Modelling Complexity and Uncertainty in Fisheries Stock Assessment

by

D'Arcy N. Webber

A thesis
submitted to the Victoria University of Wellington
in fulfilment of the
requirements for the degree of
Doctor of Philosophy
in Statistics.
Victoria University of Wellington
2015

Abstract

Stock assessment models are used to determine the population size of fish stocks. Although stock assessment models are complex, they still make simplifying assumptions. Generally, they treat each species separately, include little, if any, spatial structure, and may not adequately quantify uncertainty. These assumptions can introduce bias and can lead to incorrect inferences. This thesis is about more realistic models and their inference. This realism may be incorporated by explicitly modelling complex processes, or by admitting our uncertainty and modelling it correctly.

We develop an agent-based model that can describe fish populations as a collection of individuals which differ in their growth, maturation, migration, and mortality. The aim of this model is to better capture the richness in natural processes that determine fish abundance and subsequent population response to anthropogenic removals. However, this detail comes at considerable computational cost. A single model run can take many hours, making inference using standard methods impractical. We apply this model to New Zealand snapper (*Pagurus auratus*) in northern New Zealand.

Next, we developed an age-structured state-space model. We suggest that this sophisticated model has the potential to better represent uncertainty in stock assessment. However, it pushes the boundaries of the current practical limits of computing and we admit that its practical application remains limited until the MCMC mixing issues that we encountered can be resolved.

The processes that underpin agent-based models are complex and we may need to seek new sources of data to inform these types of models. To make a start here we derive a state-space model to estimate the path taken by individual fish from the day they are tagged to the day of their recapture.

The model uses environmental information collected using pop-up satellite archival tags. We use tag recorded depth and oceanographic temperature to estimate the location at any given time. We apply this model to Antarctic toothfish (*Dissostichus mawsoni*) in the Ross Sea.

Finally, to reduce the computational burden of agent-based models we use Bayesian emulation. This approach replaces the simulation model with an approximating algorithm called an emulator. The emulator is calibrated using relatively few runs of the original model. A good emulator provides a close approximation to the original model and has significant speed gains. Thus, inferences become tractable.

We have made the first steps towards developing a tractable approach to fisheries modelling in complex settings through the creation of realistic models, and their emulation. With further development, Bayesian emulation could result in the increased ability to consider and evaluate innovative methods in fisheries modelling. Future avenues for application and exploration range from spatial and multi species models, to ecosystem-based models and beyond.

Acknowledgments

Firstly, I would like to thank my supervisors. Richard Arnold, thank you for the opportunity to work with you in developing this thesis and for all of your time and ideas. You have helped me develop an invaluable skill set. Alistair Dunn, thank you for being my go to fisheries science encyclopedia. Your enthusiasm and wealth of fisheries knowledge has provided me a great deal of encouragement. And Shirley Pledger, thank you for all of your guidance and help.

I would like to thank the Ministry for Primary Industries (MPI) and the National Institute of Water and Atmospheric Research Ltd (NIWA) for their support of my post-graduate study through the provision of a post-graduate scholarship in Quantitative Fisheries Science. I would also like to thank the Victoria University of Wellington for the provision of a Victoria Doctoral Scholarship.

Thanks Neville Smith and Ben Sharp for your mentorship and advice. And, thanks to Daniel Fernandez, Roy Costilla, Kath Large, Ash Eaden, Boyd Anderson, Tom Nation, Matt Dunn, Pamela Mace, Richard Ford, Jim Thorson, Jim Ianelli, Paul Breen, Paul Starr, Vivian Haaste, Charles Edwards, Nokome Bentley, Daryl Sykes, Steve Parker, Megan and Christie Webber, and many more for their help and discussions over the past three years.

I would also like to thank my reviewers Nokuthaba Sibanda, Andre Punt and Chris Francis for all of their time and input.

Finally, thank you mum and dad for encouraging me and supporting me in everything I do and guiding me on my way.

Contents

Glossary

$B_{\mathbf{year}}$ is the estimated or predicted biomass in the named year (usually a mid-year biomass). 76, 80

abundance a measure of fish density, numbers or total biomass. i, 1, 7, 17, 36, 37, 51, 54, 55, 75, 132, 138, 171

abundance index a quantitative but relative (i.e. uncalibrated) measure of fish density, numbers or total biomass. An abundance index can be specific to an area or to a segment of the stock (e.g. mature fish), or it can refer to abundance stock-wide. 2, 9, 47, 50, 51

ACE Annual Catch Entitlement. 57, 58

age-frequency (P_a) the proportions of fish of different ages in the stock, or in the catch taken by either the commercial fishery or research fishing. This is often estimated based on a sample. Sometimes called an age composition. 49, 75

age-length key ($M_{a,\ell}^{\text{key}}$) the proportion of fish of each age in each length-group in a catch (or stock) of fish. 49, 50

biomass (B_t) refers to the size of the stock in units of weight. Often, biomass refers to only one part of the stock (e.g. spawning stock biomass, recruited biomass, or vulnerable biomass the later two of which are essentially equivalent). 1, 2, 6, 7, 9, 17, 33–36, 50, 51, 54, 123, 129, 137, 139

bycatch refers to fish species, or size classes of those species, caught unintentionally in association with key target species. 81

C++ is a general-purpose programming language. It has object-oriented programming features, while also providing the facilities for low-level memory manipulation. 93, 123, 125

carrying capacity (K) is the average stock size expected in the absence of fishing. Even without fishing the stock size varies through time in response to stochastic environmental conditions. 7, 33, 35, 137–139, 296

catch (C_t) is the total weight (or sometimes numbers) of fish caught by fishing operations. Sometimes referred to as landings. The catch can be split by age $C_{a,t}$ or by size $C_{\ell,t}$. i, 1–4, 9, 24, 33, 35, 37, 42, 44, 47, 49–51, 54, 58, 75, 83, 84, 86, 118, 122, 130, 132, 138, 139, 141, 148, 163, 171, 172, 179, 180, 293, 296, 300, 305

catch-per-unit-effort (CPUE) (I_t) is the quantity of fish caught with one standard unit of fishing effort; e.g. the number of fish taken per 1000 hooks per day or the weight of fish taken per hour of trawling. CPUE is often assumed to be an abundance index. 9, 36, 47, 50–52, 54, 86, 87, 130, 137, 138, 171, 296, 305

catchability (q) is the proportion of fish that are caught by a defined unit of fishing effort. It is a constant relating an abundance index to the true biomass (the abundance index is approximately equal to the true biomass multiplied by the catchability). 36, 51, 54, 90, 122, 137, 139, 141, 172, 181, 293, 296, 305

CCAMLR Convention for the Conservation of Antarctic Marine Living Resources. 73–76

CELR Catch Effort Landing Return. 86

coefficient of variation (CV) is a standardised measure of dispersion of a probability distribution or frequency distribution. It is defined as the ratio of the standard deviation to the mean. 99, 166

cohort refers to those individuals of a stock born in the same spawning season. For annual spawners, a years recruitment of new individuals to a stock is a single cohort or year-class. 2, 17, 34, 36, 37, 91, 92, 125

CRA red rock lobster (*Jasus edwardsii*), commonly known as crayfish. 80–82, 86

CSIRO Commonwealth Scientific and Industrial Research Organisation. 215

demersal species live and feed on or near the bottom of seas or lakes (the demersal zone). 255

deterministic a deterministic system is a system in which no randomness

is involved in the development of future states of the system. A deterministic model will thus always produce the same output from a given starting condition or initial state. 6, 19, 94, 109, 110, 171, 240, 259
DV Deemed Value. 58

EEZ Exclusive Economic Zone. 56, 83
equilibrium A theoretical condition that arises when the fishing mortality, exploitation patterns and other fishery or stock characteristics (growth, natural mortality, recruitment) are balanced and sustainable and thus do not change from year to year. 33, 35, 107
exploitation rate (U) is the proportion of the recruited or vulnerable biomass that is caught during a certain period, usually a fishing year. 24, 40, 42, 44, 118, 123, 171

fishing mortality (F) is the fishing mortality rate. 3, 11, 24, 42–44, 54, 101, 138
fishing year (t) For most stocks in New Zealand, the fishing year runs from 1 October in one year to 30 September in the next. The second year is usually used as shorthand for the split years. For example, 2005 refers to the fishing year running from 1 October 2004 to 30 September 2005. 3, 52, 57, 214, 244
FishServe FishServe is the trading name of a privately owned company called Commercial Fisheries Services (CFS). CFS is a wholly owned subsidiary of Seafood New Zealand (SNZ). FishServe provides administrative services to the New Zealand commercial fishing industry to support the 1996 Fisheries Act. 57
FMA Fisheries Management Area. 56
Fortran (previously FORTRAN, derived from Formula Translating System) is a general-purpose, imperative programming language that is especially suited to numeric computation and scientific computing. 256

Git is a distributed revision control system for distributed, non-linear workflows. Git was initially designed and developed by Linus Torvalds for Linux kernel development in 2005, and has since become the most widely adopted version control system for software development. 3, 4
GitHub is a web-based Git repository hosting service, which offers all of

the distributed revision control and source code management functionality of Git as well as adding its own features (`https://github.com/`). 16

GPS global positioning system. 214, 215, 217, 218, 234–236, 239, 253

heterogeneous composed of parts of different kinds; having widely dissimilar elements or constituents; not homogeneous. 4, 10, 13, 92, 125, 317

homogeneous composed of parts or elements that are all of the same kind; not heterogeneous. 4, 11, 56, 75

ITQ Individual Transferable Quota. 57

Julia is a high-level, dynamic programming language for technical computing (`http://julialang.org/`). 206, 220, 254, 256, 322

length-frequency (Q_ℓ) the proportions of fish of different lengths in the stock, or in the catch, taken by either the commercial fishery or research fishing. This is often estimated based on a sample. Sometimes called a length or size composition. 49, 50, 75, 86

Linux is a Unix-like computer operating system assembled under the model of free and open-source software development and distribution. The defining component of Linux is the Linux kernel, an operating system kernel first released on 5 October 1991 by Linus Torvalds. 3, 123

maturity refers to the ability of fish to reproduce. i, 1, 10, 17, 18, 21–24, 40, 89, 90, 92, 93, 96, 112, 171, 172

MCMC Markov chain Monte Carlo. 13, 59, 62–69, 133–137, 139, 140, 142, 143, 145–149, 151, 155, 157, 158, 160, 162, 163, 174, 177, 180, 182–184, 191–194, 201–203, 205, 220, 227, 232, 233, 235, 237, 240, 241, 244, 246, 247, 253, 254, 256, 259, 299, 309, 310, 325, 327–340

MHR Monthly Harvest Return. 84, 86

MLS minimum legal size. 79, 83, 86

MPI Ministry for Primary Industries. iii, 56, 58, 83, 86

multithreading is a widespread programming and execution model that allows multiple threads to exist within the context of a single process. These threads share the process's resources, but are able to execute independently. Multithreading can be applied to a single process to enable parallel execution on a multiprocessing system. 95, 125, 220

natural mortality (M) is the natural mortality rate caused by predation and other natural processes and is normally calculated on an annual basis. 3, 22–24, 33, 37, 39, 40, 42–44, 46–48, 90, 93, 96, 101, 107, 108, 117, 130, 138, 163, 171, 172

NIWA National Institute of Water and Atmospheric Research Ltd. iii, 17, 79, 213

otolith one of the small bones in the internal ear of fish that can sometimes be used to determine their age. This involves counting rings in the bone that correspond to annual growing seasons. 19, 49

PHC packhorse rock lobster (*Sagmariasus verreauxi*). 81, 83, 86, 87, 129, 140

photic zone or sunlight zone, extends from the surface down to a depth where light intensity falls to one percent of that at the surface. Accordingly, this depth depends on the extent of light attenuation in the water column. Typical depths vary from only a few centimetres in highly turbid eutrophic lakes, to around 200m in the open ocean. 212

pointer in computer science, a pointer is a programming language object, whose value refers to (or "points to") another value stored elsewhere in the computer memory using its address. A pointer references a location in memory. Obtaining the value stored at that location is known as dereferencing the pointer. As an analogy, a page number in a book's index could be considered a pointer to the corresponding page; dereferencing such a pointer would be done by flipping to the page with the given page number. 94–96, 101, 103

PSAT pop-up satellite archival tag. 75, 76, 209, 212–214, 216, 253, 256

QMA Quota Management Area. 52, 56, 81

QMS Quota Management System. 56–58, 79, 83

R is high-level software environment for statistical computing and graphics (`http://www.r-project.org/`). 62, 220

recruitment (R_t) is the addition of new individuals to the fished component of a stock. This is determined by the size and age at which fish are first able to be caught. 2, 3, 24–27, 33, 35, 37, 42, 47, 90, 107, 114, 115

selectivity (S_x) a curve describing the relative vulnerability of fish of different ages ($x = a$), lengths ($x = \ell$) or weights ($x = w$) to the fishing gear

used. 24, 27, 29–31, 33, 37, 40, 47, 90, 118, 122, 130, 163, 171

selectivity ogive (S_x) see selectivity. 122

SNA snapper (*Pagurus auratus*). 77, 79, 80, 129, 180

spawning stock biomass (SSB_t) refers to the portion of a stocks biomass that is mature. 1, 18, 24, 25, 27, 32, 37, 38, 40, 42, 92, 107, 112, 114, 115, 123, 124

standard deviation (σ) (SD) is a measure that is used to quantify the amount of variation or dispersion of a set of data values. 2, 15, 50, 90, 133, 140–142, 148–155, 157–159, 210, 211, 215, 216, 218, 221, 222, 225, 226, 230–232, 234–237, 240, 241, 245–247, 254, 255, 274–277, 293

stochastic a purely stochastic system is one whose state is randomly determined, having a random probability distribution or pattern that may be analysed statistically but may not be predicted precisely. In this regard, it can be classified as non-deterministic. 109, 111, 137, 259, 278, 281

stock a biological stock is a population of a given species that forms a reproductive unit and spawns little if at all with other units. However, there are many uncertainties in defining spatial and temporal geographical boundaries for such biological units that are compatible with current data collection systems. For this reason, the term stock is often synonymous with an assessment/management unit, even if there is migration or mixing of some components of the assessment/management unit between areas. i, 1–7, 9, 11, 17, 24, 25, 27, 31–33, 35–38, 47, 50, 51, 56–58, 75–77, 80, 86, 90, 99, 100, 107, 124, 129, 163, 209, 300, 319

stock assessment is the application of statistical and mathematical tools to relevant data to obtain a quantitative understanding of the status of a stock relative to defined benchmarks or reference points (e.g. B_{MSY}, F_{MSY}). i, 9–14, 18–21, 23, 27, 31, 35, 36, 47, 51, 54, 55, 60, 75, 76, 86, 87, 91, 92, 125, 127, 129, 163, 166, 171, 204, 206, 209, 212, 319

TAC Total Allowable Catch. 57

TACC Total Allowable Commercial Catch. 57, 58, 79, 83, 84

TOT Antarctic toothfish (*Dissostichus mawsoni*). 73, 209

variance (σ^2) variance measures how far a set of numbers is spread out. 15, 26, 31, 47, 61, 139, 142, 148, 154, 155, 157–159, 165, 168–172, 181, 200, 204, 205, 207, 229, 254, 264, 266, 270, 275, 276, 278, 279, 281, 286, 293, 296,

305, 323, 325

virgin biomass (B_0) is the theoretical carrying capacity of the spawning, recruited or vulnerable biomass of a fish stock. In some cases, it refers to the average biomass of the stock in the years before fishing started. More generally, it is the average biomass that theoretically would have occurred if the stock had never been fished. 25, 27, 76, 114, 123

vulnerable biomass (V_t) refers to the portion of a stocks biomass that is available to the fishery. Also called exploitable biomass or recruited biomass. 1, 3, 7, 38, 40, 42, 50, 54, 118, 122–124, 163

WGS84 The World Geodetic System of 1984. 216

year class (cohort) refers to fish in a stock that were born in the same year. Occasionally, a stock produces a very small or very large year class which can be pivotal in determining stock abundance in later years. 2, 25, 26, 115

Chapter 1

Introduction

Commercial and ecological management of fisheries requires good estimates of stock sizes: that is, the number of individuals or total biomass in a fish population of a particular species. The current practice in fisheries science is to fit a model, on which stock assessment is based, to data. The results of these models are then passed on to managers to make decisions about allowable catches, taking into account the outcome and uncertainty of the model (Harwood & Stokes 2003, Hilborn 2003). Modern stock assessment models are generally large statistical models and often make use of all available data. These data include: estimates of total removals (catch); fishery independent surveys; indices of abundance such as catch per unit effort (CPUE); age and length structure of the catch and/or population; biological parameters; and in special cases may incorporate tagging data, environmental covariates, genetic information, and/or economic data (Anderson 1977).

Despite their complexity, stock assessment models are still simplified versions of reality. They are simplistic in that they tend to lump fish together into broad categories or groups (i.e. all fish within a year are lumped together in biomass dynamics models or all fish of the same age in the same year are lumped together in age-structured models) and treat all of the individuals within each of these groups in the same way. For example, stock assessment models might lump populations together for management convenience, and/or they may treat fish from broad spatial areas

together in the same way. We expand on each of these concepts in the following paragraphs.

Organisms in the marine environment are likely to exhibit variation in behavioral or phenotypic traits among individuals (e.g. growth rates, condition, maturity), and this variation may be persistent (e.g. particular individuals growing faster/slower throughout their entire lifetime) or transient (e.g. particular individuals growing faster in one year than in another year). Many recent studies of captive or wild populations have demonstrated examples of such persistent differences, termed differences in "personality" (Wolf & Weissing 2012). For example, persistent differences in activity level or tolerance of predation risk (i.e. a tendency to forage in high vs. low-quality habitat) will likely lead to persistent differences in growth rates among individuals. Subsequently, persistent differences in growth rate, combined with size-selective harvest targeting larger individuals, can result in older individuals being composed primarily of slow-growing individuals (termed "Rosa Lees Phenomenon"), and has been demonstrated to occur in small-lake mesocosm experiments (Biro & Post 2008).

Individuals are also likely to experience transient variation in natural processes. Such transient variation could be caused by many different processes including movement between warmer/colder ambient temperatures (and hence transient variation in metabolic rates), periodic access to improved feeding (Armstrong & Schindler 2011), and year-specific decisions regarding the allocation of resources between growth and reproduction (Jorgensen & Fiksen 2006). Several recent studies have also demonstrated transient differences in behavioural or phenotypic traits among individuals (Shelton et al. 2013, Webber & Thorson 2015). Failure to account for persistent or transient differences in growth rate can lead to biased estimation of average growth rates in populations and subsequently lead to biases in stock assessment.

Population variability, (such as average growth, maturity, etc) can also be important and Punt (2003*b*) demonstrates that spatial heterogeneity in growth can affect stock assessment outcomes and that better outcomes

can be realised by doing assessments at the population level rather than pooling data across different populations and carrying out assessments on these pooled data.

Despite these known risks, stock assessment models are typically developed for broad scale management and fish stocks are often assumed to be discrete and spatially homogeneous (Stephenson 1999). As a result, many stock assessment models have little or no incorporation of spatial variation or dependence and often stocks are managed over areas that are either smaller or larger than the true (biological) stock area. In reality, fish populations are far from spatially homogeneous (Ralston & O'Farrell 2008). Yet the implementation of spatial structure in stock assessment models has been slow. Although the importance of accounting for spatial population structure in stock assessment is acknowledged (Cadrin & Secor 2009), the impacts of simplifying assumptions are not yet fully understood. Spatial complexity can take many forms, including: gradients of fishing mortality occurring across stock or management boundaries (Siler et al. 1986); or fishing mortality can be much higher in spatially aggregated clumps (Prince 2003); and high fishing mortality in small areas can lead to localised depletion while the stock as a whole may still appear healthy. Furthermore, additional layers of complexity may be introduced when marine protected areas (MPAs) are declared within stock boundaries by displacing fishing effort.

Further complications arise when we consider that often stock assessment models further simplify fisheries systems by largely ignoring ecosystem interactions (e.g. other species or environmental influences).

Finally, stock assessment models should provide estimates with as little bias as possible, have the ability to deal with uncertainty, and the uncertainty in the data should be properly reflected in the estimates produced by the models. Although quantifying uncertainty is an important topic in fisheries, current stock assessment models tend to underestimate the true uncertainty (Magnusson et al. 2013). This can produce biased results (Mormede, Dunn & Hanchet 2013) which can lead to incorrect inferences (Hoshino et al. 2014). One of the key problems we are faced

with is separability of the different variance components (i.e. observation and process error). This is a well known problem in these types of models (Hilborn & Mangel 1997, Schnute 1987). Therefore, models are needed that better represent the uncertainty that is characteristic of fisheries data.

Stock assessment models that do take into account some of these additional complexities in fish populations are at the forefront of fisheries modelling, but often push the limits of available data and computing power. However, before models like this are adopted, and resources are allocated to collecting the data to "feed" them, we must first consider whether or not our current models do an adequate job. If not, then we open a Pandora's box of questions. When does modelling space begin to matter? How can we include individual level variability in our models? Do environmental drivers influence populations more than fishing? Where do we stop?

This thesis is about more realistic models and their inference. We identify four specific research questions below.

1.1 Research questions

The primary research questions considered in this thesis include:

1. **Can we develop models to better capture complexity, including spatial richness and individual variability, inherent in real fish populations?**

2. **Can we do better at modelling uncertainty in "classic" stock assessment?**

3. **More complex stock assessments will require richer sources of data, particularly spatially explicit data. Can we extract some of these data from pop-up satellite archival tag technology?**

4. **Bayesian inference of fisheries models can be slow and we are looking to make them more complex and thus slower. Can we speed up the Bayesian inference of complex fisheries models?**

While the focus of this thesis is stock assessment, no formal stock assessments are done. All applications of the models developed use simulated data, based loosely on case study species (except for Chapter 6). The main contributions in this thesis are found in Chapters 4, 5, 6 and 7. The preceding chapters provide the background material required for these core chapters.

The remainder of this chapter introduces stock assessment modelling and provides a literature review on stock assessment. Section 1.2 introduces some of the mathematical terminology used throughout the thesis. Section 1.3 gives an overview of the components of a stock assessment model. Section 1.4 discusses some of the model structures that have been developed to date. Section 1.5 introduces the types of data used to inform the parameters of stock assessment models. Finally, Section 1.6 briefly introduces the fisheries management system in New Zealand.

Chapter 2 introduces Bayesian inference methods including Metropolis-Hastings Markov chain Monte Carlo (MCMC), parallel tempering and Bayesian emulation. These methods are discussed and used extensively throughout the rest of the thesis.

Chapter 3 introduces three case study species: Antarctic toothfish (*Dissostichus mawsoni*), snapper (*Pagurus auratus*) and packhorse rock lobster (*Sagmariasus verreauxi*). Both snapper and packhorse rock lobster are used in later chapters to develop simulation models. The data from a pop-up satellite archival tag (PSAT) that was attached to an Antarctic toothfish for a year is used to develop a novel state-space model in Chapter 6.

Chapter 4 describes a spatially explicit multi-generational agent-structured fish simulation model that allows flexibility in specifying population and spatial dynamics. The model has the potential to consider individual variability, individual movement, and spatial heterogeneity in the environment. The aim was to construct a model that is sufficiently rich that it can be used to simulate complete, realistic fish populations. The simulated data can be used to test stock assessment methodologies - which are usually based on samples from the population, and incomplete data. The capability of the model is illustrated through an application to

snapper.

Chapter 5 introduces state-space models and their inference. The chapter begins by introducing state-space models using biomass dynamics models as an example (see Section 1.4.1, page 35). While these are nothing new, the age-structured state-space model described next is novel. This age-structured state-space model includes process error in the mid-year numbers at age in the modelled population, attempting to better capture model uncertainty in stock assessment. The posterior distribution of this model is developed, and sophisticated MCMC methods are implemented in an attempt to sample from the posterior distribution. However, efficient sampling proved difficult for this model.

In Chapter 6, a novel state-space model for estimating the path taken by individual fish is developed making use of PSAT data. The difference between this model and other models is that this model is conditional on the start and end location. In contrast, other models may not fix the end point (often called diffusion models or random walks). Our model also attempts to use depth and temperature, rather than the standard light and temperature, to estimate the location of the fish at any given time. The model is applied to data collected from a PSAT attached to an Antarctic toothfish in the Ross Sea region. Although this application was not entirely successful, the model shows promise for future applications to this and other species.

Chapter 7 explores Bayesian emulation in detail and applies the method in a series of examples - starting with simple deterministic univariate examples, up to stochastic fisheries models nested within a state-space framework.

Finally, the thesis concludes with a discussion in Chapter 8.

We provide a list of original contributions to fisheries science and statistics that this thesis makes here to aid the reader as they progress through the document:

- A comprehensive agent-based simulation model (Chapter 4)
- Further development of age-structured state-space models (Chapter 5)

- A new process model for modelling the dynamics of fish tagged using pop-up satellite archival tags coupled with a new observation model for geolocating fish using depth/bathymetric data (Chapter 6)
- Bayesian emulators nested within a state-space framework (Chapter 7)
- Further development of stochastic Bayesian emulators and a proof of concept applied to a spatially explicit agent based model (Chapter 7).

1.2 Notation, terminology and layout

This section describes some of the mathematical terminology used throughout this thesis.

Generally a bold capital symbol $\mathbf{A}$ refers to a matrix, a bold lowercase symbol $\mathbf{a}$ to a vector and an unbolded italic symbol a to a scalar. $\{a_i\}_{i=1}^n$ is an ordered n-tuple. θ and $\boldsymbol{\theta}$ are generally used to represent a parameter or parameter vector, respectively. Data are usually denoted y. $\mathbb{R}$ is a real number.

The expected value of a random variable a is $\mathbb{E}[a]$, while $\mathbb{V}[a]$ is the variance of a, and $\mathbb{C}[a]$ is the covariance of a. The terms $\pi(\cdot)$, $p(\cdot)$ or $P(\cdot)$ represent probability distributions. iid is short for independent and identically distributed. $a|b$ means event a conditional on event b having occurred. The symbol $\forall$ means for all values, usually referring to all of the values within an ordered tuple.

Throughout we use σ for standard deviation and σ^2 for variance. A normal distribution with mean μ and variance σ^2 is written $\mathcal{N}(\mu, \sigma^2)$. We use $\mathcal{U}$ to represent a uniform distribution, $\mathcal{IG}$ inverse gamma, $\log\mathcal{N}$ log-normal, $\mathcal{B}$in binomial, and $\mathcal{G}$a gamma. Other distributions are defined as they are used. The random variable ε is usually used to represent an error term.

It is common to see log-normal errors applied in fisheries science (typically used for innovations). For example, if we have a random variable α that is assumed to be log-normally distributed with standard deviation σ we can

write

$$\alpha_t = \alpha_{t-1} e^{\eta} \quad \text{where} \quad \eta \sim \mathcal{N}\left(-\sigma^2/2, \sigma^2\right),$$

or

$$\alpha_t = \alpha_{t-1} e^{\varepsilon - \sigma^2/2} \quad \text{where} \quad \varepsilon \sim \mathcal{N}\left(0, \sigma^2\right),$$

noticing the $-\sigma^2/2$ adjustment. In both cases $\log(\alpha_t) = \log(\alpha_{t-1}) + \eta$, but $\mathbb{E}[\alpha_t|\alpha_{t-1}] = \alpha_{t-1}$ only if $\eta \sim \mathcal{N}(-\sigma^2/2, \sigma^2)$ (see Appendix A.2, page 323). Without this adjustment the expected value of a random variable α will tend to increase. However, if σ is small, say $\sigma = 0.01$, then the effect can be negligible (i.e. $e^{-0.01^2/2} = 0.99995$) so sometimes the adjustment is omitted in the literature.

Throughout this document we use boxes, like the one below, to develop ideas alongside the text.

> Boxes like this are used to illustrate ideas or concepts alongside the text.

If you are viewing this document electronically as a pdf then it is useful to know that all references to acronyms, appendices, chapters, cited literature, contents entries, equations, figures, pages, sections, and tables are hyperlinked (i.e. clicking on the reference will take you to the relevant part of this document). Any web-sites referred to in the text are also hyperlinked and clicking the link will open the web page in your default web browser.

A glossary of acronyms, technical terms and commonly used fisheries parameters is also provided. The words contained in the glossary are also hyperlinked throughout this document. If the reader is ever unsure of a word then clicking the word will take the reader to the relevant glossary entry (if that word is in the glossary, this can be checked by hovering a mouse cursor over the word and if the cursor changes then that word is in the glossary). The glossary lists hyperlinked page numbers to all of the pages containing these words.

This thesis is fully version controlled on GitHub (`https://github.com/quantifish/PhD`).

1.3 Components of a fisheries model

If B_t is the total biomass (e.g. tonnes) of a fish population at time t, N_t is the total number of individuals within that population at time t, and w_t is the mean weight (tonnes) of a single fish at time t, then

$$B_t = N_t w_t \qquad (t = 0, \ldots, T).$$

A fisheries model provides a mathematical or statistical description of the way B_t and N_t evolve over time, and specifically, how that population responds to anthropogenic removals (exploitation)[1]. Often these models are tailored to suit both the population in question and the data that are available. The components of these models, the way that these models can be structured, and the types of data that are used to inform them follows. Many of the equations in this section are drawn from Bull et al. (2012).

1.3.1 The partition

Often in models of a fish population we specify $N_{a,t}$ individuals of age a in year t. We can in general represent counts of fish in any given year as a matrix of numbers of fish where the columns are either age-classes (often referred to as cohorts) or size-classes, and the rows are defined by some categorisation, such as sex, maturity, area, stock, or tag year. We call this matrix the "partition" (e.g. Table 1.1). In a model, the partition is updated over time by applying suitable transformations. For example, to age the fish after one year we simply move all of the fish in the partition one cell to the right

$$N_{a,t} = N_{a-1,t-1} \qquad (a = a_{\text{min}}, \ldots, A), \tag{1.1}$$

and absorb all of the oldest fish into a single final age group A

$$N_{A,t} = N_{A-1,t-1} + N_{A,t-1}. \tag{1.2}$$

This is how the numbers of fish in stock assessment models have been viewed conceptually for some time, but R.I.C.C. Francis (NIWA) coined

[1]Not all fisheries models express the abundance of fish in a population as a biomass; some simply track the numbers of fish through time.

Table 1.1: An example in which the partition ($N_{a,m,s}$) is a matrix filled with numbers of fish at age a, up to a maximum age A, for different categories. Here the categories structure the population by sex s (female, male) and maturity m (immature, mature).

		Age (a)				
		1	2	...	$A-1$	A
Immature	Female			...		
Immature	Male			...		
Mature	Female			...		
Mature	Male			...		

the term partition to describe the population matrix. Consequently, the use of the term is mainly limited to New Zealand literature.

The partition can be used in combination with other information to derive properties of the population. For instance, the mean weight of each individual in the partition at age a is w_a, so by summing the product of numbers-at-age and weight-at-age we can derive the total biomass of the population

$$B_t = \sum_a N_{a,t} w_a. \tag{1.3}$$

Sex in the partition

Sex is included in the partition so that processes within the stock assessment can be sex-specific. For instance, different growth models may be applied to females and males, or the spawning stock biomass might be calculated as the biomass of mature females only. If sex is not included in the partition (i.e. we choose a single sex model) we are effectively assuming that the fish are reproducing asexually. This is often a reasonable approximation to the truth, particularly if there is little difference between the sexes in their sizes and spatial distribution.

1.3.2 Length and weight at age

Length and weight data from the commercial catch are perhaps the most commonly available data in fisheries, because they are the cheapest and easiest to collect. When analysed in conjunction with age data, one can construct growth curves that inform us of the mean length or weight of a fish at a given age. In many fish species, the age can be determined by examining the otolith and counting the rings (much like the growth rings of a tree). The ability to easily age a fish often determines whether an age-structured or length-based model will be used (see Section 1.4, page 34 for details on these different model structures). The relationship between length or weight and age is often described deterministically using a growth model. The purpose of these growth models is to provide an estimate of the expected length of a fish for a given age. Perhaps the most commonly used growth model is that proposed by von Bertalanffy (1934)

$$L_a = L_\infty \left(1 - e^{-k(a-t_0)}\right), \tag{1.4}$$

where L_a is the mean length (cm) of an individual in the population at age a, L_∞ is the asymptotic length or the mean length of very old organisms (cm), k is the Brody growth coefficient (a curvature parameter that describes how fast the organism approaches L_∞ with units years^{-1}), and t_0 is the growth intercept (the hypothetical age at which the organism has zero length with units years, e.g. Figure 1.1). The parameter t_0 need not be 0 since fish are only caught at ages > 0, and a good fit of this simple model may require $t_0 \neq 0$ without ever predicting a fish of negative length within a stock assessment model. These parameters are typically expressed in centimetres (cm) and years, but may take other units. When fitting this model, we estimate the parameters L_∞, k and t_0 given a sample of fish of known age a and length L_a.

However, this growth function may not be the best curve to describe the growth of a particular species. There are many other growth models that may be more suitable (e.g. Richards 1959, Gompertz, logistic, exponential). Moreover, such growth curves model the mean and variability of growth in the population, but not the trajectories of individuals.

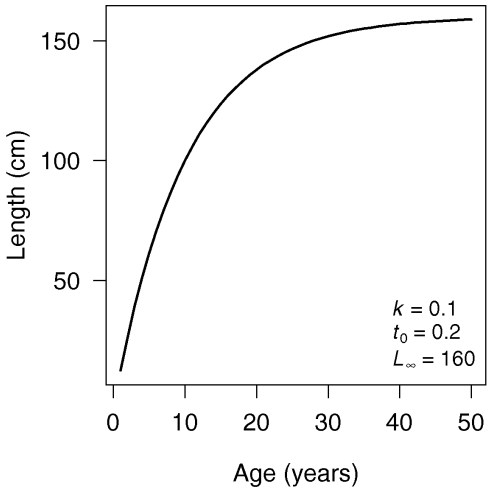


Figure 1.1: Length-at-age example plot using the von Bertalanffy growth model (Equation 1.4). The parameter values for this model are given in the bottom right of the figure.

In most stock assessment models it is necessary to determine the weight-at-age of individuals in the population. We can do this using a power law length-weight relationship

$$w_a = \alpha L_a^{\beta}. \tag{1.5}$$

The exponent β is close to 3.0 for most species while the coefficient α varies between species. If the exponent β is greater than three for a certain fish species, that species tends to become relatively fatter or have more girth as it grows longer. If the exponent β is less than three, the species tends to be more streamlined. Alternatively, the length-weight relationship may be combined with the von Bertalanffy relationship (Equation 1.4),

$$w_a = w_\infty \left(1 - e^{-k(a-t_0)}\right)^{\beta}, \tag{1.6}$$

where w_a is the mean weight (tonnes) of a fish at age a, α and β are parameters of the length-weight relationship, w_∞ is the asymptotic weight (tonnes) of fish in the population which may be found using $w_\infty = \alpha L_\infty^{\beta}$, k is the Brody growth coefficient (years^{-1}), and t_0 is the growth intercept (years) (e.g. Figure 1.2).

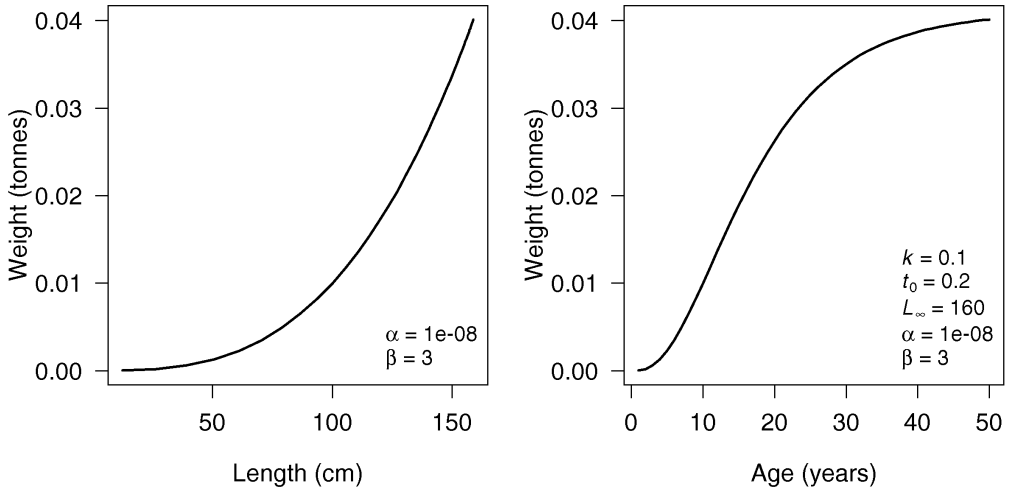


Figure 1.2: Examples of the length-weight relationship (left, Equation 1.5) and weight-at-age (right, Equation 1.6). The parameter values used are given in the bottom right of each figure.

1.3.3 Maturation

Maturation is the process in which immature fish become sexually mature and therefore able to reproduce. This process can be modelled in one of two ways in a stock assessment model, by including maturity in the partition, or not including maturity in the partition. If maturity is included in the partition, then as fish become mature they can be moved accordingly within the partition. One function that may be used to achieve this is the logistic-producing ogive

$$\eta_a = \begin{cases} 0 & \text{if } a < L \\ \lambda(L) & \text{if } a = L \\ (\lambda(a) - \lambda(a-1))/(1 - \lambda(a-1)) & \text{if } L < a < H \\ 1 & \text{if } a \geq H \end{cases}, \tag{1.7}$$

where

$$\lambda(a) = 1 \Big/ \left(1 + 19^{(A_{50}-a)/A_{to95}}\right),$$

where η_a represents the proportion of immature fish at age a maturing each year ($0 \leq \eta_a \leq 1$) and not the proportion of fish mature in that year. The logistic-producing ogive has the parameters L, H, A_{50} and A_{to95}. L and H are used to define the age at which no fish are mature and the age at which all fish are mature, respectively. A_{50} and A_{to95} describe the age at which 50% of individuals within the population are mature and the difference in age at which 95% of individuals within the population are mature, respectively.

When maturity is not explicitly included in the partition a parametric distribution can be used to describe the proportion of mature fish in each age- or size-class at any given time. This approach assumes that the proportion of mature fish remains constant over time. A logistic ogive is commonly used to model maturity when maturity is not included in the partition

$$m_a = 1 \Big/ \left(1 + 19^{(A_{50}-a)/A_{to95}}\right) \qquad (0 \leq m_a \leq 1), \tag{1.8}$$

where m_a is the proportion of individuals at age a that are sexually mature, and A_{50} and A_{to95} are parameters describing the age at which 50% of individuals within the population are mature and the difference in age at which 95% of individuals within the population are mature, respectively (e.g. Figure 1.3).

Note that Equation 1.8 can be written

$$m_a = 1 \Big/ \left(1 + \exp\left(-\left(\frac{a - A_{50}}{A_{to95}}\right)\log(19)\right)\right),$$

$$\text{logit}\,(m_a) = \left(\frac{a - A_{50}}{A_{to95}}\right)\log(19),$$

$$\text{where logit}(p) \equiv \log\left(\frac{p}{1-p}\right).$$

1.3.4 Natural mortality

Natural mortality is the death of fish due to causes not associated with fishing (e.g. cannibalism, competition, disease, old age, predation). Nat-

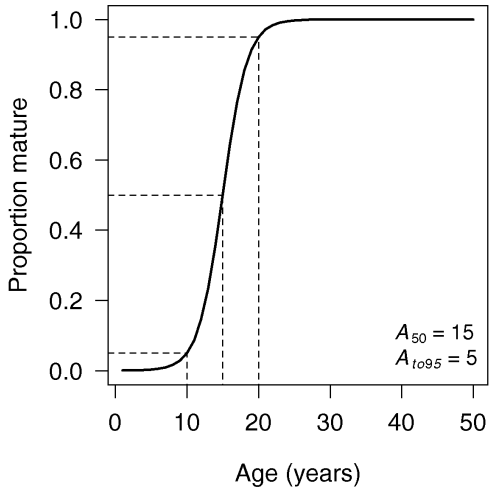


Figure 1.3: Example of the logistic ogive used to model maturity (Equation 1.8). The parameter values used are given in the bottom right of the figure. The dashed lines represent the ages at which 5%, 50% and 95% of the population are mature.

ural mortality is usually applied as an instantaneous rate and the proportion surviving a time interval of τ years[2] is $e^{-\tau M}$, where M is the natural mortality rate[3]. It is often assumed that M is independent of age and year (an exception to this can be found in the New Zealand hoki stock assessment where M is modelled as being age-dependent M_a, see Francis 2006).

Natural mortality is one of the most difficult parameters to estimate, particularly when we have only a short time series of observations, and is a crucial element of a stock assessment (Hewitt et al. 2007). In New Zealand M is usually fixed and assumed known within stock assessment models (i.e. not estimated). However, in countries with much longer fishery time-series (e.g. the US) M is sometimes estimated within stock assessment models.

[2]In most models we would specify $0 \leq \tau \leq 1$ because mortality is calculated from year to year. However, we could potentially develop a model that tracks numbers in multiple year time-steps.

[3]It is also interesting to note that the mean age of fish in a model is $\frac{1}{M}$.

1.3.5 Fishing mortality

Fishing mortality is the removal of fish from a population due to fishing activities using fishing gear (fishing activities could be commercial, recreational, customary or illegal). The catch biomass (C_t) during year t is directly removed from the population in biomass dynamics models (see Section 1.4.1, page 35) or used to derive an exploitation rate (U_t) or a fishing mortality (F_t) in statistical catch at age models (see Section 1.4.2, page 37). Age or length dependent selectivity functions can be used in these models to remove the relative proportion of these fish from the age or length partitions in the model (see Section 1.3.9, page 27).

1.3.6 Spawning stock biomass (SSB)

The spawning stock biomass (SSB_t) is the biomass (tonnes) of mature fish within a fish stock at time t and can be determined by

$$SSB_t = \sum_a N_{a,t} w_a m_a e^{-\tau M}, \tag{1.9}$$

where $N_{a,t}$ is the number of fish of age a at time t, w_a is the mean weight (tonnes) of a fish at age a, m_a is the proportion of fish mature at age a, M is the natural mortality rate, and τ is used to set the time of year we wish to calculate SSB_t. By convention, the SSB is often assumed to be a mid-season estimate, and hence we set $\tau = 0.5$ on the assumption that natural mortality occurs at a constant rate throughout the year. This would give us the mid-year SSB by applying half of the natural mortality ($e^{-0.5M}$) before calculating SSB_t.

1.3.7 Recruitment

Recruitment is the addition of new individuals to the fished component of a stock. Fish usually recruit to the fished (vulnerable) component of a stock when they reach a size, or age, sufficient to be caught by the fishing gear used. In an age-based model, all recruiting fish are of some specified minimum age (a_{min}). This minimum age is usually the lower of some

approximate age corresponding to a minimum catchable size and the age for which the analyst wishes to report results for (and is almost always assumed to be age 1).

The number of fish that recruit to a stock at time t is usually assumed to be dependent on some underlying average annual recruitment in the unfished population R_0 (numbers of fish, see Section 1.3.8, page 27), the SSB (biomass rather than numbers) of fish during time $t-1$, and the strength (a multiplier) of the year class (cohort) during time t. The recruitment R_t (number of fish) at time t can be written

$$R_t = R_0 \times SR(SSB_{t-t_e}) \times YCS_t, \tag{1.10}$$

where t_e is the number of time-steps after spawning that a year class (cohort) enters the fished component of a stock and can incorporate egg development time as well as the time taken for the fish to reach recruitment age a_{min} (usually $t_e = 1$), $SR(\cdot)$ is a function of the SSB, and is a measure of the productivity of a spawning biomass of the stock, $SR(SSB_{t-t_e})$ is the stock recruitment value at time $t-t_e$, and YCS_t is the relative year class strength (YCS) at time t. This allows environmental/ecological variations to lead to higher or lower numbers of recruits in a given year.

The stock recruitment function $SR(\cdot)$ is used to model the relationship between SSB and the number of recruits (R_t) entering the fishery each year. The stock recruitment function scales R_0, the stock's expected average recruitment if there had been no anthropogenic mortality. The two most commonly used stock recruitment functions are those developed by Beverton & Holt (1957) and Ricker (1954). The Beverton-Holt function is

$$SR(SSB_t) = \frac{SSB_t}{B_0} \Bigg/ \left(1 - \frac{5h-1}{4h}\left(1 - \frac{SSB_t}{B_0}\right)\right), \tag{1.11}$$

and the Ricker function is

$$SR(SSB_t) = \frac{SSB_t}{B_0}\left(\left(\frac{1}{5h}\right)^{\frac{5}{4}\left(\frac{SSB_t}{B_0}-1\right)}\right), \tag{1.12}$$

where $SR(SSB_t)$ is the stock recruitment value at time t ($0 \leq SR(SSB_t)$), SSB_t is the SSB (tonnes) at time t, B_0 is the initial biomass (tonnes), and h is

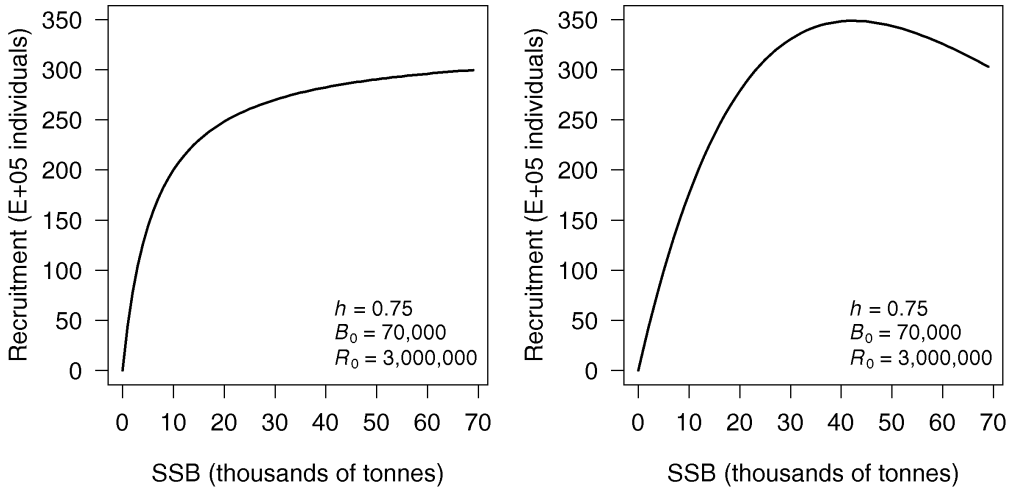


Figure 1.4: Examples of Beverton-Holt (left, Equation 1.11) and Ricker (right, Equation 1.12) recruitment. The parameters used are given in the bottom right of each figure.

the steepness parameter defined as $h = SR(0.2B_0)$ (e.g. Figure 1.4, Mangel et al. 2010).

Although recruitment could be defined using just R_0 and the stock recruitment relationship $SR(SSB_t)$ (i.e. $R_t = R_0 \times SR(SSB_{t-t_e})$), we know that the true numbers of fish recruiting to a stock each year in the absence of fishing would vary about R_0 (e.g. see Section 1.3.8 below). To account for this we use YCS multipliers (also known as recruitment multipliers). The YCS multipliers allow the recruitment to vary between year's within the model while maintaining the definition of R_0 (i.e. the number of recruits that would be observed, on average, in the absence of fishing, e.g. see Figure 1.5). The year class (cohort) strengths can be defined as

$$YCS_t = e^{\varepsilon_t^R - \sigma_R^2/2} \quad \text{where} \quad \varepsilon_t^R \sim \mathcal{N}\left(0, \sigma_R^2\right), \tag{1.13}$$

and σ_R^2 is the recruitment variance.

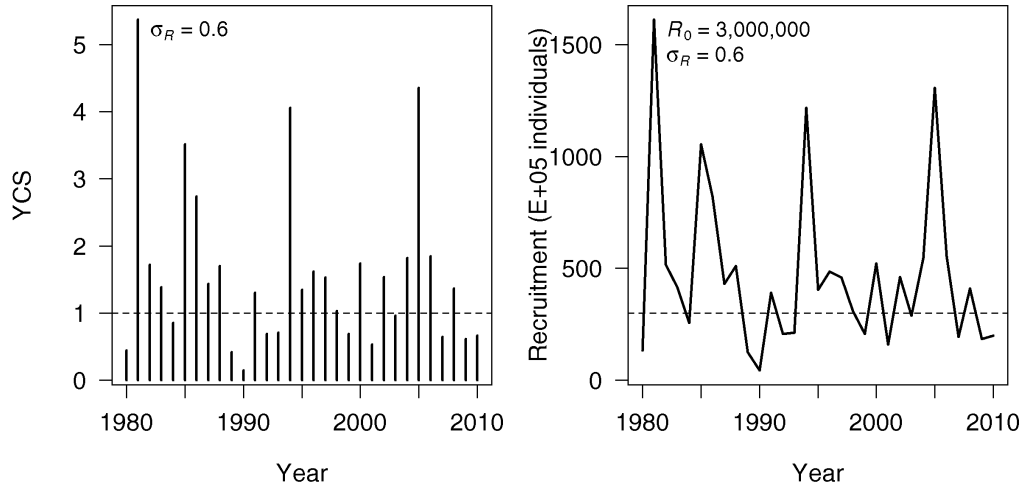


Figure 1.5: Example of year class strength multipliers YCS_t (left, Equation 1.13) and recruitment R_t (right, Equation 1.10). The parameters used are given in the top left of each figure. In the recruitment plot (right) the YCS_t from the left plot is used and $SR(SSB_t) = 1\ \forall t$.

1.3.8 R_0 and B_0

The stock's average unfished recruitment R_0 (numbers) and average biomass B_0 (tonnes) are perhaps the two most important parameters in stock assessment models. R_0 is the underlying parameter that determines how large a stock would be, on average, if there had been no fishing. From R_0 one can calculate B_0, which is defined as the SSB that would exist if recruitment were equal to R_0 every year and there was no fishing (see Equation 1.20). It is often assumed (e.g. for the purposes of initialising simulations) that $B_0 = SSB_{t=1}$. Figure 1.6 illustrates the definition of B_0.

1.3.9 Selectivity

Selectivity refers to the relative vulnerability of fish of different ages or sizes to the fishing gear used. Although we often assume that selectivity is independent of time, this is likely to be an unrealistic assumption as

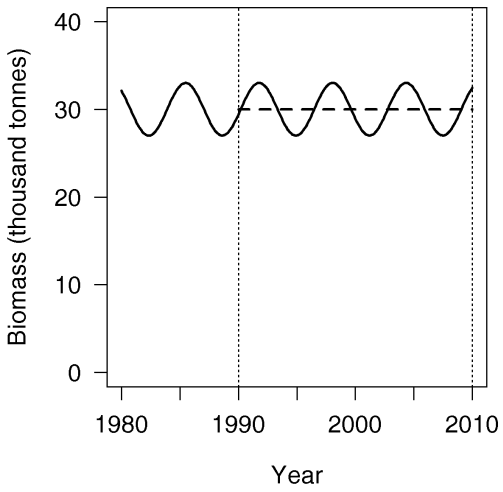


Figure 1.6: The definition of B_0. The solid line represents the actual biomass of a fish population that would occur in the absence of fishing. This example illustrates a population with a cyclic biomass, however the biomass of a real population may take any form. The horizontal dashed line represents B_0 or the mean biomass that would exist in the absence of fishing. The two vertical dotted lines represent the time period of fishing or the set of years S for which we wish to define R_0.

fishing gear can improve over time or the fleet can fish in different areas and have differing degrees of success. Below is an example of selectivity parameterised using a double-normal ogive

$$S_a = \begin{cases} 2^{-[(a-\gamma_1)/\gamma_L]^2} & \text{if } a \leq \gamma_1 \\ 2^{-[(a-\gamma_1)/\gamma_R]^2} & \text{if } a > \gamma_1 \end{cases}, \tag{1.14}$$

where S_a is the selectivity (proportion of fish vulnerable) of fish in the population at age a, γ_1 is the mode, γ_L describes the shape of the left hand limb, and γ_R describes the shape of the right hand limb (e.g. Figure 1.7). The scale of selectivity is arbitrary and S_a is scaled to have a maximum value of 1.

Note that Equation 1.14 can be written

$$S_a = \begin{cases} e^{-\left(\frac{a-\gamma_1}{\gamma_L}\right)^2 \log(2)} & \text{if } a \leq \gamma_1 \\ e^{-\left(\frac{a-\gamma_1}{\gamma_R}\right)^2 \log(2)} & \text{if } a > \gamma_1 \end{cases}.$$

The double-normal ogive is useful as it allows the model to specify the parameters γ_1, γ_L, and γ_R, in such a way that selectivity can approximate a logistic curve or have a declining right hand limb (dome-shaped). "Logistic" type selectivities suggests that younger/smaller fish are less vulnerable to the fishing gear used in the fishery than the older/larger fish. This could be because the younger fish are small enough to fit through the mesh in a trawl net and escape, too small to take the bait on a longline, or may live elsewhere. A dome-shaped selectivity suggests that younger fish are less vulnerable than middle aged fish, and that the vulnerability of fish decreases again as they grow older. This may occur if the oldest fish in the population are alive, but are not vulnerable to the fishery for some reason (e.g. they live somewhere else, they are large enough to outrun a trawl net, or are big enough to simply pull the hook off a longline).

Selectivity may also be parameterised in a way that allows it to vary over time (e.g. Butterworth et al. 2003, Ianelli et al. 2013, Nielsen & Berg 2014). However, estimating the selectivity of each age group every year as individual parameters could potentially result in hundreds of selectivity pa-

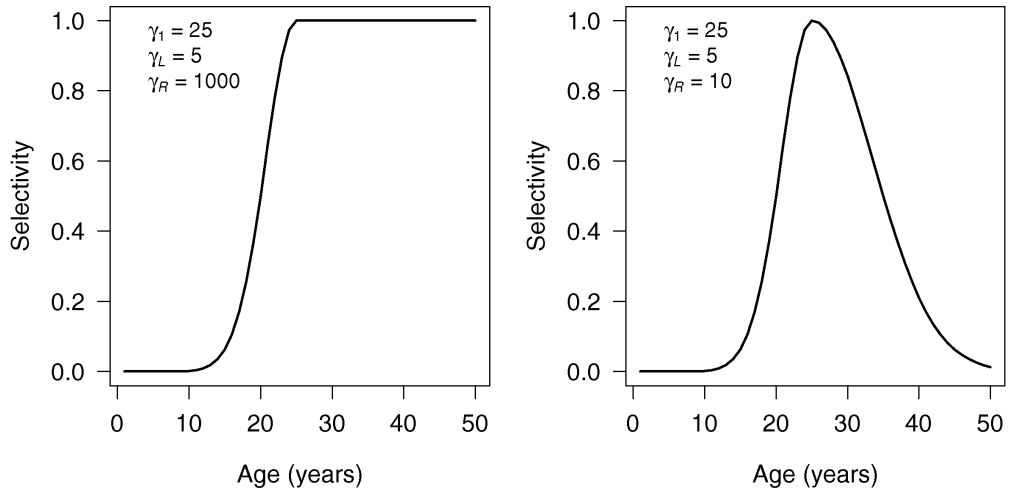


Figure 1.7: Two examples of selectivity modelled using a double-normal ogive (Equation 1.14). The figures show examples of selectivity, increasing to a plateau and right hand limb descending (dome-shaped selectivity) on the left and right respectively. The parameters used are given in the top left of each figure.

rameters. Ianelli et al. (2013) estimated selectivity in this way for the Eastern Bering Sea walleye pollock stock assessment but restricted the amount of change from year to year (in each age) by treating the selectivity in each age group as a random walk[4]

$$S_{a,t} = S_{a,t-1}e^{\gamma_t - \sigma^2/2} \quad \text{where} \quad \gamma_t \sim \mathcal{N}(0, \sigma_S^2). \tag{1.15}$$

Treating the temporal component of selectivity as a random walk in this way could allow selectivity to "creep" or change in some structured way over time. However, the values allowed for the variance hyperparameter σ_S^2 would need to be considered carefully to avoid overfitting of the model. Alternatively, the random-walk or random-effect concept could be imposed on the parameters of a selectivity ogive. For example, Ianelli et al. (2013) implemented the logistic ogive as a random walk using

$$\begin{aligned} S_{a,t} &= \left(1 + e^{-a\alpha_t - \beta_t}\right)^{-1}, \\ \alpha_t &= \bar{\alpha}e^{\delta_t^\alpha} \quad \text{where} \quad \delta_t^\alpha - \delta_{t-1}^\alpha \sim \mathcal{N}(0, \sigma_{\delta^\alpha}^2), \\ \beta_t &= \bar{\beta}e^{\delta_t^\beta} \quad \text{where} \quad \delta_t^\beta - \delta_{t-1}^\beta \sim \mathcal{N}(0, \sigma_{\delta^\beta}^2). \end{aligned}$$

In any case, care should be taken if implementing time-varying selectivity and selectivity models should be evaluated carefully (Maunder & Harley 2011, Punt et al. 2014).

1.3.10 Areas, stocks and fisheries

Several terms are used in fisheries science that can be confusing, especially the terms "area", "stock", and "fishery". So far only single-area, single-stock models have been discussed. However, there are many other ways to structure a stock assessment model. For example, we can have single-area single-stock multi-fishery models (e.g. Antarctic toothfish, Mormede & Dunn 2014), or multi-area multi-stock multi-fishery models (e.g. snapper in SNA 1, see Francis & McKenzie 2013). Here we introduce the concepts of areas, stocks and fisheries.

[4]Actually, the equation presented in Ianelli et al. (2013) was $S_{a,t} = S_{a,t-1}e^{\gamma_t}$ but we have added in the $-\sigma^2/2$ for consistency.

Areas

Areas $z \in Z$ define non overlapping geographic regions of a population. When multiple areas are defined within a model then the location or movement of fish between areas must be specified. Location is usually used for sessile organisms (i.e. animals that don't move like paua), but do recruit to different areas within the model. Movement is typically modelled as a migration, usually using a $z \times z$ matrix for each partition that specifies the proportion of the numbers of fish within each area $N_{a,z}$ that move in a given time step. The migration matrix will often differ between ages and sex (i.e. a $z \times z$ matrix for each age or sex).

Stocks

A stock is a biological production unit in which the effects of emigration and immigration can be considered negligible. However, in fisheries models, this definition is often blurred. Stocks $j \in J$ define the different spawning sub-populations within a fishery. In a multi-area, single-stock model the spawning stock biomass of the stock is simply determined by summing the biomass of all mature fish in all areas

$$SSB_t = \sum_{a,z} N_{a,t,z} w_a m_a \tag{1.16}$$

where $N_{a,t,z}$ is the number of fish of age a at time t in area z. In a multi-area, multi-stock model there are a number of ways that the spawning stock biomass could be derived. The first way is to sum the biomass of mature fish in each area to obtain a spawning stock biomass for each of these areas

$$SSB_{t,z} = \sum_{a} N_{a,t,z} w_a m_a. \tag{1.17}$$

Alternatively, fish from different stocks may contribute towards reproduction only if they are present in their stock area at the time of spawning. In this case it is necessary to introduce the subscript j to indicate which stock the fish came from

$$SSB_{t,j} = \sum_{a} N_{a,j,t,z} w_a m_a. \tag{1.18}$$

Fisheries

A fishery $f \in F$ is an action applied to a fish stock or area. The fishery removes biomass from the stock as catch. We may have multiple fisheries in a single area, each fishery having a distinct selectivity and operating on a particular set of fish species. Stock assessments often represent spatial structure through a "fisheries-as-areas" approach whereby multiple fisheries, each with different selectivity patterns, are used as a proxy for spatial availability.

1.3.11 Simulation

To run a statistical catch-at-age model (see Section 1.4.2, page 37), we specify the minimum age (a_{min}) of the fish to be modelled, the final age (A) if a plus group is to be used, the average unfished recruitment in numbers of fish (R_0) and the natural mortality rate (M). In the absence of fishing we assume that there is some (equilibrium) carrying capacity (B_0). We evaluate B_0 numerically by running a simulation of the model over a long period until it reaches equilibrium. This allows us to determine the initial age structure (N_a^0) of the population before fishing. We call this initial age structure the initial state and the process of solving for the initial state is called initialisation. The initial state can be found by solving for N_a^0

$$N_a^0 = \begin{cases} R_0 e^{-(a-1)M} & \text{if } a = a_{\text{min}}, \ldots, A \\ \sum_{a=A}^{\infty} R_0 e^{-(a-1)M} & \text{if } a = A \end{cases}.$$

This equation actually has an analytic form that is found using geometric series

$$N_a^0 = \begin{cases} R_0 e^{-(a-1)M} & \text{if } a = a_{\text{min}}, \ldots, A \\ R_0 \frac{e^{M-AM}}{1-e^{-M}} & \text{if } a = A \end{cases}, \tag{1.19}$$

see Appendix B.1 (page 327) for the proof. From this B_0 is calculated mid-year as

$$B_0 = \sum_a N_a^0 w_a m_a e^{-0.5M}. \tag{1.20}$$

where w_a is the mean weight at age and m_a is the maturity at age. Figure 1.8 shows an initial age structure, with a plus group at 50 years of age.

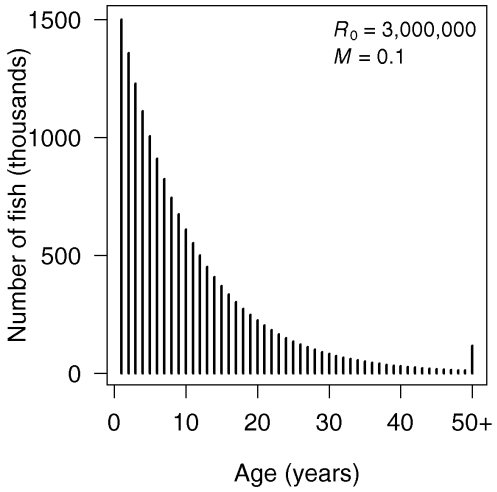


Figure 1.8: Example of the initial state or initial age structure of a population (Equation 1.19). The $50+$ refers to the use of a plus group (i.e. the sum of fish aged 50 and above) and is the reason why more fish are found in this age group than some of the earlier age groups. The parameters used are given in the top right of the figure.

1.4 Types of model

Two modelling structures are commonly employed for assessing the dynamic response of fish populations to exploitation; biomass dynamic models, and age- or length-structured models. However, in the last few years individual-based models have increased in popularity. Biomass dynamics models, age- or length-structured models, and individual-based models represent a spectrum of how we group or bin fish in our models. These models range from lumping the total biomass of fish within a given year together in biomass dynamic models, to modelling the age structure of co-

horts in age-structured models or groups of fish of similar length in length-structured models, to accounting for each individual fish within a population in individual-based models. However, it is entirely possible for the modeller to bin fish together anywhere along this spectrum, for example, age-structured models are somewhere between biomass dynamics models and individual-based models because they bin fish together according to their age.

1.4.1 Biomass dynamics models

Biomass dynamic models, also known as production models or surplus production models, are the simplest stock assessment models that are commonly used. These models consider the net effects of recruitment, growth, and mortality on the entire population each year. The Schaefer surplus production model (Schaefer 1954) is perhaps the most widely used biomass production model and can be written

$$\begin{aligned} B_t &= B_{t-1} + g\left(B_{t-1}\right) - C_{t-1}, \\ g\left(B_t\right) &= rB_t\left(1 - \frac{B_t}{K}\right), \end{aligned} \tag{1.21}$$

where B_t is the biomass of the stock (tonnes) at time t, $g(B_t)$ is a function of biomass known as the surplus production (tonnes) at time t, C_t is the catch (tonnes) at time t, r is an intrinsic rate of population growth (unitless), and K is the carrying capacity (tonnes), a parameter which corresponds to the unfished equilibrium stock biomass.

In biomass dynamics models (and age-structured models) it is usually assumed that the catch (C_t) is proportional to the stock biomass (B_t) and fishing effort (E_t)[5]

$$C_t = qE_tB_t, \tag{1.22}$$

[5]In biomass dynamics and age-structured models the catch (C_t) is usually treated as a covariate (i.e. known without error) and the catch per unit effort (I_t) is usually treated as data (i.e. an outcome variable). This suggests that we determine the catch (C_t) in advance, and then discover how much effort (E_t) is required to observe this catch, and hence observe $I_t = \frac{C_t}{E_t}$.

where E_t is the fishing effort (year^{-1}), and q is a dimensionless parameter, known as the catchability coefficient, that describes the effectiveness of each unit of fishing effort (see Equation 1.41). Equation 1.22 implies that the catch per unit of fishing effort (I_t) is an index proportional to stock abundance (B_t, see Equation 1.42)

$$I_t = \frac{C_t}{E_t} = qB_t, \tag{1.23}$$

or

$$I_t = q\frac{1}{2}\left(B_t + B_{t+1}\right).$$

There are many other forms of biomass dynamics models including: the Pella & Tomlinson (1969) model where

$$g\left(B_t\right) = \frac{r}{z}B_t\left(1 - \left(\frac{B_t}{K}\right)^z\right),$$

which includes the additional shape parameter z; the form suggested by Fox (1970)

$$g\left(B_t\right) = rB_t\left(1 - \frac{\log\left(B_t\right)}{\log\left(K\right)}\right);$$

difference models (Walters & Hilborn 1976); and regression methods. All of these models pool aspects of production (i.e. recruitment, growth and natural mortality) into the single production function $g\left(B_t\right)$. Biomass dynamics models might be considered a crude over-simplification of a population because they may ignore too many details to produce reliable estimates of the population biomass. In fisheries without age data, or easily distinguished cohorts, they may nevertheless be useful for estimating population trends. We use a biomass dynamics model later in Chapter 5 (page 137).

1.4.2 Age- and length-structured models

A significant improvement over biomass dynamics models are models that incorporate age- or length-structured data. Current fisheries stock assessments are mainly based on two approaches, both relying on catch-at-age or catch-at-length data: statistical catch-at-age or -length models (SCA or SCL) and virtual population analysis (VPA).

Virtual population analysis (VPA), also known as cohort analysis, uses recursive algorithms to calculate past stock abundances based on past catches with no underlying statistical assumptions. VPA can give unreliable estimates for cohorts that have not completely disappeared from the fishery, and it requires an assumption about the natural mortality rate (Hilborn & Kennedy 1992). We do not discuss VPA further.

Statistical catch-at-age models (SCA)

Statistical catch-at-age (SCA) models provide more formal methods for estimating the current abundance of cohorts still being fished and synthesise many aspects of fisheries theory. SCA models commonly estimate a separate recruitment for every year and every initial age class, and two to four selectivity parameters for every fishing fleet, depending on available data. Furthermore, these models can easily be extended to include spatial or economic components when additional complexity is required. Bayesian methods are often used in SCA models to describe uncertainty, and include prior distributions for parameters derived from meta-analysis. A major impetus for using SCA models is their ability to integrate almost any form of data.

Here we present an example of a general SCA model using the exploitation rate formulation (but see page 44 for an alternative catch equation). We present this model for a single time-step t (normally this would be a single year). If $N_{a,t}$ is the number of fish of age a at time t then the fish can be aged at the beginning of the year using

$$N'_{a,t} = N_{a-1,t-1},$$

where $N'_{a,t}$ is the numbers at age after ageing (Equations 1.1 and 1.2). Next, fish recruit to the population using the recruitment function (Equation 1.10). Given R_0, last years spawning stock biomass SSB_{t-1}, and the stock recruitment function $SR(\cdot)$, the number at age after recruitment is

$$N''_{a=1,t} = R_t = R_0 \times SR(SSB_{t-1}) \times e^{\varepsilon_t^R - \sigma_R^2/2} \quad \text{where } \varepsilon_t^R \sim \mathcal{N}\left(0, \sigma_R^2\right).$$

Now we apply half of the natural mortality $M_{a,t}$ to the population

$$N'''_{a,t} = N''_{a,t} e^{-0.5 M_{a,t}},$$

where $N'''_{a,t}$ is the mid-year numbers at age. We can now calculate a series of mid-year biomasses (all with units tonnes). If $w_{a,t}$ is the mean weight (tonnes) of a fish of age a at time t (during the middle of the year), then the total mid-year biomass at age $B_{a,t}$ and the total mid-year biomass B_t is

$$B_{a,t} = N'''_{a,t} w_{a,t} \quad \text{or} \quad B_t = \sum_a N'''_{a,t} w_{a,t}.$$

If $S_{a,t}$ is the selectivity of fish of age a at time t then the mid-year vulnerable biomass at age $V_{a,t}$ and the total mid-year biomass that is vulnerable to the gear used by the fishery V_t is

$$V_{a,t} = N'''_{a,t} w_{a,t} S_{a,t} \quad \text{or} \quad V_t = \sum_a N'''_{a,t} w_{a,t} S_{a,t}.$$

The vulnerable biomass is the portion of a stock's biomass that is available to the fishery. If $m_{a,t}$ is the proportion of fish of age a at time t that mature, then the mid-year spawning stock biomass at age $SSB_{a,t}$ and the total mid-year spawning stock biomass SSB_t is

$$SSB_{a,t} = N'''_{a,t} w_{a,t} m_{a,t} \quad \text{or} \quad SSB_t = \sum_a N'''_{a,t} w_{a,t} m_{a,t}.$$

Now this years exploitation rate at age $U_{a,t}$ (actually the proportion of vulnerable biomass by number caught per annum, rather than by biomass) is applied to the fishery, generating this years catch at age $C_{a,t}$ as a biomass (tonnes).

$$C_{a,t} = U_{a,t} V_{a,t},$$

or

$$U_{a,t} = \frac{C_{a,t}}{V_{a,t}} = \frac{C_{a,t}}{N_{a,t} w_{a,t} S_{a,t}}.$$

This catch is removed from the population using

$$\begin{aligned}
N''''_{a,t} &= N'''_{a,t}\left(1 - U_{a,t}S_{a,t}\right) \\
&= N'''_{a,t}\left(1 - \frac{C_{a,t}}{V_{a,t}}S_{a,t}\right) \\
&= N'''_{a,t}\left(1 - \frac{C_{a,t}\cancel{S_{a,t}}}{N'''_{a,t}w_{a,t}\cancel{S_{a,t}}}\right) \\
&= N'''_{a,t}\left(1 - \frac{C_{a,t}}{N'''_{a,t}w_{a,t}}\right) \\
&= N'''_{a,t} - \frac{\cancel{N'''_{a,t}}C_{a,t}}{\cancel{N'''_{a,t}}w_{a,t}} \\
&= N'''_{a,t} - \frac{C_{a,t}}{w_{a,t}},
\end{aligned}$$

Finally, the remaining half of the natural mortality is removed

$$N_{a,t} = N''''_{a,t}e^{-0.5M_{a,t}},$$

before progressing to the next time-step t.

However, this general form is far from practical and several assumptions need to be made before these models are useful. To identify these assumptions, we start again. As before, $N_{a,t}$ is the number of fish of age a at time t. The fish are aged and new recruits are added to the population using the recruitment function (Equation 1.10)

$$\begin{aligned}
N'_{a,t} &= N_{a-1,t-1}, \\
N''_{a=1,t} &= R_t.
\end{aligned}$$

Next, half of the natural mortality M is applied, but the assumption that M does not change with age or time ($M_{a,t} = M$) is made

$$N'''_{a,t} = N''_{a-1,t-1}e^{-0.5M},$$

where $N'''_{a,t}$ is the mid-year numbers at age. Assuming that the weight at age of fish does not change with time ($w_{a,t} = w_a$), the total mid-year biomass B_t is

$$B_t = \sum_a N'''_{a,t}w_a.$$

Assuming that selectivity does not change with time ($S_{a,t} = S_a$), the mid-year vulnerable biomass at age $V_{a,t}$ and the total mid-year vulnerable biomass V_t is

$$\begin{aligned} V_{a,t} &= N'''_{a,t} w_a S_a, \\ V_t &= \sum_a N'''_{a,t} w_a S_a. \end{aligned}$$

Assuming that maturity does not change with time ($m_{a,t} = m_a$), the mid-year spawning stock biomass SSB_t is

$$SSB_t = \sum_a N'''_{a,t} w_a m_a.$$

Finally, assuming that the exploitation rate is independent of age ($U_{a,t} = U_t$), the catch at time t is

$$\begin{aligned} C_t &= \sum_a C_{a,t} \\ &= \sum_a U_t V_{a,t} \\ &= U_t V_t, \end{aligned} \tag{1.24}$$

or

$$U_t = \frac{C_t}{V_t}$$

The catch is removed from the population using

$$\begin{aligned} N''''_{a,t} &= N'''_{a,t} \left(1 - U_t S_a\right) \\ &= N'''_{a,t} \left(1 - \frac{C_t}{V_t} S_a\right) \\ &= N'''_{a,t} \left(1 - \frac{C_t S_a}{\sum_a N'''_{a,t} w_a S_a}\right). \end{aligned}$$

Finally the remaining half of the natural mortality is removed

$$N_{a,t} = N''''_{a,t} e^{-0.5M}.$$

In practice we never have very reliable estimates of $C_{a,t}$. We do know C_t though, so we can state

$$\begin{aligned}
C_{a,t} &= C_t f_{a,t}, \\
f_{a,t} &= \frac{B_{a,t}S_a}{\sum_a B_{a,t}S_a} = \frac{N_{a,t}w_a S_a}{\sum_a N_{a,t}w_a S_a}, \\
N''''_{a,t} &= N'''_{a,t}\left(1 - \frac{C_{a,t}}{N'''_{a,t}w_a}\right) \\
&= N'''_{a,t}\left(1 - \frac{C_t f_{a,t}}{N'''_{a,t}w_a}\right) \\
&= N'''_{a,t}\left(1 - \frac{C_t S_a N'''_{a,t}w_a}{N'''_{a,t}w_a \sum_a N'''_{a,t}w_a S_a}\right) \\
&= N'''_{a,t}\left(1 - \frac{C_t S_a}{\sum_a N'''_{a,t}w_a S_a}\right). \qquad (1.25)
\end{aligned}$$

Or, we can allow selectivity to be time-varying

$$N''''_{a,t} = N'''_{a,t}\left(1 - \frac{C_t S_{a,t}}{\sum_a N'''_{a,t}w_a S_{a,t}}\right).$$

Also, note if $C_{a,t} = U_t V_{a,t}$ then we can also define the catch in numbers (rather than tonnes) as

$$\check{C}_{a,t} = \frac{C_{a,t}}{w_a}$$

and

$$\check{V}_{a,t} = \frac{V_{a,t}}{w_a} = N'''_{a,t}S_a$$

therefore

$$\begin{aligned}
\check{C}_{a,t} &= U_t \check{V}_{a,t}, \\
\check{C}_t &= U_t \check{V}_t = U_t N'''_{a,t} S_a, \\
N''''_{a,t} &= N'''_{a,t} - \check{C}_{a,t} = N'''_{a,t} - U_t \check{V}_{a,t} = N'''_{a,t} - U_t N'''_{a,t} S_a = N'''_{a,t}(1 - U_t S_a).
\end{aligned}$$

In summary, SCA models are implemented as

$$
\begin{aligned}
N'_{a,t} &= N_{a-1,t-1} && \text{age the fish (Equations 1.1 and 1.2)} \\
N''_{t=1,t} &= R_t && \text{recruitment (Equation 1.10)} \\
N'''_{a,t} &= N''_{a,t} e^{-0.5M} && \text{apply half of the natural mortality} \\
B_t &= \sum_a N'''_{a,t} w_a && \text{total biomass} \\
V_t &= \sum_a N'''_{a,t} w_a S_a && \text{biomass vulnerable to fishing} \\
SSB_t &= \sum_a N'''_{a,t} w_a m_a && \text{spawning stock biomass (Equation 1.9)} \\
C_t &= U_t V_t && \text{calculate the catch} \\
N''''_{a,t} &= N'''_{a,t} \left(1 - U_t S_a\right) && \text{remove the catch} \\
N_{a,t} &= N''''_{a,t} e^{-0.5M} && \text{apply remaining half of the natural mortality}
\end{aligned}
\tag{1.26}
$$

where $N'_{a,t}$ represents the numbers at age a and time t after aging, $N''_{a,t}$ after recruitment, $N'''_{a,t}$ after half of the natural mortality, $N''''_{a,t}$ after fishing mortality and $N_{a,t}$ at the end of the year after the remaining half of the natural mortality. The total biomass, vulnerable biomass and exploitation rate are calculated after aging, recruitment and half of the natural mortality. The order in which ageing, recruitment, mortality and spawning are applied in Equation 1.26 is simply an example and the order in which these processes are applied in practice may differ depending on what species/stock is being modelled.

Statistical catch-at-length models (SCL)

Statistical catch-at-length (SCL) models provide more formal methods for estimating the current abundance of length-classes being fished. SCL models are applicable when the animals cannot be aged or the growth curve is indeterminate (i.e. t_0, see Equation 1.4, cannot be estimated), or if the primary source of data is length-composition observations. SCL models are much less common than SCA models.

The structure of SCL models is very similar to that of SCA models, except

that the basic dynamics are size- rather than age-structured and recruitment is spread over several length-classes ℓ rather than just the minimum age a_{min}. The numbers of fish in size-class ℓ at time t in a SCL model is $N_{\ell,t}$ where $\ell = 1, \ldots, n$ and n is the number of size-classes. Alternatively the vector $\mathbf{N}_t$ could be used. To move fish between length-classes due to growth requires an $n \times n$ growth matrix $\mathbf{X}$

$$\mathbf{X} = X_{\ell,\ell'} = \begin{pmatrix} X_{1,1} & X_{1,2} & \ldots & X_{1,n} \\ X_{2,1} & X_{2,2} & \ldots & X_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n,1} & X_{n,2} & \ldots & X_{n,n} \end{pmatrix} \quad \text{where } \sum_{\ell'} X_{\ell,\ell'} = 1 \quad \forall \ell, \tag{1.27}$$

where $X_{\ell,\ell'}$ represents the proportion of fish growing from the length-class ℓ to length-class ℓ' and has the property $X_{\ell,\ell'} \geq 0$. The matrix $\mathbf{X}$ is often constrained to prevent "negative growth" (i.e. fish that shrink, $X_{\ell,\ell'} = 0$ if $\ell > \ell'$)

$$\mathbf{X} = X_{\ell,\ell'} = \begin{pmatrix} X_{1,1} & X_{1,2} & \ldots & X_{1,n} \\ 0 & X_{2,2} & \ldots & X_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & X_{n,n} \end{pmatrix} \quad \text{where } \sum_{\ell'} X_{\ell,\ell'} = 1 \quad \forall \ell.$$

This implies that $X_{n,n} = 1$. A SCL model also requires specification of the natural mortality and fishing mortality. The natural mortality may be applied using an $n \times n$ diagonal survival matrix $\mathbf{S}$

$$\mathbf{S} = S_{\ell,\ell'} = \begin{pmatrix} e^{-M} & 0 & \ldots & 0 \\ 0 & \ddots & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & e^{-M} \end{pmatrix}, \tag{1.28}$$

and the fishing mortality an $n \times n$ diagonal matrix

$$\mathbf{H}_t = H_{\ell,\ell',t} = \begin{pmatrix} 1 - S_1 F_t & 0 & \ldots & 0 \\ 0 & \ddots & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 - S_n F_t \end{pmatrix}. \tag{1.29}$$

where S_ℓ is the selectivity on size-class ℓ and F_t is the fishing mortality during time t. Now we can write the numbers of fish in each length-class the following year $N_{\ell,t+1}$ as

$$\begin{aligned} N_{\ell,t+1} &= R_{\ell,t} + \sum_{\ell'}\sum_{\ell''}\sum_{\ell'''} X^T_{\ell,\ell'} S_{\ell',\ell''} H_{\ell'',\ell''',t} N_{\ell''',t} \\ &= R_{\ell,t} + (X^T S \mathbf{H}_t \mathbf{N}_t)_\ell \qquad \text{or} \\ \mathbf{N}_{t+1} &= \mathbf{R}_t + \mathbf{X}^T \mathbf{S} \mathbf{H}_t \mathbf{N}_t, \end{aligned} \tag{1.30}$$

where $\mathbf{R}_t$ is a vector of the number of recruits (to each size-class) at the start of year t.

Catch equations

The catch equation in SCA (and SCL) models can be formulated in two different ways (Bull et al. 2012). The first option, **instantaneous mortality**, applies half of the annual natural mortality, then applies the mortalities from all fisheries instantaneously, then applies the remaining half of the natural mortality. The second option is to use the **Baranov** catch equation, in which natural and fishing mortality occur simultaneously. In both cases, we often assume that $M_{a,t} = M$.

When using **instantaneous mortality** the fishery removals are applied using an exploitation rate U_t each year t. Thus $N_{a,t}$, the number of fish of age a in year t is

$$N_{a,t} = N_{a-1,t-1} e^{-0.5M} (1 - U_{t-1} S_a) e^{-0.5M}, \tag{1.31}$$

where M is the natural mortality rate and S_a is the selectivity of the fishing gear used for fish of age a. This model ages all of the fish in the partition by one year ($N_{a-1,t-1} \leftarrow N_{a,t}$ reduced by all mortality) and updates the year. This equation assumes that all fishing takes place instantaneously during the middle of the year because V_t is calculated during the middle of the year (i.e. after removing half of the natural mortality using $e^{-0.5M}$), and that U_t is independent of age. Work by Mertz & Myers (1996) suggests that age-structured models are largely insensitive to this assumption.

When using **Baranov mortality** the fishery removals are applied each year t using the fishing mortality rate F_t (again independent of age). Thus $N_{a,t}$,

the number of fish of age a in year t is

$$N_{a,t} = N_{a-1,t-1}e^{-Z_{a-1,t-1}}, \tag{1.32}$$

and F_t can be found by solving the Baranov catch equation

$$C_t = \sum_a \left[\frac{F_t S_a}{Z_{a,t}} w_a N_{a,t} \left(1 - e^{-Z_{a,t}}\right) \right], \tag{1.33}$$

where $Z_{a,t}$ is the total annual mortality and is defined as $Z_{a,t} = M + F_t S_a$. There is no closed form equation for F_t given the other parameters, so this equation must either be solved numerically (e.g. using the Newton-Raphson method, Press et al. 1986), approximated (e.g. Popes approximation, Pope 1972), or by treating the catches as observations with small coefficients of variation, so that the F_t's are estimated as parameters (specifying small coefficients of variation forces the estimated catches to be close to the observed catches).

There has been some debate in the literature over the use of instantaneous mortality and Baranov mortality equations (Branch 2009*b*, Branch 2009*a*, Francis 2010, Branch 2010). Put simply, both methods are wrong. Instantaneous mortality models assume that all fishing occurs instantaneously at some time in the model, and it is obvious that this would never be the case in reality. However, Baranov mortality models assume that fishing occurs constantly throughout the year, which is also unrealistic as fishing activity is never constant but instead fluctuates according to the weather, price and availability of fish, and many other factors. Furthermore, due to the need for numerical methods, Baranov mortality equations result in more computationally expensive models.

1.4.3 Individual-based models

Individual-based models (IBMs) represent populations in which individuals differ in their maturation, migration, growth, and mortality and follow these individuals, or small groups of individuals (super-individuals or agents, e.g. Scheffer et al. 1995), through their life history. Although

it is a very flexible modelling approach, the computational overheads involved in an individual-based model can overwhelm even the most powerful computers. This can make some models infeasible particularly if trying to estimate parameter values. However, new hybrid methods that combine classical models with individual-based aspects may be the key to solving such dilemmas (e.g. Gray et al. 2006).

The diversity of methods/equations that may be used in IBMs is enormous so only a couple of examples will be given here (for further examples see Grimm & Railsback 2005, Kim et al. 2002 or Gray et al. 2006). In an agent-based model we have $f^p_{k,t}$ individuals in agent k at time t. k could refer to an individual, a group of individuals (super-individual), an age group a, or a size-class ℓ. To determine the total number of individuals N_t in an agent-based model in any given year we would use

$$N_t = \sum_k f^p_{k,t}. \tag{1.34}$$

There are several ways to apply processes (e.g. natural mortality) to individuals (or bins of individuals) within an IBM. Taking natural mortality M as an example, for each year in the life of individual (or agent) k the Bernoulli distribution can be used to determine if the individual survives. The individual is alive if $Y_{k,t}$ is one or is dead if $Y_{k,t}$ is zero where $Y_{k,t}$ is given by

$$Y_{k,t}|Y_{k,t-1} = 1 \sim \text{Bern}(1 - e^{-M}), \tag{1.35}$$

conditional on $Y_{k,t-1} = 1$ (i.e the fish had to be alive in the first place).

While modelling populations as individuals or small groups of individuals may come at a computational cost, the benefits may make it worthwhile. In species or populations with small numbers of individuals such as dolphins or whales, such models make sense if the data is available as the computational overheads will be reasonably low and complex behavioral attributes can be built in. However, if modelling populations with large numbers of individuals (e.g. fish) then there must be a good reason to do so using IBM's. Chapter 4 (page 89) of this this thesis presents an agent-based model in some detail.

1.5 Data

We commonly find two time series of observations in stock assessment. The first is a history of catches removed from the stock. The second is an index of abundance: some measure that indicates the size of the stock. Other information such as knowledge of the age structure of the population, individual growth rates, fecundity at different ages, breeding seasons, or other basic biology is usually available and may be useful.

The purpose of observations in stock assessment is to inform the parameters of the model. The values for some parameters within stock assessment models usually cannot be determined from auxiliary information (e.g. data that is not included in the stock assessment model), nor estimated reliably by fitting a model to data and must, therefore, be prespecified (Magnusson & Hilborn 2007). Examples of these parameters include natural mortality (M), the steepness of the stock-recruitment relationship (h, Equation 1.11 or 1.12), and the variation in recruitment (σ_R^2, Equation 1.13). These parameters are typically set using estimates from similar species, other stocks of the same species, meta-analysis, expert opinion, or some other method (e.g. see Figure 1.9).

The most common observations used to inform stock assessment models are proportions in the commercial catch-at-age data and an abundance index (usually catch per unit effort or a trawl survey). Proportions in the catch-at-age data inform the selectivity and age structure in stock assessment models. The abundance index informs the stock biomass. Each of these data types will be discussed below. Tag-recapture data is also briefly described below as an alternative to abundance indices.

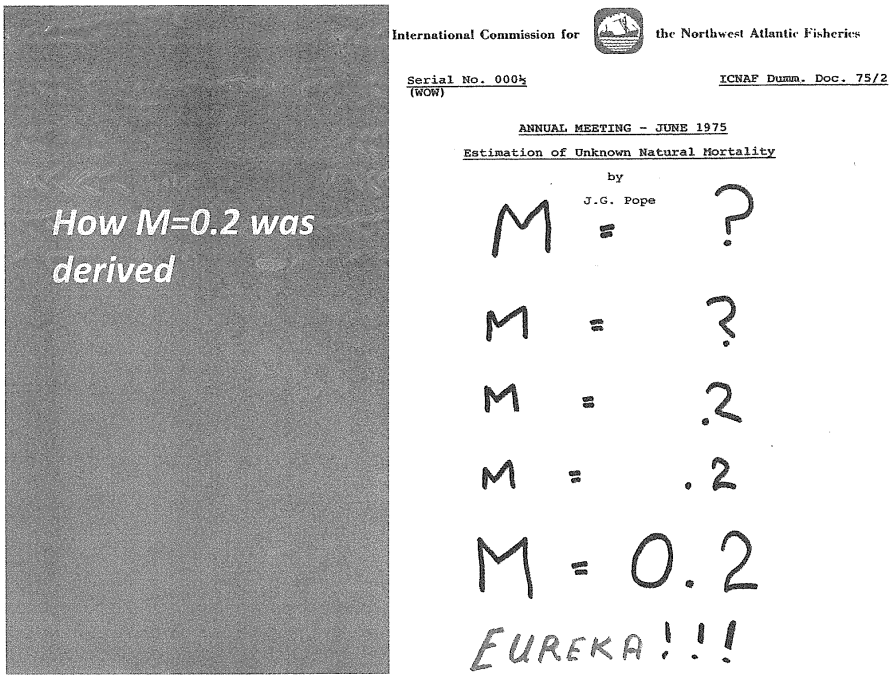


Figure 1.9: The derivation of natural mortality (M) by J.G. Pope. This illustrates the fact that 0.2 is a common, but sometimes unjustified, value for M.

1.5.1 Proportions in the catch-at-age

The proportions at age in the catch ($(P_a)_t$) can be derived from the population as

$$\begin{aligned} \lambda_{a,t} &= N_{a,t} U_t S_a e^{-0.5M}, \\ (P_a)_t &= \frac{\lambda_{a,t}}{\sum_a \lambda_{a,t}} \quad \text{where} \quad \sum_a (P_a)_t = 1 \quad \forall t, \end{aligned} \tag{1.36}$$

where $\lambda_{a,t}$ is the vulnerable numbers (not biomass) at age a that are caught at time t. The proportions at age in the catch ($(P_a)_t$) for a single year t is often referred to as an age-frequency. Observations of the proportions at age may be collected and compared to those in our model using a multinomial or Dirichlet distribution. However, these observations are rare as they require that many fish be aged (e.g. reading their otoliths).

A reasonably common type of data in fisheries science are length-frequencies $(Q_\ell)_t$ that specify the proportions of fish in length-class ℓ at time t in the catch

$$(Q_\ell)_t \quad \text{where} \quad \sum_\ell (Q_\ell)_t = 1 \quad \forall t. \tag{1.37}$$

These samples are measured by observers onboard fishing vessels or by collecting market samples (i.e. when fish are landed). These length-frequency data can be used as observations in the model as is, but often they are combined with age data to get age-frequencies.

Often a small set of otoliths (and the length and weight of the fish associated with each otolith) are also sampled, by observers onboard fishing vessels or by collecting market samples, for ageing each year. These aged fish can be used to derive age-length relationships (e.g. Equation 1.4) and length-weight relationships (e.g. Equation 1.5). These may then be used to develop an age-length key $M^{\text{key}}_{a,\ell}$. An age-length key is an $a \times \ell$ age-size joint distribution matrix where each row corresponds to an age-class a and each column to a length-class ℓ. An age-length key specifies the relative proportion of fish of length ℓ that belong in age-class a

$$\begin{aligned} M^{\text{key}}_{a,\ell} &= P\left(A_i = a | Y_i = \ell\right) \\ &= \phi(Z_{a,\ell}). \end{aligned} \tag{1.38}$$

where $P\left(A_i = a|Y_i = \ell\right)$ is the conditional probability that a randomly selected fish i is of age a given it is of length ℓ. The conditional probability $P\left(A_i = a|Y_i = \ell\right)$ is determined using the appropriate growth parameters and the coefficient of variation c of the age-length relationship and is equivalent to $\phi(Z_{a,\ell})$. $\phi(Z_{a,\ell})$ is the normal probability density that a fish of length ℓ is age a, where $Z_{a,\ell}$ is the normal Z-score for a fish of age a and length ℓ, calculated as

$$Z_{a,\ell} = \frac{\ell - L_a}{\sigma_a} \quad \text{where } \sigma_a = cL_a, \tag{1.39}$$

and L_a is the mean length (cm) of a fish of age a and σ_a is the standard deviation of the age-length relationship for a. The normal Z-score is then converted to a cumulative normal distribution for each age a to give the probability that a fish of length ℓ will exceed age a, and finally this is converted to the probability that a fish of length ℓ is age a or $P\left(A_i = a|Y_i = \ell\right)$. The age-length key is applied to length-frequency distributions for each time t to derive proportions in the catch-at-age observations

$$(\widehat{P}_a)_t = \sum_{\ell} \left[\frac{M^{\text{key}}_{a,\ell} (Q^{\text{obs}}_{\ell})_t}{\sum_{a'} M^{\text{key}}_{a',\ell}} \right] \quad \text{where} \quad \sum_{a} (\widehat{P}_a)_t = 1 \quad \forall t. \tag{1.40}$$

$(\widehat{P}_a)_t$ is then compared to the proportions at age of the catch in the model using a multinomial or Dirichlet distribution.

This is different to the proportion at age in the population which would be $N_{a,t} S_a e^{-0.5M}$ (note the lack of U_t). This relationship is used if comparing to a trawl survey rather than commercial catch.

1.5.2 Abundance indices

An index of abundance is a measure that is assumed to be proportional to the biomass of a stock, or at least the biomass that is vulnerable to fishing. Abundance indices include catch per unit effort (CPUE) or catch rate, trawl surveys, and acoustic surveys. CPUE is calculated using catch and effort data collected from commercial fishers. Fishery-independent surveys are less common as the data are often costly or difficult to collect.

Francis (2006) provides an example of a stock assessment informed using CPUE and trawl surveys. Dunn & Hanchet (2011) provide an example of a stock assessment informed using acoustic survey data. Here we discuss CPUE in more detail.

Catch per unit effort

In many fisheries, CPUE is the primary index of abundance. For some, it is the only index. CPUE is the quantity of fish (numbers or biomass) caught with a standard unit of fishing effort (e.g. the number of fish taken per 1000 hooks per hour or the weight of fish taken per hour of trawling with a net of a fixed size at time t). For a standard unit of effort applied to a given fish stock, we assume that the expected catch is proportional to the stock biomass. This implies that if the stock biomass halves over time, then so will the expected catch for a given standard unit of effort. This relationship can be expressed as

$$C_t = qE_tV_t, \tag{1.41}$$

where C_t is the catch (tonnes) during time t, E_t is the effort at time t, V_t is the biomass (tonnes) that is vulnerable to fishing at time t, and q is called the catchability coefficient (a nuisance parameter) and represents the proportion of fish vulnerable to capture caught in a year for a standard unit of effort (Arreguin-Sanchez 1996). However, many factors other than stock biomass are known to affect catches. For example, fishing time (time of day), gear specifications (e.g. the headline height of a trawl net), vessel type (capacity and horsepower), season, and location to name a few. Thus, in commercial fisheries any change in catch rate from year to year will be caused partly by variation in all of these other factors and partly by annual changes in abundance. Therefore, to specify CPUE we must remove the effect of all of these other factors so that we can infer how much the abundance has changed from year to year. We call this "standardising" the CPUE.

Standardisation of CPUE is usually done using generalised linear models (GLMs, Nelder & Wedderburn 1972), although generalised additive models (GAMs) and generalised linear mixed models (GLMMs) have also been used. GLMs are defined by the statistical distribution for the response variable (usually catch rate, e.g. catch per day or catch per potlift) and how some linear combination of a set of explanatory variables relate to the expected value of the response variable. These explanatory variables may be continuous or discrete and might include, for example, fishing year, month, statistical area and/or vessel. The variables used represent a compromise between available variables and those that explain an effect that we wish to remove. Year must be one of the explanatory variables because the primary objective of standardising catch and effort data is to detect trends in abundance over time. These models can be written

$$g(\mu_i) = \mathbf{x}_i^\top \boldsymbol{\beta},$$

where $g(\cdot)$ is the link function, $\mathbf{x}_i$ is a $p \times 1$ vector of explanatory variables for the ith value of the response variable, and $\boldsymbol{\beta}$ is a $p \times 1$ vector of parameters.

For example, in the rock lobster fishery (where lobsters are caught using pots) data is collected at the **trip** level (i.e. a single day within a single Quota Management Area (QMA). If the fisher moves into a different QMA within a single day and fishes then this would consist of two trips). We denote a single trip by j. For a given trip a fisher might do several **potlifts**, we define E_j the number of potlifts during trip j (the effort). They would then record an estimate of the **catch** C_j associated with those potlifts during that trip. After several trips the fisher might need to land their catch (as there will be crayfish all over the boat). This is called the **landing event**. We denote a landing event as φ. Upon landing the catch, their estimates of the catch during each fishing event can be verified by summing up their estimates $\sum_{j \in \varphi} C_j$ and comparing this to the **landed weight** C_φ. The ith value in a GLM could consist of just one observation per explanatory variable per year (e.g.

by summing up the catch and effort in each year, month, area, vessel combination and calculating the catch rate in each of these strata) or there could be multiple observations per stratum (e.g. by leaving the observations at the trip level). It is usually better to use the former of these two approaches to avoid zero catches that might be common at the trip level (as these become an issue if wanting to use a GLM with a log-normal response variable).

For example, the normal linear model is a special case of a generalised linear model

$$\begin{aligned}\mathbb{E}[Y_i] = \mu_i &= \mathbf{x}_i^\top \boldsymbol{\beta},\\ Y_i &\sim \mathcal{N}(\mu_i, \sigma^2),\end{aligned}$$

where $Y_1, \ldots, Y_N$ are assumed independent. In this case the link function is the identity function, $g(\mu_i) = \mu_i$. This model is usually written

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{e},$$

where

$$\mathbf{y} = [y_1 \cdots y_N]^\top, \qquad \mathbf{X} = [\mathbf{x}_1 \cdots \mathbf{x}_N]^\top, \qquad \mathbf{e} = [e_1 \cdots e_N]^\top,$$

and the e_i's are iid random variables with $e_i \sim \mathcal{N}(0, \sigma^2)$ for $i = 1, \ldots, N$. In this form, the component $\boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}$ represents the signal and $\mathbf{e}$ the error.

Applying these ideas to our rock lobster example, we could have have the response variable $y_t = C_t$ with the explanatory variables

$$\begin{aligned}\mathbf{x}_1 &= \text{fishing year},\\ \mathbf{x}_2 &= \text{potlifts},\\ \mathbf{x}_3 &= \text{vessel},\\ \mathbf{x}_4 &= \text{area},\\ \mathbf{x}_5 &= \text{month}.\end{aligned}$$

The definition of CPUE I_t is the ratio of catch to effort in year t, which from Equation 1.41 is

$$I_t = \frac{C_t}{E_t} = qV_t. \tag{1.42}$$

The units of catch are usually tonnes. The units of effort are in some sense arbitrary, and time series of CPUE data are usually normalised over all years for convenience using

$$\tilde{I}_t = \frac{I_t}{\overline{I}}, \tag{1.43}$$

where $\tilde{I}_t$ is the normalised CPUE series (in canonical form) and $\overline{I}$ is the geometric mean of I_t (i.e. $\overline{I} = \exp(\frac{1}{n}\sum_{j=1}^{n}\log I_j) = \exp(\frac{1}{n}\log(\prod_{j=1}^{n} I_j))$). An important property of normalised CPUE indices in stock assessments is that they are scale independent. That is, the biomass estimates are unchanged if the CPUE index values $I_1, I_2, ..., I_n$ are replaced by $kI_1, kI_2, ..., kI_n$ for some constant k. The only effect of this replacement is that the value of q is multiplied by k (Francis 1999).

This approach assumes a strict proportionality between CPUE and vulnerable abundance. However, CPUE may not always be proportional to abundance. For this reason, nonlinear models have also been proposed, the simplest being the power curve

$$I_t = qV_t^{\beta}, \tag{1.44}$$

where if $\beta = 1$ the equation reduces to Equation 1.42 and if $\beta \neq 1$ then catchability changes with abundance (Figure 1.10). When $\beta > 1$, then CPUE I_t declines faster than the biomass V_t, a phenomenon known as hyperdepletition which can result in the underestimation of biomass. If $\beta < 1$, then CPUE I_t declines slower than the biomass V_t; this is called hyperstability which can result in overestimation of biomass and underestimation of fishing mortality. Empirical studies studies suggest that the most common form of nonproportionality is hyperstability (Harley et al. 2001).

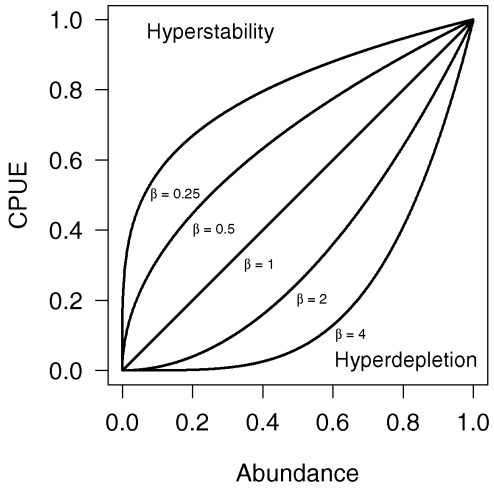


Figure 1.10: Relationship between CPUE and abundance based on different values of the shape parameter β.

Tag-recapture

Tag-release and recapture data may be used to estimate the population abundance and age-structure, growth, mortality, and/or movement within stock assessment models. For example, the current Antarctic toothfish stock assessment model uses tag-recapture data to inform population abundance and age-structure (Mormede et al. 2011). The current New Zealand rock lobster stock assessment models use tag-recapture data to determine the growth rate of lobsters spatially and temporally (Haist et al. 2011).

To use tag data to estimate the population abundance of fish within a stock assessment model, a length stratified extension of the Lincoln-Peterson (Seber 1982) estimator is used

$$\widehat{N}_y = \frac{M_{t,y} n_y}{m_{t,y}}, \tag{1.45}$$

where $\widehat{N}_y$ is the estimated number of fish during year y in the available population (tagged and untagged), $M_{t,y}$ is the number of fish tagged in year t in the available population in year y, n_y is the number of fish during year y that were scanned for a tag, and $m_{t,y}$ is the number of fish tagged

during year t that were recaptured in year y. The model assumes that the population mixes homogeneously, that no tags are lost (i.e. fall off), and that the population is closed.

1.6 Fisheries management

We briefly describe fisheries management in New Zealand. All of this information can be found on the Ministry for Primary Industries (MPI) website (`http://fs.fish.govt.nz/Page.aspx?pk=81`).

1.6.1 Quota Management System (QMS)

The Quota Management System (QMS) was introduced to New Zealand in 1986, to help ensure the sustainable utilisation of fisheries resources through the control of commercial catches for each species within specified geographical areas.

Each species in the QMS is subdivided into separate fish stocks defined by Quota Management Areas (QMAs). There are about 100 different species (or species groupings) in New Zealand that are split up into 638 separate stocks. Each of these stocks are managed independently.

QMAs for a species are determined on introduction of that species into the QMS. QMAs are based on a combination of biological and administrative factors at the time of introduction. The starting point for determining QMA boundaries for each species are the ten Fisheries Management Area (FMA) which make up New Zealand's Exclusive Economic Zone (EEZ). Some QMAs incorporate multiple FMAs while others cover only part of a single FMA.

1.6.2 Total Allowable Commercial Catch (TACC)

The Ministry (MPI), scientists, and other stakeholders (including industry), work together to assess the population size of major commercial fish

species in their major fishing grounds. Using the assessment data, the Minister for Primary Industries then sets an annual Total Allowable Catch (TAC, tonnes) limit for each fish stock.

In fisheries where non-commercial users fish (e.g. customary Maori, recreational fishers, illegal fishing), a portion of the stock is set aside for them before the Total Allowable Commercial Catch (TACC) is set. The TACC is set in volume (e.g. tonnes) allowed to be caught each year by commercial fishers and can vary from year to year. Thus the TAC each year is

$$TAC = TACC + \text{customary} + \text{recreational} + \text{illegal}$$

The TACC is then divided into a number of Individual Transferable Quotas (ITQ), which are effectively rights to fish a defined portion of the TACC.

1.6.3 Annual catch entitlement (ACE)

Annual Catch Entitlement (ACE) or ITQ is the right to harvest a defined amount of a species (percentage by weight of the TACC) in a specified area during a single fishing year. For most quota-managed species the year runs from 1 October to 30 September. However, the fishing year for rock lobster and southern blue whiting, as well as some other minor stocks, is from 1 April to 30 March.

If someone holds quota for 6% of the TACC for a particular species in an area, they hold the right to harvest 6% of that area's TACC. However, the amount harvested will change each year - depending on what the year's TACC volume is set at.

Quota is an asset and can be sold, leased or given away. Its value depends on the market value of the species, the TACC and demand for that particular quota. Most quota trading is by personal contacts and advertisements in daily papers and in seafood trade magazines. There are also a number of well-established quota broking companies.

All quota trades must be registered with FishServe, who provide registry services to the New Zealand commercial fishing industry for the QMS. The Fisheries Act limits how much quota any one person or company can

own - so that no one company or individual can develop a monopoly on fishing in any one area or for any one species. These aggregation limits are set by MPI, in consultation with industry representatives.

1.6.4 Deemed values (DV)

The QMS requires all catches of quota species to be constrained within the TACC set for each stock. Dumping of QMS species (disposal at sea with or without reporting) is prohibited (except in limited circumstances) and it is important that all fish that are taken are landed and reported, so that fisheries management decisions can be made from accurate information of catch levels. The Act requires fishers to balance their catch of quota stocks with ACE.

Deemed Values (DV) are monetary demands on a commercial fisher whose catch of quota stocks exceeds their annual ACE holding. The Minister sets a DV rate for each of the fish stocks in the QMS. In setting DVs the Minister must take into account the need to provide an incentive for every commercial fisher to acquire or maintain sufficient ACE to cover their catch each fishing year. When setting the DV rates, the Minister may take into account other matters, such as the market value of the ACE and the fish, economic benefits gained by the fisher or receiver/processor, and the extent to which the catch of the stock exceeds or is likely to exceed the TACC. If a fishing permit holder's reported catch for the year is greater than the ACE held for that fish stock the permit holder is charged a DV.

Chapter 2

Bayesian inference

In this chapter we discuss Bayesian inference approaches that can be used in fisheries problems. These methods are used and referred to extensively throughout Chapters 5, 6, and 7. A list of variables commonly used in this chapter is given in Table 2.1.

Table 2.1: List of variables commonly used throughout this chapter.

Symbol	Type	Description
y	vector	Data
θ	vector	Parameter set
$f(y)$	distribution	The marginal likelihood of the data
$\pi(\theta)$	distribution	Joint prior distribution of the parameters
$f(y\|\theta)$	distribution	The likelihood of the data conditional on the parameters
$f(\theta\|y)$	distribution	Joint posterior distribution
i	scalar	The current MCMC step
$\theta^{(i)}$	vector	Current state during step i
θ^*	vector	Candidate point or proposal
$q(\cdot\|\cdot)$	distribution	Proposal distribution for a parameter in an MCMC update
r	scalar	Acceptance ratio

2.1 Introduction

Over the last few decades cultures in stock assessment have developed around the world. Here in New Zealand almost all stock assessment models use Bayesian methods with a preference for statistical catch-at-age (or -length) models if at all possible. Conversely, with the exception of a few, much of the United States and Europe use maximum likelihood methods.

The main advantages of Bayesian inference over frequentist inference is that the method estimates a full probability distribution over the model parameters resulting in a more complete representation of the uncertainty associated with a model or parameter estimate. Furthermore, Bayesian inference with proper priors can be made immune to singularities and near-singularities with matrix inversions, unlike frequentist inference. Bayesian estimation also provides a mechanism to incorporate additional information about model parameters, in the form of a prior probability distribution. Bayesian methods are generally recommended in fisheries models for quantifying uncertainty (Magnusson et al. 2013). For these reasons we do not consider maximum likelihood methods further in this thesis.

2.1.1 Bayes' theorem

Bayes' theorem is the foundation of Bayesian inference. Bayes' theorem expresses the conditional probability, or "posterior probability", of an event A after B is observed in terms of the "prior probability" of A, prior probability of B, and the conditional probability of B given A, denoted $B|A$

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}.$$

For model-based inference, B is replaced with observed data y, A with parameter set θ

$$f(\theta|y) = \frac{f(y|\theta)\pi(\theta)}{f(y)}, \tag{2.1}$$

where $\pi(\theta)$ is the joint **prior** distribution for parameter set θ, and uses probability as a means of quantifying uncertainty about θ before taking the

data into account, $f(y|\theta)$ is the **likelihood** or likelihood function, which encodes the data generation process as a full probability model, $f(y)$ is the **marginal likelihood** of the data, and $f(\theta|y)$ is the joint **posterior** distribution that expresses uncertainty about the parameter set θ after taking both the prior and data into account. Obtaining the posterior distribution is the objective of Bayesian inference. The marginal likelihood $f(y)$ is often not computable, therefore interest focuses on the computable parts of the posterior, related by the proportionality

$$f(\theta|y) \propto f(y|\theta)\pi(\theta). \tag{2.2}$$

2.2 Priors $\pi(\theta)$

The prior probability distribution $\pi(\theta)$ for the parameter θ summarises the investigator's knowledge of θ before new evidence is taken into account. Priors can broadly be categorised into two types: **informative** priors and **vague** priors.

An informative prior expresses specific, definite information about a variable. Informative priors are usually based on previous analyses or meta-analysis.

A vague prior expresses vague or general information about a parameter. Vague priors can express minimal information such as simply defining that the parameter is positive or that the parameter is less than some limit. For instance, it is standard practice to use high variance, uniform priors in fisheries models for almost all model parameters while recognising that transformed versions of parameters with uniform priors are not uniform. There are many alternative vague priors including constant, Jeffreys' and Zellners's priors (Marin & Robert 2010). Also, the inverse gamma distribution with high variance is sometimes a good choice of vague prior for variance parameters (Gelman 2006).

However, a vague prior can induce more information than one might expect. Lambert et al. (2005) discourage use of the term "non-informative prior" for this reason. For example, if a uniform prior distribution is used,

results can be sensitive to the choice of lower and upper limits. In Gelman (2006), sensitivity to the choice of parameters of an inverse gamma prior for variance parameters is highlighted. There is also the issue that some non-informative priors can give improper posterior distributions. Therefore, improper priors should be used with caution, and sensitivity tests to prior choice is recommended.

2.3 Likelihoods $f(y|\theta)$

The likelihood function defines the probability (density) of the data y given the parameters θ. The likelihood $f(y|\theta)$ is commonly expressed on the log scale, i.e. as a log-likelihood $\ell(\theta|y)$. Likelihood functions play a key role in Bayesian inference.

2.4 Markov chain Monte Carlo (MCMC)

Markov chain Monte Carlo (MCMC) algorithms, or samplers, are numerical approximation algorithms used for drawing samples from probability distributions where direct sampling is not possible. In Bayesian inference, the distribution of interest is usually a posterior distribution. There are a large number of MCMC algorithms. Popular families include Gibbs sampling (Geman & Geman 1984), Metropolis-Hastings (MH), Hamiltonian Monte Carlo, and many others (see `http://www.bayesian-inference.com/mcmc` for a list of algorithms supported by the R package "Laplaces Demon"). All MCMC algorithms are known as special cases of Metropolis et al. (1953) and Hastings (1970). Regardless of the algorithm, MCMC seeks to sample from the posterior and map out the entire posterior distribution, and not just find its maximum. If we wanted to find the maximum posterior density we would just use an optimiser on the posterior. The need for MCMC arises as sometimes we cannot compute or sample directly from the joint posterior distribution $f(\theta|y)$.

In general, we have model space $\mathbb{M}$ with countable elements $M \in \mathbb{M}$.

Model $M \in \mathbb{M}$ has parameters $\theta_M \in \Theta_M$ for parameter space Θ_M. The target distribution is $f(\theta_M, M) = f(\theta_M|M)f(M)$, which may be a Bayesian posterior distribution

$$f(\theta_M, M|y) = f(\theta_M|M, y)f(M|y) = \frac{f(y|\theta_M, M)\pi(\theta_M, M)\pi(M)}{f(y)}.$$

In an MCMC sampler we propose moves between states (θ_M, M) to (θ_M^*, M^*). At each step of the chain a move type x is selected from the move space $\mathbb{X}$. In the most general formulation, these move types may include trans-dimensional moves between models of greater and lesser complexity as well as parameter updates within a given model. In this project we implement only fixed dimensional MCMC samplers, but for more information on trans-dimensional samplers see Green (1995).

