Project

Xinrui Hu

2023-04-20

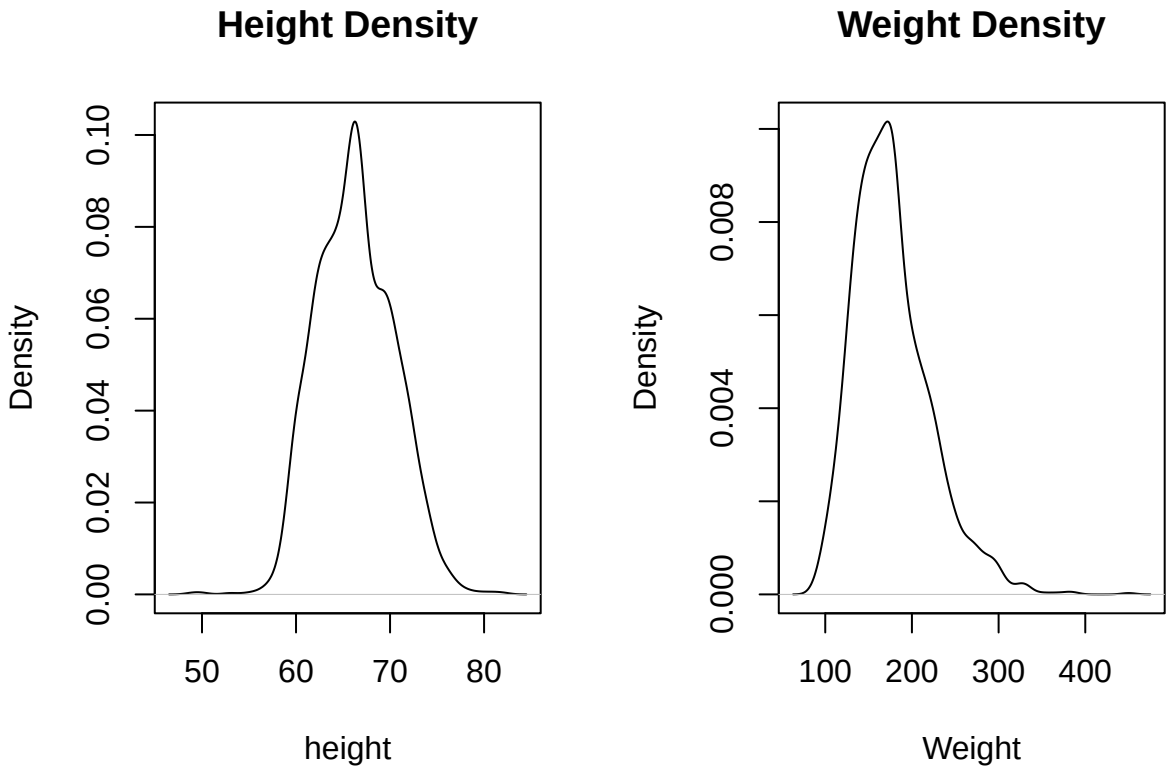
```
library(dplyr)
library(tidyverse)
library(knitr)
library(car)
options(scipen = 999)
df <- read.csv('newdf.csv')
df <- df %>% rename( "age" = "RIDAGEYR",
                     "gender" = "RIAGENDR",
                     "edu" = "DMDEDUC2",
                     "marry" = "DMDMARTL",
                     "race" = "RIDRETH3",
                     "height" = "WHD010",
                     "weight" = "WHD020",
                     "hi" = "HIQ011",
                     "ogtt" = "LBXGLT") %>% na.omit(df$ogtt)
# Deal with the abnormal value
df$height[df$height > 90] <- NA
df$weight[df$weight > 700] <- NA
df$hi[df$hi != 1] <- 2
# put the missing value with the mean of the data
df$weight[is.na(df$weight)] <- mean(df$weight, na.rm = TRUE)
df$height[is.na(df$height)] <- mean(df$height, na.rm = TRUE)
df$bmi <- signif((df$weight / (df$height^2))*703,4)
```

```
summary(df)
```

