Module 3 Assignment 3

Ellen Bledsoe

2023-11-13

Distance Sampling

Assignment Details

Purpose

The goal of this assignment is to understand, apply, and interpret detection probability in estimating densities through distance sampling.

Task

Write R code to successfully answer each question below or write text to successfully answer the question.

Criteria for Success

- Code is within the provided code chunks
- Code chunks run without errors
- Code produces the correct result
  - Code attempts will get half points
  - Code that produces the correct answer will receive full points
- Text answers correctly address the question asked

Due Date

April 24 at 2pm MST

Assignment Questions

All questions are worth 1 point unless otherwise specified.

Resources

In addition to the slides and practice code, the following might be helpful:

- Information on unmarked
- User's manual
- Using unmarked with distance data

Set-Up

Some colleagues went out and did some distance sampling! They want to know if “young” saguaros (< 2 meters tall) or ocotillos are found in higher densities in Saguaro National Park. They need us to get the density estimates for them to answer their question.

First things first, let’s load the `unmarked` package.

```r
library(unmarked)
```

Let’s also read in our data.

```r
saguaro <- read.csv("../data_raw/saguaro.csv")
ocotillo <- read.csv("../data_raw/ocotillo.csv")
```

Saguaros (< 2m)

We will use the saguaro data for our first analyses.

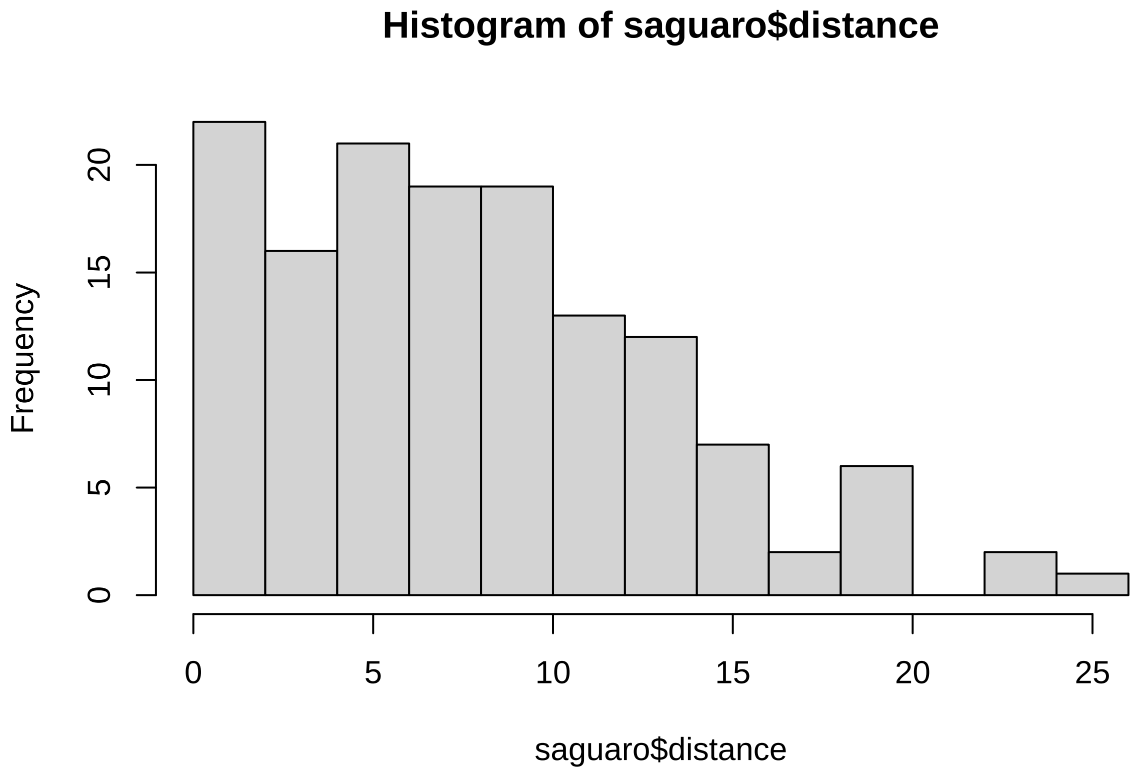
Data Preparation

1. Make a data frame with transect length (i.e., survey effort). Call this `transect_saguaro`. There are 10 groups.

```r
transect_saguaro <- data.frame(group_id = 1:10,
                               length_m = mean(saguaro$length))
```

2. Plot the distribution of detection distances. Does is look how you expected?

```r
hist(saguaro$distance)
```



3. Create an object called `trunc` that sets the truncation distance to 20m to eliminate extreme observations.

   Using that `trunc` value, create a vector called `distance_bins` to set "cut points" for distance bins; for the saguaros data, let's do every 4 meters. Be sure to adjust the arguments accordingly!

```
trunc <- 20
distance_bins <- seq(0, trunc, by = 4)
```

