UNIVERSITY OF CALIFORNIA
SANTA CRUZ

BAYESIAN NONPARAMETRICS FOR INFERENCE OF ECOLOGICAL DYNAMICS

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

STATISTICS AND STOCHASTIC MODELING

by

Anand Patil

September 2007

The Dissertation of Anand Patil
is approved:

Marc Mangel, Chair

Athanasios Kottas

Stephan Munch

Raquel Prado

Lisa C. Sloan
Vice Provost and Dean of Graduate Studies

Table of Contents

List of Figures

List of Tables

Abstract

Bayesian nonparametrics for inference of ecological dynamics

by

Anand Patil

Bayesian analysis allows 'forward' dynamical models from ecological theory to be incorporated directly into statistical models. This procedure is useful because ecological models are the best available representations of the processes that give rise to ecological data. Prior information is often available for ecological models' parameters, and posterior inferences of those parameters are biologically meaningful.

However, when viewed as priors on the space of relationships between past and future system states dynamical models tend to be relatively restrictive, as many ecological systems can plausibly be modeled in several different ways. Placing nonparametric Gaussian process priors on rate functions captures more of the range of plausible dynamics and allows inference of the rate functions.

This dissertation introduces two Bayesian methods for such inference. The first method is applicable to relatively complete models for multispecies dynamics, but is computationally expensive. This method is used to infer the functional response from a classic predator-prey dataset. The results contribute to the discussion of ratio-dependence in predator attack rates: the inferred functional response is prey-dependent at low predator densities and ratio-dependent at higher predator densities.

The second method is much cheaper and easier to implement, but is relatively assumption-bound. This method is applied to fishery datasets, and optimization and forecasting for resource management are found to become more difficult as a result of the model uncertainty expressed by the nonparametric prior. New management tools based on local dynamics are proposed to address the difficulties.

The dissertation concludes by presenting software that allows users to construct and fit a wide range of probability models involving Gaussian processes using a directed acyclic graph-based Bayesian model specification language. Use of the software requires conceptual understanding only, as opposed to familiarity with Gaussian process-related algorithms.

Acknowledgments

Thanks first to Marc Mangel, who offers his students unconditional support, genuine interest and an unmatched breadth of knowledge and experience in applied mathematics and applied science. Perhaps most importantly, he gives them the freedom to find their own way, even if it takes a while. I am very fortunate to have had the opportunity to work with him.

Thanasis Kottas introduced me to Bayesian nonparametrics, a wonderful set of techniques that opens up possibilities I couldn't have imagined four years ago. Raquel Prado taught me the theory and craft of time-series analysis, an eminently useful skill set for analysis of ecological data. I am grateful to them for what they've taught me and for taking the time to be on my dissertation committee. Thanks to Steve Munch for working with me on the most challenging projects I've ever tried and making it fun.

My other collaborators and advisors have included Alec MacCall and Steve Ralston, who provided ideas, context and data for my fisheries work; Toufic Suidan, who helped me out frequently when I got in over my mathematical head; Bobby Gramacy and Dan Merl, who answered a long sequence of programming questions; and David Huard and Chris Fonnesbeck, who have worked with me on the Bayesian statistics package mentioned in chapter 4.

Thanks to the Mangel lab for good times and great feedback. I will fondly remember lab retreats and weekly homework sessions at Gelatto Mania. I deeply appreciate the Slatkin lab's provision of interesting work, warm friendship and a roof over

my head in Berkeley. The AMS faculty have been very generous with their time in answering my questions over the years.

I would never have made it through the bewildering dissertation process without the guidance of Tracie Tucker. Zack Sanborn and Vaidehi Campbell made their spare beds and couches available when I had to spend the night in Santa Cruz, even on a moment's notice.

And finally, thanks to Anne Short for much-needed advice and understanding.

My financial support came from the following sources:

- UCSC Dean's Fellowship.
- CSTAR, a partnernship between NMFS/SCL and UCSC.
- NSF grant DMS 0310542 to Marc Mangel.
- CalFed award U05SC040 to Marc Mangel.

Chapter 1

Introduction

Many important concepts in theoretical population ecology originated in 'forward' dynamical models [47]. Some model-derived concepts have become so important that they appear in natural resource law. The Magnuson-Stevens fishery conservation act, for example, actually requires fish stocks to be managed by reference to maximum sustainable yield, one of the most basic results of mathematical fishery models [71].

Standard statistical models do not tend to be readily interpretable in terms of ecologists' dynamical models of choice. Despite the central role of these models in ecological theory, they are generally shoehorned into an ill-fitting statistical framework when the data arrive. Taper and Lele [88] argue that dynamical thinking as opposed to 'plain' statistical investigation of association is indispensible for learning about causiation:

> A dynamical model is simply a model concerned with change. Elucidating causation requires a dynamical model at a very fundamental level. An agent of some kind influences a situation and causes a change. This is a dynamical description.

...

> A dynamical model, when formed as the basis for statistical models, helps refine our thinking about the forces acting on the system, their underlying mechanisms, the time course of the response, and the critical variables to measure. In this sense, a dynamical model arises *in advance of* a statistical model.

From a Bayesian point of view in particular, the dynamical model and the statistical model do not need to be separated at all; there is no conceptual reason why complex, possibly nonlinear 'forward' ecological models should not be incorporated seamlessly into statistical models and fit directly to data. At the conceptual level, this is very clear. The factors currently limiting the potential applications of forward models as statistical models tend to be technical: there is a relative dearth of efficient fitting algorithms.

Ecological models present another, non-computational difficulty: ecological situations usually can be modeled in several different ways, and there is rarely agreement on which is the 'right' model. As the following sections will show, even when the field more or less agrees on a dynamical framework the forms of certain rate functions can be highly controversial. A rate function in ecological modeling gives the expected rate at which some process proceeds, for example birth rates, death rates and predation rates. The arguments of rate functions always include the state of the system, and sometimes include extrinsic inputs such as climactic forcing.

Such uncertainty could cause problems for the direct use of forward models as a basis for statistical inference, but Bayesian nonparametrics offer an attractive resolution. Controversial or unknown functions can be declared unknown *a priori*, and inferred

directly from data.

1.1 Inference of the functional response in a stochastic predator-prey (*Paramecium-Didinium*) system

Chapter 2 addresses a particularly contentious issue: the functional response in predator-prey systems, or the rate at which prey are consumed per unit predator. After the pioneering work of Holling [37], it became a common assumption that this rate depended only on prey density. Arditi and Ginzburg brought that assumption into question [5] and contended that the functional response should depend on the ratio of predator and prey densities. The essence of their argument is that when predator densities are high relative to prey densities, predators will be compelled to share prey in some way and their foraging will become less efficient.

Arditi and Ginzburg's arguments drew a number of followers, but also a backlash led by Abrams [3]. Abrams preferred the old prey-dependent formulation, arguing that it could be made to account for predator-sharing in a more appropriate way and that ratio-dependence was based on a misinterpretation of the purpose of biological differential equation models. The advocates of ratio dependence rejoined [7], citing a large volume of indirect evidence based on observations of trophic levels.

In 2000, Abrams and Ginzburg [5] authored a paper together, explaining the points on which they agreed and those on which they still differed. They enumerated their agreements as follows:

- Precise predator dependence and ratio dependence will both be rare.
- Predator dependence in the functional response will be common, though it will not necessarily be ratio dependence.
- Responses will often be affected by species other than predator and prey.
- Predator dependence of the numerical response will be common. The numerical response is the rate at which prey consumed are converted to new predators, and is commonly assumed to be a linear function.

A common thread through these points is a recognition of model uncertainty.

Possibly as a result of the controversy surrounding the functional response, modeling studies using unique functional forms have proliferated. In the same year as Abrams and Ginzburg's paper was published, Jost and Ellner published a study [45] which listed several major currently-available functional responses in a table. They are shown in table 1.1, in which x denotes prey density and y denotes predator density. They stated that there were over 20 functional responses in existence, and new forms have been proposed since then (e.g., Weitz and Levin [92]).

Jost and Ellner's paper is relevant to this dissertation for a much more important reason: it is a recent installment of the efforts of a small group of researchers to infer ecological functions using nonparametric or semiparametric methods. The earliest paper I am aware of in this series was published in 1997 [24]. In this paper, Ellner et al. prove the concept by fitting the stochastically forced differential equation

Holling type I	ax
Holling type II	$\frac{ax}{1+ahx}$
Holling type III	$\frac{ax^2}{1+ahx^2}$
Ivlev	$a\left(1-e^{-tx}\right)$
ratio-dependent II	$\frac{ax}{y+ahx}$
ratio-dependent III	$\frac{ax^2}{y^2+ahx^2}$
Hassell-Varley type I	axy^{-m}
Hassell-Varley type II	$\frac{ax}{y^2+ahx}$
DeAngelis-Beddington	$\frac{ax}{(1+ahx)(1+cy)}$
Crowley-Martin	$\frac{ax}{(1+ahx)(1+cy)}$
Watts	$a\left(1-e^{-cx/y^m}\right)$

Table 1.1: Several commonly-used functional forms, compiled by Jost and Ellner [45]. y denotes predator density and x denotes prey density; all other variables are parameters.

$\dot{x}(t) = a(t)f(x(t-\tau)) + g(x(t))$, where x is the size of a simulated population, a is 'effectively a white noise process,' and f and g are unknown functions. Observing that the model is an approximate generalized additive model, the authors describe f and g using a spline basis and are able to recover the functions with encouraging accuracy. Ellner, Seifu and Smith [25] make the role of nonparametrics in the functional response debate more explicit, and obtain more encouraging results using real data.

In chapter 2, I continue the line of investigation opened by these researchers by grounding nonparametric inference of the functional response in a Bayesian framework. The probability model incorporates process randomness, observation error and prior information about life-history parameters in an interpretable and coherent way. Most importantly, a Gaussian process prior is used to restrict the functional response to a reasonable class without imposing the extra restrictions that come with parametric forms.

Nonparametric inference of interspecies interactions has enormous potential for rational management of human interventions in ecosystems that incorporates model uncertainty. With this in mind, a fitting method is chosen with a view to facilitating scaling up to multispecies systems. Several alternative fitting methods are discussed in light of this goal.

1.2 Model uncertainty for fisheries science and implications for management

Ellner, Seifu and Smith [25] make the following observation regarding statistical model comparison:

> When models are compared with data in order to evaluate competing hypotheses about causal processes, the choice of functional forms for each process-rate equation is an undesirable confounding factor. Model 1 may fit better than Model 2 because Model 1 makes the right mechanistic assumptions and Model 2 doesn't. But it is also possible that Model 2 simply suffers from a poor choice of functional form for a process rate that is not a part of Model 1. Seemingly innocuous choices between alternative functional forms can have drastic effects on model predictions.

Whereas in chapter 2 I present some of the benefits of incorporating model uncertainty in inferences of ecological dynamics, in chapter 3 I explore the potential dangers of *not* doing so.

The point quoted above is explored in more depth by Wood and Thomas [95]. These researchers consider a model for the infective dynamics of the pest grasshopper *Heiroglyphus daganensis* and the fungal entomopathogen *Metarhizium flavoviride*, which is used for biocontrol. They model the dynamics within a particular wet season using the delay differential equation

$$\dot{H} = -f(A_1)H - SH$$

$$\dot{A}_0 = cf(A_1(t-\tau))H(t-\tau) - cA_0$$

$$\dot{A}_1 = c(A_0 - A_1).$$

Here H represents the density of healthy individuals, A_0 the density of infected individuals, and A_1 the density of infectious individuals and cadavers. c gives the efficiency of transitions between categories. The inter-wet season dynamics and the rule governing infection rate S due to human spraying are described in their paper.

Wood and Thomas fit three functional forms for the transmission rate f to direct measurements of transmission:

Hochberg	$f(A_1) = \beta A_1^{1+q}$
Briggs-Godfray	$f(A_1) = k \log\left(1 + \alpha \frac{A_1}{k}\right)$
Michaelis-Menten	$f(A_1) = a \frac{A_1}{b + A_1}$.

The parameters β, α, a, k and b are positive-valued, and q is a real number. As shown in the left panel of figure 1.1, the three functions with their respective best-fitting parameter values are visually very similar within the range of the data. However, as shown in figure 1.2, the projected dynamics of infection are radically different. The reason is simple: the Hochberg model's slope is infinite at the origin, so it is able to rapidly produce an appreciable infectious population every wet season. This model assumption calls for careful scrutiny, especially in light of its dynamical relevance, but that is beside Wood and Thomas' point.

Wood and Thomas provide a conceptual framing for this 'super-sensitivity to model structure': they think of parametric families as manifolds in a function space, and argue that the dynamics may be sensitive to perturbations in directions orthogonal to those manifolds. Their explanation is a commonly-stated motivation for Bayesian nonparametrics, and in fact assigning a Gaussian process or other functional prior to uncertain functions seems like a good way to evade such 'gotchas' in practice. If de-

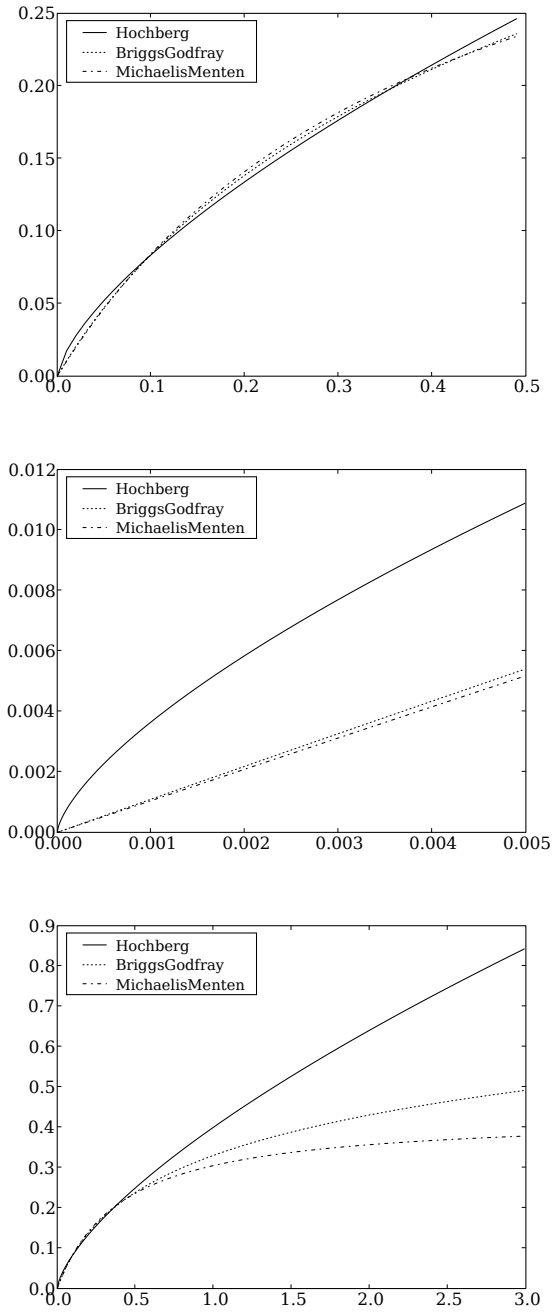


Figure 1.1: **Upper:** Wood and Thomas's figure shows almost imperceptible differences between the fits for the three candidate functional forms. **Middle:** Close to the origin, the functions necessarily diverge; the Briggs-Godfray and Michaelis Menten models approach the origin linearly, whereas the Hochberg model approaches as a radical. **Lower:** The three models diverge to the right of the apparent data range as well.

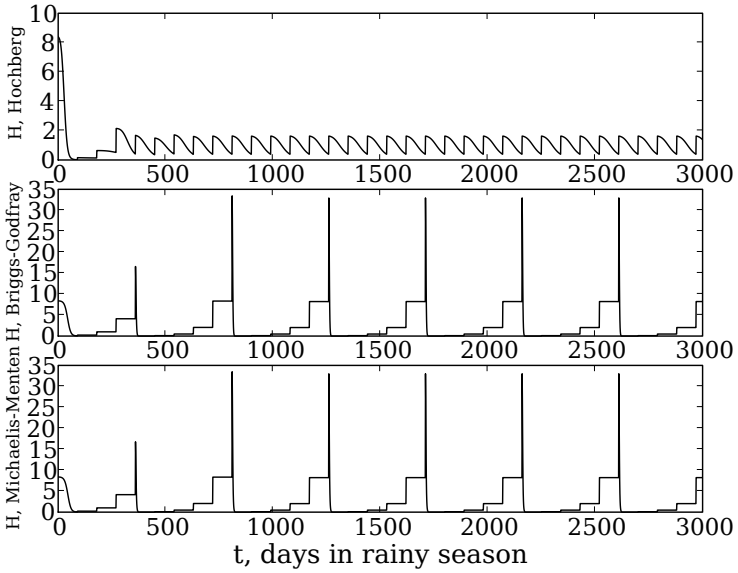
sired, a prior could have been developed to express uncertainty regarding the Lipschitz parameter [86] of the transmission function at the origin.

Wood and Thomas' point is compelling: in instances where predictions [52] must be made in the presence of model uncertainty, that uncertainty *should* be encoded via a prior on unknown functions that captures the range of plausible dynamics. In chapter 3, I apply this concept to the problem of forecasting the future of overfished stocks. The results are only the very beginning of the potential applications in fisheries science, and the implications have consequences that extend beyond basic science and into natural resource policy.

1.3 An intuitive implementation of Gaussian processes

The nonparametric priors used in the first two body chapters are not difficult to understand in words and pictures. Model uncertainty is a very pervasive issue in ecology, and in my experience colleagues in ecology tend to feel that Gaussian process priors, as methods for encoding classes of sensible functions without the unwanted baggage of particular parametric forms, make a great deal of sense.

In fields such as fluid mechanics and dynamical systems, it often happens that the actual mathematics behind a model are very involved but the big picture can be understood graphically and physically without pushing any symbols. Strogatz [87] provides graphics that are incredibly helpful in understanding various bifurcations and features of the phase plane. The convective derivative $\partial_t + v \cdot \nabla$ in the Navier-Stokes



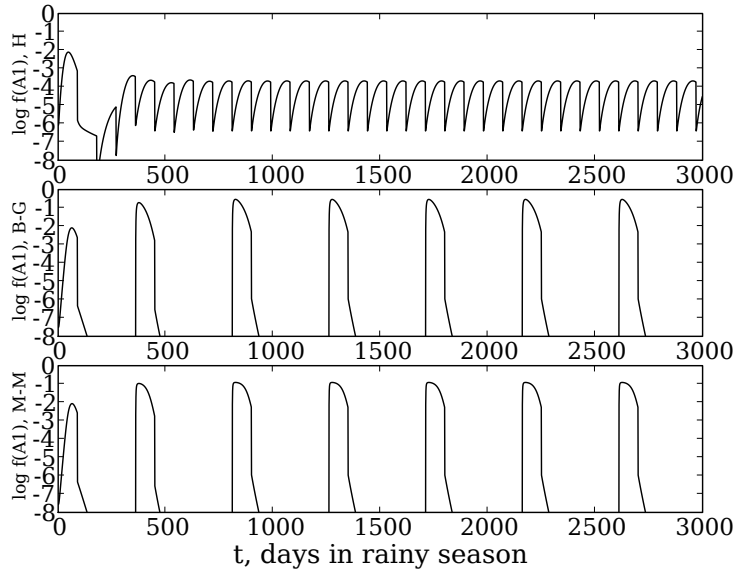


Figure 1.2: An approximate reproduction of Wood and Thomas' projected infection dynamics for the three candidate transmission functions. **Top**: density H of healthy individuals. **Bottom**: the log of A_1, the density of infectious individuals and cadavers. Because the Hochberg transmission rate increases very rapidly in the neighborhood of the origin, it is able to produce an appreciable density of infectious individuals in every wet season, controlling the grasshopper population. However, in the other two models the pathogen effectively dies out in wet seasons in which humans do not spray.

equations [72] is best understood as 'the rate of change due to new material plus material that flows in' and learned via thought experiments or the graphical output of computer experiments before attempting to learn the math.

This type of learning is currently not well supported for nonparametric methods like Gaussian processes. Ecologists who want to use them but do not have sufficient training in linear algebra and/or normal distribution theory to implement the relevant methods currently have two options: depend on a collaborating statistician to elicit their model, or adapt their problem to one of several models 'canned' in a software package.

In chapter 4, I describe a software package that is intended to help users, especially ecologists, gain a conceptual and graphical understanding of Gaussian processes and to incorporate them into larger probability models of their own design using a directed acyclic graph-based [41] model-specification language. The approach of the package is to provide objects that strongly resemble the conceptual and mathematical definitions of mean functions, covariance functions and realizations [1], and that insulate the user from the details of the implementation. The basic objects can be incorporated directly into larger probability models and fit using Markov Chain Monte Carlo (MCMC), a widely applicable algorithm for sampling from posterior distributions. [27]. Like the basic objects, the package's MCMC support is designed to allow users to proceed from a conceptual understanding.

The package's documentation (available from `code.google.com/p/gaussian-process`) introduces users to these objects using graph-

ical examples and a handful of applications drawn from ecology.

Chapter 2

A predator-prey (*Paramecium-Didinium*) system with unknown functional response

2.1 Introduction

Population cycles have drawn the interest of ecologists for more than three quarters of a century [53, 90]. Although the most prominent explanation has been coupling between predators and prey, the presence or absence of cycling behavior depends very strongly on the functional response, or the rate at which prey are eaten per predator.

The time evolution of predator (y_t) and prey (x_t) population sizes or biomass

is commonly modeled using stochastic systems of the form

$$\begin{aligned} dx_t &= [h_x(x_t) - f(x_t, y_t)]\, dt + q_x(x, y) dW_{x,t} \\ dy_t &= [h_y(y_t) + \beta f(x_t, y_t)]\, dt + q_x(x, y) dW_{y,t}, \end{aligned} \tag{2.1}$$

or their deterministic skeletons. Here h_x and h_y give the growth or decay rates of the prey and predator, respectively, and β gives the efficiency with which consumed predators are converted to new prey. The stochastic differentials $dW_{x,t}$ and $dW_{y,t}$ can be heuristically interpreted as independent normal random variables with mean zero and variance dt [82], which are scaled by q_x and q_y. The functional response is given by $f(x_t, y_t)/y_t$.

Since the classic work of Holling [37], the functional response has been the subject of much empirical and theoretical research. For the past twenty years a debate has persisted in the literature over whether the functional response is solely determined by the number of prey available ('prey-dependent' [53]), by the relative abundance of prey to predators ('ratio-dependent' [8]), or more generally by both prey and predator densities [12, 21]. More than twenty models have been derived to approximate the effects of space, interference, reproductive delays, etc. These are reviewed by Jeschke et al. [42].

Many attempts have been made to distinguish between competing theories of the functional response (See e.g. Abrams [4]), ranging from large-scale compilations of coarse-grained predictions [7] to likelihood-based model selection [33, 44]. Unfortuantely, conclusions drawn from such exercises are limited to the set of models considered. Although one model will always fit better than the others, this approach offers no way to

identify alternatives that are not among the candidates. This issue seems quite serious in light of the fact that the choice of model structure can have extreme consequences for the quality of predictions [95].

2.1.1 Ratio dependence

As noted in the introduction to this dissertation, there has been particularly intense controversy in the predator-prey literature over whether it is preferable to model the functional response as a function of prey density y_t or as a function of the ratio of predator and prey densities y_t/x_t [5]. These competing assumptions are known as 'prey dependence' and 'ratio dependence', respectively. Although ratio dependence has been criticized as ad hoc and difficult to justify mechanistically [3], it can be shown to result from predator sharing of prey at high predator densities by a simple argument.

Suppose that a predator engaged in searching for prey encounters its next victim between times t and $t + dt$ with probability $\lambda x_t dt + o(dt)$, and that upon encountering a victim it must spend time τ handling the victim before moving on to the next. Assuming x_t is changing slowly, the expected time that elapses between successive attacks by the predator is approximately $\tau + 1/\lambda x_t$. The expected proportion of predators currently engaged in prey attacks is $p_e = \tau\lambda x_t/(1 + \tau\lambda x_t)$.

Assuming further that predators attack prey without regard to whether the prey are currently being handled by another predator, the expected proportion of prey not being handled at any given time is $(1 - 1/x_t)^{p_e y_t} \approx \exp(-p_e y_t/x_t)$ [49]. This is also the probability that any given attack by a predator results in a new victim being

consumed. The rate of attacks on new victims by any given predator, or the functional response, is

$$\frac{\lambda x_t}{\tau \lambda x_t + 1} \exp\left(-p_e \frac{y_t}{x_t}\right).$$

When the ratio y_t/x_t is small, the exponential term is approximately one and the functional response is the Holling disk equation [4], a classic prey-dependent functional response. The dependence on the ratio y_t/x_t is linear in this limit. When $\lambda x_t \tau$ is large, the functional response is approximately $\exp(-y_t/x_t)/\tau$, which is strictly ratio-dependent.

2.1.2 Previous semiparametric inferences of the functional response

In a pioneering study, Jost and Ellner [45] use gradient matching [24, 25] to fit a regression spline embedded in a set of coupled delay-differential equations to infer the functional response. They allow functional responses of the form $f(xy^{-a})$, where x and y represent the prey and predator population sizes. This model allows for prey-dependent ($a = 0$), ratio-dependent ($a = 1$), or Hassell-Varley ($0 < a < 1$) [34] dynamics. Overall, they conclude that models with some degree of predator dependence in the functional response fit their data better than models with only prey dependence.

Although the work of Jost and Ellner [45] represents a huge step forward, it should be extended in two critical ways. First, general inferences of fully two-place functional responses are needed. Second, there is abundant evidence that stochasticity can be important in driving population dynamics [70]; this possibility should be allowed

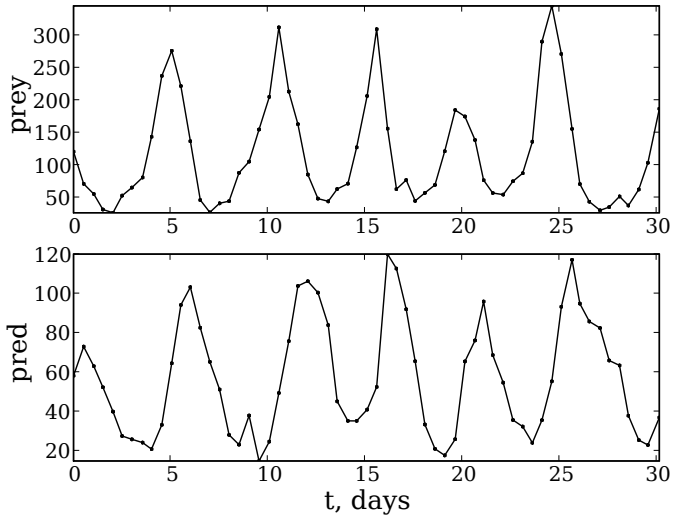


Figure 2.1: The dataset of Veilleux [89] that will be considered in this chapter. The time-series of prey abundance is shown in the upper panel, and the time-series of predator abundance is shown in the lower panel. Time is measured in days.

for when making inferences.

The work in this chapter was done in collaboration with Stephan B. Munch of Stony Brook University. In this chapter, we describe a method that extends the work of Jost and Ellner [45] in two critical ways. First, it provides general inferences of fully two-place functional responses. Second, it allows for the possibility that stochasticity can be important in driving population dynamics [70]. We apply the method to a classic protozoan predator-prey time series obtained by Veilleux [89].

2.2 Bayesian probability model

2.2.1 Observation model

Veilleux's experiments were conducted in 6ml of fluid in a Petri dish. The population density of each species in the dish was determined by counting the individuals in each of eight .1ml samples twice daily and averaging the observed density. The dataset is shown in figure 2.1. In the following, the observation times are denoted $\{t_i\}$, $i = 0 \ldots N_o$. The last observation time is denoted T.

Veilleux's sampling protocol allows us to construct a model for the process of observing the population density of the dish at sampling time i. Assume that all organisms are distributed randomly (as a binomial process with constant intensity [63]) throughout the fluid in the petri dish. The probability that an individual finds itself in each of the eight .1ml samples is 1/60, so the joint distribution of the number of individuals in each sample is multinomial:

$$\{\mathrm{x}_i^j\} \sim \mathrm{multi}(6x_{t_i}, \{1/60\}), \quad j = 1 \ldots 8$$

$$\{\mathrm{y}_i^j\} \sim \mathrm{multi}(6y_{t_i}, \{1/60\}), \quad j = 1 \ldots 8$$

where the bin corresponding to not being in any of the samples and its probability are implicit. Here j indexes the .1ml samples. The sum of the samples is therefore binomially distributed, and the normal approximation to the binomial distribution can

be applied:

$$\sum_{j=1}^{8} \mathrm{x}_i^j \sim \mathrm{bin}(6x_{t_i}, 8/60) \approx \mathrm{N}(8x_{t_i}/10, 8x_{t_i}(1-8/60)/10)$$
$$\sum_{j=1}^{8} \mathrm{y}_i^j \sim \mathrm{bin}(6y_{t_i}, 8/60) \approx \mathrm{N}(8y_{t_i}/10, 8y_{t_i}(1-8/60)/10).$$

The density estimates x and y, in individuals per ml, which are the actual data Veilleux provides, are scalar multiples of the total population sizes, so they are also approximately normal:

$$\begin{aligned} \mathrm{x}_i &= \tfrac{10}{8} \textstyle\sum_{j=1}^{8} \mathrm{x}_i^j \ \dot{\sim} \mathrm{N}(x_{t_i}, (13/12)x_{t_i}) \\ \mathrm{y}_i &= \tfrac{10}{8} \textstyle\sum_{j=1}^{8} \mathrm{y}_i^j \ \dot{\sim} \mathrm{N}(y_{t_i}, (13/12)y_{t_i}). \end{aligned} \tag{2.2}$$

Note that we have assumed that paramecia and didinia occur independently in samples. This assumption may not hold if didinia are able to spend most of their time in areas of high paramecium density, or if paramecia are able to spend most of their time in areas of low didinium density.

2.2.2 Dynamical model

We model the population dynamics using the following stochastic Lotka-Volterra-like predator-prey system:

$$\begin{aligned} dx_t &= \left[r_x x_t - \tfrac{x_t^2}{m} - f(x_t, y_t) \right] dt + \sigma_x dW_x \\ dy_t &= \left[r_y y_t + \beta f(x_t, y_t) \right] dt + \sigma_y dW_y, \end{aligned} \tag{2.3}$$

and we condition on $x_t, y_t > 0$ for all $t \in [0, T]$ (since we know a priori that neither species went extinct over the course of the experiment). We will sometimes write this

in vector form,

$$d\mathbf{x}_t = \left(\begin{bmatrix} r_x & 0 \\ 0 & r_y \end{bmatrix} \mathbf{x}_t - \mathbf{x}_t^T \begin{bmatrix} -1/m & 0 \\ 0 & 0 \end{bmatrix} \mathbf{x}_t - \begin{bmatrix} -1 \\ \beta \end{bmatrix} f(x_t, y_t) \right) dt + \sigma d\mathbf{W}, \quad (2.4)$$

where

$$\mathbf{x_t} = \begin{bmatrix} x_t \\ y_t \end{bmatrix}, \quad \sigma = \begin{bmatrix} \sigma_x & 0 \\ 0 & \sigma_y \end{bmatrix}.$$

We interpret the stochastic differentials in the Itô sense; see appendix A.2. We leave extensions such as colored noise [39] for future research.

The prey population tends to increase when small with rate constant $r_x > 0$, and the predator population tends to decrease with rate constant $r_y < 0$ in the absence of prey. In the absence of predation, the stationary distribution of the prey population would be centered in the neighborhood of $r_x m$ because in that region birth balances density-dependent mortality (due to competition for resources, etc) $r_x m$ is commonly denoted K, which is called the carrying capacity of the environment; we use an alternative parametrization for convenience.

We assume that the conversion efficiency is constant, so that the 'numerical response', or the rate at which new predators are created, is a linear function of the rate at which prey are consumed. Deviations from linearity might occur because the value of prey to predators depends on the nutritional state of the prey [89], but we do not consider this possibility. β is the efficiency with which the predator is able to convert consumed prey into new individuals of its own species.

2.2.3 Gaussian process prior for functional response and Fourier series representation

We use a Gaussian process prior for f,

$$f \sim \mathrm{GP}(f_{\mathrm{guess}}, C),$$

where C is a covariance function. We would like to make a choice for C that allows us to constrain the smoothness of f and sample it efficiently. Unfortunately, the standard technique of 'marginalizing' the Gaussian process by only explicitly considering f evaluated at the value of the $\mathbf{x}$ at observation times t_i is difficult to apply here. The value of f at any point along the sample path $\mathbf{x}$, even for non-observation times t, could affect the data. In addition, storing f on a fine grid of N points on a side is not desirable because the covariance matrix would contain N^4 elements.

These considerations point to a spectral representation as a natural way to store f. We will only concern ourselves with f on an interval $[0, L_x] \times [0, L_y]$, where L_x and L_y are loosely determined by the maximum population sizes observed; we do not expect to gain much information about f in regions where we have no data. Also, it makes biological sense to set $f(0, y) = f(x, 0) = 0$. If we restrict ourselves to identical boundary conditions at $f(L_x, y)$ and $f(x, L_y)$, we have two orthogonal and complete trigonometric bases available:

$$\begin{aligned}
&1) \qquad \sin\left(\tfrac{n\pi x}{L_x}\right)\sin\left(\tfrac{m\pi y}{L_y}\right), \qquad n, m = 1 \ldots \infty \\
&2) \quad \sin\left(\tfrac{(n-1/2)\pi x}{L_x}\right)\sin\left(\tfrac{(m-1/2)\pi y}{L_y}\right), \quad n, m = 1 \ldots \infty
\end{aligned}$$

For basis 1 we have $f(L_x, y) = f(x, L_y) = 0$, and for basis 2) we have $\partial_x f(L_x, y) = \partial_y f(x, L_y) = 0$. We have no biological reason to believe either of these conditions is actually true, so we will consider our inferred function as an extension of f. That is, we will set L_x and L_y large enough that changing them by a small amount has little effect on our inference of f in the region of interest (the region where data have been observed.) We do have biological reasons to expect that f increases with both x and y, however, so we select basis 2 on the grounds that f increasing monotonically and coming in flat to the boundary in both x and y is more realistic than f reaching a maximum value and returning to 0 for x and y sufficiently large. The practical upshot of this choice is that we can set L_x and L_y relatively small and still have our inference be insensitive in the region of interest. Smaller L_x and L_y allow us to use fewer Fourier modes while maintaining the same degree of resolution. See Appendix A.1 for a brief discussion of the importance of choosing appropriate boundary conditions.

