Analyzing passerine response to restoration on Lagoon Island, UCSB

Madeline Berger, Renee Albretch, Robert Heim

2/17/2020

Introduction: Description of Data and Project

Efforts to remove invasive grasses and restore coastal sage scrub and oak woodlands are widespread throughout California. The goal of most of these restoration projects is to improve and restore ecosystem structure and functioning under the assumption that doing so will support populations of native wildlife (Gardali et al., 2006). Unfortunately, the impact of these restoration projects on local species populations is often unknown (Brawn et al., 2006). This makes it challenging to assess whether a restoration initiative has been successful. Restoration projects benefit from studies that evaluate the impacts of restoration efforts on local wildlife populations because findings from these studies can inform strategies to refine future restoration efforts (Ruiz-Jaen et al., 2005). Furthermore, providing tangible evidence of restoration success can be beneficial for securing funding for future restoration initiatives.

The Cheadle Center for Biodiversity and Ecological Restoration (CCBER) has conducted oak woodland and coastal sage scrub restoration activities on Lagoon Island on the campus of the University of California, Santa Barbara intermittently since 2005. Research has suggested that restoration activities in small patches of habitat, such as those on Lagoon Island, can increase reproductive success and breeding density of savannah and woodland birds (Brawn et al., 2006). In this analysis, we evaluate the impact of restoration efforts on Lagoon Island for two species of local songbirds: Song Sparrows and the California Towhee.

Data cleaning - CCBER

```
birds_all$species <- factor(birds_all$species)

###keeping original data - this is the one you need for modelling. this code just cleans up the month co

birds_all_clean <- birds_all %>%
  mutate(
    date = paste(day, month, year),
    date = dmy(date)
  ) %>%
  dplyr::select(-month, -day) %>%
  mutate(
    month = month(date, label = TRUE)
  ) %>%
  drop_na(month) %>%
  filter(year != 2017) %>%
  as.data.frame()

####wrangling data for nice graphs
```

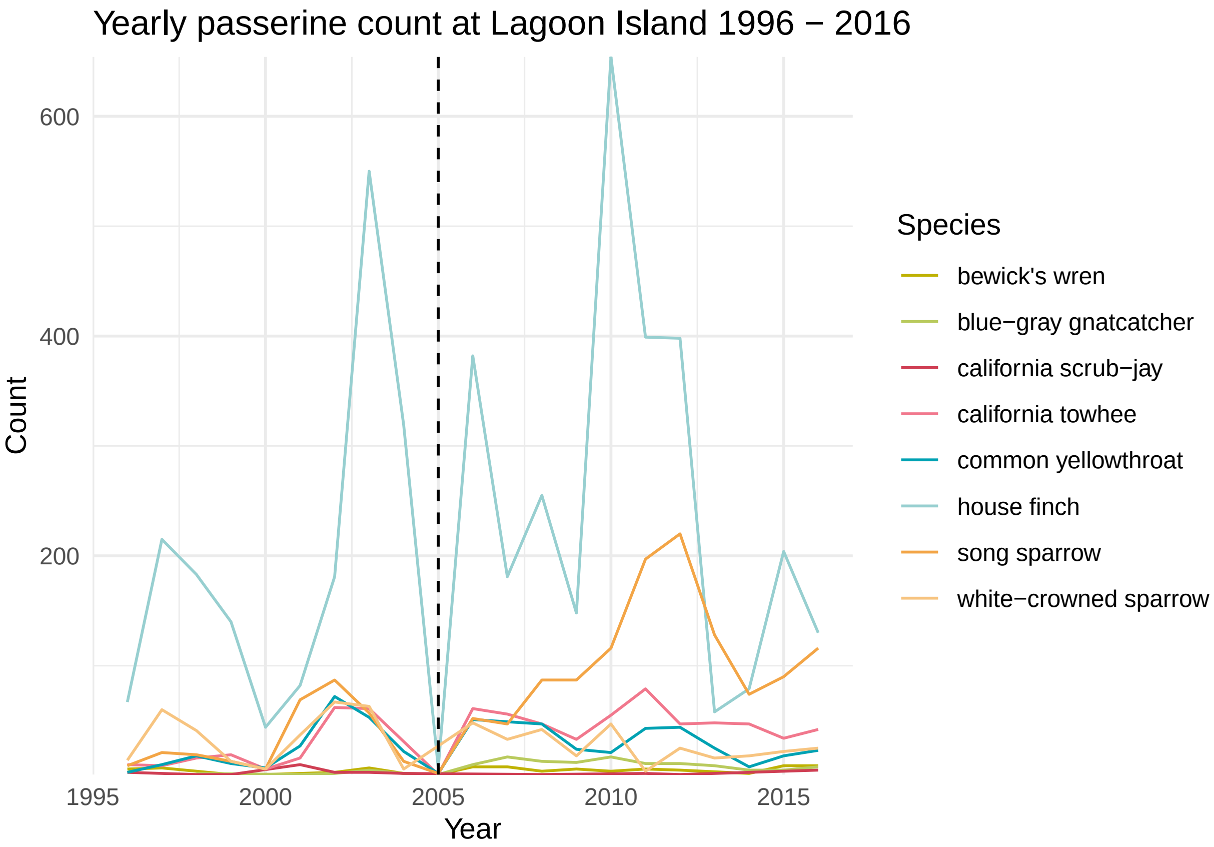
```
#break out and then parse dates - this will make graphing nicer
data_long <- melt(birds_all,
        # ID variables - all the variables to keep but not split apart on
    id.vars=c("day", "month","year", "species"),
        # The source columns
    measure.vars=c("lagoon_island_zone_summary"),
        # Name of the destination column that will identify the original
        # column that the measurement came from
    variable.name="location",
    value.name="count"
) %>%
  mutate(
    species = str_to_lower(species),
    date = paste(day, month, year),
    date = dmy(date)
  ) %>%
  dplyr::select(-month, -day) %>%
  mutate(
    month = month(date, label = FALSE)
  ) %>%
  mutate(species = case_when(
    species %in% "western scrub-jay" ~ "california scrub-jay",
    TRUE ~ species
  ))

#further wrangling if we want each line to represent one bird, also change the name of the scrub jay to
data_tidy<- uncount(data_long, weights = count) %>%
  mutate(species = str_to_lower(species)) %>%
  mutate(species = case_when(
    species %in% "western scrub-jay" ~ "california scrub-jay",
    TRUE ~ species
  ))


#filter by the species we are interested in, and up to 2016 since that is what best matches the bbs

#tidy version (missing zero counts)
bird_species_tidy <- data_tidy %>%
  filter(species == "california scrub-jay" | species =="common yellowthroat"| species == "song sparrow"

#version with counts
bird_species_count <- data_long %>%
  filter(species == "california scrub-jay" | species =="common yellowthroat"| species == "song sparrow"
```

Data Cleaning - BBS

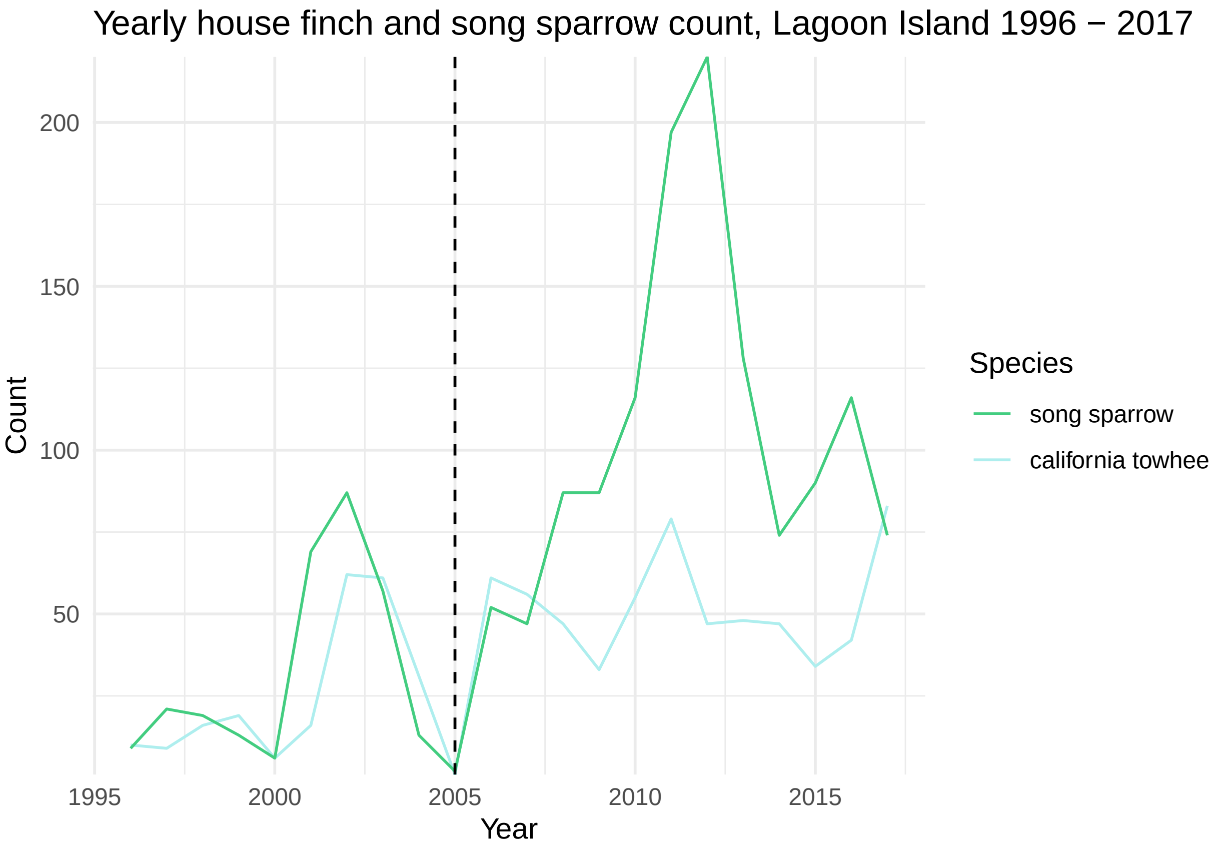
```
#change the id codes into species names
bbs_clean <- bbs_all %>%
  mutate(
    species = case_when(
      aou %in% 5190 ~ "house finch",
      aou %in% 5540 ~ "white-crowned sparrow",
      aou %in% 5810 ~ "song sparrow",
      aou %in% 5911 ~ "california towhee",
      aou %in% 6810 ~ "common yellowthroat",
      aou %in% 7190 ~ "bewick's wren",
      aou %in% 7510 ~ "blue-gray gnatcatcher",
      aou %in% 4812 ~ "california scrub jay"
    )
  ) %>%
  select(-aou, -country)
```

