Physiology of Mt. Graham Red Squirrels

Ahyoung Amy Kim, Haozhe Xu,Samir Rachid Zaim, and Elmira Torabzadehkhorasani

Contents

1 Executive Summary

1.1 Study Summary

The Mt. Graham red squirrel project is a study to better understand the reproductive activity and cycle of these squirrels. Since these squirrels are unique to Mt. Graham, gaining a better understanding of their reproductive patterns can better inform conservation strategies to help increase their numbers. The strategy is to study their reproductive behavior by analyzing and understanding the behavior of their physiological markers, identify potential ovulation events, and conduct data and statistical analyses. The working hypothesis is that Mt. Graham squirrels have spontaneous ovulation, rather than induced ovulation, and this can be observed physiologically through a coupled process of observing increased estradiol, followed by increases in progesterone. Formally, we conducted our statistical analysis as follows:

1. Identify ovulation events via simultaneous or paired increases in Estradiol and Progesterone
2. Model time between ovulation as an exponential distribution, and calculate empirical distribution
3. Formally test whether they follow an exponential process via Kolmogorov-Smirnoff Test

1.2 Findings

Based on our analyses, there seems to be visual and statistical evidence supporting the idea that Mt. Graham Red Squirrels follow spontaneous ovulation, and that they follow a cyclical reproductive pattern across the measurements observed, both in 2015 and 2016.

Table 1: Peak Distribution Across Years

	Total_Peaks	Mean_Time	Median_Time	Std_Deviation	P_value
2015	20	7.00	4.0	12.78	0.135
2016	21	5.85	6.5	3.10	0.070

Table 1 displays the summaries for ovulation events between 2015 and 2016. The p-value in the last column denotes the probability that the wait-time between ovulation events follows a true exponential distribution, with a p-value > 0.05, providing statistical evidence and support for this claim.

The current analysis defined

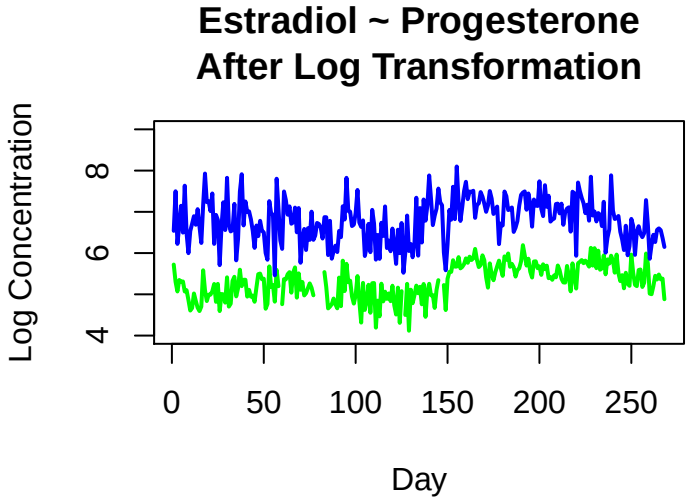
$$\text{Ovulation event} = \begin{cases} 1 & \Delta_{days} \leq X \\ 0 & o.w. \end{cases}$$

where Δ_{days} is the time difference between the first estradiol peak and the subsequent progesterone peak (in days), and X is a positive integer counting the days between peaks. The default, is to let X = 0, so that these events only capture "same-day" peaks. We recommend re-running the analyses allowing for a 1-day and 2-day lag to compare and determine their physiological validity. We wrote R functions that we will re-run these same exact analyses, by simply changing the parameter *lag* in the function-call.

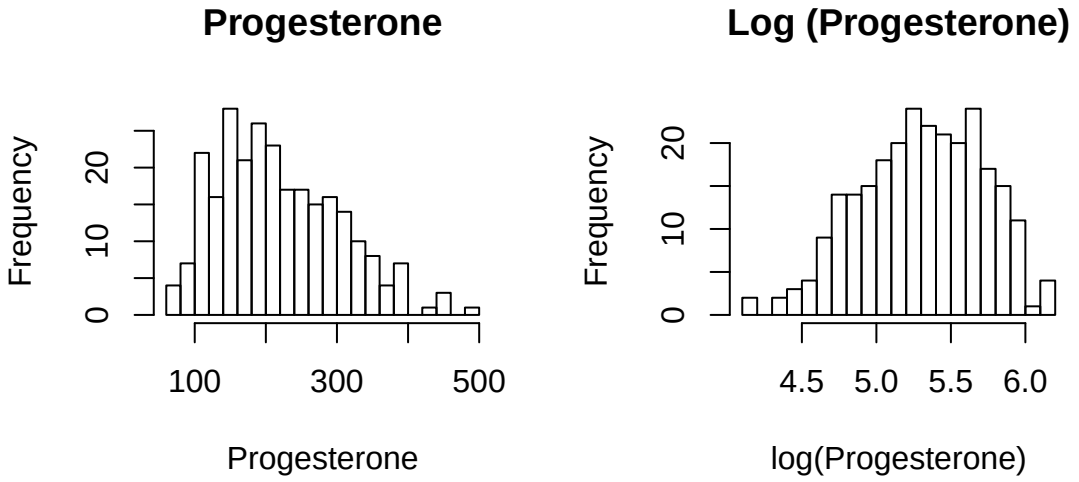
2 Data Exploration and Statistical Analysis

2.1 Log Transformations

In visually analyzing data that operate on different scales - the different magnitudes makes it difficult to visually interpret because of how tight one measurement is distributed compared to another. One remedy to fix is to log- or power-transform the data to shrink the scale of the observed variance. The plot below shows the side-by-side progesterone and estradiol daily measurements, after log-transformations.



