

Frog Breeding Responses to Climate Change

Shannon Carter

February 19, 2019

Climate change shifts plants' and animals' seasonal schedules, because events like breeding, migration, and metamorphosis are timed by climate cues. Here, I examine this response in populations of frogs in East Texas. 20 years ago, a collaborator in the USDA installed audio recorders in 8 ponds in East Texas. These recorders are programmed to record 6 times daily, capturing the breeding calls of frogs at the ponds. I've analyzed this data in R to measure if/how frog breeding timing is shifting and how this will affect frogs' interactions with each other. I was the lead author on this project, meaning I conceptualized the questions, performed all data management and analysis, and interpreted and wrote up results for publication. This RMarkdown document provides a summary of the analysis for one of two major questions for this project. For more information, please feel free to contact me at skcarter25@gmail.com or see the full publication in Ecology Letters.

Process and tidy raw data

We have 8 raw data files, each representing 1 pond. Each row represents one time point (6 per day from May 2000 through Dec. 2018) and includes the number of calls made by 12 species of frogs (in columns)

##	POND	YEAR	FROGTIME	FROGDATE	HV	HC	BV	BW	RCA	RCL	RS	GC	PC	PT	AC	RP
## 1	1	2000	18:00	5/9/2000	2	0	0	0	0	0	0	0	0	0	0	0
## 2	1	2000	19:00	5/9/2000	3	0	0	0	0	0	0	0	0	0	0	0
## 3	1	2000	20:00	5/9/2000	1	0	0	0	0	0	0	0	0	0	0	0
## 4	1	2000	21:00	5/9/2000	0	0	0	0	0	0	0	0	0	0	0	0
## 5	1	2000	22:00	5/9/2000	0	0	0	0	0	0	0	0	0	0	0	0
## 6	1	2000	23:00	5/9/2000	0	0	0	0	0	0	0	0	0	0	0	0

Here, I merge the data for the ponds, melt to long format, sum the six daily observations, and adjust time & date format.

```
## Stack raw data into one df
p18 <- rbind(p1, p2, p3, p4, p5, p6, p7, p8)

## Change to long format and rename columns
mp18 <- melt(p18, id.vars = c("POND", "YEAR", "FROGTIME", "FROGDATE"),
             measure.vars = c("HV", "HC", "BV", "BW", "RCA", "RCL",
                              "RS", "GC", "PC", "PT", "AC", "RP")) # codes for 12 species
names(mp18) <- c("pond", "year", "time", "date", "sp", "calls")

## Adjust time variable
time <- paste(mp18$date, mp18$time)
time1 <- strptime(time, "%m/%d/%Y %H:%M")
time2 <- as.POSIXlt(time1)
time3 <- as.Date(time2, "%m/%d/%Y %H:%M")
mp18$time <- time3
mp18 <- subset(mp18, select = c("pond", "year", "time", "sp", "calls"))

## Sum all 6 daily observations
sums <- mp18 %>%
  group_by(pond, year, time, sp) %>%
  summarize(dailysum = sum(calls))
```

```
## Clean it up and write it
daily$doy <- as.numeric(strftime(daily$time, format = "%j")) # adding a 'day of year'
daily <- daily[daily$year != 2000 ,] # 2000 is an incomplete year, so we drop it
```

