APPENDIX F

Parameter identifiability by data cloning...

There are 2 reasons why a particular model parameter might not be identifiable. The first is because the parameter may be confounded with 1 or more other parameters in the model. An example is the last φ and p parameters in a time-specific Cormack-Jolly-Seber model, where only the product of φ and p can be estimated, but not the unique values of each. In this case, the parameters are not identifiable because of the *structure* of the model. This is referred to as *intrinsic non-identifiability*. The second situation arises either because the data are inadequate, or as an artifact of the parameter being 'poorly estimated' near either the 0 or 1 boundaries. This is referred to as *extrinsic non-identifiability*. While there has been significant progress in formal 'analytical' analysis of intrinsic identifiability (see Gimenez *et al*. 2004, and Hunter & Caswell 2009, and references therein), these methods are complex, and do not apply generally to problems related to inadequate data or parameters estimated near the boundary. Cloning the data is a numerical approach which can be used generally to help identify parameters that are not estimable, for either reason (Lele *et al*. 2007, Lele *et al*. 2010).

To apply this approach, the data are *cloned* by including multiple copies of the encounter histories, i.e., duplicating the encounter histories. In **MARK**, all that needs to be done is to multiply the encounter history frequencies of each group by the number of clones desired. Consider the example of cloning the data 100 times. An encounter history for an analysis with 2 groups and no individual covariates that looks like this

```
11001010010 3 2;
```

could be cloned 100 times by entering the following encounter history:

```
11001010010 300 200;
```

By cloning the data, the sample size is increased without changing the parameter estimates. So, if the original estimates are compared to the cloned estimates, the values of the estimates will remain the same for parameters that are not confounded and are otherwise properly estimated.

How does this help us with problems of 'parameter identifiability'? Here is the key logic step – because the sample size has been increased by cloning, the standard errors of the cloned estimates will be smaller than the original standard errors. The expected result for parameters that are estimable is

$$\text{SE(original)} = \text{SE(cloned)} \times (\text{number of clones})^{0.5}$$

As an example, if the data are cloned 100 times, then the expected standard errors of the cloned data will be 1/10 of the original standard errors. The key word here is 'expected' – if in fact the estimated

standard errors are not a fraction of the original values (by some proportion related to the size of the cloned sample), then this suggests that there may be a problem with parameter identifiability.

F.1. Worked example (1) – the Dippers

The following example will use the European Dipper data (contained in **ed.inp**). Recall that the dipper data consist of 2 groups (males and females) with 7 encounter occasions. So, for a fully time-dependent model, $\{\varphi_{g*t}\, p_{g*t}\}$ model, we expect 22 estimable parameters: 5 survival probabilities for males, 5 survival probabilities for females, 5 encounter probabilities for males, 5 encounter probabilities for females, and 2 confounded estimates of φ_6 and p_7 for each of the 2 groups.

Let's run model $\{\varphi_{g*t}\, p_{g*t}\}$, first using the sine link, and then again using the logit link.

Results Browser: Live Recaptures (CJS)

Model	AICc	Delta AICc	AICc Weight	No. Par.	Deviance
{phi(g*t)p(g*t) -- logit link}	698.2382	0.0000	0.75250	21	71.4740
{phi(g*t)p(g*t) -- sin link}	700.4622	2.2240	0.24750	22	71.4740

We see that both models have the same model deviance (71.4740), but report different number of estimated parameters. Recall from earlier chapters that the 'sin link tends to do better at estimating parameters near the boundaries than the logit link'. Thus, we might suspect that the difference in the number of reported parameters between the two link functions is due to at least one parameter being estimated near the boundary. Let's look at the estimates themselves. From the model fit using the sine link,

mrk5605z.tmp - Notepad

File Edit Format View Help

European dippers (males + females)

Real Function Parameters of {phi(g*t)p(g*t) -- sin link}

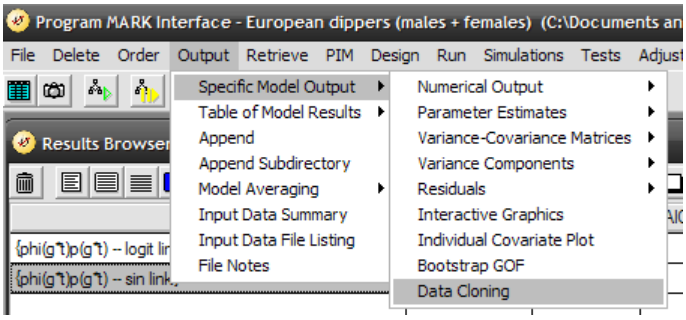
Parameter	Estimate	Standard Error	95% Confidence Interval Lower	Upper
1:Phi	0.6969698	0.2049831	0.2555230	0.9390714
2:Phi	0.4230769	0.0968907	0.2519587	0.6148826
3:Phi	0.5052879	0.0874935	0.3396458	0.6697773
4:Phi	0.6094017	0.0838412	0.4389146	0.7567907
5:Phi	0.5708180	0.0776952	0.4166816	0.7123428
6:Phi	0.7637587	0.0000000	0.7637587	0.7637587
7:Phi	0.7428572	0.2372217	0.2021058	0.9705431
8:Phi	0.4468410	0.0982829	0.2703613	0.6378177
9:Phi	0.4538126	0.0815350	0.3036420	0.6128857
10:Phi	0.6404243	0.0832336	0.4672411	0.7834076
11:Phi	0.6280454	0.0810516	0.4610285	0.7692151
12:Phi	0.6928203	0.0000000	0.6928203	0.6928203
13:p	0.7173914	0.2238537	0.2257356	0.9567136
14:p	1.0000000	0.2616558E-007	0.9999999	1.0000001
15:p	0.9093023	0.0855744	0.5674378	0.9871171
16:p	0.9274193	0.0693437	0.6291491	0.9897162
17:p	0.9358289	0.0616791	0.6607893	0.9909235
18:p	0.7637665	0.0000000	0.7637665	0.7637665
19:p	0.6730771	0.2452306	0.1881302	0.9481657
20:p	0.8600683	0.1263738	0.4397367	0.9796466
21:p	0.9164970	0.0792066	0.5907658	0.9881583
22:p	0.8788732	0.0794409	0.6269547	0.9690646
23:p	0.9283667	0.0684261	0.6330113	0.9898349
24:p	0.6928203	0.0000000	0.6928203	0.6928203

we see that parameter 6 (male survival probability, final interval) is confounded with parameter 18 (male encounter probability, final occasion), and that parameter 12 (female survival probability, final interval) is confounded with parameter 24 (female encounter probability, final occasion). In fact, if you look at the parameter estimates for the model fit using the logit link, you'll see essentially the same thing.

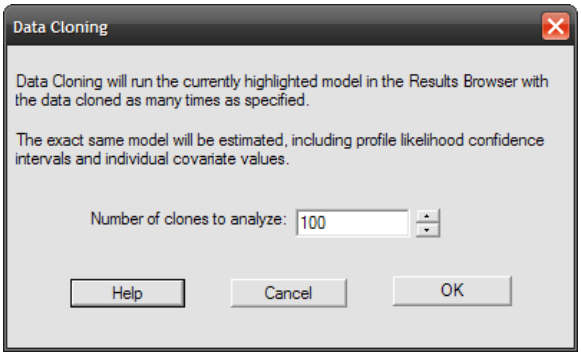
So why 22 reported parameters for the model fit with the sine link, and only 21 for the model fit with the logit link? The answer lies with parameter 14 – encounter probability for the 3rd occasion for males. We see (from the estimates shown at the bottom of the preceding page) that this parameter is estimated at 1.000, with a SE of zero (effectively). If we look at the estimates from the model fit using the logit link, we see essentially the same thing. However, because of the differences in the shape of the respective link functions near the boundaries (in this case, near the 1.0 boundary), **MARK** is unable to derive a robust estimate for survival, and thus, there is some uncertainty as to whether or not survival should be estimated at 1.0, or if this is an artifact of estimation using a particular link function. Specific details on how **MARK** numerically 'counts' parameters are given in the addendum to Chapter 4, and won't be duplicated here. However, what does remain is the question of whether or not this boundary estimate for $\hat{p}_{3,\text{male}} = 1$, or not.

F.1.1. Structural identifiability – confounded parameters

One approach to resolve this problem, specifically, and the problem of parameter identifiability generally is to use data cloning. This is straightforward in **MARK**. First, highlight the model you want to use for estimation in the results browser, i.e., $\{\varphi_{g*t}\, p_{g*t}\}$. For now, select the model fit using the sine link. Then, select the '**`Output | Specific Model Output | Data Cloning`**' menu choice.



This will generate a pop-up menu asking you to specify the number of clones to analyze. The default of 100 is generally sufficient.



Once you click the '**OK**' button, **MARK** will proceed to fit the same model to the cloned data – the results of the model fit to the cloned data, and the estimates of the same model fit to the original data (i.e., your original analysis) will be exported to a single Excel spreadsheet:

	A	B	C	D	E	F	G	H	I	J	K
1	Index	Label	Estimate	SE	LCB	UCB	Estimate >	SE X 100	LCB X 100	UCB X 100	SE Ratio
2	1	Phi	0.69697	0.204983	0.352111	1	0.69697	0.020498	0.657559	0.738037	10.00002
3	2	Phi	0.423077	0.096891	0.246976	0.613749	0.423077	0.009689	0.40417	0.442135	10.00001
4	3	Phi	0.505288	0.087494	0.341276	0.682289	0.505288	0.008749	0.488191	0.522483	10.00001
5	4	Phi	0.609402	0.083841	0.445458	0.774226	0.609402	0.008384	0.592954	0.625815	10.00001
6	5	Phi	0.570818	0.077695	0.419989	0.722316	0.570818	0.00777	0.555586	0.586037	10
7	6	Phi	0.763759	0	0.442094	1	0.763759	15.00366	0.569346	1	0
8	7	Phi	0.742857	0.237222	0.350893	1	0.742857	0.023722	0.697331	0.790467	10.00002
9	8	Phi	0.446841	0.098283	0.274397	0.675965	0.446841	0.009828	0.427794	0.46633	10
10	9	Phi	0.453813	0.081535	0.302868	0.618985	0.453813	0.008154	0.437899	0.469854	9.999996
11	10	Phi	0.640424	0.083234	0.477012	0.804037	0.640424	0.008323	0.624093	0.656717	10
12	11	Phi	0.628045	0.081052	0.469938	0.79321	0.628045	0.008105	0.612161	0.643933	10
13	12	Phi	0.69282	0	0.345022	1	0.69282	6.915748	0.466166	1	0
14	13	p	0.717391	0.223854	0.289025	0.980072	0.717391	0.022385	0.672597	0.760115	10.00002
15	14	p	1	0	0.731975	1	1	1.74E-09	0.996718	1	0
16	15	p	0.909302	0.085574	0.666237	0.994537	0.909302	0.008557	0.891629	0.925162	9.999997
17	16	p	0.927419	0.069344	0.723089	0.995678	0.927419	0.006934	0.913062	0.940242	9.999997
18	17	p	0.935829	0.061679	0.750786	0.9962	0.935829	0.006168	0.923043	0.947222	10.00001
19	18	p	0.763767	0	0.442094	1	0.763766	15.00381	0.569346	1	0
20	19	p	0.673077	0.245231	0.246829	0.975782	0.673077	0.024523	0.624421	0.720265	10.00002
21	20	p	0.860068	0.126374	0.539856	0.991226	0.860068	0.012637	0.834186	0.88367	10.00001
22	21	p	0.916497	0.079207	0.688034	0.994995	0.916497	0.007921	0.900121	0.931163	9.999996
23	22	p	0.878873	0.079441	0.676142	0.978594	0.878873	0.007944	0.862749	0.893878	10
24	23	p	0.928367	0.068426	0.726786	0.995733	0.928367	0.006843	0.9142	0.94102	10
25	24	p	0.69282	0	0.345022	1	0.69282	6.915747	0.466166	1	0
26											

Now, step back a few pages and remember the key 'logic step' – because the sample size has been increased by cloning, the standard errors of the cloned estimates will be smaller than the original standard errors. The expected result for parameters that are estimable is

$$\text{SE(original)} = \text{SE(cloned)} \times (\text{number of clones})^{0.5}$$

In this example, the data were cloned 100 times. Thus, the expected standard errors of the cloned data should be 1/10 of the original standard errors, such that the ratio of the original SE to the cloned SE should be exactly 10. If this ratio is not close to 10 then this suggests that there may be a problem with parameter identifiability.

