Quantifying uncertainty

Christopher L. Cahill

Introduction

Quantifying uncertainty represents a critical component of fisheries modeling and as a result there exists extensive literature on this topic (e.g., see Hilborn and Walters 1992; Magnusson et al. 2013; Monnahan et al. 2017). This section outlines a minimum set of principles and guidelines for describing uncertainty in parametric statistical models. We first develop uncertainty from a likelihood perspective because the likelihood function represents a core concept in statistical inference (Hilborn and Mangel 1997; Pawitan 2001; Bolker 2008). For example, if one begins with a likelihood function it is then possible to describe uncertainty as a frequentist using the long-run frequency of events via a statistical test (e.g., Bolker 2008), as a likelihoodist by comparing likelihoods themselves to explore the plausibility of parameter values given observed data (Hilborn and Mangel 1997; Pawitan 2001), or even as a Bayesian via the inclusion of prior information (Punt and Hilborn 1997; Gelman et al. 2013). The differences among these approaches are deeply philosophical and controversy remains regarding the use of each approach in the literature (e.g., Mayo 1996; Royall 1997; Gelman et al. 2013). However, in the context of modern stock assessment modeling nearly all methods used to quantify parametric uncertainty ultimately rely on the use of one or more likelihood functions, which by definition must convey all information contained within a given dataset as it pertains to an assumed model's parameters. Consequently, a thorough understanding of likelihood-based descriptions of uncertainty is important regardless of the philosophical approach adopted for an analysis (e.g., see Royle and Dorazio 2008; Gelman et al. 2013). In the sections below, we re-introduce likelihood theory and use it to motivate three distinct descriptions of frequentist uncertainty in maximum likelihood models: Hessian-based uncertainty, profile likelihood, and bootstrap or

simulation-based methods. Lastly, we provide a brief and high-level overview of Bayesian methods as an alternative approach to describing uncertainty in statistical models.

A crash course in likelihood theory

We begin by re-introducing likelihood approaches. We start by assuming that an analyst has data, a statistical model that describes those data probabilistically, and that they seek to estimate the parameters of this model via maximum likelihood (see chapter XX). It is worth noting that while the examples below are chosen for simplicity, the methods themselves can be applied to highly parameterized models of arbitrary complexity (see models in Fournier and Archibald 1982; Thorson and Kristensen 2024). This focus on simplicity allows us to describe concepts surrounding uncertainty using both mathematics as well as visual representations of toy problems.

An analyst begins a maximum likelihood estimation problem by specifying a statistical model f parameterized by fixed, but unknown parameter(s) θ to describe observed data Y:

$$f(Y \mid \theta). \qquad (1)$$

In this case, the likelihood $\mathcal{L}$ then reverses the role of data and parameters, and in doing so describes the probability of having observed data Y as a function of θ:

$$\mathcal{L}(\theta \mid Y). \qquad (2)$$

Here, $\mathcal{L}$ conveys information from Y about the unknown quantity of interest θ. For discrete data this definition is directly applicable since probabilities are nonzero, while for continuous data the probability of observing data in an infinitesimally small region near Y is approximately equal to the probability density function times a small constant (Lee et al. 2017). Maximum likelihood estimation then proceeds from these definitions to find an estimator $\hat{\theta}$ that maximizes $\mathcal{L}(\theta \mid Y)$:

$$\hat{\theta} = \text{argmax}\mathcal{L}(\theta \mid Y). \qquad (3)$$

Because the likelihood values of individual datum are often very small numerical quantities, and because law of probability states that the joint probability of two independent events A and B occurring is $A \cdot B$, one often works with the natural logarithm of likelihood values to prevent numerical overflow and underflow issues (i.e., the log-likelihood). Maximum likelihood estimates of $\hat{\theta}$ can sometimes be determined analytically, but often numerical search methods are used to find values of $\hat{\theta}$ that maximize this function. One additional point to note is that most numerical optimization algorithms actually minimize rather than maximize functions (e.g., see Nash 2014), and thus one typically finds the maximum likelihood estimates $\hat{\theta}$ via the minimization of a negative log-likelihood function $\hat{\theta} = \text{argmin} - \ln\mathcal{L}(\theta \mid Y)$.

To visualize how a likelihood function carries information about $\hat{\theta}$ consider a scenario where a biologist samples fish from a pond for marks at three different levels of sampling effort. In this example, we assume the true capture probability of individual fish is $\theta_{true} = 0.6$ and that the biologist sampled the pond on three occasions and which resulted in n = 5, n = 15, and n = 30 fish examined for marks, respectively. We define a statistical model for this problem as a Bernoulli probability mass function for marked $y_i = 1$ and unmarked $y_i = 0$ individuals with capture probability p:

$$P(Y = y_i) = p^{y_i}(1-p)^{1-y_i}, \quad \text{for } Y \in \{0,1\} \qquad (4)$$

Given that each observation y_i follows a Bernoulli distribution with parameter p, the likelihood function for each set of observations collected by the biologist can be expressed as

$$\mathcal{L}(p \mid Y) = \prod_{i=1}^{n} p^{y_i}(1-p)^{1-y_i} \qquad (5)$$

and which can be broken down into successes (where $y_i = 1$) and failures (where $y_i = 0$). This particular model is simple enough to apply analytical methods. To do so, we first let

$$S = \sum_{i=1}^{n} y_i \qquad (6)$$

be the total number of successes and where n is sample size on a particular occasion sampled by the biologist. Doing so allows us to simplify the likelihood function in Eq. 5, take its log, and differentiate it with respect to zero to solve for the maximum likelihood estimate:

$$\begin{gathered} \ln\big(\mathcal{L}(p \mid Y)\big) = S\ln(p) + (n - S)\ln(1 - p) \\ \frac{\partial}{\partial p}\ln\big(\mathcal{L}(p \mid Y)\big) = \frac{S}{p} - \frac{n - S}{1 - p} = 0 \\ S(1 - p^*) = (n - S)p^* \\ p^* = \frac{S}{n} \end{gathered} \qquad (7)$$

where p^* represents the maximum likelihood estimate. The plot below visualizes both the normalized (i.e., scaled to have unit maximum) likelihood and negative log-likelihood surfaces according to these equations for each hypothetical dataset.

