hw_2.2 Elizaveta Grigoreva

```
library(data.table)
library(dplyr)
```

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

```
## The following objects are masked from 'package:data.table':
##
##     between, first, last
```

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

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

```
library(ggplot2)
library(ggpubr)
library(tidyr)
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
## The following objects are masked from 'package:data.table':
##
##     dcast, melt
```

```
library(plyr)
```

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

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

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

```
##
## Attaching package: 'plyr'
```

```
## The following object is masked from 'package:ggpubr':
##
##     mutate
```

```
## The following objects are masked from 'package:dplyr':
##
```

```
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

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

```
library(magrittr)
```

```
##
## Attaching package: 'magrittr'
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
library(corrplot)
```

```
## corrplot 0.84 loaded
```

Load and clean dataset

```
aq <- read.csv2("/Users/Lisa/Downloads/AirQualityUCI/AirQualityUCI.csv", header=T)
aq_cleaned <-  aq  %>% select_if(~sum(!is.na(.)) > 0)  %>% drop_na()
airq_fil <- aq_cleaned %>% filter_all(all_vars(. != -200))
air_long <- melt(airq_fil)
```

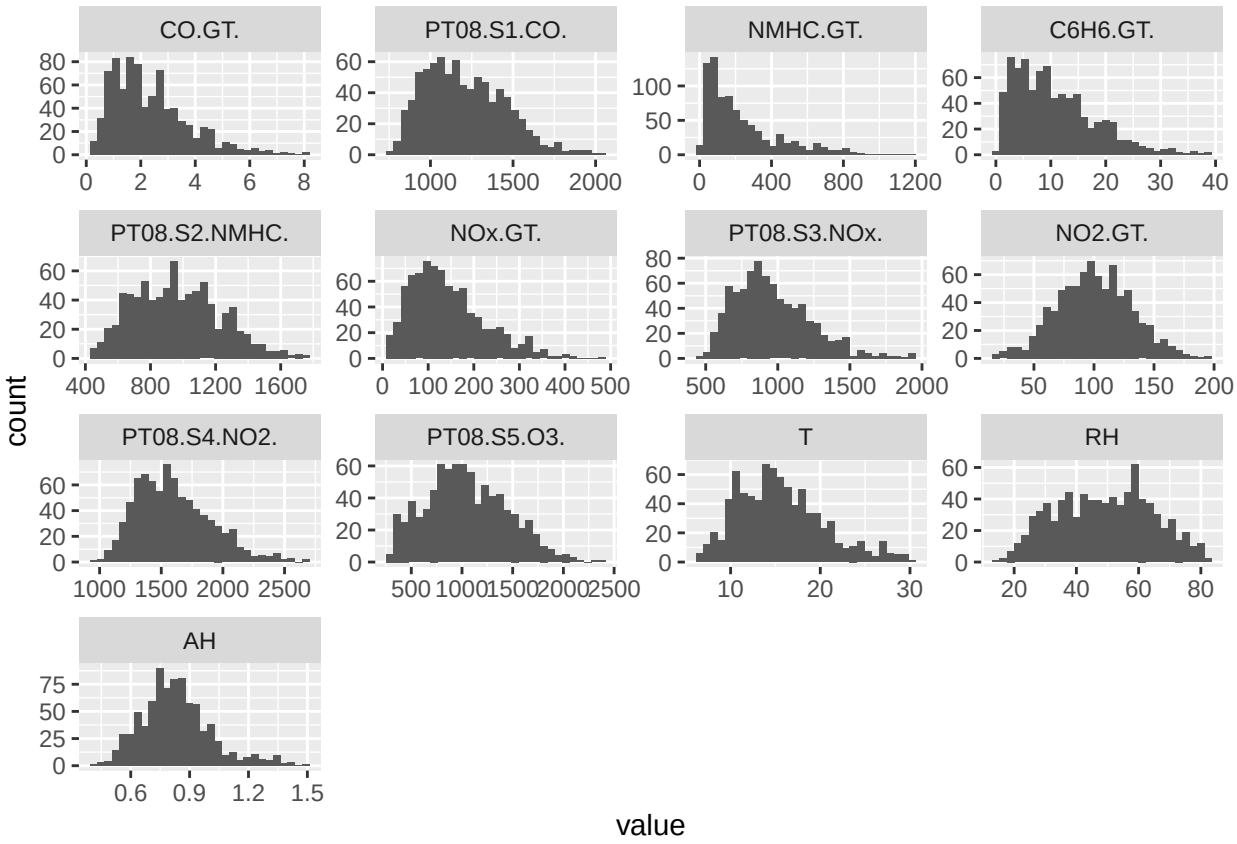
```
## Using Date, Time as id variables
```

```
head(air_long)
```

```
##         Date     Time variable value
## 1 10/03/2004 18.00.00   CO.GT.   2.6
## 2 10/03/2004 19.00.00   CO.GT.   2.0
## 3 10/03/2004 20.00.00   CO.GT.   2.2
## 4 10/03/2004 21.00.00   CO.GT.   2.2
## 5 10/03/2004 22.00.00   CO.GT.   1.6
## 6 10/03/2004 23.00.00   CO.GT.   1.2
```

```
ggplot(air_long, aes(value)) +
  geom_histogram() +
  facet_wrap(~variable, scales = "free")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
aq <- aq[,c(1:4,6:15)]
aq <- drop_na(aq)
air_long <- melt(aq)
```

```
## Using Date, Time as id variables
```

```
air_long <- air_long[,-c(1,2)]
summary(aq)
```

```
##          Date             Time          CO.GT.          PT08.S1.CO.
##  01/01/2005:  24   00.00.00: 390   Min.   :-200.00   Min.   :-200
##  01/02/2005:  24   01.00.00: 390   1st Qu.:   0.60   1st Qu.: 921
##  01/03/2005:  24   02.00.00: 390   Median :   1.50   Median :1053
##  01/04/2004:  24   03.00.00: 390   Mean   : -34.21   Mean   :1049
##  01/04/2005:  24   04.00.00: 390   3rd Qu.:   2.60   3rd Qu.:1221
##  01/05/2004:  24   05.00.00: 390   Max.   :  11.90   Max.   :2040
##  (Other)   :9213   (Other) :7017
##     C6H6.GT.         PT08.S2.NMHC.       NOx.GT.         PT08.S3.NOx.
##  Min.   :-200.000   Min.   :-200.0   Min.   :-200.0   Min.   :-200
##  1st Qu.:   4.000   1st Qu.: 711.0   1st Qu.:  50.0   1st Qu.: 637
##  Median :   7.900   Median : 895.0   Median : 141.0   Median : 794
##  Mean   :   1.866   Mean   : 894.6   Mean   : 168.6   Mean   : 795
##  3rd Qu.:  13.600   3rd Qu.:1105.0   3rd Qu.: 284.0   3rd Qu.: 960
##  Max.   :  63.700   Max.   :2214.0   Max.   :1479.0   Max.   :2683
##
##     NO2.GT.          PT08.S4.NO2.    PT08.S5.O3.           T
##  Min.   :-200.00   Min.   :-200   Min.   :-200.0   Min.   :-200.000
##  1st Qu.:  53.00   1st Qu.:1185   1st Qu.: 700.0   1st Qu.:  10.900
```

```
##  Median :  96.00   Median :1446   Median : 942.0   Median :  17.200
##  Mean   :  58.15   Mean   :1391   Mean   : 975.1   Mean   :   9.778
##  3rd Qu.: 133.00   3rd Qu.:1662   3rd Qu.:1255.0   3rd Qu.:  24.100
##  Max.   : 340.00   Max.   :2775   Max.   :2523.0   Max.   :  44.600
##
##        RH                AH
##  Min.   :-200.00   Min.   :-200.0000
##  1st Qu.:  34.10   1st Qu.:   0.6923
##  Median :  48.60   Median :   0.9768
##  Mean   :  39.49   Mean   :  -6.8376
##  3rd Qu.:  61.90   3rd Qu.:   1.2962
##  Max.   :  88.70   Max.   :   2.2310
##
```

Normalization

```
normalize <- function(x) {
  return ((x - min(x)) / (max(x) - min(x)))
}
air_norm <- as.data.frame(lapply(aq[3:14], normalize))
air_long <- melt(air_norm)
```

```
## No id variables; using all as measure variables
```

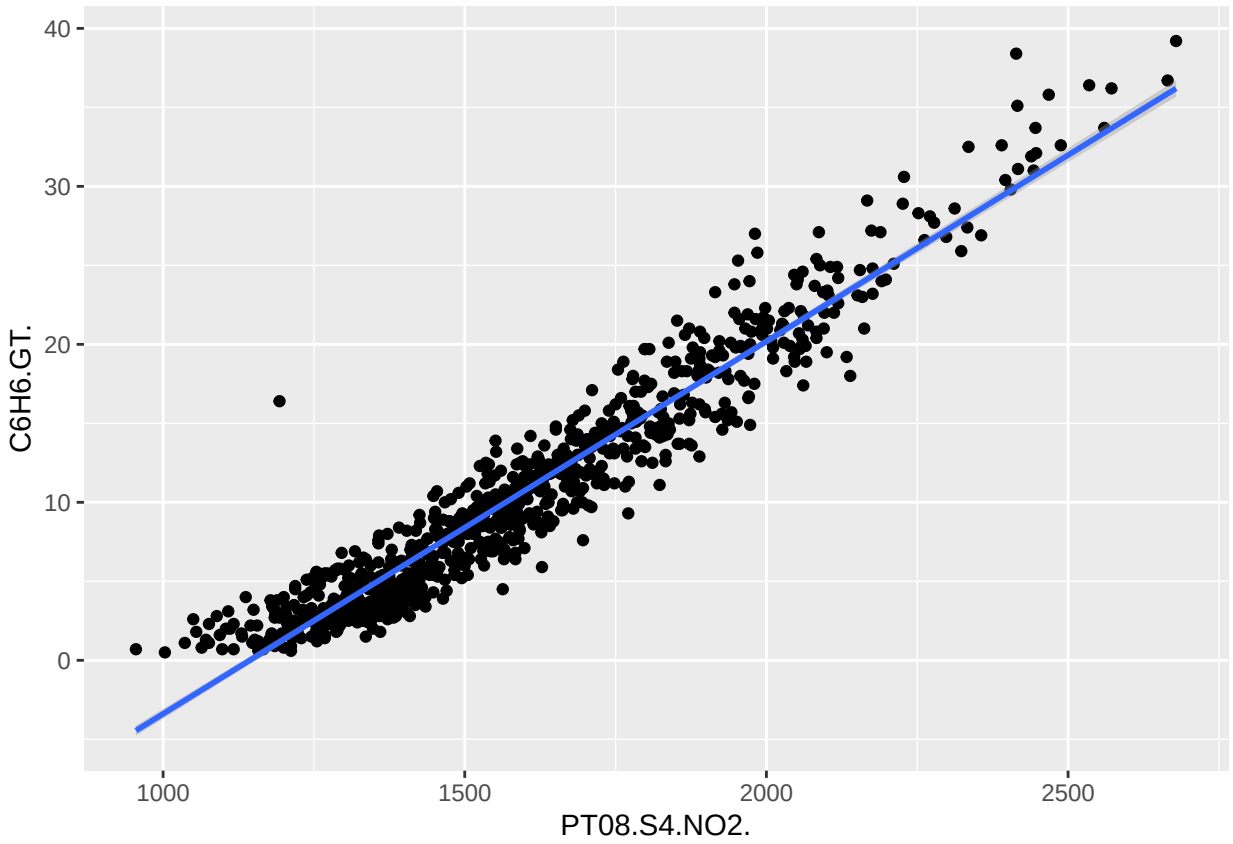
```
head(air_norm)
```