Summary graphs - CCBER

```
#create dfs that summarize yearly and monthly totals
bird_summary_year <- bird_species_tidy %>%
  select(-location) %>%
  #filter(species != "house finch") %>%
  filter(year != 2017) %>%
  group_by(year, species) %>%
  summarize(
    count = length(species)
  )

bird_summary_month_yr <- bird_species_tidy %>%
  select(-location) %>%
  group_by(month, year, species) %>%
  summarize(
    count = length(species)
  )

#change to sparrow bewicks only
sparrow_towhee_only <- bird_species_tidy %>%
  select(-location) %>%
  filter(species == "song sparrow" | species == "california towhee") %>%
  group_by(year, species) %>%
  summarize(
    count = length(species)
  )

#yearly abdunace graph with all the birds
yearly_abundance_graph <- ggplot(bird_summary_year, aes(x = year, y = count, group = species))+
  geom_line(aes(color = species))+
  theme_minimal()+
  labs(x = "Year", y = "Count", title = "Yearly passerine count at Lagoon Island 1996 - 2016", color =
  scale_y_continuous(expand = c(0,0))+
  geom_vline(xintercept = 2005, linetype = "dashed")+
  scale_color_paletteer_d("ggthemes::Summer")

yearly_abundance_graph
```



```
ggsave("count_all_yr.png", width = 6, height = 3, units = "in")

#just sparrow and finch

sparrow_towhee_gg <- ggplot(sparrow_towhee_only, aes(x = year, y = count, group = species)) +
  geom_line(aes(color = species))+
  theme_minimal()+
  labs(x = "Year", y = "Count", title = "Yearly house finch and song sparrow count, Lagoon Island 1996
  scale_y_continuous(expand = c(0,0))+
  geom_vline(xintercept = 2005, linetype = "dashed")+
  scale_color_manual(breaks = c("song sparrow", "california towhee"), values = c("paleturquoise2", "sea

sparrow_towhee_gg
```



Analysis: Song Sparrow

CCBER data

Model

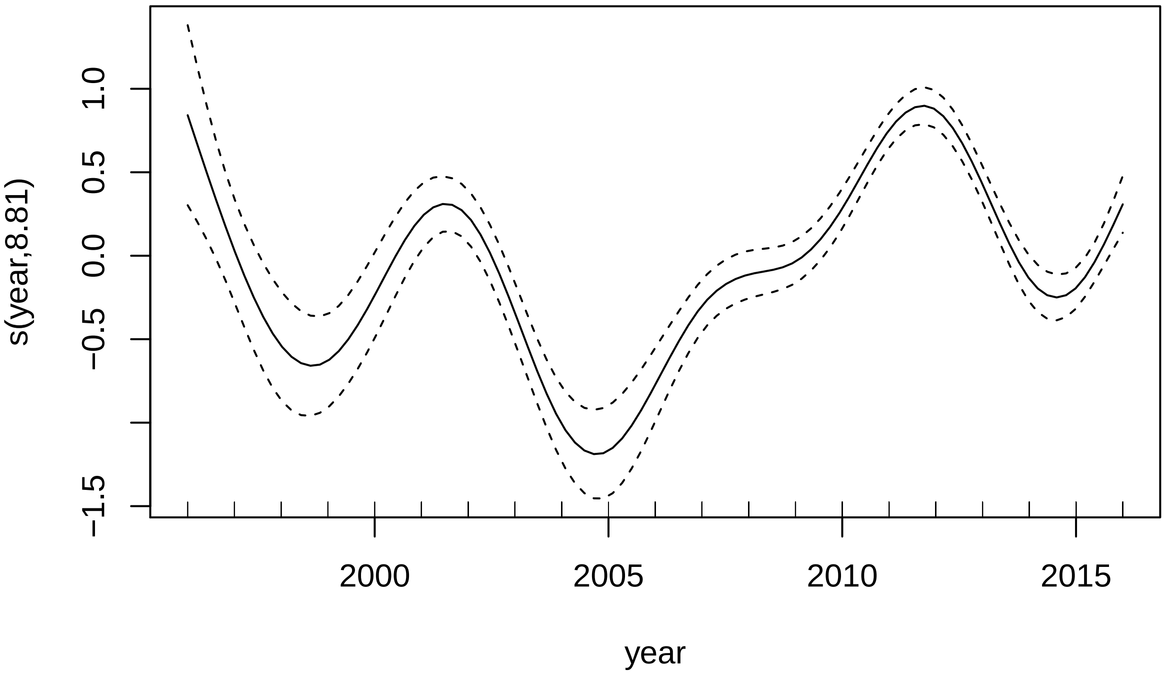
```
#go back to data read in `birds_all_clean` - pre major wrangling
#pick a species

birds_ssparrow <- birds_all_clean %>%
  filter(species == "Song Sparrow")

#run GAM on CCBER data

ccber_ssparrow_model <- mgcv::gam(lagoon_island_zone_summary ~ s(year) + month, family = poisson, data =

summary(ccber_ssparrow_model)
```

```
##
## Family: poisson
## Link function: log
##
## Formula:
## lagoon_island_zone_summary ~ s(year) + month
##
## Parametric coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.877255   0.032522  57.722  < 2e-16 ***
## month.L     -1.642981   0.116000 -14.164  < 2e-16 ***
## month.Q     -0.752906   0.105557  -7.133 9.84e-13 ***
## month.C      0.493234   0.099160   4.974 6.55e-07 ***
## month^4      0.225735   0.107942   2.091 0.036504 *
## month^5      0.079547   0.111208   0.715 0.474426
## month^6      0.418280   0.112499   3.718 0.000201 ***
## month^7      0.004361   0.108600   0.040 0.967967
## month^8     -0.149341   0.104018  -1.436 0.151081
## month^9     -0.144213   0.096673  -1.492 0.135762
## month^10     0.165043   0.088065   1.874 0.060918 .
## month^11     0.230684   0.088698   2.601 0.009301 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##           edf Ref.df Chi.sq p-value
## s(year) 8.814  8.987  330.1  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.473   Deviance explained = 52.8%
## UBRE = 2.7388  Scale est. = 1         n = 180
```

```
#create new data frame to use with the model
newd <- data.frame(year = 1996:2016, month = "Jul")

#predict abundance using new data, type predict shows actual abudance instead of log

ssparrow_predict <- predict.gam(ccber_ssparrow_model, newd, type="response")

ssparrow_predict
```

```
##         1         2         3         4         5         6         7         8
## 12.768047  5.675227  3.209127  2.934476  4.342316  6.931429  6.956482  3.904981
##         9        10        11        12        13        14        15        16
##  2.006397  1.709732  2.540911  4.076164  4.916368  5.312501  7.300413 11.670532
##        17        18        19        20        21
## 13.192476  8.440094  4.793886  4.536138  7.486751
```

```
#notes /issues
#didn't month because I couldn't figure it out
#how can we ask the models for integers

plot(ccber_ssparrow_model)
```

