

First Steps...

In this chapter we will introduce the basic mechanics of running SURGE, using a small data set consisting of 7 years of capture-recapture data on a small passerine bird, the European Dipper (*Cinclus cinclus*). This data set is the same as that used in “Example 1” in Lebreton *et al.* (1992), and consists of marking and recaptures of 294 breeding adults each year during the breeding period, from early March to 1 June. All birds in the sample were at least 1 year old when initially banded. We’ll forgo GOF (Goodness of Fit) testing for the moment, although we emphasize that, in fact, this is the prerequisite step before you analyze your data. GOF testing is covered in the Appendix.

- For this chapter, you need to find the European Dipper data files (ED.REL). If you purchased the “commercial” version of SURGE over the last couple of years, the data files are in the /EXAMPLES subdirectory that was created when SURGE was installed. The data are also available from the SURGE Web page (<http://www.biol.sfu.ca/cmr/surge>).
- If the data are in RELEASE format, you’ll first need to transform the data to SURGE format, using the utility RELTOSUR (see Chapter 2). The ED.REL data file contains the capture histories for both males and females. Use RELTOSUR to create 2 SURGE files: MALES.SUR and FEMALES.SUR. We will start the process of learning how to use SURGE using only one of these 2 data sets, MALES.SUR. In later chapters, we will consider the analysis of multiple data sets. Copy this file MALES.SUR into your working directory on your computer.

STEP 1) starting SURGE

- How you start SURGE depends on whether or not you are running the CR interface (which came with the commercial version), or are working directly from the DOS prompt, or are using the new Windows-based version. If you are running SURGE from CR, the documentation provided with CR will provide the necessary details. In

this book, we will assume you are running SURGE from either the DOS prompt or the newer Windows-version (for simplicity, we’ll refer to this version of SURGE as WinSURGE).

- If you are using the older DOS version, you should first check to see that the directory containing the SURGE executable is in the PATH. You can do this by simply typing

```
PATH <enter>
```

If the directory containing SURGE is not in the PATH, then edit the AUTOEXEC.BAT file to add the appropriate directory to the PATH, and then re-boot your computer to make this change effective.

- If SURGE is in the PATH, then simply switch to the directory containing the file(s) you want to analyze (in this case, the directory into which you copied the file MALES.SUR), and start the program by typing:

```
SURGE4 <enter>
```

If SURGE has been properly installed on your machine, the program should initialize, and present you with the opening screen (Fig. 3.1).

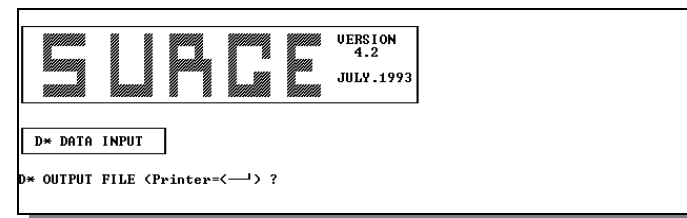


Fig. 3.1

- If nothing happens (or you get some typically obscure DOS error message), this is a good indication that either (a) the directory containing the SURGE program is not in the PATH (go back and check that it is), or (b) SURGE was not installed correctly. If (b), then you’ll need to consult the installation instructions that accompanied the

software.

- If you're using WinSURGE, you should see the following opening screen (Fig. 3.2)