2.4.1 Metropolis-Hastings (MH)

Here we introduce the Metropolis-Hastings algorithm for a single model (M) and a single move type (x). As we only deal with a single model we drop the model subscript M. At each step i of the Markov chain, the current state $\theta^{(i)}$ is chosen by first sampling a candidate point θ^* from a proposal distribution $q(\theta^*|\cdot)$. The choice of the proposal distribution $q(\cdot|\cdot)$ is almost entirely arbitrary in MCMC. However, a well chosen $q(\cdot|\cdot)$ will result in an MCMC chain that mixes well and therefore converges within our lifetime. The proposal distribution may depend on the previous state $\theta^{(i-1)}$ such that $q(\cdot|\theta^{(i-1)})$. For example, $q(\cdot|\theta^{(i-1)})$ could be a multivariate normal distribution with mean $\theta^{(i-1)}$ and a fixed covariance matrix Σ. Once we have decided on a proposal distribution we can randomly generate a new candidate point θ^*. The candidate point is then accepted with probability r where

$$r = \min\left(1, \frac{f(\theta^*|y)}{f(\theta|y)} \times \frac{q(\theta|\theta^*)}{q(\theta^*|\theta)}\right) = \min\left(1, \frac{\frac{f(y|\theta^*)\pi(\theta^*)}{\cancel{f(y)}}}{\frac{f(y|\theta)\pi(\theta)}{\cancel{f(y)}}} \times \frac{q(\theta|\theta^*)}{q(\theta^*|\theta)}\right),$$

noticing that the marginal likelihood $f(y)$ cancels top and bottom so we have

$$r = \min\left(1, \frac{f(y|\theta^*)}{f(y|\theta)} \times \frac{\pi(\theta^*)}{\pi(\theta)} \times \frac{q(\theta|\theta^*)}{q(\theta^*|\theta)}\right). \tag{2.3}$$

If the candidate point (θ^*) is accepted, the current state becomes $\theta^{(i)} = \theta^*$. If the candidate is rejected, the chain does not move (i.e. $\theta^{(i)} = \theta^{(i-1)}$).

The acceptance ratio (r) is made up of the **likelihood ratio** $\frac{f(y|\theta^*)}{f(y|\theta)}$, the **prior ratio** $\frac{\pi(\theta^*)}{\pi(\theta)}$, and the **proposal ratio** $\frac{q(\theta|\theta^*)}{q(\theta^*|\theta)}$. Often we can simplify the acceptance ratio (which can reduce the computational workload during each iteration and greatly speed up MCMC).

If a symmetric proposal distribution ($q(\theta^*|\theta) = q(\theta|\theta^*)$, e.g. a multivariate normal) is used then the proposal ratio cancels top and bottom as well, thus further reducing the acceptance ratio to

$$r = \min\left(1, \frac{f(y|\theta^*)}{f(y|\theta)} \times \frac{\pi(\theta^*)}{\pi(\theta)}\right).$$

Often components of the likelihood or prior cancel top and bottom as well. Suppose for example we are using a gamma prior $\pi(\theta) \sim \mathcal{G}\text{a}(\alpha, \beta)$. We know the density of a gamma distribution to be

$$\pi(\theta) = \frac{\beta^\alpha}{\Gamma(\alpha)}\theta^{\alpha-1}e^{-\beta\theta},$$

therefore

$$\frac{\pi(\theta^*)}{\pi(\theta)} = \frac{\cancel{\frac{\beta^\alpha}{\Gamma(\alpha)}}(\theta^*)^{\alpha-1}e^{-\beta\theta^*}}{\cancel{\frac{\beta^\alpha}{\Gamma(\alpha)}}\theta^{\alpha-1}e^{-\beta\theta}} = \left(\frac{\theta^*}{\theta}\right)^{\alpha-1} e^{-\beta(\theta^*-\theta)}. \tag{2.4}$$

It also follows that

$$\pi\left(\theta^*\right) = \pi(\theta)\left(\frac{\theta^*}{\theta}\right)^{\alpha-1} e^{-\beta(\theta^*-x)}. \tag{2.5}$$

Therefore, it is well worth taking the time to consider how the computation of the acceptance rate (r) can be simplified.

To summarise, consider a MH MCMC sampler for a single parameter θ, observed data y, a prior distribution of our model parameter $\pi(\theta)$, a model that defines the likelihood of the data given the parameter $f(y|\theta)$, a proposal distribution, and an acceptance probability

$$r = \min\left(1, \frac{f(y|\theta^*)}{f(y|\theta^{(i-1)})} \times \frac{\pi(\theta^*)}{\pi(\theta^{(i-1)})} \times \frac{q_\theta(\theta^{(i-1)}|\theta^*)}{q_\theta(\theta^*|\theta^{(i-1)})}\right).$$

Pseudo code for such a sampler is as follows. First, set the random number generator seed, then initialise parameter $\theta^{(0)} \sim q_0\left(\theta\right)$. This initialised state

could be set at the maximum likelihood estimate, drawn from the prior or set at some sensible value. Then for $i = 1, 2, ...$

1. Propose $\theta^* \sim q_\theta\left(\theta^{(i)}|\theta^{(i-1)}\right)$.

2. Compute the acceptance probability (r).

3. Draw $u \sim \mathcal{U}(0, 1)$.

4. Accept θ^* if $u < r$ and set $\theta^{(i)} \leftarrow \theta^*$, otherwise reject θ^* and set $\theta^{(i)} \leftarrow \theta^{(i-1)}$.

2.4.2 Blockwise MCMC

Usually, the target distribution is for a multivariate parameter $\boldsymbol{\theta}$, in which case it must be determined whether it is best to sample from separate components of $\boldsymbol{\theta}$ individually, in groups, or all at once. Block updating refers to splitting a multivariate vector $\boldsymbol{\theta}$ into groups called blocks, and each block is sampled separately. A block may contain one or more parameters. One advantage of blockwise sampling over multivariate sampling is that a different MCMC algorithm may be used for each block, or parameter, creating a more specialised approach. Furthermore, the acceptance of candidate proposals is likely to be higher than sampling from the full joint distribution at once in high dimensions as the variance of proposal distributions can be tuned for each component separately. This also avoids the need to provide a variance covariance matrix that usually requires model fitting using maximum likelihood first.

In a Metropolis-Hastings MCMC suppose we partition $\boldsymbol{\theta}$ into two pieces $\boldsymbol{\theta} = \{\theta_1, \boldsymbol{\theta}_{-1}\}$ where θ_1 is a single parameter and $\boldsymbol{\theta}_{-1}$ is the vector of the remaining parameters. Consider the process of updating just θ_1, we can

state that $\boldsymbol{\theta}^* = \{\theta_1^*, \boldsymbol{\theta}_{-1}^*\} = \{\theta_1^*, \boldsymbol{\theta}_{-1}\}$. Thus the acceptance ratio becomes

$$\begin{aligned} r &= \min\left(1, \frac{f(y|\boldsymbol{\theta}^*)}{f(y|\boldsymbol{\theta})} \times \frac{\pi(\boldsymbol{\theta}^*)}{\pi(\boldsymbol{\theta})} \times \frac{q(\boldsymbol{\theta}|\boldsymbol{\theta}^*)}{q(\boldsymbol{\theta}^*|\boldsymbol{\theta})}\right) \\ &= \min\left(1, \frac{f(y|\theta_1^*, \boldsymbol{\theta}_{-1})}{f(y|\theta_1, \boldsymbol{\theta}_{-1})} \times \frac{\pi(\theta_1^*|\boldsymbol{\theta}_{-1})\cancel{\pi(\boldsymbol{\theta}_{-1})}}{\pi(\theta_1|\boldsymbol{\theta}_{-1})\cancel{\pi(\boldsymbol{\theta}_{-1})}} \times \frac{q(\theta_1|\theta_1^*, \boldsymbol{\theta}_{-1})\cancel{q(\boldsymbol{\theta}_{-1}|\boldsymbol{\theta}_{-1})}}{q(\theta_1^*|\theta_1, \boldsymbol{\theta}_{-1})\cancel{q(\boldsymbol{\theta}_{-1}|\boldsymbol{\theta}_{-1})}}\right) \\ &= \min\left(1, \frac{f(y|\theta_1^*, \boldsymbol{\theta}_{-1})}{f(y|\theta_1, \boldsymbol{\theta}_{-1})} \times \frac{\pi(\theta_1^*|\boldsymbol{\theta}_{-1})}{\pi(\theta_1|\boldsymbol{\theta}_{-1})} \times \frac{q(\theta_1|\theta_1^*, \boldsymbol{\theta}_{-1})}{q(\theta_1^*|\theta_1, \boldsymbol{\theta}_{-1})}\right). \end{aligned} \tag{2.6}$$

Thus, we can see that the acceptance ratio for each parameter can be simplified. We may even be able to take this a little further if we can split $f(y|\theta_1, \boldsymbol{\theta}_{-1})$, $f(\theta_1|\boldsymbol{\theta}_{-1})$ or $q(\theta_1^*|\theta_1, \boldsymbol{\theta}_{-1})$ up into smaller parts, for example

$$r = \min\left(1, \frac{g(y|\theta_1^*, \boldsymbol{\theta}_{-1})\cancel{h(y|\boldsymbol{\theta}_{-1})}}{g(y|\theta_1, \boldsymbol{\theta}_{-1})\cancel{h(y|\boldsymbol{\theta}_{-1})}} \times \frac{\pi(\theta^*|\boldsymbol{\theta}_{-1})}{\pi(\theta|\boldsymbol{\theta}_{-1})} \times \frac{\cancel{c(\boldsymbol{\theta}_{-1})}d(\theta_1|\theta_1^*, \boldsymbol{\theta}_{-1})}{\cancel{c(\boldsymbol{\theta}_{-1})}d(\theta_1^*|\theta_1, \boldsymbol{\theta}_{-1})}\right). \tag{2.7}$$

Doing so can result in significant improvements in speed of our MCMC sampler.

2.4.3 Blockwise MH with log-normal proposals

Here we provide an example of a blockwise MH MCMC algorithm that uses a log-normal proposal distribution. Given a scalar parameter θ and data y, priors for each of the model parameters $\pi(\theta)$ and a model that defines the likelihood of the data given the parameters $f(y|\theta)$ we can derive the acceptance probability

$$\begin{aligned} r &= \min\left(1, \frac{f(y|\theta^*)}{f(y|\theta^{(i-1)})} \times \frac{\pi(\theta^*)}{\pi(\theta^{(i-1)})} \times \frac{q_\theta(\theta^{(i-1)}|\theta^*, y)}{q_\theta(\theta^*|\theta^{(i-1)}, y)}\right) \\ &= \min\left(1, \frac{f(y|\theta^*)}{f(y|\theta^{(i-1)})} \times \frac{\pi(\theta^*)}{\pi(\theta^{(i-1)})} \times \frac{\theta^*}{\theta^{(i-1)}}\right). \end{aligned}$$

Notice that the proposal ratio is simplified to $\frac{\theta^*}{\theta^{(i-1)}}$ (see Appendix A.3, page 325, for the proof). We then initialise the parameter value $\theta^{(0)} \sim q_0(\theta)$. Because we are using a log-normal proposal distribution, we must also specify a proposal variance σ_q^2. Then for $i = 1, 2, ...$

1. Propose $\theta^* \sim q_\theta\left(\theta^{(i)}|\theta^{(i-1)}\right)$. Here we draw $\log(\theta^*) \sim \mathcal{N}\left(\log\left(\theta^{(i-1)}\right), \sigma_q^2\right)$. See Appendix A.1 for the PDF of a log-normal.

2. Compute the acceptance probability (r).
3. Draw $u \sim \mathcal{U}(0,1)$.
4. Accept θ^* if $u < r$ and set $\theta^{(i)} \leftarrow \theta^*$, otherwise reject θ^* and set $\theta^{(i)} \leftarrow \theta^{(i-1)}$.

A log-normal proposal distribution is a good choice for parameters that are constrained to be positive numbers. It is also convenient as it simplifies calculation of the acceptance ratio.

2.4.4 Transformations of random variables

Say we have a two parameter model $p(y|\theta_1, \theta_2)$. We can express the joint density as

$$p(\theta_1, \theta_2) = p(\theta_1|\theta_2)p(\theta_2).$$

The posterior distribution conditional on data y is

$$\begin{aligned} p(\theta_1, \theta_2|y) &= \frac{p(y, \theta_1, \theta_2)}{p(y)} = \frac{p(y|\theta_1, \theta_2)p(\theta_1, \theta_2)}{p(y)} \\ &= \frac{p(y|\theta_1, \theta_2)p(\theta_1|\theta_2)p(\theta_2)}{p(y)} \\ &\propto p(y|\theta_1, \theta_2)p(\theta_1|\theta_2)p(\theta_2). \end{aligned}$$

If using an MCMC sampler with proposal distribution $q(\theta_1^*, \theta_2^*|\theta_1, \theta_2, y)$, we calculate the acceptance probability as

$$\begin{aligned} r &= \min\left(1, \frac{p(\theta_1^*, \theta_2^*|y)}{p(\theta_1, \theta_2|y)} \times \frac{q(\theta_1, \theta_2|\theta_1^*, \theta_2^*, y)}{q(\theta_1^*, \theta_2^*|\theta_1, \theta_2, y)}\right) \\ &= \min\left(1, \frac{p(y|\theta_1^*, \theta_2^*)}{p(y|\theta_1, \theta_2)} \times \frac{p(\theta_1^*, \theta_2^*)}{p(\theta_1, \theta_2)} \times \frac{q(\theta_1, \theta_2|\theta_1^*, \theta_2^*, y)}{q(\theta_1^*, \theta_2^*|\theta_1, \theta_2, y)}\right). \end{aligned}$$

But if we prefer to work with the transformed variables ϕ_1 and ϕ_2 where

$$\phi_1 = \phi_1(\theta_1, \theta_2) \qquad \text{and} \qquad \phi_2 = \phi_2(\theta_1, \theta_2),$$

then we need to include a Jacobian to adjust for the transformation

$$\begin{aligned} p_\theta(\theta_1, \theta_2|y) &= p_\phi(\phi_1, \phi_2|y)\left|\frac{\partial(\phi_1, \phi_2)}{\partial(\theta_1, \theta_2)}\right|, \\ q_\theta(\theta_1, \theta_2|\theta_1^*, \theta_2^*, y) &= q_\phi(\phi_1, \phi_2|\phi_1^*, \phi_2^*, y)\left|\frac{\partial(\phi_1, \phi_2)}{\partial(\theta_1, \theta_2)}\right|, \end{aligned}$$

therefore

$$r = \min\left(1, \frac{p\left(y|\theta_1^*, \theta_2^*\right)}{p\left(y|\theta_1, \theta_2\right)} \times \frac{p\left(\theta_1^*, \theta_2^*\right)}{p\left(\theta_1, \theta_2\right)} \times \frac{q_\phi\left(\phi_1, \phi_2|\phi_1^*, \phi_2^*, y\right) \times \left|\frac{\partial(\phi_1,\phi_2)}{\partial(\theta_1,\theta_2)}\right|_{\theta_1,\theta_2}}{q_\phi\left(\phi_1^*, \phi_2^*|\phi_1, \phi_2, y\right) \times \left|\frac{\partial(\phi_1,\phi_2)}{\partial(\theta_1,\theta_2)}\right|_{\theta_1^*,\theta_2^*}}\right).$$

2.4.5 Proposal variances

Setting the proposal variances (σ_q^2) that are used during MCMC is important as they affect the efficiency of an MCMC sampler. Setting the proposal variance parameters throughout this thesis was done iteratively in an initial adaptive phase prior to the burn-in phase. At each iteration of an MCMC in this initial phase the acceptance rate is checked. If the acceptance rate for a particular parameter is too low (below 15%) then the proposal variance for that parameter is decreased (by 5%). If the acceptance rate for a particular parameter is too high (above 50%) then the proposal variance for that parameter is increased (by 5%). This "adaptive" phase is run before the burn-in begins. Therefore, the MCMC's are split into three phases: an adaptive phase that tunes the proposal variances (σ_q^2), a burn-in phase and a sampling phase.

2.4.6 Parallel tempering

Parallel tempering can improve MCMC methods which perform poorly when they are used to simulate samples from complicated multi-modal distributions. The method can improve mixing of the MCMC algorithm leading to faster convergence of the sampling chain to the target distribution (i.e. the multi-modal distribution of interest).

The concept of tempering is similar to that of simulated annealing: when the posterior surface is flattened, it is easier to move around, while if it is sharpened, it gets harder to do so. Tempering a distribution usually just involves selecting an index value β and raising the density to the power of that value (i.e. $\pi^{(\beta)}(x) = c(\beta)\left(\pi(x)\right)^\beta$ for constant $c(\beta)$ or $\pi^{(\beta)}(x) \propto \left(\pi(x)\right)^\beta$). In Figure 2.1 we provide an example of a multi-modal distribution that

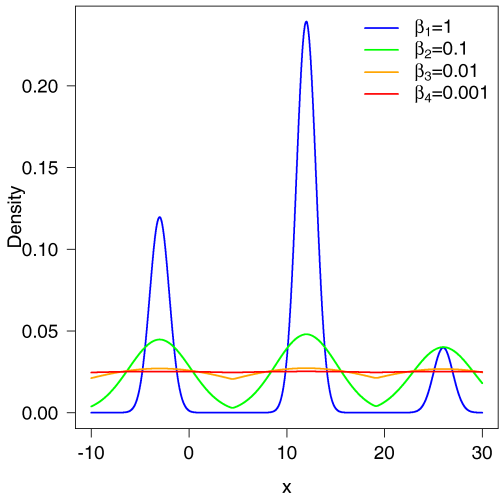


Figure 2.1: A multi-modal distribution that has been tempered with different powers of β. The powers of β used are given in the top right of the figure.

has been tempered using $\boldsymbol{\beta} = \{1, 0.1, 0.01, 0.001\}$. Parallel tempering is when several different MCMC chains are run simultaneously, randomly initialised, at different "temperatures". As the value of β decreases (i.e. the temperature increases), the distributions being explored by these parallel chains progressively flatten out making it easier for each of the MCMC chains to explore their likelihood surface as moves to areas of the state space that previously had a very low acceptance ratio now have higher probability of being accepted. Thus, higher temperature chains can explore wide areas of the state space, while cooler chains sample more precisely in local areas.

At each step of the chain, a swap is proposed between chains of high and low temperature making configurations at high temperatures available to the simulations at low temperatures and vice versa. Then, based on the Metropolis criterion, this swap can be accepted or rejected.

The parallel tempering algorithm simulates parallel Markov chains defined on tempered distributions $\pi_k(\boldsymbol{\theta}|\mathbf{y}) \propto \pi(\boldsymbol{\theta}|\mathbf{y})^{\beta_k}$, where $\{\beta_1, \beta_2, \ldots, \beta_k\}$ is a sequence of inverse temperature values and $\pi(\boldsymbol{\theta}|\mathbf{y})^{\beta_1} = \pi(\boldsymbol{\theta}|\mathbf{y})$ (i.e. $\beta_1 = 1$ meaning this chain is not tempered). The algorithm alternates be-

tween two different chain update steps: a **within chain move** and a **between chain swap**. The probability of a move from the current state of chain k ($\boldsymbol{\theta}^{(i-1)}$) to the proposed state ($\boldsymbol{\theta}^*$) is

$$\begin{aligned} r_k &= \min\left(1, \left(\frac{\cancel{c(\beta_k)}\pi(\mathbf{y}|\boldsymbol{\theta}_k^*)}{\cancel{c(\beta_k)}\pi(\mathbf{y}|\boldsymbol{\theta}_k^{(i-1)})}\right)^{\beta_k} \times \frac{q(\boldsymbol{\theta}_k^{(i-1)}|\boldsymbol{\theta}_k^*, \mathbf{y})}{q(\boldsymbol{\theta}_k^*|\boldsymbol{\theta}_k^{(i-1)}, \mathbf{y})}\right) \\ &= \min\left(1, \left(\frac{\pi(\mathbf{y}|\boldsymbol{\theta}_k^*)}{\pi(\mathbf{y}|\boldsymbol{\theta}_k^{(i-1)})}\right)^{\beta_k} \times \frac{q(\boldsymbol{\theta}_k^{(i-1)}|\boldsymbol{\theta}_k^*, \mathbf{y})}{q(\boldsymbol{\theta}_k^*|\boldsymbol{\theta}_k^{(i-1)}, \mathbf{y})}\right). \end{aligned} \tag{2.8}$$

The between chain swap attempts to swap the current values of two randomly selected chains, chains a and b. This proposed swap is accepted according to a symmetric Metropolis algorithm probability

$$r_s = \min\left(1, \left(\frac{\pi(\mathbf{y}|\boldsymbol{\theta}_b)}{\pi(\mathbf{y}|\boldsymbol{\theta}_a)}\right)^{\beta_a} \times \left(\frac{\pi(\mathbf{y}|\boldsymbol{\theta}_a)}{\pi(\mathbf{y}|\boldsymbol{\theta}_b)}\right)^{\beta_b}\right). \tag{2.9}$$

In summary, parallel tempering involves initialising K chains $\boldsymbol{\theta}_k^{(0)} \sim q_0(\boldsymbol{\theta})$. Then for $k = 1, \ldots, K$ inverse temperatures ($\{\beta_1, \beta_2, \ldots, \beta_K\}$) and $i = 1, 2, \ldots$

1. Propose $\boldsymbol{\theta}_k^* \sim q_\theta\left(\boldsymbol{\theta}_k^{(i)}|\boldsymbol{\theta}_k^{(i-1)}\right)$.
2. Compute the acceptance probability (r_k).
3. Draw $u \sim \mathcal{U}(0, 1)$.
4. Accept $\boldsymbol{\theta}_k^*$ if $u < r_k$ and set $\boldsymbol{\theta}_k^{(i)} \leftarrow \boldsymbol{\theta}_k^*$, otherwise reject $\boldsymbol{\theta}_k^*$ and set $\boldsymbol{\theta}_k^{(i)} \leftarrow \boldsymbol{\theta}_k^{(i-1)}$.
5. Randomly select any two chains a and b to exchange states.
6. Compute the acceptance probability (r_s).
7. Draw $u \sim \mathcal{U}(0, 1)$.
8. Accept the swap if $u < r_s$ and set $\boldsymbol{\theta}_a^{(i)} \leftarrow \boldsymbol{\theta}_b^{(i)}$ and $\boldsymbol{\theta}_b^{(i)} \leftarrow \boldsymbol{\theta}_a^{(i)}$, otherwise reject.

Return the values from the $k = 1$ (i.e. $\beta = 1$) chain.

2.5 Bayesian emulation

Consider a complex computer model which computes a scalar output y from a vector of inputs $\boldsymbol{\theta}$, thus $y = f(\boldsymbol{\theta})$. $f(\cdot)$ is called the simulator. It is common that one or more values of the inputs $\boldsymbol{\theta}$ are uncertain or unknown. It is also common, particularly in fisheries science, for $f(\boldsymbol{\theta})$ to take a long time to evaluate making standard Bayesian inference methods prohibitive. Bayesian emulation is an alternative approach that may be used for inference of computationally expensive models.

The emulator is a stochastic representation of the simulator $f(\boldsymbol{\theta}_i)$ conditioned on evaluations of the simulator at known inputs $\boldsymbol{\theta}_i$. The emulator allows us to interpolate or extrapolate the evaluations of $f(\boldsymbol{\theta})$ to beliefs about the simulator response for any input (Goldstein & Rougier 2006).

The key requirement is that $f(\cdot)$ be a smooth function, so that, if we know the value of $f(\boldsymbol{\theta})$, we should have some idea about the value of $f(\boldsymbol{\theta}')$ for an unknown set of parameters $\boldsymbol{\theta}'$ close to $\boldsymbol{\theta}$. However, we relax this requirement in Chapter 7 while trying to emulate complex stochastic fisheries models.

Bayesian emulation is an alternative to approximate Bayesian computation (ABC). In the ABC rejection algorithm, a set of parameter points is first sampled from the prior. Given a sampled parameter point θ, a data set $\widehat{y}$ is then simulated using the statistical model $\pi(y|\theta)$. If the generated $\widehat{y}$ is too different from the observed y^o, the sampled parameter values is discarded. Thus, $\widehat{y}$ is accepted with tolerance $\epsilon \geq 0$ if $\rho(\widehat{y}, y) \leq \epsilon$ where the distance measure $\rho(\widehat{y}, y)$ determines the level of discrepancy between $\widehat{y}$ and y based on a given metric (e.g. the Euclidean distance).

We cover Bayesian emulation in more detail later in Chapter 7 (page 259). We do not consider ABC further.

Chapter 3

Case studies

Three species are used as case studies in this project. This chapter provides background information on these species.

3.1 Antarctic toothfish

Antarctic toothfish (*Dissostichus mawsoni*, TOT, hereafter referred to as toothfish, Figure 3.1) are large Nototheniids native to the Southern Ocean. They can grow to be more than 2m in length, weighing over 100kg, and can live for up to 50 years of age. The exploratory toothfish fishery in the Ross Sea region began in 1997 (Figure 3.2) and is managed by the Convention for the Conservation of Antarctic Marine Living Resources (CCAMLR). Little or no fishing took place in the region before then. The fishery continues to this day with vessels returning to fish the area each summer. The fish are not accessible for the remainder of the year because an ice sheet forms over much of the area during the colder months. Since its beginning, the fishery has increased to about 3000 tonnes per annum.

Smaller fish are generally caught in the south and larger fish in the north. Hanchet et al. (2008) describe a hypothetical life cycle of Antarctic toothfish in the Ross Sea where sub-adult fish gradually move from the relatively shallow shelf to the deeper waters of the continental slope as

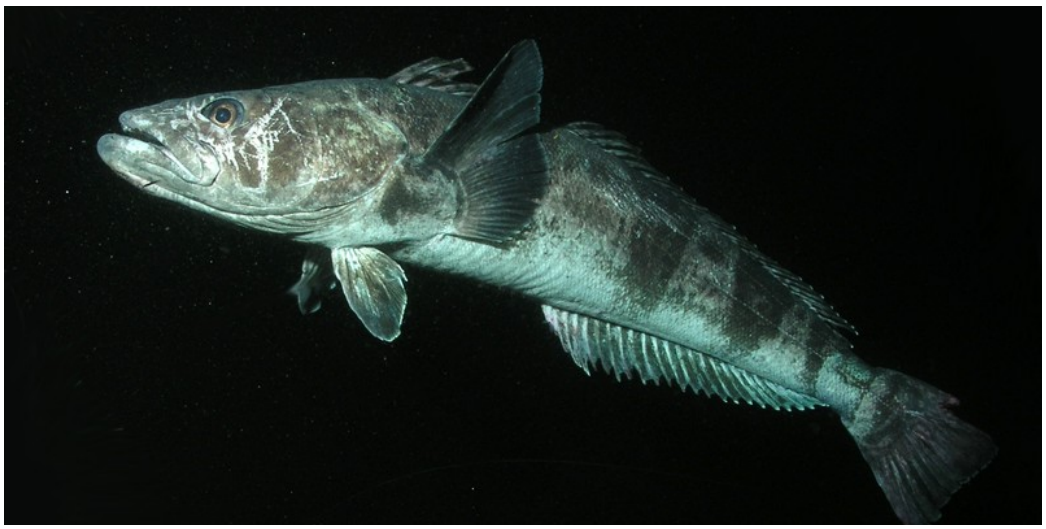


Figure 3.1: Antarctic toothfish (*Dissostichus mawsoni*).

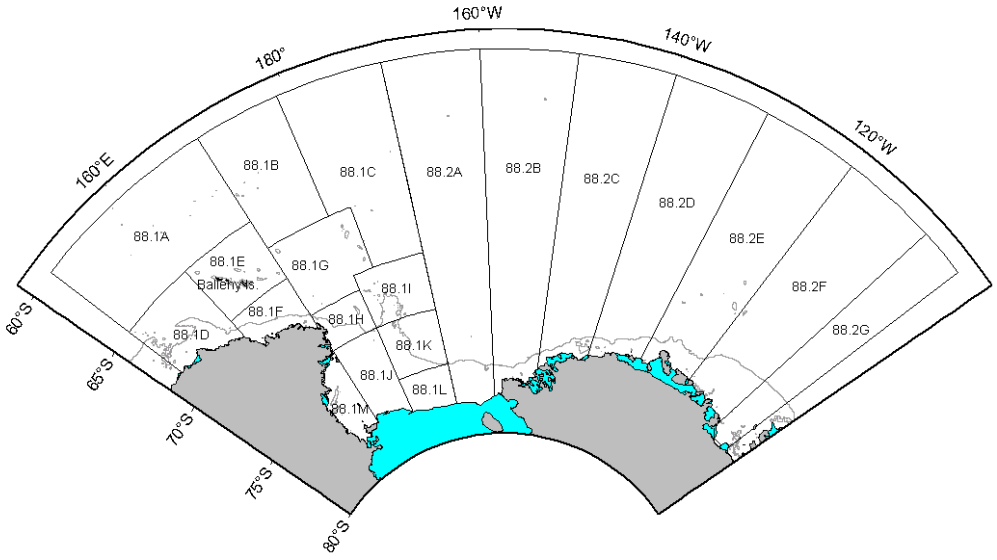


Figure 3.2: The Ross Sea region showing the CCAMLR fisheries management areas. Depth contour plotted at 1000m.

they become adults, with spawning migrations to and from seamounts to the North.

The Antarctic toothfish tagging programme was initiated in the Ross Sea region in the 2001 fishing season by New Zealand vessels involved in the fishery. In 2004, toothfish tagging was made compulsory for all vessels participating in the fishery. Currently toothfish are required to be double tagged at a rate of 1 fish per tonne landed. The programme also records information on the date, depth, location, sex (of recaptured fish), and size of each tagged/recaptured fish. Overall, a total of 37,047 Antarctic toothfish have been reported released and 1903 recaptured. For more information see Parker et al. (2013).

Several sources of data useful for stock assessment are available for Antarctic toothfish in the Ross Sea region (Stevenson et al. 2011). These data include a catch history, tag-recapture data, age- and length-frequencies, and newly available pop-up satellite archival tagging data (PSAT, Parker, Webber & Arnold 2014). These are very good data given that the catch history extends back to the beginning of the fishery in 1997 and geo-referenced tag-release and tag-recapture observations are recorded over much of the fishery.

The current Ross Sea Antarctic toothfish stock assessment model used by CCAMLR is a Bayesian statistical catch at age model that includes sex in the partition and uses tag-recapture data to determine the stock abundance within the fishery (Mormede, Dunn & Hanchet 2013). The model assumes a single homogeneous area with three geographically defined fisheries (shelf, slope and north). This is a pseudo-area style model in which a single area is defined, but the catch is removed using the three concurrent fisheries, each with their own selectivities. The need to account for the catch using three fisheries separately in the model arises because the size composition of the catch taken from each area is different. The three areas (shelf, slope and north) broadly represent the recruitment and juvenile feeding grounds, the adult feeding grounds, and the spawning grounds of Antarctic toothfish, respectively.

A more complex spatially explicit population model has also been devel-

oped using Spatial Population Model (SPM, Dunn & Rasmussen 2009). SPM is a generalised spatially explicit age-structured population dynamics model that can model a range of population processes and spatial movement as a function of environment and space. It can model populations over one or two areas, as well as populations in many hundreds of areas. The toothfish version of this model (Dunn et al. 2009, Mormede, Dunn, Parker & Hanchet 2013) was developed to test ideas on migration and movement of toothfish and how these processes may manifest bias in stock assessment models (Mormede & Dunn 2013).

The outcome of these modelling efforts suggests that the Ross Sea stock of Antarctic toothfish was estimated to be at $B_{2013} = 74\%$ (i.e. the stock biomass in 2013 is estimated to be 74% of B_0). The model predicts that it is virtually certain ($> 99\%$) that the stock is above the long term target of $50\%B_0$ set by CCAMLR (Ministry for Primary Industries 2014*b*).

An understanding of the spawning movements and migrations in relation to spatial population structure is vital for the stock assessment and management of toothfish in the Ross Sea. The actual degree of connectivity, range of spawning destinations, level of return migration, and duration of the migration remain uncertain (Parker, Hanchet & Horn 2014).

Modelling the spatial movements of toothfish populations requires estimates of the routes, timing, and duration of movements of individuals, not simply demonstrating a link between geographical regions. This includes information on fish that may migrate to previously unfished areas. Other types of movement information, such as patterns in vertical movements, are also becoming important in stock assessment and in understanding the ecosystem role of toothfish, as their depth distribution affects which species they interact with both as predators and as prey.

Chapter 6 aims to shed light on patterns of movement in individual Antarctic toothfish using newly available data from PSAT's coupled with data on depth, temperature from oceanographic models, and magnetic field strength in a novel state-space modelling method.

3.2 Snapper (SNA 1)

The Australasian snapper or silver seabream (*Pagurus auratus*, SNA, hereafter referred to as snapper, Figure 3.3) is a species of porgie found in coastal waters of Philippines, Indonesia, China, Taiwan, Japan, New Zealand, and Australia. Although it is commonly known in New Zealand and Australia as snapper, it does not belong to the Lutjanidae family of snappers. The species is capable of living up to 70 years and almost growing to 20kg.

The snapper fishery is one of the largest and most valuable coastal fisheries in New Zealand. New Zealand snapper are thought to comprise

Figure 3.3: New Zealand snapper (*Pagurus auratus*).

either seven or eight biological stocks based on: the location of spawning and nursery grounds; differences in growth rates, age structure and recruitment strength; and the results of tagging studies. Here we consider the fisheries management area known as SNA 1 (Figure 3.4), an area split into three stocks (East Northland, Hauraki Gulf and Bay of Plenty). Tagging studies reveal that limited mixing occurs between the three SNA 1 biological stocks, with greatest exchange between the Bay of Plenty and

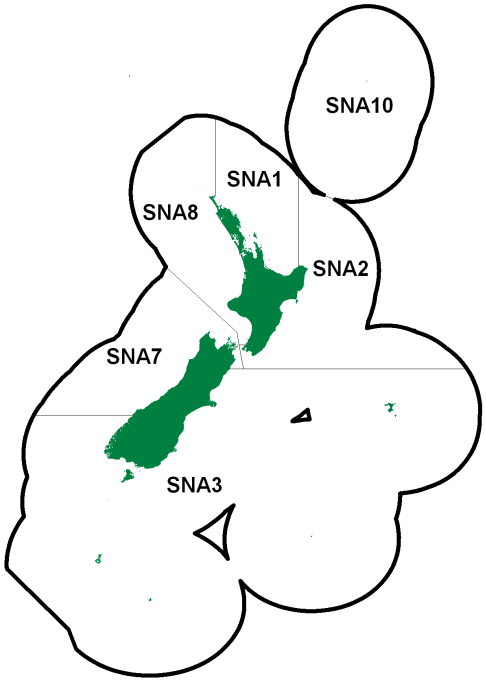


Figure 3.4: Snapper fisheries management areas.

the Hauraki Gulf.

The commercial fishery, which developed last century, expanded in the 1970s and peaked in 1978 (Figure 3.5). By the mid 1980s catches had de-

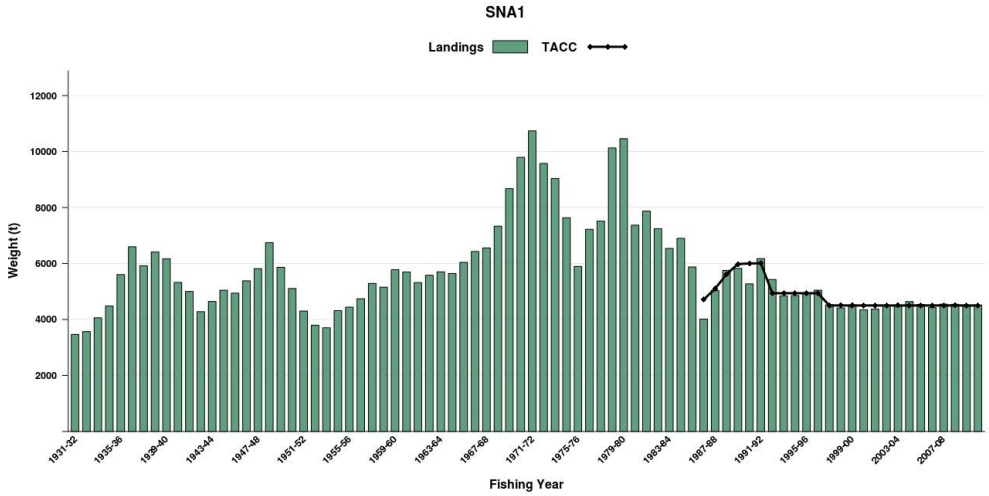


Figure 3.5: SNA 1 catch history showing the landings (tonnes) and Total Allowable Commercial Catch (TACC, tonnes) each fishing year.

clined. With the introduction of the QMS in 1986, the Total Allowable Commercial Catch (TACC) in all snapper stocks was set at levels intended to allow for some stock rebuilding. All commercial fisheries have a minimum legal size (MLS) for snapper of 25cm.

The data available that are useful for stock assessment include an incomplete catch history, CPUE indices, age- and length-frequency data, and tag-recapture data. The catch history in the earlier years of the fishery are uncertain. Tags were released into this fishery in 1983 and 1993.

The model used for the 2013 assessment (Francis & McKenzie 2013) was written using CASAL (Bull et al. 2012), a software package developed by NIWA for fish stock assessment. The software implements a generalised age-structured stock assessment model. The SNA 1 model is a three-stock, three-area model that covers the time period from 1900 to 2013 (i.e. fishing years 1899-1900 to 2012-13), with two time steps in each year. The assess-

ment modelled the movement of fish between areas and assumed home fidelity (HF) movement dynamics. Under the HF movement, fish spawn in their home area and some move to other areas at other times of the year where they are subject to fishing. There are two sets of migrations: in time step 1, all fish return to their home (i.e. spawning) area just before spawning; and in time step 2, fixed proportions of fish move away from their home area into another area.

The model partitions the modelled population by age (ages 1-20, where the last age is a plus group), stock (three stocks, corresponding to the parts of the population that spawn in each of three subareas of SNA 1), area (the three subareas), and tag status (grouping fish into six categories - one for untagged fish, and one each for each of five tag release episodes). That is to say, at any point in time, each fish in the modelled population would be associated with one cell in a $20 \times 3 \times 3 \times 6$ array, depending on its age, the stock it belonged to, the area it was currently in and its tag status at that time. The model does not distinguish fish by sex.

The SNA 1 stock is estimated to be low at $B_{2013} = 24\%$ in East Northland and with $B_{2013} = 19\%$ in the Hauraki Gulf and Bay of Plenty (these are percentages of B_0). It is likely that overfishing is occurring in SNA 1 (Ministry for Primary Industries 2014*a*).

An agent-based snapper model is developed in Chapter 4. This agent-based model is then used later to develop a Bayesian emulator for the snapper model in Chapter 7. The SNA 1 stock is thought to be a spatially complex stock. Due to recent changes in the recreational catch limits, the fishery has received much attention in the last couple of years. These factors motivated the use of snapper as a case study species in the development of agent-based models.

3.3 Packhorse rock lobster

Two species of rock lobster are commonly harvested in New Zealand inshore waters. The red rock lobster (*Jasus edwardsii*, CRA) and the pack-

horse rock lobster (*Sagmariasus verreauxi*, hereafter referred to as PHC, Figure 3.6). The red rock lobster supports the most valuable inshore fishery in

Figure 3.6: New Zealand packhorse rock lobster (*Sagmariasus verreauxi*, PHC).

New Zealand and is important to commercial, recreational and customary users. Conversely, few fishermen target the packhorse rock lobster. PHC are taken mainly in the north of the North Island and usually as bycatch of the red rock lobster fishery. However, there is some overlap of vessels targeting red rock lobsters and packhorse in the CRA 1 Quota Management Area (QMA, Figure 3.7) and when vessels do target PHC their catch rates are much higher (Webber 2013).

PHC is found in New Zealand and southern Australia, however, some provisional genetic evidence suggests that the variety found in Australia may be a different species (Brasher et al. 1992). While much work has been done on the lobsters in Australia, where they are commonly referred to as the eastern rock lobster, there is little available information on PHC in New Zealand. A noteworthy exception to this is John Booth's book on spiny lobsters which focuses on the packhorse rock lobster (Booth 2011). This book compiles much of the knowledge of the biology, ecology and history of the PHC fishery.

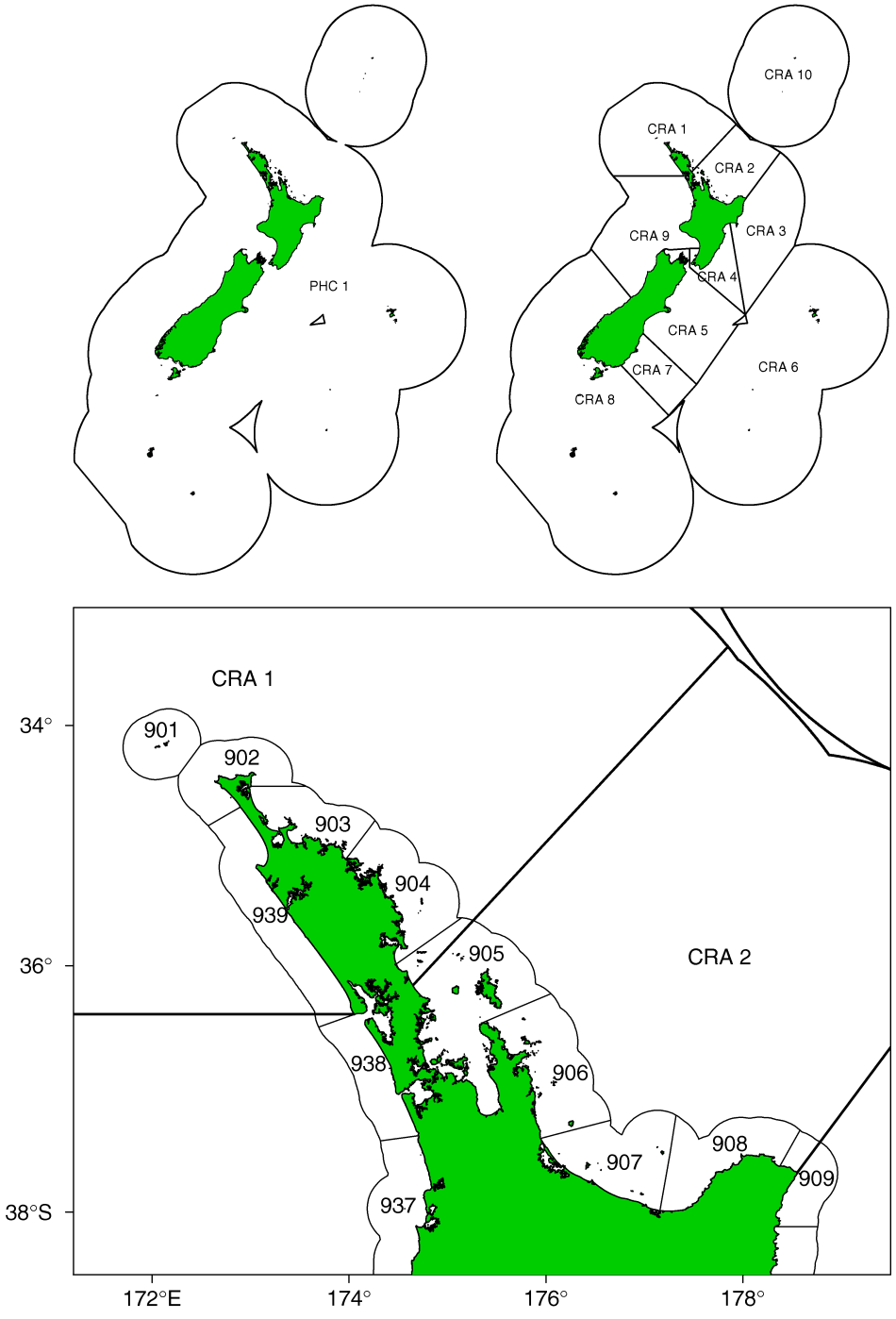


Figure 3.7: Rock lobster fisheries management areas for the packhorse rock lobster (*Sagmariasus verreauxi*) [top-left] and the red rock lobster (*Jasus edwardsii*) [top-right], and the rock lobster statistical areas in northern New Zealand within the CRA 1 and CRA 2 fisheries management areas [bottom].

PHC is reported to be the largest species of rock lobster in the world, reaching a total length of over half a metre and possibly weighing as much as 18kg. PHC in Australia are known to grow faster than red rock lobsters here in New Zealand (Montgomery et al. 2009), but little is known of the growth rates of packhorse in New Zealand. PHC in New Zealand are thought to be the most migratory spiny lobster on Earth in terms of proportions migrating and distance covered. They generally march northwards up the east coast of the North Island, the primary destination being breeding grounds in the far north of the North Island. Little is known of the timing or proportion of individuals that migrate northwards or what they do once they reach their destination and breed. The size at onset of breeding is large in females at about 160mm carapace-length (Booth 1984). Typical of many other rock lobster species, females and males move seasonally inshore and off as they respectively moult and reproduce.

Packhorse rock lobsters were amalgamated into the QMS on 1 April 1990 and the fishery is currently managed as a single QMA (PHC 1, Figure 3.7) by MPI. Initially the TACC was set at 27 tonnes. This was raised in the same year to 30.5 tonnes due to quota appeals (Ministry for Primary Industries 2014*b*). The TACC remained at 30.5 tonnes for just two years and was increased to 40.3 tonnes during the 1992/93 fishing year where it has remained since. Since the introduction of the QMS, packhorse catches have been relatively low and have only begun to approach the TACC in the last five years (Figure 3.8). Historically landings were much higher (Kensler & Skrzynski 1970, Booth 2011, Figure 3.8). While PHC 1 covers New Zealand's entire Exclusive Economic Zone (EEZ, Figure 3.7), PHC are caught mainly in the north of the North Island (Figure 3.9).

Prior to the QMS, the PHC fishery was managed using input controls, including: minimum legal size (MLS) regulations; a prohibition on taking berried females (i.e. carrying external eggs) and soft shell lobsters; making it illegal to commercially dive for or spear them; requiring pots to have gaps or mesh large enough to allow small lobsters to escape; requiring that they are landed ashore alive; and some local area closures. Today, all of these input controls still remain.

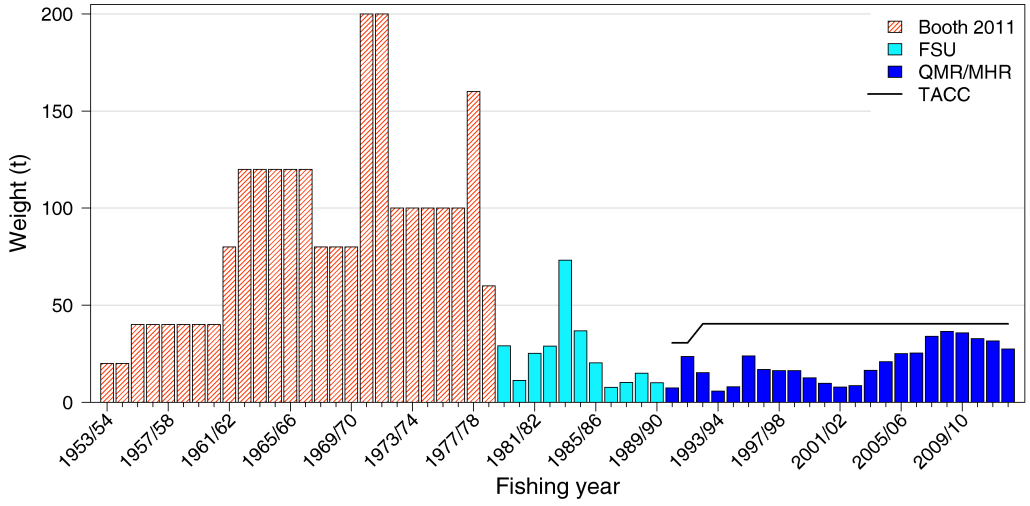


Figure 3.8: Packhorse rock lobster (*Sagmariasus verreauxi*) landings (tonnes) and Total Allowable Commercial Catch (TACC, tonnes) from 1953/54 to 2012/13. The landings for 1953/54 to 1978/79 are estimates from Booth (2011), for 1979/80 to 1989/90 are FSU data, and 1990/91 onwards are QMR/MHR data.

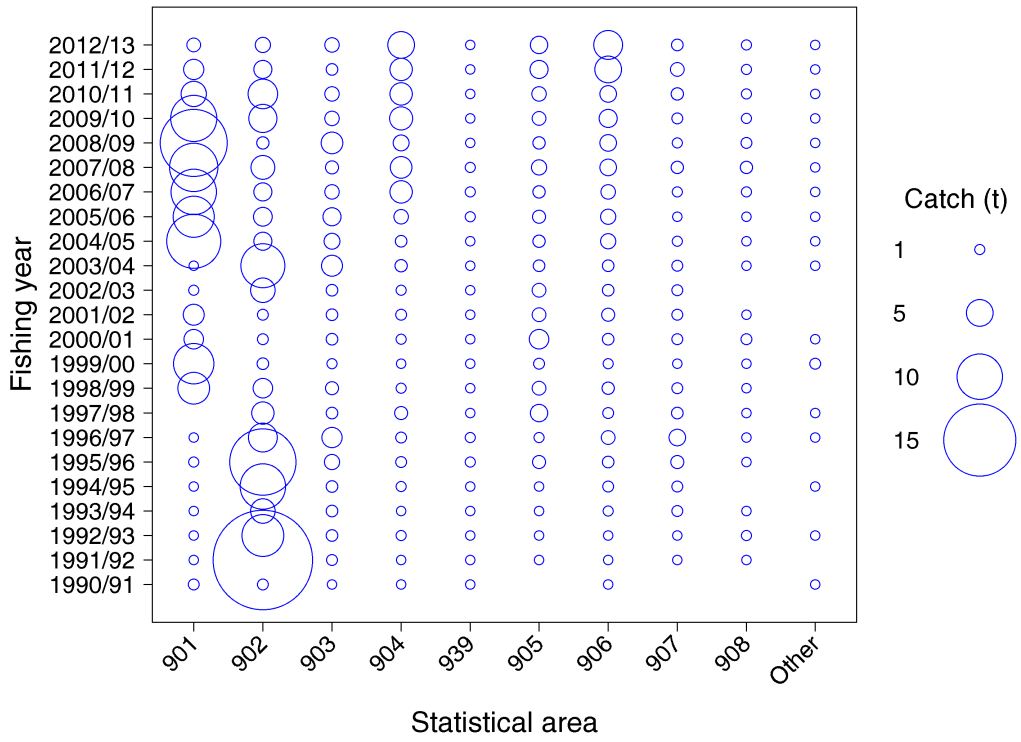


Figure 3.9: Packhorse rock lobster catch by statistical area and year. The location of these statistical areas is shown in Figure 3.7.

Fishers targeting PHC generally use different methods to those targeting CRA. Generally longer soak times and more bait are used (this is also the case in the Australian fisheries for PHC). Larger pots are also common when targeting the species, often due to strong tidal currents and the sandy substrate on which they are set.

Recreational fishers may gather PHC by pot or hand (freediving or scuba) but are subject to most of the other input controls in place for commercial fishers. Recreational fishers may not take soft shell individuals or berried females, and it is illegal to spear them. Additionally, a bag limit of 6 rock lobsters (CRA and PHC combined) applies to recreational fishers. The total recreational and customary catches for PHC are unknown.

The MLS of PHC for commercial and recreational fishers is 216mm tail-length for both sexes. In many fish stocks a MLS is put in place to allow most females to breed at least once before becoming legal to harvest. The current MLS of PHC is smaller than the size at which 50% of females reach maturity (Booth 1984). In comparison, the MLS in Australia is 104mm carapace-length, much smaller than New Zealand where for females it is equivalent to 155mm carapace-length. In Australia there is also a maximum legal size of 180mm carapace-length that is intended to look after the stock of large breeders.

Two sources of information useful for stock assessment are available for the packhorse rock lobsters in northern New Zealand. A catch history and a catch per unit effort (CPUE) index. Catch data (QMR/MHR), from 1979/80 to 2012/13, were obtained from MPI. Historical catch estimates were obtained from Booth (2011). The CPUE index and its development is described in Webber (2013) and shown in Figure 3.10. The standardised CPUE index is based on Catch Effort Landing Return (CELR) data from 1991/92 to 2012/13. The trend in the CPUE index shows a rise in catch rates since the start of the series (Figure 3.10).

Some tag recapture and length-frequency data are also available for this species but the sample sizes are very low. Only 12 lobsters have been recaptured during the late 1970s and the number of individuals measured range from 6 to 508 from the 1990/91 to 2012/13 fishing years. The

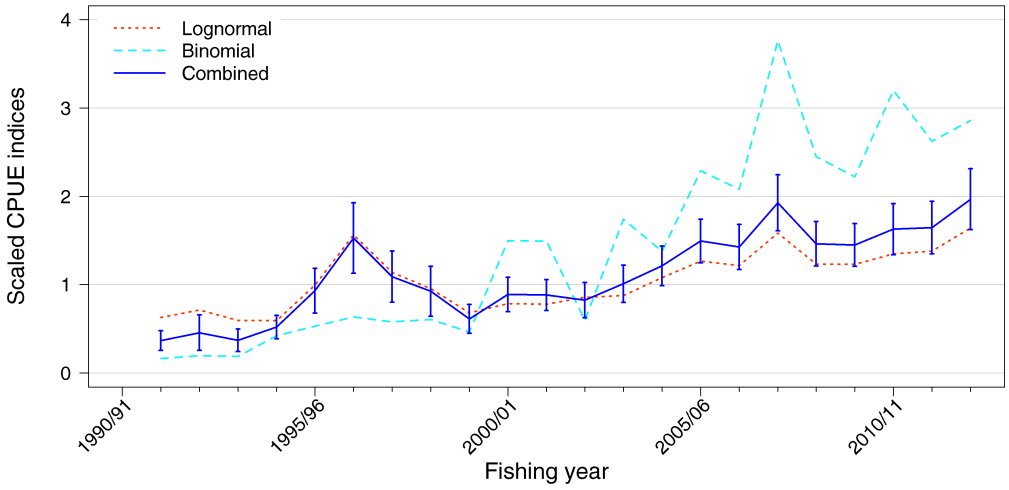


Figure 3.10: Standardised annual indices of CPUE (I_t) for log-normal, binomial and combined models for packhorse rock lobster *Sagmariasus verreauxi* (PHC) from 1991/92 to 2012/13. The combined model is shown $\pm$ 2 s.e. All indices are scaled to have a geometric mean of 1. Source: Webber (2013).

lack of these data and the limited knowledge on growth in PHC mean that a length-structured stock assessment is currently not possible for this species.

Packhorse rock lobster data are used in Chapter 5 as an example data set for state-space biomass dynamics models. To fit properly, stock assessment models require contrast in the catch history and/or CPUE indices. The contrast in the catch history and CPUE series motivated the use of PHC data as an example here.

Chapter 4

An agent-based simulation model

In this chapter we describe an agent-based simulation model. Throughout this chapter we use an example based on snapper in northern New Zealand (*Pagurus auratus*, SNA 1, for more information on snapper see Chapter 3, page 77) to illustrate how the model works. All of the variables defined in this model are given in Table 4.1.

4.1 Introduction

Individuals are the building blocks of ecological systems. The properties and behaviour of individuals determine the properties of the systems they comprise. Population-level properties emerge from the interactions of individuals with each other and with their environment (Grimm & Railsback 2005) and variation among individuals is increasingly recognized as fundamental to predictions about populations, communities, and ecosystems (Smallegange & Coulson 2012).

Individual-based models (IBMs) and agent-based models (ABMs) describe populations in which individuals, or small groups of individuals (super-individuals, Scheffer et al. 1995), differ in their growth, maturation, movement, and mortality and follow these individuals through their life history. They can be used to formulate theories about the behaviour of

Table 4.1: List of model parameters. SD is the standard deviation.

Symbol	Type	Dimensions	Description
f	scalar	1	Index of fisheries $f = \{0, 1, \ldots, F\}$
j	scalar	1	Index of stocks $j = \{0, 1, \ldots, J\}$
s	scalar	1	Index of sexes
y	scalar	1	Index of years $y = \{0, 1, \ldots, Y\}$
z	scalar	1	Index of areas $z = \{0, 1, \ldots, Z\}$
R_0, σ_R	vector	$J + 1$	Mean and SD of recruitment
h	vector	$J + 1$	Stock recruitment steepness
ρ	vector	$J + 1$	Recruitment autocorrelation
p_{male}	vector	$J + 1$	The proportion of recruits that are male
$\mathbf{\Xi}$	matrix	$J + 1 \times Z + 1$	The proportion of total recruitment from each stock into each area
$\Im$	vector	$Z + 1$	This defines the areas that make up a stock
$\mu_{A_{50}}$, $\sigma_{A_{50}}$	vector	s	Mean and SD of age at 50% maturity
$\mu_{A_{to95}}$, $\sigma_{A_{to95}}$	vector	s	Mean and SD of difference in age at 95% maturity
μ_L, σ_L	vector	s	Maturity
μ_R, σ_R	vector	s	Maturity
μ_{L_∞}, σ_{L_∞}	vector	s	Mean and SD of asymptotic length (cm)
μ_k, σ_k	vector	s	Brody growth coefficient (years^{-1})
t_0	vector	s	Time at which length is zero (years)
c_ℓ	vector	s	Coefficient of variation of growth
μ_α, σ_α	vector	s	Length-weight
μ_β, σ_β	vector	s	Length-weight
μ_M, σ_M	vector	s	Mean and SD of natural mortality (M)
μ_q, σ_q	vector	f	Mean and SD of the catchability coefficient (q)
γ_1	matrix	$s \times f$	Double normal selectivity
γ_L	matrix	$s \times f$	Double normal selectivity
γ_R	matrix	$s \times f$	Double normal selectivity
$\mathbf{\Omega}$	matrix	$Z + 1 \times Z + 1$	Migration matrix
ψ	scalar	1	Probability of home switching
σ_o	scalar	1	Observation error standard deviation

individuals. We can then test these theories by seeing how well they reproduce patterns observed at the system level.

Despite being a very flexible modelling approach, the computational overheads of IBMs and ABMs can be very high. This can make some models impractical, particularly if trying to estimate parameter values. However, new hybrid methods that tie classical models with individual-based aspects (e.g. Gray et al. 2006) and some modern computational methods may be the key to solving such dilemmas. The benefits of such approaches may be worthwhile in some cases. In species or populations with few individuals (e.g. marine mammals), such models make sense if the data is available as the computational overheads will be reasonably low and complex behavioral attributes may be incorporated. However, when modelling populations with many individuals (e.g. fish) then there must be a good reason to do so using ABMs rather than age-structured models.

Classic statistical age-structured stock assessment models track cohorts of fish though time and in some cases space. In a spatially explicit model, a proportion of fish within each cohort may move between different areas within the model in a time step. When fish move from one area to another, they are lumped together with other fish in the new area of the same age (and/or sex/maturity) in the partition. This means that we no longer know which fish came from where in the next time step. Now, consider that these fish may be moving to a certain area to spawn, a common occurrence in fish stocks. If these fish do move to a certain area to spawn, will they return to the same general area they came from, or will they move on to any suitable area within the model space? The former implies that we must know the area from which the fish came from before embarking on their spawning migration.

There is evidence in some fish populations (e.g. Antarctic toothfish and New Zealand snapper, see Chapter 3, pages 73 and 77) that individuals may return to the area that they originally came from, a phenomenon known as site or home fidelity. If site fidelity is operating within a population, then this will inflate the recapture probabilities of individuals within a population in tag-recapture experiments (resulting in the model under-

estimating population size). To account for this, it may be necessary to track tagged fish as individuals, or agents, and store information on their home site which would adjust our recapture probabilities. Site or home fidelity could also affect other processes such as the spawning stock biomass (SSB) of a population. For example, if all fish return home to spawn then the SSB in the home area could be different than if fish simply stayed where they were to spawn. To test these ideas, a fish population could be simulated in a way that the individuals (or groups of individuals) within the population display site fidelity.

This chapter describes a spatially explicit multi-generational agent-structured fish simulation model that allows flexibility in specifying population and spatial dynamics. The model has the potential to consider individual variability, individual movement, and spatial heterogeneity in the environment. The aim is to construct a model that is sufficiently rich that it can be used to simulate complete, realistic fish populations. The simulated data can be used to test stock assessment methodologies - which are usually based on samples from the population, and incomplete data.

4.2 The agent

This model was developed to track agents which are collections of individual fish. The advantage of an agent based model is that all fish in an agent have identical properties, and can therefore be treated simultaneously. An agent can contain a single fish (individual-based), any number of fish, or an entire cohort of fish (all fish of the same age in the population). Each agent i contains all of the attributes of the fish in that agent. The attributes of an agent include how many individuals an agent contains f_i, biological information such as the age (a_i), sex (s_i), length (ℓ_i), and maturity (m_i) and information on the location of agents (z_i), the home site of the fish in the agent (h_i) if the fish in the agent are tagged or not (t_i), and the year that the fish was tagged (y_i) (Table 4.2). The areas in the model are labelled as $z = 0, \ldots, Z$ where $Z + 1$ is the total number of areas in the model. The years in the model, after initialisation, are labelled as $y = 0, \ldots, Y$ where

$Y + 1$ is the total number of years. Agents also store a set of parameters

Table 4.2: The attributes of an agent i including how many individuals an agent contains (f_i), biological information such as the age (a_i), sex (s_i), length (ℓ_i), and maturity (m_i) and information on the location of agents (z_i), the home site of the fish in the agent (h_i) if the fish in the agent are tagged or not (t_i), and the year that the fish received a tag (y_i). The number of areas in the model is $Z + 1$ and the number of years in the model is $Y + 1$. Agents also store parameters specific to that agent including growth parameters ($L_{\infty,i}$ and k_i), length-weight (α_i and β_i), maturity ($A_{50,i}$, $A_{to95,i}$, L_i and R_i) and natural mortality (M_i).

Attribute	Symbol	C++ type	Possible values
Frequency	f_i	int	$0 \leq f_i \leq F_{\text{max}}$
Age	a_i	int	$a_{\text{min}} \leq a_i \leq \infty$
Sex	s_i	bool	$0, 1$
Length	ℓ_i	double	-
Maturity	m_i	bool	$0, 1$
Location	z_i	int	$0 \leq z_i \leq Z$
Home	h_i	int	$0 \leq h_i \leq Z$
Tagged	t_i	int	$-1 \leq t_i \leq Z$
Tag year	y_i	int	$-1 \leq y_i \leq Y$
Length	$L_{\infty,i}$	double	-
Length	k_i	double	-
Length-weight	α_i	double	-
Length-weight	β_i	double	-
Maturity	$A_{50,i}$	double	-
Maturity	$A_{to95,i}$	double	-
Maturity	L_i	double	-
Maturity	R_i	double	-
Natural mortality	M_i	double	-

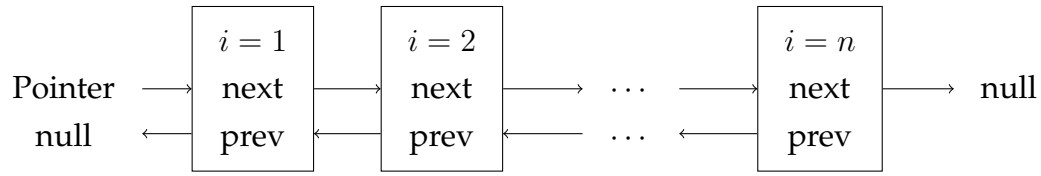
specific to the fish in that agent, and a set of pointers to other agents in the population. These pointers are further described in Section 4.2.1 below.

Agents are created via recruitment or if an already extant agent splits. At

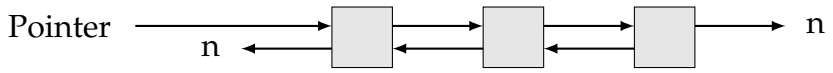
recruitment the value of some agent attributes are set to -1 (i.e. $t_i = -1$ and $y_i = -1$), this simply indicates that the agent has not been tagged yet and helps differentiate between an agent tagged in location $z_i = 0$ or during year $y_i = 0$. Agents can move randomly via migration or deterministically via home fidelity. Agents are either merged when they get too small or destroyed when an agent becomes depleted to $f_i = 0$ through natural mortality, fishing and/or splitting (see Section 4.2.4, page 102).

4.2.1 Accessing agents

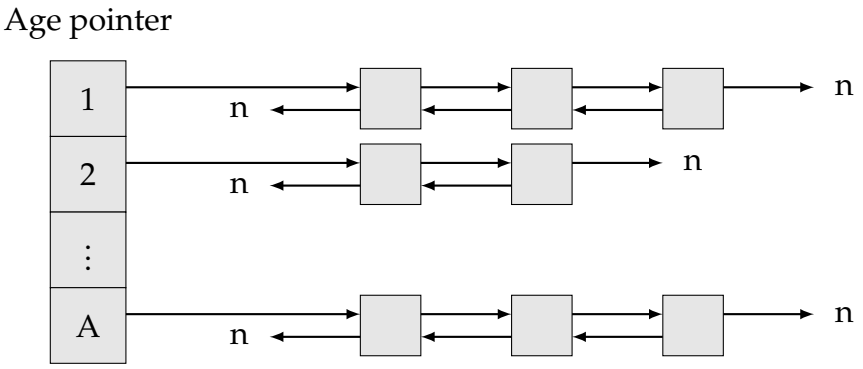
In this model the population is made up of a collection of agents. The agents are placed in sequences ordered by unique paths (mapped by sets of pointers) that can be used to traverse sequences of agents. One path traverses the entire population and is used to sequentially access all of the agents in the population and modify the content of each agent along the way (e.g. applying natural mortality) or gather information about the population (e.g. summing up the weight of all of the individuals in the population to get the biomass). We represent this sequence of agents graphically as

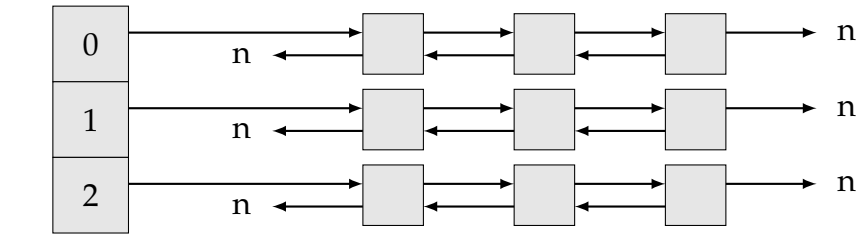


where each of the boxes represent a single agent containing all of the information outlined in Table 4.2 as well as the pointers used to traverse the sequence. The first agent in the sequence is accessed using a stored pointer to that agent (represented as "Pointer" here). The arrows represent pointers to other agents in the sequence, these can point to the next ("next") agent or the previous ("prev") agent in a sequence. The last agent in the sequence of agents points next to "null". Thus, traversing a sequence of agents in the code is stopped once the pointer to the next agent becomes null. Similarly, the first agent in a sequence points backwards to null. We simplify this diagram to



where the "n" represents "null". Other paths exist that access the agents by age and by location. The population can be traversed by whichever path is most relevant for the operations at hand. This is also useful as tasks can be passed off as different threads by age or by area. On a multi-core computer this can improve the time the model takes to run. For example, the biomass can be summed up by area as different threads, then the separate results for each area can be summed to get the total biomass. Graphical representations of the age- and area-sequences are shown below. These sequences are accessed by storing a vector of pointers to the first age or area in a sequence.



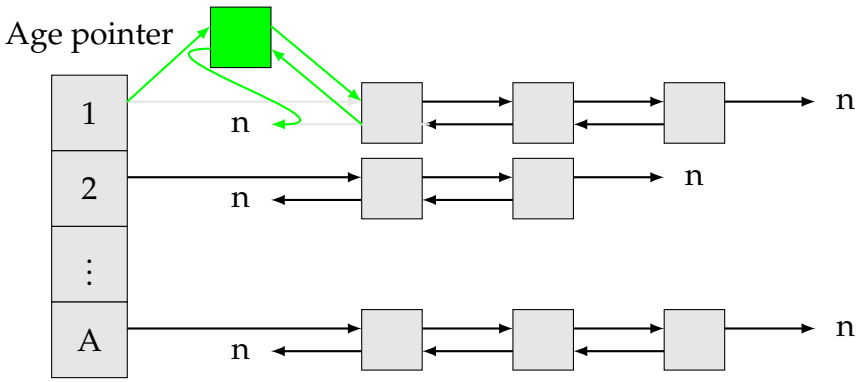


Arranging the population of agents as strings of objects connected by pointers, rather than a set of matrices and vectors holding all of the information about the population, is done to reduce the memory requirements of the model. A model structured in the standard way, using matrices and vectors, requires that memory be allocated for the entire object (matrix or vector) even if many of the values within that object are unused (i.e. zero fish are associated with many values in each object). This model only al-

locates memory if there are fish within an agent, otherwise that agent is destroyed and the memory for that agent freed.

4.2.2 Creating agents

New agents are created by recruitment (or splitting agents, Section 4.2.4, page 102). When an agent recruits to a population it is added to the beginning of the minimum age sequence. In doing so, several pointers require redirection shown graphically below



where the new agent is green, as are the redirected pointers.

When a new agent is created via recruitment that agent requires initialisation with agent specific parameter values, age, sex, length, location and so on. The values these variables take are described below.

Agent-specific parameters

Several parameters are allocated to agents when they are recruited to the population including parameters relating to growth ($L_{\infty,i}$ and k_i), the length-weight relationship (α_i and β_i), maturity ($A_{50,i}$, $A_{to95,i}$, L_i and R_i) and natural mortality (M_i). In the model, each of these parameters have their own mean and standard deviation. When an agent is created the value of each of these parameters for each agent is drawn from a normal distribution (e.g. Figure 4.1). The parameter t_0 is not stored within an agent because it is only required at the time the agent is created (see Table 4.1 and Section 4.2.2 above).

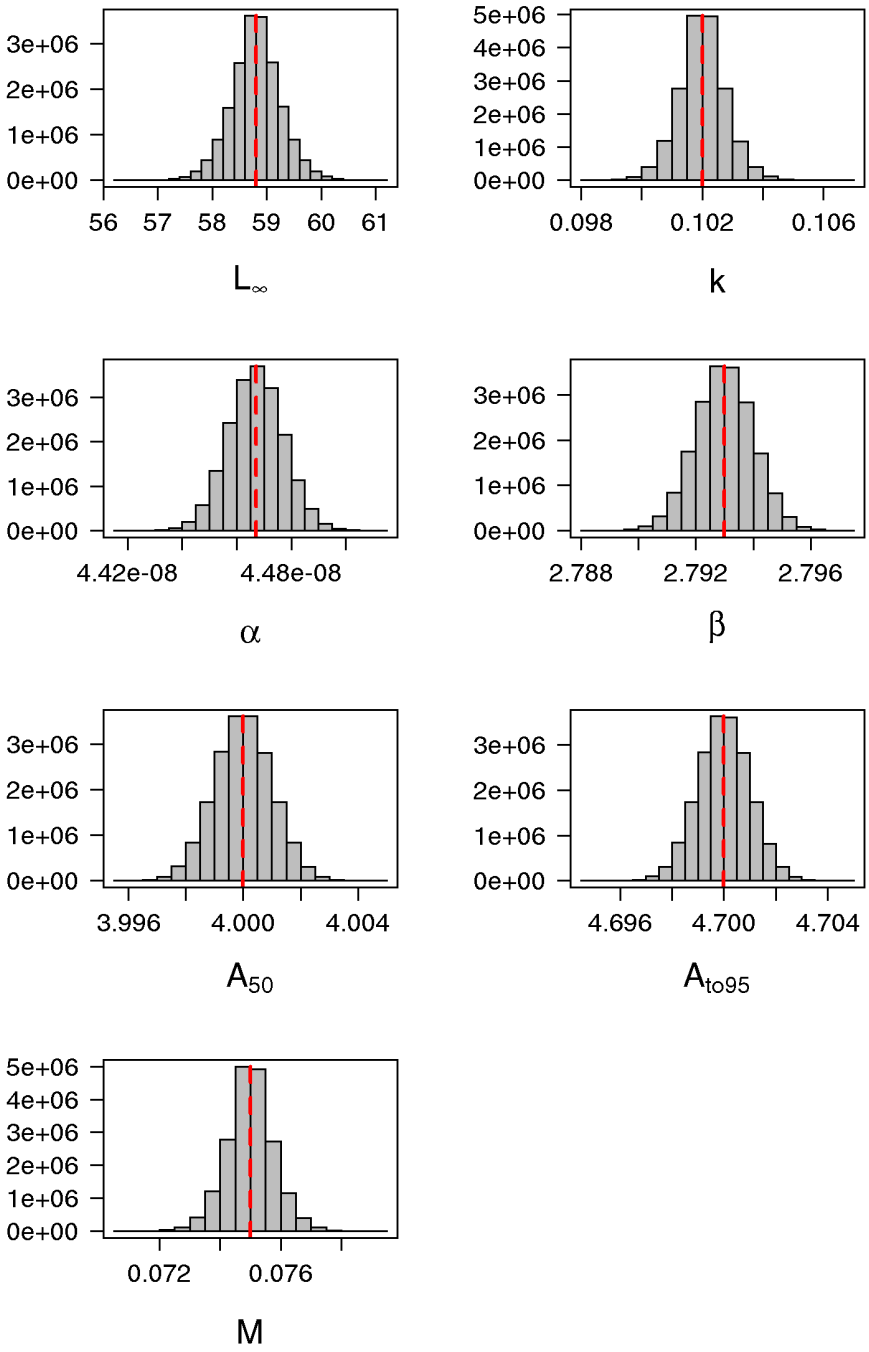


Figure 4.1: Frequency histograms of the parameters allocated to agents upon their creation. The values plotted are from the population after initialisation but before fishing begins (i.e. the end of phase 2).

Frequency ($f_i = \{0, 1, \ldots, F_{\mathbf{max}}\}$)

The frequency or number of individuals contained in an agent is specified at recruitment. This is further described in Section 4.4.5, page 115.

Age ($a_i = \{a_{\mathbf{min}}, \ldots, A\}$)

The age of recruited individuals within each agent is

$$a_i = a_{\text{min}},$$

where a_{min} is the specified minimum age (years) within the model. For more information see Chapter 1, page 17. Ageing is described later in this chapter in Section 4.4.1, page 110.

Sex ($s_i = \{0, 1\}$)

All of the individuals within an agent are either female ($s_i = 0$) or male ($s_i = 1$). Sex is randomly allocated at recruitment using a binomial distribution (Equation 4.17). The sex of all individuals within an agent is determined at recruitment and does not change. The model may be run as a single sex model. For more information see Chapter 1, page 18.

Length (ℓ_i)

Growth in the model can be either deterministic or stochastic. If growth is set to be deterministic then the initial length ℓ_i of recruits within agent i is equal to the mean length $\mu_{\ell,i}$ of a fish at a_{min} in agent i. This is determined using the von Bertalanffy growth function

$$\mu_{\ell,i} = L_{\infty,i}\left(1 - e^{-k_i(a_{\text{min}} - t_0)}\right), \tag{4.1}$$

where $L_{\infty,i}$, k_i, and t_0 are the parameters of the von Bertalanffy growth function (see Chapter 1, page 19). $L_{\infty,i}$ and k_i are agent specific parameters, t_0 is not. If growth is set to be stochastic then the initial length of new

recruits is drawn randomly from a normal distribution

$$\ell_i \sim \mathcal{N}\left(\mu_{\ell,i}, \sigma^2\right) \qquad \text{where} \qquad \sigma = c_\ell \mu_{\ell,i}, \tag{4.2}$$

and c_ℓ is the coefficient of variation of growth (e.g. Figure 4.2). Growth is described later in this chapter in Section 4.4.2, page 110.

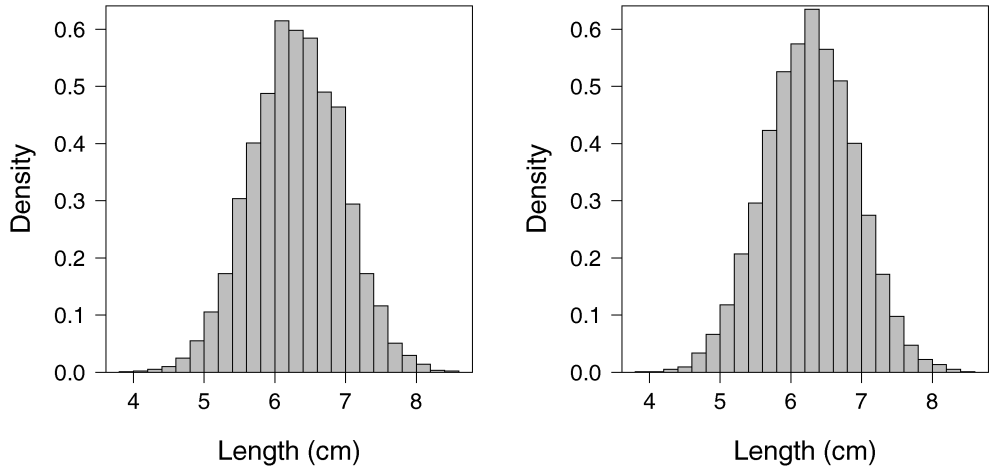


Figure 4.2: The initial length (cm) of females [left] and males [right] allocated to agents during initialisation.

Maturity ($m_i = \{0, 1\}$)

All of the individuals within an agent are either immature ($m_i = 0$) or mature ($m_i = 1$). At recruitment, all agents are immature. For more information see Chapter 1, page 21. Maturation is described later in this chapter in Section 4.4.3, page 112.

Location ($z_i = \{0, 1, \ldots, Z\}$)

A multinomial distribution is used to determine the recruitment location z using a $z \times 1$ stock definition vector $\Im$ and a $(J + 1) \times (Z + 1)$ recruitment matrix $\boldsymbol{\Xi}$ that defines the proportion of stock j recruiting to area z.

The stock definition vector simply specifies which areas belong to which stocks. The number of stocks cannot exceed the number of areas. For example, a three stock, three area model would be specified as $\Im = \{0, 1, 2\}$. A three area, two stock model in which the first two areas make up the first stock would be $\Im = \{0, 0, 1\}$. The recruitment matrix defines the proportion of each stock's recruits that recruit to each area in the model. For example, a three stock, three area model where all recruitment from each stock j is to each area z would be defined using

$$\mathbf{\Xi} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

where all rows sum to one. A three area, two stock model where recruitment from the first stock is into the first area and recruitment from the second stock is into the remaining two areas would be defined using

$$\mathbf{\Xi} = \begin{pmatrix} 1 & 0 \\ 0 & 0.5 \\ 0 & 0.5 \end{pmatrix}.$$

A three area, one stock model is defined as

$$\mathbf{\Xi} = \begin{pmatrix} \frac{1}{3} \\ \frac{1}{3} \\ \frac{1}{3} \end{pmatrix}.$$

Alternatively, one could specify a three area, three stock model where recruits from different stocks "leak" into different areas

$$\mathbf{\Xi} = \begin{pmatrix} 0.7 & 0.2 & 0.1 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.2 & 0.7 \end{pmatrix}$$

where $\sum_z \mathbf{\Xi}_{z,j} = 1 \quad \forall$ stocks j. Migration is described later in this chapter in Section 4.4.7, page 117. For more information on stocks and areas see Chapter 1, page 31.

Home ($h_i = \{0, 1, \ldots, Z\}$)

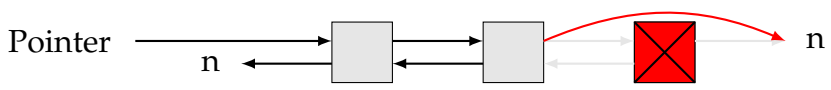
The home site is defined as the area or cell that the agent recruited to. The parameter ψ defines the probability of an agent switching its home site to its current location at a particular time step. If the probability of switching is low, then agents will rarely switch their home site, if the probability is high, then agents will switch their home site often. If $\psi = 0$ then agents will never switch their home site, if $\psi = 1$ then agents will essentially follow a random walk. The only way for a fish to pick a home site is if they have visited that cell. This structure implements the behaviours of a random walk versus attraction to an area.

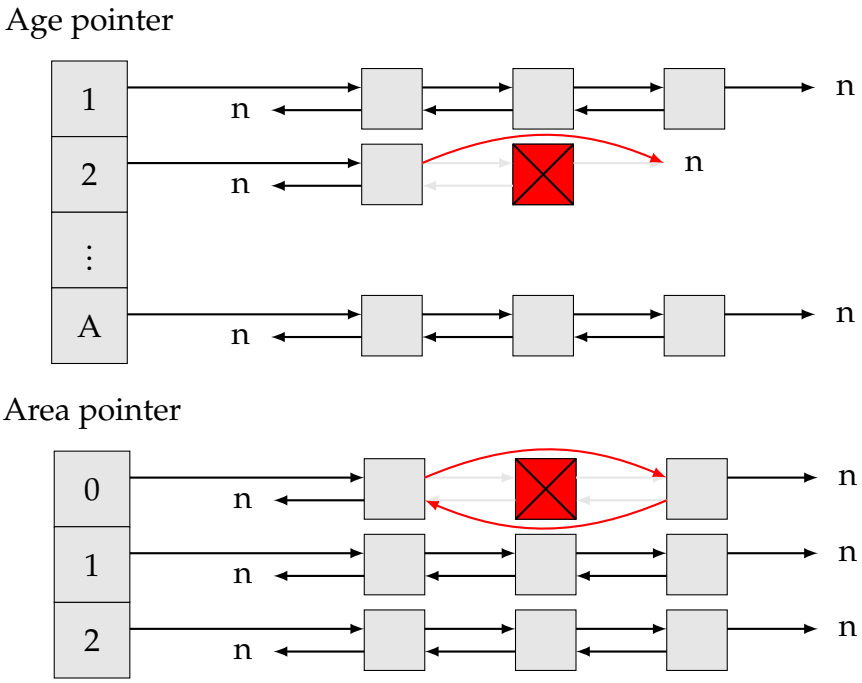
Tagged and tag year ($t_i = \{-1, 0, \ldots, Z\}$ and $y_i = \{-1, 0, \ldots, Y\}$)

All of the individuals within an agent are either untagged ($t_i = -1$) or tagged ($t_i = z$) where z is the cell that the fish was tagged in. At recruitment no fish are tagged. Similarly, no fish have a tag year at recruitment ($y_i = -1$). Tagging is described later in this chapter in Section 4.4.9, page 122.

4.2.3 Deleting agents

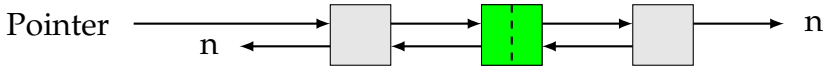
If an agent becomes depleted (by natural or fishing mortality) so that no more individuals are contained in that agent ($f_i = 0$) then the agent is removed from the population, the memory for that agent is freed, and all pointers pointing to that agent are appropriately redirected. We provide some examples below of an agent being deleted from the end of a sequence, from the end of an age sequence, and from the middle of an area sequence. In these examples, the deleted agent is shown in red with a cross and the redirected pointers in red. The old pointers are shown in grey.



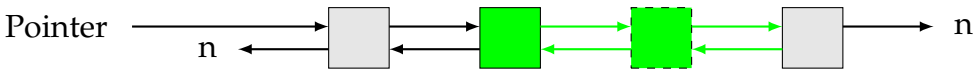


4.2.4 Splitting agents

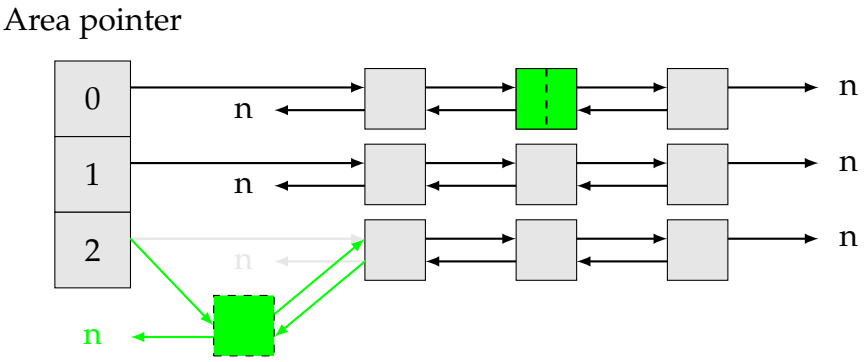
Some processes require agents to split into two or more different agents. These process include maturation, migration, and tag-release events. For example, if an individual within an agent is tagged and released, this individual splits from its parent agent into its own agent and all the attributes of the parent agent are copied into the new agent except for the tag status. Or, if a portion of the individuals within an agent migrate to another location, those individuals split from the parent agent to form their own agent in the new location. For example, if the agent shown in green below was to split



the agent pointer sequence would become

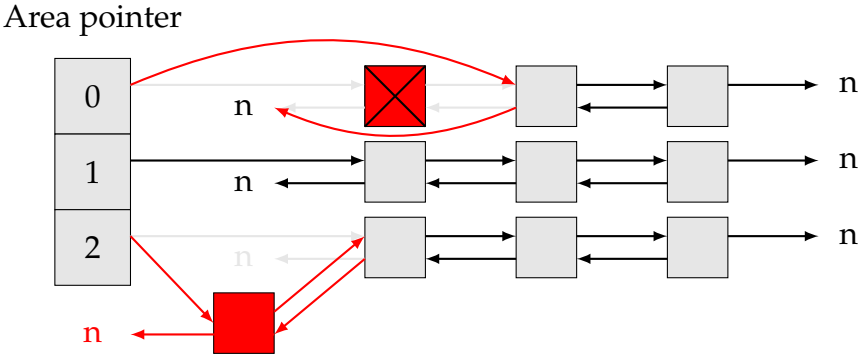


If the agent being split is moved to a different area the area sequence would become

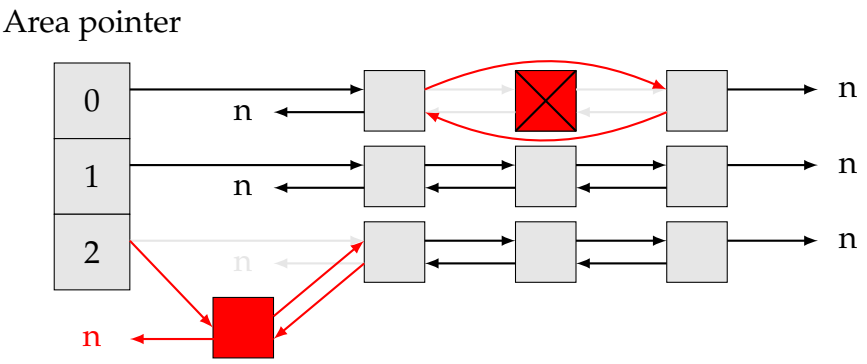


4.2.5 Moving agents

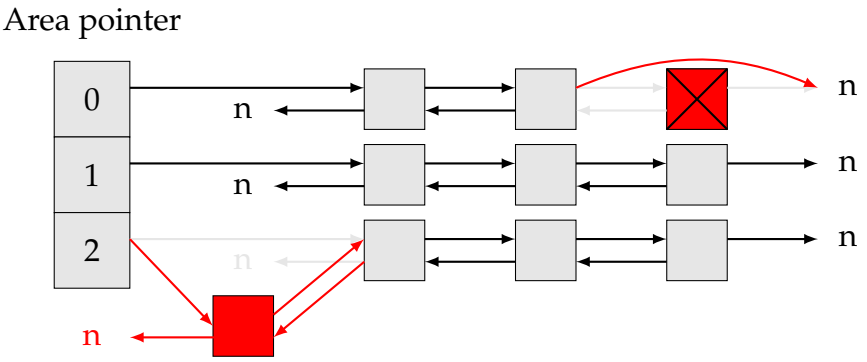
If an agent is moved from the start of a sequence then the array of pointers by area must be redirected to point to the next agent, and the next agents backwards pointer must be set to null. The agent being moved is then set as the first agent in the new area sequence, and all pointers must be redirected as appropriate. In the diagram below, all pointers being redirected are shown in red, the old pointers are grey, and the agent being moved is red.



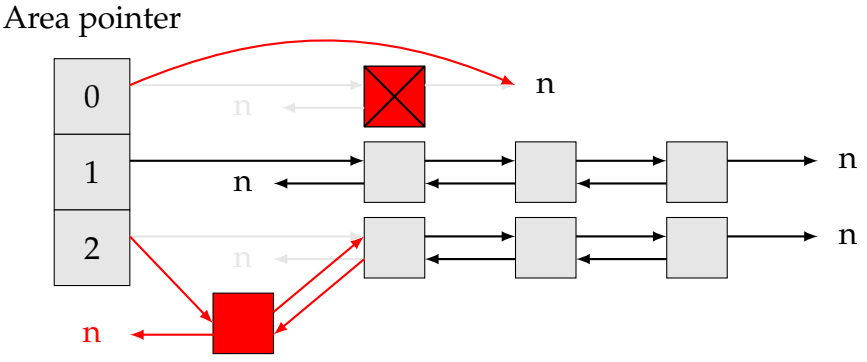
If an agent is being moved from the middle of a sequence



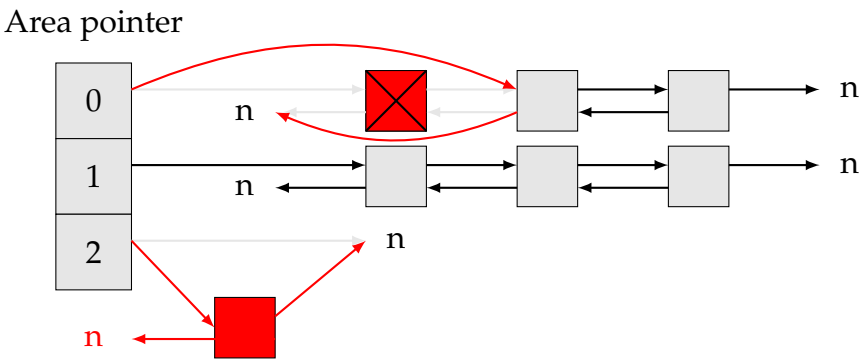
If an agent is being moved from the end of a sequence



If an agent is being moved from the start and end of a sequence



If an agent moves to the start and end of a sequence



4.2.6 Merging agents

When an agent represents a school or group rather than an individual, for computational efficiency (and in some cases for biological plausibility) it is necessary to merge significantly depleted agents (this would be analogous to schools merging). If the number of individuals within an agent (f_i) is equal to or less than the user specified merge threshold (F_{min}), then that agent (i_1) is selected for merging. The program then searches for another agent in the population that is suitable for merging with the selected agent. If another agent (i_2) is found with sufficiently depleted individuals, then the two agents must meet a set of criteria for merging to occur. Agents will only merge if they are the same age ($a_{i_1} = a_{i_2}$), same sex ($s_{i_1} = s_{i_2}$), in the same location ($z_{i_1} = z_{i_2}$), from the same home site ($h_{i_1} = h_{i_2}$), are the same maturity ($m_{i_1} = m_{i_2}$), have the same tag status ($t_{i_1} = t_{i_2}$) and were tagged in the same year ($y_{i_1} = y_{i_2}$), and the sum of the fish frequencies merging agents will not be greater than the maximum agent size ($f'_i \leq F_{\text{max}}$). The merging of the two agents then involves combining the individuals and finding the weighted mean length and weighted mean parameter values

of the individuals in the agents.

$$f'_i = f_{i_1} + f_{i_2} \qquad \text{if} \quad \begin{cases} f_{i_1} \leq F_{\text{min}} & \\ f_{i_2} \leq F_{\text{min}} & \\ f'_i \leq F_{\text{max}} & \\ a_{i_1} = a_{i_2} & \text{age} \\ h_{i_1} = h_{i_2} & \text{home} \\ m_{i_1} = m_{i_2} & \text{maturity} \\ s_{i_1} = s_{i_2} & \text{sex} \\ t_{i_1} = t_{i_2} & \text{tag} \\ y_{i_1} = y_{i_2} & \text{tag year} \\ z_{i_1} = z_{i_2} & \text{location} \end{cases}, \tag{4.3}$$

where f'_i is the number of individual fish represented by the merged agent. The weighted mean length and weighted mean parameter values (e.g. L_∞, k, M, etc) are found using

$$\theta'_i = \frac{\sum_{i \in \{i_1, i_2\}} f_i \theta_i}{f'_i}, \tag{4.4}$$

where θ_i is the length or parameter value before merging and θ'_i is the weighted mean length or parameter value. This is a computationally expensive exercise so has been developed to work on multi-core computers (multi-threading) by spawning a thread for each age group or cell and running the algorithm on each simultaneously.

Care must be taken if allowing the merging of agents. If merging is allowed in a model run then the merging process (specifically taking the weighted mean of the length or parameter values) can dilute the individual nature of the model such that the model will begin to behave more like an age-structured model. For example, the length of an agent will approach the mean length within the population (for a given cohort) as more and more agents are merged. The parameter F_{min} can be used to reduce the merging issue (i.e. setting a lower F_{min} will prevent too much merging at a computational cost).

4.3 Model structure

The main body of the model is structured in two parts: initialisation and application of the fishery.

4.3.1 Initialisation

Initialisation is done in two phases. The first phase recruits $R_{0,j}$ individuals into each stock (j) in the population each year until an equilibrium age-structure is achieved. During each year the following sequence is followed:

1. Ageing
2. Growth
3. Maturation
4. Recruitment
5. Spawning
6. Natural mortality

The details of each of these processes is described in Section 4.4. The population is initialised over many years until the population reaches equilibrium age-structure and the spawning stock biomass stabilises in each stock (Figure 4.3). Phase 2 then follows wherein random variation is introduced into the recruitment and migration is allowed:

1. Ageing
2. Growth
3. Maturation
4. Migrate home (new step)
5. Recruitment
6. Spawning

7. Migration (new step)

8. Natural mortality

In our example, we have specified phase 1 and phase 2 to both be 100 years (Figure 4.3).

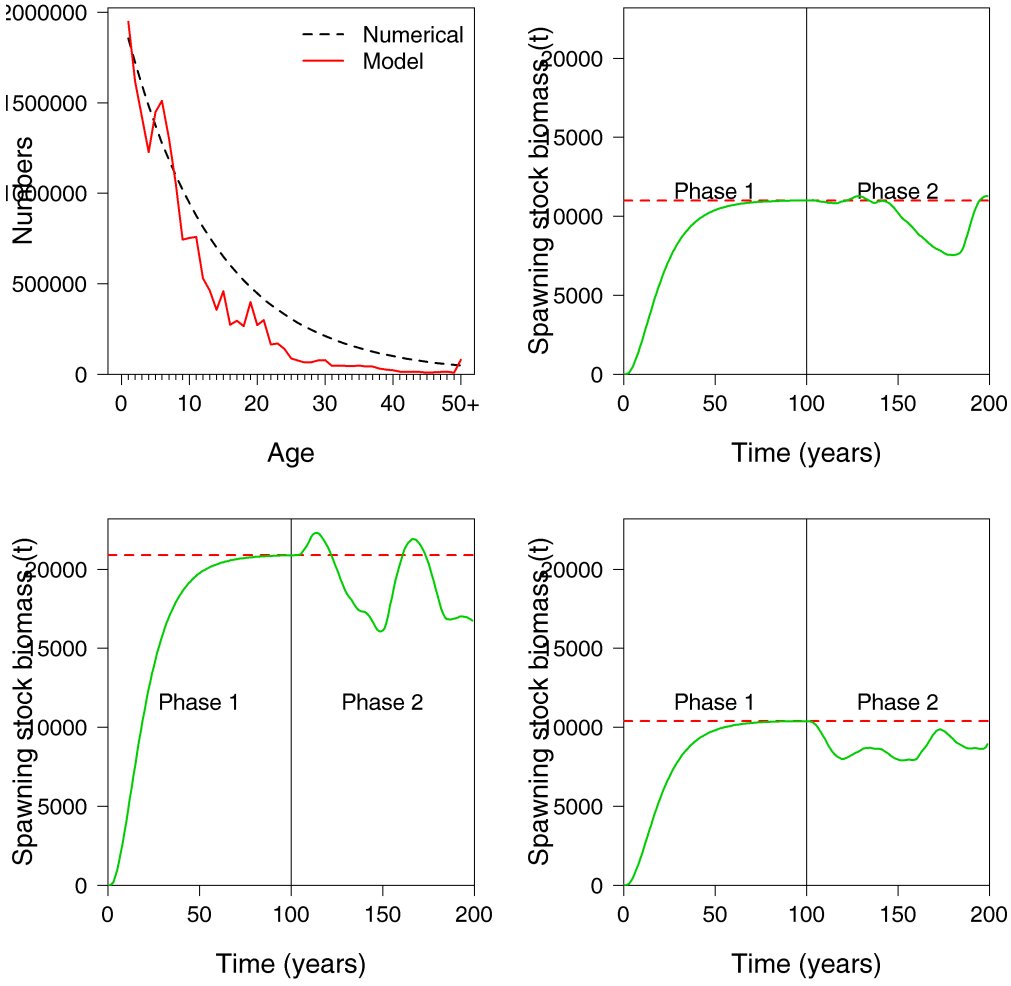


Figure 4.3: Numbers at age in the population at the end of phase 2 compared to the non-stochastic equilibrium numbers at age [top left] and the spawning stock biomass (tonnes) of each of three stocks during phase 1 and phase 2.

4.3.2 Applying the fishery

The annual cycle during the years that the fishery operates includes the following processes:

1. Ageing
2. Growth
3. Maturation
4. Migrate home
5. Recruitment
6. Spawning
7. Tagging (new step)
8. Migration
9. Natural mortality and fishing mortality (new step)

There are switches between deterministic and stochastic settings for some of these processes. These may be used to allow a user to explore differences between these approaches and the influence of variability on model outcomes. Each year is split up into time-steps so that some processes may be split between multiple time steps (e.g. applying half of the natural mortality before calculating the spawning stock biomass).

4.4 Processes

Processes in the model include

1. Ageing (deterministic)
2. Growth (deterministic or stochastic)
3. Maturation (stochastic)
4. Spawning (deterministic)
5. Recruitment (stochastic)

6. Natural mortality (stochastic)
7. Fishing mortality (stochastic)
8. Tagging (stochastic)
9. Migration home (deterministic)
10. Migration (stochastic)

We now describe each of these processes in detail.

4.4.1 Ageing

During ageing, all agents are aged by one year

$$a_i' = a_i + 1, \tag{4.5}$$

where a_i is the age of fish in agent i before ageing and a_i' is the age of fish after ageing. There is no limit on the age a fish can reach in the model. This means that the fish in an agent could reach a very old age (much older than is possible). We accept that, biologically, this is not possible, however, very few of these old agents will persist in the modelled population so we ignore this problem. When presenting these data we deal with these very old fish by simply amalgamating them into a plus group (see Chapter 1, page 17).

4.4.2 Growth

Using the von Bertalanffy growth parameters ($L_{\infty,i}$ and k_i) for agent i, the expected annual growth increment ($\Delta\bar{\ell}_i$) for an agent of length ℓ_i is

$$\Delta\bar{\ell}_i = \begin{cases} (L_{\infty,i} - \ell_i)(1 - e^{-k_i}) & \text{if } \ell_i \leq L_{\infty,i} \\ 0 & \text{if } \ell_i > L_{\infty,i} \end{cases}. \tag{4.6}$$

If growth is set to be deterministic then $\Delta\bar{\ell}_i$ is the growth increment for agent i in that year and the new length of the agent after growth ℓ_i' is

$$\ell_i' = \ell_i + \Delta\bar{\ell}_i. \tag{4.7}$$

If growth is set to be stochastic then the growth increment is simulated as a normally distributed random variable with mean $\Delta\bar{\ell}_i$ and standard deviation $\sigma_{\Delta\ell}$

$$\Delta\ell'_i \sim \mathcal{N}\left(\Delta\bar{\ell}_i, \sigma^2_{\Delta\ell}\right) \qquad \text{where} \qquad \sigma_{\Delta\ell_i} = c_\ell \Delta\bar{\ell}_i, \tag{4.8}$$

$\Delta\ell'_i$ is the simulated growth increment to be added to the length of the individuals in agent i and c_ℓ is the coefficient of variation of growth. The new length of the agent after growth ℓ'_i is

$$\ell'_i = \ell_i + \Delta\ell'_i. \tag{4.9}$$

An example is given in Figure 4.4. Growth in fisheries models is intro-

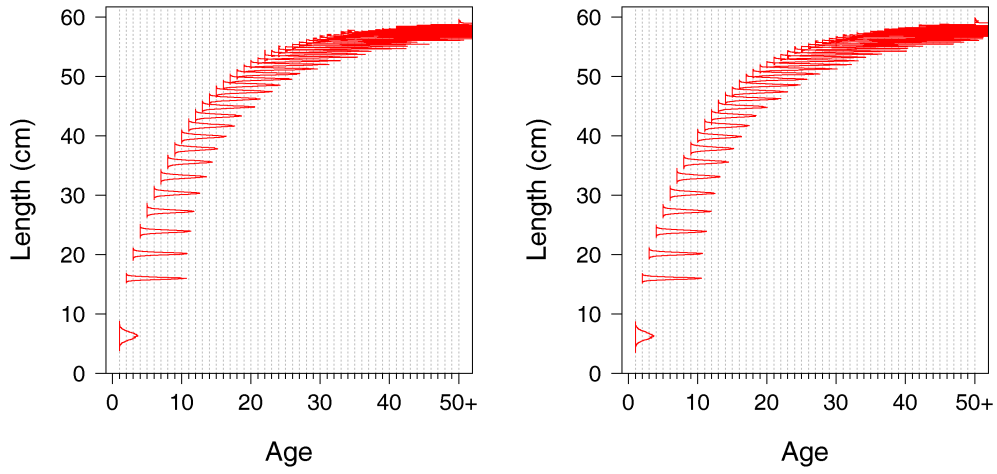


Figure 4.4: Distributions of length (cm) at age (years) of females [left] and males [right] of agents in the initialised population.

duced in Chapter 1, page 19.

4.4.3 Maturation

A logistic producing ogive is used to describe the proportion of fish in the population that are mature in any given year

$$\mu_{m,i} = \begin{cases} 0 & \text{if } x_i < L_i \\ \lambda(L_i) & \text{if } x_i = L_i \\ (\lambda(x_i) - \lambda(x_i - 1))/(1 - \lambda(x_i - 1)) & \text{if } L_i < x_i < H_i \\ 1 & \text{if } x_i \geq H_i \end{cases}, \tag{4.10}$$

where

$$\lambda(x_i) = 1 \Big/ \left(1 + 19^{(A_{50,i} - x)/A_{to95,i}}\right),$$

$\mu_{m,i}$ represents the probability a fish of size x_i (where x_i could be age a_i, length ℓ_i, or weight w_i) will become mature in agent i. The parameters L_i and H_i respectively define the x at which no fish are mature and the x_i at which all fish are mature, respectively. $A_{50,i}$ and $A_{to95,i}$ are respectively the x_i at which 50% of individuals within the population are mature and the difference in x_i at which 95% of individuals within the population are mature. The number of fish within the agent that mature is a draw from a binomial distribution

$$m'_i \sim \mathcal{B}\text{in}(f_i, \mu_{m,i}), \tag{4.11}$$

If a non zero subset of individuals within the agent reach maturity, then these individuals will split from the parent agent and form their own new agent, leaving the as yet immature individuals behind (e.g. Figure 4.5). Maturation is covered in more detail in Chapter 1, page 21.

4.4.4 Spawning

Spawning refers to the time at which the spawning stock biomass SSB_y and stock recruitment relationship $SR(SSB_y)$ are calculated. The spawning stock biomass is calculated as

$$SSB_y = \sum_{i=1}^{I_y} f_{i,y} w_{i,y} m_{i,y}, \tag{4.12}$$

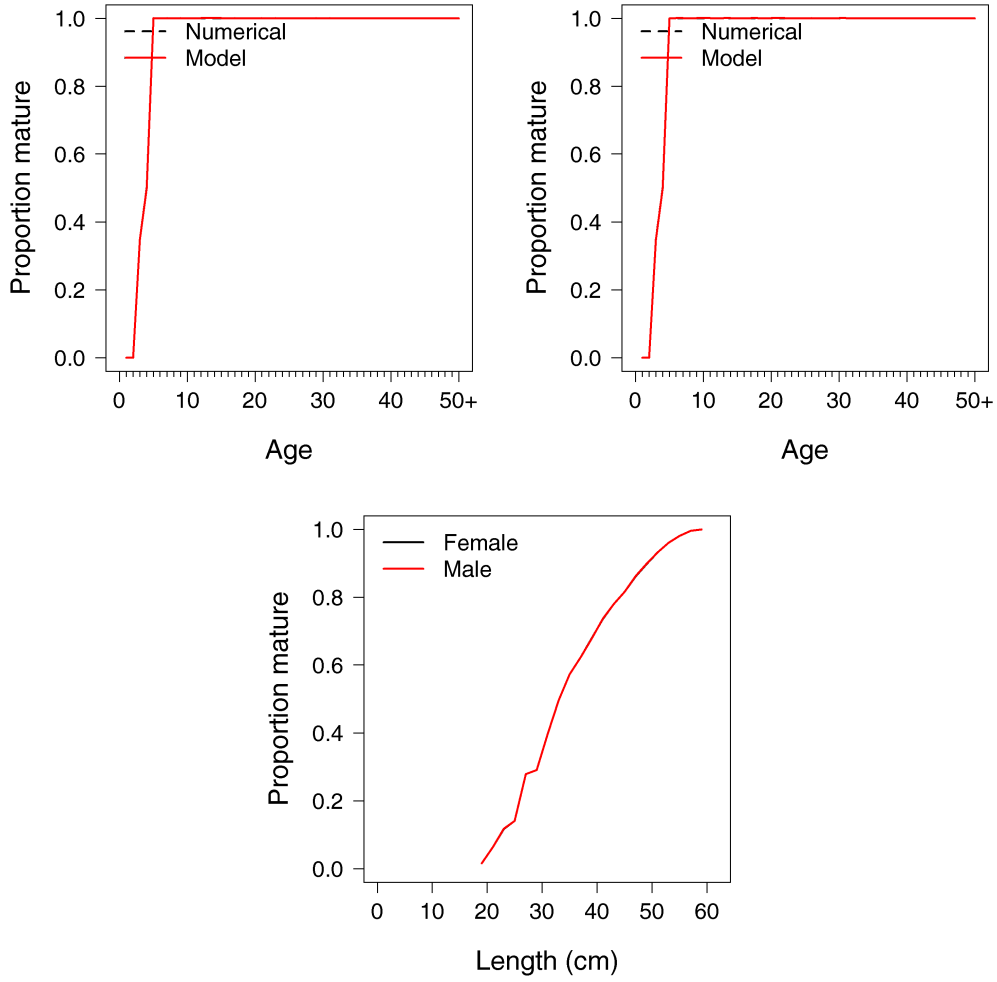


Figure 4.5: The proportion of fish mature by age (years) in the initialised population for females [top left] and males [top right] showing the deterministic maturity ogive (Numerical), and the proportion of mature female and male fish by length (cm) [bottom]. The black lines sit directly behind the red lines.

where $w_{i,y}$ is the weight (tonnes) of agent i during year y calculated as

$$w_i = \alpha_i \ell_i^{\beta_i}. \tag{4.13}$$

Calculating the total weight of agent i also allows comparisons between length and weight, and age and weight to be made (Figure 4.6).

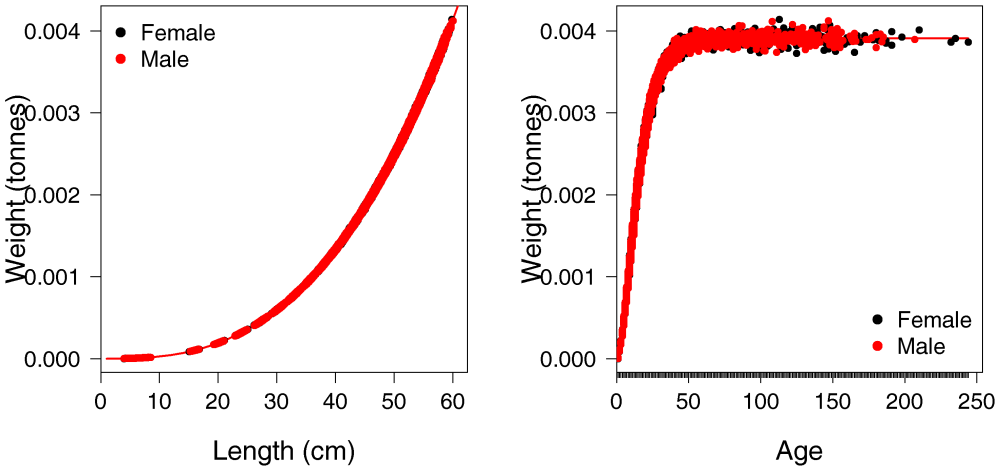


Figure 4.6: The length-weight relationship [left] and the age-weight relationship [right].

A Beverton-Holt stock recruitment relationship is used

$$SR(SSB_y) = \frac{SSB_y}{B_0} \Bigg/ \left(1 - \frac{5h-1}{4h}\left(1 - \frac{SSB_y}{B_0}\right)\right), \tag{4.14}$$

where B_0 is the deterministic virgin biomass (tonnes) and h is the steepness parameter. For more information on spawning stock biomass see Chapter 1, page 24. For more information on stock recruitment see Chapter 1, page 24.

