Model selection

Owen Petchey & Stefanie Muff

2/15/2017

```
rm(list=ls())
knitr::opts_chunk$set(eval=F, message=F, warning=F)
```

Introduction

We're going to work on the body fat data, and do some model selection. That is, find a selection of the available explanatory variables that gives a good, even the best model of the data, or a set of good models.

Note that we are going to be doing totally brainless model selection, the worst kind, but the easiest. And that its not particularly difficult with the body fat data, so we get the same answer lots of ways. With other datasets, it may not be so straightforward.

I should caveat that I'm not an expert in model selection. One might even say that I attempt to avoid it, as I think its just a big mess. But, its commonly used, you might need it, and you should definitely know about it.

First lets do the preliminaries and get the data:

```
library(tidyverse)
library(AICcmodavg)
dd <- read_delim("~/Desktop/bodyfat.txt", "\t", escape_double = FALSE, trim_ws = TRUE)
```

And, we'll do one more, and that is set R to fail if there are NAs. This is quite important, as some model comparison methods will happily compare models using different datasets, which can be caused by have NAs in some variables and not others.

```
options(na.action = "na.fail")
```

Manual model selection

Here we start with a model, and manually add or take away variables. Below, I start with the model with all main effects, and take one by one remove the variable with the highest p-value. Note that I'm not saying this is a good approach, I'm just showing it.

```
m1 <- lm(bodyfat ~ age + weight + height + neck + chest + abdomen +
           hip + thigh + knee + ankle + biceps + forearm + wrist, data=dd)
summary(m1)
m2 <- lm(bodyfat ~ age + weight + height + neck + chest + abdomen +
           hip + thigh + ankle + biceps + forearm + wrist, data=dd)
summary(m2)
m3 <- lm(bodyfat ~ age + weight + height + neck + abdomen +
           hip + thigh + ankle + biceps + forearm + wrist, data=dd)
summary(m3)
m4 <- lm(bodyfat ~ age + weight + neck + abdomen +
           hip + thigh + ankle + biceps + forearm + wrist, data=dd)
summary(m4)
m5 <- lm(bodyfat ~ age + weight + neck + abdomen +
           hip + thigh + biceps + forearm + wrist, data=dd)
summary(m5)
```

```
m6 <- lm(bodyfat ~ age + weight + neck + abdomen +
           hip + thigh + forearm + wrist, data=dd)
summary(m6)
m7 <- lm(bodyfat ~ age + weight + neck + abdomen +
         thigh + forearm + wrist, data=dd)
summary(m7)
```

For comparison, lets make a model without one of the clearly important variables... abdomen.

```
m8 <- lm(bodyfat ~ age + weight + neck +
         thigh + forearm + wrist, data=dd)
m0 <- lm(bodyfat ~ 1, data=dd)
```

We can conveniently look at all these models. We first put them in a list...

```
mods <- list(m0=m0, m1=m1, m2=m2, m3=m3, m4=m4, m5=m5, m6=m6, m7=m7, m8=m8)
aictab(mods)
```

Automatic model selection

Here we let the computer try to find the best model.

```
library(MASS)
m0 <- lm(bodyfat ~ 1, dd)

s1 <- stepAIC(m1, direction = "backward", AICc=TRUE)

s2 <- stepAIC(m0, direction = "forward", AICc=TRUE,
        scope=list(lower=m0, upper=m1))

s3 <- stepAIC(m0, direction = "both", AICc=TRUE,
        scope=list(lower=m0, upper=m1))


summary(s1)
summary(s2)
summary(s3)

AICc(s1)
AICc(s2)
AICc(s3)
```

Same model selected by each search direction. This is not always the case.

Package glmulti

This is like giving the deathstar superlaser to a toddler.

```
library(glmulti)
multi1 <- glmulti(bodyfat ~ age + weight + height + neck + chest + abdomen +
           hip + thigh + knee + ankle + biceps + forearm + wrist, data = dd,
            level = 1,               # No interaction considered
            method = "h",            # Exhaustive approach
            crit = "aicc",            # AIC as criteria
```

```
            confsetsize = 20,          # Keep 5 best models
            plotty = F, report = F,  # No plot or interim reports
            fitfunction = "lm",      # lm function
            includeobjects = T)
```

We can get a summary of what was done...

```
print(multi1)
```

A table for model weights (how well they fit the data)

```
weightable(multi1)
```

```
plot(multi1, type="s")
```

```
plot(multi1, type="p")
```

```
plot(multi1, type="w")
```

```
summary(multi1@objects[[1]])
```

Model average of the models returned by glmulti:

```
coef(multi1)
```

And predicts works as well:

```
predict(multi1)
```

Model averaging more manually

Say we mannually created some models and want to average across them. If we pick those with 4 AICc units of the best, and average across them, the best model is m7, but m6, m5, and m4 are each within 4 AICc units of the m7.

To average across those models...

```
library(AICcmodavg)
mods <- list(m1=m1, m2=m2, m3=m3, m4=m4, m5=m5, m6=m6, m7=m7)
aictab(mods)
modavg(mods, parm="age")
modavg(mods, parm="abdomen")
```