SECR for acoustic data

Murray Efford and Deanna Dawson

February 25, 2010

Sounds recorded at an array of microphones may be analysed to provide an estimate of the 'population density' of sound sources using an extension of the usual methods for spatially explicit capture-recapture (SECR) (Efford et al. 2009; Dawson and Efford 2009). This vignette shows how the analysis may be performed in package **secr**, using as an example the dataset `ovensong` from an acoustic survey of ovenbirds (*Seiurus aurocapilla*).

Background to acoustic analysis with SECR

Typical SECR jointly models the 2-D distribution of animal home-range centres and the probability an animal is detected over several time intervals at multiple detectors (traps), given their location. Acoustic SECR models the instantaneous 2-D distribution of sound sources and the acoustic power received during a brief recording at multiple microphones, given their location.

The acoustic analysis differs because (i) data come from a single interval rather than several 'occasions' (Efford et al. 2009), and (ii) each detection has an associated continuous measurement, a measure of 'signal strength'. Signal strength may be the average power associated with the recorded sound, as measured in software such as Raven Pro (Charif et al. 2008). A sound is considered to have been 'detected' at a microphone when the signal strength exceeds a threshold level. Sounds appear in the analysis only when they are detected on at least one microphone.

The acoustic model may be fitted by numerically maximizing the likelihood. As in SECR, the actual locations of the sound sources (= home-range centres) are unknown, and the likelihood is evaluated by integrating over a region containing all potential locations. The region is specified as a set of grid cells called a habitat mask.

The ovensong dataset

Over five days, four microphones were placed in a square (21-m side) centred at each of 75 points in a regular 50-m grid. Recordings of 5 minutes duration were made in .wav format on a 4-channel digital sound recorder. The data are estimates of average power on each channel (microphone) for the first song of each ovenbird distinguishable in a particular 5-minute recording. Power was estimated using a window of 0.7 s duration for frequencies between 4200 and 5200 Hz. When song in this frequency range was obscured by insect noise, power was measured for an alternative 1000-Hz range and the values adjusted by regression.

As usual in package **secr**, the data are arranged for analysis in a `capthist` object. The construction of such an object from input data is described in the help page for `make.capthist`. For sound data, the core of a `capthist` object is a 3-dimensional array of 0/1 codes indicating whether a sound was detected at each microphone; the 'occasion' (interval) dimension of the array always has length 1 because each sound is sampled only once at any microphone.

The `signal` attribute of an acoustic `capthist` object contains the signal strength (power) measurements in decibels as a vector with one value for each 'detection'. A 'detection' occurs when the measured power on a channel exceeds the power threshold (`cutval`). For the `signalCH` object, the power threshold (attribute `cutval` = 35) is less than any signal value (range 38.4 dB to 80.4 dB) and all detection histories are complete (1,1,1,1) across microphones. Some of these 'signal' measurements will be largely noise.

```
> library(secr)
> data(ovensong)
> summary(signalCH)

Object class      capthist
Detector type     signal
Detector number   4
Average spacing   21 m
x-range           0 21 m
y-range           0 21 m
Counts by occasion
                    1 Total
n                  76    76
u                  76    76
f                  76    76
M(t+1)             76    76
losses              0     0
detections        304   304
detectors visited   4     4
detectors used      4     4

Signal threshold  35
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  38.42   50.71   54.13   55.28   59.35   80.40
```

Note the number of detections is 4 times the number of different sounds (n) because every sound is detected on every microphone. For analysis we choose a higher threshold that treats weaker signals as 'not detected'. The choice of threshold is somewhat ad hoc; we use 52.5 dB because this excludes 95% of false positive signals background noise) while discarding few genuine ones (Dawson and Efford 2009).

```
> signalCH.525 <- subset(signalCH, cutval = 52.5)
> summary(signalCH.525)

Object class      capthist
Detector type     signal
```

```
Detector number   4
Average spacing   21 m
x-range           0 21 m
y-range           0 21 m
Counts by occasion
                    1 Total
n                  60    60
u                  60    60
f                  60    60
M(t+1)             60    60
losses              0     0
detections        180   180
detectors visited   4     4
detectors used      4     4

Signal threshold  52.5
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  52.59   55.01   58.04   59.60   62.99   80.40
```

By applying this threshold we lose 16 of the 76 original sounds (those that did not exceed the `cutval` on any microphone) and some of the remaining 60 sounds are 'undetected' on some microphones.

