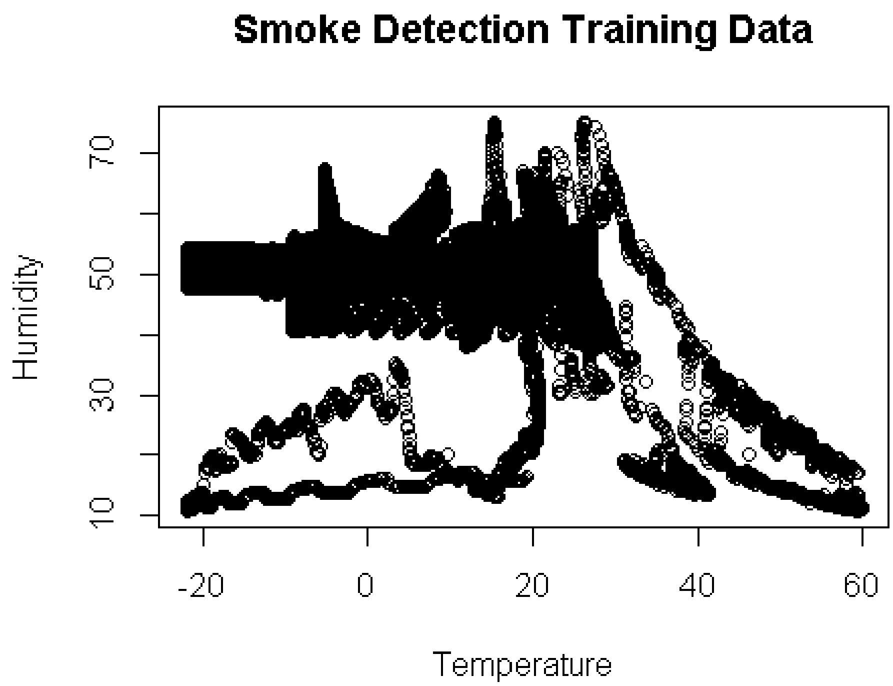
Regression

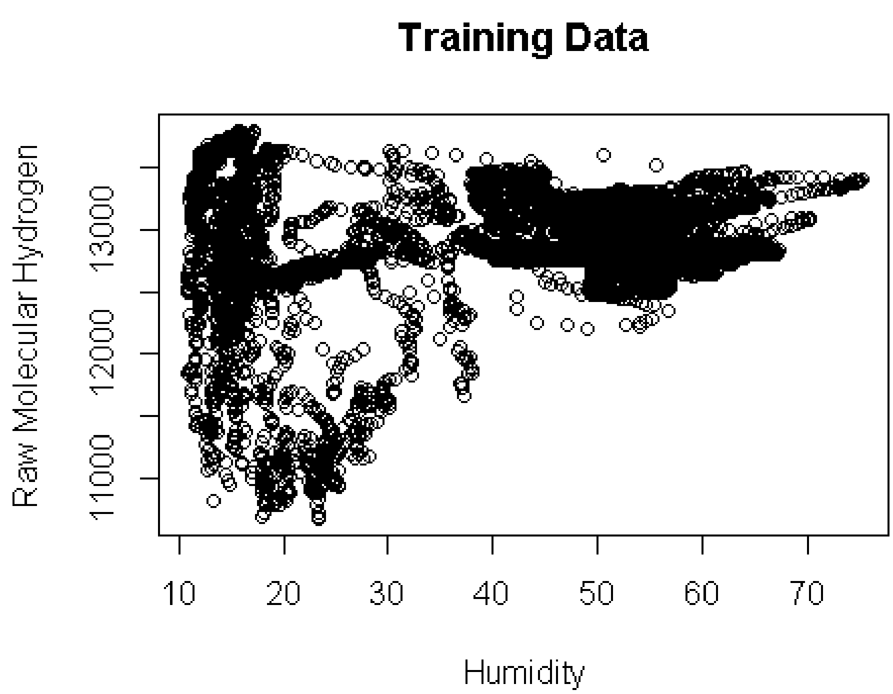
Matthew Naughton CS 4375.003 Portfolio: Linear Models

```
SmokeDetection <- read.csv(file = 'smoke_detection_iot.csv')

# a. Divide the Data into train and test data 80/20
i <- sample(1:nrow(SmokeDetection), nrow(SmokeDetection)*0.80, replace=FALSE)
train <- SmokeDetection[i,]
test <- SmokeDetection[-i,]

# b. Use 5 R functions for data exploration using the training data
head(train)

##           X        UTC Temperature.C. Humidity... TVOC.ppb. eCO2.ppm.
Raw.H2
## 52456 52455 1654713500         27.700       40.77        89       420
12774
## 60150 60149 1655127571         16.734       49.30       185       409
12783
## 47580 47579 1654783928         26.950       47.78      1295       400
12974
## 48049 48048 1654784397         26.720       48.57      1350       406
12975
## 52979 52978 1654714023         28.590       41.99       123       400
12786
## 3058   3057 1654736388         10.890       51.68       171       400
13162
##       Raw.Ethanol Pressure.hPa. PM1.0 PM2.5 NC0.5 NC1.0 NC2.5   CNT
Fire.Alarm
## 52456       20638       937.484  1.72  1.78 11.81 1.841 0.042  1313
0
## 60150       20540       937.388  1.81  1.88 12.44 1.940 0.044  3263
0
## 47580       19407       938.753  1.94  2.02 13.37 2.085 0.047 22585
1
## 48049       19397       938.725  1.65  1.72 11.37 1.773 0.040 23054
1
## 52979       20591       937.433  1.61  1.67 11.07 1.726 0.039  1836
0
## 3058        20005       939.671  0.84  0.88  5.80 0.904 0.020  3057
0

mean(train$Temperature.C., na.rm=TRUE)

## [1] 16.02103

mean(train$Humidity..., na.rm=TRUE)
```

```
## [1] 48.54535

range(train$Temperature.C.)

## [1] -22.01  59.93

range(train$Humidity...)