The ratio of the original estimated SE to the SE estimated from the cloned data is given in spreadsheet column 'K' (labeled as the 'SE Ratio'). We see that for many – but not all – parameters, the ratio is ≈ 10. Let's focus on those ratios which are not particularly close to 10 (highlighted in the spreadsheet). For parameters 6, 12, 18, and 24 the values of the SE Ratio are ≠ 10. We recall that these are the 'confounded' parameters mentioned earlier. We refer to these as *intrinsically* non-identifiable, because the structure of the model prevents them from being identifiable.

F.1.2. 'boundary problems' – data limits and link functions

All the rest of the parameters in the spreadsheet at the top of the preceding page show a SE Ratio = 10 to at least 5 decimal places, except parameter 14, which happens to have been estimated at its boundary, i.e. 1.0. We return here to the question we noted before – is this parameter truly estimable as 1.0, or is it being estimated at 1.0 because (i) it is near the boundary, and (ii) the data, and the link function, are insufficient to resolve the parameter. To differentiate between the two (i.e., to confirm that this parameter is truly estimable), we need to compare its *profile confidence intervals* for the original and cloned data sets. A parameter at a boundary, e.g., a survival estimate equal to 1, will generally have a zero (or at least unrealistically small) standard error. Cloning the data does not change this small standard error.

However, if you have computed *profile likelihood confidence intervals* for this parameter, the profile likelihood confidence intervals for the cloned data will be considerably shorter (assuming you clone a 100 copies) than the original data. So, data cloning is also useful for verifying that a parameter estimated at the boundary is also estimable.

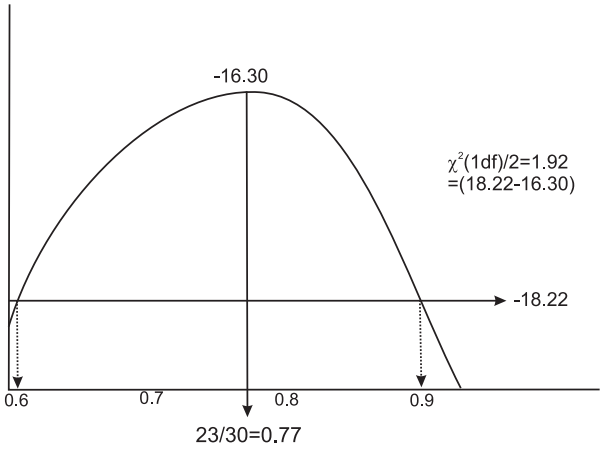
begin sidebar

profile likelihoods: a (brief) re-introduction

The classic MLE approach to variance calculation (for purposes of generating CI) is to use the negative inverse of the 2nd derivative of the MLE evaluated at the MLE. However, the problem with this approach is that it leads to derivation of symmetrical 95% CI which can yield nonsensical results, – especially for parameters that are bounded $[0, 1]$ (e.g., UCI>1).

For example, suppose we release 30 animals, and find 1 survivor. We know from last time that the MLE for the survival probability is $1/30 = 0.03333$. We also know (from Chapter 1) that the classical estimator for the variance, based on the 2nd derivative, is $\widehat{\text{var}}(\hat{p}) = \hat{p}(1-\hat{p})/N = 0.001074$. So, based on this, the 95% CI using classical approaches would be ±1.96(SE), where the SE = square-root of the variance. Thus, given var = 0.001074, the 95% CI would be ±1.96(0.03277), or $[-0.031, 0.098]$. Clearly nonsensical, since the LCI<0.

Fortunately, there is a better way, using something called the *profile likelihood* approach, which makes more explicit use of the shape of the likelihood. Consider the following diagram, which shows the maximum part of the log likelihood for φ, given $N = 30$, $y = 23$ (i.e., 23/30 survive).



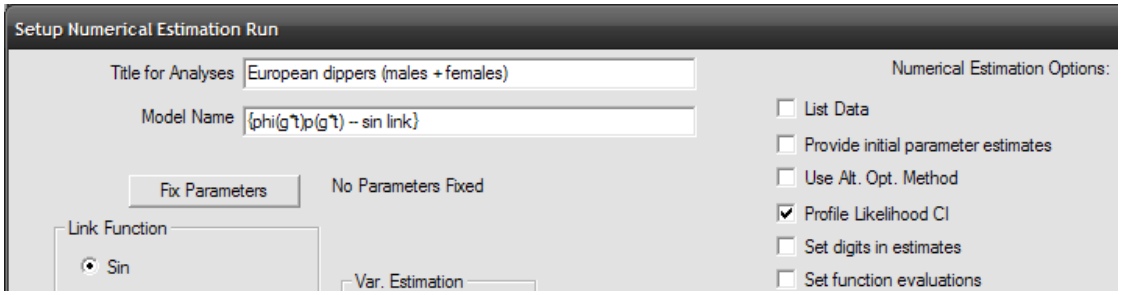
Profile likelihood confidence intervals are based on the log-likelihood function. For a single parameter, likelihood theory shows that the 2 points 1.92 units down from the maximum of the log likelihood function provide a 95% confidence interval when there is no extra-binomial variation (i.e., when $c = 1$). The value 1.92 is half of the critical value $\chi^2_1 = 3.84$. Thus, the same confidence interval

can be computed with the *deviance* by adding 3.84 to the minimum of the deviance function, where the deviance is the log-likelihood multiplied by −2 minus the $-2\ln(\mathcal{L})$ value of the saturated model (for an introduction to saturated models, consult Chapter 5).

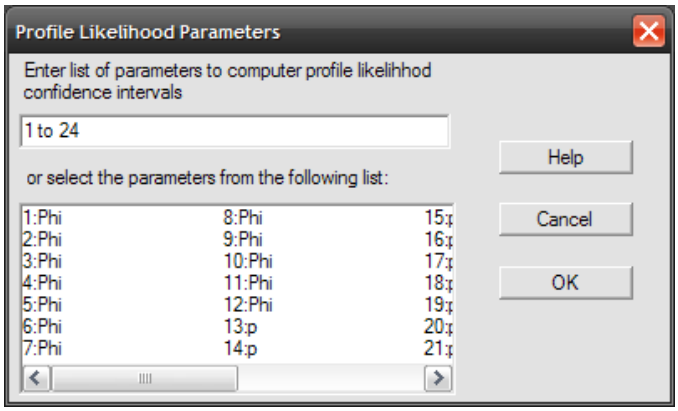
Put another way, we use the critical χ^2 value of 1.92 to derive the *profile* - you take the value of the log likelihood at the maximum (for this example, the maximum occurs at −16.30), add 1.92 to it (yielding −18.22 - note we keep the negative sign here), and look to see where the −18.22 line intersects with the *profile* of the log likelihood function. In this case, we see that the intersection occurs at approximately 0.6 and 0.9. The MLE is 23/30 = 0.767, so clearly, the profile 95% CI is not symmetrical around the MLE. But, it **is** bounded [0, 1]. The profile likelihood is the preferred approach to deriving 95% CI. The biggest limit to using it is computational - it simply takes more work (and computational time) to derive it.

end sidebar

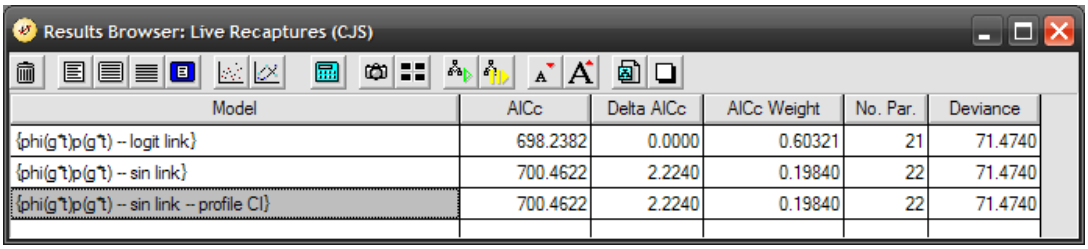
To specify the use of profile likelihoods, you first select the model you want to clone in the results browser, $\{\varphi_{g*t}\ p_{g*t}\}$. Then, re-run the model, this time checking the box specifying '**`profile likelihood CI`**' on the right-hand side of the numerical estimation specification window:



You might add the phrase '`profile CI`' to the title, so you can identify the model when it is added to the browser. When you click the '**`OK`**' button, you'll be asked to specify which parameters you want to estimate the profile likelihood CI for. For this example, we'll specify all the structural parameters in the model (1 to 24).

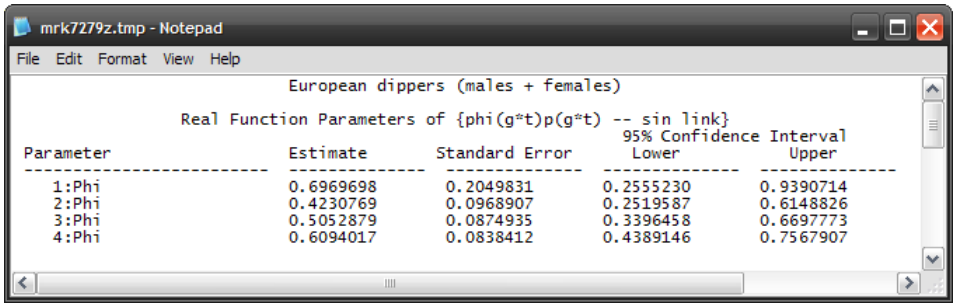


We can see from the results browser (below) that the model fit (i.e., deviance) is identical (in other words, changing the way the CI are estimated doesn't change anything else about the model).

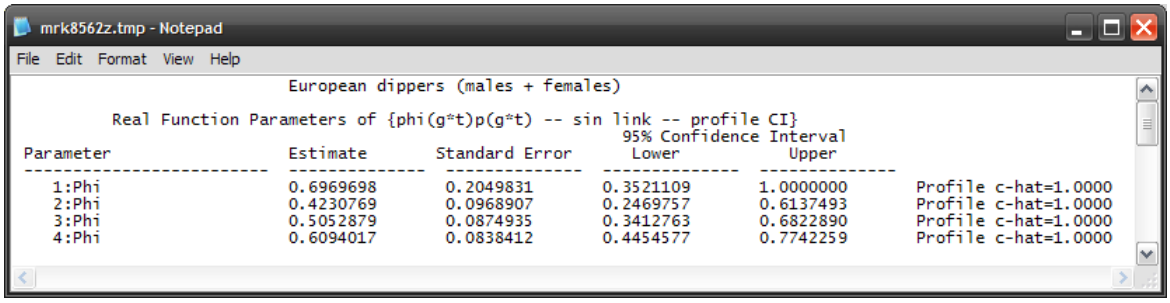


Model	AICc	Delta AICc	AICc Weight	No. Par.	Deviance
{phi(g*t)p(g*t) -- logit link}	698.2382	0.0000	0.60321	21	71.4740
{phi(g*t)p(g*t) -- sin link}	700.4622	2.2240	0.19840	22	71.4740
{phi(g*t)p(g*t) -- sin link -- profile CI}	700.4622	2.2240	0.19840	22	71.4740

A quick comparison of the CI's estimated (for the first 4 survival parameters) using the standard 95% approach



