11

MODELING SPATIAL VARIATION IN DENSITY

Underlying every SCR model is a spatial point process that describes the number and distribution of animal activity centers. Spatial point processes are characterized by two key elements: a spatial domain, or state-space $\mathcal{S}$, and an intensity function which returns the expected density of points at any location in $\mathcal{S}$. If the intensity is constant throughout $\mathcal{S}$, the point process is said to be homogeneous. Thus far we have focused our attention on homogeneous point processes whose realized values are the locations of the N activity centers. When a Poisson prior is placed on N, the model is known as a homogeneous Poisson point process, which is the classic model of "complete spatial randomness." A similar model, that we often use in conjunction with data augmentation and MCMC, places a binomial prior on N. This is also a model of spatial randomness, and in this chapter we will compare and contrast the two.

The spatial randomness assumption is often viewed as restrictive because ecological processes such as habitat selection can result in non-uniform distributions of organisms. We have argued, however, that this assumption is less restrictive than may be recognized because a homogeneous point process actually allows for infinite possible "point patterns", or realized configurations of activity centers. Furthermore, given enough data, the uniform prior will have very little influence on the estimated locations of activity centers. Nonetheless, a homogeneous point process does not allow one to model population density using covariates, which is an important objective in much ecological research. For example, even when assuming a homogeneous point process for the activity centers, an estimated density surface may strongly suggest that individuals are more abundant in one habitat than another; however, such results do not provide the basis for formally testing hypotheses about spatial variation in density, and they could not be used to make predictions about habitat-specific abundance in other regions. A more direct approach is to replace the homogeneous model with an inhomogeneous model in which the point process intensity is allowed to vary spatially.

In this chapter, we cover methods for fitting inhomogeneous Poisson and binomial point process models so that density can be modeled as a function of covariates in much the same way as is done in generalized linear models. The covariates we consider differ from those covered in previous chapters, which were typically attributes of the animal (e.g. sex or age) or the trap (e.g. baited or not) and were used to model movement or encounter rate. In contrast, here we wish to model covariates that are defined at all points in $\mathcal{S}$, and so we will refer to them as state-space covariates or density covariates. These may include continuous covariates such as elevation, or categorical covariates such as habitat type. Typically, these state-space covariates are formatted as raster images with a prescribed resolution and extent.

One thing to keep in mind when modeling density is that the SCR definition of density is different than what is perhaps a more common definition of density in ecology. In SCR models, density is defined as the number (or expected number) of *activity centers* in some region, whereas in other ecological studies, density is often defined as the number of *individuals* in some region at some instant in time. The latter definition is closer to the quantity being estimated in distance sampling studies. So which definition is better? Does it make more sense to contemplate activity centers or individuals at an instant in time? From our perspective, either definition may suffice for a given objective, but we note that there exists a formal relationship between the two since an activity center is the *average* of an individual's locations during some time period. As such, an activity center may be a better descriptor of an individual's preferences than is a location during a single instant in time. Moreover, with SCR models we can model both the distribution of activity centers (as we will do in this chapter) as well as the distribution of individuals during specific instances in time, as is demonstrated in Chapt. 15.

Inhomogeneous Poisson point process models were discussed in the original formulation of SCR models by Efford (2004) and were described in more detail by Borchers and Efford (2008). We will show that an inhomogeneous point process with a binomial prior on N is quite similar to the Poisson model, but is more easily implemented in MCMC algorithms. To do so, we will define the data augmentation parameter ψ in terms of the point process intensity function, and we will replace the uniform prior on the activity centers with a prior that is also derived from the intensity function. Development of this prior, which does not have a standard form, is a central component of this chapter. First we begin with a review of homogeneous point process models.

11.1 HOMOGENEOUS POINT PROCESS REVISITED

The homogeneous Poisson point process is *the* model of complete spatial randomness and is often used in ecology as a null model to test for departures from randomness (Cressie, 1991; Diggle, 2003; Illian et al., 2008). The Poisson model asserts that the number of points in $\mathcal{S}$ is Poisson distributed: $N \sim \text{Poisson}(\mu|\mathcal{S}|)$ where $\mu > 0$ is the intensity parameter and $|\mathcal{S}|$ is the area of the state-space. The intensity parameter μ is the density of points, and thus multiplying the intensity by the area of some region yields the expected number of points in that region. As with all homogeneous point process models, the N points are distributed uniformly, which implies that they do not interact with each other in any way – for example, they neither attract nor repel one another.

Unlike the Poisson point process, the binomial point process assumes that N is fixed, not random. The distinction is illustrated by this simple **R** code that generates realizations from Poisson and binomial point processes in the unit square ($\mathcal{S} = [0,1] \times [0,1]$):

```
> Area <- 1                          # Area of unit square
> muP <- 4                           # intensity
> nP <- rpois(1, muP*Area)           # number of points: random
> PPP <- cbind(runif(nP), runif(nP)) # Poisson point pattern
> nB <- 4                            # number of points: fixed
> muB <- nB/Area                     # intensity
> BPP <- cbind(runif(nB), runif(nB)) # binomial point pattern
```

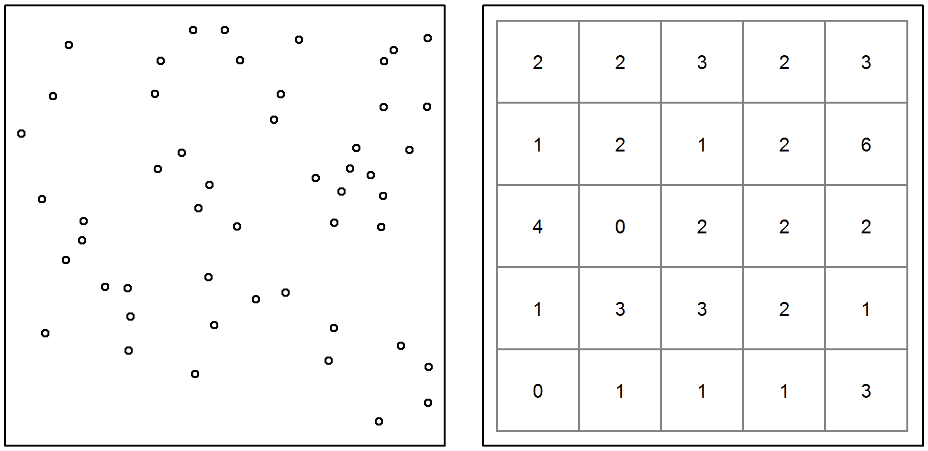
Both of these models are homogeneous because the intensity parameter is constant ($\mu = 4$ in both cases) and the locations of N the points are mutually independent and uniformly distributed~~.~~ with each other. The key distinction is that N is random in the former and fixed in the latter.

