Myxomatosis example using ADMB via R2admb

Mollie Brooks

September 14, 2012

1 Introduction

This model is found in chapter 6 of *Ecological Models and Data in R* by Bolker 2008 (hereafter EMD book) and also in Dwyer et al 1990. The mean titer is predicted to follow a Ricker function through time and to be Gamma distributed.

```
m = ate^{-bt}
 T \sim \text{Gamma}(\text{shape} = s, \text{ scale} = m/s)
```

2 ADMB Code

The ADMB code (in a file called myxomatosis.tpl) looks like this:

```
DATA_SECTION
2
            init\_int nobs
3
            init_vector titer(1, nobs)
            init_vector day(1, nobs)
4
   PARAMETER_SECTION
5
6
            init_number a
7
            init_number b
8
            init_number s
9
            vector m(1, nobs)
            objective_function_value nll
10
   PROCEDURE SECTION
11
            m=Ricker(day, a, b);
12
            nll=dgamma(titer, s, s/m); //shape=s, rate=s/m
13
   GLOBALS_SECTION
14
            #include <statsLib.h>
15
16
            #include <ecolib.h>
```

2.1 DATA_SECTION

This is where you define and initialize data objects. It is typical to first define data objects, such as **nobs**, that control the size of data objects defined further down in the code. This is so that the same .tpl file can be used on different data sets without making changes to the code.

2.2 PARAMETER_SECTION

This is where you define and initialize parameters to be fit. These will start with init_. Also in this section, you can define objects, such as m where you'll store calculated values. The last thing in this section is the objective_function_value which will get minimized. I like to name it nll for 'negative log-likelihood'. Many people name the objective_function_value 'f'.

2.3 PROCEDURE_SECTION

This is where the negative log-likelihood is calculated and stored in the objective_function_value. Line 12 is where we calculate the mean m through time as following a Ricker function with parameters a and b. The predictions are stored in the vector m which we defined to be the appropriate size in the PARAMETER_SECTION. You can find out how the function is parameterized by going to http://admb-project.org/documentation/api/ and searching for 'ricker'. Then line 13 calculates the negative log-likelihood of the data given that it is Gamma distributed. The function dgamma takes a shape and rate parameter, so we need to know that the rate is the shape divided by the mean. On line 13, everything after the double slashes is a comment and is ignored in the computation.

2.4 GLOBALS_SECTION

This is where we include any libraries that we want to use in our calculations. In this example, we need the Ricker function which is contained in ecolib.h and the dgamma function which is contained in statslib.h. Information on functions contained in these libraries is available here http://admb-project.org/documentation/api/group__CONTRIB.html

3 R Packages (if you want to use R to organize the data and read the results)

- > library(emdbook)
- > library(R2admb)
- > setup_admb()
- [1] "/usr/local/admb"

```
> sessionInfo()
R version 2.15.0 (2012-03-30)
Platform: x86_64-apple-darwin9.8.0/x86_64 (64-bit)
locale:
[1] C
attached base packages:
[1] methods
              stats
                        graphics grDevices utils
                                                       datasets base
other attached packages:
[1] R2admb_0.7.5.2
                      emdbook_1.3.2
                                         lattice_0.20-6
                                                           MASS_7.3-17
[5] cacheSweave_0.6-1 stashR_0.3-5
                                         filehash_2.2-1
loaded via a namespace (and not attached):
[1] digest_0.5.2 grid_2.15.0 tools_2.15.0
```

The command setup_admb() finds the location of ADMB so that it can be used by R2admb.

4 Data

Data organization is the same as in the EMD book. Then we use write_dat() to put the values for the DATA_SECTION of our .tpl file into a .dat file. The order of the objects in the list must match lines 2 through 4 of the .tpl file. write_pin() puts the initial values for our parameters into a .pin file. The order of these objects must match lines 7 through 9 of our .tpl file. m and nll are the only objects in our PARAMETER_SECTION that do not need to be initialized because they are calculated in the PROCEDURE_SECTION. Also, note that m and nll are the only objects not beginning with init_

```
> data(MyxoTiter_sum)
> myxdat=subset(MyxoTiter_sum, grade==1)
> write_dat("myxomatosis", L=list(nobs=nrow(myxdat), titer=myxdat$titer, day=myxdat$day))
> write_pin("myxomatosis", L=list(a=1, b=0.2, s=50))
```

5 Running the model via R2admb

```
> compile_admb("myxomatosis")
> run_admb("myxomatosis", verbose=FALSE)
```

6 Reading the results via R2admb

```
> myxo1_admb=read_admb("myxomatosis")
> summary(myxo1_admb)
Model file: myxomatosis
Negative log-likelihood: 29.5091
Coefficients:
  Estimate Std. Error z value Pr(>|z|)
  3.561247
             0.190280
                       18.716 < 2e-16 ***
                       17.451 < 2e-16 ***
  0.171329
             0.009818
s 90.511940
            24.589000
                         3.681 0.000232 ***
Signif. codes: 0 '***' 0.001 '**' 0.05 '.' 0.1 ' ' 1
> logLik(myxo1_admb)
[1] -29.5091
```

7 Conclusion

We got the same parameter estimates and log-likelihood as the example in the book. It might seem like this isn't worth the extra effort, but if you wanted to increase the complexity of the model and/or include any random effects, then the flexibility of ADMB would be the selling point.

8 Appendix: Using R2admb with an incomplete tpl file

R2admb can write the DATA_SECTION and PARAMETER_SECTION of the .tpl file if you tell it what to write.

8.1 ADMB Code

The ADMB code (in a file called do_myx.tpl) looks like this:

```
PROCEDURE SECTION

dvar_vector m=Ricker(day, a, b);

nll=dgamma(titer, s, s/m);//shape=s, rate=s/m

GLOBALS_SECTION

#include <statsLib.h>
#include <ecolib.h>
```

Compare this code to the code in section 2. Notice on line 2 here that we defined the vector m "on the fly"; we do this instead of defining it in the PARAMETER_SECTION as we did on line 10 of the code above. We do this with objects that are used for storage en route to calculating the negative log-likelihood (i.e. not parameters). Notice that if we define m in the PARAMETER_SECTION, we use vector but if we define it on the fly in the PROCEDURE_SECTION then we use dvar_vector; m will serve the same function regardless. Vectors can be defined on the fly using dvar_vector even when not using R2admb, but we must do it this way when using the R2admb function do_admb.

8.2 Running the model via R2admb

First we create a list of data objects. These must have the same names as used in the PROCE-DURE_SECTION.

```
> Ldat=list(titer=myxdat$titer, day=myxdat$day)
```

Then we create a list of parameters with starting values. These also must have the same names as in the PROCEDURE_SECTION.

```
> Lpars=list(a=1, b=0.2, s=50)
```

Then the R2admb function do_admb accepts the name of the tpl file (minus the file extension), the data list, the parameter list, the name of the objective function, and some control parameters telling it to write the DATA_SECTION and PARAMETER_ SECTION and remove files after it runs.

```
> m1=do_admb("do_myx",
          data=Ldat,
          params=Lpars,
          objfunname="nll",
         run.opts=run.control(checkparam="write", checkdata="write", clean="all"))
> summary(m1)
Model file: do_myx_gen
Negative log-likelihood: 29.5091
Coefficients:
  Estimate Std. Error z value Pr(>|z|)
a 3.561247
             0.190280 18.716 < 2e-16 ***
 0.171329
             0.009818 17.451 < 2e-16 ***
s 90.511940 24.589000
                         3.681 0.000232 ***
Signif. codes: 0 '*** 0.001 '** 0.01 '* 0.05 '.' 0.1 ' 1
```