Capture-recapture models in `unmarked`

Richard Chandler

January 30, 2012

Abstract

The "`un`" in `unmarked` is somewhat misleading because the package can be used to analyze data from marked animals. The three most common sampling methods that produce suitable data are removal sampling, double observer sampling, and capture-recapture methods. This document focuses on the analysis of capture-recapture data using a class of models known as multinomial N-mixture models [Royle, 2004, Chandler et al., 2011], which assume that capture-recapture data have been collected at a collection of sample locations ("sites"). Capture-recapture models can be fitted with constant parameters (M_0), time-specific parameters (M_t), and behavioral responses (M_b). In addition, spatial variation in abundance or capture probability can also be modeled using site-specific covariates. unmarked has two functions for fitting capture-recapture models: multinomPois and gmultmix. Both allow for user-defined functions to describe the capture process, and the latter allows for modeling of temporary emigration when data have been collected using the so-called robust design [Kendall et al., 1997, Chandler et al., 2011].

Introduction

In traditional capture-recapture models, n individuals are captured at a site during the course of J sampling occasions. The encounter history for each individual is used as information about capture probability p such that the total population size N can be regarded as the size parameter of a binomial distribution, $n \sim \text{Binomial}(N, p)$.

Although traditional capture-recapture models are useful for estimating population size when $p < 1$, they do not allow one to model variation in abundance which is a central focus of much ecological research. Royle [2004] developed a framework for modeling variation in both abundance and capture probability when capture-recapture data is collected at a set of R sites. Site-specific abundance ($N_i; i = 1, 2, ..., R$) is regarded as latent variable following a discrete distrubution such as the Poisson or negative binomial. The encounter histories are then tabulated at each site so that they can be regarded as an outcome of a multinomial distribution with cell probabilities π determined by a protocol-specific function of capture probability (see next section for details). Assuming a Poisson distribution, the model can be written as

$$\begin{aligned} N_i &\sim \text{Poisson}(\lambda) \\ \mathbf{y_i}|N_i &\sim \text{Multinomial}(N_i, \pi(p)) \end{aligned} \tag{1}$$

In the above, λ is the expected number of individuals at each site. $\mathbf{y_i}$ is a vector containing the number of individuals with encounter history $k; k = 1, 2, ...K$ at site i. The number of observable encounter histories K depends on the sampling protocol. For a capture-recapture study with 2 time periods, $K = 3$ because the possibilities are $H = (11, 10, 01)$. In Equation 1, $\pi(p)$ is a function that that converts capture probability p to multinomial cell probabilities, *i.e.*, the proportion of individuals expected to have capture history k. The definition of $\pi(p)$ is also specific to the sampling protocol. For example, the cell probabilities corresponding to the capture histories listed above are

$$\pi(p) = \{p^2, p(1-p), (1-p)p.$$

The probability of not capturing an individual in this case ($H = 00$) is $(1-p)^2$.

Spatial variation in abundance can be modeled using covariates with a log-link function

$$\log(\lambda_i) = \beta_0 + \beta_1 x_i$$

where x_i is some site-specific covariate such as habitat type or elevation. Of course, multiple covariate effects can be consered an a more general form of the above can be written as $\log(\lambda_i) = \mathbf{Xi'}\beta$ where

$\mathbf{X}$ is a design matrix and β is a vector of coefficients, possibly including an intercept. Capture probability can be modeled in the same way using the logit- instead of the log-link. For instance, we could have

$$\text{logit}(p_{ij}) = \alpha_0 + \alpha_1 v_{ij}$$

where v_{ij} is some covariate specific to the site and capture occasion.

Data

As previously mentioned, the data required by unmarked are an $R \times K$ matrix in which each row is the vector of tabulated encounter histories for animals captured at some site. Capture-recapture data, however, is typically recorded in the format shown in Table 1.

Table 1: Capture-recapture data for 3 individuals. There were 3 trapping occasions

Animal ID	Site	Capture history
GB	A	101
YR	A	101
RO	A	111
PP	A	100
GY	B	100
PR	B	010

In the absence of individual covariates, the data in Table 1 can be converted to the requisite format as shown in Table 2. Notice that no captures were made in sites C and D. It is important that such sites are retained in the analysis in order to make inference about spatial variation in abundance.