4.4.5 Recruitment

Recruitment involves the addition of new individuals and thus new agents to the population each year. The number of individual fish that recruit in

year y is assumed to be dependent on an underlying average recruitment (R_0), the spawning stock biomass during the previous year ($y-1$), and the strength of the year class (cohort) during year y. During phase two of initialisation, the recruitment for each year is modified by log-normal annual deviations and the stock recruitment relationship

$$R_y = R_0 \times SR(SSB_{y-1}) \times e^{\varepsilon_y^R - \sigma_\varepsilon^2/2}, \tag{4.15}$$

where

$$\varepsilon_y^R = \sqrt{\rho}\varepsilon_{y-1}^R + \sqrt{1-\rho}\varepsilon_y \qquad \text{and} \qquad \varepsilon_y \sim \mathcal{N}\left(0, \sigma_\varepsilon^2\right),$$

σ_ε is the standard deviation of ε_y, and ρ determines the serial auto-correlation in recruitment between years. The total number of recruits R_y in year y is then split by location and sex using Equations 4.16 and 4.17 below

$$R_{y,z} \sim \mathcal{M}\text{ultinomial}\left(R_y, \xi_1, \ldots, \xi_Z\right), \tag{4.16}$$

where $R_{y,z}$ is the number of individuals recruiting to location z in year y and $\xi_1, \ldots, \xi_Z$ is a column of $\boldsymbol{\Xi}$ specifying the probabilities of recruiting in location z (see Section 4.2.2, page 99). $R_{y,z}$ is then passed to

$$R_{s,y,z} \sim \mathcal{B}(R_{y,z}, p_{\text{male}}), \tag{4.17}$$

where $R_{s,y,z}$ is the number of individuals recruiting as sex s, to location z, in year y, and p_{male} is the is the probability that a fish will be male (i.e. $s_i = 1$). These individual fish are then allocated into agent structures where the maximum number of individuals contained in an agent is F_{max}

$$f_{i,y} = F_{\text{max}} \qquad \text{where} \quad \sum_{i=1}^{I_{a_{\text{min}},y}} f_{i,y} = R_y \quad \forall y, \tag{4.18}$$

$I_{a_{\text{min}},y}$ is the number of agents in a cohort in year y which is determined by

$$I_{a_{\text{min}},y} = \left\lceil \frac{R_y}{F_{\text{max}}} \right\rceil. \tag{4.19}$$

If R_y/F_{max} is not a whole number in year y, then the remaining $f_{I_{a_{\text{min}},y}}$ individuals are placed in agent $I_{a_{\text{min}},y}$ so that $\sum_{i=1}^{I_{a_{\text{min}},y}} f_{i,y} = R_y$ is true for that year (i.e. we create $I_{a_{\text{min}},y} - 1$ agents of full size F_{max} and one of size $R_y - (I_{a_{\text{min}},y} - 1)F_{\text{max}}$). The following values of F_{max} can be supplied to structure the model as an individual-based model, agent-based model, or age-structured model:

- Individual-based ($F_{\max} = 1$, each agent is a single fish)
- Agent-based ($1 < F_{\max} < R_y$, intermediate case)
- Age-structured ($F_{\max} \geq R_y \quad \forall y$, each agent is a complete age cohort)

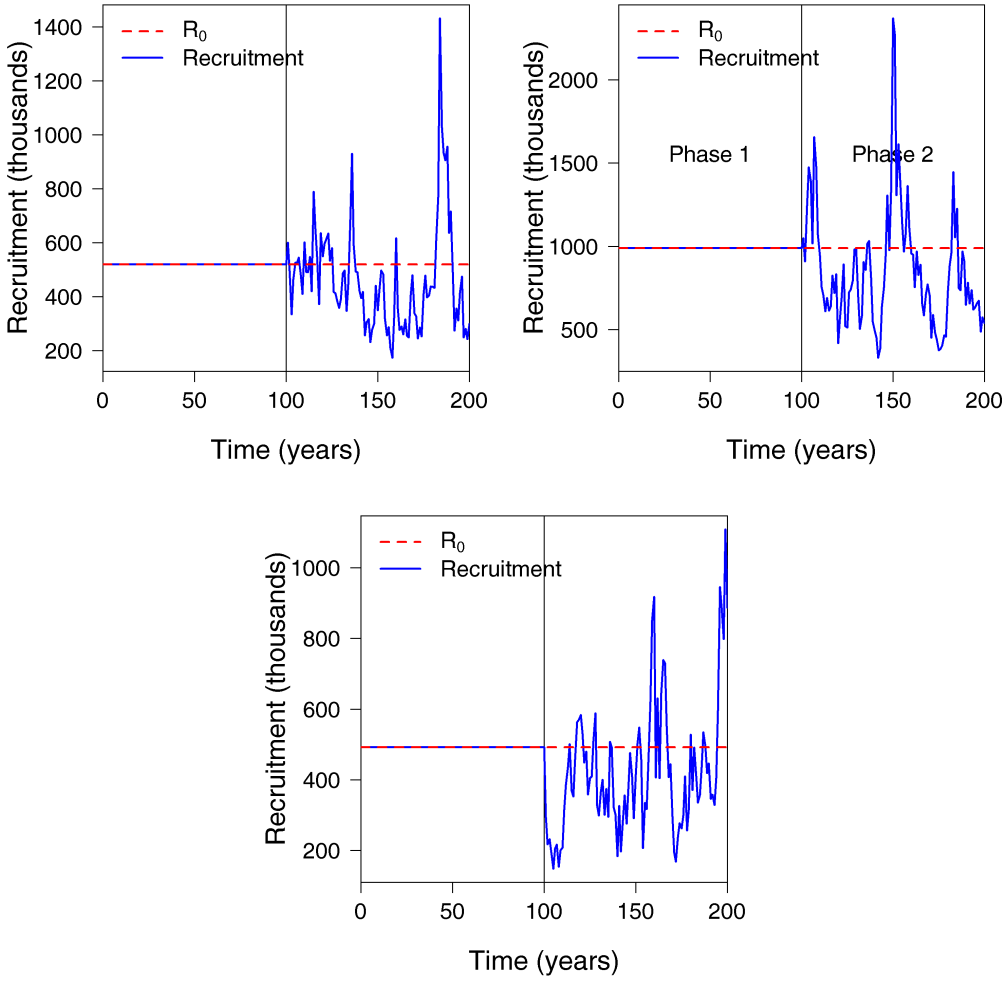


Figure 4.7: Recruitment (thousands of individuals) during phase 1 and phase 2 of initialisation in each of three stocks. Each phase was 100 years.

An introduction to recruitment is provided in Chapter 1, page 24.

4.4.6 Natural mortality

Natural mortality is the death of fish due to causes not associated with fishing (e.g. cannibalism, competition, disease, old age, predation). Natural mortality is applied to the frequency of individuals within agents f_i using a binomial distribution

$$f'_i \sim \mathcal{B}\text{in}\left(f_i, e^{-\tau M_i}\right), \tag{4.20}$$

where f'_i is the number of individuals represented by agent i after natural mortality has been applied, τ is the portion of the natural mortality to be applied in the current time step, and M_i is the natural mortality rate for agent i. If $f'_i = 0$ then the agent will be deleted as described in Section 4.2.3, page 101, or if f'_i is too small it may be merged with another agent as described in Section 4.2.6, page 105.

4.4.7 Migration

There are two types of migration processes in this model: migrations home and random migrations. Home migration is when mature agents return to their home (i.e. spawning) area. Random migrations use a specified $z \times z$ migration matrix ($\mathbf{\Omega}$). For example

$$\mathbf{\Omega} = \begin{pmatrix} 0.770 & 0.054 & 0.177 \\ 0.087 & 0.514 & 0.399 \\ 0.241 & 0.276 & 0.483 \end{pmatrix} = (\omega_{z,z'}) = \text{Prob}(z \to z')$$

where $\sum_{z'} \omega_{z,z'} = 1 \quad \forall\, z$, specifies migration between three areas. A multinomial distribution is then used to move fish at any given point in time in the model.

$$f'_{i_1,z_1}, \ldots, f'_{i_Z,z_Z} \sim \mathcal{M}\text{ultinomial}\left(f_i, \omega_1, \ldots, \omega_Z\right) \tag{4.21}$$

where $\omega_1, \ldots, \omega_Z$ is a row of $\mathbf{\Omega}$, and $f'_{i_1,z_1}, f'_{\ldots,\ldots}, f'_{i_Z,z_Z}$ are the numbers of individuals in each agent (the agent splits upon migration of some individuals within that agent).

4.4.8 Fishing mortality

During fishing events fish are removed from the population and the catch in numbers and weight is recorded (Figures 4.8 and 4.9). If a caught fish has a tag then this is also recorded (along with the cell that the fish was originally tagged in and when the fish was first tagged). First we determine the vulnerable biomass V_y during year y as

$$V_y = \sum_{i=1}^{I_y} S_i f_{i,y} w_{i,y} \tag{4.22}$$

where I_y is the number of agents in the population, S_i is the selectivity of the fishery on agent i, $f_{i,y}$ is the number of individuals contained in agent i in year y, and $w_{i,y}$ is the weight of each individual in agent i in year y. The selectivity of an agent is determined using

$$S_i = f(\boldsymbol{\theta}, x_i), \tag{4.23}$$

where $\boldsymbol{\theta}$ are the parameters of the selectivity ogive used, and x_i is the size of agent i (can be the age a_i, length ℓ_i or weight w_i in agent i). Next we specify a set of catches ($C_{y,z}$) for each year y in each area z. These may be real values from an observed catch history, or simulated. The exploitation rate is then calculated using

$$U_{y,z} = \frac{C_{y,z}}{V_{y,z}}, \tag{4.24}$$

where $U_{y,z}$ is the exploitation rate to be applied during year y in cell z, $C_{y,z}$ is the total catch (tonnes) taken from each cell during each year, and $V_{y,z}$ is the biomass (tonnes) that is vulnerable to fishing in each cell each year. The fishery is applied to each agent using a binomial distribution

$$C_{i,y,z}^{\text{numbers}} \sim \mathcal{B}(f_{i,y}, U_{y,z} S_i), \tag{4.25}$$

where $C_{i,y,z}^{\text{numbers}}$ is the number of fish removed from agent i in year y in cell z. The new number of individuals represented by agent i can be written

$$f_i' = f_i - C_{i,y,z}^{\text{numbers}}. \tag{4.26}$$

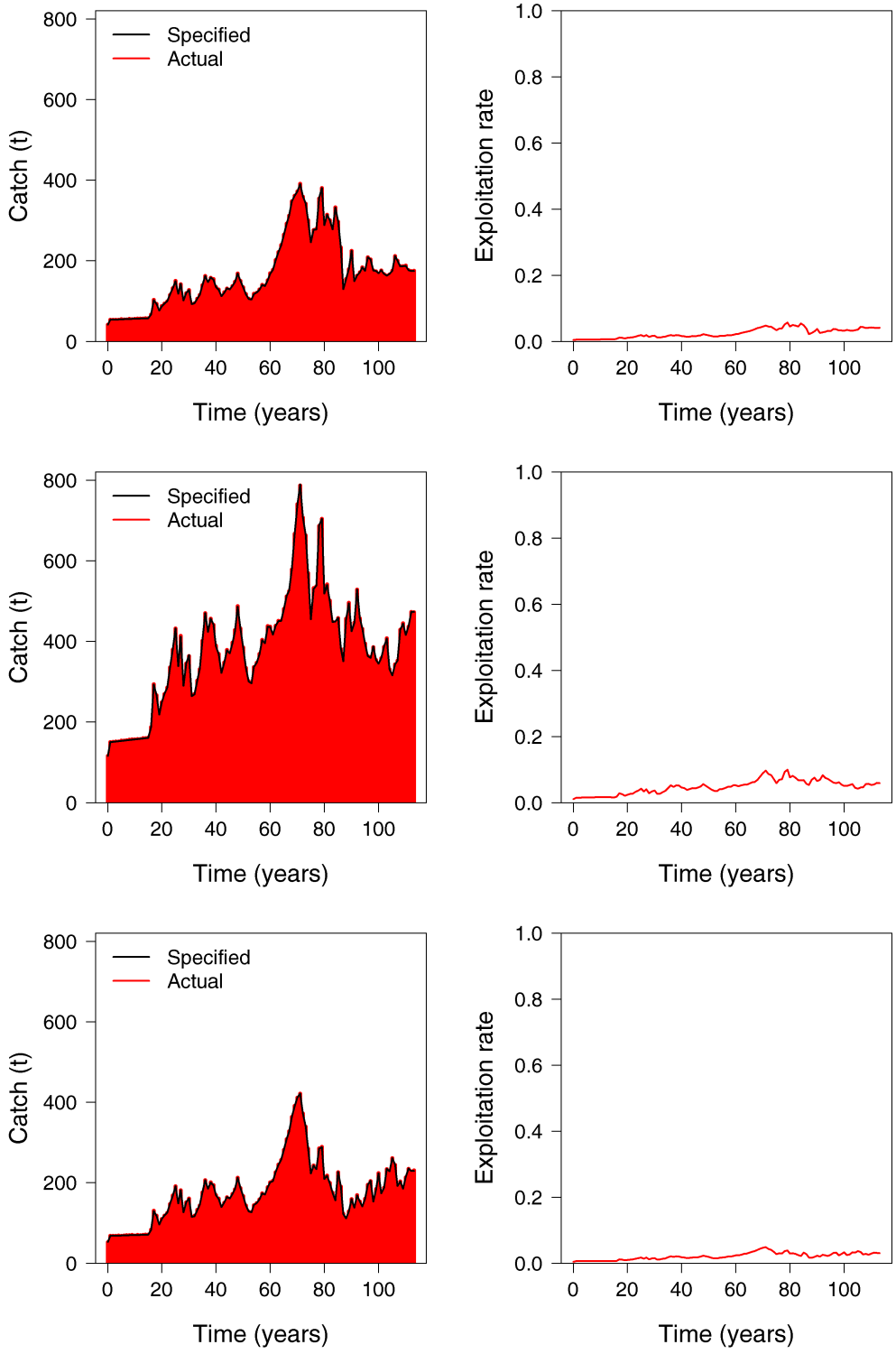


Figure 4.8: The catch ($C_{y,z}$, tonnes) by year (y) taken from each of the three areas (z) in the model [left column] and the exploitation rate ($U_{y,z}$) by year and area of each of these catch histories [right column].

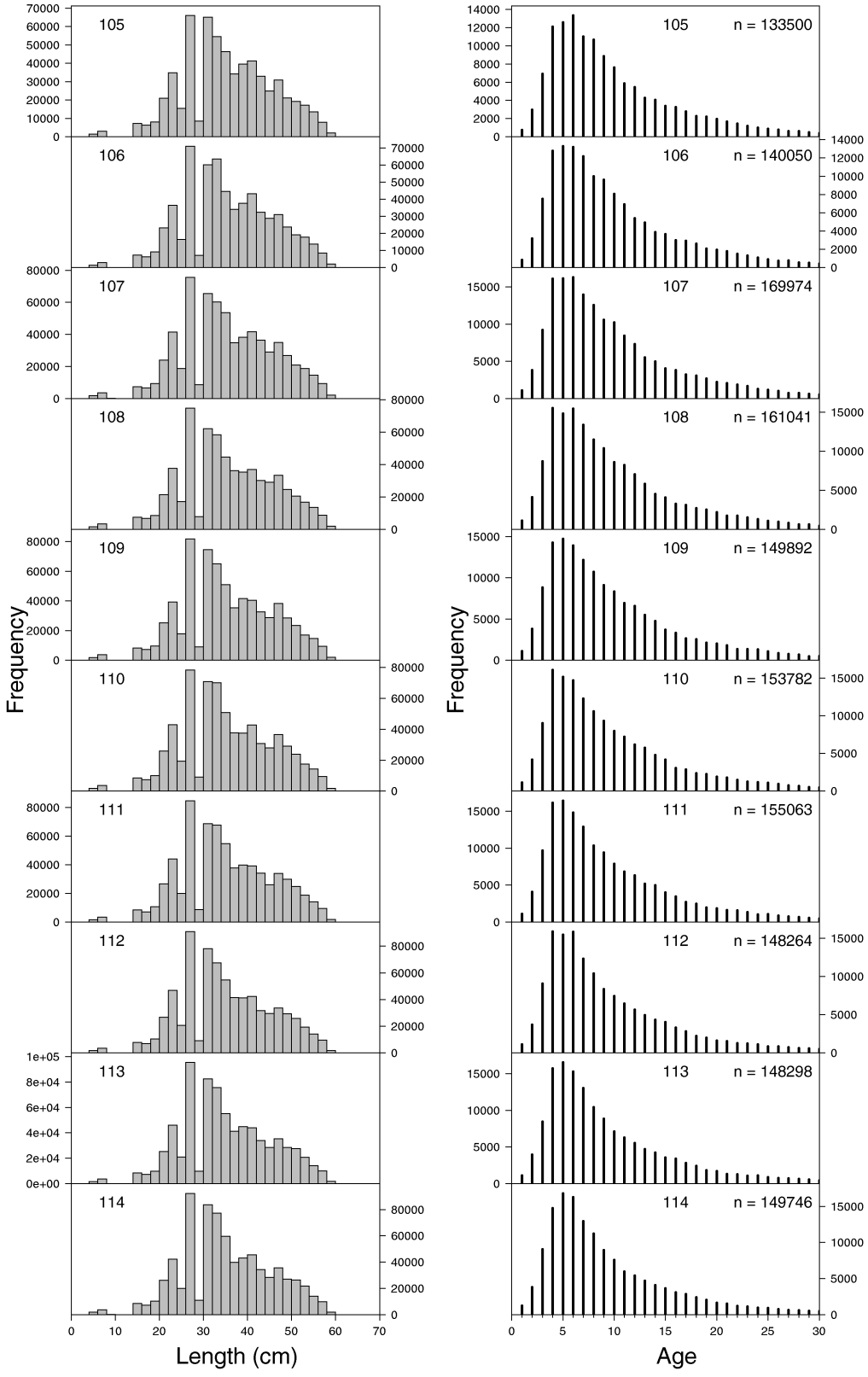


Figure 4.9: Length-frequency [left] and age-frequency histograms for the final 10 years of a single run of the model.

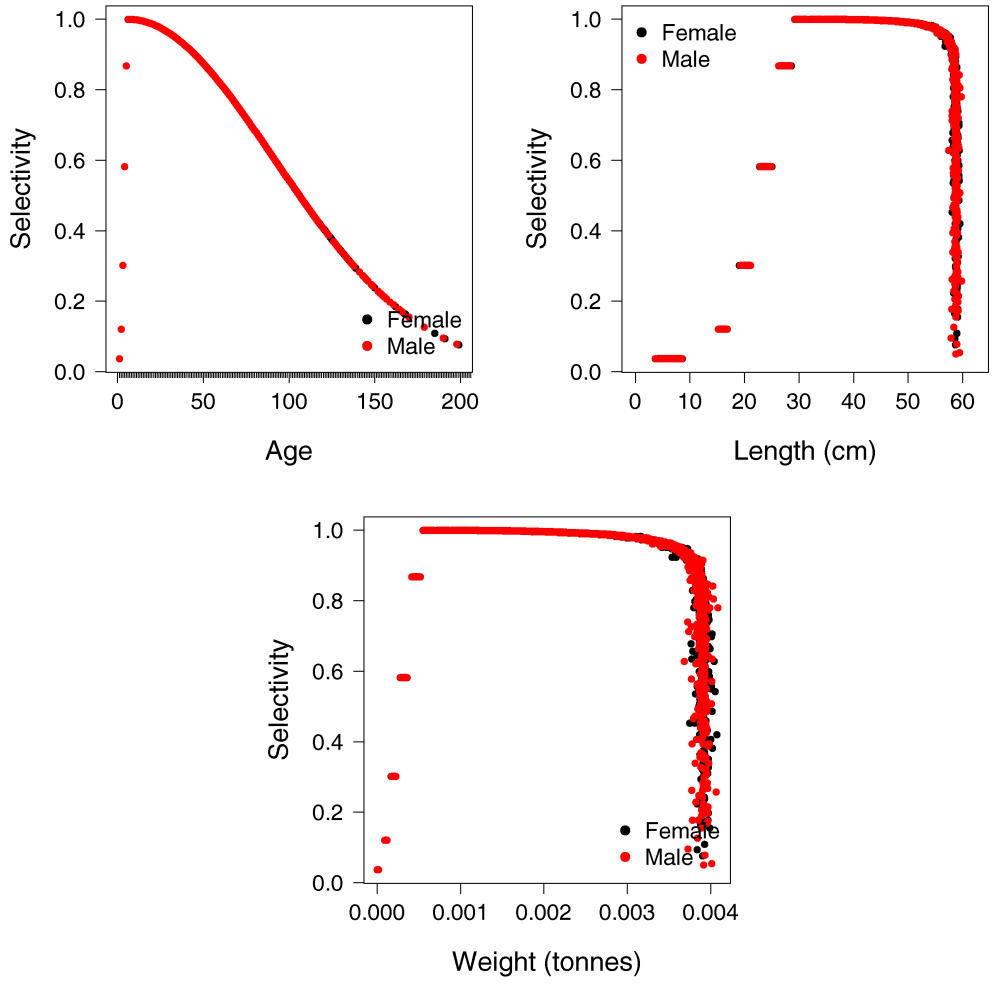


Figure 4.10: Selectivity by age [top left], by length (cm) [top right] and by weight (tonnes) [bottom] for males and females in the population.

The number of fish caught from each agent are converted to a weight using

$$C_{y,z}^{\text{weight}} = \sum_k C_{i,y,z}^{\text{numbers}} \times w_{i,y}, \tag{4.27}$$

where $C_{y,z}^{\text{weight}}$ is the total weight of fish caught in year y in cell z. This means that the catch removed from each year-cell combination will not be exactly the same as the specified catch in each area (i.e. $C_{y,z} \simeq C_{y,z}^{\text{weight}}$). The effort ($E_{y,z}$) during year y in area z required to obtain the catch is calculated using

$$E_{y,z} = \frac{C_{y,z}^{\text{weight}}}{q_{y,z} V_{y,z}}, \tag{4.28}$$

where $q_{y,z}$ is the catchability coefficient during year y in area z calculated as

$$q_{y,z} \sim \mathcal{N}(\mu_q, \sigma_q^2). \tag{4.29}$$

4.4.9 Tagging

When applying tagging in the model, we first calculate the vulnerable component of the population $V_{y,z}^{\text{tag}}$ (numbers) during year y in area z using

$$V_{y,z}^{\text{tag}} = \sum_i S_i^{\text{tag}} f_{i,y}. \tag{4.30}$$

where S_i^{tag} is the tagging selectivity ogive used. Note that this differs from the familiar vulnerable biomass equation because the weight component has been dropped. The tagging selectivity is

$$S_i^{\text{tag}} = f_{\text{tag}}\left(\boldsymbol{\theta}, x_i\right), \tag{4.31}$$

where $\boldsymbol{\theta}$ are the parameters of the selectivity ogive used, and x_i is the size of agent i (can be the age a_i, length ℓ_i or weight w_i of agent i).

The rate of tagging ($U_{y,z}^{\text{tag}}$) is then calculated as

$$U_{y,z}^{\text{tag}} = \frac{T_{y,z}}{V_{y,z}^{\text{tag}}}, \tag{4.32}$$

where $T_{y,z}$ is the number of fish in each cell in each year that we want to tag. This differs from the exploitation rate, which is a proportion. Tags are then applied to individuals within agent using a binomial distribution

$$t'_i \sim \mathcal{B}\text{in}(f_i, U^{\text{tag}}_{y,z} S^{\text{tag}}_i). \tag{4.33}$$

If some individuals within an agent do receive a tag, then those individuals split to form their own agent and the year that the individuals were tagged is also recorded within the agent (i.e. $y'_i = y$).

4.5 Population calculations

Finally we can use the properties of the agents to compute aggregate properties of the whole population. To determine the number of individuals in any given year we use

$$N_y = \sum_{i=1}^{I_y} f_{iy}, \tag{4.34}$$

where f_{iy} is the frequency of individuals in agent i during year y in the population and is found by traversing the sequence of agents and summing as we go. Similarly, the biomass during year y can be calculated as

$$B_y = \sum_{i=1}^{I_y} f_{i,y} w_{i,y}, \tag{4.35}$$

where $w_{i,y}$ is described in Equation 4.13. Calculating the virgin biomass and spawning stock biomass were covered earlier (Equations 4.22 and Equation 4.12). Figure 4.11 gives the SNA 1 example showing the total, vulnerable and spawning stock biomass of the population during each year of the fishery.

4.6 Technical details

The model is written in C++ (C++11) and compiled in Linux using g++ (GCC) version 4.9.2. The software is designed to be compiled each time a

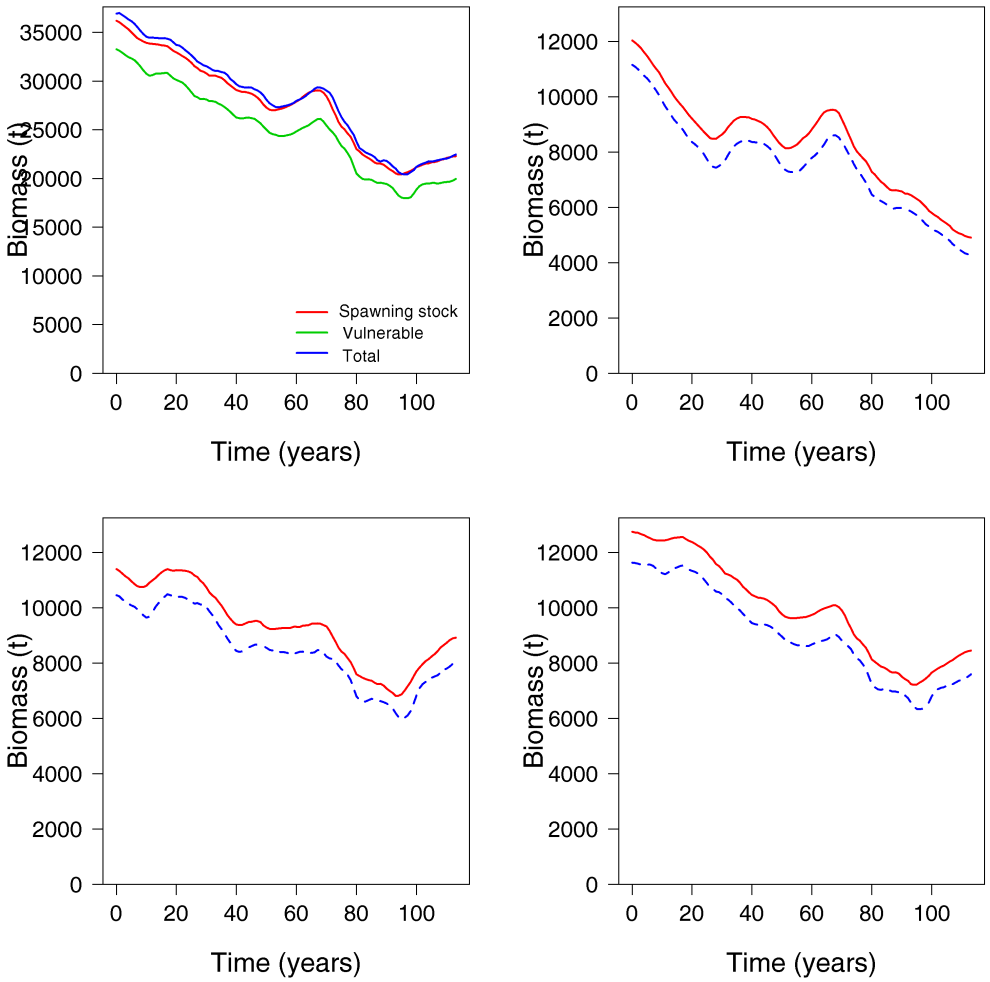


Figure 4.11: Total, vulnerable and spawning stock biomasses (tonnes) during each year of the fishery [top left] and the total and vulnerable biomass for each of the three stocks during the fishery [top right and bottom].

model run is done. Preprocessor directive flags within the C++ code indicate how the model is to be structured at compile time including switches to turn on/off processes such as stochasticity and multithreading. Multithreading is done using pthreads. There is some overhead involved when splitting, passing and collecting multiple threads. A balance must be struck between the number of threads passed and actual speed gains.

The primary memory (i.e. RAM) required to run the model is a function of the population size being simulated and the complexity or resolution desired of the model. Memory problems will arise if trying to simulate populations with an R_0 parameter in the millions of fish with few fish in each agent (i.e. many agents will be created each year potentially resulting in billions of agents needing to be stored in the primary memory after initialisation of the model). The solution is to run the model on a computer with more RAM, or scale down the size and complexity of the population.

4.7 Discussion

This chapter describes a spatially explicit multi-generational agent-structured fish simulation model that allows flexibility in specifying population and spatial dynamics. The model has the potential to consider individual variability, individual movement, and spatial heterogeneity in the environment. The aim was to construct a model that is sufficiently rich that it can be used to simulate more complete, realistic fish populations. The simulated data can be used to test stock assessment methodologies - which are usually based on samples from the population, and incomplete data.

In this model, the population is made up of a collection of agents. Agents are collections of individual fish. The advantage of an agent based model is that all fish in an agent have identical properties, and can therefore be treated simultaneously, much like in age-structured models (or the super-individual concept, Scheffer et al. 1995). However, unlike age-structured models, an agent can contain a single fish (individual-based), any number of fish, or an entire cohort of fish (all fish of the same age in the same area

in the population). The agents that make up a population can be accessed and modified to apply standard process including ageing, growth, maturation, recruitment, spawning, natural mortality, fishing mortality and movement. Special types of migration are also easily implemented, such as migrations back to the area that the fish recruited to (i.e. home or site fidelity).

The additional complexity of this model comes at the cost of computational time and memory. The model can take many hours to do a single run, and the larger the population that we want to model, the more primary computer memory (RAM) the model will use. Setting R_0 at over 10 million, even with relatively few agents (by specifying that more individuals may be contained within each agent), results in models that will use more than 8 GB of RAM. These issues can be solved in one of three ways: (1) by brute force (using a computer with more RAM and running for longer); (2) parts of the model could be written the the computers secondary memory (the hard drive) and the model could be worked on one part at a time; or (3) by reducing the population size (i.e. using a smaller R_0) or complexity of the model (i.e. fewer areas, allowing agents to merge).

While the structure of this model may be novel in fisheries research, it may be overly complex as it stretches the limits of even the most modern high performance computers. As an alternative, one might revert back to the old matrix-style stock assessment structure while maintaining some of the additional complexity through binning those agent attributes that are measured on a continuous scale and important to the model (e.g. age-structured models that are also binned by a number of length-classes or some other variable like home site). For example, persistent individual variation in growth has been approximated by tracking individual platoons of fishes having the same age but different average growth rates (Taylor & Methot 2013). Tracking abundance by platoon then allows stock assessment models to account for the impact of size-selective fishing on average growth rates.

In describing this model we developed a spatially explicit agent-based snapper model (for more detail on snapper see Chapter 3, page 77). This

snapper model is used later to develop a Bayesian emulator of the agent-based model in Chapter 7 (page 298). While this model has the capacity to test many ideas or theories in fished or unfished populations, this was not the focus of this thesis. However, we discuss the future research potential for this model below and recommend Thorson et al. (2012) as an example that does use an ABM to test fisheries theory.

Each agent can store additional information such as agent-specific parameters relating to size, maturity and natural mortality. Therefore, size-specific fishing pressure theory may be tested. Many recent studies of captive or wild populations have demonstrated persistent differences in behavioural or phenotypic traits among individuals (Shelton et al. 2013, Webber & Thorson 2015). Persistent differences in activity level or tolerance of predation risk (i.e., a tendency to forage in high vs. low-quality habitat) will likely lead to persistent differences in growth rates among individuals. Subsequently, persistent differences in growth rate, combined with size-selective harvest targeting larger individuals, can result in older individuals being composed primarily of slow-growing individuals (termed "Rosa Lees Phenomenon"), and has been demonstrated to occur in small-lake mesocosm experiments (Biro & Post 2008). In this way, failure to account for persistent differences in growth rate can lead to biased estimation of average growth rates in wild populations; population dynamics models are increasingly being developed to account for these effects (Taylor & Methot 2013). Failure to account for such persistent differences can therefore lead to biases in stock assessment. This model provides a framework with which to test ideas like these by introducing different growth models and experimenting with different selectivity patterns.

Agents also have the ability to store information about the agents past, in essence giving the agent a "memory". In a spatially-explicit model, this could allow us to test site fidelity theory and the impacts of heterogeneous mixing on estimates of population size. It is well known that homogeneous mixing is a key assumption of tag-recapture analyses that aim to estimate population size. Therefore, this model could serve as a useful platform for testing tag-recapture theory in fish populations and

integrated stock assessments.

Chapter 5

State-space models

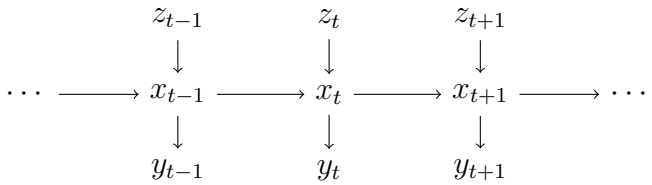
In this chapter we develop biomass dynamics and age-structured state-space models. We use examples to illustrate the construction and inference of these models. The biomass dynamics model example uses the packhorse rock lobster (*Sagmariasus verreauxi*, PHC) fishery in northern New Zealand (see Chapter 3, page 80). The age-structured model example uses the snapper (*Pagurus auratus*, SNA) fishery in northern New Zealand (SNA 1, see Chapter 3, page 77).

5.1 Introduction

Commercial and ecological management of fisheries requires good estimates of stock sizes. These estimates are obtained by modelling fish population dynamics. The models are fitted to commercial fisheries data, and/or information collected independently of the commercial fishery. Generally these data are patchy in space and time and not collected randomly, breaking almost all of the assumptions of classical statistical models and tests. These models should provide estimates with as little bias as possible and the uncertainty in the data should be properly reflected in the estimates produced by the models. Therefore, stock assessment models should have the ability to deal with uncertainty and adequately convey the uncertainty in the estimates they provide. Current stock as-

sessment models tend to underestimate the true uncertainty (Magnusson et al. 2013). This can produce biased results (Mormede, Dunn & Hanchet 2013) which can lead to incorrect inferences (Hoshino et al. 2014).

State-space models incorporate both observation and process uncertainty. State-space models are statistical Markov models, also known as hidden Markov models (HMM) or latent state models, in which the system is assumed to be a Markov process with unobserved (hidden) states. Specifically, state-space models relate observations y_t (e.g. CPUE) at a time t to unobserved states x_t (e.g. biomass) through stochastic observation equations for y_t. Stochastic transition equations define how the hidden states (x_t) are assumed to evolve in time. An observed or specified control variable or covariate (z_t) may also be included in a state-space system (e.g. catch). Thus, the process model could take the form $x_t = f(x_{t-1}, z_t, \boldsymbol{\theta}) + \varepsilon_t^p$ and the observation model $y_t = g(x_t, z_t, \boldsymbol{\theta}) + \varepsilon_t^o$ where $\boldsymbol{\theta}$ are model parameters, ε_t^o are observation errors, and ε_t^p are process errors. These equations can be represented graphically as:



By incorporating both observation and process error, state-space models can help us better quantify the uncertainty of parameters of interest (Harwood & Stokes 2003). State-space models can incorporate variability in key population parameters by allowing these parameters to follow a first order autoregressive process over time. Also, state-space features in models can help reduce the number of model parameters (Nielsen & Berg 2014).

For example, state-space methods have been used to some extent in age-structured models to estimate time-varying selectivity and random walk fishing mortalities (Butterworth et al. 2003, Nielsen & Berg 2014). Although state-space models have had some limited use in fisheries modelling they are the exception rather than the norm, particularly in age-structured models. However, Meyer & Millar (1999) have developed basic

state-space biomass dynamics models.

Fully state-space age-structured models that estimate the numbers of fish at age each year ($N_{a,t}$) as a latent state are almost absent from the literature. However, Millar & Meyer (2000) develop an age-structured model that treats total mortality ($M_{a,t}$ being made up of natural mortality and unreported catch) as being state-space. We describe this model briefly below, but for more detail see their original paper. Their process model is

$$N_{a,t} = N_{a-1,t-1}e^{-M_{a,t}} - C_{a,t}e^{-0.5M_{a,t}},$$

where $N_{a,t}$ is the mid-year numbers at age a and time t, $C_{a,t}$ is the catch in numbers at age and time, and in their Bayesian framework they specify the priors

$$\begin{aligned} M_{a,t} &\sim \log\mathcal{N}\left(\mu_a, \tau^2\right), \\ \mu_a &\sim \mathcal{N}\left(\nu, \sigma^2\right), \\ \tau^2 &\sim \mathcal{IG}\left(\alpha, \beta\right). \end{aligned}$$

In this way, setting $\nu = \log(0.2)$ gives each $M_{a,t}$ a prior median of 0.2 and a high value of σ^2 represents little prior confidence that each $M_{a,t}$ is close to 0.2 and independent of age. Similarly, a high variance hyper-prior for τ^2 suggests that we have little prior belief in $M_{a,t}$ being constant over time within each age-class. Their model was developed for an example where

- data are indices of numbers-at-age from research vessel surveys ($I_{a,t} = \log\mathcal{N}\left(\log\left(q_a N_{a,t}\right), \sigma^2\right)$)
- reported catch is provided as numbers-at-age ($C_{a,t}$)

This is a little bit of an unrealistic data set by New Zealand standards as age-structured catch and therefore CPUE data are very hard to come by. They also admit that, for a variety of reasons, "it would be inappropriate to use the results from their model for inference about current stock status or for risk management".

Despite considerable interest in state-space models, and previous work suggesting that they have superior performance when compared with deterministic models (Millar & Meyer 2000), they are not widely used, likely

due to their added complexity in implementation. Quinn (1992) raises the point that "few statisticians have been involved in fisheries modelling and that this has resulted in a whole class of methods unlike anything found in the mainstream statistics literature". Whether or not this is a good thing is up for discussion.

Much of the existing work on state-space modelling in fisheries has used maximum likelihood inference methods. However, it can be difficult to estimate both observation and process error simultaneously using maximum likelihood methods (Meyer & Millar 1999). Therefore, sticking to the theme of this thesis, we limit our inference to Bayesian methods only.

Before we get into fisheries state-space models, we present a very simple (non state-space) example to illustrate some concepts in basic inference. We then introduce state-space fisheries models with a brief example using biomass dynamics models. These models were first introduced in Chapter 1, page 35. Biomass dynamics models are the the method of choice (and really the only method available) if the only data available include a times series of catches and an index of abundance (Punt 2003*a*). Finally, we reformulate age-structured models, first introduced in Chapter 1, page 37, to include process error associated with the numbers of fish at age and time ($N_{a,t}$) within the population. The major contribution provided in this chapter is the construction of the posterior for a state-space age-structured model. While this model has the potential to better represent uncertainty in stock assessment, we warn that its practical application remains limited as MCMC mixing was unsatisfactory. The future challenge is to sample from this posterior efficiently.

5.2 A simple example

We begin with a very simple (non state-space) model that we use to illustrate the accuracy that can be expected when simulating from a model, then back estimating parameters used in the simulation model. The model

has parameters a and σ_ε^2 and covariates z_t for time steps $t = 1, \ldots, T$

$$y_t = a z_t e^{\varepsilon_t - \sigma_\varepsilon^2/2} \quad \text{where} \quad \varepsilon_t \overset{\text{iid}}{\sim} \mathcal{N}\left(0, \sigma_\varepsilon^2\right), \tag{5.1}$$

where $-\sigma_\varepsilon^2/2$ is a log-normal correction term (see Appendix A.2 for proof). The observations (y_t) are therefore assumed to be log-normal random variables

$$\log\left(y_t\right) | z_t, a, \sigma_\varepsilon^2 \sim \mathcal{N}\left(\log\left(a z_t\right) - \sigma_\varepsilon^2/2, \sigma_\varepsilon^2\right). \tag{5.2}$$

We simulate from this model for $T = 100$ time steps, setting $a = 3$ and $\sigma_\varepsilon = 0.1$.

We are interested in the probabilistic relationship between the following:

- **The data:** $\mathbf{y} = \{y_t\}_{t=1}^T$
- **The covariates:** $\mathbf{z} = \{z_t\}_{t=1}^T$
- **The parameters of interest:** $\boldsymbol{\theta} = \{a, \sigma_\varepsilon^2\}$

Using Bayes theorem, the posterior distribution of the model parameters ($\boldsymbol{\theta}$), given the data ($\mathbf{y}$) and covariates ($\mathbf{z}$) is

$$\pi\left(\boldsymbol{\theta}|\mathbf{y}, \mathbf{z}\right) \propto \pi\left(\boldsymbol{\theta}\right) \pi\left(\mathbf{y}|\mathbf{z}, \boldsymbol{\theta}\right), \tag{5.3}$$

where the prior and the likelihood are

$$\pi\left(\boldsymbol{\theta}\right) = \pi\left(a, \sigma_\varepsilon^2\right) = \pi(a)\pi\left(\sigma_\varepsilon^2\right),$$
$$\pi\left(\mathbf{y}|\mathbf{z}, \boldsymbol{\theta}\right) = \pi\left(\mathbf{y}|\mathbf{z}, a, \sigma_\varepsilon^2\right) = \prod_{t=1}^{T} \pi\left(y_t|z_t, a, \sigma_\varepsilon^2\right).$$

Using MCMC we can sample from the posterior distribution (Equation 5.3) to obtain probability distributions of the model parameters. We placed uniform priors with wide bounds on both parameters specifying $a \sim \mathcal{U}(-9e99, 9e99)$ and $\sigma_\varepsilon \sim \mathcal{U}(0, 9e99)$. We ran our MCMC for 100,000 iterations, with a thinning rate of 10 resulting a sample size of 10,000. The MCMC burn-in was 1000 iterations and the standard deviation of the log-normal proposal distribution was set to $\sigma_q = 0.05$ (see Chapter 2, page 66 for a description of MCMC using a log-normal proposal distribution). We repeat using different random number seeds in three different simulated

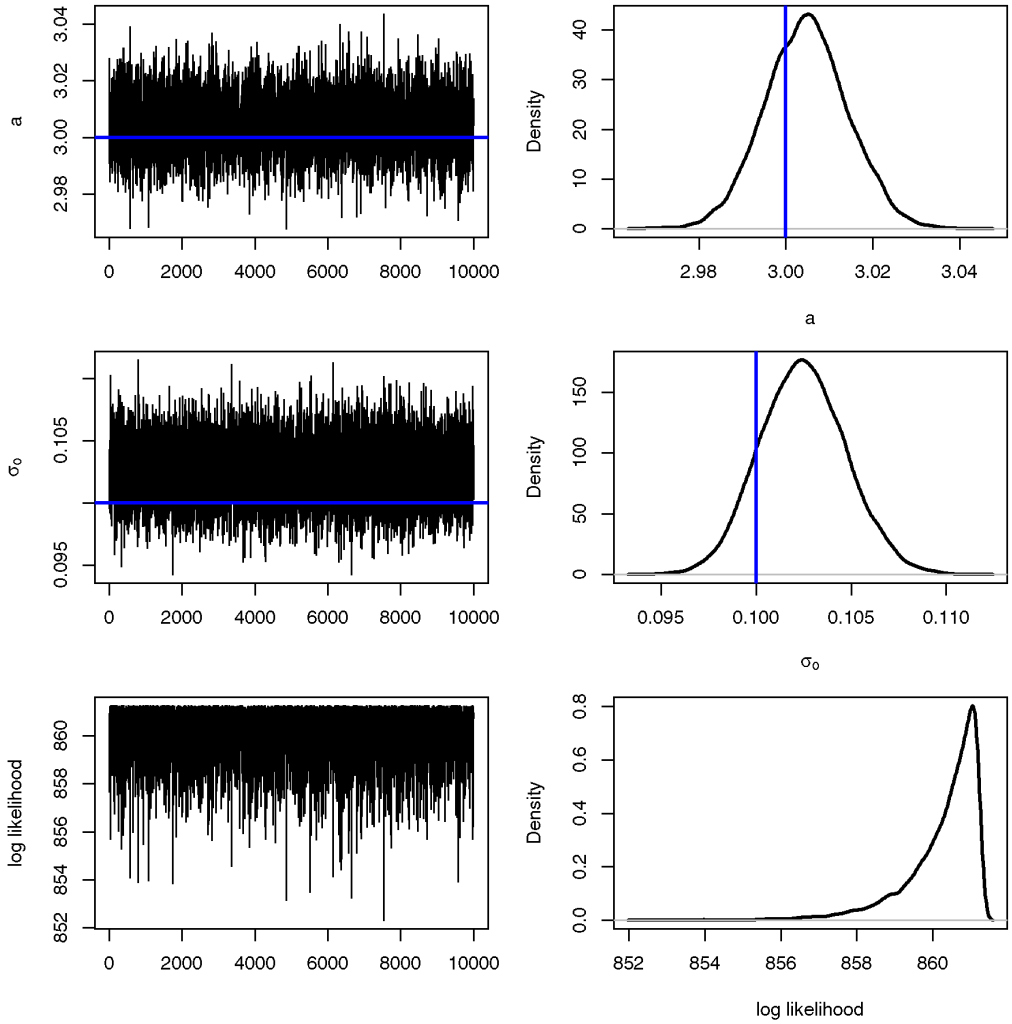


Figure 5.1: MCMC trace plots [left] and posterior densities [right] for the model parameters (a and σ_ε) and the log-likelihood of the model. The true values of the model parameters in the simulation are indicated as solid blue lines in the top two panels. The log-prior was not plotted as it was constant in this example.

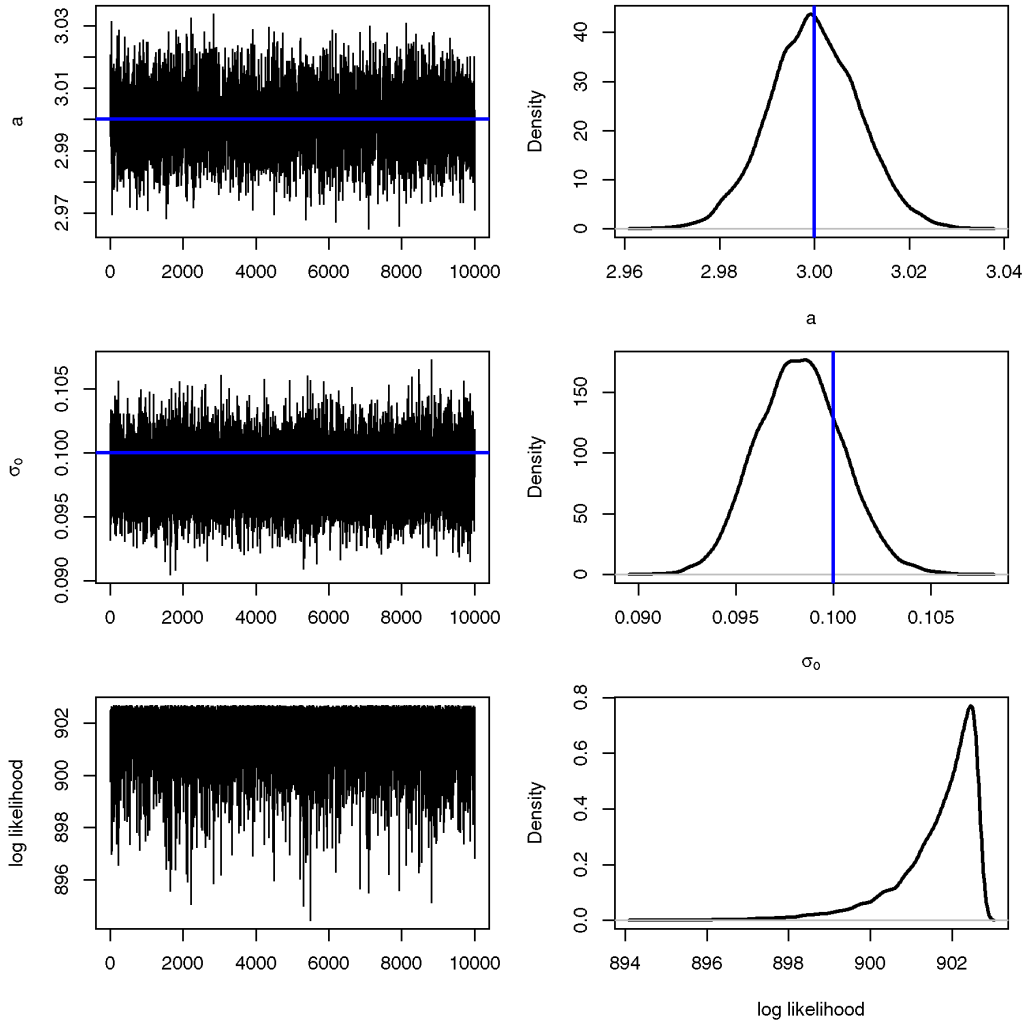


Figure 5.2: MCMC trace plots [left] and posterior densities [right] for the model parameters (a and σ_ε) and the log-likelihood of the model. The true values of the model parameters in the simulation are indicated as solid blue lines in the top two panels. The log-prior was not plotted as it was constant in this example.

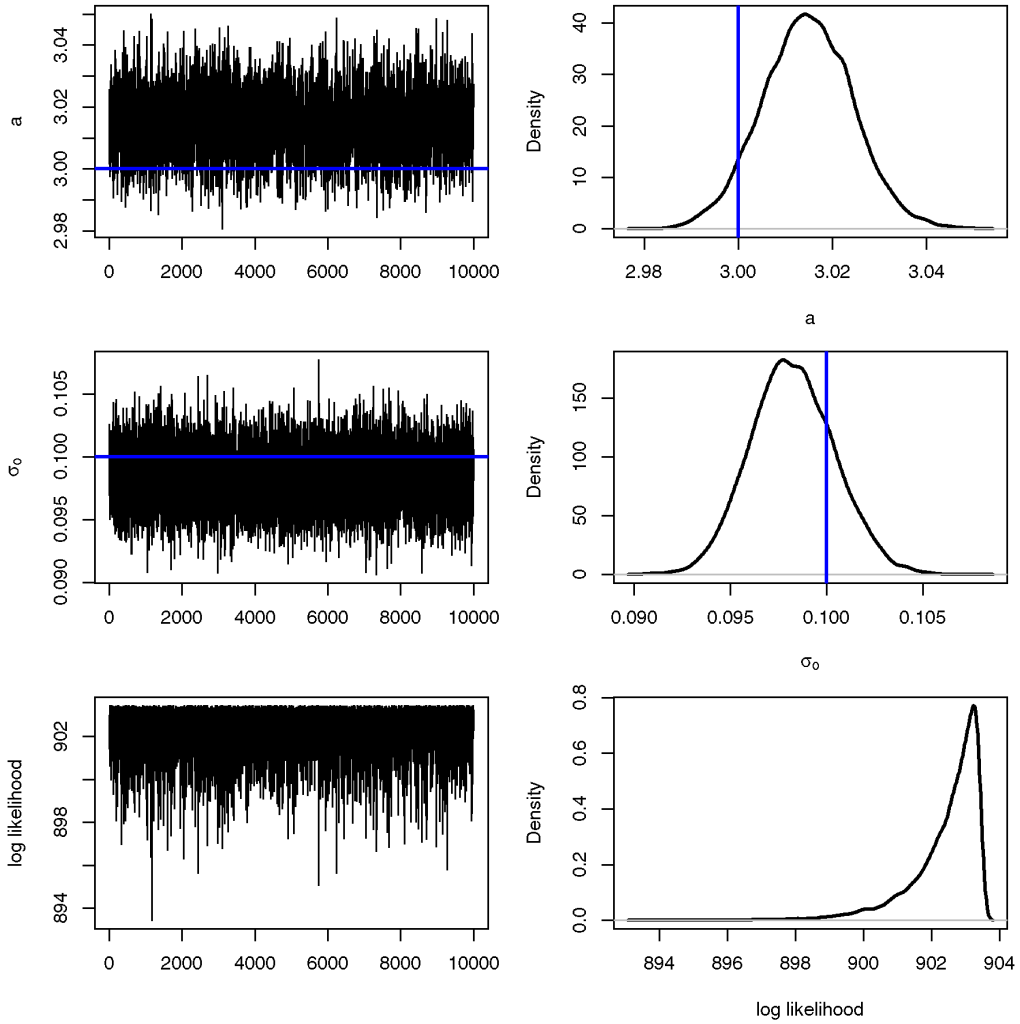


Figure 5.3: MCMC trace plots [left] and posterior densities [right] for the model parameters (a and σ_ε) and the log-likelihood of the model. The true values of the model parameters in the simulation are indicated as solid blue lines in the top two panels. The log-prior was not plotted as it was constant in this example.

data sets. MCMC trace plots and densities are shown in Figures 5.1, 5.2 and 5.3. These results show a model that is working well. All chains have converged and the true values are within credible intervals.

5.3 Biomass dynamics state-space models

A biomass dynamics state-space model relates catch per unit effort observations (I_t) to the unobserved biomass states (B_t) through a stochastic observation model for I_t given by

$$I_t = qB_t e^{\varepsilon_t^o - \sigma_o^2/2} \quad \text{where} \quad \varepsilon_t^o \overset{\text{iid}}{\sim} \mathcal{N}\left(0, \sigma_o^2\right), \tag{5.4}$$

where q is the catchability coefficient and ε_t^o is the normally distributed observation error at time t with variance σ_o^2. The states are assumed to follow a stochastic transition model of surplus production

$$B_t = \left(B_{t-1} + rB_{t-1}\left(1 - \frac{B_{t-1}}{K}\right) - C_{t-1}\right) e^{\varepsilon_t^p - \sigma_p^2/2} \quad \text{where} \quad \varepsilon_t^p \overset{\text{iid}}{\sim} \mathcal{N}\left(0, \sigma_p^2\right), \tag{5.5}$$

where C_t is the catch (tonnes) at time t, r is the intrinsic rate of population increase, K is the carrying capacity (tonnes), and ε_t^p is the normally distributed process error at time t with variance σ_p^2.

Due to known difficulties with the parameterisation in Equation 5.5 leading to poor performance in the Metropolis-Hastings sampler it is common practice to reparameterise the model by replacing the states B_t with $J_t = B_t/K$ (Meyer & Millar 1999). The new states are the ratio of biomass to carrying capacity also known as depletion. Thus B_t is replaced by KJ_t, and the form becomes

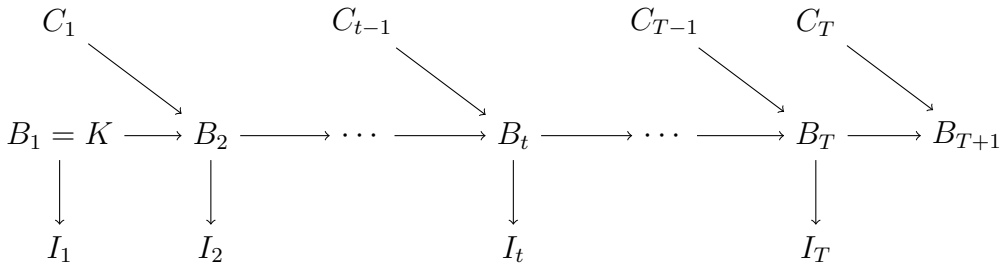
$$\begin{aligned} \log(J_t) &\sim \mathcal{N}\left(\log\left(\mu_1\right) - \sigma_p^2/2, \lambda_1\sigma_p^2\right) && t = 1, \\ J_t &= \left(J_{t-1} + rJ_{t-1}(1 - J_{t-1}) - \frac{C_{t-1}}{K}\right) e^{\varepsilon_t^p - \sigma_p^2/2} && t = 2, \ldots, T, \\ I_t &= qKJ_t e^{\varepsilon_t^o - \sigma_o^2/2} && \forall t. \end{aligned} \tag{5.6}$$

The states (J_t) and observations (I_t) are therefore assumed to be log-

normal random variables

$$\begin{aligned}
\log(J_1)|\mu_1, \lambda_1, \sigma_p^2 &\sim \mathcal{N}\left(\log\left(\mu_1\right) - \sigma_p^2/2, \lambda_1\sigma_p^2\right), \\
\log(J_t)|J_{t-1}, r, K, \sigma_p^2, C_{t-1} &\sim \mathcal{N}\left(\log\left(J_{t-1} + rJ_{t-1}(1 - J_{t-1}) - \frac{C_{t-1}}{K}\right) - \sigma_p^2/2, \sigma_p^2\right), \\
\log(I_t)|K, q, \sigma_o^2, J_t &\sim \mathcal{N}\left(\log\left(qKJ_t\right) - \sigma_o^2/2, \sigma_o^2\right). \qquad (5.7)
\end{aligned}$$

Usually it is assumed that $\mu_1 = 0$ and $\lambda_1 = 1$ (i.e. the biomass in the first year is at carrying capacity $B_1 = K$) and we make this assumption here. This model can be represented graphically as:



In summary, this model makes the following assumptions:

- the parameters r, K, and q are constant over time t
- the intrinsic rate of population increase (r) is independent of the age/size composition of the population
- the population is at its carrying capacity (K) at the start of the model (i.e. $B_1 = K$)
- the population is closed (no immigration or emigration)
- fishing and natural mortality are applied simultaneously
- the catch (C_t) is measured without error
- the CPUE (I_t) is proportional to abundance (B_t).

5.3.1 Inference

We are interested in the probabilistic relationship between the following:

- **The data y:** the catch per unit effort (I_t). Let $\mathbf{y} = \{I_t\}_{t=1}^T$

- **The covariates z:** the catch (C_t). Let $\mathbf{z} = \{C_t\}_{t=1}^{T}$
- **The unknown parameters of interest $\boldsymbol{\theta}$:** the intrinsic rate of population increase (r) and the carrying capacity of the population (K). Let $\boldsymbol{\theta} = \{r, K\}$
- **The unknown nuisance parameters $\boldsymbol{\omega}$:** the catchability coefficient (q) and the observation and process error variances (σ_o^2 and σ_p^2). Let $\boldsymbol{\omega} = \{q, \sigma_o^2, \sigma_p^2\}$
- **The unknown latent states x:** the depletion (J_t). Let $\mathbf{x} = \{J_t\}_{t=1}^{T}$

Using Bayes theorem, the posterior distribution of the model parameters ($\boldsymbol{\theta}$ and $\boldsymbol{\omega}$) and the states ($\mathbf{x}$), given the data ($\mathbf{y}$) and covariates ($\mathbf{z}$) is

$$\pi(\boldsymbol{\theta}, \boldsymbol{\omega}, \mathbf{x}|\mathbf{y}, \mathbf{z}) \propto \pi(\boldsymbol{\theta}, \boldsymbol{\omega}, \mathbf{x}|\mathbf{z})\pi(\mathbf{y}|\boldsymbol{\theta}, \boldsymbol{\omega}, \mathbf{x}), \tag{5.8}$$

where

$$\begin{aligned}
\pi\left(\boldsymbol{\theta}, \boldsymbol{\omega}, \mathbf{x}|\mathbf{z}\right) &= \pi\left(r, K, q, \sigma_o^2, \sigma_p^2, \mathbf{x}|\mathbf{z}\right) \\
&= \pi(r)\pi(K)\pi(q)\pi(\sigma_o^2)\pi(\sigma_p^2)\pi\left(\mathbf{x}|\mathbf{z}, r, K, \sigma_p^2\right) \\
&= \pi(r)\pi(K)\pi(q)\pi(\sigma_o^2)\pi(\sigma_p^2)\pi\left(J_1|K, \sigma_p^2\right)\prod_{t=2}^{T}\pi\left(J_t|J_{t-1}, C_{t-1}, r, K, \sigma_p^2\right), \\
\pi\left(\mathbf{y}|\mathbf{x}, \boldsymbol{\theta}, \boldsymbol{\omega}\right) &= \pi\left(\mathbf{y}|\mathbf{x}, K, q, \sigma_o^2\right) = \prod_{t=1}^{T}\pi\left(I_t|J_t, K, q, \sigma_o^2\right). \qquad (5.9)
\end{aligned}$$

Using MCMC we can sample from the posterior distribution (Equation 5.8) to obtain probability distributions of the parameters, latent states and other quantities of interest in the model (e.g. the biomass B_t). We provide an example below (Section 5.3.2).

Using Equations 5.7 and 5.8 and the priors described below we estimate posterior distributions for each of the parameters using blockwise Metropolis-Hastings using log-normal proposals (see Chapter 2, page 66). It is well known that the parameters r and K are highly correlated. To reduce this correlation and improve MCMC performance we set $\phi_1 = rK$ and $\phi_2 = r/K$ and propose ϕ_1 and ϕ_2 in the MCMC instead (Gilks & Roberts 1996). One can solve for r and K when required

$$r = \sqrt{\phi_1\phi_2} \qquad \text{and} \qquad K = \sqrt{\frac{\phi_1}{\phi_2}}.$$

If a transformation like this is used then the priors for these two parameters must be updated using the Jacobian

$$\left| \begin{array}{cc} K & r \\ \frac{1}{K} & -\frac{r}{K^2} \end{array} \right| = \left| \frac{\partial(\phi_1, \phi_2)}{\partial(r, K)} \right| = \left| -\frac{r}{K} - \frac{r}{K} \right| = \left| -\frac{2r}{K} \right| = 2\phi_2,$$

thus the prior for the two parameters becomes

$$\pi(\phi_1, \phi_2) = \pi(r)\pi(K) \left| \frac{\partial(r, K)}{\partial(\phi_1, \phi_2)} \right| = \pi(r)\pi(K)\frac{1}{2\phi_2} = \pi(r)\pi(K)\frac{K}{2r}.$$

See Chapter 2, page 67, for notes on parameter transformations in MCMC. Therefore, the acceptance probability is

$$\begin{aligned} r &= \min\left(1, \frac{\pi(\mathbf{y}|\boldsymbol{\theta}^*)}{\pi(\mathbf{y}|\boldsymbol{\theta}^{(i-1)})} \times \frac{\pi(\boldsymbol{\theta}^*)}{\pi(\boldsymbol{\theta}^{(i-1)})} \times \frac{q_\theta(\boldsymbol{\theta}^{(i-1)}|\boldsymbol{\theta}^*, \mathbf{y})}{q_\theta(\boldsymbol{\theta}^*|\boldsymbol{\theta}^{(i-1)}, \mathbf{y})}\right) \\ &= \min\left(1, \frac{\pi(\mathbf{y}|\boldsymbol{\theta}^*)}{\pi(\mathbf{y}|\boldsymbol{\theta}^{(i-1)})} \times \frac{\pi(\boldsymbol{\theta}^*)}{\pi(\boldsymbol{\theta}^{(i-1)})} \times \frac{\theta^*}{\theta^{(i-1)}}\right) \\ &= \min\left(1, \frac{\pi(\mathbf{y}|\boldsymbol{\theta}^*)}{\pi(\mathbf{y}|\boldsymbol{\theta}^{(i-1)})} \times \frac{\pi(r^*)}{\pi(r^{(i-1)})} \times \frac{\pi(K^*)}{\pi(K^{(i-1)})} \times \frac{\pi(q^*)}{\pi(q^{(i-1)})} \times \frac{\pi(\sigma_o^{2*})}{\pi(\sigma_o^{2(i-1)})}\right. \\ &\qquad \left. \times \frac{\pi(\sigma_p^{2*})}{\pi(\sigma_p^{2(i-1)})} \times \frac{\frac{K^*}{2r^*}}{\frac{K^{(i-1)}}{2r^{(i-1)}}} \times \frac{\theta^*}{\theta^{(i-1)}}\right). \end{aligned}$$

5.3.2 Packhorse rock lobster example

Simulation

We simulate data based on the packhorse rock lobster (*Sagmariasus verreauxi*, PHC) fishery in northern New Zealand using Equations 5.4 and 5.5 (for more information on the fishery see Chapter 3, page 80). We use a set of plausible parameter values in each simulation (Table 5.1). There is always considerable uncertainty in estimates of the observation and process error within these models (McAllister et al. 2001), so two different simulations were done setting the observation and process error standard deviations (σ_o and σ_p) to **low** and **high** values. The high values might sit somewhere close to what is realistic. It is important that there is contrast (i.e. a clear trend over time) in the CPUE time series to get a good fit

Table 5.1: Parameter values used in packhorse rock lobster simulation.

Parameter	Value	Units	Description
r	0.17	-	Intrinsic rate of population increase
K	1800	tonnes	Carrying capacity
q	0.002	tonnes^{-1}	Catchability coefficient
σ_o	0.01, 0.1	-	Observation error standard deviation
σ_p	0.001, 0.01	tonnes	Process error standard deviation

using a biomass dynamics models (this is also true of age-structured models, Hilborn & Walters 1992). The actual catch history of the packhorse rock lobster fishery is used in both simulation runs and provides adequate contrast in the simulated CPUE when using the parameters specified above (Figure 5.4). The simulated CPUE (I_t) and biomass (B_t) are given for vari-

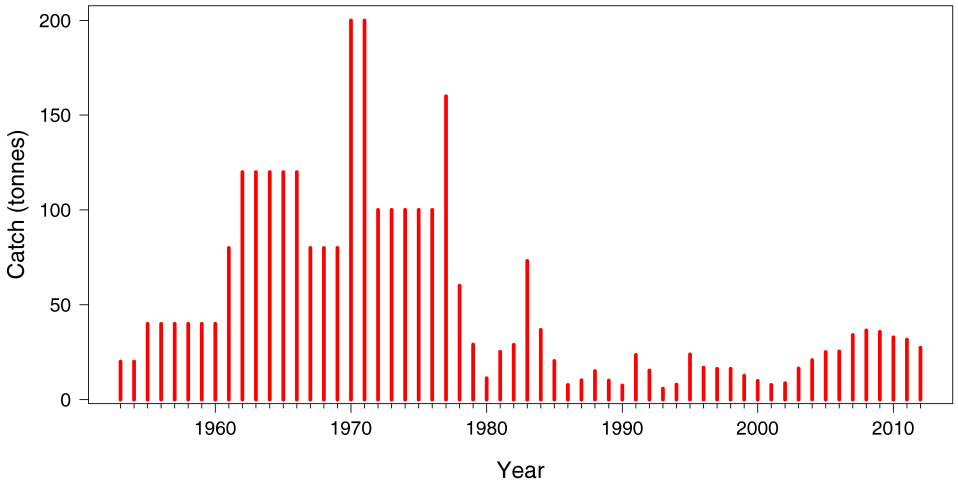


Figure 5.4: True catch history (C_t, tonnes) of the packhorse rock lobster from 1953 to 2012 (60 years) used in the simulation.

ous scenarios in Figures 5.6, 5.8, 5.10, 5.12, 5.14 and 5.16 below. We discuss each of these scenarios in turn.

Estimation with highly informative priors

We begin by specifying highly informative priors for each of the model parameters to check for any problems in the model or MCMC specification. The full list of priors is

$$\begin{aligned}
\pi(r) &\sim \log\mathcal{N}(\log(0.17), 0.01), \\
\pi(K) &\sim \log\mathcal{N}(\log(1800), 0.005), \\
\pi(q) &\sim \log\mathcal{N}(\log(0.002), 0.005), \\
\pi(\sigma_o^2) &\sim \log\mathcal{N}(\log(0.01^2), 0.05) \quad \text{or} \quad \pi(\sigma_o^2) \sim \log\mathcal{N}(\log(0.1^2), 0.05), \\
\pi(\sigma_p) &\sim \log\mathcal{N}(\log(0.001), 0.1) \quad \text{or} \quad \pi(\sigma_p) \sim \log\mathcal{N}(\log(0.01), 0.1),
\end{aligned} \tag{5.10}$$

noting that priors are applied to the variance of observation error (σ_o^2) and to the standard deviation of process error (σ_p).

We ran three million iterations using blockwise Metropolis-Hastings with log-normal proposals (see Chapter 2, page 66). These three million iterations were run as three separate MCMC chains on three different computer cores, all initialised with different random number seeds. Each chain ran an additional burn-in of two million iterations. The chains were thinned to save every 1000th iteration. This resulted in a total of 3000 samples of the posterior distribution for each of the parameters and latent states.

In the **low** observation and process error model ($\sigma_o = 0.01, \sigma_p = 0.001$), all parameters were recovered well with the mode of the posterior distribution for each parameter centered near the true value (Figure 5.5). The densities of the observation error variance (σ_o^2) and standard deviation of process error (σ_p) are not constrained by the data and MCMC is simply recovering the prior. The mode of the catchability coefficient (q) is the parameter furtherest from the true value set in the simulation. Despite this, the model fits the CPUE (I_t) observations well and results in a good match between the posterior biomass trajectory (B_t) and the simulated biomass trajectory (Figure 5.6).

In the **high** observation and process error model ($\sigma_o = 0.1, \sigma_p = 0.01$), the parameters were also well estimated (Figure 5.7). Again, the densities of the observation error variance (σ_o^2) and standard deviation of process error

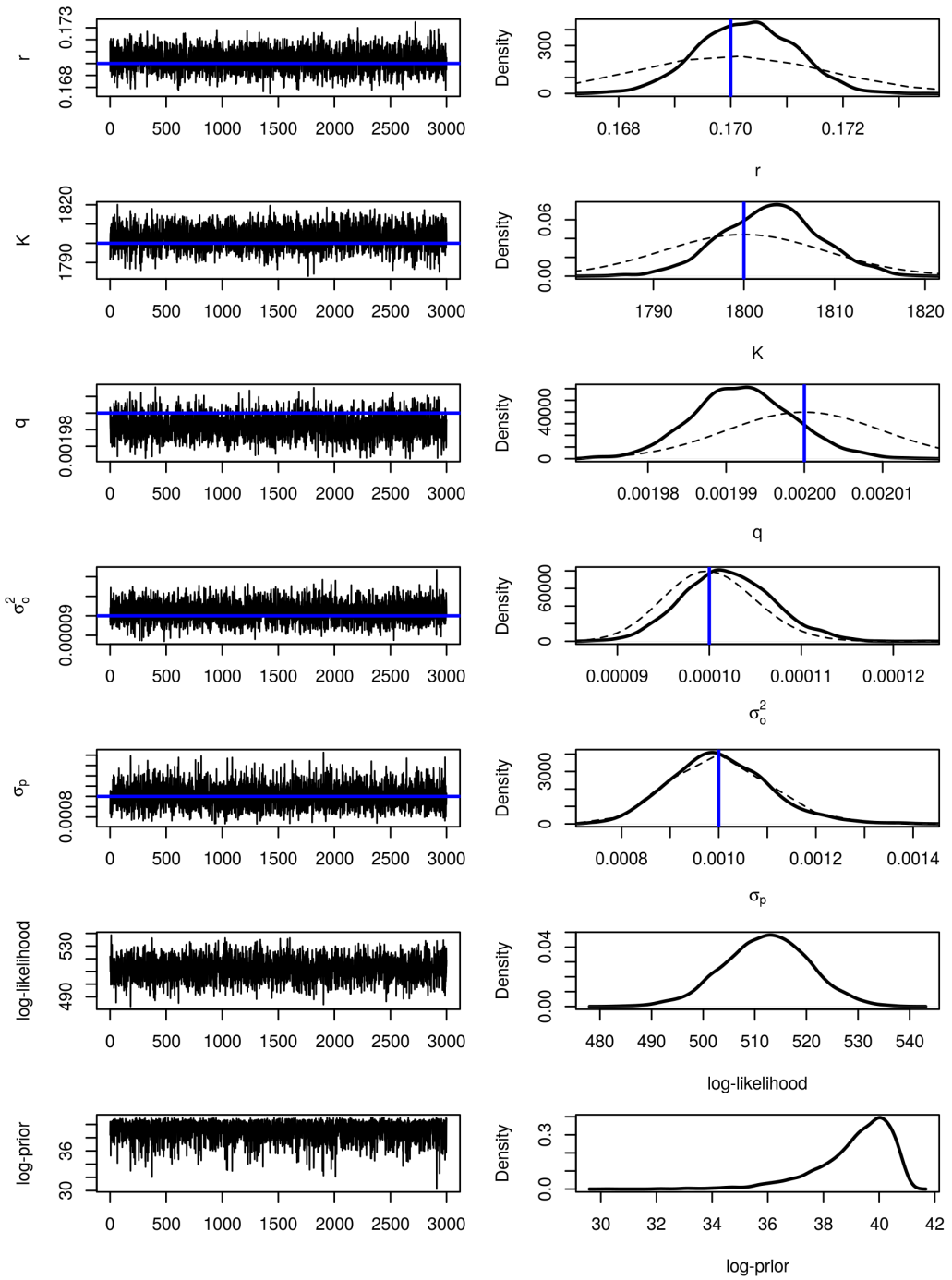


Figure 5.5: MCMC trace plots and posterior densities for the model parameters using data from the **low** observation error ($\sigma_o = 0.01$) and process error ($\sigma_p = 0.001$) model estimated using **highly informative priors**. Posterior traces and densities are indicated as black lines, priors as dashed black lines, and values specified in simulation as solid blue lines.

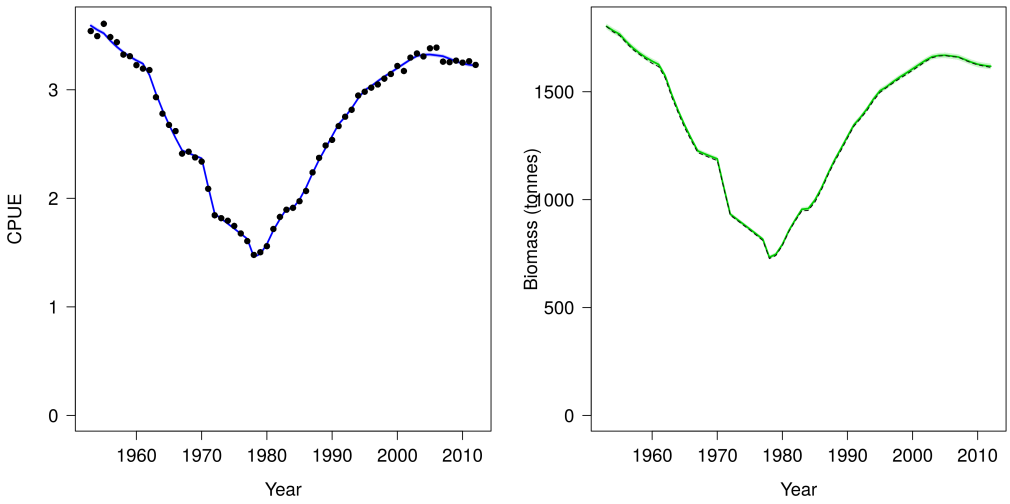


Figure 5.6: Fit to CPUE observations (I_t) [left] and the posterior distribution of biomass (B_t) [right] for the **low** observation error ($\sigma_p = 0.01$) and process error ($\sigma_p = 0.001$) model estimated using **highly informative priors**. CPUE observations are shown as black points [•] and the posterior distribution of the fit to CPUE is shown in blue. The posterior distribution of biomass is shown in green and the simulated biomass as the dashed black line. The shading indicates the 5th, 25th, 50th, 75th and 95th percentiles.

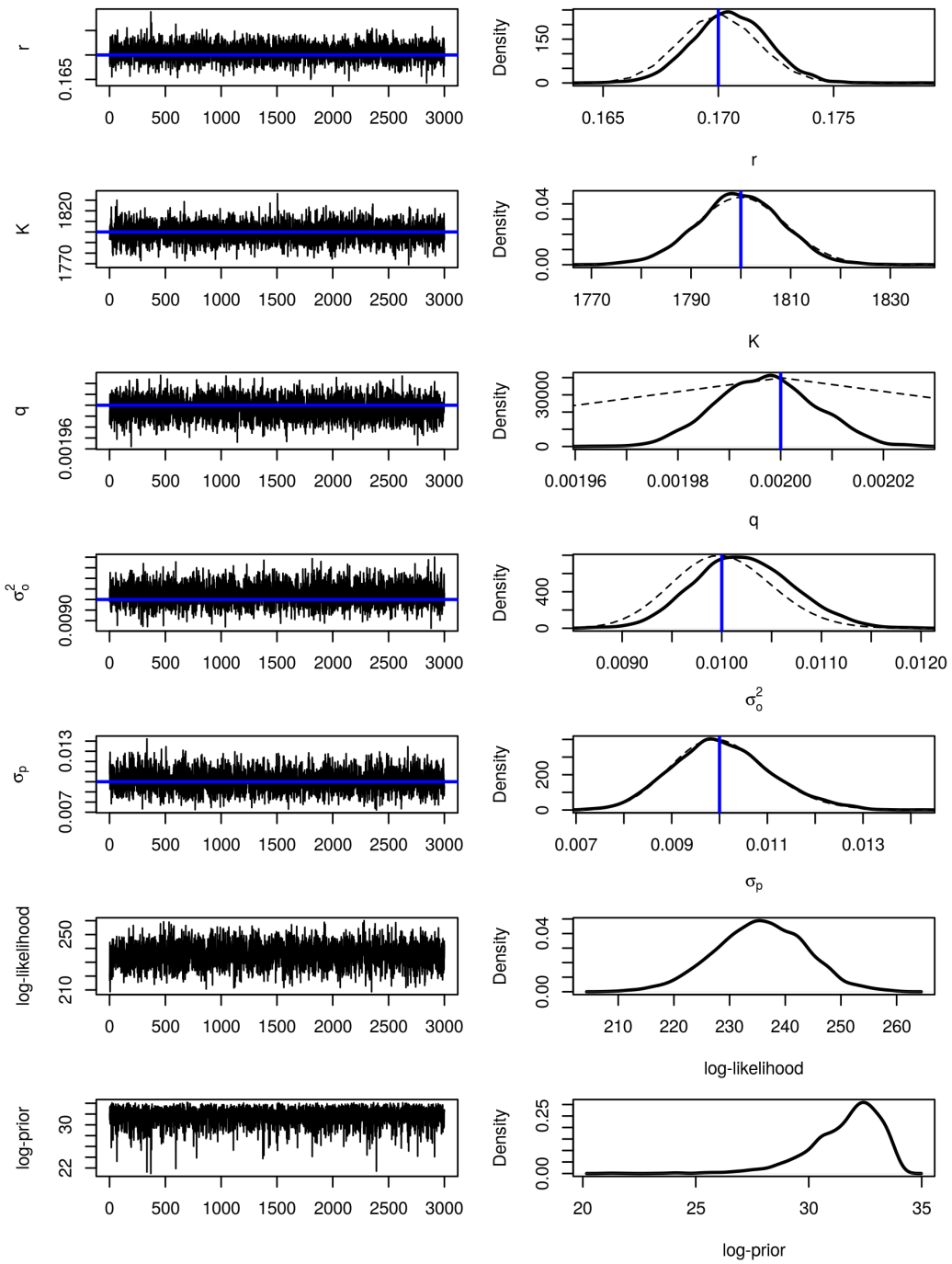


Figure 5.7: MCMC trace plots and posterior densities for the model parameters using data from the **high** observation error ($\sigma_o = 0.1$) and process error ($\sigma_p = 0.01$) model estimated using **highly informative priors**. Posterior traces and densities are indicated as black lines, priors as dashed black lines, and values specified in simulation as solid blue lines.

(σ_p) are not constrained by the data and MCMC is simply recovering the prior. The model fits the CPUE (I_t) observations well and the estimated biomass trajectory (B_t) is biased high only a little when compared with the simulated biomass trajectory (Figure 5.8).

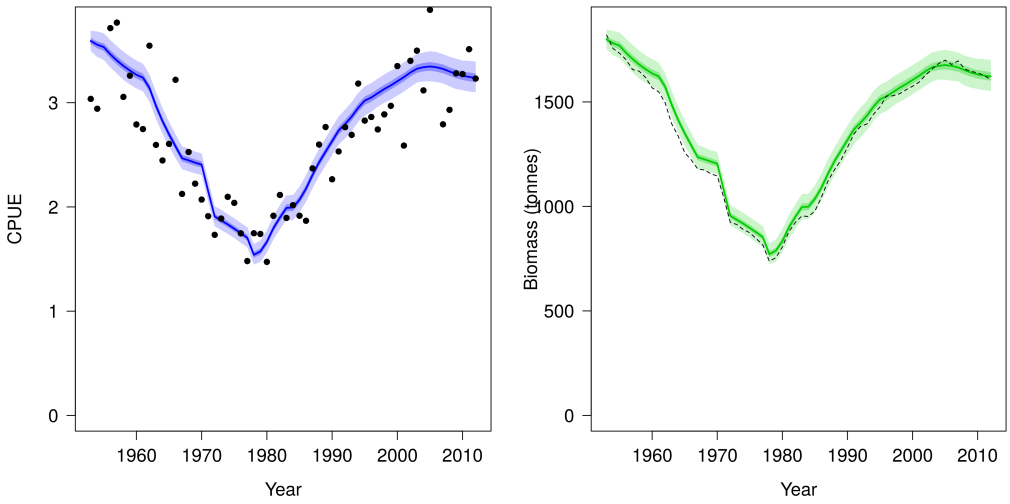


Figure 5.8: Fit to CPUE observations (I_t) [left] and the posterior distribution of biomass (B_t) [right] for the **high** observation error ($\sigma_p = 0.1$) and process error ($\sigma_p = 0.01$) model estimated using **highly informative priors**. CPUE observations are shown as black points [•] and the posterior distribution of the fit to CPUE is shown in blue. The posterior distribution of biomass is shown in green and the simulated biomass as the dashed black line. The shading indicates the 5th, 25th, 50th, 75th and 95th percentiles.

Relaxing the priors on r, K and q

We relax the priors placed on r, K and q and repeat. Usually we have no prior knowledge of the value of K and q. It it standard practice in fisheries to use uniform priors with wide bounds for almost all model parameters so that the priors are as uninformative as possible (in the absence of any prior information of course). Instead of using improper uniform priors in

this way, which can lead to poor MCMC mixing and may result in a lack of convergence, we use a range of uninformative continuous proper priors. For the parameters K and q we develop uninformative log-normal priors. Instead of using a uniform with wide bounds a and b

$$X \sim \mathcal{U}(a, b),$$

we assume that there is some negligibly small probability that parameter X will be below a or above b (e.g. 1%) and use a log-normal prior

$$\log(X) \sim \mathcal{N}\left(\mu, \sigma^2\right) \quad \text{where} \quad \mathbb{E}[\log(X)] = \mu \quad \text{and} \quad \mathbb{V}[\log(X)] = \sigma.$$

We can write

$$P(X < a) = P\left(\log(X) < \log(a)\right) = P\left(z_\alpha < \frac{\log(a) - \mu}{\sigma}\right) = \alpha,$$
$$P(X < b) = P\left(\log(X) < \log(b)\right) = P\left(z_\beta < \frac{\log(b) - \mu}{\sigma}\right) = \beta,$$

or

$$z_\alpha = \frac{\log(a) - \mu}{\sigma} \qquad \text{and} \qquad z_\beta = \frac{\log(b) - \mu}{\sigma}.$$

Solving for μ and σ we get

$$\mu = \frac{z_\alpha \log(b) - z_\beta \log(a)}{z_\alpha - z_\beta} \qquad \text{and} \qquad \sigma = \frac{\log(b) - \log(a)}{z_\beta - z_\alpha}. \tag{5.11}$$

For example, setting $\alpha = 1\%$ and $\beta = 99\%$ would yield $z_\alpha = -2.326$ and $z_\beta = 2.326$.

For the parameter K we set $a = 100$ and $b = 10000$, with $\alpha = 1\%$ and $\beta = 99\%$. For q we set $a = 0.001$ and $b = 1$, with $\alpha = 1\%$ and $\beta = 99\%$. The priors derived from these are provided below (Equation 5.12). Finally, some work has gone into the development of informed priors for use in biomass dynamics models (McAllister et al. 2001), particularly for the parameter r. Therefore, we assume some prior knowledge of r. The full list of priors is

$$\begin{aligned}
\pi(r) &\sim \log\mathcal{N}(\log(0.17), 0.1), \\
\pi(K) &\sim \log\mathcal{N}(6.90776, 0.989933), \\
\pi(q) &\sim \log\mathcal{N}(-3.45388, 1.4849), \\
\pi(\sigma_o^2) &\sim \log\mathcal{N}(\log(0.01^2), 0.05) \quad \text{or} \quad \pi(\sigma_o^2) \sim \log\mathcal{N}(\log(0.1^2), 0.05), \\
\pi(\sigma_p) &\sim \log\mathcal{N}(\log(0.001), 0.1) \quad \text{or} \quad \pi(\sigma_p) \sim \log\mathcal{N}(\log(0.01), 0.1),
\end{aligned} \tag{5.12}$$

In the **low** observation and process error model ($\sigma_o = 0.01, \sigma_p = 0.001$), parameters were not recovered as well as before (Figure 5.9). The densities of the observation error variance (σ_o^2) and standard deviation of process error (σ_p) are not constrained by the data and the MCMC is simply recovering the prior. The mode of the catchability coefficient (q) and the carrying capacity (K) were quite far from the true value set in the simulation. Despite this, the model fits the CPUE (I_t) observations well and results in a good match between the posterior biomass trajectory (B_t) and the simulated biomass trajectory (Figure 5.10).

In the **high** observation and process error model ($\sigma_o = 0.1, \sigma_p = 0.01$), most parameters were not well recovered (Figure 5.11). Again, the densities of the observation error variance (σ_o^2) and standard deviation of process error (σ_p) are not constrained by the data and MCMC is simply recovering the prior. The catchability coefficient (q) was underestimated and the carrying capacity (K) overestimated. While the model fit to the CPUE observations (I_t) looks adequate, the biomass (B_t) is consistently overestimated (Figure 5.12).

Relaxing the priors on $\pi(\sigma_o^2)$ and $\pi(\sigma_p)$

The inverse gamma distribution with high variance is recommended as a good choice of uninformative prior for variance parameters (Gelman 2006). However, we found that MCMC generally resulted in us underestimating the observation error variance (σ_o^2) and over estimating the process error standard deviation (σ_p). Therefore, we develop a prior for σ_p. In doing so we consider how much it might be reasonable for the population biomass to be changing from year to year. We do this assuming a closed population (i.e. the effects of migration and other things that could affect the biomass are negligible). Let's say that in most circumstances we would not expect the population biomass to be different by more than 10% between any two years, after taking into account production and the catch. This would yield

$$0.9 \leq e^{\varepsilon_t^p} \leq 1.1 \qquad \text{or} \qquad \log(0.9) \leq \varepsilon_t^p \leq \log(1.1).$$

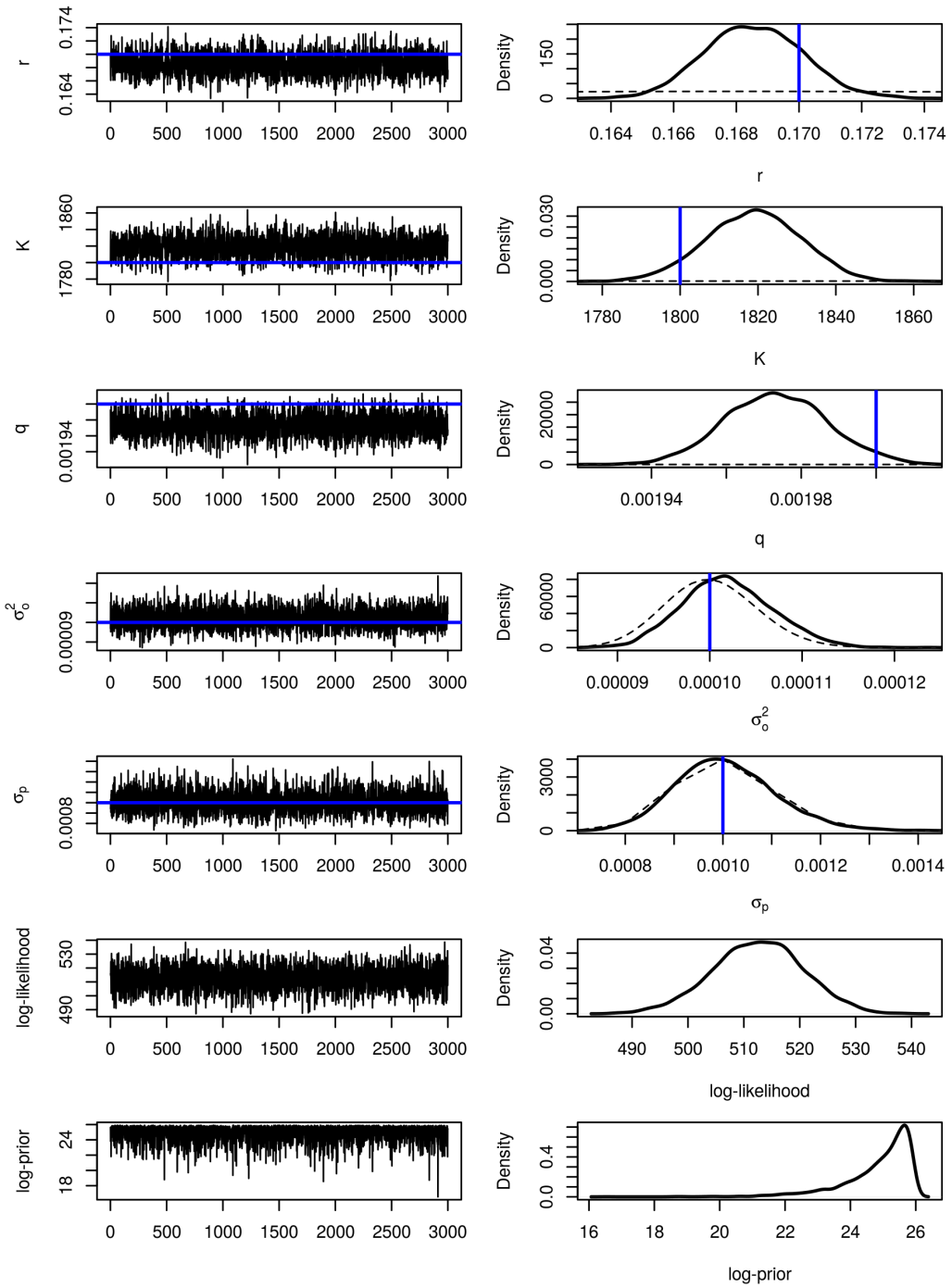


Figure 5.9: MCMC trace plots and posterior densities for the model parameters using data from the **low** observation error ($\sigma_o = 0.01$) and process error ($\sigma_p = 0.001$) model estimated using **informative priors** for observation error variance and process error standard deviation. Posterior traces and densities are indicated as black lines, priors as dashed black lines, and values specified in simulation as solid blue lines.

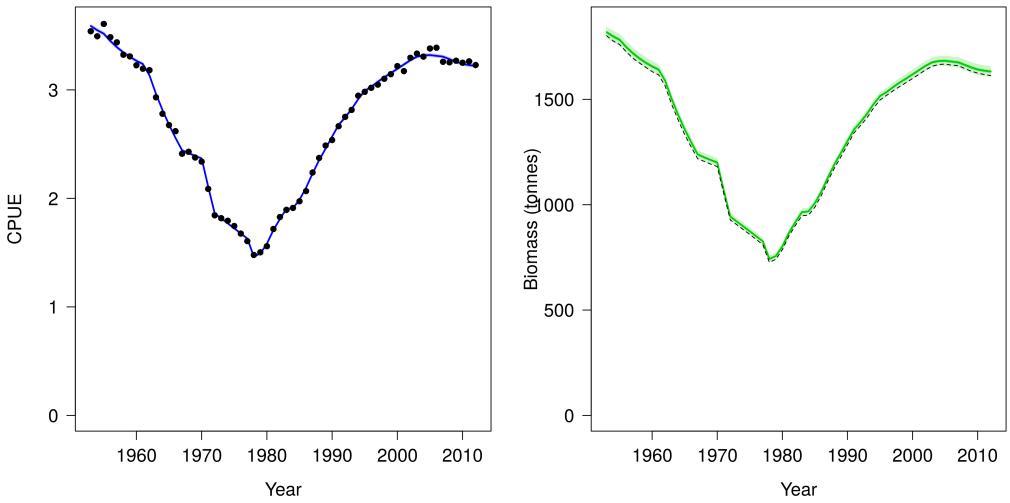


Figure 5.10: Fit to CPUE observations (I_t) [left] and the posterior distribution of biomass (B_t) [right] for the **low** observation error ($\sigma_p = 0.01$) and process error ($\sigma_p = 0.001$) model estimated using **informative priors** for observation error variance and process error standard deviation. CPUE observations are shown as black points [•] and the posterior distribution of the fit to CPUE is shown in blue. The posterior distribution of biomass is shown in green and the simulated biomass as the dashed black line. The shading indicates the 5th, 25th, 50th, 75th and 95th percentiles.

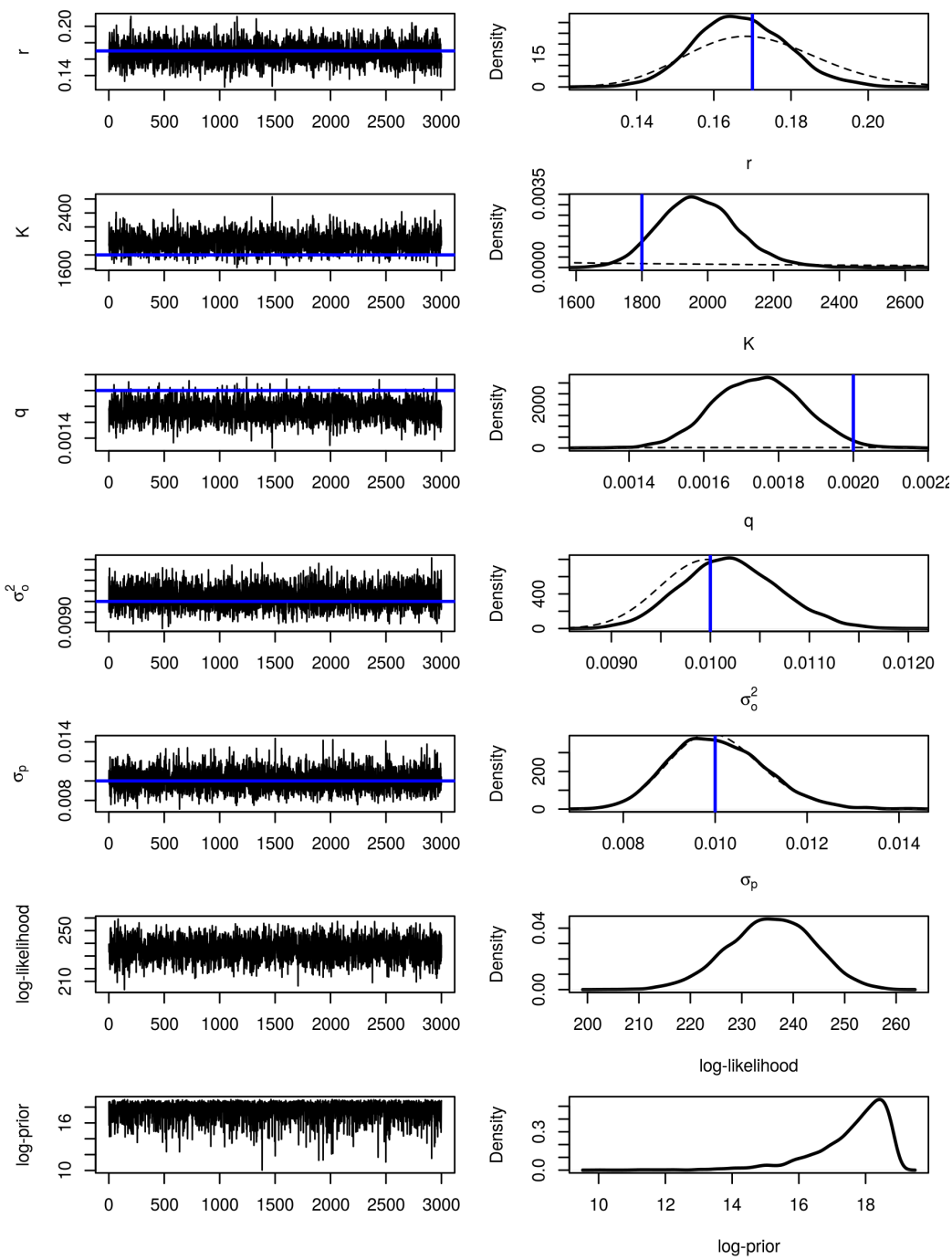


Figure 5.11: MCMC trace plots and posterior densities for the model parameters using data from the **high** observation error ($\sigma_o = 0.1$) and process error ($\sigma_p = 0.01$) model estimated using **informative priors** for observation error variance and process error standard deviation. Posterior traces and densities are indicated as black lines, priors as dashed black lines, and values specified in simulation as solid blue lines.

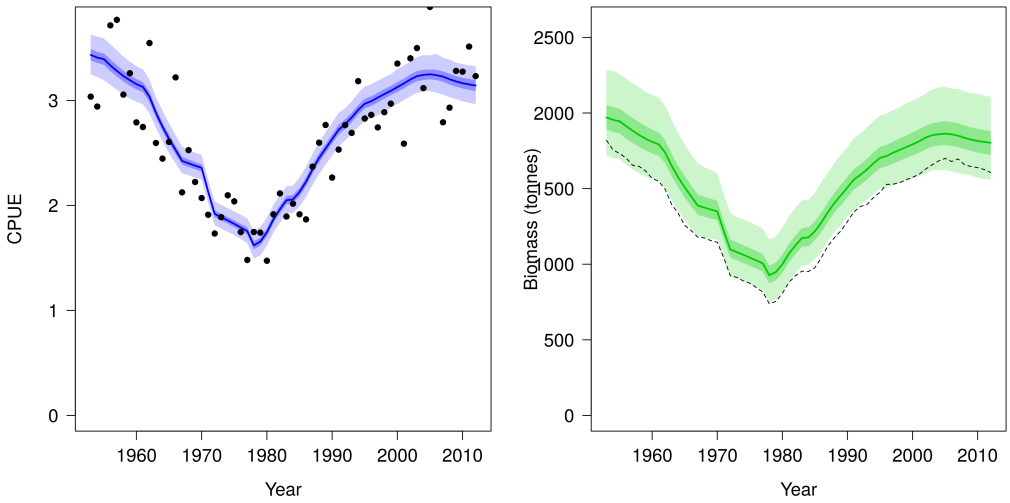


Figure 5.12: Fit to CPUE observations (I_t) [left] and the posterior distribution of biomass (B_t) [right] for the **high** observation error ($\sigma_p = 0.1$) and process error ($\sigma_p = 0.01$) model estimated using **informative priors** for observation error variance and process error standard deviation. CPUE observations are shown as black points [•] and the posterior distribution of the fit to CPUE is shown in blue. The posterior distribution of biomass is shown in green and the simulated biomass as the dashed black line. The shading indicates the 5th, 25th, 50th, 75th and 95th percentiles.

We can generalise this and write

$$\log(a) \leq e^{\varepsilon_t^p} \leq \log(b). \tag{5.13}$$

We then solve for the process error standard deviation (σ_p)

$$\sigma_p = \frac{\log(b) - \log(a)}{z_\beta - z_\alpha} \approx 0.043. \tag{5.14}$$

Thus, we state that σ_p should be less than or equal to the value derived above. To capture this property we place a gamma prior distribution on σ_p with $\alpha = 1$ (i.e. an exponential). Thus

$$\begin{aligned} \pi(\sigma_p) &\sim \mathcal{G}a(\alpha, \beta), \\ \mathbb{E}[\sigma_p] &= \frac{\alpha}{\beta} = \frac{1}{\beta} = 0.043, \\ \therefore \beta &\approx 23, \end{aligned} \tag{5.15}$$

The full list of priors is

$$\begin{aligned} \pi(r) &\sim \log\mathcal{N}(\log(0.17), 0.1), \\ \pi(K) &\sim \log\mathcal{N}(6.90776, 0.989933), \\ \pi(q) &\sim \log\mathcal{N}(-3.45388, 1.4849), \\ \pi(\sigma_o^2) &\sim \mathcal{IG}(0.001, 0.001), \\ \pi(\sigma_p) &\sim \mathcal{G}a(1, 23). \end{aligned} \tag{5.16}$$

The inverse gamma distributions density function is defined over the support $x > 0$ with shape parameter α and scale parameter β

$$f(x|\alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{-\alpha-1} \exp\left(-\frac{\beta}{x}\right),$$

where

$$\begin{aligned} \mathbb{E}(x) &= \frac{\beta}{\alpha + 1} \text{ for } \alpha > 1, \\ \mathbb{V}(x) &= \frac{\beta^2}{(\alpha-1)^2\alpha - 2()} \text{ for } \alpha > 2. \end{aligned}$$

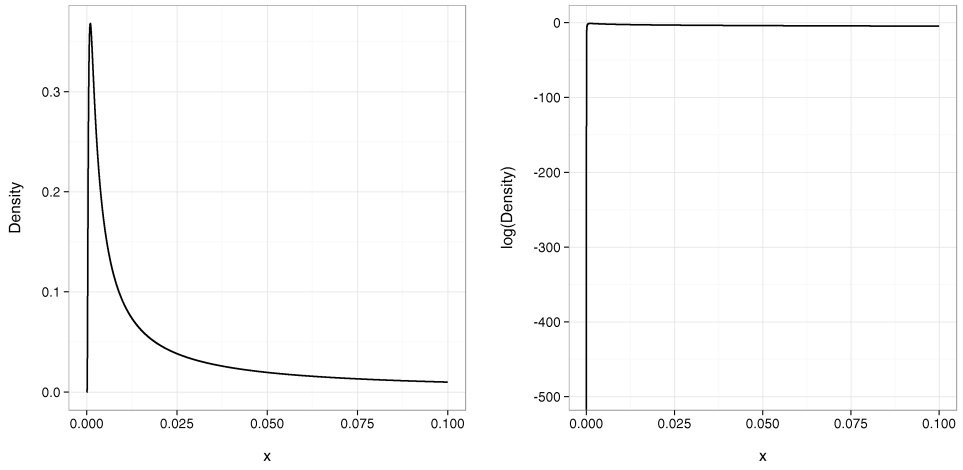
This distribution is commonly used as a weak prior for variance parameters with $\alpha = 0.001$ and $\beta = 0.001$ (Gelman 2006). Using the inverse gamma we get

$$\pi(\sigma^2|\alpha,\beta) \propto (\sigma^2)^{-\alpha-1} \exp\left(-\frac{\beta}{\sigma^2}\right).$$

As β and α approach zero then the inverse gamma will approach the Jeffreys prior

$$\pi(\sigma^2|\alpha,\beta) \propto \frac{1}{\sigma^2}.$$

We show the density and log-density below.



In the **low** observation and process error model ($\sigma_o = 0.01, \sigma_p = 0.001$), most parameters were recovered well with the mode of the posterior distribution for each parameter centered near the true value (Figure 5.13). However, the observation error variance (σ_o^2) was overestimated. Despite this, the model fits the CPUE (I_t) observations well and results in a good match between the posterior biomass trajectory (B_t) and the simulated biomass trajectory (Figure 5.14).

In the **high** observation and process error model ($\sigma_o = 0.1, \sigma_p = 0.01$), most parameters were not well recovered (Figure 5.15). The observation error variance (σ_o^2) was overestimated, while the standard deviation of process error (σ_p^2) was underestimated due to the influence of the prior. The density of the standard deviation of process error (σ_p) is not constrained by

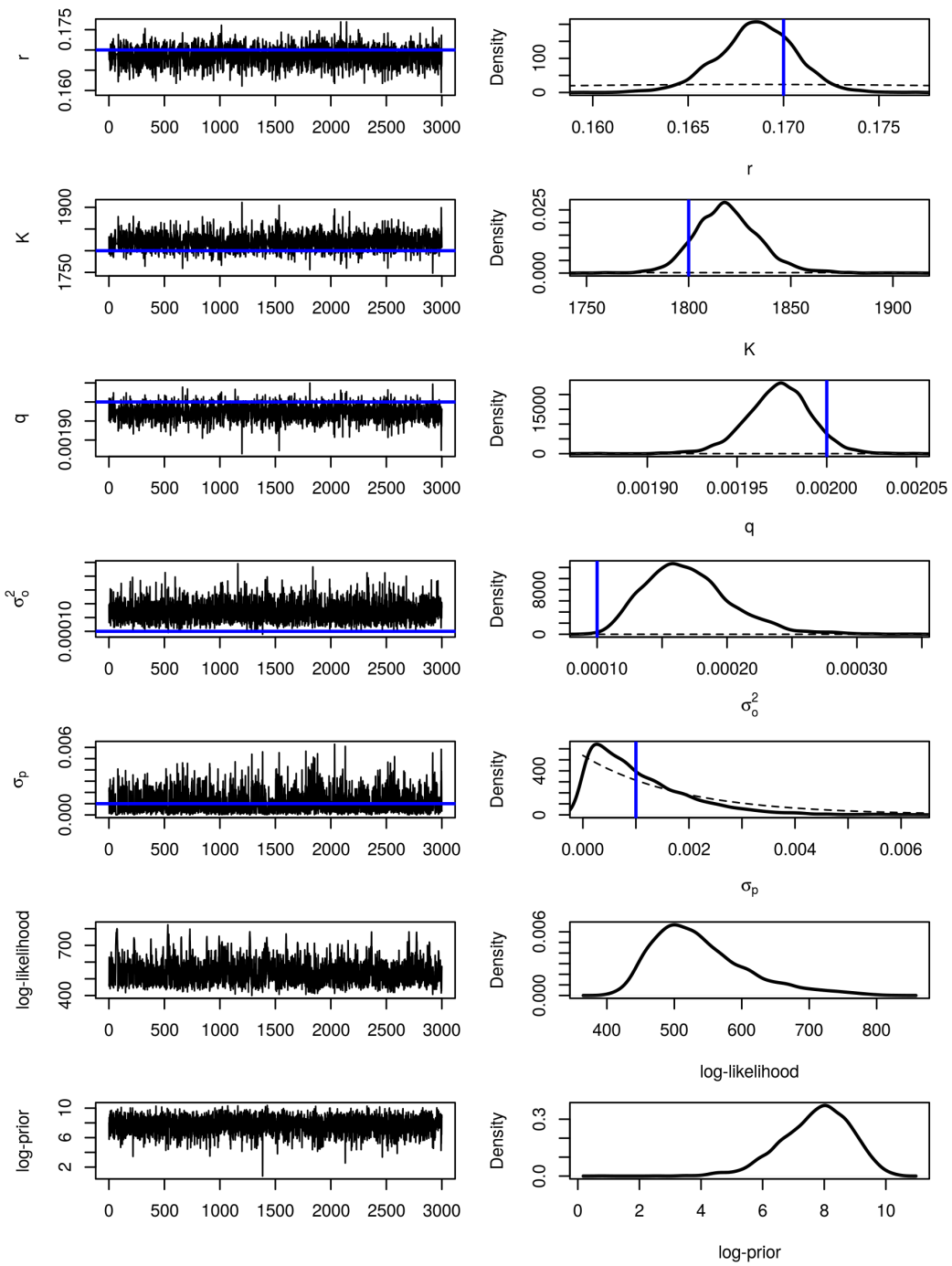


Figure 5.13: MCMC trace plots and posterior densities for the model parameters using data from the **low** observation error ($\sigma_o = 0.01$) and process error ($\sigma_p = 0.001$) model estimated using **uninformative priors** for observation error variance and process error standard deviation. Posterior traces and densities are indicated as black lines, priors as dashed black lines, and values specified in simulation as solid blue lines.

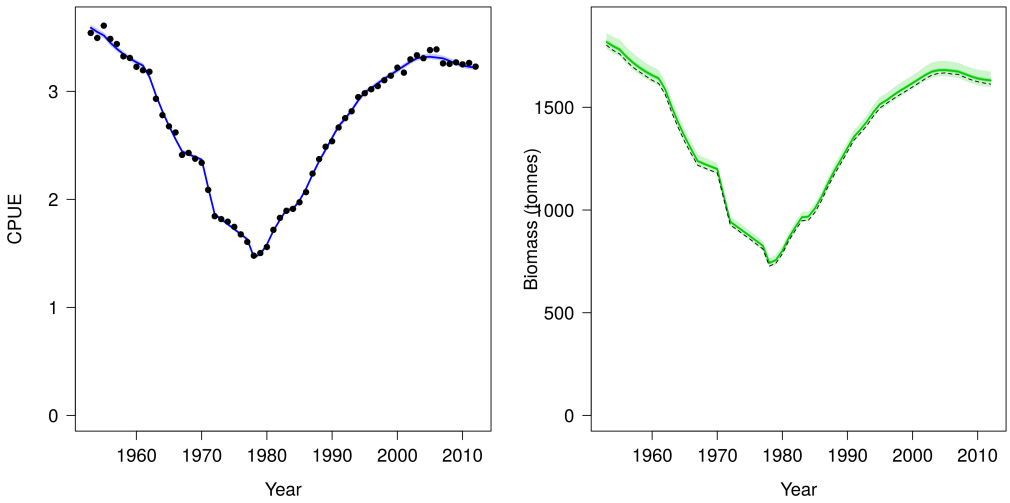


Figure 5.14: Fit to CPUE observations (I_t) [left] and the posterior distribution of biomass (B_t) [right] for the **low** observation error ($\sigma_p = 0.01$) and process error ($\sigma_p = 0.001$) model estimated using **uninformative priors** for observation error variance and process error standard deviation. CPUE observations are shown as black points [•] and the posterior distribution of the fit to CPUE is shown in blue. The posterior distribution of biomass is shown in green and the simulated biomass as the dashed black line. The shading indicates the 5th, 25th, 50th, 75th and 95th percentiles.

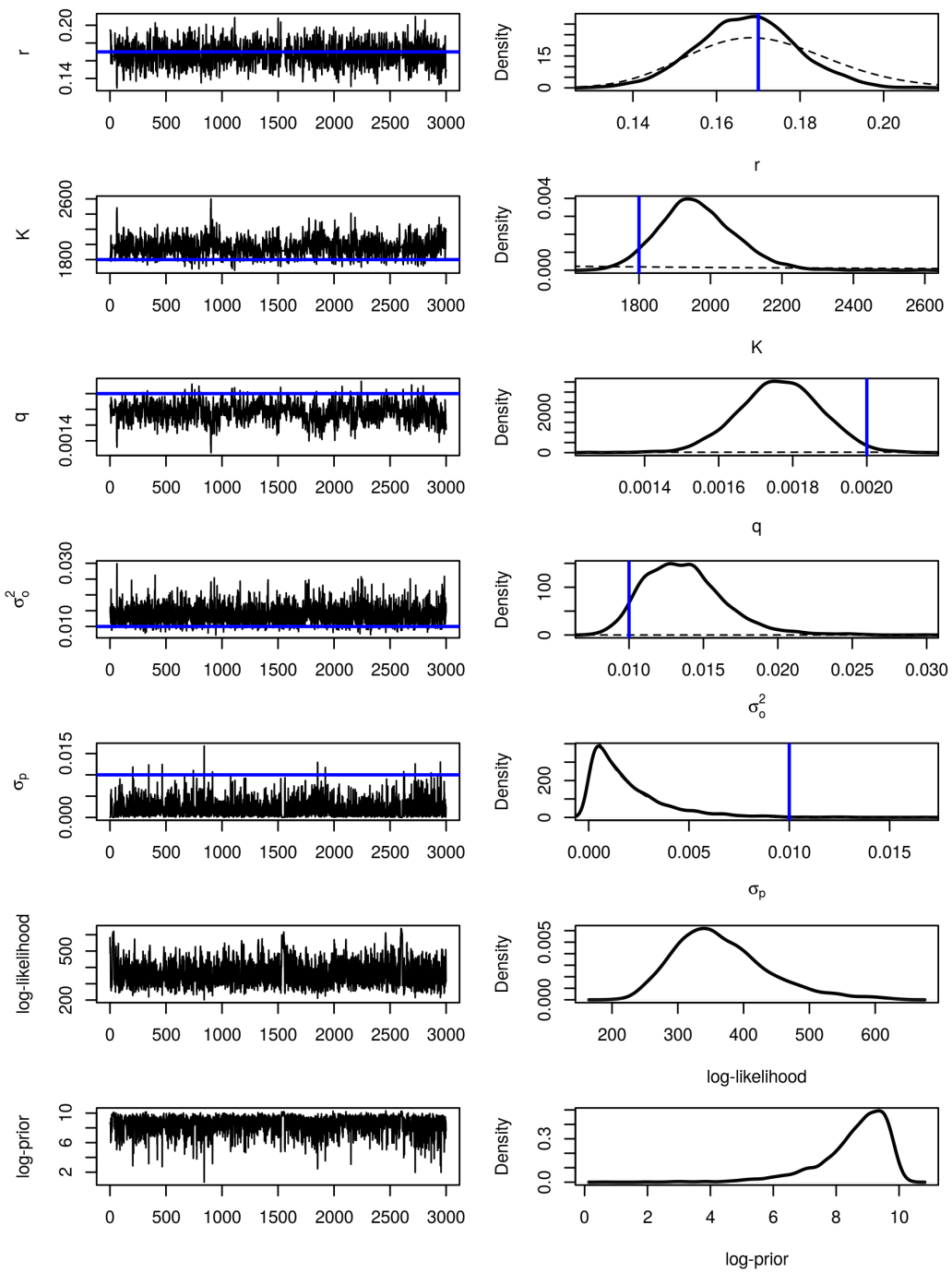


Figure 5.15: MCMC trace plots and posterior densities for the model parameters using data from the **high** observation error ($\sigma_o = 0.01$) and process error ($\sigma_p = 0.001$) model estimated using **uninformative priors** for observation error variance and process error standard deviation. Posterior traces and densities are indicated as black lines, priors as dashed black lines, and values specified in simulation as solid blue lines.

the data and MCMC is simply recovering the prior. The catchability coefficient (q) was underestimated and the carrying capacity (K) overestimated. While the model fit to the CPUE observations (I_t) looks adequate, the biomass (B_t) is consistently overestimated (Figure 5.16). The fit is no

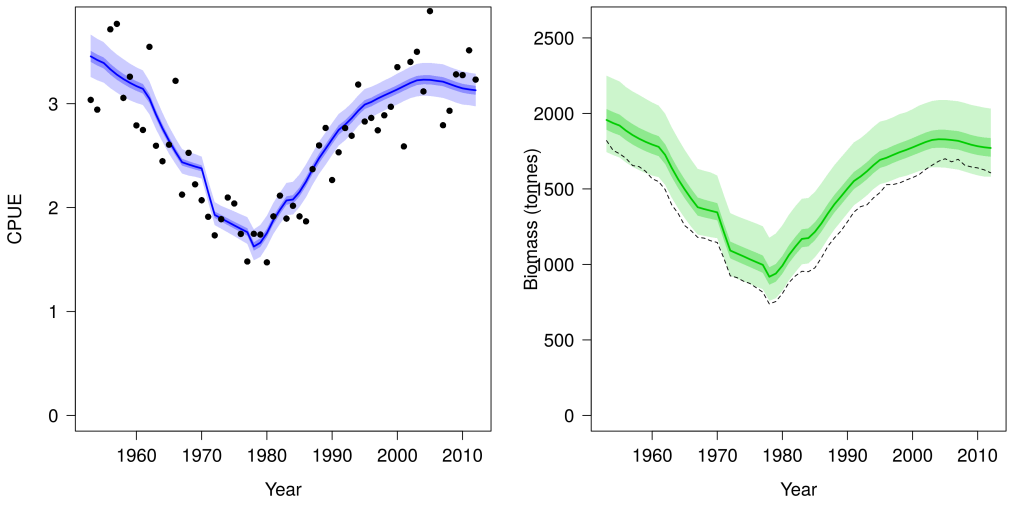


Figure 5.16: Fit to CPUE observations (I_t) [left] and the posterior distribution of biomass (B_t) [right] for the **high** observation error ($\sigma_p = 0.01$) and process error ($\sigma_p = 0.001$) model estimated using **uninformative priors** for observation error variance and process error standard deviation. CPUE observations are shown as black points [•] and the posterior distribution of the fit to CPUE is shown in blue. The posterior distribution of biomass is shown in green and the simulated biomass as the dashed black line. The shading indicates the 5th, 25th, 50th, 75th and 95th percentiles.

worse than in the previous section using informative priors for σ_o^2 and σ_p (Figure 5.12).

