

APPENDIX G

The ‘data bootstrap...’

The following introductory material was adopted liberally from Bishop *et al.* (2008), the main published paper describing the data bootstrap in Program **MARK**. The data bootstrap is also commonly referred to as the nonparametric bootstrap in the statistical literature. Efron & Tibshirani (1993) and Chernick (1999) provide extensive descriptions and advice on the use of the bootstrap method.

Independence of sample units is required in survival analyses to obtain appropriate estimates of sampling variance. The independence assumption is likely violated when sample units include monogamous mates or siblings. For example, adult pairs of Canada geese (*Branta canadensis*) typically mate for life and, therefore, do not behave independently of one another (Anderson *et al.* 1994, Sheaffer *et al.* 2004). Littermates of numerous species access the same nutrient supply and are often subjected to the same mortality risks. Some degree of biological dependence likely exists among individuals that use the same resources in time and space.

When the independence assumption is violated, survival point estimates are typically unbiased, but sample variances are underestimated (Wedderburn 1974, Cox 1983, Firth 1987, Breslow 1990, Schmutz *et al.* 1995). This condition is referred to as overdispersion or extra-binomial variation. Sample data will be overdispersed if siblings each die, or each survive, more often than expected under the assumption of independence. An overdispersion parameter, or variance inflation factor, c , is required to correct for overdispersed data, thereby facilitating appropriate inference using quasi-likelihood theory (Wedderburn 1974, Burnham and Anderson 2002; see Chapter 4 and Chapter 5).

The simplest approach for estimating c requires only the standard goodness-of-fit (GOF) test statistic for the global (i.e., most parameterized) model and its degrees of freedom (Cox and Snell 1989). The global model is needed to avoid mistaking model structure variation (i.e., lack of model fit) for overdispersion (Burnham and Anderson 2002). Chi-square GOF approaches have been used to address sibling dependence for a variety of species (Winterstein 1992, Gaillard *et al.* 1998, Schwartz *et al.* 2006, Wiens *et al.* 2006). Unfortunately, these approaches fail to provide an appropriate estimate of c under many circumstances because of inadequate sample sizes. Instead, more sophisticated numerical approaches are necessary (Schmutz *et al.* 1995; Bishop *et al.* 2008). The data bootstrap we introduce here provides a method to evaluate whether theoretical variance estimates are valid when the individuals are considered statistically independent, or whether variance inflation procedures are required to account for dependence among (say) siblings.

The simulation procedure in **MARK** allows bootstrapping the encounter histories data to generate bootstrap replication where an individual covariate is used to identify the blocks of data (litters, family groups, etc.) that are resampled with the bootstrap procedure. To bootstrap encounter histories, you select the ‘**Bootstrap Data**’ option from the ‘**Simulation**’ menu choices. A potential trap in using this procedure is that the encounter history file should be set up to have each history represent the

appropriate number of animals. Typically, this would be 1 animal, but as the example below shows, such may not be the case. Encounter history files with encounter histories pooled so that the count frequency is > 1 will work correctly with this procedure because the histories are unpooled for sampling. Be careful about the group structure of your data – encounter histories *within* groups are resampled, never *across* groups.

Bootstrapping of individual encounter histories will not provide any additional information from the usual analyses. That is, resampling individual encounter histories produces the same asymptotic estimates and standard errors as the default maximum likelihood approach in MARK. Where bootstrapping the encounter histories is most useful is when there are dependencies across animals. As described above, the most typical example of such data are marking of young that are known to be siblings. So, all the young in a litter or nest are marked. When the encounters of these animals are treated as if they are independent, the estimates will be generally unbiased, but the standard errors will be too small because the sampling unit is really the litter or nest, and not the individual animal. So, in the maximum likelihood estimation assuming independence of individuals, the sample size is assumed to be larger than it really should be.

G.1. Empirical example

To bootstrap litter or nest data such as described above, include an individual covariate that defines the nest or litter. This variable can be continuous, such as nest number 1 to the number of nests, or litter number. The main criterion is that each litter or nest is *uniquely identified* with this covariate. This covariate is then used to bootstrap the encounter histories, such that the nests or litters are resampled, instead of the individual encounter histories.

The example we use here to illustrate the data bootstrap analysis is the neonatal fawn survival study described in Bishop *et al.* (2008). Adult female deer commonly produce twins and occasionally triplets. Siblings should not be assumed to have independent fates, because they have the same dam, share ≈ 0.25 of their genome, and are exposed to nearly identical environmental conditions postpartum. Maternal body condition and disease status, as well as predation, are mechanisms that could cause dependence among sibling fates. However, siblings are commonly radio-collared in neonatal survival studies (Hamlin *et al.* 1984, Whittaker & Lindzey 1999, Carstensen *et al.* 2003, Jarnemo & Liberg 2005, Bishop *et al.* 2008). There are advantages to marking siblings, but the potential dependence of survival outcomes should be addressed. Sample unit dependence is also encountered in measurements of fetal survival, which can be made in free-ranging deer populations with the aid of ultrasonography and vaginal implant transmitters (Bishop *et al.* 2007). Inclusion of siblings is required when estimating fetal survival because the fetus sample is obtained from *in utero* counts, and each of an adult female's fetuses must be counted without error. Twin or triplet fetuses cannot be individually marked *in utero*, and, therefore, one fetus cannot be randomly chosen to include in the sample and others disregarded.

