MAT3375_Assignment-3

Rahul Atre

2023-10-31

Exercise 5.15

Suppose that we want to fit the no-intercept model $y = \beta x + \epsilon$ using weighted least squares. Assume that the observations are uncorrelated but have unequal variances.

a. Find a general formula for the weighted least-squares estimator of β.

Ans: In order to derive the weighted least-squares estimator of β, we need to minimize the weighted sum of squares, which is defined as follows:

$$S(\beta) = \sum_{i=1}^{n} w_i (y_i - \beta x_i)^2$$

where w_i is the weight assigned to the i^{th} observation. Next, we can take the derivative of the weighted sum of squares with respect to β and set it equal to 0:

$$\frac{\partial}{\partial \beta} \sum_{i=1}^{n} w_i (y_i - \beta x_i)^2 = 0$$

After expanding and simplifying the above equation, we obtain:

$$\sum_{i=1}^{n} w_i (y_i - \beta x_i)(-x_i) = 0$$

Multiplying and rearranging:

$$\sum_{i=1}^{n} w_i x_i y_i - \beta \sum_{i=1}^{n} w_i x_i^2 = 0$$

Lastly, solving for the β estimate gives us the below equation, which is the weighted least-squares estimator for β:

$$\hat{\beta} = \frac{\sum_{i=1}^{n} w_i x_i y_i}{\sum_{i=1}^{n} w_i x_i^2}$$

b. What is the variance of the weighted least-squares estimator?

Using the defined weighted estimate of β in part (a), we can solve the variance of the weighted-least squares of β as follows:

$$Var(\hat{\beta}) = (\frac{1}{\sum_{i=1}^{n} w_i x_i^2})^2 \sum_{i=1}^{n} w_i^2 x_i^2 Var(y_i)$$
$$= (\frac{1}{\sum_{i=1}^{n} w_i x_i^2})^2 \sum_{i=1}^{n} w_i^2 x_i^2 \frac{\sigma^2}{w_i}$$

After simplifying the above equation, the variance of the weighted least-squares estimator is:

$$= \frac{\sigma^2}{\sum_{i=1}^{n} w_i x_i^2}$$

c. Suppose that $Var(y_i) = cx_i$, that is, variance of y_i is proportional to the corresponding x_i. Using the results of parts a and b, find the weighted least-squares estimator of β and the variance of this estimator.

From the above statement, we know that the variance is proportional to the x observations. In weighted least-squares estimate, the weights are given by $w_i = \frac{1}{var(y_i)}$. Logically, more weight is given to observations with smaller variance (i.e. decrease weight of more variable obs. and increase weight of less variable obs.). In simpler terms, the weight of an observation is reciprocal to the variance of the response.

So, $w_i = \frac{1}{x_i}$. Since we are estimating the beta parameter, the c can be omitted.

$$\hat{\beta} = \frac{\sum_{i=1}^{n} \frac{1}{x_i} x_i y_i}{\sum_{i=1}^{n} \frac{1}{x_i} x_i^2}$$
$$= \frac{\sum_{i=1}^{n} y_i}{\sum_{i=1}^{n} x_i}$$

$$Var(\hat{\beta}) = \frac{\sigma^2}{\sum_{i=1}^{n} \frac{1}{x_i} x_i^2}$$
$$= \frac{\sigma^2}{\sum_{i=1}^{n} x_i}$$

d. Suppose that $Var(y_i) = cx_i^2$, that is, variance of y_i is proportional to the square of the corresponding x_i. Using the results of parts a and b, find the weighted least-squares estimator of β and the variance of this estimator.

Similar to part (c), $w_i = \frac{1}{var(y_i)} = \frac{1}{x_i^2}$. So:

$$\hat{\beta} = \frac{\sum_{i=1}^{n} \frac{1}{x_i^2} x_i y_i}{\sum_{i=1}^{n} \frac{1}{x_i^2} x_i^2}$$
$$= \frac{\sum_{i=1}^{n} \frac{y_i}{x_i}}{\sum_{i=1}^{n} 1}$$
$$= \frac{1}{n} \sum_{i=1}^{n} \frac{y_i}{x_i}$$

$$
\begin{aligned}
Var(\hat{\beta}) &= \frac{\sigma^2}{\sum_{i=1}^{n} \frac{1}{x_i^2} x_i^2} \\
&= \frac{\sigma^2}{\sum_{i=1}^{n} 1} \\
&= \frac{\sigma^2}{n}
\end{aligned}
$$

Exercise 5.17

Consider the model

$$y = X\beta + \epsilon$$

where $E(\epsilon) = 0$, and $Var(\epsilon) = \sigma^2 V$. Assume that V is known but not σ^2. Show that

$$(y'V^{-1}y - y'V^{-1}X(X'V^{-1}X)^{-1}X'V^{-1}y/(n-p)$$

is an unbiased estimator of σ^2.

Recall: An estimator is unbiased if its expected value is the value of the parameter.