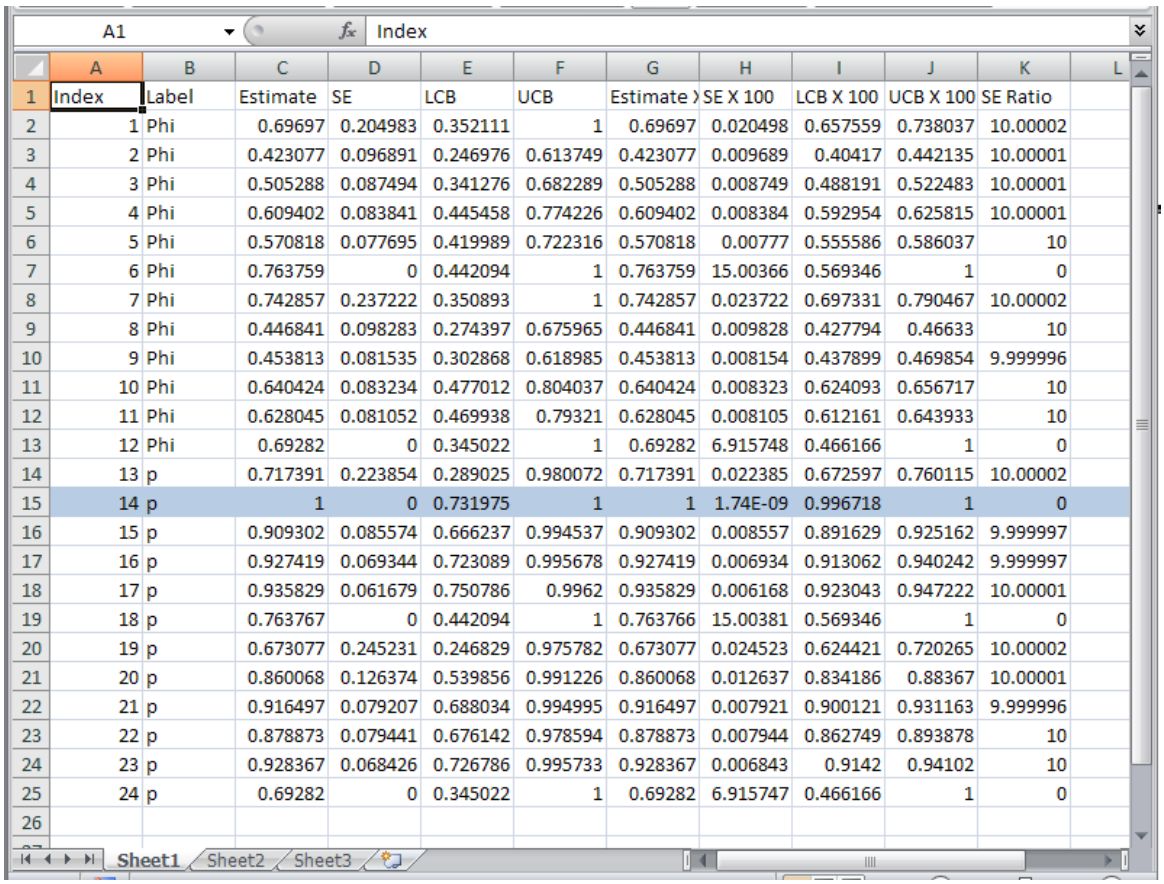
and the profile likelihood CI method



shows clear differences between the two methods for several of the parameters.

OK, back to our problem – parameter estimability. Now that we have the model fit using the profile likelihood CI in the browser, retrieve it to make sure that it is active. Then, select '**`Output | Specific Model Output | Data Cloning`**', and use the default of 100 clones. This will take a bit longer than the first time you 'analyzed the cloned data' (since profile likelihood CI take significantly longer to estimate than the classical CI – which is why it is not the default procedure in **MARK**).

As before, once completed, **MARK** will export the results from both the original analysis and the analysis of the cloned data into an Excel spreadsheet (shown at the top of the next page). We are particularly interested in the CI for parameter 14 (highlighted in the spreadsheet), which we identified earlier as being 'problematic' – is the encounter probability really 1.0, or is that an artifact of the data and the link function used?



Index	Label	Estimate	SE	LCB	UCB	Estimate	SE X 100	LCB X 100	UCB X 100	SE Ratio
1	Phi	0.69697	0.204983	0.352111	1	0.69697	0.020498	0.657559	0.738037	10.00002
2	Phi	0.423077	0.096891	0.246976	0.613749	0.423077	0.009689	0.40417	0.442135	10.00001
3	Phi	0.505288	0.087494	0.341276	0.682289	0.505288	0.008749	0.488191	0.522483	10.00001
4	Phi	0.609402	0.083841	0.445458	0.774226	0.609402	0.008384	0.592954	0.625815	10.00001
5	Phi	0.570818	0.077695	0.419989	0.722316	0.570818	0.00777	0.555586	0.586037	10
6	Phi	0.763759	0	0.442094	1	0.763759	15.00366	0.569346	1	0
7	Phi	0.742857	0.237222	0.350893	1	0.742857	0.023722	0.697331	0.790467	10.00002
8	Phi	0.446841	0.098283	0.274397	0.675965	0.446841	0.009828	0.427794	0.46633	10
9	Phi	0.453813	0.081535	0.302868	0.618985	0.453813	0.008154	0.437899	0.469854	9.999996
10	Phi	0.640424	0.083234	0.477012	0.804037	0.640424	0.008323	0.624093	0.656717	10
11	Phi	0.628045	0.081052	0.469938	0.79321	0.628045	0.008105	0.612161	0.643933	10
12	Phi	0.69282	0	0.345022	1	0.69282	6.915748	0.466166	1	0
13	p	0.717391	0.223854	0.289025	0.980072	0.717391	0.022385	0.672597	0.760115	10.00002
14	p	1	0	0.731975	1	1	1.74E-09	0.996718	1	0
15	p	0.909302	0.085574	0.666237	0.994537	0.909302	0.008557	0.891629	0.925162	9.999997
16	p	0.927419	0.069344	0.723089	0.995678	0.927419	0.006934	0.913062	0.940242	9.999997
17	p	0.935829	0.061679	0.750786	0.9962	0.935829	0.006168	0.923043	0.947222	10.00001
18	p	0.763767	0	0.442094	1	0.763766	15.00381	0.569346	1	0
19	p	0.673077	0.245231	0.246829	0.975782	0.673077	0.024523	0.624421	0.720265	10.00002
20	p	0.860068	0.126374	0.539856	0.991226	0.860068	0.012637	0.834186	0.88367	10.00001
21	p	0.916497	0.079207	0.688034	0.994995	0.916497	0.007921	0.900121	0.931163	9.999996
22	p	0.878873	0.079441	0.676142	0.978594	0.878873	0.007944	0.862749	0.893878	10
23	p	0.928367	0.068426	0.726786	0.995733	0.928367	0.006843	0.9142	0.94102	10
24	p	0.69282	0	0.345022	1	0.69282	6.915747	0.466166	1	0

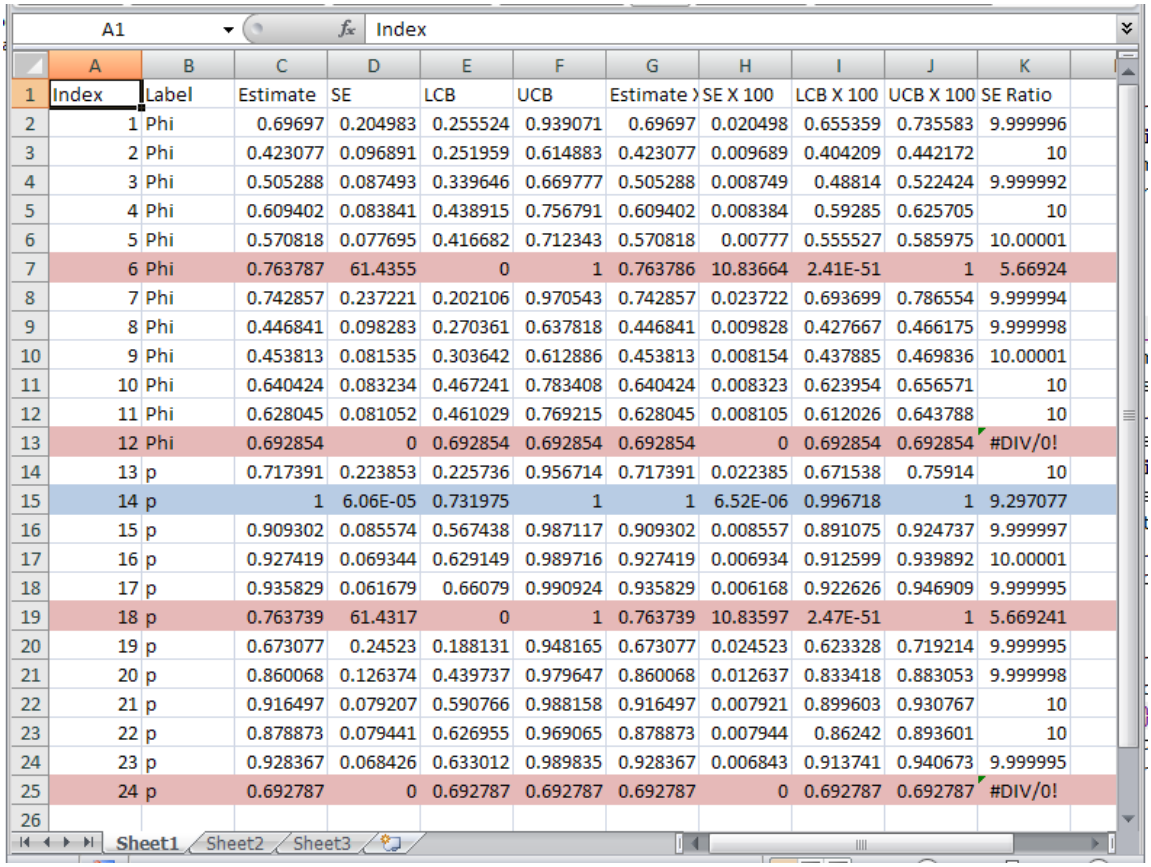
You can now see that the profile interval for parameter 14 has shortened considerably for the cloned data, with the lower bound changing from 0.732 to 0.997, indicating that this parameter was actually being estimated. In other words, parameter 14 was *extrinsically* non-identifiable. In contrast, the 4 intrinsically confounded parameters (6, 12, 18 and 24) still show a relatively wide profile likelihood confidence interval for the cloned analysis, only slightly reduced from the original values.

F.1.3. Choice of link function – does it matter?

In the preceding, we considered data cloning based on model $\{\varphi_{g*t}\, p_{g*t}\}$, fit using the sine link. Recall that the number of reported parameters for this model differed depending on whether or not the sine (22 parameters) or the logit link was used (21 parameters). As noted earlier, the difference (for the dipper example) is due to parameter 14 – the encounter probability for the 3rd occasion for males. While neither the sine or logit link function estimate this parameter particularly 'well', because of the differences in the shape of the respective link functions near the boundaries (in this case, near the 1.0 boundary), **MARK** is both unable to derive a robust estimate of the parameter, and (more to the point), whether or not the parameter is estimable, and should be counted. Again, specific details on how **MARK** numerically 'counts' parameters are given in the addendum to Chapter 4.

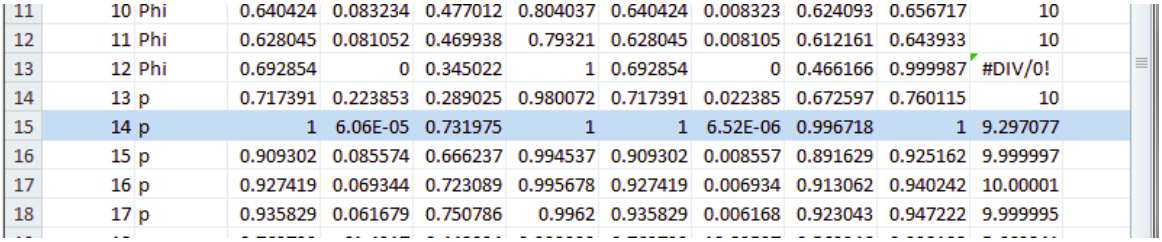
However, while this is 'interesting', our immediate interest is whether or not the choice of the link function influences the data cloning applications we're introducing here. Here, we replicate the 'cloning', but using model $\{\varphi_{g*t}\, p_{g*t}\}$ fit using the logit link. To do this, select the appropriate model in the browser, and retrieve it (to make sure it is the currently 'active' model). Then, go ahead and run the data cloning

as before. Here is the Excel spreadsheet containing the results. We have highlighted the confounded parameters (in red), and the 'problem' parameter 14 (in blue).



