Prey Delivery Analysis

Eliza Stein

11/8/2020

Libraries

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.3
```

```
## -- Attaching packages ---------------------------------------------------------------------------
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------------------------------------------------
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Load data

```
pdOriginal <- read.csv("~/final-project/data/pd_main.csv")


#rename the first column, which imported with a special character
names(pdOriginal)[1] <- "nest"
```

Tidy data

Filter to only include males in B (Missouri) and C (Hayman)

```
pdHayMiss <- pdOriginal %>%
  separate(col = nest, into = c("study_site", "territory"), sep = 1, remove = TRUE) %>%
  filter(study_site == "B" | study_site == "C", sex == "M")
```

Check structure of data

```
str(pdHayMiss) #columns 't180' and 't225' are character classes
```

```
## 'data.frame':    335 obs. of  55 variables:
##  $ study_site       : chr  "B" "B" "B" "B" ...
##  $ territory        : chr  "14_2006" "6_2004" "13_2006" "13_2004" ...
##  $ year             : int  2006 2004 2006 2004 2012 2006 2006 2017 2009 2011 ...
##  $ clutch_size      : chr  "2" "3" "2" "3" ...
##  $ brood_size       : chr  "2" "3" "2" "â‰¥2" ...
##  $ num_fledged      : chr  "0" "3" "0" "â‰¥2" ...
##  $ incubation_start : chr  "6/6/2006" "5/30/2004" "6/5/2006" "5/29/2004" ...
##  $ julian_incubation: int  157 151 156 150 161 149 149 161 151 154 ...
##  $ nest_age         : int  1 2 2 3 3 4 4 4 5 5 ...
##  $ sunset           : int  2023 2019 2023 2019 2026 2020 2020 2027 2022 2023 ...
##  $ sex              : chr  "M" "M" "M" "M" ...
##  $ t15              : int  NA NA NA NA 0 NA NA NA 0 NA ...
##  $ t30              : int  NA 0 NA 0 0 NA NA NA 0 0 ...
##  $ t45              : int  NA 2 NA 0 2 NA NA NA 1 1 ...
##  $ t60              : int  NA 1 NA 0 NA NA NA 1 NA 1 ...
##  $ t75              : int  1 3 NA 0 NA NA 1 4 NA 0 ...
##  $ t90              : int  0 0 NA 0 NA NA 2 2 NA NA ...
##  $ t105             : int  0 0 NA NA 0 NA NA NA NA NA ...
##  $ t120             : int  0 0 1 NA 0 NA NA NA NA NA ...
##  $ t135             : int  0 NA 1 NA NA NA NA 1 NA NA ...
##  $ t150             : int  1 NA 1 NA NA 1 NA NA 0 NA ...
##  $ t165             : int  0 NA 0 NA NA 1 NA NA 0 NA ...
##  $ t180             : chr  "" "" "0" "" ...
##  $ t195             : int  NA NA NA NA NA NA NA NA 0 NA ...
##  $ t210             : int  NA NA NA NA NA NA NA NA 0 NA ...
##  $ t225             : chr  "" "" "" "" ...
##  $ t240             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t255             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t270             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t285             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t300             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t315             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t330             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t345             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t360             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t375             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t390             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t405             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t420             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t435             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t450             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t465             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t480             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t495             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t510             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t525             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ t540             : int  NA NA NA NA NA NA NA NA NA NA ...
##  $ Comments         : chr  "Appears Passive, but notes are somewhat ambiguous" "Passive, pre-incubati
##  $ obs_date         : chr  "6/6/2006" "5/31/2004" "6/6/2006" "5/31/2004" ...
##  $ obs_time         : chr  "2128-2258" "2040-2143" "2210-2315" "2043-2114" ...
```

```
##  $ start_time       : chr  "21:28" "20:40" "22:10" "20:43" ...
##  $ stop_time        : chr  "22:58" "21:43" "23:15" "21:14" ...
##  $ weather          : chr  "breezy 5-10mph" "slight breeze" "~5mph" "<5mph" ...
##  $ fledge_date      : chr  "" "7/14/2004" "" "7/9/2004" ...
##  $ fledge_accuracy  : chr  "Not Stated" "known for 2nd, 1st prbly day before" "w/in 1 day" "known fo
```

```
unique(pdHayMiss$t180) #at least one cell has an asterisk after the value
```

```
## [1] ""   "0"  "1"  "4*" "9"  "2"  "3"  "5"
```

```
unique(pdHayMiss$t225) #same here
```

```
## [1] ""   "6*" "3"  "2"  "0"  "1"  "7"
```

```
#remove asterisks
pdClean <- pdHayMiss %>%
  mutate(t180 = gsub("\\*", "", t180)) %>%
  mutate(t225 = gsub("\\*", "", t225))

#change these columns to integers
pdClean$t180 <- as.integer(pdClean$t180)
pdClean$t225 <- as.integer(pdClean$t225)
```

