Home Work 3

Shashi Roshan (NetID : sroshan2)

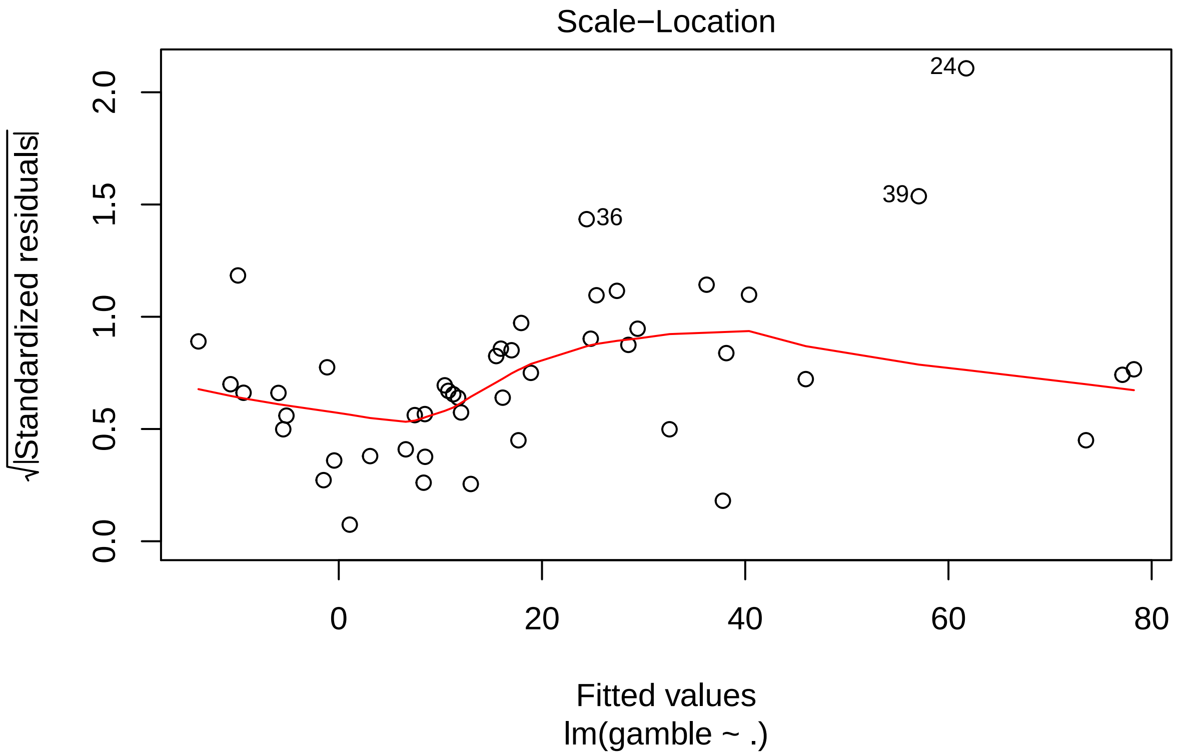
```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

Question 1 (a)

```
data(teengamb)

m1 = lm(gamble ~ ., data = teengamb)