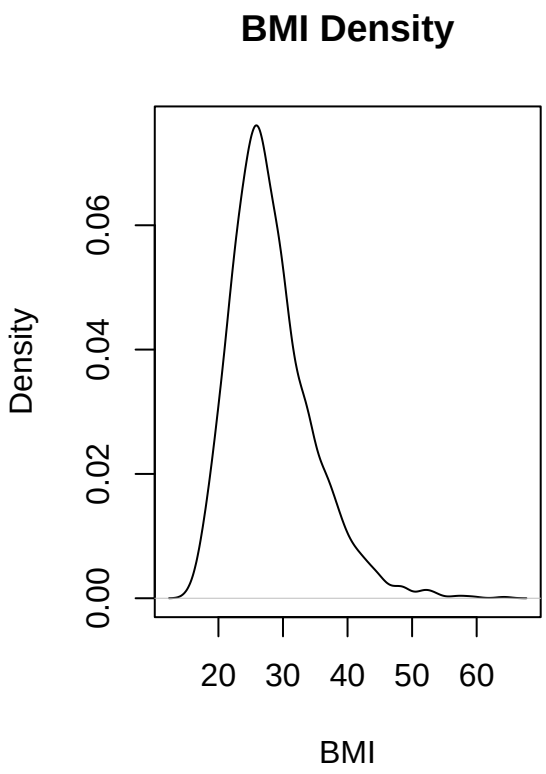
```
##      height          weight          hi             ogtt            age
##  Min.   :49.00   Min.   : 88   Min.   :1.000   Min.   : 35.0   Min.   :20.0
##  1st Qu.:63.00   1st Qu.:145   1st Qu.:1.000   1st Qu.: 92.0   1st Qu.:34.0
##  Median :66.00   Median :170   Median :1.000   Median :111.0   Median :47.5
##  Mean   :66.31   Mean   :177   Mean   :1.206   Mean   :123.6   Mean   :48.3
##  3rd Qu.:69.00   3rd Qu.:200   3rd Qu.:1.000   3rd Qu.:142.0   3rd Qu.:62.0
##  Max.   :82.00   Max.   :450   Max.   :2.000   Max.   :542.0   Max.   :80.0
##      gender           edu            marry            race
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000
##  1st Qu.:1.000   1st Qu.:3.000   1st Qu.:1.000   1st Qu.:2.000
##  Median :2.000   Median :4.000   Median :1.000   Median :3.000
##  Mean   :1.509   Mean   :3.504   Mean   :2.634   Mean   :3.203
##  3rd Qu.:2.000   3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:4.000
##  Max.   :2.000   Max.   :5.000   Max.   :6.000   Max.   :7.000
##       bmi
##  Min.   :15.83
##  1st Qu.:23.80
```

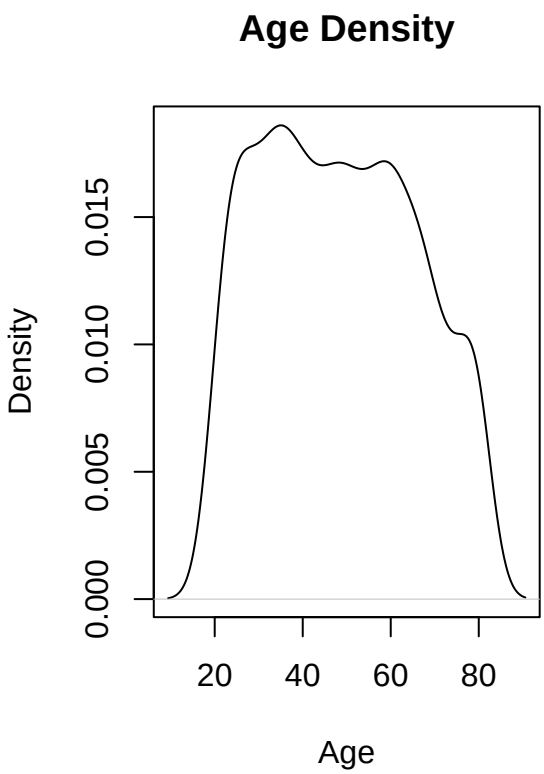
```
##  Median :27.17
##  Mean   :28.22
##  3rd Qu.:31.41
##  Max.   :64.23
```

```r
# Set up the plotting window with two plots side-by-side
par(mfrow = c(1, 2))
plot(density(df$height), main = "Height Density", xlab = "height")
plot(density(df$weight), main = "Weight Density", xlab = "Weight")
```

