FAST AND EFFECTIVE STATISTICAL INFERENCE FOR SPATIO-TEMPORAL DATA, WITH APPLICATIONS IN MARINE ECOLOGY

by

Ethan Lawler

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy

at

Dalhousie University
Halifax, Nova Scotia
April 2022

Table of Contents

List of Tables

List of Figures

Abstract

Data in marine ecology are often characterized by substantial observation noise, patchiness in sampling coverage, and complex dynamics. Statistical techniques for analyzing these data need to account for complicated data generating mechanisms while still being computationally tractable. In this thesis we develop statistical models (in a maximum likelihood framework) that account for these difficulties to address two important questions in marine ecology: how can we infer an individual animal's behaviour from observing only their movement? how can we use observations of fish presence, counts, and/or abundance from scientific surveys to gain insight into the spatio-temporal dynamics of the fish population?

Standard hidden Markov models (HMMs) for inferring behaviour from animal movement paths are shown to perform poorly when the movement includes autocorrelation not accounted for by the HMM framework. We develop an extension of the HMM that can account for this additional autocorrelation and provide diagnostics for determining when the autocorrelation is present. We analyze the movement paths of two grey seals to further validate our new modelling framework.

Scientific surveys often use a stratified random sampling design to generate data with good spatial coverage over multiple decades with hundreds of spatially referenced observations per year. These datasets are too large to be analyzed with traditional methods, so we adapt the nearest neighbour Gaussian process, a modern advance in computationally efficient spatial modelling, for use within a hierarchical spatio-temporal generalized linear mixed modelling framework. We analyse two real datasets: Carolina wren counts in the state of Missouri, Haddock survey data obtained on the Scotian Shelf, off Nova Scotia.

After developing our spatio-temporal model framework for a univariate response, we investigate how to generalize it to a multivariate response variable. Such data often arises in fisheries surveys wherein fish of different age-classes are counted. Specifically, we introduce a cross-sectional copula for our spatio-temporal nearest neighbour Gaussian process and study the behaviour of the copula and inference procedure through simulations.

List of Abbreviations and Symbols Used

$B_t,\ b_t$	Random variable (B) of and realized (b) behavioural state at time t.
$W_t(\mathbf{s}),\ w_t(\mathbf{s})$	Random function (W) or realization (w) indexed by discrete time t and continuous space $\mathbf{s}$. Index t may be omitted for a function indexed only by space. Also referred to as spatial or spatio-temporal random effects.
$Y_{i,t}(\mathbf{s}_i), y_{i,t}(\mathbf{s}_i)$	Random variable (Y) or data point (y) for the ith observation of a response variable at time t and location $\mathbf{s}_i$
Φ	Ch 4, standard normal distribution function.
$\Theta,\ \theta,\ \hat{\Theta},\ \hat{\theta}$	A set of parameters ($\Theta,\ \theta$) and their estimates ($\hat{\Theta},\ \hat{\theta}$).
$\boldsymbol{\Sigma}$	Covariance matrix.
$\boldsymbol{\beta}$	Vector of regression parameters.
$\boldsymbol{\delta}$	Stationary distribution of the behavioural state process implied by the transition probability matrix $\mathbf{A}$.
$\boldsymbol{\epsilon}$	Temporal random effects.
$\boldsymbol{\rho}_c$	Ch 4, copula parameters.
$\delta\mathbf{s}$	The nearest neighbours of location $\mathbf{s}$.
ℓ	Various loglikelihood (ℓ) components. Joint loglikelihood (ℓ_J), loglikelihood components for the response distribution (ℓ_y), random effects (ℓ_w), predictions (ℓ_p), copula (ℓ_c), and subcomponents of ℓ_w corresponding to the persistent graph (ℓ_{per}) and transient graph (ℓ_{trans}), respectively.
$\mathbb{R}^+$	The non-negative real numbers.
$\mathbb{R}^d$	d dimensional space with coordinates in the real numbers $\mathbb{R}$.
$\mathbf{A}$	Transition probability matrix for the behavioural state process.

$\mathbf{H}(\theta_t)$	Rotation function or matrix for rotating by a deflection angle θ_t.
$\mathbf{I}$	Identity matrix.
$\mathbf{L}\left(d_{(t,t+1)}, \theta_t\right)$	Likelihood component for the observations at time t.
$\mathbf{X}_i$	Design matrix of fixed effects / covariates.
$\mathbf{c}$	Chs 3 & 4, cross-covariance vector between a target location and it's neighbours.
$\mathbf{q}$	Vector of Uniform[0,1] quantiles.
$\mathbf{s}$	Spatial coordinates.
$\mathbf{x}_t$	Ch 2, location in a movement path at time t.
$\mathcal{C}(\mathbf{s}_1,\ \mathbf{s}_2)$	Covariance function giving the covariance between two points $\mathbf{s}_1$ and $\mathbf{s}_2$ of a spatial random function.
$\mathcal{G}_{per}$	Persistent graph. The set of nodes and directed edges which describe the locations and nearest neighbours of the random effect locations which appear at every time point.
$\mathcal{G}_{trans,t}$	Transient graph at time t. The set of nodes and directed edges which describe the locations and nearest neighbours of the random effect locations which appear only when there is a corresponding observation at a location that is *not* part of the persistent graph.
$\mathcal{G}_t$	Directed acyclic graph at time t. Index t may be omitted for a general graph.
$\mathcal{S}_{per}$	Locations of the persistent graph. In a spatio-temporal nearest-neighbour Gaussian process, the nodes of the directed acyclic graph which appear at every time point.
$\mathcal{S}_{trans}$	Locations of the transient graph. In a spatio-temporal nearest-neighbour Gaussian process, the nodes of the directed acyclic graph which appear only when there is a corresponding observation at a location that is *not* part of the persistent graph.
μ	Chs 3 & 4, Mean of temporal random effects.

μ_{RL}	Ch 2, mean reversion level. The long-term mean of the step-lengths within a single behavioural state.
$\mu_{i,t}(\mathbf{s}_i)$	Chs 3 & 4, response mean for the ith observation at time t and location $\mathbf{s}_i$.
$\mu_t(\mathbf{s})$	Ch 1, mean function giving the mean of a spatio-temporal random function indexed by discrete time t and continuous space $\mathbf{s}$. Index t may be omitted for a function indexed only by space.
ϕ	Ch 2, autoregressive parameter for the step length process. Chs 3 & 4, correlation between consecutive times for the temporal random effects.
ρ	Ch 2, concentration parameter for the wrapped Cauchy distribution. Chs 3 & 4, spatial range parameter.
σ	Standard deviation parameter. Chapters 3 & 4, the one-step-ahead standard deviation of the temporal random effects.
τ	Spatial standard deviation parameter.
θ_t	Ch 2, deflection angle between the locations at time $t-1$, t, and $t+1$.
$\tilde{w}_{t-1}(\mathbf{s})$	A proxy random effect computed using a predictor of the local mean.
c	Ch 2, center parameter for the wrapped Cauchy distribution.
c_{Gaus}	Gaussian copula.
$d_{(t,t+1)}$	Distance between the location at time t and the location at time $t+1$.
f_θ	Probability density function with parameters θ.
g	Link function.
$m_t(\mathbf{s})$	Chs 3 & 4, mean function giving the mean of a spatio-temporal random function indexed by discrete time t and continuous space $\mathbf{s}$. Index t may be omitted for a function indexed only by space.
$n_{\text{prop}},\ n_{\text{adj}}$	Metrics used in choosing a time step and group cutoff level.
t	Time coordinates.

AR(1)	Autoregressive process of order 1.
ARIMA	Autoregressive integrated moving average process.
CarHMM	Conditionally autoregressive hidden Markov model.
CDF	Cumulative distribution function.
DAG	Directed acyclic graph.
DCRWS	First-difference correlated random walk with switching.
GIS	Geographic information system.
GLMM	Generalized linear mixed model.
GP	Gaussian process.
GPS	Global positioning system.
HMM	Hidden Markov model.
HMMM	Hidden Markov movement model.
IID	Independent and identically distributed.
MVN	Multivariate normal distribution.
NNGP	Nearest-neighbour Gaussian process.
SPDE	Stochastic partial differential equation.
staRVe	Spatio-temporal analysis of research vessel data.
TMB	Template model builder.

WC Wrapped Cauchy distribution.

0.1 Note on Notation

We use the following conventions:

- Scalars and univariate random variables will be given non-bold letters e.g. k for a scalar, Y_i for a single data point
- Vectors or matrices will be given bold letters e.g. $\boldsymbol{\beta}$ for a vector of regression coefficients, $\boldsymbol{\Sigma}$ for a covariance matrix
- Functions will be given non-bold letters e.g. $W_t(\mathbf{s})$ for the spatio-temporal process indexed by time t and location $\mathbf{s}$. An exception will be when we want to emphasize that we are evaluating the function at each element of a vector and returning a vector, e.g. $\mathbf{W}_t(\mathbf{s}_{1:3})$ which is the vector $[W_t(\mathbf{s}_1),\ W_t(\mathbf{s}_2),\ W_t(\mathbf{s}_3)]$.
- Random variables will be given uppercase letters; realizations from random variables will be given lowercase letters e.g. $W_t(\mathbf{s}_i)$ for the random variable at time t and location $\mathbf{s}_i$, $w_t(\mathbf{s}_i)$ when conditioning on the value of that random variable

Chapter 1

Introduction

Many ecological phenomena are inseparably linked to their environment. At the individual level an animal's movement is motivated by the environment they are moving through and the ecological niches they prefer to inhabit [Shepard et al., 2013]. At the population level the cumulative effect of every individual's movements is scaled up to produce an ever-changing spatial distribution of abundance [Morales et al., 2010, Michelot, 2019]. Observing either an individual's movement path or a population's spatial distribution of abundance is difficult in a marine environment where the deep water makes exhaustive data collection impossible. Accordingly, statistical methods for analyzing the data we are able to collect must be able to account for these difficulties in order to answer the scientific questions asked of them [Mills Flemming and Field, 2014].

Statistical models for movement ecology answer questions such as "what is the home range of this particular community?" or "how can we recover a highly detailed movement path from a sample of inaccurately observed locations?" The one we will focus on here is "what does an animal's movement tell us about the underlying behaviour driving that movement?" Statistical models used to answer this question have to accurately capture the biology in order to accurately infer animal behaviour from their movement. For example foraging behaviour is associated with tortuous, erratic movement while transiting behaviour is associated with steady and directed movement [Whoriskey et al., 2017]. These inferences are complicated by available data being opportunistic in nature where locations are only sampled when an animal is at or near the surface of the water. The data are then prone irregular sampling with both long stretches of time with no sampled locations and short durations of time with many observations. Most methods for inferring behaviour from movement are some form of hidden Markov model and require isochronal observations. Linear interpolation of the observed locations is a common method to coerce the data to be regularly timed,

however the interpolation procedure must be accounted for by the inference procedure [Glennie et al., 2022].

Irregular sampling appears in data for the spatial distribution of fish populations as well, although in a different form. These data often come from general scientific surveys which follow a geographically-stratified random sampling scheme, repeated at the same time every year but with a different sampling of locations [Smith, 1999]. Each sample location within a year is carried out by a single fishing set from which is recorded the total number, the total weight, and the presence or absence in the sample of a wide variety of species. These data can be used by themselves to produce complete maps of the predicted spatial distribution of a species, combined with data on the total commercial catch to estimate the total population abundance, or used with auxiliary information to estimate the growth characteristics of species through an age length key [Kristensen et al., 2014, Babyn et al., 2021]. Traditional statistical methods for spatio-temporal data are not immediately applicable for fisheries surveys. The random sampling scheme is incompatible with state-space methods which require a set of fixed monitoring stations that do not change from year to year. In addition the surveys collect large amounts of data, sometimes reaching hundreds of observations per year and often spanning multiple decades.

Modern statistical methods for both animal movement and population distributions are available that overcome the above issues, although the animal movement literature is more well-developed than the spatio-temporal species distribution literature. The spatio-temporal species distribution literature lags behind in part because the statistical machinery used is more specialized and more technical than that used in animal movement. In this thesis we continue the development of these modern methods. Both methods build on the historical methods, particularly true for the spatio-temporal distribution modelling, so we first give a history of these methods.

1.1 1963 - 1985 : The Principles of Geostatistics

Sunday 01 December, 1963. The eighth issue of Volume 58 of the journal *Economic Geology* is published. It offers a smattering of papers. Most are field reports on the geology of different mines or mining districts. A few give details of geochemical

reactions. One of them was a short twenty page article by Georges Matheron entitled *Principles of Geostatistics* [Matheron, 1963]. In it Matheron introduced the variogram and the spatial prediction technique known as kriging and with those he had introduced the new field of geostatistics. (The field of geostatistics had already coalesced into a distinct field of statistics through the work of geological research groups in France. Matheron's 1963 article is the first appearance of these techniques in the English language. Being an overview of the existing techniques in a single collection, the article acts as a convenient starting point for the field of geostatistics.)

Matheron first introduces a regionalized variable, an early name for what we now call a spatial random field, defining it simply as "an actual function, taking a definite value in each point of space." These regionalized variables could be represented as realized values from a collection of correlated random variables, one for each point $\mathbf{s}$ of space. The locations $\mathbf{s}$ are points in $\mathbb{R}^d$, and typically $d = 2$ for (projected) geographic data. Matheron implicitly assumes two things about the spatial random field. First that $\mathrm{E}\left[W\left(\mathbf{s}+\mathbf{h}\right) - W\left(\mathbf{s}\right)\right] = 0$ for any location $\mathbf{s}$ and any separation vector $\mathbf{h}$ which, pathologies aside, implies that the mean of each individual $W\left(\mathbf{s}_1\right)$ is the same as any other $W\left(\mathbf{s}_2\right)$. Second that the variance of the difference between two regionalized variables is a function solely of their separation and not their actual location, $\mathrm{Var}\left[W\left(\mathbf{s}+\mathbf{h}\right) - W\left(\mathbf{s}\right)\right] = 2\gamma\left(\mathbf{h}\right)$. These two properties taken together are known as intrinsic stationarity. The function $2\gamma\left(\mathbf{h}\right)$ is known as the variogram.

After introducing the variogram Matheron discusses simple kriging, which is the procedure of finding the best linear unbiased predictor of some unobserved regionalized variable $W\left(\mathbf{s}_0\right)$ as a function of the observed regionalized variables $W\left(\mathbf{s}_1\right)$, ... , $W\left(\mathbf{s}_n\right)$. While Matheron does not quite detail the math relying instead on pre-computed tables, the variogram is used to obtain the coefficients a_i in said linear predictor

$$\hat{W}\left(\mathbf{s}_0\right) = \sum_{i=1}^{n} a_i\left(\mathbf{s}_0\right) W\left(\mathbf{s}_i\right). \tag{1.1}$$

The variogram provides a way to calculate the coefficients $a_i\left(\mathbf{s}_0\right)$ for any location of interest $\mathbf{s}_0$. Thus kriging provides an interpolation method to produce a complete spatial map of the predicted regionalized variables, with guarantees under certain

conditions that the predicted map is the best that one could feasibly produce.

A few years later Matheron extended simple kriging to universal kriging by allowing the mean of the regionalized variables to depend on external variables or covariates. The method was first published as a book in French, and soon after was packaged as an article for English speaking audiences with the help of C.J. Huijbregts [Matheron, 1969, Huijbregts and Matheron, 1971]. In essence universal kriging works by first de-trending the regionalized variables by fitting a linear or polynomial regression relating the regionalized variables to the covariates and then performing simple kriging on the residuals of that regression. Matheron noted that some of the covariates used in the universal kriging model could themselves be regionalized variables.

His next contribution was to formalize the notion of pairs (or more) of regionalized variables using intrinsic random functions [Matheron, 1973]. While the appearances of intrinsic random functions in 1973 marked the first multivariate geostatistics, the method was not very practical due to the arcane mathematical machinery involved. In 1979 Matheron simplified the intrinsic random function approach into the friendlier cokriging technique [Matheron, 1979]. Cokriging augments simple kriging (or universal kriging) by treating a target variable and covariates all as regionalized variables. In addition to each regionalized variable having its own variogram, each pair of regionalized variables has a covariogram $\gamma_{i,j}(\mathbf{h}) = \mathrm{E}\left[\left(W_i(\mathbf{s}+\mathbf{h}) - W_j(\mathbf{s})\right)\left(W_j(\mathbf{s}+\mathbf{h}) - W_j(\mathbf{s})\right)\right]$. The variograms and covariograms are then used to find the best linear unbiased predictors of all the regionalized variables, although typically only the single target variable was of interest. Many covariates could be sampled more easily than the target variable and co-kriging allowed all of the covariate data to be used to predict the target variable even in cases where the target variable and covariates were not sampled in the same location.

While Matheron was developing cokriging another statistician at the same institution was improving universal kriging in a slightly different way. The same year that Matheron published his cokriging report, J.P. Delhomme published kriging with external drift in a technical report [Delhomme, 1979]. Instead of using covariograms to describe the relationship between different regionalized variables, Delhomme adjusted the linear regression equation of universal kriging to adjust predictions of the target

regionalized variable by accounting for the spatial structure of the covariate regionalized variable. Cokriging and kriging with external drift would eventually be found out to be equivalent parametrizations of the same technique. Cokriging focussed on joint specification of the multiple regionalized variables while kriging with external drift focussed on the conditional model of the target regionalized variable given the covariate regionalized variables.

Following the parallel development of cokriging and kriging with external drift at the Paris School of Mines, a doctoral student at the school developed the final classical approach to multivariate geostatistics. In his 1985 thesis H. Wackernagel introduced the linear model of coregionalization, which he published in English three years later [Wackernagel, 1985, Wackernagel, 1988]. While the main formulation of the linear model of coregionalization was in terms of linear combinations of variograms, Wackernagel also gave a more interpretable formulation in terms of linear combinations of independent regionalized "factor" variables. In this way the linear model of coregionalization translates a principal component analysis into the multivariate spatial setting.

By this time mining companies had been using kriging methods quite frequently, a fact which did not sit well with everybody. A war between Matheron's camp and the anti-krigers came to a head in 1986 when the journal *Mathematical Geography* invited both sides to submit pamphlets arguing the case for or against the use of kriging in mining operations. From Matheron's camp came A.G. Journel who we will meet momentarily. From the anti-krigers came G.M. Philip and D.F. Watson. Given the nature of the invitation, Philip and Watson's diatribe against kriging was cast in purposefully provocative style. I can only do it justice by providing an extended quote.

> The terms "unbiased," "optimal," "best estimator," "minimum variance," "tree grade," etc. scatter their writings like confetti... How can an observation have a "variance"? How can a mining block have a "variance" different from that of a deposit? How can mineral grades have an infinite "variance"? How can accuracy and precision be thought synonymous? Why is a linear method of estimation thought to be good? How can arcane concepts

> such as “one-dimensional Brownian movement" and “microergodicity" have relevance to ore deposit assessment? How can stationarity in grade values of an ore deposit be assumed? And so on...
>
> [...]
>
> For a growing understanding of geostatistics served to show us that geostatistical estimation is a sham–an expensive computer guessing game which is unscientific in its approach, based on statistical misconceptions, claims the world, and provides merely a bad result. That we write so acerbically is because we resent the time that we have wasted upon it. [Philip and Watson, 1986]

Of great concern to Philip and Watson were the “probabilistic trappings" of the variogram. They argue that a variogram could be useful if it were adapted as merely a local deterministic smoothing function. Only time would tell if Philip and Watson would successfully divorce mining operations from variograms and their probabilistic trappings.

1.2 1983 - 1989 : Probabilistic Geostatistical Models

A variogram for a regionalized variable implies a covariance function for the regionalized variable $C(\mathbf{h}) = \text{Cov}[W(\mathbf{s}),\ W(\mathbf{s}+\mathbf{h})]$, and best linear unbiased prediction (with squared error loss) implies an underlying Gaussian distribution. Prediction of regionalized variables using a variogram can thus be recast as the conditional distribution of a Gaussian probabilistic model. To see how assume that the unobserved regionalized variable $W(\mathbf{s}_0)$ and the observed regionalized variables $\mathbf{W}(\mathbf{s}_{1:N}) = [W(\mathbf{s}_1),\ ...,\ W(\mathbf{s}_n)]^T$ have a joint multivariate normal distribution

$$\begin{bmatrix} W(\mathbf{s}_0) \\ \mathbf{W}(\mathbf{s}_{1:N}) \end{bmatrix} \sim \text{MVN}\left(\begin{bmatrix} 0 \\ \mathbf{0} \end{bmatrix},\ \begin{bmatrix} \Sigma_{0,0} & \mathbf{\Sigma}_{0,1:N} \\ \mathbf{\Sigma}_{0,1:N}^T & \mathbf{\Sigma}_{1:N,1:N} \end{bmatrix} \right), \qquad \Sigma_{i,j} = C(\mathbf{h} = \mathbf{s}_i - \mathbf{s}_j) \quad (1.2)$$

Then the conditional distribution of the unobserved variable given the observed variables is

$$W(\mathbf{s}_0) \;\Big|\; \mathbf{w}(\mathbf{s}_{1:N}) \sim \mathrm{N}\left(\boldsymbol{\Sigma}_{0,1:N}\boldsymbol{\Sigma}_{1:N,1:N}^{-1}\mathbf{w}(\mathbf{s}_{1:N}), \Sigma_{0,0} - \boldsymbol{\Sigma}_{0,1:N}\boldsymbol{\Sigma}_{1:N,1:N}^{-1}\boldsymbol{\Sigma}_{0,1:N}^{T}\right) \quad (1.3)$$

The conditional mean is a linear combination of the observed regionalized variables with the linear coefficients given by the corresponding elements of the vector $\boldsymbol{\Sigma}_{0,1:N}\boldsymbol{\Sigma}_{1:N,1:N}^{-1}$. These coefficients are the same coefficients that give the best linear unbiased predictor obtained from the original variogram.

Most if not all of Matheron's work on kriging had rejected the notion of using a full probabilistic model for the regionalized variables. His techniques were instead motivated by intrinsic stationarity and the best linear unbiased prediction criteria. The first mention of the connection between variograms, kriging, and the multivariate normal distribution that I can find in the literature is in a 1978 geostatistical textbook by A.G. Journel and C.J. Huijbregts [Journel and Huijbregts, 1978]. Even if the connection had not been explicitly mentioned in the literature before this book, the result was almost certainly previously known to geostatisticians.

The recasting of regionalized variables, variograms, and kriging into multivariate Gaussian distributions, covariance functions, and conditional distributions would eventually pave the way for full probabilistic modelling in geostatistics. In 1984 K.V. Mardia and R.J. Marshall published *Maximum Likelihood Estimation of Models for Residual Covariance in Spatial Regression*, what should be considered a landmark paper but is often overlooked, in *Biometrika* [Mardia and Marshall, 1984]. They bring the concept of a Gaussian process into the geostatistical literature, replace the independent and identically distributed residuals of the linear regression model with spatially-correlated residuals, and provide some consistency and asymptotic normality results for maximum likelihood estimation of the spatial covariance function.

The Gaussian process formalizes the notion of Matheron's regionalized variables. Mathematically a Gaussian process is a collection of random variables $\{W(\mathbf{s}); \mathbf{s} \in \mathbf{S}\}$ indexed by some set $\mathbf{S}$. In the spatial setting $\mathbf{S}$ is just the set of spatial coordinates in $\mathbb{R}^2$ or more rarely $\mathbb{R}^3$, although as we will see it can be extended to include coordinates

in space and time. To form a Gaussian process the collection of random variables must have the property that any finite subset of random variables has a jointly normal distribution. A Gaussian process is parameterized by a mean function and covariance function. The mean function $\mu(\mathbf{s})$ gives the mean of each $W(\mathbf{s})$ as a function of $\mathbf{s}$ including the possibility of $W(\mathbf{s})$ depending on spatially-referenced covariates. The covariance function $\mathcal{C}(\mathbf{s}_1,\ \mathbf{s}_2)$ encodes the first law of geography: "everything is related to everything else, but near things are more related than distant things" [Tobler, 1970]. There are many classes of covariance functions but by far the most famed covariance function in all of geostatistics is the Matérn covariance function

$$\mathcal{C}_\nu(\mathbf{s}_1, \mathbf{s}_2) = \mathcal{C}_\nu(d = ||\mathbf{s}_1 - \mathbf{s}_2||) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\frac{d}{\rho}\right)^\nu K_\nu \left(\sqrt{2\nu}\frac{d}{\rho}\right) \tag{1.4}$$

originally derived in 1960 by B. Matérn [Matérn, 1960]. The Matérn covariance function has three parameters: the marginal variance σ^2, the smoothness ν, and the spatial range ρ. Γ is the gamma function and K_ν is the modified Bessel function of the second kind.

After introducing the Gaussian process Mardia and Marshall write down the log-likelihood for observations coming from such a process. Letting $\mathbf{w}(\mathbf{s}_{1:n})$ be the data collected at locations $\mathbf{s}_1,\ ...,\ \mathbf{s}_n$, the log-likelihood is simply the joint (log) density of their multivariate normal distribution

$$\ell(\theta;\ \mathbf{w}(\mathbf{s}_{1:n})) = -\frac{1}{2}\text{logdet}\boldsymbol{\Sigma}_{1:n} - \frac{1}{2}\left[\left(\mathbf{w}(\mathbf{s}_{1:n}) - \boldsymbol{\mu}(\mathbf{s}_{1:n})\right)^T \boldsymbol{\Sigma}_{1:n}^{-1} \left(\mathbf{w}(\mathbf{s}_{1:n}) - \boldsymbol{\mu}(\mathbf{s}_{1:n})\right)\right] \tag{1.5}$$

where $\boldsymbol{\mu}(\mathbf{s}_{1:n}) = [\mu(\mathbf{s}_1),\ ...,\ \mu(\mathbf{s}_n)]^T$ and as in Equation 1.2, $\Sigma_{i,j} = C(\mathbf{h} = \mathbf{s}_i - \mathbf{s}_j)$. After discussing some of the computational aspects of computing this likelihood function some asymptotic normality and weak consistency results are proven for both the regression coefficients and the covariance function parameters.

Unfortunately it quickly became apparent maximum likelihood estimation for these models exhibited some significant flaws. The inversion of a dense $n \times n$ covariance matrix proved to be computationally prohibitive with simulation experiments in the Mardia and Marshall paper limited to just 10 locations of observed data. More seriously, J.J. Warnes and B.D. Ripley almost immediately found the maximum

likelihood function of the covariance function to be prone to multimodality and would sometimes give non-sensical parameter estimates [Warnes and Ripley, 1987]. In 1987 Mardia quickly responded to concerns of multimodality by introducing a profile likelihood to estimate the parameters of the spatial covariance function. By estimating the regression coefficients and marginal variance as a deterministic function of the spatial range parameter he was able to produce a unimodal profile log-likelihood of the spatial range parameter and achieve a well-behaved maximum likelihood estimate of the model parameters [Mardia and Watkins, 1989].

1.3 1993 - 1998 : Spatio-Temporal Analysis and Hierarchical Models

Early techniques for analyzing spatio-temporal data grew out of the time series literature instead of the geostatistics literature. The data for these early models nearly always consisted of repeated measurements at a set of fixed monitoring stations leading to a time series at each monitoring station location. For example one of the simplest spatio-temporal models is the spatio-temporal autoregressive (of order 1,1) model

$$\mathbf{y}_{t+1} = \phi \cdot \mathbf{y}_t + \psi \cdot \mathbf{V}\mathbf{y}_t + \boldsymbol{\epsilon}_t \tag{1.6}$$

where ϕ is a temporal autocorrelation parameter, ψ is a spatial autocorrelation parameter, $\mathbf{V}$ is spatial weight matrix, and $\boldsymbol{\epsilon}_t$ is white noise error. The spatial weight matrix was typically estimated through the empirical cross-correlation function, and it is analogous to the variogram. A 1975 paper by R.L. Martin and J.E. Oeppen provides a brief contemporaneous review of these time series-centric methods [Martin and Oeppen, 1975].

The first geostatistical approach to analyzing spatio-temporal data came in a 1983 paper by B.P. Eynon and P. Switzer, simply titled *The Variability of Rainfall Acidity* [Eynon and Switzer, 1983]. They write the observed rainfall acidity $\mathrm{pH}(t, \mathbf{s})$ at time t and location $\mathbf{s}$ as the sum of four components, a deterministic trend $\mu(t, \mathbf{s})$ taking into account e.g. seasonality, an autocorrelated spatial process $W(\mathbf{s})$, an autocorrelated temporal process $Y(t)$, and residual white noise $\epsilon(t, \mathbf{s})$

$$\mathrm{pH}(t, \mathbf{s}) = \mu(t, \mathbf{s}) + w(\mathbf{s}) + y(t) + \epsilon(t, \mathbf{s}). \tag{1.7}$$

To estimate the parameters of this model Eynon and Switzer first fit the deterministic trend $\mu(t, \mathbf{s})$ using least squares. Then the residuals $\widetilde{\mathrm{pH}}(t, \mathbf{s}) = \mathrm{pH}(t, \mathbf{s}) - \hat{\mu}(t, \mathbf{s})$ were taken as estimates of the combined effect of the spatial, temporal, and white noise components. An empirical variogram was estimated for $y(t)$ by looking at the differences in residual rainfall acidity between two different times at the same location $\widetilde{\mathrm{pH}}(t, \mathbf{s}_0) - \widetilde{\mathrm{pH}}(t + \Delta t, \mathbf{s}_0)$. An empirical variogram was estimated for $w(\mathbf{s})$ in a similar fashion by looking at differences in residual rainfall acidity between two different locations at the same point in time.

Because Eynon and Switzer's model parameterized $w(\mathbf{s})$ and $y(t)$ with variograms they were able to produce predicted maps of rainfall acidity for multiple years. Previous spatial methods could of course accomplish the same thing by analyzing each year of data separately, possibly using a single variogram across years. The new spatio-temporal method allowed one year's predicted map to be informed by all the observed data instead of just the observations made within that same year.

By the early 90s Gaussian processes had supplanted the variogram as the default tool for geostatistics. Accordingly, focus was shifted from estimating a variogram to modelling a covariance matrix. Early work in this area reformulated Eynon and Switzer's rainfall acidity model to have a single combined spatio-temporal process with a separable spatio-temporal covariance function. A spatio-temporal covariance function is an extension of the spatial covariance function to additionally allow for a time index

$$\mathcal{C}((\mathbf{s}_1,\ t_1),\ (\mathbf{s}_2, t_2)) = \mathrm{Cov}[W(\mathbf{s}_1, t_1),\ W(\mathbf{s}_2, t_2)] \tag{1.8}$$

A separable spatio-temporal covariance function can be factored as a product of a purely spatial covariance function and a purely temporal covariance function

$$\mathcal{C}((\mathbf{s}_1,\ t_1),\ (\mathbf{s}_2, t_2)) = \mathcal{C}_{\mathbf{s}}(\mathbf{s}_1,\ \mathbf{s}_2) \cdot \mathcal{C}_t(t_1,\ t_2). \tag{1.9}$$

The covariance matrix for observations from a general spatio-temporal Gaussian process would have been computationally prohibitive. A separable covariance function allowed the covariance matrix to be factored similarly to the covariance function itself,

and in doing so provided a cheap way to compute the inverse of the covariance matrix

$$\mathbf{\Sigma}^{-1} = (\mathbf{\Sigma_s} \otimes \mathbf{\Sigma}_t)^{-1} = \left(\mathbf{\Sigma_s}^{-1} \otimes \mathbf{\Sigma}_t^{-1}\right). \tag{1.10}$$

There are three covariance matrices involved: the full covariance matrix $\mathbf{\Sigma}$ is a $ST \times ST$ matrix, $\mathbf{\Sigma_s}$ is a $S \times S$ spatial covariance matrix, $\mathbf{\Sigma}_t$ is a $T \times T$ temporal covariance matrix, and $\otimes$ is the Kronecker matrix product. This factorization is only possible when there are repeated observations for S fixed locations at T points in time.

Separable covariance functions proved to be incredibly useful. In 1993 Mardia and C. Goodall published *Spatial-Temporal Analyses of Multivariate Environmental Monitoring Data* in which they analyzed multivariate spatio-temporal data using a separable covariance function and maximum likelihood inference [Mardia and Goodall, 1993]. Their covariance function further extended the spatio-temporal covariance function to include multiple different variables, and their covariance matrix factored into three terms

$$\mathbf{\Sigma} = \mathbf{\Sigma_s} \otimes \mathbf{\Sigma}_t \otimes \mathbf{\Sigma}_V. \tag{1.11}$$

A year later the same authors gave a completely different technique to analyze multivariate spatio-temporal data by adapting the machinery of state-space models and the Kalman filter to geostatistical models [Goodall and Mardia, 1994]. Before deriving the spatio-temporal state-space model they introduce the general state-space model

$$\mathbf{X}_t = \mathbf{H}\mathbf{Y}_t + \boldsymbol{\epsilon}_t \tag{1.12}$$

$$\mathbf{Y}_t = \mathbf{P}\mathbf{Y}_{t-1} + \boldsymbol{\eta}_t \tag{1.13}$$

State-space models were originally invented in 1960 to solve the problem of reconstructing the movement path of an object from noisy observations of the object's location at regular intervals. In this scenario $\mathbf{Y}_t$ comprises the coordinates of the object's true location at time t with movement determined by the transition matrix $\mathbf{P}$, $\mathbf{X}_t$ are the noisy observations related to the true locations by the observation matrix $\mathbf{H}$, and $\boldsymbol{\epsilon}_t$ and $\boldsymbol{\eta}_t$ are both Gaussian noise. The original goal of the state-space

model was to recover the true locations $\mathbf{Y}_t$ from the observations $\mathbf{X}_t$, a task which was accomplished by the Kalman filter [Kalman, 1960]. Goodall and Mardia used this existing machinery but ripped the movement path reconstruction interpretation away from the model. They formulate their spatio-temporal state-space model as

$$\mathbf{W}_t(\mathbf{s}) = \mathbf{H}(\mathbf{s})\,\mathbf{Y}_t + \boldsymbol{\epsilon}_t(\mathbf{s}) \tag{1.14}$$

$$\mathbf{Y}_t = \mathbf{P}\mathbf{Y}_{t-1} + \boldsymbol{\eta}_t \tag{1.15}$$

Now $\mathbf{Y}_t$ is a (multivariate) time series which represents the global year-to-year changes in the (multivariate) spatio-temporal variable $\mathbf{W}_t(\mathbf{s})$. In the spatio-temporal state-space model the static observation matrix $\mathbf{H}$ is replaced by a spatially-varying version $\mathbf{H}(\mathbf{s})$, which gives local adjustments to the global year-to-year changes in $\mathbf{Y}_t$. In addition to the spatially-varying observation matrix the residuals $\boldsymbol{\epsilon}_t(\mathbf{s})$ are allowed to be spatially correlated, however $\boldsymbol{\epsilon}_{t_1}(\mathbf{s})$ and $\boldsymbol{\epsilon}_{t_2}(\mathbf{s})$ are independent whenever $t_1 \neq t_2$. After modifying the state-space model they present the spatio-temporal Kalman filter which combines kriging with the ordinary Kalman filter to provide predictions of $\mathbf{W}_t(\mathbf{s})$ at any point in time and space. Importantly the spatio-temporal Kalman filter is the first technique for analyzing spatio-temporal data that allowed the locations and even number of observations to change from one year to the next, allowing for applications more complicated than repeated measurements of fixed monitoring stations.

The spatio-temporal Kalman filter was an ingenious bit of handiwork, but the presentation of the spatio-temporal state-space model was awkward. In their 1998 paper *Hierarchical Bayesian Space-Time Models* C.K. Wikle, L.M. Berliner, and N. Cressie adopted the framing of hierarchical Bayesian time series models, which had been codified two years prior by Berliner [Wikle et al., 1998, Berliner, 1996]. Hierarchical Bayesian models are a more general formulation of state-space models, which allowed spatio-temporal geostatistical models a cleaner description. The basic ingredients of a

hierarchical Bayesian model are the different levels of the hierarchy

$$\text{Stage 1} \quad [\text{data} \mid \text{process, parameters}] \tag{1.16}$$

$$\text{Stage 2} \quad [\text{process} \mid \text{parameters}] \tag{1.17}$$

$$\text{Stage 3} \quad [\text{parameters}] \tag{1.18}$$

Square brackets are a shorthand notation for the probability density, for example [process | parameters] signifies the probability density for the conditional distribution of the "process" given the parameters. As the name implies hierarchical Bayesian models were originally limited to Bayesian statistical methods, hence stage 3 represents the prior distributions placed on the model parameters. However frequentist methods can still be used to estimate the model parameters, in which case one would simply remove stage 3 of the hierarchy. Stage 2, often known as the process model, is used to model the behaviour of some spatio-temporal variable. Stage 1, often known as the measurement or observation model, is used to model the variability inherent in collecting data from the process.

Wikle, Berliner, and Cressie use the hierarchical Bayesian framework to construct a general example model for analyzing spatio-temporal data. They start with the observation model $[Y_t(\mathbf{s}_i) \mid \mu_t(\mathbf{s}_i),\ \theta]$ and give the standard example of the data $Y_t(\mathbf{s}_i)$ being conditionally independent of all other data given the mean $\mu_t(\mathbf{s}_i)$ and the observation variance parameter σ_y^2

$$Y_t(\mathbf{s}_i) \mid \mu_t(\mathbf{s}_i),\ \theta \sim N\left(\mu_t(\mathbf{s}_i),\ \sigma_y^2\right). \tag{1.19}$$

They note that the locations of the observations Y do not need to coincide with the locations used to model the mean process $\mu_t(\mathbf{s})$, all that is needed is a probabilistic model to connect the two. In fact they suggest that the values of μ need only be known on a regular grid. (This off-handed remark would prove to be an especially prophetic one. In modern geostatistics it is almost an unspoken assumption that a grid or more generally a mesh is used when modelling the process.)

They then move on to the process model $[\mu_t(\mathbf{s}) \mid m(\mathbf{s}),\ \beta(\mathbf{s}),\ W_t(\mathbf{s}),\ \boldsymbol{\theta}]$ decomposing the mean process $\mu_t(\mathbf{s})$ as the sum of contributions from a spatially-varying

but deterministic mean function $m(\mathbf{s})$, a spatially-varying but deterministic temporal trend $M(t;\ \beta(\mathbf{s}))$, and a mean-zero spatio-temporal process $W_t(\mathbf{s})$. A sub-stage in the process model is used to model the spatio-temporal process $W_t(\mathbf{s})$. For their example model they use the same process equation as the general state-space model where $W_t(\mathbf{s})$ is treated as repeated measurements from a number of fixed sites, where the fixed sites include all of the observation locations. All together their example model is

$$
\begin{aligned}
Y_t(\mathbf{s}_i) \mid \mu_t(\mathbf{s}_i), \theta &\sim N\left(\mu_t(\mathbf{s}_i),\ \sigma_y^2\right) && (1.20)\\
\mu_t(\mathbf{s}) \mid m(\mathbf{s}),\ \beta(\mathbf{s}),\ W_t(\mathbf{s}) &= m(\mathbf{s}) + M(t;\beta(\mathbf{s})) + W_t(\mathbf{s}) && (1.21)\\
\mathbf{W}_t(\mathbf{s}) \mid \mathbf{W}_{t-1}(\mathbf{s}) &= \mathbf{P}\mathbf{W}_{t-1}(\mathbf{s}) + \boldsymbol{\eta}_t && (1.22)\\
\boldsymbol{\eta}_t &\sim N\left(0, \sigma_w^2\mathbf{I}\right) && (1.23)
\end{aligned}
$$

The same year as the hierarchical Bayesian spatio-temporal model was introduced, P. Diggle, J. Tawn, and R. Moyeed introduced the geostatistical generalized linear model [Diggle et al., 1998]. These two models now form the bedrock for nearly all modern geostatistical modelling. They also brought new attention to the field of geostatistics, and the body of literature quickly grew to a staggering size which I will not attempt to cover here. Instead the relevant references will be given as needed in the remainder of this thesis.

1.4 Overview of the Thesis

Chapter 2 builds on the body of literature related to animal movement. The general state-space model was an early model in analyzing marine animal movement tracks, where the main goal of the analysis was to recreate the movement path of an animal given noisy observations of their locations collected from Argos satellites or GPS pings. Eventually those recreated tracks were used to relate the characteristics of an animal's movement with their behaviour such as foraging or migrating. The hidden Markov model was introduced as a way to predict these behaviours from the animal's movement. Using the state-space terminology the process of the hidden Markov model was a sequence of behavioural states, and the observations consisted of the animal's

movement characteristics. One of the assumptions for a hidden Markov model is that the movement characteristics at time t are conditionally independent of the movement characteristics at all other times given the behavioural state at time t. I show that this assumption is often violated in marine animal movement. I offer a diagnostic tool to detect when this assumption is violated, and extend the hidden Markov model used in animal movement studies to account for the lack of conditional independence in movement behaviours. This work has been published along with a fledgling R package in [Lawler et al., 2019].

Chapters 3 and 4 together are situated in the species distribution modelling literature, with the motivating data coming from scientific surveys of fish populations. Most statistical models for fish populations have not taken into account any spatial information, instead focusing on estimating and predicting the total biomass of a fish stock and the population dynamics of the stock from year to year. State-space models are now commonly used to analyze fisheries data for these purposes, having been adapted from the location estimation task to instead estimating the true population biomass given noisy observations from scientific surveys and reported values of commercial catch [Aeberhard et al., 2018]. While the fisheries community has long recognized the value that incorporating spatial information in these statistical models would give, only recently has the statistical machinery caught up to allow such analyses to be performed [Berger et al., 2017]. These spatio-temporal analyses are still far less developed compared to their purely-temporal counterparts, and instead of replacing state-space methods they complement them. The current main use of spatio-temporal fisheries models is to apply them to fisheries-independent surveys and produce predicted maps of an index of abundance and then aggregate the predicted maps to yearly indices. Since these methods produce predictions of an *index* of abundance, the maps are useful for identifying geographic areas with relatively high or relatively low abundance but are not suitable for estimating the total biomass in the fishery.

In addition to producing indices of abundance, spatio-temporal modelling can be used to identify hotspots of other quantities. For example the work in this thesis was motivated by joint work with Isabelle Jubinville, in which we applied spatio-temporal models to identify hotspots of skate and ray bycatch in an Atlantic Canadian

groundfish fishery [Jubinville et al., 2021]. In this work we produced spatio-temporal maps of relative abundance for three bycaught skate species and ten target species using data from fisheries-independent surveys. We then overlaid these predicted maps to identify areas at high risk of bycatch, which was defined as areas with high relative abundance of a bycaught species and a high abundance of the group of target species (and vice versa). We also gave a proof-of-concept validation of these predictions using fisheries-dependent data where we found that, under certain conditions, the bycatch risk predictions from the fisheries-independent data were significant in predicting the amount of bycatch in the commercial catch.

In Chapter 3 I develop a new spatio-temporal geostatistical modelling framework that is highly flexible and takes advantage of recent advances in computationally efficient spatial modelling. The major innovation comes in the form of a spatio-temporal version of a nearest-neighbour Gaussian process, and is formulated to be easily extendable past a standard spatio-temporal geostatistical model. As a significant component of this chapter I created an easy-to-use R package that implements this new modelling framework and makes it accessible to a wide audience. The modelling framework and package provide a quick method for producing maps of predicted species abundance from large datasets, such as the ones we used in identifying bycatch risk hotspots. Part of this chapter has been submitted for publication and is pending peer review.

Chapter 4 extends the univariate model of the previous chapter to a multivariate model through the novel use of cross-sectional copulas in a geostatistical setting. In our bycatch risk work we analyzed each species individually and composited the resulting maps into a prediction of bycatch risk. This procedure assumes the different species are independent of each other, a strong assumption which is almost certainly not true in nature. By explicitly modelling the dependence between species we can quantify the dependence between species and in doing so hopefully obtain more accurate and more precise predictions of the index of abundance for each species. Cross-sectional copulas provide an almost trivial and computationally efficient way to model the dependence between species. To extend the univariate model using copulas, we first compute a version of one-step-ahead quantiles for each of the spatio-temporal random effects in the univariate model for each species. The one-step-ahead distributions for the random

effects are already computed when evaluating the likelihood function for the univariate models, so computing these quantiles does not add any significant computational burden. Then at each location and time where we have a random effect, we collect the corresponding quantiles for each species and place a multivariate distribution on those quantiles to model the dependence structure between species. We explore the behaviour of the multivariate model with an extended simulation study.