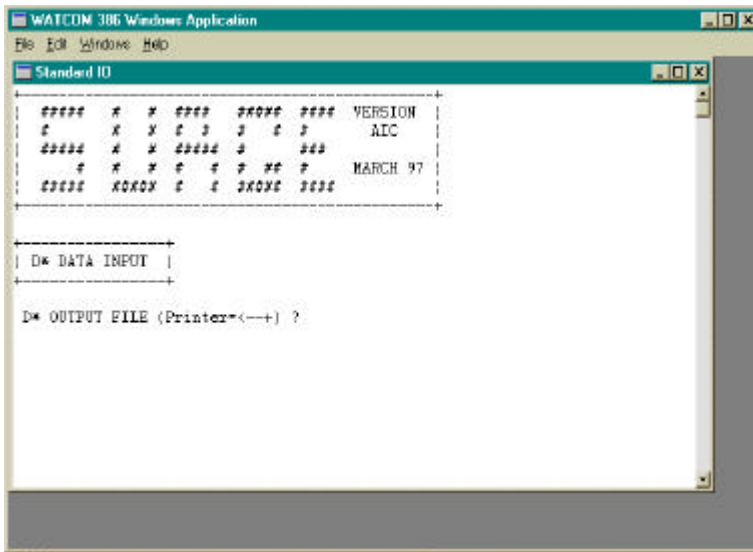


Fig. 3.2

- As you can see, WinSURGE is superficially very similar to the DOS version of SURGE - only the title ('Version AIC'), and the fact that WinSURGE places SURGE inside a Window, are notably different in the newer version. Since the two versions are virtually identical in appearance, we will generally use screen-captures from the DOS version, and will show screens from WinSURGE only when the information presented on-screen differs significantly.

STEP 2) The opening screen

- The opening screen is more accurately referred to as the "data input screen", since it is here that you tell SURGE where to find the data you want to analyze, and where to send the results of your analyses.

The "data entry" function of this screen is indicated in a characteristic way in SURGE, by the character string "D*" which precedes every prompt on the screen. As you will see, SURGE makes use of these simple mnemonic devices to help you keep track of the different screens and their functions.

- The first thing SURGE asks you for is the name you want to give to the output file - the file where the results from your analyses will be stored. If you want to store the output file in your current directory, simply type in the name of the file. If you want to store the output in a different directory, you must write out the full path and filename. SURGE also gives you the option of sending the output directly to a locally attached printer, simply by hitting the <enter> key at this prompt. For the moment, we'll store our output in the same directory, in a file called MALES.LST. Although SURGE does not impose a specific file extension, we suggest using either .LST or .OUT.
- Once you've entered the name of the output file, you are then prompted for a title for your output. This title will be printed at the top of the output from each analysis. Obviously, it is helpful to use a title which will allow you to keep track of which analysis you've done (since SURGE appends the results of successive analyses in a given session together in the same output file - the title will provide a logical separator). Obviously, the title you choose should be meaningful to you.
- For our initial analysis of the male dipper data, we are going to fit the fully time-dependent Cormack-Jolly-Seber model (CJS) - in model notation, we write " $\Phi(t), p(t)$ ", indicating that both survival (Φ , ϕ) and recapture (p) are time-dependent.
- Once you have entered the title, you will be prompted for the number of data sets you want to analyze. As we saw in Chapter 2, when you created the .SUR (SURGE) files, using RELTOSUR, you created a separate .SUR file for each group in your data set. This can be somewhat confusing at first - SURGE asks you for the number of data sets you want to analyze. What it really means is "how many groups do you want to analyze?". For the moment, however, we're only interested in analyzing the male dipper data, so we enter the number 1, for one group.
- Finally, you will be asked for the name of the file containing the

capture-recapture data you want to analyze. In this first example, we enter the name MALES.SUR (Fig. 3.3). If you enter the name of a file that does not exist, SURGE will respond by asking the question again. And again...and again, until you enter a valid filename. You can break out of this loop using <ctrl>-Z. As with defining the output file (above), you can use the full path and filename to access files that are not in the current working directory.

```

SURGE          VERSION
              4.2
              JULY.1993

D* DATA INPUT

D* OUTPUT FILE <Printer=<---> ? males.lst
D* TITLE           ? phi<t>,p<t>
D* NUMBER OF DATA SETS ? 1
D* FILENAME FOR DATA SET # 1 ? males.sur
  
```

Fig. 3-3

STEP 3) modelling ϕ and p

- Once you've completed the data input screen, you will immediately be presented with the next screen, the first of two "model specification" menus (Fig. 3-4)

```

M* MODEL CHOICE : SURVIVAL PROBABILITIES

M*      AGE DEPENDENCE           = 1
M*      TIME DEPENDENCE          = 2
M*      CONSTANCY OVER AGE AND TIME = 3
M*      USER DEFINED MODEL       = 4
M*      TIME DEPENDENCE, DEP. ON EXT.UAR. = 5
M*      2 AGE CLASSES 1=TIME DEP. 2=TIME DEP. = 6
M*      1=TIME DEP. 2=EXT.UAR. = 7
M*      1=EXT.UAR. 2=TIME DEP. = 8
M*      1=EXT.UAR. 2=EXT.UAR. = 9

M* OPTION FOR DATA SET males.sur ?
  
