APPENDIX B

The 'Delta method' . . .

Suppose you have conducted a mark-recapture study over 4 years which yields 3 estimates of apparent annual survival (say, $\hat{\varphi}_1$, $\hat{\varphi}_2$, and $\hat{\varphi}_3$). But, suppose what you are really interested in is the estimate of the product of the three survival values (i.e., the probability of surviving from the beginning of the study to the end of the study)? While it is easy enough to derive an estimate of this product (as $[\hat{\varphi}_1 \times \hat{\varphi}_2 \times \hat{\varphi}_3]$), how do you derive an estimate of the *variance* of the product? In other words, how do you derive an estimate of the variance of a transformation of one or more random variables, where in this case, we transform the three random variables (in this case, $\hat{\varphi}_1$, $\hat{\varphi}_2$ and $\hat{\varphi}_3$) by considering their product?

One commonly used approach which is easily implemented, not computer-intensive*, and can be robustly applied in many (but not all) situations is the so-called *Delta method* (also known as the method of propagation of errors). In this appendix, we briefly introduce some of the underlying background theory, and the application of the Delta method.

B.1. Background – mean and variance of random variables

Our interest here is developing a method that will allow us to estimate the variance for functions of random variables. Let's start by considering the formal approach for deriving these values explicitly, based on the *method of moments*.† For continuous random variables, consider a continuous function $f(x)$ on the interval $[-\infty, +\infty]$. The first four moments of $f(x)$ can be written as:

$$M_0 = \int_{-\infty}^{+\infty} f(x)dx,$$

$$M_1 = \int_{-\infty}^{+\infty} x f(x)dx,$$

$$M_2 = \int_{-\infty}^{+\infty} x^2 f(x)dx,$$

$$M_3 = \int_{-\infty}^{+\infty} x^3 f(x)dx.$$

* We briefly discuss some compute-intensive approaches in an Addendum to this appendix.

† In simple terms, a *moment* is a specific quantitative measure, used in both mechanics and statistics, of the shape of a set of points. If the set of points represents a probability density, then the moments relate to measures of shape and location such as mean, variance, skewness, and so forth.

In the particular case that the function is a probability density (as for a continuous random variable), then $M_0 = 1$ (i.e., the area under the pdf must equal 1).

For example, consider the uniform distribution on the finite interval $[a, b]$. A uniform distribution (sometimes also known as a rectangular distribution), is a distribution that has constant probability over the interval. The probability density function (pdf) for a continuous uniform distribution on the finite interval $[a, b]$ is:

$$P(x) = \begin{cases} 0 \text{ for } x < a \\ 1/(b-a) \text{ for } a < x < b \\ 0 \text{ for } x > b. \end{cases}$$

Integrating the pdf for $p(x) = 1/(b-a)$:

$$\begin{aligned} M_0 &= \int_a^b p(x)dx \\ &= \int_a^b \frac{1}{b-a}dx = 1, \\ M_1 &= \int_a^b xp(x)dx \\ &= \int_a^b \frac{x}{b-a}dx = \frac{a+b}{2}, \\ M_2 &= \int_a^b x^2p(x)dx \\ &= \int_a^b x^2\frac{1}{b-a}dx = \frac{1}{3}\left(a^2+ab+b^2\right), \\ M_3 &= \int_a^b x^3p(x)dx \\ &= \int_a^b x^3\frac{1}{b-a}dx = \frac{1}{4}\left(a^3+a^2b+ab^2+b^3\right). \end{aligned}$$

If you look closely, you should see that M_1 is the mean of the distribution. What about the variance? How do we interpret/use the other moments?

Recall that the variance is defined as the average value of the fundamental quantity [`distance from mean`]^2. The squaring of the distance is so the values to either side of the mean don't cancel out. The standard deviation is simply the square-root of the variance.

Given some discrete random variable x_i, with probability p_i, and mean μ, we define the variance as:

$$\text{var} = \sum \left(x_i - \mu\right)^2 p_i.$$

Note we don't have to divide by the number of values of x because the sum of the discrete probability distribution is 1 (i.e., $\sum p_i = 1$).

For a continuous probability distribution, with mean μ, we define the variance as:

$$\text{var} = \int_a^b (x-\mu)^2 p(x)dx.$$

Given our moment equations, we can then write:

$$\begin{aligned}
\text{var} &= \int_a^b (x-\mu)^2 p(x)dx \\
&= \int_a^b \left(x^2 - 2\mu x + \mu^2\right) p(x)dx \\
&= \int_a^b x^2 p(x)dx - \int_a^b 2\mu x p(x)dx + \int_a^b \mu^2 p(x)dx \\
&= \int_a^b x^2 p(x)dx - 2\mu \int_a^b x p(x)dx + \mu^2 \int_a^b p(x)dx.
\end{aligned}$$

Now, if we look closely at the last line, we see that in fact the terms represent the different moments of the distribution. Thus we can write:

$$\begin{aligned}
\text{var} &= \int_a^b (x-\mu)^2 p(x)dx \\
&= \int_a^b x^2 p(x)dx - 2\mu \int_a^b x p(x)dx + \mu^2 \int_a^b p(x)dx \\
&= M_2 - 2\mu\left(M_1\right) + \mu^2\left(M_0\right).
\end{aligned}$$

Since $M_1 = \mu$, and $M_0 = 1$ then:

$$\begin{aligned}
\text{var} &= M_2 - 2\mu\left(M_1\right) + \mu^2\left(M_0\right) \\
&= M_2 - 2\mu(\mu) + \mu^2(1) \\
&= M_2 - 2\mu^2 + \mu^2 \\
&= M_2 - \mu^2 \\
&= M_2 - \left(M_1\right)^2.
\end{aligned}$$

In other words, the variance for the pdf is simply the second moment (M_2) minus the square of the first moment ((M_1)^2).

Thus, for a continuous uniform random variable x on the interval $[a, b]$:

$$\begin{aligned}
\text{var} &= M_2 - \left(M_1\right)^2 \\
&= \frac{(a-b)^2}{12}.
\end{aligned}$$

It turns out, most of the usual measures by which we describe random distributions (mean, variance,...) are functions of the moments.

B.2. Transformations of random variables and the Delta method

OK – that's fine. If the pdf is specified, we can use the method of moments to formally derive the mean and variance of the distribution. But, what about functions of random variables having poorly specified or unspecified distributions? Or, situations where the pdf is not easily defined?

In such cases, we may need other approaches. We'll introduce one such approach here (the Delta method), by considering the case of a simple linear transformation of a random normal distribution.

Let

$$X_1, X_2, \ldots \sim N(10, \sigma^2 = 2).$$

In other words, random deviates drawn from a normal distribution with a mean of 10, and a variance of 2. Consider some transformations of these random values. You might recall from some earlier statistics or probability class that linearly transformed normal random variables are themselves normally distributed. Consider for example, $X_i \sim N(10, 2)$ – which we then linearly transform to Y_i, such that $Y_i = 4X_i + 3$.

Now, recall that for real scalar constants a and b we can show that

i.) $E(a) = a$, $E(aX + b) = aE(X) + b$

ii.) $\text{var}(a) = 0$, $\text{var}(aX + b) = a^2\text{var}(X)$.

Thus, given $X_i \sim N(10, 2)$ and the linear transformation $Y_i = 4X_i + 3$, we can write:

$$Y \sim N\left(\left[4(10) + 3 = 43\right], \left[(4^2)(2)\right]\right) = N(43, 32).$$

Now, an important point to note is that some transformations of the normal distribution are close to normal (i.e., are linear) and some are not. Since linear transformations of random normal values are normal, it seems reasonable to conclude that approximately linear transformations (over some range) of random normal data should also be approximately normal.

OK, to continue. Let $X \sim N(\mu, \sigma^2)$, and let $Y = g(X)$, where g is some transformation of X (in the previous example, $g(X) = 4X + 3$). It is hopefully relatively intuitive that the closer $g(X)$ is to linear over the likely range of X (i.e., within 3 or so standard deviations of μ), the closer $Y = g(X)$ will be to normally distributed. From calculus, we recall that if you look at any differentiable function over a narrow enough region, the function appears approximately linear. The approximating line is the tangent line to the curve, and its slope is the derivative of the function.

Since most of the mass (i.e., most of the random values) of X is concentrated around μ, let's figure out the tangent line at μ, using two different methods. First, we know that the tangent line passes through $(\mu, g(\mu))$, and that its slope is $g'\mu$ (we use the 'prime' notation, g', to indicate the first derivative of the function g). Thus, the equation of the tangent line is $Y = g'X + b$ for some b. Replacing (X, Y) with the known point $(\mu, g(\mu))$, we find $g(\mu) = g'(\mu)\mu + b$ and so $b = g(\mu) - g'(\mu)\mu$. Thus, the equation of the tangent line is $Y = g'(\mu)X + g(\mu) - g'(\mu)\mu$.

Now for the big step – we can derive an approximation to the same tangent line by using a *Taylor series expansion* of $g(x)$ (to first order) around $X = \mu$:

$$\begin{aligned} Y &= g(X) \\ &\approx g(\mu) + g'(\mu)(X - u) + \epsilon \\ &= g'(\mu)X + g(\mu) - g'(\mu)\mu + \epsilon. \end{aligned}$$

OK, at this point you might be asking yourself 'so what?'. [You might also be asking yourself 'what

the heck is a Taylor series expansion?'. If so, see the -sidebar-, below.]

Well, suppose that $X \sim N(\mu, \sigma^2)$ and $Y = g(X)$, where $g'(\mu) \neq 0$. Then, whenever the tangent line (derived earlier) is approximately correct over the likely range of X (i.e., if the transformed function is approximately linear over the likely range of X), then the transformation $Y = g(X)$ will have an approximate normal distribution. That approximate normal distribution may be found using the usual rules for linear transformations of normals.

Thus, to first order:

$$\begin{aligned} E(Y) &\approx g'(\mu)\mu + g(\mu) - g'(\mu)\mu \\ &= g(\mu) \end{aligned}$$

$$\begin{aligned} \text{var}(Y) \approx \text{var}\big(g(X)\big) &= \Big(g(X) - g(\mu)\Big)^2 \\ &= \big(g'(\mu)(X-\mu)\big)^2 \\ &= \big(g'(\mu)\big)^2 (X-\mu)^2 \\ &= \big(g'(\mu)\big)^2 \text{var}(X). \end{aligned}$$

In other words, for the expectation (mean), the first-order approximation is simply the transformed mean calculated for the original distribution. For the first-order approximation to the variance, we take the derivative of the transformed function with respect to the parameter, square it, and multiply it by the estimated variance of the untransformed parameter.

These first-order approximations to the expectation and variance of a transformed parameter are usually referred to as the *Delta method*.*

begin sidebar

Taylor series expansions?

Briefly, the *Taylor series* is a power series expansion of an infinitely differentiable real (or complex) function defined on an open interval around some specified point. For example, a one-dimensional Taylor series is an expansion of a real function $f(x)$ about a point $x = a$ over the interval $(a - r, a + r)$, is given as:

$$f(x) \approx f(a) + \frac{f'(a)(x-a)}{1!} + \frac{f''(x)(x-a)^2}{2!} + \ldots,$$

where $f'(a)$ is the first derivative of f with respect to a, $f''(x)$ is the second derivative of f with respect to a, and so on.

For example, suppose the function is $f(x) = e^x$. The convenient fact about this function is that all its derivatives are equal to e^x as well (i.e., $f(x) = e^x$, $f'(x) = e^x$, $f'' = e^x, \ldots$). In particular, $f^{(n)}(x) = e^x$ so that $f^{(n)}(0) = 1$. This means that the coefficients of the Taylor series are given by:

$$a_n = \frac{f^{(n)}(0)}{n!} = \frac{1}{n!},$$

and so the Taylor series is given by:

$$1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \ldots + \frac{x^n}{n!} + \ldots = \sum_{n=0}^{\infty} \frac{x^n}{n!}.$$

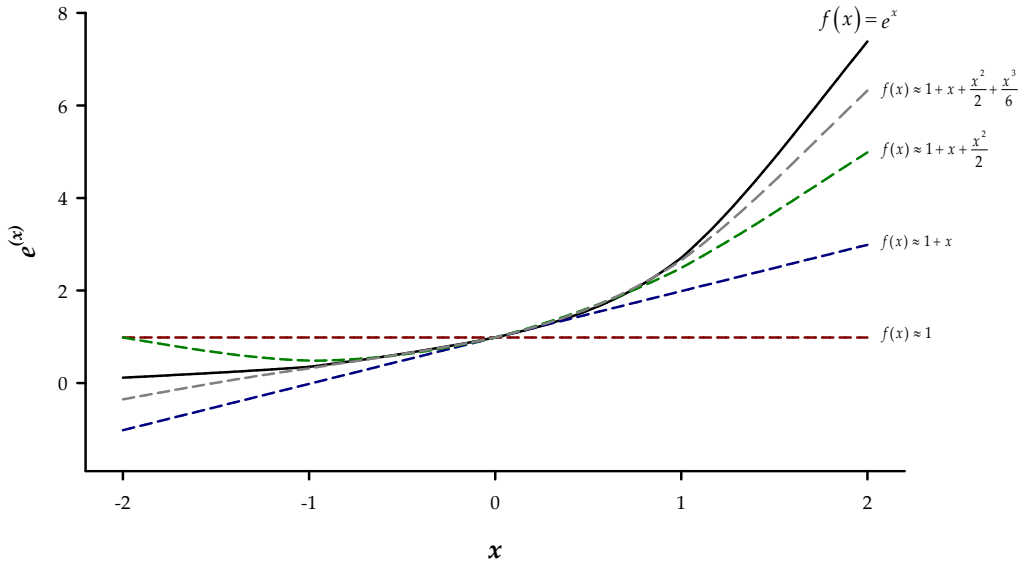
* For an interesting review of the history of the Delta method, see Ver Hoef (2012).

The primary utility of such a power series in simple application is that differentiation and integration of power series can be performed term by term and is hence particularly (or, at least relatively) easy. In addition, the (truncated) series can be used to compute function values approximately.

Now, let's look at an example of the "fit" of a Taylor series to a familiar function, given a certain number of terms in the series. For our example, we'll expand the function $f(x) = e^x$, at $x = a = 0$, on the interval $[a-2, a+2]$, for $n = 0, n = 1, n = 2, \ldots$ (where n is the number of terms in the series). For $n = 0$, the Taylor expansion is a scalar constant (1):

$$f(x) \approx 1,$$

which we anticipate to be a poor approximation to the function $f(x) = e^x$ at any point. The relationship between 'the number of terms', and the 'fit' of the Taylor series expansion to the function $f(x) = x^x$, is shown clearly in the following figure. The solid black line in the figure is the function $f(x) = e^x$, evaluated over the interval $[-2, 2]$. The dashed lines represent different orders (i.e., number of terms) in the expansion. The red dashed line represents the 0th order expansion, $f(x) \approx 1$, the blue dashed line represents the 1st order expansion, $f(x) \approx 1 + x$, and so on.



We see that when we add more terms (i.e., use a higher-order series), the fit gets progressively better. Often, for 'nice, smooth' functions (i.e., those nearly linear at the point of interest), we don't need many terms at all. For this example, the 3rd order expansion ($n = 4$) yields a relatively good approximation to the function over much of the interval $[-2, 2]$.

Another example – suppose the function of interest is $f(x) = (x)^{1/3}$ (i.e., $f(x) = \sqrt[3]{x}$). Suppose we're interested in $f(x) = (x)^{1/3}$ where $x = 27$ (i.e., $f(27) = \sqrt[3]{27}$). Now, it is straightforward to show that $f(27) = \sqrt[3]{27} = 3$. But suppose we want to know $f(25) = \sqrt[3]{25}$, using a Taylor series approximation?

We recall that to first order:

$$f(x) = f(a) + f'(a)(x - a),$$

where in this case, $a = 25$ and $x = 27$. The derivative of f with respect to x for this function $f(a) = (a)^{1/3}$ is:

$$f'(a) = \frac{a^{-2/3}}{3} = \frac{1}{3\sqrt[3]{x^2}}.$$

Thus, using the first-order Taylor series, we write:

$$\begin{aligned} f(25) &\approx f(27) + f'(27)(25-27) \\ &= 3 + (0.037037)(-2) \\ &= 2.926. \end{aligned}$$

Clearly, 2.926 is very close to the true value of $f(25) = \sqrt[3]{25} = 2.924$. In other words, the first-order Taylor approximation works well for this function. As we will see later, this is not always the case, which has important implications.

end sidebar

B.3. Transformations of one variable

OK, enough background for now. Let's see some applications. Let's check the Delta method out in a few cases where we (probably) know the answer.