Chapter 2

The Conditionally Autoregressive Hidden Markov Model (CarHMM): Inferring Behavioural States from Animal Tracking Data Exhibiting Conditional Autocorrelation

2.1 Abstract

One of the central interests of animal movement ecology is relating movement characteristics to behavioural characteristics. The traditional discrete-time statistical tool for inferring unobserved behaviours from movement data is the hidden Markov model (HMM). While the HMM is an important and powerful tool, sometimes it is not flexible enough to appropriately fit the data. Data for marine animals often exhibit conditional autocorrelation, self-dependence of the step length process which cannot be explained solely by the behavioural state, which violates one of the main assumptions of the HMM. Using a grey seal track as an example, along with multiple simulation scenarios, we motivate and develop the conditionally autoregressive hidden Markov model (CarHMM), which is a generalization of the HMM designed specifically to handle conditional autocorrelation.

In addition to introducing and examining the new CarHMM, we provide guidelines for all stages of an analysis using either an HMM or CarHMM. These include guidelines for pre-processing location data to obtain deflection angles and step lengths, model selection, and model checking. In addition to these practical guidelines, we link estimated model parameters to biologically meaningful quantities such as activity budget and residency time. We also provide interpretations of traditional “foraging" and “transiting" behaviours in the context of the new CarHMM parameters.

2.2 Introduction

The study of animal movement is fundamental to ecology because it is inherently linked to critical processes that scale from individuals to populations and communities to ecosystems [Hooten et al., 2017a]. Rapid technological advancements over the past several decades have given rise to a variety of electronic tracking devices that can remotely monitor animals in challenging environments [Hussey et al., 2015] as well as an assortment of statistical methods for analyzing the resulting (big) movement data.

Statistical models for animal movement data are most commonly formulated in discrete time [Hooten et al., 2017b], and are increasingly aimed at inferring behavioural "states" from observed tracks. In this context, the data (called tracks, or location data) generally consist of a regularly observed time series of locations of an animal. Inferring behavioural states from location data was initially made possible by a proposal in [Morales et al., 2004] to transform the data into a bivariate series of step lengths and deflection angles. In their example, they use characteristics of the step length and deflection angle series to determine when an elk is in an "encamped" state and when it is in an "exploratory" state.

There are many different ways to estimate behavioural states from this type of tracking data. While traditionally achieved using likelihood methods (frequentist or Bayesian), any unsupervised classification method can be used. Some examples are mixture models [Morales et al., 2004], clustering models, and k-means clustering [Curry, 2014]. If researchers have actually observed an animal's behaviour at some points in time (for example through recorded video), then any supervised classification method could also be used.

The work of [Morales et al., 2004] along with that of [Jonsen et al., 2005] popularized the state-switching model framework into the *de facto* way of analysing animal movement data in discrete time [McClintock et al., 2012, Whoriskey et al., 2017, Patterson et al., 2017]. While not all animal movement models which incorporate state-switching into the movement process have distinct behavioural states, the ones that do generally fall under the hidden Markov model (HMM) framework. These models assume that there are underlying behaviours driving the animal movement

process [Michelot et al., 2016].

Hidden Markov models for animal movement have a number of desirable properties: they have an easily computable likelihood which is typically fast to optimize, the model parameters have clear interpretations, and they can fairly easily handle different types of data (including missing data) in the same model [Zucchini et al., 2016]. The baseline formulation of the HMM has a few key assumptions: the underlying state process is assumed to form a Markov chain, and the observed step lengths and deflection angles are conditionally independent given the behavioural state.

The effect of various violations of these assumptions are discussed in [Pohle et al., 2017]. They found that neglecting a semi-Markov state process (which directly models state residency time), a higher order Markov chain for the behavioural process, or violations of conditional independence can introduce bias to parameter estimates and favour models which have more behavioural states than actually exist. Semi-Markov state processes are also considered in [Langrock et al., 2012] while higher order state processes are presented in [Zucchini et al., 2016].

The papers in the recent Journal of Agricultural, Biological, and Environmental Statistics special edition on animal movement also started to address other problems associated with the discrete-time model framework in general, such as telemetry error, irregularly spaced data, and occasional missing data [McClintock, 2017], the temporal scale and resolution of the behaviours involved in the data [Leos-Barajas et al., 2017], and choosing the number of behavioural states to use [Pohle et al., 2017]. Methods for assessing goodness of fit for animal movement models were discussed in [Potts et al., 2014], wherein they found that none of the 20 highest cited papers at the time tested goodness of fit to the data. Since then, the moveHMM [Michelot et al., 2016] and momentuHMM [McClintock and Michelot, 2018] `R` packages have implemented easy to use residuals, although the use of residuals in the literature is still uncommon, or at least under-reported.

The current paper introduces a conditionally autoregressive hidden Markov model (CarHMM) that does not require the assumption of conditional independence of the movement process given the behavioural process. We do this by introducing an

autocorrelation parameter in the step length distribution of the traditional HMM (such as that implemented in [Michelot et al., 2016]). The use of an autocorrelated step length process was also present in a continuous-state model for estimating the effect of environmental covariates on behavioural memory in [Forester et al., 2007].

Throughout, we provide general practice guidelines wherever possible. Since analyses of animal movement typically use offline data, we propose standardizing the observed step lengths by dividing by the mean observed step length. This allows comparison of models across data sources, animals, species, etc. We use a lag-plot of step length for determining if the conditional autocorrelation is necessary, and note its possible use to help in choosing the number of behavioural states to use. In the case of irregularly observed tracks, we also discuss how to choose an appropriate interpolation time step for the model, as well as how to deal with extensive missing data by grouping observations.

In Section 2.3, we present the formulation of the model including computation of the likelihood and give references for the theoretical properties. Section 2.4 discusses the biology associated with specific transformations of the model parameters. Section 2.5 deals with pre-processing the locations by choosing a time step and dealing with missing data, as well as model selection and validation. Both Section 2.4 and Section 2.5 are useful for most types of discrete-time models, including the HMM and the CarHMM. Section 2.6 presents four short simulation studies. Section 2.7 demonstrates best practice for using the CarHMM through the analysis of a male grey seal.

2.3 CarHMM Formulation

We assume that the data consist of a set of step lengths $d_{(t,t+1)}$ between locations at time t to $t+1$ and deflection angles θ_t between locations at times $t-1$, t, and $t+1$. Locations are assumed to be observed on a discrete and evenly spaced time grid. Here, step lengths measure the distance between consecutive locations, and deflection angles measure the angular change in direction between three locations. We discuss observations which are irregularly spaced in Section 2.5.1.

We introduce a behavioural state process B_t which is a Markov chain on a finite set of

states $\{1,\ ...,\ k\}$. Thus the distribution for B_t is completely determined by the value b_{t-1} of B_{t-1} and the transition probability matrix $\mathbf{A}$. The $(i,j)^{\text{th}}$ entry $a_{i,j}$ of $\mathbf{A}$ gives the probability of transitioning from state i to state j. We assume (other choices are possible) that the initial distribution of the behavioural state Markov chain is given by the stationary distribution $\boldsymbol{\delta}$, which is the vector such that $\boldsymbol{\delta}\mathbf{A} = \boldsymbol{\delta}$ and $\sum_i \delta_i = 1$.

Given the behavioural state at time t, the step length at time $(t, t+1)$ and deflection angle at time t are assumed to be conditionally independent of all other observations and behavioural states, with the key exception that step length at time $(t, t+1)$ is allowed to be dependent on step length at time $(t-1, t)$. A first order autoregressive process is assumed for step lengths $d_{(t,t+1)}$. While any valid distributions can be used, in this presentation we assume a gamma (Γ) distribution for step lengths and a wrapped Cauchy (WC) distribution for deflection angles θ_t. A $\Gamma\left[(1-\phi)\cdot\mu_{RL} + \phi\cdot d_{(t-1,t)},\ \sigma\right]$ distribution has mean $\mu = (1-\phi)\cdot\mu_{RL} + \phi\cdot d_{(t-1,t)} > 0$ (reversion level $\mu_{RL} > 0$, autocorrelation $0 < \phi < 1$) and standard deviation $\sigma > 0$, with the more traditional shape and scale parameters being $(\mu/\sigma)^2$ and σ^2/μ, respectively. The $\text{WC}\,(c,\ \rho)$ distribution has center $c \in [-\pi,\ \pi]$ and concentration $\rho \in (0, 1)$ with density function

$$f\,(\theta;\ c,\ \rho) = \frac{1}{2\pi}\cdot\frac{1-\rho^2}{1+\rho^2-2\rho\cdot\cos\,(\theta-\mu)}. \tag{2.1}$$

In cases where we have all of the data before analysis (i.e. we are not streaming the data), we standardize all step lengths by dividing by the observed mean step length. This removes units (for example, kilometers) and standardizes parameter interpretation across data sources, animals, species, etc. Comparison of standardized parameter estimates across data sources is dependent on many factors, including the temporal resolution of each data source, choices made during the model procedure, and the biology/ecology of the animals being compared. Conditional on the mean observed step length, dividing by the mean does not alter parameter inference since dividing a gamma distribution by a (non-zero) constant results in another gamma distribution. In practice, we store the observed mean step length so we can un-standardize later if desired. For the rest of the paper, we will assume that the symbol $d_{(t,t+1)}$ stands for standardized step length.

With all of the terms now defined, the CarHMM is formulated as

$$\begin{aligned}
\text{Location:} \quad & \mathbf{x}_{t+1} = \mathbf{x}_t + d_{(t,t+1)} \cdot \mathbf{H}\left(\theta_t\right) \cdot \left[d_{(t-1,t)}^{-1} \cdot \left(\mathbf{x}_t - \mathbf{x}_{t-1}\right)\right] \\
\text{Action:} \quad & d_{(t,t+1)} \mid B_t = b \sim \Gamma\left((1-\phi_b)\cdot\mu_{RL,b} + \phi_b \cdot d_{(t-1,t)},\ \sigma_b\right) \\
& \theta_t \mid B_t = b \sim \mathrm{WC}\left(c_b,\ \rho_b\right) \\
\text{Behaviour:} \quad & \Pr\left[B_t = j \mid B_{t-1} = i\right] = a_{ij}, \quad i,j \in \{1,2,...,k\} \\
\text{Initial Conditions:} \quad & \Pr\left[B_1 = i\right] = \delta_i, \qquad d_{(0,1)} \text{ is fixed from the data as the first observed step length.}
\end{aligned} \tag{2.2}$$

Although the locations $\mathbf{x}_t$ themselves do not enter the likelihood for the model directly, we include the "Movement" equation to show the connection between the locations and the step lengths and deflection angles. In this equation, $\mathbf{H}\left(\theta_t\right)$ represents the change in direction at time t. If, as we strongly recommend, the coordinates are latitude-longitude pairs, then the equation as given is more of a symbolic representation. The fully written equation is based on spherical geometry. If the coordinates are projected, then $\mathbf{H}$ can be written as a standard 2×2 rotation matrix.

The likelihood is computed as the matrix product

$$L = \boldsymbol{\delta}\mathbf{L}\left(d_{(1,2)}, \theta_1\right)\mathbf{A}\mathbf{L}\left(d_{(2,3)}, \theta_2\right)\mathbf{A}\cdots\mathbf{L}\left(d_{(n,n+1)}, \theta_n\right)\mathbf{1}_{k\times 1} \tag{2.3}$$

where $\mathbf{L}\left(d_{(t,t+1)}, \theta_t\right)$ is the diagonal matrix

$$\mathrm{Diag}\left[f\left(d_{(t,t+1)},\ \theta_t \mid B_t = 1\right),\ ...,\ f\left(d_{(t,t+1)},\ \theta_t \mid B_t = k\right)\right], \tag{2.4}$$

and $\mathbf{1}_{k\times 1}$ is a vector of ones. Since $d_{(t,t+1)}$ and θ_t are considered conditionally independent given the behavioural state, their joint probability density function is the product of the individual densities. In practice, we compute the log-likelihood using forward recursion with scaling as presented in Section 3.2 of [Zucchini et al., 2016].

When the autocorrelation ϕ_b is fixed at 0 for all b, the CarHMM reduces to a standard HMM. In addition, when ϕ_b is fixed at 1 for all b, it is possible to show that the

CarHMM reduces to a component-wise relative of the hidden Markov movement model of [Whoriskey et al., 2017] though the details will not be shown here. Further, other generalizations to the standard movement HMM such as adding a semi-Markov state process could be applied to the CarHMM.

When using the Γ distribution $\mu_{RL,b}$ must be non-negative and ϕ_b must be within the unit interval. Another useful choice of distribution for step length is the log-normal distribution, where the log of the step length has mean $(1-\phi_b)\cdot\mu_{RL,b}+\phi_b\cdot d_{(t-1,t)}$. In this case the parameters are unrestricted. However to ensure the step length process is stable, it is sufficient (but not necessary) that the estimates satisfy $\left|\hat{\phi}_b\right|<1$ for all b. If this is not the case it may be a sign of numerical instability in the optimizer.

We use the maximum likelihood framework; the parameters to be estimated are $\mu_{RL,b}$, ϕ_b, σ_b, c_b, ρ_b for $b\in 1,\ ...,\ k$ giving $5k$ parameters for the "Action" distribution, and the off-diagonal transition probabilities $a_{i,j}$ for $i,j\in 1,\ ...,\ k,\ i\neq j$ giving $k\cdot(k-1)$ parameters for the "Behaviour" distribution, for a total of k^2+4k parameters. The remaining transition probabilities are not free parameters since the row sums of $\mathbf{A}$ must equal 1. The unobserved behavioural states B_t are predicted using the well known Viterbi algorithm, see e.g. [Zucchini et al., 2016].

Identifiability of models in the Markov-switching autoregressive class, which includes the CarHMM, is proven in [Douc et al., 2004], with consistency and asymptotic normality of the ML estimates when using the log-normal distribution resulting from the same paper. Consistency and asymptotic normality when using the Γ distribution is studied in [Ailliot, 2006]. One notable consistency condition is that the entries of $\mathbf{A}$ must be strictly positive for parameter estimation to be consistent. If any estimated value is close to zero, this could be a sign of having too many states in the model, unless there is a biologically meaningful reason for including the extra state.

2.4 Interpretation of CarHMM Parameters

Here we discuss the concepts of activity budget, behavioural residency time, and mean reversion level. These are all obtained as transformations of the model parameters and are related to the biology of the animal.

Both activity budget and behavioural residency time can be obtained from the transition probability matrix $\mathbf{A}$. The stationary distribution $\boldsymbol{\delta}$ itself can be interpreted as an activity budget, where the i^{th} entry of $\boldsymbol{\delta}$ gives the expected proportion of time that the animal spends in the i^{th} behavioural state. For example, the activity budget can give estimates of the proportion of time spent transiting as compared to foraging.

Behavioural residency time is the amount of time that an animal will remain in a given behavioural state before switching to a different state. These can be modelled explicitly using semi-Markov state processes, though in our case they follow a geometric distribution [Langrock et al., 2012]. For a geometric distribution, the expected number of time steps spent in state i is given by $\boldsymbol{E}_t(\text{state } i) = 1 \ / \ (1 - p_{i,i})$. When converted to real time units (hours, minutes, etc.) by multiplying by the chosen time step, this value gives an estimate of the time scale of the behaviour being modelled and is important in giving biologically meaningful interpretations to the behavioural states.

Since the step length process within a given behavioural state follows a first-order autoregressive process, the parameter $\mu_{RL,b}$ gives the reversion level of the process for a given state. The mean reversion level is defined to be the expected value of step length as the time spent within the behavioural state approaches infinity.

$$\mu_{RL,b} = \lim_{t \to \infty} \boldsymbol{E} \left(D_{(t,t+1)} \,\middle|\, B_{t'} = b \ \forall \ t' \leq t \right). \tag{2.5}$$

Within each behavioural state this value acts as an attractor in the step length distribution. Thus consecutive step lengths in the same behavioural state will tend to converge on this value, with the strength of attraction being inversely proportional to ϕ_b and proportional to the distance of the previous step length to $\mu_{RL,b}$.

One of the attractive features of hidden Markov models for movement data is the connection between the underlying behavioural state of the Markov chain with behaviours exhibited by the animal. While the connection between the behavioural state in the model and the behaviours exhibited by the animal is sometimes tenuous, it can be useful to label the behavioural states of the model. Two common “behaviours" used in the HMM context are “foraging" and “transiting" (quotations are used to emphasise that these labels may not reflect actual behaviour).

In the standard HMM, foraging is typically characterized by short step lengths and diffuse deflection angles. Transiting is typically characterized by long step lengths and deflection angles concentrated at zero degrees. We can update these behaviours in the CarHMM framework. Here foraging is characterized by short step lengths with little autocorrelation along with diffuse deflection angles. Transiting is characterized by longer and highly autocorrelated step lengths along with deflection angles concentrated at zero degrees.

2.5 Data Inspection and Model Checking

2.5.1 Pre-Processing Locations

When dealing with marine animal tagging data the observed locations are typically not on a regular time grid. In order to use the CarHMM, we linearly interpolate the observed locations to a regular time grid. An alternative is to use the multiple imputation approach proposed in [McClintock, 2017]. However even when using this multiple imputation approach one must choose a sensible time grid.

Interpolation requires making a few decisions such as: how are observations which are very far apart in time (long stretches of missing data) dealt with? and, what is the best time step (the time between points on the temporal grid) to use for the interpolation? We deal with the first by splitting the track into separate groups whenever the time between consecutive observations is greater than some cutoff level, which we call the group cutoff level. After defining the time step, which we require to be the same for all groups, and the group cutoff level the observed locations are interpolated within their separate groups on to the regular time grid. The interpolated locations are then processed to obtain the deflection angles and step lengths which enter the likelihood.

There are two metrics we propose to help in choosing a time step and group cutoff level: the proportional sample size

$$n_{\text{prop}} = \frac{\#\text{ interpolated locations}}{\#\text{ observed locations}} \tag{2.6}$$

and the adjusted proportional sample size

$$n_{\text{adj}} = \frac{\#\text{ interpolated locations} - 2 \cdot \#\text{ groups}}{\#\text{ observed locations} - 2}. \tag{2.7}$$

The first is designed to preserve the number of locations in the track, while the second is designed to preserve the number of data points which enter the likelihood.

To choose a (heuristically) best time step and group cutoff level, we recommend:

1. restrict the time step to be somewhere between the median and 3^{rd} quartile of the observed time differences in the original data;

2. for whichever time step is chosen, restrict the group cutoff level to be no more than twice the time step (and no less than the time step itself);

3. with the above restrictions, set up a grid of time steps and group cutoff levels and compute n_{prop} and n_{adj} for each point in the grid. Choose whichever makes both n_{prop} and n_{adj} as close to 1 as possible.

We follow these guidelines in choosing the time step for the best practice analysis of Section 2.7. These guidelines are meant to avoid both over-smoothing the data (and therefore losing information) by choosing too large a time step, or inadvertently replicating the data by choosing too small a time step. Values of n_{prop} or n_{adj} less than one are indicators of over-smoothing while values greater than one are indicators of data replication.

A brief experiment suggested that interpolation (either linear interpolation or with a variety of different splines) may introduce a significant amount of autocorrelation in the step lengths. Further investigation is needed to determine the exact effects of interpolation. Whether most of the autocorrelation is inherent to the track or is introduced by interpolating the locations, accounting for it in the model (such as the CarHMM does) is necessary.

Once the locations are interpolated and grouped with a regular time step, they are processed to obtain deflection angles and step lengths. These should be obtained from unprojected coordinates so that both the deflection angles and step lengths

are accurate throughout the spatial extent of the data. The model is fitted to all of the grouped data in a single likelihood, assuming that groups are independent of each other and that the groups share the same true parameters. Within a group, the first step length of that group is taken to be the initial condition for the step length autoregressive process, and the initial distribution of the underlying state is always taken to be the stationary distribution.

2.5.2 Model Selection

We consider two components to model selection for the CarHMM: deciding whether to use the HMM or the CarHMM (i.e., whether to fix $\phi_b = 0$ or not), and choosing the number of behavioural states.

First we introduce the lag-plot, an exploratory graphic useful for understanding the nature of any autocorrelation present in the step length process. The lag-plot at lag k is a kernel-density plot of $d_{(t,t+1)}$ against $d_{(t-k,t+1-k)}$. Examples of these plots are given in Figure 2.3 of Section 2.6. These lag-plots give a more detailed description of the autocorrelation than the simple autocorrelation function, and have a couple of immediately helpful uses.

Most importantly in the current case, by examining a lag-plot at lag 1, it can be possible to determine which of the HMM and CarHMM is more appropriate for the data. These plots show the different types of autocorrelation present within the HMM and CarHMM, compared with a real dataset. The HMM plot will have a pattern of distinct circular droplets along the line $y = x$, a result of the autocorrelation in the behavioural states, while the CarHMM plot will have an elongated smear along the line $y = x$, due to the within-state autocorrelation in step lengths.

In ideal cases, the lag-plot at lag 1 may also suggest the number of states exhibited in the data. Particularly for data with HMM-like autocorrelation, the number of distinct droplets corresponds to the number of distinct states. This becomes more complicated with more latent states and with more CarHMM-like autocorrelation. Choosing the number of states to use is a notoriously difficult problem, with traditional metrics such as AIC and BIC generally selecting too many states to be biologically meaningful. For in-depth discussion, we refer the reader to [Pohle et al., 2017]. A recommended

starting point is to use as few states as necessary to achieve an adequate fit.

Other uses of the lag-plot include comparing the autocorrelation characteristics of different time step choices. For example, we could test the intuitively attractive idea that a short time step results in step lengths with high within-state autocorrelation, while a long time step results in step lengths with low within-state autocorrelation. Further, with multi-state models the traditional autocorrelation function can do a poor job of quantifying the autocorrelation. The lag-plot includes more detail such that the characteristics of each state can be discerned.

2.5.3 Residuals

For model checking we follow the probability scale residuals of [Shepherd et al., 2016] and in particular use one-step-ahead forecast residuals. Here, the forecast distribution for step length is a mixture of gamma distributions with means $(1 - \phi_b)\mu_{RL,b} + \phi_b d_{(t-1,t)}$, standard deviations σ_b, and mixture rates given by the b_{t-1}^{th} row of $\mathbf{A}$.

If we have specified the model structure correctly then these residuals should be uniformly distributed on (-1,1) and exhibit no autocorrelation. Further, they should have this property within each behavioural state, though small sample size can be a problem here. Departures from a uniform distribution can be detected by looking at a quantile-quantile (Q-Q) plot of the residuals. Residual autocorrelation can be identified in plots of the autocorrelation function of these residuals, or in lagplots such as those proposed for model selection.

2.6 Simulation Study

To address the performance and properties of the new CarHMM model, we present brief vignettes of four simulation studies. First, we investigate the effect of track length. Second, we look at the effect of within-state autocorrelation. The third study looks at the effect of the transition probabilities. The fourth and final study compares the regular HMM and the CarHMM. When simulating data we simulate both the underlying behavioural state and observed data from scratch. We do not reuse the behavioural states estimated from original data.

The main metric we use to assess these simulations is the interquartile range (first and third quartiles) of the state estimate error. For a particular track, the state estimate error is the percentage of state estimates which do not agree with the true simulated state. The quartiles are then computed over e.g. 50 simulations under the same model.

To account for numerical instability in the maximization of the likelihood, our fitting procedure for these studies attempts to fit the model to each simulated track at most 10 times, with different random starting values each time, until the model converges. If the model does not converge successfully on any of those 10 attempts, we remove that particular track from consideration. We also remove tracks which give unreasonable parameter estimates (in particular, any model fit which gives a stationary distribution with an entry less than 0.01), or estimates which give clear signs of numeric instability (deflection angle concentration less than 10^{-3}, or a transition probability matrix with any row having all equal entries). When using real data we can tweak exactly how we optimize the likelihood (change various control parameters, pick starting values, etc.) so this practice is not an inherent shortcoming of the model or fitting method.

The Viterbi algorithm is currently the most common way to estimate behavioural states in HMM-like models. Briefly, the algorithm takes as input parameter point estimates and the observed data, and outputs the most likely sequence of behavioural states as a point estimate. With the Viterbi algorithm, the accuracy of the state estimates is dependent on the amount of overlap of the state-dependent distribution. If two states have significant overlap, the Viterbi algorithm will perform much worse than if the two states were distinct.

At no point are standard errors of parameter estimates or uncertainty statements about the behavioural states considered. Because of this, any error in the parameter estimates directly translates to a source of error in the state estimates, with no hope of correcting for the uncertainty in the parameter estimates. Because the Viterbi algorithm only gives the most likely sequence of behavioural states with no uncertainty estimates, the estimated behavioural states must be interpreted with care. In addition to the tenuous connection between the behavioural state labels and the biology, there is the additional problem that we do not know how likely the most likely behavioural state path is. Simulations, such as the ones below, can help determine how much error

to expect. However, since the actual states are unobserved, it is not possible to know the actual error rate.

2.6.1 Effect of Track Length

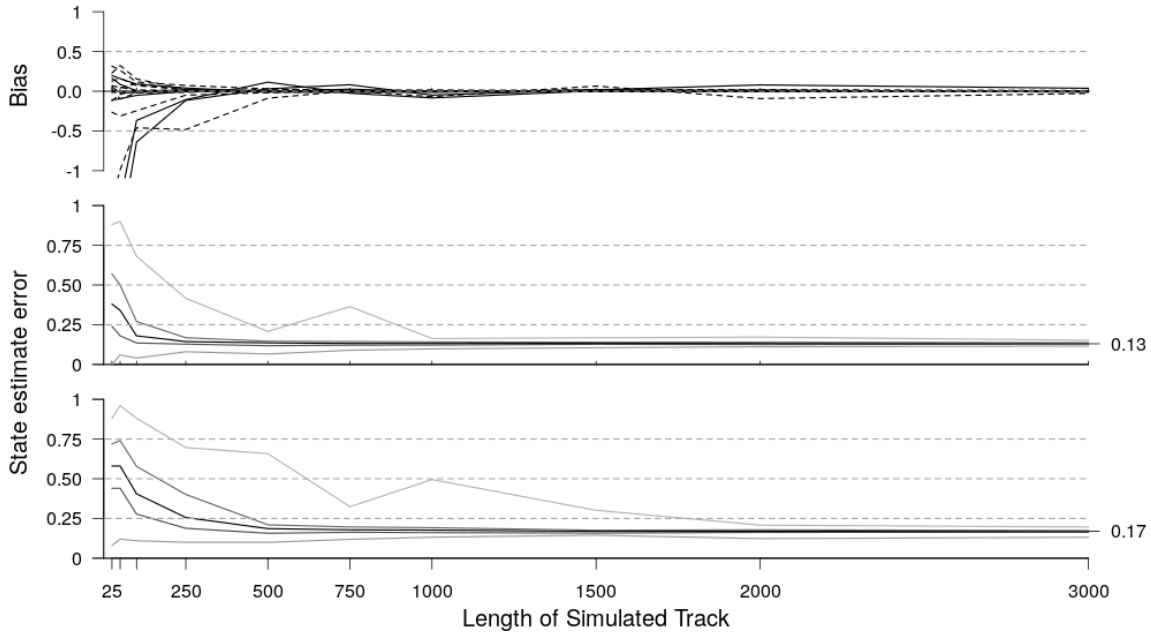

Figure 2.1: The top panel gives the bias for simulated tracks of different lengths under the same parameters, for both a two- and three-state model. The important feature is that the bias for all parameters converges to zero (∼500 locations), showing that the parameters can be successfully estimated given a long enough track. The bottom two panels give the state estimate error five number summary (min, median, max, and quartiles). Each track length used 50 simulations. In both the two state model and the three state model, the median error rate quickly stabilizes (∼250 observations for the two state model, ∼500 for the three state model), but does not converge to zero.

In practice, one would hope that collecting more data (i.e. longer animal tracks) would decrease the amount of error in both parameter estimates and state estimates. Simulations suggest that, while error in parameter estimates will decrease with longer track lengths, there is an inherent amount of error to be expected in behavioural state estimates that cannot be overcome with increased track length.

Figure 2.1 shows the bias and state estimate error for two different models. Here we compute bias as the median difference, across simulations, of parameter estimates from

the true parameter. The two state model is a HMM which takes (slightly modified) parameters estimated from an elk dataset analyzed in the vignette for the R package moveHMM [Michelot et al., 2016]. The three state model is a CarHMM which takes parameters estimated from a grey seal dataset (different from the one presented in Section 2.7). The parameters for both models can be found in Table 2.1 and Table 2.2, respectively.

Two State Elk Parameters

$\mathbf{d_{(t,t+1)}}$	State 1	State 2	$\boldsymbol{\theta_t}$	State 1	State 2	$\mathbf{A}$	$\mathbf{p}_{\cdot,\mathbf{1}}$	$\mathbf{p}_{\cdot,\mathbf{2}}$
$\boldsymbol{\mu}_{\mathbf{RL,b}}$	3.364	0.355	$\mathbf{c}$	0	0	$\mathbf{p}_{\mathbf{1},\cdot}$	0.75	0.25
$\boldsymbol{\phi}_{\mathbf{b}}$	0	0	$\boldsymbol{\rho}$	0.228	0.6	$\mathbf{p}_{\mathbf{2},\cdot}$	0.15	0.85
$\boldsymbol{\sigma}$	4.329	0.378						

Table 2.1: Parameters for the two state elk simulations. Low autocorrelation is 0.1, medium autocorrelation is 0.4, high autocorrelation is 0.85. These parameters are slightly modified from the original source, the vignette to the R package moveHMM [Michelot et al., 2016].

Three State Seal Parameters

$\mathbf{d_{(t,t+1)}}$	State 1	State 2	State 3	$\boldsymbol{\theta_t}$	State 1	State 2	State 3
$\boldsymbol{\mu}_{\mathbf{RL,b}}$	0.202	0.998	2.091	$\mathbf{c}$	0	0	0
$\boldsymbol{\phi}_{\mathbf{b}}$	0.04	0.429	0.945	$\boldsymbol{\rho}$	0.209	0.681	0.867
$\boldsymbol{\sigma}$	0.157	0.529	0.235				

$\mathbf{A}$	$\mathbf{p}_{\cdot,\mathbf{1}}$	$\mathbf{p}_{\cdot,\mathbf{2}}$	$\mathbf{p}_{\mathbf{3},\cdot}$
$\mathbf{p}_{\mathbf{1},\cdot}$	0.848	0.142	0.005
$\mathbf{p}_{\mathbf{2},\cdot}$	0.065	0.754	0.164
$\mathbf{p}_{\mathbf{3},\cdot}$	0.005	0.164	0.831

Table 2.2: Parameters for the three state seal simulations.

Figure 2.1 shows that collecting more and more data for a single track is not effective past a certain point. For the remaining simulation studies we use track lengths of 1,000. We report the first and third quartiles of the state estimate error, which Figure 2.1 suggests will have stabilized at this track length.

2.6.2 Effect of Autocorrelation

The accuracy of the Viterbi algorithm depends heavily on the amount of overlap of the state-dependent distributions. Recall that the mean of the step length distribution is given by $(1-\phi)\cdot\mu_{RL}+\phi\cdot d_{(t-1,t)}$. Consider the autocorrelation ϕ as a weight between μ_{RL}, which will depend on the state at time t, and $d_{(t-1,t)}$, which does not. If ϕ is close

		State 2		
		Low	Med	High
State 1	Low	(0.131, 0.148) [4]	(0.111, 0.128) [2]	(0.074, 0.087) [0]
	Med		(0.139, 0.161) [5]	(0.082, 0.098) [0]
	High			(0.209, 0.240) [0]

Table 2.3: First and third quartiles for the state estimate error for different combinations of low, medium, and high autocorrelation. The number in square brackets gives the number of simulations which did not converge, out of 100 simulations.

		State 2 ($\phi = 0.892$)			
		0.5	0.6	0.7	0.9
State 1 ($\phi = 0.407$)	0.5	(0.214, 0.234) [1]	(0.204, 0.227) [0]	(0.188, 0.207) [1]	(0.125, 0.154) [2]
	0.6	(0.226, 0.240) [3]	(0.204, 0.223) [2]	(0.194, 0.210) [0]	(0.127, 0.147) [2]
	0.7	(0.224, 0.245) [2]	(0.207, 0.223) [2]	(0.196, 0.218) [2]	(0.115, 0.134) [0]
	0.9	(0.213, 0.242) [6]	(0.179, 0.226) [6]	(0.154, 0.180) [1]	(0.082, 0.104) [2]

Table 2.4: The row and column headings give the probability of staying within the given state from one time to the next. First and third quartiles for the state estimate error. The number in square brackets gives the number of simulations which did not converge, out of 20 simulations. The amount of error decreases as the probability of remaining in state 2 increases. The error is not significantly affected by the probability of remaining in state 1.

to one in two states with drastically different μ_{RL}, then the two states will overlap since μ_{RL} is essentially irrelevant in both states.

Table 2.3 shows the state error rate for a two state model with different amounts of autocorrelation (each state taking either a low, medium, or high amount). The parameters are modified from the same elk example used in Section 2.6.1. Overall we see that increasing the autocorrelation of both states leads to an increase in the amount of state estimate error, while differentiating the amount of autocorrelation between the two states leads to a decrease in the amount of error.

2.6.3 Effect of Transition Probabilities

Unlike in the standard HMM, the observed state-dependent distributions for the CarHMM are indirectly affected by the transition probabilities of the underlying behavioural states. States with low autocorrelation act as anchors in the step length series, while states with high autocorrelation tend to wander. The longer that an animal is in a state with high autocorrelation (by having a high probability of remaining

		Two State Model	
Simulated Model		HMM	CarHMM
Fitted	HMM	(0.120, 0.138) [8]	(0.434, 0.474) [0]
Model	CarHMM	(0.125, 0.145) [7]	(0.072, 0.083) [0]

		Three State Model	
Simulated Model		HMM	CarHMM
Fitted	HMM	(0.044, 0.058) [15]	(0.373, 0.445) [30]
Model	CarHMM	(0.047, 0.060) [3]	(0.157, 0.186) [2]

Table 2.5: First and third quartiles for the state estimate error. The number in square brackets gives the number of simulations which did not converge, out of 100 simulations. When the data is simulated with no within-state autocorrelation, the HMM and the CarHMM have essentially the same error rate. However, when the data is simulated with within-state autocorrelation, the HMM performs very poorly compared to the CarHMM.

in the same state), the more we expect that step length series to wander.

Figure 2.2 gives observed step length distributions for a variety of different transition probabilities. Table 2.4 gives state estimate error under the same variety of transition probabilities. We see that the probability of remaining in state 2 (with high autocorrelation) affects the state estimate error, as this probability is what determines how free the second state is to wander. The more that the high autocorrelation state drifts away from the mean of the low autocorrelation state (from left to right in the table), the less overlap there is in their distribution, which increases the accuracy of the Viterbi algorithm. The parameters can be found in Table 2.6.

2.6.4 Comparison of HMM and CarHMM

To show the importance of accounting for conditional autocorrelation in the data, we simulate data under both the HMM and the CarHMM and fit both the HMM and CarHMM to each simulation. We use parameters from two different datasets: the "Low-High" two state parameters from the elk track considered in subsection 2.6.2, and three state parameters estimated from the grey seal track analyzed in Section 2.7. The parameters for the models can be found in Table 2.1 and Table 2.7 respectively. Figure 2.3 shows example lag-plots under the HMM and the CarHMM for the three state model. As mentioned earlier, these plots can help in model selection.

Table 2.5 shows the state estimation error rate for the four different scenarios. The CarHMM is just as effective as the HMM when fitted to HMM data with no conditional autocorrelation. However, the two-state HMM ($\sim 40 - 45\%$ error) performs only slightly better than random guessing (50% error) when fitted to CarHMM data with conditional autocorrelation. We expect this amount of error to persist across models that have at least one state with significant autocorrelation. The three-state HMM has the same problem, although performs much better than the $\sim$66% error expected from random guessing.

These simulations raise interesting questions about the validity of previous research utilizing hidden Markov models with irregularly timed data, especially since we suspect a non-trivial amount of autocorrelation is introduced through interpolating the locations to a regular grid. However, we only mention this point and leave the discussion for another time.

Computation Time and Implementation

All simulations were computed on a laptop running Linux with a quadcore Intel Core i7-7500U CPU with 8GB of RAM. To compare the computation speed of the HMM with the CarHMM, we timed how long it took to fit a three state HMM and a three state CarHMM 100 times to the seal data in Section 2.7. We also timed how long it took to simulate and refit 100 simulations from each model. The HMM averaged 2.43 seconds per fit, and an additional 0.77 seconds per simulation. The CarHMM averaged 2.37 seconds per fit, and an additional 1.23 seconds per simulation. The difference in computation time between the two models is essentially negligible. Our implementation of the CarHMM uses the R package Template Model Builder [Kristensen et al., 2016], which allows for fast computation through automatic differentiation. It also has the ability to fix parameters at given values, allowing our HMM and CarHMM implementation to be identical. Our implementation and other functional tools discussed earlier are available as an R package at the first author's GitHub page. This package also includes the data used in Section 2.7.

2.7 Best Practice Analysis of a Male Grey Seal Track

In this section we demonstrate what we now consider to be basic best practice for analyzing animal movement data and reporting subsequent results.

The data are a subset of a male grey seal track on the Scotian shelf, analyzed previously in [Whoriskey et al., 2017]. The seal was tracked using GPS with negligible observation error. Due to some data collection issues (the median time differences abruptly change without explanation) we will look at only the final 3,158 locations with time differences having a mean, median, and 3rd quartile of 100, 64, and 122 minutes, respectively.

First, one must choose values for the time step and group cutoff. To do this, set up a grid of values where the time step ranges from 60 minutes to 120 minutes by increments of 3 minutes, and the group cutoff ranges from the time step to twice the time step in increments representing 5% of the time step. This range of values for the time step is chosen to range approximately from the median to the 3rd quartile. Refer to Section 2.5.1 for more detail.

Both metrics for choosing a good time step and group cutoff discussed in Section 2.5.1 chose an optimal time step of 66 minutes and group cutoff of 132 minutes. The resulting interpolated track consists of 3,129 locations in 251 groups with $n_{\text{prop}} = 0.991$ and $n_{\text{adj}} = 0.832$. The mean of the unstandardized step lengths is 2.10 kilometres per time step (1.91 km/hr).

The most useful plot of the data is the lag-plot of $d_{(t,t+1)}$ vs. $d_{(t-1,t)}$ and is shown as part of Figure 2.4. This plot shows the smeared texture that is characteristic of the CarHMM. The residuals for a two state CarHMM have autocorrelation on the border of significance. Neither a three state or a four state CarHMM give improved residuals (not shown). The data nor any of the residuals showed evidence of long-term seasonality. Given no other reason to choose a specific number of states, we recommend using the least number of states which you feel accurately describe the data.

We also remind the reader that behavioural state labels such as “foraging" and “transiting" may not be reflective of the actual biology.

Two State "Best Practice" Seal Parameters

$\mathbf{d_{(t,t+1)}}$	State 1	State 2	$\boldsymbol{\theta_t}$	State 1	State 2	$\mathbf{A}$	$\mathbf{p_{\cdot,1}}$	$\mathbf{p_{\cdot,2}}$
$\boldsymbol{\mu_{RL,b}}$	0.552	1.731	$\mathbf{c}$	0	0	$\mathbf{p_{1,\cdot}}$	0.826	0.174
$\boldsymbol{\phi_b}$	0.407	0.892	$\boldsymbol{\rho}$	0.508	0.858	$\mathbf{p_{2,\cdot}}$	0.12	0.88
$\boldsymbol{\sigma}$	0.351	0.244						

Table 2.6: Parameter estimates for a male grey seal track using the two state CarHMM. These parameters are also used for the two state simulations studying the effect of the transition probabilities.

Two state model The parameter estimates are given in Table 2.6. State 2 is interpretable as a "transiting" behaviour. The autocorrelation parameter ($\phi_2 = 0.89$) and concentration parameter ($\rho_2 = 0.86$) are suitably high, and the standard deviation ($\sigma_2 = 0.244$; $\sigma_2/\mu_{RL,2} = 0.14$) is suitably low. A map of the state estimates in Figure 2.5 also indicates a "transiting" behaviour.

State 1 does not have as clear an interpretation. It may be tempting to label it a "foraging" behaviour to complement the "transiting" behaviour of State 2, however the parameter estimates for State 1 do not fully support this view. The autocorrelation parameter ($\phi_1 = 0.41$) is not close to 0 and the concentration parameter ($\rho_1 = 0.51$) is higher than expected. Further, the map of the state estimates shows that some of the behaviour picked up by this state does not have traditional "foraging" characteristics. This suggests that State 1 may be picking up two distinct behaviours. We believe these behaviours may be a "foraging" behaviour and a "large area search" behaviour, although many other possibilities may exist. For this reason, we suggest using a third state to further differentiate these behaviours.

Three state model The parameter estimates are given in Table 2.7. State 1 is closer to a "foraging" behaviour than it was in the two state model, and a map of the state estimates places State 1 where we might *a priori* expect "foraging" to take place based solely on the locations. State 3 is archetypal "transiting" behaviour with both ϕ_3 and ρ_3 close to 1. Based on the transition probabilities which do not allow

Three State CarHMM Parameter Estimates

$\mathbf{d_{(t,t+1)}}$	State 1	State 2	State 3	$\boldsymbol{\theta_t}$	State 1	State 2	State 3
$\boldsymbol{\mu_{RL,b}}$	0.398	1.291	2.074	$\mathbf{c}$	-0.129	-0.050	0.002
$\boldsymbol{\phi_b}$	0.277	0.781	0.961	$\boldsymbol{\rho}$	0.402	0.780	0.906
$\boldsymbol{\sigma}$	0.279	0.318	0.164				

$\mathbf{A}$	$\mathbf{p}_{\cdot,1}$	$\mathbf{p}_{\cdot,2}$	$\mathbf{p}_{\cdot,3}$
$\mathbf{p}_{1,\cdot}$	0.713	0.287	0.000
$\mathbf{p}_{2,\cdot}$	0.149	0.797	0.054
$\mathbf{p}_{3,\cdot}$	0.000	0.120	0.880
$\boldsymbol{\delta}$	0.264	0.508	0.228

Table 2.7: Parameter estimates for a male grey seal track using the three state CarHMM.

transitions between State 1 and State 3, one would label State 2 a “transitional" behaviour. Based on parameter estimates and the map of the state estimates given in Figure 2.6 there is no reason to believe that a fourth state is needed.

The expected residency times are: 3.48 timesteps (3 hr 50 min) for the “foraging" behaviour; 4.93 timesteps (5 hr 25 min) for the “transitional" behaviour; and 8.33 timesteps (9 hr 10 min) for the “transiting" behaviour. The expected activity budget gives 26.4% of the seal’s time spent “foraging", 22.8% of its time spent “transiting", and 50.8% of its time transitioning between the two. A simulation study of 93 convergent simulations out of 100 gave first and third quartiles of the state estimate error as $(20.5\%,\ 22.9\%)$.

2.8 Discussion

We have introduced the conditionally autoregressive hidden Markov model (CarHMM) for highly accurate tracking data as an alternative to both the HMM originally developed in [Morales et al., 2004] and the HMMM documented in [Whoriskey et al., 2017].

Subjective choices are often involved during data processing and model fitting. When fitting discrete-time movement models, the choice of time step often depends on the discrete behaviour of interest as well as the observation frequency [Breed et al., 2012]. We propose a statistic to help the user choose a time step based on producing a roughly similar number of interpolated locations and data points as the original tracking data

set. This could be combined with the multi-scale model of [Leos-Barajas et al., 2017] to study the discrete behaviour of interest. We have additionally proposed a method to deal with long periods of missing data. In some formulations of the HMM, a missing location enters the joint likelihood by including the contribution of the underlying behavioural state Markov chain ($\mathbf{A}$ in our formulation above) while removing the observation contribution for that location ($\mathbf{L}$ in our formulation) [Zucchini et al., 2016]. We instead decided to split the track into multiple groups for compartmentalized model fitting, and offered metrics for choosing how to perform this partition. Frequent long periods of missing data can be common in marine environments.