Index	Label	Estimate	SE	LCB	UCB	Estimate X	SE X 100	LCB X 100	UCB X 100	SE Ratio
1	Phi	0.69697	0.204983	0.255524	0.939071	0.69697	0.020498	0.655359	0.735583	9.999996
2	Phi	0.423077	0.096891	0.251959	0.614883	0.423077	0.009689	0.404209	0.442172	10
3	Phi	0.505288	0.087493	0.339646	0.669777	0.505288	0.008749	0.48814	0.522424	9.999992
4	Phi	0.609402	0.083841	0.438915	0.756791	0.609402	0.008384	0.59285	0.625705	10
5	Phi	0.570818	0.077695	0.416682	0.712343	0.570818	0.00777	0.555527	0.585975	10.00001
6	Phi	0.763787	61.4355	0	1	0.763786	10.83664	2.41E-51	1	5.66924
7	Phi	0.742857	0.237221	0.202106	0.970543	0.742857	0.023722	0.693699	0.786554	9.999994
8	Phi	0.446841	0.098283	0.270361	0.637818	0.446841	0.009828	0.427667	0.466175	9.999998
9	Phi	0.453813	0.081535	0.303642	0.612886	0.453813	0.008154	0.437885	0.469836	10.00001
10	Phi	0.640424	0.083234	0.467241	0.783408	0.640424	0.008323	0.623954	0.656571	10
11	Phi	0.628045	0.081052	0.461029	0.769215	0.628045	0.008105	0.612026	0.643788	10
12	Phi	0.692854	0	0.692854	0.692854	0.692854	0	0.692854	0.692854	#DIV/0!
13	p	0.717391	0.223853	0.225736	0.956714	0.717391	0.022385	0.671538	0.75914	10
14	p	1	6.06E-05	0.731975	1	1	6.52E-06	0.996718	1	9.297077
15	p	0.909302	0.085574	0.567438	0.987117	0.909302	0.008557	0.891075	0.924737	9.999997
16	p	0.927419	0.069344	0.629149	0.989716	0.927419	0.006934	0.912599	0.939892	10.00001
17	p	0.935829	0.061679	0.66079	0.990924	0.935829	0.006168	0.922626	0.946909	9.999995
18	p	0.763739	61.4317	0	1	0.763739	10.83597	2.47E-51	1	5.669241
19	p	0.673077	0.24523	0.188131	0.948165	0.673077	0.024523	0.623328	0.719214	9.999995
20	p	0.860068	0.126374	0.439737	0.979647	0.860068	0.012637	0.833418	0.883053	9.999998
21	p	0.916497	0.079207	0.590766	0.988158	0.916497	0.007921	0.899603	0.930767	10
22	p	0.878873	0.079441	0.626955	0.969065	0.878873	0.007944	0.86242	0.893601	10
23	p	0.928367	0.068426	0.633012	0.989835	0.928367	0.006843	0.913741	0.940673	9.999995
24	p	0.692787	0	0.692787	0.692787	0.692787	0	0.692787	0.692787	#DIV/0!

As we saw when we applied data cloning to the model fit with the sine link, the ratio of the SE for the confounded parameters is not equal to 10, which we interpret as diagnostic of a confounding problem. For parameter 14, the ratio is not directly informative – we need to rerun the model using the profile likelihood approach.

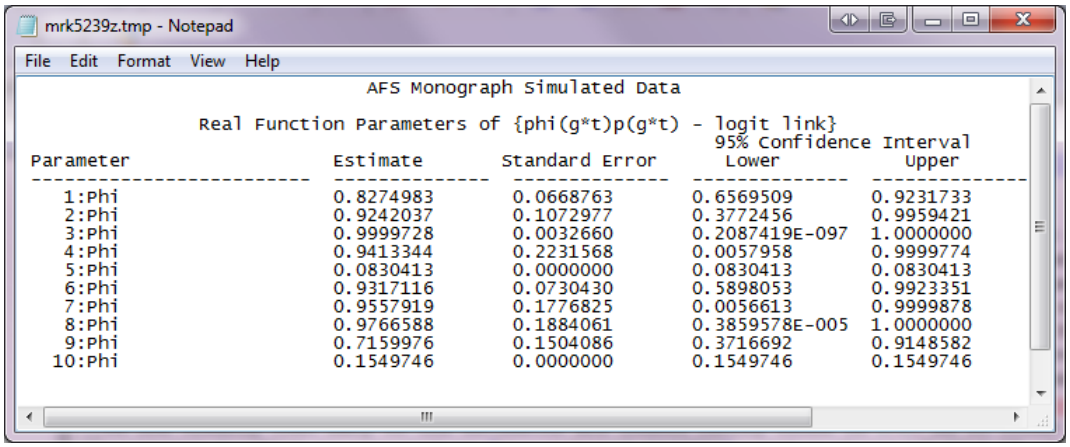


10	Phi	0.640424	0.083234	0.477012	0.804037	0.640424	0.008323	0.624093	0.656717	10
11	Phi	0.628045	0.081052	0.469938	0.79321	0.628045	0.008105	0.612161	0.643933	10
12	Phi	0.692854	0	0.345022	1	0.692854	0	0.466166	0.999987	#DIV/0!
13	p	0.717391	0.223853	0.289025	0.980072	0.717391	0.022385	0.672597	0.760115	10
14	p	1	6.06E-05	0.731975	1	1	6.52E-06	0.996718	1	9.297077
15	p	0.909302	0.085574	0.666237	0.994537	0.909302	0.008557	0.891629	0.925162	9.999997
16	p	0.927419	0.069344	0.723089	0.995678	0.927419	0.006934	0.913062	0.940242	10.00001
17	p	0.935829	0.061679	0.750786	0.9962	0.935829	0.006168	0.923043	0.947222	9.999995

As was the case when we applied this approach to the model fit with the sine link, we see that the CI for parameter 14 gets significantly smaller when the data are cloned (lower CI changes from 0.732 to 0.998). This is consistent with our determination that in fact parameter 14 is being correctly estimated at 1.000. So, in this case (and probably generally), the choice of the link function (sin or logit, typically) shouldn't make any difference.

F.2. Worked example (2) – AFS monograph example

For this example we use a data set which is distributed with **MARK** (`\examples\AFSMONGR.DBF`). These live encounter data were collected over 6 sampling occasions, with 2 groups (control and treatment). The `.DBF` file distributed with **MARK** shows a series of 'standard' CJS models, fit with different link functions and design matrix structures. As in the first example, we consider model $\{\varphi_{g*t}\, p_{g*t}\}$. We know that there will be confounding of the final estimates of survival and encounter probability for each of the two groups. Here, we focus on the problem of estimates at or near the boundary. For these data, survival probability over an interval is anticipated to be relatively high. It is just this sort of situation which can lend itself to the 'boundary estimate' problem. If you look at the survival estimates for model $\{\varphi_{g*t}\, p_{g*t}\}$, fit using the logit link (shown below), we see clearly that many of the estimates are indeed fairly close to 1.0. We'll focus in particular on survival parameter 3, which is reported as $\hat{\varphi} = 0.9999729$, with a classical 95% CI of $[0, 1]$. Needless to say, we have significant uncertainty about the estimation of this parameter.



First, let's see if re-running the model, but using a profile likelihood CI, helps us at all. When a profile likelihood interval is requested for parameter 3, the following results are obtained: $\hat{\varphi} = 0.999728$ (essentially identical to what was reported, above). The profile CI is [0.7728344, 0.9999728]. Here we see that the optimization procedure used in generating the profile CI was able to move the LCI away from the boundary at 0. Similar results are obtained if the sine link is used.

However, our purpose here is to consider the application of data cloning to the sort of 'boundary estimate' problem. First, we re-run model $\{\varphi_{g*t}\, p_{g*t}\}$, using the logit link, first requesting the profile likelihood CI for parameter 3. Now, we re-run the model, with the data cloned 100 times. From the generated Excel spreadsheet, the LCI for the cloned data is reported as 0.990629. You can now see that the profile interval for parameter 3 has shortened considerably for the cloned data, with the lower bound changing from 0.773 to 0.991, indicating that this parameter was actually being estimated.

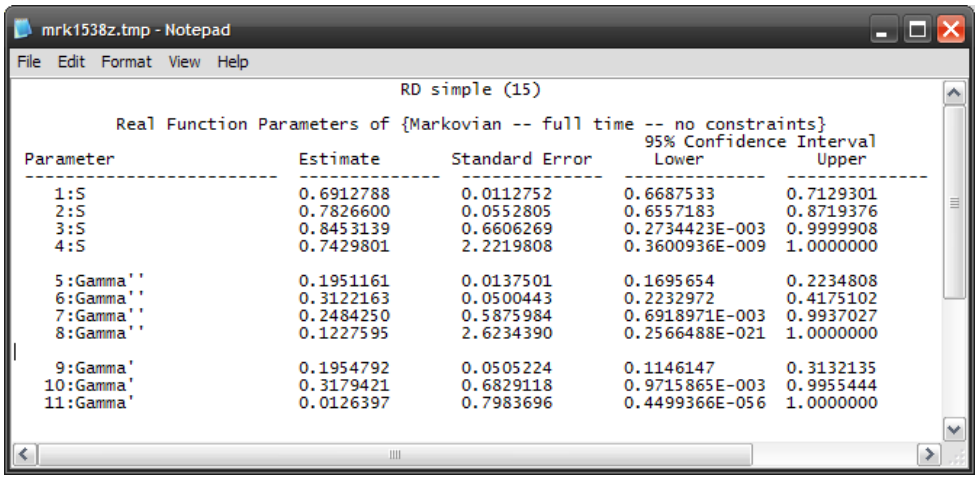
F.3. Worked example (3) – robust design example

In section 16.3 of Chapter 16, we introduced an extension of the classical 'robust design' to account for temporary movements in and out of the sampled population (Kendall *et al.* 1995a, 1997). In the extended robust design, we introduced two different parameters to describe temporary movements into and out of the sample: γ' and γ'' (read as 'gamma-prime' and 'gamma-double-prime', respectively). Here we are not concerned with the specific details of this model. Instead, our focus is on parameter

identifiability, using the 'Markovian movement' model as an example. As discussed in Chapter 16, to provide identifiability of the parameters for the *Markovian emigration* model when parameters are time-specific, Kendall *et al*. (1997) stated that γ''_k and γ'_k need to be set equal to γ''_t and γ'_t, respectively, for some earlier period. Otherwise these parameters are confounded with S_{t-1}. They suggested setting them equal to γ''_{k-1} and γ'_{k-1}, respectively.

Can we 'demonstrate' the confounding of S_{t-1} with the γ parameters in the absence of the suggested constraints (and, by extension, can we show that the constraints do in fact 'solve' the confounding), using data cloning? To address this, we'll re-visit the 'simple robust design' example introduced in section 16.6.1 in Chapter 16 (the data are contained in **`rd_simple1.inp`**). From the analysis of these data described in Chapter 16, here are the survival and γ estimates from fitting a Markovian time-dependent model *without* the 'identifiability constraints' suggested by Kendall *et al*. (1997). The model was fit using a logit link.