Another difference between the Poisson and binomial models is that if the state-space is divided into K disjunct regions, the number of points in each region $n(B_k) : k = 1, \dots, K$; are independent and identically distributed (i.i.d.) under the Poisson model but not under the binomial model. In the Poisson case, the counts are $n(B_k) \sim \text{Poisson}(\mu|B_k|)$, where $|B_k|$ is the area of the region B_k. For the binomial model, $n(B_k) \sim \text{Binomial}(N, \pi(B_k))$ where $\pi(B_k)$ is the proportion of the state-space in B_k; however, these counts are not i.i.d. because the number of points in one region is informative about the number of points in another region. For example, if $N = 10$ and if there are 7 points outside the region B_1, then we can say with certainty that $B_1 = 10 - 7 = 3$.

Fig. 11.1 is meant to further illustrate the characteristics of the binomial model. The left panel shows a point pattern realized from a homogeneous binomial point process with $N = 50$. The right panel shows the same realization, except that the state-space has been discretized into 25 equally-sized disjunct regions, or pixels, and the counts in each pixel are shown. Since the pixels are the same size, we have that $\pi(B_k) = 1/25$, and the expected number of point in each pixel is $\mathbb{E}(n(B_k)) = N\pi(B_k) = 50/25 = 2$, which happens to be the empirical mean in this instance. However, as previously stated, these counts are not independent realizations from a binomial distribution since $\sum_k n(B_k) = N$. Rather, the model for the entire vector is multinomial: $\{n(B_1), n(B_2), \dots, n(B_k)\} \sim \text{Multinomial}(N, \{p(B_1), p(B_2), \dots, p(B_K)\})$ (Illian et al., 2008). If you need a refresher on the multinomial distribution, refer to Sec. 2.2.3, and consider the following **R** code, which generates counts similar to those seen in Fig. 11.1:

```
> n.Bk <- rmultinom(1, size=50, prob=rep(1/25, 25))
> matrix(n.Bk, 5, 5)
     [,1] [,2] [,3] [,4] [,5]
[1,]    2    2    2    2    1
[2,]    2    4    0    5    0
[3,]    0    3    2    4    1
[4,]    1    2    1    4    1
[5,]    4    2    4    1    0
```

The dependence among counts has virtually no practical consequence when the number of pixels is large. For example, if there are 100 pixels, the number of points in one pixels carries very little information about the expected number of points in another pixel. However, if there are only 2 pixels, then clearly the number of points in one pixel allows one to determine how many points will occur in the remaining pixel.



2	2	3	2	3
1	2	1	2	6
4	0	2	2	2
1	3	3	2	1
0	1	1	1	3

Figure 11.1. Homogeneous binomial point process with N=50 points represented in continuous and discrete space.

The discrete representation of space shown in Fig. 11.1 is not only helpful for understanding the properties of a point process, it is also of practical importance when fitting SCR models because spatial covariates are almost always represented as rasters, i.e. grids with predetermined extent and resolution. In such cases, the definition of the prior for the point locations can be changed from the probability that a point occurs at some location in space to the probability that it occurs in some pixel of the raster. As we will explain in Sec. 11.4.2, this typically involves changing the prior from a uniform distribution to a multinomial or categorical distribution.

Having sketched out the basic characteristics of homogeneous Poisson and binomial point process models, we will now review their relevance to SCR models before moving on to the inhomogeneous models. In a SCR model with a homogeneous point process, the intensity parameter μ is interpreted as population density, and N is interpreted as population size (i.e. the number of activity centers in $\mathcal{S}$). These interpretations are true regardless of whether we consider the Poisson model or the binomial model, but since N is always unknown, one might wonder why we are discussing the binomial model at all.

In our work, we typically adopt the binomial model simply because it is easy to implement using MCMC and data augmentation. And while N is truly unknown, we use an upper bound, M, which is fixed. Thus, the standard point process we use in Bayesian analyses can be regarded in two ways. First, it is a binomial point process with M points. Second, in terms of N, it is a thinned binomial point process, where ψ is the thinning parameter. With this in mind, the only real difference between the Poisson and binomial models, as implemented in SCR contexts, is that in the former, we have $N \sim \text{Poisson}(\mu|\mathcal{S}|)$, and in the latter we have $N \sim \text{Binomial}(M, \psi)$. In other words, we just have a different prior on N, and when using MCMC, the binomial prior is much more convenient because it fixes the size of the parameter space and makes it easy to extend the model in each of the ways discussed in this book. It is also worth remembering that the Poisson distribution is the limit of the binomial distribution when M is very large and ψ is very small (Chapt. 2), and thus the two models are much more similar than may appear.

You might have noticed that the intensity parameter μ was not shown for the binomial prior $N \sim \text{Binomial}(M, \psi)$. Instead, we see the data augmentation parameter ψ, which has been used throughout this book, but without much mention of the point process intensity. What then is the relationship between ψ and μ? As first discussed in Chapt.5, under data augmentation, the expected value of N is $\mathbb{E}[N] = M\psi$. But, from this chapter, we also know that the expected value of N can be written in terms of μ as $\mathbb{E}[N] = \mu|\mathcal{S}|$. Therefore, $\psi = \mu|\mathcal{S}|/M$ and hence we can directly estimate μ rather than ψ if we want, as will be demonstrated in the next section where the objective is to model μ as a function of spatially-referenced covariates. First, consider the following **R** code, which illustrates some the concepts we just covered:

```
> Area <- 1                    # Area of state-space
> M <- 100                     # Data augmentation size
> mu <- 10                     # Intensity (points per area)
> psi <- (mu*Area)/M           # Data augmentation parameter (thinning rate)
> N <- rbinom(1, M, psi)       # Realized value of N under binomial prior
> cbind(runif(N), runif(N))    # Point pattern from thinned binomial model
            [,1]       [,2]
 [1,] 0.52779588 0.84306878
 [2,] 0.11529168 0.80635046
 [3,] 0.06777632 0.66072116
 [4,] 0.18694649 0.56761245
 [5,] 0.30176929 0.03159091
 [6,] 0.84352724 0.89691452
 [7,] 0.52766808 0.08871199
 [8,] 0.73007529 0.63184825
 [9,] 0.01119023 0.69807029
```

11.2 INHOMOGENEOUS POINT PROCESSES

The principal difference between homogeneous and inhomogeneous point processes is that the intensity parameter μ is allowed to vary spatially in the inhomogeneous model. Thus, rather than μ being a fixed constant, it is now a function defined at each point $\mathbf{x} \in \mathcal{S}$. A vast number of options exist for modeling spatial variation in the intensity of a point process (Cox, 1955; Stoyan and Penttinen, 2000; Illian et al., 2008), but here we focus on modeling μ as a function of spatially-referenced covariates and a vector of regression coefficients $\boldsymbol{\beta}$; a function we will denote $\mu(\mathbf{x}, \boldsymbol{\beta})$. To be clear, $\mu(\mathbf{x}, \boldsymbol{\beta})$, is a function that returns the expected density of activity centers at location $\mathbf{x}$, given the covariate values at $\mathbf{x}$[1]. Since the intensity must be positive, and because the natural logarithm is the

[1] The use of $\mathbf{x}$ to denote any point in the state-space could cause confusion because we use $\mathbf{x}_j$ as the location of a trap, but it is standard notation, and the distinction should be evident by the context.

canonical link function of the Poisson generalized linear model (McCullagh and Nelder, 1989), it is natural to consider the following model:

$$\log(\mu(\mathbf{x},\boldsymbol{\beta})) = \beta_0 + \sum_{v=1}^{V} \beta_v C_v(\mathbf{x}) \tag{11.2.1}$$

which says that there are V covariates and β_v is the regression coefficient for covariate $C_v(\mathbf{x})$. This covariate, $C_v(\mathbf{x})$, could be any variable defined at all points in the state-space, such as habitat type or elevation. Eq. 11.2.1 should look familiar because it is the standard linear predictor used in Poisson regression. As with other GLMs, one could consider alternative link functions.