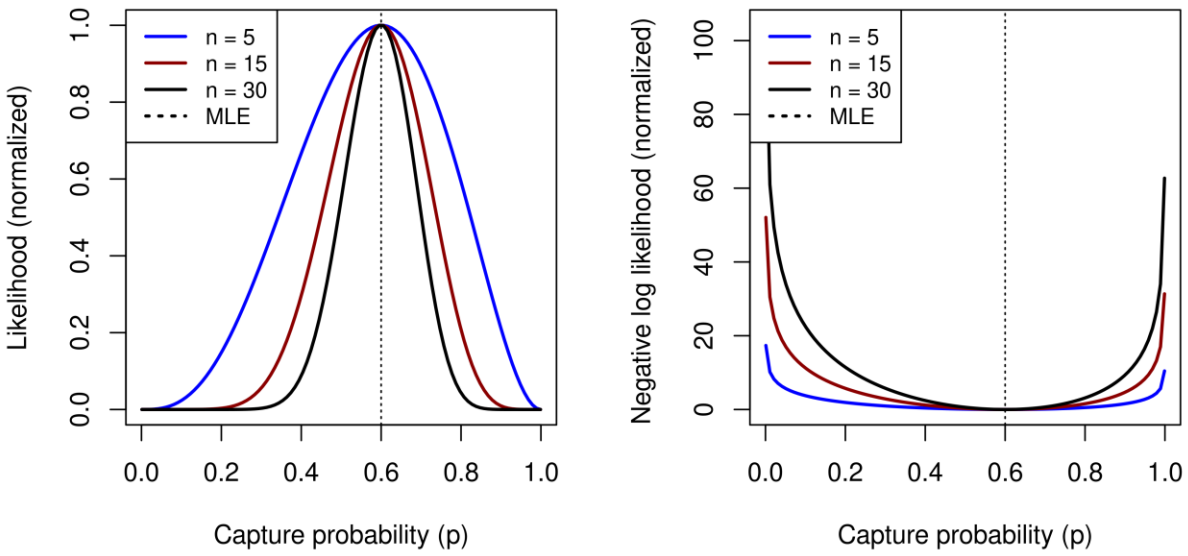

Figure 1. Left panel: Normalized likelihoods for a range of capture probabilities for each of three datasets with differing sample sizes. Right panel: visualizing the normalized negative log-likelihoods corresponding to the likelihoods on the left panel.

From these plots it is clear that in the absence of any other information it is the likelihood function rather than the maximum likelihood estimate alone that conveys information about the unknown quantity of interest p^* (see arguments in Pawitan 2001; Royle and Dorazio 2008). Additionally, we see from this plot that the inclusion of additional data results in the likelihood function becoming narrower around p^*. We might even intuit that there is something about the curvature of $\mathcal{L}(p \mid Y)$ in the location of p^* that tells us about the plausibility of a given value of p, and we use this intuition to motivate asymptotic standard errors in the sections below.

Method one: uncertainty derived from the Hessian

Given a generic maximized log-likelihood function such as that in Eq. 3 we can estimate parameter uncertainty by determining the rate at which the log-likelihood decays as we move away from the maximum likelihood estimate $\hat{\theta}$. Mathematically, this quantity is represented as

the second derivative. Large second derivatives imply steep changes around the point $\hat{\theta}$ and more certainty in an estimate while smaller derivatives imply flatter surfaces and suggest lower precision. In a new hypothetical example where $\hat{\theta}$ now represents a vector of two parameters θ_1 and θ_2, the matrix of second derivatives (i.e., the Hessian matrix) of the log likelihood function at the maximum likelihood estimate $\hat{\theta}$ is defined as

$$H(\hat{\theta}) = \begin{bmatrix} \frac{\partial^2 \log \mathcal{L}(\hat{\theta} \mid Y)}{\partial \theta_1^2} & \frac{\partial^2 \log \mathcal{L}(\hat{\theta} \mid Y)}{\partial \theta_1 \, \partial \theta_2} \\ \frac{\partial^2 \log \mathcal{L}(\hat{\theta} \mid Y)}{\partial \theta_2 \, \partial \theta_1} & \frac{\partial^2 \log \mathcal{L}(\hat{\theta} \mid Y)}{\partial \theta_2^2} \end{bmatrix} \quad (8)$$

where the elements along the diagonal describe curvature of the log likelihood function in the direction of the corresponding parameter, and the off-diagonal elements describe how parameters interact with one another. The asymptotic normality property of the maximum likelihood estimator states that as sample size n approaches infinity, the distribution of the maximum likelihood estimator $\hat{\theta}$ approaches that of a normal distribution (Pawitan 2001; Royle and Dorazio 2008). This implies that for some problems with large sample sizes we can construct approximate confidence intervals for $\hat{\theta}$ using a normal distribution approximation. To do so, we calculate the standard errors of the parameter estimates as

