**General Outline for Estimating seabird bycatch.**

This outline is a general guide to estimating the seabird bycatch in space and time for a given fleet or aggregation of fleets. The outline is a description of a two step process, where the first step is to estimate seabird bycatch rate (BPUE) , and the second is to extrapolate from that rate to a pre-defined study area and estimate the total seabird bycatch for a given amount of effort. This outline assumes that you are familiar with basic quantitative fisheries and statistical modeling concepts, and can be used in any appropriate statistical programming languages, R is recommended.

**Required Data and Format**

**Fishing effort**. For the total effort data, and the observed effort data, please create CSV files with the following fields, for all surface longline fishing with 5-degree grid cells in the southern hemisphere (e.g., one cell is would be defined by longitude 165 to 170, and latitude -35 to -30).

● Flag : Flag country of the vessel.

● Year : 4-digit year

● Month : 1 to 12 (month of the year)

● Latitude: latitude of the center of the cell (e.g, 167.5)

● Longitude : longitude of the center of the cell (e.g, -32.5)

● Effort : total effort (sum of of hooks in each fishing group, observed, month, cell, stratum)

For the total effort data, please include all fishing from the years 2012, 2013, 2014, 2015, and 2016. For the observed effort data, please include years for as long as the associated seabird bycatch observations are considered reliable and are considered to be representative of current practice.

**Seabird bycatch**. A csv file of observed seabird captures, with the following fields, the first six of which are necessary:

* Flag: Flag country of the vessel.
* Year : 4-digit year of capture
* Month : 1 to 12 (month of the year)
* Latitude: latitude of the center of the cell (e.g, 167.5)
* Longitude : longitude of the center of the cell (e.g, -32.5)
* Captures (BIRD\_NUM): total number of observed captures (live and dead)
* Set date: the date of the set
* ToriLn\_Num: The number of bird scaring (Tori) lines used.
* Moonphase: moon phase of the set
* Set.duration: Time in decimal hours
* hook\_set: the total number of hooks set
* sd.depth the standard deviation of the depth in 5 degree cell where the data are
* mu.depth Average depth of the cell

**Code Heuristics for estimating a seabird bycatch using a GAM**

**Overall Steps:** First the outlier data must be trimmed and then the GAM can be run, prediction step completed and results saved. The data fields include bathymetric data that should be aligned with the set locations. All data must be relatively continuous and not have too many outliers, therefore use the following steps to reduce the chances of outlying data having an undue effect. For the rest of this guide the observer data will be called ‘obsdat’.

Create a working director

setwd("C:/Seabirds")

#Check if its there

getwd()

# Dump results here

folder = "results"

dir.create(paste(folder),showWarnings = FALSE)

#Read in data with Birdbycatch information

obsdat=read.csv("obsdat.csv")[,-1]

obsdat = obsdat[obsdat$set.duration>2 & obsdat$set.duration<=8,]

Trim the set time to fall between two and 12 hours

## Calculate the observed BPUE

## Birds/1000 hooks

obsdat$y = obsdat$BIRD\_NUM/obsdat$hook\_set\*1000

## Moonphase calculation using suncalc package in R

Moonlight<-getMoonIllumination(date=as.Date(obsdat$set.date),

keep = c("fraction", "phase","angle"))

obsdat$Moonphase<-Moonlight$phase

## Define Breeding Season

# Define breeding season months

Breeding = c(10:12,1:4)

# Add breeding season 0: Non-Breeding; 1: Breeding

obsdat$Season = ifelse(obsdat$mm %in% Breeding,1,0)

##Tori line

obsdat$ToriLn\_Num<-ifelse(obsdat$ToriLn\_Num<0,0,obsdat$ToriLn\_Num)

obsdat$ToriYN<-ifelse(obsdat$ToriLn\_Num>1,1,0)

obsdat$ToriYN<-as.factor(obsdat$ToriYN)

summary(obsdat$ToriYN)

# FIT GAM

# The assumption is that Lat varies between breeding and non-breeding

# The rest is a fairly 'reduced' gam to avoid funky behaviour

fit = gam(y ~ s(lat\_cen,by=as.factor(Season),k=5)+Season+s(mu.depth,k=5)+s(sd.depth,k=5)+s(Moonphase,k=5)+s(set.daylight,k=4)+(flag\_id)+lat\_cen,data=obsdat,family=Tweedie(p=1.8))