The Fourier coefficients c_{nm} associated with f and f itself are related as follows:

$$c_{nm} = \frac{4}{L_x L_y} \int_0^{L_x} \int_0^{L_y} \sin\left((n-1/2)\pi \frac{x}{L_x}\right) \sin\left((m-1/2)\pi \frac{y}{L_y}\right) f(x,y) dx dy$$

$$f(x,y) = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} c_{nm} \sin\left((n-1/2)\pi \frac{x}{L_x}\right) \sin\left((m-1/2)\pi \frac{y}{L_y}\right).$$

Under the rescaling $x = \bar{x} L_x$, $y = \bar{y} L_y$, $f(x,y) = \bar{f}(\bar{x}, \bar{y})$, this becomes

$$c_{nm} = 4 \int_0^1 \int_0^1 \sin\left((n-1/2)\pi \bar{x}\right) \sin\left((m-1/2)\pi \bar{y}\right) \bar{f}(\bar{x}, \bar{y}) d\bar{x} d\bar{y}$$

$$\bar{f}(\bar{x}, \bar{y}) = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} c_{nm} \sin\left((n-1/2)\pi \bar{x}\right) \sin\left((m-1/2)\pi \bar{y}\right).$$

Denote the infinite covariance tensor of the Fourier coefficients by K and the covariance

function of $\bar{f}$ by C:

$$\mathrm{Cov}\left(c_{n_1 m_1}, c_{n_2 m_2}\right) = K_{n_1 m_1, n_2 m_2},$$

$$\mathrm{Cov}\left(\bar{f}(\bar{x}, \bar{y}), \bar{f}(\bar{\xi}, \bar{\eta})\right) = C((\bar{x}, \bar{y}), (\bar{\xi}, \bar{\eta})).$$

We will generate the four-dimensional covariance tensor K and the function C on $[0,1] \times [0,1] \times [0,1] \times [0,1]$ by tensor products:

$$K_{n_1 m_1, n_2 m_2} = K_{n_1, n_2} K_{m_1, m_2},$$

$$C\left((\bar{x}, \bar{y}), (\bar{\xi}, \bar{\eta})\right) = C(\bar{x}, \bar{\xi}) C(\bar{\eta}, \bar{y}).$$

As pointed out by Stein [86] on page 55, generating covariances by tensor products is not reasonable for random fields whose behavior should not have a strong dependence on choice of axes. In our case, however, axes are very important, and in fact $f(x, y)$ is most often modeled as a product of a function of x and a function of y (though a tensor product covariance does not restrict f to functions of this form). However, it is not immediately clear that a tensor product covariance induces an *appropriate* dependence on the axes. Further work is warranted to weigh the tradeoff between speed and flexibility.

2.2.4 Matèrn-like covariance function

The covariance matrix $K_{n_1 n_2}$ of the Fourier coefficients and the covariance function C on $[0,1] \times [0,1]$ are related by

$$K_{n_1 n_2} = 4 \int_0^1 \int_0^1 \sin((n_1 - 1/2)\pi \bar{x}) \sin((n_2 - 1/2)\pi \bar{\xi}) C\left(\bar{x}, \bar{\xi}\right) d\bar{x} d\bar{\xi},$$

$$C(\bar{x},\bar{\xi}) = \sum_{n_1=1}^{\infty}\sum_{n_2=1}^{\infty} \sin((n_1-1/2)\pi\bar{x})\sin((n_2-1/2)\pi\bar{\xi})K_{n_1n_2}.$$

To specify the one-dimensional covariances $K_{n_1n_2}$ and $C(\bar{x},\bar{\xi})$, we follow the reasoning outlined by Stein in chapter 2 supporting the Matèrn covariance function. A Matèrn-like covariance function is a desirable choice because one of its parameters is directly related to the differentiability of realizations.

We would like to specify a covariance function using a parameter ν which sets the highest derivative of $\bar{f}$ that exists and is continuous. Because of the boundary conditions we have imposed and our construction using a tensor product, our covariance function will be neither stationary nor isotropic. However, it is easy to see that realizations are continuous if $C(\bar{x},\bar{\xi})$ is continuous at $\bar{x}=\bar{\xi}$. For convenience, we will restrict ourselves to diagonal covariance matrices $K_{n_1n_2} = K_{n_1}1_{n_1=n_2}$.

We make the change of variables $\bar{\xi} = \bar{x}+\hat{x}$ and write $Q_{\bar{x}}(\hat{x}) = C(\bar{x},\bar{x}+\hat{x})$. $Q_{\bar{x}}(\hat{x})$ is like an autocovariance function in that realizations are ν times continuously differentiable if $Q_{\bar{x}}(\hat{x})$ is at least 2ν times continuously differentiable with respect to $\hat{x}$ at $\hat{x}=0$. The Fourier coefficients of $Q_{\bar{x}}$ can be related to K_{n_1} using trigonometric identities:

$$Q_{\bar{x}}(\hat{x}) = \sum_{n_1=1}^{\infty} M_{n_1,\bar{x}}\sin(n_1\pi\hat{x}) + N_{n_1,\bar{x}}\cos(n_1\pi\hat{x}),$$

$$M_{n_1,\bar{x}} = K_{n_1}\sin(n_1\pi\bar{x})\cos(n_1\pi\bar{x}),$$

$$N_{n_1,\bar{x}} = K_{n_1}\sin^2(n_1\pi\bar{x}).$$

If p is an integer, the Fourier coefficients of $Q_{\bar{x}}^{(2p)}$ that contribute to the sum in the neighborhood of the origin are proportional to $n_1^{2p}N_{n_1,\bar{x}}$. It follows that $Q_{\bar{x}}(\hat{x})$ is $2p$-

times continuously differentiable in $\hat{x}$ at the origin if $\sum_{n_1=0}^{\infty} n_1^{2p} K_{n_1} < \infty$. We therefore choose

$$K_{n_1} = \frac{\phi}{\sum_{l=1}^{\infty} l^{-(2\nu+1)} \sin^2\left((l-1/2)\pi/2\right)} n_1^{-(2\nu+1)}. \tag{2.5}$$

Realizations are p times continuously differentiable if $\nu > p$. The length parameters L_x and L_y, though they are not visible in (2.5), play a role analogous to $1/\alpha$ in the Matèrn autocovariance function. The prior variance of $f(L_x/2, L_y/2)$ is ϕ^2.

Practically, of course, the series must be truncated. The largest index N_1 should be set so that the truncation has little effect; we set it so that $N_1^{-(2\nu+1)} \leq .0001$. While draws from the truncated series are technically infinitely differentiable, they retain a rough character for small values of ν. Purists have the option of using enough terms to capture oscillations on the lengthscale of the mesh spacing they will use to display the results.

Stein obtains the spectral representation of the Matèrn covariance function by treating it as an autocovariance function. Obtaining a spectral covariance matrix analogous to K_{n_1} from the Matèrn autocovariance function would require rewriting it as a covariance function and then integrating it against the Fourier basis in both its arguments over $[0, L_x] \times [0, L_y]$. The relationship between K_{n_1} and the spectral representation of the Matèrn covariance function given by Stein is not direct, but they are both constructed with the same purpose in mind. We were able to obtain a closed expression for $C(\bar{x}, \bar{\xi})$, but it is quite complicated. We have plotted it and some realizations in Figure 2.2.

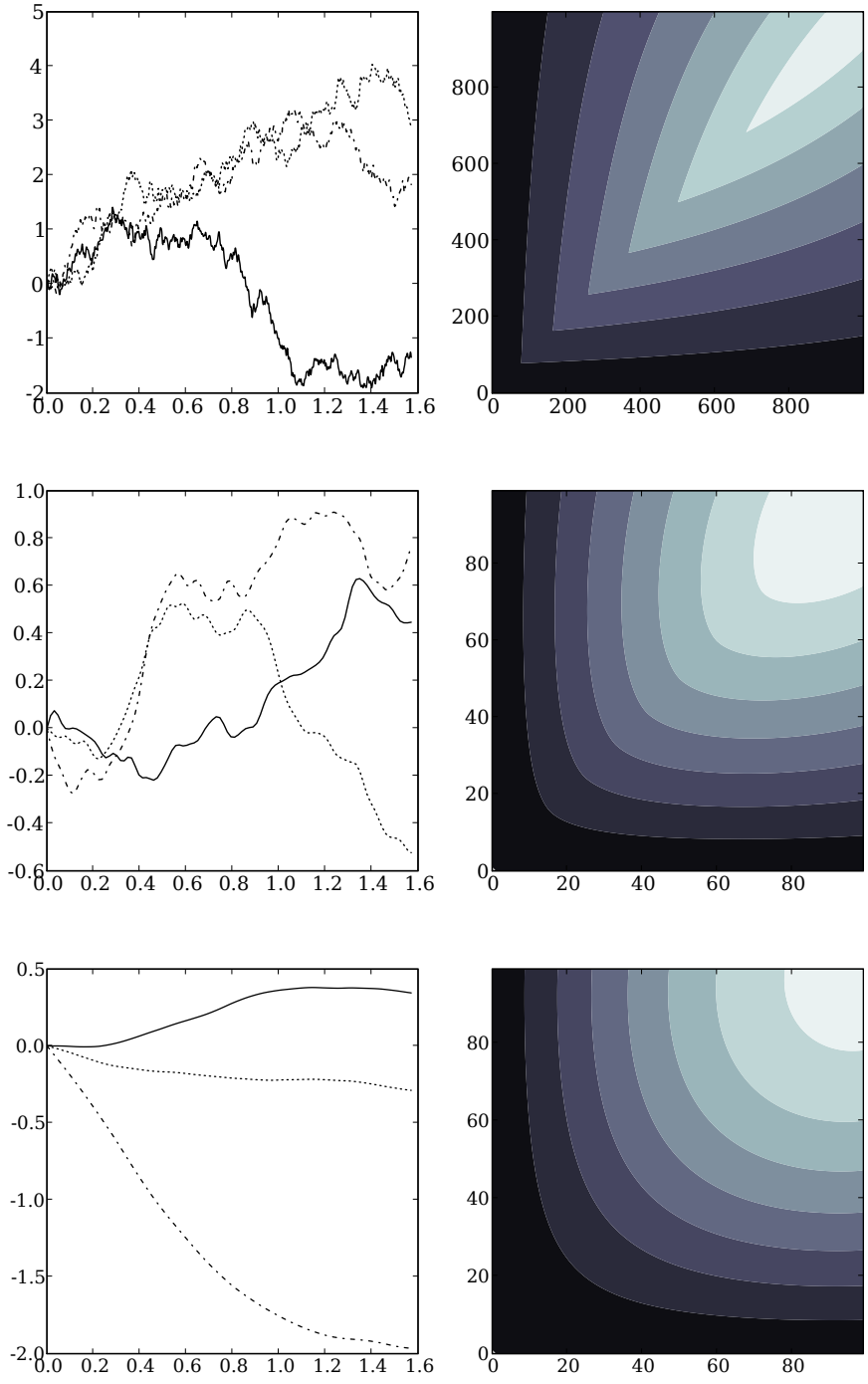


Figure 2.2: Covariances for f generated from the Fourier-space diagonal covariance matrix $K_{n_1} = \frac{\phi}{\sum_{l=1}^{\infty} l^{-(2\nu+1)} \sin^2((l-1/2)\pi/2)} n_1^{-(2\nu+1)}$ for $\phi = 1$, $\alpha = 2$, $\nu = [0.5, 1, 2]$, $L_x = 1.6$ and three realizations from each. Realizations are p times continuously differentiable if $\nu \geq p$.

r_x	m	β
2	250	.45

Table 2.1: Point estimates for the dynamical parameters r_x, m and β obtained from supporting experiments of Veilleux.

This representation for f requires storage and manipulation of only N_1^2 Fourier coefficients with an independent prior, yet provides flexibility in local surface behavior that is similar to that provided by the Matèrn covariance function. We set $f_{\text{guess}} = 0$, primarily because preliminary analysis indicates that the coefficient of a bilinear f_{guess} tends to be colinear with the first Fourier coefficient.

2.2.5 Priors for scalar parameters

Veilleux provides point estimates and confidence intervals for $r_x m$, r_x, and $1/\beta$ for each of his experimental setups [89]. Rough point estimates for r_x, m, and β derived from these are summarized in table 2.1 for the dataset we consider (corresponding to figure 8 in his paper), which differ in the concentration of bacterial nutrient available to the *Paramecium*. We set the shape parameter of the prior of each of these parameters to 10, and set the scale parameter so that the prior expectation is equal to the point estimate given above. Veilleux does not consider $-r_y$, so we assign it a similarly constructed prior with point estimate -1.

We place priors on the other scalar parameters as follows:

$$\begin{aligned} \tfrac{1}{\phi^2} &\sim \text{Gamma}(10, 100) \\ \sigma_x &\sim \text{Gamma}(10, 160) \\ \sigma_y &\sim \text{Gamma}(10, 80) \end{aligned} \tag{2.6}$$

We set $L_x = 1.25 \max_i x_i$, $L_y = 1.25 \max_i y_i$, and $\nu = 4$. We assign improper uniform priors to the initial condition x_0 and y_0; these parameters are constrained by the likelihoods of the initial observations, and it is difficult to assign informative priors to them before making observations.

2.3 Markov Chain Monte Carlo algorithm

We seek a model fitting method that satisfies two criteria:

1. Sampling of f is efficient without sacrificing flexibility.

2. Computational effort of evaluating transition densities increases polynomially in the number of species considered, to the extent possible.

3. Observations are not required to be particularly frequent in time.

Criterion 1 has been satisfied by our choice of covariance function for f. As we discuss later in this section, it makes available a very efficient Gibbs sampler for the Fourier coefficients. We adopt criterion 2 in hopes of opening the door to statistical inversion of multi-species stochastic dynamical systems. The caveat 'to the extent possible' acknowledges that encoding and manipulating an arbitrary function of n variables is

necessarily NP-hard in n; very high-dimensional systems will have to be parametrized using lower-dimensional unknown functions.

The primary difficulty in satisfying criterion 2 is handling the hidden trajectory $\mathbf{x}$. This requires either computing the transition density for $\mathbf{x}$ between adjacent observation times or imputing the 'unobserved' values of the trajectory and sampling them as a parameter. Most methods in the literature take the first option. Although none of them satisfies property 2, they may be appropriate for low-dimensional problems. We briefly review them in the next subsection.

An important note regarding condition 2 is that the deterministic skeletons of systems with three or more interacting species may exhibit chaotic dynamics [87]. The effects of this phenomenon on the accuracy and precision of inferences, as well as mixing properties of MCMC, are important topics for future research.

2.3.1 A review of options

A number methods have been developed in recent years for calculating $p(\mathbf{x}_{t_{i+1}}|\mathbf{x}_{t_i})$ between unobserved population sizes $\mathbf{x}_{t_i}$ at times corresponding to observations. We discuss four candidates in terms of the canonical stochastic differential equation with scalar diffusion coefficient

$$dz = a(z, \theta)dt + \sigma dW, \tag{2.7}$$

where z may be a scalar or a vector. Note that this canonical stochastic differential equation does not necessarily bear any relationship to the predator-prey model 2.4, and for now we are not interpreting it biologically.

1. The most obvious method is conversion of the stochastic differential equation to a partial differential equation in the transition density:

$$dz = a(z,\theta)dt + \sigma dW$$
$$\Rightarrow \frac{\partial p(z_{t+\tau}|z_t)}{\partial \tau} = \nabla^T(p(z_{t+\tau}|z_t)a(z,\theta)) + \frac{1}{2}\nabla^T\sigma\sigma^t\nabla p(z_{t+\tau}|z_t),$$

but preliminary analysis indicates that solving this equation directly is very expensive, especially in more than one spatial dimension.

2. Nicolau [69] uses an expression for the transition density of a scalar stochastic differential equation derived by Dacunha-Castelle and Florens-Zmirou [20] in an MCMC context. For the scalar version of (2.7), the transition density is

$$p(z_{t+\Delta t}|z_t) = \frac{1}{2\pi\sigma^2\Delta t}\exp\left\{-\frac{(z_t - z_{t+\Delta t})^2}{2\sigma^2\Delta t} + \frac{G(z_{t+\Delta t}) - G(z_t)}{\sigma^2}\right\}\mathrm{E}[\psi]$$

where

$$\psi = \exp\left\{\sigma^2\Delta t\int_0^1 h(K_u(z_t, z_{t+\Delta t}) + \sqrt{\sigma^2\Delta t}B_u)du\right\},$$
$$G(w) = \int_0^w a(u,\theta)du,$$
$$K_u(v,w) = v(1-u) + wu,$$
$$B_t = W_t - tW_1, 0 \le t \le 1,$$
$$h(w) = -\frac{1}{2}\left(\frac{a(w,\theta)^2}{\sigma^4} + \frac{1}{\sigma^2}\frac{da(w,\theta)}{dw}\right).$$

While this is an excellent way to calculate transition probabilities for scalar diffusions, it is not straightforward to generalize it to higher dimensions because the

integral associated with $G(w)$ becomes path-dependent unless a happens to be conservative.

3. A less refined expression, obtained by retracing a few steps in the derivation, can be evaluated in higher dimensions. For the vector version of (2.7), with $D = \sigma\sigma^T$,

$$p(z_{t+\Delta t}|z_t) = \frac{1}{2\pi|D|\Delta t}\exp\left\{-\frac{1}{2\Delta t}(z_t - z_{t+\Delta t})^T D^{-1}(z_t - z_{t+\Delta t})\right\}$$
$$\cdot \mathrm{E}\left\{\exp\left[\int_{\tau=t}^{t+\Delta t} D^{-1}a(w_\tau,\theta)^T dw_\tau - \frac{1}{2}\int_{\tau=t}^{t+\Delta t} D^{-1}a(w_\tau,\theta)^T a(w_\tau,\theta)d\tau\right]\right\}$$

where w satisfies the stochastic differential equation for z with initial condition z_t but is conditioned on $w_{t+\Delta t} = z_{t+\Delta t}$. Beskos *et al.* [13] present a method along these lines, which was published after the analysis contained in this chapter was complete.

4. Particle methods are attractive alternatives. Milstein *et al.* [61] present a particle method involving both forward- and reverse- time simulations from the stochastic differential equation. For (2.7) and a function h,

$$\mathrm{E}[h(z_t)|z_{t+\Delta t}] = \mathrm{E}[h(z^*_{\Delta t})|z^*_0]\mathcal{Z}_{\Delta t},$$

where the time-reversed process $(z^*, \mathcal{Z})$ is governed by

$$dz^* = -a(z^*,\theta)dz^* + \sigma dW^*, \quad z^*_0 = z_{t+\Delta t}$$

$$d\mathcal{Z} = -\nabla\cdot a(z^*)\mathcal{Z}dt, \qquad \mathcal{Z}_0 = 1.$$

The main idea behind their estimator for $p(z_{t+\Delta t}|z_t)$ is that the functional

$$J[f] = \mathbb{E}[f(z_\tau, z^*_{\Delta t-(\tau-t)})|z_t, z_{t+\Delta t}]$$

converges to the target transition density as f approaches the Dirac delta function, in the sense of distributions [85]. By choosing an f that is 'close' to the delta function, the authors obtain an estimate with much better properties than standard particle density estimates.

While it would undoubtedly prove more than satisfactory for the dataset we consider, we choose not to use this method because the number of particles needed to 'fill out' the transition density still increases exponentially with the number of species in the system, violating our criterion 2. Consider a trivial SDE and its time-reversed representation

$$dz = dW$$

$$dz^* = dW^*, \quad z_0^* = z_{t+\Delta t}$$

$$\mathcal{Z} = 1.$$

The associated transition probabilities are

$$z_\tau | z_t \sim \mathrm{N}(z_t, (\tau - t)I)$$

$$z^*_{\Delta t-(\tau-t)} | z_0^* \sim \mathrm{N}(z_t^*, [\Delta t - (\tau - t)]I).$$

Using the function

$$f(x, y) = \frac{1}{l^d} \prod_{i=1}^{d} 1_{|x_i - y_i| \leq l/2}$$

for a small edge length l, the probability that $f(z_\tau, z^*_{\Delta t-(\tau-t)}) = 1/l^d$ for a single particle pair is proportional to l^d. Assuming approximate independence between

particle pairs when the number N of particles is large, the estimate of $p(z_{t+\Delta t}|z_t)$ will have binomial variance and mean that scale as $N^2 l^d$ and a standard-deviation-to-mean ratio that scales as $l^{-d/2}N^{-1}$. The number of particles N required to maintain a specified accuracy increases exponentially with d.

5. We opt for a variation on the missing data method of Roberts and Stramer [79], who impute the unobserved trajectory at small temporal increments, and approximate the transition densities over these increments using the Euler-Maruyama discretization [62].

   This method passes criterion 2 in terms of the computational expense of computing the log-probability of a trajectory, but has several drawbacks:

   - Even for our two-dimensional model mixing is very slow (we found that 30m iterations were required to produce acceptable posteriors). The reason for the poor mixing is σ_x, σ_y, and $\mathbf{x}$ are highly dependent with respect to the likelihood, and the method makes it difficult to propose these parameters jointly.
   - The posteriors of the diffusion parameters are biased. This bias should vanish as mesh spacing is decreased, but in our case the bias was appreciable even at the highest resolution we could afford computationally.
   - The rescaling is only possible for some state-dependent drift coefficients σ. Aït-Sahalia [6] presents a condition.

   These drawbacks are discussed at greater length in the following sections.

For equation (2.7) with $D = \sigma\sigma^T$, the Euler-Maruyama approximation gives

$$\Delta z_t \approx a(z_t, \theta)\Delta t + \sigma \Delta W_t,$$

$$p_\Delta(z_{t+\Delta t}|z_t) \approx \mathbf{N}(\Delta z_t; a(z_t, \theta)\Delta t, D\Delta t).$$

As pointed out by Roberts and Stramer [79], simple-minded proposal distributions for the time-series z on a Δt-spaced mesh will result in disaster. Consider the log-probability of a sample path with z scalar,

$$\log p\left(z_0 \ldots z_T, \frac{1}{D}\right) = \log p\left(\frac{1}{D}\right) + \log p(z_0) + \frac{T}{2\Delta t}\log\left(\frac{1}{D}\right) - \frac{1/D}{2\Delta t}\sum_{i=0}^{T/\Delta t}(\Delta z_i - a(z_i)\Delta t)^2$$

plus constants. Since Δz is of order $\sqrt{\Delta t}$, the term $a(z_i)\Delta t$ can be neglected at leading order. The distribution of $\frac{1}{D}$ conditional on the sample path is approximately Gamma with shape parameter

$$\frac{T}{2\Delta t}$$

and scale parameter

$$\frac{\sum_{i=0}^{T/\Delta t}\Delta z_i^2}{2\Delta t}.$$

This distribution has mean

$$\frac{T}{\sum_{i=0}^{T/\Delta t}\Delta z_i^2}$$

and variance

$$\frac{2T\Delta t}{(\sum_{i=0}^{T/\Delta t}\Delta z_i^2)^2},$$

which means the posterior shrinks to a point mass as $\Delta t \to 0$. Conversely, conditional on $1/D$ the distribution of the quadratic variantion of the approximate process $z_0 \dots z_t$ is

$$\left(\frac{\sum_{i=0}^{T/\Delta t} \Delta z_i^2}{\Delta t} \right) \sim \frac{D\Delta t}{T} \chi^2_{T/\Delta t},$$

which has mean D and variance $2D^2\Delta t/T$. To paraphrase Roberts and Stramer, the conditional distribution of the quadratic variation approaches a point mass centered at a value that will only confirm the current value of D.

For another point of view, consider the $T/\Delta t$ values of Δz_i as iid draws from a normal distribution with mean zero and variance $D\Delta t$. The length of the $T/\Delta t$-dimensional vector whose components are given by the Δz_i's is proportional to

$$p(r) \propto r^{\frac{T}{\Delta t}-1} \exp(-D\Delta t^2 r^2/2).$$

This distribution approaches a point mass at $r = DT$ as Δt approaches zero. The joint distribution of the Δz_i's is not actually excluding smoother sample paths (corresponding to small r's); in fact, the density of $\Delta z_i = 0,\ \forall i$ approaches infinity as $\Delta t \to 0$. However, the ratio of the volumes of two balls of different radii increases exponentially in the number of spatial dimensions. The balance between the relatively smaller volume and relatively higher probability density associated with small values of r becomes so precise as $\Delta t \to 0$ that essentially all samples from the joint distribution of the Δz's will yield values of r very near the ring $r = DT$.

This all means that Gibbs sampling of σ and z is reducible in the limit $\Delta t \to 0$, and the mixing of the Markov chain can be made arbitrarily slow by choosing an appropriately small Δt. Since the error of the Euler-Maruyama approximation increases with Δt, we are in a predicament.

2.3.2 A variation on the method of Roberts and Stramer

Return to equation (2.3). Following Roberts and Stramer, we begin to address the dilemma of the last section by left-multiplying by σ^{-1} and setting $\tilde{\mathbf{x}} = \sigma^{-1}\mathbf{x}$:

$$d\tilde{\mathbf{x}} = \left\{ \sigma^{-1} \begin{bmatrix} r_x & 0 \\ 0 & r_y \end{bmatrix} \sigma\tilde{\mathbf{x}} + \sigma^{-1}\tilde{\mathbf{x}}^T\sigma^T \begin{bmatrix} -1/m & 0 \\ 0 & 0 \end{bmatrix} \sigma\tilde{\mathbf{x}} + \sigma^{-1} \begin{bmatrix} -1 \\ \beta \end{bmatrix} f(\sigma\tilde{\mathbf{x}}) \right\} dt + d\mathbf{W}.$$

Exploiting the fact that our σ is diagonal, we can write

$$d\tilde{\mathbf{x}} = \left\{ \begin{bmatrix} r_x & 0 \\ 0 & r_y \end{bmatrix} \tilde{\mathbf{x}} + \tilde{\mathbf{x}}^T \begin{bmatrix} -1/\tilde{m} & 0 \\ 0 & 0 \end{bmatrix} \tilde{\mathbf{x}} + \begin{bmatrix} -1 \\ \tilde{\beta} \end{bmatrix} \tilde{f}(\sigma\tilde{\mathbf{x}}) \right\} dt + d\mathbf{W}, \qquad (2.8)$$

where the scaled parameters are

$$\tilde{m} = m/\sigma_x$$

$$\tilde{\beta} = \beta\sigma_x/\sigma_y$$

$$\tilde{f} = f/\sigma_x \Rightarrow \tilde{c}_{nm} = c_{nm}/\sigma_x.$$

The likelihood (2.2) also changes somewhat:

$$\begin{bmatrix} \mathrm{x}_i \\ \mathrm{y}_i \end{bmatrix} \overset{iid}{\sim} \mathrm{N}(\sigma\tilde{\mathbf{x}}_{t_i}, V_i). \qquad (2.9)$$

If we work with $\tilde{\mathbf{x}}$ directly within the MCMC loop and reconstruct $\mathbf{x}$ for display afterward, this rescaling places σ in the drift terms and likelihood rather than in the diffusion term, so that good mixing is possible even with Δt small. This rescaling is not possible for all models [6].

Roberts and Stramer provide a proposal distribution for the discretization of the hidden process $\tilde{\mathbf{x}}$ that ensures that the quadratic variation $\frac{1}{T}\int_0^T d\tilde{\mathbf{x}}_t d\tilde{\mathbf{x}}_t^T$ is close to the identity matrix. Their algorithm proposes $\tilde{\mathbf{x}}$ by first proposing $\mathbf{x}_{t_i}$, then proposing the unobserved hidden process between observation points as either a Brownian bridge or a conditioned Ornstein-Uhlenbeck process. This algorithm contains no mechanism for tuning the 'jump size,' and its proposals will be rejected quite often for nonlinear systems with widely-spaced observation times. In the interest of satisfying our criterion 3, we make a minor modification to the algorithm and develop a random-walk proposal that preserves the quadratic variation of $\mathbf{x}$.

We propose $\tilde{\mathbf{x}}$ for the Metropolis-Hastings algorithm as follows.

1. For each observation index $i \in [0, N_o]$, propose $\tilde{\mathbf{x}}_{t_i}^p$ as follows: $\tilde{\mathbf{x}}_{t_i}^p \sim \mathrm{N}(\tilde{\mathbf{x}}_{t_i}, \epsilon^2)$, where ϵ is small.

2. Define

$$\tilde{\mathbf{x}}_{mis,j} = \{\tilde{\mathbf{x}}_{t_j} \dots \tilde{\mathbf{x}}_{t_{j+1}}\}.$$

   The subscript 'mis' is for 'missing data.' Propose $\tilde{\mathbf{x}}_{mis,j}^p$ for all $j \in (\max(0, i-1), \min(i, N-1))$ as follows:

$$\tilde{\mathbf{x}}_{mis,j}^p = a\tilde{\mathbf{x}}_{mis,j} + bZ,$$

where b is small and $a^2 + b^2 = 1$, so that the sum of squared deviations $1/(t_{j+1} - t_j) \int_{t_j}^{t_{j+1}} d\tilde{\mathbf{x}}_t d\tilde{\mathbf{x}}_t^T$ remains close to the identity matrix. Z is a Brownian bridge: a Brownian motion on $[t_j, t_{j+1}]$ whose diffusion coefficient is the identity matrix conditioned on

$$Z_{t_j} = \frac{\tilde{\mathbf{x}}_{t_j}^p}{b} - \frac{a\tilde{\mathbf{x}}_{t_j}}{b},$$
$$Z_{t_{j+1}} = \frac{\tilde{\mathbf{x}}_{t_{j+1}}^p}{b} - \frac{a\tilde{\mathbf{x}}_{t_{j+1}}}{b},$$

so that $\tilde{\mathbf{x}}_{mis,j}^p$ at time t_j is equal to $\tilde{\mathbf{x}}_{t_j}^p$ and likewise for t_{j+1}.

Discretized Brownian bridges can be easily constructed in several ways; see for example Milstein and Tretyakov [61]. For Δt-spaced discretizations of $\tilde{\mathbf{x}}$, after many accepted proposals, the quadratic variation of the proposed hidden process in each dimension is iid with distribution $\frac{\Delta t}{T}\chi^2_{t/\Delta t}$. The quadratic variation of the sequence of proposals can be shown to be ergodic in the sense that the quadratic variation of the initial state is eventually forgotten, assuming no bias in acceptance.

3. Let $t_{min,i} = t_{\max(0,i-1)}$ and $t_{max,i} = t_{\min(i,N-1)}$, and let $\tilde{\mathbf{x}}_{mis,i}^p$ collect $\tilde{\mathbf{x}}_{mis,j}^p$ for all integers $j \in [t_{min,i}, t_{max,i}]$. According to the Metropolis-Hastings algorithm, we accept the jump to $\tilde{\mathbf{x}}_{t_i}^p$ and $\tilde{\mathbf{x}}_{mis,i}^p$ if

$$\frac{G(\tilde{\mathbf{x}}_{t_i}^p, \tilde{\mathbf{x}}_{mis,i}^p) Q(\tilde{\mathbf{x}}_{mis,i} \leftarrow \tilde{\mathbf{x}}_{mis,i}^p)}{G(\tilde{\mathbf{x}}_{t_i}, \tilde{\mathbf{x}}_{mis,i}) Q(\tilde{\mathbf{x}}_{mis,i} \rightarrow \tilde{\mathbf{x}}_{mis,i}^p)} \geq \omega,$$
$$\omega \sim U(0,1),$$

and we reject it otherwise. Here $G(\tilde{\mathbf{x}}_{t_i}^p, \tilde{\mathbf{x}}_{mis,i}^p)$ is proportional to the density of $\tilde{\mathbf{x}}_{t_i}^p$ and $\tilde{\mathbf{x}}_{mis,i}^p$ with respect to the standard Weiner process conditional on all scalar

parameters and the function $\tilde{f}$; $Q(\tilde{\mathbf{x}}^p_{mis,i} \leftarrow \tilde{\mathbf{x}}_{mis,i})$ is the density of the jump to $\tilde{\mathbf{x}}^p_{mis,i}$ from $\tilde{\mathbf{x}}_{mis,i}$ with respect to the standard Weiner process; and the other two terms are analogously defined. Note that $\tilde{\mathbf{x}}^p_{t_i}$ and $\tilde{\mathbf{x}}_{t_i}$ do not come into the Hastings factors $Q(\cdot)$ because of the multivariate normal proposal distribution. Denote

$$\Xi_t = \left\{ \begin{bmatrix} r_y & 0 \\ 0 & r_y \end{bmatrix} \tilde{\mathbf{x}}_{mis,i,t} + \tilde{\mathbf{x}}^T_{mis,i,t} \begin{bmatrix} -1/\tilde{m} & 0 \\ 0 & 0 \end{bmatrix} \tilde{\mathbf{x}}_{mis,i,t} + \begin{bmatrix} -1 \\ \tilde{\beta} \end{bmatrix} \tilde{f}(\sigma \tilde{\mathbf{x}}_{mis,i,t}) \right\},$$

$$\Xi^p_t = \left\{ \begin{bmatrix} r_y & 0 \\ 0 & r_y \end{bmatrix} \tilde{\mathbf{x}}^p_{mis,i,t} + (\tilde{\mathbf{x}}^p_{mis,i,t})^T \begin{bmatrix} -1/\tilde{m} & 0 \\ 0 & 0 \end{bmatrix} \tilde{\mathbf{x}}^p_{mis,i,t} + \begin{bmatrix} -1 \\ \tilde{\beta} \end{bmatrix} \tilde{f}(\sigma \tilde{\mathbf{x}}^p_{mis,i,t}) \right\},$$

where the index t takes values between $t_{min,i}$ and $t_{max,i}$. Also define Z^* as draw for Z which would have been required to jump from $\tilde{\mathbf{x}}^p_{mis,i}$ back to $\tilde{\mathbf{x}}_{mis,i}$,

$$Z^* = \frac{\tilde{\mathbf{x}}_{mis,i} - a\tilde{\mathbf{x}}^p_{mis,i}}{b}.$$

From Girsanov's formula, we have up to irrelevant constants:

$$\log G(\mathbf{x}^p_{t_i}, \tilde{\mathbf{x}}^p_{mis,i}) = -1/2 \left(\mathbf{x}^p_{t_i} - \begin{bmatrix} \mathrm{x}_i \\ \\ \mathrm{y}_i \end{bmatrix} \right)^T \begin{bmatrix} \sigma^2_{o,x} & 0 \\ \\ 0 & \sigma^2_{o,y} \end{bmatrix} \left(\mathbf{x}^p_{t_i} - \begin{bmatrix} \mathrm{x}_i \\ \\ \mathrm{y}_i \end{bmatrix} \right) + \int_{t_{min,i}}^{t_{max,i}} \Xi^{pT}_t d\tilde{\mathbf{x}}^p_{mis,i,t} - \frac{1}{2} \int_{t_{min,i}}^{t_{max,i}} \Xi^{pT}_t \Xi^p_t dt$$

$$\log Q(\tilde{\mathbf{x}}^p_{mis,i} \leftarrow \tilde{\mathbf{x}}_{mis,i}) = \int_{t_{min,i}}^{t_{max,i}} \left[\frac{Z_t - (\frac{1}{b}\tilde{\mathbf{x}}^p_{t_{i+1}} - \frac{a}{b}\tilde{\mathbf{x}}_{t_{i+1}})}{t_{i+1} - t_i - t} \right]^T dZ_t - \frac{1}{2} \int_{t_{min,i}}^{t_{max,i}} \left[\frac{Z_t - (\frac{1}{b}\tilde{\mathbf{x}}^p_{t_{i+1}} - \frac{a}{b}\tilde{\mathbf{x}}_{t_{i+1}})}{t_{i+1} - t_i - t} \right]^T \left[\frac{Z_t - (\frac{1}{b}\tilde{\mathbf{x}}^p_{t_{i+1}} - \frac{a}{b}\tilde{\mathbf{x}}_{t_{i+1}})}{t_{i+1} - t_i - t} \right] dt$$

$$\log G(\mathbf{x}_{t_i}, \tilde{\mathbf{x}}_{mis,i}) = -1/2 \left(\mathbf{x}_{t_i} - \begin{bmatrix} \mathrm{x}_i \\ \\ \mathrm{y}_i \end{bmatrix} \right)^T \begin{bmatrix} \sigma^2_{o,x} & 0 \\ \\ 0 & \sigma^2_{o,y} \end{bmatrix} \left(\mathbf{x}_{t_i} - \begin{bmatrix} \mathrm{x}_i \\ \\ \mathrm{y}_i \end{bmatrix} \right) + \int_{t_{min,i}}^{t_{max,i}} {\Xi_t}^T d\tilde{\mathbf{x}}_{mis,i,t} - \frac{1}{2} \int_{t_{min,i}}^{t_{max,i}} {\Xi_t}^T \Xi_t dt$$

$$\log Q(\tilde{\mathbf{x}}^p_{mis,i} \rightarrow \tilde{\mathbf{x}}_{mis,i}) = \int_{t_{min,i}}^{t_{max,i}} \left[\frac{Z^*_t - (\frac{1}{b}\tilde{\mathbf{x}}_{t_{i+1}} - \frac{a}{b}\tilde{\mathbf{x}}^p_{t_{i+1}})}{t_{i+1} - t_i - t} \right]^T dZ^*_t - \frac{1}{2} \int_{t_{min,i}}^{t_{max,i}} \left[\frac{Z^*_t - (\frac{1}{b}\tilde{\mathbf{x}}_{t_{i+1}} - \frac{a}{b}\tilde{\mathbf{x}}^p_{t_{i+1}})}{t_{i+1} - t_i - t} \right]^T \left[\frac{Z^*_t - (\frac{1}{b}\tilde{\mathbf{x}}_{t_{i+1}} - \frac{a}{b}\tilde{\mathbf{x}}^p_{t_{i+1}})}{t_{i+1} - t_i - t} \right] dt,$$

Intuitively, $G(\mathbf{x}^p_{t_i}, \tilde{\mathbf{x}}^p_{mis,i})$ is proportional to the log-probability-density of the proposed trajectory, and $\log Q(\tilde{\mathbf{x}}^p_{mis,i} \leftarrow \tilde{\mathbf{x}}_{mis,i})$ is proportional to the log-probability-density of a jump from the current trajectory to the proposed trajectory. The

expressions for $\log Q(\cdot)$ come from the Markovian representation of the Brownian Bridge. For $i = 0$, it is necessary to add a term corresponding to the prior density of $(\tilde{x}_0, \tilde{y}_0)$ (see (2.6)) to the expressions for $\log G$, but we have used uninformative priors for these initial conditions.

