Vignette for package JointSurvey

Jason Matthiopoulos

2021-12-03

Abstract

This is a demonstration of the `JointSurvey` package, written as a controlled environment in which to conduct simulated analyses of survey data for seabird populations. The first main part of the package deals with the design and inherrent biases of line transects. Multiple surveys can be specified and executed over real sea-scapes. These data can be pooled for joint analysis. Hence, the second part of the package deals with joint estimation and spatial prediction. For this, we showcase three different modelling approaches of increasing complexity (Inhomogeneous Point Process, IPP with overdispersion and IPP with spatial structure) and compare their preformance under different biological scenarios. We specify this to simulated distributions mimicking the properties of four seabirds in the UK, to illustrate multispecies comparisons. The package provides functionality for generating simulated data within a broad space of survey designs, hence facilitating experimentation on the sensitivity of results on different aspects of design. Although the package is developed for the purposes of the scoping exercise commissioned by Marine Scotland, its modelling functionality can be applied directly to real data from marine and terrestrial systems.

Contents

1 Introduction

Ideally, the data used for robust spatial prediction of species distributions should be both high-resolution and spatially expansive. However, logistic trade-offs between spatiotemporal extent and resolution mean that such in-depth and geographically broad data are rarely available in practice. Instead, researchers need to piece together data from different places, times, or survey methods. Such integration presents several challenges, but it also offers considerable opportunities. For example, data from different places and times, can allow us to increase the spatial extent of our maps, and our historical reconstructions, but importantly, they allow us to model the focal species under distinct and different circumstances, hence increasing the transferability of our model predictions. Also, if the survey designs are different (e.g. different resolutions and different field methodologies), simultaneous analysis has the potential to allow the combined survey data to effectively ground-truth (i.e. calibrate) each-other. The `JointSurvey` package demonstrates how analyses of such data can be conducted in an integrated way. The package provides functionality for generating simulated data within a broad space of survey designs, hence facilitating experimentation on the sensitivity of results on different aspects of design. It also provides several pre-defined models for the analysis of pooled multiplatform data. The objective of this vignette is to demonstrate the usage of the data and commands introduced in the package, as part of a comprehensive workflow of simulation and analysis. As a resource, it can be read in conjuction with the

package's R manual, which presents more details about the functions in alphabetical order. The presentation is split into three parts. Part one (section 2 and section 3), deals with survey design and simulated data collection. Part two (sections 4 and 5), presents the key stages of the analysis, mainly model fitting and spatial prediction. The implementations of elementary and advanced models are reviewed. Finally, the third part (section 6) explores comparative performance of methods under different realistic scenarios for species and survey designs.

2 Creating an underlying species distribution

2.1 Poisson Point Process models

The current thinking in this area (Warton, Shepherd, and Others 2010; Aarts, Fieberg, and Matthiopoulos 2011; Renner and Warton 2013; Renner et al. 2015) is to formulate SDMs as Inhomogeneous Poisson Process models (IPPs). A Poisson point process is a stochastic process in space, or time which models the occurrence of single events (e.g. the occurrence of an individual bird) in continuous space. At its core, the IPP requires the definition of an *intensity* function $\lambda(\mathbf{s})$ that expresses the rate of occurrence of the study species per unit of space or time, at any given location $\mathbf{s}$ in a map. The intensity function must be non-negative and is most frequently expressed as an exponential function of covariates.

$$\lambda(\mathbf{s}) = exp(a_0 + a_1 u_1(\mathbf{s}) + a_2 u_2(\mathbf{s}) + a_3 u_3(\mathbf{s})...) \tag{1}$$

The expression inside the exponential is the *predictor* of the intensity function, based on a set of arbitrary parameter values $a_0, a_1, a_2, a_3, ...$ For survey data, the intercept (a_0) has an important biological interpretation in these models because $exp(a_0)$ determines the absolute abundance of the species in space. Note that a_0 need not be positive, indeed it will be negative when densities are expected to be lower than 1 individual per unit space. To develop this model for seabirds, it is necessary to first consider accessibility from colony locations and ranging constraints. On top of this background it is then necessary to incorporate environmental covariates to usage. For the particular application of survey analysis in colonial animals, independent estimates of population size may be available from colony counts. We may therefore choose to be more specific about the intercept in this expression. Specifically, if we considered the entire predictor, excluding the intercept, as relative usage of space by a population, then normalising that expression and scaling up to population size, would give us the absolute usage

$$\lambda(\mathbf{s}) = N \frac{exp(\sum a_j u_j(\mathbf{s}))}{\int exp(\sum a_j u_j(\mathbf{s})) d\mathbf{s}} \tag{2}$$

The combination of eqs (1) and (2) implies that, in general,

$$a_0 = \log N - log \int exp\left(\sum a_j u_j(\mathbf{s})\right) d\mathbf{s} \tag{3}$$

It is important to note that this is a population-specific intercept for usage. In the case of usage contributed by birds from different colonies (each with a different size N_i) it is important to account for that. If we can assume that the parameters in the linear predictor are the same between colonies and that the colonies have unconstrained access around them, then the integral in the above expression can be assumed colony-independent. However, this will unfortunately not, in general, be the case.

Therefore, the integral in this expression causes problems for parameter estimation. Two approaches present themselves: Either let a0 be a free parameter with a $logN_i$ offset to ensure that a0 can be assumed to be shared between colonies, or incorporate some numerical approximation of the integral into the estimation. The latter option would be required only if the detectability properties of all survey methods were unknown and if the value of N was known from independent surveys.

2.2 Colony locations & null usage

A particularly important aspect driving the distribution of central place foragers is the constraint of accessibility from their central place. In colonial central place foragers this issue is scaled up by, often, large numbers of animals that are based at the same colony. This issue, was originally highlighted by (Matthiopoulos 2003) who discussed the principle of *null usage*, the expected usage distribution of a central place forager in the absence of habitat selection. The approach was applied to colonial pinnipeds as a basis from which to build more elaborate distribution models (Matthiopoulos et al. 2004). Since then, it has been proposed by (Thaxter et al. 2012) as a first approximation to highlighting marine areas of importance for seabirds. Broadly, the colony characteristics can be specified in terms of their locations and associated population sizes. The toy data set `Colonies` contains a small set of three colonies, for use with this example

```
Colonies
```

```
##    x  y  Size
## 1 35 40  2000
## 2 91 60  5000
## 3 50 51 10000
```

```
J<-length(Colonies[,3]) # Number of colonies in the data
```

If we consider a single colony in isolation, we can describe the decay in expected usage as a function of distance from the colony location. Spatial layers of distance can be obtained by using the function `colonyDist` which generates an array of Euclidean distance maps from a vector of colony locations. In this example, the `x` and `y` coordinates are extracted from the first and second columns of the `Colonies` data frame. The matrix `land` is provided as a template for the distance maps to be generated and the option `masked` declares that land is to be used as a mask, setting the distances of landlocked coordinates to an arbitrarily large value (100000 units of length, in this example).

```
# Calculate array of distance maps.
dis<-colonyDist(Colonies[,1], Colonies[,2], land, masked=100000)
```

If $d_i(\mathbf{s}) = |\mathbf{s}_i - \mathbf{s}|$ is the distance between the location $\mathbf{s}_i$ of the i^{th} colony and a given point $\mathbf{s}$ at sea, then the intensity under a null model of usage can be written as

$$\lambda(\mathbf{s}) = exp(a_0 - c_0 d_i(\mathbf{s})) \tag{4}$$

The key to this formulation is the rate of decay c_0 (where $c_0 > 0$). It is useful to think of this as a distinguishing characteristic of a species, related to its foraging range (so that far-ranging species will be characterised by lower values of c_0). Equation 4 is defined as a function of distance from the colony (Figure 1). Hence, even though it can be used to generate expected usage of a cell in a map, it cannot be interpreted as the amount of usage that should be expected at a particular distance from the colony. For example, setting $c_0 = 0$ gives a function that is unresponsive to distance, and therefore predicts uniform usage of space. However, looking at cells along concentric circles with ever-increasing radii, will naturally give increasing overall usage as a function of distance.

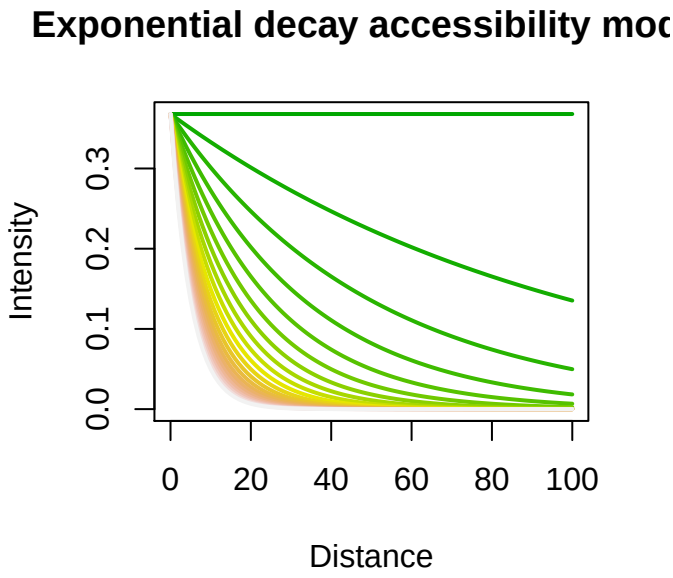


Figure 1: The basic accessibility model with different constraints. The unconstrained model (in dark green) is not affected by distance. As c_0 increases, the relationship between distance and expected usage becomes steeper.

