STAT 210
Applied Statistics and Data Analysis
Problem List 10
(Due on Week 11)

We will use the data in the file `iguanodon.csv` for the first two problems in the list. The data come from measurements taken in Jurassic Park and has information on 27 specimens of a type of dinosaur known as iguanodon. Read the data onto a data frame. There are nine variables in the set:

- `length`, the length from head to tail in m,
- `weight`, the weight in tons,
- `width`, the width in m,
- `height`, the height in m,
- `leg_length`, the average length for the two (rear) legs in m,
- `arm_length`, the average length for the two arms in m,
- `head_length`, the length of the head in cm,
- `species`, the species with two values, `A` and `B`,
- `power`, the strength index for the dinosaur, and
- `bite_st`, the bite strength in normalized units.

Read the data:

```
data <- read.csv('iguanodon.csv')
str(data)
```

```
## 'data.frame':    27 obs. of  10 variables:
##  $ length     : num  9.8 9.7 8.8 9.4 9.4 9.7 8.6 9.1 10.5 10.6 ...
##  $ weight     : num  6 5.5 6.8 6.8 7.8 4.2 7.5 6.5 6.3 7.8 ...
##  $ width      : num  5.7 5.6 5.4 5.3 5.1 5.4 4.6 5.6 6.1 6.1 ...
##  $ height     : num  7.4 7.2 7.1 7.5 6.8 6.5 8.3 7.6 7.8 8.2 ...
##  $ leg_length : num  1.4 1.2 1.8 1.3 1.5 1.5 1.7 1.7 1.8 1.4 ...
##  $ arm_length : num  0.9 0.8 1.2 1.1 1.1 1.1 1.2 1.1 1.3 1 ...
##  $ head_length: num  92.7 92.7 90 91.3 93.9 93.1 91.8 91.6 93.9 94 ...
##  $ species    : chr  "A" "B" "B" "A" ...
##  $ bite_st    : num  77.6 76.8 78.4 77.2 78.5 77.6 77.8 77.1 74.9 78.1 ...
##  $ power      : num  80.2 76.9 76.9 77.2 76.5 76.8 79.3 79.2 82.3 84.4 ...
```

Problem 1

In this question you have to explore the relationship between the variables `bite_st` and `head_length`.

(a) Graph a scatterplot of `bite_st` as a function of `head_length`. Fit a simple regression model for these variables and add the regression line to the plot. Comment on the plot. Print the summary table. What is the R^2 for this model? Write down the equation for the regression line, including all the terms, and give an interpretation of the parameters. Predict the bite strength of an iguanodon with a head length of 90 cm. and include a prediction interval at the 98% confidence level.

Solution

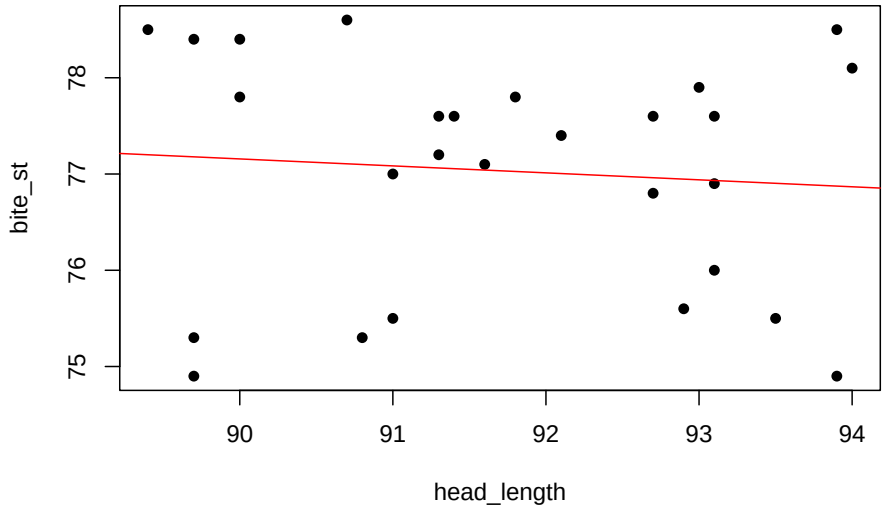
For convenience, we create a smaller data frame with the variables needed for this question and transform `species` into a factor.

```
q1_data <- subset(data,select = c(head_length,bite_st, species))
str(q1_data)
```

```
## 'data.frame':    27 obs. of  3 variables:
##  $ head_length: num  92.7 92.7 90 91.3 93.9 93.1 91.8 91.6 93.9 94 ...
##  $ bite_st    : num  77.6 76.8 78.4 77.2 78.5 77.6 77.8 77.1 74.9 78.1 ...
##  $ species    : chr  "A" "B" "B" "A" ...
```

```
q1_data$species <- factor(q1_data$species)
```

```
plot(bite_st ~ head_length, pch = 16, data = q1_data)
md1 <- lm(bite_st ~ head_length, data = q1_data)
abline(md1, col = 'red')
```



There is not a clear linear relation between the two variables. The dots on the plot look randomly scattered. The regression line shows a small decreasing slope.

```
summary(md1)
```

```
##
## Call:
## lm(formula = bite_st ~ head_length, data = q1_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2784 -1.1395  0.3953  0.8670  1.6256
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  83.6727    15.1197   5.534 9.42e-06 ***
## head_length  -0.0724     0.1648  -0.439    0.664
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.228 on 25 degrees of freedom
## Multiple R-squared:  0.007664,   Adjusted R-squared:  -0.03203
```

```
## F-statistic: 0.1931 on 1 and 25 DF,  p-value: 0.6641
```

The slope has a small negative value -0.0724 and has a large p-value, which says that it is not significantly different from zero. The R^2 is practically zero: 0.00766. The equation for the model, including the slope (even if it is not significant), is

$$\texttt{bite\_st} = 83.6727 - 0.0724 * \texttt{head\_length}$$

The predicted value with a prediction interval is

```
predict(md1, newdata = data.frame(head_length = 90), interval = 'p', level = 0.98)
```

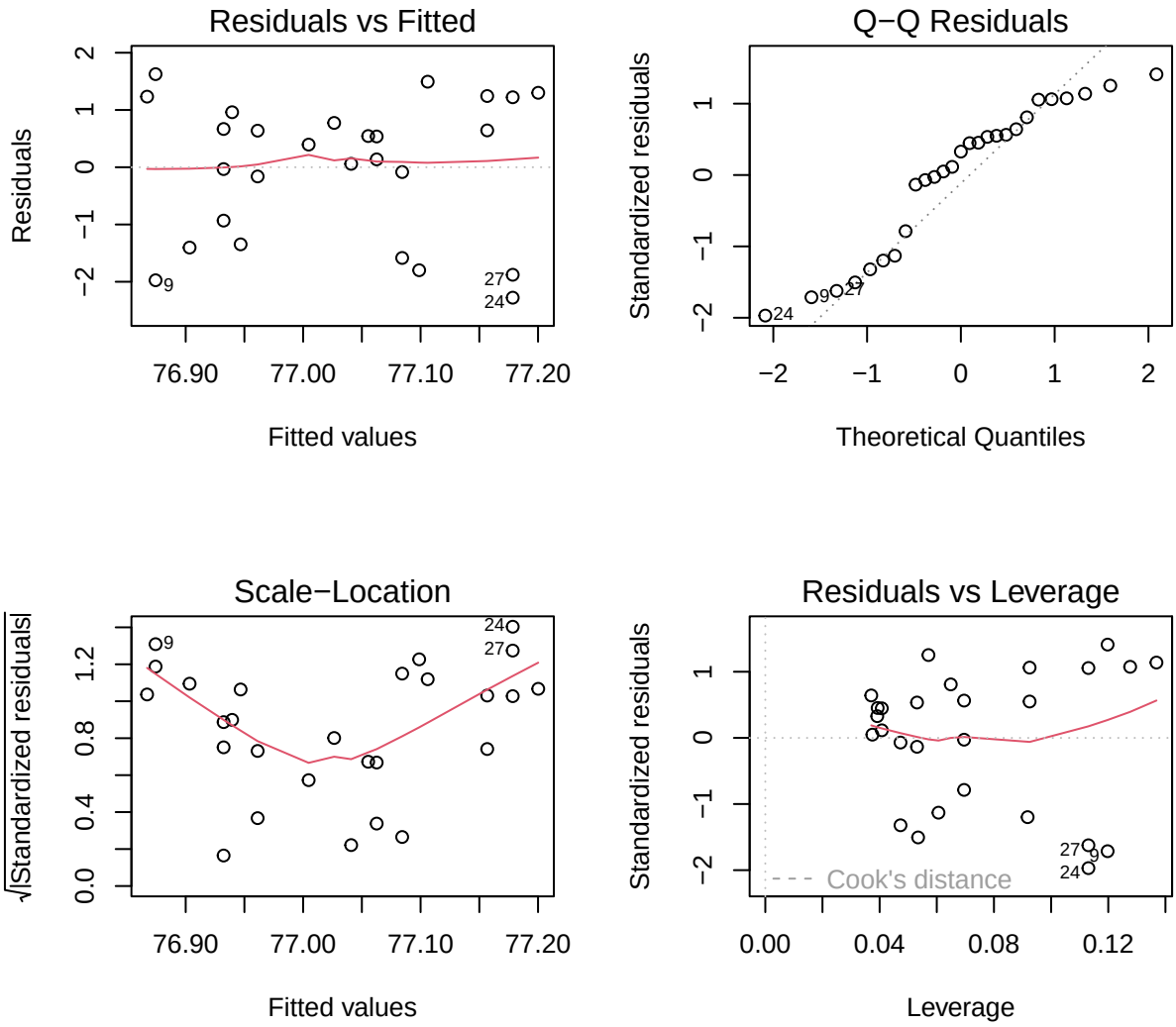
```
##        fit      lwr      upr
## 1 77.15673 73.96595 80.34751
```

(b) State clearly the assumptions on which the linear regression model is based. Use graphical methods and tests to check these assumptions. What are your conclusions?