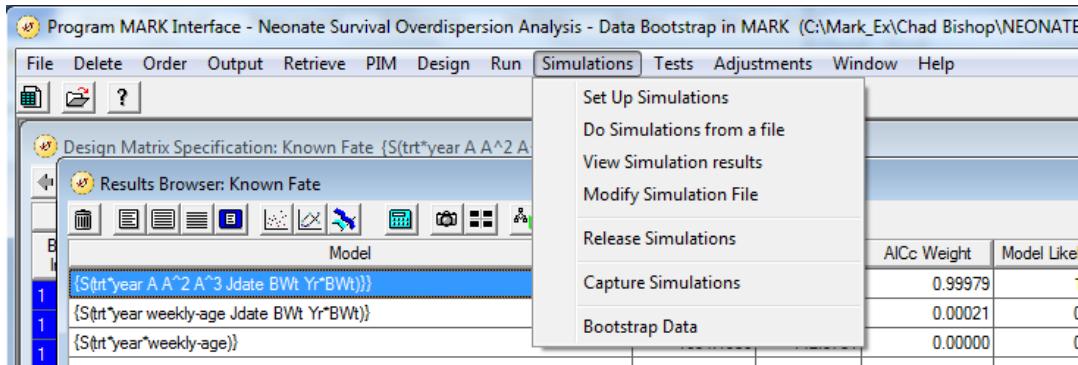
All fetuses and neonates used in the study described in Bishop *et al.* (2008) were offspring of free-ranging, radio-collared adult female mule deer. We administered a nutrition enhancement treatment to half of the adult females during winter and early spring to meet research objectives described elsewhere (Bishop *et al.* 2009). No nutrition treatment was applied to the other half of the adult females, hereafter control deer. Neonatal fawns were followed for 182 days (26 weeks). The encounter histories were entered as 182 occasions for 6 groups. The top of the input file (**deer-bootstrap.inp**), plus the first 3 encounter histories are shown on the next page, where the encounter histories wrap across lines because of their length (264 characters).

Survival was estimated on a weekly basis by creating PIMs with daily survival constant within the 26 weeks. Thus, 26 real parameters are estimated for each of the 6 groups, giving 156 real parameters.

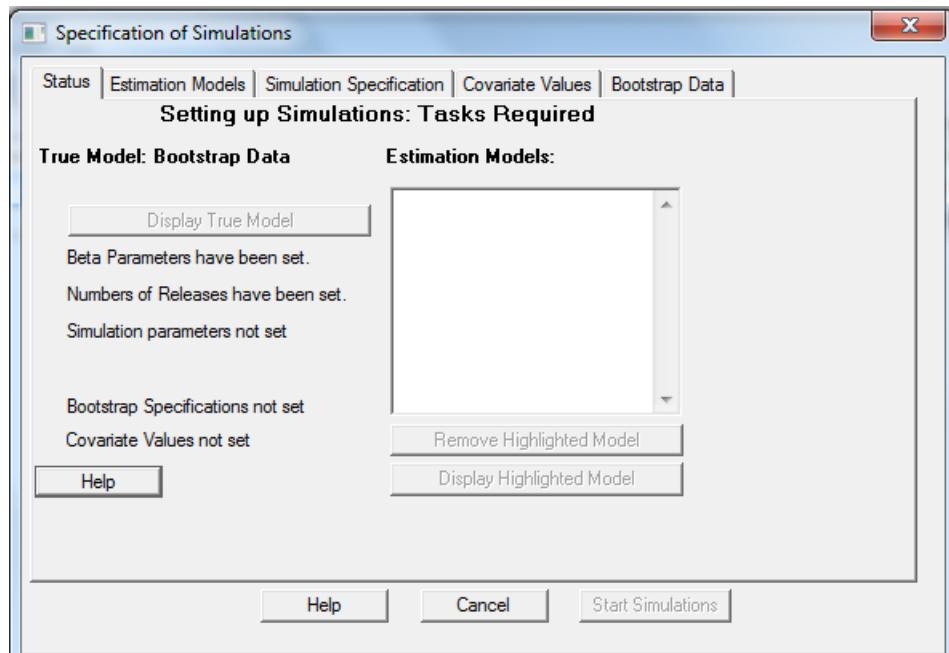
Four individual covariates are specified as **Jdate** = date of capture expressed as the difference between capture date and earliest capture date of the year, **LitID** = Litter ID (to be used to perform the bootstrap resampling, **BWt** = Estimated Birth Weight, and **BHFT** = Estimated Birth Hind Foot Length. So, the first 2 encounter histories above correspond to **LitID** = 1, and the third encounter history is for **LitID** = 2.