Formulations of the null model will need to account for physical and biological phenomena that contract or expand the range of a colony. We present the most important of those here.

Possible extension 1: *The coefficient c_0 will depend on the availability of points at a particular distance from a colony. In open, unobstructed space, every colony would have the*

same relative availability of points at a particular distance. However, different colonies are differentially affected by the morphology of the coastline around them. Explicitly acounting for the relative availability of different distances, and also accounting for non-Euclidean distances could be an area of future improvement. This can, for example, be done by expressing c_0 as a linear function of the mean and higher moments of the distribution of distances ■

The null model above can be extended with further biological realism to take account of different forms of competition. To begin with, the range of the i^{th} colony could be expressed as an increasing function of its size due to intra-colony competition (Lewis et al. 2001). We could use an interaction term in the rate model to achieve this effect

$$\lambda(\mathbf{s}) = exp(a_0 - c_0 d_i(\mathbf{s}) + c_1 d_i(\mathbf{s}) N_i) \tag{5}$$

Biologically, the coefficient c_1 can be interpreted as the reduction in the steepness of decay of usage from the colony, caused by each additional conspecific operating from that colony (Figure 2).

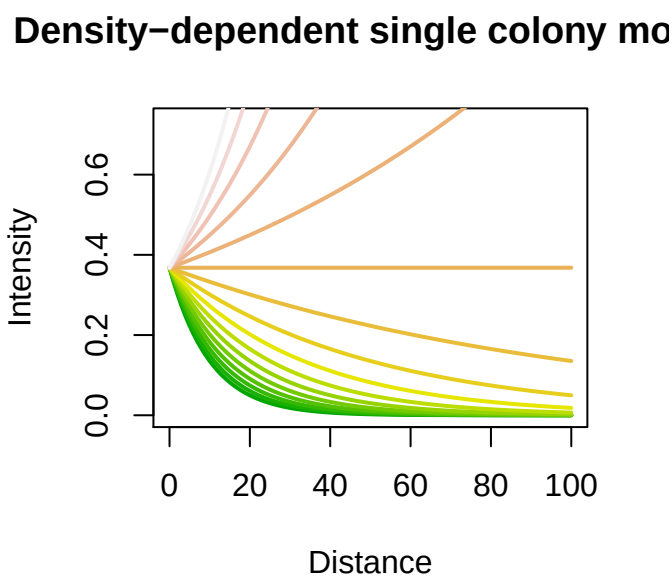


Figure 2: The accessibility model with different effects of intracolony density. A model with no effect of density ($c_1 = 0$, in dark green) gives a colony with a characteristic range (defined by the value of c_0) . The effect of density dependence is to push individuals away from the colony. The behaviour of this function can be somewhat problematic, because at high values of c_1 it becomes a positive relationship with distance, implying that individuals lose the central-place constraint.

For values $c_1 > c_0$, this formulation runs the risk of yielding intensity functions that increase with distance. This does not make sense, because it results in animals that are no longer central-place foragers (see e.g. grey curve in Figure 2). We can modify Equation 5 so that

the effect of colony density is eliminated at very large distances. One possible way to achieve this is

$$\lambda(\mathbf{s}) = exp\left(a_0 - c_0 d_i(\mathbf{s}) + \frac{c_1 d_i(\mathbf{s}) N_i}{1 + c_2 d_i(\mathbf{s})}\right) \tag{6}$$

This approach to intra-colony competition introduces a non-linearity in the model because of the coefficient c_2, however it has considerable biological appeal. The interpetation of the coefficient c_2 is a little more involved: It is the inverse of the distance from the colony at which the effect of intra-colony competition drops to 50% of the strength that it has at the colony.

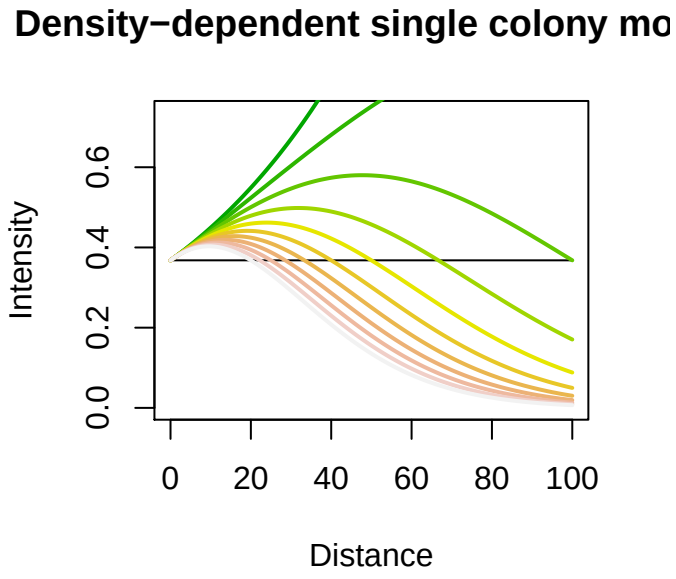


Figure 3: The accessibility model with density dependence and nonlinear constraint to range. A model with no eventual constraint ($c_2 = 0$, in dark green) gives a colony whose members are expected to be increasingly far away. As values of c_2 increase, the density dependent term is eliminated with increasing distance.

Possible extension 2: *Such nonlinear expressions are both realistic and economical, but remain heuristic. More direct approaches to competition are possible, by treating conspecifc usage from the same colony as an autocovariate. This considerably more advanced approach would be entirely defensible biologically, but very demanding in implementation. It would require joint modelling of all points in space to allow for (negative) autocorrelations between local density and nearby conspecific density.*■

Another feature of importance for the distribution of colonial animals is inter-colony competition, which is expected to increase in places that are more readily accessible from other colonies, as well as the size of those competing colonies (Wakefield et al. 2013). Biologically,

this would result in birds from colonies that experience competition ranging further away from home. Here, we extend the basic model of accessibility to account for these effects by using the following intensity function

$$\lambda(\mathbf{s}) = exp\left(a_0 - c_0 d_i(\mathbf{s}) + \frac{c_1 d_i(\mathbf{s}) N_i}{1 + c_2 d_i(\mathbf{s})} + \sum_{j \neq i} \frac{c_3 d_i(\mathbf{s}) N_j}{1 + c_4 d_j(\mathbf{s})}\right) \tag{7}$$

The logic behind this expression is that inter-colony, intra-specific competition can affect the range of a focal colony in a similar way as intra-colony competition, but its effect diminishes depending on how far they are from the point of interest. This formulation has the right kind of relationship that might be expected from competing conspecifics from different colonies. Specifically, large competing colonies with N_j members, that are closer to the point of interest than the focal colony will tend to make the point of interest less used by the members of the focal colony i. Indeed, if we can argue that conspecifics from different colonies compete in identical ways (allowing for the effect of distance), then we can simplify the expression considerably:

$$\lambda(\mathbf{s}) = exp\left(a_0 - c_0 d_i(\mathbf{s}) + \sum_{j} \frac{c_1 d_i(\mathbf{s}) N_j}{1 + c_2 d_j(\mathbf{s})}\right) \tag{8}$$

Possible extension 3: *A more direct approach to inter-colony competition would require us to treat usage from other colonies as a covariate in a simultaneous regression for all colonies. This approach would require joint modelling of all colonies to allow for negative correlations in their spatial use.*■

Similar approaches to Equation 8 can be developed to account for competition with other species. For example, different colonies (j) of other species (k) could be incorporated into our basic model as follows:

$$\lambda(\mathbf{s}) = exp\left(a_0 - c_0 d_i(\mathbf{s}) + \sum_{k} \sum_{j} \frac{c_{1,k} d_i(\mathbf{s}) N_j}{1 + c_{2,k} d_j(\mathbf{s})}\right) \tag{9}$$

Possible extension 4: *Once again,* Equation 9 *is an indirect approach to modelling competition. The ideal (but computationally prohibitive) approach would be to treat other species as covariates within a simultaneous (stacked) regression* (Distler et al. 2015).■

Putting these ideas together, and by supplying a set of arbitrary values to the c coefficients of Equation 8, we can create a map of expected null usage (Figure 4). This is achieved by the function `colambda` which we will generalise to the case of environmental covariates in the next section.

```
# Specify distance-decay rate parameters
a0<--1
c0<--0.02
```

```
c1<-0.000003
c2<-0.025
Dist<-data.frame(c(dis[1,,]),c(dis[2,,]),c(dis[3,,]))
nr<-length(Dist[,1]) # Total number of cells in map
Xpred<-data.frame("Intercept"=rep(1,nr))

null<-colambda(Colonies, acoef=c(a0), ccoef=c(c0,c1,c2),
               Xpred, Dist, mask=land)

# Colour palette specification
cols <- brewer.pal(3, "BrBG")
pal <- colorRampPalette(cols)
image(x,y,null, col=pal(15), main="Null map of usage")
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
# Plots the locations of the colonies
points(Colonies[,1], Colonies[,2], pch=16, col="Brown", cex=2)
```

