Chapter 3

A Taste of Model Selection for Multiple Regression

3.1 Model

Given data on a response variable Y and k predictor variables $X_1, X_2, \ldots, X_k$, we wish to develop a regression model to predict Y. Assuming that the collection of variables is measured on the correct scale, and that the candidate list of predictors includes all the important predictors, the most general model is

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_k X_k + \varepsilon.$$

In most problems one or more of the predictors can be eliminated from this general or **full model** without (much) loss of information. We want to identify the important predictors, or equivalently, eliminate the predictors that are not very useful for explaining the variation in Y (conditional on the other predictors in the model).

We will study several **automated** methods for model selection, which, given a specific criterion for selecting a model, gives the best predictors. Before applying any of the methods, you should plot Y against each predictor

$X_1, X_2, \ldots, X_k$ to see whether transformations are needed. If a transformation of X_i is suggested, include the transformation along with the original X_i in the candidate list. Note that you can transform the predictors differently, for example, $\log(X_1)$ and $\sqrt{X_2}$. However, if several transformations are suggested for the response, then you should consider doing one analysis for each suggested response scale before deciding on the final scale.

At this point, I will only consider the **backward elimination method**. Other approaches will be addressed later this semester.

3.2 Backward Elimination

The backward elimination procedure deletes unimportant variables, one at a time, starting from the full model. The steps is the procedure are:

1. Fit the full model

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_k X_k + \varepsilon.$$

2. Find the variable which when omitted from the full model (1) reduces R^2 the least, or equivalently, increases the Residual SS the least. This is the variable that gives the largest p-value for testing an individual regression coefficient $H_0 : \beta_i = 0$ for $i > 0$. Suppose this variable is X_k. If you reject H_0, stop and conclude that the full model is best. If you do not reject H_0, delete X_k from the full model, giving the **new full model**

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_{k-1} X_{k-1} + \varepsilon.$$

   Repeat steps 1 and 2 sequentially until no further predictors can be deleted.

In backward elimination we isolate the least important predictor left in the model, and check whether it is important. If not, delete it and repeat the process. Otherwise, stop. A 0.10 significance level is common to use for this strategy.

Epidemiologists use a slightly different approach to building models. They argue strongly for the need to always include **confounding** variables in a model, regardless of their statistical significance. I will briefly discuss this issue, but you should recognize that there is no universally accepted best approach to building models. A related issue is that several sets of predictors might give nearly identical fits and predictions to those obtained using any model selection method. This should not be too surprising because predictors are often correlated with each other. However, this should make us question whether one could ever completely unravel which variables are important (and which are not) for predicting a response.

3.2.1 Maximum likelihood and AIC/BIC

The **Akaike information criterion (AIC)** and **Bayesian information criterion (BIC)** are related penalized-likelihood criteria of the relative goodness-of-fit of a statistical model to the observed data. For model selection, a parsimonious model minimizes (one of) these quantities, where the penalty term is larger in BIC ($k\ln(n)$) than in AIC ($2k$). They are defined as

$$\begin{aligned} \text{AIC} &= -2\ln(L) + 2k \qquad \text{and} \\ \text{BIC} &= -2\ln(L) + k\ln(n) \end{aligned}$$

where n is the number of observations, k is the number of model parameters, and L is the maximized value of the likelihood function for the estimated model.

Maximum-likelihood estimation (MLE) applied to a data set and given a statistical model, estimates the model's parameters (βs and σ^2 in regression). MLE finds the particular parametric values that make the observed data the most probable given the model. That is, it selects the set of values of the model parameters that maximizes the likelihood function.

In practice, start with a set of candidate models, and then find the models' corresponding AIC/BIC values. There will almost always be information lost due to using one of the candidate models to represent the "true" (unknown) model. We choose the model that minimizes the (estimated) information loss

(the Kullback-Leibler divergence of the "true" unknown model represented with a candidate model).

The penalty discourages overfitting. Increasing the number of free parameters in the model will always improve the goodness-of-fit, regardless of the number of free parameters in the data-generating process. In the spirit of Occam's razor, the principle of parsimony, economy, or succinctness, the penalty helps balance the complexity of the model (low) with its ability to describe the data (high).

There are many methods for model selection. AIC or BIC are good tools for helping to choose among candidate models. A model selected by BIC will tend to have fewer parameters than one selected by AIC. Ultimately, you have to choose a model. I think of automated model selection as a starting point among the models I ultimately consider, and I may decide upon a different model than AIC, BIC, or another method.

3.3 Example: Peru Indian blood pressure

I will illustrate backward elimination on the Peru Indian data, using systolic blood pressure (`sysbp`) as the response, and seven candidate predictors: `wt` = weight in kilos; `ht` = height in mm; `chin` = chin skin fold in mm; `fore` = forearm skin fold in mm; `calf` = calf skin fold in mm; `pulse` = pulse rate-beats/min, and `yrage` = fraction.

The program given below generates simple summary statistics and plots. The plots do not suggest any apparent transformations of the response or the predictors, so we will analyze the data using the given scales. The correlations between the response and each potential predictor indicate that predictors are generally not highly correlated with each other (a few are).

