Proposed Solutions

Mwavu Kennedy

Task 1

Test for normality of the data first; Use shapiro-wilk normality test.

- H_0: The data are normally distributed
- H_1: The data are not normally distributed

Shapiro-wilk normality test for male plasma ferretin concentration:

```
male_normality <- with(athlete_data, shapiro.test(x = Ferr[Sex == "male"]))
male_normality$p.value
```

```
## [1] 2.560565e-05
```

Shapiro-wilk normality test for female plasma ferretin concentration:

```
female_normality <- with(athlete_data, shapiro.test(x = Ferr[Sex == "female"]))
female_normality$p.value
```

```
## [1] 2.929953e-06
```

The p-values for both males and males are less than the significance level 0.05 implying that the distributions of the data are significantly different from the normal distribution.

This implies we can't use the unpaired two-samples t-test. We'll instead use the non-parametric unpaired two-samples wilcoxon test:

```
ferritin_diff <- wilcox.test(
  formula = Ferr ~ Sex, data = athlete_data, exact = FALSE
)

ferritin_diff
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Ferr by Sex
## W = 2604, p-value = 1.877e-09
## alternative hypothesis: true location shift is not equal to 0
```

Since the p-value is less than the significance level `alpha = 0.05`, we can conclude that males' median plasma ferritin concentration is significantly different from females' median plasma ferritin concentration with a p-value of 1.8769595×10^{-9}

Task 2

Randomly divide the dataset into two sets, training ($n1 = 141$) and testing ($n2 = 61$):

```
# training indices:
set.seed(91)
n1 <- sample(x = nrow(athlete_data), size = 141, replace = FALSE)

# training data:
training <- athlete_data[n1, ]

# testing data:
testing <- athlete_data[-n1, ]
```

a) Equation of a regression model

```
# all column names:
colnms <- colnames(training)

# predictor variables:
predictors <- colnms[!colnms %in% c("Sport", "Ferr")]

# Equation:
frmla <- reformulate(predictors, response = "Ferr")
frmla
```

```
## Ferr ~ Sex + LBM + RCC + WCC + Hc + Hg + BMI + SSF + X.Bfat
```

b) Fit the model

```
full_model <- lm(formula = frmla, data = training)
summary(full_model)
```

```
##
## Call:
## lm(formula = frmla, data = training)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -67.821 -25.611  -6.845  20.603 131.239
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  25.5310    60.9688   0.419  0.67608
## Sexmale      75.0916    16.5940   4.525 1.34e-05 ***
## LBM          -2.0800     0.6528  -3.186  0.00180 **
## RCC           0.2336    19.3996   0.012  0.99041
## WCC           3.1718     1.9932   1.591  0.11395
## Hc           -2.8174     4.0680  -0.693  0.48980
## Hg            3.5004     9.0376   0.387  0.69915
## BMI           8.2488     2.5758   3.202  0.00171 **
## SSF          -0.1513     0.4827  -0.314  0.75440
## X.Bfat        1.0657     2.9408   0.362  0.71767
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.68 on 131 degrees of freedom
## Multiple R-squared:  0.3152, Adjusted R-squared:  0.2682
## F-statistic: 6.701 on 9 and 131 DF,  p-value: 7.345e-08
```

The only significant predictors are: - Sex - LBM - BMI

Fit a model with only the significant predictors

```
smaller_model <- lm(
  formula = Ferr ~ Sex + LBM + BMI, data = training
)
summary(smaller_model)
```

```
##
## Call:
## lm(formula = Ferr ~ Sex + LBM + BMI, data = training)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -73.632 -25.039  -6.685  20.931 126.054
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -21.241     28.373  -0.749 0.455370
## Sexmale       67.768     11.537   5.874 3.07e-08 ***
## LBM           -2.322      0.613  -3.788 0.000227 ***
## BMI            9.258      1.934   4.786 4.35e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.33 on 137 degrees of freedom
## Multiple R-squared:  0.2965, Adjusted R-squared:  0.2811
## F-statistic: 19.25 on 3 and 137 DF,  p-value: 1.791e-10
```

Is a the full model better than the smaller model?

