Model Short Report

October 20, 2020

Data Cleaning

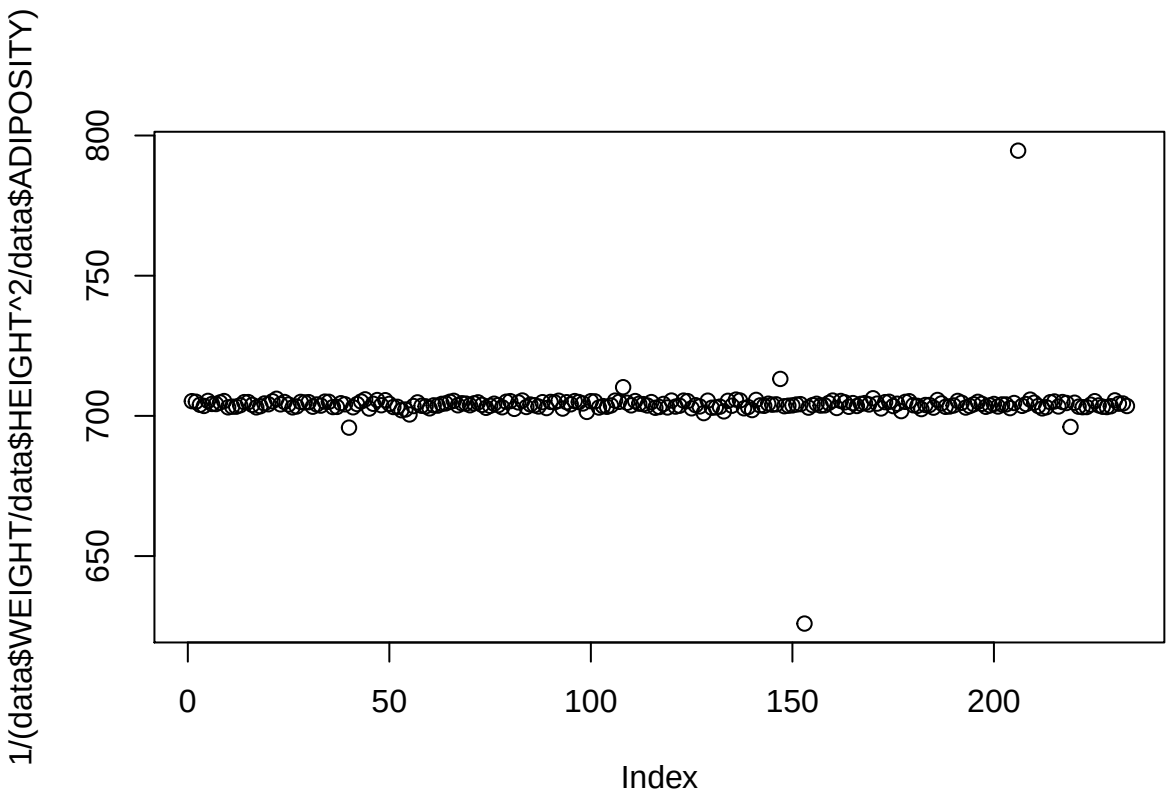
We first clean the data by boxplot.

```
data = read.csv("BodyFat.csv")
index = c()
for(i in 1:length(data))
  {
  index = c(index,which(is.element(data[,i], boxplot(data[,i],plot=FALSE)$out)==1))
  }
index = sort(unique(index))

data = data[-index,]
```

Now, we clean the data by the relaionship among height weight and Adiposity.

```
plot(1/(data$WEIGHT/data$HEIGHT^2/data$ADIPOSITY))
```