Assume we have an estimate of density $\hat{D}$ and its conditional sampling variance, $\widehat{\text{var}}(\hat{D})$. We want to multiply this by some constant c to make it comparable with other values from the literature. Thus, we want $\hat{D}_s = g(D) = c\hat{D}$, and $\widehat{\text{var}}(D_s)$.

The Delta method gives:

$$\begin{aligned} \widehat{\text{var}}(\hat{D}_s) &\approx (g'(D))^2 \dot{\sigma}_D^2 \\ &= \left(\frac{\partial \hat{D}_s}{\partial \hat{D}}\right)^2 \cdot \widehat{\text{var}}(\hat{D}) \\ &= c^2 \cdot \widehat{\text{var}}(\hat{D}), \end{aligned}$$

which we know to be true for the variance of a random variable multiplied by a real constant.

For example, suppose we take a large random normal sample, with mean $\mu = 1.5$, and true sample variance of $\sigma^2 = 0.25$. We are interesting in approximating the variance of this sample multiplied by some constant, $c = 2.25$. From the preceding, we expect that the variance of the transformed sample is approximated as:

$$\begin{aligned} \widehat{\text{var}}(\hat{D}_s) &\approx c^2 \cdot \widehat{\text{var}}(\hat{D}) \\ &= 2.25^2(0.25) \\ &= 1.265625. \end{aligned}$$

The following snippet of **R** code simulates this situation – the variance of the (rather large) random normal sample (`sample`), multiplied by the constant $c = 2.25$ (where the transformed sample is `trans_sample`) is 1.265522, which is quite close to the approximation from the Delta method (1.265625).

```
> sample <- rnorm(1000000,1.5,0.5)
> c <- 2.25
> trans_sample <- c*sample
> var(trans_sample)
  [1] 1.265522
```

Another example of much the same thing – consider a known number of N harvested fish, with an average weight ($\hat{\mu}_w$) and variance. If you want an estimate of total biomass (B), then $\hat{B} = N \cdot \hat{\mu}_w$ and applying the Delta method, the variance of $\hat{B}$ is approximated as $N^2 \cdot \widehat{\text{var}}(\hat{\mu}_w)$.

So, if there are $N = 100$ fish in the catch, with an average mass $\hat{\mu}_w = 20$ pounds, with an estimated variance of $\hat{\sigma}_w^2 = 1.44$, then by the Delta method, the approximate variance of the total biomass $\hat{B} = (100 \times 20) = 2{,}000$ is:

$$\begin{aligned}\widehat{\text{var}}(\hat{B}) &\approx N^2 \cdot \widehat{\text{var}}(\hat{\mu}_w) \\ &= (100)^2(1.44) \\ &= 14{,}400.\end{aligned}$$

The following snippet of **R** code simulates this particular example – the estimated variance of the (rather large) numerically simulated sample (`biomass`), is very close to the approximation from the Delta method (14,400).

```
  N <- 100  # size of sample of fish

  mu <- 20;  # average mass of fish
  v <- 1.44;   # variance of same...

  reps <- 1000000  # number of replicate samples desired to generate dist of biomass

# set up replicate samples - recall that biomass is the product of N and
# the 'average' mass, which is sampled from a rnorm with mean mu and variance v
  biomass  <- replicate(reps,N*rnorm(1,mu,sqrt(v)));

# output result from the simulated biomass data
  print(var(biomass));
    [1] 14399.1
```

One final example – you have some parameter θ, which you transform by dividing it by some constant c. Thus, by the Delta method:

$$\widehat{\text{var}}\left(\frac{\hat{\theta}}{c}\right) \approx \left(\frac{1}{c}\right)^2 \cdot \widehat{\text{var}}(\hat{\theta}).$$

So, using a normal probability density function, for $\theta = \mu = 1.8$, and $\sigma_\theta^2 = 1.5$, where the constant $c = 3.14159$, then

$$\begin{aligned}\widehat{\text{var}}\left(\frac{\hat{\theta}}{c}\right) &\approx \left(\frac{1}{3.14159}\right)^2 \cdot (1.5) \\ &= 0.15198.\end{aligned}$$

The following snippet of **R** code simulates this situation – the variance of the (rather large) random normal sample (`sample`), divided by the constant, $c = 3.14159$ (where the transformed sample is `trans_sample`) is 0.1513828, which is again quite close to the approximation from the Delta method (0.15198).

```
> sample <- rnorm(1000000,mean=1.8,sd=sqrt(1.5))
> c <- 3.14159
> sample_trans <- sample/c

> var(sample_trans)
  [1] 0.1513828
```

B.3.1. A potential complication – violation of assumptions

A final – and conceptually important – example for transformations of single variables. The importance lies in the demonstration that the Delta method does not always work. Remember, the Delta method assumes that the transformation is approximately linear over the expected range of the parameter.

Suppose one has a MLE for the mean and estimated variance for some parameter θ which is bounded random uniform on the interval $[0, 2]$. Suppose you want to transform this parameter such that:

$$\psi = e^{\theta}.$$

[Recall that this is a convenient transformation since the derivative of e^x is e^x, making the calculations very simple. Also recall for the preceding -sidebar- that the Taylor series expansion to first-order may not 'do' particular well with this function.]

Now, based on the Delta method, the variance for ψ would be estimated as:

$$\begin{aligned}\widehat{\text{var}}(\hat{\psi}) &\approx \left(\frac{\partial\hat{\psi}}{\partial\hat{\theta}}\right)^2 \cdot \widehat{\text{var}}(\hat{\theta}) \\ &= \left(e^{\theta}\right)^2 \cdot \widehat{\text{var}}(\hat{\theta}).\end{aligned}$$

Now, suppose that $\hat{\theta} = 1.0$, and $\widehat{\text{var}}(\hat{\theta}) = 0.3\dot{3}$. Then, from the Delta method:

$$\begin{aligned}\widehat{\text{var}}(\hat{\psi}) &\approx \left(e^{\theta}\right)^2 \cdot \widehat{\text{var}}(\hat{\theta}) \\ &= (7.38906)(0.3\dot{3}) \\ &= 2.46302.\end{aligned}$$

Is this a reasonable approximation? The only way we can answer that question is if we know what the 'true' (correct) estimate of the variance should be.

There are two approaches we might use to come up with the 'true' (correct) variance: (1) analytically, or (2) by numerical simulation.

We'll start with the formal, analytical approach, and derive the variance of ψ using the method of moments introduced earlier. To do this, we need to integrate the pdf (uniform, in this case) over some range. Since the variance of a uniform distribution is $(b-a)^2/12$ (as derived earlier in this appendix), and if b and a are symmetric around the mean (1.0), then we can show by algebra that given a variance of $0.3\dot{3}$, then $a = 0$ and $b = 2$ (check: $(b-a)^2/12 = (2-0)^2/12 = 0.3\dot{3}$).

Given a uniform distribution, the pdf is $p(\theta) = 1/(b-a)$. Thus, by the method of moments:

$$M_1 = \int_a^b \frac{g(x)}{b-a} dx = -\frac{e^b - e^a}{a-b},$$

$$M_2 = \int_a^b \frac{g(x)^2}{b-a} dx = \left(\frac{1}{2}\right) \cdot \frac{e^{2a} - e^{2b}}{a-b}.$$

Thus, by moments, var$(E(\psi)$ is:

$$\text{var}\big(E(\psi)\big) = M_2 - \big(M_1\big)^2 = \left(\frac{1}{2}\right) \cdot \frac{-e^{2b} + e^{2a}}{-b+a} - \frac{\big(e^b - e^a\big)^2}{(a-b)^2}.$$

If $a = 0$ and $b = 2$, then the true variance is given as:

$$\begin{aligned} \text{var}\big(E(\psi)\big) &= M_2 - (M_1)^2 \\ &= \left(\frac{1}{2}\right) \cdot \frac{-e^{2b} + e^{2a}}{-b+a} - \frac{\big(e^b - e^a\big)^2}{(a-b)^2} \\ &= 3.19453, \end{aligned}$$

which is not particularly close to the value estimated by the Delta method (2.46302).

Let's also consider coming up with an estimate of the 'true' variance by numerical simulation. The steps are pretty easy: (i) simulate a large data set, (ii) transform the entire data set, and (iii) calculate the variance of the transformed data set.

For our present example, here is one way you might set this up in **R**:

```
> sim.data <- runif(10000000,0,2);
> transformed.data <- exp(sim.data);
> var(transformed.data);
    [1] 3.19509
```

which is pretty close to the value derived analytically, above (3.19453) – the slight difference reflects Monte Carlo error (generally, the bigger the simulated data set, the smaller the error).

Ok, so now that we have derived the 'true' variance in a couple of different ways, the important question is – why the discrepancy between the 'true' variance of the transformed distribution (3.19453), and the first-order approximation to the variance using the Delta method (2.46302)?

As discussed earlier, the Delta method rests on the assumption the first-order Taylor expansion around the parameter value is effectively linear over the range of values likely to be encountered. Since in this example we're using a uniform pdf, then all values between a and b are equally likely. Thus, we might anticipate that as the interval between a and b gets smaller, then the approximation to the variance (which will clearly decrease) will get better and better (since the smaller the interval, the more likely it is that the function is approximately linear over that range).

For example, if $a = 0.5$ and $b = 1.5$ (same mean of 1.0), then the true variance of θ will be $0.08\dot{3}$. Thus, by the Delta method, the estimated variance of ψ will be 0.61575, while by the method of moments (which is exact), the variance will be 0.65792. Clearly, the proportional difference between the two values has declined markedly. But, we achieved this 'improvement' by artificially reducing the true variance of the untransformed variable θ. Obviously, we can't do this in general practice.

So, what are the practical options? Well, one possible solution is to use a higher-order Taylor series approximation – by including higher-order terms, we can (and should) achieve a better 'fit' to the function (see the preceding -sidebar-). In other words, our approximation to the variance should be improved by using a higher-order Taylor expansion. The only 'technical' challenge (which really isn't too difficult, with some practice, potentially assisted by some good symbolic math software) is coming up with the higher-order terms.

One convenient approach to deriving those higher-order terms for the present problem is to express the transformation function ψ in the form $\text{var}[g(X)] = \text{var}[g(\mu + X - \mu)]$, which, after some fairly tedious bits of algebra, can be Taylor expanded as (written sequentially below, each row representing the next order of the expansion):*

$$\begin{aligned}
\text{var}[g(\mu + X - \mu)] \approx\ & g'(\mu)^2 \text{var}(X) \\
& + 2g'(\mu)\frac{g''(\mu)}{2} E\Big((X-\mu)^3\Big) \\
& + \left[\frac{g''(\mu)^2}{4} + 2g'(\mu)\frac{g'''(\mu)}{3!}\right] E\Big((X-\mu)^4\Big) \\
& + \left[2g'(\mu)\frac{g^{(4)}(\mu)}{4!} + 2\frac{g''(\mu)}{2}\frac{g'''(\mu)}{3!}\right] E\Big((X-\mu)^5\Big) \\
& + \ldots
\end{aligned}$$

Now, while this looks a little ugly (ok, maybe more than 'little' ugly), it actually isn't too bad – the whole expansion is written in terms of 'things we know': the derivatives of our transformation ($g', g'', \ldots$), simple scalars and scalar factorials, and, expectations of sequential powers of the deviations of the data from the mean of the distribution (i.e., $E\big((X-\mu)^n\big)$). You already know from elementary statistics that $E\big((X-\mu)^1\big) = 0$, and $E\big((X-\mu)^2\big) = \sigma^2$ (i.e, the variance). But what about $E\big((X-\mu)^3\big)$, or $E\big((X-\mu)^4\big)$. The higher-order terms in the Taylor expansion are often functions of the expectation of these higher-power deviations of the data from the mean. How do we calculate these expectations?

In fact, it isn't hard at all, and involves little more than applying an approach you've already seen – look back a few pages and have another look at how we derived the variance as a function of the first and second moments of the pdf. Remember, the variance is simply $E\big((X-\mu)^2\big) = \sigma^2$. Thus, we might anticipate that the the same logic used in deriving the estimate $E\big((X-\mu)^2\big)$ as a function of the moments could be used for $E\big((X-\mu)^3\big)$, or $E\big((X-\mu)^4\big)$, and so on.

The mechanics for $E\big((X-\mu)^3\big)$ are laid out in the following -sidebar-. You can safely skip this section if you want, and jump ahead to the the calculation of the variance using a higher-order expansion, but it might be worth at least skimming through this material, if for no other reason that to demonstrate that this is quite 'doable'. It is also a pretty nifty demonstration that a lot of interesting things can be and are developed as a function of the moments of the pdf.

begin sidebar

approximating $E\big((X-\mu)^3\big)$

Here, we demonstrate the mechanics for evaluating $E\big((X-\mu)^3\big)$. While it might look a bit daunting, in fact it is relatively straightforward, and is It is a convenient demonstration of how you can use the

* For simplicity, we're dropping (not showing) the terms involving $\text{Cov}[(x-\mu)^m, (X-\mu)^n]$ – thus, the expression as written isn't a complete expansion to order n, but it is close enough to demonstrate the point.

'algebra of moments' to derive some interesting and useful things.

$$\begin{aligned}
E\big((X-\mu)^3\big) &= \int_a^b (x-\mu)^3 p(x)dx \\
&= \int_a^b \Big(x^3 + 3\mu^2 x - 3\mu x^2 - \mu^3\Big)p(x)dx \\
&= \int_a^b x^3 p(x)dx + \int_a^b 3\mu^2 x p(x)dx - \int_a^b 3\mu x^2 p(x)dx - \int_a^b \mu^3 p(x)dx \\
&= \int_a^b x^3 p(x)dx + 3\mu^2 \int_a^b x p(x)dx - 3\mu \int_a^b x^2 p(x)dx - \mu^3 \int_a^b p(x)dx \\
&= M_3 + 3\mu^2(M_1) - 3\mu(M_2) - \mu^3(M_0).
\end{aligned}$$

Since $M_1 = \mu$, and $M_0 = 1$, then

$$\begin{aligned}
E\big((X-\mu)^3\big) &= \int_a^b (x-\mu)^3 p(x)dx \\
&= M_3 + 3\mu^2(M_1) - 3\mu(M_2) - \mu^3(M_0) \\
&= M_3 + 3M_1^3 - 3M_1M_2 - M_1^3.
\end{aligned}$$

At this point, all that remains is substituting in the expressions for the moments corresponding to the particular pdf (in this case, $U(a, b)$, as derived a few pages back), and you have your function for the expectation $E\big((X-\mu)^3\big)$.

We'll leave it to you to confirm the algebra – the 'answer' is

$$\begin{aligned}
E\big((X-\mu)^3\big) &= 2\left(\frac{1}{2}a + \frac{1}{2}b\right)^3 - 3\left(\frac{1}{2}a + \frac{1}{2}b\right)\left(\frac{1}{3}a^2 + \frac{1}{3}ab + \frac{1}{3}b^2\right) + \frac{1}{4}a^3 + \frac{1}{4}a^b + \frac{1}{4}ab^2 + \frac{1}{4}b^3 \\
&= 0.
\end{aligned}$$

Yes, a lot of work and 'some algebra', for what seems like an entirely anti-climatic result:

$$\text{"}E\big((X-\mu)^3\big) \text{ for the pdf } U(a, b) \text{ is } 0.\text{"}$$

But you're happier knowing how it's done (no, really). We use the same procedure for $E\big((X-\mu)^4\big)$, and so on.

