

Problem Set 3, Winter 2024

Michael Ghattas

```
library(MASS)
library(ggplot2)
```

CONTEXT: Factorial experiment with doughnuts

Donna is the owner of a boutique doughnut shop. Because many of her customers are conscious of their fat intake but want the flavor of fried doughnuts, she decided to develop a doughnut recipe that minimizes the amount of fat that the doughnuts absorb from the fat in which the doughnuts are fried.

She conducted a factorial experiment that had a similar procedure as Lowe (1935). Like Lowe, she used four types of fats (fat_type). She also used three types of flour (flour_type): all-purpose flour, whole wheat flour, and gluten-free flour. For each combination of fat type and flour type, she cooked six identical batches of doughnuts. Each batch contained 24 doughnuts, and the total fat (in grams) absorbed by the doughnuts in each batch was recorded (sim_tot_fat).

Question 1 - 5 points

You may need to process your data before you begin your analysis. Specifically, you will need to make sure that the variable type is set to 'factor' for both of your grouping variables and 'num' for your outcome variable.

```
# Load data
doughnuts.factorial <- read.csv("doughnutsfactorial-1.csv", header = TRUE, sep = ",")
```

Like in Problem Set 1, please create two new variables in the doughnuts.factorial data set. The first new variable will be called fat_type_factor and will contain the same values as in the fat_type variable but will have a variable type of factor. The second new variable will be called flour_type_factor and will contain the same values as in the flour_type variable but will also have a variable type of factor.

```
# Create factor variables
doughnuts.factorial$fat_type_factor <- as.factor(doughnuts.factorial$fat_type)
doughnuts.factorial$flour_type_factor <- as.factor(doughnuts.factorial$flour_type)
```

Check your work by running the following code chunk. Be sure that fat_type_factor and flour_type_factor are factor-type variables before you complete the rest of the problem set.

```
# Check structure
str(doughnuts.factorial)

## 'data.frame':    72 obs. of  5 variables:
## $ fat_type      : chr  "Canola" "Canola" "Canola" "Canola" ...
## $ flour_type    : chr  "ap" "ap" "ap" "ap" ...
## $ sim_tot_fat   : int   78 71 80 88 62 72 78 75 89 74 ...
## $ fat_type_factor : Factor w/ 4 levels "Canola","Peanut",...: 1 1 1 1 1 1 3 3 3 3 ...
## $ flour_type_factor: Factor w/ 3 levels "ap","gf","ww": 1 1 1 1 1 1 1 1 1 1 ...
```

Question 2 - 10 points

Using the code in the code chunk below, I fitted a regression model that has a specification equivalent to a one-way ANOVA. Specifically, I used the `doughnuts.factorial` data set to re-create the one-way ANOVA with `sim_tot_fat` as the outcome variable and `fat_type_factor` as the grouping variable. This is the same model as the first model you estimated in Problem Set 2, Question 4. Run this code before continuing with this question.

```
# Fitting the one-way ANOVA equivalent regression model
doughnuts.reg <- lm(sim_tot_fat ~ fat_type_factor, data=doughnuts.factorial)
summary(doughnuts.reg)
```

```
##
## Call:
## lm(formula = sim_tot_fat ~ fat_type_factor, data = doughnuts.factorial)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -35.944  -4.736  -0.167   5.514  21.056
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      66.944      2.529  26.467 < 2e-16 ***
## fat_type_factorPeanut      8.722      3.577   2.438 0.017372 *
## fat_type_factorShortening  11.722      3.577   3.277 0.001654 **
## fat_type_factorSunflower -13.611      3.577  -3.805 0.000306 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.73 on 68 degrees of freedom
## Multiple R-squared:  0.4708, Adjusted R-squared:  0.4475
## F-statistic: 20.17 on 3 and 68 DF,  p-value: 1.856e-09
```

You will now use the output of this regression model to answer the following four questions. Note that the `lm()` function dummy coded the fat type variable for you.

Question A: If you plug in the appropriate zeros and ones into the estimated regression equation to obtain the mean of the *Canola oil* group, which of the four terms shown in the regression output remain (i.e., would not be multiplied by zero)?

Your answer here: If you plug in the appropriate zeros and ones to obtain the mean of the Canola oil group, only the intercept term remains. This is because the Canola oil group is the reference group in the dummy coding, meaning the other terms (`fat_type_factorPeanut`, `fat_type_factorShortening`, and `fat_type_factorSunflower`) will be multiplied by zero.

Question B: If you plug in the appropriate zeros and ones into the estimated regression equation to obtain the mean of the *Peanut oil* group, which of the four terms shown in the regression output remain (i.e., would not be multiplied by zero)?

Your answer here: For the Peanut oil group, the intercept term remains, and the coefficient for `fat_type_factorPeanut` also remains because the dummy variable for Peanut oil will take a value of 1.

Question C: Based on the output, which fat type had a *lower* mean than that of Canola oil?

Your answer here: The fat type that had a lower mean than Canola oil is Sunflower oil, as indicated by the negative coefficient (-13.611).

Question D: Based on the output, which of the three other fat types - Peanut oil Shortening, and Sunflower oil - was significantly different from Canola oil? For full credit, list the names of the fat types (if any) that are significantly different from Canola oil.

Your answer here: All three fat types (Peanut oil, Shortening, and Sunflower oil) are significantly different from Canola oil.

Question 3 - 10 points

In Problem Set 2, Question 5, you conducted a two-way factorial ANOVA with an interaction. First, copy the code you wrote for Problem Set 2, Question 5 into the code chunk below and display the results using the `summary()` function. You won't answer any questions about the two-way ANOVA model directly, but you should notice some similarities between the two-way ANOVA with interaction output and your equivalently-specified regression model that may help you answer the questions about the regression model.

```
# Two-way ANOVA with interaction between fat and flour type
fat_flour_int.aov <- aov(sim_tot_fat ~ fat_type_factor * flour_type_factor, data = doughnuts.factorial)
summary(fat_flour_int.aov)
```

```
##                                Df Sum Sq Mean Sq F value    Pr(>F)
## fat_type_factor                 3   6967   2322.5    21.976 1.01e-09 ***
## flour_type_factor               2   1063    531.3     5.028 0.00958 **
## fat_type_factor:flour_type_factor 6    427     71.2     0.674 0.67095
## Residuals                      60   6341    105.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Next, conduct a regression analysis that is equivalently-specified; that is, it should have `sim_tot_fat` as the outcome and `fat_type_factor`, `flour_type_factor`, and their interaction as predictors. (Hint: much like in the `aov()` function, interactions are specified in `lm()` by using `*` between the two variables that interact). Display the summary of this model using the `summary()` function. Once you have done this, answer the five questions below based on the regression model output.

```
# Equivalently-specified regression model with interaction
doughnuts.reg <- lm(sim_tot_fat ~ fat_type_factor * flour_type_factor, data = doughnuts.factorial)

# Display the summary
summary(doughnuts.reg)
```

```
##
## Call:
## lm(formula = sim_tot_fat ~ fat_type_factor * flour_type_factor,
##     data = doughnuts.factorial)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -28.333  -5.958  -0.250   6.667  21.667
##
## Coefficients:
##                                Estimate Std. Error t value
## (Intercept)                   75.167      4.197   17.910
## fat_type_factorPeanut           3.667      5.935    0.618
```

```
## fat_type_factorShortening          7.167      5.935      1.207
## fat_type_factorSunflower          -15.167      5.935     -2.555
## flour_type_factorgf              -8.833      5.935     -1.488
## flour_type_factorww             -15.833      5.935     -2.668
## fat_type_factorPeanut:flour_type_factorgf      2.333      8.394      0.278
## fat_type_factorShortening:flour_type_factorgf   3.667      8.394      0.437
## fat_type_factorSunflower:flour_type_factorgf  -3.833      8.394     -0.457
## fat_type_factorPeanut:flour_type_factorww     12.833      8.394      1.529
## fat_type_factorShortening:flour_type_factorww  10.000      8.394      1.191
## fat_type_factorSunflower:flour_type_factorww    8.500      8.394      1.013
##                                     Pr(>|t|)
## (Intercept)                            < 2e-16 ***
## fat_type_factorPeanut                   0.53906
## fat_type_factorShortening               0.23199
## fat_type_factorSunflower                0.01316 *
## flour_type_factorgf                    0.14191
## flour_type_factorww                    0.00981 **
## fat_type_factorPeanut:flour_type_factorgf      0.78198
## fat_type_factorShortening:flour_type_factorgf  0.66380
## fat_type_factorSunflower:flour_type_factorgf   0.64954
## fat_type_factorPeanut:flour_type_factorww      0.13154
## fat_type_factorShortening:flour_type_factorww  0.23820
## fat_type_factorSunflower:flour_type_factorww    0.31529
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.28 on 60 degrees of freedom
## Multiple R-squared:  0.5715, Adjusted R-squared:  0.493
## F-statistic: 7.275 on 11 and 60 DF, p-value: 1.026e-07
```

Question A: Not counting the intercept, how many predictor coefficients were estimated in this model?

Your answer here: There are 11 predictor coefficients estimated in this model. These include:

Question B: Not counting the intercept, how many predictor coefficients in this model are associated with the *main effect* of fat type? Hint: ANOVA is a special case of regression and will not contradict the results of an equivalently-specified regression model.

Your answer here: Three predictor coefficients are associated with the main effect of fat type (fat_type_factorPeanut, fat_type_factorShortening, and fat_type_factorSunflower).

Question C: Not counting the intercept, how many predictor coefficients in this model are associated with the *main effect* of flour type? Hint: ANOVA is a special case of regression and will not contradict the results of an equivalently-specified regression model.

Your answer here: Two predictor coefficients are associated with the main effect of flour type (flour_type_factorgf and flour_type_factorww).

Question D: Not counting the intercept, how many predictor coefficients in this model are associated with the *interaction* between fat type and flour type? Hint: ANOVA is a special case of regression and will not contradict the results of an equivalently-specified regression model.

Your answer here: Six predictor coefficients are associated with the interaction between fat type and flour type (3 for the interaction between fat_type_factor and flour_type_factorgf, and 3 for the interaction between fat_type_factor and flour_type_factorww).

Question E: What is the predicted amount of fat absorbed by a doughnut made from gluten-free flour and cooked in Sunflower oil? For full credit, you may use any valid method of finding this value (e.g., manually inputting values into an equation, using the predict() function, etc.), but you must show your work.