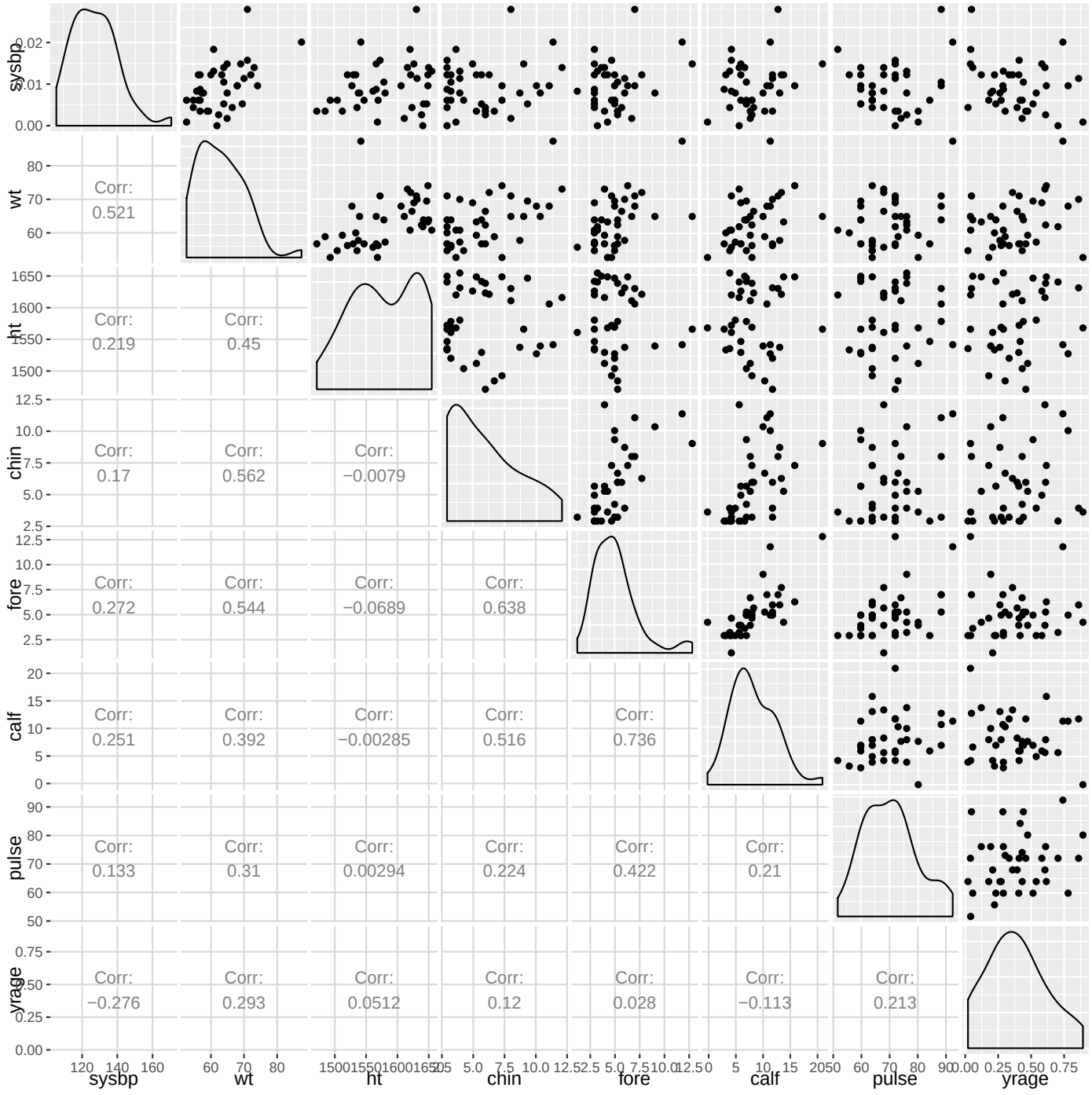
```
#### Example: Indian
# filename
fn.data <- "http://statacumen.com/teach/ADA2/ADA2_notes_Ch02_indian.dat"
indian <- read.table(fn.data, header=TRUE)

# Create the "fraction of their life" variable
```

```
#   yrage = years since migration divided by age
indian$yrage <- indian$yrmig / indian$age

# subset of variables we want in our model
indian2 <- subset(indian, select=c("sysbp", "wt", "ht", "chin"
                                   , "fore", "calf", "pulse", "yrage")
                  )

str(indian2)
```

```
## 'data.frame': 39 obs. of  8 variables:
##  $ sysbp: int  170 120 125 148 140 106 120 108 124 134 ...
##  $ wt   : num  71 56.5 56 61 65 62 53 53 65 57 ...
##  $ ht   : int  1629 1569 1561 1619 1566 1639 1494 1568 1540 1530 ...
##  $ chin : num  8 3.3 3.3 3.7 9 3 7.3 3.7 10.3 5.7 ...
##  $ fore : num  7 5 1.3 3 12.7 3.3 4.7 4.3 9 4 ...
##  $ calf : num  12.7 8 4.3 4.3 20.7 5.7 8 0 10 6 ...
##  $ pulse: int  88 64 68 52 72 72 64 80 76 60 ...
##  $ yrage: num  0.0476 0.2727 0.2083 0.0417 0.04 ...
```

```
# Description of variables
#    id = individual id
#   age = age in years            yrmig = years since migration
#    wt = weight in kilos            ht = height in mm
#  chin = chin skin fold in mm     fore = forearm skin fold in mm
#  calf = calf skin fold in mm    pulse = pulse rate-beats/min
# sysbp = systolic bp             diabp = diastolic bp

## print dataset to screen
#indian2
```

```
library(ggplot2)
#suppressMessages(suppressWarnings(library(GGally)))
library(GGally)
#p <- ggpairs(indian2)
# put scatterplots on top so y axis is vertical
p <- ggpairs(indian2, upper = list(continuous = "points")
                    , lower = list(continuous = "cor")
             )
print(p)

  # detach package after use so reshape2 works (old reshape (v.1) conflicts)
  #detach("package:GGally", unload=TRUE)
  #detach("package:reshape", unload=TRUE)
```




```
# correlation matrix and associated p-values testing "H0: rho == 0"
library(Hmisc)
rcorr(as.matrix(indian2))
```

```
##       sysbp   wt    ht  chin  fore  calf pulse yrage
## sysbp  1.00 0.52  0.22  0.17  0.27  0.25  0.13 -0.28
## wt     0.52 1.00  0.45  0.56  0.54  0.39  0.31  0.29
## ht     0.22 0.45  1.00 -0.01 -0.07  0.00  0.00  0.05
## chin   0.17 0.56 -0.01  1.00  0.64  0.52  0.22  0.12
## fore   0.27 0.54 -0.07  0.64  1.00  0.74  0.42  0.03
## calf   0.25 0.39  0.00  0.52  0.74  1.00  0.21 -0.11
```

```
## pulse  0.13 0.31  0.00  0.22  0.42  0.21  1.00  0.21
## yrage -0.28 0.29  0.05  0.12  0.03 -0.11  0.21  1.00
##
## n= 39
##
##
## P
##       sysbp  wt     ht     chin   fore   calf   pulse  yrage
## sysbp        0.0007 0.1802 0.3003 0.0936 0.1236 0.4211 0.0888
## wt    0.0007        0.0040 0.0002 0.0003 0.0136 0.0548 0.0702
## ht    0.1802 0.0040        0.9619 0.6767 0.9863 0.9858 0.7570
## chin  0.3003 0.0002 0.9619        0.0000 0.0008 0.1708 0.4665
## fore  0.0936 0.0003 0.6767 0.0000        0.0000 0.0075 0.8656
## calf  0.1236 0.0136 0.9863 0.0008 0.0000        0.1995 0.4933
## pulse 0.4211 0.0548 0.9858 0.1708 0.0075 0.1995        0.1928
## yrage 0.0888 0.0702 0.7570 0.4665 0.8656 0.4933 0.1928
```

Below I fit the linear model with all the selected main effects.

```
# fit full model
lm.indian2.full <- lm(sysbp ~ wt + ht + chin + fore + calf + pulse + yrage
                      , data = indian2)

library(car)
Anova(lm.indian2.full, type=3)
```

```
## Anova Table (Type III tests)
##
## Response: sysbp
##              Sum Sq Df F value    Pr(>F)
## (Intercept)  389.46  1  3.8991 0.0572767 .
## wt          1956.49  1 19.5874 0.0001105 ***
## ht           131.88  1  1.3203 0.2593289
## chin         186.85  1  1.8706 0.1812390
## fore          27.00  1  0.2703 0.6068061
## calf           2.86  1  0.0287 0.8666427
## pulse         14.61  1  0.1463 0.7046990
## yrage       1386.76  1 13.8835 0.0007773 ***
## Residuals   3096.45 31
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(lm.indian2.full)
```

```
##
## Call:
## lm(formula = sysbp ~ wt + ht + chin + fore + calf + pulse + yrage,
##     data = indian2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.3993  -5.7916  -0.6907   6.9453  23.5771
```

```
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 106.45766   53.91303   1.975 0.057277 .
## wt            1.71095    0.38659   4.426 0.000111 ***
## ht           -0.04533    0.03945  -1.149 0.259329
## chin         -1.15725    0.84612  -1.368 0.181239
## fore         -0.70183    1.34986  -0.520 0.606806
## calf          0.10357    0.61170   0.169 0.866643
## pulse         0.07485    0.19570   0.383 0.704699
## yrage       -29.31810    7.86839  -3.726 0.000777 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.994 on 31 degrees of freedom
## Multiple R-squared:  0.5259,Adjusted R-squared:  0.4189
## F-statistic: 4.913 on 7 and 31 DF,  p-value: 0.0008079
```

Remarks on Step 0: The full model has 7 predictors so REG df = 7. The F-test in the full model ANOVA table ($F = 4.91$ with p-value = 0.0008) tests the hypothesis that the regression coefficient for each predictor variable is zero. This test is highly significant, indicating that one or more of the predictors is important in the model.

