Multiple regression

October 4, 2017

This chapter discusses the case of regression analysis with multiple predictors. The basic model for multiple regression analysis is:

$$y = \beta_0 + \beta_1 x_1 + ... + \beta_k x_k + \epsilon$$

where $x_1, ..., x_k$ are explanatory variables (also called predictors) and the parameters $\beta_1, ..., \beta_k$ can be estimated using the method of least squares.

Plotting multivariate data

As an example, we use a study concerning lung function in patients with cystic fibrosis in Altman (1991, p. 338). The data are in the *cystfibr* data frame in the *ISwR package*.
You can obtain *pairwise scatterplots* between all the variables in the data set. This is done using the function *pairs*.

```
par(mex=0.5)
pairs(cystfibr, gap=0, cex.labels=0.9)
```

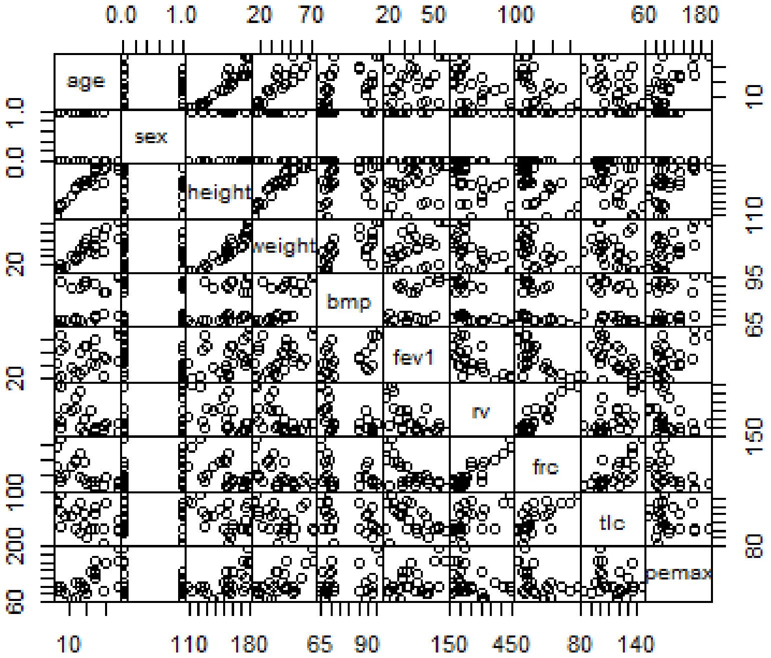

Figure 1: Pairwise plots for cystic fibrosis data.

The arguments *gap* and *cex.labels* control the visual appearance by removing the space between subplots and decreasing the font size. The *mex* graphics parameter reduces the interline distance in the margins.
The individual plots do get rather small, probably not suitable for direct publication, but such plots are quite an effective way of obtaining an overview of multidimensional issues. For example, the close relations among age, height, and weight appear clearly on the plot. In order to be able to refer directly to the variables in *cystfibr*, we add it to the search path.

Model specification and output

Specification of a multiple regression analysis is done by setting up a model formula with + between the explanatory variables:

```
lm.1=lm(pemax~age+sex+height+weight+bmp+fev1+rv+frc+tlc)
```

```
Call:
lm(formula = pemax ~ age + sex + height + weight + bmp + fev1 +
rv + frc + tlc)

Coefficients:
(Intercept)      age          sex        height       weight
176.0582       -2.5420      -3.7368      -0.4463       2.9928
       bmp          fev1           rv
    -1.7449       1.0807       0.1970
```

```
  frc          tlc
-0.3084       0.1886
```

As usual, there is not much output from **lm** itself, but with the aid of ***summary*** you can obtain some more interesting output:

```
> summary(lm(pemax~age+sex+height+weight+bmp+fev1+rv+frc+tlc))

Call:
lm(formula = pemax ~ age + sex + height + weight + bmp + fev1 +
rv + frc + tlc)

Residuals:
Min      1Q  Median      3Q     Max
-37.338 -11.532   1.081  13.386  33.405

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 176.0582   225.8912   0.779    0.448
age          -2.5420     4.8017  -0.529    0.604
sex          -3.7368    15.4598  -0.242    0.812
height       -0.4463     0.9034  -0.494    0.628
weight        2.9928     2.0080   1.490    0.157
bmp          -1.7449     1.1552  -1.510    0.152
fev1          1.0807     1.0809   1.000    0.333
rv            0.1970     0.1962   1.004    0.331
frc          -0.3084     0.4924  -0.626    0.540
tlc           0.1886     0.4997   0.377    0.711

Residual standard error: 25.47 on 15 degrees of freedom
Multiple R-squared:  0.6373,Adjusted R-squared:  0.4197
F-statistic: 2.929 on 9 and 15 DF,  p-value: 0.03195
```

Notice that there is not one single significant t value, but the joint F *test* is nevertheless significant, so there must be an effect somewhere.