Table 2: Capture-recapture data from Table 1 in the format required by unmarked

Site	Encounter history						
	100	010	001	110	011	101	111
A	1	0	0	0	0	2	1
B	1	1	0	0	0	0	0
C	0	0	0	0	0	0	0
D	0	0	0	0	0	0	0

Analysis in unmarked

Closed population capture-recapture models

In this example we will analyze point count data collected on alder flycatchers (*Empidonax alnorum*) by Chandler et al. [2009]. Point count data such as these are collected on unmarked animals, but one can apply capture-recapture models because it is possible to keep track of inidividual birds during a short period of time. That is, we can pretend like birds are marked by noting which time intervals they are detected in during a short survey. The alder flycatcher data were collected using fixed-area 15-minute point counts, which were divided into 3 5-minute intervals. Each point was surveyed 3 times during 2005. The following command imports the capture histories for 50 individuals detected in 2005 at 49 point count locations.

```
> alfl <- read.csv(system.file("csv", "alfl.csv", package="unmarked"))
> head(alfl, 5)
        id survey interval1 interval2 interval3
1 crick1_05      1         1         1         1
2 crick1_05      3         1         0         1
3   his1_05      1         0         1         1
4   his1_05      1         1         1         1
5   his1_05      2         0         1         1
```

We see 5 rows of data representing the encounter histories for 5 birds detected at 2 points during 3 survey occasions. From these 5 birds, it appears as though detection probability is high since each bird was detected during at least 2 of the three time intervals.

Associated with the bird data are site- and visit-specific covariates for each of the 49 sites. We can import these data using the following command:

```
> alfl.covs <- read.csv(system.file("csv", "alflCovs.csv",
     package="unmarked"), row.names=1)
> head(alfl.covs)
          struct woody time.1 time.2 time.3 date.1 date.2 date.3
crick1_05   5.45  0.30   8.68   8.73   5.72      6     25     34
his1_05     4.75  0.05   9.43   7.40   7.58     20     32     54
hisw1_05   14.70  0.35   8.25   6.70   7.62     20     32     47
hisw2_05    5.05  0.30   7.77   6.23   7.17     20     32     47
kenc1_05    4.15  0.10   9.57   9.55   5.73      8     27     36
kenc2_05    9.75  0.40   9.10   9.12   9.12      8     27     36
```

Each row of this data.frame corresponds to a point count location. The variable struct is a measure of vegetation structure, and woody is the percent cover of woody vegetation at each of the 50-m radius plots. Time of day and date were measured for each of the three visits.

To format the data for unmarked, we need to tabulate the encounter histories for each site. Before doing so, let's first put our capture histories in a single column. Let's also be explicit about the levels of our factors for both the newly created captureHistory column and the point id column.

```
> alfl$captureHistory <- paste(alfl$interval1, alfl$interval2, alfl$interval3, sep="")
> alfl$captureHistory <- factor(alfl$captureHistory,
     levels=c("001", "010", "011", "100", "101", "110", "111"))
> ## Don't do this:
> #levels(alfl$id) <- rownames(alfl.covs)
> alfl$id <- factor(alfl$id, levels=rownames(alfl.covs))
```

Specifying the levels of captureHistory ensures that when we tabulate the encounter histories, we will include zeros for histories that were not observed. Similarly, setting the levels of alfl$id tells R that there were some sites where no ALFL were detected. This way, when we tabulate the data, we get a frequency for each site, not just the ones with > 1 detection. Here are the commands to extract data from the first primary period and to tabulate the encounter histories.

```
> alfl.v1 <- alfl[alfl$survey==1,]
> alfl.H1 <- table(alfl.v1$id, alfl.v1$captureHistory)
> head(alfl.H1, 5)

            001 010 011 100 101 110 111
  crick1_05   0   0   0   0   0   0   1
  his1_05     0   0   1   0   0   0   1
  hisw1_05    0   0   0   0   0   0   0
  hisw2_05    0   0   0   0   0   0   1
  kenc1_05    0   0   0   0   0   0   0
```

The object alfl.H1 contains the tabulated capture histories for each site. This is the format required by unmarked. The data from the first 5 sites suggest that detection probability was high since the most common encounter history was 111.

Now we are almost ready to create our unmarkedFrame and begin fitting models. We will fit our first series of models using the multinomPois function, which requires data formated using the unmarkedFrameMPois function. This constructor function recognizes two types of capture-recapture data: removal sampling data and double observer sampling data. In the future, we may add an option to automatically handle standard capture-recapture data too, but here we show how to supply it using a user-defined piFun, which allows extreme flexibility in converting detection probability to multinomial cell probabilities π. The piFun must take a matrix of detection probabilities with as many columns as there are secondary sampling periods (3 in this case), and convert them to a matrix of multinomial cell probabilities with K columns. Each column corresponds to the probability of observing the encounter history k. Here is a piFun to compute the multinomial cell probabilities when there were 3 sampling occasions. Note that the probabilities are simply p for a detection and $1 - p$ for a non-detection.

```
> crPiFun <- function(p) { # p should have 3 columns
     cbind("001" = (1-p[,1]) * (1-p[,2]) * p[,3],
           "010" = (1-p[,1]) * p[,2]     * (1-p[,3]),
           "011" = (1-p[,1]) * p[,2]     * p[,3],
           "100" = p[,1]     * (1-p[,2]) * (1-p[,3]),
           "101" = p[,1]     * (1-p[,2]) * p[,3],
           "110" = p[,1]     * p[,2]     * (1-p[,3]),
           "111" = p[,1]     * p[,2]     * p[,3])
 }
```

