

Creating and Applying an Age-Length Key

Derek H. Ogle, Northland College

5-Mar-2015

Preliminaries

```
> library(fishWiDNR)    # for setDBClasses()
> library(FSA)          # for lencat(), filterD()
> library(dplyr)        # for %>%
> library(magrittr)     # for %<>%
> library(lubridate)    # for month()
> library(plotrix)      # for plotH(), histStack()
```

Loading Data and Initial Manipulations

This first part is a repeat from the previous handout.

```
> setwd("C:/aaaWork/Web/fishR/Courses/WiDNR_Statewide_2015/Day1_IntroR_FMDData")
> d <- read.csv("SAWYER_fish_raw_data_012915.csv", stringsAsFactors=FALSE, na.strings=c("-", "NA", "")) %>%
  setDBClasses(type="RDNR") %>%
  select(County, Waterbody.Name, Survey.Year, Sample.Date, Gear, Fish.Data.Seq.No, Species,
         Length.or.Lower.Length.IN, Gender, Age..observed.annuli., Edge.Counted.Desc, Age.Structure) %>%
  mutate(mon=month(Sample.Date, label=TRUE)) %>%
  mutate(lcat=lencat(Length.or.Lower.Length.IN, w=0.5)) %>%
  arrange(Species, Length.or.Lower.Length.IN)
```

This part is new.

```
> wae <- filterD(d, Waterbody.Name=="NELSON LAKE", Survey.Year==2014, mon=="May",
                 Species=="WALLEYE", Gender!="U") %>%
  filterD(Length.or.Lower.Length.IN>11.5, Length.or.Lower.Length.IN<21)
>
> waeF <- filterD(wae, Gender=="F")
> waeM <- filterD(wae, Gender=="M")
```

Construct an Age-Length Key – Males

```
> waeM.aged <- filterD(waeM, !is.na(Age..observed.annuli.))
> waeM.aged$Age..observed.annuli.
[1] 3 2 3 4 3 3 3 3 4 4 3 5 4 4 4 5 5 5 5 4 4 5 4 4 5 4 6 4 4 4 5 5 5 4 5 4 5 4 6 5 5 5 6 6 6 7 6 8
[49] 7 5 6 7 7 9
```