Furthermore, the advantage of log- and power-transformations is that they can sometimes load normalize the data, which can facilitate further statistical analyses.



For all subsequent analyses in this report, we have transformed all concentrations using a log-transformation.

2.2 Defining and Identifying Ovulation Events

In order to detect whether spontaneous ovulation occurs in Mount Grahama Red Squirrels, we have to determine physiologically whether we observe increases in estradiol that are followed by increases in progesterone. To do this, we wrote a function in R called "localMaxima" that identifies the local peaks of these concentration curves, and then we identify candidate "prepregnancy events" based on whether these peaks are aligned chronologically. Algorithmically we define an event as follows:

$$\text{Ovulation event} = \begin{cases} 1 & \Delta_{days} \leq X \\ 0 & o.w. \end{cases}$$

where Δ_{days} is the time difference between peaks (in days), and X is a positive integer counting the days between peaks. The default, is to let X = 0, so that these events only capture "same-day" peaks. We parameterized the function with an argument called "lag" to allow for a 1-2 day flexible window to conduct the analysis.We developed two visual tools to evaluate graphically whether we are capturing biologically meaningful events or simply noise, by comparing the yearly data, and then subsetting and analzying the data month by month.

For example, to first evaluate all the candidate points we identified, we plot the progesterone and estradiol using line plots, and annotate the graph with the candidate ovulation events. Then, we can use this information to subset the data in a month by month analysis, for further examination.

2.3 Hypothesis Testing: Do waiting times between ovulation event follow an exponential distribution?

Using the visual tools from above, there seems to be some informal visual evidence of regular, periodic candidate ovulation events. To support this visual evidence, we can conduct hypothesis tests measuring how closely these observed data follow a theoretical exponential distribution, a probability distribution that measures time between events. The hypothesis would be as follows:

$$\text{Hypothesis Test} = \begin{cases} H_0 & Ovul \sim Exp(\lambda) \\ H_A & Ovul \perp Exp(\lambda) \end{cases}$$

where the null hypothesis H_0 assumes that the observed ovulation events follow an exponential distribution with some rate parameter λ, and the alternative suggests that it does not. This means, that contrary to most

hypothesis tests, the higher the p-value, the greater evidence there is to support that Mount Graham Red Squirrels follow a regular spontaneous ovulation cycle. To formally test the hypothesis test above we:

- Fit the observed data to an exponential distribution, and calculate an empirical $\hat{\lambda}$
- Use the Kolmogorov-Smirnoff test to formally test how close the observed distribution matches the theoretical distribution.

The Kolmogorov-Smirnoff (KS) test used to compares two data distributions and provides a p-value that tests how likely these two distributions come from the same distribution. The test is based on the cumulative distribution function (CDF) of each data set. It is useful because it does not require any distributional assumptions such as the normal distribution assumption. Although the KS test does not require the normality assumption, the analyses were still conducted based on the log-transformed data. The null and alternative hypotheses for the KS test are the followings:

H_0: The true distribution of the Mt. Graham Squirrel Ovulation follows an exponential distribution, with regular wait-times between events.