The CarHMM draws a new link between HMMs and the DCRWS model of [Jonsen et al., 2005]. Within the marine context, the two most commonly sought-after behavioural states are foraging and transiting. These states are typically assumed to follow an area-restricted search pattern, whereby foraging patches are characterized by shorter step lengths occurring in diffuse directions, and are interspersed with periods of directed travel consisting of longer step lengths directed straight ahead (see e.g. [Whoriskey et al., 2017]). While these states can be directly inferred from the state-dependent distributions of the HMM, the interpretation of these state estimates resulting from the DCRWS is less straightforward. Within the DCRWS, the main parameter influencing the step lengths is an autocorrelation term (γ). Usually (again see e.g. [Whoriskey et al., 2017]), high γ values are interpreted as highly persistent movement (indicative of transiting) and low γ values constitute highly random movement (representing foraging). As a result, transiting and foraging are not necessarily delineated by longer and shorter step lengths. The CarHMM combines the two approaches such that we now have a clear interpretation of the step lengths but can still account for the fact that some animals will tend to move in a similar (or dissimilar) manner across time. These properties make the CarHMM a useful model for linking movement data to behavioural characteristics.

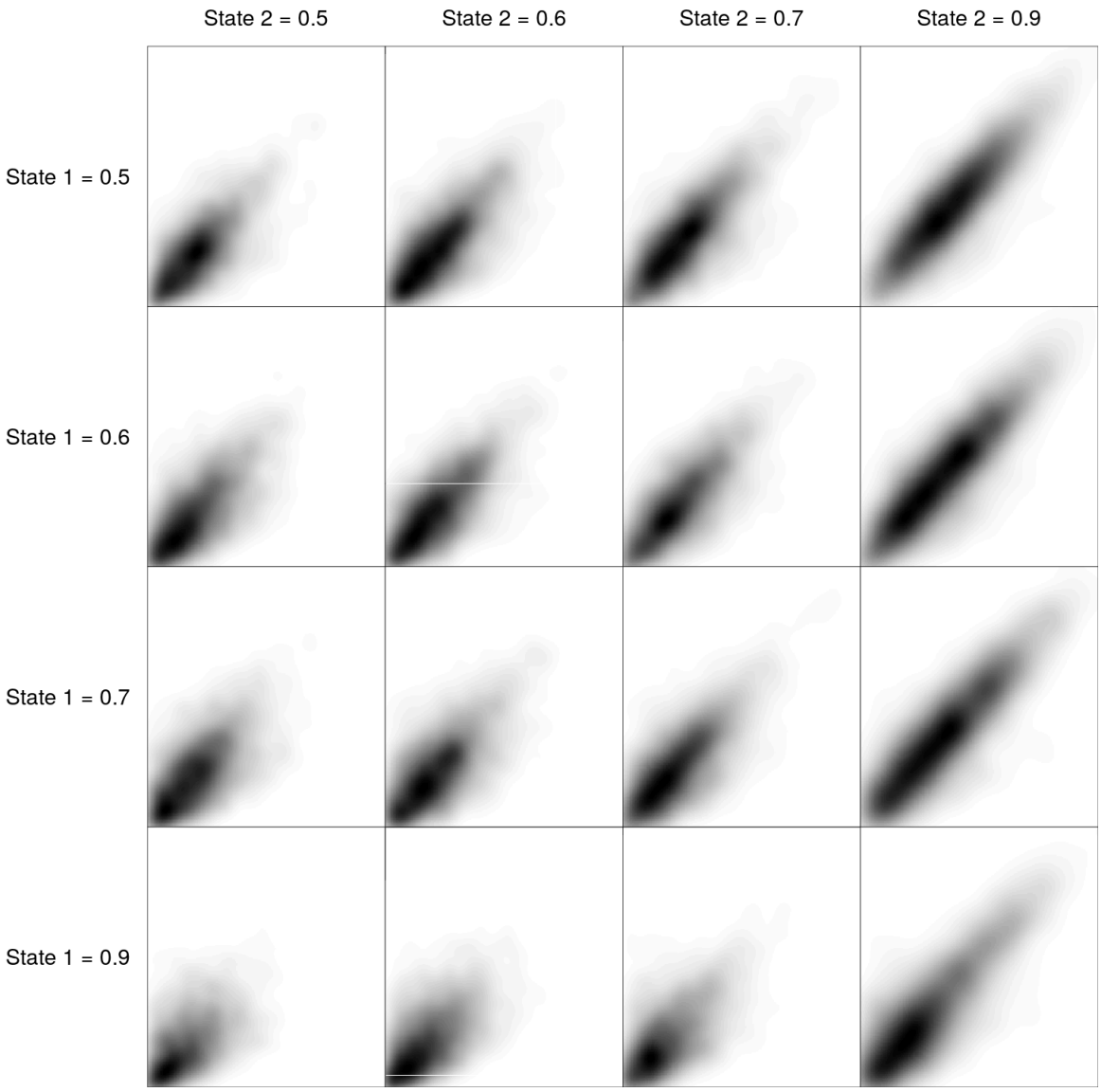

Figure 2.2: The row and column headings give the probability of staying within the given state. State 1 has low autocorrelation, state 2 has high autocorrelation. The density becomes more tightly spread along the line $y = x$ as the data remains in state 2 longer, as well as becoming less concentrated at any particular mean value.

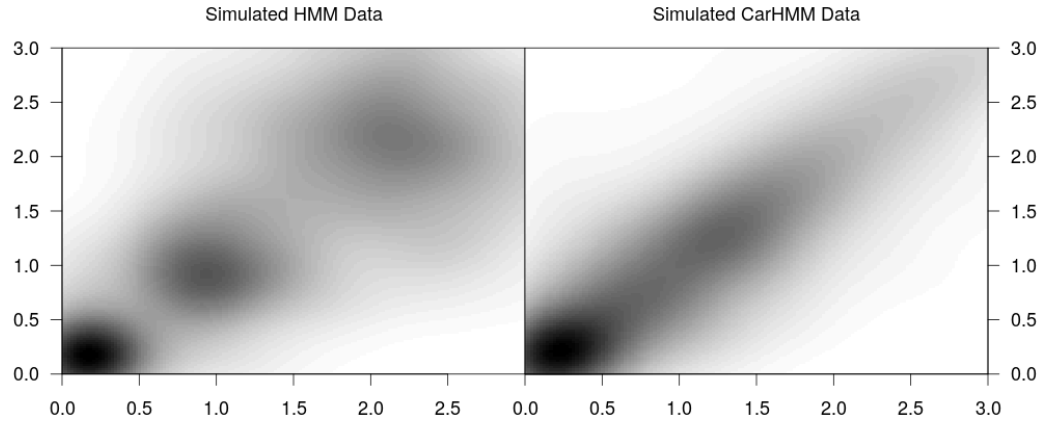

Figure 2.3: Lagplots for simulated HMM and CarHMM data. The three states of the HMM data are clearly shown by the three droplet patterns caused by the lack of within-state autocorrelation. The CarHMM does not clearly show the number of states, but shows the characteristic smeared line of the within-state autocorrelation. One can compare these plots to lag-plots of real data to help determine an appropriate model for the data.

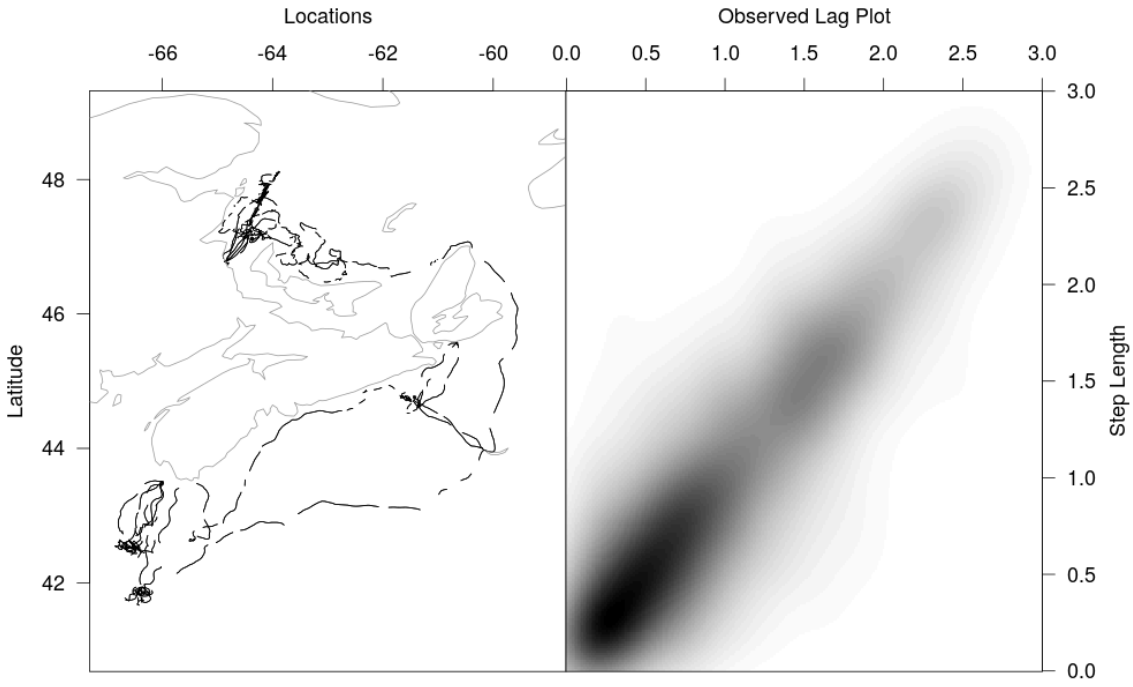

Figure 2.4: Map and lag plot of the grey seal track used in the best practice case study. Grey seals are large marine predators found in the North Atlantic ocean that are commonly observed travelling hundreds of kilometres to forage. This grey seal came from the Sable Island colony of Eastern Canada.

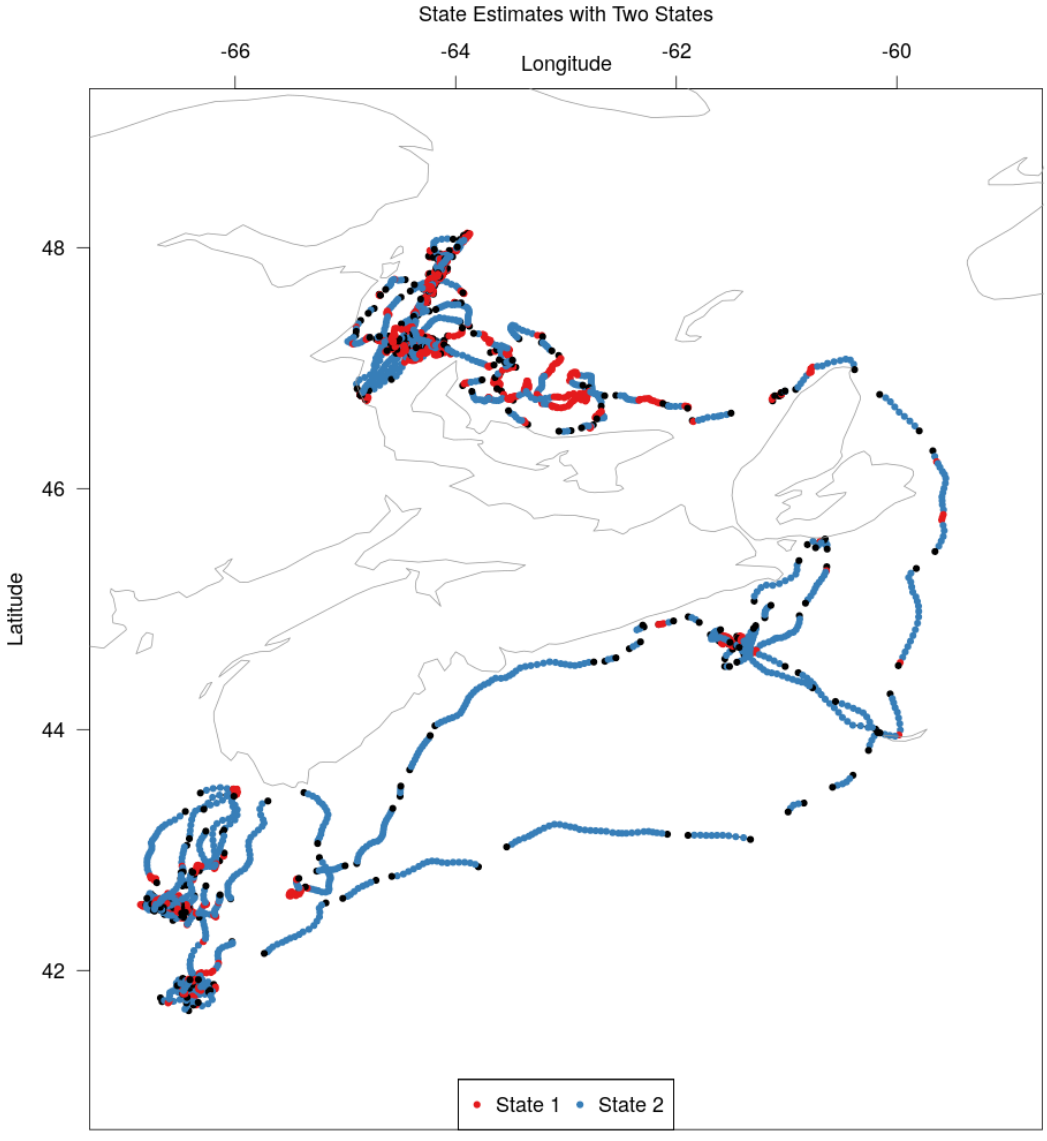

Figure 2.5: Map of the male grey seal track with state estimates from the two state CarHMM.

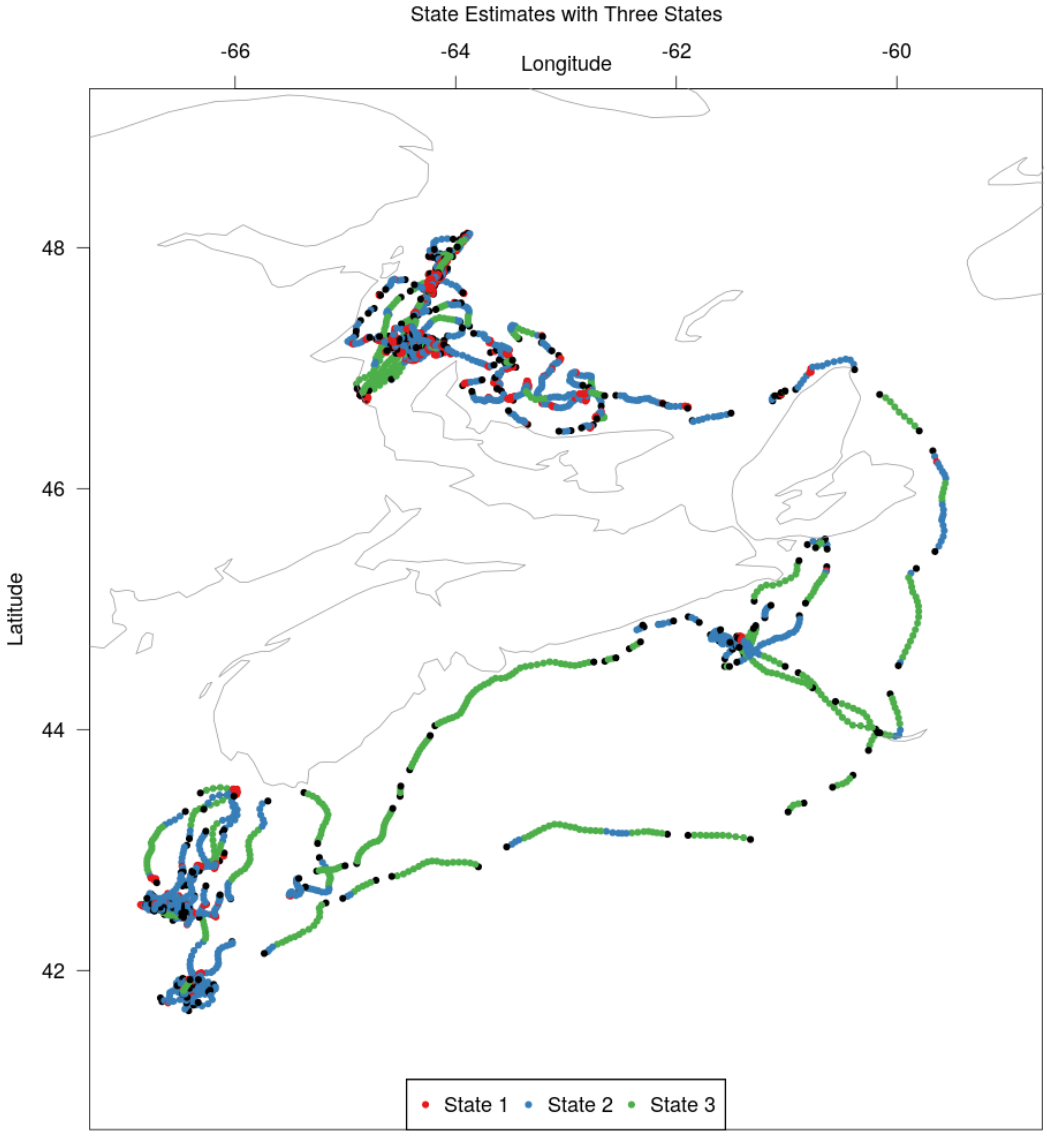

Figure 2.6: Map of the male grey seal track with state estimates from the three state CarHMM.

Chapter 3

Species Distribution Modelling with Spatio-Temporal Nearest-Neighbour Gaussian Processes

3.1 Abstract

1. Spatio-temporal datasets that are difficult to analyze are common in ecological surveys. There are software packages available to analyze these datasets, but many of them require advanced coding skills. There is a growing need for easy to use packages that researchers can use to analyze common ecological datasets.

2. We develop a particular generalized linear mixed model for spatio-temporal point-referenced data that is flexible enough to accommodate data from most ecological surveys while being structured enough to facilitate analyses without advanced coding. Our implementation in the **staRVe** package uses a computationally efficient version of a nearest neighbour Gaussian process enabling analysis of relatively large datasets.

3. A brief simulation study shows our model produces accurate predictions and forecasts, while a tutorial analysis of a Carolina wren survey suggests a recommended workflow for analyses. We also analyze a more complicated scientific survey of haddock to showcase the capabilities of our model.

4. Our model and package are tools that can easily be added to researchers' workflow to help make sense of data from ecological surveys. We emphasize the ability of our model to create useful visualisations of data which can then lead to identification of important trends in species distributions.

3.2 Introduction

Spatio-temporal datasets that are difficult to analyze are common in ecological surveys. Such surveys typically record either the count, weight, or simply presence or absence of a species observed within a small spatial transect. To track the health of the population through time surveys are typically repeated at the same time every year. This leads to data that represent points in continuous space and discrete time. In some surveys the same spatial transects are sampled every year, although others such as marine surveys use a stratified random sampling design which leads to different transects being sampled every year. Repeated transect designs are much easier to analyze and associated spatially explicit statistical models have a relatively long history in the literature. The stratified random sampling designs have traditionally been modelled on aggregate by discarding the spatial information, with spatially explicit models appearing only recently. Statistical models for both types of sampling design, however, rely in some form on Gaussian processes.

Gaussian processes underlie nearly all statistical modelling of spatial data so we briefly review the recent literature and relevant software. A Gaussian process is a (possibly infinite) collection of random variables where any finite subset has a multivariate normal distribution, and can be fully specified by a mean function and covariance function. Fitting models with a Gaussian process is computationally challenging, and recent applied interest in analyzing large spatial or spatio-temporal datasets has led to theoretical innovations making these models computationally tractable. Many of these innovations introduce different ways to approximate a Gaussian process – either by approximating the process itself or by imposing a low-rank condition on the covariance matrix. Examples of methods that approximate the process are the stochastic partial differential equation (SPDE) approach [Lindgren et al., 2011] and the nearest-neighbour Gaussian process [Datta et al., 2016]. Low-rank methods include fixed-ranked kriging [Cressie and Johannesson, 2008] and the predictive process [Banerjee et al., 2008]. An enjoyable case-study competition between a wide variety of these methods is described in [Heaton et al., 2018].

Some of the aforementioned examples have existing and well-developed `R` packages. However due to their flexibility they often require expert knowledge of `R` to correctly

code the desired model. Perhaps the most widely used package for spatial ecology is `R-INLA` [Lindgren and Rue, 2015], which implements the SPDE approach mentioned earlier. Another popular package in spatial ecology is Template Model Builder, or `TMB` [Kristensen et al., 2016]. While `TMB` is not connected with any particular Gaussian process approximation, it takes advantage of the Laplace approximation and automatic differentiation for fast optimization of a user-coded likelihood function, and was developed with spatial applications in mind.

The `R-INLA` and `TMB` packages are quite flexible, but require the user to spend a significant amount of time learning how to code the model or likelihood function. What is currently missing is a suite of more specialized `R` packages that are easy to use in specific domains. A few of these packages already exist, such as `LatticeKrig` [Nychka et al., 2016] for purely spatial data or `SpatioTemporal` [Lindström et al., 2013] for fitting Gaussian spatio-temporal processes using basis functions. For more case-specific packages that support spatial analyses we have `Hmsc` [Tikhonov et al., 2020] for multi-species community data and the `VAST` package [Thorson and Barnett, 2017] which is popular in fisheries research and supports spatio-temporal data.

To the above literature we add a new dynamic spatio-temporal model with its own flavour of modelling. Our model uses a new spatio-temporal version of the nearest neighbour Gaussian process, and is implemented using the `TMB` package to provide computationally fast inference, simulations, and predictions. This framework is specific enough to admit a user-friendly interface with little advanced programming knowledge required. At the same time many applied examples fit the continuous-space discrete-time data description giving the package wide applicability. Examples include daily ozone levels, yearly cancer incidence levels, and the ecological examples we explore in this paper. Our initial motivating domain in fisheries research also gives our package its acronym: **S**patio-**T**emporal **A**nalysis of **R**esarch **VE**ssel data, or `staRVe`.

3.3 Methods and Materials

3.3.1 Modelling Framework

We assume that we have univariate spatio-temporally referenced point data in continuous space $\mathbf{s} \in \mathbb{R}^d$ (or a connected subset of $\mathbb{R}^d$) and discrete time $t \in \mathbb{N}$.

Our model can be written in the familiar generalized linear mixed model framework:

$$Y_i \mid \mu_i \sim f_\theta\left(y_i; \mu_i\right) \tag{3.1}$$

$$\mu_i \mid \mathbf{w} = g^{-1}\left(\mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{w}\right) \tag{3.2}$$

where Y_i is the ith observation of the response variable, $f_\theta\left(y_i; \mu_i\right)$ is a response distribution depending on parameters θ with mean μ_i, g^{-1} is an inverse link function, $\mathbf{X}_i$ is the ith row of a design matrix for fixed effects/covariates with coefficients $\boldsymbol{\beta}$, and $\mathbf{Z}_i$ is the ith row of a design matrix for random effects given by $\mathbf{w}$. To remove identifiability issues between an intercept term in $\mathbf{X}$ and the mean of the random effects $\mathbf{w}$, we exclude the intercept term in $\mathbf{X}$ [Ogle and Barber, 2020]. For notational simplicity, we replace the design matrix $\mathbf{Z}$ with indices specifying the time point and location, so that an observation $Y_{i,t}\left(\mathbf{s}_i\right)$ at time point t and location $\mathbf{s}_i$ is dependent on the random effect $w_t\left(\mathbf{s}_i\right)$. With this change we write the spatio-temporally referenced model as

$$Y_{i,t}\left(\mathbf{s}_i\right) \mid \mu_{i,t}\left(\mathbf{s}_i\right) \sim f_\theta\left(y_{i,t}\left(\mathbf{s}_i\right);\ \mu_{i,t}\left(\mathbf{s}_i\right)\right) \tag{3.3}$$

$$\mu_{i,t}\left(\mathbf{s}_i\right) \mid w_t\left(\mathbf{s}\right) = g^{-1}\left(\mathbf{X}_i\boldsymbol{\beta} + w_t\left(\mathbf{s}_i\right)\right) \tag{3.4}$$

We note that $\mathbf{X}_i$ may be spatially and/or temporal referenced, however we can still associate such a covariate with an observation $Y_{i,t}\left(\mathbf{s}\right)$ so we leave that possibility implicit in the notation. It remains to specify the distribution for the random effects $W_t\left(\mathbf{s}\right)$.

The spatio-temporal random effects

We model our spatio-temporal random effects as a time series of spatial fields, or a spatial field that evolves over time, using a two-level hierarchical specification. The

'temporal' level in the hierarchy models the global temporal evolution of the random effects. The 'spatial' level in the hierarchy models the spatial variation in the random effects, or more precisely, it models the spatial variation in how each location evolves from one time to the next.

The 'temporal' level is specified as a simple autoregressive order 1, or AR(1), process. Specifying a set of temporal random effects $\boldsymbol{\epsilon}$, with a random effect ϵ_t for each time point in the model, we have

$$\boldsymbol{\epsilon} \sim \mathrm{N}\left(\mu,\ \frac{\sigma^2}{1-\phi^2}\boldsymbol{\Sigma}\right) \quad \text{where} \quad \Sigma_{ij} = \phi^{||i-j||}. \tag{3.5}$$

where μ is the mean of the process, σ^2 is the one-step-ahead temporal variance $\mathrm{Var}\left[\epsilon_t\right]$, and ϕ is the correlation between adjacent times $\mathrm{Cor}\left[\epsilon_{t+1},\ \epsilon_t\right]$.

Moving to the 'spatial' level of the hierarchy, the Gaussian process (GP) is a ubiquitous choice for any spatial process due to its attractive stochastic properties. A Gaussian process is a generalization of a multivariate normal random variable to infinite dimension, where any finite subset has a multivariate normal distribution. However, the computation cost in evaluating the likelihood for a Gaussian process is prohibitive even for modestly sized datasets. We use a nearest-neighbour Gaussian process (NNGP) as a computationally efficient alternative [Datta et al., 2016]. A GP is described completely by a mean function $m\left(\mathbf{s}\right) : \mathbb{R}^d \to \mathbb{R}$ that gives the mean at each location $\mathbf{s}$, and a covariance function $\mathcal{C}\left(\mathbf{s}_1, \mathbf{s}_2\right) : \left(\mathbb{R}^d, \mathbb{R}^d\right) \to \mathbb{R}^+$. The covariance function encodes Tobler's first law of geography that "everything is related to everything else, but near things are more related than distant things" [Tobler, 1970]. Additionally the NNGP requires a directed acyclic graph to be specified on a set of random effect nodes, acting much like the mesh used in the `R-INLA` package.

We have a different spatial process at each time point, so we index these random effects by both time and space $W_t\left(\mathbf{s}\right)$ and model them through their conditional distributions:

$$W_t\left(\mathbf{s}\right) \mid w_{t-1}\left(\mathbf{s}\right),\ \boldsymbol{\epsilon} \sim \mathrm{NNGP}\left(m_t\left(\mathbf{s}\right),\ \mathcal{C}\left(\mathbf{s}_1, \mathbf{s}_2\right)\right). \tag{3.6}$$

The 'temporal' level in our hierarchy is used to inform the mean function at each time

point such that

$$m_t(\mathbf{s}) = \phi \cdot (w_{t-1}(\mathbf{s}) - \epsilon_{t-1}) + \epsilon_t, \tag{3.7}$$

ensuring that the temporal evolution at any specific location follows an AR(1) process. The time series at each location are spatially correlated, the strength of which is described by the covariance function.

The Matérn covariance function is the most widely used covariance function in the literature, containing many special cases such as the exponential. Here we present only the exponential covariance function for simplicity.

$$\mathcal{C}(\mathbf{s}_1, \mathbf{s}_2) = \tau^2 \rho \, \text{Exp}\left[-\frac{||\mathbf{s}_1 - \mathbf{s}_2||}{\bar{d}} \cdot \frac{1}{\rho}\right]. \tag{3.8}$$

We estimate the spatial standard deviation parameter τ and spatial range parameter ρ. The average distance $\bar{d}$ between 'connected' random effect nodes in the directed acyclic graph is included to standardise parameter estimates for datasets with different spatial scales (technically, it is the spatial scale of the random effect nodes that is standardised).

The spatial range parameter ρ is nominally the distance at which locations become approximately spatially uncorrelated, that is ρ is defined to be the distance between two locations such that $\mathcal{C}(||\mathbf{s}_1 - \mathbf{s}_2|| = \rho) \approx 0.05$. The range parameter is weakly estimable but strictly speaking it is unidentifiable with a mostly flat profile likelihood, and does not affect inference of other model parameters nor prediction of the random effects [Zhang, 2004]. Since estimating the parameter is fairly difficult a common approach is to fix the value of the ρ to be some large value, which can in some cases provide predictions that are nearly as good as they would be if the true value of ρ were known [Kaufman and Shaby, 2013]. When possible though, we still recommend trying to estimate ρ via maximum likelihood.

We give a more complete description of our model in Appendix A.

The log-likelihood function

Writing the complete model, we have

$$Y_{i,t}(\mathbf{s}_i) \mid \mu_{i,t}(\mathbf{s}_i) \sim f_\theta\left(y_{i,t}(\mathbf{s}_i);\ \mu_{i,t}(\mathbf{s}_i)\right) \tag{3.9}$$

$$\mu_{i,t}(\mathbf{s}_i) \mid w_t(\mathbf{s}) = g^{-1}\left(\mathbf{X}_i\boldsymbol{\beta} + w_t(\mathbf{s}_i)\right) \tag{3.10}$$

$$W_t(\mathbf{s}) \;\Big|\; \{w_{t-1}(\mathbf{s}),\ \boldsymbol{\epsilon}\} \sim \mathrm{NNGP}\left(\phi \cdot (w_{t-1}(\mathbf{s}) - \epsilon_{t-1}) + \epsilon_t,\ \mathcal{C}(\mathbf{s}_1, \mathbf{s}_2)\right) \tag{3.11}$$

$$\boldsymbol{\epsilon} \sim \mathrm{N}\left(\mu,\ \frac{\sigma^2}{1-\phi^2}\boldsymbol{\Sigma}\right) \quad \text{where} \quad \Sigma_{ij} = \phi^{||i-j||}. \tag{3.12}$$

The mean for the initial spatial field is spatially constant and equal to ϵ_1, and the covariance function for the initial spatial field is identical but scaled by a factor of $1/(1-\phi^2)$.

We proceed with maximum likelihood inference and write the joint log-likelihood function as $\ell_J(\Theta) = \ell_y(\Theta) + \ell_w(\Theta)$ where $\ell_y(\Theta)$ is the likelihood contribution from the response distribution and $\ell_w(\Theta)$ is the likelihood contribution from the random effects in both the 'temporal' and 'spatial' levels of the hierarchy. We integrate out the random effects and use the restricted or marginal likelihood function

$$L(\Theta) = \int \mathrm{Exp}\left[\ell_J(\Theta)\right] d\mathbf{W} = \int \mathrm{Exp}\left[\ell_y(\Theta) + \ell_w(\Theta)\right] d\mathbf{W}. \tag{3.13}$$

The above integral is computed using the Laplace approximation as implemented in the `R` package `TMB` [Kristensen et al., 2016]. We obtain maximum likelihood estimates $\hat{\Theta}$ (with standard errors) for the parameters $\Theta = \{\theta,\ \boldsymbol{\beta},\ \mu,\ \phi,\ \sigma^2,\ \tau,\ \rho\}$. Predictions (with standard errors) for the random effects $W_t(\mathbf{s})$ and $\boldsymbol{\epsilon}$ are then obtained by maximizing the joint likelihood $\ell_J\left(\hat{\Theta}\right)$ with respect to the random effects.

Predictions

A common task in spatial and spatio-temporal statistics is to predict the random effects and response mean at unobserved locations or forecasting into future times. We obtain predictions in three steps: first the random effects, then the linear predictor including the covariates, and finally the response mean after applying the link function. These predictions are easily obtained from our model coupled with the functionality in the `TMB`.

To our joint likelihood $\ell_J(\Theta)$ we add another contribution $\ell_p(\Theta)$, the conditional distribution of the random effects at the prediction locations given the random effects in $\ell_J(\Theta)$. These conditional distributions share the same mean and covariance function as the nearest-neighbour Gaussian process, so in essence we are adding locations to the likelihood of the spatio-temporal nearest-neighbour Gaussian process used in ℓ_w. The predicted value for $W_t(\mathbf{s})$ is then taken to be the mode of $\ell_J\left(\hat{\Theta}\right) + \ell_p\left(\hat{\Theta}\right)$ as a function of $w_t(\mathbf{s})$, and the standard errors are computed from the hessian evaluated at the mode. This is essentially a plug-in estimate using the maximum likelihood estimates $\hat{\Theta}$, but since we take the predictions directly from the likelihood function all uncertainty in parameter estimation is propagated to the predictions of the random effects [Zheng and Cadigan, 2021].

With a new set of covariates $\mathbf{X}$, regression coefficient estimates, and predicted random effects we obtain the linear predictor

$$\mathbf{X}\hat{\boldsymbol{\beta}} + \hat{\mathbf{w}}_t(\mathbf{s}), \tag{3.14}$$

Finally, the predicted response means are found by applying the inverse link function $g^{-1}\left[\mathbf{X}\hat{\boldsymbol{\beta}} + \hat{\mathbf{w}}_t(\mathbf{s})\right]$.

3.3.2 Simulations

To develop an understanding of our model framework we visualise data simulated from different scenarios within it. By visualising these simulations we help develop an intuition for parameter interpretation and the performance of the package. Our scenarios are

1. spatial simulations with no observation error
2. spatio-temporal simulations with Gaussian response
3. spatio-temporal simulations with Poisson response

We take the simulation design used by [Zhang, 2004] and adapt it to visualise our model in the spatio-temporal setting. We simulate random effects on a grid covering the unit square, with points at $(0.1 * x, 0.1 * y)$ for $x, y \in 0,\ 1,\ ...,\ 10$ and $(0.1 * x +$

$0.05, 0.1 * y + 0.05)$ for $x, y \in 0,\ 1,\ ...,\ 9$ for a total of 221 locations. We also simulate random effects on a transect covering $(0.387, 0.01 * y)$ for $y \in 0,\ ...,\ 100$. This study design is shown in Figure For our two spatio-temporal scenarios we simulate data for ten years. We then apply the inverse link function to the simulated random effects to get the expected value for each data point, then simulated data from the corresponding response distribution. The data for scenario 1 are taken to be the simulated random effects, which corresponds to a scenario of no observation error.

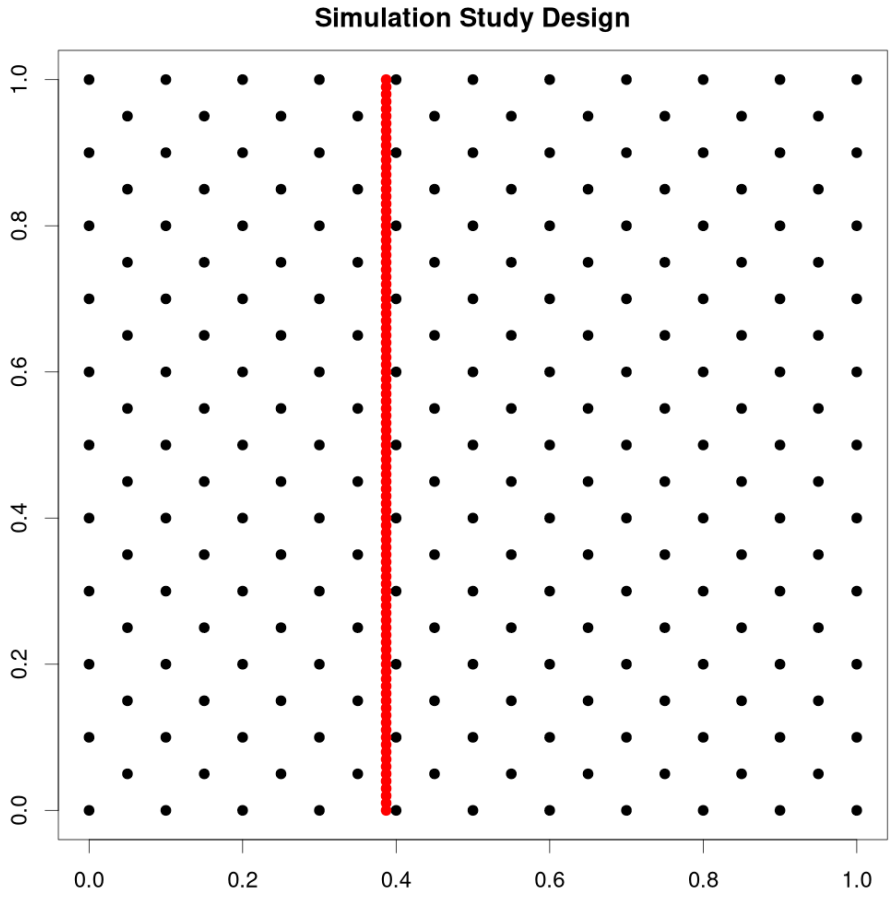

Figure 3.1: Location design for the simulation studies. Locations in black constitute data locations. Locations in red are not included as data when fitting the model, and predictions within each year are made for these locations.

We then fit a model to the simulated data on the grid, and use the fitted model to produce predictions for the transect. In the case of no observation error, such as our purely spatial example, our model gives an exact interpolation procedure, thus the predictions are guaranteed to be equal to the true values at locations where data are

observed. The transect we are giving predictions for is within the spatial extent of the data but does not coincide with any of the data locations, so within the transect the guarantee given by exact interpolation does not apply. In the two spatio-temporal scenarios we fit the model using data from years one through eight and then use the fitted model to forecast predictions into years nine and ten. For all scenarios we simulate with the spatial range $\rho = 0.3$ and estimate it via maximum likelihood.

Spatial simulations with no observation error

We simulate a realization of the Gaussian random effects, without observation error, and for a single year. Since μ only shifts the data up or down we let $\mu = 0$. The temporal parameters ϕ and σ do not enter the model since there is only a single year of observations. The only remaining parameter in this scenario is the spatial standard deviation τ. We simulate the random effects using the value $\tau = 1$. To see the effect of τ on the model we fix the estimated value of τ at three pre-selected values $\tau = 0.5$, 1, and 2. We then use the fitted model to produce predictions for the transect and compare the point predictions and standard errors to see the effects of τ.

Spatio-temporal simulations with Gaussian response

We simulate a realization of a dataset using a Gaussian response distribution with eight years of data to fit the model, and an additional two years to forecast predictions. We set the temporal autocorrelation parameter $\phi = 0.5$ and response standard deviation $\sigma_{\text{obs}} = 2$. The mean μ is still equal to zero. We simulate data under four different spatio-temporal characteristics, the combinations of low or high spatial variance ($\tau = 1$ and $\tau = 3$) and low or high temporal variance ($\sigma = 5$ and $\sigma = 15$). Unlike in the previous section we fit all models using maximum likelihood, and plot the predicted mean and 95% confidence band only for ML parameter estimates. We predict and plot the random effects instead of the data to remove the observation noise when plotting the simulated data.

Spatio-temporal simulations with Poisson response

We replace the Gaussian response distribution of the previous simulation settings with

a Poisson distribution. When plotting we convert the random effects to the response scale so the simulations and predictions shown represent the spatio-temporally varying rate parameter of a Poisson distribution. We set the autocorrelation parameter $\phi = 0.5$ and spatial standard deviation $\sigma = 0.5$. We then simulate data under four different combinations: low or high spatial variance ($\tau = 0.05$ and $\tau = 0.2$) and low or high global mean ($\mu = \log 8$ and $\mu = \log 15$).

Interpreting the plots

We plot only the transect instead of the whole domain, which allows us to include multiple years of the same transect in a single plot. The transect is labeled from south (S) to north (N) corresponding to the points $(0.387, 0)$ and $(0.387, 1)$ respectively. When plotting multiple years of the same transect, we paste the transect corresponding to year t at the end of the transect for year $t-1$. The years are labelled and separated into vertical bands by a dashed line. The spatial variability is the amount of variation the plot exhibits within each vertical band (i.e., the south-to-north transect within a particular year). The temporal variability is the amount of variation the plot exhibits from one vertical band to the next (i.e., the evolution of the transect from one year to the next).

3.3.3 Carolina Wren Survey

We analyze a dataset containing the number of sightings of Carolina wren *Thryothorus ludovicianus* recorded once a year along spatial transects in Missouri, U.S.A taken from the package `STRbook` [Zammit-Mangion, 2020], which in turn was modified from [Pardieck et al., 2017]. This analysis is meant to demonstrate the oft-unreported typical steps – from model fitting and checking to predictions and interpretation – and to highlight some of the surrounding `R` packages that our implementation is enhanced by.

The Carolina wren dataset is stored using the "simple features" open standard for spatial vector data as implemented in the `R` package `sf` [Pebesma, 2018]. We could not find the coordinate reference system used for this data, so for demonstrative purposes we assume it is WGS84. The staRVe package does not require projected

geographic data so we leave the data unprojected. The dataset has 783 observations across 21 years, ranging from 1994 to 2014. The same locations are surveyed every year, however there are some missing observations which have been removed from the dataset. Without these missing data, there is a median 36 observations per year, with as few as 26 and as many as 45.

Our initial model for this dataset uses a Poisson distribution with the canonical log link function. The convergence message is “relative convergence", so we are confident that we have found a local maximum of the log-likelihood function. With ecological count data it is usually a good idea to check for over- or under-dispersion of the data relative to the fitted model. While over- or under-dispersion can be caused by any number of model mis-specifications, the easiest one to check is for mis-specification of the response distribution. We use quantile residuals provided by the `DHARMa` package, which uses simulations to construct a parametric bootstrap estimator of the cumulative distribution function (CDF) for each data point [Hartig, 2020]. If the model and response distribution are correct then the residuals from the `DHARMa` package will be uniformly distributed on the interval [0,1].

For our analysis we simulate 100 set of observations from the fitted model to construct the bootstrap CDF. We then simulate one set of observations each from a model with a Poisson response distribution, an over-dispersed negative binomial response distribution, and an under-dispersed Conway-Maxwell-Poisson response distribution. The Conway-Maxwell-Poisson distribution can be seen as a generalized version of the Poisson distribution that can exhibit both under- and over-dispersion and has been used in a variety of applications [Sellers et al., 2012]. To ensure we are checking only the response distribution we simulate new observations conditional on the fitted random effects and parameters, so that all of the simulated datasets share the same spatio-temporal pattern as the Carolina wren dataset. Quantile residuals for the Carolina wren dataset and each of the different response distributions are computed relative to the Poisson model, i.e. using the bootstrap CDF from the fitted model. Through this procedure we have four sets of residuals: one set where we know the model is correct (Poisson simulation), one set where we know the data is over-dispersed relative to the model (negative binomial simulation), one set where we know the data

is under-dispersed relative to the model (Conway-Maxwell-Poisson simulation), and the final set of residuals coming from the real data set. We then compare the residuals from the Carolina wren dataset to the residuals coming from the simulated datasets to determine whether the data exhibit under- or over-dispersion relative to the fitted model using the Poisson distribution.

After checking the fitted model we inspect the parameter estimates to get a general sense of spatio-temporal behaviour of the random effects and which covariates, if included, have significant effects. With those general impressions in mind we then create a spatio-temporal map of the response mean as an important visualisation of the data and fitted model. We produce a smooth map by predicting the response mean on a raster using the procedure detailed in Section 3.3.1, and using the raster data format supplied by the `raster` package [Hijmans, 2020]. The final graphic is produced using the raster predictions as a data layer in the `tmap` package [Tennekes, 2018], though the raster could also be exported to an external GIS program.

3.3.4 Haddock Survey

To showcase the flexibility of our model we analyze a fairly complex dataset from a long running scientific survey on the Scotian shelf. The survey spans the years of 1982 to 2016 with between 114 to 235 observations each year, except for one year with 30 observation, for a total of 6,259 observations across 35 years. Each observation represents a scientific survey vessel fishing set along a spatial transect, from which we take the total biomass of haddock that is caught along that transect. The transect locations are selected each year by a stratified random sampling scheme so the locations are not repeated from one year to the next. We fit a tweedie model to the data then use the fitted model to forecast a predicted map of relative density for 2017. We also include maps of predicted density for the years 2014-2016 to provide the recent context of the fishery. These maps could then be used to help determine a management strategy for the haddock fishery.

