

# Capture-Recapture with Bayesian statistics

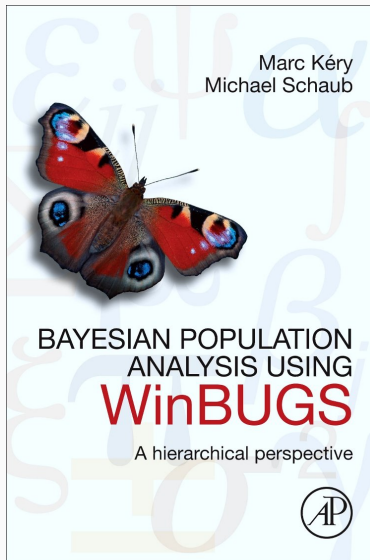
---

Lorelei Guéry

July 2020

- All material prepared with R.
- R Markdown used to write reproducible material.
- Material available via Github [here](#).

- Workshops material shared by Andy Royle and the Biometrics Working Group [there](#) or [there](#)
- Materials shared by [Olivier Gimenez](#), [Murray Efford](#) and [Andy Royle](#) [here](#) and [there](#)



## Spatial Capture-Recapture

J. Andrew Royle • Richard B. Chandler • Rahel Sollmann • Beth Gardner

*Spatial Capture-Recapture* provides a revolutionary extension of traditional capture-recapture methods for studying animal populations using data from live trapping, camera trapping, DNA sampling, acoustic sampling, and related field methods. This book is a conceptual and methodological synthesis of spatial capture-recapture modeling. As a comprehensive how-to manual, this reference contains detailed examples of a wide range of relevant spatial capture-recapture models for inference about population size and spatial and temporal variation in demographic parameters. Practicing field biologists studying animal populations will find this book to be a useful resource, as will graduate students and professionals in ecology, conservation biology, and fisheries and wildlife management.

### Key features:

- Offers comprehensive coverage of revolutionary new methods in ecology
- Includes detailed worked examples with R and BUGS code for each methodological element along with software instructions and a companion R package so you can implement analyses and learn by doing
- Presents a practical approach, embracing Bayesian and classical inference strategies in order to provide a variety of options to best get the job done



elsevier.com

Front cover: Jaguar "captured" in a camera trap near Iguazú National Park, Argentina (Credit: Agustin Posadas)  
Back cover: Top: Map showing marine camera trap locations in SE Alaska  
Bottom: Camera trap photograph of radio-collared coyote on Fort Briggs, NC (Credit: North Carolina State University)



## Spatial Capture-Recapture

Royle  
Chandler  
Sollmann  
Gardner



## Spatial Capture-Recapture

J. Andrew Royle  
Richard B. Chandler  
Rahel Sollmann  
Beth Gardner



1. Bayesian Capture-Recapture models in closed population
  - Example 1: Fit Model M0 to the bear data using JAGS and data augmentation
2. Spatially Explicit Capture Recapture (SECR) models
  - Example 2: Fit a basic SECR model in closed population with data augmentation

Many different packages can be used to run JAGS from R such as:

- `rjags`
- `jagsUI`
- `R2jags`

Different packages can be used to run SECR models from R such as:

- `scrbook` from the *Spatial Capture-Recapture* book by Andy Royle, Richard Chandler, Rahel Sollmann and Beth Gardner



Different packages can be used to run SECR models from R such as:

- `scrbook` from the *Spatial Capture-Recapture* book by Andy Royle, Richard Chandler, Rahel Sollmann and Beth Gardner
- `secr` developed by Murray Efford

Different packages can be used to run SECR models from R such as:

- `scrbook` from the *Spatial Capture-Recapture* book by Andy Royle, Richard Chandler, Rahel Sollmann and Beth Gardner
- `secr` developed by Murray Efford
- `oSCR` developed by Chris Sutherland, Andy Royle, and Dan Linden

## Bayesian Capture-Recapture models in closed population

---

## Capture-recapture, quésako?

- Information on  $N$  or density  $D$  is the main interest.

## Capture-recapture, quésako?

- Information on  $N$  or density  $D$  is the main interest.
- Only a sample of individuals  $n$  is observed due to an encounter or detection probability  $p$ .

## Capture-recapture, qué sako?

- Information on  $N$  or density  $D$  is the main interest.
- Only a sample of individuals  $n$  is observed due to an encounter or detection probability  $p$ .
- To estimate or model  $p$ , studies to generate encounter history information are conducted.

## Capture-recapture, quésako?

- Information on  $N$  or density  $D$  is the main interest.
- Only a sample of individuals  $n$  is observed due to an encounter or detection probability  $p$ .
- To estimate or model  $p$ , studies to generate encounter history information are conducted.
- The statistical models to describe these encounter histories are capture-recapture (CR) models.

## Individual encounter probability

	Occasion				
individual	1	2	3	4	5
-----					
1	1	0	1	0	1
2	0	1	0	0	0
3	0	1	1	1	0
4	0	0	1	0	1
5	0	1	0	0	0
...	• •	• •		• •	
...					



## Starting point of CR

- Random sampling of individuals: detection is a Bernoulli trial (binomial distribution)

## Starting point of CR

- Random sampling of individuals: detection is a Bernoulli trial (binomial distribution)
- CR models are, one way or another, logistic regression models or GLMs where  $N$  is unknown.

