Lab 9 Multiple Linear Regression (Training, Validation)

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

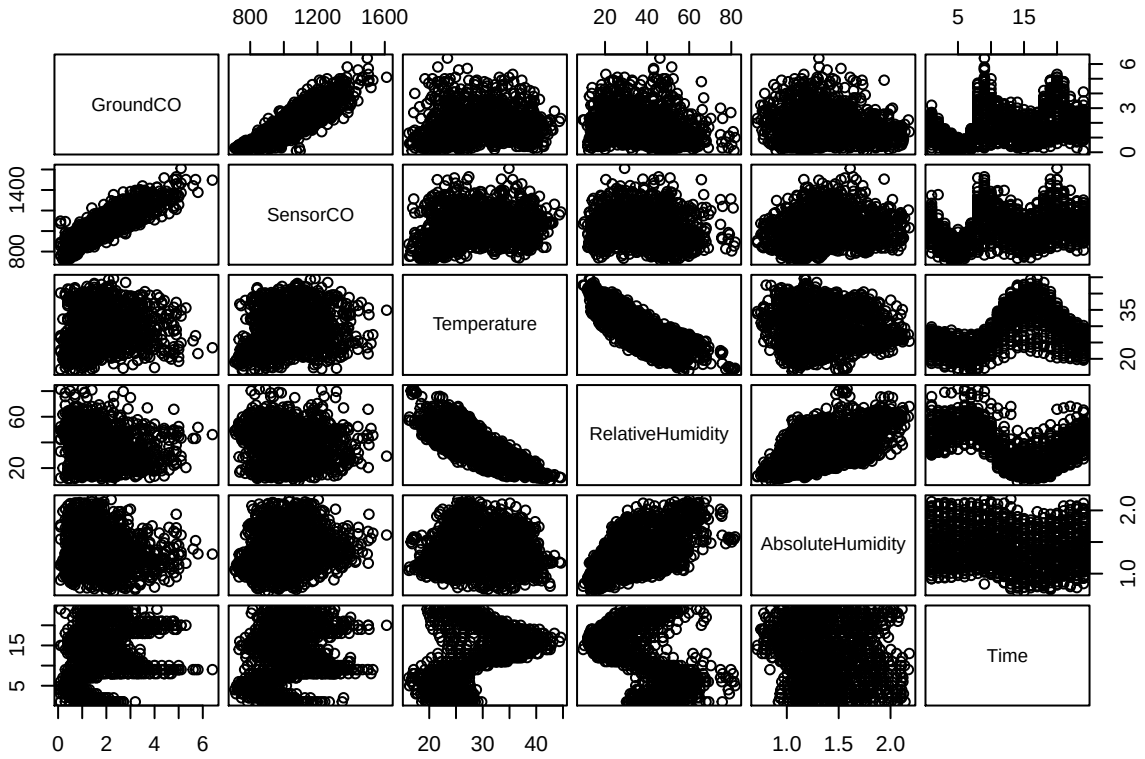
```
# Run initialization R file
setwd("C:/Users/hugo1/Documents/ma575/Proj")
rm(list = ls())
source("PreProcessingLab9.R")
library(corrplot) # for correlation plot
```

```
## corrplot 0.84 loaded
```

```
# Extract summer data only
DataSetSummer = Dataset[(Dataset$Date >= "2004-06-1" & Dataset$Date <= "2004-8-31") | (Dataset$Date >= "

# Randomize rows
DataSetSummer = DataSetSummer[sample(nrow(DataSetSummer)),]

# Form Training, Validation and Testing sets
DataSetSummerTraining = DataSetSummer[1:796,]; # 50% for the data
DataSetSummerValidation = DataSetSummer[796:1194,]; # 25% for the data
DataSetSummerTesting = DataSetSummer[1194:1593,]; # 25% for the data

# Perform training
attach(DataSetSummerTraining)

# Plot scatter matrix
library(car)
```

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

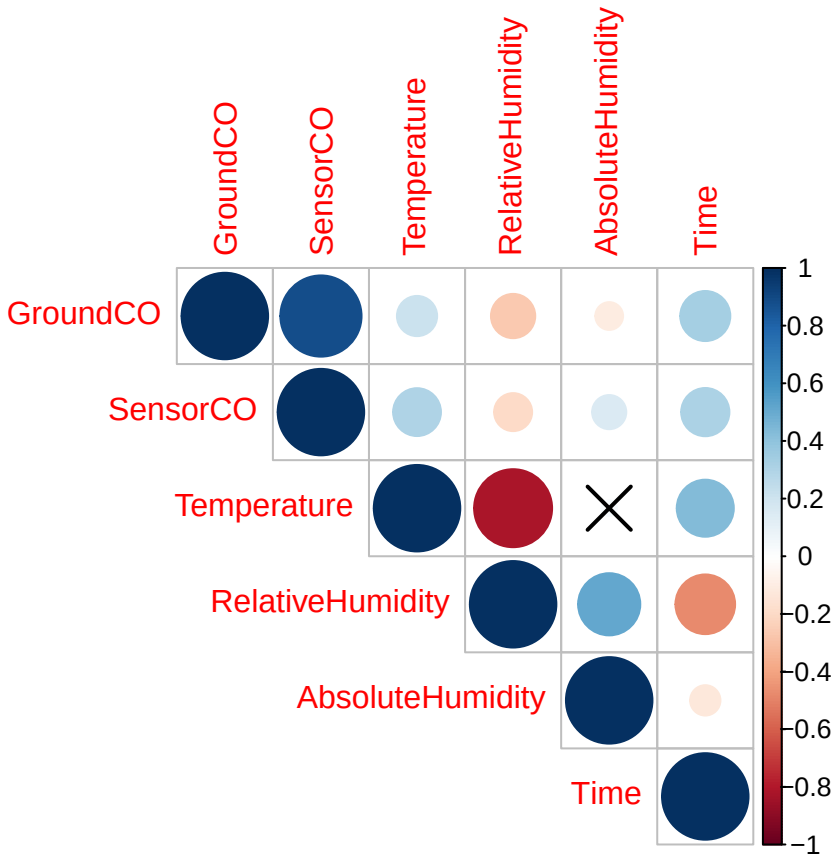
```
pairs(~GroundCO+SensorCO+Temperature+RelativeHumidity+AbsoluteHumidity+Time,