A naive final run

Finally, we do one last run fitting to the **high** observation and process error model ($\sigma_o = 0.1, \sigma_p = 0.01$) only. We specifying an inverse gamma prior

distribution with high variance for the process error standard deviation (σ_p) and a less informative prior on the intrinsic rate of population increase (r). The full list of priors is

$$\begin{aligned}
\pi(r) &\sim \log\mathcal{N}(\log(0.17), 0.2),\\
\pi(K) &\sim \log\mathcal{N}(6.90776, 0.989933),\\
\pi(q) &\sim \log\mathcal{N}(-3.45388, 1.4849),\\
\pi(\sigma_o^2) &\sim \mathcal{IG}(0.001, 0.001),\\
\pi(\sigma_p) &\sim \mathcal{IG}(0.001, 0.001).
\end{aligned} \tag{5.17}$$

Most parameters were not well recovered (Figure 5.17). The observation error variance (σ_o^2) was severely underestimated, while the standard deviation of process error (σ_p^2) was overestimated. The catchability coefficient (q) was underestimated, as was the intrinsic rate of population increase (r). Due to the extremely low estimates of observation error, the model fits the CPUE observations (I_t) too well, resulting in unrealistic biomass estimates (B_t, Figure 5.18).

This final model illustrates some of the difficulties faced when trying to tease apart observation and process error in the face of uncertainty. Estimating parameters relating to observation error (e.g. σ_o^2) and process error (e.g. σ_p^2) simultaneously is difficult due to the anti-correlation of these (and other) parameters (Figure 5.19).

To summarise, estimating parameters based on simulated populations with low observation and process errors is relatively easy. However, when faced with uncertainty, the choice of prior distributions becomes important. Of particular importance is the choice of prior controlling the magnitude of process error in these types of models. When tight, highly informative, priors are used, the MCMC method performs well and parameter estimation is straightforward. However, it is unrealistic to expect this degree of prior knowledge in practice. A further issue is the severe confounding between the models key parameters r, K and q (Hilborn & Walters 1992). However, this is largely overcome with the transformation suggested on page 5.3.1. The key problem with these models is their inability to separate the variance into the observation (σ_o^2) and process (σ_p^2) error components.

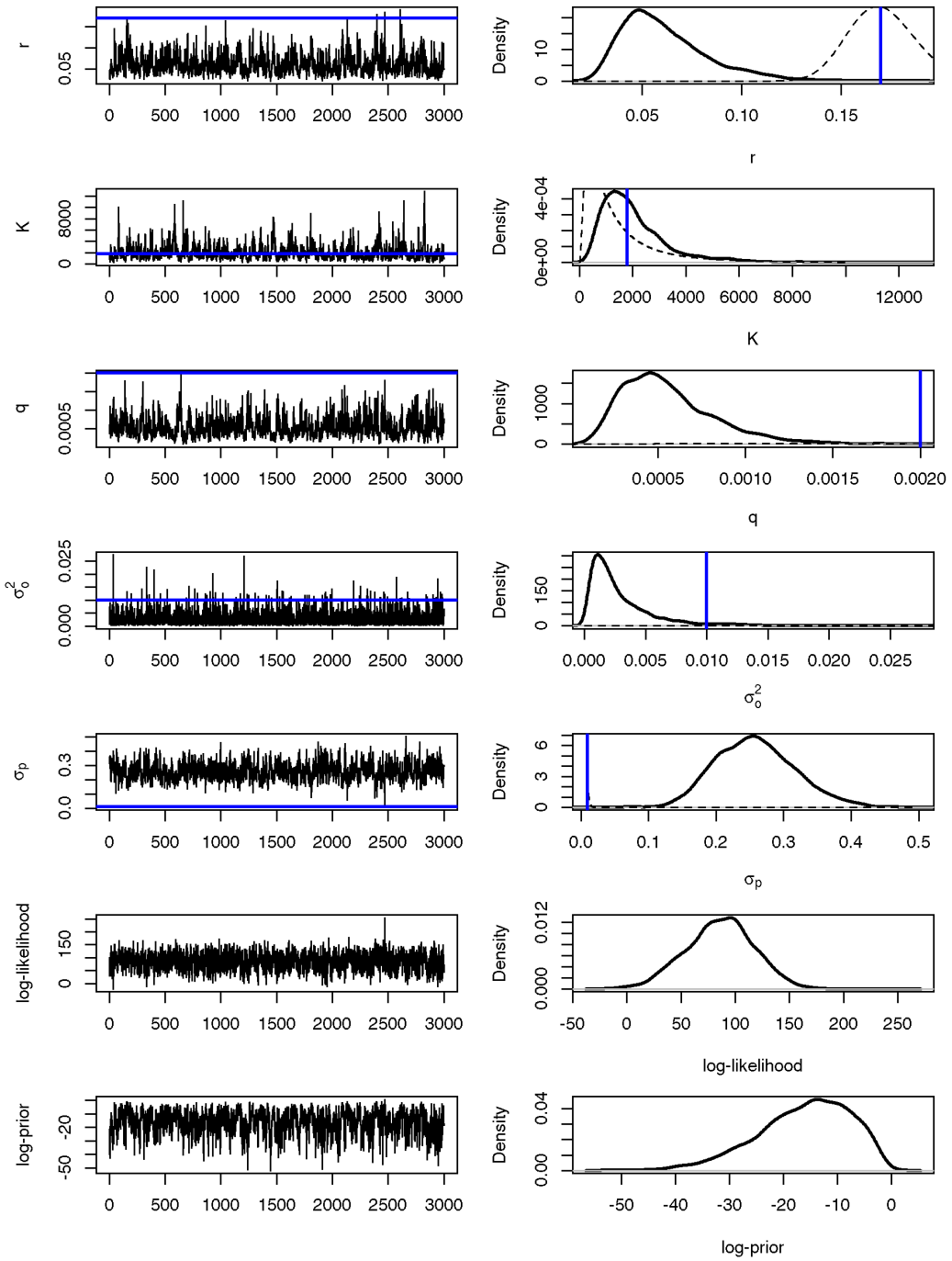


Figure 5.17: MCMC trace plots and posterior densities for the model parameters using data from the **high** observation error ($\sigma_o = 0.01$) and process error ($\sigma_p = 0.001$) model. Posterior traces and densities are indicated as black lines, priors as dashed black lines, and values specified in simulation as solid blue lines.

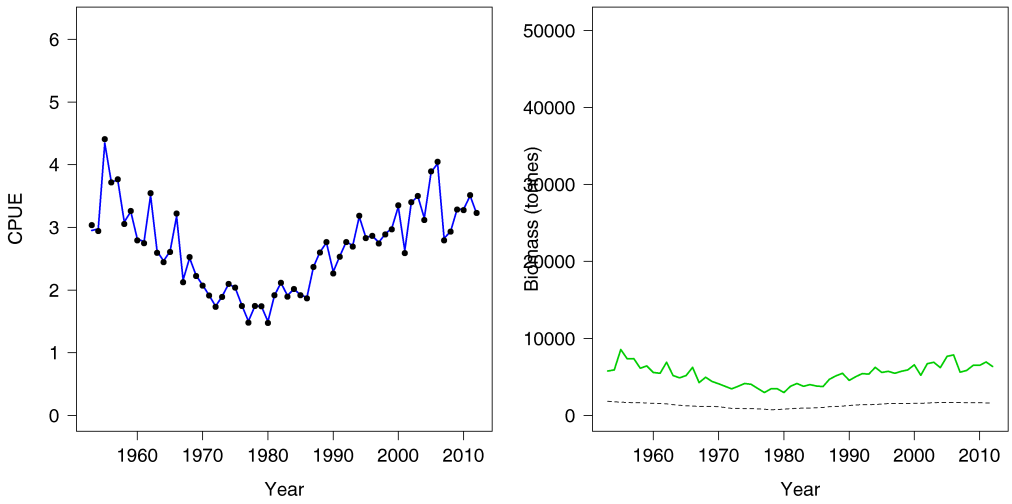


Figure 5.18: Fit to CPUE observations (I_t) [left] and the posterior distribution of biomass (B_t) [right] for the **high** observation error ($\sigma_p = 0.01$) and process error ($\sigma_p = 0.001$) model estimated using **highly uninformative priors**. CPUE observations are shown as black points [•] and the posterior distribution of the fit to CPUE is shown in blue. The posterior distribution of biomass is shown in green and the simulated biomass as the dashed black line. The shading indicates the 5th, 25th, 50th, 75th and 95th percentiles.

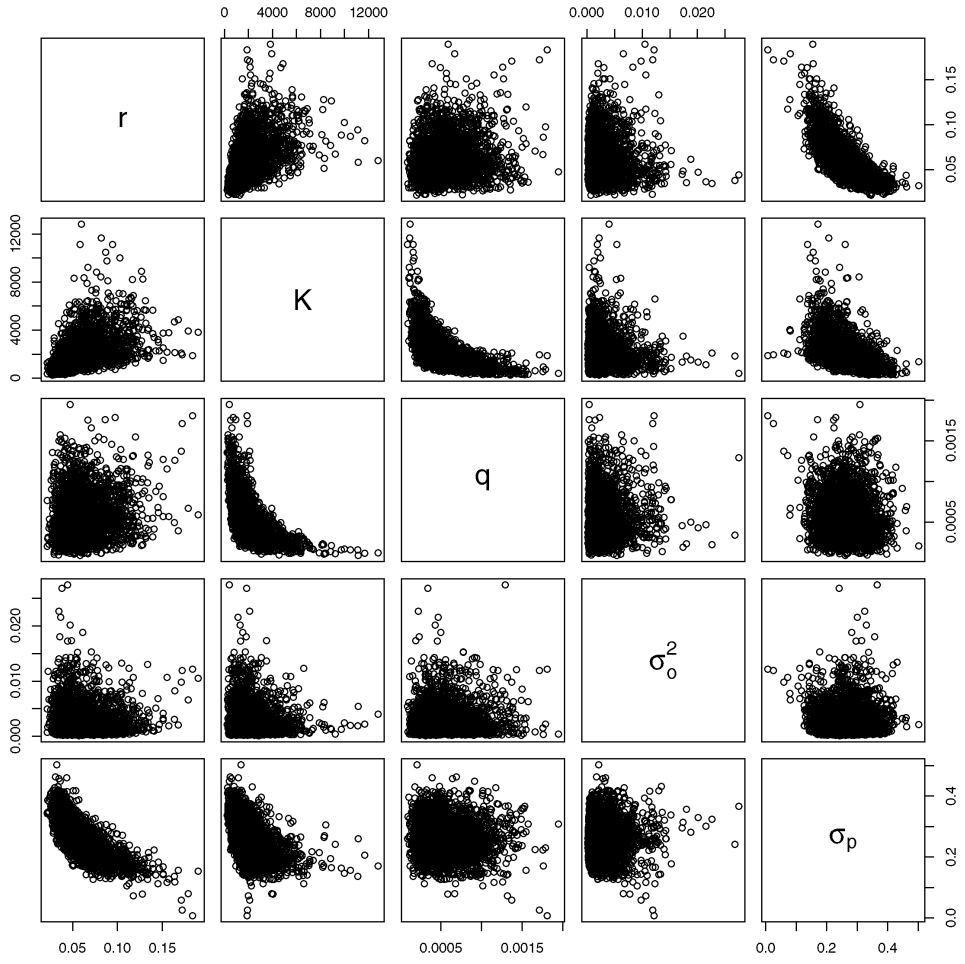


Figure 5.19: MCMC correlation plot for the **high** observation error ($\sigma_p = 0.01$) and process error ($\sigma_p = 0.001$) model estimated using **highly uninformative priors**.

This problem arises largely due to the anti-correlation between the two parameters (i.e. an increase in σ_o^2 can be compensated for by a decrease in σ_p^2 during MCMC). This is a well known problem in these types of models (see chapter 7 of Hilborn & Mangel 1997) and it is usually recommended that we must specify the variance of either the observation error or the process error, or the ratio of the variances (Schnute 1987). It has also been suggested that alternative methods of inference to MCMC (i.e. sample importance algorithms) can sometimes perform better here, but still do not solve the problem entirely (pers. comm. M. McAllister). However, this is beyond the scope of this thesis as sample importance methods are slow when sampling high dimensional models.

5.4 Age-structured state-space models

State-space features are commonly included in age-structured stock assessment models via the year class strengths (YCS_t). However, the state-space concept is rarely (if at all) applied across all ages and all years within age-structured models. The worry might be that trying to estimate all of the model parameters within an age-structured model along with a latent state for every single age in every single year might be intractable or take a very long time to converge. Here we attempt to make this problem more tractable by splitting the likelihood up into components and using a blockwise Metropolis-Hasting MCMC algorithm to try speed up MCMC convergence.

First we define the stock's average recruitment (R_0), the stock recruitment function ($SR\left(SSB_t\right)$), the spawning stock biomass (SSB_t, tonnes), the natural mortality at age a and time t ($M_{a,t}$), the mean weight (tonnes) of a fish of age a during time t ($w_{a,t}$), the time-varying selectivity of the fishery ($S_{a,t}$), the vulnerable biomass ($V_{a,t}$, tonnes), the observed catch ($C_{a,t}$, tonnes), and the exploitation rate ($U_{a,t}$). The numbers of fish at age and time ($N_{a,t}$) in a fully state-space age-structured model (referring back to Equation 1.26 in Chapter 1, page 42) from the beginning of the year could

evolve as follows

$$
\begin{aligned}
N'_{a,t} &= N_{a-1,t-1} && \text{ageing,}\\
N'_{1,t} &= R_0 \times SR\left(SSB_{t-1}\right) \times e^{\varepsilon_t^R - \sigma_R^2/2} && \text{recruitment,}\\
N''_{a,t} &= N'_{a,t} e^{-0.5M_{a,t} + \varepsilon_{a,t}^{p1} - \sigma_p^2/2} && \text{half natural mortality,}\\
SSB_t &= \sum_a N''_{a,t} w_{a,t} m_{a,t} e^{\varepsilon_{a,t}^w - \sigma_w^2/2 + \varepsilon_{a,t}^m - \sigma_m^2/2} && \text{spawning stock biomass,}\\
V_{a,t} &= N''_{a,t} w_{a,t} S_{a,t} e^{\varepsilon_{a,t}^w - \sigma_w^2/2 + \varepsilon_{a,t}^s - \sigma_s^2/2} && \text{vulnerable biomass,}\\
U_{a,t} &= \frac{C_{a,t} e^{\varepsilon_{a,t}^c - \sigma_c^2/2}}{V_{a,t}} && \text{fishing exploitation,}\\
I_{a,t} &= q V_{a,t} e^{\varepsilon_{a,t}^o - \sigma_o^2/2} && \text{catch per unit effort,}\\
\lambda_{a,t} &= N''_{a,t} S_{a,t}, &&\\
(Q_a)_t &= \frac{\lambda_{a,t}}{\sum_a \lambda_{a,t}} \quad \text{where } \sum_a (Q_a)_t = 1 \; \forall t, &&\\
(P_a)_t &\sim \mathcal{D}\text{irichlet}\left(\alpha_0 \alpha_{a,t} (Q_a)_t\right) \quad \text{where } \sum_a (P_a)_t = 1 \; \forall t && \text{proportions at age,}\\
N'''_{a,t} &= N''_{a,t} \left(1 - U_{a,t} S_{a,t} e^{\varepsilon_{a,t}^s - \sigma_s^2/2}\right) && \text{fishing,}\\
N_{a,t} &= N'''_{a,t} e^{-0.5M_{a,t} \varepsilon_{a,t}^{p2} - \sigma_p^2/2} && \text{half natural mortality,}
\end{aligned}
$$

where we have log-normal error (indicated above in red) associated with recruitment (ε_t^R), natural mortality at age when it is applied each time ($\varepsilon_{a,t}^{p1}$ and $\varepsilon_{a,t}^{p2}$), weight at age ($\varepsilon_{a,t}^w$), maturity at age ($\varepsilon_{a,t}^m$), selectivity at age ($\varepsilon_{a,t}^s$), the catch at age ($\varepsilon_{a,t}^c$), and the catch per unit effort ($\varepsilon_{a,t}^o$)

$$
\begin{aligned}
\varepsilon_t^R &\overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_R^2) & \varepsilon_{a,t}^{p1} &\overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_p^2),\\
\varepsilon_{a,t}^w &\overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_w^2) & \varepsilon_{a,t}^m &\overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_m^2),\\
\varepsilon_{a,t}^c &\overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_c^2) & \varepsilon_{a,t}^s &\overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_s^2),\\
\varepsilon_{a,t}^{p2} &\overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_p^2) & \varepsilon_{a,t}^o &\sim \mathcal{N}\left(0, \sigma_o^2\right).
\end{aligned}
$$

We use the notation $(P_a)_t$ to make it clear that we are talking about proportions in the catch at age composition P_a each year t.

> In fisheries science the scaled multinomial distribution is commonly used to fit age or length composition data (i.e. proportions in the catch

at age, proportions at age in the population, or proportions at length, sometimes by sex). The multinomial distribution is a generalisation of the binomial distribution and has parameters n (the integer number of trials $n > 0$) and $p_1, \ldots, p_k$ where $\sum_j^k p_j = 1$

$$\begin{aligned} y_j &\sim \mathcal{M}\text{ultinomial}(n, p_j)/n, \\ \mathbb{E}[y_j] &= p_j, \\ \mathbb{V}[y_j] &= \frac{p_j(1-p_j)}{n}, \end{aligned}$$

with discrete support $y_j \in \{0, \frac{1}{n}, \frac{2}{n}, \ldots, 1\}$ where $\sum y_j = 1$. In fisheries science, the parameter n is derived as the number of fish in the sample for which ages are measured, or the number of hauls in a year, or some combination of factors that relates to the amount of sampling done. Alternatively one might bootstrap re-sample length-frequency distributions and an age-length key to come up with some measure of the variance between proportions in the catch at age by year ($(P_a)_t$). These variances would then be used to derive an effective n (scaled by the relative variance in each year).

However, a similar modelling approach is possible using the Dirichlet distribution with the advantage that the distribution has continuous support. The Dirichlet distribution is the multivariate generalisation of the beta distribution. It has concentration parameters $\alpha_1, \ldots, \alpha_k$, where $\alpha_j > 0$ and continuous support $y_j \in [0, 1]$ and $\sum_{j=1}^k y_j = 1$

$$\begin{aligned} y_j &\sim \mathcal{D}\text{irichlet}(\alpha_0 \alpha_j), \\ \mathbb{E}[y_j] &= \alpha_j = p_j \quad \text{where } \sum_j \alpha_j = 1, \\ \mathbb{V}[y_j] &= \frac{\alpha_j(1-\alpha_j)}{(\alpha_0 + 1)}. \end{aligned}$$

Here the parameter α_0 is analogous to the n parameter in a multinomial distribution. Small values of α_0 will result in a "sloppy" (high variance) distribution, while a large α_0 will result in the expected value of y_j strongly concentrated towards p_j.

The Dirichlet distribution is a proper self-weighting distribution designed for continuous composition data. It has been shown that the multinomial distribution is in fact a bad choice for composition data, despite being the most commonly used distribution for composition data in fisheries science (Francis 2014). The multinomial distribution is not considered further in this thesis.

In practice, this model must be greatly simplified to make inference tractable. The reduced model that is commonly used for stock assessment in New Zealand is

$$
\begin{aligned}
N'_{a,t} &= N_{a-1,t-1}, \\
N'_{1,t} &= R_0 \times SR\left(SSB_{t-1}\right) \times e^{\varepsilon_t^R - \sigma_R^2/2} && \text{where } \varepsilon_t^R \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_R^2) \\
N''_{a,t} &= N'_{a,t} e^{-0.5M}, \\
SSB_t &= \sum_a N''_{a,t} w_a m_a, \\
V_t &= \sum_a N''_{a,t} w_a S_a, \\
U_t &= \frac{C_t}{V_t}, \\
I_t &= q V_t e^{\varepsilon_t^o - \sigma_o^2/2} && \text{where } \varepsilon_t^o \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_o^2), \\
\lambda_{a,t} &= N''_{a,t} S_a, \\
(Q_a)_t &= \frac{\lambda_{a,t}}{\sum_a \lambda_{a,t}} && \text{where } \sum_a (Q_a)_t = 1 \; \forall t, \\
(P_a)_t &\sim \mathcal{M}\text{ultinomial}\left(n, (Q_a)_t\right) && \text{where } \sum_a (P_a)_t = 1 \; \forall t, \\
N'''_{a,t} &= N''_{a,t}\left(1 - U_t S_a\right), \\
N_{a,t} &= N'''_{a,t} e^{-0.5M}.
\end{aligned}
$$

This model includes log-normal recruitment error (ε_t^R) and observation error (ε_t^o) only[1], ignoring any error associated with the natural mortality, weight at age, maturity at age, selectivity at age, or catch at age. This

[1]Actually, often the magnitude of observation error (a coefficient of variation) each year for the catch per unit effort time series is calculated outside of the model and included as a covariate. Sometimes some additional error is estimated as well.

model also assumes that natural mortality (M) is the same for all ages and years (rather than $M_{a,t}$, M_a or M_t), w_a rather than $w_{a,t}$, S_a rather than $S_{a,t}$ which results in the deconstruction of $U_{a,t}$ into $U_t S_a$ (see Chapter 1, page 40). Many of these assumptions are necessary if we are to have stable models with MCMC samplers that converge within our lifetime.

In this model $N''_{a,t}$ are the mid-year numbers at age. Alternatively, the numbers at age from mid-year to mid-year can be written

$$N_{a,t} = N_{a-1,t-1} e^{-0.5M} \left(1 - U_t S_a\right) e^{-0.5M}.$$

From now on we drop the $'$ notation and any reference to numbers at age is during the middle of the year unless specified otherwise.

Now, we describe a different model that includes process error in the mid-year numbers at age. Firstly, the weight at age (w_a), maturity at age (m_a) and selectivity at age (S_a) are all calculated as

$$w_a = w_\infty \left(1 - e^{-k(a-t_0)}\right)^\beta \quad \text{where } w_\infty = \alpha {L_\infty}^\beta, \tag{5.18}$$

$$m_a = 1 \Big/ \left(1 + 19^{(A_{50}-a)/A_{to95}}\right), \tag{5.19}$$

$$S_a = 1 \Big/ \left(1 + 19^{(\gamma_{50}-a)/\gamma_{95}}\right), \tag{5.20}$$

where fixed parameters are indicated in blue and estimated parameters in red. Equations 5.19 and 5.20 are logistic curves describing maturity and selectivity at age. See Sections 1.3.2, 1.3.3 and 1.3.9 in Chapter 1 for descriptions of each of these equations.

The deterministic equilibrium numbers at age in the population at the beginning of time $t = 0$ (i.e. before the first year of the model) are calculated as

$$N_a^0 = \begin{cases} R_0 e^{-(a-1)M} & \text{if } a = 1, \ldots, A \\ R_0 \frac{e^{M-AM}}{1-e^{-M}} & \text{if } a = A \end{cases},$$

referring back to Equation 1.19 on page 33. From this B_0 is calculated mid-year as

$$B_0 = \sum_a N_a^0 w_a m_a e^{-0.5M}.$$

The mid-year numbers at age at time $t = 1$ is calculated by removing half

of the total annual natural mortality (M) and adding process error ($\varepsilon^p_{a,t}$)

$$N_{a,t=1} = N^0_a e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} = \begin{cases} R_0 e^{-(a-0.5)M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = 1,\ldots,A \\ R_0 \frac{e^{M-AM}}{1-e^{-M}} e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = A \end{cases},$$

where

$$\varepsilon^p_{a,t} \overset{\text{iid}}{\sim} \mathcal{N}\left(0, \sigma^2_{a,t}\right),$$

noticing that the process error ($\varepsilon^p_{a,t}$) and process error variance ($\sigma^2_{a,t}$) are age and time specific (discussed below). Recruitment to the population each year (R_t) is defined as

$$R_t = \begin{cases} R_0 e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = 1, t = 1 \\ R_0 \times SR(SSB_{t-1}) \times e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = 1, t = 2,\ldots,T \end{cases},$$

noting that we continue to use process error terms ($\varepsilon^p_{a,t}$ and $\sigma^2_{a,t}$) rather than the standard recruitment variation (σ^2_R) and recruitment deviation (ε^R_t) or year class strength (YCS_t) notation, this is discussed further below. Finally, the numbers of fish transitioning between ages and years is

$$N_{a,t} = \begin{cases} N_{a-1,t-1} e^{-0.5M}(1 - U_{t-1}S_{a-1}) e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = 2,\ldots,A, t = 2,\ldots,T \\ N_{a-1,t-1} e^{-0.5M}(1 - U_{t-1}S_{a-1}) e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \\ \quad + N_{A,t-1} e^{-0.5M}(1 - U_{t-1}S_A) e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = A, t = 2,\ldots,T \end{cases},$$

where the bottom half of this equation describes the numbers at age in the plus group. Drawing the above definitions together, the mid-year numbers at age in the population ($N_{a,t}$) at all ages and all times can be summarised as

$$N_{a,t} = \begin{cases} R_0 e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = 1, t = 1 \\ R_0 e^{-(a-0.5)M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = 2,\ldots,A, t = 1 \\ R_0 \frac{e^{M-AM}}{1-e^{-M}} e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = A, t = 1 \\ R_0 \times SR(SSB_{t-1}) \times e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = 1, t = 2,\ldots,T \\ N_{a-1,t-1} e^{-0.5M}(1 - U_{t-1}S_{a-1}) e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = 2,\ldots,A, t = 2,\ldots,T \\ N_{a-1,t-1} e^{-0.5M}(1 - U_{t-1}S_{a-1}) e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \\ \quad + N_{A,t-1} e^{-0.5M}(1 - U_{t-1}S_A) e^{-0.5M+\varepsilon^p_{a,t}-\sigma^2_{a,t}/2} & \text{if } a = A, t = 2,\ldots,T \end{cases}. \quad (5.21)$$

Given the mid-year numbers at each age a and at each time t we can derive the following standard quantities

$$SSB_t = \sum_a N_{a,t} w_a m_a, \tag{5.22}$$

$$SR(SSB_t) = \frac{SSB_t}{B_0} \Big/ \left(1 - \frac{5h-1}{4h}\left(1 - \frac{SSB_t}{B_0}\right)\right), \tag{5.23}$$

$$V_t = \sum_a N_{a,t} w_a S_a, \tag{5.24}$$

$$U_t = \frac{C_t}{V_t}, \tag{5.25}$$

$$I_t = q V_t e^{\varepsilon_t^o - \sigma_o^2/2} \quad \text{where } \varepsilon_t^o \sim \mathcal{N}\left(0, \sigma_o^2\right), \tag{5.26}$$

$$\begin{aligned} \lambda_{a,t} &= N_{a,t} S_a, \\ (Q_a)_t &= \frac{\lambda_{a,t}}{\sum_a \lambda_{a,t}} \quad \sum_a (Q_a)_t = 1 \; \forall t, \\ (P_a)_t &\sim \mathcal{D}\text{irichlet}\left(\alpha_0 \alpha_t (Q_a)_t\right) \quad \sum_a (P_a)_t = 1 \; \forall t, \end{aligned} \tag{5.27}$$

where estimated parameters, latent states, and functions arising from estimated parameters are indicated in red. Fixed parameters, covariates, and functions arising from fixed parameters are indicated in blue. Observations are indicated in green. The CPUE (I_t) is assumed to be log-normally distributed and the proportions in the catch at age ($(P_a)_t$) are assumed to be Dirichlet distributed. Here, the Dirichlet distribution uses an annual scaling covariate (α_t) that is used to define the relative weight of proportions in the catch at age ($(P_a)_t$) between years. The values of these covariates are assumed known and could be estimated outside of the stock assessment model, much like an effective n is estimated for the multinomial distribution. The estimated parameter α_0 scales $\alpha_t (Q_a)_t$ and is analogous to data weighting within the model. For a thorough discussion on data weighting in stock assessment see Francis (2011). Also see the discussion at the end of this chapter (page 204).

The variance of the process error in this model is defined as an $A \times T$ matrix

for convenience, i.e. specifying a process error for each age and time

$$\left(\sigma^2_{a,t}\right) = \begin{pmatrix} \sigma^2_R & \sigma^2_R & \cdots & \sigma^2_R \\ \sigma^2_R & \sigma^2_p & \cdots & \sigma^2_p \\ \vdots & \vdots & \ddots & \vdots \\ \sigma^2_R & \sigma^2_p & \cdots & \sigma^2_p \end{pmatrix}_{A\times T} . \tag{5.28}$$

Here we have specified two process error variance parameters, variance associated with recruitment and numbers at age in the first year (σ^2_R), and the variance associated with all other transitions (σ^2_p, Figure 5.20). Con-

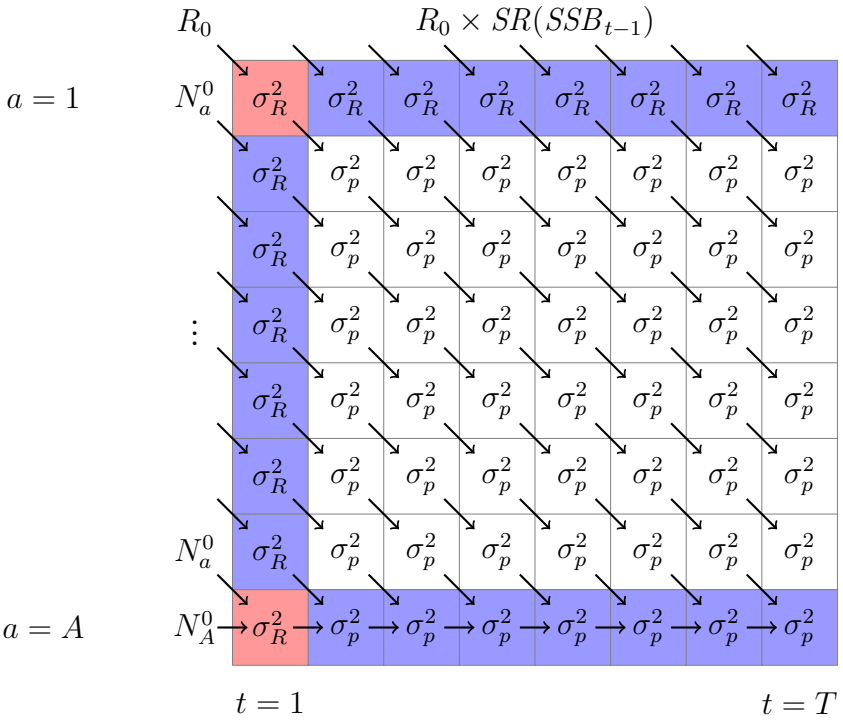


Figure 5.20: Matrix showing the direction of transitions between numbers of fish at age and time ($N_{a,t}$) within the model and the process errors variance to be used at each age a and time t. The numbers of fish for all ages and all times is defined by Equation 5.21. The colours are used to highlight the 6 parts of Equation 5.21.

sequently, the year class strengths (YCS_t) for each year can be calculated

as

$$YCS_t = e^{\varepsilon^p_{a=1,t} - \sigma^2_{a=1,t}/2} = e^{\varepsilon^p_{a=1,t} - \sigma^2_R/2} \quad \forall t. \tag{5.29}$$

In summary, this model makes the following assumptions:

- the natural mortality rate (M), the maturity parameters (A_{50} and A_{to95}), the von Bertalanffy growth parameters (L_∞, k and t_0) and the length-weight parameters (α and β) are all known parameters ($\boldsymbol{\omega}$)
- natural mortality (M) is constant over age a and time t ($M_{a,t} = M$)
- mean weight is deterministic (i.e. without error) and constant over time ($w_{a,t} = w_a$)
- selectivity is deterministic (i.e. without error) and constant over time ($S_{a,t} = S_a$)
- catch (C_t) is measured without error
- the exploitation rate (U_t) is constant by age ($U_{a,t} = U_t$)
- CPUE (I_t) is proportional to abundance.
- the population is closed (no immigration or emigration)
- the numbers at age at the start of the model are log-normally distributed about N^0_a with variance σ^2_R (Equation 5.21)[2].

5.4.1 Inference

We are interested in attaining a probabilistic relationship between the following:

- **The data y:** the catch per unit effort (I_t) and proportions in the catch at age ($P_{a,t}$). Let $\mathbf{y} = \{\{I_t\}^T_{t=1}, \{\{P_{a,t}\}^A_{a=1}\}^T_{t=1}\}$

[2]Here σ^2_p could be used rather than σ^2_R. This would assume that the age composition at the beginning of the model is more similar to the equilibrium numbers at age. Composition data that enables us to estimate these year classes are rarely available in the early years of stock assessment models so this mat be a better choice in many cases. Alternatively, an entirely different variance parameter (e.g. σ^2_N) could be estimated.

- **The covariates z:** the catch (C_t) and the relative variances (weights) of the Dirichlet distribution for the proportions in the catch at age data (α_t). Let $\mathbf{z} = \{C_t, \alpha_t\}_{t=1}^{T}$

- **The unknown parameters of interest $\boldsymbol{\theta}$:** the virgin recruitment (R_0) and the selectivity parameters (γ_{50} and γ_{95}). Let $\boldsymbol{\theta} = \{R_0, \gamma_{50}, \gamma_{95}\}$

- **The unknown nuisance parameters $\boldsymbol{\phi}$:** the catchability coefficient (q), the observation error variance (σ_o^2), the process error variance (σ_p^2), the recruitment deviation variance (σ_R^2), and the scaling of the Dirichlet distribution variances (α_0). Let $\boldsymbol{\phi} = \{q, \sigma_o^2, \sigma_p^2, \sigma_R^2, \alpha_0\}$

- **The known parameters $\boldsymbol{\omega}$:** the natural mortality rate (M), the maturity parameters (A_{50} and A_{to95}), the von Bertalanffy growth parameters (L_∞, k and t_0) and the length-weight parameters (α and β). Let $\boldsymbol{\omega} = \{M, h, A_{50}, A_{to95}, L_\infty, k, t_0, \alpha, \beta\}$

- **The unknown latent states x:** the numbers at age during the middle of the year ($N_{a,t}$, i.e. after half of the natural mortality has been applied, but before fishing). Let $\mathbf{x} = \{N_{a,t}\}_{a=1,t=1}^{A,T}$

Using Bayes theorem, the posterior distribution of the model parameters ($\boldsymbol{\theta}$ and $\boldsymbol{\phi}$) and states (x), given the fixed parameters ($\boldsymbol{\omega}$), data (y) and covariates (z) is

$$\pi(\boldsymbol{\theta}, \boldsymbol{\phi}, \mathbf{x}|\boldsymbol{\omega}, \mathbf{y}, \mathbf{z}) \propto \pi(\boldsymbol{\theta}, \boldsymbol{\phi}, \mathbf{x}|\boldsymbol{\omega}, \mathbf{z})\pi(\mathbf{y}|\boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\omega}, \mathbf{x}, \mathbf{z}), \tag{5.30}$$

where the prior is

$$\begin{aligned}
\pi(\boldsymbol{\theta}, \boldsymbol{\phi}, \mathbf{x}|\boldsymbol{\omega}, \mathbf{z}) &= \pi(R_0, \gamma_{50}, \gamma_{95}, q, \sigma_o^2, \sigma_p^2, \sigma_R^2, \alpha_0, \mathbf{x}|\boldsymbol{\omega}, \mathbf{z}) \\
&= \pi(R_0)\pi(\gamma_{50})\pi(\gamma_{95})\pi(q)\pi(\sigma_o^2)\pi(\sigma_p^2)\pi(\sigma_R^2)\pi(\alpha_0)\pi(\mathbf{x}|\boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\omega}, \mathbf{z}) \\
&= \pi(R_0)\pi(\gamma_{50})\pi(\gamma_{95})\pi(q)\pi(\sigma_o^2)\pi(\sigma_p^2)\pi(\sigma_R^2)\pi(\alpha_0) \\
&\quad \times \prod_{t=1}^{T} \pi\left(N_{a=1,t}|R_0, SSB_{t-1}, \sigma_R^2, \boldsymbol{\omega}\right) \\
&\quad \times \prod_{a=1}^{A} \pi\left(N_{a,t=1}|R_0, \sigma_R^2, \boldsymbol{\omega}\right) \\
&\quad \times \prod_{a=2}^{A-1} \prod_{t=2}^{T} \pi\left(N_{a,t}|N_{a-1,t-1}, V_{t-1}, C_{t-1}, \gamma_{50}, \gamma_{95}, \sigma_p^2, \boldsymbol{\omega}\right), \\
&\quad \times \prod_{t=2}^{T} \pi\left(N_{a=A,t}|N_{A-1,t-1}, N_{A,t-1}, V_{t-1}, C_{t-1}, \gamma_{50}, \gamma_{95}, \sigma_p^2, \boldsymbol{\omega}\right),
\end{aligned}$$

and the likelihood is

$$\begin{aligned}
\pi\left(\mathbf{y}|\boldsymbol{\theta}, \boldsymbol{\phi}, \boldsymbol{\omega}, \mathbf{x}, \mathbf{z}\right) &= \pi\left(\mathbf{I}, \mathbf{P}|\gamma_{50}, \gamma_{95}, q, \sigma_o^2, \alpha_0, \boldsymbol{\omega}, \mathbf{x}, \mathbf{z}\right) \\
&= \pi(\mathbf{I}|\gamma_{50}, \gamma_{95}, q, \sigma_o^2, \mathbf{x})\pi(\mathbf{P}|\gamma_{50}, \gamma_{95}, \alpha_0, \mathbf{x}, \mathbf{z}, \boldsymbol{\omega}) \\
&= \prod_{t=1}^{T} \pi(I_t|N_{a,t}, \gamma_{50}, \gamma_{95}, q, \sigma_o^2, \boldsymbol{\omega}) \\
&\quad \times \prod_{a=1}^{A} \prod_{t=1}^{T} \pi\left((P_a)_t|N_{a,t}, C_t, \alpha_t, \gamma_{50}, \gamma_{95}, \alpha_0, \boldsymbol{\omega}\right).
\end{aligned}$$

The likelihood of this model is made up of three main components: the likelihood of the CPUE observations (I_t), the likelihood of the proportions in the catch at age observations ($(P_a)_t$), and the likelihood of the numbers at age latent states ($N_{a,t}$). The likelihood of the CPUE observations is

$$\log\left(I_t\right)|N_{a,t}, \gamma_{50}, \gamma_{95}, q, \sigma_o^2, \boldsymbol{\omega}, \sim \mathcal{N}\left(\log\left(qV_t\right) - \sigma_o^2/2, \sigma_o^2\right). \tag{5.31}$$

The likelihood of the proportions in the catch at age is

$$(P_a)_t|N_{a,t}, C_t, \alpha_t, \gamma_{50}, \gamma_{95}, \alpha_0, \boldsymbol{\omega} \sim \mathcal{D}\text{irichlet}\left(\alpha_0\alpha_t(Q_a)_t\right), \tag{5.32}$$

where α_0 is a scaling parameter defining the variability in the Dirichlet distribution and α_t is the annual variation. The likelihood of the numbers

at age is

$$\log\left(N_{a,t}\right)|\mu_{a,t}, C_t, \gamma_{50}, \gamma_{95}, \sigma^2_{a,t}, \boldsymbol{\omega} \sim \mathcal{N}\left(\log\left(\mu_{a,t}\right) - \sigma^2_{a,t}/2, \sigma^2_{a,t}\right), \tag{5.33}$$

where

$$\mu_{a,t} = \begin{cases} R_0 e^{-0.5M} & \text{if } a = 1, t = 1 \\ R_0 e^{-(a-0.5)M} & \text{if } a = 2, \ldots, A, t = 1 \\ R_0 \frac{e^{M-AM}}{1-e^{-M}} e^{-0.5M} & \text{if } a = A, t = 1 \\ R_0 \times SR(SSB_{t-1}) \times e^{-0.5M} & \text{if } a = 1, t = 2, \ldots, T \\ N_{a-1,t-1} e^{-0.5M}(1 - U_{t-1} S_{a-1}) e^{-0.5M} & \text{if } a = 2, \ldots, A, t = 2, \ldots, T \\ N_{a-1,t-1} e^{-0.5M}(1 - U_{t-1} S_{a-1}) e^{-0.5M} & \\ \quad + N_{A,t-1} e^{-0.5M}(1 - U_{t-1} S_A) e^{-0.5M} & \text{if } a = A, t = 2, \ldots, T \end{cases}, \tag{5.34}$$

noting the similarity with Equation 5.21. Using MCMC we can sample from the posterior distribution (Equation 5.30) to obtain probability distributions of the parameters, latent states and other quantities of interest in the model. We provide an example below (Section 5.4.2, page 179). Using Equation 5.30 and the priors described below we estimate posterior distributions for each of the parameters using blockwise Metropolis-Hastings with log-normal proposals (see Chapter 2, page 66).

The three main components of the likelihood described above can be split into even smaller subcomponents made up of just single ages a and years t. When a parameter or latent state is proposed within MCMC, then only the probability density function (PDF) of those subcomponents relevant to the proposal, and the prior, need be evaluated. Each of the MCMC proposals and the subcomponents of the likelihood for which the PDF needs to be evaluated, excluding the priors (these are specified in the preceding sections), are as follows:

- **When proposing** R_0^* update B_0 and $SR(SSB_t)$ then calculate the PDF of

$$\begin{aligned} &\log\left(N_{a=1,t}\right) \sim \mathcal{N}\left(\log\left(\mu_{a=1,t}\right), \sigma^2_{a=1,t}\right) \quad \forall t, \\ &\log\left(N_{a,t=1}\right) \sim \mathcal{N}\left(\log\left(\mu_{a,t=1}\right), \sigma^2_{a,t=1}\right) \quad a = 2, \ldots, A. \end{aligned}$$

i.e. at age 1 and at time 1.

- **When proposing** q^* calculate the PDF of

$$\log(I_t) \sim \mathcal{N}\left(\log(q^* V_t), \sigma_o^2\right) \quad \forall t.$$

- **When proposing** γ_{50}^* **or** γ_{95}^* update S_a, V_t, U_t, $(Q_a)_t$ then calculate the PDF of

$$\begin{aligned}
\log(I_t) &\sim \mathcal{N}\left(\log(qV_t), \sigma_o^2\right) \quad \forall t, \\
(P_a)_t &\sim \mathcal{D}\text{irichlet}\left(\alpha_0 \alpha_t (Q_a)_t\right) \quad \forall t, \\
\log(N_{a,t}) &\sim \mathcal{N}\left(\log(\mu_{a,t}), \sigma_{a,t}^2\right) \quad a = 2, \ldots, A, t = 2, \ldots, T.
\end{aligned}$$

- **When proposing** σ_o^{2*} calculate the PDF of

$$\log(I_t) \sim \mathcal{N}\left(\log(qV_t), \sigma_o^{2*}\right) \quad \forall t.$$

- **When proposing** σ_p^{2*} calculate the PDF of

$$\log(N_{a,t}) \sim \mathcal{N}\left(\log(\mu_{a,t}), \sigma_p^{2*}\right) \quad a = 2, \ldots, A, \forall t.$$

- **When proposing** σ_R^{2*} calculate the PDF of

$$\log(N_{a=1,t}) \sim \mathcal{N}\left(\log(\mu_{a=1,t}), \sigma_R^{2*}\right) \quad \forall t.$$

- **When proposing** α_0^* calculate the PDF of

$$(P_a)_t \sim \mathcal{D}\text{irichlet}\left(\alpha_0^* \alpha_t (Q_a)_t\right) \quad \forall a \forall t,$$

- **When proposing** $N_{a,t}^*$ update SSB_t, $SR(SSB_t)$, V_t, U_t, $(Q_a)_t$ then calculate the PDF of

$$\begin{aligned}
\log(I_t) &\sim \mathcal{N}\left(\log(qV_t), \sigma_o^2\right), \\
(P_a)_t &\sim \mathcal{D}\text{irichlet}\left(\alpha_0 \alpha_t (Q_a)_t\right), \\
\log\left(N_{a,t}^*\right) &\sim \mathcal{N}\left(\log(\mu_{a,t}), \sigma_{a,t}^2\right) \\
\log(N_{a,t+1}) &\sim \mathcal{N}\left(\log(\mu_{a,t+1}), \sigma_{a,t}^2\right) \quad \forall a.
\end{aligned}$$

See Figure 5.21 for a graphic example of the subcomponents of the likelihood for which PDFs need to be evaluated when proposing $N_{a,t}^*$ for a single a and t.

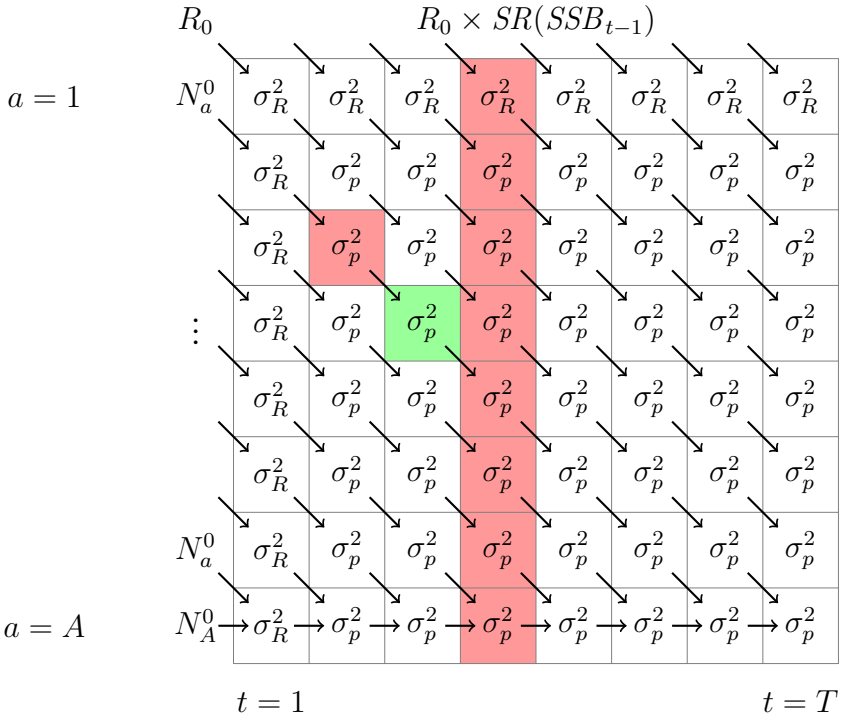


Figure 5.21: If updating the green cell ($N_{a,t}$), then the likelihood of that cell given the previous numbers at age and time ($N_{a-1,t-1}$, shown in red) and the numbers at all ages in the following time step ($N_{a,t+1}\ \forall a$, shown in red) need to be evaluated. The probability of the numbers at age for time $t+1$ need to be evaluated as they depend on variables derived from the numbers in the previous time step t, namely the spawning stock biomass (SSB_t), the stock recruitment ($SR(SSB_t)$), the vulnerable biomass (V_t) and the exploitation rate (U_t).

One of the greatest challenges in MCMC is achieving good mixing of the chains. When simultaneously sampling a large number of parameters (i.e. multivariate proposals), the algorithm might find it difficult to achieve good mixing as multiple parameter updates can be difficult to construct and tune. On the other hand, single parameter updates can result in slow mixing, if the proposal variance is low, many proposals are accepted, but mixing is slow. If the proposal variance is high, many proposals are rejected. In practice it is often a good idea to form small groups of correlated parameters that belong to the same context in the formulation of the model. The best mixing is usually obtained with a blocking strategy somewhere between the all-at-once and one-at-a-time strategies.

Initial exploration suggested that mixing of the numbers at age latent states ($N_{a,t}$) was very slow. This is likely to be due to the high correlation between numbers at age and time within cohorts (i.e. $N_{a,t}$ and $N_{a+1,t+1}$ are highly correlated). Rather than using a multivariate proposal distribution, we developed a proposal that updates a diagonal block of numbers at age (i.e. a cohort) by scaling the entire cohort up or down. This diagonal cohort update is achieved by drawing a scaling parameter (λ^*) from a log-normal distribution

$$\lambda^* \sim \log \mathcal{N}\left(0, \sigma_q^2\right). \tag{5.35}$$

We define each cohort as a vector $\boldsymbol{\eta}_i$ where there are $i = 1, \ldots, k$ cohorts in the model and $\boldsymbol{\eta}_i = \{\eta_j\}_{j=1}^p$ and we have $j = 1, \ldots, p$ age classes within a cohort. We then scale a cohort i using

$$\eta_{i,j}^* = \eta_{i,j}\lambda_i^*. \tag{5.36}$$

Now we derive the proposal ratio for this move, we write

$$\begin{aligned} \eta_{i,j}^* &= \eta_{i,j}\lambda_i^*\upsilon_j^* \quad \text{forwards move,} \\ \eta_{i,j} &= \eta_{i,j}^*\lambda_i\upsilon_j \quad \text{reverse move,} \end{aligned}$$

and show that

$$\begin{aligned}
\eta^*_{i,j} &= \eta^*_{i,j}\lambda_i \upsilon_j \lambda^*_i \upsilon^*_j \\
&= \eta^*_{i,j}\lambda_i \lambda^*_i \upsilon_j \upsilon^*_j \\
1 &= \lambda_i \lambda^*_i \upsilon_j \upsilon^*_j = \lambda_i \lambda^*_i \\
\lambda_i &= \frac{1}{\lambda^*_i}, \\
\log \lambda_i &= -\log \lambda^*_i,
\end{aligned}$$

where $\upsilon_j = \{\upsilon_j\}_{j=1}^p$ and

$$\begin{aligned}
\lambda^*_i &\sim \log \mathcal{N}\left(0, \sigma^2_\lambda\right), \\
\upsilon^*_j &\overset{\text{iid}}{\sim} \log \mathcal{N}\left(0, \sigma^2_\upsilon\right) \ \forall j, \\
\frac{\partial \eta^*_{i,j}}{\partial \upsilon^*_j} &= \eta_{i,j}\lambda^*_i.
\end{aligned}$$

The proposal density for a single cohort is

$$\begin{aligned}
q\left(\eta^*_{i,j}, \lambda^*_i | \eta_{i,j}, \lambda_i\right) &= q\left(\eta^*_{i,j} | \lambda^*_i, \eta_{i,j}, \lambda_i\right) q\left(\lambda^*_i | \eta_{i,j}, \lambda_i\right), \\
&= q\left(\upsilon^*_j | \lambda^*_i, \eta_{i,j}, \lambda_i\right) \left|\frac{\partial \upsilon^*_j}{\partial \eta^*_{i,j}}\right| q\left(\lambda^*_i | \eta_{i,j}, \lambda_i\right), \\
&= \left[\prod_j^p \frac{1}{\upsilon^*_j} \left(2\pi\sigma^2_\upsilon\right)^{-\frac{1}{2}} e^{-\frac{1}{2\sigma^2_\upsilon}(\log \upsilon^*_j)^2} \frac{1}{\eta_{i,j}\lambda^*_i}\right] \left(2\pi\sigma^2_\lambda\right)^{-\frac{1}{2}} e^{-\frac{1}{2\sigma^2_\lambda}(\log \lambda^*_i)^2} \\
&= \left(2\pi\sigma^2_\upsilon\right)^{-\frac{1}{2}} \left(2\pi\sigma^2_\lambda\right)^{-\frac{1}{2}} \left[\prod_j^p \frac{1}{\eta_{i,j}\lambda^*_i e^{-\frac{1}{2\sigma^2_\upsilon}(\log \upsilon^*_j)^2}}\right] \frac{1}{\lambda^*_i} e^{-\frac{1}{2\sigma^2_\lambda}(\log \lambda^*_i)^2},
\end{aligned}$$

therefore the proposal ratio is

$$\begin{aligned}
\frac{q\left(\eta_{i,j},\lambda_i|\eta^*_{i,j},\lambda^*_i\right)}{q\left(\eta^*_{i,j},\lambda^*_i|\eta_{i,j},\lambda_i\right)} &= \frac{\prod_j^p \frac{1}{\eta^*_{i,j}\lambda_i} e^{-\frac{1}{2\sigma^2_\upsilon}(\log \upsilon_j)^2}}{\prod_j^p \frac{1}{\eta_{i,j}\lambda^*_i} e^{-\frac{1}{2\sigma^2_\upsilon}(\log \upsilon^*_j)^2}} \times \frac{\frac{1}{\lambda_i} e^{-\frac{1}{2\sigma^2_\lambda}(\log \lambda_i)^2}}{\frac{1}{\lambda^*_i} e^{-\frac{1}{2\sigma^2_\lambda}(\log \lambda^*_i)^2}} \\
&= \prod_j^p \frac{\eta_{i,j}\lambda^*_i}{\eta^*_{i,j}\lambda_i} e^{-\frac{1}{2\sigma^2_\upsilon}\left((\log \upsilon_j)^2-(\log \upsilon^*_j)^2\right)} \times \frac{\lambda^*_i}{\lambda_i} \times e^{-\frac{1}{2\sigma^2_\lambda}\left((\log \lambda_i)^2-(\log \lambda^*_i)^2\right)} \\
&\quad \text{if } \sigma_\upsilon \leftarrow 0 \text{ then } \upsilon_j \leftarrow 1 \text{ therefore} \\
&= \prod_j^p \frac{\eta^*_{i,j}\lambda_i\lambda^*_i}{\eta^*_{i,j}\lambda_i} \left(\frac{\lambda^*_i}{\lambda_i}\right) \\
&= \prod_j^p \frac{\eta^*_{i,j}\lambda_i\lambda^*_i}{\eta^*_{i,j}\lambda_i} \left(\lambda^*_i\right)^2 \\
&= \left(\lambda^*_i\right)^{p+2}.
\end{aligned}$$

5.4.2 Snapper simulation example

In the following sections we present three examples in which we simulate data based loosely on the snapper (*Pagurus auratus*) fishery in northern New Zealand (see Chapter 3, page 77). For most parameters we use a set of plausible values in each example. In these examples the only parameter that is altered is the recruitment variance parameter in the third example (σ^2_R).

In all simulations the ages modelled are $a = 1, \ldots, A$ where A is a plus group at 20 years of age. The actual catch history of the snapper fishery (aggregated to a single area) is used in all simulation runs (Figure 5.22). We cut down the number of years modelled in this simulation to 100 (from 114, purely for ease of presenting outputs). In summary, this is a single area, single sex, single fishery, age-structured model with 20 age categories ($A = 20$) and 100 time-steps or years ($T = 100$).

In each example, we fit the fully state-space age-structured model using different priors for each of the models key parameters. We then attempt to

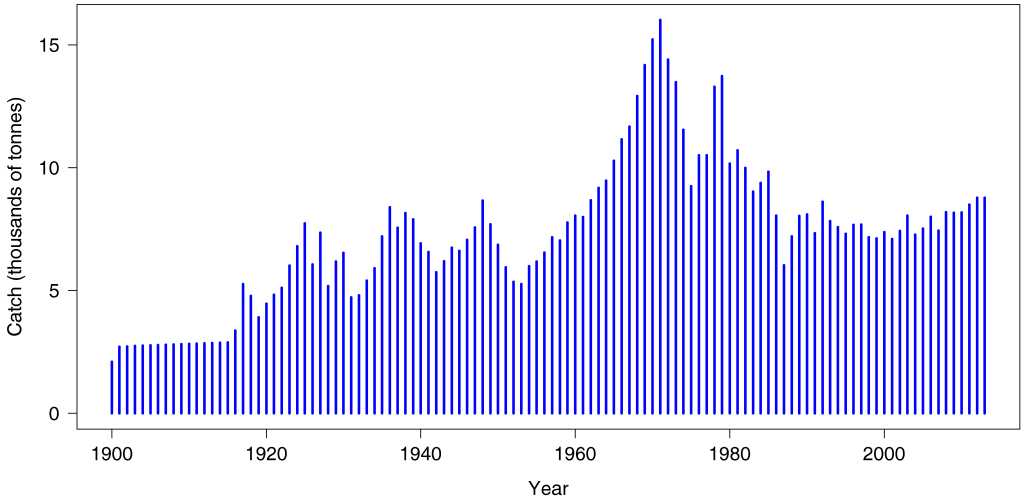


Figure 5.22: Actual catch history (tonnes) of the snapper fishery (SNA 1) from 1900 to 2013 (114 years) used in the simulation.

sample from the posterior distribution using MCMC. In the first example, all of the key model parameters are fixed and only the numbers at age and time latent states ($N_{a,t}$) are estimated. In the remaining two examples, all key parameters but the process error variance (σ_p^2) are estimated. Additional MCMC diagnostic plots for each of the examples presented below are provided in Appendix B.

Model validation

The first example simulates data using the model parameters shown in Table 5.2. Notice that the recruitment variance (σ_R^2) parameter is set at a very low value. We come back to a discussion of this important and highly influential parameter in Section 5.4.2 (page 201) below and in the discussion at the end of this chapter (page 204).

We began by fixing all of the key model parameters to their true values during estimation (i.e. the values specified in Table 5.2) and ran the MCMC to obtain posterior distributions of the numbers at age latent states only. This example was done to ensure there were no problems in the

Table 5.2: Parameter values used in age-structured simulation. The parameters are grouped into estimated parameters of interest ($\boldsymbol{\theta}$), nuisance parameters ($\boldsymbol{\phi}$) and fixed parameters ($\boldsymbol{\omega}$).

Parameter	Value	Units	Description
R_0	20 million	-	Recruitment at virgin biomass
γ_{50}	6.5	years	Logistic selectivity
γ_{95}	3	years	Logistic selectivity
q	0.006	tonnes^{-1}	Catchability coefficient
σ_o^2	0.02^2	-	Observation error variance
σ_p^2	0.001^2	-	Process error variance for $a = 2, \ldots, A$ and $t = 2, \ldots, T$
σ_R^2	0.001^2	-	Process error variance for $a = 1\ \forall t$ and $t = 1\ \forall a$
α_0	1.0	-	Dirichlet scaling
M	0.075	-	Natural mortality rate
h	0.85	-	Stock recruit steepness
A_{50}	4	years	Logistic maturity
A_{to95}	4.7	years	Logistic maturity
L_∞	58.8	cm	von Bertalanffy growth
k	0.102	years^{-1}	von Bertalanffy growth
t_0	-1.11	years	von Bertalanffy growth
α	4.467e-8	-	Length-weight
β	2.793	-	Length-weight

model specification or the MCMC sampler. Initial exploration showed that getting the numbers at age latent states ($N_{a,t}$) to mix properly was difficult, even when the key parameters were fixed. However, by reducing the recruitment variance (σ_R^2) in the simulation to the low values specified above resulted in better mixing. Including the diagonal numbers at age update also improved mixing substantially.

Two million iterations were run using blockwise Metropolis-Hastings with log-normal proposals (see Chapter 2, page 66). These two million iterations were run as two separate MCMC chains on different computer cores, both initialised with different random number seeds (the starting values were the same though). Each chain ran an additional burn-in of ten thousand iterations. The chains were thinned to save every 1000th iteration. This resulted in a total of 2000 samples from the posterior distribution for each of the parameters and latent states. This MCMC took about four days to complete.

The different components of the log-likelihood mixed well in the MCMC sampler (Figure 5.23). The numbers at age latent states ($N_{a,t}$) are reasonably well mixed. We provide a selection of MCMC trace plots for these in Figure 5.24. More trace plots for the numbers at age can be found in Appendix B.2, page 328. In most cases, the posterior density of the numbers at age states are centered over the simulated numbers at age and time (Figure 5.24). When plotted against the simulated numbers at age and time, the posterior density of the numbers at age match very closely (Figures 5.25 and 5.26). Consequently, the model fits the CPUE (I_t) and proportions in the catch at age ($(P_a)_t$) data very well (Figures 5.27, 5.28, and 5.29). Finally, looking at some of the derived quantities, the posterior distribution of YCSs for each year of the model encompass the simulated YCSs but do not show the same pattern through time that was estimated (Figure 5.30). But, remember that the recruitment variation (σ_R^2) was fixed at a very low value so strong year classes are absent from this simulated population making their estimation difficult. The estimated biomass (vulnerable, spawning stock and total) all match the simulated biomass very well (Figure 5.30).

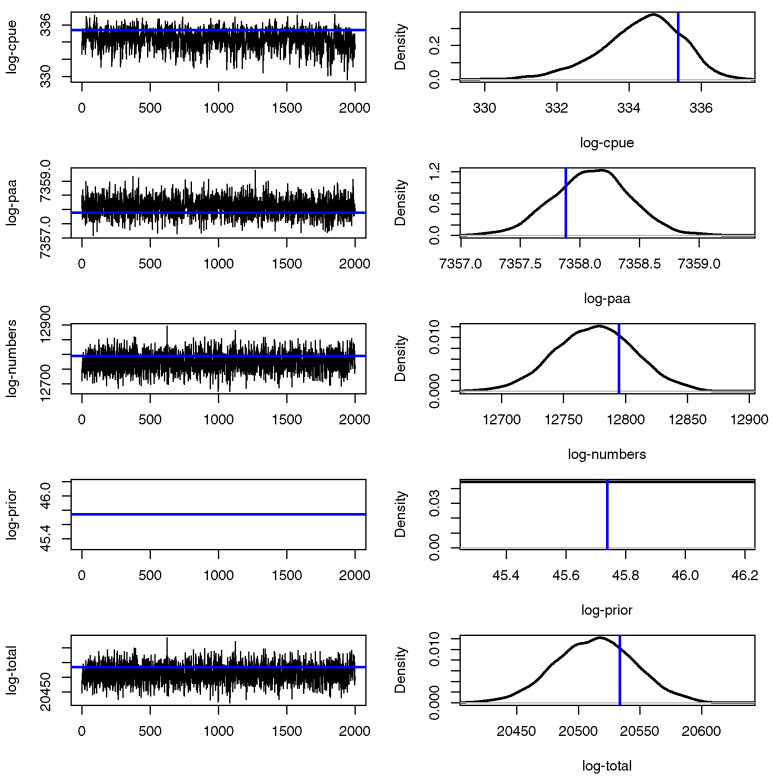


Figure 5.23: MCMC trace plots [left] and density plots [right] of the different components of the model log-likelihood including the log-likelihood of the CPUE observations, the proportions in the catch at age observations, the numbers at age latent states, the prior contribution (which is fixed as all of the key parameters are fixed) and the total log-likelihood. The horizontal blue lines indicate the log-likelihood for each of the likelihood components in the simulation given the true parameter values. All panels are made up of two separate MCMC chains of 1000 samples initialised with different random number seeds.

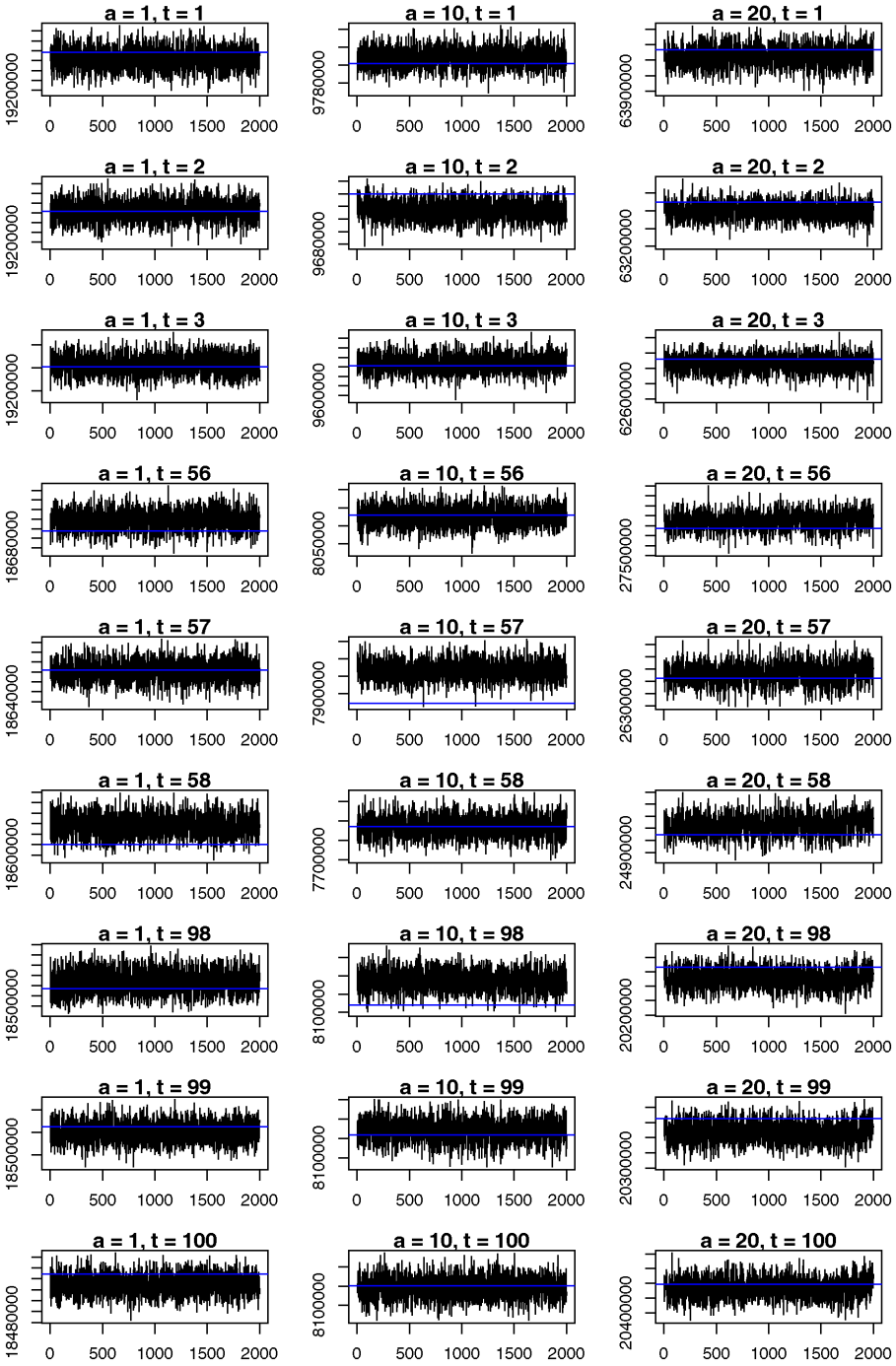


Figure 5.24: MCMC trace plots for some of the mid-year numbers at age latent states ($N_{a,t}$). The age a and time t of each latent state is given at the top of each trace plot. The simulated number at age is shown as a solid horizontal blue line. All panels are made up of two separate MCMC chains of 1000 samples initialised with different random number seeds.

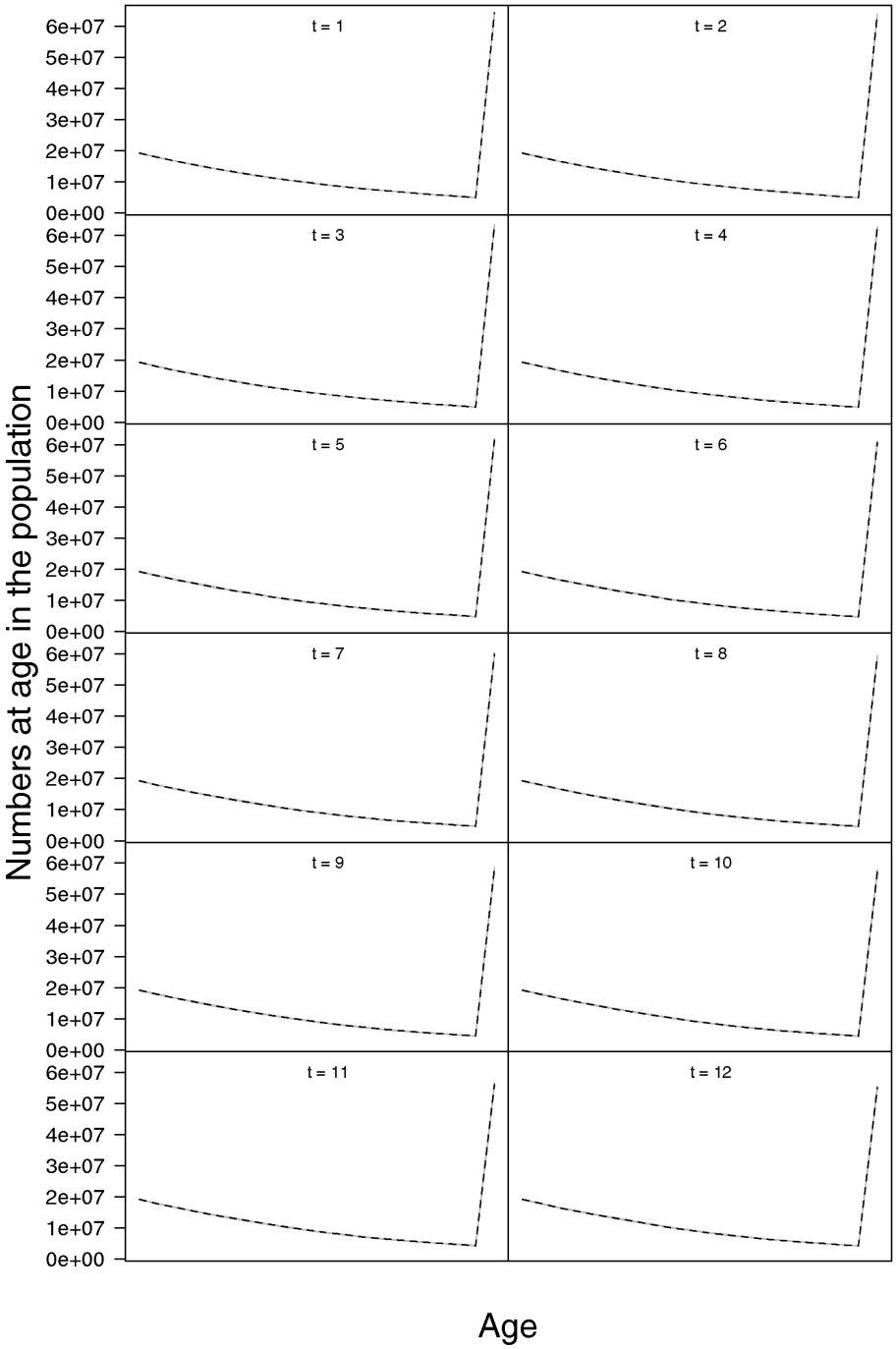


Figure 5.25: Mid-year numbers at age (a) and time (t) in the population ($N_{a,t}$) for the first 12 years in the model. In each plot the dashed black line is the simulated true value and the grey lines are the samples from the posterior distribution.

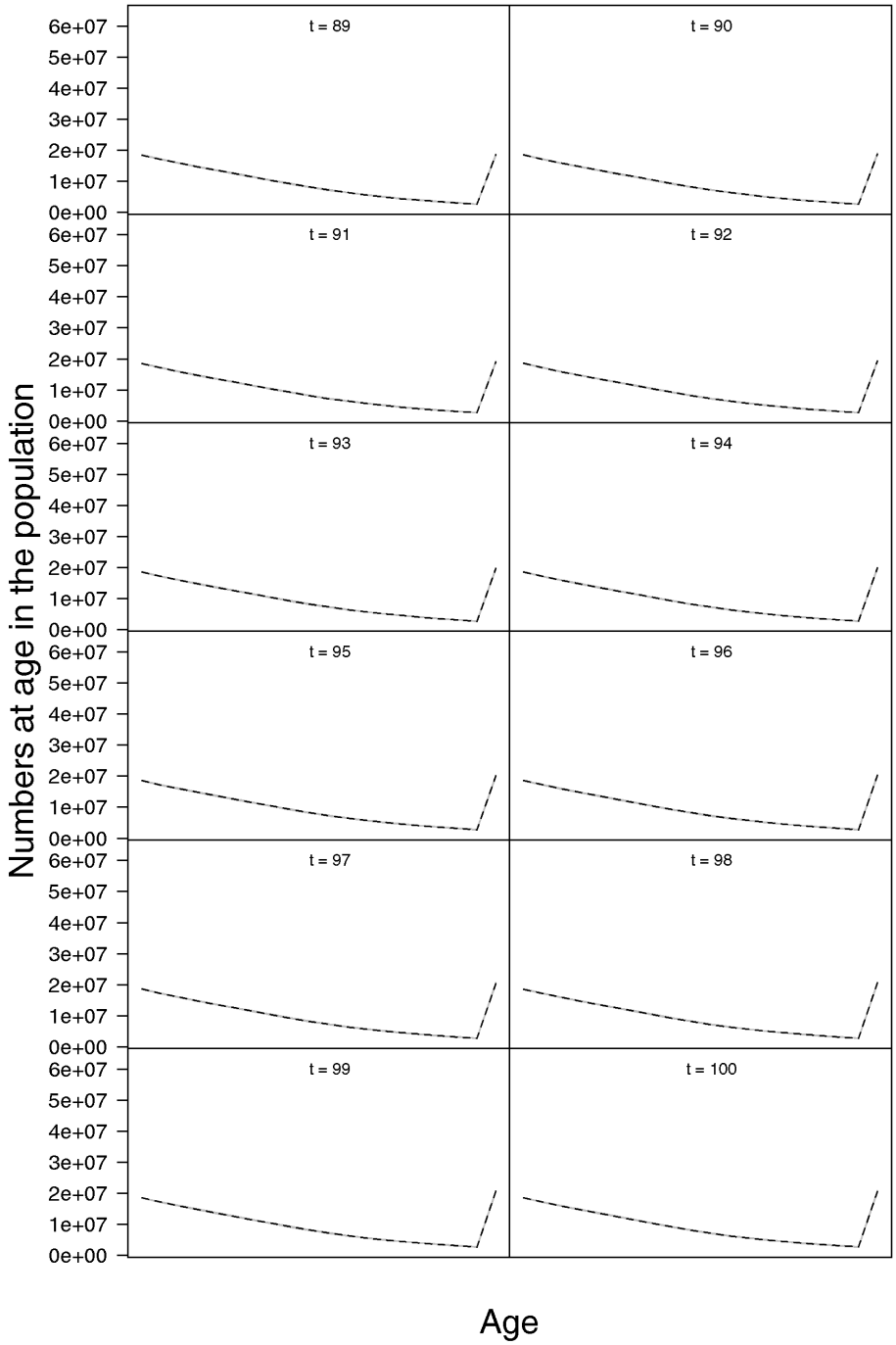


Figure 5.26: Mid-year numbers at age (a) and time (t) in the population ($N_{a,t}$) for the last 12 years in the model. In each plot the dashed black line is the simulated true value and the grey lines are the samples from the posterior distribution.

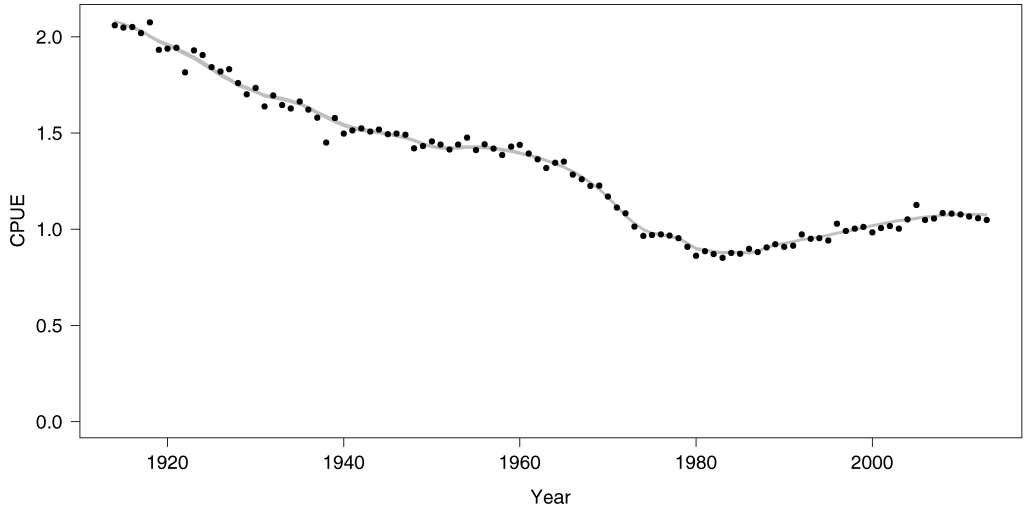


Figure 5.27: Observed CPUE (I_t) [•] and samples from the posterior distribution (qV_t) [grey lines].

In conclusion, the model appears to be able to recover the true numbers at age latent states reasonably well and the MCMC performance is acceptable.

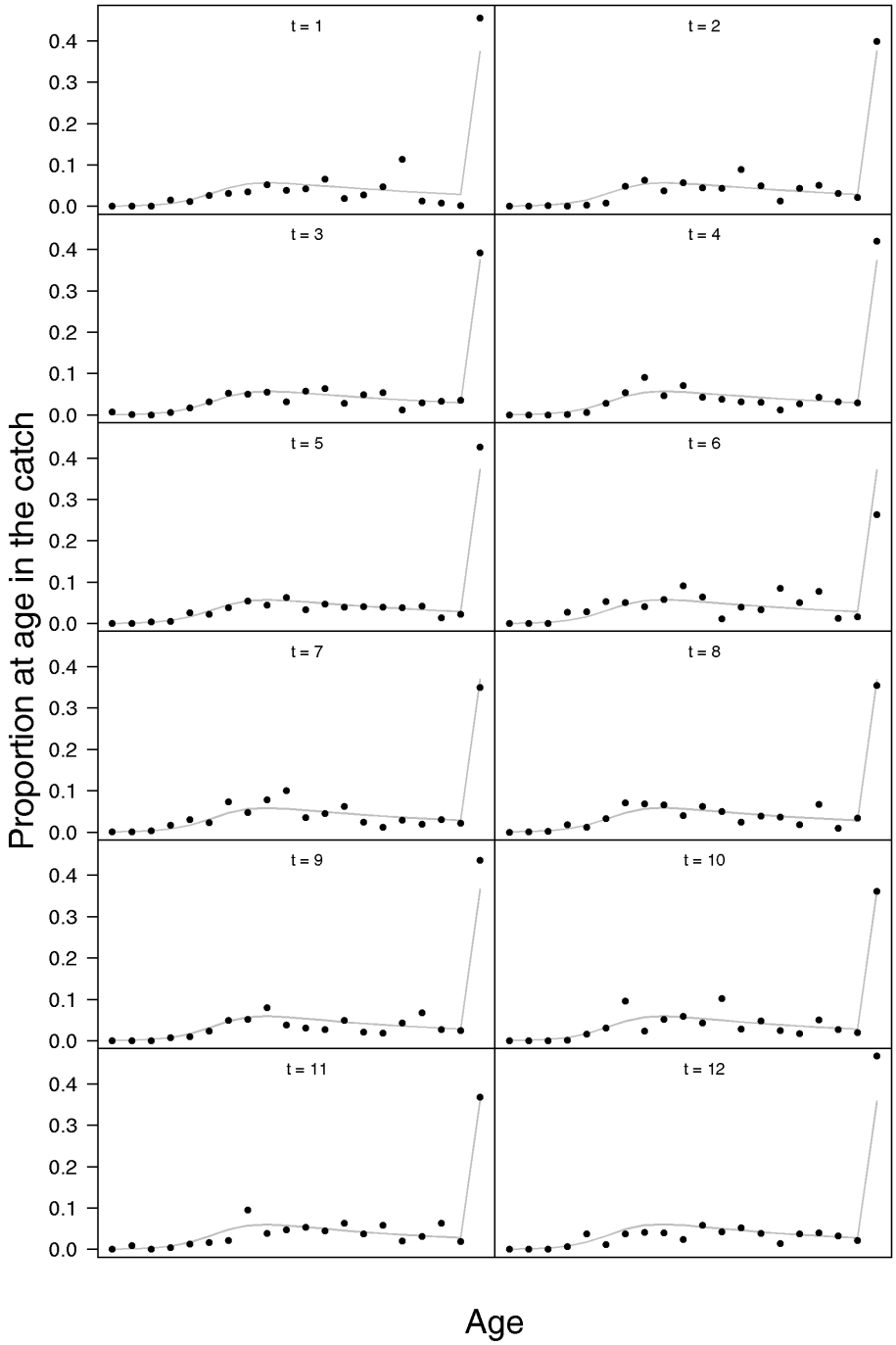


Figure 5.28: Samples from the posterior distribution of the proportions in the catch at age in the vulnerable portion of the population ($(Q_a)_t$) [grey lines] and observed proportions in the catch at age ($(P_a)_t$) [•] for the first 12 months of the model.

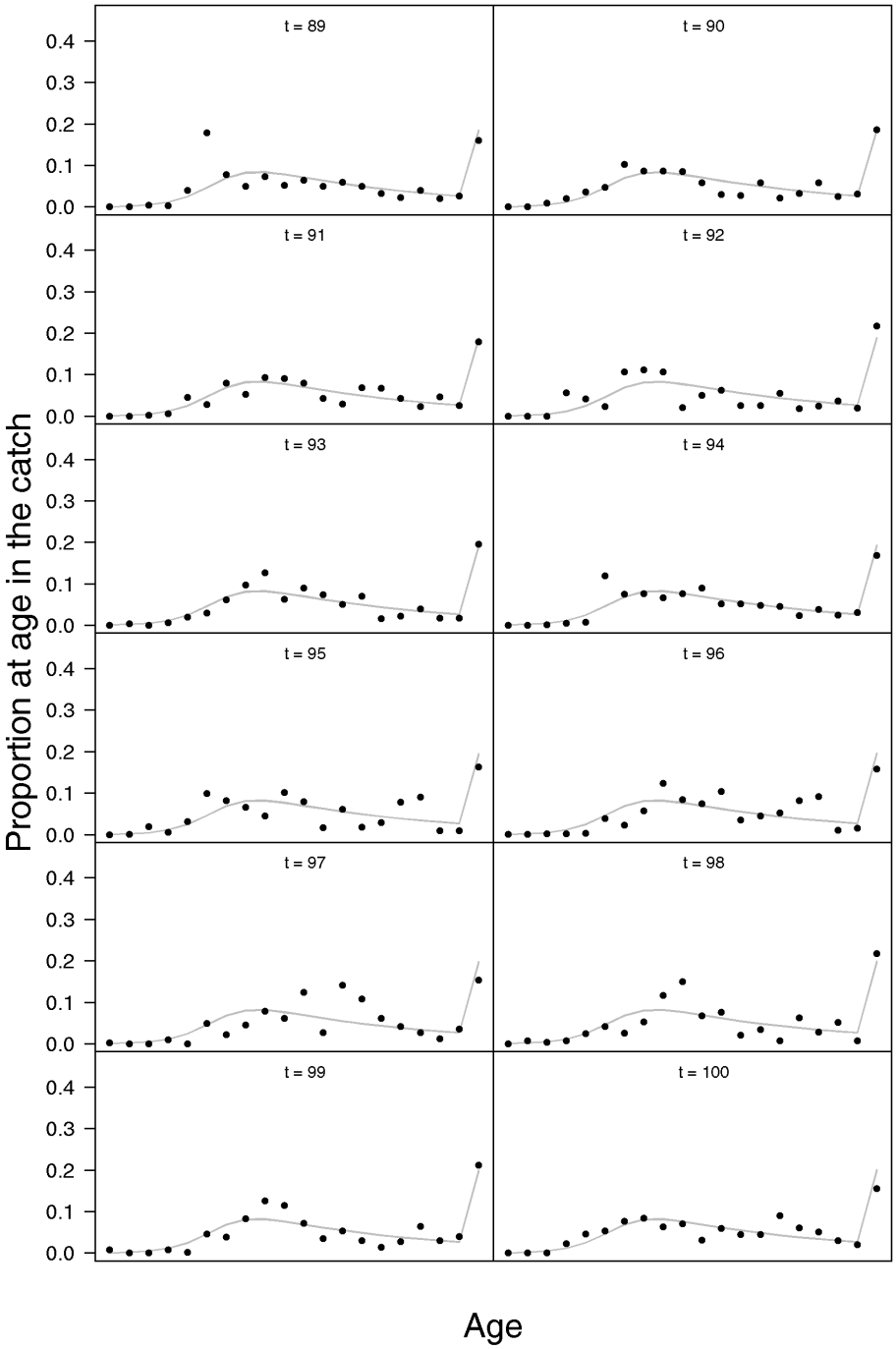


Figure 5.29: Samples from the posterior distribution of the proportions in the catch at age in the vulnerable portion of the population ($(Q_a)_t$) [grey lines] and observed proportions in the catch at age ($(P_a)_t$) [•] for the last 12 months of the model.

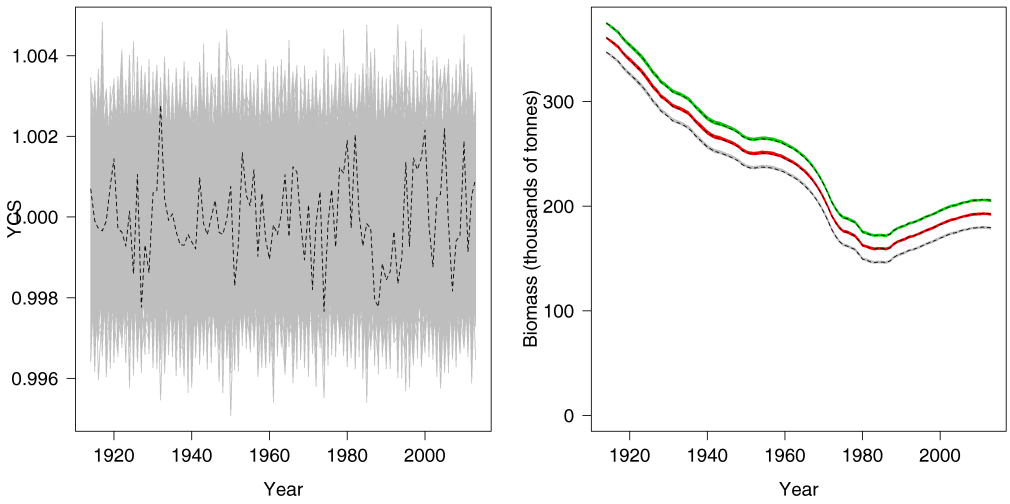


Figure 5.30: Year class strengths by year ($YCS_t = e^{\varepsilon^p_{a=1,t} - \sigma^2_R/2}$) [left], and measures of biomass in the population (including the total biomass (B_t) in green, the spawning stock biomass (SSB_t) in red and the vulnerable biomass (V_t) in grey) [right]. In each plot the dashed black line is the simulated true value and the coloured lines are the samples from the posterior distribution.

Model fit (fixed process error)

The second example uses exactly the same parameter set specified in Table 5.2 in the simulation for this example. However, now we allow all but one of the key parameters to be estimated. We leave the process error variance (σ_p^2) fixed and begin by specifying highly informative priors for each of the remaining model parameters to check for any problems in the MCMC. The full list of priors is

$$\begin{aligned}
\pi(R_0) &\sim \log\mathcal{N}(\log(20000000), 0.005),\\
\pi(\gamma_{50}) &\sim \log\mathcal{N}(\log(6.5), 0.005),\\
\pi(\gamma_{95}) &\sim \log\mathcal{N}(\log(3), 0.005),\\
\pi(q) &\sim \log\mathcal{N}(\log(0.006), 0.003),\\
\pi(\sigma_o^2) &\sim \log\mathcal{N}(\log(0.02^2), 0.05),\\
\pi(\sigma_R^2) &\sim \log\mathcal{N}(\log(0.001^2), 0.05),\\
\pi(\alpha_0) &\sim \log\mathcal{N}(\log(1), 0.1).
\end{aligned} \tag{5.37}$$

Two million iterations were run using blockwise Metropolis-Hastings with log-normal proposals (see Chapter 2, page 66). These three million iterations were run as three separate MCMC chains on different computer cores, all initialised with different random number seeds. Each chain ran an additional burn-in of two million iterations. The chains were thinned to save every 1000th iteration. This resulted in a total of 1000 samples of the posterior distribution for each of the parameters and latent states.

Both the key model parameters and the different components of the log-likelihood mixed reasonably well in the MCMC sampler (Figures 5.31 and 5.32). The numbers at age latent states ($N_{a,t}$) are reasonably well mixed. We provide a selection of MCMC trace plots for these in Figure 5.33. More trace plots for the numbers at age can be found in Appendix B.3, page 328. In most cases, the posterior density of the numbers at age states are centered over the simulated numbers at age and time (Figure 5.24). When plotted against the simulated numbers at age and time, the posterior density of the numbers at age match very closely (Figures 5.34 and 5.35). Consequently, the model fits the CPUE (I_t) and

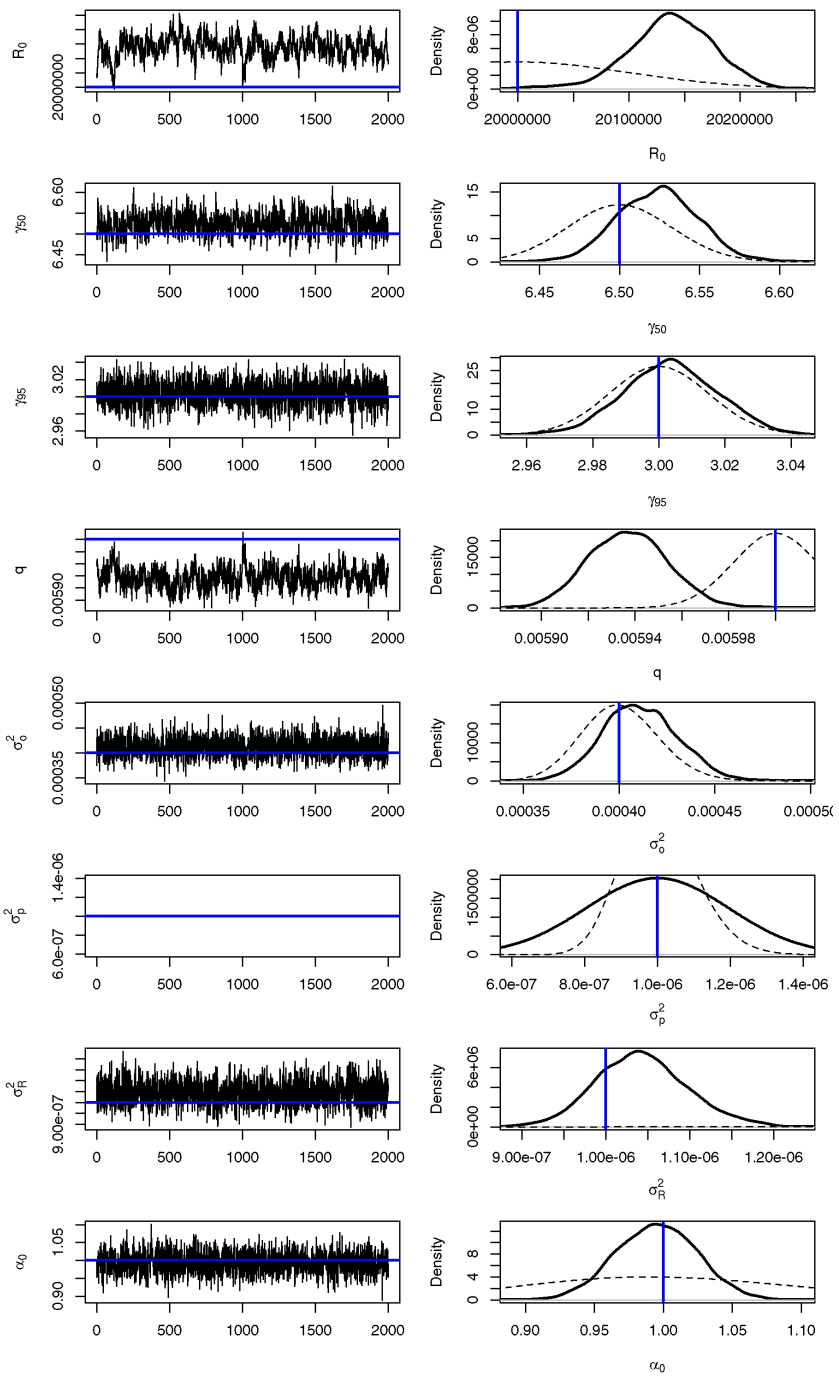


Figure 5.31: MCMC trace plots [left] and posterior densities [right] for the key model parameters. The horizontal blue lines indicate the true parameter values as specified in the simulation. All panels are made up of two separate MCMC chains of 1000 samples initialised with different random number seeds.

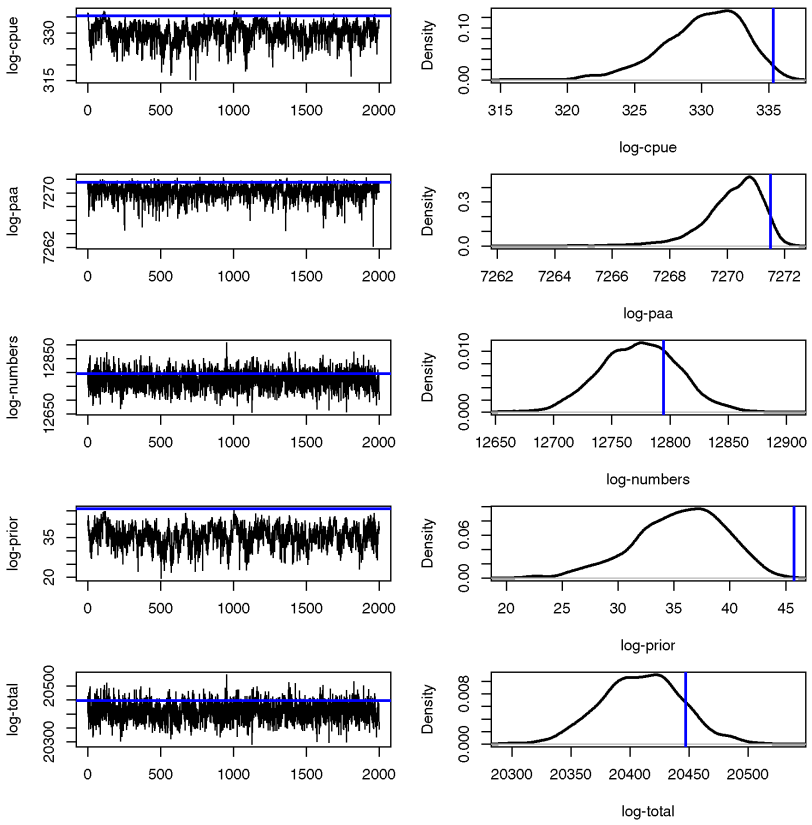


Figure 5.32: MCMC trace plots [left] and density plots [right] of the different components of the model log-likelihood including the log-likelihood of the CPUE observations, the proportions in the catch at age observations, the numbers at age latent states, the prior contribution and the total log-likelihood. The horizontal blue lines indicate the log-likelihood for each of the likelihood components in the simulation given the true parameter values. All panels are made up of two separate MCMC chains of 1000 samples initialised with different random number seeds.

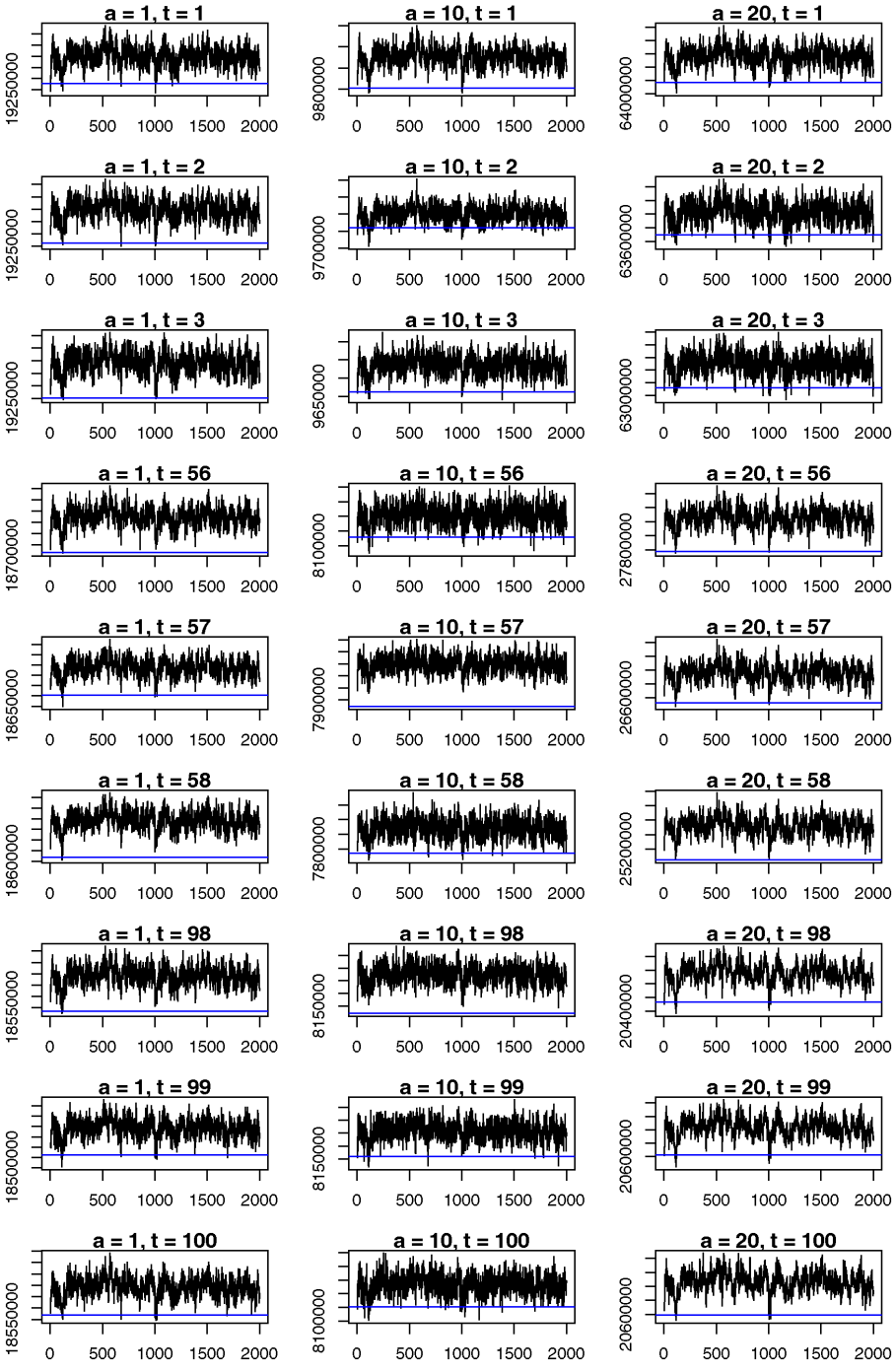


Figure 5.33: MCMC trace plots for some of the mid-year numbers at age latent states ($N_{a,t}$). The age a and time t of each latent state is given at the top of each trace plot. The simulated number at age is shown as a solid horizontal blue line. All panels are made up of two separate MCMC chains of 1000 samples initialised with different random number seeds.

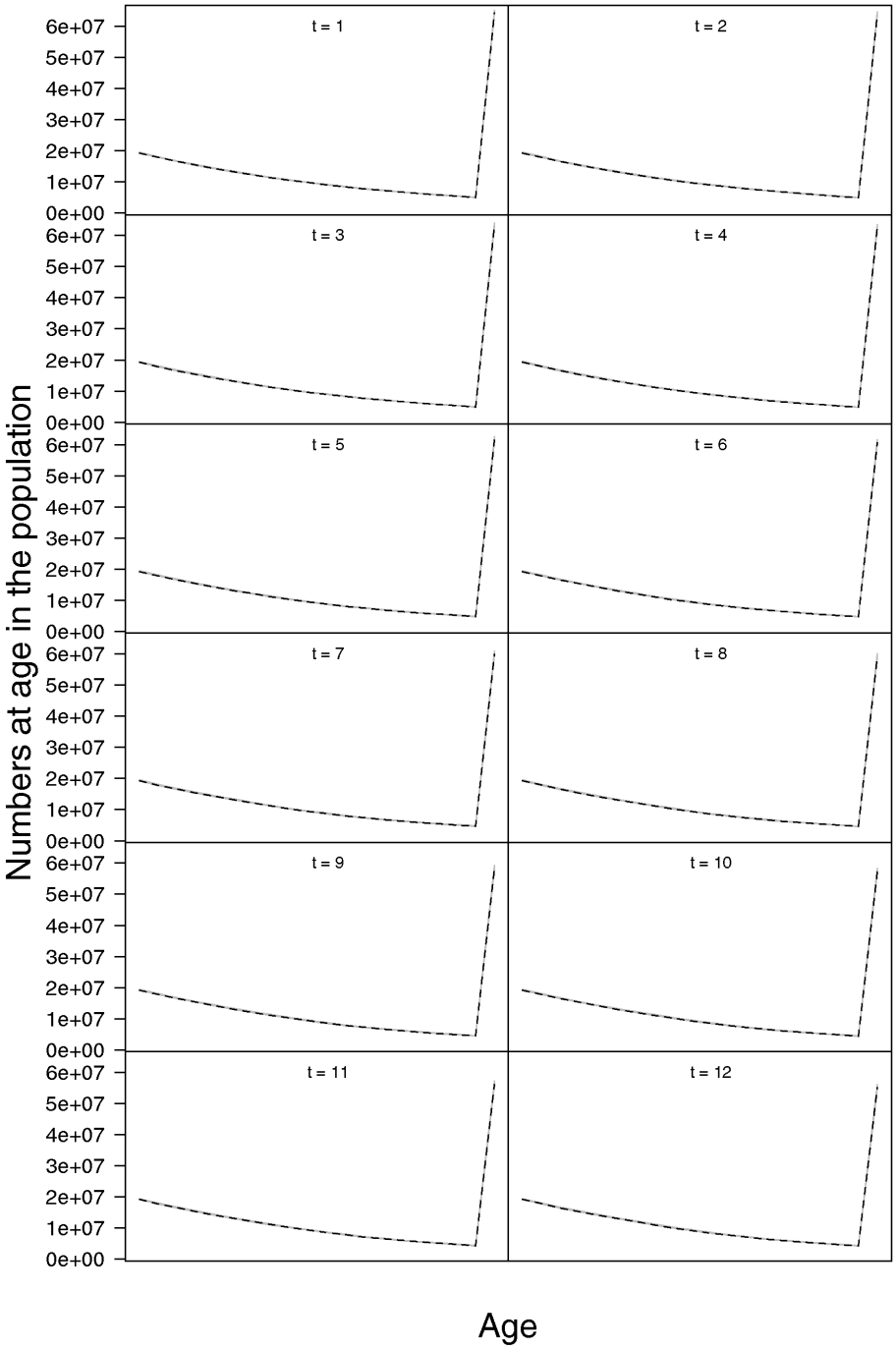


Figure 5.34: Mid-year numbers at age (a) and time (t) in the population ($N_{a,t}$) for the first 12 years in the model. In each plot the dashed black line is the simulated true value and the grey lines are the samples from the posterior distribution.

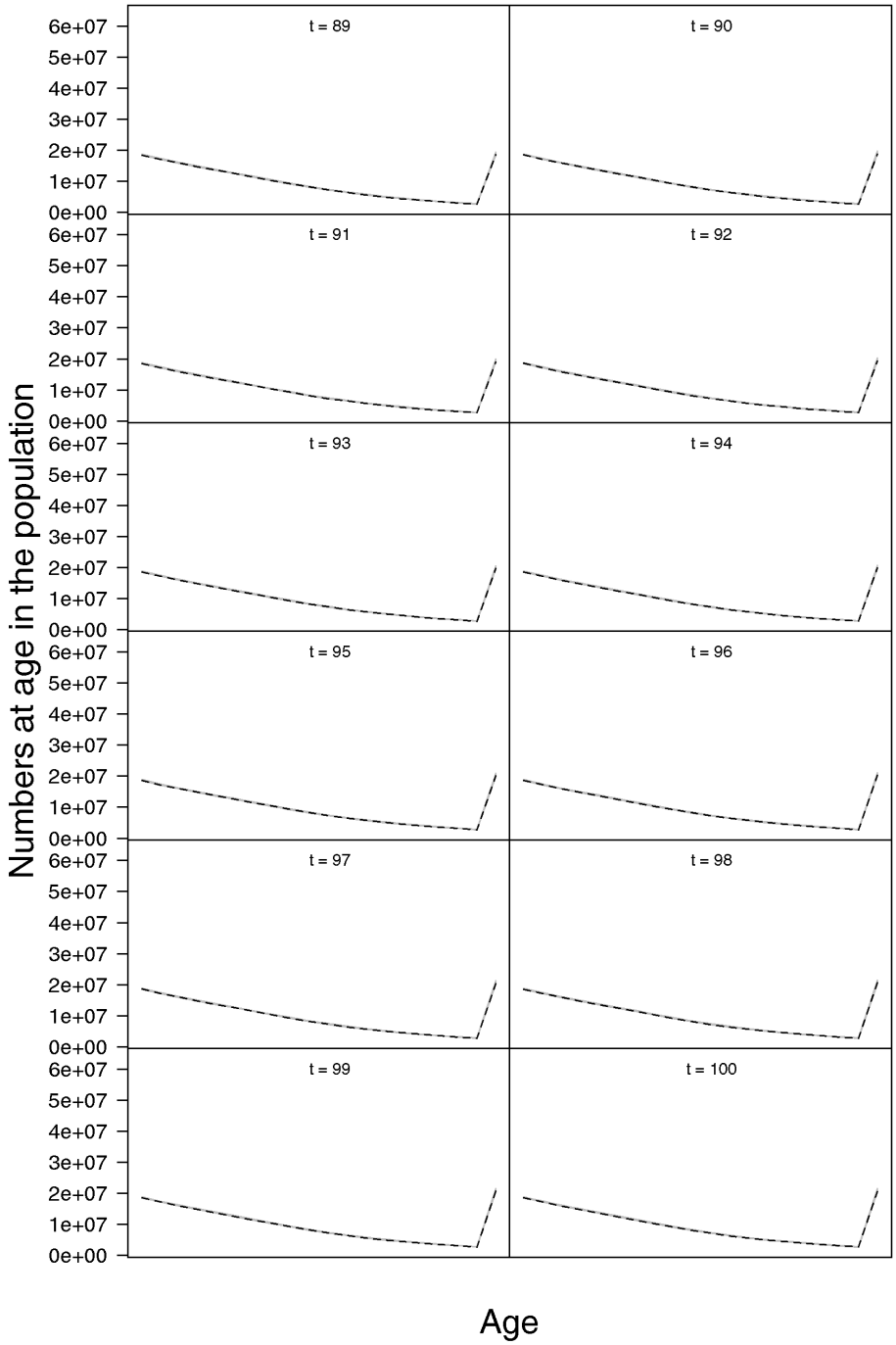


Figure 5.35: Mid-year numbers at age (a) and time (t) in the population ($N_{a,t}$) for the last 12 years in the model. In each plot the dashed black line is the simulated true value and the grey lines are the samples from the posterior distribution.

proportions in the catch at age ($(P_a)_t$) data very well (Figures 5.36, 5.37, and 5.38). Finally, looking at some of the derived quantities, the poste-

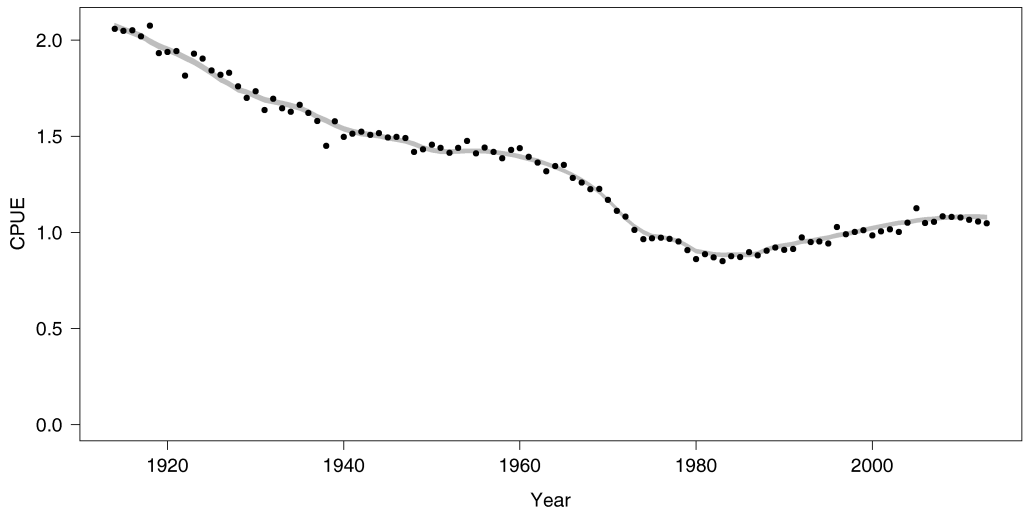


Figure 5.36: Observed CPUE (I_t) [•] and samples from the posterior distribution (qV_t) [grey lines].

rior distribution of YCSs for each year of the model encompass the simulated YCSs but are not well estimated (Figure 5.39), remembering that the recruitment variation (σ_R^2) was fixed at a very low value so strong year classes are absent from this simulated population making their estimation difficult. The estimated Dirichlet distribution scaling ($\alpha_0\alpha_t$) was very well estimated, as was the selectivity at age curve (Figure 5.30). Finally, the estimated biomass (vulnerable, spawning stock and total) all match the simulated biomass very well (Figure 5.30).

In this example the model appears to be able to recover the true numbers at age latent states and key parameters reasonably well and the MCMC performance is acceptable.