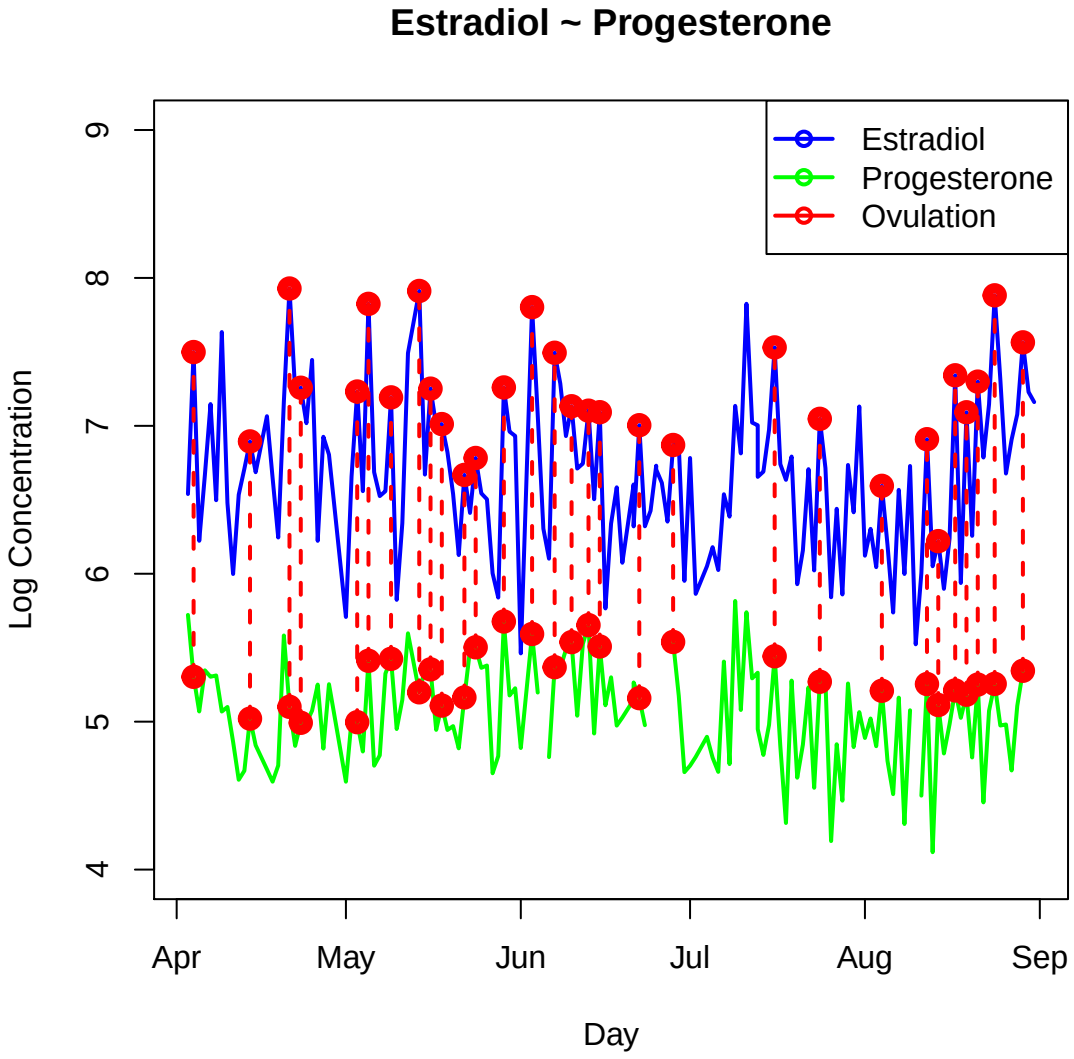
H_1: The true distribution of the Mt. Graham Squirrel Ovulation does not follow an exponential distribution, with irregular wait-times between events.