To demonstrate how this works, imagine that we surveyed 2 sites and detection probability was constant ($p = 0.2$) among sites and survey occasions. The function converts these capture probabilities to multinomial cell probabilites. Note that these cell probabilites will sum to < 1 if capture probability less than 1 over the 3 occasions.

```
> p <- matrix(0.4, 2, 3)
> crPiFun(p)
       001   010   011   100   101   110   111
[1,] 0.144 0.144 0.096 0.144 0.096 0.096 0.064
[2,] 0.144 0.144 0.096 0.144 0.096 0.096 0.064
> rowSums(crPiFun(p))
[1] 0.784 0.784
```

When providing a user-defined piFun, we also need to provide information about how to handle missing values. That is, if we have a missing value in a covariate, we need to know which values of **y** are affected. In unmarked, this can be done by supplying a mapping-matrix to the obsToY argument in the unmarkedFrameMPois function. obsToY needs to be a matrix of zeros and ones with the number of rows equal to the number of columns for some obsCov, and the number columns equal to the number of columns in y If obsToY[j,k] is 1, then a missing value in obsCov[i,j] translates to a missing value in y[i,k]. For capture-recapture data, all elements of obsToY should be 1.

```
> o2y <- matrix(1, 3, 7)
```

We are now ready to create the unmarkedFrame. In order to fit model M_t, we need a covariate that references the secondary sampling period, which we call intervalMat below. We also provide a couple of site-specific covariates: the percent cover of woody vegetation and vegetation structure.

```
> library(unmarked)
> intervalMat <- matrix(c('1','2','3'), 50, 3, byrow=TRUE)
> class(alfl.H1) <- "matrix"
> umf.cr1 <- unmarkedFrameMPois(y=alfl.H1,
     siteCovs=alfl.covs[,c("woody", "struct", "time.1")],
     obsCovs=list(interval=intervalMat),
     obsToY=o2y, piFun="crPiFun")
```

Writing a piFun and creating the obsToY object are the hardest parts of a capture-recapture analysis in unmarked. Again, this is done automatically for removal models and double observer models, and we may add an option to do this automatically for capture-recapture data too, but hopefully the flexibility allowed by specifying user-defined functions is evident.

Now that we have our data formatted we can fit some models. The following correspond to model M_0, model M_t, and a model with a continous covariate effect on p.

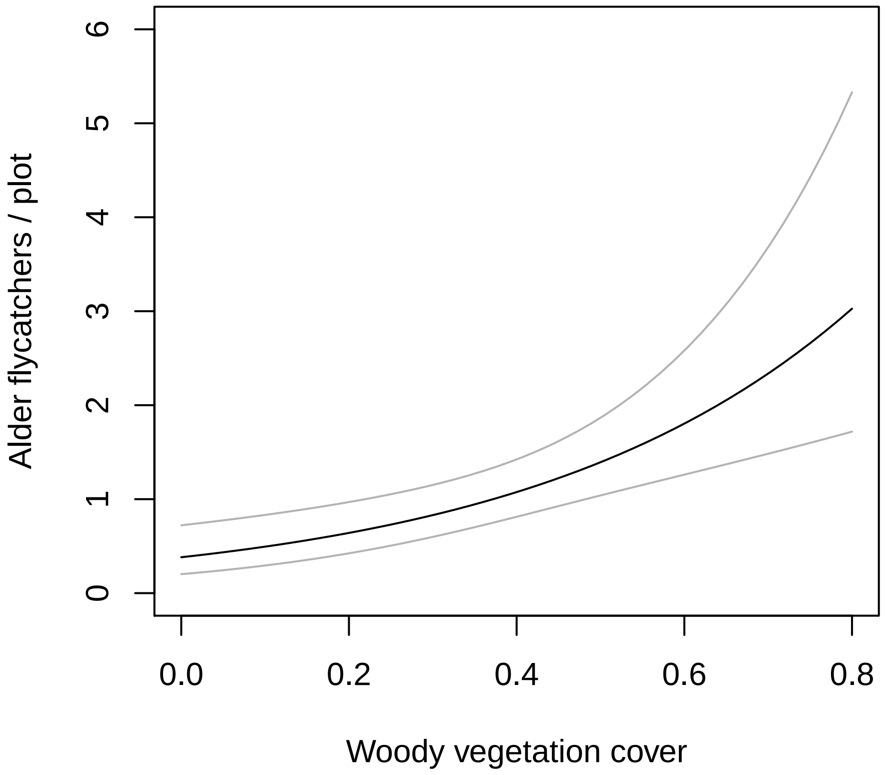
```
> M0 <- multinomPois(~1 ~1, umf.cr1)
> Mt <- multinomPois(~interval-1 ~1, umf.cr1)
> Mc <- multinomPois(~time.1 ~1, umf.cr1)
```

