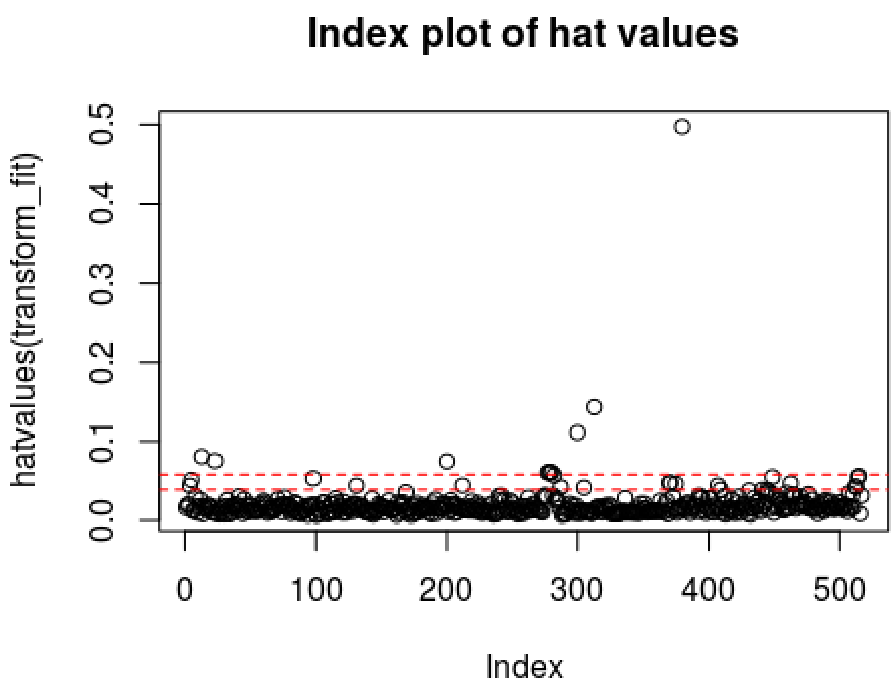
```
## 
## Call:
## lm(formula = Area ~ ., data = new_forest_fires)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -1.7772 -1.0634 -0.5548  0.8624  4.9483 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -2.577e-01  2.324e+00  -0.111   0.9118  
## X            3.680e-02  3.151e-02   1.168   0.2435  
## Y            4.865e-03  5.945e-02   0.082   0.9348  
## Month        2.186e-02  1.759e-02   1.243   0.2145  
## Day          2.505e-02  3.245e-02   0.772   0.4405  
## FFMC         1.192e-03  3.273e-02   0.036   0.9710  
## DMC         -6.953e-02  4.089e-02  -1.700   0.0897 .
## DC           2.983e-03  6.594e-03   0.452   0.6512  
## ISI          3.225e+00  2.671e+00   1.208   0.2278  
## Temp         4.986e-03  1.819e-02   0.274   0.7841  
## RH           4.069e-03  2.006e-02   0.203   0.8393  
## Wind         7.834e-02  3.737e-02   2.097   0.0365 *
## Rain         5.762e-02  2.120e-01   0.272   0.7859  
## FFMC.DMC     7.840e-04  4.492e-04   1.745   0.0816 .
## FFMC.DC     -2.075e-05  8.432e-05  -0.246   0.8058  
## FFMC.ISI    -3.485e-02  2.678e-02  -1.301   0.1937  
## DC.ISI      -4.306e-04  8.371e-04  -0.514   0.6072  
## RH_sq       -8.218e-05  1.897e-04  -0.433   0.6650  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 1.375 on 498 degrees of freedom
## Multiple R-squared:  0.03348,    Adjusted R-squared:  0.0004877 
## F-statistic: 1.015 on 17 and 498 DF,  p-value: 0.44
```

After removing the outlier the accuracy score increased.