3 Results

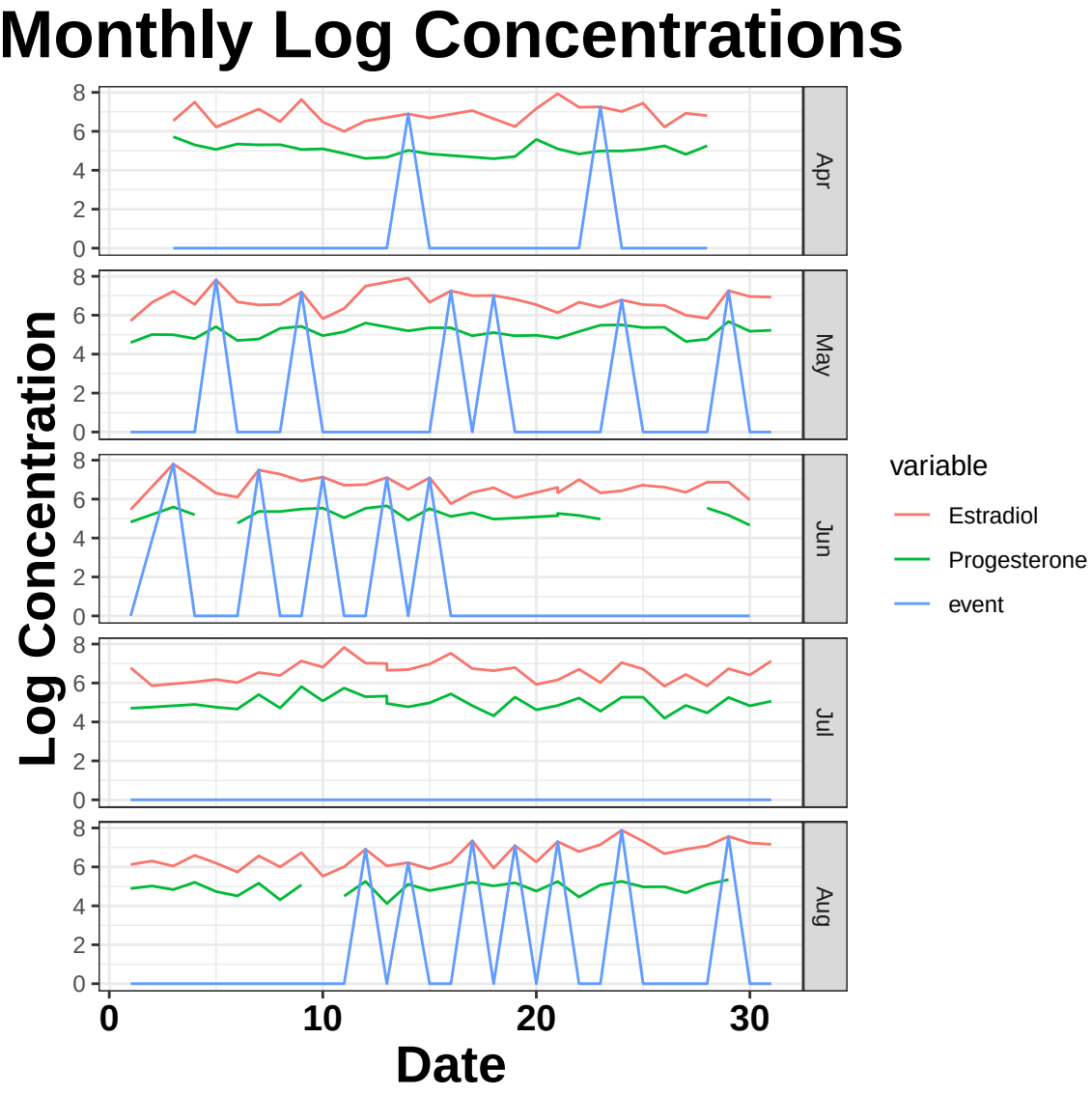
3.1 2015

3.1.1 Identifying Candidate Ovulation Events

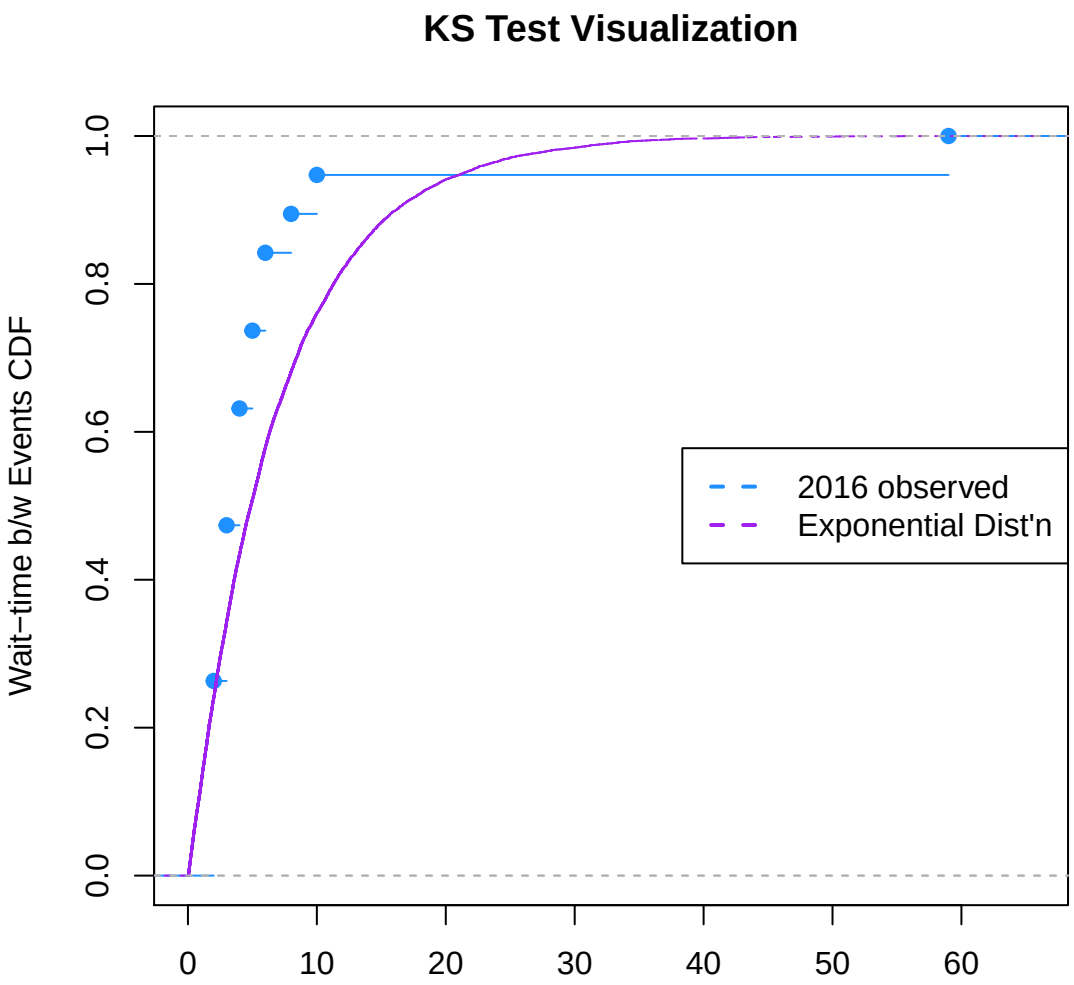
The line plot below shows the identified candidate ovulation events for the observations for 2015.