To save you from having to build some of the models described in Bishop *et al.* (2008), open up deer-bootstrap.dbf (making sure that deer-bootstrap.fpt is in the same directory).

To run the data bootstrap, select the ‘Bootstrap Data’ menu choice from the ‘Simulation’ menu in the results browser.

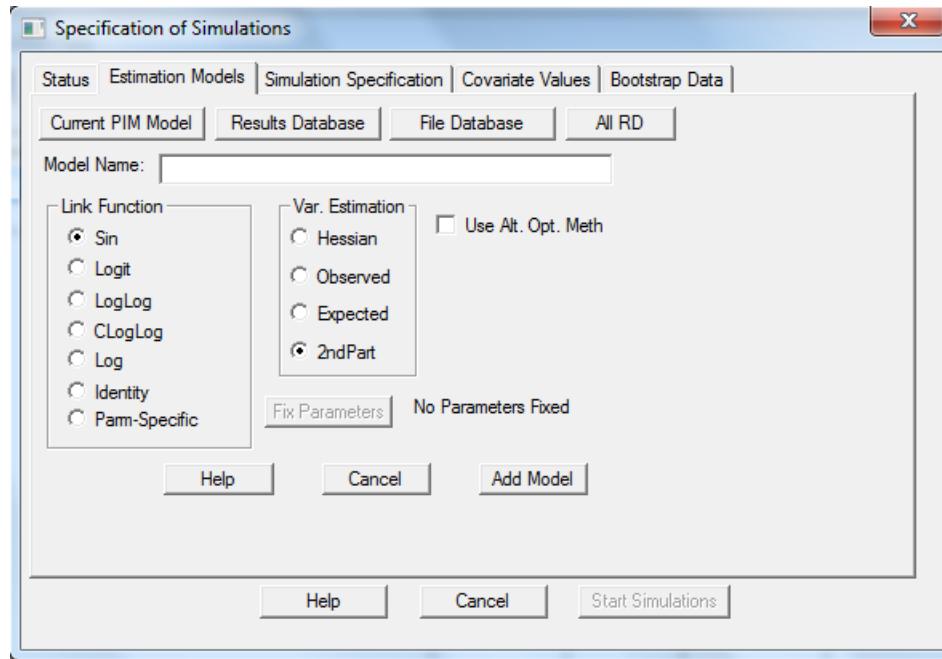


Now you need to provide the information requested in the 5 tab windows (if you are new to running simulations in **MARK**, it might be useful to review Appendix A before continuing – here we only provide a summary of the simulation capabilities in **MARK**). The ‘Status’ tab just summarizes your progress in providing the necessary information, and is the tab window shown first (below). You are returned to the ‘Status’ tab as you move through the other 4 tabs to view your progress.



The ‘Status’ window for a data bootstrap analysis, with all the default values shown. Notice that none of the 5 messages displayed on the left side of the screen indicate you have specified these values. After you have completed the entries in the 4 right tab windows, all of these messages will indicate you have set up the bootstrap analysis. At that time, the ‘Start Simulations’ button will become live for you to proceed with the numerical analysis.

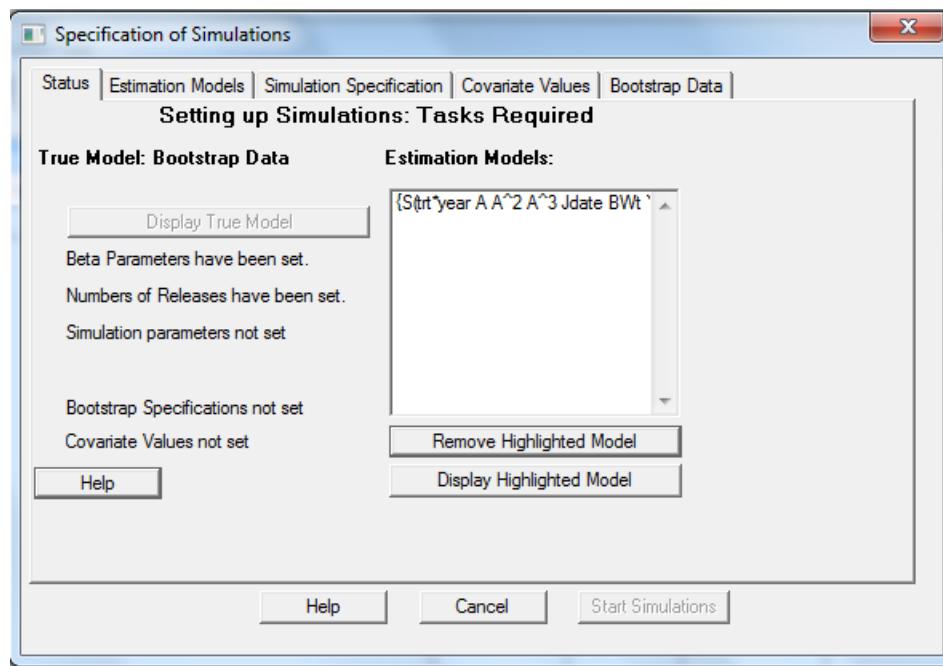
To start, select 1 or more models to estimate from the ‘**Estimation Models**’ tab (below).