## Starting point of CR

- Random sampling of individuals: detection is a Bernoulli trial (binomial distribution)
- CR models are, one way or another, logistic regression models or GLMs where  $N$  is unknown.
- Status of individuals is not known. You don't observe "all zero encounter histories".

## Starting point of CR

- Random sampling of individuals: detection is a Bernoulli trial (binomial distribution)
- CR models are, one way or another, logistic regression models or GLMs where  $N$  is unknown.
- Status of individuals is not known. You don't observe "all zero encounter histories".
- Initial CR models developed for geographically closed populations.

## Starting point of CR

- Random sampling of individuals: detection is a Bernoulli trial (binomial distribution)
- CR models are, one way or another, logistic regression models or GLMs where  $N$  is unknown.
- Status of individuals is not known. You don't observe "all zero encounter histories".
- Initial CR models developed for geographically closed populations.
- Heterogeneity in  $p$  is important (bias in  $N$ ) and CR models are all about modeling variation in  $p$  (Otis et al. 1978)

## Closed population

- Demographic closure (no births, no deaths) and geographical closure (no entry, no exit)
- Closed models characterization (Otis *et al.* 1978):
  - $M_0$  = “the null model”,  $p$  is constant in all dimensions
  - $M_t = p$  is a function of sample occasion,  $p(t)$
  - $M_b$  = behavioral response model. Trap happiness or shyness
  - $M_h$  = individual heterogeneity
  - $M_{bt}$  = time + behavior, or time\*behavior
  - $M_{bh}$ ,  $M_{th}$ ,  $M_{bth}$

## Closed population

- Demographic closure (no births, no deaths) and geographical closure (no entry, no exit)
- Closed models characterization (Otis *et al.* 1978):
  - $M_0$  = “the null model”,  $p$  is constant in all dimensions
  - $M_t = p$  is a function of sample occasion,  $p(t)$
  - $M_b$  = behavioral response model. Trap happiness or shyness
  - $M_h$  = individual heterogeneity
  - $M_{bt}$  = time + behavior, or time\*behavior
  - $M_{bh}$ ,  $M_{th}$ ,  $M_{bth}$
- See [Kery and Schaub \(2012\)](#) Chapter 6 to go further

## Basic model M0

- Model M0 can be considered as a null model
- The main assumptions are:
  - $p$  is constant for all sample occasions and all individuals
  - Encounters are independent among and within individuals



## Basic model M0

- Model M0 can be considered as a null model
- The main assumptions are:
  - $p$  is constant for all sample occasions and all individuals
  - Encounters are independent among and within individuals
- Encounter observations are Bernoulli random variables

## Basic model M0

- Model M0 can be considered as a null model
- The main assumptions are:
  - $p$  is constant for all sample occasions and all individuals
  - Encounters are independent among and within individuals
- Encounter observations are Bernoulli random variables
- Close to a binomial GLM or logistic regression but where  $N$ , size of some ideal data set, is unknown

## N unknown, so what?

- If N is known, Model M0 is a logistic regression.

```
model{  
  p ~ dunif(0,1)  
  for (i in 1:N){  
    y[i] ~ dbin(p,K)  
  }  
}
```

# N unknown, so what?

- If N is known, Model M0 is a logistic regression.

```
model{  
  p ~ dunif(0,1)  
  for (i in 1:N){  
    y[i] ~ dbin(p,K)  
  }  
}
```

- But N is not known. Why couldn't we put a prior on N (e.g.  $N \sim \text{dunif}(0, 1000)$ ) and analyze the model using standard methods of MCMC?

## N unknown, so what?

- If N is known, Model M0 is a logistic regression.

```
model{  
  p ~ dunif(0,1)  
  for (i in 1:N){  
    y[i] ~ dbin(p,K)  
  }  
}
```

- But N is not known. Why couldn't we put a prior on N (e.g.  $N \sim \text{dunif}(0, 1000)$ ) and analyze the model using standard methods of MCMC?
- Because N would be a parameter of the model and would be updated in the MCMC algorithm. The size of the data set would have to change, which is not possible with JAGS

- *Concept underlying DA is adding “observations” to create a dataset composed of a known number of individuals.*

## Data augmentation (DA), see [Royle and Dorazio paper](#)

- *Concept underlying DA is adding “observations” to create a dataset composed of a known number of individuals.*
- *For CR models, addition of a set of “all zero” encounter histories which are not observable in practice.*

## Data augmentation (DA), see [Royle and Dorazio paper](#)

- *Concept underlying DA is adding “observations” to create a dataset composed of a known number of individuals.*
- *For CR models, addition of a set of “all zero” encounter histories which are not observable in practice.*
- *The model of the augmented dataset is a zero-inflated version of either a binomial or a multinomial base model.*



## Data augmentation (DA), see [Royle and Dorazio paper](#)

- *Concept underlying DA is adding “observations” to create a dataset composed of a known number of individuals.*
- *For CR models, addition of a set of “all zero” encounter histories which are not observable in practice.*
- *The model of the augmented dataset is a zero-inflated version of either a binomial or a multinomial base model.*
- *Their use of DA provides a general approach for analyzing both closed and open population models of all types.*

	Occasion				
individual	1	2	3	4	5
-----					
1	1	0	1	0	1
2	0	1	0	0	0
3	0	1	1	1	0
4	0	0	1	0	1
5	0	1	0	0	0
...	• •	• •		• •	
...					