In the ANOVA table, the F-value column gives the square of the t-statistic (from the parameter [Coefficients] estimate table) for testing the significance of the individual predictors in the full model (conditional on all other predictors being in the model). The p-value is the same whether the t-statistic or F-value is shown.

The least important variable in the full model, as judged by the p-value, is calf skin fold. This variable, upon omission, reduces R^2 the least, or equivalently, increases the Residual SS the least. The p-value of 0.87 exceeds the default 0.10 cut-off, so $calf$ will be the first to be omitted from the model.

Below, we will continue in this way. After deleting `calf`, the six predictor model can be fitted. Manually, you can find that at least one of the predictors left is important, as judged by the overall F-test p-value. The least important predictor left is `pulse`. This variable is omitted from the model because the p-value for including it exceeds the 0.10 threshold.

This is repeated until all predictors remain significant at a 0.10 significance level.

```
# model reduction using update() and subtracting (removing) model terms
lm.indian2.red <- lm.indian2.full;

# remove calf
lm.indian2.red <- update(lm.indian2.red, ~ . - calf ); summary(lm.indian2.red);
##
## Call:
## lm(formula = sysbp ~ wt + ht + chin + fore + pulse + yrage, data = indian2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.6993  -5.3152  -0.7725   7.2966  23.7240
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 106.13739   53.05581   2.000 0.053993 .
## wt            1.70900    0.38051   4.491 8.65e-05 ***
## ht           -0.04478    0.03871  -1.157 0.256008
## chin         -1.14165    0.82823  -1.378 0.177635
## fore         -0.56731    1.07462  -0.528 0.601197
## pulse         0.07103    0.19142   0.371 0.713018
## yrage       -29.54000    7.63983  -3.867 0.000509 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.841 on 32 degrees of freedom
## Multiple R-squared:  0.5255,Adjusted R-squared:  0.4365
## F-statistic: 5.906 on 6 and 32 DF,  p-value: 0.0003103
# remove pulse
lm.indian2.red <- update(lm.indian2.red, ~ . - pulse); summary(lm.indian2.red);
##
## Call:
## lm(formula = sysbp ~ wt + ht + chin + fore + yrage, data = indian2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.6147  -5.9803  -0.2065   6.6755  24.9269
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 110.27872   51.18665   2.154 0.038601 *
## wt            1.71825    0.37470   4.586 6.22e-05 ***
## ht           -0.04504    0.03820  -1.179 0.246810
## chin         -1.17716    0.81187  -1.450 0.156514
## fore         -0.43385    0.99933  -0.434 0.667013
```

```
## yrage       -28.98171    7.39172  -3.921 0.000421 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.712 on 33 degrees of freedom
## Multiple R-squared:  0.5234,Adjusted R-squared:  0.4512
## F-statistic: 7.249 on 5 and 33 DF,  p-value: 0.0001124
```

```
# remove fore
lm.indian2.red <- update(lm.indian2.red, ~ . - fore ); summary(lm.indian2.red);
```

```
##
## Call:
## lm(formula = sysbp ~ wt + ht + chin + yrage, data = indian2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.1030  -6.3484   0.2834   6.7766  24.8883
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 104.52292   48.84627   2.140 0.039629 *
## wt            1.64631    0.33203   4.958 1.94e-05 ***
## ht           -0.03957    0.03563  -1.111 0.274530
## chin         -1.31083    0.74220  -1.766 0.086348 .
## yrage       -28.32580    7.14879  -3.962 0.000361 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.595 on 34 degrees of freedom
## Multiple R-squared:  0.5207,Adjusted R-squared:  0.4643
## F-statistic: 9.235 on 4 and 34 DF,  p-value: 3.661e-05
```

```
# remove ht
lm.indian2.red <- update(lm.indian2.red, ~ . - ht   ); summary(lm.indian2.red);
```

```
##
## Call:
## lm(formula = sysbp ~ wt + chin + yrage, data = indian2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -16.6382  -6.6316   0.4521   6.3593  24.2086
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  52.9092    15.0895   3.506 0.001266 **
## wt            1.4407     0.2766   5.209 8.51e-06 ***
## chin         -1.0135     0.6945  -1.459 0.153407
## yrage       -27.3522     7.1185  -3.842 0.000491 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.627 on 35 degrees of freedom
## Multiple R-squared:  0.5033,Adjusted R-squared:  0.4608
## F-statistic: 11.82 on 3 and 35 DF,  p-value: 1.684e-05
```

```
# remove chin
lm.indian2.red <- update(lm.indian2.red, ~ . - chin ); summary(lm.indian2.red);
```

```
##
## Call:
## lm(formula = sysbp ~ wt + yrage, data = indian2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.4330  -7.3070   0.8963   5.7275  23.9819
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  60.8959    14.2809   4.264 0.000138 ***
## wt            1.2169     0.2337   5.207 7.97e-06 ***
## yrage       -26.7672     7.2178  -3.708 0.000699 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.777 on 36 degrees of freedom
## Multiple R-squared:  0.4731,Adjusted R-squared:  0.4438
## F-statistic: 16.16 on 2 and 36 DF,  p-value: 9.795e-06
```

```
# all are significant, stop.
# final model: sysbp ~ wt + yrage
lm.indian2.final <- lm.indian2.red
```

AIC/BIC automated model selection The AIC/BIC strategy is more commonly used for model selection, though resulting models are usually the same as the method described above. I use the `step()` function to perform backward selection using the AIC criterion (and give code for the BIC) then make some last-step decisions. Note that because the BIC has a larger penalty, it arrives at my chosen model directly.

At each step, the predictors are ranked (least significant to most significant) and then a decision of whether to keep the top predictor is made. `<none>` represents the current model.

```
## AIC
# option: test="F" includes additional information
#           for parameter estimate tests that we're familiar with
```

```
# option: for BIC, include k=log(nrow( [data.frame name] ))
lm.indian2.red.AIC <- step(lm.indian2.full, direction="backward", test="F")

## Start:  AIC=186.6
## sysbp ~ wt + ht + chin + fore + calf + pulse + yrage
##
##         Df Sum of Sq    RSS    AIC F value    Pr(>F)
## - calf   1      2.86 3099.3 184.64  0.0287 0.8666427
## - pulse  1     14.61 3111.1 184.79  0.1463 0.7046990
## - fore   1     27.00 3123.4 184.94  0.2703 0.6068061
## - ht     1    131.88 3228.3 186.23  1.3203 0.2593289
## <none>               3096.4 186.60
## - chin   1    186.85 3283.3 186.89  1.8706 0.1812390
## - yrage  1   1386.76 4483.2 199.04 13.8835 0.0007773 ***
## - wt     1   1956.49 5052.9 203.70 19.5874 0.0001105 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=184.64
## sysbp ~ wt + ht + chin + fore + pulse + yrage
##
##         Df Sum of Sq    RSS    AIC F value    Pr(>F)
## - pulse  1     13.34 3112.6 182.81  0.1377 0.7130185
## - fore   1     26.99 3126.3 182.98  0.2787 0.6011969
## - ht     1    129.56 3228.9 184.24  1.3377 0.2560083
## <none>               3099.3 184.64
## - chin   1    184.03 3283.3 184.89  1.9000 0.1776352
## - yrage  1   1448.00 4547.3 197.59 14.9504 0.0005087 ***
## - wt     1   1953.77 5053.1 201.70 20.1724 8.655e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=182.81
## sysbp ~ wt + ht + chin + fore + yrage
##
##         Df Sum of Sq    RSS    AIC F value    Pr(>F)
## - fore   1     17.78 3130.4 181.03  0.1885  0.667013
## - ht     1    131.12 3243.8 182.42  1.3902  0.246810
## <none>               3112.6 182.81
## - chin   1    198.30 3310.9 183.22  2.1023  0.156514
## - yrage  1   1450.02 4562.7 195.72 15.3730  0.000421 ***
## - wt     1   1983.51 5096.2 200.03 21.0290 6.219e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=181.03
## sysbp ~ wt + ht + chin + yrage
##
##         Df Sum of Sq    RSS    AIC F value    Pr(>F)
```

```
## - ht     1    113.57 3244.0 180.42  1.2334 0.2745301
## <none>               3130.4 181.03
## - chin   1    287.20 3417.6 182.45  3.1193 0.0863479 .
## - yrage  1   1445.52 4575.9 193.84 15.7000 0.0003607 ***
## - wt     1   2263.64 5394.1 200.25 24.5857 1.945e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=180.42
## sysbp ~ wt + chin + yrage
##
##         Df Sum of Sq    RSS    AIC F value    Pr(>F)
## <none>               3244.0 180.42
## - chin   1    197.37 3441.4 180.72  2.1295 0.1534065
## - yrage  1   1368.44 4612.4 192.15 14.7643 0.0004912 ***
## - wt     1   2515.33 5759.3 200.81 27.1384 8.512e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
# BIC (not shown)
# step(lm.indian2.full, direction="backward", test="F", k=log(nrow(indian2)))
```