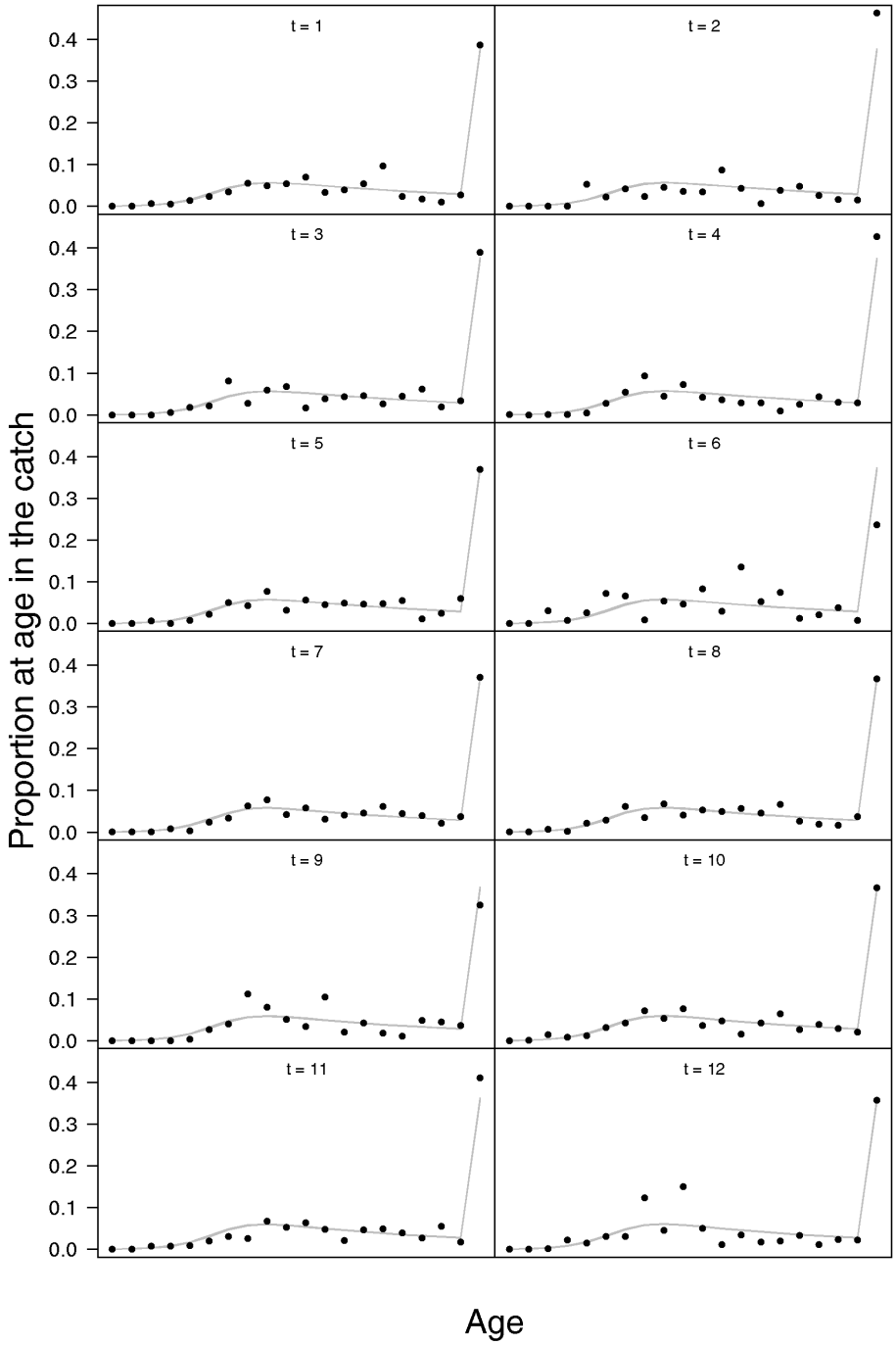


Figure 5.37: Samples from the posterior distribution of the proportions in the catch at age in the vulnerable portion of the population ($(Q_a)_t$) [grey lines] and observed proportions in the catch at age ($(P_a)_t$) [•] for the first 12 months of the model.

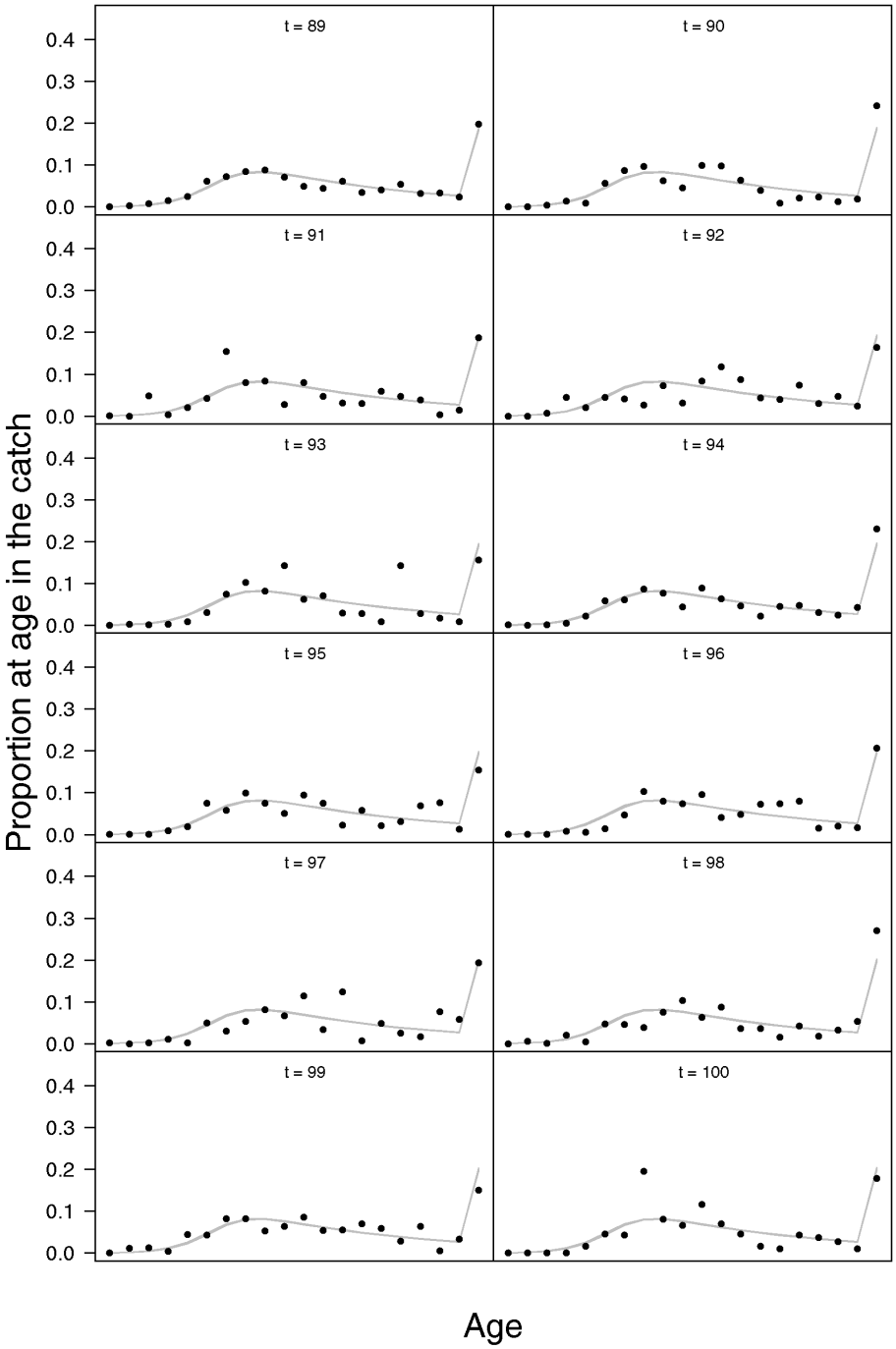


Figure 5.38: Samples from the posterior distribution of the proportions in the catch at age in the vulnerable portion of the population ($(Q_a)_t$) [grey lines] and observed proportions in the catch at age ($(P_a)_t$) [•] for the last 12 months of the model.

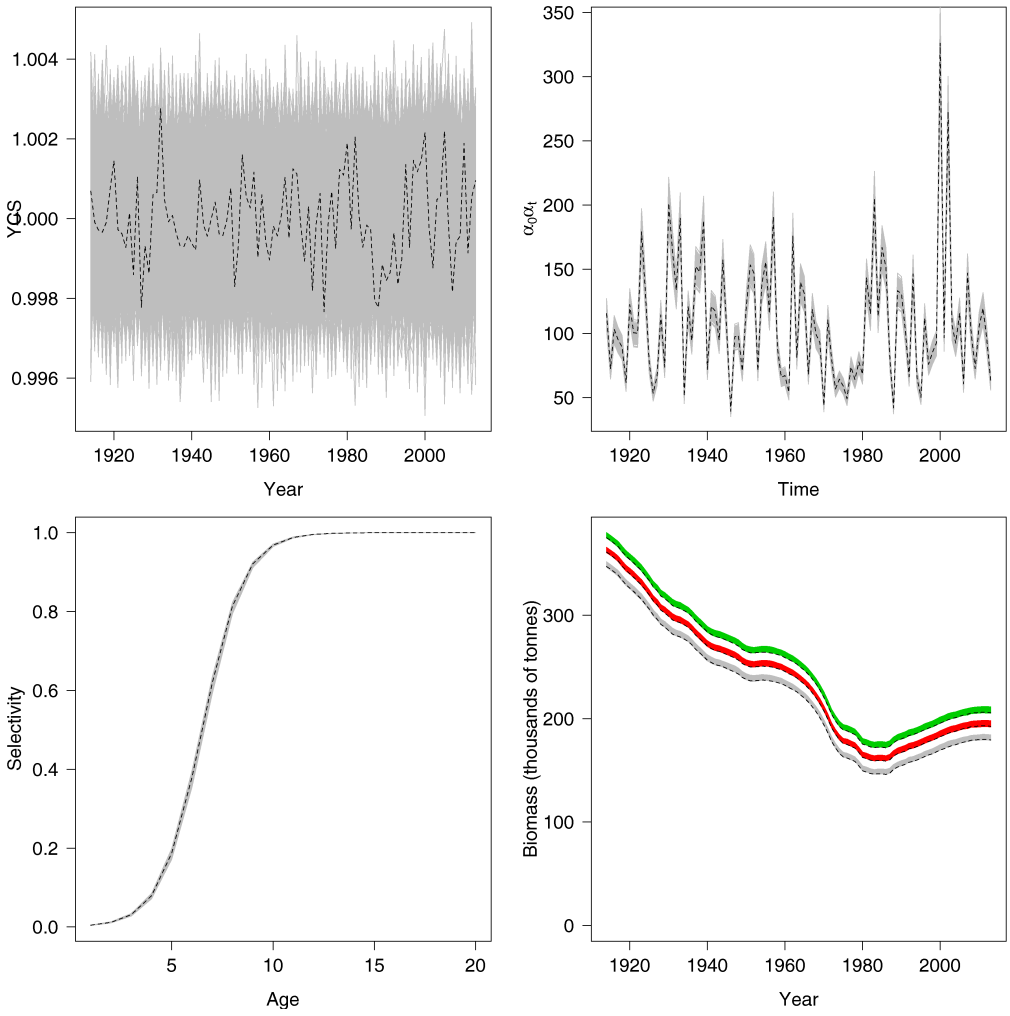


Figure 5.39: Year class strengths by year ($YCS_t = e^{\varepsilon^p_{a=1,t} - \sigma^2_R/2}$) [top left], variability of the Dirichlet distribution each year ($\alpha_0\alpha_t$) [top right], selectivity at age (S_a) [bottom left] and biomass (tonnes) by year (including the total biomass (B_t) in green, the spawning stock biomass (SSB_t) in red and the vulnerable biomass (V_t) in grey) [bottom right]. In each plot the dashed black line is the simulated true value and the grey lines are the samples from the posterior distribution.

Model fit (releasing σ_R^2)

The final test we do is to test the models performance when supplied data simulated with more realistic recruitment variability. The final example uses the same parameter set specified in Table 5.2 in the simulation except for the recruitment variance parameter (σ_R^2) which is changed to be $\sigma_R^2 = 0.02^2$ (rather than $\sigma_R^2 = 0.001^2$). Despite this increase, we are still not really using a realistic recruitment variability value, a realistic value might be $\sigma_R^2 = 0.3^2$. However, as we show below, the model struggles with the recruitment variability defined above.

Again, we leave the process error variance (σ_p^2) fixed and begin by specifying highly informative priors for each of the model parameters to check for any problems in the MCMC. The recruitment variance prior is the only change here

$$\pi(\sigma_R^2) \sim \log\mathcal{N}(\log(0.02^2), 0.05).$$

Unfortunately, mixing was too slow and MCMC trace plots show that the model does not converge (Figures 5.40 and 5.41). Trace plots for the numbers at age latent states can be found in Appendix B.4, page 336. Many attempts were made to improve the mixing in the MCMC given realistic simulations (i.e. that used realistic recruitment variability parameters) including MCMC runs that took over a week to complete. However, exploratory runs suggest that even if all of the key parameters are fixed, with a high recruitment variance, adequate mixing is still not achieved in the numbers at age and time latent states. We discuss these results further below.

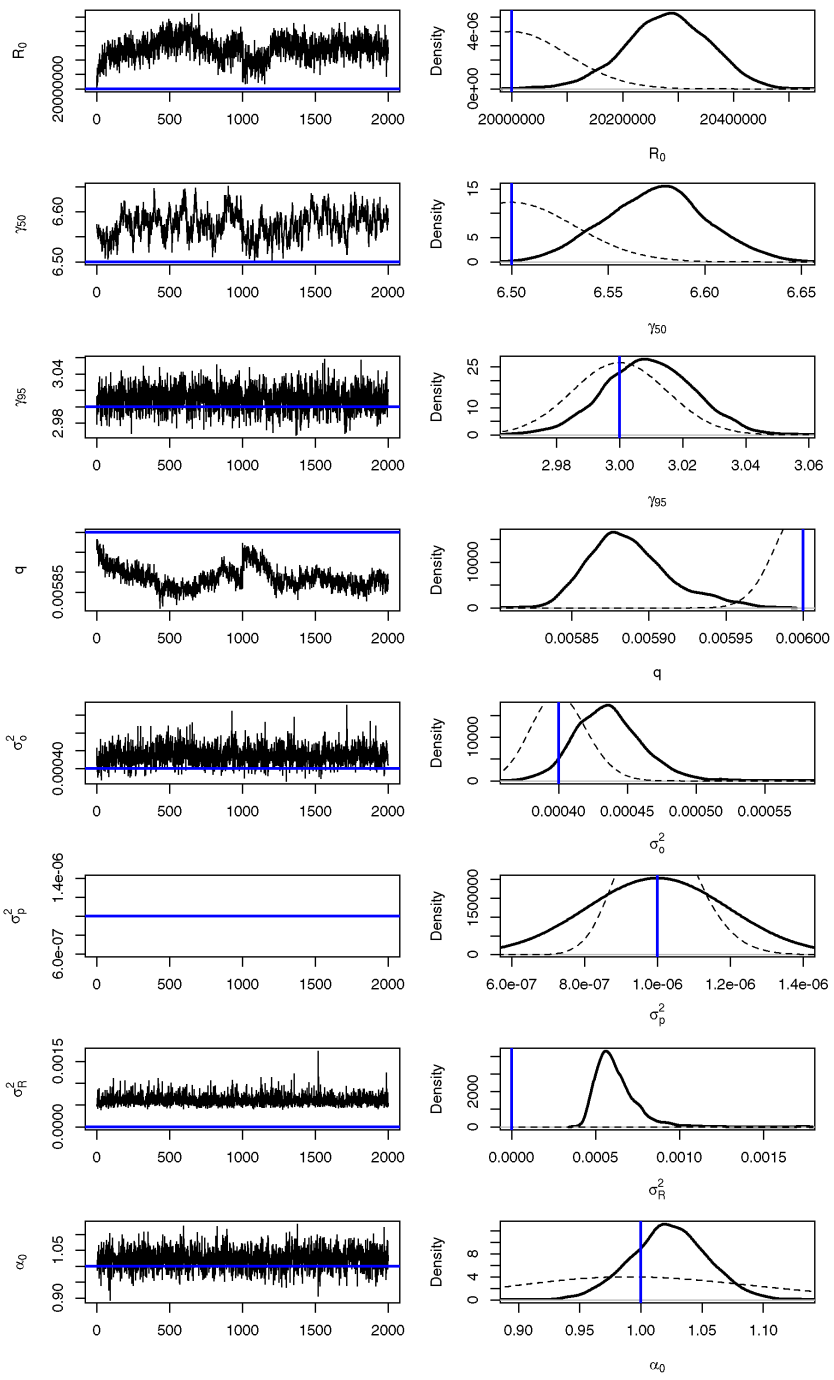


Figure 5.40: MCMC trace plots [left] and posterior densities [right] for the key model parameters. The horizontal blue lines indicate the true parameter values as specified in the simulation. All panels are made up of two separate MCMC chains of 1000 samples initialised with different random number seeds.

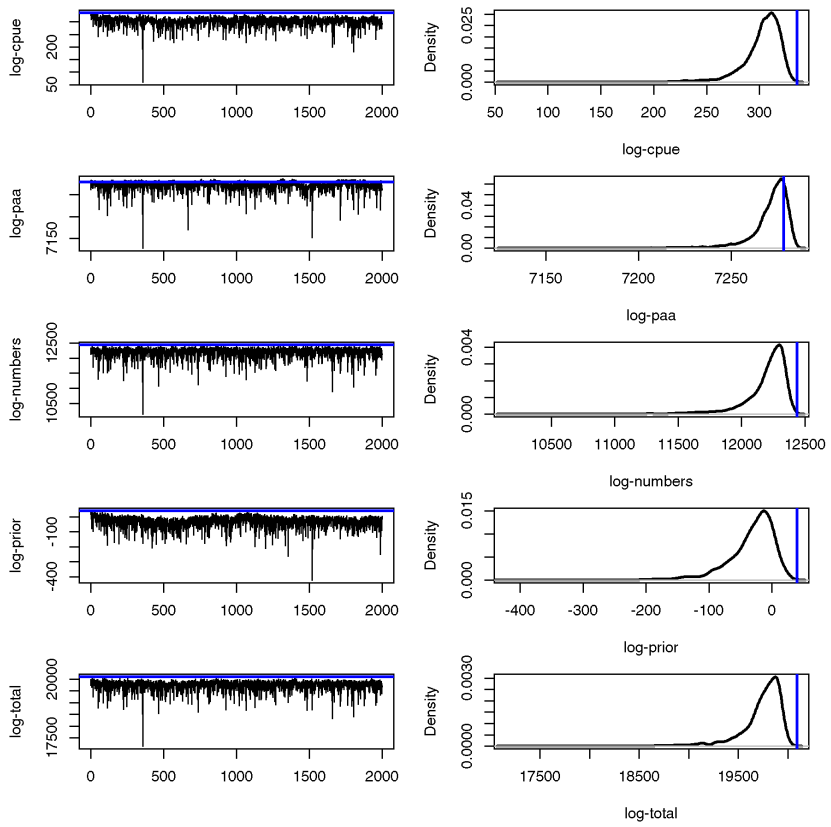


Figure 5.41: MCMC trace plots [left] and density plots [right] of the different components of the model log-likelihood including the log-likelihood of the CPUE observations, the proportions in the catch at age observations, the numbers at age latent states, the prior contribution and the total log-likelihood. The horizontal blue lines indicate the log-likelihood for each of the likelihood components in the simulation given the true parameter values. All panels are made up of two separate MCMC chains of 1000 samples initialised with different random number seeds.

5.5 Discussion

The aim of this chapter was to introduce state-spaces models, to illustrate how Bayesian inference performs when estimating observation and process error simultaneously, and to develop an age-structured state-space model that might better quantify the uncertainty in parameters of interest within stock assessment.

Biomass dynamics models were used to introduce state-space modelling in fisheries stock assessment. While biomass dynamics models were not the focus of this chapter, they do provide some insight into model behaviour in the face of uncertainty. Estimating model parameters in simulated biomass dynamics systems is easy if uncertainty is minimal (i.e. low observation and process error) or if informative priors are used. High uncertainty or uninformative priors make parameter estimation more difficult. Unfortunately, the latter is the reality. Uncertainty is a defining feature of stock assessment and rarely do we have much prior knowledge about key parameters. However, sensible priors for key parameters can be developed for such models. Examples illustrated in this chapter include log-normal priors with high variance (Equations 5.11 and 5.12) or the relatively informative prior developed for the magnitude of process error (σ_p^2, Equations 5.15 and 5.16).

Next, we developed an age-structured state-space model that includes process error in the mid-year numbers at age in the population. The aim here was to develop a more probabilistic approach to age-structured stock assessment modelling, by replacing the standard deterministic population dynamics equations with fully state-space formulations. This model should better represent uncertainty in the estimates they provide. A correctly specified model like this should also avoid the data weighting paradigm.

It is common practice in stock assessment to assign relative weights to different data sets (i.e. age composition data is often assigned a lower weight relative to abundance indices). This is done so that the model fits to abundance index data primarily, and then fits to age-composition data

so long as the fit to these data does not result in a poor fit to the abundance data. Quoting Francis (2011), if we do not assign a lower weight to the composition data "there is a danger that any signal from abundance data will be swamped by that from composition data, simply because the latter data type is typically much more numerous (in terms of individual data points". This holds true for standard stock assessment models that are based on deterministic population dynamics equations and often use the multinomial distribution in fitting to composition data. However, a properly defined probabilistic Bayesian approach should not require data weighting because a prior distribution becomes the appropriate mechanism by which we specify our prior uncertainty. For example, a prior distribution can be used to inform the scaling parameter of the Dirichlet distribution, which is essentially analogous to data weighting.

Estimation of model parameters and latent states in our age-structured state-space was done using blockwise Metropolis-Hastings, avoiding the need for maximum likelihood methods and the need to estimate a variance covariance matrix for multivariate proposals. Treating the numbers at age and time ($N_{a,t}$) for all ages and times as latent states requires thousands of different MCMC proposals within a single MCMC step. The computational workload here was reduced by splitting the likelihood up into smaller components and only evaluating those components of the likelihood that require evaluation during each of the MCMC proposals.

Initial exploration suggested that mixing was slow for the numbers at age and time latent states ($N_{a,t}$). To help speed up mixing, a diagonal cohort proposal was developed (Section 5.4.1, page 177). This proposal updated the numbers at age for each diagonal element of the $N_{a,t}$ matrix (cohort) in the model by scaling the numbers at age within each cohort by a common factor (λ^*). The goal here was to improve mixing within cohorts by updating the entire cohort at the same time, thus increasing the acceptance rate of proposals.

This diagonal update proposal did improve mixing substantially. However, it was not enough. When the recruitment variation (σ_R^2) is low, the model is well behaved and efficient sampling of the posterior using

MCMC can be achieved. However, when σ_R^2 is increased (in this case to $\sigma_R^2 = 0.02^2$) then the MCMC fails to mix properly, even with millions of iterations. The high correlation between the models latent states makes efficient sampling difficult.

Future research should focus on proposals that speed up mixing for the numbers at age and time ($N_{a,t}$) latent states. For example, rather than the diagonal cohort update that we implemented, an update based on the expected value ($\mu_{a,t}$) could be developed. Here one might draw $N_{a=1,t}$ for any selected year t, then update $N_{a,t}$ for years $t+1, \ldots, T$. The challenge here would be deriving the correct acceptance ratio (r). Failing this, the state-space formulation presented here requires simplification. An obvious fix would be to drop the state-space structure for some of the numbers at age and time latent states. For example, treating $N_{a=1,t}\ \forall t$ (i.e. YCS_t as is done in current stock assessment models, row 1) and for $N_{a,t=1}\ \forall a$ (i.e. non-equilibrium numbers at age in the first year, column 1) as latent states. This is similar to what happens in current stock assessment methods.

Another area that requires further thought is around the process error structure. Composition data that enables us to detect year classes is rarely available in the early years of stock assessment models so there may be little point in modelling non-equilibrium numbers at age in practice. Instead of using the process error matrix ($\sigma_{a,t}^2$) defined in Equation 5.28 and Figure 5.20, the errors could be structured in a way that allows year classes to be semi-deterministic at the beginning of the model (e.g. Figure 5.42).

We used the programming language Julia to implement the simulation model and MCMC sampler for this model. These types of models require us to sample many millions of times from the posterior distribution, and this is time consuming. Julia provides the best of both worlds with computational performance approaching that of C (it is faster than C in some cases) and a user friendly language with the ability to run interpreted code (i.e. line-by-line as in R) or fully compiled at the command line (like C or C++) for maximum speed. This provides the programmer the ability to check code a single line at a time when debugging, or run as a fast MCMC sampler/computer program.

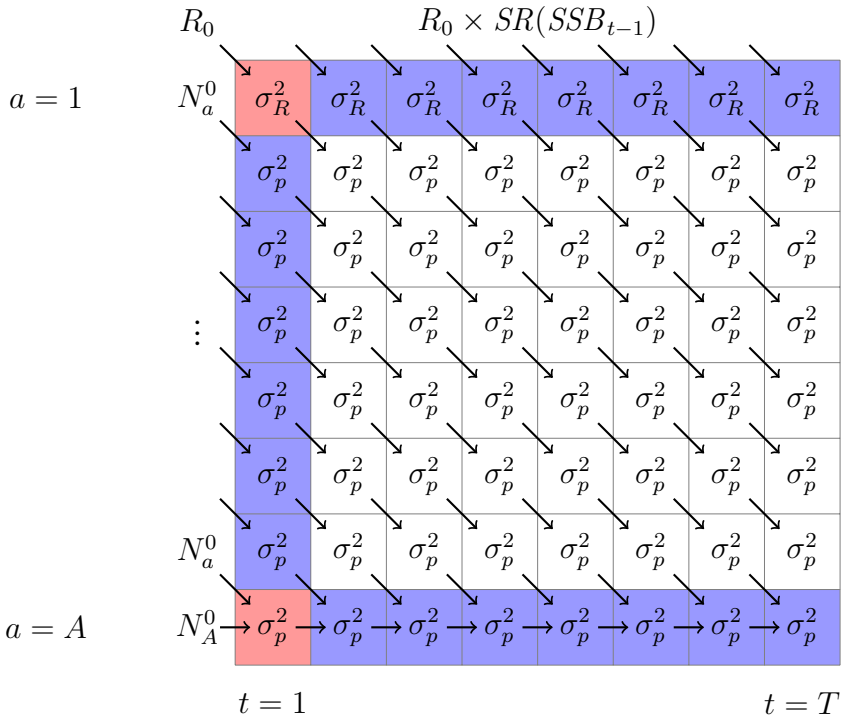


Figure 5.42: Matrix showing the direction of transitions between numbers of fish at age and time ($N_{a,t}$) within the model and the process errors variance to be used at each age a and time t. The colours are used to highlight the 6 parts of Equation 5.21.

While this is a sophisticated age-structured model, it pushes the boundaries of the practical limits of computing. The continual improvements being made to Julia may help speed up the model in the future, as would the use of high performance computers. The model is already multi-threaded, allowing multiple chains to be passed to individual computer cores to run independently. These are then amalgamated at the conclusion of MCMC sampling. However, the most successful speed gains are often achieved with minor changes to the code (e.g. reparameterisation, more efficient code). Also, MCMC can be made better with smarter proposals, an area worthy of further research.

In summary, the major contribution provided in this chapter is the construction of the posterior for an state-space age-structured model. The future challenge is to sample from this posterior efficiently.

Chapter 6

Pop-up satellite archival tagging

A pop-up satellite archival tag (PSAT) is an archival tag (or data logger) that can be attached to a fish to record data measured using sensors in the tag. PSATs can then detach from the fish after a prespecified length of time and are able to transmit the collected data via satellite. Alternatively, the fish may be recaptured and the tag recovered[1]. In this chapter we discuss the development and application of a state-space model that aims to estimate the path taken by a fish between two points (tag-release and tag-recapture location) using data recorded by a PSAT attached to the fish. We use Bayesian inference methods to estimate the posterior distributions of model parameters and latent states (the position of a fish at discrete time-steps). We use data from a PSAT deployed on an Antarctic toothfish (*Dissostichus mawsoni*) in the Ross Sea as a case study (for more information on Antarctic toothfish see Chapter 3, page 73). Lists of the variables used throughout this chapter are given in Tables 6.1 and 6.2.

6.1 Introduction

One of the important information needs influencing the stock assessment and management of fish stocks is understanding their movements and mi-

[1]Usually a lot more data is recorded by the tag than is transmitted via satellite. If the tag is recovered then a lot more data is available, but this is not always possible.

Table 6.1: Description of variables, including their dimensions and units, used throughout this chapter when discussing the process model. Equal-area refers to the variable being in an equal-area projection (Section 6.1.3, page 216).

Symbol	Dimension	Units	Description
T	1	hours or weeks	Number of time-steps
t	1	hours or weeks	A single time-step where $t = 0, \ldots, T$
$\mathbf{x}_t$	2	m	Equal-area projected position of a fish at time t, $\mathbf{x}_t = (x_t, y_t)$ (Equation 6.5)
$\mathbf{x}_0$	2	m	Start position (equal-area), $\mathbf{x}_0 = \mathbf{x}_{t=0}$ (Equation 6.5)
$\mathbf{x}_T$	2	m	End position (equal-area), $\mathbf{x}_T = \mathbf{x}_{t=T}$ (Equation 6.5)
$\sigma_{\mathbf{x}}$	1	m	Standard deviation of the hourly or weekly horizontal movement of a fish (Equation 6.5)
ϕ	1	degrees	Latitude (Equation 6.1)
λ	1	degrees	Longitude (Equation 6.1)
ϕ_0	1	degrees	Projection centre (latitude, Equation 6.1)
λ_0	1	degrees	Projection origin (longitude, Equation 6.1)
R	1	m	Radius of the Earth (e.g. $R = 6378137$m at the South Pole, Equation 6.3)

Table 6.2: Description of variables, including their dimensions and units, used throughout this chapter when discussing the observation models.

Symbol	Dimension	Units	Description
a_t	T	g	Fish acceleration at time t (1g = 9.8ms^{-1})
b_t	T	nT	Magnetic field strength at time t
c_t	T	°C	Sea temperature at time t (Equations 6.7, 6.8, and 6.9)
d_t	T	m	Fish depth at time t (Equations 6.10, 6.11, and 6.12)
m_t	T	months	Month at time t
$\mu^c_{\mathbf{x},d,m}$	$31 \times 121 \times 79 \times 12$	°C	Temperature covariate layer (latitude $\times$ longitude $\times$ depth $\times$ month, Equations 6.7 and 6.9)
$\sigma^c_{\mathbf{x},d}$	$31 \times 121 \times 79$	°C	Standard deviation of temperature covariate layer (latitude $\times$ longitude $\times$ depth, Equations 6.7 and 6.9)
σ^c	1	°C	Standard deviation of temperature when assumed constant with location and depth (Equations 6.8 and 6.9)
$\mu^d_{\mathbf{x}}$	1001×5001	m	Depth covariate layer (latitude $\times$ longitude, Equations 6.10 and 6.11)
$\boldsymbol{\sigma}^d$	2	m	Split-normal standard deviation of depth (Equation 6.11)

grations. Modelling the movements of fish populations requires estimates of the routes, timing, and duration of movements, not simply demonstrating a link between geographical regions. This includes information on fish that may migrate to previously unfished or closed areas (e.g. areas closed as marine protected areas). Other types of movement information, such as patterns in vertical movements, are also becoming important in stock assessment and in understanding the ecosystem role of different species, as their depth distribution affects which species they interact with, both as predators and as prey (Horodysky et al. 2007).

Both horizontal and vertical movement information about individuals can be gathered through the use of PSATs (Godo & Michalsen 2000, Williams & Lamb 2002, Seitz et al. 2005, Brown et al. 2011). Most positioning studies using archival tag data rely on light-based geolocation methods, sometimes with the addition of sea surface temperature data to improve estimates of location (Welch & Eveson 1999, Teo et al. 2004, Neilsen et al. 2006). The variables recorded by modern PSATs include (but are not limited to) acceleration, depth, light, magnetic field strength and temperature.

Hanchet et al. (2008) interpreted existing fishery and biological data to hypothesise that adult Antarctic toothfish (in the Ross Sea) move from the continental slope region to northern seamounts to spawn, and then return to the slope to feed and regain condition. However, the actual degree of connectivity, range of spawning destinations, level of return migration, and duration of migrations remains uncertain (Parker, Hanchet & Horn 2014). Furthermore, toothfish inhabit deep waters under ice and below the photic zone and long periods of constant daylight or constant darkness preclude the use of light for Antarctic toothfish geolocation year round. Despite these challenges, we suspect that the sensors in PSATs may be able to provide other information that might prove useful in geolocating Antarctic toothfish.

In January 2013, four PSATs were attached to toothfish and released in the Ross Sea (Table 6.3, Figure 6.1). The four fish chosen were large and in excellent condition (Parker, Webber & Arnold 2014). Despite being programmed to pop-up after one year and transmit data via satellite, none of

Table 6.3: Details of four pop-up satellite archival tag (PSAT) releases on Antarctic toothfish (*Dissostichus mawsoni*) in the Ross Sea during January 2013. Columns include a unique tag identification number for each tagged fish, the date of capture/release, the length of each fish, their weight, the depth of water that the fish were caught in, and the latitude and longitude of their capture/release.

Tag ID	Date (dd-mm-yyyy)	Length (cm)	Weight (kg)	Depth caught (m)	Latitude	Longitude
206	22-01-2013	167	51.2	1058	-71.72	176.97
121	22-01-2013	150	39.0	1006	-71.70	176.97
162	23-01-2013	150	39.0	838	-71.80	177.11
179	23-01-2013	160	47.5	914	-71.80	177.17

Figure 6.1: A toothfish being tagged with a pop-up satellite archival tag (PSAT) [left] and a PSAT [right] (photo credit S. Parker, NIWA).

the tags popped off and transmitted. However, one of these fish was recaptured prior to its programmed pop-off date the following fishing season on 24 December 2013, 335 days later (**tag 121**, Figure 6.2). In addition, a fifth tag (**tag 186**) was deployed as a towed body at a depth ranging between about 8m depth and the surface, approximately 100m behind the vessel on a transect from the southern Ross Sea to a latitude of about 60°S (Figure 6.2). A series of global positioning system (GPS) recordings were taken along this transect (Appendix C.1, page 342).

Our aim is to develop a state-space model that can estimate the path taken by a fish between tag-release and tag-recapture location using data recorded by PSATs.

6.1.1 Variables recorded by the tags

Variables recorded by the deployed PSATs included date and time, **acceleration** in three dimensions (3D), **depth** with a resolution of 15m, **light**, **magnetic field strength** in 3D with a resolution of approximately 15 nano Teslas (nT), and **temperature** with a resolution of 0.002°C. Acceleration and magnetic field strength were measured along three orthogonal axes (x, y, z). These correspond to the 3D components of acceleration and the northerly, easterly and vertical components of the Earths magnetic field, respectively. Tag 121 was programmed to record all variables every 10 minutes. Tag 186 recorded every 16 seconds.

6.1.2 Environmental data or models (covariates)

The PSAT data can be compared with values of the same variables from measured data sets or global or regional models. Depth (m) observed by the tag d_t at time t can be compared to a bathymetric map of the Ross Sea region (Figure 6.2, Davey 2004). These data are available in cells measuring 0.01° latitude by 0.01° longitude from -79° to -69° latitude and 160° to 210° longitude (a 1001×5001 matrix of latitude by longitude). We refer to these data the depth covariate layer $\mu^d_{\mathbf{x}}$ where the subscript $\mathbf{x}$ indexes

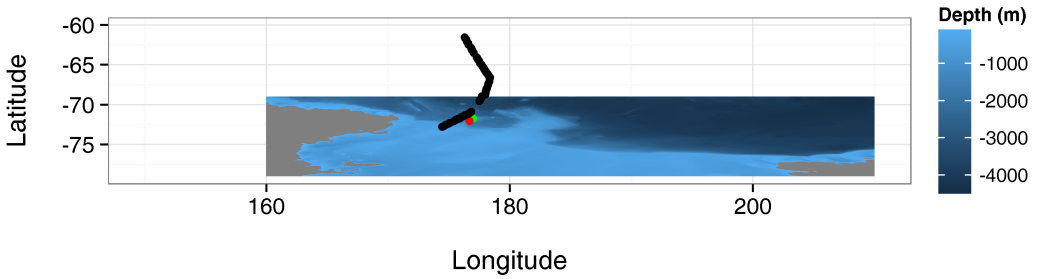


Figure 6.2: Expected depth (m) throughout the Ross Sea region (Davey 2004). The tag-release [•] and recapture [•] locations of the tagged fish (tag 121) are shown, as are the recorded GPS coordinates associated with the towed tag (tag 186) [•]. Regions in grey represent land. The aspect ratio has been chosen so that the plot is approximately equal area projected.

the 2D horizontal location. Any depth below the surface (0m) is given as a negative value (e.g. the depth at (-75.562, 180.021) is -565.0m). Anything above the surface is given a positive value (these depths are actually heights, e.g. the "depth" at (-75.562, 160.021) is 1149.5m which is on land).

Expected temperature (°C) and the standard deviation of temperature (°C) are available from the Commonwealth Scientific and Industrial Research Organisation (CSIRO) Atlas of Regional Seas (CARS) 2009 model (Figures 6.3 and 6.4, Ridgeway et al. 2002, `www.cmar.csiro.au/cars`). This model provides estimates of temperature by latitude, longitude, depth and month. These model outputs are available in cells measuring 0.5° latitude by 0.5° longitude across a variable bin size for depth (ranging from 0 to 5631m). The spatial range of these data extends from -75° to -60° latitude and 150° to 210° longitude (a $121 \times 31 \times 79 \times 12$ array of latitude by longitude by depth by month). The standard deviation of temperature in each cell is provided by longitude, latitude and depth (a $31 \times 121 \times 79$ array of latitude by longitude by depth). We refer to these model outputs as the temperature covariate layer $\mu_{\mathbf{x},d,m}$ and the standard deviation of temperature covariate layer $\sigma_{\mathbf{x},d}$ where the subscript $\mathbf{x}$ indexes the 2D horizontal location, d the depth, and m the month. The temperature can be negative or positive. Some of the values in the array are NaN (not a number),

these refer to latitude/longitude/depth/month combinations that are on land or below the sea floor. The standard deviation of temperature values are all positive besides the values in the array are NaN. We note that the CARS model does not represent estimates of the temperature at the time of tagging. Instead, CARS is a digital climatology, representing the average monthly conditions over the period of modern ocean measurement, and average seasonal cycles for that period. It is derived from all available historical subsurface ocean property measurements - primarily research vessel instrument profiles and autonomous profiling buoys. This is an important limitation on its utility in inferring fish movements, particularly on small spatial scales.

6.1.3 Projection

The start and end locations of the PSATs were recorded in latitude and longitude (WGS84), as were the data on depth and temperature described above. However, the process model (described below in Section 6.2.1, page 220) must operate in equal-area projection. This is because the fish's movement on a 2D horizontal plane is assumed to follow a bivariate normal distribution with a constant variance in physical distance (rather than in latitude/longitude).

The spherical Lambert azimuthal equal-area projection is a mapping from a sphere to a disk (Snyder 1926). It accurately represents area in all regions of the sphere (but it does not accurately represent angles, Figure 6.5). Given the parameters R, ϕ_0, λ_0, ϕ and λ we can use the transformation equations

$$\begin{aligned} x &= Rk' \cos\phi \sin(\lambda - \lambda_0), \\ y &= Rk' (\cos\phi_0 \sin\phi - \sin\phi_0 \cos\phi \cos(\lambda - \lambda_0)), \end{aligned} \tag{6.1}$$

where (x, y) are the equal-area projected latitude and longitude (m), (ϕ, λ) are the latitude and longitude (WGS84, in radians) to be converted, (ϕ_0, λ_0) are the latitude and longitude of the projection center and origin (in radi-

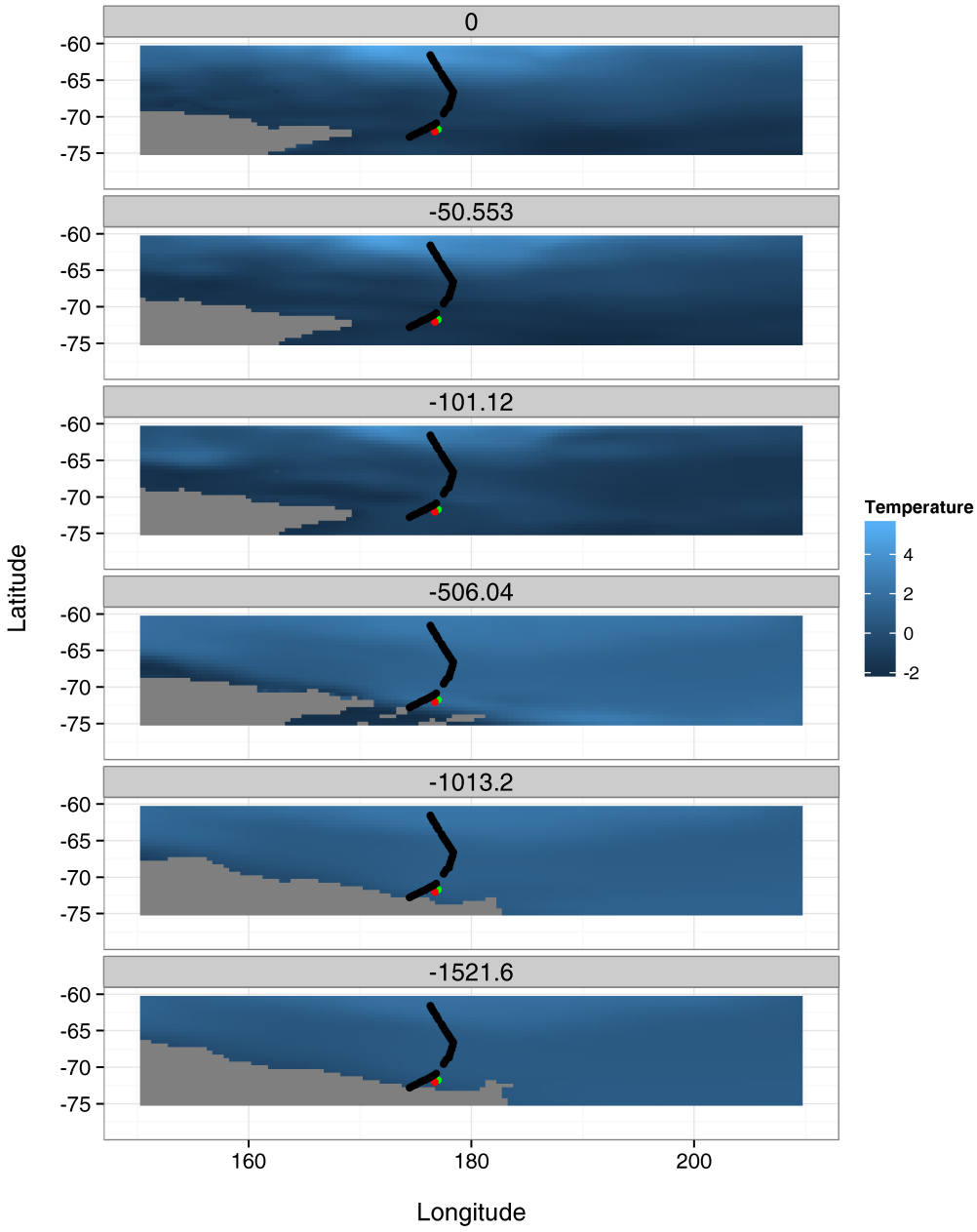


Figure 6.3: Expected temperature (°C) at approximately 0, -50, -100, -500, -1000 and -1500m depth during February. The tag-release [●] and recapture [●] locations of the tagged fish (tag 121) are shown, as are the recorded GPS coordinates associated with the towed tag (tag 186) [●]. Regions in grey represent land or the sea floor. The aspect ratio has been chosen so that the plot is approximately equal area projected.

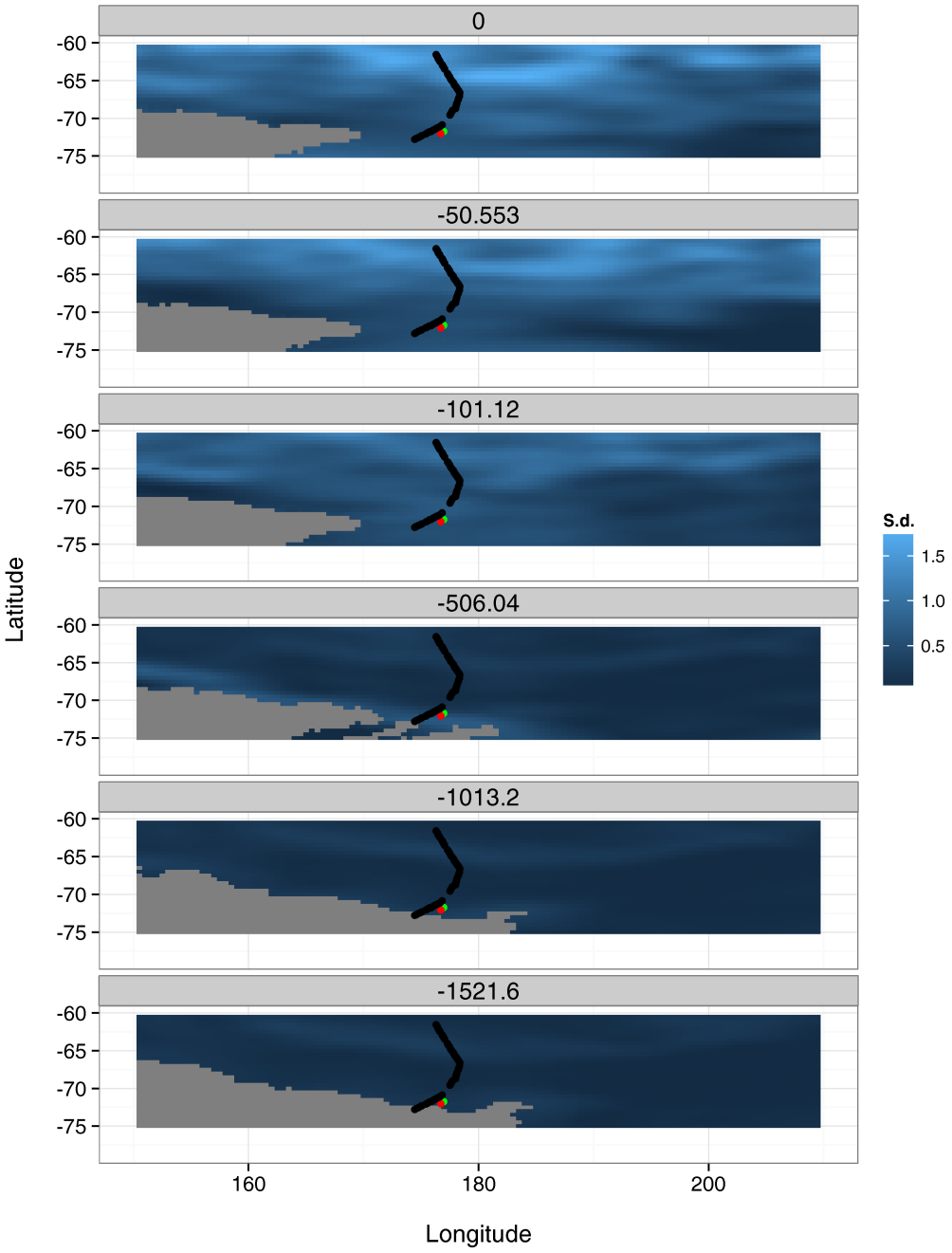


Figure 6.4: Expected standard deviation of temperature (°C) at approximately 0, -50, -100, -500, -1000 and -1500m depth. The tag-release [•] and recapture [•] locations of the tagged fish (tag 121) are shown, as are the recorded GPS coordinates associated with the towed tag (tag 186) [•]. Regions in grey represent land or the sea floor. The aspect ratio has been chosen so that the plot is approximately equal area projected.

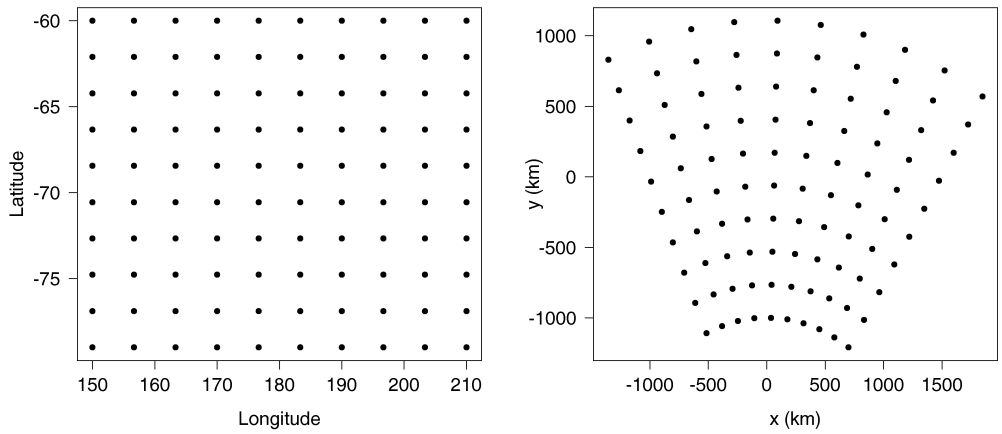


Figure 6.5: A series of points plotted in latitude and longitude [left] and the same points converted to Lambert azimuthal equal-area projection in km [right].

ans), R is the radius of the earth (m, Equation 6.3), and

$$k' = \sqrt{\frac{2}{1 + \sin\phi_0 \sin\phi + \cos\phi_0 \cos\phi \cos(\lambda - \lambda_0)}}.$$

The inverse formulas are

$$\begin{aligned} \phi &= \sin^{-1}\left(\cos c \sin\phi_0 + \frac{y \sin c \cos\phi_0}{\rho}\right), \\ \lambda &= \lambda_0 + \tan^{-1}\left(\frac{x \sin c}{\rho \cos\phi_0 \cos c - y \sin\phi_0 \sin c}\right), \end{aligned} \tag{6.2}$$

where

$$\rho = \sqrt{x^2 + y^2} \quad \text{and} \quad c = 2\sin^{-1}\left(\frac{1}{2}\rho\right).$$

These equations allow us to transform any estimated location of a fish from equal-area projection $\mathbf{x}_t$ (in the process equation of the state-space model) to latitude and longitude (the Bathymetric and temperature data are in this form), or vice versa. We set $\phi_0 = -70^\circ$ and $\lambda_0 = 175^\circ$ so that our projection centre and origin are close to the tagged fish and the towed tag to give an accurate projection.

The geocentric radius of the Earth R (distance from the Earth's center to a point on the spheroid) given the geodetic latitude ϕ is

$$R = \sqrt{\frac{\left(a^2 \cos\phi\right)^2 + \left(b^2 \sin\phi\right)^2}{\left(a\cos\phi\right)^2 + \left(b\sin\phi\right)^2}}, \tag{6.3}$$

where a is the equatorial radius (6378.1370km) and b is the polar radius (6356.7523km). Given $\phi_0 = -70^\circ$ we calculate $R = 6359272.44$m.

6.2 Model development

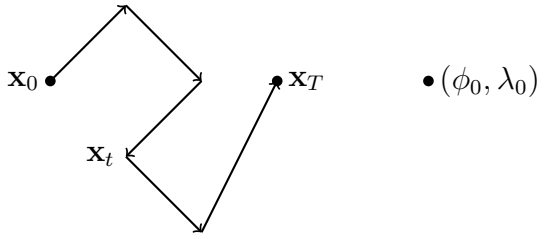
State-space models (SSMs) can be used to estimate the latent state of unobservable processes from observed data sets. There are two main components to any SSM: the **process model** which describes how the states evolve in time, and the **observation model**(s) that describe how the states are linked to observations. Here the latent states are the geographical locations of the fish between release and recapture (equal-area projected latitude and longitude). The observed data set includes measures of depth, temperature, light, magnetic field strength and acceleration collected at regular intervals by the tag.

The model is written in Julia (`http://julialang.org/`), a high-level, high-performance dynamic programming language for technical computing. Unlike languages such as R, Julia code is compiled before execution, which is the main reason for its superior speed. Moreover, Julia makes parallel computing easy, therefore our MCMC algorithm is multi-threaded which greatly reduces computational time. The model is written so that different components of the model can be turned on and off easily. This allows us to turn off or modify the prior, process model, or any of the observation models independently.

6.2.1 The process model

The process model describes how an individual fish moves through **continuous space** (equal-area horizontal) and **discrete time**. It does not model

the vertical dynamics of the fish as the depth is recorded by the tag (and assumed to be known without error). The parameters of the model are the standard deviation of the fish's horizontal movement $\sigma_{\mathbf{x}}$ (m) and the discrete hidden location latent states $\mathbf{x}_1, \ldots, \mathbf{x}_{T-1}$ (an equal-area projected latitude/longitude pair at each time-step t with units m). The start location ($\mathbf{x}_0$) and the end location ($\mathbf{x}_T$) are assumed to be known without error (i.e. these points are fixed). These points and all latent states ($\mathbf{x}_t$) are defined as the distance (m) from the specified reference point (ϕ_0, λ_0), schematically:



For simplicity, we begin by describing the model in one dimensional (1D) space. The notation is changed to two dimensional (2D) space later. In 1D, the full probability of a path $\{x_t\}_{t=0}^{T}$ is

$$P(x_0, ..., x_T) = P(x_T|x_{T-1}) \times \cdots \times P(x_1|x_0)P(x_0).$$

The probability of a path conditional on the start point (x_0) and the end point (x_T) is

$$\begin{aligned} P(x_1, ..., x_{T-1}|x_0, x_T) &= \frac{P(x_0, \cdots, x_T)}{P(x_0, x_T)} = \frac{P(x_0, \cdots, x_T)}{P(x_T|x_0)P(x_0)} \\ &= \frac{1}{P(x_T|x_0)} P(x_T|x_{T-1}) \times \cdots \times P(x_1|x_0) \\ &\propto P(x_T|x_{T-1}) \times \cdots \times P(x_1|x_0) \end{aligned}$$

where the proportionality neglects any terms not explicitly dependent on $x_1, \ldots, x_{T-1}$. The probability of any point along a path (x_t) conditional on the previous point (x_{t-1}) and the next point (x_{t+1}) is

$$\begin{aligned} P(x_t|x_{t-1}, x_{t+1}) &= \frac{P(x_{t-1}, x_t, x_{t+1})}{P(x_{t-1}, x_{t+1})} = \frac{P(x_{t+1}|x_t)P(x_t|x_{t-1})P(x_{t-1})}{P(x_{t-1}, x_{t+1})} \\ &\propto P(x_{t+1}|x_t)P(x_t|x_{t-1}). \end{aligned}$$

Assuming a normally distributed random walk with standard deviation σ_x we write

$$x_t|x_{t-1}, \sigma_x \sim \mathcal{N}\left(x_{t-1}, \sigma_x^2\right).$$

Conditional on both the previous point (x_{t-1}) and the next point (x_{t+1}) this becomes

$$\begin{aligned}
P(x_t|x_{t-1}, x_{t+1}, \sigma_x) &\propto \left(2\pi\sigma_x^2\right)^{-\frac{1}{2}} \exp\left[-\frac{1}{2\sigma_x^2}\left(x_{t+1} - x_t\right)^2\right] \times \left(2\pi\sigma_x^2\right)^{-\frac{1}{2}} \exp\left[-\frac{1}{2\sigma_x^2}\left(x_t - x_{t-1}\right)^2\right] \\
&\propto \exp\left[-\frac{1}{2\sigma_x^2}\left(2x_t^2 - 2x_t x_{t+1} - 2x_t x_{t-1}\right)\right] \\
&\propto \exp\left[-\frac{1}{2\sigma_x^2} 2\left(x_t^2 - 2x_t\left(\frac{x_{t+1} + x_{t-1}}{2}\right)\right)\right] \\
&\propto \exp\left[-\frac{1}{2\left(\frac{\sigma_x^2}{2}\right)}\left(x_t - \frac{x_{t+1} + x_{t-1}}{2}\right)^2\right] \\
x_t|x_{t-1}, x_{t+1}, \sigma_x &\sim \mathcal{N}\left(\frac{1}{2}(x_{t-1} + x_{t+1}), \frac{\sigma_x^2}{2}\right).
\end{aligned}$$

In 2D this simply becomes

$$\mathbf{x}_t|\mathbf{x}_{t-1}, \sigma_{\mathbf{x}} \sim \mathcal{N}\left(\mathbf{x}_{t-1}, \sigma_{\mathbf{x}}^2\mathbf{I}\right), \tag{6.4}$$

$$\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_{t+1}, \sigma_{\mathbf{x}} \sim \mathcal{N}\left(\frac{1}{2}\left(\mathbf{x}_{t-1} + \mathbf{x}_{t+1}\right), \frac{\sigma_{\mathbf{x}}^2}{2}\mathbf{I}\right), \tag{6.5}$$

where $\mathbf{I}$ is a 2×2 identity matrix. We can generalise this further for any point $\mathbf{x}_j$ to any other point $\mathbf{x}_k$

$$\mathbf{x}_t|\mathbf{x}_{t-j}, \mathbf{x}_{t+k}, \sigma_{\mathbf{x}} \sim \mathcal{N}\left(\frac{1}{j+k}\left(k\mathbf{x}_{t-j} + j\mathbf{x}_{t+k}\right), \frac{jk\sigma_{\mathbf{x}}^2}{j+k}\mathbf{I}\right).$$

Conditional only on $\mathbf{x}_0$ and $\mathbf{x}_T$ we can sample from Equation 6.5 using a Gibbs sampler. The expected path of $\mathbf{x}_t$ is a straight line

$$\mathbb{E}[\mathbf{x}_t|\mathbf{x}_0, \mathbf{x}_T] = \mathbf{x}_0 + \frac{t}{T}\left(\mathbf{x}_T - \mathbf{x}_0\right). \tag{6.6}$$

Unlike horizontal movement, we place no constraint on the vertical movement of the fish (i.e. d_t is assumed to be independent of d_{t-1}).

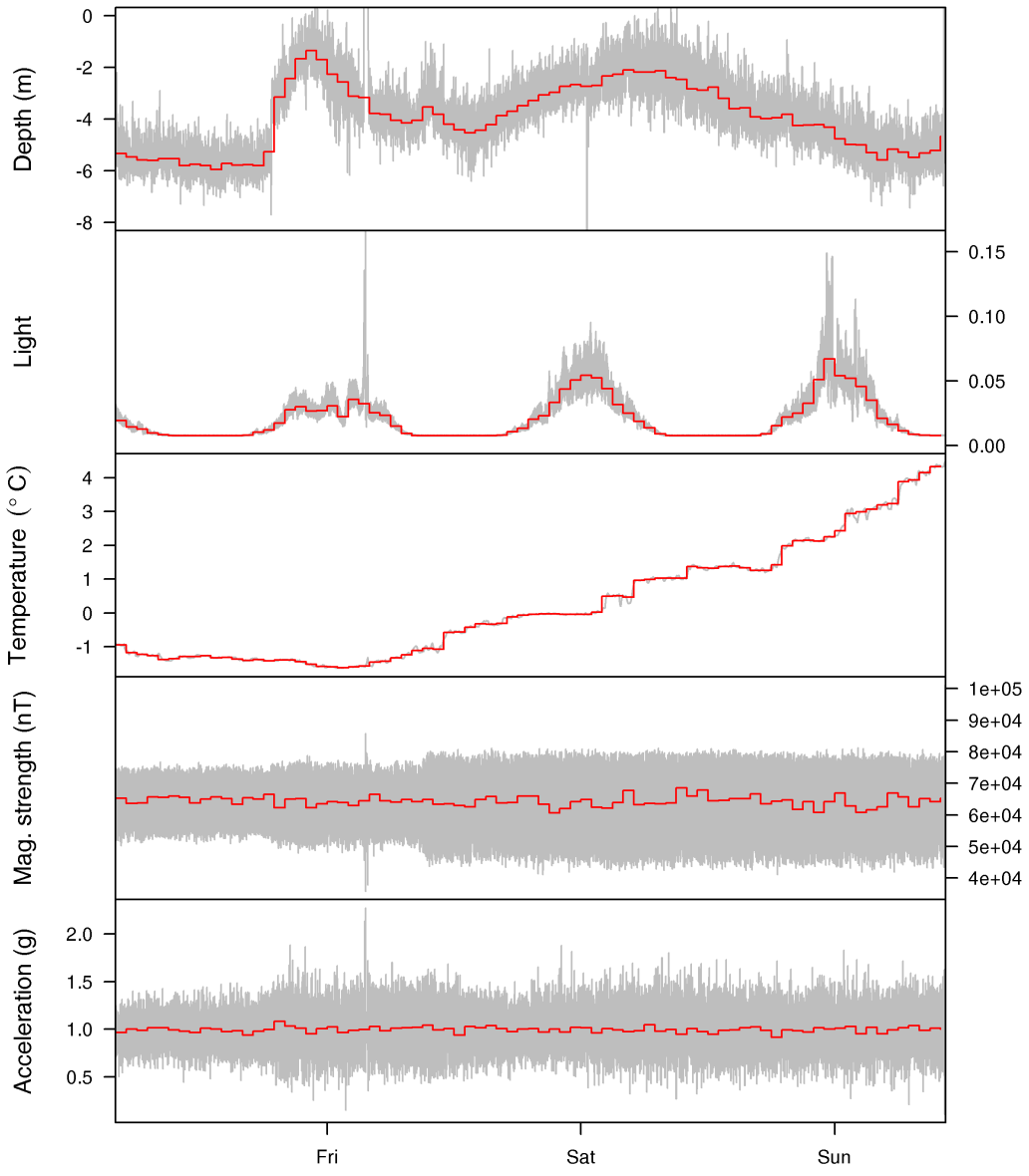


Figure 6.6: Observations recorded by the towed tag (**tag 186**) every 16 seconds [grey] and hourly median [red] from 23 February 2012 to 26 February 2012.

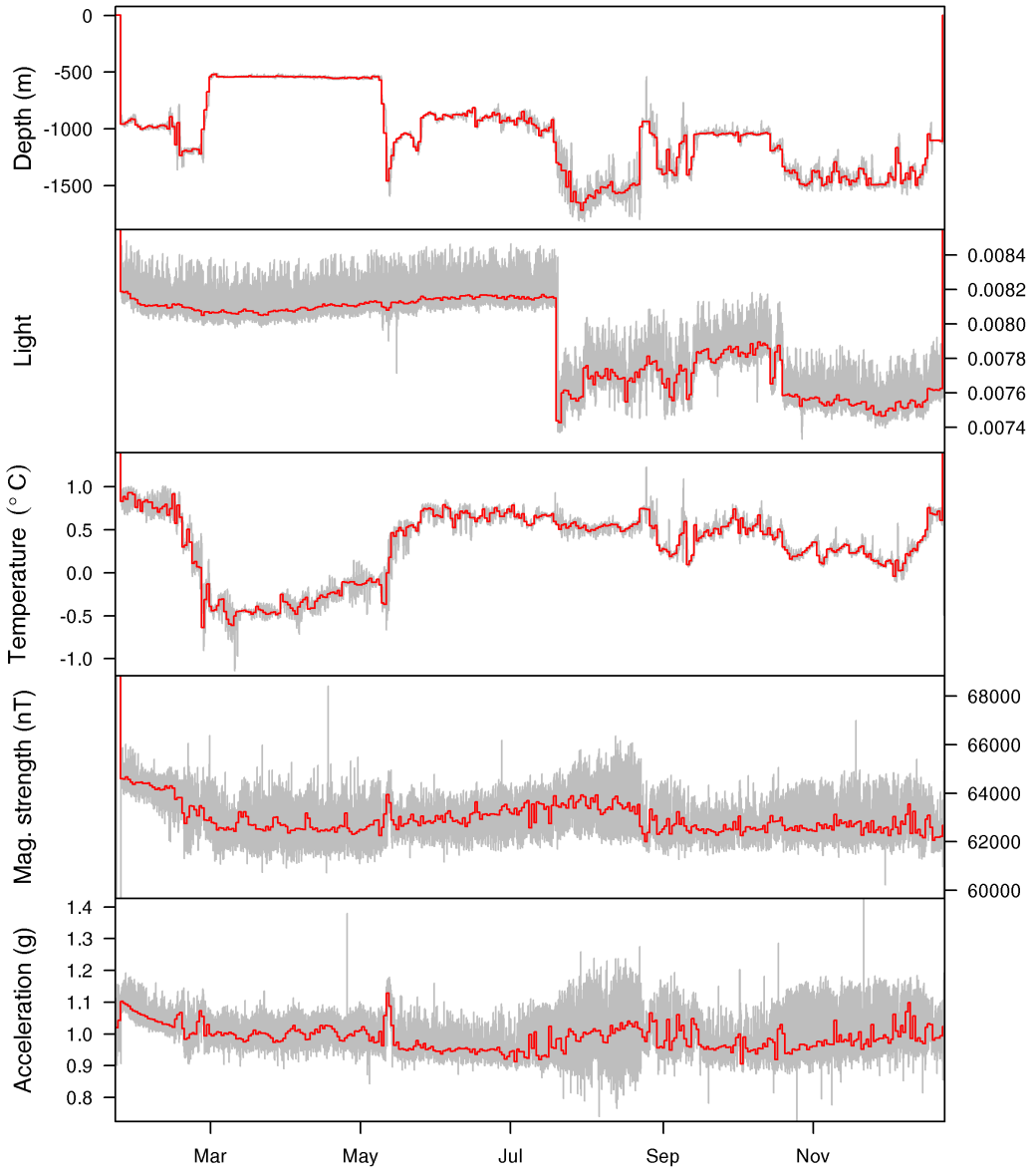


Figure 6.7: Observations recorded by the tagged fish (**tag 121**) every 10 minutes [grey] and daily median [red].

6.2.2 Observation models

Data for the towed tag (tag 186) and the tagged fish (tag 121) are shown in Figures 6.6 and 6.7. Observation models were developed to model the recorded variables **depth** and **temperature**. Although acceleration, light, and magnetic field strength were recorded by the tag, they are not used here.

Light has been used successfully in the past to help estimate a fish's location through time (Welch & Eveson 1999). However, toothfish inhabit deep waters under ice and below the photic zone. In addition, long periods of constant daylight or constant darkness preclude the use of light for Antarctic toothfish geolocation year round.

Models are available that describe the total magnetic field strength both spatially and temporally. However, preliminary exploration of the data suggested that the variability in the magnetic field strength recorded by the tags drowned out any pattern in these data across their spatial range (Figures 6.6 and 6.7). We simply plot the magnitude of total magnetic field strength.

We do not attempt to model acceleration in this project. However, we do provide some ideas for future research in the discussion (Section 6.7, page 253) and plot the magnitude of acceleration.

Temperature

The temperature observation model links the temperature observed by the tag c_t to the temperature covariate layer $\mu^c_{\mathbf{x},d,m}$ and the standard deviation of temperature covariate layer $\sigma^c_{\mathbf{x},d}$ (or the fixed standard deviation of temperature σ^c if this is being used). This model consists of a lookup function coupled with a likelihood function.

The lookup function is written to access the temperature and standard deviation of temperature arrays given a latitude, longitude, depth and month. If the given latitude and/or longitude exceeds the bounds of the temperature layer, then the function returns the value NaN (not a num-

ber). Otherwise it returns the temperature at that location and month along with the standard deviation of temperature in that location. The returned values could be NaN (see Section 6.1.2, page 216).

The temperature given the horizontal location of the tag $\mathbf{x}_t$ (latitude and longitude), depth (m) of the tag d_t, and the month m_t at time $t = 1, \ldots, T-1$ is is assumed to be

$$c_t|\mathbf{x}_t, d_t, m_t, \mu^c_{\mathbf{x},d,m}, \sigma^c_{\mathbf{x},d} \sim \mathcal{N}\left(\mu^c_{\mathbf{x}_t,d_t,m_t}, \left(\sigma^c_{\mathbf{x}_t,d_t}\right)^2\right) \tag{6.7}$$

or

$$c_t|\mathbf{x}_t, d_t, m_t, \mu^c_{\mathbf{x},d,m}, \sigma^c \sim \mathcal{N}\left(\mu^c_{\mathbf{x}_t,d_t,m_t}, (\sigma^c)^2\right) \tag{6.8}$$

depending on whether or not the spatially varying standard deviation of temperature covariate layer $\sigma^c_{\mathbf{x},d}$ is to be used or not (the alternative being a fixed standard deviation σ^c). The log-likelihood function of the temperature is

$$\ell\left(c_t|\mathbf{x}_t, d_t, m_t, \mu^c_{\mathbf{x},d,m}, \sigma^c_{\mathbf{x},d}\right) = \begin{cases} -\infty & \text{if } \mu^c_{\mathbf{x}_t,d_t,m_t} = \text{NaN} \\ -\log\left(\sigma^c_{\mathbf{x}_t,d_t}\right) - \frac{\left(c_t - \mu^c_{\mathbf{x}_t,d_t,m_t}\right)^2}{2\left(\sigma^c_{\mathbf{x}_t,d_t}\right)^2} & \text{otherwise} \end{cases}. \tag{6.9}$$

If the lookup function returns NaN, then the log-likelihood evaluates to $-\infty$ (i.e. the path is limited to the range of the temperature data).

Depth

A lookup function is written to access the depth covariate layer matrix (Section 6.1.2, page215) given a latitude and longitude. If the given location exceeds the bounds of the depth layer, then the function returns NaN (not a number). Otherwise it returns the depth at that location (even if the depth is > 0).

The depth data may be used in two ways: using depth to simply exclude certain areas of 2D space (by identifying land as inaccessible to the fish), or by assuming that the fish follows the sea floor and developing a distribution to describe the fish's behaviour with respect to the sea floor.

We use the first option to model the towed tag (tag 186, Section 6.4, page 233) and in a simulation study (Section 6.5, page 240). If using this option the log-likelihood function of the depth is

$$\ell\left(d_t|\mathbf{x}_t,\mu^d_{\mathbf{x}}\right) = \begin{cases} -\infty & \text{if } d_t < \mu^d_{\mathbf{x}_t} \\ 0 & \text{otherwise} \end{cases}. \tag{6.10}$$

The choice of a constant log-likelihood of 0 here is arbitrary and any constant value could be used. This function stops the model from exploring space where there is land or where the sea floor is shallower than the depth of the fish. It does not stop the model from exploring space beyond the boundaries of the data (i.e. when the lookup function returns NaN). Also, unlike horizontal movement, we place no constraints on the vertical position of the fish relative to the sea floor. To explain, we use Figure 6.8 below. Let us assume that each of the red points in the figure represent the

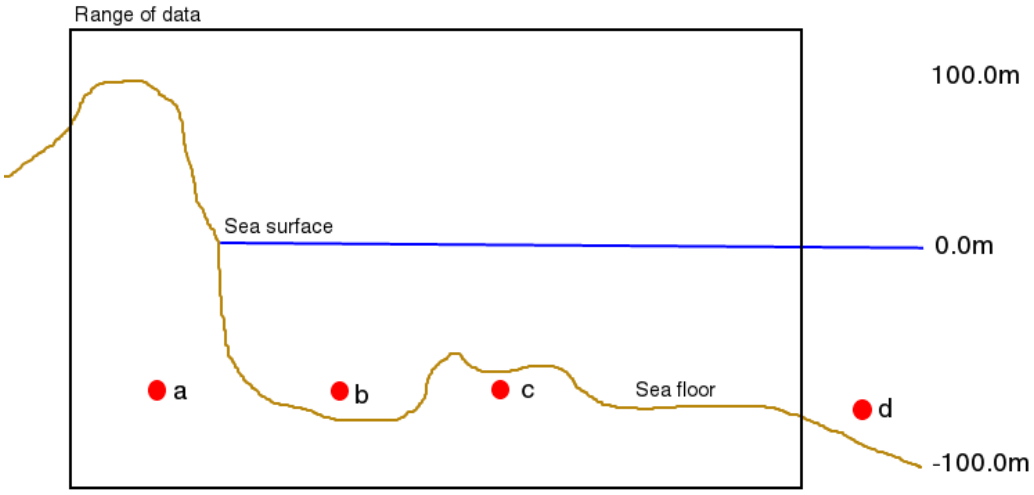


Figure 6.8: An example diagram of the sea floor, sea surface (0m), and the range of some depth data (e.g. $\mu^d_{\mathbf{x}}$). Proposed locations for model exploration are shown using the [•] points **a**, **b**, **c** and **d**.

proposed location of our fish within a time-step in our MCMC. Point **a** is within the range of the depth covariate layer, but the proposed location is on land (actually underground, $d_t < \mu^d_{\mathbf{x}}$). In this case, the log-likelihood

would evaluate to $-\infty$ and the move would be rejected. Point **b** is within the range of the depth covariate layer, and above the sea floor ($d_t \geq \mu^d_{\mathbf{x}}$), therefore the log-likelihood evaluates to 0 and the proposed move could be accepted. Point **c** is within the range of the depth covariate layer, but the proposed location is deeper than the expected depth in this location ($d_t < \mu^d_{\mathbf{x}}$). In this case, the log-likelihood would evaluate to $-\infty$ and the move would be rejected. Finally, point **d** exceeds the range of the depth covariate layer. In this case, we don't know if the proposed point is above or below the sea floor or over land. Here we assume that the move is acceptable and set the log-likelihood to 0 (but see Section 6.2.3, page 229 below).

When modelling the tagged fish (tag 121) we used a split-normal distribution to link the known depth (m) of the fish to the depth throughout the Ross Sea (Figure 6.9). In the first instance, we assume the fish will gener-

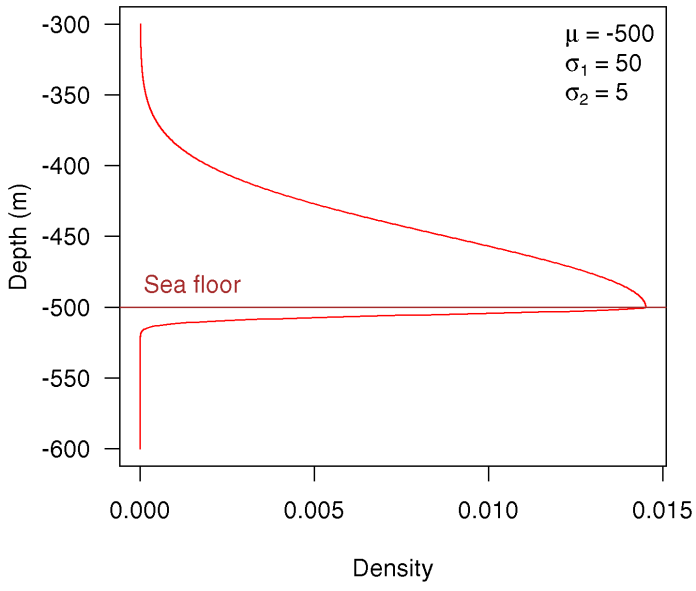


Figure 6.9: The assumed prior of the distribution of tagged fish (**tag 121**) recorded depth relative to bathymetric data (shown for example at 500m with $\sigma_1 = 50$m and $\sigma_2 = 5$m).

ally follow the sea floor. The split-normal also provides some probability density a short distance below the sea floor to account for any uncertainty in the measured depth throughout the region. In this way, we are using the split-normal distribution to "stick" the fish to the sea floor, but allow the fish to go little deeper than the bathymetric data suggests the depth is, and allow the fish to explore space above the sea floor as well. We compare the median depth observed by the tag d_t during time t to the depth expected in that location

$$d_t|\mathbf{x}_t, \mu^d_{\mathbf{x}}, \boldsymbol{\sigma}^d \sim \mathcal{SN}\left(\mu^d_{\mathbf{x}}, (\boldsymbol{\sigma}^d)^2\right), \tag{6.11}$$

where $\boldsymbol{\sigma}^d = \{\sigma^d_1, \sigma^d_2\}$. The log-likelihood function for depth is

$$\ell\left(d_t|\mathbf{x}_t, \mu^d_{\mathbf{x}}, \boldsymbol{\sigma}^d\right) = \log \mathcal{SN}\left(\mu^d_{\mathbf{x}}, (\boldsymbol{\sigma}^d)^2\right). \tag{6.12}$$

The split-normal distribution will only be an appropriate choice for species that tend to follow the sea-floor. If modelling pelagic species then the method that excludes areas would be a better choice (Equation 6.10).

The split-normal distribution is formed by merging two normal distributions about a common mode. The probability density function (PDF) of a split-normal distribution is given by

$$f(x; \mu, \sigma_1, \sigma_2) = \begin{cases} A \exp\left[-\frac{(x-\mu)^2}{2\sigma_1^2}\right] & \text{if } x < \mu \\ A \exp\left[-\frac{(x-\mu)^2}{2\sigma_2^2}\right] & \text{otherwise,} \end{cases}$$

where

$$A = \sqrt{\frac{2}{\pi}} \left(\sigma_1 + \sigma_2\right)^{-1}.$$

The component distributions of the split-normal distribution can have two different variances. To ensure that the resulting PDF integrates to 1, the normalising constant A is used.

6.2.3 Additional considerations

To stop the fish from exceeding the bounds of any, or both, covariate layers we can use uniform priors. These priors could be applied to the equal-area

projected proposals (x, y) or the proposals in latitude and longitude (ϕ, λ). For example, if we wanted to restrict the path to being within the range of the temperature covariate layer we would use

$$\pi(\phi) \sim \mathcal{U}(-75, -60),$$
$$\pi(\lambda) \sim \mathcal{U}(150, 210)$$

noting that this is not uniform in (x, y).

In summary, we make the following modelling assumptions:

- all data measured by the tag (a_t, b_t, c_t, d_t, m_t) are measured without error
- the start location ($\mathbf{x}_0$) and end location ($\mathbf{x}_T$) are fixed and known without error
- the standard deviation parameter $\sigma_{\mathbf{x}}$ is constant over time t
- the fish does not leave the bounds of the temperature covariate layer in the towed tag (tag 186) and simulation models
- the fish does not leave the bounds of the depth or temperature covariate layers in the tagged fish (tag 121) model
- if using the split-normal distribution when modelling depth, the parameter $\boldsymbol{\sigma}^d$ is assumed known and is $\boldsymbol{\sigma}^d = \{50, 5\}$
- if using a constant (with location and depth) standard deviation of temperature, the value is fixed at $\sigma^c = 0.1°\text{C}$

The assumed parameters of the split-normal distribution ($\boldsymbol{\sigma}^d = \{50, 5\}$) were chosen as they represent reasonable guesses at how far above the sea floor a toothfish might venture (i.e. standard deviation of 50m) and how uncertain our bathymetric data are (standard deviation of 5m).

6.3 Bayesian inference

We are interested in the probabilistic relationship between the following:

- **The data:** the median observed temperature (c_t) and median observed depth (d_t) during time t. Let $\mathbf{y} = \{c_t, d_t\}_{t=1}^{T-1}$
- **The covariates:** the month (m_t) during time t, the expected temperature ($\mu^c_{\mathbf{x},d,m}$) in location $\mathbf{x}$, at depth d during month m, the standard deviation of temperature ($\sigma^c_{\mathbf{x},d}$) in location $\mathbf{x}$ at depth d, and the expected depth ($\mu^d_{\mathbf{x}}$) in location $\mathbf{x}$. Let $\mathbf{z} = \{\{m_t\}_{t=1}^{T-1}, \mu^c_{\mathbf{x},d,m}, \sigma^c_{\mathbf{x},d}, \mu^d_{\mathbf{x}}\}$
- **The unknown nuisance parameter:** the standard deviation of horizontal movement ($\sigma_{\mathbf{x}}$)
- **The known parameters:** the start ($\mathbf{x}_0$) and end ($\mathbf{x}_T$) locations and the and the split-normal standard deviation of depth ($\boldsymbol{\sigma}^d$). Let $\boldsymbol{\omega} = \{\mathbf{x}_0, \mathbf{x}_T, \boldsymbol{\sigma}^d\}$ (and possibly also σ^c)
- **The unknown latent states:** the horizontal location of the fish ($\mathbf{x}_t$) at time t. Let $\mathbf{x} = \{\mathbf{x}_t\}_{t=1}^{T-1}$

Using Bayes theorem, the posterior distribution of the model parameters (ϕ) and the latent states ($\mathbf{x}$), given the data ($\mathbf{y}$), covariates ($\mathbf{z}$) and known parameters ($\boldsymbol{\omega}$) is

$$\pi(\sigma_{\mathbf{x}}, \mathbf{x}|\mathbf{y}, \mathbf{z}, \boldsymbol{\omega}) \propto \pi(\sigma_{\mathbf{x}})\pi(\mathbf{x}|\boldsymbol{\omega}, \sigma_{\mathbf{x}})\pi(\mathbf{y}|\mathbf{x}, \mathbf{z}, \boldsymbol{\omega}), \tag{6.13}$$

where for any t

$$\begin{aligned}
\pi(\mathbf{x}|\boldsymbol{\omega}, \sigma_{\mathbf{x}}) &= \pi(x_t|x_{t-1}, x_{t+1}, \sigma_{\mathbf{x}}) \quad \text{see Equation 6.5,} \\
\pi(\mathbf{y}|\mathbf{x}, \mathbf{z}, \boldsymbol{\omega}) &= \pi(\mathbf{y}|\mathbf{x}, \mathbf{z}, \boldsymbol{\sigma}^d) \\
&= \prod_{t=1}^{T-1} \pi(c_t, d_t|\mathbf{x}_t, d_t, m_t, \mu^c_{\mathbf{x},d,m}, \sigma^c_{\mathbf{x},d}, \mu^d_{\mathbf{x}}, \boldsymbol{\sigma}^d) \\
&= \prod_{t=1}^{T-1} \pi(c_t|\mathbf{x}_t, d_t, m_t, \mu^c_{\mathbf{x},d,m}, \sigma^c_{\mathbf{x},d}) \times \prod_{t=1}^{T-1} \pi(d_t|\mathbf{x}_t, \mu^d_{\mathbf{x}}, \boldsymbol{\sigma}^d) \\
&\qquad \text{see Equations 6.7 and 6.11.}
\end{aligned}$$

6.3.1 Blockwise Metropolis-Hastings algorithm

Posterior distributions of the model parameters and states are estimated using MCMC. We draw samples from the joint posterior distribution using a blockwise Metropolis-Hastings algorithm with a log-normal proposal for the model's parameter ($\sigma_{\mathbf{x}}$) and bivariate normal proposals for the latent states ($\{\mathbf{x}_t\}_{t=1}^{T-1}$, see Appendix A.3, page 325, for a description of a generalised sampler). We begin by initialising $\sigma_{\mathbf{x}}^{(0)} \sim \pi(\sigma_{\mathbf{x}})$ and $\mathbf{x}_t^{(0)}$ (simply a straight line between $\mathbf{x}_0$ and $\mathbf{x}_T$, see Equation 6.6). We specify the standard deviation of the proposal distribution σ_q, and then begin sampling for $i = 1, 2, ...$

1. Propose $\sigma_{\mathbf{x}}^* \sim q_\sigma\left(\sigma_{\mathbf{x}}^{(i)} | \sigma_{\mathbf{x}}^{(i-1)}\right)$. Here we draw $\log(\sigma_{\mathbf{x}}^*) \sim \mathcal{N}\left(\log\left(\sigma_{\mathbf{x}}^{(i-1)}\right), \sigma_q^2\right)$.

2. Compute the acceptance probability r_σ (see Equation 6.14).

3. Draw $u \sim \mathcal{U}(0,1)$.

4. Accept $\sigma_{\mathbf{x}}^*$ if $u < r_\sigma$ and set $\sigma_{\mathbf{x}}^{(i)} \leftarrow \sigma_{\mathbf{x}}^*$, otherwise reject $\sigma_{\mathbf{x}}^*$ and set $\sigma_{\mathbf{x}}^{(i)} \leftarrow \sigma_{\mathbf{x}}^{(i-1)}$.

5. For $t = 2, ..., T-1$

   (a) Propose $\mathbf{x}_t^* \sim q_{\mathbf{x}}\left(\mathbf{x}_t^{(i)} | \mathbf{x}_{t-1}^{(i-1)}, \mathbf{x}_{t+1}^{(i-1)}, \sigma_{\mathbf{x}}^{(i)}\right)$.
   Here we draw a new proposal from a multivariate normal $\mathbf{x}_t^* | \mathbf{x}_{t-1}^{(i-1)}, \mathbf{x}_{t+1}^{(i-1)}, \sigma_{\mathbf{x}}^{(i)} \sim \mathcal{N}\left(\frac{1}{2}\left(\mathbf{x}_{t-1}^{(i-1)} + \mathbf{x}_{t+1}^{(i-1)}\right), \frac{\left(\sigma_{\mathbf{x}}^{(i)}\right)^2}{2}\mathbf{I}\right)$.

   (b) Compute the acceptance probability $r_{\mathbf{x}}$ (see Equation 6.15).

   (c) Draw $u \sim \mathcal{U}(0,1)$.

   (d) Accept $\mathbf{x}_t^*$ if $u < r_{\mathbf{x}}$ and set $\mathbf{x}_t^{(i)} \leftarrow \mathbf{x}_t^*$, otherwise reject $\mathbf{x}_t^*$ and set $\mathbf{x}_t^{(i)} \leftarrow \mathbf{x}_t^{(i-1)}$.

The acceptance probabilities (r_σ and $r_{\mathbf{x}}$) are defined as

$$\begin{aligned} r_\sigma &= \min\left(1, \frac{\pi\left(\mathbf{x}|\boldsymbol{\omega}, \sigma_{\mathbf{x}}^*\right)}{\pi\left(\mathbf{x}|\boldsymbol{\omega}, \sigma_{\mathbf{x}}^{(i-1)}\right)} \times \frac{\pi\left(\sigma_{\mathbf{x}}^*\right)}{\pi\left(\sigma_{\mathbf{x}}^{(i-1)}\right)} \times \frac{q_\sigma\left(\sigma_{\mathbf{x}}^{(i-1)}|\sigma_{\mathbf{x}}^*\right)}{q_\sigma\left(\sigma_{\mathbf{x}}^*|\sigma_{\mathbf{x}}^{(i-1)}\right)}\right) \\ &= \min\left(1, \frac{\pi\left(\mathbf{x}|\boldsymbol{\omega}, \sigma_{\mathbf{x}}^*\right)}{\pi\left(\mathbf{x}|\boldsymbol{\omega}, \sigma_{\mathbf{x}}^{(i-1)}\right)} \times \frac{\pi\left(\sigma_{\mathbf{x}}^*\right)}{\pi\left(\sigma_{\mathbf{x}}^{(i-1)}\right)} \times \frac{\sigma_{\mathbf{x}}^*}{\sigma_{\mathbf{x}}^{(i-1)}}\right), \end{aligned} \tag{6.14}$$

and

$$\begin{aligned} r_{\mathbf{x}} = \min\Bigg(1, &\frac{\pi\left(\mathbf{y}_t|\mathbf{x}_t^*, \mathbf{z}, \boldsymbol{\omega}\right)}{\pi\left(\mathbf{y}_t|\mathbf{x}_t^{(i-1)}, \mathbf{z}, \boldsymbol{\omega}\right)} \times \frac{\pi\left(\mathbf{x}_t^*|\mathbf{x}_{t-1}^{(i-1)}, \mathbf{x}_{t+1}^{(i-1)}, \boldsymbol{\omega}, \sigma_{\mathbf{x}}^{(i)}\right)}{\pi\left(\mathbf{x}_t^{(i-1)}|\mathbf{x}_{t-1}^{(i)}, \mathbf{x}_{t+1}^{(i)}, \boldsymbol{\omega}, \sigma_{\mathbf{x}}^{(i)}\right)} \\ &\times \frac{q_{\mathbf{x}}\left(\mathbf{x}_t^{(i-1)}|\mathbf{x}_{t-1}^*, \mathbf{x}_{t+1}^*, \sigma_{\mathbf{x}}^{(i)}\right)}{q_{\mathbf{x}}\left(\mathbf{x}_t^*|\mathbf{x}_{t-1}^{(i-1)}, \mathbf{x}_{t+1}^{(i-1)}, \sigma_{\mathbf{x}}^{(i)}\right)}\Bigg). \end{aligned} \tag{6.15}$$

The calculation of r_σ requires the evaluation of the likelihood along the whole path (for $t = 1, \ldots, T-1$). The calculation of $r_{\mathbf{x}}$ only requires evaluation of the likelihood for a single location (i.e. a time-step t).

All MCMC simulations consisted of a burn-in of 55000 iterations, followed by 1 million iterations with a thinning rate of 500 resulting in a sample of 2000. Parallel tempering was used as the depth and temperature likelihood surfaces are very complex and multi-modal (imagine the bottom of the sea floor as a likelihood surface). Three different tempered chains were used throughout ($\beta = \{1.0, 0.9, 0.8\}$, see Chapter 2, page 68 for an explanation of parallel tempering).

6.4 Tag 186: the towed tag

Tag 186 was deployed as a towed body approximately 100m behind a vessel on a transect though the Ross Sea (Figure 6.10). While being towed the tag recorded environmental variables every 16 seconds (Figure 6.6). These data show that the tag was between 8m depth and the surface along the transect. There is a clear pattern in the light level throughout each day. As the tag is towed north the temperature gradually increases from below

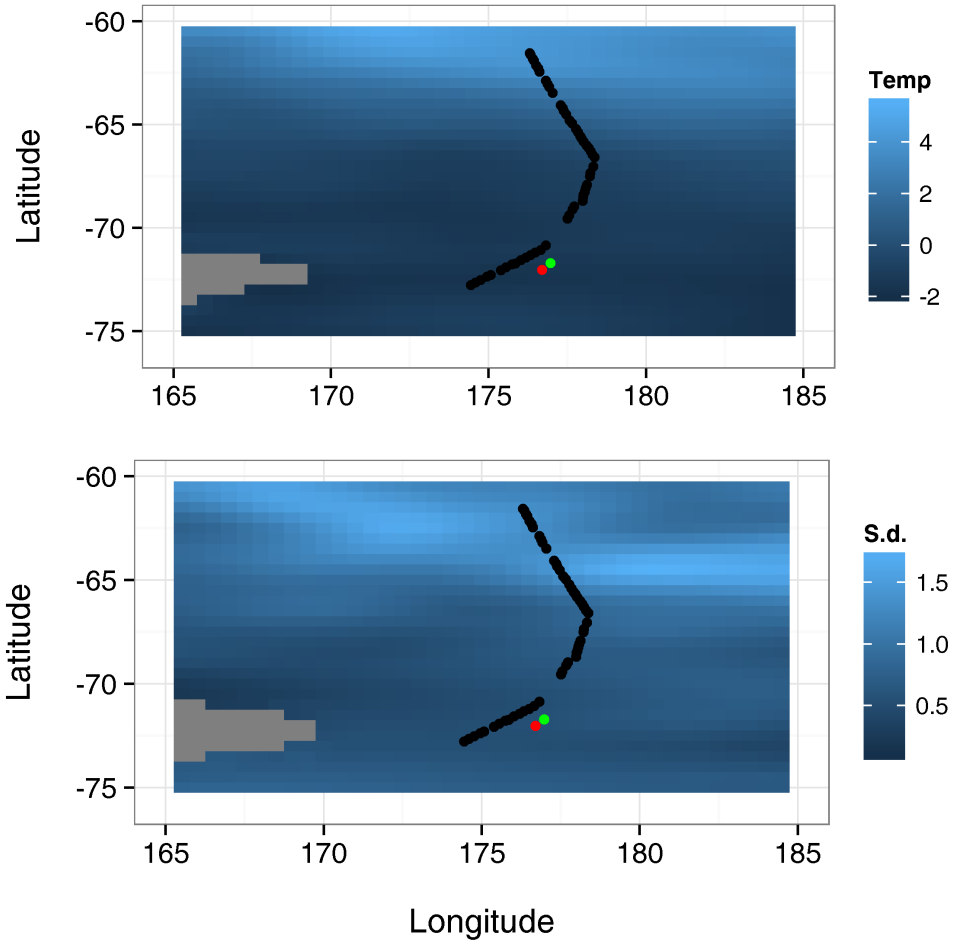


Figure 6.10: The temperature (°C) [top] and standard deviation of temperature [bottom] at approximately 5m depth. The black points [•] indicate the series of recorded GPS coordinates associated with the towed tag (**tag 186**). Also shown are the start [•] and end [•] location of tag 121. Regions in grey represent land. The plot region is adjusted to be approximately equal area projected.

zero to about 4°C. There is little pattern and high variation in both the total magnetic field strength and acceleration of the tag along the transect.

GPS coordinates were recorded along this transect. Several of the GPS points that appear to be outliers were removed along with those points that were outside the range of the CARS temperature data set leaving 72 GPS coordinates (Figure 6.10, Appendix C.1, page 341). The start location was $\mathbf{x}_0 = (-72.770833, 174.436667)$ and the end location was at $\mathbf{x}_T = (-61.550000, 176.301944)$. At each of the GPS locations the temperature recorded by the tag was identified and compared with the temperature expected by the CARS model at each of the GPS locations. The temperature expected by the CARS model showed the same general increasing trend as the temperature observed by the tag over the four day period, however the temperature observed by the tag was often higher than that expected by the model, particularly on Friday (23 February 2012, Figure 6.11).

We fit the model to the tag 186 **temperature** data to estimate the horizontal location of the tag at each time-step (Equation 6.7). Instead of using the standard deviation of depth covariate layer ($\sigma^c_{\mathbf{x},d}$) we decided to use the fixed value $\sigma^c = 0.1$. This was done because of the poor fit of the model to the observed temperatures (Figure 6.11). Moreover, strong spatial variations in the modelled standard deviation of temperature caused spurious instability in the estimation of the tag paths (see the simulation below in Section 6.5, page 240). We also used **depth** to exclude space where there is land (Equation 6.10). An **hourly** time-step was chosen, so all data (at 16 second resolution) were aggregated to the hourly median resulting in $T = 79$ time-steps.

Trace plots suggest that the MCMC was mixing well and the samples resulted in reasonable looking posterior distributions (Figure 6.12). The standard deviation parameter ($\sigma_{\mathbf{x}}$) passed the Heidelberger and Welch's stationarity test and had a Geweke Z-score of -1.036 (i.e. a p-value of 0.300). The Heidelberger and Welch's and Geweke diagnostics for most 2D positions along the path passed and resulted in acceptable Z-scores (Appendix C.3, page 345). The model expected temperature matched very

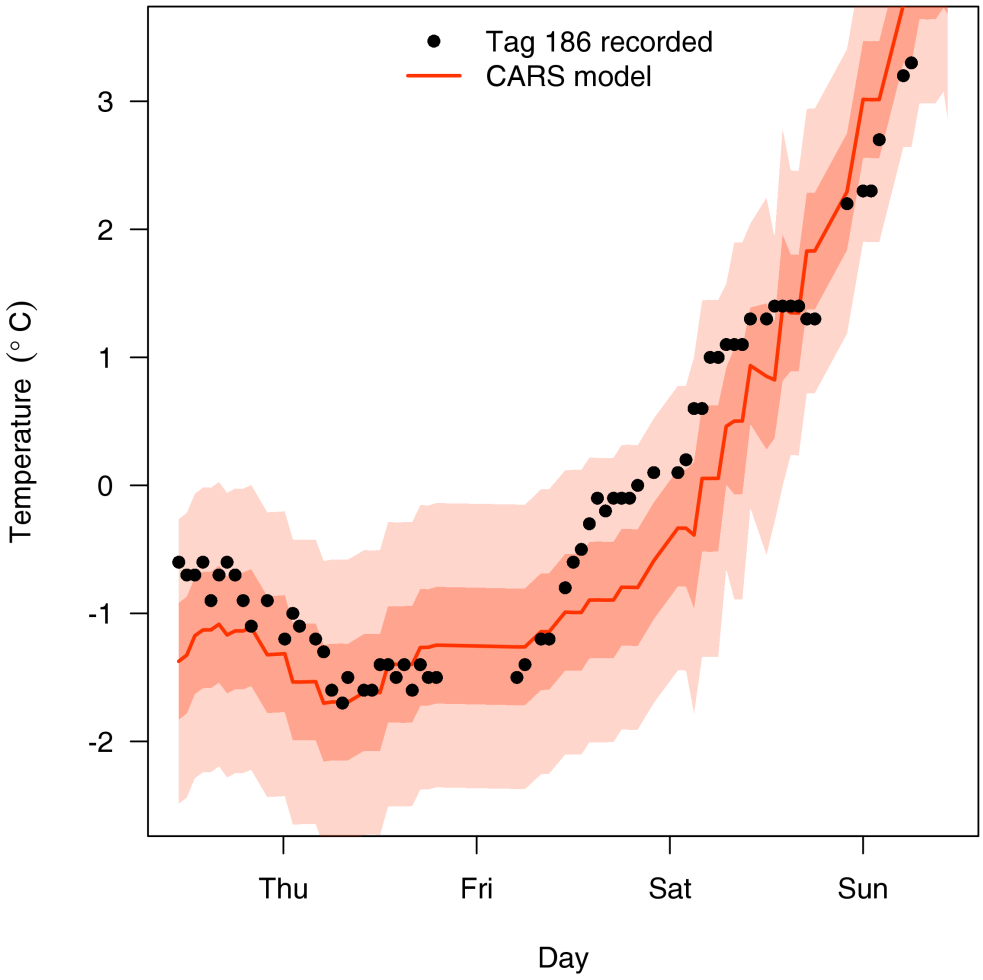


Figure 6.11: The temperature c_t (°C) observed by the towed tag (**tag 186**) at each of the recorded GPS locations [•] and the temperature expected by the CARS model $\mu^c_{\mathbf{x},d,m}$ given the location (latitude, longitude and depth) and month of the tag at each GPS location (see Figure 6.10) from 23 February 2012 to 26 February 2012. The shaded regions represent the 5, 25, 50, 75 and 95 percentiles of the temperature derived using the standard deviation of temperature at each location and depth $\sigma^c_{\mathbf{x},d}$.

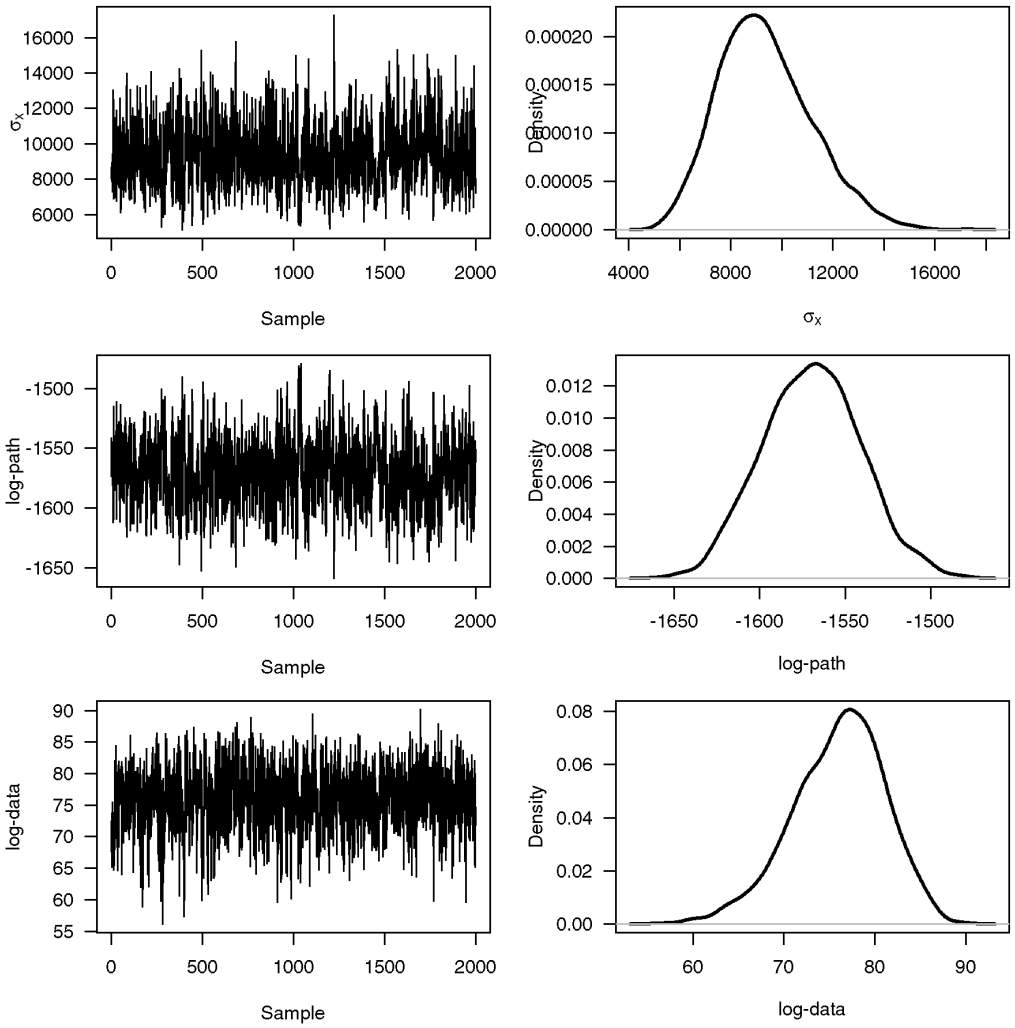


Figure 6.12: MCMC trace plots [left] and posterior distributions [right] and for the standard deviation parameter (σ_x), the log-likelihood of the path and the log-likelihood of the data in the towed tag (**tag 186**) model. The log-prior probability density is not plotted as this was constant.

well with the temperature observed by the tag (Figure 6.13). However, the

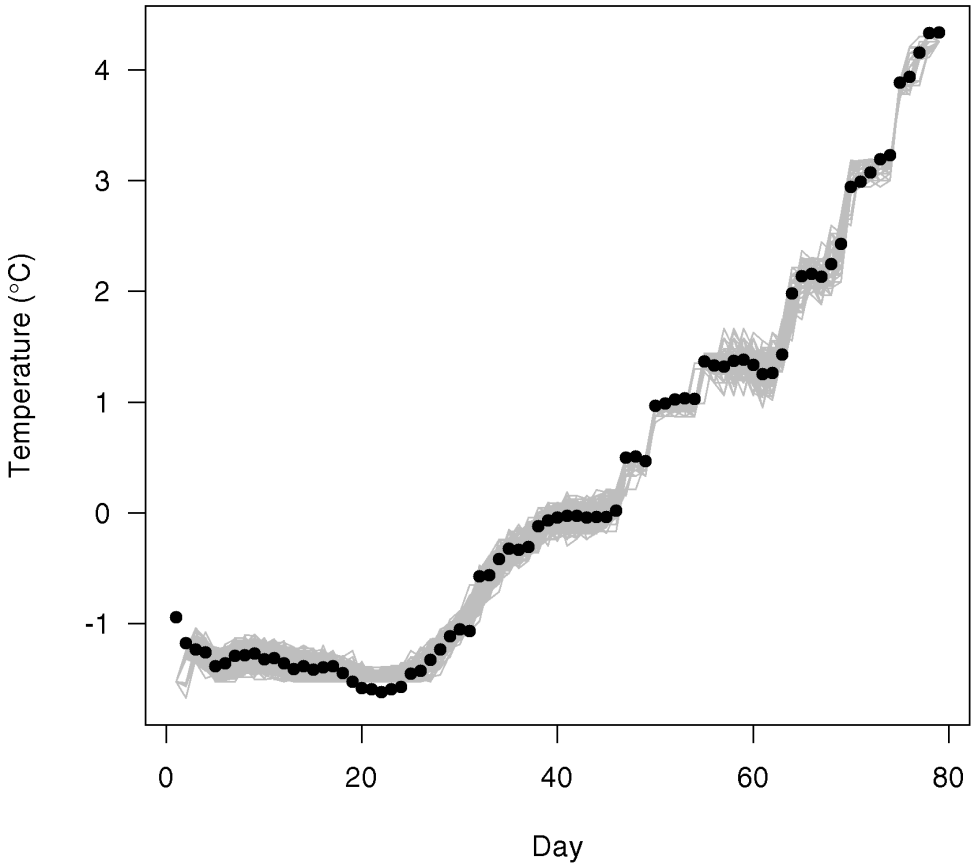


Figure 6.13: Sampled temperature (°C) expected by the model given the path [grey lines] and temperature observed by the towed tag (**tag 186**) [•].

model did not perform well in estimating the path of the tag during the tow (Figure 6.14).

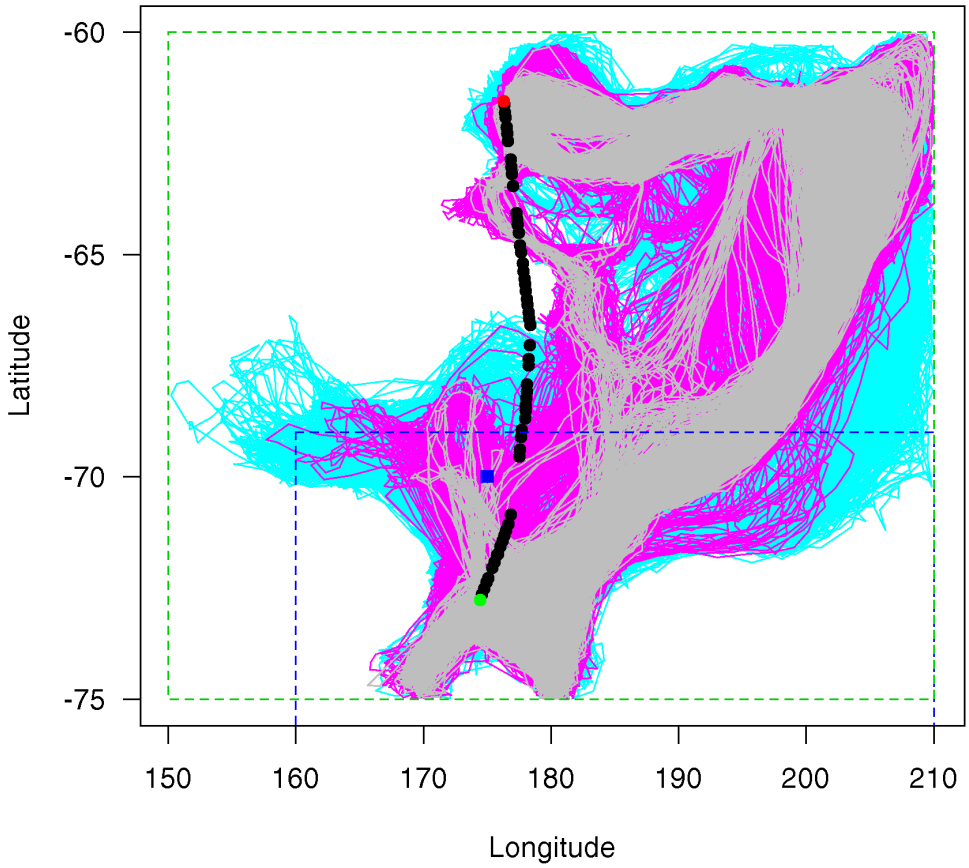


Figure 6.14: Sampled path taken by the towed tag (**tag 186**) [grey lines], the other two tempered chains $\beta_2 = 0.9$ [pink lines] and $\beta_3 = 0.8$ [cyan lines], and the known GPS path [•]. Also shown are the start [•] and end [•] locations, the projection centre [blue square], and the range of the depth [dashed blue box] and temperature [dashed green box] covariate layers.

6.5 Simulation

The poor performance found in recovering the path of the towed tag (tag 186) is apparently due to the poor fit of the temperature model to the observed data. We therefore re-fit the tag 186 data by replacing the observed temperatures with modelled temperatures. In this simulation, the observed depth (d_t), month (m_t), start position ($\mathbf{x}_0$), end position ($\mathbf{x}_T$) and known path ($\mathbf{x}_t$) were all left as they were above (Section 6.4, page 233). However, instead of using temperature recorded by tag 186, we used our lookup function to determine (deterministically) what the CARS temperature model (for each latitude, longitude, depth and month) expected the temperature to be at each time-step t. We fit the model in exactly the same way as described for the towed tag (tag 186) above (Section 6.4, page 233). Initially, we did use the standard deviation of depth covariate layer ($\sigma^c_{\mathbf{x},d}$) in fitting this model, however we found that the model could not recover the known path. This was because the standard deviation covariate layer changed spatially, and the model consistently preferred areas where $\sigma^c_{\mathbf{x},d}$ was low. Given our reservations about the CARS temperature model, at all locations and all times we use a fixed standard deviation for temperature $\sigma^c = 0.1°\text{C}$.

Trace plots suggest that the MCMC was mixing well and the samples resulted in reasonable looking posterior distributions (Figure 6.12). The parameter $\sigma_{\mathbf{x}}$ passed the Heidelberger and Welch's stationarity test and had a Geweke Z-score of 0.336 (i.e. a p-value of 0.737). The path passed the Heidelberger and Welch's stationarity test at all 2D positions and resulted in acceptable Z-scores from the Geweke test (Appendix C.3, page 347). The model expected temperature matched very well with the temperature observed by the tag (Figure 6.13). The model did an excellent job in estimating the path of the simulated tag (Figure 6.17).

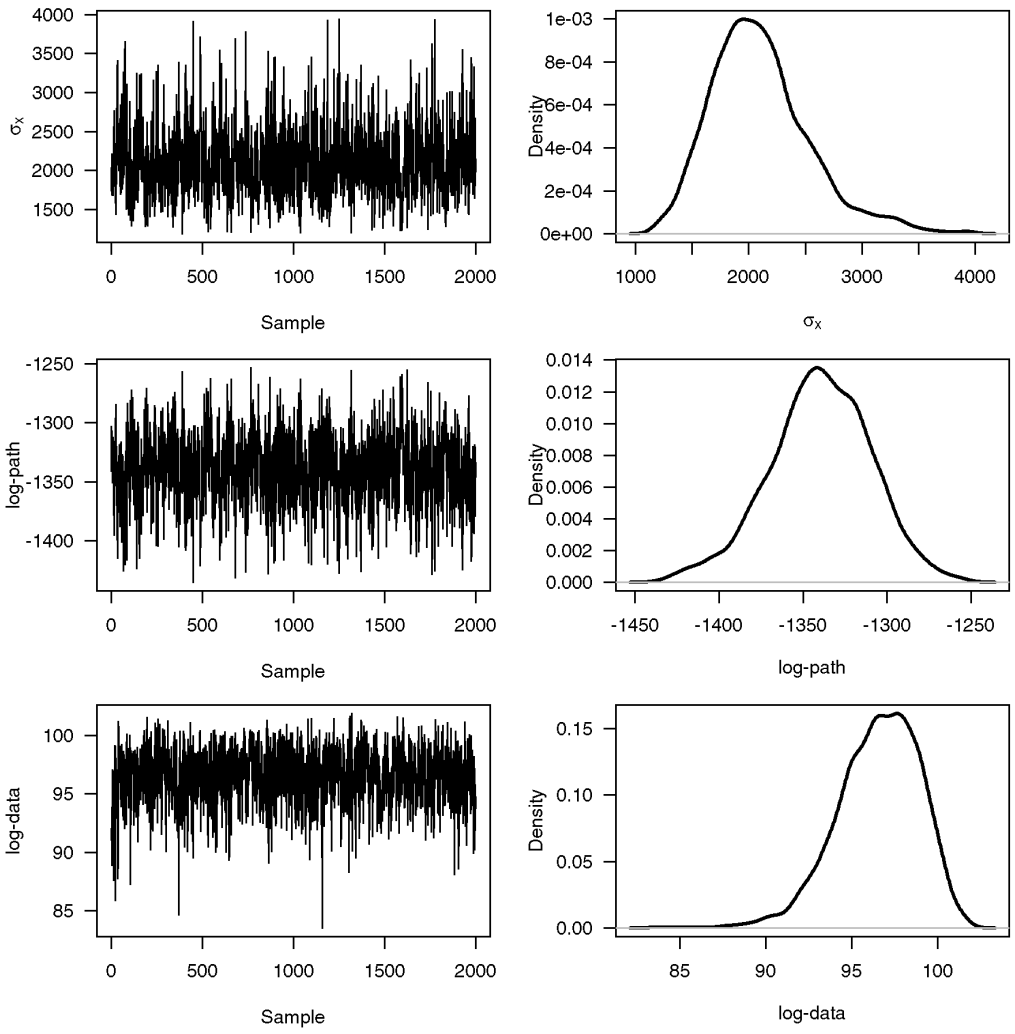


Figure 6.15: MCMC trace plots [left] and posterior distributions [right] and for the standard deviation parameter (σ_x), the log-likelihood of the path and the log-likelihood of the data for the **simulated** data model. The log-prior probability density is not plotted as this was constant.

