CHAPTER

Alternative Observation Models

9

In previous chapters we considered various models of *encounter probability*, both in terms of parametric functions of distance and also in terms of covariate models (Chapter 7 and elsewhere). However, we have so far only considered a specific probability model for the observations (we'll call this the "observation model")—the Bernoulli observation model which, in `secr` (Efford, 2011a), is the *proximity detector* model. This assumes that individual and trap-specific encounters are independent Bernoulli trials.

In this chapter, we focus on developing additional observation models. The observation model could be thought of as being determined by the type of device—or the type of "detector" using the terminology of `secr` (Efford, 2011a). We consider models that apply when observations are not binary and, in some cases, that do not require independence of the observations. For example, if sampling devices can detect an individual some arbitrary number of times during an interval, then it is natural to consider observation models for encounter frequencies, such as the Poisson model. Another type of encounter device is the "multi-catch" device (Efford et al., 2009a), which is a physical device that can capture and hold an arbitrary number of individuals. A typical example is a mist net for birds (Borchers and Efford, 2008). It is natural to regard observations from these kinds of studies as independent multinomial observations. A related type of device that produces *dependent* multinomial observations are the so-called *single-catch* traps (Efford, 2004; Efford et al., 2009a). The canonical example are small-mammal live traps which catch and hold a single individual. Competition among individuals for traps induces a complex dependence structure among individual encounters. To date, no formal inference framework has been devised for this method although it stands to reason that the independent multinomial model should be a good approximation in some situations (Efford et al., 2009a). We analyze a number of examples of these different observation models using **JAGS** and also the **R** package `secr` (Efford, 2011a).

9.1 Poisson observation model

The models we analyze in Chapter 5 assumed binary observations—i.e., standard encounter history data—so that individuals are captured at most one time in a trap on

Spatial Capture-Recapture. http://dx.doi.org/10.1016/B978-0-12-405939-9.00009-8
© 2014 Elsevier Inc. All rights reserved.

any given sample occasion. This makes sense for many types of DNA sampling (e.g., based on hair snares) because distinct visits to sampled locations or devices within a sampling occasion cannot be differentiated. However, for some encounter devices, or methods, the potential number of encounters is *not* fixed, and so it is possible to encounter an individual some arbitrary number of times during any particular sampling episode. That is, we might observe encounter frequencies $y_{ijk} > 1$ for individual i, trap j, and sampling interval k. As an example, if a camera device is functioning properly it may be programed to take photos every few seconds if triggered. For a second example, suppose we are searching a quadrat or length of trail for scat, we may find multiple samples from the same individual. Therefore, we seek observation models that accommodate such encounter frequency data. In general, any discrete probability mass function could be used for this purpose, including the standard models for count data used throughout ecology, the Poisson, and negative binomial. Here we focus on using the Poisson model only although other count frequency models are possible for SCR models (Efford et al., 2009b).

Let y_{ijk} be the frequency of encounter for individual i in trap j during occasion k, then assume:

$$y_{ijk} \sim \text{Poisson}(\lambda_{ij}),$$

where the expected encounter frequency λ_{ij} depends on both individual and trap (we could, of course, also make λ time-dependent). As we did in the binary model of Chapter 5, we now seek to model the expected value of the observation (which was p_{ij} in Chapter 5) as a function of the individual activity center $\mathbf{s}_i$. We propose

$$\lambda_{ij} = \lambda_0 g(\mathbf{x}_j, \mathbf{s}_i),$$

where $g(\mathbf{x}, \mathbf{s})$ is any positive valued function, such as the negative exponential or the bivariate Gaussian kernel, and λ_0 is the baseline encounter rate—the expected number of encounters if a trap is placed precisely at an individual's home-range center (note: in `secr` the notation for this is g_0). Then, $\lambda_0 g(\mathbf{x}_j, \mathbf{s}_i)$ is the expected encounter rate in trap $\mathbf{x}_j$ for an individual having activity center $\mathbf{s}_i$. Note that

$$\log(\lambda_{ij}) = \log(\lambda_0) + \log(g(\mathbf{x}_j, \mathbf{s}_i)).$$

Equating $\alpha_0 \equiv \log(\lambda_0)$, and, if $g(\mathbf{x}, \mathbf{s}) \equiv \exp(-d(\mathbf{x}, \mathbf{s})^2/(2\sigma^2))$ (i.e., the Gaussian encounter probability model), then:

$$\log(\lambda_{ij}) = \alpha_0 - \alpha_1 d(\mathbf{x}_j, \mathbf{s}_i)^2, \qquad (9.1.1)$$

where $\alpha_1 = 1/(2\sigma^2)$, which is the same linear predictor as we have seen for the Bernoulli model in Chapter 5. This Poisson SCR model is therefore a type of Poisson generalized linear mixed model (GLMM).

We can accommodate covariates at the level of the individual-, trap-, or sample-occasion by including them on the baseline encounter rate parameter λ_0. For example, if C_j is some covariate that depends on trap only, then we express the relationship between λ_0 and C_j as:

$$\log(\lambda_{0,ijk}) = \alpha_0 + \alpha_2 C_j$$

and therefore covariates on the logarithm of baseline encounter probability appear also as linear effects on λ_{ij}. In general, covariates might also affect the coefficient on the distance term (α_1) (e.g., sex of individual). We don't get into too much discussion of general covariate models here, because the same principles apply as we discussed in Chapters 7 and 8.

For models in which we do not have covariates that vary across the sample occasions k, we can aggregate the observed data by the property of compound additivity of the Poisson distribution (if x and y are iid Poisson with mean λ then $x + y$ is Poisson with mean 2λ). Therefore,

$$y_{ij} = \left(\sum_{k=1}^{K} y_{ijk} \right) = \text{Poisson}(K \lambda_0 g(\mathbf{x}_j, \mathbf{s}_i)).$$

We see that K and λ_0 serve the same role as affecting the base encounter rate. Since the observation model is the same, probabilistically speaking, for all values of K, evidently we need only $K = 1$ "survey" from which to estimate model parameters (Efford et al., 2009b). We know this intuitively, as sampling by multiple traps serves as replication in SCR models. This has great practical relevance to the conduct of capture-recapture studies and the use of SCR models. For example, if individuality is obtained by genetic information from scat sampling, one should only have to carry out a single spatial sampling of the study area. However, one must be certain that sufficient spatial recaptures will be obtained so that effective parameter estimation is possible.

9.1.1 Poisson model of space usage

It is natural to interpret the Poisson encounter model as a model of space usage resulting from movement of individuals about their home range (Section 5.4). Imagine we have perfect samplers in every pixel of the landscape so that whenever an individual moves from one pixel to another, we can record it. Let m_{ij} be the number of times individual i was recorded in pixel j (i.e., it selected or used pixel j). Then, we might think of the Poisson model for the observed *use* frequencies:

$$m_{ij} \sim \text{Poisson}(\lambda_0 g(\mathbf{x}_j, \mathbf{s}_i)),$$

where λ_0 is related to the baseline movement rate of the animal (how often it moves). This model of space usage gives rise to the standard resource selection function (RSF) models (see Chapter 13). But now suppose our samplers are not perfect but, rather, record only a fraction of the resulting visits. A sensible model is

$$y_{ij} | m_{ij} \sim \text{Binomial}(m_{ij}, p).$$

The marginal distribution of y_{ij} is:

$$y_{ij} \sim \text{Poisson}(p_0 g(\mathbf{x}_j, \mathbf{s}_i)),$$

where p_0 is a composite of the movement rate and conditional detection probability p. Therefore, we see that encounters accumulate in proportion to the frequency of outcomes of an individual using space (or "selecting resources").

We introduced an interpretation of SCR models in terms of movement and space usage in Section 5.4, and it is one of the main underlying concepts of SCR models that is not present in ordinary capture-recapture models. As we noted there, the underlying model of space usage is only as complex as the encounter probability model, which has been, so far in this book, only symmetric and stationary (does not vary in space). We generalize this model of space usage substantially in Chapter 13.

9.1.2 Poisson relationship to the Bernoulli model

There is a sense in which the Poisson and Bernoulli models can be viewed as consistent with one another. Note that under the Poisson model, the relationship between the expected count and the probability of "at least 1 detection" is given by

$$\Pr(y > 0) = 1 - \exp(-\lambda), \tag{9.1.2}$$

where $\mathbb{E}(y) = \lambda$. Therefore, if we equate the event "encountered" with the event that the individual was captured at least once time under the Poisson model, i.e., $y > 0$, then it would be natural to set $p_{ij} = \Pr(y > 0)$ according to Eq. (9.1.2). That is, we can use Eq. (9.1.2) as the model for encounter probability for binary observations. This is the "hazard rate" model in distance sampling.