## **Example 1: Fit Model M0 to the bear data using JAGS and data augmentation**

---

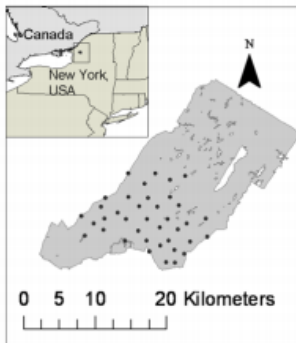
- Install the package `scrbook` [there](#) and get the bear data

```
library(scrbook)
```

```
data(beardata)
```

# Analysis of the Fort Drum bear data

- Hair snare study
  - $J = 38$  hair snares
  - $K = 8$  weeks of sampling
  - $n = 47$  individuals captured



## Step 1: Create a text file with the model description, written in the BUGS language

```
cat("
model {
psi ~ dunif(0, 1) # DA parameter
p ~ dunif(0,1) # prior distribution
for (i in 1:M){
  z[i] ~ dbern(psi) # binary DA latent variables which indicates if individual i is
                    # a member of the population - Abundance is just the sum of
                    # these binary latent variables

  for(k in 1:K){
    tmp[i,k] <- p*z[i]
    y[i,k] ~ dbin(tmp[i,k],1) # likelihood
  }
}
N <- sum(z[1:M])
}
,file="code/modelM0.txt")
```

## Step 2: Store the different values of interest

```
M = 175 # number of all individuals (encountered and DA)
nind <- dim(beardata$bearArray)[1] # number of encounter histories (individuals)
ntraps <- dim(beardata$bearArray)[2] # number of traps
K <- dim(beardata$bearArray)[3] # number of occasions

# How many "all zero" encounter histories are there?
nz <- M-nind

nz
#> [1] 128
```

### Step 3: Set up the data augmentation and create the 2-d matrix “individual x occasions”

```
# Fill up an array with zeros
Yaug <- array(0, dim=c(M,ntraps,K))

# Store the real data into the first nind slots
Yaug[1:nind,,] <- beardata$bearArray

# Because traditional CR models ignore space create a 2-d matrix
# "individuals x occasions" of 0/1 data where 1 = "captured" 0 = "not captured"

y <- apply(Yaug,c(1,3),sum) # summarize by ind * occ
y[y>1] <- 1 # make sure that multiple encounters do not occur
```



## Step 4: Set input and output

- Format your data in R as a named list

```
set.seed(2020)  
data <- list(y=y,M=M,K=K)
```

- Make an object containing the names of the parameters that you are interested in

```
params <- c("psi","p","N")
```

## Step 5: Initial values

- Create a function to generate random initial values

```
zst = c(rep(1,nind),rbinom(M-nind, 1, .5))  
inits = function(){list(z=zst, psi=runif(1), p=runif(1))}
```

## Step 6: Run

- Compile the model and obtain posterior samples

```
# Package rjags
library(rjags)
jm <- jags.model("code/modelM0.txt", data=data, inits=inits, n.chains=3, n.adapt=1000)
fit0j <- coda.samples(jm, params, n.iter=1000)

# Package jagsUI
library(jagsUI)
fit0j = jags(data, inits, params, model.file="code/modelM0.txt", n.chains=3,
             n.iter=2000, n.burnin=1000, n.thin=1)
```

## Results: summary

```
Iterations = 1001:2000  
Thinning interval = 1  
Number of chains = 3  
Sample size per chain = 1000
```

1. Empirical mean and standard deviation for each variable,  
plus standard error of the mean:

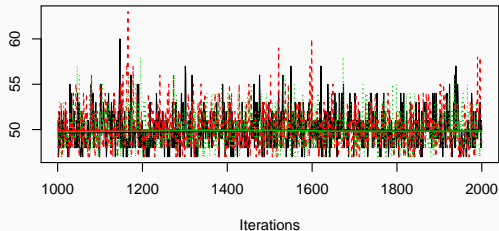
	Mean	SD	Naive SE	Time-series SE
N	50.0347	2.04586	0.0373520	0.0511018
p	0.3017	0.02620	0.0004783	0.0007502
psi	0.1015	0.01425	0.0002602	0.0002971

2. Quantiles for each variable:

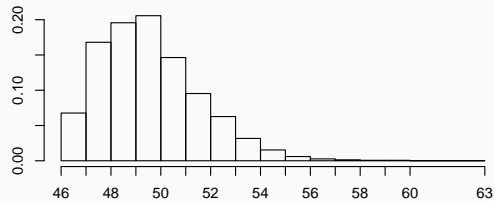
	2.5%	25%	50%	75%	97.5%
N	47.00000	49.00000	50.0000	51.0000	55.0000
p	0.25171	0.28413	0.3014	0.3192	0.3533
psi	0.07499	0.09157	0.1011	0.1105	0.1318