Recall from the previous section that for a homogeneous point process, the expected number of points in the state-space was simply the intensity parameter multiplied by area: $\mathbb{E}(N) = \mu|\mathcal{S}|$. But now that we are regarding the intensity as a function, rather than a scalar, this equation is not very useful. So what is $\mathbb{E}(N)$ for an inhomogeneous point process? Contemplating a discrete state-space is useful for figuring this out. Imagine that the state-space is represented as a raster with many tiny pixels. In this case, we will associate $\mathbf{x}$ with pixel ID, i.e. $\mathbf{x}$ just references some pixel with V covariates values associated with it. The expected number of individuals in this pixel, say $\mathbb{E}(n(\mathbf{x}))$, can intuitively be found by evaluating the intensity function (Eq. 11.2.1) and multiplying it by the area of the pixel. In other words, we compute the expected number of individuals in a pixel by multiplying the expected value of density for that pixel by the area of the pixel. If we do this for each pixel in the state-space, then summing up these values gives us what we are after, the expected value of N. Specifically, $\mathbb{E}(N) = \sum_{\mathbf{x}\in\mathcal{S}} \mathbb{E}(n(\mathbf{x}))$. As the area of the pixels approaches zero, such that we move from discrete space back to continuous space, the summation must be replaced with an integration of the form:

$$\mathbb{E}(N) = \int_{\mathcal{S}} \mu(\mathbf{x},\boldsymbol{\beta})\mathrm{d}\mathbf{x}. \tag{11.2.2}$$

Together, Eqs. 11.2.1 and 11.2.2 describe a model for spatial variation in density as well as population size. The key task in fitting such inhomogeneous point process models is to estimate the $\boldsymbol{\beta}$ parameters.

We have now described an approach for modeling the point process intensity, yet in order to define the likelihood or to develop an MCMC algorithm for the inhomogeneous model, we need to specify the prior distribution for the activity centers. Recall that under the homogeneous point process, the prior was $\mathbf{s}_i \sim \text{Uniform}(\mathcal{S})$, for $i = 1,\dots,N$, or equivalently:

$$[\mathbf{s}_i] = 1/|\mathcal{S}| \tag{11.2.3}$$

where, as before, $|\mathcal{S}|$ is the area of the state-space. This simply indicates that an activity center is just as likely to occur at any location as another. However, if animals exhibit habitat selection or simply occur in one region more often than another, it would be preferable to replace this prior with one describing the spatial variation in density. Clearly this prior should be determined in some way by the spatially-varying intensity function $\mu(\mathbf{x},\boldsymbol{\beta})$. Since the integral of a probability density function (pdf) must be unity, we can convert $\mu(\mathbf{x},\boldsymbol{\beta})$ into a pdf by dividing it by a normalizing constant. In this case, the normalizing constant is found by integrating $\mu(\mathbf{x}, \boldsymbol{\beta})$ over the entire state-space. The probability density function of the new prior is therefore:

$$[\mathbf{s}_i|\boldsymbol{\beta}] = \frac{\mu(\mathbf{s}_i, \boldsymbol{\beta})}{\int_{\mathcal{S}} \mu(\mathbf{x}, \boldsymbol{\beta})\,\mathrm{d}\mathbf{x}} \tag{11.2.4}$$

Substituting the uniform prior with this new distribution allows us to fit inhomogeneous binomial point process models to spatial capture-recapture data.

As a practical matter, note that the integral in the denominator of Eq. 11.2.4 is evaluated over space, and since we always regard space as two-dimensional (the state-space is planar), this is a two-dimensional integral that can be approximated using the methods discussed in Chapt. 9, which include Monte Carlo integration and Gaussian quadrature. Alternatively, if our state-space covariates are in raster format, i.e. they are in discrete space, the integral can be replaced with a summation over all the pixels in the raster,

$$[\mathbf{s}_i|\boldsymbol{\beta}] = \frac{\mu(\mathbf{s}_i, \boldsymbol{\beta})}{\sum_{\mathbf{x}\in\mathcal{S}} \mu(\mathbf{x}, \boldsymbol{\beta})} \tag{11.2.5}$$

where $\mathbf{s}$ is now defined as "pixel ID" rather than a point in space.

Although the discrete space approach is standard practice, it is technically unjustified because covariate values must be known for all points in space, and a raster is simply a set of spatially-referenced covariate values at an evenly-spaced subset of points (the pixel centers). This same problem is present anytime that we have a sample of the spatial covariates, rather than a function defining their value for all points in space. In such cases, it may be necessary to interpolate the values of the covariates for points in space where they were not measured. One option would be to use a Kriging interpolator, as demonstrated by Rathbun (1996). Another option is to sample the spatial covariates using probabilistic sampling methods, which allow for design-based estimators of their values for the entire study area (Rathbun et al., 2007). Either option could be implemented within maximum likelihood or MCMC estimation methods; however, we do not demonstrate them here because it seems likely that they will be inconsequential in most cases where the raster data are of high resolution, such that the loss of information is negligible when going from continuous space to discrete space. Furthermore, the validity of this assertion, and the level of resolution required to adequately approximate continuous space can often be assessed by checking the consistency of the parameter estimates among varying levels of resolution, as was demonstrated in Chapt. 5.

We now have all the tools needed to fit inhomogeneous point process models. Likelihood-based inference for inhomogeneous Poisson point process models was described by Borchers and Efford (2008) and reviewed in Chapt. 6. Another example is demonstrated in the next section, but first we focus on the binomial model that we favor when conducting Bayesian inference. In the previous section we noted that the data augmentation parameter ψ can be expressed in terms of the intensity parameter μ. The same is true for inhomogeneous models. Specifically, rather than $\mathbb{E}(N) = \psi M$ as before, we use the expected value of N shown in Eq. 11.2.2 which results in

$$\psi = \frac{\int_{\mathcal{S}} \mu(\mathbf{x}, \boldsymbol{\beta})\,\mathrm{d}\mathbf{x}}{M} \tag{11.2.6}$$

Note that the data augmentation limit M must be high enough so that it is greater than the numerator – i.e., the expected value of N must be less than M.

In the next sections we walk through a few examples, building up from the simplest case where we actually observe the activity centers as though they were data. In the second example, we fit the inhomogeneous model to simulated data in which density is a function of a single continuous covariate. The next example shows an analysis in discrete space using both `secr` (Efford, 2011a) and **JAGS** (Plummer, 2003), and in the final example, we model the intensity of activity centers for a real dataset collected on jaguars (*Panthera onca*) in Argentina.