4. Use unmarked function `formatDistData` to count the number of detections in each distance bin for each transect. Save this as an object called `sag_data`. Ignore the *warning* message (but if you get an *error* message, that is different!).

```
sag_data <- formatDistData(saguaro,
                           distCol = "distance",
                           transectNameCol = "group",
                           dist.breaks = distance_bins)
```

```
## Warning in formatDistData(saguaro, distCol = "distance", transectNameCol =
## "group", : The transects were converted to a factor
```

5. Assemble data into the format required by `unmarked`. Call this unmarked frame `UMF_sag`.

```r
UMF_sag <- unmarkedFrameDS(y = as.matrix(sag_data),
                           survey = "line",
                           tlength = transect_saguaro$length,
                           dist.breaks = distance_bins,
                           unitsIn = "m")
```

Models

6. Use the `distsamp` function to create 4 models: half-normal (HN), hazard rate (HR), uniform (Unif), and negative exponential (Exp).

```r
HN_sag   <- distsamp(~1 ~1, UMF_sag, keyfun = "halfnorm", output = "density", unitsOut = "ha")
HR_sag   <- distsamp(~1 ~1, UMF_sag, keyfun = "hazard", output = "density", unitsOut = "ha")
Unif_sag <- distsamp(~1 ~1, UMF_sag, keyfun = "uniform", output = "density", unitsOut = "ha")
Exp_sag  <- distsamp(~1 ~1, UMF_sag, keyfun = "exp", output = "density", unitsOut = "ha")
```

7. Assemble models into a list to compare their AIC values. Identify the top model from this list (HN or HR or Unif or Exp), which you'll use below.

```r
models <- fitList('Half Normal' = HN_sag,
                  'Hazard Rate' = HR_sag,
                  'Uniform'     = Unif_sag,
                  'Exponential' = Exp_sag)
modSel(models)
```

```
##             nPars    AIC delta   AICwt cumltvWt
## Half Normal     2 196.66  0.00 5.3e-01     0.53
## Hazard Rate     3 196.93  0.27 4.7e-01     1.00
## Exponential     2 226.19 29.53 2.1e-07     1.00
## Uniform         1 226.19 29.53 2.1e-07     1.00
```

8. Which model should we use? Why?

   Answer: Half-normal, lowest AIC value

Parameter Estimates

9 . Use the `backTransform()` function to calculate the density estimate. On the next line, calculate the 95% confidence intervals.

```r
backTransform(HN_sag, type = "state") # Density estimate (no./ha)
```

```
## Backtransformed linear combination(s) of Density estimate(s)
##
##  Estimate   SE LinComb (Intercept)
##      47.8 5.46    3.87           1
##
## Transformation: exp
```

```r
exp(confint(HN_sag, type = "state"))  # CI for density
```

```
##                0.025    0.975
## lam(Int) 38.16831 59.76423
```

10. Write 2-3 sentences addressing the following questions. (2 points)

- What does "D hat" represent? Report "D hat" and the confidence intervals around it.
- If we had calculated a density estimate using just our data and not our detection function, would we have under- or overestimated the density of saguaros? Why?

Answer: number of saguaros <2m per hectare; under because we could not account for the ones we missed during sampling

Ocotillos

Now, let's do the same thing for the ocotillo data that they collected.