In fact, if you go through the exercise of calculating $E\big((X-\mu)^n\big)$ for $n = 4, 5, \ldots$, you'll find that they generally alternate between 0 (e.g., $E\big((X-\mu)^3\big) = 0$ for $U(a, b)$), and non-zero (e.g., $E\big((X-\mu)^4\big) = 0.2$, for $U(0, 2)$). This can be quite helpful in simplifying the Taylor expansion.

end sidebar

How well does a higher-order approximation do? Let's start by having another look at the Taylor expansion we presented a few pages back – we'll focus on the expansion out to order 5:

$$\begin{aligned}
\text{var}[g(\mu + X - \mu)] \approx\ & g'(\mu)^2 \text{var}(X) \\
& + 2g'(\mu)\frac{g''(\mu)}{2} E\Big(\big(X-\mu\big)^3\Big)
\end{aligned}$$

$$+\left[\frac{g''(\mu)^2}{4}+2g'(\mu)\frac{g'''(\mu)}{3!}\right]E\Big((X-\mu)^4\Big)$$
$$+\left[2g'(\mu)\frac{g^{(4)}(\mu)}{4!}+2\frac{g''(\mu)}{2}\frac{g'''(\mu)}{3!}\right]E\Big((X-\mu)^5\Big).$$

If you had a look at the preceding -sidebar-, you'd have seen that some of the expectation terms (and products of same) equal 0, and thus can be dropped from the expansion. So, our order 5 Taylor series expansion can be written as:

$$\begin{aligned}\text{var}[g(\mu+X-\mu)] \approx\ & g'(\mu)^2\text{var}(X)\\ & +2g'(\mu)\frac{g''(\mu)}{2}E\Big((X-\mu)^3\Big)\\ & +\left[\frac{g''(\mu)^2}{4}+2g'(\mu)\frac{g'''(\mu)}{3!}\right]E\Big((X-\mu)^4\Big)\\ & +\left[2g'(\mu)\frac{g^{(4)}(\mu)}{4!}+2\frac{g''(\mu)}{2}\frac{g'''(\mu)}{3!}\right]E\Big((X-\mu)^5\Big)\\ =\ & g'(\mu)^2\text{var}(X)+\left[\frac{g''(\mu)^2}{4}+2g'(\mu)\frac{g'''(\mu)}{3!}\right]E\Big((X-\mu)^4\Big).\end{aligned}$$

So, how much better does this higher-order approximation do? If we 'run through the math', and for $U(a,b)$ where $a=0, b=2$, such that $\mu=1, \sigma^2=0.3\dot{3}, E\big((X-\mu)^4\big)=0.2$, we end up with

$$\begin{aligned}\text{var}[g(\mu+X-\mu)] &\approx g'(\mu)^2\text{var}(X)+\left[\frac{g''(\mu)^2}{4}+2g'(\mu)\frac{g'''(\mu)}{3!}\right]E\Big((X-\mu)^4\Big)\\ &=\big(e^1\big)^2(0.3\dot{3})+\left[\frac{\big(e^1\big)^2}{4}+2\big(e^1\big)\frac{e^1}{3!}\right](0.20)\\ &=3.325075,\end{aligned}$$

which is much closer to the true value of 3.19453 (the fact that the estimated value is slightly larger than the true value is somewhat odd, and possibly reflects not including the $\text{Cov}[(x-\mu)^m,(X-\mu)^n]$ terms in the Taylor expansion). Regardless, it is a much better approximation than the first-order value of 2.46302.

OK, the preceding is arguably a somewhat artificial example. Now we'll consider a more realistic situation where the first-order approximation may be insufficient to our needs.

Delta method applied to the expectation of the transformed data

Consider the following situation. Suppose you are interested in simulating some data on the logit scale, where variation around the mean if normal (so, you're going to simulate logit-normal data). Suppose the mean of some parameter on the real probability scale is $\theta = 0.3$. Transformed to the

logit scale, the mean of the sample you're going to simulate would be $\log(\theta/(1-\theta)) = -0.8472979$. So, you want to simulate some normal data, with some specified variance, on the logit scale, centered on $\mu_{\text{logit}} = -0.8472979$.

Here, using **R**, we generate a vector (which we've called `samp`, below) of 50,000 logit-normal deviates, with a $\mu_{\text{logit}} = -0.8472979$, and a standard deviation of of $\sigma_{\text{logit}} = 0.75$ (corresponding to a variance of $\sigma^2_{\text{logit}} = 0.5625$). We'll set the random number seed at `1234` so you can try this yourself, if inclined:

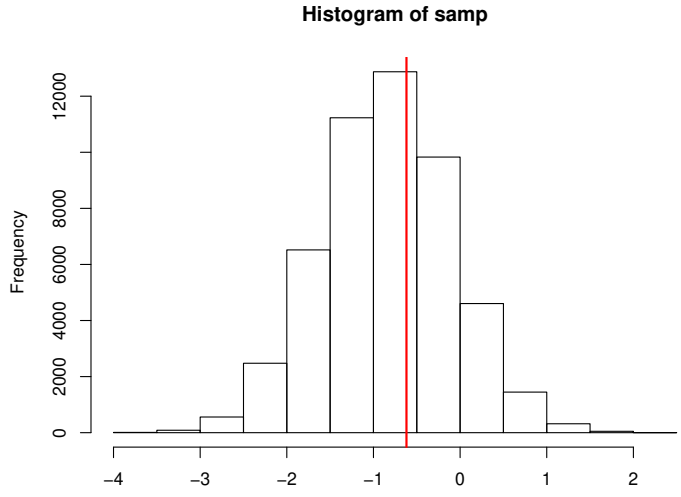
```
> set.seed(1234);
> samp <- rnorm(50000,-0.8472979,0.75);
```

If we check the mean and variance of our random sample, we find they're quite close to the true parameters used in simulating the data (perhaps not surprising given the size of the simulated sample).

```
> mean(samp)
     [1] -0.8456896

> var(samp)
     [1] 0.5630073
```

If we plot a histogram of the simulated data, we see a symmetrical distribution centered around the true mean $\mu_{\text{logit}} = -0.8472979$ (vertical red line):



What is the variance of the back-transformed estimate of the mean, on the real probability scale? We know from what we've covered so far that if we try to calculate the variance of the back-transform of these data from the logit scale → real probability scale, by simply taking the back transform of the estimated variance $\hat{\sigma}^2_{\text{logit}} = 0.5630073$, we'll get the incorrect answer. If we do that, we would get

$$\frac{e^{0.5630073}}{1 + e^{0.5630073}} = 0.6371481.$$

How can we confirm our developing intuition that this value is incorrect? Well, if we simply back-transform the entire random sample, and then calculate the variance of this back transformed sample (which we call back) directly,

```
> expit=function(x) exp(x)/(1+exp(x));
> back <- expit(samp)
> var(back)
    [1]  0.02212163
```

we get a value which, as we might have expected, isn't remotely close to the value of 0.6371481 we obtained by simply back-transforming the variance estimate.

Of course, we know by now we should have used the Delta method here. First, we recall that the back-transform f from the logit $\rightarrow$ to the real scale is:

$$f = \frac{e^{\theta}}{1 + e^{\theta}}.$$

Then, we apply the Delta method as:

$$\begin{aligned}
\widehat{\text{var}}(f) &\approx \left(\frac{\partial f}{\partial \hat{\theta}}\right)^2 \times \widehat{\text{var}}(\hat{\theta}) \\
&= \left(\frac{e^{\hat{\theta}}}{\left(1 + e^{\hat{\theta}}\right)^2}\right)^2 \times \widehat{\text{var}}(\hat{\theta}) \\
&= \left(\frac{e^{-0.8456896}}{\left(1 + e^{-0.8456896}\right)^2}\right)^2 \times 0.5630073 \\
&= 0.024860,
\end{aligned}$$

which is very close to the value we derived (above) by calculating the variance of the entire back-transformed sample (0.022121).

However, the main point we we want to cover here is applying the Delta method to other moments. Specifically, the mean. Recall that the mean from our logit-normal sample was −0.8456896. Can we simply back-transform this mean from the logit $\rightarrow$ real probability scale? In other words,

$$\frac{e^{-0.8456896}}{1 + e^{-0.8456896}} = 0.3003378.$$

Now, compare this value to the mean of the entire back-transformed sample:

```
> mean(back)
   [1]  0.3199361
```

You might think that the two values (0.3003378, 0.3199361) are 'close enough for government work' (although the difference is roughly 6%), but since we don't work for the government, let's apply the Delta method to generate a correct approximation to the back-transformed mean.

First, recall that the transformation function f (from logit $\rightarrow$ real) is

$$f = \frac{e^{\theta}}{1 + e^{\theta}}.$$

Next, remember that the Delta method as we've been applying generally (and in the preceding for the variance) it is based on the first-order Taylor series approximation.

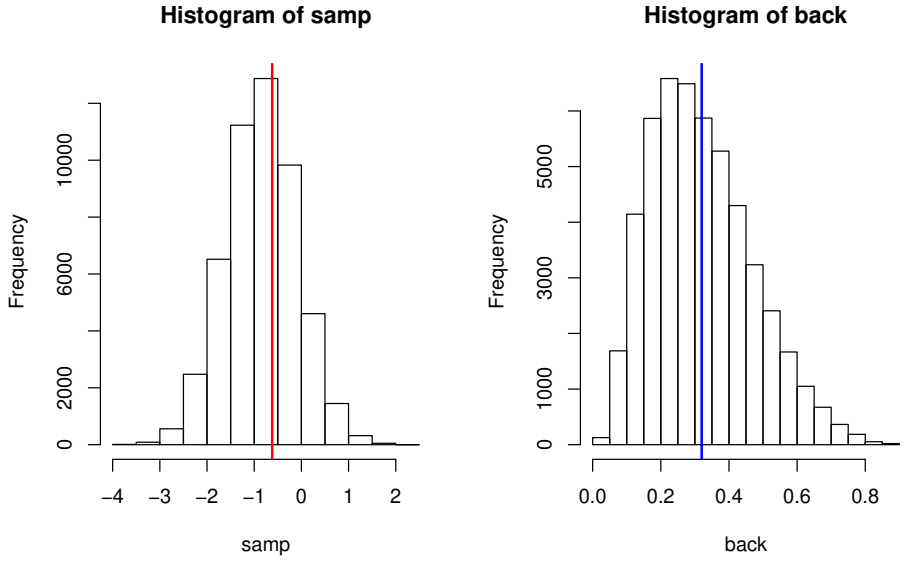
What is the first-order Taylor series expansion for f, if $\theta = \mu$? In fact, it is simply:

$$\frac{e^{\mu}}{1 + e^{\mu}} + O\left((\theta - \mu)^2\right),$$

where there term $O\left((\theta - \mu)^2\right)$ is the asymptotic bound of the growth of the error. But, more to the point, the first-order approximation is basically our back-transformation, with some (possibly a lot) of error added.

In fact, we might expect this error term to be increasingly important if the assumptions under which the first-order approximation applies are strongly violated. In particular, if the transformation function is highly non-linear over the range of values being examined.

Do we have such a situation in the present example? Compare the histograms of our simulated data, on the logit (`samp`) and back-transformed real scales (`back`), respectively:



Note that the mean of the back-transformed distribution (vertical blue line) is somewhat to the right of the mass of the distribution, which is fairly asymmetrical.

This suggests that the back-transformation might be sufficiently non-linear that we need to use a higher-order Taylor series approximation.

If you do the math (which isn't that difficult), the second-order approximation is given as

$$\frac{e^{\mu}}{\left(1+e^{\mu}\right)}+\frac{e^{\mu}}{\left(1+e^{\mu}\right)^{2}}(\theta-\mu)+O\left((\theta-\mu)^{2}\right).$$

Now, while the preceding might look a bit 'complicated', the key here is remembering that we're dealing with '*expectations*'. What is the expectation of $(\theta-\mu)$? In this situation, θ is a random variable – where each estimated mean from a set of replicated data sets on the logit scale represents θ, and μ is the overall parametric mean. We know from even the most basic statistics class that the expectation of the difference of a random variable X_i from the mean of the set of random variables, $\bar{X}$, is 0 (i.e., $E(X_i-\bar{X})=0$).

By the same logic, then, the expectation of $E(\theta-\mu)=0$. And, anything multiplied by 0 is 0, so, after dropping the error term, our second-order approximation reduces to

$$\frac{e^{\mu}}{\left(1+e^{\mu}\right)}+\frac{e^{\mu}}{\left(1+e^{\mu}\right)^{2}}(\theta-\mu)=\frac{e^{\mu}}{\left(1+e^{\mu}\right)},$$

which brings us right back to our standard first-order approximation.

What about a third-order approximation? After a bit more math, we end up with

$$\frac{e^{\mu}}{\left(1+e^{\mu}\right)}+\frac{e^{\mu}}{\left(1+e^{\mu}\right)^{2}}(\theta-\mu)-\frac{1}{2}\frac{e^{\mu}\left(e^{\mu}-1\right)}{\left(1+e^{\mu}\right)^{3}}(\theta-\mu)^{2}+O\left((\theta-\mu)^{3}\right).$$

Again, the expectation for $E(\theta-\mu)=0$. So, that terms drops out:

$$\frac{e^{\mu}}{\left(1+e^{\mu}\right)}+\frac{e^{\mu}}{\left(1+e^{\mu}\right)^{2}}(\theta-\mu)-\frac{1}{2}\frac{e^{\mu}\left(e^{\mu}-1\right)}{\left(1+e^{\mu}\right)^{3}}(\theta-\mu)^{2}$$

What about for the term $E(\theta-\mu)^2$? Look closely – '*variate minus mean, squared*'. Look familiar? It should – it's the variance! So, $E(\theta-\mu)^2=\hat{\sigma}^2$.

Thus, after dropping the error term, our third-order approximation to the mean is given as

$$\frac{e^{\mu}}{\left(1+e^{\mu}\right)}-\frac{1}{2}\frac{e^{\mu}\left(e^{\mu}-1\right)}{\left(1+e^{\mu}\right)^{3}}\sigma^{2}.$$

So, given our estimate of $\hat{\mu}=-0.8456896$ and $\hat{\sigma}^2=0.5630073$ on the logit-scale, our third-order Delta method approximation for the expectation (mean) on the back-transformed real probability scale, using this third-order approximation is

$$\frac{e^{-0.8456896}}{\left(1+e^{-0.8456896}\right)}-\frac{1}{2}\frac{e^{-0.8456896}\left(e^{-0.8456896}-1\right)}{\left(1+e^{-0.8456896}\right)^{3}}(0.5630073)=0.3239593842,$$

which is quite a bit closer to the empirical estimate of the mean derived from the entire back-transformed sample (0.3199361) than was our first attempt using the first-order approximation (0.3003378).

So, we see that the classical Delta method, which is based on a first-order Taylor series expansion of the transformed function, may not do particularly well if the function is highly non-linear over the range of values being examined. Of course, it would be fair to note that the preceding example made the assumption that the distribution was random uniform over the interval.

For most of our work with **MARK**, the interval is likely to have a near-symmetric mass around the estimate, typically β. As such, most of data, and thus the transformed data, will actually fall closer to the parameter value in question (the mean in this example) than we've demonstrated here. So much so, that the discrepancy between the first order 'Delta' approximation to the variance and the true value of the variance will likely be significantly smaller than shown here, even for a strongly non-linear transformation. We leave it to you as an exercise to prove this for yourself.

But, this point notwithstanding, it is important to be aware of the assumptions underlying the Delta method. If your transformation is non-linear, and there is considerable variation in your data, the first-order approximation may not be particularly good.

B.4. Transformations of two or more variables

We are often interested in transformations involving more than one variable. Fortunately, there are also multivariate generalizations of the Delta method.

Suppose you've estimated p different random variables $X_1, X_2, \ldots, X_p$. In matrix notation, these variables would constitute a $(p \times 1)$ random vector:

$$\mathbf{X} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix},$$

which has a mean vector:

$$\boldsymbol{\mu} = \begin{bmatrix} EX_1 \\ EX_2 \\ \vdots \\ EX_p \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{bmatrix},$$

and the $(p \times p)$ variance-covariance matrix is:

$$\mathrm{cov}(X_1, X_2) = \begin{bmatrix} \mathrm{var}(X_1) & \mathrm{cov}(X_1, X_2) & \ldots & \mathrm{cov}(X_1, X_p) \\ \mathrm{cov}(X_2, X_1) & \mathrm{var}(X_2) & \ldots & \mathrm{cov}(X_2, X_p) \\ \vdots & \vdots & \vdots & \vdots \\ \mathrm{cov}(X_p, X_1) & \mathrm{cov}(X_p, X_2) & \ldots & \mathrm{var}(X_p) \end{bmatrix}.$$

Note that if the variables are independent, then the off-diagonal elements (i.e., the covariance terms) are all zero.