$$SE(\hat{\theta}_i) \approx \sqrt{-\left(H(\hat{\theta})^{-1}\right)_{ii}} \quad (9)$$

where ii refers to the i^{th} diagonal element of the inverse of the Hessian matrix, and the leading negative sign arises from the definition of the Fisher information matrix (not shown) and ensures that variance remains positive. Given $\hat{\theta}$ and $SE(\hat{\theta})$, we calculate asymptotic confidence intervals as

$$\hat{\theta} \pm z_{\alpha/2} \cdot SE(\hat{\theta}) \quad (10)$$

where $z_{\alpha/2}$ represents the critical value from a standard normal distribution given a chosen significance level α specified by the analyst.

In frequentist statistics, confidence intervals are interpreted as the range within which the unknown θ_{true} would fall with probability $1-\alpha$ given repeated experimentation. Phrased differently, if an experiment or data collection scheme was repeated many times and the model re-fitted to each new dataset, we might expect the confidence intervals for an estimated quantity from each replicate to include the unknown but true parameter value with probability $1-\alpha$. (Pawitan 2001; Bolker 2008). This notion of hypothetical repeated sampling or replicating a procedure is fundamental to frequentist statistical methodologies (Pawitan 2001). In the box below, we calculate the maximum likelihood estimates and 95% confidence intervals of capture probability p for the pond scenario above assuming $n = 30$. We calculate these quantities using analytical and numerical search methods and use base R and `RTMB` for all examples (see Kristensen 2024).

```
# Hessian-based uncertainty via analytical and numerical solutions
library(RTMB)
dat <- list(X = c(
    1, 1, 0, 1, 0, 1, 1, 0, 0, 1,
    1, 0, 1, 0, 1, 0, 1, 1, 0, 1,
    1, 0, 0, 1, 0, 1, 1, 1, 0, 1
))
par <- list(logit_p = 0) # starting value, logit space

f <- function(par) {
    p <- plogis(par$logit_p) # -> exp (logitp)/ (1+ exp (logitp))
    -sum(dbinom(x = dat$X, size = 1, p = p, log = TRUE))
}

obj <- MakeADFun(f, par, silent = TRUE)
opt <- nlminb(obj$par, obj$fn, obj$gr)
cov_mat <- solve(obj$he())
SE <- sqrt(diag(cov_mat))
SE

[1] 0.372678

summary(sdreport(obj)) # automated way
```

```
          Estimate Std. Error
logit_p 0.4054651   0.372678
```

```
# Calculate critical z-value for 95% confidence interval
alpha <- 0.05
z <- qnorm(1 - alpha / 2) # --> 1.96

# approximate 95% CI
logit_CI <- opt$par + c(-1, 1) * z * SE
plogis(logit_CI)  # numerical
```

```
[1] 0.4194649 0.7569262
```

```
# compare with analytical solution
p_hat <- mean(dat$X)
n <- length(dat$X)
SE_analytical <- sqrt(p_hat * (1 - p_hat) / n)
p_hat + c(-1, 1) * z * SE_analytical # analytical
```

```
[1] 0.4246955 0.7753045
```

In this example we see that the asymptotic approximation and analytical solution correspond closely to one another, and we direct readers to Royle and Dorazio (2008) for the mathematics of the analytical solution for the variance of a Bernoulli random variable. Analytical and numerical solutions like this work well when sample sizes are large, when the likelihood function is smooth, and when certain regularity conditions are met (Pawitan 2001; see Lee et al. 2017). In situations where the uncertainty of a derived quantity is also needed, the delta method (or generalized delta method in the case of mixed effects models) can also be used to derive an asymptotic distribution for a random variable of interest (see Kristensen et al. 2016). The delta-method is implemented in RTMB via the `ADREPORT()` macro (Kristensen et al. 2016), but the mathematics of the approach is beyond the scope of this chapter.

The above descriptions of uncertainty rely heavily on the smooth and approximately quadratic geometry of the maximum likelihood function that often arises with large n. However, in applied situations with small sample sizes, when parameters are estimated near bounds, or in situations with highly skewed or irregular likelihood surfaces asymptotic uncertainty estimates can be unreliable. Moreover, these problems often cannot be detected using standard errors

calculated via the Hessian matrix or the delta-method (see also Auger-Méthé et al. 2021), which motivates the following section on profile likelihood.

Method two: profile likelihood

Profile likelihood is a method for "mapping out" the likelihood surface in one parameter dimension, which helps an analyst explore how precise an estimate is and whether the likelihood function for a parameter is well defined. In our simple pond example above, we were able to map out and visualize the entire likelihood function for capture probability p because this model was simple and featured a single parameter (Figure 1). We now consider more complicated scenarios with multi-parameter models to demonstrate the generality of the profile likelihood approach. In addition to the situations noted above, profile likelihood methods are useful for dealing with nuisance parameters, which arise when an analyst is interested in the values of one parameter rather than the values of all parameters at once (Cole et al. 2014). In the context of stock assessment models which estimate many parameters (Quinn and Deriso 1999), the profile likelihood represents an important tool for detecting model identifiability issues. To develop a profile likelihood, one follows these steps:

1. Estimate $\hat{\theta}$ for the model in question using maximum likelihood, noting that $\hat{\theta}$ now represents a vector parameters:

$$\hat{\theta} = \text{argmin} - \ln\mathcal{L}(\theta \mid Y). \qquad (11)$$

2. Generate a sequence of values to profile a parameter of interest θ_i across, and compute the profile negative log-likelihood as:

$$-\ln\mathcal{L}_{\text{profile}}(\theta_i) = \underset{\theta_{-i}}{\text{argmin}} - \ln\mathcal{L}(\theta_i, \theta_{-i} \mid Y), \qquad (12)$$

noting that we fix the value of θ_i at one value in the sequence and plug in all remaining parameters θ_{-i} at their maximum likelihood estimates. We record the optimized value of $-\ln\mathcal{L}_{\text{profile}}(\theta_i)$.