First, we'll look at the numerator of the equation:

$$y'V^{-1}y - y'V^{-1}X(X'V^{-1}X)^{-1}X'V^{-1}y$$

Notice that it can be simplified in quadratic form:

$$y'[V^{-1} - V^{-1}X(X'V^{-1}X)^{-1}X'V^{-1}]y$$

Now, we can let the midde section of the quadratic form be a matrix A, where $A = V^{-1} - V^{-1}X(X'V^{-1}X)^{-1}X'V^{-1}$. Let A be a k x k matrix of constants, y be a k x 1 random vector with mean μ and a non-singular variance-covariance matrix V.

According to definition C.2.3 (Expectations) No. 4, $E(y'Ay) = trace(AV) + \mu' A\mu$. Also, from the information given in the question, $\mu = E(y) = 0$.

So, $E(y'Ay) = trace(AV) + \mu' A\mu = trace(AV)$.

Recall: C.2 - Background from the Theory of Linear Models:

- 6. Idempotent Matrix Let A be a k × k matrix. A is called idempotent if A = AA
- 11. Rank of an Idempotent Matrix Let A be an idempotent matrix. The rank of A is its trace.

From the above definition, it is trivial that AV are idempotent, so therefore, the rank (n-p) is equal to the trace of AV. Therefore, $E(y'Ay) = (n-p)\sigma^2$.

Lastly, if we included both the numerator and denominator:

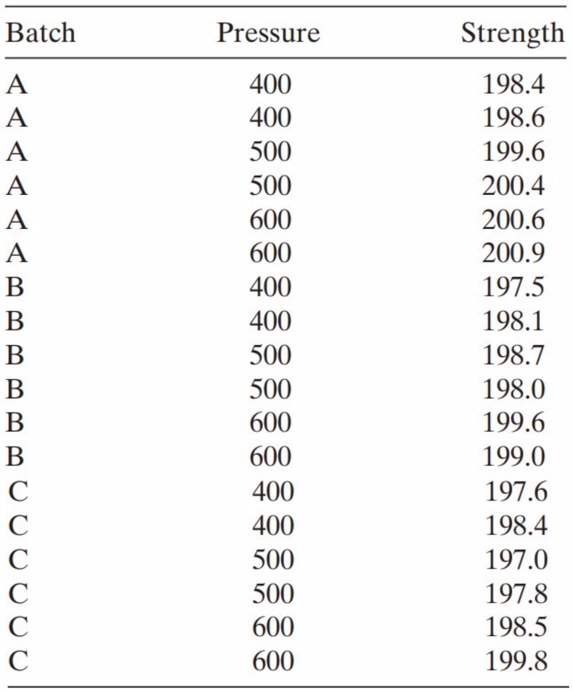
$$
\begin{aligned}
&E(\frac{y'V^{-1}y - y'V^{-1}X(X'V^{-1}X)^{-1}X'V^{-1}y}{(n-p)}) \\
&= E(\frac{y'Ay}{n-p}) \\
&= \frac{(n-p)\sigma^2}{n-p}
\end{aligned}
$$

$$= \sigma^2$$

Hence, we have proved that the given equation is an unbiased estimator of σ^2 since the expected value is the value of the parameter.

Exercise 5.23

A paper manufacturer studied the effect of three vat pressures on the strength of one of its products. Three batches of cellulose were selected at random from the inventory. The company made two production runs for each pressure setting from each batch. As a result, each batch produced a total of six production runs. The data follows. Perform the appropriate analysis.



Batch	Pressure	Strength
A	400	198.4
A	400	198.6
A	500	199.6
A	500	200.4
A	600	200.6
A	600	200.9
B	400	197.5
B	400	198.1
B	500	198.7
B	500	198.0
B	600	199.6
B	600	199.0
C	400	197.6
C	400	198.4
C	500	197.0
C	500	197.8
C	600	198.5
C	600	199.8

First, we must insert all the data into a single dataframe in R.

The independent variables (x) are Pressure and Batch, while the response (y) is strength.