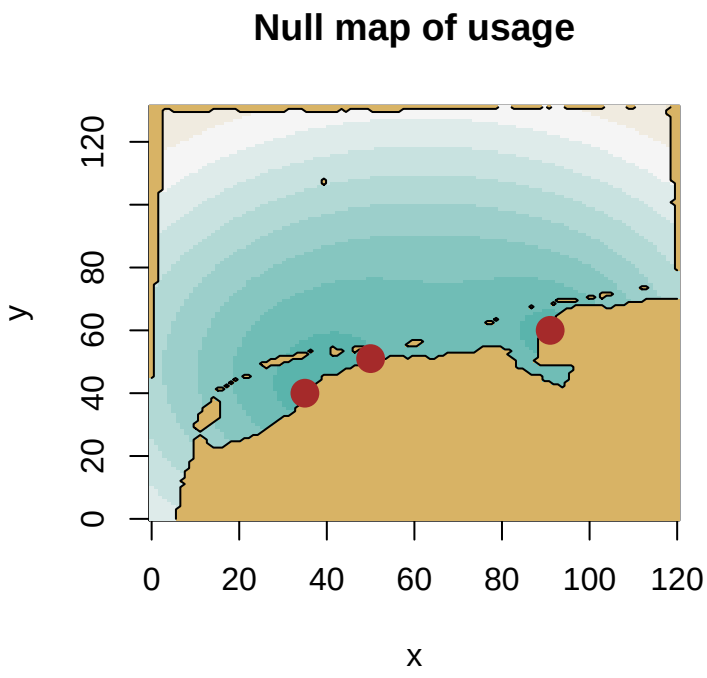


Figure 4: The combined null usage from three colonies, weighted by their size, in numbers of birds

2.3 Environmental layers

The package comes with three example layers (these are rasters transformed to matrices), corresponding to areas of land and sea (`land`), bathymentry (`u1`) and sediment composition (`u2`). They can be accessed and plotted as images (Figure 5)

```
# Explanatory variables can be centred to improve model convergence later
u1<-u1-mean(u1*land) # Centres bathymetry to zero
u2<-u2-mean(u2*land) # Centres sediment to zero

# Colour palette specification
cols <- brewer.pal(3, "BrBG")
pal <- colorRampPalette(cols)

# Plotting (sea-only values) and coastline
par(mfrow=c(1,2))
image(x,y,u1/land, col=pal(20), main="Bathymetry")
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
image(x,y,u2/land, col=pal(20), main="Sediment")
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
```

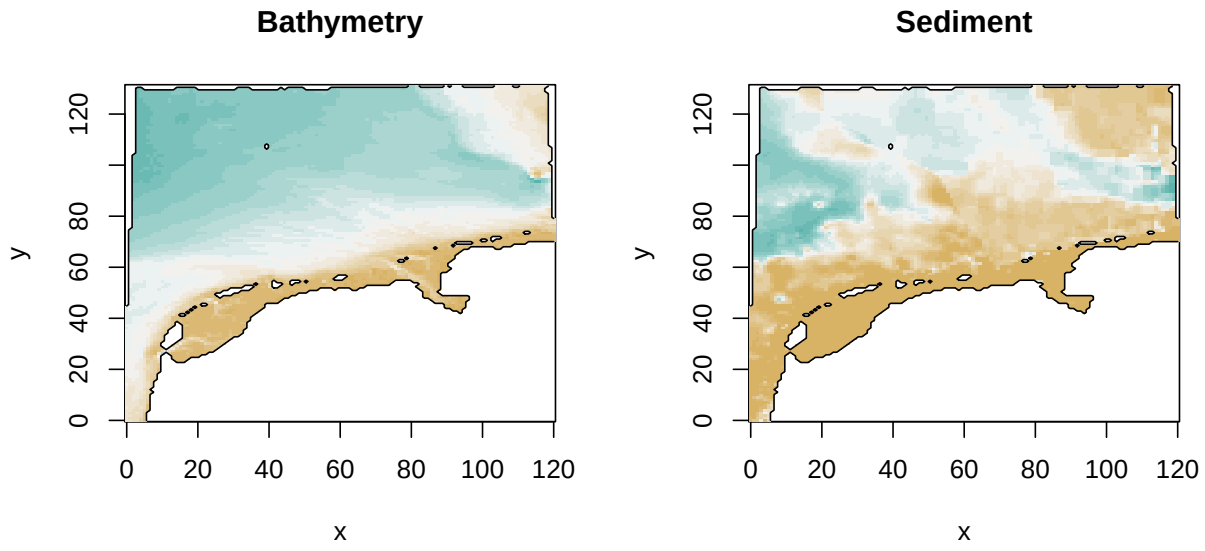


Figure 5: The two environmental layers used as covariates for the simulation and the analysis

```
par(mfrow=c(1,1))
```

2.4 A species distribution

We can create and plot an example of an intensity function by using the function `colambda`. This combines the covariates and the null usage (based on the ranging model above). The intesity function can then be used to generate a population distribution, as a realisation of a stochastic process. The default is Poisson, but we may consider overdispersed processes around lambda. The function `synthDistr` can perform this task. It requires a map of the intensity function (`lambda`, in our example) and an overdispersion parameter (set to one by default, representing a Poisson process. If values >1 are provided, then a negative binomial process is used).

```
# Specification of intensity function
a1<- -0.01   # Bathymetry coefficient
a2<-0.08    # Sediment coefficient

Xpred<-data.frame("Intercept"=rep(1,nr),"Depth"=c(u1), "Sediment"=c(u2))
Dist<-data.frame(c(dis[1,,]),c(dis[2,,]),c(dis[3,,]))
lambda<-colambda(Colonies, acoef=c(a0,a1,a2), ccoef=c(c0,c1,c2),
                 Xpred, Dist, mask=land)

# Realisation of animal distribution
od<-1
distribution<-synthDistr(lambda+10^-100, q=od) # Implements no overdispersion

# ------ Plotting ------
par(mfrow=c(1,2))
# Plotting of intensity function
image(x,y,lambda/land, col=pal(50),
      main="Species distribution intensity function")
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
points(Colonies[,1], Colonies[,2], pch=16, col="Brown", cex=2)
# Plotting of actual distribution of animals
image(x,y,distribution/land, col=pal(50), main="Species distribution")
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
points(Colonies[,1], Colonies[,2], pch=16, col="Brown", cex=2)
```

```
par(mfrow=c(1,1))
```

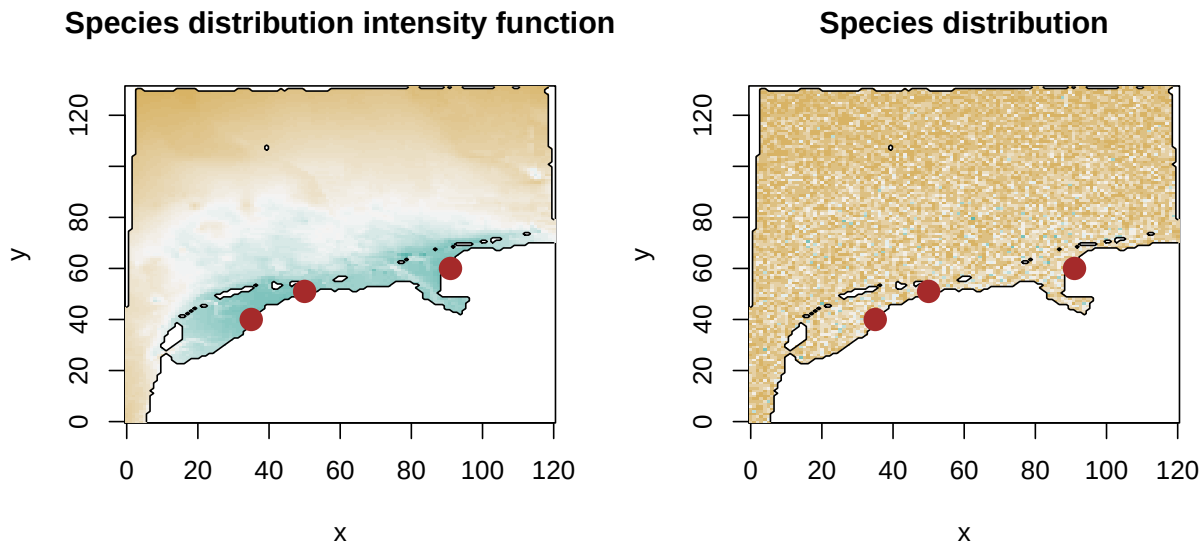


Figure 6: Comparison of the underlying intensity and a realisation of the distribution of the species in space. Notice that stochasticity on the right (particularly for small populations) can obscure the patterns visible on the left.