# Plot Partial effects

Par = list(mfrow=c(3,2),mar = c(3.5, 3.5, 0.1, 0.1), mgp =c(2.,0.5,0), tck = -0.02,cex=0.8)

png(file = paste0(folder,"/GAM\_PartialEffect.png"), width = 5, height = 6,

res = 200, units = "in")

par(Par)

plot(fit,ylim=c(-2.5,2.5))

dev.off()

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Predict Grids for all Fleets

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# mean set duration by fleet

fleets = levels(obsdat$flag\_id)

mu.setduration = aggregate(set.duration~flag\_id,obsdat,mean)

mu.daylight = aggregate(set.daylight~flag\_id,obsdat,mean)

ngrids = nrow(predgrid)

# repeat grid times the number of fleets

predgrids = NULL

for(i in 1:length(fleets)) predgrids = rbind(predgrids,predgrid)

# Make prediction dataset grouped by fleets and their characteristics:

predat1=data.frame(predgrids,Season=0, Moonphase=0.2,set.daylight=rep(mu.daylight[,2],each=ngrids),set.duration=rep(mu.setduration[,2],each=ngrids),flag\_id=rep(fleets,each=ngrids))

head(predat1)

nrow(predat1)

# Now iterate over Moonphase in 5 steps

moon.steps = c(0.4,0.6,0.8,1)

predat2 = predat1

for(i in 1:4){

addmoon = predat1

addmoon$Moonphase = moon.steps[i]

predat2 = rbind(predat2,addmoon)

}

# Add a breeding season

predat3 = predatNB = predat2 # NB non-breeding

predatBR = predat2 # BR Breeding

predatBR$Season = 1

predat = rbind(predatNB,predatBR) # Combine

# Predict for all combinations

predatBR$yhat = predict(fit,predatBR,type="response")

predatNB$yhat = predict(fit,predatNB,type="response")

predat$yhat = predict(fit,predat,type="response")

# Get predicted mean BPUE per grid

BPUExy = aggregate(yhat~lat\_cen+lon\_cen, predat,mean)

BPUExyBR = aggregate(yhat~lat\_cen+lon\_cen, predatBR,mean)

BPUExyNB = aggregate(yhat~lat\_cen+lon\_cen, predatNB,mean)

# Plot BPUE distribution for Breeding vs Non-Breeding

Par = list(mfrow=c(2,1),mar = c(3.5, 3.5, 0.1, 0.1), mgp =c(2.,0.5,0), tck = -0.02,cex=0.8)

png(file = paste0(folder,"/Spatial.avgBPUE.png"), width = 7.5, height = 6.,

res = 200, units = "in")

par(Par)

ylim=c(-50.5,-20.5)

xlim=c(-60.5,60)

ypl = ylim;xpl = xlim

byx = byy =1 # Set grid size (here 0.5\*degree)

x = BPUExyBR$lon\_cen # dE$set\_start\_longitude

y = BPUExyBR$lat\_cen #set\_start\_latitude

z = BPUExyBR$yhat #dM$Nhooks/1000/5

breaks.z =c(seq(0,0.1,0.02),seq(0.11,.5,0.1),seq(1,4,0.5))# ,seq(0.4,1,0.1),1.5,2,2.5)

#breaks.z = c(seq(0,0.02,0.005),seq(0.021,0.1,0.01))#,seq(0.105,.15,0.05))

grd <- make.grid(x,y,z, byx, byy, xlim, ylim,fun=mean)

basemap(xpl, ypl, main = "",bg="lightblue", ylab="Latitude",xlab="Longitude")

draw.grid(grd,breaks.z)

obsBR = obsdat[obsdat$Season==1,]

ox = obsBR$lon\_cen # dE$set\_start\_longitude

oy = obsBR$lat\_cen #set\_start\_latitude

oz = obsBR$pa #dM$Nhooks/1000/5

ogrd <- make.grid(ox,oy,oz, byx, byy, xlim, ylim,fun=mean)

legend.grid("bottomright", breaks=breaks.z, type=2, inset=0.02, title="BPUE/1000 hooks",cex=0.7)

legend("top","Non-Breeding Season",bty="n",x.intersp = -4,y.intersp = -0.3)

# make grid for observed effort

prob.obs = NULL

lat = as.numeric(colnames(ogrd))

nlat = length(lat)

long = as.numeric(rownames(ogrd))

nlong = length(long)

for(i in 1:nlat){

prob.obs = rbind(prob.obs,cbind(lat[i],long,ogrd[,i]))