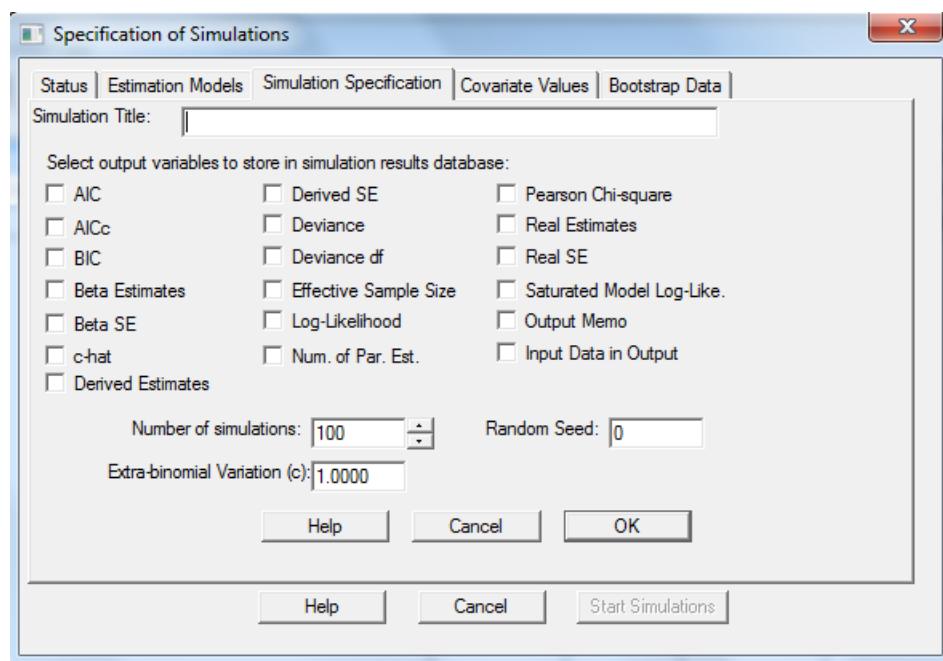
You can select models with the 4 push buttons across the top of the window. The ‘**Current PIM Model**’ button selects the model you currently have constructed in the PIMs (and the associated design matrix if you have created one). The ‘**Results Database**’ push button retrieves the currently highlighted model from the Results Browser. The ‘**File Database**’ button retrieves a model from another MARK file - you will be given a list of the models in the file once you have selected it with the usual Windows file selection windows. For each of these buttons, you have to manually add the model to the list of models for which you want estimates with the ‘**Add Model**’ button in the lower center portion of the tab window. The fourth button for selection models is the ‘**All RD**’ button, which automatically adds all the models in the results database, and is useful for when you want to do model selection using the bootstrap results. You do not need to use the ‘**Add Model**’ button with the ‘**All RD**’ button.

For this example, we’ll assume that the ‘**Current PIM Model**’ button is selected, and that the highlighted model in the results browser is ‘ $S(\text{trt} * \text{year } A \text{ A2 } A \text{ 3 } \text{Jdate } \text{Bwt } \text{Yr} * \text{Bwt})$ ’. The model name is filled in, and the link function is changed to ‘**Logit**’ because this model uses a logit link with the design matrix.

After the model is added with the ‘**Add Model**’ button, the ‘**Status**’ tab now looks like that shown in at the top of the next page. The ‘**Use Alt. Opt. Meth**’ check box allows you to specify that the simulated annealing optimization method be used (which typically takes 10× longer to converge) instead of the default optimization method. Also notice that now the ‘**Remove Highlighted Model**’ and ‘**Display Highlighted Model**’ buttons are active because there is an active estimation model displayed. You can continue to add models to the ‘**Estimation Model**’ list by going back to the estimation models tab. Likewise, the messages show that the ‘**Beta Parameters**’ have been set, and the ‘**Number of Releases**’ have been set. These values are obtained from the data for the estimation model.