Solution:

The model assumes that the errors are independent and have common normal distribution with mean zero and unknown variance σ^2. The diagnostic plots are:

```
par(mfrow=c(2,2))
plot(md1)
```



```
par(mfrow = c(1,1))
```

The first and fourth plots do not show important departures from the assumptions. The residuals are centered and are symmetrically distributed around their mean. There are no standardized residuals with large values or high leverage. On the other hand, the quantile plot shows deviations from the reference line, not only at the extremes, but also in the central portion of the plot, while the scale-location plot does not seem to support the assumption of homogeneous variances. We use tests to verify these observations.

```
shapiro.test(rstandard(md1))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rstandard(md1)
## W = 0.91011, p-value = 0.02295
```

```
library(car)
ncvTest(md1)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.4291138, Df = 1, p = 0.51242
```

The Shapiro-Wilk test has a small p-value, indicating that the assumption of normality is not valid. The homogeneity test has a large p-value, indicating that the variance is constant.

(c) There are two species of iguanodons in the file, denoted `A` and `B`, and this characteristic is available in the categorical variable `species`. If this variable was not read as a `factor`, transform it before you continue. Fit a model that includes `head_length`, `species`, and the interaction between the two. Using a critical value for α of 0.05 and starting with the complete model, select a minimal adequate model. Compare the adjusted R^2 with the previous model. Check the assumptions for the final model.

Solution:

We have already transformed `species` into a factor.

```
md2 <- lm(bite_st ~ head_length*species, data = q1_data)
summary(md2)
```

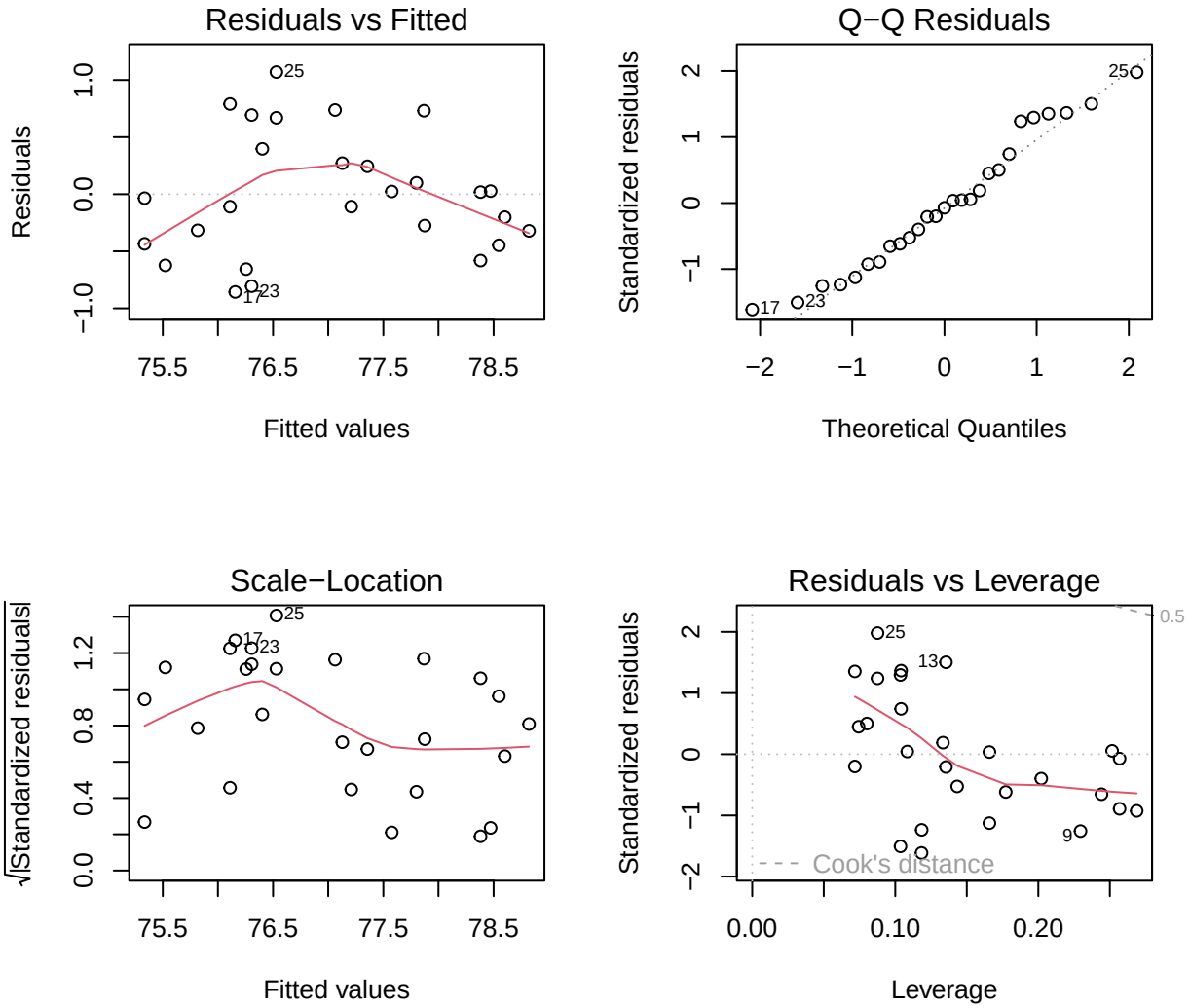
```
##
## Call:
## lm(formula = bite_st ~ head_length * species, data = q1_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.85673 -0.37807 -0.03486  0.33455  1.06969
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)            8.3150    10.4121   0.799    0.433
## head_length            0.7472     0.1134   6.589 1.01e-06 ***
## speciesB             136.0234    14.0097   9.709 1.33e-09 ***
## head_length:speciesB  -1.4800     0.1527  -9.695 1.37e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5656 on 23 degrees of freedom
```

```
## Multiple R-squared:  0.8065, Adjusted R-squared:  0.7813
## F-statistic: 31.95 on 3 and 23 DF,  p-value: 2.244e-08
```

All the terms in the model are significant, and therefore this is the minimal adequate model. The R^2 is 0.806 while the adjusted R^2 is 0.781, a huge difference with the values for the previous model.

Diagnostic plots:

```
par(mfrow = c(2,2))
plot(md2)
```




```
par(mfrow = c(1,1))
```

The first plot shows a curve in the local smoother that points to an asymmetric distribution of the residuals. The quantile and scale-location plots now look better than for the previous model. There are no points with large standardized residuals or leverage values. We check these appreciations with tests

```
shapiro.test(rstandard(md2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rstandard(md2)
## W = 0.95977, p-value = 0.3651
```

```
ncvTest(md2)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 1.628065, Df = 1, p = 0.20197
```

Both tests have large p-values, and the assumptions are satisfied in this case.

(d) Write down the equation for the regression model in (c) and predict the value of the bite strength for iguanodons of both species having head length 90 cm, including prediction intervals at the 98% confidence level. Compare with the previous prediction and comment.

Solution:

Let $\mathbf{1}_B$ be a dummy variable that takes value 1 when the iguanodon belongs to species `B`, and value 0 otherwise. Then the equation is

$$\texttt{bite\_st} = 8.315 + \mathbf{1}_B * 136.023 + (0.747 - \mathbf{1}_B * 1.48) * \texttt{head\_length}$$

The predicted values are

```
predict(md2, newdata = data.frame(head_length = 90, species = 'A'),
        interval = 'p', level = 0.98)
```

```
##        fit      lwr      upr
## 1 75.55901 74.00421 77.11381
```

```
predict(md2, newdata = data.frame(head_length = 90, species = 'B'),
        interval = 'p', level = 0.98)
```

```
##        fit      lwr      upr
## 1 78.38157 76.85505 79.90809
```

The prediction with the previous model was 77.157, roughly the average between the two predicted values obtained with the second model. On the other hand, the width of the prediction interval was approximately 6.4, about twice as large as the intervals obtained with the second model.