}

colnames(prob.obs) = c("lat","lon","PE")

rownames(prob.obs) = 1:nrow(prob.obs)

prob.obs = data.frame(prob.obs)

prob.obs = prob.obs[is.na(prob.obs[,3])==FALSE,]

points(lat~lon,prob.obs,cex=0.5,pch=16,col=rgb(0,0,1,0.4))

draw.shape(world, col="cornsilk2")

draw.shape(sa, col="cornsilk3")

x = BPUExyNB$lon\_cen # dE$set\_start\_longitude

y = BPUExyNB$lat\_cen #set\_start\_latitude

z = BPUExyNB$yhat #dM$Nhooks/1000/5

grd <- make.grid(x,y,z, byx, byy, xlim, ylim,fun=mean)

basemap(xpl, ypl, main = "",bg="lightblue", ylab="Latitude",xlab="Longitude")

draw.grid(grd,breaks.z)

obsNB = obsdat[obsdat$Season==0,]

ox = obsNB$lon\_cen # dE$set\_start\_longitude

oy = obsNB$lat\_cen #set\_start\_latitude

oz = obsNB$pa #dM$Nhooks/1000/5

ogrd <- make.grid(ox,oy,oz, byx, byy, xlim, ylim,fun=mean)

legend.grid("bottomright", breaks=breaks.z, type=2, inset=0.02, title="BPUE/1000 hooks",cex=0.7)

legend("top","Breeding Season",bty="n",x.intersp = -4,y.intersp = -0.3)

# make grid for observed effort

prob.obs = NULL

lat = as.numeric(colnames(ogrd))

nlat = length(lat)

long = as.numeric(rownames(ogrd))

nlong = length(long)

for(i in 1:nlat){

prob.obs = rbind(prob.obs,cbind(lat[i],long,ogrd[,i]))

}

colnames(prob.obs) = c("lat","lon","PE")

rownames(prob.obs) = 1:nrow(prob.obs)

prob.obs = data.frame(prob.obs)

prob.obs = prob.obs[is.na(prob.obs[,3])==FALSE,]

points(lat~lon,prob.obs,cex=0.5,pch=16,col=rgb(0,0,1,0.4))

draw.shape(world, col="cornsilk2")

draw.shape(sa, col="cornsilk3")

dev.off()

# Get mean BPUE by Season Lat/Lon and Flag (average across moon phase)

pdat= aggregate(yhat~Season+lat\_cen+lon\_cen+flag\_id,predat,mean)

# Define CPUE groupings

# aggregate across moon

COUNTRY\_X = pdat[pdat$flag\_id==" COUNTRY\_X ",]

COUNTRY\_y = pdat[pdat$flag\_id==" COUNTRY\_y ",]

# Average to get fleet grouping's mean BPUE

JPN = aggregate(yhat~Season+lat\_cen+lon\_cen, COUNTRY\_X,mean)

COUNTRY\_X = aggregate(yhat~Season+lat\_cen+lon\_cen, COUNTRY\_Y,mean)

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Multiply with RFMO Effort data

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

load("total\_effort\_ll\_agg5x5\_cntry\_20S\_30May.rdata",verbose=T)

summary(total\_eff\_5x5)

effort = total\_eff\_5x5[total\_eff\_5x5$RFMO=="IOTC" | total\_eff\_5x5$RFMO=="ICCAT",]

effort=effort[is.na(effort$flag)==FALSE,]

effort = subset(effort,lon5<=50 & lat5<= -20)

# Plot mean BPUE and mean annual Effort

Par = list(mfrow=c(2,1),mar = c(3.5, 3.5, 0.1, 0.1), mgp =c(2.,0.5,0), tck = -0.02,cex=0.8)

png(file = paste0(folder,"/Spatial\_5x5.png"), width = 7.5, height = 6.,

res = 200, units = "in")

par(Par)

ylim=c(-50.5,-20.5)

xlim=c(-60.5,60)

ypl = ylim;xpl = xlim

byx = byy =5 # Set grid size (here 0.5\*degree)

x = BPUExy$lon\_cen # dE$set\_start\_longitude

y = BPUExy$lat\_cen #set\_start\_latitude

z = BPUExy$yhat #dM$Nhooks/1000/5

breaks.z =c(seq(0,0.1,0.02),seq(0.2,0.5,0.1),seq(0.6,2,0.2))#,seq(0.301,0.3,0.05),seq(0.4,1,0.1),1.5,2,2.5)

grd.BB <- make.grid(x,y,z, byx, byy, xlim, ylim,fun=mean)

basemap(xpl, ypl, main = "",bg="lightblue", ylab="Latitude",xlab="Longitude")

draw.grid(grd.BB,breaks.z)

draw.shape(world, col="cornsilk2")

draw.shape(sa, col="cornsilk3")

legend.grid("bottomright", breaks=breaks.z, type=2, inset=0.02, title="BPUE/1000 hooks",cex=0.7)

effy = aggregate(hooks~year+lon5+lat5,effort,sum)