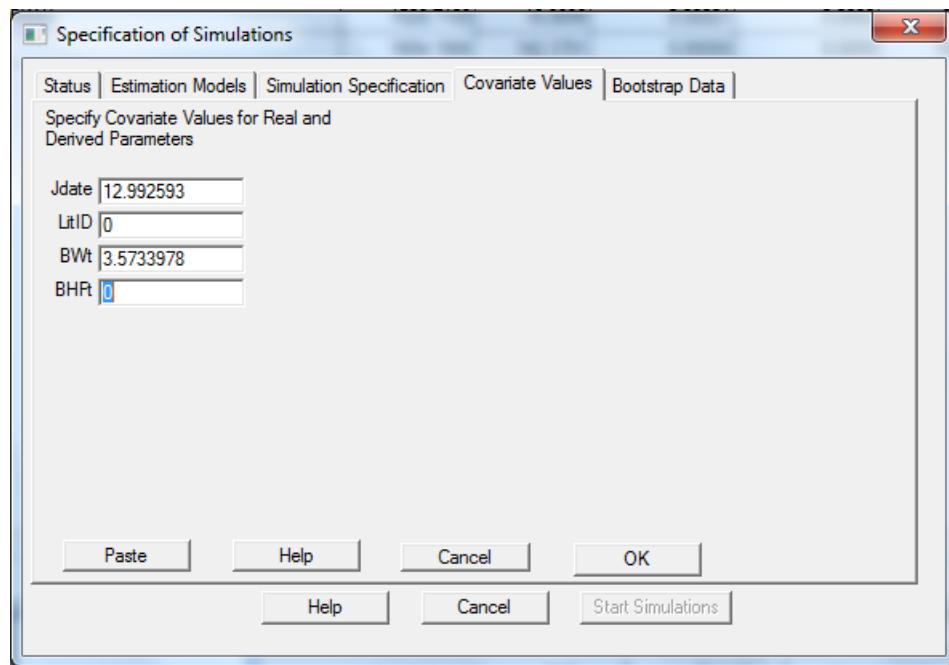


Next, the parameters of the simulation that you want to save in the simulation results database are selected with the Simulation Specification tab (Figure 4). For this example, we are most interested in the derived parameters of the survival estimate across the 6-month interval, so we would check the 'Derived Estimates' and the 'Derived SE' boxes. In addition, we would want to specify more than the 100 simulations, so would likely specify 1,000, or as Bishop *et al.* (2008) did, 10,000. Chernick (1999) suggests a minimum of 5,000 given the speed of today's computers.



The ‘**Extra-binomial Variation (c)**’ box is not useful for this example, but provides a method to add overdispersion to the existing data. That is, the encounter histories are duplicated as necessary to provide data representing the value of c specified. The ‘**Random Seed**’ box allows you to specify a random integer, so that you can repeat the analysis at some future time with exactly the same bootstrap samples. The value of zero that is the default uses the computer’s clock to generate a random value to start the resampling process. The ‘**Simulation Title**’ currently does nothing, so can be left blank. When the appropriate values have been entered, push the ‘**OK**’ button to proceed, which returns you to the Status tab window. The ‘**Status**’ tab window now shows that the ‘**Simulation Parameters**’ have been set.

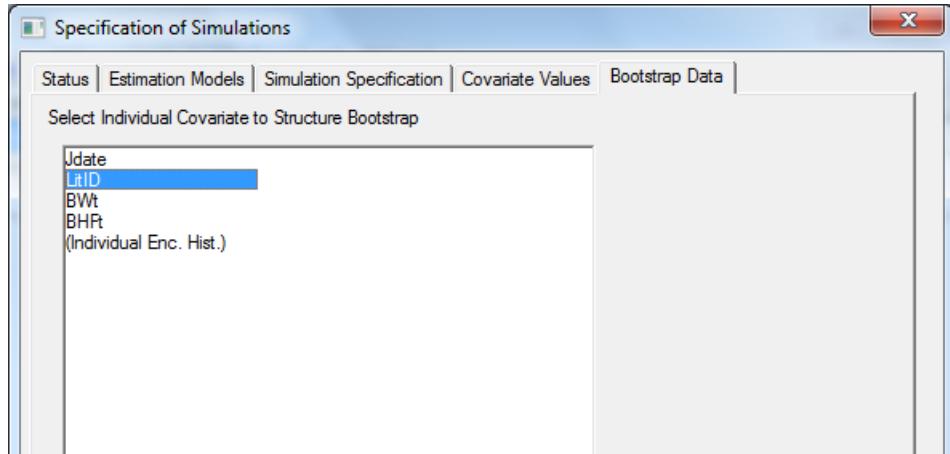
Because the model being estimated uses values of the individual covariates **Jdate** (Julian date of birth) and **BWt** (birth weight) in the design matrix, we need to specify values to use for the bootstrap resampling with the ‘**Covariate Values**’ tab window (below). For the example demonstrated here, these are the values used to compute the real parameter values (26 weekly survival rates for each of 6 groups), and thus the survival rate across the entire 26-week interval.



The ‘**Covariate Values**’ tab window specifies the individual covariate values to use for computing real and derived parameter values. The default values are all zeros, but in the above screen, the mean values for Julian date (**Jdate**) and birth weight (**BWt**) have been entered because these 2 individual covariates are used to compute the real parameters in the model being estimated.

After the ‘**OK**’ button is pushed, you are returned to the ‘**Status**’ tab window, which now specifies that the ‘**Covariate Values**’ have been set.