Without additional external information our scientific survey data is only able to produce prediction of relative density, and not absolute density. That is, we can produce predictions whose spatial pattern should match the spatial pattern of abundance but

is only proportional to the true abundance. To reliably produce predictions of absolute density an estimate of a catchability parameter is typically needed, which requires either a separate experiment or additional data such as total amount of commercial catch with additional modelling.

The tweedie response distribution provides a continuous probability density function for non-zero observations and a point mass at zero to account for the zero-inflation typical of fisheries data [Foster and Bravington, 2013]. The tweedie distribution is a member of the exponential family of distributions whose mean/variance relationship is governed by the power law $\text{Var}[Y] = \sigma^2 \, \text{E}[Y]^p$ where σ^2 is the dispersion parameter and p is the power parameter. We include the area swept by each fishing set as an offset term for the mean to account for differences in fishing effort between different observations.

3.4 Results

3.4.1 Simulations

Our brief simulation study shows that our model provides useful predictions with generally good point estimates and close to nominal 95% confidence intervals. As would be expected, forecast predictions are much more inaccurate than within-sample years but the confidence intervals are still close to nominal. We interpret the plots of the simulations below to give an idea how the model parameters affect the model behaviour.

Spatial simulations with no observation error

	True	Scenario 1		Scenario 2		Scenario 3	
spatial std. dev. τ	1.00	0.500	(0.000)	1.000	(0.000)	2.000	(0.000)
spatial range ρ	0.3	5.467	(13.529)	0.621	(0.425)	0.094	(0.014)
global mean μ	0.0	-0.167	(0.174)	-0.207	(0.334)	-0.613	(0.260)

Table 3.1: Simulation parameters and estimated values (with standard errors) for the spatial simulation scenario.

Simulations and model predictions are shown in figure 3.2. The upper panel shows the predicted mean for the random effects with a 95% confidence band around the

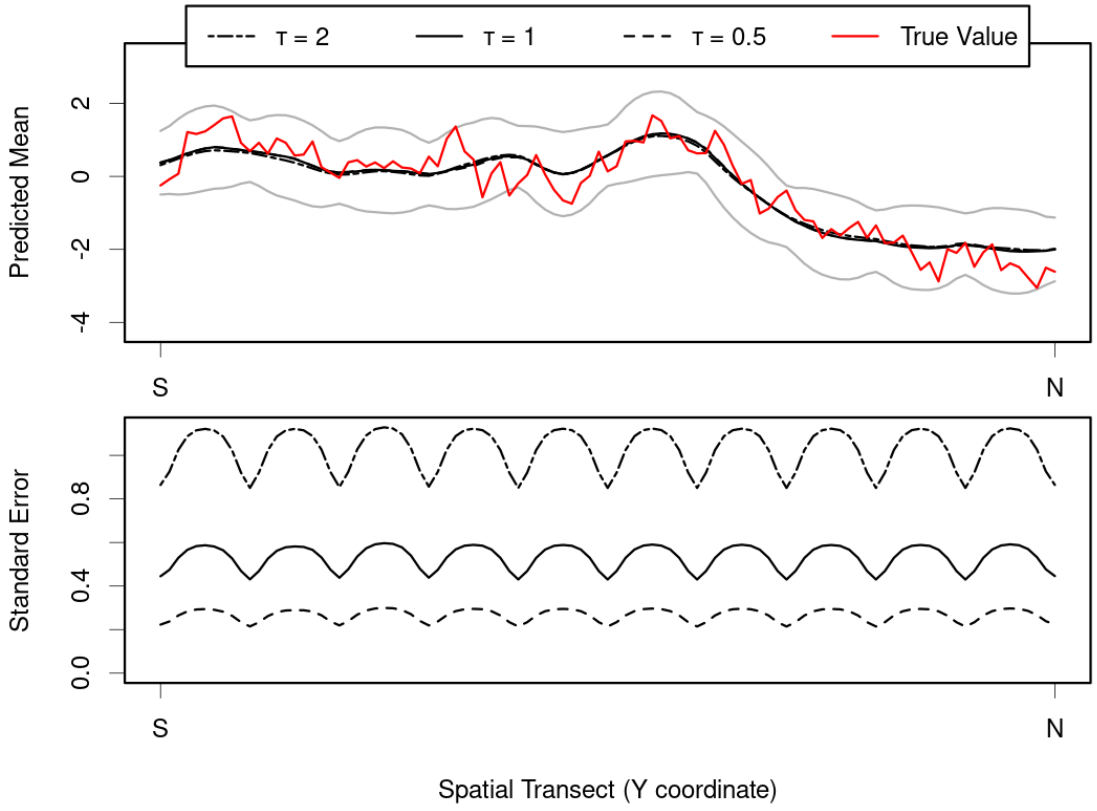

Figure 3.2: Simulated random effects (red) along the transect $(0.387, y)$ for $y \in [0, 1]$. The spatial variance used to simulate the random effects is $\tau = 1$. Predictions (top, black) and standard errors (bottom) for the random effects are obtained using different values of τ. A 95% confidence band (grey) is shown around the predicted mean when predicting under the same value of τ used to simulate the random effects.

predictions made using $\tau = 1$, and the bottom panel shows the prediction standard error. The mean predictions are nearly identical when using $\tau = 0.5$, $\tau = 1$, or $\tau = 2$. The bottom panel shows that the scale of the standard errors is directly proportional to the spatial standard deviation parameter. For example the 95% confidence band under $\tau = 2$ is twice as wide as the 95% confidence band using $\tau = 1$, and four times as wide as the 95% confidence band using $\tau = 0.5$.

Spatio-temporal simulations with Gaussian response

Simulations and fitted model predictions are shown in figure 3.3. The spatial and temporal variance parameters clearly affect the right components of the model behaviour. A larger spatial variance corresponds to larger peaks and valleys within each year

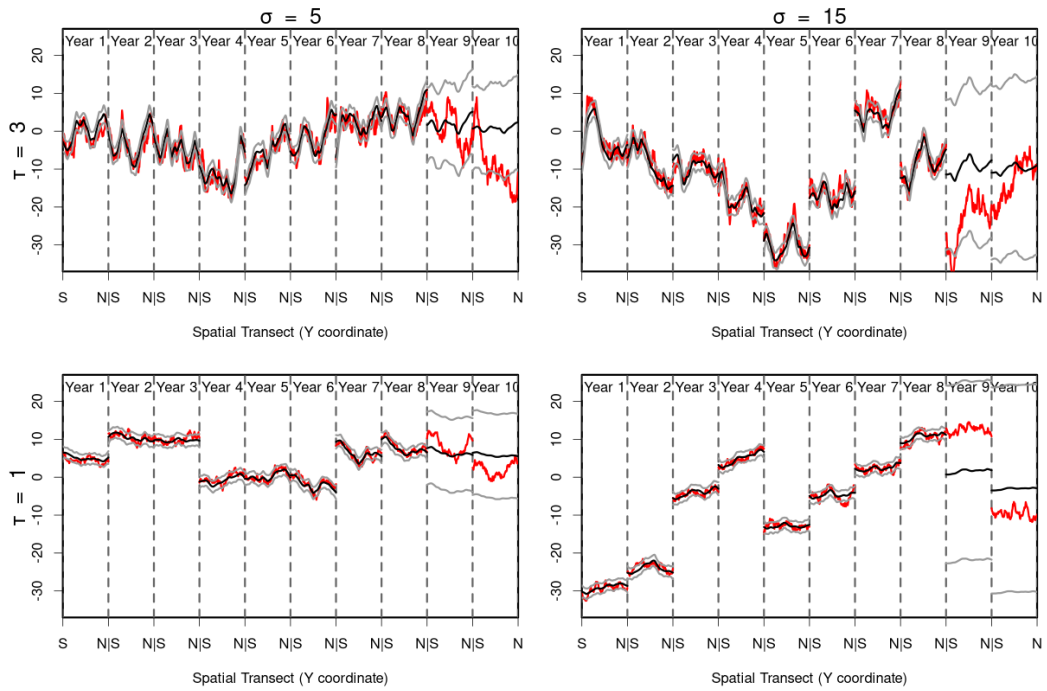

Figure 3.3: 10 years of simulated (red) and predicted (black) random effects along the transect $(0.387, y)$ for $y \in [0, 1]$ in a spatio-temporal model with a Gaussian response distribution. We have a 2 x 2 design with low and high spatial variance (bottom and top row, respectively) and low and high temporal variance (left and right column, respectively). The last two panels in each plot correspond to years 9 and 10. The fitted model did not include data from these years, so come as forecast predictions from the fitted model.

while a larger temporal variance exhibits bigger jumps in the overall mean between years.

Spatio-temporal simulations with Poisson response

Simulations and fitted model predictions are shown in figure 3.4. The spatial variance acts the same as in the Gaussian case. A larger average intensity exhibits a larger mean and increased overall variability, behaviour which is due to the log link function used.

3.4.2 Carolina Wren Survey

The first attempt Poisson model successfully converged to a local maximum of the log-likelihood function, so we proceed to model validation. The Q-Q plot of the Carolina

	True	$\tau = 3,\ \sigma = 5$		$\tau = 3,\ \sigma = 15$		$\tau = 1,\ \sigma = 5$		$\tau = 1,\ \sigma = 15$	
spatial std. dev. τ	– –	3.837	(0.211)	3.729	(0.241)	1.351	(0.170)	1.195	(0.142)
spatial range ρ	0.3	0.286	(0.060)	0.265	(0.060)	0.232	(0.065)	0.345	(0.117)
ar1 corr. ϕ	0.5	0.481	(0.038)	0.498	(0.042)	0.509	(0.066)	0.509	(0.078)
ar1 std. dev. σ	– –	3.369	(0.988)	10.362	(2.656)	5.199	(1.339)	13.097	(3.320)
global mean μ	0.0	-0.262	(1.933)	-10.477	(5.705)	5.192	(2.884)	-7.852	(7.239)
response std. dev. σ_{obs}	2.0	1.894	(0.124)	2.082	(0.126)	1.987	(0.057)	2.078	(0.051)

Table 3.2: Simulation parameters and estimated values (with standard errors) for the spatio-temporal simulation with Gaussian response.

	True	$\mu = 2.08,\ \tau = 0.2$		$\mu = 2.71,\ \tau = 0.2$		$\mu = 2.08,\ \tau = 0.05$		$\mu = 2.71,\ \tau = 0.05$	
spatial std. dev. τ	– –	0.251	(0.023)	0.209	(0.014)	0.077	(0.035)	0.084	(0.019)
spatial range ρ	0.3	0.297	(0.090)	0.387	(0.112)	0.147	(0.092)	0.214	(0.091)
ar1 corr. ϕ	0.5	0.515	(0.072)	0.462	(0.066)	0.485	(0.226)	0.239	(0.190)
ar1 std. dev. σ	0.5	0.580	(0.152)	0.743	(0.190)	0.467	(0.141)	0.304	(0.077)
global mean μ	– –	1.781	(0.328)	2.569	(0.396)	1.940	(0.253)	2.915	(0.133)

Table 3.3: Simulation parameters and estimated values (with standard errors) for the spatio-temporal simulation with Poisson response.

wren residuals is shown in Figure 3.5 alongside the residuals for simulated Poisson data, over-dispersed negative binomial data, and under-dispersed Conway-Maxwell-Poisson data. The Q-Q plot of the Carolina wren residuals most closely resembles the Q-Q plot of the under-dispersed Conway-Maxwell-Poisson residuals, which suggests that the Carolina wren data exhibits under-dispersion relative to the Poisson model. We refit the data using a Conway-Maxwell-Poisson distribution which also successfully converges. The Conway-Maxwell-Poisson distribution is under-dispersed when the dispersion parameter is less than 1, over-dispersed when it is greater than 1, and becomes a (equidispersed) Poisson distribution when it is equal to 1. The dispersion parameter is estimated to be 0.7919 with a standard error of 0.101 which confirms our suspicion of under-dispersion. Estimates of all model parameters are given in Table 3.4. Maps of the predicted mean intensity and standard errors are given in Figures 3.6 and 3.7, respectively.

Interpretation

Missouri lies at the north-eastern edge of the Carolina wren's habitat range, and our analysis here seems to agree. Figure 3.6 shows two population centers that are fairly stable throughout the years studied. The first group lives in the south-eastern corner of the state, with the Mississippi River to the east and Mark Twain National Forest

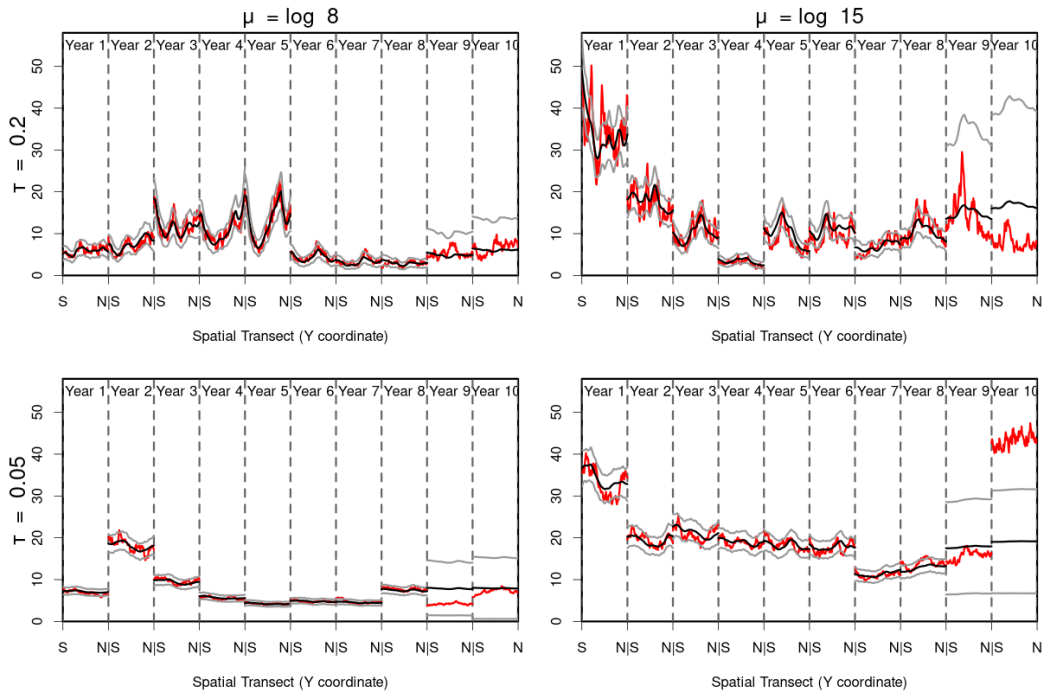

Figure 3.4: 10 years of simulated (red) and predicted (black) response mean along the transect $(0.387, y)$ for $y \in [0, 1]$ in a spatio-temporal model with a Poisson response distribution. We have a 2 x 2 design with low and high spatial variance (bottom and top row, respectively) and low and high average intensity (left and right column, respectively).

to the north-west of the population center. The second population is present in the south-west corner of the state, on the northern edge of the Ozarks. The predictions for this second population seem to be driven by observations at a single survey location with high counts. Given the observations at this particular location are fairly unique, it would be worth investigating if the wren population is actually abundant at this location or if we are observing higher counts due to a particularly skilled bird watcher. A log of the bird watcher names could identify if the same person ran the survey at that location each year, while a second independent survey could be used to try and validate the results from this dataset.

The model predictions show some interesting dynamics in the first eleven years. From 1994 to 1997 the wren population is not very abundant, but seems to move south from around the St. Louis area to the south-eastern corner of the state. Then the population flourishes, and the sub-population north of the Ozarks appears as well.

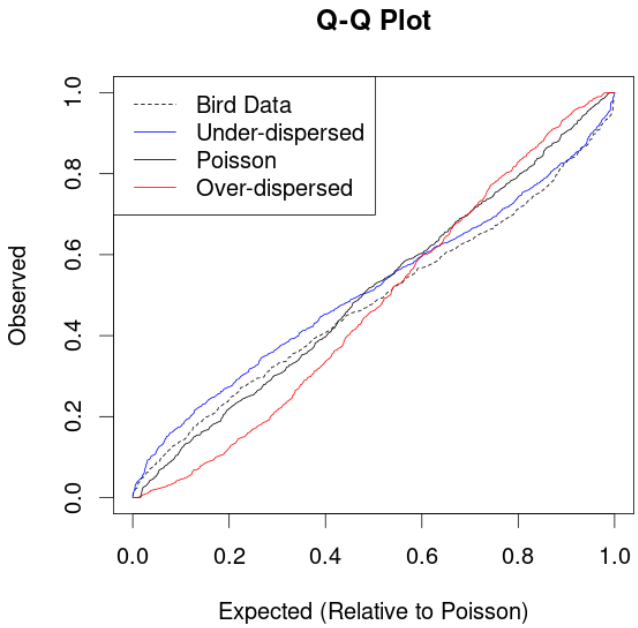

Figure 3.5: Q-Q plot comparing residual patterns for data sources with possible under- or over-dispersion. The observed dataset has residuals consistent with under-dispersion. The over-dispersed data come from a negative binomial response distribution with (over-)dispersion parameter of 1.5. The under-dispersed data come from a Conway-Maxwell-Poisson distribution with a dispersion parameter of 0.5.

But disaster strikes! Between the survey in 2000 and the one in 2001, the Carolina wren population seems to completely vanish. By 2006 the population has recovered and stays fairly stable until 2011. Within these years, the south-eastern population and the Ozark population are not completely distinct groups as they are connected by a moderately abundant band across the Mark Twain National Forest. In 2011 the population contracts spatially and wrens are seen in very low number everywhere except the south-eastern population, which is concentrated in a smaller geographic area than seen in the any of the previous years. From 2011 to 2014 the Ozark population recovers once more and the south-eastern population shifts slightly northwards by about 50 kilometres, from the Sikeston area to the Cape Girardeau area.

A few questions arise from the above observations. What caused the sudden population crash in 2001? Was the decline in 2011 due to a similar cause? What caused the subsequent population recoveries? While our fitted model may not be able to answer

	Estimate	Standard Error
(log) global mean μ	1.8659	0.5879
AR(1) correlation ϕ	0.9250	0.0221
temporal std. dev. σ	0.7535	0.1907
spatial std. dev. τ	0.4649	0.0681
spatial range ρ	72.9335	25.0282
CMP dispersion	0.7919	0.1010

Table 3.4: Parameter estimates for Conway-Maxwell-Poisson model fitted to the Carolina wren dataset

these questions, it does provide a useful visualisation of the data that prompts us to ask these questions in the first place. If we had included environmental covariates such as ground cover (forest, urban, farmland, etc.) or rainfall in the analysis, we might be able to explore the causes of the above behaviours more closely.

3.4.3 Haddock Survey

	Estimate	Standard Error
(log) global mean μ	-4.8936	0.6844
AR(1) correlation ϕ	0.9536	0.0071
temporal std. dev. σ	0.8200	0.1859
spatial std. dev. τ	1.5037	0.0651
spatial range ρ	35.0967	2.7712
tweedie dispersion	3.7144	0.1007
tweedie power	1.5750	0.0065

Table 3.5: Parameter estimates for tweedie model fitted to the haddock survey

Estimates of the model parameters are given in Table 3.5. Maps of the predicted density and standard errors in the years 2014 to 2017 are given in Figures 3.8 and 3.9, respectively. First we look at the predictions in the non-forecast years, 2014 to 2017. The maps show three discrete populations. The most prominent is a large population centred just south of the southwestern tip of Nova Scotia that spreads north along the coastline and into the Bay of Fundy in the later years. There seems to be a large population at the very southeastern edge of the study area in 2014, but since it is on the edge of the study area we don't feel it appropriate to interpret that prediction with this particular set of data. The third population is directly south of Cape Breton and just west of Sable Island, the small sliver of land seen in the open ocean. This

population is a fair bit smaller than the first population both in geographical area occupied and in peak density, but the population seems to be fairly stable during the three years. The area between the first and third populations is not completely uninhabited by haddock, so while these two populations do seem to be distinct we might still expect some movement of individuals between the two populations. Of course it is impossible to say for sure if this individual movement is occurring and an animal movement study or a capture-recapture study would be more appropriate to answer that question. A genetic study would also show if these two populations are indeed distinct.

The forecast into the next year (2017) looks nearly identical to the map from 2016, but this is expected since the model uses an AR(1) structure and the AR(1) correlation was estimated to be very high at +0.95. Arguably the forecasted standard errors are the more interesting part of the projections since they allow use to quantify how sure we are of the predictions. The standard errors for the historically (at least from 2014-2016) uninhabited areas are very close to zero, so the model is fairly confident that new populations will not spontaneously arise from nowhere. The standard errors for the population centres are higher, so while the model projects the population centres to be fairly stable it does not rule out the possibility of fairly dramatic changes in density in those areas.

Disregarding any insights we would gain from looking further back than 2014, we identify the emerging hotspot in the Bay of Fundy as an area of particular management interest. The two main populations discussed above have a historical record which shows they are fairly stable and so management of those areas probably does not need to be altered much. The new hotspot in the Bay of Fundy doesn't have the same longevity and deserves more thought. If conservation is the goal, would it be worth spending the resources to designate that area as a temporary fisheries closure to try and protect the new population? Or will the haddock have moved back to the main population in a year and those resources would be better spent elsewhere? If fishing is the goal, is the allure of a high-risk high-reward area close to the shoreline enough to drag part of the fishing fleet away from the more stable areas that are farther out in the ocean? No statistical model can answer these questions, but the visualisations

certainly provide a useful starting point for identifying areas of interest and directions for future studies.

3.5 Discussion

The emergence and importance of citizen science in ecology, the ubiquity of atmospheric raster data from satellite images, and the desire to understand complex geographical processes have created a huge collection of spatio-temporally referenced point data. While these datasets can be quite complex, they are often conceptually similar and are too numerous for each researcher to have to implement their own model from scratch. We have introduced a new modelling framework that can be used for common ecological survey data, expanding the toolbox available to researchers.

The structure of our modelling framework provides access to simulation-based methods for model interpretation and validation which is otherwise quite difficult for spatio-temporal data. We showed one method for checking the response distribution using simulation methods but the fitted model could also have other problems such as collinearity in covariates, a mis-specified structure for the temporal random effects, an incorrect spatial covariance function, and more. The relevant literature is rather undeveloped and there are not many common practices, but the method we use for the response distribution could potentially be modified to inspect other parts of the model. The general procedure would be simulating datasets from multiple alternative models including the fitted model, and comparing the residuals from the original dataset to those of the simulated datasets. Future research could focus on which alternative models to use and how to compare between them in order to check for validity of the temporal and spatial structures assumed in the model.

We end with a discussion of data sources and applications that could benefit from our modelling framework. Additional modelling may be necessary to fully capture the intricacies of each dataset, for example preferential sampling in the citizen science eBird database may be too important to ignore, however the modelling presented here represents an important first step for providing a easy-to-use modelling tool.

eBird

A real-time, online checklist program, eBird has revolutionised the way that the birding community reports and accesses information about birds. eBird provides rich data with basic information on bird abundance and distribution at a variety of spatial and temporal scales. eBird is one of the largest and fastest growing biodiversity data resources in existence.

This data enables us to look at population changes over space and time. If we consider data from Nova Scotia, Canada, there are about 25,000 checklists provided by about 600 observers on a yearly basis with about 350 species reported annually. A particular question of interest relates to changes in abundance of particular species over time and space in response to climate change. For species such as boreal chickadee and Bicknell's thrush where Nova Scotia is at the southern limit of their range, anecdotal evidence shows these species much less abundant in southern Nova Scotia in recent years. In addition Bicknell's thrush is a bird of the highlands and boreal forest. With the models proposed in this paper we can answer important questions about their population trends temporally and spatially.

Disease Mapping

Disease mapping has its own set of complications – preferential sampling and privacy concerns among them – but the basic scenario is similar to the Carolina wren analysis. We refer to [Giorgi et al., 2018] for an introduction to spatio-temporal models for disease prevalence. The model they explore is a binomial GLMM which can be fit using the `staRVe` package.

Many disease prevalence datasets must comply with privacy regulations, and thus are typically reported as areal data. However following [Giorgi et al., 2018] these areal data can sometimes be treated as point-referenced data. Their example samples individual villages and treats each village as a single point. In this case the size of the village is small compared to the distance between villages, which allows them to treat the observations as point data.

In other cases where data is point-referenced but still restricted by privacy concerns the data are usually available to only a small set of specialists, perhaps containing no

dedicated statisticians. Our package could serve a vital role here in supplying an easy to use method with little advanced coding necessary.

Fisheries Science

The original motivating example for our development of the `staRVe` package is the analysis of fisheries data in Atlantic Canada, so we add to some of the more general ideas introduced in our analysis of haddock. Imagine we had collected data on lobster abundance off the coast of Maine and discovered a potential northward shift which could have huge economic consequences. A large enough shift north or east, even in the range of 150km, could see the lobster stock cross international boundaries from Maine in the United States to New Brunswick and Nova Scotia in Canada, making the lobster unavailable to American fishers (but available to Canadian fishers). Identifying the causes of these shifts can help determine if they are likely to be temporary or permanent. In addition, reliable forecasts of lobster stock behaviour are invaluable in planning any stock management strategies, ecosystem protection, or economic decisions in response to these changes.

Acknowledgments

This research was funded by a Canadian Statistical Sciences Institute Collaborative Research Team Project and the Ocean Frontier Institute. EL is also supported by a Vanier Canada Graduate Scholarship and Killam Predoctoral Scholarship. We thank the Department of Fisheries and Oceans Canada for providing the survey data used in the haddock analysis.

Conflict of Interest Statement

The authors have no conflicts of interest to disclose.

Author Contributions

EL designed the model and package with input from CF and JMF. All authors contributed to writing the paper and approved it for publication.

Data Availability

The data used in this paper are included as part of the `staRVe` package, which is available at the first author's GitHub page (https://github.com/lawlerem/staRVe). The specific version of the package used for this manuscript is archived at doi:10.5281/zenodo.5823555.

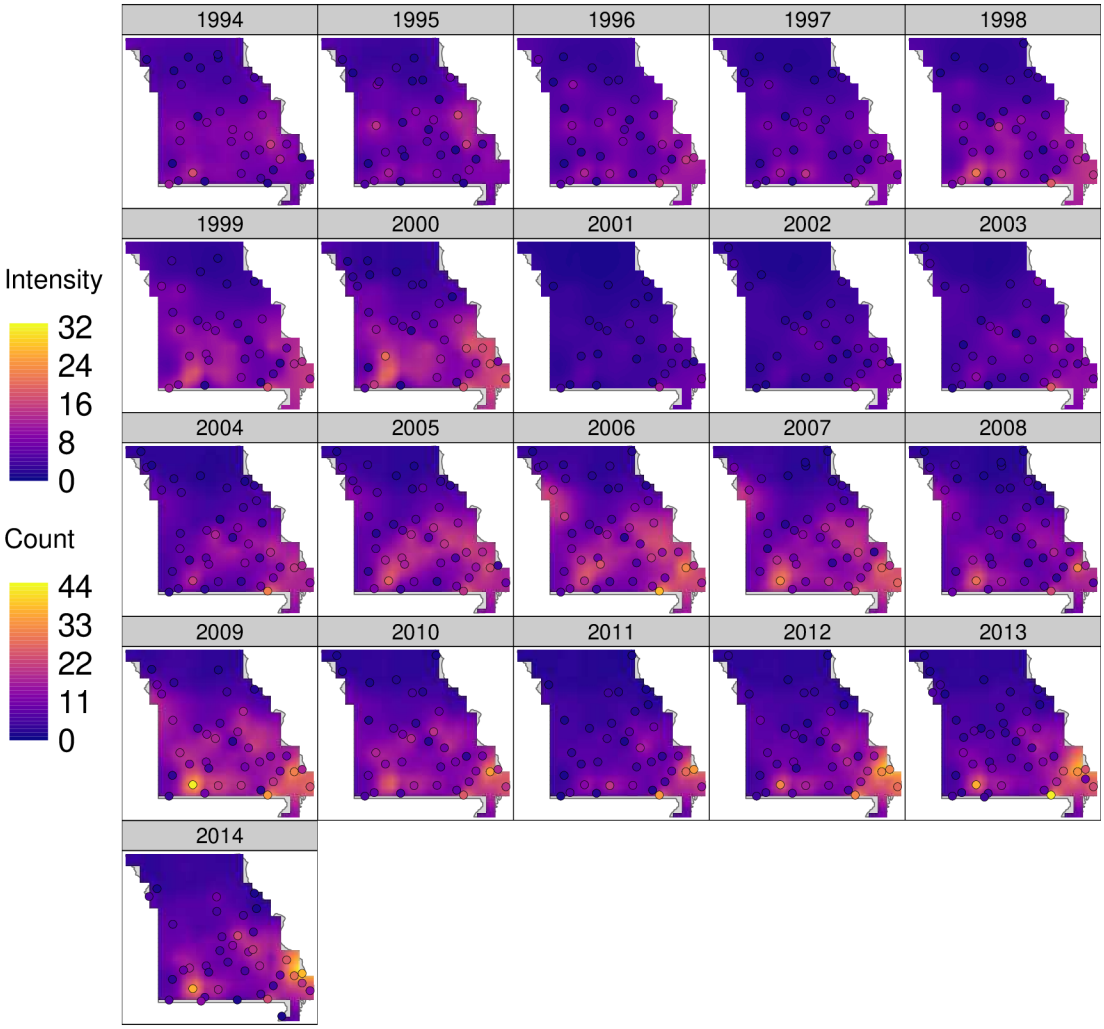

Figure 3.6: Model-based predictions of mean intensity for all years of the Carolina wren dataset.

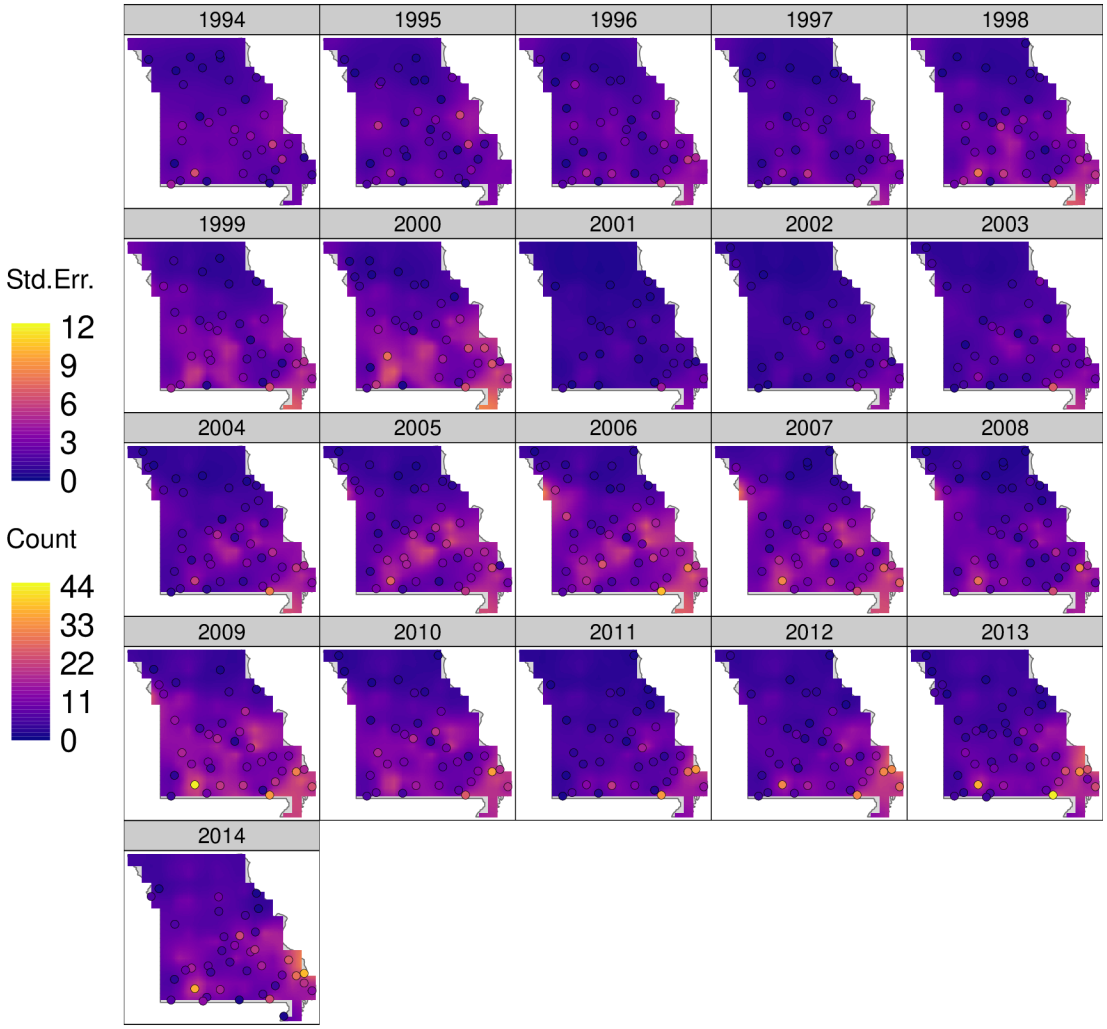

Figure 3.7: Model-based standard errors for predicted mean intensity for all years of the Carolina wren dataset.

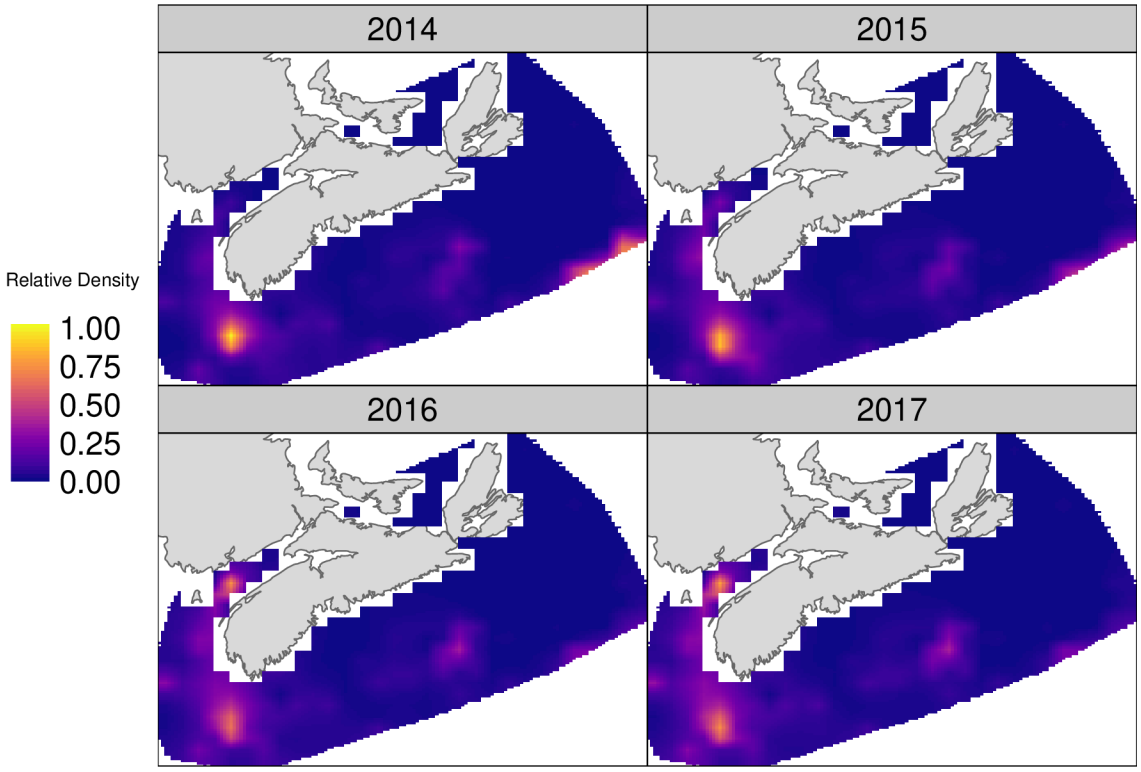

Figure 3.8: Model-based predictions of relative density for 2014-2017 of the haddock survey.

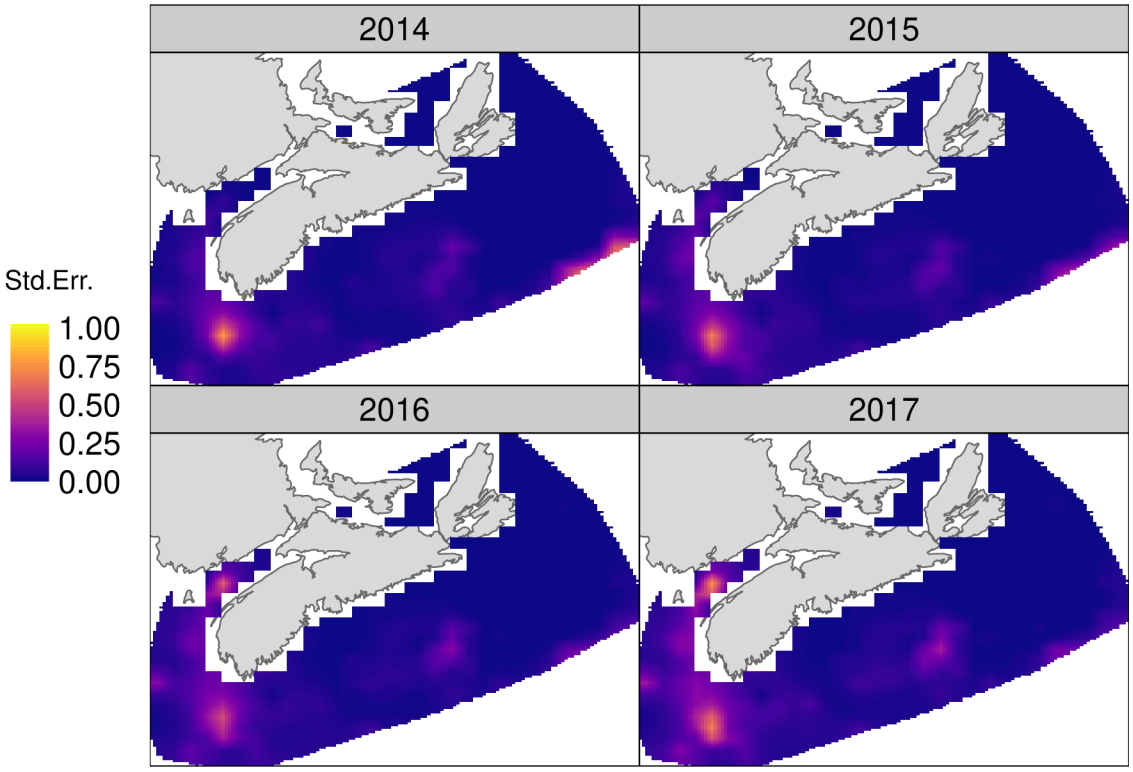

Figure 3.9: Model-based standard errors of predicted relative density for 2014-2017 of the haddock survey.

Chapter 4

From Fish to Fishes: A Multivariate Extension to the staRVe Model Using Copulas

4.1 Introduction

Age-structured populations. Catch and bycatch. Predator and prey. Many ecological questions involve more than just a single element of interest. The dynamics of spawning stock biomass depend on the growth of young fish into mature adults. Population centers of at-risk species aren't necessarily in danger of bycatch unless a commercially valuable species also inhabits the same area. Prey species decline when their predators abound.

Modelling the individual elements of these phenomena separately from each other clearly misses an essential part of the ecology. For example estimates of spawning stock biomass will be improved by incorporating data on juveniles so that the dynamics of ageing are captured. In addition to their multivariate nature, ecological phenomena evolve over time and space. With any two features – multivariate, spatial, temporal – the literature is vast with multiple decades of research. The supporting literature for multivariate spatio-temporal modelling – with all three types of dependence – is much smaller and still in active development [Wikle, 2015, Zammit-Mangion et al., 2015, Salvaña and Genton, 2020].

In this paper we present a simple yet flexible way to extend a univariate spatio-temporal model into a multivariate one. We accomplish this using copulas, which tie together arbitrary univariate models through their quantiles. Quantiles are nearly trivial to compute for the multivariate normal distribution once the precision matrix is known making our method natural for many current spatio-temporal models that work directly with the precision matrix, such as the stochastic partial differential

equation approach [Lindgren et al., 2011]. Quantiles are also easy to compute when the dimension of the multivariate normal distribution is small and the covariance matrix is known, making models which use one-step-ahead conditional distributions such as nearest neighbour Gaussian processes another viable candidate for our copula approach [Datta et al., 2016]. Copulas have only very recently started to appear in the statistical ecology literature, for example [Anderson et al., 2019] and [Liebscher et al., 2022]. Anticipating some unfamiliarity we provide a brief introduction to copulas before utilizing them to extend the staRVe model, which synergises well with the needs of a copula.

4.1.1 Crash Course on Copulas

A copula is a joint probability distribution whose marginal distributions are uniformly distributed on the unit interval. However this simple definition doesn't quite do them justice. Since the quantiles of any (continuous) random variable are uniformly distributed on the unit interval, copulas can be used to model the dependence structure between random variables. Depending on the particular marginal distributions of the random variables, describing their joint relationship directly can be difficult.

To see a concrete example of copulas, we can write the joint density of a set of correlated standard normal random variables in two ways:

$$f_1(\mathbf{x}) = \det\left(2\pi\boldsymbol{\Sigma}\right)^{-\frac{1}{2}} \ \mathrm{Exp}\left[-\frac{1}{2}\mathbf{x}^T\boldsymbol{\Sigma}^{-1}\mathbf{x}\right] \tag{4.1}$$

$$f_2(\mathbf{x}) = \left(\det\left(2\pi\mathbf{I}\right)^{-\frac{1}{2}} \ \mathrm{Exp}\left[-\frac{1}{2}\mathbf{x}^T\mathbf{I}\mathbf{x}\right]\right)\left(\det\left(\boldsymbol{\Sigma}\right)^{-\frac{1}{2}} \ \mathrm{Exp}\left[-\frac{1}{2}\mathbf{x}^T\left(\boldsymbol{\Sigma}^{-1}-\mathbf{I}\right)\mathbf{x}\right]\right) \tag{4.2}$$

The standard form f_1 contains both the standard normal marginal distributions as well as the correlation matrix ($\boldsymbol{\Sigma}$) in a single term. The second form f_2 factors the joint density into the marginal distributions on the left and their dependence structure or correlation on the right. Both f_1 and f_2 describe exactly the same joint distribution but conceptually f_2 handles the correlation separately from the marginal distributions, which is the same compartmentalization that copulas achieve. In fact, if $\mathbf{q}$ is a vector of Uniform$[0,1]$ quantiles and Φ is the standard normal distribution function then

replacing $\mathbf{x}$ with $\Phi^{-1}(\mathbf{q})$ in the second factor of f_2 gives us the density of the Gaussian copula

$$c_{\text{Gaus}}(\mathbf{q}) = \det(\mathbf{\Sigma})^{-\frac{1}{2}} \cdot \text{ Exp}\left[-\frac{1}{2}\Phi^{-1}(\mathbf{q})^T \cdot \left(\mathbf{\Sigma}^{-1} - \mathbf{I}\right) \cdot \Phi^{-1}(\mathbf{q})\right] \tag{4.3}$$

where $\mathbf{\Sigma}$ is a correlation matrix.

Copulas have been used fairly extensively in the econometrics literature. We will focus on a paper by [Patton, 2006] which develops the foundation for our extension to the staRVe model. In its simplest form the paper introduces cross-sectional copulas for modelling dependence between two autoregressive order 1, or AR(1), Gaussian time series. There are three components to this model:

1. The marginal AR(1) process of X:

$$X_t|x_{t-1} \sim N\left(\phi_x x_{t-1}, \sigma_x^2\right) \tag{4.4}$$

2. The marginal AR(1) process of Y:

$$Y_t|y_{t-1} \sim N\left(\phi_y y_{t-1}, \sigma_y^2\right) \tag{4.5}$$

3. The cross-sectional copula for the dependence between X and Y:

$$\begin{bmatrix} q(X_t) \\ q(Y_t) \end{bmatrix} \sim c_{\text{Gaus}}(\rho) \tag{4.6}$$

   where $q(X_t) = \Phi\left(\frac{X_t - \phi_x x_{t-1}}{\sigma_x}\right)$ is the Uniform$[0,1]$ quantile corresponding to $X_t|x_{t-1}$ (similarly for $q(Y_t)$) and $c_{\text{Gaus}}(\rho)$ is the bivariate Gaussian copula with correlation parameter ρ.