```
# Prediction for gluten-free flour and Sunflower oil
new_data <- data.frame(fat_type_factor = "Sunflower", flour_type_factor = "gf")
predict(doughnuts.reg, newdata = new_data)
```

```
##          1
## 47.33333
```

Your answer here: The predicted amount of fat absorbed by a doughnut made from gluten-free flour and cooked in Sunflower oil is approximately 47.33 grams.

CONTEXT - FISHERMAN DATA (many thanks to Dr. Durso for obtaining this data set)

Data Source: N.B. Al-Majed and M.R. Preston (2000). "Factors Influencing the Total Mercury and Methyl Mercury in the Hair of Fishermen in Kuwait," Environmental Pollution, Vol. 109, pp. 239-250.

<http://users.stat.ufl.edu/~winner/datasets.html>, downloaded on 4/23/2019

Description: Factors related to mercury levels among fishermen and a control group of non-fishermen.

Variables (names of variables in the data set)

Fisherman indicator ("fisherman"), categorical 0 = No 1 = Yes

Age in years ("age"), continuous

Residence Time in years ("restime"), continuous

Height in cm ("height"), continuous

Weight in kg ("weight"), continuous

Fish meals per week ("fishmlwk"), continuous

Parts of fish consumed ("fishpart"), categorical 0 = none 1 = muscle tissue only 2 = muscle tissue and sometimes whole fish 3 = whole fish

Methyl Mercury in mg/g ("MeHg"), continuous

Total Mercury in mg/g ("TotHg"), continuous

Preamble to Questions 4-5 - do this part before starting these questions!

Before moving on, set the variables you'll use to the proper data types by completing the lines in the code chunk below.

```
# Load fishermen data
fish <- read.csv("fishermen_mercury.csv", header=TRUE, sep=",")
fish$fishpart_factor <- as.factor(fish$fishpart)
```

Check your work by running the following code chunk. Be sure that fishmlwk and weight are integer-type or numeric variables (R should type these two appropriately automatically) and fishpart_factor is a factor-type variable before you complete the rest of the problem set.

```
str(fish)
```

```
## 'data.frame': 135 obs. of 10 variables:
## $ fisherman : int 1 1 1 1 1 1 1 1 1 1 ...
## $ age : int 45 38 24 41 43 58 45 46 46 46 ...
## $ restime : int 6 13 2 2 11 2 6 0 14 5 ...
## $ height : int 175 173 168 183 175 176 184 170 175 175 ...
## $ weight : int 70 73 66 80 78 75 85 68 80 75 ...
## $ fishmlwk : int 14 7 7 7 21 21 21 7 21 7 ...
## $ fishpart : int 2 1 2 1 1 1 1 2 1 1 ...
## $ MeHg : num 4.01 4.03 3.58 10.99 10.52 ...
## $ TotHg : num 4.48 4.79 3.86 11.44 10.85 ...
## $ fishpart_factor: Factor w/ 4 levels "0","1","2","3": 3 2 3 2 2 2 2 3 2 2 ...
```

Question 4 - 10 points

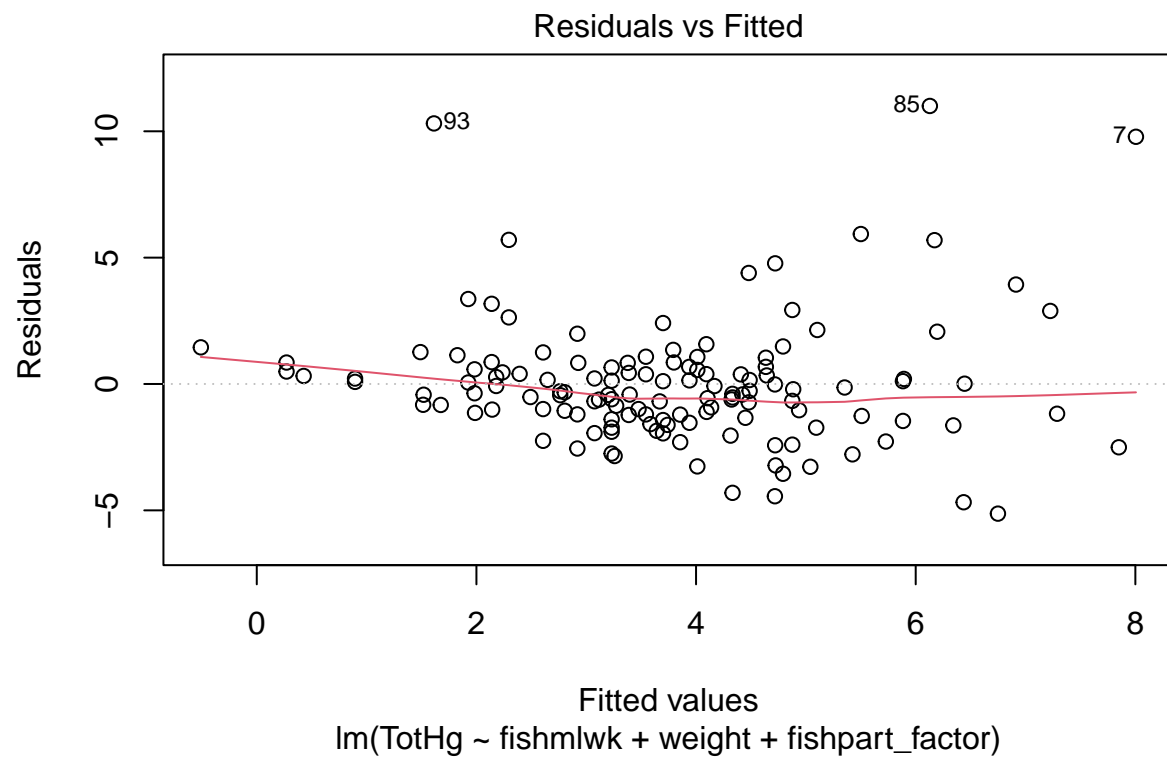
Fit a regression model with “TotHg” as the outcome variable and “fishmlwk”, “weight”, and “fishpart_factor” as predictor variables. Do not include interaction terms or polynomial terms in this model. Please display the model output using the `summary()` function.

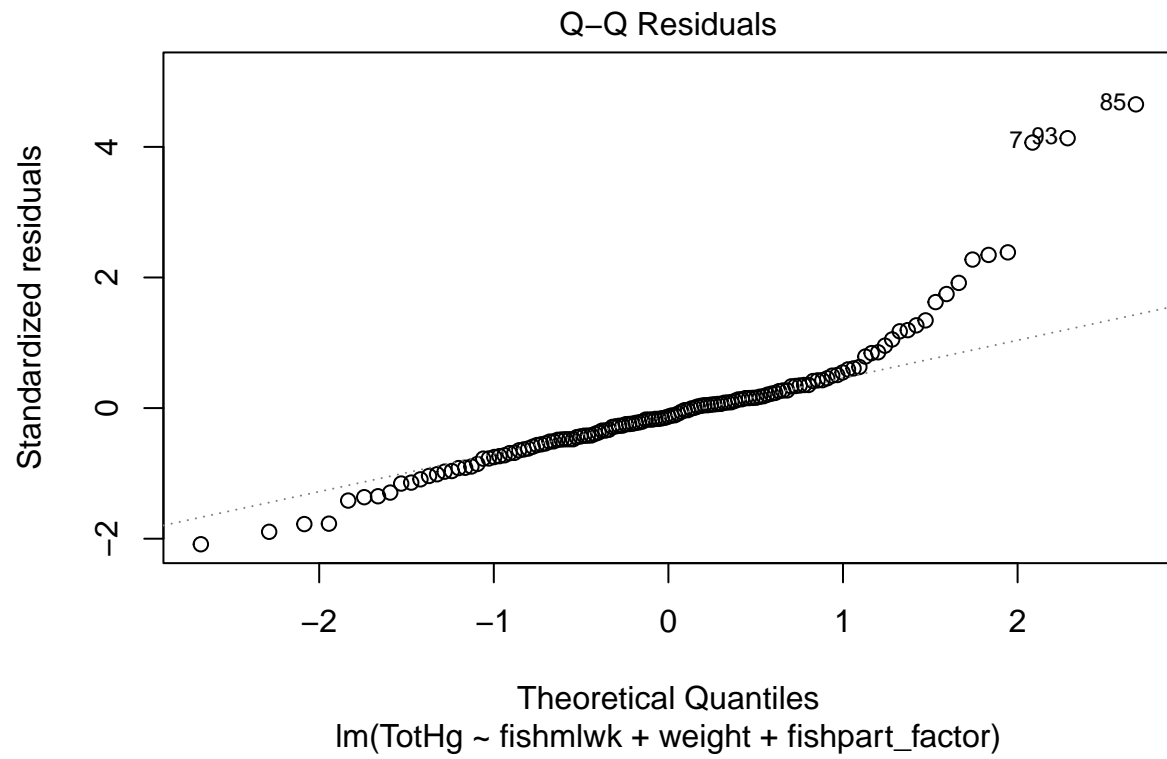
```
# Fit regression model
TotHg.reg <- lm(TotHg ~ fishmlwk + weight + fishpart_factor, data=fish)
summary(TotHg.reg)
```

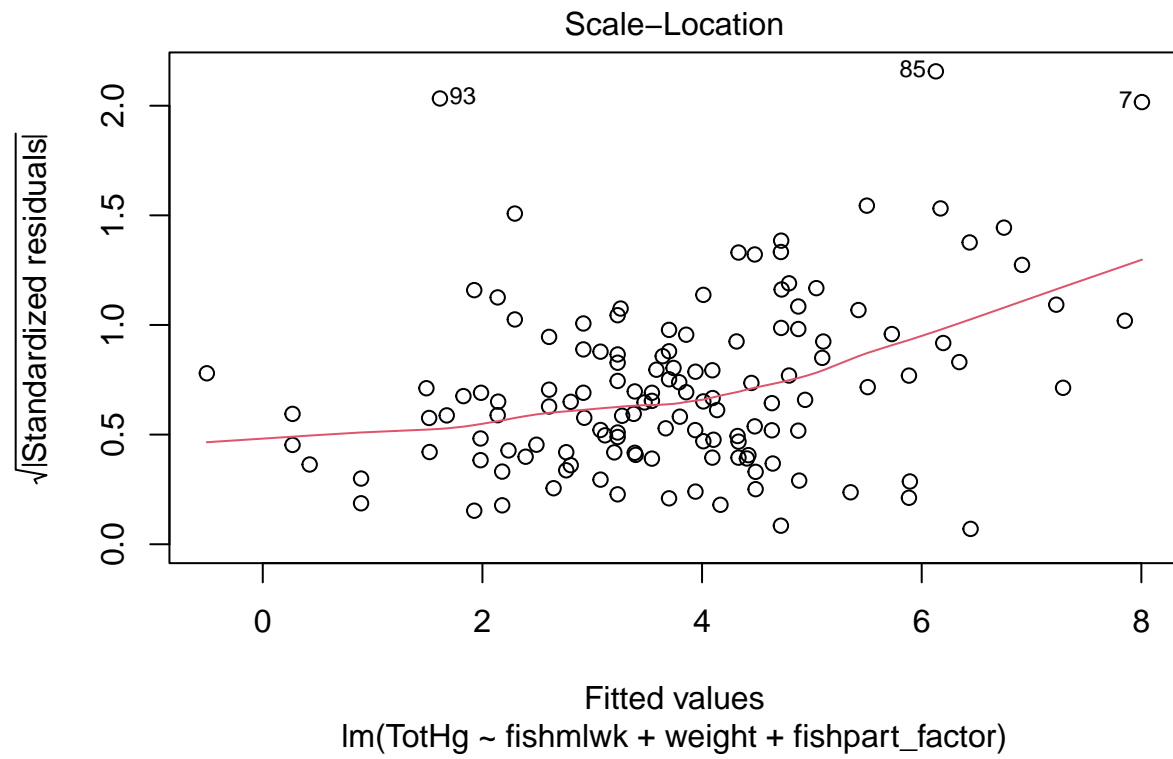
```
##
## Call:
## lm(formula = TotHg ~ fishmlwk + weight + fishpart_factor, data = fish)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.1298 -1.2455 -0.3262  0.6778 11.0020
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -10.02782    2.54490   -3.940 0.000133 ***
## fishmlwk         0.12320    0.04440    2.775 0.006347 **
## weight         0.15604    0.03431    4.549 1.23e-05 ***
## fishpart_factor1  2.18255    1.02701    2.125 0.035480 *
## fishpart_factor2  1.47379    0.89973    1.638 0.103854
## fishpart_factor3  2.55652    1.22244    2.091 0.038461 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.537 on 129 degrees of freedom
## Multiple R-squared:  0.2822, Adjusted R-squared:  0.2543
## F-statistic: 10.14 on 5 and 129 DF, p-value: 3.296e-08
```

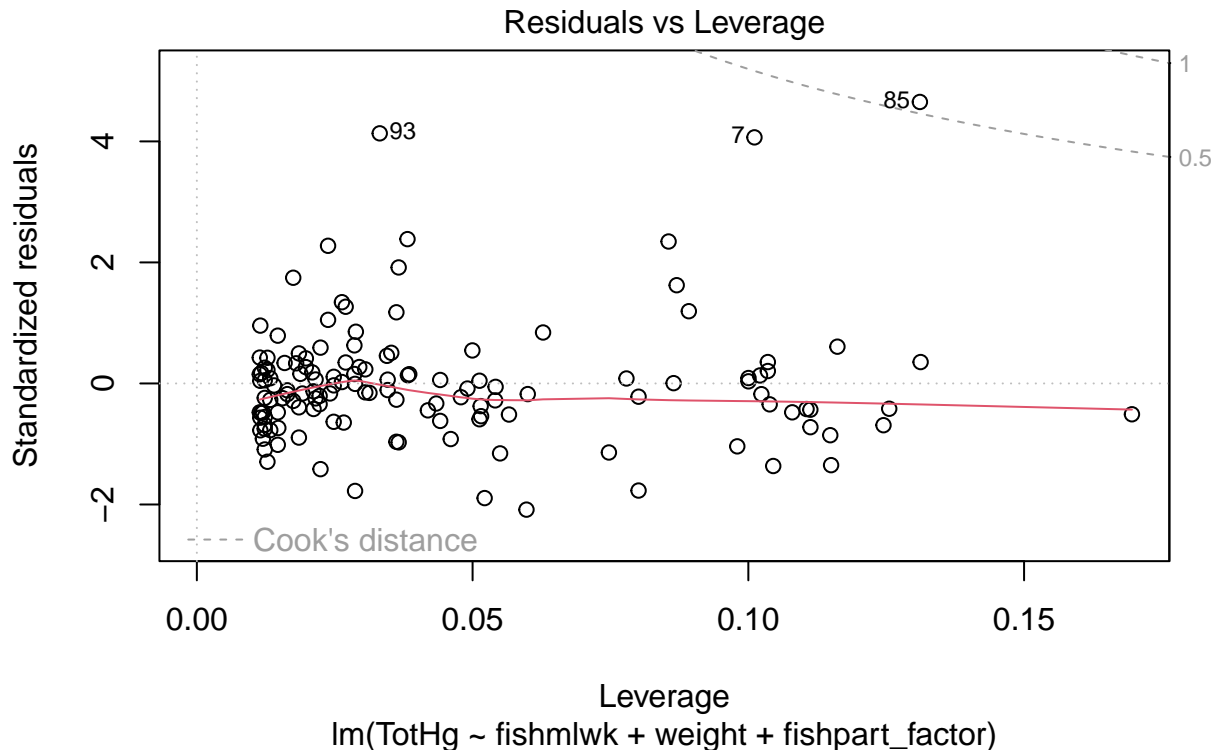
Next, generate the diagnostic plots for this model. Be sure that these are visible in your knitted document. Once you’ve done so, answer the questions below

