Linear Regression Analysis

Suresh Ooty

Table of Contents

Regression Model on Full datasets

Choices of Dataset

Min-Max Transformed dataset

Note : We noted in the *Dataset section* that Min-Max transformation did not remove the correlation to a greater extent as LOG transformation did. Hence, both the transformations are applied on the dataset for analysis.

```
mins <- apply(data_15, 2, min)
maxs <- apply(data_15, 2, max)
scaled_data <- as.data.frame(scale(data_15, center = mins,scale = maxs -
mins))
train_ <- scaled_data[1:4000,]
test_  <- scaled_data[4001:29414,]
head(train_)

##   LeadToFailure Production Hay_out_waste   CmpACon    EleCon   NatGCon
## 1             0  0.6312849    0.19047619 0.1661696 0.5902301 0.5702479
## 2             0  0.6312849    0.00000000 0.1673625 0.6106525 0.5785124
## 3             0  0.6256983    0.04761905 0.1727305 0.6243814 0.5785124
## 4             0  0.6312849    0.05555556 0.1798879 0.5717462 0.5950413
```

```
## 5             0  0.6256983    0.00000000 0.1775021 0.5986682 0.5950413
## 6             0  0.6312849    0.09523810 0.1804843 0.4342744 0.5950413
##      SteCon   WatGCon    WatMCon   WatWGen
## 1 0.6614602 0.2307692 0.02380952 0.3928571
## 2 0.6549219 0.2435897 0.00000000 0.3928571
## 3 0.6509263 0.2435897 0.02380952 0.4285714
## 4 0.6592808 0.2307692 0.02380952 0.3928571
## 5 0.6694515 0.2435897 0.00000000 0.4285714
## 6 0.6342172 0.2307692 0.02380952 0.3928571
```

```
head(test_)
```

```
##      LeadToFailure Production Hay_out_waste   CmpACon    EleCon   NatGCon
## 4001             0  0.6536313     0.0000000 0.2389359 0.6007549 0.6694215
## 4002             0  0.6536313     0.0000000 0.2335679 0.6617483 0.6694215
## 4003             0  0.6424581     0.0000000 0.2371466 0.5914958 0.6694215
## 4004             0  0.6480447     0.0000000 0.2425146 0.6049055 0.6694215
## 4005             0  0.6480447     0.0000000 0.2317786 0.6423750 0.6694215
## 4006             0  0.6480447     0.2063492 0.2317786 0.5985541 0.6776860
##         SteCon   WatGCon    WatMCon   WatWGen
## 4001 0.6236833 0.2307692 0.09523810 0.6428571
## 4002 0.6244097 0.2435897 0.02380952 0.6428571
## 4003 0.6240465 0.2307692 0.04761905 0.6428571
## 4004 0.6189611 0.2435897 0.02380952 0.6428571
## 4005 0.6273157 0.2307692 0.02380952 0.6428571
## 4006 0.6276789 0.2435897 0.02380952 0.6428571
```

LOG transformed dataset

```
log.data_15 <- cbind.data.frame("LeadToFailure"=data_15$LeadToFailure,
log(data_15[,2:10]+1))
log.train_ <- log.data_15[1:4000,]
log.test_ <- log.data_15[4001:29414,]
head(log.train_)
```

```
##   LeadToFailure Production Hay_out_waste  CmpACon   EleCon  NatGCon
## 1         FALSE  0.7561220      5.484797 4.026973 7.260839 6.542126
## 2         FALSE  0.7561220      0.000000 4.031997 7.291809 6.556494
## 3         FALSE  0.7514161      4.110874 4.054299 7.312103 6.556494
## 4         FALSE  0.7561220      4.262680 4.083281 7.231956 6.584626
## 5         FALSE  0.7514161      0.000000 4.073713 7.273752 6.584626
## 6         FALSE  0.7561220      4.795791 4.085658 6.986127 6.584626
##     SteCon  WatGCon  WatMCon  WatWGen
## 1 7.248378 8.467183 3.311171 7.974840
## 2 7.238451 8.521239 0.000000 7.974840
## 3 7.232336 8.521239 3.311171 8.061823
## 4 7.245080 8.467183 3.311171 7.974840
## 5 7.260378 8.521239 0.000000 8.061823
## 6 7.206350 8.467183 3.311171 7.974840
```

```r
head(log.test_)
```

```
##      LeadToFailure Production Hay_out_waste  CmpACon   EleCon  NatGCon
## 4001         FALSE  0.7747272       0.00000 4.294894 7.276919 6.702255
## 4002         FALSE  0.7747272       0.00000 4.277403 7.365338 6.702255
## 4003         FALSE  0.7654678       0.00000 4.289098 7.262786 6.702255
## 4004         FALSE  0.7701082       0.00000 4.306387 7.283191 6.702255
## 4005         FALSE  0.7701082       0.00000 4.271505 7.338092 6.702255
## 4006         FALSE  0.7701082       5.56452 4.271505 7.273578 6.714510
##        SteCon  WatGCon  WatMCon  WatWGen
## 4001 7.189614 8.467183 4.669729 8.467183
## 4002 7.190777 8.521239 3.311171 8.467183
## 4003 7.190196 8.467183 3.985913 8.467183
## 4004 7.182019 8.521239 3.311171 8.467183
## 4005 7.195417 8.467183 3.311171 8.467183
## 4006 7.195995 8.521239 3.311171 8.467183
```

