

Exercise 2: The CJS Model

This exercise closely follows the content of the 5th lecture and is mostly intended to show how to run program MARK to compute survival and capture probabilities from 'capture-recapture' data.. We have prepared exercises for use in either MARK or RMark.



Data input

Background

Data for this example came from the trapping of meadow voles, *Microtus pennsylvanicus*, at Patuxent Wildlife Research Center, Laurel, MD (Nichols *et al.*, 1984). Data were collected on a 10 x 10 grid of trapping stations spaced at 7.6m intervals in old field habitat. A single modified Fitch live trap (Rose, 1973) was placed at each station. Hay and dried grass were placed in the traps and whole corn was used as bait. Sampling occurred for five consecutive days each month, from June 1981 through December 1981. During each 5day trapping session, traps were opened in the evening of the first day, checked the following morning, locked open during the day, and reset in the evening, with the sequence repeated each day until 5 days had elapsed. A racoon, *Procyon lotor* (later captured), visited the traps on the final two nights of the second trapping session, essentially leaving only 3 days of trapping for this session. At each capture, animals were examined for a tag, sexed, weighed, and examined for external reproductive characteristics. Tagged animals were ear-tagged with numbered fingerling tags, and tag numbers of marked animals were recorded at each capture.

We used 'adult' (>22g) and 'young' (<=22g) animals and collapsed the 5 days of sampling each month into a single assessment of presence or absence, leaving six monthly sampling occasions.

For R users

- 1) Open the file "cjs2age.r" in your preferred text editor for. Note that the hash (#) symbol denotes comments in R and everything on the line after that symbol is ignored by R. The

first non-comment lines in the file clear the R workspace (memory) and set the working directory(folder) to the folder where you have the exercises. **You will need to modify the working directory.** Execute these lines in R.

```
rm(list=ls())           # clear workspace
setwd('h:/x/workshops/uf2016/exercises') # set working directory(folder)
library(RMark)
```

- 2) Assuming that you have already created a MARK input file, the first step in using RMark is to convert the input file into an RMark input “data-frame”. (You can think of a data-frame as a spreadsheet in R, where you can access the spreadsheet by rows and columns, or by variable(column) name. This conversion is done with the “convert.inp” function. The only required argument is the input filename (mp2age.inp), but since we have 4 groups in our input file, we also need to specify the group names to the convert.inp function. This is done with the group.df argument. The converted data-frame is saved with the name, “mpinp”. Execute that line, then type the new variable name into the R window.

```
mpinp=convert.inp('mp2age.inp',
  group.df=data.frame(agegrp=c('a','a','y','y'),sex=c('F','M','F','M'))
)
Mpinp
```

Notice that I didn't name the age-group variable, “age”. This is intentional and due to the fact that RMark has a pre-defined variable named “age”. The RMark variable, “age” can be used for certain things, but not in this case where we want to classify animals into two groups: young at first capture and adult at first capture. We'll use the “age” variable to make models where animals can be “young” in the 1st occasion, then adult after.

- 3) The 2nd step is to create a processed-data variable which contains other variables needed to setup and run the MARK models. The variable, “mppr” is a “list” variable, or a variable which contains other variables. The RMark function to do this is “process.data” and requires the converted input from the previous step, the type of MARK models which will be run, and the group variable name (if applicable). Execute this line, then type ‘mppr’ to see the contents of this variable.

```
mppr = process.data(mpinp, model='CJS', groups=c("agegrp","sex"))
```

- 4) The 3rd step is to create design-matrix data variables, needed by MARK to build models. The RMARK function is “make.design.data” and the processed-data variable created in the previous step is needed as an argument to the function. Execute this line and type ‘mpdd’ to view the contents of this variable. This is a list-type variable and it contains a data-frame variables, Phi (for apparent survival) and p (capture probability) for the estimated parameters for the model-type (CJS) we specified. The columns of Phi and p are the variables we can use in building MARK models.

```
mpdd = make.design.data(mppr)
mpdd$Phi
```

- 5) In order to build models where animals can be “young” for 1 capture occasion and “adult” for other occasions, we need to add a variable to our design-data variable. We’d like to create a new variable for Phi which takes on two values:

0 if animal is in group, “young” AND months-since-orig-capture <= 0

1 otherwise

```
mpdd$Phi$agecl=1 # create new variable, agecl
# next, get row numbers where group="y" and age=0
i=((substr(mpdd$Phi$group,1,1)=="y") & (mpdd$Phi$Age==0))
mpdd$Phi$agecl[i]=0 # set new variable to zero for those rows
print(mpdd$Phi) # look at new Phi data-frame
```

Modeling strategy is to develop models corresponding to our hypotheses of how survival and/or capture probabilities are affected by age, sex and time.