Get mean number of prey deliveries for each study site, by time interval

```
#create independet dfs for Hayman and Missouri. (since the last observation for Hayman is at t240, we'l
pdHay <- pdClean %>%
  filter(study_site == "C") %>%
  select(t15:t240)
pdMiss <- pdClean %>%
  filter(study_site =="B") %>%
  select(t15:t240)

#change column names to remove "t" in front of time interval
colnames(pdHay) <- c("15", "30", "45", "60", "75", "90", "105", "120", "135", "150", "165", "180", "195
colnames(pdMiss) <- c("15", "30", "45", "60", "75", "90", "105", "120", "135", "150", "165", "180", "19

#make new data frames with column means for each time interval
meanHay <- data.frame(Hayman = colMeans(pdHay, na.rm = TRUE))
meanMiss <- data.frame(Missouri = colMeans(pdMiss, na.rm = TRUE))



#create new transposed df with means for Hayman and Missouri
wideMeanPD <- as.matrix(t(cbind(meanHay, meanMiss)))
wideMeanPD
```

```
##                 15       30       45       60        75        90      105
## Hayman   0.4285714 0.826087 1.310345 1.269231 0.9583333 0.8636364 0.900000
## Missouri 0.7600000 1.496644 2.382550 2.013986 1.7540984 0.9591837 0.989011
##                120       135       150       165       180       195 210
```

```
## Hayman   0.5882353 0.4705882 1.0000000 1.0666667 1.0909091 2.5000000 2.0
## Missouri 0.7297297 0.9076923 0.7818182 0.6046512 0.8947368 0.7575758 0.8
##                225 240
## Hayman   1.571429 1.0
## Missouri 1.250000 1.2
```

Below is a data.frame of pds for Hayman and Missouri, which can be grouped by "study_site"