```
# Diagnostic plots
plot(TotHg.reg)
```









Question A: The Residuals vs Fitted plot (the first plot) allows you to assess the assumptions of linearity and equality of residual variance across the range of the fitted values. What specific part of this plot should be examined to evaluate the *linearity* assumption?

Your answer here: Examine whether the residuals are scattered randomly around the red line without forming any noticeable patterns or trends.

Question B: The Normal Q-Q plot (the second plot) allows you to assess the assumption of normality across the range of the fitted values, and is interpreted similarly to plots you've seen previously. With regard to the assumption of *normality*, is the left side of the QQ plot more concerning or is the right side more concerning?

Your answer here (left or right): The right side is more concerning.

Question C: The Scale-Location plot (the third plot) allows you to assess the assumption of equality of residual variance across the range of the fitted values, and we discussed three shapes that indicate potential violations of this assumption (bowtie, bullhorns, and diamond). Are any of these shapes obviously present in this plot?

Your answer here (yes or no): Based on my visual reasoning, no, none of these shapes are obviously present in the plot.

Question D: The Residuals vs Leverage plot (the four plot) helps you identify potentially influential points. If you use the Cook's D guideline of 0.5, which point/s would you identify as being influential? Please list the observation number/s of any point/s you would consider removing based on this guideline.

Your answer here (list the observation number or numbers): Observation 85 should be considered for removal based on Cook's distance. Observations 93 and 70 may also need further investigation.

Question 5 - 5 points

If we follow a Cook's D guideline of 0.05, we may want to remove this potentially influential point. We can remove the point and see how the point's removal changes the model estimates and diagnostic plots.

First, remove observation 85 from the fish data set. Save the data set with the removed point into a data set called "fish.remove"