2.3.3 Sampling the unknown functional response $\tilde{f}$ and dynamical parameters

Most of the scalar parameters do not have standard conditional distributions; however, the conditional likelihood for many of them is normal. In other words, if these parameters had uninformative priors, their full conditional distributions would be normal. Proposing from the conditional likelihood and accepting based on the prior yields a fast, well-mixing sampler, especially when the conditional likelihood is very informative compared to the prior. It is important to note that these likelihoods involve transformations (tilde'd versions) of the parameters; when evaluating the prior density of a proposed value, the transformations must be inverted.

The workhorse for these conditional likelihood proposals is as follows. As was recently pointed out by Elerian et al. [23], if for some (vector or scalar) parameter υ

$$d\tilde{x}_t = \upsilon\lambda_t - \Xi_t + dW_t$$

for all $t \in [0, T]$, then the conditional likelihood of υ is normal with variance

$$V = \left[\int_0^T \lambda_t^T \lambda_t dt\right]^{-1}$$

and mean

$$\mu = \left[\int_0^T \lambda_t^T \lambda_t dt\right]^{-1} \int_0^T \lambda_t^T (d\tilde{x}_t + \Xi_t dt).$$

Each of the Fourier coefficients of $\tilde{f}$, the predator and prey growth rates r_x and r_y, the scaled conversion efficiency $\tilde{\beta}$, and the scaled inverse carrying capacity $1/\tilde{m}$ can be sampled efficiently using this scheme.

2.3.4 Sampling the incremental standard deviations σ_x and σ_y

σ_x and σ_y must be sampled using the Metropolis-Hastings algorithm. Their full conditional distribution, up to constants not involving σ_x or σ_y, is

$$\begin{aligned}
\log p(\sigma_x, \sigma_x | b, \tilde{m}, \tilde{\beta}, \phi, c, \tilde{\mathbf{x}}, \mathrm{x}, \mathrm{y}) \propto & \int_0^T \Xi_t^T d\tilde{\mathbf{x}}_t - \frac{1}{2}\int_0^T \Xi_t^T \Xi_t dt \\
& - \frac{\sum_i \tilde{x}_{t_i}^2}{2\sigma_{o,x}^2}\left(\sigma_x - \frac{\sum_i \tilde{x}_{t_i}\mathrm{x}_i}{\tilde{x}_{t_i}^2}\right)^2 - \frac{\sum_i \tilde{y}_{t_i}^2}{2\sigma_{o,y}^2}\left(\sigma_y - \frac{\sum_i \tilde{y}_{t_i}\mathrm{y}_i}{\tilde{y}_{t_i}^2}\right)^2 \\
& + (-a_\beta + a_x + a_m)\log\sigma_x + (a_\beta - 1 + a_y)\log\sigma_y \\
& - \sigma_x(b_m\tilde{m} + b_x\tilde{x}_0) - b_\beta\tilde{\beta}\frac{\sigma_y}{\sigma_x} - b_y\sigma_y\tilde{y}_0 \\
& - N_f^2 \log\sigma_x - \sigma_x^2 \frac{1}{2}\sum_{n,m}\frac{\tilde{c}_{nm}^2}{K_n K_m}.
\end{aligned}$$

where

$$\Xi_t = \left\{ \begin{bmatrix} r_y & 0 \\ 0 & r_y \end{bmatrix} \tilde{\mathbf{x}}_t + \tilde{\mathbf{x}}_t^T \begin{bmatrix} -1/\tilde{m} & 0 \\ 0 & 0 \end{bmatrix} \tilde{\mathbf{x}}_t + \begin{bmatrix} -1 \\ \tilde{\beta} \end{bmatrix} \tilde{f}(\sigma\tilde{\mathbf{x}}_t) \right\}.$$

As stated earlier, σ_x and σ_y mix very poorly because they are highly dependent on $\tilde{\mathbf{x}}$ with respect to the observation likelihood, $p(\mathrm{x}, \mathrm{y}|\mathbf{x}, \sigma_x, \sigma_y)$. It is difficult to propose σ_x, σ_y and $\mathbf{x}$ together in a way that negotiates their joint support efficiently without

disturbing the quadratic variation of $\mathbf{x}$. However, when σ_x and σ_y are fixed a priori all other parameters, including $\tilde{\mathbf{x}}$ and $\tilde{f}$, mix quite well.

2.3.5 Sampling the overall covariance magnitude ϕ

Since the coefficients $\tilde{c}_{nm}$ are mutually independent with respect to their prior, the full conditional distribution of $1/\phi^2$ is Gamma with shape parameter

$$\frac{N_f^2}{2} + a_\phi$$

and scale parameter

$$b_\phi + \frac{1}{2}\sum_{n,m} \frac{\tilde{c}_{nm}^2 \sigma_x^2}{(K_n/\phi)(K_m/\phi)}.$$

Note that K_n/ϕ and K_m/ϕ do not depend on ϕ.

2.4 Results

2.4.1 Method validation using simulated data

Figure 2.3 shows the posterior distributions of the scalar parameters inferred using simulated data, as well as their true values. The true values tend to be in the dense supports of the posteriors. Figure 2.5 shows the fit to the simulated data superimposed with the true trajectory used to generate the data; the latter tends to stay within the 95% posterior probability interval. Figure 2.6 compares our inference of f with the function used to generate the simulated data (in this case, the asymmetric function $.04x_t y_t/(1 + x_t/100)$). Inference of f is quite good within the range of the data, in that

the posterior median is near the true value and the 95% posterior probability interval is narrow.

In spite of these successes, however, figure 2.4 shows that our inference of σ_x and σ_y is not accurate. The posterior modes for these parameters are too large by roughly a factor of two. A possible explanation for this is that the quadratic variation of discretized solution paths of equation 2.3 is significantly larger in magnitude than $\sigma\sigma^T$, and the Brownian bridge proposal distribution necessitates artificially inflating σ to provide a good fit. Fortunately, the diffusion parameters tend to be relatively uncorrelated with the drift parameters and the Fourier coefficients of f, which are the primary inferential targets. In practice, it may be preferable to estimate σ *a priori* and then fit the drift parameters and and the unobserved trajectories conditional on the estimated value.

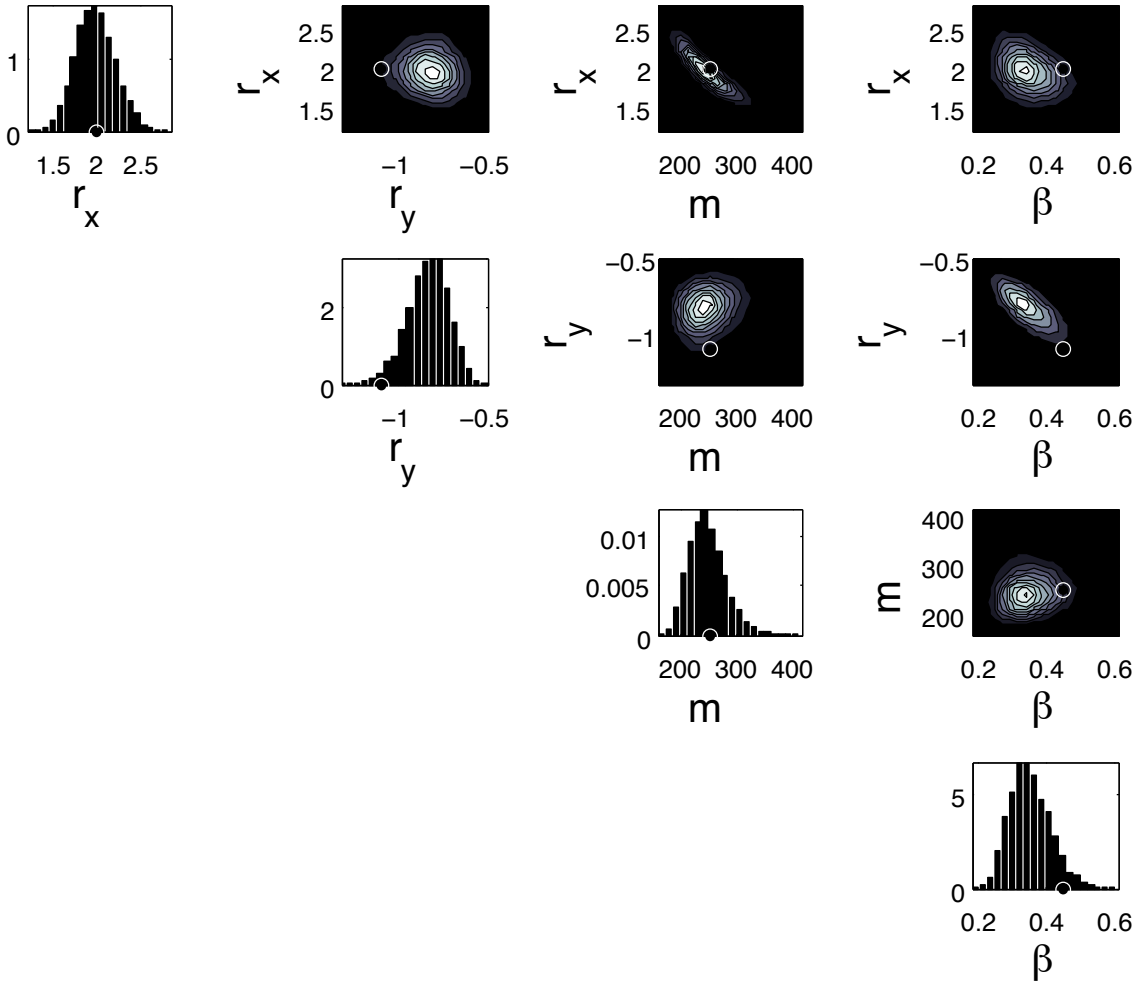


Figure 2.3: The posterior of r_x, r_y, β, m for simulated data. Correlated pairs are r_x and m; r_x and β; and r_y and β. Correct values are shown as circles

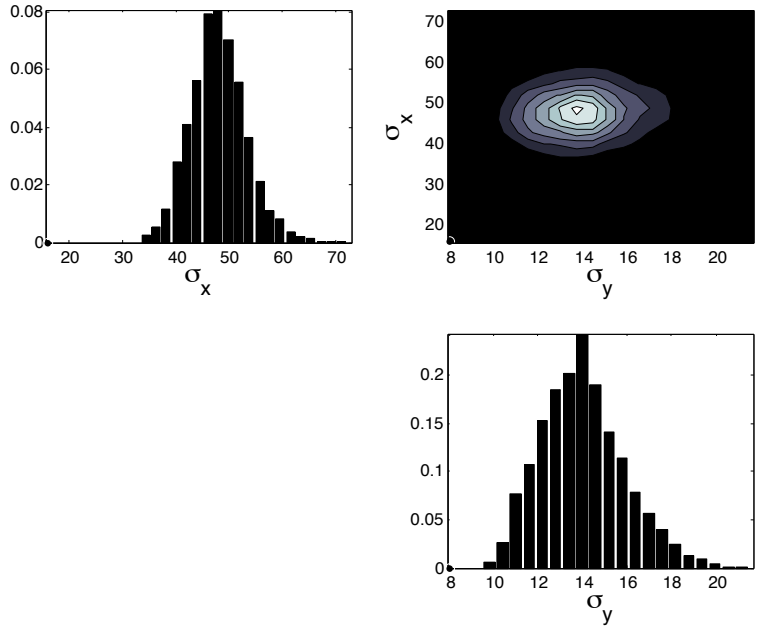


Figure 2.4: Posterior of σ_x and σ_y, simulated data. Circles (at the lower left corner of each plot) indicate correct values. The posteriors are biased high by roughly a factor of two.

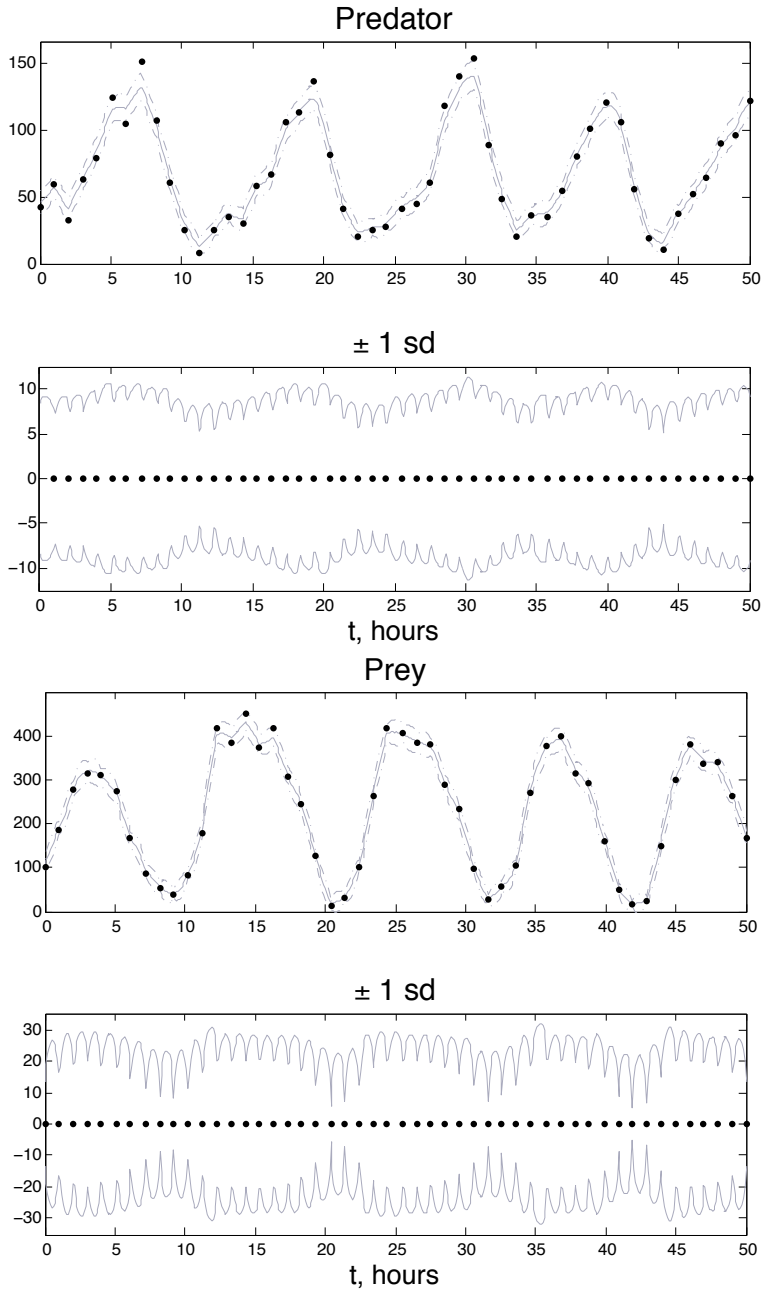


Figure 2.5: Posterior means of x and y with standard deviations for the simulated data. For means, datapoints are shown as black dots; for standard deviations, observation times are shown as black dots. True simulated trajectory used to simulated the data is shown as a dark grey line.

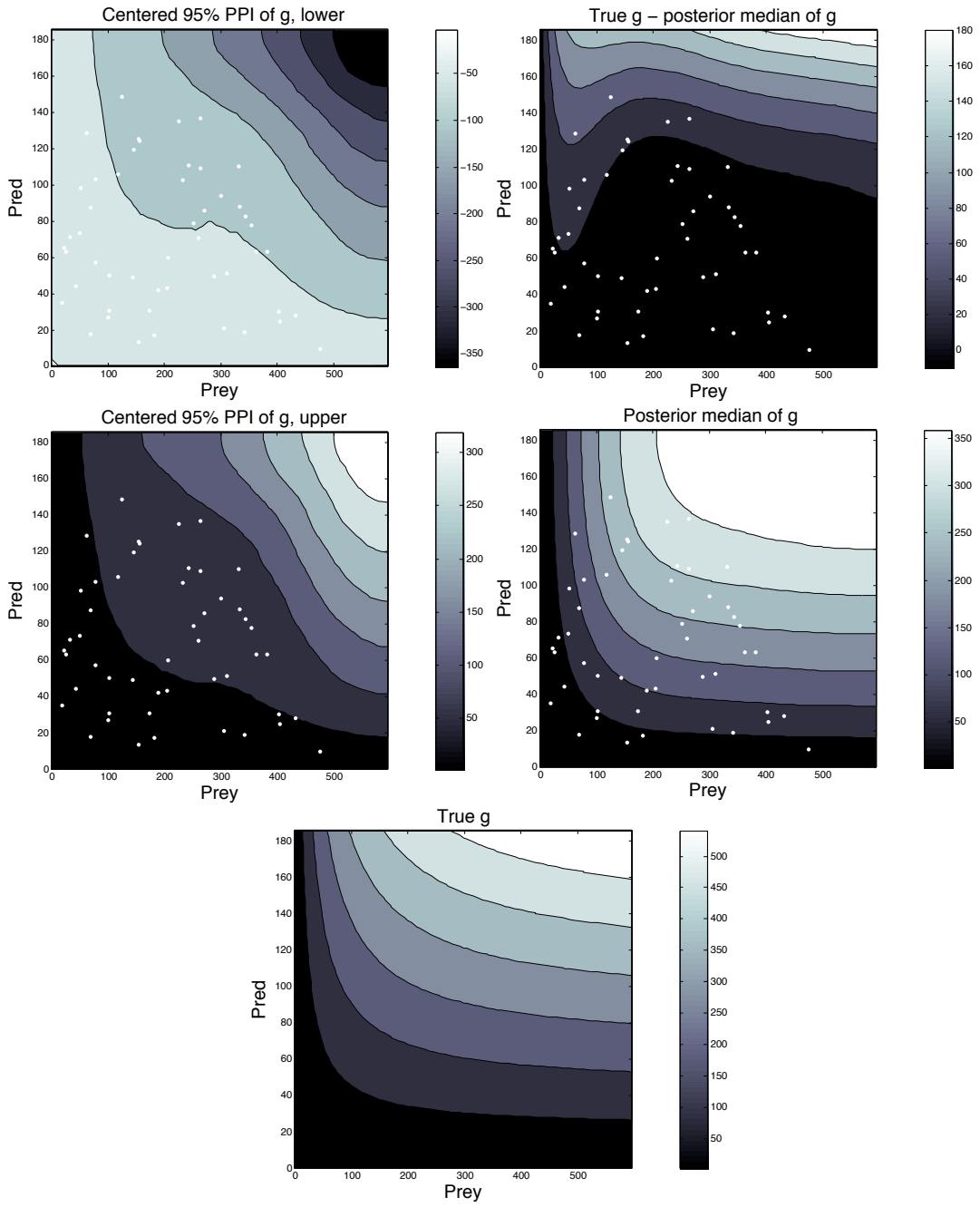


Figure 2.6: **Upper left and middle left:** The upper and lower widths of the centered 95% posterior probability envelope of f for the simulated data. **Upper right:** The difference between the posterior median and the true value. **Middle right:** The posterior median of f for the simulated data and the true value. Observations are shown as white dots. **Lower right:** The true value of f.

	estimate	prior median	prior 95% PI	posterior median	posterior 95% PI
σ_x	-	15.5	19.6	39.3	14.6
σ_y	-	7.68	9.88	12.5	7.03
r_x	2	1.94	2.41	2.27	.907
r_y	-	-1.07	1.34	-1.01	.412
m	250	241	305	322	267
β	.451	.439	.560	.276	.162

Table 2.2: The prior and posterior median and centered 95% probability interval for the dynamical and diffusion parameters, based on Veilleux's data.

2.4.2 Inferences based on Veilleux's data

Table 2.2 summarizes the prior and posterior distributions we found for the scalar parameters, as well as the estimates we found from Veilleux's dataset 1 in section 2.2.5. The posterior centered 95% probability interval was quite a bit smaller than the prior centered 95% probability interval for parameters r_x, r_y, and β. The interval for m did not shrink much from prior to posterior, possibly because the observed dynamics could have resulted if predation tended to pull large prey populations down before they felt the effects of carrying capacity. Note that the posterior median of m is comparable to the largest prey population size observed. The intervals for σ_x and σ_y also failed to shrink much, but note that the prior medians grew by roughly a factor of two.

Marginal and bivariate posteriors for the scalar parameters are shown in Figures 2.7 and 2.8. Notable posterior correlations can be seen between m and r_x; between β and r_x; and between r_y and β. Each of these makes some dynamical sense. For instance, if the predator population is observed to grow quickly it must either have a high value of r_y or a high value of β; but given a value of conversion efficiency β, it is

difficult to improve on an estimate of carrying capacity m. No notable correlations were observed between the drift parameters and the diffusion parameters.

The fit of $\mathbf{x}$ to the data is shown in Figure 2.9.

2.4.3 The functional response

Figure 2.10 shows the posterior distribution of the inferred functional response. The posterior median of the functional response appears to be approximately prey-dependent at low predator densities (the level curves are approximately vertical), but evinces higher degrees of ratio dependence at higher predator densities (if the functional response were purely ratio-dependent, the level curves would be rays from the origin). This result is highly suggestive, and further modeling work is warranted to elucidate the mechanisms underlying the apparent ratio dependence. As stated in the introduction, prey-sharing due to long handling time seems a likely candidate.

2.5 Dynamical insight

We now turn to the question of how to gain insight from the fit results beyond simple posterior distributions. We will proceed along the lines of deterministic dynamical modeling in ecology. In that field, once parameters are estimated the dynamics of the model are investigated by simulation and linear stability analysis. The dynamical insight thus derived is used to make qualitative inferences about the system. We will

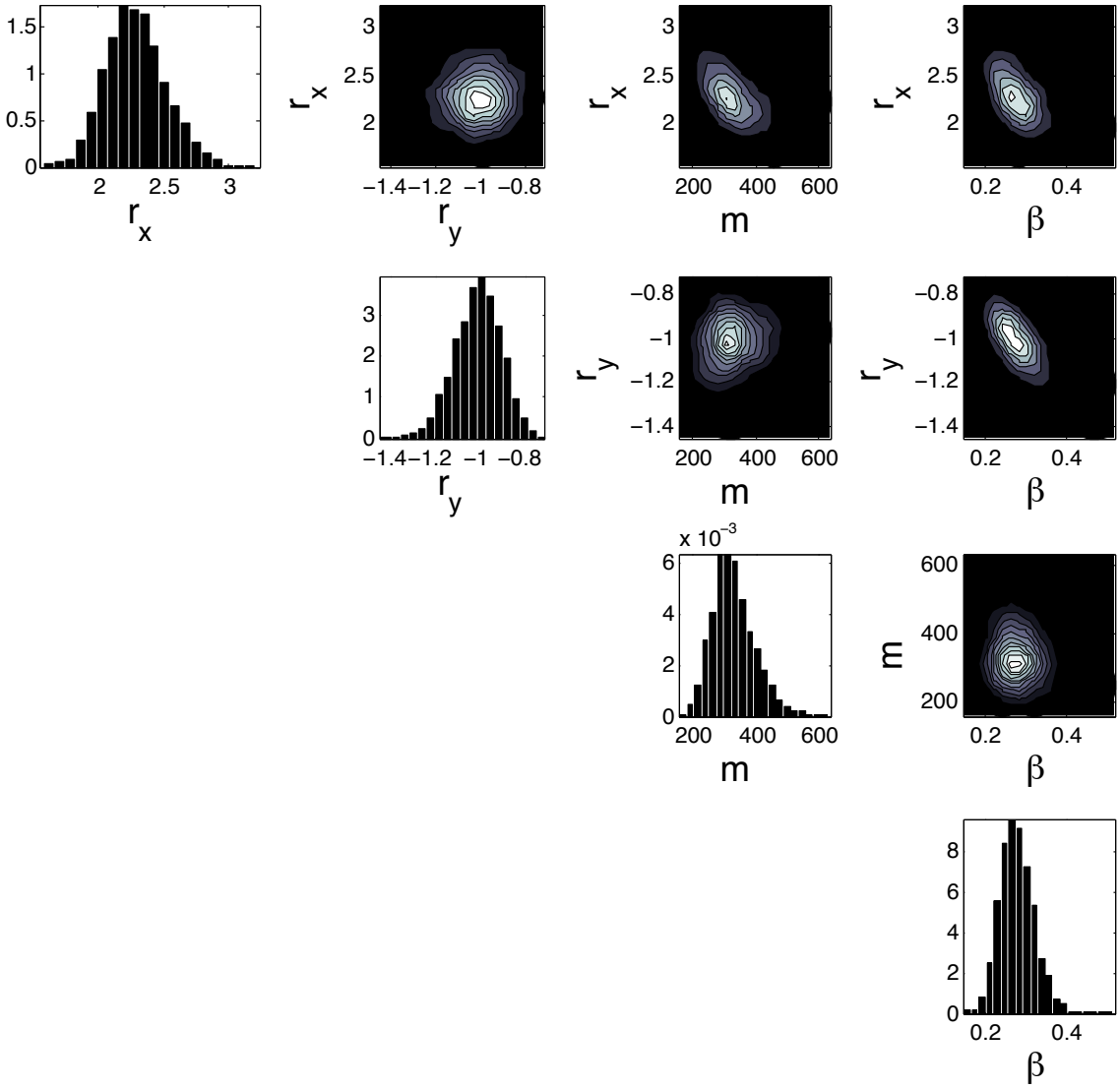


Figure 2.7: The posterior of r_x, r_y, β and m for Veilleux's experiment. Correlated pairs are r_x and m; r_x and β; and r_y and β.

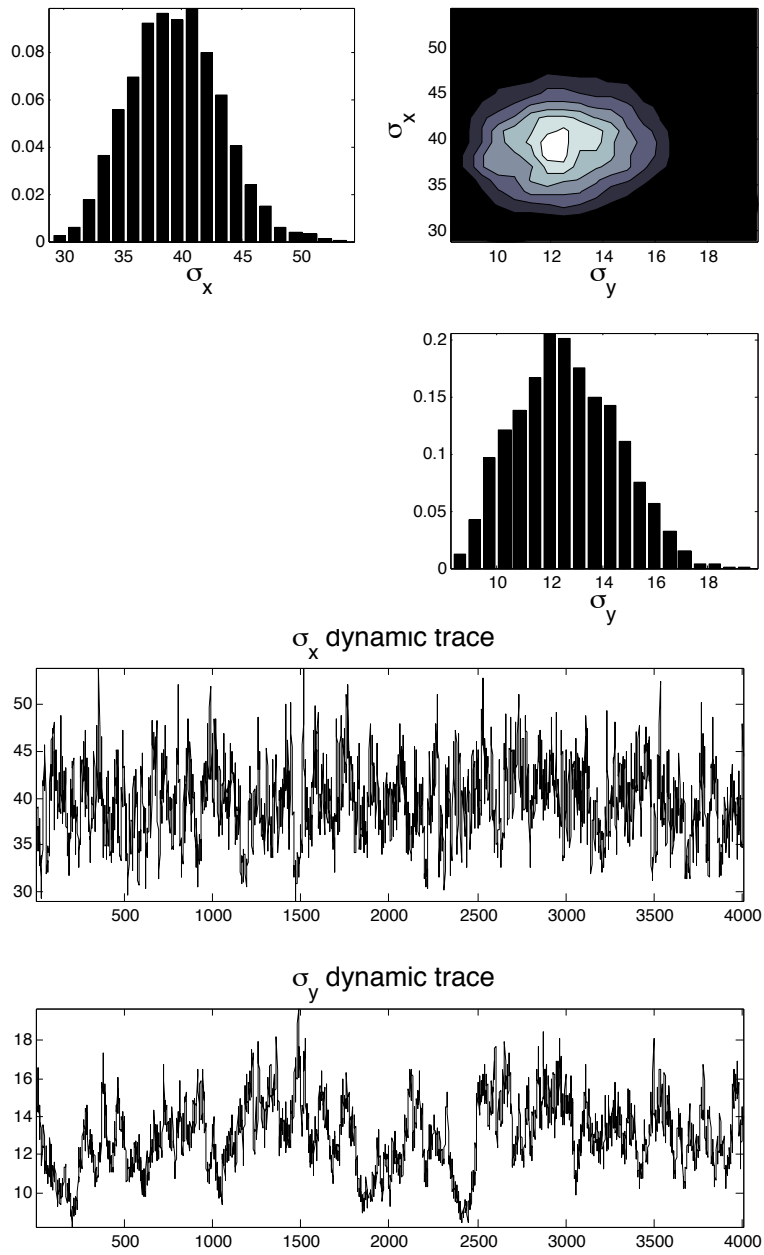


Figure 2.8: The posterior of σ_x and σ_y for Veilleux's experiment, with dynamic traces. 30 million MCMC iterations were used.

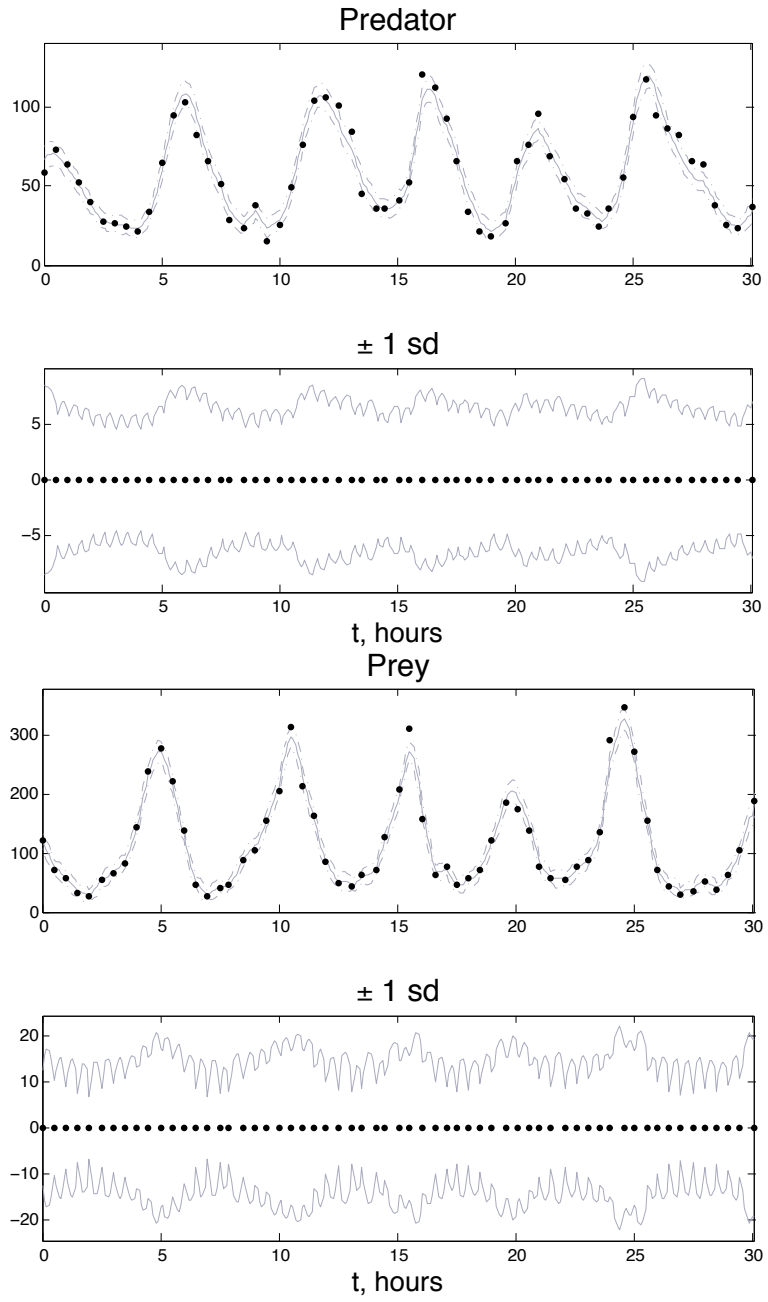


Figure 2.9: Posterior means of x and y with standard deviation envelopes for Veilleux's experiment. For means, datapoints are shown as black dots. For standard deviations, observation times are shown as black dots.