For this particular example you could write the model equivalently as

$$\begin{bmatrix} X_t \\ Y_t \end{bmatrix} | \begin{bmatrix} x_{t-1} \\ y_{t-1} \end{bmatrix} \sim MVN\left(\begin{bmatrix} \phi_x x_{t-1} \\ \phi_y y_{t-1} \end{bmatrix}, \begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix}\right). \tag{4.7}$$

	Scenario 1		Scenario 2		Scenario 3		Scenario 4	
	X	**Y**	**X**	**Y**	**X**	**Y**	**X**	**Y**
mean μ	0.00	0.00	0.00	0.00	0.00	-1.0	0.00	2.0
autocorrelation ϕ	0.95	0.95	0.95	0.95	0.95	0.0	0.95	0.9
forecast std. dev. σ	0.40	0.40	0.40	0.40	0.40	0.3	0.40	1.0
copula family	Independence		Gaussian		Gaussian		Gumbel	
copula parameter	– –		0.90		0.90		4.00	

Table 4.1: Parameter values used to simulate dependent time series. All series are simulated from a Gaussian AR(1) series except for **Y** in scenario 4, which is simulated from a Gamma AR(1) series. Scenario 1 uses the independence copula, scenarios 2 and 3 use a Gaussian copula, and scenario 4 uses a Gumbel copula.

However the copula formulation is much more flexible. You could change the distribution of X_t to a Gamma distribution without worrying about the other two components, or change Y to follow an ARIMA(1,1,1) process, or change the copula to a Student's t copula. On the other hand the multivariate normal formulation is locked in to have normal marginals and a Gaussian joint distribution with linear correlation.

A few simulated pairs of time series are shown in figure 4.1. The first scenario consists of realizations **X** and **Y** that are independent and identically distributed Gaussian AR(1) series; the second scenario has **X** and **Y** highly dependent with a Gaussian copula, and simulated from identically distributed Gaussian AR(1) series; the third has **X** and **Y** highly dependent with a Gaussian copula, but simulated from two different Gaussian AR(1) series; and to showcase the versatility of copulas the fourth has **X** and **Y** highly dependent with a Gumbel copula, **X** is simulated from a Gaussian AR(1) series, and **Y** is simulated from a Gamma AR(1) series and then shifted down for better visualisation. The parameter values used are given in Table 4.1. The cross-correlation functions $\text{Cor}\left[X_{t+k},\ Y_t\right]$ are shown in figure 4.2.

The first pair is given as a reference for independent time series. Since the AR(1) distributions are identical and the copula correlation is very high the two simulated realizations track each other very closely with only a few small differences. The cross-correlation function between **X** and **Y** shows a marginal correlation between X_t and Y_t close to the copula correlation $\rho = 0.9$.

The third pair shows how the copula correlation between $X_t|x_{t-1}$ and $Y_t|y_{t-1}$ differs

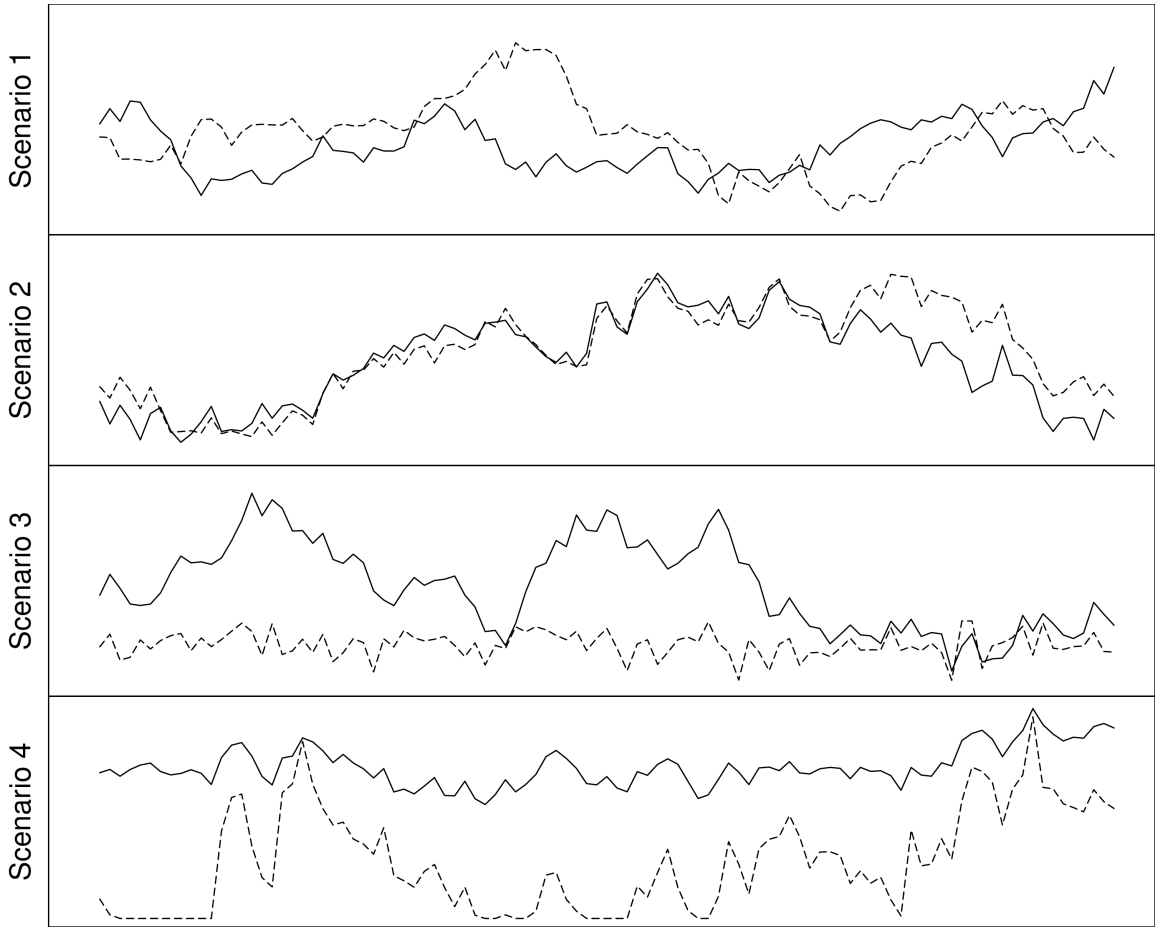

Figure 4.1: A few examples of AR(1) time series. For each scenario, $\mathbf{X}$ is shown as a solid line and $\mathbf{Y}$ is shown as a dashed line. **Scenario 1**: Independent and identically distributed Gaussian AR(1) series; **Scenario 2**: Gaussian copula and identically distributed marginal Gaussian AR(1) series; **Scenario 3**: Gaussian copula with two different marginal Gaussian AR(1) series; **Scenario 4**: Gumbel copula with a marginal Gaussian AR(1) series ($\mathbf{X}$) and a marginal Gamma AR(1) series ($\mathbf{Y}$).

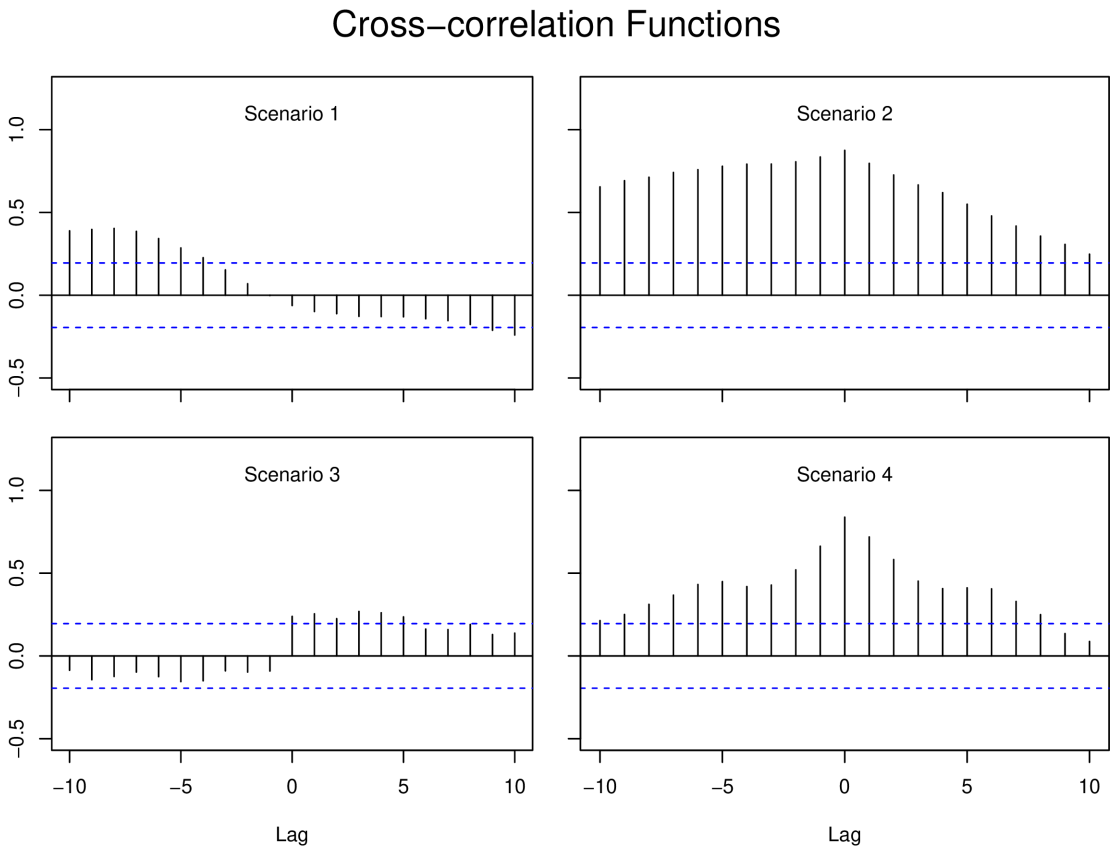

Figure 4.2: Cross-correlation function between two time series for (left) identically distributed AR(1) processes with $\rho = 0.9$ and (right) differently distributed AR(1) processes with $\rho = 0.9$. The blue dotted line is the threshold of statistical significance for the 95% confidence band.

from the marginal correlation between X_t and Y_t. The two simulated realizations don't track each other like the second pair did but a crest (dip) in $\mathbf{X}$ happens at the same time as a crest (dip) in $\mathbf{Y}$, and the two series are clearly highly dependent. However, the cross-correlation function points out a very weak, almost negligible, marginal correlation between X_t and Y_t.

The fourth pair shows just one example of how the flexibility of copulas can lead to behaviour that is difficult to model by directly specifying a joint distribution for the two time series. The Gamma distribution with its positive skew allows for the occasional large increase with a gradual decrease down to a constant minimum level, behaviour which would be uncharacteristic of a Gaussian distribution. Then the Gumbel copula has a property called *upper tail dependence* which means, colloquially, that extreme increases will occur in both time series at the same time. The dependence between the two series is not as strong when the series are either relatively constant or decreasing than it is when the two series are increasing. The behaviour produced by the Gumbel copula cannot be accurately captured by the linear correlation of a Gaussian copula.

Copula models are incredibly flexible due to their modular nature, which allows for the separation of the marginal behaviours from their dependence structure. Even with Gaussian marginal distributions a copula model is not restricted to a Gaussian copula. One could use e.g. a Student's t copula, a Gumbel copula, or a Clayton copula. There are a few other reasons why copulas are useful. As we saw above it is possible, and often easy, to simulate from a copula. You can estimate a copula either jointly with the marginal distributions or post-hoc after the marginal distributions have been estimated. The only real restriction for copulas are that quantiles need to be easily computable from the marginal processes.

4.2 Cross-Sectional Copulas for the staRVe Model

The univariate staRVe model takes the form of a hierarchical generalized linear mixed model

$$Y_{i,t}(\mathbf{s}_i) \mid \mu_{i,t}(\mathbf{s}_i) \sim f_\theta(y_{i,t}(\mathbf{s}_i);\ \mu_{i,t}(\mathbf{s}_i)) \tag{4.8}$$

$$\mu_{i,t}(\mathbf{s}_i) \mid w_t(\mathbf{s}) = g^{-1}(\mathbf{X}_i\boldsymbol{\beta} + w_t(\mathbf{s}_i)) \tag{4.9}$$

$$W_t(\mathbf{s}) \mid \{w_{t-1}(\mathbf{s}),\ \boldsymbol{\epsilon}\} \sim \text{NNGP}(\phi \cdot (w_{t-1}(\mathbf{s}) - \epsilon_{t-1}) + \epsilon_t,\ \mathcal{C}(\mathbf{s}_1, \mathbf{s}_2)) \tag{4.10}$$

$$\boldsymbol{\epsilon} \sim \text{N}\left(\mu,\ \frac{\sigma^2}{1-\phi^2}\boldsymbol{\Sigma}\right) \quad \text{where} \quad \Sigma_{ij} = \phi^{||i-j||}. \tag{4.11}$$

where i indexes the observation, t indexes time, and s indexes location. The spatio-temporal random effects $W_t(\mathbf{s})$ within a single year follow a nearest neighbour Gaussian process and each individual location follows an AR(1) process across years. A nearest neighbour Gaussian process requires a set of nodes $\mathcal{S}$ at which the random effects are located, and we use the same random effect locations every year. If an observation location does not coincide with a random effect location we add another random effect node at that observation location, but only for the year in which that observation is present.

To convert the univariate model to a multivariate model we will introduce a dependence structure between the spatio-temporal random effects of the various response variables. The observation locations of the different response variables do not need to be the same, though clearly the estimation of the dependence between variables will be improved if they are observed together. We will however require that the random effect locations $\mathcal{S}$ of the various response variables are identical.

We first describe how to standardize the spatio-temporal random effects after which their quantiles are trivially computed. Then we write the likelihood of the multivariate staRVe model, and explore the behaviour of a bivariate model under various simulation scenarios.

4.2.1 Spatio-Temporal Random Effects to IID Standard Normal Random Effects

In order to compute the likelihood of the staRVe model, we have a series of univariate one-step-ahead distributions for each individual random effect $W_t(\mathbf{s}_i) | \mathbf{w}_{t-1}(\delta\mathbf{s}_i)$ where $\delta\mathbf{s}_i$ is the set of nearest neighbours of location $\mathbf{s}_i$. This one-step-ahead distribution is

$$W_t(\mathbf{s}_i) \;\Big|\; \{\mathbf{w}_{t-1}(\delta\mathbf{s}_i),\ \boldsymbol{\epsilon}\} \sim N\left(m_t(\mathbf{s}_i) + \mathbf{c}^\top\boldsymbol{\Sigma}^{-1}\left(\mathbf{w}_t(\delta\mathbf{s}_i) - \mathbf{m}_t(\delta\mathbf{s}_i)\right),\ \tau^2\rho - \mathbf{c}^\top\boldsymbol{\Sigma}^{-1}\mathbf{c}\right) \tag{4.12}$$

where $m_t(\mathbf{s}) = \phi\cdot(w_{t-1}(\mathbf{s}) - \epsilon_{t-1}) + \epsilon_t$, ϕ a temporal autocorrelation parameter, $\mathbf{c}$ is the cross-covariance matrix between $W_t(\mathbf{s}_i)$ and $\mathbf{W}_{t-1}(\delta\mathbf{s}_i)$, $\boldsymbol{\Sigma}$ is the covariance matrix of $\mathbf{W}_{t-1}(\delta\mathbf{s}_i)$, and τ and ρ are spatial correlation parameters. This conditional distribution is already computed when evaluating the likelihood so it is trivial to standardize $W_t(\mathbf{s}_i)$ and compute the its quantile like we did with $\mathbf{X}$ and $\mathbf{Y}$ in section 4.1.1. We let

$$q(W_t(\mathbf{s}_i)) = \Phi\left(\frac{W_t(\mathbf{s}_i) - \left[m_t(\mathbf{s}_i) + \mathbf{c}^\top\boldsymbol{\Sigma}^{-1}\left(\mathbf{w}_t(\delta\mathbf{s}_i) - \mathbf{m}_t(\delta\mathbf{s}_i)\right)\right]}{\sqrt{\tau^2\rho - \mathbf{c}^\top\boldsymbol{\Sigma}^{-1}\mathbf{c}}}\right) \tag{4.13}$$

and similarly defined for the realized quantile $q(w_t(\mathbf{s}_i))$. Because these quantiles are computed relative to the one-step-ahead conditional distributions the entire set of quantiles across all years and random effect locations will be independent and uniformly distributed on the unit interval [Rosenblatt, 1952, Dawid, 1984]. If we had used the marginal distribution of $W_t(\mathbf{s}_i) \sim N(m_t(\mathbf{s}_i), \tau^2\rho)$ to compute the quantiles, then the quantiles would be spatially correlated and would be inappropriate to use in a cross-sectional copula between variables.

4.2.2 Multivariate staRVe Model

The multivariate extension to the staRVe model follows naturally now that we have independent quantiles for the random effects. Just as we did in section 4.1.1 we split the model into components. The marginal components for each variable are all

univariate staRVe models

$$Y_{i,t,v}(\mathbf{s}_i) \mid \mu_{i,t,v}(\mathbf{s}_i) \sim f_{\theta,v}(y_{i,t,v}(\mathbf{s}_i);\ \mu_{i,t,v}(\mathbf{s}_i)) \tag{4.14}$$

$$\mu_{i,t,v}(\mathbf{s}_i) \mid w_{t,v}(\mathbf{s}_i) = g^{-1}(\mathbf{X}_{i,v}\boldsymbol{\beta}_v + w_{t,v}(\mathbf{s}_i)) \tag{4.15}$$

$$W_{t,v}(\mathbf{s}) \;\Big|\; \{w_{t-1,v}(\mathbf{s}),\ \boldsymbol{\epsilon}_v\} \sim \mathrm{NNGP}(\phi_v \cdot (w_{t-1,v}(\mathbf{s}) - \epsilon_{t-1,v}) + \epsilon_{t,v},\ \mathcal{C}_v(\mathbf{s}_1, \mathbf{s}_2)) \tag{4.16}$$

$$\boldsymbol{\epsilon}_v \sim \mathrm{N}\left(\mu_v,\ \frac{\sigma_v^2}{1-\phi_c^2}\boldsymbol{\Sigma}_v\right) \quad \text{where} \quad \Sigma_{ij,v} = \phi_v^{||i-j||}. \tag{4.17}$$

where v indexes the variable. The response distributions, covariates, and spatial and temporal correlation parameters can all be changed to whatever best suits each individual variable. The copula component is given by

$$\begin{bmatrix} q(W_{t,1}(\mathbf{s})) \\ q(W_{t,2}(\mathbf{s})) \\ \vdots \\ q(W_{t,V}(\mathbf{s})) \end{bmatrix} \sim c(\boldsymbol{\rho}_c) \tag{4.18}$$

where $c(\boldsymbol{\rho}_c)$ is some choice of copula with parameters $\boldsymbol{\rho}_c$ describing the dependence between variables. We require the random effect locations $\mathcal{S}$ of each variable to be identical so that the log-likelihood of the copula component can be computed. When computing the log-likelihood function ℓ_c for the copula component of the model there will be a single term for each location and time point pair, that is

$$\ell_c = \sum_{\mathbf{s}_i \in \mathcal{S}} \sum_{t=0}^{T} \log f_c \left(\begin{bmatrix} q(w_{t,1}(\mathbf{s}_i)) \\ q(w_{t,2}(\mathbf{s}_i)) \\ \vdots \\ q(w_{t,V}(\mathbf{s}_i)) \end{bmatrix},\ \boldsymbol{\rho}_c \right) \tag{4.19}$$

The one-step-ahead distributions for $W_{t,v}(\mathbf{s}_i) | \mathbf{w}_{t-1,v}(\delta\mathbf{s}_i)$ were already computed for the univariate staRVe model, so following section 4.2.1 computing the quantiles $q(w_{t,v}(\mathbf{s}_i))$ comes with negligible additional computational burden. Additionally the computational cost to evaluate the log-likelihood of the copulas can be very cheap

compared to the cost of evaluating the rest of the likelihood function. For example the Gaussian copula comes with complexity $O(V^3)$ needed to invert a correlation matrix, which for small V ($V \approx 15$) is not problematic. Memory issues can quickly rear their head as the number of variables increase but this is due to having a set of random effects for all of the different variables and unrelated to the copula itself.

4.3 Spatial Simulations

4.3.1 Description of Simulation Scenarios

We follow the same general framework as the simulations in Chapter 3, which was adapted from [Zhang, 2004]. We simulate random effects and observations for two dependent variables X and Y on a grid covering the unit square, with points at $(0.1{*}x, 0.1{*}y)$ for $x, y \in 0,\ 1,\ ...,\ 10$ and $(0.1{*}x{+}0.05, 0.1{*}y{+}0.05)$ for $x, y \in 0,\ 1,\ ...,\ 9$ for a total of 221 locations. We also simulate random effects and observations for X and Y on a transect covering $(0.387, 0.01 * y)$ for $y \in 0,\ ...,\ 100$. We fit a model using the various subsets of the simulated data, and then predict values of X for the transect.

Scenarios 1 through 5 investigate the effects of removing observations of X near the prediction locations, including observations of Y at the predictions locations, and of analyzing X by itself or incorporating Y either through a copula or as a linear fixed effect. We simulate a single dataset that will be reused for scenarios 1 through 5 so that we can isolate the effects of the incorporating the second variable and changing which data locations are available for each variable. We then fit two models for each scenario: the first model includes only the data for X while the second model includes the copula and uses the data for both X and Y. All the parameters of the models are estimated via maximum likelihood. In each scenario the predictions for the joint model are compared to the predictions from the univariate model.

In scenario 1 we establish the baseline performance of the model. The model is fitted using data for both X and Y at the grid locations, so that X and Y are always observed together. The data for scenario 2 is the same as the data for scenario 1 except we remove observations of X in a buffer of radius 0.1 around the prediction locations.

For this scenario, the 42 observations within the vertical band $0.287 < x < 0.487$ only include observations of Y. Scenarios 3 and 4 are identical to scenario 1 but in addition to observations of X and Y at each of the grid locations, we include observations of Y at the prediction locations. These two scenarios are designed to test the effect of including these observation locations as part of the persistent graph or as part of the transient graph (see Chapter 3). In scenario 3 the persistent graph does not include the prediction locations so that the random effects used to predict X are the random effects on the grid, and the predictions of X are only tangentially informed by the observations of Y at the prediction locations In scenario 4 the persistent graph does include the prediction locations so that the random effects used to predict X include the random effects of the prediction locations, which are informed directly by the additional observations of Y. Briefly breaking from the copula approach, scenario 5 incorporates Y as a linear covariate for X instead of modelling them jointly but otherwise is identical to scenario 4.

In scenario 6 we alter scenario 4 by comparing how the predictions of X are affected by changes in the response distribution of Y. We change the distribution of Y to either a Gaussian response, a Gamma response, a Poisson response, or a Bernoulli response. To do this we keep the random effects and observations of X the same as they were in the previous scenarios. We preserve the spatial structure of the random effects of Y, but change their global mean (by adding or subtracting a constant to each random effect) and magnitude (by multiplying all of the random effects by a constant). We first normalize the random effects by subtracting their sample mean and then divide by their sample standard deviation, then scale and shift them to their new mean and standard deviation. We then simulate a new set of observations conditional on these shifted and scaled random effects to produce observations for the different response distributions of Y. We choose the scaling and shifting parameters to achieve fairly informative but realistic looking data.

The parameters used for simulations of X are global mean $\mu = 0$, spatial standard deviation $\tau = 1$, spatial range $\rho = 0.2$, and response standard deviation $\sigma = 1$. For Y the parameters are global mean $\mu = 3$, spatial standard deviation $\tau = 1$, spatial range $\rho = 0.6$, and response standard deviation $\sigma = 0.4$. We use a bivariate Gaussian

copula with correlation parameter $\rho_c = 0.75$ so that the random effects of X and Y are strongly correlated.

For scenario 6 the Gaussian random effects are unchanged. The Gamma random effects are scaled to have sample standard deviation 1 and sample mean 2, and the response distribution has observation variance $\sigma = 0.4$ similar to the Gaussian response. The Poisson random effects are scaled to have sample standard deviation 2 and mean 1. The Bernoulli random effects are scaled to have sample standard deviation 2.5 and sample mean -0.2

4.3.2 Results

Scenario 1: In Which X and Y Observed Together on a Grid

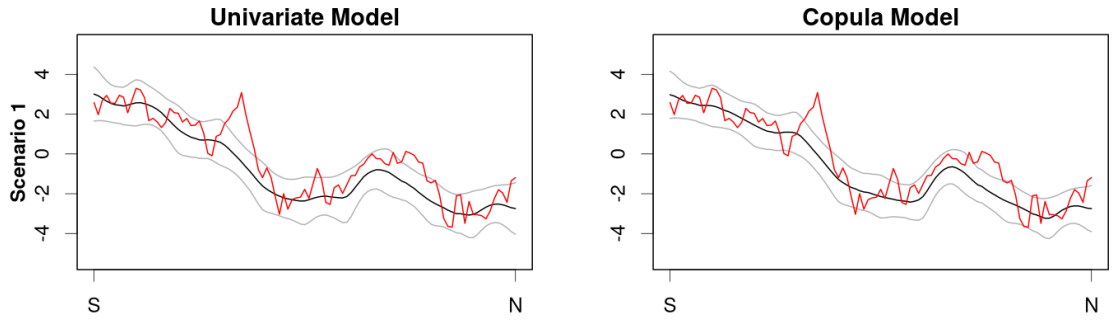

Figure 4.3: X and Y are observed together at the grid locations. The true simulated values of X at the prediction locations are shown in red. The predicted mean is shown in black with a 95% confidence band marked in grey.

The predictions under the univariate model for X are not changed by including data for Y in this scenario. This is not entirely surprising, since copulas are designed so that they do not interfere with the marginal distributions of the individual variables. While the predictions of X are not affected, the copula parameter was estimated at $\hat{\rho}_c = 0.658$ with a standard error of 0.073 which shows the dependence between X and Y can be accurately estimated in this ideal scenario.

Scenario 2: In Which X Loses Its Integrity

The predictions of X using the univariate model are considerably less accurate than they were in scenario 1, however the loss of accuracy from removing observations

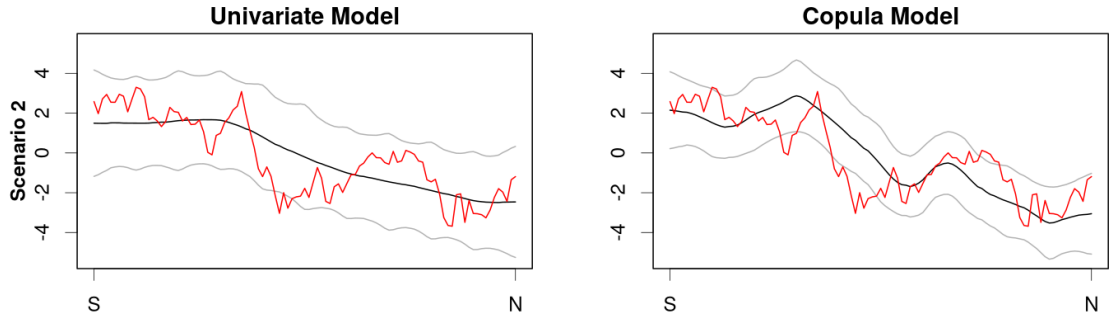

Figure 4.4: X and Y are observed together at the grid locations, but the 42 observations of X within a distance of 0.1 of the prediction locations are removed.

near the prediction locations is offset by a corresponding increase in the standard errors for prediction. The joint model for X and Y gives somewhat more accurate predictions than those for the univariate model, however they are not quite as good as the predictions for X in scenario 1. While extra data of Y can compensate for missing data in X, the observations of Y do not provide as much information as the data for X even with the strong dependence between X and Y. The copula parameter is estimated at $\hat{\rho}_c = 0.722$ with a standard error of 0.071, so despite some missing data the dependence between X and Y can still be estimated.

Scenario 3: In Which Y is Observed at the Prediction Locations

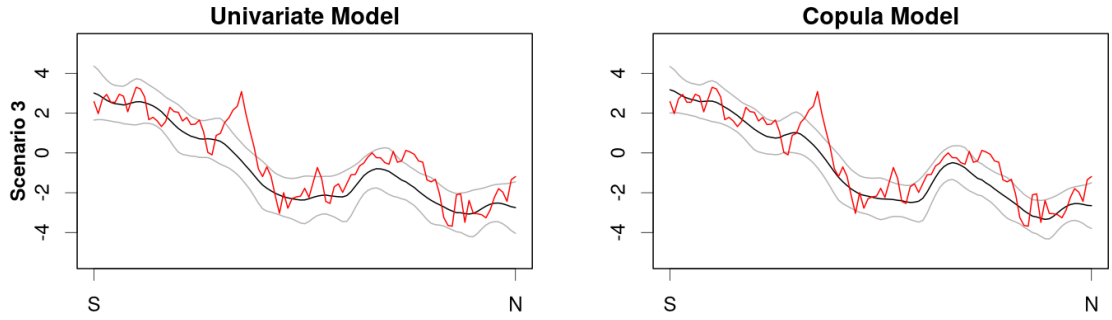

Figure 4.5: X and Y are observed together at the grid locations, and additional data for Y is included at the prediction locations. The persistent graph does not include the prediction locations.

The predictions of X are not changed by including data for Y at the prediction locations, at least when the persistent graph does not include the prediction locations. The copula parameter is estimated at $\hat{\rho}_c = 0.679$ with a standard error of 0.069, which

is slightly closer to the true value than the estimate from scenario 1.

Scenario 4: In Which the Prediction Locations are Added to the Persistent Graph

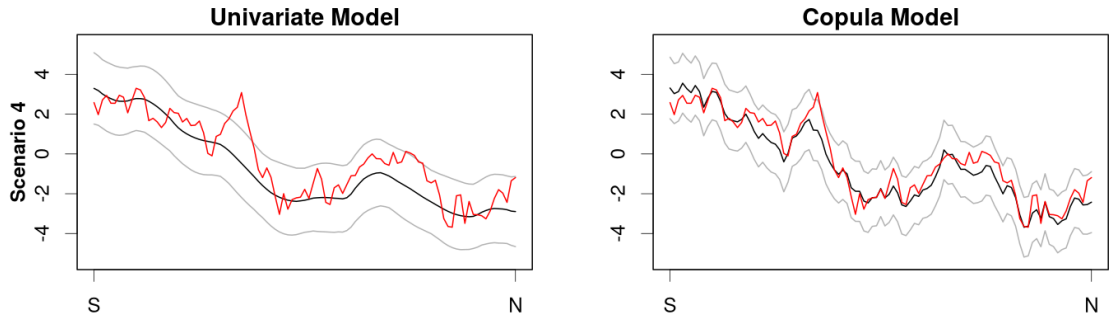

Figure 4.6: X and Y are observed together at the grid locations, and additional data for Y is included at the prediction locations. The persistent graph includes the prediction locations.

In opposition to scenario 3, the predictions of X are greatly improved by including the data for Y at the prediction locations when the persistent graph includes the prediction locations. The resulting predictions are more jagged than the predictions under the univariate model, which is more realistic-looking behaviour under the simulation scenario. In particular, the copula model in this scenario gives the first set of predictions that able to capture the sharp increase from $X \approx 0$ to $X \approx 3$ and immediate decrease from $X \approx 3$ to $X \approx -3$ just to the left of the center of the prediction locations. The copula parameter is estimated at $\hat{\rho}_c = 0.662$ with a standard error of 0.068.

Scenario 5: In Which Y is Used as a Linear Predictor

The results for this scenario are essentially identical to the results from scenario 4. This is more or less expected since we simulated the data using a Gaussian copula. A Gaussian copula gives rise to linear correlations as the dependence structure, so a linear regression should be able to identify this relationship between X and Y. However, the difference between the copula approach and the linear regression approach becomes clear when comparing parameter estimates. The regression parameter for Y is estimated at $\hat{\beta} = 0.334$ with a standard error of 0.039. Clearly the linear relationship

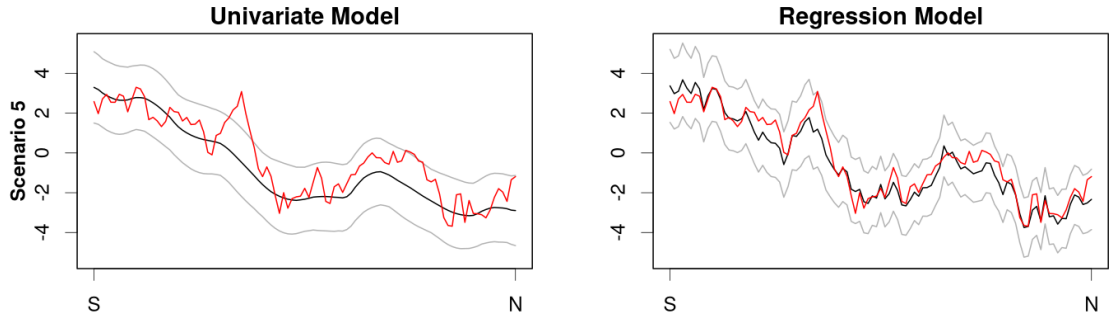

Figure 4.7: Y is included as a linear covariate in the model for X instead of modelling X and Y jointly through a copula. X and Y are observed together at the grid locations, and additional data for Y is included at the prediction locations. The persistent graph includes the prediction locations.

between X and Y is statistically significant but it is more difficult to use this regression coefficient to determine how closely related X and Y are, particularly in the context of spatial regression. The copula parameter estimated in scenario 4 is much easier to interpret since it is independent of the marginal distributions of X and Y.

Scenario 6: In Which Y has a Change of Costume

A major benefit of using copulas on the spatio-temporal random effects is that the response distribution of any of the response variables can be non-Gaussian and the approach still works. While there are some slight differences between them the predictions for the Gaussian, the Gamma, and the Poisson models are very similar with predictions that are quite close to the true simulated values. The Bernoulli model gives predictions that are virtually identical to the predictions under the univariate model in scenario 4, which does show that not all the different types of data for Y are able to lend enough information to X to significantly improve predictions. The Bernoulli model likely struggles because it is difficult to give precise estimates of the underlying spatio-temporal random field from data that consists strictly of 0s and 1s.

Encouragingly though, the copula parameter was able to be estimated under each of the models including the Bernoulli model. For the Gaussian model the estimate is $\hat{\rho}_c = 0.662$ with a standard error of 0.068, the Gamma model gives an estimate of $\hat{\rho}_c = 0.656$ with a standard error of 0.069, the Poisson model gives an estimate of $\hat{\rho}_c = 0.648$ with a standard error of 0.073, and the Bernoulli model gives an estimate

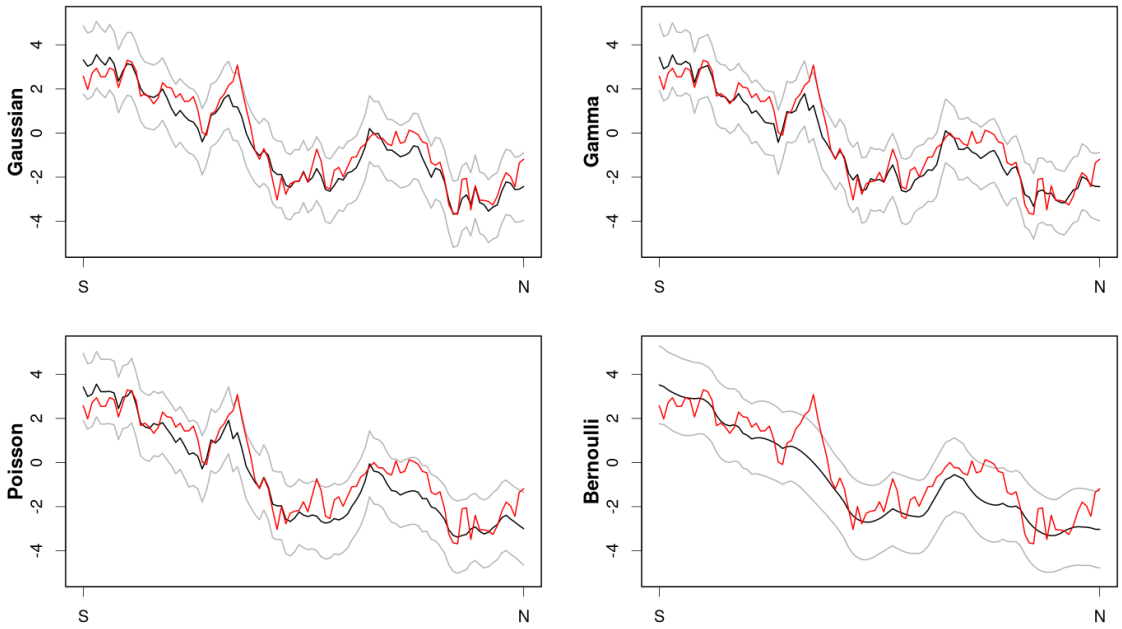

Figure 4.8: Scenario 6: The response distribution of Y is changed to either a Gaussian, a Gamma, a Poisson, or a Bernoulli distribution. X and Y are observed together at the grid locations, and additional data for Y is included at the prediction locations. The persistent graph includes the prediction locations.

of $\hat{\rho}_c = 0.653$ with a standard error of 0.115

4.4 Discussion

The copula approach to generalizing a univariate spatio-temporal model to a multivariate one is quite promising. While the simulations here only cover a purely spatial setting, the fact that the copula operates independently from the marginal distributions of each individual variable suggests that the results for the spatial setting will transfer easily to the spatio-temporal setting. Importantly, simulation scenarios 4 and 6 show that the copula approach can use additional data from auxiliary variables to improve predictions for a target variable and that many different response distributions can be used for the auxiliary variables. Despite the very good results from scenario 4, scenario 3 shows that the benefits of the extra data are only gained when the persistent graph includes the locations of the extra data. This suggests that the design of the persistent graph and transient graph needs to be looked at again and altered, particularly in the context of producing good predictions.

Comparing a copula approach with a regression approach shows that the prediction results are nearly identical, at least when a Gaussian copula is used. However the benefits of the copula approach are still apparent when trying to interpret the parameters governing the relationship between X and Y. The copula parameter is more interpretable than the regression parameter when trying to estimate how closely interdependent X and Y are, but comes at the cost of having to specify a model for the auxiliary variable Y. For future research, it will be interesting to see if the problem of spatial confounding in the regression approach can be solved by using the copula approach instead. Briefly, spatial confounding appears when a spatially-referenced covariate is colinear or correlated with the spatial random effects in a spatial regression [Paciorek, 2010]. When this is the case, the regression coefficients may be different in the spatial regression than they were in an ordinary linear regression and it is unclear which estimates should be used. Restricted spatial regression has been suggested as a solution to this problem, but restricted spatial regression has been studied and seen to perform poorly [Khan and Calder, 2020]. Since the copula approach estimates the marginal spatial structure of each variable completely separately from the dependence between variables, it seems a promising avenue forward for incorporating auxiliary variables in the presence of spatial confounding.

While effective, the dependence structures that copulas provide can be rather opaque and may not lend themselves to direct modelling of biologically important processes such as aging and survival. For example the copula structure presented here would not directly connect the number of fish in a cohort at year 1 with the number of fish in the same cohort the next year, even though those two number are very closely related. These processes could be modelled as part of the mean structure within a single variable or as a transition matrix to provide a more mechanistic model for the dependence between variables. The mechanistic description could then be used in addition to the copula approach to provide more accurate predictions and to infer a more mechanistic relationship between variable at the same time.

Chapter 5

Conclusion

5.1 CarHMM

The CarHMM for animal movement introduced in Chapter 2 has its origins in early behavioural state models such as [Morales et al., 2004] and [Jonsen et al., 2005], and directly responds to more recent models such as moveHMM [Michelot et al., 2016] and HMMM [Whoriskey et al., 2017] which together formalized the hidden Markov model as a key framework for linking animal movement and behaviour. The latter papers provided software that allowed practitioners to apply hidden Markov models for animal movement, however extensions to these models had already been discussed in the literature. Many of these extensions focussed on the dynamics of the behavioural state process, for example by explicitly modelling the amount of time spent in a particular behavioural state or by including a second layer of long-term behaviours [Langrock et al., 2012, Leos-Barajas et al., 2017]. The extensions provide important flexibility needed to capture a better description of the behaviours and motivations of an animal.

The CarHMM extends the hidden Markov model in a different direction by addressing the dynamics of the movement process. We showed that movement data originating from marine species can include additional autocorrelation in the step lengths that needs to be accounted for in order to accurately classify an animal's behavioural state from its movement data. Lagplots between consecutive step lengths can easily diagnose when the conditional autocorrelation is present in the data, and residual analysis can show if a fitted model has accounted for that extra structure in the data. The CarHMM introduces a simple-in-form generalization to a regular HMM, and thus is able to take advantage of all the tools developed for HMMs. These include the Viterbi algorithm for decoding the most likely sequence of behavioural states, forward

recursion for computing the log-likelihood, and easily computable residuals for model checking. Further, the improvements made by including conditional dependence in the step lengths can easily be combined with other alterations of the HMM like those mentioned above.

However, the CarHMM is still susceptible to many of the problems inherent to other hidden Markov models for animal movement. The behaviours estimated from the statistical model are idealized mathematical behaviours that may or may not correspond to the actual behaviours driving animal movement. Without additional confirmation through additional data sources such as video evidence, depth data, or body monitoring data such as heart rate, we cannot say for certain whether the mathematical behaviour we describe as “foraging" is actually a foraging behaviour. Different qualitative behaviour may drive similar-looking quantitative movement patterns.

Our model also assumes regularly timed location data with negligible observation error, which in practice is rarely the case particularly for marine species. The data we analyzed were accurate, however were irregularly spaced in time. We dealt with this through a rather simple linear interpolation, and although we gave some metrics to help guide this linear interpolation a more sophisticated approach is desired. Our generalization to the hidden Markov model could be applied to the multiple imputation approach to fitting hidden Markov models with observation error introduced in [McClintock, 2017].

Finally, determining the number of behavioural states to include in a hidden Markov model is already a difficult task [Pohle et al., 2017]. The lagplots we introduced make it possible to determine the number of behavioural states from data simulated from a hidden Markov model with a small number of states, however for data coming from a model with more states or coming from the CarHMM identifying the number of behavioural states to use seems to be an impossible task.

5.2 staRVe

After hierarchical Bayesian space-time models [Wikle et al., 1998] and geostatistical generalized linear mixed models [Diggle et al., 1998] popularized the use of spatial

statistics, researchers quickly found that blunt application of these models was computationally intractable for datasets of real interest. Intense effort went into computationally efficient alternatives to Gaussian process modelling and kriging, the results of which are neatly summed up in [Heaton et al., 2018]. Some of these computationally efficient alternatives are able to be embedded in larger models. In the staRVe model introduced in Chapter 3 and further developed in Chapter 4 we use one of the more recent alternatives, the nearest-neighbour Gaussian process, to model spatio-temporal variation. In order to accommodate the study design typical of ecological survey data, sometimes fixed monitoring stations and often a geographically stratified random sampling design, we introduced a tailored spatio-temporal version of the nearest-neighbour Gaussian process. Embedding this spatio-temporal nearest-neighbour Gaussian process in a generalized linear mixed model framework allows us to give quick inference and accurate predictions of spatio-temporal species distributions from presence/absence data, count data, and weight data.

With future research in mind we formulated the staRVe model to be easily extensible. After developing the staRVe model for univariate data we introduced cross-sectional copulas to be able to analyze multiple data streams in the same model and to infer dependence between variables. While the development and study of the multivariate extension are still in their early stages, the simulations shown in Chapter 4 are promising. The simulations show that the dependence between variables can be accurately estimated, and that data-rich variables can be used to improve predictions of data-poor variables. However, the simulations shown were limited in scope. Scenarios that need to be studied before declaring the copula approach a success include fully spatio-temporal data, data where observations of the different variables are never or only rarely co-located, more than two variables being analyzed, and classes of copulas other than the Gaussian copula.