plot(m1, which=3)
```



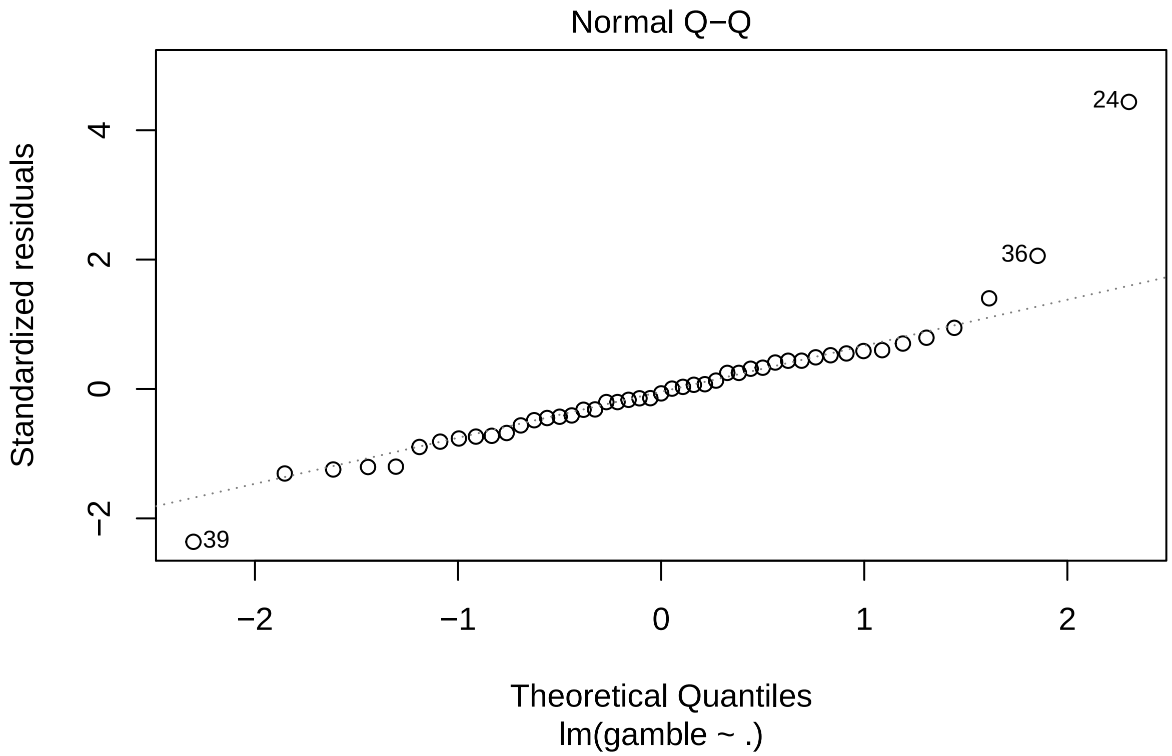
```r
bptest(m1)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  m1
## BP = 6.4288, df = 4, p-value = 0.1693
```

From the scale-location graph, we can see that the variance is constant. Also, we performed the Breusch Pagen test, which gave a p value above the threshold, hence we cannot reject the null hypothesis that the error variance is homoskedastic.

Question 1 (b)

```r
plot(m1, which=2)
```



From the Normal Q-Q plot above, we can see that all the residuals are not on the straight line, hence the residuals do not follow the normality assumption. Also, we need to investigate these points further : 39, 36, 24.

Question 1 (c)

```r
hat_matrix_val = hatvalues(m1)
model_matrix = model.matrix(m1)
cutoff = (2 * ncol(model_matrix)) / (nrow(model_matrix))
leverage_points = which(hat_matrix_val >= cutoff)
leverage_points
```

```
## 31 33 35 42
## 31 33 35 42
```

High leverage points are : 31, 33, 35, 42

Question 1 (d)

```
# Assuming outliers are outside the range of [mean - 1.5*IQR, mean + 1.5*IQR], where IQR is the interqu
mean_val = mean(teengamb$gamble)
max_limit = mean_val + IQR(teengamb$gamble) * 1.5
min_limit = mean_val - IQR(teengamb$gamble) * 1.5

# Finding the outliers
which(teengamb$gamble > max_limit | teengamb$gamble < min_limit)
```

```
## [1] 24 31 32 33 36 38 42
```

```
# Number of outliers
length(which(teengamb$gamble > max_limit |
               teengamb$gamble < min_limit))
```