Let us try removing high influential observations(212,416,514)

2) High Leverage points

```
hat.plot<-function(transform_fit){

  p<-length(coefficients(transform_fit))
  n<-length(fitted(transform_fit))
  plot(hatvalues(transform_fit),main='Index plot of hat values')
  abline(h=c(2,3)*p/n,col='red',lty=2)
  identify(1:n, hatvalues(transform_fit),names(hatvalues(transform_fit)))
}

hat.plot(transform_fit)
```

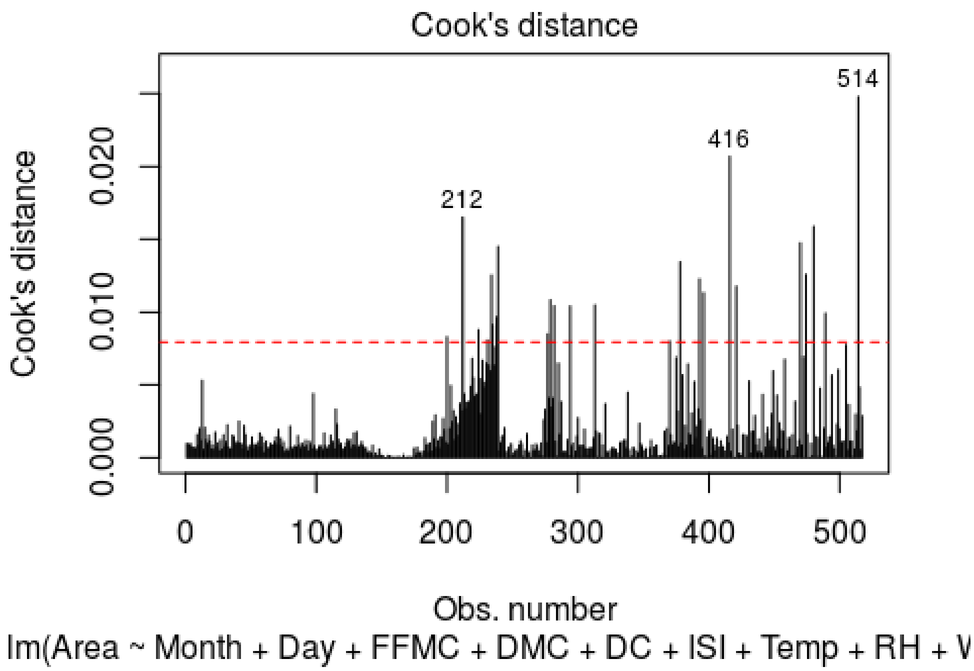



```
## integer(0)
```

There are some outliers according to hat values.

3) Influential Observations

```
cutoff<-4/(nrow(forest_fires)-length(transform_fit$coefficients)-2)
plot(transform_fit,which=4,cook.levels=cutoff)
abline(h=cutoff,lty=2,col="red")
```



The graph identifies that 380,416 and 514 as influential observations.

Corrective measures.

```
influential_observation<-as.data.frame(new_forest_fires[-c(380,416,514),])
head(new_forest_fires)
```

```
## # A tibble: 6 x 18
##       X     Y Month   Day  FFMC   DMC    DC   ISI  Temp    RH  Wind  Rain
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     7     5     8     1  86.2  26.2  94.3  1.81   8.2    51   6.7   0
## 2     7     4    11     6  90.6  35.4 669.   2.04  18      33   0.9   0
## 3     7     4    11     3  90.6  43.7 687.   2.04  14.6    33   1.3   0
## 4     8     6     8     1  91.7  33.3  77.5  2.30   8.3    97   4     0.2
## 5     8     6     8     4  89.3  51.3 102.   2.36  11.4    99   1.8   0
## 6     8     6     2     4  92.3  85.3 488    2.75  22.2    29   5.4   0
## # … with 6 more variables: Area <dbl>, FFMC.DMC <dbl>, FFMC.DC <dbl>,
## #   FFMC.ISI <dbl>, DC.ISI <dbl>, RH_sq <dbl>
```

```
influential_fit<-lm(Area~.,data=influential_observation)
summary(influential_fit)
```

```
##
## Call:
## lm(formula = Area ~ ., data = influential_observation)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -1.7603 -1.0614 -0.5893  0.8788  4.9562
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.499e-01  2.331e+00  -0.064   0.9488
## X            3.384e-02  3.169e-02   1.068   0.2860
## Y            1.035e-02  5.982e-02   0.173   0.8627
## Month        2.241e-02  1.768e-02   1.268   0.2055
## Day          2.237e-02  3.265e-02   0.685   0.4935
## FFMC         7.896e-04  3.279e-02   0.024   0.9808
## DMC         -6.806e-02  4.101e-02  -1.659   0.0977 .
## DC           2.792e-03  6.610e-03   0.422   0.6729
## ISI          3.260e+00  2.679e+00   1.217   0.2242
## Temp         2.761e-03  1.837e-02   0.150   0.8806
## RH           2.946e-03  2.012e-02   0.146   0.8836
## Wind         7.374e-02  3.772e-02   1.955   0.0511 .
## Rain         6.709e-02  2.125e-01   0.316   0.7523
## FFMC.DMC     7.690e-04  4.505e-04   1.707   0.0884 .
## FFMC.DC     -1.999e-05  8.447e-05  -0.237   0.8130
## FFMC.ISI    -3.520e-02  2.687e-02  -1.310   0.1907
## DC.ISI      -3.755e-04  8.404e-04  -0.447   0.6552
## RH_sq       -7.710e-05  1.902e-04  -0.405   0.6854
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.377 on 495 degrees of freedom
## Multiple R-squared:  0.0321, Adjusted R-squared:  -0.001142
## F-statistic: 0.9657 on 17 and 495 DF,  p-value: 0.4962
```