The reason is that the t *tests* only say something about what happens if you remove one variable and leave in all the others. You cannot see whether a variable would be statistically significant in a reduced model; all you can see is that *no variable must be included.*

The ANOVA table for a multiple regression analysis is obtained using anova and gives a rather different picture:

```
> anova(lm(pemax~age+sex+height+weight+bmp+fev1+rv+frc+tlc))
Analysis of Variance Table

Response: pemax
           Df  Sum Sq Mean Sq F value   Pr(>F)
age         1 10098.5 10098.5 15.5661 0.001296 **
```

```
sex        1   955.4   955.4  1.4727 0.243680
height     1   155.0   155.0  0.2389 0.632089
weight     1   632.3   632.3  0.9747 0.339170
bmp        1  2862.2  2862.2  4.4119 0.053010 .
fev1       1  1549.1  1549.1  2.3878 0.143120
rv         1   561.9   561.9  0.8662 0.366757
frc        1   194.6   194.6  0.2999 0.592007
tlc        1    92.4    92.4  0.1424 0.711160
Residuals 15  9731.2   648.7
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

Note that, except for the very last line *(tlc)*, there is practically no correspondence between these *F tests* and the *t tests* from summary. In particular, the effect of age is now significant. That is because these tests **are successive**; they correspond to (reading upward from the bottom) a stepwise removal of terms from the model until finally only *age* is left. During the process, *bmp* came close to the magical 5% limit, but in view of the number of tests, this is hardly noteworthy. The tests in the ANOVA table are not completely independent, but the approximation should be good.

Some useful fuctions in R

After fitting a regression line, we may extract the following numbers:

```
coefficients(lm.1) # model coefficients
confint(lm.1, level=0.95) # CIs for model parameters
fitted(lm.1) # predicted values
residuals(lm.1) # residuals
anova(lm.1) # anova table
vcov(lm.1) # covariance matrix for model parameters
influence.measyrs(lm.1) # regression diagnostics
```

Comparing Models: Use anova(m1,m2)

When you use *anova(lm.1,lm.2)*, it performs the *F-test* to compare lm.1 and lm.2 (i.e. it tests whether *reduction in the residual sum of squares are statistically significant* or not). Note that this makes sense only if lm.1 and lm.2 **are nested models**. Note that you can compare nested models with the anova( ) function.
We will compare two models: a full model and a reduced model. For example, the full model might have more variables than the reduced model. For example, in the multiple regression case we have:

$$\begin{aligned} \text{Reduced model}: &\quad y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon \\ \text{Full model}: &\quad y = \beta_0 + \beta_1 x_1 + ... + \beta_k x_k + \epsilon \end{aligned}$$

A procedure leading directly to:

```
> m1<-lm(pemax~age+sex+height+weight+bmp+fev1+rv+frc+tlc)
> m2<-lm(pemax~age)
> anova(m1,m2)           ########################################order is important!!!
Analysis of Variance Table

Model 1: pemax ~ age + sex + height + weight + bmp + fev1 + rv + frc +
tlc
Model 2: pemax ~ age
Res.Df     RSS Df Sum of Sq      F Pr(>F)
1     15  9731.2
2     23 16734.2 -8   -7002.9 1.3493 0.2936
> anova(m2,m1)                                       #### correct format
Analysis of Variance Table

Model 1: pemax ~ age
Model 2: pemax ~ age + sex + height + weight + bmp + fev1 + rv + frc +
tlc
Res.Df     RSS Df Sum of Sq      F Pr(>F)
1     23 16734.2
2     15  9731.2  8    7002.9 1.3493 0.2936
```

Notice, however, that you need to be careful to ensure that the two models are **actually nested**. From the ANOVA table, we can thus see that it is allowable to remove all variables except age.

Model search

R has the step() function for performing model searches by the Akaike information criterion.

```
 step(fullmodel, data=, direction="backward")
 .
 .
 .
 Step:  AIC=160.66
 pemax ~ weight + bmp + fev1 + rv

 Df Sum of Sq   RSS    AIC
 <none>              10355 160.66
 - rv      1    1183.6 11538 161.36
 - bmp     1    3072.6 13427 165.15
 - fev1    1    3717.1 14072 166.33
 - weight  1   10930.2 21285 176.67

 Call:
 lm(formula = pemax ~ weight + bmp + fev1 + rv)

 Coefficients:
 (Intercept)       weight           bmp         fev1           rv
 63.9467       1.7489      -1.3772       1.5477       0.1257
```

One advantage of doing model reductions by hand is that you may impose some logical structure on the process.