Based on evaluation of the SE and CI of the parameter estimates (shown below), we see some 'suggestion' that some of the parameters are 'not well estimated', and may be confounded.



mrk1538z.tmp - Notepad

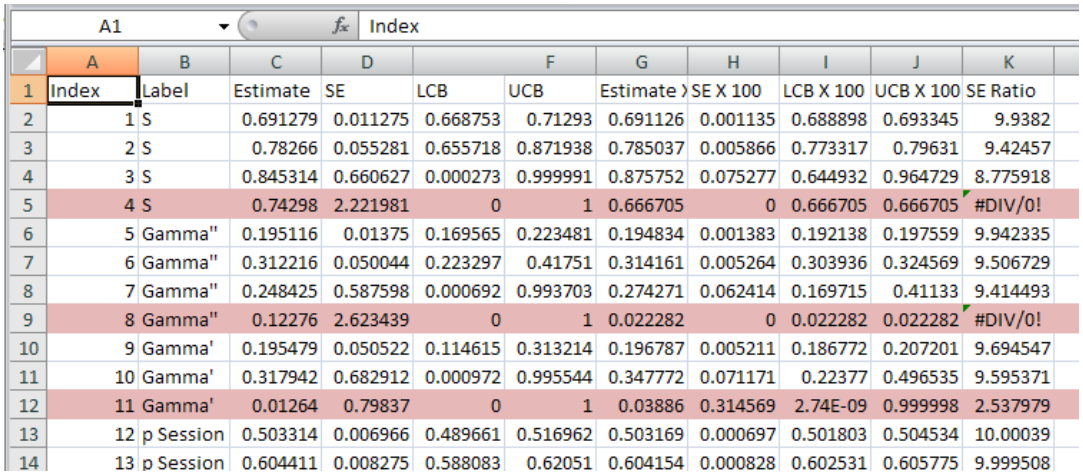
File Edit Format View Help

RD simple (15)

Real Function Parameters of {Markovian -- full time -- no constraints}

Parameter	Estimate	Standard Error	95% Confidence Interval Lower	Upper
1:S	0.6912788	0.0112752	0.6687533	0.7129301
2:S	0.7826600	0.0552805	0.6557183	0.8719376
3:S	0.8453139	0.6606269	0.2734423E-003	0.9999908
4:S	0.7429801	2.2219808	0.3600936E-009	1.0000000
5:Gamma''	0.1951161	0.0137501	0.1695654	0.2234808
6:Gamma''	0.3122163	0.0500443	0.2232972	0.4175102
7:Gamma''	0.2484250	0.5875984	0.6918971E-003	0.9937027
8:Gamma''	0.1227595	2.6234390	0.2566488E-021	1.0000000
9:Gamma'	0.1954792	0.0505224	0.1146147	0.3132135
10:Gamma'	0.3179421	0.6829118	0.9715865E-003	0.9955444
11:Gamma'	0.0126397	0.7983696	0.4499366E-056	1.0000000

We can use data cloning to explore this directly. In the spreadsheet shown below



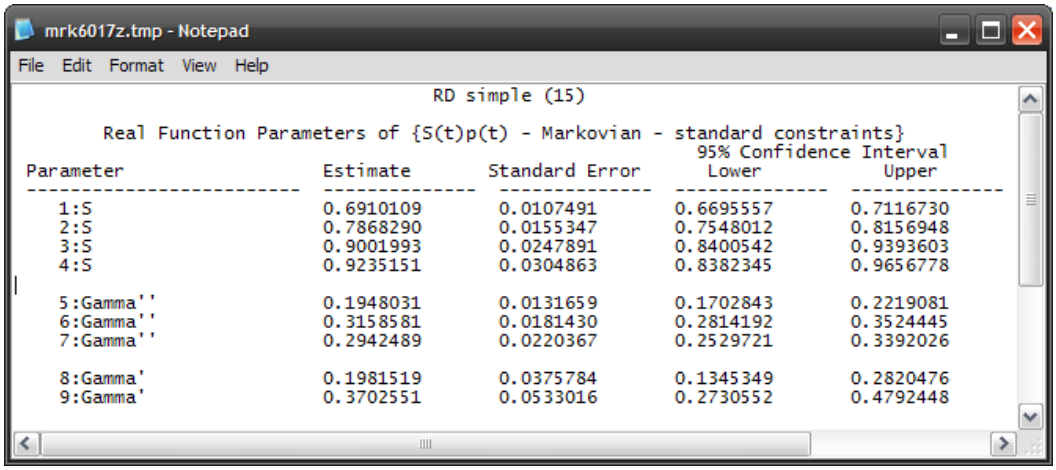
A1 fx Index

	A	B	C	D		F	G	H	I	J	K
1	Index	Label	Estimate	SE	LCB	UCB	Estimate >	SE X 100	LCB X 100	UCB X 100	SE Ratio
2	1	S	0.691279	0.011275	0.668753	0.71293	0.691126	0.001135	0.688898	0.693345	9.9382
3	2	S	0.78266	0.055281	0.655718	0.871938	0.785037	0.005866	0.773317	0.79631	9.42457
4	3	S	0.845314	0.660627	0.000273	0.999991	0.875752	0.075277	0.644932	0.964729	8.775918
5	4	S	0.74298	2.221981	0	1	0.666705	0	0.666705	0.666705	#DIV/0!
6	5	Gamma"	0.195116	0.01375	0.169565	0.223481	0.194834	0.001383	0.192138	0.197559	9.942335
7	6	Gamma"	0.312216	0.050044	0.223297	0.41751	0.314161	0.005264	0.303936	0.324569	9.506729
8	7	Gamma"	0.248425	0.587598	0.000692	0.993703	0.274271	0.062414	0.169715	0.41133	9.414493
9	8	Gamma"	0.12276	2.623439	0	1	0.022282	0	0.022282	0.022282	#DIV/0!
10	9	Gamma'	0.195479	0.050522	0.114615	0.313214	0.196787	0.005211	0.186772	0.207201	9.694547
11	10	Gamma'	0.317942	0.682912	0.000972	0.995544	0.347772	0.071171	0.22377	0.496535	9.595371
12	11	Gamma'	0.01264	0.79837	0	1	0.03886	0.314569	2.74E-09	0.999998	2.537979
13	12	p Session	0.503314	0.006966	0.489661	0.516962	0.503169	0.000697	0.501803	0.504534	10.00039
14	13	p Session	0.604411	0.008275	0.588083	0.62051	0.604154	0.000828	0.602531	0.605775	9.999508

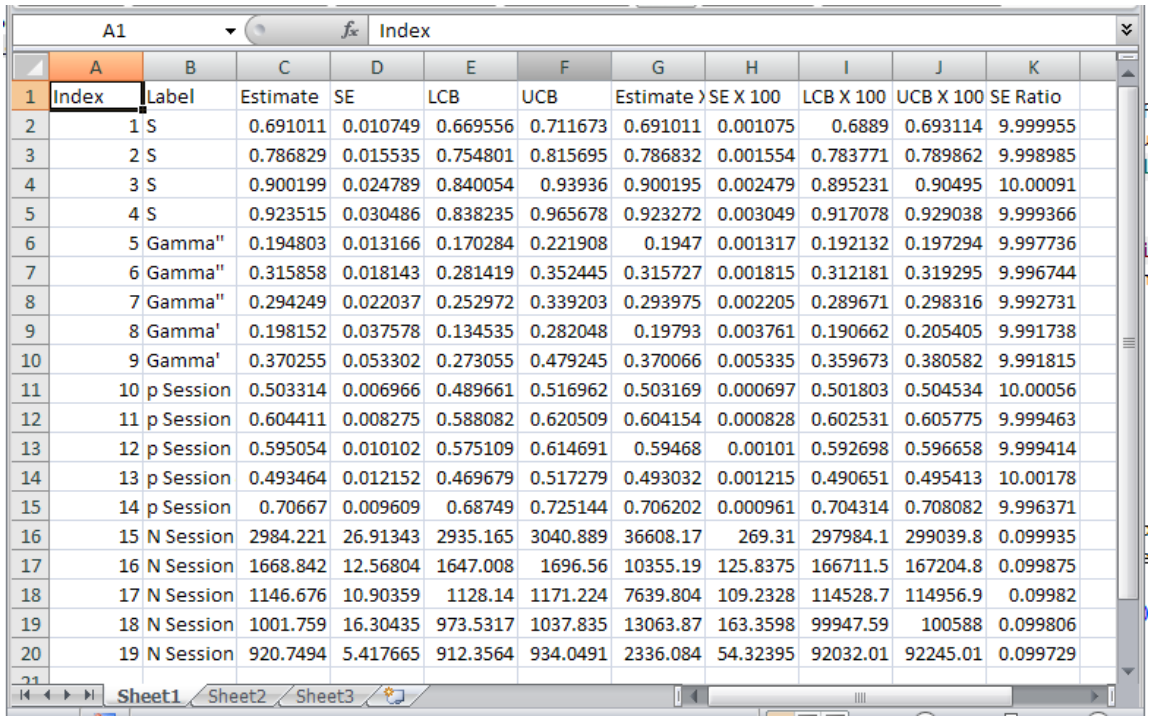
we see clear evidence that parameters S_4, γ''_4 and γ'_4 (highlighted in red) are confounded. If we look

closely, we also see that there is some evidence that the survival estimates might be biased, since the SE ratio for $S_1 \rightarrow S_3$ is somewhat <10, with a negative trend approaching the confounded parameter. In addition, we see that the SE ratio for abundance estimates $\hat{N}$ are all $\ll 10$.

Does fitting the constraint $\gamma''_{k-1} = \gamma''_k$ and $\gamma'_{k-1} = \gamma'_k$ (*sensu* Kendall *et al*. 1997) solve the problem of confounding with S_k? Here are the estimates for S and γ from the constrained model:



We see that, indeed, all of the parameters seem to be 'well estimated'. We can confirm this assessment by applying the data cloning approach to this model, we see (below) that all of the S and unconstrained γ parameters are indeed well-estimated.



	A	B	C	D	E	F	G	H	I	J	K
1	Index	Label	Estimate	SE	LCB	UCB	Estimate	SE X 100	LCB X 100	UCB X 100	SE Ratio
2	1	S	0.691011	0.010749	0.669556	0.711673	0.691011	0.001075	0.6889	0.693114	9.999955
3	2	S	0.786829	0.015535	0.754801	0.815695	0.786832	0.001554	0.783771	0.789862	9.998985
4	3	S	0.900199	0.024789	0.840054	0.93936	0.900195	0.002479	0.895231	0.90495	10.00091
5	4	S	0.923515	0.030486	0.838235	0.965678	0.923272	0.003049	0.917078	0.929038	9.999366
6	5	Gamma"	0.194803	0.013166	0.170284	0.221908	0.1947	0.001317	0.192132	0.197294	9.997736
7	6	Gamma"	0.315858	0.018143	0.281419	0.352445	0.315727	0.001815	0.312181	0.319295	9.996744
8	7	Gamma"	0.294249	0.022037	0.252972	0.339203	0.293975	0.002205	0.289671	0.298316	9.992731
9	8	Gamma'	0.198152	0.037578	0.134535	0.282048	0.19793	0.003761	0.190662	0.205405	9.991738
10	9	Gamma'	0.370255	0.053302	0.273055	0.479245	0.370066	0.005335	0.359673	0.380582	9.991815
11	10	p Session	0.503314	0.006966	0.489661	0.516962	0.503169	0.000697	0.501803	0.504534	10.00056
12	11	p Session	0.604411	0.008275	0.588082	0.620509	0.604154	0.000828	0.602531	0.605775	9.999463
13	12	p Session	0.595054	0.010102	0.575109	0.614691	0.59468	0.00101	0.592698	0.596658	9.999414
14	13	p Session	0.493464	0.012152	0.469679	0.517279	0.493032	0.001215	0.490651	0.495413	10.00178
15	14	p Session	0.70667	0.009609	0.68749	0.725144	0.706202	0.000961	0.704314	0.708082	9.996371
16	15	N Session	2984.221	26.91343	2935.165	3040.889	36608.17	269.31	297984.1	299039.8	0.099935
17	16	N Session	1668.842	12.56804	1647.008	1696.56	10355.19	125.8375	166711.5	167204.8	0.099875
18	17	N Session	1146.676	10.90359	1128.14	1171.224	7639.804	109.2328	114528.7	114956.9	0.09982
19	18	N Session	1001.759	16.30435	973.5317	1037.835	13063.87	163.3598	99947.59	100588	0.099806
20	19	N Session	920.7494	5.417665	912.3564	934.0491	2336.084	54.32395	92032.01	92245.01	0.099729

F.4. Data cloning and 'unbounded' parameters

In the preceding examples, we considered only parameters that are bounded on the interval [0, 1]. Can data cloning be robustly applied to estimation of a parameter that is not [0, 1] bounded (say, abundance, or λ or f in a Pradel model)? There are several considerations in answering this question.

First, remember that the methods described here using data cloning are largely intended to help identify parameters which are (i) confounded (i.e., intrinsically non-identifiable), or (ii) which might be poorly estimated given the data (i.e., extrinsically non-identifiable), generally *because they are estimated at either the* [0, 1] *boundary*. Clearly, neither of these criterion apply to some model parameters for some data types, say, abundance N, since estimates of N are not confounded with structural parameters in any model, and since boundary issues don't apply in the usual sense (since M_{t+1} must be the lower boundary, and $M_{t+1} > 0$; see Chapter 15).

Second, and perhaps more importantly, when using data cloning for [0, 1] bounded parameters, only the SE changes, not the estimates themselves. This will not generally be the case for parameters where the estimates themselves are functions of the sample size – clearly, abundance is such a parameter. If you look at the final spreadsheet for the robust design example (section F.3), you will see that the estimate of abundance using the cloned data is significantly larger than the original estimate (which of course, is not surprising, since the minimum bound of the estimate of abundance is M_{t+1}, which is multiplied by 100 during the cloning).