#eff = mean(effy$hooks/1000)

eff = aggregate(hooks~lon5+lat5,effort,sum)

ey = aggregate(hooks~year,effort,sum)

nyrs = nrow(ey)

mean(ey$hooks)/1000

sum(eff$hooks)/nyrs/1000

eff$hooks= eff$hooks/1000/nyrs

breaks.z =c(seq(0,2000,200),seq(3000,6000,1000))#,seq(0.301,0.3,0.05),seq(0.4,1,0.1),1.5,2,2.5)

#breaks.z = c(seq(0,0.02,0.005),seq(0.021,0.1,0.01))#,seq(0.105,.15,0.05))

grd.E <- make.grid(x=eff$lon5,y=eff$lat5,z=eff$hooks, byx, byy, xlim, ylim,fun=mean)

basemap(xpl, ypl, main = "",bg="lightblue", ylab="Latitude",xlab="Longitude")

draw.grid(grd.E,breaks.z)

draw.shape(world, col="cornsilk2")

draw.shape(sa, col="cornsilk3")

legend.grid("bottomright", breaks=breaks.z, type=2, inset=0.02, title="Effort 1000 hooks",cex=0.7)

dev.off()

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Define Effort groupings

#\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# here you must define the available flags and groupings that you are using

flag.def = c("Country\_X", "Counry\_Y”)

for(i in 1:length(flag.def)){

assign(paste0(flag.def[i],".E"), effort[effort$flag==flag.def[i],])

}

OTHER.E = effort[effort$flag %in% flag.def==FALSE, ]

flag.def = c(flag.def,"OTHER")

# Set up BB estimation grid

ngroups = length(flag.def)

years= unique(effort$year)

nyrs = length(years)

months = 12

BB.grid = Eff.grid = matrix(NA,nrow=nyrs\*12,ncol=ngroups)

# Define grid

ylim=c(-47.5,-17.5)

xlim=c(-62.5,50)

ypl = ylim;xpl = xlim

byx = byy = 5 # Set grid size (here 0.5\*degree)

# Run over year/0month

index = 0

for(y in 1:nyrs){

for(m in 1:12){

index = index +1

for(g in 1:ngroups){

temp.E = get(paste0(flag.def[g],".E"))

temp.E = temp.E[temp.E$year==years[y] & temp.E$mm == m,]

grdE = make.grid(temp.E$lon5,temp.E$lat5,temp.E$hooks/1000, byx, byy, xlim, ylim,fun=sum)

temp.BB = get(paste0(flag.def[g]))

if(m %in%Breeding){

temp.BB = temp.BB[temp.BB$Season ==1,]

} else {

temp.BB = temp.BB[temp.BB$Season == 0,]

}

grdBB= make.grid(temp.BB$lon\_cen,temp.BB$lat\_cen,temp.BB$yhat, byx, byy, xlim, ylim,fun=sum)

# note Effort

Eff.grid[index,g] = sum(grdE,na.rm = TRUE)

# get Bird Bycatch

BB.grid[index,g] = sum(grdE\*grdBB,na.rm = TRUE)

}

}

}

colnames(BB.grid) = paste(flag.def)

rownames(BB.grid) = 1:nrow(BB.grid)

colnames(Eff.grid) = paste(flag.def)

rownames(Eff.grid) = 1:nrow(Eff.grid)

BBgrid = data.frame(Year = rep(years,each=12),Month=rep(1:12,nyrs),BB.grid)

Eff.grid = data.frame(Year = rep(years,each=12),Month=rep(1:12,nyrs),Eff.grid)

BBgrid$Total = apply(BBgrid[,-c(1:2)],1,sum)

Eff.grid$Total = apply(Eff.grid[,-c(1:2)],1,sum)

aggregate(Total~Year,BBgrid,sum)

aggregate(Total~Year,Eff.grid,sum)

# Birds per year

mean(aggregate(Total~Year,BBgrid,sum)[,2])

write.csv(BBgrid,paste0(folder,"/BBgrid.csv"))

write.csv(Eff.grid,paste0(folder,"/Effort.grid.csv"))