```
## [1] 7
```

Question 1 (e)

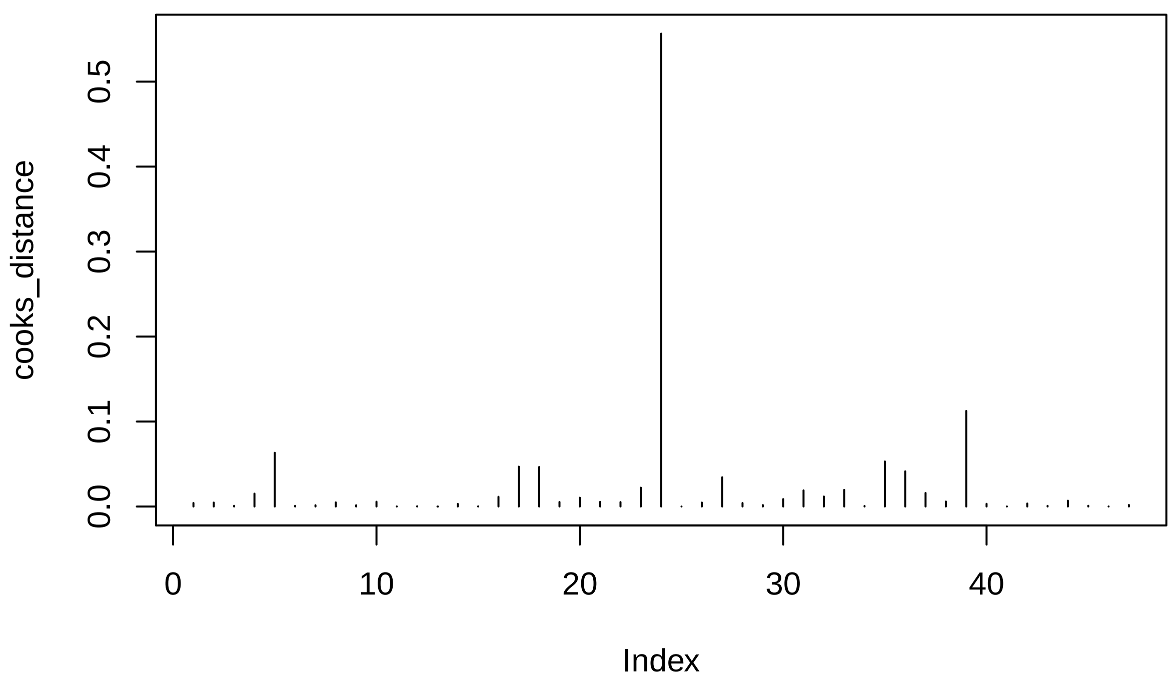
```
cutoff = 0.5
cooks_distance = cooks.distance(m1)
outliers = which(cooks_distance >= cutoff)
outliers
```

```
## 24
## 24
```

```
cutoff = 1
cooks_distance = cooks.distance(m1)
outliers = which(cooks_distance >= cutoff)
outliers
```

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

```r
plot(cooks_distance, type = "h")
```



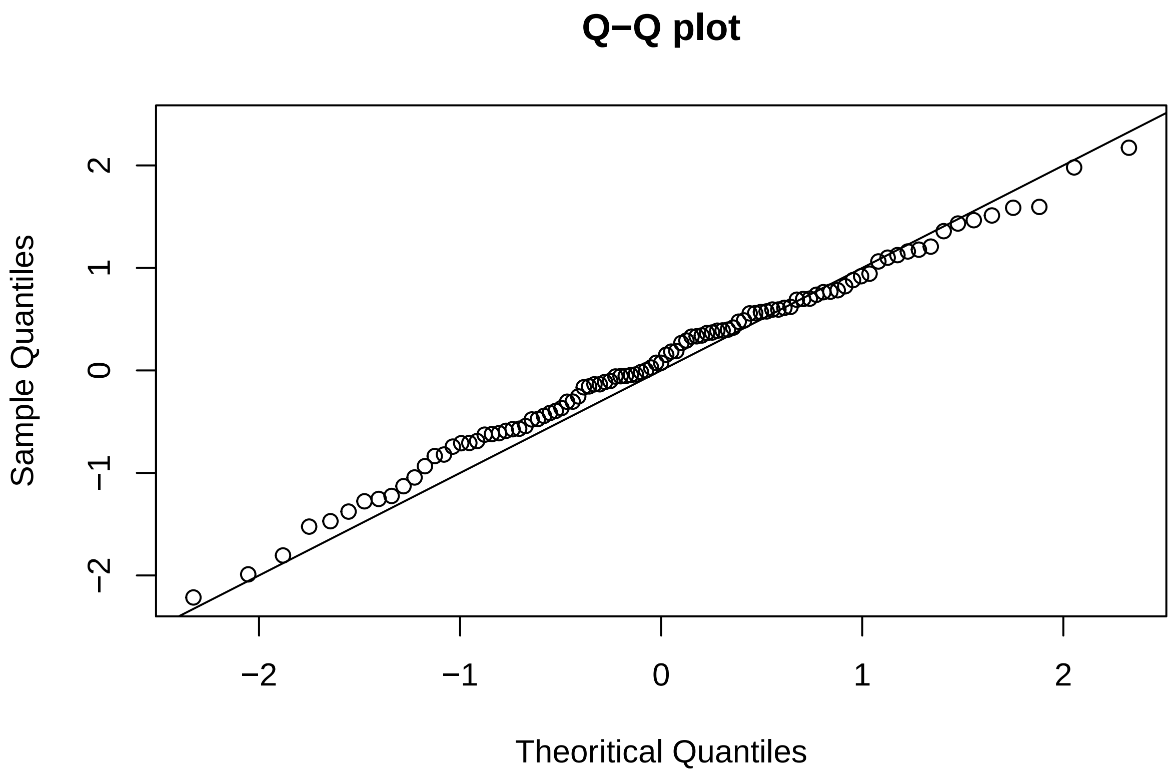
Using cook's distance, and cutoff of 0.5, there is one influential point : 24
Using cook's distance, and cutoff of 1, there are no influential points.

Question 2 (a)