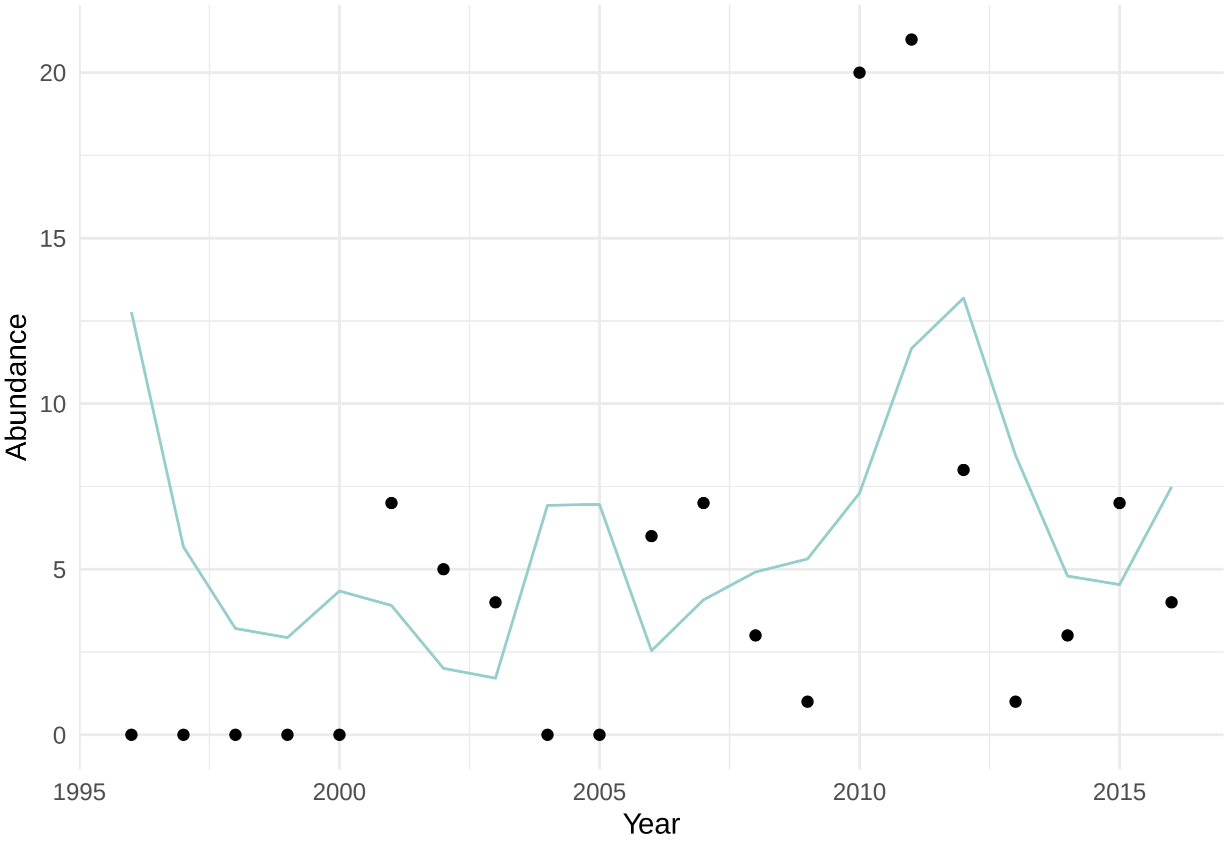


```r
ssparrow_test <- lm(lagoon_island_zone_summary~year+ month, data = birds_ssparrow)

summary(ssparrow_test)
```

```
##
## Call:
## lm(formula = lagoon_island_zone_summary ~ year + month, data = birds_ssparrow)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.021  -4.595  -1.165   2.386  35.529
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -606.5553   215.9879  -2.808  0.00557 **
## year           0.3062     0.1075   2.847  0.00496 **
## month.L      -10.4055     1.9277  -5.398 2.28e-07 ***
## month.Q       -4.4754     1.9037  -2.351  0.01989 *
## month.C        5.2925     1.9007   2.785  0.00598 **
## month^4        0.8196     1.9288   0.425  0.67145
## month^5       -1.4057     1.9896  -0.707  0.48085
## month^6        3.6026     1.9805   1.819  0.07069 .
## month^7       -0.5720     1.9845  -0.288  0.77351
## month^8       -1.1741     1.9629  -0.598  0.55057
## month^9       -1.1811     1.9399  -0.609  0.54345
## month^10       1.6855     1.9734   0.854  0.39426
## month^11       1.3494     1.9437   0.694  0.48851
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.51 on 167 degrees of freedom
## Multiple R-squared:  0.2556, Adjusted R-squared:  0.2021
## F-statistic: 4.777 on 12 and 167 DF,  p-value: 1.047e-06
```

Tidy model and visualize

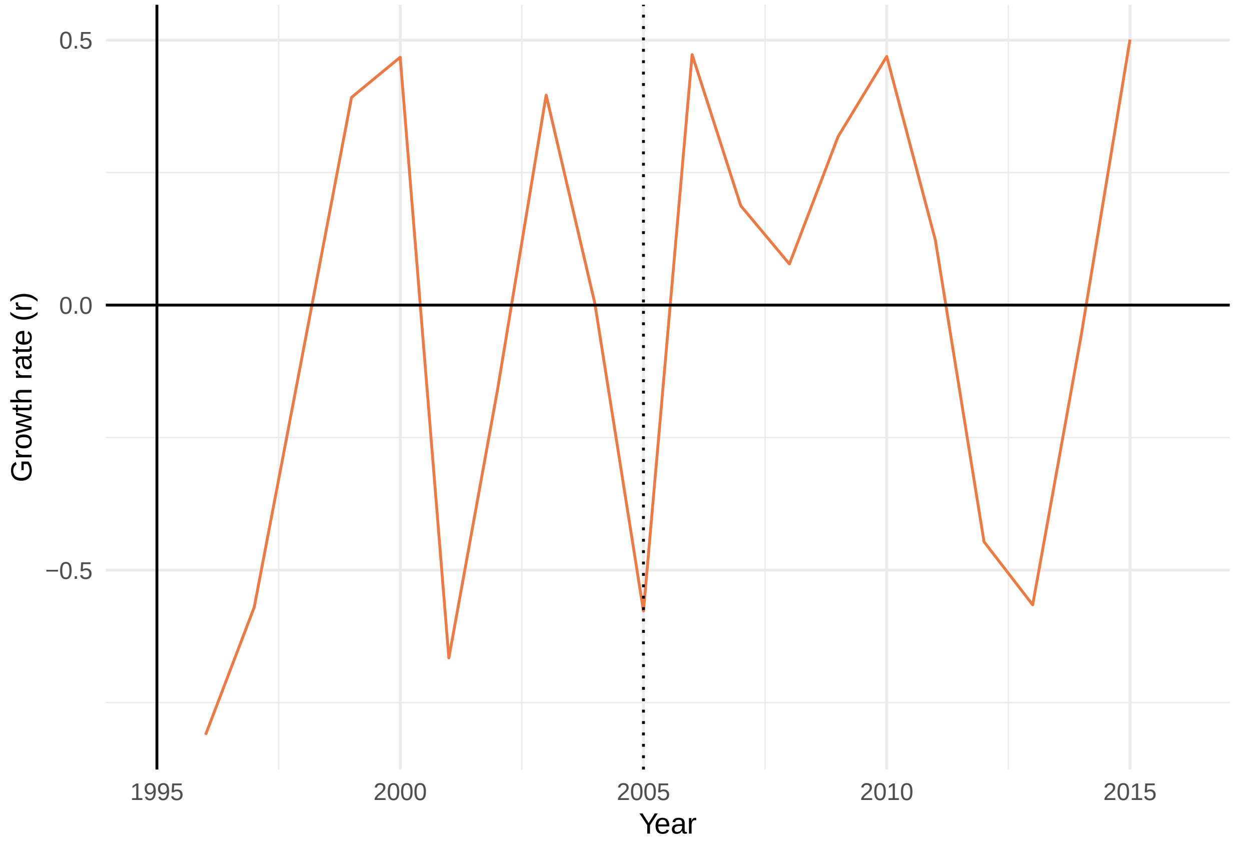
```
ccber_ssparrow_tidy <- broom::tidy(ccber_ssparrow_model) #this didn't end up being necessary

#we want the actual counts per year for song sparrows, so use the `birds_summmary_year` df

ssparrow_by_year <- bird_summary_month_yr %>%
  filter(species == "song sparrow") %>%
  filter(month == 7) %>%
  filter(year != 2017)

#we need data for 1996 - 2000, going to create a dataframe with count = 0

month <- rep(7,7)
year <- c(1996, 1997, 1998, 1999, 2000, 2004, 2005)
species <- c("song sparrow", "song sparrow", "song sparrow", "song sparrow","song sparrow","song sparro
count <- rep(0,7)

#make into dataframe
missing_ss <- data.frame(month, year,species,count)

ssparrow_jul <- bind_rows(missing_ss, ssparrow_by_year)

#bind obs and predictions
ssparrow_complete <- data.frame(ssparrow_jul, ssparrow_predict)


ssparrow <- ggplot(data = ssparrow_complete, aes(x = year, y = count)) +
  geom_point()+
  geom_line(aes(x = year, y = ssparrow_predict), color = "paleturquoise3") +
  theme_minimal()+
  labs(x = "Year", y = "Abundance")

ssparrow
```



```
#ggsave("count_v_predict.png", width = 6, height = 3, units = "in")
```

Calculate growth rates

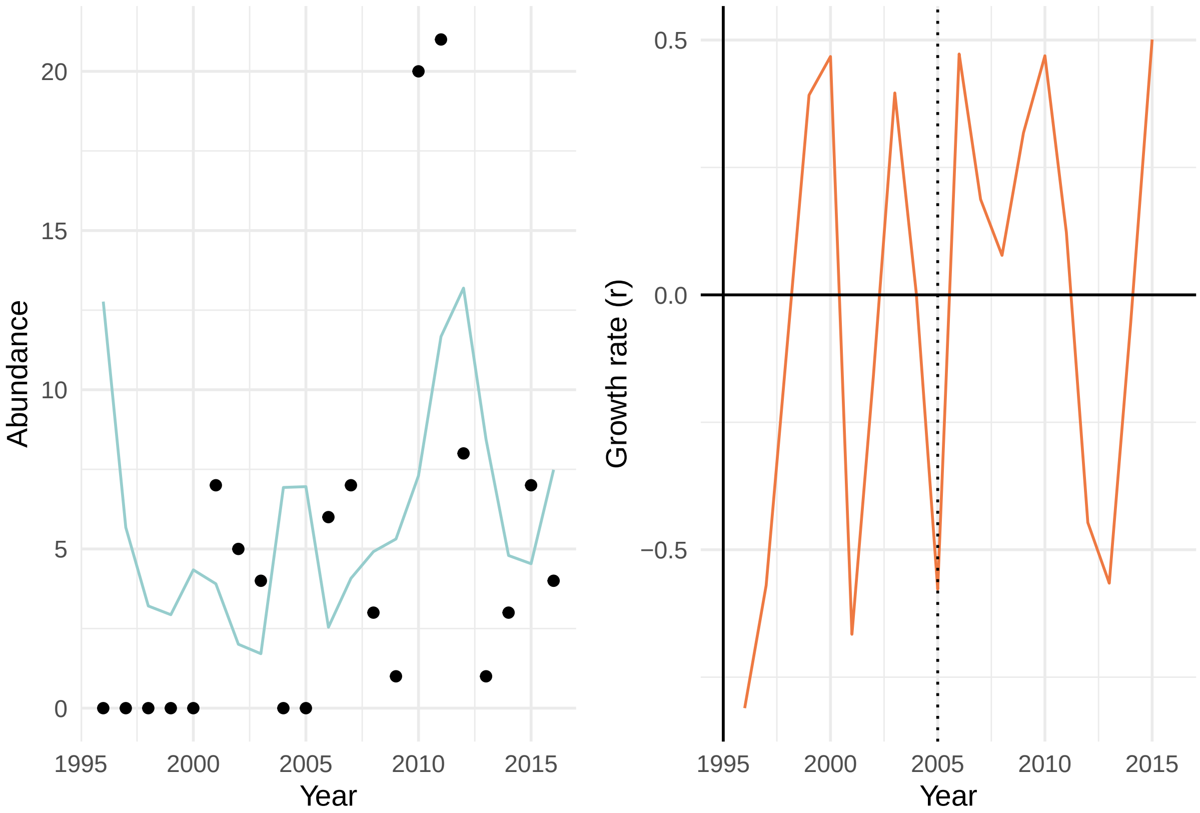
```
#create a new column in sparrow complete that calculates r based on the year before using the log equat

ssparrow_gr <- ssparrow_complete %>%
  mutate(
    r = log(lead(ssparrow_predict)/ssparrow_predict)
  )

#graph - what should we test?

ssparrow_r <- ggplot(ssparrow_gr, aes(x = year, y = r))+
  geom_line(color = "sienna2")+
  theme_minimal()+
  geom_hline(yintercept = 0)+
  geom_vline(xintercept = 1995)+
  geom_vline(xintercept = 2005, linetype = "dotted")+
  labs(x = "Year", y = "Growth rate (r)")

ssparrow_r
```



```
ggsave("ssparrow_gr_ccber.png", width = 6, height = 3, units = "in")


#find averages - I didn't make a new data frame for each test I just changed the names and re -ran this

ss_ccber_before <- ssparrow_gr %>%
  filter(year < 2006) %>%
  filter(year != 2016)

ss_avg_bef_ccber <- mean(ss_ccber_before$r) #this always stays the same
sd_bef<- sd(ss_ccber_before$r) # feed in a vector
se_bef <- sd_bef/sqrt(nrow(ss_ccber_before)) #calculate se

ss_ccber_after <- ssparrow_gr %>%
  filter(year > 2010) %>%
  filter(year != 2016)

ss_avg_after_5 <- mean(ss_ccber_after$r)
sd_after_5<- sd(ss_ccber_after$r) # feed in a vector
se_after_5<- sd_bef/sqrt(nrow(ss_ccber_after))
```