```
##      CO.GT. PT08.S1.CO.  C6H6.GT. PT08.S2.NMHC.   NOx.GT. PT08.S3.NOx.
## 1 0.9561114   0.6964286 0.8035647     0.5161558 0.2179869    0.4356573
## 2 0.9532798   0.6660714 0.7940842     0.4784590 0.1804646    0.4765869
## 3 0.9542237   0.7151786 0.7925673     0.4718310 0.1971412    0.4647936
## 4 0.9542237   0.7035714 0.7933257     0.4755592 0.2215605    0.4481443
## 5 0.9513922   0.6571429 0.7830868     0.4291632 0.1971412    0.4873396
## 6 0.9495045   0.6236607 0.7762609     0.3935377 0.1721263    0.5331252
##     NO2.GT. PT08.S4.NO2. PT08.S5.O3.         T        RH        AH
## 1 0.5796296    0.6359664   0.5391113 0.8732625 0.8621406 0.9927153
## 2 0.5407407    0.5912605   0.4304076 0.8720360 0.8579841 0.9925555
## 3 0.5814815    0.5899160   0.4678663 0.8663123 0.8798060 0.9926777
## 4 0.5962963    0.5996639   0.5152405 0.8626329 0.9005888 0.9928582
## 5 0.5851852    0.5680672   0.4810870 0.8634505 0.8992033 0.9928686
## 6 0.5481481    0.5354622   0.4219611 0.8634505 0.8978178 0.9928488
```

Explore multicollinearity, choose good predictors. I choose predictors based on the previous hw (that had linear deoendency)

```
airq_fil %>%
  ggplot(aes(x= PT08.S4.NO2., y= C6H6.GT.)) +
  geom_point() +
  geom_smooth(method="lm")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

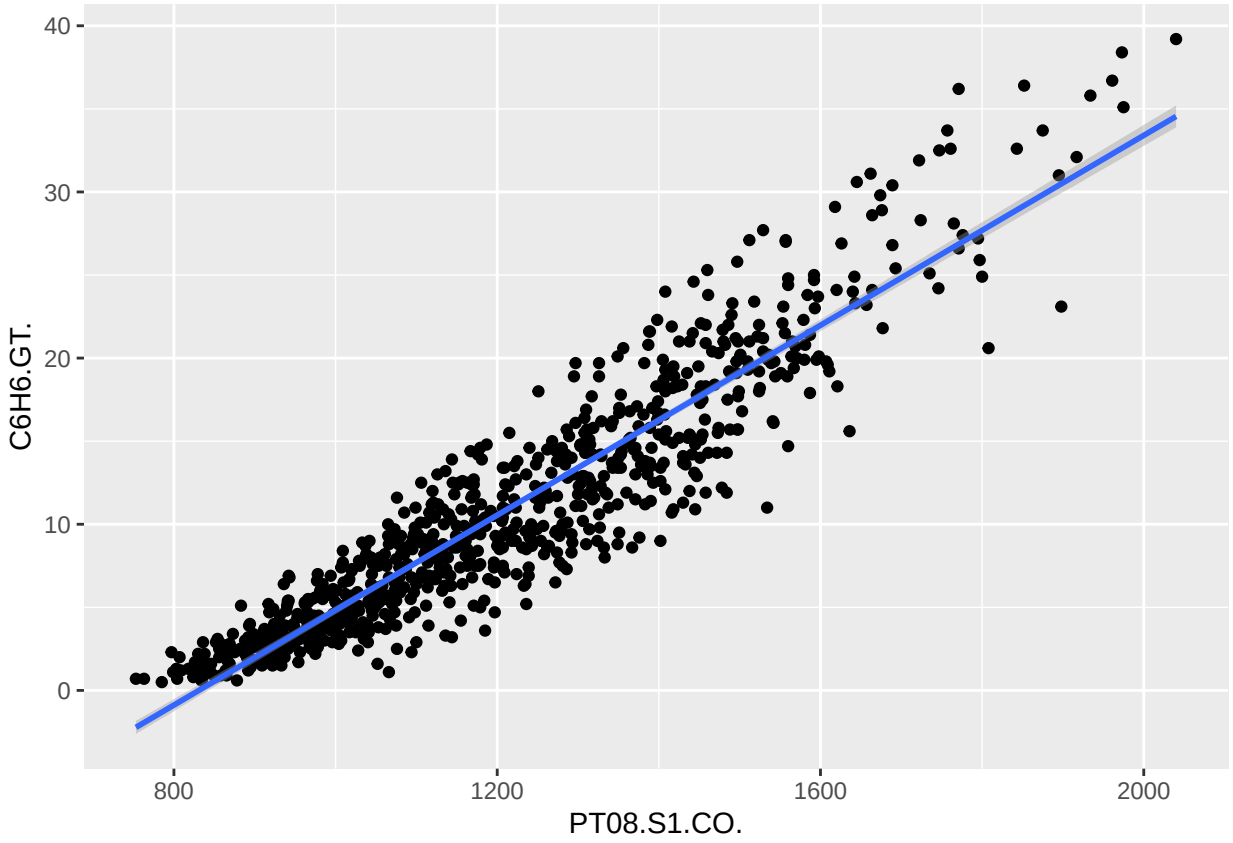



```
airq_fil  %>%
  lm(data= .,PT08.S4.NO2. ~ C6H6.GT.)%>%
  summary()
```

```
##
## Call:
## lm(formula = PT08.S4.NO2. ~ C6H6.GT., data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -628.01  -47.41    1.52   55.96  255.34
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1178.8949     5.1425  229.25   <2e-16 ***
## C6H6.GT.      39.1534     0.3933   99.56   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 83.85 on 825 degrees of freedom
## Multiple R-squared:  0.9232, Adjusted R-squared:  0.9231
## F-statistic:  9911 on 1 and 825 DF,  p-value: < 2.2e-16
```

```
airq_fil %>%
  ggplot(aes(x= PT08.S1.CO., y= C6H6.GT.)) +
  geom_point() +
  geom_smooth(method="lm")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

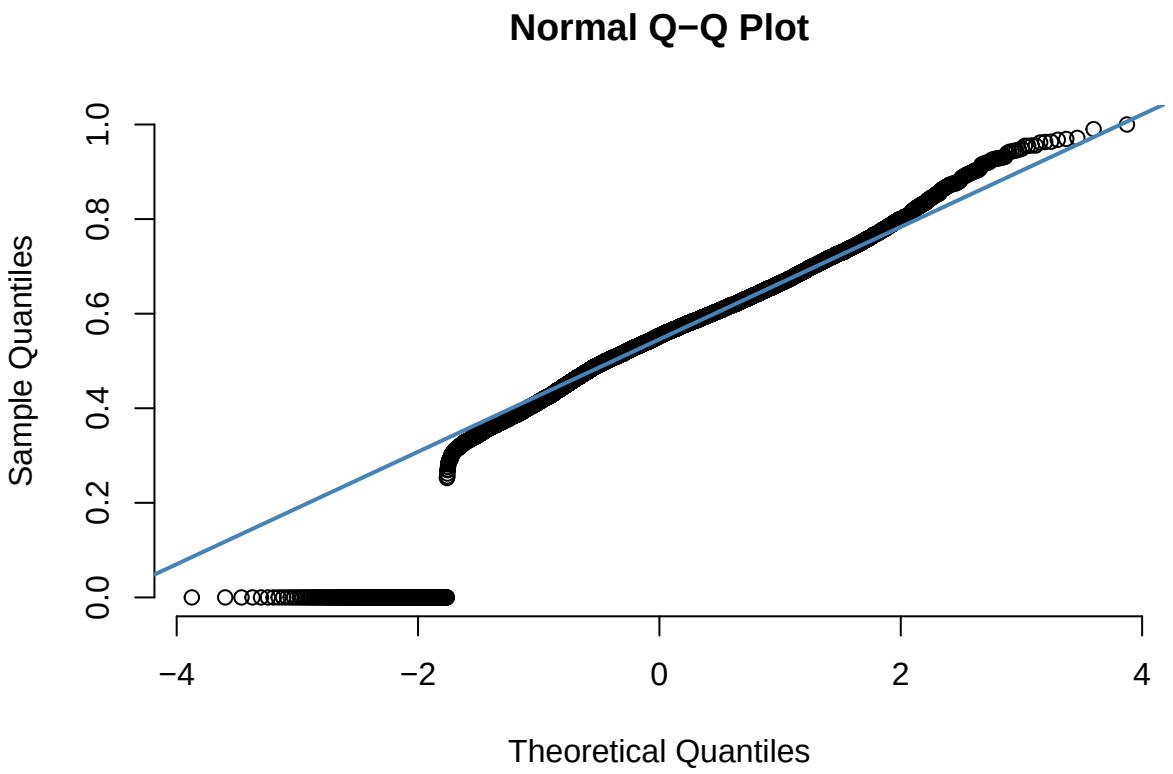



```
airq_fil  %>%
  lm(data= .,C6H6.GT. ~ PT08.S1.CO.)%>%
  summary()
```

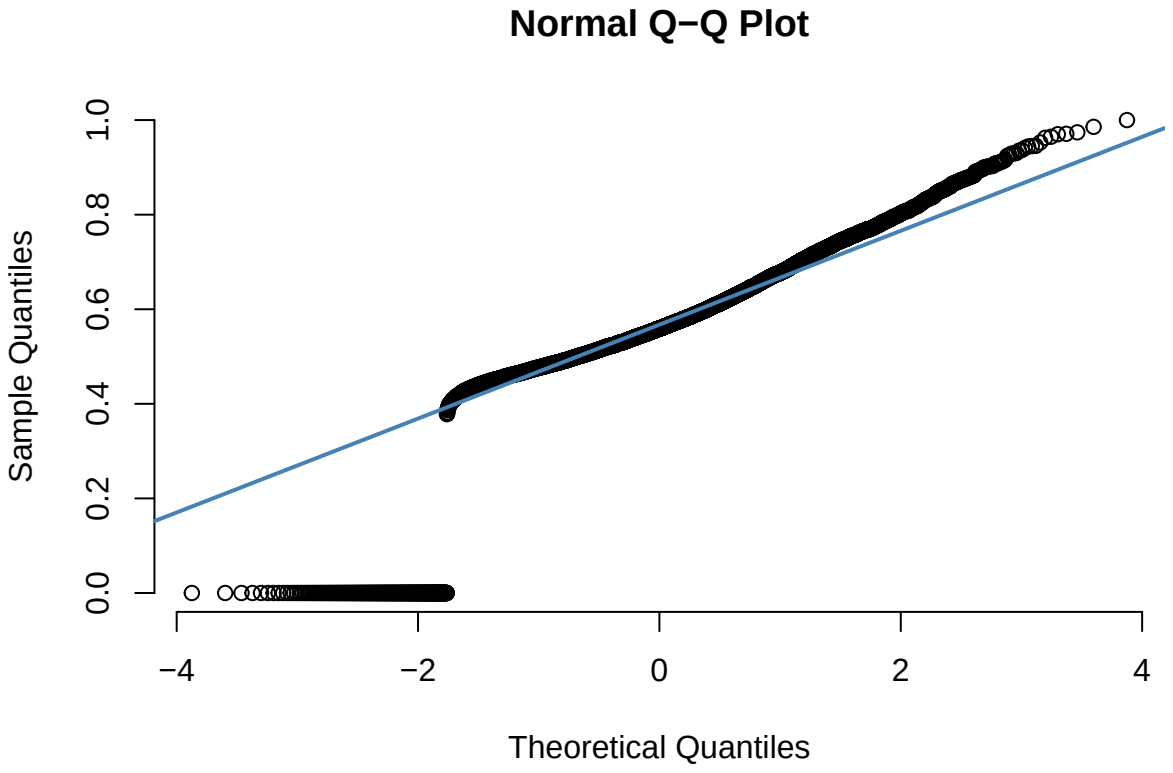
```
##
## Call:
## lm(formula = C6H6.GT. ~ PT08.S1.CO., data = .)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0888 -1.6245  0.0254  1.6468  9.3398
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.374e+01  4.790e-01  -49.56   <2e-16 ***
## PT08.S1.CO.  2.857e-02  3.888e-04   73.48   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.702 on 825 degrees of freedom
## Multiple R-squared:  0.8674, Adjusted R-squared:  0.8673
## F-statistic:  5399 on 1 and 825 DF,  p-value: < 2.2e-16
```

We can check quantiles for this data