Compare the two models using anova test:

```
comparison <- anova(smaller_model, full_model)
comparison
```

```
## Analysis of Variance Table
##
## Model 1: Ferr ~ Sex + LBM + BMI
## Model 2: Ferr ~ Sex + LBM + RCC + WCC + Hc + Hg + BMI + SSF + X.Bfat
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1    137 211954
## 2    131 206310  6    5644.4 0.5973  0.732
```

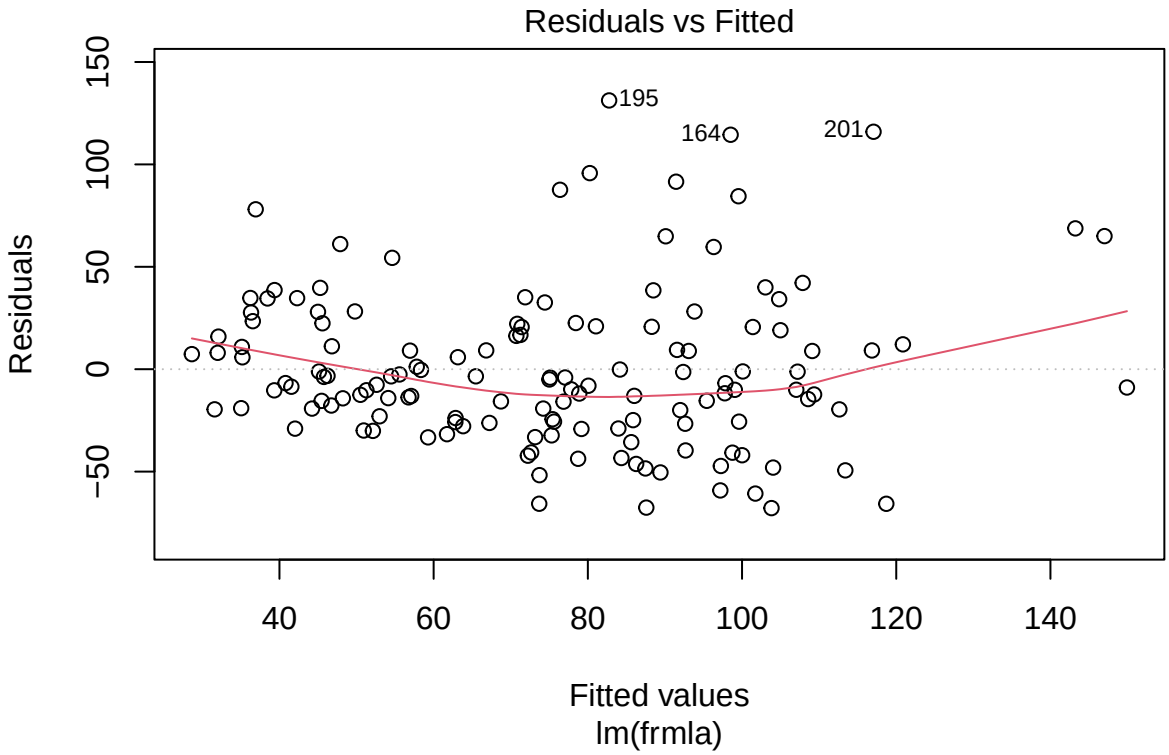
The p-value = 0.732 is not sufficiently low (less than 0.05) implying that the complex model (`full_model`) is not significantly better than the simpler model (`smaller_model`). We should favor the smaller model. In other words, the full model is not better than the smaller model.

c) Check the linear regression assumptions for the model fitted in part (b)

1. Linearity of the data

The linearity assumption can be checked by inspecting the Residuals vs Fitted plot:

```
plot(full_model, 1)
```