First, let's fit a linear model for a single response variable, the Pressure. Then, we will add Batch to see if the model better fits the data.

```
library(lattice)
library(MPV)
```

```
## Loading required package: KernSmooth
```

```
## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009
```

```
## Loading required package: randomForest
```

```
## randomForest 4.7-1.1
## Type rfNews() to see new features/changes/bug fixes.
```

```r
data("p5.23")

batch_x = p5.23$Batch
pressure_x = p5.23$Pressure
strength_y = p5.23$Strength

lin_model1 = lm(strength_y ~ pressure_x) #Fitting a linear model for pressure
```

```r
summary(lin_model1)
```

```
##
## Call:
## lm(formula = strength_y ~ pressure_x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.80556 -0.58889  0.04444  0.56111  1.59444
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.947e+02  1.343e+00 145.023  < 2e-16 ***
## pressure_x  8.167e-03  2.650e-03   3.081  0.00715 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.9181 on 16 degrees of freedom
## Multiple R-squared:  0.3724, Adjusted R-squared:  0.3332
## F-statistic: 9.495 on 1 and 16 DF,  p-value: 0.007153
```

```r
anova(lin_model1)
```

```
## Analysis of Variance Table
##
## Response: strength_y
##            Df  Sum Sq Mean Sq F value   Pr(>F)
## pressure_x  1  8.0033  8.0033  9.4952 0.007153 **
## Residuals  16 13.4861  0.8429
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

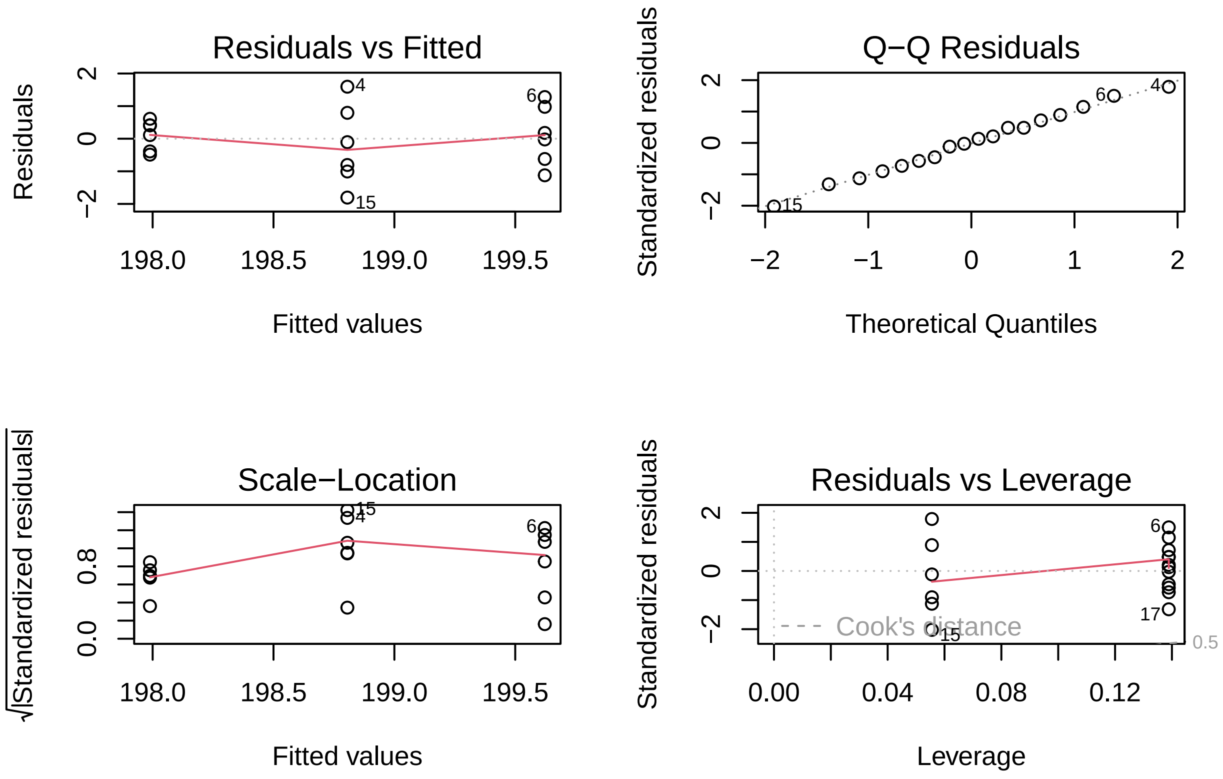
From the above summary() and anova() function call, we obtain $b_0 = 194.7$ and $b_1 = 0.008167$.

Therefore, the linear regression model will be $\hat{Y} = 194.7 + 0.008167X_1$, where X_1 rep. the pressure, and $\hat{y}$ rep. the strength. Additionally, the p-value for both the intercept and pressure coefficients are very small, which implies that these parameters are statistically significant.

The F-statistic is 9.495, which is quite large. The R^2 value is 0.3724, which is not a particularly high value for a given model (we would like it to be higher than 70%, in most cases). The adjusted R^2 is 0.3332, which means about 33.32% of the total variability in y is explained by this model. Thus, while the estimated beta parameter for pressure is significant, it does not explain the variability of strength very well.

In addition, we can examine the residuals to determine the quality of the model.

```
par(mfrow = c(2, 2))
plot(lin_model1)
```



From the QQ-plot, we can see that most of the residual points follow the line of best fit, so there is no problem with the normality assumption (Residuals are normally distributed). From the residuals vs. fitted, we can see that most of the points are close to the horizontal line res = 0. Since there is no clear pattern, it implies that the residuals are random and not correlated to the predicted response. Therefore, the quality of this model is satisfied.

Now, let us add the second predictor variable, Batch, to the model and determine if adding it helps fit the data better.