Thus, for a $(k \times p)$ matrix of constants $\mathbf{A} = a_{ij}$, the expectation of a random vector $\mathbf{Y} = \mathbf{AX}$ is given as:

$$\begin{bmatrix} EY_1 \\ EY_2 \\ \vdots \\ EY_p \end{bmatrix} = \mathbf{A}\boldsymbol{\mu},$$

with a variance-covariance matrix:

$$\text{cov}(\mathbf{Y}) = \mathbf{A}\boldsymbol{\Sigma}\mathbf{A}^{\mathsf{T}}.$$

Now, using the same logic we first considered for developing the Delta method for a single variable, for each x_i near μ_i, we can write:

$$\begin{aligned} y &= \begin{bmatrix} g_1(x) \\ g_2(x) \\ \vdots \\ g_p(x) \end{bmatrix} \\ &\approx \begin{bmatrix} g_1(\mu) \\ g_2(\mu) \\ \vdots \\ g_p(\mu) \end{bmatrix} + \mathbf{D}(x - \mu), \end{aligned}$$

where $\mathbf{D}$ is the matrix of partial derivatives of g_i with respect to x_j, evaluated at $(x - \mu)$.

As with the single-variable Delta method, if the variances of the X_i are small (so that with high probability Y is near μ, such that the linear approximation is usually valid), then to first-order we can write:

$$\begin{bmatrix} EY_1 \\ EY_2 \\ \vdots \\ EY_p \end{bmatrix} = \begin{bmatrix} g_1(\mu) \\ g_2(\mu) \\ \vdots \\ g_p(\mu) \end{bmatrix}$$

$$\widehat{\text{var}}(\hat{Y}) \approx \mathbf{D}\boldsymbol{\Sigma}\mathbf{D}^{\mathsf{T}}.$$

In other words, to approximate the variance of some multi-variable function $\mathbf{Y}$, we (i) take the vector of partial derivatives of the function with respect to each parameter in turn (generally known as the *Jacobian*), $\mathbf{D}$, (ii) right-multiply this vector by the variance-covariance matrix, $\boldsymbol{\Sigma}$, and (iii) right-multiply the resulting product by the transpose of the original vector of partial derivatives, $\mathbf{D}^{\mathsf{T}}$.

Note: interpretation of the variance estimated using the Delta method is dependent on the source of the variance-covariance matrix, Σ, used in the calculations. If Σ is constructed using standard ML estimates of the variances and covariances, then the resulting Delta method estimate for variance is an estimate of the '*total*' variance, which is the sum of '*sampling*' + '*biological process*' variance. In contrast, if Σ is based on estimated '*process*' variances and covariances only, then the Delta method estimate for

variance is an estimate of the '*process*' variance. Decomposition of total variance into sampling and process components is covered in detail in Appendix D.

begin sidebar

alternative algebras for Delta method

There are alternative formulations of this expression which may be more convenient to implement in some instances. When the variables $\theta_1, \theta_2 \ldots \theta_k$ (in the function, Y) are independent, then

$$\begin{aligned}\widehat{\text{var}}(\hat{Y}) &\approx \text{var}\big(f(\theta_1, \theta_2, \ldots \theta_k)\big) \\ &= \sum_{i=1}^{k} \text{var}(\theta_i)\left(\frac{\partial f}{\partial \theta_i}\right)^2,\end{aligned}$$

where $\partial f / \partial \theta_1$ is the partial derivative of Y with respect to θ_i. When the variables $\theta_1, \theta_2 \ldots \theta_k$ (in the function, Y) are **not** independent, then the covariance structure among the variables must be accounted for:

$$\begin{aligned}\widehat{\text{var}}(\hat{Y}) &\approx \text{var}\big(f(\theta_1, \theta_2, \ldots \theta_k)\big) \\ &= \sum_{i=1}^{k} \text{var}(\theta_i)\left(\frac{\partial f}{\partial \theta_i}\right)^2 + 2\sum_{i<j}^{k}\sum \text{cov}(\theta_i, \theta_j)\left(\frac{\partial f}{\partial \theta_i}\right)\left(\frac{\partial f}{\partial \theta_j}\right)\end{aligned}$$

end sidebar

Example (1) – variance of a product of survival probabilities

Let's consider the application of the Delta method in estimating sampling variances of a fairly common function – the product of several parameter estimates.

From the preceding, we see that:

$$\begin{aligned}\widehat{\text{var}}(\hat{Y}) &\approx \mathbf{D}\boldsymbol{\Sigma}\mathbf{D}^{\mathsf{T}} \\ &= \left[\frac{\partial(\hat{Y})}{\partial(\hat{\theta})}\right] \cdot \widehat{\boldsymbol{\Sigma}} \cdot \left[\frac{\partial(\hat{Y})}{\partial(\hat{\theta})}\right]^{\mathsf{T}},\end{aligned}$$

where Y is some linear or nonlinear function of the k parameter estimates $\hat{\theta}_1, \hat{\theta}_2, \ldots \hat{\theta}_k$.

The first term, $\mathbf{D}$, on the RHS of the variance expression is a row vector containing partial derivatives of Y with respect to each of these k parameters $(\hat{\theta}_1, \hat{\theta}_2, \ldots \hat{\theta}_k)$. The right-most term of the RHS of the variance expression, $\mathbf{D}^{\mathsf{T}}$, is simply a transpose of this row vector (i.e., a column vector). The middle-term, $\boldsymbol{\Sigma}$ is simply the estimated variance-covariance matrix for the parameters.

To demonstrate the steps in the calculation, we'll use estimates from model $\{\varphi_t\, p_.\}$ fit to the male European dipper data set. Suppose we're interested in the probability of surviving from the start of the first interval to the end of the third interval. The estimate of this probability is easy enough:

$$\begin{aligned}\hat{Y} &= \big(\hat{\varphi}_1 \times \hat{\varphi}_2 \times \hat{\varphi}_3\big) \\ &= \big(0.6109350 \times 0.458263 \times 0.4960239\big) \\ &= 0.138871.\end{aligned}$$

So, the estimated probability of a male Dipper surviving over the first three intervals is ~ 14% (again, assuming that our time-dependent survival model is a valid model).

To derive the estimate of the variance of the product, we will also need the variance-covariance matrix for the survival estimates. You can generate the matrix easily in **MARK** by selecting '**`Output | Specific Model Output | Variance-Covariance Matrices | Real Estimates`**'.

The variance-covariance matrix for the male Dipper data, generated from model $\{\varphi_t\, p_.\}$, as output to the default editor (e.g., Windows Notepad), is shown below:

male dippers
Real Parameter Estimates Variances and Covariances
{phi(t)p(.)}

Variance-Covariance matrix of estimates on diagonal and below,
Correlation matrix of estimates above diagonal.

	1	2	3	4	5	6	7
1	0.02243	-0.02638	0.00513	0.00735	0.00516	0.02379	-0.09253
2	-0.00039	0.00997	-0.02779	0.00545	0.00383	0.01765	-0.06865
3	0.00007	-0.00024	0.00724	-0.03332	0.00309	0.01427	-0.05549
4	0.00009	0.00004	-0.00023	0.00661	-0.04175	0.02042	-0.07941
5	0.00006	0.00003	0.00002	-0.00026	0.00581	-0.02857	-0.05572
6	0.00028	0.00014	0.00010	0.00013	-0.00017	0.00634	-0.25711
7	-0.00053	-0.00026	-0.00018	-0.00025	-0.00016	-0.00078	0.00146

Here, the variance-covariance values are *below* the diagonal, whereas the standardized correlation values are *above* the diagonal. The variances are given *along* the diagonal.

However, it is **very important** to note that the V-C matrix that **MARK** outputs to the editor is *rounded* to 5 significant digits. For the actual calculations, we need to use the full precision values.* To get those, you need to either (i) output the V-C matrix into a dBase file (which you could then open with dBase, or Excel), or (ii) copy the V-C matrix into the Windows clipboard, and then paste it into some other application. Failure to use the full precision V-C matrix will almost always lead to significant 'rounding errors'.

The 'full precision' V-C matrix for the 3 Dipper survival estimates is shown below.

$$\widehat{\mathrm{cov}}(\hat{Y}) = \sum\widehat{}$$

$$= \begin{bmatrix} \widehat{\mathrm{var}}(\hat{\varphi}_1) & \widehat{\mathrm{cov}}(\hat{\varphi}_1, \hat{\varphi}_2) & \widehat{\mathrm{cov}}(\hat{\varphi}_1, \hat{\varphi}_3) \\ \widehat{\mathrm{cov}}(\hat{\varphi}_2, \hat{\varphi}_1) & \widehat{\mathrm{var}}(\hat{\varphi}_2) & \widehat{\mathrm{cov}}(\hat{\varphi}_2, \hat{\varphi}_3) \\ \widehat{\mathrm{cov}}(\hat{\varphi}_3, \hat{\varphi}_1) & \widehat{\mathrm{cov}}(\hat{\varphi}_3, \hat{\varphi}_2) & \widehat{\mathrm{var}}(\hat{\varphi}_3) \end{bmatrix}$$

* The variance-covariance estimates **MARK** generates will occasionally depend somewhat on which optimization method you use (i.e., default, or simulated annealing), and on the starting values used to initialize the optimization. The differences in the reported values are often very small (i.e., apparent only several decimal places out from zero), but you should be aware of them. For all of the examples presented in this Appendix, we have used the default optimization routines, and default starting values.

$$= \begin{bmatrix} 0.0224330125 & -0.0003945405 & 0.0000654469 \\ -0.0003945405 & 0.0099722201 & -0.0002361998 \\ 0.0000654469 & -0.0002361998 & 0.0072418858 \end{bmatrix}.$$

For this example, the transformation we're applying to our 3 survival estimates (which we'll call Y) is the product of the estimates (i.e., $\hat{Y} = \hat{\varphi}_1\hat{\varphi}_2\hat{\varphi}_3$).

Thus, our variance estimate is given as

$$\widehat{\text{var}}(\hat{Y}) \approx \left[\left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_1}\right) \quad \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_2}\right) \quad \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_3}\right)\right] \cdot \widehat{\sum} \cdot \begin{bmatrix} \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_1}\right) \\ \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_2}\right) \\ \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_3}\right) \end{bmatrix}.$$

Each of the partial derivatives for $\hat{Y}$ is easy enough to derive for this example. Since $\hat{Y} = \hat{\varphi}_1\hat{\varphi}_2\hat{\varphi}_3$, then $\partial\hat{Y}/\partial\hat{\varphi}_1 = \hat{\varphi}_2\hat{\varphi}_3$. And so on.

So,

$$\widehat{\text{var}}(\hat{Y}) \approx \left[\left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_1}\right) \quad \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_2}\right) \quad \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_3}\right)\right] \cdot \widehat{\sum} \cdot \begin{bmatrix} \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_1}\right) \\ \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_2}\right) \\ \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_3}\right) \end{bmatrix}$$

$$= \left[(\hat{\varphi}_2\hat{\varphi}_3) \quad (\hat{\varphi}_1\hat{\varphi}_3) \quad (\hat{\varphi}_1\hat{\varphi}_2)\right] \cdot \begin{bmatrix} \widehat{\text{var}}(\hat{\varphi}_1) & \widehat{\text{cov}}(\hat{\varphi}_1, \hat{\varphi}_2) & \widehat{\text{cov}}(\hat{\varphi}_1, \hat{\varphi}_3) \\ \widehat{\text{cov}}(\hat{\varphi}_1, \hat{\varphi}_1) & \widehat{\text{var}}(\hat{\varphi}_2) & \widehat{\text{cov}}(\hat{\varphi}_2, \hat{\varphi}_3) \\ \widehat{\text{cov}}(\hat{\varphi}_3, \hat{\varphi}_1) & \widehat{\text{cov}}(\hat{\varphi}_3, \hat{\varphi}_2) & \widehat{\text{var}}(\hat{\varphi}_3) \end{bmatrix} \cdot \begin{bmatrix} (\hat{\varphi}_2\hat{\varphi}_3) \\ (\hat{\varphi}_1\hat{\varphi}_3) \\ (\hat{\varphi}_1\hat{\varphi}_2) \end{bmatrix}.$$

Clearly, the estimator is getting more and more 'impressive' as we progress.

The resulting expression (written in piecewise fashion to make it easier to see the basic pattern) is shown below:

$$\begin{aligned} \widehat{\text{var}}(\hat{Y}) \approx\ & \hat{\varphi}_2^2\hat{\varphi}_3^2[\widehat{\text{var}}(\hat{\varphi}_1)] \\ & + 2\hat{\varphi}_2\hat{\varphi}_3^2\hat{\varphi}_1[\widehat{\text{cov}}(\hat{\varphi}_1, \hat{\varphi}_2)] \\ & + 2\hat{\varphi}_2^2\hat{\varphi}_3\hat{\varphi}_1[\widehat{\text{cov}}(\hat{\varphi}_1, \hat{\varphi}_3)] \\ & + \hat{\varphi}_1^2\hat{\varphi}_3^2[\widehat{\text{var}}(\hat{\varphi}_2)] \\ & + 2\hat{\varphi}_1^2\hat{\varphi}_3\hat{\varphi}_2[\widehat{\text{cov}}(\hat{\varphi}_2, \hat{\varphi}_3)] \\ & + \hat{\varphi}_1^2\hat{\varphi}_2^2[\widehat{\text{var}}(\hat{\varphi}_3)]. \end{aligned}$$

After substituting in our estimates for φ_i and the variances and covariances, our estimate for the variance of the product $\hat{Y} = (\hat{\varphi}_1\hat{\varphi}_2\hat{\varphi}_3)$ is (approximately) $\widehat{\text{var}}(Y) = 0.0025565$.

Example (2) – variance of estimate of reporting rate

In some cases animals are tagged or banded to estimate a "*reporting rate*" – the proportion of tagged animals reported (say, to a conservation management agency), given that they were killed and retrieved by a hunter or angler (see chapter 8 for more details). Thus, N_c animals are tagged with normal (*control*) tags and, of these, R_c are recovered the first year following release. The *recovery rate* of these control animals is merely R_c/N_c and we denote this as f_c.

Another group of animals, of sample size N_r, are tagged with special *reward* tags; these tags indicate that some amount of money (say, \$50) will be given to people reporting these special tags. It is assumed that \$50 is sufficient to ensure that all such tags will be reported, thus these serve as a basis for comparison and the estimation of a reporting rate. The recovery probability for the reward tagged animals is merely R_r/N_r, where R_r is the number of recoveries of reward-tagged animals the first year following release. We denote this recovery probability as f_r.

The estimator of the *reporting rate* is a ratio of the *recovery rates* and we denote this as λ. Thus:

$$\hat{\lambda} = \frac{\hat{f}_c}{\hat{f}_r}.$$

Now, note that both recovery probabilities are binomials.

Thus:

$$\widehat{\text{var}}(\hat{f}_c) = \frac{\hat{f}_c\left(1-\hat{f}_c\right)}{N_c} \quad \text{and} \quad \widehat{\text{var}}(\hat{f}_r) = \frac{\hat{f}_r\left(1-\hat{f}_r\right)}{N_r}.$$

In this case, the samples are independent, thus $\text{cov}(f_c, f_r)$ and the sampling variance-covariance matrix is diagonal:

$$\begin{bmatrix} \widehat{\text{var}}(\hat{f}_c) & 0 \\ 0 & \widehat{\text{var}}(\hat{f}_r) \end{bmatrix}.$$

Next, we need the derivatives of λ with respect to f_c and f_r:

$$\frac{\partial\hat{\lambda}}{\partial\hat{f}_c} = \frac{1}{\hat{f}_r}, \quad \text{and} \quad \frac{\partial\hat{\lambda}}{\partial\hat{f}_r} = -\frac{\hat{f}_c}{\hat{f}_r^2}.$$

Thus,

$$\widehat{\text{var}}(\hat{\lambda}) \approx \begin{bmatrix} \dfrac{1}{\hat{f}_r}, -\dfrac{\hat{f}_c}{\hat{f}_r^2} \end{bmatrix} \begin{bmatrix} \widehat{\text{var}}(\hat{f}_c) & 0 \\ 0 & \widehat{\text{var}}(\hat{f}_r) \end{bmatrix} \begin{bmatrix} \dfrac{1}{\hat{f}_r} \\ -\dfrac{\hat{f}_c}{\hat{f}_r^2} \end{bmatrix}.$$

Example (3) – variance of back-transformed estimates - simple

In Chapter 6, we demonstrated how we can 'back-transform' from the estimate of β on the logit scale to an estimate of some parameter θ (e.g., φ or p) on the probability scale (which is bounded $[0, 1]$). But, we're clearly also interested in an estimate of the variance (precision) of our estimate, on both scales. Your first thought might be to simply back-transform from the link function (in our example, the logit link), to the probability scale, just as we did above. But, as discussed in chapter 6, this does not work.