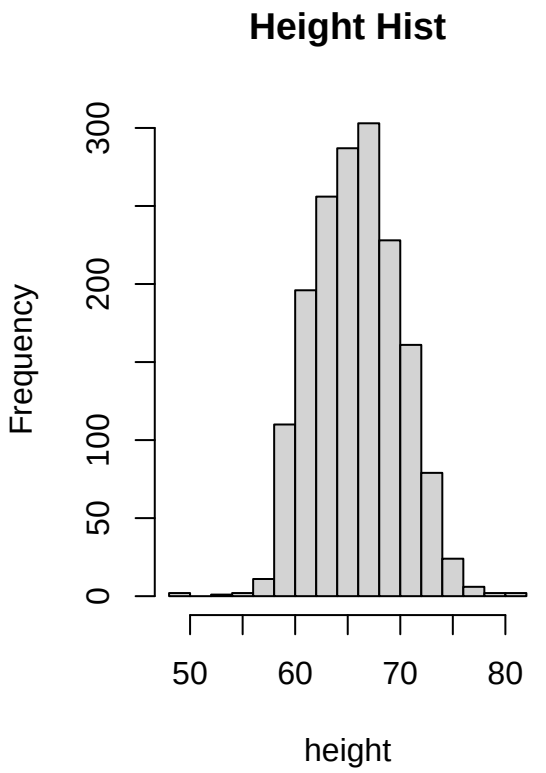


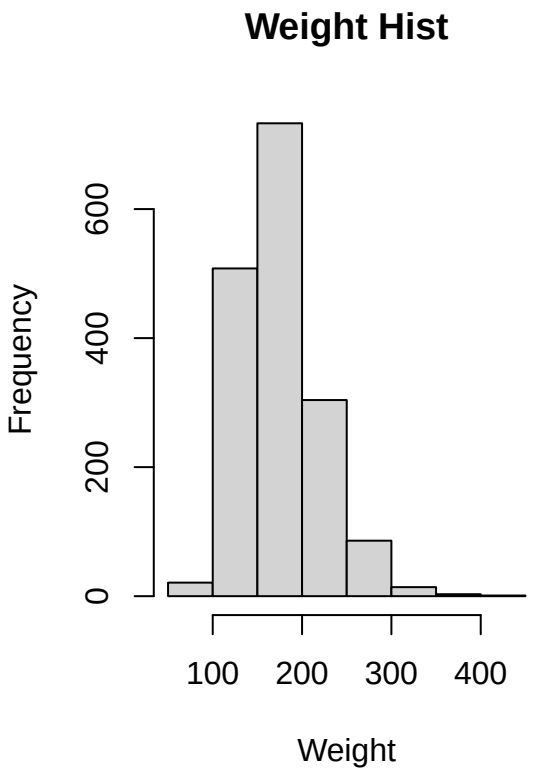

```r
par(mfrow = c(1, 2))
plot(density(df$bmi), main = "BMI Density", xlab = "BMI")
plot(density(df$age), main = "Age Density", xlab = "Age")
```



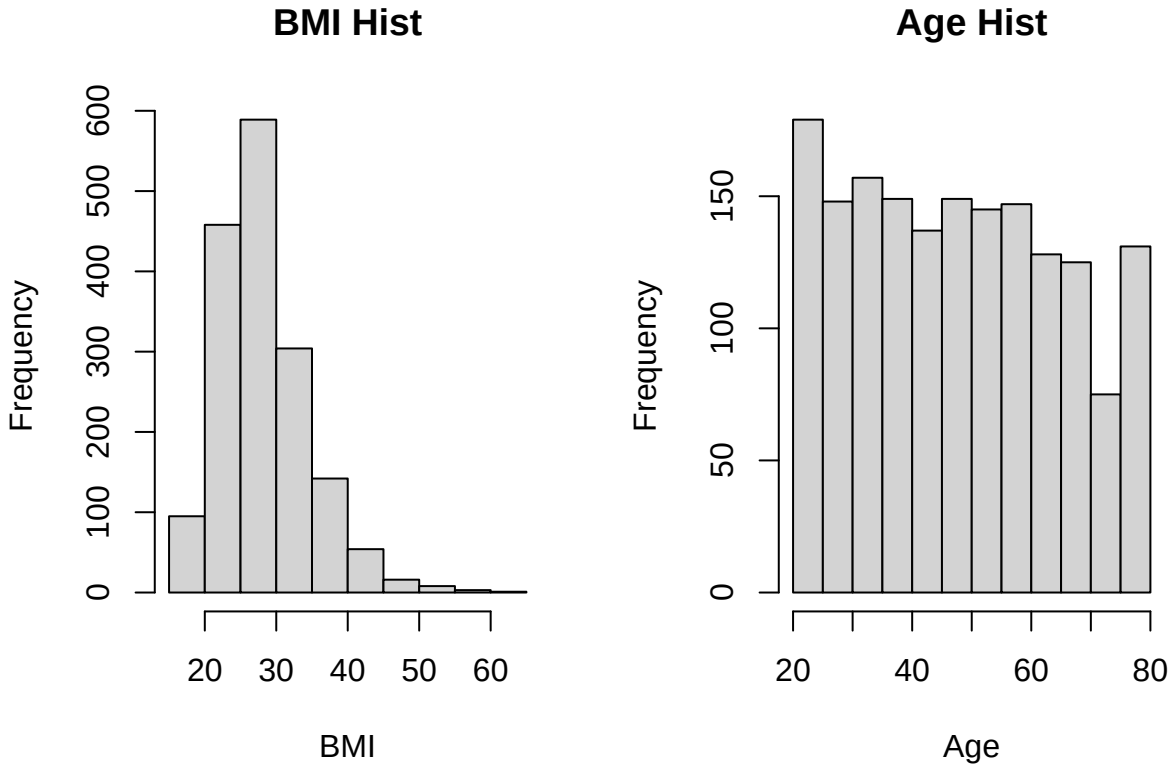



```r
par(mfrow = c(1, 2))
hist(df$height, main = "Height Hist", xlab = "height")
hist(df$weight, main = "Weight Hist", xlab = "Weight")
```