This has improved our accuracy by removing influential observations and outliers.

5. Select the best regression model

Anova Approach

```
anova(fit,transform_fit,poly_fit,interact_fit,interaction_fit,mod)
```

```
## Analysis of Variance Table
##
## Model 1: Area ~ X + Y + Month + Day + FFMC + DMC + DC + ISI + Temp + RH +
##     Wind + Rain
## Model 2: Area ~ Month + Day + FFMC + DMC + DC + ISI + Temp + RH + Wind
## Model 3: Area ~ X + Y + Month + Day + FFMC + DMC^2 + DC^2 + ISI^3 + Temp +
##     RH + Wind
## Model 4: Area ~ X + Y + Month + Day + FFMC + DMC + DC + ISI + Temp + RH +
##     Wind + Rain + FFMC.DMC + FFMC.DC + FFMC.ISI + DMC.DC + DMC.ISI +
##     DC.ISI + Wind.Temp + Temp.RH + Wind.RH
## Model 5: Area ~ Month + Day + (FFMC + DMC + DC + ISI + Temp + RH + Wind)^2
## Model 6: Area ~ X + Y + Month + DMC + DC + FFMC.DMC + FFMC.DC + FFMC.ISI +
##     DC.ISI + RH + RH_sq
##   Res.Df     RSS  Df Sum of Sq      F Pr(>F)
```

```
## 1    504 2039170
## 2    507     986  -3   2038184
## 3    505     981   2         6 1.4031 0.2468
## 4    495     964  10        16 0.8353 0.5947
## 5    486     957   9         8 0.4333 0.9171
## 6    505     988 -19       -31 0.8385 0.6607
```

AIC approach for selecting the best model:

```
AIC(fit,transform_fit,poly_fit,interaction_fit,mod)

##                 df      AIC
## fit             14 5775.948
## transform_fit   11 1823.060
## poly_fit        13 1824.157
## interaction_fit 32 1849.284
## mod             13 1827.960
```

The p-value(0.2515,0.6192,0.8753.0.6346) indicates that the model does not add to linear prediction and we can remove it, with AIC score of 1823.119 we can choose transform fit as our best model.

After the selection of best model let's use it for linear regression

```
linear<-read_excel('Forest Fires Data.xlsx')
linear$Month <- as.numeric(as.factor(linear$Month))
linear$Day <- as.numeric(as.factor(linear$Day))
linear<-linear[-c(239,380,416,514),]
linear$Area<-transform(linear$Area)
linear$ISI<-transform(linear$ISI)
head(linear)

## # A tibble: 6 x 13
##       X     Y Month   Day  FFMC   DMC    DC   ISI  Temp    RH  Wind  Rain
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1     7     5     8     1  86.2  26.2  94.3  1.81   8.2    51   6.7   0
## 2     7     4    11     6  90.6  35.4 669.   2.04  18      33   0.9   0
## 3     7     4    11     3  90.6  43.7 687.   2.04  14.6    33   1.3   0
## 4     8     6     8     1  91.7  33.3  77.5  2.30   8.3    97   4     0.2
## 5     8     6     8     4  89.3  51.3 102.   2.36  11.4    99   1.8   0
## 6     8     6     2     4  92.3  85.3 488    2.75  22.2    29   5.4   0
## # … with 1 more variable: Area <dbl>
```