```
# Remove observation 85
fish.remove <- fish[-85, ]
```

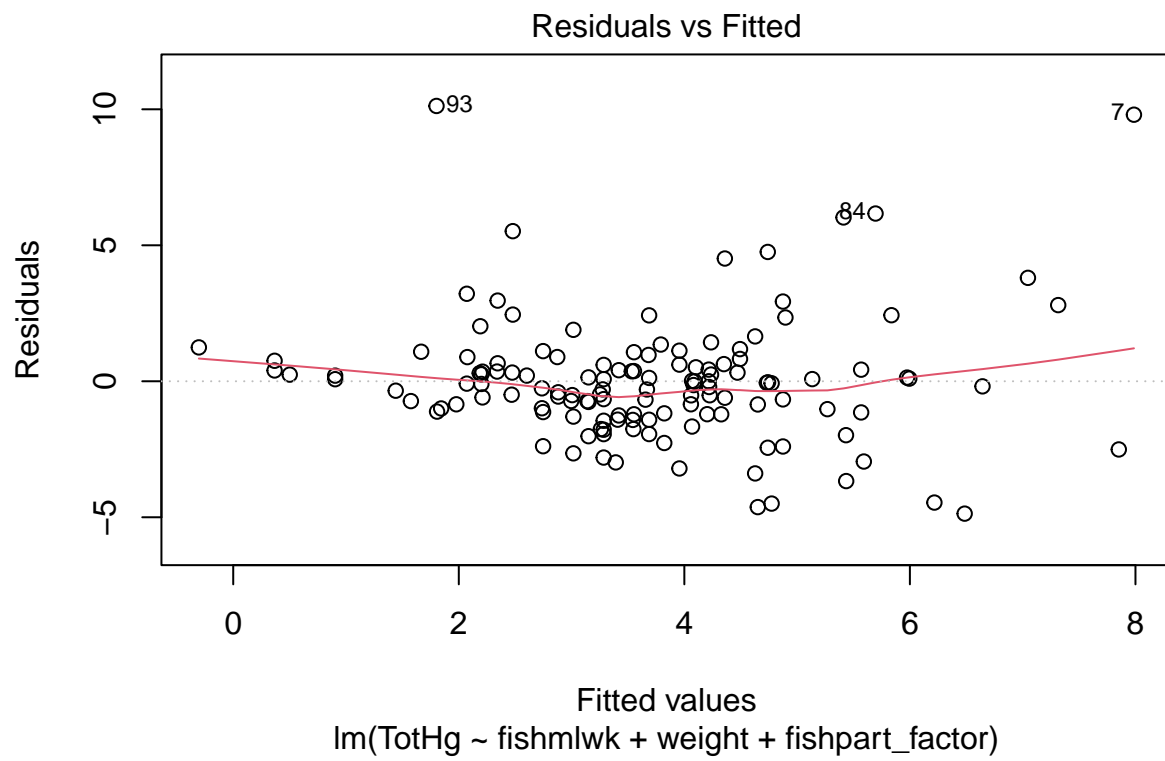
Next, re-fit the *same* model as you fitted in Question 4 and save it as a model object called "fish.reg.remove". This time, use the fish.remove data set.

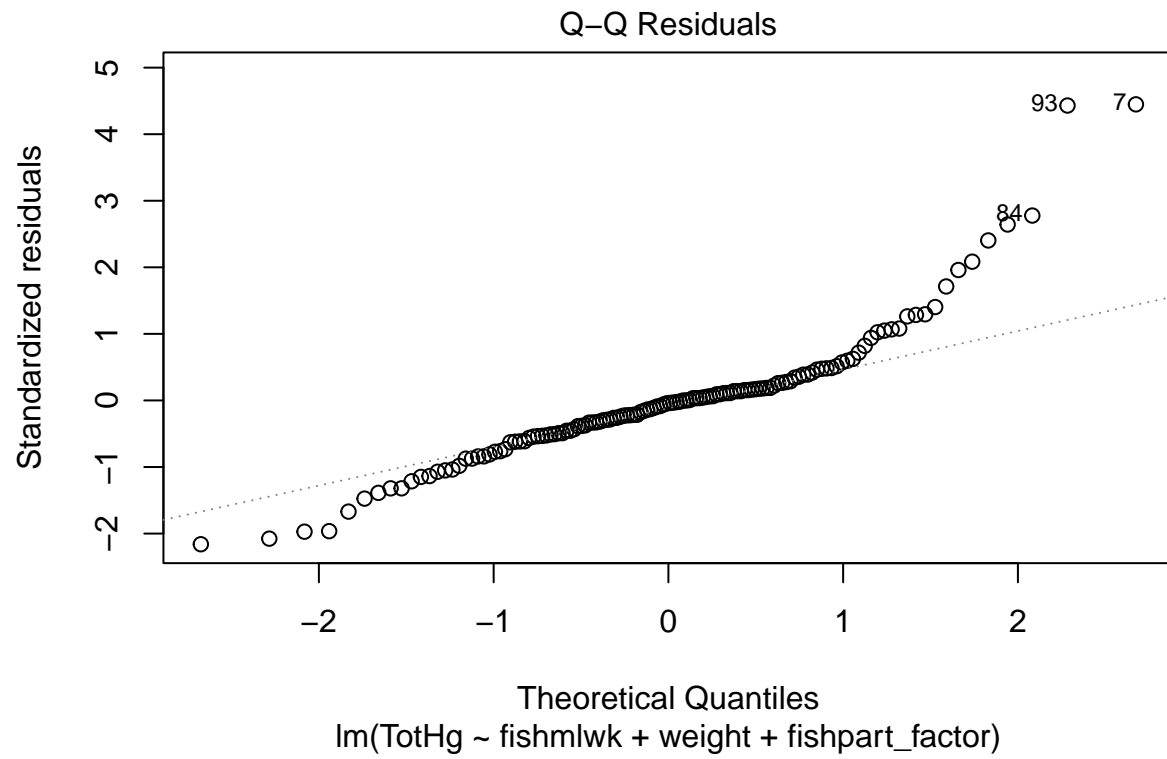
```
# Refit the model
fish.reg.remove <- lm(TotHg ~ fishmlwk + weight + fishpart_factor, data=fish.remove)
summary(fish.reg.remove)
```

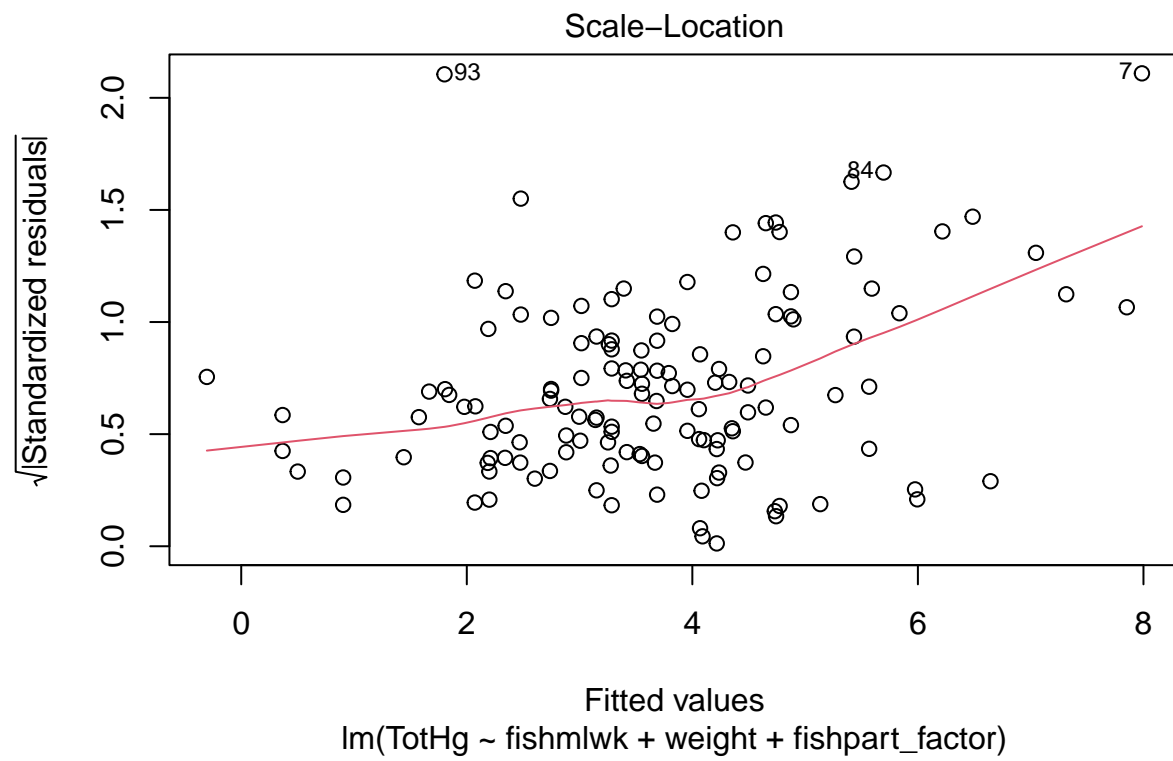
```
##
## Call:
## lm(formula = TotHg ~ fishmlwk + weight + fishpart_factor, data = fish.remove)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.8667 -1.1706 -0.0931  0.6257 10.1226
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -8.49716     2.35016   -3.616  0.00043 ***
## fishmlwk         0.13599     0.04074    3.338  0.00111 **
## weight          0.13430     0.03171    4.235 4.32e-05 ***
## fishpart_factor1 2.21262     0.94061    2.352  0.02018 *
## fishpart_factor2 1.42901     0.82407    1.734  0.08531 .
## fishpart_factor3 1.14095     1.15375    0.989  0.32458
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.324 on 128 degrees of freedom
## Multiple R-squared:  0.2927, Adjusted R-squared:  0.2651
## F-statistic: 10.59 on 5 and 128 DF, p-value: 1.572e-08
```

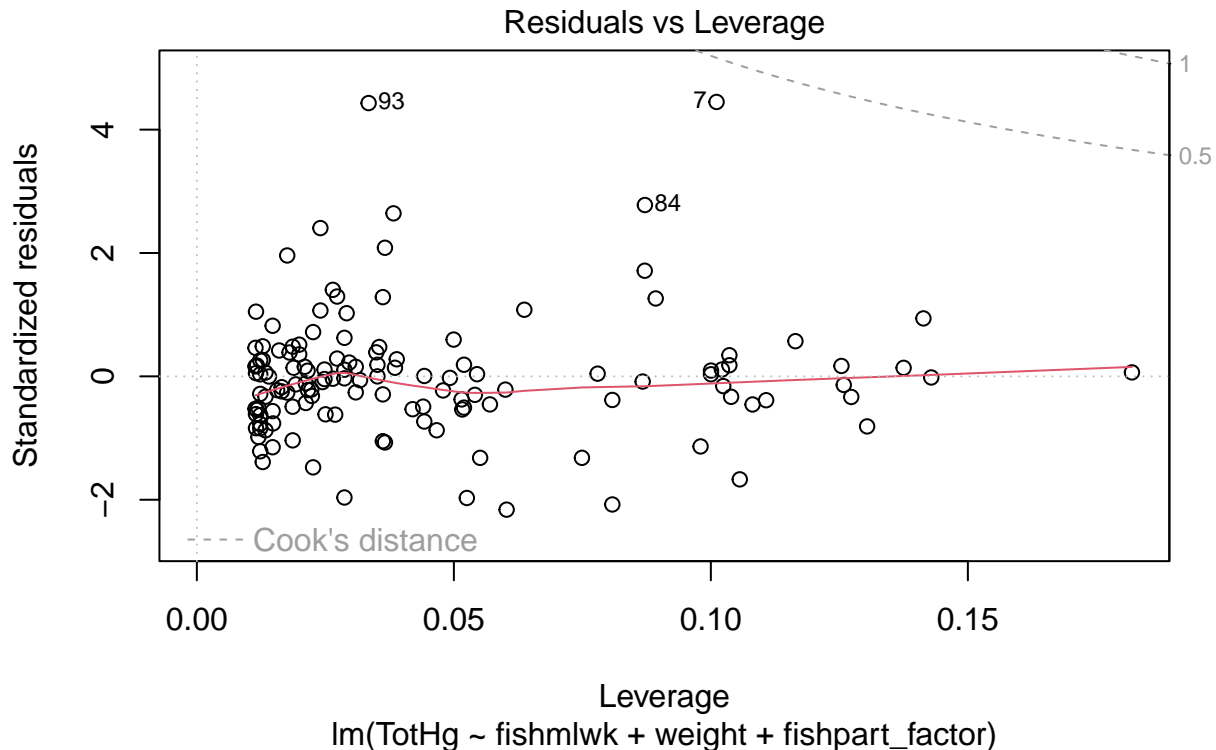
Finally, display the diagnostic plots of the re-fitted model.