Combine graphs using patchwork - this is more for knitting

```
ssparrow + ssparrow_r
```



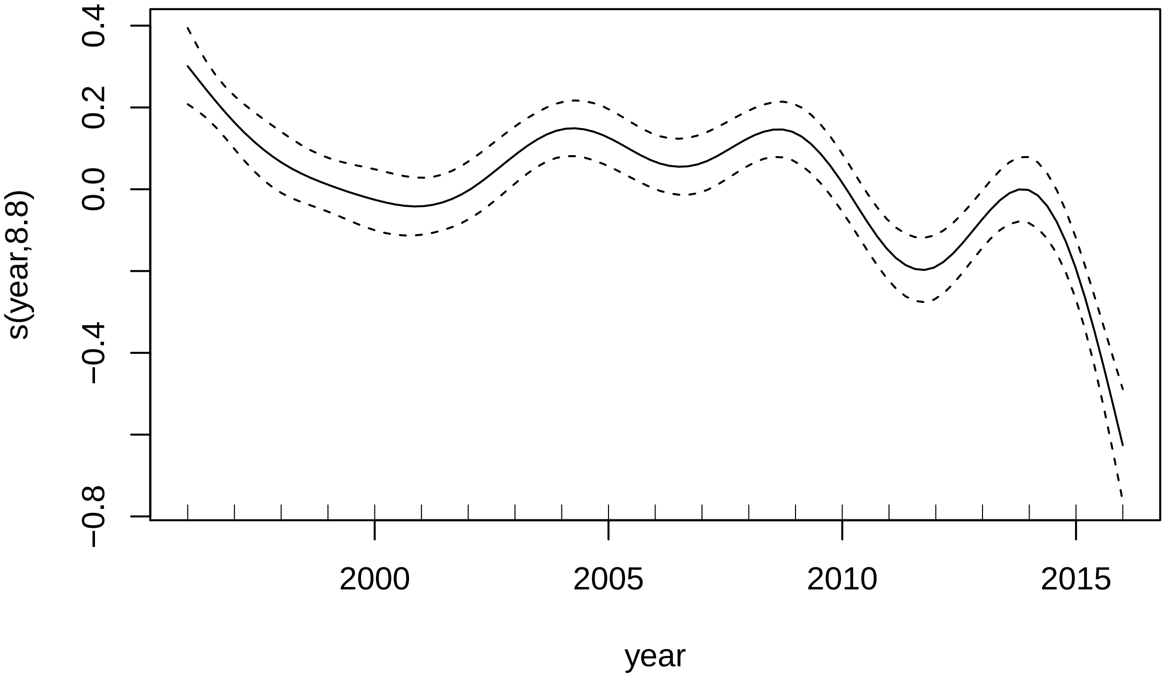
BBS

Model

```
#filter out song sparrow

bbs_ssparrow <- bbs_clean %>%
  filter(species == "song sparrow") %>%
  group_by(year) %>%
  summarize(
    species_total = sum(species_total)
  )

#run GAM
bbs_ssparrow_model <- mgcv::gam(species_total ~ s(year), family = poisson, data = bbs_ssparrow)

summary(bbs_ssparrow_model)
```

```
##
## Family: poisson
## Link function: log
##
## Formula:
## species_total ~ s(year)
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   5.6908     0.0128   444.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##          edf Ref.df Chi.sq p-value
## s(year) 8.8  8.988    163  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.659   Deviance explained = 83.2%
## UBRE =   1.67  Scale est. = 1         n = 21
```

```
plot(bbs_ssparrow_model)
```



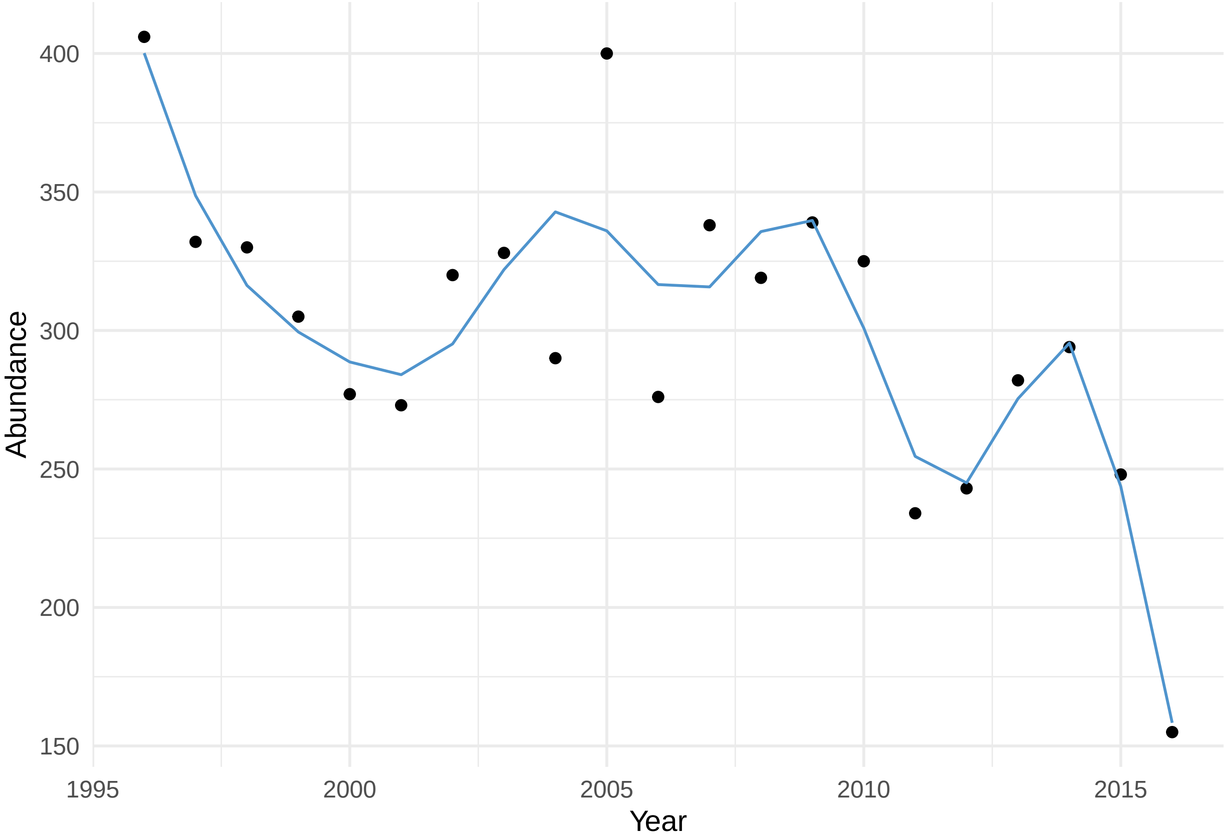
```
#create new data frame for predictions
newd_bbs <- data.frame(year = 1996:2016)

#predict abundance using new data

ssparrow_predict_bbs <- predict.gam(bbs_ssparrow_model, newd_bbs, type="response")

ssparrow_predict_bbs
```

```
##        1        2        3        4        5        6        7        8
## 400.1306 348.6577 316.2559 299.4398 288.6209 284.0409 295.1348 321.9773
##        9       10       11       12       13       14       15       16
## 342.8080 335.9626 316.5712 315.7245 335.7077 339.7127 300.8595 254.5384
##       17       18       19       20       21
## 244.9862 275.3828 295.4363 243.6869 158.3655
```