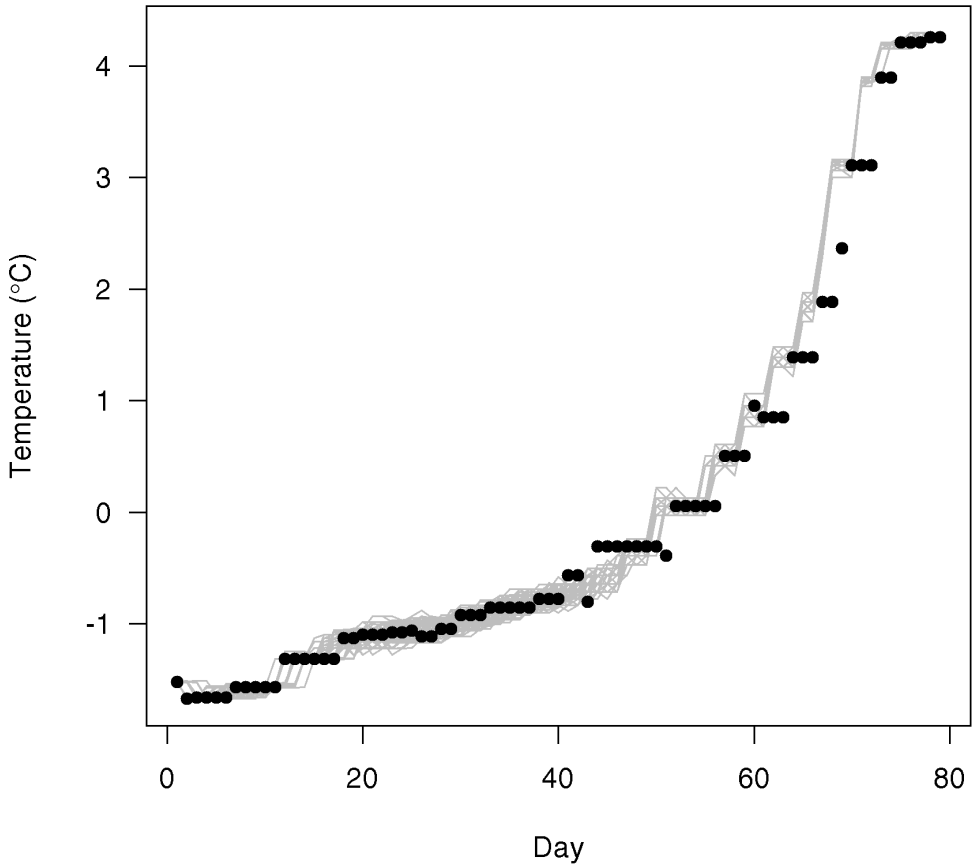


Figure 6.16: Sampled temperature (°C) expected by the model given the path [grey lines] and temperature observed by the **simulated** tag [•].

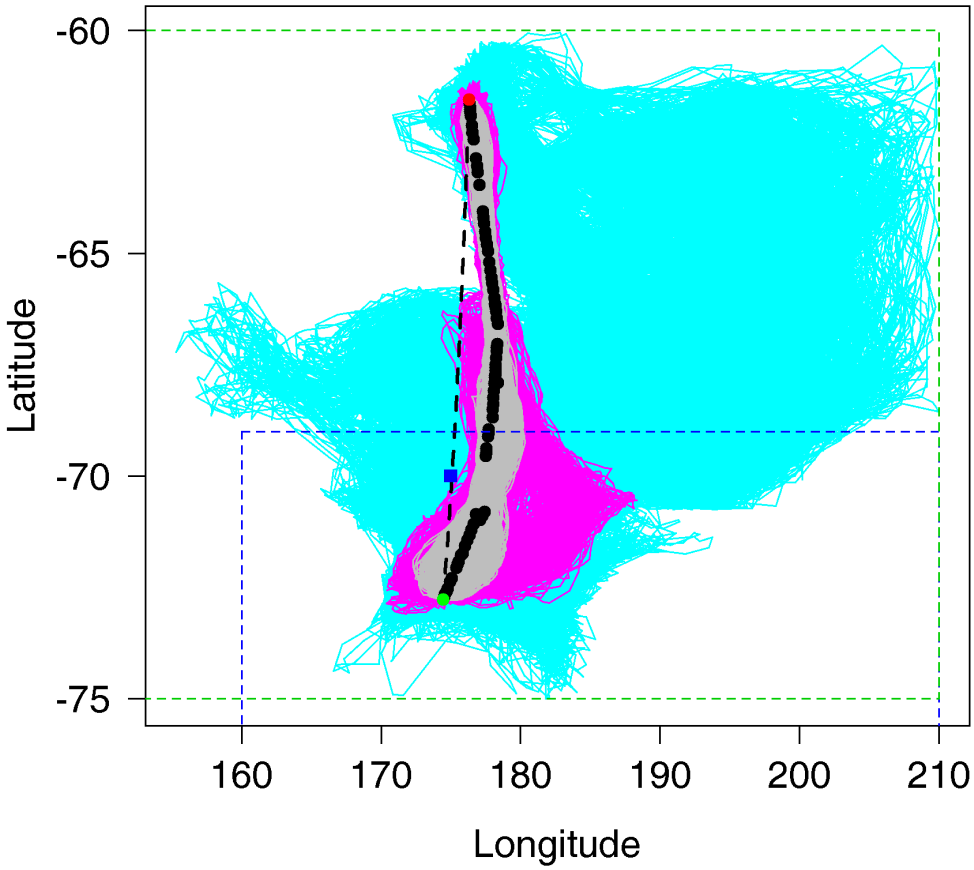


Figure 6.17: Sampled path taken by the simulated tag [grey lines], the other two tempered chains $\beta_2 = 0.9$ [pink lines] and $\beta_3 = 0.8$ [cyan lines], and the known **simulated** path [•]. Also shown are the start [•] and end [•] locations, the projection centre [blue square], and the range of the depth [dashed blue box] and temperature [dashed green box] covariate layers.

6.6 Tag 121: the tagged fish

The model was used to estimate the geographic location of the tagged fish (tag 121). The fish was released at $\mathbf{x}_0 = (-71.7145, 176.9678)$ and recaptured at $\mathbf{x}_T = (-72.0250, 176.6975)$. The tag was programmed to recorded environmental variables every 10 minutes (Figure 6.7). These data show that upon its release the fish descended to about 1000m depth where it remained until early March. It then ascended to about 500m depth for approximately one month. During this time, its depth varied little and the temperature remained relatively low. In early May the fish descended again and the temperature increased. For some unknown reason, in mid-July the light levels plummeted. At this time the fish did descend to a depth of about 1500m, but the fish did this previously at the beginning of May and the light levels did not drop this dramatically. Perhaps the fish moved under ice or there is some error in the data. Throughout the time series the total magnetic field strength varied a lot but did show some temporal differences. However, given the lack of any discernible pattern shown in the towed tag (tag 186) data across a much wider spatial range (Figure 6.6), we assume that these data may be reflecting some unmeasured variable (e.g. proximity to the sea floor) and chose not to model magnetic field strength.

In the model, we used a **weekly** time-step, rather than a daily time-step, to reduce MCMC run time. We discuss the implications of this choice in time-step further in the discussion, (Section 6.7, page 253)

In the Ross Sea region, Antarctic toothfish are also tagged using standard tagging methods (see Chapter 3, page 75). Within season recapture data include those fish that were tagged and recaptured again within the same fishing season. We used within season recapture data to gain some insight into the distances (m) moved by fish in a single time-step. For any given individual i, the expected distance from the origin of a random walk d_i after t_i time steps is

$$\begin{aligned} d_i &= \mu^j \sqrt{t_i} e^{\varepsilon_i}, \\ \log(d_i) &= \log(\mu^j) + \frac{1}{2}\log(t_i) + \varepsilon_i. \end{aligned} \tag{6.16}$$

where ε_i represents the unknown error distribution for this model. The mean distance moved in a single time-step μ^j is estimated using linear regression while fixing the slope at $\frac{1}{2}$ (Figure 6.18). The distribution of the

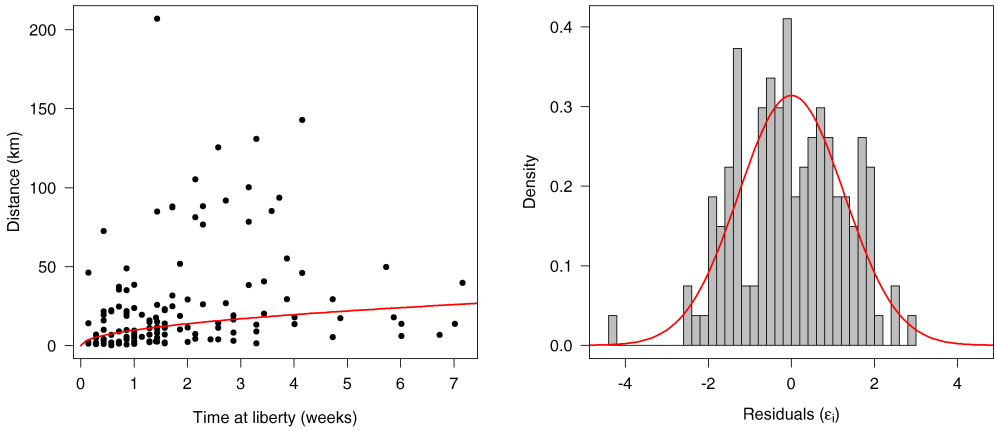


Figure 6.18: Individual time at liberty t_i (weeks) versus distance d_i (converted to km) travelled between tag-release and tag-recapture locations using within season recapture data with the fit of the linear model (Equation 6.16) to these data shown in red [left], and the distribution of residuals for the fit ε_i with the fitted normal distribution in red [right].

residuals ε_i of this fit informs the prior distribution. We found that $\varepsilon_i \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_j^2)$ (Figure 6.18). Therefore the distance moved by a fish in a single time-step could be expressed as $\log \mathcal{N}\left(\mu^j, \sigma_j^2\right)$ where $\mu^j = 9792.331$m and $\sigma_j = 1.270742$m. We can use this knowledge to check that the distances moved by modelled fish do not greatly exceed what is expected by this distribution. We also used this information to set an upper bound for the standard deviation parameter ($\sigma_{\mathbf{x}}$) at 32000m

$$\pi\left(\sigma_{\mathbf{x}}\right) \sim \mathcal{U}\left(0.001, 32000\right). \tag{6.17}$$

The choice of an upper bound at 3200m is simply three times the mean, plus a little bit. Therefore, our prior is no longer uninformative.

Both depth and temperature were used to estimate location (Equations 6.7

and 6.11). The model was restricted to be within the bounds of both the depth and temperature covariate layers. The MCMC for this model was done in two phases. The first phase involved fitting to temperature and depth, while turning the log-likelihood contribution of the path off. This resulted in a path with unrealistically large jumps between locations (e.g. $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$) at each time-step in the model. However, it allowed the model to explore parameter space a lot faster and find locations with suitable depths and temperature at each time-step. In the second phase, the log-likelihood contribution of the path was turned on and the MCMC was run.

Trace plots suggest that the MCMC was mixing well and the samples resulted in reasonable looking posterior distributions, although the posterior for $\sigma_{\mathbf{x}}$ was against the upper bound some of the time skewing the distribution slightly (Figure 6.19). The standard deviation parameter $\sigma_{\mathbf{x}}$ passed the Heidelberger and Welch's stationarity test and had a Geweke Z-score of 1.701 (i.e. a p-value of 0.089, only just passed). The distances moved by the tag between each time-step in the MCMC simulations were a little higher than those expected (from investigating the distances between tag-release and tag-recapture of other toothfish tagged in the Ross Sea, Figure 6.20). This suggests that the upper bound of the uniform prior on $\sigma_{\mathbf{x}}$ could be decreased a little.

The model expected depth of the sea floor was generally not much deeper than the fish, except in the first four days (Figure 6.21). The model expected temperature did not agree with the temperature observed by the tag (Figure 6.22). The model indicated that in the sampled locations and depths, the temperature should be much lower (about -2°C) than that observed by the tag (about 0°C). Finally, the estimated path of the tagged fish (tag 121) is shown in Figure 6.23 below. The hole in the middle of the paths is largely caused by the bathymetry as it includes depths that are much greater than those measured toothfish (Figure 6.24). The path passed the Heidelberger and Welch's stationarity test at all 2D positions and resulted in acceptable Z-scores from the Geweke test with 13 points failing in 1D at the 5% level (Appendix C.4, page 349).

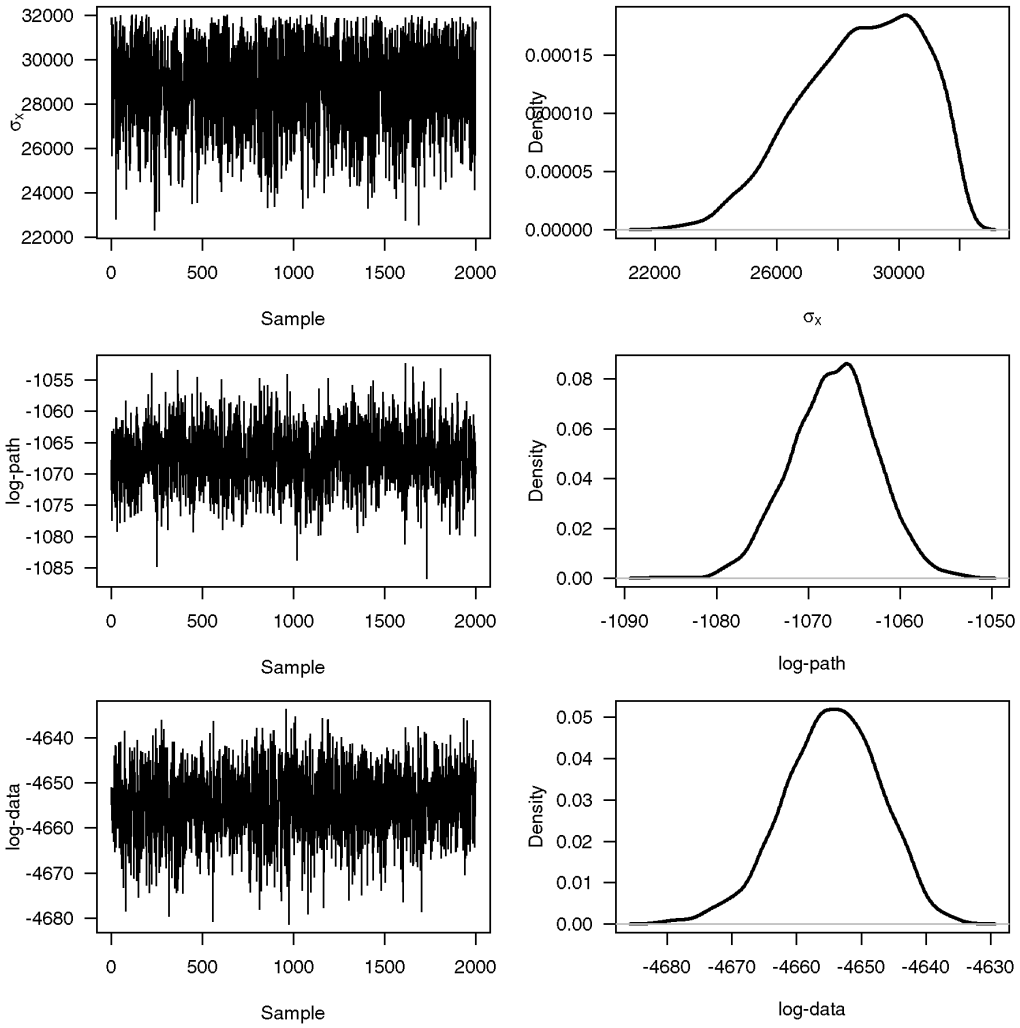


Figure 6.19: MCMC trace plots [left] and posterior distributions [right] and for the standard deviation parameter (σ_x), the log-likelihood of the path and the log-likelihood of the data in the tagged fish (**tag 121**) model. The log-prior probability density is not plotted as this was constant.

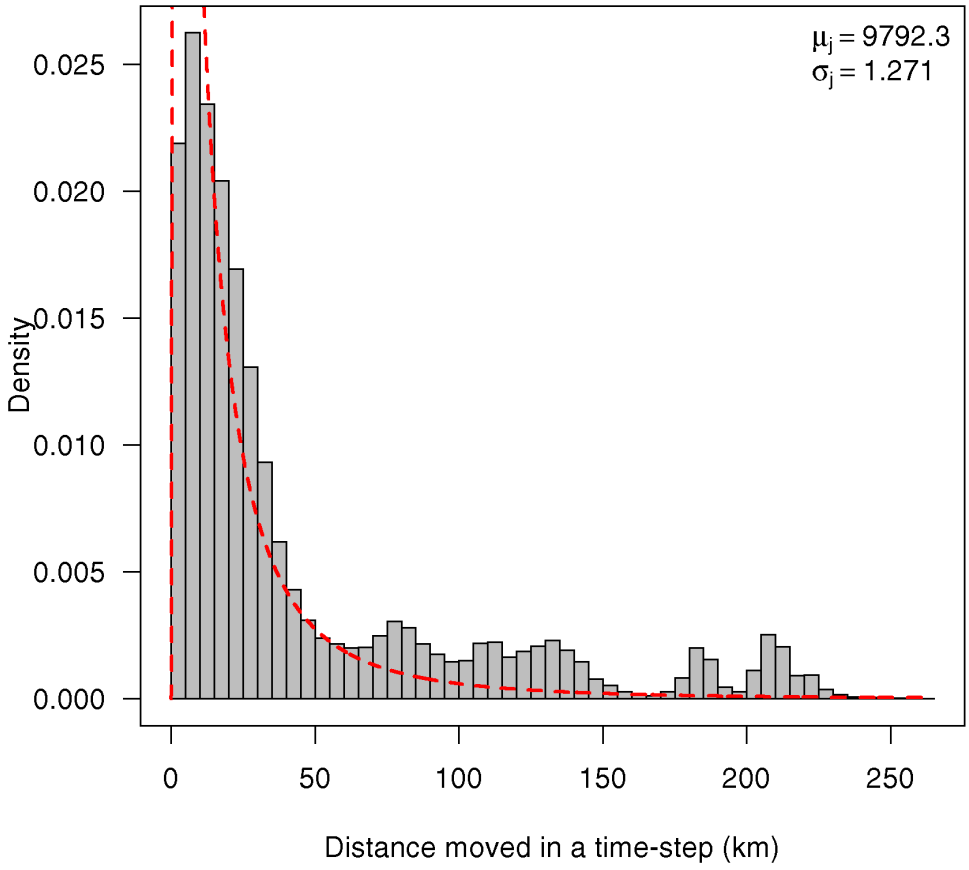


Figure 6.20: Histogram of the sampled distances (km) between locations $\mathbf{x}_t$ and $\mathbf{x}_{t+1}$ for $t = 0, \ldots, T-1$ at each time-step and the distance that we might expect a fish to move estimated using alternative information [dashed red line]. The estimated parameters of the lognormal distribution shown in red are given at the top right of the plot.

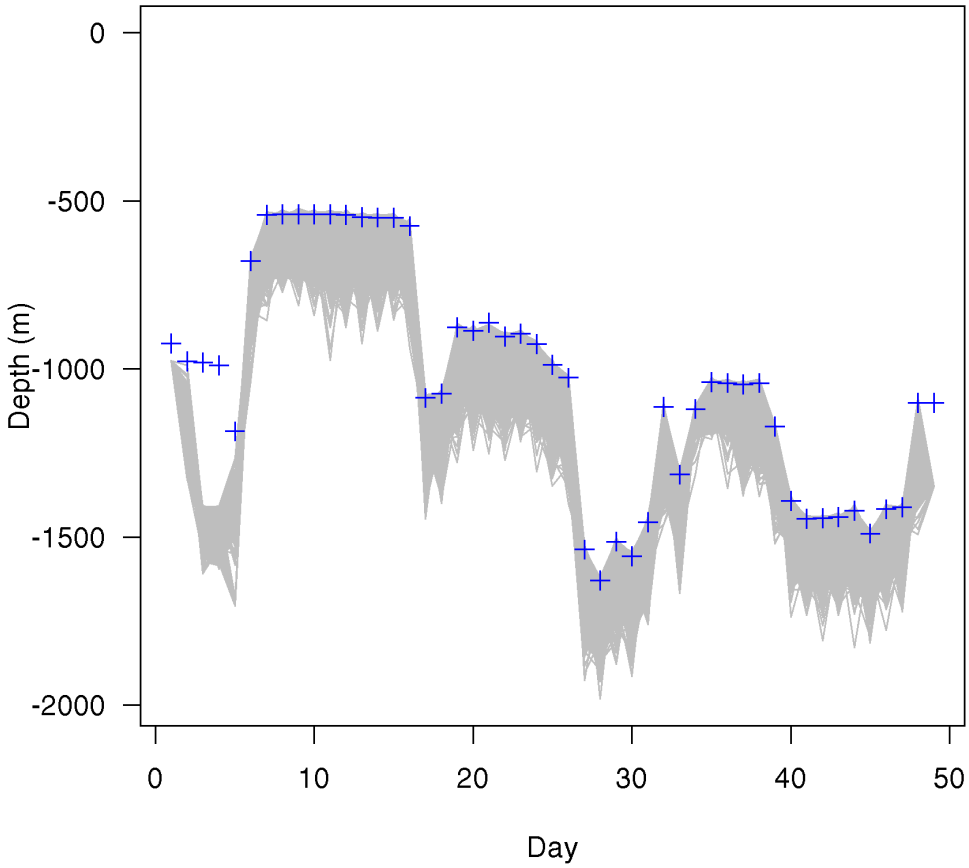


Figure 6.21: Depth (m) of the sea floor under the fish given the expected path [grey lines] and observed depth of the tagged fish (**tag 121**) [+].

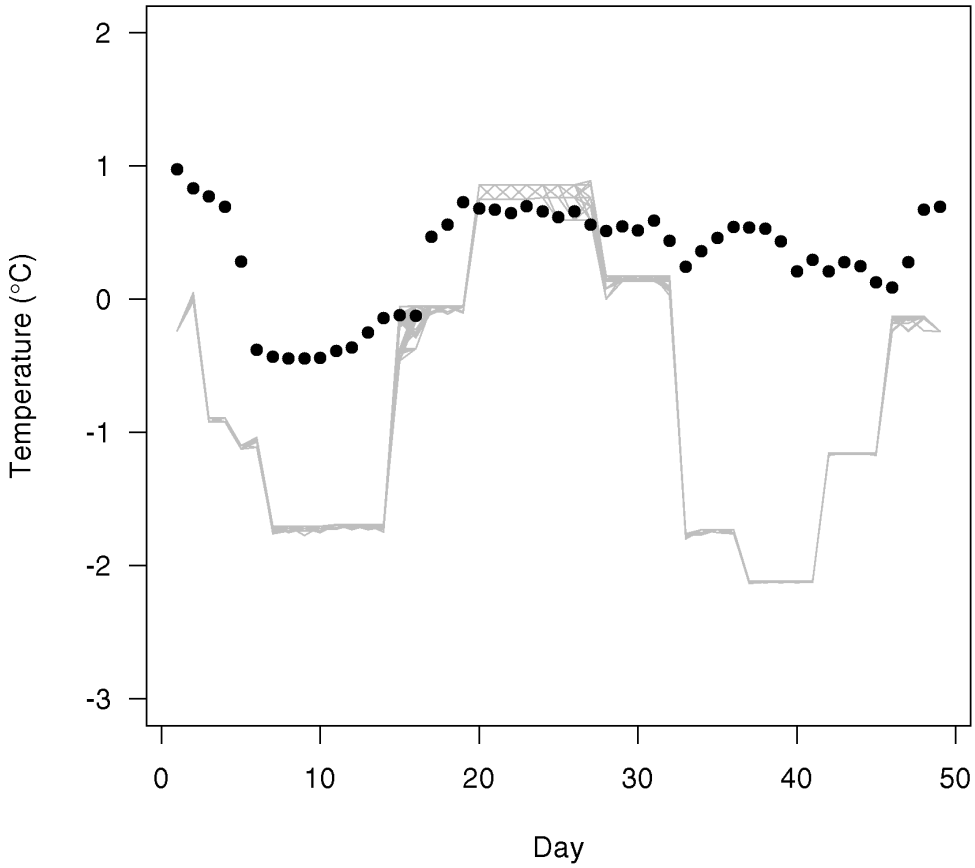


Figure 6.22: Sampled temperature (°C) expected by the model given the path [grey lines] and temperature observed by the tagged fish (**tag 121**) [•].

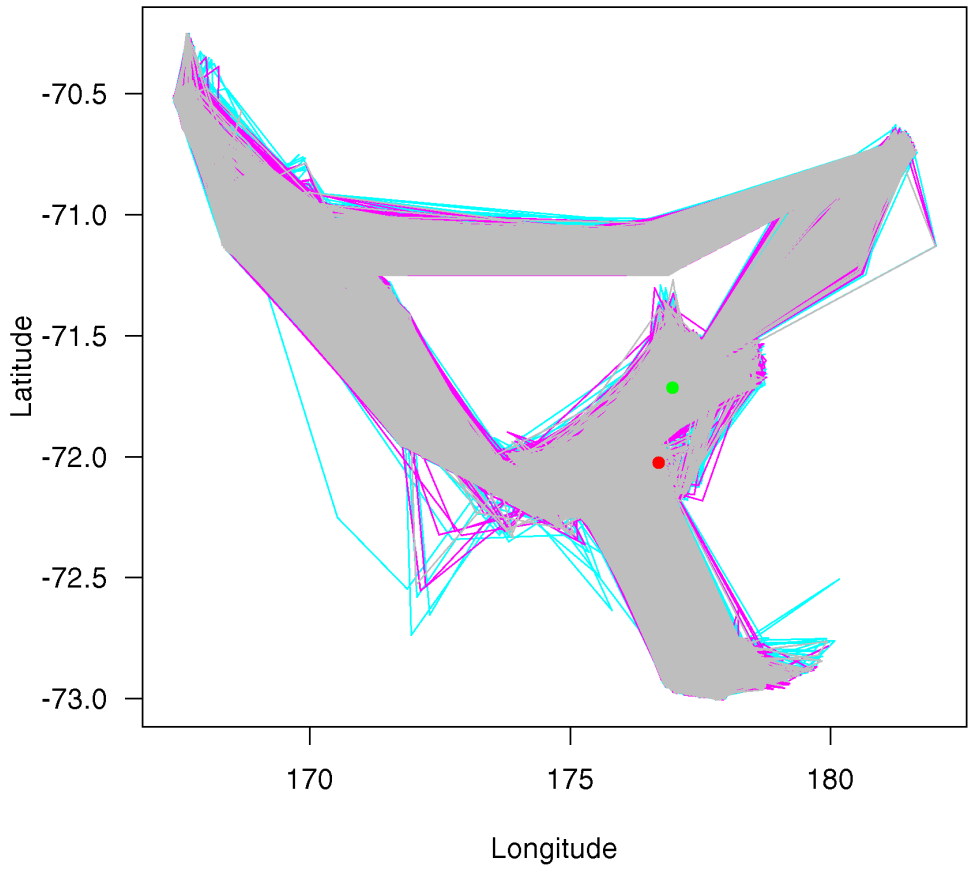


Figure 6.23: Sampled path taken by the tagged fish (**tag 121**) [grey lines], the other two tempered chains $\beta_2 = 0.9$ [pink lines] and $\beta_3 = 0.8$ [cyan lines]. Also shown are the start [•] and end [•] locations.

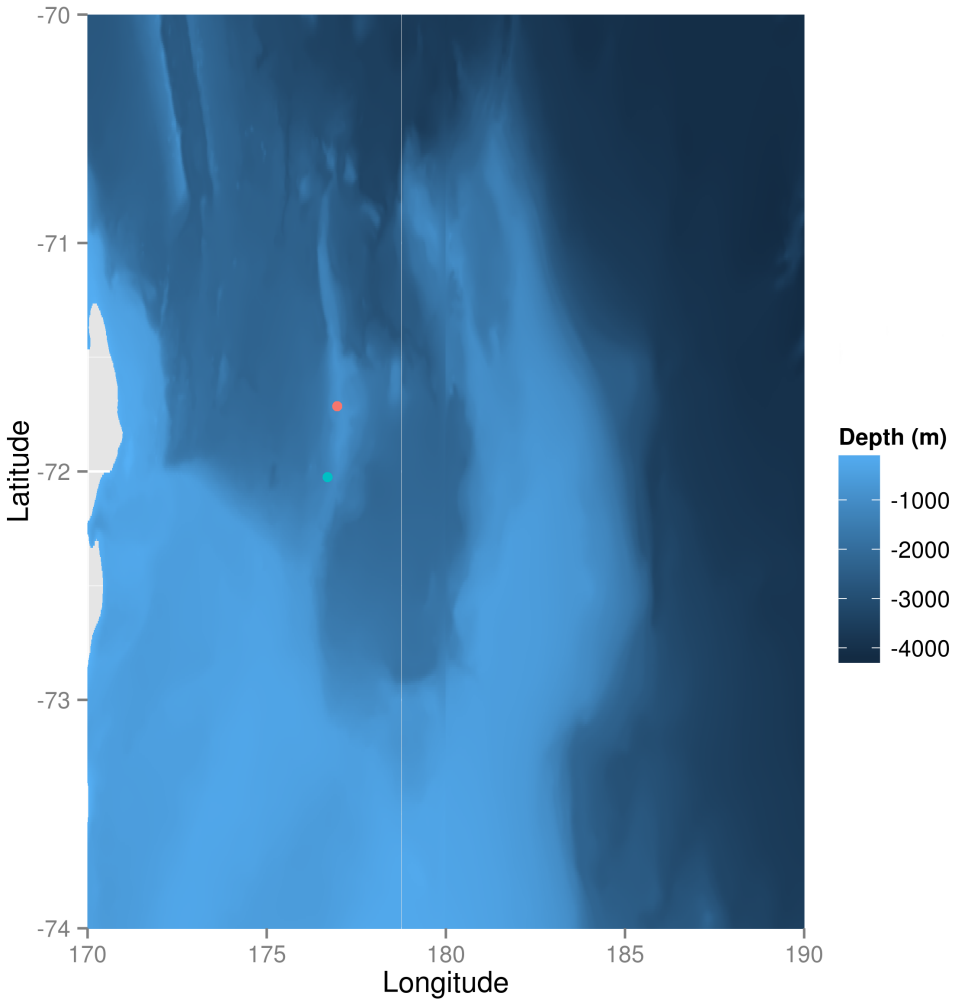
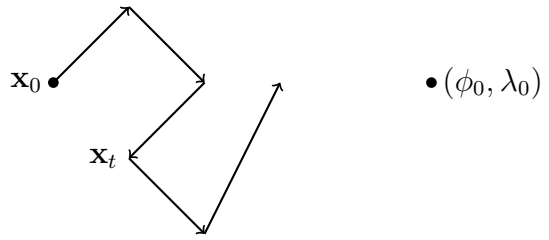


Figure 6.24: Bathymetry in the region surrounding the tagged fish (**tag 121**). Also shown are the start [•] and end [•] locations.

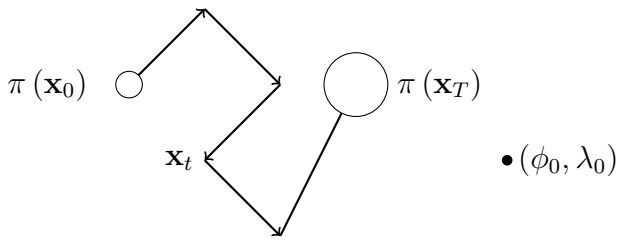
Despite our doubts about the quality of the CARS temperature model, we attempted to at least try and find a good fit to temperature in the model. The MCMC was restarted (from the last recorded state in the MCMC simulation presented above), but the path and depth contribution to the log-likelihood was turned off (i.e. only the prior and temperature contributed to the total log-likelihood). Running the model in this way allows the fish to explore as far as it likes between each time-step, and even explore space that would otherwise be excluded by the depth data (i.e. we let the fish cross land). However, the model could not find locations that could match the two dips in temperature expected by the CARS temperature model.

6.7 Discussion

We have developed novel Bayesian methods for geolocating individual fish using PSATs. The difference between this model and other models is that our **process model** is conditional on $\mathbf{x}_0$ and $\mathbf{x}_T$. In other words, we have fixed the start and end location, assuming that they are known without error. In contrast, other process models may not fix the end point (often called diffusion models or random walks):



These models may additionally include some kind of systematic bias or a "drift" component (advection-diffusion or biased random walks, see Jonsen et al. 2013). By fixing the start and the end of the path, we are using the best two pieces of information we have (the known start and end locations) to help estimate the path taken by an individual fish and reduce uncertainty associated with these estimates. Alternatively, one might wish to admit some uncertainty here (i.e. GPS recorded coordinates do have some error associated with them), and let the start and end be random variables but place informed priors on these locations:



In this way, we could admit as little or as much uncertainty about the start and end locations as is necessary. Or, rather than the process model in Equation 6.5, we could use

$$\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_{t+1}, \sigma_{\mathbf{x}}, a_t \sim \mathcal{N}\left(\frac{1}{2}\left(\mathbf{x}_{t-1} + \mathbf{x}_{t+1}\right), a_t\frac{\sigma_{\mathbf{x}}^2}{2}\mathbf{I}\right),$$

thus scaling the variance by the acceleration a_t (or we could replace a_t with some mean $(a_{t-1} + a_t + a_{t+1})/3$). This may help the estimation of the standard deviation parameter $\sigma_{\mathbf{x}}$. Also, the process model only considers movement on a 2D horizontal plane. Any vertical movement of the fish is not captured by the process model and may contribute to the average total movement of an individual fish within any given time-step (e.g. distance travelled within a day). This may result in us underestimating the distance moved by a fish within a time-step. Further improvements to the process model might include this additional vertical component of the distance travelled (i.e. a 3D process model). Finally, the process model is set up as a continuous space, discrete time model. This means that the user must specify the size of a time-step (e.g. hourly, daily, weekly) and the data measured by the tag must be aggregated to this temporal resolution. If using a relatively large time-step then information may be lost. For example, the fish may descend to depth for just a single day within a week, if a weekly time-step is chosen then we may miss this in our model. However, small time-steps result in the model taking a lot longer to run making MCMC very slow. Any improvements here might involve changing the model to a continuous time formulation (i.e. a Markov process model), using smaller time-steps and investigating the computational potential of Julia further, summarising the data differently and doing sensitivities (e.g. doing model runs that use the minimum and maximum depth within a time-step, rather than just the median), or developing likelihoods that include variation in the variables measured by the tag as well as the environ-

ment (e.g. a likelihood that stipulates that the fish was within some depth range in any given time-step).

We then develop **observation models** that attempt to use depth and temperature to geolocate Antarctic toothfish in the Ross Sea. While not entirely successful, the use of depth data to help estimate position is novel. Our implementation here has made strong assumptions about fish behaviour relative to the sea floor. These assumptions may not be valid except for species that we are sure are exclusively demersal. It is unlikely that Antarctic toothfish are exclusively demersal (J. Fenaughty, Pers. Comm.). Also, we did not include uncertainty in the variables measured by the tag itself, or incorporate any uncertainty when aggregating the data measured by the tag (from measurements taken every 16 seconds and 10 minutes to hourly or weekly time-steps for tag 186 and tag 121 respectively).

The model did not do well in estimating the known path of the towed tag (**tag 186**) when fit to the data observed by the tag. In this model, temperature was the main source of information (depth was only used to exclude some areas that were too shallow or land). However, the model did very well in estimating the known path of the tag when the temperature observed by the tag was **simulated** from the CARS temperature model (and the standard deviation of temperature was fixed). When the spatially explicit standard deviation estimates provided by the CARS were used, for both tag 186 and in the simulation, they resulted in poor estimates of the known paths. Therefore, the simulation study provides the proof of concept for this modelling framework. However, the method relies on good data to accurately geolocate fish. The CARS temperature model is inadequate for such fine-scale modelling applications. Both the standard deviation estimates and the temperature estimates have proven themselves to be problematic in trying to geolocate fish.

Light has been used successfully in the past to help estimate a fishes location through time (Welch & Eveson 1999). However, toothfish inhabit deep waters under ice and below the photic zone. In addition, long periods of constant daylight or constant darkness preclude the use of light for Antarctic toothfish geolocation year round. Despite this, we recommend

further research to incorporate **light** and **magnetic field strength** as additional observation models. Light could be used to improve the estimates of location for the towed tag (tag 186), given the strong light signals in the recorded data (Figure 6.6). However, light would not be useful in estimating the location of the actual fish (tag 121). Despite this, light would be well worth incorporating into this modelling framework if we are to apply it to other species.

The pattern, or lack of, in total magnetic field strength shown by the towed tag (tag 186, Figure 6.6) suggests that the pattern shown by tag 121 (Figure 6.7) might be an artifact caused by some other factor (i.e. the proximity to the sea floor). This discouraged us from using these data to help improve geolocation of toothfish. While magnetic field strength was not used here, it may be used successfully elsewhere, particularly in species with broader spatial ranges (e.g. tuna). It also appears that the newer versions of the PSAT tags provide more accurate magnetic field strength data (the technology has improved; K. Echave, K. Coutre and J. Nielsen Pers. Comm.). Total magnetic field strength is modelled globally and available by depth and with monthly time steps (`https://www.ngdc.noaa.gov/geomag/WMM/back.shtml`). The best way to include these information within our modelling framework would be to call the magnetic field model as sub-models of our model (i.e. the Julia code accesses the Fortran magnetic field models directly within the MCMC algorithm). Alternatively, these data may be provided as a grid in the same way that depth and temperature are used in this chapter.

The implications of this work suggest that the use of PSATs for geolocating Antarctic toothfish may not be appropriate and perhaps other avenues of research would be better pursued if geolocation is the goal (e.g. acoustic tagging). However, discussions with NOAA scientists (K. Echave and K. Coutre) and J. Neilsen suggest that the technology has improved in the newer tags and that they provide more accurate magnetic field strength and acceleration data. Therefore, we recommend that more tags be placed on Antarctic toothfish so that magnetic field strength data can be utilised within the modelling framework. With the addition of light and magnetic field observation models, this modelling framework could be a powerful

tool for other species as well.

Chapter 7

Bayesian emulation

In this chapter we develop Bayesian emulators for practical applications in fisheries research and provide examples. We begin by introducing deterministic univariate emulators. We then extend the emulation framework to include stochasticity and develop a stochastic multivariate emulator of an agent-based model (ABM) of snapper in northern New Zealand (*Pagurus auratus*, SNA 1, for more information on snapper see Chapter 3, page 77) as described in Chapter 4.

7.1 Introduction

Complex models can be computationally expensive, taking many hours or days to do a single run. Models of this ilk make inference using standard methods impractical. For example, the ABM described in Chapter 4 takes about 16 hours to do a single run (for the snapper example). Attempting to use standard MCMC methods for relatively few samples from the posterior distribution of this model, say 1000 samples with no thinning, would take about two years.

However, methods do exist for speeding up inference of such computationally expensive models. In Chapter 2 (page 71) we briefly described approximate Bayesian computation (ABC) and Bayesian emulation as feasible alternatives for Bayesian inference in complex models. In this chapter

we cover Bayesian emulation in more detail, extend the method beyond what is currently described in the literature (e.g. Goldstein & Rougier 2006, Hankin 2005, Oakley & O'Hagan 2004), and apply the method to fisheries problems.

A Bayesian emulator's task is to quickly estimate a function $f(\theta)$ for an arbitrary value of its argument θ. This might be achieved by drawing a smooth curve through a set of design points $(\theta_i, f(\theta_i))$ and using this curve to predict $f(\theta)$ for any θ. If the desired θ is in between the largest and smallest of the θ_i's the problem is called **interpolation**, if θ is outside that range it is called **extrapolation**. Of course, care must be taken when choosing a set of design points (θ_i's) so as to avoid extrapolation, because estimates of $f(\theta)$ beyond θ_i can be unstable (e.g. Figure 7.1). Interpolation

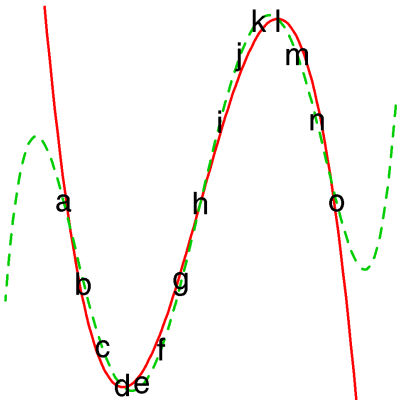


Figure 7.1: The fit of a third order polynomial [solid red line] and a sixth order polynomial [dashed green line] to the points **a** to **o**.

is related to, but distinct from, **function approximation**. This consists of finding an approximate (but easily computable) function to use in place of a more complicated one. In the case of interpolation we know the output of function $f(\cdot)$ at points that may not be of our choosing or limited to a smaller set than desired. In function approximation the function $f(\cdot)$ can be computed at any desired points for the purpose of developing our approximation. See Chapter 3 of Press et al. (1986) for more detail on interpolation and extrapolation and Chapter 5 for details on function approxi-

mation.

Bayesian emulation includes aspects of interpolation/extrapolation and function approximation. Furthermore, an emulator acts both as an approximation to the function and as an assessment of the uncertainty introduced by the approximation. Bayesian emulation coupled with Bayesian inference provides a method for making inference of computationally expensive computer models tractable (e.g. Henderson et al. 2009) as a good emulator will be much faster than the model it is emulating.

A univariate emulator provides an approximation to a function that takes multiple input parameters ($\boldsymbol{\theta}$)[1] and returns a scalar output (y)

$$y = f(\boldsymbol{\theta}).$$

A multivariate emulator is a generalisation of a univariate emulator and provides an approximation to a function that takes multiple input parameters ($\boldsymbol{\theta}$) and returns a vector of outputs ($\mathbf{y}$)

$$\mathbf{y} = f(\boldsymbol{\theta}).$$

Bayesian emulators are based on Gaussian processes ($\mathcal{GP}$). A Gaussian process is a type of stochastic process in which every point in some input space is associated with a normally distributed random variable and any collection of those random variables is multivariate normal. If we write

$$g(x) \sim \mathcal{GP}\left(m(x), c(x, \cdot)\right),$$

then we are stating that the random function $g(x)$ is distributed as a Gaussian process with mean function $m(x)$ and covariance function $c(x, \cdot)$. Therefore, by the definition of Gaussian processes, if we take any set $\mathcal{T} : (t_1, \ldots, t_{\mathcal{T}})$ then $\mathbf{x} = (x_1, \ldots, x_{\mathcal{T}})$ is multivariate normal

$$\mathbf{x} \sim \mathcal{N}\left(\boldsymbol{\mu}, \Sigma\right),$$

with $\mu_t = m(x_t)$ and $\Sigma_{t,t'} = c(t, t')$ for $t, t' \in \{1, \ldots, \mathcal{T}\}$.

[1] Actually, an emulated function could take a single input parameter (θ). We provide an example of this in Section 7.2.2 (page 275).

Computationally expensive models are common in many scientific disciplines including fisheries science. However, thus far there have been very few applications of Bayesian emulation in the literature (but see Vernon et al. 2010 which applies the method to a galaxy formation model). Indeed, the method has not been applied in fisheries despite the fact that we commonly encounter computationally expensive models that preclude the use of standard Bayesian inference methods. Examples of these complex models include the likes of Atlantis (`http://www.cmar.csiro.au/research/mse/atlantis.htm`) and InVitro (Gray et al. 2006, Little et al. 2006). Therefore, in this chapter we further develop Bayesian emulators in the fisheries context, and provide a proof of concept for their use in complex fisheries settings.

7.2 Univariate emulators

A list of the variables used in describing univariate emulators is provided in Tables 7.1 and 7.2. We define the parameters, or inputs, of a computer model to be some vector $\boldsymbol{\theta} = \{\theta_j\}_{j=1}^{p}$, and the output of the model to be a scalar y. We represent the computer model as some deterministic function $f(\cdot)$, thus

$$y = f(\boldsymbol{\theta}).$$

By deterministic we mean that the same input ($\boldsymbol{\theta}$) always yields the identical output (y). In Section 7.3 (page 278) below we relax this requirement so that we can emulate stochastic simulation models.

Our goal is to estimate $\boldsymbol{\theta}^*$, the "true" value(s) of the inputs conditional on some data or observations. We define y^* to be the output of the model when $\boldsymbol{\theta}^*$ is the input, thus

$$y^* = f(\boldsymbol{\theta}^*).$$

If $f(\cdot)$ is computationally expensive, then finding $\boldsymbol{\theta}^*$ using standard inference methods can be impractical.

Instead, we approximate $f(\boldsymbol{\theta})$ by a Gaussian process

$$y = \mathbf{h}(\boldsymbol{\theta})^T\boldsymbol{\beta} + \varepsilon(\boldsymbol{\theta})$$

Table 7.1: Notation used in discussing and defining Bayesian emulation. The inputs and outputs of a computer model.

Symbol	Type	Dimension	Description
n	scalar	1	Number of design points $i \in n$
p	scalar	1	Dimension of input space $j \in p$
q	scalar	1	Number of regression functions $k \in q$
$\boldsymbol{\Theta}$	vector	p	The input space
$\boldsymbol{\theta}$	vector	$p \times 1$	An element of the input space $\boldsymbol{\theta} = \{\theta_j\}_{j=1}^{p}$ where $\boldsymbol{\theta} \in \boldsymbol{\Theta}$
$\boldsymbol{\theta}^d$	matrix	$n \times p$	Matrix of inputs $\boldsymbol{\theta}^d = \{\boldsymbol{\theta}_i^d\}_{i=1,j=1}^{n,p}$ for which we have outputs $\mathbf{y}^d = \{y_i^d\}_{i=1}^{n}$
$\boldsymbol{\theta}'$	vector	$p \times 1$	Vector of inputs for which an output is to be approximated
$\boldsymbol{\theta}^*$	vector	$p \times 1$	The true value of $\boldsymbol{\theta}$
$\mathbf{y}^d$	vector	$n \times 1$	Vector of outputs at the design points $\mathbf{y}^d = \{y_i^d\}_{i=1}^{n}$
y'	scalar	1	The approximation of y given the vector of inputs $\boldsymbol{\theta}'$
y^*	scalar	1	The true value of y
y^o	scalar	1	Observed data

Table 7.2: Notation used in discussing and defining Bayesian emulation. The emulator.

Symbol	Type	Dimension	Description
n	scalar	1	Number of design points $i \in n$
p	scalar	1	Dimension of input space $j \in p$
q	scalar	1	Number of regression functions $k \in q$
$\boldsymbol{\beta}$	vector	$q \times 1$	Vector of regression coefficients
$\widehat{\boldsymbol{\beta}}$	vector	$q \times 1$	The estimated value of $\boldsymbol{\beta}$ updated in light of the design outputs $\mathbf{y}^d$
σ_e^2	scalar	1	Emulator variance
$\widehat{\sigma}_e^2$	scalar	1	A posteriori estimate for the variance
$\mathbf{h}(\boldsymbol{\theta})$	vector	$q \times 1$	Basis function
$\mathbf{H}$	matrix	$n \times q$	Matrix of basis function values evaluated at each input $\boldsymbol{\theta}^d$ (from simulator)
$\mathbf{Q}$	matrix	$p \times p$	Diagonal matrix of roughness parameters
$c(\boldsymbol{\theta}, \boldsymbol{\theta}'\|\mathbf{Q})$	scalar	1	Correlation function
$\mathbf{A}$	matrix	$n \times n$	Correlation matrix
$\mathbf{t}(\boldsymbol{\theta}\|\boldsymbol{\theta}^{\mathbf{d}}, \mathbf{Q})$	vector	$n \times 1$	Vector of covariances between $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^d$
$m^*(\boldsymbol{\theta}\|\boldsymbol{\beta}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q})$	scalar	1	Prior expectation
$m^{**}(\boldsymbol{\theta}\|\widehat{\boldsymbol{\beta}}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q})$	scalar	1	A posteriori expectation
$c^*(\boldsymbol{\theta}, \boldsymbol{\theta}'\|\boldsymbol{\theta}^d, \mathbf{Q})$	scalar	1	Prior variance
$c^{**}(\boldsymbol{\theta}, \boldsymbol{\theta}'\|\boldsymbol{\theta}^d, \mathbf{Q})$	scalar	1	A posteriori variance

where

$$\varepsilon(\boldsymbol{\theta}) \sim \mathcal{GP}\left(0, \sigma^2 c(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q})\right).$$

We explain each of the components of this approximation and the construction of an emulator in more detail below.

By running the computer model relatively few times we can generate a set of model outputs (y_i^d's) for a set of **design points** ($\boldsymbol{\theta}_i^d$)

$$\begin{aligned} y_i^d &= f(\boldsymbol{\theta}_i^d) \quad \text{where} \quad i = 1, \ldots, n, \\ \mathbf{y}^d &= (y_1^d, \ldots, y_n^d)^T, \end{aligned} \tag{7.1}$$

where n is the number of design points. The design points are chosen such that they are spread to adequately cover $\boldsymbol{\Theta}$, the input space of $\boldsymbol{\theta}$. McKay et al. (1979) propose the use of Latin hypercube sampling. Suppose $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$ and we wish to draw n random values for $\boldsymbol{\theta}$. For $j = 1, \ldots, p$, we divide the sample space of θ_j into n regions of equal marginal probability. This requires the specification of a prior for $\boldsymbol{\theta}$ in advance and this prior should not be too diffuse. Here we simply use a uniform prior within a fixed range. We then draw one random value of θ_i from each region. This process is repeated, sampling without replacement for $i = 1, \ldots, n$. This ensures that each dimension of the input space is better represented than sampling from a uniform distribution for each input parameter (θ_j) independently.

As mentioned earlier, we approximate $f(\boldsymbol{\theta})$ by a Gaussian process

$$\begin{aligned} y = f_a(\boldsymbol{\theta}|\boldsymbol{\beta}, \sigma_e^2, \mathbf{Q}) &\sim \mathcal{GP}\left(m(\boldsymbol{\theta}|\boldsymbol{\beta}), \sigma_e^2 c(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q})\right), \\ \mathbb{E}[y] = \mathbb{E}\left[f_a(\boldsymbol{\theta}|\boldsymbol{\beta}, \sigma_e^2, \mathbf{Q})\right] &= m(\boldsymbol{\theta}|\boldsymbol{\beta}) = \mathbf{h}(\boldsymbol{\theta})^T \boldsymbol{\beta}, \end{aligned} \tag{7.2}$$

conditional on the unknown vector of coefficients $\boldsymbol{\beta} = \{\beta_k\}_{k=1}^q$. The vector $\mathbf{h}(\cdot)$ is referred to as the **basis function** and consists of q known regression functions of $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)^T$. The choice of $\mathbf{h}(\cdot)$ should incorporate any knowledge we have about $f(\cdot)$. A common choice is simply the set of linear functions $\mathbf{h}(\boldsymbol{\theta}) = (1, \theta_1, \ldots, \theta_p)^T$, but other functions of $\boldsymbol{\theta}$ may be chosen[2].

[2]Note that we do not change the symbol y here or in the following pages (i.e. when stating $y = f_a(\boldsymbol{\theta})$, $y = f_u(\boldsymbol{\theta})$ or $y = f_c(\boldsymbol{\theta})$) because we are sequentially building up the way in which we emulate y.

We now consider how we expect the approximation $\mathbf{h}(\boldsymbol{\theta})^T\boldsymbol{\beta}$ to deviate from $f(\boldsymbol{\theta})$. By the definition of the Gaussian process the covariance between $f_a(\boldsymbol{\theta}|\boldsymbol{\beta}, \sigma_e^2, \mathbf{Q})$ and $f_a(\boldsymbol{\theta}'|\boldsymbol{\beta}, \sigma_e^2, \mathbf{Q})$ is given by

$$\mathbb{C}\left[f_a(\boldsymbol{\theta}|\boldsymbol{\beta}, \sigma_e^2, \mathbf{Q}), f_a(\boldsymbol{\theta}'|\boldsymbol{\beta}, \sigma_e^2, \mathbf{Q})\right] = \sigma_e^2 c(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q}), \tag{7.3}$$

conditional on a constant variance parameter σ_e^2, where $c(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q})$ is a **correlation function** that measures the correlation between $f(\cdot)$ at $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ conditional on a set of roughness parameters ($\mathbf{Q}$). The function $c(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q})$ decreases as $|\boldsymbol{\theta} - \boldsymbol{\theta}'|$ increases, satisfies $c(\boldsymbol{\theta}, \boldsymbol{\theta}|\mathbf{Q}) = 1\ \forall \boldsymbol{\theta}$, and must ensure that the covariance matrix of any set of outputs $\{y_1 = f(\boldsymbol{\theta}_1), \ldots, y_n = f(\boldsymbol{\theta}_n)\}$ is positive semi-definitive. A typical choice is

$$c(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q}) = \exp\left(-(\boldsymbol{\theta} - \boldsymbol{\theta}')^T\mathbf{Q}(\boldsymbol{\theta} - \boldsymbol{\theta}')\right), \tag{7.4}$$

where $\mathbf{Q}$ is a $p \times p$ diagonal matrix of **roughness scales**[3]. This form has the advantage that $f(\cdot)$ has derivatives of all orders; other forms for the covariance function may not have this desirable property (Oakley 1999).

The Gaussian process (Equation 7.2) implies that on the design points our data can be approximated by

$$\mathbf{y}^d \sim \mathcal{N}\left(\mathbf{H}\boldsymbol{\beta}, \sigma_e^2\mathbf{A}\right), \tag{7.5}$$

where

$$\mathbf{H}^T = \left(\mathbf{h}\left(\boldsymbol{\theta}_1^d\right), \ldots, \mathbf{h}\left(\boldsymbol{\theta}_n^d\right)\right), \tag{7.6}$$

$$\mathbf{A}_{i,j} = c\left(\boldsymbol{\theta}_i^d, \boldsymbol{\theta}_j^d|\mathbf{Q}\right), \tag{7.7}$$

with $\mathbf{A}_{i,j} = \mathbf{A}_{j,i}$ and $\mathbf{A}_{j,j} = 1$. More explicitly

$$\mathbf{A} = \begin{pmatrix} 1 & c\left(\boldsymbol{\theta}_1^d, \boldsymbol{\theta}_2^d|\mathbf{Q}\right) & \cdots & c\left(\boldsymbol{\theta}_1^d, \boldsymbol{\theta}_n^d|\mathbf{Q}\right) \\ c\left(\boldsymbol{\theta}_2^d, \boldsymbol{\theta}_1^d|\mathbf{Q}\right) & 1 & & \vdots \\ \vdots & & \ddots & \\ c\left(\boldsymbol{\theta}_n^d, \boldsymbol{\theta}_1^d|\mathbf{Q}\right) & \cdots & & 1 \end{pmatrix}.$$

[3] Vernon et al. (2010) provides an example of an alternative formulation

$$c(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q}) = \exp\left(-\left|\left|\boldsymbol{\theta} - \boldsymbol{\theta}'\right|\right|^2/\mathbf{Q}^2\right)$$

that we do not use here.

We then **calibrate** the approximation by finding the restricted maximum likelihood (REML) estimates of $\boldsymbol{\beta}$ and σ_e^2 using standard methods (Diggle et al. 2002, Marin & Robert 2010, Patterson & Thompson 1971)

$$\widehat{\boldsymbol{\beta}}\left(\boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right) = \left(\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{A}^{-1}\mathbf{y}^d, \tag{7.8}$$

$$\widehat{\sigma}_e^2\left(\boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right) = \frac{\left(\mathbf{y}^d\right)^T\left(\mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{H}\left(\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{A}^{-1}\right)\mathbf{y}^d}{n-q}. \tag{7.9}$$

Using weak prior distributions for $\boldsymbol{\beta}$ and σ_e^2

$$\begin{aligned}\pi\left(\boldsymbol{\beta}\right) &\propto 1,\\ \pi\left(\sigma_e^2\right) &\propto \frac{1}{\sigma_e^2},\end{aligned}$$

or

$$\pi\left(\boldsymbol{\beta}, \sigma_e^2\right) \propto \sigma_e^{-2}, \tag{7.10}$$

and combining with Equation 7.5 using Bayes' theorem we derive the posterior[4]. First we define the independent residual projection matrix ($\mathbf{G}$)

$$\begin{aligned}\mathbf{G} &= \mathbf{I} - \mathbf{H}\left(\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{A}^{-1}\\ \mathbf{GH} &= \mathbf{H} - \mathbf{H}\left(\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H} = 0\\ \widehat{\mathbf{e}} &= \mathbf{G}\mathbf{y}^d = \mathbf{y}^d - \mathbf{H}\widehat{\boldsymbol{\beta}},\end{aligned}$$

from which it follows that

$$\begin{aligned}\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta} &= \mathbf{y}^d - \mathbf{H}\widehat{\boldsymbol{\beta}} + \mathbf{H}\left(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}\right)\\ &= \mathbf{y}^d - \mathbf{H}\left(\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{A}^{-1}\mathbf{y}^d + \mathbf{H}\left(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}\right)\\ &= \mathbf{G}\mathbf{y}^d + \mathbf{H}\left(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}\right)\\ \left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right)^T\mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right) &= \left(\mathbf{G}\mathbf{y}^d + \mathbf{H}\left(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}\right)\right)^T\mathbf{A}^{-1}\left(\mathbf{G}\mathbf{y}^d + \mathbf{H}\left(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}\right)\right)\\ &= \mathbf{y}^d\mathbf{G}^T\mathbf{A}^{-1}\mathbf{G}\mathbf{y}^d + \left(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right)^T\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\left(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right)\\ &\quad + \left(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right)^T\mathbf{H}^T\mathbf{A}^{-1}\mathbf{G}\mathbf{y}^d\\ &= \widehat{\mathbf{e}}^T\mathbf{A}^{-1}\widehat{\mathbf{e}} + \left(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right)^T\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\left(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}\right) + 0,\end{aligned}$$

[4]Note that the prior $\pi\left(\mathbf{Q}\right)$ is left unspecified.

and

$$\widehat{\sigma}_e^2 = \frac{\widehat{\mathbf{e}}^T \mathbf{A}^{-1} \widehat{\mathbf{e}}}{n-q}$$

The posterior for $\boldsymbol{\beta}$, σ_e^2 and $\mathbf{Q}$ given the design data $\mathbf{y}^d$ is therefore

$$\begin{aligned}
p\left(\boldsymbol{\beta}, \sigma_e^2, \mathbf{Q}|\mathbf{y}^d\right) &\propto p\left(\mathbf{y}^d|\boldsymbol{\beta}, \sigma_e^2, \mathbf{Q}\right) \pi(\boldsymbol{\beta}) \pi\left(\sigma_e^2\right) \pi\left(\mathbf{Q}\right) \\
&\propto \frac{1}{\sigma_e^2} p\left(\mathbf{y}^d|\boldsymbol{\beta}, \sigma_e^2, \mathbf{Q}\right) \pi\left(\mathbf{Q}\right) \quad \text{using Equation 7.10} \\
&\propto \frac{1}{\sigma_e^2} (2\pi\sigma_e^2)^{-\frac{n}{2}} |\mathbf{A}|^{-\frac{1}{2}} \exp\left[-\frac{1}{2\sigma_e^2}\left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right)^T \mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right)\right] \pi\left(\mathbf{Q}\right) \\
&\qquad \text{using Equation 7.5} \\
&\propto \left(\sigma_e^2\right)^{-\left(\frac{n+2}{2}\right)} |\mathbf{A}|^{-\frac{1}{2}} \exp\left[-\frac{1}{2\sigma_e^2}\left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right)^T \mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right)\right] \pi\left(\mathbf{Q}\right) \\
&\propto \left(\sigma_e^2\right)^{-\left(\frac{n+2}{2}\right)} |\mathbf{A}|^{-\frac{1}{2}} \exp\left[-\frac{1}{2\sigma_e^2}\widehat{\mathbf{e}}^T \mathbf{A}^{-1}\widehat{\mathbf{e}} + \left(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}\right)^T \mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\left(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}\right)\right] \pi\left(\mathbf{Q}\right) \\
&\propto p\left(\boldsymbol{\beta}|\sigma_e^2, \mathbf{Q}, \mathbf{y}^d\right) p\left(\sigma_e^2|\mathbf{Q}, \mathbf{y}^d\right) p\left(\mathbf{Q}|\mathbf{y}^d\right) \\
&\propto \left(2\pi\sigma_e^2\right)^{-\frac{q}{2}} \left|\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right|^{\frac{1}{2}} \exp\left[\frac{1}{2\sigma_e^2}\left(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}\right)^T \mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\left(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}\right)\right] \\
&\qquad \times \left(\sigma_e^2\right)^{-\left(\frac{n-q}{2}\right)-1} \exp\left[-\frac{1}{2\sigma_e^2}\widehat{\mathbf{e}}^T\mathbf{A}^{-1}\widehat{\mathbf{e}}\right] \frac{\left(\frac{1}{2}\widehat{\mathbf{e}}^T\mathbf{A}^{-1}\widehat{\mathbf{e}}\right)^{\frac{n-q}{2}}}{\Gamma\left(\frac{n-q}{2}\right)} \\
&\qquad \times |\mathbf{A}|^{-\frac{1}{2}} |\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}|^{-\frac{1}{2}} \left(\frac{1}{2}\widehat{\mathbf{e}}^T\mathbf{A}^{-1}\widehat{\mathbf{e}}\right)^{-\left(\frac{n-q}{2}\right)} \pi\left(\mathbf{Q}\right).
\end{aligned}$$

This can be split up into the components that we are interested in

$$p\left(\boldsymbol{\beta}|\sigma_e^2, \mathbf{Q}, \mathbf{y}^d\right) = \left(2\pi\sigma_e^2\right)^{-\frac{q}{2}} |\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}|^{\frac{1}{2}} \exp\left[-\frac{1}{2\sigma_e^2}\left(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}\right)^T \mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\left(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}\right)\right], \tag{7.11}$$

$$p\left(\sigma_e^2|\mathbf{Q}, \mathbf{y}^d\right) = \frac{\left(\frac{1}{2}\widehat{\mathbf{e}}^T\mathbf{A}^{-1}\widehat{\mathbf{e}}\right)^{\frac{n-q}{2}}}{\Gamma\left(\frac{n-q}{2}\right)} \left(\sigma_e^2\right)^{-\frac{n-q}{2}-1} e^{-\frac{1}{\sigma_e^2}\left(-\frac{1}{2}\widehat{\mathbf{e}}^T\mathbf{A}^{-1}\widehat{\mathbf{e}}\right)}, \tag{7.12}$$

$$p\left(\mathbf{Q}|\mathbf{y}^d\right) \propto \left(\widehat{\sigma}_e^2\right)^{-\frac{n-q}{2}} |\mathbf{A}|^{-\frac{1}{2}} \left|\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right|^{-\frac{1}{2}} \pi\left(\mathbf{Q}\right). \tag{7.13}$$

Equation 7.13 defines the posterior distribution for the roughness scales ($\mathbf{Q}$). The roughness scales are known to be difficult to estimate (Kennedy & O'Hagan 2000, Oakley 2004). In practice, the roughness scales are either fixed by assumption, or are estimated by maximising, over allowable scales, the posterior assuming a uniform prior $\pi(\mathbf{Q}) \propto 1$

$$f(\mathbf{Q}|\mathbf{y}^d) \propto \left(\widehat{\sigma}_e^2\right)^{-\frac{n-q}{2}} |\mathbf{A}|^{-\frac{1}{2}} \left|\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right|^{-\frac{1}{2}}. \tag{7.14}$$

Therefore, the Bayesian posteriors for $\boldsymbol{\beta}$ and σ_e^2, conditional on $\boldsymbol{\theta}^d$, $\mathbf{y}^d$ and the roughness scales ($\mathbf{Q}$), are

$$\boldsymbol{\beta}|\sigma_e^2, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q} \sim \mathcal{N}\left(\widehat{\boldsymbol{\beta}}, \sigma_e^2 \left(\mathbf{H}^T \mathbf{A}^{-1} \mathbf{H}\right)^{-1}\right), \tag{7.15}$$

$$\sigma_e^2|\boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q} \sim \mathcal{IG}\left(\frac{n-q}{2}, \frac{1}{2}\widehat{\mathbf{e}}^T \mathbf{A}^{-1} \widehat{\mathbf{e}}\right). \tag{7.16}$$

Next, we construct an approximation $f_u\left(\boldsymbol{\theta}_i^d|\boldsymbol{\beta}, \sigma_e^2, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right)$ constrained such that $y_i^d = f_u\left(\boldsymbol{\theta}_i^d|\boldsymbol{\beta}, \sigma_e^2, \boldsymbol{\theta}^d, \mathbf{Q}\right)$ and $\mathbb{V}\left[y_i^d\right] = 0$ given the true values of $\boldsymbol{\beta}$ and σ_e^2, the design inputs ($\boldsymbol{\theta}^d$), and the roughness matrix ($\mathbf{Q}$). This is accomplished by considering the joint distribution of $\mathbf{y}^d$ and a further observation y and input $\boldsymbol{\theta}'$

$$\begin{bmatrix} y \\ \mathbf{y}^d \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{h}(\boldsymbol{\theta})^T\boldsymbol{\beta} \\ \mathbf{H}\boldsymbol{\beta} \end{bmatrix}, \begin{bmatrix} \sigma^2 & \sigma^2\mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q})^T \\ \sigma^2\mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q})^T & \sigma^2\mathbf{A} \end{bmatrix}\right),$$

where

$$\mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q})^T = \left(c(\boldsymbol{\theta}, \boldsymbol{\theta}_1^d|\mathbf{Q}), \ldots, c(\boldsymbol{\theta}, \boldsymbol{\theta}_n^d|\mathbf{Q})\right). \tag{7.17}$$

A standard result for multivariate normal distributions (Theorem 4.4 in Rencher 2000) states that if

$$\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \begin{bmatrix} \Sigma_{1,1} & \Sigma_{1,2} \\ \Sigma_{2,1} & \Sigma_{2,2} \end{bmatrix}\right),$$

then

$$\mathbf{x}_1|\mathbf{x}_2 \sim \mathcal{N}\left(\boldsymbol{\mu}_1 + \Sigma_{1,2}\Sigma_{2,2}^{-1}\left(\mathbf{x}_2 - \boldsymbol{\mu}_2\right), \Sigma_{1,1} - \Sigma_{1,2}\Sigma_{2,2}^{-1}\Sigma_{2,1}\right).$$

Therefore

$$y|\boldsymbol{\theta}, \boldsymbol{\theta}^d, \mathbf{y}^d, \boldsymbol{\beta}, \sigma^2, \mathbf{Q} \sim \mathcal{GP}\left(\mathbf{h}(\boldsymbol{\theta})^T\boldsymbol{\beta} - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q})^T\mathbf{A}^{-1}\left(\mathbf{y}^d + \mathbf{H}\boldsymbol{\beta}\right), \right.$$
$$\left. \sigma^2\left(1 - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q})^T\mathbf{A}^{-1}\mathbf{t}(\boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q})^T\right)\right).$$

If we define

$$m^*\left(\boldsymbol{\theta}|\boldsymbol{\beta}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right) = \mathbf{h}(\boldsymbol{\theta})^T\boldsymbol{\beta} + \mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T\mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right), \tag{7.18}$$

$$c^*\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right) = c\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q}\right) - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q})^T\mathbf{A}^{-1}\mathbf{t}(\boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}), \tag{7.19}$$

then we can write

$$y = f_u\left(\boldsymbol{\theta}|\boldsymbol{\beta}, \sigma_e^2, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right) \sim \mathcal{GP}\left(m^*\left(\boldsymbol{\theta}|\boldsymbol{\beta}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right), \sigma_e^2 c^*\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right)\right). \tag{7.20}$$

We now verify that if $\boldsymbol{\theta} = \boldsymbol{\theta}_i^d$ then $\mathbb{E}\left[f_u\left(\boldsymbol{\theta}_i^d|\boldsymbol{\beta}, \sigma_e^2, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right)\right] = \mathbf{y}_i^d$ and $\mathbb{V}\left[f_u\left(\boldsymbol{\theta}_i^d|\boldsymbol{\beta}, \sigma_e^2, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right)\right] = 0$ (i.e. the emulator matches the simulator exactly and the variance is zero)

$$\begin{aligned}
\mathbb{E}\left[f_u\left(\boldsymbol{\theta}_i^d|\boldsymbol{\beta}, \sigma_e^2, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right)\right] &= m^*\left(\boldsymbol{\theta}_i^d|\boldsymbol{\beta}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right) \\
&= \mathbf{h}(\boldsymbol{\theta}_i^d)^T\boldsymbol{\beta} + \mathbf{t}\left(\boldsymbol{\theta}_i^d|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T \mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right) \\
&= \mathbf{h}(\boldsymbol{\theta}_i^d)^T\boldsymbol{\beta} + \boldsymbol{\delta}_i^T\left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right) \\
&= \mathbf{h}(\boldsymbol{\theta}_i^d)^T\boldsymbol{\beta} + \boldsymbol{\delta}_i^T\mathbf{y}^d - \boldsymbol{\delta}_i^T\mathbf{H}\boldsymbol{\beta} \\
&= \mathbf{h}(\boldsymbol{\theta}_i^d)^T\boldsymbol{\beta} + \mathbf{y}_i^d - \mathbf{h}(\boldsymbol{\theta}_i^d)^T\boldsymbol{\beta} \\
&= \mathbf{y}_i^d,
\end{aligned}$$

and

$$\begin{aligned}
\mathbb{V}\left[f_u\left(\boldsymbol{\theta}_i^d|\boldsymbol{\beta}, \sigma_e^2, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right)\right] &= \sigma_e^2 c^*\left(\boldsymbol{\theta}_i^d, \boldsymbol{\theta}_i^d|\boldsymbol{\theta}^d, \mathbf{Q}\right) \\
&= \sigma_e^2\left(c\left(\boldsymbol{\theta}_i^d, \boldsymbol{\theta}_i^d|\mathbf{Q}\right) - \mathbf{t}\left(\boldsymbol{\theta}_i^d|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T \mathbf{A}^{-1}\mathbf{t}\left(\boldsymbol{\theta}_i^d|\boldsymbol{\theta}^d, \mathbf{Q}\right)\right) \\
&= \sigma_e^2\left(1 - \boldsymbol{\delta}_i^T\mathbf{t}(\boldsymbol{\theta}_i^d|\boldsymbol{\theta}^d, \mathbf{Q})\right) = \sigma_e^2\left(1 - c\left(\boldsymbol{\theta}_i^d, \boldsymbol{\theta}_i^d|\mathbf{Q}\right)\right) \\
&= \sigma_e^2(1-1) \\
&= 0.
\end{aligned}$$

Here $\boldsymbol{\delta}_i^T$ is a vector of length n containing all zeros except that element i is equal to one. In words, both Equations 7.18 and 7.19 consist of two components. In Equation 7.18, the first component $\mathbf{h}(\boldsymbol{\theta})^T\boldsymbol{\beta}$ is our prior expectation of $f(\cdot)$, which conditional on $\boldsymbol{\beta}$ is $\mathbf{h}(\boldsymbol{\theta})^T\boldsymbol{\beta}$. The expected value of $\boldsymbol{\beta}$ has been updated in light of the outputs $\mathbf{y}^d$. The second component $\mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T \mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right)$ adjusts the posterior mean so that it passes exactly through all of the outputs (i.e. if we have observed the output $y_i^d = f_u\left(\boldsymbol{\theta}_i^d|\boldsymbol{\beta}, \sigma_e^2, \boldsymbol{\theta}^d, \mathbf{Q}\right)$, then $\mathbb{E}\left[f_u\left(\boldsymbol{\theta}_i^d|\boldsymbol{\beta}, \sigma_e^2, \boldsymbol{\theta}^d, \mathbf{Q}\right)\right] = y_i^d$). How smoothly $m^*(\cdot)$ departs from $\mathbf{h}(\boldsymbol{\theta})^T\boldsymbol{\beta}$ towards any observed output y_i^d for $\boldsymbol{\theta}'$ close to $\boldsymbol{\theta}_i^d$ will depend on $\mathbf{Q}$. In Equation 7.19, the first component $c(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q})$ is the correlation function. The second component, $\mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q})^T\mathbf{A}^{-1}\mathbf{t}(\boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q})$ adjusts the variance so that it is equal to zero if $f(\boldsymbol{\theta})$ is known.

Finally, we combine Equations 7.15 and 7.20 and integrate out $\boldsymbol{\beta}$, using its posterior (Equation 7.15), to construct the **emulator**

$$\begin{aligned}
\mathbb{E}\left[y|\mathbf{y}^d,\mathbf{Q}\right] &= \mathbb{E}\left[\mathbb{E}\left[y|\boldsymbol{\beta},\sigma_e^2,\mathbf{y}^d,\mathbf{Q}\right]\left|\mathbf{y}^d,\mathbf{Q}\right.\right] \\
&= \mathbb{E}\left[m^*\left(\boldsymbol{\theta}\right)\left|\mathbf{y}^d,\mathbf{Q}\right.\right] \\
&= \mathbb{E}\left[\mathbf{h}\left(\boldsymbol{\theta}\right)^T\boldsymbol{\beta} - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right)\right] \\
&= \mathbf{h}\left(\boldsymbol{\theta}\right)^T\mathbb{E}\left[\boldsymbol{\beta}\right] - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\mathbb{E}\left[\boldsymbol{\beta}\right]\right) \\
&= \mathbf{h}\left(\boldsymbol{\theta}\right)^T\widehat{\boldsymbol{\beta}} - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\widehat{\boldsymbol{\beta}}\right), \\
\mathbb{V}\left[y|\mathbf{y}^d,\mathbf{Q}\right] &= \mathbb{E}\left[\mathbb{V}\left[y|\boldsymbol{\beta},\sigma_e^2,\mathbf{y}^d,\mathbf{Q}\right]\left|\mathbf{y}^d,\mathbf{Q}\right.\right] + \mathbb{V}\left[\mathbb{E}\left[y|\boldsymbol{\beta},\sigma_e^2,\mathbf{y}^d,\mathbf{Q}\right]\left|\mathbf{y}^d,\mathbf{Q}\right.\right] \\
&= \mathbb{E}\left[\sigma_e^2\left(c\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\mathbf{Q}\right) - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\mathbf{t}(\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q})\right)\left|\mathbf{y}^d,\mathbf{Q}\right.\right] \\
&\quad + \mathbb{V}\left[\left[\mathbf{h}\left(\boldsymbol{\theta}\right)^T\boldsymbol{\beta} + \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right)\right]\left|\mathbf{y}^d,\mathbf{Q}\right.\right] \\
&= \mathbb{E}\left[\sigma_e^2|\mathbf{y}^d,\mathbf{Q}\right]\left(c\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\mathbf{Q}\right) - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\mathbf{t}(\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q})\right) \\
&\quad + \mathbb{V}\left[\left(\mathbf{h}\left(\boldsymbol{\theta}\right)^T - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\mathbf{H}\right)\boldsymbol{\beta} + \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{y}^d\left|\mathbf{y}^d,\mathbf{Q}\right.\right] \\
&= \widehat{\sigma}_e^2\left(c\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\mathbf{Q}\right) - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\mathbf{t}(\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q})\right) \\
&\quad + \left(\mathbf{h}\left(\boldsymbol{\theta}\right)^T - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\mathbf{H}\right)\mathbb{V}\left[\boldsymbol{\beta}|\mathbf{y}^d,\mathbf{Q}\right]\left(\mathbf{h}\left(\boldsymbol{\theta}\right)^T - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\mathbf{H}\right)^T \\
&= \widehat{\sigma}_e^2\left(c\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\mathbf{Q}\right) - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\mathbf{t}(\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q})\right) \\
&\quad + \left(\mathbf{h}\left(\boldsymbol{\theta}\right)^T - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\mathbf{H}\right)\left(\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right)^{-1}\left(\mathbf{h}\left(\boldsymbol{\theta}\right)^T - \mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q})^T\mathbf{A}^{-1}\mathbf{H}\right)^T
\end{aligned}$$

Therefore, we write

$$y \sim \mathcal{GP}\left(m^{**}\left(\boldsymbol{\theta}'|\widehat{\boldsymbol{\beta}},\boldsymbol{\theta}^d,\mathbf{y}^d,\mathbf{Q}\right),\widehat{\sigma}_e^2 c\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\mathbf{Q}\right)\right),$$

or

$$y = f_c\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}},\widehat{\sigma}_e^2,\boldsymbol{\theta}^d,\mathbf{y}^d,\mathbf{Q}\right) \sim \mathcal{GP}\left(m^{**}\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}},\boldsymbol{\theta}^d,\mathbf{y}^d,\mathbf{Q}\right),\widehat{\sigma}_e^2 c^{**}\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q}\right)\right), \tag{7.21}$$

where

$$m^{**}\left(\boldsymbol{\theta}'|\widehat{\boldsymbol{\beta}},\boldsymbol{\theta}^d,\mathbf{y}^d,\mathbf{Q}\right) = \mathbf{h}(\boldsymbol{\theta}')^T\widehat{\boldsymbol{\beta}} + \mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q}\right)^T\mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\widehat{\boldsymbol{\beta}}\right), \tag{7.22}$$

$$\begin{aligned}
c^{**}\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q}\right) = {} & c^*\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q}\right) + \left(\mathbf{h}(\boldsymbol{\theta})^T - \mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q}\right)^T\mathbf{A}^{-1}\mathbf{H}\right)\left(\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right)^{-1} \\
& \times \left(\mathbf{h}(\boldsymbol{\theta}')^T - \mathbf{t}\left(\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q}\right)^T\mathbf{A}^{-1}\mathbf{H}\right)^T.
\end{aligned} \tag{7.23}$$

To summarise, a Bayesian emulator provides an approximation of a computer model $f(\cdot)$. An emulator for $f(\cdot)$ is specified by a choice of regressors $\mathbf{h}(\boldsymbol{\theta})$ and a choice of correlation function $c(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q})$. The emulator is conditioned, or trained, using a relatively small set of evaluations of $f(\cdot)$, i.e. $\mathbf{y}^d = f(\boldsymbol{\theta}^d)$, given a carefully selected set of inputs $\boldsymbol{\theta}^d$. The conditioning process updates our estimates of the vector $\widehat{\boldsymbol{\beta}}$, the scalar $\widehat{\sigma}_c^2$ and is used to estimate the matrix $\mathbf{Q}$. The design points $\boldsymbol{\theta}^d$ and $\mathbf{y}^d$ are also used during evaluations of the emulator to interpolate (or extrapolate) a value for y' given a vector of inputs $\boldsymbol{\theta}'$.

Therefore, the construction of a Bayesian emulator involves the following steps:

1. Develop a sample design ($\boldsymbol{\theta}^d$)
2. Evaluate $\mathbf{y}^d = f(\boldsymbol{\theta}^d)$
3. Decide on a basis function ($\mathbf{h}(\boldsymbol{\theta})$)
4. Decide on a correlation function ($c(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q})$) and estimate the roughness scales ($\mathbf{Q}$)
5. Condition the emulator

The conditioned emulator may then be used for anything that the original function $f(\cdot)$ could be used for, including model evaluation, simulation, or inference.

Summary of deterministic univariate emulation

• model inputs/outputs

$$\mathbf{y}^d = (f(\boldsymbol{\theta}_1^d), \ldots, f(\boldsymbol{\theta}_n^d))^T,$$

• basis function evaluations

$$\mathbf{H}^T = \left(\mathbf{h}\left(\boldsymbol{\theta}_1^d\right), \ldots, \mathbf{h}\left(\boldsymbol{\theta}_n^d\right)\right),$$

• correlation matrix

$$\mathbf{A}_{i,j} = c\left(\boldsymbol{\theta}_i^d, \boldsymbol{\theta}_j^d|\mathbf{Q}\right),$$

• estimated regression coefficients

$$\widehat{\boldsymbol{\beta}}|\boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q} = \left(\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{A}^{-1}\mathbf{y}^d,$$

• estimated variance

$$\widehat{\sigma}_e^2|\boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q} = \frac{\left(\mathbf{y}^d\right)^T\left(\mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{H}\left(\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{A}^{-1}\right)\mathbf{y}^d}{n-q-2},$$

• a posteriori expectation

$$\begin{aligned}\mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q})^T &= \left(c(\boldsymbol{\theta}, \boldsymbol{\theta}_1^d|\mathbf{Q}), \ldots, c(\boldsymbol{\theta}, \boldsymbol{\theta}_n^d|\mathbf{Q})\right), \\ m^{**}\left(\boldsymbol{\theta}'|\widehat{\boldsymbol{\beta}}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right) &= \mathbf{h}(\boldsymbol{\theta}')^T\widehat{\boldsymbol{\beta}} + \mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T\mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\widehat{\boldsymbol{\beta}}\right),\end{aligned}$$

• a posteriori covariance

$$\begin{aligned}c^{**}\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right) = c^*\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right) &+ \left(\mathbf{h}(\boldsymbol{\theta})^T - \mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T\mathbf{A}^{-1}\mathbf{H}\right)\left(\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right)^{-1} \\ &\times \left(\mathbf{h}(\boldsymbol{\theta}')^T - \mathbf{t}\left(\boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T\mathbf{A}^{-1}\mathbf{H}\right)^T.\end{aligned}$$

7.2.1 A one-dimensional example

To illustrate some of the details of standard univariate Bayesian emulation, we begin with a simple one-dimensional example (i.e. a single input θ and a single output y). We define the function

$$f(\theta) \sim \begin{cases} 0 & \theta \leq 0.5 \\ 1 & \theta > 0.5 \end{cases}, \tag{7.24}$$

where $0 \leq \theta \leq 1$. We set $\boldsymbol{\theta}^d = \{\theta_i^d\}_{i=1}^n$ as 10 evenly spaced points from 0 to 1 (i.e. $n = 10$). Outputs are simulated from Equation 7.24 for $\mathbf{y}^d = \{y_i^d\}_{i=1}^n = f(\boldsymbol{\theta}^d)$. A set of four emulators are then conditioned on $\boldsymbol{\theta}^d$ and $\mathbf{y}^d$ by selecting $n = \{5, 7, 8, 10\}$ design points from the total of 10 above. The basis function $\mathbf{h}(\theta) = (1, \cos(\theta))$ is used for all four emulators. We would

not expect this basis function to do a very good job of emulating the step function above. The roughness scale is set a priori to be uninformative at $\mathbf{Q} = 100$.

The emulator is then used to make inference about $f(\theta')$ for $\theta' = 0, \ldots, 1$ (Figure 7.2). This example illustrates how the basis function ($\mathbf{h}(\theta)$) forms

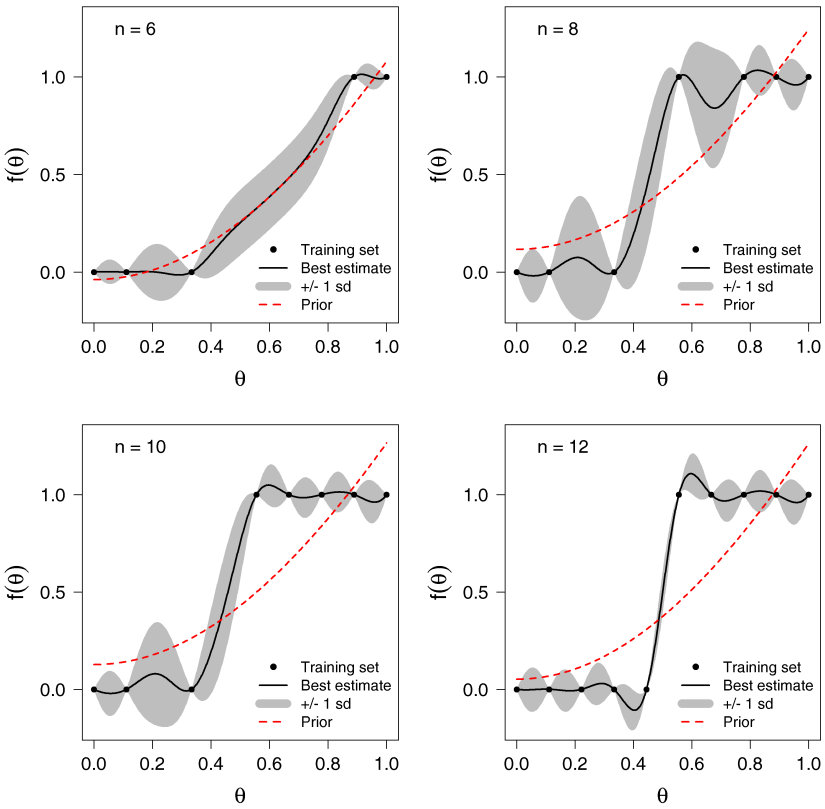


Figure 7.2: A sequence of emulators conditioned on $\boldsymbol{\theta}^d = \{\theta_i^d\}_{i=1}^n$ and $\mathbf{y}^d = \{y_i^d\}_{i=1}^n$ for $n = \{6, 8, 10, 12\}$ design points. In each plot $\boldsymbol{\theta}^d$ is the training set or model input plotted against the model output [•]. $m^{**}(\cdot)$ is the best estimate [solid line], ± 1 standard deviation [grey band], and $\mathbf{h}(\theta)\widehat{\boldsymbol{\beta}}$ is the prior [dashed red line].

the prior ($\mathbf{h}(\theta)\widehat{\boldsymbol{\beta}}$, shown as the dashed red line in Figure 7.2), and how this prior is updated as the number of inputs (n) are increased from $n = 6$ to $n = 12$. It also illustrates how the prior becomes less important as n is

increased and the coverage of the parameter space ($\boldsymbol{\Theta}$) is improved.

The expectation of $f(\cdot)$ is shown as the solid black line in Figure 7.2. The expectation of the emulator "sticks" to the prior in the absence of any design points (y_i^d's, see the top left plot), but threads exactly through any known design points. The variance or uncertainty produced by the emulator is shown ± 1 standard deviation about the expectation (the grey shaded region in Figure 7.2). As the number of design points (n) is increased, the variance tightens near known points (θ^d) and is higher in the absence of known points. The original emulator construction specifies that the variance about any known point is zero. This is clearly shown in the figure.

7.2.2 An example with a stochastic function

We now present an example that illustrates how standard univariate Bayesian emulation copes, when applied incorrectly, to a simple stochastic function. We use a very similar setup to the previous example. We define the function

$$f(\theta) \sim \begin{cases} & \mathcal{N}(0,1)/40 & \theta \leq 0.5 \\ 1+ & \mathcal{N}(0,1)/40 & \theta > 0.5 \end{cases}, \tag{7.25}$$

where $0 \leq \theta \leq 1$. We set $\boldsymbol{\theta}^d = \{\theta_i^d\}_{i=1}^n$ as 10 evenly spaced points from 0 to 1, but again repeated two of the points twice (i.e. $n = 12$). Outputs are simulated from Equation 7.25 for $\mathbf{y}^d = \{y_i^d\}_{i=1}^n = f(\boldsymbol{\theta}^d)$. A set of four emulators are then conditioned on $\boldsymbol{\theta}^d$ and $\mathbf{y}^d$ by selecting $n = \{6, 8, 10, 12\}$ design points from the total of 12 above. The uninformative basis function $\mathbf{h}(\theta) = (1, \cos(\theta))$ is used for all four emulators. The roughness scale is set a priori to be naive at $\mathbf{Q} = 100$.

Because the design matrix includes two instances of an identical input parameter ($\theta_1^d = \theta_2^d$) but different outputs ($y_1^d \neq y_2^d$), standard matrix inversion is no longer possible. Therefore, matrix inversion is done using the Moore-Penrose (MP) pseudo inverse.

> Sometimes it is not possible to invert the matrix $\mathbf{A}$ (or $\mathbf{A}_\gamma$) if the matrix is singular. This may arise because multiple outputs are observed

given the same inputs (i.e. $\boldsymbol{\theta}_i^d = \boldsymbol{\theta}_j^d$ but $y_i^d \neq y_j^d$). This cannot happen under a deterministic model, but it may occur if a model is stochastic. When this occurs the Moore-Penrose generalised pseudo-inverse of a matrix ($\mathbf{A}^+$) can be used to solve the matrix (Moore 1920, Penrose 1955). The Moore-Penrose pseudo-inverse has the following properties:

- If $\mathbf{A}$ is invertible, its pseudo-inverse is its inverse $\mathbf{A}^+ = \mathbf{A}^{-1}$
- The pseudo-inverse of the pseudo-inverse is the original matrix $(\mathbf{A}^+)^+ = \mathbf{A}$
- Pseudo-inversion commutes with transposition, conjugation, and taking the conjugate transpose $(\mathbf{A}^T)^+ = (\mathbf{A}^+)^T, \overline{\mathbf{A}}^+ = \overline{\mathbf{A}^+}, (\mathbf{A}^*)^+ = (\mathbf{A}^+)^*$.

The emulator is used to make inference about $f(\theta')$ for $\theta' = 0, \ldots, 1$ (Figure 7.3). As in the previous example, this example illustrates how the basis function ($\mathbf{h}(\theta)$) forms the prior ($\mathbf{h}(\theta)\widehat{\boldsymbol{\beta}}$, shown as the dashed red line in Figure 7.3), and how this prior is updated as the number of inputs (n) are increased from $n = 6$ to $n = 12$. It also illustrates how the prior becomes less important as n is increased and the coverage of the parameter space ($\boldsymbol{\Theta}$) is improved.

The expectation of $f(\cdot)$ is shown as the solid black line in Figure 7.3. Again, the expectation of the emulator "sticks" to the prior in the absence of any design points (y_i^d's, see the top left plot), but threads exactly through any known design points, or between any two design points if there are two inputs that are the same ($\theta_1^d = \theta_2^d$) that result in a different output ($y_1^d \neq y_2^d$), due to stochastic processes. The ability to thread the expectation between two points like this is only possible due to MP matrix inversion.

The variance or uncertainty produced by the emulator is shown ± 1 standard deviation about the expectation (the grey shaded region in Figure 7.3). As the number of design points (n) is increased, the variance tightens near known points (θ^d) and is higher in the absence of known points. The original emulator construction specifies that the variance

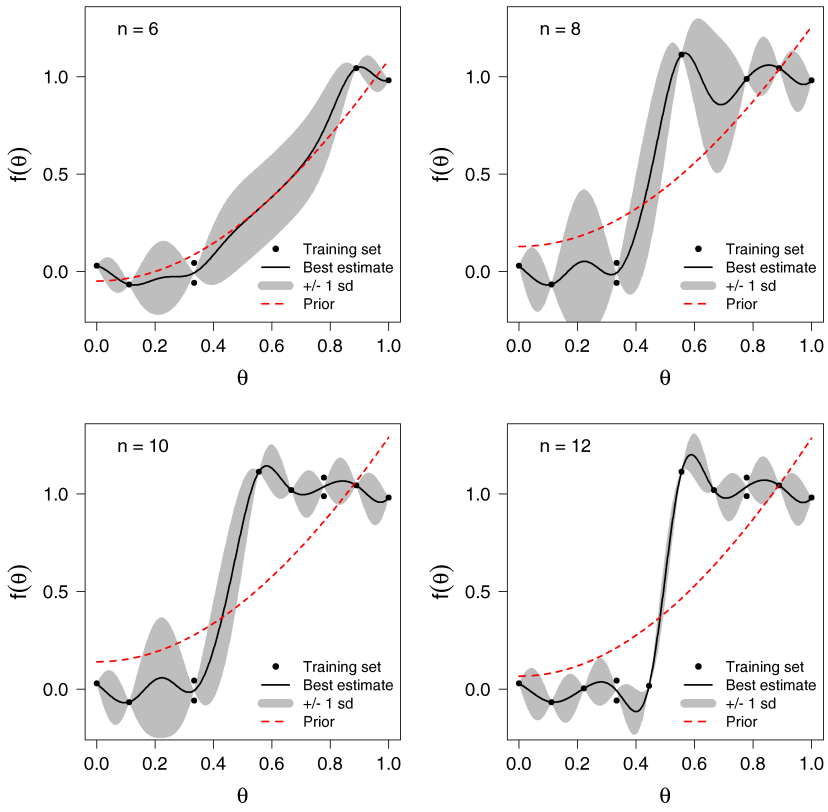


Figure 7.3: A sequence of emulators conditioned on $\boldsymbol{\theta}^d = \{\theta_i^d\}_{i=1}^n$ and $\mathbf{y}^d = \{y_i^d\}_{i=1}^n$ for $n = \{6, 8, 10, 12\}$ design points. In each plot $\boldsymbol{\theta}^d$ is the training set or model input plotted against the model output [•]. $m^{**}(\cdot)$ is the best estimate [solid line], ± 1 standard deviation [grey band], and $\mathbf{h}(\theta)\widehat{\boldsymbol{\beta}}$ is the prior [dashed red line].

about any known point is zero. This is clearly shown in the figure, but is obviously wrong if emulating stochastic models as is the case here. Notice that the variance about points where there are two inputs that are the same resulting in a different output is also zero. Therefore, while MP matrix inversion allows us to solve the linear system, and gives us the correct expected value between the two known outputs, the emulated variance about these two points is inadequate for stochastic models under the previous Bayesian emulation construction.

7.3 Stochasticity in emulators

In the previous section we introduced deterministic univariate Bayesian emulators. However, it was shown that these deterministic emulators do not adequately deal with stochasticity. Here we extend the Bayesian emulation framework beyond what is currently described in the literature (that we are aware of), and introduce stochastic Bayesian emulators. That is, they can be used to emulate stochastic computer models.

We represent a stochastic computer model as some function $f(\cdot)$ and the error of this model to be $u(\cdot)$, thus

$$y = f(\boldsymbol{\theta}) + u(\boldsymbol{\theta}), \tag{7.26}$$

where we assume this error to be normally distributed

$$u(\boldsymbol{\theta}) \overset{\text{iid}}{\sim} \mathcal{N}\left(0, \sigma_u^2(\boldsymbol{\theta})\right).$$

We will assume here that σ_u^2 is a constant. As before, we generate a training set or set of design points ($\boldsymbol{\theta}^d$) for $\boldsymbol{\theta}_i^d$ where $i = 1, \ldots, n$ and evaluate our stochastic computer model at these points

$$y_i^d = f\left(\boldsymbol{\theta}_i^d\right) + u\left(\boldsymbol{\theta}_i^d\right). \tag{7.27}$$

We approximate $f(\boldsymbol{\theta})$ by a Gaussian process

$$y_i^d = \mathbf{h}\left(\boldsymbol{\theta}_i^d\right)^T \boldsymbol{\beta} + \varepsilon_i\left(\boldsymbol{\theta}_i^d\right) + u_i\left(\boldsymbol{\theta}^d\right), \tag{7.28}$$

$$\mathbf{y}^d = \mathbf{H}\boldsymbol{\beta} + \boldsymbol{\varepsilon}^d + \mathbf{u}^d, \tag{7.29}$$

where $\varepsilon(\boldsymbol{\theta}) \overset{\text{iid}}{\sim} \mathcal{GP}(0, \sigma_e^2 c(\boldsymbol{\theta}, \boldsymbol{\theta}', \mathbf{Q}))$. The variance of $\mathbf{y}^d$ is

$$\mathbb{V}[\mathbf{y}^d] = \mathbb{V}[\boldsymbol{\varepsilon}^d] + \mathbb{V}[\mathbf{u}^d],$$

where $\mathbb{V}[\boldsymbol{\varepsilon}^d] = \sigma_e^2 \mathbf{A}$, $\mathbb{V}[\mathbf{u}^d] = \sigma_u^2 \mathbf{I}$, $\mathbf{A}$ is defined in Equation 7.2, and $\mathbf{I}$ is an $n \times n$ identity matrix. Therefore, $\mathbb{V}[\mathbf{y}^d]$ can be written

$$\begin{aligned}
\mathbb{V}[\mathbf{y}^d] &= \mathbb{V}[\boldsymbol{\varepsilon}^d] + \mathbb{V}[\mathbf{u}^d] \\
&= \sigma_e^2 \mathbf{A} + \sigma_u^2 \mathbf{I} \\
&= \sigma_e^2 \left(\mathbf{A} + \gamma \mathbf{I}\right) \quad \text{where } \gamma = \frac{\sigma_u^2}{\sigma_e^2} \\
&= \sigma_e^2 \mathbf{A}_\gamma \quad \text{where } \mathbf{A}_\gamma = \mathbf{A} + \gamma \mathbf{I}.
\end{aligned} \tag{7.30}$$

So we have

$$\mathbf{y}^d \sim \mathcal{N}\left(\mathbf{H}\boldsymbol{\beta}, \sigma_e^2 \mathbf{A}_\gamma\right). \tag{7.31}$$

For a given γ we can calculate the restricted maximum likelihood (REML) estimates of $\boldsymbol{\beta}$ and σ_e^2 (Diggle et al. 2002, Patterson & Thompson 1971)

$$\widehat{\boldsymbol{\beta}}_\gamma | \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \gamma = \left(\mathbf{H}^T \mathbf{A}_\gamma^{-1} \mathbf{H}\right)^{-1} \mathbf{H}^T \mathbf{A}_\gamma^{-1} \mathbf{y}^d, \tag{7.32}$$

$$\widehat{\sigma}_\gamma^2 | \boldsymbol{\theta}^d, \mathbf{y}^d, \widehat{\boldsymbol{\beta}}_\gamma, \mathbf{Q}, \gamma = \frac{\left(\mathbf{y} - \mathbf{H}\widehat{\boldsymbol{\beta}}_\gamma\right)^T \mathbf{A}_\gamma^{-1} \left(\mathbf{y} - \mathbf{H}\widehat{\boldsymbol{\beta}}_\gamma\right)}{n - q}, \tag{7.33}$$

We then search to find the γ that maximises the log-likelihood

$$L^*(\gamma) = -\frac{1}{2} \log |\mathbf{A}_\gamma| - \frac{1}{2} \log \left|\mathbf{H}^T \mathbf{A}_\gamma^{-1} \mathbf{H}\right| - \frac{n-q}{2} \log(SSE), \tag{7.34}$$

where

$$SSE = \left(\mathbf{y} - \mathbf{H}\widehat{\boldsymbol{\beta}}_\gamma\right)^T \mathbf{A}_\gamma^{-1} \left(\mathbf{y} - \mathbf{H}\widehat{\boldsymbol{\beta}}_\gamma\right).$$

Given $\widehat{\boldsymbol{\beta}}_\gamma$, $\widehat{\sigma}_\gamma^2$ and γ, we proceed with the construction of a univariate emulator that incorporates stochasticity. We construct the approximation

$$y = f_u\left(\boldsymbol{\theta} | \boldsymbol{\beta}, \sigma_e^2, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \gamma\right) \sim \mathcal{GP}\left(m^*\left(\boldsymbol{\theta} | \boldsymbol{\beta}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \gamma\right), \sigma_e^2 c^*\left(\boldsymbol{\theta}, \boldsymbol{\theta}' | \boldsymbol{\theta}^d, \mathbf{Q}, \gamma\right)\right), \tag{7.35}$$

where

$$m^*\left(\boldsymbol{\theta} | \boldsymbol{\beta}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \gamma\right) = \mathbf{h}(\boldsymbol{\theta})^T \boldsymbol{\beta} + \mathbf{t}\left(\boldsymbol{\theta} | \boldsymbol{\theta}^d, \mathbf{Q}\right)^T \mathbf{A}_\gamma^{-1} \left(\mathbf{y}^d - \mathbf{H}\boldsymbol{\beta}\right), \tag{7.36}$$

$$c^*\left(\boldsymbol{\theta}, \boldsymbol{\theta}' | \boldsymbol{\theta}^d, \mathbf{Q}, \gamma\right) = c\left(\boldsymbol{\theta}, \boldsymbol{\theta}' | \mathbf{Q}\right) - \mathbf{t}(\boldsymbol{\theta} | \boldsymbol{\theta}^d, \mathbf{Q})^T \mathbf{A}_\gamma^{-1} \mathbf{t}(\boldsymbol{\theta}' | \boldsymbol{\theta}^d, \mathbf{Q}), \tag{7.37}$$

$$\mathbf{t}(\boldsymbol{\theta} | \boldsymbol{\theta}^d, \mathbf{Q})^T = \left(c(\boldsymbol{\theta}, \boldsymbol{\theta}_1^d | \mathbf{Q}), \ldots, c(\boldsymbol{\theta}, \boldsymbol{\theta}_n^d | \mathbf{Q})\right).$$

Finally, we construct the **emulator**

$$y = f_c\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}}_\gamma, \widehat{\sigma}^2_\gamma, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \gamma\right) \sim \mathcal{GP}\left(m^{**}\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}}_\gamma, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \gamma\right), \widehat{\sigma}^2_\gamma c^{**}\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}, \gamma\right)\right), \tag{7.38}$$

where

$$m^{**}\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}}_\gamma, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \gamma\right) = \mathbf{h}(\boldsymbol{\theta})^T\widehat{\boldsymbol{\beta}}_\gamma + \mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T \mathbf{A}_\gamma^{-1}\left(\mathbf{y}^d - \mathbf{H}\widehat{\boldsymbol{\beta}}_\gamma\right), \tag{7.39}$$

and

$$\begin{aligned} c^{**}\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}, \gamma\right) = {} & c^*\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}, \gamma\right) + \left(\mathbf{h}(\boldsymbol{\theta})^T - \mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T \mathbf{A}_\gamma^{-1}\mathbf{H}\right)\left(\mathbf{H}^T\mathbf{A}_\gamma^{-1}\mathbf{H}\right)^{-1} \\ & \times \left(\mathbf{h}(\boldsymbol{\theta}')^T - \mathbf{t}\left(\boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T \mathbf{A}_\gamma^{-1}\mathbf{H}\right)^T. \end{aligned} \tag{7.40}$$

We confirm that as $\gamma \to 0$, $\mathbf{A}_\gamma \to \mathbf{A}$ (see Equation 7.30) and therefore

$$f_c\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}}_\gamma, \widehat{\sigma}^2_\gamma, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \gamma\right) \xrightarrow{d} \mathcal{N}\left(\mathbf{y}^d, \widehat{\sigma}^2 0\right).$$

Summary of stochastic univariate emulation

- model inputs/outputs

$$\mathbf{y}^d = (f(\boldsymbol{\theta}^d_1), \ldots, f(\boldsymbol{\theta}^d_n))^T,$$

- basis function evaluations

$$\mathbf{H}^T = \left(\mathbf{h}\left(\boldsymbol{\theta}^d_1\right), \ldots, \mathbf{h}\left(\boldsymbol{\theta}^d_n\right)\right),$$

- correlation matrix

$$\mathbf{A}_\gamma = \mathbf{A} + \gamma\mathbf{I} \text{ where } \mathbf{A}_{i,j} = c\left(\boldsymbol{\theta}^d_i, \boldsymbol{\theta}^d_j|\mathbf{Q}\right),$$

- estimated regression coefficients

$$\widehat{\boldsymbol{\beta}}_\gamma|\boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \gamma = \left(\mathbf{H}^T\mathbf{A}_\gamma^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{A}_\gamma^{-1}\mathbf{y}^d,$$

- estimated variance

$$\widehat{\sigma}^2_\gamma|\boldsymbol{\theta}^d, \mathbf{y}^d, \widehat{\boldsymbol{\beta}}_\gamma, \mathbf{Q}, \gamma = \frac{\left(\mathbf{y} - \mathbf{H}\widehat{\boldsymbol{\beta}}_\gamma\right)^T \mathbf{A}_\gamma^{-1}\left(\mathbf{y} - \mathbf{H}\widehat{\boldsymbol{\beta}}_\gamma\right)}{n - q - 2},$$

• estimation of γ

$$\arg\max L^*(\gamma) = -\frac{1}{2}\log|\mathbf{A}_\gamma| - \frac{1}{2}\log\left|\mathbf{H}^T\mathbf{A}_\gamma^{-1}\mathbf{H}\right| - \frac{n-q}{2}\log(SSE),$$
$$SSE = \left(\mathbf{y} - \mathbf{H}\widehat{\boldsymbol{\beta}}_\gamma\right)^T \mathbf{A}_\gamma^{-1}\left(\mathbf{y} - \mathbf{H}\widehat{\boldsymbol{\beta}}_\gamma\right)$$

• a posteriori expectation

$$\mathbf{t}(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q})^T = \left(c(\boldsymbol{\theta}, \boldsymbol{\theta}_1^d|\mathbf{Q}), \ldots, c(\boldsymbol{\theta}, \boldsymbol{\theta}_n^d|\mathbf{Q})\right),$$
$$m^{**}\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}}_\gamma, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \gamma\right) = \mathbf{h}(\boldsymbol{\theta})^T\widehat{\boldsymbol{\beta}}_\gamma + \mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T \mathbf{A}_\gamma^{-1}\left(\mathbf{y}^d - \mathbf{H}\widehat{\boldsymbol{\beta}}_\gamma\right),$$

• a posteriori covariance

$$c^{**}\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right) = c^*\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}, \gamma\right) + \left(\mathbf{h}(\boldsymbol{\theta})^T - \mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T \mathbf{A}_\gamma^{-1}\mathbf{H}\right)\left(\mathbf{H}^T\mathbf{A}_\gamma^{-1}\mathbf{H}\right)^{-1}$$
$$\times \left(\mathbf{h}(\boldsymbol{\theta}')^T - \mathbf{t}\left(\boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right)^T \mathbf{A}_\gamma^{-1}\mathbf{H}\right)^T.$$

7.3.1 A one-dimensional example with stochasticity

We repeat the example presented in Section 7.2.2 (page 275), but properly incorporate stochasticity as described above, rather than the standard Bayesian emulation methods. Figure 7.4 shows the same sequence of plots as the previous example, but using the full construction instead. There are several subtle differences between these plots and the example that does not use stochasticity. The prior ($\mathbf{h}(\theta)\widehat{\boldsymbol{\beta}}_\gamma$) is much the same as before, and the expectation still threads between points, but not exactly through the centre as before. Now we can see that given two inputs that are the same ($\theta_1 = \theta_2$) that don't produce the same output ($y_1 \neq y_2$), the variance is no longer zero, but is instead enveloping both of the points as we would expect.

To the best of our knowledge, the treatment of stochasticity within the Bayesian emulation framework presented here is novel and provides a useful improvement when emulating stochastic models.

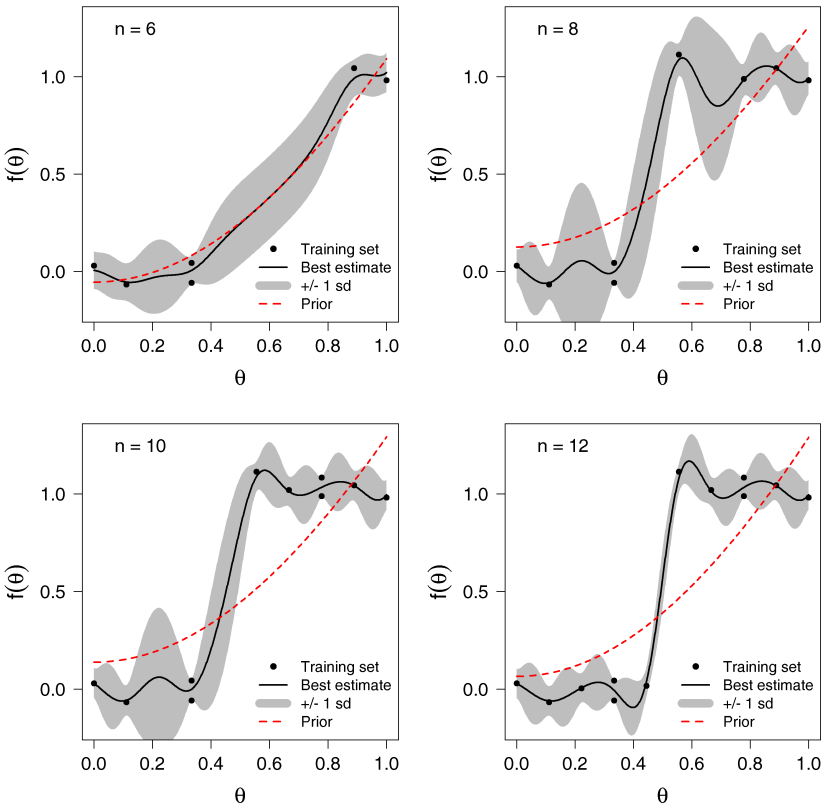


Figure 7.4: A sequence of emulators conditioned on $\boldsymbol{\theta}^d = \{\theta_i^d\}_{i=1}^n$ and $\mathbf{y}^d = \{y_i^d\}_{i=1}^n$ for $n = \{6, 8, 10, 12\}$ design points. In each plot $\boldsymbol{\theta}^d$ is the training set or model input plotted against the model output [•]. $m^{**}(\cdot)$ is the best estimate [solid line], ± 1 standard deviation [grey band], and $\mathbf{h}(\theta)\widehat{\boldsymbol{\beta}}_\gamma$ is the prior [dashed red line].

7.4 Multivariate emulators

A list of the variables used in describing multivariate emulators is provided in Tables 7.3 and 7.4. The univariate emulator described in Section 7.2 can be used to approximate the scalar function $f(\boldsymbol{\theta})$, i.e. $f : \boldsymbol{\Theta}_{p\times 1} \rightarrow \mathbb{R}^1$. We now consider the more general case where $f(\boldsymbol{\theta})$ is **vector** valued: $f : \boldsymbol{\Theta}_{p\times 1} \rightarrow \mathbb{R}^a$ (Hankin 2012). In the multivariate case there are a different types of observation for $h = 1, \ldots, a$. Each type of observation is a Gaussian process and the covariances between each of the observation types is incorporated in the emulator. For example, the different observations might be the numbers for each age if emulating an age structured model.

Table 7.3: Notation used in discussing and defining Bayesian emulation with multivariate emulators.

Symbol	Type	Dimensions	Description
a	scalar	1	Number of outputs $h \in a$
n	scalar	1	Number of design points $i \in n$
p	scalar	1	Dimension of input space $j \in p$
q	vector	1	Total number of regression functions for all output types $q = \sum_a q_a$ where q_a is the number of regression functions for output a, i.e. $k \in q_a$
$\boldsymbol{\Theta}$	matrix	p	The input space
$\boldsymbol{\theta}$	matrix	$p \times 1$	An element of the input space $\boldsymbol{\theta} \in \boldsymbol{\Theta}$
$\boldsymbol{\theta}^d$	matrix	$n \times p$	Stacked matrix of inputs $\boldsymbol{\theta}^d = \{\{\boldsymbol{\theta}_i^d\}_{i=1}^{an}\}_{j=1}^{p}$ for which we have outputs $\mathbf{y}^d = \{\{\mathbf{y}_{i,j}^d\}_{i=1}^{an}\}_{j=1}^{p}$
$\boldsymbol{\theta}'$	matrix	$p \times 1$	Matrix of inputs for which an output is to be approximated
$\mathbf{y}^d$	vector	$an \times 1$	Stacked vector of outputs at the design points $\mathbf{y}^d = \{\mathbf{y}_i^d\}_{i=1}^{an}$

In this section the notation for some variables changes slightly and we spend much more time describing the dimensions of these variables by

Table 7.4: Notation used in discussing and defining Bayesian emulation with multivariate emulators.