Regression Model - MinMax

```r
n <- names(train_)
f <- as.formula(paste("Production ~", paste(n[!n %in% c("LeadToFailure",
"Production")], collapse = " + ")))
lmdl <- lm(f,data = train_)
summary(lmdl)
```

```
##
## Call:
## lm(formula = f, data = train_)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.36922 -0.03707 -0.00643  0.03160  0.34540
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.018015   0.009674  -1.862   0.0627 .
## Hay_out_waste -0.052432   0.011264  -4.655 3.35e-06 ***
## CmpACon        0.023915   0.039230   0.610   0.5422
## EleCon         0.236533   0.021797  10.852  < 2e-16 ***
## NatGCon        0.436355   0.015149  28.804  < 2e-16 ***
## SteCon         0.502598   0.018442  27.253  < 2e-16 ***
## WatGCon       -0.282336   0.020208 -13.971  < 2e-16 ***
## WatMCon        0.053326   0.026697   1.997   0.0458 *
## WatWGen       -0.011252   0.005163  -2.179   0.0294 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0619 on 3991 degrees of freedom
```

```
## Multiple R-squared:  0.8528, Adjusted R-squared:  0.8525
## F-statistic:  2891 on 8 and 3991 DF,  p-value: < 2.2e-16
```

CmpACon - Compressed Air Consumption is not significant

After reducing the independent variables

```
f <- as.formula(paste("Production ~", paste(n[!n %in% c("LeadToFailure",
"Production","CmpACon")], collapse = " + ")))
lmdl <- lm(f, data = train_)
summary(lmdl)

##
## Call:
## lm(formula = f, data = train_)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.36915 -0.03693 -0.00643  0.03159  0.34563
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -0.013963   0.007029  -1.987   0.0470 *
## Hay_out_waste -0.052326   0.011262  -4.646 3.49e-06 ***
## EleCon         0.237797   0.021697  10.960  < 2e-16 ***
## NatGCon        0.436848   0.015126  28.880  < 2e-16 ***
## SteCon         0.502481   0.018439  27.251  < 2e-16 ***
## WatGCon       -0.281131   0.020110 -13.980  < 2e-16 ***
## WatMCon        0.053237   0.026695   1.994   0.0462 *
## WatWGen       -0.011193   0.005162  -2.168   0.0302 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06189 on 3992 degrees of freedom
## Multiple R-squared:  0.8528, Adjusted R-squared:  0.8526
## F-statistic:  3305 on 7 and 3992 DF,  p-value: < 2.2e-16
```

Estimates : *Natural Gas consumption,Steam Consumption, Electricity consumption* are positive. *Hay out waste* is negative - indicates that it will have positive values, when production is zero during switch over to a different variety of paper,So a -ve value is justified.However this dataset has zero values for *"Production"* to study the response variable *"Production"* , all 'zero' values need to be dropped to study the influence.

Model for Positive Production values

'Zero' values from the dependent variable *Production* were dropped for further analysis.

```
train1_ <- train_[which(train_$Production != 0),]
n <- names(train1_)
f <- as.formula(paste("Production ~", paste(n[!n %in% c("LeadToFailure",
```

```r
"Production")], collapse = " + ")))
summary(lm(f,data = train1_))
```

```
##
## Call:
## lm(formula = f, data = train1_)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.42760 -0.03255 -0.00642  0.03128  0.33512
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.155103   0.015001  10.339  < 2e-16 ***
## Hay_out_waste -0.060931   0.011069  -5.505 3.95e-08 ***
## CmpACon        0.109381   0.040371   2.709  0.00677 **
## EleCon         0.202444   0.022126   9.149  < 2e-16 ***
## NatGCon        0.322092   0.017094  18.843  < 2e-16 ***
## SteCon         0.333155   0.021766  15.306  < 2e-16 ***
## WatGCon       -0.265233   0.021017 -12.620  < 2e-16 ***
## WatMCon        0.038001   0.026460   1.436  0.15104
## WatWGen       -0.007398   0.005102  -1.450  0.14717
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06071 on 3780 degrees of freedom
## Multiple R-squared:  0.3052, Adjusted R-squared:  0.3037
## F-statistic: 207.5 on 8 and 3780 DF,  p-value: < 2.2e-16
```