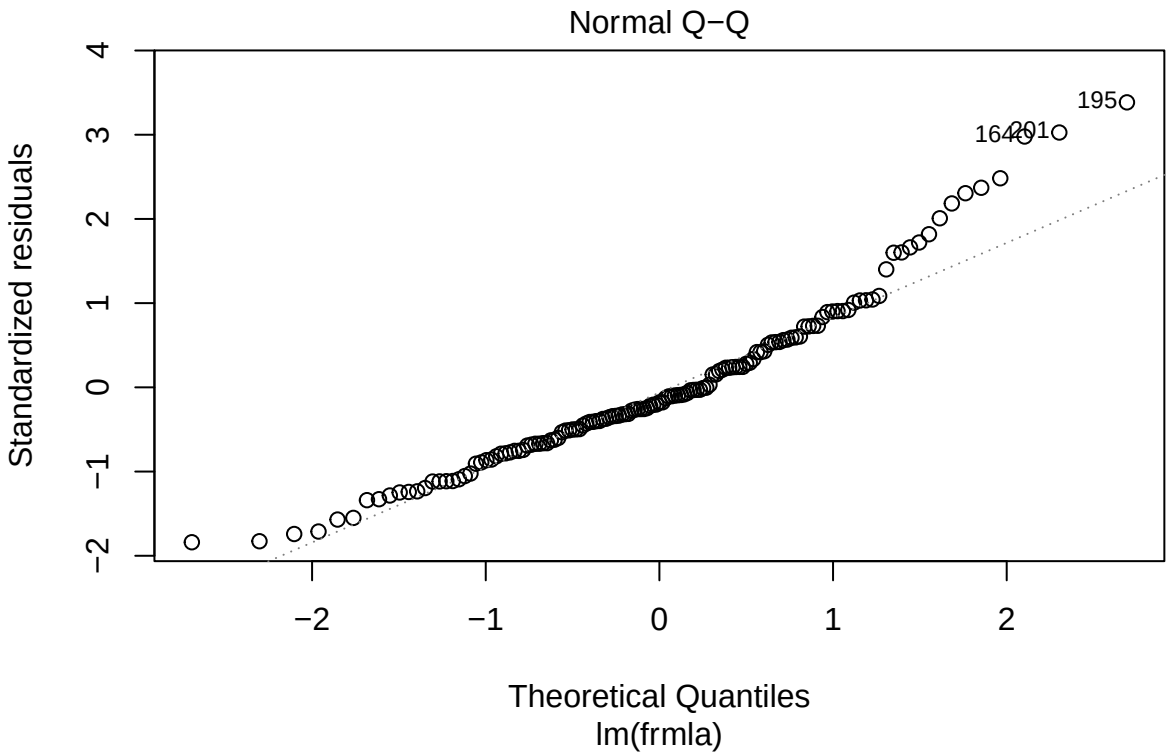


There is a pattern in the residual plot meaning we can't assume a linear relationship between the predictors and outcome variable

2. Normality of residuals

The QQ plot of residuals can be used to visually check the normality assumption. The normal probability plot of residuals should approximately follow a straight line.

```
plot(full_model, 2)
```



Most of the points do not fall approximately along the reference line, so we can assume non-normality.

shapiro test for normality:

```
shapiro.test(full_model$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  full_model$residuals
## W = 0.94519, p-value = 2.381e-05
```

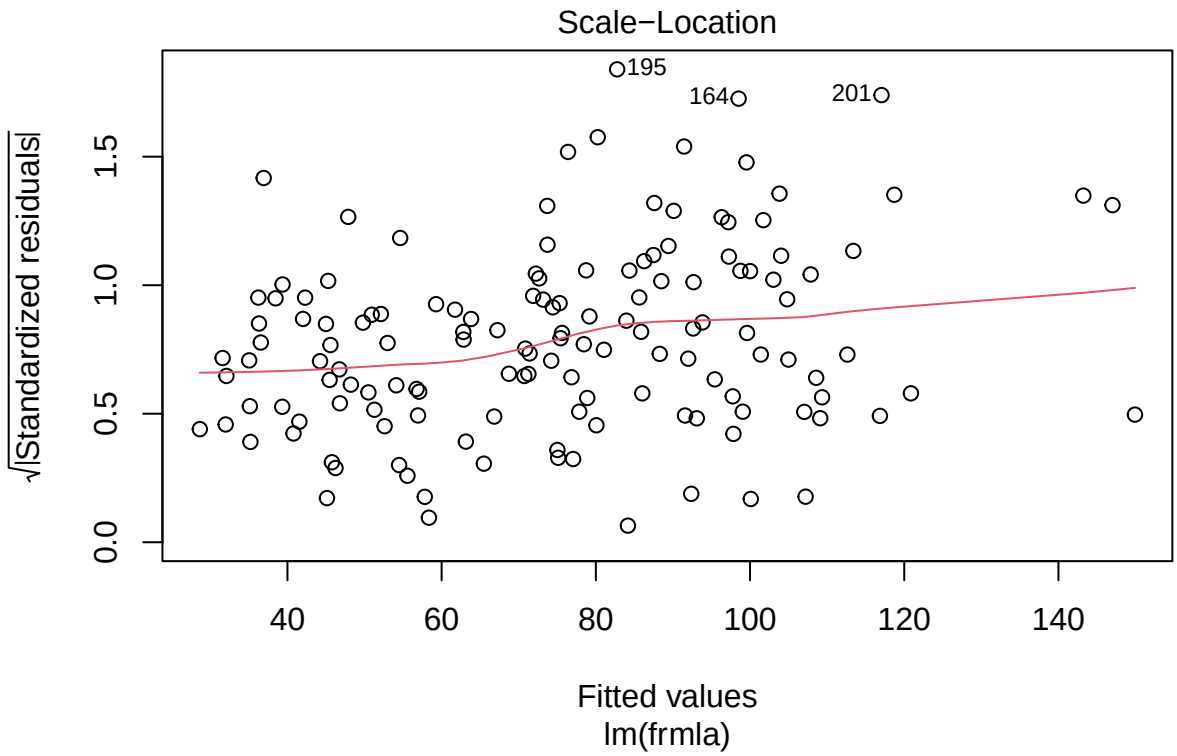
The p-value is less than 0.05, confirming that the residuals are not normally distributed.