The `traps` attribute of `signalCH` and its offspring `signalCH.525` holds x-y coordinates for the relative positions of the microphones. In this example each 5-minute recording used the same 4-microphone array (a square centred on each point) and data are pooled across the 75 points.

```
> traps(signalCH.525)

   x  y
1  0  0
2  0 21
3 21 21
4 21  0
```

`signal(signalCH.525)` displays a vector of signal strengths, one for each detection. The data become more intelligible if we collapse the detection histories to a matrix and replace '1's with the matching signal strengths:

```
> temp <- signalCH.525[, 1, ]
> temp[temp > 0] <- round(signal(signalCH.525), 1)
> temp[1:8, ]

      [,1] [,2] [,3] [,4]
1055B 63.8 57.1 56.3 64.4
1055D 68.6 66.4 74.4 74.1
1155A 70.3 63.6 67.7 71.6
1155B 57.0 53.6  0.0 55.0
1250A  0.0  0.0  0.0 54.2
1250B 76.0 70.3 67.3 80.4
1250C  0.0  0.0  0.0 54.0
1255A 67.0 66.2 60.4 60.4
```

Each row corresponds to a sound, identified by the point number and individual ovenbird it's associated with (A, B, etc.), and each column to a microphone (we display just the first 8 sounds). Signals below the threshold appear as '0'.

Fitting the basic model

Now we can try fitting a model with `secr.fit`. First we define a habitat mask and starting values. We use a 200-m buffer rather than the default (100 m) to ensure that sounds at the edge of the mask are very unlikely to be detected (given what we eventually learn about attenuation). We use `trace = FALSE` to suppress output of the log likelihood during numerical maximisation. Fitting is straightforward:

```
> omask <- make.mask(traps(signalCH), buffer = 200)
> ostart <- c(log(20), 80, log(0.1), log(2))
> sound.1 <- secr.fit(signalCH.525, mask = omask, start = ostart,
     details = list(trace = FALSE))
```

(We could have dropped the `mask` argument of `secr.fit` and set its `buffer` argument to construct a habitat mask 'on the fly' rather than as a separate step. In this case we could also have omitted 'start' and used the default starting values.)

A warning message reminds us that we have fitted the default model for sound attenuation. This is a log-linear decline with distance from the sound source $S = \beta_0 + \beta_1 d + \epsilon$ where S is the signal strength in decibels, d is distance from the source in metres, and ϵ is a random normal error term with variance σ_s^2 dB. Detection probability is given by $g(d) = F((c - (\beta_0 + \beta_1 d))/\sigma_s)$ where F is the standard cumulative normal distribution, c is the signal threshold.

The `print` method for secr objects displays data summaries and parameter estimates and other useful results:

```
> sound.1

secr.fit( capthist = signalCH.525, mask = omask, start = ostart,
  details = list(trace = FALSE) )
secr 1.3.0, 20:10:05 11 Mar 2010

Detector type     signal
Detector number   4
Average spacing   21 m
x-range           0 21 m
y-range           0 21 m
N animals       :  60
N detections    :  180
N occasions     :  1
Mask area       :  17.7241 ha

Model           :  D~1 beta0~1 beta1~1 sdS~1
Fixed (real)    :  none
Detection fn    :  signal strength
Distribution    :  poisson
```

```
N parameters    :  4
Log likelihood  :  -470.2435
AIC             :  948.487
AICc            :  949.2142

Beta parameters (coefficients)
            beta    SE.beta       lcl        ucl
D      2.6378555 0.17550978  2.293863  2.9818484
beta0 78.1648006 1.35818646 75.502804 80.8267972
beta1 -1.3782941 0.05813555 -1.492238 -1.2643506
sdS    0.6386267 0.08282640  0.476290  0.8009635

Variance-covariance matrix of beta parameters
                  D       beta0         beta1           sdS
D      0.0308036813 -0.08370856  0.0046641344 -0.0002065143
beta0 -0.0837085611  1.84467047  0.0221276481 -0.0265022791
beta1  0.0046641344  0.02212765  0.0033797422 -0.0008674606
sdS   -0.0002065143 -0.02650228 -0.0008674606  0.0068602119

Fitted (real) parameters evaluated at base levels of covariates
          link   estimate SE.estimate        lcl        ucl
D          log 13.9831847  2.47320694  9.9131551 19.7242403
beta0 identity 78.1648006  1.35818646 75.5028041 80.8267972
beta1   neglog -0.2520081  0.01466302 -0.2248689 -0.2824227
sdS        log  1.8938783  0.15713253  1.6100898  2.2276862
```