In fact, as λ gets small, the Poisson model is a close approximation to the Bernoulli model in the sense that outcomes concentrate on $\{0, 1\}$, i.e., $\Pr(y \in \{0, 1\}) \rightarrow 1$ as $\lambda \rightarrow 0$. Indeed, under the Poisson model, $\Pr(y > 0) \rightarrow \lambda$ for small values of λ. This phenomenon is shown in Figure 9.1 where the left panel shows a plot of $\lambda_{ij} = \lambda_0 g(\mathbf{x}_j, \mathbf{s}_i)$ vs. distance, and superimposed on that is a plot of $p_{ij} = 1 - \exp(-\lambda_{ij})$ vs. distance, for values $\lambda_0 = 0.1$ and $\sigma = 1$, and the right

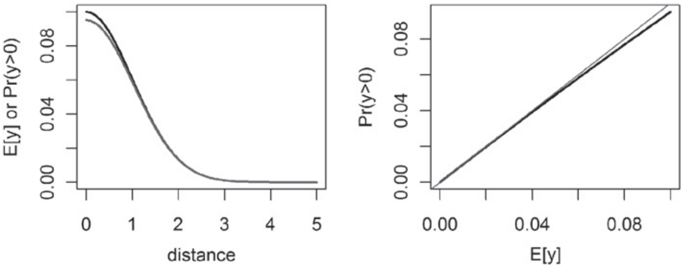

FIGURE 9.1

Poisson approximation to the binomial model. As the Poisson mean approaches 0, then $\Pr(y > 0)$ under the Poisson model approaches λ and therefore $y \sim \text{Poisson}(\lambda)$ is well approximated by a Bernoulli model with parameter $p = \lambda$.

panel shows a plot of $\Pr(y > 0)$ vs. $\mathbb{E}(y)$. We see that the two quantities are practically indistinguishable. To evaluate the closeness of the approximation, you can execute the following **R** commands, which we used to produce Figure 9.1:

```
> x <- seq(0.001,5, ,200)
> lam0 <- .1
> sigma <- 1
> lam <- lam0*exp(-x*x/(2*sigma*sigma))

> par(mfrow=c(1,2))
> p1 <- 1-exp(-lam)
> plot(x, lam, ylab="E[y] or Pr(y>0)",xlab="distance",type="l",lwd=2)
> lines(x,p1,lwd=2,col="red")
> plot(lam, p1, xlab="E[y]",ylab="Pr(y>0)",type="l",lwd=2)
> abline(0,1,col="red")
```

To summarize, if y is Poisson, then, as λ gets small,

$$\begin{gathered}\Pr(y > 0) \approx \mathbb{E}(y), \\ 1 - \exp(-\lambda_0 g(\mathbf{x}, \mathbf{s})) \approx \lambda_0 g(\mathbf{x}, \mathbf{s}).\end{gathered} \tag{9.1.3}$$

What all of this suggests is that if we have very few observations > 1 in our SCR data set, then we won't lose much information by using the Bernoulli model. On the other hand, the Poisson model may have some advantages in terms of analytic or numerical tractability in some cases. Further, this approximation explains the close correspondence we have found between these two versions of the Gaussian encounter probability model (Section 5.4). Namely, the Gaussian hazard model and the Gaussian encounter probability model are close approximations because $1 - \exp(-\lambda) \approx \lambda$ if λ is small.

Even in such cases where the Poisson and Bernoulli models are not quite equivalent, we might choose to truncate individual encounter frequencies to binary observations anyhow (transforming counts to 0/1 is called "quantizing"). We might do this intentionally in some cases, such as when the distinct encounter events are highly dependent as often happens in camera trap studies when the same individual moves back and forth in front of a camera during a short period of time. But sometimes, truncation is a feature of the sampling. For example, in the case of bear hair snares, the number of encounters might be well approximated by a Poisson distribution but we cannot determine unique visits and so only get to observe the binary event "$y > 0$." In this case, we might choose to model the encounter probability for the binary encounter using Eq. (9.1.3). This is equivalent to the complementary log-log link model from Chapter 5:

$$\text{cloglog}(p_{ij}) = \log(\lambda_0) + \log(g(\mathbf{x}, \mathbf{s})),$$

where $\text{cloglog}(u) = \log(-\log(1-u))$.

9.1.3 A cautionary note on modeling encounter frequencies

Other models for counts might be appropriate. For example, ecologists are especially fond of negative binomial models for count data because it accommodates overdispersion (White and Bennetts, 1996; Kéry et al., 2005; Ver Hoef and Boveng, 2007) but other models for excess-Poisson variation are possible. For example, we might add a normally distributed random effect to the linear predictor for λ (Coull and Agresti, 1999).

As a general rule, we favor the Bernoulli observation model even if our sampling scheme produces encounter frequencies. The main reason is that, with frequency data, we are forced to confront a model choice problem (i.e., Poisson, negative binomial, log-normal mixture) that is wholly unrelated to the fundamental space usage process that underlies the genesis of many types of SCR data. Repeated encounters over short time intervals are not likely to be the result of independent encounter events. E.g., an individual moving back and forth in front of a camera yields a cluster of observations that is not informative about the underlying process of space usage. Similarly, in scat surveys dogs are used to locate scats which are processed in the laboratory for individuality (Kohn et al., 1999; MacKay et al., 2008; Thompson et al., 2012). The process of local scat deposition is not strictly the outcome of movement or space usage but rather the outcome of complex behavioral considerations as well as dependence in detection of scat by dogs. For example, dogs find (or smell) one scat and then are more likely to find one or more nearby ones, if present, or they get into a den or latrine area and find many scats. The additional assumption required to model variation in observed frequencies (i.e., conditional on location) provides no information about space usage and density, and we feel that the model selection issue should therefore be avoided.

9.1.4 Analysis of the Poisson SCR model in BUGS

We consider the simplest possible model here in which we have no covariates that vary over sample occasions $k = 1, 2, \dots, K$ so that we work with the aggregated individual- and trap-specific encounters:

$$y_{ij} \sim \left(\sum_{k=1}^{K} y_{ijk}\right) \sim \text{Poisson}(K\lambda_0 g(\mathbf{x}_j, \mathbf{s}_i))$$

and we consider the bivariate normal form of $g(\mathbf{x}, \mathbf{s})$:

$$g(\mathbf{x}, \mathbf{s}) = \exp(-d(\mathbf{x}, \mathbf{s})^2/(2\sigma^2))$$

so that

$$\log(\lambda_{ij}) = \alpha_0 - \alpha_1 d(\mathbf{x}_j, \mathbf{s}_i)^2,$$

where $\alpha_0 = \log(\lambda_0)$ and $\alpha_1 = 1/(2\sigma^2)$.

As usual, we approach Bayesian analysis of these models using data augmentation (Section 4.2). Under data augmentation, we introduce a collection of all-zero encounter histories to bring the total size of the data set up to M, and a corresponding set of data augmentation variables $z_i \sim \text{Bernouli}(\psi)$. Then the observation model is specified conditional on z according to:

$$y_{ij} \sim \text{Poisson}(z_i K \lambda_{ij}),$$

which evaluates to a point mass at $y = 0$ if $z = 0$. In other words, the observation model under data augmentation is a zero-inflated Poisson model which is easily analyzed by Bayesian methods, e.g., in one of the **BUGS** dialects or, alternatively, using likelihood methods where the same principles as in Chapter 6 apply.

9.1.5 Simulating data and fitting the model

Simulating a sample SCR data set under the Poisson model requires only a couple of minor modifications to the procedure we used in Chapter 5 (see the function simSCR0). In particular, we modify the block of code that defines the model to be that of $\mathbb{E}(y)$ and not $\Pr(y = 1)$, and we change the random variable generator from rbinom to rpois:

```
##
## S =activity centers and traplocs defined as in simSCR0()
##
## Compute distance between activity centers and traps:
> D <- e2dist(S,traplocs)

## Define parameter values:
> alpha0 <- -2.5
> sigma <- 0.5
> alpha1 <- 1/(2*sigma*sigma)

## Encounter probability model:
> muy <- exp(alpha0)*exp(-alpha1*D*D)

## Now generate the encounters of every individual in every trap
> Y <-matrix(NA,nrow=N,ncol=ntraps)
> for(i in 1:nrow(Y)){
   Y[i,] <- rpois(ntraps,K*muy[i,])
  }
```

We modified our simulation code from Chapter 5 to simulate Poisson encounter frequencies for each trap, and we use this new code to generate, and subsequently analyze, an ideal data set using **BUGS**. The Poisson simulator function simPoissonSCR is available in the scrbook package (it can produce 3-d encounter history data too, although we don't do that here). Here is an example of simulating a data set, harvesting the required data objects, and doing the data augmentation:

```
## Simulate data and extract data elements
##
> data <- simPoissonSCR(discard0=TRUE,rnd=2013)
> y <- data$Y
> nind <- nrow(y)
> X <- data$traplocs
> K <- data$K
       > J <- nrow(X)
       > xlim <- data$xlim
       > ylim <- data$ylim
```

```
## Data augmentation
> M <- 200
> y <- rbind(y,matrix(0,nrow=M-nind,ncol=ncol(y)))
> z <- c(rep(1,nind),rep(0,M-nind))
```

The process for fitting the model in **WinBUGS** or **JAGS** is identical to what we've done previously in Chapter 5. In particular, we set up some starting values, package the data and inits, identify the parameters to be monitored, and then send everything off to our MCMC engine. Here are the commands for fitting the Poisson observation model (these commands are also shown in the help file for `simPoissonSCR`):

```
## Starting values for activity centers
##
> sst <- X[sample(1:J,M,replace=TRUE),]
> for(i in 1:nind){
  if(sum(y[i,])==0) next
  sst[i,1] <- mean( X[y[i,]>0,1] )
  sst[i,2] <- mean( X[y[i,]>0,2] )
  }
## Dithered a little bit from trap locations
> sst <- sst + runif(nrow(sst)*2,0,1)/8
> data <- list (y=y,X=X,K=K,M=M,J=J,xlim=xlim,ylim=ylim)
> inits <- function(){
   list (alpha0=rnorm(1,-2,.4),alpha1=runif(1,1,2),s=sst,z=z,psi=.5)
   }
> parameters <- c("alpha0","alpha1","N","D")
```

Next, we write the **BUGS** model to an external file:

```
> cat("
model{
 alpha0 ~ dnorm(0,.1)
 alpha1 ~ dnorm(0,.1)
 psi ~ dunif(0,1)

 for (i in 1:M){
   z[i] ~ dbern(psi)
   s[i,1] ~ dunif(xlim[1],xlim[2])
   s[i,2] ~ dunif(ylim[1],ylim[2])
   for(j in 1:J){
      d[i,j] <- pow(pow(s[i,1]-X[j,1],2) + pow(s[i,2]-X[j,2],2),0.5)
      y[i,j] ~ dpois(lam[i,j])
      lam[i,j] <- z[i]*K*exp(alpha0)*exp(- alpha1*d[i,j]*d[i,j])
   }
 }
 N <- sum(z[])
 D <- N/64
}
",file = "SCR-Poisson.txt")
```