```
# Diagnostic plots
plot(fish.reg.remove)
```









Have a look at both the model coefficients and the diagnostic plots and compare them to the coefficients and plots of the original model in Question 4. Once you've done that, answer the questions below:

- A) Look at the p-values associated with fishmlwk, weight, fishpart_factor1, fishpart_factor2, and fishpart_factor3 in both models. Using an alpha level of 0.05, would your conclusion about any of the coefficients be different for any of them? That is, would your conclusion switch from *reject* -> *fail to reject* or from *fail to reject* -> *reject* for any of the coefficients?

Your answer here (yes/no): Yes, the conclusion for fishpart_factor3 would change.

- B) If you answered 'yes' to the previous question, which coefficient/s would your conclusion change between models and how would the conclusion change? Be sure to name the coefficient explicitly (i.e., fishmlwk/weight/fishpart_factor1/fishpart_factor2/fishpart_factor3) and state how the decisions would differ (reject in first model -> fail to reject in second model or from fail to reject in first model -> reject in second model)

Your answer here: The conclusion for fishpart_factor3 would change from reject in the original model to fail to reject in the re-fitted model.

Question 6 - 10 points

There are several reasons to transform variables, one of which we will explore in this question. If the diagnostic plots indicate that the residuals are not normally distributed across the range of fitted values, one can apply a nonlinear transformation to the outcome variable to change the shape of the distribution

of the residuals. A common method for doing this is the Box-Cox transformation. Dr. Cathy Durso offered the following explanation of this approach:

“The Box-Cox transformations are a parametrized family of power transformations designed to be applied to the outcome variable to improve the Normality of residuals of a linear model. For $\lambda \neq 0$, the transformation maps y to $\frac{y^\lambda - 1}{\lambda}$ while for $\lambda = 0$, the transformation maps y to $\ln y$.

For each value of λ in the range of the argument “lambda”, the “boxcox” function in the “MASS” package fits the linear model it is given as an argument but with the Box-Cox transformation applied to the outcome variable, assumed to be positive. The function “boxcox” computes the log likelihood of the residuals under the assumption of Normality. This is plotted against the λ ’s and the λ ’s and the corresponding log likelihoods are returned. In typical use, a value of λ close to maximizing the log likelihood is chosen and regression performed with this transformation applied to the outcome variable.”

In this problem, you will walk through the steps of conducting a Box-Cox transformation.

Fitting the base model You start the process by fitting the model and examining the diagnostic plots to determine if there is non-normality in the model residuals. This time, the model contains fishmlwk, weight, fishpart_factor, age, and height.

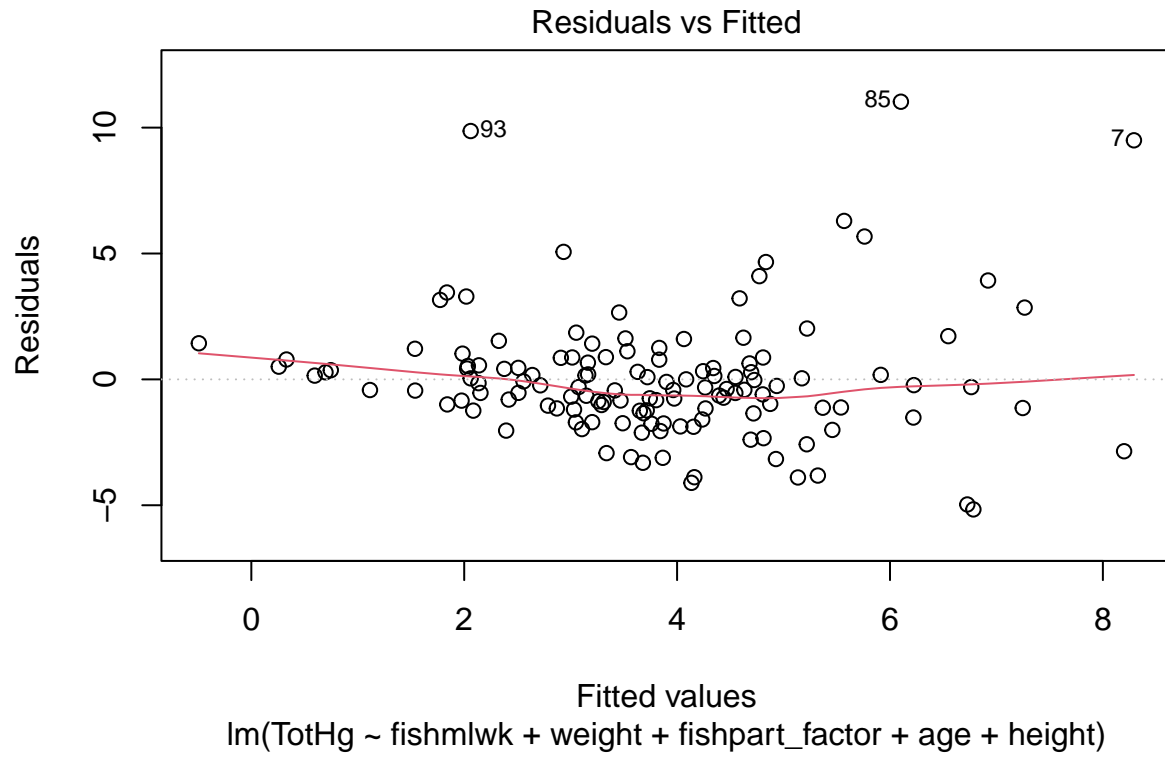
Run the code chunk below and examine the output. You do not need to modify anything (except maybe the very first line; see the comment). Once you’ve done that, continue onto the next section.

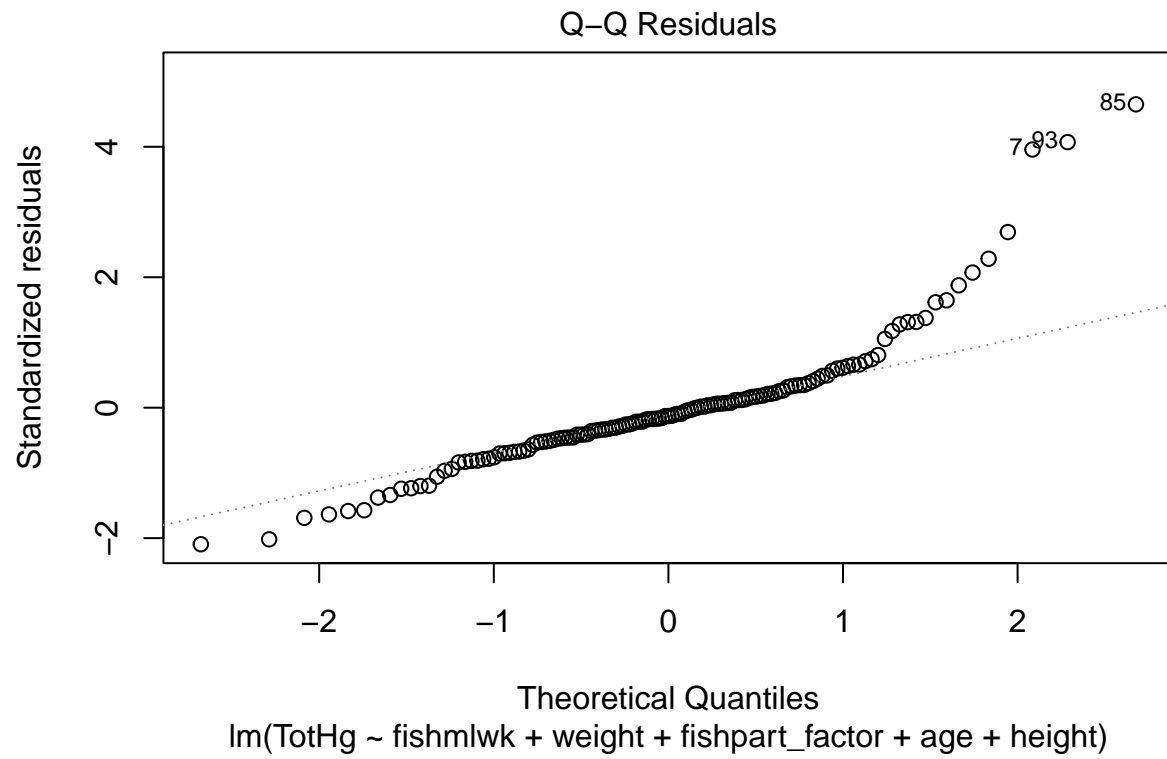
```
fish.demo <- read.csv("fishermen_mercury.csv", header=TRUE, sep=",")
fish.demo$fishpart_factor <- as.factor(fish$fishpart)
fish.reg.demo = lm(TotHg ~ fishmlwk + weight + fishpart_factor + age + height, data=fish.demo)
summary(fish.reg.demo)
```

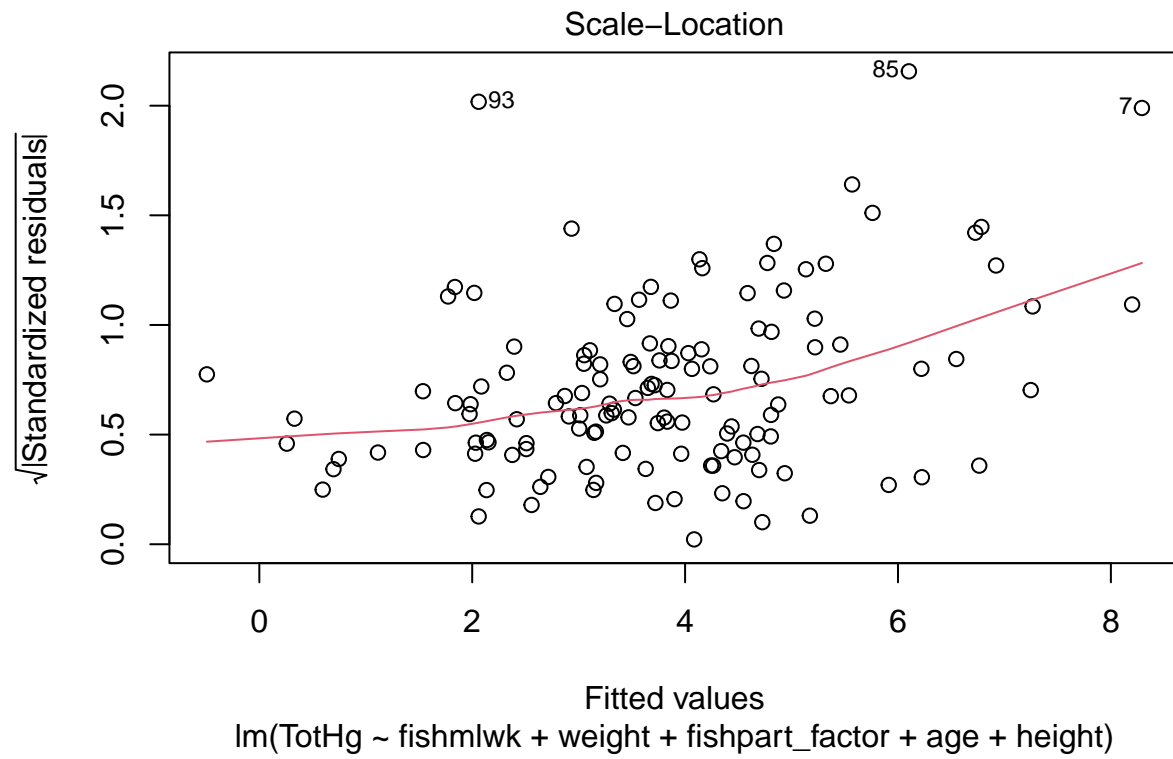
```
##
## Call:
## lm(formula = TotHg ~ fishmlwk + weight + fishpart_factor + age +
##     height, data = fish.demo)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -5.1645 -1.2147 -0.3064  0.7235 11.0284
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -15.85401     5.95019  -2.664  0.00871 **
## fishmlwk         0.12113     0.04555   2.659  0.00883 **
## weight         0.14693     0.03574   4.112 7.01e-05 ***
## fishpart_factor1  2.05185     1.04238   1.968  0.05120 .
## fishpart_factor2  1.47024     0.90256   1.629  0.10580
## fishpart_factor3  2.40495     1.23425   1.949  0.05356 .
## age             0.01657     0.02916   0.568  0.57075
## height         0.03432     0.03416   1.005  0.31701
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.544 on 127 degrees of freedom
## Multiple R-squared:  0.2894, Adjusted R-squared:  0.2503
## F-statistic:  7.39 on 7 and 127 DF, p-value: 1.938e-07
```

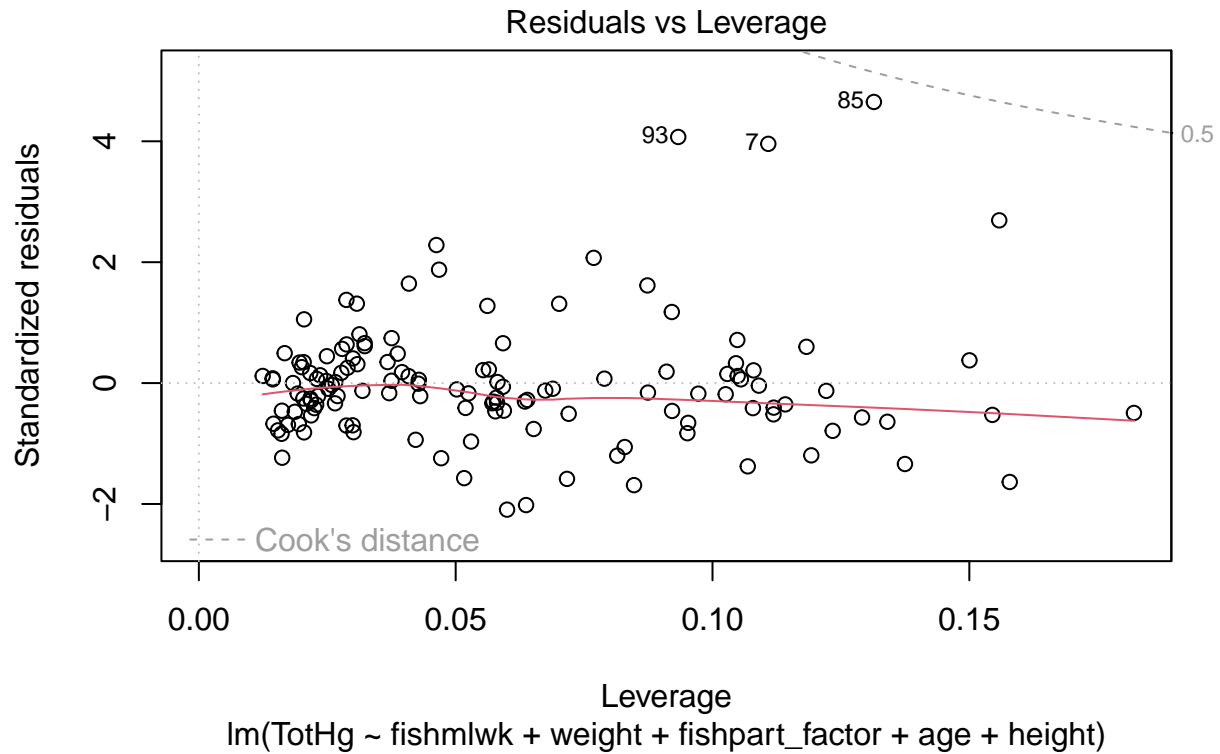