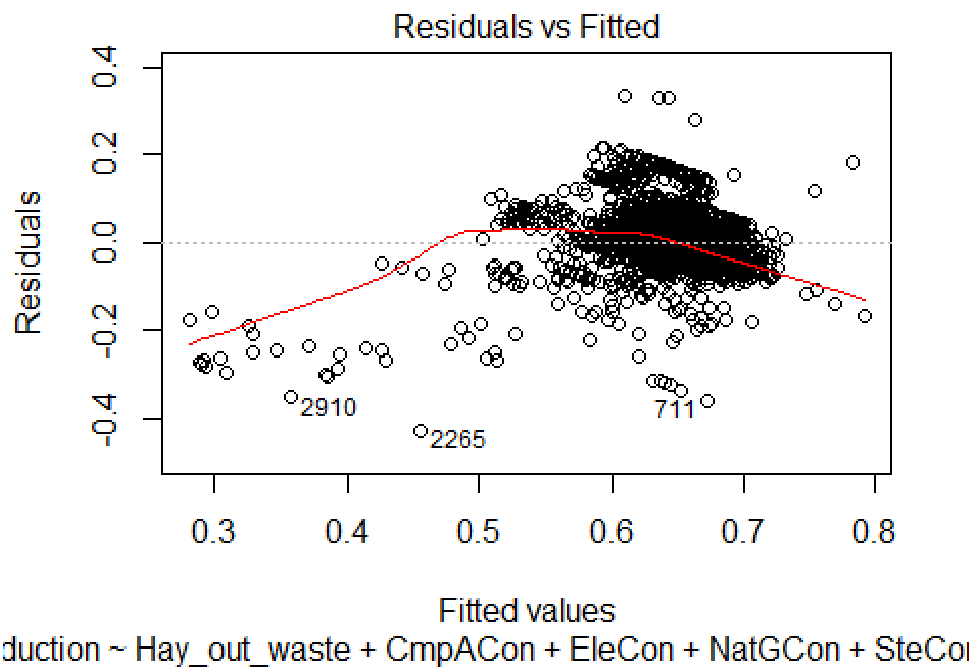
WatMCon & WatWGen are not significantly influencing Production values. Hence dropping them from the model.

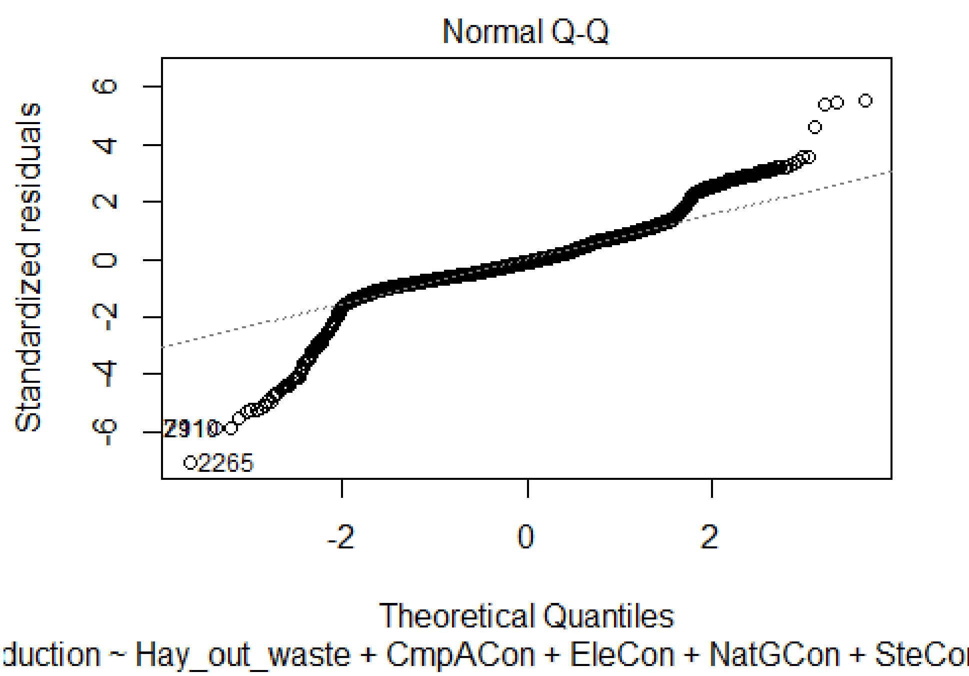
```r
lmdl <- lm(Production ~ Hay_out_waste + CmpACon + EleCon + NatGCon + SteCon +
               +     WatGCon, data = train1_)
summary(lmdl)
```

