SimulatedAnalysis

Reed Scott

2025-02-06

# Introduction

Hello Brittany!

I have been working on some simulated analysis for VTADs. The goal was to simulate data similar to what we’re trying to do with VTADs, so that when we finally get results we can almost immediately plug and play. This does require some discussion of what our hypotheses will be, which I discuss in an earlier doc I sent (Hypotheses.docx). I think that that will help inform the full model set we will build for this project but for right now I wanted to create a simulated dataset and compare a few simple models.

I have choosen to simulate a single season occupancy model, and I have choosen to work in unmarked. My reasoning for a single season is that for our real data, while we have multiple seasons it’s only two years and seperated by a missing year of data. So, I am suggesting we run 2 single season models. I want to work in unmarked because 1) I think it’s good professional development to learn the multiple ways of doing ecological modeling and 2) I think unmarked is a bit easier to work with for smaller model sets.

With all this in mind, let’s look at the simulation I did

# Building the simulated data

The first thing I wanted to do was set the parameters that would define our data. In this simulation I chose to use 24 ponds as our number of sites. I simulated 3 visits, and 3 tests. However, unmarked couldn’t handle 108 tests (12 \* 3 \* 3) for simulation purposes, so instead of having 12 frogs per survey I only had 2 (so 18 total tests). With these parameters in place we can form our detection matrix, which ends up being 24x18 like we would expect.

#Initialize -----------------------------------------  
rm(list=ls())  
library(unmarked)  
library(MuMIn)  
library(ggplot2)  
# Load functions--------------------------------------  
repo <- getwd()  
graphs <- paste(repo, "/Graphs", sep = "")  
  
# Global Variables-------------------------------------  
  
  
# Program Body------------------------------------------  
# Initialize variables  
set.seed(1234)  
ponds <- 24 #number of sites  
frogs <- 2 #number of frogs PER VISIT  
visits <- 3 #number of visits to each site  
tests <- 3 #number of tests per frog  
total\_frogs <- frogs\*visits  
total\_tests <- frogs\*visits\*tests #total number of tests for each pond for the season  
  
# Create an empty dataframe for the detection data  
det <- matrix(NA, nrow = ponds, ncol = total\_tests)

Now the goal is to populate this empty dataframe based on covariates that effect , , and p. To do this in unmarked you need to create seperate objects for the site covariates, availability covariates, and observation covariates. I am going to spare you the code for the moment, but I did want to take this opporunity to reiterate my hypotheses at the moment. I have hypotheses that will be negatively correlated with and , and positively correlated with . For I believe the relationships with , , and to be the same, but adding in an effect of Host Species ID. The hypotheses for p are all the same as for .

With the covariates simulated, we’re now ready to create and unmarkedFrame object and simulate detection data based on our covariates. In this example, I simulated a negative effect of on and , and an effect of on p.

umf <- unmarkedFrameGOccu(y = det, siteCovs = site\_covs, obsCovs = test\_covs, yearlySiteCovs = avail\_covs,   
 numPrimary = total\_frogs)  
coeff <- list(psi = c(0,-0.7),   
 phi = c(0,-0.9),   
 det = c(0,-0.75))  
sim2 <- simulate(umf, model = goccu, psiformula = ~betadiv, phiformula = ~betadiv, pformula =~ alpha, coefs = coeff)

And just like that we’ve simulated some data! But does it work? We can check using the goccu function. In a simple example below, I fit a null model and three models for the variables effecting . If the simulation worked correctly, then psi.beta should be the top model.

#null model  
null.mod <- goccu(psiformula = ~1, phiformula = ~1, pformula = ~1,data = sim2[[1]])  
#Models for psi  
psi.beta <- goccu(psiformula = ~betadiv, phiformula = ~1, pformula = ~1, data = sim2[[1]])  
psi.alpha <- goccu(psiformula = ~alpha, phiformula = ~1, pformula =~ 1, data = sim2[[1]])  
  
fitlist.psi <- fitList("nullpsi" = null.mod,  
 "betapsi" = psi.beta,  
 "alphapsi" = psi.alpha)  
  
modSel(fitlist.psi)

## nPars AIC delta AICwt cumltvWt  
## betapsi 4 338.40 0.00 0.53 0.53  
## nullpsi 3 339.38 0.98 0.32 0.85  
## alphapsi 4 340.89 2.49 0.15 1.00

And it is!

We can check the same thing for , and this should show that the top model includes an affect of species:

theta.beta <- goccu(psiformula = ~1, phiformula = ~betadiv, pformula = ~1, data = sim2[[1]])  
theta.alpha <- goccu(psiformula = ~1, phiformula = ~alpha, pformula = ~1, data = sim2[[1]])  
  
fitlist.theta <- fitList("null" = null.mod,  
 "beta" = theta.beta,  
 "alpha" = theta.alpha)  
modSel(fitlist.theta)

## nPars AIC delta AICwt cumltvWt  
## beta 4 328.55 0.00 0.9677 0.97  
## alpha 4 335.64 7.08 0.0280 1.00  
## null 3 339.38 10.83 0.0043 1.00