```
plot(fish.reg.demo)
```



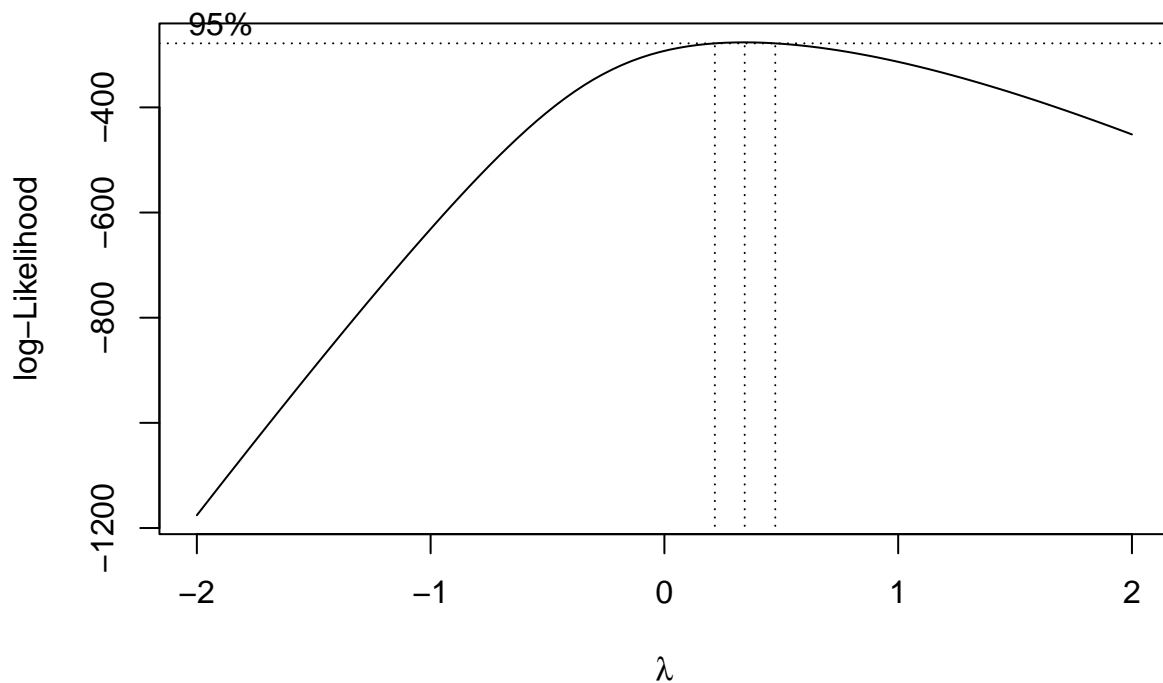






Apply the `boxcox()` function from the MASS package (loaded at the beginning of this document) to your fitted model. Run the code below and examine what it produces before moving on to the next part. You do not need to modify anything in this code chunk.

```
# Apply Box-Cox transformation
lambda <- boxcox(fish.reg.demo)
```



```
lambda.best <- lambda[[1]][which(lambda[[2]] == max(lambda[[2]]))]
```

Obtain the λ corresponding to the maximum profile log likelihood The code below pulls the lambda for which the log likelihood is maximized. You do not need to modify anything in this chunk. Run this code and examine what it produces, then answer the two questions before moving onto the next section.

```
ll.best<-which(lambda[[2]]==max(lambda[[2]]))
ll.best
```

```
## [1] 59
```

```
lambda.best<-lambda[[1]][ll.best]
lambda.best
```

```
## [1] 0.3434343
```

Question A: What is the *number* of the lambda that corresponds to the maximum profile likelihood?

Your answer here: Using the Box-Cox plot, we determined that the number of the lambda which maximizes the profile log likelihood is 59.

Question B: What is the *value* of the lambda that corresponds to the maximum profile likelihood?

Your answer here: The value of lambda, based on the Box-Cox transformation and the log-likelihood maximization, is 0.3434.

Apply the transformation to the output variable and re-fit the model. Now that you have the value of the lambda that corresponds to the maximum profile likelihood, you can now apply the Box-Cox transformation to your model.

Transforming the outcome variable

First, use `lambda.best` to transform the outcome variable. Recall from Dr. Durso's overview that the formula is $\frac{y^\lambda - 1}{\lambda}$. Complete the line in the code chunk to compute and save the transformed variable as a new variable ("TotHg.BC") in the fish data set, then answer the question below to check that you applied the transformation correctly.

```
# Apply transformation
fish.demo$TotHg.BC <- ((fish.demo$TotHg^lambda.best) - 1) / lambda.best
head(fish.demo)
```

```
##   fisherman age restime height weight fishmlwk fishpart   MeHg TotHg
## 1         1  45      6    175    70      14         2  4.011  4.484
## 2         1  38     13    173    73       7         1  4.026  4.789
## 3         1  24      2    168    66       7         2  3.578  3.856
## 4         1  41      2    183    80       7         1 10.988 11.435
## 5         1  43     11    175    78      21         1 10.520 10.849
## 6         1  58      2    176    75      21         1  6.169  6.457
##   fishpart_factor TotHg.BC
## 1                2 1.963078
## 2                1 2.074504
## 3                2 1.716901
## 4                1 3.811666
## 5                1 3.691286
## 6                1 2.613444
```

Question C: Look at the first observation in the fish.demo data set. The TotHg value for this observation is 4.484. What is the value of TotHg.BC (i.e., the transformed value) for this same observation?

Your answer here: the transformed value (TotHg.BC) for the first observation is ~1.963

Re-fitting the regression model using the transformed outcome

Be sure to have completed the previous section before starting this one.

Next, re-fit the regression model that was originally fitted. Keep the predictors the same - fishmlwk, weight, fishpart_factor, age, and height - but use the transformed outcome (TotHg.BC) instead of the original outcome (TotHg). Display the output for this model.