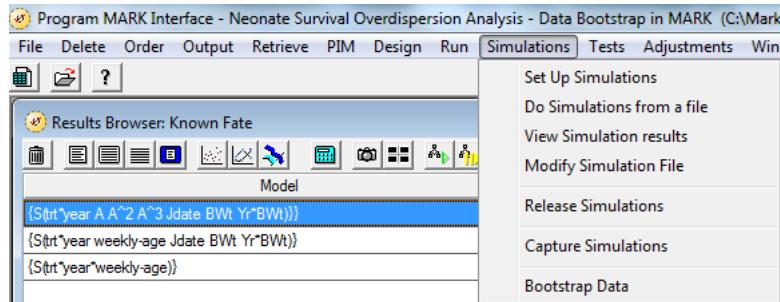
The last tab window is the ‘Bootstrap Data’ tab, which is where you select the individual covariate to be used to resample the data. In our example, the **LitID** covariate (litter ID) is selected because this covariate identifies the individual litters. Although the **LitId** covariate is continuous across all 6 groups, resampling only occurs *within* groups to form the resampled dataset for each group.



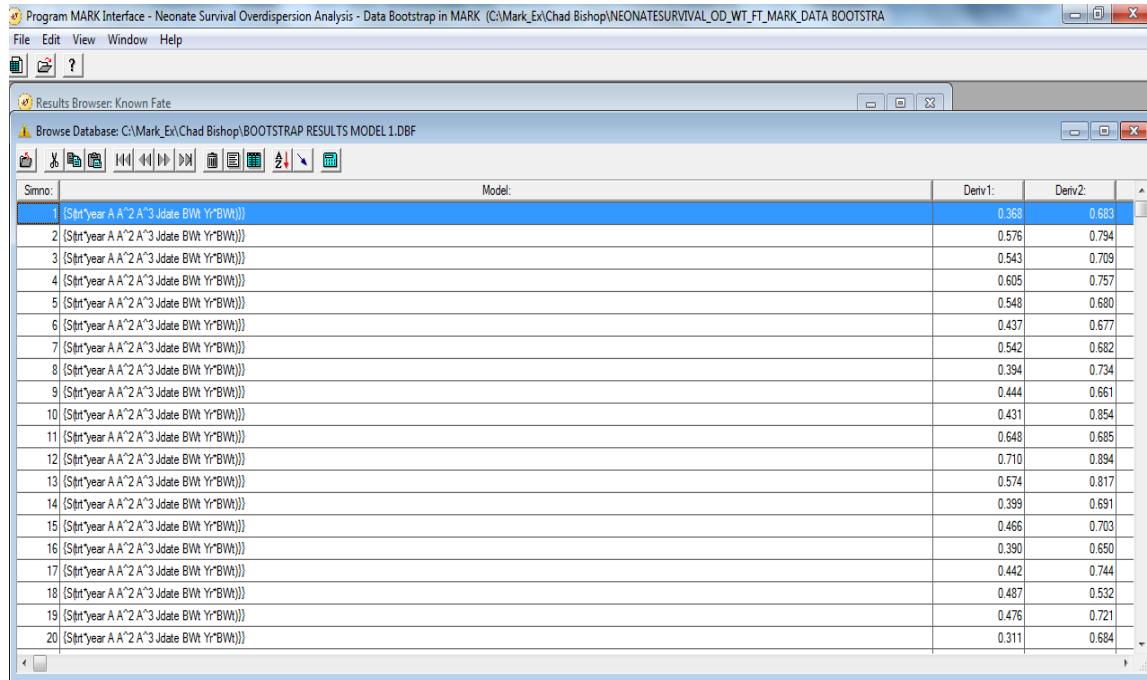
The ‘(Individual Enc. Hist.)’ value is also displayed in the list of individual covariates, but is not actually a covariate. Rather, this option can be used to resample the individual encounter histories. Care must be taken that each encounter history represents the same number of animals, or else the resampling will be a sample biased towards encounter histories with the most animals.

Clicking on the ‘OK’ button returns you to the ‘Status’ tab window, which now shows that ‘Bootstrap Specifications’ have been set. In addition, the ‘Start Simulations’ push button at the lower right corner of the window is ‘live’, meaning that all the information needed to perform the bootstrap sampling process has been provided and you can proceed to the numerical work.

When you click on the ‘Start Simulations’ button, you will be asked to specify a file to store the bootstrap results into. The default file name is SimResults.DBF - so you should probably provide a more informative name. We’ll use Bootstrap Results Model1.DBF. For our example here, the values of the 6 derived parameters and their standard errors will be stored for 1,000 bootstrap resamples. When the simulations finish, you can view the results with the ‘View Simulation Results’ menu choice (shown at the top of the next page). This menu choice takes you to a Windows dialog to select the file where simulations were placed, in this case ‘Bootstrap Results Model1.DBF’.



Select this file, and you get the following window:



The column widths can be changed by placing the cursor on the column boundary and dragging. You will find that you have 6 columns of estimates of the 6 derived parameters (survival rate across the entire 26 weeks for each of the 6 groups) labeled DERIV1 to DERIV6, and you'll also have 6 columns with the standard errors of each of these estimates labeled SED1 to SED6.