# Results: plot

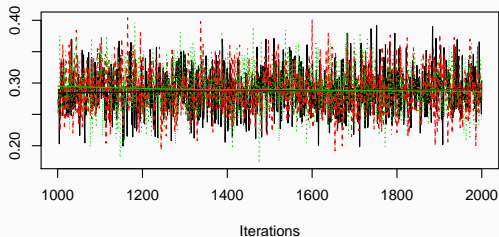
Trace of N



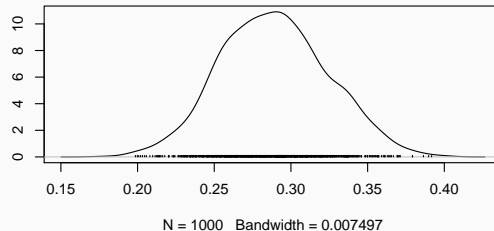
Density of N



Trace of psi



Density of psi



## Your turn

- Try different values of  $M$ : 50 and 400.

- Try different values of  $M$ : 50 and 400.
- Compare estimates (with summary function)

## Your turn

- Try different values of  $M$ : 50 and 400.
- Compare estimates (with summary function)
- Make a plot of the posterior distribution of  $N$  for both of them



## Solution M = 50

```
Iterations = 1001:2000  
Thinning interval = 1  
Number of chains = 3  
Sample size per chain = 1000
```

1. Empirical mean and standard deviation for each variable,  
plus standard error of the mean:

	Mean	SD	Naive SE	Time-series SE
N	48.7943	1.00018	0.0182608	0.0281167
p	0.3075	0.02418	0.0004414	0.0005999
psi	0.9572	0.03389	0.0006188	0.0008712

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
N	47.0000	48.0000	49.0000	50.0000	50.0000
p	0.2614	0.2914	0.3070	0.3232	0.3567
psi	0.8738	0.9391	0.9651	0.9837	0.9986

## Solution M = 400

```
Iterations = 1001:2000  
Thinning interval = 1  
Number of chains = 3  
Sample size per chain = 1000
```

1. Empirical mean and standard deviation for each variable,  
plus standard error of the mean:

	Mean	SD	Naive SE	Time-series SE
N	49.9753	2.02009	0.0368816	0.0463588
p	0.3019	0.02570	0.0004692	0.0007004
psi	0.1268	0.01746	0.0003187	0.0003484

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
N	47.00000	48.0000	50.0000	51.0000	55.0000
p	0.25192	0.2837	0.3017	0.3195	0.3521
psi	0.09523	0.1148	0.1257	0.1378	0.1632

## **Spatially Explicit Capture Recapture (SECR) models**

---

## Some references

- Efford (2004): initial idea, ad hoc simulation based approach

## Some references

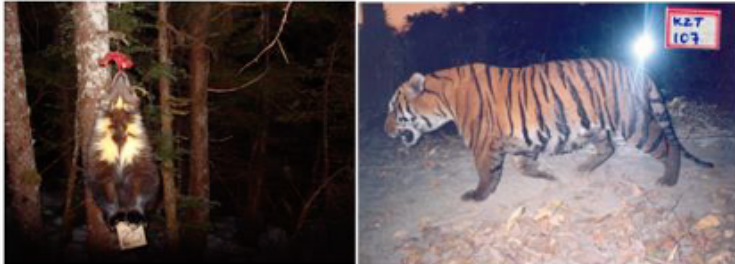
- Efford (2004): initial idea, ad hoc simulation based approach
- Likelihood-based approaches: Borchers and Efford (2008), Efford *et al.* (2008, 2009)

## Some references

- Efford (2004): initial idea, ad hoc simulation based approach
- Likelihood-based approaches: Borchers and Efford (2008), Efford *et al.* (2008, 2009)
- Hierarchical MCMC approach: Royle and Young (2008), Royle *et al.* (2009*ab*), Royle and Gardner (2010), Gardner *et al.* (2009, 2010)

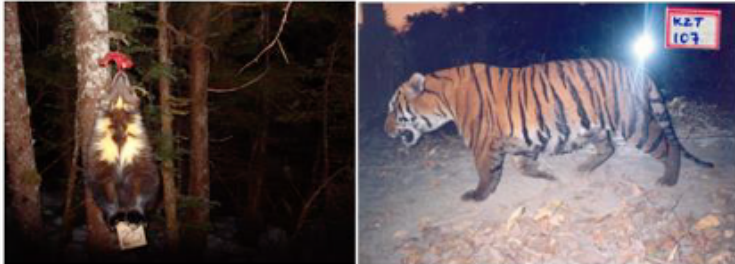
## Why SECR models?

- New technologies producing vast quantities of encounter history data



## Why SECR models?

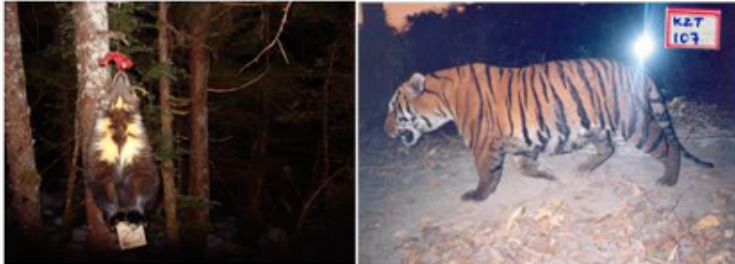
- New technologies producing vast quantities of encounter history data
- Camera traps





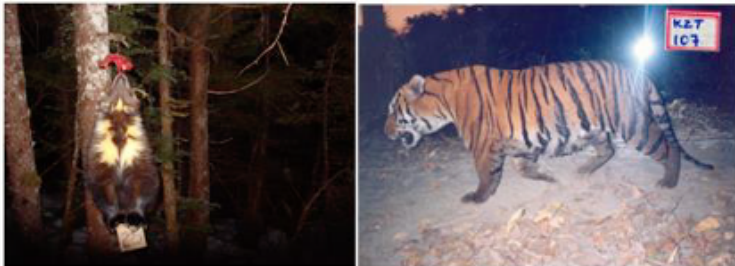
## Why SECR models?