```r
set.seed(1)

random_data = rnorm(100, mean = 0, sd = 1)
```

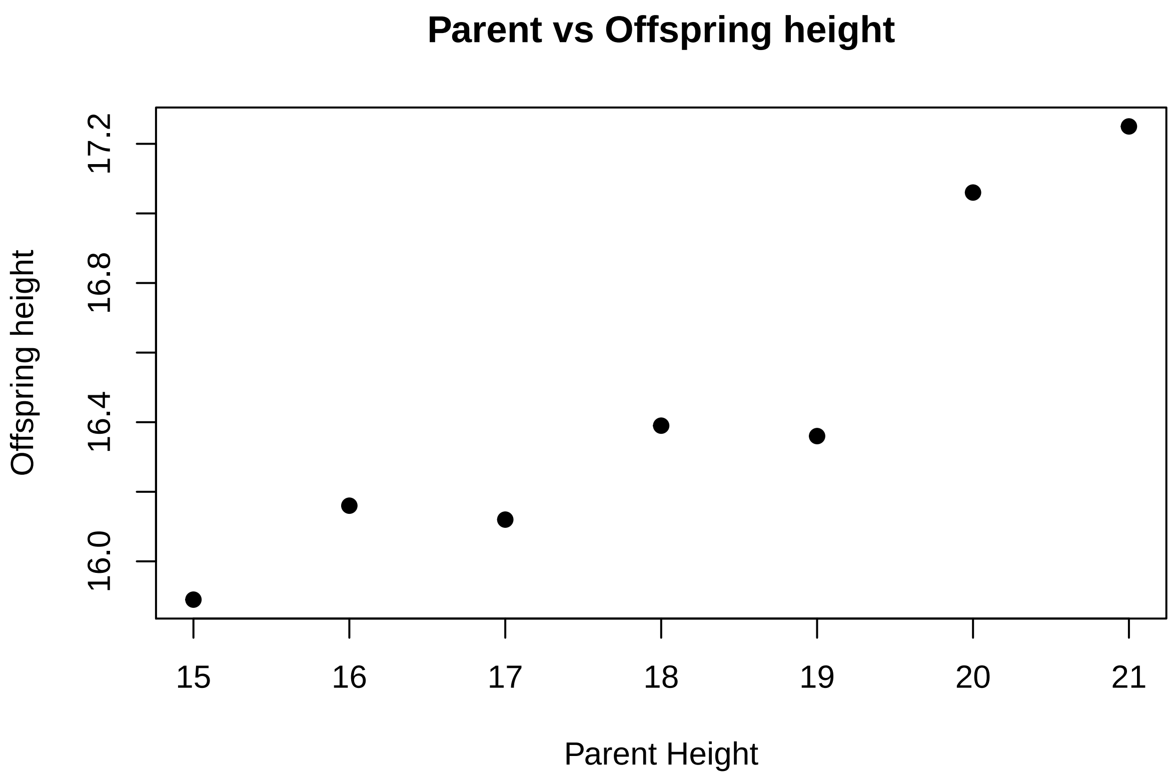
Question 2 (b)