## [1] 10.74 75.20

names(train)

##  [1] "X"              "UTC"            "Temperature.C." "Humidity..."
##  [5] "TVOC.ppb."      "eCO2.ppm."      "Raw.H2"         "Raw.Ethanol"
##  [9] "Pressure.hPa."  "PM1.0"          "PM2.5"          "NC0.5"
## [13] "NC1.0"          "NC2.5"          "CNT"            "Fire.Alarm"

str(train)

## 'data.frame':    50104 obs. of  16 variables:
##  $ X             : int  52455 60149 47579 48048 52978 3057 7562 5690 40836
36039 ...
##  $ UTC           : int  1654713500 1655127571 1654783928 1654784397
1654714023 1654736388 1654740893 1654739021 1654777185 1654772388 ...
##  $ Temperature.C.: num  27.7 16.7 26.9 26.7 28.6 ...
##  $ Humidity...   : num  40.8 49.3 47.8 48.6 42 ...
##  $ TVOC.ppb.     : int  89 185 1295 1350 123 171 275 49 1097 1038 ...
##  $ eCO2.ppm.     : int  420 409 400 406 400 400 400 400 400 657 ...
##  $ Raw.H2        : int  12774 12783 12974 12975 12786 13162 13121 13245
12886 12790 ...
##  $ Raw.Ethanol   : int  20638 20540 19407 19397 20591 20005 19995 20201
19448 19468 ...
##  $ Pressure.hPa. : num  937 937 939 939 937 ...
##  $ PM1.0         : num  1.72 1.81 1.94 1.65 1.61 0.84 0.34 2.23 1.81 2.23
...
##  $ PM2.5         : num  1.78 1.88 2.02 1.72 1.67 0.88 0.35 2.31 1.88 2.32
...
##  $ NC0.5         : num  11.8 12.4 13.4 11.4 11.1 ...
##  $ NC1.0         : num  1.84 1.94 2.08 1.77 1.73 ...
##  $ NC2.5         : num  0.042 0.044 0.047 0.04 0.039 0.02 0.008 0.054
0.044 0.054 ...
##  $ CNT           : int  1313 3263 22585 23054 1836 3057 7562 5690 15842
11045 ...
##  $ Fire.Alarm    : int  0 0 1 1 0 0 1 1 1 1 ...

# c. Create 2 informative graphs using the training data
plot(train$Temperature.C., train$Humidity..., xlab="Temperature",
ylab="Humidity", main="Smoke Detection Training Data")
```