To fit the model we execute bugs in the usual way:

```
> library(R2WinBUGS)
> out1 <- bugs (data, inits, parameters, "SCR-Poisson.txt", n.thin=1,
                n.chains=3,n.burnin=1000,n.iter=2000,working.dir=getwd(),
                debug=TRUE)
```

Or, using **JAGS**, we would do something like this:

```
> library(rjags)
> jm <- jags.model("SCR-Poisson.txt", data=data, inits=inits,
                   n.chains=3, n.adapt=1000)
> out2 <- coda.samples(jm, parameters, n.iter=1000, thin=1)
```

Summarizing the output from the **WinBUGS** run produces the following:

```
> print(out1,digits=2)
Inference for Bugs model at "SCR-Poisson.txt", fit using WinBUGS,
 3 chains, each with 2000 iterations (first 1000 discarded)
 n.sims = 3000 iterations saved
           mean    sd   2.5%    25%    50%    75%  97.5% Rhat n.eff
alpha0    -2.57  0.19  -2.95  -2.69  -2.57  -2.44  -2.19 1.00  2600
alpha1     2.34  0.36   1.69   2.08   2.32   2.57   3.12 1.00  3000
N        114.13 15.25  87.97 103.00 113.00 124.00 147.00 1.01   370
D          1.78  0.24   1.37   1.61   1.77   1.94   2.30 1.01   370
deviance 329.95 21.92 290.00 314.20 329.50 344.40 375.80 1.00  1700
...
[..some output deleted..]
...
```

We see that the estimates are close to the data generating values: $\alpha_0 = 2.5$, $\alpha_1 = 2.0$, and $N = 100$, as we should expect.

9.1.6 Analysis of the wolverine study data

We reanalyzed the data from the wolverine camera trapping study that were first introduced in Section 5.9. We modified the **R** script from the function wolvSCR0 to fit the Poisson model (see the help file for wolvSCR0pois). Executing this function produces the results shown in Table 9.1. The results are almost indistinguishable from the Bernoulli model fitted previously, where we had a posterior mean for N of 59.84 and σ was 0.64. You can edit the script wolvSCR0pois to obtain more posterior samples, or modify the model in some way.

9.1.7 Count detector models in the secr package

The **R** package secr can fit Poisson or negative binomial encounter frequency models. The formatting of data and structure of the analysis proceeds in a similar fashion to the Bernoulli model described in Section 6.5, except that we specify the detector="count" option when the traps object is created. The setup proceeds as follows:

Table 9.1 Results of fitting the SCR model with Poisson observation model to the wolverine camera trapping data. Posterior summaries were obtained using **WinBUGS** with 3 chains, each with 6,000 iterations, discarding the first 1,000 as burn-in, to yield a total of 15,000 posterior samples.

Parameter	Mean	SD	2.5%	50%	97.5%	Rhat	n.eff
N	60.12	11.91	40.00	59.00	87.00	1.00	630
D	5.80	1.15	3.86	5.69	8.39	1.00	630
$\log(p_0)$	−2.89	0.17	−3.22	−2.89	−2.57	1.00	5000
λ_0	0.06	0.01	0.04	0.06	0.08	1.00	5000
σ	0.64	0.06	0.54	0.64	0.76	1.00	730
ψ	0.30	0.07	0.19	0.30	0.45	1.00	650

```
> library(secr)
> library(scrbook)
> data(wolverine)

> traps <- as.matrix(wolverine$wtraps)
> dimnames(traps) <- list(NULL,c("trapID","x","y",
+                         paste("day",1:165,sep="")))
> traps1 <- as.data.frame(traps[,1:3])
> trapfile1 <- read.traps(data=traps1,detector="count")
```

You can proceed with analysis of these data and compare/contrast with the Bayesian analysis given above, or the results of the Bernoulli model fitted in Chapter 6.

9.2 Independent multinomial observations

Several types of encounter devices yield multinomial observations in which an individual can be caught in a single trap during a particular encounter occasion, but traps might catch any number of individuals. Mist netting is the canonical example of such a "multi-catch" device (Efford et al., 2009a). Also, some kinds of bird or mammal cage-traps, hold multiple animals, as do pit-fall traps, which are commonly used for many species of herptiles. Another type of sample method that might be viewed (in some cases) as a multi-catch device are area-searches for example, of reptiles where we think of a small polygon as the "trap"—we could get multiple individuals in the same plot but not, in the same sample occasion, at different plots. The key features of this independent multinomial or multi-catch model are: (1) capture of an individual in a trap is *not* independent of its capture in other traps, because initial capture precludes capture in any other trap and (2) individuals behave independently of one another, so whether a trap captures some individual doesn't have an affect on whether it captures another. A situation in which the second assumption is violated arises under "single-catch" trap systems, which we address in Section 20.1.5.

In this case we assume the observation $\mathbf{y}_{ik}$ for individual i during sample occasion k is a multinomial observation that consists of a sequence of 0's and a single 1 indicating the trap of capture, or "not captured". For the "not captured" event we define an additional outcome, by convention element $J+1$ of the vector. As an example, if we capture an individual in trap 2 during some occasion of a study involving $J=6$ traps. Then, the multinomial observation has length $J+1=7$, and the observation is $\mathbf{y}_i = (0,1,0,0,0,0,0)$. An individual not captured at all would have the observation vector $(0,0,0,0,0,0,1)$. If we sample for five occasions in all and the individual is also caught in trap 4 during occasion 3, but otherwise uncaptured, then the five encounter observations for that individual are as follows:

```
occasion |-------trap ------|  "not captured"
         1    2    3   4    5   6       7
         ----------------------------------------------
   1     0    1    0   0    0   0       0
   2     0    0    0   0    0   0       1
   3     0    0    0   1    0   0       0
   4     0    0    0   0    0   0       1
   5     0    0    0   0    0   0       1
```

Statistically we regard the *rows* of this data matrix as *independent* multinomial trials.

Analogous to our previous Bernoulli and Poisson models, we seek to construct the multinomial cell probabilities for each individual as a function of *where* that individual lives, through its center of activity $\mathbf{s}$. Thus, we suppose that

$$\mathbf{y}_{ik}|\mathbf{s}_i \sim \text{Multinomial}(1, \boldsymbol{\pi}(\mathbf{s}_i)), \tag{9.2.1}$$

where $\boldsymbol{\pi}(\mathbf{s}_i)$ is a vector of length $J+1$, where $\pi_{i,J+1}$, the last cell, corresponds to the probability of the event "not captured". Now we have to construct these cell probabilities in some meaningful way that depends on each individual's $\mathbf{s}$. We use the standard multinomial logit with distance as a covariate:

$$\pi_{ij} = \frac{\exp(\alpha_0 - \alpha_1 d_{ij})}{1 + \sum_j \exp(\alpha_0 - \alpha_1 d_{ij})}$$

for $j = 1, 2, \ldots, J$, and for $J+1$, i.e., "not captured",

$$\pi_{i,J+1} = \frac{\exp(0)}{1 + \sum_j \exp(\alpha_0 - \alpha_1 d_{ij})}$$

or, more commonly, we use d_{ij}^2 to correspond to our Gaussian kernel model for encounter probability. Whatever function of distance we use in the construction of multinomial probabilities will have a direct correspondence to the standard encounter probability models we used in the Bernoulli or Poisson models as well (see Section5.4).

It is convenient to express these multinomial models short-hand as follows, e.g., for the Gaussian encounter probability model:

$$\text{mlogit}(\pi_{ij}) = \alpha_0 - \alpha_1 d_{ij}^2.$$

In this way we can refer to models with covariates in a more concise way. For example, a model with a trap-specific covariate, say C_j, is:

$$\text{mlogit}(\pi_{ij}) = \alpha_0 - \alpha_1 d_{ij}^2 + \alpha_2 C_j,$$

or we could include occasion-specific covariates too, such as behavioral response.

A statistically equivalent distribution to the multinomial is the *categorical* distribution. If $\mathbf{y}$ is a multinomial trial with probabilities $\boldsymbol{\pi}$ then the *position* of the non-zero element of $\mathbf{y}$ is a categorical random variable with probabilities $\boldsymbol{\pi}$. We express this for SCR models as

$$\mathbf{y}|\mathbf{s} \sim \text{Categorical}(\boldsymbol{\pi}(\mathbf{s})).$$

In the SCR context, the categorical version of the multinomial trial corresponds to the *trap of capture*. Using our example above with 6 traps we could as well say y_{ik} is a categorical random variable with possible outcomes $(1, 2, 3, 4, 5, 6, 7)$, where outcome $y = 7$ corresponds to "not captured." Obviously, how this is organized or labeled is completely arbitrary, although it is convenient to use the categorical formulation. Therefore, for our illustration in the previous table, $y_{i1} = 2, y_{i2} = 7, y_{i3} = 4$, and so on.

For simulating and fitting data in the **BUGS** engines we will typically use the categorical representation of the model because it is somewhat more convenient. We have found that fitting multinomial models in **WinBUGS** is less efficient than **JAGS** (Royle and Converse, in review), which we use in the subsequent examples involving multinomial observation models.

9.2.1 Multinomial resource selection models

The multinomial probabilities in Eq. (9.2.2) look similar to the multinomial resource selection function (RSF) model for telemetry data (Manly et al., 2002; Lele and Keim, 2006). This suggests how we might model landscape or habitat covariates using such methods—i.e., by including them as explicit covariates in a larger multinomial model for "use"—which, if we take the product of use with encounter, produces a model for the observable encounter data. This leads naturally to the development of models that integrate RSF data from telemetry studies with SCR data (Royle et al., 2012a), which is the topic of Chapter 13.

