Bolas\_Hwk6\_Occupancy

Ellie Bolas

May 10, 2018

# Occupancy Modeling

Data for this homework assignment are located on Canvas, in the file “flycatcher data.csv”. Species detection/non-detection data (columns “y.1” to “y.3”) were collected on American fly catchers at 50 sites, each visited 3 times. For each site, the amount of woody habitat (as a percentage) was recorded (column “woody” in the data frame). For each survey, the Julian date (day of the year, where 0 = January 1) was also collected (columns “date.1” to “date.3” in the data frame).

## Task 1: Reading in and formatting data (5 Pts)

Read the data into R and use them to create an unmarkedFrameOccu object with detections, site covariates and observation covariates. Tip: Site covariates are those that have a single value for each site. Observation covariates are those that have a value for each site and visit.

library(unmarked)

## Warning: package 'unmarked' was built under R version 3.3.3

## Loading required package: reshape

## Warning: package 'reshape' was built under R version 3.3.3

## Loading required package: lattice

## Loading required package: parallel

## Loading required package: Rcpp

getwd()

## [1] "C:/Users/ebola/Google Drive/Git/WFC198\_Git/WFC198/Lab6\_Hwk6\_Occupancy"

dat <- read.csv("Flycatcher\_data.csv")  
  
#need 3 arguments  
#det df is the detection/non-detection data by site  
det <- data.frame(dat[, c("y.1", "y.2", "y.3")])  
  
#siteCovs is any covariates that vary by site (usually habiat stuff)  
site\_covs <- data.frame(woody = dat$woody) #need to give the column head or else it returns dat.woody for column head  
  
#obsCovs is covariates that vary within sites by visit, have to make this as a list for some reason, making it as a df made the column headers strange  
obs\_covs <- list(date = dat[, c("date.1", "date.2", "date.3")])   
head(obs\_covs[[1]])

## date.1 date.2 date.3  
## 1 6 25 34  
## 2 20 32 54  
## 3 20 32 47  
## 4 20 32 47  
## 5 8 27 36  
## 6 8 27 36

umffly <- unmarkedFrameOccu(y=det, siteCovs = site\_covs, obsCovs = obs\_covs)  
summary(umffly)

## unmarkedFrame Object  
##   
## 50 sites  
## Maximum number of observations per site: 3   
## Mean number of observations per site: 3   
## Sites with at least one detection: 34   
##   
## Tabulation of y observations:  
## 0 1 <NA>   
## 85 65 0   
##   
## Site-level covariates:  
## woody   
## Min. : 0.0   
## 1st Qu.:20.0   
## Median :30.0   
## Mean :32.1   
## 3rd Qu.:45.0   
## Max. :80.0   
##   
## Observation-level covariates:  
## date   
## Min. : 1.00   
## 1st Qu.:11.00   
## Median :27.00   
## Mean :24.83   
## 3rd Qu.:34.00   
## Max. :54.00

## Task 2: Raw occupancy (5 Points)

Calculate raw occupancy (percentage of sites where species was observed at least once) – make sure your “knitted” report contains both the R code and the value for raw occupancy.

#naive occupancy: # sites with at least one detection /total number of sites, ignores imperfect detection  
  
#two ways to do this:  
  
#1)  
summary(umffly)

## unmarkedFrame Object  
##   
## 50 sites  
## Maximum number of observations per site: 3   
## Mean number of observations per site: 3   
## Sites with at least one detection: 34   
##   
## Tabulation of y observations:  
## 0 1 <NA>   
## 85 65 0   
##   
## Site-level covariates:  
## woody   
## Min. : 0.0   
## 1st Qu.:20.0   
## Median :30.0   
## Mean :32.1   
## 3rd Qu.:45.0   
## Max. :80.0   
##   
## Observation-level covariates:  
## date   
## Min. : 1.00   
## 1st Qu.:11.00   
## Median :27.00   
## Mean :24.83   
## 3rd Qu.:34.00   
## Max. :54.00

#output includes: Sites with at least one detection: 34   
rawoccu <-34/50  
rawoccu

## [1] 0.68

#doing this with R code as in lab  
# apply(X, MARGIN, FUN)  
#x = det df, margin = how many rows at a time, fun= sum bc each row is being summed  
totalobs <- apply(det, 1, sum, na.rm = T)  
totalobs #gives you a vector of numbers (0,1,2,3) each element of the vector is the sum of the 3 survey occasions for that row of whether you detected the species. The range is 0:3 because you can either detect 0,1,2,3 times