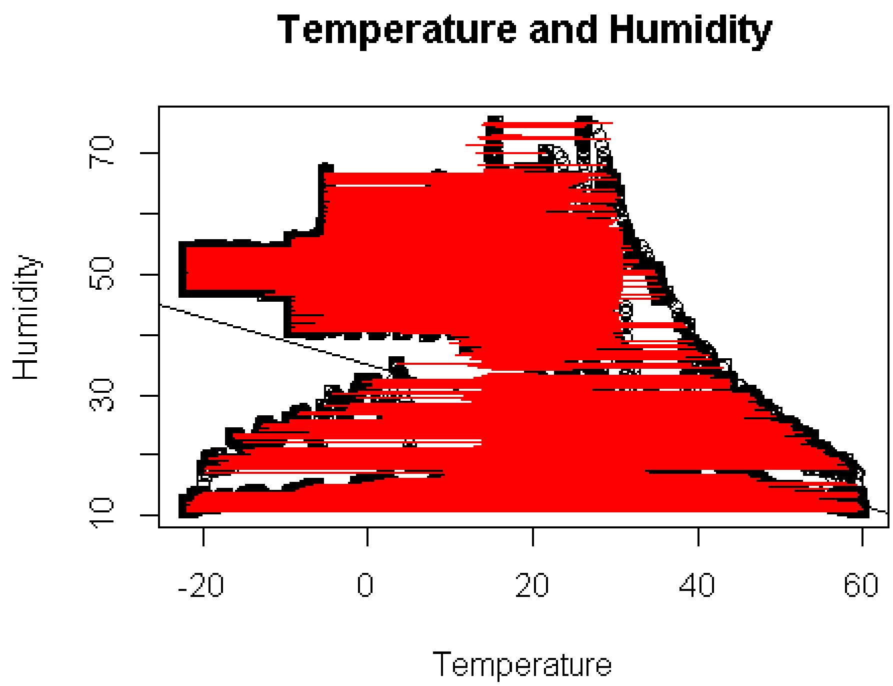

```r
plot(train$Humidity..., train$Raw.H2, xlab="Humidity", ylab="Raw Molecular Hydrogen", main="Training Data")
```



```r
# d. Build a simple linear regression model (one predictor). Output the
summary
lm1 <- lm(train$Temperature.C.~train$Humidity..., data=train)
lm1

##
## Call:
## lm(formula = train$Temperature.C. ~ train$Humidity..., data = train)
##
## Coefficients:
##       (Intercept)  train$Humidity...
##           35.1524            -0.3941

summary(lm1) # summary

##
## Call:
## lm(formula = train$Temperature.C. ~ train$Humidity..., data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -52.926  -5.635   5.373  10.226  30.967
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       35.152434   0.346474  101.46   <2e-16 ***
## train$Humidity... -0.394093   0.007022  -56.13   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.9 on 50102 degrees of freedom
## Multiple R-squared:  0.05916,    Adjusted R-squared:  0.05914
## F-statistic:  3150 on 1 and 50102 DF,  p-value: < 2.2e-16
```

As you can see by the R-squared being 0.0596, the linear regression model has very little to no correlation between the Temperature and Humidity.

```r
#e. Plot the residuals and write a thorough explanation.
plot(SmokeDetection$Temperature.C., SmokeDetection$Humidity...,
main="Temperature and Humidity", xlab="Temperature", ylab="Humidity")
abline(lm1)
pred <- predict(lm1, newdata=test)

## Warning: 'newdata' had 12526 rows but variables found have 50104 rows

points(test$Temperature.C., test$Humidity..., pch=0)
segments(test$Temperature.C., test$Humidity..., pred, col="red")
```



Residuals tell us how far off our predictions were. Here, the residuals severely deviate from our predicted data. One thing it does tell us however, is that after a certain temperature, the humidity decreases as the temperature increases. This idea is reinforced by there being fewer residuals on that downward slope.