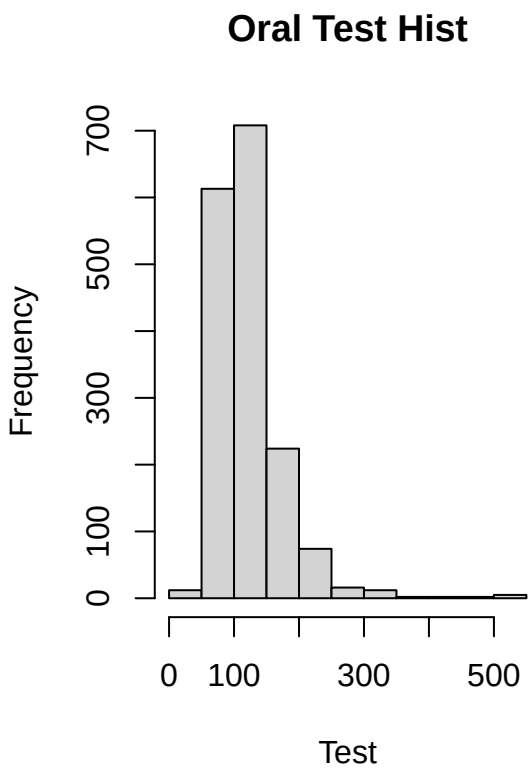


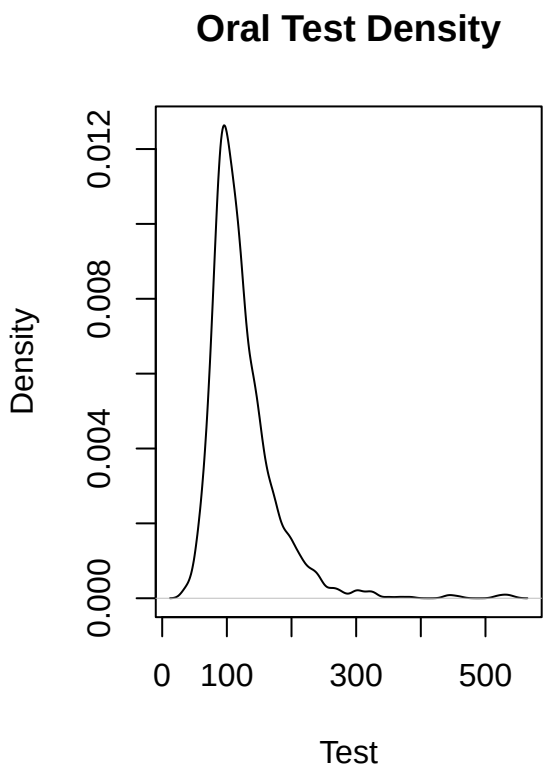


```r
par(mfrow = c(1, 2))
hist(df$bmi, main = "BMI Hist", xlab = "BMI")
hist(df$age, main = "Age Hist", xlab = "Age")
```




```r
par(mfrow = c(1, 2))
hist(df$ogtt, main = "Oral Test Hist", xlab = "Test")
plot(density(df$ogtt), main = "Oral Test Density", xlab = "Test")
```






```
# 10% of data as train to predict 90%
tr_ind <- 1:(nrow(df)/10)
train <- df[tr_ind, ]
test <- df[-tr_ind, ]
model <- lm(ogtt ~ .-weight-height, train)
summary(model)
```