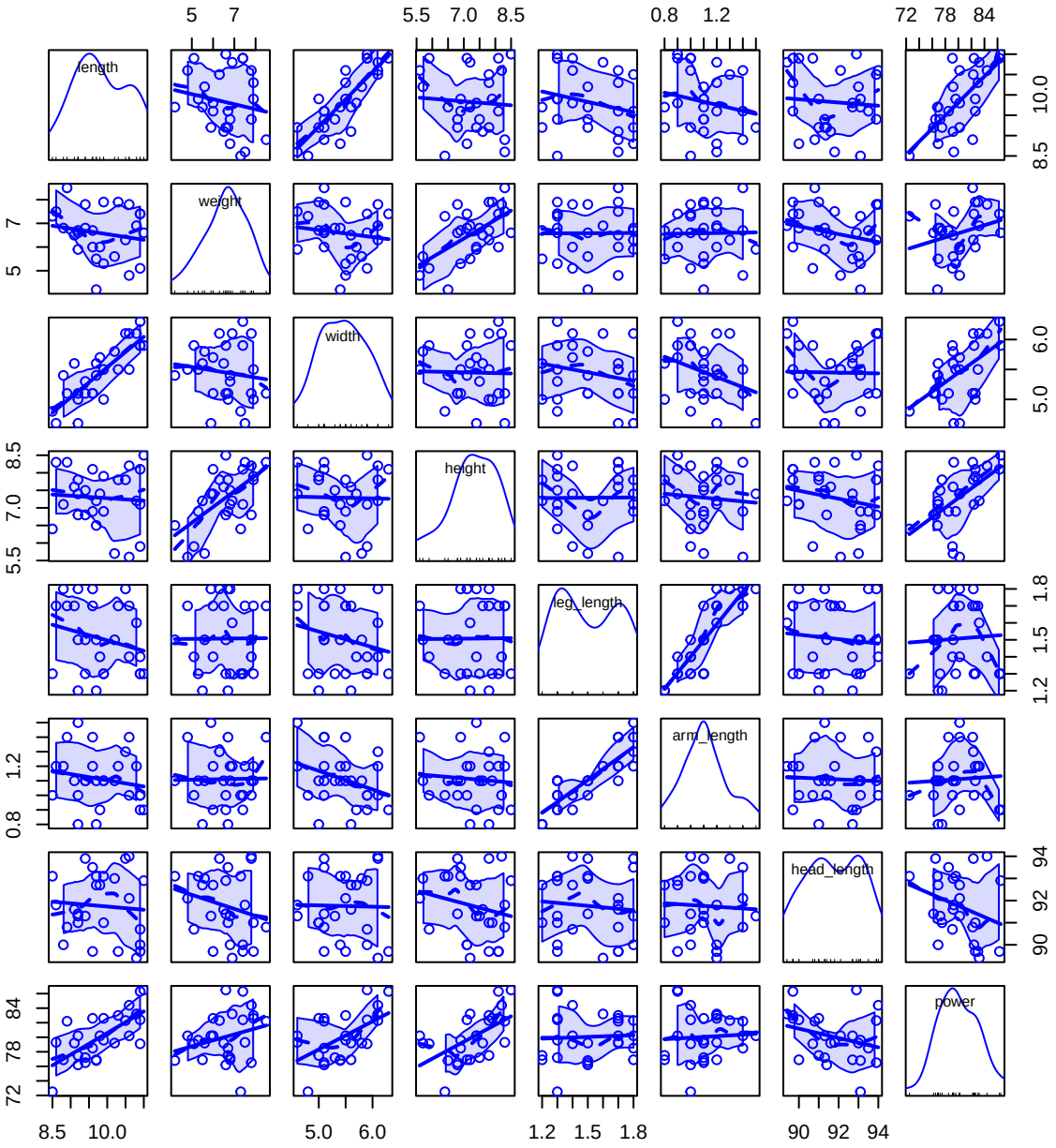
Problem 2

In this question you have to develop a model for `power` as a function of the numerical variables in the set, excluding `bite_st`.

(a) Do a scatterplot matrix for the numerical variables in the data set, excluding `bite_st`. Calculate and graph the correlation matrix for these variables. Comment on the results.

Solution:

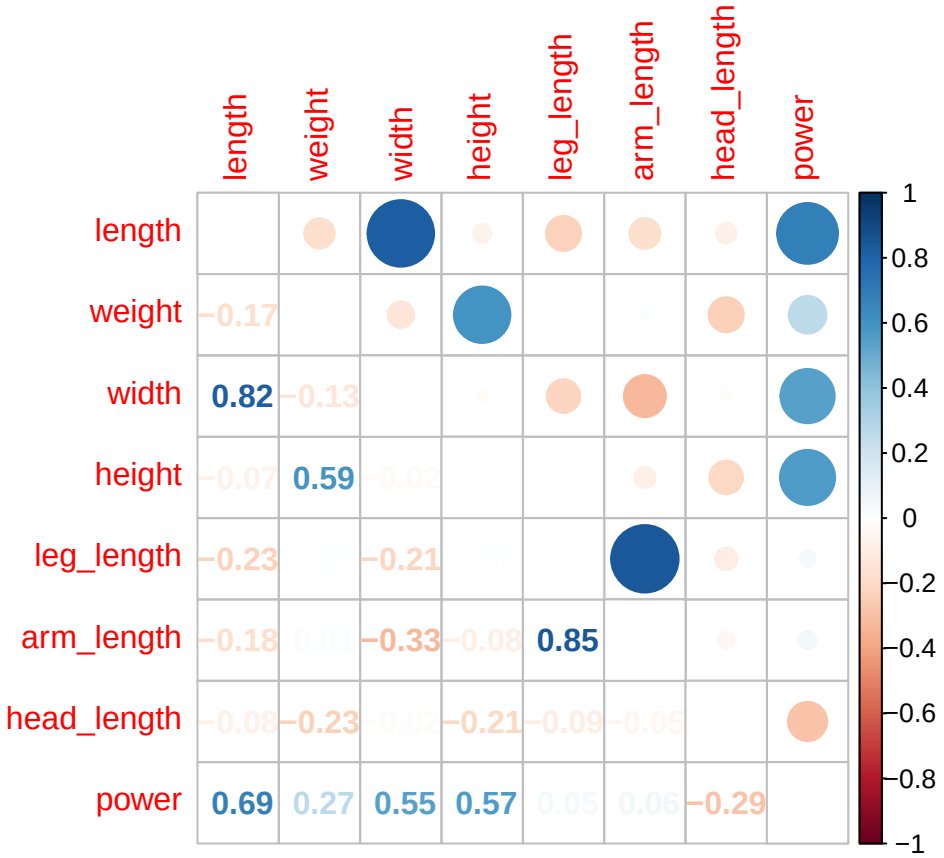
```
scatterplotMatrix(data[,c(-8,-9)])
```



In the scatterplot matrix, `power` is in the last row and we see that `length`, `width`, and `height` show positive correlation, `weight` has moderate positive correlation, `head_length` has moderate negative correlation, while `leg_length` and `arm_length` do not seem to be correlated with `power`. We also see strong correlations between some of the regressors, such as `height` and `width`, `leg_length` and `arm_length`, and `height` and

weight. Since these variables correspond to physical measurements on the same dinosaur, it is to be expected that some of them have high correlations.

```
cor_mat <- cor(data[,c(-8,-9)])
corrplot::corrplot.mixed(cor_mat, tl.pos = 'lt')
```



The correlation plot confirms our previous observations, length, width and height have positive correlations with power, while the other variables have moderate or null correlation. Also, the correlations between arm_length and leg_length, length and width, and weight and height are bigger than 0.5, while all the rest are smaller (in absolute value).

(b) Fit a regression model for power as a function of the variables mentioned in (a). With a threshold for the variance inflation factor of 2, use a sequential procedure to eliminate variables that may cause multicollinearity problems.