Remark on Summary Table: The partial R^2 is the reduction in R^2 achieved by omitting variables sequentially.

The backward elimination procedure eliminates five variables from the full model, in the following order: calf skin fold `calf`, pulse rate `pulse`, forearm skin fold `fore`, height `ht`, and chin skin fold `chin`. The model selected by backward elimination includes two predictors: weight `wt` and fraction `yrage`. As we progress from the full model to the selected model, R^2 decreases as follows: 0.53, 0.53, 0.52, 0.52, 0.50, and 0.47. The decrease is slight across this spectrum of models.

Using a mechanical approach, we are led to a model with weight and years by age fraction as predictors of systolic blood pressure. At this point we should closely examine this model.

3.3.1 Analysis for Selected Model

The summaries and diagnostics for the selected model follow.

Model sysbp $= \beta_0 + \beta_1$ wt $+ \beta_2$ yrage $+ \varepsilon$:

```
library(car)
Anova(lm.indian2.final, type=3)
```

```
## Anova Table (Type III tests)
##
## Response: sysbp
##             Sum Sq Df F value    Pr(>F)
## (Intercept) 1738.2  1  18.183 0.0001385 ***
## wt          2592.0  1  27.115 7.966e-06 ***
## yrage       1314.7  1  13.753 0.0006991 ***
## Residuals   3441.4 36
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(lm.indian2.final)
```

```
##
## Call:
## lm(formula = sysbp ~ wt + yrage, data = indian2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.4330  -7.3070   0.8963   5.7275  23.9819
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  60.8959    14.2809   4.264 0.000138 ***
## wt            1.2169     0.2337   5.207 7.97e-06 ***
## yrage       -26.7672     7.2178  -3.708 0.000699 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.777 on 36 degrees of freedom
## Multiple R-squared:  0.4731,Adjusted R-squared:  0.4438
## F-statistic: 16.16 on 2 and 36 DF,  p-value: 9.795e-06
```

Comments on the diagnostic plots below.

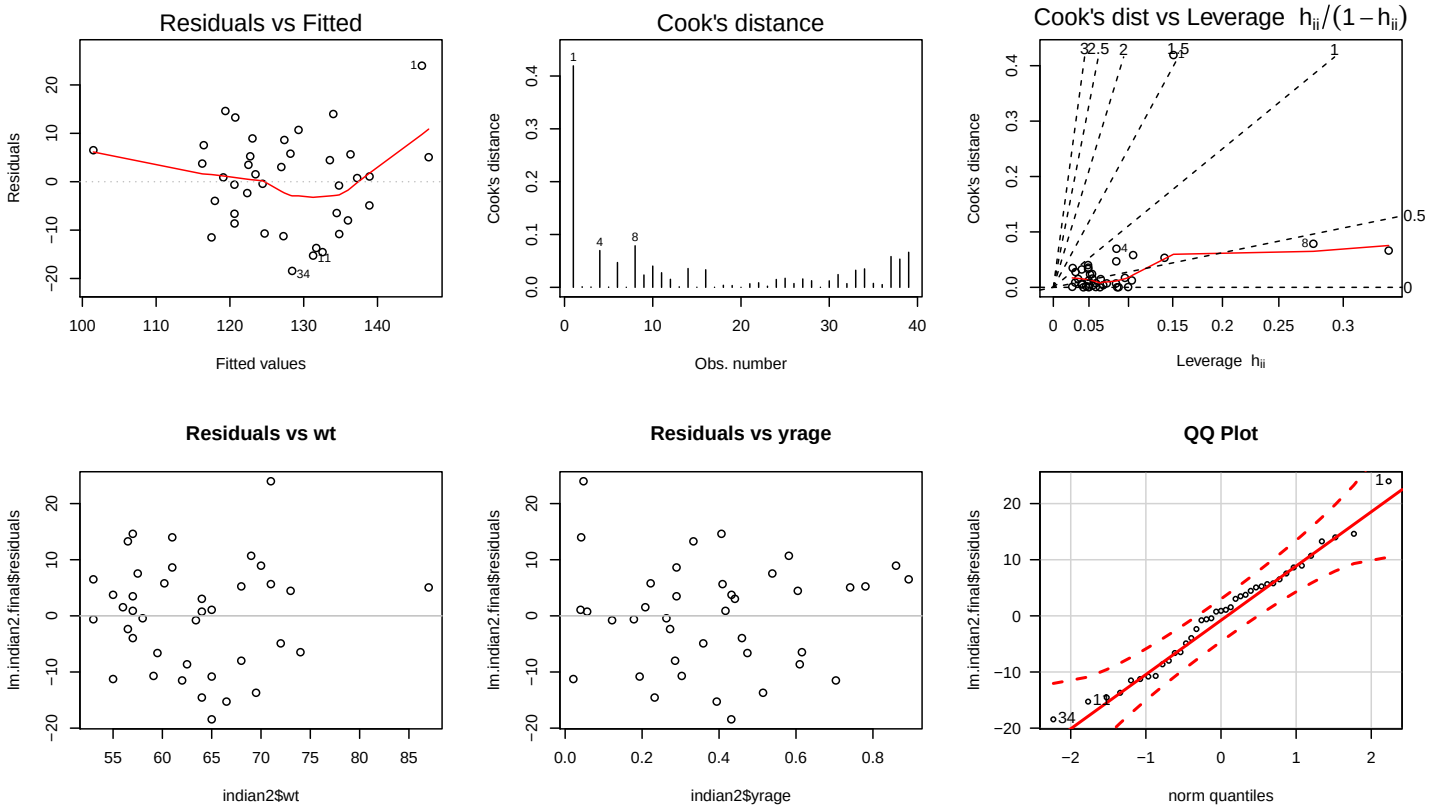
1. The individual with the highest systolic blood pressure (case 1) has a large studentized residual r_i and the largest Cook's D_i.
2. Except for case 1, the rankit plot and the plot of the studentized residuals against the fitted values show no gross abnormalities.
3. The plots of studentized residuals against the individual predictors show no patterns. The partial residual plots show roughly linear trends. These plots collectively do not suggest the need to transform either of the predictors. Although case 1 is prominent in the partial residual plots, it does

not appear to be influencing the significance of these predictors.

```
# plot diagnistics
par(mfrow=c(2,3))
plot(lm.indian2.final, which = c(1,4,6))

plot(indian2$wt, lm.indian2.final$residuals, main="Residuals vs wt")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(indian2$yrage, lm.indian2.final$residuals, main="Residuals vs yrage")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# Normality of Residuals
library(car)
qqPlot(lm.indian2.final$residuals, las = 1, id.n = 3, main="QQ Plot")
```

```
##  1 34 11
## 39  1  2
```

```
## residuals vs order of data
#plot(lm.indian2.final$residuals, main="Residuals vs Order of data")
#  # horizontal line at zero
#  abline(h = 0, col = "gray75")
```