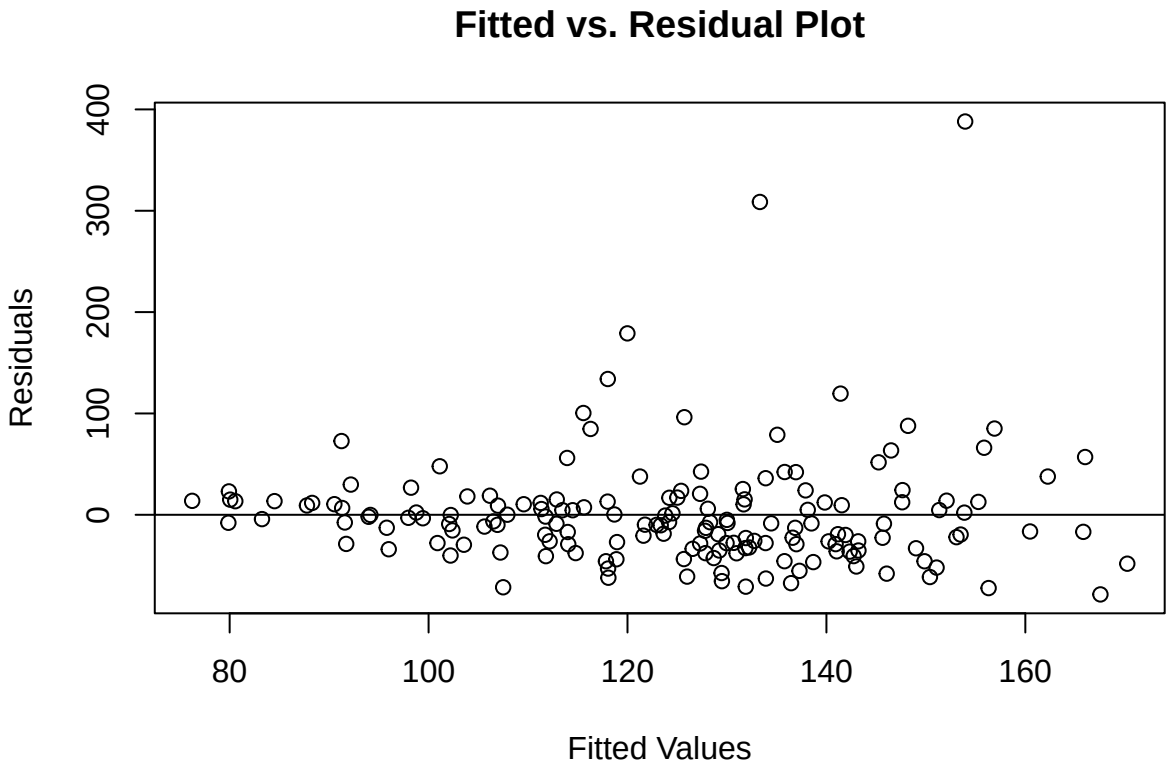
```
##
## Call:
## lm(formula = ogtt ~ . - weight - height, data = train)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -78.55 -29.01  -8.45  13.46 388.05
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  85.4607    42.2468   2.023   0.0448 *
## hi            6.9977    12.2896   0.569   0.5699
## age           0.7765     0.3023   2.569   0.0111 *
## gender        0.6456     9.0570   0.071   0.9433
## edu          -9.2477     3.8997  -2.371   0.0189 *
## marry        -3.4255     2.6254  -1.305   0.1939
## race          2.2359     2.9622   0.755   0.4515
## bmi           0.9177     0.7326   1.253   0.2122
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 57.32 on 159 degrees of freedom
## Multiple R-squared:  0.1193, Adjusted R-squared:  0.08049
```

```
## F-statistic: 3.076 on 7 and 159 DF,  p-value: 0.004577
```