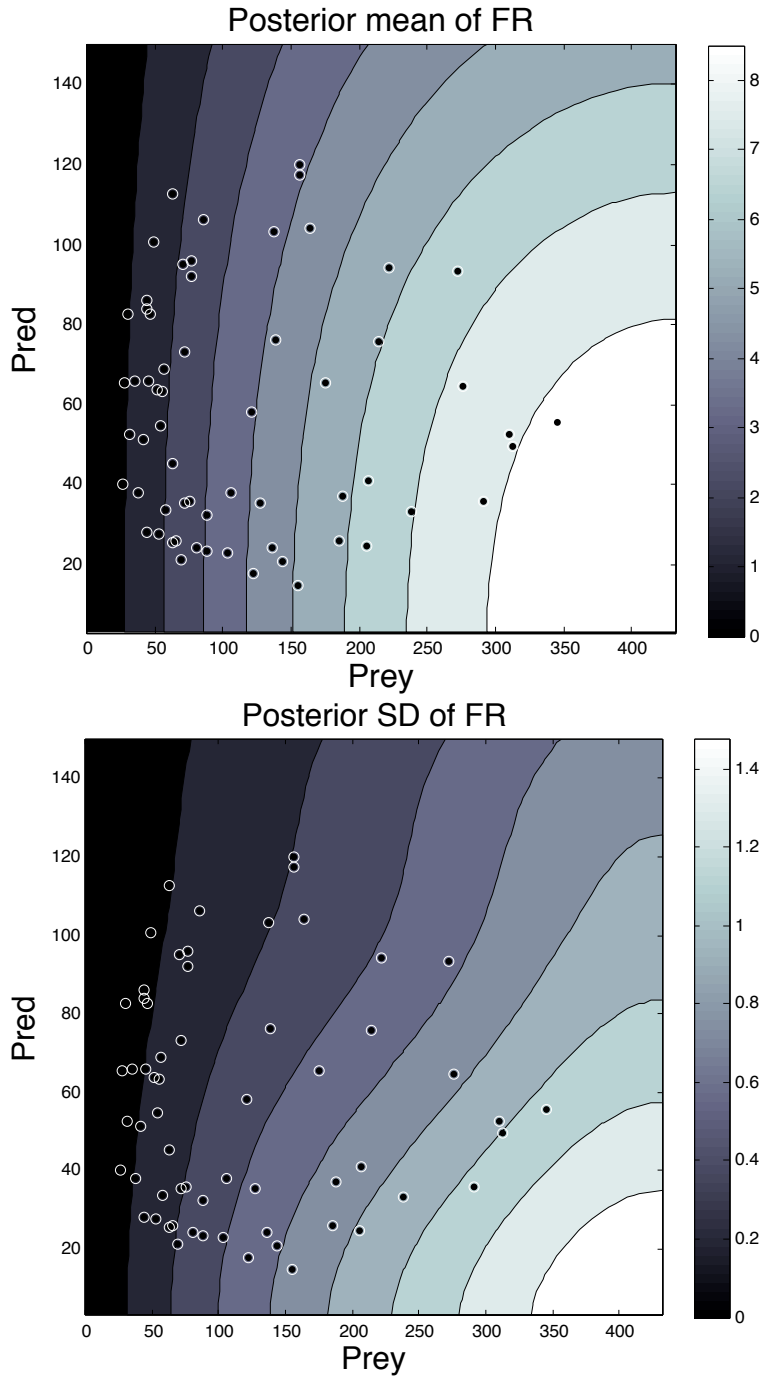


Figure 2.10: The posterior mean and standard deviation of the functional response for Veilleux's experiment. The posterior mean appears prey-dependent at low predator densities, but ratio dependence appears to increase with predator density.

begin by attempting to follow that strategy closely, then branch out to techniques that are better-suited to stochastic, uncertain contexts.

The deterministic skeleton of the SDE 2.3 is the dynamical system

$$\begin{aligned} \dot{x}_t &= r_x x_t - \tfrac{x_t^2}{m} - f(x_t, y_t) \\ \dot{y}_t &= r_y y_t + \beta f(x_t, y_t). \end{aligned} \tag{2.10}$$

Two-dimensional dynamical systems like this are usually characterized by their fixed points and limit cycles (often summarized along with flow lines in a phase portrait) because the Poincaré-Bendixson theorem guarantees that trajectories will either approach a stable fixed point, escape to infinity, or approach a limit cycle [87]. Fixed points (x_*, y_*) satisfy the equations

$$\begin{aligned} r_x x_* - \frac{x_*^2}{m} - f(x_*, y_*) &= 0 \\ r_y y_* + \beta f(x_*, y_*) &= 0. \end{aligned}$$

All solutions to these equations inhabit the parabola

$$y_* = \frac{\beta}{r_y}\left(\frac{x_*^2}{m} - r_x x_*\right),$$

so solving for fixed points is a one-dimensional root-finding problem. Stability of the fixed points is determined by the eigenvalues of the Hessian matrix

$$\begin{bmatrix} r_x - 2\frac{x_*}{m} - f_x(x_*, y_*) & -f_y(x_*, y_*) \\ \beta f_x(x_*, y_*) & r_y + \beta f_y(x_*, y_*) \end{bmatrix}.$$

Since Veilleux's data are oscillatory, we might expect there to be an unstable spiral point enclosed in a limit cycle [87]. However, all of our samples had one stable

spiral point. Figure 2.11 shows the intensity surface [63] of stable spiral points and the posterior distribution of the most unstable eigenvalue given that a stable spiral point is present. Interestingly, the real and imaginary components of the eigenvalue are negatively correlated. If approach to the spiral point is relatively slow, oscillations will be slower also.

Figure 2.12 shows a stochastic trajectory in one sample from the posterior of the flow field. It is clear from this figure how oscillations can persist indefinitely even in the presence of a stable spiral point: the trajectory does indeed tend to wind around the spiral point, but sometimes moves away from it. Figure 2.12 shows three samples from the posterior of the flow field, along with error ellipses of a trajectory at times 1,2,3,4,5 after it starts. Rather than approaching the spiral point, the trajectories are settling down to a stationary distribution. The deterministic theory of dynamical systems has valuable insight to offer, but it does not appear to be quite the right tool.

Figure 2.13 shows one way to visualize the range of stochastic dynamics encoded by the posterior distribution. The arrowheads in the flow field are replaced by error ellipses indicating the posterior variation in the range of flow vectors. The sizes of the error ellipses tend to be small in comparison with the mean vectors, which is not surprising. We should be able to get a good idea of the dynamics from Veilleux's dataset, even if it is difficult to precisely parse out the values individual parameters. The role of fixed points and limit cycles is played by the subjective stationary distribution

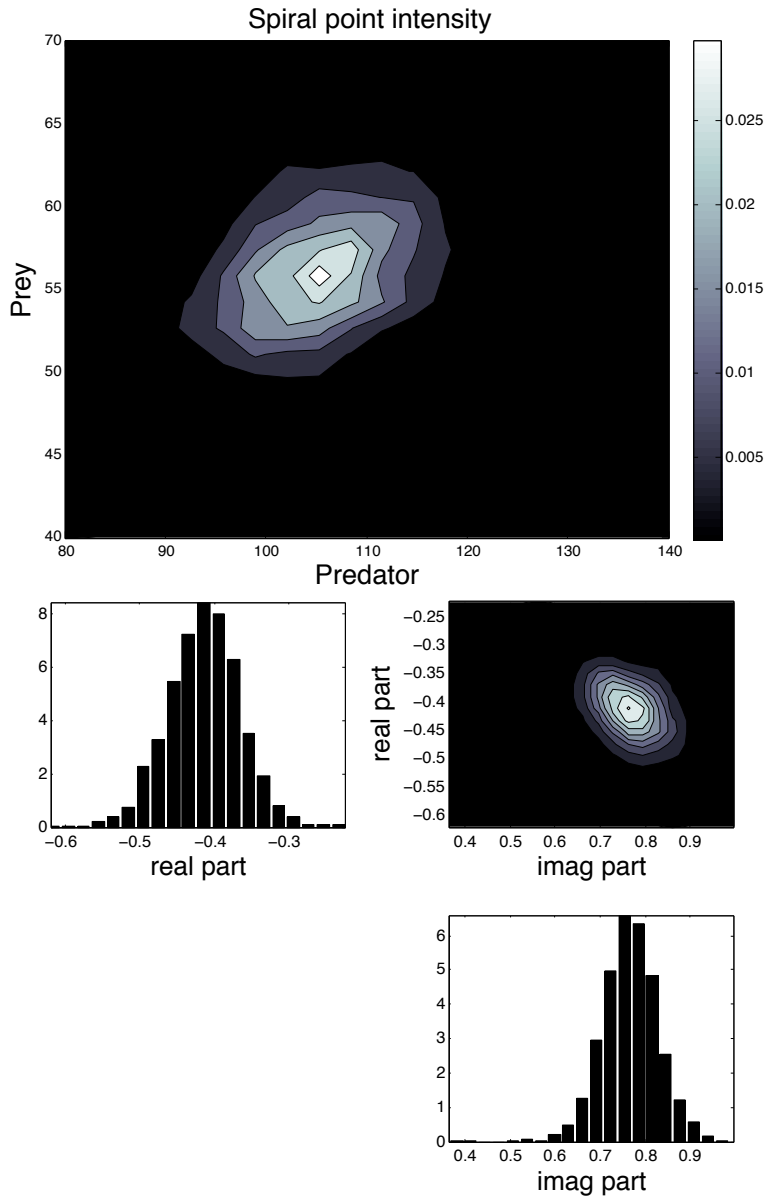


Figure 2.11: Posterior intensity of the stable spiral point for Veilleux's experiment, and the posterior of the most unstable eigenvalue.

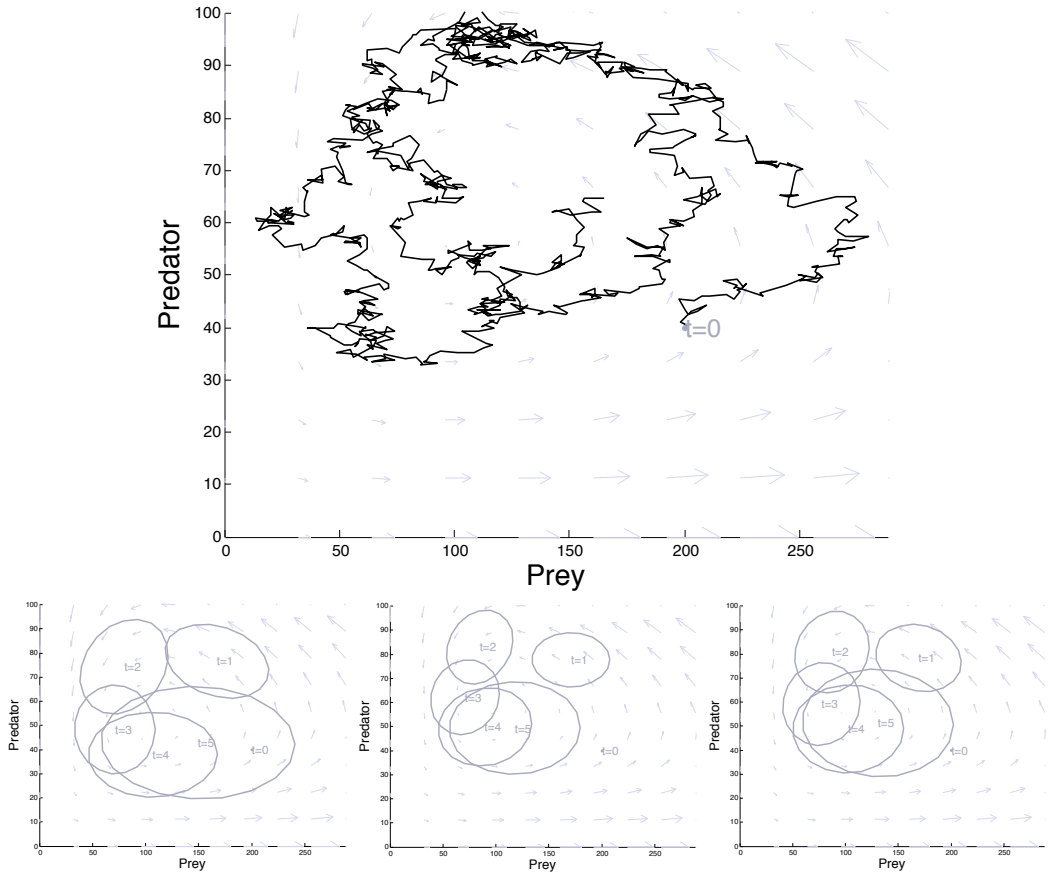


Figure 2.12: Some preliminary phase-space analysis for Veilleux's experiment. A trajectory conditional on one parameter sample from the posterior. It tends to wind around the spiral point, but does not converge. Also shown are three samples from the posterior of the flow field, with error ellipses of a trajectory starting at a particular point conditional on those samples.

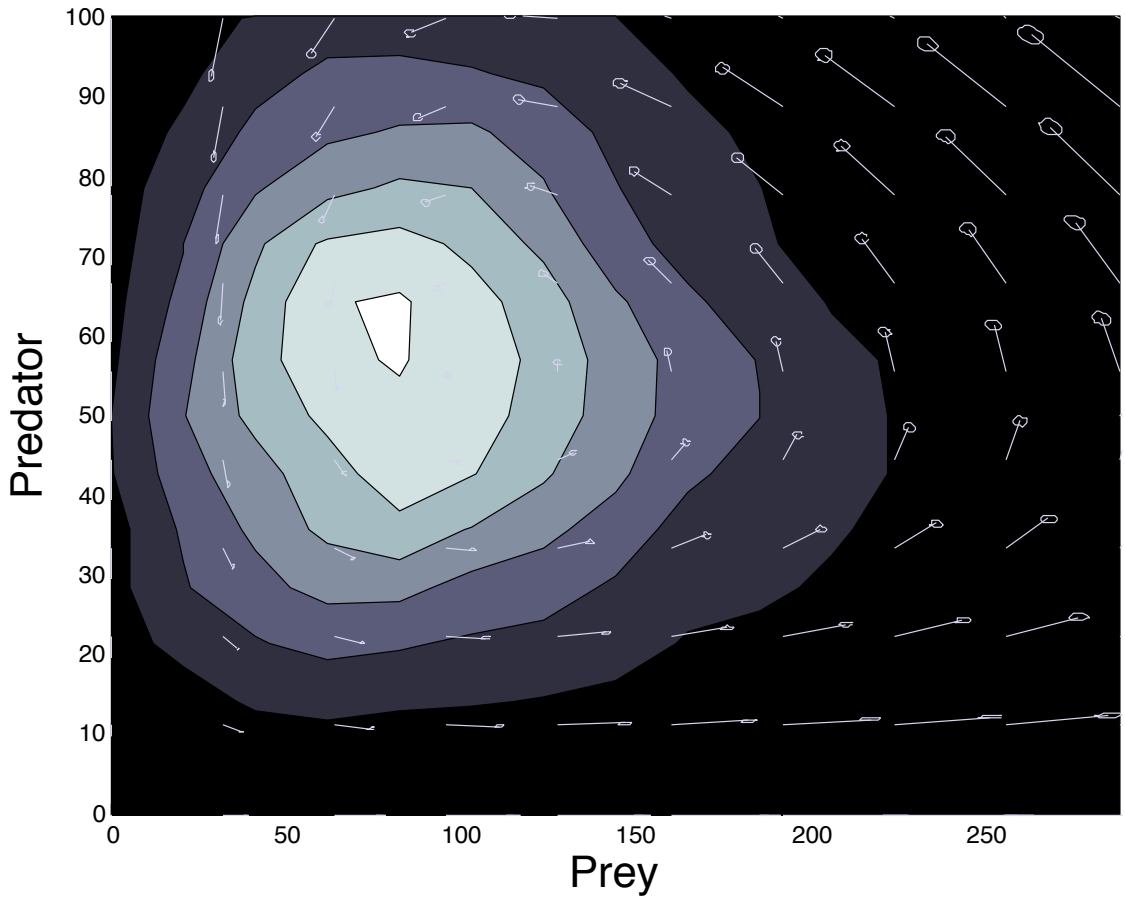


Figure 2.13: A visualization of the posterior of the phase portrait for Veilleux's experiment, incorporating process stochasticity. The heads of the flow vectors have been replaced by error ellipses. The role of a limit cycle or fixed point is played by the stationary distribution.

of predator and prey population sizes.

We have attempted to provide a method with good scaling properties because rational management of human intervention in ecosystems under model uncertainty would seem to be a natural endpoint for this line of research. Such dynamical summaries will undoubtedly prove useful to both basic science and ecological managers if this endpoint is attained. However, in a stochastic Bayesian dynamical framework it becomes feasible to ask questions such as 'what is the probability that a particular event will take place, given a particular management action will be implemented?'. Answers to such questions are usually not available in deterministic dynamical studies; deterministic dynamical systems are generally considered tools for providing qualitative understanding, rather than quantitatively faithful summaries of the forces governing the population dynamics. However, in management applications a case could be made that such probabilistic questions are more relevant than qualitative dynamical understanding.

Answering probabilistic questions will usually require forecasting, and fortunately forecasting is very easy once a dynamical model has been fit using MCMC. Of course forecasts can be generated using stochastic population models with prespecified parameter values, but conducting forecasts as part of Bayesian inference is often preferable because it propagates all uncertainty in parameter values appropriately. Figure 2.14 shows forecasts generated for Veilleux's dataset. Note that the forecast trajectories are conditioned to remain positive, but for purposes of fitting we have assumed that conditioning on $x_t, y_t > 0$ for $t > T$ makes minimal difference. An obvious alterna-

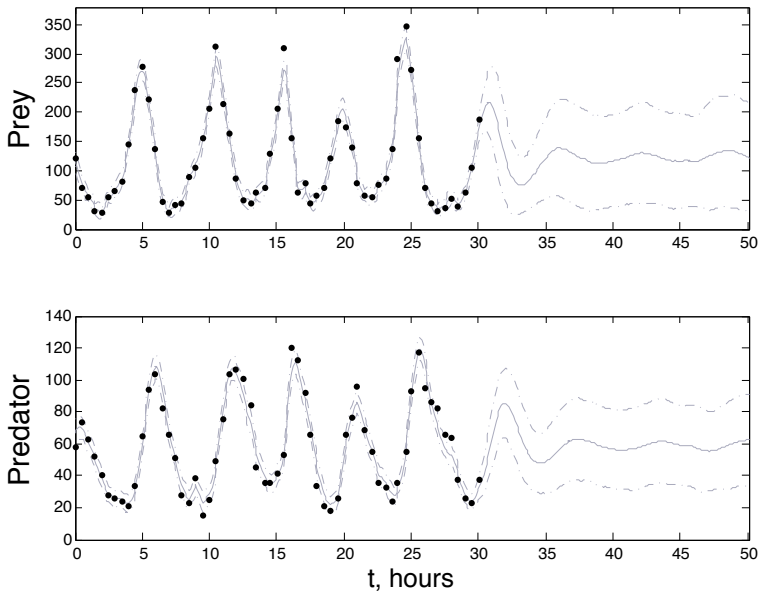


Figure 2.14: Forecasts for Veilleux's dataset. The apparent dampening of oscillations is illusory; although individual trajectories can be expected to oscillate indefinitely, it is impossible to predict the phase of oscillations far into the future. The mean and standard deviations shown are essentially averaged over phase.

tive would have been to model the population dynamics in log space, but this is not necessarily the 'correct' model. See Appendix A.2 for a derivation of the model from mechanistic considerations.

2.6 Discussion

The shape of the coupling function $f(x, y)$ and the functional response $f(x, y)/y$ is a central question in population dynamics, and it has generated substantial debate in recent decades [4]. Previous analyses of this particular dataset have come to different conclusions regarding the shape of the functional response. Jost and Arditi [44] find

that a prey-dependent model is a better fit under the assumption of pure process error, whereas a ratio-dependent model is a better fit under the assumption of pure observation error. As previously mentioned, Jost and Ellner [45] find evidence for some degree of predator dependence. Here we account for both measurement and process uncertainty and find that the fitted functional response is neither ratio nor prey dependent, nor is it simply a function of the scalar argument xy^{-a}.

The ratio dependence/ prey dependence question is more than merely academic; it has important implications for rational management of human interventions in ecosystems. Arditi and Ginzburg [8] and Ginzburg and Akçakaya [30] point out some of these. In particular, ratio-dependent theory predicts that the steady-state biomass of all trophic levels increases in response to increasing primary productivity, whereas prey-dependent theory predicts that steady-state biomass at increasing trophic levels will exhibit 'alternating positive, negative and zero' responses.

Our inferred functional response bears only a very rough resemblance to the functional form shown in the introduction, indicating that the analysis in the introduction is an incomplete description of the predation dynamics. However, our inferred functional response does appear to develop stronger ratio-dependence at higher predator densities, and there is reason to believe that the type of prey-sharing discussed here is partly responsible; the protozoan predators we consider have been observed to share prey, and the process of prey consumption can take two to three minutes [89]. Further modeling work is warranted to more completely characterize the predation dynamics in this system, and the statistical method presented in this chapter should be applied

to more datasets for systems with possibly ratio-dependent dynamics to investigate the domain of applicability of ratio dependence.

Analyses with purely deterministic models [97, 44, 33] have found deterministic skeletons that produce limit cycle behavior. Interestingly, this is Veilleux's [89] original conclusion as well despite the observation that the fixed point occurs to the right of the hump in his reconstructed prey isocline. By simultaneously accounting for measurement errors and allowing for the possibility of process randomness, we find that the population oscillations are likely to be the result of stochastic forcing around a stable spiral point. This 'quasi-cycle' behavior has been noted in ecology for several decades [70] but no previous analysis of Veilleux's data were able to identify these dynamics. Veilluex's results with low growth medium concentrations tend to show damped cycling while those at the highest concentrations tend to show extinction. Oscillations obtain at intermediate concentrations, e.g. the dataset we consider. Our results suggest that growth medium level causes oscillations by changing the magnitude of damping relative to the stochastic forcing rather than by forcing actual bifurcations from stable spiral to limit cycle.

For the past two decades a small group of researchers have been arguing for the importance of semi-parametric methods in ecology for inference of dynamical system components from time series data [24, 94]. Microbial model systems have been a valuable tool for testing ecological theory and have made important contributions to our understanding of community dynamics [43]. We find that the application of modern semi-parametric statistical methods to a classic microbial predator-prey system

reveals the importance of stochasticity in maintaining population oscillations that have previously been characterized as deterministic limit cycles.

We have conducted our analysis using a fitting algorithm that has a good chance of retaining tractibility for multispecies systems. In addition to contributing to the functional response and transmission function literatures, we hope this work will eventually help provide tools for rational management of human interventions in more complicated species communities and infectious diseases.

Chapter 3

Implications of dynamical uncertainty for fishery management

3.1 Introduction

This chapter explores the implications of dynamical uncertainty (expressed by placing a GP prior on a rate function) for fishery management. The purpose of the statistical modeling is different than in chapter 2. The model presented here is simple and highly schematic, so it is not well-suited for direct use in management. However, the simplicity of the model makes it easier to understand several phenomena that will likely arise even when more sophisticated versions are applied and to map out the directions in which appropriate responses may be found. Several compromises are made to keep the model simple enough to fit using only linear algebra so that the concepts can be explored without the major time commitment required to fit models like the one in

chapter 2. The work described in this chapter was done in collaboration with Stephan B. Munch of Stony Brook University.

The Magnuson-Stevens act, the primary legal instrument used to manage US marine fisheries, requires fishery management plans to "assess and specify the present and probable future condition of, and the maximum sustainable yield and optimum yield from, the fishery, and include a summary of the information utilized in making such specification." [71] It further requires "objective and measurable criteria for identifying when the fishery to which the plan applies is overfished." In practice, the act has been widely interpreted as requiring depleted stocks to be rebuilt at least to B_{MSY}, or the steady-state biomass at maximum sustainable yield [77]. If a stock has not been observed at non-depleted biomass levels recently, calculating its B_{MSY} and forecasting its biomass to that level require extrapolation of stock dynamics to unobserved biomasses using population models.

In statistical contexts, population models can be interpreted as very restrictive Bayesian priors on the space of relationships between past and future stock states. As is evident from the recurrent controversy over functional forms for stock-recruitment relationships [67], theory in fishery science is rarely unambiguous enough to provide such prior information. Forecasts based on particular population models fail to propagate the model uncertainty currently endemic to fish population theory.

Munch et al. [65] demonstrate the utility of Gaussian process regression for inference of stock-recruitment functions. They argue that this framework is appropri-

ate because it is better to 'let the data speak for themselves' in light of the prevailing uncertainty in stock-recruitment theory. They find that the stock-recruitment function can be inferred with reasonable precision for stock biomasses that have been observed, but outside the region where data exist its uncertainty grows rapidly. This is a manifestation of the fact that Gaussian process regressions are better for interpolation than for extrapolation [86], and Munch et al. [65] argue that it is a desirable feature. The modeler shouldn't claim to know much about the stock-recruitment relationship for unobserved stock states.

In what follows, we address the qualitative effects of explicitly accounting for both process and structural uncertainties on forecasting for rebuilding management as called for in the Magnuson-Stevens act. As in chapter 2 , our dynamical framework is nonparametric and may be considered a Bayesian extension of previous work on nonparametric inference for ecological models pioneered by Wood et al. [96], Jost and Ellner [45], and others. We apply the framework to two species with very different life histories, bluefish (*Pomatomus saltatrix*) and canary rockfish (*Sebastes pinniger*). We find that forecast uncertainty grows quickly when the simulated biomass trajectories reach regions where the stock dynamics are uncertain, especially at the fringes of the available data.

A species' life-history parameters and stock-recruitment relationship may appear to change through time due to demographic [66], evolutionary [35] or ecological [17] changes that cannot be captured in an unstructured single-species modeling framework. The difficulties associated with explicitly modeling these factors are well-known [54],

but simply excluding them from the model will not make them go away. The concept of maximum sustainable yield itself becomes problematic under structural uncertainty and with temporal change because many 'local' B_{MSY}'s may exist simultaneously, and because MSY and B_{MSY} can change with time along with the population model. These observations echo qualitative concerns that others have raised regarding operational definitions of the MSY concept [77]. Note that Ricker [78] included the caveat "For species with fluctuating recruitment, the maximum might be obtained by taking fewer fish in some years than in others" in his original definition of MSY, and since then many authors have studied MSY in stochastic contexts [66, 48, 55].

We explicitly allow the population model to change with time without sacrificing the model's simplicity, so that observations in the distant past can be considered less relevant to the present. For stocks that have been in decline for some time, this means the apparent maximum observed biomass decreases with time, further inflating forecast uncertainty.

To address these difficulties, we develop a simple means of assessing 'local' stock health given the current state of the stock dynamics. We find indications that this assessment is feasible even under extreme model uncertainty, but that the precision associated with it may be lower than desired.

Concerns regarding the theoretical and practical performance of management based on using MSY as a target have led some researchers to advocate considering F_{MSY}, or the mortality rate due to fishing that will maintain a population at the B_{MSY} level, as an upper limit on fishing mortality [54, 15, 16]. Although Mace [54] argues that

overfishing 'alarms' have been more successful as fishery management tools than fishing mortality targets and limits, some idea of an appropriate mortality rate is an important part of any fishery management plan. We demonstrate that annual surplus production can in principle be estimated (that is, the underlying dynamical model can be used to provide a more informed estimate than that obtained by simply subtracting adjacent datapoints), and this knowledge can be used to calculate simple heuristic upper limits for fishing mortality. However, the estimate we obtain from the simple model we present tends to be biased high.

3.2 Model development

3.2.1 Stochastic production model with unknown function

In order to make the main ideas as transparent as possible, we make our Bayesian inferences and forecasts using a stochastic production model that is very flexible with regard to stock dynamics, but allows calculation of exact posterior distributions using linear algebra only (rather than simulation techniques like Markov Chain Monte Carlo [27]). The top levels of the hierarchy are

$$\begin{aligned} B_t &= f_t(B_{t-1}) - Y_t + \epsilon_t, \\ \epsilon_t | V &\sim N(0, V), \end{aligned} \tag{3.1}$$

where B_t represents adult biomass at time t; Y_t is yield, i.e. the biomass removed by fishing; and ϵ_t is uncorrelated random forcing. The function f_t governs the stock dynamics at time t; it is implicitly tasked with encapsulation of all population processes

except human-induced mortality, including growth, spatial inhomogeneity, time-varying natural mortality due to interspecies interactions, and reproduction [66, 17, 38]. Models of this kind are known as single-species surplus production models, see Quinn and Deriso [76] and Hilborn and Walters [36] for reviews of other types of models used in fishery stock assessments. We will treat time-constant stock dynamics, where $f_t = f_0$ for all t, as a special case.

We assume that we observe Y_t and B_t with no error. We recognize that this assumption is restrictive, but retain it for the sake of clear exposition and closed-form posterior distributions. Measurement error can be incorporated relatively easily, but it requires simulation-based state-space methods [60]. West and Harrison [93] present Markov Chain Monte Carlo methods for fitting dynamic linear models that are embedded in more complicated models in chapter 15. However, the framework outlined below can be generalized to incorporate age structure without resorting to simulation techniques.

Throughout the remainder of this chapter, the observed biomass in year t is denoted B_t while nonspecific values on the biomass axis are denoted b or b'. The symbol f_t with no argument refers to the whole function, while $f_t(b)$ refers to the number obtained by evaluating f_t at b.

3.2.2 Nonparametric prior for f_0

The lower levels of the hierarchy are priors for the functions f_t and the process variance V. Because it is difficult to make the 'right' choice for f_t [67], we leave it to be

inferred nonparametrically as Munch et al. did. We begin specifying a prior for f_t by specifying a Gaussian process prior for its state f_0 at the time of the first stock biomass measurement conditional on V:

$$f_0|V \sim \text{GP}\left(m_0, \frac{V}{S_0}C_0\right). \tag{3.2}$$

We assign an inverse-gamma prior to V:

$$\frac{1}{V} \sim \text{Gamma}\left(\frac{n_0}{2}, \frac{n_0 S_0}{2}\right) \tag{3.3}$$

S_0 is an initial guess for V, and n_0 is the 'effective number of observations' supporting this guess. We set $n_0 << 1$, so that the prior on V is diffuse.

We use mean guess function $m_0(b) = kb$, where $k = \frac{1}{T-1}\sum_{t=1}^{T-1}\frac{B_{t+1}+Y_t}{B_t}$. We begin constructing our prior covariance function, C_0, with the Matèrn covariance function

$$C_M(b, b'; \phi, \nu, \alpha) = \frac{\phi}{2^{\nu-1}\Gamma(\nu)}(\alpha|b - b'|)^\nu K_\nu(\alpha|b - b'|).$$

Γ is the gamma function, which interpolates the factorials. K_ν is a modified Bessel function of the second kind of order ν [2]. Random functions generated using this covariance function have pointwise variance ϕ and are ν times differentiable [86]. Tunable differentiability makes the Matèrn covariance function a standard choice in spatial statistics. $1/\alpha$ plays the role of 'lengthscale of wiggling.'

We believe a priori that $f_0(0) = 0$, but a Gaussian process with a Matèrn covariance function does not have this property. Using standard multivariate normal conditioning formulas [93] and passing back to the continuous limit, we condition on

$f_0(0) = 0$ to obtain the covariance function

$$C_0(b, b') = \frac{\phi}{2^{\nu-1}\Gamma(\nu)} \left[(\alpha|b - b'|)^\nu K_\nu(\alpha|b - b'|) - \frac{(\alpha^2|bb'|)^\nu K_\nu(\alpha|b|) K_\nu(\alpha|b'|)}{2^{\nu-1}\Gamma(\nu)} \right].$$

We choose $\nu = 3$, $1/\alpha = 2 \max_t B_t$, and $\sqrt{\phi} = \max_t B_t/4$, so that f_0 will be three times continuously differentiable and will wiggle with lengthscale and amplitude both proportional to $\max_t B_t$.

Multiplying the covariance function C_0 by $\frac{V}{S_0}$ in (3.2) seems odd because it introduces prior dependence between f_0 and V. This rescaling is common practice in dyamic linear models because it gives rise to posterior distributions with a standard form, and it is not as odd as it seems at first glance. If we discretize f_0 to obtain the finite-dimensional normal random variable $\vec{f_0}$, (3.2) can be expressed using much more pleasing notation:

$$\left.\begin{array}{l} \frac{1}{V} \sim \text{Gamma}\left(\frac{n_0}{2}, \frac{n_0 S_0}{2}\right), \\ \vec{f_0}|V \sim \text{N}\left(\vec{m}_0, \frac{V}{S_0}\vec{C}_0\right) \end{array}\right\} \Rightarrow \vec{f_0} \sim \text{T}_{n_0}(\vec{m}_0, \vec{C}_0).$$

T_{n_0} denotes Student's T distribution with n_0 degrees of freedom [93]. However, we do not know of any standard notation for Gamma mixtures of Gaussian processes (though 'T process' seems like a sensible choice) and we resist the temptation to discretize at the model-specification stage or to introduce non-standard notation. We set $n_0 = 1$ and $S_0 = \sum_{t=1}^{T-1} (B_{t+1} - B_t)^2/(T-1)$.

3.2.3 Priors for f_t with discounted historical information

In order to specify conditional priors for f_t, $t > 0$, we opt for the discount factor approach of West and Harrison [93], which they describe in section 6.3. This

approach will allow us to specify these priors via a single 'discount factor' δ, which has a straightforward interpretation in terms of the rate at which historical information is forgotten.

This approach is as follows: Suppose the posterior of f_t, given data $B_0 \dots B_t$ up to time t and a value for process uncertainty V, is a Gaussian process with some mean function m_t and covariance function $\frac{V}{S_t}C_t$:

$$f_t|B_0 \dots B_t, V \sim \text{GP}\left(m_t, \frac{V}{S_t}C_t\right).$$

The scalar S_t is a point estimate for V given data up to time t which will be defined in the next section. We specify the τ-step-ahead prior of $f_{t+\tau}$, conditional on the the same dataset, as

$$f_{t+\tau}|B_0 \dots B_t, V \sim \text{GP}\left(m_t, \delta^{-\tau}\frac{V}{S_t}C_t\right), \quad 0 < \delta \leq 1. \tag{3.4}$$

To emphasize, this is a conditional prior that depends on the data up to time t; it is not an unconditional prior, which would be specified before seeing the dataset. It is important to note that this parametrization is not meant to be a realistic model for the evolution of the stock-recruitment relationship. The discounting approach is based on the view that, if we are able to learn f_t with good precision, we should be able to guess f_{t+1} with comparable precision.

It turns out that setting $\delta = 1$ is equivalent to setting $f_t = f_0$ for all $t > 0$. In this case, historical information is relevant forever. If $\delta < 1$, on the other hand, the stock dynamics as encapsulated in f_t are allowed to change through time. The magnitude of the covariance function of the τ-step-ahead conditional prior of $f_{t+\tau}$ given data up to

time t is an exponentially increasing function of time. To rephrase, $-\log\delta \approx 1-\delta$ can be interpreted as the rate at which historical information is forgotten.