```

Fig. 3.4

- Again, notice the use of the mnemonic character pair "M*" preceding each prompt, indicating that this is a "model specification" screen.



Why does the model selection screen that I see on my computer look different than the one shown in Fig. 3-4? Unfortunately, there are a number of different "flavours" of SURGE floating around, each of which may differ in the way certain things are presented on the screen. Structurally, they all work the same - the differences are generally just cosmetic. In most cases, with a bit of study, you will quickly figure out the correspondence between the version of SURGE you are using, and the version used in this book.

To our knowledge, the model specification screen in all versions of SURGE $\geq 4.x$ will let you choose amongst the following key models:

- constant over time
- time dependent
- age dependent
- various 2-age models
- user-defined models
- temporal covariables

In fact, many of the selections included in the model choice menu(s) are provided more as a convenience to the user, and are not necessary to use SURGE effectively. In fact, as we will see, all of the models presented in the model choice menus (for either ϕ or p) could be implemented using a user-defined model.

- In our first example, we are going to fit model ϕp_t - the basic CJS model. To do this, we could either create a user-defined model (see Chapter 8), or use the "short-cut" provided by the model choice menu.
- It is important to remember that our initial model choice menu refers to the survival rates! Take another look at Fig. 3-4. Notice the title at



the top of the screen, telling you that you are modelling the survival probabilities. *SURGE treats survival and recapture separately, and sequentially, starting first with survival.*

- We now proceed with “informing” SURGE that we want to use a time-dependent model for the survival rates. If you scan down the input screen, you will see that the time dependent model corresponds to choice number 2 (“time dependency”). So, to select this model, we enter the number 2, and hit the <enter> key.
- Once you’ve hit the <enter> key, you will be presented with the model specification menu for modelling recaptures. The first thing you will notice is that the appearance of the menu for modelling recaptures (Fig. 3-5) is **identical** to the one you just worked with for modelling survival (Fig. 3-4) - only the title at the top differs.

```

M* MODEL CHOICE : RECAPTURE PROBABILITIES

M*      AGE DEPENDENCE           = 1
M*      TIME DEPENDENCE         = 2
M*      CONSTANCY OVER AGE AND TIME = 3
M*      USER DEFINED MODEL      = 4
M*      TIME DEPENDENCE, DEP. ON EXT. UAR. = 5
M*      2 AGE CLASSES 1=TIME DEP. 2=TIME DEP. = 6
M*      1=TIME DEP. 2=EXT. UAR. = 7
M*      1=EXT. UAR. 2=TIME DEP. = 8
M*      1=EXT. UAR. 2=EXT. UAR. = 9

M* OPTION FOR DATA SET males.sur ?

```

Fig. 3-5

- Again, in this first example we are fitting the fully time-dependent CJS model - time dependency in both survival and recapture rates. Thus, we again select choice 2 from the menu. and hit the <enter> key.
- We have now defined the basic underlying structure of the CJS model for survival and recapture. SURGE will now ask you a series of questions relating to some “fancy things” you can do with this basic models, now that it has been defined. One of the greatest strengths of SURGE is that it allows you to quickly and (fairly) easily vary the structure of these basic models, allowing you to test more complex and powerful models against your data. Although we are not going to deal with this now (they will be dealt with in considerable detail in

subsequent chapters), SURGE will ask you if you want to anyway. Thus, we need to quickly run through the next set of questions SURGE will direct your way.

STEP 4) constraining parameters

- As we will see in subsequent chapters, SURGE allows the user to constrain parameter estimates (for either ϕ or p , or both) to be functions of external covariates. SURGE presents you with the “constraints menu” (Fig. 3-6), which is where you would “tell” SURGE how many and what type of constraints to the underlying model you want to apply. SURGE informs us that, at this stage, there are 0 constraints in place (i.e., all we have is the basic time-dependent models we just defined for survival and recapture - no constraints currently exist). Note the use of the “C*” character mnemonic.

```

C* CONSTRAINING PARAMETERS

C* There are already 0 constraints
C* HOW MANY MORE CONSTRAINTS? <display model=-1; back to M=-2> :

```

Fig. 3-6

- SURGE then asks you how many more constraints do you want to add. Nested within this prompt are 2 options: “display model=-1”, and “back to M=-2”. These options allow you to display the current model structure (discussed later on), or to go back to the model specification menus (remember the “M*” character mnemonic?), respectively. We will discuss these options in more detail in the next chapter. For the moment, we want to bypass this option (i.e., we don’t want to add any constraints). The default is 0, so we simply hit the <enter> key.