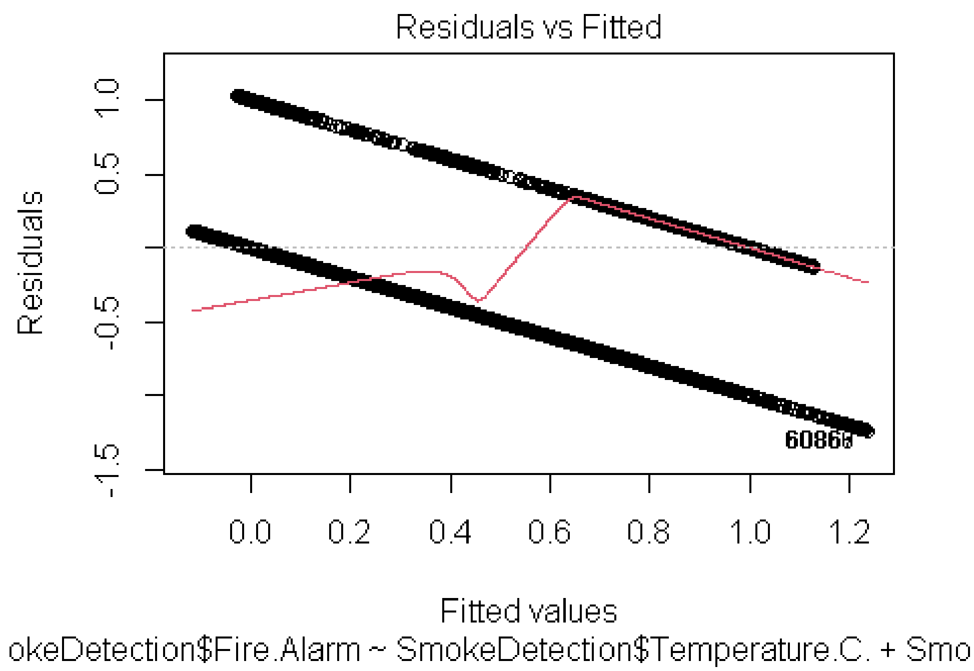
```
# f. Build a multiple linear regression model (multiple predictors), output
the summary and
lm2 <- lm(SmokeDetection$Fire.Alarm~SmokeDetection$Temperature.C. +
SmokeDetection$Humidity..., data = train)
lm2

##
## Call:
## lm(formula = SmokeDetection$Fire.Alarm ~ SmokeDetection$Temperature.C. +
##     SmokeDetection$Humidity..., data = train)
##
## Coefficients:
##                   (Intercept)  SmokeDetection$Temperature.C.
##                     -0.196034                      -0.002219
##    SmokeDetection$Humidity...
##                      0.019491

summary(lm2)

##
## Call:
## lm(formula = SmokeDetection$Fire.Alarm ~ SmokeDetection$Temperature.C. +
##     SmokeDetection$Humidity..., data = train)
```

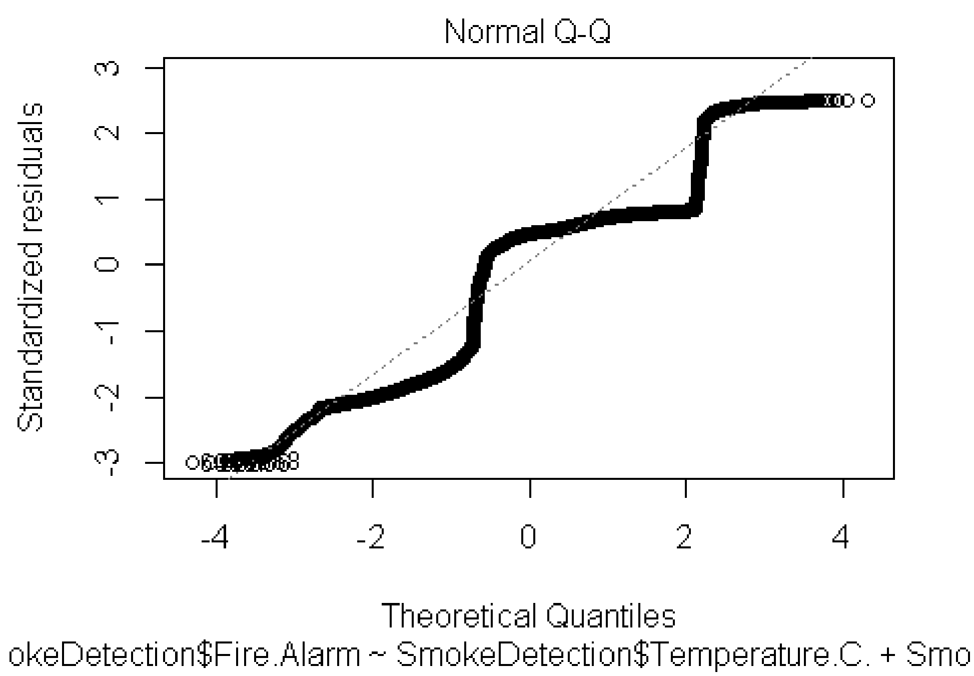
```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.2355 -0.2172  0.1959  0.2605  1.0262
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -0.1960341  0.0100803  -19.45   <2e-16 ***
## SmokeDetection$Temperature.C. -0.0022186  0.0001184  -18.73   <2e-16 ***
## SmokeDetection$Humidity...     0.0194912  0.0001918  101.60   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4128 on 62627 degrees of freedom
## Multiple R-squared:  0.1646, Adjusted R-squared:  0.1645
## F-statistic:  6168 on 2 and 62627 DF,  p-value: < 2.2e-16
```