```
full_model = lm(strength_y ~ pressure_x + batch_x)

summary(full_model)
```

```
## 
## Call:
## lm(formula = strength_y ~ pressure_x + batch_x)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.18333 -0.37083 -0.05833  0.32500  1.03333 
## 
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) 195.666667   0.913148 214.277  < 2e-16 ***
## pressure_x    0.008167   0.001757   4.647 0.000377 ***
## batch_xB     -1.266667   0.351471  -3.604 0.002876 **
## batch_xC     -1.566667   0.351471  -4.457 0.000542 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.6088 on 14 degrees of freedom
## Multiple R-squared:  0.7586, Adjusted R-squared:  0.7068
## F-statistic: 14.66 on 3 and 14 DF,  p-value: 0.0001334
```

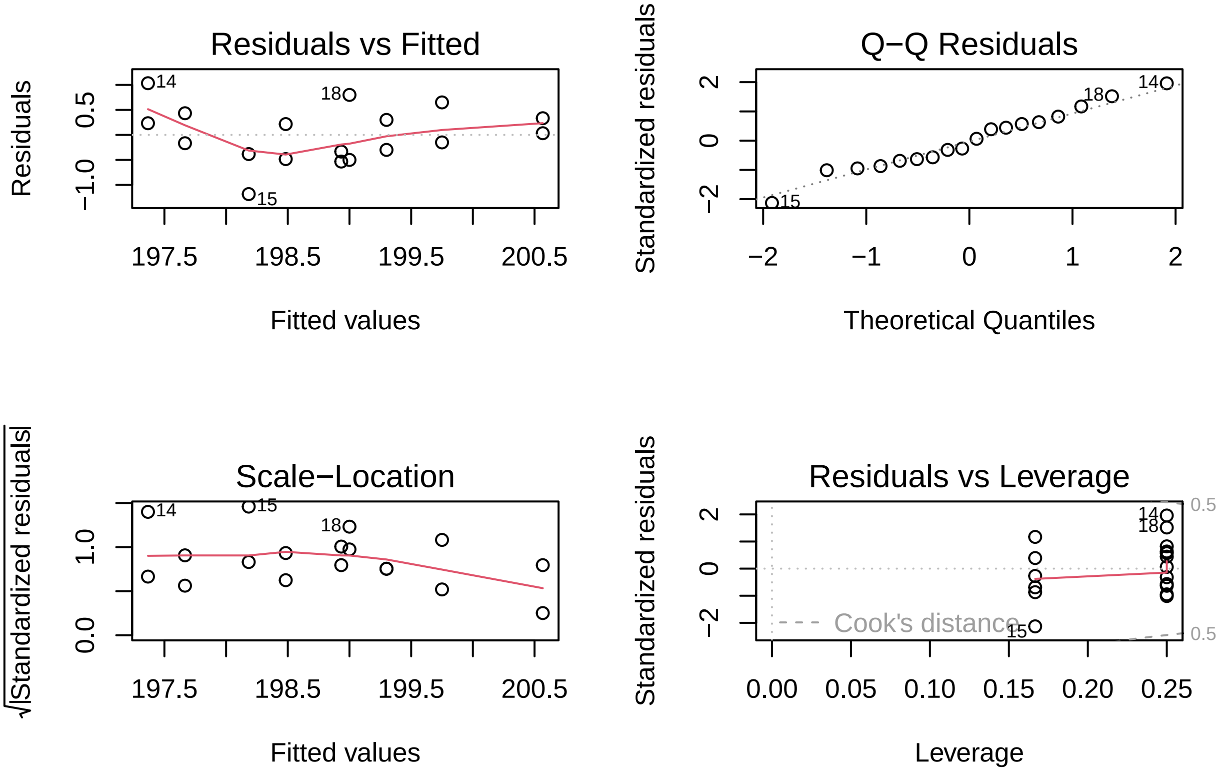
```r
anova(full_model)
```

```
## Analysis of Variance Table
##
## Response: strength_y
##            Df Sum Sq Mean Sq F value    Pr(>F)
## pressure_x  1 8.0033  8.0033  21.596 0.0003771 ***
## batch_x     2 8.2978  4.1489  11.195 0.0012473 **
## Residuals  14 5.1883  0.3706
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

From the above summary() and anova() function call, we have obtained $\hat{Y} = 195.666667 + 0.008167X_1 - 1.266667X_2 - 1.566667X_3$, where X_2 and X_3 represent the respective batches for B and C. The reason Batch A is not included in the equation is because out of the 3 batch categories, Batch A was used as a reference category, whereas the remaining were dummy variables.

From the analysis, we can see that all parameters are statistically significant and the F-value is quite large at 14.66. The R^2 value is a lot bigger now, at 0.7586, as well as the adjusted R^2 at 0.7068. So, 70.68% of the variability is explained by the model. Therefore, including the Batch predictor category dramatically improved the fitting of the model. Let us lastly examine the residuals to see the quality of the model/

```r
par(mfrow = c(2, 2))
plot(full_model)
```



Similar to before, the QQ-plot indicates that most of the residual points follow the line of best fit, so there is no problem with the normality assumption (Residuals are normally distributed). From the residuals vs. fitted, we can see that most of the points are close to the horizontal line res = 0. There is a slight u-shape in the middle, however it is negligible. Since there is no clear pattern, it implies that the residuals are random and not correlated to the predicted response. Therefore, the quality of this model is satisfied, and it can be concluded that both predictors are important to better fit the model.