11.3 OBSERVED POINT PROCESSES

In SCR models, the points (activity centers) are not directly observed, but in other contexts they are. Examples include the locations of disease outbreaks, the locations of trees in a forest, or the locations of radio-tracked animals. In such cases, it is straightforward to fit inhomogeneous point process models and estimate the the parameters $\boldsymbol{\beta}$ from Eq. 11.2.1, as we will do in the following example.

Suppose we knew the locations of N animal activity centers, perhaps as the result of an extensive telemetry study. If we assume N is Poisson distributed and the points are mutually independent of one another, we can fit the inhomogeneous Poisson point process model. The likelihood of this model has two components: $[\{\mathbf{s}_1, \dots, \mathbf{s}_N\}|N]$ and $[N]$. The pdf of the first part is given by Eq. 11.2.4, and with the Poisson assumption we have:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\beta}|\{\mathbf{s}_1, \dots, \mathbf{s}_N\}) &= [\{s_1, \dots, \mathbf{s}_N\}|N][N] \\ &= \left\{\prod_{i=1}^{N} \frac{\mu(\mathbf{s}_i, \boldsymbol{\beta})}{\int_{\mathcal{S}} \mu(\mathbf{x}, \boldsymbol{\beta})\mathrm{d}\mathbf{x}}\right\} \frac{e^{-\int_{\mathcal{S}} \mu(\mathbf{x},\boldsymbol{\beta})\mathrm{d}\mathbf{x}} \int_{\mathcal{S}} \mu(\mathbf{x}, \boldsymbol{\beta})\mathrm{d}\mathbf{x}^N}{N!}.\end{aligned}$$

This can be simplified by noting that the denominator in the first component of the model cancels with the corresponding piece in the numerator of the second component. And, since N is observed and thus does not depend on the parameters, $N!$ can be omitted as well. After log-transforming the remaining pieces, we have the log-likelihood often seen in textbooks, such as Diggle (2003, pg. 104):

$$\ell(\boldsymbol{\beta}|\{\mathbf{s}_i\}) = \sum_{i=1}^{N} \log(\mu(\mathbf{s}_i, \boldsymbol{\beta})) - \int_{\mathcal{S}} \mu(\mathbf{x}, \boldsymbol{\beta})\,\mathrm{d}\mathbf{x}.$$

Having arrived at the likelihood we could choose a prior distribution for $\boldsymbol{\beta}$ and obtain the posterior distribution using Bayesian methods, or we can find the maximum likelihood estimates (MLEs) using standard numerical methods as is demonstrated below.

First, we simulate some data under the model $\mu(\mathbf{x}, \boldsymbol{\beta}) = \beta_0 + \beta_1\mathrm{ELEV}(\mathbf{x})$, where $\mathrm{ELEV}(\mathbf{x})$ is a spatial covariate, say elevation, and $\beta_0 = -6$ and $\beta_1 = 1$. It is worth emphasizing that a spatial covariate must be defined at any location in the state-space, as is true of the following covariate `elev.fn`:

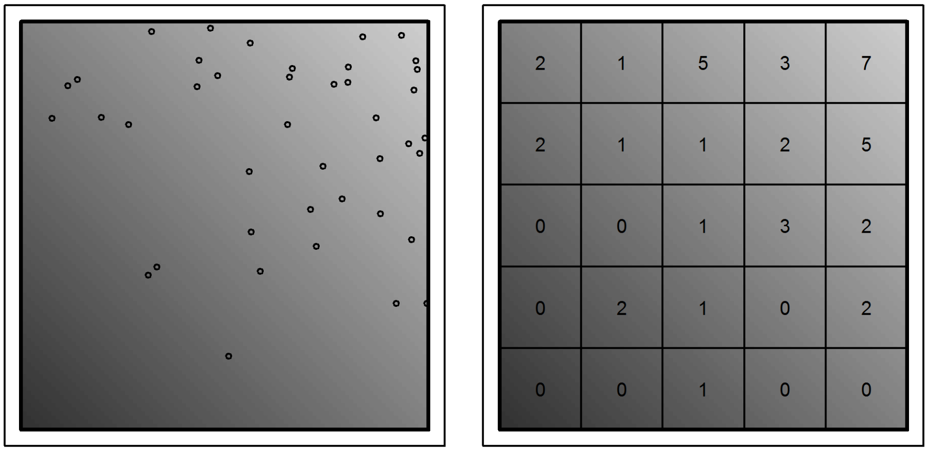
```
> elev.fn <- function(x) {          # spatial covriate
+     x <- matrix(x, ncol=2)        # Force x to be a matrix
+     (x[,1] + x[,2] - 100) / 40.8  # Returns (standardized) "elevation"
+ }
```

```
> # intensity function
> mu <- function(x, beta0, beta1) exp(beta0 + beta1*elev.fn(x=x))
> beta0 <- -6 # intercept of intensity function
> beta1 <- 1  # effect of elevation on intensity
> # Next line computes integral
> EN <- cuhre(2, 1, mu, beta0=beta0, beta1=beta1,
+             lower=c(0,0), upper=c(100,100))$value
```

The function `elev.fn` returns the value of elevation at any location `x`. The standardization bit is not necessary, but helps with the model fitting below. The next lines of the code define the intensity function $\mu(\mathbf{x}, \boldsymbol{\beta})$ in terms of elevation and the regression coefficients. The last line uses the `cuhre` function in the `R2Cuba` package (Hahn et al., 2010) to compute the expected value of N in a $[0, 100] \times [0, 100]$ square state-space, which is the two-dimensional integral of Eq. 11.2.4. This integral could also be computed using a fine grid of points as we have done in previous chapters, but it is useful to gain familiarity with more efficient integration functions in **R**.

The **R** code above demonstrates how to obtain the expected value of N given a spatial covariate and the coefficients defining the intensity function. Now we need to generate a realized value of N and distribute the N points in proportion to the intensity function. This is not as simple as it was to simulate data from a homogeneous point process because the points are no longer uniformly distributed within the state-space. Instead one must resort to methods such as rejection sampling, which involves simulating data from a standard distribution and then accepting or rejecting each point using probabilities defined by the distribution of interest. For more information, readers should consult an accessible text such as Robert and Casella (2010). In our example, we simulate from a uniform distribution and then accept or reject using the (scaled) probability density function $[\mathbf{s}_i|\boldsymbol{\beta}]$ (Eq. 11.2.4). The following **R** commands demonstrate the use of rejection sampling to simulate an inhomogeneous point process for the elevation covariate depicted in Fig. 11.2.

```
> set.seed(31025)
> beta0 <- -6 # intercept of intensity function
> beta1 <- 1  # effect of elevation on intensity
> # Next line computes integral, which is expected value of N
> EN <- cuhre(2, 1, mu, beta0=beta0, beta1=beta1,
+             lower=c(0,0), upper=c(100,100))$value
> EN
[1] 39.96634
> N <- rpois(1, EN) # Realized N
> s <- matrix(NA, N, 2) # This matrix will hold the coordinates
> elev.min <- elev.fn(c(0,0))
> elev.max <- elev.fn(c(100, 100))
> Q <- max(c(exp(beta0 + beta1*elev.min),
+            exp(beta0 + beta1*elev.max)))
> counter <- 1
> while(counter <= N) {
+   x.c <- runif(1, 0, 100); y.c <- runif(1, 0, 100)
+   s.cand <- c(x.c,y.c)
```