The fitted density 'D' is the estimated density of sound sources inflated by the number of replicate points in the pooled dataset. We therefore divide by 75 to get the estimated density per hectare (0.186, SE 0.033).

The fitted parameters beta0, beta1 and sdS correspond to the parameters β_0, β_1 and σ_s and define the detection function (see Dawson and Efford 2009 for more on this). The 'link' column reminds us that the 'beta' parameters (all 4 of them) are maximized on their transformed (link) scales; the confidence limits (lcl, ucl) are also computed on that scale and back-transformed. The default link for beta1 is the unorthodox neglog(x) = log(-x); this imposes the intuitively sensible constraint that acoustic power should decline with distance from the source ($\beta_1 < 0$).

Adding spherical spreading

Log-linear sound attenuation (x dB per 100 metres) is only a rough approximation. For greater realism we can 'hardwire' the inverse-square reduction in sound energy with distance that that is expected when a sound radiates from a point source. This is termed 'spherical spreading' and results in 6 dB loss for each doubling of distance. When attenuation includes spherical spreading we measure distances relative to a point 1 m from the sound source, rather than true zero.

To fit a model with spherical spreading we specify `detectfn = 11`, rather the default `detectfn = 10` (numeric codes for detection functions are listed on the 'Detection functions' help page).

```
> sound.2 <- secr.fit(signalCH.525, detectfn = 11, mask = omask,
     start = ostart, details = list(trace = FALSE))
```

We can compare the fit of the models with the AIC method for secr objects:

```
> AIC(sound.1, sound.2)

                             model                  detectfn npar
sound.2 D~1 beta0~1 beta1~1 sdS~1 signal strength spherical    4
sound.1 D~1 beta0~1 beta1~1 sdS~1           signal strength    4
           logLik     AIC    AICc  dAICc AICwt
sound.2 -463.9727 935.945 936.673  0.000     1
sound.1 -470.2435 948.487 949.214 12.541     0
```

The spherical spreading term substantially increases the log likelihood and reduces AIC without adding any parameters. What effect does this have on the density estimates? The collate function in **secr** is a convenient way to compare parameter estimates. Here we select density estimates from the first session (there's only one) and adjust for replication:

