Multi session analysis 2019

Fernando Mateos-Gonzalez (ALKA Wildlife)

This report analises data from capture-recapture sessions during 17-18-19 April, 24-25 April and 15-19, 24 July 2019, in order to calculate the density of the population.

library(secr)

## This is secr 3.2.1. For overview type ?secr

library(here)

## here() starts at D:/FPI/2019/CZ/SOUSLIK/susliks\_july\_2019

## Analysis

This analysis was carried out using the package ‘secr’ version 3.2.1 and R 3.6.1

We start by creating the capthist, the file combining our captures with the trap locations:

multi <- read.capthist(here("data", "multicaptures.txt"), here("data", "traps.txt"), detector = "proximity")

## Session 2   
## More than one detection per detector per occasion at binary detector(s)

summary(multi)

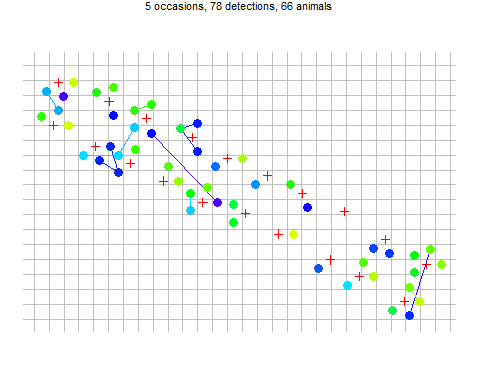
## $`1`  
## Object class capthist   
## Detector type proximity   
## Detector number 20   
## Average spacing 13.57938 m   
## x-range -623778.5 -623653.2 m   
## y-range -1187164 -1187091 m   
##   
## Counts by occasion   
## 1 2 3 4 5 Total  
## n 15 23 15 11 14 78  
## u 15 22 11 7 11 66  
## f 56 8 2 0 0 66  
## M(t+1) 15 37 48 55 66 66  
## losses 0 0 0 0 0 0  
## detections 15 23 15 11 14 78  
## detectors visited 8 13 9 10 10 50  
## detectors used 20 20 20 20 20 100  
##   
## $`2`  
## Object class capthist   
## Detector type proximity   
## Detector number 20   
## Average spacing 13.57938 m   
## x-range -623778.5 -623653.2 m   
## y-range -1187164 -1187091 m   
##   
## Counts by occasion   
## 1 2 3 4 5 6 Total  
## n 23 27 16 19 8 7 100  
## u 23 22 12 11 5 3 76  
## f 59 11 5 1 0 0 76  
## M(t+1) 23 45 57 68 73 76 76  
## losses 0 0 0 0 0 0 0  
## detections 24 27 17 21 8 7 104  
## detectors visited 14 12 12 15 8 7 68  
## detectors used 20 20 20 20 20 20 120

*n number of distinct individuals detected on each occasion t*  
u number of individuals detected for the first time on each occasion t  
*f number of individuals detected on exactly t occasions*  
M(t+1) cumulative number of detected individuals on each occasion t

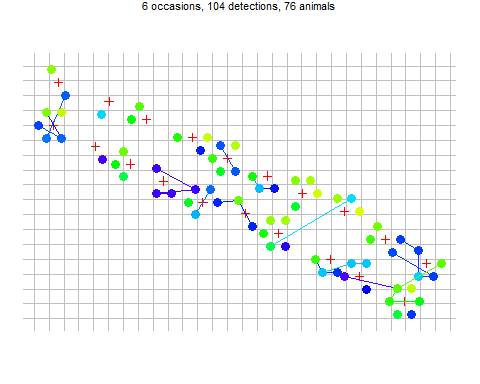
Now we use the plot method, which for capthist objects has additional arguments; we set tracks = TRUE to join consecutive captures of each individual.

par(mfrow = c(1,1), mar = c(1,1,1,1)) # reduce margins  
plot (multi, tracks = TRUE, gridsp = 5,border = 10)

## Warning in plot.capthist(multi, tracks = TRUE, gridsp = 5, border = 10):  
## screen layout does not allow for all sessions and some plots may be lost;  
## set par mfrow



## Warning in FUN(X[[i]], ...): track for repeat detections on same occasion  
## joins points in arbitrary sequence

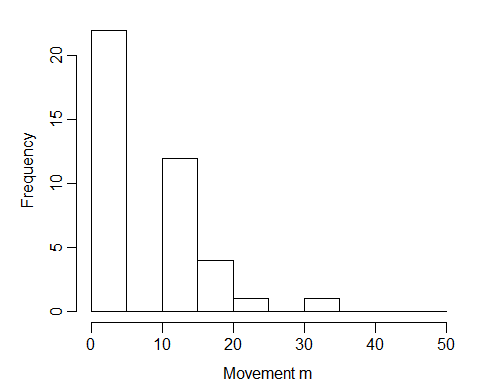


## 1 2   
## 78 104

The most important insight from this figure is that individuals tend to be recaptured near their site of first capture. This is expected when the individuals of a species occupy home ranges. In SECR models the tendency for detections to be localised is reflected in the spatial scale parameter σ.