2	1	5	3	7
2	1	1	2	5
0	0	1	3	2
0	2	1	0	2
0	0	1	0	0

Figure 11.2. An example of a spatial covariate, say elevation, and a realization from an inhomogeneous Poisson point process with $\mu(\mathbf{x}, \boldsymbol{\beta}) = \exp(\beta_0 + \beta_1 \mathsf{ELEV}(\mathbf{x}))$ where $\beta_0 = -6$ and $\beta_1 = 1$.

```
+   pr <- mu(s.cand, beta0, beta1) #/ EN
+   if(runif(1) < pr/Q) {
+     s[counter,] <- s.cand
+     counter <- counter+1
+     }
+   }
```

Similar methods are also implemented in the **R** package `spatstat` (Baddeley and Turner, 2005).

The 41 simulated points are shown in Fig 11.2. High elevations are represented by light gray and low elevations are darker. The density of points in apparently higher in lighter regions suggesting that these simulated animals prefer high elevations. Given these points, we will now estimate β_0 and β_1 by minimizing the negative-log-likelihood using **R**'s `optim` function.

```
> nll <- function(beta) {
+     beta0 <- beta[1]
+     beta1 <- beta[2]
+     EN <- cuhre(2, 1, mu, beta0=beta0, beta1=beta1,
+                 lower=c(0,0), upper=c(100,100))$value
+     -(sum(beta0 + beta1*elev.fn(s)) - EN)
+ }
> starting.values <- c(-10, 0)
> fm <- optim(starting.values, nll, hessian=TRUE)
> cbind(Est=fm$par, SE=sqrt(diag(solve(fm$hessian)))) # estimates and SEs
            Est        SE
[1,] -5.9335547 0.2204693
[2,]  0.9545532 0.1771507
```

Maximizing the Poisson likelihood took a fraction of a second, and we obtained estimates of $\hat{\beta}_0 = -5.93$ and $\hat{\beta}_1 = 0.95$, which are very close to the data-generating values. The 95% confidence interval for $\hat{\beta}_1$ is [0.61, 1.3] and since it does not include zero, the null hypothesis that $\beta_1 = 0$, i.e. that there is no effect of elevation on density, can be rejected. In addition to testing hypotheses, these results can be used to predict population size in new regions or create predicted density surface maps by plugging the parameter estimates into Eqs. 11.2.1 and 11.2.2.

You might wonder if the results would differ if we assumed a binomial rather than a Poisson distribution for N. This can be checked using the following code:

```
> nllBin <- function(beta, M=100) {
+     beta0 <- beta[1]
+     beta1 <- beta[2]
+     EN <- cuhre(2, 1, mu, beta0=beta0, beta1=beta1,
+                 flags=list(verbose=0),
+                 lower=c(0,0), upper=c(100,100))$value
+     N <- nrow(s)
+     psi <- EN/M
+     -(sum(beta0 + beta1*elev.fn(s) - log(EN)) +
+       dbinom(N, M, psi, log=TRUE))
+ }
> cbind(Est=fmBin$par, SE=sqrt(diag(solve(fmBin$hessian)))) # est and SE
            Est        SE
[1,] -5.9339490 0.1965479
[2,]  0.9545742 0.1771962
```

which indicates that the MLEs are almost identical, and supports the claim that the prior on N has little influence in SCR models. Notice, however, that the standard error for β_0 is smaller under the binomial model than it was under the Poisson model – a difference that will dissipate as M tends toward infinity.

This example demonstrates that if we had the data we wish we had, i.e. if we knew the coordinates of the activity centers, we could easily estimate the parameters governing the underlying point process and make inferences about spatial variation in density and abundance. Unfortunately, in virtually all animal ecology studies, the locations of the N animals, or the N activity centers, cannot be directly observed. Thus we need extra information to estimate the locations of these unobserved points, which in the case of SCR, comes from the locations where each animal is captured.

11.4 FITTING INHOMOGENEOUS POINT PROCESS SCR MODELS

11.4.1 Continuous space

In this example, we will use the same set of points simulated in the previous section to generate spatial capture-recapture data. Specifically, we overlay a grid of 49 traps on the map shown in Fig. 11.2 and simulate capture histories conditional on the activity centers. Then, we will attempt to estimate the activity center locations as though we did not know where they were, as is the case in real applications. We will also estimate β_0 and β_1 as before and see how the estimates compare when the points are not actually observed. The following **R** code simulates encounter histories under a Poisson observation model (see Chapt. 9), which could be appropriate in camera trapping studies or when using other methods in which animals could be detected multiple times at a trap during a single occasion.

```
> xsp <- seq(20, 80, by=10); len <- length(xsp)
> X <- cbind(rep(xsp, each=len), rep(xsp, times=len)) # traps
> ntraps <- nrow(X); noccasions <- 5
> y <- array(NA, c(N, ntraps, noccasions)) # capture data
> sigma <- 5  # scale parameter
> lam0 <- 1   # basal encounter rate
> lam <- matrix(NA, N, ntraps)
> set.seed(5588)
> for(i in 1:N) {
+     for(j in 1:ntraps) {
+         # The object "s" was simulated in previous section
+         distSq <- (s[i,1]-X[j,1])^2 + (s[i,2] - X[j,2])^2
+         lam[i,j] <- exp(-distSq/(2*sigma^2)) * lam0
+         y[i,j,] <- rpois(noccasions, lam[i,j])
+     }
+ }
```

Now that we have a simulated capture-recapture dataset `y`, we can ~~simulation~~ the posterior distributions of the model parameters using MCMC. A commented Gibbs sampler written in **R** is available in the accompanying **R** package `scrbook` (see `?scrIPP`). This function is not meant to be an all purpose tool for fitting SCR models using MCMC. Instead, it is presented so that interested readers can better understand the computational aspects of the problem and can modify it for their purposes. The function can be used as so:

```
> fm1 <- scrIPP(y, X, M=150, 10000, xlims=c(0,100), ylims=c(0,100),
+               space.cov=elev.fn, tune=c(0.4, 0.2, 0.3, 0.3, 7))
> plot(mcmc(fm1$out))
```

which requests 10000 posterior samples and estimates the effect of the spatial covariate, elevation, on density. The argument `space.cov` accepts any spatial covariate that returns a real value for any location in the rectangular state-space defined by the `xlims` and `ylims` arguments. Currently, the function places uniform priors on the parameters σ, λ_0, β_0 and β_1, although this could easily be modified. The `tune` argument specifies the tuning parameters used in the Metropolis-within-Gibbs steps of the algorithm. These should be chosen using trial and error to achieve an acceptance rate of between 0.4 and 0.6, roughly. See Chapt. 17 for more details about MCMC.