These sort of models can be fit in other software programs. What is unique about unmarked is that we can also model variation in abundance among sites. The following model treats abundance as a function of the percent cover of woody vegetation.

```
> (M0.woody <- multinomPois(~1 ~woody, umf.cr1))
Call:
multinomPois(formula = ~1 ~ woody, data = umf.cr1)

Abundance:
            Estimate    SE     z  P(>|z|)
(Intercept)   -0.962 0.325 -2.96 0.003059
woody          2.587 0.680  3.80 0.000143

Detection:
 Estimate    SE    z  P(>|z|)
     1.43 0.216 6.63 3.42e-11

AIC: 245.9301
```

This final model has a much lower AIC score than the other models, and it indicates that ALFL abundance increases with the percent cover of woody vegetation. We can plot this relationship by predicting abundance at a sequence of woody vegetation values.

```
> nd <- data.frame(woody=seq(0, 0.8, length=50))
> E.abundance <- predict(M0.woody, type="state", newdata=nd, appendData=TRUE)
> plot(Predicted ~ woody, E.abundance, type="l", ylim=c(0, 6),
      ylab="Alder flycatchers / plot", xlab="Woody vegetation cover")
> lines(lower ~ woody, E.abundance, col=gray(0.7))
> lines(upper ~ woody, E.abundance, col=gray(0.7))
```



What about detection probability? Since there was no evidence of variation in p, we can simply backTrasform the logit-scale estimate to obtain $\hat{p}$.

```
> backTransform(M0.woody, type="det")
Backtransformed linear combination(s) of Detection estimate(s)

 Estimate     SE LinComb (Intercept)
    0.808 0.0336    1.43           1

Transformation: logistic
```

As suggested by the raw data, detection probability was very high. The cooresponding multinomial cell probabilities can be computed by pluggin in this estimate of detection probability into our piFun. This can process is automated using the getP function.

```
> head(round(getP(M0.woody), 2), 1)
      001  010  011  100  101  110  111
[1,] 0.03 0.03 0.13 0.03 0.13 0.13 0.53
```

Note that the encounter probability most likely to be observed was 111. In fact p was so high that the probability of not detecting an alder flycatcher was essentially zero, $(1 - 0.81)^3 = 0.007$.

Capture-recapture models allowing for temporary emigration

In the previous analysis we used data from the first visit only. Chandler et al. [2011] proposed a model that allows us to make use of the entire alder flycatcher dataset. The model is similar to the temporary emigration model of Kendall et al. [1997] except that we are interested in modeling variation in abundance among sites.

The model assumes that no births or deaths occur during the study period, but animals may move on and off the plots between sampling occasions. This type of movement is referred to as temporary emigration. To account for it, define M to be the super-population size, the total number of individuals that use site i during the study period. Assuming that we visit each site on T occasions, primary periods, define N_{it} to be the subset of M_i exposed to sampling during occasion t. We now collect capture-recapture data at each site during each primary period t, and obtain the data $\mathbf{y_{ijt}}$. The model can be written as

$$\begin{aligned} M_i &\sim \text{Poisson}(\lambda) \\ N_{it}|M_i &\sim \text{Binomial}(M_i, \phi) \\ \mathbf{y_{it}}|N_{it} &\sim \text{Multinomial}(N_{it}, \pi(p)) \end{aligned} \tag{2}$$

where ϕ is the probability of being available for capture. This can be modeled as a function of covariates using the logit-link.