9.2.2 Simulating data and analysis using JAGS

We're going to show the nugget of a simulation function, which is implemented in the function `simMnSCR` found in the **R** package `scrbook`. The first lines of the following **R** code make use of some things that you need to define, but we omit them here (e.g., `xlim`, `ylim` are the boundaries of the state-space, `N` is the population size, etc.):

```
##
## Simulate random activity centers:
```

```
##       (first define N, xlim, ylim, etc.)
##
> S <- cbind(runif(N,xlim[1],xlim[2]),runif(N,ylim[1],ylim[2]))
## Distance from each individual to each trap
> D <- e2dist(S,traplocs)

## Set paramter values
> sigma <- 0.5
> alpha0 <- −1
> alpha1 <- −1/(2*sigma*sigma)

## make an empty data matrix and fill it up with data
> Ycat <- matrix(NA,nrow=N,ncol=K)
>  for(i in 1:N){
    for(k in 1:K){
    lp <- alpha0 + alpha1*D[i,]*D[i,]
    cp <- exp(c(lp,0))
    cp <- cp/sum(cp)
    Ycat[i,k] <- sample(1:(ntraps+1),1,prob=cp)
   }
  }
```

We save the data in the matrix `Ycat` to clarify that it is the categorical observation representing “trap of capture”. The matrix `Ycat` here has N rows and, to do an analysis that mimics a real situation, we would have to discard the uncaptured individuals. The function `simMnSCR` will also simulate data that includes a behavioral response, which will be the typical situation in small-mammal trapping problems (see Converse and Royle, 2012, for details).

Here we use our function `simMnSCR` to simulate a data set with $K = 7$ occasions. We’ll run the model using **JAGS** which we have found is much more effective for this class of models. We get the data set up for analysis by augmenting the size of the data set to $M = 200$. In addition, we choose starting values for **s** and the data augmentation variables z. For starting values of **s** we cheat a little bit here and use the true values for the observed individuals and then augment the $M \times 2$ matrix **S** with $M - n$ randomly selected activity centers. In practice, we recommend using the average encounter location of observed individuals as starting values for their activity centers. Our function `spiderplot` returns the mean observed location of the `nind` encountered individuals. The parameters input to `simMnSCR` are the intercept α_0, $\sigma = \sqrt{1/(2\alpha_1)}$ for the Gaussian encounter probability model, and α_2 is the behavioral response parameter. The data simulation and setup proceeds as follows:

```
> set.seed(2013)
> parms <- list(N=100,alpha0= -.40, sigma=0.5, alpha2= 0)
> data <- simMnSCR(parms, K=7, ssbuff=2)
> nind <- nrow(data$Ycat)

> M <- 200
> Ycat <- rbind(data$Ycat,matrix(nrow(data$X)+1,nrow=(M-nind),ncol=data$K))
        > Sst <- rbind(data$S,cbind(runif(M-nind,data$xlim[1],data$xlim[2]),
                                   runif(M-nind,data$ylim[1],data$ylim[2])))
        > zst <- c(rep(1,160),rep(0,40))
```

The model specification is not much more complicated than the binomial or Poisson models given previously. The main consideration is that we define the cell probabilities for each trap $j = 1, 2, \ldots, J$ and then define the last cell probability, $J+1$, for "not captured", to be the complement of the sum of the others. The code is shown in Panel 9.1. In the last lines of code we specify N and density, D, as derived parameters.

To fit the model, we need to package everything up (initial values, parameters, data) and send it off to **JAGS** to build an MCMC simulator for us (these commands are shown in the help file for `simMnSCR`). In addition to the usual data objects, we also pass the limits of the assumed rectangular state-space (`ylim`, `xlim`, both 1×2

```
model{
psi ~ dunif(0,1)
alpha0 ~ dnorm(0,10)
sigma ~dunif(0,10)
alpha1 <-  1/(2*sigma*sigma)

for(i in 1:M){
  z[i] ~ dbern(psi)
  S[i,1] ~ dunif(xlim[1],xlim[2])
  S[i,2] ~ dunif(ylim[1],ylim[2])
  for(j in 1:ntraps){
    #distance from capture to the center of the home range
    d[i,j] <- pow(pow(S[i,1]-X[j,1],2) + pow(S[i,2]-X[j,2],2),1)
  }
  for(k in 1:K){
    for(j in 1:ntraps){
      lp[i,k,j] <- exp(alpha0 - alpha1*d[i,j])*z[i]
      cp[i,k,j] <- lp[i,k,j]/(1+sum(lp[i,k,]))
    }
    cp[i,k,ntraps+1] <- 1-sum(cp[i,k,1:ntraps])  # last cell = not captured
    Ycat[i,k] ~ dcat(cp[i,k,])
  }
}

N <- sum(z[1:M])
A <- ((xlim[2]-xlim[1])*trap.space)*((ylim[2]-ylim[1])*trap.space)
D <- N/A
}
```

PANEL 9.1

BUGS model specification for the independent multinomial observation model. For data simulation and model fitting see the help file `?simMnSCR` in the **R** package `scrbook`.

vectors) and the scale of the standardized units, called `trap.space` here because we typically will define the trap coordinates to be an integer grid. If the trap spacing is 10 m and we want units of density in terms of individuals per m^2, then we input `trap.space=10`. The analysis is carried out as follows:

```
> inits <- function(){  list (z=zst,sigma=runif(1,.5,1) ,S=Sst) }

# Parameters to monitor
> parameters <- c("psi","alpha0","alpha1","sigma","N","D")

# Bundle the data. Note this reuses "data"
> data <- list (X=data$X,K=data$K, trap.space=1,Ycat=Ycat,M=M,
               ntraps=nrow(data$X),ylim=data$ylim,xlim=data$xlim)

> library(R2jags)
> out <- jags (data, inits, parameters, "model.txt", n.thin=1,
               n.chains=3, n.burnin=1000, n.iter=2000)
```

The posterior summaries are provided in the following **R** output (recall that $N = 100$, $\alpha_0 = -.40$, and $\sigma = 0.5$):

```
> out
Inference for Bugs model at "model.txt", fit using jags,
 3 chains, each with 2000 iterations (first 1000 discarded)
 n.sims = 3000 iterations saved
         mu.vect sd.vect    2.5%     25%     50%     75%   97.5%  Rhat n.eff
D          1.873   0.189   1.531   1.750   1.859   2.000   2.250 1.006  1300
N        119.867  12.107  98.000 112.000 119.000 128.000 144.000 1.006  1300
alpha0    -0.435   0.151  -0.738  -0.535  -0.439  -0.331  -0.146 1.004   580
alpha1     2.195   0.286   1.658   2.004   2.180   2.372   2.785 1.003  2400
psi        0.599   0.069   0.465   0.552   0.599   0.645   0.739 1.006  1400
sigma      0.480   0.032   0.424   0.459   0.479   0.500   0.549 1.003  2400
deviance 892.164  21.988 850.922 877.417 891.561 906.246 937.728 1.003   950

[... output deleted ...]
```

We see that the posterior means of each parameter are very close to the data generating values, considering the high degree of posterior uncertainty.

9.2.3 Multinomial relationship to the Poisson

The multinomial is related to the Poisson encounter rate model by a conditioning argument. Let y_{ij} be the number of encounters for individual i in trap j. If $y_{ij} \sim \text{Poisson}\,(\lambda_{ij})$, then, conditional on the *total* number of captures (i.e., across all traps), $y_i = \sum_j y_{ij}$, the trap encounter frequencies are multinomial with probabilities

$$\pi_{ij} = \frac{\lambda_{ij}}{\sum_j \lambda_{ij}}$$

for $j = 1, 2, \ldots, J$. Or equivalently the *trap of capture* is categorical with probabilities π_{ij} as given above. Under the Gaussian kernel model, these probabilities are:

$$\pi_{ij} = \frac{\exp(-\alpha_1 d(\mathbf{x}_j, \mathbf{s}_j)^2)}{\sum_j \exp(-\alpha_1 d(\mathbf{x}_i, \mathbf{s}_i)^2)}, \tag{9.2.2}$$

where, we note, the intercept α_0 has been canceled from both the numerator and denominator. This makes sense because, here, these probabilities describe the trap-specific capture probabilities *conditional on capture*. Therefore, the model is not completely specified, absent a model for the "overall" probability of encounter or the expected frequency of captures, say ϕ_i. Depending on how we specify a model for this quantity ϕ_i, we can reconcile it directly with the Poisson model. Let $y_{i.}$ be the total number of encounters for individual i and suppose $y_{i.}$ has a Poisson distribution with mean ϕ_i. Then, marginalizing Eq. (9.2.1) over the Poisson distribution for $y_{i.}$ produces the original set of iid Poisson frequencies with probabilities:

$$\lambda_{ij} = \phi_i \pi_{ij}$$

for $j = 1, 2, \ldots, J$. In particular, if we suppose that $\phi_i = \sum_j \exp(\alpha_0 - \alpha_1 d(\mathbf{x}, \mathbf{s})^2)$ then the marginal distribution of y_{ij} is Poisson with mean $\exp(\alpha_0 - \alpha_1 d(\mathbf{x}, \mathbf{s})^2)$, equivalent to Eq. (9.1.1).

In summary, the Poisson and multinomial models are equivalent in how they model the distribution of captures among traps. It stands to reason that, if the encounter rate of individuals is low, we could use the Poisson and multinomial models interchangeably. In fact, based on our discussion in Section 9.1.2 above we could use any of the binomial/Poisson/multinomial models with little ill effect when encounter rate is low.

9.2.4 Avian mist-netting example

We analyze data from a mist-netting study of ovenbirds (*Seiurus aurocapilla*), conducted at the Patuxent Wildlife Research Center, Laurel MD, by D.K. Dawson and M.G. Efford. The data from this study are available in the `secr` package and have been analyzed previously by Efford et al. (2004), see also Borchers and Efford (2008). Forty-four mist-nets spaced 30 m apart on the perimeter of a 600-m × 100-m rectangle were operated on 9 or 10 non-consecutive days in late May and June for 5 years from 2005 to 2009. The ovenbird data can be loaded as follows:

```
> library(secr)
> data(ovenbird)
```

The data set consists of adult ovenbirds caught during sampling in each of the 5 years (one ovenbird was killed in 2009 indicated by a negative net number in the encounter data file). As with most mist-netting studies, nets are checked multiple times during a day (e.g., every hour during a morning session). However, for this data set, the within-day recaptures are not included so each bird has at most a single capture per day. Therefore, the multinomial model (detector type "multi" in `secr`) is appropriate. Although several individuals were captured in more than 1 year, this information is not used in the models presently offered in `secr`, but we do make use of it in the development of open models in Chapter 16.