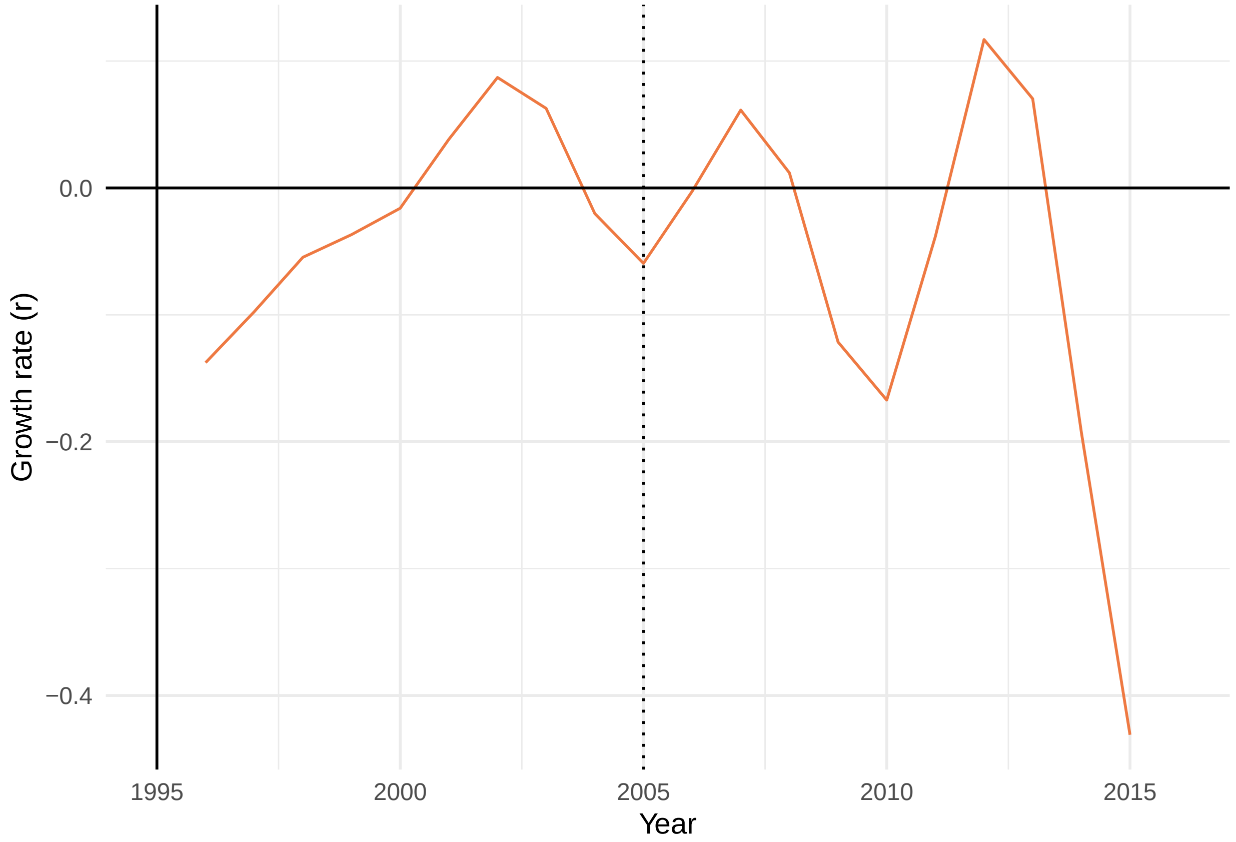
Join observations and predictions

```
ssparrow_complete_bbs <- data.frame(bbs_ssparrow, ssparrow_predict_bbs)
```

```
ssparrow_gr_bbs <- ssparrow_complete_bbs %>%
  mutate(
    r = log(lead(ssparrow_predict_bbs)/ssparrow_predict_bbs)
  )

#note - this graph of the model prediction + observation for BBS is all the data because it is compiled
ssparrow_bbs <- ggplot(data = ssparrow_complete_bbs, aes(x = year, y = species_total)) +
  geom_point()+
  geom_line(aes(x = year, y = ssparrow_predict_bbs), color = "steelblue3") +
  theme_minimal()+
  labs(x = "Year", y = "Abundance")

ssparrow_bbs
```



```
ggsave("count_v_predict_bbs.png", width = 6, height = 3, units = "in")


ssparrow_bbs_r <- ggplot(ssparrow_gr_bbs, aes(x = year, y = r))+
  geom_line(color = "sienna2")+
  theme_minimal()+
  geom_hline(yintercept = 0)+
  geom_vline(xintercept = 1995)+
  geom_vline(xintercept = 2005, linetype = "dotted")+
  labs(x = "Year", y = "Growth rate (r)")

ssparrow_bbs_r
```



```
ggsave("ssparrow_bbs_growthrate.png", width = 6, height = 3, units = "in")
```

Calculate growth rates

```
#df
ss_bbs_before <- ssparrow_gr_bbs %>%
  filter(year < 2006) %>%
  filter(year != 2016)
#numbers
ss_bbs_bef <- mean(ss_bbs_before$r)
sd_bbs_bef<- sd(ss_bbs_before$r) # feed in a vector
se_bbs_bef <- sd_bef/sqrt(nrow(ss_bbs_before)) #calculate se

#df
ss_bbs_after <- ssparrow_gr_bbs %>%
  filter(year > 2010) %>%
  filter(year != 2016)

#numbers
ss_bbs_after_5 <- mean(ss_bbs_after$r)
sd_bbs_after_5<- sd(ss_bbs_after$r) # feed in a vector
se_bbs_after_5 <- sd_bef/sqrt(nrow(ss_bbs_after))
```

Test if difference between growth rates for CCBER and BBS is significant

```
#three tests: before, after 2005, after 2010.

t.test(ss_ccber_before$r, ss_bbs_before$r, conf.level = 0.95, paired = TRUE)
```

```
##
##  Paired t-test
##
## data:  ss_ccber_before$r and ss_bbs_before$r
## t = -0.90234, df = 9, p-value = 0.3904
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.4840278  0.2079918
## sample estimates:
## mean of the differences
##               -0.138018
```

```
t.test(ss_ccber_after$r,ss_bbs_after$r, conf.level = 0.95, paired = TRUE)
```

```
##
##  Paired t-test
##
## data:  ss_ccber_after$r and ss_bbs_after$r
## t = 0.021413, df = 4, p-value = 0.9839
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.7877922  0.8000379
## sample estimates:
## mean of the differences
##             0.006122855
```

Analysis: California towhee

CCBER

Model

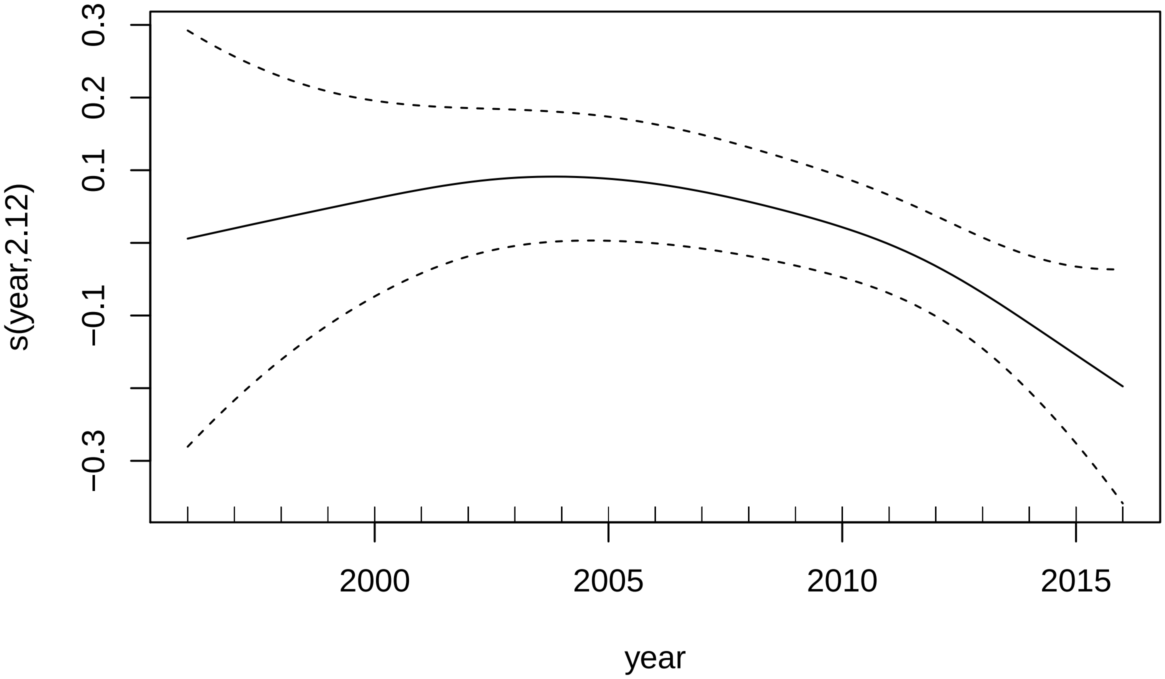
```
#Go back to data read in `birds_all_clean` - pre major wrangling
#Pick a species - California towhee

birds_ctowhee <- birds_all_clean %>%
  filter(species == "California Towhee")

#Run GAM on CCBER data

ccber_ctowhee_model <- mgcv::gam(lagoon_island_zone_summary ~ s(year) + month, family = poisson, data =

summary(ccber_ctowhee_model)
```

```
##
## Family: poisson
## Link function: log
##
## Formula:
## lagoon_island_zone_summary ~ s(year) + month
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.42313    0.03745  38.006  < 2e-16 ***
## month.L     -0.25575    0.13904  -1.839   0.0658 .
## month.Q     -0.60935    0.12972  -4.697 2.64e-06 ***
## month.C      0.10530    0.12399   0.849   0.3958    
## month^4      0.19531    0.13139   1.487   0.1371    
## month^5     -0.09022    0.14221  -0.634   0.5258    
## month^6      0.30809    0.14071   2.190   0.0286 *  
## month^7     -0.03975    0.13492  -0.295   0.7683    
## month^8      0.18003    0.12642   1.424   0.1544    
## month^9     -0.07847    0.11862  -0.662   0.5083    
## month^10    -0.03854    0.11963  -0.322   0.7473    
## month^11     0.20470    0.11569   1.769   0.0768 .  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Approximate significance of smooth terms:
##           edf Ref.df Chi.sq p-value  
## s(year) 2.123  2.666  6.711  0.0682 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## R-sq.(adj) =  0.0256   Deviance explained = 10.3%
## UBRE = 1.4526  Scale est. = 1         n = 181
```

```
#create new data frame to use with the model
newd <- data.frame(year = 1996:2016, month = "Jul")

#predict abundance using new data, type predict shows actual abudance instead of log

ctowhee_predict <- predict.gam(ccber_ctowhee_model, newd, type="response")

ctowhee_predict
```

```
##        1        2        3        4        5        6        7        8 
## 4.421172 4.484312 4.547134 4.609643 4.671869 4.730498 4.778146 4.807332 
##        9       10       11       12       13       14       15       16 
## 4.814733 4.800977 4.767665 4.716658 4.651596 4.576583 4.491043 4.387433 
##       17       18       19       20       21 
## 4.257549 4.102416 3.934177 3.767013 3.607516
```

```
#notes /issues
#didn't month because I couldn't figure it out
#how can we ask the models for integers

plot(ccber_ctowhee_model)
```