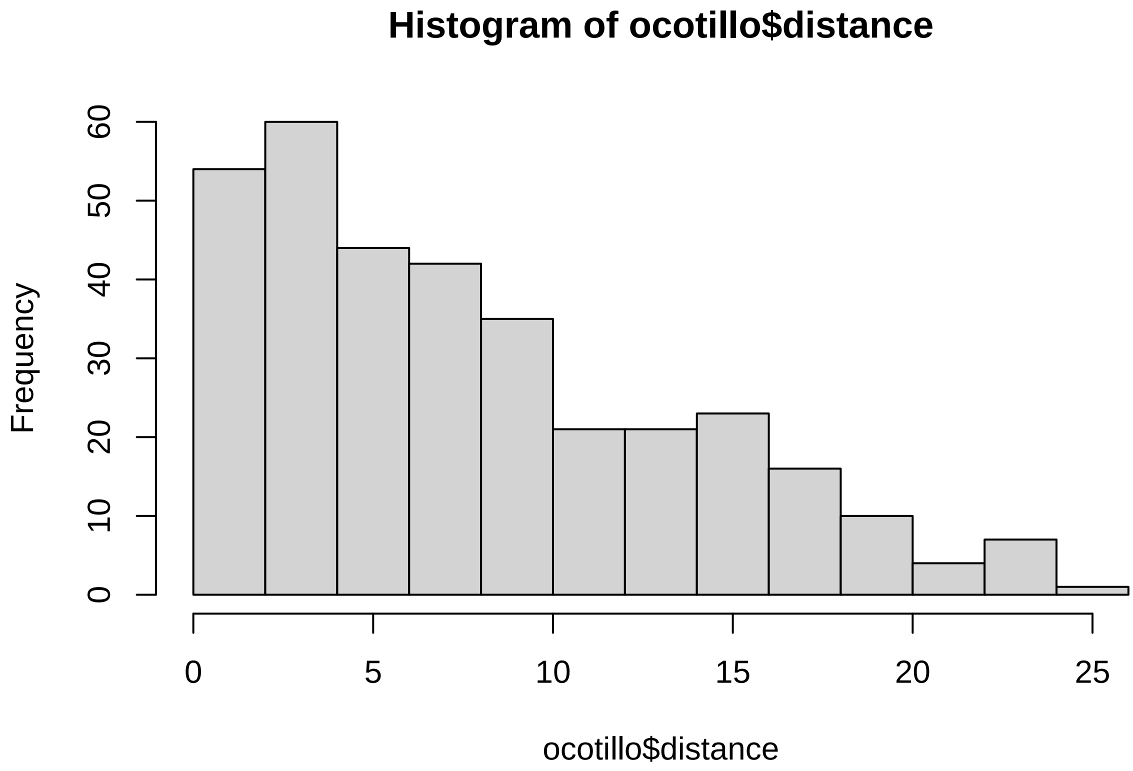
Data Preparation

11. Make your data frame with the transect lengths. Call this `transect_oco`. Again, there were 10 groups.

```r
transect_oco <- data.frame(group_id = 1:10,
                           length_m = mean(ocotillo$length))
```

12. Plot the distribution of detection distances to get an idea of what they look like.

```r
hist(ocotillo$distance)
```



13. This time around, set truncation distance to 25m; save this as an object called `trunc_oco`. Then create your distance vector (`distance_bins_oco`) and set the cut points at every 5 meters.

```
trunc_oco <- 25
distance_bins_oco <- seq(0, trunc_oco, by = 5)
```

14. Use the function `formatDistData` to count the number of detection in each distance bin for each transect. Ignore the warning message. Remember to use the correct data and distance bins.

```
oco_data <- formatDistData(ocotillo,
                           distCol = "distance",
                           transectNameCol = "group",
                           dist.breaks = distance_bins_oco)
```

```
## Warning in formatDistData(ocotillo, distCol = "distance", transectNameCol =
## "group", : The transects were converted to a factor
```

15. Assemble data into the format required by unmarked, called an “unmarked frame.” Remember to refer to the correct data frames and values! Save this as `UMF_oco`

```
UMF_oco <- unmarkedFrameDS(y = as.matrix(oco_data),
                           survey = "line",
                           tlength = transect_oco$length,
                           dist.breaks = distance_bins_oco,
                           unitsIn = "m")
UMF_oco
```

```
## Data frame representation of unmarkedFrame object.
##                  y.1 y.2 y.3 y.4 y.5
## AB JA SK          10   8   2   0   0
## AB, SS, ML        17   9   9   7   4
## CB, CC, MZ         9  17   8  11   5
## EF, LB, YX         4   8   3   3   0
## HA, JB, BD, VM    22  16   6   6   0
## HB, SI, BM         3   7   5   2   0
## JG, AV, MN        17   6   6   4   0
## KH, NT, BS        13  15   2   0   0
## KT, MRD, BC       14   7   2   0   0
## NL, MS, MM        26   7   8   7   3
```

Models

16. Let’s go ahead and use the half-normal model for ocotillos (it also has the lowest AIC value). Write a line of code to fit the half-normal model to the ocotillo data.

```
HN_oco   <- distsamp(~1 ~1, UMF_oco, keyfun = "halfnorm", output = "density", unitsOut = "ha")
```

17. Using the HN_coc model, calculate the density estimate and confidence intervals for ocotillos.

```
backTransform(HN_oco, type = "state") # Density estimate (no./ha)
```

```
## Backtransformed linear combination(s) of Density estimate(s)
##
##  Estimate  SE LinComb (Intercept)
##       108 7.6    4.68           1
##
## Transformation: exp
```

```
exp(confint(HN_oco, type = "state"))  # CI for density
```

```
##              0.025    0.975
## lam(Int) 94.34775 124.2356
```

Compare Densities

We’ve done a lot of calculations just to get to this point!

18. Which has the higher density per hectare: saguaros or ocotillos? How do you know? (2 points)

Answer: based on the d-hat estimates, ocotillos have a higher density