```
# Refit with transformed outcome
fish.reg.BC <- lm(TotHg.BC ~ fishmlwk + weight + fishpart_factor + age + height, data=fish.demo)
summary(fish.reg.BC)
```

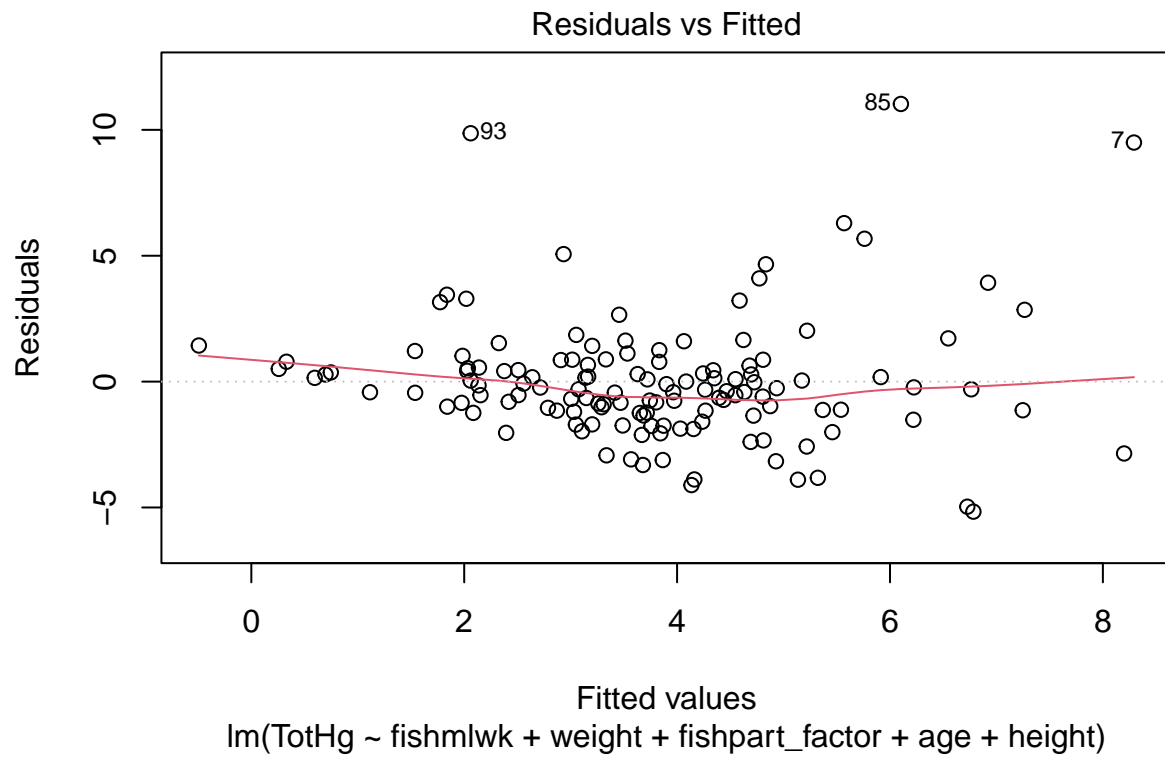
```
##
## Call:
## lm(formula = TotHg.BC ~ fishmlwk + weight + fishpart_factor +
```

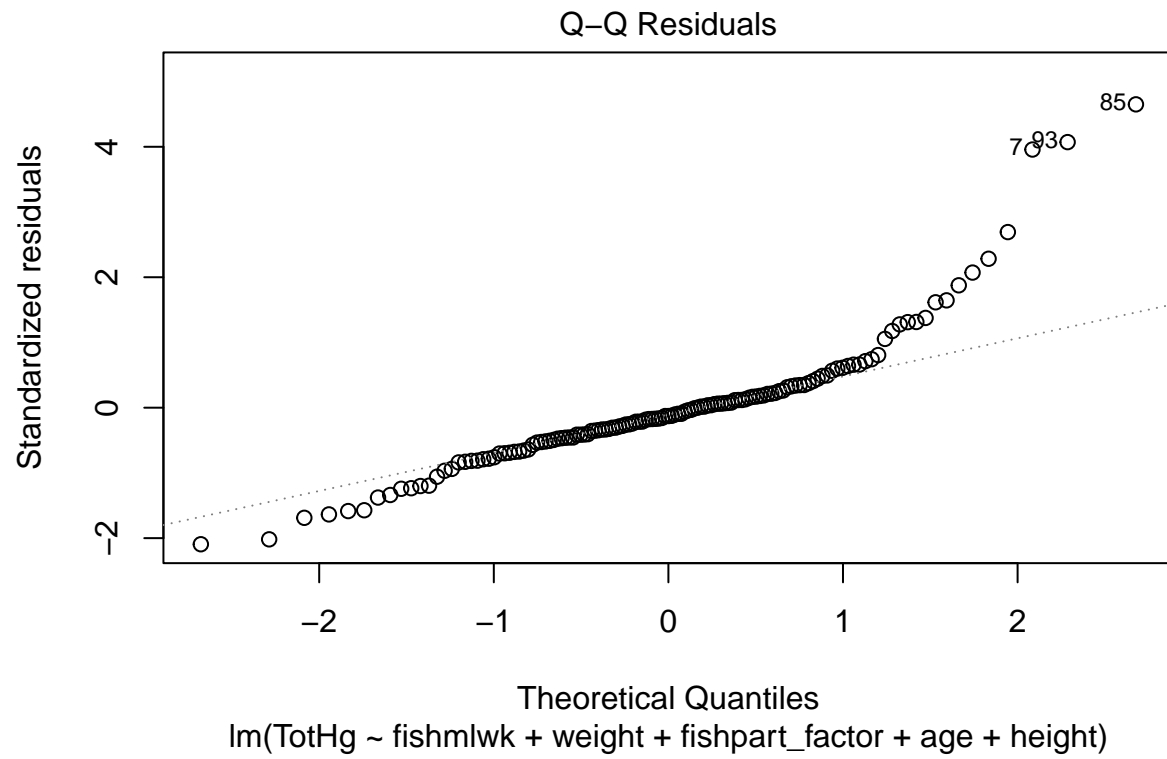
```
##      age + height, data = fish.demo)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.4276 -0.4156  0.0023  0.4627  3.0815
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -5.877885    2.292472  -2.564  0.01151 *
## fishmlwk       0.027332    0.017548   1.558  0.12184
## weight        0.056810    0.013769   4.126 6.63e-05 ***
## fishpart_factor1 1.345115    0.401604   3.349  0.00107 **
## fishpart_factor2 1.125900    0.347738   3.238  0.00154 **
## fishpart_factor3 1.444877    0.475530   3.038  0.00289 **
## age           0.003221    0.011233   0.287  0.77477
## height        0.009951    0.013163   0.756  0.45103
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9803 on 127 degrees of freedom
## Multiple R-squared:  0.3012, Adjusted R-squared:  0.2627
## F-statistic:  7.82 on 7 and 127 DF,  p-value: 7.369e-08
```

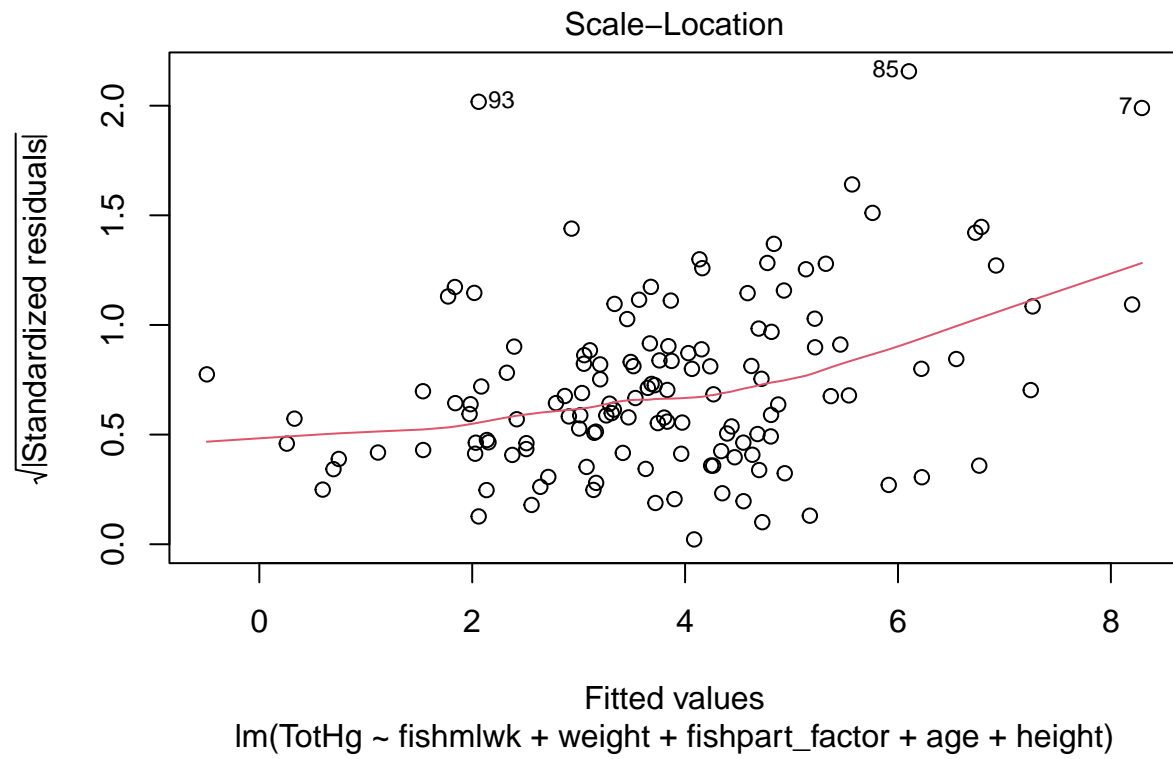
Display diagnostic plots for model with transformed outcome

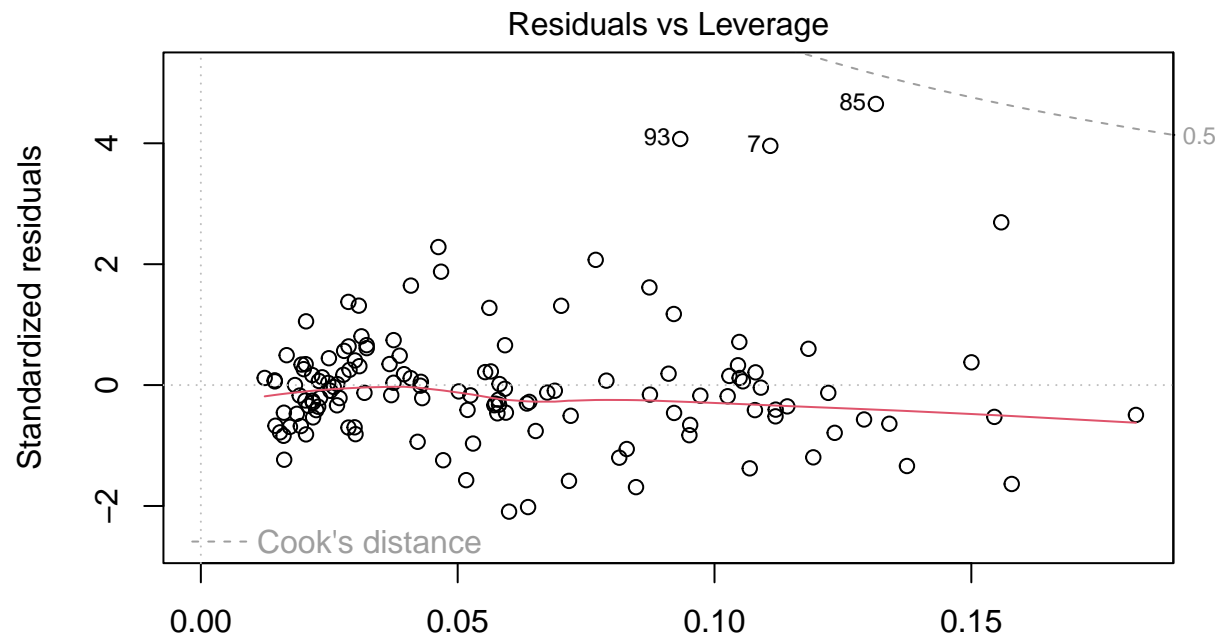
For your convenience, I've included code to produce the plots from the original model. Run this chunk (no modifications needed) and continue.