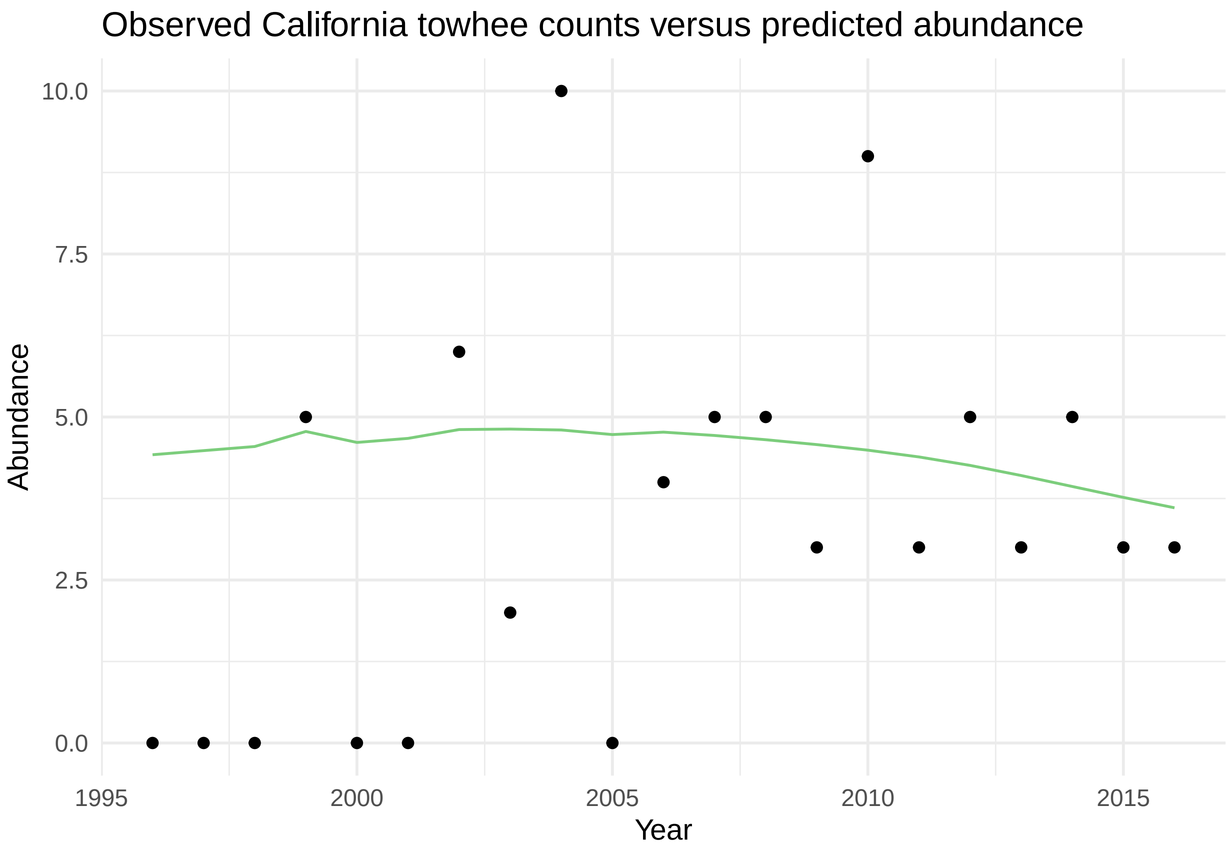


```
ctowhee_test <- lm(lagoon_island_zone_summary~year+ month, data = birds_ctowhee)

summary(ctowhee_test)
```

```
##
## Call:
## lm(formula = lagoon_island_zone_summary ~ year + month, data = birds_ctowhee)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.8384 -1.9552 -0.5194  1.0360 11.7240
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 117.40838   93.95792   1.250  0.21319
## year         -0.05633    0.04678  -1.204  0.23027
## month.L      -1.07559    0.85206  -1.262  0.20857
## month.Q      -2.55860    0.83785  -3.054  0.00263 **
## month.C       0.59985    0.83038   0.722  0.47106
## month^4       0.77246    0.84794   0.911  0.36361
## month^5      -0.48168    0.88186  -0.546  0.58564
## month^6       0.98040    0.87820   1.116  0.26586
## month^7      -0.16033    0.87386  -0.183  0.85465
## month^8       0.63695    0.85955   0.741  0.45971
## month^9      -0.37723    0.84416  -0.447  0.65555
## month^10     -0.27522    0.86232  -0.319  0.75000
## month^11      1.07718    0.85336   1.262  0.20860
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.301 on 168 degrees of freedom
## Multiple R-squared:  0.09132,    Adjusted R-squared:  0.02642
## F-statistic: 1.407 on 12 and 168 DF,  p-value: 0.1669
```

Tidy model and visuzalize

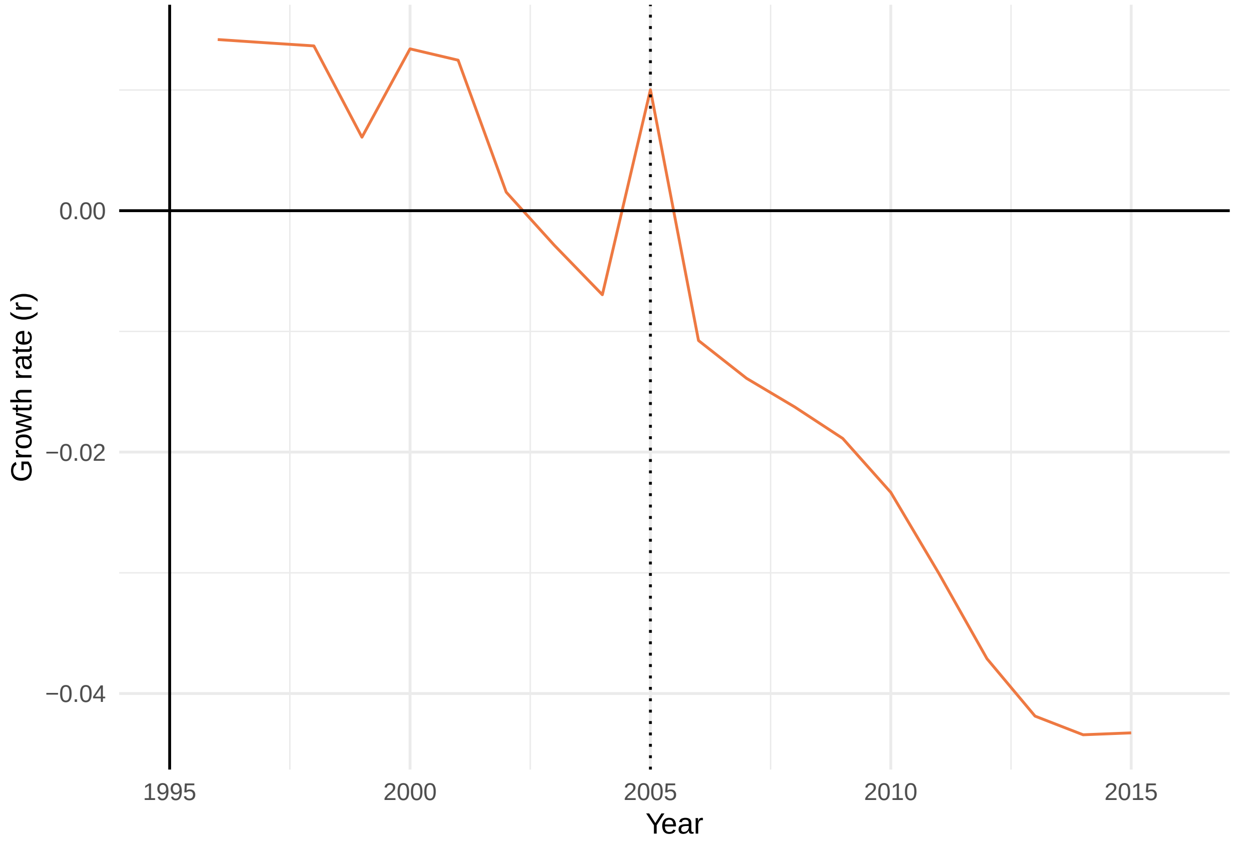
```
ccber_ctowhee_tidy <- broom::tidy(ccber_ctowhee_model) #this didn't end up being necessary

#we want the actual counts per year for towhees, so use the `birds_summmary_year` df

ctowhee_by_year <- bird_summary_month_yr %>%
  filter(species == "california towhee") %>%
  filter(month == 7) %>%
  filter(year != 2017)

#we need data for 1996 - 2000, going to create a dataframe with count = 0
#years will be different for Bewick's Wren

month <- rep(6,6)
year <- c(1996, 1997, 1998, 2000, 2001, 2005)
species <- c("california towhee", "california towhee", "california towhee", "california towhee","califo
count <- rep(0,6)

#make into dataframe
missing_ct <- data.frame(month, year,species,count)

ctowhee_jul <- bind_rows(missing_ct, ctowhee_by_year)

#bind obs and predictions
ctowhee_complete <- data.frame(ctowhee_jul, ctowhee_predict)


ctowhee <- ggplot(data = ctowhee_complete, aes(x = year, y = count)) +
  geom_point()+
  geom_line(aes(x = year, y = ctowhee_predict), color = "palegreen3") +
  theme_minimal()+
  labs(x = "Year", y = "Abundance", title="Observed California towhee counts versus predicted abundance

ctowhee
```



```
ggsave("count_v_predict.png", width = 6, height = 3, units = "in")
```

