Single-Species Single-Season Occupancy Model in unmarked

EEB 5894-003 "Detection, Occurrence, and Abundance"

Lab: Tuesday, September 15, 2015

Today we are going to do our first 'official' occupancy model, following the model structure of Mackenzie *et al.* 2002. This model estimates the probability of detection (p) and the probability of occupancy (ψ) of a species. As we will have gone over in class, the repeated pattern of detection histories can be used to create a likelihood that is a function of these two variables, p and ψ. We can then use computational algorithms to maximize that likelihood and recover the most likely estimates of both variables. Rather than doing this mathematics by hand in lab, we will be using the package UNMARKED. UNMARKED provides functions for lots of different occupancy and occupancy-type models. We will be using it throughout the semester, but we will also be using other methods too so that we get experience using multiple methods.

1. Naïve occupancy estimates

Download the script named `exercise_3_functions.R` into your working directory. Source this code using the command: `source("Exercise3Functions.R")`[1]. This script contains functions that you will use in this exercise to make certain calculations. Feel free to open this code and try to figure out what these functions do (and ask if you have questions).

For this lab, you will be supplementing last week's dataset of Yellow-rumped Warblers (`yrwa.csv`) with two additional species: Fox Sparrow (`fosp.csv`) and Brown Creeper (`brcr.csv`). We will be comparing the occupancy and detectability of each seperately. Take a moment now to load these three files into R.

1.1. *Calculate the naïve occupancy separately for each of the three surveys for each species. Then take the average of these estimates. This average should give you an estimate of what naïve occupancy would be if we had just completed one survey at each site.*

1.2. *Calculate the naïve occupancy for each species after summarizing the detection/non-detection data over the three surveys. Record these estimates in a table in your notebook. We will later compare these estimates to the estimates you get from an occupancy model implemented in unmarked.*

[1] 'source' means that R will load these functions into your session, and you can now refer to the functions and run them. It's like loading a package, except that the package is an R script.

1.3. *Estimate the detection probability using all of the detection histories. This can be done by first estimating the detection probability at each site*[2] *and then taking the mean across all sites.*

1.4. *Estimate the detection probability using only the detection histories where the species was detected at least once. You can use code similar to the following to remove the rows of data where the species was not detected on any of the three surveys*: `data[apply(data, 1, sum) > 0,]`[3]. *Record these estimates in your table.*

How do your estimates of p differ among the species and the methods?

2. *Using* unmarked

Now that we have estimates occupancy and detectability using this informal method, we will get the 'official' estimates using commands in unmarked. If you have not already done so, please install the unmarked package and load it into your session[4]. In order to get unmarked to work with your data, you will need to load your occurrence data in as a special class of object that unmarked will recognize as occurrence data[5].

2.1. *Using the Yellow-rumped Warbler dataset, create an object called* `yrwa.un` *that is an* unmarked *dataframe using the command* `unmarkedFrameOccu()`. *Use* `?unmarkedFrameOccu` *if you need help.* Examine this object after creating it.

2.2. *Plot your unmarked object.* What does this show?

Now that you have your data loaded into an object that unmarked will understand, it's time to run your first model. We will be using a single-species single-season model, which is the function `occu()`.

2.3. *Read through the help file for* `occu()`.

The main thing you have to parameterize for `occu()` is the formula. If you've used `lm()` or `glm()` or `lmer()` in other applications, you will be familiar with writing formulas that look something like this:

```
response ~ var1 + var2
```

This is the typical format for writing a linear formula for a single response variable with two covariates. Implicit in this formula is also the intercept—you do not have to put it in there, but R will interpret your formula as:

[2] Hint: get a count of the number of detections at each site using the apply function then divide this number by the total number of surveys at each site.

[3] Where `data` is whatever you have named your objects for each species.

[4] `install.packages("unmarked")` then `library(unmarked)`

[5] In future exercises, this unmarked data object will also contain your covariates, but for now, we only have occurrence data

```
response ~ 1 + var1 + var2
```

If we did not have any covariates (like in our exercise today), then we would only need the intercept, and we would have to put it into our formula.

2.4. *Using your Yellow-rumped Warbler unmarked dataset object, run* `occu()`*, storing the model as the object* `yrwa.occ`*. Note that unlike for* `glm()` *or other simpler linear models, here we have two response variables that we are parameterizing,* p *and ψ. You will have to specify 2 formula in your* `occu()` *command, as specified in the help file.*

2.5. *Examine your occupancy model object* `yrwa.occ`*. Note that it gives you an AIC score as well as parameter estimates.*

WHAT ARE EACH OF THESE ESTIMATES? WHAT IS THEIR SCALE? HOW CAN OCCUPANCY BE NEGATIVE?