9.2.4.1 *Multiple sample sessions*

Up to this point we have only dealt with a basic closed population sampling situation consisting of repeated sample occasions on a single population of individuals using a single array of traps. In practice, many studies produce repeated samples over longer periods of time over which demographic closure isn't valid, or at different locations where the populations are completely distinct. We adopt the `secr` terminology of *session* for such replication by groups in time or space, and the models are *multi-session* models, although we think of such models as being relevant to any stratified population (see Chapter 14). We introduced `secr`'s multi-session models in Section 6.5.4. In the case of the ovenbird data, sampling was carried out in multiple years, with a number of sample occasions within each year (9 or 10), a type of data structure commonly referred to as "the robust design" (Pollock, 1982). In this context, it stands to reason that there is recruitment and mortality happening across years. In Chapter 16 we model these processes explicitly, but here, we provide an analysis of the data that does not require explicit models for recruitment and survival, regarding the yearly populations as independent strata, and fitting a multi-session model.

When the sessions represent explicit time periods, the multi-session model of `secr` can be thought of as a type of open population model. In particular, a special case of open models arises when we assume N_t (time-specific population sizes) are independent from one time period or session to the next—this can be thought of as a "random temporary emigration" model of the Kendall et al. (1997) variety, and this is the multi-session model implemented in `secr`. In particular, by assuming that N_t is Poisson with mean Λ_t, one can model variation in abundance among sessions based on the Poisson-integrated likelihood in which parameters of Λ_t appear directly in the likelihood as we noted in Section 6.5.4.[1] We provide an analysis (below) of the ovenbird data here using the multi-session models in `secr` with a multinomial observation model. We formalize the multi-session model approach from a Bayesian perspective using data augmentation in Chapter 14 (Converse and Royle, 2012; Royle and Converse, in review).

A third way to develop models for stratified or grouped populations, not based on multi-session models, but that is convenient in **BUGS**, is to regard the data from each session as an independent data set with its own N_t parameter, and do T distinct data augmentations. Because each N_t is regarded as a free parameter, independent of the other parameters, we'll call this the non-parametric multi-session model to distinguish it from the multi-session model which assumes the N_t are related to one another by having been generated from a common Poisson distribution. By augmenting each year separately in the same **BUGS** model specification, we avoid making explicit model assumptions about the N_t parameters.

Below, and elsewhere in the book, we demonstrate these three approaches to analyzing grouped/stratified data using the ovenbird data: (1) In the following section,

[1] We do not know of `secr` documentation that states this (or contradicts it). We think this is what is being done, based partially on conversations or emails with M.G. Efford, D.L. Borchers, the various publications on `secr`, and our own thinking about it.

we provide the non-parametric multi-session model with unconstrained N_t; (2) we demonstrate the Poisson multi-session models from `secr` both here (following section) and in Chapter 14 from a Bayesian standpoint; (3) later, in Chapter 16, we provide a fully dynamic "spatial Jolly-Seber" model and apply it to the ovenbird data.

9.2.4.2 Analysis in JAGS

The ovenbird data are provided as a multi-session `capthist` object `ovenCH`, which, by regarding years as independent strata or sessions, allows for the fitting of the multi-session model. For doing a Bayesian analysis in one of the **BUGS** engines (we use **JAGS** here) there are a number of ways to structure the data and describe the model. We can analyze either a 2-d data set with all years (data augmented) "stacked" into a data set of dimension $(5M) \times 10$ (5 years, M = size of the augmented data set, $K = 10$ replicate sample occasions). Or, we could produce a 3-d array $(M \times J \times K)$. We adopted the former approach, analyzing the data as a 2-d array and creating an additional categorical variable for "year" to indicate which stratum (year) each record goes with. We used $M = 100$ per year for a total size of the augmented data of 500. The **BUGS** model specification is shown in Panel 9.2.

Data on individual sex is included with `secr`, but we provide an analysis (model specification shown in Panel 9.2) of a single model for all adults, constant σ across years, constant p_0, and year-specific values of N_t (and hence D_t). For the habitat mask, we used a rectangular state-space created by buffering the mist net grid by 200 meters. There was a single loss-on-capture which we accounted for by fixing $p = 0$ for all subsequent encounters of that individual (indicated by the binary variable `dead`, as shown in Panel 9.2). We have an **R** script in the `scrbook` package called `SCRovenbird`, so you can see how to set up the data and run the model. Executing the script `SCRovenbird` produces the posterior summaries given in Table 9.2. Here, density is in units of birds per ha. The posterior mean of σ is about 76 m, and there is considerable variability in density over the 5-year period, with density peaking at 1.2 birds/ha in year 3, although there is considerable posterior uncertainty. The $\hat{\text{R}}$ statistics suggest satisfactory mixing and convergence of the Markov chains.

9.2.4.3 Analysis in `secr`

Included with the ovenbird data are a number of models fitted as examples. Those include:

```
ovenbird.model.1  fitted secr model -- null
ovenbird.model.1b fitted secr model -- g0 net shyness
ovenbird.model.1T fitted secr model -- g0 time trend within years
ovenbird.model.h2 fitted secr model -- g0 finite mixture
ovenbird.model.D  fitted secr model -- trend in density across years
```

The model fit objects provided in `secr` are based on the use of the habitat mask. To make the analyses consistent with our previous analysis in **JAGS**, we refit all of the models here without the habitat mask. The re-analysis proceeds as follows, changing the "trend in density across years" model to allow for year-specific density:

```
model{
 alpha0 ~ dnorm(0,.1)
 sigma ~ dunif(0,200)
 alpha1 <- 1/(2*sigma*sigma)

 A <- ((xlim[2]-xlim[1]))*((ylim[2]-ylim[1]))
 for(t in 1:5){
   N[t] <- inprod(z[1:bigM],yrdummy[,t])
   D[t] <- (N[t]/A)*10000  # Put in units of per ha
   psi[t] ~ dunif(0,1)
 }

 for(i in 1:bigM){  # bigM = total size of jointly augmented data set
   z[i] ~ dbern(psi[year[i]])
   S[i,1] ~ dunif(xlim[1],xlim[2])
   S[i,2] ~ dunif(ylim[1],ylim[2])

 for(j in 1:ntraps){  # X = trap locations, S = activity centers
    d2[i,j] <- pow(pow(S[i,1]-X[j,1],2) + pow(S[i,2]-X[j,2],2),1)
  }
 for(k in 1:K){
   Ycat[i,k] ~ dcat(cp[i,k,])
   for(j in 1:ntraps){
     lp[i,k,j] <- exp(alpha0 - alpha1*d2[i,j])*z[i]*(1-dead[i,k])
     cp[i,k,j] <- lp[i,k,j]/(1+sum(lp[i,k,1:ntraps]))
   }
   cp[i,k,ntraps+1] <- 1-sum(cp[i,k,1:ntraps])  # Last cell = not captured
  }
 }
}
```

PANEL 9.2

BUGS model specification for the non-parametric multi-session model in which each N_t is independent of each other. The implied prior (by data augmentation) is that $N_t \sim$ Uniform(0,100). To fit this model to the ovenbird data, see `?SCRovenbird` in the **R** package `scrbook`.

```
## Fit constant-density model
> ovenbird.model.1 <- secr.fit(ovenCH)
## Fit net avoidance model
> ovenbird.model.1b <- secr.fit(ovenCH, model =    list (g0 ~ b))
## Fit model with time trend in detection
> ovenbird.model.1T <- secr.fit(ovenCH, model =    list (g0 ~ T))
## Fit model with 2-class mixture for g0
```

Table 9.2 Posterior summary statistics for the ovenbird mist-netting data based on the independent multinomial (“multi-catch”) encounter process model. Parameters ψ,N, and D are indexed by year. MCMC was done using `jags` with 3 chains, each with 11,000 iterations, discarding the first 1,000, for a total of 30,000 posterior samples. Note the parameter α_1 is 0 to 3 decimal places, but $\sigma = (\sqrt{1/2\alpha_1})$.