3.2.4 Fitting the model

Despite its flexibility, the model can be fit without the Markov Chain Monte Carlo algorithm [27]. Although it is not immediately apparent, the model defined by (3.1), (3.2) and (3.4) is closely related to the Dynamic Linear Model (DLM) of West and Harrison [93], and the following fitting algorithm can be derived from their results.

Theorem 1 *Assume that the joint posterior of* $(f_{t-1}, 1/V)$ *given* $B_0 \dots B_{t-1}$ *is, in hierarchical form,*

$$f_{t-1}|B_0 \dots B_{t-1}, V \sim \text{GP}\left(m_{t-1}, \frac{VC_{t-1}}{S_{t-1}}\right),$$
$$\frac{1}{V}\middle|B_0 \dots B_{t-1} \sim \text{Gamma}\left(\frac{n_0+(t-1)}{2}, \frac{S_{t-1}(n_0+(t-1))}{2}\right).$$

Then the joint posterior of $(f_t, 1/V)$ *given* $B_0 \dots B_t$ *is*

$$f_t|B_0 \dots B_t, V \sim \text{GP}\left(m_t, \frac{VC_t}{S_t}\right),$$
$$\frac{1}{V}\middle|B_0 \dots B_t \sim \text{Gamma}\left(\frac{n_0+t}{2}, \frac{S_t(n_0+t)}{2}\right).$$

where m_t, C_t and S_t are defined for arbitrary arguments b and b' by

$$m_t(b) = m_{t-1}(b) + \frac{C_{t-1}(b,B_{t-1})}{C_{t-1}(B_{t-1},B_{t-1})+\delta S_{t-1}} \left(B_t + Y_t - m_{t-1}(B_{t-1})\right),$$

$$C_t(b,b') = \frac{S_t}{\delta S_{t-1}} \left[C_{t-1}(b,b') - \frac{C_{t-1}(b,B_{t-1})C_{t-1}(B_{t-1},b')}{C_{t-1}(B_{t-1},B_{t-1})+\delta S_{t-1}}\right], \tag{3.5}$$

$$S_t = S_{t-1}\left[1 + \frac{1}{n_0+t}\left(\frac{\delta(B_t+Y_t-m_{t-1}(B_{t-1}))^2}{C_{t-1}(B_{t-1},B_{t-1})+\delta S_{t-1}} - 1\right)\right].$$

See Appendix B.2 for the proof.

This fitting algorithm only takes into account data obtained to date, so the results we present should be interpreted as the inferences managers would have made as data arrived. When it is needed, West and Harrison's methodology for retrospectively updating past posterior distributions in light of later data can also be adapted to model (3.1).

3.2.5 Fit to a simulated dataset

To see how this statistical framework performs when the population dynamics are known, we generate data using the stochastic production model

$$\begin{aligned} B_{t+1} &= Ge^{-M}B_t + \frac{\alpha B_t}{1+\beta B_t} e^{X_\sigma + \sigma^2/2} - Y_t, \\ X_\sigma &\sim \mathrm{N}(0,\sigma^2). \end{aligned} \tag{3.6}$$

As before, B_t represents adult biomass and Y_t represents yield biomass. e^{-M} represents the proportion of adults surviving from year to year, G represents their proportional growth, and α and β are parameters. We choose $\alpha = .4$, $\beta = 1/100,000$, $\sigma = .2$, and $Ge^{-M} = .8$.

This model is not of the same form as (3.1) because process randomness is multiplicative, but we use it to simulate data because it is more biologically plausible and commonly used for single-species dynamics; our hope is that this procedure provides a reasonable test of model (3.1)'s ability to deal with real data. It would have been preferable to use multiplicative process error in (3.1), but methodological constraints preclude this. Despite the fact that the data were generated from a model of a different form, our inference of f_0 (with $\delta = 1$) is generally accurate where data have been observed (figure 3.1).

If f_0 is a Gaussian process, its derivatives (if they exist) are also Gaussian processes. Figure 3.2 shows the posteriors of f_0' and f_0'' given the simulated data.

3.2.6 B_{MSY} and an overfishing indicator

Biomass at maximum sustainable yield, or B_{MSY}, is defined for deterministic population models as the steady-state solution for which yield is maximized. For (3.1) with V set to zero and δ set to 1 (so that $f_t = f_0$ for all t), assuming f_0 is continuously differentiable,

$$\begin{aligned} B_{MSY} &= \arg\max[f_0(b) - b] \\ &\Rightarrow f_0'(B_{MSY}) = 1. \end{aligned} \tag{3.7}$$

Since $f_t|B_0 \dots B_t, V$ is a Gaussian process, $f_t'|B_0 \dots B_t, V$ and $f_t''|B_0 \dots B_t, V$ are also Gaussian processes.

$f_0'(b) = 1$ is a necessary condition for b to equal B_{MSY}, but it is not sufficient. Flexible priors like (3.2) allow f_0' to be equal to 1 for many values of biomass, as is

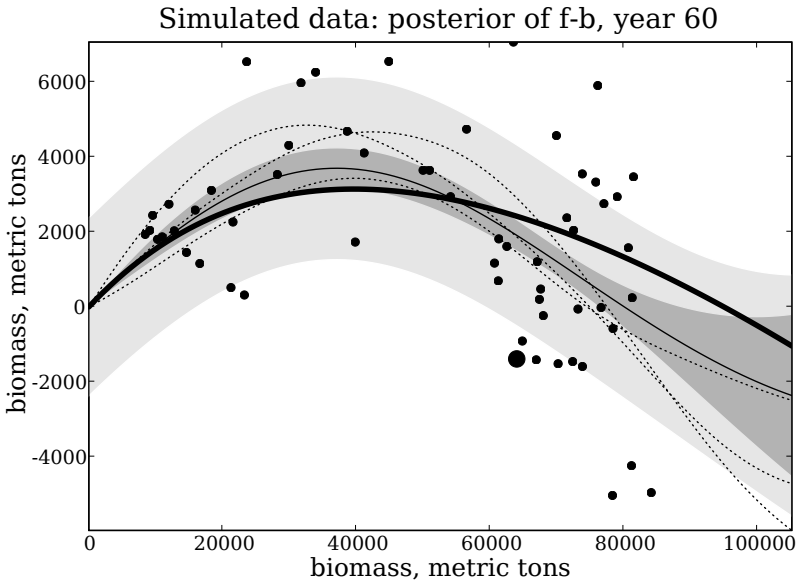


Figure 3.1: Model fit to a simulated dataset. The posterior mean of surplus production $f_{60}(b) - b$ is shown as a solid curve, the standard deviation of $f_{60}(b) - b$ is shaded dark grey and the predictive standard deviation $\sqrt{\mathrm{Var}[f_{60}(b) - b] + V}$ is shaded light grey. The value of $f(b) - b$ for the function f that was actually used to generate the data is shown as a heavy line. Three realizations from the posterior are shown as dotted lines. Points indicate pairs $(B_{t-1}, B_t + Y_{t-1} - B_{t-1})$ that have been observed. The heavy point indicates the last observed pair.

evident from the posterior draws shown as dotted lines in figure 3.2. The problem of searching for a global B_{MSY} is complicated, because the global B_{MSY} will likely depend heavily on our choice of prior; we don't know much about stock dynamics at unobserved biomasses. Also, when $\delta < 1$ the stock dynamics can change with time, so it seems strange to search for an eternal maximum sustainable yield. These observations echo concerns that have previously been raised with the practical application of maximum sustainable yield calculated with deterministic models [77].

Recognizing these difficulties associated with managing based on global opti-

mization, we present a vector-valued indicator for assessing whether a stock is depleted based on local stock dynamics. The overfishing indicator attempts to characterize f_t locally via its value and its first two derivatives:

$$\text{overfishing indicator} = \begin{bmatrix} \frac{f_t(B_{t-1})-B_{t-1}}{B_{t-1}} \\ f_t'(B_{t-1}) - 1 \\ B_{t-1} f_t''(B_{t-1}) \end{bmatrix}. \tag{3.8}$$

The first component determines whether expected surplus production is positive, the second component determines whether B_{t-1} was above or below the nearest point b where $f_t'(b) - b = 1$, and the third component determines the concavity of f at B_{t-1}. All three components are dimensionless. We choose B_{t-1} rather than B_t because B_{t-1} is the argument of f_t in (3.1).

We consider a stock to be 'locally depleted' if:

- the second component is positive *OR*
  - the first component is negative *AND*
  - the third component is positive.

The second component tests the condition in equation (3.7), but alone it is not sufficient to rule out the possibility that the stock is above a local optimum within a region of negative production (e.g., due to Allee effects); the first component is a safeguard against this possibility. However, the first component is too stringent a test in that if a stock is actually above its local equilibrium biomass surplus production may be negative;

therefore we do not take negative surplus production as evidence of overfishing if f is locally concave down.

Conditional on V and data up to time t, the overfishing indicator has a multivariate normal posterior distribution. It is straightforward to compute the probabilty that the overfishing indicator is in the locally-depleted octant using the properties of the multivariate Student's T distribution or by simulating from its posterior (recall that the T distribution is a a scale-mixture of normals) [93]. It is important to note that the indicator makes no attempt to measure the *dregee* of local depletion. One practical consequence is that the indicator cannot distinguish between surplus production functions that are known to be locally nearly flat and those whose slope is not known with good precision.

The plots in the upper panel of figure 3.3 show the posterior of the overfishing indicator at each time given data collected up to that time. They illustrate how the overfishing indicator would have been calculated over time, rather than a retrospective assessment of the stock's past status given all data available to date. The lower panel shows the time course of the posterior probability that the simulated stock is locally not depleted given data available to date. The indicator lags the time when the stock ceases to be depleted by roughly a decade, but overall provides a relatively good indication of overfishing.

3.2.7 Estimation of surplus production and mortality limits limits

An additional quantity of interest is the projected population growth rate $f_t(B_t)/B_t - 1$ at year $t+1$ in the absence of fishing, conditional on data up to time t. Like the components of the overfishing indicator, this quantity depends on local stock dynamics, and can in principle be used to set an upper limit for fishing mortality Y_t/B_t. As an example, figure 3.4 shows the largest value of Y_t/B_t that would have been reckoned to provide positive population growth with 75% posterior probability at each year given data to date for the simulated data. Although such mortality limits have no relationship to optimality, they at least ensure that the stock will continue to grow.

Unfortunately, figure 3.4 shows that our simple statistical framework badly overestimates this mortality limit for the simulated dataset; many of the take limits it provides would give the stock less than a 25% chance of growing. It is to be hoped that more detailed models will be able to do better.

3.3 Application to canary rockfish and bluefish data

In this section, we investigate our ability to generate long-term forecasts using estimated bluefish (*Pomatomus saltatrix*) and canary rockfish (*Sebastes pinniger*) biomass data from stock assessments [58, 68], to assess current stock health using the overfishing indicator, and to estimate surplus production. It is important to note that these 'data' are actually the output of previous statistical analyses. This is helpful in one sense, because our statistical model is too assumption-bound to account for obser-

vation uncertainty. On the other hand, it is undesirable because model assumptions used in previous analysis may confound our statistical analysis. In practice it would be best to embed a model like ours in a larger model that can connect the underlying dynamics to the actual observations, which are likely to be gleaned from logs of fishing vessels and trawl data.

Canary rockfish are found associated with rocky reefs in deep waters off the Pacific coasts of the US and Canada. They are slow growing, mature at between 4-12 years of age and live up to 80 years. In contrast, bluefish are voracious piscivores found worldwide, typically in coastal regions with strong boundary currents [46]. On the US east coast, bluefish are found from the Gulf of Mexico to as far north as Maine [18]. Bluefish grow rapidly, live for about 15 years, and mature between ages 1 and 2. Commercial fisheries for both species exist in the US. Both have experienced massive declines within the last three decades prompting severe restrictions on landings.

3.3.1 Long-term forecasting

We first consider the case $\delta = 1$, so that the function governing stock dynamics is constant in time: $f_t = f_0$ for all t. Fits of the model to the datasets are shown in figures 3.5 and 3.6, and biomass forecasts are shown in figure 3.7. No simple closed form exists for the predictive distribution of future biomasses, so these forecasts have to be generated using Monte Carlo simulations. Because our simple framework models error additively, it tends to predict extinction with unreasonably high probability for unfished stocks. We therefore condition forecasts on stock persistence over the forecast

intervals.

The standard deviation of forecast bluefish biomass grows very quickly with time. The standard deviation of forecast canary rockfish biomass, on the other hand, grows at a more reasonable rate until the expected biomass reaches the upper limits of the dataset. At that point, the interval begins to grow much more quickly.

These results are easy to understand. Once the stock biomass B_t has grown large enough that $f_0(B_t)$ is uncertain, the forecast distribution of B_{t+1} becomes correspondingly uncertain. Since the forecast distribution of B_{t+2} depends on B_{t+1}, it inherits the uncertainty in B_{t+1} and adds its own, and so on. Since f_{25} for bluefish is appreciably uncertain even within the range of the data, model uncertainty begins contributing to forecast uncertainty right away.

For canary rockfish, historical stock biomasses were large enough that preliminary analysis with some parametric models (not shown) tends to indicate that early recorded stock biomasses exceeded B_{MSY}. Our results with $\delta = 1$ give some hope that this historical data will allow reasonably precise forecasting up to B_{MSY} even in our nonparametric framework, but this precision should be taken with a grain of salt. The stock's environment, spatial distribution, and age and size composition have changed to varying degrees since the 1940s, and fishing pressure may have selected for fish phenotypes that were rare prior to overfishing. The population dynamics and B_{MSY} long ago may have been very different from what they are today.

In our modeling framework, we can express the possibility of changes in stock dynamics by setting δ to some value less than 1, so that historical information is progressively forgotten. Figure 3.8 shows the fit to the canary rockfish dataset and biomass forecasts with $\delta = .95$. Note that f_{61} is much more uncertain for biomass levels near the upper limit of the dataset than it was with $\delta = 1$, and that the explosion of forecast uncertainty begins earlier (consider the time at which the interval hits the upper limit of the plot for $\delta = 1$ vs. $\delta = .95$). Since we are a priori uncertain of what value of δ most appropriately captures the environmental changes that have taken place, the observation that decreasing δ 'unzips' our region of confidence in f_{61} to progressively lower biomasses is enough to make us wary of putting too much faith in our forecasts, especially over the long term.

In practice, detailed prior information relevant to the historical course of f_t may be available from biological knowledge, fishing mortality records for associated species, records of climate change and habitat degradation, etc. Translation of such knowledge into priors without recourse to parametric population models is an area for future research. The DLM framework could be used to search for many types of temporal trends, oscillations, and autoregressive behaviors in f_t, see chapters 7, 8, and 9 of West [93]. However, in practice the DLM framework may not be appropriate because the prior variance of f_t will grow without bound in t if $\delta < 1$.

Figure 3.9 shows the likelihood of the discount factor δ for the two datasets. For both species, the likelihood attains its maximum at $\delta = 1$, indicating that constant

f_t provides the best statistical fit. However, as previously discussed we have strong a priori reasons to believe that f_t is not constant. Converting these reasons systematically into models for the time-evolution of f_t is an area for future research.

3.3.2 Assessment of overfishing status

Figures 3.10 and 3.11 show the posterior expectation of the overfishing indicator with two standard deviation intervals for bluefish and canary rockfish, respectively, given data available to date. The overfishing indicator seems to be fairly robust to δ, which is a desirable feature since it is meant to assess stock health even in changing environments.

The two standard deviation intervals for the first two components are somewhat wider than desirable; as we have parametrized the model, assessment of overfishing status would never have been unambiguous for these two species. Preliminary analysis indicates that the choice of covariance function C_0 affects the calculated value of the overfishing indicator and the size of the intervals, so in practice making a careful choice for C_0 (possibly by inferring its parameters rather than fixing them a priori) will be important. Fitting simulated data can be a useful check on the choice of covariance function.

Using the overfishing indicator, canary rockfish would have been considered locally not depleted during the period from about 1955 to about 1975 with appreciable probability. Prior to 1955, canary rockfish would have been considered depleted with

probability considerably greater than half although its biomass was large, possibly due to the limited dataset available at that time. Bluefish would have been considered depleted with probability greater than half at all years. Because of the poor performance of our estimates of mortality limits with simulated data, we do not estimate mortality limits for canary rockfish or bluefish.

3.4 Discussion

We have considered a very flexible nonparametric model which acknowledges widely-recognized uncertainty in the mechanism behind stock dynamics and the possibility that that mechanism changes through time. Having acknowledged this model uncertainty, we find it difficult to forecast the state of depleted stocks or to do the global optimization required for computation of MSY, F_{MSY} and B_{MSY}.

With these considerations in mind, we explore the possibility of management based on local stock dynamics. We present a local overfishing indicator and indications that it is capable of doing its job even when stock dynamics are allowed to change. However, its assessment of the status of real species tends to be fairly ambiguous. This uncertainty is hardly a surprise given the complexity of stock dynamics in nature and the model uncertainty our framework reflects, but in practice there may be prior biological information available that could help pin down the stock state more precisely. We also present a fishing mortality limit designed to maintain the instantaneous growth rate of stocks above a given level.

The overfishing alarm and mortality limit we present are extremely simple. More sophisticated models would of course permit computation of more biologically sophisticated equivalents, and in particular might provide more accurate estimates of take limits. Even in our simple framework it would be possible to expand the region considered 'local,' possibly by simulating from the posterior.

However, even when sophisticated dynamical models are used there is value in keeping management guidelines simple. As has been widely recognized [57, 84, 54] simplicity gives management advice a much better chance of being seriously implemented. Nonparametric inference of stock dynamics encourages development of simple indicators by shifting focus to the dynamical big picture rather than the analytical details of particular functional forms.

Assuming that our prior (or another nonparametric prior that gives rise to similar extrapolation uncertainty) is a reasonable summary of the current state of knowledge of a stock's surplus production function, the Magnuson-Stevens act seems to be asking too much of fishery science. Use of more complete dynamical models is unlikely to resolve the difficulties, as most models will introduce their own unknown functions. Our results suggest that fishery managers should focus locally, in both time and stock state, and not depend on global optimization or long-term forecasting except in cases where a stock's dynamics are very well understood.

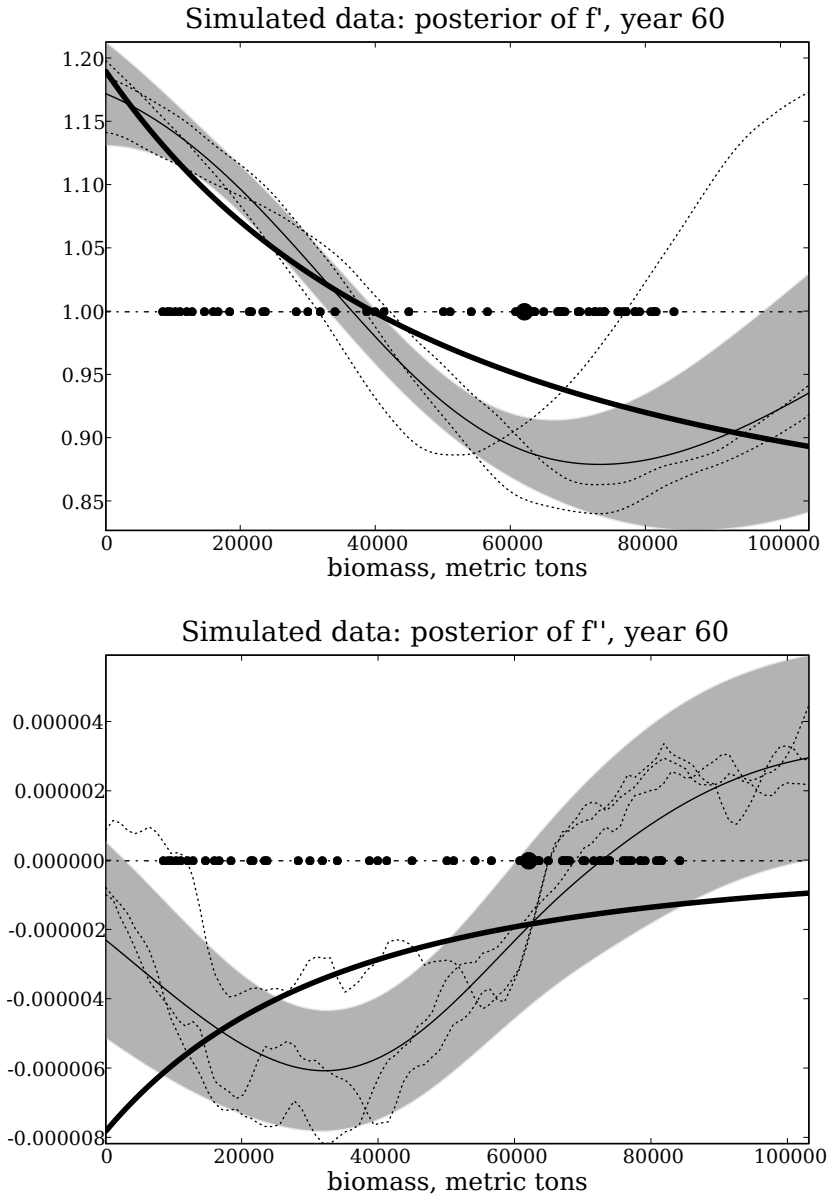


Figure 3.2: **Top:** Posterior of $f'_{60}(b)$ given simulated biomasses $B_0 \ldots B_{60}$ with $\delta = 1$. The posterior mean is shown as a solid line, and the posterior standard deviation is shaded in grey. The true $f'(b)$, the derivative of the function that was used to generate the data, is shown as a heavy line. The horizontal line indicates the recovery threshold, one. Points are placed at stock biomasses that have been observed; the heavy point indicates the last observation. The dotted curves are three realizations from the posterior of f'_{60}. **Bottom:** The posterior of $f''_{60}(b)$ given simulated biomasses $B_0 \ldots B_{60}$ with $\delta = 1$. The units of $f''_{60}(b)$ are inverse metric tons.

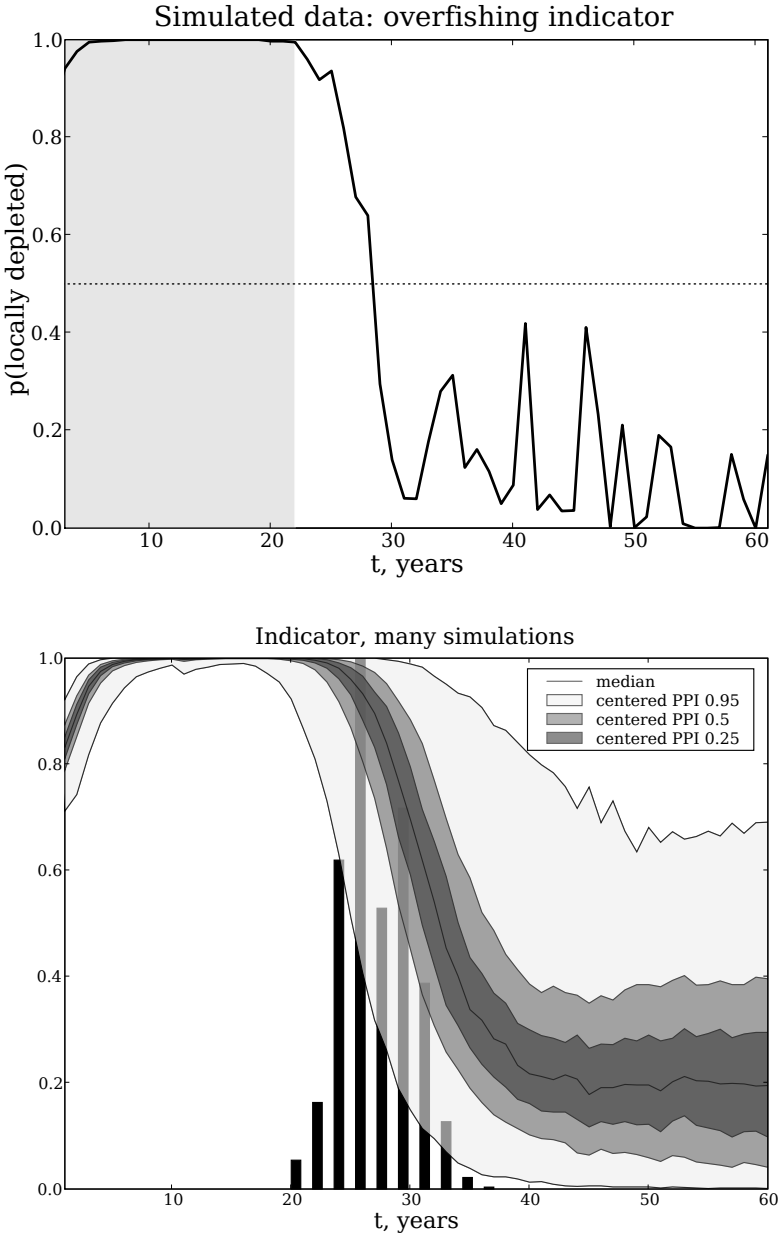


Figure 3.3: **Top:** The indicator for the simulated dataset. The shaded region indicates the times at which the stock actually was below B_{MSY}. **Bottom:** The distribution of the indicator, calculated based on data available to date, for many realizations from the stochastic model 3.6. The histogram indicates the distribution of times when the stock biomass actually exceeded its B_{MSY}.

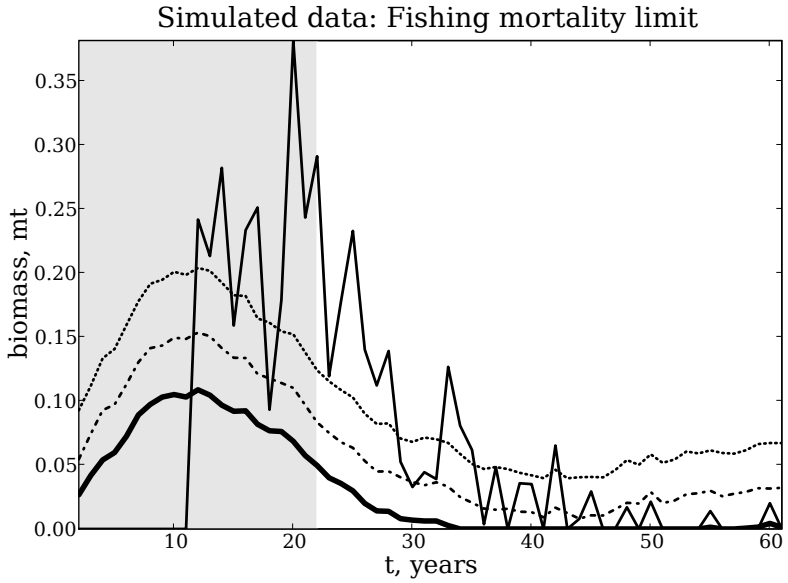


Figure 3.4: The light solid line shows the estimated maximum value of $F_t = Y_t/B_t$ that will allow a positive growth $(f_t(B_{t-1}) + B_{t-1} + \epsilon_t - Y_t)/B_{t-1}$ with 75% posterior probability given data available to date for the simulated dataset. The heavy solid, dashdot and dotted lines show the values of Y_t/B_{t-1} allowing a positive growth rate with 75%, 50% and 25% probability respectively, computed using the model that was used to produce the data. The shaded region indicates the years when the stock biomass was below B_{MSY}. Ensuring positive growth would usually be considered most important during these years.

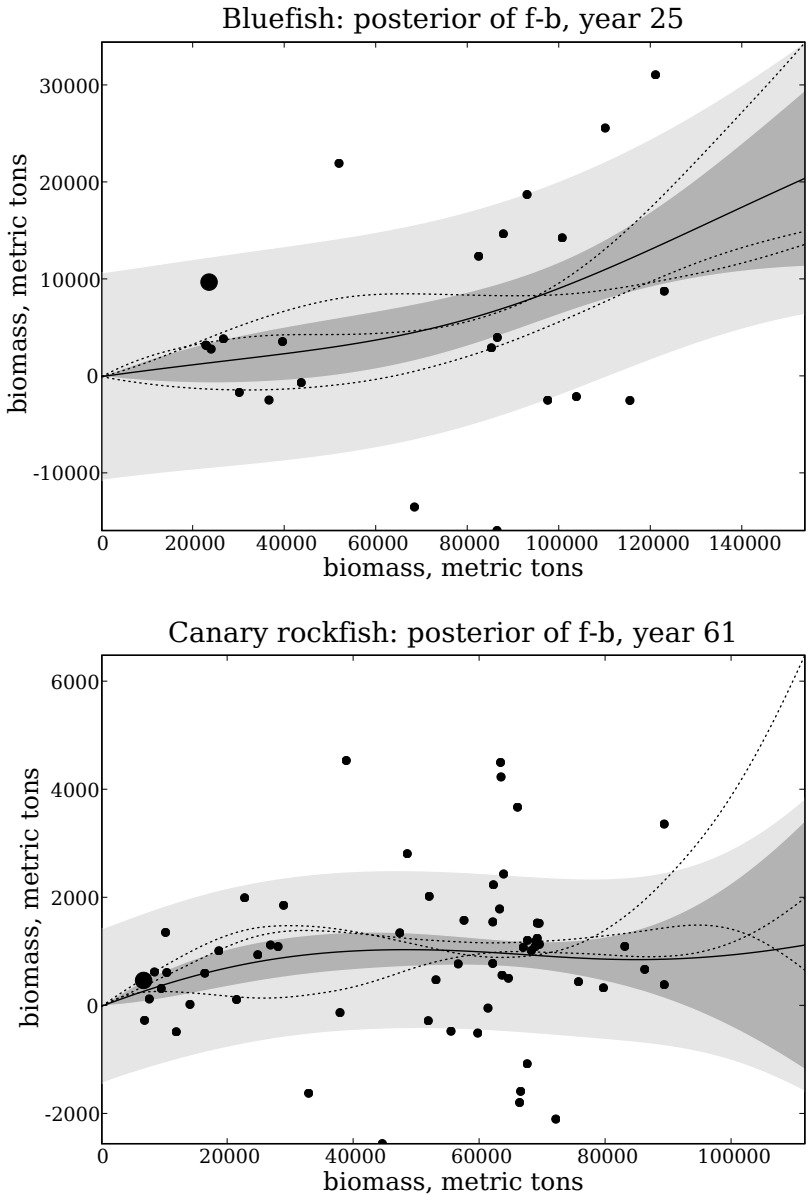


Figure 3.5: Fits to the canary rockfish and bluefish datasets with $\delta = 1$, so that $f_t = f_0$ for all t. **Top:** The posterior of $f_{25}(b) - b$ for bluefish. The solid line indicates the posterior mean, the dark grey region indicates the posterior standard deviation and the light grey region indicates the posterior predictive standard deviation. Points indicate observed pairs $(B_{t-1}, B_t + Y_{t-1} - B_{t-1})$. The heavy point indicates the last observation. The dotted lines are draws from the posterior. **Bottom:** The posterior of $f_{61}(b) - b$ for canary rockfish.

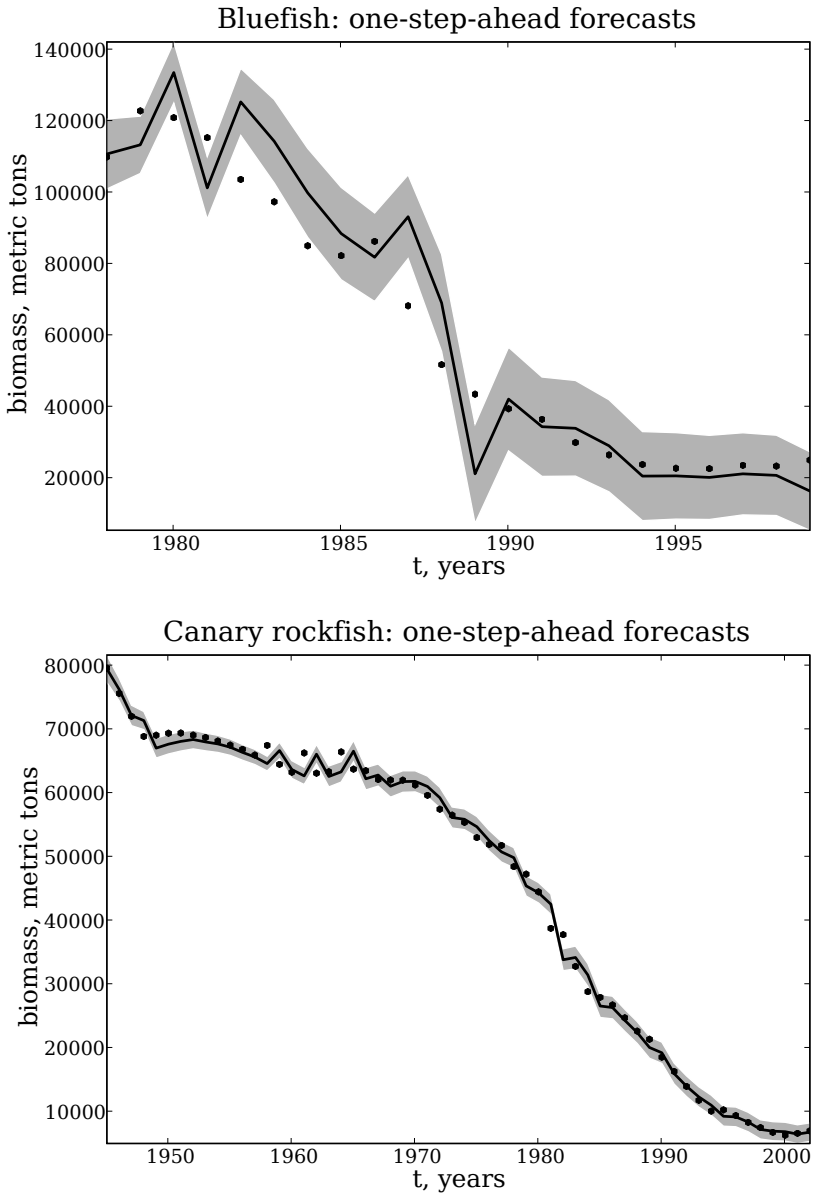


Figure 3.6: One-step-ahead forecasts with $\delta = 1$ for the canary rockfish and bluefish datasets. These provide a visual indication of the goodness of the model's fit to the data [93].