      data=DataSetSummer,gap=0.4)
```




```
# Correlation matrix
X <- cbind(GroundCO,SensorCO,Temperature,RelativeHumidity,AbsoluteHumidity,Time)
c <- cor(X)
round(c,3)
```

```
##                  GroundCO SensorCO Temperature RelativeHumidity
## GroundCO            1.000    0.881       0.216           -0.262
## SensorCO            0.881    1.000       0.306           -0.193
## Temperature         0.216    0.306       1.000           -0.813
## RelativeHumidity   -0.262   -0.193      -0.813            1.000
## AbsoluteHumidity   -0.103    0.156       0.032            0.517
## Time                0.335    0.311       0.435           -0.476
##                  AbsoluteHumidity   Time
## GroundCO                   -0.103  0.335
## SensorCO                    0.156  0.311
## Temperature                 0.032  0.435
## RelativeHumidity            0.517 -0.476
## AbsoluteHumidity            1.000 -0.123
## Time                       -0.123  1.000
```

```
res1 <- cor.mtest(X, conf.level = .95)
## corrplot 0.84 loaded
corrplot(round(c,3),p.mat = res1$p, sig.level = .05, type = "upper")
```



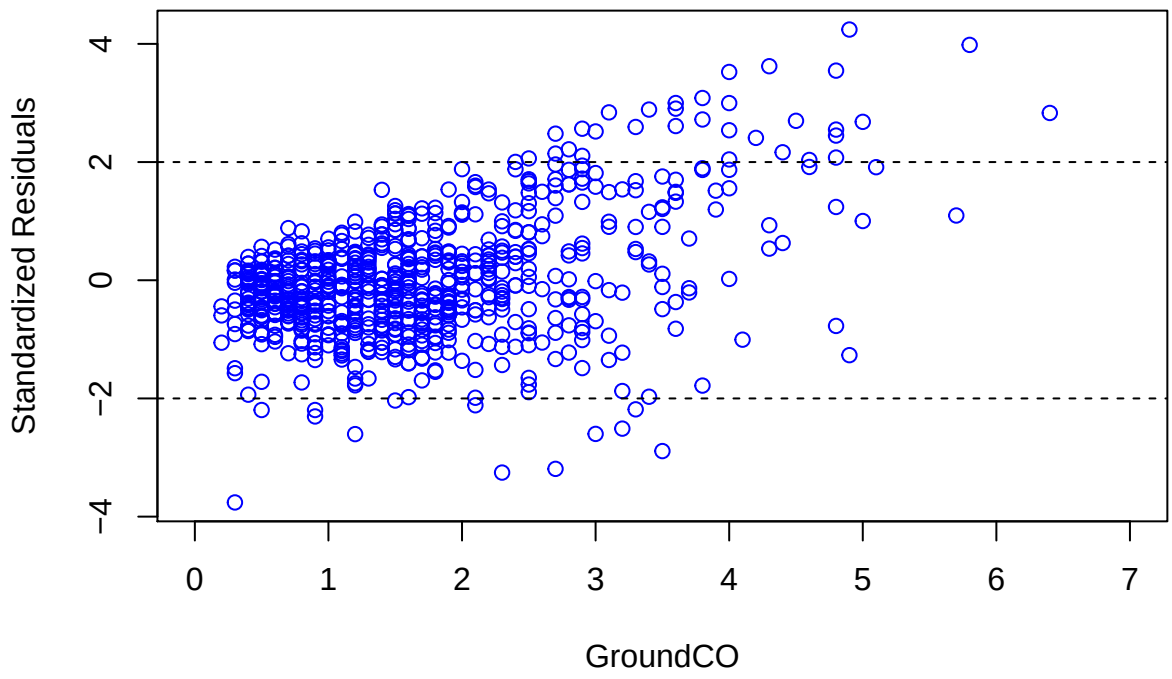

```
# temperature and relatively humidity are highly correlated (negatively)
# absolute humidity and relatively humidity are highly correlated (positively)
# ground CO and sensor CO are highly correlated (positively)

# QUESTION1 - keep ground CO, shouldn't we remove sensor CO? ####
# keep relative humidity, remove temperature, and absolute humidity, then keep time
```

```
# Perform Multiple Linear Regression between GroundCO vs Temperature + SensorCO + RelativeHumidity
# + AbsoluteHumidity + Time
# QUESTION2 - why is there a square term? (SensorCO^2) ####
m.mls <- lm(GroundCO ~ SensorCO + I(SensorCO^2) + RelativeHumidity + Time)

# Examine R output for MLS
summary(m.mls)
```

```
##
## Call:
## lm(formula = GroundCO ~ SensorCO + I(SensorCO^2) + RelativeHumidity +
##     Time)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.72982 -0.25980 -0.04499  0.19728  1.95493
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       1.140e+00  6.298e-01   1.810 0.070644 .
```