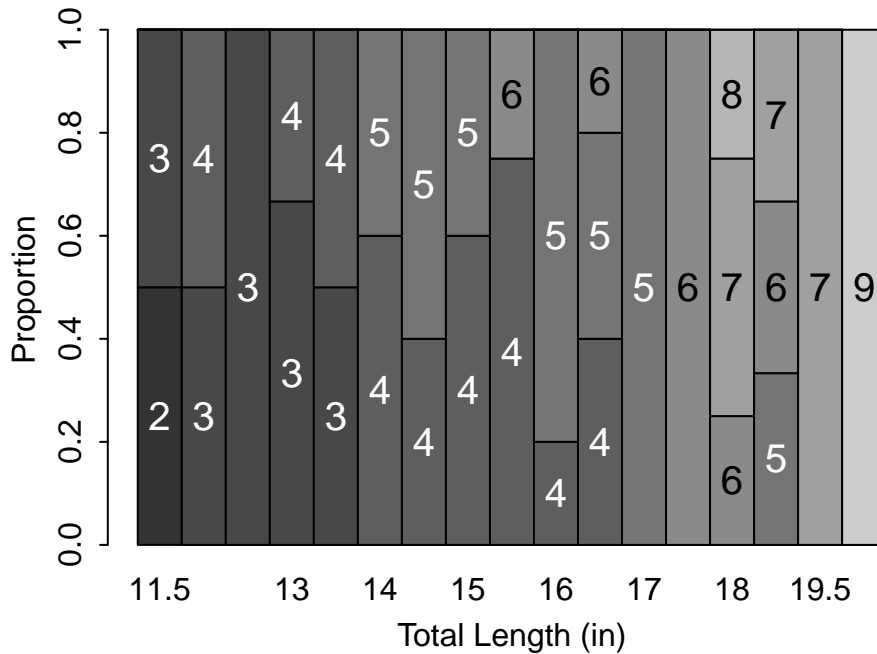
```

> ( rawM <- xtabs(~lcat+Age..observed.annuli.,data=waeM) )
      Age..observed.annuli.
lcat   2 3 4 5 6 7 8 9
 11.5  1 1 0 0 0 0 0 0
 12    0 1 1 0 0 0 0 0
 12.5  0 2 0 0 0 0 0 0
 13    0 2 1 0 0 0 0 0
 13.5  0 1 1 0 0 0 0 0
 14    0 0 3 2 0 0 0 0
 14.5  0 0 2 3 0 0 0 0
 15    0 0 3 2 0 0 0 0
 15.5  0 0 3 0 1 0 0 0
 16    0 0 1 4 0 0 0 0
 16.5  0 0 2 2 1 0 0 0
 17    0 0 0 2 0 0 0 0
 17.5  0 0 0 0 3 0 0 0
 18    0 0 0 0 1 2 1 0
 18.5  0 0 0 1 1 1 0 0
 19.5  0 0 0 0 0 1 0 0
 20.5  0 0 0 0 0 0 0 1

> alkM1 <- prop.table(rawM,margin=1)
> print(alkM1,digits=2,zero.print="-")                                     # for display only
      Age..observed.annuli.
lcat   2    3    4    5    6    7    8    9
 11.5  0.50 0.50  -    -    -    -    -    -
 12    - 0.50 0.50  -    -    -    -    -
 12.5  - 1.00  -    -    -    -    -    -
 13    - 0.67 0.33  -    -    -    -    -
 13.5  - 0.50 0.50  -    -    -    -    -
 14    -  - 0.60 0.40  -    -    -    -
 14.5  -  - 0.40 0.60  -    -    -    -
 15    -  - 0.60 0.40  -    -    -    -
 15.5  -  - 0.75  - 0.25  -    -    -
 16    -  - 0.20 0.80  -    -    -    -
 16.5  -  - 0.40 0.40 0.20  -    -    -
 17    -  -  - 1.00  -    -    -    -
 17.5  -  -  -  - 1.00  -    -    -
 18    -  -  -  - 0.25 0.50 0.25  -
 18.5  -  -  - 0.33 0.33 0.33  -    -
 19.5  -  -  -  -  - 1.00  -    -
 20.5  -  -  -  -  -  -  - 1.00

```

```
> alkPlot(alkM1,pal="gray",xlab="Total Length (in)")
```



Apply an Age-Length Key – Males

```
> waeM.unaged <- filter(waeM,is.na(Age..observed.annuli.))
> waeM.unaged <- alkIndivAge(alkM1,Age..observed.annuli.~Length.or.Lower.Length.IN,data=waeM.unaged)
> waeM.fnl <- rbind(waeM.aged,waeM.unaged)
> waeM.fnl$Age..observed.annuli.
[1] 3 2 3 4 3 3 3 3 4 4 3 5 4 4 4 5 5 5 5 4 4 5 4 4 5 4 6 4 4 4 5 5 5 4 5 4 5 4 6 5 5 5 6 6 6 7 6
[48] 8 7 5 6 7 7 9 3 3 4 3 3 3 3 3 4 3 4 4 4 4 3 3 3 5 4 5 5 4 4 4 5 4 4 4 4 4 5 5 5 4 5 5 4 5 5 5
[95] 5 5 4 4 4 4 5 4 5 4 5 5 5 4 4 5 5 4 4 4 4 5 5 5 4 4 4 5 5 4 4 4 6 4 4 6 4 4 4 4 6 4 4 4 4 4
[142] 4 4 6 6 6 6 4 4 4 6 4 4 4 4 4 5 5 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 6 6
[189] 6 6 7 7 7 9 9
```

Construct and Aply an Age-Length Key – Females

Code is in the script. However, it is mostly a copy-and-paste of the code from above with the 'M's changed to 'F's

Application Assignment

Create a script that performs the following tasks:

1. Load your FM data into R.
2. Filter your to a waterbody, species, year, and sampling date such that some sampled fish were aged and some were not. Perhaps, also filter by sex.
3. Construct an **appropriate** age-length key (ALK) from the aged fish. Visualize the ALK in both tabular and graphical form.
4. Apply the ALK to the unaged fish.
5. Combine the fish aged from a structure and those with ages estimated from the ALK to form a combined data.frame.
6. Confirm that all fish in the final data.frame have assigned ages.
7. (*Time Permitting*) Repeat the above for the other sex or another species.

Save your script!