Spatial models for detection parameters in **secr**

Murray Efford

2025-09-28

Contents

Introduction

This note describes extensions in **secr 5.3.0** that allow detection parameters to vary in space.

Historically, only the **secr** model for density D has allowed full spatial variation. Models for detection parameters (e.g., λ_0, σ) include other effects (time, individual, detector etc.), but not spatial variation *per se*. This was the result of a decision was taken in the original design to speed the computation of each likelihood (see Appendix for explanation).

secr 5.3.0 defines new 'real' parameters 'sigmaxy' and 'lambda0xy' that are modelled like D. This means that their respective models may include spatial coordinates and mask covariates. The internal mechanism involves re-scaling the distance matrix as described in Efford (2025, Appendix F) and the Appendix.

Simple example

This example uses a subset of the Orongorongo Valley brushtail possum data included in **secr**. See `?OVpossum` for a description of the data and code to create the mask `ovmask`

```
library(secr)
if (packageVersion('secr') < "5.3") stop ("requires secr version >= 5.3.0")
```

Start by fitting a null model for sigma (a warning is suppressed).

```
fit0 <- secr.fit(ovposs, mask = ovmask, detectfn = "HHN",
                 model = sigma~1, trace = FALSE)
predict(fit0)
```

```
##          link estimate SE.estimate     lcl     ucl
## D         log  14.3802    1.003093 12.5447 16.4842
## lambda0   log   0.1015    0.008948  0.0854  0.1206
## sigma     log  27.3765    0.972969 25.5349 29.3508
```

Now specify a null model for sigmaxy (we no longer need to specify userdist or explicitly fix sigma as in Efford (2025) Appendix F4.1).

```
fit1 <- secr.fit(ovposs, mask = ovmask, detectfn = "HHN",
    model = sigmaxy ~ 1, trace = FALSE)
predict(fit1)
```

```
##          link estimate SE.estimate     lcl     ucl
## D         log  14.3802    1.003093 12.5447 16.4842
## lambda0   log   0.1015    0.008948  0.0854  0.1206
## sigma     log   1.0000    0.000000  1.0000  1.0000
## sigmaxy   log  27.3765    0.972968 25.5349 29.3508
```

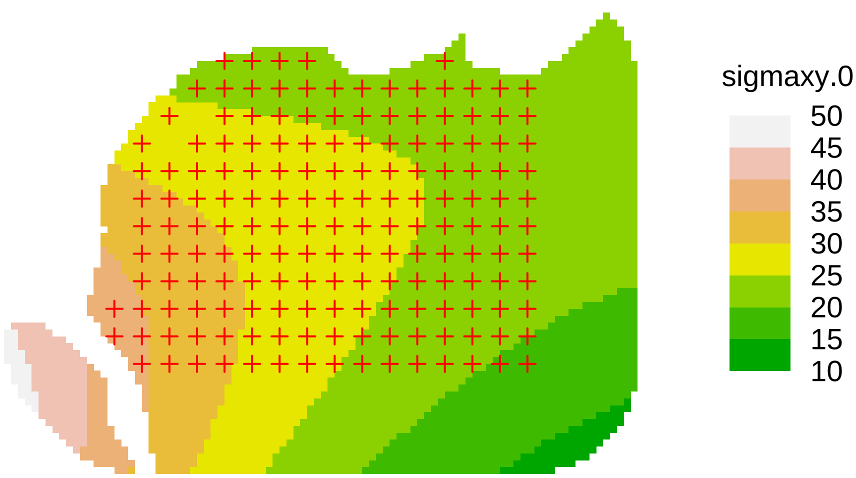
The output now has rows for both sigma and the new sigmaxy parameter. Note that sigma is fixed at 1.0 and sigmaxy has taken over. Next we do some real work: model sigmaxy as a quadratic surface:

```
fit2 <- secr.fit(ovposs, mask = ovmask, detectfn = "HHN",
    model = sigmaxy ~ x + y + x2 + y2 + xy, trace = FALSE)
predict(fit2)
```

```
##          link estimate SE.estimate      lcl     ucl
## D         log  14.6843    1.091474 12.69612 16.9838
## lambda0   log   0.1085    0.009626  0.09123  0.1291
## sigma     log   1.0000    0.000000  1.00000  1.0000
## sigmaxy   log  25.9243    1.301934 23.49554 28.6040
```

The estimate for sigmaxy has changed; it is now the predicted value at the centre of the habitat mask. More usefully, we can plot predicted sigmaxy across the habitat mask. To do this we treat the fitted spatial detection model as if it was a density model. Thus

```
plot(predictDsurface(fit2, parameter = 'sigmaxy'))
plot(traps(ovposs), add=TRUE)
```



We can compare the models; there is some evidence for spatial variation in σ, or at least something that is not captured by the null model. Further investigation is warranted.

```
AIC(fit0,fit1, fit2)[,-c(2,6)] # drop detectfn, AICc to save space
```

```
##                                                    model npar logLik  AIC  dAIC AICwt
## fit2 D~1 lambda0~1 sigma~1 sigmaxy~x + y + x2 + y2 + xy    8  -1543 3102  0.00     1
## fit0                             D~1 lambda0~1 sigma~1    3  -1556 3118 16.28     0
## fit1                   D~1 lambda0~1 sigma~1 sigmaxy~1    3  -1556 3118 16.28     0
```

Extensions

Shadow spatial parameters are also defined for the (less often used) parameters 'a0' and 'sigmak'.