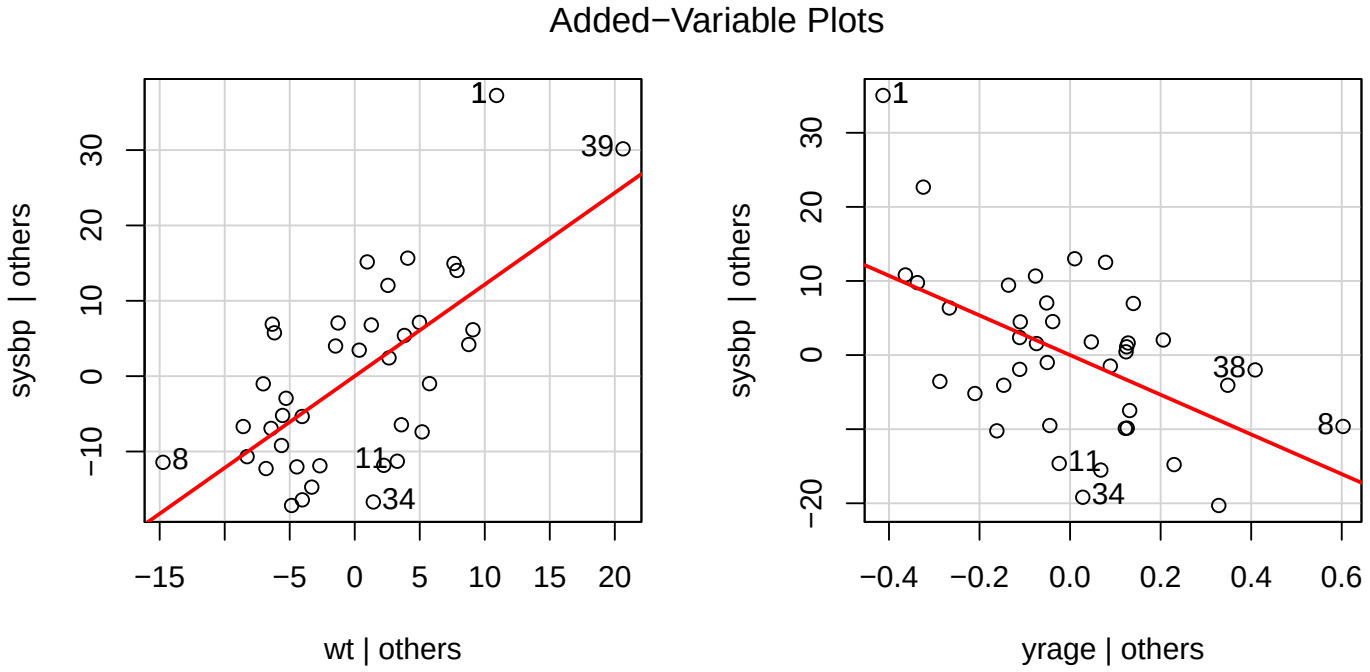


Recall that the partial regression residual plot for weight, given below, **adjusts** systolic blood pressure and weight for their common dependence on all the other predictors in the model (only years by age fraction here). This plot tells us whether we need to transform weight in the multiple regression model, and whether any observations are influencing the significance of weight in the fitted model. A roughly linear trend, as seen here, suggests that no transfor-

mation of weight is warranted. The positive relationship seen here is consistent with the coefficient of weight being positive in the multiple regression model.

The partial residual plot for fraction exhibits a stronger relationship than is seen in the earlier 2D plot of systolic blood pressure against year by age fraction. This means that fraction is more useful as a predictor after taking an individual's weight into consideration.

```
library(car)
avPlots(lm.indian2.final, id.n=3)
```



Model selection methods can be highly influenced by outliers and influential cases. We should hold out case 1, and rerun the backward procedure to see whether case 1 unduly influenced the selection of the two predictor model. If we hold out case 1, we find that the model with weight and fraction as predictors is suggested again. After holding out case 1, there are no large residuals, no extremely influential points, or any gross abnormalities in plots. The R^2 for the selected model is now $R^2 = 0.408$. This decrease in R^2 should have been anticipated. Why?[1]

The two analyses suggest that the "best model" for predicting systolic blood

[1]Obs 1 increases the SST, but greatly increases model relationship so greatly increases SSR.

pressure is

$$\text{sysbp} \;=\; \beta_0 + \beta_1\ \text{wt} + \beta_2\ \text{yrage} + \varepsilon.$$

Should case 1 be deleted? I have not fully explored this issue, but I will note that eliminating this case does have a significant impact on the estimates of the regression coefficients, and on predicted values. What do you think?

3.4 Example: Dennis Cook's Rat Data

This example illustrates the importance of a careful diagnostic analysis.

An experiment was conducted to investigate the amount of a particular drug present in the liver of a rat. Nineteen (19) rats were randomly selected, weighed, placed under light ether anesthesia and given an oral dose of the drug. Because it was felt that large livers would absorb more of a given dose than small livers, the actual dose an animal received was approximately determined as 40mg of the drug per kilogram of body weight. (Liver weight is known to be strongly related to body weight.) After a fixed length of time, each rat was sacrificed, the liver weighed, and the percent of the dose in the liver determined.