Estradiol ~ Progesterone
Estradiol
Progesterone
Ovulation
Log Concentration
4
5
6
7
8
9
Apr
May
Jun
Jul
Aug
Sep
Day

3.1.2 Monthly Examination and KS Test



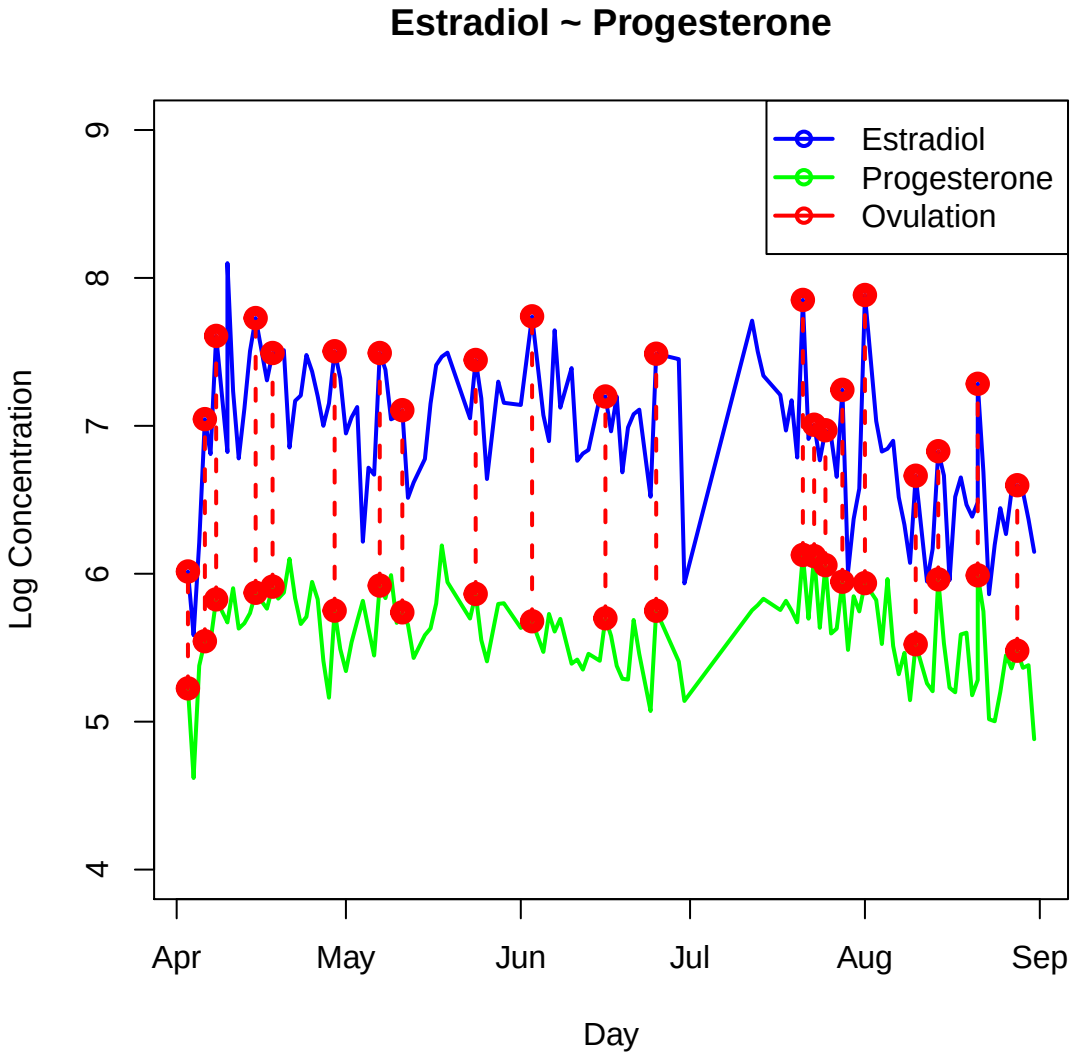

```
##
## To determine the strength of the distribution of the data we run a KS test
## from fitting an exponential distribution to the observed data
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  time_between_peaks
## D = 0.26648, p-value = 0.1346
## alternative hypothesis: two-sided
##
## The p-value > 0.05 suggests that the spontaneous ovulation events
## are occuring at regular intervals following an exponential distribution
## with rate parameter = 0.14
```