Covariation of λ_{0xy} with σ_{xy}

'a0xy' is a special case that allows spatial variation in the 'a0' surrogate parameter (Efford 2025 Appendix H.2). If the model includes a formula for a0xy then

1. The intercept for lambda0 is fixed
2. λ_{0xy} for d''_{ik} in Table 1 is inferred from the modelled 'sigmaxy' using $\lambda_{0xy} = a_{0xy}/(2\pi\sigma^2_{xy})$

The main use is to compare models with a hard-wired relationship between λ_{0xy} and σ_{xy} (i.e., a0xy~1) to models in which each varies independently.

Covariation of σ_{xy} with D_{xy}

'sigmakxy' is a further special case that allows spatial variation in the 'sigmak' surrogate parameter (Efford 2025 Appendix H.2). If the model includes a formula for sigmakxy then

1. The intercept for sigma is fixed
2. σ_{xy} for d''_{ik} in Table 1 is inferred from the modelled density D_{xy} using $\sigma_{xy} = 100\ \sigma_{k\,xy}/\sqrt{D_{xy}}$

The main use is to compare models with a hard-wired relationship between σ_{xy} and D_{xy} (i.e., sigmakxy~1) to models in which each varies independently.

Cascading covariation is permitted in which a spatial model for density D_{xy} drives variation in σ_{xy} which in turn drives variation in λ_{0xy}.

Limitations

Link functions

The link functions of sigmaxy, lambda0xy, a0xy and 'sigmakxy' should not be changed from their default ('log').

Choice of detection function

'sigmaxy' and 'sigmakxy' are limited to simple detection functions in which σ appears as a divisor of d_{ik}, i.e. HN, HR, EX, HHN, HHR, HEX, HVP.

'lambda0xy' and 'a0xy' are further limited to HHN and HEX.

Although g0 is generally an alternative to lambda0, implying that HN and EX should also work (e.g. g0xy ~ x) this is not implemented as there is not a convenient null value on the default link scale (logit). Note logit(0) = 0.5 and logit^{-1}(1) = ∞.

Groups

Spatial detection parameters do not work with groups (CL = FALSE). This may change.

Interactions

Fixing the intercept of sigma allows other sigma models to operate that use individual, time or detector covariates (e.g. CL = TRUE, model = list(sigma ~ sex, sigmaxy~x+y)). However, interactions between sigmaxy and these covariates are not allowed.

A model with a0xy should not also include lambda0xy, as that is determined from a0xy and sigmaxy.

The spatial detection options cannot be used in combination with other dynamic non-Euclidean distance models because it takes over the 'userdist' function.

Functions not updated in 5.3.0

- pmixProfileLL

Reference

Efford, M. G. (2025) The SECR book. A handbook of spatially explicit capture–recapture methods. Version 1.0.2. https://murrayefford.github.io/SECRbook/.

Appendix

Technical background

This section gives detail on the implementation that would clutter the main presentation.

Historically in **secr**, submodels for detection parameters (e.g., $\log(\sigma_{ijklm}) = \beta X_{ijklm}$) are pre-computed for the unique levels of X_{ijklm} that appear in the model; these are accessed internally via a parameter index array (PIA) rather than recalculated for each i, j, k, l, m (individual, occasion, detector, session, mixture class). Spatial variation in density D is modelled separately as a function of AC location (mask cell), group and session; variation by individual, occasion, detector and mixture class does not apply to D.

The PIA framework does not allow direct modelling of spatial (AC-dependent) variation in detection parameters. However, it was noted previously that spatial variation in σ may be introduced by scaling distances. Changes in **secr** 5.3.0 make this trick more accessible and extend it to λ_0, a_0 and k.

We notice that σ appears in many detection functions [1] only as the ratio d_{ik}/σ where d_{ik} is the distance between the AC of animal {i} and a detector k. An *ad hoc* way to model spatial variation in a detection parameter such as sigma is to fix $\sigma = 1$ and model d_{ik} as a function of habitat covariates at the AC (Efford 2025 Appendix F). As initially proposed, this used the machinery for non-Euclidean distances in **secr** (i.e. parameter 'noneuc'). There are diverse alternative non-Euclidean models for d_{ik}, possibly using the habitat along the path between an AC and each detector (Efford 2025 Appendix F), but we focus on the initial case.