Results of the analysis are shown in Fig. 11.3 and Table 11.1. Fig. 11.3 displays the trace plots of the Markov chains as well as the posterior distributions for three parameters.

The chains appear to converge rapidly but may need to be run longer to reduce Monte Carlo error. Summaries of the posterior distributions are presented in Table 11.1. The posterior means for β_0 and β_1 are quite similar to MLEs from the analysis in the previous section in which we assumed no observation error. However, we see that the confidence intervals are wider. With respect to the other parameters in the model, we see that all of the data generating parameter fall within the 95% credible intervals. One thing to note is that, although the point estimates for the expected and realized values of N are quite similar, the posterior for the realized value of N is more precise. This is to be expected because the uncertainty associated with the realized value of N is entirely determined by the sampling error. That is, if we could perfectly detect all of the individuals in $\mathcal{S}$, there would be no uncertainty about N. In contrast, the variance for expected value of N is composed both process error and sampling error. See Chapt. 5 and Efford and Fewster (2012) for additional discussion on the difference between realized and expected values of abundance.

Fitting continuous space inhomogeneous point process models is somewhat difficult in **BUGS** because the "IPP" prior $[\mathbf{s}_i|\boldsymbol{\beta}]$, unlike the uniform prior, is not one of the available distributions that comes with the software. It is possible to add new distributions in **BUGS**, but it is somewhat cumbersome. `secr` allows users to fit continuous space models using linear or polynomial functions of the easting and northing coordinates, but it does not accept truly continuous covariates that are functions of space. However, these are not really important limitations because discrete space versions of the model are straightforward, and virtually all spatial covariates are, or can be, defined as such.

Table 11.1. Summary of posterior distributions from SCR model with inhomogeneous point process.

Parameter	Mean	SD	2.5%	97.5%
$\sigma = 5$	5.232	0.310	4.681	5.858
$\lambda_0 = 1$	0.802	0.119	0.595	1.049
$\beta_0 = -6$	-5.856	0.254	-6.376	-5.393
$\beta_1 = 1$	0.985	0.209	0.575	1.378
$N = 41$	47.615	8.041	35.000	66.000
$\mathbb{E}(N) = 39.9$	47.551	10.992	29.837	71.332

11.4.2 Discrete space

To fit inhomogeneous point process models using covariates in discrete space, i.e. in raster format, we follow the same steps as outlined in Chapt. 9 – we define $\mathbf{s}_i$ as pixel ID, and we use the categorical distribution as a prior. This effectively changes the problem from estimating the coordinates of an activity center, to estimating the pixel in which an activity center is located. As pixel size approaches zero, these two become equivalent. A good example is found in ~~(~~Mollet et al., In review). Here we present an analysis of the simulated data shown in the Fig. 11.2. The spatial covariate, let's call it forest canopy height (CANHT), was simulated using using the code shown on the help page `ch11` in `scrbook`. The points are the number of activity centers in each pixel, generated from

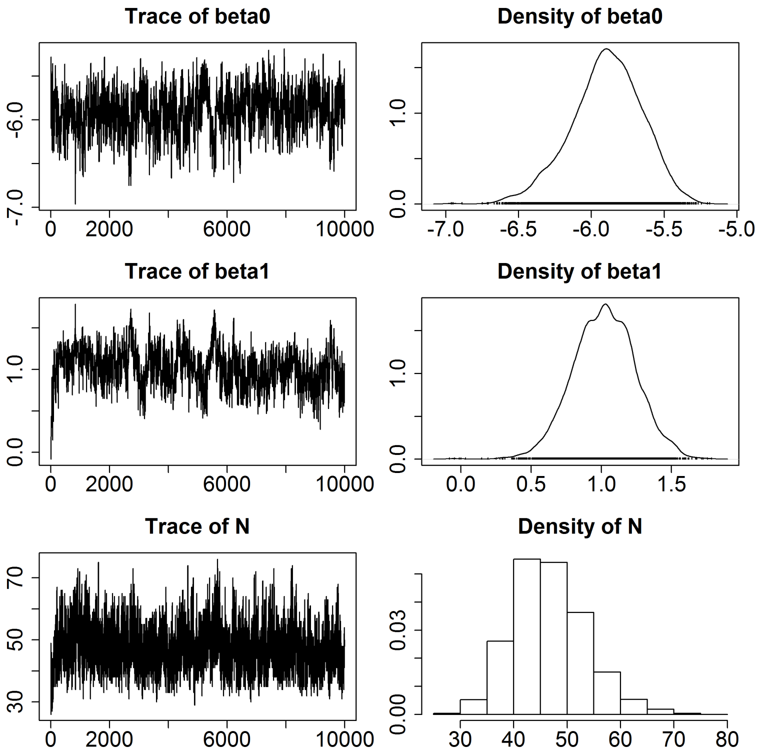


Figure 11.3. Trace plots and posterior distributions from MCMC analysis of SCR model with inhomogeneous point process. Analysis was conducted using the `scrIPP` function in the accompanying **R** package `scrbook`.

a single realization of the inhomogeneous point process model with intensity $\mu(\mathbf{x}, \boldsymbol{\beta}) = \exp(\beta_0 + \beta_1 \text{CANHT}(\mathbf{x})) \times \text{pixelArea}$, where $\beta_0 = -6$ and $\beta_1 = 1$.

The **BUGS** description of the model is shown in panel 11.1. The vector `probs[]` is the prior probability defined by Eq. 11.2.5, which is the probability that an individual's activity center is located at pixel $\mathbf{x}$. `grid` is the matrix of coordinates for each pixel.

This model can also be fit in `secr`, which refers to the raster data as a "habitat mask". The habitat mask is essentially a `data.frame` with attributes. The `data.frame` itself has 2 columns for the coordinates of each of the pixel centers. The attributes of the object include information such as the area of the pixels and the spacing between pixel centers. If there are covariates, these too are stored as an attribute of the habitat mask, and are formatted as a `data.frame` with 1 row per pixel and 1 column per covariate. Once the data have been formatted correctly, fitting the model in `secr` is as simple as:

```
> secr1 <- secr.fit(ch, model=D~canht, mask=msk)
```

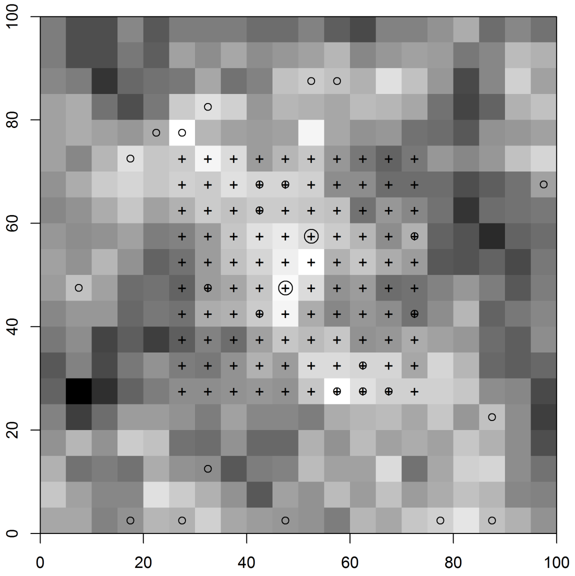


Figure 11.4. Simulated activity centers in discrete space. The spatial covariate, canopy height, is highest in the lighter areas and density increases with canopy height. A single activity center is shown as a small circle, and larger circles represent two activity centers in a pixel. Trap locations are shown as crosses.