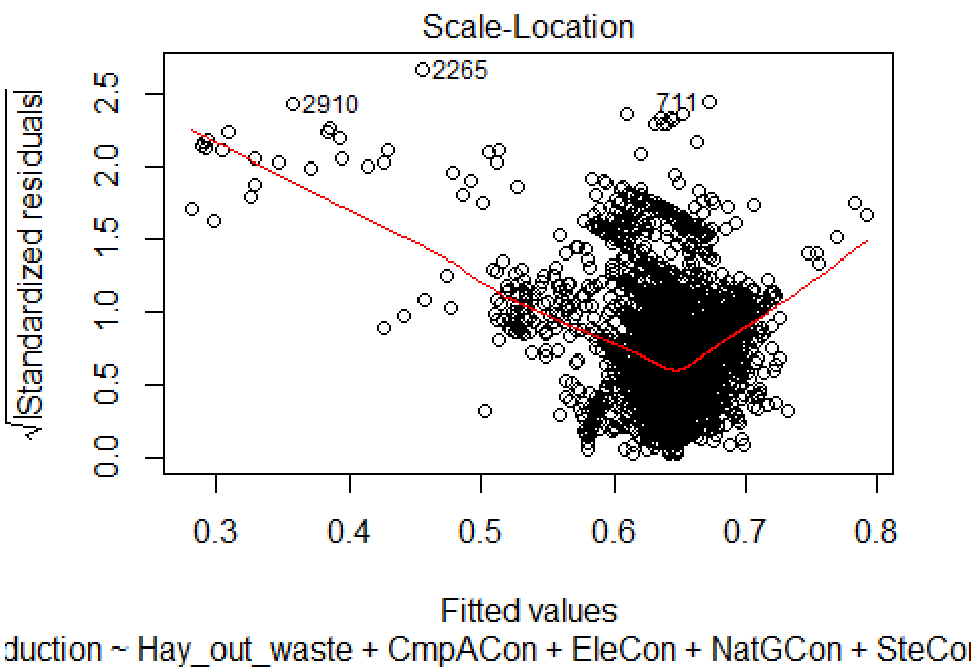
```
##
## Call:
## lm(formula = Production ~ Hay_out_waste + CmpACon + EleCon +
##     NatGCon + SteCon + +WatGCon, data = train1_)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.42805 -0.03254 -0.00650  0.03112  0.33464
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.15425    0.01487  10.370  < 2e-16 ***
## Hay_out_waste -0.06137    0.01107  -5.544 3.15e-08 ***
## CmpACon        0.10836    0.04037   2.684   0.0073 **
## EleCon         0.20049    0.02208   9.078  < 2e-16 ***
```