```
qqnorm(air_norm$PT08.S4.NO2., pch = 1, frame = FALSE)
qqline(air_norm$PT08.S4.NO2., col = "steelblue", lwd = 2)
```




```
qqnorm(air_norm$PT08.S1.CO., pch = 1, frame = FALSE)
qqline(air_norm$PT08.S1.CO., col = "steelblue", lwd = 2)
```

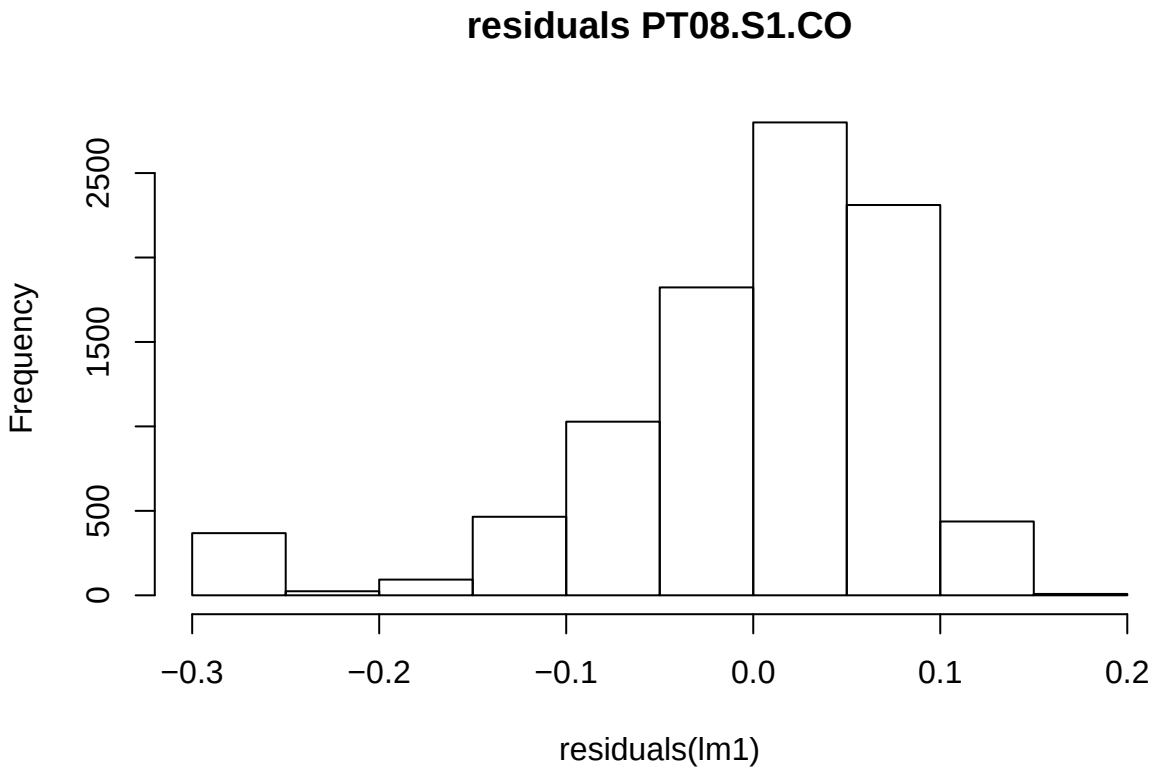


Check residuals for this predictors

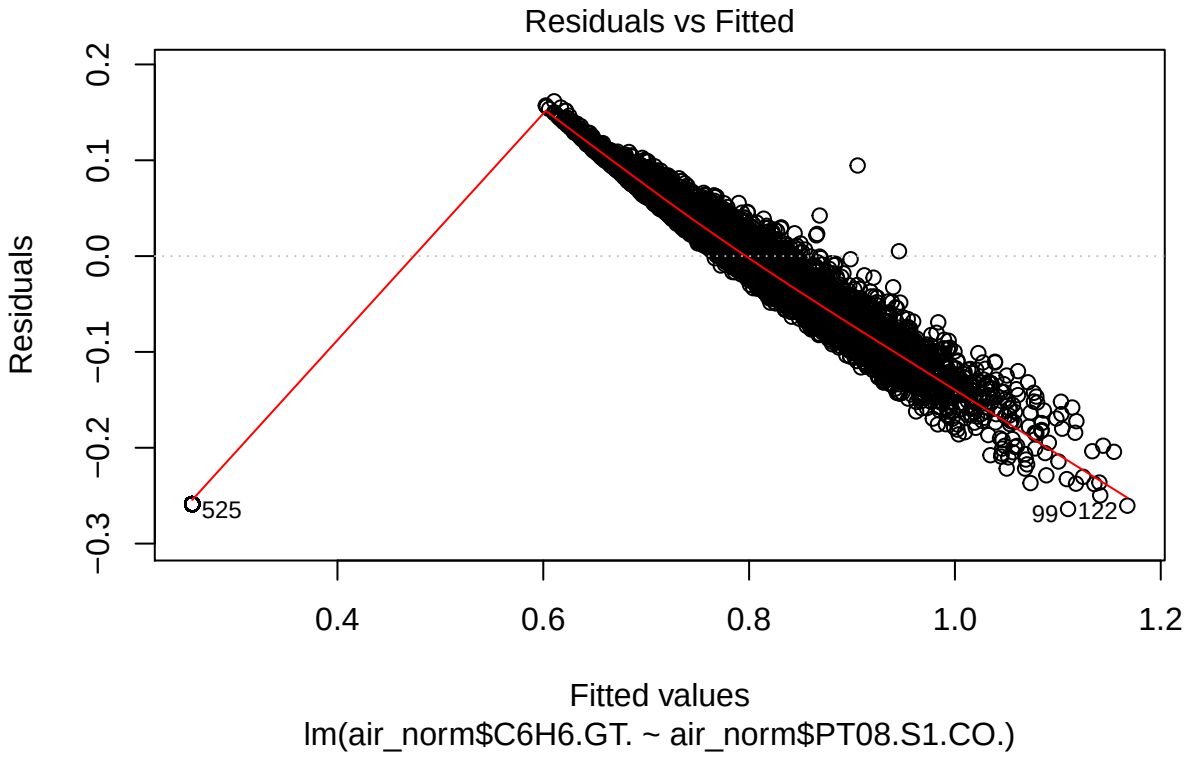
```r
lm1 <- lm(air_norm$C6H6.GT. ~ air_norm$PT08.S1.CO.)
summary(lm1)
```

```
##
## Call:
## lm(formula = air_norm$C6H6.GT. ~ air_norm$PT08.S1.CO.)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.26390 -0.03846  0.01710  0.05671  0.16154
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           0.258828   0.003320   77.97   <2e-16 ***
## air_norm$PT08.S1.CO. 0.908713   0.005756  157.87   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08198 on 9355 degrees of freedom
## Multiple R-squared:  0.7271, Adjusted R-squared:  0.727
## F-statistic: 2.492e+04 on 1 and 9355 DF,  p-value: < 2.2e-16
```

```r
hist(residuals(lm1),main ="residuals PT08.S1.CO")
```



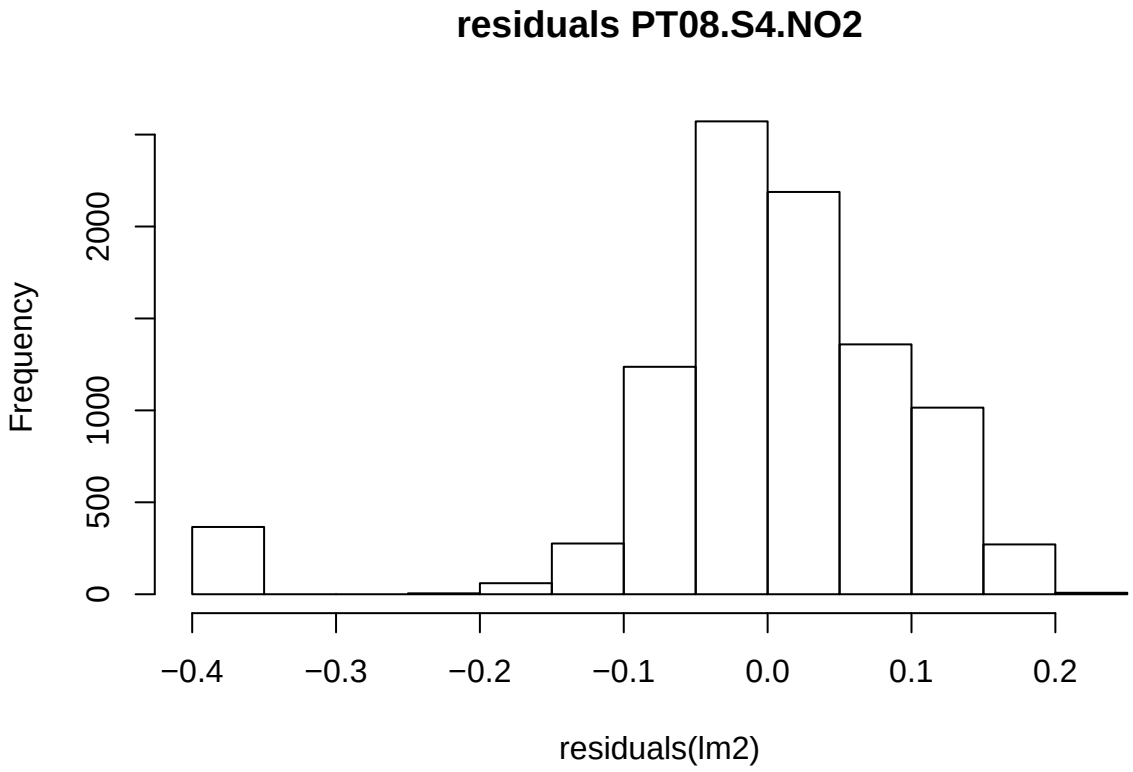