Exercise 6.12

Table B.11 contains data on the quality of Pino Noir wine. Fit a regression model using clarity, aroma, body, flavor, and oakiness as the regressors. Investigate this model for influential observations and comment on your findings.

Let us first fit the model, then look at the summary/anova, then check the residuals to diagnose the overall goodness-of-fit and quality of the model, and lastly examine closely for influential observations.

```
library(MPV)
data("table.b11")

wine_model = lm(Quality ~ Clarity + Aroma + Body + Flavor + Oakiness, data = table.b11)

summary(wine_model)
```

```
##
## Call:
```

```
## lm(formula = Quality ~ Clarity + Aroma + Body + Flavor + Oakiness,
##     data = table.b11)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.85552 -0.57448 -0.07092  0.67275  1.68093
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.9969     2.2318   1.791 0.082775 .
## Clarity       2.3395     1.7348   1.349 0.186958
## Aroma         0.4826     0.2724   1.771 0.086058 .
## Body          0.2732     0.3326   0.821 0.417503
## Flavor        1.1683     0.3045   3.837 0.000552 ***
## Oakiness     -0.6840     0.2712  -2.522 0.016833 *
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 1.163 on 32 degrees of freedom
## Multiple R-squared:  0.7206, Adjusted R-squared:  0.6769
## F-statistic: 16.51 on 5 and 32 DF,  p-value: 4.703e-08
```

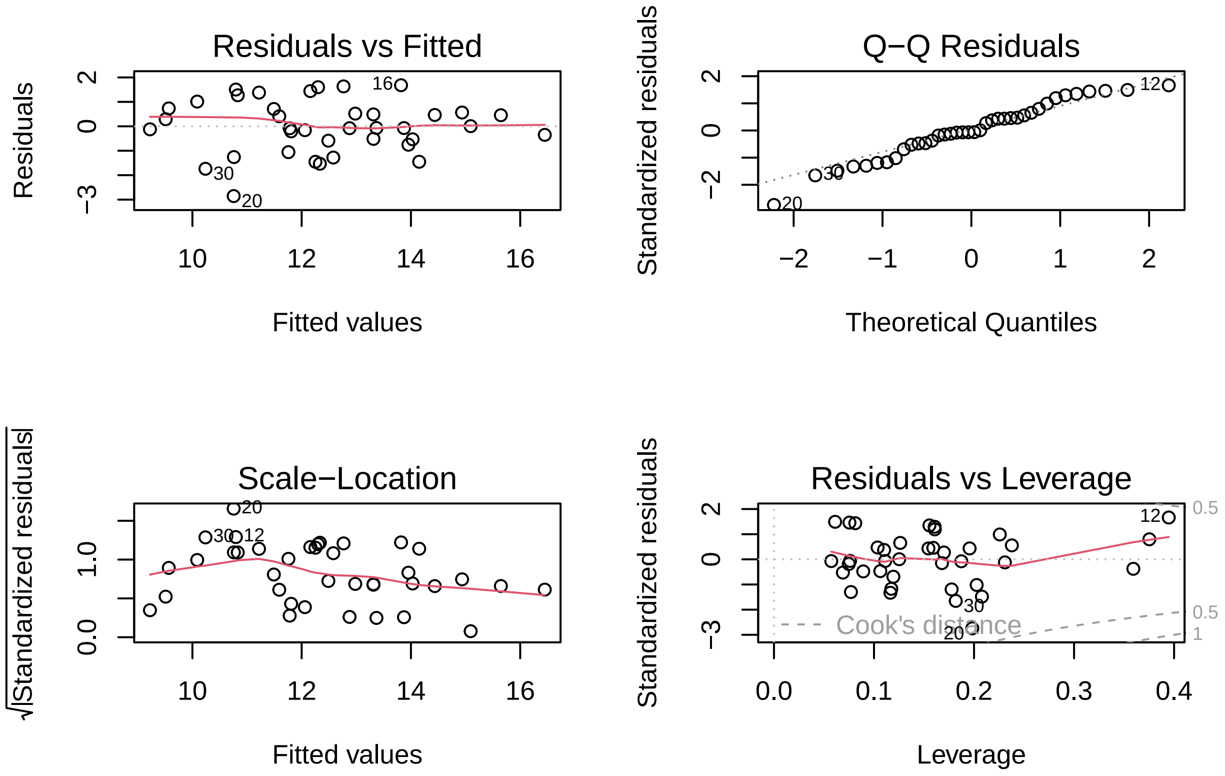
```r
anova(wine_model)
```

```
## Analysis of Variance Table
##
## Response: Quality
##           Df Sum Sq Mean Sq F value    Pr(>F)
## Clarity    1  0.125   0.125  0.0926 0.7628120
## Aroma      1 77.353  77.353 57.2351 1.286e-08 ***
## Body       1  6.414   6.414  4.7461 0.0368417 *
## Flavor     1 19.050  19.050 14.0953 0.0006946 ***
## Oakiness   1  8.598   8.598  6.3616 0.0168327 *
## Residuals 32 43.248   1.352
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

From the above function call, we have obtained the linear function $\hat{Y} = 3.9969 + 2.3395X_1 + 0.4826X_2 + 0.2732X_3 + 1.1683X_4 - 0.6840X_5$. The predictor for Flavor has an extremely low p-value of 0.000552, and Oakiness has a p-value of 0.016833. This indicates that Flavor is the most statistically significant parameter in explaining the response, followed by Oakiness. The remaining coefficients are not significant as they have a high p-value. In addition, the adjusted R^2 value is 0.6769, meaning that 67.69% of the variability is explained by the model, which is fairly decent (not the best however).

As for residuals:

```r
par(mfrow = c(2, 2))
plot(wine_model)
```



From the four plots:

Residuals vs Fitted: The residuals are randomly scattered around the horizontal line at zero, indicating that the linear regression model is a good fit for the data.

Q-Q Residuals: The residuals follow a normal distribution, indicating that the linear regression model is a good fit for the data.

Scale-Location: The residuals are homoscedastic (have equal variance) across the range of fitted values, indicating that the linear regression model is a good fit for the data.

Residuals vs Leverage: It is hard to tell if there are influential observations affecting the regression line. A closer examination needs to be done, using specific R commands for the influential observations.

In R, there are a few commands used to identify influential data points:

```
library(olsrr)
```

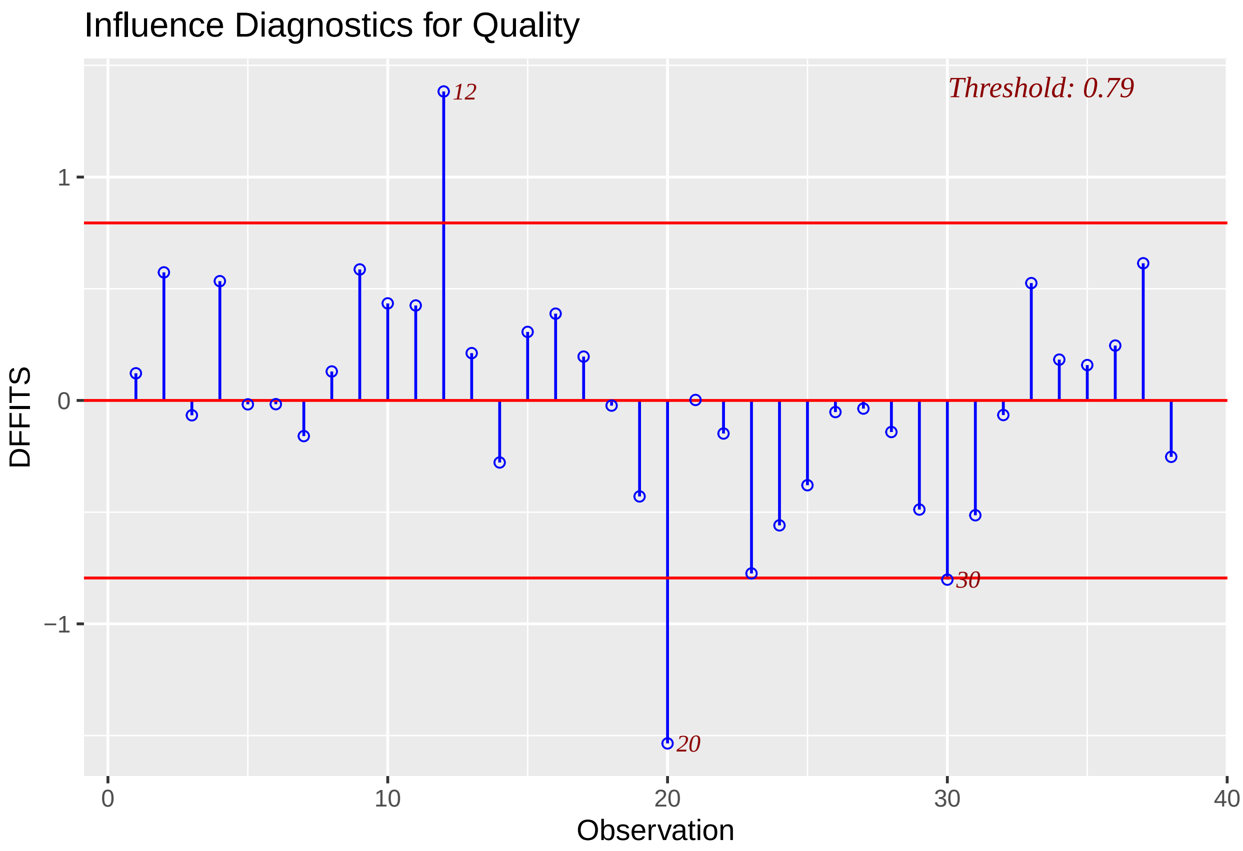
```
## Warning: package 'olsrr' was built under R version 4.3.2