STEP 5) fixing parameters

- Once you have left the “constraints menu”, SURGE will ask you if you want to fix any parameters (for example, if you know that the recapture rate at a particular occasion is zero - because you weren’t there). You

can fix any of the parameters at this stage (Fig. 3-7). Again, note the “F*” mnemonic.

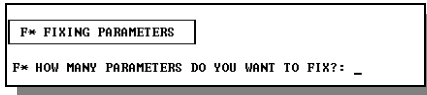


Fig. 3-7

- For this example, we don't want to fix any parameters, so we bypass this option by simply hitting the <enter> key (the default is fix 0 parameters).

STEP 6) changing the initial parameter estimates

- SURGE is very good at picking starting values for the maximum likelihood estimation. It almost always will quickly find the solution. However, if it doesn't, it may be necessary to change the starting values. SURGE gives you the opportunity to “change initial values” in the next menu (“I*”), first for survival (Fig. 3-8), then for recapture. Since the structure is the same for both, we will examine the details for survival only.

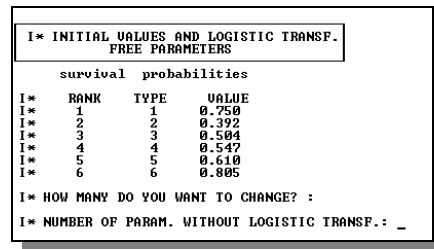


Fig. 3-8

- First, SURGE lists the starting values for each estimate of survival, based on the current model (in this case, time-dependence in survival). There are 6 recapture occasions, and thus 6 estimated survival rates (Rank 1, 2...6). SURGE follows this list by asking you how many you want to change. The default is zero, so we simply hit

the <enter> key.

- SURGE then asks you if you want to change the transformation from the logit scale (the default) to the identity scale (at present no other scales are available). In most cases, we will want to use the default logit scale, so we simply hit the <enter> key again, accepting the default.



What is the “logit scale “ or “logit transformation” referred to in the preceding paragraph? The logit transform is a transformation of the probability of survival or recapture from a [0,1] scale (all probabilities are bounded [0,1]) to a $[-\infty, +\infty]$ scale, which is necessary to model the probability of survival or recapture as a linear function of other variables. This is discussed in considerable detail in Chapter 6. Some other software, such as JOLLY and JOLLYAGE, does not use this transformation, which contributes to their limited utility.

STEP 7) running the analysis

- we have now told SURGE virtually everything we need to do for this particular analysis, and are ready to “run the analysis”. The final screen (Fig. 3-9) allows you to either (i) proceed with the calculations, or (ii) go back to any of the intermediate steps.

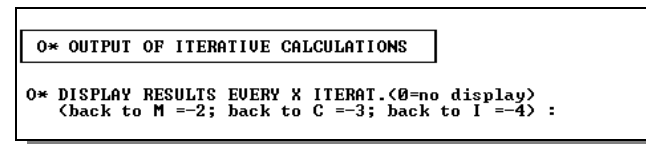


Fig. 3-9

- Suppose you decide that you've made a mistake somewhere along the way, and want to go back and change one or more of your responses. It is at this stage that the mnemonic character pairs that we have been noting come into play. At this point, you could decide to go back to any of the preceding screens, simply by typing in the appropriate number. For example, you could go back to the model selection

screens (M*) by entering “-2”, the constraints screen (C*) by entering “-3”, and so on.