Partitioning the data into 60 and 40

```
train<-round(nrow(linear)*0.60)
test<-round(nrow(linear)*0.40)
train

## [1] 308

test
```

```
## [1] 205
```

```r
set.seed(1)
train.index<-sample(c(1:513),308)
train.df<-linear[train.index,]
test.df<-linear[-train.index,]
test.df<-test.df[which(test.df$Area>0),]
summary(train)
```

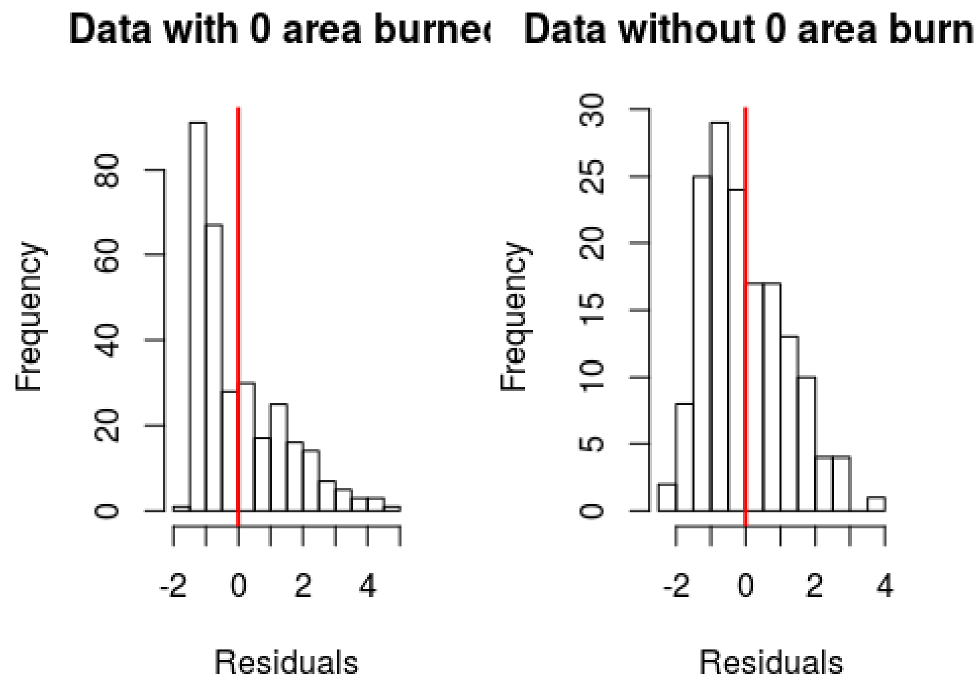
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##     308     308     308     308     308     308
```

```r
summary(test)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##     205     205     205     205     205     205
```

fitting linear regression to training set

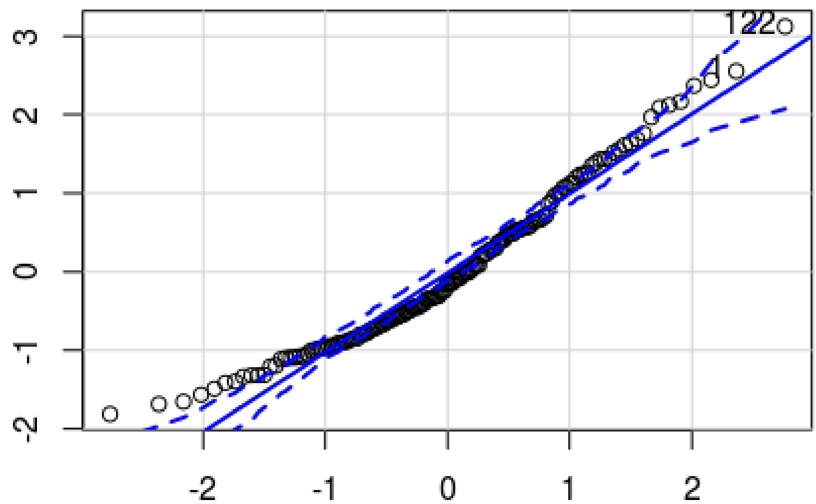
```r
train.lm_1<-lm(Area~ Month + Day + FFMC + DMC + DC + ISI + Temp + RH + 
Wind,data=train.df)
train.lm<-lm(Area~Month + Day + FFMC + DMC + DC + ISI + Temp + RH + 
Wind,data=train.df[which(train.df$Area>0),])

par(mfrow=c(1,2))
hist(train.lm_1$residuals, main = "Data with 0 area burned", xlab = 
'Residuals')
abline(v=mean(train.lm_1$residuals), col='red', lwd=2)
hist(train.lm$residuals,main = "Data without 0 area burned", xlab = 
'Residuals')
abline(v=mean(train.lm$residuals), col='red', lwd=2)
```



By Removing the Skewed data at 0, the normality condition is almost satisfied.This removes lot of values but it is needed for better prediction. Let us plot the QQ Plot and observe whether removing 0 helps in normality or not.