- New technologies producing vast quantities of encounter history data
- Camera traps
- DNA sampling (Scat picked up by searching space, Urine on scent sticks or in snow, Tissue samples from treed individuals, Hair snares)



## Why SECR models?

- New technologies producing vast quantities of encounter history data
- Camera traps
- DNA sampling (Scat picked up by searching space, Urine on scent sticks or in snow, Tissue samples from treed individuals, Hair snares)
- Acoustic sampling (whales, birds, bats)



## Main Goals

- Extension of CR to make use of encounter location data in order to study spatial aspects of animal populations

# Main Goals

- Extension of CR to make use of encounter location data in order to study spatial aspects of animal populations
- Examples: spatial variation in animal density, resource selection or animal movement

# Main Goals

- Extension of CR to make use of encounter location data in order to study spatial aspects of animal populations
- Examples: spatial variation in animal density, resource selection or animal movement
- Models developed to deal explicitly with 2 main problems associated with density estimation in trapping studies

# Main Goals

- Extension of CR to make use of encounter location data in order to study spatial aspects of animal populations
- Examples: spatial variation in animal density, resource selection or animal movement
- Models developed to deal explicitly with 2 main problems associated with density estimation in trapping studies
- 1 -Unknown sample area varies with trap layout – home range size

# Main Goals

- Extension of CR to make use of encounter location data in order to study spatial aspects of animal populations
- Examples: spatial variation in animal density, resource selection or animal movement
- Models developed to deal explicitly with 2 main problems associated with density estimation in trapping studies
  - 1 -Unknown sample area varies with trap layout – home range size
  - 2- Heterogeneity in capture probability associated with animal location relative to traps

- Spatial distribution of organisms is naturally described by point process models

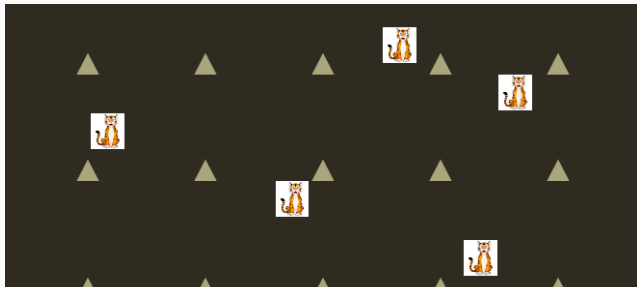


- Spatial distribution of organisms is naturally described by point process models
- In the CR context, Efford (2004, Oikos) described two concepts: the biological or state process and the observation process

## Concept 1: Biological or state process

Describe how individuals are distributed in space = a point process model for “activity centers” or home range centers

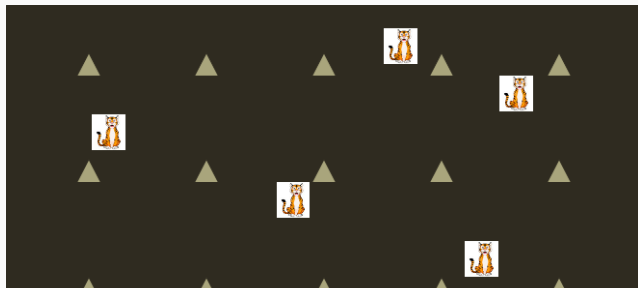
- $s_i$  = coordinates of the activity center or home range center for individual  $i$



## Concept 1: Biological or state process

Describe how individuals are distributed in space = a point process model for “activity centers” or home range centers

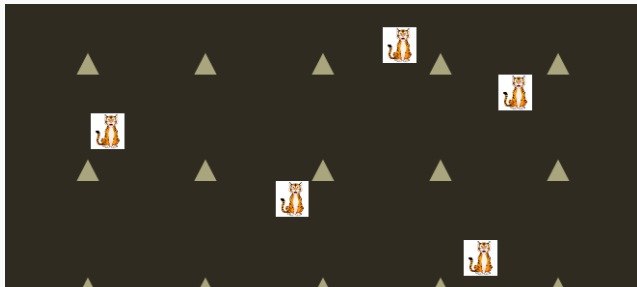
- $s_i$  = coordinates of the activity center or home range center for individual  $i$
- $s_1, s_2, \dots, s_N$  = realization of a point process



## Concept 1: Biological or state process

Describe how individuals are distributed in space = a point process model for “activity centers” or home range centers

- $s_i$  = coordinates of the activity center or home range center for individual  $i$
- $s_1, s_2, \dots, s_N$  = realization of a point process
- $s_i \sim \text{Uniform}(S)$ ;  $S$  = state-space of point process, i.e. the spatial region where the  $N$  activity centers occur