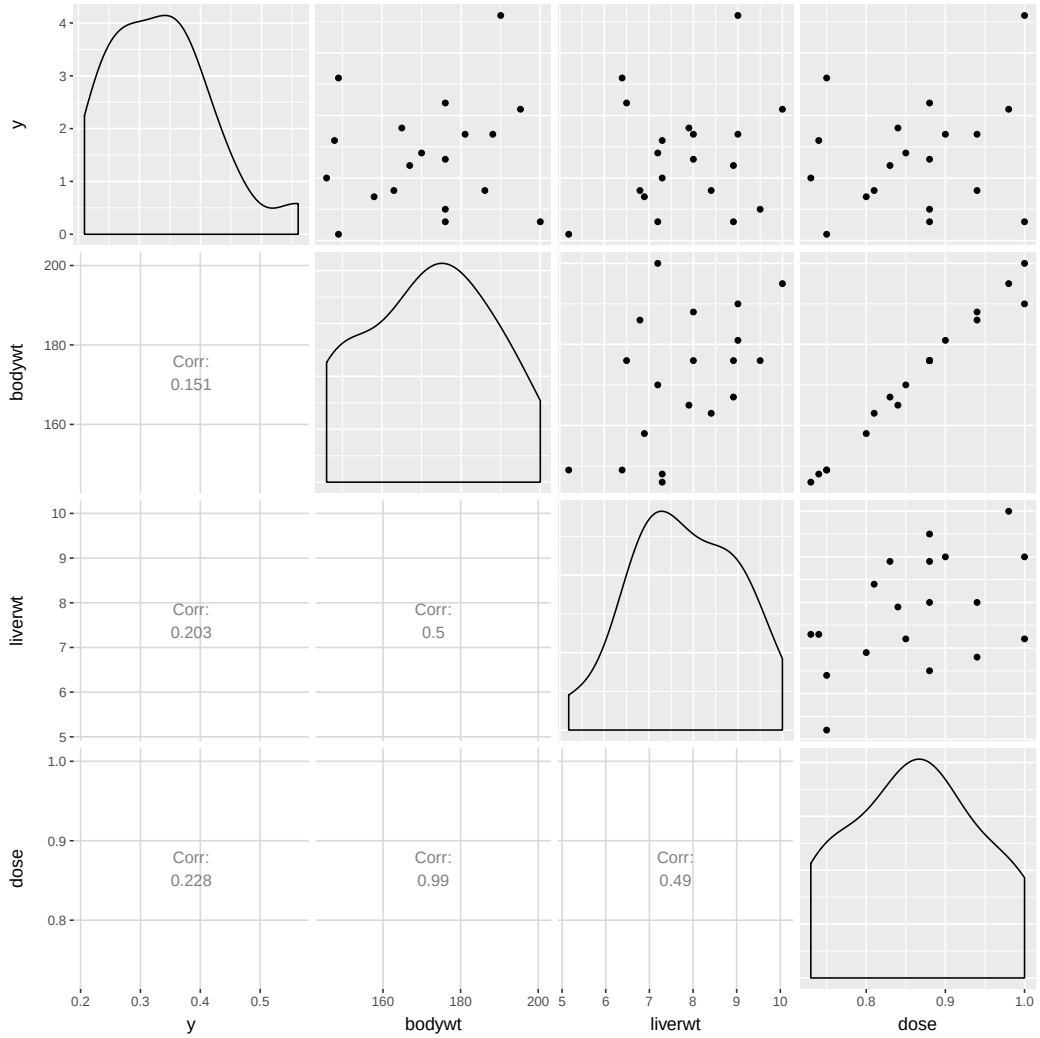
The experimental hypothesis was that, for the method of determining the dose, there is no relationship between the percentage of dose in the liver (Y) and the body weight, liver weight, and relative dose.

```
#### Example: Rat liver
fn.data <- "http://statacumen.com/teach/ADA2/ADA2_notes_Ch03_ratliver.csv"
ratliver <- read.csv(fn.data)

ratliver <- ratliver[,c(4,1,2,3)] # reorder columns so response is the first

str(ratliver)
```

```
## 'data.frame': 19 obs. of  4 variables:
##  $ y      : num  0.42 0.25 0.56 0.23 0.23 0.32 0.37 0.41 0.33 0.38 ...
##  $ bodywt : int  176 176 190 176 200 167 188 195 176 165 ...
##  $ liverwt: num  6.5 9.5 9 8.9 7.2 8.9 8 10 8 7.9 ...
##  $ dose   : num  0.88 0.88 1 0.88 1 0.83 0.94 0.98 0.88 0.84 ...
```

	y	bodywt	liverwt	dose
1	0.42	176	6.50	0.88
2	0.25	176	9.50	0.88
3	0.56	190	9.00	1.00
4	0.23	176	8.90	0.88
5	0.23	200	7.20	1.00
6	0.32	167	8.90	0.83
7	0.37	188	8.00	0.94
8	0.41	195	10.00	0.98
9	0.33	176	8.00	0.88
10	0.38	165	7.90	0.84
11	0.27	158	6.90	0.80
12	0.36	148	7.30	0.74
13	0.21	149	5.20	0.75
14	0.28	163	8.40	0.81
15	0.34	170	7.20	0.85
16	0.28	186	6.80	0.94
17	0.30	146	7.30	0.73
18	0.37	181	9.00	0.90
19	0.46	149	6.40	0.75

```
library(ggplot2)
#suppressMessages(suppressWarnings(library(GGally)))
library(GGally)
#p <- ggpairs(ratliver)
# put scatterplots on top so y axis is vertical
p <- ggpairs(ratliver, upper = list(continuous = "points")
                     , lower = list(continuous = "cor")
             )
print(p)

  # detach package after use so reshape2 works (old reshape (v.1) conflicts)
  #detach("package:GGally", unload=TRUE)
  #detach("package:reshape", unload=TRUE)
```



The correlation between Y and each predictor is small, as expected.

```
# correlation matrix and associated p-values testing "H0: rho == 0"
library(Hmisc)
rcorr(as.matrix(ratliver))
```

```
##            y bodywt liverwt dose
## y       1.00   0.15    0.20 0.23
## bodywt  0.15   1.00    0.50 0.99
## liverwt 0.20   0.50    1.00 0.49
## dose    0.23   0.99    0.49 1.00
##
## n= 19
##
##
## P
##         y      bodywt liverwt dose
## y              0.5370 0.4038  0.3488
## bodywt  0.5370        0.0293  0.0000
## liverwt 0.4038 0.0293         0.0332
## dose    0.3488 0.0000 0.0332
```

Below I fit the linear model with all the selected main effects.

```
# fit full model
lm.ratliver.full <- lm(y ~ bodywt + liverwt + dose, data = ratliver)

library(car)
Anova(lm.ratliver.full, type=3)
```

```
## Anova Table (Type III tests)
##
## Response: y
##               Sum Sq Df F value  Pr(>F)
## (Intercept) 0.011157  1  1.8676 0.19188
## bodywt      0.042408  1  7.0988 0.01768 *
## liverwt     0.004120  1  0.6897 0.41930
## dose        0.044982  1  7.5296 0.01507 *
## Residuals   0.089609 15
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(lm.ratliver.full)
```

```
##
## Call:
## lm(formula = y ~ bodywt + liverwt + dose, data = ratliver)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.100557 -0.063233  0.007131  0.045971  0.134691
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.265922   0.194585   1.367   0.1919
## bodywt      -0.021246   0.007974  -2.664   0.0177 *
## liverwt      0.014298   0.017217   0.830   0.4193
## dose         4.178111   1.522625   2.744   0.0151 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07729 on 15 degrees of freedom
## Multiple R-squared:  0.3639,Adjusted R-squared:  0.2367
## F-statistic:  2.86 on 3 and 15 DF,  p-value: 0.07197
```

The backward elimination procedure selects weight and dose as predictors. The p-values for testing the importance of these variables, when added last to this two predictor model, are small, 0.019 and 0.015.

```
lm.ratliver.red.AIC <- step(lm.ratliver.full, direction="backward", test="F")
```

```
## Start:  AIC=-93.78
## y ~ bodywt + liverwt + dose
##
##           Df Sum of Sq      RSS     AIC F value  Pr(>F)
```

```
## - liverwt  1  0.004120 0.093729 -94.924  0.6897 0.41930
## <none>                 0.089609 -93.778
## - bodywt   1  0.042408 0.132017 -88.416  7.0988 0.01768 *
## - dose     1  0.044982 0.134591 -88.049  7.5296 0.01507 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Step:  AIC=-94.92
## y ~ bodywt + dose
##
##          Df Sum of Sq      RSS     AIC F value  Pr(>F)
## <none>                0.093729 -94.924
## - bodywt  1  0.039851 0.133580 -90.192  6.8027 0.01902 *
## - dose    1  0.043929 0.137658 -89.621  7.4989 0.01458 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
lm.ratliver.final <- lm.ratliver.red.AIC
```

This cursory analysis leads to a conclusion that a combination of dose and body weight is associated with Y, but that neither of these predictors is important of its own (low correlations with Y). Although this commonly happens in regression problems, it is somewhat paradoxical here because dose was approximately a multiple of body weight, so to a first approximation, these predictors are linearly related and so only one of them should be needed in a linear regression model. Note that the correlation between dose and body weight is 0.99.