Parameter	Mean	SD	2.5%	50%	97.5%	Rhat	n.eff
D[1]	0.983	0.211	0.636	0.966	1.455	1.002	1,900
D[2]	1.023	0.209	0.673	1.003	1.492	1.001	7,100
D[3]	1.208	0.238	0.807	1.186	1.749	1.004	740
D[4]	0.896	0.195	0.575	0.880	1.333	1.002	3,000
D[5]	0.753	0.177	0.465	0.734	1.149	1.001	4,000
α_0	−3.479	0.160	−3.797	−3.477	−3.171	1.005	490
α_1	0.000	0.000	0.000	0.000	0.000	1.003	1,100
σ	76.214	6.125	65.569	75.758	89.360	1.003	1,100
N[1]	80.423	17.283	52.000	79.000	119.000	1.002	1,900
N[2]	83.685	17.077	55.000	82.000	122.000	1.001	7,100
N[3]	98.822	19.483	66.000	97.000	143.000	1.004	740
N[4]	73.288	15.962	47.000	72.000	109.000	1.002	3,000
N[5]	61.589	14.468	38.000	60.000	94.000	1.001	4,000
ψ[1]	0.403	0.092	0.246	0.395	0.606	1.002	1,600
ψ[2]	0.419	0.091	0.260	0.412	0.620	1.001	6,400
ψ[3]	0.494	0.102	0.315	0.486	0.723	1.004	760
ψ[4]	0.368	0.086	0.221	0.361	0.555	1.002	3,200
ψ[5]	0.310	0.079	0.178	0.302	0.485	1.002	3,500

```
> ovenbird.model.h2 <- secr.fit(ovenCH, model =    list (g0 ~ h2))
## Fit a model with session (year)-specific Density
> ovenbird.model.DT <- secr.fit(ovenCH, model =    list (D ~ session))
```

All of these can be fitted easily in **JAGS** but the model we fitted previously is roughly equivalent to the last model, `ovenbird.model.DT`, because we allowed for year-specific population sizes (and hence density). So, we’ll compare our results from **JAGS** to that model. The `secr` output is extensive and so we do not reproduce it completely here. By default, it summarizes the trap information for each year, encounter information, and then output for each year. Here is an abbreviated version for `ovenbird.model.DT`:

```
> print(ovenbird.model.DT,digits=2)
```

```
secr.fit( capthist = ovenCH, model = list(D ~ session), buffer = 300 )
secr 2.3.1, 14:46:52 23 Jan 2013
```

```
$‘2005’
Object class      traps
Detector type     multi
Detector number   44
Average spacing   30.27273 m
x-range           −50 49 m
y-range           −285 285 m
```

```
[... deleted ...]
```

```
           2005 2006 2007 2008 2009
Occasions     9   10   10   10   10
Detections   35   42   52   30   33
Animals      20   22   26   19   16
Detectors    44   44   44   44   44

Model              : D~session g0~1 sigma~1
Fixed (real)       : none
Detection fn       : halfnormal
Distribution       : poisson
N parameters       : 7
Log likelihood     : -1119.845
AIC                : 2253.689
AICc               : 2254.868
```

```
[... deleted ...]
```

To do model selection we use the handy helper-function `AIC` as follows (output edited to fit on the page):

```
AIC (ovenbird.model.1, ovenbird.model.1b, ovenbird.model.1T,
   ovenbird.model.h2, ovenbird.model.DT)


                  model detectfn npar  logLik    AIC     AICc   dAICc
 ovenbird.model.1T [edited output]  4  −1111.85 2231.70  2232.10   0.00
 ovenbird.model.1b      ....       4  −1117.61 2243.22  2243.63  11.52
 ovenbird.model.h2      ....       3  −1121.16 2248.32  2248.57  16.46
 ovenbird.model.1       ....       5  −1119.76 2249.52  2250.14  18.03
 ovenbird.model.DT      ....       7  −1119.84 2253.68  2254.86  22.75
```

We see that our `DT` model is way down at the bottom of the list. Instead, the model with a time trend (within-season) in detection probability is preferred, followed by a behavioral response. We encourage you to adapt the **JAGS** model specification for such models which is easily done (see Chapter 7 for many examples). We provide the summary results for the model having `D ∼ session` as follows:

```
> print(ovenbird.model.DT,digits=2)

secr.fit(capthist = ovenCH, model = list(D ~ session), buffer = 300)
secr 2.3.1, 14:46:52 23 Jan 2013

[...deleted...]
```

```
Fitted (real) parameters evaluated at base levels of covariates

 session = 2005
       link estimate SE.estimate    lcl    ucl
D       log    0.920       0.228  0.571  1.484
g0    logit    0.028       0.004  0.021  0.037
sigma   log   78.566       6.379 67.025 92.095

 session = 2006
       link estimate SE.estimate    lcl    ucl
D       log    0.963       0.238  0.598  1.553
g0    logit    0.028       0.004  0.021  0.037
sigma   log   78.566       6.379 67.025 92.095

 session = 2007
       link estimate SE.estimate    lcl    ucl
D       log    1.139       0.282  0.706  1.836
g0    logit    0.028       0.004  0.021  0.037
sigma   log   78.566       6.379 67.025 92.095

 session = 2008
       link estimate SE.estimate    lcl    ucl
D       log    0.832       0.206  0.516  1.341
g0    logit    0.028       0.004  0.021  0.037
sigma   log   78.566       6.379 67.025 92.095

 session = 2009
       link estimate SE.estimate    lcl    ucl
D       log    0.701       0.173  0.435  1.130
g0    logit    0.028       0.004  0.021  0.037
sigma   log   78.566       6.379 67.025 92.095
```

The point estimates (MLEs) of density are uniformly lower than the Bayesian estimates (posterior means) shown in Table 9.2. We expect some difference in this direction due to small-sample skew of the posterior. In addition, there may be slight differences due to the fact that `secr` multi-session model assumes that the N_t have a Poisson prior, but the implementation in **JAGS** using data augmentation is based on a binomial prior. The estimated σ is very similar between the **JAGS** analysis and `secr`.

9.3 Single-catch traps

The classical animal trapping experiment is based on a physical trap that captures a single animal and holds that individual until subsequent handling by a biologist. This type of observation model—the "single-catch" trap—was the original situation considered in the context of spatial capture-recapture by Efford (2004). Nowadays, capture-recapture data are more often obtained by other methods (DNA from hair snares, or scat sampling, camera traps, etc.) but nevertheless the single-catch traps are still widely used in small-mammal studies (Converse et al., 2006b; Converse and Royle, 2012) and other situations.

The single-catch model is basically a multinomial model but one in which the number of available traps is reduced as each individual is captured. As such, the constraints on the joint likelihood for the sample of n encounter histories are very complicated. As a result, at the time of this writing, there has not been a formal development of either likelihood or Bayesian analysis of this model, and applications of SCR models to single-catch systems have used the independent multinomial model as an approximation (see below).

Nevertheless, we can make some progress to describing the basic observation model formally. In particular, if we imagine that all of the individuals captured queued up at the beginning of the capture session to draw a number indicating their order of capture, then there is a nice conditional structure resulting from a "removal process" operating on the traps. The first individual captured has the multinomial observation model:

$$\mathbf{y}_1 \sim \text{Multinomial}(\boldsymbol{\pi}_1),$$

whereas the second individual captured also has a multinomial encounter probability model but with the trap that captured the first individual removed. We might express this as:

$$\mathbf{y}_2 \sim \text{Multinomial}(\boldsymbol{\pi}_2),$$

where

$$\pi_{2j} = \frac{(1 - y_{1j}) \times \exp(\alpha_0 - \alpha_1 d_{ij}^2)}{\sum_j (1 - y_{1j}) \times \exp(\alpha_0 - \alpha_1 d_{ij}^2)}$$

and so on for $i = 3, 4, \ldots, n$. In a certain way, this model is a type of local behavioral response model but where the response is to other individuals being captured. Evidently, the **order of capture** is relevant to the construction of these multinomial cell probabilities. More generally, the *time* of capture of an individual in any trapping interval will affect the encounter probability of subsequently captured individuals, but we think that order of capture might lead to a practical approximation to the single-catch process (this is how we simulate the data in our function `simScSCR`). In the simulation of single-catch data, we randomly ordered the population of individuals for each sample occasion, and then cycled through them, turning off each trap if an individual was captured in it.

9.3.1 Inference for single-catch systems

For the single-catch model, we argued that the observations have a multinomial type of observation model, but the multinomial observations have a unique conditional dependence structure among them owing to the "removal" of traps as they fill up with individuals. Thus, competition for single-catch traps renders the independence assumptions for the independent multinomial model invalid. However, as Efford et al. (2009a) noted, we expect "bias to be small when trap saturation (the proportion of traps occupied) is low. Trap saturation will be higher when population density is high..."

relative to trap density, or when net encounter probability is high. Efford et al. (2009a) did a limited simulation study and found essentially no effective bias and concluded that estimators of density from the misspecified independent multinomial model are robust to the mild dependence induced when trap saturation is low. Naturally then, we expect that the Poisson model could also be an effective approximation under the same set of circumstances.

In the **R** package `scrbook` we provide a function for simulating data from a single-catch system (function `simScSCR`) and fitting the misspecified model (`example(simScSCR)`) in **JAGS** so that you can evaluate the effectiveness of this misspecified model for situations that interest you.

9.3.2 Analysis of Efford's possum trapping data

We provide an analysis here of data from a study of brushtail possums in New Zealand. The data are available with the **R** package `secr` (Efford et al., 2009a); see the help file `?possum` after loading the `secr` package. Originally, the data were analyzed by Efford et al. (2005), and a detailed description of the data set is available in the help file, from which we summarize:

> *Brushtail possums (Trichosurus vulpecula) are an unwanted invasive species in New Zealand. Although most abundant in forests, where they occasionally exceed densities of 15/ha, possums live wherever there are palatable food plants and shelter.*

To load the possum data, execute the following commands:

```
> library(secr)
> data(possum)
```

The study area encompasses approximately 300 ha, and 180 live traps were organized in five distinct grids, shown in Figure 9.2. Each square arrangement of traps consisted of 36 traps with a spacing of 20 m. Thus the squares are 180 m on a side. Individuals were captured, tagged, and released over 5 days during April, 2002. A noteworthy aspect of this study is that it involves replicated grids selected in some fashion from within a prescribed region. From an analysis standpoint, we could adopt the use of the multi-session models, which we used previously to analyze the ovenbird data. This would be useful if we had covariates at the trapping grid level that we wanted to model. Alternatively, we could pool the data from all of the grids and analyze them jointly as if they were based on a single trapping grid (with 180 traps), which is clearly a reasonable approach in this case.