3 Collection of survey data

3.1 Transect surveys: Span characteristics

Each survey has a distinct list of survey design characteristics. These are divided into the two categories of **span** and **detectability**. Span characteristics are the following:

1. Extent of rectangular box enclosing the survey, provided as two vectors of `x`$= (x_m, x_M)$ and `y`$= (y_m, y_M)$ coordinate ranges
2. Spacing (l) of successive sampling locations along the transect (in units of length)
3. Spacing (L) between transects (in units of length)
4. Optionally, the orientation (ϕ) of line transects (in radians from the horizontal, `x` axis)
5. An optional raster `mask` (M provided as a matrix) to excude unsurveyed areas from the extent box

The transects for a single survey can be defined via the function `survey()`, by using the span characteristics. For example, the code below implements and plots a survey with transects spaced 5 length units apart and with one unit between observations, at a slight angle of about 29^o (Figure 7a). We can also use the `land` layer in the built-in data set of the `JointSurvey` package as a mask, to exclude areas that are not at sea (Figure 7b).

```
xm<-40;xM<-80;ym<-40;yM<-80 # Boundaries of survey
ll<-1 # Spacing between survey points
lL<-5 # Spacing between transects
fi<-0.5 # Orientation of transects

par(mfrow=c(1,2))
# Unmasked transects
pts<-survey(c(xm,xM),c(ym,yM),ll,lL,fi)
image(x,y,lambda/land, col=pal(10), main="a")
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
points(pts, cex=0.4, pch=16)

# Masked transects
pts<-survey(c(xm,xM),c(ym,yM),ll,lL,fi,land)
image(x,y,lambda/land, col=pal(10), main="b")
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
points(pts, cex=0.4, pch=16)
```

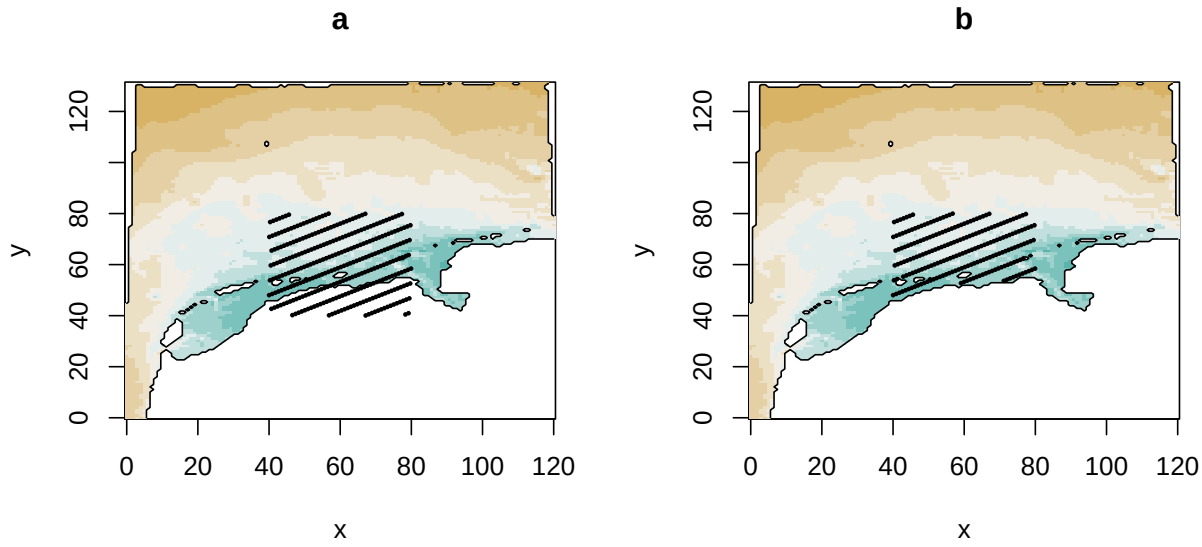


Figure 7: Realisation of a single survey, over both land and sea, and over sea alone (land-points removed).

```
par(mfrow=c(1,1))
```

Multiple surveys with different span characteristics can be generated and stored in separate data structures of sampling coordinates (here, `pts1, pts2, pts3`). For example, note that for the three surveys in Figure 8, the span design parameters are 3-element vectors:

```
# Design specification of thee surveys
# Boundaries of surveys
xm<-c(40,30,100);xM<-c(80,50,120)
ym<-c(40,20,100);yM<-c(80,100,120)
ll<-c(1,1,1) # Spacing between survey points
lL<-c(5,10,5) # Spacing between transects
fi<-c(0.5,0.1,2) # Orientation of transects

# Generation of the transect points for the three surveys
pts1<-survey(c(xm[1],xM[1]),c(ym[1],yM[1]),ll[1],lL[1],fi[1],land)
pts2<-survey(c(xm[2],xM[2]),c(ym[2],yM[2]),ll[2],lL[2],fi[2],land)
pts3<-survey(c(xm[3],xM[3]),c(ym[3],yM[3]),ll[3],lL[3],fi[3],land)

# Plotting of combined surveys
image(x,y,lambda/land, col=pal(10))
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
points(pts1, cex=0.4, pch=16)
points(pts2, cex=0.4, pch=3)
points(pts3, cex=0.4, pch=21)
```

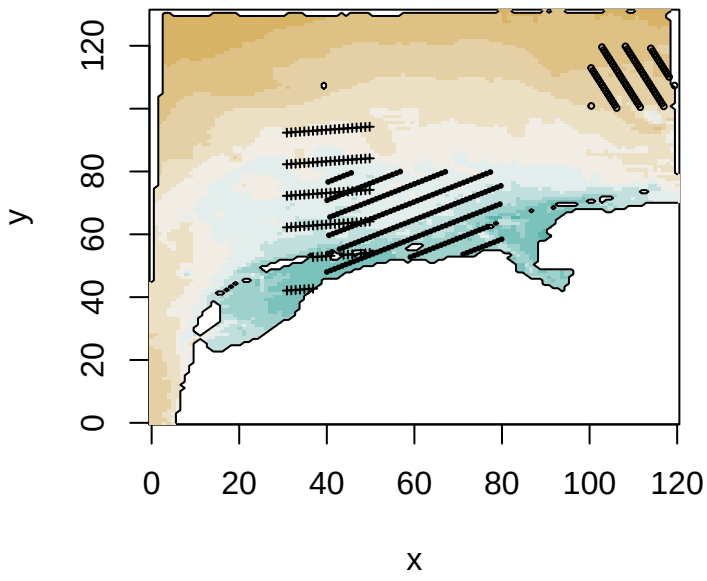


Figure 8: Combination of three surveys, each with its own design profile.

3.2 Transect surveys: Detectability characteristics

There are two characteristics reqired for detectability: The *effective detection distance* w and the *probability of detection at zero distance* p_0. Both of these ideas come from the distance sampling literature (Buckland et al. 2001). In this work, we have treated all transect surveys as strip transects, because an independent pre-analysis of data to estimate a detection function can always be used to derive an effective detection distance, allowing line transect surveys to be treated with strip transect methodology. This completes the design specification (say, T) of a survey. To recap, in this work a survey is completely specified by the following list of characteristics:

$$T = ((x_m, x_M), (y_m, y_M), l, L, \phi, M, w, p_0) \tag{10}$$

To generate some seabird abundance data under a particular transect survey design, the transect sampling points can be passed to the function `counts` together with the detectability characteristics of Equation 10. The results can be plotted as circles of variable radius on the map, to represent the "raw" data (Figure 9). In the example below, all three surveys designed in the previous section are conducted, each with its own detectability characteristics.

```
# Detection distances for each of the three surveys
w<-c(0.5,1,1.5)
# Detection probabilities at zero for each of the three surveys
p0<-c(0.9,0.8,0.2)
cnts1<-counts(pts1, distribution, w[1], ll[1], p0[1])
cnts2<-counts(pts2, distribution, w[2], ll[2], p0[2])
cnts3<-counts(pts3, distribution, w[3], ll[3], p0[3])

# Extract upper count limit for plotting
maxC<-max(cnts1,cnts2,cnts2)

image(x,y,lambda/land, col=pal(10))
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
col1<-brewer.pal(9, name="Reds")[9*cnts1/maxC] #Colour palette for survey 1
symbols(pts1, circles=cnts1, add=TRUE, inches=maxC/200,  fg=col1, bg=col1)
col2<-brewer.pal(9, name="Greens")[9*cnts1/maxC] #Colour palette for survey 2
symbols(pts2, circles=cnts2, add=TRUE, inches=maxC/200,  fg=col2, bg=col2)
col3<-brewer.pal(9, name="Blues")[9*cnts1/maxC] #Colour palette for survey 3
symbols(pts3, circles=cnts3, add=TRUE, inches=maxC/200,  fg=col3, bg=col3)
```

We can now collect the seabird data together into a pooled data frame (`data`) that can be used for analysis.

```
data<-data.frame(
  # x coordinates
```

