R code to estimate observed and expected/randomized detection percentages of a species (i.e., skunk) one, four, twelve, or one day after the detection of another species (i.e., coyote). Simulation code is modified from Niedballa et al. (2019).

#OBSERVED DETECTION PERCENTAGES DATASET

#enter all observed detection data

dat<-read.csv("allDettrue.csv")

#subset to two species

skunk<-subset(dat,dat$SpeciesID=="striped\_skunk")

coyote<-subset(dat,dat$SpeciesID=="coyote")

#create a matrix to hold observed detection percentages for each detection of the species with the least detections

coyoteskunk=matrix(NA, nrow(skunk),5)

#name matrix columns after time periods

colnames(coyoteskunk)<-list("one hour",

"four hour","twelve hour","one day")

#for loop to get observed detection %s

for (i in 1:nrow(skunk)) {

print(i)

#subsetting those detections from the same location

site<-skunk$CamID[i]

skunk.temp<-subset(skunk,skunk$CamID==site)

coyote.temp<-subset(coyote,coyote$CamID==site)

#for sites where there are coyotes

if(nrow(coyote.temp)>0){

#sort the detections of the species from the camera location i by date/time value (increasing)

coyote.adjusted<-sort(coyote.temp$AdjustedDT)

skunk.adjusted<-sort(skunk.temp$AdjustedDT)

#estimate the difference between each skunk detection date/time value by each coyote detection date/time value

hold<-skunk.adjusted-coyote.adjusted

#estimate the total number of date/time values calculated

total.obs<-length(hold)

#take out negative values

hold<-hold[hold>=0]

if(length(hold)>0){

#figure out how many fall into the time period categories

coyoteskunk[i,1]<-(length(hold[hold<=0.0417]))/total.obs

coyoteskunk[i,2]<-(length(hold[hold<=0.167]))/total.obs

coyoteskunk[i,3]<-(length(hold[hold<=0.5]))/total.obs

coyoteskunk[i,4]<-(length(hold[hold<=1]))/total.obs

coyoteskunk[i,5]<-(length(hold[hold<=2]))/total.obs

} else{

coyoteskunk[i,1]<-"NA"

coyoteskunk[i,2]<-"NA"

coyoteskunk[i,3]<-"NA"

coyoteskunk[i,4]<-"NA"

coyoteskunk[i,5]<-"NA"}

} else{

coyoteskunk[i,1]<-"NA"

coyoteskunk[i,2]<-"NA"

coyoteskunk[i,3]<-"NA"

coyoteskunk[i,4]<-"NA"

coyoteskunk[i,5]<-"NA"}

}

write.csv(coyoteskunk,"coyoteskunkreal.csv")

coyoteskunk<-read.csv("coyoteskunkreal.csv")

beep(2)

#estimate the median detection % for each time period

onehour.real<-median(as.numeric(na.omit(coyoteskunk[,2])))

fourhour.real<-median(as.numeric(na.omit(coyoteskunk[,3])))

twelvehour.real<-median(as.numeric(na.omit(coyoteskunk[,4])))

oneday.real<-median(as.numeric(na.omit(coyoteskunk[,5])))

##make final results holding matrix

observed=matrix(NA,2,4)

colnames(observed)<-list("one hour",

"four hour","twelve hour","one day")

rownames(observed)<-list("median.real","median.sim")

observed[1,1]<-onehour.real

observed[1,2]<-fourhour.real

observed[1,3]<-twelvehour.real

observed[1,4]<-oneday.real

#SIMULATED DETECTION PERCENTAGE DATASET

#read in data needed

#probability density for number of survey days

survey<-read.csv("density.csv")

##relative abundance probability densities for species

density.ra.coyote<-read.csv("coyotedensityra.csv")

density.ra.skunk<-read.csv("skunkdensityra.csv")

#min and max relative abundance for each species

coyotemaxra<-max(density.ra.coyote$ra)

skunkmaxra<-max(density.ra.skunk$ra)

coyoteminra<-min(density.ra.coyote$ra)

skunkminra<-min(density.ra.skunk$ra)

#create a range of values bounded by the min and max ra values

coyoterarange<-seq(from=coyoteminra,to=coyotemaxra,

length.out=nrow(density.ra.coyote))

skunkrarange<-seq(from=skunkminra,to=skunkmaxra,

length.out=nrow(density.ra.skunk))

#density\_distribution is actually taken from dvmkern fit of radian times against 1440 minutes/day

density.activity.coyote<-read.csv("coyoteactivity.csv")[,2]

density.activity.skunk<-read.csv("striped\_skunkactivity.csv")[,2]