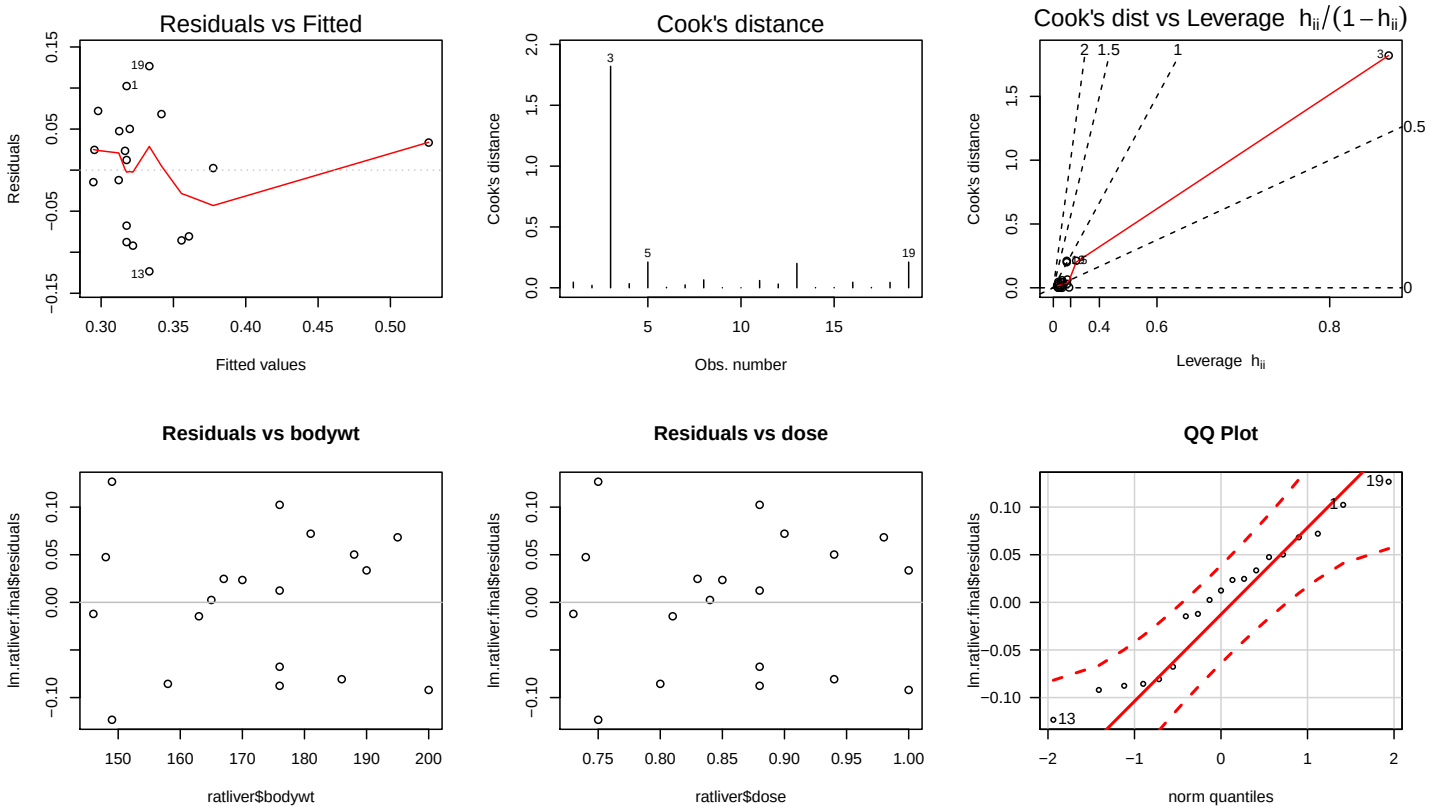
The apparent paradox can be resolved only with a careful diagnostic analysis! For the model with dose and body weight as predictors, there are no cases with large $|r_i|$ values, but case 3 has a relatively large Cook's D value.

```
# plot diagnistics
par(mfrow=c(2,3))
plot(lm.ratliver.final, which = c(1,4,6))

plot(ratliver$bodywt, lm.ratliver.final$residuals, main="Residuals vs bodywt")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

plot(ratliver$dose, lm.ratliver.final$residuals, main="Residuals vs dose")
  # horizontal line at zero
  abline(h = 0, col = "gray75")

# Normality of Residuals
library(car)
qqPlot(lm.ratliver.final$residuals, las = 1, id.n = 3, main="QQ Plot")
```

```
## 19 13  1
## 19  1 18
```

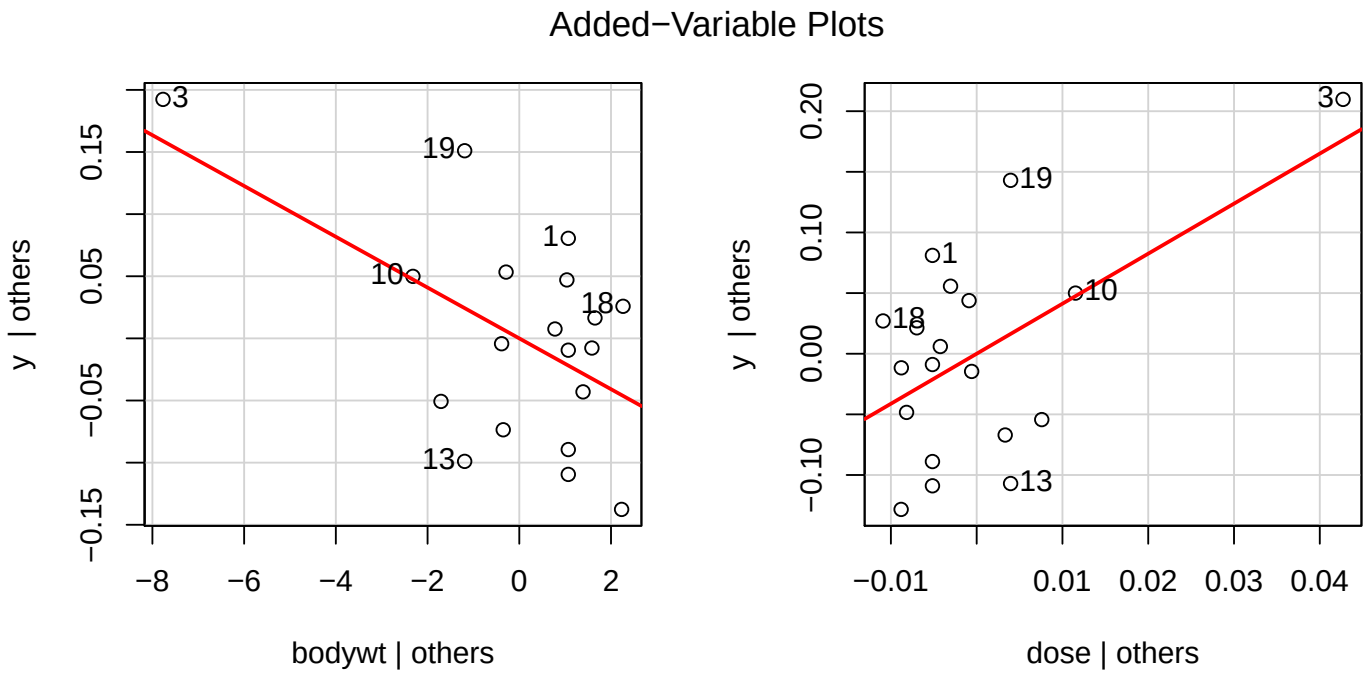
```
## residuals vs order of data
#plot(lm.ratliver.final$residuals, main="Residuals vs Order of data")
#  # horizontal line at zero
```

```
#  abline(h = 0, col = "gray75")
```



