bearded seal CKMR data formatting

Paul Conn

5/2/2022

Introduction

This document is intended to describe the processes and rationale in transforming bearded seal data provided by ADF&G into a format suitable for analyzing with close-kin mark-recapture (CKMR) models. These data are almost there, but it will be necessary to (1) remove a few records and reformat data, and (2) formulate posterior distributions for the age of each sample. We'll describe each of these processes in turn after describing data inputs.

Source data

The data sets provided by ADF&G include "Samples.csv", "MHS_matches.csv", "PHS_matches.csv", and "POP_matches.csv". The "Samples.csv" file includes a record for each individual seal, as well as sex, age data, date of harvest, village of harvest, and some information from the genetic analysis The other files provide row indices for maternal half-sibling, paternal half-sibling, and parent-offspring pairs, respectively, as well as a "PLOD" log odds score. The PLOD score was used in kin finding and may be used to eliminate certain half-sibling pairs from the match list, in the event we try to use a higher PLOD threshold to discriminate matches from non-matches.

Eliminating individuals

There are several individuals included in "Samples.csv" that we'll likely need to remove from analysis. However, this means that the row numbers of all individuals below the removed animals in the Samples dataset will change, and we'll want to be able to adjust the various row indices in the various matches files to match updated Sample datasets. Let's provide a framework for this by attaching individual identifiers to each of the "matches" datasets. We'll then remove some duplicates and at one individual missing a year of harvest.

```
Samples <- read.csv("../data/Samples.csv",stringsAsFactors=FALSE, fileEncoding="latin1")
head(Samples)
```

```
##   Record Well Dart_job Our_plate Our_sample Fishtot
## 1      1   B5 9.13e+11     Cap12   2010BS40  801677
## 2      2   C9 9.13e+11      Cap4 EB20GAM002 1009179
## 3      3   H5 9.13e+11      Cap4 EB16GAM070  988459
## 4      4  D11 9.13e+11      Cap9  SH-G18-04  798706
## 5      5  F10 9.13e+11     Cap14  SH-S04-05  729770
## 6      6   C2 9.13e+11      Cap1      09BS8  842154
##                                          File                              MD5
## 1 Cluster_Count_0_Target_mainspeciesONLY.csv 07ccd032f47248dc13be14c102f98e47
## 2 Cluster_Count_0_Target_mainspeciesONLY.csv 07ccd032f47248dc13be14c102f98e47
## 3 Cluster_Count_0_Target_mainspeciesONLY.csv 07ccd032f47248dc13be14c102f98e47
## 4 Cluster_Count_0_Target_mainspeciesONLY.csv 07ccd032f47248dc13be14c102f98e47
## 5 Cluster_Count_0_Target_mainspeciesONLY.csv 07ccd032f47248dc13be14c102f98e47
## 6 Cluster_Count_0_Target_mainspeciesONLY.csv 07ccd032f47248dc13be14c102f98e47
##   TargetID              UID       SP      Lcode      lat          lon  MONTH
## 1  2409050   2010BS40_B5_12 BARBATUS     Barrow 71.29056  -156.788611    JUL
## 2  2408331  EB20GAM002_C9_4 BARBATUS    Gambell 63.77611  -171.700833    JAN
## 3  2408304  EB16GAM070_H5_4 BARBATUS    Gambell 63.77611  -171.700833 Winter
## 4  2408818  SH-G18-04_D11_9 BARBATUS Shishmaref 66.25556  -166.072222    OCT
## 5  2409282 SH-S04-05_F10_14 BARBATUS Shishmaref 66.25556  -166.072222    OCT
## 6  2407993       09BS8_C2_1 BARBATUS     Barrow 71.29056  -156.788611
##        DATE YEAR AgeCombined AgeType ToothAgeQuality Age1 Age2 Aging.comments
## 1 22-Jul-10 2010           6   Tooth               B    5    6
## 2 21-Jan-20 2020           4   Tooth               A   NA   NA
## 3           2016           0   Tooth               A   NA   NA
## 4 13-Oct-04 2004           0   Tooth               A   NA   NA
## 5 21-Oct-05 2005           0   Tooth               A   NA   NA
## 6           2009           1   Tooth               A   NA   NA
##   SexDArT   SLcm Batch PlateIDcombined Tissue Weightg Comments Source npoly
## 1       F     NA     2              15  Liver  0.0093                      0
## 2       M 167.64     2               7 Muscle  0.0104     none             0
## 3       M     NA     2               7 Muscle  0.0099     none             0
## 4       M 149.00     2              12  Liver  0.0105             UAM      0
## 5       M 154.00     2              17   Skin  0.0096     DMSO  SWFSC      0
## 6       F     NA     2               4 Muscle  0.0080     none             0
```

```r
matches_POP <- read.csv("../data/POP_matches.csv")
matches_MHS <- read.csv("../data/MHS_matches.csv")
matches_PHS <- read.csv("../data/PHS_matches.csv")
matches_POP$IDi = Samples$Our_sample[matches_POP$i]
matches_POP$IDj = Samples$Our_sample[matches_POP$j]
matches_MHS$IDi = Samples$Our_sample[matches_MHS$i]
matches_MHS$IDj = Samples$Our_sample[matches_MHS$j]
matches_PHS$IDi = Samples$Our_sample[matches_PHS$i]
matches_PHS$IDj = Samples$Our_sample[matches_PHS$j]


Which_remove = c(unique(grep("LowGene",Samples$Aging.comments)),which(is.na(Samples$YEAR)))
Samples_new = Samples[-Which_remove,]

#now reorder i,j based on those remaining
for(irec in 1:nrow(matches_POP)){
  matches_POP$i[irec] = which(Samples_new$Our_sample %in% matches_POP$IDi[irec])
  matches_POP$j[irec] = which(Samples_new$Our_sample %in% matches_POP$IDj[irec])
}
for(irec in 1:nrow(matches_MHS)){
  matches_MHS$i[irec] = which(Samples_new$Our_sample %in% matches_MHS$IDi[irec])
  matches_MHS$j[irec] = which(Samples_new$Our_sample %in% matches_MHS$IDj[irec])
}
for(irec in 1:nrow(matches_PHS)){
  matches_PHS$i[irec] = which(Samples_new$Our_sample %in% matches_PHS$IDi[irec])
  matches_PHS$j[irec] = which(Samples_new$Our_sample %in% matches_PHS$IDj[irec])
}
```