#let's do 1k iterations

a=1000

#make a matrix to hold the resulting simulated values

meds.sims=matrix(NA, a,4)

colnames(meds.sims)<-list("one hour",

"four hour","twelve hour","one day")

for(a in 1:nrow(meds.sims)){

print(a)

#run a weighted sample on the number of survey days

#max number of survey days is 2899

#survey$x has the probability of that number of days being chosen

sim.days<-sample(x=(1:2899),size=length(1:2899),

prob=survey$x,replace = F)[1]

#X survey days at 1440 minutes per day

n.events<-(sim.days\*1440)

#run a weighted sample on the number of detections for each species

#bound by the max number of detections for that species

#SPECIES$x has the probability of that number of detections being chosen

#DO NOT ALLOW 0 IN THIS SINCE REAL OBSERVATIONS ARE TAKING FROM LOCATIONS

#WHERE BOTH SPECIES ARE DETECTED

n\_records\_coyote<-round((sample(x=coyoterarange,size=length(coyoterarange),

prob=density.ra.coyote$prob,replace = F)[1])\*sim.days)

n\_records\_skunk<-round((sample(x=skunkrarange,size=length(skunkrarange),

prob=density.ra.skunk$prob,replace = F)[1])\*sim.days)

#total number of observations

total.obs<-max(n\_records\_coyote,n\_records\_skunk)

# calculate observation probabilites for each minute in the study period incorporating daily activity patterns

# this is a relative measure, not an actual probability, and will be used as a probability weight for the random sample which realises the species records

# takes into account daily activity patterns, but not yet avoidance

#this creates a vector of n\_events length, repeating the probabilities in

#density\_distribution n\_events times.

observation.prob.coyote <- rep(1, times = n.events) \* density.activity.coyote

observation.prob.skunk <- rep(1, times = n.events) \* density.activity.skunk

##so this creates a vector of numbers from 1 to whatever n\_events is

#then tells sample to look through that list of numbers, pick out n\_records\_A

#amount, with each vector being given a weight and likelihood

#of being picked based on what time it happens

#its basically a date time number value

#divide it by 1440 to get the day and the actual time

time.obs.coyote <- sort(sample(x = seq(1, n.events),

size = n\_records\_coyote,

prob = observation.prob.coyote,

replace = FALSE))

time.obs.skunk <- sort(sample(x = seq(1, n.events),

size = n\_records\_skunk,

prob = observation.prob.skunk,

replace = FALSE))

#once you have the two species A and B's times of detection

#subtract them, B-A

time.obs<-time.obs.skunk-time.obs.coyote

#then remove the negative values and divide by 1440

time.obs<-(time.obs[time.obs>=0])/1440

if(length(time.obs)>0){

#figure out how many fall into the time period categories

meds.sims[a,1]<-(length(time.obs[time.obs<=0.0417]))/total.obs

meds.sims[a,2]<-(length(time.obs[time.obs<=0.167]))/total.obs

meds.sims[a,3]<-(length(time.obs[time.obs<=0.5]))/total.obs

meds.sims[a,4]<-(length(time.obs[time.obs<=1]))/total.obs

} else{

meds.sims[a,1]<-"NA"

meds.sims[a,2]<-"NA"

meds.sims[a,3]<-"NA"

meds.sims[a,4]<-"NA"}

}

#write the simulation data

write.csv(meds.sims,"coyoteskunk\_simulatedrandomresults.csv")

meds.sims<-read.csv("coyoteskunk\_simulatedrandomresults.csv")

#estimate median

onehour.median.sims<-median(meds.sims$one.hour,na.rm = T)

fourhour.median.sims<-median(meds.sims$four.hour,na.rm = T)

twelvehour.median.sims<-median(meds.sims$twelve.hour,na.rm = T)

oneday.median.sims<-median(meds.sims$one.day,na.rm = T)

observed[2,1]<-onehour.median.sims

observed[2,2]<-fourhour.median.sims

observed[2,3]<-twelvehour.median.sims

observed[2,4]<-oneday.median.sims

write.csv(observed,"coyoteskunk\_realsimfinalrandom.csv")

Literature Cited

Niedballa, J., A. Wilting, R. Sollmann, H. Hofer, A. Courtiol, M. Rowcliffe, and J. Ahumada. 2019. Assessing analytical methods for detecting spatiotemporal interactions between species from camera trapping data. Remote Sensing in Ecology and Conservation.