Visualizing the raw data

This custom function allows me to plot the overlapping distributions of any two species in any year and pond. The example plot shows number of calls over time for the gray tree frog (in orange) and the green frog (in blue) in 2007 at pond 4. The major objective of this project was to calculate how the area of overlap (visible in the plot at the area where the blue and orange distributions intersect) changed across years for pairwise combinations of species. This would tell us if climate change-driven shifts in breeding timing influence the competition between species.

```
## The function takes four inputs: year, pond, species 1, and species 2
## I applied the function to all combinations of these parameters
overlapplot <- function(s1, s2, y, p) {

  ## Subset year and pond of interest for each species
  s1 <- subset(daily, subset = (sp == s1 & year == y & pond == p))
  s2 <- subset(daily, subset = (sp == s2 & year == y & pond == p))

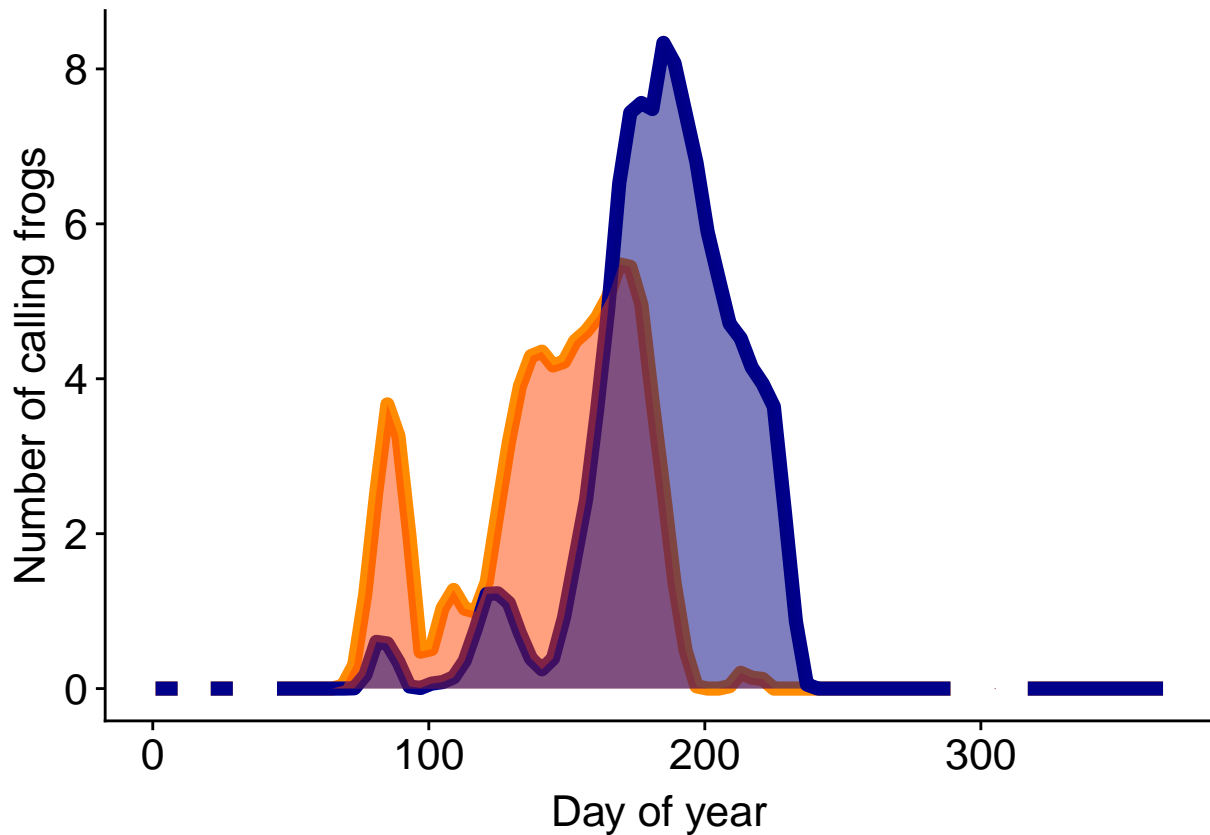
  ## Run lowess functions-- this smoothes scatter plot data into a distribution
  ## f, iter, and delta are parameters that control the degree and method of smoothing
  l1 <- as.data.frame(lowess(s1$dailysum, f = 1/20, iter = 3, delta = 4))
  l2 <- as.data.frame(lowess(s2$dailysum, f = 1/20, iter = 3, delta = 4))

  ## Make a data frame that stores the lowess curves for each species
  l <- data.frame(day = l1$x, s1 = l1$y, s2 = l2$y)

  ## Plot
  p <- ggplot(l, aes(x = day, y = max(s1, s2))) +
    xlab("Day of year") + ylab("Number of calling frogs") +
    geom_line(size = 2.5, colour = "darkorange", aes(x = day, y = s1, stat = "identity", position = "dodge")) +
    geom_line(size = 2.5, colour = "blue4", aes(x = day, y = s2, stat = "identity", position = "dodge")) +
    geom_ribbon(aes(x = day, ymax = s1), ymin = 0, fill = "orangered", alpha = 0.5) +
    geom_ribbon(aes(x = day, ymax = s2), ymin = 0, fill = "navy", alpha = 0.5) +
    theme(panel.border = element_rect(colour = "black", fill = NA, size = 1.5),
          axis.text.y = element_text(size = 16),
          axis.text.x = element_text(size = 16),
          axis.title.y = element_text(size = 16),
          axis.title.x = element_text(size = 16))

  print(p)
}

## Now, we can input any species pair and year to visualize
## This is useful for spotchecking our analysis and providing a visual reference for the workflow
overlapplot("RCL", "HV", 2007, 4)
```