The data structure for the robust design is more complex than before, but it is easy to create in R. We can once again use the table function—but this time, we create a three-dimensional table rather than a two-dimensional one. We also need to expand the obsToY mapping matrix. This isn't so intuitive, but the commands below are generic and can be applied to other capture-recapture designs.

```
> alfl.H <- table(alfl$id, alfl$captureHistory, alfl$survey)
> alfl.Hmat <- cbind(alfl.H[,,1], alfl.H[,,2], alfl.H[,,3])
> nVisits <- 3
> o2yGMM <- kronecker(diag(nVisits), o2y)
> umf.cr <- unmarkedFrameGMM(y=alfl.Hmat,
     siteCovs=alfl.covs[,c("woody", "struct")],
     yearlySiteCovs=list(date=alfl.covs[,3:5], time=alfl.covs[,6:8]),
     obsCovs=list(interval=cbind(intervalMat,intervalMat,intervalMat)),
     obsToY=o2yGMM, piFun="crPiFun", numPrimary=nVisits)
```

Notice that we have 3 types of covariates now. The site-specific covariates are the same as before. Now, however, the observation covariates must match the dimensions of the $\mathbf{y}$ matrix. We can also have a class of covariates that vary among primary periods but not within primary periods. These are called yearlySiteCovs, which is a misleading name. It is a carry-over from other "open population" models in unmarked, but it should be remembered that these models are most suitable for data from a single year, since we assume no births or mortalities.

We can fit the model using the gmultmix function, which has a slightly different set of arguments. Rather than a single formula, the function takes 3 formulas for abundance covariates, availability covariates, and detection covariates in that order.

```
> (fm1 <- gmultmix(~woody, ~1, ~time+date, umf.cr))
Call:
gmultmix(lambdaformula = ~woody, phiformula = ~1, pformula = ~time +
    date, data = umf.cr)

Abundance:
            Estimate    SE      z P(>|z|)
(Intercept)  -0.0896 0.393 -0.228 0.81953
woody         2.4294 0.591  4.107 0.00004

Availability:
 Estimate    SE     z P(>|z|)
   -0.782 0.495 -1.58   0.114

Detection:
            Estimate     SE      z  P(>|z|)
(Intercept)   2.1050 1.1139  1.890 0.058792
time         -0.0499 0.0142 -3.526 0.000422
date         -0.0285 0.1369 -0.208 0.835258

AIC: 580.4508
```

Results from this model are similar to those obtained using the subset of data, but the standard error for the woody estimate has decreased.

Individual Heteogeneity in Capture Probability

The capture-recapture models that can be fit in unmarked assume that variation in capture probability can be explained by site, time, or behavioral factors such as trap-shyness or trap-happiness. Individual heterogeneity cannot be explicitly modeled, although, one could partition the data into strata and analyze the strata separately. For example, sex-specific differences could be studied by dividing the data into 2 subsets. Continuous animal-specific covariates, however, cannot be considered in unmarked. The so-called model M_h, which assumes random variation in capture probability among individuals is also not allowed. Some might find comfort in this given the concerns about M_h raised by Link [2003].

Another source of individual heterogeneity in capture probability arises from the distance between an individual's area of activity and the trap location. Traditional capture-recapture models ignore this important source of variation in capture probability, but recently developed spatial capture-recapture models overcome this limitation. SCR models also yield estimates of density rather than just population size, which is important when the study area cannot be well-defined. See xyz for more information.

Spatial capture-recapture models would not work well with the alder flycatcher data because there were no "spatial recaptures". That is, SCR models work best when individuals are encountered at multiple locations, but the points in the alder flycatcher dataset were independent of one another. Furthermore, distance-related heterogeneity in detection was probably insubstantial for this dataset because the sample plots were small (0.785 ha) and only data from singing individuals was included.

References

R. B. Chandler, J. A. Royle, and D. I. King. Inference about density and temporary emigration in unmarked populations. *Ecology*, 92(7):1429–1435, July 2011.

R.B. Chandler, D.I. King, and S. DeStefano. Scrub-shrub bird habitat associations at multiple spatial scales in beaver meadows in massachusetts. *The Auk*, pages 186–197, 2009.

Ian Fiske and Richard Chandler. **unmarked**: An **R** package for fitting hierarchical models of wildlife occurrence and abundance. *Journal of Statistical Software*, 43(10):1–23, 2011. URL `http://www.jstatsoft.org/v43/i10/`.

William L. Kendall, James D. Nichols, and James E. Hines. Estimating temporary emigration using capture-recapture data with Pollock's robust design. *Ecology*, 78(2):563–578, 1997.

W.A. Link. Nonidentifiability of population size from capture-recapture data with heterogeneous detection probabilities. *Biometrics*, 59(4):1123–1130, 2003.

J. A. Royle. Generalized estimators of avian abundance from count survey data. *Animal Biodiversity and Conservation*, 27(1):375–386, 2004.