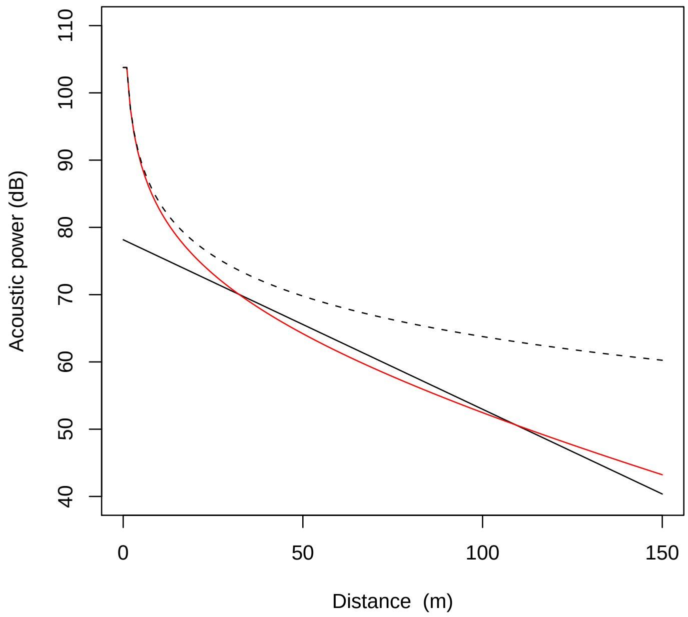
```
> collate(sound.1, sound.2)[1, , , "D"]/75

         estimate SE.estimate       lcl       ucl
sound.1 0.1864425  0.03297609 0.1321754 0.2629899
sound.2 0.1910839  0.02853758 0.1428240 0.2556508
```

The effect of spherical spreading on $\hat{D}$ is minimal, just a slight narrowing of the confidence interval. We consider the different fitted attenuation curves in the next section.

Attenuation curves and detection functions

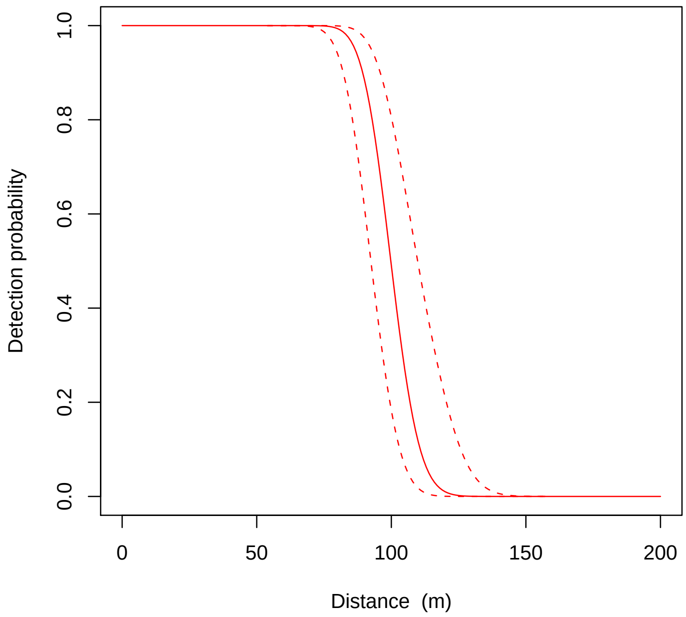
In acoustic SECR, the detection model (probability of detection as a function of distance) follows from the model for sound attenuation, so it makes sense to start by examining the fitted attenuation model. We extract the real coefficients of the log-linear and spherical-spreading models and then plot the respective attenuation curves. We also show the effect of spherical spreading alone by setting beta1 to zero (dashed line).

```
> pars1 <- predict(sound.1)[c("beta0", "beta1"), "estimate"]
> pars2 <- predict(sound.2)[c("beta0", "beta1"), "estimate"]
> attenuationplot(pars1, xval = 0:150, spherical = FALSE,
     ylim = c(40, 110))
> attenuationplot(pars2, xval = 0:150, spherical = TRUE,
     add = TRUE, col = "red")
> pars2[2] <- 0
> attenuationplot(pars2, xval = 0:150, spherical = TRUE,
     add = TRUE, lty = 2)
```



The `plot` method for secr model objects is a direct way to display the detection function. By default, `plot.secr` also shows 95% confidence limits for the detection function. These use the asymptotic variance estimates and a first-order delta-method approximation.