## Concept 2: Observation process

Describe  $P(\text{encounter in trap})$  conditional on where an individual lives ( $s_i$ )

- $y_{i,j}|s_i \sim \text{Bern}(p(x_j, s_i))$

## Concept 2: Observation process

Describe  $P(\text{encounter in trap})$  conditional on where an individual lives ( $s_i$ )

- $y_{i,j}|s_i \sim \text{Bern}(p(x_j, s_i))$
- $x_j = \text{traplocation}$

## Concept 2: Observation process

Describe  $P(\text{encounter in trap})$  conditional on where an individual lives ( $s_i$ )

- $y_{i,j}|s_i \sim \text{Bern}(p(x_j, s_i))$
- $x_j = \text{traplocation}$
- $p(x_j, s_i) = p_0 * \exp(-d_{ij}^2(x_j, s_i)/\sigma^2)$

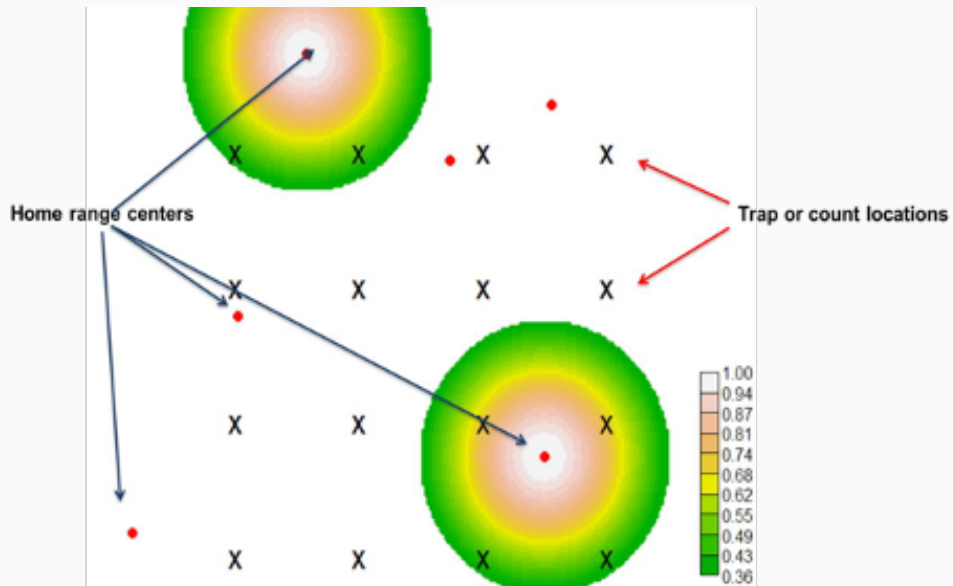
## Concept 2: Observation process

Describe  $P(\text{encounter in trap})$  conditional on where an individual lives ( $s_i$ )

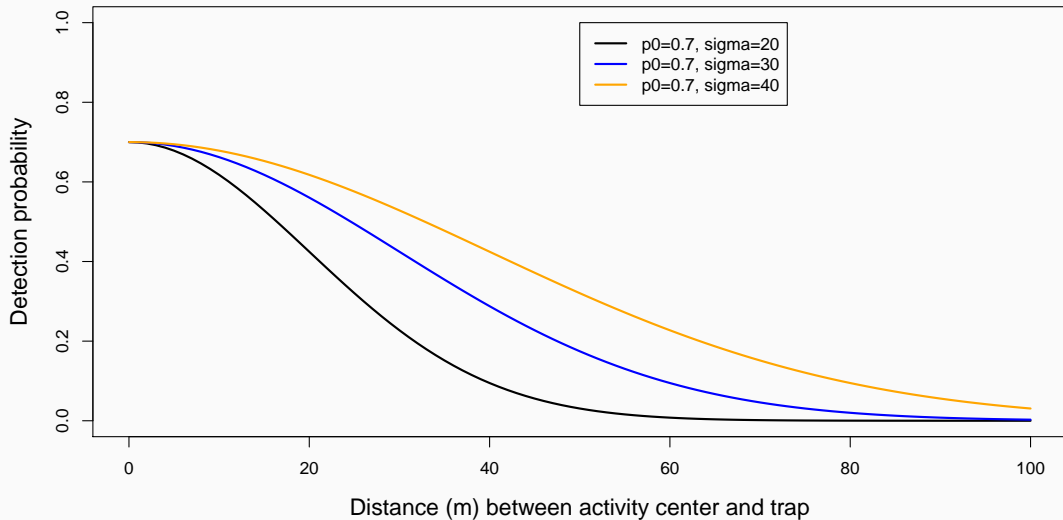
- $y_{i,j}|s_i \sim \text{Bern}(p(x_j, s_i))$
- $x_j = \text{traplocation}$
- $p(x_j, s_i) = p_0 * \exp(-d_{ij}^2(x_j, s_i)/\sigma^2)$
- $d_{ij} = ||s_i \sim x_j|| = \sqrt{(s_{1i} \sim x_{1i})^2 + (s_{2i} \sim x_{2i})^2}$