Table 11.2. Comparison of `secr` and **JAGS** results. Point estimates from the Bayesian analysis are posterior means. Intervals are lower and upper 95% CIs.

Parameter	Truth	Software	Mean	SD	2.5%	97.5%
λ_0	1.00	**JAGS**	1.04	0.087	0.88	1.22
	1.00	`secr`	1.08	0.089	0.92	1.27
σ	10.00	**JAGS**	10.16	0.373	9.46	10.94
	10.00	`secr`	9.84	0.350	9.18	10.55
β_1	1.00	**JAGS**	1.20	0.350	0.50	1.88
	1.00	`secr`	1.09	0.316	0.47	1.71
N	30.00	**JAGS**	26.63	2.585	23.00	33.00
	30.00	`secr`	28.19	3.037	24.49	37.39
$\mathbb{E}(N)$	32.30	**JAGS**	26.39	5.048	17.25	36.96
	32.30	`secr`	28.19	6.117	18.52	42.93

where `D~canht` indicates that we want to model density as a function of canopy height, which is defined in the `msk` object. **R** code to format the data and fit the models using `secr` and **JAGS** is available in `scrbook`, found by issuing the command: `help(ch11secr-jags)`.

Results of fitting the model in **JAGS** and `secr` are shown in Table 11.2 and are similar as expected. The differences that do exist are likely due to the differences in Bayesian and frequentist estimation methods, as discussed in Chapt 3. Either answer may be "more correct" depending upon one's criteria for correctness!

11.5 ARGENTINA JAGUAR STUDY

Estimating density of large felines has been a priority for many conservation organizations, but few robust methodologies existed before the advent of SCR. Distance sampling is not feasible for such rare and cryptic species, and traditional capture-recapture methods yield estimates that are highly sensitive to the subjective choice of the effective survey area. SCR models provide a powerful alternative because density can be estimated directly and data can be collected using non-invasive methods such as camera traps or hair snares.

In this example, we demonstrate how readily density can be estimated for a globally imperiled species using SCR. Furthermore, we show how inhomogeneous point process models can be used to test important hypotheses regarding the factors affecting density. The data come from an 8-year camera-trapping study designed to assess the impacts of poaching on jaguar density in Argentina, near the borders of Brazil and Paraguay. Additional information about the study is presented in Paviolo et al. (2008) and Paviolo et al. (2009). The expected effect of poaching is a decline in jaguar density due to the direct removal of individuals and the depletion of its main prey species. To conserve jaguars and related species, protected areas have been established and three levels of protection are recognized, as depicted in Fig. 11.5. The dark gray area is the Iguazú National Park that is patrolled regularly by law enforcement officials. The medium gray areas are not protected and rarely patrolled. Finally, the light gray areas are large soybean monocultures, cities and dams which provide no suitable habitat for jaguars

To test for differences in density between the three regions, we modeled the point process intensity parameter as a function of protection status (PROTECT), which we

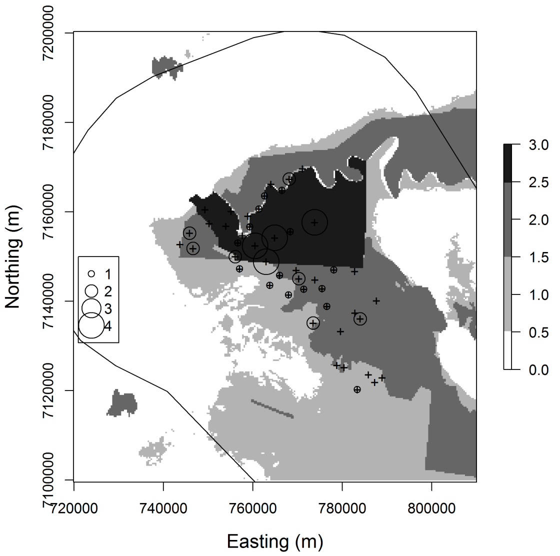


Figure 11.5. Jaguar detections at 46 camera trap stations. The three levels of protection status are no protection (light gray), some protection (gray), and Iguazú National Park (dark gray). Non-habitat (soybean monocultures) is shown in white.

treated as an ordinal variable:

$$\mu(\mathbf{x}, \boldsymbol{\beta}) = \exp(\beta_0 + \beta_1 \text{PROTECT}(\mathbf{x})) \times \text{pixelArea}.$$

We predicted that β_1 would be greater than zero, indicating that jaguar density increases with protection status. In addition to modeling spatial variation in density, we also modeled the scale parameter of the half-normal (or Gaussian) encounter model as sex-specific because male cats typically have larger home ranges than females (Sollmann et al., 2011). Since sex is an individual-specific covariate, and not observed for the individuals that were not captured, a prior distribution is required for the sex of uncaptured individuals. We used a Bernoulli prior with probability 0.5 to describe our uncertainty about sex ratio. Another equivalent option is to augment the data with an equal number of males and females and let the MCMC algorithm determine which of these individuals are actually members of the population.

An additional unique aspect of this study is the highly irregular state-space. Unlike in the examples of simulated data, the geometry of this state-space is not a simple rectangular region. Instead, it is the area south of the Iguazú River, which runs along the northern border of the park shown in dark green in Fig. 11.5, and it excludes the large soybean monocultures. Fitting models in highly convoluted spatial regions raises the question: How does one integrate Eq. 11.2.4 over this irregular space? Earlier we used the function `cuhre`

Table 11.3. Summaries of posterior distributions from the model of jaguar density. σ is the scale parameter of the half-normal detection function. λ_0 is baseline encounter rate, β_1 is the effect of protection status on jaguar density, ρ is the sex-ratio, N is population size. The last three parameters are the density estimates (jaguars/100 km^2) for the three levels of protection.

	Mean	SD	2.5%	97.5%
N	35.819	7.9749	23.0000	54.0000
D_{low}	0.906	0.3265	0.3813	1.6682
D_{med}	0.770	0.2841	0.2698	1.4392
D_{high}	1.370	0.3069	0.8315	1.9955
σ_{female}	5501.204	876.8774	4142.2756	7578.5692
σ_{male}	6452.570	915.3623	4970.3215	8505.5219
λ_0	0.006	0.0016	0.0034	0.0098
ψ	0.355	0.0937	0.1998	0.5638
β_0	-4.686	0.2602	-5.2346	-4.2129
β_1	0.174	0.3500	-0.5104	0.8649
Sex Ratio	0.489	0.0550	0.3824	0.6000

in **R** for the two-dimensional integration, but its `lower` and `upper` arguments essentially assume that the state-space is square. There are methods of transforming the state-space that might allow us to work around this problem, but once again we find that it is most convenient to work in discrete space and sum over all the pixels defining $\mathcal{S}$.