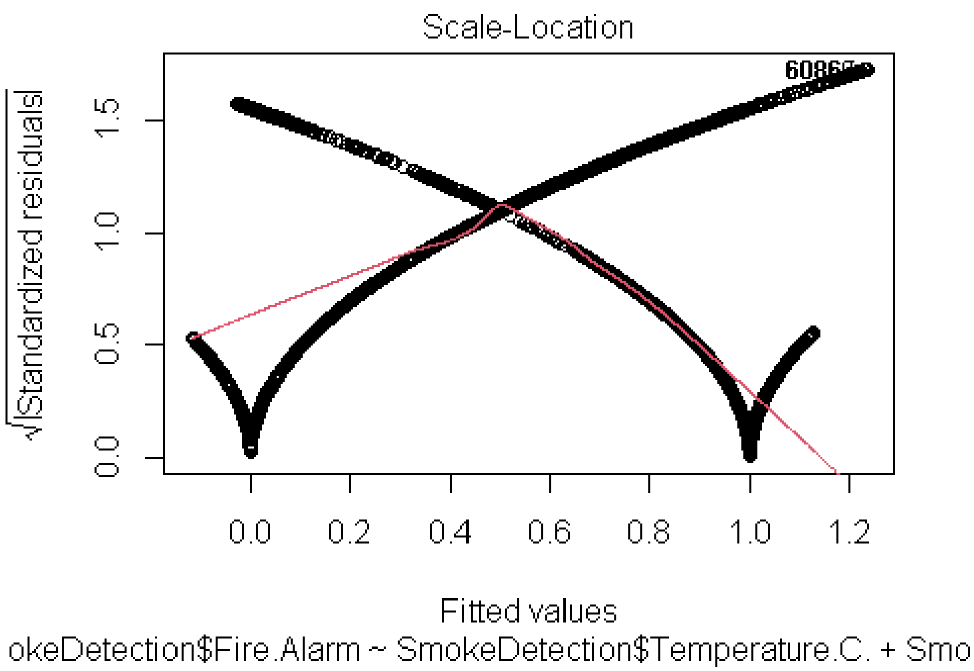
```
pred2 <- predict(lm2, newdata=test)
```