## Concept 2: Observation process



## Concept 2: Observation process



## Distributions for observation model

- Binomiale (hare snare): The individual can be encountered maximum once in each trap but in different traps in the same occasion

## Distributions for observation model

- Binomiale (hare snare): The individual can be encountered maximum once in each trap but in different traps in the same occasion
- Multinomiale (physical capture): The individual can be encountered maximum once in one trap in each occasion

## Distributions for observation model

- Binomiale (hare snare): The individual can be encountered maximum once in each trap but in different traps in the same occasion
- Multinomiale (physical capture): The individual can be encountered maximum once in one trap in each occasion
- Poisson (camera trap): The individual can be encountered several times by trap in each occasion

- SCR models are GLMMs where  $N$  is unknown

- SCR models are GLMMs where  $N$  is unknown
- MLE based on marginal likelihood: Borchers, D.L. and M.G. Efford. 2008. Spatially explicit maximum likelihood methods for capture-recapture studies. Biometrics 64:377-385 -> R package 'secr' (M.G. Efford)

- SCR models are GLMMs where  $N$  is unknown
- MLE based on marginal likelihood: Borchers, D.L. and M.G. Efford. 2008. Spatially explicit maximum likelihood methods for capture-recapture studies. Biometrics 64:377-385 -> R package 'secr' (M.G. Efford)
- Bayesian Analysis by MCMC (data augmentation): Royle, J.A. and K.Y. Young. 2008. A hierarchical model for spatial capture-recapture data. Ecology 89:2281-2289.



## **Example 2: Fit a basic SECR model in closed population with data augmentation**

---

# Instructions

- Import the encounter history data

```
#>   animalID trapID   occasion  
#> 1      a6     t1 2020-01-01  
#> 2      a6     t2 2020-01-01  
#> 3      a6     t3 2020-01-01  
#> 4      a8     t4 2020-01-01
```

- Import trap coordinates

```
#>      x1  x2  
#> t1 0.3 0.3  
#> t2 0.3 0.4  
#> t3 0.3 0.5  
#> t4 0.3 0.6
```

## Step 0: Create 3D capture history array using the table function

Add missing traps and occasions in the levels to have a complete 3D array and order

```
nocapTraps <- setdiff(rownames(traps), encounters$trapID)
nocapTraps
#> [1] "t5" "t9" "t13" "t14" "t15" "t17" "t19" "t24"
levels(encounters$trapID) <- c(levels(encounters$trapID), nocapTraps)
levels(encounters$trapID) # All trapIDs should be here now
#> [1] "t1" "t10" "t11" "t12" "t16" "t18" "t2" "t20" "t21" "t22" "t23" "t25"
#> [13] "t3" "t4" "t6" "t7" "t8" "t5" "t9" "t13" "t14" "t15" "t17" "t19"
#> [25] "t24"

y3D <- table(encounters$animalID, encounters$trapID,
             encounters$occasion)
```

## Step 0: Create 3D capture history array using the table function

Add missing traps and occasions in the levels to have a complete 3D array and order

```
y3D[1:4,1:10,1] ## Data on first 4 ind at first 10 trap on k=1
```

```
#>
```

```
#>      t1 t10 t11 t12 t16 t18 t2 t20 t21 t22
```

```
#> a1    0  0  0  0  0  0  0  0  0  0
```

```
#> a10   0  0  1  0  0  0  0  0  0  0
```

```
#> a11   0  0  0  0  0  0  0  0  0  0
```

```
#> a2    0  0  0  0  0  0  0  0  0  0
```

```
all(rownames(traps)==colnames(y3D)) ## Not good
```

```
#> [1] FALSE
```

```
y3D <- y3D[,rownames(traps),] ## Re-order
```

```
all(rownames(traps)==colnames(y3D)) ## Good
```

```
#> [1] TRUE
```

## Step 1: Create a text file with the model description

```
cat("
model {
  p0 ~ dunif(0, 1) # baseline encounter probability
  sigma ~ dunif(0, 2) # scale parameter of encounter function
  psi ~ dunif(0, 1) # DA parameter:  $E(N) = M \cdot \psi$ 

  for(i in 1:M) {
    z[i] ~ dbern(psi) # Is individual real?
    s[i,1] ~ dunif(xlim[1], xlim[2]) # x-coordinate of activity center
    s[i,2] ~ dunif(ylim[1], ylim[2]) # y-coordinate

    for(j in 1:J) {
      # dist between activity center and trap
      d[i,j] <- sqrt((s[i,1] - x[j,1])^2 + (s[i,2] - x[j,2])^2)
      p[i,j] <- p0*exp(-d[i,j]^2/(2*sigma^2)) # capture prob at trap j
    }
  }
}
```

## Step 1: Create a text file with the model description

```
cat("
...

for(k in 1:K) {
y[i,j,k] ~ dbern(p[i,j]*z[i]) # model for data
}
}
}

N <- sum(z) # realized abundance
EN <- M*psi # expected abundance
A <- (xlim[2]-xlim[1])*(ylim[2]-ylim[1]) # area of state-space
D <- N/A # realized density
ED <- EN/A # expected density
}

",file="code/SECR0.txt")
```