- 6) Now we’re ready to build our first MARK model. We’ll start with the most simple model (survival constant over time and equal among age/sex groups). To create a model, we call the ‘make.mark.model’ function with the processed-data variable created in step 3 (mppr), the design-data variable created in step 4 (mpdd), title, and list of parameters as arguments. Here, formula=~1 means the parameter is a constant value. Execute these 3 lines and type ‘phi_1_p_1’ to examine what we’ve created. This variable is another list-variable which contains other variables to be used when we run the model.

```
phi_1_p_1 = make.mark.model(mppr,mpdd,title='Patuxent Mp data',parameters=list(
  Phi=list(formula=~1),
  p=list(formula=~1)
))
```

- 7) We can run the model by calling the 'run.mark.model' function with the model variable created in step 6 as the argument. Execute this line then type 'phi_1_p_1_out' to examine the contents. The output from MARK is stored in a text file and by typing 'phi_1_p_1_out', the text file is opened in notepad. This is the usual output you would get if you ran MARK interactively using its GUI. The output is also stored as an R list-variable. Type 'str(phi_1_p_1_out)' to display the structure of this list-variable. What is the survival rate estimate from this model? You can get it from the text output displayed in notepad, or by looking at the phi_1_p_1_out\$results\$real variable.

```
phi_1_p1_out=run.mark.model(phi_1_p_1)
Phi_1_p_1_out$results$real
```

- 8) To run other models, we only need to repeat the last 2 steps: create.mark.model, and run.mark.model. With these data, the interest was in whether survival and/or capture probabilities were different among age and sex classes, as well as time. So, we make new mark models, with different assumptions about survival and capture probabilities. Execute these lines, as well as the line to run the models.

```
#           make model with age-specific survival, phi(a)p(.)
phi_a_p_1 = make.mark.model(mppr,mpdd,parameters=list(
  Phi=list(formula=~agecl),
  p=list(formula=~1)
))
phi_a_p_1_out = run.mark.model(phi_a_p_1)
```

```
#           make model with age and sex-specific survival, phi(a*s)p(.)
phi_axs_p_1 = make.mark.model(mppr,mpdd,parameters=list(
  Phi=list(formula=~agecl*sex),
  p=list(formula=~1)
))
phi_axs_p_1_out = run.mark.model(phi_axs_p_1)
```

```
#           make model with age and sex-specific survival, sex-specific capt. probs,
phi(a*s)p(s)
phi_axs_p_s = make.mark.model(mppr,mpdd,parameters=list(
  Phi=list(formula=~agecl*sex),
  p=list(formula=~sex)
))
```

```

phi_axs_p_s_out = run.mark.model(phi_axs_p_s)
#           make model with age,sex,time-specific survival, sex-specific capt. probs,
phi(a*s*t)p(s)
phi_axsXt_p_s = make.mark.model(mppr,mpdd,parameters=list(
  Phi=list(formula=~agecl*sex*time),
  p=list(formula=~sex)
))
phi_axsXt_p_s_out = run.mark.model(phi_axsXt_p_s)

#           make model with age,time-specific survival, sex-specific capt. probs,
phi(a*t)p(s)
phi_aXt_p_s = make.mark.model(mppr,mpdd,parameters=list(
  Phi=list(formula=~agecl*time),
  p=list(formula=~sex)
))
phi_aXt_p_s_out = run.mark.model(phi_aXt_p_s)

#           make model with age,time-specific survival, sex-specific capt. probs
#           with additive effect of age on survival
phi_aPt_p_s = make.mark.model(mppr,mpdd,parameters=list(
  Phi=list(formula=~agecl+time),
  p=list(formula=~sex)
))
phi_aPt_p_s_out = run.mark.model(phi_aPt_p_s)

```

- 9) We can create a table of model results by calling the ‘model.table’ function, with the names of the variables which contain the output of each model as an argument. Execute this line, and the next line to print the table.

```

tbl=model.table(model.list=ls(pattern="phi.+out"),adjust=T,use.lnl=T)
print(tbl)

```

Shortcut: Instead of listing all of the models by name in the model list, I used the R function, “ls”, which lists all variables in the workspace which start with “phi” and end with “out”.

Questions:

1. The models with interaction between age and time and additive time + age effects were both competitive. In the interactive model, adult survival is higher in

some months and young survival in others. Can this happen as well in the additive model? Why or why not?

2. In many species of vertebrates, young are predicted to have lower apparent survival than adults. Was this true in the example? What biological stories might explain the direction of the estimated average difference between young and adult survival?
3. The a priori hypothesis was that males would have higher capture probabilities than females. Was this true? What biology might underlie this prediction and difference?
4. Some population models (such as stochastic projection matrices) require estimates of true temporal variance of vital rates such as survival. But what other source of variation is present in the monthly variation among survival estimates? How can the true temporal variance be separately estimated?
5. The CJS model permits inference about capture and apparent survival probabilities, as shown. But under the JS model, we can also estimate abundance of adults. How do we do this?
6. Why can't we estimate the number of young in the same manner as for adults? What piece of information are we missing?