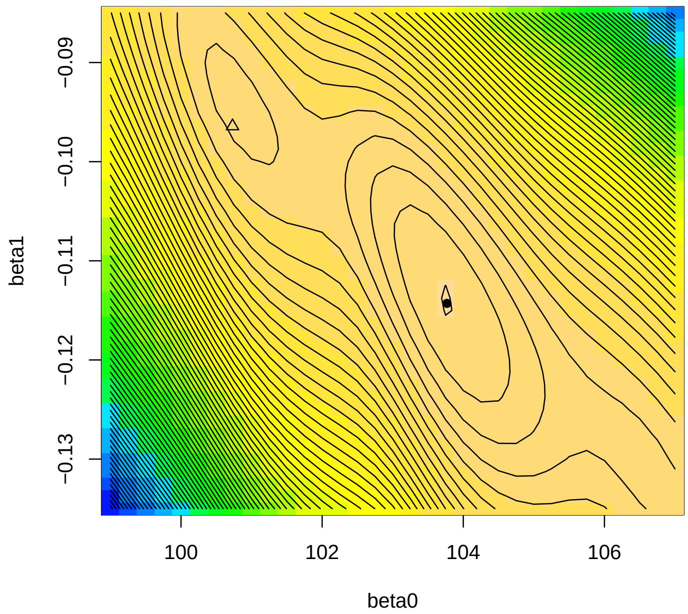
```
> plot(sound.2, col = "red")
```



Local maxima in likelihood surface

Including a spherical spreading term in the sound attenuation model causes the likelihood surface to become multimodal, at least in this example. Newton-Raphson, the default maximization method in secr.fit, is particularly inclined to settle on a local maximum, so care is needed. In the example above we cheated by specifying starting values for $(D, \beta_0, \beta_1, \sigma_s)$ on their respective link scales that had been found by trial and error to yield the global maximum of the likelihood surface. Here we investigate the issue further by plotting the log likelihood surface for the spherical spreading model. D (0.191 / ha) and σ_s (1.68 dB) are held constant while β_0 and β_1 are varied. We use the `LLsurface.secr` function with plotting suppressed to generate a data matrix that we then use for a customised plot.

```
> beta0 <- seq(99, 107, 0.25)
> beta1 <- seq(-0.135, -0.085, 0.00125)
> LL <- LLsurface.secr(sound.2, betapar = c("beta0", "beta1"),
     xval = beta0, yval = beta1, plot = FALSE)

Evaluating log likelihood across grid of 1353 points...

> image(LL, x = beta0, y = beta1, breaks = c(-510, seq(-500,
     -460, 2)), col = topo.colors(21))
> contour(LL, x = beta0, y = beta1, add = TRUE, levels = seq(-510,
     -460, 0.5), drawlabels = FALSE)
> points(103.77, -exp(-2.169), pch = 16)
> points(100.73, -exp(-2.339), pch = 2)
```



Contours are at spacings of one half a log-likelihood unit. The plotted points correspond to the global maximum likelihood (solid dot) and a local maximum (triangle).

Other approaches to acoustic analysis with SECR

Our example used measurements of relative acoustic power. The same method may be used with signal strength defined in other ways, so long as the measure is expected to decline steadily with distance from the source. For example, spectrogram cross correlation scores may be suitable.

One can also model signal attributes that encode source location in ways other than scalar 'signal strength'. Time of arrival of sounds at different microphones is one such attribute, and the intersection of bearings to each source is another (bearings may be obtained from multiple arrays of closely spaced microphones, under some conditions). These analyses are not yet provided in **secr**. It is not difficult to include time delays and bearings in the likelihood, but the resulting models are less elegant than those based on signal strength because an additional component is needed to explain detection and nondetection as a function of distance (the signal strength model serves both purposes). Time and bearing data are often more difficult to collect than signal strengths, and our unpublished simulations suggest the resulting estimates of density are typically no better than those based on signal strength.

References

Charif, R. A., Waack, A. M. and Strickman, L. M. (2008) Raven Pro 1.3 User's Manual. Cornell Laboratory of Ornithology, Ithaca, New York.

Dawson, D. K. and Efford, M. G. (2009) Bird population density estimated from acoustic signals. *Journal of Applied Ecology* **46**, 1201–1209.
Efford, M. G., Dawson, D. K. and Borchers, D. L. (2009) Population density estimated from locations of individuals on a passive detector array. *Ecology* **90**, 2676–2682.