3. Repeat step two for each value in the sequence of θ_i values and plot values of $-ln\mathcal{L}_{\text{profile}}(\theta_i)$ vs. θ_i.

Figure 2. The good, the bad, and the ugly negative log likelihood profiles adapted from Auger-Methe et al. 2021.

In general, simply plotting the resulting negative log-likelihood profile often reveals whether a particular parameter is well-defined and behaves asymptotically or is altogether inestimable [see Auger-Méthé et al. (2021); Figure 2]. Assuming good (asymptotic) behavior is detected, the resulting profile likelihood can be used to develop asymptotic uncertainty intervals around θ_i via a likelihood ratio test, which is conceptually similar to the approaches in the previous section (see also Bolker 2008). A useful property of uncertainty intervals calculated using the likelihood ratio approach is that they are invariant to re-parameterization (Pawitan 2001). First, we calculate a profile likelihood ratio for each value in the profile sequence as

$$\Lambda(\theta_i) = \frac{\mathcal{L}(\hat{\theta})}{\mathcal{L}_{\text{profile}}(\theta_i)} \qquad (13)$$

and leverage the fact that the $-2\ln\big(\Lambda(\theta_i)\big)$ is asymptotically distributed as a chi-squared variable:

$$-2\ln\big(\Lambda(\theta i)\big) \sim \chi^2_{k=1,\alpha}. \qquad (14)$$

This expression can be algebraically manipulated to compute the negative log profile likelihood values falling within a specified interval about the original optimized negative log likelihood:

$$-\ln\mathcal{L}_{\text{profile}}(\theta_i) \geq -\ln\mathcal{L}(\hat{\theta}) + \frac{1}{2}\chi^2_{k=1,\alpha}. \qquad (15)$$

To demonstrate how these ideas translate into practice we code a simple linear regression with parameters β_0, β_1 and profile the error term σ in R below.

```
# profile log-sigma from a linear regression

# first, simulate
set.seed(123)
n <- 100
x <- rnorm(n, mean = 0, sd = 1)
beta0 <- 1
beta1 <- 2
sigma <- 1
y <- beta0 + beta1 * x + rnorm(n, mean = 0, sd = sigma)

# now, estimate
dat <- list(y = y, x = x)
par <- list(beta0 = 0, beta1 = 0, log_sig = 0)
f <- function(par) {
    getAll(dat, par, warn = FALSE) # works like attach()
    y <- OBS(y) # flags observations (used later for simulation)
    sig <- exp(log_sig)
    -sum(dnorm(y, beta0 + beta1 * x, sig, TRUE))
}
obj <- MakeADFun(f, par, silent = TRUE)
opt <- nlminb(obj$par, obj$fn, obj$gr)
map <- list("log_sig" = factor(NA)) # fix at value in par$log_sig
par$beta0 <- opt$par["beta0"] # set at MLE
par$beta1 <- opt$par["beta1"]

lsigs <- seq(from = -0.19, to = 0.13, length.out = 1000)
nll_prof <- rep(NA, length(lsigs))

# profile likelihood-"by hand"
for (i in 1:length(lsigs)) {
    par$log_sig <- lsigs[i]
    obj <- MakeADFun(f, par, map = map, silent = TRUE)
    nll_prof[i] <- nlminb(obj$par, obj$fn, obj$gr)$objective
}
target <- opt$objective + (1 / 2) * qchisq(0.95, 1)
CIs <- lsigs[order(abs(nll_prof - target))][1:2]
CIs

[1] -0.1723824  0.1053353

# note: an automatic way using library(TMB)
# lsig_pro <- TMB:::tmbprofile(obj, "log_sig")
```