Solution:

We fit the model and look at the VIF.

```
mod1 <- lm(power ~ ., data = data[,c(-8,-9)])
summary(mod1)
```

```
##
## Call:
## lm(formula = power ~ ., data = data[, c(-8, -9)])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9919 -0.6305 -0.1436  0.6583  2.1974
```

```
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 41.35338   15.83886   2.611   0.0172 *
## length       3.24637    0.58450   5.554 2.34e-05 ***
## weight       0.02922    0.26298   0.111   0.9127
## width        0.14476    0.97151   0.149   0.8831
## height       2.52616    0.34419   7.339 5.88e-07 ***
## leg_length   0.25032    2.32813   0.108   0.9155
## arm_length   4.20560    2.73496   1.538   0.1406
## head_length -0.19213    0.15488  -1.240   0.2299
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.09 on 19 degrees of freedom
## Multiple R-squared:  0.9197, Adjusted R-squared:  0.8902
## F-statistic:  31.1 on 7 and 19 DF,  p-value: 4.131e-09
```

```
vif(mod1)
```

```
##      length      weight       width      height  leg_length  arm_length
##    4.363021    1.691988    4.561635    1.650376    5.006983    5.451407
## head_length
##    1.121318
```

Four values are bigger than 2, the biggest being `arm_length`, so we drop this variable from the model.

```
mod2 <- update(mod1, .~. - arm_length)
vif(mod2)
```

```
##      length      weight       width      height  leg_length head_length
##    3.270955    1.633020    3.161675    1.570274    1.074806    1.105472
```

The largest value now corresponds to `length`, so we drop this variable from the model:

```
mod3 <- update(mod2, .~. - length)
vif(mod3)
```

```
##      weight       width      height  leg_length head_length
##    1.617067    1.076528    1.567925    1.059591    1.079587
```

Now all the values are below the threshold. We keep five variables: `weight`, `width`, `height`, `leg_length`, and `head_length`.

(c) Using a backward selection procedure with a critical α of 0.10 and starting with the variables you selected in (b), obtain a minimal adequate model. Comment on the steps that you take.

Solution:

We start with the summary table for `mod3`:

```
summary(mod3)
```

```
##
## Call:
## lm(formula = power ~ weight + width + height + leg_length + head_length,
##     data = data[, c(-8, -9)])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -3.8193 -1.3832  0.1916  0.8991  3.3430
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 68.36126   28.76699   2.376 0.027074 *
## weight      -0.08177    0.48488  -0.169 0.867687
## width        4.11408    0.89012   4.622 0.000147 ***
## height       2.31472    0.63273   3.658 0.001466 **
## leg_length   2.48564    2.01994   1.231 0.232097
## head_length -0.33567    0.28663  -1.171 0.254669
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.057 on 21 degrees of freedom
## Multiple R-squared:  0.6844, Adjusted R-squared:  0.6093
## F-statistic: 9.108 on 5 and 21 DF,  p-value: 0.0001002
```

The largest p-value above the critical α corresponds to `weight`, so we drop this variable from the model:

```
mod4 <- update(mod3, . ~ . - weight)
summary(mod4)
```

```
##
## Call:
## lm(formula = power ~ width + height + leg_length + head_length,
##     data = data[, c(-8, -9)])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8396 -1.3720  0.2258  0.9210  3.3668
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  67.4802    27.6570   2.440 0.023210 *
## width         4.1381     0.8591   4.817 8.22e-05 ***
## height        2.2532     0.5055   4.458 0.000197 ***
## leg_length    2.5012     1.9728   1.268 0.218109
## head_length  -0.3287     0.2773  -1.185 0.248513
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.011 on 22 degrees of freedom
## Multiple R-squared:  0.684,  Adjusted R-squared:  0.6265
## F-statistic:  11.9 on 4 and 22 DF,  p-value: 2.677e-05
```

The largest p-value above the critical α corresponds to `head_length`, so we drop this variable from the model:

```
mod5 <- update(mod4, .~. - head_length)
summary(mod5)
```

```
##
## Call:
## lm(formula = power ~ width + height + leg_length, data = data[,
##     c(-8, -9)])
##
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -3.3059 -1.1861  0.0252  1.3948  3.5334
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  35.7895     7.1449   5.009 4.56e-05 ***
## width         4.1840     0.8657   4.833 7.06e-05 ***
## height        2.3794     0.4985   4.773 8.19e-05 ***
## leg_length    2.7384     1.9798   1.383     0.18
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.028 on 23 degrees of freedom
## Multiple R-squared:  0.6638, Adjusted R-squared:  0.6199
## F-statistic: 15.14 on 3 and 23 DF,  p-value: 1.181e-05
```

The largest p-value above the critical α corresponds to `leg_length`, so we drop this variable from the model:

```
mod6 <- update(mod5, .~. - leg_length)
summary(mod6)
```

```
##
## Call:
## lm(formula = power ~ width + height, data = data[, c(-8, -9)])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5409 -1.4291  0.2667  1.3143  3.9628
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  41.2758     6.0548   6.817 4.74e-07 ***
## width         3.9317     0.8622   4.560 0.000127 ***
## height        2.3816     0.5079   4.689 9.15e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.067 on 24 degrees of freedom
## Multiple R-squared:  0.6358, Adjusted R-squared:  0.6055
## F-statistic: 20.95 on 2 and 24 DF,  p-value: 5.442e-06
```

Now all the p-values are below the critical α and this is the minimal adequate model. It has only two variables: `width` and `height`.

(d) Fit a model using the BIC criterion starting with the variables you selected in (b). Compare your final model with the result of (c).

Solution:

We start with `mod3` that was built after eliminating the variables with large VIF.

```
library(MASS)
stepAIC(mod3, k = log(27))
```

```
## Start:  AIC=51.93
## power ~ weight + width + height + leg_length + head_length
##
```

```
##               Df Sum of Sq     RSS    AIC
## - weight       1     0.120  88.952 48.670
## - head_length  1     5.802  94.633 50.342
## - leg_length   1     6.405  95.237 50.513
## <none>                      88.831 51.929
## - height       1    56.612 145.443 61.946
## - width        1    90.363 179.194 67.580
##
## Step:  AIC=48.67
## power ~ width + height + leg_length + head_length
##
##               Df Sum of Sq     RSS    AIC
## - head_length  1     5.681  94.633 47.046
## - leg_length   1     6.499  95.451 47.278
## <none>                      88.952 48.670
## - height       1    80.342 169.293 62.750
## - width        1    93.815 182.766 64.817
##
## Step:  AIC=47.05
## power ~ width + height + leg_length
##
##              Df Sum of Sq     RSS    AIC
## - leg_length  1     7.872 102.504 45.907
## <none>                    94.633 47.046
## - height      1    93.749 188.382 62.339
## - width       1    96.106 190.738 62.674
##
## Step:  AIC=45.91
## power ~ width + height
##
##          Df Sum of Sq    RSS    AIC
## <none>                102.50 45.907
## - width   1    88.807 191.31 59.460
## - height  1    93.923 196.43 60.172

##
## Call:
## lm(formula = power ~ width + height, data = data[, c(-8, -9)])
##
## Coefficients:
## (Intercept)        width       height
##      41.276        3.932        2.382
```

We get exactly the same model as in (c).

(e) Write an equation for the final model in (c) and interpret the coefficients. Predict the `power` for an iguanodon with the following covariates. Include confidence intervals at the 99% level.

Table 1: Covariates for prediction

length	weight	width	height	leg_length	arm_length	head_length
10	5.1	5.2	7.3	1.4	1.3	89

Solution:

The equation for the model is

$$\texttt{power} = 41.28 + 3.93 * \texttt{width} + 2.38 * \texttt{height}.$$

The intercept represents the smallest value that `power` can have, i.e., it is the value that corresponds to setting the two covariates to zero in the equation. On the other hand, an increase of 1 unit in `width` produces an increase of 3.93 units in power, assuming that `height` is kept fixed. Similarly, if `width` is kept fixed, an increase in one unit in `height` produces an increase of 2.38 units in power.

To obtain the predicted value, we use the function `predict`:

```
predict(mod6, newdata = data.frame(width = 5.2, height = 7.3),
        interval = 'c', level = 0.99)
```

```
##        fit      lwr      upr
## 1 79.10636 77.84325 80.36947
```

The predicted value is 79.106, with confidence interval (77.843, 80.369).

(f) Print an anova table for the final model and find the estimated standard deviation of the errors. Describe explicitly the sampling distribution for the estimated parameters.

Solution:

The anova table is

```
anova(mod6)
```

```
## Analysis of Variance Table
##
## Response: power
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## width      1  85.039  85.039  19.911 0.0001631 ***
## height     1  93.923  93.923  21.991 9.147e-05 ***
## Residuals 24 102.504   4.271
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The variance of the errors is estimated by the MSE, which is 4.3. The standard deviation is 2.0736441.

We have three estimated parameters, $\hat{\beta}_0, \hat{\beta}_w$, and $\hat{\beta}_h$, corresponding to the `intercept`, `width` and `height`: $\hat{\beta} = (\hat{\beta}_0, \hat{\beta}_w, \hat{\beta}_h)$. This vector has a multivariate normal distribution with mean $\beta = (\beta_0, \beta_w, \beta_h)$, the vector of true values for the parameters of the regression, and covariance matrix $U = \sigma^2(\mathbf{X}'\mathbf{X})^{-1}$, where σ^2 is the variance of the errors and $\mathbf{X}$ is the design matrix. U is estimated by

```
round(vcov(mod6),4)
```

```
##             (Intercept)   width  height
## (Intercept)     36.6610 -4.1177 -1.9302
## width           -4.1177  0.7435  0.0092
## height          -1.9302  0.0092  0.2579
```

Problem 3

Using the `sat` dataset in the `faraway` package, fit a model with the `total` SAT score as the response and `expend`, `salary`, `ratio` and `takers` as predictors. Perform regression diagnostics on this model to answer the following questions. Display any plots that are relevant. Do not provide any plots about which you have nothing to say.