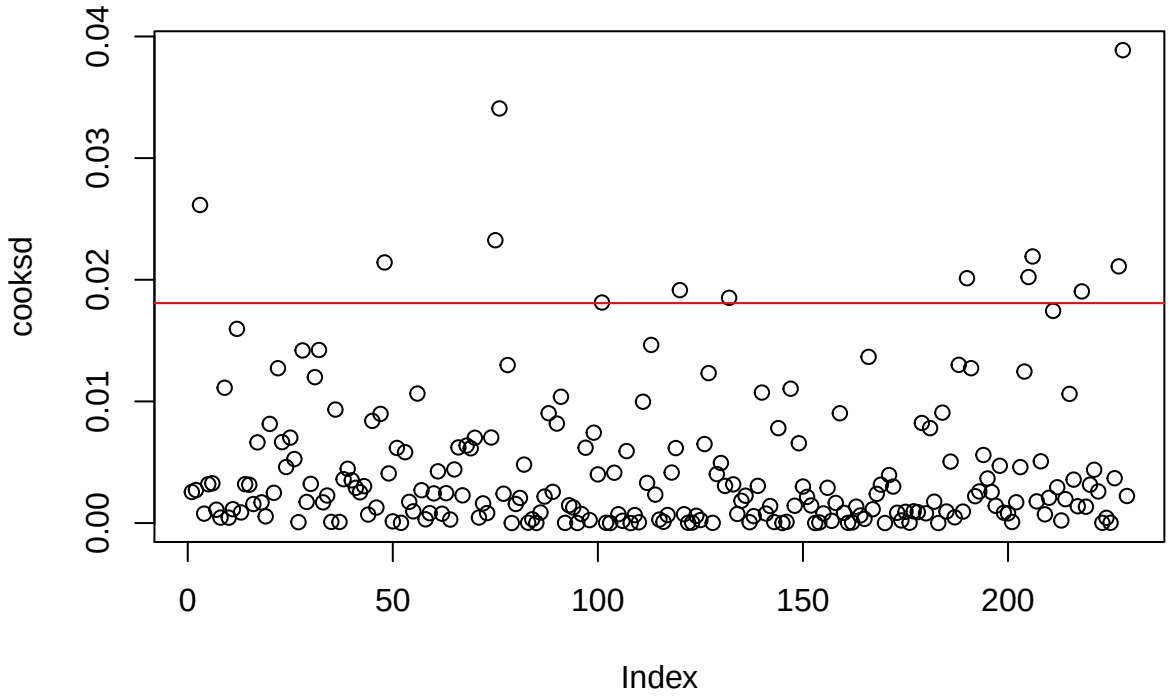

```
index = which(abs(1/(data$WEIGHT/data$HEIGHT^2/data$ADIPOSITY)-700)>50)
data = data[-index,]
```

Then, we clean the data manually.

```
selected_WEIGHT = data[,c('WEIGHT')] <300
selected_BODYFAT = data[,c('BODYFAT')]> 2 & data[,c('BODYFAT')]< 45
selected_HEIGHT = data[,c('HEIGHT')] > 30
selected_item = which(selected_WEIGHT&selected_HEIGHT&selected_BODYFAT)
data = data[selected_item,]
```

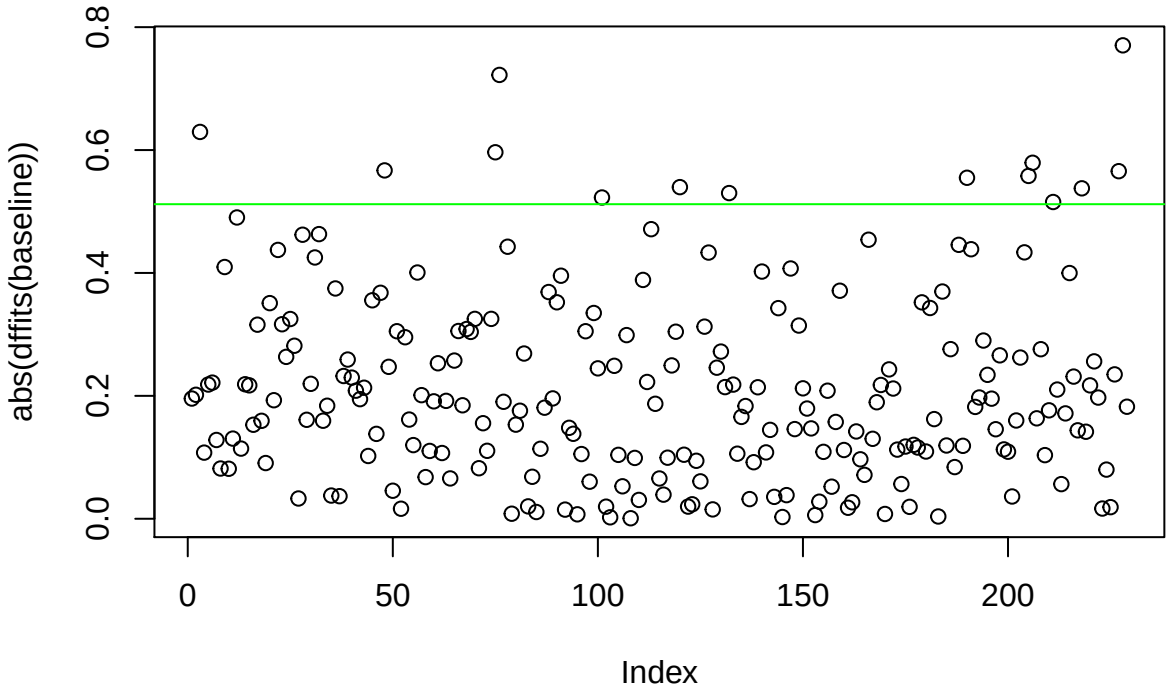
Now, we clean the data by detecting the influtial points of the baseline model(linear model lm(BODYFAT ~.)).

```
data = data[,-c(1,3)]
baseline = lm(BODYFAT~.,data = data)
cooksd = cooks.distance(baseline)
plot(cooksd)
abline(h = 4*mean(cooksd, na.rm=T), col="red")
```




```
index = which(cooksd>=4*mean(cooksd, na.rm=T))
#index = which(cooksd>=0.03)

n=dim(model.matrix(baseline))[1]
p=dim(model.matrix(baseline))[2]
plot(abs(dffits(baseline)))
abline(h=1,col='red')
abline(h=2*sqrt(p/n),col='green')
```




```
index = c(index,which(abs(dffits(baseline))>1))
index = sort(unique(index))
data = data[-index,]
```