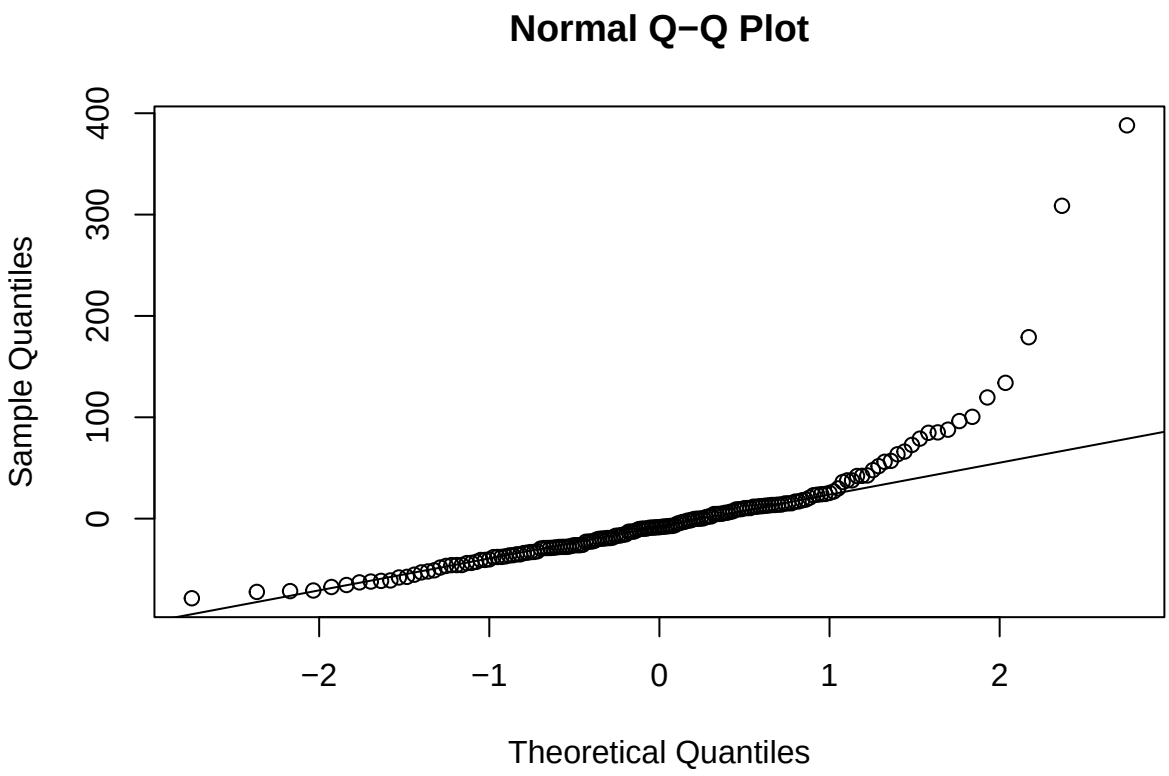
```
#check collinearty
vif(model)
```

```
##       hi      age   gender      edu    marry     race      bmi
## 1.271549 1.381776 1.033806 1.136180 1.263652 1.156472 1.076878
```

```
# Fitted vs. Residuals, Full model
library(ggplot2)
train_pred <- predict(model, newdata = train)
test_pred <- predict(model, newdata = test)
plot(train_pred, residuals(model), xlab = "Fitted Values", ylab = "Residuals", main = "Fitted vs. Residu
abline(h = 0)
```




```
# QQ plot
qqnorm(residuals(model))
qqline(residuals(model))
```




```
# AIC
require(leaps)
```

```
## Loading required package: leaps
```

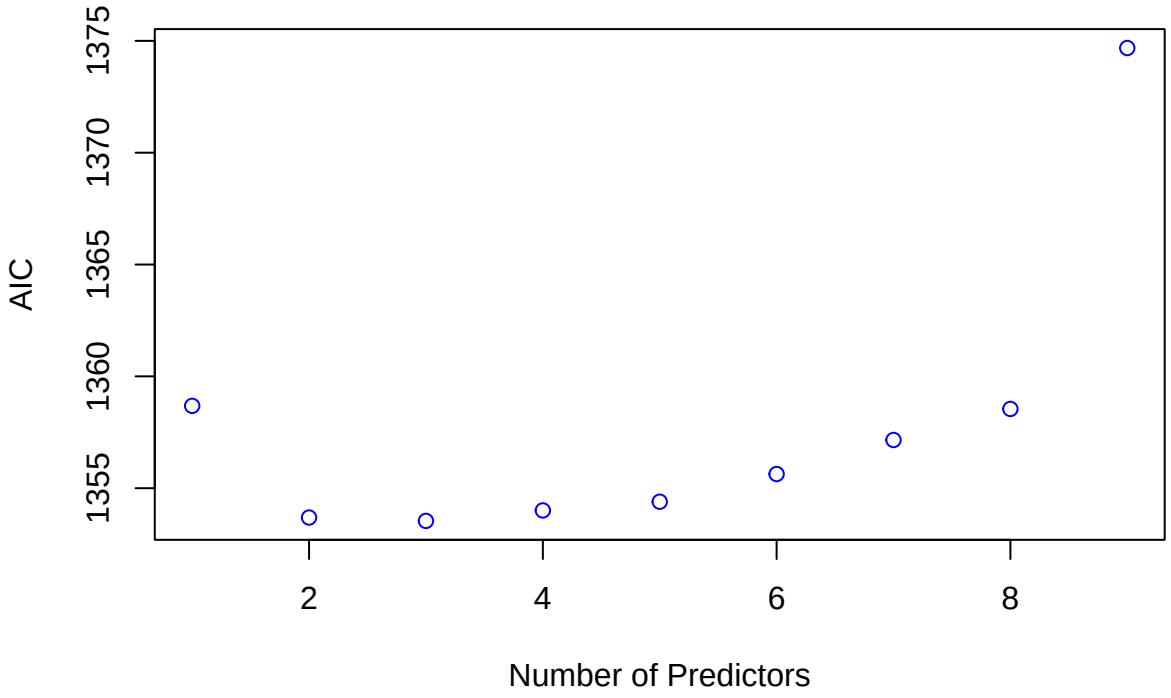
```
amod <- regsubsets(ogtt ~ ., train)
rs <- summary(amod)
rs$which
```

```
##   (Intercept) height weight    hi  age gender   edu marry  race   bmi
## 1        TRUE  FALSE  FALSE FALSE TRUE  FALSE FALSE FALSE FALSE FALSE
## 2        TRUE  FALSE  FALSE FALSE TRUE  FALSE  TRUE FALSE FALSE FALSE
## 3        TRUE  FALSE   TRUE FALSE TRUE  FALSE  TRUE FALSE FALSE FALSE
## 4        TRUE  FALSE   TRUE FALSE TRUE  FALSE  TRUE  TRUE FALSE FALSE
## 5        TRUE   TRUE  FALSE FALSE TRUE   TRUE  TRUE  TRUE FALSE FALSE
## 6        TRUE   TRUE  FALSE FALSE TRUE   TRUE  TRUE  TRUE FALSE  TRUE
## 7        TRUE   TRUE  FALSE  TRUE TRUE   TRUE  TRUE  TRUE FALSE  TRUE
## 8        TRUE   TRUE  FALSE  TRUE TRUE   TRUE  TRUE  TRUE  TRUE  TRUE
```

```
rs$rss
```

```
## [1] 556739.2 533910.4 527077.5 522264.9 517255.5 514898.0 513429.3 511548.0
```