```
library(faraway)
library(car)
str(sat)
```

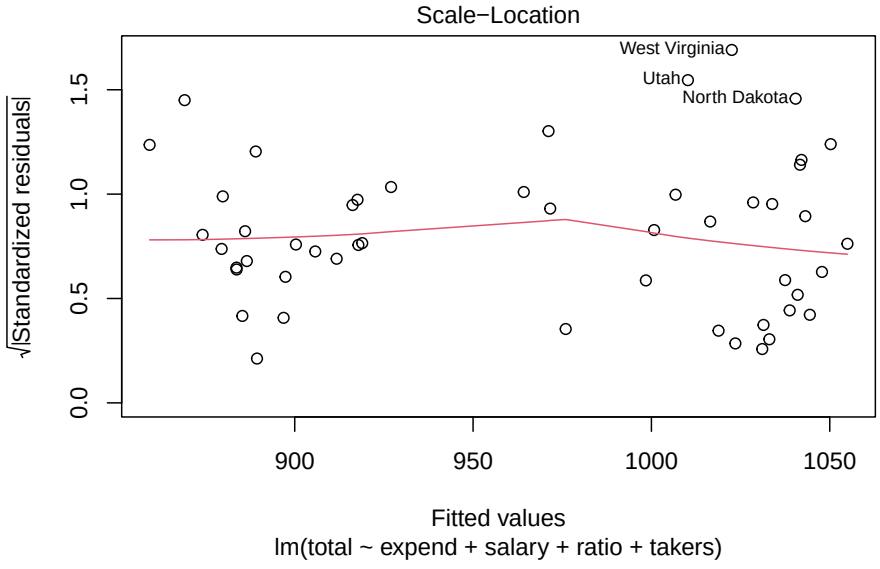
```
## 'data.frame':    50 obs. of  7 variables:
##  $ expend: num  4.41 8.96 4.78 4.46 4.99 ...
##  $ ratio : num  17.2 17.6 19.3 17.1 24 18.4 14.4 16.6 19.1 16.3 ...
##  $ salary: num  31.1 48 32.2 28.9 41.1 ...
##  $ takers: int  8 47 27 6 45 29 81 68 48 65 ...
##  $ verbal: int  491 445 448 482 417 462 431 429 420 406 ...
##  $ math  : int  538 489 496 523 485 518 477 468 469 448 ...
##  $ total : int  1029 934 944 1005 902 980 908 897 889 854 ...
```

```
q4.mod <- lm(total ~ expend + salary + ratio + takers, data = sat)
summary(q4.mod)
```

```
##
## Call:
## lm(formula = total ~ expend + salary + ratio + takers, data = sat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -90.531 -20.855  -1.746  15.979  66.571
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1045.9715    52.8698  19.784  < 2e-16 ***
## expend         4.4626    10.5465   0.423    0.674
## salary         1.6379     2.3872   0.686    0.496
## ratio         -3.6242     3.2154  -1.127    0.266
## takers        -2.9045     0.2313 -12.559 2.61e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.7 on 45 degrees of freedom
## Multiple R-squared:  0.8246, Adjusted R-squared:  0.809
## F-statistic: 52.88 on 4 and 45 DF,  p-value: < 2.2e-16
```

(a) Check the constant variance assumption for the errors.

```
plot(q4.mod, which = 3)
```



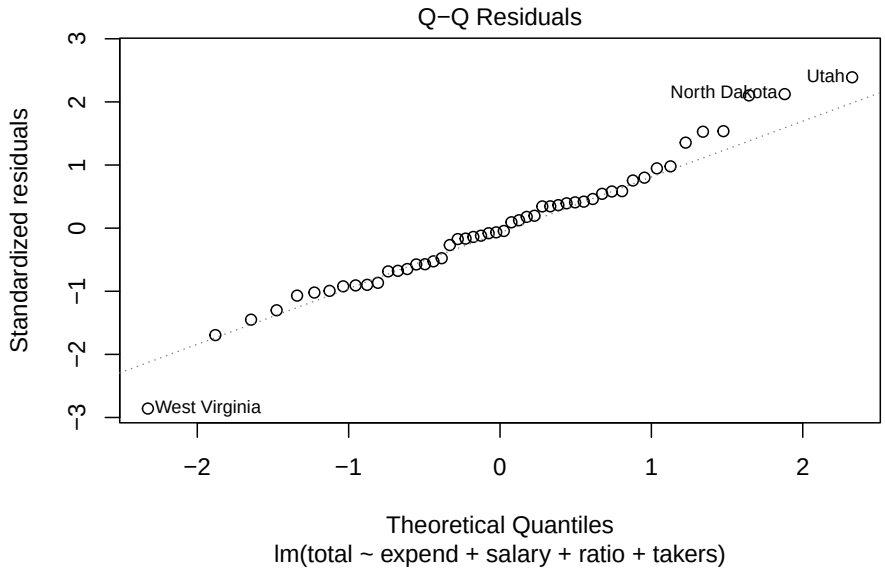
```
ncvTest(q4.mod)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 0.6972119, Df = 1, p = 0.40372
```

The assumption seems satisfied.

(b) Check the normality assumption.

```
plot(q4.mod, which = 2)
```



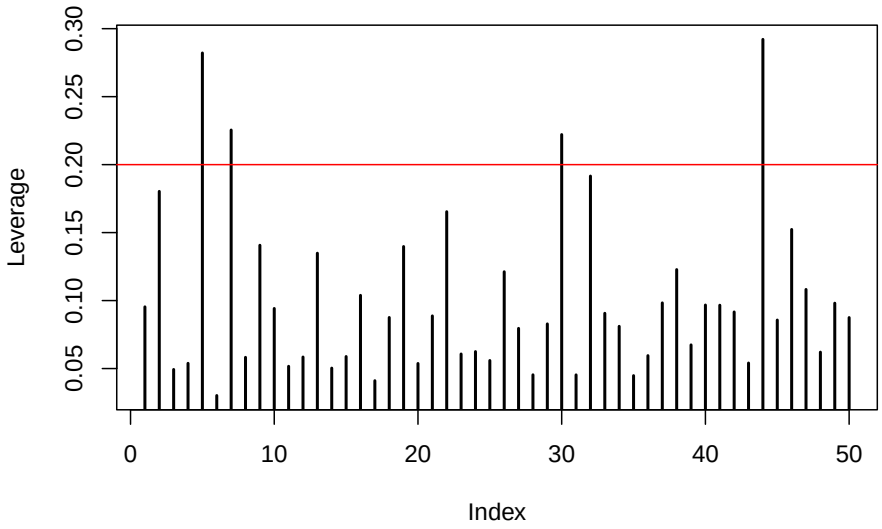

```
shapiro.test(q4.mod$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  q4.mod$residuals
## W = 0.97691, p-value = 0.4304
```

The plot and test say that the normality assumption is satisfied.

(c) Check for large leverage points.

```
plot(hatvalues(q4.mod), type = 'h', lwd=2, ylab='Leverage')
abline(h=0.2, col='red')
```



There are four points above the red line at $2p/n = 10/20 = 0.2$ that correspond to

