Bayesian Inference in Camera Trapping Studies for a Class of Spatial Capture-Recapture Models

J. Andrew Royle, U.S. Geological Survey, Patuxent Wildlife Research Center, Laurel, Maryland, 20708, *email:* aroyle@usgs.gov

K. Ullas Karanth, Wildlife Conservation Society-India Program, Centre for Wildlife Studies, Bangalore, Karnataka-560042, India, *email:* ukaranth@wcs.org

Arjun M. Gopalaswamy, Wildlife Conservation Society-India Program, Centre for Wildlife Studies, Bangalore, Karnataka-560042, India, *email:* agopalaswamy@wcs.org

N. Samba Kumar, Wildlife Conservation Society-India Program, Centre for Wildlife Studies, Bangalore, Karnataka-560042, India, *email:* samba.wcs@gmail.com

Appendix A: Analysis of the Poisson encounter model

For the Poisson model where N and the activity centers are known, the observation model is a simple Poisson regression and a Bayesian analysis of it can be carried out in WinBUGS (Gilks, et al. 1994) without difficulty. The WinBUGS model specification is shown in Fig. A1. In adopting a Bayesian analysis of the model, we require prior distributions for parameters. As shown in Fig. A1, we assumed conventional priors meant to reflect the absence of prior information: For σ, a uniform prior on $[0, 5]$ and, for λ_0, a gamma prior with scale parameters 0.1 and 0.1. We see that this only requires a few lines of WinBUGS model description, and most of that is computing the distances between traps and activity centers. The activity centers are `sx` and `sy` and they are input to WinBUGS as data along with `trap`, which are the coordinates of the trap, `N`, `ntrap`, and `K` which are the number of individuals, number of traps and number of samples. Finally, the dependent variable is the 3-dimensional array `y`. Thinking about the model *conditional* on the latent variables reveals the essential simplicity of the model as a generalized linear model.

Model Extensions and Reductions by Sufficiency

The model specification in Fig. A1 is more general than may be required in many situations because it specifies the observation model in terms of the finest scale observation – encounter frequencies of every individual, in every trap, and for all K occasions. When

```
model {
  sigma~dunif(0,5)
  lam0~dgamma(.1,.1)
  for(i in 1:N){
    for(j in 1:ntrap){
       dist2[i,j]<- ( pow(sx[i]-trap[j,1],2) + pow(sy[i]-trap[j,2],2) )
       exposure0[i,j]<- lam0*exp(-dist2[i,j]/sigma)
       for(k in 1:K){
         y[i,k,j] ~ dpois(exposure0[i,j])
       }
  }
 }
 }
```

Fig. A1. **WinBUGS** model specification for the spatial capture-recapture model when $\mathbf{s}_i$ are known for $i = 1, 2, \ldots, N$. Thus, here, `nind` $\equiv N$. The activity centers, $\mathbf{s}$, are input to WinBUGS as data in the form of `sx` and `sy`, vectors of the x- and y-coordinates of all N activity centers.

specified at the level of the most basic observation, the model allows for many extensions. For example, we could add time effects or individual effects to a model for λ_0 by adding only a couple lines to the code in Fig. A1.

When there are no time effects, we can improve the efficiency of the analysis by recognizing that the total number of captures of each individual in trap j is a Poisson random variable with mean $K \times \lambda_0 g_{ij}$. This specification would still allow for the fitting of individual effects. Due to additivity of Poisson random variables, other simplifications are possible in some cases. We do not provide implementations of the possible simplifications and extensions as there are no additional technical considerations.

Unknown s but known N

We consider the incremental extension where we suppose that N is known, but we do not know the point locations $\mathbf{s}$. In this case, the resulting model is analogous to a conventional generalized linear mixed model. To implement this in WinBUGS it is helpful to describe $\mathcal{S}$ by a regular polygon and that way the uniform assumption on $\mathbf{s}$ can be described with the expressions `sx[i]~dunif(Xl,Xu)` and `sy[i]~dunif(Yl,Yu)` as illustrated in Fig. A2. The upper and lower limits (Xl, Xu, Yl, Yu) are input into WinBUGS as data. We see the implementation of this model for fixed N is straightforward.

In order to compute summaries of density over subsets of the point process state space we simply tally-up the individual coordinates $\mathbf{s}$ that are located within a prescribed polygon. WinBUGS properly simulates those summary statistics from the required posterior distri-

```
model {
   sigma~dunif(0,5)
   lam0~dgamma(.1,.1)

   for(i in 1:N){
      sx[i]~dunif(Xl,Xu)
      sy[i]~dunif(Yl,Yu)
      for(j in 1:ntrap){
         dist2[i,j]<- ( pow(sx[i]-trap[j,1],2) + pow(sy[i]-trap[j,2],2) )
         exposure0[i,j]<- lam0*exp(-dist2[i,j]/sigma)
         for(k in 1:K){
            y[i,k,j] ~ dpois(exposure0[i,j])
         }
      }
    }
}
```