Newborn pups vs yearlings

One issue is that there are many individuals harvested in the spring that are recorded as age ‘0’ but when examining further notes it is possible to determine that they are either weeks old or 11 months old. Since we anticipate employing a population model with a postbreeding census with an annual time step that occurs on

May 1, we'll want to adjust the month of harvest for the 'weeks old' ones so that they are harvested in May (to be associated with that years cohort), and to adjust the 11 month old ones so that they're harvested in April (so that they are associated with the previous years cohort).

First, we'll treat 0 year old observations recorded by Lara Horstmann as if they are truly "weeks old" since weeks vs. 11 month old determinations are not available for these. Based on the time samples were collected, this is clearly the most likely option. Using this approach, here are eight newborn pups that are harvested in April - for these we will shift the month of harvest to May. There are also 3 individuals harvested in May that have an age of "11 months" - we'll adjust these to have a harvest in April so that they're associated with the previous years cohort.

```r
Pup_data = read.csv("../data/Spring_pups_info.csv",header=TRUE)
Pup_data$Adjusted.aging.comments[which(Pup_data$Adjusted.aging.comments=="weeks")]="Weeks"
Which_Lara = grep("Lara",Pup_data$Adjusted.aging.comments)
Pup_data[Which_Lara,"Adjusted.aging.comments"]="Weeks"
Newborns_april = which(Pup_data$MONTH=="APR" & Pup_data$Adjusted.aging.comments=="Weeks")
Which_new = which(Samples_new$Our_sample %in% Pup_data$Our_sample[Newborns_april])
Samples_new[Which_new,"MONTH"]="MAY"
Yearlings_may = which(Pup_data$MONTH=="MAY" & Pup_data$Adjusted.aging.comments=="11 months")
Which_yearling = which(Samples_new$Our_sample %in% Pup_data$Our_sample[Yearlings_may])
Samples_new[Which_yearling,"MONTH"]="APR"
```

Uncertain months

In most cases a month of harvest was specified, but in others it was not. Let's specify a month for those that are missing or are inexact (e.g., "Winter"). Since our annual time step is May 1, we'll index May by month 0, and April by month 11. Given this time step, we'll also associate harvests made January - April with the previous year.

```r
Month = Samples_new$MONTH
Month[which(Month=="Spring")]="MAY"
Month[which(Month=="Summer")]="JULY"
Month[which(Month=="Fall")]="SEP"
Month[which(Month=="Winter")]="JAN"
Month = replace(Month,Month=="MAY","0")
Month = replace(Month,Month=="JUN","1")
Month = replace(Month,Month=="JUL","2")
Month = replace(Month,Month=="AUG","3")
Month = replace(Month,Month=="SEP","4")
Month = replace(Month,Month=="OCT","5")
Month = replace(Month,Month=="NOV","6")
Month = replace(Month,Month=="DEC","7")
Month = replace(Month,Month=="JAN","8")
Month = replace(Month,Month=="FEB","9")
Month = replace(Month,Month=="MAR","10")
Month = replace(Month,Month=="APR","11")
Month = as.numeric(Month)
```

```
## Warning: NAs introduced by coercion
```

```r
mean_month=mean(Month,na.rm=TRUE)
Month[which(is.na(Month))]=mean_month
Samples_new$MONTH = Month

Which_winter=which(Samples_new$MONTH > 7)
Samples_new$YEAR[Which_winter]=Samples_new$YEAR[Which_winter]-1
```