When one of the new spatial parameters appears in a model the effect is to –

- automatically fix the intercept of the corresponding parameter to zero (assuming log link) (e.g. details$fixedbeta[parindx$sigma[1]] <- 0).
- set the 'userdist' function for calculating detector-mask distances to a custom internal function modelled on `fn1()` from Appendix F.

[1] Detection functions 0,1,2,14,15,16,19 in **secr**

- pass scaled distances d'_{ik} or d''_{ik} to the usual detection algorithm. Typical formula for d'_{ik} (spatial variation in sigmaxy alone) and d''_{ik} are in the following table. These allow spatial variation in both sigma and lambda0 for detection functions HHN and HEX; there are variations for lambda0xy alone and a0 alone.

Table 1. Encoding of spatial variation in detection parameters as transformed distance (d', d''). The 'spatial' form corresponds to the 'simple' form when parameters λ_0 and σ are set to 1.0 and the spatial parameters $(\lambda_{0xy}, \sigma_{xy})$ are constant across space.

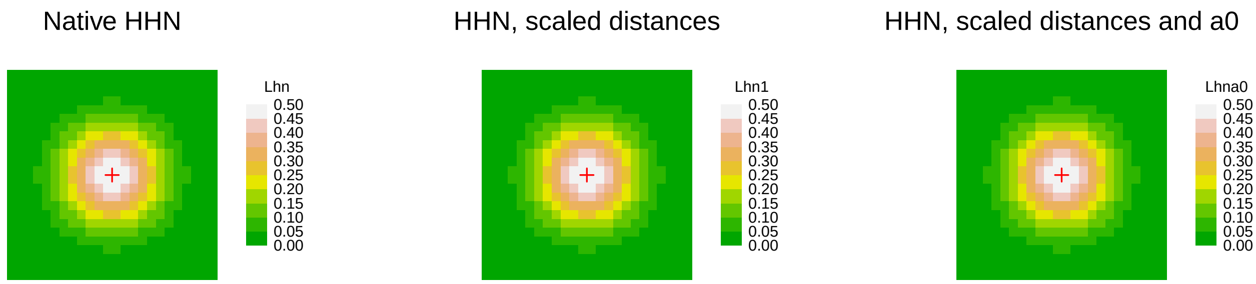
Detection function	Simple λ_{ik}	Spatial λ_{ik}	
HHN hazard half-normal	$\lambda_0 \exp\{-d_{ik}^2/(2\sigma^2)\}$	$\lambda_0 \exp\{-d_{ik}''^2/(2\sigma^2)\}$	$d''_{ik} = \sqrt{d_{ik}^2/\sigma_{xy}^2 - 2\log\lambda_{0xy}}$
HEX hazard exponential	$\lambda_0 \exp\{-d_{ik}/\sigma\}$	$\lambda_0 \exp\{-d''_{ik}/\sigma\}$	$d''_{ik} = d_{ik}/\sigma_{xy} - \log\lambda_{0xy}$
HVP hazard variable power	$\lambda_0 \exp\{-(d_{ik}/\sigma)^z\}$	$\lambda_0 \exp\{-(d'_{ik}/\sigma)^z\}$	$d'_{ik} = d_{ik}/\sigma_{xy}$

The usual output component 'parindx' includes components for both the detection parameter and its spatial shadow (e.g. sigma and sigmaxy). If the formula for the shadow defines a smooth then the usual machinery for non-Euclidean models is applied to generate a separate smoothsetup. Function `makeStart` applies the default starting value of the corresponding base parameter.

Validation

For the purposes of validation, code is provided here to derive $\lambda(i,k)$ (detection hazard given distance from mask point i to detector k) in three different ways and visually compare the results. The plotted values are the computed $\lambda(i,k)$ for points on a mask relative to a single, central, detector (+).

```
grid <- make.grid(1,1, spacing = 2)
msk  <- make.mask(grid, buffer=3, spacing = 0.25)
d    <- edist(msk, grid)    # i, k
lam0 <- 0.5
sig  <- 1
a0   <- 2 * pi * lam0 * sig^2
covariates(msk) <- data.frame(
    Lhn = lam0 * exp(-d^2/2/sig^2),
    Lhn1 = lambdahn1 <- exp(-(  sqrt(d^2/sig^2 - 2 * log(lam0))  )^2/2),
    Lhna0 = exp(-(  sqrt(d^2/sig^2 - 2 * log(a0/(2 * pi * sig^2)))  )^2/2)
    )
par(mfrow=c(1,3))
plot(msk, cov = 'Lhn', dots=F, border=1)
plot(grid, add=T)
mtext(side=3, 'Native HHN')
plot(msk, cov = 'Lhn1', dots=F, border=1)
plot(grid, add=T)
mtext(side=3, 'HHN, scaled distances')
plot(msk, cov = 'Lhna0', dots=F, border=1)
plot(grid, add=T)
mtext(side=3, 'HHN, scaled distances and a0')
```


Native HHN
Lhn
0.50
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
HHN, scaled distances
Lhn1
0.50
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
HHN, scaled distances and a0
Lhna0
0.50
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00