Clicking on the right-most button on this window (the 'Calculator' button) will calculate some basic summary statistics and put them into a notepad window.

Statistical Summary of Numerical Variables
(Number of Observations = 1000)

Variable	Mean	Standard Dev.	Standard Error	95% Confidence Interval	
				Lower	Upper
DERIV1	0.526	0.1313	0.0042	0.518	0.534
DERIV2	0.738	0.1099	0.0035	0.732	0.745
DERIV3	0.471	0.0867	0.0027	0.465	0.476
DERIV4	0.613	0.0753	0.0024	0.608	0.618
DERIV5	0.428	0.0749	0.0024	0.423	0.433
DERIV6	0.405	0.0757	0.0024	0.400	0.409
SED1	0.112	0.0094	0.0003	0.112	0.113
SED2	0.090	0.0185	0.0006	0.088	0.091
SED3	0.077	0.0041	0.0001	0.076	0.077
SED4	0.069	0.0036	0.0001	0.069	0.069
SED5	0.073	0.0041	0.0001	0.073	0.073
SED6	0.064	0.0041	0.0001	0.064	0.064

This summary is just a quick look at your results, often enough to verify that everything is working okay. However, to perform a more intensive analysis, you will need to import the DBF file into a statistical package or into Excel. As an example, the following SAS code will import the DBF file into SAS.

```
libname library V9 '.';
title 'Analyze Bootstrap Data from Program MARK';
filename BootData 'BOOTSTRAP RESULTS MODEL 1.DBF';
proc dbf db4=BootData out=BootstrapData;
proc means;
run;
```

For our example here, the necessary information to compute overdispersion is available. For DERIV1, the standard deviation of the 1,000 bootstraps is 0.1313. From the maximum likelihood analysis stored in the results browser, the standard error of this parameter is 0.1139. So an estimate of c for DERIV1 is $(0.1313)^2/(0.1139)^2 = 1.329$, which is in close agreement with the published result of 1.297 in Table 4 of Bishop *et al.* (2008). Because the bootstrap procedure is a random resampling process, you will never generate the exact same results unless the same random seed is specified. Further note that the value of the standard error used above does not match exactly the value of 0.1143 reported by Bishop *et al.* (2008) - probably because of a slightly improved optimization procedure currently in **MARK** over the procedure used by them.

G.2. Extensions

The data bootstrap procedure can also be used for bootstrapping the model selection process. To do this, you must compute estimates for all the models with each of the bootstrap resamples. Note that the '**Status**' tab window (Figure 3) allows more than one estimation model, i.e., you can keep going back to the '**Estimation**' tab window (Figure 2) to add additional models. Or you can use the '**All RD**' button in the '**Estimation**' tab window to add all of the models in your results browser at one time.

Once the bootstrap resampling is completed, you can then evaluate model selection results, although these more complex analyses will require analyzing the bootstrap results in a statistical package. For example the proportion of times a model was selected as the minimum AIC_c model in the bootstrap resamples could be taken as the model's weight. Buckland *et al.* (1997) suggest more sophisticated methods of model selection based on bootstrap resampling.

G.3. Limitations

The bootstrap data procedure only works with encounter history files containing encounter histories, and will not work with the summary input that is allowed for band recovery data or known fate data. You have to convert these formats to the corresponding encounter histories. One trick to do this is to use the residuals output file in Notepad, where the data are given as encounter histories.

One issue that users of the bootstrap data procedure should be aware of is how the frequency counts for encounter histories are handled. These values are maintained with the encounter history, so are not changed by the bootstrap procedure. So, if for some reason you want to bootstrap individual encounter histories, you may want to specify each animal separately, each with a frequency of 1. With the use of the individual covariate to specify the resampling structure, the frequencies can be > 1 and make perfect sense. As an example, consider an input file for known fate data with 1 occasion. An individual covariate of Litter, with values of 1 through 4, is also included in the encounter histories file.

```

11 1 1;
10 2 1;
11 2 2;
10 2 2;
11 1 3;
10 3 4;

```

The above input file shows 4 litters of size 3, 4, 1, and 3, respectively. For this reason, the count frequency for each encounter history can exceed 1, because the litter identifier provides the blocking to bootstrap the encounter histories. Bootstrapping would be performed by resampling from the 4 litters if the '**Litter individual covariate**' was specified in the '**Bootstrap Data**' tab window. If the '**Indiv. Enc. Hist.**' choice was selected (p. G-8), the preceding file would still generate useful bootstrap estimates because the encounter histories with counts > 1 would be resampled according to the encounter history count.

Groups are kept separate in the bootstrap resampling process. Thus, the number of litters is kept constant for each group in the resampled data. Even though your model being estimated might ignore or combine groups, the bootstrap resampling will not ignore these groups. The number of unique blocks of histories within each group will be sampled within that group, and never across attribute groups. Also, for multi-state data, the bootstrap resampler does not distinguish between initial states, so that the conditioning on number of animals starting in each state is lost.