Multivariate modelling is also closely linked with spatial confounding, the phenomenon where regression coefficient estimates are affected when the covariate is spatially structured and correlated with the spatial pattern of the response variable [Paciorek, 2010]. A few solutions to the problem of spatial confounding have been published recently, however some so-called solutions such as restricted spatial regression have been shown

to give poor inference [Khan and Calder, 2020] and other solutions based on adjusting a least squares or maximum likelihood estimator are either computationally intractable for modern datasets or are not directly applicable to spatial random effects embedded as part of a larger model [Chiou et al., 2019]. Other promising methods for accounting for spatial confounding first explicitly model the covariate as a spatial process and then feed the (hopefully) spatially-independent residuals into the regression equation in place of the original covariate [Thaden and Kneib, 2018, Dupont et al., 2022]. Our copula approach for multivariate data is very similar to these latter methods, as the dependence between variables is modelled on the spatially-independent quantiles of the random effects. With the amount of attention spatial confounding has been given in the contemporary literature, the possibility of using copulas to alleviate such problems should be explored.

In addition to further testing the capabilities of the copula approach for multivariate data, other properties of the staRVe model should be investigated. This is especially important when comparing staRVe to some of its immediate competitors such as R-INLA [Lindgren and Rue, 2015] and VAST [Thorson and Barnett, 2017], both of which have been established in the literature long enough to have accepted best practices. The main choices that a user of the staRVe model and package have to make relate to the graph used for the nearest-neighbour Gaussian process. The original paper introducing the nearest-neighbour Gaussian process suggests that inference using a nearest-neighbour Gaussian process with $k \approx 15$ nearest neighbours is effectively as good as inference using a normal Gaussian. But the number of neighbours is not the only choice to make, and computational efficiency will be improved by decreasing the number of neighbours used. Computational gains made by decreasing the number of neighbours used could allow more locations to be used in the persistent graph and thus allow for better spatial resolution in the model. Thus a study investigating the benefits, disadvantages, and tradeoffs of the number of neighbours and the choice and density of locations in the persistent graph is important in understanding how to best use the staRVe model.

In tandem with deciding how best to choose the set of locations for the persistent graph for a given dataset, determining optimal study design for scientific surveys will

be useful for ecologists regardless of which particular model is used to analyze the data. Particularly in the context of fisheries science, the accepted best practice for study design is the geographically stratified random sampling of locations within each year that was used for the haddock example we analyzed, where the locations sampled are different from year to year. Another study design is the use of fixed monitoring stations where the same locations are sampled every year. A simulation study used to determine how these sampling designs affect inference of model parameters and predictions of spatio-temporal dynamics will provide useful guidance to ecologists and fisheries managers on how to collect data as effectively as possible. Because of the limitations of the fixed monitoring station design, geographically stratified random sampling is better able to detect shifts in the spatial distribution of a species. However, because the locations sampled change from year to year it makes intuitive sense that a fixed station design will give better estimates of parameters related to temporal variability and thus improved forecasts of the species distribution into future years. Given the monetary (and sometimes ecological) cost of collecting data for scientific surveys in fisheries, ecologists should be guided by an understanding the relative merits of a fixed station design, a geographically stratified design, or possibly a mixed design with some fixed monitoring stations and some additional stratified random sampling.

The final avenue of future research we will mention here is creating a fisheries-specific version of the staRVe model for use in stock assessment. Historically stock assessment models have been limited to purely temporal models [Aeberhard et al., 2018], although recently there has been increased attention paid to extend assessment models to include spatial information, see for example [Cadigan et al., 2017]. Some of the hallmarks of a fisheries stock assessment model include the abilities to model multiple age classes (including ageing of a cohort from one year to the next, and the number of recruits introduced into the population each year), to incorporate external information to help estimate natural mortality, and to simultaneously model data from commercial catch and from scientific surveys. Incorporating both commercial catch and scientific surveys is already possible through the copula approach for multivariate analysis. Modelling the ageing and recruits will require adaptation of the equations used in the temporal setting to be applicable in the spatio-temporal setting. Such adaptation may

be as simple as having each location in the persistent graph follow the same ageing and recruit equations typically used in temporal models, but likely will require more sophisticated approaches to account for dispersal of fish within each age class between years. There are likely many examples in addition to ageing that will complicate the transition from temporal stock assessment models to spatio-temporal stock assessment models. However, there is clear recognition in the scientific community that spatial structure is vitally important in assessing the health of a fish population and so this transition should be made as soon as possible.

Bibliography

[Aeberhard et al., 2018] Aeberhard, W. H., Mills Flemming, J., and Nielsen, A. (2018). Review of state-space models for fisheries science. *Annual Review of Statistics and Its Application*, 5(1):215–235.

[Ailliot, 2006] Ailliot, P. (2006). Some theoretical results on Markov-switching autoregressive models with gamma innovations. *Comptes Rendus Mathematique*, 343(4):271–274.

[Anderson et al., 2019] Anderson, M. J., de Valpine, P., Punnett, A., and Miller, A. E. (2019). A pathway for multivariate analysis of ecological communities using copulas. *Ecology and Evolution*, 9(6):3276–3294.

[Babyn et al., 2021] Babyn, J., Varkey, D., Regular, P., Ings, D., and Mills Flemming, J. (2021). A Gaussian field approach to generating spatial age length keys. *Fisheries Research*, 240:105956.

[Banerjee et al., 2008] Banerjee, S., Gelfand, A. E., Finley, A. O., and Sang, H. (2008). Gaussian predictive process models for large spatial data sets. *Journal of the Royal Statistical Society B (Statistical Methodology)*, 70(4):825–848.

[Berger et al., 2017] Berger, A. M., Goethel, D. R., Lynch, P. D., Quinn, T. J., Mormede, S., McKenzie, J., and Dunn, A. (2017). Space oddity: The mission for spatial integration. *Canadian Journal of Fisheries and Aquatic Sciences*, (ja).

[Berliner, 1996] Berliner, L. M. (1996). Hierarchical bayesian time series models. In *Maximum Entropy and Bayesian Methods*, pages 15–22, Dordrecht. Springer Netherlands.

[Breed et al., 2012] Breed, G. A., Costa, D. P., Jonsen, I. D., Robinson, P. W., and Mills Flemming, J. (2012). State-space methods for more completely capturing behavioral dynamics from animal tracks. *Ecological Modelling*, 235-236:49 – 58.

[Cadigan et al., 2017] Cadigan, N. G., Wade, E., and Nielsen, A. (2017). A spatiotemporal model for snow crab (Chionoecetes opilio) stock size in the southern Gulf of St. Lawrence. *Canadian Journal of Fisheries and Aquatic Sciences*, (999):1–13.

[Chiou et al., 2019] Chiou, Y.-H., Yang, H.-D., and Chen, C.-S. (2019). An adjusted parameter estimation for spatial regression with spatial confounding. *Stochastic Environmental Research and Risk Assessment*, 33(8):1535–1551.

[Cressie, 1990] Cressie, N. (1990). The origins of kriging. *Mathematical Geology*, 22(3):239–252.

[Cressie and Johannesson, 2008] Cressie, N. and Johannesson, G. (2008). Fixed rank kriging for very large spatial data sets. *Journal of the Royal Statistical Society B (Statistical Methodology)*, 70(1):209–226.

[Curry, 2014] Curry, D. M. (2014). An algorithm for clustering animals by species based upon daily movement. *Procedia Computer Science*, 36:629 – 636.

[Datta et al., 2016] Datta, A., Banerjee, S., Finley, A. O., and Gelfand, A. E. (2016). Hierarchical nearest-neighbor Gaussian process models for large geostatistical datasets. *Journal of the American Statistical Association*, 111(514):800–812.

[Dawid, 1984] Dawid, A. P. (1984). Present position and potential developments: Some personal views: Statistical theory: The prequential approach. *Journal of the Royal Statistical Society. Series A (General)*, 147(2):278–292.

[Delhomme, 1979] Delhomme, J. P. (1979). Etude géostatistique de la géométrie du réservoir de Chemery à partir des données sismiques et de sondage. Technical report, Ecole des Mines de Paris, Fontainebleau, France.

[Diggle et al., 1998] Diggle, P. J., Tawn, J. A., and Moyeed, R. A. (1998). Model-based geostatistics. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 47(3):299–350.

[Douc et al., 2004] Douc, R., Moulines, E., and Rydén, T. (2004). Asymptotic properties of the maximum likelihood estimator in autoregressive models with Markov regime. *The Annals of statistics*, 32(5):2254–2304.

[Dupont et al., 2022] Dupont, E., Wood, S. N., and Augustin, N. H. (2022). Spatial+: A novel approach to spatial confounding. *Biometrics*, n/a(n/a).

[Eynon and Switzer, 1983] Eynon, B. P. and Switzer, P. (1983). The variability of rainfall acidity. *Can J Statistics*, 11(1):11–23.

[Forester et al., 2007] Forester, J. D., Ives, A. R., Turner, M. G., Anderson, D. P., Fortin, D., Beyer, H. L., Smith, D. W., and Boyce, M. S. (2007). State-space models link elk movement patterns to landscape characteristics in Yellowstone National Park. *Ecological Monographs*, 77(2):285–299.

[Foster and Bravington, 2013] Foster, S. D. and Bravington, M. V. (2013). A poisson-gamma model for analysis of ecological non-negative continuous data. *Environmental and Ecological Statistics*, 20(4):533–552.

[Giorgi et al., 2018] Giorgi, E., Diggle, P. J., Snow, R. W., and Noor, A. M. (2018). Geostatistical methods for disease mapping and visualisation using data from spatio-temporally referenced prevalence surveys. *International Statistical Review*, 86(3):571–597.

[Glennie et al., 2022] Glennie, R., Adam, T., Leos-Barajas, V., Michelot, T., Photopoulou, T., and McClintock, B. T. (2022). Hidden Markov models: Pitfalls and opportunities in ecology. *Methods in Ecology and Evolution*, n/a(n/a).

[Goodall and Mardia, 1994] Goodall, C. and Mardia, K. V. (1994). Challenges in multivariate spatio-temporal modeling. In *Proceedings of XVIIth International Biometrics Conference*, volume 1, pages 1–17, Hamilton, Ontario.

[Hartig, 2020] Hartig, F. (2020). *DHARMa: Residual Diagnostics for Hierarchical (Multi-Level / Mixed) Regression Models*. R package version 0.3.3.0.

[Heaton et al., 2018] Heaton, M. J., Datta, A., Finley, A. O., Furrer, R., Guinness, J., Guhaniyogi, R., Gerber, F., Gramacy, R. B., Hammerling, D., Katzfuss, M., Lindgren, F., Nychka, D. W., Sun, F., and Zammit-Mangion, A. (2018). A case study competition among methods for analyzing large spatial data. *Journal of Agricultural, Biological and Environmental Statistics*, page 1.

[Hijmans, 2020] Hijmans, R. J. (2020). *raster: Geographic Data Analysis and Modeling*. R package version 3.0-12.

[Hooten et al., 2017a] Hooten, M., Johnson, D., McClintock, B., and Morales, J. (2017a). *Animal Movement: Statistical Models for Telemetry Data*. CRC Press.

[Hooten et al., 2017b] Hooten, M. B., King, R., and Langrock, R. (2017b). Guest editor's introduction to the special issue on "animal movement modeling". *Journal of Agricultural, Biological and Environmental Statistics*, 22(3):224–231.

[Huijbregts and Matheron, 1971] Huijbregts, C. and Matheron, G. (1971). Universal kriging-an optimal approach to trend surface analysis. *Decision Making in the Mineral Industry*, 12:159–169.

[Hussey et al., 2015] Hussey, N. E., Kessel, S. T., Aarestrup, K., Cooke, S. J., Cowley, P. D., Fisk, A. T., Harcourt, R. G., Holland, K. N., Iverson, S. J., Kocik, J. F., Mills Flemming, J. E., and Whoriskey, F. G. (2015). Aquatic animal telemetry: A panoramic window into the underwater world. *Science*, 348(6240):1255642.

[Jonsen et al., 2005] Jonsen, I. D., Mills Flemming, J., and Myers, R. A. (2005). Robust state-space modeling of animal movement data. *Ecology*, 86(11):2874–2880.

[Journel and Huijbregts, 1978] Journel, A. G. and Huijbregts, C. J. (1978). *Mining Geostatistics*. Academic Press.

[Jubinville et al., 2021] Jubinville, I., Lawler, E., Tattrie, S., Shackell, N. L., Mills Flemming, J., and Worm, B. (2021). Distributions of threatened skates and commercial fisheries inform conservation hotspots. *Mar Ecol Prog Ser*, 679:1–18.

[Kalman, 1960] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45.

[Kaufman and Shaby, 2013] Kaufman, C. G. and Shaby, B. A. (2013). The role of the range parameter for estimation and prediction in geostatistics. *Biometrika*, 100(2):473–484.

[Khan and Calder, 2020] Khan, K. and Calder, C. A. (2020). Restricted spatial regression methods: Implications for inference. *Journal of the American Statistical Association*, 0(0):1–34.

[Koller and Friedman, 2009] Koller, D. and Friedman, N. (2009). *Probabilistic Graphical Models: Principles and Techniques*. Adaptive Computation and Machine Learning. MIT Press.

[Kristensen et al., 2016] Kristensen, K., Nielsen, A., Berg, C. W., Skaug, H. J., and Bell, B. M. (2016). TMB: Automatic differentiation and Laplace approximation. *Journal of Statistical Software*, 70(5):1–21.

[Kristensen et al., 2014] Kristensen, K., Thygesen, U. H., Andersen, K. H., and Beyer, J. E. (2014). Estimating spatio-temporal dynamics of size-structured populations. *Can. J. Fish. Aquat. Sci.*, 71(2):326–336.

[Langrock et al., 2012] Langrock, R., King, R., Matthiopoulos, J., Thomas, L., Fortin, D., and Morales, J. M. (2012). Flexible and practical modeling of animal telemetry data: Hidden Markov models and extensions. *Ecology*, 93(11):2336–2342.

[Lawler et al., 2019] Lawler, E., Whoriskey, K., Aeberhard, W. H., Field, C., and Mills Flemming, J. (2019). The conditionally autoregressive hidden Markov model (carhmm): Inferring behavioural states from animal tracking data exhibiting conditional autocorrelation. *Journal of Agricultural, Biological and Environmental Statistics*, 24(4):651–668.

[Leos-Barajas et al., 2017] Leos-Barajas, V., Gangloff, E. J., Adam, T., Langrock, R., van Beest, F. M., Nabe-Nielsen, J., and Morales, J. M. (2017). Multi-scale modeling of animal movement and general behavior data using hidden Markov models with hierarchical structures. *Journal of Agricultural, Biological and Environmental Statistics*, pages 1–17.

[Liebscher et al., 2022] Liebscher, E., Taubert, F., Waltschew, D., and Hetzer, J. (2022). Modelling multivariate data using product copulas and minimum distance estimators: an exemplary application to ecological traits. *Environmental and Ecological Statistics*.

[Lindgren and Rue, 2015] Lindgren, F. and Rue, H. (2015). Bayesian spatial modelling with R-INLA. *Journal of Statistical Software, Articles*, 63(19):1–25.

[Lindgren et al., 2011] Lindgren, F., Rue, H., and Lindström, J. (2011). An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach. *Journal of the Royal Statistical Society B (Statistical Methodology)*, 73(4):423–498.

[Lindström et al., 2013] Lindström, J., Szpiro, A., Sampson, P. D., Bergen, S., and Sheppard, L. (2013). Spatiotemporal: An r package for spatio-temporal modelling of air pollution. R package.

[Mardia and Goodall, 1993] Mardia, K. V. and Goodall, C. (1993). *Multivariate Environmental Statistics*, chapter Spatial-Temporal Analyses of Multivariate Environmental Monitoring Data, pages 347–386. North-Holland Series in Statistics and Probability. Elsevier Science.

[Mardia and Marshall, 1984] Mardia, K. V. and Marshall, R. J. (1984). Maximum likelihood estimation of models for residual covariance in spatial regression. *Biometrika*, 71(1):135–146.

[Mardia and Watkins, 1989] Mardia, K. V. and Watkins, A. J. (1989). On multimodality of the likelihood in the spatial linear model. *Biometrika*, 76(2):289–295.

[Martin and Oeppen, 1975] Martin, R. L. and Oeppen, J. E. (1975). The identification of regional forecasting models using space-time correlation functions. *Transactions of the Institute of British Geographers*, (66):95–118.

[Matérn, 1960] Matérn, B. (1960). *Spatial Variation: Stochastic Models and Their Application to Some Problems in Forest Surveys and Other Sampling Investigations*. Meddelanden från Statens Skogsforskningsinstitut. Centraltryckeriet, Esselte.

[Matheron, 1963] Matheron, G. (1963). Principles of geostatistics. *Economic Geology*, 58(8):1246.

[Matheron, 1969] Matheron, G. (1969). *Le Krigeage Universel*, volume 1. École Nationale Supérieure des Mines de Paris.

[Matheron, 1973] Matheron, G. (1973). The intrinsic random functions and their applications. *Advances in Applied Probability*, 5(3):439–468.

[Matheron, 1979] Matheron, G. (1979). *Recherche de Simplication dans un Probieme de Cokrigeage*, volume N-698. Centre de Geostatistique, Fountainebleu, France.

[McClintock, 2017] McClintock, B. T. (2017). Incorporating telemetry error into hidden Markov models of animal movement using multiple imputation. *Journal of Agricultural, Biological and Environmental Statistics*, 22(3):249–269.

[McClintock et al., 2012] McClintock, B. T., King, R., Thomas, L., Matthiopoulos, J., McConnell, B. J., and Morales, J. M. (2012). A general discrete-time modeling framework for animal movement using multistate random walks. *Ecological Monographs*, 82(3):335–349.

[McClintock and Michelot, 2018] McClintock, B. T. and Michelot, T. (2018). momentuHMM: R package for generalized hidden Markov models of animal movement. *Methods in Ecology and Evolution*, 9(6):1518–1530.

[Michelot, 2019] Michelot, T. (2019). Stochastic models of animal movement and habitat selection.

[Michelot et al., 2016] Michelot, T., Langrock, R., and Patterson, T. A. (2016). moveHMM: An R package for the statistical modelling of animal movement data using hidden Markov models. *Methods in Ecology and Evolution*, 7(11):1308–1315.

[Mills Flemming and Field, 2014] Mills Flemming, J. and Field, C. A. (2014). *Statistics in Action: A Canadian Outlook*, chapter Challenges in Statistical Marine Ecology, pages 305–319. CRC Press.

[Morales et al., 2004] Morales, J. M., Haydon, D. T., Frair, J., Holsinger, K. E., and Fryxell, J. M. (2004). Extracting more out of relocation data: Building movement models as mixtures of random walks. *Ecology*, 85(9):2436–2445.

[Morales et al., 2010] Morales, J. M., Moorcroft, P. R., Matthiopoulos, J., Frair, J. L., Kie, J. G., Powell, R. A., Merrill, E. H., and Haydon, D. T. (2010). Building the bridge between animal movement and population dynamics. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 365(1550):2289–2301.

[Nychka et al., 2016] Nychka, D., Hammerling, D., Sain, S., and Lenssen, N. (2016). LatticeKrig: Multiresolution kriging based on Markov random fields. R package version 8.4.

[Ogle and Barber, 2020] Ogle, K. and Barber, J. J. (2020). Ensuring identifiability in hierarchical mixed effects Bayesian models. *Ecological Applications*, 30(7):e02159.

[Paciorek, 2010] Paciorek, C. J. (2010). The importance of scale for spatial-confounding bias and precision of spatial regression estimators. *Statistical science : a review journal of the Institute of Mathematical Statistics*, 25:107–125.

[Pardieck et al., 2017] Pardieck, K., Ziolkowski Jr., D., Lutmerding, M., Campbell, K., and Hudson, M. A. (2017). North American breeding bird survey dataset 1966 - 2016, version 2016.0. U.S. Geological Survey, Patuxent Wildlife Research Center.

[Patterson et al., 2017] Patterson, T. A., Parton, A., Langrock, R., Blackwell, P. G., Thomas, L., and King, R. (2017). Statistical modelling of individual animal

movement: An overview of key methods and a discussion of practical challenges. *AStA Advances in Statistical Analysis*, 101(4):399–438.

[Patton, 2006] Patton, A. J. (2006). Modelling asymmetric exchange rate dependence. *International Economic Review*, 47(2):527–556.

[Pebesma, 2018] Pebesma, E. (2018). Simple features for R: Standardized support for spatial vector data. *The R Journal*, 10(1):439–446.

[Philip and Watson, 1986] Philip, G. M. and Watson, D. F. (1986). Matheronian geostatistics–quo vadis? *Mathematical Geology*, 18(1):93–117.

[Pohle et al., 2017] Pohle, J., Langrock, R., van Beest, F. M., and Schmidt, N. M. (2017). Selecting the number of states in hidden Markov models: Pragmatic solutions illustrated using animal movement. *Journal of Agricultural, Biological and Environmental Statistics*, pages 1–24.

[Potts et al., 2014] Potts, J. R., Marie, A., Karl, M., and Lewis, M. A. (2014). A generalized residual technique for analysing complex movement models using earth mover's distance. *Methods in Ecology and Evolution*, 5(10):1012–1022.

[Rosenblatt, 1952] Rosenblatt, M. (1952). Remarks on a multivariate transformation. *The Annals of Mathematical Statistics*, 23(3):470–472.

[Salvaña and Genton, 2020] Salvaña, M. L. O. and Genton, M. G. (2020). Nonstationary cross-covariance functions for multivariate spatio-temporal random fields. *Frontiers in Spatial and Spatio-temporal Research*, 37:100411.

[Sellers et al., 2012] Sellers, K. F., Borle, S., and Shmueli, G. (2012). The COM-poisson model for count data: a survey of methods and applications. *Appl. Stochastic Models Bus. Ind.*, 28(2):104–116.

[Shepard et al., 2013] Shepard, E. L. C., Wilson, R. P., Rees, W. G., Grundy, E., Lambertucci, S. A., and Vosper, S. B. (2013). Energy landscapes shape animal movement ecology. *The American Naturalist*, 182(3):298–312. PMID: 23933722.

[Shepherd et al., 2016] Shepherd, B. E., Li, C., and Liu, Q. (2016). Probability-scale residuals for continuous, discrete, and censored data. *Canadian Journal of Statistics*, 44(4):463–479.

[Smith, 1999] Smith, S. J. (1999). Survey methods for fisheries: Challenges and statistical issues. *Proceedings of the Statistical Society of Canada Annual Meeting, Survey Methods Section*.

[Tennekes, 2018] Tennekes, M. (2018). tmap: Thematic maps in R. *Journal of Statistical Software*, 84(6):1–39.

[Thaden and Kneib, 2018] Thaden, H. and Kneib, T. (2018). Structural equation models for dealing with spatial confounding. *The American Statistician*, 72(3):239–252.

[Thorson and Barnett, 2017] Thorson, J. T. and Barnett, L. A. K. (2017). Comparing estimates of abundance trends and distribution shifts using single- and multispecies models of fishes and biogenic habitat. *ICES Journal of Marine Science*, 74(5):1311–1321.

[Tikhonov et al., 2020] Tikhonov, G., Opedal, Ø. H., Abrego, N., Lehikoinen, A., de Jonge, M. M. J., Oksanen, J., and Ovaskainen, O. (2020). Joint species distribution modelling with the R package Hmsc. *Methods in Ecology and Evolution*, 11(3):442–447.

[Tobler, 1970] Tobler, W. R. (1970). A computer movie simulating urban growth in the Detroit region. *Economic Geography*, 46:234–240.

[Wackernagel, 1985] Wackernagel, H. (1985). *L'Inférence d'un Modèle Linéaire en Géostatistique Multivariable*. PhD thesis, Paris School of Mines.

[Wackernagel, 1988] Wackernagel, H. (1988). *Quantitative Analysis of Mineral Energy Resources*, chapter Geostatistical Techniques for Interpreting Multivariate Spatial Information, pages 393–409. Reidel, Dordrecht.

[Warnes and Ripley, 1987] Warnes, J. J. and Ripley, B. D. (1987). Problems with likelihood estimation of covariance functions of spatial Gaussian processes. *Biometrika*, 74(3):640–642.

[Whoriskey et al., 2017] Whoriskey, K., Auger-Méthé, M., Albertsen, C. M., Whoriskey, F. G., Binder, T. R., Krueger, C. C., and Mills Flemming, J. (2017). A hidden Markov movement model for rapidly identifying behavioral states from animal tracks. *Ecology and Evolution*, 7(7):2112–2121.

[Wikle, 2015] Wikle, C. K. (2015). Modern perspectives on statistics for spatio-temporal data. *WIREs Comput Stat*, 7(1):86–98.

[Wikle et al., 1998] Wikle, C. K., Berliner, L. M., and Cressie, N. (1998). Hierarchical Bayesian space-time models. *Environmental and Ecological Statistics*, 5(2):117–154.

[Zammit-Mangion, 2020] Zammit-Mangion, A. (2020). *STRbook: Supplementary Package for Book on ST Modelling with R*. R package version 0.1.0.

[Zammit-Mangion et al., 2015] Zammit-Mangion, A., Rougier, J., Schön, N., Lindgren, F., and Bamber, J. (2015). Multivariate spatio-temporal modelling for assessing Antarctica's present-day contribution to sea-level rise. *Environmetrics*, 26(3):159–177.

[Zhang, 2004] Zhang, H. (2004). Inconsistent estimation and asymptotically equal interpolations in model-based geostatistics. *Journal of the American Statistical Association*, 99(465):250–261.

[Zheng and Cadigan, 2021] Zheng, N. and Cadigan, N. (2021). Frequentist delta-variance approximations with mixed-effects models and tmb. *Computational Statistics & Data Analysis*, 160:107227.

[Zucchini et al., 2016] Zucchini, W., MacDonald, I. L., and Langrock, R. (2016). *Hidden Markov Models for Time Series: An Introduction Using R*, volume 150. CRC press.

Appendix A

Technical Details for the staRVe Model

A.1 More Details on the NNGP Log-Likelihood

Here we detail the computation of the log-likelihood contribution $\ell_w(\Theta)$ for the spatio-temporal nearest-neighbour Gaussian process (NNGP), in particular the "spatial" level of the hierarchy. We note that there are modelling choices that could have been made differently from the ones we have implemented and describe below. We made our choices based on a combination of model simplicity, computational efficiency, and personal preference. The original paper introducing the NNGP in a spatial context is [Datta et al., 2016], and to learn more about the broader class of probabilistic graphical models we suggest [Koller and Friedman, 2009]. We provide a visual introduction to a spatial NNGP in Figure A.1.

A Gaussian process is a (possibly infinite) collection of random variables, where any finite subset of that collection has a multivariate normal distribution. In the spatial context, a Gaussian process is specified completely by a mean function and a covariance function. Throughout this section we speak of the locations and the random effects at those locations interchangeably. For inference we look at the joint distribution of the observed locations and any additional locations we want to include.

We start by defining the nearest neighbour Gaussian process in the purely spatial case. Evaluating the joint density of a regular Gaussian process for even a moderately sized number of locations is computationally challenging because it requires computing the inverse of the covariance matrix. Using basic rules of conditional probabilities, we can factor the joint density as a product of (univariate) conditional densities:

$$f\left(\mathbf{W}\left(\mathbf{s}_{1:N}\right)\right) = f_1\left(W\left(\mathbf{s}_1\right)\right) \cdot \prod_{i=1}^{N-1} f_{i+1}\left(W\left(\mathbf{s}_{i+1}\right) \mid \mathbf{W}\left(\mathbf{s}_{1:i}\right)\right) \tag{A.1}$$

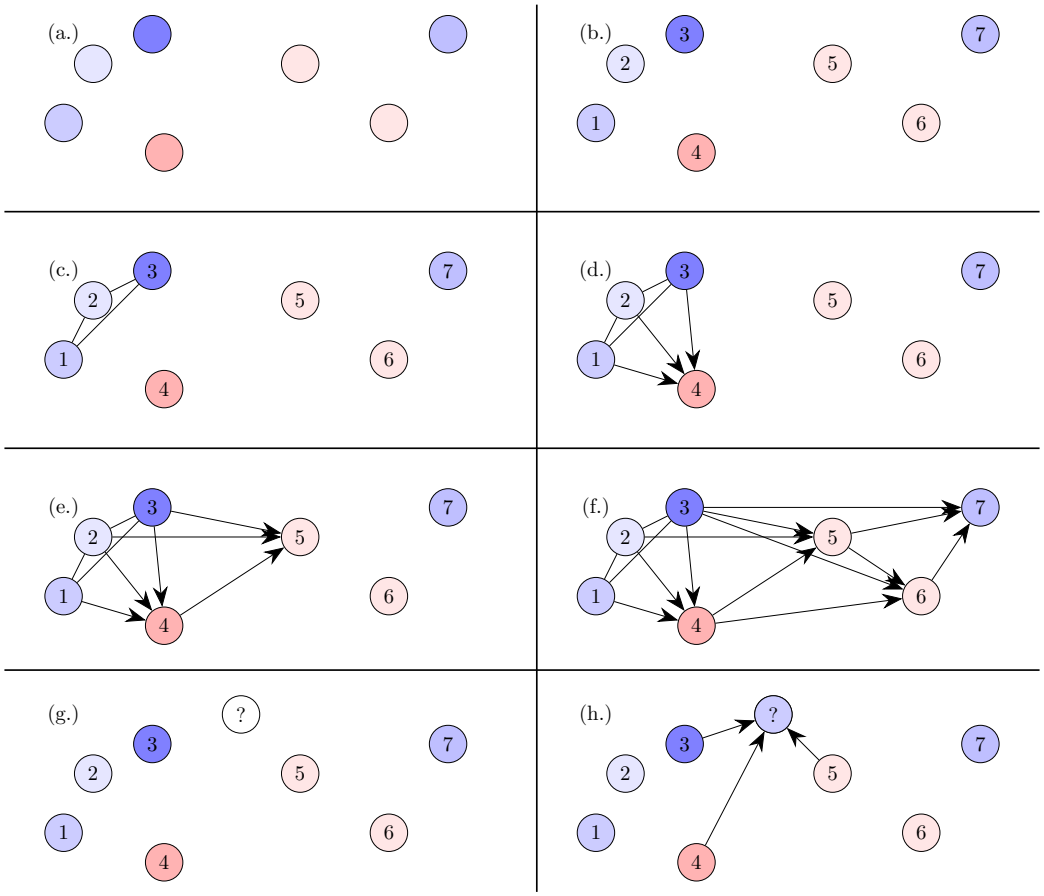

Figure A.1: Graphical model representation of the steps for computing the likelihood of a nearest neighbour Gaussian process using $k = 3$ nearest neighbours. **(a.)** There are 7 observed locations, with the value of each observation given by the colour of each circle. **(b.)** Order the observed locations. Observations that are close in space should be close together in the ordering. In this example we put the locations in increasing order from left to right. **(c.)** Find the joint multivariate normal distribution for the first $k = 3$ locations implied by the mean function and covariance function. **(d.)** Compute the conditional distribution of the 4th location given its $k = 3$ nearest neighbours that came before it in the ordering, in this case locations 1, 2, and 3. **(e.)** Compute the conditional distribution of the 5th location given its $k = 3$ nearest neighbours that came before it in the ordering, in this case locations 2, 3, and 4. **(f.)** Continue with the conditional distributions for each of the remaining locations: in this example the conditional distribution of location 6 given locations 3, 4, and 5; and the conditional distribution of location 7 given locations 3, 5, and 6. **(g.)** We want to predict the value of the spatial field at a new unobserved location. **(h.)** The predictive distribution of the new location is the conditional distribution of that location given the $k = 3$ nearest observed locations, regardless of where those locations fall in the ordering, in this case locations 3, 4, and 5.

where we write $\mathbf{W}(\mathbf{s}_{1:i})$ to mean random variables at the first i locations of the multivariate normal variable. This isn't immediately useful, as the term $f_N\left(W(\mathbf{s}_N) \mid \mathbf{W}\left(\mathbf{s}_{1:(N-1)}\right)\right)$ is almost as computationally expensive to compute as the full joint density.

However, the nearest neighbour Gaussian process imposes a set of conditional independence constraints. These constraints allows us to replace the conditioning set $\mathbf{W}(\mathbf{s}_{1:i})$ with a much smaller one, which drastically reduces the computational burden of computing the densities. To describe the conditional independence constraints we introduce a set of reference locations $\mathcal{S}_{per}$ and give them an order. In our implementation the first location in the order is arbitrary, and subsequent locations in the order are taken to be whichever as yet unordered location is closest to any location that is already ordered. A location $\mathbf{s}$ in $\mathcal{S}_{per}$ is conditionally independent of all other locations, given the k geographically closest locations in $\mathcal{S}_{per}$ that came before it in the order. A location $\mathbf{s}$ not in $\mathcal{S}_{per}$ is conditionally independent of all other locations, given the k geographically closest locations in $\mathcal{S}_{per}$ regardless of the order. We call these "k closest locations" the parents of $\mathbf{s}$, or $\delta\mathbf{s}$ for short, and choose k to be some small number e.g., $k = 15$.

Given a set of observed locations and user-chosen $\mathcal{S}_{per}$ we encode the above conditional independence constraints in a directed acyclic graph (DAG) named $\mathcal{G}$. The vertices of the graph are the set of all locations considered. There is a directed edge from vertex i to vertex j iff $\mathbf{s}_i$ is a parent of $\mathbf{s}_j$.

With these conditional independence constraints, the joint density of locations $\mathbf{s}_{1:N}$ and reference locations $\mathbf{S}_{per}$ is

$$f\left(\mathbf{W}(\mathbf{s}_{1:N}),\ \mathbf{W}(\mathcal{S}_{per})\right) = \prod_{\mathbf{s}\in\mathcal{S}_{per}} f_{\mathbf{s}}\left(W(\mathbf{s}) \mid \mathbf{W}(\delta\mathbf{s})\right) \cdot \prod_{i=1}^{N} f_i\left(W(\mathbf{s}_i) \mid \mathbf{W}(\delta\mathbf{s}_i)\right). \quad \text{(A.2)}$$

Thus $\mathcal{G}$ also describes the series of conditional densities used to compute the joint density.

We generalise the above to our spatio-temporal setting by creating a graph $\mathcal{G}_t$ for each time point t which is split in two parts: the persistent graph $\mathcal{G}_{per}$ corresponding to

the reference locations $\mathcal{S}_{per}$, and the transient graphs $\mathcal{G}_{trans,t}$ encoding the parents $\delta\mathbf{s}_i$ of the observed locations at time point t. Since the reference locations do not change from year to year, the persistent graph $\mathcal{G}_{per}$ and thus the conditional independence constraints and density factorization are the same from year to year. Combined with property that Gaussian conditional variances do not depend on the realised value of the random variables conditioned upon, this allows us to cache most of the computations used in computing the likelihood for the persistent graph. The transient graphs $\mathcal{G}_{trans,t}$ can change from year to year since different locations may be observed, but a location $\mathbf{s}$ will have the same parents every year.

We split the computation of $\ell_w(\Theta)$ into contributions $\ell_{per}(\Theta)$ from the reference locations and $\ell_{tran}(\Theta)$ from the observed locations.

A.1.1 Persistent Graph Log-Likelihood

Here we describe the conditional distributions used for the reference locations and persistent graph. We recall our notation that $W_t(\mathbf{s})$ is the random effect at time point t and location $\mathbf{s}$. We also use $\mathbf{W}_t(\delta\mathbf{s})$ to be joint random effects at the locations in $\delta\mathbf{s}$. As a reminder the "temporal" level in the model hierarchy defines a set of purely temporal random effects defined by

$$\boldsymbol{\epsilon} \sim \mathrm{N}\left(\mu,\ \frac{\sigma^2}{1-\phi^2}\boldsymbol{\Sigma}\right) \quad \text{where} \quad \Sigma_{ij} = \phi^{||i-j||}. \tag{A.3}$$

An equivalent formulation that will be useful later is

$$\begin{aligned} \epsilon_t\ |\epsilon_{t-1} &= \phi\cdot\epsilon_{t-1} + (1-\phi)\,\mu + \nu_t && (\nu_t \sim N(0,\sigma^2)) \\ \epsilon_1 &= \mu + \nu_1 && \left(\nu_1 \sim N\left(0, \tfrac{\sigma^2}{1-\phi^2}\right)\right) \end{aligned}$$

For presentation purposes we use the exponential covariance function

$$\mathcal{C}(\mathbf{s}_1,\mathbf{s}_2) = \tau^2\rho\ \mathrm{Exp}\left[-\frac{||\mathbf{s}_1-\mathbf{s}_2||}{\bar{d}}\cdot\frac{1}{\rho}\right], \tag{A.4}$$

but any Matérn covariance function can be used.

At the initial time point $t = 1$ we assume a constant marginal mean equal to ϵ_1. The conditional distribution for $W_1(\mathbf{s}_i)$ is

$$W_1(\mathbf{s}_i) \;\Big|\; \mathbf{w}_1(\delta\mathbf{s}_i),\ \boldsymbol{\epsilon} \sim N\left(\epsilon_1 + \mathbf{c}^\top \boldsymbol{\Sigma}^{-1}\left(\mathbf{w}_1(\delta\mathbf{s}_i) - \epsilon_1\right),\ \tau^2\rho - \mathbf{c}^\top\boldsymbol{\Sigma}^{-1}\mathbf{c}\right) \tag{A.5}$$

where $\delta\mathbf{s}_i$ is the set of parents of $\mathbf{s}_i$ in $\mathcal{G}_{per}$, $\mathbf{c}^\top$ is the cross-covariance vector between $W_1(\mathbf{s}_i)$ and $\mathbf{W}_1(\delta\mathbf{s}_i)$, $\boldsymbol{\Sigma}$ is the covariance matrix of $\mathbf{W}_1(\delta\mathbf{s}_i)$, and $\tau^2\rho$ is the marginal spatial variance. In the spatial statistics literature, the above conditional mean and variance are known as the kriging mean and variance. The likelihood for the first k random effects in the persistent graph is computed through their joint distribution

$$\mathbf{W}_1(\mathbf{s}_{1:k}) \sim N(\epsilon_1,\ \boldsymbol{\Sigma}), \tag{A.6}$$

where here $\boldsymbol{\Sigma}$ is the covariance matrix of $\mathbf{W}_1(\mathbf{s}_{1:k})$.

For times $t > 1$ we define the spatial marginal mean (but conditioned on previous times) of $W_t(\mathbf{s})$ to be $m_t(\mathbf{s}) = \phi \cdot (w_{t-1}(\mathbf{s}) - \epsilon_{t-1}) + \epsilon_t$. This mean function gives a marginal AR(1) process to each location, see Section A.3 Using this mean function $m_t(\mathbf{s}) = \phi \cdot (w_{t-1}(\mathbf{s}) - \epsilon_{t-1}) + \epsilon_t$, the conditional distribution for $W_t(\mathbf{s}_i)$ is then

$$W_t(\mathbf{s}_i) \;\Big|\; \mathbf{w}_{t-1}(\delta\mathbf{s}_i),\ \boldsymbol{\epsilon} \sim N\left(m_t(\mathbf{s}_i) + \mathbf{c}^\top\boldsymbol{\Sigma}^{-1}\left(\mathbf{w}_t(\delta\mathbf{s}_i) - \mathbf{m}_t(\delta\mathbf{s}_i)\right),\ \tau^2\rho - \mathbf{c}^\top\boldsymbol{\Sigma}^{-1}\mathbf{c}\right). \tag{A.7}$$

As in the case for $t = 0$, the likelihood for the first k random effects in the persistent graph is computed through their joint distribution

$$\mathbf{W}_t(\mathbf{s}_{1:k}) \sim N(\phi \cdot (\mathbf{w}_{t-1}(\mathbf{s}_{1:k}) - \epsilon_{t-1}) + \epsilon_t,\ \boldsymbol{\Sigma}). \tag{A.8}$$

There is a subtle identifiability issue related to the first k random effects in the persistent graph of each year, which we solve in an admittedly strange-looking fashion. The identifiability issue arises because the spatial range parameter ρ is unidentifiable and contributes to the marginal spatial variance $\tau^2\rho$. Both the marginal spatial variance and the temporal variance (by virtue of ϵ_t appearing in the mean of the spatial random effects) control the total variance of the first k random effects, so the

unidentifiability of ρ creeps in to cause the spatial variance to be unidentifiable. To deal with these crossed wires we temporarily use a different value of ρ for the first k random effects, choosing the value that sets the marginal variance equal to the average conditional spatial variance of the remaining random effects in the persistent graph. This forces the spatial component of the variance for the first k random effects to be similar in magnitude to the spatial component of the variance for the remaining random effects, and results in the temporal variance being identifiable.

A.1.2 Transient Node Log-Likelihood

The conditional distributions for the observation locations and transient graphs require only a small change from the distributions in Section A.1.1. Since a location $\mathbf{s}$ observed at time point t may not have been observed at time point $t-1$, we don't have a random effect $w_{t-1}(\mathbf{s})$ explicitly modelled to use in $m_t(\mathbf{s}) = \phi \cdot (w_{t-1}(\mathbf{s}) - \epsilon_{t-1}) + \epsilon_t$. We replace $w_{t-1}(\mathbf{s})$ with the predictor

$$\tilde{w}_{t-1}(\mathbf{s}) = \frac{\mathbf{c}^\top \boldsymbol{\Sigma}^{-1} \mathbf{w}_{t-1}(\delta\mathbf{s}_i)}{\mathbf{c}^\top \boldsymbol{\Sigma}^{-1} \mathbf{1}} \tag{A.9}$$

where $\mathbf{1}$ is a vector of ones. This prediction is the best linear unbiased predictor of the mean of the process $\mathbf{W}_{t-1}(\delta\mathbf{s})$ in an ordinary kriging framework [Cressie, 1990]. Since this is a weighted average of $\mathbf{w}_{t-1}(\delta\mathbf{s})$, it is a local estimator of the mean. The conditional distribution is then

$$W_t(\mathbf{s}) \;\Big|\; \mathbf{w}_t(\delta\mathbf{s}),\ \boldsymbol{\epsilon} \sim N\left(\tilde{m}_t(\mathbf{s}_i) + \mathbf{c}^\top \boldsymbol{\Sigma}^{-1}(\mathbf{w}_t(\delta\mathbf{s}) - \mathbf{m}_t(\delta\mathbf{s})),\ \sigma_s^2 - \mathbf{c}^\top \boldsymbol{\Sigma}^{-1} \mathbf{c}\right) \tag{A.10}$$

where $\tilde{m}_t(\mathbf{s}) = \phi \cdot (\tilde{w}_{t-1}(\mathbf{s}) - \epsilon_{t-1}) + \epsilon_t$.

A.2 More Details on Predictions

We expand on our description of our prediction procedure given in the main text. Our aim is to produce predictions for the random effect $W_t(\mathbf{s})$, the linear predictor on the link scale $L_t(\mathbf{s})$, and the response mean $\mu_t(\mathbf{s})$ at some unobserved spatial location $\mathbf{s}$ and time point t. Since there is uncertainty involved in the inference of the random effects and fixed parameters $\Theta = \{\theta,\ \boldsymbol{\beta},\ \phi,\ \mu,\ \sigma^2,\ \tau^2,\ \rho\}$ we also want to propagate this uncertainty to our prediction standard errors.

A.2.1 The Random Effects

The predictive distribution for the new random effect $W_t(\mathbf{s}_i)$ is the same as the conditional distributions we used for the observation locations in Section A.1.2, namely

$$W_t(\mathbf{s}) \ \Big|\ \{\mathbf{w}_t(\delta\mathbf{s}),\ \boldsymbol{\epsilon}\} \sim N\left(\tilde{m}_t(\mathbf{s}_i) + \mathbf{c}^\top\boldsymbol{\Sigma}^{-1}\left(\mathbf{w}_t(\delta\mathbf{s}) - \mathbf{m}_t(\delta\mathbf{s})\right),\ \sigma_s^2 - \mathbf{c}^\top\boldsymbol{\Sigma}^{-1}\mathbf{c}\right) \tag{A.11}$$

where $\tilde{m}_t(\mathbf{s}) = \phi \cdot (\tilde{w}_{t-1}(\mathbf{s}) - \epsilon_{t-1}) + \epsilon_t$. To the joint likelihood $\ell_J(\Theta)$ we add a likelihood contribution $\ell_p(\Theta)$ using the above conditional distribution. The predicted value for $W_t(\mathbf{s})$ is then taken to be the mode of $\ell_J\left(\hat{\Theta}\right) + \ell_p\left(\hat{\Theta}\right)$ as a function of $w_t(\mathbf{s})$, and the standard errors are computed from the hessian evaluated at the mode. This is essentially a plug-in estimate using the maximum likelihood estimates $\hat{\Theta}$, but since we take the predictions directly from the likelihood function all uncertainty in parameter estimation is propagated to the predictions.