## Step 2: Store the different values of interest

```
M <- 50 # number of all individuals (encountered and DA)
J <- dim(y3D)[2] # number of traps
K <- dim(y3D)[3] # number of occasions
n0 <- nrow(y3D) # number of encounter histories, i.e. encountered individuals
```

## Step 3: Set up the data augmentation

```
# Fill up an array with zeros
```

```
yz <- array(0, c(M, J, K))
```

```
# Store the real data into the first nind slots
```

```
yz[1:n0,,] <- y3D
```



## Step 4: Set input and output

- Format your data in R as a named list

```
set.seed(2020)  
jd <- list(y=yz, J=J, K=K, M=M, x=traps, xlim=c(0,1), ylim=c(0,1))
```

- Make an object containing the names of the parameters that you are interested in

```
jp <- c("N", "p0", "sigma")
```

## Step 5: Initial values

- Create a function to generate random initial values

```
ji <- function() list(z=rep(1,M), p0=runif(1), sigma=runif(1))
```

## Step 6: Run

- Compile the model and obtain posterior samples

```
# Package rjags
library(rjags)
jm <- jags.model("code/SECR0.txt", jd, ji, n.chains=1, n.adapt=1000)
#> Compiling model graph
#>   Resolving undeclared variables
#>   Allocating nodes
#> Graph information:
#>   Observed stochastic nodes: 5000
#>   Unobserved stochastic nodes: 153
#>   Total graph size: 32475
#>
#> Initializing model
jc <- coda.samples(jm, jp, 1000)
```

## Results: summary

```
Iterations = 1001:2000
Thinning interval = 1
Number of chains = 1
Sample size per chain = 1000
```

1. Empirical mean and standard deviation for each variable,  
plus standard error of the mean:

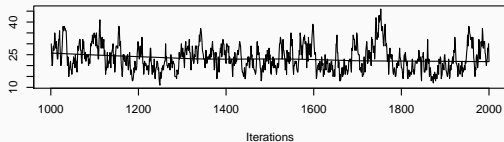
	Mean	SD	Naive SE	Time-series SE
N	24.18200	6.05091	0.1913465	0.663743
p0	0.43581	0.10340	0.0032697	0.007888
sigma	0.08308	0.01112	0.0003516	0.001256

2. Quantiles for each variable:

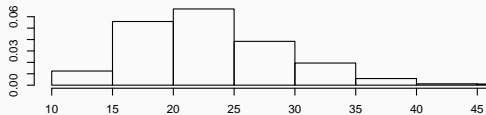
	2.5%	25%	50%	75%	97.5%
N	15.00000	20.00000	23.00000	28.00000	39.00000
p0	0.25688	0.36190	0.42724	0.49724	0.6695
sigma	0.06489	0.07491	0.08207	0.08925	0.1084

# Results: plot

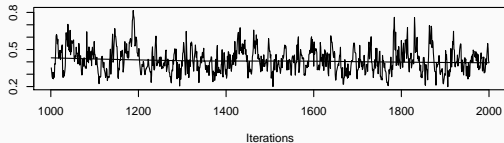
Trace of N



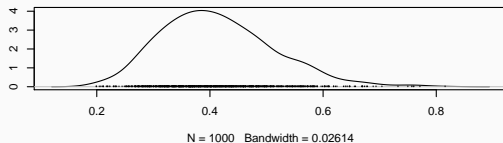
Density of N



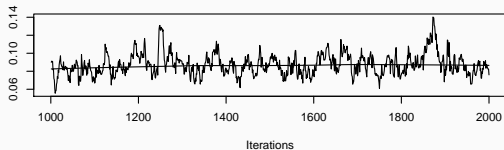
Trace of p0



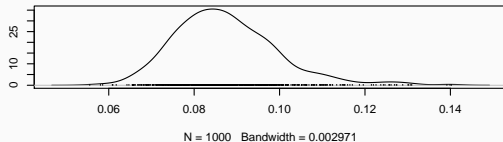
Density of p0



Trace of sigma



Density of sigma



**Your turn**

## More complex model

- Open population ouverte (survival, recruitment. . .)

```
knitr::purl('BayesianCR_LGuery.Rmd')
```

## More complex model

- Open population ouverte (survival, recruitment. . .)
- Time and/or sex effect

```
knitr::purl('BayesianCR_LGuery.Rmd')
```



## More complex model

- Open population ouverte (survival, recruitment. . .)
- Time and/or sex effect
- Variation in effort, e.g. duration of trap activity

```
knitr::purl('BayesianCR_LGuery.Rmd')
```

## More complex model

- Open population ouverte (survival, recruitment. . .)
- Time and/or sex effect
- Variation in effort, e.g. duration of trap activity
- Habitat mask

```
knitr::purl('BayesianCR_LGuery.Rmd')
```

## More complex model

- Open population ouverte (survival, recruitment. . .)
- Time and/or sex effect
- Variation in effort, e.g. duration of trap activity
- Habitat mask
- Mobile activity centers

```
knitr::purl('BayesianCR_LGuery.Rmd')
```

## More complex model

- Open population ouverte (survival, recruitment. . .)
- Time and/or sex effect
- Variation in effort, e.g. duration of trap activity
- Habitat mask
- Mobile activity centers
- Mixing count data with SECR data

```
knitr::purl('BayesianCR_LGuery.Rmd')
```