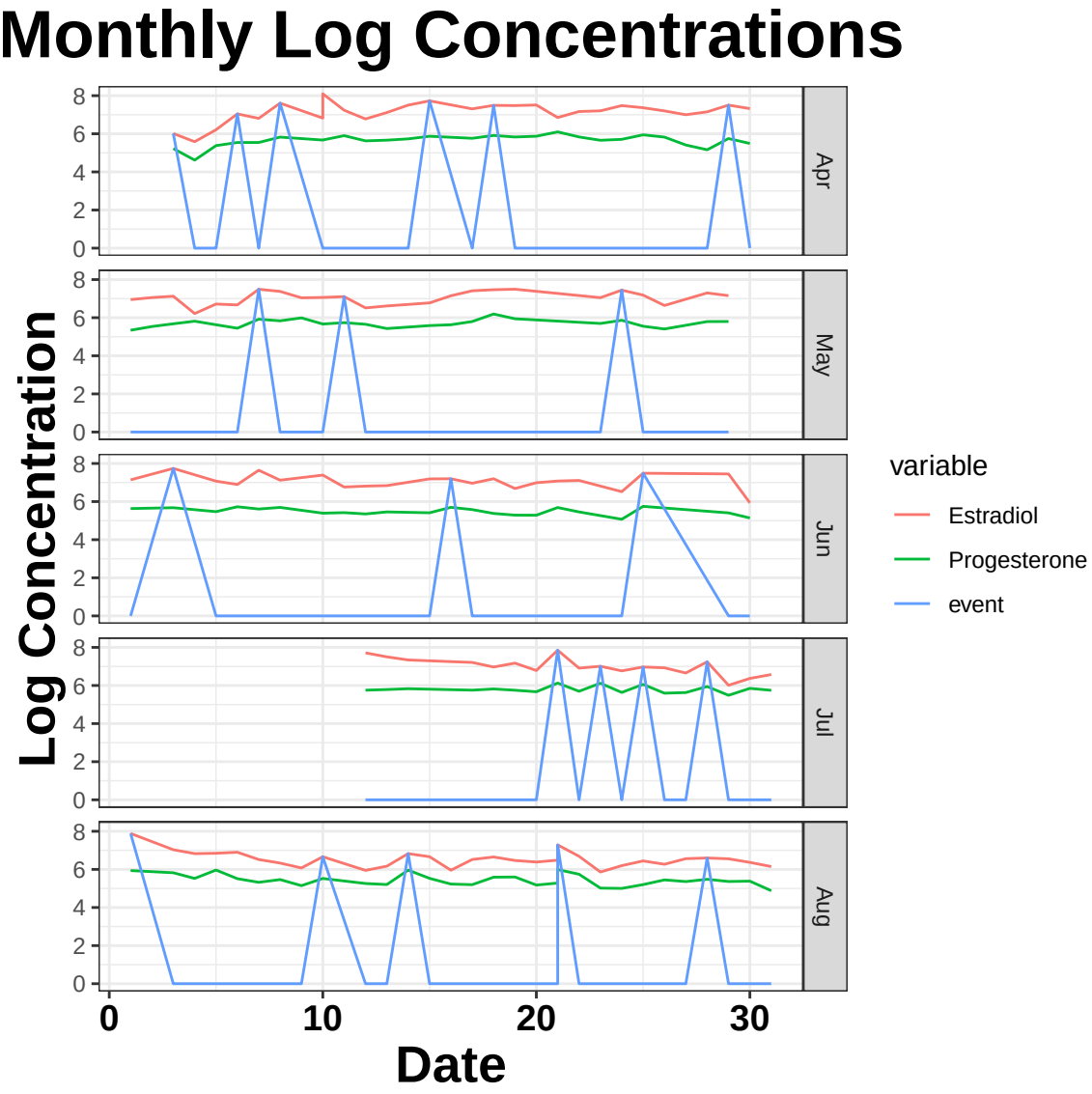
3.2 2016

3.2.1 Identifying Candidate Ovulation Events

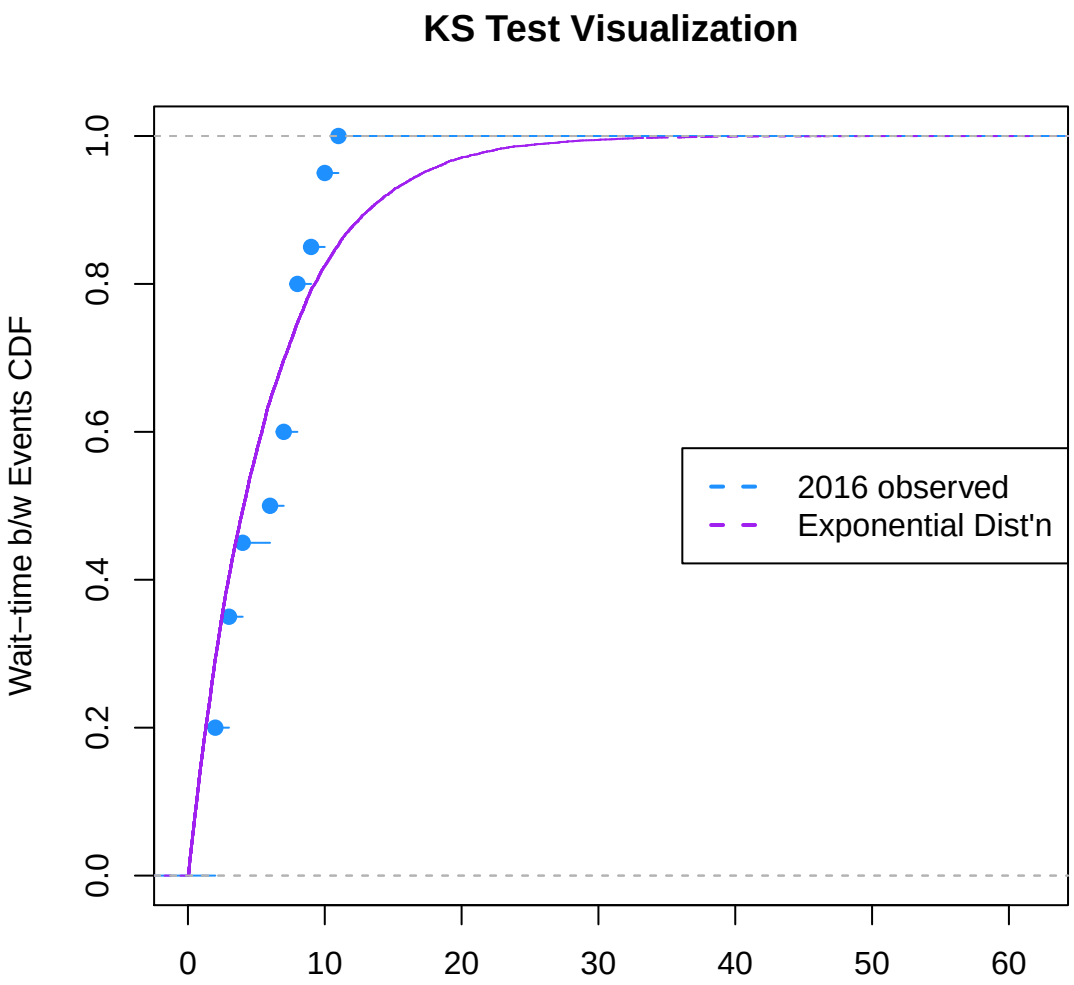
The line plot below shows the identified candidate ovulation events for the observations for 2016.