A.2.2 The Linear Predictor

Having estimates and standard errors for $\hat{W}_t(\mathbf{s}_i)$ and $\hat{\boldsymbol{\beta}}$ and having a set of prediction covariates $\mathbf{X}$, we can easily find the predictions and standard errors for the linear predictor

$$\hat{L}_t(\mathbf{s}) = \mathbf{X}\hat{\boldsymbol{\beta}} + \hat{w}_t(\mathbf{s}_i) \tag{A.12}$$

$$\text{Var}\left[\hat{L}_t(\mathbf{s})\right] = \mathbf{X}\ \text{Var}\left[\hat{\boldsymbol{\beta}}\right]\mathbf{X}^\top +\ \text{Var}\left[\hat{w}_t(\mathbf{s}_i)\right] \tag{A.13}$$

In the variance calculation we are assuming the estimator $\hat{\boldsymbol{\beta}}$ is uncorrelated with $\hat{w}_t(\mathbf{s})$. Note that this assumption may not be true when the corresponding covariate

$\mathbf{X}$ is spatially or spatio-temporally structured [Paciorek, 2010, Chiou et al., 2019].

A.2.3 The Response Mean

Our last objective is to predict the response mean $\hat{\mu}_t(\mathbf{s}) = g^{-1}\left[\hat{L}_t(\mathbf{s})\right]$. Letting $h(x) = (g^{-1})''(x)$ be the second derivative of $g^{-1}(x)$, we apply a second-order Taylor approximation variance of $g^{-1}\left[\hat{L}_t(\mathbf{s})\right]$:

$$\mathrm{E}\left[g^{-1}\left(\hat{L}_t(\mathbf{s})\right)\right] \approx g^{-1}\left(\hat{L}_t(\mathbf{s})\right) \tag{A.14}$$

$$\mathrm{Var}\left[g^{-1}\left(\hat{L}_t(\mathbf{s})\right)\right] \approx g^{-1}\left(\hat{L}_t(\mathbf{s})\right)^2 \cdot \mathrm{Var}\left[\hat{L}_t(\mathbf{s})\right] + \frac{1}{2} \cdot h\left(\hat{L}_t(\mathbf{s})\right)^2 \cdot \mathrm{Var}\left[\hat{L}_t(\mathbf{s})\right]^2 \tag{A.15}$$

A.2.4 Prediction Intervals for New Observations

Keeping with accepted practice for generalized linear models we do not attempt to create prediction intervals for new observations. While computing closed form prediction intervals is relatively straightforward for a Gaussian response distribution such intervals either may not make sense, in the case of the Bernoulli distribution, or may not have a closed form, as in the case of a Poisson distribution. However observations are assumed conditionally independent of all other observations given the covariates and spatio-temporal random effects. The predicted response mean and estimated response distribution parameters are all that are needed to produce probability statements for any particular new observation, including prediction intervals, which could be found either from the density function of the response distribution or through simulations of new observations conditional on the fitted random effects.

A.3 Distributional Properties for the staRVe Model

To describe the distributional properties of the staRVe model it suffices to show the distributional properties of the spatio-temporal random effects. The properties of the temporal random effects are evident directly from the specification, and given the spatio-temporal random effects the conditional independence assumption for the observations makes their properties known from standard generalized linear mixed model theory. First we give the marginal forecast distribution of the spatial field $W_t(\mathbf{s}) \mid w_{t-1}(\mathbf{s})$ by marginalizing out the temporal random effects from the conditional distribution directly modelled $W_t(\mathbf{s}) \mid \{w_{t-1}(\mathbf{s}), \boldsymbol{\epsilon}\}$. Using these marginal forecast distributions, we then show that the spatio-temporal random effects have the same probabilistic model as a particular separable spatio-temporal model.

Lemma 1. *Assume we have the joint distribution* $\boldsymbol{\epsilon} \sim N\left(\mu, \frac{\sigma_t^2}{1-\phi^2}\boldsymbol{\Sigma}\right)$ *where* $\Sigma_{ij} = \phi^{||i-j||}$*, and the conditional distributions*

$$\begin{bmatrix} W_{1,1} \\ W_{1,2} \end{bmatrix} \mid \epsilon_1 \sim N\left(\epsilon_1, \frac{\sigma_s^2}{1-\phi^2}\begin{bmatrix} 1 & f(d) \\ f(d) & 1 \end{bmatrix}\right)$$

and

$$\begin{bmatrix} W_{2,1} \\ W_{2,2} \end{bmatrix} \mid \left\{\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix}, \epsilon_2, \epsilon_1\right\} \sim N\left(\epsilon_2 + \phi\left(\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} - \epsilon_1\right), \sigma_s^2\begin{bmatrix} 1 & f(d) \\ f(d) & 1 \end{bmatrix}\right),$$

where $f(d)$ *is the spatial correlation function evaluated at the distance* d *between* $W_{\cdot,1}$ *and* $W_{\cdot,2}$*.*

Then marginalizing out $\boldsymbol{\epsilon}$ *gives the distributions*

$$\begin{bmatrix} W_{1,1} \\ W_{1,2} \end{bmatrix} \sim N\left(\begin{bmatrix} \mu \\ \mu \end{bmatrix}, \frac{\sigma_t^2}{1-\phi^2} + \frac{\sigma_s^2}{1-\phi^2}\begin{bmatrix} 1 & f(d) \\ f(d) & 1 \end{bmatrix}\right)$$

and

$$\begin{bmatrix} W_{2,1} \\ W_{2,2} \end{bmatrix} \mid \begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} \sim N\left(\mu + \phi\left(\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} - \mu\right), \sigma_t^2 + \sigma_s^2\begin{bmatrix} 1 & f(d) \\ f(d) & 1 \end{bmatrix}\right)$$

Proof. The joint distribution of $\boldsymbol{\epsilon}$ is well known as an AR(1) structure, so that we have $\epsilon_1 \sim \mathrm{N}\left(\mu, \frac{\sigma_t^2}{1-\phi^2}\right)$ and $\epsilon_2|\epsilon_1 \sim \mathrm{N}\left(\mu + \phi\left(\epsilon_1 - \mu\right),\ \sigma_t^2\right)$, and also $\epsilon_2 - \phi\epsilon_1 \sim \mathrm{N}\left(\left(1-\phi\right)\mu,\ \sigma_t^2\right)$.

Note that we can define

$$\begin{bmatrix} W_{1,1} \\ W_{1,2} \end{bmatrix} = \epsilon_1 + \nu_1 \quad \text{where} \quad \nu_1 \sim \mathrm{N}\left(0, \frac{\sigma_s^2}{1-\phi^2}\begin{bmatrix} 1 & f\left(d\right) \\ f\left(d\right) & 1 \end{bmatrix}\right)$$

and ν_1 is independent of ϵ_1. Since this is a sum of independent normal random variables, we have

$$\begin{bmatrix} W_{1,1} \\ W_{1,2} \end{bmatrix} \sim \mathrm{N}\left(\begin{bmatrix} \mu \\ \mu \end{bmatrix},\ \frac{\sigma_t^2}{1-\phi^2} + \frac{\sigma_s^2}{1-\phi^2}\begin{bmatrix} 1 & f\left(d\right) \\ f\left(d\right) & 1 \end{bmatrix}\right).$$

Similarly we can define

$$\begin{bmatrix} W_{2,1} \\ W_{2,2} \end{bmatrix} \mid \begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} = \epsilon_2 + \phi\left(\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} - \epsilon_1\right) + \nu_2 \quad \text{where} \quad \nu_2 \sim \mathrm{N}\left(0, \sigma_s^2\begin{bmatrix} 1 & f\left(d\right) \\ f\left(d\right) & 1 \end{bmatrix}\right)$$

and ν_2 is independent of ϵ_1 and ϵ_2. Re-arranging terms we have

$$\begin{aligned} \epsilon_2 + \phi\left(\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} - \epsilon_1\right) + \nu_2 &= \epsilon_2 - \phi\left(\epsilon_1\right) + \phi\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} + \nu_2 \\ &= \mu - \phi\mu + \zeta + \phi\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} + \nu_2 \end{aligned}$$

where $\zeta \sim \mathrm{N}\left(0, \sigma_t^2\right)$,

$$= \mu + \phi\left(\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} - \mu\right) + \zeta + \nu_2$$

and thus

$$\begin{bmatrix} W_{2,1} \\ W_{2,2} \end{bmatrix} \mid \begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} \sim \mathrm{N}\left(\mu + \phi \left(\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} - \mu \right),\ \sigma_t^2 + \sigma_s^2 \begin{bmatrix} 1 & f(d) \\ f(d) & 1 \end{bmatrix} \right)$$

which completes the proof □

Lemma 2. *A spatio-temporal Gaussian process where 1.) any pair of random variables at the initial time point has joint distribution*

$$\begin{bmatrix} W_{1,1} \\ W_{1,2} \end{bmatrix} \sim N\left(\begin{bmatrix} \mu \\ \mu \end{bmatrix},\ \frac{\sigma_t^2}{1-\phi^2} + \frac{\sigma_s^2}{1-\phi^2} \begin{bmatrix} 1 & f(d) \\ f(d) & 1 \end{bmatrix} \right)$$

where $f(d)$ is the spatial correlation function evaluated at the distance d between $W_{1,1}$ and $W_{1,2}$, and 2.) any pair of random variables at any subsequent time points at the same locations as $W_{1,1}$ and $W_{1,2}$ has joint distribution

$$\begin{bmatrix} W_{t,1} \\ W_{t,2} \end{bmatrix} \mid \begin{bmatrix} w_{t-1,1} \\ w_{t-1,2} \end{bmatrix} \sim N\left(\mu + \phi \left(\begin{bmatrix} w_{t-1,1} \\ w_{t-1,2} \end{bmatrix} - \mu \right),\ \sigma_t^2 + \sigma_s^2 \begin{bmatrix} 1 & f(d) \\ f(d) & 1 \end{bmatrix} \right)$$

and 3.) any random effect at time t is conditionally independent of all random effects at time $t-2$ and earlier given all random effects at time $t-1$, is equivalent to a separable spatio-temporal Gaussian process with

1. *constant marginal mean $\mu' = \mu$*
2. *marginal variance $\sigma'^2 = (\sigma_t^2 + \sigma_s^2)/(1-\phi^2)$,*
3. *temporal correlation function $C_t'(\Delta t) = \phi^{||\Delta t||}$, and*
4. *spatial correlation function $C_s'(d) = (\sigma_t^2 + \sigma_s^2 f(d))/(\sigma_t^2 + \sigma_s^2)$.*

Proof. Let $\begin{bmatrix} W_{2,1} & W_{2,2} & W_{1,1} & W_{1,2} \end{bmatrix}^T$ be random effects where the subscript t, s indexes time and location, respectively, with the distance between locations 1 and 2 being equal to d. A separable spatio-temporal model for these random effects is given

in general by

$$\begin{bmatrix} W_{2,1} \\ W_{2,2} \\ W_{1,1} \\ W_{1,2} \end{bmatrix} \sim \mathrm{N}\left(\begin{bmatrix} \mu_{2,1} \\ \mu_{2,2} \\ \mu_{1,1} \\ \mu_{1,2} \end{bmatrix}, \ \sigma'^2 \begin{bmatrix} 1 & C'_s(d) & C'_t(1) & C'_t(1)C'_s(d) \\ C'_s(d) & 1 & C'_t(1)C'_s(d) & C'_t(1) \\ C'_t(1) & C'_t(1)C'_s(d) & 1 & C'_s(d) \\ C'_t(1)C'_s(d) & C'_t(1) & C'_s(d) & 1 \end{bmatrix} \right).$$

We immediately have the marginal joint distribution

$$\begin{bmatrix} W_{1,1} \\ W_{1,2} \end{bmatrix} \sim \mathrm{N}\left(\begin{bmatrix} \mu_{1,1} \\ \mu_{1,2} \end{bmatrix}, \ \sigma'^2 \begin{bmatrix} 1 & C'_s(d) \\ C'_s(d) & 1 \end{bmatrix} \right),$$

and without showing the straightforward but cumbersome details of the conditional normal calculations, the conditional joint distribution

$$\begin{bmatrix} W_{2,1} \\ W_{2,2} \end{bmatrix} \ | \ \begin{bmatrix} W_{1,1} \\ W_{1,2} \end{bmatrix} \sim \mathrm{N}\left(\begin{bmatrix} \mu_{2,1} \\ \mu_{2,2} \end{bmatrix} + C'_t(1) \left(\begin{bmatrix} w_{1,1} \\ w_{1,2} \end{bmatrix} - \begin{bmatrix} \mu_{1,1} \\ \mu_{1,2} \end{bmatrix} \right), \ \sigma'^2 \left(1 - C'_t(1)^2\right) \begin{bmatrix} 1 & C'_s(d) \\ C'_s(d) & 1 \end{bmatrix} \right).$$

For $\mathrm{E}\left[\begin{bmatrix} W_{1,1} & W_{1,2} \end{bmatrix}^T \right]$ to be the same between the initial model and the separable model we must have $\mu_{1,1} = \mu_{1,2} = \mu$. For $\mathrm{E}\left[\begin{bmatrix} W_{2,1} & W_{2,2} \end{bmatrix}^T \mid \begin{bmatrix} W_{1,1} & W_{1,2} \end{bmatrix}^T \right]$ to be the same, we must have

$$\mu_{2,1} + C'_t(1)\left(w_{1,1} - \mu_{1,1}\right) = \mu + \phi\left(w_{1,1} - \mu\right)$$

for all values of $w_{1,1}$. Letting $w_{1,1} = 0$ and substituting $\mu_{1,1} = \mu$ we have $\mu_{2,1} - C'_t(1)\mu = \mu - \phi\mu$. Now let $w_{1,1} = 1$ to get $(\mu_{2,1} - C'_t(1)\mu) + C'_t(1) = (\mu - \phi\mu) + \phi$, and thus $C'_t(1) = \phi$. Substituting $C'_t(1) = \phi$ into $\mathrm{E}\left[\begin{bmatrix} W_{2,1} & W_{2,2} \end{bmatrix}^T \mid \begin{bmatrix} W_{1,1} & W_{1,2} \end{bmatrix}^T \right]$ for the separable model gives $\mu_{2,1} = \mu_{2,2} = \mu$.

Moving to the covariance matrices, for $\mathrm{Var}\left[\begin{bmatrix} W_{1,1} & W_{1,2} \end{bmatrix}^T \right]$ to be the same we compare entries of the respective covariance matrices and must have

$$\frac{\sigma_t^2 + \sigma_s^2}{1 - \phi^2} = \sigma'^2$$

which gives our expression for σ'^2, and

$$\frac{\sigma_t^2}{1-\phi^2} + \frac{\sigma_s^2}{1-\phi^2} f(d) = \sigma'^2 C'_s(d).$$

Substituting $\sigma'^2 = (\sigma_t^2 + \sigma_s^2)/(1-\phi^2)$ and rearranging gives

$$C'_s(d) = \frac{\sigma_t^2 + \sigma_s^2 f(d)}{\sigma_t^2 + \sigma_s^2},$$

which completes the specification of the separable spatio-temporal model.

Straightforward calculations show that $\text{Var}\left[\begin{bmatrix} W_{2,1} & W_{2,2} \end{bmatrix}^T \mid \begin{bmatrix} W_{1,1} & W_{1,2} \end{bmatrix}^T\right]$ is the same for both models. Less straightforward but doable calculations for

$$\text{Var}\left[\begin{bmatrix} W_{3,1} & W_{3,2} \end{bmatrix}^T \mid \begin{bmatrix} W_{2,1} & W_{2,2} & W_{1,1} & W_{1,2} \end{bmatrix}^T\right]$$

show that $\begin{bmatrix} W_{3,1} & W_{3,2} \end{bmatrix}^T$ is conditionally independent of $\begin{bmatrix} W_{1,1} & W_{1,2} \end{bmatrix}^T$ given $\begin{bmatrix} W_{2,1} & W_{2,2} \end{bmatrix}^T$. Since the covariance matrix is invariant with respect to shifts in time, the conditional independence holds any three adjacent time points. The conditional independence and the above calculations also ensure that the conditional distributions $\begin{bmatrix} W_{t,1} & W_{t,2} \end{bmatrix}^T \mid \begin{bmatrix} W_{t-1,1} & W_{t-1,2} \end{bmatrix}^T$ is the same between the two models. With the marginal joint distribution for the first time point and conditional joint distributions for subsequent time points identical between the models, the proof is complete. □

Lemmas 1 and 2 show that if the staRVe model used a regular Gaussian process instead of a nearest-neighbour Gaussian process that the spatio-temporal random effects in the staRVe model are equivalent to spatio-temporal random effects with a separable covariance function. However the modification the staRVe model uses to turn a Gaussian process into a nearest neighbour Gaussian process can be applied identically to the Gaussian process with separable covariance function, including which locations and times are nearest neighbours of other locations and times. Thus:

Theorem 1. *The spatio-temporal random effects of the staRVe model have the same distributional properties as a spatio-temporal nearest neighbour Gaussian process whose covariance function is that given in Lemma 2.*

Appendix B

An Introduction to the staRVe Package

B.1 Introduction

```
set.seed(4326)
library(staRVe)
#> Loading required package: sf
#> Linking to GEOS 3.8.0, GDAL 3.0.4, PROJ 6.3.1
```

The staRVe package implements a spatio-temporal GLMM for analyzing discrete-time, continuous-space data. Throughout this vignette we will refer to the time unit as year/yearly, although any other regular unit (daily, monthly, etc.) can be used. Functionality include maximum likelihood inference, model-based spatial interpolation and forecasting and hindcasting, and simulations. The sf and raster packages are tightly integrated into the staRVe package, allowing users to easily incorporate the staRVe package into their existing workflows.

The spatio-temporal GLMM is structured as a hierarchical model with 3 levels:

1. an observation equation describing the response distribution and covariate effects
2. a first level of random effects describing the value of an unobserved spatio-temporal process, and
3. a second level of random effects describing how the spatial surface changes as a whole from year to year.

B.2 Data

The staRVe package expects data to be a sf data.frame with point geometries. See the sf package. The point geometries can have any valid geographic coordinate reference system, projected or unprojected, latitude/longitude or eastings/northings, or have

no coordinate reference system if the locations are not geographically referenced. In addition to the geometry column the sf data.frame should also include a response variable, the year of the observation, and optionally any covariates desired. The bird_survey dataset included in the staRVe package is an example of the necessary data format.

```
data(bird_survey)
bird_survey
#> Simple feature collection with 783 features and 2 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.23445 ymin: 36.11935 xmax:
    -89.2478 ymax: 40.47128
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>    cnt year                          geom
#> 1    4 1994 POINT (-89.24779 36.82526)
#> 2    2 1994 POINT (-90.74799 36.59369)
#> 3    8 1994 POINT (-91.68653 36.86373)
#> 4    6 1994  POINT (-93.9101 36.66086)
#> 5   15 1994 POINT (-94.23886 36.56423)
#> 6    2 1994 POINT (-89.52656 37.30663)
#> 7   17 1994 POINT (-90.30749 37.25986)
#> 8    7 1994 POINT (-91.55293 37.37596)
#> 9   13 1994 POINT (-91.42393 37.90398)
#> 10  10 1994   POINT (-92.0354 37.7511)
```

The 'bird_survey' data holds the results of a yearly survey of Carolina wren in the Missouri, USA. The number of individual birds (species: Carolina wren) encountered is stored in the 'cnt' column. The year that each observation was taken in stored in the 'year' column. The location of each observation, taken as the centroid of transect, is given in the 'geom' column.

```
bird_survey<- subset(bird_survey, year >= 2005)
```

B.2.1 Creating and Fitting a Model

The 'prepare_staRVe_model' function uses a supplied sf data.frame to create a 'staRVe_model' object.

```
bird_model<- prepare_staRVe_model(
  cnt ~ time(year),
  data = bird_survey,
  distribution = "poisson"
)
```

The first argument supplied to 'prepare_staRVe_model' is a formula, similar those used in calls to 'lm' or 'glm'. Formulae used here need to have a special syntax, described more thoroughly in Section B.8.1. The response distribution must also be specified, defaulting to a Gaussian distribution. The available response distributions are described in Section B.8.2 and can also be viewed in your R session with the command 'get_staRVe_distributions("distribution")'. Optional arguments for prepare_staRVe_model are described in the help file for the function.

The new staRVe_model object is shown below.

```
bird_model
#>
#> An object of class "staRVe_parameters"
#> Slot "covariance_function":
#> [1] "exponential"
#>
#> Slot "spatial_parameters":
#> $cnt
#>       par se fixed
#> sd    0.0  0 FALSE
#> range 0.0  0 FALSE
#> nu    0.5  0  TRUE
#>
#>
```

```
#> Slot "time_parameters":
#> $cnt
#>      par se fixed
#> mu     0 NA FALSE
#> ar1    0 NA FALSE
#> sd     0 NA FALSE
#>
#>
#> Slot "response_distribution":
#> [1] "poisson"
#>
#> Slot "response_parameters":
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
#>
#>
#> Slot "link_function":
#> [1] "log"
#>
#> Slot "fixed_effects":
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
#>
#>
#>
#> Data
#> Simple feature collection with 404 features and 2 fields
#> Geometry type: POINT
#> Dimension:      XY
```

```
#> Bounding box:  xmin: -95.23445 ymin: 36.11935 xmax:
   -89.2478 ymax: 40.47128
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>     cnt year                          geom
#> 422   1 2005 POINT (-95.23445 40.46577)
#> 417   7 2005 POINT (-95.03312 40.02546)
#> 402   5 2005 POINT (-94.66548 39.65398)
#> 415   7 2005 POINT (-94.50103 39.39702)
#> 416   3 2005 POINT (-94.14222 40.10026)
#> 392   3 2005   POINT (-94.1192 37.1404)
#> 420   4 2005 POINT (-94.10504 38.66985)
#> 384  11 2005  POINT (-93.9101 36.66086)
#> 391   7 2005 POINT (-93.85185 37.77059)
#> 397   2 2005 POINT (-93.84015 38.19498)
```

```
spatial_parameters(bird_model)$cnt["range","par"]<- 80
spatial_parameters(bird_model)$cnt["range","fixed"]<- TRUE
```

The individual components of a 'staRVe_model" object are explained in the following sections. For now, note the columns labelled 'par' in the 'time_parameters', 'spatial_parameters', 'response_parameters', and 'fixed_effects' slots (the latter two of which have 0 rows in this particular model). These are the parameter values for the model. The 'staRVe_fit' function finds the maximum likelihood (ML) estimates of these parameters. The parameter values in the staRVe_model object will be the starting values used when finding the ML estimate. Details for changing these values is given in Section B.4.

```
bird_model<- staRVe_fit(
  bird_model,
  silent=T
)
```

The output of the 'staRVe_fit' function is a 'staRVe_model_fit' object.

```
bird_model
#>
#> [1] "relative convergence (4)"
#>
#> An object of class "staRVe_parameters"
#> Slot "covariance_function":
#> [1] "exponential"
#>
#> Slot "spatial_parameters":
#> $cnt
#>                par          se fixed
#> sd      0.5522385 0.04145799 FALSE
#> range  80.0000000 0.00000000  TRUE
#> nu      0.5000000 0.00000000  TRUE
#>
#>
#> Slot "time_parameters":
#> $cnt
#>             par         se fixed
#> mu   2.0066523 0.3639902 FALSE
#> ar1  0.9235844 0.0316828 FALSE
#> sd   0.3816032 0.2212175 FALSE
#>
#>
#> Slot "response_distribution":
#> [1] "poisson"
#>
#> Slot "response_parameters":
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
#>
```

```
#>
#> Slot "link_function":
#> [1] "log"
#>
#> Slot "fixed_effects":
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
#>
#>
#>
#> Data
#> Simple feature collection with 404 features and 2 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.23445 ymin: 36.11935 xmax:
   -89.2478 ymax: 40.47128
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>     cnt year                          geom
#> 422   1 2005 POINT (-95.23445 40.46577)
#> 417   7 2005 POINT (-95.03312 40.02546)
#> 402   5 2005 POINT (-94.66548 39.65398)
#> 415   7 2005 POINT (-94.50103 39.39702)
#> 416   3 2005 POINT (-94.14222 40.10026)
#> 392   3 2005   POINT (-94.1192 37.1404)
#> 420   4 2005 POINT (-94.10504 38.66985)
#> 384  11 2005  POINT (-93.9101 36.66086)
#> 391   7 2005 POINT (-93.85185 37.77059)
#> 397   2 2005 POINT (-93.84015 38.19498)
```

A 'staRVe_model_fit' object is a 'staRVe_model' object with some extra functionality

providing tracing information (such as the parameter estimator covariance matrix, optimizer convergence, among other) and model-based predictions.

B.3 The staRVe Model Object

In this section we detail the various components of the 'staRVe_model' and 'staRVe_model_fit' classes. The staRVe package makes use of S4 classes. Users who are unfamiliar with S4 can think of an S4 class as a list, but elements of the list are accessed or modified using functions instead of the '$' or '[[' syntax.

B.3.1 Parameter Estimates

```
convergence(bird_model)
#> [1] "relative convergence (4)"
```

The convergence message is only available after fitting a model. Some of the convergence messages that indicate successful convergence of the optimization routine are

- "X-convergence (3)"
- "Relative convergence (4)"
- "Both X- and relative convergence (5)"

Some messages that indicate the optimization routine failed to converge are

- "Singular convergence"
- "false convergence"
- "function evluation limit reached"

See '?nlminb' and the PORT routines documentation for more details.

```
parameters(bird_model)
#> An object of class "staRVe_parameters"
#> Slot "covariance_function":
#> [1] "exponential"
#>
#> Slot "spatial_parameters":
```

```
#> $cnt
#>                par          se fixed
#> sd      0.5522385 0.04145799 FALSE
#> range  80.0000000 0.00000000  TRUE
#> nu      0.5000000 0.00000000  TRUE
#>
#>
#> Slot "time_parameters":
#> $cnt
#>            par         se fixed
#> mu   2.0066523 0.3639902 FALSE
#> ar1  0.9235844 0.0316828 FALSE
#> sd   0.3816032 0.2212175 FALSE
#>
#>
#> Slot "response_distribution":
#> [1] "poisson"
#>
#> Slot "response_parameters":
#> $cnt
#> [1] par   se    fixed
#> <0 rows> (or 0-length row.names)
#>
#>
#> Slot "link_function":
#> [1] "log"
#>
#> Slot "fixed_effects":
#> $cnt
#> [1] par   se    fixed
#> <0 rows> (or 0-length row.names)
```

The parameters hold the parameter values/estimates, the standard errors of the parameter estimates (for a fitted model), and whether the parameters are held fixed in the model. These quantities are held in the 'par', 'se', and 'fixed' columns, respectively. The spatial covariance function, response distribution, and link function used are also part of the parameter descriptions. The parameters are split into groups according to the different conceptual parts of the model.

```
covariance_function(bird_model)
#> [1] "exponential"
spatial_parameters(bird_model)
#> $cnt
#>                 par          se fixed
#> sd        0.5522385 0.04145799 FALSE
#> range    80.0000000 0.00000000  TRUE
#> nu        0.5000000 0.00000000  TRUE
```

The spatial parameters describe the spatial covariance function for each response variable. The staRVe package uses a Matérn covariance function with the following parameters:

- **sd**: a scale parameter. Higher values correspond to spatial surfaces with larger peaks and lower valleys. Must be greater than 0.
- **range**: a range parameter. Typically interpreted as the distance by which two locations need to be separated to be approximately uncorrelated.
- **nu**: a smoothness parameter. Higher values correspond to smoother spatial surfaces. The smoothness parameter is only estimated when the covariance function is 'matern'. Must be greater than 0.

```
time_parameters(bird_model)
#> $cnt
#>              par          se fixed
#> mu      2.006652 0.363990 FALSE
#> ar1     0.923584 0.031683 FALSE
#> sd      0.381603 0.221218 FALSE
```

The time parameters describe the temporal structure of the random effects. An autoregressive order 1 AR(1) model is used:

- **mu**: A mean parameter. The overall mean or the baseline level of the random effects.
- **ar1**: A correlation parameter. Higher values correspond to adjacent years being more highly correlated. Must be between -1 and +1.
- **sd**: A scale parameter. Higher values correspond to more year-to-year variation. Must be greater than 0.

```
response_distribution(bird_model)
#> [1] "poisson"
response_parameters(bird_model)
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
```

The response parameters describe e.g. the observation error, overdispersion, etc. The specific parameters depend on the response distribution used and are described in Section B.8.2.

```
fixed_effects(bird_model)
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
```

The fixed effects describe the effects of covariates, as in a generalized linear model. A description of including covariates in a staRVe analysis is given in Section B.7.

B.3.2 Random Effect Predictions

```
time_effects(bird_model)
#> stars object with 2 dimensions and 2 attributes
#> attribute(s):
#>              Min.   1st Qu.    Median      Mean   3rd Qu.
      Max.
```

```
#> w    1.8029336 1.9524988 1.9773845 2.0001214 2.0835892
   2.1469207
#> se   0.1354262 0.1829776 0.2140345 0.2091062 0.2409867
   0.2626483
#> dimension(s):
#>             from to offset delta refsys point values
#> year           1 10   2005     1     NA FALSE   NULL
#> variable       1  1     NA    NA     NA FALSE    cnt
```

The time effects are non-spatial random effects that describe the year-to-year change of the spatio-temporal random effects. Higher predicted values of the time effects correspond to years with a higher yearly mean, but the time effects themselves are not interpretable as yearly means. The main purpose of the time effects is to induce a specific correlation structure in the spatio-temporal random effects.

```
random_effects(bird_model)
#> stars object with 3 dimensions and 2 attributes
#> attribute(s):
#>             Min.    1st Qu.     Median       Mean    3rd Qu.
        Max.
#> w    0.0571734 1.2907434 1.8610282 1.8200177 2.3879034
   3.6965094
#> se   0.1390956 0.2317796 0.3151041 0.3574349 0.4570183
   0.9009029
#> dimension(s):
#>             from to offset delta refsys point
#> geom           1 63     NA    NA WGS 84  TRUE
#> year           1 10   2005     1     NA FALSE
#> variable       1  1     NA    NA     NA FALSE
#>

   values
```

```
#> geom       POINT (-89.2478 36.8253) ,... ,POINT (-95.2345
   40.4658)
#> year

   NULL
#> variable
                                                          cnt
```

These random effects describe a spatial surface that varies from year to year. The locations of the random effects act like the nodes in a mesh used in the R-INLA package, or basis functions used in a spline model. The spatial surface at new locations are predicted using the values of these random effects. Since the same random effect locations are repeated at every time point they form a data cube, and we hold the random effect in a stars object.

B.3.3 Data

```
data_predictions(bird_model)
#> An object of class "staRVe_predictions"
#> Slot "predictions":
#> stars object with 2 dimensions and 6 attributes
#> attribute(s):
#>                    Min.   1st Qu.     Median       Mean   3rd
   Qu.        Max.
#> w             0.0571734 1.430851   1.973692   1.947796
   2.531512   3.696509
#> w_se          0.0571734 1.430851   1.973692   1.947796
   2.531512   3.696509
#> linear        0.0571734 1.430851   1.973692   1.947796
   2.531512   3.696509
#> linear_se     0.0571734 1.430851   1.973692   1.947796
   2.531512   3.696509
#> response      1.0605700 8.464193 21.215507 41.498566
   52.858235 315.683102
```

```
#> response_se  0.0605869 8.513740 24.388735 51.448540
   65.259989 416.969502
#> dimension(s):
#>           from  to offset delta refsys point values
#> i            1 404      1     1     NA FALSE   NULL
#> variable     1   1     NA    NA     NA FALSE    cnt
#>
#> Slot "locations":
#> Simple feature collection with 404 features and 2 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.23445 ymin: 36.11935 xmax:
   -89.2478 ymax: 40.47128
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>     cnt year                        geom
#> 422   1 2005 POINT (-95.23445 40.46577)
#> 417   7 2005 POINT (-95.03312 40.02546)
#> 402   5 2005 POINT (-94.66548 39.65398)
#> 415   7 2005 POINT (-94.50103 39.39702)
#> 416   3 2005 POINT (-94.14222 40.10026)
#> 392   3 2005   POINT (-94.1192 37.1404)
#> 420   4 2005 POINT (-94.10504 38.66985)
#> 384  11 2005  POINT (-93.9101 36.66086)
#> 391   7 2005 POINT (-93.85185 37.77059)
#> 397   2 2005 POINT (-93.84015 38.19498)
```

The data used in the model holds the data supplied to 'prepare_staRVe_model', and is held in a 'staRVe_predictions' object. The 'predictions' slot contains a stars object for the random effect predictions (w), the response mean before applying the link function (linear), and the response mean after applying the link function (response). Standard errors are also given for these predictions. The 'locations' slot contains an sf

object with columns for the response variable, any covariates, and the year column.

```
dat(bird_model)
#> Simple feature collection with 404 features and 2 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.23445 ymin: 36.11935 xmax:
    -89.2478 ymax: 40.47128
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>     cnt year                        geom
#> 422   1 2005 POINT (-95.23445 40.46577)
#> 417   7 2005 POINT (-95.03312 40.02546)
#> 402   5 2005 POINT (-94.66548 39.65398)
#> 415   7 2005 POINT (-94.50103 39.39702)
#> 416   3 2005 POINT (-94.14222 40.10026)
#> 392   3 2005   POINT (-94.1192 37.1404)
#> 420   4 2005 POINT (-94.10504 38.66985)
#> 384  11 2005  POINT (-93.9101 36.66086)
#> 391   7 2005 POINT (-93.85185 37.77059)
#> 397   2 2005 POINT (-93.84015 38.19498)
```

The 'locations' slot can be easily accessed through the 'dat' function.

```
tmp<- cbind(
  do.call(cbind, predictions(data_predictions(bird_model))
     [,,1]),
  dat(bird_model)
)
colnames(tmp)[1:6]<- c("w","w_se","linear","linear_se","
   response","response_se")
tmp
#> Simple feature collection with 404 features and 8 fields
#> Geometry type: POINT
```

```
#> Dimension:      XY
#> Bounding box:  xmin: -95.23445 ymin: 36.11935 xmax:
   -89.2478 ymax: 40.47128
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>             w     w_se   linear linear_se  response
   response_se cnt year
#> 422 1.182316 1.182316 1.182316  1.182316  5.541796
   5.026848   1 2005
#> 417 1.712440 1.712440 1.712440  1.712440 13.668978
   14.905105   7 2005
#> 402 1.802291 1.802291 1.802291  1.802291 15.911438
   17.702799   5 2005
#> 415 2.034726 2.034726 2.034726  2.034726 23.486400
   27.274030   7 2005
#> 416 1.331535 1.331535 1.331535  1.331535  7.143872
   6.925622   3 2005
#> 392 1.854636 1.854636 1.854636  1.854636 17.378065
   19.542885   3 2005
#> 420 1.888163 1.888163 1.888163  1.888163 18.385112
   20.810472   4 2005
#> 384 2.340759 2.340759 2.340759  2.340759 38.850897
   47.026920  11 2005
#> 391 1.995752 1.995752 1.995752  1.995752 22.010749
   25.397791   7 2005
#> 397 1.528492 1.528492 1.528492  1.528492  9.997779
   10.378214   2 2005
#>                                 geom
#> 422 POINT (-95.23445 40.46577)
#> 417 POINT (-95.03312 40.02546)
#> 402 POINT (-94.66548 39.65398)
#> 415 POINT (-94.50103 39.39702)
```

```
#> 416 POINT (-94.14222 40.10026)
#> 392   POINT (-94.1192 37.1404)
#> 420 POINT (-94.10504 38.66985)
#> 384  POINT (-93.9101 36.66086)
#> 391 POINT (-93.85185 37.77059)
#> 397 POINT (-93.84015 38.19498)
```

The rows of 'predictions' are in the same order as the rows of 'locations'.

B.3.4 Settings

```
settings(bird_model)
#> An object of class "staRVe_settings"
#> Slot "formula":
#> cnt ~ time(year)
#>
#> Slot "n_neighbours":
#> [1] 15
#>
#> Slot "p_far_neighbours":
#> [1] 0
#>
#> Slot "distance_units":
#> [1] "km"
#>
#> Slot "max_distance":
#> [1] Inf
#>
#> Slot "obs_dag_method":
#> [1] "standard"
#>
#> Slot "extras":
#> list()
```

The call to 'prepare_staRVe_model' can include various options that tweak the behaviour of the data pre-processing step. Some of these options are stored as slots in model settings:

- **formula**: this is the formula supplied to 'prepare_staRVe_model'
- **n_neighbours**: the maximum number of parents that each node in the directed acyclic graph
- **p_far_neighbours**: the percentage of parents of each node that should be randomly selected, or the percentage of parents of each node that are not nearest neighbours
- **distance_units**: units used for distance calculations. All distances in the model (for example max_distance, init_range, and graph distances) have these units.
- **max_distance**: the maximum distance that a location is allowed to be from nearest-neighbour parents
- **obs_dag_method**: either "standard" or "inla.mesh", depending on how the directed acyclic graph was constructed
- **extras**: extra settings that are currently only used if the obs_dag_method is "inla.mesh"

B.4 Modifying a staRVe Model

```
spatial_parameters(bird_model)
#> $cnt
#>                  par           se  fixed
#> sd         0.5522385  0.04145799  FALSE
#> range     80.0000000  0.00000000   TRUE
#> nu         0.5000000  0.00000000   TRUE

covariance_function(bird_model)<- "matern32"

spatial_parameters(bird_model)
#> $cnt
#>          par  se  fixed
```

```
#>  sd      0.0   0 FALSE
#>  range   0.0   0 FALSE
#>  nu      1.5   0  TRUE
```

Changing the covariance function also resets the spatial parameters, and updates the value of nu to reflect the specific covariance function as a special case of the Matérn covariance.

```
spatial_parameters(bird_model)
#> $cnt
#>         par  se  fixed
#>  sd      0.0   0 FALSE
#>  range   0.0   0 FALSE
#>  nu      1.5   0  TRUE
time_parameters(bird_model)
#> $cnt
#>              par          se  fixed
#>  mu   2.0066523  0.3639902 FALSE
#>  ar1  0.9235844  0.0316828 FALSE
#>  sd   0.3816032  0.2212175 FALSE

time_parameters(bird_model)$cnt[c("mu","ar1"),"par"]<- c
    (10,0.75)
spatial_parameters(bird_model)$cnt["range","par"]<- 80
spatial_parameters(bird_model)$cnt["range","fixed"]<- T

spatial_parameters(bird_model)
#> $cnt
#>          par  se  fixed
#>  sd       0.0   0 FALSE
#>  range   80.0   0  TRUE
#>  nu       1.5   0  TRUE
time_parameters(bird_model)
```

```
#> $cnt
#>               par          se  fixed
#> mu   10.0000000  0.3639902 FALSE
#> ar1   0.7500000  0.0316828 FALSE
#> sd    0.3816032  0.2212175 FALSE
```

You can change the parameter values by accessing the relevant set of parameters, and changing the 'par' column of that data.frame. Parameter values can be set to any numeric value, but values outside the valid range will be replaced with a default value when fitting, simulating from, or predicting using the model. For example a 'sd' (standard deviation) parameter must be positive; if it is set to a negative value then a default of +1 will be used instead. You can also change the 'se' and 'fixed' columns in the same way.

```
response_distribution(bird_model)
#> [1] "poisson"
response_parameters(bird_model)
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
link_function(bird_model)
#> [1] "log"

response_distribution(bird_model)<- "gaussian"

response_distribution(bird_model)
#> [1] "gaussian"
response_parameters(bird_model)
#> $cnt
#>     par se fixed
#> sd    0  0 FALSE
link_function(bird_model)
#> [1] "identity"
```

Changing the response distribution will automatically update the response parameters, and will change the link function based on the valid range of the response mean for that distribution. In the example above, the response distribution was changed from a Poisson distribution with no parameters and log link to a Gaussian distribution with a standard deviation parameter and identity link.

```
link_function(bird_model)
#> [1] "identity"

link_function(bird_model)<- "logit"

link_function(bird_model)
#> [1] "logit"
```

The link function can be changed to any implemented link function, even if it may give out of range predictions for the response distribution used in the model. For example an identity link is allowed to be used with a Poisson response distribution even though an identity link may result in predictions with a negative mean. Non-standard link functions should be used with caution, but are sometimes more desirable depending on the specific situation.

```
dat(bird_model)
#> Simple feature collection with 404 features and 2 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.23445 ymin: 36.11935 xmax:
   -89.2478 ymax: 40.47128
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>     cnt year                        geom
#> 422   1 2005 POINT (-95.23445 40.46577)
#> 417   7 2005 POINT (-95.03312 40.02546)
#> 402   5 2005 POINT (-94.66548 39.65398)
#> 415   7 2005 POINT (-94.50103 39.39702)
```

```
#> 416    3 2005 POINT (-94.14222 40.10026)
#> 392    3 2005    POINT (-94.1192 37.1404)
#> 420    4 2005 POINT (-94.10504 38.66985)
#> 384   11 2005  POINT (-93.9101 36.66086)
#> 391    7 2005 POINT (-93.85185 37.77059)
#> 397    2 2005 POINT (-93.84015 38.19498)
```

```
dat(bird_model)$elev<- rnorm(nrow(dat(bird_model)))
```

```
dat(bird_model)
#> Simple feature collection with 404 features and 3 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.23445 ymin: 36.11935 xmax:
   -89.2478 ymax: 40.47128
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>     cnt year                         geom        elev
#> 422   1 2005 POINT (-95.23445 40.46577) -1.3221890
#> 417   7 2005 POINT (-95.03312 40.02546)  0.9720445
#> 402   5 2005 POINT (-94.66548 39.65398)  0.7193741
#> 415   7 2005 POINT (-94.50103 39.39702)  0.8373455
#> 416   3 2005 POINT (-94.14222 40.10026)  0.6659404
#> 392   3 2005    POINT (-94.1192 37.1404) -0.6571608
#> 420   4 2005 POINT (-94.10504 38.66985) -0.9045444
#> 384  11 2005  POINT (-93.9101 36.66086) -0.5842451
#> 391   7 2005 POINT (-93.85185 37.77059)  1.3389180
#> 397   2 2005 POINT (-93.84015 38.19498)  1.1219474
```

You can easily modify the data in a 'staRVe_model' object, adding columns for new variables or rows for additional observations, amending data entries, or removing observations when cross-validating. Since 'dat(bird_model)' is an 'sf' object you can

use spatial operations such as spatial joins or intersections to assist in adding new data.

```
formula(bird_model)
#> cnt ~ time(year)
fixed_effects(bird_model)
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)

formula(bird_model)<- cnt~elev+time(year)

formula(bird_model)
#> cnt ~ elev + time(year)
fixed_effects(bird_model)
#> $cnt
#>       par se fixed
#> elev    0 NA FALSE
```

The model formula for including covariates such as the newly created 'elev' variable will be explained later. Any variables used in the new formula must be present in the data held in the staRVe_model object, so add them before changing the formula. The fixed effects parameters will be automatically updated to reflect the new formula.

```
spatial_parameters(bird_model)
#> $cnt
#>          par se fixed
#> sd       0.0  0 FALSE
#> range 80.0  0  TRUE
#> nu       1.5  0  TRUE
graph(bird_model)
#> $persistent_graph
#>
```

```
#> [1] "A directed acyclic graph with 49 nodes, with an
   median in-degree of 15."
#> [1] "The average edge distance is 112.2km."
#>
#> $transient_graph
#>
#> [1] "A directed acyclic graph with 404 nodes, with an
   median in-degree of 1."
#> [1] "The average edge distance is 0km."
distance_units(bird_model)<- "ft"
spatial_parameters(bird_model)
#> $cnt
#>             par se fixed
#> sd          0.0  0 FALSE
#> range 262467.2  0  TRUE
#> nu          1.5  0  TRUE
graph(bird_model)
#> $persistent_graph
#>
#> [1] "A directed acyclic graph with 49 nodes, with an
   median in-degree of 15."
#> [1] "The average edge distance is 368123.15ft."
#>
#> $transient_graph
#>
#> [1] "A directed acyclic graph with 404 nodes, with an
   median in-degree of 1."
#> [1] "The average edge distance is 0ft."
```

The distance units used in calculations can be changed to any valid distance unit provided by the 'units' package. Distances are standardized within the model so changing the distance units should not affect inference or predictions.