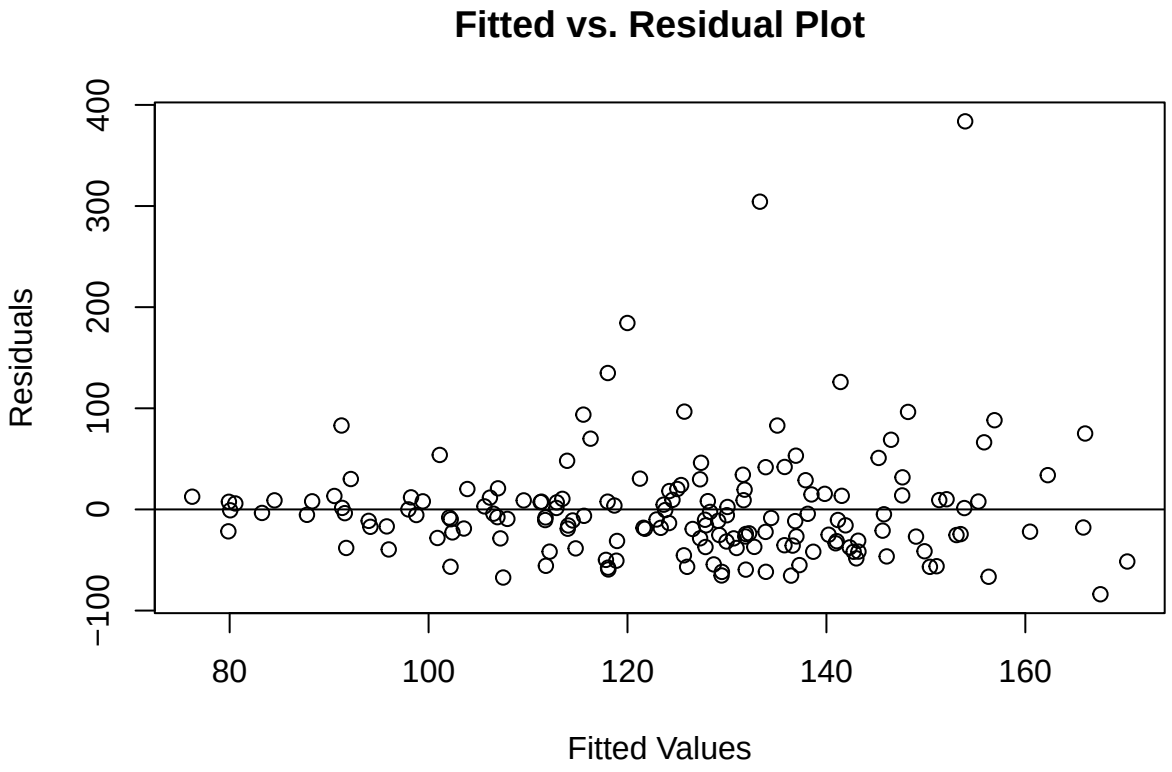
```
n <- nrow(train)
p <- 2:10
AIC <- n*log(rs$rss / n) + 2 * p
plot(AIC ~ I(p - 1), ylab = "AIC", xlab = "Number of Predictors", col = "blue")
```