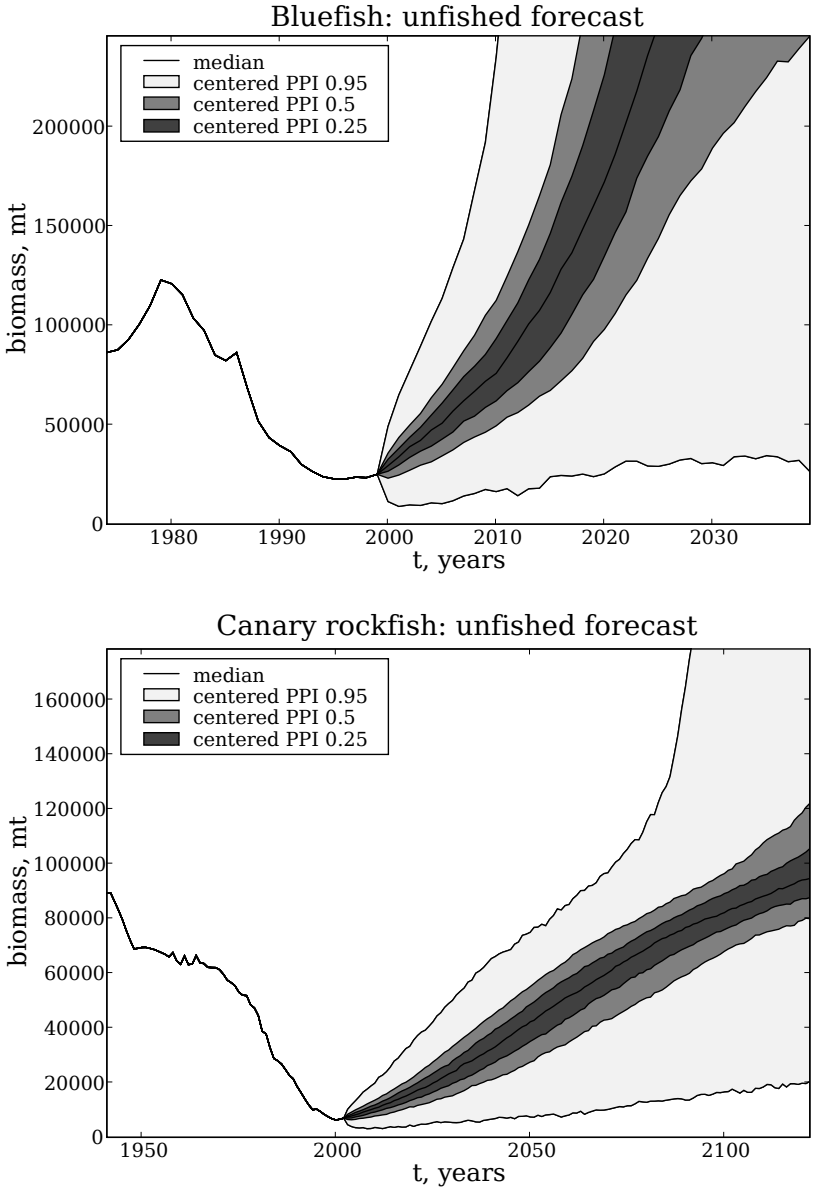


Figure 3.7: Forecasts with $\delta = 1$. Top: The forecast of unfished bluefish biomass. The posterior probability intervals grow quite quickly, mostly due to model uncertainty. Bottom: Forecast of unfished canary rockfish biomass. The intervals grow more slowly relative to the median, but once simulated trajectories reach the largest size that has been observed they begin to accumulate dynamical uncertainty.

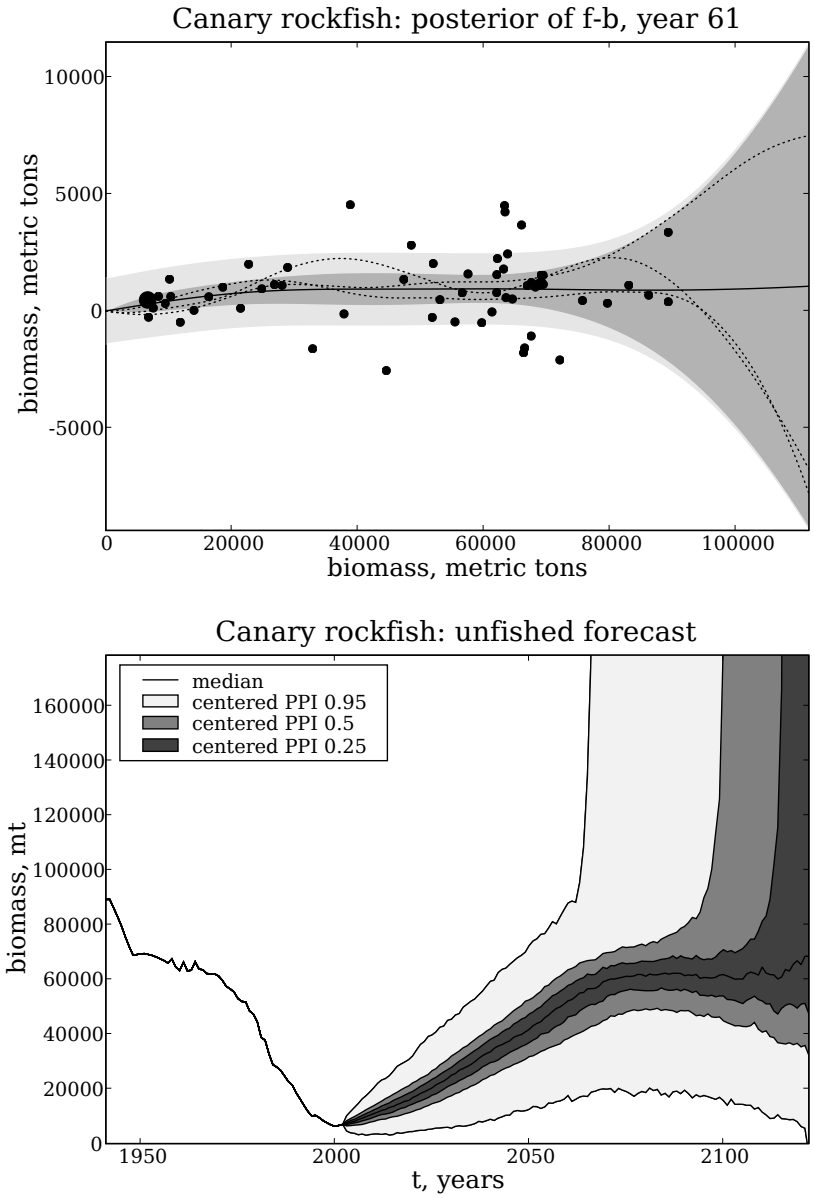


Figure 3.8: Results of fitting the Canary rockfish dataset with $\delta = .95$, allowing for environmental changes. **Top:** The posterior of $f_{60}(b) - b$. The solid line indicates the posterior mean, the dark grey region indicates the posterior standard deviation and the light grey region indicates the posterior predictive standard deviation. Points indicate observed pairs $(B_{t-1}, B_t + Y_{t-1} - B_{t-1})$. The heavy point indicates the last observation. The dotted lines are draws from the posterior. Note that the shaded intervals are 'unzipped' for smaller values of biomass than with $\delta = 1$. **Bottom:** The posterior probability intervals of forecast unfished biomass grow more quickly than they did with $\delta = 1$; the upper interval hits the upper limit of the plot about twenty years earlier.

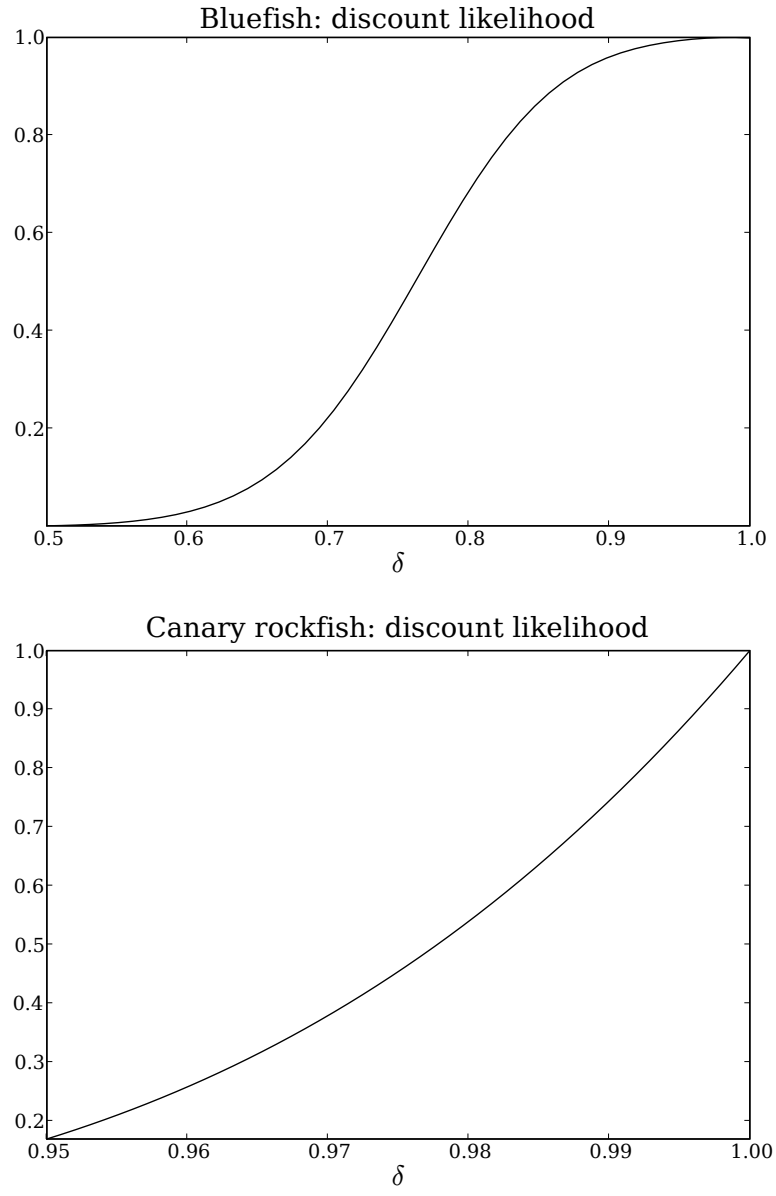


Figure 3.9: The scaled likelihood of the discount factor for the canary rockfish and bluefish datasets. Likelihood is maximized by $\delta = 1$ for both models.

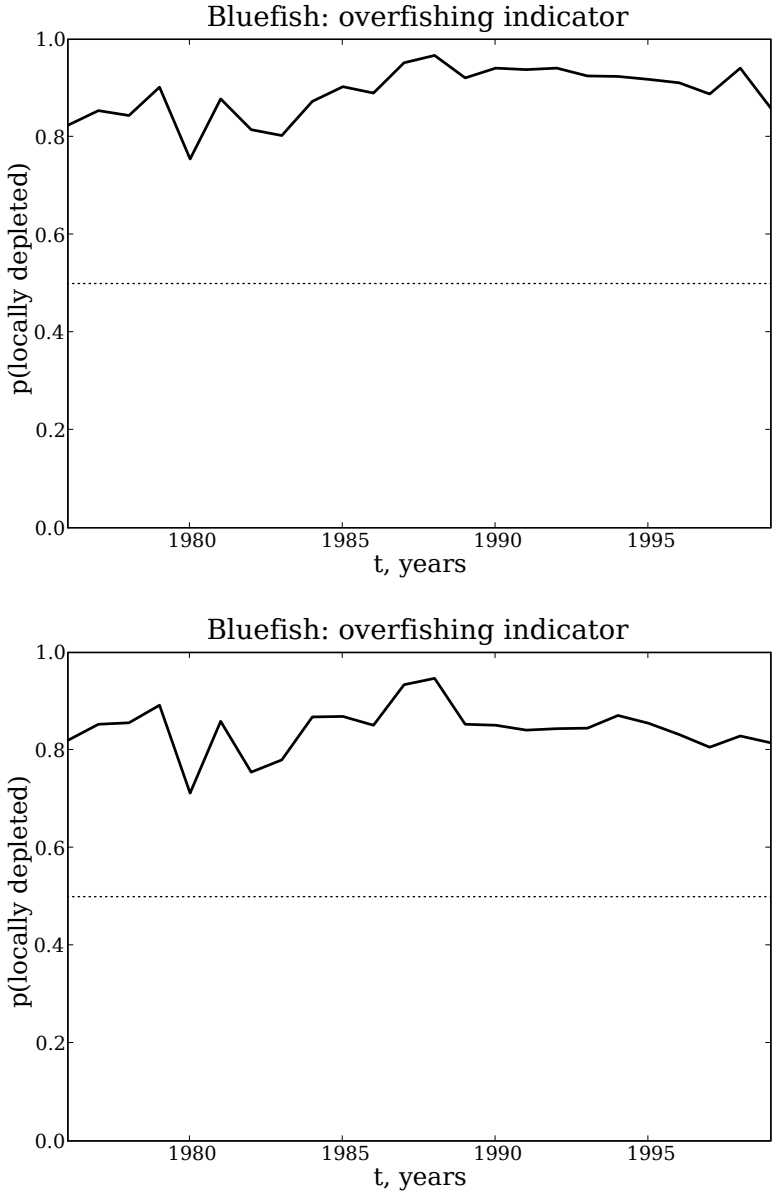


Figure 3.10: **Top:** The posterior probability that bluefish is locally depleted over time, with $\delta = 1$. **Bottom:** The posterior probability that bluefish is locally depleted over time, $\delta = .95$. Because it uses local information, the indicator is qualitatively insensitive to environmental changes as measured by δ.

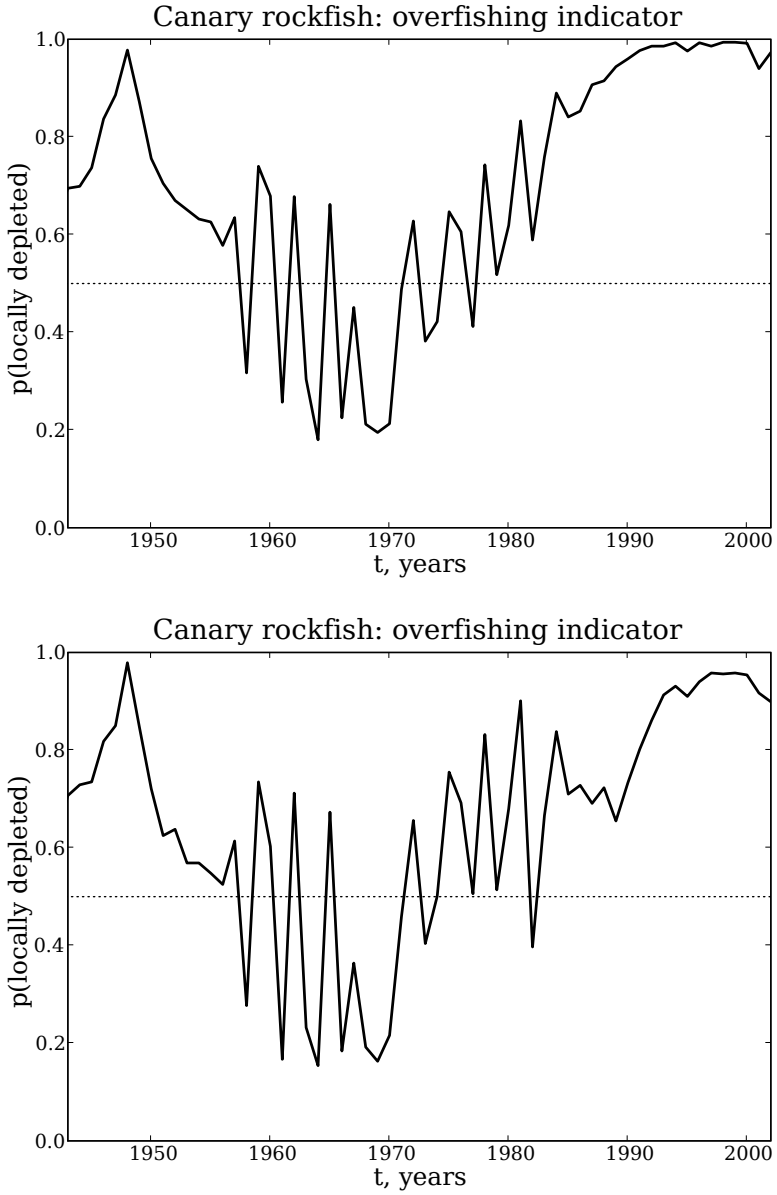


Figure 3.11: **Top:** The posterior probability that canary rockfish is locally depleted over time, with $\delta = 1$. **Bottom:** The posterior probability that canary rockfish is locally depleted over time, $\delta = .95$. Because it uses local information, the indicator is qualitatively insensitive to environmental changes as measured by δ.

Chapter 4

An accessible implementation of Gaussian processes

4.1 Introduction

This chapter describes a handful of building blocks designed to make unrestricted and innovative use of GPs more accessible. These building blocks are:

- Objects representing mean functions, covariance functions, and realizations of Gaussian processes.
- A function that implements standard Gaussian process regression by 'imposing' normally-distributed observations on the mean and covariance functions.
- An object representing GP realization-valued random variables.
- Infrastructure for incorporating such random variables into a general-purpose

Markov Chain Monte Carlo (MCMC) [27] package along the lines of Bugs (`www.mrc-bsu.cam.ac.uk/bugs/`).

The approach described here attempts to provide an intuitive implementation by relying on objects that closely resemble the functional concepts they represent. It allows users to specify and fit probability models of their own devising involving Gaussian processes based on a conceptual understanding, without worrying about the underlying linear algebra.

An open-source Python (`www.python.org`) implementation, called 'Random Realizations', is available from

`code.google.com/p/gaussian-process`

The accompanying documentation [75] is much more tutorial in orientation than this chapter, but it assumes some familiarity with Python. Potential users of the package are strongly recommended to consult the documentation rather than this chapter. This chapter describes the implementation of the objects and is not intended as a user's guide.

The particular MCMC package with which Random Realizations interfaces is PyMC, version 2.0 and higher (`code.google.com/p/pymc`). PyMC is an open-source collaboration between Christopher Fonnesbeck of the Florida Fish and Wildlife Research Institute, David Huard of INRS-ETE at Université du Québec, and myself. Like Random Realizations, it is implemented in Python.

4.1.0.1 Why Python?

There are several reasons behind my choice of Python as the implementation language.

- Python's broad goal is 'niceness' to the programmer, rather than speed or adherence to a particular programming paradigm (`http://www.python.org/doc/essays/cp4e.html`).

- Python is dynamically typed, meaning that the types of arguments to functions do not need to be known ahead of time. This makes it much easier to incorporate random variables valued as arbitrary objects, such as GP realizations, into a general-purpose MCMC framework.

- As an interpreted language, Python is relatively slow. However, experience shows that in MCMC the computational expense of the program logic tends to be dwarfed by the expense of the floating-point computation, especially calculation of log-probabilities. These slow portions tend to be small and fairly insignificant in terms of mental burden.

  Python has numerous 'escape hatches' to C and Fortran, which make it possible to optimize slow portions of the code and recover most of the performance loss. These escape hatches are much easier to use than their counterparts in R and MATLAB. For speed, Random Realizations' floating-point computation is either done in Fortran 77 or using Numerical Python's [73] linear algebra utilities.

- As an interpreted object-oriented language, Python allows the objects described in this chapter to be created in an interactive session and manipulated. I feel that this feature makes the basic objects much more valuable as teaching and learning tools.

- Python is a real programming language, with extensive libraries available to support non-statistical scientific and application programming. It is supported and developed by a large community of programmers.

4.1.1 Terminology, notation and calling conventions

In this chapter the building blocks' user interfaces are presented Python, but their internal workings and the algorithms they use are presented in pseudocode. An overview of the notation and some terminology from object-oriented programming follows.

- A *class* is a prototype for a class of objects, each of which is an *instance* of the class. New class instances are created by passing initialization arguments to the class. For example, a new instance of class `Realization` called f would be created from mean function M and covariance function C as follows:

```
f = Realization(M, C).
```

- An *attribute* of an object is an item associated with the object, which can be accessed via the object. I will denote object attribute access with a period. If G is

a random variable valued as a Gaussian process realization, `G.value` gives access to its current value (which is an instance of class `Realization`, like `f`).

- A *method* of an object is a special attribute of the object that can be called like a function. Calling an object's method causes the object to do something. If C is a covariance function, The call `C.cholesky(x)` causes C to compute the Cholesky factorization of its evaluation on the vector x and return it.

- A *callable object* is an object that can be called like a function. For example, the Gaussian process realization objects are callable because they are intended to represent random functions:

  ```
  f(x)
  ```

  would evaluate f at vector of values x.

Note that there is much more to object-oriented programming than this, and that different authors have varying views on the fundamental concepts. See Armstrong [9] for a review of object-oriented concepts in practice or Silvert [83] for an introduction to object-oriented programming in the context of ecological simulation.

In this chapter, code is represented in a typewriter font:

```
f = Realization(M, C)
```

Square brackets denote array indexing, and colons indicate ranges of consecutive integers:

$$K[1:m, 2:n] = \begin{bmatrix} K[1,2] & \cdots & K[1,n] \\ \vdots & \ddots & \vdots \\ K[m,2] & \cdots & K[m,n] \end{bmatrix}.$$

Array indexing begins at 1 (note that in numerical Python indexing begins at 0). Curly braces denote concatenation:

$$\{x, y\} = [x[1] \ldots x[n_x], y[1] \ldots y[n_y]]. \tag{4.1}$$

Specialized notation is used for evaluating functions on vector arguments. Here $\mathbf{x}$ is a vector of values and x is a single value. For one-place functions (mean functions and realizations),

$$f(\mathbf{x}) = [f(x[1]) \quad \ldots \quad f(x[n])]. \tag{4.2}$$

For two-place functions (covariance functions),

$$\begin{aligned} C(x[1], x) &= \begin{bmatrix} C(x[1], x[1]) & \cdots & C(x[1], x[n]) \end{bmatrix}, \\ C(x, x) &= \begin{bmatrix} C(x[1], x[1]) & \cdots & C(x[1], x[n]) \\ \vdots & \ddots & \vdots \\ C(x[n], x[1]) & \cdots & C(x[n], x[n]) \end{bmatrix}, \\ C(x, x[1]) &= \begin{bmatrix} C(x[1], x[1]) & \cdots & C(x[n], x[1]) \end{bmatrix}^T. \end{aligned} \tag{4.3}$$

4.2 The basic objects and normally-distributed observations

4.2.1 Mean functions

Objects of class `Mean` represent mean functions. These objects are 'wrappers' for ordinary functions, meaning that their purpose is to mediate between the ordinary functions and the outside world. A mean function M can be created from an ordinary Python function g as follows:

```
def g(x, a, b, c):
    return a * x ** 2 + b * x + c


M = Mean(g, a = 2, b = 1, c = 3).
```

The first initialization argument is the function g, and the subsequent initialization arguments are values for the extra parameters that g requires. These values are 'memorized' by M and passed to g at each subsequent call, so the call

```
M(5)
```

is equivalent to

```
g(5, 2, 1, 3).
```

Mean's only notable attribute is a method called `observe`, which will be described in section 4.2.3

4.2.2 Covariance functions

GP covariance functions are represented by the class `Covariance`, and like `Mean` instances they are wrappers for ordinary functions. However, the underlying functions of `Covariance` instances take two input arguments. A Gaussian covariance function could be produced as follows:

```
def euclidean_distance(x, y):
    D = matrix((len(x), len(y)))
    for i in range(len(x)):
        for j in range(len(y)):
            D[i,j] = sqrt((x[i] - y[j]) ** 2)
    return D


def gaussian(x, y, amp, scale):
    C = euclidean_distance(x, y)
    C *= scale
    for i in range(len(x)):
        for j in range(len(y)):
            C[i,j] = amp * exp(- (C[i,j] / scale) ** 2)
    return C


C = Covariance(gaussian, amp = 10, scale = .1)
```

The separation between the two stages of computation of the covariance matrix,

1. computation of a distance matrix
2. overwriting the distance matrix with an isotropic and stationary covariance function,

is emphasized intentionally. This arrangement makes it relatively painless to accomodate new coordinate systems and to build in anisotropy and/or nonstationarity by deforming the input space [80] or by redefining distance [74]. Conversely, new functional forms can be easily applied to all available coordinate systems. Random Realizations' 'plugs' for such extensions are described in more detail in the documentation [75].

Like `Mean` instances, `Covariance` instances memorize the extra arguments required by their underlying functions. The following two calls are equivalent:

```
C(2, 3)
gaussian(2, 3, 10, .1).
```

Covariances can also be called with a single argument. For a vector `x`, the call

```
C(x)
```

returns a result equivalent to

```
diag(C(x,x))
```

but does not compute the off-diagonal terms.

Covariance objects have a method called `observe` that, like `Mean.observe`, will be described in 4.2.3. They have two additional methods: `cholesky` and `continue_cholesky`.

4.2.2.1 Covariances' Cholesky method

The following method call with length-n vectors x and V

```
C.cholesky(x, V)
```

returns the following items:

- A length-n vector of pivot indices p.
- An m-by-n upper triangular matrix U such that $U_*^T U_* = C(x,x) + \mathrm{diag}(V)$ for the pivoted matrix $U_* = U[1:m, q]$, where $m \leq n$ and $p[q] = 1:n$.

U is an incomplete Cholesky factor [26] for $C(x,x)+\mathrm{diag}(V)$, with diagonal pivoting applied. If $C(x,x)+\mathrm{diag}(V)$ is full-rank, $m = n$ and U is simply a pivoted Cholesky factor. The following algorithm, based on the algorithm implemented in the MATLAB package 'chol_incomplete' (`http://www.kyb.tuebingen.mpg.de/bs/people/seeger/software.html`), allows U to be computed in $O(m^2 n)$ operations plus a storage cost of $O(n^2)$:

- $d = C(x,x) + \mathrm{diag}(V)$ (the diagonal can be computed without computing the full matrix)
- $d_* = \max d$
- $p = 1:n$
- $U =$ zero matrix of size n by n
- for i from 1 to n:

- $l =$ value of index $j \in i : n$ where $d[j]$ is largest.
- if $d[l]/d_* < \epsilon$:
  - return p, $U[1 : i, 1 : n]$. Rank of $C(x, x) + \text{diag}(V)$ is i.
- $d[i] \leftrightarrow d[l]$, $p[i] \leftrightarrow p[l]$, $x[i] \leftrightarrow x[l]$, $U[1 : i, i] \leftrightarrow U[1 : i, n]$
- $U[i, i] = \sqrt{d[i]}$
- $r = C(x[i], x[i + 1 : n])$
- if $i > 1$:
  - $r = r - U[1 : i, i]^T U[1 : i, i + 1 : n]$
- $U[i, i + 1 : n] = r/U[i, i]$
- for j from $i + 1$ to n:
  - $d[j] = d[j] - U[i, j]^2$

- return p, $U[1 : n, 1 : n]$. $C(x, x) + \text{diag}(V)$ is full-rank.

Similar algorithms are given by Golub and van Loan [32] and implemented in LINPACK (as subroutine `DCHDC`) [22]. Note that the full matrix $C(x, x)$ does not need to be computed ahead of time, and in fact if $m < n$ many of its elements will never be computed at all. More sophisticated pivot selection schemes than the greedy algorithm presented here are available, see for instance Bach and Jordan [10].

After iteration i, the diagonals $d[i + 1 : n]$ give the conditional variance of a realization with covariance C evaluated at $x[i + 1 : n]$ given the value of the realization evaluated at $x[1 : i]$, observed with error variance $\text{diag}(V)$. The algorithm proceeds by

selecting points in order of variance conditional on points already used, and terminates when all remaining points have conditional variance less than $d_*\epsilon$.

This Cholesky factor should eventually be computed as a low-rank update [81] of the (diagonal) Cholesky factor of $\mathrm{diag}(V)$ (Matthias Seeger, pers. comm.), but I have not implemented that optimization.

4.2.2.2 Covariances' Cholesky continuation method

It is frequently desirable to compute an incomplete Cholesky factor for the covariance matrix $C(\{x,y\},\{x,y\}) + \mathrm{diag}(\{V_x, V_y\})$ when the corresponding factor U_x of $C(x,x) + \mathrm{diag}(V_x)$ is known.

In this section, U_x, p_x and m_x denote the incomplete Cholesky factor of $C(x,x) + \mathrm{diag}(V_x)$ and the associated pivots and rank; n_x and n_y denote the lengths of x and y. The combined length of x and y is denoted n.

The call

```
C.continue_cholesky(x,y,Vx,Vy,Ux,px)
```

efficiently computes and returns the following:

- A length-n vector of pivot indices p, n being the length of $\{x,y\}$.
- An m-by-n ($m \leq n$) upper triangular matrix U such that $U_*^T U_* = C(\{x,y\},\{x,y\}) + \mathrm{diag}(\{V_x, V_y\})$ for the pivoted matrix $U_* = U[1:m, q]$, where $p[q] = 1:n$.

The algorithm begins as follows:

- U = zero matrix of size $m_x + n_y$ by n

- $U[1:m, 1:n] =$

$$\left\{ \begin{array}{lll} U_x[1:m_x, 1:m_x], & U_x[1:m_x, 1:m_x]^{-T} C(x[p[1:m_x]], y), & U_x[1:m_x, m_x+1:n_x] \end{array} \right\}$$

- $p = \{p_x[1:m_x],\ m_x+1:m_x+n_y,\ p_x[m_x+1:n_x]\}$

- $x = \{x[p_x[1:m_x]],\ y,\ x[p_x[m_x+1:n_x]]\}$.

The initial layout of U looks like it would during the call

```
C.cholesky({x, y}, {Vx, Vy})
```

when $i = m_x$, so the rest of the algorithm is the same as the incomplete Cholesky algorithm above, but with i ranging from $m_x + 1$ to n. The expense of this algorithm is $O(n(m^2 - m_x^2)) + O(m_x^2 n_y)$, plus a storage cost of $O(n(m_x + n_y))$.

4.2.2.3 Basis covariances

In some cases, significant speed gains can be realized by representing a Gaussian process as a linear combination of a finite set of basis functions e_i with normally-distributed coefficients c_i:

$$\left. \begin{array}{l} f = M + \sum_{i=1}^{N} c_i e_i, \\ c_0 \ldots c_N \sim \mathrm{N}(0, K) \end{array} \right\} \Rightarrow f \sim \mathrm{GP}(M, C)$$

where the covariance function C is defined by

$$C(x, y) = \sum_{i=1}^{N} \sum_{j=1}^{N} e_i(x) e_j(y) K_{i,j}.$$

Particularly successful applications of this idea are:

Random Fourier series: $e_i(x) = \sin(i\pi x/L)$ or $\cos(i\pi x/L)$, for instance [28].

Gaussian process convolutions: $e_i(x) = \exp(-(x - \mu_i)^2)$, for instance [51].

Splines: Several spline interpolations can be represented as linear combinations of basis functions, usually polynomials. [59, 50]

Such covariances are represented by the class `BasisCovariance`, which has the same methods and behavior as `Covariance`. The first initialization argument to `BasisCovariance` is a basis function which takes a coefficient array c and an input argument x. For example, the random Fourier sine series covariance shown above might be implemented as:

```
def sinbasis(c, x, L):
    basis = zeros((len(c), len(x)))
    for i in range(len(c)):
        basis[i,:] = c[i] * sin((i+1) * pi * x / L)
    return basis


C = BasisCovariance(sinbasis, x, N, L=25)
```

`BasisCovariance`'s Cholesky and Cholesky continuation methods are essentially the same as those of `Covariance`, but only a $\min(N, n)$-by-n matrix is allocated ahead of time for the Cholesky factors where N is the total number of elements in c. The diagonal and row vectors are computed on demand in the obvious way.

4.2.3 Normally-distributed observations

The `observe` function and the `observe` methods of `Mean` and `Covariance` share responsibility for solving the following conjugate statistical model:

$$\left.\begin{array}{l} D_i \overset{\text{ind}}{\sim} \mathrm{N}(f(o_i), V_i) \\ f \sim \mathrm{GP}(M, C) \end{array}\right\} \Rightarrow f|D \sim \mathrm{GP}(M_o, C_o). \tag{4.4}$$

More specifically, the call

```
observe(M, C, o, D, V)
```

essentially transforms M and C to M_o and C_o above by calling the `observe` methods.

The formulas for M_o and C_o are as follows, for arbitrary input vectors x and y:

$$M_o(x) = M(x) + C(x, o)[C(o, o) + \mathrm{diag}(V)]^{-1}(D - M(o))$$

$$C_o(x, y) = C(x, y) - C(x, o)[C(o, o) + \mathrm{diag}(V)]^{-1}C(o, y).$$

The ultimate effect of the call to `observe` is the following:

- to store a square rank-m_o Cholesky factor $U_o[1 : m_o, 1 : m_o]$ and associated pivot vector p_o of $C(o[p_o[1 : m_o]], o[p_o[1 : m_o]]) + \mathrm{diag}(V[p_o[1 : m_o]])$ in `C`
- to store the vector $U_o[1 : m_o, 1 : m_o]^{-T}(D[p_o[1 : m_o]] - M(o[p_o[1 : m_o]]))$ in `M`.

This information can be used to compute the above formulas efficiently when input vectors x and y are provided.

Denote $o[p_o[1 : m_o]]$ by o_*, the complement $o[p_o[m_o + 1 : n_o]]$ by o_{**}, and define V_*, V_{**}, D_* and D_{**} analogously. As discussed earlier, due to the properties of the

incomplete Cholesky decomposition the conditional variance of $f(o_{**})|D_*$ is less than or equal to ϵ times the maximum unconditional variance d_* of $f(o_*)$.

Normally-distributed observations are used in three important ways:

- Simple nonparametric regression: direct user application of model (4.4).
- Gibbs sampling a Gaussian process realization when its Markov blanket [41] is of the form (4.4).
- After each call, a `Realization` object 'remembers' the values it returned by means of the `observe` methods (with $V = 0$) to maintain consistency.

Covariances' and means' observe methods are fairly complicated, but most users should not need to call them directly. Denote by o_p, D_p and V_p the values at which observations have been made previously, the observed values and the corresponding variances. These vectors may be length-0. Denote by o_n, D_n and V_n the analogous vectors corresponding to new observations, and by o, D and V the concatenation of these. Denote by n_p and m_p the size and rank of $C(o_p, o_p)$, and by n and m the rank of $C(\{o_p, o_n\}, \{o_p, o_n\})$.

4.2.3.1 Covariances' observe methods

The call

```
C.observe(o, V)
```

does the following:

- Endows `C` with the following attributes:
  - An incomplete rank-m Cholesky factor U_o of $C(\{o_p, o_n\}, \{o_p, o_n\})$+diag$(\{V_p, V_n\})$, truncated to be square.
  - A corresponding vector p of pivots.
- Returns the following values:
  - The subvector $o_* = o[p[1:m]]$ that is represented in the rows of U_o.
  - $I_* = p[1:m]$
  - A matrix $U_{o_n|o_p}$ such that $U_{o_n|o_p}^T U_{o_n|o_p} = C_{o_p}(o_n, o_n)$ up to order ϵ. This matrix is given by

$$U_{o_n|o_p} = U_o[m_p + 1 : m, q[m_p + 1 : n]],$$

    where $p[q] = 1 : n$.

After a covariance's observe method has been called, calls to the covariance will return values according to C_o rather than C in equation (4.4). The return values o_* and I_* are used to prepare input values for `Mean.observe`, and the return value $U_{o_n|o_p}$ is used by realizations to generate new values.