For example, consider the male Dipper data. Using the logit link, we fit the time-invariant model $\{\varphi_{.}\, p_{.}\}$ to the data. Let's consider only the estimate for $\hat{\varphi}$. The estimate for $\hat{\beta}$ for φ is 0.2648275. Thus, our estimate of $\hat{\varphi}$ on the probability scale (which is what **MARK** reports) is:

$$\hat{\varphi} = \frac{e^{0.2648275}}{1 + e^{0.2648275}} = \frac{1.303206}{2.303206} = 0.5658226.$$

But, what about the variance? Well, if we look at the β estimates, **MARK** reports that the standard error for the estimate of β corresponding to survival is 0.1446688. If we simply back-transform this from the logit scale to the probability scale, we get:

$$\widehat{\mathrm{SE}} = \frac{e^{0.1446688}}{1 + e^{0.1446688}} = \frac{1.155657}{2.155657} = 0.5361043.$$

However, **MARK** reports the estimated standard error for φ as 0.0355404, which isn't even remotely close to our back-transformed value of 0.5361043.

What has happened? Well, hopefully you now realize that you're 'transforming' the estimate from one scale (logit) to another (probability). And, since you're working with a 'transformation', you need to use the Delta method to estimate the variance of the back-transformed parameter.

Since

$$\hat{\varphi} = \frac{e^{\hat{\beta}}}{1 + e^{\hat{\beta}}},$$

then

$$\begin{aligned}
\widehat{\mathrm{var}}(\hat{\varphi}) &\approx \left(\frac{\partial \hat{\varphi}}{\partial \hat{\beta}}\right)^2 \times \widehat{\mathrm{var}}(\hat{\beta}) \\
&= \left(\frac{e^{\hat{\beta}}}{1 + e^{\hat{\beta}}} - \frac{\left(e^{\hat{\beta}}\right)^2}{1 + \left(e^{\hat{\beta}}\right)^2}\right)^2 \times \widehat{\mathrm{var}}(\hat{\beta}) \\
&= \left(\frac{e^{\hat{\beta}}}{\left(1 + e^{\hat{\beta}}\right)^2}\right)^2 \times \widehat{\mathrm{var}}(\hat{\beta}).
\end{aligned}$$

It is again worth noting that if

$$\hat{\varphi} = \frac{e^{\hat{\beta}}}{1 + e^{\hat{\beta}}},$$

then it can be easily shown that

$$\hat{\varphi}(1-\hat{\varphi}) = \frac{e^{\hat{\beta}}}{\left(1+e^{\hat{\beta}}\right)^2},$$

which is the derivative of φ with respect to β.

So, we could rewrite our expression for the variance of $\hat{\varphi}$ conveniently as

$$\widehat{\text{var}}(\hat{\varphi}) \approx \left(\frac{e^{\hat{\beta}}}{\left(1+e^{\hat{\beta}}\right)^2}\right)^2 \times \widehat{\text{var}}(\hat{\beta}) = \left(\hat{\varphi}\left(1-\hat{\varphi}\right)\right)^2 \times \widehat{\text{var}}(\hat{\beta}).$$

From **MARK**, the estimate of the SE for $\hat{\beta}$ was 0.1446688. Thus, the estimate of var(β) is $(0.1446688)^2$ = 0.02092906. Given the estimate of $\hat{\beta}$ of 0.2648275, we substitute into the preceding expression, which yields

$$\begin{aligned}\widehat{\text{var}}(\hat{\varphi}) &\approx \left(\frac{e^{\hat{\beta}}}{\left(1+e^{\hat{\beta}}\right)^2}\right)^2 \times \widehat{\text{var}}(\hat{\beta}) \\ &= (0.0603525 \times 0.02092906) \\ &= 0.001263.\end{aligned}$$

So, the estimated SE for $\hat{\varphi}$ is $\sqrt{0.001263}$ = 0.0355404, which is what is reported by **MARK**.

begin sidebar

SE and 95% CI

The standard approach to calculating 95% confidence limits for some parameter θ is $\theta \pm (1.96 \times \text{SE})$. Is this how **MARK** calculates the 95% CI on the real probability scale? Well, take the example we just considered – the estimated SE for $\hat{\varphi}$ = 0.5658226 was $\sqrt{0.001263}$ = 0.0355404. So, you might assume that the 95% CI on the real probability scale would be $0.5658226 \pm (2 \times 0.0355404)$: [0.4947418, 0.6369034].

However, this is not what is reported by **MARK** - [0.4953193, 0.6337593], which is quite close, but not exactly the same. Why the difference? The difference is because **MARK** first calculated the 95% CI on the logit scale, before back-transforming to the real probability scale. So, for our estimate of $\hat{\varphi}$, the 95% CI on the logit scale for $\hat{\beta}$ = 0.2648275 is [−0.0187234, 0.5483785], which, when back-transformed to the real probability scale is [0.4953193, 0.6337593], which is what is reported by **MARK**.

In this case, the very small difference between the two CI's is because the parameter estimate was quite close to 0.5. In such cases, not only will the 95% CI be nearly the same (for estimates of 0.5, it will be identical), but they will also be symmetrical.

However, because the logit transform is not linear, the *reconstituted* 95% CI will not be symmetrical around the parameter estimate, especially for parameters estimated near the [0, 1] boundaries. For example, consider the estimate for $\hat{p}$ = 0.9231757. On the logit scale, the 95% CI for the β corresponding to p ($\widehat{\text{SE}}$ = 0.5120845) is [1.4826128, 3.4899840]. The back-transformed CI is [0.8149669, 0.9704014], which is what is reported by **MARK**. This CI is clearly **not** symmetric around $\hat{p}$ = 0.9231757. The degree of asymmetry is a function of how close the estimated parameter is to either the 0 or 1 boundary.

Further, the estimated variance for $\hat{p}$:

$$\begin{aligned}\widehat{\text{var}}(\hat{p}) &\approx \left[\hat{p}(1-\hat{p})\right]^2 \times \widehat{\text{var}}(\hat{\beta}) \\ &= \left[0.9231757(1-0.9231757)\right]^2 \times 0.262231 \\ &= 0.001319,\end{aligned}$$

yields an estimated SE of 0.036318 on the normal probability scale (which is what is reported by **MARK**).

Estimating the 95% CI on the probability scale as $0.9231757 \pm (2 \times 0.036318)$ yields [0.85054, 0.99581], which is clearly quite a bit different, and more symmetrical, than what is reported by **MARK** (from above, [0.8149669, 0.9704014]). **MARK** uses the back-transformed CI to ensure that the reported CI is bounded [0, 1]. As the estimated parameter approaches either the 0 or 1 boundary, the degree of asymmetry in the back-transformed 95% CI that **MARK** reports will increase.

end sidebar

Example (4) – variance of back-transformed estimates - harder

In Chapter 6 we considered the analysis of variation in the survival of the European Dipper, as a function of whether or not there was a flood in the sampling area. Here, we consider just the male Dipper data (the encounter data are contained in `ed_males.inp`). Recall that a flood occurred during over the second and third intervals. For convenience, we'll assume that encounter probability is constant over time, and that survival is a linear function of '`flood`'.

Using a logit link function, where '`flood`' years were coded in the design matrix using a '`1`', and '`non-flood`' years were coded using a '`0`', the estimated linear model for survival on the logit scale was:

$$\text{logit}(\hat{\varphi}) = 0.4267863 - 0.5066372(\texttt{flood})$$

So, in a flood year:

$$\begin{aligned}\text{logit}(\hat{\varphi}_{\text{flood}}) &= 0.4267863 - 0.5066372(\texttt{flood}) \\ &= 0.4267863 - 0.5066372(1) \\ &= -0.0798509\end{aligned}$$

Back-transforming onto the real probability scale yields the precise value reported by **MARK**:

$$\begin{aligned}\hat{\varphi}_{\text{flood}} &= \frac{e^{-0.0798509}}{1 + e^{-0.0798509}} \\ &= 0.48005.\end{aligned}$$

Now, what about the estimated variance for φ_{flood}? First, what is our 'transformation function' (Y)? Simple – it is the 'back-transform' of the linear equation on the logit scale.

Given that:

$$\begin{aligned}\text{logit}(\hat{\varphi}) &= \beta_1 + \beta_2(\texttt{flood}) \\ &= 0.4267863 - 0.5066372(\texttt{flood}),\end{aligned}$$

then the back-transform function Y is

$$\hat{Y} = \frac{e^{0.4267863 - 0.5066372(\texttt{flood})}}{1 + e^{0.4267863 - 0.5066372(\texttt{flood})}}.$$

Second, since our transformation clearly involves multiple parameters (β_1, β_2), the estimate of the variance is given to first-order by

$$\begin{aligned} \widehat{\text{var}}(\hat{Y}) &\approx \mathbf{D}\boldsymbol{\Sigma}\mathbf{D}^{\mathsf{T}} \\ &= \left[\frac{\partial(\hat{Y})}{\partial(\hat{\theta})}\right] \cdot \widehat{\sum} \cdot \left[\frac{\partial(\hat{Y})}{\partial(\hat{\theta})}\right]^{\mathsf{T}} \end{aligned}$$

Given our linear (transformation) equation, then the vector of partial derivatives is (we've transposed it to make it easily fit on the page):

$$\begin{aligned} &\left[\left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_1}\right) \quad \left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_2}\right)\right]^{\mathsf{T}} \\ &= \begin{bmatrix} \dfrac{e^{\beta_1 + \beta_2(\texttt{flood})}}{1 + e^{\beta_1 + \beta_2(\texttt{flood})}} - \dfrac{\left(e^{\beta_1 + \beta_2(\texttt{flood})}\right)^2}{\left(1 + e^{\beta_1 + \beta_2(\texttt{flood})}\right)^2} \\ \dfrac{\texttt{flood} \times e^{\beta_1 + \beta_2(\texttt{flood})}}{1 + e^{\beta_1 + \beta_2(\texttt{flood})}} - \dfrac{\texttt{flood} \times \left(e^{\beta_1 + \beta_2(\texttt{flood})}\right)^2}{\left(1 + e^{\beta_1 + \beta_2(\texttt{flood})}\right)^2} \end{bmatrix} \end{aligned}$$

While this is fairly 'ugly' looking, the structure is quite straightforward – the only difference between the 2 elements of the vector is that the numerator of both terms (on either side of the minus sign) are multiplied by 1, and `flood`, respectively. Where do these scalar multipliers come from? They're simply the partial derivatives of the linear model (we'll call it Y) on the logit scale:

$$Y = \text{logit}(\hat{\varphi}) = \beta_1 + \beta_2(\texttt{flood}),$$

with respect to each of the parameters (β_i) in turn. In other words, $\partial Y/\partial\beta_1 = 1$, and $\partial Y/\partial\beta_2 = \texttt{flood}$.

Substituting in our estimates for $\hat{\beta}_1 = 0.4267863$ and $\hat{\beta}_2 = -0.5066372$, and setting `flood=1` (to indicate a 'flood year') yields:

$$\left[\left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_1}\right) \quad \left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_2}\right)\right] = \begin{bmatrix} 0.249602 & 0.249602 \end{bmatrix}$$

From the **MARK** output (after exporting to a dBase file – and **not** to the Notepad – in order to get full precision), the full V-C matrix for the parameters β_1 and β_2 is:

$$\widehat{\text{cov}}(\hat{\beta}_1, \hat{\beta}_2) = \widehat{\sum} = \begin{bmatrix} 0.0321405326 & -0.0321581167 \\ -0.0321581167 & 0.0975720877 \end{bmatrix}.$$

So,

$$\widehat{\text{var}}(\hat{Y}) \approx \begin{bmatrix} 0.249602 & 0.249602 \end{bmatrix} \times \begin{bmatrix} 0.0321405326 & -0.0321581167 \\ -0.0321581167 & 0.0975720877 \end{bmatrix} \times \begin{bmatrix} 0.249602 \\ 0.249602 \end{bmatrix}$$

$$= 0.0040742678.$$

The estimated SE for the variance for the reconstituted value of survival for an individual during a 'flood year' is $\sqrt{0.0040742678} = 0.0638300$, which is what is reported by **MARK** (to within rounding error).

begin sidebar

Once again...SE and 95% CI

As noted in the precceding example, the standard approach to calculating 95% confidence limits for some parameter θ is $\theta \pm (1.96 \times \text{SE})$. However, to guarantee that the calculated 95% CI is $[0, 1]$ bounded for parameters that are $[0, 1]$ bounded (like φ), **MARK** first calculates the 95% CI on the logit scale, before back-transforming to the real probability scale. However, because the logit transform is not linear, the *reconstituted* 95% CI will not be symmetrical around the parameter estimate, especially for parameters estimated near the $[0, 1]$ boundaries.

For the present example, the estimated value of survival for an individual during a 'flood year' ($\hat{\varphi}_{\text{flood}} = 0.48005$), **MARK** reports a 95% CI of $[0.3586850, 0.6038121]$. But, where do the values $[0.3586850, 0.6038121]$ come from? Clearly, they are not based on $0.48005 \pm 1.96(\text{SE})$. Given $\widehat{\text{SE}} = 0.06383$, this would yield a 95% CI of $[0.35494, 0.60516]$, which is close, but not exactly what **MARK** reports.

In order to derive the 95% CI, we first need to calculate the variance (and SE) of the estimate *on the logit scale*. In the preceding example, this was very straightforward, since the model we considered had a single β term for the parameter of interest. Meaning, we could simply use the estimated SE for β to derive the 95% CI on the logit scale, which we then back-transformed onto the real probability scale.

For the present example, however, the parameter is estimated from a function (transformation) involving more than one β term. In this example, the linear equation, which for consistency with the preceding we will denote as Y, was written as:

$$\hat{Y} = \text{logit}(\hat{\varphi}) = \beta_1 + \beta_2(\texttt{flood})$$

Thus, the estimated variance of $\text{logit}(\hat{\varphi}_{\text{flood}})$ is approximated as

$$\widehat{\text{var}}(\hat{Y}) \approx \mathbf{D}\boldsymbol{\Sigma}\mathbf{D}^{\mathsf{T}}$$

$$= \begin{bmatrix} \frac{\partial(\hat{Y})}{\partial(\hat{\beta}_1)} & \frac{\partial(\hat{Y})}{\partial(\hat{\beta}_2)} \end{bmatrix} \cdot \widehat{\sum} \cdot \begin{bmatrix} \frac{\partial(\hat{Y})}{\partial(\hat{\beta}_1)} & \frac{\partial(\hat{Y})}{\partial(\hat{\beta}_2)} \end{bmatrix}^{\mathsf{T}}.$$

Since

$$\begin{bmatrix} \frac{\partial(\hat{Y})}{\partial(\hat{\beta}_1)} & \frac{\partial(\hat{Y})}{\partial(\hat{\beta}_2)} \end{bmatrix} = [1 \quad \texttt{flood}] = [1 \quad 1],$$

and the VC matrix for $\hat{\beta}_1$ and $\hat{\beta}_2$ is

$$\widehat{\text{cov}}(\hat{\beta}_1, \hat{\beta}_2) = \widehat{\sum} = \begin{bmatrix} 0.0321405356 & -0.0321581194 \\ -0.0321581194 & 0.0975720908 \end{bmatrix},$$

then

$$\begin{aligned}\widehat{\text{var}}(\hat{Y}) &\approx \mathbf{D}\boldsymbol{\Sigma}\mathbf{D}^{\mathsf{T}} \\ &= \begin{bmatrix}1 & 1\end{bmatrix}\begin{bmatrix}0.0321405356 & -0.0321581194 \\ -0.0321581194 & 0.0975720908\end{bmatrix}\begin{bmatrix}1 \\ 1\end{bmatrix} \\ &= 0.065396.\end{aligned}$$

So, the $\widehat{\text{SE}}$ – on the logit scale! – is $\sqrt{0.065396} = 0.255727$. Thus, the 95% CI on the estimate on the logit scale, $\text{logit}(\hat{\varphi}_{\text{flood}}) = -0.0798509 \pm 1.96(0.255727) = [-0.581076, 0.421374]$.