```
high.lev <- (1:50)[hatvalues(q4.mod)>0.2]
dimnames(sat)[[1]][high.lev]
```

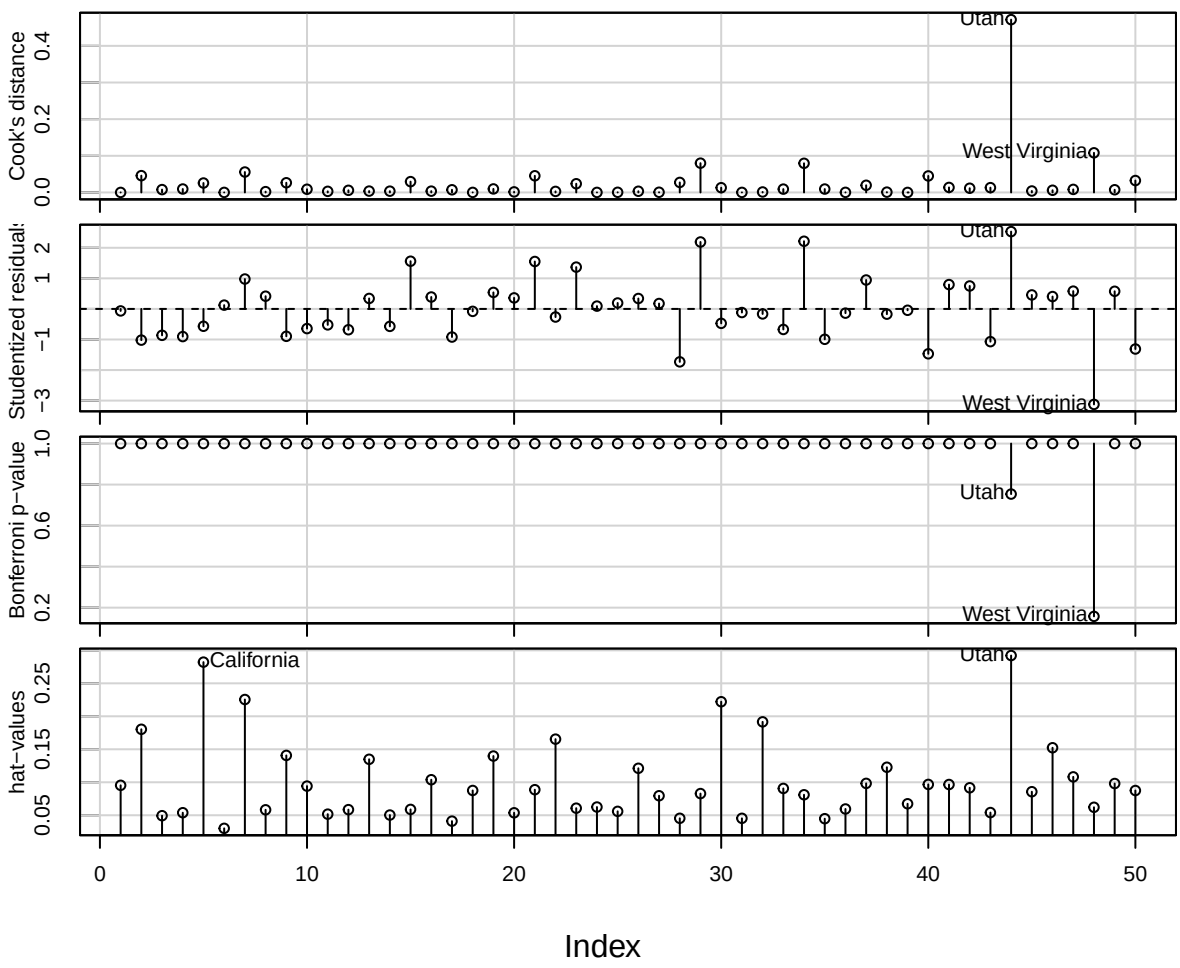
```
## [1] "California"  "Connecticut" "New Jersey"  "Utah"
```

The two highest values correspond to Utah and California.

(d) Check for outliers.

The following plots are from the `car` package. There are other alternatives.

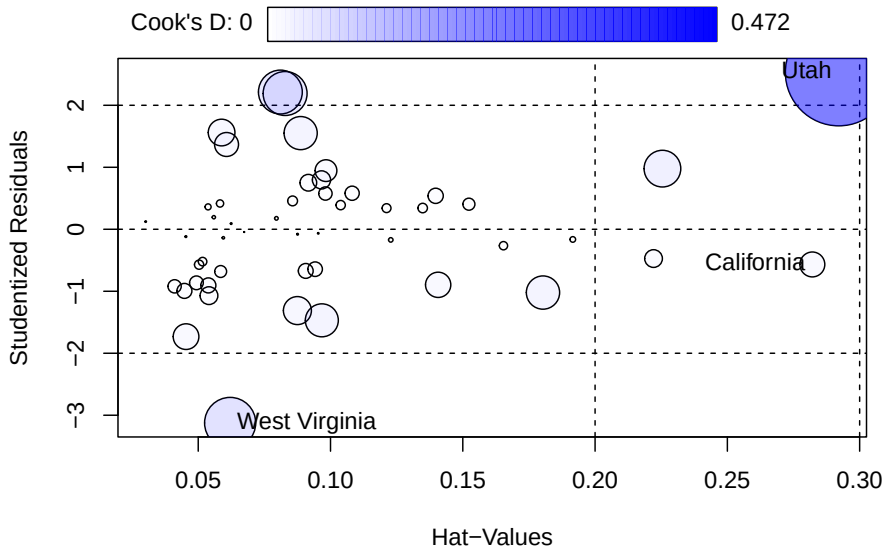
```
influenceIndexPlot(q4.mod)
```



In these plots West Virginia also appears as a possible outlier, even though the p-value in the Bonferroni graph is not small enough. However, this test is conservative and this may well be an outlier.

(e) Check for influential points.

```
influencePlot(q4.mod)
```




```
##                  StudRes        Hat      CookD
## California    -0.5676458 0.28211791 0.02571304
## Utah           2.5295873 0.29211280 0.47152866
## West Virginia -3.1244283 0.06206536 0.10813954
```

These are the three points that should be checked. As an example (but this was not required) we can look at the changes in the model parameters produced by leaving out these points. These differences are included in the DFBETAS vector that can be extracted from the model using the function `dfbeta()`. For instance, for Utah the DFBETAS are

```
dfbeta(q4.mod)[44,]
```

```
##  (Intercept)       expend       salary        ratio       takers
## -47.87443743   5.40533354  -1.45851225   4.01491196   0.02632387
```

and the coefficients for the complete model are

```
coef(q4.mod)
```

```
## (Intercept)      expend      salary       ratio      takers
## 1045.971536    4.462594    1.637917   -3.624232   -2.904481
```

We see that the changes in some of the coefficients are quite significant.

For West Virginia

```
dfbeta(q4.mod)[48,]
```

```
##  (Intercept)       expend       salary        ratio       takers
## -11.70596970  -2.89670620   0.55038543   0.31550287   0.06791298
```

and for California

```
dfbeta(q4.mod)[5,]
```

```
## (Intercept)      expend      salary       ratio      takers
##  9.31992536  1.26239336 -0.30738250 -0.36782899 -0.00878391
```

Problem 4

For this question, use the data set `uscrime` in the package `HH`. After loading the library, you need to run `data("uscrime")`. Do not mistake with `UScrime`. For this exercise, values for the variance inflation factor

(vif) below 5 are considered acceptable. The following commands load the data:

```
library(HH)
data("uscrime")
```

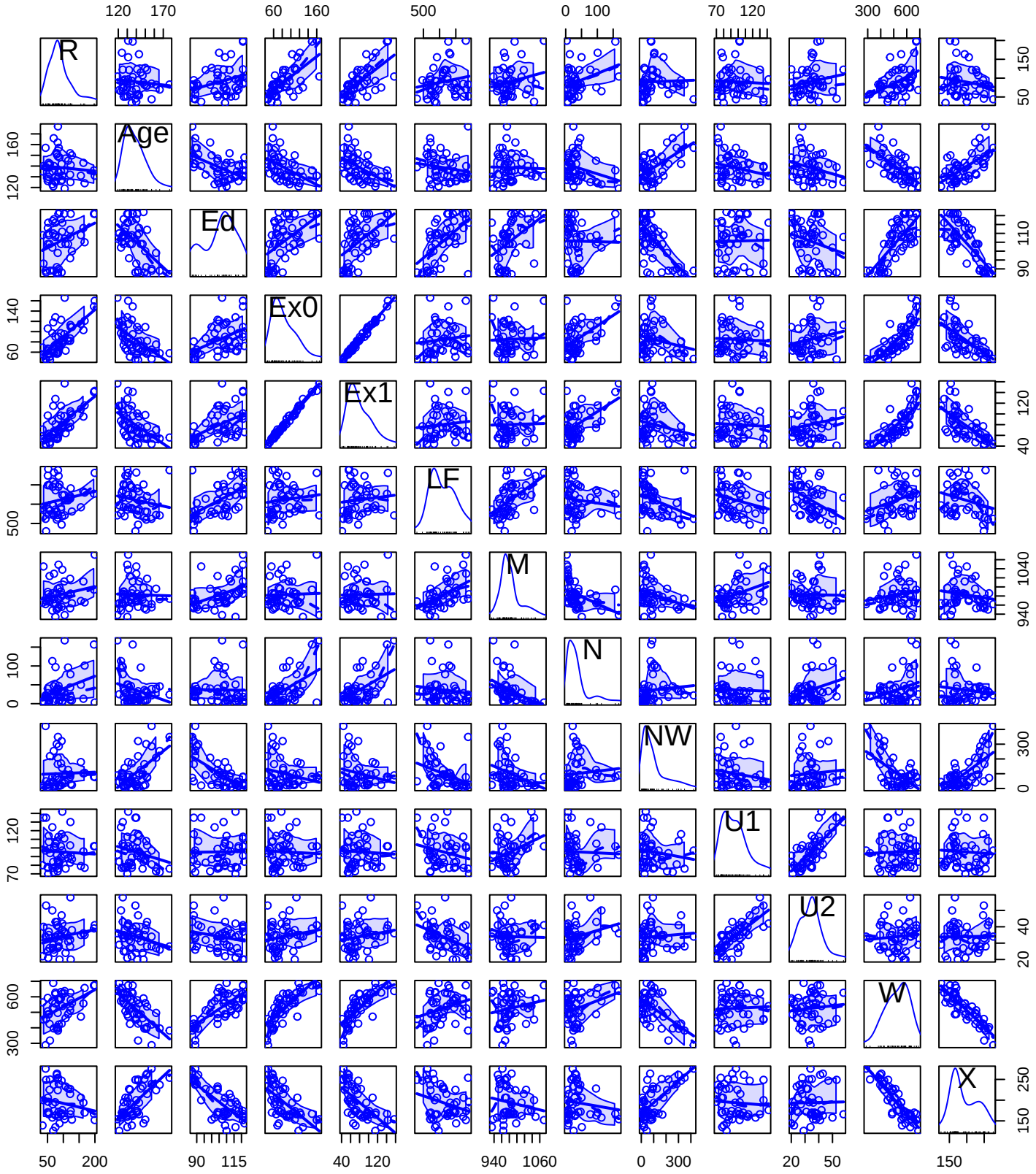
(a) Fit a multiple regression model for `R` using all the other variables except `State`. Look at the summary and variance inflation factors and comment.

We start by looking at the structure of the data set and plotting a scatterplot matrix.