```r
plot(
  qnorm(seq(
    from = 0.01, to = 1, by = 0.01
  )),
  sort(random_data),
  main = "Q-Q plot",
  xlab = "Theoritical Quantiles",
  ylab = "Sample Quantiles"
)

abline(0, 1)
```



Question 3 (a)

```
parent <- 15:21
offspring <- c(15.89, 16.16, 16.12, 16.39, 16.36, 17.06, 17.25)
sd = c(1.764, 1.595, 1.655, 2.036, 1.895, 1.937, 1.987)
Gru <- data.frame(parent, offspring, sd)

plot(
  parent,
  offspring,
  main = "Parent vs Offspring height",
  xlab = "Parent Height",
  ylab = "Offspring height",
  pch = 19
)
```



Question 3 (b)

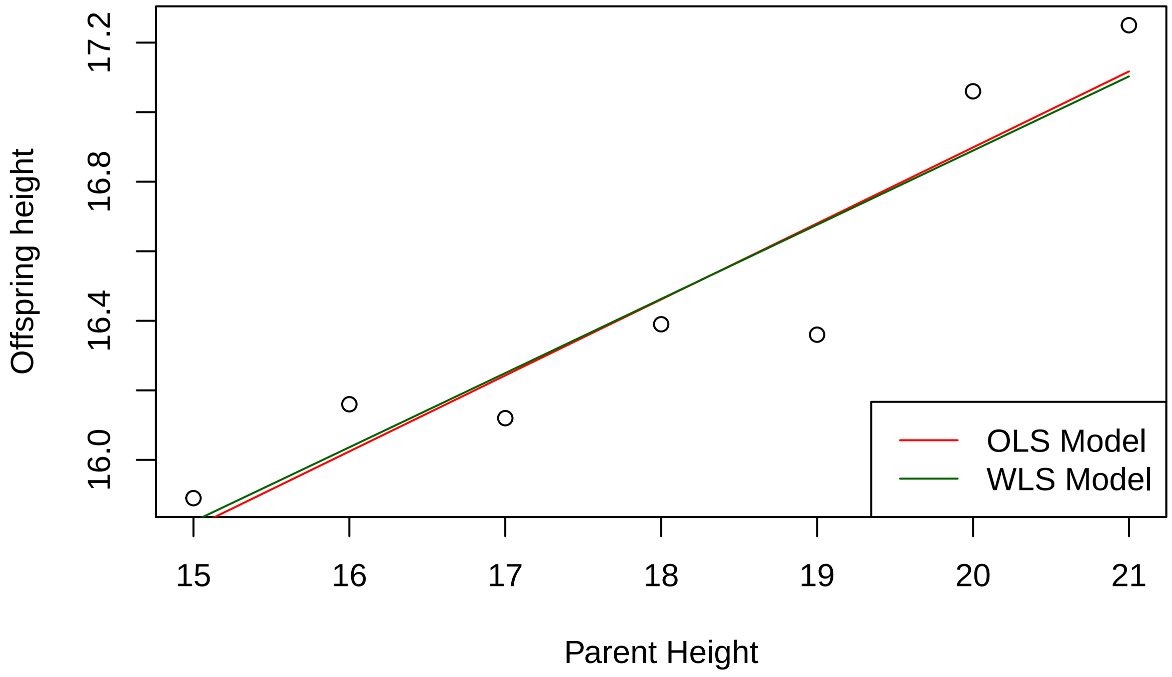
```
ols_model = lm(offspring ~ parent, data = Gru)
```

Question 3 (c)

```
wls_model = lm(offspring ~ parent, data = Gru, weights = sd ^ -2)
```

Question 3 (d)

```
plot(parent, offspring, xlab = "Parent Height",
     ylab = "Offspring height")

lines(parent, predict(ols_model, Gru), col = "red")

lines(parent, predict(wls_model, Gru), col = "darkgreen")

legend(
  "bottomright",
  legend = c("OLS Model", "WLS Model"),
  col = c("red", "darkgreen"),
  lty = 1
)
```



Question 3 (e)

```
t_statistic = (summary(wls_model)[['coefficients']][2, 1] - 0.5) / summary(wls_model)[['coefficients']]
p_value = pt(t_statistic, df = 5)
p_value
```