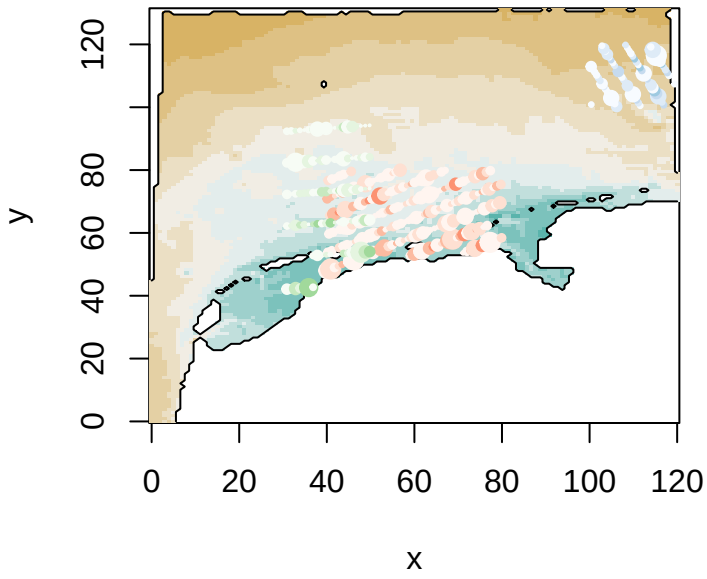


Figure 9: The successive locations along the transects (shown as coloured circles) lead to different counts of birds (indicated by the radius of each circle). Each survey is represented by a different colour group.

```
  "x"=c(pts1[,1],pts2[,1],pts3[,1]),
  # y coordinates
  "y"=c(pts1[,2],pts2[,2],pts3[,2]),
  # response data
  "counts"=c(cnts1,cnts2,cnts3),
  # Bathymetry values at survey locations
  "depth"=c(u1[round(pts1)],u1[round(pts2)],u1[round(pts3)]),
  # Sediment values at survey locations
  "sedm"=c(u2[round(pts1)],u2[round(pts2)],u2[round(pts3)]),
  # Originating survey for datum
  "sv"=c(rep(1,length(cnts1)),rep(2,length(cnts2)),rep(3,length(cnts3)))
)
n<-length(data[,1]) # Sample size of data
head(data)
```

```
##          x        y counts     depth       sedm sv
## 1 71.13541 53.63287      1 -17.86621 -0.9220512  1
## 2 72.01299 54.11230      2 -17.87621 -0.9220512  1
## 3 72.89057 54.59172      2 -16.83121 -0.9220512  1
## 4 73.76816 55.07115      4 -17.59371 -0.9220512  1
## 5 74.64574 55.55057      3 -17.50621 -0.9220512  1
## 6 75.52332 56.03000      3 -17.19852 -0.9220512  1
```

This concludes the simulation part of this demonstration. The next steps are now to try and retrieve the biological parameters from this pooled data frame.

4 Simple joint inference for the distribution of species from pooled survey data

In analyses of real data, this would usually be the starting point of the main processing (assuming that several stages of pre-processing that examine detection distances have been completed). The task here is to infer the coefficients associated with the distribution of the species and determine the overdispersion (if any) in the data. Also, some of the design parameters of different surveys (such as the detection probability at zero p_0) may be only partly known, so we could seek some refinement through the analysis of the values for those. Ideally however, the design parameters of the surveys should be known with high precision, because otherwise ignorance in these parameters will interfere with our ability to estimate baseline abundances, and may even bias our estimation of the species responses to different covariates.

4.1 An inhomogeneous Poisson model

We will use JAGS (Plummer and Others 2003) as an inferrential engine. We have developed the function `jointHSF` which implements and runs several different probabilistic models coded in JAGS. First, we will examine how to set up a call to `jointHSF` and demonstrate this with the simplest of its packaged models, the Inhomogeneous Poisson Point Process. We will pair that biological model with an observation model that exactly reflects the design parameters (i.e. span and detectability) of the 3 surveys conducted above. The model first requires a matrix `X` of covariate data containing transect points along the rows and distinct covariates running down the columns. The second requirement is for a vector `surv` that declares which survey each location comes from. Internally, this will allow the model to match the survey characteristics of detectability, with the observed count. `count`, a vector of seabird counts at each of the transect points, is the third required input. The call to the function `jointHSF` combines these data with the detectability parameters w, p_0 and the spacing between survey points l. These parameter values are used internally to scale the value of the spatial intensity function at the location of the survey point, into an expected count of animals within the effective area of detection.

```
# Coercion of a covariate data-frame into a matrix
X<-as.matrix(data[,4:5])
# Survey tags for each point in the data
surv<-data$sv
# Response data
count<-data$counts
# Distance data
dist<-matrix(0,n,J)
for(j in 1:J) dist[,j]<-dis[cbind(rep(j,n),round(data$x),round(data$y))]
# Colony sizes
N<-Colonies[,3]

# Model fitting
resultsIPP<-jointHSF(count, X, dist, surv, ll, w, p0, N,
                     model="IPP", ABS=c(1000,10000,5000))

# Extraction of summaries for posteriors
sumsIPP<-summary(resultsIPP)
```

The MCMC results and statistical summaries from the model fit are stored in the names `resultsIPP` and `sumsIPP` respectively.

4.2 Summaries and diagnostics

Here, we can pull together a comparison between the parameter values estimated from the above code (along with 95% credible intervals), together with the true values that were used in the simulation above.

Parameter	True values	2.5% limit	Median	97.5% limit
a_1	-0.01	-0.0199	-0.0103	0.001
a_2	0.08	-0.1272	0.0594	0.2333
c_0	-0.02	-0.0093	-0.0045	0
c_1	3×10^{-6}	3.61188×10^{-9}	5.05043×10^{-8}	9.83588×10^{-8}
c_2	0.025	0.0015461	0.0492535	0.0964961

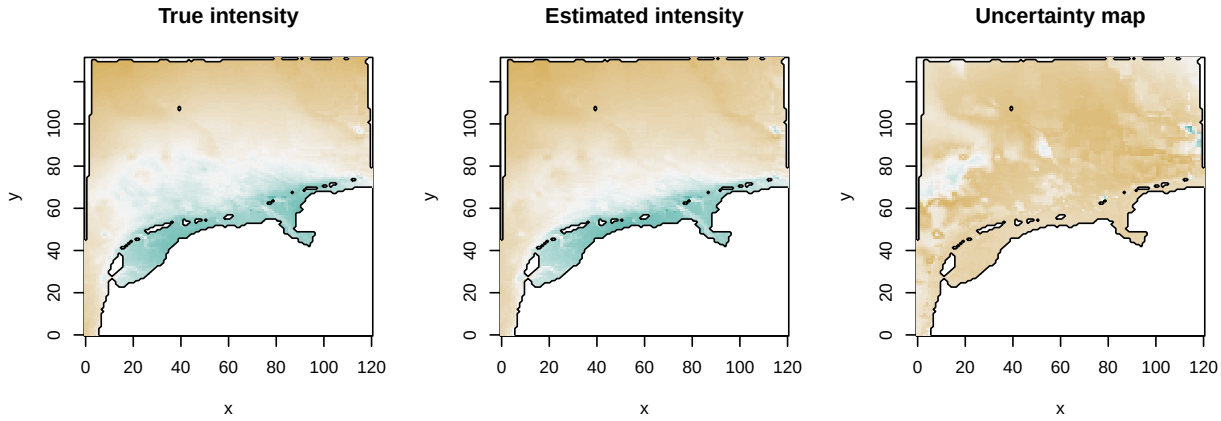
For this model, all the true habitat parameter values are retrieved correctly, however, the colony ranging parameters are a little harder to pin down.

4.3 Spatial predictions from the model

Although it would be theoretically possible to embed the spatial prediction functionality into the model fitting, the sheer size of the maps over which prediction will be required makes this option less ideal. Instead, we can extract a sensible sample from the posterior of the JAGS output by using `as.mcmc` from the library `coda`. This functionality is incorporated in the `jointHSF.predict` function, which requires the fitted model object (in this case `resultsIPP`), a matrix of prediction covariates (here called `Xpred`) and a template of the map (here `land`). The prediction covariates matrix needs to have columns for all the covariates of the model, preceded by a column of 1's, standing in for the value of the "covariate" to the model's intercept. The predictions for mean and standard deviation of the intensity are returned in matrix format, in the dimensions of the map template, so they can be plotted directly as images.

```
Xpred<-cbind(rep(1,length(c(u1))),c(u1),c(u2))
preds<-jointHSF.predict(resultsIPP, aLocs=c(1,2,3), cLocs=c(4,5,6),
                        Xpred=Xpred, Dist=Dist, mask=land, Colonies=Colonies)

par(mfrow=c(1,3))
image(x,y,lambda/land, col=pal(50),
      main="True intensity")
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
image(x,y,preds$muMap/land, col=pal(50),
      main="Estimated intensity")
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
image(x,y,preds$sdMap/(preds$muMap*land), col=pal(50),
      main="Uncertainty map")
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
```




```
par(mfrow=c(1,1))
```