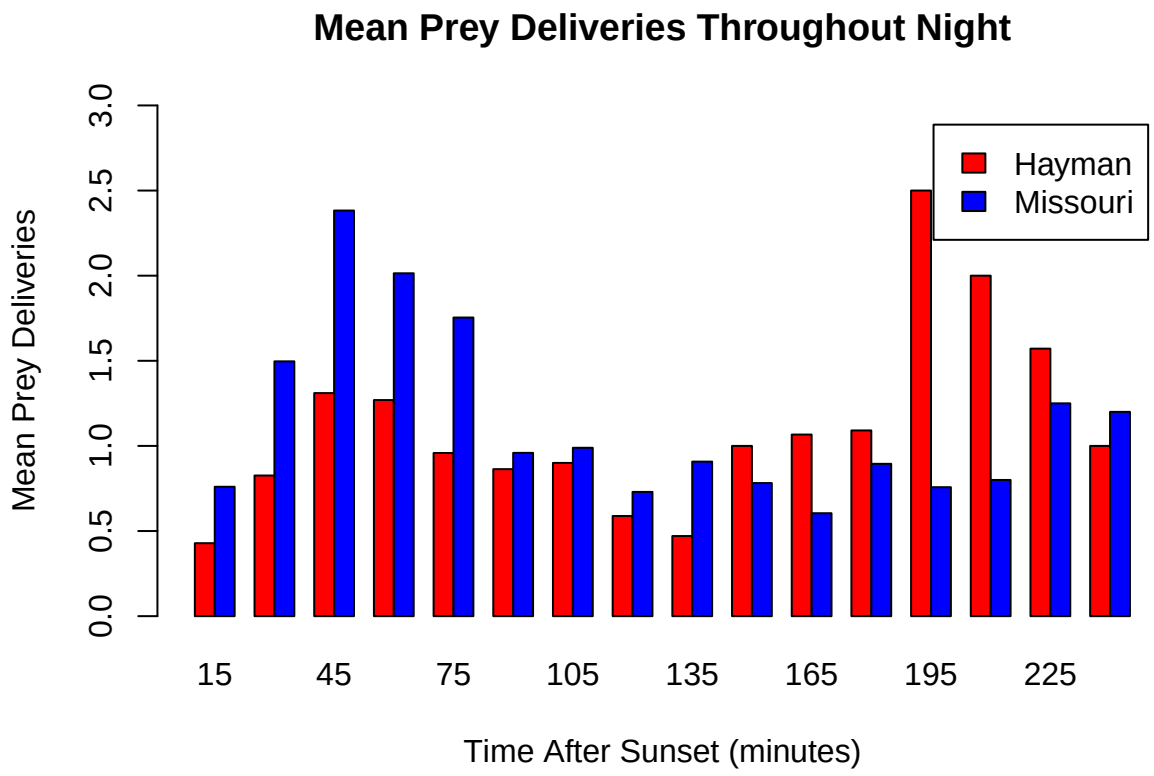
```
#bind only time observations for each study site
bindHay <- cbind(study_site = "Hayman", pdHay)
bindMiss <- cbind(study_site = "Missouri", pdMiss)

pdHayMiss <- rbind(bindHay, bindMiss)
```

Plot mean prey deliveries

Barplot:

```
meanPlot <- barplot(
  wideMeanPD,
  beside = TRUE,
  legend.text = TRUE,
  col = c("red", "blue"),
  ylim = c(0, 3),
  xlab = "Time After Sunset (minutes)",
  ylab = "Mean Prey Deliveries",
  main = "Mean Prey Deliveries Throughout Night"
)
```



Calculate confidence intervals

Scott's code:

Here's a start...

```
#vector with Hayman PDs for t=15
vec <- pdHay$'15'[!is.na(pdHay$'15')]
vec
```

```
##  [1] 0 2 0 0 0 0 0 2 1 0 0 1 0 1 0 0 0 0 0 0 2
```

```
sample(vec, length(vec), replace = TRUE)
```

```
##  [1] 0 0 0 0 0 0 0 0 1 0 1 0 0 0 1 0 0 2 1 1 0
```

Eventually will append CIs to meanHay and meanMiss:

```
longMeanPD <- as.matrix(cbind(meanHay, meanMiss))
longMeanPD
```

```
##        Hayman  Missouri
## 15  0.4285714 0.7600000
```

```
## 30  0.8260870 1.4966443
## 45  1.3103448 2.3825503
## 60  1.2692308 2.0139860
## 75  0.9583333 1.7540984
## 90  0.8636364 0.9591837
## 105 0.9000000 0.9890110
## 120 0.5882353 0.7297297
## 135 0.4705882 0.9076923
## 150 1.0000000 0.7818182
## 165 1.0666667 0.6046512
## 180 1.0909091 0.8947368
## 195 2.5000000 0.7575758
## 210 2.0000000 0.8000000
## 225 1.5714286 1.2500000
## 240 1.0000000 1.2000000
```