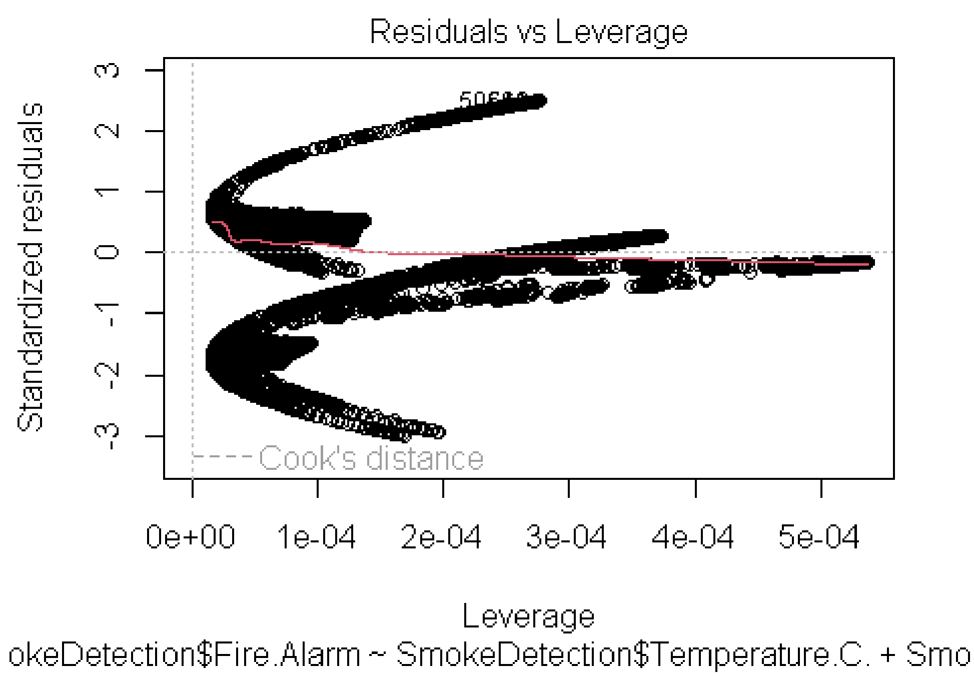
```
## Warning: 'newdata' had 12526 rows but variables found have 62630 rows
```

```
plot(lm2)
```


Residuals vs Fitted
Residuals
1.0
0.5
-0.5
-1.5
0.0
0.2
0.4
0.6
0.8
1.0
1.2
60868
Fitted values
okeDetection$Fire.Alarm ~ SmokeDetection$Temperature.C. + Smol

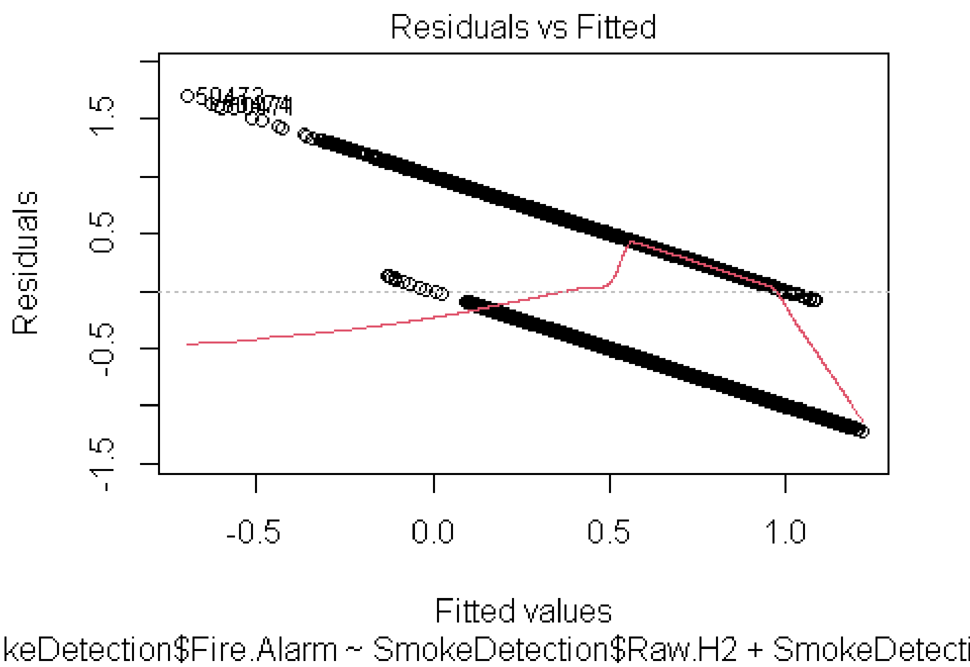

Normal Q-Q
Standardized residuals
3
2
1
0
-1
-2
-3
-4
-2
0
2
4
Theoretical Quantiles
okeDetection$Fire.Alarm ~ SmokeDetection$Temperature.C. + Smol