All that is left is to back-transform the limits on the CI to the real probability scale:

$$[-0.94321, 0.78351] \longrightarrow \left[\frac{e^{-0.581076}}{1+e^{-0.581076}}, \quad \frac{e^{0.421374}}{1+e^{0.421374}}\right] = [0.358685, 0.603812]$$

which is what is reported by **MARK** (to within rounding error).

end sidebar

Example (5) – variance of back-transformed estimates - harder still

In Chapter 11, we considered analysis of the effect of various functions of mass, `m`, and mass-squared, `m2`, on the survival of a hypothetical species of bird (the simulated data are in file `indcov1.inp`). The linear function relating survival to `m` and `m2`, *on the logit scale*, is:

$$\text{logit}(\hat{\varphi}) = 0.2567341 + 1.1750463(\mathtt{m}_s) - 1.0554957(\mathtt{m}_s^2)$$

Note that for the two mass terms in the equation, there is a small subscript 's', reflecting the fact that these are '*standardized*' masses. Recall that we standardized the covariates by subtracting the mean of the covariate, and dividing by the standard deviation (the use of standardized or non-standardized covariates is discussed at length in Chapter 11).

Thus, for each individual in the sample, the estimated survival probability (on the logit scale) for that individual, given its mass, is given by:

$$\text{logit}(\hat{\varphi}) = 0.2567333 + 1.1750526\left(\frac{\mathtt{m}-\bar{\mathtt{m}}}{\text{SD}_{\mathtt{m}}}\right) - 1.0555024\left(\frac{\mathtt{m}^2-\bar{\mathtt{m}}^2}{\text{SD}_{\mathtt{m}^2}}\right)$$

In this expression, `m` refers to `mass` and $\mathtt{m}^2$ refers to `mass2`. The output from **MARK** (preceding page) actually gives you the mean and standard deviations for both covariates: for `mass`, mean = 109.9680, and SD = 24.7926, while for `mass2`, the mean = 12,707.4640, and the SD = 5,532.0322. The '**`value`**' column shows the standardized values for `mass` and `mass2` (0.803 and 0.752) for the first individual in the data file.

Now let's consider a worked example of the calculation of the variance of estimated survival. Suppose the mass of the bird was 110 units, so that `m` = 110, `m2` = $(110)^2$ = 12,100.

Thus:

$$\begin{aligned}\text{logit}(\hat{\varphi}) &= 0.2567333 + 1.1750526\left(\frac{(110-109.9680)}{24.7926}\right) - 1.0555024\left(\frac{(12{,}100-12{,}707.4640)}{5{,}532.0322}\right) \\ &= 0.3742\end{aligned}$$

So, if $\text{logit}(\hat{\varphi}) = 0.374$, then the reconstituted estimate of φ, transformed back from the logit scale is:

$$\frac{e^{0.374152}}{1 + e^{0.374152}} = 0.5925$$

Thus, for an individual weighing 110 units, the expected annual survival probability is approximately 0.5925 (which is what **MARK** reports if you use the '**`User specify covariate`**' option).

What about the variance (and corresponding SE) for this estimate?

First, what is our 'transformation function' (Y)? For the present example, it is the 'back-transform' of the linear equation on the logit scale. Given that:

$$\begin{aligned}\text{logit}(\hat{\varphi}) &= \beta_1 + \beta_2(\mathtt{m}_s) + \beta_3(\mathtt{m}_s^2) \\ &= 0.2567333 + 1.1750526(\mathtt{m}_s) - 1.0555024(\mathtt{m}_s^2),\end{aligned}$$

then the back-transform Y is:

$$\hat{Y} = \frac{e^{0.2567333+1.1750526(\mathtt{m}_s)-1.0555024(\mathtt{m}_s^2)}}{1 + e^{0.2567333+1.1750526(\mathtt{m}_s)-1.0555024(\mathtt{m}_s^2)}}$$

As in the preceding example, since our transformation clearly involves multiple parameters ($\beta_1, \beta_2, \beta_3$), the estimate of the variance is given by:

$$\begin{aligned}\widehat{\text{var}}(\hat{Y}) &\approx \mathbf{D\Sigma D}^{\mathsf{T}} \\ &= \left[\frac{\partial(\hat{Y})}{\partial(\hat{\theta})}\right] \cdot \widehat{\sum} \cdot \left[\frac{\partial(\hat{Y})}{\partial(\hat{\theta})}\right]^{\mathsf{T}}\end{aligned}$$

Given our linear (transformation) equation (from above) then the vector of partial derivatives is is:

$$\begin{bmatrix}\left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_0}\right) \\ \left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_1}\right) \\ \left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_2}\right)\end{bmatrix} = \begin{bmatrix}\frac{e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}}{1 + e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}} - \frac{\left[e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}\right]^2}{\left[1 + e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}\right]^2} \\ \frac{\mathtt{m} \times e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}}{1 + e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}} - \frac{\mathtt{m} \times \left[e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}\right]^2}{\left[1 + e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}\right]^2} \\ \frac{\mathtt{m2} \times e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}}{1 + e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}} - \frac{\mathtt{m2} \times \left[e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}\right]^2}{\left[1 + e^{\beta_1 + \beta_2(\mathtt{m}) + \beta_3(\mathtt{m2})}\right]^2}\end{bmatrix}$$

Although this looks complicated, the structure is actually quite straightforward – the only difference between the 3 elements of the vector is that the numerator of both terms (on either side of the minus sign) are multiplied by 1, `m`, and `m2`, respectively, which are simply the partial derivatives of the linear model (we'll call it Y) on the logit scale:

$$\hat{Y} = \text{logit}(\hat{\varphi}) = \beta_1 + \beta_2(\mathtt{m}_s) + \beta_3(\mathtt{m}_s^2),$$

with respect to each of the parameters (β_i) in turn. In other words, $\partial Y/\partial\beta_1 = 1$, $\partial Y/\partial\beta_2 = \mathtt{m}$, and $\partial Y/\partial\beta_3 = \mathtt{m2}$.

So, now that we have our vectors of partial derivatives of the transformation function with respect to each of the parameters, we can simplify things considerably by substituting in the standardized values for `m` and `m2`, and the estimated parameter values ($\hat{\beta}_1$, $\hat{\beta}_2$, and $\hat{\beta}_3$).

For a `mass` of 110 g, the standardized values for `m` and `m2` are:

$$\mathtt{m}_s = \left(\frac{110 - 109.9680}{24.7926}\right) = 0.0012895$$

$$\mathtt{m2}_s = \left(\frac{12100 - 12707.4640}{5532.0322}\right) = -0.1098085$$

The estimates for $\hat{\beta}_i$ we read directly from **MARK**:

$$\hat{\beta}_1 = 0.2567333,\ \hat{\beta}_2 = 1.1750526, \text{and } \hat{\beta}_3 = -1.0555024.$$

Substituting in these estimates for $\hat{\beta}_i$ and the standardized `m` and `m2` values into our vector of partial derivatives (which we've transposed in the following to save space) yields:

$$\left[\left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_1}\right) \quad \left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_2}\right) \quad \left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_3}\right)\right]^{\mathsf{T}} = \begin{bmatrix} 0.241451 \\ 0.000311 \\ -0.026513 \end{bmatrix}$$

From the **MARK** output (after exporting to a dBase file – and **not** to the editor – in order to get full precision), the full V-C matrix for the β parameters is

$$\begin{bmatrix} 0.0009006967 & -0.0004110129 & 0.0003662771 \\ -0.0004110129 & 0.0373928740 & -0.0364291221 \\ 0.0003662771 & -0.0364291221 & 0.0362817338 \end{bmatrix}$$

So,

$$\begin{aligned} \widehat{\mathrm{var}}(\hat{Y}) \approx & \begin{bmatrix} 0.241451 & 0.000311 & -0.026513 \end{bmatrix} \\ & \times \begin{bmatrix} 0.0009006967 & -0.0004110129 & 0.0003662771 \\ -0.0004110129 & 0.0373928740 & -0.0364291221 \\ 0.0003662771 & -0.0364291221 & 0.0362817338 \end{bmatrix} \times \begin{bmatrix} 0.241451 \\ 0.000311 \\ -0.026513 \end{bmatrix} \\ = & \ 0.000073867 \end{aligned}$$

So, the estimated SE for var for the reconstituted value of survival for an individual weighing 110 g is $\sqrt{0.000073867} = 0.0085946$, which is exactly what is reported by **MARK**.

It is important to remember that the estimated variance will vary depending on the mass you use - the estimate of the variance for a 110 g individual (0.000073867) will differ from the estimated variance for a (say) 120 g individual. For a 120 g individual, the standardized values of `m` and `m2` are 0.404636 and 0.3059512, respectively.

Based on these values, then:

$$\left[\left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_1}\right)\left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_2}\right)\left(\frac{\partial(\hat{Y})}{\partial\hat{\beta}_3}\right)\right]^{\mathsf{T}} = \begin{bmatrix} 0.239817 \\ 0.097039 \\ 0.073372 \end{bmatrix}.$$

Given the variance covariance-matrix for this model (shown above), then

$$\widehat{\text{var}}(\hat{Y}) \approx \mathbf{D\Sigma D}^{\mathsf{T}} = 0.000074246$$

Thus, the estimated SE for the reconstituted value of survival for an individual weighing 120 g is $\sqrt{0.000074246} = 0.0086166$, which again is exactly what is reported by **MARK**.

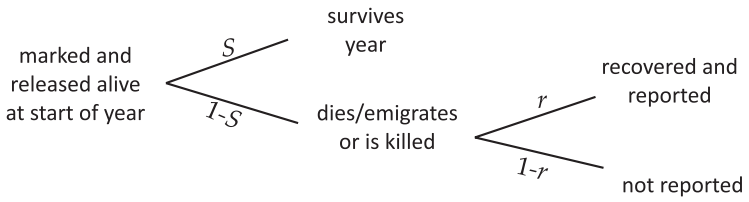
Note that this value for the SE for a 120 g individual (0.008617) differs from the SE estimated for a 110 g individual (0.008595), albeit not by much (the small difference here is because this is a very large simulated data set based on a deterministic model – see Chapter 11 for details). Since each weight would have its own estimated survival, and associated estimated variance and SE, to generate a curve showing the reconstituted values and their SE, you'd need to iteratively calculate $\mathbf{D\Sigma D}^{\mathsf{T}}$ over a range of weights. We'll leave it to you to figure out how to handle the programming if you want to do this on your own. For the less ambitious, **MARK** has the capacity to do much of this for you – you can output the 95% CI 'data' over a range of individual covariate values to a spreadsheet (see section 11.5 in Chapter 11).

Example (6) – estimating variance + covariance in transformations

Here, we consider application of the Delta method to joint estimation of the variance of a parameter, and the *covariance* of that parameter with another, where one of the two parameters is a linear transformation of the other. This is somewhat complicated, but quite useful example, since it illustrates how you can use the Delta method to estimate not only the variance of individual parameters, but the covariance structure among parameters as well.

There are many instances where the magnitude of the covariance is of particular interest. Here, we consider such a situation, in terms of different parameterizations for analysis of dead recovery data. Dead recovery models are covered in detail in Chapter 8 – here, we briefly review two different parameterizations (the 'Seber' and 'Brownie' parameterizations), and the context of our interest in the covariance between two different parameters.

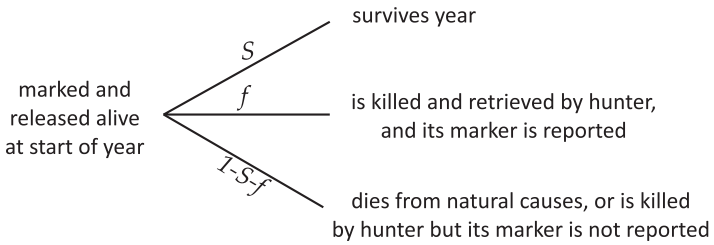
The encounter process for the Seber parameterization (1973: 254) is illustrated in the following:



Marked individuals are assumed to survive from release i to $i + 1$ with probability S_i. Individuals may die during the interval, either due to harvest or to 'natural' mortality. The probability that dead marked individuals are reported during each period i between releases, and (most generally) where the death is not necessarily related to harvest, is r_i. In other words, r_i is the joint probability of (i) the marked individual dying from either harvest or natural causes, and (ii) being recovered and reported (i.e., 'encountered').

Brownie *et al*. (1985) (hereafter, simply 'Brownie') developed a different parameterization for dead recovery data, where the sources of mortality (harvest, versus 'natural' or non-harvest) are modeled separately.

The encounter process for the Brownie parameterization is illustrated in the following:



Following Brownie, S_i is the probability that the individual survives the interval from release occasion i to $i+1$ (note that the definition for the probability of survival is logically identical between the Seber and Brownie parameterizations). The probability that the individual dies from either source of mortality is simply $1-S$. However, in contrast to the Seber parameterization, Brownie specified a parameter f, to represent the probability that an individual dies specifically due to harvest during interval i, and is reported ('encountered'). Thus, the probability that the individuals dies from natural causes is $(1-S-f)$.

Under the Seber parameterization, the probability of the encounter history '11' is given as $r(1-S)$. Under the Brownie parameterization, the expected probability of this event is simply f. Since the encounter history is the same, we can set the different parameterizations for the expected probability of the event equal to each other, generating the following expressions relating the two parameterizations:

$$f_i = r_i(1-S_i) \qquad r_i = \frac{f_i}{(1-S_i)}$$

Clearly, the parameter r_i is a reduced parameter, and can be expressed as a function of two other parameters normally found in the Brownie parameterization. An obvious practical question is, why choose one parameterization over the other, and does it matter?

This issue is discussed more fully in Chapter 8, but for now, we focus on the left-hand expression:

$$f_i = r_i(1-S_i).$$

So, given estimates of $\hat{r}_i$ and $\hat{S}_i$ from a Seber analysis, we could use this algebraic relationship (i.e., transformation) to generate estimates of $\hat{f}$. Naturally, we wish to be able to estimate $\widehat{\text{var}}(\hat{f})$.

However, in addition, we are potentially interested in estimating the covariance $\widehat{\text{cov}}(\hat{f}, \hat{S})$. Recall from above that the parameter f relates in part to the probability of being harvested. We might naturally be interested in the relationship between harvest mortality f, and overall annual survival, S. For example, if harvest and natural mortality are strictly additive, then we might expect a negative covariance between survival and harvest (i.e., as the probability of mortality due to harvest increases, annual survival will decrease). Whether or not the covariance is negative has important implications for harvest management (see full discussion in the Williams, Nichols & Conroy 2001 book).

We'll begin by considering estimation of the variance for $\hat{f}$ only, using the Delta method. Let the transformation g be $f = (1-S)r$.

Given $\hat{S}$, $\hat{r}$, $\widehat{\text{var}}(\hat{S})$ and $\widehat{\text{var}}(\hat{r})$, then the Jacobian for g is

$$\begin{bmatrix} \frac{\partial g}{\partial S} & \frac{\partial g}{\partial r} \end{bmatrix} = \begin{bmatrix} -\hat{r} & 1-\hat{S} \end{bmatrix},$$

and thus

$$\widehat{\text{var}}(\hat{f}) \approx \begin{bmatrix} -\hat{r} & 1-\hat{S} \end{bmatrix} \cdot \widehat{\sum} \cdot \begin{bmatrix} -\hat{r} \\ 1-\hat{S} \end{bmatrix},$$

where $\widehat{\sum}$ is the variance-covariance matrix for S and r:

$$\widehat{\sum} = \begin{bmatrix} \widehat{\text{var}}(\hat{S}) & \widehat{\text{cov}}(\hat{S},\hat{r}) \\ \widehat{\text{cov}}(\hat{S},\hat{r}) & \widehat{\text{var}}(\hat{r}) \end{bmatrix}.$$

So,

$$\widehat{\text{var}}(\hat{f}) \approx \begin{bmatrix} -\hat{r} & 1-\hat{S} \end{bmatrix} \cdot \widehat{\sum} \cdot \begin{bmatrix} -\hat{r} \\ 1-\hat{S} \end{bmatrix},$$

yields

$$\widehat{\text{var}}(\hat{f}) \approx \hat{r}^2 \widehat{\text{var}}(\hat{S}) - 2\hat{r} \cdot \widehat{\text{cov}}(\hat{S},\hat{r}) + 2\hat{r} \cdot \widehat{\text{cov}}(\hat{S},\hat{r})\hat{S} + \widehat{\text{var}}(\hat{r}) - 2 \cdot \widehat{\text{var}}(\hat{r})S + \widehat{\text{var}}(\hat{r})S^2,$$

which, with a little re-arranging, yields

$$\widehat{\text{var}}(\hat{f}) \approx \hat{r}^2 \cdot \widehat{\text{var}}(\hat{S}) - 2\left[(1-\hat{S})\hat{r}\right] \cdot \widehat{\text{cov}}(\hat{S},\hat{r}) + (1-\hat{S})^2 \cdot \widehat{\text{var}}(\hat{r}),$$

If you substitute in $r = f/(1-S)$ into the preceding expression, we end up with

$$\widehat{\text{var}}(\hat{f}) \approx \left(\frac{\hat{f}}{1-\hat{S}}\right)^2 \cdot \widehat{\text{var}}(\hat{S}) - 2\hat{f} \cdot \widehat{\text{cov}}(\hat{S},\hat{r}) + (1-\hat{S})^2 \cdot \widehat{\text{var}}(\hat{r}).$$

Now, what if instead of $\widehat{\text{var}}(\hat{f})$ only, we are also interested in estimating the *covariance* of (say) f and S? Such a covariance might be of interest since f is a function of S, and there may be interest in the degree to which S varies as a function of f (see above). Thus, we want to apply the Delta method to a function (the covariance) of two parameters, f and S. The key step here is recognizing that there are in fact two different functions (or, transformations) involved, which we'll call g_1 and g_2:

$$g_1 : S \rightarrow S \qquad \text{and} \qquad g_2 : (1-S)r \rightarrow f$$

You might be puzzled by $g_1 : S \rightarrow S$. In fact, this represents a null transformation – a direct, non-transformative 1:1 mapping between S under the Seber parameterization, and survival under the Brownie parameterization (since the probability of surviving is, logically, the same under the two parameterizations). This is analogous to generating the estimate for $\hat{S}_i$ under one paramterization by multiplying the same estimate under the other parameterization by the scalar constant 1.