```
str(uscrime)
```

```
## 'data.frame':    47 obs. of  15 variables:
##  $ R    : num  79.1 163.5 57.8 196.9 123.4 ...
##  $ Age  : int  151 143 142 136 141 121 127 131 157 140 ...
##  $ S    : int  1 0 1 0 0 0 1 1 1 0 ...
##  $ Ed   : int  91 113 89 121 121 110 111 109 90 118 ...
##  $ Ex0  : int  58 103 45 149 109 118 82 115 65 71 ...
##  $ Ex1  : int  56 95 44 141 101 115 79 109 62 68 ...
##  $ LF   : int  510 583 533 577 591 547 519 542 553 632 ...
##  $ M    : int  950 1012 969 994 985 964 982 969 955 1029 ...
##  $ N    : int  33 13 18 157 18 25 4 50 39 7 ...
##  $ NW   : int  301 102 219 80 30 44 139 179 286 15 ...
##  $ U1   : int  108 96 94 102 91 84 97 79 81 100 ...
##  $ U2   : int  41 36 33 39 20 29 38 35 28 24 ...
##  $ W    : int  394 557 318 673 578 689 620 472 421 526 ...
##  $ X    : int  261 194 250 167 174 126 168 206 239 174 ...
##  $ State: Factor w/ 47 levels "Alabama","Arizona",..: 1 2 3 4 5 6 7 8 9 10 ...
```

```
scatterplotMatrix(uscrime[,c(-3,-15)])
```



We fit a linear regression model and look at the summary table and vif values.

```
lm1 <- lm(R ~ . - State, data = uscrime)
summary(lm1)
```

```
##
## Call:
## lm(formula = R ~ . - State, data = uscrime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -34.884 -11.923  -1.135  13.495  50.560
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.918e+02  1.559e+02  -4.438 9.56e-05 ***
## Age          1.040e+00  4.227e-01   2.460  0.01931 *
## S           -8.308e+00  1.491e+01  -0.557  0.58117
## Ed           1.802e+00  6.496e-01   2.773  0.00906 **
## Ex0          1.608e+00  1.059e+00   1.519  0.13836
## Ex1         -6.673e-01  1.149e+00  -0.581  0.56529
## LF          -4.103e-02  1.535e-01  -0.267  0.79087
## M            1.648e-01  2.099e-01   0.785  0.43806
## N           -4.128e-02  1.295e-01  -0.319  0.75196
## NW           7.175e-03  6.387e-02   0.112  0.91124
## U1          -6.017e-01  4.372e-01  -1.376  0.17798
## U2           1.792e+00  8.561e-01   2.093  0.04407 *
## W            1.374e-01  1.058e-01   1.298  0.20332
## X            7.929e-01  2.351e-01   3.373  0.00191 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.94 on 33 degrees of freedom
## Multiple R-squared:  0.7692, Adjusted R-squared:  0.6783
## F-statistic: 8.462 on 13 and 33 DF,  p-value: 3.686e-07
```

```
vif(lm1)
```

```
##       Age         S        Ed       Ex0       Ex1        LF         M         N
##  2.698021  4.876751  5.049442 94.633118 98.637233  3.677557  3.658444  2.324326
##        NW        U1        U2         W         X
##  4.123274  5.938264  4.997617  9.968958  8.409449
```

We see that many regressors are not significant and that some have very high vif values, particularly `Ex0` and `Ex1`.

(b) Use the function `stepAIC` in package `MASS` to get a reduced model. Get information about this model using `summary` and look at the variance inflation factors. **Comment on these results**.

```
modelAIC <- stepAIC(lm1)
```

```
## Start:  AIC=301.66
## R ~ (Age + S + Ed + Ex0 + Ex1 + LF + M + N + NW + U1 + U2 + W +
##     X + State) - State
##
##        Df Sum of Sq   RSS    AIC
## - NW    1       6.1 15885 299.68
## - LF    1      34.4 15913 299.76
## - N     1      48.9 15928 299.81
## - S     1     149.4 16028 300.10
```

```
## - Ex1   1     162.3 16041 300.14
## - M     1     296.5 16175 300.53
## <none>              15879 301.66
## - W     1     810.6 16689 302.00
## - U1    1     911.5 16790 302.29
## - Ex0   1    1109.8 16988 302.84
## - U2    1    2108.8 17988 305.52
## - Age   1    2911.6 18790 307.57
## - Ed    1    3700.5 19579 309.51
## - X     1    5474.2 21353 313.58
##
## Step:  AIC=299.68
## R ~ Age + S + Ed + Ex0 + Ex1 + LF + M + N + U1 + U2 + W + X
##
##        Df Sum of Sq   RSS    AIC
## - LF    1      28.7 15913 297.76
## - N     1      48.6 15933 297.82
## - Ex1   1     156.3 16041 298.14
## - S     1     158.0 16043 298.14
## - M     1     294.1 16179 298.54
## <none>              15885 299.68
## - W     1     820.2 16705 300.05
## - U1    1     913.1 16798 300.31
## - Ex0   1    1104.3 16989 300.84
## - U2    1    2107.1 17992 303.53
## - Age   1    3365.8 19250 306.71
## - Ed    1    3757.1 19642 307.66
## - X     1    5503.6 21388 311.66
##
## Step:  AIC=297.76
## R ~ Age + S + Ed + Ex0 + Ex1 + M + N + U1 + U2 + W + X
##
##        Df Sum of Sq   RSS    AIC
## - N     1      62.2 15976 295.95
## - S     1     129.4 16043 296.14
## - Ex1   1     134.8 16048 296.16
## - M     1     276.8 16190 296.57
## <none>              15913 297.76
## - W     1     801.9 16715 298.07
## - U1    1     941.8 16855 298.47
## - Ex0   1    1075.9 16989 298.84
## - U2    1    2088.5 18002 301.56
## - Age   1    3407.9 19321 304.88
## - Ed    1    3895.3 19809 306.06
## - X     1    5621.3 21535 309.98
##
## Step:  AIC=295.95
## R ~ Age + S + Ed + Ex0 + Ex1 + M + U1 + U2 + W + X
##
##        Df Sum of Sq   RSS    AIC
## - S     1     104.4 16080 294.25
## - Ex1   1     123.3 16099 294.31
## - M     1     533.8 16509 295.49
## <none>              15976 295.95
```

```
## - W     1     748.7 16724 296.10
## - U1    1     997.7 16973 296.80
## - Ex0   1    1021.3 16997 296.86
## - U2    1    2082.3 18058 299.71
## - Age   1    3425.9 19402 303.08
## - Ed    1    3887.6 19863 304.19
## - X     1    5896.9 21873 308.71
##
## Step:  AIC=294.25
## R ~ Age + Ed + Ex0 + Ex1 + M + U1 + U2 + W + X
##
##        Df Sum of Sq   RSS    AIC
## - Ex1   1     171.5 16252 292.75
## - M     1     563.4 16643 293.87
## <none>              16080 294.25
## - W     1     734.7 16815 294.35
## - U1    1     906.0 16986 294.83
## - Ex0   1    1162.0 17242 295.53
## - U2    1    1978.0 18058 297.71
## - Age   1    3354.5 19434 301.16
## - Ed    1    4139.1 20219 303.02
## - X     1    6094.8 22175 307.36
##
## Step:  AIC=292.75
## R ~ Age + Ed + Ex0 + M + U1 + U2 + W + X
##
##        Df Sum of Sq   RSS    AIC
## - M     1     691.0 16942 292.71
## <none>              16252 292.75
## - W     1     759.0 17010 292.90
## - U1    1     921.8 17173 293.35
## - U2    1    2018.1 18270 296.25
## - Age   1    3323.1 19574 299.50
## - Ed    1    4005.1 20256 301.11
## - X     1    6402.7 22654 306.36
## - Ex0   1   11818.8 28070 316.44
##
## Step:  AIC=292.71
## R ~ Age + Ed + Ex0 + U1 + U2 + W + X
##
##        Df Sum of Sq   RSS    AIC
## - U1    1     408.6 17351 291.83
## <none>              16942 292.71
## - W     1    1016.9 17959 293.45
## - U2    1    1548.6 18491 294.82
## - Age   1    4511.6 21454 301.81
## - Ed    1    6430.6 23373 305.83
## - X     1    8147.7 25090 309.16
## - Ex0   1   12019.6 28962 315.91
##
## Step:  AIC=291.83
## R ~ Age + Ed + Ex0 + U2 + W + X
##
##        Df Sum of Sq   RSS    AIC
```