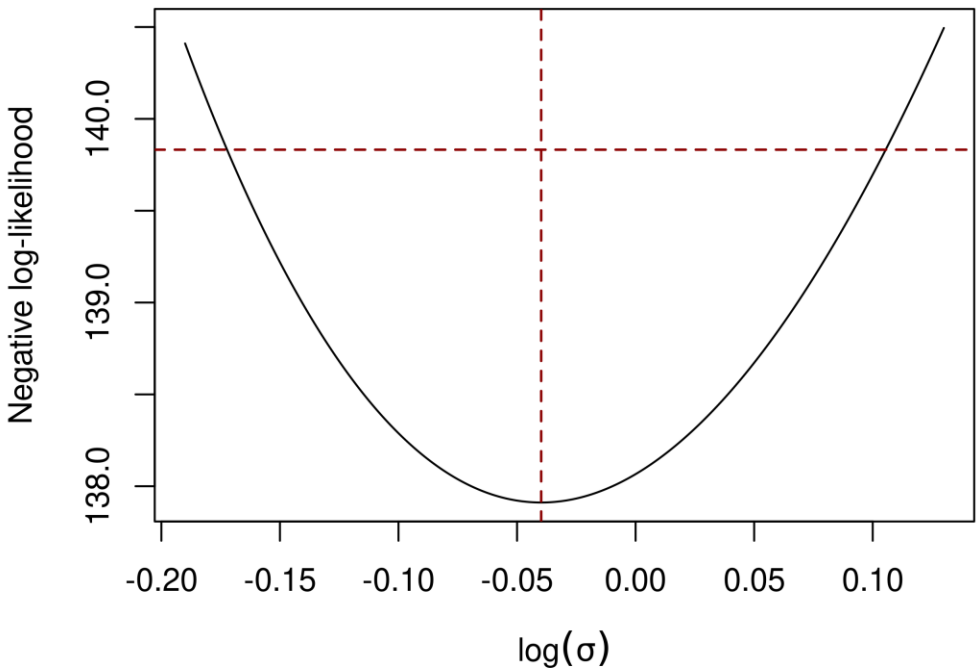

Figure 3. Negative log-likelihood profile for log sigma. The vertical line corresponds to the maximum likelihood estimate of log sigma, while the horizontal line corresponds to the values of the profile contained within the 95% confidence interval calculated using a likelihood ratio test.

Method three: bootstrap or simulation-based approaches

While the methods in the previous sections perform well when $\hat{\theta}$ behaves asymptotically, there are no guarantees of such behavior for most real-world problems. In cases where $\hat{\theta}$ misbehaves, all of the previous approximations can fail in unexpected ways. The resulting profile likelihoods are often skewed or highly irregular in these situations (Figure 2). It is in these situations that bootstrapping or simulation-based methods can be useful (Efron 1987; Chernick and LaBudde 2014). Bootstrapping randomly shuffles a model's residuals or samples the data many times with replacement and repeatedly fits the model in question to those shuffled datasets, which are then summarized accordingly. The same general approach can be used with Monte

Carlo simulation, in which a new sample is generated from a distribution based upon a fitted model.

The cost of the bootstrap and simulation approaches vs. the previous approaches used to characterize uncertainty is that it takes more computational effort to generate uncertainty estimates. The reward of these approaches is that unlike Hessian or profile-based approaches to calculating uncertainty, simulation-based methods assume nothing about the asymptotic behavior of $\hat{\theta}$ and can also be used to calculate uncertainty intervals around derived quantities. In terms of characterizing uncertainty, it is worth noting that there are still situations in which bootstrap methods do not work well: examples include situations with very small sample sizes or datasets with extreme values (Chernick and LaBudde 2014), and care must be taken when re-shuffling or sampling data with complex dependency structures (Roberts et al. 2017). Additionally, the number of replicate samples needed for accurately characterizing uncertainty depends upon the level at which one seeks to characterize a particular quantity (i.e., fewer samples are needed for 95% quantiles vs. the 99.9% quantiles). While these approaches assume less about asymptotic behavior of $\hat{\theta}$, care must still be taken to ensure the model assumptions are valid (i.e., see model validation section).

The bootstrap works by approximating the sampling distribution of an unknown statistic through repeated (empirical) resampling (Efron 1987). If we have data $Y = y_1, \ldots, y_n$ originating from an unknown distribution F, the statistic of interest $T(Y)$ also has an unknown sampling distribution. Bootstrapping treats the empirical distribution F_n based on the observed data as a proxy for F, and resamples Y with replacement to create bootstrap samples Y^* from which we compute $T(Y^*)$ from each sample. In doing so, bootstrapping effectively substitutes the unknown F with the observed data (see Pawitan 2001). This particular flavor of bootstrap is sometimes

called a ‘real’ or ‘true’ bootstrap in the literature (Efron 1987), as it quantifies uncertainty based on the assumption that both y and x were sampled randomly from a population of interest. We demonstrate the real bootstrap with the linear regression example in the box below.

```
# true bootstrap
dat <- list(y = y, x = x)
# note par same as above
obj <- MakeADFun(f, par, silent = TRUE)
opt <- nlminb(obj$par, obj$fn, obj$gr)

odat <- dat # save original data

do_one <- function() {
    idx <- sample(1:length(y), replace = TRUE)
    yboot <- y[idx]
    xboot <- x[idx]
    dat <<- list(y = yboot, x = xboot)
    objs <- MakeADFun(data = dat, f, par, silent = TRUE)
    opts <- nlminb(objs$par, objs$fn, objs$gr)
    opts$par
}
sim_bs <- replicate(1000, do_one())
t(apply(sim_bs, 1, function(x) quantile(x, probs = c(0.025, 0.5, 0.975))))
```

```
              2.5%         50%      97.5%
beta0    0.7104084  0.89918881 1.08516169
beta1    1.7424942  1.94882371 2.15550493
log_sig -0.2274563 -0.05369994 0.08501464
```

```
opt$par
```

```
     beta0      beta1    log_sig
 0.8972003  1.9475299 -0.0398241
```