```r
plot(lm1,which=1)
```



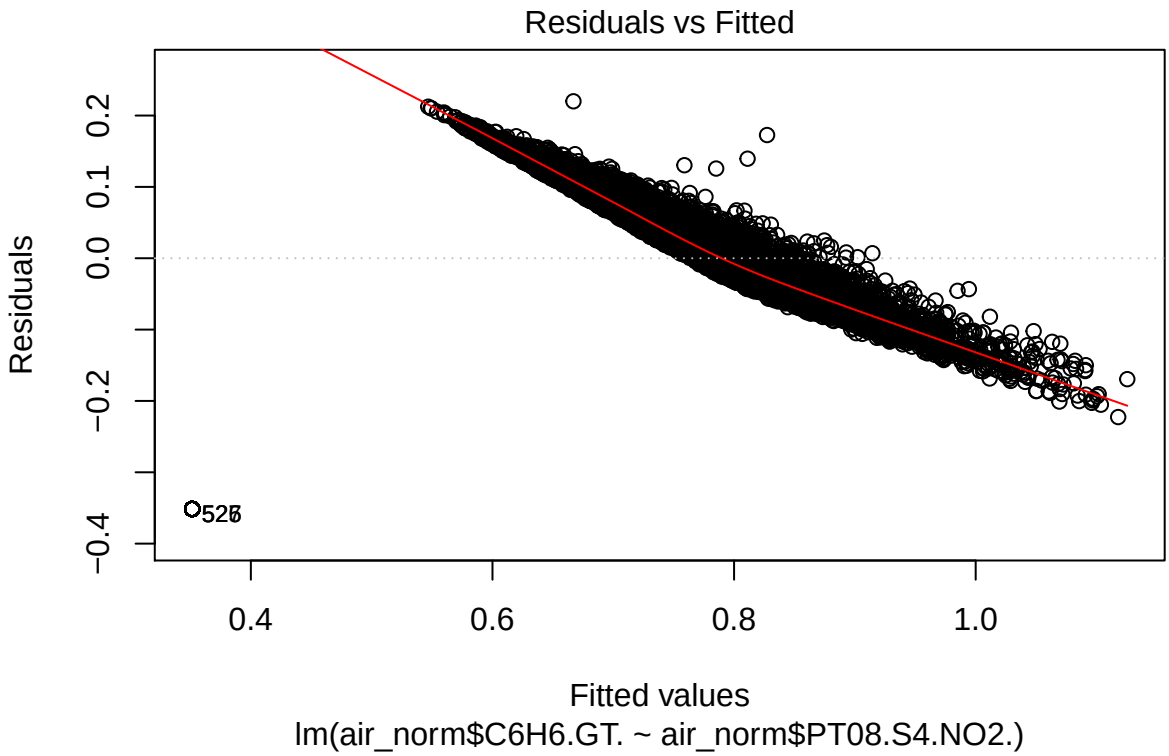

```
lm2 <- lm(air_norm$C6H6.GT. ~ air_norm$PT08.S4.NO2.)
summary(lm2)
```

```
##
## Call:
## lm(formula = air_norm$C6H6.GT. ~ air_norm$PT08.S4.NO2.)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.35143 -0.04100  0.00319  0.06002  0.21996
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           0.351427   0.003642   96.49   <2e-16 ***
## air_norm$PT08.S4.NO2. 0.774062   0.006533  118.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.09924 on 9355 degrees of freedom
## Multiple R-squared:  0.6001, Adjusted R-squared:  0.6001
## F-statistic: 1.404e+04 on 1 and 9355 DF,  p-value: < 2.2e-16
```

```
hist(residuals(lm2),main ="residuals PT08.S4.NO2")
```

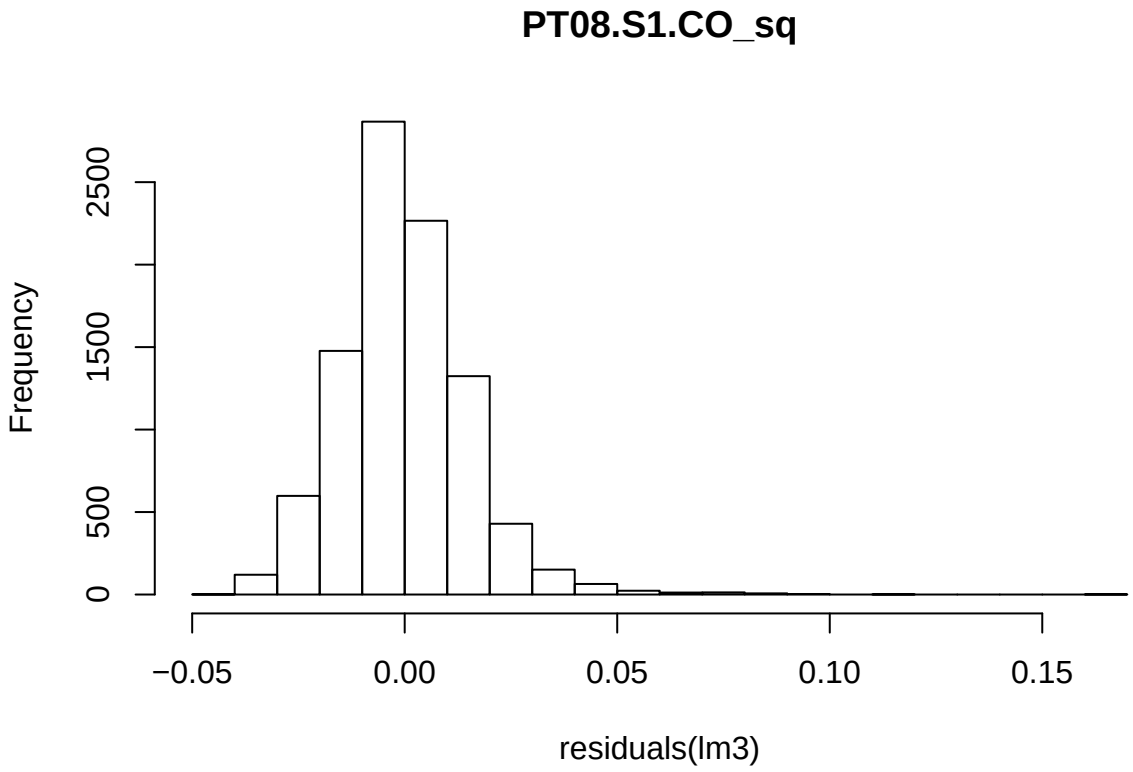



```
plot(lm2,which=1)
```

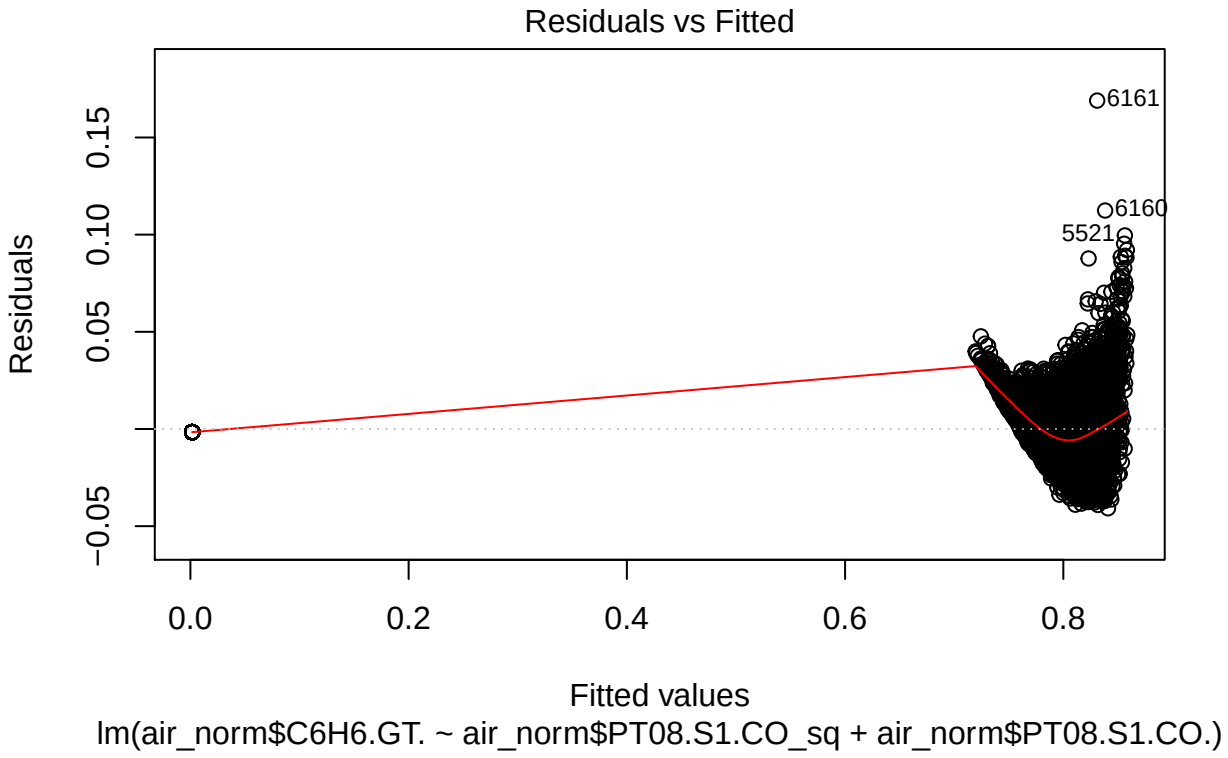


Data is normaly distributed but we have some outliers. We can try to transform our predictors using xˆ2, log or sqrt and see what will improve plot PT08.S1.CO sqrt

```r
air_norm$PT08.S1.CO_sq <- sqrt(air_norm$PT08.S1.CO.)
lm3 <- lm(air_norm$C6H6.GT. ~ air_norm$PT08.S1.CO_sq + air_norm$PT08.S1.CO.)
hist(residuals(lm3),main ="PT08.S1.CO_sq")
```