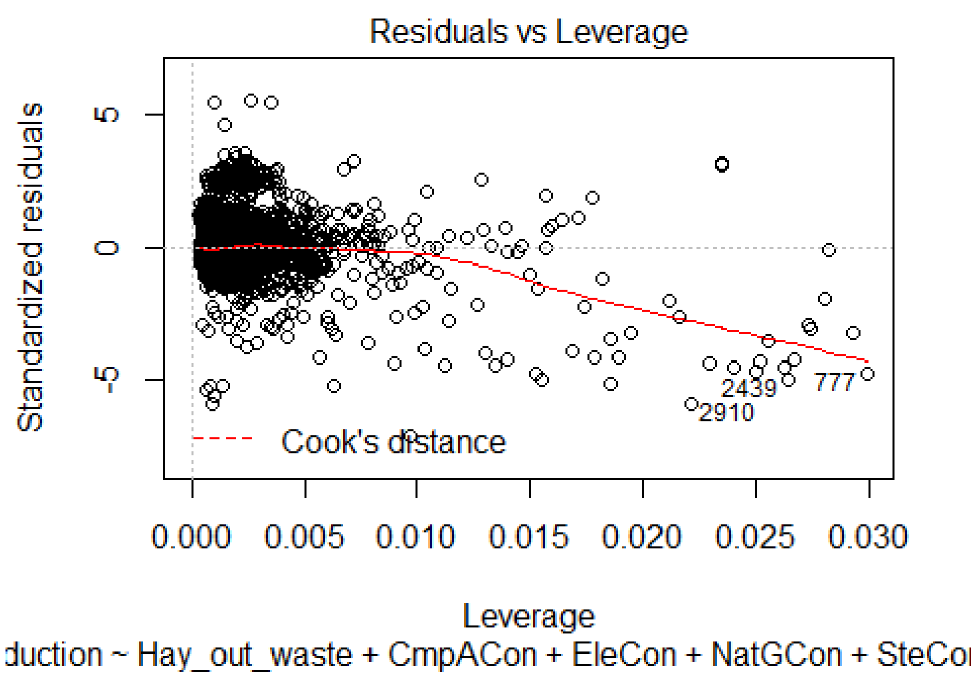
```
## NatGCon        0.32249    0.01708  18.879  < 2e-16 ***
## SteCon         0.33340    0.02175  15.330  < 2e-16 ***
## WatGCon       -0.26848    0.02089 -12.854  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06073 on 3782 degrees of freedom
## Multiple R-squared:  0.3044, Adjusted R-squared:  0.3033
## F-statistic: 275.9 on 6 and 3782 DF,  p-value: < 2.2e-16

plot(lmdl)
```


Residuals vs Fitted
Residuals
0.4
0.2
0.0
-0.2
-0.4
2910
2265
711
0.3
0.4
0.5
0.6
0.7
0.8
Fitted values
duction ~ Hay_out_waste + CmpACon + EleCon + NatGCon + SteCor


Normal Q-Q
Standardized residuals
6
4
2
0
-2
-4
-6
2910
2265
-2
0
2
Theoretical Quantiles
duction ~ Hay_out_waste + CmpACon + EleCon + NatGCon + SteCor





Notes on Model: All the predictors in the model came out as significant. *Hay_out_waste* has -ve intercept value, which is known (this value will be high at lower Production values and vice versa). The adjusted R-squared value is 0.3033, which could be due to high DF (3782).

A multicollinearity is not ruled out from this model, like it was noticed in *Min-Max Standardization* section under *Dataset* page.

Notes on Residual Plots: The plots indicate that the assumptions of *Independence*,*Homoscedasticity*,*Normality* are violated.

Regression Model - LOG

The LOG transformed dataset is applied on Linear Regression to study the response variable *Production*. The variable *LeadToFailure* is not considered as it is a manipulated data for Failure Prediction which is discussed in other sections.

```r
n <- names(log.train_)
f <- as.formula(paste("Production ~", paste(n[!n %in% c("LeadToFailure",
"Production")], collapse = " + ")))
log_mdl <- lm(f,data = log.train_)
summary(log_mdl)
```

```
##
## Call:
## lm(formula = f, data = log.train_)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.60839 -0.02904 -0.00694  0.02716  0.35104
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.3456465  0.0599713 -39.113  < 2e-16 ***
## Hay_out_waste -0.0044148  0.0005081  -8.688  < 2e-16 ***
## CmpACon       -0.0101235  0.0116170  -0.871   0.3836
## EleCon         0.3583780  0.0088530  40.481  < 2e-16 ***
## NatGCon        0.0585822  0.0030597  19.147  < 2e-16 ***
## SteCon         0.0308244  0.0017656  17.458  < 2e-16 ***
## WatGCon       -0.0059616  0.0009173  -6.499 9.09e-11 ***
## WatMCon        0.0007912  0.0006076   1.302   0.1929
## WatWGen       -0.0010633  0.0004588  -2.317   0.0205 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0638 on 3991 degrees of freedom
## Multiple R-squared:  0.8807, Adjusted R-squared:  0.8805
## F-statistic:  3684 on 8 and 3991 DF,  p-value: < 2.2e-16
```