## [1] 2 2 2 2 0 3 0 0 2 0 1 2 2 2 2 1 0 0 2 2 0 3 0 1 1 3 3 2 0 2 0 2 1 0 2  
## [36] 0 0 0 2 3 1 1 0 0 2 1 2 2 3 1

detections <-length(which(totalobs >0)) #use length instead of sum because it's an index, tells you how many sites had at least one detection  
  
rawoccu2 <- detections/(length(totalobs))  
rawoccu2

## [1] 0.68

#Both methods return 0.68, 68% raw occupancy

# Task 3: Running occupancy models (10 Pts)

1. Run all possible models (meaning, models representing all possible covariate combinations). Tip 1: there are 8 models total. Tip 2: Remember that site covariates can be used for both the detection and occupancy components of the model, whereas observation covariates can only be used for the detection component. (8 Pts) Note: Producing summaries for some of the models will result in a warning message. You can ignore that message.

occu(formula, data) formula= detectionoccupancy data= umffly

#intercept-only/null model  
mod0 <- occu(~1~1, data= umffly)  
summary(mod0)

##   
## Call:  
## occu(formula = ~1 ~ 1, data = umffly)  
##   
## Occupancy (logit-scale):  
## Estimate SE z P(>|z|)  
## 0.987 0.374 2.64 0.0083  
##   
## Detection (logit-scale):  
## Estimate SE z P(>|z|)  
## 0.384 0.235 1.63 0.102  
##   
## AIC: 196.3941   
## Number of sites: 50  
## optim convergence code: 0  
## optim iterations: 25   
## Bootstrap iterations: 0

#woody cover on detection  
modwoodydet <- occu(~woody~1, data = umffly)  
summary(modwoodydet)

##   
## Call:  
## occu(formula = ~woody ~ 1, data = umffly)

## Warning in sqrt(diag(vcov(obj))): NaNs produced

## Occupancy (logit-scale):  
## Estimate SE z P(>|z|)  
## 10.3 NaN NaN NaN  
##   
## Detection (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) -1.8949 0.3907 -4.85 1.24e-06  
## woody 0.0495 0.0104 4.75 2.03e-06  
##   
## AIC: 182.7696   
## Number of sites: 50  
## optim convergence code: 0  
## optim iterations: 23   
## Bootstrap iterations: 0

#woody cover on occupancy  
modwoodyoccu <- occu(~1~woody, data = umffly)  
summary(modwoodyoccu)

##   
## Call:  
## occu(formula = ~1 ~ woody, data = umffly)

## Warning in sqrt(diag(vcov(obj))): NaNs produced

## Occupancy (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) -7.73 36.9 -0.209 0.834  
## woody 4.59 NaN NaN NaN  
##   
## Detection (logit-scale):  
## Estimate SE z P(>|z|)  
## -0.195 0.167 -1.16 0.244  
##   
## AIC: 204.2646   
## Number of sites: 50  
## optim convergence code: 0  
## optim iterations: 37   
## Bootstrap iterations: 0

#woody cover on both  
modwoody <- occu(~woody~woody, data = umffly)  
summary(modwoody)

##   
## Call:  
## occu(formula = ~woody ~ woody, data = umffly)  
##   
## Occupancy (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) -1.2156 0.9707 -1.25 0.2105  
## woody 0.0932 0.0411 2.27 0.0232  
##   
## Detection (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) -0.7197 0.6006 -1.20 0.2308  
## woody 0.0279 0.0136 2.05 0.0399  
##   
## AIC: 179.2103   
## Number of sites: 50  
## optim convergence code: 0  
## optim iterations: 59   
## Bootstrap iterations: 0

#date on det  
moddatedet <- occu(~date~1, data = umffly)  
summary(moddatedet)

##   
## Call:  
## occu(formula = ~date ~ 1, data = umffly)  
##   
## Occupancy (logit-scale):  
## Estimate SE z P(>|z|)  
## 0.89 0.34 2.62 0.00888  
##   
## Detection (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) 2.0402 0.5903 3.46 0.000548  
## date -0.0602 0.0186 -3.24 0.001194  
##   
## AIC: 185.6511   
## Number of sites: 50  
## optim convergence code: 0  
## optim iterations: 22   
## Bootstrap iterations: 0

#date on det, woody on occu  
moddatewoody <- occu(~date~woody, data = umffly)  
summary(moddatewoody)

##   
## Call:  
## occu(formula = ~date ~ woody, data = umffly)  
##   
## Occupancy (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) -1.715 0.820 -2.09 0.03637  
## woody 0.102 0.035 2.91 0.00356  
##   
## Detection (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) 2.1104 0.5739 3.68 0.000236  
## date -0.0624 0.0185 -3.38 0.000720  
##   
## AIC: 170.2177   
## Number of sites: 50  
## optim convergence code: 0  
## optim iterations: 50   
## Bootstrap iterations: 0

#date and woody on det  
moddatewoodydet <- occu(~date+woody~1, data = umffly)  
summary(moddatewoodydet)