Function to calculate overlap for every species pair

Now, I want to calculate the overlap between each species pair for each year and at each pond. To do this, I built a custom function that 1) subsets the data by species, year, and pond 2) makes loess smoothed distributions of calling activity for each species through the year 3) calculates the integrated area of overlap between the two species. I then apply this function to all 132 pairings of species at each of 8 site and each of 15 years.

```
## The function take 4 inputs: year, pond, species 1 and species 2
## I applied the function to all combinations of these parameters
overlapmatrix <- function(y, p, s1, s2) {

  ## Subset each species, year and pond from daily calls data
  of1 <- subset(daily, subset = (sp == s1 & year == y & pond == p))
  of2 <- subset(daily, subset = (sp == s2 & year == y & pond == p))

  ## Lowess function for each
  lf1 <- lowess(of1$dailysum, f = 1/50, iter = 3, delta = 4)
  lf2 <- lowess(of2$dailysum, f = 1/50, iter = 3, delta = 4)

  ## Calculate overlap
  d <- data.frame(day = lf1$x,      # make a data frame with time as x
                  frog1 = lf1$y,    # and each species' lowess curve as a separate y
                  frog2 = lf2$y)
  d$min <- pmin(d$frog1, d$frog2)   # the upper bound of the integrated area the lower lowess curve
}
```

```

inter <- integrate.xy(d$day, d$min)
prop1 <- inter/integrate.xy(d$day, d$frog1) # make overlap a proportion of the overlapping area relat
prop2 <- inter/integrate.xy(d$day, d$frog2)

## Designate and return output
out <- cbind(prop1, prop2)
return(out)
}

```

After running the function through all species and years, the result is a dataframe with the proportional overlap for each species pair in each year and pond in which both species were present. There are two overlap values because this was standardized relative to the absolute size of each species' distribution.

```

##   X year pond sp1 sp2  overlap  overlapf2  spsp
## 1 1 2002    1  HC  HV 0.2876729 0.14013007 HC HV
## 2 2 2004    1  HC  HV 0.8335911 0.19707868 HC HV
## 3 3 2005    1  HC  HV 0.0000000 0.00000000 HC HV
## 4 4 2006    1  HC  HV 0.1953258 0.04375981 HC HV
## 5 5 2008    1  HC  HV 0.2490100 0.14031609 HC HV
## 6 6 2014    1  HC  HV 1.0000000 0.02264580 HC HV