The data file `possumCH` contains 112 encounter histories, and we analyze those here although the last eight of those are recaptures treated as new individuals.[2] The encounter process is not strictly a single-catch multinomial process because, as noted in the `possum` help file, "One female possum was twice captured at two sites on one

[2] M. Efford, personal communication

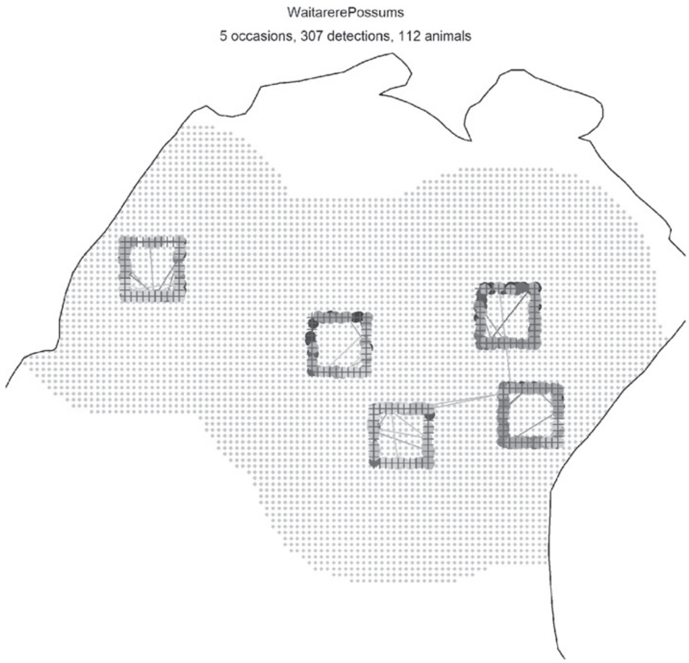

FIGURE 9.2

Trapping grids used in possum study from Efford et al. (2005), data are contained in the **R** package `secr` (Efford, 2011a), refer to the help file `?possum` for additional details of this study.

day, having entered a second trap after being released; one record in each pair was selected arbitrarily and discarded", which is a similar situation to what might happen in bird mist-net studies, as a bird might fly into a net upon release from another. By discarding the two extra-capture events, we can satisfactorily view these data as single-catch data, for which `secr` uses the independent multinomial likelihood (M. Efford, pers. comm.). If multiple same-session captures were common, then it might be worth developing a model describing the number of captures of individual i during sample occasion k, in order to make use of all captures.

For our Bayesian analysis here, we used a rectangular state-space, which doesn't account for any geographic boundaries of the survey region, but we note that a customized habitat mask is included in `secr` and it could be used in a Bayesian analysis. Whether or not we use the mask is probably immaterial as long as we understand the predictions of N or D over the water don't mean anything biological and we probably wouldn't report such predictions. The **JAGS** model specification is based on that of the ovenbird analysis given previously, so we don't reproduce the model here. The

Table 9.3 Results of fitting the independent multinomial observation model to the possum trapping data. Strictly speaking, the trapping device is a "single-catch" trap, and the model represents an intentional misspecification. Density is reported in individuals per ha (Dha). Posterior summaries were obtained using **JAGS** with 3 chains, each with 2,000 iterations, discarding the first 1,000 as burn-in, to yield a total of 3,000 posterior samples.

Parameter	Mean	SD	2.5%	50%	97.5%	Rhat	n.eff
N	235.407	17.435	204.000	235.000	270.000	1.009	340
Dha	1.549	0.115	1.343	1.547	1.777	1.009	340
α_0	−0.935	0.167	−1.270	−0.934	−0.605	1.007	870
α_1	0.000	0.000	0.000	0.000	0.000	1.001	2,800
σ	52.020	2.675	47.067	51.933	57.585	1.001	2,800
ψ	0.783	0.062	0.666	0.782	0.903	1.008	340

R/JAGS script is called `SCRpossum`, which is in the `scrbook` package. The results are summarized in Table 9.3.

The estimated density (posterior mean) is about 1.53 possums/ha. To obtain the `secr` results for the equivalent null model, we execute the following command.

```
> secr.fit(capthist = possumCH, trace = F)
```

which produces (edited) summary output:

```
[... some output deleted ...]

 Fitted (real) parameters evaluated at base levels of covariates
        link   estimate    SE.estimate   lcl         ucl
 D      log     1.6988930  0.17352645    1.3913904   2.0743547
 g0     logit   0.1968542  0.02256272    0.1563319   0.2448321
 sigma  log    51.4689114  2.59981905   46.6204139  56.8216500

 [... some output deleted ...]
```

As we've discussed previously, there are many reasons for why there might be differences between Bayesian and likelihood estimates. For now we just observe that the estimated density is certainly in the ballpark (compared to those in Table 9.3), and so, too, is the estimated σ.

9.4 Acoustic sampling

The last decade has seen an explosion of technology that benefits the study of animal populations. This includes DNA sampling methods that allow for identification from hair or scat, camera trapping and identification software that allow efficient sampling of many mammals, and the resulting statistical technology that helps us to make sense of such data (Borchers and Efford, 2008; Royle and Young, 2008; Efford et al., 2009b; Gopalaswamy et al., 2012b; Sollmann et al., 2013a; Chandler and Royle, 2013). One other extremely promising technology is that of acoustic sampling using microphones or recording devices. That is, instead of having cameras record encounters, or humans pick up scat, we can establish an array of (usually) electronic recording devices which, instead of establishing a visual identity of individuals, record a vocal expression of each individual. In this context, Efford et al. (2009b) referred to audio recorders as "signal strength proximity detectors" to distinguish them from other types of proximity detections, including camera traps, which are *visual* proximity detector. Using audio records, the spatial pattern of the *signal strength* at the different audio recorders or microphones can be used for inference about density (Dawson and Efford, 2009; Efford et al., 2009b) in the same way as the spatial pattern of detections is used in the types of SCR models we have discussed so far. The basic technical formulation of these models comes from Efford et al. (2009b), and it was applied to a field study of birds by Dawson and Efford (2009). In that study, recording devices were organized in groups of 4 (in a square pattern), with an array of 5 × 15 such clusters, separated by 100 m (300 total recorder locations). This data set, called `signalCH`, is provided with the `secr` package along with some sample analyses and help files. The document `secr-sound.pdf` (Efford and Dawson (2010) that comes with the `secr` package), which you can access directly from the main help file (`?secr`).

Our development here mostly follows Efford et al. (2009b), but we change some notation to be consistent with our previous material. Let $S(\mathbf{x}, \mathbf{u})$ be the strength of a signal emanating from location $\mathbf{u}$, as recorded by a device at location $\mathbf{x}$. Just as ordinary SCR models represent a model of *encounter frequency* as a function of distance, the acoustic SCR model is a model of sound attenuation as a function of distance. In particular, the acoustic models assumes that S (or a suitable transformation) declines with distance d from the origin of the sound to the recording device. In the context of spatial sampling of animals, the origin is the actual location of some individual animal, and the recording device is something we nailed to a tree, or mounted on a post. For example, a model of sound attenuation used by Dawson and Efford (2009) is the following:

$$S(\mathbf{x}, \mathbf{u}) = \alpha_0 + \alpha_1 d(\mathbf{x}, \mathbf{u}) + \epsilon, \tag{9.4.1}$$

where $\epsilon \sim \text{Normal}(0, \sigma_s^2)$. In many standard situations, S will be measured in decibels, which can be any value on the real line. In this model, the parameters α_0, α_1, and σ_s^2 are to be estimated. We abbreviate the set of parameters by $\boldsymbol{\theta}$.

The basic structure of an acoustic SCR study is not really much different from ordinary SCR studies. Just as ordinary SCR models require that individuals be encountered at >1 trap, these acoustic models require that individuals be heard at >1 recorder. Therefore, the acoustic signals (calls or vocalizations) must be reconcilable and, in fact, reconciled successfully by the investigator. In practice, this would require associating signals that occur at the same instant with the same individual (or making a decision one way or the other). Further, if individuals are actively moving during the sample period (that recorders are functioning) then individuals might be double-counted, thereby biasing estimates of density. In general, the models produce an estimate of density of *sources*, and how that is interpreted depends on whether individuals are stationary or mobile, and other things. In particular, if multiple survey occasions are used (e.g., on different days), then modeling movement of individuals would be essential in order to interpret estimates of density meaningfully. Models that allow some movement should be possible (see Chapters 15 and 16).

9.4.1 The signal strength model

We assert that an individual is detected if S exceeds a threshold, c. The reason for introducing this threshold c is that sound recorders will always record some background sound, and so effective use of the acoustic SCR models requires specification of the threshold of measured signal below which the record is censored (non-detection occurs) and below which the recorded sound is assumed to be background noise. So we assert that an individual is detected if $S > c$, which occurs with probability $\Pr(S > c)$, the encounter probability. To expand on and formalize this, let S_{ij} be the observed value of S for animal i at detector j. The encounter probability is $\Pr(S_{ij} > c)$, which is $\Pr(S_{ij} > c) = 1 - \Pr(S_{ij} < c)$. Expressed in terms of the cumulative distribution function,

$$1 - \Pr\left(\frac{(S_{ij} - \mathbb{E}(S))}{\sigma_s} < \frac{(c - \mathbb{E}(S))}{\sigma_s}\right).$$

Calculation of this requires evaluation of the CDF of a standard normal variate say, $\eta = (S_{ij} - \mathbb{E}(S))/\sigma_s$, being less than $\gamma(\boldsymbol{\theta}) = (c - \mathbb{E}(S))/\sigma_s$, which is a function of all the parameters $\alpha_0, \alpha_1, \sigma_s^2$ and the individual location $\mathbf{u}$ and detector location $\mathbf{x}$. We'll identify it by $\gamma(\boldsymbol{\theta}, \mathbf{x}, \mathbf{u})$ when we need to be explicit about those things. We can compute $\Pr(S_{ij} > c) = 1 - \Pr(\eta < \gamma(\boldsymbol{\theta}, \mathbf{x}, \mathbf{u}))$ easily using any software package including **R** which has a standard function, `pnorm`, for computing the normal cdf. To be more concise, we'll use the $\Phi()$ to represent the normal cdf. Therefore, an individual is encountered whenever $S_{ij} > c$, which happens with probability $\Pr(S_{ij} > c) = 1 - \Phi(\gamma(\boldsymbol{\theta}, \mathbf{x}, \mathbf{u}))$.