- However, suppose you want to run the analysis at this point, rather than go back to an earlier step. How do you tell SURGE to proceed? Look carefully at Fig. 3-9. Notice that the program is asking you whether or not you want to display the results every X iterations. What does this mean? Well, SURGE uses an iterative approach to solving the maximum likelihood estimates of survival and recapture, and is asking if you want to display the results of this iterative process every X steps through the process. So, for example, if you want SURGE to print the results every 10 iterations, you would enter “-10”. For every 20 iterations, enter “-20”, and so on. Note that you can suppress printing of these intermediate steps by entering 0 (for no display). In some ways, it is probably useful to have SURGE print something on the screen, since this at least gives you some indication that “something is happening”. SURGE doesn’t have the “cute” little clock or hour-glass icons so popular with Windows programs, so your only indication that SURGE is calculating is to have it print the results every X iterations.
- Note, however, there are some limits to the frequency at which SURGE will print results to the screen. We certainly can’t use “-2” (i.e., can’t have SURGE print the results every 2 iterations). Why? If you look back at Fig. 3-9, you’ll see why. If we enter “-2”, rather than print the results every 2 iterations, it will cause us to jump back to the “model selection” part of the program. Clearly, this isn’t what we had in mind. Thus, if you choose to have SURGE print the results every X iterations, X must be either “-1” or “≤-5”. In our example, we choose to have the results printed to the screen every 10 iterations, so we enter “-10”, and then hit the <enter> key (Fig. 3-10).

```

O* OUTPUT OF ITERATIVE CALCULATIONS

O* DISPLAY RESULTS EVERY X ITERAT.(0=no display)
  (back to M =-2; back to C =-3; back to I =-4) : -10
O* PRINT VAR-COVAR MATRIX? <YES=<—> : no_
    
```

Fig 3-10

- Why do we use a negative number here? Well, in fact, you could use a positive value (any positive value >0). If you use a positive number, SURGE prints a LOT more of the details of the iterative process to the screen, including the current model parameter estimates. Using the negative values (e.g., “-10”) reduces the amount of detail sent to the screen. For our example, we’ll use the reduced information mode.
- SURGE then asks if you want to print the “var-covar” (for variance-covariance) matrix. The variance-covariance matrix is potentially useful for several purposes. For this example, however, we choose not to print this matrix. The default in SURGE is to print the matrix (i.e., to print the matrix, all you need to do is hit the <enter> key). To override the default, we type “N” or “no”, and hit the <enter> key (Fig. 3-9).
- As soon as you press the <enter> key, SURGE will begin executing. If you chose to have SURGE print out the results of the calculations every X iterations, you will get a good indication of how quickly SURGE is running. However, *the more you have SURGE print to the screen, the slower it will run* (in any program, printing to the screen slows down execution). If you chose option 0 (no display), SURGE will indicate that it is finished in a rather uneventful way, by printing an unformatted table of the final parameter estimates, followed by the function minimums (or gradient components), followed in turn by the final model deviance - in our example, 313.08 (Fig. 3-11). If you see this on the screen, SURGE has finished calculating.

```

53 63 1
0.3130804848520692D+03
0.8329091D+00 -0.3101549D+00 0.2115238D-01 0.4447980D+00 0.2851896D+00
0.1173420D+01 0.9315582D+00 0.3244376D+02 0.2305146D+01 0.2547708D+01
0.2679879D+01 0.1173420D+01
0.2316799D-08 -0.3841513D-08 0.4934777D-08 0.3188580D-08 0.4291513D-08
-0.3137272D-08 -0.4932597D-08 -0.5855416D-13 -0.1258474D-09 0.2096421D-09
0.1041160D-08 -0.3137272D-08
DEVIANCE = 0.3130805D+03
strike <— to continue
    
```

Fig. 3-11

- Notice at the top of Fig. 3-11 the sequence “53 61 1”. While the precise meaning of these numbers is not important, it is useful to note that if the final value is not “1”, then SURGE has not completely reached the solution. If in fact this has happened, SURGE will also

prompt you directly by telling to “Convergence not reached. Recycle?” - basically, asking you if you want to continue. This is not an uncommon occurrence with large or poorly conditioned data sets. If, however, you are positive that SURGE has finished calculation (i.e., convergence has been achieved), and the final number is not “1”, then you may have a problem. A number other than “1”, provided that SURGE has reached a solution, means something has gone wrong.