__Notes:__ The variables *CmpACon*,*WatMCon*,*WatWGen* are not significant. A step by step backward elimination method was applied. The *WatWGen* turned out to be significant. For simplicity of documentation, not all the steps are covered. However, the dataset still contains the 'zero' values for production in this case.

```
f <- as.formula(paste("Production ~", paste(n[!n %in% c("LeadToFailure",
"Production","CmpACon","WatMCon")], collapse = " + ")))
log.mdl <- lm(f, data = log.train_)
summary(log.mdl)

##
## Call:
## lm(formula = f, data = log.train_)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.60709 -0.02911 -0.00642  0.02735  0.34920
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.3743308  0.0477110 -49.765  < 2e-16 ***
## Hay_out_waste -0.0044430  0.0005078  -8.750  < 2e-16 ***
## EleCon         0.3566821  0.0086791  41.097  < 2e-16 ***
## NatGCon        0.0584129  0.0030556  19.117  < 2e-16 ***
## SteCon         0.0310308  0.0017529  17.702  < 2e-16 ***
## WatGCon       -0.0060263  0.0009138  -6.595 4.82e-11 ***
## WatWGen       -0.0010325  0.0004580  -2.255   0.0242 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0638 on 3993 degrees of freedom
## Multiple R-squared:  0.8807, Adjusted R-squared:  0.8805
## F-statistic:  4911 on 6 and 3993 DF,  p-value: < 2.2e-16
```

After dropping zero values

The zero values were dropped from the initial dataset. Earlier a +1 approach was used to avoid log(0). So, a reconstruction of dataset is done.

```
tempdat <- data_15[which(data_15$Production !=0),]
log.data_15_1 <- cbind.data.frame("LeadToFailure"=tempdat$LeadToFailure,
log(tempdat[,2:10]+1))
log.train1_ <- log.data_15_1[1:4000,]
log.test1_ <- log.data_15_1[4001:27995,]
#Note that 27995 is the new size of dataset after dropping zeroes

n <- names(log.train1_)
f <- as.formula(paste("Production ~", paste(n[!n %in% c("LeadToFailure",
"Production")], collapse = " + ")))
summary(lm(f,data = log.train1_))

##
## Call:
## lm(formula = f, data = log.train1_)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -0.51126 -0.02712 -0.00550  0.02638  0.25386
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.4763895  0.0675006 -36.687  < 2e-16 ***
## Hay_out_waste -0.0042748  0.0003948 -10.827  < 2e-16 ***
## CmpACon       -0.0067892  0.0095016  -0.715    0.475
## EleCon         0.1126178  0.0108206  10.408  < 2e-16 ***
## NatGCon        0.2000236  0.0069320  28.855  < 2e-16 ***
## SteCon         0.1635839  0.0089130  18.353  < 2e-16 ***
## WatGCon       -0.0044748  0.0007903  -5.662  1.6e-08 ***
## WatMCon        0.0007874  0.0004792   1.643    0.100
## WatWGen       -0.0004199  0.0003633  -1.156    0.248
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05054 on 3991 degrees of freedom
## Multiple R-squared:  0.4548, Adjusted R-squared:  0.4537
## F-statistic: 416.1 on 8 and 3991 DF,  p-value: < 2.2e-16
```

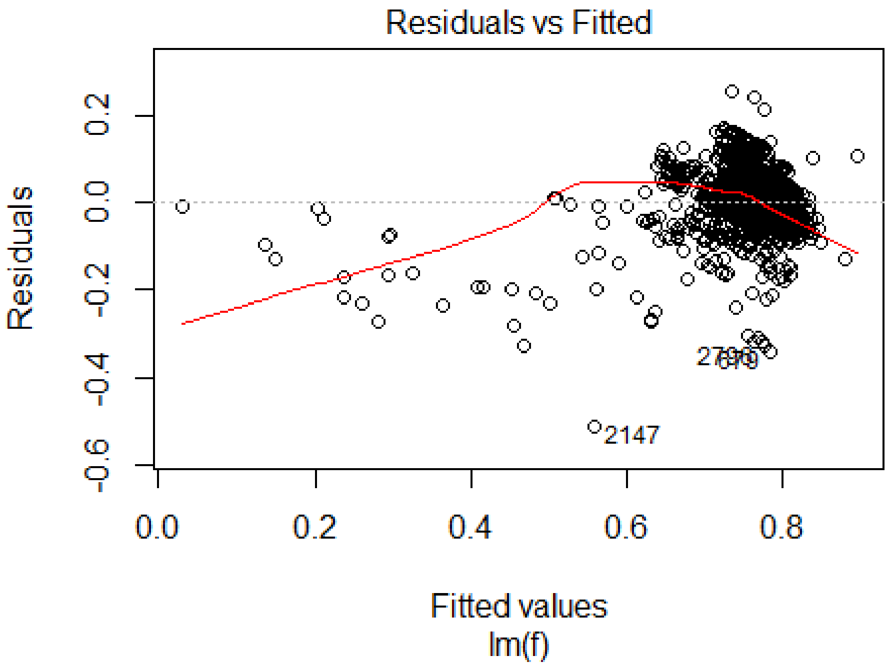
A backward elimination approach(manual) was adopted to reach this model.