A related approach is the simulation-based parametric bootstrap for generating uncertainty intervals, which simulates data conditional upon a fitted model and then re-fits the model to each of these new, simulated datasets. Conceptually, this approach is something of a cross between the true bootstrap method described above and a Monte Carlo simulation. The simulation approach is useful for describing uncertainty conditional upon an observed set of independent x axis values, or in situations where the levels of these independent variables represent an explicit experimental design choice. We demonstrate this simulation-based approach to generating uncertainty intervals in the box below.

```
# simulation based uncertainty
dat <- list(y = y, x = x)
# note par same as above
obj <- MakeADFun(f, par, silent = TRUE)
opt <- nlminb(obj$par, obj$fn, obj$gr)
obj$simulate()$y[1:10] # simulate first ten obs from fitted model
```

```
 [1]  0.1378705  0.6486596  4.4666451  0.6632228  1.2033439  5.4753530
 [7]  3.6068883 -2.0882701  0.3920951 -0.6210370
```

```
do_one <- function() {
    dat$y <<- obj$simulate()$y
    objs <- MakeADFun(f, par, silent = TRUE)
    opts <- nlminb(objs$par, objs$fn, objs$gr)
    opts$par
}
sim <- replicate(1000, do_one())
t(apply(sim, 1, function(x) quantile(x, probs = c(0.025, 0.5, 0.975))))
```

```
              2.5%         50%      97.5%
beta0    0.7212787  0.89950709 1.08405222
beta1    1.7465347  1.93744359 2.15175144
log_sig -0.1970770 -0.05365373 0.08146748
```

In this simple linear regression both the true bootstrap and simulation-based approaches result in similar 95% confidence intervals.

A related use of simulation is to determine whether the maximum likelihood estimates for a given model-data combination converge on 'true' values used for simulation and whether the coverage of confidence intervals is adequate (e.g., see Cahill et al. 2020 for an inland fisheries example). To see how this works, consider an important property of maximum likelihood theory that states that maximum likelihood estimates $\hat{\theta}_n$ converge to θ_{true} as sample size n increases:

$$\hat{\theta}_n \to \theta_{true} \quad \text{as} \quad n \to \infty. \qquad (16)$$

This so-called consistency property of the maximum likelihood estimator implies that as we increase sample size and repeat our data-collection and estimation scheme many times that the maximum likelihood estimates almost surely converge on θ_{true} (Pawitan 2001; Lee et al. 2017). In our experience, self-testing via simulation is a powerful tool for detecting model mis-

specification and coding issues in statistical models in general. We provide a visual display of self-testing below.

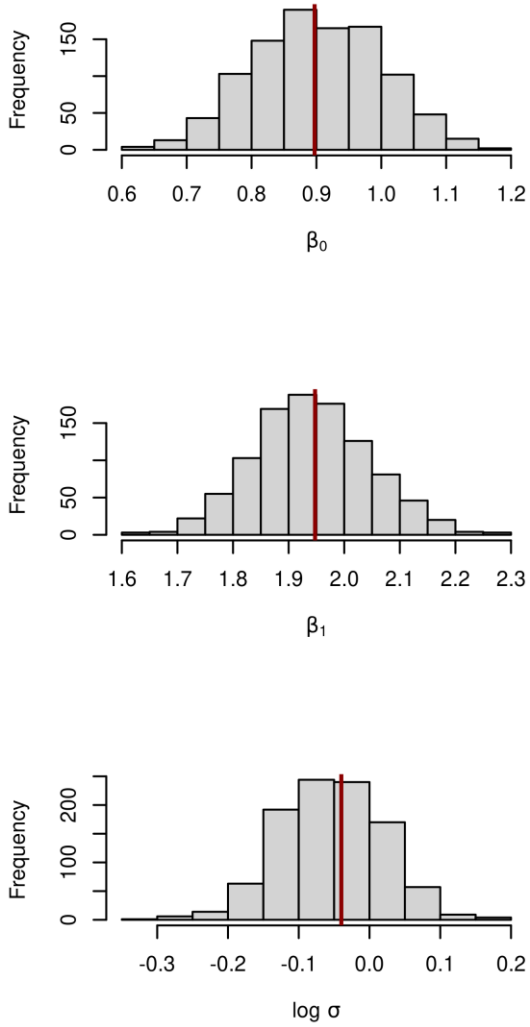

Figure 4. Distribution of maximum likelihood estimates from simulation replicates (histogram bars) vs. true values used for simulation (vertical lines).

In this case, the maximum likelihood estimates from the simulation appear distributed about the θ_{true}, which demonstrates that this statistical model can estimate the parameters used to simulate the data.

The Bayesian detour

Unlike the likelihood-based methods above that assume a particular parameter value is the random outcome of an experiment, Bayesian statistics begins from the axiom that an unknown parameter θ has a distribution with prior density $f(\theta)$ (Hobbs and Hooten 2015). From this axiom, Bayesian inference proceeds according to Bayes' rule:

$$f(\theta \mid Y) = \frac{f(Y \mid \theta) f(\theta)}{f(Y)} \qquad (17)$$

where $f(Y \mid \theta)$ is the likelihood of an observable quantity (data), $f(\theta)$ describes prior belief, $f(Y)$ is the marginal distribution of Y, and $f(\theta \mid Y)$ represents the posterior probability distribution upon which inferences are based. In a Bayesian analysis, the likelihood is combined with a prior distribution to generate a posterior distribution via Bayes' rule, and probability is the fundamental yardstick with which Bayesians quantify or measure uncertainty (Punt and Punt 1997; Hilborn and Mangel 1997; Gelman et al. 2013). Bayesians use Bayes' rule to combine prior beliefs and observed data to characterize uncertainty and update beliefs using the posterior distribution, and Eq. 17 usually is solved using numerical simulation methods like Markov chain Monte Carlo (Hobbs and Hooten 2015).