Modeling

```
#SLRmodel = lm(BODYFAT~AGE+WEIGHT+HEIGHT+ADIPOSITY+NECK+CHEST+ABDOMEN+HIP+THIGH+KNEE+ANKLE+BICEPS+FOREA

model0 = lm(BODYFAT~1,data = data)
model2 = lm(BODYFAT~.*.*.,data = data)
model1 = lm(BODYFAT~.*.,data = data)

modelAIC = step(model0, scope=list(upper = model1,lower = model0), direction="both",trace = 0)

#modelBIC = step(model0, scope=list(upper = model1,lower = model0), direction="both",trace = 0,k=log(le

summary(modelAIC)
```

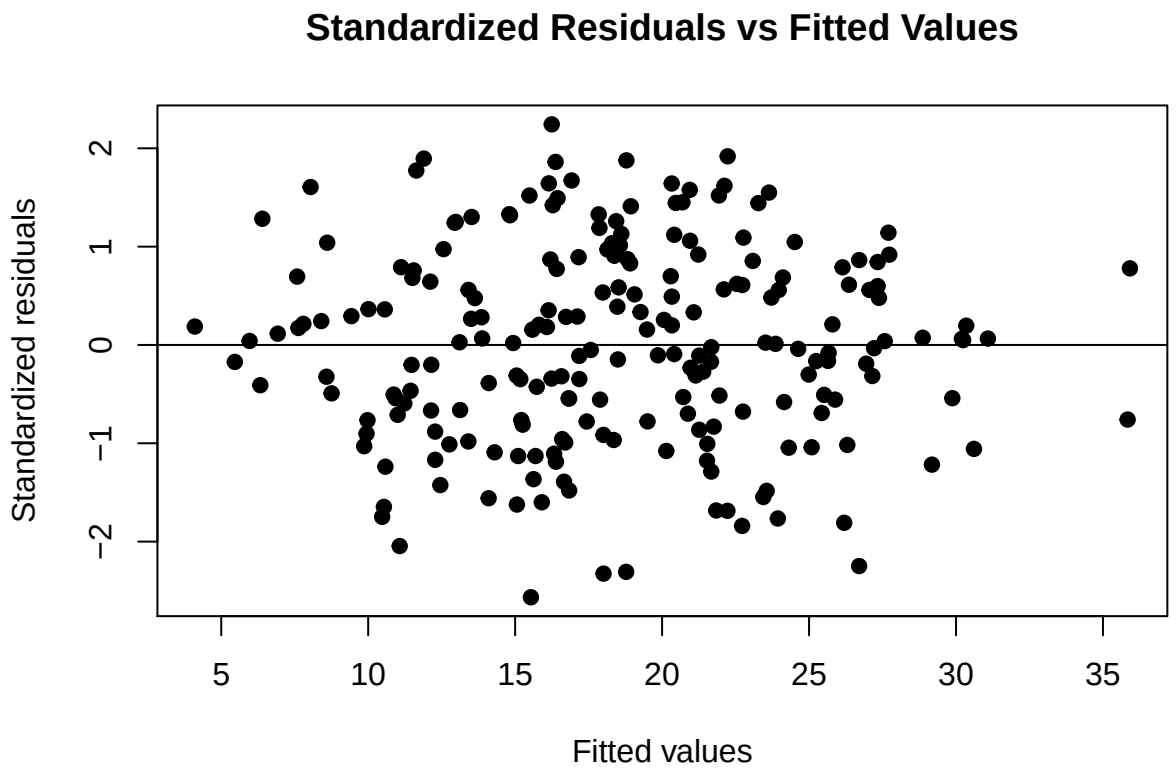
```
##
## Call:
## lm(formula = BODYFAT ~ ABDOMEN + WEIGHT + WRIST + FOREARM + NECK +
##     AGE + THIGH + KNEE + HIP + CHEST + WEIGHT:THIGH + ABDOMEN:NECK +
##     FOREARM:CHEST, data = data)
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -8.7369 -2.3932  0.0852  2.5596  7.5548
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    7.598992  58.555399   0.130 0.896874
## ABDOMEN       -1.355830   0.747080  -1.815 0.071032 .
## WEIGHT         0.369880   0.155162   2.384 0.018060 *
## WRIST         -1.664533   0.497676  -3.345 0.000982 ***
## FOREARM        4.895043   2.980447   1.642 0.102066
## NECK          -6.026065   1.842921  -3.270 0.001265 **
## AGE            0.074661   0.027505   2.714 0.007212 **
## THIGH          1.621559   0.462329   3.507 0.000558 ***
## KNEE          -0.371934   0.228121  -1.630 0.104570
## HIP           -0.198447   0.132480  -1.498 0.135711
## CHEST          1.074049   0.844073   1.272 0.204672
## WEIGHT:THIGH  -0.006896   0.002456  -2.808 0.005480 **
## ABDOMEN:NECK   0.060035   0.019884   3.019 0.002861 **
## FOREARM:CHEST -0.042138   0.029255  -1.440 0.151314
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.446 on 202 degrees of freedom
## Multiple R-squared:  0.7701, Adjusted R-squared:  0.7553
## F-statistic: 52.04 on 13 and 202 DF,  p-value: < 2.2e-16
```