So, clearly, cloning should not be applied to parameters where the estimate will change as a function of cloning, and probably should not be applied without considerable care to parameters that are not simple [0, 1] bounded. We will consider a data type involving such unbounded parameters in the next worked example (section F.5). Having said that, it is fair to point out that whether or not the cloning procedure will work for all [0, 1] bounded parameters is somewhat of an 'open question'.

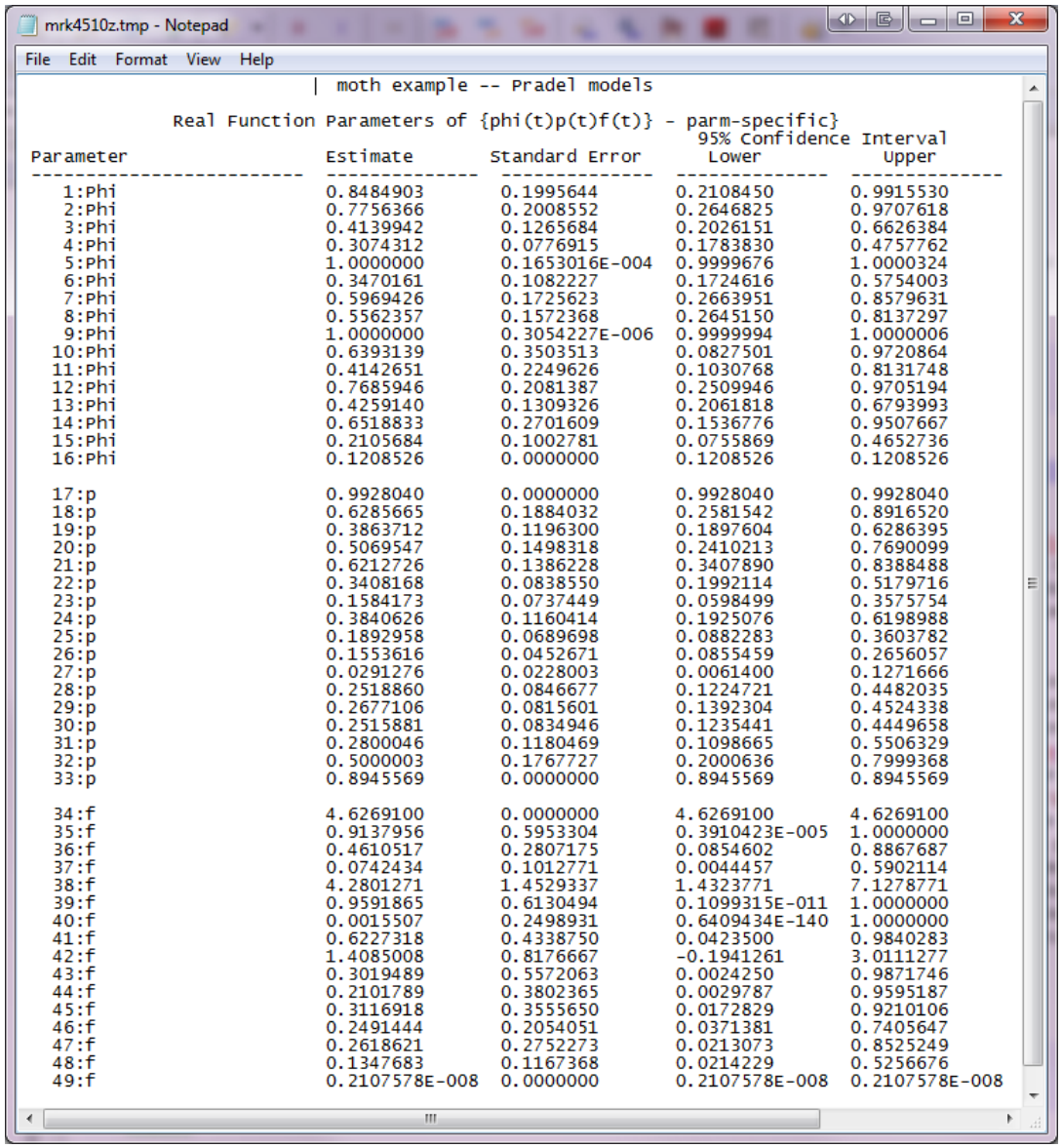
F.5. Worked example (4) – Pradel model example

Our final example re-visits the analysis of a famous set of data, the moth (*Gonodontis bidenta*) data reported on by Bishop *et al*. (1978), presented earlier in Appendix D (section D4.4). The data consist of records for 689 male moths that were captured, marked, and released daily over 17 days in northwest England. These moths were non-melanic; demographic parameters were estimated as part of a larger study looking at comparative fitness of distinct color morphs.

In Appendix D we considered the use of random effect Pradel models, focussing on estimation of process variance, and possible trend, in realized growth rate λ. Here, we consider using data cloning to evaluate possible intrinsic non-identifiability in time-dependent Pradel models. The encounter data for this example are contained in **`moth-example.inp`**. We will fit a fully time-dependent model using the '**`Pradel survival & Recruitment`**' data type. Recall (from Chapter 12) that there are 3 structural parameters specifying this model: φ, p and f (where f is the per capita recruitment probability). We will go ahead and fit model $\{\varphi_t\, p_t\, f_t\}$ to the moth data, using the default PIM structure and sine link. Given 17 sampling occasions, there are 49 structural parameters in the model (16 φ, 17 p, and 16 f parameters).

As an *a priori* check against possible numerical problems, we'll use the '**`Alt. Opt. Method`**' (i.e., simulated annealing). Since the recruitment parameter f is not necessarily bounded [0, 1] we'll use the log link function for the recruitment parameters. Once the numerical estimation has finished, we'll add the results to the browser. Although there are 49 structural parameters in the model, **MARK** reports that only 45 are identifiable, given the structure of the model, and the data.

Here are the real parameter estimates from model $\{\varphi_t\, p_t\, f_t\}$ fit to the moth encounter data:



mrk4510z.tmp - Notepad

File Edit Format View Help

moth example -- Pradel models

Real Function Parameters of {phi(t)p(t)f(t)} - parm-specific}

Parameter	Estimate	Standard Error	95% Confidence Interval Lower	Upper
1:Phi	0.8484903	0.1995644	0.2108450	0.9915530
2:Phi	0.7756366	0.2008552	0.2646825	0.9707618
3:Phi	0.4139942	0.1265684	0.2026151	0.6626384
4:Phi	0.3074312	0.0776915	0.1783830	0.4757762
5:Phi	1.0000000	0.1653016E-004	0.9999676	1.0000324
6:Phi	0.3470161	0.1082227	0.1724616	0.5754003
7:Phi	0.5969426	0.1725623	0.2663951	0.8579631
8:Phi	0.5562357	0.1572368	0.2645150	0.8137297
9:Phi	1.0000000	0.3054227E-006	0.9999994	1.0000006
10:Phi	0.6393139	0.3503513	0.0827501	0.9720864
11:Phi	0.4142651	0.2249626	0.1030768	0.8131748
12:Phi	0.7685946	0.2081387	0.2509946	0.9705194
13:Phi	0.4259140	0.1309326	0.2061818	0.6793993
14:Phi	0.6518833	0.2701609	0.1536776	0.9507667
15:Phi	0.2105684	0.1002781	0.0755869	0.4652736
16:Phi	0.1208526	0.0000000	0.1208526	0.1208526
17:p	0.9928040	0.0000000	0.9928040	0.9928040
18:p	0.6285665	0.1884032	0.2581542	0.8916520
19:p	0.3863712	0.1196300	0.1897604	0.6286395
20:p	0.5069547	0.1498318	0.2410213	0.7690099
21:p	0.6212726	0.1386228	0.3407890	0.8388488
22:p	0.3408168	0.0838550	0.1992114	0.5179716
23:p	0.1584173	0.0737449	0.0598499	0.3575754
24:p	0.3840626	0.1160414	0.1925076	0.6198988
25:p	0.1892958	0.0689698	0.0882283	0.3603782
26:p	0.1553616	0.0452671	0.0855459	0.2656057
27:p	0.0291276	0.0228003	0.0061400	0.1271666
28:p	0.2518860	0.0846677	0.1224721	0.4482035
29:p	0.2677106	0.0815601	0.1392304	0.4524338
30:p	0.2515881	0.0834946	0.1235441	0.4449658
31:p	0.2800046	0.1180469	0.1098665	0.5506329
32:p	0.5000003	0.1767727	0.2000636	0.7999368
33:p	0.8945569	0.0000000	0.8945569	0.8945569
34:f	4.6269100	0.0000000	4.6269100	4.6269100
35:f	0.9137956	0.5953304	0.3910423E-005	1.0000000
36:f	0.4610517	0.2807175	0.0854602	0.8867687
37:f	0.0742434	0.1012771	0.0044457	0.5902114
38:f	4.2801271	1.4529337	1.4323771	7.1278771
39:f	0.9591865	0.6130494	0.1099315E-011	1.0000000
40:f	0.0015507	0.2498931	0.6409434E-140	1.0000000
41:f	0.6227318	0.4338750	0.0423500	0.9840283
42:f	1.4085008	0.8176667	-0.1941261	3.0111277
43:f	0.3019489	0.5572063	0.0024250	0.9871746
44:f	0.2101789	0.3802365	0.0029787	0.9595187
45:f	0.3116918	0.3555650	0.0172829	0.9210106
46:f	0.2491444	0.2054051	0.0371381	0.7405647
47:f	0.2618621	0.2752273	0.0213073	0.8525249
48:f	0.1347683	0.1167368	0.0214229	0.5256676
49:f	0.2107578E-008	0.0000000	0.2107578E-008	0.2107578E-008

We show the full set of real parameter estimates because we want to make several points. First, we should have an intuitive sense by now that there are likely (inevitably?) going to be intrinsically non-identifiable parameters in fully time-dependent models. Typically, these show up at the 'end' of a time-series.

Because the Pradel model uses encounter histories going both 'backwards' and 'forwards' through time, we might suspect there would be intrinsic non-identifiability of parameters both at the start and end of the time-series. However, here we have 3 structural parameters (φ, p and f), which may interact in unexpected ways.

Finally, we probably don't anticipate that intrinsic non-identifiability will show up in the middle of a time-series for a parameter – non-identifiable parameters 'in the middle' of a time-series are more likely to reflect extrinsic non-identifiability (i.e., limitations of data, or boundary estimation problems).

Based on our estimates (above) we see good evidence that the final estimates for φ, p and f (parameters 16, 33 and 49) are all poorly estimated (any parameter with a reported SE of 0.000 is not well estimated). However, we also see some other parameters that weren't well estimated. In particular, the first p and f estimates (parameters 17 and 34) are also reported with SE of 0.000. Finally, several of the 'interior' estimates for φ and f (i.e., those in the middle of the time-series) are not well-estimated (parameters 5, 9, 35, 39 and 40, respectively).

We use the data cloning procedure to identify both the intrinsically and potentially extrinsically non-identifiable parameters. In the following spreadsheet, we have highlighted the intrinsically non-identifiable parameters (we've edited out some of the rows corresponding to some of the 'interior parameters' to make it fit the page).