5 More advanced joint inference for the distribution of species from pooled survey data

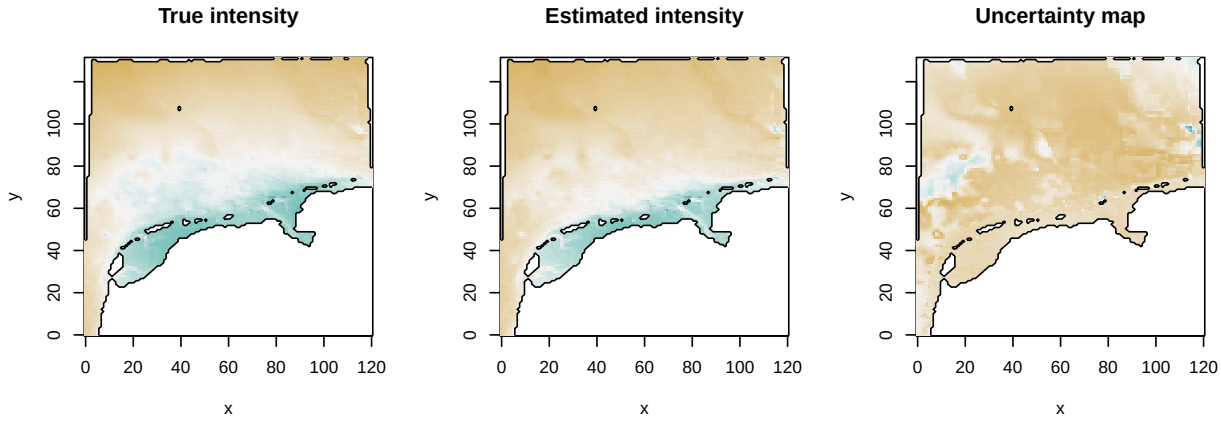
The `jointHSF` function has access to more advanced models intented to deal with overdispersion and explicitly spatial relationships.

5.1 Modelling overdispersion with a negative binomial model

The model with negative binomial likelihood can be called by altering the model name in the `jointHSF` function. The summaries from this model report on the degree of overdispersion in the data.

```
resultsNB<-jointHSF(count, X, dist, surv, ll, w, p0, N,
                    model="NEGBIN", ABS=c(1000,10000,5000))
sumsNB<-summary(resultsNB)
```

Parameter	True values	2.5% limit	Median	97.5% limit
a_1	-0.01	-0.0187	-0.0101	0.0015
a_2	0.08	-0.1158	0.0755	0.2267
c_0	-0.02	-0.0099	-0.0053	-9×10^{-4}
c_1	3×10^{-6}	5.5524×10^{-9}	5.07279×10^{-8}	9.99905×10^{-8}
c_2	0.025	0.0011085	0.0478849	0.0955328
q	1	1.0002	1.0101	1.0598



5.2 Model `spatial`

This model accounts for spatial autocorrelation in the response variable. This could be caused by missing, but important covariates, or by social aggregations between individuals of the species, assuming that these aggregations give rise to durable hotspots in distribution that are detectable in the data. In addition, the autocorrelation features allow the model to operate as a density smoother in the absence of any covariate information. However, these models are quite expensive computationally and they cannot be used for spatial extrapolation. So, they should really only be used for estimation of distributions within the survey areas. We will demonstrate their application within the region close to surveys 1 and 2.

```
# Extent of region for modelling
xms<-min(c(pts1[,1],pts2[,1]),pts3[,1])+10
xMs<-xms+40
yms<-min(c(pts1[,2],pts2[,2]),pts3[,2])+10
yMs<-yms+40

cs<-as.matrix(expand.grid(seq(xms,xMs,1),seq(yms,yMs,1)))
cs<-cs[land[round(cs)]==1,]
csL<-length(cs[,1])

# Data frame for spatial grid locations
data1<-data.frame(
  # x coordinates
  "x"=cs[,1],
  # y coordinates
  "y"=cs[,2],
  # response data
  "counts"=rep(0,csL),
  # Bathymetry values at survey locations
  "depth"=c(u1[round(cs)]),
```

```
  # Originating survey for datum
  "sv"=c(rep(4,csL))
)

# Data frame for survey locations
data2<-data.frame(
  # x coordinates
  "x"=c(pts1[,1],pts2[,1],pts3[,1]),
  # y coordinates
  "y"=c(pts1[,2],pts2[,2],pts3[,2]),
  # response data
  "counts"=c(cnts1,cnts2,cnts3),
  # Bathymetry values at survey locations
  "depth"=c(u1[round(pts1)],u1[round(pts2)], u1[round(pts3)]),
  # Originating survey for datum
  "sv"=c(rep(1,length(cnts1)),rep(2,length(cnts2)),rep(3,length(cnts3)))
)

dataS<-rbind(data1,data2) # Combined survey and unobserved locations
n<-length(dataS[,1]) # Sample size of data
disp<-as.matrix(dist(dataS[,1:2],diag=TRUE,upper=TRUE),byrow=TRUE)


cov<-0.4*exp(-0.01*disp^2)
mu<-rep(0,n)
ers<-mvrnorm(1,mu,cov)
ersS<-(ers-min(ers))/(max(ers)-min(ers))

image(x,y,lambda/land, col=pal(10))
contour(x,y,land, levels=c(0.5),add=TRUE,drawlabels=FALSE)
col<-brewer.pal(9, name="Reds")[1+round(8*ersS)] #Colour palette for survey 1
symbols(dataS$x,dataS$y, circles=rep(0.001,length(dataS$x)), inches=max(ersS)/100,add=TR
polygon(c(xms,xms,xMs,xMs,xms), c(yms,yMs,yMs,yms,yms))
```

```
#dataS<-rbind(data1,data2)
dataS<-data2[seq(1,length(data2[,1]),5),]
n<-length(dataS[,1]) # Sample size of data
disp<-as.matrix(dist(dataS[,1:2],diag=TRUE,upper=TRUE),byrow=TRUE)

# Coercion of a covariate data-frame into a matrix
X<-as.matrix(dataS[,4]) # Note use of one-less covariate
# Survey tags for each point in the data
surv<-dataS$sv
```

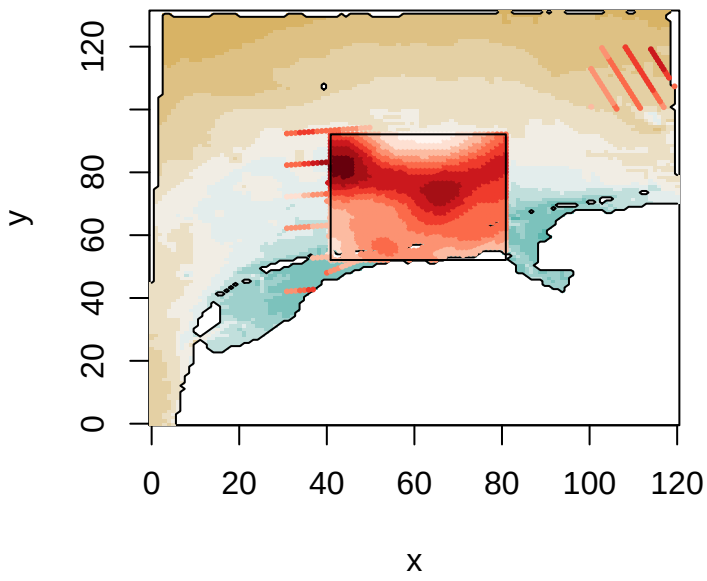


Figure 10: Autocorrelated species density along transects and at points of a regular grid within a subset of space

```
# Response data
count<-dataS$counts
# Colony sizes
N<-Colonies[,3]
# Distance data
dist<-matrix(0,n,J)
for(j in 1:J) dist[,j]<-dis[cbind(rep(j,n),round(dataS$x),round(dataS$y))]


# Augmenting survey characteristic vectors with values for
# "4th survey type" - corresponding to unobserved locations
lls<-c(ll,1)
ws<-c(w,1)
p0s<-c(p0,0) # note Zero probability of detection

resultsSPA<-jointHSF(count, X, dist, surv, lls, ws, p0s, N,
                     model="SPATIAL", disp=disp, ABS=c(1000,10000,5000))
resultsIPPS<-jointHSF(count, X, dist, surv, lls, ws, p0s, N,
                      model="IPP", ABS=c(1000,10000,5000))
# # Extraction of summaries for posteriors
sumsSPA<-summary(resultsSPA)
sumsIPPS<-summary(resultsIPPS)
```

Parameter	True values	2.5% limit	Median NS	97.5% limit	2.5% limit	Median S	97.5% limit
a_1	-0.01	-0.0128	0.0029	0.0187	-0.0104	0.006	0.0233
c_0	-0.02	-0.0135	-0.0045	0	-0.0283	-0.01	10^{-4}
c_1	0	5.3×10^{-9}	5.11×10^{-8}	10^{-7}	10^{-10}	4.84×10^{-8}	9.44×10^{-8}
c_2	0.025	2×10^{-4}	0.0499	0.0953	8×10^{-4}	0.0497	0.0958

