caphistory

Bethany Liberto

January 10, 2019

Using CJS and JS. JS -> for population estimates. CJS -> for time-dependent survival and recapture rates of a given species

library(marked)

## Loading required package: lme4

## Loading required package: Matrix

## Loading required package: parallel

## This is marked 1.2.1

library(RMark)

## This is RMark 2.2.6  
## Documentation available at http://www.phidot.org/software/mark/rmark/RMarkDocumentation.zip

##   
## Attaching package: 'RMark'

## The following objects are masked from 'package:marked':  
##   
## collapseCH, compute.real, create.model.list, make.design.data,  
## merge\_design.covariates, model.table, process.ch,  
## process.data, setup.model, setup.parameters, splitCH

library(ggplot2)

cap <- read.csv("capturehistory.csv", stringsAsFactors = F)  
str(cap)  
print(cap)  
summary(cap)  
cap <- cap[which(!is.na(cap$c1)), ]  
summary(cap)

## Format data for MARK

library(reshape)  
  
# get data formatted for MARK/RMark  
cap$ch <- paste(cap$c1,  
 cap$c2,  
 cap$c3,  
 cap$c4,  
 cap$c5,  
 cap$c6,  
 cap$c7,  
 cap$c8,  
 cap$c9,  
 cap$c10,  
 cap$c11,  
 " ",  
 ";",  
 sep="")  
  
  
data=data.frame(cap)  
data  
transform=melt(cap, id.vars="mark.number")  
pivot=cast(transform, mark.number ~ value)  
pivot[is.na(pivot)]=0  
  
pivot[,2:ncol(pivot)][pivot[,2:ncol(pivot)] != 0] = 1  
lh <- 11;  
  
  
pivot$eh <- apply(pivot[2:lh],1,paste,collapse="") # concatenates encounter columns into eh  
pivot[2:lh] <- NULL # drops individual encounter columns  
# create commented tag  
pivot$mark.number <- paste("/\*", pivot$mark.number, "\*/", sep=" ")  
# sort by descending encounter histories  
pivot <- pivot[order(data$ch,decreasing=TRUE),]  
# tack on the frequency for the individual  
pivot$end <- "1;";  
# write out the input file  
write.table(pivot,file="cjs-pivot.inp",sep=" ",quote=F,col.names=F,row.names=F);  
  
pivot

## Format data for marked package

marked\_df <- cap  
  
marked\_df$ch <- paste0(cap$c1,  
 cap$c2,  
 cap$c3,  
 cap$c4,  
 cap$c5,  
 cap$c6,  
 cap$c7,  
 cap$c8,  
 cap$c9,  
 cap$c10,  
 cap$c11)  
   
model=crm(marked\_df)

## Model: CJS

## Processing data...

## 569 capture histories collapsed into 569

## Creating design data...

## Fitting model

## Computing initial parameter estimates

## Starting optimization for 2 parameters...

model  
model=cjs.hessian(model)  
model

str(marked\_df)  
marked\_df$sex.fac <- as.factor(marked\_df$sex)  
marked\_df$species.fac <- as.factor(marked\_df$species)  
mod <- crm(marked\_df,   
 model="JS",   
 groups = c("sex.fac"),   
 model.parameters=list(Phi = list(formula = ~sex.fac),   
 p = list(formula = ~1)))

## Model: JS

## Processing data...

## 572 capture histories collapsed into 572

## Creating design data...

## Fitting model

## Computing initial parameter estimates

mod$results$beta #found online but not sure how to use it  
mod  
  
cjs.hessian(mod)  
  
str(mod)  
# get values on real scale and not logit or log scales  
mod$results$reals  
  
  
mod <- crm(marked\_df,   
 model="JS",   
 groups = c("species.fac"),   
 model.parameters=list(Phi = list(formula = ~species.fac),   
 p = list(formula = ~species.fac)))

## Model: JS

## Processing data...

## 572 capture histories collapsed into 572

## Creating design data...

## Fitting model

## Computing initial parameter estimates

mod$results$beta #found online but not sure how to use it  
mod  
mod$results$reals

# Run with RMark (need to download Program MARK in order for this to work)

MarkPath='C:/Program Files (x86)/MARK'  
mark(marked\_df, model.parameters=list(p=list(formula=~1)),output=FALSE)$results$beta

abundance/effort AIC vvvv

data <- read.csv("Dragonfly1718Abundance.csv", stringsAsFactors = F)  
  