	B	C	D	E	F	G	H	I	J	K
1	Label	Estimate	SE	LCB	UCB	Estimate)	SE X 100	LCB X 100	UCB X 100	SE Ratio
2	Phi	0.848486	0.199565	0.21085	0.991552	0.848486	0.019956	0.805106	0.883605	10.00021
3	Phi	0.775641	0.20085	0.264692	0.970762	0.775641	0.020085	0.733847	0.812549	10.00006
4	Phi	0.41399	0.126566	0.202615	0.662632	0.413993	0.012657	0.389423	0.438998	9.999946
5		<edited to save space>								
6	Phi	0.768589	0.208134	0.251007	0.970516	0.768584	0.020812	0.725317	0.806852	10.00056
7	Phi	0.425927	0.13094	0.206182	0.679423	0.425924	0.013093	0.400482	0.451765	10.00048
8	Phi	0.65188	0.270159	0.153679	0.950765	0.651878	0.027015	0.59724	0.702794	10.00017
9	Phi	0.21057	0.100281	0.075586	0.465283	0.210573	0.010028	0.19159	0.2309	9.999976
10	Phi	0.125317	0	0.125317	0.125317	0.125315	0.713065	4.16E-07	0.99998	0
11	p	0.469369	0	0.469369	0.469369	0.469369	4.603408	1.64E-16	1	0
12	p	0.628573	0.188413	0.258142	0.891663	0.628572	0.01884	0.590959	0.664689	10.00061
13	p	0.386364	0.119627	0.189758	0.628628	0.386363	0.011963	0.363196	0.410057	10.00008
14	p	0.50696	0.149838	0.241017	0.769022	0.50696	0.014984	0.477601	0.53627	10.00005
15		<edited to save space>								
16	p	0.267712	0.081559	0.139233	0.452433	0.267714	0.008156	0.252035	0.283999	10.00039
17	p	0.251582	0.083497	0.123537	0.444967	0.251583	0.008349	0.235574	0.268298	10.0004
18	p	0.280004	0.118045	0.109868	0.550627	0.280003	0.011804	0.25746	0.303712	10.00008
19	p	0.499996	0.176777	0.200056	0.79994	0.499995	0.017677	0.465403	0.534587	10.00023
20	p	0.862679	0	0.862679	0.862679	0.862677	4.90865	3.37E-35	1	0
21	f	1.740151	0	1.740151	1.740151	1.740151	25.38825	-48.0208	51.50113	0
22	f	0.913817	0.595371	3.9E-06	1	0.913831	0.059533	0.706719	0.979024	10.00063
23	f	0.461034	0.280715	0.085455	0.886761	0.461033	0.028071	0.406704	0.516303	10.00015
24		<edited to save space>								
25	f	0.074242	0.101285	0.004444	0.590274	0.074241	0.010128	0.056672	0.096699	10.00015
26	f	0.249158	0.205417	0.037139	0.740588	0.249161	0.020541	0.211103	0.291545	10.00056
27	f	0.261842	0.275226	0.021303	0.852528	0.261852	0.027522	0.211577	0.319237	10.00033
28	f	0.134771	0.116737	0.021424	0.525668	0.134767	0.011674	0.113481	0.159329	10.00014
29	f	0	0	0	0	2.09E-09	2.7E-07	-5.3E-07	5.31E-07	0

We see clearly that, as we suspected, several of the parameters are intrinsically non-identifiable. Specifically, the first and last p and f parameters, and the final φ parameter. We note that some of these parameters may become identifiable, provided the proper constraints are applied (e.g., if we fix the first and last p parameters to 1, then all of the f and φ parameters become estimable).

Next, we look for extrinsically non-identifiable parameters. We re-run our model, using profile likelihood estimation for the CI, followed by data cloning on this model run with the profile CI. Given the number of structural parameters in this model, this can take some time (and, in fact, the profile CI is not properly estimated for some parameters, due to numerical optimization issues). The results for a subset of the 'problem' parameters are shown at the top of the next page. To start, focus on parameters 5 and 9.

G23 fx 0.340813343989751

	A	B	C	D	E	F	G	H	I	J	K
1	Index	Label	Estimate	SE	LCB	UCB	Estimate)	SE X 100	LCB X 100	UCB X 100	SE Ratio
2	1	Phi	0.848486	0.199565	0.500489	1	0.848486	0.019956	0.80998	0.888287	10.00021
3	2	Phi	0.775641	0.20085	0.473309	1	0.775641	0.020085	0.737452	0.816272	10.00006
4	3	Phi	0.41399	0.126566	0.224775	0.767104	0.413993	0.012657	0.389923	0.439582	9.999946
5	4	Phi	0.307435	0.077691	0.180211	0.502477	0.307438	0.007769	0.292495	0.322958	9.999904
6	5	Phi	1	6.93E-05	0.524986	1	1	5.99E-06	0.971486	1	11.56387
7	6	Phi	0.347019	0.108281	0.185801	0.633276	0.347019	0.010796	0.326413	0.367871	10.02943
8	7	Phi	0.596933	0.172574	0.335786	1	0.59693	0.017251	0.564157	0.631756	10.00389
9	8	Phi	0.556247	0.157239	0.313581	1	0.556247	0.015724	0.526202	0.588105	10.00016
10	9	Phi	1	0.000185	0.489078	1	1	1.75E-05	0.94974	1	10.5753
11	10	Phi	0.639277	0.350268	0.258449	1	0.639283	0.035026	0.575419	0.71342	10.00019
12	11	Phi	0.414279	0.224937	0.169009	1	0.414274	0.022492	0.371566	0.459759	10.00055

For parameters 5 and 9, we see that in both instances, the lower CI bound has 'shortened' considerably for the cloned data, indicating that these parameters were actually being estimated. In contrast, the intrinsically non-identifiable parameters still show a relatively wide profile CI for the cloned analysis, with only a slight reduction from the original values. Here, for example, and the results for the final φ and first p parameters:

15	14	Phi	0.65188	0.270159	0.301829	1	0.651878	0.027015	0.601396	0.707536	10.00017
16	15	Phi	0.21057	0.100281	0.082806	0.616568	0.210573	0.010028	0.191885	0.231303	9.999976
17	16	Phi	0.125317	0	0.034872	1	0.125315	0.713065	0.098376	1	0
18	17	p	0.469369	0	0.073434	1	0.469369	4.603408	0.144224	1	0
19	18	p	0.628573	0.188413	0.305333	0.919877	0.628572	0.01884	0.591486	0.665222	10.00061
20	19	p	0.386364	0.119627	0.203168	0.64007	0.386363	0.011963	0.363264	0.410135	10.00008

However, recall that both φ and p are bounded $[0, 1]$. How well does our 'data cloning' approach apply for parameters that are not bounded $[0, 1]$? Say, the recruitment parameter, f, in a Pradel model, where the upper limit can potentially be > 1. Look back at the parameter estimates for $\hat{f}_i$ tabulated at the top of p. F.14. We see that recruitment parameters 35, 39 and 40 seem to be extrinsically non-identifiable. The reported CI for all three estimates is effectively $[0, 1]$, which seems suspicious. The first and last estimates (parameters 34 and 49) are also intrinsically confounded (as we showed earlier).

If we re-run our model using the profile likelihood CI routine, we find that there are several 'numerical problems' in estimating the profile CI for some parameters, in particular, some of the recruitment parameters. These problems were not 'solved' by using a different link function (say, the log link for the f parameters). So, we proceed cautiously, and apply the data cloning approach to the profile CI analysis. We see (below) that the profile CI reported for the cloned samples are narrower, suggesting that these parameters are being estimated correctly.

Index	Label	Estimate	SE	LCB	UCB	Estimate)	SE X 100	LCB X 100	UCB X 100	SE Ratio
34	f	4.627742	49.837	2E-07	13.00067	4.627628	10.42264	1.51E-05	5.077974	4.781611
35	f	0.913797	0.595365	0.048999	2.699406	0.913804	0.059535	0.800775	1.034633	10.00022
36	f	0.461027	0.280716	0.083079	1.340597	0.461033	0.028071	0.408324	0.51857	10.00023
37	f	0.074245	0.101287	2E-07	0.358755	0.074244	0.010128	0.054948	0.094689	10.00026
38	f	4.280204	1.452989	1.697582	8.817474	4.280152	0.145296	4.003777	4.575701	10.00022
39	f	0.959156	0.63183	0.230015	2.474151	0.959163	0.06113	0.845306	1.039754	10.33579
40	f	0.001561	0.262515	1E-07	0.81718	0.001561	0.024876	0.000446	0.001561	10.55294
41	f	0.622721	0.434489	0.071249	2.22489	0.622733	0.043378	0.541724	0.711583	10.01629

F.6. Limitations & other thoughts and approaches

As mentioned in our discussion of the 'robust design' example (section F.3), data cloning (as currently implemented in **MARK**) should not be applied to parameters where the estimate will change as a function of cloning (e.g., abundance), and probably should not be applied (or applied with some care) to parameters that are not simple $[0, 1]$ bounded (such as the recruitment parameter f in the Pradel model in the preceding example). Having said that, whether or not the cloning procedure will work for all $[0, 1]$ bounded parameters is 'a work in progress'. Stay tuned.

One limitation in the current implementation of data cloning is that the results are only reported in the Excel spreadsheet for the real parameters, and not the β parameters. However, you should be able to work backwards from the real parameters to determine which of the beta parameters are causing the confounding.

What about other 'numerical' approaches to the problem of confounding, and estimability? The two most commonly suggested are 'random effects' (Appendix D), and 'MCMC' (Appendix E). The basic idea motivating consideration of a 'random effects' approach is that since a RE model 'shrinks' the estimate toward the model-specified mean, then if a particular 'boundary estimate' is not shrunk much away from the boundary that this estimate is in fact 'correctly' estimated near the boundary.

Alas, this is not correct, for both conceptual and practical reasons (see Appendix D). Specifically, the amount of shrinkage is a function of the relative magnitude of process and sampling variance for a particular parameter, which has nothing specifically to do with whether or not a parameter is estimated near a boundary.

There has also been some suggestion that MCMC (see Appendix E) is 'robust' to estimation along the boundary. Again, this is not correct, especially for monotonic link functions (like the logit). The MCMC sampler in **MARK** samples on the transformed (beta) scale, and there is no particular reason to assume it will do better at estimating the parameter (although there is an argument that it will perform well at generating a credible CI for the estimate – whether or not such an interval is 'better' than a profile likelihood CI is an open question).

begin sidebar

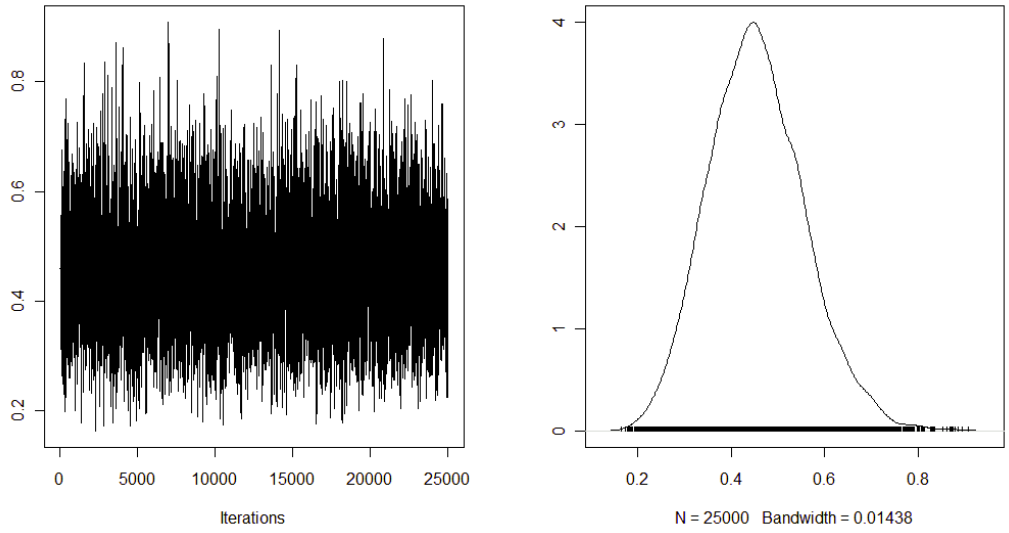
using MCMC to diagnose non-identifiable parameters

While MCMC will not 'solve' the problems of non-identifiable parameters, it is possible in some cases to use MCMC as a tool to diagnose (identify) parameters which are potentially non-identifiable (Gimenez *et al*. 2009). Recall that in a Bayesian analysis, the posterior is a function of the likelihood and the prior. Now, consider intrinsic non-identifiability. In this case, the likelihood surface for the parameter is fairly 'flat' (non-informative', and the posterior will strongly reflect the prior. In the cases of extrinsic non-identifiability, there is so little 'information' in the data for a given parameter, that the posterior again will strongly reflect the prior. In other words, if the 'shape' of the posterior is not appreciably different than the prior, then there is reason to expect that the parameter is not identifiable.

To demonstrate the basic idea, let's consider an MCMC analysis of the European Dipper data set (males and females). As discussed in section F.1, for a fully time-dependent model $\{\varphi_{s*t}\, p_{s*t}\}$, the final survival and encounter probability parameters φ_{k-1} and p_k are confounded for both sexes, respectively. These intrinsically non-identifiable parameters were clearly identified by the data cloning procedure.