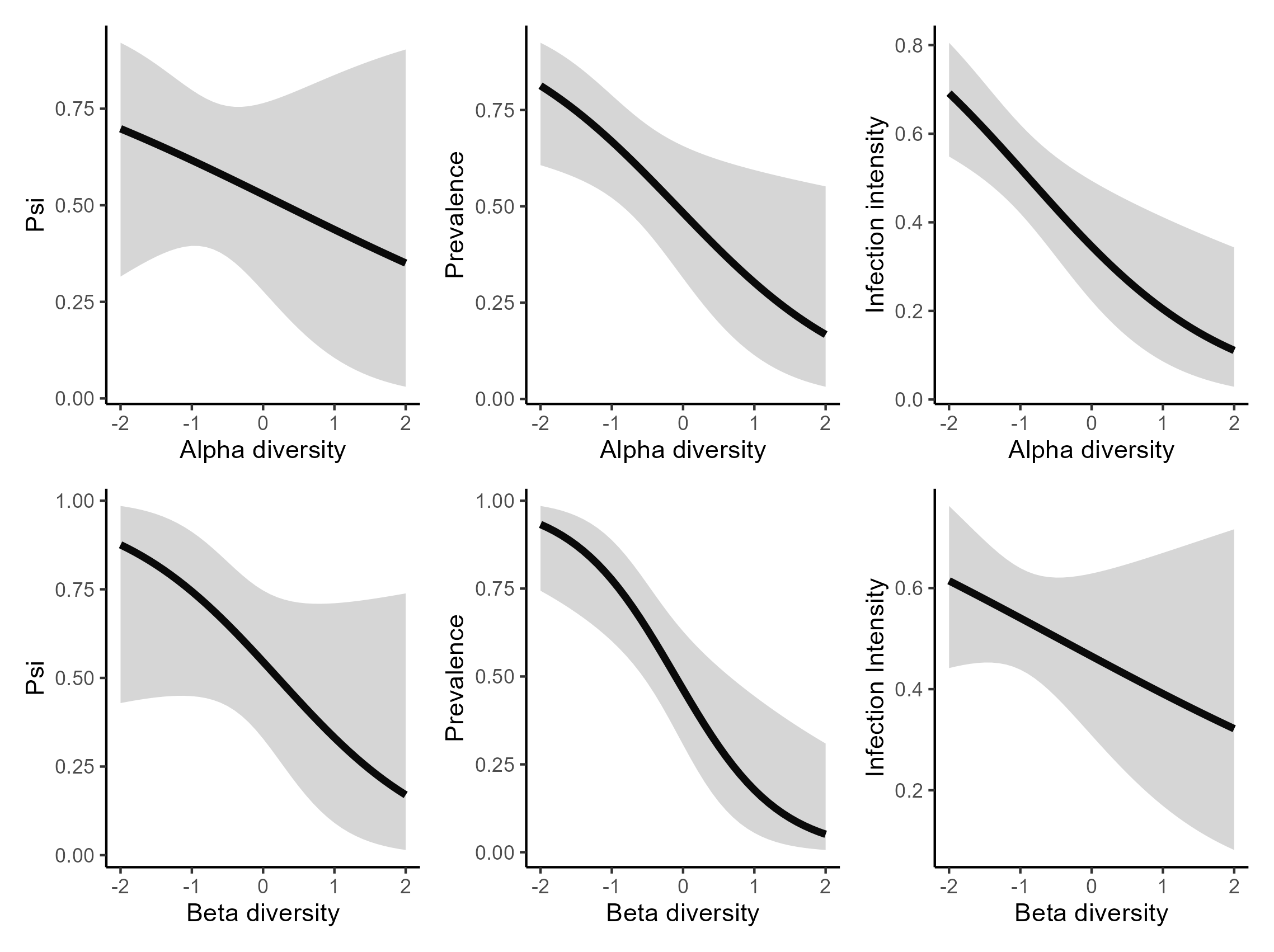
This indicates that the simulation worked and the models are behaving as expected. But let’s check p just to be sure

p.beta <- goccu(psiformula = ~1, phiformula = ~1, pformula = ~betadiv, data = sim2[[1]])  
p.alpha <- goccu(psiformula = ~1, phiformula = ~1, pformula = ~alpha, data = sim2[[1]])  
  
fitlist.theta <- fitList("null" = null.mod,  
 "beta" = p.beta,  
 "alpha" = p.alpha)  
modSel(fitlist.theta)

## nPars AIC delta AICwt cumltvWt  
## alpha 4 332.26 0.00 0.955 0.95  
## null 3 339.38 7.12 0.027 0.98  
## beta 4 340.20 7.94 0.018 1.00

# Graphing

I have put more thought in to what plots I would like to include in this project. I envision the final paper having two main figures, each with 6 panels: 1 figure for RV results and 1 figure for BD. Each panel will show the effect of or on , , or . Additionally, each plot will have two lines: one for the 2022 data and one for the 2024 data. I have included an example figure to show what I mean (although this only includes 1 line, not the 2 years of data).



# Conclusion + next steps

I have determined that, as of right now, we have 27 possible models to explore. I have provided a few short + easy examples of how this can be implemented in unmarked. I now have a vision for structure I would like for our results section. I think at this point I could pretty easily write out a script to have ready to go. I think the next big step is just getting the data.