B.5 Predictions

Once a staRVe model has been fit, you can then use the 'staRVe_predict' function to predict the mean value of the response variable at unobserved locations and years. This allows us to fill in the gaps and produce smooth maps of the response variable and forecast those maps into the future (or the past). Predictions can be made as individual locations or as a raster.

B.5.1 Point Predictions

```
pred_points<- st_sample(
  st_as_sfc(st_bbox(bird_survey)),
  size = 25
)
pred_points<- st_sf(geom = pred_points,year=rep(2012:2016,
   each=5))
pred_points
#> Simple feature collection with 25 features and 1 field
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.1653 ymin: 36.1487 xmax:
   -89.41267 ymax: 40.43233
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>    year                       geom
#> 1  2012 POINT (-93.44969 37.48708)
#> 2  2012  POINT (-92.76769 36.7374)
#> 3  2012 POINT (-89.66307 36.49022)
#> 4  2012 POINT (-89.41267 36.74711)
#> 5  2012  POINT (-95.1653 40.43233)
#> 6  2013 POINT (-91.23448 36.59283)
#> 7  2013  POINT (-94.56619 36.1487)
#> 8  2013 POINT (-92.09355 38.93433)
#> 9  2013 POINT (-89.75907 36.99413)
```

```
#> 10 2013 POINT (-89.79193 37.16898)
```

We first randomly sample a small number of points for predictions each year, and store those points as an sf data.frame. The prediction locations can be the same every year or they can be different, and the years of the predictions can be different from those in the original data.

```
sf_pred<- staRVe_predict(
  bird_model,
  pred_points
)
sf_pred
#> An object of class "staRVe_predictions"
#> Slot "predictions":
#> stars object with 2 dimensions and 6 attributes
#> attribute(s):
#>                    Min.    1st Qu.     Median       Mean    3rd
    Qu.       Max.
#> w             0.5778759 1.3982734 1.8673959 1.8267373
    2.2084364  3.2386205
#> w_se          0.2205018 0.3062865 0.4562022 0.4713493
    0.6400974  0.7676515
#> linear        0.5778759 1.3982734 1.8673959 1.8267373
    2.2084364  3.2386205
#> linear_se     0.2205018 0.3062865 0.4562022 0.4713493
    0.6400974  0.7676515
#> response      1.9800204 4.4580421 7.1448397 8.5969414
    9.7274763 26.1184036
#> response_se   0.8849767 1.9462821 2.9582826 3.3045932
    4.3035790  6.1377897
#> dimension(s):
#>           from to offset delta refsys point values
#> i            1 25      1     1     NA FALSE   NULL
```

```
#> variable     1  1      NA     NA      NA FALSE     cnt
#>
#> Slot "locations":
#> Simple feature collection with 25 features and 1 field
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.1653 ymin: 36.1487 xmax:
   -89.41267 ymax: 40.43233
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>    year                        geom
#> 1  2012 POINT (-93.44969 37.48708)
#> 2  2012  POINT (-92.76769 36.7374)
#> 3  2012 POINT (-89.66307 36.49022)
#> 4  2012 POINT (-89.41267 36.74711)
#> 5  2012  POINT (-95.1653 40.43233)
#> 6  2013 POINT (-91.23448 36.59283)
#> 7  2013  POINT (-94.56619 36.1487)
#> 8  2013 POINT (-92.09355 38.93433)
#> 9  2013 POINT (-89.75907 36.99413)
#> 10 2013 POINT (-89.79193 37.16898)
```

The output is a staRVe_predictions object with two slots, a 'locations' slot containing an sf data.frame with any covariates used, the year, and the geometry, and a 'predictions' slot containing a stars object with attributes for:

- The spatio-temporal random effects **w**, and their standard errors **w_se**
- The response mean before applying the link function **linear**, and their standard errors **linear_se**
- the response mean after applying the link function **response**, and their standard errors **response_se**

These predictions can be converted to a single sf data.frame fairly easily.

```
tmp<- cbind(
  do.call(cbind, predictions(sf_pred)[,,1]),
  locations(sf_pred)
)
colnames(tmp)[1:6]<- c("w","w_se","linear","linear_se","
    response","response_se")
tmp
#> Simple feature collection with 25 features and 7 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.1653 ymin: 36.1487 xmax:
    -89.41267 ymax: 40.43233
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>            w      w_se    linear linear_se  response
    response_se year
#> 1  1.8955527 0.2865756 1.8955527 0.2865756  6.929549
    1.9462821 2012
#> 2  2.2084364 0.3062865 2.2084364 0.3062865  9.528385
    2.8522875 2012
#> 3  2.5666439 0.4376121 2.5666439 0.4376121 14.268937
    5.9651965 2012
#> 4  2.9497886 0.3030083 2.9497886 0.3030083 19.978828
    5.9194043 2012
#> 5  0.5778759 0.4710997 0.5778759 0.4710997  1.980020
    0.8849767 2012
#> 6  2.0253056 0.3628616 2.0253056 0.3628616  8.077346
    2.8389970 2013
#> 7  1.4405066 0.4669391 1.4405066 0.4669391  4.683191
    2.0765062 2013
#> 8  1.3647773 0.3618615 1.3647773 0.3618615  4.171164
    1.4622734 2013
```

```
#> 9   2.8740004 0.2918432 2.8740004 0.2918432 18.461819
   5.2767688 2013
#> 10 3.2386205 0.2205018 3.2386205 0.2205018 26.118404
   5.6904033 2013
#>                                  geom
#> 1   POINT (-93.44969 37.48708)
#> 2    POINT (-92.76769 36.7374)
#> 3   POINT (-89.66307 36.49022)
#> 4   POINT (-89.41267 36.74711)
#> 5    POINT (-95.1653 40.43233)
#> 6   POINT (-91.23448 36.59283)
#> 7    POINT (-94.56619 36.1487)
#> 8   POINT (-92.09355 38.93433)
#> 9   POINT (-89.75907 36.99413)
#> 10 POINT (-89.79193 37.16898)
```

B.5.2 Raster Predictions

```
library(raster)
#> Loading required package: sp
pred_raster<- raster(
  bird_survey,
  nrow = 5,
  ncol = 5
)
pred_raster
#> class       : RasterLayer
#> dimensions  : 5, 5, 25  (nrow, ncol, ncell)
#> resolution  : 1.197331, 0.8703872  (x, y)
#> extent      : -95.23445, -89.24779, 36.11935, 40.47128  (
   xmin, xmax, ymin, ymax)
#> crs         : +proj=longlat +datum=WGS84 +no_defs
```

To predict a raster for a smooth prediction surface you can supply a raster object from the 'raster' package. The 'staRVe_predict' function handles rasters by extracting the centroid of each raster cell and giving point predictions at those centroids.

```
staRVe_predict(
  bird_model,
  pred_raster
)
#> stars object with 4 dimensions and 6 attributes
#> attribute(s):
#>                    Min.    1st Qu.     Median       Mean    3
   rd Qu.       Max.
#> w             0.1615796 1.2324019 1.8309501 1.7339035
   2.2950971   3.3125145
#> w_se          0.2479852 0.3589512 0.4197384 0.4126651
   0.4536496   0.6910432
#> linear        0.1615796 1.2324019 1.8309501 1.7339035
   2.2950971   3.3125145
#> linear_se     0.2479852 0.3589512 0.4197384 0.4126651
   0.4536496   0.6910432
#> response      1.3424364 3.7481071 6.7117281 7.6227424
   10.4576359 28.4854183
#> response_se   0.6431953 1.5589156 2.5806644 2.7841798
   3.7161660   7.6652846
#> dimension(s):
#>          from to    offset      delta refsys point
   values x/y
#> x           1  5 -95.2344    1.19733 WGS 84 FALSE
   NULL [x]
#> y           1  5  40.4713 -0.870387 WGS 84 FALSE
   NULL [y]
#> year        1 10      2005          1     NA    NA
   2005,...,2014
```

```
#> variable     1   1        NA          NA       NA      NA
                cnt
```

The output is a stars object with predictions at each raster location and every year.

```
pred_raster[]<- 0
pred_raster[3,3]<- NA
pred_raster
#> class       : RasterLayer
#> dimensions  : 5, 5, 25  (nrow, ncol, ncell)
#> resolution  : 1.197331, 0.8703872  (x, y)
#> extent      : -95.23445, -89.24779, 36.11935, 40.47128  (
   xmin, xmax, ymin, ymax)
#> crs         : +proj=longlat +datum=WGS84 +no_defs
#> source      : memory
#> names       : layer
#> values      : 0, 0  (min, max)
as.matrix(pred_raster)
#>      [,1] [,2] [,3] [,4] [,5]
#> [1,]    0    0    0    0    0
#> [2,]    0    0    0    0    0
#> [3,]    0    0   NA    0    0
#> [4,]    0    0    0    0    0
#> [5,]    0    0    0    0    0
pred_out<- staRVe_predict(
  bird_model,
  pred_raster
)
pred_out$response[,,1,1]
#>          [,1]     [,2]     [,3]     [,4]     [,5]
#> [1,] 4.881666 8.163354 7.295701 7.391376 8.414994
#> [2,] 3.682900 5.169961 5.692919 14.923899 11.632753
#> [3,] 3.548757 5.856864       NA 10.854109 7.546488
```

```
#>  [4,]  5.332430  6.682625  13.862354  10.150499  11.357214
#>  [5,]  7.864344  8.895579  11.347361  14.763676  12.048752
```

If the supplied raster has values, then predictions will only be made at cells that are not NA. See '?raster::mask' for details on creating raster with NAs.

```
staRVe_predict(
  bird_model,
  pred_raster,
  time = 2015:2019
)
#> stars object with 4 dimensions and 6 attributes
#> attribute(s):
#>                    Min.    1st Qu.     Median       Mean    3
   rd Qu.       Max. NA's
#> w             0.5570024  1.3213114  1.7767307  1.7402417
   2.1608859   3.215017    5
#> w_se          0.4142825  0.6395741  0.7250052  0.7219626
   0.8223665   0.971460    5
#> linear        0.5570024  1.3213114  1.7767307  1.7402417
   2.1608859   3.215017    5
#> linear_se     0.4142825  0.6395741  0.7250052  0.7219626
   0.8223665   0.971460    5
#> response      2.0947640  4.7463053  7.5401192  8.4564559
   10.4040055  27.040819    5
#> response_se   1.2097661  3.1631306  4.5959377  5.2098173
   6.5891112  13.678664    5
#> dimension(s):
#>           from  to    offset      delta  refsys  point
   values  x/y
#> x            1   5  -95.2344    1.19733  WGS 84  FALSE
   NULL  [x]
```

```
#> y              1   5   40.4713  -0.870387 WGS 84 FALSE
   NULL [y]
#> year           1   5       2015           1      NA      NA
   2015,...,2019
#> variable       1   1         NA          NA      NA      NA
                cnt
```

For raster predictions an additional 'time' option is available to specify which years to make predictions for, with the default behaviour being to predict every year between the earliest and latest year in the original dataset.

B.6 Simulation

Any 'staRVe_model' object can simulate new realizations of both the random effects and observations. You can simulated from an already fitted model, or by first defining a new model from scratch.

```
sim_model<- prepare_staRVe_model(
  cnt~time(year),
  bird_survey,
  distribution="poisson",
  fit=T
)
sim_model<- staRVe_simulate(sim_model)
sim_model
#>
#> [1] "relative convergence (4)"
#>
#> An object of class "staRVe_parameters"
#> Slot "covariance_function":
#> [1] "exponential"
#>
#> Slot "spatial_parameters":
#> $cnt
```

```
#>                 par se fixed
#> sd        0.4892491 NA FALSE
#> range 154.7750060 NA FALSE
#> nu        0.5000000 NA  TRUE
#>
#>
#> Slot "time_parameters":
#> $cnt
#>                par se fixed
#> mu  2.238773344 NA FALSE
#> ar1 0.933426418 NA FALSE
#> sd  0.000111516 NA FALSE
#>
#>
#> Slot "response_distribution":
#> [1] "poisson"
#>
#> Slot "response_parameters":
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
#>
#>
#> Slot "link_function":
#> [1] "log"
#>
#> Slot "fixed_effects":
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
#>
#>
```

```
#>
#> Data
#> Simple feature collection with 404 features and 2 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.23445 ymin: 36.11935 xmax:
    -89.2478 ymax: 40.47128
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>      cnt year                          geom
#> 422   31 2005 POINT (-95.23445 40.46577)
#> 417   21 2005 POINT (-95.03312 40.02546)
#> 402   13 2005 POINT (-94.66548 39.65398)
#> 415   23 2005 POINT (-94.50103 39.39702)
#> 416   18 2005 POINT (-94.14222 40.10026)
#> 392   23 2005   POINT (-94.1192 37.1404)
#> 420   22 2005 POINT (-94.10504 38.66985)
#> 384   19 2005  POINT (-93.9101 36.66086)
#> 391   18 2005 POINT (-93.85185 37.77059)
#> 397   26 2005 POINT (-93.84015 38.19498)
```

Passing a fitted staRVe model to 'staRVe_simulate' simulates new time effects, new random effects, and new observations, using the parameter estimates (or whichever parameter values are current). The output of 'staRVe_simulate' is itself a 'staRVe_model' object. The parameter values are the same as those of the supplied model, however all standard errors have been replace with NA. The time effects, random effects, and observations in the output are the simulated values. No record of the previous values are stored.

```
sim_df<- st_as_sf(
  data.frame(
    y = 0,
    t = rep(1:5, each=16),
```

```
    ycoord = rep(1:4,4),
    xcoord = rep(1:4,each=4)
  ),
  coords = c("ycoord","xcoord")
)
sim_df
#> Simple feature collection with 80 features and 2 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: 1 ymin: 1 xmax: 4 ymax: 4
#> CRS:           NA
#> First 10 features:
#>    y t     geometry
#> 1  0 1 POINT (1 1)
#> 2  0 1 POINT (2 1)
#> 3  0 1 POINT (3 1)
#> 4  0 1 POINT (4 1)
#> 5  0 1 POINT (1 2)
#> 6  0 1 POINT (2 2)
#> 7  0 1 POINT (3 2)
#> 8  0 1 POINT (4 2)
#> 9  0 1 POINT (1 3)
#> 10 0 1 POINT (2 3)

sim_model<- prepare_staRVe_model(
  y~time(t),
  sim_df,
  distribution = "gaussian",
  fit = F
)
```

To simulate from a completely new model you first have to create a template sf

data.frame. The template data.frame needs to contain a column for the response variable, a column for any covariates desired, a column for the time variable, and a geometry column. The values in the response variable column are irrelevant however the covariate values will be the ones used in the model, the time and geometry columns are also used to set up the spatio-temporal layout of the simulations. The sf data.frame constructed above consists of a 4x4 grid (16 points) replicated over 5 years.

```
time_parameters(sim_model)$y[c("mu","ar1","sd"),"par"]<- c
    (10,0.7,1)
spatial_parameters(sim_model)$y[c("sd","range"),"par"]<- c
    (2,0.5)
response_parameters(sim_model)$y["sd","par"]<- 0.5
sim_model<- staRVe_simulate(sim_model)
sim_model
#>
#> An object of class "staRVe_parameters"
#> Slot "covariance_function":
#> [1] "exponential"
#>
#> Slot "spatial_parameters":
#> $y
#>        par se fixed
#> sd     2.0 NA FALSE
#> range  0.5 NA FALSE
#> nu     0.5 NA  TRUE
#>
#>
#> Slot "time_parameters":
#> $y
#>       par se fixed
#> mu   10.0 NA FALSE
#> ar1   0.7 NA FALSE
#> sd    1.0 NA FALSE
```

```
#>
#>
#> Slot "response_distribution":
#> [1] "gaussian"
#>
#> Slot "response_parameters":
#> $y
#>    par se fixed
#> sd 0.5 NA FALSE
#>
#>
#> Slot "link_function":
#> [1] "identity"
#>
#> Slot "fixed_effects":
#> $y
#> [1] par   se    fixed
#> <0 rows> (or 0-length row.names)
#>
#>
#>
#> Data
#> Simple feature collection with 80 features and 2 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: 1 ymin: 1 xmax: 4 ymax: 4
#> CRS:           NA
#> First 10 features:
#>             y t     geometry
#> 1   11.518110 1 POINT (1 1)
#> 5   12.138294 1 POINT (1 2)
#> 9   11.746391 1 POINT (1 3)
```

```
#> 13 10.835629 1 POINT (1 4)
#> 2  11.809868 1 POINT (2 1)
#> 6  11.390783 1 POINT (2 2)
#> 10 11.392126 1 POINT (2 3)
#> 14 10.754744 1 POINT (2 4)
#> 3   9.380358 1 POINT (3 1)
#> 7  12.269180 1 POINT (3 2)
```

Once a staRVe model is created from the template data, the parameter values and other model options (such as the covariance function, link function, response distribution, etc.) need to be set to their desired value.

B.6.1 Simulating Random Effects and Observations

time_effects(sim_**model**)

```
#> stars object with 2 dimensions and 2 attributes
#> attribute(s):
#>        w                 se
#>  Min.    : 9.325    Mode:logical
#>  1st Qu.: 9.730    NA's:5
#>  Median : 9.940
#>  Mean   :10.202
#>  3rd Qu.:10.474
#>  Max.   :11.543
#> dimension(s):
#>            from to offset delta refsys point values
#> t             1  5      1     1     NA FALSE   NULL
#> variable      1  1     NA    NA     NA FALSE      y
```

random_**effects**(sim_**model**)

```
#> stars object with 3 dimensions and 2 attributes
#> attribute(s):
#>        w                 se
#>  Min.    : 5.995    Mode:logical
#>  1st Qu.: 9.211    NA's:80
```

```
#>  Median :10.028
#>  Mean   :10.055
#>  3rd Qu.:11.295
#>  Max.   :12.492
#> dimension(s):
#>           from to offset delta refsys point
                                values
#> geom         1 16     NA    NA     NA  TRUE POINT (1 1)
   ,...,POINT (4 4)
#> t            1  5      1     1     NA FALSE
                                 NULL
#> variable     1  1     NA    NA     NA FALSE
                                     y
head(dat(sim_model)$y)
#>           [,1]
#> [1,] 11.51811
#> [2,] 12.13829
#> [3,] 11.74639
#> [4,] 10.83563
#> [5,] 11.80987
#> [6,] 11.39078
sim_model<- staRVe_simulate(
  sim_model
)
time_effects(sim_model)
#> stars object with 2 dimensions and 2 attributes
#> attribute(s):
#>        w              se
#>  Min.   : 8.759   Mode:logical
#>  1st Qu.: 9.123   NA's:5
#>  Median : 9.194
#>  Mean   : 9.505
```

```
#>  3rd Qu.: 9.948
#>  Max.    :10.503
#> dimension(s):
#>            from to offset delta refsys point values
#> t             1  5      1     1     NA FALSE   NULL
#> variable      1  1     NA    NA     NA FALSE      y
random_effects(sim_model)
#> stars object with 3 dimensions and 2 attributes
#> attribute(s):
#>        w              se
#>  Min.   : 6.224   Mode:logical
#>  1st Qu.: 8.488   NA's:80
#>  Median : 9.472
#>  Mean   : 9.482
#>  3rd Qu.:10.475
#>  Max.   :12.777
#> dimension(s):
#>            from to offset delta refsys point
                                values
#> geom          1 16     NA    NA     NA  TRUE POINT (1 1)
   ,...,POINT (4 4)
#> t             1  5      1     1     NA FALSE
                                  NULL
#> variable      1  1     NA    NA     NA FALSE
                                      y
head(dat(sim_model)$y)
#>              [,1]
#> [1,]  6.820019
#> [2,]  9.606529
#> [3,] 10.048682
#> [4,]  8.637561
#> [5,]  8.360983
```

```
#> [6,]   9.829688
```

The default behaviour for simulations is to simulate new time effects, new random effects, and new data. We show only the first few of each to show their difference before and after simulations.

B.6.2 Simulating Only Observations

time_effects(sim_**model**)

```
#> stars object with 2 dimensions and 2 attributes
#> attribute(s):
#>        w              se
#>  Min.   : 8.759   Mode:logical
#>  1st Qu.: 9.123   NA's:5
#>  Median : 9.194
#>  Mean   : 9.505
#>  3rd Qu.: 9.948
#>  Max.   :10.503
#> dimension(s):
#>           from to offset delta refsys point values
#> t            1  5      1     1     NA FALSE   NULL
#> variable     1  1     NA    NA     NA FALSE      y
```

random_**effects**(sim_**model**)

```
#> stars object with 3 dimensions and 2 attributes
#> attribute(s):
#>        w              se
#>  Min.   : 6.224   Mode:logical
#>  1st Qu.: 8.488   NA's:80
#>  Median : 9.472
#>  Mean   : 9.482
#>  3rd Qu.:10.475
#>  Max.   :12.777
#> dimension(s):
```

```
#>            from to offset delta refsys point
                                    values
#> geom          1 16     NA     NA     NA  TRUE POINT (1 1)
    ,...,POINT (4 4)
#> t             1  5      1      1     NA FALSE
                                   NULL
#> variable      1  1     NA     NA     NA FALSE
                                      y
head(dat(sim_model)$y)
#>              [,1]
#> [1,]  6.820019
#> [2,]  9.606529
#> [3,] 10.048682
#> [4,]  8.637561
#> [5,]  8.360983
#> [6,]  9.829688
sim_model<- staRVe_simulate(
  sim_model,
  conditional=T
)
time_effects(sim_model)
#> stars object with 2 dimensions and 2 attributes
#> attribute(s):
#>        w               se
#>  Min.    : 8.759   Mode:logical
#>  1st Qu.: 9.123   NA's:5
#>  Median : 9.194
#>  Mean   : 9.505
#>  3rd Qu.: 9.948
#>  Max.    :10.503
#> dimension(s):
#>            from to offset delta refsys point values
```

```
#> t              1   5        1       1       NA FALSE    NULL
#> variable       1   1       NA      NA       NA FALSE       y
random_effects(sim_model)
#> stars object with 3 dimensions and 2 attributes
#> attribute(s):
#>         w                se
#>  Min.    : 6.224    Mode:logical
#>  1st Qu.: 8.488    NA's:80
#>  Median : 9.472
#>  Mean   : 9.482
#>  3rd Qu.:10.475
#>  Max.    :12.777
#> dimension(s):
#>              from to offset delta refsys point
                                    values
#> geom           1 16       NA      NA       NA   TRUE POINT (1 1)
   ,...,POINT (4 4)
#> t              1   5        1       1       NA FALSE
                                      NULL
#> variable       1   1       NA      NA       NA FALSE
                                           y
head(dat(sim_model)$y)
#>               [,1]
#> [1,]  6.423825
#> [2,]  9.059653
#> [3,] 10.961186
#> [4,]  8.742246
#> [5,]  8.405299
#> [6,]  9.495317
```

Simulations of only new data, conditional on the existing time effects and random effects, can be made by giving the 'conditional=T' option. Note that the time effects

and random effects are the same before and after simulation, but that the response variable is different.

B.7 Including Covariates

B.7.1 Creating a Model

```
bird_survey$elev<- rnorm(
  nrow(bird_survey)
)
bird_survey$temp<- rnorm(
  nrow(bird_survey)
)
```

First we add two dummy covariates 'elev' and 'temp' to our original data.

```
bird_model<- prepare_staRVe_model(
  cnt ~ elev+temp+time(year),
  bird_survey,
  distribution = "poisson",
  fit = T
)

fixed_effects(bird_model)
#> $cnt
#>                par          se fixed
#> elev  0.03690417 0.02610423 FALSE
#> temp  0.03256844 0.02531721 FALSE
dat(bird_model)
#> Simple feature collection with 404 features and 4 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.23445 ymin: 36.11935 xmax:
    -89.2478 ymax: 40.47128
#> Geodetic CRS:  WGS 84
```

```
#> First 10 features:
#>              elev        temp cnt year
    geom
#> 422 -1.70221026  1.0958843   1 2005 POINT (-95.23445
    40.46577)
#> 417  1.34828796  2.1010647   7 2005 POINT (-95.03312
    40.02546)
#> 402  0.39185501  0.7861318   5 2005 POINT (-94.66548
    39.65398)
#> 415 -0.89223203  0.9132973   7 2005 POINT (-94.50103
    39.39702)
#> 416  0.34172737  0.4733139   3 2005 POINT (-94.14222
    40.10026)
#> 392 -2.27217139  0.8688053   3 2005   POINT (-94.1192
    37.1404)
#> 420 -0.34456416 -1.0913277   4 2005 POINT (-94.10504
    38.66985)
#> 384  0.70189276  0.5817127  11 2005  POINT (-93.9101
    36.66086)
#> 391 -0.08886507 -0.1119984   7 2005 POINT (-93.85185
    37.77059)
#> 397 -0.06566962 -0.1125152   2 2005 POINT (-93.84015
    38.19498)
```

The estimates for the covariate effects are given in the fixed effects parameters. Offset terms can be included by setting 'fixed=TRUE' for that covariate in the fixed effects parameters. All covariates that are named in the model formula are also present in the data stored in the staRVe_model object.

```
bird_model<- prepare_staRVe_model(
  cnt ~ I(abs(temp))+poly(elev,2) + time(year),
  bird_survey,
  distribution = "poisson",
```

```
  fit = T
)
```

More complex covariate effects such as interaction terms, transformations, and polynomial functions can be included in the model formula.

```
fixed_effects(bird_model)
#> $cnt
#>                              par          se fixed
#> I(abs(temp))      0.004984383 0.04346838 FALSE
#> poly(elev, 2)1    0.718185714 0.50600637 FALSE
#> poly(elev, 2)2 -0.295383191 0.49307979 FALSE
dat(bird_model)
#> Simple feature collection with 404 features and 4 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.23445 ymin: 36.11935 xmax:
   -89.2478 ymax: 40.47128
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>           temp         elev cnt year
   geom
#> 422  1.0958843 -1.70221026   1 2005 POINT (-95.23445
   40.46577)
#> 417  2.1010647  1.34828796   7 2005 POINT (-95.03312
   40.02546)
#> 402  0.7861318  0.39185501   5 2005 POINT (-94.66548
   39.65398)
#> 415  0.9132973 -0.89223203   7 2005 POINT (-94.50103
   39.39702)
#> 416  0.4733139  0.34172737   3 2005 POINT (-94.14222
   40.10026)
```

```
#> 392  0.8688053 -2.27217139   3 2005   POINT (-94.1192
   37.1404)
#> 420 -1.0913277 -0.34456416   4 2005 POINT (-94.10504
   38.66985)
#> 384  0.5817127  0.70189276  11 2005  POINT (-93.9101
   36.66086)
#> 391 -0.1119984 -0.08886507   7 2005 POINT (-93.85185
   37.77059)
#> 397 -0.1125152 -0.06566962   2 2005 POINT (-93.84015
   38.19498)
```

B.7.2 Point Predictions

```
bird_model<- prepare_staRVe_model(
  cnt ~ elev+temp + time(year),
  bird_survey,
  distribution = "poisson",
  fit = T
)

pred_points$elev<- rnorm(nrow(pred_points))
pred_points$temp<- rnorm(nrow(pred_points))
pred_points
#> Simple feature collection with 25 features and 3 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.1653 ymin: 36.1487 xmax:
   -89.41267 ymax: 40.43233
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>    year                       geom       elev       temp
#> 1  2012 POINT (-93.44969 37.48708) -1.3105843 -1.6585110
#> 2  2012  POINT (-92.76769 36.7374) -0.3320745 -0.6631657
#> 3  2012 POINT (-89.66307 36.49022)  2.0318708  0.4648332
```

```
#> 4  2012 POINT (-89.41267 36.74711)  0.9529447 -2.2180651
#> 5  2012  POINT (-95.1653 40.43233) -1.1302198 -2.1743246
#> 6  2013 POINT (-91.23448 36.59283) -0.4650467  0.9670220
#> 7  2013  POINT (-94.56619 36.1487)  1.3268975 -0.3559770
#> 8  2013 POINT (-92.09355 38.93433) -0.5159771  0.6772537
#> 9  2013 POINT (-89.75907 36.99413) -1.6592957  0.1270728
#> 10 2013 POINT (-89.79193 37.16898)  1.0610343  1.1181849
```

The code above creates an sf data.frame with covariate values at all prediction locations, in every year for which we want predictions. More locations or years can be supplied, only the locations/years that are needed will be used.

```
staRVe_predict(
  bird_model,
  pred_points
)
#> An object of class "staRVe_predictions"
#> Slot "predictions":
#> stars object with 2 dimensions and 6 attributes
#> attribute(s):
#>                    Min.    1st Qu.     Median       Mean    3
   rd Qu.        Max.
#> w             0.5619715 1.4077200 1.8651499 1.8303675
   2.2344000   3.2606559
#> w_se          0.2010697 0.2856713 0.4525358 0.4867302
   0.7098073   0.8741451
#> linear        0.4494473 1.4170646 1.7876059 1.8217022
   2.2005468   3.3362301
#> linear_se     0.2049178 0.2887172 0.4531565 0.4884664
   0.7099041   0.8743883
#> response      1.7621981 4.7266634 6.4909149 8.7339936
   11.0950042 28.7031925
```

```
#> response_se  0.8284835 1.9753874 3.1227205 3.3958283
   4.6894352  7.3238130
#> dimension(s):
#>           from to offset delta refsys point values
#> i            1 25      1     1     NA FALSE   NULL
#> variable     1  1     NA    NA     NA FALSE    cnt
#>
#> Slot "locations":
#> Simple feature collection with 25 features and 3 fields
#> Geometry type: POINT
#> Dimension:     XY
#> Bounding box:  xmin: -95.1653 ymin: 36.1487 xmax:
   -89.41267 ymax: 40.43233
#> Geodetic CRS:  WGS 84
#> First 10 features:
#>    year                       geom       elev       temp
#> 1  2012 POINT (-93.44969 37.48708) -1.3105843 -1.6585110
#> 2  2012  POINT (-92.76769 36.7374) -0.3320745 -0.6631657
#> 3  2012 POINT (-89.66307 36.49022)  2.0318708  0.4648332
#> 4  2012 POINT (-89.41267 36.74711)  0.9529447 -2.2180651
#> 5  2012  POINT (-95.1653 40.43233) -1.1302198 -2.1743246
#> 6  2013 POINT (-91.23448 36.59283) -0.4650467  0.9670220
#> 7  2013  POINT (-94.56619 36.1487)  1.3268975 -0.3559770
#> 8  2013 POINT (-92.09355 38.93433) -0.5159771  0.6772537
#> 9  2013 POINT (-89.75907 36.99413) -1.6592957  0.1270728
#> 10 2013 POINT (-89.79193 37.16898)  1.0610343  1.1181849
```

Covariates are given to the 'staRVe_predict' function as a third argument, separately from the sf data.frame holding the observation locations. The output is the same as predictions without covariates, but the covariates used are also included in the resulting sf data.frame.

B.7.3 Raster Predictions

```
pred_raster<- raster(
  bird_survey,
  nrow = 5,
  ncol = 5
)
raster_elev<- stack(lapply(min(bird_survey$year):2019,
   function(year) {
  band<- pred_raster
  band[]<- rnorm(ncell(band))
  names(band)<- paste0("T",year)
  return(band)
}))
raster_temp<- stack(lapply(min(bird_survey$year):2019,
   function(year) {
  band<- pred_raster
  band[]<- rnorm(ncell(band))
  names(band)<- paste0("T",year)
  return(band)
}))
pred_covar<- list(
  elev = raster_elev,
  temp = raster_temp
)
pred_covar
#> $elev
#> class       : RasterStack
#> dimensions  : 5, 5, 25, 15  (nrow, ncol, ncell, nlayers)
#> resolution  : 1.197331, 0.8703872  (x, y)
#> extent      : -95.23445, -89.24779, 36.11935, 40.47128  (
   xmin, xmax, ymin, ymax)
#> crs         : +proj=longlat +datum=WGS84 +no_defs
```

```
#> names      :      T2005,      T2006,      T2007,      T2008,
      T2009,      T2010,      T2011,      T2012,      T2013,
      T2014,      T2015,      T2016,      T2017,      T2018,
      T2019
#> min values : -2.015260, -1.155031, -3.418857, -2.062440,
   -1.518286, -1.957272, -2.600349, -1.742927, -2.254533,
   -1.757465, -2.662935, -1.916323, -1.714059, -1.912300,
   -2.004042
#> max values :  1.895613,  1.574167,  1.537536,  1.605621,
   1.845269,  1.249079,  1.440349,  1.403905,  1.843411,
   2.197493,  1.855871,  2.438909,  1.775691,  2.286333,
   2.742359
#>
#>
#> $temp
#> class      : RasterStack
#> dimensions : 5, 5, 25, 15  (nrow, ncol, ncell, nlayers)
#> resolution : 1.197331, 0.8703872  (x, y)
#> extent     : -95.23445, -89.24779, 36.11935, 40.47128  (
   xmin, xmax, ymin, ymax)
#> crs        : +proj=longlat +datum=WGS84 +no_defs
#> names      :      T2005,      T2006,      T2007,      T2008,
      T2009,      T2010,      T2011,      T2012,      T2013,
      T2014,      T2015,      T2016,      T2017,      T2018,
      T2019
#> min values : -2.211060, -2.030086, -2.222009, -2.118209,
   -1.847952, -2.085452, -2.822138, -3.701192, -1.474486,
   -2.303146, -2.639094, -2.298241, -1.420547, -1.350512,
   -2.093928
```

```
#> max values :  2.219493,  1.490701,  2.165628,  2.040302,
   1.842104,  1.729857,  1.938088,  1.379020,  2.884803,
   2.630402,  1.032724,  1.767252,  1.460482,  2.863586,
   2.086351
```

Covariates for raster predictions need to be supplied in a specific format: a list with a named element for each covariate. The elements need to have the same name as the covariate. Each covariate element needs to be a raster with the same geometry (dimensions, resolution, extent, coordinate reference system) as the raster template for the prediction locations. Each layer of the covariate raster must correspond to a single year and be named "T####" where #### is replaced with the year. The leading "T" in each layer name is necessary, and needs to specifically be a capitol "T". For example, the layer "T2005" gives the covariate values for the year 2005. Extra years of covariate values can be supplied, the 'staRVe_predict' function will only use the years for which predictions are made.

```
staRVe_predict(
  bird_model,
  pred_raster,
  pred_covar,
  time = min(bird_survey$year):2019
)
#> stars object with 4 dimensions and 8 attributes
#> attribute(s):
#>                          Min.     1st Qu.        Median
   Mean     3rd Qu.
#> w              -0.04336930   1.2066847   1.832483172
   1.73390918   2.2751983
#> w_se            0.23212793   0.3772480   0.451334089
   0.53356075   0.6358433
#> linear         -0.04674833   1.2116741   1.837925553
   1.73221998   2.2792946
```

```
#> linear_se      0.23233494   0.3781776   0.453049988
   0.53501874   0.6363145
#> response       1.11659634   4.0766160   7.268342018
   8.12886303  11.0473781
#> response_se    0.60197806   1.8885849   3.203611374
   3.77791871   4.8931177
#> elev          -3.41885710  -0.5830220  -0.006813093
  -0.01892806   0.6426037
#> temp          -3.70119214  -0.7568049  -0.005274774
  -0.03041850   0.6593663
#>                      Max.
#> w                3.339845
#> w_se             1.178643
#> linear           3.323359
#> linear_se        1.180056
#> response        29.652344
#> response_se     16.176172
#> elev             2.742359
#> temp             2.884803
#> dimension(s):
#>              from to    offset      delta refsys point
   values x/y
#> x               1  5 -95.2344    1.19733 WGS 84 FALSE
   NULL [x]
#> y               1  5  40.4713 -0.870387 WGS 84 FALSE
   NULL [y]
#> year            1 15      2005          1     NA    NA
   2005,...,2019
#> variable        1  1        NA         NA     NA    NA
              cnt
```

Covariates are supplied to 'staRVe_predict' the same way as they were for point

predictions. The raster list for the covariates is supplied as a third argument. The output is the same as raster predictions without covariates, but the resulting raster for each year has an additional layer containing the covariate values.

B.8 Options

B.8.1 Formula Syntax

A model formula specifies the response variable, the covariate effects, the temporal structure, and the spatial structure. Put together a model formula should have this general format:

response ~ covariates + **sample**.size(...) + **time**(...) + space(...)

Some options to the individual terms in the above model formula can be specified, as described below.

- **covariates**: Covariate effects can be specified as they would be in a typical call to 'lm' or 'glm'. Can be omitted from the formula if no covariates effects are desired.
- **sample.size(...)**: Used only for specific response distributions, for example the binomial distribution. A single variable name should be supplied. Can be omitted from the formula, and sample sizes are assumed to be equal to 1 for all observations.
- **time(...,type=)**: The name of the column holding the time information should be given. Valid options for the type option are:
  - **independent**: Each year in the model is assumed to be independent. The 'ar1' time parameter is fixed at 0.
  - **ar1** (default): An autoregressive order 1 model is used. The 'ar1' time parameter is freely estimated.
  - **rw**: A random walk is used. The 'ar1' time parameter is fixed at 1.
- **space(...covariance=,nu=)**: The name of the covariance function should be given to the covariance argument. If using the 'matern' covariance function, supplying a value to the nu option fixes the smoothness parameter nu at that

value. Can be omitted from the formula, in which case the default exponential covariance function will be used.

B.8.2 Response Distribution

```
get_staRVe_distributions("distribution")
#>          distribution          distribution
    distribution
#>            "gaussian"              "poisson"   "negative
    binomial"
#>          distribution          distribution
    distribution
#>           "bernoulli"               "gamma"          "
    lognormal"
#>          distribution          distribution
    distribution
#>            "binomial" "atLeastOneBinomial"               "
    compois"
#>          distribution
#>              "tweedie"
```

- **gaussian**: Used for continuous data. Random effects and covariates determine the mean.
  - Parameters
    - **sd**: standard deviation of response variable.
  - Default link: identity
- **gamma**: Models strictly positive continuous data. Random effects and covariates determine the mean.
  - Parameters
    - **sd**: standard deviation of response variable.
  - Default link: log
- **lognormal**: Models strictly positive continuous data. Random effects and covariates determine the mean.
  - Parameters

    - ∗ **sd**: standard deviation of the log of response variable.
  - Default link: log
- **tweedie**: Models non-negative continuous data with a point mass at zero. Random effects and covariates determine the mean. The relationship between the mean and variance is given by $Var(Y) = \sigma^2\mu^p$.
  - Parameters
    - ∗ **dispersion**: The parameter σ^2 above.
    - ∗ **power**: The parameter p above.
  - Default link: log
- **poisson**: Models non-negative integer data. Random effects and covariates determine the intensity/mean.
  - Parameters
    - ∗ N/A
  - Default link: log
- **negative binomial**: Model non-negative integer data with possible overdispersion. Random effects and covariates determine the mean.
  - Parameters
    - ∗ **overdispersion**: Values >1 correspond to overdispersion.
  - Default link: log
- **compois**: Models non-negative integer data with possible over- or underdispersion. Random effects and covariates determine the mean.
  - Parameters
    - ∗ **dispersion**: Values <1 correspond to underdispersion. Values >1 correspond to overdispersion
  - Default link: log
- **bernoulli**: Models binary data. Random effects and covariates determine the probability of success.
  - Parameters
    - ∗ N/A
  - Default link: logit
- **binomial**: Models the number of successes in k IID Bernoulli trials. Random effects and covariates determine the probability of success in a single trial. The

number of trials k is specified through the 'sample.size' term in the model formula.

  - Parameters
    * N/A
  - Default link: logit

- **atLeastOneBinomial**: Models the probability of observing at least one success in k IID Bernoulli trials. Random effects and covariates determine the probability of success in a single trial. The number of trials k is specified through the 'sample.size' term in the model formula.
  - Parameters
    * N/A
  - Default link: logit

B.8.3 Link Functions

```
get_staRVe_distributions("link")
#>        link        link        link
#> "identity"       "log"     "logit"
```

- **identity**
  - Range (-∞,∞)
- **log**:
  - Range [0,∞)
- **logit**:
  - Range [0,1]

B.8.4 Spatial Covariance Function

All covariance functions currently available are special cases of the Matérn covariance function.

```
get_staRVe_distributions("covariance")
#>      covariance      covariance      covariance      covariance
#> "exponential"      "gaussian"         "matern"       "matern32"
```

- **exponential**
  - Smoothness parameter nu = 0.5
- **matern32**
  - Smoothness parameter nu = 1.5
- **gaussian**
  - Limit of Matérn covariance function as nu$\rightarrow \infty$
- **matern**
  - General covariance function with a freely estimated smoothness parameter nu. Evaluation is very slow compared to the named special cases.

B.8.5 Holding Parameters Fixed

```
parameters(bird_model)
#> An object of class "staRVe_parameters"
#> Slot "covariance_function":
#> [1] "exponential"
#>
#> Slot "spatial_parameters":
#> $cnt
#>                par          se fixed
#> sd       0.5522385 0.04145799 FALSE
#> range 80.0000000 0.00000000  TRUE
#> nu       0.5000000 0.00000000  TRUE
#>
#>
#> Slot "time_parameters":
#> $cnt
#>             par          se fixed
#> mu   2.0066523 0.3639902 FALSE
#> ar1  0.9235844 0.0316828 FALSE
#> sd   0.3816032 0.2212175 FALSE
#>
#>
```

```
#> Slot "response_distribution":
#> [1] "poisson"
#>
#> Slot "response_parameters":
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
#>
#>
#> Slot "link_function":
#> [1] "log"
#>
#> Slot "fixed_effects":
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)

time_parameters(bird_model)$cnt["ar1","par"]<- 0
time_parameters(bird_model)$cnt["ar1","fixed"]<- T
bird_model<- staRVe_fit(bird_model,silent=T)

parameters(bird_model)
#> An object of class "staRVe_parameters"
#> Slot "covariance_function":
#> [1] "exponential"
#>
#> Slot "spatial_parameters":
#> $cnt
#>              par          se fixed
#> sd      1.112652 0.05371455 FALSE
#> range  80.000000 0.00000000  TRUE
#> nu      0.500000 0.00000000  TRUE
```

```
#>
#>
#> Slot "time_parameters":
#> $cnt
#>                 par           se fixed
#> mu   2.019741e+00 0.08374696 FALSE
#> ar1  0.000000e+00 0.00000000  TRUE
#> sd   5.548022e-05 0.07227968 FALSE
#>
#>
#> Slot "response_distribution":
#> [1] "poisson"
#>
#> Slot "response_parameters":
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
#>
#>
#> Slot "link_function":
#> [1] "log"
#>
#> Slot "fixed_effects":
#> $cnt
#> [1] par    se     fixed
#> <0 rows> (or 0-length row.names)
```

You can hold parameters fixed during model fitting. Instead of being estimated the parameter value held in the staRVe model will be assumed "true". In the example above the ar1 parameter is held fixed at 0, setting up each year in the model to be independent of the other years. You can set as many parameters as you want to be held fixed, however the random effects cannot be fixed to particular values. A model

with all parameters held constant can still be fitted with the ‘staRVe_fit’ function, in which case the time effects and random effects will be estimated along with their standard errors.

Appendix C

Copyright Release

Chapter 2 was fully published under the following information:

Lawler, E., Whoriskey, K., Aeberhard, W. H., Field, C., and Mills Flemming, J., The Conditionally Autoregressive Hidden Markov model (CarHMM): Inferring Behavioural States from Animal Tracking Data Exhibiting Conditional Autocorrelation. *Journal of Agricultural, Biological and Environmental Statistics*, 24(4):651–668., published 2019, Springer Nature.

The *Journal of Agricultural, Biological and Environmental Statistics* is a Springer journal whose author re-use policy for theses is at the time of writing (https://www.springer.com/gp/rights-permissions/obtaining-permissions/882):

"Authors have the right to reuse their article's Version of Record, in whole or in part, in their own thesis. Additionally, they may reproduce and make available their thesis, including Springer Nature content, as required by their awarding academic institution. Authors must properly cite the published article in their thesis according to current citation standards. Material from: 'AUTHOR, TITLE, JOURNAL TITLE, published [YEAR], [publisher - as it appears on our copyright page]'"