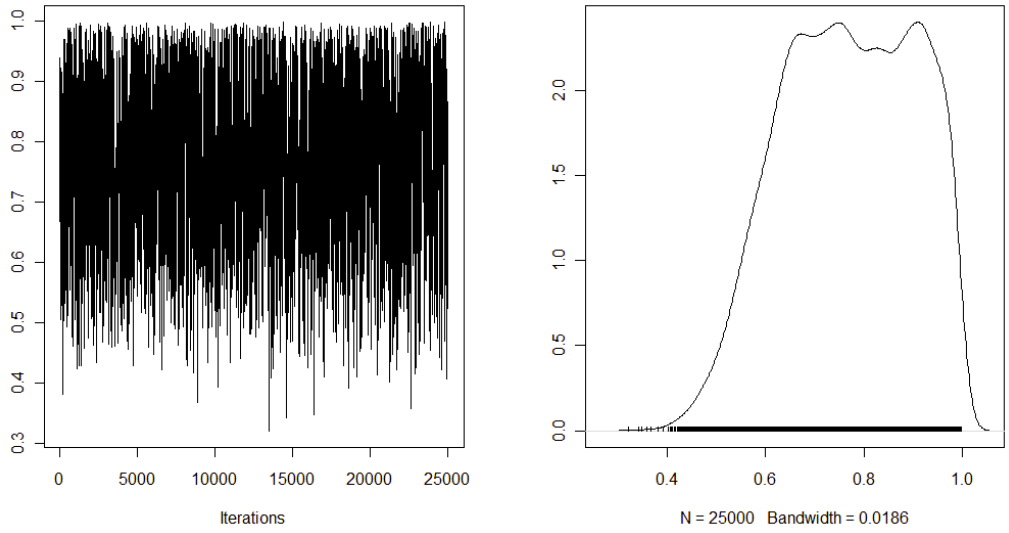
Can we achieve the same result using an MCMC approach? Skipping the details of the mechanics of implementing MCMC in **MARK** (see Appendix E), let's consider the trace and density plots for a couple of parameters from this model. Recall that the default prior used by **MARK** for MCMC estimation is a uniform (flat) prior.

First, consider the MCMC output for real parameter 2 (corresponding to $\varphi_{2,m}$).



We see that the trace is dense, regular, and symmetrical around the mode. The corresponding density plot is clearly unimodal, peaked, and very different in shape from the uniform ('flat') prior. This parameter is clearly identifiable, and well-estimated.

Contrast this with the trace and density plot for real parameter 6 (corresponding to $\varphi_{6,m}$), which we know to be intrinsically confounded with real parameter 12 ($p_{7,m}$).

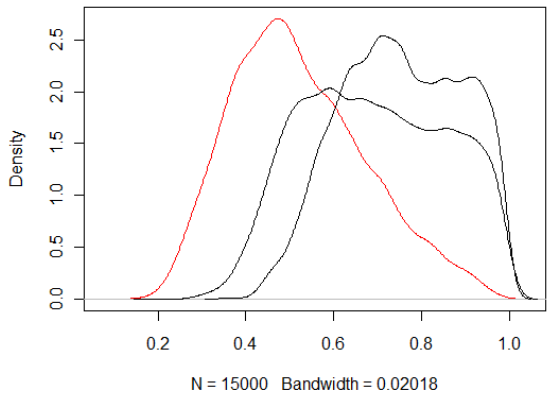


Here, we see that not only is the trace plot rather 'strange looking' (by various subjective criteria), but (more informatively), the density plot appears 'flat' over most of the range, clearly reflecting the strong influence of the default uniform (flat) prior. This is 'diagnostic' of an intrinsically non-identifiable parameter.* Examination of the other intrinsically non-identifiable parameters for this model (real parameters 12, 18 and 24) show the same 'flat' density plots.

* Gimenez *et al.* (2009) present a more 'quantitative' approach to comparing the posterior density and the prior.

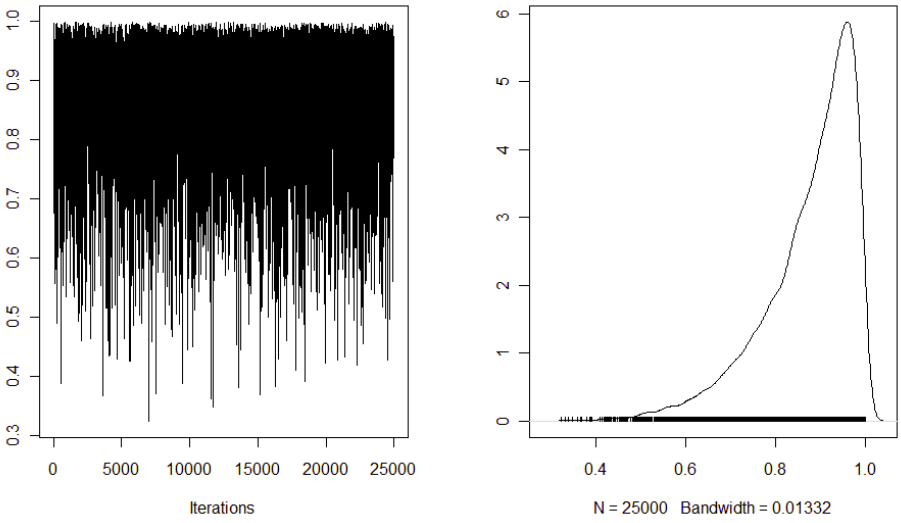
The underlying reason that the posterior plots for parameters that are flat for some distance across the top is that there is only a limited range for each of the parameters over which the product is defined to give exactly the same value equal to the MLE of the product (recall that **MARK** estimates a function of the product of φ and p for the confounded parameters).

Further, if you plot the posterior for the product (by multiplying the 2 values for each sample from the posterior), we would get a 'pointed' posterior (the mode of which is what **MARK** reports). For example, compare the density plot (below) for the product of the posterior sample for parameters 6 ($\varphi_{6,m}$) and 12 ($p_{7,m}$), indicated by the red line, with the density plots for each parameter separately, shown by the black lines.



It is also worth noting that the width of the 'flat top' is a function of what the product value is. So, as the value of the product goes toward one, the width of the posterior for each of the pieces becomes progressively narrower, and hence your ability to detect non-identifiability declines.

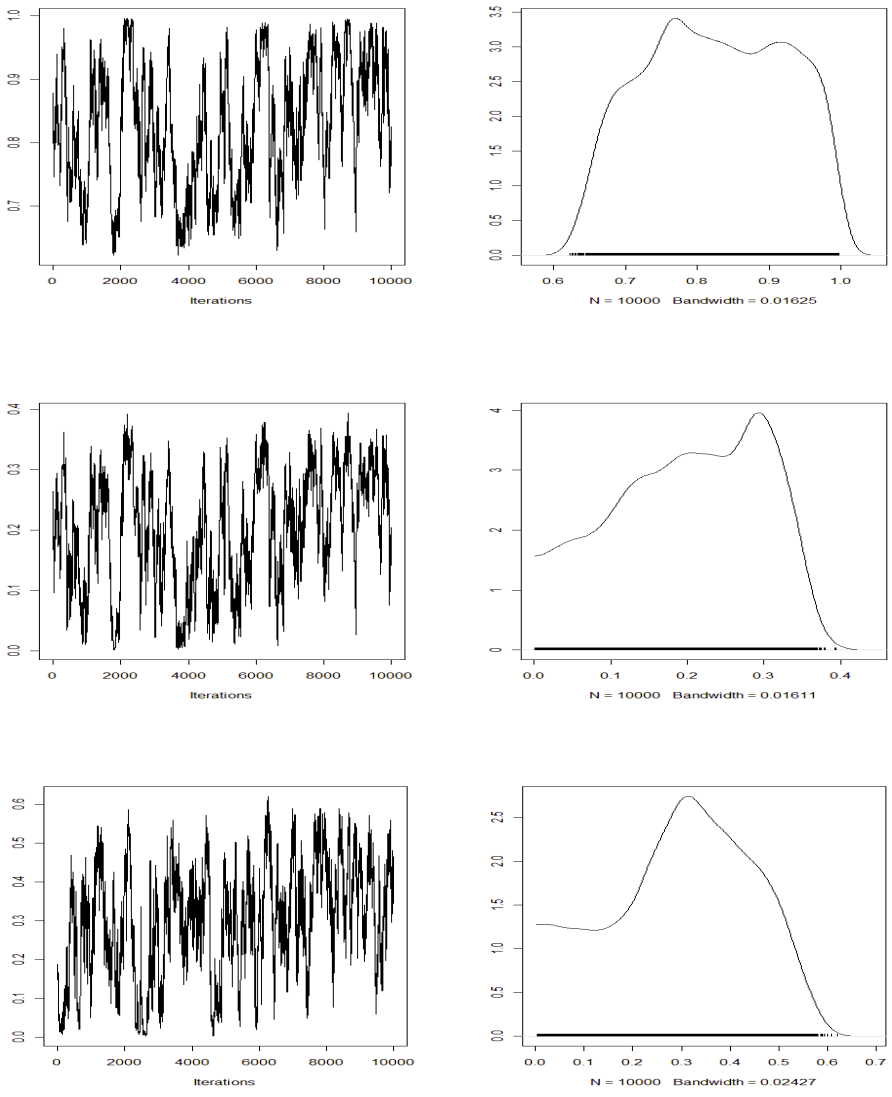
What about an *extrinsically* non-identifiable parameter? Recall from section F.1 that real parameter 14 (second recapture probability for male Dippers) was estimated at or near the boundary. The data cloning approach led us to conclude that this parameter was in fact correctly estimated as being at the boundary. What does the MCMC approach show?



We see (above) that the density plot is strongly left-skewed, as the information in the data 'piles

up' against the 1.0 boundary. Since extrinsic non-identifiability is frequently reflected in a parameter being estimated at either the [0, 1] boundary, it may be unlikely that the MCMC approach will be of much use in diagnosing extrinsic non-identifiability.

Further, even for intrinsically non-identifiable parameters, the density plot can be difficult to interpret, and may in fact bear no particular resemblance to the prior distribution (i.e., does not look particular similar to the default 'flat' uniform prior). For example, consider the robust design analysis presented in section F.3. In the absence of certain structural constraints, we expect that the final survival parameter S and the final two γ parameters will be intrinsically confounded (for the example data set, this involved real parameters 4, 8 and 11). This intrinsic confounding was demonstrated clearly using the data cloning approach. But, have a look at the trace and density plots for any of these 3 parameters (below, for parameters 4, 8 and 11, respectively):



While these plots are clearly 'strange looking', that is a somewhat 'subjective' assessment, and not based on a clear resemblance of the posterior to the prior. However, even such a 'subjective assessment may be enough to warrant further consideration of the identifiability of those parameters.

end sidebar

F.7. Summary

In this appendix, we've briefly introduced a convenient, fairly straightforward method for numerically assessing whether a parameter is 'estimable' or not. This method, based on 'data cloning', appears to work well, especially for 'standard' parameters that are $[0, 1]$ bounded (e.g., φ and p in a CJS model) – both for parameters that may be confounded (and thus not separately identifiable) because of the structure of the model, and for parameters that are poorly estimated because of inadequate data, or because they are near the $[0, 1]$ boundary.

F.8. References

Gimenez, O., Viallefont, A., Choquet, R., Catchpole, E. A., and Morgan, B. J. T. (2004) Methods for investigating parameter redundancy. *Animal Biodiversity and Conservation*, **27**, 561-572.

Gimenez, O., Morgan, B. J. T., and Brooks, S. P. (2009) Weak identifiability in models for mark-recapture-recovery data. In *Modeling Demographic Processes in Marked Populations*, D. L. Thomson, E. G. Cooch and M. J. Conroy, eds. Springer, New York, New York, USA, pp. 1055-1067.

Hunter, C. M., and Caswell, H. (2009) Rank and redundancy of multistate mark-recapture models for seabird populations with unobservable states. In *Modeling Demographic Processes in Marked Populations*, D. L. Thomson, E. G. Cooch and M. J. Conroy, eds. Springer, New York, New York, USA, pp. 324-333.

Lele, S. R., Dennis, B., and Lutscher, F. (2007) Data cloning: easy maximum likelihood estimation for complex ecological models using Bayesian Markov chain Monte Carlo methods. *Ecology Letters*, **10**, 551-563.

Lele, S. R., Nadeem, K., and Schmuland, B. (2010) Estimability and likelihood inference for generalized linear mixed models using data cloning. *Journal of the American Statistical Association*, **105**, 1617-1625.