Calculate growth rates

```
#create a new column in ctowhee complete that calculates r based on the year before using the log equat

ctowhee_gr <- ctowhee_complete %>%
  mutate(
    r = log(lead(ctowhee_predict)/ctowhee_predict)
  )

#graph - what should we test?

ctowhee_r <- ggplot(ctowhee_gr, aes(x = year, y = r))+
  geom_line(color = "sienna2")+
  theme_minimal()+
  geom_hline(yintercept = 0)+
  geom_vline(xintercept = 1995)+
  geom_vline(xintercept = 2005, linetype = "dotted")+
  labs(x = "Year", y = "Growth rate (r)")

ctowhee_r
```

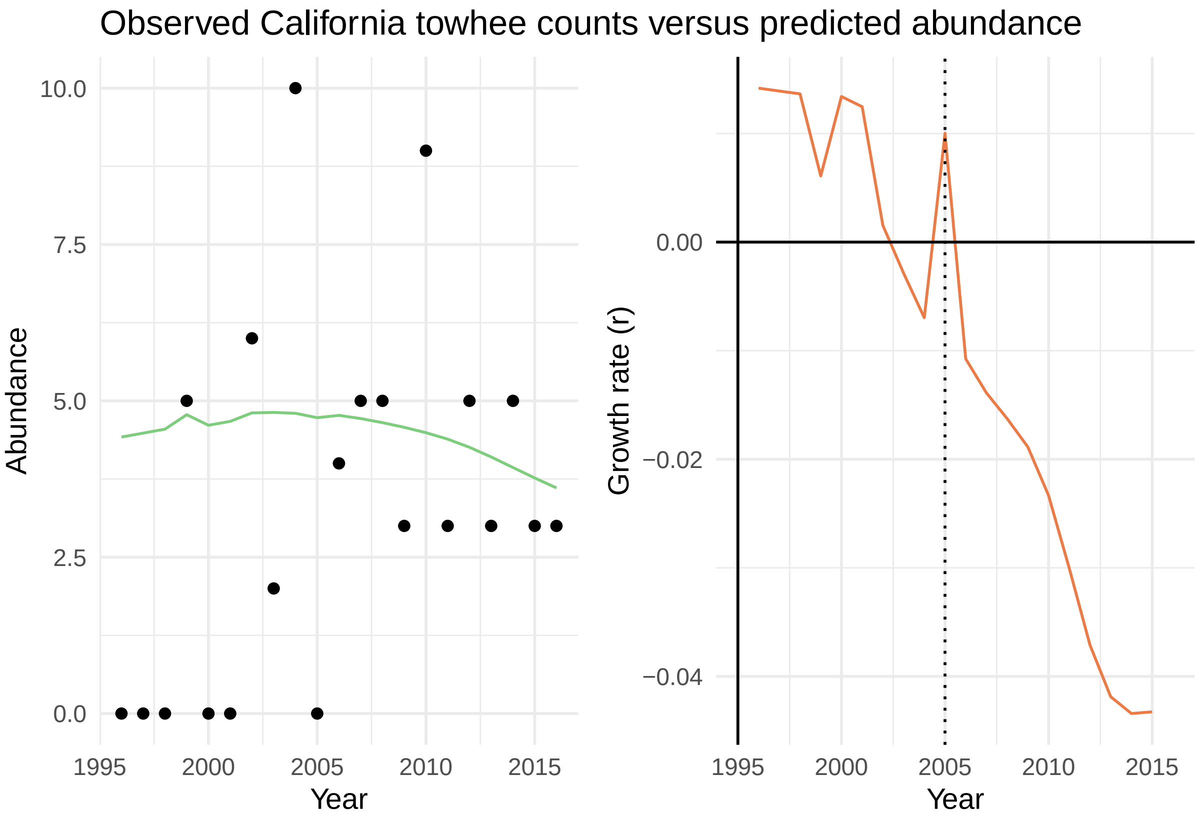


```
ggsave("ctowhee_gr_ccber.png", width = 6, height = 3, units = "in")


#find averages - I didn't make a new data frame for each test I just changed the names and re -ran this

ct_ccber_before <- ctowhee_gr %>%
  filter(year < 2006) %>%
  filter(year != 2016)

ct_avg_bef_ccber <- mean(ct_ccber_before$r) #this always stays the same
sd_bef<- sd(ct_ccber_before$r) # feed in a vector
se_bef <- sd_bef/sqrt(nrow(ct_ccber_before)) #calculate se

ct_ccber_after <- ctowhee_gr %>%
  filter(year > 2005) %>%
  filter(year != 2016)

#redo for 5 and 10 years after restoration

ct_avg_after_5 <- mean(ct_ccber_after$r)
sd_after_5<- sd(ct_ccber_after$r) # feed in a vector
se_after_5<- sd_bef/sqrt(nrow(ct_ccber_after))

ct_ccber_after_10 <- ctowhee_gr %>%
  filter(year > 2010) %>%
  filter(year != 2016)
```

```
ct_avg_after_10 <- mean(ct_ccber_after_10$r)
sd_after_10<- sd(ct_ccber_after_10$r) # feed in a vector
se_after_10<- sd_bef/sqrt(nrow(ct_ccber_after_10))
```

Combine graphs using patchwork - this is more for knitting

```
ctowhee + ctowhee_r
```



```
# May want to change y-axis scale
```

BBS

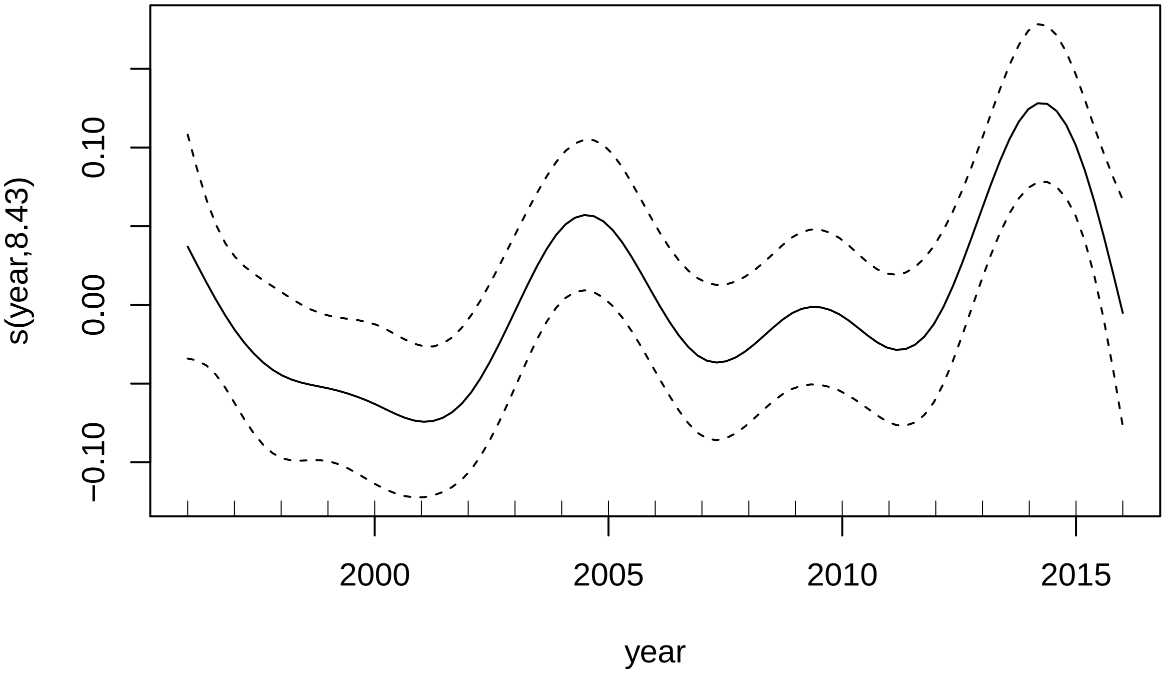
Model

```
#filter out song sparrow

bbs_ctowhee <- bbs_clean %>%
  filter(species == "california towhee") %>%
  group_by(year) %>%
  summarize(
    species_total = sum(species_total)
  )

#run GAM
bbs_ctowhee_model <- mgcv::gam(species_total ~ s(year), family = poisson, data = bbs_ctowhee)

summary(bbs_ctowhee_model)
```

```
##
```

```
## Family: poisson
## Link function: log
##
## Formula:
## species_total ~ s(year)
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 6.433559   0.008752   735.1   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##           edf Ref.df Chi.sq  p-value
## s(year) 8.433  8.909  36.73 1.68e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.152   Deviance explained = 50.5%
## UBRE = 1.8368  Scale est. = 1         n = 21
```

```
plot(bbs_ctowhee_model)
```