- Now that SURGE has completed the calculations, simply hit the <enter> key to continue. However, what you see next depends on whether or not you are using DOS SURGE, or the newer WinSURGE. If you are using the older DOS-based version of SURGE, hitting the <enter> key will bring you immediately to the final screen (Step 8 - below). However, if you are using WinSURGE, hitting the <enter> key will present you with the results of the only real important difference between the two versions - the number of estimable and non-estimable parameters, and the calculated Akaike Information Criterion (AIC - much more on AIC in later chapters) (Fig. 3-12).

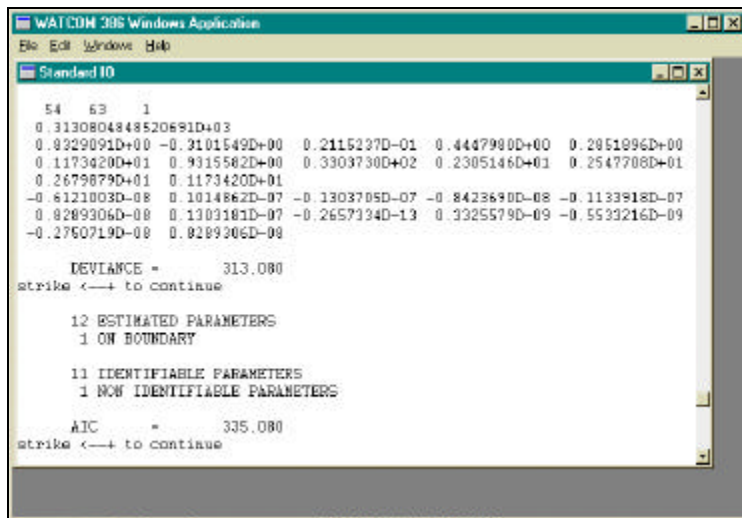


Fig. 3-12

- Experienced users of the old DOS-based version of SURGE will probably want to take a moment now to rejoice - counting parameters by hand (which the DOS version required you to do), was (admittedly) a pain. However, having said that, it is still **very** important that you know how to count parameters for a given model, and to be able to determine which parameters are estimable and which are not. Relying on the software to do it for you is **never** a good idea.
- The number of parameters and the AIC are also output to your results file, so there is no real reason to right them down now - WinSURGE prints them to the screen at this stage merely as a convenience. Hitting the <enter> key again will lead you to the final screen (Step 8),

STEP 8) the final screen

- At this point, you can either run another analysis, or quit SURGE and examine your output. The final screen (Fig. 3-13) shows the format for making these choices - you simply enter the number of the option you wish to use.

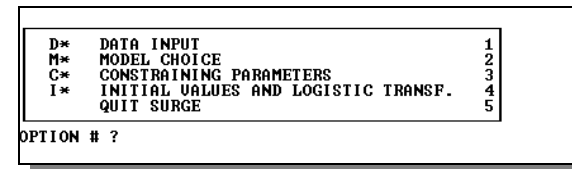


Fig. 3-13

- The results of picking option 5 should be obvious - SURGE terminates, leaving you at the DOS prompt. Your output will be in the file you specified on the opening “data input” screen (p. 3-2). Choice 1 to 4 will move you back to different points in the program. For example, if you enter “1”, you can move back to the “data input” screen, and select a new data set to analyze. Or, choice “2”, which

allows you select a new model, and so forth. As long as you stay within SURGE (i.e., don't terminate the program), SURGE will append the results from your next analysis at the end of the previous output. If you leave, and then re-start SURGE, and try to send your output to the same file, it will NOT append it to the existing file, but will over-write the previous contents. So, be aware!

- For this first example with the male dipper data, we decide to terminate SURGE, and examine our output. We enter option "5", and hit the <enter> key.