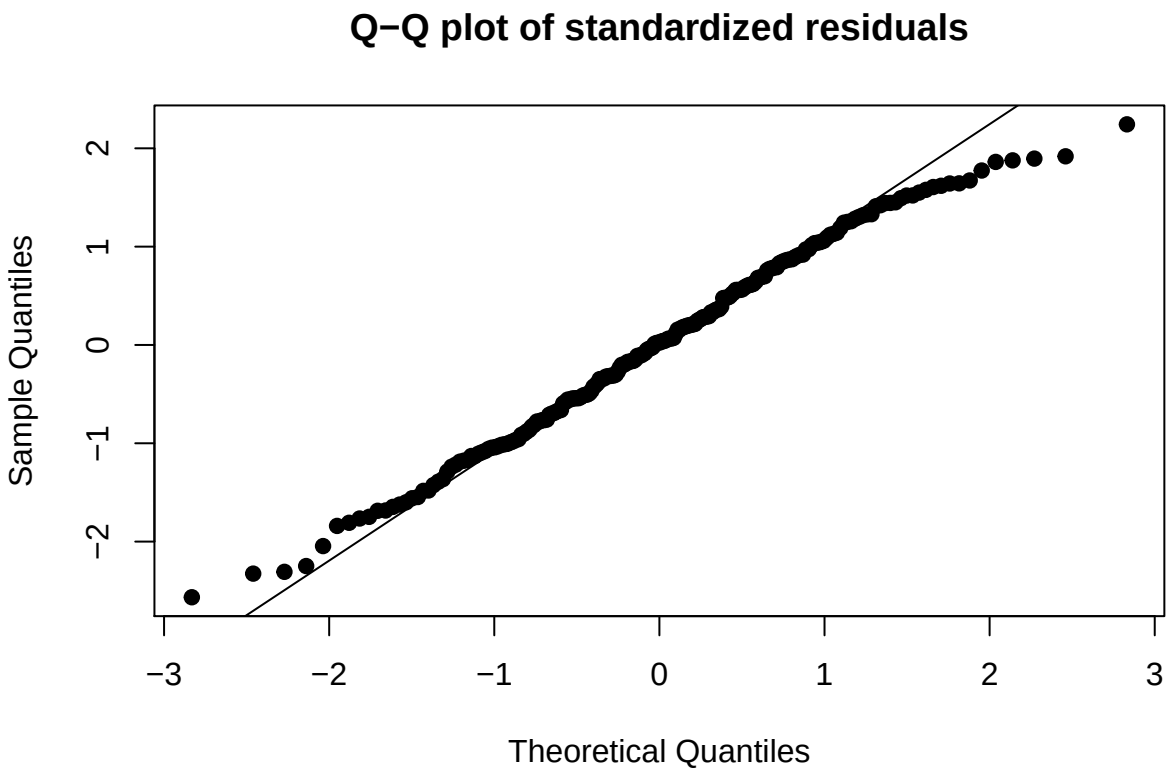
```
#summary(modelBIC)
```

Model Diagnostics

```
plot(predict(modelAIC),rstandard(modelAIC),type="p",pch=19,main = "Standardized Residuals vs Fitted Valu
abline(a=0,b=0)
```

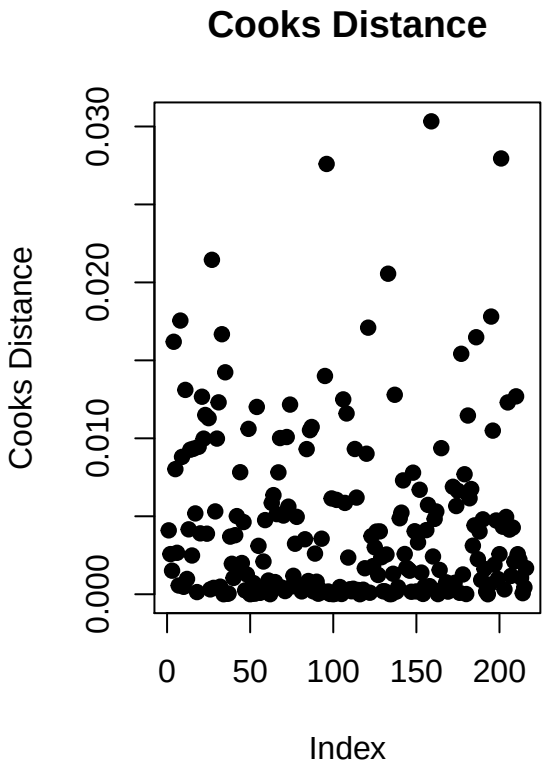


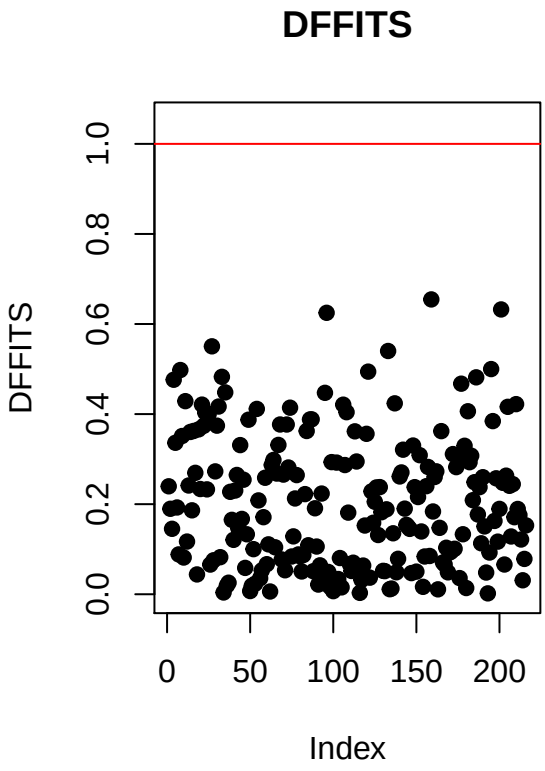

```
qqnorm(rstandard(modelAIC),pch=19,main = "Q-Q plot of standardized residuals")
qqline(rstandard(modelAIC))
```




```
cooksd = cooks.distance(modelAIC)
par(mfrow = c(1,2))
plot(cooksd,type="p",pch=19,ylab="Cooks Distance",main = "Cooks Distance")

n=dim(model.matrix(baseline))[1]
p=dim(model.matrix(baseline))[2]
plot(abs(dffits(modelAIC)),ylab ="DFFITS",type = "p",pch=19,ylim = c(0,1.05),main = "DFFITS")
abline(h=1,col='red')
```


Cooks Distance
Cooks Distance
0.000
0.010
0.020
0.030
0
50
100
150
200
Index


DFFITS
DFFITS
0.0
0.2
0.4
0.6
0.8
1.0
0
50
100
150
200
Index