```

Overlap trend across years

Now, I want to see how the area of overlap changes across years for each species pair. To do this, I built a custom function that 1) subsets the dataframe built above by species pair and pond 2) runs a linear model of overlap predicted by year 3) returns model diagnostics. I then apply this function to all 132 pairings of species and 8 ponds.

```

## The function take two inputs: species pair and pond
## I applied the function to all combinations of these parameters
overlaptrend <- function(sp, p) {

  ## Subset each species-pond
  s <- subset(fall, spsp == sp & pond == p)

  ## Linear model
  l <- lm(overlap ~ year, data = s)
  ls <- summary(l)

  ## Extract regression coefficient
  c <- round(coef(l)[2], 3) # regression slope
  r <- round(ls$r.squared, 3) # r-squared
  pv <- round(ls$coefficients[2, 4], 3) # p-value

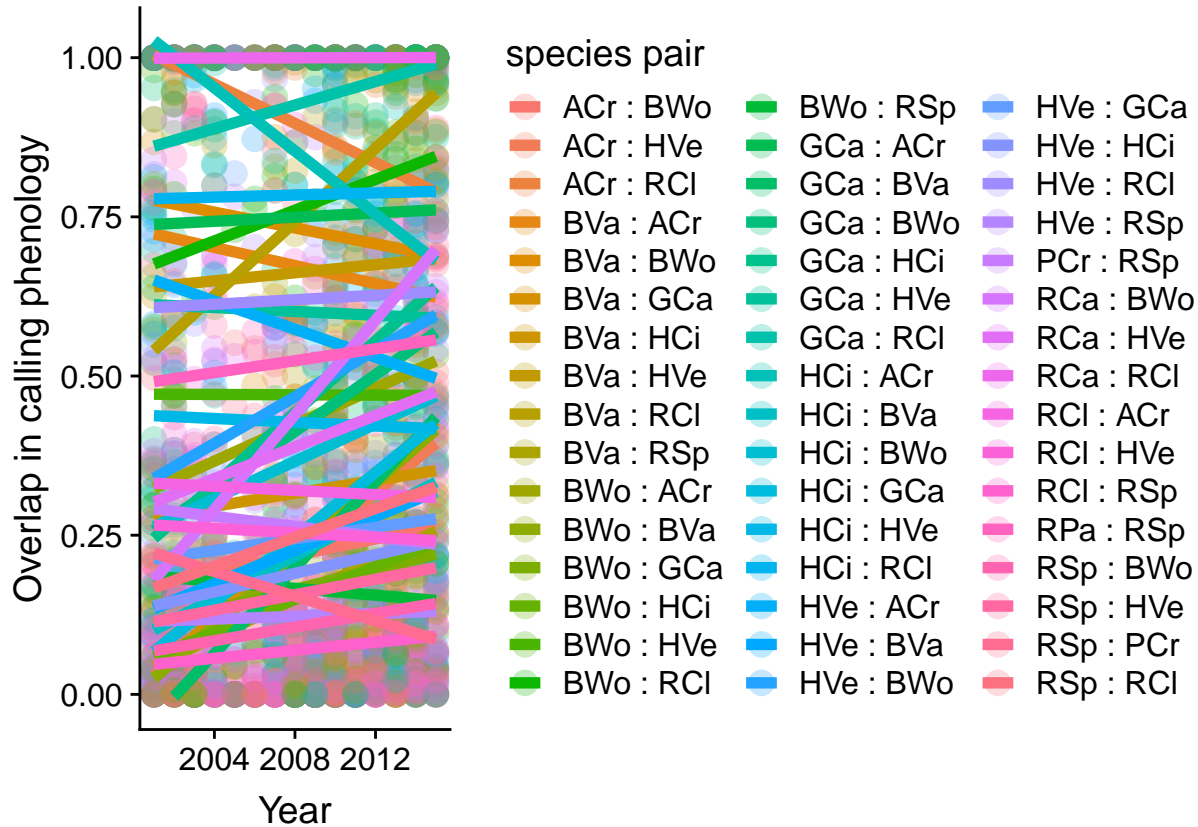
  ## Output- species pairs and model diagnostics
  out <- cbind(sp, p, c, r, pv)
  return(out)
}

```

Plotting all results

Now, we want to see how overlap between species temporal distributions changes across years for each species pair. This plot shows the regressions for each year, but it's super hard to read

and ugly. In this figure, each color represents a species pair. The slope of the line indicates how the overlap in calling between that species pair changed across the 15 year period of this data. Positive slopes indicate that overlap increased, negative slopes indicate that overlap decreased.



The info we really want from this figure is the slope of each line and the amount of error around that slope (from replicate ponds). The following figure gives exactly that and is much easier to interpret. Here, each row represents a species pair. Black points represent the average regression slope (pictured above) with error from pond replicates. Colored points represent the regression slope for each pond

```
## Order species pairs based on the magnitude of the regression slope so it's easier to see overall pat
over_year$spsp = with(over_year, factor(spsp, levels = rev(levels(spsp))))

## Overlap trend over years for each species pair
oy <- ggplot(over_year, aes(x = coef, y = reorder(spsp, coef))) + mytheme +

  ## Ponds shown independently
  geom_point(size = 3, alpha = 0.75, aes(color = as.factor(pond))) +

  ## Average of all 8 ponds
  geom_point(size = 4, colour = "black", shape = 18,
            data = over_year_err, aes(x = coef, y = spsp)) +

  ## SE bars on averages
  geom_errorbarh(data = over_year_err, size = 1, aes(xmin = coef - se, xmax = coef + se, height = 0)) +

  ## Style elements
  theme(panel.border = element_rect(colour = "black", fill = NA, size = 2),
        axis.text.y = element_text("mono", size = 12),
```

```

axis.text.x = element_text(size = 15),
legend.position = c(0.05, 0.85),
axis.title.x = element_text(size = 13),
axis.title.y = element_text(size = 15),
legend.background = element_blank(),
legend.box.background = element_rect(colour = "black")) +
geom_hline(yintercept = seq(0.5, 48.5, 1), color = "lightgray", linetype = 2) +
geom_vline(size = 1.5, linetype = 2, xintercept = 0) +
xlab("Regression slope \n(numerical overlap ~ year)") + ylab("Species pair") + labs(color = 'pond') +
## Add a Wes Anderson color palette for extra fun
scale_color_manual(values = wes_palette(n = 8, name = "Zissou1", type = 'continuous'))

```

oy