Symbol	Type	Dimensions	Description
a	scalar	1	Number of outputs $h \in a$
n	scalar	1	Number of design points for each output type $i \in n$
p	scalar	1	Dimension of input space $j \in p$
q	vector	1	Total number of regression functions for all output types $q = \sum_a q_a$ where q_a is the number of regression functions for output a, i.e. $k \in q_a$
$\mathbf{Q}_h$	array	$p \times p$	Diagonal matrix of roughness parameters for each output $h = 1, \ldots, a$
$\mathbf{V}$	matrix	$a \times a$	Matrix of covariances between each output
$\mathbf{\Sigma}$	matrix	$an \times an$	Correlation matrix
$\mathbf{H}^d$	matrix	$an \times q$	Matrix of basis function values evaluated at each input (from simulator)
$\boldsymbol{\beta}$	vector	$q \times 1$	Stacked vector of regression coefficients
$\widehat{\boldsymbol{\beta}}$	vector	$q \times 1$	The estimated value of $\boldsymbol{\beta}$ updated in light of the design outputs $\mathbf{y}^d$
$\mathbf{H}'$	matrix	$a \times q$	Matrix of basis function values evaluated at each input (from simulator)
$\mathbf{T}'$	matrix	$a \times an$	Matrix of covariances between $\boldsymbol{\theta}'$ and $\boldsymbol{\theta}^d$
$m^{**}(\boldsymbol{\theta})$	vector	$a \times 1$	A posteriori expectation
$c^{**}(\boldsymbol{\theta}, \boldsymbol{\theta}')$	matrix	$a \times a$	A posteriori covariance

introducing additional subscripts to make our description of multivariate emulation as clear as possible. However, they are constructed in much the same way as univariate emulators so we provide a much less detailed account of their construction.

As before, we generate a training set known as the design points ($\boldsymbol{\theta}_i^d$) and evaluate our computer model $f(\boldsymbol{\theta})$ relatively few times to generate a set of outputs ($\mathbf{y}_i^d$)

$$\begin{aligned} \mathbf{y}_i^d &= f(\boldsymbol{\theta}_i^d) \quad \text{where} \quad i = 1, \ldots, n, \\ \mathbf{y}_{an\times 1}^d &= \left((\mathbf{y}_1^d)^T, \ldots, (\mathbf{y}_n^d)^T\right)^T, \end{aligned} \tag{7.41}$$

where n is the number of design points. Here $\mathbf{y}_i^d$ is an $a \times 1$ vector of outputs for the $p \times 1$ vector of inputs $\boldsymbol{\theta}$.

We then approximate $f(\boldsymbol{\theta})$ by a Gaussian process

$$\begin{aligned} \mathbf{y}_{a\times 1} &= f_a\left(\boldsymbol{\theta}|\boldsymbol{\beta}, \sigma_e^2, \mathbf{Q}\right) \sim \mathcal{GP}\left(\mathbf{m}_{a\times 1}\left(\boldsymbol{\theta}|\boldsymbol{\beta}\right), \mathbf{V}_{a\times a}\right), \\ \mathbb{E}[\mathbf{y}]_{a\times 1} &= \mathbb{E}\left[f_a\left(\boldsymbol{\theta}|\boldsymbol{\beta}, \sigma_e^2, \mathbf{Q}\right)\right] = \mathbf{m}_{a\times 1}\left(\boldsymbol{\theta}|\boldsymbol{\beta}\right) = \mathbf{H}'_{a\times q}\boldsymbol{\beta}_{q\times 1}, \end{aligned} \tag{7.42}$$

conditional on the unknown stacked vector of coefficients $\boldsymbol{\beta} = \{\beta_k\}_{k=1}^q$. This Gaussian process implies that on the design points our data can be approximated by

$$\mathbf{y}_{an\times 1}^d \sim \mathcal{N}\left(\mathbf{H}_{an\times q}^d \boldsymbol{\beta}_{q\times 1}, \boldsymbol{\Sigma}_{an\times an}\right) \tag{7.43}$$

where $\mathbf{H}^d_{an\times q}$ is an $an \times q$ matrix of basis functions

$$\mathbf{H}^d_{an\times q} = \begin{bmatrix} \begin{bmatrix} \mathbf{h}_1(\boldsymbol{\theta}^d_1)^T & 0 & \dots & 0 \\ 0 & \mathbf{h}_2(\boldsymbol{\theta}^d_1)^T & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{h}_a(\boldsymbol{\theta}^d_1)^T \end{bmatrix}_{a\times q} \\ \begin{bmatrix} \mathbf{h}_1(\boldsymbol{\theta}^d_2)^T & 0 & \dots & 0 \\ 0 & \mathbf{h}_2(\boldsymbol{\theta}^d_2)^T & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{h}_a(\boldsymbol{\theta}^d_2)^T \end{bmatrix}_{a\times q} \\ \vdots \\ \begin{bmatrix} \mathbf{h}_1(\boldsymbol{\theta}^d_n)^T & 0 & \dots & 0 \\ 0 & \mathbf{h}_2(\boldsymbol{\theta}^d_n)^T & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \mathbf{h}_a(\boldsymbol{\theta}^d_n)^T \end{bmatrix}_{a\times q} \end{bmatrix}_{an\times q} = \begin{bmatrix} \mathbf{H}^1(\boldsymbol{\theta}_1)_{a\times q} \\ \mathbf{H}^1(\boldsymbol{\theta}_2)_{a\times q} \\ \vdots \\ \mathbf{H}^1(\boldsymbol{\theta}_n)_{a\times q} \end{bmatrix}_{an\times q}, \tag{7.44}$$

and $\boldsymbol{\beta}_{q\times 1}$ is a $q \times 1$ stacked vector of regression coefficients

$$\boldsymbol{\beta}_{q\times 1} = \begin{bmatrix} \begin{bmatrix} \boldsymbol{\beta}_{1,1} \\ \vdots \\ \boldsymbol{\beta}_{1,2} \end{bmatrix}_{q_1\times 1} \\ \begin{bmatrix} \boldsymbol{\beta}_{2,1} \\ \vdots \\ \boldsymbol{\beta}_{2,2} \end{bmatrix}_{q_2\times 1} \\ \vdots \\ \begin{bmatrix} \boldsymbol{\beta}_{a,1} \\ \vdots \\ \boldsymbol{\beta}_{a,q_a} \end{bmatrix}_{q_a\times 1} \end{bmatrix}_{q\times 1}. \tag{7.45}$$

Rather than specifying the covariance as $\sigma^2_e\mathbf{A}$ (as in Section 7.2 above and in Oakley & O'Hagan 2002) we change the notation for the covariance of multivariate emulators so that the univariate emulator variance (σ^2_e) is generalised to an $a \times a$ matrix of covariances between outputs ($\mathbf{V}_{a\times a}$) in which the diagonal elements correspond to the univariate variances

$\{\sigma^2_{e,h}\}_{h=1}^{a}$. The univariate matrix of roughness scales ($\mathbf{Q}$) is simply repeated for each output type to $\{\mathbf{Q}_h\}_{h=1}^{a}$. Thus, the correlation function is

$$\mathbf{c}(\boldsymbol{\theta}, \boldsymbol{\theta}'|\mathbf{Q}_h)_{a\times a} = \mathbf{c}_{i_1,i_2}^{h_1,h_2} = \frac{\exp\left(-\left(\boldsymbol{\theta}_{i_1} - \boldsymbol{\theta}_{i_2}\right)^T \left(\frac{1}{2}\mathbf{Q}_{h_1}^{-1} + \frac{1}{2}\mathbf{Q}_{h_2}^{-1}\right)^{-1} \left(\boldsymbol{\theta}_{i_1} - \boldsymbol{\theta}_{i_2}\right)\right)}{\left|\left(\frac{1}{2}\mathbf{Q}_{h_1} + \frac{1}{2}\mathbf{Q}_{h_2}\right)\left(\frac{1}{2}\mathbf{Q}_{h_1}^{-1} + \frac{1}{2}\mathbf{Q}_{h_2}^{-1}\right)\right|^{\frac{1}{4}}}, \tag{7.46}$$

for $h_1 = 1, \ldots, a$, $h_2 = 1, \ldots, a$, $i_1 = 1, \ldots, n$ and $i_2 = 1, \ldots, n$, such that if $h_1 = h_2$, then $\mathbf{Q}_{h_1} = \mathbf{Q}_{h_2}$, therefore $(\frac{1}{2}\mathbf{Q}_{h_1}^{-1} + \frac{1}{2}\mathbf{Q}_{h_2}^{-1})^{-1} = \mathbf{Q}_{h_1} = \mathbf{Q}_{h_2}$ and $|(\frac{1}{2}\mathbf{Q}_{h_1} + \frac{1}{2}\mathbf{Q}_{h_2})(\frac{1}{2}\mathbf{Q}_{h_1}^{-1} + \frac{1}{2}\mathbf{Q}_{h_2}^{-1})|^{\frac{1}{4}} = 1$, which gives

$$\mathbf{c}_{i_1,i_2}^{h_1,h_1} = \exp\left(-\left(\boldsymbol{\theta}_{i_1} - \boldsymbol{\theta}_{i_2}\right)^T \left(\mathbf{Q}_{h_1}^{-1}\right)\left(\boldsymbol{\theta}_{i_1} - \boldsymbol{\theta}_{i_2}\right)\right).$$

This is the same form used in univariate emulators (Equation 7.4, page 266). The multivariate form basically uses the average roughness scale when calculating the correlation between two different output parameters. It then follows that the covariance matrix is

$$\boldsymbol{\Sigma}_{an\times an} = \mathbf{V}_{h_1,h_2} c_{i_1,i_2}^{h_1,h_2}, \tag{7.47}$$

or more explicitly

$$\boldsymbol{\Sigma}_{an\times an} = \begin{bmatrix} \mathbf{V}_{a\times a} \star \begin{bmatrix} \mathbf{c}_{1,1}^{1,1} & \mathbf{c}_{1,1}^{1,2} & \dots & \mathbf{c}_{1,1}^{1,a} \\ \mathbf{c}_{1,1}^{2,1} & \mathbf{c}_{1,1}^{2,2} & \dots & \mathbf{c}_{1,1}^{2,a} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{1,1}^{a,1} & \mathbf{c}_{1,1}^{a,2} & \dots & \mathbf{c}_{1,1}^{a,a} \end{bmatrix}_{a\times a} & \mathbf{V}_{a\times a} \star \begin{bmatrix} \mathbf{c}_{1,2}^{1,1} & \mathbf{c}_{1,2}^{1,2} & \dots & \mathbf{c}_{1,2}^{1,a} \\ \mathbf{c}_{1,2}^{2,1} & \mathbf{c}_{1,2}^{1,2} & \dots & \mathbf{c}_{1,2}^{2,a} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{1,2}^{a,1} & \mathbf{c}_{1,2}^{a,2} & \dots & \mathbf{c}_{1,2}^{a,a} \end{bmatrix}_{a\times a} & \dots & \mathbf{V}_{a\times a} \star \begin{bmatrix} \mathbf{c}_{1,n}^{1,1} & \mathbf{c}_{1,n}^{1,2} & \dots & \mathbf{c}_{1,n}^{1,a} \\ \mathbf{c}_{1,n}^{1,1} & \mathbf{c}_{1,n}^{1,2} & \dots & \mathbf{c}_{1,n}^{2,a} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{1,n}^{a,1} & \mathbf{c}_{1,n}^{a,2} & \dots & \mathbf{c}_{1,n}^{a,a} \end{bmatrix}_{a\times a} \\ \mathbf{V}_{a\times a} \star \begin{bmatrix} \mathbf{c}_{2,1}^{1,1} & \mathbf{c}_{2,1}^{1,2} & \dots & \mathbf{c}_{2,1}^{1,a} \\ \mathbf{c}_{2,1}^{2,1} & \mathbf{c}_{2,1}^{2,2} & \dots & \mathbf{c}_{2,1}^{2,a} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{2,1}^{a,1} & \mathbf{c}_{2,1}^{a,2} & \dots & \mathbf{c}_{2,1}^{a,a} \end{bmatrix}_{a\times a} & \mathbf{V}_{a\times a} \star \begin{bmatrix} \mathbf{c}_{2,2}^{1,1} & \mathbf{c}_{2,2}^{1,2} & \dots & \mathbf{c}_{2,2}^{1,a} \\ \mathbf{c}_{2,2}^{1,2} & \mathbf{c}_{2,2}^{2,2} & \dots & \mathbf{c}_{2,2}^{2,a} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{2,2}^{1,a} & \mathbf{c}_{2,2}^{2,a} & \dots & \mathbf{c}_{2,2}^{a,a} \end{bmatrix}_{a\times a} & \dots & \mathbf{V}_{a\times a} \star \begin{bmatrix} \mathbf{c}_{2,n}^{1,1} & \mathbf{c}_{2,n}^{1,2} & \dots & \mathbf{c}_{2,n}^{1,a} \\ \mathbf{c}_{2,n}^{1,2} & \mathbf{c}_{2,n}^{2,2} & \dots & \mathbf{c}_{2,n}^{2,a} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{2,n}^{1,a} & \mathbf{c}_{2,n}^{2,a} & \dots & \mathbf{c}_{2,n}^{a,a} \end{bmatrix}_{a\times a} \\ & \vdots & & \vdots & \ddots & \vdots & \\ \mathbf{V}_{a\times a} \star \begin{bmatrix} \mathbf{c}_{n,1}^{1,1} & \mathbf{c}_{n,1}^{1,2} & \dots & \mathbf{c}_{n,1}^{1,a} \\ \mathbf{c}_{n,1}^{1,2} & \mathbf{c}_{n,1}^{2,2} & \dots & \mathbf{c}_{n,1}^{2,a} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{n,1}^{1,a} & \mathbf{c}_{n,1}^{2,a} & \dots & \mathbf{c}_{n,1}^{a,a} \end{bmatrix}_{a\times a} & \mathbf{V}_{a\times a} \star \begin{bmatrix} \mathbf{c}_{n,2}^{1,1} & \mathbf{c}_{n,2}^{1,2} & \dots & \mathbf{c}_{n,2}^{1,a} \\ \mathbf{c}_{n,2}^{1,2} & \mathbf{c}_{n,2}^{2,2} & \dots & \mathbf{c}_{n,2}^{2,a} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{n,2}^{1,a} & \mathbf{c}_{n,2}^{2,a} & \dots & \mathbf{c}_{n,2}^{a,a} \end{bmatrix}_{a\times a} & \dots & \mathbf{V}_{a\times a} \star \begin{bmatrix} \mathbf{c}_{n,n}^{1,1} & \mathbf{c}_{n,n}^{1,2} & \dots & \mathbf{c}_{n,n}^{1,a} \\ \mathbf{c}_{n,n}^{1,2} & \mathbf{c}_{n,n}^{2,2} & \dots & \mathbf{c}_{n,n}^{2,a} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{n,n}^{1,a} & \mathbf{c}_{n,n}^{2,a} & \dots & \mathbf{c}_{n,n}^{a,a} \end{bmatrix}_{a\times a} \end{bmatrix}_{an\times an}, \tag{7.48}$$

where $\star$ indicates element-wise multiplication.

We then **calibrate** our approximation by finding the REML estimate of $\boldsymbol{\beta}$

$$\widehat{\boldsymbol{\beta}}_{\gamma}|\boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \mathbf{D}^1 = \left((\mathbf{H}^d_{an\times q})^T(\boldsymbol{\Sigma}^{-1}_{\gamma})_{an\times an}\mathbf{H}^d_{an\times q}\right)^{-1}(\mathbf{H}^d_{an\times q})^T(\boldsymbol{\Sigma}^{-1}_{\gamma})_{an\times an}\mathbf{y}^d_{an\times 1}, \tag{7.49}$$

where

$$\begin{aligned}\boldsymbol{\Sigma}_{\gamma} &= \boldsymbol{\Sigma}_{an\times an} + \mathbf{D}_{an\times an},\\ \mathbf{D}_{an\times an} &= \text{diag}\left(\mathbf{D}^1_{a\times a}, \ldots, \mathbf{D}^1_{a\times a}\right),\\ \mathbf{D}^1_{a\times a} &= \text{diag}\left(\sigma^2_{\gamma,1}, \ldots, \sigma^2_{\gamma,a}\right),\end{aligned}$$

where $(\sigma^2_{\gamma,1}, \ldots, \sigma^2_{\gamma,a})$ must be estimated. Similar to Hankin (2012), we choose to to this independently for each output type a and so the methods used in Section 7.3 (page 278) above can be used. However, this is an area that requires further research. Future choices here include: further generalisation so that $\mathbf{D}^1$ is a full $a \times a$ matrix (rather than a diagonal matrix), or; further simplification (if all σ^2_{γ}'s are assumed constant) to $\mathbf{D}^1 = \sigma^2_{\gamma}\mathbf{I}$.

Given $\widehat{\boldsymbol{\beta}}_{\gamma}$, $\boldsymbol{\Sigma}_{\gamma}$ and $\mathbf{D}^1$, we proceed with the construction of a multivariate emulator that incorporates REML estimation. We construct the approximation

$$\mathbf{y} = f_u\left(\boldsymbol{\theta}|\boldsymbol{\beta}, \boldsymbol{\Sigma}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right) \sim \mathcal{GP}\left(\mathbf{m}^*_{a\times 1}\left(\boldsymbol{\theta}|\boldsymbol{\beta}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right), \mathbf{c}^*_{a\times a}\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right)\right), \tag{7.50}$$

where

$$\mathbf{m}^*_{a\times 1}\left(\boldsymbol{\theta}|\boldsymbol{\beta}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right) = (\mathbf{H}^1)^T_{a\times q}\boldsymbol{\beta}_{q\times 1} + \mathbf{T}^T_{a\times an}\boldsymbol{\Sigma}^{-1}_{an\times an}\left(\mathbf{y}^d_{an\times 1} - \mathbf{H}^d_{an\times q}\boldsymbol{\beta}_{q\times 1}\right), \tag{7.51}$$

$$\mathbf{c}^*_{a\times a}\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right) = \mathbf{c}\left(\boldsymbol{\theta}_{a\times p}, \boldsymbol{\theta}'_{a\times p}|\mathbf{Q}_a\right)_{a\times a} - (\mathbf{T}_{a\times an})^T\boldsymbol{\Sigma}^{-1}_{an\times an}\mathbf{T}'_{a\times an}, \tag{7.52}$$

$$\mathbf{T}_{a\times an} = \left(\mathbf{c}\left(\boldsymbol{\theta}_{a\times p}|\boldsymbol{\theta}^d_{an\times p}, \mathbf{Q}_a\right)_{a\times a}, \ldots, \mathbf{c}\left(\boldsymbol{\theta}_{a\times p}|\boldsymbol{\theta}^d_{an\times p}, \mathbf{Q}_a\right)_{a\times a}\right)_{a\times an}. \tag{7.53}$$

Finally, we construct the **emulator**

$$\mathbf{y}_{a\times 1} = f_c\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}}, \boldsymbol{\Sigma}_{\gamma}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right) \sim \mathcal{GP}\left(m^{**}_{a\times 1}\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right), (\boldsymbol{\Sigma}_{\gamma})_{an\times an}\right), \tag{7.54}$$

where

$$\mathbf{m}^{**}_{a\times 1}\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}},\boldsymbol{\theta}^d,\mathbf{y}^d,\mathbf{Q}\right) = \mathbf{H}^{1'}_{a\times q}(\widehat{\boldsymbol{\beta}}_\gamma)_{q\times 1} + \mathbf{T}_{a\times an}(\boldsymbol{\Sigma}^{-1}_\gamma)_{an\times an}\left(\mathbf{y}^d_{an\times 1} - \mathbf{H}^d_{an\times q}(\widehat{\boldsymbol{\beta}}_\gamma)_{q\times 1}\right)_{an\times 1}, \tag{7.55}$$

$$\begin{aligned}\mathbf{c}^{**}_{a\times a}\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q}\right) = \mathbf{c}^{*}_{a\times a}&\left(\boldsymbol{\theta}_{a\times p},\boldsymbol{\theta}'_{a\times p}|\boldsymbol{\theta}^d_{an\times p},\mathbf{Q}_a\right)\\ &+\left(\mathbf{H}^1_{a\times q} - \mathbf{T}_{a\times an}(\boldsymbol{\Sigma}^{-1}_\gamma)_{an\times an}\mathbf{H}^d_{an\times q}\right)\\ &\times\left((\mathbf{H}^d)^T_{an\times q}(\boldsymbol{\Sigma}^{-1}_\gamma)_{an\times an}\mathbf{H}^d_{an\times q}\right)^{-1}\\ &\times\left(\mathbf{H}^{1'}_{a\times q} - \mathbf{T}'_{a\times an}(\boldsymbol{\Sigma}^{-1}_\gamma)_{an\times an}\mathbf{H}^d_{an\times q}\right)^T,\end{aligned} \tag{7.56}$$

and $\mathbf{H}^{1'}_{a\times q}$ is an $a \times q$ matrix of basis functions

$$\mathbf{H}^{1'}_{a\times q} = \begin{bmatrix} \mathbf{h}_1(\boldsymbol{\theta}')^T & 0 & \cdots & 0 \\ 0 & \mathbf{h}_2(\boldsymbol{\theta}')^T & & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{h}_a(\boldsymbol{\theta}')^T \end{bmatrix}_{a\times q},$$

where $\mathbf{h}_j(\boldsymbol{\theta}^{(j)})^T$ are the basis functions for the output types $j = 1,\ldots,p$.

7.5 Inference on an emulator

The purpose of a Bayesian emulator is to make inference of computationally expensive models tractable. That is, a good emulator should provide an unbiased and relatively precise approximation of a computationally expensive model, yet be much faster to evaluate, allowing the use of standard inference procedures such as MCMC (e.g. Henderson et al. 2009). Once an emulator of a computationally expensive model has been developed and conditioned, the emulator simply replaces the original model when making inference about the system.

We now describe the relationship between the physical system, the simulator (i.e. the computationally expensive model) and the emulator. We denote observations of the system as the vector y^o. We denote the true

physical system values as the vector y^*. We describe the relationship between the observations y^o and and the true value y^* as

$$y^o = y^* + \varepsilon^o, \tag{7.57}$$

where ε^o is the observation error. Assuming that the simulator is a sequential evolving system (e.g. a state-space biomass dynamics model) then the relationship between the simulator and the system is then expressed as

$$y^* = f(\boldsymbol{\theta}) + \varepsilon^p, \tag{7.58}$$

where ε^p is the process error of the model (if we are not considering a sequential evolving system then we simply ignore this step). A belief network for standard inference is

$$\begin{array}{ccccccc} & & \boldsymbol{\theta} & & \varepsilon^p & & \varepsilon^o \\ & & \downarrow & & \downarrow & & \downarrow \\ f(\cdot) & \longrightarrow & f(\boldsymbol{\theta}) & \longrightarrow & y & \longrightarrow & y^o \end{array}$$

We now consider how this belief network looks if we introduce a Bayesian emulator with a conditioned approximating function $f_c(\cdot)$ (which has mean and variance $m^{**}(\cdot)$ and $\widehat{\sigma}_e^2 c^{**}(\cdot,\cdot)$ respectively). Because $f_c(\cdot)$ is assumed to be a close approximation of $f(\cdot)$, a belief network for inference on an emulator is

$$\begin{array}{ccccccccc} & & \boldsymbol{\theta} & & \boldsymbol{\theta}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q} & & \varepsilon^p & & \varepsilon^o \\ & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\ f(\cdot) & \longrightarrow & f(\boldsymbol{\theta}) & \text{- - - - - - -} & f_c\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right) & \longrightarrow & y & \longrightarrow & y^o \end{array}$$

Given that we observe data y^o, using Bayes' theorem we can write

$$p(\boldsymbol{\theta}|y^o) = \frac{p(y^o|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(y^o)} \propto p(y^o|\boldsymbol{\theta})p(\boldsymbol{\theta}), \tag{7.59}$$

where $p(\boldsymbol{\theta})$ is the prior distribution of the parameter(s) before any data is observed, $p(y^o|\boldsymbol{\theta})$ is the sampling distribution of the observed data conditional on its parameters (also termed the likelihood $L(\boldsymbol{\theta}; y^o) = p(y^o|\boldsymbol{\theta})$), $p(y^o)$ is the marginal likelihood and $p(\boldsymbol{\theta}|y^o)$ is the posterior distribution of the parameters conditional on the observed data y^o. Under the emulator we write

$$p(\boldsymbol{\theta}|y^o) = p\left(\boldsymbol{\theta}|y^o, \left(\mathbf{y}^d, \boldsymbol{\theta}^d, \mathbf{Q}\right)\right) \propto p\left(y^o|\boldsymbol{\theta}\left(\mathbf{y}^d, \boldsymbol{\theta}^d, \mathbf{Q}\right)\right) p(\boldsymbol{\theta}), \tag{7.60}$$

where the emulator itself defines the likelihood

$$p\left(y^o|\boldsymbol{\theta}\right) = p\left(y^o|\boldsymbol{\theta}\left(\mathbf{y}^d, \boldsymbol{\theta}^d, \mathbf{Q}\right)\right) \sim \mathcal{N}\left(m^{**}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right), \widehat{\sigma}_e^2 c^{**}\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right)\right). \tag{7.61}$$

Now we can introduce observational errors σ_o^2 (iid)

$$y|\boldsymbol{\theta}, \sigma_o^2\left(\widehat{\boldsymbol{\beta}}, \widehat{\sigma}_e^2, \mathbf{y}^d, \boldsymbol{\theta}^d, \mathbf{Q}\right) \sim \mathcal{N}\left(m^{**}\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right), \sigma_o^2 + \widehat{\sigma}_e^2\left(\boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}\right) c^{**}\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right)\right), \tag{7.62}$$

and finally we can proceed using standard methods

$$\begin{aligned} p(\boldsymbol{\theta}, \sigma_o^2|y^o) &\propto p(y^o|\boldsymbol{\theta}, \sigma_o^2)p(\boldsymbol{\theta})p(\sigma_o^2) \\ &\propto \int p(y^o, y|\boldsymbol{\theta}, \sigma_o^2)p(\boldsymbol{\theta})p(\sigma_o^2)dy \\ &\propto \int p(y^o|y, \sigma_o^2)p(y|\boldsymbol{\theta}, \sigma_o^2)p(\boldsymbol{\theta})p(\sigma_o^2)dy \\ \text{approx.} \quad &\propto \int p(y^o|y, \sigma_o^2)p^{**}(y|\boldsymbol{\theta}, \mathbf{y}^d, \boldsymbol{\theta}^d, \mathbf{Q})p(\boldsymbol{\theta})p(\sigma_o^2)dy, \end{aligned} \tag{7.63}$$

identifying the emulator as $p^{**}(y|\boldsymbol{\theta}, \mathbf{y}^d, \boldsymbol{\theta}^d, \mathbf{Q})$.

7.6 Univariate emulation of a biomass dynamics model

This example develops a univariate emulator of a biomass dynamics model and uses this emulator, nested inside a state-space framework, to make inference about a simulated data set.

7.6.1 The simulator

We begin with a biomass dynamics simulation model. As in Chapter 5 (page 137), we use the state-space version of this model

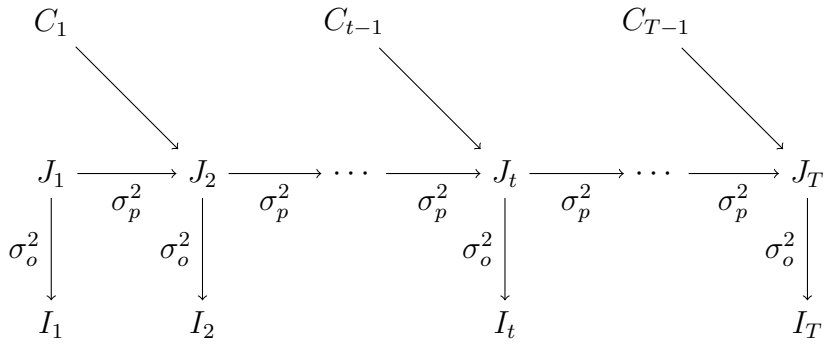
$$\begin{aligned} J_1 &= K, \\ J_t &= \left(J_{t-1} + rJ_{t-1}\left(1 - J_{t-1}\right) - \frac{C_{t-1}}{K}\right)e^{\varepsilon_t^p} \quad 2 \leq t \leq T, \end{aligned} \tag{7.64}$$

$$I_t = qKJ_t e^{\varepsilon_t^o} \quad \forall t, \tag{7.65}$$

where

$$\varepsilon_t^p \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_p^2) \qquad \text{and} \qquad \varepsilon_t^o \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_o^2).$$

Here $J_t = B_t/K$ where B_t is the biomass (tonnes) at the beginning of time t, r is the intrinsic rate of population increase, K is the carrying capacity (tonnes), and C_t is the catch (tonnes) at time t. ε_t^o and ε_t^p are iid observation and process deviations at time t with variances σ_o^2 and σ_p^2. The hidden Markov of this model can be represented as:



Similar to the packhorse rock lobster simulations in Chapter 5, we simulate CPUE (I_t) and biomass (B_t) from this model using the parameter set outlined in Table 7.5 and the catch history used in the packhorse rock lobster simulations (Section 5.3.2, page 141).

Table 7.5: Parameter values used in the biomass dynamics model simulation.

Parameter	Value	Units	Description
r	0.17	tonnes^{-1}	Intrinsic rate of population increase
K	1500	tonnes	Carrying capacity
q	0.002	tonnes^{-1}	Catchability coefficient
σ_o	0.5	-	Observation error standard deviation
σ_p	0.001	tonnes	Process error standard deviation

7.6.2 The emulator

Our goal is to construct a univariate emulator of the process component of the simulation model (Equation 7.64) and do inference on the emulator coupled with the observation component (Equation 7.65). To do this, we replace the process equation that defines each latent state in our model with a conditioned univariate emulator, i.e. $J_t = f(\boldsymbol{\theta}_t)$ is replaced by $J_t = f_c(\boldsymbol{\theta}_t)$ where $\mathbb{E}[J_t] = m^{**}(\boldsymbol{\theta}_t)$ and the emulator inputs are $\boldsymbol{\theta}_t = (J_{t-1}, r, K, C_{t-1})$.

Emulator inputs ($\boldsymbol{\theta}_t$)

We use a Latin hypercube design to produce $n = 100$ input vectors ($\boldsymbol{\theta}^d = \{\boldsymbol{\theta}_i^d\}_{i=1}^n$) within realistic ranges for each of the emulated model parameters (Figure 7.5). We then use Equation 7.64 to obtain the design points $\mathbf{y}^d = \{y_i^d\}_{i=1}^n$.

Emulator outputs ($\mathbf{y}_t$)

The basis function used was $\mathbf{h}(\boldsymbol{\theta})^T = (1, J_{t-1}, r, K, C_{t-1})$. Roughness lengths ($\mathbf{Q}$) were estimated and fixed a priori. The emulator was conditioned on the design inputs ($\boldsymbol{\theta}^d$) and outputs ($\mathbf{y}^d$) of the simulator.

We test the performance of the emulator by drawing another 100 input parameters ($\boldsymbol{\theta}$) from the parameter space ($\boldsymbol{\Theta}$), running the model another 100 times using each of these parameter sets to obtain outputs ($\mathbf{y}$), but not using these inputs/outputs to condition the emulator. Instead we provide these inputs to the already conditioned emulator and obtain the emulated outputs also to test the performance of the emulator. The biomass predicted by the emulator was very close to the biomass derived using Equation 7.64, given the same input parameters and covariates (Figure 7.6).

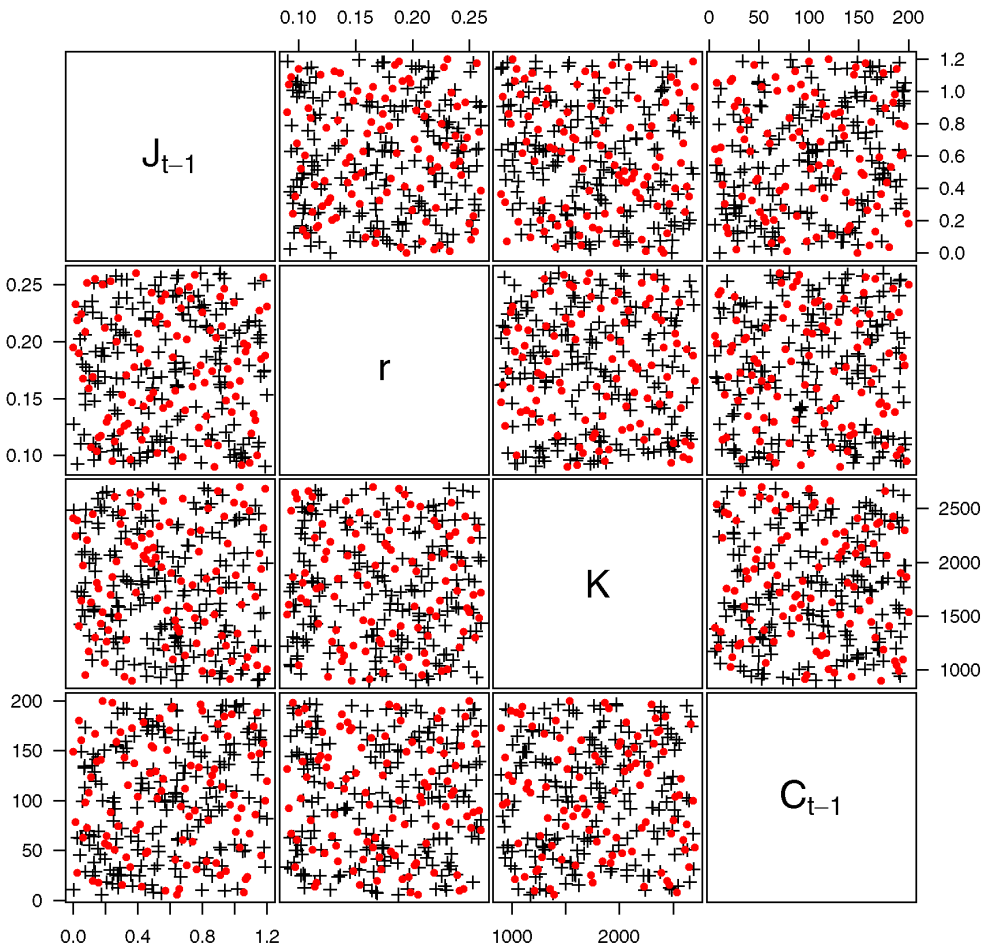


Figure 7.5: Points (•) were used to condition the emulator, crosses (+) used to test the performance of the emulator.

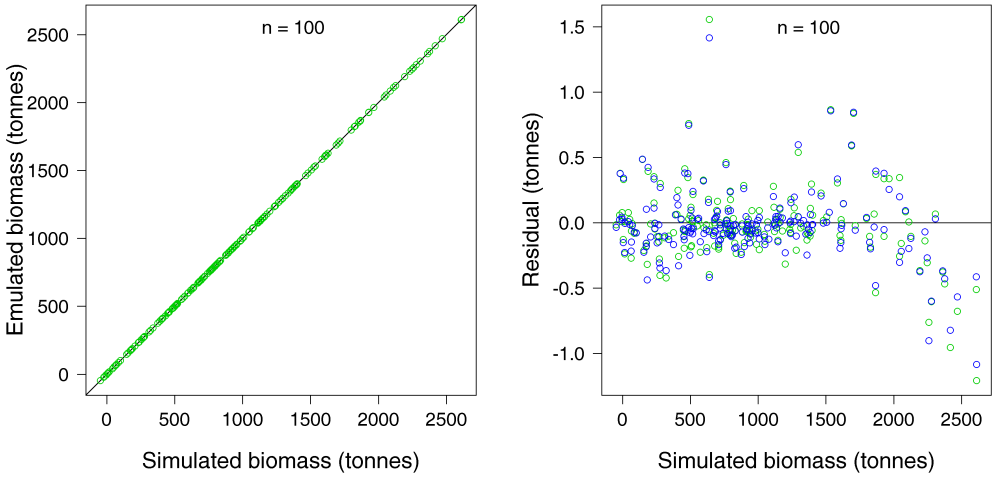


Figure 7.6: Simulated biomass versus the emulated biomass [left] and residual [right].

7.6.3 Inference

We are interested the probabilistic relationship between the following:

- **The data y:** the catch per unit effort (I_t). Let $\mathbf{y} = \{I_t\}_{t=1}^{T}$
- **The covariates z:** the catch (C_t). Let $\mathbf{z} = \{C_t\}_{t=1}^{T}$
- **The unknown process parameters $\boldsymbol{\psi}$:** the intrinsic rate of population increase (r), the carrying capacity of the population (K) and the process error variance (σ_p^2). Let $\boldsymbol{\psi} = \{r, K, \sigma_p^2\}$
- **The unknown observation parameters $\boldsymbol{\phi}$:** the catchability coefficient (q) and the observation error variance (σ_o^2). Let $\boldsymbol{\phi} = \{q, \sigma_o^2\}$
- **The unknown latent states x:** the depletion (J_t). Let $\mathbf{x} = \{J_t\}_{t=1}^{T}$

Using Bayes' theorem, the posterior distribution of the model parameters ($\boldsymbol{\psi}$ and $\boldsymbol{\phi}$) and the states ($\mathbf{x}$), given the data ($\mathbf{y}$) and covariates ($\mathbf{z}$) is

$$\pi(\boldsymbol{\psi}, \boldsymbol{\phi}, \mathbf{x}|\mathbf{y}, \mathbf{z}) \propto \pi(\boldsymbol{\psi}, \boldsymbol{\phi}, \mathbf{x}|\mathbf{z})\pi(\mathbf{y}|\boldsymbol{\psi}, \boldsymbol{\phi}, \mathbf{x}), \tag{7.66}$$

where

$$\begin{aligned}\pi\left(\boldsymbol{\psi},\boldsymbol{\phi},\mathbf{x}|\mathbf{z}\right) &= \pi\left(r,K,q,\sigma_o^2,\sigma_p^2,\mathbf{x}|\mathbf{z}\right)\\ &= \pi(r)\pi(K)\pi(q)\pi(\sigma_o^2)\pi(\sigma_p^2)\pi\left(\mathbf{x}|\mathbf{z},r,K,\sigma_p^2\right)\\ &= \pi(r)\pi(K)\pi(q)\pi(\sigma_o^2)\pi(\sigma_p^2)\pi\left(J_1|K,\sigma_p^2\right)\prod_{t=2}^{T}\pi\left(J_t|J_{t-1},C_{t-1},r,K,\sigma_p^2\right),\end{aligned}$$

$$\pi\left(\mathbf{y}|\mathbf{x},\boldsymbol{\psi},\boldsymbol{\phi}\right) = \pi\left(\mathbf{y}|\mathbf{x},K,q,\sigma_o^2\right) = \prod_{t=1}^{T}\pi\left(I_t|J_t,K,q,\sigma_o^2\right). \tag{7.67}$$

Here the component $\prod_{t=2}^{T}\pi\left(J_t|J_{t-1},C_{t-1},r,K,\sigma_p^2\right)$ is evaluated by the emulator and we state

$$J_t|\boldsymbol{\theta},\widehat{\boldsymbol{\beta}},\widehat{\sigma}_e^2,\boldsymbol{\theta}^d,\mathbf{y}^d,\mathbf{Q} \sim \mathcal{N}\left(m^{**}\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}},\boldsymbol{\theta}^d,\mathbf{y}^d,\mathbf{Q}\right),\widehat{\sigma}_e^2 c^{**}\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q}\right)\right), \tag{7.68}$$

where $\boldsymbol{\theta}$ is a vector of emulator inputs (i.e. a collection of parameters, latent states and covariates). Here the emulator is constructed using the inputs $\boldsymbol{\theta} = \{J_{t-1}, C_{t-1}, r, K\}$. Notice that the observation parameters ($\boldsymbol{\phi}$) are not required by the emulator. We also exclude the process error variance (σ_p^2) from the emulator and model process error within the state-space framework. It would be more common for the stochastic computer model to provide outputs after process error is applied (i.e. if we are thinking about complex computationally expensive computer models). Although we do not incorporate the process error in the emulator in this example, it is implicitly included in the next example. Here we embed the emulator within a state-space biomass dynamics model replacing the core (i.e. the process equation) with the conditioned emulator

$$J_t|\boldsymbol{\theta}_{t-1},\sigma_p^2 = f_c\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}},\widehat{\sigma}_e^2,\boldsymbol{\theta}^d,\mathbf{y}^d,\mathbf{Q}\right)e^{\varepsilon_t^p}, \tag{7.69}$$

$$I_t|J_t,\boldsymbol{\phi} = qKJ_te^{\varepsilon_t^o} = qB_te^{\varepsilon_t^o}, \tag{7.70}$$

noticing that the observation equation is only conditional on J_t and $\boldsymbol{\phi}$. Now standard inference procedures may be applied to the emulator to obtain best input parameters ($\boldsymbol{\theta}$) and observation parameters ($\boldsymbol{\phi}$) for our simulator conditional on the observations of the system. We specify reasonably informative log-normal priors for the parameters r and K (see Figure 7.7 below), and high variance inverse gamma distributions for the

parameters q, σ_o^2 and σ_p^2 (i.e. $\mathcal{IG}(0.001, 0.001)$). We run an MCMC of 1 million iterations, saving every 1000th sample, to obtain 1000 samples from the posterior distribution.

We only report the MCMC trace plots and posterior densities (Figure 7.7), but note that the model fit to the CPUE data and the biomass trajectory arising from this fit provide an excellent match to the simulated CPUE/biomass. MCMC did an excellent job of recovering most of the parameter values specified in the simulation, as we would expect given that the performance of the Bayesian emulator (Figure 7.6). As we also found in Chapter 5 (page 129), the observation and process error variance parameters were slightly under and overestimated, respectively. This is to be expected given the reasonably high observation error specified in the simulation (see Table 7.5).

7.7 Multivariate emulation of a stochastic agent-based model

Finally we describe a stochastic multivariate Bayesian emulator for the agent-based snapper model described in Chapter 4 (page 89). The ABM is a spatially explicit multi-generational agent-structured fish simulation model. The parameter values used in the simulation and structural aspects of the ABM were cherry picked from Francis & McKenzie (2013), and older versions of this stock assessment. Because the ABM used here simplified some aspects of the SNA 1 stock assessment and includes untested hypothesis about the dynamics of these stocks, none of the work in this chapter should be used to make inference about the SNA 1 fish stocks. This work is intended as a proof of concept only.

7.7.1 The simulator

A spatially explicit ABM of snapper in northern New Zealand (SNA 1) was developed using the agent-based simulation model discussed in Chapter 4

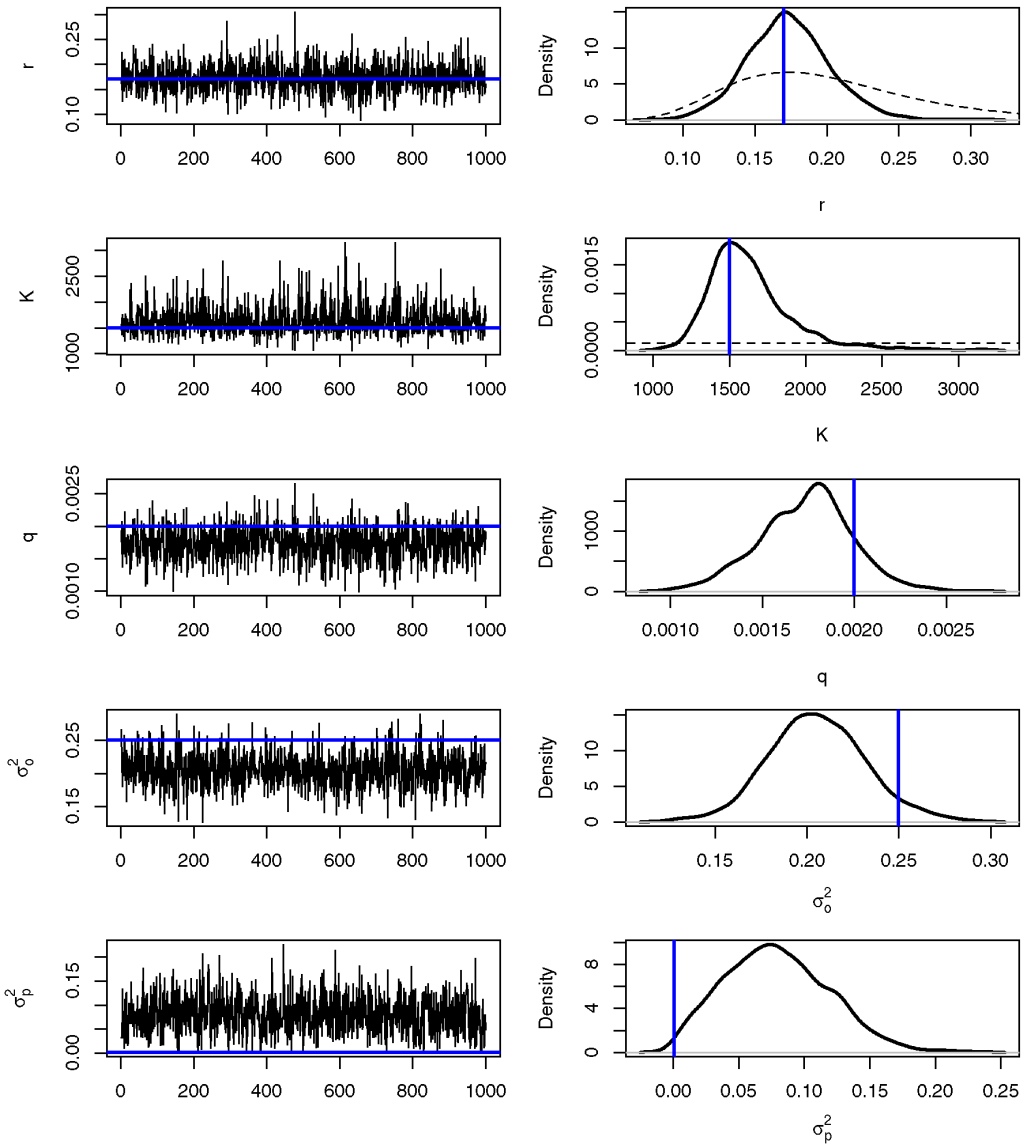


Figure 7.7: MCMC trace plots and posterior densities sampled from the state-space biomass dynamics Bayesian emulator. Posterior traces and densities are indicated as black lines, priors as dashed black lines (the priors for the catchability coefficient q, the observation error, and the process error are all uninformative inverse gamma priors so are omitted from the plot), and values specified in simulation as solid blue lines.

(page 89). Background information on this species can be found in Chapter 3 (page 77).

The ABM was developed to capture the hypothetical life cycle of snapper in SNA 1. The model was constructed as a three-stock (j), three-area (z), two sex (s) model (with the same demographic parameters for each sex, i.e. really a single sex model) with fifty age groups ($a = 1, \ldots, 50+$). The spawning stock biomass (SSB_t) is defined as the mature biomass of both females and males. The three areas in the model are labelled: East Northland (EN), the Hauraki Gulf (HG) and the Bay of Plenty (BP). Immature fish are distributed in each of the three areas. Each year (t) the fish move about each of the three model areas with probability based on a fixed migration matrix

$$\mathbf{\Omega} = \begin{pmatrix} 0.77 & 0.05 & 0.18 \\ 0.09 & 0.51 & 0.40 \\ 0.24 & 0.28 & 0.48 \end{pmatrix}.$$

Each agent's (i) home (h) is defined as the area that they recruited to in the model. Mature fish return home to spawn. The fish in each area at the time spawning stock biomass is calculated make up the stock in that area. Thus there were two migration events per year in the model.

The model was initialised with an equilibrium age and spatial distribution structure. In the model, the initial equilibrium state was found iteratively in two phases. The first phase involved applying recruitment, natural mortality and ageing processes. Movement was not permitted during this phase. In the second phase, movement processes were introduced. The duration of each phase of initialisation was 100 years. The length of these two phases was determined by running the model a few times and checking that population biomass in each area reached a plateau and remained there for at least a couple of decades and that the age-structure of the population looked realistic (i.e. the numbers at age roughly showed an exponential decline).

Following initialisation, the model was run over a period of 114 years, from 1900 to 2013. During this period, fishing mortality and tagging processes consistent with actual catches and tagging in the history of the SNA

1 fishery were applied. The parameter values specified in the simulation model are provided below in Table 7.6. Initial exploration showed that

Table 7.6: The key parameter values used in specifying the agent based snapper (SNA 1) simulation model. For a more detailed list of the parameter used in this simulation see Appendix D (page 351).

Attribute	Parameter	Value
Length	L_∞	58.8
	k	0.102
	t_0	-1.11
	σ_{L_∞}	0.01
	σ_k	0.001
	$cv_{\Delta\ell}$	0.001
Length-weight	α	4.467e-08
	β	2.793
	σ_α	1e-10
	σ_β	0.001
Maturity	A_{50}	4
	A_{to95}	4.7
	L	3
	R	5
	$\sigma_{A_{50}}$	0.001
	$\sigma_{A_{to95}}$	0.001
Natural mortality	M	0.075
	σ_M	0.001
Selectivity	γ_1	6.05
	γ_L	2.31
	γ_R	100
Recruitment	R_0	443493, 950050, 318619
	h	0.85
	σ_R	0.14
	ρ	0.6

the actual R_0 estimated in the 2013 SNA 1 stock assessment could not be used because the model consumed all of the computer's primary memory

(RAM) (i.e. it resulted in too many agents being created). To avoid this, we tried specifying a model with more fish per agent. But, because agents are allowed to split (see Chapter 4, page 102) and we wanted to avoid merging too many agents as this tends to results in the dilution of agent attributes (see the see Chapter 4, page 106 for more detail), the number of agents created was still too great and resulted in the model exceeding the computers memory. Instead, the smaller R_0 parameters identified in Table 7.6 were used and the catches scaled down accordingly. The actual input file for the simulation and plots illustrating a selection of the model dynamics are included in Appendix D (page 351).

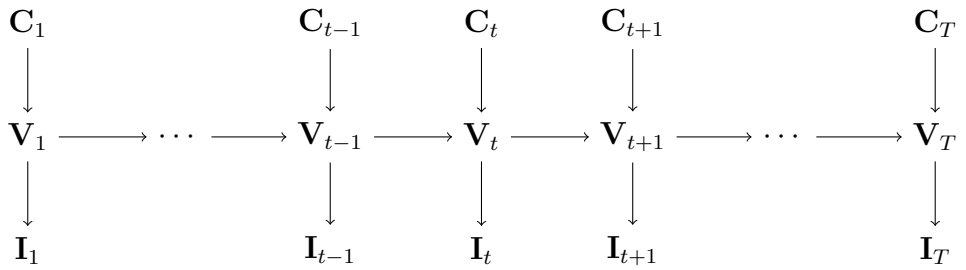
The ABM has the ability to introduce stochasticity in almost all modelled processes (e.g. growth, length-weight relationship, natural mortality). In this example we did not turn stochasticity off, but instead specified relatively small standard deviation parameters for all stochastic processes (see Table 7.6 and Appendix D, page 351). We proceeded in this way so that we can better test the performance of Bayesian emulation in capturing some of the complexity inherent in this model, rather than testing how well the method copes in the face of uncertainty. However, this would be an interesting avenue for further research.

7.7.2 The emulator

Ideally we would construct and condition a Bayesian emulator that takes all of the numbers at age in a single time-step, key model parameters, and the the catch as inputs (i.e. $\boldsymbol{\theta}_t = \{N_{a,t,z}, C_{t,z}, R_{0,z}, \ldots\}$) and provides the numbers at age in the following time-step as the output (i.e. $\mathbf{y}_{t+1} = \{N_{a,t+1}\}$). A series of emulators could then be linked together to form the process model of the numbers at age in the population.

However, initial exploration and attempts to emulate the numbers at age in this way proved difficult because emulators performed poorly and were too slow to be of any use (in MCMC). Instead we decided to wrap most of the complexity of the ABM into a stochastic multivariate emulator that models the evolution of the vulnerable biomass ($V_{t,z}$) of fish by area (z)

and time (t). This system can be represented graphically as:



where $\mathbf{V}_t$, $\mathbf{C}_t$ and $\mathbf{I}_t$ are vectors of the vulnerable biomass, the catch and the CPUE by area, respectively.

We provide further discussion on age-structured Bayesian emulators in the discussion at the end of this chapter (section 7.8, page 310).

Emulator inputs ($\boldsymbol{\theta}_t$)

We choose a subset of the models key parameters (Table 7.6) to estimate and fix the remaining parameters. The parameters that we chose to emulate include each stocks average recruitment ($R_{0,z}$), the peak (γ_1) and left hand limb (γ_L) of the double normal selectivity curve, the catchability coefficient (q), the recruitment standard deviation (σ_R) and the CPUE observation error standard deviation (σ_o). These parameters, along with the catch at any given time-step ($C_{t,z}$) and the vulnerable biomass in any given time-step ($V_{t,z}$), will be the inputs for our Bayesian emulator (i.e. $\boldsymbol{\theta}_t = \{V_{t,z}, C_{t,z}, R_{0,z}, \gamma_1, \gamma_L, q, \sigma_R, \sigma_o\}$).

The input design, on which the emulator is conditioned, is constructed with $n = 1000$ sets of parameters drawn using the Latin hypercube design within sensible bounds (Figure 7.8).

We then run the ABM 1000 times, using these 1000 sets of input parameters ($\boldsymbol{\theta}^d$). The the inputs and outputs ($\mathbf{y}^d$) of these 1000 runs we compiled into a single matrix of inputs and a single matrix of linked outputs. Because the ABM is simulating a fishery over 114 years, and the emulator only emulates a single year at a time, we really have 1000 sets of inputs that relate to 1000 $\times$ 114 sets of outputs. As an input design of this di-

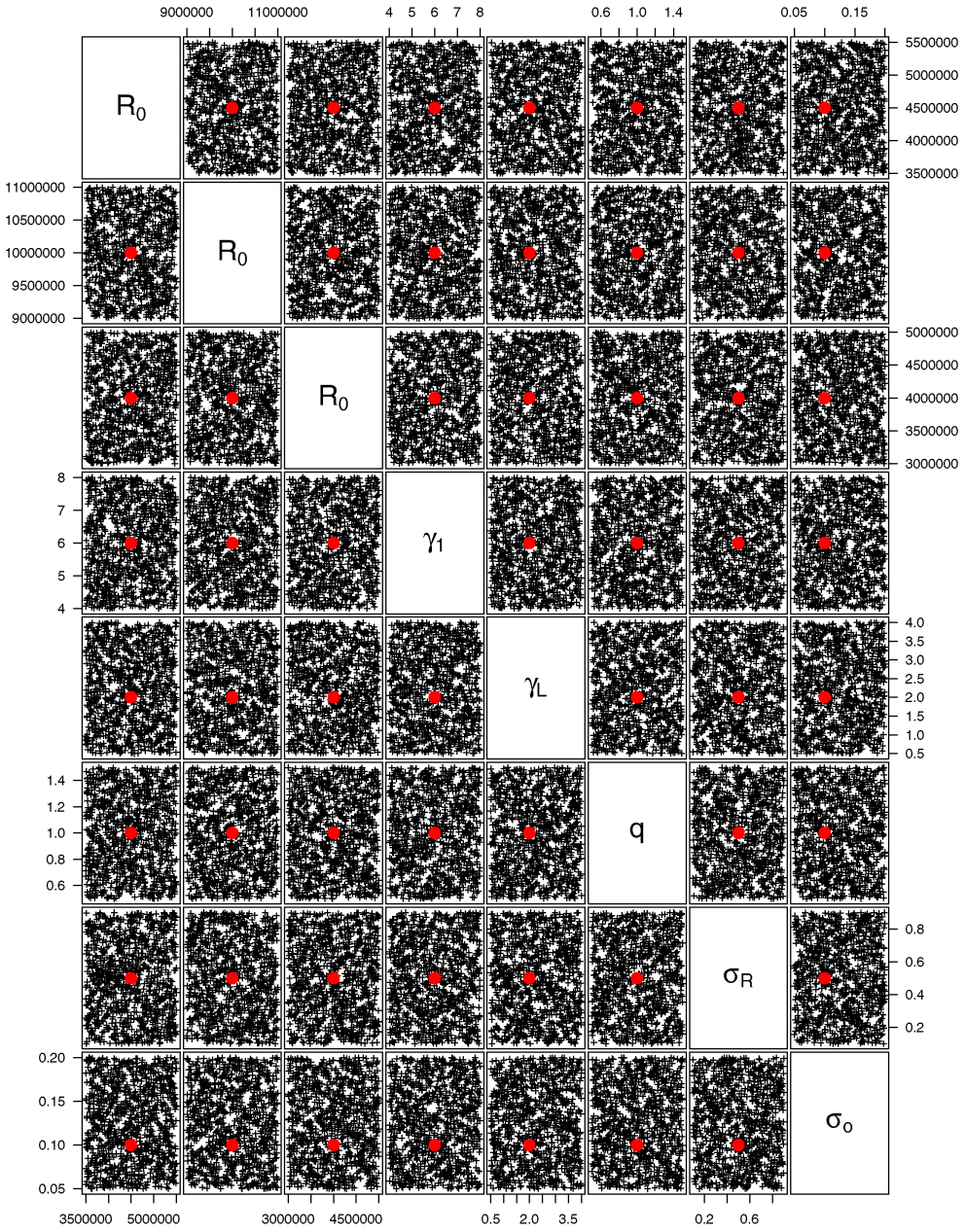


Figure 7.8: The input design ($\boldsymbol{\theta}^d$) for each key parameter that was used to condition the emulator [+] and the true parameter values that were specified in the agent-based simulation model [•].

mension is far to big for use in Bayesian emulation (and probably too big for any computer to handle, I tried inverting this matrix by mistake and crashed my computer), we reduced the dimensionality of the input design by selecting a single output year at random to pair with each input, thus we are back to having and input design of $n = 1000$.

Emulator outputs ($\mathbf{y}_t$)

The multivariate emulator is conditioned to return the vulnerable biomass in the following time-step as its output (i.e. $\mathbf{y}_t = \{V_{t,z}\}$). We test the performance of the emulator by drawing another 1000 input parameters ($\boldsymbol{\theta}$) from the parameter space ($\boldsymbol{\Theta}$), running the ABM another 1000 times using each of these parameter sets to obtain outputs ($\mathbf{y}$), but not using these inputs/outputs to condition the emulator. Instead we provide these inputs to the already conditioned emulator and obtain the emulated outputs also (Figure 7.9).

7.7.3 Inference

We are interested the probabilistic relationship between the following:

- **The data $\mathbf{y}$:** the catch per unit effort by area ($I_{t,z}$). Let $\mathbf{y} = \{\{I_{t,z}\}_{t=1}^{T}\}_{z=1}^{Z}$
- **The covariates $\mathbf{z}$:** the catch by area ($C_{t,z}$). Let $\mathbf{z} = \{\{C_{t,z}\}_{t=1}^{T}\}_{z=1}^{Z}$
- **The unknown process parameters $\boldsymbol{\psi}$:** the virgin recruitment by area ($R_{0,z}$), the selectivity parameters (γ_1 and γ_L), and the recruitment variance (σ_R^2). Let $\boldsymbol{\psi} = \{\{R_{0,z}\}_{z=1}^{Z}, \gamma_1, \gamma_L, \sigma_R^2\}$
- **The unknown observation parameters $\boldsymbol{\phi}$:** the catchability coefficient (q) and the observation error variance (σ_o^2). Let $\boldsymbol{\phi} = \{q, \sigma_o^2\}$
- **The unknown latent states $\mathbf{x}$:** the vulnerable biomass (tonnes) in each area ($V_{t,z}$). Let $\mathbf{x} = \{\{V_{t,z}\}_{t=1}^{T}\}_{z=1}^{Z}$

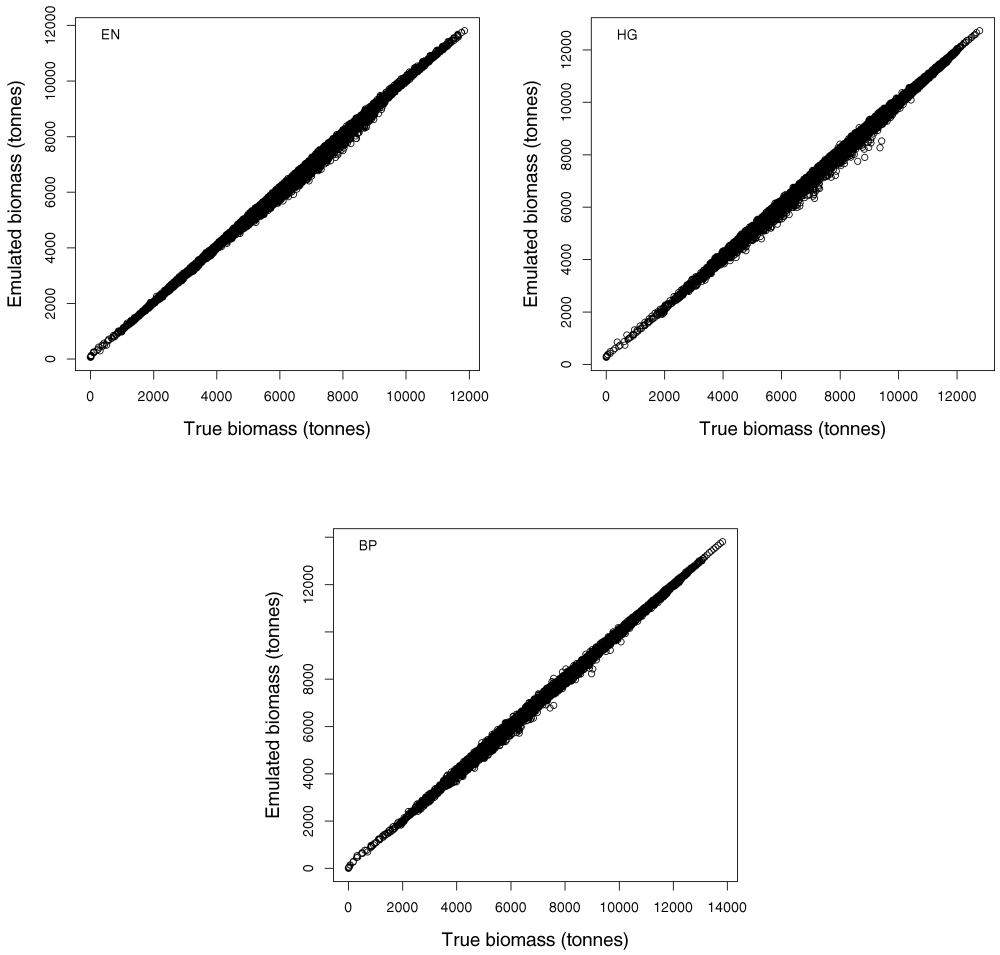


Figure 7.9: The performance of the emulator. The true vulnerable biomass in each area ($V_{t,z}$) produced by the agent-based simulation model and the vulnerable biomass produced by the emulator given the same input parameters.

Using Bayes theorem, the posterior distribution of the model parameters ($\boldsymbol{\psi}$ and $\boldsymbol{\phi}$) and the states ($\mathbf{x}$), given the data ($\mathbf{y}$) and covariates ($\mathbf{z}$) is

$$\pi(\boldsymbol{\psi}, \boldsymbol{\phi}, \mathbf{x}|\mathbf{y}, \mathbf{z}) \propto \pi(\boldsymbol{\psi}, \boldsymbol{\phi}, \mathbf{x}|\mathbf{z})\pi(\mathbf{y}|\mathbf{x}, \boldsymbol{\phi}), \tag{7.71}$$

where

$$\begin{aligned}
\pi\left(\boldsymbol{\psi}, \boldsymbol{\phi}, \mathbf{x}|\mathbf{z}\right) &= \pi\left(R_{0,z}, q, \gamma_{50}, \gamma_{95}, \sigma_o^2, \sigma_R^2, \mathbf{x}|\mathbf{z}\right) \\
&= \pi(R_{0,z})\pi(q)\pi(\gamma_{50})\pi(\gamma_{95})\pi(\sigma_o^2)\pi(\sigma_R^2)\pi\left(\mathbf{x}|\mathbf{z}, R_0, \gamma_{50}, \gamma_{95}, \sigma_R^2\right) \\
&= \pi(R_{0,z})\pi(q)\pi(\gamma_{50})\pi(\gamma_{95})\pi(\sigma_o^2)\pi(\sigma_R^2)\pi(V_{1,z}) \\
&\quad \times \prod_{t=2}^{T}\prod_{z=1}^{Z} \pi\left(V_{t,z}|V_{t-1,z}, C_{t-1,z}, R_{0,z}, \gamma_{50}, \gamma_{95}, \sigma_R^2\right) \\
\pi\left(\mathbf{y}|\mathbf{x}, \boldsymbol{\phi}\right) &= \pi\left(\mathbf{y}|\mathbf{x}, q, \sigma_o^2\right) = \prod_{t=1}^{T}\prod_{z=1}^{Z} \pi\left(I_{t,z}|V_{t,z}, q, \sigma_o^2\right).
\end{aligned} \tag{7.72}$$

The component $\pi\left(\mathbf{x}|\mathbf{z}, R_0, \gamma_{50}, \gamma_{95}, \sigma_R^2\right)$ is the ABM, and is computationally expensive, taking several hours to do a single evaluation, making standard inference impractical. Instead, we condition a multivariate Bayesian emulator to this evolution equation

$$V_{t,z}|\boldsymbol{\theta}, \widehat{\boldsymbol{\beta}}, \widehat{\sigma}_e^2, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q} \sim \mathcal{N}\left(m^{**}\left(\boldsymbol{\theta}'|\widehat{\boldsymbol{\beta}}, \boldsymbol{\theta}^d, \mathbf{y}^d, \mathbf{Q}, \mathbf{V}\right), \widehat{\sigma}_e^2 c^{**}\left(\boldsymbol{\theta}, \boldsymbol{\theta}'|\boldsymbol{\theta}^d, \mathbf{Q}\right)\right), \tag{7.73}$$

where $\boldsymbol{\theta}$ is a vector of emulator inputs (i.e. a collection of parameters, latent states and covariates). Here the emulator is constructed using the inputs $\boldsymbol{\theta} = \{V_{t-1,z}, C_{t-1,z}, R_{0,z}, \gamma_{50}, \gamma_{95}, \sigma_R^2\}$. The likelihood of this model is made up of two main components: the likelihood of the CPUE observations (I_t) and the likelihood of the vulnerable biomass (tonnes) latent states ($V_{t,z}$). The likelihood of the CPUE observations is

$$\log\left(I_{t,z}\right)|V_{t,z}, q, \sigma_o^2, \boldsymbol{\omega}, \sim \mathcal{N}\left(\log\left(qV_{t,z}\right), \sigma_o^2\right). \tag{7.74}$$

The two main components of the likelihood can be split into even smaller subcomponents made up of just single areas z and years t. When a parameter or latent state is proposed within MCMC, then only those subcomponents relevant to the proposal need be evaluated. Each of the MCMC proposals and the subcomponents of the likelihood that need to be evaluated are as follows:

- **When proposing** $R^*_{0,z}$, γ^*_{50}, γ^*_{95} **or** σ^{2*}_R evaluate

$$\log\left(V_{t,z}\right) \sim \mathcal{N}\left(\log\left(\mu_{t,z}\right), \widehat{\sigma}^2_e c^{**}(\cdot)\right) \quad 2 \leq t \leq T, \forall z.$$

- **When proposing** q^* **or** σ^{2*}_o evaluate

$$\log\left(I_{t,z}\right) \sim \mathcal{N}\left(\log\left(qV_{t,z}\right), \sigma^2_o\right) \quad \forall t, \forall z.$$

- **When proposing** $V^*_{t,z}$ evaluate

$$\begin{aligned}
\log\left(I_{t,z}\right) &\sim \mathcal{N}\left(\log\left(qV^*_{t,z}\right), \sigma^2_o\right), \\
\log\left(V^*_{t,z}\right) &\sim \mathcal{N}\left(\log\left(\mu_{t,z}\right), \widehat{\sigma}^2_e c^{**}(\cdot)\right) \quad \forall z, \\
\log\left(V_{t+1,z}\right) &\sim \mathcal{N}\left(\log\left(\mu_{t+1,z}\right), \widehat{\sigma}^2_e c^{**}(\cdot)\right) \quad \forall z.
\end{aligned}$$

Early exploration suggested that MCMC was not mixing well. We therefore specified highly informative log-normal prior distributions for all model parameters in an attempt to improve mixing (see Figure 7.10 for plots of these priors). Despite running a long MCMC chain of 3 million iterations, taking roughly one week to complete (saving every 1000th iteration to yield 3000 samples from the posterior distribution), the MCMC chains still did not mix well (Figure 7.10). The lack of mixing was most pronounced in the catchability coefficient (q). Despite the poor mixing of the key parameters, the log-likelihoods and log-priors appeared to be mixing well (Figure 7.11). The fit to the CPUE data was also excellent (Figure 7.12) and the posterior distribution of the vulnerable biomass in each area matched closely with the simulated vulnerable biomass (Figure 7.13). However, given the highly informative priors placed on the key parameters, this is not surprising. We therefore conclude that Bayesian emulation did not work very well in this example. The root of the problem is its speed, or rather lack of. Despite simplifying the problem considerably by reducing the number of parameters to be emulated to the bare minimum, the emulator was still relatively slow, resulting in a slow MCMC sampler. We consider that an MCMC that takes longer than a week, and does not result in adequately mixed MCMCs, is beyond practical application.

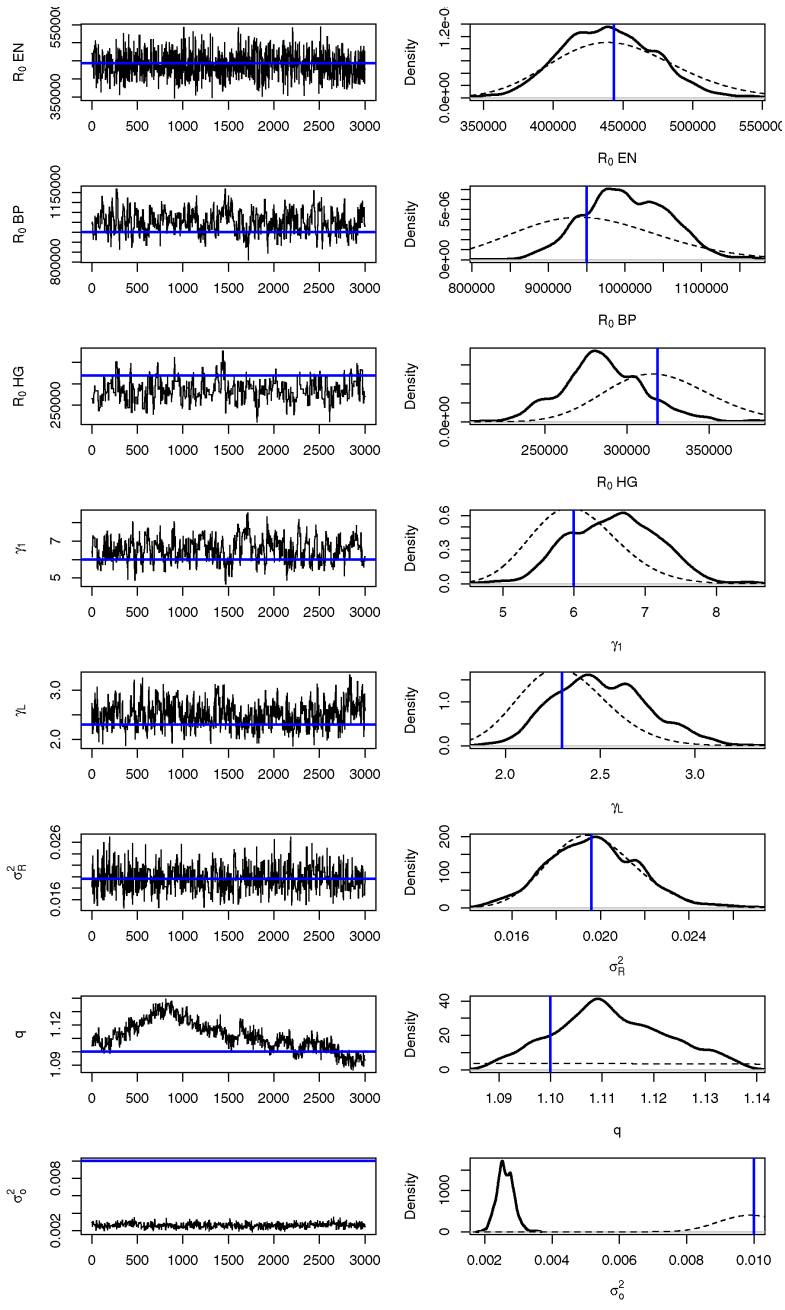


Figure 7.10: MCMC trace plots [left column] and posterior densities [right column] for each of the emulated key model parameters.

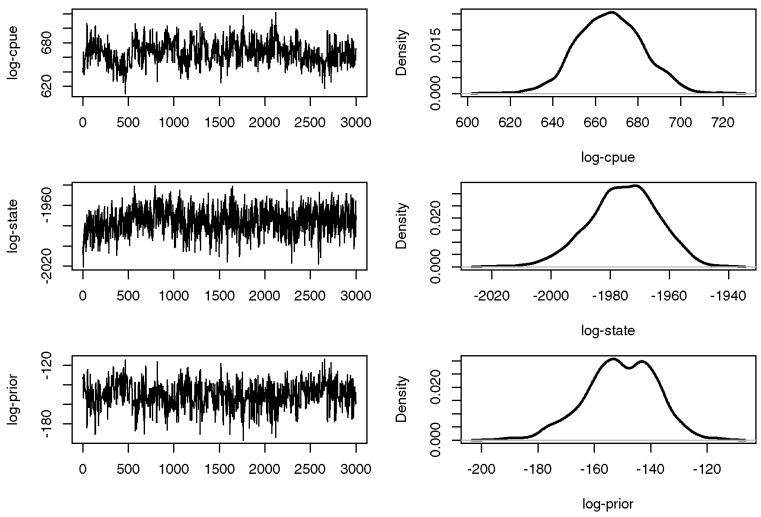


Figure 7.11: MCMC trace plots [left column] and posterior densities [right column] log-likelihood of the CPUE, the log-likelihood of the vulnerable biomass state and the log-prior.

Although the method has fallen short in this example, we have developed a proof of concept for the method, and further developed the method itself. We hope that these beginnings will stimulate further research into Bayesian emulation, in fisheries science or otherwise. We discuss some good starting points for future research below.

7.8 Discussion

This chapter synthesised aspects from all of the preceding chapters of this thesis. Snapper (Chapter 3, page 77) was used as case study species and an agent-based model was developed based broadly on the dynamics of the species in SNA 1 (Chapter 4, page 89). Bayesian emulation (first introduced in Chapter 2, page 71) is covered in more detail, the methods are extended beyond the scope of the current literature, and the methods are applied to fisheries specific problems in a series of examples.

New ideas built into or around the Bayesian emulation framework include emulators that use Moore-Penrose (MP) matrix inversion and stochastic-

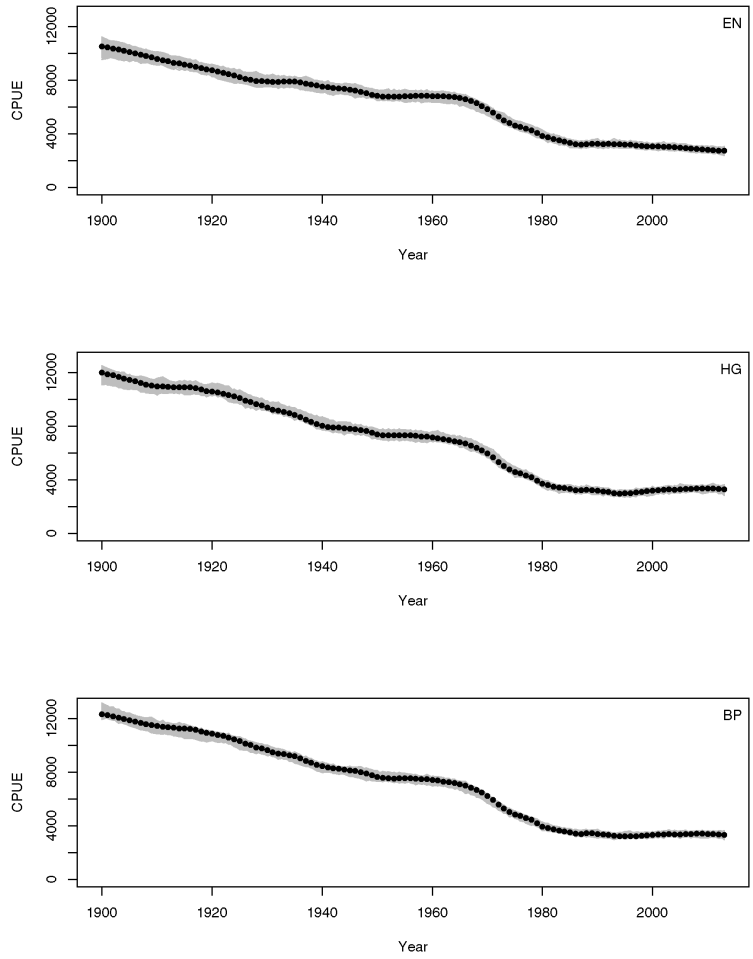


Figure 7.12: Fit to CPUE observations in each of the areas ($I_{t,z}$). CPUE observations are shown as black points [•] and the posterior distribution of the fit to CPUE is shown in grey.

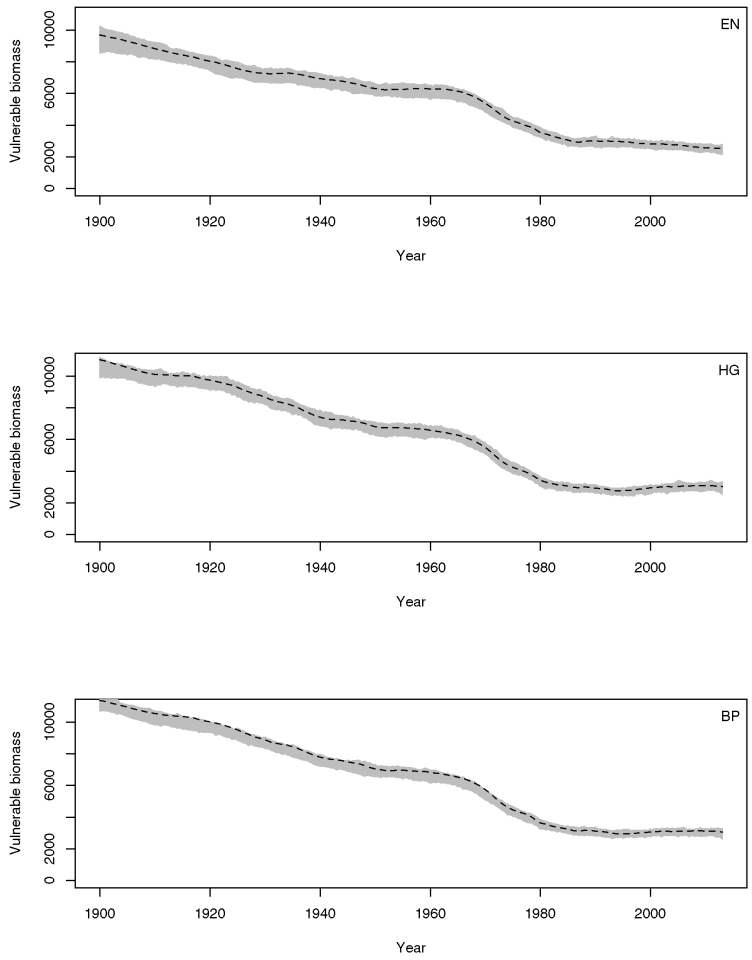


Figure 7.13: Posterior distribution of the vulnerable biomass ($V_{t,z}$) of snapper in East Northland (EN), the Hauraki Gulf (HG) and the Bay of Plenty (BP). The simulated biomass is shown as the dashed black line.

ity. As far as we know, there is no literature that extends the formalism of the Bayesian emulation framework to properly incorporate stochasticity in this way (but see Henderson et al. 2009 which applies to the method to a rather different stochastic model). These emulators are also nested within a state-space framework (state-space models are introduced and discussed in Chapter 5, page 129). The incorporation of stochasticity results in a Bayesian emulator better suited to practical applications in fisheries science because they can deal with stochastic models and better represent their uncertainty. Nesting emulators within state-space models and treating them as components of evolution models is also novel.

The examples presented in this chapter serve as a proof of concept and demonstrate the potential uses of Bayesian emulation in fisheries. While our "toy" example that developed a univariate emulator of a biomass dynamics system (Section 7.6, page 292) was a simple problem that would be better done without emulation, it is a good starting point for discussion and leads in to the more complex multivariate emulators. However, multivariate emulation proved difficult and further research is required to overcome these limitations.

Making emulators fast and efficient is one of the biggest challenges when developing multivariate Bayesian emulators. The whole point of Bayesian emulation is to provide a means by which to speed up the inference process so that we can make at least some inference about computationally expensive models within our lifetimes. However, as we increase the complexity of the models we are emulating, we necessarily increase the dimensionality of our emulator and reduce their speed. In fact, our attempts at developing age-structured emulators resulted in such slow MCMCs that we dropped them.

Despite these challenges, there are ways to improve the speed of these complex emulators. A very simple way to increase their speed is to reduce the number of design points ($\boldsymbol{\theta}^d$ and $\mathbf{y}^d$) that are used by the emulator. Not only are these points used to estimate parameters within the emulator (specifically $\boldsymbol{\beta}$ and σ_e^2), they are also used by the basis function within the emulator as part of the emulator estimate of the output. Although

a lot of the variables that the emulator depends upon can be calculated before attempting to emulate a value for any given input (e.g. the matrix $\mathbf{A}$ or $\boldsymbol{\Sigma}$ can be inverted a priori), one component of the emulator requires solving a linear system. Specifically, in the univariate case, the component highlighted in red below

$$
\begin{aligned}
m^{**}\left(\boldsymbol{\theta}|\widehat{\boldsymbol{\beta}},\boldsymbol{\theta}^d,\mathbf{y}^d,\mathbf{Q}\right) &= \mathbf{h}(\boldsymbol{\theta})^T\widehat{\boldsymbol{\beta}} + \mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q}\right)^T\mathbf{A}^{-1}\left(\mathbf{y}^d - \mathbf{H}\widehat{\boldsymbol{\beta}}\right), \\
c^{**}\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q}\right) &= c^{*}\left(\boldsymbol{\theta},\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q}\right) \\
&\quad + \left(\mathbf{h}(\boldsymbol{\theta})^T - \mathbf{t}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^d,\mathbf{Q}\right)^T\mathbf{A}^{-1}\mathbf{H}\right)\left(\mathbf{H}^T\mathbf{A}^{-1}\mathbf{H}\right)^{-1} \\
&\quad \times \left(\mathbf{h}(\boldsymbol{\theta}')^T - \mathbf{t}\left(\boldsymbol{\theta}'|\boldsymbol{\theta}^d,\mathbf{Q}\right)^T\mathbf{A}^{-1}\mathbf{H}\right)^T.
\end{aligned}
$$

Unfortunately, this is a computationally expensive calculation that cannot be avoided. By reducing the number of design points, we reduce the dimensionality of the system being solved, thus speeding up this component of the emulator. However, reducing the number of design points also reduces the precision of the emulator. Future research is needed here into how one can optimise the number of design points retained and the precision and speed of Bayesian emulators. Alternatives to improve the speed of these methods include better MCMC samplers (as was the case in Chapter 5), and improved parrallelisation of the code so that many different chains can be spawned and run independently on different computer cores.

Other aspects of Bayesian emulation that we spent little time investigating but require further research include:

- methods for optimising the input design ($\boldsymbol{\theta}^d$ and $\mathbf{y}^d$),
- the form of the basis function ($\mathbf{h}(.)$), here we suggest that genetic algorithms may be of use in finding the best functional forms,
- alternative correlation functions ($\mathbf{c}(\boldsymbol{\theta},\boldsymbol{\theta}'|\mathbf{Q})$), Vernon et al. (2010) provides an example of an alternative formulation but there are of course other forms this could take,
- estimating roughness parameters ($\mathbf{Q}$).

With further development, Bayesian emulation could result in the in-

creased ability to consider and evaluate innovative methods and approaches where model complexity is currently a barrier, such as complex fisheries models, ecosystem modelling and climate forecasting. There may also be other potential uses for the Bayesian emulation framework in fisheries research. For example, due to limited resources, it is often not possible to do a stock assessment for some species every year, despite additional information becoming available annually (e.g. abundance indices). Although stock assessment models can and do provide projections of the population into the future, they assume much of dynamics to be constant during these projection years and do not take into account additional information gathered in these years. A Bayesian emulator could be conditioned to the original stock assessment model (during or after the stock assessment has been done) and used to provide more accurate inference about the stock during the years in between stock assessments.

Chapter 8

Conclusions and future research

This thesis is about more realistic models and their inference. This realism may be incorporated by explicitly modelling complex processes, or by admitting our uncertainty and modelling it correctly.

In Chapter 4 (page 9) we described a novel spatially explicit multi-generational **agent-structured fish simulation model**. This flexible model has the potential to consider individual variability in population dynamics, including movement, and spatial heterogeneity in the environment. The aim was to construct a model that is sufficiently rich that it can be used to simulate more complete, realistic fish populations. Complex fisheries models like this, provide a potential framework for exploring complex dynamics in populations, communities and ecosystems. Inference that ignores individual variability and/or spatial complexity may provide biased, imprecise or overly-precise platforms for management advice. Although this model opens up exciting avenues for future practical research, our goal was not to use this model as a test bed, but to apply Bayesian inference instead.

However, the additional complexity in this model comes at the cost of computational time and primary computer memory. The model can take many hours to do a single run, and the larger the population that we want to model, the more memory the model will use. Hence, standard inference procedures (e.g. maximum likelihood or MCMC) are out of the ques-

tion. Therefore, we needed to investigate alternative methods of inference, namely Bayesian emulation. But first, it was necessary to introduce state-space models as they form a foundation for the emulation framework we used.

By incorporating both observation and process error, **state-space models** can help us better quantify the uncertainty of parameters of interest (Harwood & Stokes 2003, Meyer & Millar 1999). Yet, despite considerable interest in state-space models, and previous work suggesting that they have superior performance when compared with their deterministic counterparts (Millar & Meyer 2000), they are not widely used, likely due to their added complexity in implementation.

We developed an age-structured state-space model that includes process error in the numbers at age in the population (Chapter 5, page 129). The major contribution provided in this chapter was the construction of the posterior for this model. While this sophisticated age-structured model has the potential to better represent uncertainty in stock assessment, it pushes the boundaries of the current practical limits of computing and we admit that its practical application remains limited until the MCMC mixing issues that we encountered can be resolved.

Therefore, smarter MCMC proposals are an area worthy of further research. Alternatively, different methods for obtaining samples from the posterior distribution would be worth investigating. We limited our approach to element-wise MCMC proposals and a slightly more complex blockwise update of the diagonal elements of the numbers at age matrix (i.e. a cohort update). Other options to pursue include multidimensional proposals of blocks of parameters (e.g. multivariate normal or multivariate-t updates of all numbers at age latent states simultaneously and element-wise proposals for the remaining key parameters), particle filtering methods also known as Sequential Monte Carlo (SMC, of which there is a rich literature that focuses on applying particle filter methods to state-space systems), or importance sampling (e.g. see Marin & Robert 2010 or Ianelli & McAllister 1997).

If adequate posterior sampling algorithms are identified then models of

this ilk could become commonplace in the future. Stock assessment may change from its humble beginnings on single core computers estimating a few hundred parameters using a deterministic population dynamics core, to fully probabilistic state-space models that estimate thousands of parameters and latent states, and run on hundreds of computer cores from the cloud.

One of the important information needs influencing the stock assessment and management of fish stocks is understanding their movements and migrations. The next component of this thesis took the state-space framework and used it to develop a new process model for modelling the dynamics of fish tagged using **pop-up satellite archival tags** (PSAT; Chapter 6, page 209). We coupled this process model with a new observation model for geolocating fish using depth/bathymetric data and temperature (Chapter 6). Our aim was to develop a method for estimating the path taken by a fish between tag-release and tag-recapture location. The difference between this model and other models is that our process model is fully conditional on the start and end location, which are basically the best two pieces of information we have.

Simulation suggested that, given accurate data, the method should be able to accurately estimate a fish's path, providing a proof of concept for this modelling framework. We note that the use of depth data (as well as temperature) to help estimate position is novel. However, our application to Antarctic toothfish was not entirely successful, attributable to the quality of the temperature data available in the Ross Sea and the poor quality of the magnetic field data collected by the tag itself.

We discuss several points for further research at the end of Chapter 6, but the major improvements will likely come from the addition of more observation models. Specifically, light (e.g. Welch & Eveson 1999), and more importantly, magnetic field strength. While magnetic field strength was not used here due to a lack of contrast and high variation in the data collected by the tag, we suggest that this is an area well worth future research. Several NOAA scientists are working on tagging a variety of species using PSAT technology in the northern hemisphere and they have found that

newer versions of the PSATs seem to provide more accurate magnetic field strength data (i.e. the technology/sensors have improved; K. Echave, K. Coutre and J. Nielsen Pers. Comm.). Future collaboration with them is likely to yield some exciting results. With the addition of light and magnetic field observation models, this modelling framework could be a powerful tool for modelling PSAT data and fish movement in the near future.

Finally, we further develop methods based on an emerging statistical theory known as **Bayesian emulation** (Haylock & O'Hagan 1996, Oakley & O'Hagan 2002, Vernon et al. 2010, Chapter 7, page 259). Within Bayesian inference, this approach replaces a computationally expensive model with an approximating algorithm called an emulator that is calibrated using relatively few runs of the original model. A good emulator provides a close approximation to the original model and has significant speed gains. The emulator allows us to interpolate (or extrapolate) the evaluations of the simulator to beliefs about the simulator output for any input and conversely to make inferences about the "best inputs", conditional on a given data set, thus making inference of computationally expensive models more tractable.

We have developed emulators into a tool that is more suitable for use in fisheries science by nesting them within the state-space framework and developing stochastic emulators. However, although the Bayesian emulation concept is simple, the technical aspects of Bayesian emulation are not. Not only does it make use of several of complex statistical methods, good programming skills are also necessary in order to speed up the method. In this thesis we developed a somewhat rudimentary proof of concept for emulation within a fisheries context. Subsequently, many aspects require further work to overcome some of the limitations.

Making emulators fast and efficient is one of the biggest challenges when developing multivariate Bayesian emulators. The whole point of Bayesian emulation is to provide a means by which to speed up the inference process so that we can make at least some inference about computationally expensive models. However, as we increase the complexity of the models we are emulating, we necessarily increase the dimensionality of our emulator

and reduce their speed. In fact, our attempts at developing age-structured emulators resulted in such slow MCMCs that we dropped them. The alternative spatially-explicit version that tracks vulnerable biomass, rather than numbers, did speed up the evaluation of the original function so that the emulator took about 10 seconds to do a single evaluation (rather than 16 hours for the ABM). But this is still too long within MCMC.

Therefore, considerable improvements are needed to make this method practical. Research should therefore focus on: better parallelisation of the code so that many different chains can be spawned and run independently on different computer cores; methods for optimising the input design; alternative basis functions and methods for finding the best functional forms; and better estimation of parameters that are fixed a priori such as the roughness scales.

With further development, Bayesian emulation could result in the increased ability to consider and evaluate innovative methods and approaches where model complexity is currently a barrier, such as complex fisheries models, ecosystem modelling and climate forecasting. There may also be other potential uses for the Bayesian emulation framework in fisheries research.

To summarise, this thesis developed three proof of concepts: the construction of the posterior for a state-space age-structured stock assessment model; improving fish geolocation methods using PSAT data within a state-space framework and the use of new types of data; and the development of Bayesian emulation methods within a fisheries context. We have made a start on the development of a tractable approach to fisheries modelling in complex settings through the creation of realistic models, and their emulation. These are all complex problems and therefore we only really made a start in each case. However, we do identify many issues, so we finish by providing a list of what we consider to be the most important issues identified in this thesis that are worthy of future research:

- smarter proposals are required that speed up mixing for the numbers at age and time ($N_{a,t}$) latent states in age-structured state-space models. Alternatively, different inference methods like particle fil-

tering or importance sampling algorithms might be better suited to these high-dimensional state-space problems;

- the addition of a magnetic field strength observation model, and of lesser importance a light observation model, into our PSAT modelling framework and the application to this methodology to species other than Antarctic toothfish;
- development of Bayesian emulation should focus on alternative basis functions and better estimation of parameters that are fixed a prior (e.g. the roughness scales);
- further research into the potential of the programming language Julia for inference in fisheries research. It would be interesting to know how well Julia does compared with the likes of AD Model Builder, CASAL and STAN. Current benchmarks suggest that Julia has speed approaching that of C (`http://julialang.org/benchmarks/`). Furthermore, we only scratched the surface of Julia's multi-threading capability and suspect that, if properly harnessed, the multi-threading abilities available in Julia may be able to overcome many of the computational challenges faced in this thesis, and fisheries science in general, through brute force (e.g. running the model on hundreds of computer cores simultaneously on the cloud).

Appendix A

The log-normal distribution

The log-normal distribution is perhaps the most important distribution in fisheries science. Because it is used so often throughout this document, we provide this appendix to describe some of the properties of the log-normal distribution.

A.1 Probability density function (PDF)

The log-normal distribution has a location parameter μ ($\mu \in \mathbb{R}$) and a scale parameter σ ($\sigma > 0$). The probability density function (PDF) of a log-normal distribution is

$$f(x|\mu,\sigma) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-\frac{(\log(x)-\mu)^2}{2\sigma^2}} \quad (x > 0), \tag{A.1}$$

with support across $x \in (0, \infty)$. The log-normal distribution has mean $e^{\mu+\sigma^2/2}$, median e^{μ}, mode $e^{\mu+\sigma^2}$ and variance $(e^{\sigma^2} - 1)e^{2\mu+\sigma^2}$.

A.2 Expectation

If we have a random variable α that is assumed to be log-normally distributed with variance σ^2 we can write

$$\alpha = ae^{\eta} \quad \text{where} \quad \eta \sim \mathcal{N}\left(-\sigma^2/2, \sigma^2\right),$$

or

$$\alpha = ae^{\varepsilon - \sigma^2/2} \quad \text{where} \quad \varepsilon \sim \mathcal{N}\left(0, \sigma^2\right),$$

noticing the need for the adjustment term $-\sigma^2/2$. Here we prove that the expectation of a log-normal is $e^{\varepsilon - \sigma^2/2}$. We start by defining

$$\begin{aligned}
\varepsilon &\sim \mathcal{N}\left(0, \sigma^2\right), \\
f(\varepsilon) &= \left(2\pi\sigma^2\right)^{-\frac{1}{2}} e^{-\frac{1}{2}\varepsilon^2}, \\
x &= e^{\varepsilon}, \\
\frac{dx}{d\varepsilon} &= e^{\varepsilon}, \\
f(x) &= f(\varepsilon)\frac{d\varepsilon}{dx} = f(\varepsilon)\frac{1}{e^{\varepsilon}} = f(\varepsilon)\frac{1}{x} \\
&= \frac{1}{x}\left(2\pi\sigma^2\right)^{-\frac{1}{2}} e^{-\frac{1}{2\sigma^2}\varepsilon^2} \\
&= \frac{1}{x}\left(2\pi\sigma^2\right)^{-\frac{1}{2}} e^{-\frac{1}{2\sigma^2}(\log x)^2}.
\end{aligned}$$

The expected value of x is therefore

$$\begin{aligned}
\mathbb{E}[x] &= \int_0^{\infty} x f(x)\, dx \\
&= \int_{-\infty}^{\infty} e^{\varepsilon} f(\varepsilon)\, d\varepsilon \\
&= \int_{-\infty}^{\infty} e^{\varepsilon}\left(2\pi\sigma^2\right)^{-\frac{1}{2}} e^{-\frac{1}{2\sigma^2}\varepsilon^2}\, d\varepsilon \\
&= \left(2\pi\sigma^2\right)^{-\frac{1}{2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}\varepsilon^2 + \varepsilon}\, d\varepsilon \\
&= \left(2\pi\sigma^2\right)^{-\frac{1}{2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}(\varepsilon^2 - 2\sigma^2\varepsilon)}\, d\varepsilon \\
&= \left(2\pi\sigma^2\right)^{-\frac{1}{2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}((\varepsilon-\sigma^2)^2 - \sigma^4)}\, d\varepsilon \\
&= \left(2\pi\sigma^2\right)^{-\frac{1}{2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}(\varepsilon-\sigma)^2}\, d\varepsilon e^{-\frac{\sigma^4}{2\sigma^2}\sigma^4} \\
&= \left[\left(2\pi\sigma^2\right)^{-\frac{1}{2}} \int_{-\infty}^{\infty} e^{-\frac{1}{2\sigma^2}(\varepsilon-\sigma)^2}\, d\varepsilon\right] e^{-\frac{1}{2}\sigma^2} \\
&= 1 \cdot e^{-\frac{1}{2\sigma^2}} \\
&= e^{-\sigma^2/2}.
\end{aligned}$$

A.3 Using a log-normal proposal distribution

If using a log-normal proposal distribution within a Metropolis-Hastings MCMC algorithm we can simplify the proposal ratio from $\frac{q_\theta(\theta^{(i-1)}|\theta^*,\mathbf{y})}{q_\theta(\theta^*|\theta^{(i-1)},\mathbf{y})}$ to $\frac{\theta^*}{\theta^{(i-1)}}$. To prove this, we begin by drawing the random variable x from a normal distribution with mean μ and variance σ^2

$$x \sim \mathcal{N}\left(\mu, \sigma^2\right).$$

We know the probability density function (PDF) of a normal distribution to be

$$f(x|\mu,\sigma) = \left(2\pi\sigma^2\right)^{\left(-\frac{1}{2}\right)} e^{-\frac{1}{2\sigma^2}(x-\mu)^2}.$$

If we define $y = e^x$ then we have $x = \log(y)$. The Jacobian is

$$\left|\frac{dx}{dy}\right| = \left|\frac{1}{y}\right|.$$

We can then state that

$$\begin{aligned} f(y^*) &= f(x)\left|\frac{dx}{dy}\right| \\ &= \left(2\pi\sigma^2\right)^{\left(-\frac{1}{2}\right)} e^{-\frac{1}{2\sigma^2}(x-\mu)^2} \times \frac{1}{y^*}, \end{aligned}$$

or for a proposed value θ^* from a log-normal with mean $\mu = \log(\theta)$ and variance σ^2

$$q\left(\theta^*|\theta,\sigma^2\right) = \left(2\pi\sigma^2\right)^{\left(-\frac{1}{2}\right)} e^{-\frac{1}{2\sigma^2}(\log(\theta^*)-\log(\theta))^2} \cdot \frac{1}{\theta^*}.$$

Therefore, we draw

$$\log\left(\theta^*\right) \sim \mathcal{N}\left(\log(\theta), \sigma^2\right),$$

and use the proposal ratio

$$\frac{q(\theta|\theta^*)}{q(\theta^*|\theta)} = \frac{\frac{1}{\theta}}{\frac{1}{\theta^*}} = \frac{\theta^*}{\theta}. \tag{A.2}$$

Appendix B

Age-structured state-space models

The first section of this appendix provides a proof related to the equilibrium numbers at age in an age-structured model. The remainder of this appendix provides output and MCMC diagnostic plots for the age-structured state-space models described in Chapter 5 (page 163).

B.1 Equilibrium numbers at age proof

The equilibrium numbers at age (N^0_a) for the plus group ($a = A$) can be calculated using

$$N^0_{a=A} = \sum_{a=A}^{\infty} R_0 e^{-(a-1)M} = R_0 \frac{e^{M-AM}}{1-e^{-M}}.$$

Here we provide the proof that $\sum_{a=A}^{\infty} R_0 e^{-(a-1)M} = R_0 \frac{e^{M-AM}}{1-e^{-M}}$ using the geometric series rules

$$\sum_{k=0}^{n-1} ar^k = a\frac{1-r^k}{1-r} \text{if} \quad r \neq 1,$$

$$\sum_{k=0}^{\infty} ar^k = \frac{a}{1-r} \text{if} \quad |r| < 1,$$

$$\sum_{k=a}^{b} r^k = \frac{r^a - r^{b+1}}{1-r} \quad \text{if } r \neq 1.$$

Given that $N_a^0 = R_0 e^{-(a-1)M}$ for $1 \leq a < A$, we can see that $N_a^0 = \sum_{a=A}^{\infty} R_0 e^{-(a-1)M}$ for $a = A$, thus

$$\begin{aligned}
\sum_{a=A}^{\infty} R_0 e^{-(a-1)M} &= R_0 \sum_{a=A}^{\infty} e^{-(a-1)M} \\
&= R_0 \sum_{a=A}^{\infty} e^{-aM+M} \\
&= R_0 \sum_{a=A}^{\infty} e^{-aM} e^{M} \\
&= R_0 e^{M} \sum_{a=A}^{\infty} (e^{-M})^a \\
&= R_0 e^{M} \frac{(e^{-M})^A - \cancel{(e^{-M})^{\infty}}}{1 - e^{-M}} \\
&= R_0 e^{M} \frac{(e^{-M})^A}{1 - e^{-M}} \\
&= R_0 \frac{e^{M-AM}}{1 - e^{-M}} \quad \text{Q.E.D.}
\end{aligned}$$

B.2 Model validation

These plots are relevant to the MCMC that was used for **model validation** (i.e. all of the key model parameters were fixed to their true values during MCMC and only numbers at age and time latent states were estimated). Discussion of this MCMC can be found in Chapter 5 (page 180).

B.3 Model fit (fixed process error)

These plots are relevant to the MCMC that attempted to estimate all but one of the key parameters (i.e. the **process error** parameter was **fixed** at the value specified in the simulation). Discussion of this MCMC can be found in Chapter 5 (page 191).

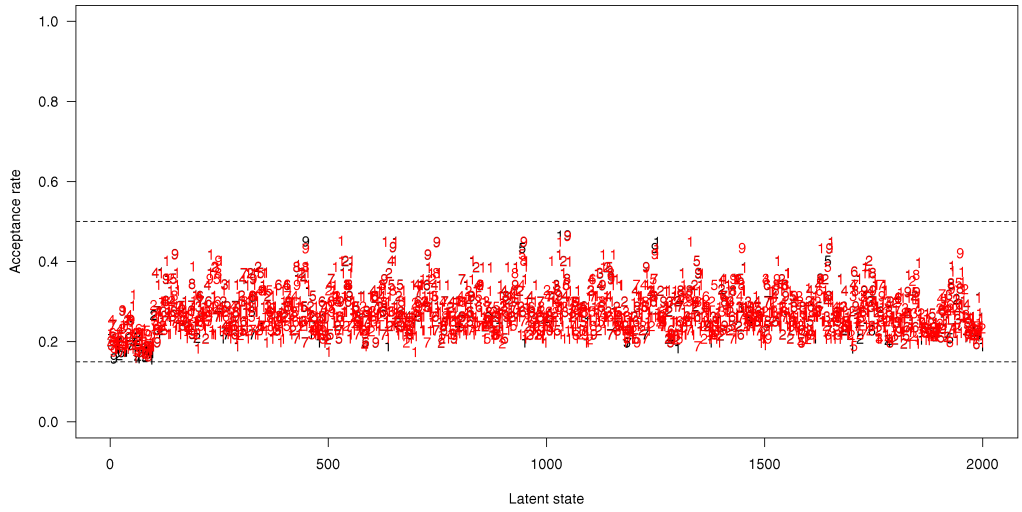


Figure B.1: MCMC acceptance rates for each of the numbers at age and time ($N_{a,t}$) latent states [bottom] in the **model validation** run. It is recommended that the acceptance rate in Metropolis-Hastings MCMC be between 15 and 50%, this range is indicated by the dashed lines. The age of the latent state that each acceptance refers to is indicated numerically (i.e. the point "1" refers to an age-1 latent state). The colours refer to each of the two MCMC chains that were run.

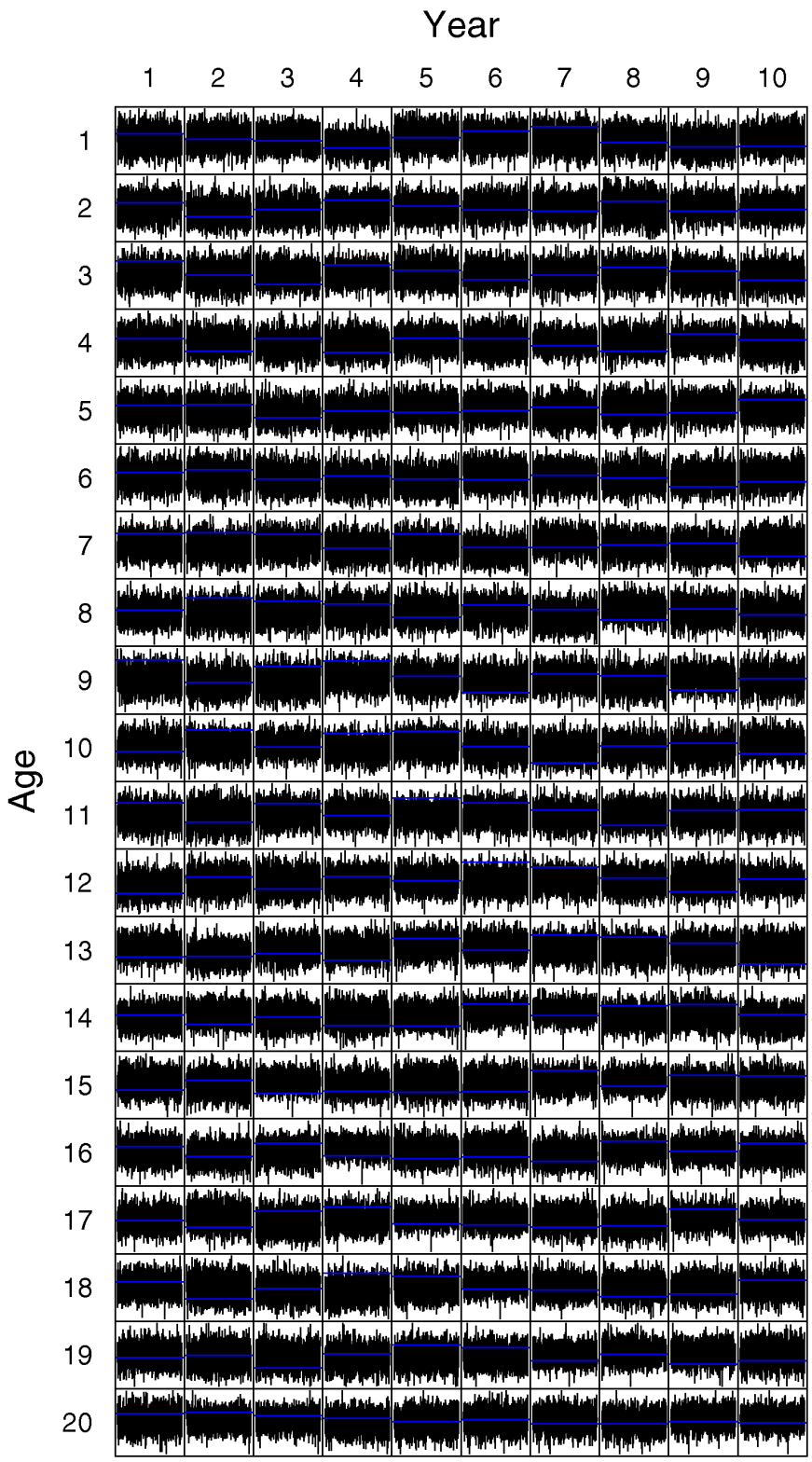


Figure B.2: MCMC trace plots for numbers at age latent states ($N_{a,t}$) during the first 10 years of the **model validation** run. The simulated truth is shown as a horizontal blue line.

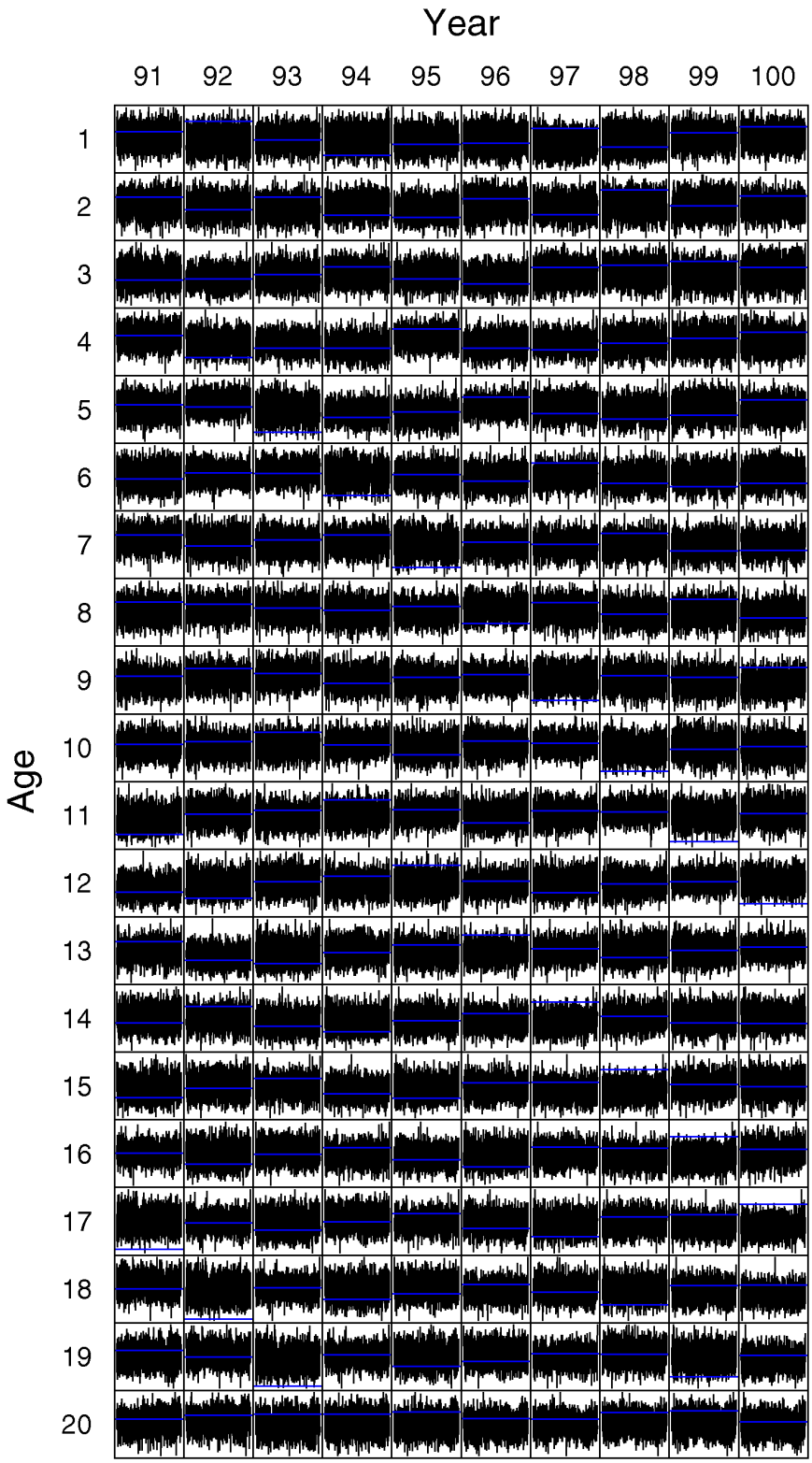


Figure B.3: MCMC trace plots for numbers at age latent states ($N_{a,t}$) during the last 10 years of the **model validation** run. The simulated truth is shown as a horizontal blue line.