Thus, with two transformations, we generate a Jacobian *matrix* of partial derivatives of each transformations with respect to S and r, respectively:

$$\begin{bmatrix} \dfrac{\partial g_1}{\partial \hat{S}} & \dfrac{\partial g_1}{\partial \hat{r}} \\ \dfrac{\partial g_2}{\partial \hat{S}} & \dfrac{\partial g_2}{\partial \hat{r}} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial \hat{S}}{\partial \hat{S}} & \dfrac{\partial \hat{S}}{\partial \hat{r}} \\ \dfrac{\partial \hat{f}}{\partial \hat{S}} & \dfrac{\partial \hat{f}}{\partial \hat{r}} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 \\ -r & 1-\hat{S} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 \\ -\dfrac{\hat{f}}{(1-\hat{S})} & 1-\hat{S} \end{bmatrix}.$$

Given the variance-covariance matrix $\widehat{\sum}$ for $\hat{S}$ and $\hat{r}$

$$\widehat{\sum} = \begin{bmatrix} \widehat{\text{var}}(\hat{S}) & \widehat{\text{cov}}(\hat{S},\hat{r}) \\ \widehat{\text{cov}}(\hat{S},\hat{r}) & \widehat{\text{var}}(\hat{r}) \end{bmatrix},$$

we evaluate sampling variance-covariance matrix for $\hat{S}$ and $\hat{f}$ as the matrix product

$$\begin{bmatrix} 1 & 0 \\ -\dfrac{\hat{f}}{(1-\hat{S})} & 1-\hat{S} \end{bmatrix} \cdot \widehat{\sum} \cdot \begin{bmatrix} 1 & -\dfrac{\hat{f}}{(1-\hat{S})} \\ 0 & 1-\hat{S} \end{bmatrix},$$

which (after a bit of algebra) yields

$$\begin{bmatrix} \widehat{\text{var}}(\hat{S}) & -\dfrac{\hat{f}}{1-\hat{S}} \cdot \widehat{\text{var}}(\hat{S}) + (1-\hat{S}) \cdot \widehat{\text{cov}}(\hat{S},\hat{r}) \\ -\dfrac{\hat{f}}{1-\hat{S}} \cdot \widehat{\text{var}}(\hat{S}) + (1-\hat{S}) \cdot \widehat{\text{cov}}(\hat{S},\hat{r}) & \hat{r}^2 \cdot \widehat{\text{var}}(\hat{S}) - 2\left[(1-\hat{S})\hat{r}\right] \cdot \widehat{\text{cov}}(\hat{S},\hat{r}) + (1-\hat{S})^2 \cdot \widehat{\text{var}}(\hat{r}) \end{bmatrix}.$$

Here, matrix elements [1,1] and [2,2] are the expressions for the approximate variance of S and f, respectively (note that the expression in element [2,2], for $\widehat{\text{var}}(\hat{f})$, is identical to the expression we derived on the preceding page). Elements [1,2] and [2,1] (which are the same) are the expressions for the approximate *covariance* of f and S.

As noted earlier, interpretation of the estimated variance and covariance is dependent on the source of the variance-covariance matrix, $\widehat{\Sigma}$, used in the calculations. If $\widehat{\Sigma}$ is constructed using variances and covariances from the usual ML parameter estimates, then the resulting estimate for variance is an estimate of the *total* variance (i.e., sampling + process, where process variation represents the underlying 'biological' variation). In contrast, if $\widehat{\Sigma}$ is based on estimated *process* (variances and covariances only, then the estimate for variance is an estimate of the *process* variance. Decomposition of total variance into sampling and process components is covered in detail in Appendix D.

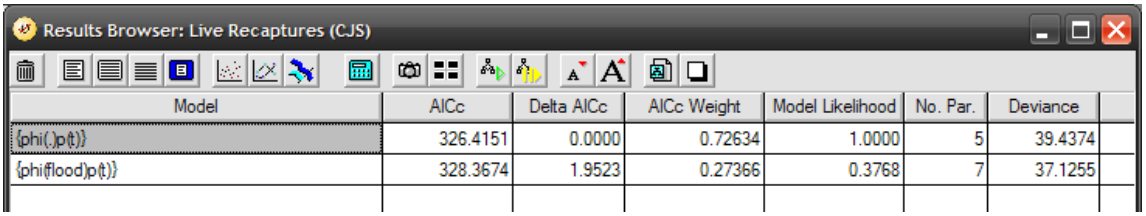
B.5. Delta method and model averaging

In the preceding examples, we focused on the application of the Delta method to transformations of parameter estimates from a single model. However, as introduced in Chapter 4 – and emphasized throughout the remainder of this book – we're often interested in accounting for model selection

uncertainty by using model-averaged values. There are no major complications for application of the Delta method to model-averaged parameter values – you simply need to make sure you use model-averaged values for each element of the calculations.

We'll demonstrate this using analysis of the male dipper data (`ed_male.inp`). Suppose that we fit 2 candidate models to these data: $\{\varphi_{\cdot}\, p_t\}$ and $\{\varphi_{flood}\, p_t\}$. In other words, a model where survival is constant over time, and a model where survival is constrained to be a function of a binary '`flood`' variable (see section 6.4 of Chapter 6).

Here are the results of fitting these 2 models to the data:



Model	AICc	Delta AICc	AICc Weight	Model Likelihood	No. Par.	Deviance
{phi(.)p(t)}	326.4151	0.0000	0.72634	1.0000	5	39.4374
{phi(flood)p(t)}	328.3674	1.9523	0.27366	0.3768	7	37.1255

As expected (based on the analysis of these data presented in Chapter 6), we see that there is some evidence of model selection uncertainty – the model where survival is constant over time has roughly 2-3 times the weight as does the 'flood' model':

The model averaged values for each interval are shown below:

	1	2	3	4	5	6
estimate	0.5673	0.5332	0.5332	0.5673	0.5673	0.5673
SE	0.0441	0.0581	0.0581	0.0441	0.0441	0.0441

Now, suppose we want to derive the best estimate of the probability of survival over (say) the first 3 intervals. Clearly, all we need to do is take the product of the 3 model-averaged values corresponding to the first 3 intervals:

$$\left(0.5673 \times 0.5332 \times 0.5332\right) = 0.1613.$$

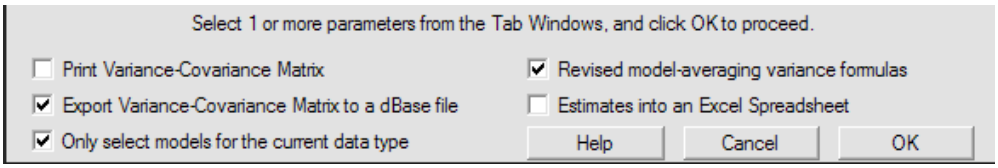
In other words, our best estimate of the probability that a male dipper would survive from the start of the time series to the end of the third interval is 0.1613.

What about the standard error of this product? Here, we use the Delta method. Recall that:

$$\begin{aligned}\widehat{\mathrm{var}}(\hat{Y}) &\approx \mathbf{D}\boldsymbol{\Sigma}\mathbf{D}^{\mathsf{T}} \\ &= \left[\frac{\partial(\hat{Y})}{\partial(\hat{\theta})}\right] \cdot \widehat{\sum} \cdot \left[\frac{\partial(\hat{Y})}{\partial(\hat{\theta})}\right]^{\mathsf{T}},\end{aligned}$$

where Y is some linear or nonlinear function of the parameter estimates $\hat{\theta}_1, \hat{\theta}_2, \ldots$. For this example, Y is the product of the survival estimates.

So, the first thing we need to do is to generate the estimated variance-covariance matrix for the model averaged survival estimates. This is easy enough to do – in the '**`Model Averaging Parameter Selection`**' window, you simply need to '**`Export Variance-Covariance Matrix to a dBase file`**' - you do this by checking the appropriate check box (lower-left, as shown at the top of the next page).



The 'rounded' values which would be output to the Notepad are shown below. (Remember, however, that for the actual calculations, you want to use the full precision variance-covariance matrix from the exported dBase file.)

```
 Unconditional Variance-Covariance Matrix of Model Averaged Estimates
     Variance-Covariance matrix of estimates on diagonal and below,
           Correlation matrix of estimates above diagonal.
     |    1           2           3
-----+-----------------------------------
     |
   1 |  0.00194     0.04923     0.04923
     |
   2 |  0.00013     0.00337     1.00000
     |
   3 |  0.00013     0.00337     0.00337
```

Remember, however, that for the actual calculations you need the full precision variance-covariance matrix from the exported dBase file.

All that remains is to substitute our model-averaged estimates for (i) $\hat{\varphi}$ and (ii) the variance-covariance matrix (above), into $\widehat{\text{var}}(\hat{Y}) \approx \mathbf{D\Sigma D}^{\mathsf{T}}$.

Thus,

$$\widehat{\text{var}}(\hat{Y}) \approx \left[\frac{\partial(\hat{Y})}{\partial(\hat{\theta})}\right] \cdot \widehat{\sum} \cdot \left[\frac{\partial(\hat{Y})}{\partial(\hat{\theta})}\right]^{\mathsf{T}}$$

$$= \begin{bmatrix}(\bar{\varphi}_2\bar{\varphi}_3) & (\bar{\varphi}_1\bar{\varphi}_3) & (\bar{\varphi}_1\bar{\varphi}_2)\end{bmatrix} \cdot \begin{bmatrix}\widehat{\text{var}}(\bar{\varphi}_1) & \widehat{\text{cov}}(\bar{\varphi}_1,\bar{\varphi}_2) & \widehat{\text{cov}}(\bar{\varphi}_1,\bar{\varphi}_3) \\ \widehat{\text{cov}}(\bar{\varphi}_1,\bar{\varphi}_2) & \widehat{\text{var}}(\bar{\varphi}_2) & \widehat{\text{cov}}(\bar{\varphi}_2,\bar{\varphi}_3) \\ \widehat{\text{cov}}(\bar{\varphi}_3,\bar{\varphi}_1) & \widehat{\text{cov}}(\bar{\varphi}_3,\bar{\varphi}_2) & \widehat{\text{var}}(\bar{\varphi}_3)\end{bmatrix} \cdot \begin{bmatrix}(\bar{\varphi}_2\bar{\varphi}_3) \\ (\bar{\varphi}_1\bar{\varphi}_3) \\ (\bar{\varphi}_1\bar{\varphi}_2)\end{bmatrix}$$

$$= \begin{bmatrix}0.284303069 & 0.3024783390 & 0.3024783390\end{bmatrix} \times \begin{bmatrix}0.0019410083 & 0.0001259569 & 0.0001259569 \\ 0.0001259569 & 0.0033727452 & 0.0033727423 \\ 0.0001259569 & 0.0033727423 & 0.0033727452\end{bmatrix} \times \begin{bmatrix}0.284303069 \\ 0.3024783390 \\ 0.3024783390\end{bmatrix}$$

$$= 0.001435.$$

B.6. Summary

In this appendix, we've briefly introduced a convenient, generally straightforward method for deriving an estimate of the sampling variance for transformations of one or more variables. Such transformations are quite commonly encountered when using **MARK**, and having a method to derive estimates of the sampling variances is convenient. The most straightforward method – based on a first-order Taylor series

expansion – is known generally as the 'Delta method'. However, a first-order Taylor series approximation may not always be appropriate, especially if the transformation is highly non-linear, and if there is significant variation in the data. In such case, you may have to resort to higher-order approximations, or numerically intensive bootstrapping approaches.

B.7. References

dos Santos Dias, C. T., Samaranayaka, A., and Manly, B. F. (2008) On the use of correlated beta random variables with animal population modelling. *Ecological Modelling*, **215**, 293-300.

Ver Hoef, J. M. (2012) Who Invented the Delta Method? *The American Statistician*, **66**, 124-127.

Addendum: 'computationally intensive' approaches

At the start of this appendix, we motivated the Delta method as an approach for deriving an estimate of the expectation or variance of a function of one or more parameters – specifically, an approach that was *not* 'compute-intensive'. While this approach has a certain elegance, application to complex functions can be cumbersome. Further, transformations that are strongly nonlinear near the mass of the data may necessitate using a higher-order Taylor series expansion, which again can be complex for a particular function.

In such cases, it is useful to at least be aware of alternative, compute-intensive approaches. Here, we briefly introduce two different approaches, applied to the estimation of the variance of the product of survival estimates, using the dipper example presented in section B.4. Again, we'll use estimates from model $\{\varphi_t\, p_\cdot\}$ fit to the male European dipper data set, and again, we'll suppose we're interested in the probability of surviving from the start of the first interval to the end of the third interval.

As noted in section B.4, the estimate of this probability is easy enough:

$$\begin{aligned}\hat{Y} &= \left(\hat{\varphi}_1 \times \hat{\varphi}_2 \times \hat{\varphi}_3\right) \\ &= \left(0.6109350 \times 0.458263 \times 0.4960239\right) = 0.138871.\end{aligned}$$

So, the estimated probability of a male Dipper surviving over the first three intervals is ~ 14% (again, assuming that our time-dependent survival model is a valid model).

i. using the Delta method...

To derive the estimate of the variance of the product using the Delta method, we require the variance-covariance matrix for the survival estimates:

$$\widehat{\mathrm{cov}}(\hat{Y}) = \widehat{\sum}$$

$$= \begin{bmatrix} \widehat{\mathrm{var}}(\hat{\varphi}_1) & \widehat{\mathrm{cov}}(\hat{\varphi}_1, \hat{\varphi}_2) & \widehat{\mathrm{cov}}(\hat{\varphi}_1, \hat{\varphi}_3) \\ \widehat{\mathrm{cov}}(\hat{\varphi}_2, \hat{\varphi}_1) & \widehat{\mathrm{var}}(\hat{\varphi}_2) & \widehat{\mathrm{cov}}(\hat{\varphi}_2, \hat{\varphi}_3) \\ \widehat{\mathrm{cov}}(\hat{\varphi}_3, \hat{\varphi}_1) & \widehat{\mathrm{cov}}(\hat{\varphi}_3, \hat{\varphi}_2) & \widehat{\mathrm{var}}(\hat{\varphi}_3) \end{bmatrix}$$

$$= \begin{bmatrix} 0.0224330125 & -0.0003945405 & 0.0000654469 \\ -0.0003945405 & 0.0099722201 & -0.0002361998 \\ 0.0000654469 & -0.0002361998 & 0.0072418858 \end{bmatrix}.$$

For this example, the transformation we're applying to our 3 survival estimates (which we'll call Y) is the product of the estimates (i.e., $\hat{Y} = \hat{\varphi}_1\hat{\varphi}_2\hat{\varphi}_3$).

Thus, our variance estimate is given as

$$\widehat{\text{var}}(\hat{Y}) \approx \left[\left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_1}\right) \quad \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_2}\right) \quad \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_3}\right)\right] \cdot \widehat{\sum} \cdot \begin{bmatrix} \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_1}\right) \\ \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_2}\right) \\ \left(\frac{\partial(\hat{Y})}{\partial\hat{\varphi}_3}\right) \end{bmatrix}.$$

Each of the partial derivatives for $\hat{Y}$ is easy enough to derive for this example. Since $\hat{Y} = \hat{\varphi}_1\hat{\varphi}_2\hat{\varphi}_3$, then $\partial\hat{Y}/\partial\hat{\varphi}_1 = \hat{\varphi}_2\hat{\varphi}_3$. And so on.