Further, the partial residual plot for `bodywt` clearly highlights case 3. Without this case we would see roughly a random scatter of points, suggesting that body weight is unimportant after taking dose into consideration. The importance of body weight as a predictor in the multiple regression model is due solely to the placement of case 3. The partial residual plot for `dose` gives the same message.

```
library(car)
avPlots(lm.ratliver.final, id.n=3)
```



Removing case 3 If we delete this case and redo the analysis we find, as expected, no important predictors of Y. The output below shows that the backward elimination removes each predictor from the model. Thus, the apparent relationship between Y and body weight and dose in the initial analysis can be ascribed to Case 3 alone. Can you see this case in the plots?

```
# remove case 3
ratliver3 <- ratliver[-3,]

# fit full model
lm.ratliver3.full <- lm(y ~ bodywt + liverwt + dose, data = ratliver3)

lm.ratliver3.red.AIC <- step(lm.ratliver3.full, direction="backward", test="F")
## Start:  AIC=-88.25
## y ~ bodywt + liverwt + dose
##
##           Df  Sum of Sq      RSS     AIC F value Pr(>F)
## - dose     1 0.00097916 0.086696 -90.043  0.1599 0.6953
## - bodywt   1 0.00105871 0.086776 -90.026  0.1729 0.6838
## - liverwt  1 0.00142114 0.087138 -89.951  0.2321 0.6374
## <none>                 0.085717 -88.247
##
## Step:  AIC=-90.04
## y ~ bodywt + liverwt
##
##           Df  Sum of Sq      RSS     AIC F value Pr(>F)
```

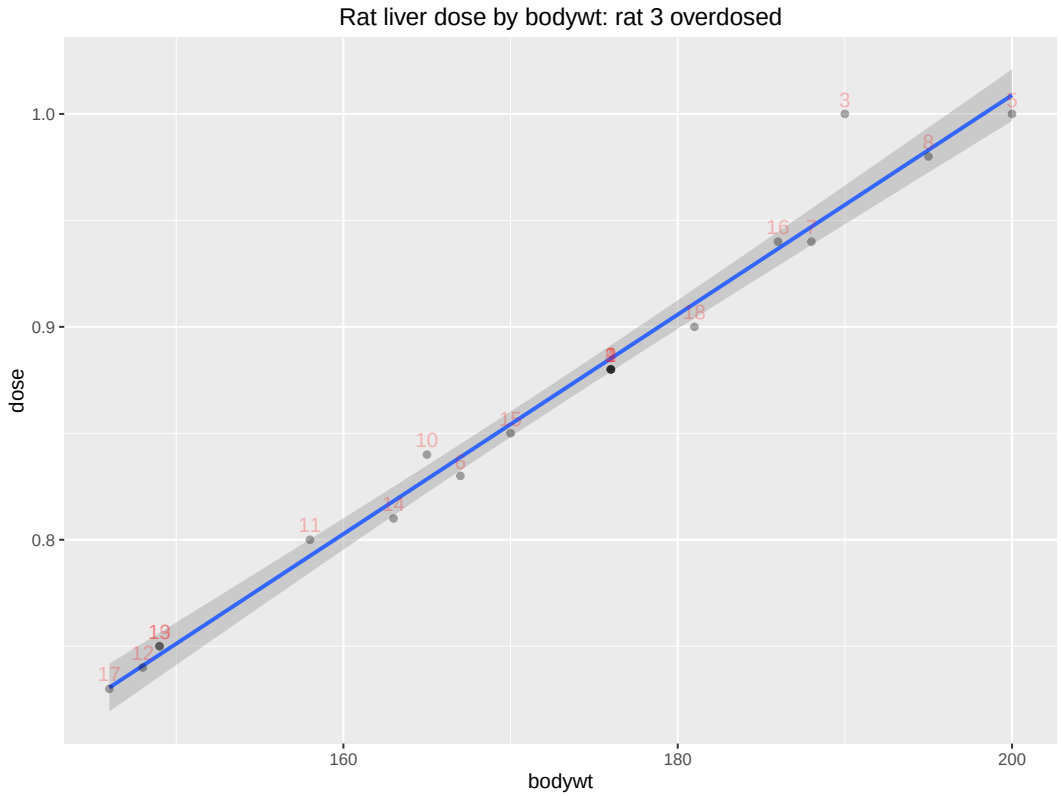
```
## - bodywt   1 0.00035562 0.087052 -91.969  0.0615 0.8075
## - liverwt  1 0.00082681 0.087523 -91.872  0.1431 0.7106
## <none>                  0.086696 -90.043
##
## Step:  AIC=-91.97
## y ~ liverwt
##
##           Df  Sum of Sq      RSS     AIC F value Pr(>F)
## - liverwt  1 0.00050917 0.087561 -93.864  0.0936 0.7636
## <none>                  0.087052 -91.969
##
## Step:  AIC=-93.86
## y ~ 1
```

All variables are omitted!

In his text[2], Weisberg writes: The careful analyst must now try to understand exactly why the third case is so influential. Inspection of the data indicates that this rat with weight 190g, was reported to have received a full dose of 1.000, which was a larger dose than it should have received according to the rule for assigning doses, see scatterplot below (e.g., rat 8 with a weight of 195g got a lower dose of 0.98).

```
# ggplot: Plot the data with linear regression fit and confidence bands
library(ggplot2)
p <- ggplot(ratliver, aes(x = bodywt, y = dose, label = 1:nrow(ratliver)))
# plot regression line and confidence band
p <- p + geom_smooth(method = lm)
p <- p + geom_point(alpha=1/3)
# plot labels next to points
p <- p + geom_text(hjust = 0.5, vjust = -0.5, alpha = 0.25, colour = 2)
p <- p + labs(title="Rat liver dose by bodywt: rat 3 overdosed")
print(p)
```

[2]Applied Linear Regression, 3rd Ed. by Sanford Weisberg, published by Wiley/Interscience in 2005 (ISBN 0-471-66379-4)



A number of causes for the result found in the first analysis are possible: (1) the dose or weight recorded for case 3 was in error, so the case should probably be deleted from the analysis, or (2) the regression fit in the second analysis is not appropriate except in the region defined by the 18 points excluding case 3. It is possible that the combination of dose and rat weight chosen was fortuitous, and that the lack of relationship found would not persist for any other combinations of them, since inclusion of a data point apparently taken under different conditions leads to a different conclusion. This suggests the need for collection of additional data, with dose determined by some rule other than a constant proportion of weight.

I hope the point of this analysis is clear! What have we learned from this analysis?