```
## <none>              17351 291.83
## - W     1    1252.6 18604 293.11
## - U2    1    1628.7 18980 294.05
## - Age   1    4461.0 21812 300.58
## - Ed    1    6214.7 23566 304.22
## - X     1    8932.3 26283 309.35
## - Ex0   1   15596.5 32948 319.97
```

```
summary(modelAIC)
```

```
##
## Call:
## lm(formula = R ~ Age + Ed + Ex0 + U2 + W + X, data = uscrime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -38.306 -10.209  -1.313   9.919  54.544
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -618.5028   108.2456  -5.714 1.19e-06 ***
## Age            1.1252     0.3509   3.207 0.002640 **
## Ed             1.8179     0.4803   3.785 0.000505 ***
## Ex0            1.0507     0.1752   5.996 4.78e-07 ***
## U2             0.8282     0.4274   1.938 0.059743 .
## W              0.1596     0.0939   1.699 0.097028 .
## X              0.8236     0.1815   4.538 5.10e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.83 on 40 degrees of freedom
## Multiple R-squared:  0.7478, Adjusted R-squared:   0.71
## F-statistic: 19.77 on 6 and 40 DF,  p-value: 1.441e-10
```

```
vif(modelAIC)
```

```
##      Age       Ed      Ex0       U2        W        X
## 2.061942 3.061153 2.875709 1.381671 8.705602 5.559788
```

The `stepAIC` function has dropped many terms but we still have two vif values above 5, `W` and `X`.

(c) Starting with the model produced in (b), drop any variables that have a vif greater that 5 or non-significant p-value. Give a summary of your final model and write down the corresponding equation.

We drop `W`, that has the highest vif value

```
modelAIC2 <- update(modelAIC, .~. - W)
vif(modelAIC2)
```

```
##      Age       Ed      Ex0       U2        X
## 1.997555 2.852816 1.796182 1.362551 3.479589
```

```
summary(modelAIC2)
```

```
##
## Call:
## lm(formula = R ~ Age + Ed + Ex0 + U2 + X, data = uscrime)
##
## Residuals:
```

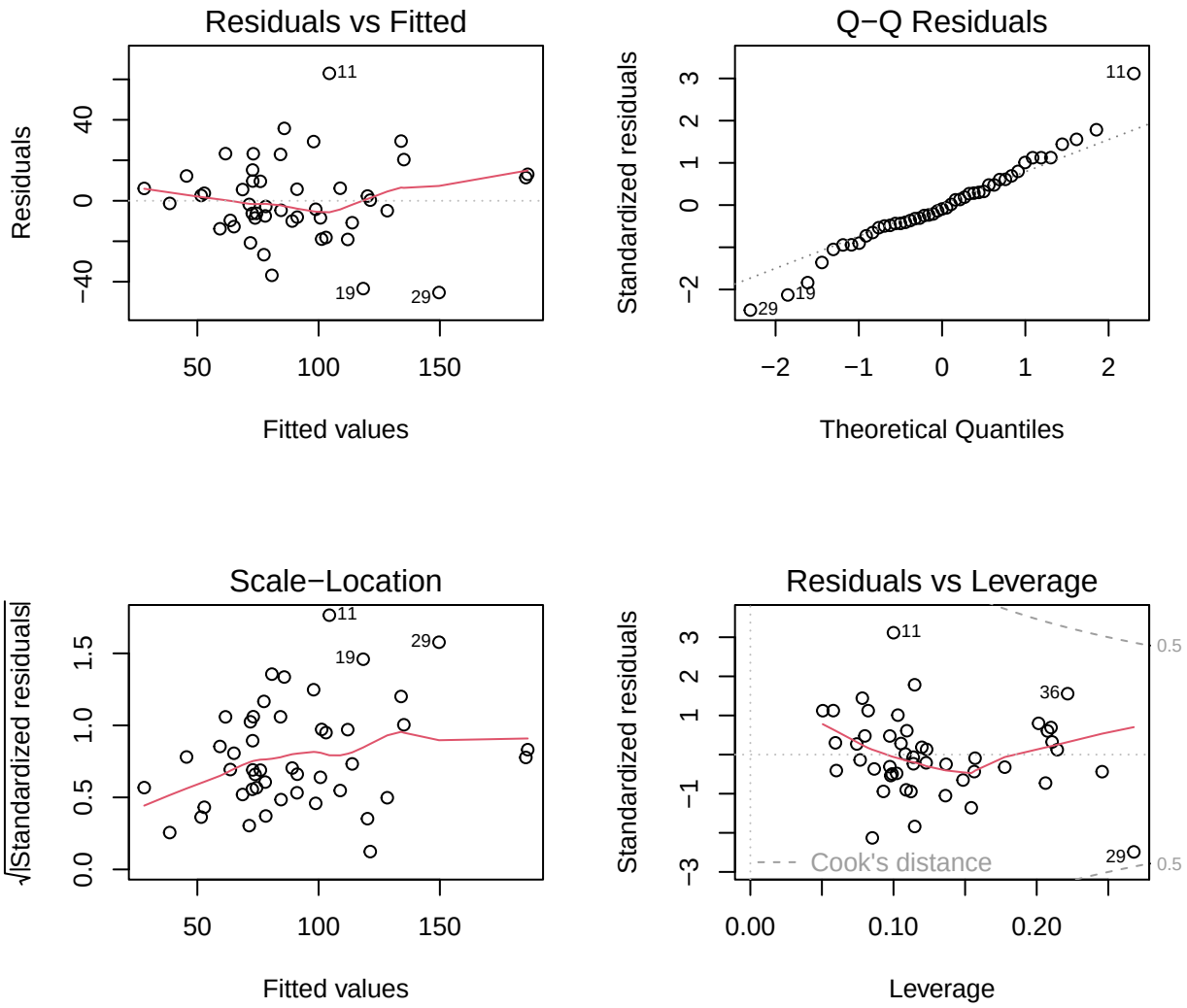
```
##     Min      1Q  Median      3Q     Max
## -45.344  -9.859  -1.807  10.603  62.964
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -524.3743    95.1156  -5.513 2.13e-06 ***
## Age            1.0198     0.3532   2.887 0.006175 **
## Ed             2.0308     0.4742   4.283 0.000109 ***
## Ex0            1.2331     0.1416   8.706 7.26e-11 ***
## U2             0.9136     0.4341   2.105 0.041496 *
## X              0.6349     0.1468   4.324 9.56e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21.3 on 41 degrees of freedom
## Multiple R-squared:  0.7296, Adjusted R-squared:  0.6967
## F-statistic: 22.13 on 5 and 41 DF,  p-value: 1.105e-10
```

Now, vif values are all below 4 and all regressors are significant, so this is the minimal model. The equation for the model is

$$R = 1.02 * \texttt{Age} + 3.031 * \texttt{Ed} + 1.233 * \texttt{Ex0} + 0.914 * \texttt{U2} + 0.635 * \texttt{X}$$

(d) Check the validity of the model assumptions starting with diagnostics plots and carry out any tests that are necessary. **Comment all your steps.**

```
par(mfrow = c(2,2))
plot(modelAIC2)
```




```
par(mfrow=c(1,1))
```

All the plots look reasonable. In the first plot, the distribution of the residuals looks random and approximately symmetric. The quantile plot shows some departures at the tails, but in general seems reasonable. We can confirm this using the Shapiro-Wilk test on the standardized residuals:

```
shapiro.test(rstandard(modelAIC2))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  rstandard(modelAIC2)
## W = 0.97494, p-value = 0.403
```

The third plot also looks reasonable although a slight increasing pattern can be seen in the local regression line. To confirm whether this is significant, we use the ncv test

```
ncvTest(modelAIC2)
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 3.758246, Df = 1, p = 0.052548
```

Since the p-value is above 0.05 threshold, we conclude that there is no heteroscedasticity. Finally, the fourth

plot shows one point with high leverage and large value for Cook's distance (close to the contour line), which is point 29. This point should be checked in a more thorough study of the regression model.