```r
plot(lm3,which=1)
```



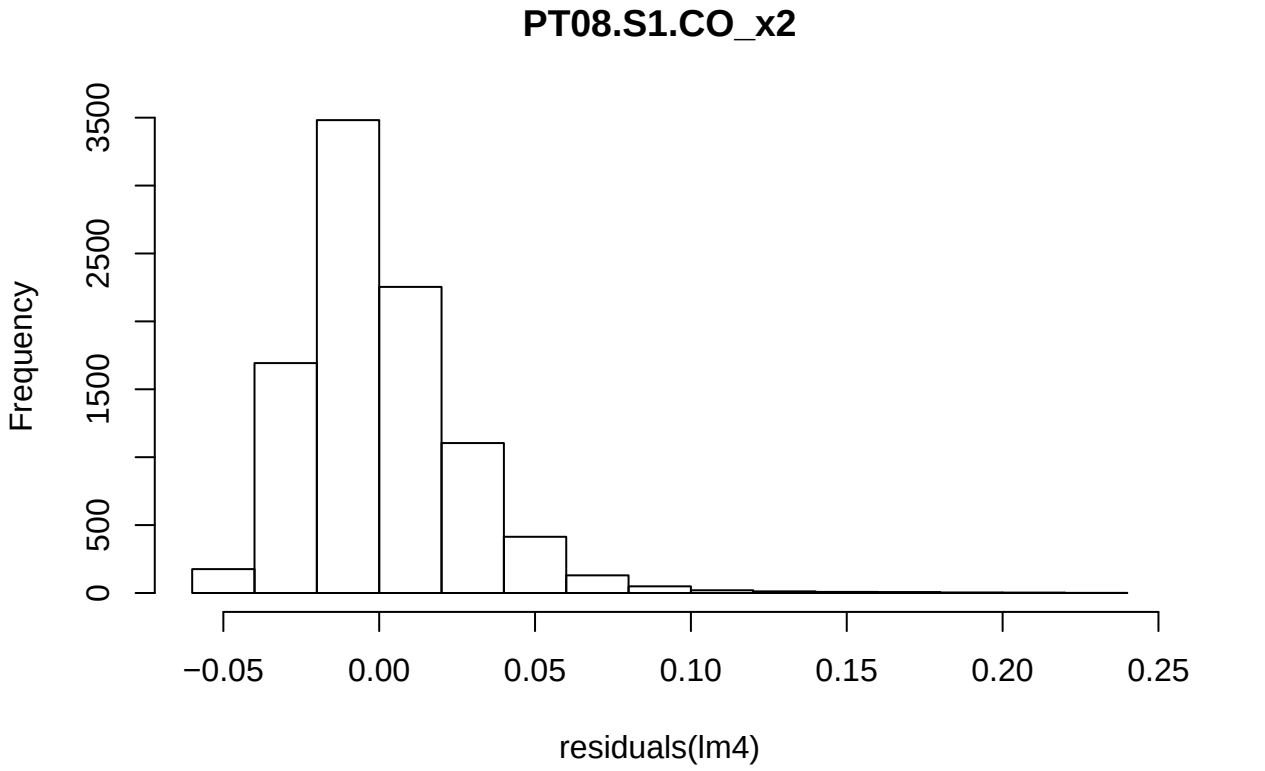
Check vif (vif >5: indicates multicollinearity)

```
vif(lm3)
```

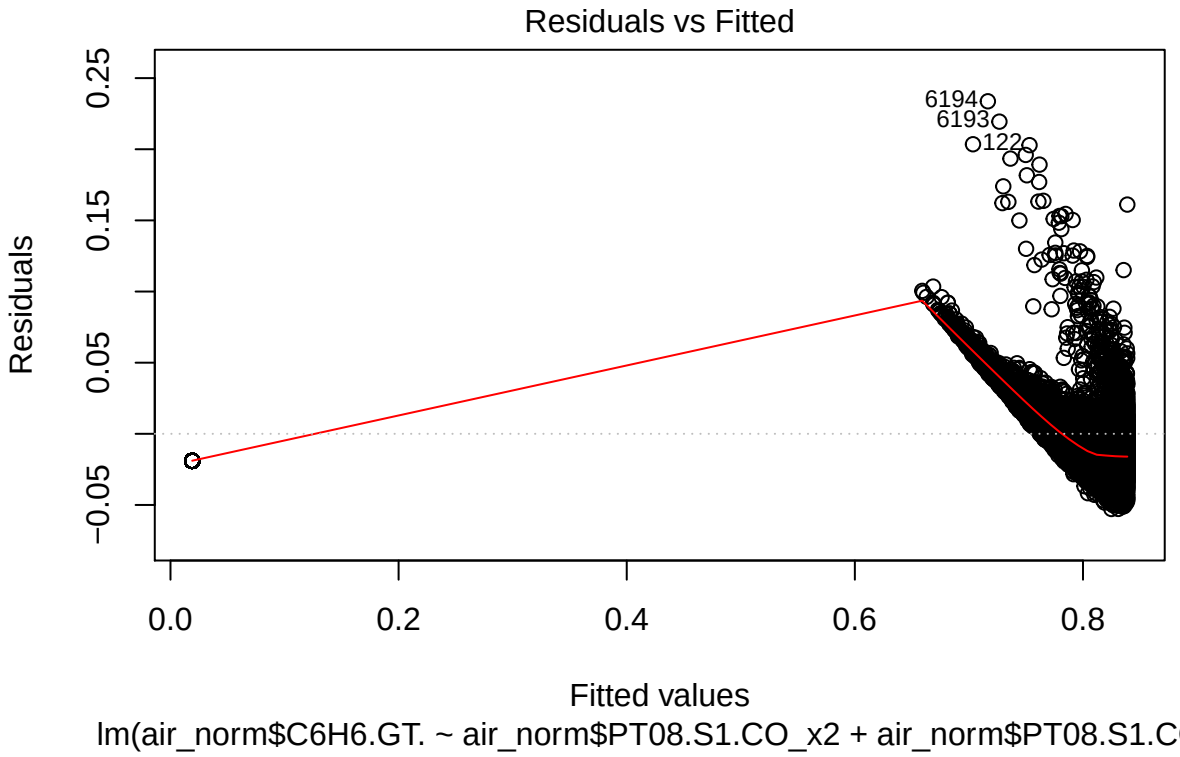
```
## air_norm$PT08.S1.CO_sq   air_norm$PT08.S1.CO.
##               10.81873               10.81873
```

PT08.S1.CO x^2

```
air_norm$PT08.S1.CO_x2 <- (air_norm$PT08.S1.CO.)^2
lm4 <- lm(air_norm$C6H6.GT. ~ air_norm$PT08.S1.CO_x2  + air_norm$PT08.S1.CO.)
hist(residuals(lm4),main ="PT08.S1.CO_x2")
```




```
plot(lm4,which=1)
```



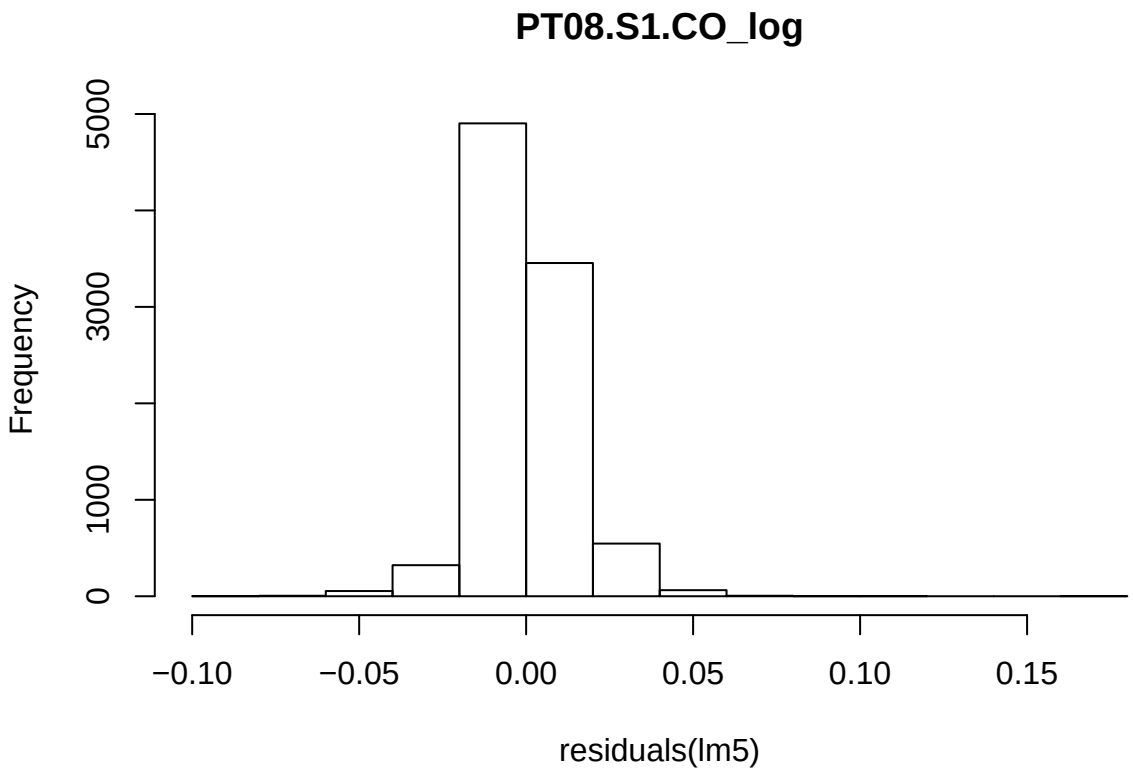
Check vif (vif >5: indicates multicollinearity) We can use x^2 transformation

```
vif(lm4)
```

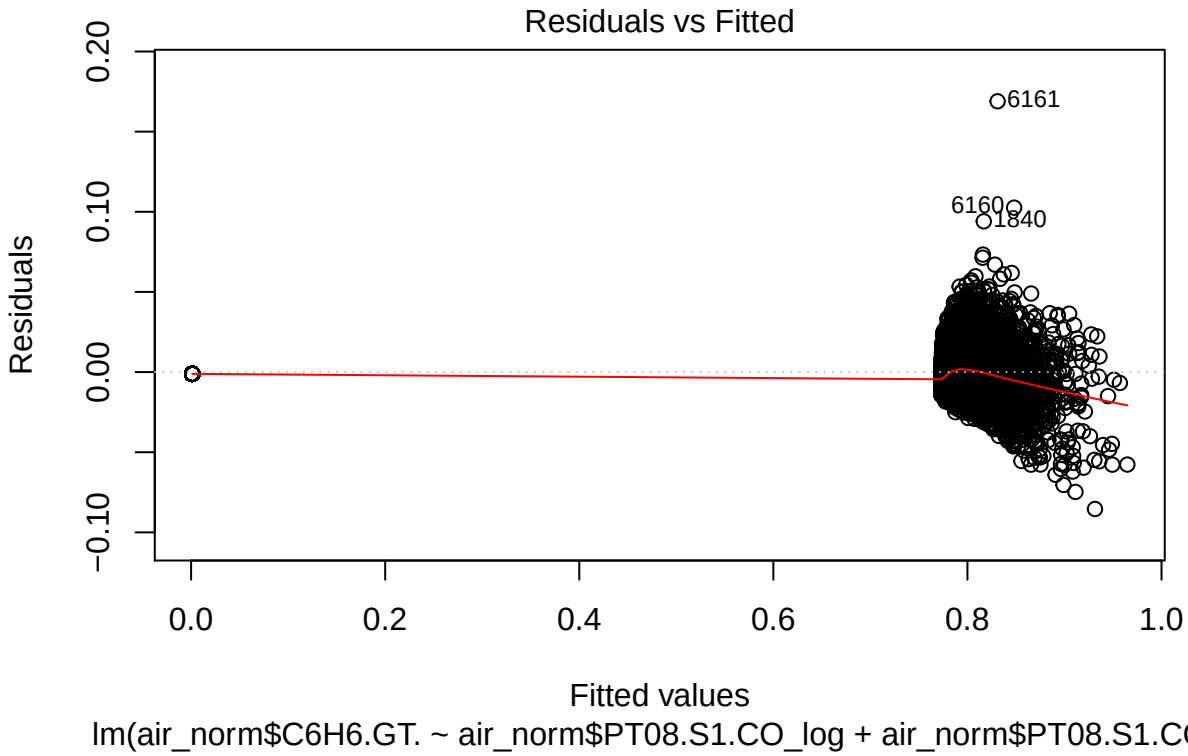
```
## air_norm$PT08.S1.CO_x2   air_norm$PT08.S1.CO.
##               8.032786               8.032786
```

PT08.S1.CO log(x)