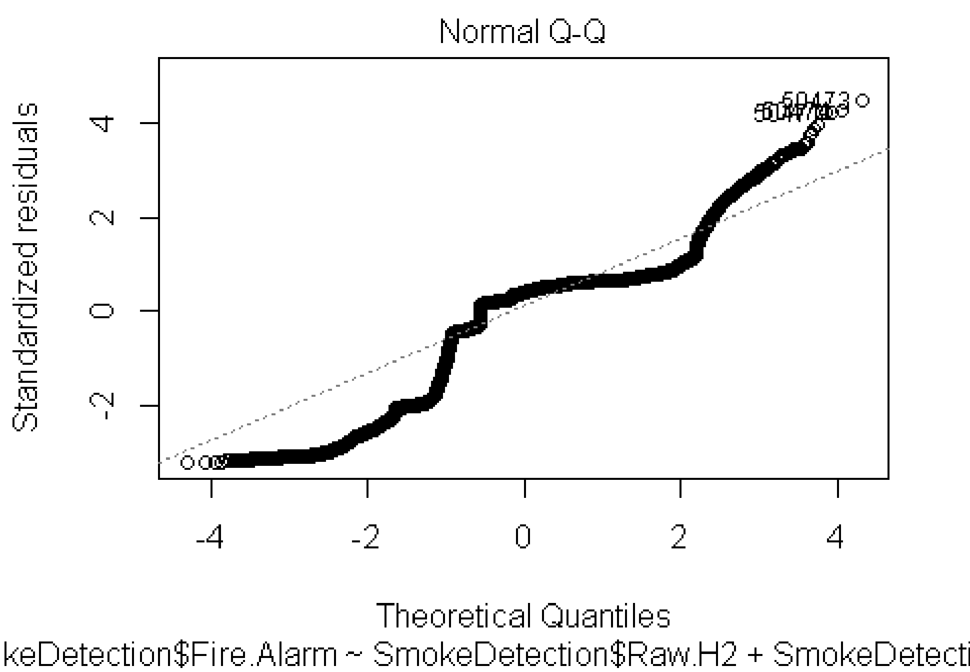


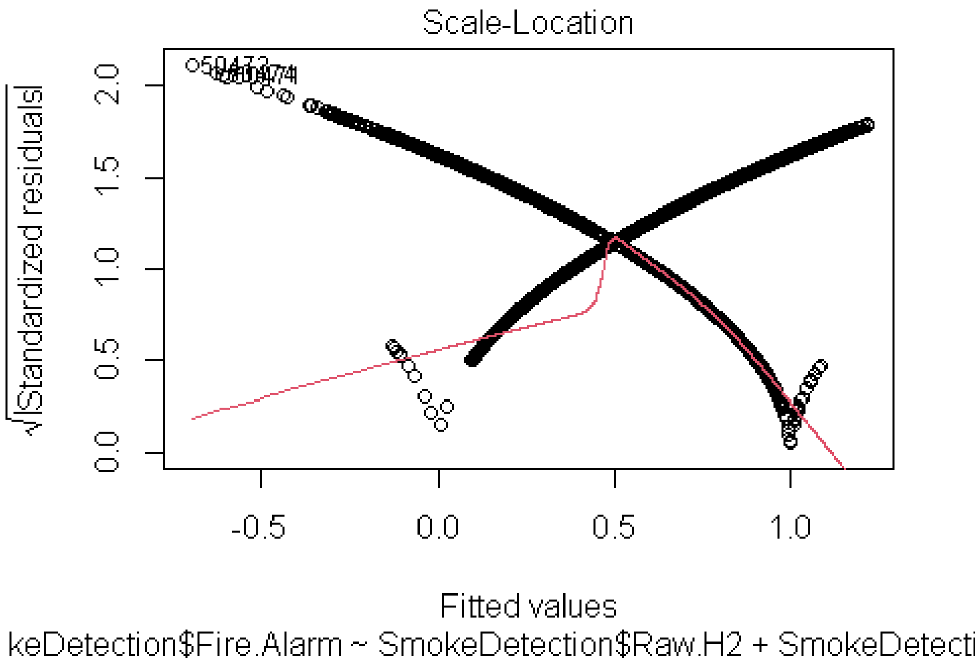


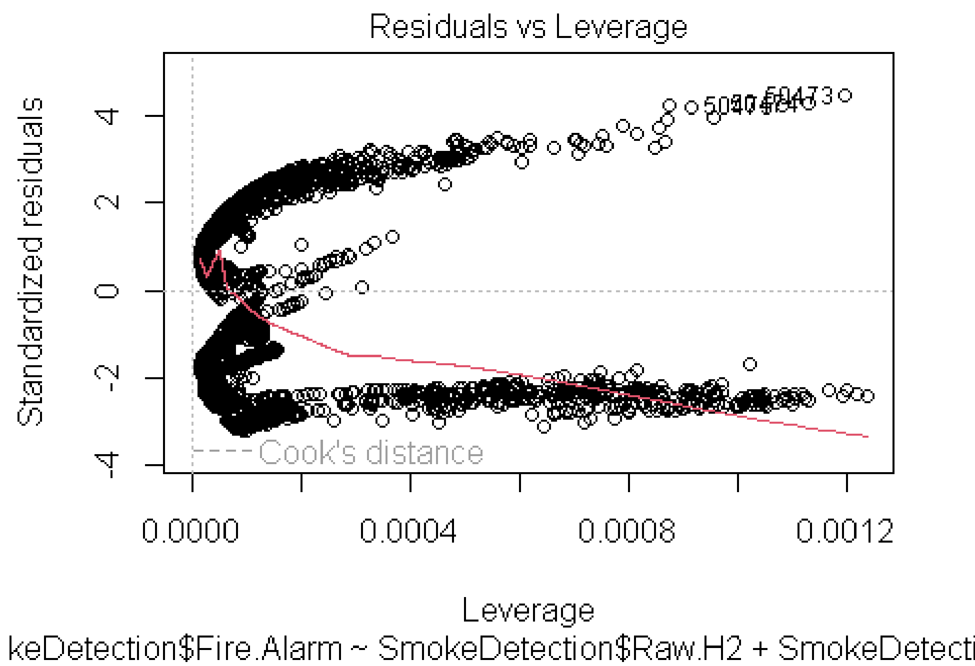

```
# g. Build a third linear regression model with a different combination of
predictors
lm3 <- lm(SmokeDetection$Fire.Alarm~SmokeDetection$Raw.H2 +
```

```
SmokeDetection$Raw.Ethanol, data = train)
lm3

##
## Call:
## lm(formula = SmokeDetection$Fire.Alarm ~ SmokeDetection$Raw.H2 +
##     SmokeDetection$Raw.Ethanol, data = train)
##
## Coefficients:
##                (Intercept)       SmokeDetection$Raw.H2
##                 -0.8409301                   0.0008881
## SmokeDetection$Raw.Ethanol
##                 -0.0005031

summary(lm3)

##
## Call:
## lm(formula = SmokeDetection$Fire.Alarm ~ SmokeDetection$Raw.H2 +
##     SmokeDetection$Raw.Ethanol, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.2166 -0.1322  0.1518  0.2352  1.6987
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                -8.409e-01  7.247e-02   -11.6   <2e-16 ***