##   
## Call:  
## occu(formula = ~date + woody ~ 1, data = umffly)  
##   
## Occupancy (logit-scale):  
## Estimate SE z P(>|z|)  
## 2.39 1.25 1.92 0.055  
##   
## Detection (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) -0.7050 0.6481 -1.09 2.77e-01  
## date -0.0440 0.0159 -2.76 5.72e-03  
## woody 0.0512 0.0121 4.23 2.31e-05  
##   
## AIC: 175.6772   
## Number of sites: 50  
## optim convergence code: 0  
## optim iterations: 30   
## Bootstrap iterations: 0

#date and woody on det and woody on occu  
modall <- occu(~date+woody~woody, data = umffly)  
summary(modall)

##   
## Call:  
## occu(formula = ~date + woody ~ woody, data = umffly)  
##   
## Occupancy (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) -1.4926 0.8665 -1.72 0.08499  
## woody 0.0966 0.0364 2.66 0.00792  
##   
## Detection (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) 1.0642 0.8315 1.28 0.20057  
## date -0.0590 0.0186 -3.17 0.00151  
## woody 0.0236 0.0138 1.70 0.08861  
##   
## AIC: 169.0779   
## Number of sites: 50  
## optim convergence code: 0  
## optim iterations: 36   
## Bootstrap iterations: 0

1. Compare all models based on their AIC using the appropriate R commands (make sure that your “knitted” report shows the model selection table) (2 Pt) Note: Producing the model selection table will cause a warning message. You can ignore that message.

fitlist <- fitList(mod0, modwoodydet, modwoodyoccu, modwoody, moddatedet, moddatewoody, moddatewoodydet, modall) #we always get that warning, just ignore

## Warning in fitList(mod0, modwoodydet, modwoodyoccu, modwoody, moddatedet, :  
## Your list was unnamed, so model names were added as object names

modselection <- modSel(fitlist)

## Warning in sqrt(diag(vcov(x, altNames = TRUE))): NaNs produced

## Warning in sqrt(diag(vcov(x, altNames = TRUE))): NaNs produced

modselection

## nPars AIC delta AICwt cumltvWt  
## modall 5 169.08 0.00 6.2e-01 0.62  
## moddatewoody 4 170.22 1.14 3.5e-01 0.97  
## moddatewoodydet 4 175.68 6.60 2.3e-02 1.00  
## modwoody 4 179.21 10.13 3.9e-03 1.00  
## modwoodydet 3 182.77 13.69 6.6e-04 1.00  
## moddatedet 3 185.65 16.57 1.6e-04 1.00  
## mod0 2 196.39 27.32 7.3e-07 1.00  
## modwoodyoccu 3 204.26 35.19 1.4e-08 1.00

#modall is the top model and moddatewoody is within 2

# Task 4: Plotting covariate relationships (5 Pts)

Plot the relationship of occupancy with the covariate in the top model from Task 3 c. In your plot, use the “ylim” argument to set the range of your y axis from 0 to 1. Label your axes. Tip: To make this plot, first create a sequence of possible values for the covariate (be careful: the covariate is a percentage so can only take on certain values); then use the predict() function to calculate expected values of occupancy for these new covariate values. ATTENTION: The predict() function does not always work – that seems to depend on the version of unmarked that you have installed on your computer. If it does not work, manually calculate the predicted values of occupancy probability, psi, based on the parameter estimates from the top model from Task 3c, and plot those values (remember to back-transform) against the new covariate values. Code for both options is in the lab R script.

summary(modall)

##   
## Call:  
## occu(formula = ~date + woody ~ woody, data = umffly)  
##   
## Occupancy (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) -1.4926 0.8665 -1.72 0.08499  
## woody 0.0966 0.0364 2.66 0.00792  
##   
## Detection (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) 1.0642 0.8315 1.28 0.20057  
## date -0.0590 0.0186 -3.17 0.00151  
## woody 0.0236 0.0138 1.70 0.08861  
##   
## AIC: 169.0779   
## Number of sites: 50  
## optim convergence code: 0  
## optim iterations: 36   
## Bootstrap iterations: 0

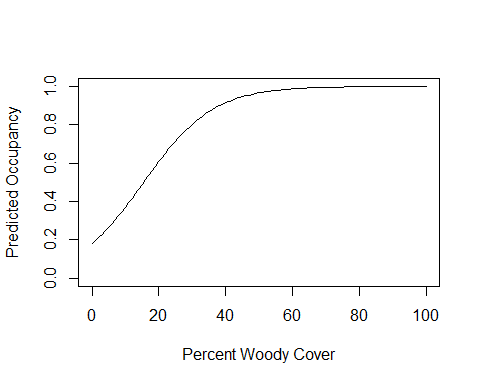
#occupancy influenced by woody cover which is a percentage, first making a new cov for woody with all possible percentages. the percentages are as whole numbers  
woodynew<-data.frame(woody = seq(0,100,1))  
head(woodynew)