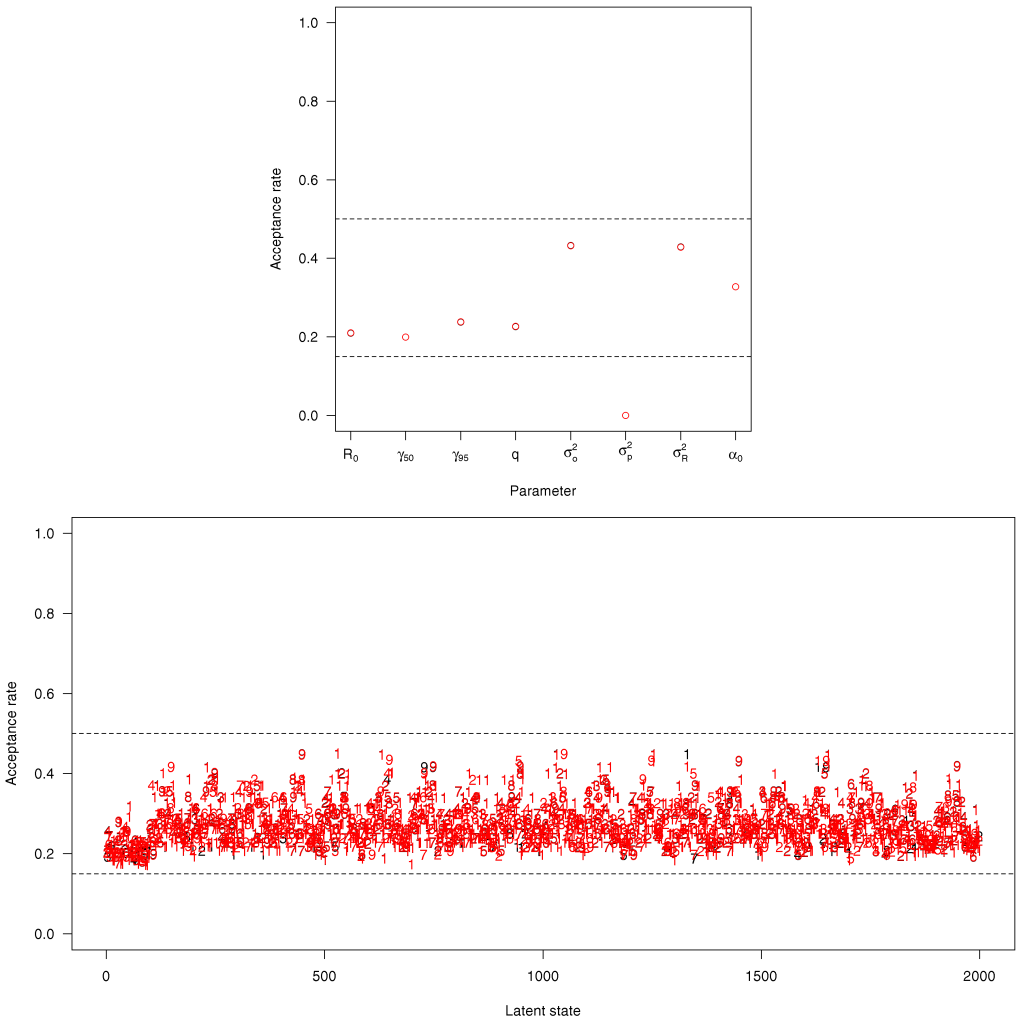


Figure B.4: MCMC acceptance rates for each of the model parameters [top] and each of the numbers at age and time ($N_{a,t}$) latent states [bottom] in the **fixed process error** run. It is recommended that the acceptance rate in Metropolis-Hastings MCMC be between 15 and 50%, this range is indicated by the dashed lines. The age of the latent state that each acceptance refers to is indicated numerically (i.e. the point "1" refers to an age-1 latent state). The colours refer to each of the two MCMC chains that were run.

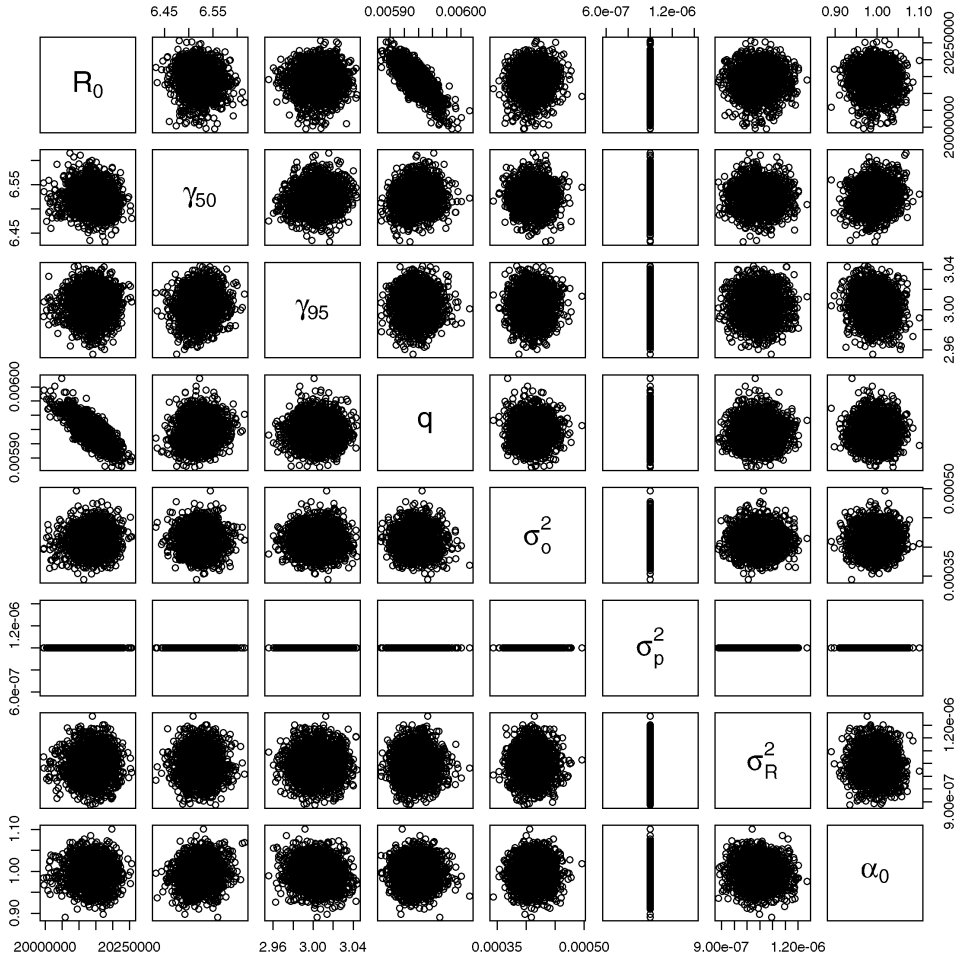


Figure B.5: MCMC correlation plots for each of the key model parameters in the **fixed process error** run.

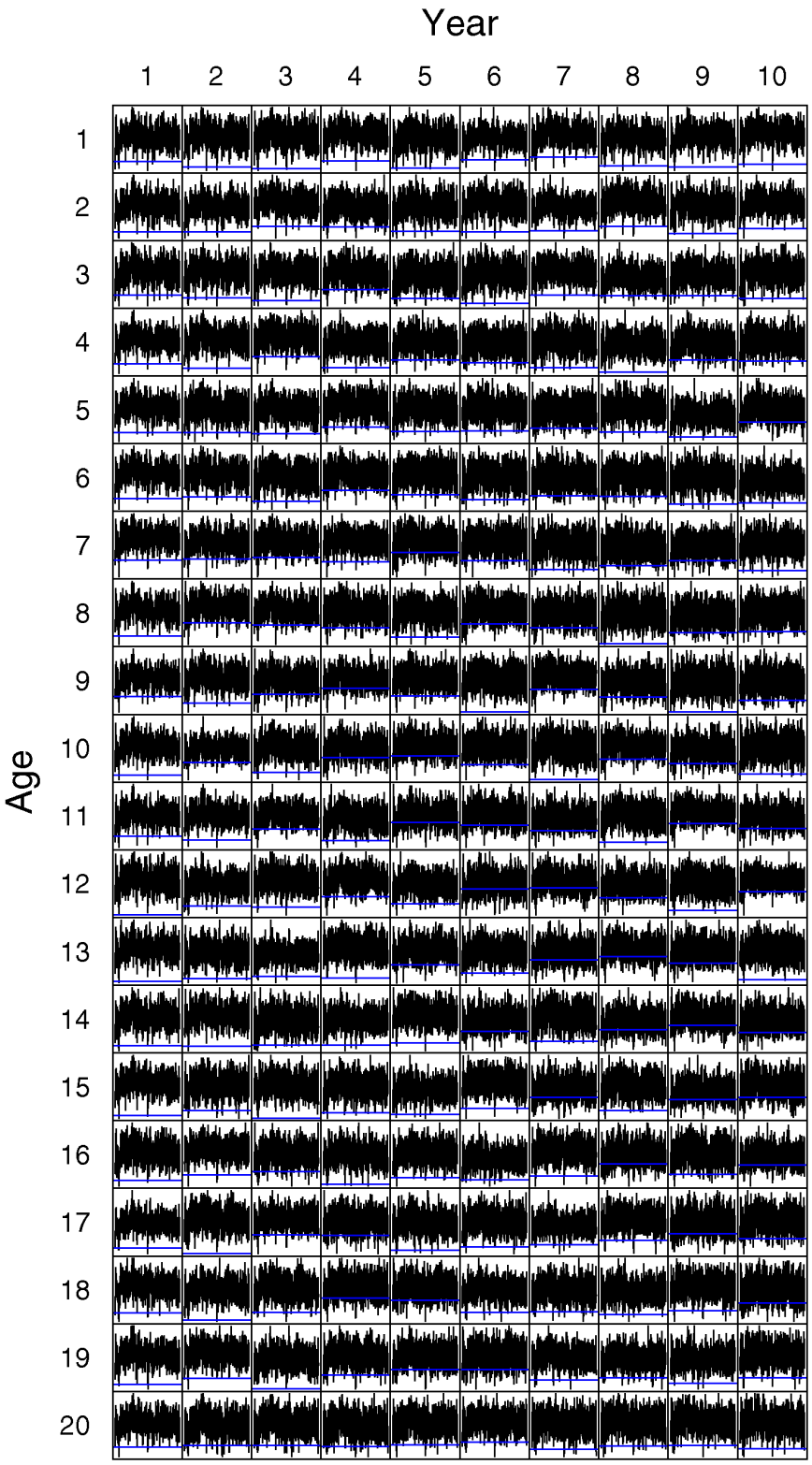


Figure B.6: MCMC trace plots for numbers at age latent states ($N_{a,t}$) during the first 10 years of the **fixed process error** run. The simulated truth is shown as a horizontal blue line.

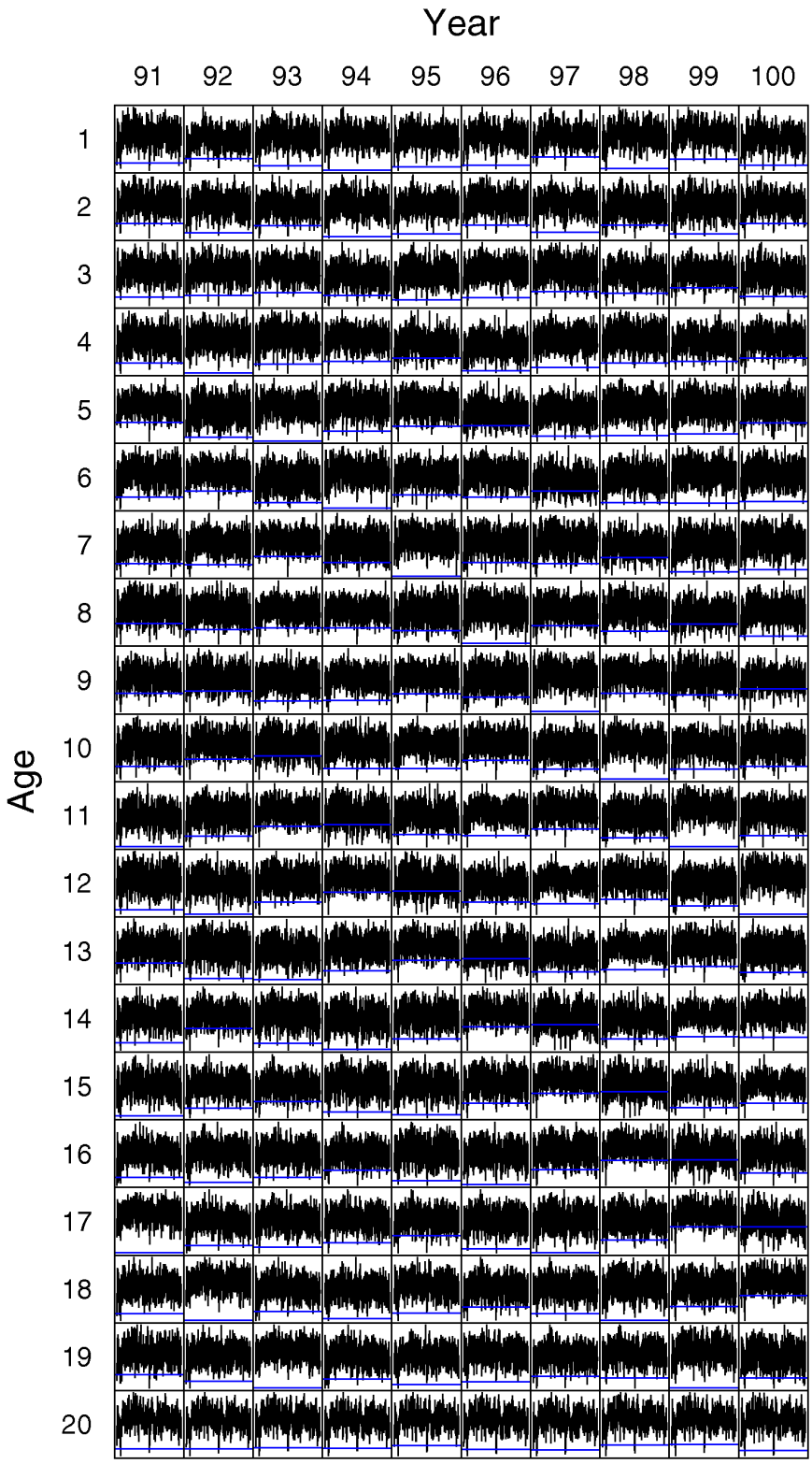


Figure B.7: MCMC trace plots for numbers at age latent states ($N_{a,t}$) during the last 10 years of the **fixed process error** run. The simulated truth is shown as a horizontal blue line.

B.4 Model fit (releasing σ_R^2)

These plots are relevant to the MCMC that attempted to estimate all but one of the key parameters (i.e. the process error parameter was fixed at the value specified in the simulation) and in which a **higher recruitment** variance was used. Discussion of this MCMC can be found in Chapter 5 (page 201).

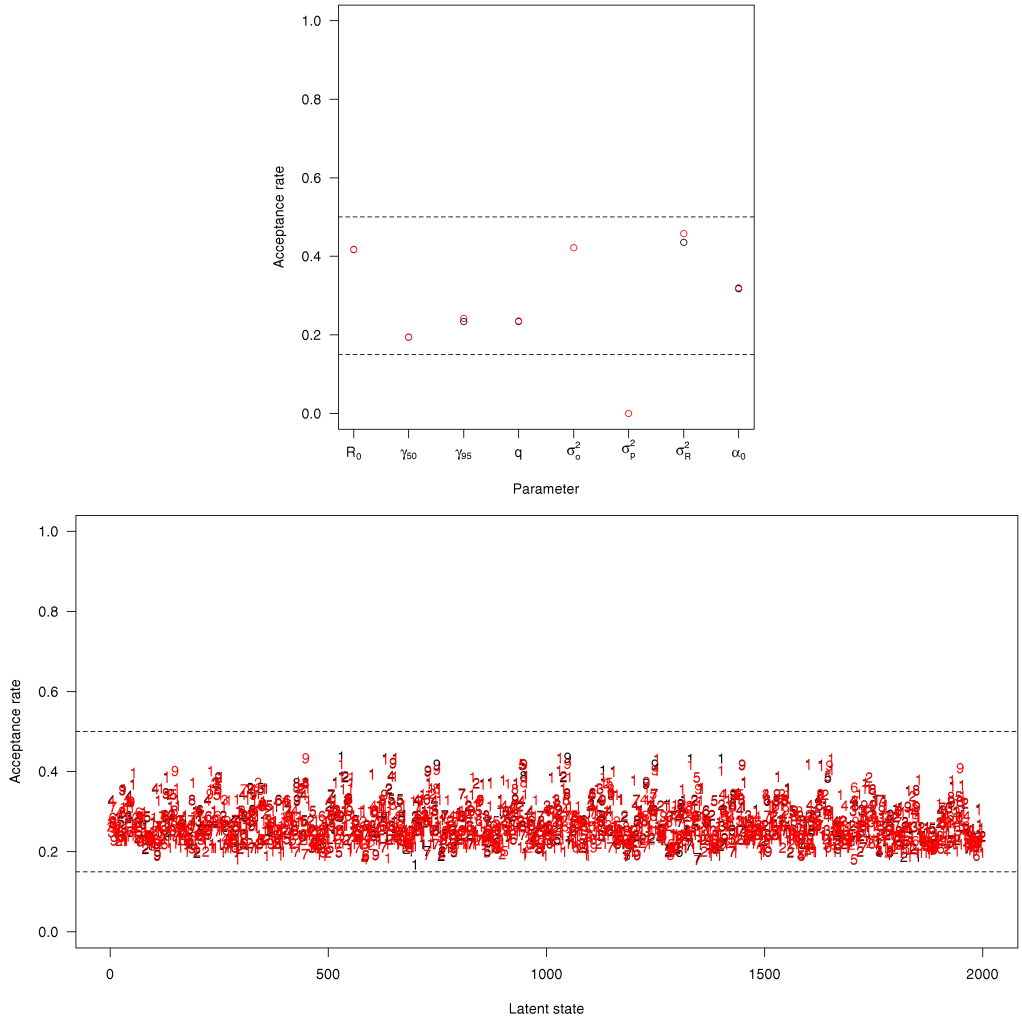


Figure B.8: MCMC acceptance rates for each of the model parameters [top] and each of the numbers at age and time ($N_{a,t}$) latent states [bottom] in the **higher recruitment** run. It is recommended that the acceptance rate in Metropolis-Hastings MCMC be between 15 and 50%, this range is indicated by the dashed lines. The age of the latent state that each acceptance refers to is indicated numerically (i.e. the point "1" refers to an age-1 latent state). The colours refer to each of the two MCMC chains that were run.

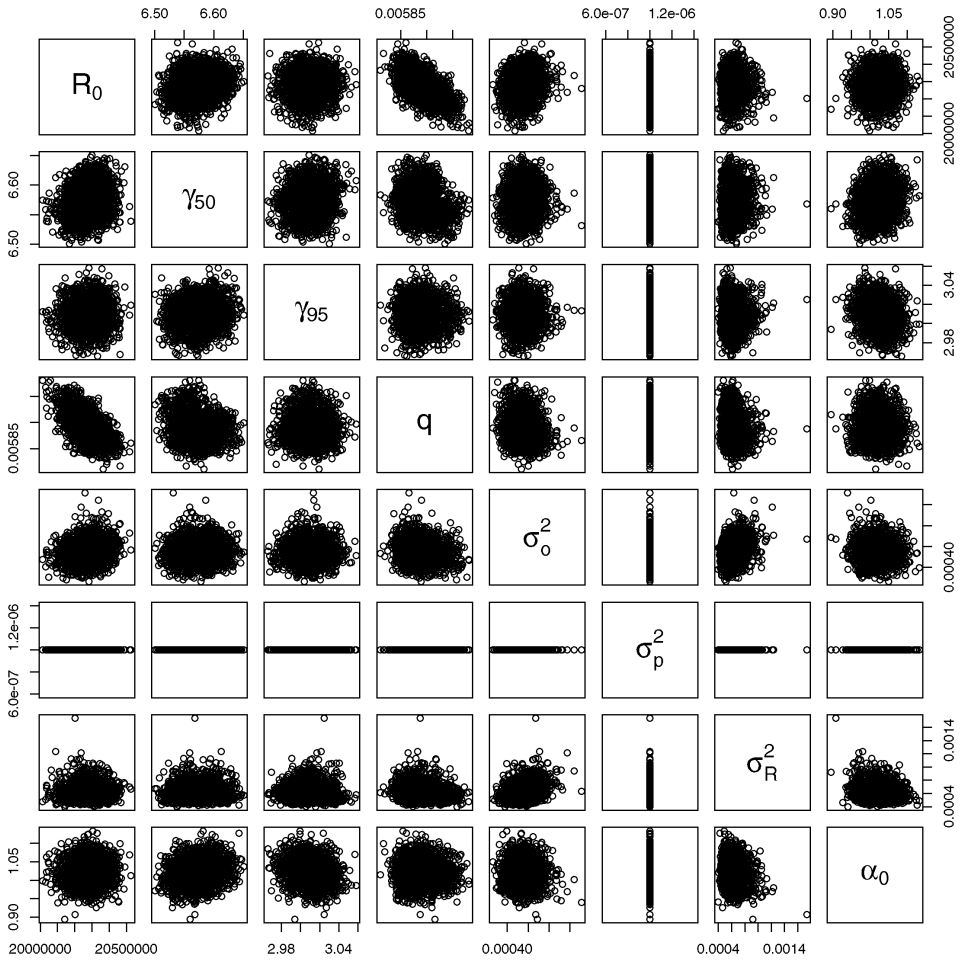


Figure B.9: MCMC correlation plots for each of the key model parameters in the **higher recruitment** run.

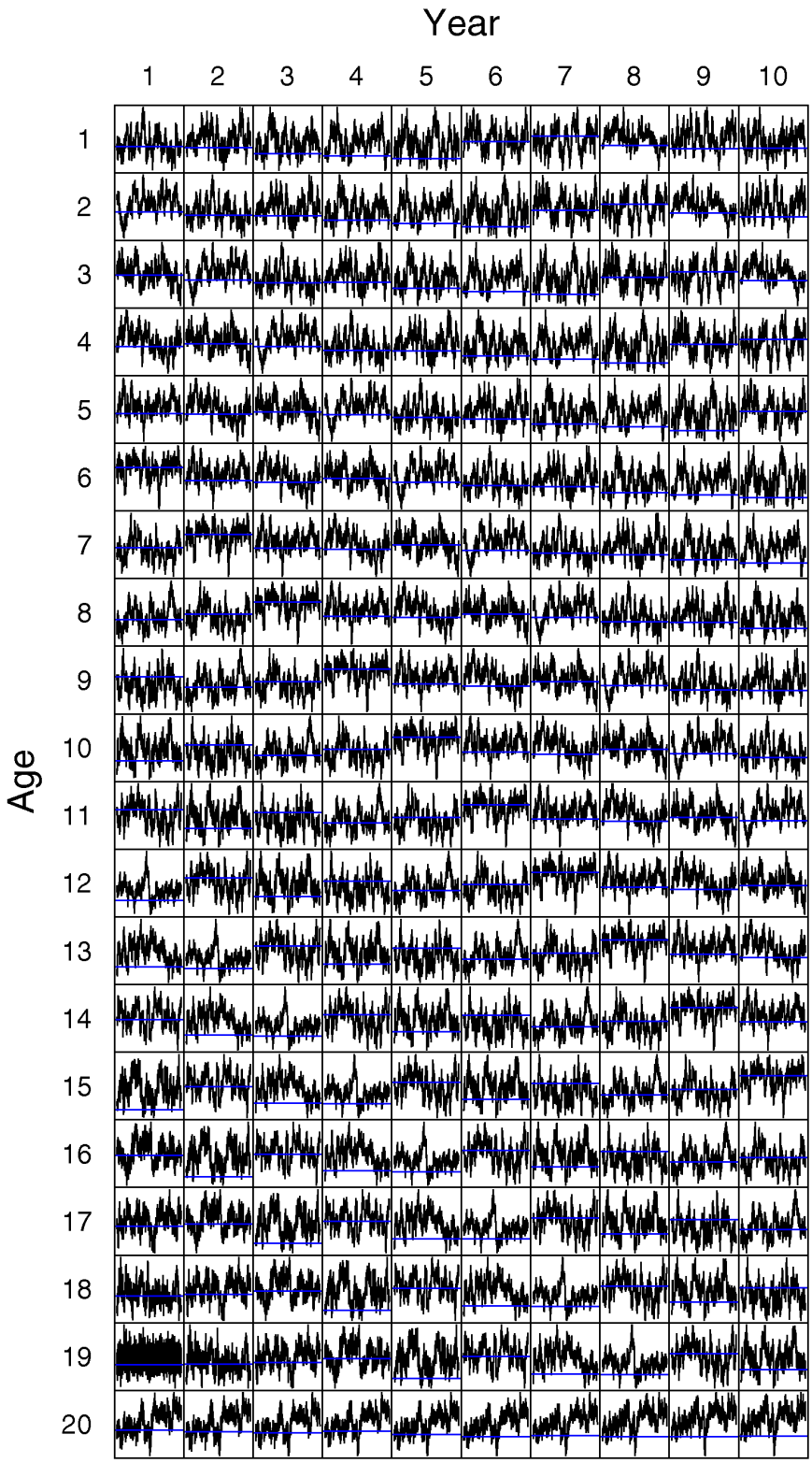


Figure B.10: MCMC trace plots for numbers at age latent states ($N_{a,t}$) during the first 10 years of the **higher recruitment** run. The simulated truth is shown as a horizontal blue line.

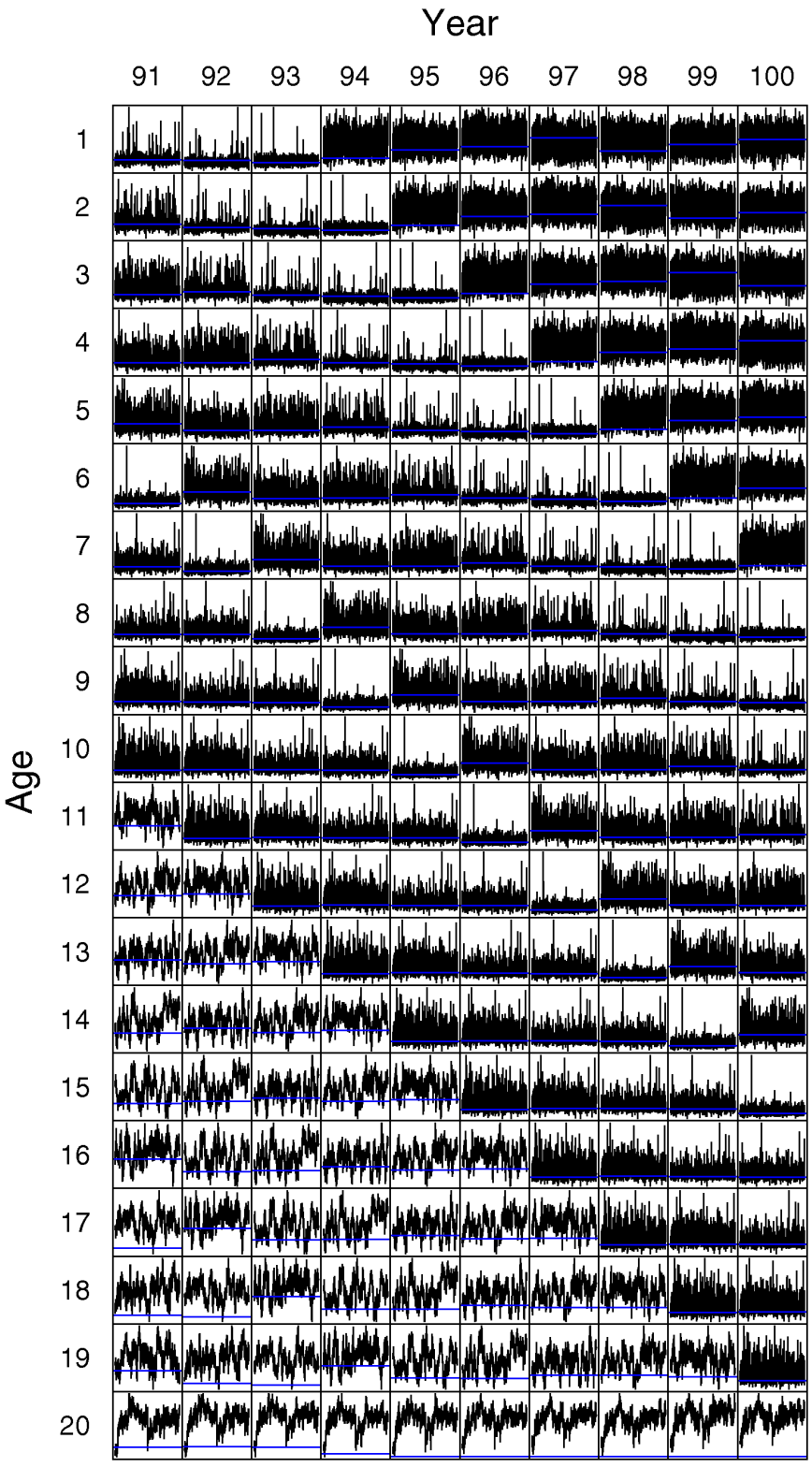


Figure B.11: MCMC trace plots for numbers at age latent states ($N_{a,t}$) during the last 10 years of the **higher recruitment** run. The simulated truth is shown as a horizontal blue line.

Appendix C

Pop-up satellite archival tagging

C.1 GPS coordinates for tag 186

Table C.1 gives the GPS coordinates of the towed tag (tag 186).

C.2 MCMC diagnostics

Table C.1: Date (during February 2012), time, latitude and longitude recorded along towed tag (tag 186) track.

Date	Time	Latitude	Longitude	Date	Time	Latitude	Longitude
22	04:21	-75.750556	168.985556	24	11:00	-68.685000	177.984722
22	05:00	-75.651944	169.168611	24	12:00	-68.500833	178.002222
22	06:00	-75.533611	169.168611	24	13:00	-68.385556	178.018889
22	07:00	-75.419167	169.650278	24	14:00	-68.235000	178.051944
22	09:00	-75.200000	170.118611	24	15:00	-68.085278	178.083611
22	10:00	-75.035833	170.402500	24	16:00	-67.917778	178.118056
22	11:00	-74.935278	170.616944	24	17:00	-67.784167	178.166667
22	12:00	-74.818056	170.850833	24	18:00	-67.635278	178.200556
22	13:00	-74.685833	171.084722	24	19:00	-67.500556	178.219167
22	14:00	-74.567500	171.318333	24	20:00	-67.350000	178.233333
22	15:00	-74.450000	171.550833	24	22:00	-67.035278	178.318889
22	16:00	-74.318333	171.784444	24	23:00	-67.900556	178.366944
22	17:00	-74.184444	172.033333	25	01:00	-66.585556	178.367500
22	18:00	-74.067778	172.268056	25	02:00	-66.451111	178.301667
22	19:00	-73.935833	172.517500	25	03:00	-66.296111	178.235556
22	20:00	-73.752500	172.768611	25	04:00	-66.134722	178.168611
22	22:00	-73.535556	173.235833	25	05:00	-66.000833	178.101389
23	00:10	-73.283333	173.666667	25	06:00	-65.818333	178.016944
23	01:10	-73.135833	173.905000	25	07:00	-65.668333	177.951944
23	02:00	-73.018056	174.100278	25	08:00	-65.551389	177.901944
23	04:00	-72.770833	174.436667	25	09:00	-65.383611	177.834167
23	05:00	-72.651111	174.584444	25	10:00	-65.202222	177.767500
23	06:00	-72.518333	174.750278	25	12:00	-64.951944	177.650556
23	07:20	-72.368333	174.951389	25	13:00	-64.800000	177.569167
23	08:00	-72.285000	175.067778	25	14:00	-64.651667	177.519167
23	10:00	-72.051667	175.384444	25	15:00	-64.518056	177.467222
23	11:00	-71.917500	175.552500	25	16:00	-64.333333	177.385833
23	12:00	-71.769167	175.735556	25	17:00	-64.201389	177.350000
23	13:00	-71.735833	175.851944	25	18:00	-64.066667	177.285833
23	14:00	-71.567778	176.017778	25	22:00	-63.466944	177.035833
23	15:00	-71.434722	176.184444	26	00:00	-63.183889	176.918333
23	16:00	-71.318333	176.334167	26	01:00	-63.034722	176.868611
23	17:00	-71.200000	176.485556	26	02:00	-62.868333	176.816667
23	18:00	-71.068889	176.651111	26	05:00	-62.450278	176.618056
23	19:00	-70.852500	176.816667	26	06:00	-62.284722	176.566944
23	20:00	-70.985000	177.068889	26	07:00	-62.133889	176.502500
24	03:30	-70.800000	177.434722	26	08:00	-61.917500	176.433611
24	05:00	-69.550278	177.500556	26	09:00	-61.816667	176.400000
24	06:00	-69.385000	177.551389	26	10:00	-61.635833	176.335000
24	08:00	-69.101111	177.667222	26	10:30	-61.550000	176.301944
24	09:00	-68.951944	177.716944				

Table C.2: Model time-step (t), Heidelberger and Welch's stationarity test, Geweke Z-score, and the p-value for the Geweke Z-score for x_t and y_t of the 2D location (i.e. $\mathbf{x}_t = (x_t, y_t)$) for the model fit to the towed tag (**tag 186**). Note that $\mathbf{x}_0$ and $\mathbf{x}_T$ are not included here as these points are fixed.

Time-step	Heidelberger-Welch		Geweke Z-score		Geweke p-value	
	x	y	x	y	x	y
1	passed	passed	1.661	0.873	0.097	0.383
2	passed	passed	1.696	0.870	0.090	0.384
3	passed	passed	1.685	0.862	0.092	0.388
4	passed	passed	1.691	0.884	0.091	0.377
5	passed	passed	1.665	0.896	0.096	0.370
6	passed	passed	1.625	0.856	0.104	0.392
7	passed	passed	1.574	0.749	0.116	0.454
8	passed	passed	1.506	0.735	0.132	0.462
9	passed	passed	1.477	0.702	0.140	0.482
10	passed	passed	1.455	0.648	0.146	0.517
11	passed	passed	1.426	0.638	0.154	0.524
12	passed	passed	1.408	0.599	0.159	0.549
13	passed	passed	1.408	0.622	0.159	0.534
14	passed	passed	1.461	0.636	0.144	0.525
15	passed	passed	1.730	0.823	0.084	0.411
16	passed	passed	1.670	1.026	0.095	0.305
17	passed	passed	1.589	1.263	0.112	0.206
18	passed	passed	1.531	1.531	0.126	0.126
19	passed	passed	1.572	1.733	0.116	0.083
20	passed	passed	1.051	1.836	0.293	0.066
21	passed	passed	0.957	1.886	0.339	0.059
22	passed	passed	0.606	1.901	0.545	0.057
23	passed	passed	0.133	1.836	0.894	0.066
24	passed	passed	-0.147	2.042	0.883	0.041
25	passed	passed	-0.459	1.822	0.646	0.068
26	passed	passed	-0.156	1.370	0.876	0.171
27	passed	passed	0.350	0.691	0.727	0.489
28	passed	passed	1.831	-0.172	0.067	0.863
29	passed	passed	2.199	-0.894	0.028	0.371
30	passed	passed	1.983	-1.333	0.047	0.182
31	passed	passed	1.684	-1.606	0.092	0.108
32	passed	passed	1.543	-1.881	0.123	0.060
33	passed	passed	1.280	-1.891	0.201	0.059

Table C.2: Model time-step (t), Heidelberger and Welch's stationarity test, Geweke Z-score, and the p-value for the Geweke Z-score for x_t and y_t of the 2D location (i.e. $\mathbf{x}_t = (x_t, y_t)$) for the model fit to the towed tag (**tag 186**). Note that $\mathbf{x}_0$ and $\mathbf{x}_T$ are not included here as these points are fixed.

Time-step	Heidelberger-Welch		Geweke Z-score		Geweke p-value	
	x	y	x	y	x	y
34	passed	passed	1.272	-1.825	0.203	0.068
35	passed	passed	1.338	-1.840	0.181	0.066
36	passed	passed	1.264	-1.847	0.206	0.065
37	passed	passed	1.092	-1.891	0.275	0.059
38	passed	passed	1.039	-1.943	0.299	0.052
39	passed	passed	0.618	-1.941	0.537	0.052
40	passed	passed	0.057	-1.926	0.954	0.054
41	passed	passed	-0.203	-1.915	0.839	0.056
42	passed	passed	-0.253	-1.929	0.800	0.054
43	passed	passed	-0.137	-1.623	0.891	0.104
44	passed	passed	0.191	-1.634	0.848	0.102
45	passed	passed	0.692	-1.650	0.489	0.099
46	passed	passed	1.611	-1.675	0.107	0.094
47	passed	passed	1.686	-1.696	0.092	0.090
48	passed	passed	0.859	-1.707	0.391	0.088
49	passed	passed	0.290	-1.718	0.772	0.086
50	passed	passed	-0.129	-1.730	0.898	0.084
51	passed	passed	-0.401	-1.750	0.689	0.080
52	passed	passed	-0.503	-1.761	0.615	0.078
53	passed	passed	-0.528	-1.741	0.597	0.082
54	passed	passed	-0.712	-1.694	0.477	0.090
55	passed	passed	-0.731	-1.650	0.465	0.099
56	passed	passed	-0.823	-1.856	0.410	0.063
57	passed	passed	-0.923	-1.527	0.356	0.127
58	passed	passed	-1.080	-1.468	0.280	0.142
59	passed	passed	-1.306	-1.662	0.191	0.096
60	passed	passed	-1.604	-1.597	0.109	0.110
61	passed	passed	-1.923	-1.525	0.054	0.127
62	passed	passed	-2.253	-1.483	0.024	0.138
63	passed	passed	-2.428	-1.481	0.015	0.139
64	passed	passed	-2.570	-1.490	0.010	0.136
65	passed	passed	-2.361	-1.522	0.018	0.128
66	passed	passed	-2.510	-1.556	0.012	0.120

Table C.2: Model time-step (t), Heidelberger and Welch's stationarity test, Geweke Z-score, and the p-value for the Geweke Z-score for x_t and y_t of the 2D location (i.e. $\mathbf{x}_t = (x_t, y_t)$) for the model fit to the towed tag (**tag 186**). Note that $\mathbf{x}_0$ and $\mathbf{x}_T$ are not included here as these points are fixed.

Time-step	Heidelberger-Welch		Geweke Z-score		Geweke p-value	
	x	y	x	y	x	y
67	passed	passed	-2.722	-1.601	0.006	0.109
68	passed	passed	-1.966	-1.647	0.049	0.099
69	passed	passed	-1.440	-1.684	0.150	0.092
70	passed	failed	-1.271	-1.444	0.204	0.149
71	passed	failed	-1.404	-1.573	0.160	0.116
72	passed	failed	-1.275	-1.772	0.202	0.076
73	failed	passed	-1.650	-1.989	0.099	0.047
74	failed	passed	-1.788	-2.185	0.074	0.029
75	passed	failed	-1.649	-2.461	0.099	0.014
76	passed	failed	-1.877	-2.238	0.061	0.025
77	failed	passed	-1.952	-2.096	0.051	0.036

Table C.3: Model time-step (t), Heidelberger and Welch's stationarity test, Geweke Z-score, and the p-value for the Geweke Z-score for x_t and y_t of the 2D location (i.e. $\mathbf{x}_t = (x_t, y_t)$) for the **simulated tag**. Note that $\mathbf{x}_0$ and $\mathbf{x}_T$ are not included here as these points are fixed.

Time-step	Heidelberger-Welch		Geweke Z-score		Geweke p-value	
	x	y	x	y	x	y
1	passed	passed	-0.415	0.048	0.678	0.961
2	passed	passed	-0.477	0.103	0.634	0.918
3	passed	passed	-0.529	0.099	0.597	0.921
4	passed	passed	-0.623	0.104	0.533	0.917
5	passed	passed	-0.778	0.115	0.437	0.908
6	passed	passed	-0.815	0.118	0.415	0.906
7	passed	passed	-0.879	0.117	0.379	0.907
8	passed	passed	-1.249	0.116	0.212	0.908
9	passed	passed	-1.371	0.111	0.170	0.912
10	passed	passed	-1.239	0.119	0.215	0.905
11	passed	passed	-0.635	0.139	0.526	0.889

Table C.3: Model time-step (t), Heidelberger and Welch's stationarity test, Geweke Z-score, and the p-value for the Geweke Z-score for x_t and y_t of the 2D location (i.e. $\mathbf{x}_t = (x_t, y_t)$) for the **simulated tag**. Note that $\mathbf{x}_0$ and $\mathbf{x}_T$ are not included here as these points are fixed.

Time-step	Heidelberger-Welch		Geweke Z-score		Geweke p-value	
	x	y	x	y	x	y
12	passed	passed	0.194	0.158	0.846	0.875
13	passed	passed	0.649	0.155	0.516	0.877
14	passed	passed	0.735	0.143	0.462	0.887
15	passed	passed	0.932	0.164	0.351	0.870
16	passed	passed	1.058	0.213	0.290	0.832
17	passed	passed	0.902	0.291	0.367	0.771
18	passed	passed	0.613	0.435	0.540	0.663
19	passed	passed	0.372	0.734	0.710	0.463
20	passed	passed	0.203	1.236	0.839	0.217
21	passed	passed	0.055	1.939	0.956	0.053
22	passed	passed	-0.049	2.310	0.961	0.021
23	passed	passed	-0.128	2.190	0.898	0.029
24	passed	passed	-0.231	2.062	0.817	0.039
25	passed	passed	-0.316	1.997	0.752	0.046
26	passed	passed	-0.373	1.774	0.709	0.076
27	passed	passed	-0.391	1.622	0.696	0.105
28	passed	passed	-0.413	1.438	0.680	0.150
29	passed	passed	-0.384	1.240	0.701	0.215
30	passed	passed	-0.379	1.068	0.705	0.285
31	passed	passed	-0.370	0.798	0.711	0.425
32	passed	passed	-0.382	0.666	0.702	0.505
33	passed	passed	-0.378	0.586	0.706	0.558
34	passed	passed	-0.366	0.573	0.714	0.567
35	passed	passed	-0.364	0.473	0.716	0.636
36	passed	passed	-0.350	0.389	0.726	0.697
37	passed	passed	-0.330	0.273	0.741	0.785
38	passed	passed	-0.302	0.165	0.763	0.869
39	passed	passed	-0.316	0.095	0.752	0.925
40	passed	passed	-0.401	0.061	0.689	0.951
41	passed	passed	-0.392	-0.002	0.695	0.999
42	passed	passed	-0.433	-0.075	0.665	0.940
43	passed	passed	-0.576	-0.241	0.565	0.810
44	passed	passed	-0.497	-0.402	0.619	0.688

Table C.3: Model time-step (t), Heidelberger and Welch's stationarity test, Geweke Z-score, and the p-value for the Geweke Z-score for x_t and y_t of the 2D location (i.e. $\mathbf{x}_t = (x_t, y_t)$) for the **simulated tag**. Note that $\mathbf{x}_0$ and $\mathbf{x}_T$ are not included here as these points are fixed.

Time-step	Heidelberger-Welch		Geweke Z-score		Geweke p-value	
	x	y	x	y	x	y
45	passed	passed	-0.276	-0.362	0.783	0.717
46	passed	passed	0.210	-0.402	0.834	0.688
47	passed	passed	0.367	-0.255	0.713	0.799
48	passed	passed	0.257	-0.119	0.797	0.905
49	passed	passed	0.293	0.074	0.770	0.941
50	passed	passed	0.816	0.185	0.415	0.853
51	passed	passed	0.417	0.237	0.677	0.813
52	passed	passed	0.291	0.061	0.771	0.951
53	passed	passed	0.337	-0.407	0.736	0.684
54	passed	passed	0.241	-0.886	0.810	0.376
55	passed	passed	-0.381	-1.537	0.703	0.124
56	passed	passed	-0.571	-2.031	0.568	0.042
57	passed	passed	0.150	-2.319	0.881	0.020
58	passed	passed	1.160	-2.136	0.246	0.033
59	passed	passed	0.769	-1.926	0.442	0.054
60	passed	passed	0.811	-1.091	0.417	0.275
61	passed	passed	1.105	-0.529	0.269	0.597
62	passed	passed	0.551	0.005	0.581	0.996
63	passed	passed	0.046	0.143	0.963	0.887
64	passed	passed	-0.022	0.327	0.982	0.743
65	passed	passed	0.506	0.115	0.613	0.909
66	passed	passed	0.721	-0.137	0.471	0.891
67	passed	passed	0.263	-0.283	0.793	0.777
68	passed	passed	0.425	-0.436	0.671	0.663
69	passed	passed	0.008	-0.501	0.994	0.616
70	passed	passed	-0.080	-0.612	0.936	0.540
71	passed	passed	-0.546	-0.769	0.585	0.442
72	passed	passed	-0.421	-0.962	0.674	0.336
73	passed	passed	-0.483	-1.180	0.629	0.238
74	passed	passed	-0.318	-1.293	0.750	0.196
75	passed	passed	-0.041	-1.372	0.967	0.170
76	passed	passed	-0.292	-1.545	0.770	0.122
77	passed	passed	-0.160	-1.289	0.873	0.197

Table C.4: Model time-step (t), Heidelberger and Welch's stationarity test, Geweke Z-score, and the p-value for the Geweke Z-score for x_t and y_t of the 2D location (i.e. $\mathbf{x}_t = (x_t, y_t)$) for the model fit to the towed tag (**tag 121**). Note that $\mathbf{x}_0$ and $\mathbf{x}_T$ are not included here as these points are fixed.

Time-step	Heidelberger-Welch		Geweke Z-score		Geweke p-value	
	x	y	x	y	x	y
1	passed	passed	0.707	-0.392	0.480	0.695
2	passed	passed	-1.942	2.452	0.052	0.014
3	passed	passed	-9.765	7.165	0.000	0.000
4	passed	passed	-1.016	1.167	0.309	0.243
5	passed	passed	2.416	-0.731	0.016	0.464
6	passed	passed	1.713	-1.257	0.087	0.209
7	passed	passed	1.756	-1.577	0.079	0.115
8	passed	passed	1.418	-1.288	0.156	0.198
9	passed	passed	0.178	-0.750	0.858	0.453
10	passed	passed	0.073	0.598	0.942	0.550
11	passed	passed	0.293	-1.300	0.770	0.194
12	passed	passed	-0.092	1.425	0.927	0.154
13	passed	passed	-0.360	2.289	0.719	0.022
14	passed	passed	-1.570	1.301	0.116	0.193
15	passed	passed	0.184	-1.334	0.854	0.182
16	passed	passed	2.439	-2.276	0.015	0.023
17	passed	passed	0.545	-0.753	0.586	0.452
18	passed	passed	-0.674	0.088	0.500	0.930
19	passed	passed	-0.733	1.754	0.463	0.079
20	passed	passed	0.713	0.121	0.476	0.904
21	passed	passed	-0.367	-0.580	0.714	0.562
22	passed	passed	1.475	-0.855	0.140	0.393
23	passed	passed	0.613	-0.451	0.540	0.652
24	passed	passed	0.257	0.329	0.797	0.742
25	passed	passed	-0.397	2.369	0.691	0.018
26	passed	passed	0.218	1.325	0.827	0.185
27	passed	passed	0.126	0.967	0.900	0.333
28	passed	passed	-1.787	-1.889	0.074	0.059
29	passed	passed	0.208	-0.818	0.835	0.414
30	passed	passed	1.121	-1.219	0.262	0.223
31	passed	passed	0.047	0.159	0.962	0.874
32	passed	passed	-0.341	1.092	0.733	0.275

Table C.4: Model time-step (t), Heidelberger and Welch's stationarity test, Geweke Z-score, and the p-value for the Geweke Z-score for x_t and y_t of the 2D location (i.e. $\mathbf{x}_t = (x_t, y_t)$) for the model fit to the towed tag (**tag 121**). Note that $\mathbf{x}_0$ and $\mathbf{x}_T$ are not included here as these points are fixed.

Time-step	Heidelberger-Welch		Geweke Z-score		Geweke p-value	
	x	y	x	y	x	y
33	passed	passed	1.386	-0.326	0.166	0.744
34	passed	passed	-0.088	0.663	0.930	0.507
35	passed	passed	0.659	-0.071	0.510	0.944
36	passed	passed	-2.000	3.102	0.045	0.002
37	passed	passed	1.565	1.198	0.118	0.231
38	passed	passed	0.244	0.766	0.808	0.443
39	passed	passed	-1.059	1.056	0.290	0.291
40	passed	passed	0.748	-0.678	0.454	0.498
41	passed	passed	-1.624	2.367	0.104	0.018
42	passed	passed	-0.765	1.468	0.444	0.142
43	passed	passed	0.053	-0.185	0.958	0.853
44	passed	passed	-0.335	0.414	0.737	0.679
45	passed	passed	-0.366	0.181	0.714	0.856
46	passed	passed	-0.860	0.453	0.390	0.650
47	passed	passed	-0.300	-0.685	0.764	0.493

Appendix D

Agent-based model of snapper (SNA 1)

This appendix provides input and output from the snapper (SNA 1) agent-based model (ABM) that was used in the development of Bayesian emulation (described in Chapter 7, page 298). The actual model input file for the agent-based simulation of snapper (SNA 1) is given, followed by series of plots of the model output.

D.1 ABM input file

```
// ================================================================================
// Species: SNA 1
// ================================================================================

#ifndef PARAMETERS_H
#define PARAMETERS_H

#define DEBUG_OFF
#define THREADING_ON
#define HOME_ON

// ================================================================================
// GLOBAL VARIABLES
// ================================================================================
const int myseed        = 7;
const int agent_size    = 100;
const int merge_thresh  = 10;
```

```
const bool merge_by_age = 1; // If =1 then merge by age, else merge by cell.
const int min_age       = 1;
const int max_age       = 50;

const int n_stocks  = 3;
const int n_areas   = 3;
const int n_sex     = 2;
const int n_fishery = 1;
const int n_fyears  = 114;
const int n_steps   = 2;

// ============================================================================
// INITIALISATION
// ============================================================================
const int phase1 = 100;
const int phase2 = 100;

// ============================================================================
// STOCHASTICITY SWITCHES (0 = off, 1 = on)
// ============================================================================
const bool stochastic_sex    = 1;
const bool stochastic_rec    = 1;
const bool stochastic_mat    = 1;
const bool stochastic_growth = 1;
const bool stochastic_mort   = 1;

// ============================================================================
// RECRUITMENT
//
============================================================================
//                                    EN,      HG,      BP
const int R0[n_stocks]              = {443493, 950050, 318619};
const double steepness[n_stocks]    = {0.85,   0.85,   0.85};
const double p_male[n_stocks]       = {0.5,    0.5,    0.5};
const int y_enter[n_stocks]         = {1,      1,      1};
const double rec_sigma[n_stocks]    = {0.1408, 0.1408, 0.1408};
const double rec_autocorr[n_stocks] = {0.6,    0.6,    0.6};

// ============================================================================
// MATURITY
// ============================================================================
//                                    F,     M
const double mat_a50[n_sex]         = {4,     4};
const double mat_ato95[n_sex]       = {4.7,   4.7};
const int mat_L[n_sex]              = {3,     3};
const int mat_R[n_sex]              = {5,     5};
const double mat_a50_sigma[n_sex]   = {0.001, 0.001};
const double mat_ato95_sigma[n_sex] = {0.001, 0.001};

// ============================================================================
// SIZE
// ============================================================================
const double size_linf[n_sex]={58.8, 58.8};
```

```
const double size_linf_cv[n_sex]={0.01,0.01};
const double size_k[n_sex]={0.102, 0.102};
const double size_k_sigma[n_sex]={0.001,0.001};
const double size_t0[n_sex]={-1.11, -1.11};
const double size_cv[n_sex]={0.102, 0.102};
const double size_sigma_min[n_sex]={0.001, 0.001};

// ================================================================================
// LENGTH-WEIGHT
// ================================================================================
const double size_alpha[n_sex]={4.467e-08, 4.467e-08};
const double size_alpha_sigma[n_sex]={1e-10, 1e-10};
const double size_beta[n_sex]={2.793, 2.793};
const double size_beta_sigma[n_sex]={0.001, 0.001};

// ================================================================================
// NATURAL MORTALITY
// ================================================================================
const double mort_M[n_sex]={0.075, 0.075};
const double mort_sigma[n_sex]={0.001, 0.001};
const double mort_dPar = 1; // introduces -ve bias

// ================================================================================
// FISHERY
// ================================================================================
const int first_yr[n_fishery]={1900};
const int last_yr[n_fishery]={2013};
const double fish_removals[n_areas][n_fyears]={
  {41.6, 53.4, 53.7, 53.9, 54.2, 54.5, 54.8, 55, 55.3, 55.6, 55.9, 56.1, 56.4,
   56.7, 57, 57.3, 66.3, 102.4, 93.1, 76.8, 87.4, 94.2, 99.7, 116.9, 132.1,
   149.5, 117.9, 142.6, 101.3, 120.2, 127, 92.6, 94.3, 105.8, 115.1, 140,
   162.7, 146.8, 158.2, 153.2, 135, 128.2, 112.5, 120.9, 131.6, 129.1, 137.7,
   147.4, 168.4, 150, 134.2, 116.8, 105.6, 103.8, 117.8, 121.2, 128.1, 140.2,
   137.6, 151.7, 167.9, 177.8, 201.5, 222, 238.6, 265, 292.4, 313.3, 347.7,
   361.9, 372.1, 390.5, 360.4, 342.7, 300.6, 246.1, 277, 278.1, 355.9, 380.6,
   288.5, 314.1, 300.5, 278, 332, 298.2, 232.1, 129.9, 154.5, 179.9, 225,
   149.3, 163.6, 171.8, 183.6, 175.4, 208.8, 203.3, 176.1, 174.3, 169.5, 177.1,
   167.3, 162.9, 167.5, 175.6, 212, 200.1, 186, 186.9, 187.2, 177.1, 174.9,
   174.9},
  {115.8, 150.2, 150.9, 151.7, 152.4, 153.2, 154, 154.7, 155.5, 156.3, 157,
    157.8, 158.5, 159.3, 160.1, 160.8, 187.4, 293.5, 266.5, 218.3, 249.2, 270,
    285.9, 336.1, 380.8, 432.5, 339.1, 411.5, 289.9, 345.6, 365.8, 264.1, 269,
    302.8, 331, 403.8, 470.5, 423.8, 456.9, 442.4, 388.3, 368.2, 322.3, 347,
    378.3, 371, 396, 424.7, 486.2, 432.3, 385.2, 334.2, 300.9, 296, 337, 347.4,
    367.7, 403.1, 395.8, 436.6, 436.4, 416.9, 438.9, 450.9, 451.7, 480.9,
    513.7, 526.6, 580.3, 665.6, 738.9, 789.7, 708, 665.3, 569.1, 455.3, 530.2,
    538.7, 688.9, 704.2, 518.7, 540.5, 500, 448.6, 449.9, 459.3, 382.6, 350.8,
    454.7, 494.7, 425.4, 445.9, 529.1, 458.1, 433.3, 395.1, 364.1, 359.5,
    387.6, 354.3, 344.8, 359.3, 387.5, 407.3, 331.5, 315.9, 343, 352.1, 429.2,
    444.3, 416.4, 437.5, 473.4, 473.4},
  {52.5, 67.7, 67.9, 68.2, 68.4, 68.6, 68.8, 69, 69.3, 69.5, 69.7, 69.9, 70.1,
    70.4, 70.6, 70.8, 82.6, 129.8, 117.7, 95.9, 109.8, 118.8, 125.6, 148.1,
    167.9, 190.9, 149, 181.3, 126.8, 151.6, 160.6, 114.9, 117.1, 132.2, 144.5,
```

```
    176.8, 206.5, 185.5, 200.3, 193.6, 169.3, 160.3, 139.7, 150.6, 164.6, 161,
    172, 184.4, 212.2, 187.8, 166.9, 143.7, 128.8, 126.6, 144.7, 149.2, 158.4,
    173.8, 170.7, 188.7, 200.9, 204.5, 226.7, 244.6, 257.3, 282.6, 309.3,
    327.7, 363.9, 390.8, 411.9, 421.6, 372.1, 340.3, 285.4, 223.3, 243.2,
    233.7, 284.7, 288.9, 209.4, 216.9, 198.5, 175.6, 156.5, 226.8, 190.2,
    122.4, 111.4, 128.8, 159.1, 138.1, 168.6, 153.1, 141.1, 160.2, 195, 205.2,
    153.3, 183.8, 223.3, 173.6, 187.8, 234.4, 228.7, 261.1, 245.5, 191.7,
    203.5, 185.2, 214.7, 235.5, 229.6, 229.6}
};
const double fish_q=1.08658658658659;
const double fish_q_sigma=0.001;
const double fish_max_exploitation=0.9;

// ================================================================================
// SELECTIVITY
// ================================================================================
const int sel_attribute=0; // 0=age, 1=length, 2=weight
const double sel_a1[n_sex] = {6.04604604604605, 6.04604604604605};
const double sel_sL[n_sex] = {2.31481481481481, 2.31481481481481};
const double sel_sR[n_sex] = {100., 100.};

// ================================================================================
// LAYERS
// ================================================================================
const char lay_names[n_areas][2] = {
  {'E','N'},
  {'H','G'},
  {'B','P'}
};
const double lay_recruitment[n_stocks][n_areas] = {
  {1, 0, 0},
  {0, 1, 0},
  {0, 0, 1}
};
const int lay_stock[n_areas] = {0, 1, 2};

// ================================================================================
// MIGRATION
// ================================================================================
const double mig_matrix[n_areas][n_areas] = {
  {0.7696967, 0.0536603, 0.176643},
  {0.0870992, 0.5140839, 0.398817},
  {0.2408820, 0.2760040, 0.483114}
};
const bool home_rec = 1; // home_rec = 1 if home is the cell that fish recruit to

// ================================================================================
// TAGGING
// ================================================================================
const double tag_a1[n_fishery][n_sex] = {
  {100., 100.}
};
const double tag_sL[n_fishery][n_sex] = {
```

```
  {21., 21.}
};
const double tag_sR[n_fishery][n_sex] = {
  {1000., 1000.}
};
const int tag_fish[n_areas][n_fyears] = {
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 6782, 0, 0, 0, 0, 0, 0, 0, 0, 8190, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 12046,0, 0, 0, 0, 0, 0, 0, 0, 13466,0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3630, 0, 0, 0, 0, 0,
   0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
};

#endif /*PARAMETERS_H*/
```

D.2 Plots of ABM output

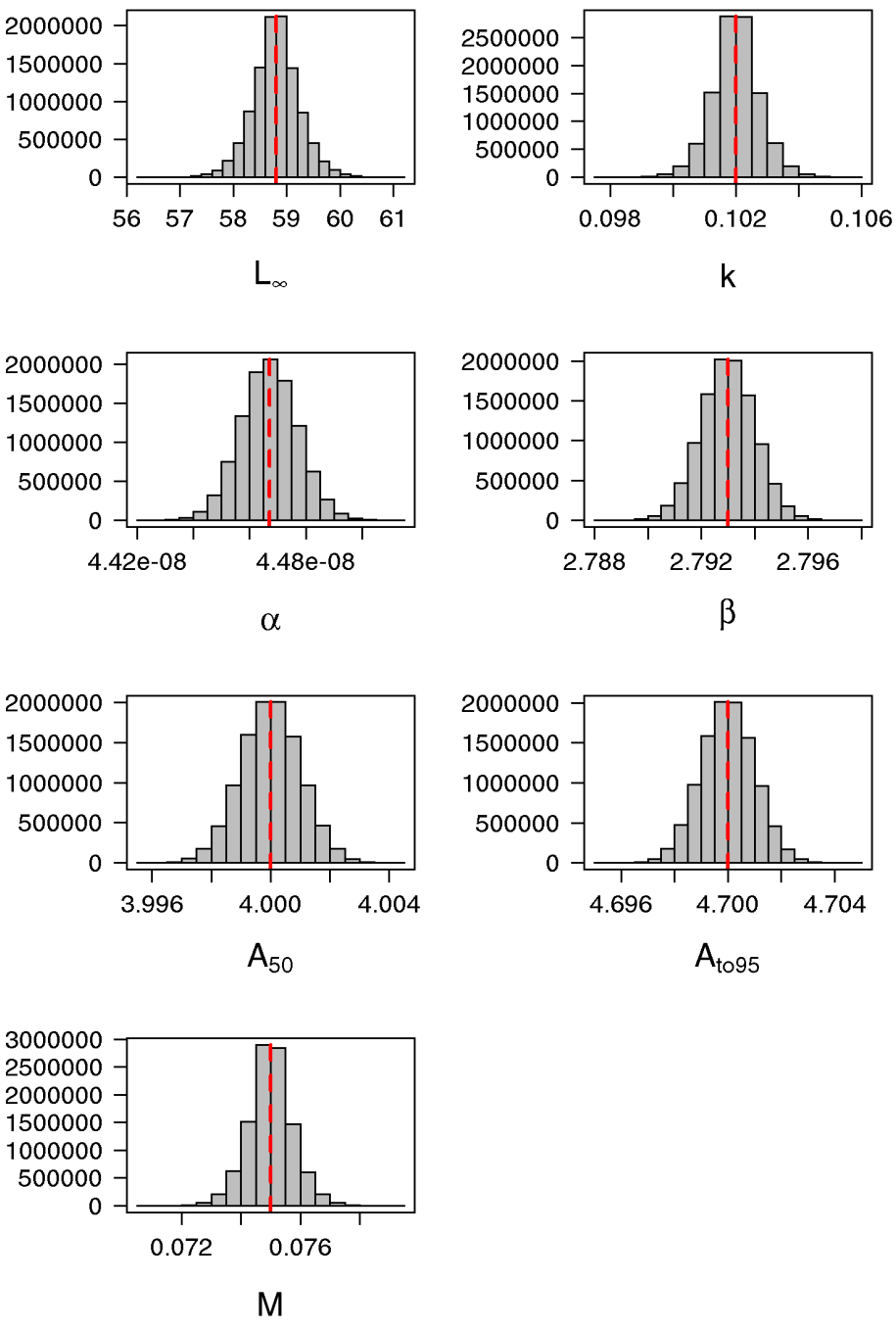


Figure D.1: Population parameters in the initialised population. The dashed red line is the specified median.

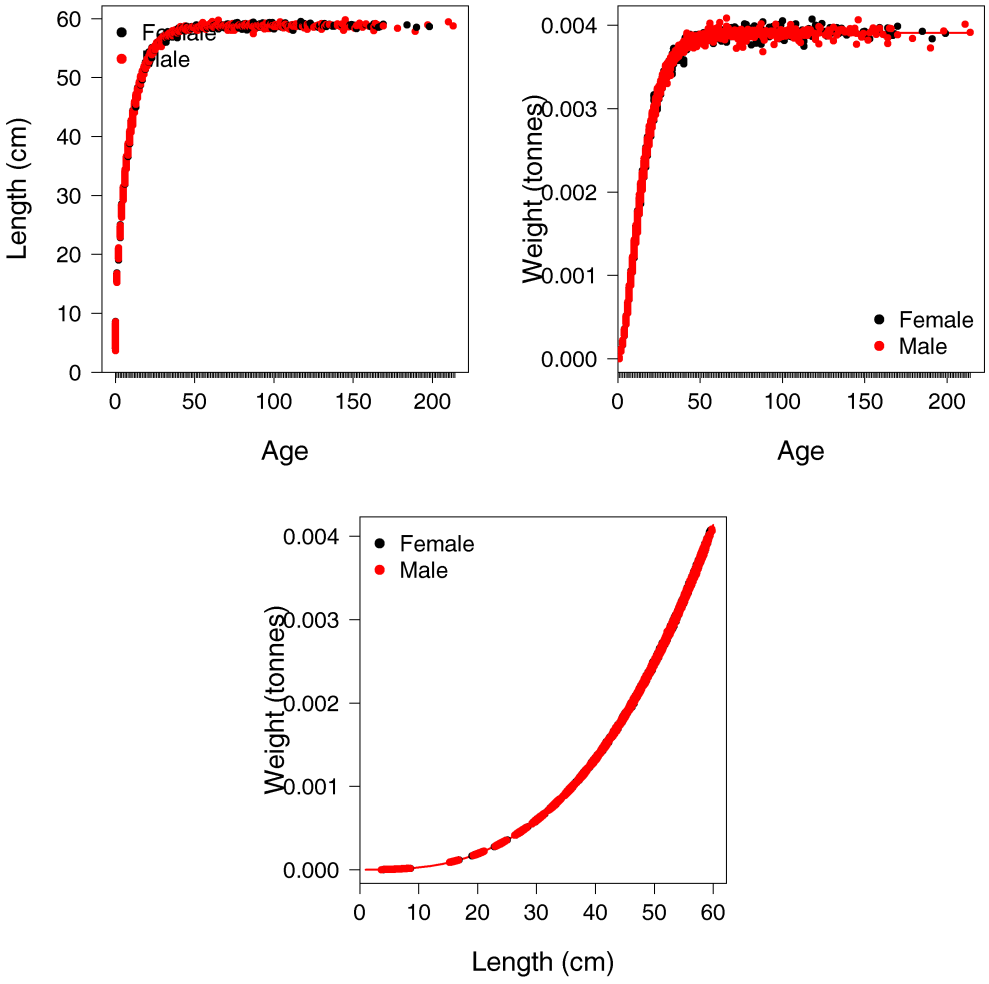


Figure D.2: Age-length [top left], age-weight [top right] and length-weight [bottom] of agents in the initialised population. These plots are split by female and male.

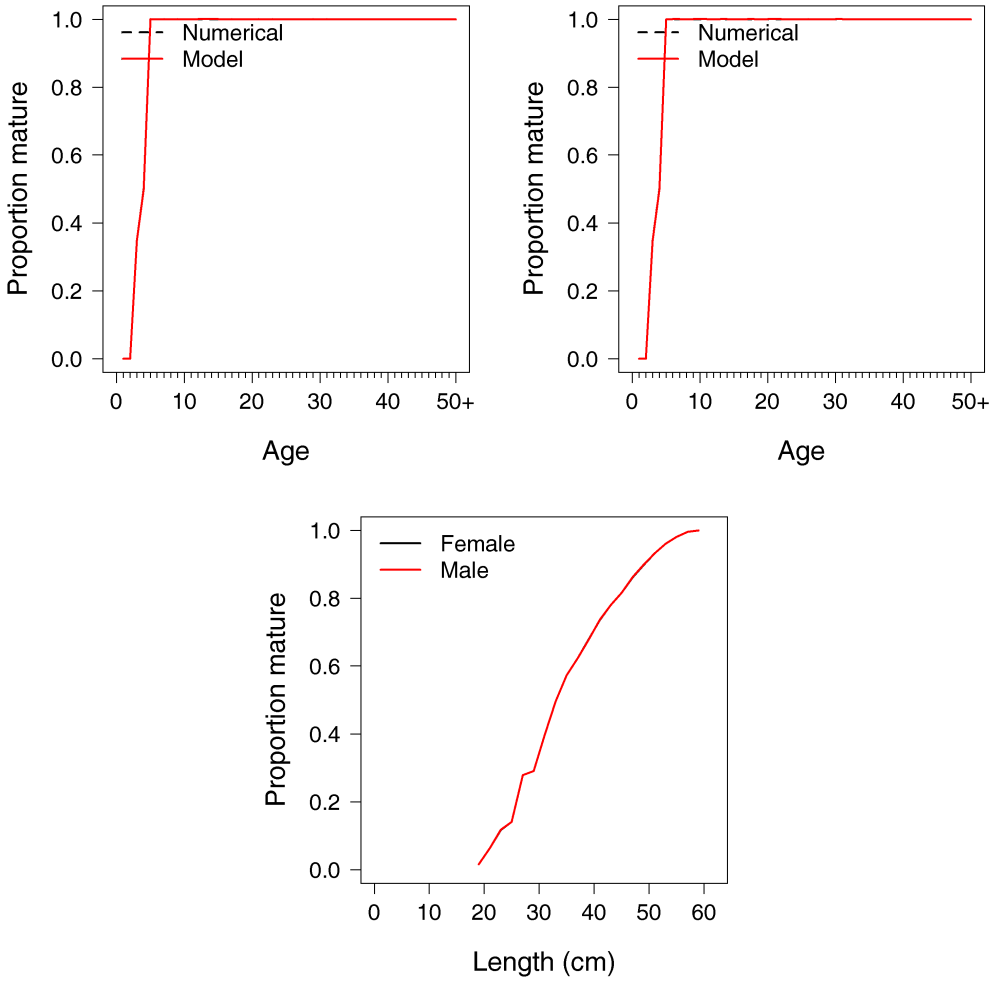


Figure D.3: Proportion mature at age for females [top left] and males [right] and the proportion mature by length [bottom] in the initialised population.

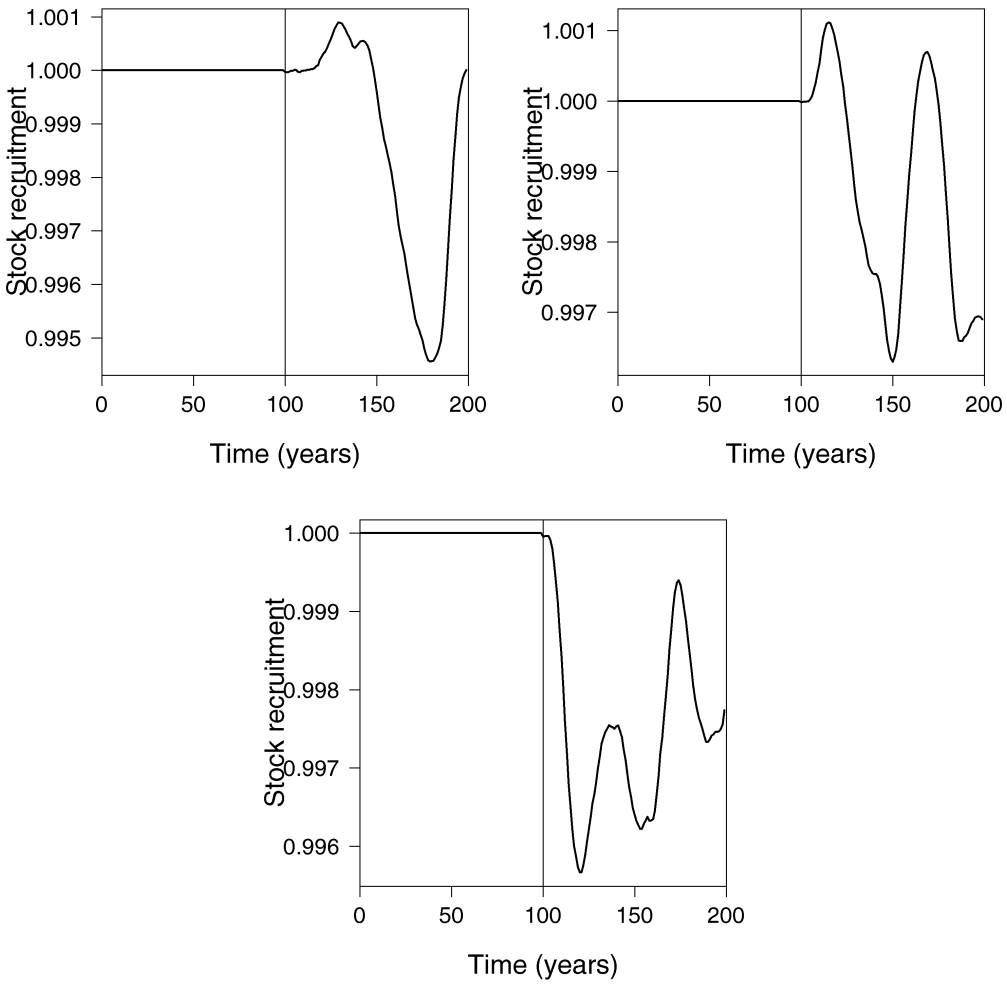


Figure D.4: Stock recruitment $SR(SSB_t)$ during both phases of initialisation in East-Northland [top left], the Hauraki Gulf [top right] and the Bay of Plenty [bottom].

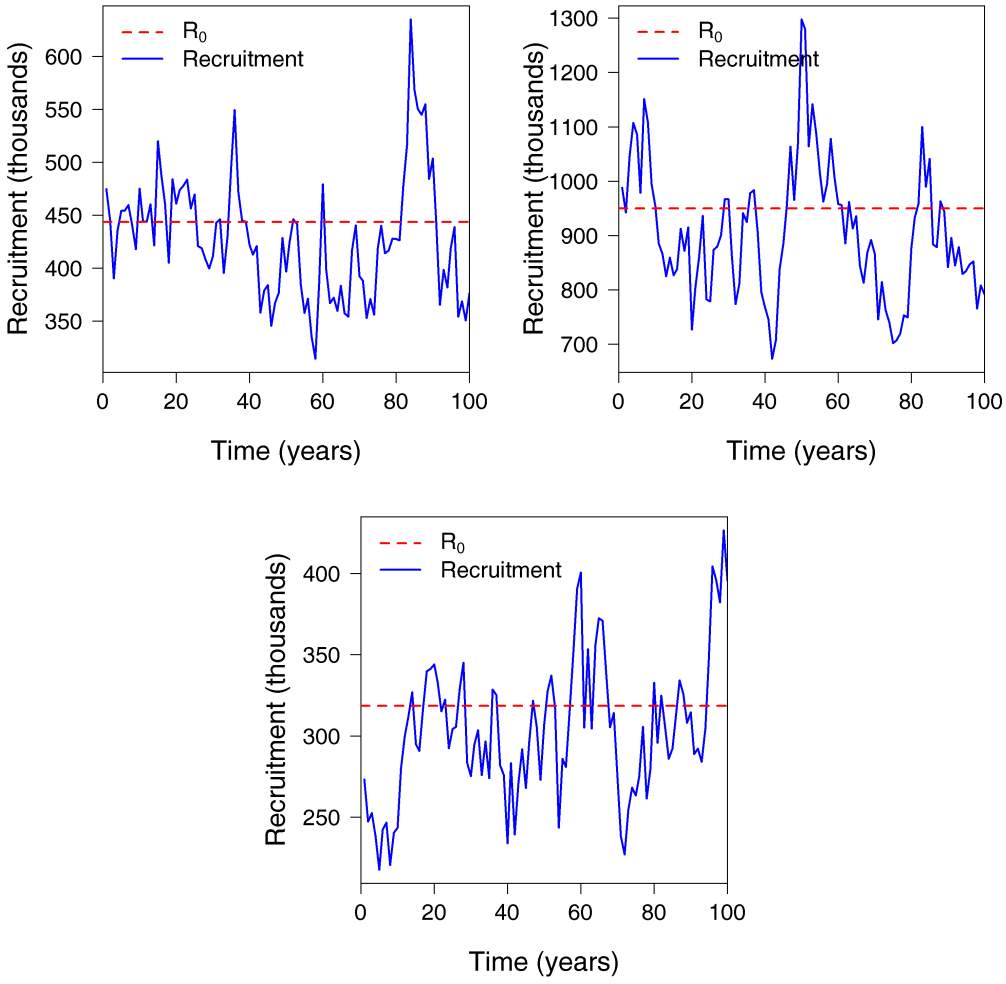


Figure D.5: Recruitment R_t during the second phase of initialisation in East-Northland [top left], the Hauraki Gulf [top right] and the Bay of Plenty [bottom].

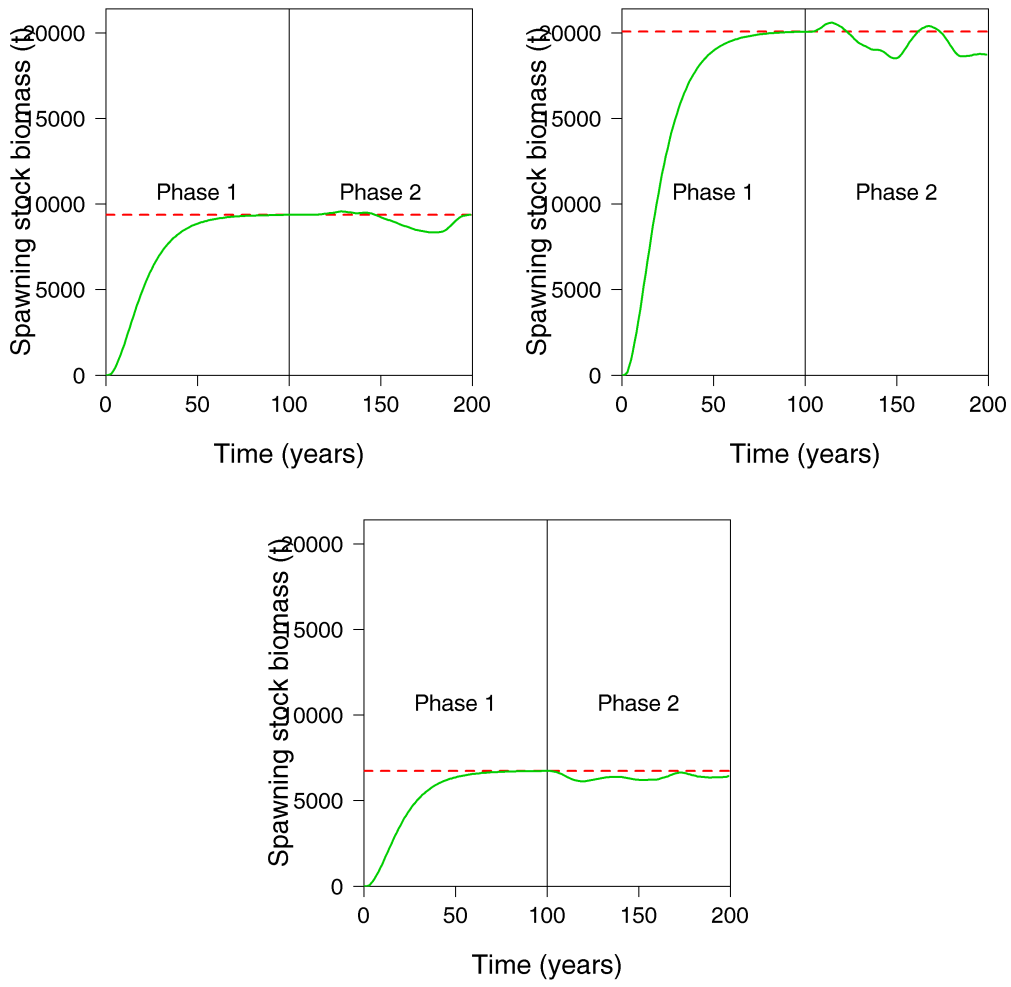


Figure D.6: Spawning stock biomass SSB_t during both phases of initialisation in East-Northland [top left], the Hauraki Gulf [top right] and the Bay of Plenty [bottom].

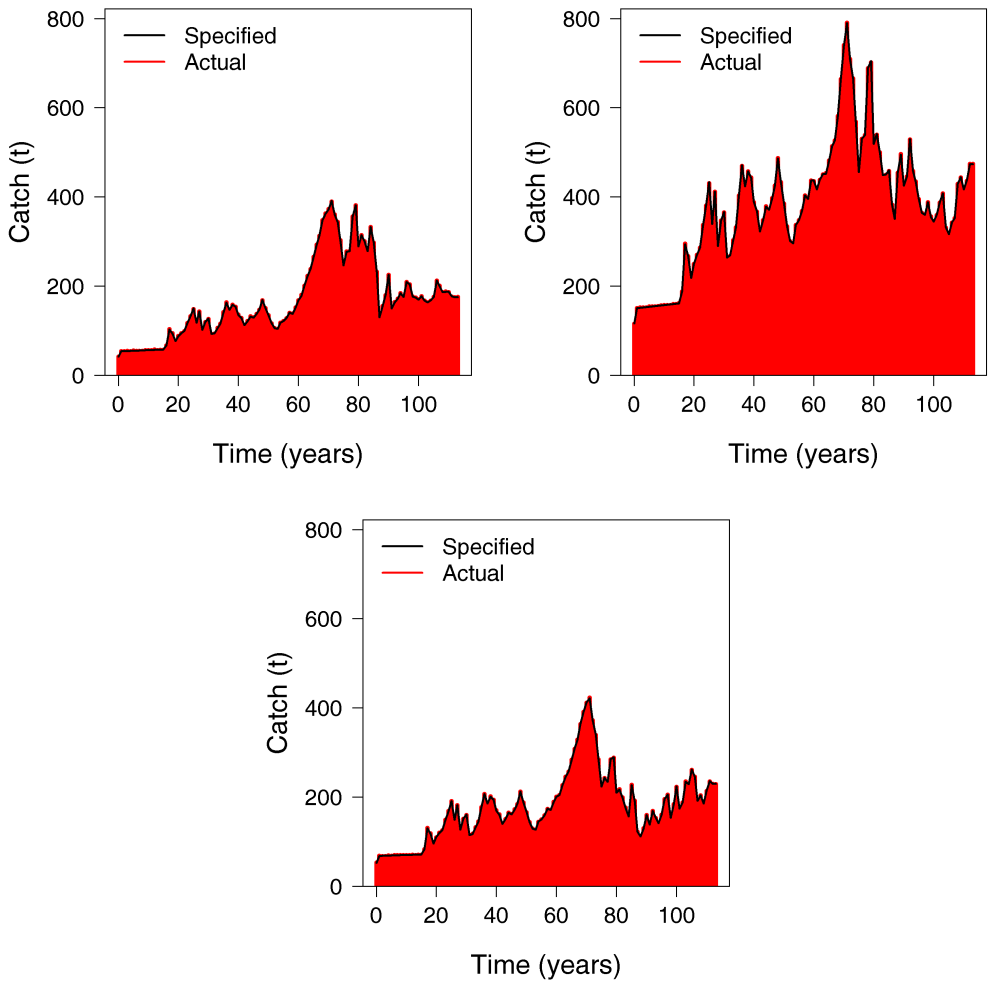


Figure D.7: Catch (tonnes) that was specified and catch in the model in East-Northland [top left], the Hauraki Gulf [top right] and the Bay of Plenty [bottom].

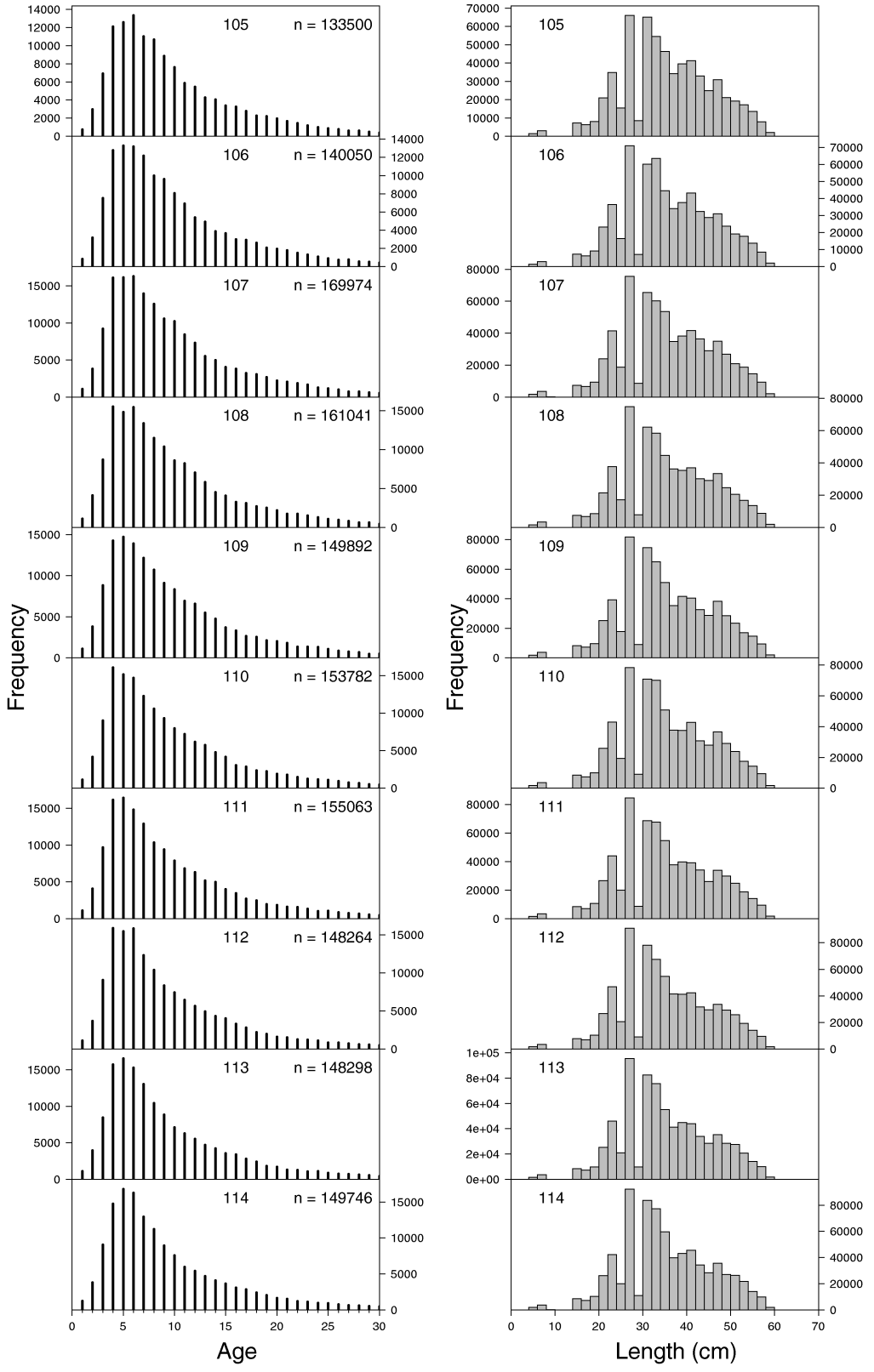


Figure D.8: The age-frequencies [left column] and the length-frequencies [right column] in the population during the final ten years of the fishery.

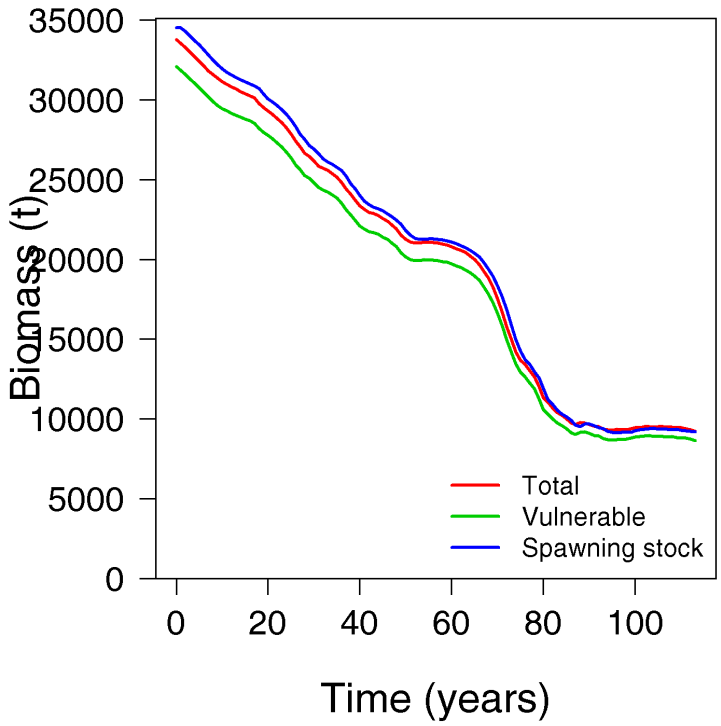


Figure D.9: Biomass (tonnes) in all areas during the fishery including the total biomass, the vulnerable biomass and the spawning stock biomass.

References

Anderson, L. (1977), *The Economics of Fisheries Management*, The Blackburn Press.

Armstrong, J. & Schindler, D. (2011), 'Excess digestive capacity in predators reflects a life of feast and famine', *Nature* **476**, 84–87.

Arreguin-Sanchez, F. (1996), 'Catchability: a key parameter for fish stock assessment', *Reviews in Fish Biology and Fisheries* **6**, 221–242.

Beverton, R. & Holt, S. (1957), *On the Dynamics of Exploited Fish Populations*, Vol. 19 of *2*, Ministry of Agriculture, Fisheries and Food, London.

Biro, P. & Post, J. (2008), 'Rapid depletion of genotypes with fast growth and bold personality traits from harvested fish populations', *PNAS* pp. 2919–2922.

Booth, J. D. (1984), 'Size at onset of breeding in female *Jasus verreauxi* (Decapoda: Palinuridae) in New Zealand', *New Zealand Journal of Marine and Freshwater Research* **18**, 159–169.

Booth, J. D. (2011), *Spiny Lobsters: Through the Eyes of the Giant Packhorse*, Victoria University Press.

Branch, T. (2009*a*), 'Corrigendum: Differences in predicted catch composition between two widely used catch equation formulations', *Canadian Journal of Fisheries and Aquatic Science* **66**, 1631.

Branch, T. (2009*b*), 'Differences in predicted catch composition between two widely used catch equation formulations', *Canadian Journal of Fisheries and Aquatic Science* **66**, 126–132.

Branch, T. (2010), 'Reply to the comment by Francis on "Differences in

predicted catch composition between two widely used catch equation formulations"', *Canadian Journal of Fisheries and Aquatic Science* **67**, 766–768.

Brasher, D., Ovenden, J., Booth, J. D. & White, R. (1992), 'Genetic subdivision of Australian and New Zealand populations of *Jasus verreauxi* (Decapoda: Palinuridae) - preliminary evidence from the mitochondrial genome', *New Zealand Journal of Marine and Freshwater Research* **26**(1), 53–58.

Brown, J., Brickle, P. & Scott, B. (2011), 'Investigating the movements and behaviour of Patagonian toothfish (*Dissostichus eleginoides* Smitt, 1898) around the Falkland Islands using satellite linked archival tags', *Journal of Experimental Marine Biology and Ecology* **443**, 65–74.

Bull, B., Francis, R., Dunn, A., McKenzie, A., Gilbert, D., Smith, M., Bian, R. & Fu, D. (2012), 'CASAL (C++ algorithmic stock assessment laboratory): CASAL User Manual v2.30-2012/03/21', *NIWA Technical Report* **135**, 280.

Butterworth, D., Ianelli, J. & Hilborn, R. (2003), 'A statistical model for stock assessment of southern bluefin tuna with temporal changes in selectivity', *African Journal of Marine Science* **25**(1), 331–361.

Cadrin, S. X. & Secor, D. (2009), 'Accounting for spatial population structure in stock assessment: past, present, and future', *Fish and Fisheries* **31**, 405–426.

Davey, F. (2004), Ross Sea Bathymetry, 1:2 000 000, version 1.0, Institute of Geological and Nuclear Sciences geophysical map 16, Technical report, Institute of Geological and Nuclear Sciences Limited.

Diggle, P., Heagerty, P., Liang, K. & Zeger, S. (2002), *Analysis of Longitudinal Data*, 2nd edn, Oxford University Press.

Dunn, A. & Hanchet, S. (2011), 'Southern blue whiting (*Micromesistius australis*) stock assessment for the Campbell Island Rise for 2009-10', *New Zealand Fisheries Assessment Report* **2011/40**, 37.

Dunn, A. & Rasmussen, S. (2009), *Spatial Population Model User Manual*

(SPM v0.1-2009-02-04-23:08:28 UTC (rev. 2987)), National Institute of Water & Atmospheric Research Ltd.

Dunn, A., Rasmussen, S. & Hanchet, S. (2009), 'Development of spatially explicit age-structured population dynamics operating models for Antarctic toothfish in the Ross Sea', *WG-SAM* **09/18**, 38.

Fox, W. (1970), 'An exponential surplus-yield model for optimizing exploited fish populations', *Transactions of the American Fisheries Society* **99**, 80–88.

Francis, R. (1999), 'The impacts of correlations in standardised CPUE indices', *New Zealand Fisheries Assessment Research Document* **1999/42**, 30.

Francis, R. (2006), 'Assessment of hoki (*Macruronus novaezelandiae*) in 2005', *New Zealand Fisheries Assessment Report* **2006/3**, 96.

Francis, R. (2010), 'Comment on "Differences in predicted catch composition between two widely used catch equation formulations"', *Canadian Journal of Fisheries and Aquatic Science* **67**, 763–765.

Francis, R. (2011), 'Data weighting in statistical fisheries stock assessment models', *Canadian Journal of Fisheries and Aquatic Science* **68**, 1124–1138.

Francis, R. (2014), 'Replacing the multinomial in stock assessment models: a first step', *Fisheries Research* **151**, 70–84.

Francis, R. & McKenzie, A. (2013), 'Assessment of the SNA 1 stocks in 2013', *New Zealand Fisheries Assessment Report* .

Gelman, A. (2006), 'Prior distributions for variance parameters in hierarchical models', *Bayesian Analysis* **1**(3), 515–533.

Geman, S. & Geman, D. (1984), 'Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**, 721–741.

Gilks, W. & Roberts, C. (1996), *Strategies for improving MCMC*, Chapman & Hall, pp. 89–114.

Godo, O. & Michalsen, K. (2000), 'Migratory behaviour of north-east Arctic cod, studied by use of data storage tags', *Fisheries Research* **48**, 127–140.

Goldstein, M. & Rougier, J. (2006), 'Bayes linear calibrated prediction for complex systems', *Journal of the American Statistical Association* **101**(475), 1132–1143.

Gray, R., Fulton, B., Little, R. & Scott, R. (2006), Ecosystem model specification with an agent based framework, Technical report, CSIRO.

Green, P. (1995), 'Reversible jump Markov chain Monte Carlo computation and Bayesian model determination', *Biometrika* **82**(4), 711–732.

Grimm, V. & Railsback, S. (2005), *Individual-Based Modeling and Ecology*, Princeton University Press.

Haist, V., Breen, P. A., Starr, P. J. & Kendrick, T. H. (2011), 'The 2010 stock assessment of red rock lobsters (*Jasus edwardsii*) in CRA 5, and development of an operational management procedure', *New Zealand Fisheries Assessment Report* **2011/12**, 68.

Hanchet, S., Rickard, G., Fenaughty, J., Dunn, A. & Williams, M. (2008), 'A hypothetical life cycle for Antarctic toothfish *Dissostichus mawsoni* in the Ross Sea region', *CCAMLR Science* **15**, 35–53.

Hankin, R. (2005), 'Introducing BACCO, an R package for Bayesian analysis of computer code output', *Journal of Statistical Software* **14**(16), 1–21.

Hankin, R. (2012), 'Introducing multivator: a multivariate emulator', *Journal of Statistical Software* **46**(8), 1–20.

Harley, S., Myers, R. & Dunn, A. (2001), 'Is catch-per-unit-effort proportional to abundance?', *Canadian Journal of Fisheries and Aquatic Science* **58**, 1760–1772.

Harwood, J. & Stokes, K. (2003), 'Coping with uncertainty in ecological advice: lessons from fisheries', *Trends in Ecology and Evolution* **18**(12), 617–622.

Hastings, W. (1970), 'Monte Carlo Sampling methods using Markov chains and their applications', *Biometrika* **57**(1), 97–109.

Haylock, R. & O'Hagan, A. (1996), *On inference for outputs of computaion-*

ally expensive algorithms with uncertainty on the inputs, Oxford University Press, pp. 629–637.

Henderson, D., Boys, R., Krishnan, K., Lawless, C. & Wilkinson, D. (2009), 'Bayesian emulation and calibration of a stochastic computer model of mitochondrial DNA deletions in substantia nigra neurons', *Journal of the American Statistical Association* **104**(485), 76–87.

Hewitt, D., Lambert, D., Hoenig, J. & Lipcius, R. (2007), 'Direct and indirect estimates of natural mortality for Chesapeake Bay blue crab', *Transactions of the American Fisheries Society* **136**, 1030–1040.

Hilborn, R. (2003), 'The state of the art in stock assessment: where we are and where we are going', *Scientia Marina* **67**(1), 15–20.

Hilborn, R. & Kennedy, R. B. (1992), 'Spatial pattern in catch rates: A test of economic theory', *Bulletin of Mathematical Biology* **54**(23), 263–273.

Hilborn, R. & Mangel, M. (1997), *The Ecological Detective: Confronting Models with Data*, Princeton Universtiy Press.

Hilborn, R. & Walters, C. J. (1992), *Quantitative Fisheries Stock Assessment: Choice, Dynamics and Uncertainty*, Chapman and Hall.

Horodysky, A., Kerstetter, D., Latour, R. & Graves, J. (2007), 'Habitat utilization and vertical movements of white marlin (*Tetrapturus albidus*) released from commercial and recreational fishing gears in the western North Atlantic Ocean: inferences from short duration pop-up archival satellite tags', *Fisheries Oceanography* **16**(3), 240–256.

Hoshino, E., Milner-Gulland, E. & Hillary, R. (2014), 'Why model assumptions matter for natural resource management: interactions between model structure and life history in fisheries models', *Journal of Applied Ecology* **51**(3), 632–641.

Ianelli, J., Honkalehto, T., Barbeaux, S., Kotwicki, S., Aydin, K. & Williamson, N. (2013), 'Assessment of the walleye pollock stock in the Eastern Bering Sea', *NPFMC Bering Sea and Aleutian Islands SAFE* pp. 53–152.

Ianelli, J. & McAllister, M. (1997), 'Bayesian stock assessment using catch-

age data and the sampling-importance resampling algorithm', *Canadian Journal of Fisheries Aquatic Science* **54**(2), 284–300.

Jonsen, I., Basson, M., Bestley, S., Bravington, M., Patterson, T., Pedersen, M., Thomson, R., Thygesen, U. & Wotherspoon, S. (2013), 'State-space models for bio-loggers: a methodological roadmap', *Deep-sea Research II* **88**, 34–46.

Jorgensen, C. & Fiksen, O. (2006), 'State-dependent energy allocation in cod (*Gadus morhua*)', *Canadian Journal of Fisheries Aquatic Science* **63**, 186–199.

Kennedy, M. & O'Hagan, A. (2000), 'Predicting the output from a complex computer code when fast approximations are available', *Biometrika* **87**(1), 1–13.

Kensler, C. B. & Skrzynski, W. (1970), 'Commercial landings of the spiny lobster *Jasus verreauxi* in New Zealand (Crustacea: Decapoda: Palinuridae)', *New Zealand Journal of Marine and Freshwater Research* **4**, 46–54.

Kim, S., Breen, P. A. & Andrew, N. L. (2002), Evaluation of the paua stock assessment model with an individual-based operating model, Technical report, National Institute of Water and Atmospheric Research Ltd.

Lambert, P. C., Sutton, A. J., Burton, P. R., Abrams, K. R. & Jones, D. R. (2005), 'How vague is vague? A simulation study of the impact of the use of vague prior distributions in MCMC using WinBUGS', *Statistics in Medicine* **24**(15), 2401–2428.

Little, R., Fulton, E., Gray, R., Hayes, D., Lyne, V., Scott, R., Sainsbury, K. & McDonald, D. (2006), Management strategy evaluation results and discussion for Australia's North West Shelf, Technical report, CSIRO.

Magnusson, A. & Hilborn, R. (2007), 'What makes fisheries data informative?', *Fish and Fisheries* **8**, 337–358.

Magnusson, A., Punt, A. & Hilborn, R. (2013), 'Measuring uncertainty in fisheries stock assessment: the delta method, bootstrap, and MCMC', *Fish and Fisheries* **14**(3), 325–342.

Mangel, M., Brodziak, J. & DiNardo, G. (2010), 'Reproductive ecology and

scientific inference of steepness: a fundamental metric of population dynamics and strategic fisheries management', *Fish and Fisheries* **11**(1), 89–104.

Marin, J.-M. & Robert, C. P. (2010), *Bayesian Core: A Practical Approach to Computational Bayesian Statistics*, Springer.

Maunder, M. & Harley, S. (2011), 'Using cross validation model selection to determine the shape of nonparametric selectivity curves in fisheries stock assessment models', *Fisheries Research* **110**(2), 283–288.

McAllister, M. K., Pikitch, E. & Babcock, E. (2001), 'Using demographic methods to construct Bayesian priors for the intrinsic rate of increase in the Schaefer model and implications for stock rebuilding', *Canadian Journal of Fisheries and Aquatic Science* **58**(9), 1871–1890.

McKay, M., Conover, W. & Beckman, R. (1979), 'Comparison of three methods for selecting values of input variables in the analysis of output from a computer code', *Techonometrics* **21**, 239–245.

Mertz, G. & Myers, R. (1996), 'An extended cohort analysis: incorporating the effect of seasonal catches', *Canadian Journal of Fisheries Aquatic Science* **53**, 159–163.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. & Teller, E. (1953), 'Equation of state calculations by fast computing machines', *Journal of Chemical Physics* **21**, 1087–1091.

Meyer, R. & Millar, R. (1999), 'Bugs in Bayesian stock assessments', *Canadian Journal of Fisheries and Aquatic Science* **56**, 1078–1087.

Millar, R. & Meyer, R. (2000), 'Bayesian state-space modeling of age-structured data: fitting a model is just the beginning', *Canadian Journal of Fisheries Aquatic Science* **57**, 43–50.

Ministry for Primary Industries (2014*a*), Fisheries Assessment Plenary, May 2014: stock assessments and stock status, Technical report, MPI.

Ministry for Primary Industries (2014*b*), Fisheries Assessment Plenary, November 2014: stock assessments and stock status, Technical report, MPI.

Montgomery, S., Liggins, G., Craig, J. & McLeod, J. (2009), 'Growth of the spiny lobster *Jasus verreauxi* (Decapoda: Palinuridae) off the east coast of Australia', *New Zealand Journal of Marine and Freshwater Research* **43**(1), 113–123.

Moore, E. (1920), 'On the reciprocal of the general algebraic matrix', *Bulletin of the American Mathematical Society* **26**(9), 394–395.

Mormede, S. & Dunn, A. (2013), 'Investigation of potential biases in the assessment of Antarctic toothfish in the Ross Sea fishery using outputs from a spatially explicit operating model', *WG-SAM* **13/36**, 9.

Mormede, S. & Dunn, A. (2014), 'A stock assessment model of Antarctic toothfish (*Dissostichus mawsoni*) in the Ross Sea region incorporating multi-year mark-recapture data', *CCAMLR Science* **21**, 39–62.

Mormede, S., Dunn, A. & Hanchet, S. (2011), 'Assessment models for Antarctic toothfish (*Dissostichus mawsoni*) in the Ross Sea for the years 1997-98 to 2010-11', *WG-FSA* **11/42**, 40.

Mormede, S., Dunn, A. & Hanchet, S. (2013), 'Assessment models for Antarctic toothfish (*Dissostichus mawsoni*) in the Ross Sea for the years 1997-98 to 2010-13', *WG-FSA* **13/51**, 36.

Mormede, S., Dunn, A., Parker, S. & Hanchet, S. (2013), 'Further development of a spatially explicit population dynamics operating model for Antarctic toothfish in the Ross Sea region', *WG-SAM* **13/35**, 21.

Neilsen, A., Bigelow, K., Musyl, M. & Sibert, J. (2006), 'Improving light-based geolocation by including sea surface temperature', *Fisheries Oceanography* **15**, 314–325.

Nelder, J. & Wedderburn, R. (1972), 'Generalised linear models', *Journal of the Royal Statistical Society: Series A* **137**, 370–384.

Nielsen, A. & Berg, C. W. (2014), 'Estimation of time-varying selectivity in stock assessments using state-space models', *Fisheries Research* **158**, 96–101.

Oakley, J. (1999), Bayesian Uncertainty Analysis for Complex Computer Codes, PhD thesis, School of Mathematics and Statistics.

Oakley, J. (2004), 'Estimating percentiles of uncertain computer code outputs', *Applied Statistics* **53**, 83–93.

Oakley, J. & O'Hagan, A. (2002), 'Bayesian inference for the uncertainty distribution of computer model outputs', *Biometrika* **89**, 769–784.

Oakley, J. & O'Hagan, A. (2004), 'Probabilistic sensitivity analysis of complex models: a Bayesian approach', *Journal of the Royal Statistical Society: Series B* **66**, 751–769.

Parker, S., Dunn, A., Mormede, S. & Hanchet, S. (2013), 'Descriptive analysis of the toothfish (*Dissostichus* spp.) tagging programme in Subareas 88.1 & 88.2 for the years 2000-01 to 2012-13', *WG-FSA* **13/49**, 35.

Parker, S., Hanchet, S. & Horn, P. (2014), 'Stock structure of Antarctic toothfish in Statistical Area 88 and implications for assessment and management', *WG-SAM* **14/26**.

Parker, S., Webber, D. & Arnold, R. (2014), 'Deployment and recovery of an archival tag on an Antarctic toothfish in the Ross Sea', *WG-FSA* **14/64**, 16.

Patterson, H. & Thompson, R. (1971), 'Recovery of inter-block information when block sizes are unequal', *Biometrika* **58**, 545–554.

Pella, J. & Tomlinson, P. (1969), 'A generalised stock-production model', *Bulletin of the Inter-America Tropical Tuna Commision* **13**, 421–458.

Penrose, R. (1955), 'A generalized inverse for matrices', *Proceedings of the Cambridge Philosophical Society* **51**, 406–413.

Pope, J. (1972), 'An investigation of the accuracy of Virtual Population Analysis using cohort analysis', *International Commission for the Northwest Atlantic Fisheries Research Bulletin* **9**, 65–74.

Press, W., Flannery, B., Teukolsky, S. & Vetterling, W. (1986), *Numerical Recipes: the Art of Scientific Computing*, Cambridge University Press.

Prince, J. (2003), 'The barefoot ecologist goes fishing', *Fish and Fisheries* **4**, 359–371.

Punt, A. (2003*a*), 'Extending production models to include process error

in the population dynamics', *Canadian Journal of Fisheries Aquatic Science* **60**, 1217–1228.

Punt, A. (2003*b*), 'The performance of a size-structured stock assessment method in the face of spatial heterogeneity in growth', *Fisheries Research* **65**, 391–409.

Punt, A., Hurtado-Ferro, F. & Whitten, A. (2014), 'Model selection for selectivity in fisheries stock assessments', *Fisheries Research* **158**, 124–134.

Quinn, T. (1992), 'Ruminations on the development and future of population dynamics models in fisheries', *Natural Resource Modeling* **16**(4), 341–392.

Ralston, S. & O'Farrell, M. (2008), 'Spatial variation in fishing intensity and its effect on yield', *Canadian Journal of Fisheries and Aquatic Science* **65**, 588–599.

Rencher, A. (2000), *Linear Models in Statistics*, Wiley.

Richards, F. (1959), 'A flexible growth function for empirical use', *Journal of Experimental Botany* **10**(2), 290–301.

Ricker, W. (1954), 'Stock and recruitment', *Journal of the Fisheries Research Board of Canada* **11**, 559–623.

Ridgeway, K., Dunn, J. & Wilkin, J. (2002), 'Ocean interpolation by four-dimensional least squares - application to the waters around Australia', *Journal of Atmospheric and Oceanic Technology* **19**(9), 1357–1375.

Schaefer, M. B. (1954), 'Some aspects of the dynamics of populations important to the management of the commercial marine fisheries', *Bulletin of Mathematical Biology* **1**(2), 25–56.

Scheffer, M., Baveco, J., DeAngelis, D., Rose, K. A. & van Nes, E. (1995), 'Super-individuals a simple solution for modelling large populations on an individual basis', *Ecological Modelling* **80**, 161–170.

Schnute, J. (1987), 'Data, uncertainty, model ambiguity, and model identification', *Natural Resource Modeling* **2**, 159–212.

Seber, G. (1982), *The Estimation of Animal Abundance and Related Parameters*, Blackburn Press.

Seitz, A., Wilson, D., Norcross, B. & Neilsen, J. (2005), 'Pop-up Archival Transmitting (PAT) tags: a method to investigate the migration and behaviour of Pacific halibut *Hippoglossus stenolepis* in the Gulf of Alaska', *Alaska Fisheries Research Bulletin* **10**, 124–136.

Shelton, A., Satterthwaite, W., Beakes, M., Munch, S., Sogard, S. & Mangel, M. (2013), 'Separating intrinsic and environmental contributions to growth and their population consequences', *American Naturalist* **181**(6), 799–814.

Siler, J., Foris, W. & McInerny, M. (1986), Spatial heterogeneity in fish parameters within a reservoir, *in* G. Hall & M. Van Den Avyle, eds, 'Reservoir Fisheries Management: Strategies for the 80's', pp. 122–136.

Smallegange, I. & Coulson, T. (2012), 'Towards a general, population-level understanding of eco-evolutionary change', *Trends in Ecology and Evolution* **28**, 143–148.

Snyder, J. (1926), *Map projections - a working manual*.

Stephenson, R. (1999), 'Stock complexity in fisheries management: a perspective of emerging issues related to population sub-units', *Fisheries Research* **43**, 247–249.

Stevenson, M., Hanchet, S., Mormede, S. & Dunn, A. (2011), 'A characterisation of the toothfish fishery in Subareas 88.1 and 88.2 from 1997-98 to 2010-11', *WG-FSA* **11/45**, 31.

Taylor, I. & Methot, R. (2013), 'Hiding or dead? A computationally efficient model of selective fisheries mortality', *Fisheries Research* **142**, 75–85.

Teo, S., Boustany, A., Blackwell, S., Walli, A., Weng, K. & Block, B. (2004), 'Validation of geolocation estimates based on light level and sea surface temperature from electronic tags', *Marine Ecology Progress Series* **283**, 81–98.

Thorson, J., Stewart, I. & Punt, A. (2012), 'Development and application of an agent-based model to evaluate methods for estimating relative abun-

dance indices for shoaling fish such as Pacific rockfish (*Sebastes* spp.)', *ICES Journal of Marine Science* .

Vernon, I., Goldstein, M. & R., B. (2010), 'Galaxy formation: a Bayesian uncertainty analysis', *Bayesian Analysis* **5**(4), 619–670.

von Bertalanffy, L. (1934), 'Untersuchungen uber die Gesetzlichkeiten des Wachstums I. Allgemeine Grundlagen der Theorie, mathematische und physiologische Gesetzlichkeiten des Wachstum bei Wassertieren', *Wilhelm Roux' Arch. Entwicklungsmech. Org.* **131**, 613–652.

Walters, C. J. & Hilborn, R. (1976), 'Adaptive control of fishing systems', *Journal of the Fisheries Research Board of Canada* **33**, 145–159.

Webber, D. (2013), 'Characterisation, catch per unit effort standardisation, and exploring alternative minimum legal sizes of packhorse rock lobsters in New Zealand', *Unpublished* p. 51.

Webber, D. & Thorson, J. (2015), 'Variation in growth among individuals and over time: a case study and simulation experiment involving tagged Antarctic toothfish', *Fisheries Research* **XX**(X).

Welch, D. & Eveson, J. (1999), 'An assessment of light-based geoposition estimates form archival tags', *Canadian Journal of Fisheries Aquatic Science* **56**, 1317–1327.

Williams, R. & Lamb, T. (2002), 'Behaviour of *Dissostichus eleginoides* fitted with archival tags at Heard Island: preliminary results', *WG-FSA* **02/60**, 16.

Wolf, M. & Weissing, F. (2012), 'Animal personalities: consequences for ecology and evolution', *Trends in Ecology and Evolution* **27**, 452–461.