```
summary(lm(pemax~age+sex+height+weight+bmp+fev1+rv+frc))
summary(lm(pemax~age+sex+height+weight+bmp+fev1+rv))
summary(lm(pemax~age+sex+height+weight+bmp+fev1))
summary(lm(pemax~age+sex+height+weight+bmp))
summary(lm(pemax~age+height+weight+bmp))
summary(lm(pemax~height+weight+bmp))
summary(lm(pemax~weight+bmp))
summary(lm(pemax~weight))
```

Notice that, once *age* and *height* were removed, *bmp* was no longer significant.

It is also a good idea to pay close attention to the age, weight, and height variables, which are heavily correlated since we are dealing with children and adolescents.

As it turns out, there is really no reason to prefer one of the three variables over the two others. The fact that an elimination method ends up with a model containing only *weight* is essentially a coincidence. You can easily be misled by model search procedures that end up with one highly significant variable it is far from certain that the same variable would be chosen if you were to repeat the analysis on a new, similar data set. What you may reasonably conclude is that there is probably a connection with the patients physical development or size, which may be described in terms of age, height, or weight. Which description to use is arbitrary. If you want to choose one over the others, a decision cannot be based on the data, although possibly on theoretical considerations and/or results from previous investigations. Alternatively, you can perform all-subsets regression using the leaps( ) function from the leaps package.

```
library(leaps)
leaps<-regsubsets(pemax~age+sex+height+weight+bmp+fev1+rv+frc+tlc,data=cystfibr,nbest=10)
summary(leaps)
# plot a table of models showing variables in each model.
# models are ordered by the selection statistic.
plot(leaps,scale="r2")
```

Other options for plot( ) are bic, Cp, and adjr2. Other options for plotting with subset( ) are bic, cp, adjr2, and rss.

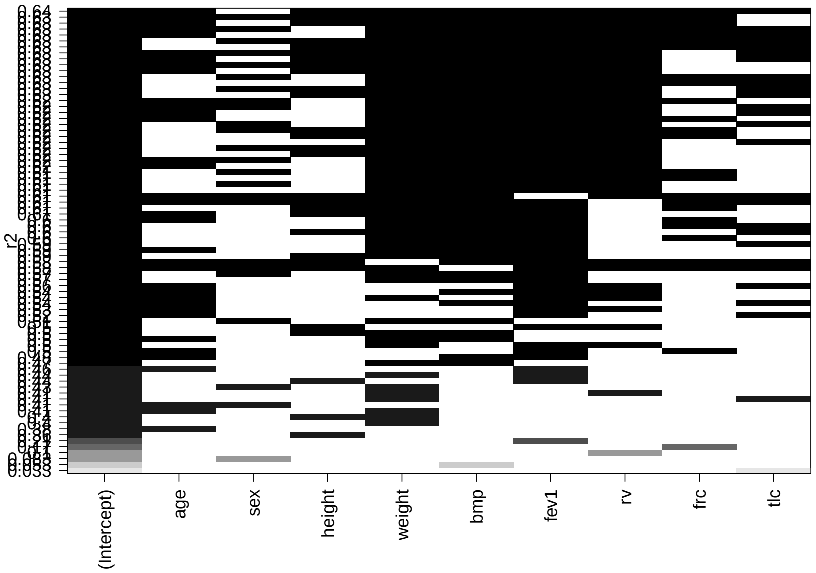

Figure 2: Table of models showing variables in each model

Relative Importance

The relaimpo package provides measures of relative importance for each of the predictors in the model.

```
# Calculate Relative Importance for Each Predictor
library(relaimpo)
?calc.relimp
calc.relimp(lm.1,type=c("lmg","last","first","pratt"),rela=TRUE)