## woody  
## 1 0  
## 2 1  
## 3 2  
## 4 3  
## 5 4  
## 6 5

#use the predict() function to calculate expected values of occupancy for these new covariate values  
  
occuwoody <- predict(modall, newdata = woodynew, "state")  
head(occuwoody)

## Predicted SE lower upper  
## 1 0.1835384 0.1298538 0.03950745 0.5512802  
## 2 0.1984571 0.1330204 0.04587204 0.5604556  
## 3 0.2142703 0.1357660 0.05315572 0.5698305  
## 4 0.2309803 0.1380440 0.06145817 0.5794224  
## 5 0.2485813 0.1398141 0.07087961 0.5892508  
## 6 0.2670577 0.1410444 0.08151718 0.5993373

#plot x by y  
plot(woodynew$woody,occuwoody$Predicted, type = "l", xlab = "Percent Woody Cover", ylab = "Predicted Occupancy", ylim = c(0,1))



# Task 5: Interpreting model output (10 Pts)

Look at the summary results of the top model from Task 3c (make sure the summary output is included in your “knitted” report) to address the following tasks. Add answers into your R script but make sure to comment them out.

summary(modall)

##   
## Call:  
## occu(formula = ~date + woody ~ woody, data = umffly)  
##   
## Occupancy (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) -1.4926 0.8665 -1.72 0.08499  
## woody 0.0966 0.0364 2.66 0.00792  
##   
## Detection (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) 1.0642 0.8315 1.28 0.20057  
## date -0.0590 0.0186 -3.17 0.00151  
## woody 0.0236 0.0138 1.70 0.08861  
##   
## AIC: 169.0779   
## Number of sites: 50  
## optim convergence code: 0  
## optim iterations: 36   
## Bootstrap iterations: 0

modall\_summary <- summary(modall)

##   
## Call:  
## occu(formula = ~date + woody ~ woody, data = umffly)  
##   
## Occupancy (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) -1.4926 0.8665 -1.72 0.08499  
## woody 0.0966 0.0364 2.66 0.00792  
##   
## Detection (logit-scale):  
## Estimate SE z P(>|z|)  
## (Intercept) 1.0642 0.8315 1.28 0.20057  
## date -0.0590 0.0186 -3.17 0.00151  
## woody 0.0236 0.0138 1.70 0.08861  
##   
## AIC: 169.0779   
## Number of sites: 50  
## optim convergence code: 0  
## optim iterations: 36   
## Bootstrap iterations: 0

modall\_summary

## $state  
## Estimate SE z P(>|z|)  
## (Intercept) -1.49255622 0.86654689 -1.722418 0.084993770  
## woody 0.09659077 0.03637398 2.655491 0.007919301  
##   
## $det  
## Estimate SE z P(>|z|)  
## (Intercept) 1.06422218 0.83147112 1.279927 0.200570878  
## date -0.05902398 0.01860079 -3.173197 0.001507701  
## woody 0.02356388 0.01383871 1.702751 0.088614592

1. In words, describe the relationship(s) between occupancy probability and the covariate(s) in the top model. Your statement should contain the appropriate number(s) form the model output, as well as appropriate units. Your statement should also address whether the relationship(s) are statistically significant, and on which number in the output your answer is based. (4 Pts)

*Answer* When there is 0% woody cover, the predicted occupancy is the logit(-1.4926 +/- 0.8665) (the intercept of the output for the occupancy model). This is not statistically significant, because the p value is greater than 0.05 (p=0.8499). For every 1% increase in woody cover, there is a logit(0.0966 +/- 0.0364) increase in predicted occupancy of flycatchers at a site. This is based on the slope in the occupancy model. Also, this relationship is statistically significcant because p = 0.00792, less than 0.05.

1. In words, describe the relationship(s) between detection probability and the covariate(s) in the top model. Your statement should contain the appropriate number(s) form the model output, as well as appropriate units. Your statement should also address whether the relationship(s) are statistically significant, and on which number in the output your answer is based. (6 Pts)

*Answer* When there is 0% woody cover and the survey day = 0 (Jan. 1), the detection probability is logit(1.0642 +/- 0.8315) (the intercept of the detection model). This is not statistically significant because p = 0.20057, which is greater than 0.05. For every 1 day past Jan 1st, there is a logit(-0.0590 +/- 0.0186) decrease in detection probability for flycatchers (the first slope listed in the detection model). This relationship is statistically significant because p = 0.00151. For every 1% increase in wood cover, detection probability of flycatchers inreases by logit(0.0236 +/- 0.0138) (the second slope in the detection model.) However, this relationship is not statistically significant because p = 0.08861.