STEP 9) examining your output

- Now that we've finished our first simple analysis, we will want to look at our results, which (in our case) are contained in the output file MALES.LST. Program SURGE stores the results as simple ASCII text, which can be browsed with any editor or word processor you prefer (if you use a word processor, the formatting of the file may not look "correct" if you are using a proportional font).
- Here are the results for our first analysis, which will illustrate the standard SURGE output format (as we progress into more complex models in later chapters, we will note any changes in this format).
- First, at the top of the file, you'll find the title. If you've appended multiple analyses into the same output file, the title will serve as the "separator" between each result. In this case, there is only one result, since we only ran one analysis.
- Immediately below the title is a triangular matrix representation of the model you chose to analyze (in this case, model ϕ_p).
- First, SURGE gives you the 'triangular matrix' of the survival parameters.

```

SURVIVAL

PARAMETER STRUCTURE FOR DATA SET males.sur

1  2  3  4  5  6
   2  3  4  5  6
    3  4  5  6
     4  5  6
      5  6
       6

```

- The survival matrix is followed immediately by the recapture matrix

```

RECAPTURE

PARAMETER STRUCTURE FOR DATA SET males.sur

7  8  9 10 11 12
   8  9 10 11 12
    9 10 11 12
     10 11 12
      11 12
       12

```

- We will discuss the format and structure of these "parameter structure matrices" in considerable detail later. However, for the moment, notice that the parameter structure is presented in 2 pieces, first the survival structure, followed by the recapture structure. In this case, because we were fitting model ϕ_p , where both survival and recapture are time-dependent, the basic structure of the two triangular matrices is the same.
- What do the numbers in these matrices represent? Consider Fig. 3-14. This figure represents the parameter structure of the CJS model. The ϕ_i values represent the survival rates between successive

occasions (there are 7 total occasions in the dipper study), while the p_i values represent the recapture rates at the i th occasion. For details, see Lebreton *et al.* (1992).

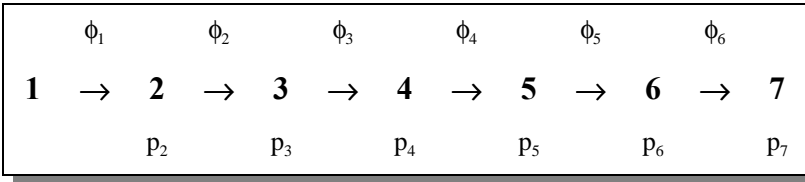


Fig. 3-14

- In Fig. 3-15, we see how SURGE numbers the parameters used in the model structure represented in Fig. 3-14.

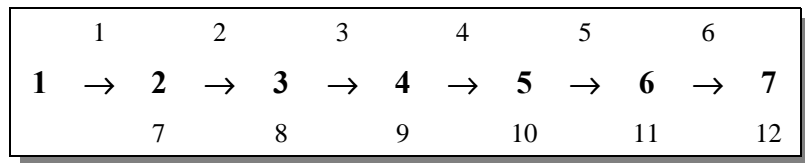


Fig. 3-15

- All SURGE has done is substitute a numerical indexing scheme for the individual ϕ_i and p_i values, respectively. The numbers 1 to 6 correspond to the survival values ϕ_1 to ϕ_6 respectively. The numbers 7 to 12 correspond to the recapture rates p_2 to p_7 , respectively. Notice that SURGE indexes the survival parameters first (1 to 6), followed in numerical sequence by the index values for the recaptures (7 to 12). In other words, the indexing SURGE uses does not correspond to the number of the particular interval or occasion involved.
- For example, the recapture index 8 corresponds to the recapture rate at occasion 3 (i.e., p_3). Although it can be a little confusing at first, with a bit of study of Fig. 3-14 and Fig. 3-15, you should be able to see the basic connection between the “true” parameter structure of the model, and the way in which SURGE indexes it, both internally

(which will become important later as we examine more complex models), and in the output file. *It is important to grasp this connection, so take some time now to ensure that you do.*



- The next item in the output file is the **model deviance**. This value is a relative model deviance, which differs from the true deviance by some unknown constant. The smaller the absolute value of the deviance, the better the model fit. For our analysis of the male dippers, the deviance of the CJS model is: 313.080 If you are using WinSURGE, the model deviance is followed by the number of estimable parameters (see below), and the calculated AIC (Fig. 3-16).

```

WATCOM 386 Windows Application
File Edit Windows Help
Standard IO

53 63 1
0.3130804848520692D+03
0.8329091D+00 -0.3101549D+00 0.2115238D-01 0.4447980D+00 0.2851896D+00
0.1173420D+01 0.9315582D+00 0.3244376D+02 0.2305146D+01 0.2547708D+01
0.2679879D+01 0.1173420D+01
0.2316799D-08 -0.3841513D-08 0.4934777D-08 0.3188580D-08 0.4291511D-08
-0.3137272D-08 -0.4932597D-08 -0.5055416D-13 -0.1258474D-09 0.2096421D-09
0.1041160D-08 -0.3137272D-08

DEVIANCE = 313.080
strike (-- to continue

12 ESTIMATED PARAMETERS
1 ON BOUNDARY

11 IDENTIFIABLE PARAMETERS
1 NOT IDENTIFIABLE PARAMETERS

AIC = 335.080
strike (-- to continue
  