## SmokeDetection$Raw.H2       8.881e-04  7.204e-06   123.3   <2e-16 ***
## SmokeDetection$Raw.Ethanol -5.031e-04  3.220e-06  -156.2   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3809 on 62627 degrees of freedom
## Multiple R-squared:  0.2886, Adjusted R-squared:  0.2886
## F-statistic: 1.271e+04 on 2 and 62627 DF,  p-value: < 2.2e-16

pred3 <- predict(lm3, newdata=test)

## Warning: 'newdata' had 12526 rows but variables found have 62630 rows

plot(lm3)
```


Residuals vs Fitted
Residuals
Fitted values
keDetection$Fire.Alarm ~ SmokeDetection$Raw.H2 + SmokeDetecti
-1.5
-0.5
0.5
1.5
-0.5
0.0
0.5
1.0
50473


Normal Q-Q
Standardized residuals
Theoretical Quantiles
keDetection$Fire.Alarm ~ SmokeDetection$Raw.H2 + SmokeDetecti
-2
0
2
4
-4
-2
0
2
4
50473






```
# The data is a little better, most noticably in the last residual plot.
```

When you compare the Results, the third linear model greatly increased R-squared value of 0.28. Additionally, if you look at all the residual plots and compare them, the third linear model has a much closer fitting line representing the residuals. Notably, the third graph has the best fit line of them all, which is a drastic difference from linear model 2 and 1.