```
# Generate diagnostic plots for transformed model
plot(fish.reg.demo)
```





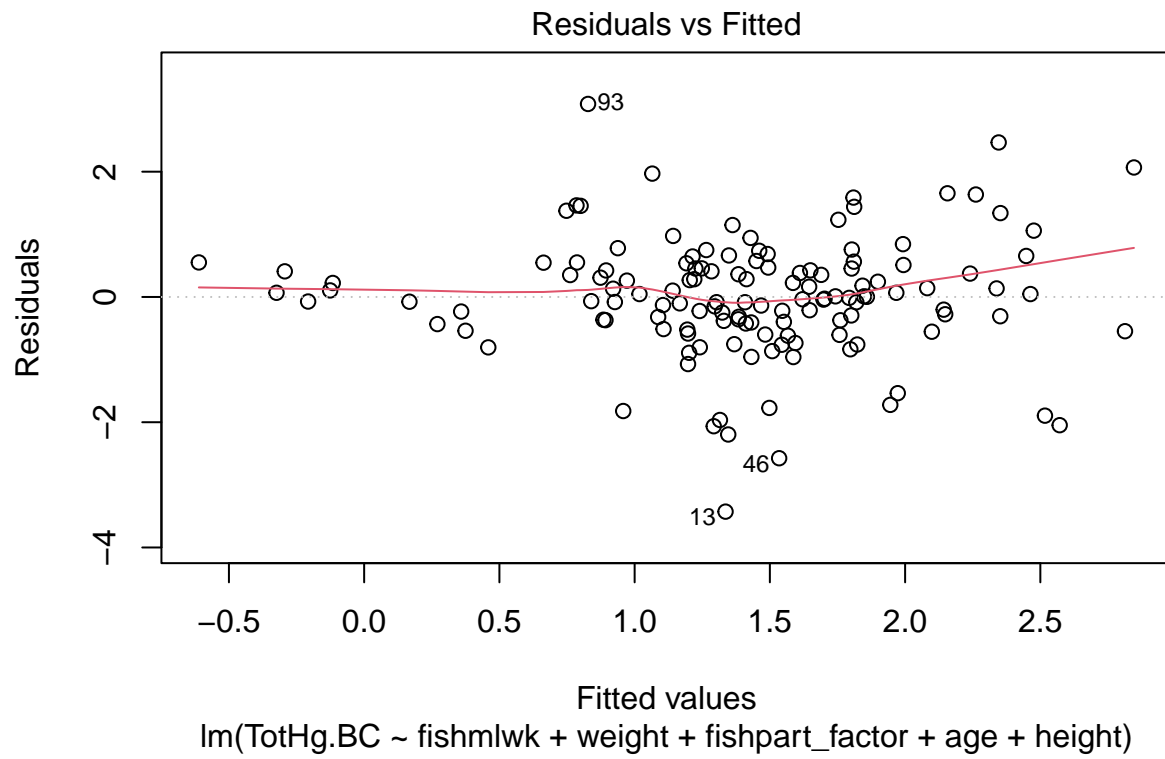


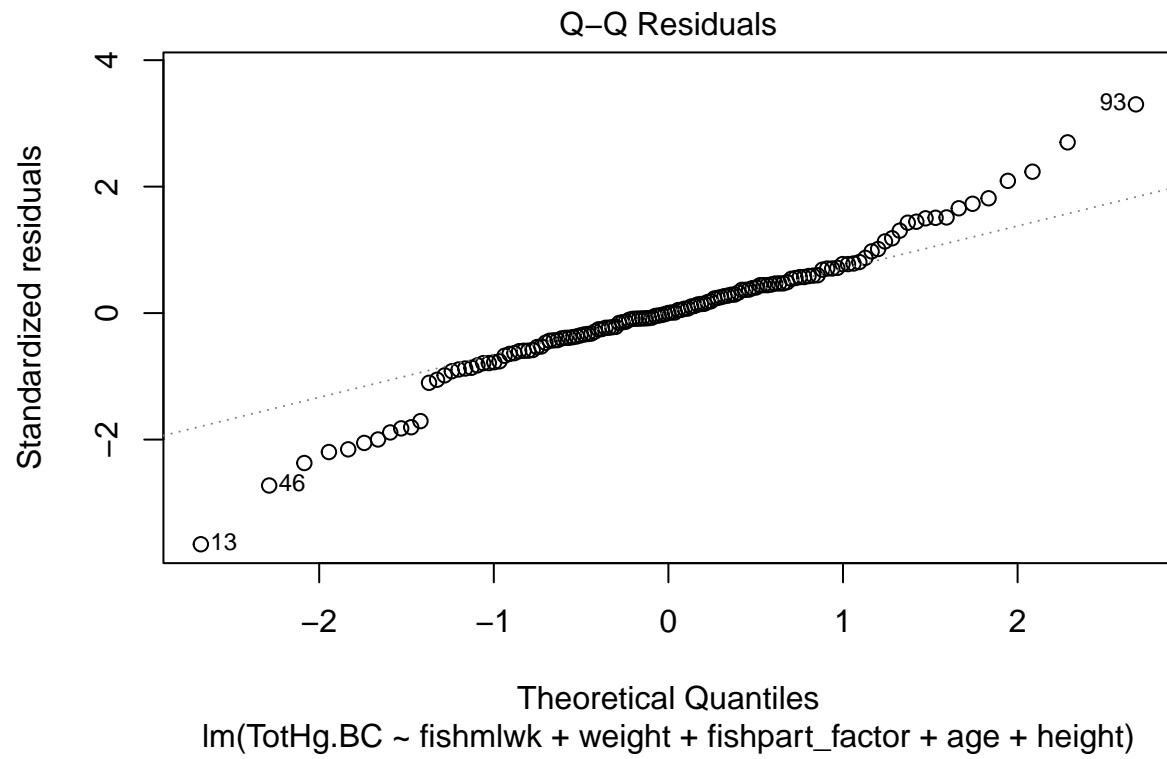


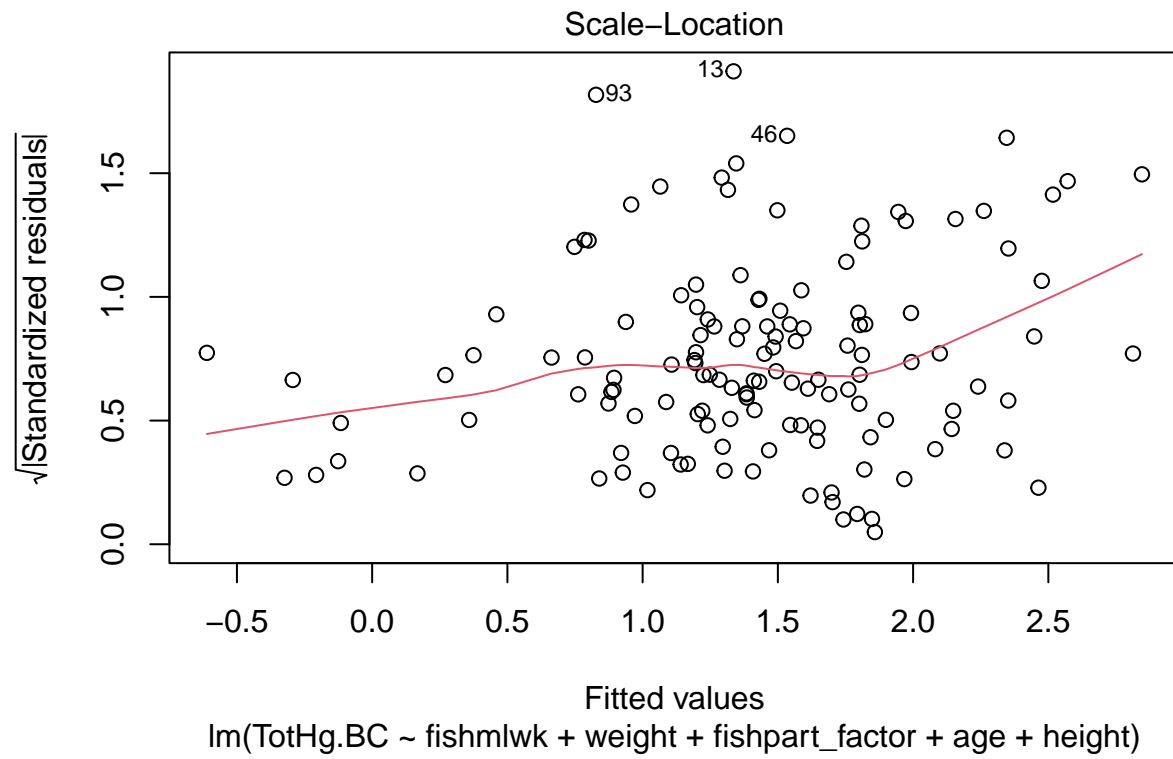
lm(TotHg ~ fishmlwk + weight + fishpart_factor + age + height)

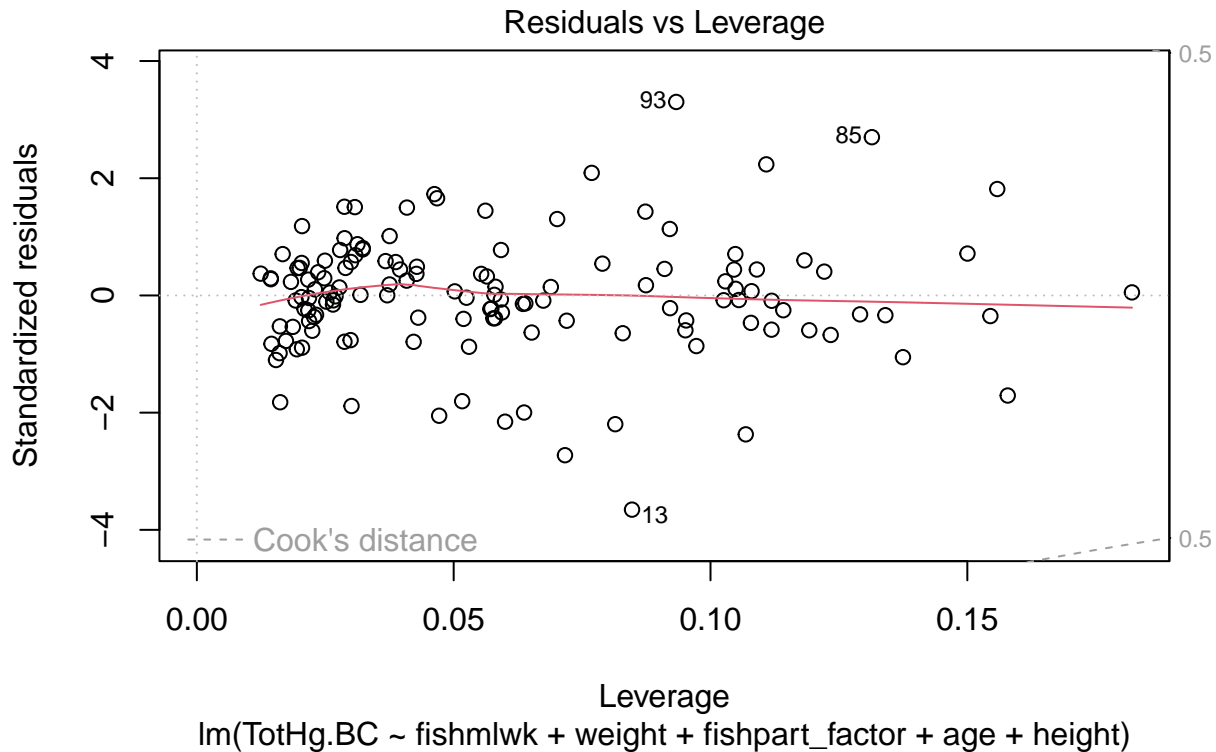
Now, generate the diagnostic plots of the re-fitted model that used the transformed outcome.

```
# Generate diagnostic plots for transformed model
plot(fish.reg.BC)
```









Look at both of the QQ plots and answer the two questions below.

Question D: In the regression model using the original outcome variable, was the most concerning side of the QQ plot the left side or the right side?

Your answer here (left or right): Right side. The points on the right side of the QQ plot deviate significantly from the theoretical line, indicating potential issues with extreme values on the right tail of the distribution.

Question E: Is the QQ plot of the model that used the transformed outcome different than the one from the model with the original outcome?

Your answer here (yes or no): Yes. The QQ plot for the transformed outcome shows better alignment with the theoretical line compared to the original model, suggesting that the transformation improved the normality of the residuals.

End.