```
qqPlot(train.lm, main="Plot of Residuals after Zero's are
removed",xlab='',ylab='')
```

Plot of Residuals after Zero's are removed




```
## [1]   1 122
```

```r
summary(train.lm)
```

```
##
## Call:
## lm(formula = Area ~ Month + Day + FFMC + DMC + DC + ISI + Temp +
##     RH + Wind, data = train.df[which(train.df$Area > 0), ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.1609 -0.9142 -0.1844  0.6972  3.6050
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.165287   5.435181  -0.214   0.8305
## Month        0.056647   0.027377   2.069   0.0403 *
## Day          0.023808   0.052390   0.454   0.6502
## FFMC         0.057363   0.067798   0.846   0.3989
## DMC          0.005405   0.002663   2.030   0.0442 *
## DC          -0.001238   0.000713  -1.736   0.0846 .
## ISI         -0.658000   0.455627  -1.444   0.1509
## Temp        -0.024204   0.025374  -0.954   0.3417
## RH          -0.007647   0.008489  -0.901   0.3692
## Wind        -0.007786   0.058915  -0.132   0.8950
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 1.22 on 144 degrees of freedom
## Multiple R-squared:  0.08433,    Adjusted R-squared:  0.0271
## F-statistic: 1.474 on 9 and 144 DF,  p-value: 0.1631
```

This model gives an accuracy of 0.08103 after removing outliers, influential observations data and removing skewed data of zeros.

fitting the data to testing set

```
pred<-predict(train.lm,test.df)
summary(pred)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.463   1.838   2.128   2.145   2.436   3.035
```

Comparing the errors between actual and predicted values

```
comparing_residuals<-test.df$Area[1:20]-pred[1:20]
comparing<-
data.frame('predicted'=pred[1:20],'Actual'=test.df$Area[1:20],'Residuals'=com
paring_residuals)
comparing

##    predicted    Actual  Residuals
## 1   2.288611 0.3074847 -1.9811261
## 2   1.915360 0.4382549 -1.4771049
## 3   1.974898 0.5364934 -1.4384050
## 4   2.059317 0.6418539 -1.4174635
## 5   2.644877 0.6678294 -1.9770479
## 6   1.863506 0.7514161 -1.1120903
## 7   1.805978 0.9001613 -0.9058166
## 8   2.439326 0.9001613 -1.5391645
## 9   2.467423 0.9593502 -1.5080730
## 10  1.818015 0.9669838 -0.8510309
## 11  1.840545 0.9707789 -0.8697664
## 12  2.676714 1.0116009 -1.6651130
## 13  2.350196 1.0818052 -1.2683907
## 14  1.867310 1.1908876 -0.6764223
## 15  2.345194 1.2612979 -1.0838966
## 16  2.524580 1.2725656 -1.2520147
## 17  1.672969 1.5040774 -0.1688917
## 18  2.206928 1.7245507 -0.4823769
## 19  2.283588 1.7715568 -0.5120313
## 20  2.080148 1.8625285 -0.2176199
```

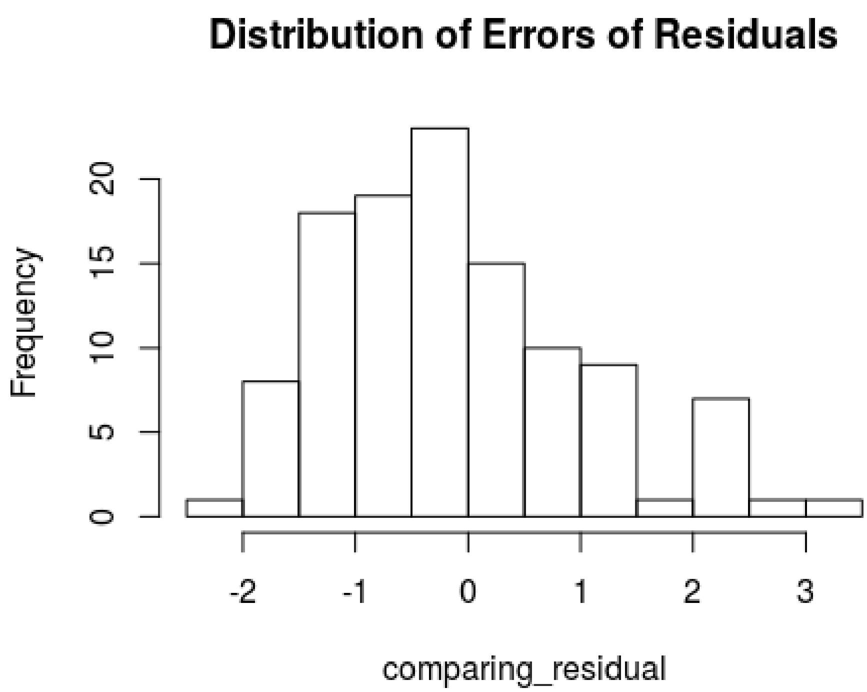
Let us check the accuracy