6 Exploring performance of survey methods for different seabird species

6.1 Covariates and colonies for Scottish seabird populations

The `JointSurvey` package contains an example data set with layers for the east coast of Scotland. These include real colony locations and sizes for four seabird species (Gannet, Greater Black-Backed Gull, Guillemot & Kittiwake), explanatory variables such as bathymetry and sediment, and masks for the boundaries of predefined survey regions. The envornmental

and survey layers are shown in Fig.11. The sediment map comprises 6 classes, that follow a gradient of coarseness, from mud to boulders.

```
par(mfrow=c(1,3))

# DEPTH ====================================
image(rowN,colN, depth, zlim=c(0,100), col=pal(10), main="Bathymetry")
contour(rowN,colN, dmask, add=TRUE, levels=c(0,1))
points(realCols[,3], realCols[,4], pch=16, col="Brown", cex=2)

# SEDIMENT =================================
image(rowN,colN, sed, zlim=c(0,5), col=pal(10), main="Sediment classes")
contour(rowN,colN, dmask, add=TRUE, levels=c(0,1))
points(realCols[,3], realCols[,4], pch=16, col="Brown", cex=2)

# WINDFARMS ================================
wfTot<-matrix(0,length(rowN),length(colN))
for(i in 1:9)
{
  wfTot<-wfTot+wf[i,,]
}
image(rowN,colN,wfTot, main="Windfarm survey areas")
contour(rowN,colN,dmask, add=TRUE, levels=c(0,1))
points(realCols[,3], realCols[,4], pch=16, col="Brown", cex=2)
```

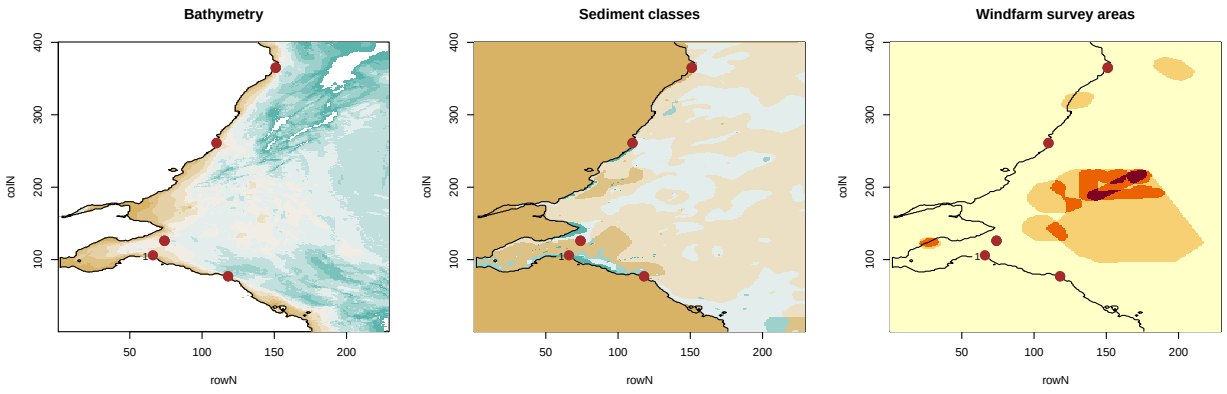


Figure 11: Three spatial layers used for the four exemplar species.

```
par(mfrow=c(1,1))
```

6.2 Simulating bird distributions for each species

The parameters determining the response of the four species to the underlying covariates are chosen arbitrarily. The plots in Fig. 12 are therefore fictitious, but constructed so that they give a different distribution that is characteristic of each species. The realised distribution of the species, the right-hand column, is derived from an IPP.

6.3 Specifying realistic survey areas and designs

The outlines shown in Fig. 13 are characteristic survey boundaries derived from real surveys in the North Sea. These are often typically separated by spatial distances but also time (for surveys that appear to overlap spatially).

6.4 Performance and spatial predictions of model by species

The comparison between true and estimated distributions of simulated seabirds is shown in 14. The tables that follow show the ability of the models to retrieve the true underlying parameters for the four exemplar species.

True and estimated parameter values for Gannet

Parameter	True values	2.5% limit	Median	97.5% limit
a_1	0	-0.01	0	0
a_{21}	3	0.66	1.23	1.84
a_{22}	-0.7	-0.41	-0.31	-0.16
c_0	-0.01	-0.01	-0.01	-0.01
c_1	0	0	0	0
c_2	0	0.01	0.05	0.1

True and estimated parameter values for GBB gull

Parameter	True values	2.5% limit	Median	97.5% limit
a_1	-0.01	-0.12	-0.05	0.04
a_{21}	4	-1.88	7.09	16.92
a_{22}	-0.4	-2.55	-0.94	0.56
c_0	-0.01	-0.02	-0.01	0
c_1	0	0	0	0
c_2	0	0	0.05	0.1

True and estimated parameter values for Guillemot

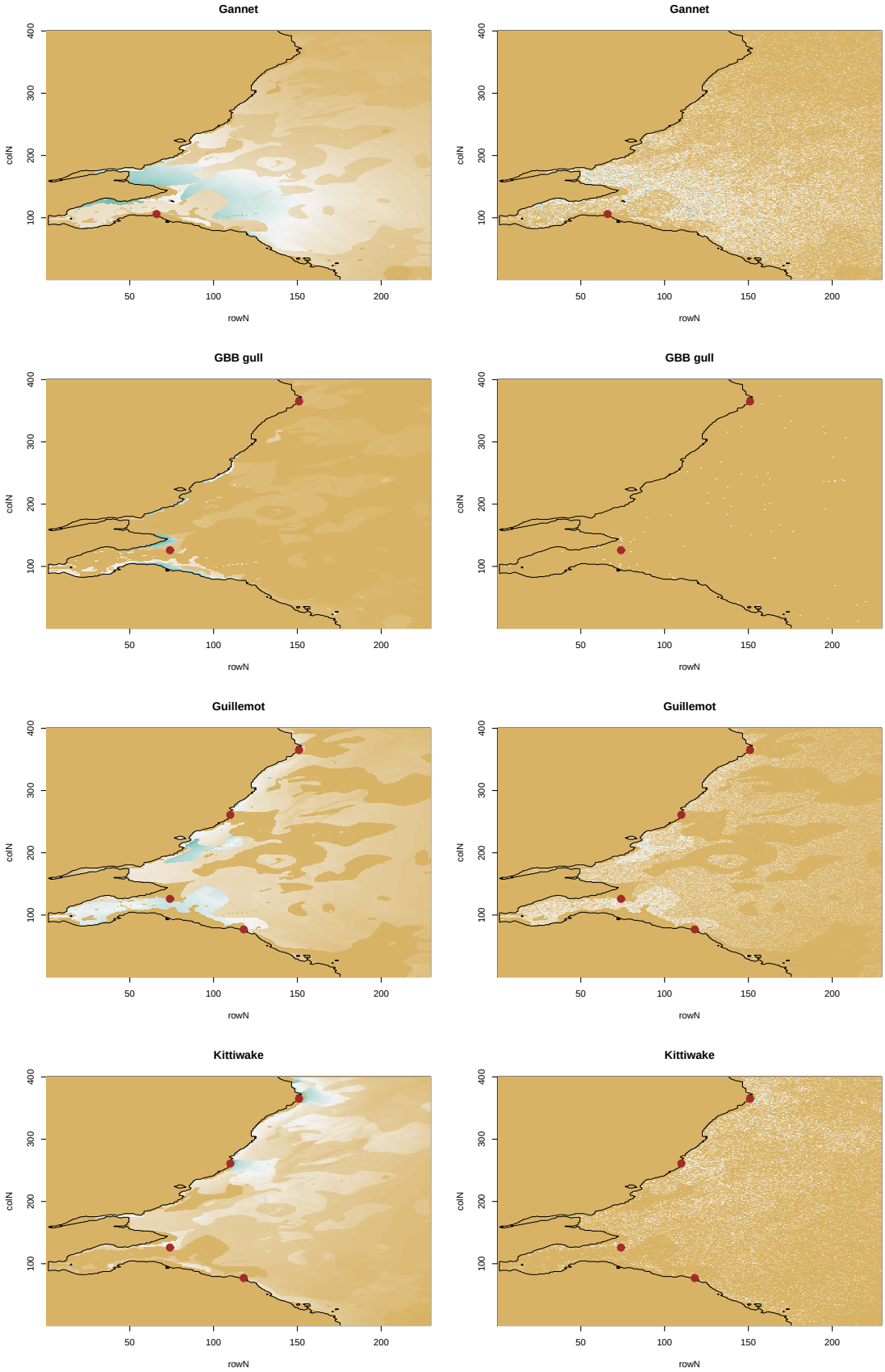


Figure 12: Hypothetical expectations (left hand side) and realisations (right hand side) of the distributions for the four exemplar species.