Age distributions

For each harvested seal, we either need to know its age exactly (for models where age is assumed known), or to specify probabilities that an animal is a particular age (for models where age is assumed uncertain). Let's assign definitive ages as well as probability distributions for both types of analyses.

Certain ages

The 'AgeCombined' field in 'Samples' has mostly been filled in with the best age estimate possible, but there are a few exceptions, and a few ages that are clearly wrong. First, one of the POPs indicates that an age 1 male, harvested in 2015, was an offspring of a mother who was killed in 2013. Clearly, this male (seal EB15PH003) must have been at least a two year old. We'll thus change it to be 2 instead of 1. Second, one of the maternal half-sibling pairs (siblings with the same mother) had two individuals with the same apparent birth years. Since bearded seals only have one pup per year, this is clearly an error of some kind: either an aging error, or the two are actually a paternal HSP that are "lucky" enough to share mitochondrial DNA. Of the two, I suspect an aging error is the most probable; for purposes of this particular analysis, we'll change the age of the 6 year old to be 7.

There are also a number of missing AgeCombined values or entries that had missing or *very* uncertain ages (e.g., tooth aging failed and there was only claws to age with which are problematic for older age classes). Although it seems possible to come up with probability distributions for these animals, deleting them is probably the best choice for an analysis relying on age being certain. There are also about 10 records where the ages were 0 or 1, with the note "age 1 if DOK spring" - these are seals for which month of kill is unknown. However, since most seals are harvested in the spring or early summer, it is likely that these animals are one year olds (rather than a young-of-year harvested in the winter that might start to have an annuli that looks like a one year old). Lets change these to one-year-olds, get rid of the really uncertain records, make sure none are involved in kin-pair matches, and readjust kin-pair entries.

```r
Samples_certain_age <- Samples_new
Samples_certain_age$AgeCombined[which(Samples_new$Our_sample=='EB15PH003')]=2
Samples_certain_age$AgeCombined[which(Samples_new$Our_sample=='EB13PH016')]=7

Which_01 = which(Samples_certain_age$AgeCombined==0.5)
Samples_certain_age$AgeCombined[Which_01]=1

Which_remove = c(which(is.na(Samples_certain_age$AgeCombined)),
                 which((Samples_certain_age$AgeCombined %% 1)!=0))  #get rid of 0.5 age increments wh

Remove_IDs = Samples_certain_age$Our_sample[Which_remove]
Kin_pair_IDs = c(matches_POP$IDi,matches_POP$IDj,matches_MHS$IDi,matches_MHS$IDj,
                 matches_PHS$IDi,matches_PHS$IDj)
which(Remove_IDs %in% Kin_pair_IDs)  #none involved in a kin pair
```

```
## integer(0)
```

```r
Samples_certain_age = Samples_certain_age[-Which_remove,]

matches_POP_ca = matches_POP
matches_MHS_ca = matches_MHS
matches_PHS_ca = matches_PHS

#now reorder i,j based on those remaining
for(irec in 1:nrow(matches_POP)){
  matches_POP_ca$i[irec] = which(Samples_certain_age$Our_sample %in% matches_POP$IDi[irec])
  matches_POP_ca$j[irec] = which(Samples_certain_age$Our_sample %in% matches_POP$IDj[irec])
}
```

```
for(irec in 1:nrow(matches_MHS)){
  matches_MHS_ca$i[irec] = which(Samples_certain_age$Our_sample %in% matches_MHS$IDi[irec])
  matches_MHS_ca$j[irec] = which(Samples_certain_age$Our_sample %in% matches_MHS$IDj[irec])
}
for(irec in 1:nrow(matches_PHS)){
  matches_PHS_ca$i[irec] = which(Samples_certain_age$Our_sample %in% matches_PHS$IDi[irec])
  matches_PHS_ca$j[irec] = which(Samples_certain_age$Our_sample %in% matches_PHS$IDj[irec])
}

#list object to export for further analysis
CKMR_certain_age = list(Samples=Samples_certain_age,matches_POP=matches_POP_ca,
                        matches_MHS=matches_MHS_ca, matches_PHS=matches_PHS_ca)

save(CKMR_certain_age,file="CKMR_sample_data.RData")
```