4.2.3.2 Basis covariances' observe methods

Unfortunately, in order for basis covariances' computational advantages to be fully exploited their observe methods need to work differently. The model

$$D_i \overset{\text{ind}}{\sim} \mathrm{N}(f(o_i, V_i))$$

$$f = M + \sum_{i=1}^{N} c_i e_i(o)$$
$$c \sim \mathrm{N}(0, K)$$

can be solved to obtain the following heirarchical posterior for f:

$$f = M_o + \sum_{i=1}^{N} c_i e_i(o),$$
$$c|D \sim \mathrm{N}(0, K_o)$$

where

$$K_o = K - KE(o)(E(o)^T KE(o) + \mathrm{diag}(V))^{-1} E(o)^T K,$$
$$M_o(x) = M(x) + E(x)^T KE(o)(E(o)^T KE(o) + \mathrm{diag}(V))^{-1}(D - M(o))$$

where $E(o)$ is the matrix obtained by stacking the vectors $e_i(o)$, $i = 1 \ldots N$. When observed, basis covariances simply store a Cholesky factor of K_o.

When a basis covariance's observe method is called with new observations o_n, D_n, V_n and previous observations o_p, D_p, V_p, the following happen:

- The basis covariance is endowed with the following attributes:
  - An incomplete Cholesky factor $Q_{\{o_p, o_n\}}$ of $K_{\{o_p, o_n\}}$.
  - A corresponding vector q of pivots.
  - A rank-m incomplete Cholesky factor U_o of $C(\{o_p, o_n\}, \{o_p, o_n\}) + \mathrm{diag}(\{V_p, V_n\})$, truncated to be square.
  - A corresponding vector p of pivots.

- The following are returned:
  - The subvector o_* of o represented in the rows of U_o
  - The corresponding index vector I_*, so that $o[I_*] = o_*$.

Basis covariances do not return the matrix $U_{o_n|o_p}$, because realizations drawn from basis covariances do not need this to produce new values.

4.2.3.3 Means' observe methods

Means' observe methods are called as follows:

```
M.observe(C, o*, d*)
```

where $o_* = o[I_*]$ and $d_* = d[I_*]$. After a mean's observe method is called, calls to the mean will return values from $M_{\{o_p,o_n\}}$ rather than M_{o_p}.

A call to a mean's observe method endows the mean with the following attributes:

- References to U_o and $\{o_p, o_n\}$, taken from C.
- The vector $U_o^{-T}(D[I_*] - M(o[I_*]))$. This vector is computed efficiently from $U_o[1 : m_p, 1 : m_p]^{-T}(D[p[1 : m_p]] - M(o[p[1 : m_p]]))$ if previous observations have been made.

If C is a basis covariance, the call endows the mean with different attributes:

- References to U_o and $[o_p, o_n]$, taken from C.

- The vector $KE(o)(E(o)^T KE(o) + \mathrm{diag}(V))^{-1}(f(o) - M(o))$, computed efficiently.

  Means' observe methods do not return any values.

4.2.3.4 The observe function

The call `observe(M, C, o, D, V)` presented above simply does the following:

- Calls `C`'s observe method with o and V.
- Calls `M`'s observe method using the output of the previous call, plus D.
- Optionally calls `cross_validate` (see below), and raises an error if the return value is false.

4.2.3.5 Cross-validation

The incomplete Cholesky decomposition provides numerical stability; even if the covariance matrix is singular, calls to `observe` will execute successfully. However, because the incomplete Cholesky decomposition effectively ignores certain datapoints it is sometimes desirable to check those datapoints for consistency post hoc. The function `predictive_check` optionally checks the remaining data values D_{**} against $M_{o*}(o_{**})$, and raises an error if any elements disagree by more than ten upper-bound standard deviations $\sqrt{\epsilon d_*}$.

4.2.4 Gaussian process realizations

Gaussian process realizations are represented by callable objects of class `Realization`, whose initialization arguments are a mean function and a covariance function:

```
f = Realization(M, C).
```

Realizations can also be forced to take values y on input values m:

```
f = Realization(M, C, y, m).
```

This is useful for Metropolis algorithms. If a realization is created this way, the initial values y are checked for consistency using the cross-validation scheme described above at initialization.

Gaussian processes are usually considered probability distributions for functions, and since realizations are draws from such distributions they behave like functions once they are created. The call `f(x)` returns values just like the call `cos(x)`.

In order to provide this appealing behavior, realizations create internal copies of their mean and covariance functions upon creation. Denote by x the concatenation of all unique values at which f has been evaluated already (x initially has length 0), and by f_x the corresponding return values. The return value for the call `f(y)` is generated as follows:

- x is searched for elements of y. If any are found, the corresponding return values are recorded. Denote by y_* the elements of y that are not in x.
- The internal covariance's `observe` method is called with mesh y_* and variance 0.

- The internal mean is evaluated at y_*.
- Recall that the third return value of `Covariance.observe`, $U_{o_n|o_p}$, gives the Cholesky factor of the covariance of $f(y_*)|f(x)$. This Cholesky factor is used to generate values for $f(y_*)$.
- The internal mean's `observe` method is called with mesh y_*, values $f(y_*)$, and the internal covariance.
- The values $f(y_*)$ are returned.

Note that even if the Cholesky factor is low-rank, no cross-validation is necessary because any nearly-determined values of $f(y_*)$ are known a priori to fall close to their mean.

4.2.4.1 GP realizations with basis covariances

Calls to GP realizations will get slower as the length of o_p increases. One of the most appealing features of the `BasisCovariance` class is that it prevents this degradation from happening. However, the tradeoff is that the expense of calls tends to increase exponentially with spatial dimension.

If a realization is initialized with a `BasisCovariance` instance, it draws values for its coefficients immediately (the covariance matrix of the coefficients is factorized using an incomplete Cholesky decomposition in case it happens to be low-rank). Return values for subsequent calls to the realization are computed by simply evaluating the mean function and basis functions. Arguments are not checked for repeat calls.

4.2.5 Purpose of the basic objects

The contributions of the package up to this point are tangible implementations of means, covariance functions and realizations. Python can be run interactively, so users can manipulate, inspect and plot these objects much as they might inspect and alter the elements of a matrix in MATLAB. Some representative output from such experimentation is shown in figure 4.1.

Because these objects strongly resemble the mathematical objects they represent, they will help users build intuition about Gaussian processes themselves. Random Realizations' documentation [75] provides many examples and an introduction to the effects of the covariance function on the amplitude, wiggliness and smoothness of realizations, and on the spatial propagation of information from normally-distributed observations.

4.3 Incorporating Gaussian processes in larger probability models

The mean, covariance and realization objects described above can be incorporated into statistical models that go beyond simple nonparametric regression. There are many inferential problems in ecology and other fields where Gaussian process priors make sense that cannot be cast as simple nonparametric regressions. Random Realizations includes objects that allow Gaussian processes to be mixed into probability mod-

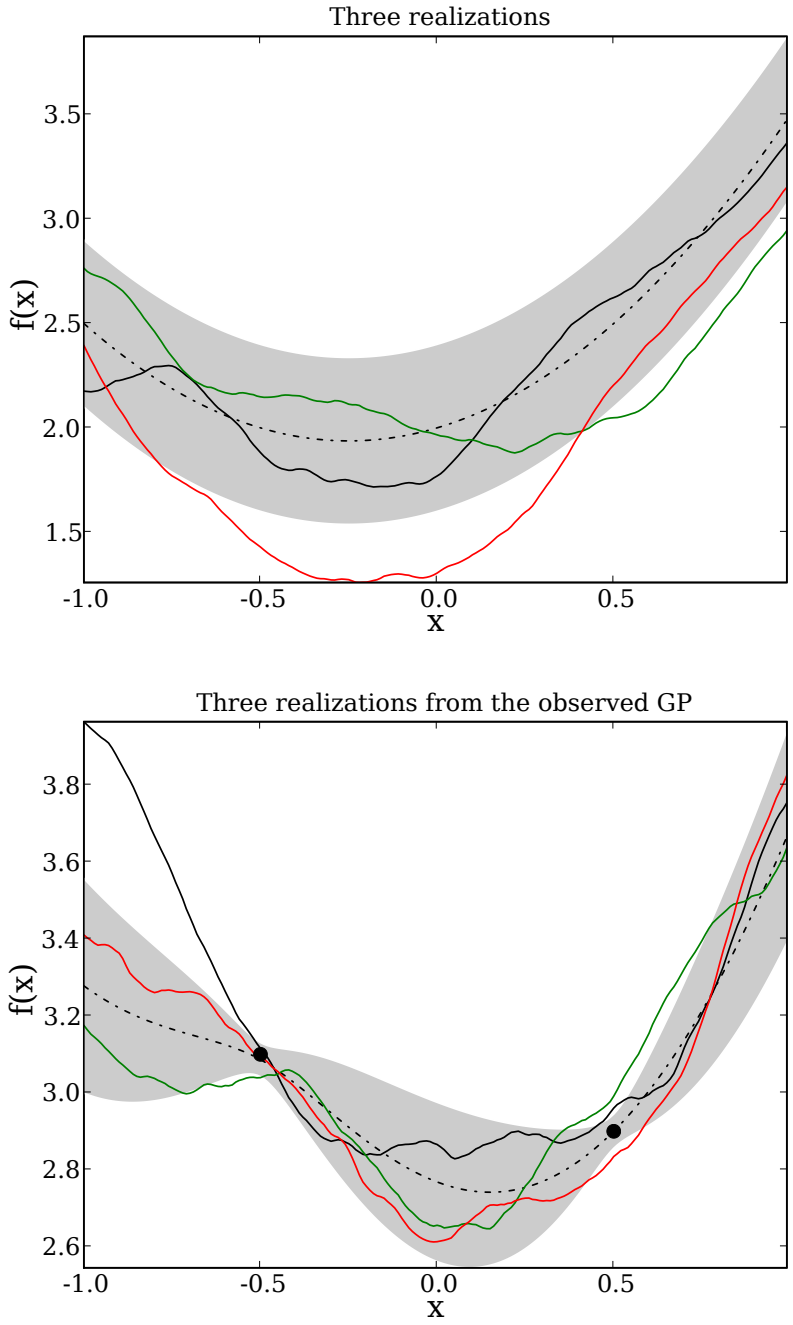


Figure 4.1: Three realizations from a Gaussian process with a Matèrn covariance function displayed with mean and one standard envelope with no observations (left) and two observations (right).

els involving more mundane random variables specified using the open-source Markov Chain Monte Carlo package PyMC (`code.google.com/p/pymc`). PyMC is implemented in Python like Random Realizations, and has a decentralized approach to model-fitting that is designed to encourage such extensions.

This section begins with a brief desccription of the basic building blocks of PyMC. These are:

- The `Parameter` class, representing random variables (stochastic nodes in WinBugs).
- The `Node` class, representing variables whose values are determined by the values of their parents (logical nodes in WinBugs).
- The `SamplingMethod` class, which is responsible for making `Parameter` instances or groups of them take single MCMC steps. This section describes the most basic type of sampling method, called `Metropolis`, but of course many others are possible.

The relationships between the objects comprising a PyMC probability model and their sampling methods is illustrated in figure 4.2.

The latter part of this section describes the objects by means of which realizations, means and covariances can be incorporated into PyMC probability models:

- The `GPParameter` class, representing a random variable whose value is a GP realization. This class is necessary because the basic `Parameter` class is not well-suited

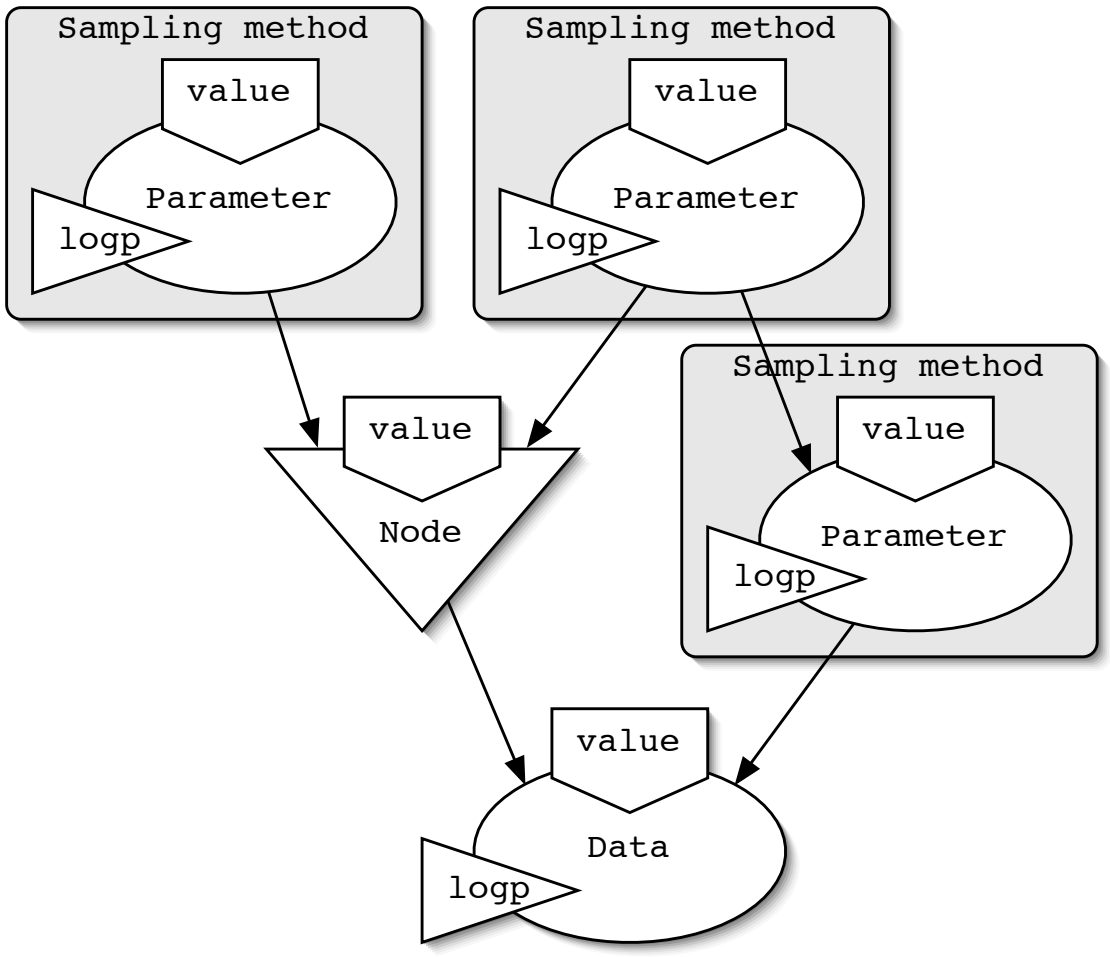


Figure 4.2: A cartoon of the relationship between PyMC's basic objects and their attributes. Sampling methods are not assigned to deterministic variables (instances of the `Node` class) or to random variables that have been observed (instances of the `Parameter` class that are flagged as data). Arrows point from parent to child.

for infinite-dimensional variables.

- Two special sampling methods, `GPMetropolis` and `GPGibbs`, which handle `GPParameter` instances.
- One additional sampling method, `GPParentMetropolis`. This sampling method handles parents of `GPParameter` instances by modifying default `Metropolis` instances.

As in previous sections the objects' internal workings are presented primarily using pseudocode, and any Python code should be sufficiently brief to be read even by non-Python programmers.

4.3.1 Random variables

Instances of the `Parameter` class represent random variables. A normally-distributed random variable B could be created with initial value 0 as follows:

```
@parameter
def B(value=0, mu=1, V=1.5):
    return -1/2 * log(det(V)*2*pi) - 1/2 * (value-mu) ** 2 / V
```

This declaration essentially specifies a function that computes the log-density of B from its value and its parents' values, then converts that function into a `Parameter` instance by means of the decorator '`@parameter`'.

The parents of random variables may be other variables (either random or determined) in addition to constants, for example:

```
@parameter
def A(value=0, mu=B, V=1):
    return -1/2 * log(det(V)*2*pi) - 1/2 * (value-mu) ** 2 / V.
```

A random variable has the following notable attributes:

- `value`: The current value of the variable. Because Python is dynamically typed, there is no restriction whatsoever on what this value can be. In most cases it will be a number or array, but it can be any object.
- `logp`: When the variable's log-probability attribute is accessed, it checks its value and its parents' values against a cache and recomputes its log-probability or log-density if necessary.
- `parents`: The parents of the variable, stored with labels (in a Python dictionary). B's parents would be {`mu: 1, V: 1.5`} and A's would be {`mu: B, V: 1.5`}.
- `children`: The children of the variable (all variables claiming the variable as a parent). These are stored without labels or any particular ordering (in a Python set). B's only child would be A, and A has no children

4.3.2 Deterministic variables

Instances of the `Node` class represent deterministic variables. Such variables generally have at least some parents that are variables (random or deterministic). A variable y that depends on another variable B as $y = B^3$ could be created as follows:

```
@node
def y(base=B, pow=3):
    return base ** pow
```

This declaration essentially specifies a function that computes the value of y from its parents' values, then converts that function into a `Node` instance.

Determined variables have the following notable attributes:

- `value`: When a determined variable's value is accessed, it checks its parents' values against a cache and recomputes its value if necessary. Again, because Python is dynamically typed this can be anything. In fact, the easiest way to incorporate mean and covariance objects into PyMC probability models is via the `Node` class, for instance:

  ```
  @node
  def C(amp=a, scale=s):
      return Covariance(gaussian, amp, scale)
  ```

  where `gaussian` is the function from section 4.2.2. An obvious exception to this rule would occur when a realization-valued random variable is used as a mean function.

- `parents`, `children`: These attributes are the same as the corresponding attributes of `Parameter`.

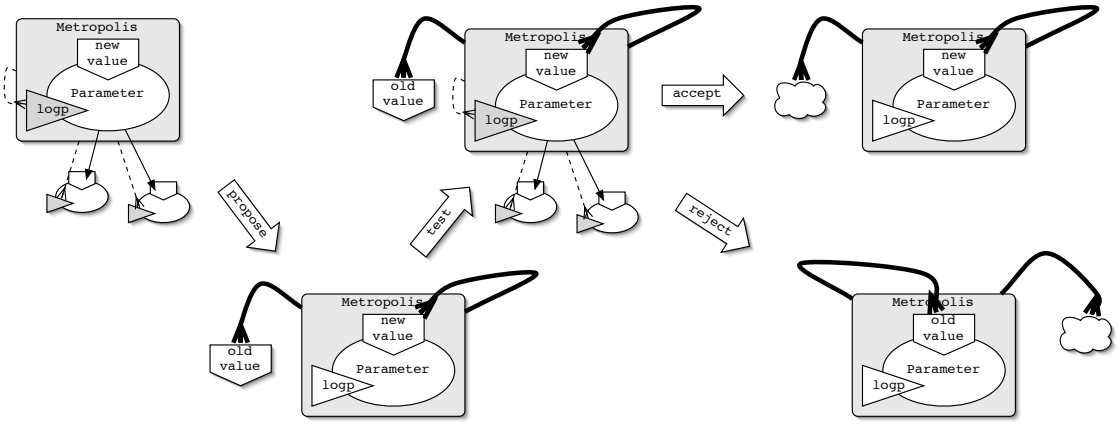


Figure 4.3: The action of a Metropolis sampling method when its `step` method is called. First, the parameter's log-probability and the log-probabilities of its children are recorded and summed. Then, the parameter's value is replaced with a proposed value. Then, the parameter's log-probability and the log-probabilities of its children are queried and summed with the proposed value in place. If the jump is accepted, the old value is forgotten; if the jump is rejected, the proposed value is replaced with the old value and forgotten.

4.3.3 Sampling methods

Sampling methods are objects (not methods in the sense that `Covariance.cholesky` is a method) that are responsible for making random variables or groups of random variables take single MCMC steps of some kind. Each sampling method has a method called `step` which essentially replaces the value of the random variable or variables it handles with a new value according to some algorithm.

The simplest sampling method is the Metropolis step, implemented by class `Metropolis`. In addition to `step`, Metropolis sampling methods have a method called `propose` which assigns proposed values to the variables they handle and a method called `reject` which replaces the proposed values with the previous values. Suppose

a Metropolis sampling method S handles the random variable B. A call to S's `step` method would result in the following chain of events, illustrated as a cartoon in figure 4.3:

- B's log-probability is recorded.
- B's log-likelihood (the sum of the log-probabilities of all B's children) is recorded.
- S's `propose` method is called.
  - B's value is set to a new value.
- B's log-probability and log-likelihood are computed.
- B's proposed value is tested according to the Metropolis algorithm.
  - If the test fails, S's `reject` method is called.
    - B's value is reset to its last value.
    - The method returns.
  - If the test passes, the method returns.

Sampling methods can be created by the user. If the user fails to create a sampling method to handle a parameter, a default sampling method is assigned to the parameter as follows:

- All available sampling method classes inspect the parameter and report their competences to handle it, on a scale of 0 to 1.

- An instance of the class reporting the highest competence is assigned to the parameter.

4.3.4 GP realization-valued random variables

`GPParameter` objects are special random variables whose value attributes are `Realization` instances. A realization-valued parameter f could be created as follows:

```
f = GPParameter(M, C, mesh=m)
```

The third initialization argument `mesh` is an array of values, which will be incorporated into f as the read-only attribute `mesh`.

Realization-valued parameters have log-probability attributes like any random variable, but of course it would usually be expensive and difficult to assign something like a log-density to an object representing an infinite-dimensional random variable. The log-probability attribute instead gives $p(f(m))|\texttt{M,C}$. In words, the log-probability attribute only returns the log-probabilty of f's value's evaluation on its mesh.

4.3.5 Sampling methods for realization-valued random variables and their parents

The a price of the 'cop-out' of computing realization-valued parameters' log-probabilities based only on their evaluation on a mesh is that the Metropolis-Hastings algorithm no longer applies. `GPMetropolis` and `GPParentMetropolis` employ a strategy that will be described here. Denote by $\tilde{f}$ the evaluation of f everywhere except on m. Denote the parents of f by P and its 'extended children', or the set of parameters

descended from it either directly or via an unbroken sequence of deterministic variables, by K.

A very important point is that neither of the sampling methods in this subsection care at all about how K depends on f. This means that, like the standard Metropolis algorithm in more mundane probability models, they can theoretically be applied in an enormous range of probability models that involve Gaussian processes. In practice, of course, novel sampling methods will often be desirable to speed convergence. I hope authors of such sampling methods will share their objects by making them available for distribution with Random Realizations.

4.3.5.1 A Metropolis sampling method for realization-valued RV's

The Metropolis-Hastings acceptance ratio for a proposed value f_p can be written as follows:

$$\frac{p(K|\tilde{f}_p, f_p(m))\ p(f_p(m)|P)\ q(f(m)|f_p(m), P)}{p(K|\tilde{f}, f(m))\ p(f(m)|P)\ q(f_p(m)|f(m), P)} \frac{Q(\tilde{f}|f(m), P)}{Q(\tilde{f}_p|f_p(m), P)},$$

where $q(f_p(m)|f(m), P)$ denotes the proposal density for $f(m)$ $Q(f|f(m), P)$ denotes the proposal density for $\tilde{f}$ with respect to $\tilde{f}$'s measure conditional on its parents and $f(m)$:

$$Q(\tilde{f}|f(m), P) = \frac{d\theta(\tilde{f}|f(m), P)}{d\pi(\tilde{f}|f(m), P)},$$

where the measures θ and π denote the proposal and prior measures respectively. The troublesome terms involving $\tilde{f}$ and $\tilde{f}_p$ in the consequent position can be neglected if the proposal measure for $\tilde{f}$ is set to its measure conditional on its parents and $f(m)$, so

that the density becomes 1; in other words, if $\tilde{f}$ is proposed from its prior conditional on $f(m)$. `GPMetropolis` sampling methods employ this strategy.

The evaluation $f(m)$ on the mesh is proposed from a multivariate normal distribution. By default, the covariance is a scalar multiple of $C(m, m)$, but this is tunable.

4.3.5.2 Metropolis sampling methods for parents of realization-valued RV's

Similarly, the Metropolis-Hastings acceptance ratio for a proposed value P_p of the parents *and* a proposed value $\tilde{f}_p$ for the realization evaluated off the mesh is as follows:

$$\frac{p(K|\tilde{f}_p)\ p(f(m)|P_p)\ q(P)}{p(K|\tilde{f})\ p(f(m)|P)\ q(P_p)} \frac{Q(\tilde{f}|f(m), P)}{Q(\tilde{f}_p|f(m), P_p)}$$

If the same proposal distribution as above is used for $\tilde{f}$, the intractable term again does not have to be computed. In other words, every time a value is proposed for one of f's parents, a value is proposed for f itself conditional on its evaluation on its mesh.

To implement this strategy, `GPParentMetropolis` modifies a host `Metropolis` instance. It replaces its host sampling method's `propose` and `reject` methods with special methods that call the host's native methods, and then propose and reject values for $\tilde{f}$ conditional on $f(m)$. `GPParentMetropolis` also modifies the host sampling method to include the likelihood term $p(K|\tilde{f})$ in the acceptance ratio.

This scheme works if K depends on P and if P has extended children other than f, but for clarity these possibilites are not included in the acceptance ratio above.

`GPParentMetropolis` can wrap Metropolis-Hastings sampling methods other than the simple Metropolis sampling method if desired.

4.3.5.3 Choosing a mesh

If a GP-valued random variable is assigned an empty mesh m, its value will be proposed from its prior, and rejection rates are likely to be quite large. If the mesh is too dense, on the other hand, computation of the log-density will be expensive, as it scales as the cube of the number of points in the mesh. Several choices for mesh density and some corresponding proposals $\tilde{f}_p$ are shown in figure 4.4.

If the mesh is very dense, the Markov chain may occasionally explore values for the parameters of the covariance C for which $C(m, m)$ is numerically singular. When `GPParentMetropolis` instances encounter this situation, they reject the jump and optionally display a warning. A more correct approach would be to simply thin the mesh and continue sampling, but I have not implemented this extension.

4.3.6 Gibbs sampling methods for realization-valued random variables

A realization-valued random variable's full conditional distribution [29] is a Gaussian process if its Markov blanket [41] is described by the following probability model:

$$K_i | f \sim \mathrm{N}(f(o_i), V_i) \quad i = 1 \ldots N$$

$$f | M, C \sim \mathrm{GP}(M, C)$$

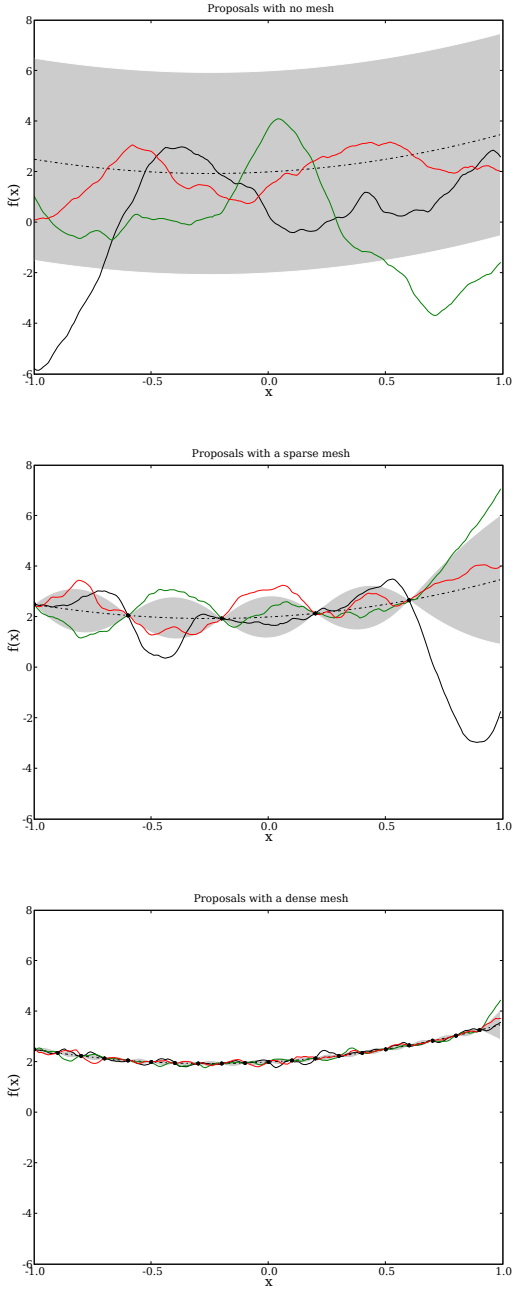


Figure 4.4: Several possible proposals of $\tilde{f}$ (curves) given proposed values for $f(m)$ (heavy dots) with no mesh (**top**), a sparse mesh (**middle**), and a dense mesh (**bottom**). Proposal distributions' one-standard-deviation envelopes are shown as shaded regions, with means shown as broken lines. With no mesh, $\tilde{f}$ is being proposed from its prior and the acceptance rate will be very low. A denser mesh permits a high degree of control over $\tilde{f}$, but computing the log-probability will be more expensive.

In other words, each of its children K_i is normally distributed with variance V_i and mean equal to $f(o_i)$, where each o_i is an array of input values. In this case, f can be handled by a `GPGibbs` sampling method. These sampling methods maintain internal copies of M and C, observed at input values $\{o_i\}$ with output values $\{K_i\}$ and variance $\{V_i\}$.

An illustration of the respective performance of `GPMetropolis` and `GPGibbs` for a model of the form

$$\begin{aligned} d|f &\sim \mathrm{N}(f(x), V) \\ f|M, C &\sim \mathrm{GP}(M, C) \\ M|x &= ax^2 + bx + c \\ C|x, y &= \text{Matèrn}(|x - y|, \nu, \phi, \alpha) \\ \nu, \phi, \alpha, a, b, c, V &\sim \text{priors} \end{aligned} \tag{4.5}$$

with only two datapoints is shown in figure 4.5. 'Matèrn' indicates the Matèrn covariance function [11].

4.4 Discussion

The following list of the objects with summaries of users' interaction with them recapitulates the contribution of this chapter:

- `Mean` is a callable object, like a simple one-place function.
- `Covariance` is a callable object, like a simple two-place function. Its return value is a matrix. It can efficiently compute and return the Cholesky factorization or

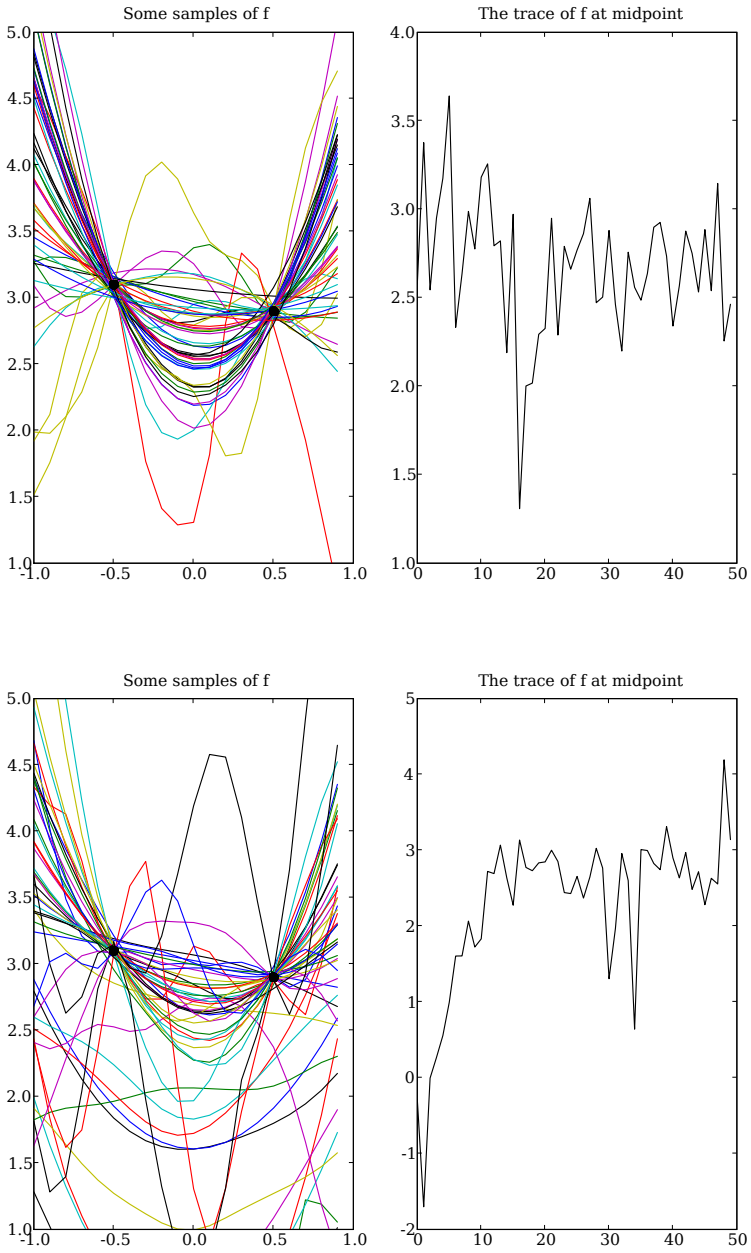


Figure 4.5: The output of an MCMC run for the model (4.5) using the `GPGibbs` (**top**) and `GPMetropolis` (**bottom**) sampling methods. Several samples are shown in the **left** panels, and the dynamic traces of the `GPParameter`'s evaluation at the midpoint of the interval are shown in the **right** panels. Both the Metropolis and Gibbs samples' jumping distribution is modulated by the value of the prior parameters, which are updated by `GPParentMetropolis` sampling methods.

diagonal of that matrix instead, if desired.

- `Realization` is a callable object representing a draw from a Gaussian process. It can be created by typing `f = Realization(M, C)`. Once created, it behaves like a simple one-place function.

- `GPParameter` is a random variable valued as a GP realization. It has a `value` attribute and a `logp` attribute, like an ordinary PyMC `Parameter` object. An enormous variety of probability models involving Gaussian processes can be easily constructed using this object.

- `GPMetropolis`, `GPParentMetropolis` and `GPGibbs` are sampling methods responsible for implementing Metropolis and Gibbs steps involving GP realization-valued parameters and their parents. Users often do not have to interact with the first two, because they are assigned and tuned automatically by PyMC.

`Mean` and `Covariance` can be modified by the function `observe`, which 'imposes' normally-distributed observations on them.

The three basic objects, `Mean`, `Covariance` and `Realization`, resemble the concepts they represent very closely; as callable objects, they are as close to mathematical functions as programming language features can be. Because they can be created, manipulated and plotted in interactive Python sessions, I hope they prove to be of value as teaching and learning tools for building intuition about Gaussian processes without making lengthy excursions into linear algebra.

The other objects allow for Gaussian processes to be mixed into nearly arbitrary probability models. Using them, users can use the basic objects that have served them as models of Gaussian process-related concepts directly in statistics with ease comparable to that provided by WinBugs. The standard caveats of MCMC algorithms apply, of course; some models will be prohibitively expensive to fit, and some will have very poor mixing properties unless treated with specialized sampling methods. However, I hope that the decentralization and extensibility of PyMC's model-fitting approach eventually fosters the growth of a library of such sampling methods.