```
accuracy(pred,test.df$Area)

##                   ME     RMSE       MAE      MPE     MAPE
## Test set -0.1323322 1.157578 0.9343717 -60.2015 82.99458
```

```
rsq <- function (x, y) {
  return (cor(x, y) ^ 2)
}

rsq(pred,test.df$Area)

## [1] 0.006943684
```

The R-squared value indicates that the model performs really bad in the validation data.

```
comparing_residual<-test.df$Area-pred
hist(comparing_residual,main='Distribution of Errors of Residuals')
```



The distribution of Rediuals error indicates how the predicted value differ from actual values.

6. Fine tuning the predictor variables.

Exhausttive Search

```
exhaustive_search <- regsubsets(Area ~ ., data = train.df, nbest = 1, nvmax =
dim(train.df)[2],
method = "exhaustive")

sum<-summary(exhaustive_search)
sum$which
```

```
##    (Intercept)    X     Y Month   Day  FFMC   DMC    DC   ISI  Temp    RH
## 1         TRUE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## 2         TRUE TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
## 3         TRUE TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
## 4         TRUE TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE
## 5         TRUE TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE
## 6         TRUE TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE
## 7         TRUE TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE
## 8         TRUE TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE
## 9         TRUE TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE
## 10        TRUE TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE
## 11        TRUE TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
## 12        TRUE TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
##     Wind  Rain
## 1  FALSE FALSE
## 2  FALSE FALSE
## 3  FALSE FALSE
## 4  FALSE FALSE
## 5   TRUE FALSE
## 6   TRUE  TRUE
## 7   TRUE  TRUE
## 8   TRUE  TRUE
## 9   TRUE  TRUE
## 10  TRUE  TRUE
## 11  TRUE  TRUE
## 12  TRUE  TRUE
```

Metrics

```
sum$rsq

##  [1] 0.01252218 0.02132376 0.02507425 0.03048489 0.03456992 0.03788318
##  [7] 0.04000324 0.04146125 0.04196392 0.04218057 0.04235040 0.04236281

sum$adjr2

##  [1] 0.009295127 0.014906210 0.015453269 0.017686007 0.018585975
##  [6] 0.018704773 0.017603320 0.015814731 0.013029943 0.009930758
## [11] 0.006762066 0.003408077

sum$cp

##  [1]  0.1924021 -0.5189234  0.3257395  0.6589926  1.4005996  2.3799489
##  [7]  3.7268638  5.2777247  7.1228783  9.0561389 11.0038241 13.0000000
```

The Exhaustive search indicates that the model fits the data really badly, by trying different combination of predictor variables and adjusted R square values indicates that the fit is not good for each combination of fits.

Relative importance of variables.

Backward Elimination