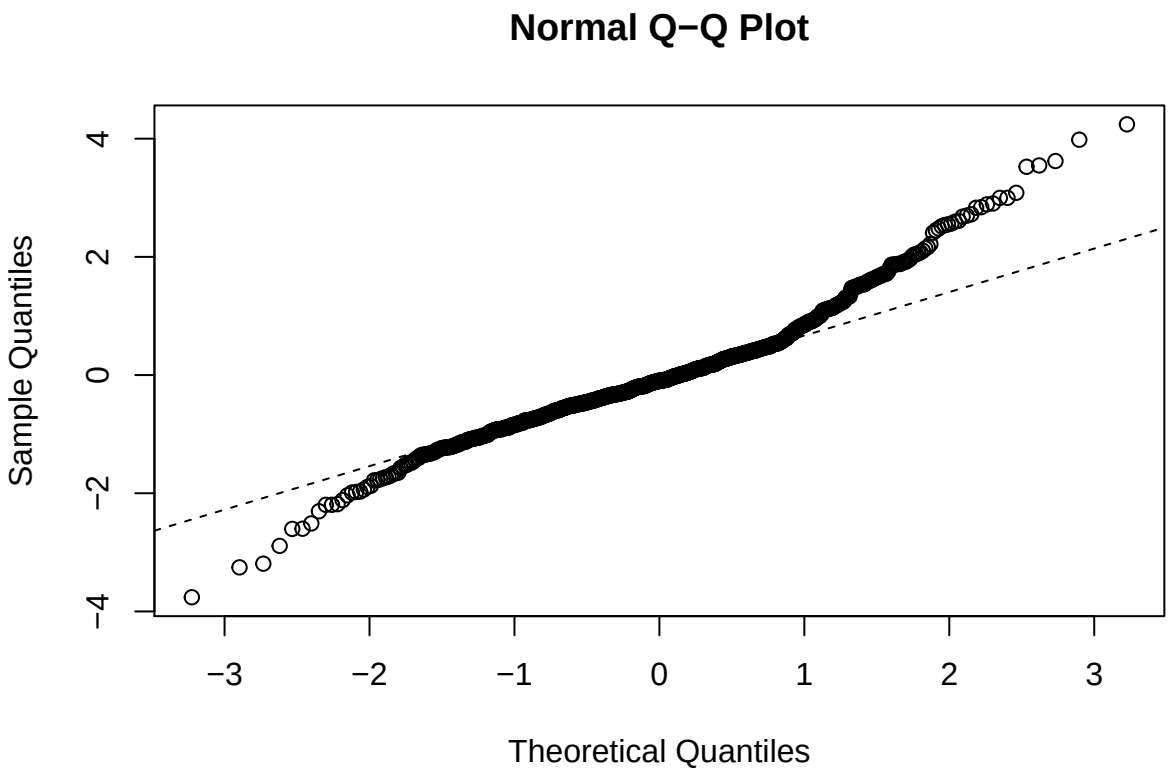
```
## SensorCO         -4.260e-03  1.198e-03  -3.557 0.000397 ***
## I(SensorCO^2)     4.734e-06  5.605e-07   8.446  < 2e-16 ***
## RelativeHumidity -6.917e-03  1.343e-03  -5.150 3.29e-07 ***
## Time              9.374e-03  2.860e-03   3.278 0.001091 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4618 on 791 degrees of freedom
## Multiple R-squared:  0.8026, Adjusted R-squared:  0.8016
## F-statistic: 803.8 on 4 and 791 DF,  p-value: < 2.2e-16
```

```
# Diagnostics ----------------------------------------------------------------------------------------

# Standarized Residuals
StanResMLS <- rstandard(m.mls)
par(mfrow=c(1,1))
plot(GroundCO,StanResMLS,xlab="GroundCO", ylab="Standardized Residuals",xlim=c(0,7), col="blue") +ablin
```

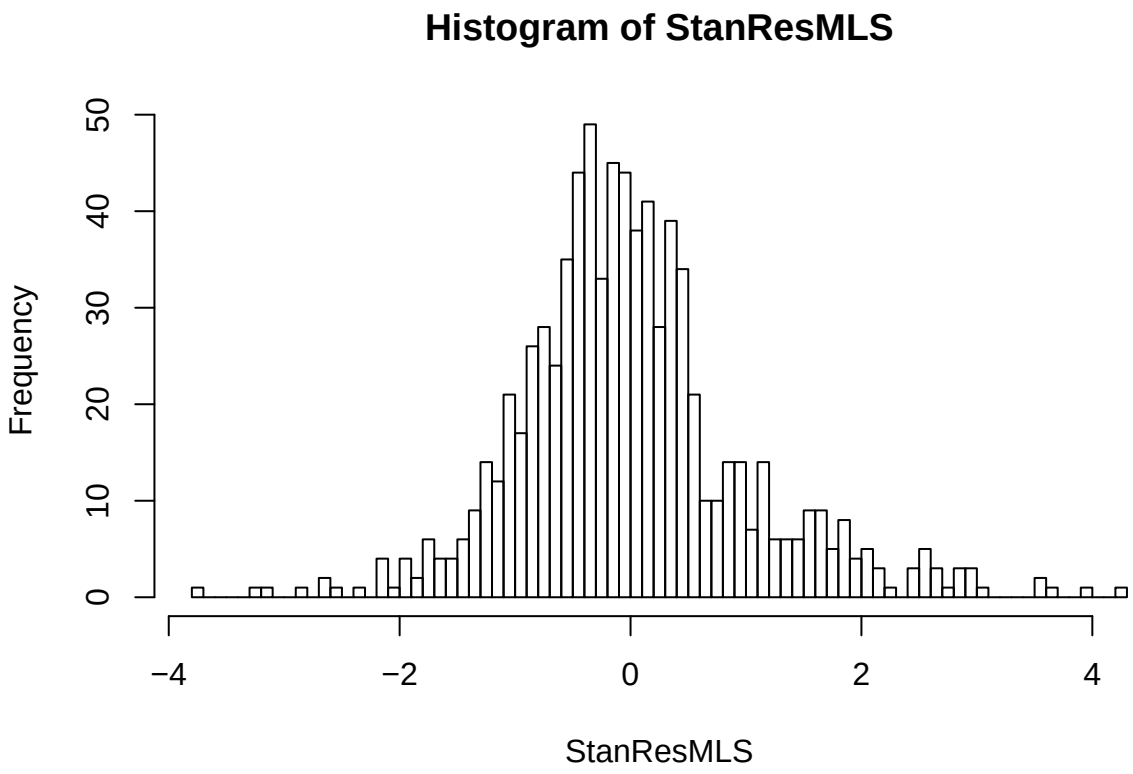



```
## integer(0)
```

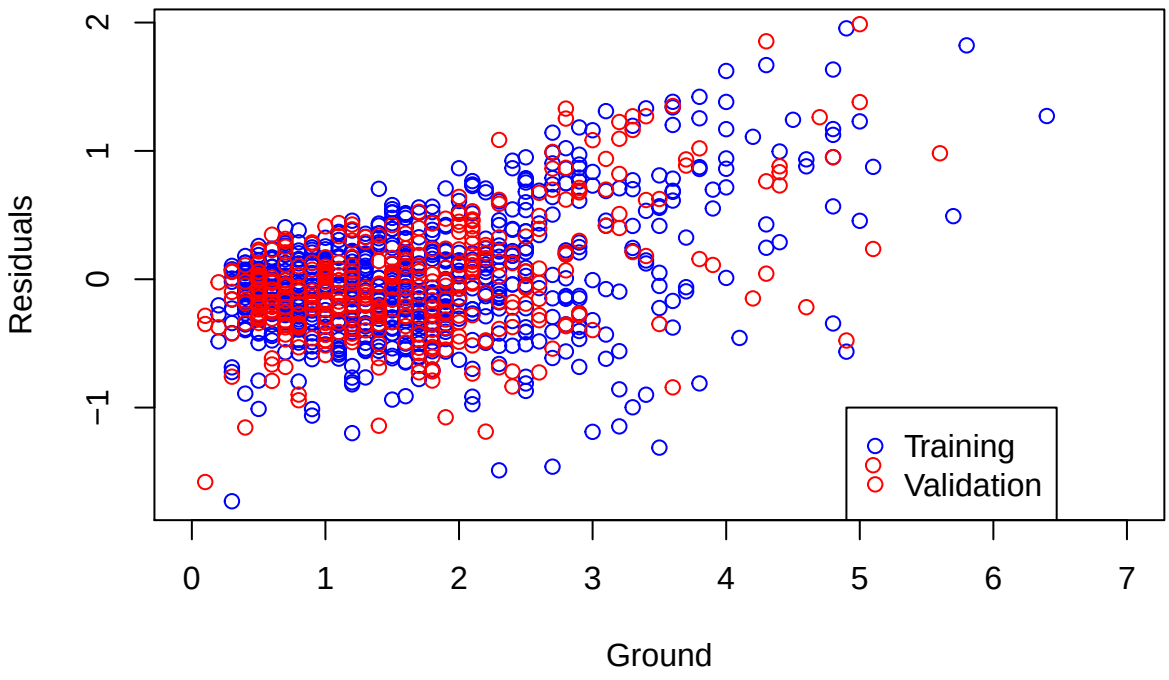
```
# COMMENT - the legend part doesn't work for me
#legend(5.5,1.5,legend=c("MLS"), col=c("blue"), lty=0, cex=1, pch=1)

# Test of Normality for Standarized Residuals of QMLS and QuartLS
q1 <- qqnorm(StanResMLS, plot.it = TRUE)
# This doesn't work me either
qqline(StanResMLS,lty = 2)
```




```
# Histogram of QMLS and QuartLS
par(mfrow=c(1,1))
hist(StanResMLS,100)
```




```
# Validation ------------------------------------------------------------------------------------

# Residuals for training data
ResMLS <- resid(m.mls)
par(mfrow=c(1,1))
plot(GroundCO,ResMLS,xlab="Ground", ylab="Residuals",xlim=c(0,7), col="blue")

# Residuals for Validation data
output<-predict(m.mls, se.fit = TRUE, newdata=data.frame(SensorCO=DataSetSummerValidation$SensorCO, Rela
ResMLSValidation <- DataSetSummerValidation$GroundCO - output$fit
points(DataSetSummerValidation$GroundCO,ResMLSValidation,xlab="GroundCO", ylab="Residuals",xlim=c(0,7),
legend(4.9, -1, legend=c("Training","Validation"), col=c("blue","red"), lty=0, cex=1, pch=1)
```




```
# Mean Square Error for training data
mean((ResMLS)^2)
```

```
## [1] 0.2118816
```

```
# Mean Square Error for validation data
mean((ResMLSValidation)^2)
```

```
## [1] 0.2194081
```

```
detach(DataSetSummerTraining)
```