##
## Attaching package: 'olsrr'

## The following object is masked from 'package:MPV':
##
##     cement

## The following object is masked from 'package:datasets':
##
##     rivers
```

```r
ols_plot_dffits(wine_model)
```

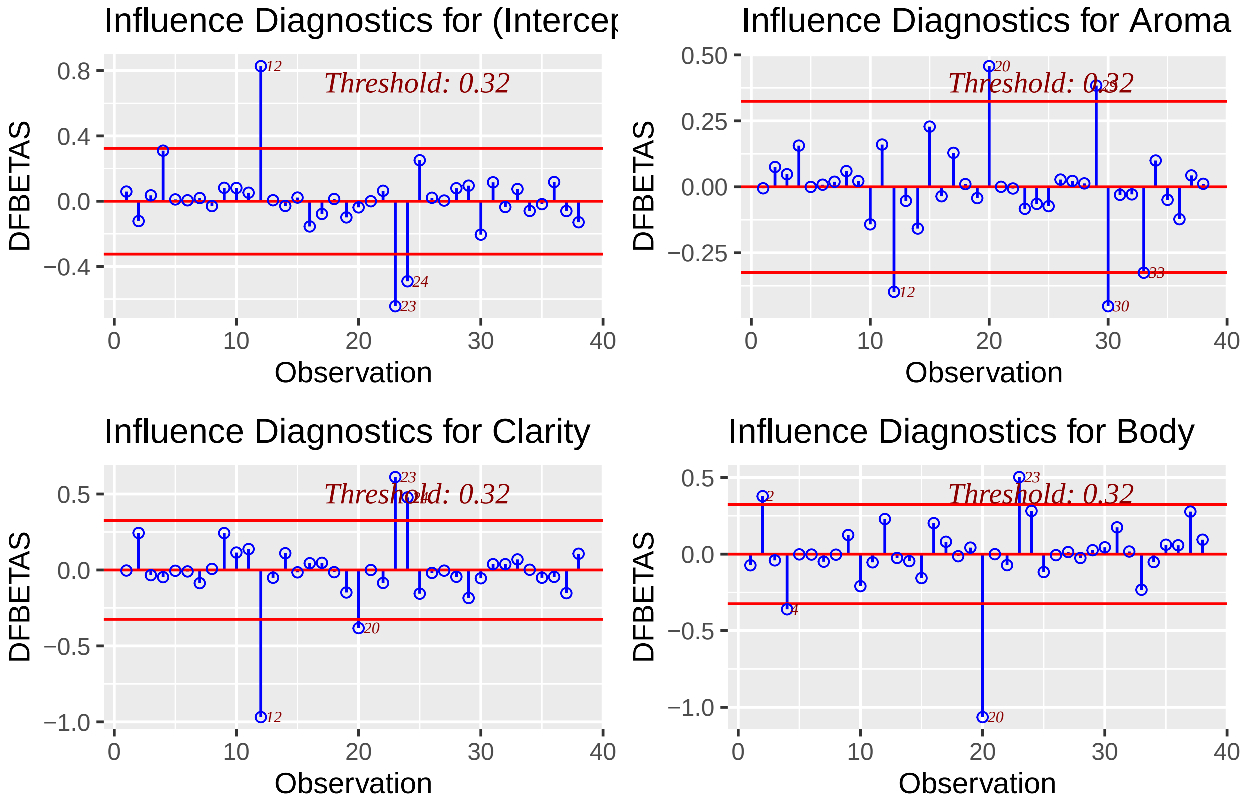


The following graph is the Influence Diagnostics for Quality, which assess the quality of any statistical model. The plot is used to identify any influential observations that may be affecting the model by quantifying the number of standard deviations that the fitted value changes when the ith data point is omitted.

As we can see, the 12th and 20th observations are deemed influential in the plot.