```
## [1] 0.0002750757
```

The t-test statistic for Beta_1 at 0.5 is -7.8144399.
Since the corresponding p-value is 2.7507575×10^{-4} , which is less than the threshold, we can reject the null hypothesis that Beta_1 is 0.5.
The distribution of null hypothesis is t-distirbution with degrees of freedom 5.

How is this different from previous scanerio :
In previous case, we were testing if all Beta values were equal to zero. In current case, our we are only testing if Beta_1 is equal to 0.5.

For hypothesis Beta_1 $<$ 0.5 : The p-value will be above threshold around Beta_1 $=$ 0.2. This hypothesis will become true then.

Question 4 (a)

```
library(faraway)
data(strongx)

diag_mat = diag(10)
diag_mat = as.matrix(diag_mat)
# var = (1/weight)
# weight = (1/var) = (1/(sd^2)) = sd^(-2)
wt = strongx$sd ^ -2
sigma_inverse = as.matrix(wt * diag_mat)

sigma = solve(sigma_inverse)

# Sigma matrix
sigma
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##  [1,]  289    0    0    0    0    0    0    0    0     0
##  [2,]    0   81    0    0    0    0    0    0    0     0
##  [3,]    0    0   81    0    0    0    0    0    0     0
##  [4,]    0    0    0   49    0    0    0    0    0     0
##  [5,]    0    0    0    0   49    0    0    0    0     0
##  [6,]    0    0    0    0    0   36    0    0    0     0
##  [7,]    0    0    0    0    0    0   36    0    0     0
##  [8,]    0    0    0    0    0    0    0   36    0     0
##  [9,]    0    0    0    0    0    0    0    0   25     0
## [10,]    0    0    0    0    0    0    0    0    0    25
```

```
# Sigma inverse matrix
sigma_inverse
```

```
##              [,1]       [,2]       [,3]       [,4]       [,5]       [,6]
##  [1,] 0.003460208 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000
##  [2,] 0.000000000 0.01234568 0.00000000 0.00000000 0.00000000 0.00000000
##  [3,] 0.000000000 0.00000000 0.01234568 0.00000000 0.00000000 0.00000000
##  [4,] 0.000000000 0.00000000 0.00000000 0.02040816 0.00000000 0.00000000
##  [5,] 0.000000000 0.00000000 0.00000000 0.00000000 0.02040816 0.00000000
##  [6,] 0.000000000 0.00000000 0.00000000 0.00000000 0.00000000 0.02777778
##  [7,] 0.000000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000
##  [8,] 0.000000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000
##  [9,] 0.000000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000
## [10,] 0.000000000 0.00000000 0.00000000 0.00000000 0.00000000 0.00000000
##             [,7]       [,8] [,9] [,10]
##  [1,] 0.00000000 0.00000000 0.00  0.00
##  [2,] 0.00000000 0.00000000 0.00  0.00
##  [3,] 0.00000000 0.00000000 0.00  0.00
##  [4,] 0.00000000 0.00000000 0.00  0.00
##  [5,] 0.00000000 0.00000000 0.00  0.00
##  [6,] 0.00000000 0.00000000 0.00  0.00
##  [7,] 0.02777778 0.00000000 0.00  0.00
##  [8,] 0.00000000 0.02777778 0.00  0.00
##  [9,] 0.00000000 0.00000000 0.04  0.00
## [10,] 0.00000000 0.00000000 0.00  0.04
```

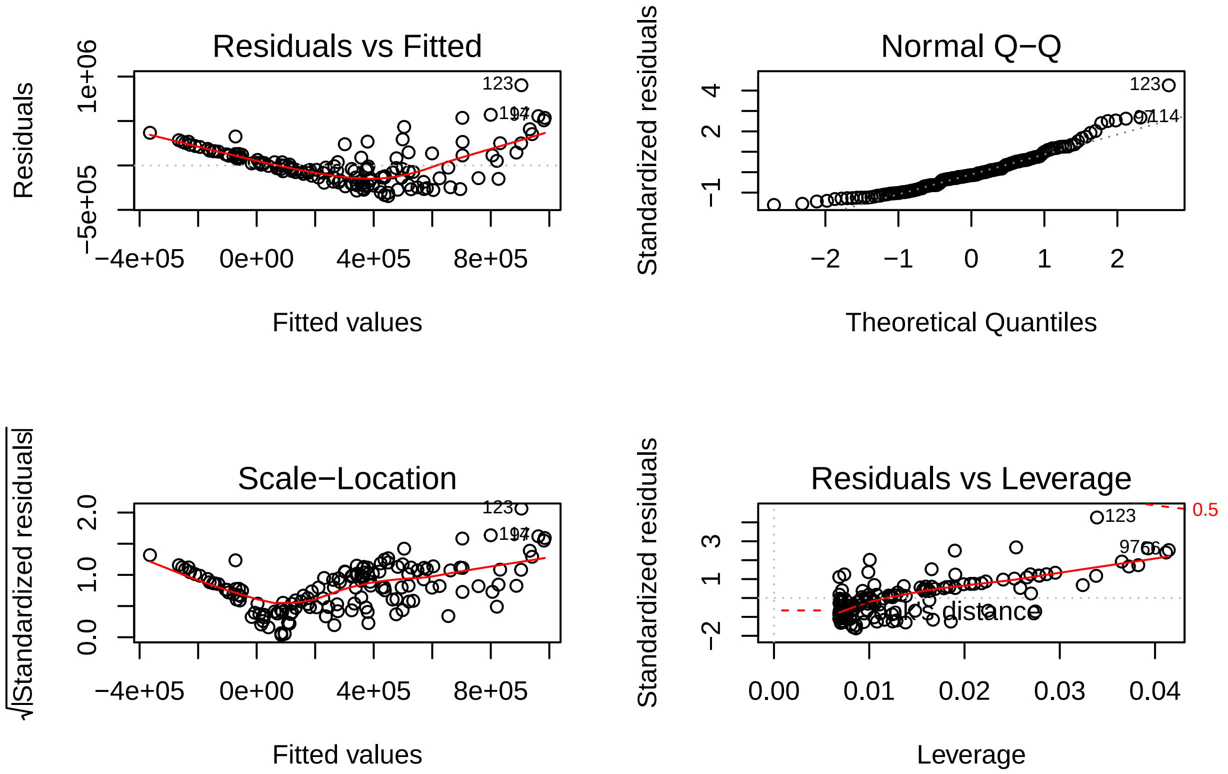
Question 4 (b)