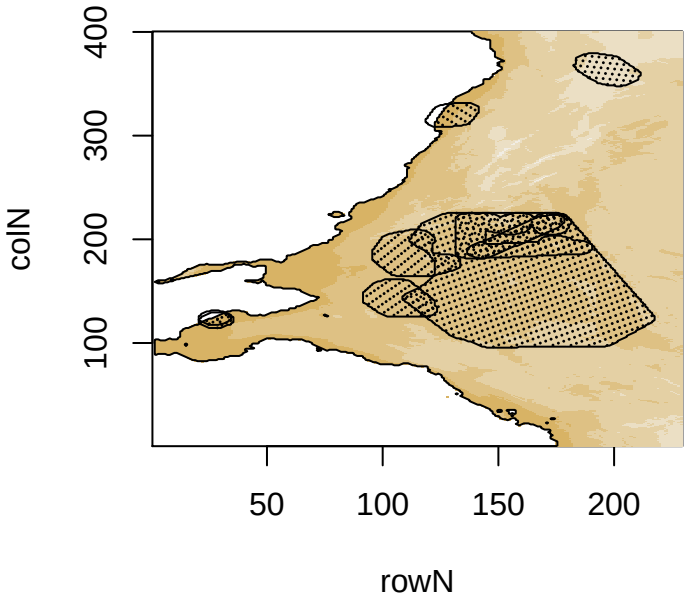


Figure 13: Realisation of a single survey

Parameter	True values	2.5% limit	Median	97.5% limit
a_1	-0.01	-0.01	-0.01	-0.01
a_{21}	2	0.67	0.93	1.34
a_{22}	-0.9	-0.61	-0.52	-0.45
c_0	-0.01	-0.01	-0.01	-0.01
c_1	0	0	0	0
c_2	0	0.01	0.06	0.1

True and estimated parameter values for Kittiwake

Parameter	True values	2.5% limit	Median	97.5% limit
a_1	-0.01	-0.01	0	0
a_{21}	4	0.49	1.33	2.24
a_{22}	-0.7	-0.42	-0.23	-0.05
c_0	-0.01	-0.02	-0.01	-0.01
c_1	0	0	0	0
c_2	0	0	0.05	0.1

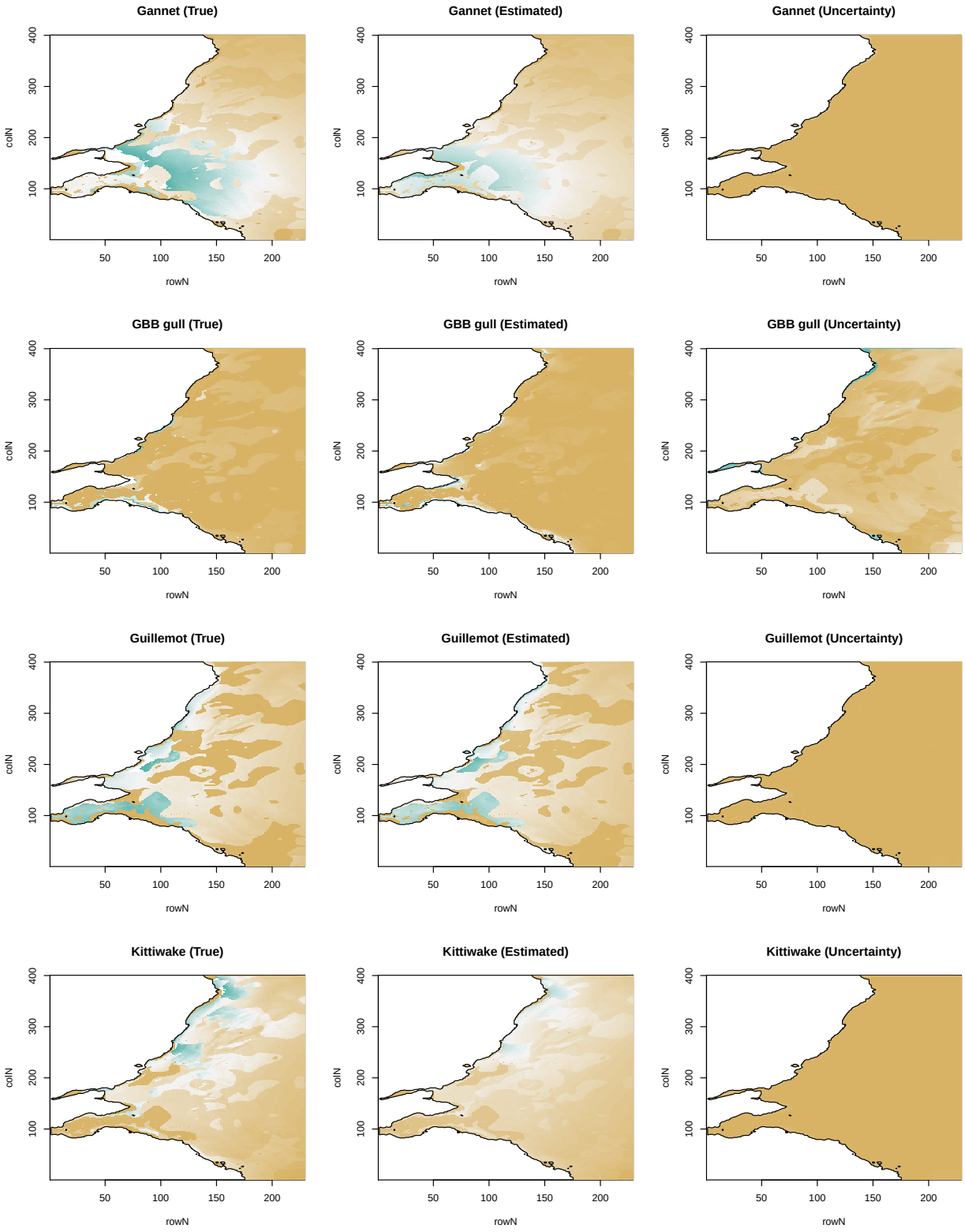


Figure 14: Model reconstructions of the true maps (1st column), represented by mean predictions (2nd column) and the coefficient of variation in predictions (3rd column), for the four exemplar seabird species (rows).

References

Aarts, Geert, John Fieberg, and Jason Matthiopoulos. 2011. "Comparative interpretation of count, presence-absence and point methods for species distribution models." *Methods in Ecology and Evolution*, no–. https://doi.org/10.1111/j.2041-210X.2011.00141.x.

Buckland, S. T., D. R. Anderson, K. P Burnham, J. L. Laake, D. Borchers, and L. Thomas. 2001. *Introduction to Distance Sampling: Estimating Abundance of Biological Populations*. Oxford University Press.

Distler, Trisha, Justin G. Schuetz, Jorge Velásquez-Tibatá, and Gary M. Langham. 2015. "Stacked species distribution models and macroecological models provide congruent projections of avian species richness under climate change." *Journal of Biogeography* 42 (5): 976–88. https://doi.org/10.1111/jbi.12479.

Lewis, S, T N Sherratt, K C Hamer, and S Wanless. 2001. "Evidence of intra-specific ccompetition for food in a pelagic seabird." *Nature* 412 (August): 816–19.

Matthiopoulos, Jason. 2003. "The use of space by animals as a function of accessibility and preference." *Ecological Modelling* 159 (2-3): 239–68. https://doi.org/10.1016/S0304-3800(02)00293-4.

Matthiopoulos, Jason, Bernie Mcconnell, Callan Duck, and Mike Fedak. 2004. "Using satellite telemetry and aerial counts to estimate space use by grey seals around the British Isles." *Journal of Applied Ecology* 41 (3): 476–91. https://doi.org/10.1111/j.0021-8901.2004.00911.x.

Plummer, Martyn, and Others. 2003. "{JAGS}: A program for analysis of Bayesian graphical models using {Gibbs} sampling." In *Proceedings of the 3rd International Workshop on Distributed Statistical Computing*, 124:125. Technische Universit at Wien Wien, Austria.

Renner, Ian W, Jane Elith, Adrian Baddeley, William Fithian, Trevor Hastie, Steven J Phillips, Gordana Popovic, and David I Warton. 2015. "Point process models for presence-only analysis." *Methods in Ecology and Evolution* 6 (4): 366–79.

Renner, Ian W, and David I Warton. 2013. "Equivalence of {MAXENT} and {Poisson} point process models for species distribution modeling in ecology." *Biometrics* 69 (1): 274–81.

Thaxter, Chris B., Ben Lascelles, Kate Sugar, Aonghais S. C. P. Cook, Staffan Roos, Mark Bolton, Rowena H. W. Langston, and Niall H. K. Burton. 2012. "Seabird foraging ranges as a preliminary tool for identifying candidate Marine Protected Areas." *Biological Conservation* 156: 53–61. https://doi.org/10.1016/j.biocon.2011.12.009.

Wakefield, Ewan D., Thomas W. Bodey, Stuart Bearhop, Jez Blackburn, Kendrew Colhoun, Rachel Davies, Ross G. Dwyer, et al. 2013. "Space partitioning without territoriality in gannets." *Science* 341 (6141): 68–70. https://doi.org/10.1126/science.1236077.

Warton, David I, Leah C Shepherd, and Others. 2010. "Poisson point process models solve the {'pseudo-absence problem'} for presence-only data in ecology." *The Annals of Applied Statistics* 4 (3): 1383–1402.