```
air_norm$PT08.S1.CO_log <- log(air_norm$PT08.S1.CO.)
air_norm$PT08.S1.CO_log[!is.finite(air_norm$PT08.S1.CO_log)] <- 0
lm5 <- lm(air_norm$C6H6.GT. ~ air_norm$PT08.S1.CO_log  + air_norm$PT08.S1.CO.)
hist(residuals(lm5),main ="PT08.S1.CO_log")
```




```
plot(lm5,which=1)
```



Look on vif

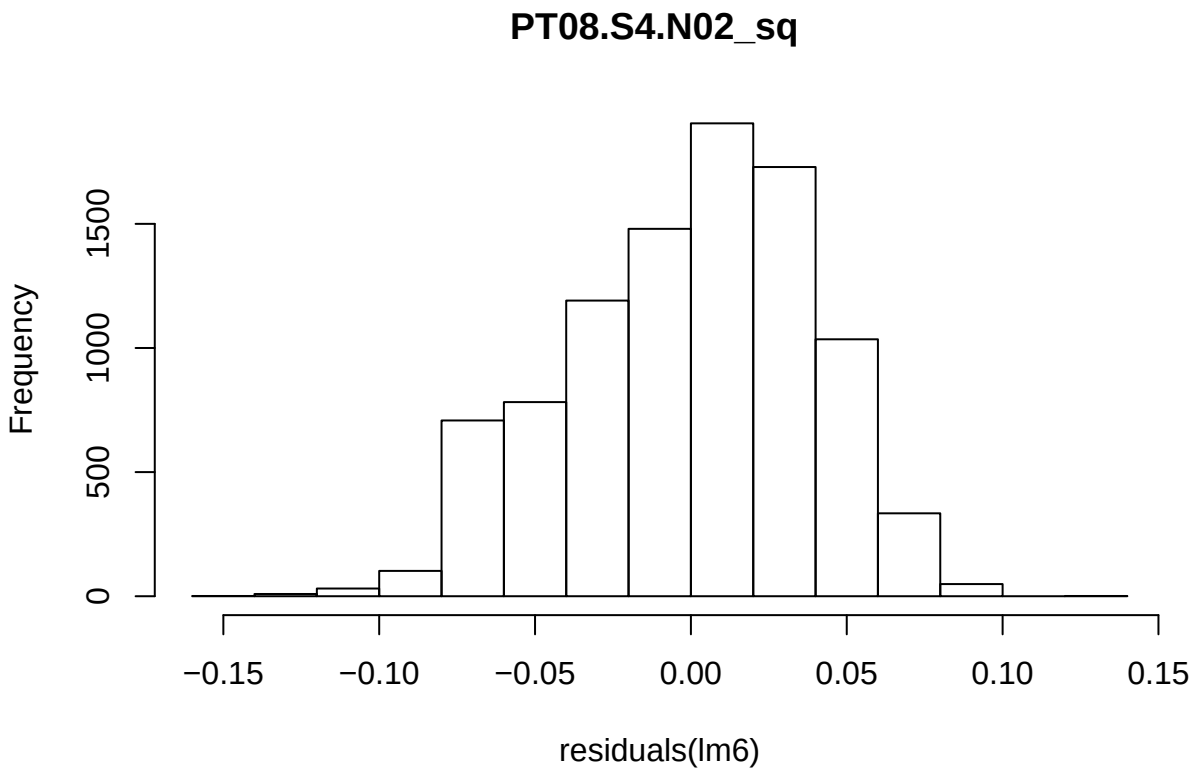
```
vif(lm5)
```

```
## air_norm$PT08.S1.CO_log     air_norm$PT08.S1.CO.
```

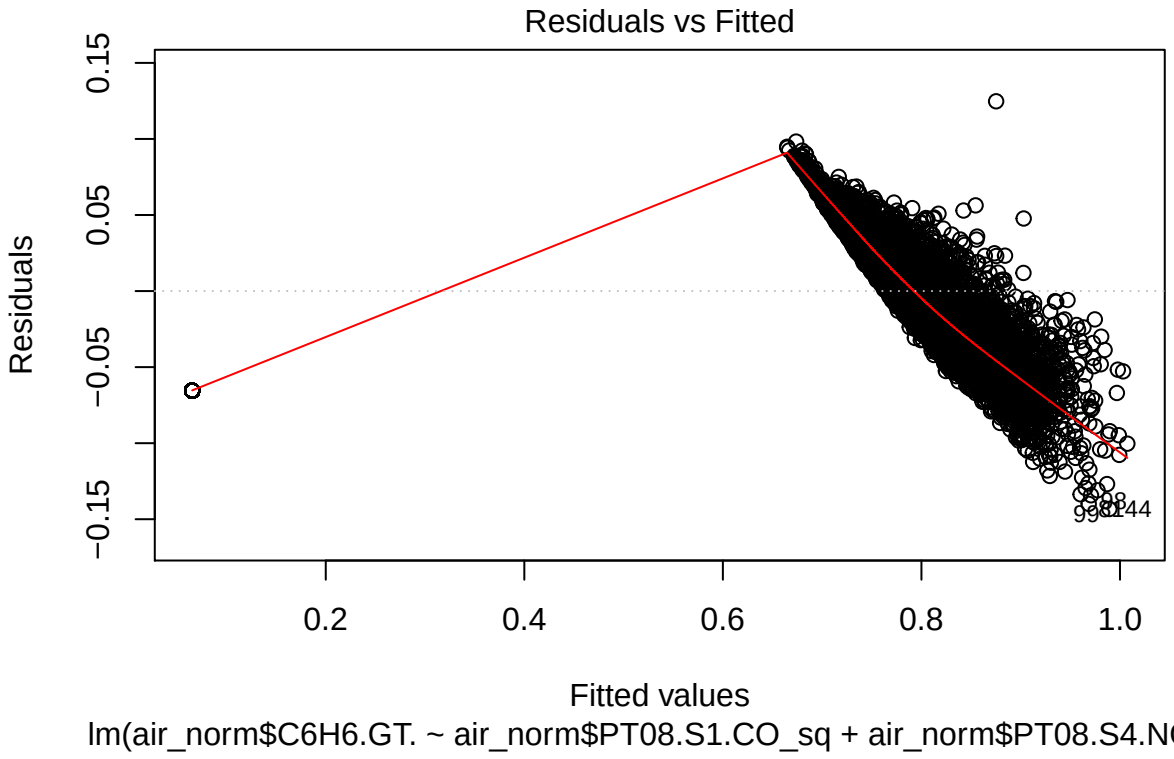
```
##                 1.01001                 1.01001
```

We can choose x^2 transformed predictor based on vif Let's look on the second one

```
air_norm$PT08.S4.NO2_sq <- sqrt(air_norm$PT08.S4.NO2.)
lm6 <- lm(air_norm$C6H6.GT. ~ air_norm$PT08.S1.CO_sq + air_norm$PT08.S4.NO2.)
hist(residuals(lm6),main ="PT08.S4.NO2_sq ")
```




```
plot(lm6,which=1)
```



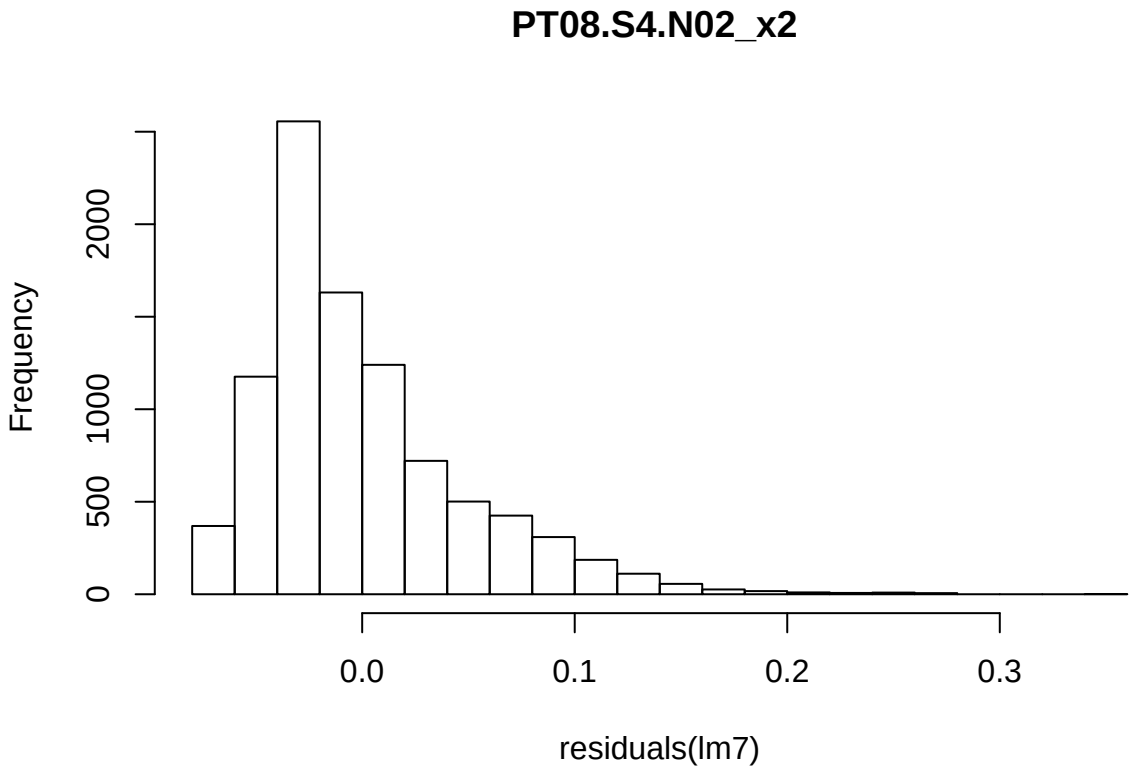
Chieck vif

```
vif(lm6)
```

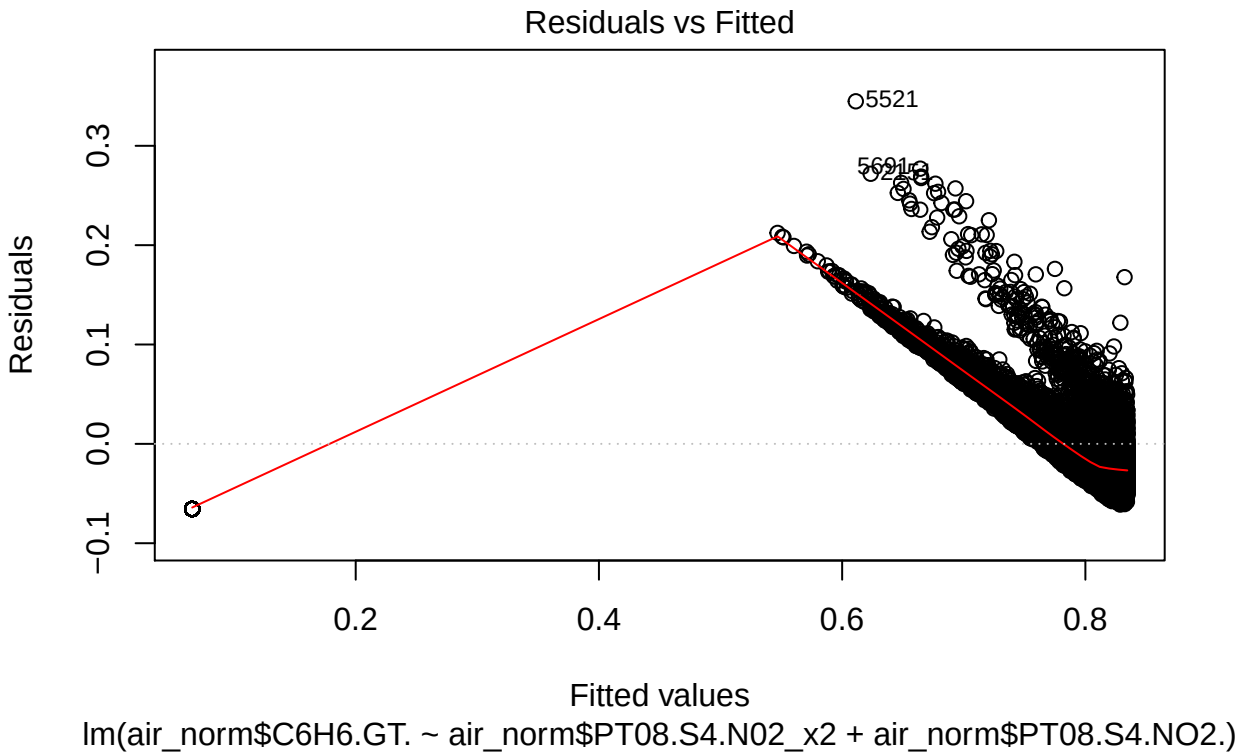
```
## air_norm$PT08.S1.CO_sq  air_norm$PT08.S4.NO2.
##               3.127223               3.127223
```

PT08.S1.CO x^2 (we cab use it)