```r
ols_plot_dfbetas(wine_model)
```

page 1 of 2

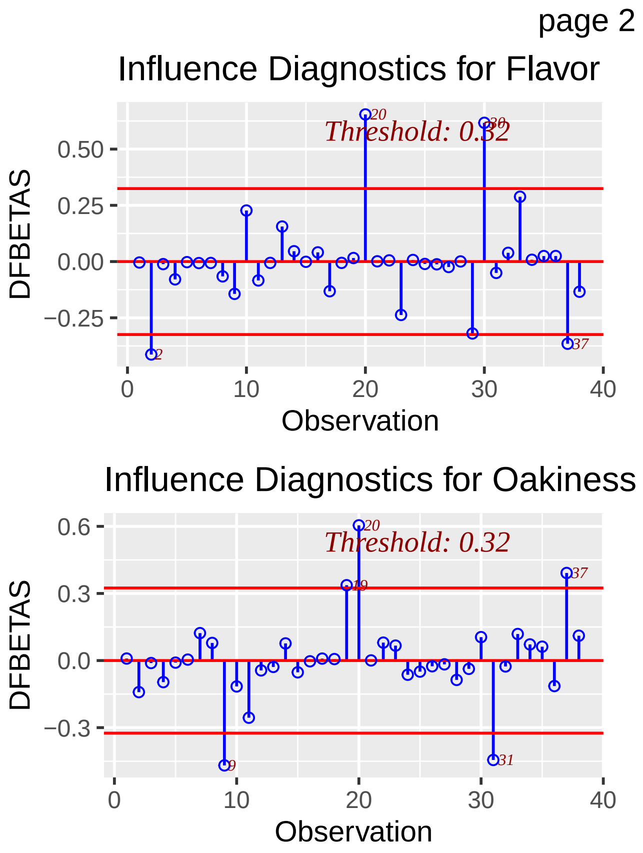


Influence Diagnostics for (Intercep

Influence Diagnostics for Aroma

Influence Diagnostics for Clarity

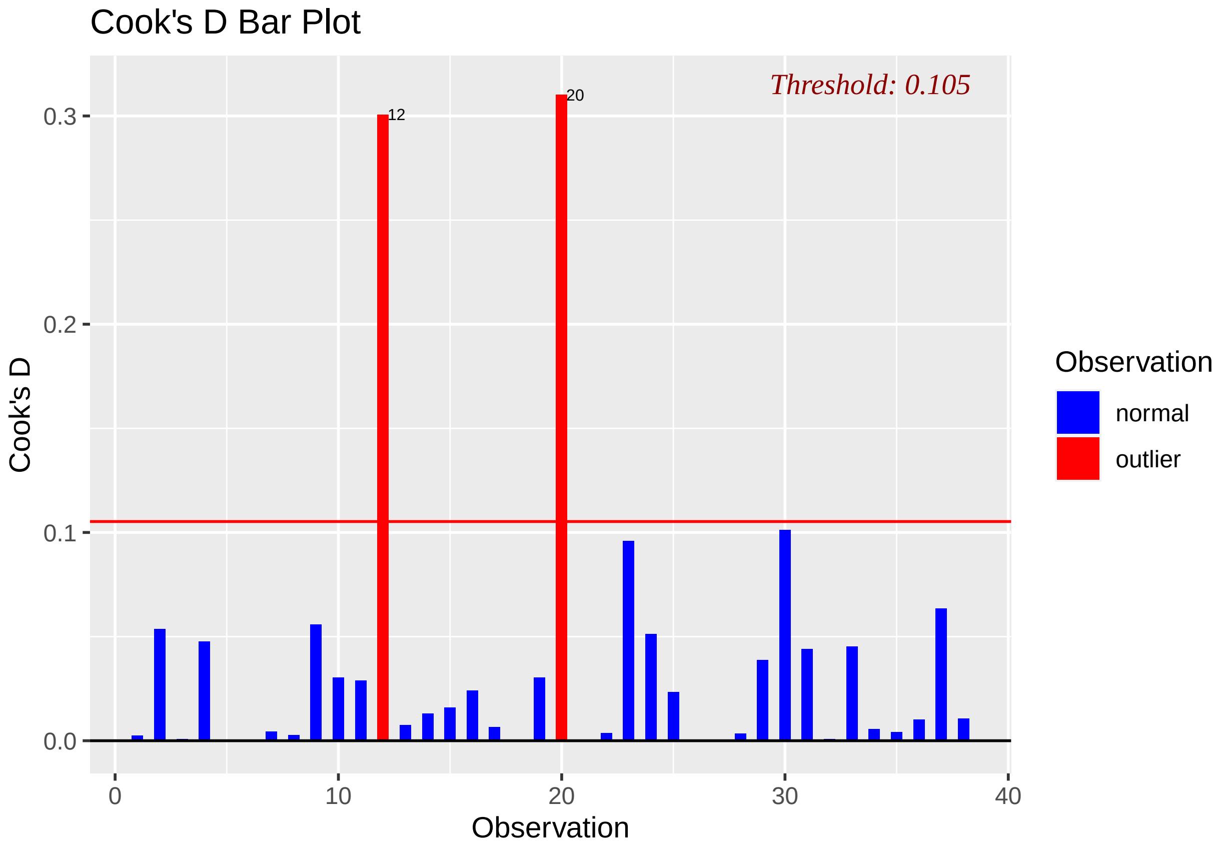
Influence Diagnostics for Body



The following function is used to detect influential observations using DFBETAs. It measures the difference in each parameter estimate with and without the influential point. For these parameters, the influential observations were:

β_0: 12, 23, 24 β_1: 12, 20, 23, 24 β_2: 12, 20, 29, 30, 33 β_3: 2, 4, 20, 23 β_4: 2, 20, 30, 37 β_5: 9, 19, 20, 31, 37

```
ols_plot_cooksd_bar(wine_model)
```



Lastly, the bar plot of cook's distance is used to detect observations that strongly influence fitted values of the model. In this case, those observations are 12 and 20.

In conclusion, the listed observations above are all influential. The most commonly seen influential observations were 12 and 20.