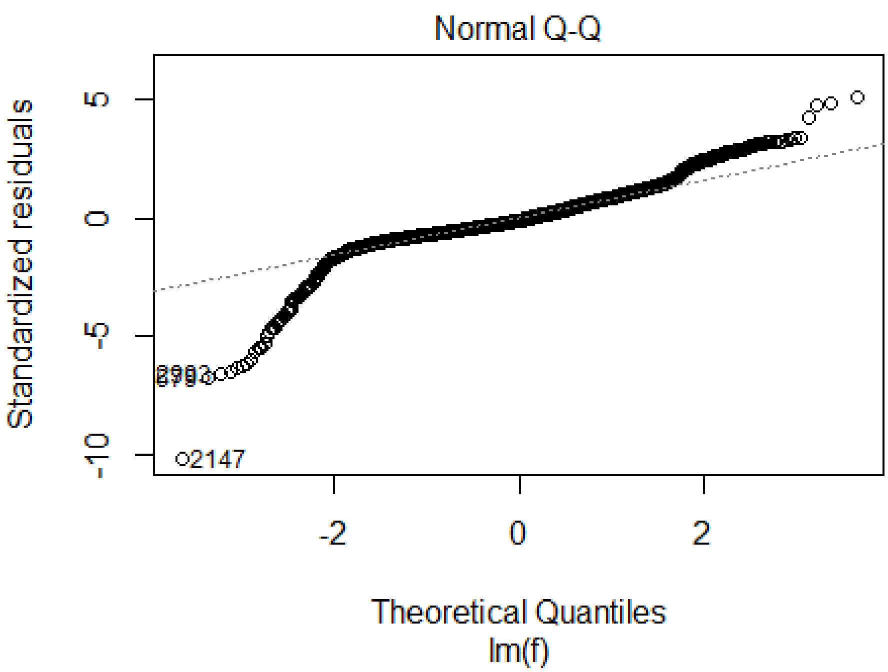
```
f <- as.formula(paste("Production ~", paste(n[!n %in% c("LeadToFailure",
"Production","CmpACon","WatWGen","WatMCon")], collapse = " + ")))
log.mdl <- lm(f,data = log.train1_)
summary(log.mdl)
```

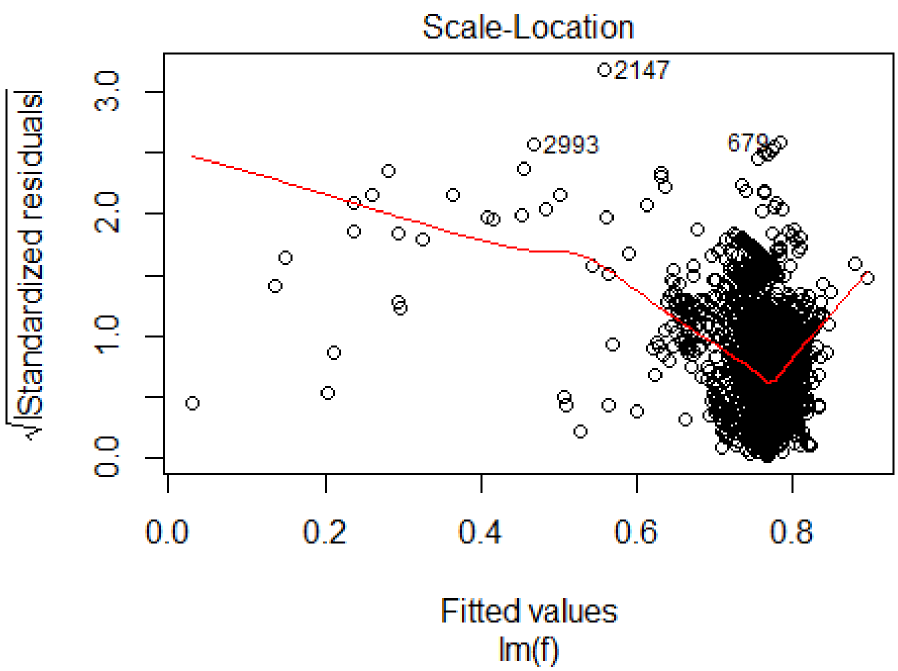
```
##
## Call:
## lm(formula = f, data = log.train1_)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.51107 -0.02716 -0.00544  0.02648  0.25456
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -2.4948071  0.0621037 -40.172  < 2e-16 ***
## Hay_out_waste -0.0043093  0.0003945 -10.923  < 2e-16 ***
## EleCon         0.1113259  0.0107735  10.333  < 2e-16 ***
## NatGCon        0.1998652  0.0069101  28.924  < 2e-16 ***
## SteCon         0.1635267  0.0088989  18.376  < 2e-16 ***
## WatGCon       -0.0045630  0.0007870  -5.798 7.24e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05055 on 3994 degrees of freedom
## Multiple R-squared:  0.4542, Adjusted R-squared:  0.4535
## F-statistic: 664.6 on 5 and 3994 DF,  p-value: < 2.2e-16
```