```

Fig. 3-16

- Note that WinSURGE tells you that, for this model, 12 parameters were estimated, one of which was estimated on the boundary (i.e., at either 0.0 or 1.0). Of the 12 estimated parameters, 11 were identifiable parameters, and 1 was not. Much more on the number of parameters, and the question of ‘boundaries’ and ‘identifiability’ in

the next chapter.

- These in turn are followed by the estimates of ϕ_i and p_i , respectively.

survival probabilities				
Index	estimate	95% confidence interval	st.dev.	
1	0.696970	0.306768	0.922806	0.177D+00
2	0.423077	0.256960	0.608621	0.936D-01
3	0.505288	0.343557	0.665920	0.852D-01
4	0.609402	0.446625	0.750992	0.800D-01
5	0.570818	0.425974	0.704470	0.729D-01
6	0.763763	0.425753	0.933767	0.136D+00
recapture probabilities				
Index	estimate	95% confidence interval	st.dev.	
7	0.717391	0.311905	0.934278	0.178D+00
8	1.000000	0.000000	1.000000	0.294D-07
9	0.909302	0.591216	0.985815	0.814D-01
10	0.927419	0.640321	0.989214	0.677D-01
11	0.935829	0.675348	0.990313	0.597D-01
12	0.763763	0.425753	0.933767	0.136D+00

- Note that the same numbering scheme used in the triangular matrix (p. 3-7 & p. 3-8), corresponding to Fig. 3-15, is used to index the values of survival and recapture rates.
- The estimate of each parameter is given with the 95% confidence interval (lower and upper bounds), followed by the standard error (not standard deviation, as the output suggests) of the estimate. For example, under the time dependent model, SURGE estimates that the survival rate of male dippers from the second to the third occasion (see Fig. 3-14

and 3-15) is 0.423077.

- Notice that the estimates for index 6 (survival rate from occasion 6 to occasion 7 in Fig. 3-13) and index 12 (recapture rate at occasion 7 in Fig. 3-14) are identical (0.763763). As discussed in detail in Lebreton et al. (1992), this reflects the fact that, for this model, the survival and recapture rates for the last interval are not individually identifiable.
- The value 0.763763 is actually the square root of the estimated product $\phi_6 p_7$, denoted as β_7 in Lebreton *et al.* (1992). The DOS-based version of SURGE does not tell you explicitly which of the parameters are non-identifiable - it assumes that you, as the user, will know this. WinSURGE does tell you which parameters are not separately identifiable.
- The distinction between “identifiable” and “non-identifiable” parameters is critical for correctly deriving the number of estimable parameters in the model. This is needed for a number of the statistical tests described in detail in subsequent chapters, and in Lebreton *et al.* (1992). In the male dippers example, under the completely time-dependent model, the estimates 1 to 5 and 7 to 11 are individually identifiable, whereas the estimate for parameters 6 and 12 are not (they are confounded, and estimated as β_7). So, for this data set, the fully time-dependent model has a total of 11 identifiable parameters. Although WinSURGE does this for you, we’ve already noted how important it is to understand *why* some parameters are not separately identifiable. We cover this and many other topics in later chapters.

Congratulations!

You have now finished your first CMR analysis using SURGE! Of course, the fact that there are MANY more pages left in this book should tell you there is a lot more left to be covered. But, you’ve at least run through SURGE once, and this is an important first step. If you don’t feel comfortable with what we’ve done, go back through the chapter again slowly - the basic mechanics involved in this chapter will be used repeatedly throughout the rest of the book, so it is important to feel comfortable with them before proceeding much further.