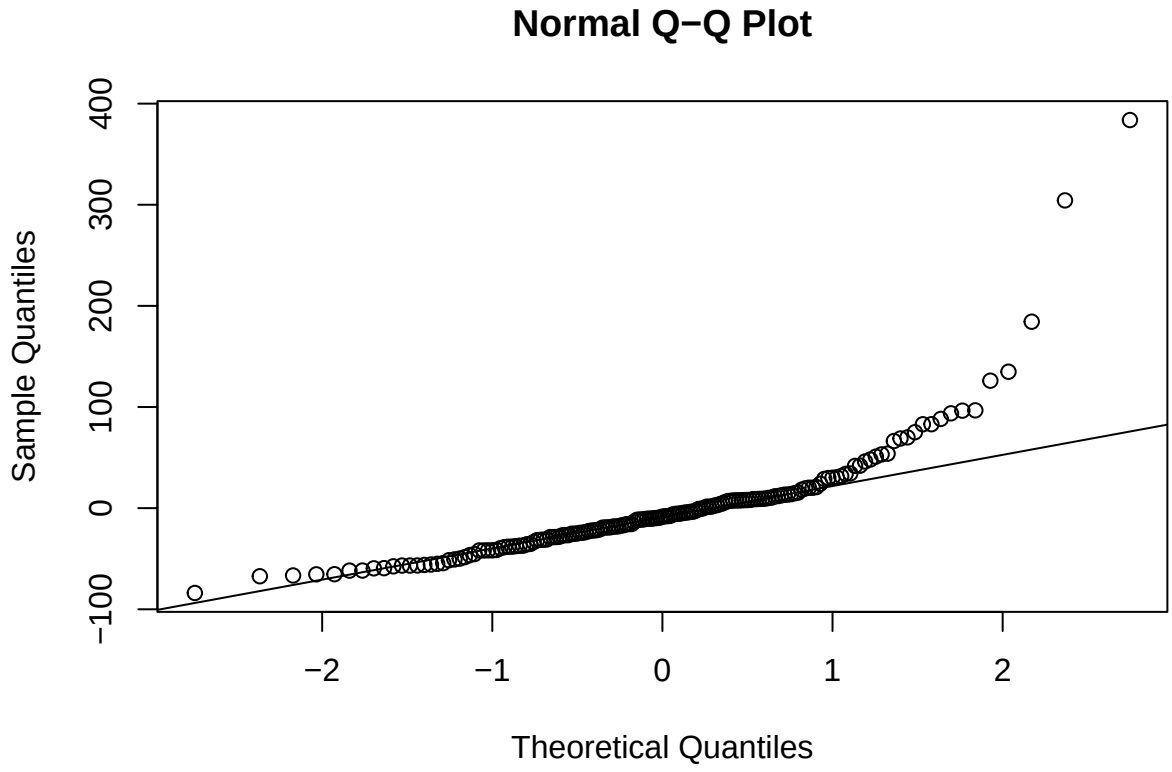

```
# Third has the loest AIC
model1 <- lm(ogtt ~ weight + age + edu, train)
summary(model1)
```

```
##
## Call:
## lm(formula = ogtt ~ weight + age + edu, data = train)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -83.86 -29.80  -8.45  11.81 383.74
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  92.87428   25.53144   3.638 0.000369 ***
## weight        0.13930    0.09583   1.454 0.147969
## age           0.85547    0.25677   3.332 0.001068 **
## edu         -10.10007    3.64779  -2.769 0.006278 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 56.86 on 163 degrees of freedom
## Multiple R-squared:  0.1115, Adjusted R-squared:  0.09515
## F-statistic: 6.818 on 3 and 163 DF,  p-value: 0.0002338
```

```
train_pred1 <- predict(model1, newdata = train)
test_pred1 <- predict(model1, newdata = test)
plot(train_pred, residuals(model1), xlab = "Fitted Values", ylab = "Residuals", main = "Fitted vs. Resi
abline(h = 0)
```




```
qqnorm(residuals(model1))
qqline(residuals(model1))
```




```
train_pred1 <- predict(model1, newdata = train)
data <- data.frame(actual = test$ogtt, predicted = train_pred1)
```

```
## Warning in data.frame(actual = test$ogtt, predicted = train_pred1): row names
## were found from a short variable and have been discarded
```

```r
# plot predicted vs. actual
plot(data$actual, data$predicted, xlab = "Actual", ylab = "Predicted", main = "Predicted vs. Actual")

# add a reference line for perfect predictions
abline(a = 0, b = 1, col = "red")
```