3. Homoscedasticity

The residuals are assumed to have a constant variance.

We'll use the scale-location plot to check the homogeneity of variance of the residuals. Horizontal line with equally spread points is a good indication of homoscedasticity.

```
plot(full_model, 3)
```



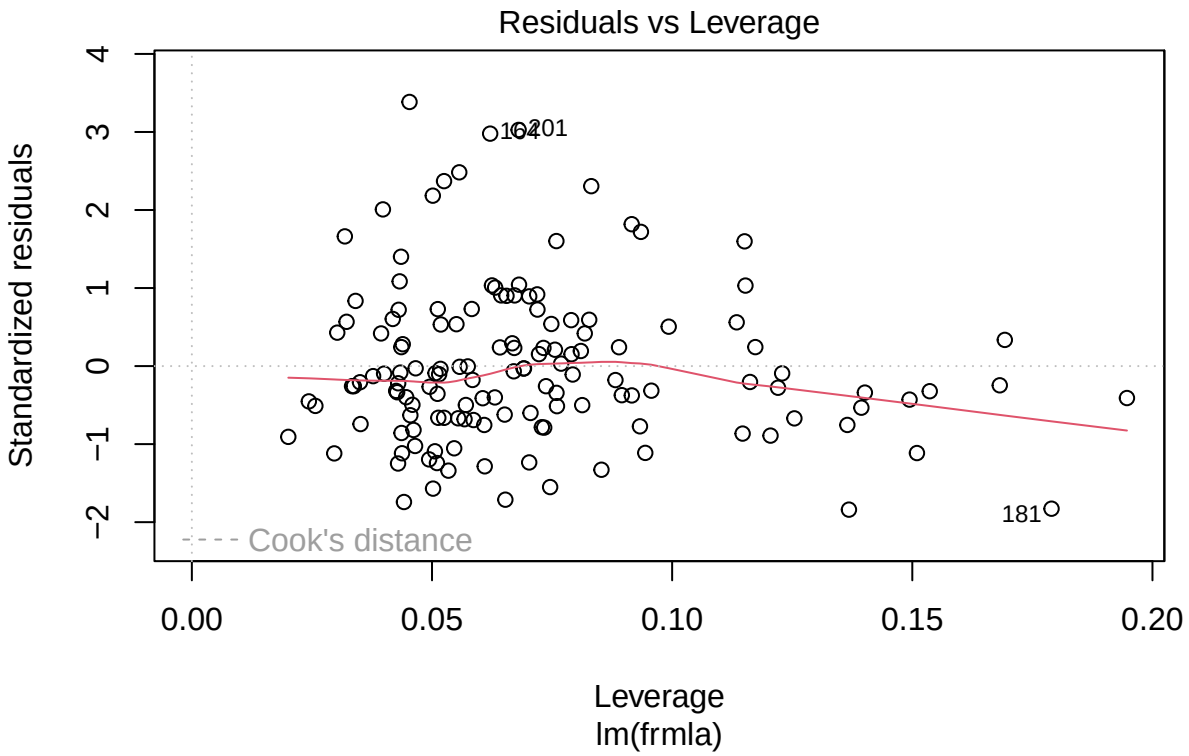
That is not the case in our example, where we have a heteroscedasticity problem.

4. Residuals vs Leverage

Used to identify influential cases, that is extreme values that might influence the regression results when included or excluded from the analysis.

Outliers and high leverage points can be identified by inspecting the Residuals vs Leverage plot:

```
plot(full_model, 5)
```



From the plot, there are no outliers which exceed 3 standard deviations, which is good.

However, there are high leverage points ie. there are some data points have a leverage statistic above $2\frac{(p+1)}{n} = 2\frac{(9+1)}{141} = 0.141844$.