We fit the model to data from a single year in which 46 camera stations were operational, each consisting of a pair of cameras placed along roads or small trails. Forty-five detections of 16 jaguars (8 males and 8 females) were made over a 95-day sampling period. The mean number of sampling days at each camera station was 48.2. The raw capture data shown in Fig. 11.5 suggest that the highest number of captures was in the national park, but there were also several traps in the park with no captures. Furthermore, few cameras were placed far from the protected areas, making it somewhat difficult to detect differences in density. **R** code to fit the model is available in `scrbook` on the help page `jaguarDataCh11`. Parameter estimates are shown in Table 11.3.

The results indicate that efforts to protect jaguars by reducing poaching in protected areas are not working as well as hoped for. The posterior probability that $\beta_1 > 0$ was only 0.69, and the posterior mean of realized density was only 51% higher in the national park than in the unprotected area. Fig. 11.6 shows the estimated density surfaces. The first map is the expected density in each of the three values, which was computed by plugging in the posterior mean values of β_0 and β_1 into the log-linear intensity function. The second map is the realized density surface – the conditional-on-N probability distribution of the number of activity centers in each pixel of the rasterized state-space. The expected values would be used if we were interested in making inferences about other areas or time periods, whereas the realized map is the best description of the system during the study period.

We note that there is room for improvement in our analysis, and our results should be considered preliminary. The political boundaries used to demarcate protected areas are not as concrete as we might like. In reality poaching pressure is likely higher near remote park boundaries than in well-guarded park interiors. One option for addressing this would be to use a continuous measure of poaching pressure such as distance from the nearest

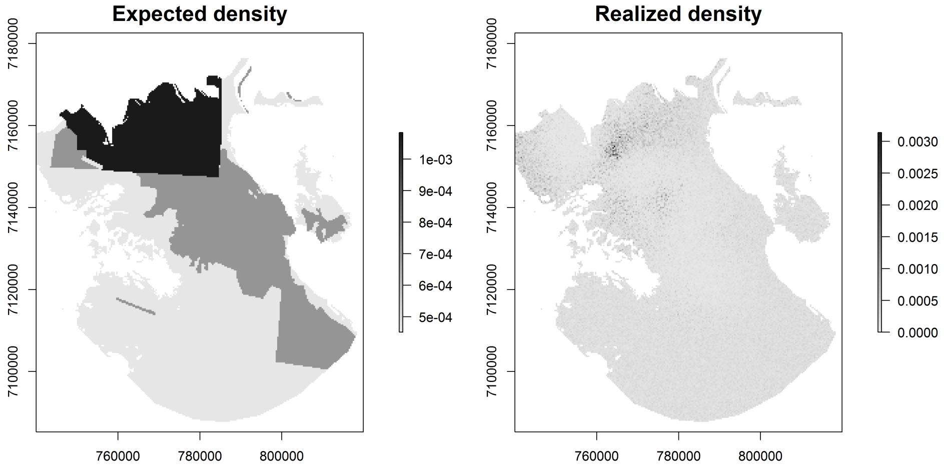


Figure 11.6. Estimated density (activity centers / pixel) surfaces from the analysis of the jaguar data.

town, or some other accessibility metric. It would also be worthwhile to model density separately for each sex because many of the detections outside of the park were of males, and thus it is possible that the sexes use habitat differently (Conde et al., 2010). Other extensions warranting investigation include treating PROTECT as a categorical rather than ordinal variable, and assessing the effects of roads and trails on jaguar movement using the methods described in Chapt. 12. Developing models for these extensions could be readily accomplished by modifying the fitting functions found in the **R** package `scrbook`.

11.6 SUMMARY AND OUTLOOK

One of the distinguishing features of spatial capture-recapture models is that they allow for inference about spatial variation in density without relying on ad hoc approaches for determining the amount of area surveyed. The approach described in this chapter involves modeling the locations of activity centers as outcomes of an inhomogeneous point process with intensity determined by covariates defined at all locations in the state-space. Covariate effects can be evaluated in exactly the same way as is done in generalized linear models, making it easy to interpret the results.

All the examples in this section included a single state-space covariate, but this was for simplicity only. Including multiple covariates poses no additional challenges. Similarly, additional model structure such sex-specific encounter rate parameters or behavioral responses can be accommodated and fit using `secr`, **BUGS**, or by extending the functions in `scrbook`. It is also possible to consider covariates that affect both density and ecological distance as will be described in the next chapter. The ramifications of this are enormous for applied ecological research and conservation efforts because researchers can use capture-recapture data to identify areas where both density and landscape connectivity are high (Royle et al., 2013). Addressing such questions is simply not possible using standard, non-spatial capture-recapture methods.

Although we focused on modeling the point process intensity as a function of covariates, other options for fitting inhomogeneous models exist (Illian et al., 2008). Cox processes are models in which the point process intensity is a function of spatial random effects. Such methods are useful for accommodating overdispersion, but it seems unlikely that most SCR datasets could support such complexity. Gibbs processes are another important class of models that are distinguished by the interactions of points. Although little work has been done on such models in the context of SCR studies (Reich et al., 2012), we expect they will receive more attention because they can be used to model processes such territoriality (points repel one another) or aggregation (points attract one another). Neyman-Scott processes are another option for modeling aggregation or clustering, and could be useful for studying gregarious species.

```
model{
sigma ~ dunif(0, 20)
lam0 ~ dunif(0, 5)
beta0 ~ dunif(-10, 10)
beta1 ~ dunif(-10, 10)
for(j in 1:nPix) {
  mu[j] <- exp(beta0 + beta1*CANHT[j])*pixArea
  probs[j] <- mu[j]/EN
}
EN <- sum(mu[]) # Expected value of N, E(N)
psi <- EN/M
for(i in 1:M) {
  z[i] ~ dbern(psi)
  s[i] ~ dcat(probs[])
  x0g[i] <- grid[s[i],1]
  y0g[i] <- grid[s[i],2]
  for(j in 1:ntraps) {
    dist[i,j] <- sqrt(pow(x0g[i]-traps[j,1],2) + pow(y0g[i]-traps[j,2],2))
    lambda[i,j] <- lam0*exp(-dist[i,j]*dist[i,j]/(2*sigma*sigma)) * z[i]
    y[i,j] ~ dpois(lambda[i,j])
    }
  }
N <- sum(z[]) # Realized value of N
}
```

Panel 11.1: **BUGS** model specification for the inhomogeneous point process model in discrete space. A nearly equivalent formulation would involve omitting β_0 and modeling the expected number of activity centers as $\mathbb{E}(N) = M\psi$ with $\psi \sim \text{Uniform}(0, 1)$.