2.6. *Repeat the previous steps for the other two datasets.*

3. *Interpreting occupancy model estimates*

You will have noticed that your parameter estimates are on a continuous scale and can be positive or negative. That is because model parameters—for both p and ψ—are modeled on the logit scale[6]. In order to interpret them properly, you will have to back-transform them. This can be done in multiple ways. First, remember the equation for the logit link[7]:

$$\text{logit}(\mu) = \log\left(\frac{\mu}{1-\mu}\right) = \beta_0$$

Your model has calculated β_0 in the above equation. You will have to inverse the equation in order to solve for μ:

$$\mu = \text{InverseLogit}\left(\beta\right) = \frac{\exp(\beta_0)}{\exp(\beta_0)+1}$$

3.1. *Using this equation for inverse logit, calculate the point estimates for* p *and ψ for the Yellow-rumped Warbler.*[8]

We can make this a bit simpler by using the function `backTransform()`.

3.2. *Use* `backTransform()` *to estimate* p *and ψ for all three species. Record the estimates in your table.*

HOW DO THE ESTIMATES OF p AND ψ DIFFER FROM THE ESTIMATES YOU OBTAINED ABOVE?

[6] Think back to the lecture on GLM models. The logit link is used to allow linear modeling for variables that range between 0 and 1.

[7] Assuming there is only an intercept.

[8] You can extra the coefficients 'by hand' from the occupancy model or use the `coef()` function.

3.3. *Using* `confint()` *on your back-transformed estimates, obtain confidence intervals for* p *and* ψ.

DO THE CONFIDENCE INTERVALS CONTAIN THE NAIVE ESTIMATES OF PSI OR P?

WHY ARE ESTIMATES OF p SO CLOSE TO THE ESTIMATES YOU OBTAINED WHEN YOU JUST EXCLUDED THE {0 0 0} DETECTION HISTORIES?

3.4. *Use the* `z.test()` *function*[9] *to conduct pairwise tests of the occupancy and detection probability estimates among species*[10].

The `z.test()` function takes four parameters:

(1) the estimate (p or ψ) from the first species (est1)

(2) the SE of the estimate form the first species (SE1)

(3) the estimate from the second species (est2), and

(4) the SE of the estimate from the second species (SE2).

The function returns the Z-statistic, which is calculated as:

$$\frac{est1 - est2}{\sqrt{SE1^2 + SE2^2}}$$

If the p-value from th Z-statistic is >0.05 we will consider the two estimates different from one another. This is the standard way to estimate whether two coefficients from a model are different from one another.

HOW DO DETECTION PROBABILITIES AND OCCURRENCE PROBABILITIES DIFFER AMONG THE THREE SPECIES?

4. *Assessing model fit*

We have fit 3 very simple models to these data. By having no covariates at all, the models assume that all sites have an equal probability of being occupied, and all surveys have an equal probability of detecting a species if it's present. It's not hard to imagine that these simple models probably are not going to capture the heterogeneity of the data very well. Just to be sure, however, we can use the function `parboot()` to check the fit of our three models.

`parboot()` simulates datasets that arise from modeled parameters (i.e., with the same probability of detection and occupancy that you have modeled). This is very similar to the random datasets we simulated last week using `rbinom()` and a given value of p, except now we have p and ψ. After simulating these datasets, parb will refit the model, and use test statistics to check whether the model is a

[9] In the code you sourced in step 1.

[10] Again, either extract the coefficiens and standard errors from the model or use `coef()` and `SE()`.

good fit. If the model is a good fit, then re-fitted parameter estimates should be no different than the original parameters. Here, we will evaluate this using a χ^2 test[11].

4.1. *Use* `parboot` *and the* `chisq` *function (from the code sourced in step 1) to test whether each occupancy model provides a reasonable fit to the data. A p-value >0.05 suggests that data generated from the model are significantly different than the observed data.*

Does the model fit the data for each species?

Does this mean that this model is a good model for the species?

Why might the model be too simple for this data?

What factors might we be leaving out or what assumptions might we be making that are invalid.

[11] There are lots of ways to evaluate the fit of occupancy models and no obvious standards. We will use this one today, but realize that this is a potentially arbitrary decision.