```
air_norm$PT08.S4.NO2_x2 <- (air_norm$PT08.S4.NO2.)^2
lm7 <- lm(air_norm$C6H6.GT. ~ air_norm$PT08.S4.NO2_x2 + air_norm$PT08.S4.NO2.)
hist(residuals(lm7),main ="PT08.S4.NO2_x2 ")
```




```
plot(lm7,which=1)
```



Chieck vif

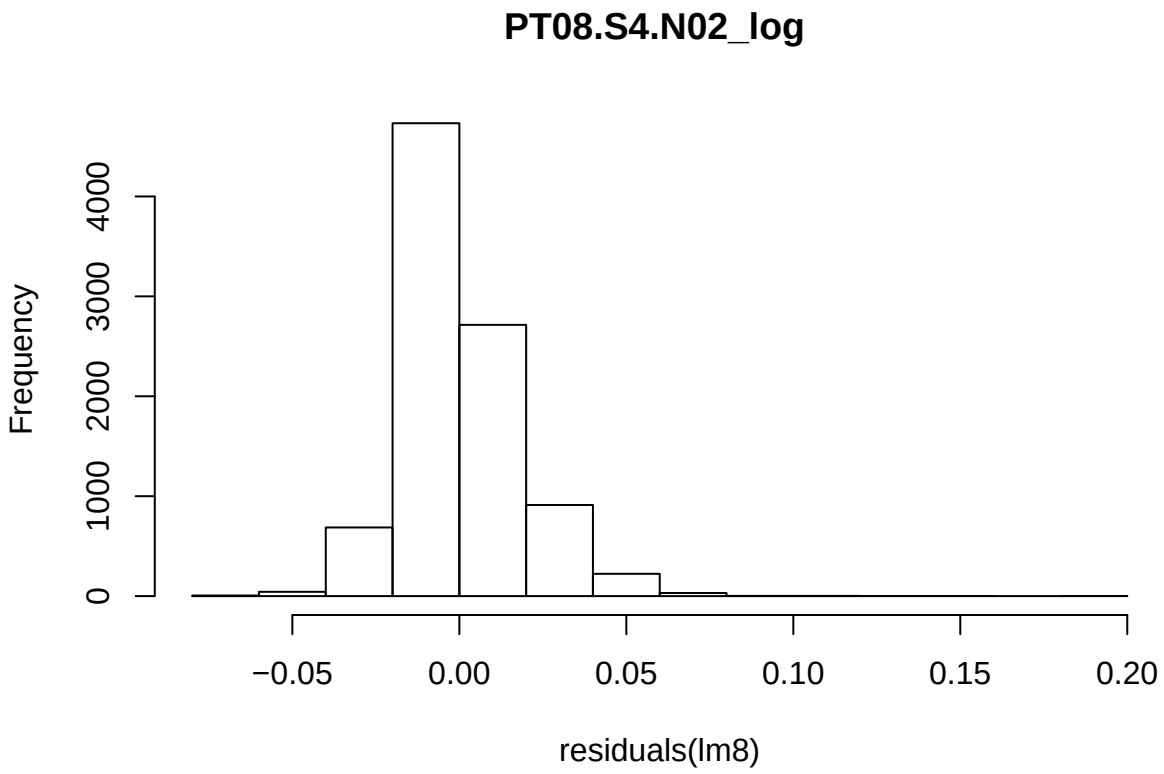
```
vif(lm7)
```

```
## air_norm$PT08.S4.N02_x2   air_norm$PT08.S4.NO2.
```

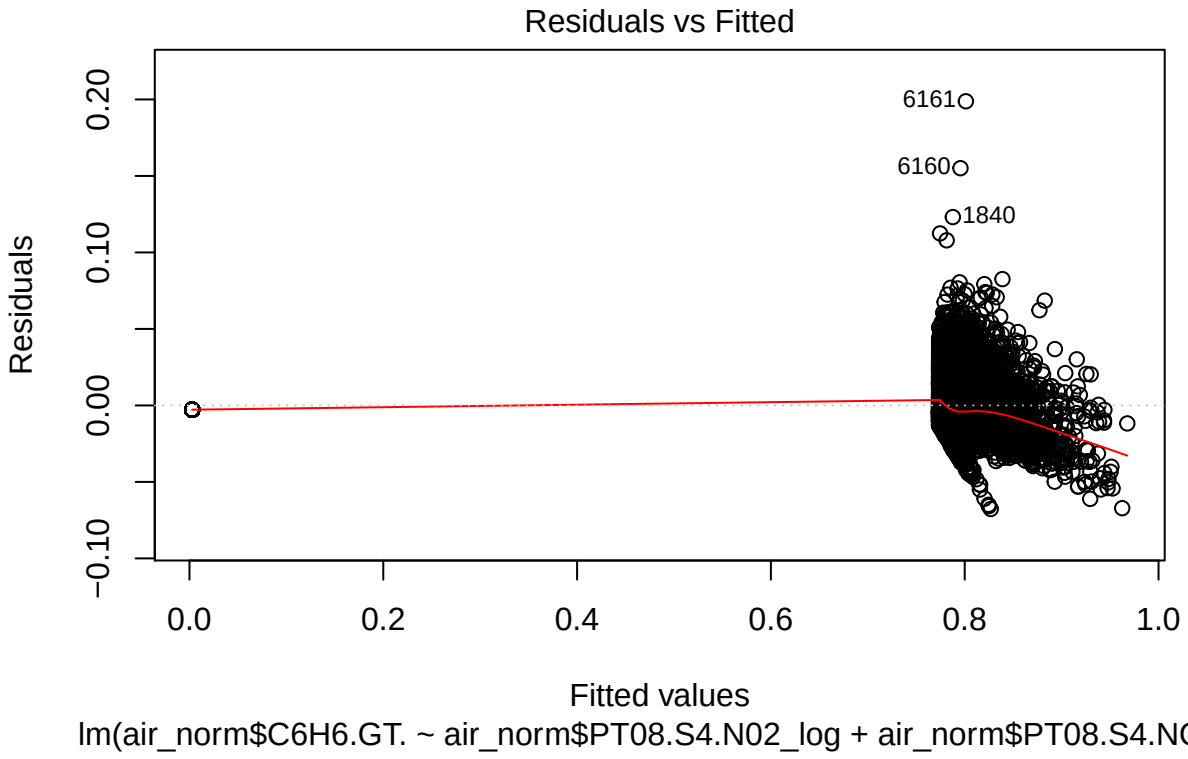
```
##                 9.343466                 9.343466
```

PT08.S1.CO log(x)

```
air_norm$PT08.S4.NO2_log <- log(air_norm$PT08.S4.NO2.)
air_norm$PT08.S4.NO2_log[!is.finite(air_norm$PT08.S4.NO2_log)] <- 0
lm8 <- lm(air_norm$C6H6.GT. ~ air_norm$PT08.S4.NO2_log + air_norm$PT08.S4.NO2.)
hist(residuals(lm8),main ="PT08.S4.NO2_log ")
```




```
plot(lm8,which=1)
```



Chieck vif

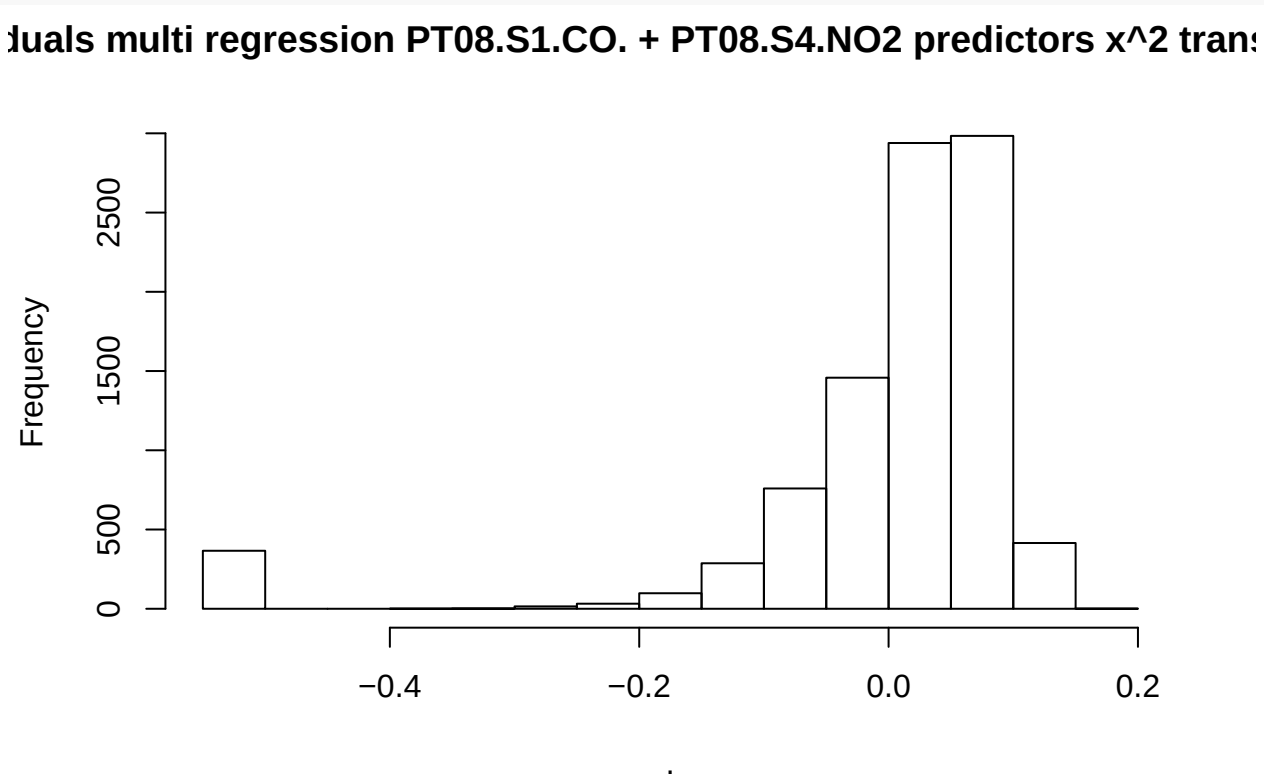
```
vif(lm8)
```