G.4. References

- Anderson, D. R., Burnham, K. P., and White, G. C. (1994) AIC model selection in overdispersed capture-recapture data. *Ecology*, **75**, 1780-1793.
- Bishop, C. J., Freddy, D. J., White, G. C., Watkins, B. E., Stephenson, T. R., and Wolfe, L. L. (2007) Using vaginal implant transmitters to aid in capture of mule deer neonates. *Journal of Wildlife Management*, **71**, 945-954.
- Bishop, C. J., White, G. C., and Lukacs, P. M. (2008) Evaluating dependence among mule deer siblings in fetal and neonatal survival analyses. *Journal of Wildlife Management*, **72**, 1085-1093.
- Bishop, C. J., White, G. C., Freddy, D. J., Watkins, B. E., and Stephenson, T. R. (2009) Effect of enhanced nutrition on mule deer population rate of change. *Wildlife Monographs*, **172**.
- Breslow, N. (1990) Tests of hypotheses in overdispersed Poisson regression and other quasi-likelihood models. *Journal of the American Statistical Association*, **85**, 565-571.
- Buckland, S. T., Burnham, K. P., and Augustin, N. H. (1997) Model selection: An integral part of inference. *Biometrics*, **53**, 603-618.
- Burnham, K. P., and Anderson, D. R. (2002) *Model selection and multimodel inference: a practical information-theoretic approach. Second edition*. Springer-Verlag, New York, New York, USA.
- Carstensen, M., DelGiudice, G. D., and Sampson, B. A. (2003) Using doe behavior and vaginal-implant transmitters to capture neonate whitetailed deer in north-central Minnesota. *Wildlife Society Bulletin*, **31**, 634-641.
- Chernick, M. R. (1999) *Bootstrap Methods – A Practitioner's Guide*. Wiley Series in Probability and Statistics, New York, New York, USA.
- Cox, D. R. (1983) Some remarks on overdispersion. *Biometrika*, **70**, 269-274.
- Cox, D. R., and Snell, E. J. (1989) *Analysis of binary data. Second edition*. Monographs on Statistics and Applied Probability 32. Chapman and Hall/CRC, New York, New York, USA.

- Efron, B., and Tibshirani, R. J. (1993) *An Introduction to the Bootstrap*. Monographs on Statistics and Applied Probability 57, Chapman & Hall, New York, New York, USA.
- Firth, D. (1987) On the efficiency of quasi-likelihood estimation. *Biometrika*, **74**, 233-245.
- Gaillard, J-M., Andersen, R., Delorme, D., and Linnell, J. D. C. (1998) Family effects on growth and survival of juvenile roe deer. *Ecology*, **79**, 2878-2889.
- Hamlin, K. L., Riley, S. J., Pyrah, D., Dood, A. R., and Mackie, R. J. (1984) Relationships among mule deer fawn mortality, coyotes, and alternate prey species during summer. *Journal of Wildlife Management*, **48**, 489-499.
- Jarnemo, A., and Liberg, O. (2005) Red fox removal and roe deer fawn survival – a 14-year study. *Journal of Wildlife Management*, **69**, 1090-1098.
- Schmutz, J. A., Ward, D. H., Sedinger, J. S., and Rexstad, E. A. (1995) Survival estimation and the effects of dependency among animals. *Journal of Applied Statistics*, **22**, 673-681.
- Schwartz, C. C., Haroldson, M. A., and White, G. C. (2006) Survival of cub and yearling grizzly bears in the Greater Yellowstone Ecosystem, 1983-2001. Pages 25-31 in C. C. Schwartz, M. A. Haroldson, G. C. White, R. B. Harris, S. Cherry, K. A. Keating, D. Moody, and C. Servheen, (2006) Temporal, spatial, and environmental influences on the demographics of grizzly bears in the Greater Yellowstone Ecosystem. *Wildlife Monographs*, **161**.
- Sheaffer, S. E., Rusch, D. H., Humburg, D. D., Lawrence, J. S., Zenner, G. G., Gillespie, M. M., Caswell, F. D., Wilds, S., and Yaich, S. C. (2004) Survival, movements, and harvest of eastern prairie population Canada geese. *Wildlife Monographs*, **156**.
- Wedderburn, R. W. M. (1974) Quasi-likelihood functions, generalized linear models, and the Gauss-Newton method. *Biometrika*, **61**, 439-447.
- Whittaker, D. G., and Lindzey, F. G. (1999) Effect of coyote predation on early fawn survival in sympatric deer species. *Wildlife Society Bulletin*, **27**, 256-262.
- Wiens, J. D., Noon, B. R., and Reynolds, R. T. (2006) Post-fledging survival of northern goshawks: the importance of prey abundance, weather, and dispersal. *Ecological Applications*, **16**, 406-418.
- Winterstein, S. R. (1992) Chi-square tests for intrabrood independence when using the Mayfield method. *Journal of Wildlife Management*, **56**, 398-402.