The Bayesian approach is elegant if one accepts the axiom that prior belief and observable quantities can be combined to describe uncertainty, but there are at least two issues that can arise in practice. First, how does one choose a prior? This choice is sometimes easy for simple models or parameters like a capture probability, which must fall between $[0,1]$ and thus have natural prior distributions, or in situations where previous studies provide relevant information on a particular prior. However, choosing minimally informative priors can be difficult in settings where no prior information exists, particularly for parameters in stock

assessments like initial abundance for which analysts often have little to no *a priori* knowledge (Punt and Punt 1997; Thorson and Cope 2017). Second, treating a prior density like a regular probability density function introduces a technical issue known as a lack of invariance to model reparameterization. This means that uninformative priors on one scale can actually be informative on different scales (Lele 2020), and as a result great care must be taken when specifying priors in general (Gelman et al. 2013).

With these considerations in mind, we proceed below and quantify uncertainty using Bayesian statistics with the simple linear regression example. We assume vague priors ranging from $-\infty$ to ∞ for all parameters, and rely on the fact that the denominator in Eq. 17 is constant with respect to θ, which allows us to write:

$$f(\theta \mid Y) \propto f(Y \mid \theta) \cdot f(\theta), \qquad (18)$$

and noting that $f(\theta)$ falls out of the equation if it is a uniform distribution. We use a variant of MCMC known as Hamiltonian Monte Carlo to generate samples from the posterior distribution $f(\theta \mid Y)$, which is widely available via the program Stan (Carpenter et al. 2017) and can be used with `RTMB` via `tmbstan` (Monnahan and Kristensen 2018). We run four independent chains for a 1000 iteration 'warmup' and then save 1000 iterations from each chain to sample the posterior distribution for a total of 4000 posterior samples.

```
# Bayesian
suppressMessages(library(tmbstan))
dat <- list(y = y, x = x)
# note par same as above
obj <- MakeADFun(f, par, silent = TRUE)
opt <- nlminb(obj$par, obj$fn, obj$gr)
options(mc.cores = parallel::detectCores()) # multiple cores
opt_bayes <- tmbstan(obj,
    lower = c(-Inf, -Inf, -Inf),
    upper = c(Inf, Inf, Inf),
    chains = 4, warmup = 1000, iter = 2000,
    refresh = 0 # shut up stan
)
```

```
opt_bayes

Inference for Stan model: RTMB.
4 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup draws=4000.

           mean se_mean   sd    2.5%     25%     50%     75%   97.5% n_eff
Rhat
beta0      0.89    0.00 0.10    0.70    0.83    0.90    0.96    1.10  3921
1
beta1      1.95    0.00 0.11    1.74    1.88    1.95    2.02    2.16  3899
1
log_sig   -0.03    0.00 0.07   -0.16   -0.07   -0.03    0.02    0.12  3510
1
lp__    -139.44    0.03 1.22 -142.62 -140.05 -139.12 -138.54 -138.03  2021
1

Samples were drawn using NUTS(diag_e) at Wed Oct 16 13:18:06 2024.
For each parameter, n_eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor on split chains (at
convergence, Rhat=1).
```

In this case, we see that both the posterior median estimates and 95% credible intervals correspond closely to the maximum likelihood versions of this model, which we expect if the likelihood rather than the prior is dominating the solution (Gelman et al. 2017). In the Bayesian paradigm, one would say that a parameter falls within this range with probability 0.95, and the uncertainty of any quantities (i.e., derived quantities, nuisance parameters, random coefficients) are assigned this probabilistic interpretation (McElreath 2018). We end this section by noting that recent advances in Bayesian model validation methods are extensive and far beyond the scope of this chapter. However, analysts interested in Bayesian modeling approaches must follow these validation techniques to ensure reliable inferences from MCMC procedures and to truly understand the role of the prior vs. the likelihood in the inferences that emerge from a Bayesian analysis (see also Monnahan et al. 2017; Gelman et al. 2017; McElreath 2018; Gabry et al. 2019; Monnahan 2024).

References

Auger-Méthé, M., Newman, K., Cole, D., Empacher, F., Gryba, R., King, A.A., Leos-Barajas, V., Mills Flemming, J., Nielsen, A., Petris, G., and Thomas, L. 2021. A guide to statespace

modeling of ecological time series. Ecological Monographs **91**(4): e01470. doi:10.1002/ecm.1470.

Bolker, B.M. 2008. Ecological Models and Data in R. Princeton University Press.

Cahill, C.L., Anderson, S.C., Paul, A.J., MacPherson, L., Sullivan, M.G., Poorten, B. van, Walters, C.J., and Post, J.R. 2020. A spatialtemporal approach to modeling somatic growth across inland recreational fisheries landscapes. Canadian Journal of Fisheries and Aquatic Sciences **77**(11): 1822–1835. doi:10.1139/cjfas-2019-0434.

Carpenter, B., Gelman, A., Hoffman, M.D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M.A., Guo, J., Li, P., and Riddell, A. 2017. Stan: A probabilistic programming language. Journal of statistical software **76**: 1. doi:10.18637/jss.v076.i01.

Chernick, M.R., and LaBudde, R.A. 2014. An Introduction to Bootstrap Methods with Applications to R. John Wiley & Sons.

Cole, S.R., Chu, H., and Greenland, S. 2014. Maximum Likelihood, Profile Likelihood, and Penalized Likelihood: A Primer. American Journal of Epidemiology **179**(2): 252–260. doi:10.1093/aje/kwt245.