```
## air_norm$PT08.S4.N02_log    air_norm$PT08.S4.NO2.
##                 1.094169                 1.094169
```

```
lm9 <- lm(air_norm$C6H6.GT. ~ air_norm$PT08.S1.CO_x2 + air_norm$PT08.S4.N02_x2)
summary(lm9)
```

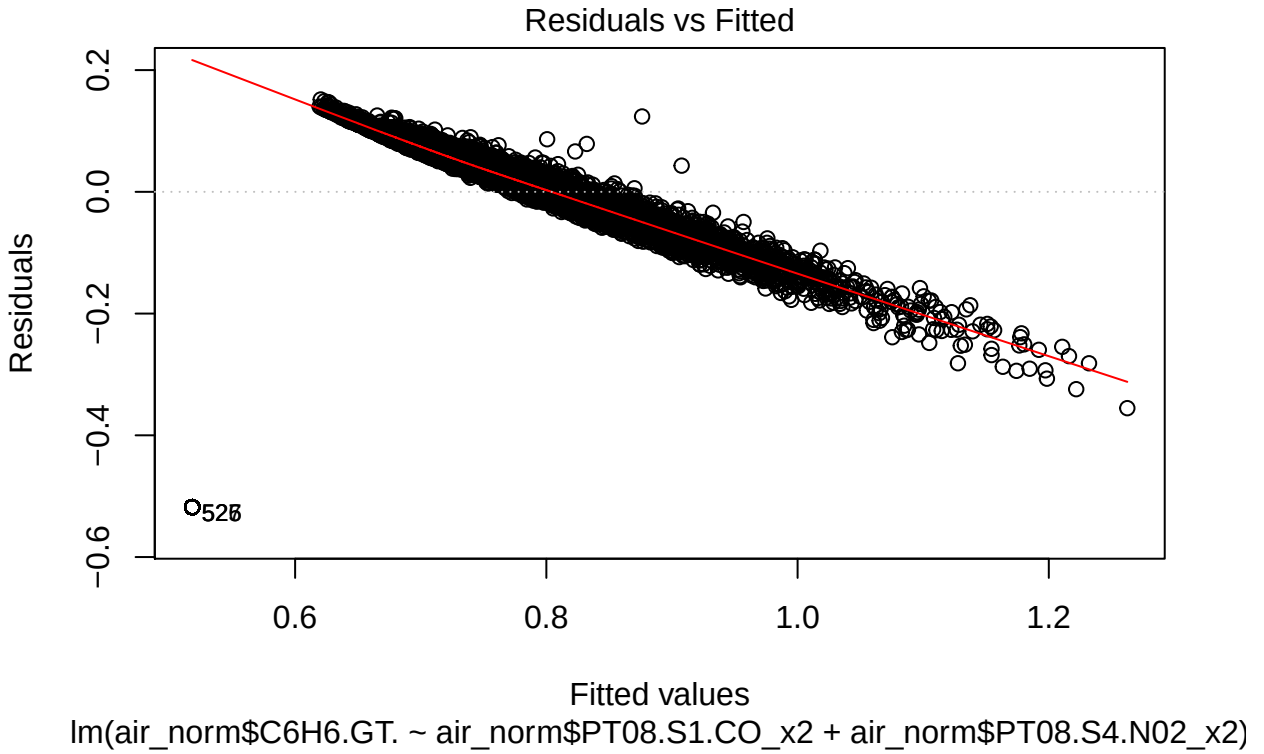
```
##
## Call:
## lm(formula = air_norm$C6H6.GT. ~ air_norm$PT08.S1.CO_x2 + air_norm$PT08.S4.N02_x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.51804 -0.01956  0.03051  0.06509  0.15169
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              0.518044   0.003362  154.09   <2e-16 ***
## air_norm$PT08.S1.CO_x2   0.561439   0.014328   39.18   <2e-16 ***
## air_norm$PT08.S4.N02_x2 0.195427   0.013369   14.62   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1213 on 9354 degrees of freedom
## Multiple R-squared:  0.4023, Adjusted R-squared:  0.4022
## F-statistic:  3148 on 2 and 9354 DF,  p-value: < 2.2e-16
```

```
residuals(lm9) %>% hist(main = "residuals multi regression PT08.S1.CO. + PT08.S4.NO2 predictors x^2 tra
```



.

```
plot(lm9, which = 1)
```



vif < 5: Indicates that predictors are not redunant (not providing overlapping data to inform response)

```r
vif(lm9)
```

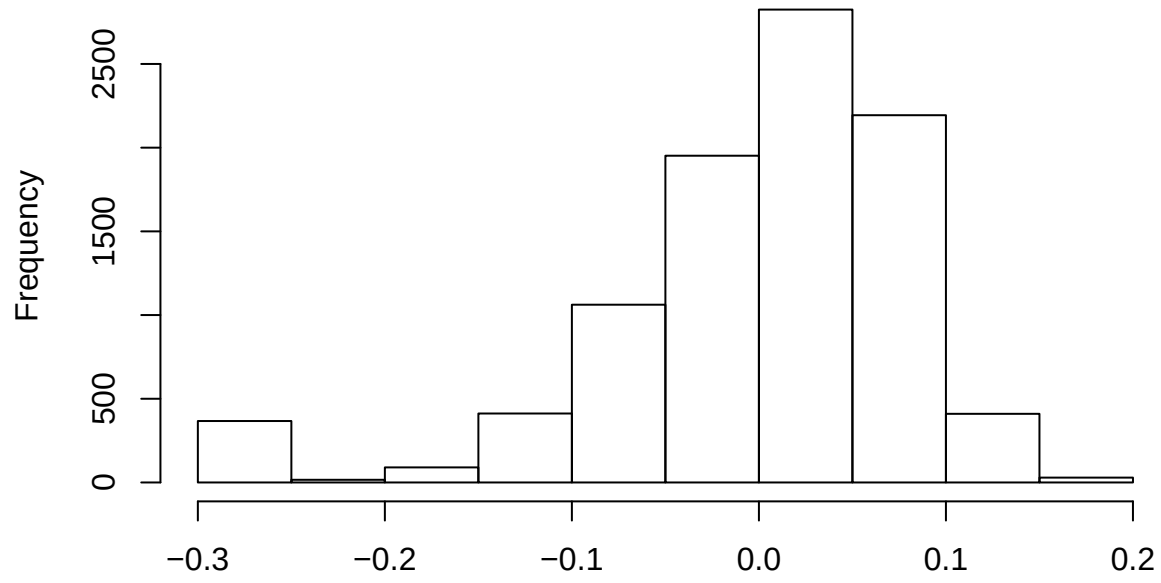
```
##  air_norm$PT08.S1.CO_x2 air_norm$PT08.S4.NO2_x2
##                2.399542                2.399542
```

Bur R^2 better with not-transformed data

```r
lm10 <- lm(air_norm$C6H6.GT. ~ air_norm$PT08.S1.CO. + air_norm$PT08.S4.NO2.)
summary(lm10)
```

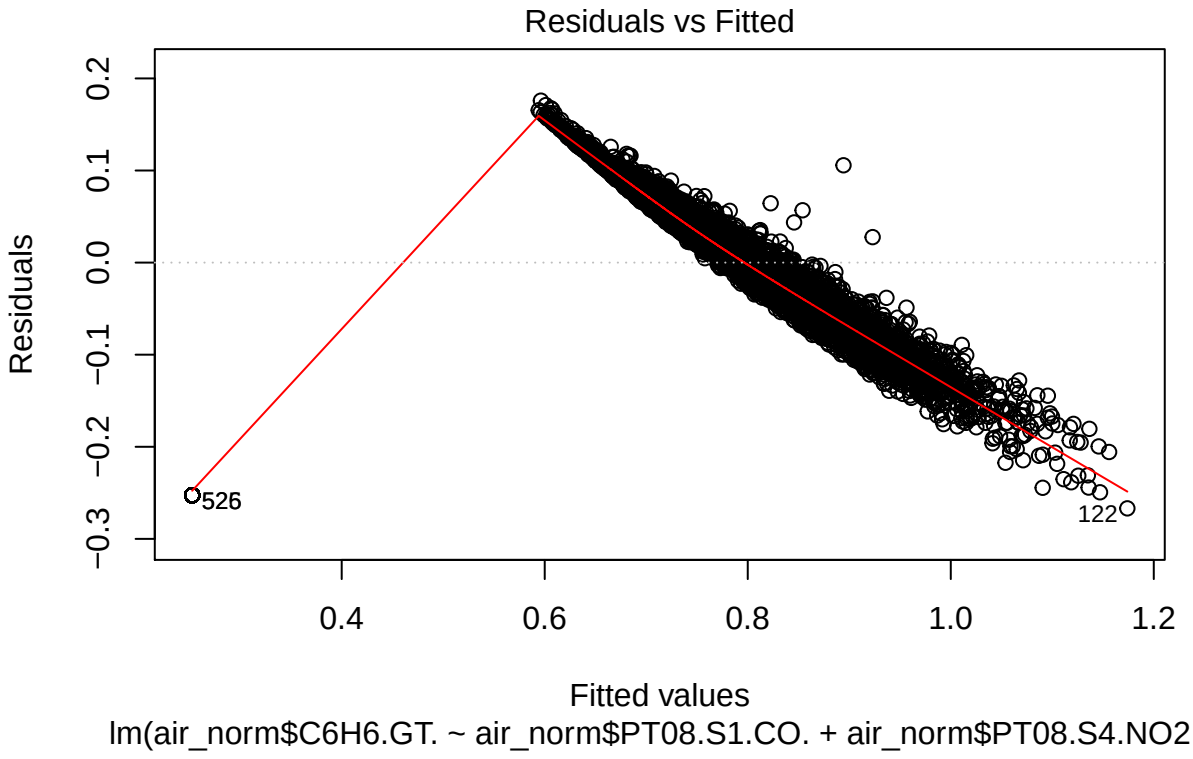
```
##
## Call:
## lm(formula = air_norm$C6H6.GT. ~ air_norm$PT08.S1.CO. + air_norm$PT08.S4.NO2.)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.26687 -0.03750  0.01460  0.05557  0.17584
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           0.252716   0.003273   77.22   <2e-16 ***
## air_norm$PT08.S1.CO.  0.738409   0.010566   69.89   <2e-16 ***
## air_norm$PT08.S4.NO2. 0.188935   0.009907   19.07   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08044 on 9354 degrees of freedom
## Multiple R-squared:  0.7373, Adjusted R-squared:  0.7372
## F-statistic: 1.313e+04 on 2 and 9354 DF,  p-value: < 2.2e-16
```

```r
residuals(lm10) %>% hist(main = "residuals multi regression PT08.S1.CO. + PT08.S4.NO2 predictors not tra
```



.

```r
plot(lm10, which = 1)
```



We have good model with few number of outliers, high R^2 and low vif