Logistic Regression Exercise Part 1

We’ll start by simulating some presence/absence data (or rather, detection/non-detection data).

**What we’re doing here is simulating occupancy of the species and then “sampling” from the simiulated system.First we specify the number of sites, M, and the number of replicate survey occasions, J. Don’t worry about the details now; we’ll get to that later.**

M<-500 # Number of sites  
J<-3 # Number of repeat surveys per site

Now we just pick some values for occupancy and detection parameters (slopes and intercepts).

**Here, a = parameters associated with occupancy and b = parameters associated with detection. Note that these are on the logit scale.**

a0<-0.25 # True occupancy intercept  
a1<-2 # True occupancy slope for site level covariate  
b0<-0.25 # True detection intercept   
b1<-0.5 # True detection slope for site-level covariate   
b2<--0.50 # True detection slope for survey-specific covariate

Now we’re going to create a site-level covariate, X1, that influences occupancy.

**We then calculate occupancy probabilities (psi) and true occurrence (1 or 0) for each of the M simulated sites.**

X1<-runif(M,-1,1) # Occupancy covariate, ranges from -1 to 1 (standardized)  
psi<-plogis(a0 + a1\*X1) # True occupancy probability  
z<-rbinom(M,1,psi) # True (latent) occurrence (z = 1 is present; z = 0 is absent)  
table(z)

## z  
## 0 1   
## 234 266

Now create 3 empty matrices that we’ll use in the next steps.

y<-array(NA,dim=c(M,J))  
X2<-array(NA, dim=c(M,J)) # Detection covariate matrix  
p<-array(NA, dim = c(M,J)) # Detection probability matrix

Now we “sample” each site

**Note that our ability to observe the species is corrupted by incomplete detection of the species during sampling (p at each site).**

for (i in 1:M){  
 for (j in 1:J){  
 X2[i,j]<-runif(1,-1,1)  
 p[i,j]<-plogis(b0 + b1\*X1[i] + b2\*X2[i,j])  
 y[i,j]<-rbinom(1,1,p[i,j]\*z[i]) # Survey data matrix  
 }  
}  
head(y)

## [,1] [,2] [,3]  
## [1,] 1 1 0  
## [2,] 1 1 1  
## [3,] 0 0 0  
## [4,] 0 1 0  
## [5,] 1 1 1  
## [6,] 1 0 0

Now extract sample data (y) for each occasion.

firsty<-y[,1] # extract first occasion's data  
secondy<-y[,2] # second occasion's data  
thirdy<-y[,3] # third occasion's data  
maxy<-apply(y,1,max) # max across all survey occasions for each site (i.e., was it ever detected)

Then create a new data frame for making predictions later.

newX1<-data.frame(X1=seq(-1,1, length.out = 250))

Now fit a logistic regression model to each occasion’s data.

# Fit model to occasion 1's data, then make predictions from the model  
summary(first<-glm(firsty~ X1, family = binomial))

##   
## Call:  
## glm(formula = firsty ~ X1, family = binomial)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.5915 -0.7732 -0.4351 0.9322 2.3438   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.065 0.119 -8.947 <2e-16 \*\*\*  
## X1 2.016 0.224 8.998 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 607.44 on 499 degrees of freedom  
## Residual deviance: 501.47 on 498 degrees of freedom  
## AIC: 505.47  
##   
## Number of Fisher Scoring iterations: 4

predfirst<-predict(first,type = "response", newdata = newX1)  
predfirst<-cbind(predfirst,newX1)  
  
# Fit model to occasion 2's data, then make predictions from the model  
summary(second<-glm(secondy~ X1, family = binomial))

##   
## Call:  
## glm(formula = secondy ~ X1, family = binomial)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.5004 -0.8279 -0.5029 1.0282 2.2826   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.9198 0.1102 -8.347 < 2e-16 \*\*\*  
## X1 1.6661 0.2045 8.146 3.77e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 619.10 on 499 degrees of freedom  
## Residual deviance: 538.69 on 498 degrees of freedom  
## AIC: 542.69  
##   
## Number of Fisher Scoring iterations: 4

predsecond<-predict(second,type = "response", newdata = newX1)  
predsecond<-cbind(predsecond,newX1)  
  
# Fit model to occasion 3's data, then make predictions from the model  
summary(third<-glm(thirdy~ X1, family = binomial))

##   
## Call:  
## glm(formula = thirdy ~ X1, family = binomial)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.4893 -0.8384 -0.5341 1.0713 2.2071   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.8386 0.1070 -7.841 4.48e-15 \*\*\*  
## X1 1.5594 0.1980 7.877 3.36e-15 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 628.37 on 499 degrees of freedom  
## Residual deviance: 554.83 on 498 degrees of freedom  
## AIC: 558.83  
##   
## Number of Fisher Scoring iterations: 4

predthird<-predict(third,type = "response", newdata = newX1)  
predthird<-cbind(predthird,newX1)

Finally, plot and compare the predicted relationship between occupancy and covariate X1 based on parameter estimates from each logistic regression model.

plot(X1, maxy, xlab = "X1", ylab = "Probability of Occurrence", frame = F, cex = 1.5) # Plot raw data  
lines(predfirst$X1, predfirst$predfirst, col = "red", type = "l", lwd = 2)  
lines(predsecond$X1, predsecond$predsecond, col = "green", type = "l", lwd = 2)  
lines(predthird$X1, predthird$predthird, col = "orange", type = "l", lwd = 2)  
legend(-1, 0.95, c("First","Second","Third"), lty = c(1, 1,   
 1, 1), lwd = c(2.5,2.5,2.5,2.5,2.5,2.5), col = c("red", "green", "orange"))