Successive trap-revealed movements can be extracted with the moves function and summarised with hist:

m <- unlist(moves(multi))  
par(mar = c(3.2,4,1,1), mgp = c(2.1,0.6,0)) # reduce margins  
hist(m, breaks = seq(0/5, 50,5), xlab = "Movement m", main = "")



We will employ the estimate of the spatial scale σ from the July session (6.21) to fit the simplest possible SECR model with function secr.fit.

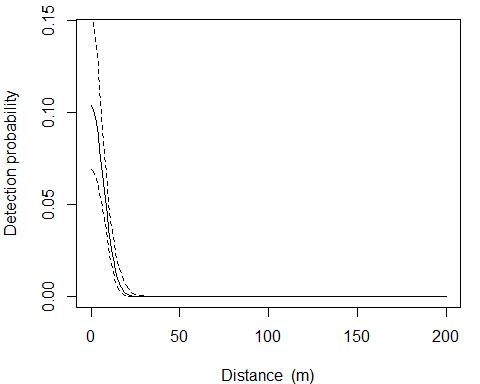
fit <- secr.fit (multi, buffer = 4 \* 6.21, trace = FALSE,biasLimit = NA, verify = FALSE)  
  
detector(traps(multi)) <- "proximity"  
  
fit

##   
## secr.fit(capthist = multi, buffer = 4 \* 6.21, verify = FALSE,   
## biasLimit = NA, trace = FALSE)  
## secr 3.2.1, 17:29:29 24 Jul 2019  
##   
## $`1`  
## Detector type proximity   
## Detector number 20   
## Average spacing 13.57938 m   
## x-range -623778.5 -623653.2 m   
## y-range -1187164 -1187091 m   
##   
## $`2`  
## Detector type proximity   
## Detector number 20   
## Average spacing 13.57938 m   
## x-range -623778.5 -623653.2 m   
## y-range -1187164 -1187091 m   
##   
##   
## 1 2  
## Occasions 5 6  
## Detections 78 104  
## Animals 66 76  
## Detectors 20 20  
##   
## Model : D~1 g0~1 sigma~1   
## Fixed (real) : none   
## Detection fn : halfnormal  
## Distribution : poisson   
## N parameters : 3   
## Log likelihood : -400.248   
## AIC : 806.4959   
## AICc : 806.6698   
##   
## Beta parameters (coefficients)   
## beta SE.beta lcl ucl  
## D 5.583114 0.15495204 5.279413 5.886814  
## g0 -2.156845 0.22702159 -2.601799 -1.711891  
## sigma 1.935015 0.09169641 1.755294 2.114737  
##   
## Variance-covariance matrix of beta parameters   
## D g0 sigma  
## D 0.024010134 -0.01368844 -0.004682768  
## g0 -0.013688442 0.05153880 -0.013199465  
## sigma -0.004682768 -0.01319946 0.008408232  
##   
## Fitted (real) parameters evaluated at base levels of covariates   
##   
## session = 1   
## link estimate SE.estimate lcl ucl  
## D log 265.8982236 41.45002678 196.25467951 360.2556917  
## g0 logit 0.1036933 0.02109962 0.06902272 0.1529186  
## sigma log 6.9241508 0.63625675 5.78514687 8.2874065  
##   
## session = 2   
## link estimate SE.estimate lcl ucl  
## D log 265.8982236 41.45002678 196.25467951 360.2556917  
## g0 logit 0.1036933 0.02109962 0.06902272 0.1529186  
## sigma log 6.9241508 0.63625675 5.78514687 8.2874065

The report comprises:  
• function call and time stamp  
• summary of the data  
• description of the model, including the maximized log likelihood, Akaike’s Information Criterion AIC  
• estimates of model coefficients (beta parameters)  
• estimates of variance-covariance matrix of the coefficients  
• estimates of the ‘real’ parameters

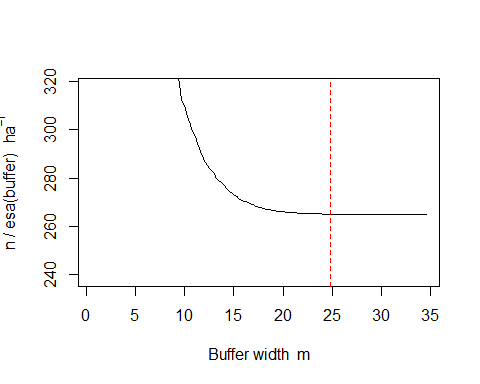
. The estimated density is 266 susliks per hectare, 95% confidence interval 196-360 susliks per hectare  
. The other two real parameters jointly determine the detection function, plotted below with 95% confidence limits

par(mar = c(4,4,1,1)) # reduce margins  
plot(fit, limits = TRUE)



The theory of SECR tells us that buffer width is not critical as long as it is wide enough that animals at the edge have effectively zero chance of appearing in our sample.We check that for the present model with the function esa.plot.  
The estimated density has easily reached a plateau at the chosen buffer width (dashed red line):

esa.plot(fit)  
abline(v = 4 \* 6.21, lty = 2, col = 'red')

 ```