```r
step <- step(train.lm, direction = "backward")
```

```
## Start:  AIC=70.95
## Area ~ Month + Day + FFMC + DMC + DC + ISI + Temp + RH + Wind
##
##         Df Sum of Sq    RSS    AIC
## - Wind   1    0.0260 214.42 68.973
## - Day    1    0.3075 214.70 69.175
## - FFMC   1    1.0658 215.46 69.718
## - RH     1    1.2083 215.60 69.820
## - Temp   1    1.3548 215.75 69.924
## <none>               214.40 70.954
## - ISI    1    3.1052 217.50 71.169
## - DC     1    4.4889 218.88 72.145
## - DMC    1    6.1343 220.53 73.299
## - Month  1    6.3744 220.77 73.466
##
## Step:  AIC=68.97
## Area ~ Month + Day + FFMC + DMC + DC + ISI + Temp + RH
##
##         Df Sum of Sq    RSS    AIC
## - Day    1    0.3330 214.75 67.212
## - RH     1    1.1855 215.61 67.822
## - FFMC   1    1.2024 215.62 67.834
## - Temp   1    1.3640 215.79 67.949
## <none>               214.42 68.973
## - ISI    1    3.4870 217.91 69.457
## - DC     1    4.4895 218.91 70.164
## - DMC    1    6.1155 220.54 71.304
## - Month  1    6.6844 221.11 71.700
##
## Step:  AIC=67.21
## Area ~ Month + FFMC + DMC + DC + ISI + Temp + RH
##
##         Df Sum of Sq    RSS    AIC
## - FFMC   1    1.1207 215.88 66.014
## - Temp   1    1.1981 215.95 66.069
## - RH     1    1.2683 216.02 66.119
## <none>               214.75 67.212
## - ISI    1    3.3963 218.15 67.628
## - DC     1    4.5610 219.32 68.448
## - DMC    1    6.1007 220.86 69.526
## - Month  1    6.4022 221.16 69.736
##
## Step:  AIC=66.01
## Area ~ Month + DMC + DC + ISI + Temp + RH
##
##         Df Sum of Sq    RSS    AIC
## - Temp   1    0.8643 216.74 64.629
## - RH     1    2.0646 217.94 65.479
```

```
## - ISI    1    2.7828 218.66 65.986
## <none>              215.88 66.014
## - DC     1    4.4502 220.33 67.156
## - Month  1    7.7145 223.59 69.421
## - DMC    1    7.8952 223.77 69.545
##
## Step:  AIC=64.63
## Area ~ Month + DMC + DC + ISI + RH
##
##         Df Sum of Sq    RSS    AIC
## - RH     1    1.2124 217.95 63.488
## <none>              216.74 64.629
## - ISI    1    4.1306 220.87 65.536
## - DC     1    5.6415 222.38 66.586
## - DMC    1    7.1107 223.85 67.600
## - Month  1    8.2691 225.01 68.395
##
## Step:  AIC=63.49
## Area ~ Month + DMC + DC + ISI
##
##         Df Sum of Sq    RSS    AIC
## <none>              217.95 63.488
## - ISI    1    3.5573 221.51 63.981
## - DC     1    5.4117 223.36 65.265
## - DMC    1    6.4937 224.45 66.009
## - Month  1    8.9682 226.92 67.698
```

```
summary(step)
```

```
##
## Call:
## lm(formula = Area ~ Month + DMC + DC + ISI, data =
train.df[which(train.df$Area >
##     0), ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0917 -0.9488 -0.1458  0.7532  3.3250
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.7208952  0.5503729   4.944 2.04e-06 ***
## Month        0.0639425  0.0258240   2.476   0.0144 *
## DMC          0.0051796  0.0024583   2.107   0.0368 *
## DC          -0.0013231  0.0006879  -1.923   0.0563 .
## ISI         -0.3814717  0.2446203  -1.559   0.1210
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.209 on 149 degrees of freedom
```

```
## Multiple R-squared:  0.06914,    Adjusted R-squared:  0.04415
## F-statistic: 2.767 on 4 and 149 DF,  p-value: 0.02957
```

```
pred_step <- predict(step,test.df)
accuracy(pred_step, test.df$Area)
```

```
##                  ME     RMSE       MAE       MPE     MAPE
## Test set -0.1262053 1.147179 0.9226794 -58.99195 81.33781
```

7.Interpret the prediction results

According to Backward elimination the best model has predictor variables as Month,DMC,DC,ISI With R squared error of 0.06914
F statistic and p-value indicates that the predictor combination is significant. This model is much better compared to the previous fit.

Fine tuning and selecting the best predictor variables.

With fine tuning the best Response variable has following intercept and slope for prediction.

Area=2.7208952+ Month(0.0639425)+DMC (0.0051796)+DC (-0.0013231)+ISI(-0.3814717)