Estradiol ~ Progesterone
Estradiol
Progesterone
Ovulation
Log Concentration
4
5
6
7
8
9
Apr
May
Jun
Jul
Aug
Sep
Day

3.2.2 Monthly Examination and KS Test




```
##
## To determine the strength of the distribution of the data we run a KS test
## from fitting an exponential distribution to the observed data
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  time_between_peaks
## D = 0.28957, p-value = 0.06989
## alternative hypothesis: two-sided
##
## The p-value > 0.05 suggests that the spontaneous ovulation events
## are occuring at regular intervals following an exponential distribution
## with rate parameter = 0.17
```



4 Appendix

4.1 Note on KS Test

The KS test provides valid probabilities when one compares data to a pre-specified probability distribution. In our case, that would be comparing the waiting times between events to an exponential distribution with a known, fixed mean λ parameter (say, $\lambda = \frac{1}{7}$ which is 7-days between events). However, in our analysis, we estimate the λ parameter from the observed data in Table 1. This paper [1] goes into a bit more technical detail and compares it to the χ^2 goodness of fit test. We recommend conducting additional statistical tests comparing observed data to distributions (i.e., χ^2 goodness of fit) to confirm the results as a way to add more statistical rigor to the results.

4.2 References

1.) Lilliefors, Hubert W. "On the Kolmogorov-Smirnov test for the exponential distribution with mean unknown." Journal of the American Statistical Association 64, no. 325 (1969): 387-389.