Naturally, this quantity should depend on *where* an individual is located at the time of recording—what we call its instantaneous location, **u**, to distinguish it from its home range center **s** (but we outline a model below that distinguishes between **u** and **s**), and the detector location. The probability of detection is therefore

$$p_{ij} = p(\alpha_0, \alpha_1, \sigma | \mathbf{x}_j, \mathbf{u}_i) = 1 - \Phi(\gamma(\boldsymbol{\theta}, \mathbf{x}, \mathbf{u})),$$

where $\mathbf{u}_i$ is the instantaneous location of individual i and $\mathbf{x}_j$ is the location of trap j. We'll suppose here that the random variables $\mathbf{u}_i$ have state-space $\mathcal{U}$.[3]

The observations from an acoustic survey are the signal strength measurements, and the likelihood of the observed signal strength from individual i at detection device j can be specified by noting that the likelihood is the normal pdf for the observed

[3] We use $\mathcal{U}$ here to avoid confusion with definition of signal strength, S. However, $\mathcal{U}$ is the same state-space as $\mathcal{S}$ in the rest of the book.

signal *if* the signal strength is $> c$ and, otherwise, the contribution to the likelihood is $\Phi(\gamma(\boldsymbol{\theta}, \mathbf{x}, \mathbf{u}))$ (see Eq. 8 of Efford et al. (2009b)):

$$\Pr(S_{ij}|\mathbf{u}_i) = \Phi(\gamma(\boldsymbol{\theta}, \mathbf{x}, \mathbf{u}))^{1-I(S_{ij}>c)}\text{Normal}(S_{ij}; \alpha_0, \alpha_1, \sigma_s, \mathbf{x}_j, \mathbf{u}_i)^{I(S_{ij}>c)}.$$

We can use this as the basis for constructing the likelihood as we did in Chapter 6, which involves the number of individuals not encountered, n_0. The probability that an individual is *not* captured is equal to the probability that its signal strength doesn't exceed c at any microphone. The probability of not being captured at a microphone $\mathbf{x}_j$ is:

$$1 - p_{\mathbf{u},j} = \Phi(\gamma(\boldsymbol{\theta}, \mathbf{x}, \mathbf{u})).$$

Therefore, the probability of not being captured at any microphone is:

$$\Pr(\text{all } S_{\mathbf{u},j} < c|\mathbf{u}) = \prod_{j=1}^{J}(1 - p_{\mathbf{u},j}) = \prod_{j=1}^{J} \Phi(\gamma(\boldsymbol{\theta}, \mathbf{x}_j, \mathbf{u})),$$

and the marginal probability of not being captured is

$$\pi_0 = [\text{all } S_{\mathbf{u},j} < c|\boldsymbol{\alpha}] = \int_{\mathcal{U}} \left\{ \prod_{j=1}^{J} \Phi(\gamma(\boldsymbol{\theta}, \mathbf{x}_j, \mathbf{u})) \right\} d\mathbf{u},$$

which is used to construct the likelihood as we did in Chapter 6 (see Eq. 6.2.1).

9.4.2 Implementation in `secr`

Fitting acoustic encounter models in `secr` is no more difficult than other SCR models. There is a handy manual (`secr-sound.pdf`) with examples (Efford and Dawson, 2010) that come with the `secr` package. The basic process is that `make.capthist` will make a `capthist` object from a three-dimensional encounter array—which is a binary array indicating whether each individual was detected or not at each recorder/microphone. In the case of signal strength data, `secr` handles the case where # occasions = 1, i.e., the recorders obtained data for a single sample occasion, but this is not a general requirement of the model for signal strength data (see next section). The "signal" attribute of the `capthist` object contains the signal strength in decibels. The best way to include the signal attribute is to use `make.capthist` in the usual way, providing it with the encounter data and trap data and, in addition, the variable "cutval" (which is c in our notation above) and then provide the signal strength data as an extra column of the `capthist` object. See `?make.capthist` for details.

9.4.3 Implementation in BUGS

We don't know of any Bayesian applications of acoustic SCR models, although we imagine that implementation of such models in the **BUGS** engines should be achievable. Instead of developing a Bayesian version of this model here, we leave it to the reader to explore simulating data and devising a Bayesian implementation of the acoustic model in one of the **BUGS** engines. Note that for a single occasion, you can simulate the data using the two-stage model (having both **s** and **u**) or you can simulate **u** uniformly without dealing with **s** in the model. The kernel of the **BUGS** model specification should resemble the following snippet:

```
model {
  # Ignoring loops and data augmentation
  u[i,1] ~ dunif(xlim[1], xlim[2])
  u[i,2] ~ dunif(ylim[1], ylim[2])
  mu[i,j] <- alpha0 + alpha1*d[i,j]
  ###
  ### JAGS has this T() truncation feature
  S[i,j] ~ dnorm(mu[i,j], 1/sigma^2)T(c,Inf)
  ###
  gamma[i,j] <- (c - mu[i,j])/sigma
  p[i,j] <- 1 - pnorm(gamma[i,j], 0, 1) # JAGS has pnorm() function
  y[i,j] ~ dbern(p[i,j])
}
```

9.4.4 Other types of acoustic data

Efford and Dawson (2010) noted that various other types of acoustic data might arise for which SCR-like models would be useful.[4] For example, we could measure the *time of arrival* of a vocal cue of some sort at multiple recorders to estimate the number and origin of N queues. Another example is that where we measure *direction* to a queue from multiple devices and do, effectively, a type of statistical triangulation to the multiple but unknown number of sources. This has direct relevance to types of double or multiple-observer sampling that are used in field studies of birds. Normally, two observers stand in close proximity and record birds, reconciling their detections after data collection. An SCR-based formulation of the double-observer method has two observers (or more) standing some distance apart, e.g., 50 or 100 m, and marking individual birds on a map (or at least a direction) and recording a time of detection. The SCR/double-observer method could be applied to such data.

9.5 Summary and outlook

In this chapter we extended SCR models to accommodate alternative models for the observation process, including Poisson and multinomial models. Along with the binomial model described in Chapter 5, this sequence of models will accommodate a substantial majority of contemporary spatial capture-recapture problems, including the four main types of encounter data: binary encounters, multinomial trials from "multi-catch" and "single-catch" (Efford, 2004, 2011a; Royle and Gardner, 2011) trap

[4] Some of the following is also related to material presented by D.L. Borchers at the ISEC 2012 conference in Oslo, Norway.

Table 9.4 Different observation models, where we discuss them in this book, and the corresponding secr terminology.

observation Model	Where in This Book?	secr Name
Bernoulli	Chapter 5	proximity
Poisson	Section 9.1	count
Multinomial (ind)	Section 9.2	multi-catch
Multinomial (dep)	Section 9.3	single-catch
Acoustic	Section 9.4	signal
Search-encounter	Chapter 15	polygon (in part)

systems, and Poisson encounter frequency data from devices that can record multiple encounters per occasion of the same individual at a device. We summarize the standard observation models and the corresponding secr terminology in Table 9.4. What we refer to as search-encounter (or area-search) models (see Chapter 15) are distinct from most of the other classes in that the observation location can also be random (in contrast to traps, where the location is fixed by design). This auxiliary data is informative about an intermediate process related to movement (Royle and Young, 2008).

There is a need for other types of encounter models that arise in practice. We identify a few of them here, although we neglect a detailed development of them at the present time or, in some cases, put that off until later chapters: (1) Removal systems—sometimes traps kill individuals and SCR models can handle that. This can be viewed as a kind of open model, with mortality only, and we handle such models (in part) in Chapter 16; (2) There are models for which only specific summary statistics are observable (Chandler and Royle, 2013; Sollmann et al., 2013a) which we cover in Chapters 18 and 19; (3) We can have multiple observation methods working together as in Gopalaswamy et al. (2012b).

There remains much research to be done to formalize models for certain observation systems. For example, while we think one will usually be able to analyze single-catch systems using the multi-catch model, or even the Bernoulli model if encounter probability is sufficiently low, a formalization of the single-catch model would be a useful development and, we believe, it should be achievable using one or another of the **BUGS** engines. In addition, classical "trapping webs" (Anderson et al., 1983; Wilson and Anderson, 1985a; Jett and Nichols, 1987; Parmenter and MacMahon, 1989; Link and Barker, 1994) have been around for quite some time and it seems like they are amenable to formulation as a type of SCR model although we have not pursued that development simply because trapping webs are rarely used in practice.

This page is intentionally left blank

Non Print Items

Abstract: In previous chapters we considered various models of encounter probability, both in terms of parametric functions of distance and types of covariate models. However, we have so far only considered a specific probability model for the observations (we'll call this the "observation model")—the Bernoulli encounter process model, which in secr is the proximity detector model. This assumes that individual and trap-specific encounters are independent Bernoulli trials. In this chapter, we focus on developing additional observation models. The observation model could be thought of as being determined by the type of device—or, in secr terminology, the type of "detector." We consider models that apply when observations are not binary and, in some cases, that do not require independence of the observations. For example, if sampling devices can detect an individual some arbitrary number of times during an interval, then it is natural to consider observation models for encounter frequencies, such as the Poisson model. Another type of encounter device is the "multi-catch" device, which is a physical device that can capture and hold an arbitrary number of individuals. A typical example is a mist-net for birds. It is natural to regard observations from these kinds of studies as independent multinomial observations. A related group of devices that produces dependent multinomial observations are the so-called single-catch traps. The canonical examples are small-mammal live traps which catch and hold a single individual. Competition among individuals for traps induces a complex dependence structure among individual encounters. To date, no formal inference framework has been devised for this method although the independent multinomial model can be a good approximation. The final sampling method covered in this chapter involves acoustic recording devices, and we explain how to analyze the resulting data. Throughout, several worked examples are presented using JAGS and secr.

Keywords: Acoustic sampling, Encounter probability, Detection function, Observation model, Poisson model, Multinomial model, Proximity detector, Count detector, Multi-catch detector, Single-catch detector