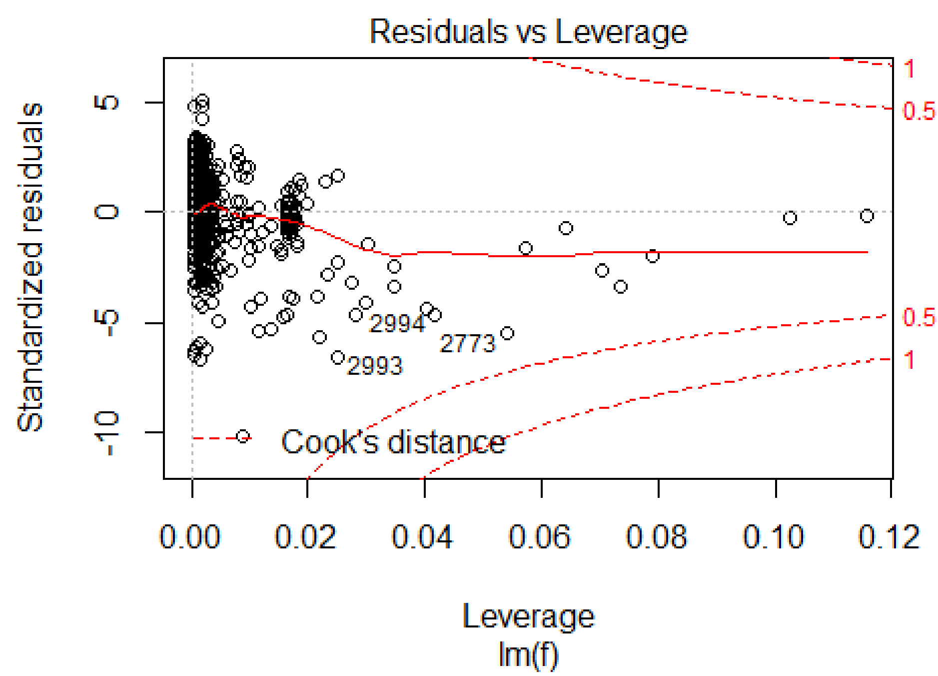
The model has all the variable significant. But the adjusted R-Squared values is 0.4535 which is low compared to the MinMax standardized values. However, the LOG transformation is likely to have removed the correlation. The *(Intercept) Estimate* is -ve, this may be an effect of transforming all the variables using LOG.

```
plot(log.mdl)
```


Residuals vs Fitted
Residuals
0.2
0.0
-0.2
-0.4
-0.6
0.0
0.2
0.4
0.6
0.8
2147
Fitted values
lm(f)


Normal Q-Q
Standardized residuals
5
0
-5
-10
-2
0
2
2147
Theoretical Quantiles
lm(f)





Notes on Model: The models have adjusted R-squared values below 0.5. But the DF values are high as the full dataset was used. There is a possibility that these models are highly generalized as it uses the full year data, without any slicing.

Notes on Residuals: The residuals indicate there are outliers and violating the assumptions of homoscedasticity, independence & normality to an extent. The residuals are skewed at the ends, which means there could be patterns behind these portions of data (Because, we are already dealing with LOG transformed data). And the dataset is real-time captured once in 15 minutes, this becomes a time-series data, that is prone to have seasonality and trends.

Conclusion on Full dataset Models

At this stage of research, the researcher feels that it is too early to fit a model without taking very closer look at the patterns of dataset. Once a clear subset of dataset is identified,then a model shall be applied to either predict the *Production* variable or to predict an upcoming *Failure* caused by one of the predictors.