4.3 R code

```
knitr::opts_chunk$set(echo = FALSE, warning=FALSE, message=FALSE)
### Load and clean Data
```

```r
require(data.table)
require(lubridate)
require(ggplot2)
setwd("~/Desktop/classes/fall 2019/Statistical Consulting/red squirrel")
squirrel <- fread('sphys_ra528 (1).csv',data.table = F)
squirrel$date <- as.Date(squirrel$date,format = '%m/%d/%Y')
squirrel$year <- year(squirrel$date)
squirrel$months <- month(squirrel$date, label = T)
squirrel$day <- day(squirrel$date)
names(squirrel)[c(4,6)] <- c('Estradiol','Progesterone')

plot(log(squirrel$Progesterone), type='l', ylim=c(4,9), lwd=2, col='green',
     main='Estradiol ~ Progesterone\nAfter Log Transformation', xlab='Day',ylab='Log Concentration')
lines(log(squirrel$Estradiol), type='l', lwd=2, col='blue')

par(mfrow=c(1,2))
hist(squirrel$Progesterone, main='Progesterone', xlab='Progesterone', 20)
hist(log(squirrel$Progesterone), main='Log (Progesterone)', xlab='log(Progesterone)', 20)

squirrel$Estradiol <- log(squirrel$Estradiol)
squirrel$Progesterone <- log(squirrel$Progesterone)

### Split year to year
m1 = squirrel[squirrel$year == 2015, -5]
m2 = squirrel[squirrel$year == 2016, -5]


###################################
## Functions to analyze Data
###################################

###################################
## localMaxima : identifies peaks
##               in the data
###################################

localMaxima <- function(x) {
  # Use -Inf instead if x is numeric (non-integer)
  y <- diff(c(-.Machine$integer.max, x)) > 0L
  rle(y)$lengths
  y <- cumsum(rle(y)$lengths)
  y <- y[seq.int(1L, length(y), 2L)]
  if (x[[1]] == x[[2]]) {
    y <- y[-1]
  }
  y
}

###################################
## analyze_yearly_trends :
## - compares yearly estradiol
##   and progesterone values
## - plots peaks and events
```

```
###################################


analyze_yearly_trends <- function(m1, lag=0){
  ### Detect events
  estra_max <- numeric(nrow(m1))
  estra_max[localMaxima(m1$Estradiol)] <- 1

  proge_max <- numeric(nrow(m1))
  proge_max[localMaxima(m1$Progesterone)] <- 1

  event <- numeric(nrow(m1))
  for(i in 1:length(event)){
    if(estra_max[i]==1 & proge_max[i+lag]==1){
      event[i] <- 1
    } else if(estra_max[i]==1 & proge_max[i]==1){
      event[i] <- 1
    }
  }

  max_mat <- data.frame(estra_max, proge_max, event)

  ### Plot all events
  plot(m1$date, m1$Progesterone, type='l', ylim=c(4,9), lwd=2, col='green',
       main='Estradiol ~ Progesterone', xlab='Day',ylab='Log Concentration')
  lines(m1$date, m1$Estradiol, type='l', lwd=2, col='blue')
  event <- m1$Estradiol * max_mat$event
  event2 <- m1$Progesterone * max_mat$event

  points(m1$date, event, col='red', lwd=5)
  points(m1$date, event2, col='red',lwd=5)

  segments(x0 =m1$date , y0 = event2, x1 = m1$date, y1=event, lwd=2, lty=2, col='red' )
  legend('topright', c('Estradiol','Progesterone','Ovulation'),
         col=c('blue','green','red'), lwd=2, pch=c(1,1,1))

}

###################################
## analyze_monthly_trends :
## - compares monthly estradiol
##   and progesterone values
## - plots peaks and events
###################################

analyze_monthly_trends <- function(m1, lag=0){
   ### Detect events
  estra_max <- numeric(nrow(m1))
  estra_max[localMaxima(m1$Estradiol)] <- 1

  proge_max <- numeric(nrow(m1))
  proge_max[localMaxima(m1$Progesterone)] <- 1
```

```
  event <- numeric(nrow(m1))
  for(i in 1:length(event)){
    if(estra_max[i]==1 & proge_max[i+lag]==1){
      event[i] <- 1
    } else if(estra_max[i]==1 & proge_max[i]==1){
      event[i] <- 1
    }
  }
  event <- m1$Estradiol * event


  max_mat <- data.frame(estra_max, proge_max, event)

  ### Do monthly panel plots
  m1$event <- event
  newM1 = melt(m1,id.vars = c('id','day','date','months','year'))
  p = ggplot(newM1, aes(x = day, value, col=variable)) + geom_line() + facet_grid(months~.) +
    theme_bw() +    theme(
    plot.title = element_text(color="black", size=26, face="bold", hjust = 0.5),
    axis.title.x = element_text(color="black", size=20, face="bold"),
    axis.title.y = element_text(color="black", size=20, face="bold"),
    axis.text.x = element_text(color="black", size=14, face="bold")) +
    ggtitle('Monthly Log Concentrations') + xlab('Date') + ylab('Log Concentration')

  print(p)
  ### Look at time between peaks
  time_between_peaks <- diff(which(event > 0 ))
  return(time_between_peaks)

}
require(knitr)

##################################
## summary_peaks :
## - provides tabular summary of
##   peaks
##################################

summary_peaks <- function(time_between_peaks){
  numPeaks <- length(time_between_peaks)+1
  AvgLength<- mean(time_between_peaks)
  AvgStd <- sqrt(var(time_between_peaks))

  summary_stats <- data.frame(Total_Events = numPeaks,
                              Mean_Time = AvgLength,
                              Median_Time = median(time_between_peaks),
                             # Max_Time_Between_Peaks = max(time_between_peaks),
                             # Min_Time_Between_Peaks = min(time_between_peaks),
                              Std_Deviation = AvgStd)

  return(summary_stats)
}
```

```
###################################
## test_between_peaks_distribution :
## - conducts KS test
###################################

test_between_peaks_distribution <- function(time_between_peaks){

  cat('\nTo determine the strength of the distribution of the data we run a KS test
from fitting an exponential distribution to the observed data\n')
  require(MASS)
  fit1 <- fitdistr(time_between_peaks, "exponential")
  KS.res <- ks.test(time_between_peaks, "pexp", fit1$estimate)
  print(KS.res)

  if(KS.res$p.value > 0.05){
    cat(paste('The p-value > 0.05 suggests that the spontaneous ovulation events
are occuring at regular intervals following an exponential distribution
with rate parameter =', round(fit1$estimate,2)))
  } else {
    cat('The p-value < 0.05 suggests that the spontaneous ovulation events are not
occuring at regular intervals, and thus do not follow an exponential distribution
of equal-peak intervals')
  }

  true_Exp <- rexp(10000, fit1$estimate)

plot(ecdf(time_between_peaks), xlim=range(c(time_between_peaks, true_Exp)), col="dodgerblue", main='KS
       ylab=paste("Wait-time b/w Events CDF"), xlab= "")
  plot(ecdf(true_Exp), add=TRUE, lty="dashed", col="purple", ylab="", xlab="")
  legend("right", legend=c("2016 observed", "Exponential Dist'n"), col=c("dodgerblue", "purple"),
         lty="dashed", lwd=2 )
  return(KS.res)
}
analyze_yearly_trends(m1=m1, lag = 2)

a = analyze_monthly_trends(m1=m1, lag = 0)
p1 = test_between_peaks_distribution(a)

analyze_yearly_trends(m1=m2, lag = 0)
b = analyze_monthly_trends(m1=m2, lag = 0)
p2 = test_between_peaks_distribution(b)
```