# Bootstrap Measures of Relative Importance (1000 samples)
boot <- boot.relimp(lm.1, b = 1000, type = c("lmg","last", "first", "pratt"), rank = TRUE,diff
booteval.relimp(boot) # print result
plot(booteval.relimp(boot,sort=TRUE)) # plot result
```

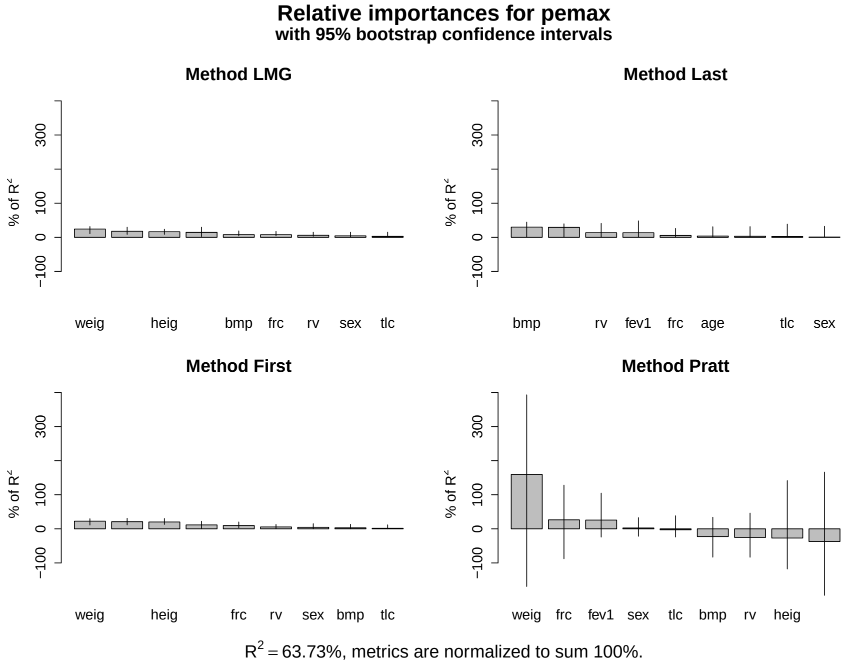

Figure 3: Bootstrap relative important

R code

```
?cystfibr              #########################Data in IsWR package
View(cystfibr)
attach(cystfibr)    #####In order to be able to refer directly to the variables
par(mex=0.5)
pairs(cystfibr, gap=0, cex.labels=0.9)
################################################fitting a regression line
lm.1=lm(pemax~age+sex+height+weight+bmp+fev1+rv+frc+tlc)
summary(lm.1)
anova(lm.1)
#########################################################useful functions
coefficients(lm.1) # model coefficients
confint(lm.1, level=0.95) # CIs for model parameters
fitted(lm.1) # predicted values
residuals(lm.1) # residuals
anova(lm.1) # anova table
vcov(lm.1) # covariance matrix for model parameters
influence.measures(lm.1)
#influencePlot(lm.1)####################library{car}
########################################################################checking outliears
leveragePlots(lm.1) # leverage plots
outlierTest(lm.1) # Bonferonni p-value for most extreme obs
###################################################### comparing two models
m1<-lm(pemax~age+sex+height+weight+bmp+fev1+rv+frc+tlc)
m2<-lm(pemax~age)
```

```
anova(m1,m2)           ########################################order is important!!!
anova(m2,m1)
#################################################### Model selection
step(m1, data=cystfibr, direction="backward")
step <- stepAIC(lm.1, direction="both")########################library{MASS}
step$anova # display results
################################################other options
library(leaps)
leaps<-regsubsets(pemax~age+sex+height+weight+bmp+fev1+rv+frc+tlc,data=cystfibr,nbest=10)
summary(leaps)
# plot a table of models showing variables in each model.
# models are ordered by the selection statistic.
plot(leaps,scale="adjr2")
#############################################Relative Importance
# Calculate Relative Importance for Each Predictor
library(relaimpo)
?calc.relimp
calc.relimp(lm.1,type=c("lmg","last","first","pratt"),rela=TRUE)

# Bootstrap Measures of Relative Importance (1000 samples)
boot <- boot.relimp(lm.1, b = 1000, type = c("lmg","last", "first", "pratt"), rank = TRUE,diff
booteval.relimp(boot) # print result
plot(booteval.relimp(boot,sort=TRUE)) # plot result

########################################################model selection manually
summary(lm(pemax~age+sex+height+weight+bmp+fev1+rv+frc))
summary(lm(pemax~age+sex+height+weight+bmp+fev1+rv))
summary(lm(pemax~age+sex+height+weight+bmp+fev1))
summary(lm(pemax~age+sex+height+weight+bmp))
summary(lm(pemax~age+height+weight+bmp))
summary(lm(pemax~height+weight+bmp))
summary(lm(pemax~weight+bmp))
summary(lm(pemax~weight))
```

Exerciser

1. The *secher* data are best analyzed after log-transforming birth weight as well as the abdominal and biparietal diameters. Fit a prediction equation for birth weight. How much is gained by using both diameters in a prediction equation? The sum of the two regression coefficients is almost exactly 3 can this be given a nice interpretation?