Fig. A2. **WinBUGS** model specification for the spatial capture-recapture model with Poisson trap encounter frequencies when $\mathbf{s}_i$ are *unknown* but N (the number of individuals) is known and must be supplied to WinBUGS as data.

bution. If the subset of the state-space is a rectangle with boundaries `xmin`, `xmax`, `ymin` and `ymax`, doing this requires only a few more lines of WinBUGS model specification. We have to add the following lines to the model specification:

```
tmp1[i]<- step(sx[i] - xmin)
tmp2[i]<- step( xmax - sx[i])
tmp3[i]<- step(sy[i] - ymin)
tmp4[i]<- step(ymax - sy[i])
centerin[i]<-tmp1[i]*tmp2[i]*tmp3[i]*tmp4[i]
```

and then, outside of all of the loops, add these lines:

```
Nin<-sum(centerin[1:nind])
D<-Nin/areaSsub
```

where `areaSsub` is the area of the subset, input as data. This is implemented in the WinBUGS specification shown in Fig. A3 (where `areaSsub` is specified to be 81 in the model specification). To improve efficiency of the analysis in WinBUGS, this version of the model is described in terms of the $n \times J$ matrix of capture frequencies obtained by summing the K frequencies for each individual as described in *Model Extensions and Reductions by Sufficiency*

```
model {

  sigma~dunif(0,5)
  lam0~dgamma(.1,.1)

  for(i in 1:N){

    sx[i]~dunif(Xl,Xu)
    sy[i]~dunif(Yl,Yu)

    tmp1[i]<- step(sx[i] - xmin)
    tmp2[i]<- step( xmax - sx[i])
    tmp3[i]<- step(sy[i] - ymin)
    tmp4[i]<- step(ymax - sy[i])
    centerin[i]<-tmp1[i]*tmp2[i]*tmp3[i]*tmp4[i]

    for(j in 1:ntrap){
        dist2[i,j]<- ( pow(sx[i]-trap[j,1],2)  +  pow(sy[i]-trap[j,2],2) )
        exposure0[i,j]<- lam0*exp(-dist2[i,j]/sigma)
        log(pmean[i,j])<-log(K)+ log(exposure0[i,j])
        y[i,j] ~ dpois(pmean[i,j])
    }
   }
  Nin<-sum(centerin[1:nind])
  D<-Nin/81
}
```

Fig. A3. WinBUGS model specification for the Poisson version of the spatial capture-recapture model when $\mathbf{s}_i$ are *unknown* but N (the number of individuals) is known. This specification also computes the derived parameters $N(\mathcal{X})$ (population size) and $D(\mathcal{X})$ (density) for $\mathcal{X}$ a subset of the point process state space defined in this case as the minimum area rectangle enclosing the traps. This version of the model is based on pooled encounter frequencies over all K surveys.

Unknown N

The incremental (but important) extension of the model requires only a little bit of additional WinBUGS model specification. In particular, we define latent indicator variables w_i associated with the data augmentation which we assume to be Bern(ψ) random variables. Then, we note that the counts for the *augmented* data are zero-inflated Poisson (Model 1) or Binomial (Model 2) counts. To implement this, redefine the parameter of the (Poisson or Binomial) distribution to be the product of the indicator `w[i]` and the parameter of the "known-N" model. For example, expanding on the Poisson model from Fig. A3, the relevant specification to zero-inflate the Poisson distribution is:

```
w[i] ~ dbern(psi)

log(pmean[i,j])<-log(K)+ log(exposure0[i,j])
tmp[i,j]<- pmean[i,j]*w[i]
y[i,j] ~ dpois(tmp[i,j])
```

This construction of the zero-inflation process means that, if `w[i] = 1` then the observations are Poisson with mean `pmean[i,j]` whereas, if `w[i] = 0` then the observations are *fixed zeroes*.

This can be seen in context in Fig. A4.

```
model {

  sigma~dunif(0,5)
  lam0~dgamma(.1,.1)
  psi ~ dunif(0,1)

  for(i in 1:M){

   w[i]~dbern(psi)
   sx[i]~dunif(Xl,Xu)
   sy[i]~dunif(Yl,Yu)

   for(j in 1:ntrap){
      dist2[i,j]<- ( pow(sx[i]-trap[j,1],2)  +  pow(sy[i]-trap[j,2],2) )
      exposure0[i,j]<- lam0*exp(-dist2[i,j]/sigma)
      log(pmean[i,j])<-log(K)+ log(exposure0[i,j])
      tmp[i,j]<-pmean[i,j]*w[i]
      y[i,j] ~ dpois(tmp[i,j])
   }
}
  N<- sum(w[1:M])
  D<-N/areaS
}
```

Fig. A4. WinBUGS model specification for the Poisson encounter process model. In this specification, the activity centers $\mathbf{s}$ and N are unknown.