Chapter 5

Conclusion

5.1 Contributions of this dissertation

Chapter 2 presents a synthesis of Bayesian nonparametric methods and recent developments in inference of diffusion trajectories and applies it to inference of the functional response, an old problem in multispecies dynamics. This combination has widespread potential in ecology, especially if more attention is paid to the scaling of methods for inference of multivariate diffusions in the rapid progress that is currently being made in that area [13, 14]. The method seems successful in broad terms and reveals several qualitatively interesting features. Its primary drawbacks are difficulty of implementation and computational expense.

Chapter 3 explores the issues associated with acknowledging functional uncertainty via nonparametric priors in fishery management. Forecasting and optimization become more difficult, but sensible management based on local considerations still seems

feasible. In contrast to chapter 2, this chapter makes use of a very simple, quick-and-dirty statistical framework that gives rise to very cheap computations. This framework is too assumption-bound to be dependable for inference, but it is useful as a meta-model for 'mocking up' more sophisticated models and exploring the inferential issues that are likely to arise.

Chapter 4 presents a software package that has two purposes: to serve as a graphically-oriented Gaussian process teaching and learning tool, and to help users construct and fit probability models of their own devising that involve Gaussian processes. To accomplish the second goal, the package dovetails with an open and modular package for general-purpose Bayesian model specification and MCMC. Ideally this combination will help methodological researchers to share their work with ecologists in a usable yet non-restrictive way.

The potential applications of Bayesian nonparametrics to inference of ecological dynamics are just beginning to unfold, so I cannot claim that this dissertation has sampled the possibilities in any representative way. However, I hope it helps catalyze the development of new areas of application.

5.2 Two issues that need to be addressed

At first blush, Bayesian inversion of dynamical models with nonparametric priors for unknown functions may seem like a universal remedy for model uncertainty. My expectations were very high when I learned to see Bayesian statistics as a framework

for inverse problems, and they grew still more upon the discovery that people actually knew how to infer whole functions from data. In light of this experience, it seems appropriate to point out what I believe will be two of the most difficult issues for ecological modelers to resolve regarding the appropriate application and interpretation of such techniques.

The first issue I would like to raise is the interpretation of simple dynamical models embedded in a Bayesian framework, nonparametric or otherwise. May [56] framed the dilemma of model complexity in terms of the goal of the model: if the model is meant to quantitatively predict events in the real world, it should be sufficiently complex; whereas if the model is meant to help the modeler develop understanding, it should be kept simple as possible without failing to represent the dynamical features under consideration.

Although dynamical models may or may not be quantitatively predictive, Bayesian statistical models are predictive by nature [40]. For example, the function of the statistical model from chapter 2

$$\begin{aligned} \mathrm{x}_i &\dot{\sim} \mathrm{N}(x_{t_i}, (13/12)x_{t_i}) \\ \mathrm{y}_i &\dot{\sim} \mathrm{N}(y_{t_i}, (13/12)y_{t_i}) \\ dx_t &= \left[r_x x_t - \frac{x_t^2}{m} - f(x_t, y_t) \right] dt + \sigma_x dW_x \\ dy_t &= \left[r_y y_t + \beta f(x_t, y_t) \right] dt + \sigma_y dW_y \\ f &\sim \mathrm{GP}(0, C) \end{aligned}$$

$$\frac{1}{\phi^2}, \sigma_x, \sigma_y, x_0, y_0 \sim \text{priors}$$

is to produce a predictive distribution for observations $\{\mathrm{x}, \mathrm{y}\}$ from knowledge of the unobservable parameters. Bayes' theorem gives a recipe for converting such predictive models into posterior distributions when actual observations arrive.

The meaning of parameter estimates for non-predictive models that are made using model predictions is unclear. Cushing et al. [19] point out one possible avenue for resolution when they suggest that even simple dynamical models can be compared to data when made stochastic in appropriate ways.

The second issue is specification of appropriate nonparametric priors for rate functions. Although there are many ways to slice the process of dynamical model formulation, the following two-stage representation is useful for highlighting this issue:

1. Choice of a modeling framework.

2. Parametrization.

Weisberg [91], who carefully examines this process, calls the first stage selection of a 'model description' and the second stage 'instantiation' of the model description, the result of which is an actual model.

Decisions regarding the general orientation of a model and the appropriate level of complexity are made at the first stage, and many authors have discussed the associated issues. The second stage is generally considered much simpler: parameters that can be known ahead of time are specified, and the remaining parameters are either estimated from data or allowed to take a range of values.

Nonparametric thinking moves the process of choosing functional forms from the 'general framework' stage of model selection to the 'parametrization' stage. This is an attractive feature, as it encourages modelers to test modeling frameworks against one another rather than simply testing representative suites of models chosen from each framework.

Just as Bayesian inference of scalars can be a useful tool for parametrizing models, Bayesian nonparametrics can free modelers from the need to specify forms for all functions. However, nonparametrics is not a free lunch; specification of appropriate priors for rate functions is a problem that needs to be taken seriously. The standard Gaussian process covariance functions were developed for geostatistical applications and may not yield the priors of choice in dynamical settings. For example, assuming for the sake of argument that the transmission function investigated by Wood and Thomas [95] (section 1.2) really could approach the origin as a radical and that simply using a power-law mean function with a hyperprior were unsatisfactory, a novel class of covariance functions would have to be developed.

Specifying functional priors will not come naturally. While modelers routinely state several desired qualities for functions such as monotonicity and concavity, they rarely undertake the task of delimiting *all* functions that could reasonably play a particular dynamical role, let alone attempt to assign relative plausibilities to them. Mixture priors will likely prove to be helpful, especially once the computational hurdles get slightly lower.

Bayesian nonparametrics currently offers several priors for unknown functions

in addition to Gaussian processes. Møller and Waagepetersen [63] review several of these in chapter 5, particularly Gamma process-based superposition priors, the context of Cox processes. Müller, West and MacEachern [64] present a class of priors for transition densities that could be adapted to inference of unknown functions, and in discrete-time models such priors would permit sophisticated and flexible modeling of process uncertainty associated with them. The relative appropriateness of these priors and Gaussian processes for unknown functions in dynamical models in ecology has yet to be investigated.

Bibliography

[1] Petter Abrahamsen. A review of Gaussian random fields and correlation functions. Technical Report 917, Norwegian Computing Center, 1997.

[2] Milton Abramowitz and Irene A. Stegun, editors. *Handbook of mathematical functions*. Dover Publications, Inc., New York, 1972.

[3] P. A. Abrams. The fallacies of "ratio-dependent" predation. *Ecology*, 75(6):1842–1850, September 1994.

[4] P. A. Abrams. The evolution of predator-prey interactions: theory and evidence. *Annual Review of Ecology and Systematics*, 31:79–105, 2000.

[5] P. A. Abrams and L. R. Ginzburg. The nature of predation: prey dependent, ratio dependent, or neither? *TREE*, pages 337–341, 2000.

[6] Yacine Aït-Sahalia. Closed-form likelihood expansions for multivariate diffusions. *Annals of statistics (forthcoming)*, 2007.

[7] H. R. Akcakaya, R. Arditi, and L. R. Ginsburg. Ratio-dependent predation: an abstraction that works. *Ecology*, 76:995–1004, 1995.

[8] R. Arditi and L. R. Ginzburg. Coupling in predator-prey dynamics: Ratio-dependence. *Journal of Theoretical Biology*, 139:311–326, 1989.

[9] Deborah J. Armstrong. The quarks of object-oriented development. *Communications of the ACM*, 49(2), 2006.

[10] Francis R. Bach and Michael I. Jordan. Predictive low-rank decomposition for kernel methods. In *Proceedings of the 22nd International Conference on Machine learning*, 2005.

[11] Sudipto Banerjee, Bradley P. Carlin, and Alan Gelfand. *Hierarchical modeling and analysis for spatial data.* Monographs on Statistics and Applied Probability. Chapman & Hall CRC, New York, 2004.

[12] J. R. Beddington. Mutual interference between parasites or predators and its effect on searching efficiency. *Journal of Animal Ecology*, 44:331–340, 1975.

[13] Alexandros Beskos, Omiros Papaspiliopolous, Gareth O. Roberts, and Paul Fearnhead. Exact and computationally efficient likelihood-based estimation for discretely-observed diffusion processes. *Journal of the Royal Statistical Society Series B*, 68:333–382, 2006.

[14] Alexandros Beskos, Gareth O. Roberts, Andrew M. Stuart, and Jochen Voss. An mcmc method for diffusion bridges. Available from url.

[15] J. F. Caddy. Current usage of fisheries indicators and reference points, and their

potential application to management of fisheries for marine invertebrates. *Canadian Journal of Fisheries and Aquatic Sciences*, 61:1307–1324, 2004.

[16] J. F. Caddy and Mahon. R. Reference points for fishery management. Fisheries Technical Paper 347, FAO, 1995.

[17] Jeremy S. Collie and Henrik Gislason. Biological reference points for fish stocks in a multispecies context. *Canadian Journal of Fisheries and Aquatic Sciences*, 58:2167–2176, 2001.

[18] E. P. Creaser and H. C. Perkins. The distribution, food, and age of juvenile bluefish, *Pomatomus saltatrix*, in maine. *Fishery Bulletin*, 92:494–508, 1994.

[19] J. M. Cushing, R. F. Constantino, Brian Dennis, R. A. Desharnais, and Shandelle M. Henson. Nonlinear population dynamics: models, experiments and data. *Journal of Theoretical Biology*, 194:1–9, 1998.

[20] D. Dacunha-Castelle and D. Florens-Zmirou. Estimation of the coefficients of a diffusion from discrete observations. *Stochastics*, 19:263–284, 1986.

[21] D. L. DeAngelis, R. A. Goldstein, and R. V. O'Neill. A model for trophic interaction. *Ecology*, 56:881–892, 2002.

[22] J. J. Dongarra, C. B. Moler, J. R. Bunch, and G. W. Stewart. *Linpack user's guide*. SIAM, 1979.

[23] O. Elerian, S. Chib, and N. Shephard. Likelihod inference for discretely observed nonlinear diffusions. *Econometrica*, 69(4):959–993, 2001.

[24] S. P. Ellner. Inferring mechanism from time-series data: Delay-differential equations. *Physica D.*, 110:182–194, 1997.

[25] S. P. Ellner, Y. Seifu, and R. H. Smith. Fitting population dynamic models to time-series data by gradient matching. *Ecology*, 83:2256–2270, 2002.

[26] Shai Fine and Katya Scheinberg. Efficient SVM training using low-rank kernel representations. *Journal of Machine Learning Research*, 2:243–264, 2001.

[27] Andrew Gelman, John B. Carlin, Hal S. Stern, and Donald B. Rubin. *Bayesian Data Analysis*. Texts in Statistical Science. Chapman & Hall / CRC, New York, 2 edition, 2004.

[28] Roger G. Ghanem and Pol D. Spanos. *Stochastic finite elements: a spectral approach*. Dover Publications, 1991.

[29] W. R. Gilks, S. Richardson, and D. Spiegelhalter, editors. *Markov chain Monte Carlo in practice*. Chapman & Hall / CRC, 1995.

[30] L. R. Ginzburg and H. R. Akçakaya. Consequences of ratio-dependent predation for steady-state properties of systems. *Ecology*, 73:1536–1543, 1992.

[31] Narendra S. Goel and Nira Richter-Dyn. *Stochastic models in biology*. Academic Press, 1975.

[32] Gene H. Golub and Charles F. van Loan. *Matrix computations*. The Johns Hopkins University Press, third edition, 1996.

[33] G. W. Harrison. Comparing predator-prey models to Luckinbill's experiment with *Paramecium* and *Didinium*. *Ecology*, 76:357–374, 1995.

[34] M. P. Hassell and G. C. Varley. New inductive population model for insect parasites and its bearing on biological control. *Nature*, 223:1133–1137, 1969.

[35] Mikko Heino. Management of evolving fish stocks. *Canadian Journal of Fisheries and Aquatic Sciences*, 55:1971–1982, 1998.

[36] R. Hilborn and C. J. Walters. *Quantitative fisheries stock assessment: choice, dynamics and uncertainty*. Springer, 1991.

[37] C. S. Holling. Some characteristics of simple types of predation and parasitism. *The Canadian Entomologist*, 91:385–398, 1959.

[38] Anne B. Hollowed, Nicholas Bax, Richard Beamish, Jeremy Collie, Michael Fogarty, Patricia Livingston, John Pope, and Jake C. Rice. Are multispecies models an improvement on single-species models for measuring fishing impacts on marine ecosystems? *ICES Journal of Marine Science*, 57:707–719, 2000.

[39] W. Horsthemke and R. Lefever. *Noise-induced transitions: applications in physics, chemistry and biology*. Springer series in synergetics. Springer, 1984.

[40] E. T. Jaynes. *Probability theory: the logic of science*. Cambridge University Press, 2003.

[41] Finn V. Jensen. *Bayesian networks and decision graphs*. Springer, 2001.

[42] J. M. Jeschke, M. Kopp, and R. Trollrian. Predator functional responses: Discriminating between handling and digesting prey. *Ecological Monographs*, 72:95–112, 2002.

[43] C. M. Jessop, R. Kassen, S. E. Forde, B. Kerr, A. Buckling, P. B. Rainey, and B. J. M. Bohannan. Big questions, small worlds: microbial model systems in ecology. *TREE*, 19:189–197, 2005.

[44] C. Jost and R. Arditi. From pattern to process: Identifying predator-prey models from time-series data. *Population Ecology*, 43:229–243, 2001.

[45] Christian Jost and Stephen P. Ellner. Testing for predator dependence in predator-prey dynamics: a non-parametric approach. *Proceedings of the Royal Society of London B*, 267:1611–1620, 2000.

[46] Francis Juanes, J. A. Hare, and A. G. Miskiewicz. Comparing early life history strategies of *Pomatomus saltatrix*: A global approach. *Marine and freshwater research*, 47(2):365–379, 1996.

[47] Sharon Kingsland. *Modeling Nature*. The University of Chicago Press, second edition, 1995.

[48] G. P. Kirkwood. Allowing for risks in setting catch limits based on MSY. *Mathematical Biosciences*, 53:119–129, 1981.

[49] V. Kolchin, B. Sewast'yanov, and V. Chistyakov. *Random allocations*. Wiley & sons, 1977.

[50] Stefan Lang and Andreas Brezger. Bayesian P-splines. *Journal of Computational and Graphical Statistics*, 13(1):183–212, 2004.

[51] Herbert K. H. Lee, Dave M. Higdon, Catherine A. Calder, and Christopher H. Holloman. Efficient models for correlated data via convolutions of intrinsic processes. *Statistical Modelling*, 5(1):53–74, 2005.

[52] Richard Levins. The strategy of model building in population biology. *American Scientist*, 54:421–431, 1966.

[53] A. J. Lotka. *Elements of physical biology*. Williams and Wilkins, Baltimore, 1925.

[54] Pamela M. Mace. A new role for MSY in single-species and ecosystem approaches to fisheries stock assessment and management. *Fish and fisheries*, 2:2–32, 2001.

[55] Marc Mangel, Baldo Marinovic, Caroline Pomeroy, and Donald Croll. Requiem for Ricker: unpacking MSY. *Bulletin of marine science*, 70(2):763–781, 2002.

[56] Robert M. May. Stability and complexity in model ecosystems. In *Landmarks in Biology*. Princeton University Press, 2001.

[57] Robert M. May, J. R. Beddington, J. W. Horwood, and J. G. Shepherd. Exploiting natural populations in an uncertain world. *Mathematical Biosciences*, 42:219–252, 1978.

[58] Richard Methot and Kevin Piner. Status of the canary rockfish resource off california, oregon, and washington in 2001. Technical report, Pacific Fishery Management Council, 26 April 2001.

[59] Georghe Micula and Sanda Micula. *Handbook of Splines*. Kluwer Academic, 1999.

[60] R. B. Millar and R. Meyer. Non-linear state space modelling of fisheries biomass dynamics by using Metropolis-Hastings within-Gibbs sampling. *Journal of the Royal Statistical Society Series C*, 49:327–342, 2000.

[61] G. N. Milstein et al. Transition density estimation for stochastic differential equations via forward-reverse representations. *Bernoulli*, 10(2):281–312, 2004.

[62] Grigori. N. Milstein and Michael V. Tretyakov. *Stochastic numerics for mathematical physics*. Springer-Verlag, Berlin, 2004.

[63] Jesper Møller and Rasmus Plenge Waagepetersen. *Statistical inference and simulation for spatial point processes*. Monographs on Statistics and Applied Probability. Chapman & Hall / CRC, 2003.

[64] Peter Müller, Mike West, and Steven MacEachern. Bayesian models for non-linear autoregressions. *Journal of Time Series Analysis*, 18(6), 1997.

[65] M. Munch, A. Kottas, and Mangel S. Bayesian nonparametric analysis of stock-recruitment relationships. *Canadian Journal of Fisheries and Aquatic Sciences*, 62:1808–1821, 2005.

[66] S. A. Murawski, P. J. Rago, and E. A. Trippel. Impacts of demographic variation in spawning characteristics on reference points for fishery management. *ICES Journal of Marine Science*, 58:1002–1014, 2001.

[67] C. L. Needle. Recruitment models: diagnosis and prognosis. *Reviews in Fish Biology and Fisheries*, 11:95–111, 2002.

[68] NEFSC. 41st northeast regional stock assessment workshop assessment report. Technical report, NEFSC, 2005.

[69] J. Nicolau. A new technique for simulating the likelihood of stochastic differential equations. *Econometrics Journal*, 5:91–103, 2002.

[70] R. M. Nisbet and W. S. C. Gurney. *Modeling fluctuating populations.* Wiley, New York, 1982.

[71] NMFS. Magnuson-Stevens fishery conservation and management act. Technical Report NMFS-F/SPO-23, NOAA Technical Memorandum, 1996 1996.

[72] Hilary Ockendon. *Viscous flow.* Cambridge University Press, 1995.

[73] Travis E. Oliphant. *Guide to NumPy.* Trelgol Publishing, 2006.

[74] Christopher J. Paciorek and Mark J. Schervish. Nonstationary covariance functions for gaussian process regression. In *NIPS conference proceedings*, 2004.

[75] Anand Patil. Random Realizations user's guide. Available from http://code.google.com/p/gaussian-process.

[76] Terrance J. Quinn and Richard B. Deriso. *Quantitative fish dynamics.* Oxford University Press, 1999.

[77] V. R. Restrepo, G. G. Thompson, P. M. Mace, W. L. Gabriel, L. L. Low, and A. D. MacCall. Technical guidance on the use of precautionary approaches to implementing national standard 1 of the magnuson-stevens fishery conservation and management act. Technical Report NMFS-F/SPO-31, NOAA Technical Memorandum, 1998.

[78] W. E. Ricker. *Computation and interpretation of biological statistics of fish populations*. Number 191 in Bulletin (Fisheries Reserves Board of Canada). Department of Environment, Fisheries and Marine Service, Ottawa, Ontario, Canada, 1975.

[79] G. O. Roberts and O. Stramer. On inference for partially observed nonlinear diffusion models using the metropolis-hastings algorithm. *Biometrika*, 88(3):603–621, 2001.

[80] P. D. Sampson and Peter Guttorp. Nonparametric estimation of nonstationary spatial covariance structure. *Journal of the American Statistical Association*, 87(417):108–119, 1992.

[81] Matthias Seeger. Low rank updates for the Cholesky decomposition. Available from http://www.kyb.tuebingen.mpg.de/bs/people/seeger/papers/cholupdate.pdf.

[82] Steven E. Shreve. *Stochastic calculus for finance: Continuous time models*. Springer, 2004.

[83] William Silvert. Object-oriented ecosystem modelling. *Ecological Modelling*, pages 91–118, 1993.

[84] Michael P. Sissenwine. Is MSY an adequate foundation for optimum yield? *Fisheries*, 3(6):22–42, 1978.

[85] I. Stakgold. *Green's functions and boundary value problems*. Wiley-Interscience, New York, second edition, 1998.

[86] M. Stein. *Interpolation of spatial data: some theory for kriging*. Springer, New York, 1999.

[87] Steven H. Strogatz. *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering*. Perseus Books, 1994.

[88] Mark L. Taper and Subhash R. Lele. Dynamical models as paths to evidence in ecology. In Mark L. Taper and Subhash R. Lele, editors, *The nature of scientific evidence*. The University of Chicago Press, 2004.

[89] B. G. Veilleux. An analysis of the predatory interaction between *Paramecium* and *Didinium*. *The Journal of Animal Ecology*, 48(3):787–803, 1979.

[90] V. Volterra. Variazioni e fluttuazioni del numero d'individui in specie animali conviventi. *Memorie della Accademia Nazionale dei Lincei*, 2:31–113, 1926.

[91] Michael Weisberg. *When less is more: tradeoffs and idealization in model building*. PhD thesis, Stanford University, 2003.

[92] Joshua S. Weitz and Simon A. Levin. Size and scaling of predator-prey dynamics. *Ecology letters*, 9:548–557, 2006.

[93] M. West and J. Harrison. *Bayesian Forecasting and Dynamic Modeling*. Springer Series in Statistics. Springer, New York, 1997.

[94] S. N. Wood. Obtaining birth and mortality patterns from structured population trajectories. *Ecological Monographs*, 64:23–44, 1994.

[95] S. N. Wood and M. B. Thomas. Super sensitivity to structure in biological models. *Proceedings of the Royal Society B*, 266:565–570, 1999.

[96] Simon N. Wood. Partially specified ecological models. *Ecological Monographs*, 71(1), 2001.

[97] J. Wu, M. Fukuhara, and T. Takeda. Parameter estimation of an ecological system by a neural network with residual minimization training. *Ecological Modeling*, pages 289–304, 189.

Appendix A

Appendices to Chapter 2

A.1 Boundary conditions

In this section, we discuss a confusing aspect of the relationship between Gaussian process realizations and their truncated Fourier series representations. For illustration, we will consider the Gaussian covariance function

$$\mathbf{Cov}\left(g(x), g(\xi)\right) = \phi e^{-\lambda(x-\xi)^2}.$$

Figure A.1 shows zero-mean draws using a truncated Fourier sine series representation for this covariance matrix for $\lambda = 1$, $\phi = 1/16$.

The inverse-transformed draws from the Fourier transformed Gaussian process exhibit high-frequency ringing near the endpoints. The ringing at the left endpoints is the Gibbs phenomenon: nonuniform convergence of the Fourier series in the neighborhood of discontinuities. Since all of our basis functions (sines) are equal to zero at the endpoints, but direct Gaussian process realizations are not forced to be zero there,

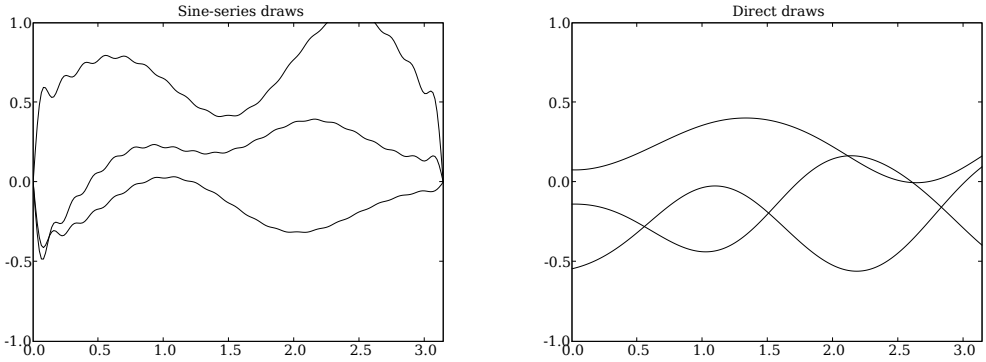


Figure A.1: Draws from a Fourier sine transformed Gaussian process distribution. Note Gibbs phenomena, which result because the covariance function does not guarantee that the boundary conditions $g(0) = 0$ and $g(\pi) = 0$ will be satisfied.

nearly every draw is discontinuous. See Stakgold [85] for analysis of Gibbs phenomena and Fourier series convergence in general.

The high-frequency ringing can be removed by conditioning on $g(0) = 0$ and $g(1) = 0$ using standard multivariate normal conditioning rules. In figure A.2, the Fourier draws closely resemble the direct draws for all values of ϕ.

A.2 Derivation of the dynamical model as a birth and death process

Denote the number of *Paramecium* by x and the number of *Didinium* by y. In a time interval dt, a given paramecium:

- Dies due to a cause other than predation with probability $dt(\mu_x + x/m_1) + o(dt)$.

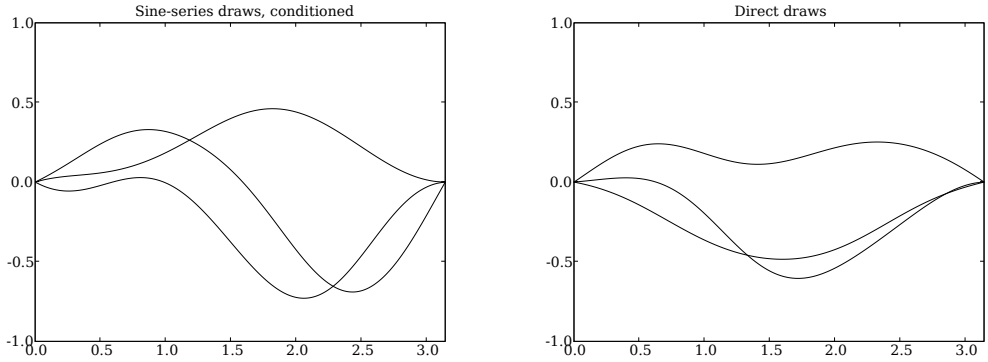


Figure A.2: Draws from a Gaussian process whose covariance function forces satisfaction of the natural boundary conditions of the basis do not exhibit Gibbs phenomena.

- Divides with probability $dt(d - x/m_2) + o(dt)$.
- Is eaten with probability $h(x, y)dt + o(dt)$.

A didinium:

- Dies with probability $\mu_y dt + o(dt)$.
- Each paramecium eaten is converted to β new didinium.

Approximating the resulting nonlinear birth-death process by a diffusion process [31] and writing it as an Itô stochastic differential equation, we have

$$\begin{bmatrix} dx \\ dy \end{bmatrix} = \begin{bmatrix} xdt(r_1 - x/k_1 - h(x,y)) \\ dt(-y\mu_y + \beta x h(x,y)) \end{bmatrix} + \sqrt{\begin{bmatrix} x(r_2 + x/k_2 + h(x,y)) & \beta x h(x,y) \\ \beta x h(x,y) & \beta x h(x,y) - \mu_y y \end{bmatrix}} \begin{bmatrix} dW_x \\ dW_y \end{bmatrix},$$

where the square root denotes any matrix square root.

The rescaling of Roberts and Stramer is not possible. Writing the SDE above as

$$d\mathbf{x} = f(\mathbf{x})dt + \sigma(\mathbf{x})d\mathbf{W}$$

left-multiplication by $\sigma(\mathbf{x})^{-1}$ yields

$$\sigma(\mathbf{x})^{-1}d\mathbf{x} = \sigma(\mathbf{x})^{-1}f(\mathbf{x})dt + d\mathbf{W}.$$

The condition we would like to check is $\sigma(\mathbf{x})^{-1}d\mathbf{x} = d\xi$. In other words,

$$\partial_x \xi_1 = \sigma(\mathbf{x})^{-1}_{11},$$

$$\partial_y \xi_1 = \sigma(\mathbf{x})^{-1}_{12},$$

$$\partial_x \xi_2 = \sigma(\mathbf{x})^{-1}_{21},$$

$$\partial_y \xi_2 = \sigma(\mathbf{x})^{-1}_{22}.$$

For h arbitrary, existence of a solution is not guaranteed.

Appendix B

Appendices to Chapter 3

B.1 Forecasting

The forecast probability distribution of future unfished biomass $B_{t+\tau}$ given $B_0 \dots B_t$ in the model described in Section 3.2 does not have a standard form for $\tau > 1$. We sample from this distribution using Monte Carlo simulations. To generate a single population trajectory, we first sample $1/V$ from its posterior distribution and then sample f_t from its posterior conditional on the corresponding value of V. Recall that the posterior distributions of f_t and V can be found by means of (3.5). We then use (3.1) with yield biomass set to 0 to simulate a trajectory of unfished biomass. We store f_t on a discrete mesh and evaluate $f_t(B_{t-1})$ by linear interpolation.

At the forecasting stage, we condition on B_t remaining positive within the forecasting period. Technically, we should enforce the same constraint at the model-specification stage, and modify our posterior distributions to reflect it using a resam-

pling technique. We omit this step because experience with this method indicates that although constraining biomass to be positive is important for forecasting, it makes minimal difference to the model fit. An alternative would be to model $\log B_t$ with a model of the form 3.1.

B.2 Proof of theorem 1

Let $\mathbf{b}$ denote an arbitrary vector of n biomass values, and let $\mathbf{b}^+$ denote the concatenation $\{\mathbf{b}, B_t\}$. Define the following notation for bivariate functions C:

$$\begin{aligned} C(b,\mathbf{b}) &= \left[\begin{array}{ccc} C(b,\mathbf{b}_1) & \cdots & C(b,\mathbf{b}_n) \end{array}\right], \\ C(\mathbf{b},\mathbf{b}) &= \left[\begin{array}{ccc} C(\mathbf{b}_1,\mathbf{b}_1) & \cdots & C(\mathbf{b}_1,\mathbf{b}_n) \\ \vdots & \ddots & \vdots \\ C(\mathbf{b}_n,\mathbf{b}_1) & \cdots & C(\mathbf{b}_n,\mathbf{b}_n) \end{array}\right], \\ C(\mathbf{b},b) &= \left[\begin{array}{ccc} C(\mathbf{b}_1,b)\cdots C(\mathbf{b}_n,b) \end{array}\right]^T, \end{aligned}$$

and the following for univariate functions m:

$$m(\mathbf{b}) = \left[\begin{array}{ccc} m(\mathbf{b}_1) & \cdots & m(\mathbf{b}_n) \end{array}\right]^T.$$

Define analogous notation for evaluation at $\mathbf{b}^+$.

From (3.4),

$$f_t|B_0 \ldots B_{t-1}, V \sim \mathrm{GP}\left(m_{t-1}, \frac{VC_{t-1}}{\delta S_{t-1}}\right).$$

By the definition of the Gaussian process [11],

$$f_t(\mathbf{b}^+)|B_0 \dots B_{t-1}, V \sim \mathrm{N}\left(m_{t-1}(\mathbf{b}^+), \frac{VC_{t-1}(\mathbf{b}^+, \mathbf{b}^+)}{\delta S_{t-1}}\right).$$

Since $f(B_t)$ is an element of the vector $f(\mathbf{b}^+)$, (3.1) and standard multivariate normal conditioning rules [93] imply

$$\begin{aligned} f_t(\mathbf{b})|B_0 \dots B_t, V \sim \\ \mathrm{N}\left[m_{t-1}(\mathbf{b}) + \frac{C_{t-1}(\mathbf{b}, B_t)}{C_{t-1}(B_t, B_t) + \delta S_{t-1}}(B_t + Y_t - m_{t-1}(B_{t-1})),\right. \\ \left.\frac{V}{\delta S_{t-1}}\left(C_{t-1}(\mathbf{b}, \mathbf{b}) - \frac{C_{t-1}(\mathbf{b}, B_t)C_{t-1}(B_t, \mathbf{b})}{C_{t-1}(B_t, B_t) + \delta S_{t-1}}\right)\right] \\ = \mathrm{N}\left(m_t(\mathbf{b}), \frac{VC_t(\mathbf{b}, \mathbf{b})}{S_t}\right), \end{aligned}$$

where m_t and C_t are as given in (3.5). Since this is true for any vector of biomass values $\mathbf{b}$, the definition of the Gaussian process implies

$$f_t|B_0 \dots B_t, V \sim \mathrm{GP}\left(m_t, \frac{VC_t}{S_t}\right).$$

The pdf of $1/V$ given $B_0 \dots B_{t-1}$ is

$$p\left(\frac{1}{V}\middle| B_0 \dots B_{t-1}\right) \propto \left(\frac{n_0 + t - 1}{2} - 1\right) \exp\left(-\frac{S_{t-1}(n_0 + (t-1))}{2V}\right).$$

From (3.1),

$$B_t|Y_t, B_0 \dots B_{t-1}, V \sim \mathrm{N}\left(m_{t-1}(B_{t-1}) - Y_t, \frac{VC_{t-1}(B_{t-1}, B_{t-1})}{\delta S_{t-1}} + 1\right).$$

By Bayes' theorem,

$$\begin{aligned} p\left(\frac{1}{V}\middle| B_0 \dots B_t\right) \propto \left(\frac{n_0 + t}{2} - 1\right) \exp\left(-\frac{S_{t-1}(n_0 + (t-1)) + \frac{\delta S_{t-1}(B_t + Y_{t-1} - m_{t-1}(B_{t-1}))^2}{C_{t-1}(B_{t-1}, B_{t-1}) + \delta S_{t-1}}}{2V}\right) \\ \Rightarrow \frac{1}{V}\middle| B_0 \dots B_t \sim \mathrm{Gamma}\left(\frac{n_0 + t}{2}, \frac{S_t(n_0 + t)}{2}\right), \end{aligned}$$

where S_t is as given in (3.5). $\square$

Theorem 1 is closely related to Theorem 4.3 of West and Harrison [93], and in fact can be derived from their result. The following table translates between our notation and that of West and Harrison's Dynamic Linear Model:

DLM variable	Interpretation in terms of our model
Y_t	Biomass plus yield, $B_t + Y_t$.
θ_t	Unknown function f_t.
F_t'	Evaluation at biomass B_{t-1} in the first argument. For example, $F_t'\theta_t$ can be read as $f_t(B_{t-1})$, and $F_t'C_{t-1}$ as $C_{t-1}(B_{t-1}, \cdot)$.
F_t	Evaluation at biomass B_{t-1} in the second argument. For example, $C_{t-1}F_t$ can be read as $C_{t-1}(\cdot, B_{t-1})$.
G_t	Identity transformation; can be disregarded. For example, $G_t\theta_t$ can be read simply as f_t.
W_t	$\frac{1-\delta}{\delta}C_{t-1}$, see section 6.3 of West and Harrison [93].

The variables n_t, S_t, m_t, C_t, δ, ϵ_t, and V play the same role in the DLM as in our model.

The key observation is that, although evaluation of f_t at B_{t-1} is usually a nonlinear operation on B_{t-1}, it is a *linear* operation on f_t:

$$(f_t + g)(B_{t-1}) = f_t(B_{t-1}) + g(B_{t-1}), \quad g : \mathbb{R}^+ \to \mathbb{R}$$

$$(cf_t)(B_{t-1}) = cf_t(B_{t-1}), \qquad c \in \mathbb{R}$$

One way to see this is to think of evaluation as integration against the Dirac delta distribution, as in chapter 2 of Stakgold [85].

Heuristically, the interpretations of the evaluation operators F_t and F_t' can be most easily understood by discretizing f_t and thinking of F_t as a column vector whose elements are zero except for a single 1, which is located at the index i corresponding to

the value b_i on the discretized biomass axis that is closest to B_{t-1}. With this definition, Theorem 4.3 of West and Harrison [93] becomes (3.5) in the continuous limit. This insight can be used to extend other results of West and Harrison to models like (3.1) with minimal analytical effort.