### Each possible combination of variables needs to be analyzed. The lowest score (not counting the full model?) wins. #####  
mod1<-glm(data$adjusted.abundance~data$pH)  
mod2<-glm(data$adjusted.abundance~data$elevation)  
mod3<-glm(data$adjusted.abundance~data$fishpresence)  
mod4<-glm(data$adjusted.abundance~data$lenticlotic)   
mod5<-glm(data$adjusted.abundance~data$openwater)  
mod6<-glm(data$adjusted.abundance~data$pH\*data$elevation)  
mod7<-glm(data$adjusted.abundance~data$pH\*data$elevation\*data$fishpresence)  
mod8<-glm(data$adjusted.abundance~data$pH\*data$elevation\*data$fishpresence\*data$lenticlotic)  
mod9<-glm(data$adjusted.abundance~data$pH\*data$elevation\*data$fishpresence\*data$lenticlotic\*data$openwater)   
mod10<-glm(data$adjusted.abundance~data$elevation\*data$fishpresence)  
mod11<-glm(data$adjusted.abundance~data$elevation\*data$fishpresence\*data$lenticlotic)  
mod12<-glm(data$adjusted.abundance~data$elevation\*data$fishpresence\*data$lenticlotic\*data$openwater)  
mod13<-glm(data$adjusted.abundance~data$fishpresence\*data$lenticlotic)  
mod14<-glm(data$adjusted.abundance~data$fishpresence\*data$openwater) #### Chicken dinner #####  
mod15<-glm(data$adjusted.abundance~data$fishpresence\*data$lenticlotic\*data$openwater) #### Almost the same ####  
mod16<-glm(data$adjusted.abundance~data$lenticlotic\*data$openwater)  
  
  
  
AIC(mod1, k=2) ### k = 2 is the "penalty" for adding more factors #####  
AIC(mod2, k=2)  
AIC(mod3, k=2)  
AIC(mod4, k=2)   
AIC(mod5, k=2)  
AIC(mod6, k=2)  
AIC(mod7, k=2)  
AIC(mod8, k=2)  
AIC(mod9, k=2)   
AIC(mod10, k=2)  
AIC(mod11, k=2)  
AIC(mod12, k=2)  
AIC(mod13, k=2)  
AIC(mod14, k=2) #### Chicken dinner #####  
AIC(mod15, k=2) #### Almost the same, 0.1258 difference ####  
AIC(mod16, k=2)

richness (jsut to see) vvvvv

### Each possible combination of variables needs to be analyzed. The lowest score (not counting the full model?) wins. #####  
mod1<-glm(data$Richness~data$pH)  
mod2<-glm(data$Richness~data$elevation)  
mod3<-glm(data$Richness~data$fishpresence)  
mod4<-glm(data$Richness~data$lenticlotic) #### Chicken dinner #####  
mod5<-glm(data$Richness~data$openwater)  
mod6<-glm(data$Richness~data$pH\*data$elevation)  
mod7<-glm(data$Richness~data$pH\*data$elevation\*data$fishpresence)  
mod8<-glm(data$Richness~data$pH\*data$elevation\*data$fishpresence\*data$lenticlotic)  
mod9<-glm(data$Richness~data$pH\*data$elevation\*data$fishpresence\*data$lenticlotic\*data$openwater)  
mod10<-glm(data$Richness~data$elevation\*data$fishpresence)  
mod11<-glm(data$Richness~data$elevation\*data$fishpresence\*data$lenticlotic)  
mod12<-glm(data$Richness~data$elevation\*data$fishpresence\*data$lenticlotic\*data$openwater)  
mod13<-glm(data$Richness~data$fishpresence\*data$lenticlotic)  
mod14<-glm(data$Richness~data$fishpresence\*data$openwater)  
mod15<-glm(data$Richness~data$fishpresence\*data$lenticlotic\*data$openwater)  
mod16<-glm(data$Richness~data$lenticlotic\*data$openwater)  
  
  
  
AIC(mod1, k=2) ### k = 2 is the "penalty" for adding more factors #####  
AIC(mod2, k=2)  
AIC(mod3, k=2)  
AIC(mod4, k=2) #### Chicken dinner #####  
AIC(mod5, k=2)  
AIC(mod6, k=2)  
AIC(mod7, k=2)  
AIC(mod8, k=2)  
AIC(mod9, k=2)  
AIC(mod10, k=2)  
AIC(mod11, k=2)  
AIC(mod12, k=2)  
AIC(mod13, k=2)  
AIC(mod14, k=2)  
AIC(mod15, k=2)  
AIC(mod16, k=2)