Efron, B. 1987. Better bootstrap confidence intervals. Journal of the American Statistical Association **82**(397): 171–185. doi:10.1080/01621459.1987.10478410.

Fournier, D., and Archibald, C.P. 1982. A general theory for analyzing catch at age data. Canadian Journal of Fisheries and Aquatic Sciences **39**(8): 1195–1207. doi:10.1139/f82-157.

Gabry, J., Simpson, D., Vehtari, A., Betancourt, M., and Gelman, A. 2019. Visualization in bayesian workflow. Journal of the Royal Statistical Society Series A: Statistics in Society **182**(2): 389–402. doi:10.1111/rssa.12378.

Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., Vehtari, A., and Rubin, D.B. 2013. Bayesian Data Analysis, Third Edition. CRC Press.

Gelman, A., Simpson, D., and Betancourt, M. 2017. The Prior Can Often Only Be Understood in the Context of the Likelihood. Entropy **19**(10): 555. doi:10.3390/e19100555.

Hilborn, R., and Mangel, M. 1997. The Ecological Detective: Confronting Models with Data. Princeton University Press.

Hilborn, R., and Walters, C.J. 1992. Quantitative Fisheries Stock Assessment: Choice, Dynamics and Uncertainty. Springer Science & Business Media.

Hobbs, N.T., and Hooten, M.B. 2015. Bayesian Models: A Statistical Primer for Ecologists. Princeton University Press. Available from https://www.degruyter.com/document/doi/10.1515/9781400866557/html.

Kristensen, K. 2024. RTMB: 'R' bindings for 'TMB'. Available from https://cran.r-project.org/web/packages/RTMB/index.html.

Kristensen, K., Nielsen, A., Berg, C.W., Skaug, H., and Bell, B. 2016. TMB: Automatic Differentiation and Laplace Approximation. Journal of Statistical Software **70**(5). doi:10.18637/jss.v070.i05.

Lee, Y., Nelder, J.A., and Pawitan, Y. 2017. Generalized linear models with random effects: Unified analysis via h-likelihood, second edition. *In* 2nd edition. Chapman; Hall/CRC, New York.

Lele, S.R. 2020. Consequences of lack of parameterization invariance of non-informative bayesian analysis for wildlife management: Survival of san joaquin kit fox and declines in amphibian populations. Frontiers in Ecology and Evolution **7**. doi:10.3389/fevo.2019.00501.

Magnusson, A., Punt, A.E., and Hilborn, R. 2013. Measuring uncertainty in fisheries stock assessment: the delta method, bootstrap, and MCMC. Fish and Fisheries **14**(3): 325–342. doi:10.1111/j.1467-2979.2012.00473.x.

Mayo, D.G. 1996. Error and the Growth of Experimental Knowledge. University of Chicago Press.

McElreath, R. 2018. Statistical rethinking: A bayesian course with examples in r and stan. Chapman; Hall/CRC. Available from https://www.taylorfrancis.com/books/mono/10.1201/9781315372495/statistical-rethinking-richard-mcelreath.

Monnahan, C.C. 2024. Toward good practices for bayesian data-rich fisheries stock assessments using a modern statistical workflow. Fisheries Research **275**: 107024. doi:10.1016/j.fishres.2024.107024.

Monnahan, C.C., and Kristensen, K. 2018. No-U-turn sampling for fast Bayesian inference in ADMB and TMB: Introducing the adnuts and tmbstan R packages. PLOS ONE **13**(5): e0197954. doi:10.1371/journal.pone.0197954.

Monnahan, C.C., Thorson, J.T., and Branch, T.A. 2017. Faster estimation of Bayesian models in ecology using Hamiltonian Monte Carlo. Methods in Ecology and Evolution **8**(3): 339–348. doi:10.1111/2041-210X.12681.

Nash, J.C. 2014. Nonlinear Parameter Optimization Using R Tools. John Wiley & Sons.

Pawitan, Y. 2001. In All Likelihood: Statistical Modelling and Inference Using Likelihood. OUP Oxford.

Punt, A.E., and Punt, A.E. 1997. Fisheries stock assessment and decision analysis: the Bayesian approach. Reviews in Fish Biology and Fisheries **7**(1): 35–63. doi:10.1023/A:1018419207494.

Quinn, T.J., and Deriso, R.B. 1999. Quantitative Fish Dynamics. Oxford University Press.

Roberts, D.R., Bahn, V., Ciuti, S., Boyce, M.S., Elith, J., Guillera-Arroita, G., Hauenstein, S., Lahoz-Monfort, J.J., Schröder, B., Thuiller, W., Warton, D.I., Wintle, B.A., Hartig, F., and Dormann, C.F. 2017. Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. Ecography **40**(8): 913–929. doi:10.1111/ecog.02881.

Royall, R. 1997. Statistical evidence: A likelihood paradigm. Routledge, New York.

Royle, J.A., and Dorazio, R.M. 2008. Hierarchical Modeling and Inference in Ecology: The Analysis of Data from Populations, Metapopulations and Communities. Elsevier.

Thorson, J., and Kristensen, K. 2024. Spatio-Temporal Models for Ecologists. CRC Press.

Thorson, J.T., and Cope, J.M. 2017. Uniform, uninformed or misinformed?: The lingering challenge of minimally informative priors in data-limited bayesian stock assessments. Fisheries Research **194**: 164–172. doi:10.1016/j.fishres.2017.06.007.