Expanding the preceding results in:

$$\begin{aligned}\widehat{\text{var}}(\hat{Y}) \approx\ & \hat{\varphi}_2^2\hat{\varphi}_3^2[\widehat{\text{var}}(\hat{\varphi}_1)] + \hat{\varphi}_1^2\hat{\varphi}_3^2[\widehat{\text{var}}(\hat{\varphi}_2)] + \hat{\varphi}_1^2\hat{\varphi}_2^2[\widehat{\text{var}}(\hat{\varphi}_3)] \\ & + 2\hat{\varphi}_2\hat{\varphi}_3^2\hat{\varphi}_1[\widehat{\text{cov}}(\hat{\varphi}_1, \hat{\varphi}_2)] + 2\hat{\varphi}_2^2\hat{\varphi}_3\hat{\varphi}_1[\widehat{\text{cov}}(\hat{\varphi}_1, \hat{\varphi}_3)] + 2\hat{\varphi}_1^2\hat{\varphi}_3\hat{\varphi}_2[\widehat{\text{cov}}(\hat{\varphi}_2, \hat{\varphi}_3)].\end{aligned}$$

After substituting in our estimates for φ_i and the variances and covariances, our estimate for the variance of the product $\hat{Y} = (\hat{\varphi}_1\hat{\varphi}_2\hat{\varphi}_3)$ is (to first-order) $\widehat{\text{var}}(Y) = 0.0025565$.

Now, we consider a couple of 'compute-intensive' approaches.

ii. simulation from a multivariate normal distribution...

The basic idea behind the approach we illustrate here is straightforward: we (i) simulate data as random draws from a multivariate normal distribution with known means and variance-covariance, (ii) generate the product of these random draws, and (iii) derive numerical estimates of the mean (expectation) and variance of these products.

However, we need to be a bit careful here. If we simulate the random normal draws on the real probability scale, then we run the risk of simulating random values which are not plausible, because they fall outside the $[0, 1]$ interval (e.g., you could simulate a survival probability > 1, or < 0, neither of which are possible). To circumvent this problem, we simulate random normal variables on the logit scale (i.e., logit-normal deviates) using the β estimates and the variance-covariance matrix (both estimated on the logit scale), back-transform the random deviates from the logit $\rightarrow$ real probability scale, and then generate the product on the real probability scale.*

For the male Dipper data, the β estimates using an identity design matrix (such that each β corresponds to the survival estimate for that interval – see Chapter 6 for specifics) are: $\hat{\beta}_1 = 0.4512441$, $\hat{\beta}_2 = -0.1673372$, $\hat{\beta}_3 = -0.0159047$. The variance-covariance matrix for the β estimates (which can be output

* An alternate approach would be to simulate correlated random values drawn from a beta distribution which is constrained on the interval $[0, 1]$, with shape parameters α and β determined by estimated parameters and variances of those estimates. Computationally, this can be done by first generating standard normal variates with the required covariance structure, and then transforming them to beta variates with the required mean and standard deviation. See dos Santos Dias *et al*. 2008.

from **MARK**) is:

$$\widehat{\text{cov}}(\hat{Y}) = \widehat{\sum}$$

$$= \begin{bmatrix} \widehat{\text{var}}(\hat{\beta}_1) & \widehat{\text{cov}}(\hat{\beta}_1, \hat{\beta}_2) & \widehat{\text{cov}}(\hat{\beta}_1, \hat{\beta}_3) \\ \widehat{\text{cov}}(\hat{\beta}_2, \hat{\beta}_1) & \widehat{\text{var}}(\hat{\beta}_2) & \widehat{\text{cov}}(\hat{\beta}_2, \hat{\beta}_3) \\ \widehat{\text{cov}}(\hat{\beta}_3, \hat{\beta}_1) & \widehat{\text{cov}}(\hat{\beta}_3, \hat{\beta}_2) & \widehat{\text{var}}(\hat{\beta}_3) \end{bmatrix}$$

$$= \begin{bmatrix} 0.3970573440 & -0.0066861059 & 0.0011014411 \\ -0.0066861059 & 0.1618026250 & -0.0038059631 \\ 0.0011014411 & -0.0038059631 & 0.1158848865 \end{bmatrix}.$$

The following **R** script uses the `mvtnorm` package to simulate the multivariate normal data:

```
# include library to simulate correlated MV norm
library("mvtnorm")
```

Now, we set up the parameter values needed to specify the simulation:

```
# number of samples to take from MV norm
iter <- 1000000;

# dipper parameter values and var-covar -- on the logit scale -- to use in simulation
beta1 <- 0.4512441; beta2 <-  -0.1673372; beta3 <-  -0.0159047;

vc <- matrix(c(0.3970573440,-0.0066861059,0.0011014411,
              -0.0066861059,0.1618026250,-0.0038059631,
               0.0011014411,-0.0038059631,0.1158848865),3,3,byrow=T);

 # generate rannor samples conditional betas and VC matric
logit_samples = rmvnorm(iter,mean=c(beta1,beta2,beta3),sigma=vc,method="svd")
```

Now, we do a visual check to confirm our simulated variance-covariance matrix is close to the estimated matrix (above) – which it is:

```
 # check to confirm simulated VC is correct...
cat("simulated VC matrix")
print(round(cov(logit_samples),10))

simulated VC matrix
             [,1]         [,2]         [,3]
[1,]  0.396846751 -0.006892055  0.001056857
[2,] -0.006892055  0.161725609 -0.003646518
[3,]  0.001056857 -0.003646518  0.115931880
```

Then, we simply back-transform our samples from the logit → real probability scale, and proceed from there.

```
# convert logit samples to data frame
logit_samples <- as.data.frame(logit_samples)

# back-transform from logit scale
real_samples <- exp(logit_samples)/(1+exp(logit_samples));

# generate the product of back-transformed deviates
real_samples$prod = real_samples[,1]*real_samples[,2]*real_samples[,3];

# summary stats
cat("expectation of product =", mean(real_samples$prod))
cat("variance of product =", var(real_samples$prod))
```

Running this script results in the following estimates, which are quite close to the expected product (0.138871), and variance of the product derived using the Delta method (0.0025565):

```
expectation of product = 0.1370664
 variance of product = 0.002377359
```

iii. using MCMC...

Another approach makes use of the Markov Chain Monte Carlo (MCMC) capabilities in **MARK**. Here, we provide only a brief description of the idea, and mechanics – for a more complete discussion, see Appendix E.

The basic idea is as follows. We'll fit model $\{\varphi_t\, p_\cdot\}$ to the male Dipper data, and use MCMC to derive estimates of the survival and encounter parameters, based on estimated moments (mean, median, or mode), and associated variances, from the posterior distribution for each of the parameters. The posterior distribution for each parameter is generated by Markov sampling over the joint probability distribution for all parameters, given the data.

If we were using a specialized MCMC application, like **JAGS**, or **BUGS**, we could simply create a derived parameter as a function of other structural parameters in the model (say, prod = $\varphi_1 \times \varphi_2 \times \varphi_3$), and then analyze the posterior samples for this derived parameter (this ability to explicitly code functions of parameters is one of the real conveniences of using MCMC, typically in a Bayesian framework).

The MCMC capabilities in **MARK** do not allow the explicit construction of a user-specified derived parameter. However, we can accomplish much the same thing, albeit in a slightly more 'brute-force' way, by simply

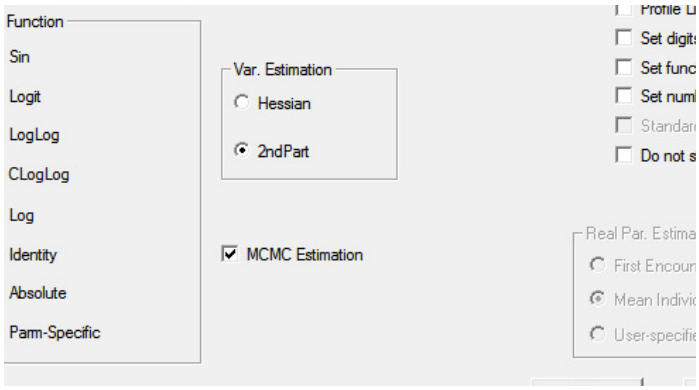
(i) taking the individual sample chains from the MCMC simulations for each of the 3 real parameters involved in the 'function' of interest (i.e., the product – $\varphi_1 \rightarrow \varphi_3$),

(ii) deriving the function of these parameters over the chains – in this case taking their product, and finally,

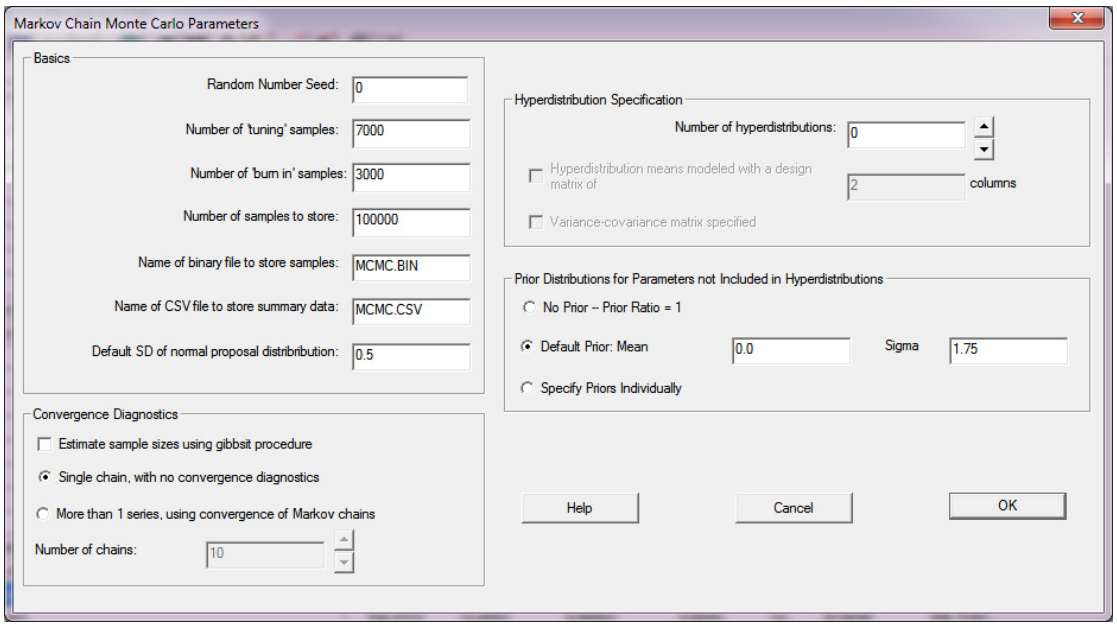
(iii) evaluating this product as the posterior distribution for the product (which it is). In fact, this is equivalent to what **JAGS** or **BUGS** does, except that instead of calculating the product of the survival parameters at each step of the sampler, we simply do it *post hoc* – after the samplers are finished.

OK, let's see how this is done. First, we fit model $\{\varphi_t\, p_\cdot\}$ to the male Dipper data. We'll use a logit link (for reasons discussed in Appendix E).

Next, re-run this model. But, before submitting the model for numerical estimation for the second time, we first check the '**MCMC Estimation**' box:



Once you click the '**OK to Run**' button, **MARK** will respond with a window where you specify the MCMC parameters that will specify aspects of the numerical estimation (see Appendix E for a complete discussion of these parameters).



What is generally important is that we want a sufficient number of samples (at all stages) to ensure that the samplers have converged on the stationary joint distribution. For this example we've used 7,000 'tuning' samples, 3,000 'burn in' samples, and 100,000 samples from the posterior distribution. We've also specified only a single chain, with no convergence diagnostics.

Once finished, **MARK** will output the results to the editor. If you scroll down to near the bottom of the output listing, you'll see various macro values that can be used for post-processing of the chains for each parameter. These macro values are copied into **SAS** or **R** programs that are provided in the **MARK** helpfile. We'll demonstrate the mechanics using **R**.

For the male Dipper data, and model $\{\varphi_t\, p.\}$, the **R** macro values are:

```
ncovs <- 7;  # Number of beta estimates
nmeans <- 0;  # Number of mean estimates
ndesigns <- 0; # Number of design matrix estimates
nsigmas <- 0; # Number of sigma estimates
nrhos <- 0; # Number of rho estimates
nlogit <- 7; # Number of real estimates
filename <- "C:\\USERS\\USER\\DESKTOP\\MCMC.BIN";  # path MCMC.BIN file
```

So, all we do is copy this into the appropriate section at the top of the **R** script provided in the **MARK** helpfile. The script is fairly lengthy, so we won't reproduce it in full here. Instead we'll focus on the additional steps you'll need to execute in order to derive an estimate of the variance for the product of the first 3 survival estimates.

First, copy the macro variables (above) into the **R** script, and execute it 'as is'. This will create an MCMC 'object', called 'mcmcdata', that is compatible with one of several **R** packages (e.g., coda). This object contains each of the individual Markov chains, for each parameter.

Normally, what you'd do at this point is use some package, like coda, to post-process the chains, and generate various descriptive statistics and associated graphics. However, what we want to do here is estimate the variance of the product of $(\varphi_1 \times \varphi_2 \times \varphi_3)$. As outlined earlier, we will (i) extract the chains for the survival parameters φ_1, φ_2 and φ_3 from the mcmcdata object, (ii) take their product, and (iii) generate various descriptive statistics for this product.

While there are any number of ways you might do this in **R**, the following works well enough. The first thing we do is convert the MCMC 'object' (mcmcdata) to a dataframe. We'll call this new dataframe 'chaindata':

```
chaindata <- as.data.frame(mcmcdata);
```

Next, we'll add a column to this new dataframe for the product $(\varphi_1 \times \varphi_2 \times \varphi_3)$, and label this new column 'prod'. Note, in the dataframe, these parameters are referred to (by their column names, which are explicitly set by the preceding **R** script) as 'real1', 'real2', and 'real3', respectively:

```
chaindata$prod <- chaindata$real1*chaindata$real2*chaindata$real3;
```

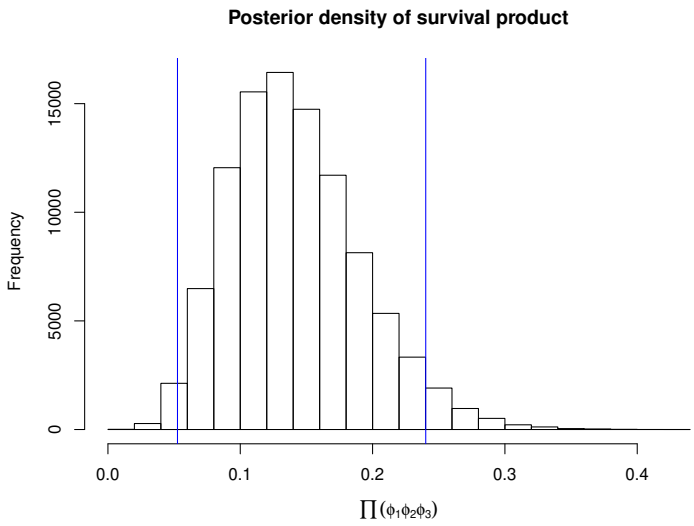
All that's left is to look at the summary statistics for this product:

```
summary(chaindata$prod)
     Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.02155 0.10520 0.13620 0.14140 0.17180 0.47390

var(chaindata$prod)
 [1] 0.002485553
```

As with the preceding approach, this script on the MCMC data results in moment estimates which are quite close to the expected product (0.138871), and variance of the product derived using the Delta method (0.0025565).

Couple of things to note. First, because the posterior for 'prod' is asymmetrically distributed around the most frequent value (as shown in the histogram at the top of the next page), the median of this



distribution, 0.13620, is perhaps the most appropriate moment to characterize the posterior, and is quite close to the expected product – somewhat closer than the mean.

Second, from a Bayesian perspective, it might be more meaningful to consider the credible interval, rather than the point estimate. Because of the asymmetry of the posterior distribution, use of the HPD (highest posterior density – see Appendix E) might be the most appropriate way to specify the interval. The upper and lower bounds of the 95% HPD for '`prod`', [0.05256, 0.24017], are plotted as vertical blue lines in the preceding frequency histogram.