```
m1 = lm(crossx ~ energy , data = strongx)
X = model.matrix(m1)

# Beta_hat (GLS) = (X' Sig^-1 X)^-1 (X'Sig^-1 y)
beta_hat = solve(t(X) %*% sigma_inverse %*% X) %*% t(X) %*% sigma_inverse %*%
  strongx$crossx

beta_hat
```

```
##                  [,1]
## (Intercept) 148.4732
## energy      530.8354
```

Question 5 (a)

```
lpga_data = read.csv("lpga2009.csv")
model_lgpa = lm(prize ~ percentile, data = lpga_data)
par(mfrow = c(2, 2))

plot(model_lgpa)
```



Question 5 (b)

The residuals are not normally distributed (from the analysis of the Normal Q-Q plot).

Question 5 (c)

```
# Assuming outliers are outside the range of [mean - 1.5*IQR, mean + 1.5*IQR]
mean_val = mean(lpga_data$prize)
max_limit = mean_val + IQR(lpga_data$prize) * 1.5
min_limit = mean_val - IQR(lpga_data$prize) * 1.5

# Finding the outliers
which(lpga_data$prize > max_limit | lpga_data$prize < min_limit)
```

```
##  [1]  17  20  42  60  66  69  74  96  97 103 104 114 123 129 134 137 138
```

```
# Number of outliers
length(which(lpga_data$prize > max_limit |
               lpga_data$prize < min_limit))
```

```
## [1] 17
```

Question 5 (d)

By analysing the "Residuals vs Leverage" plot, we can conclude that there are no influential points which have cook's distance greater than 1.

Question 5 (e)

```r
bptest(model_lgpa)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model_lgpa
## BP = 24.895, df = 1, p-value = 6.055e-07
```

The error variance is not constant.
We can reject the null hypothesis that the error variance is constant, since the p-value of BP test is less than threshold.