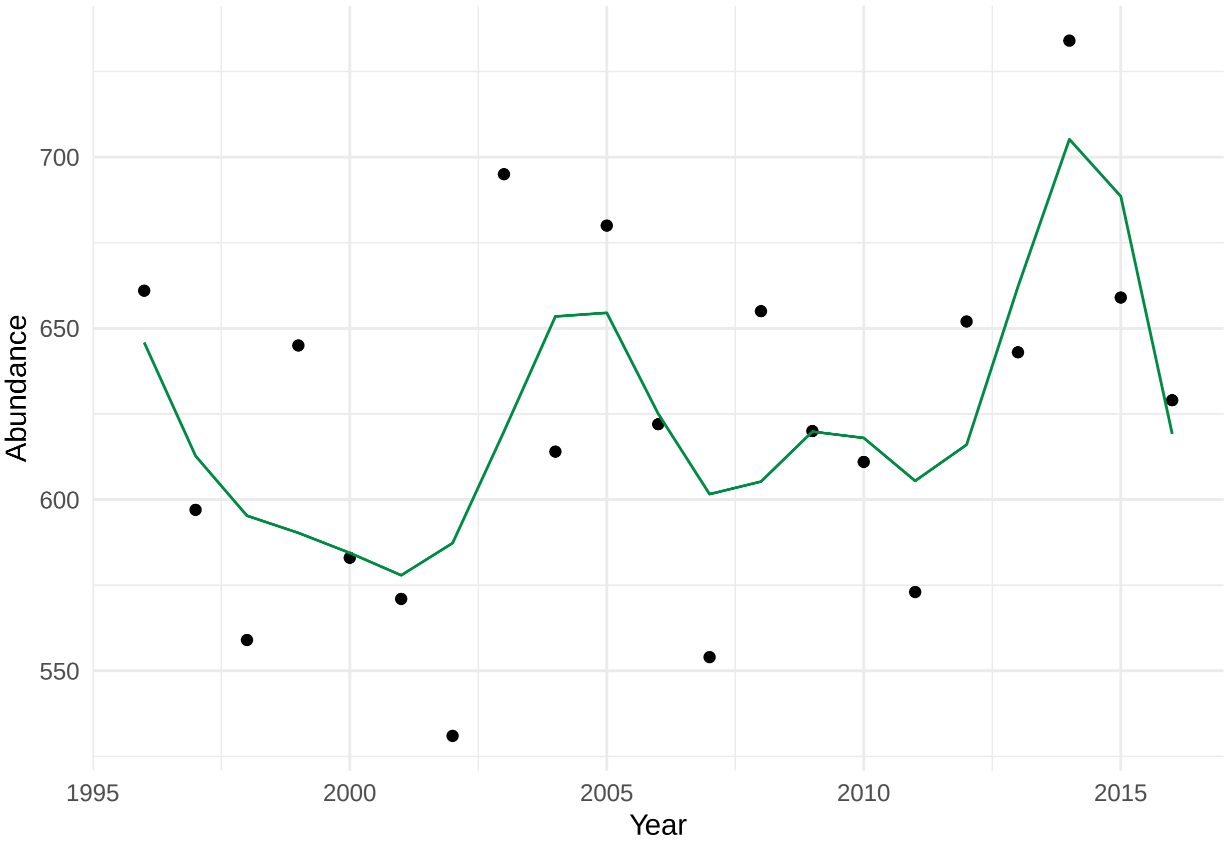


```
#create new data frame for predictions
newd_ctowhee_bbs <- data.frame(year = 1996:2016)

#predict abundance using new data

ctowhee_predict_bbs <- predict.gam(bbs_ctowhee_model, newd_ctowhee_bbs, type="response")

ctowhee_predict_bbs
```

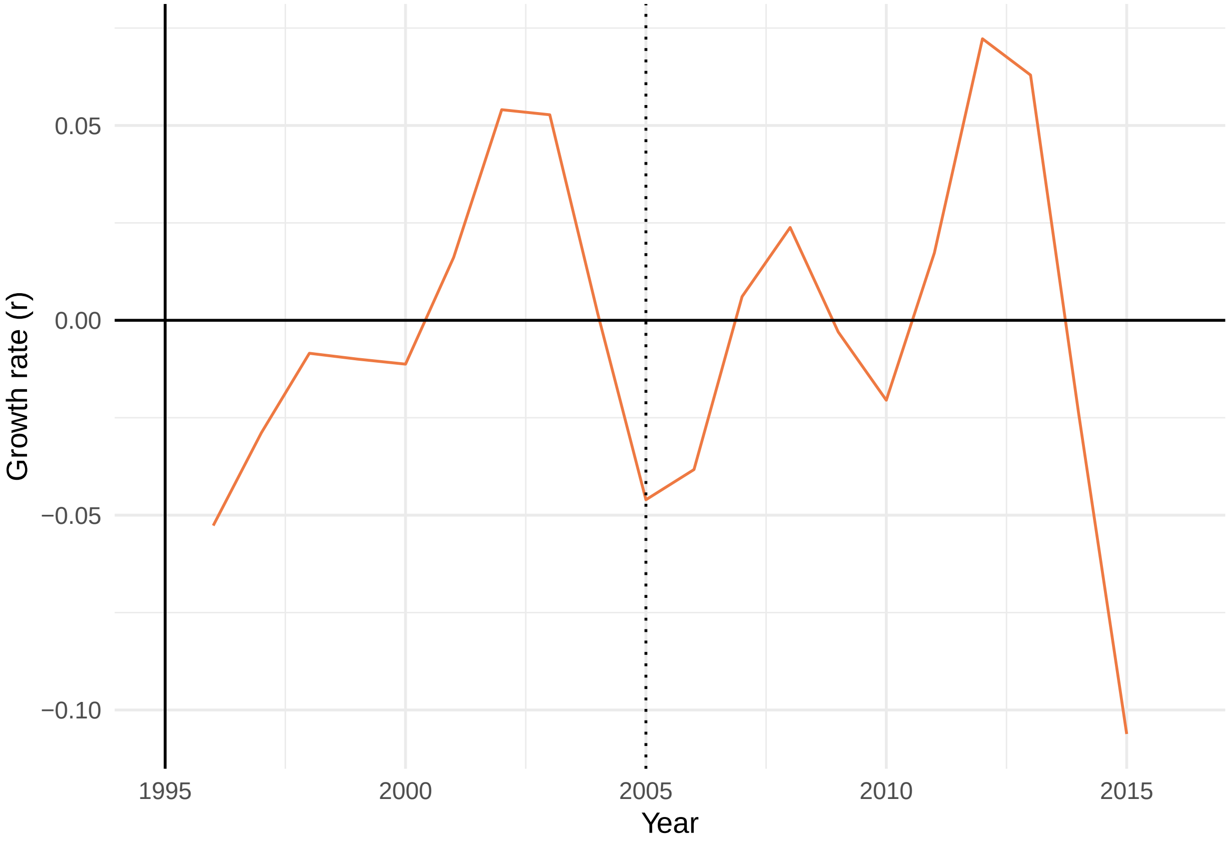
```
##         1         2         3         4         5         6         7         8
## 645.8746 612.7340 595.2848 590.2701 584.4231 577.8870 587.2845 619.8971
##        9       10       11       12       13       14       15       16
## 653.4722 654.5273 625.0684 601.5830 605.2514 619.8464 617.9978 605.4558
##       17       18       19       20       21
## 616.0170 662.1736 705.1910 688.5548 619.2062
```

Join observations and predictions

```
ctowhee_complete_bbs <- data.frame(bbs_ctowhee, ctowhee_predict_bbs)

ctowhee_gr_bbs <- ctowhee_complete_bbs %>%
  mutate(
    r = log(lead(ctowhee_predict_bbs)/ctowhee_predict_bbs)
  )

#note - this graph of the model prediction + observation for BBS is all the data because it is compiled
ctowhee_bbs <- ggplot(data = ctowhee_complete_bbs, aes(x = year, y = species_total)) +
  geom_point()+
  geom_line(aes(x = year, y = ctowhee_predict_bbs), color = "springgreen4") +
  theme_minimal()+
  labs(x = "Year", y = "Abundance")

ctowhee_bbs
```



```
ggsave("count_v_predict_bbs.png", width = 6, height = 3, units = "in")


ctowhee_bbs_r <- ggplot(ctowhee_gr_bbs, aes(x = year, y = r))+
  geom_line(color = "sienna2")+
  theme_minimal()+
  geom_hline(yintercept = 0)+
  geom_vline(xintercept = 1995)+
  geom_vline(xintercept = 2005, linetype = "dotted")+
  labs(x = "Year", y = "Growth rate (r)")

ctowhee_bbs_r
```



```
ggsave("ctowhee_bbs_growthrate.png", width = 6, height = 3, units = "in")
```

Calculate growth rates

```
#df
ct_bbs_before <- ctowhee_gr_bbs %>%
  filter(year < 2006) %>%
  filter(year != 2016)

#numbers before restoration
ct_bbs_bef <- mean(ct_bbs_before$r)
sd_bbs_bef<- sd(ct_bbs_before$r) # feed in a vector
se_bbs_bef <- sd_bef/sqrt(nrow(ct_bbs_before)) #calculate se

#df
ct_bbs_after <- ctowhee_gr_bbs %>%
  filter(year > 2005) %>%
  filter(year != 2016)
```

```
#numbers 5 and 10 years after restoration

ct_bbs_after_5 <- mean(ct_bbs_after$r)
sd_bbs_after_5<- sd(ct_bbs_after$r) # feed in a vector
se_bbs_after_5 <- sd_bef/sqrt(nrow(ct_bbs_after))

ct_bbs_after_10 <- ctowhee_gr_bbs %>%
  filter(year > 2010) %>%
  filter(year != 2016)

ct_avg_bbs_after_10 <- mean(ct_bbs_after_10$r)
sd_bbs_after_10<- sd(ct_bbs_after_10$r) # feed in a vector
se_bbs_after_10 <- sd_bef/sqrt(nrow(ct_bbs_after_10))

#Make a dataframe with averages - try again later
#
# Dates <- c(2005, 2010,2015)
# Before <- c(ss_bbs_bef, ss_bbs_bef, ss_bbs_bef)
# After <- c(ss_bbs_after_5, ss_bbs_after_10, ss_bbs_after_15)
#
# average_table_bbs <- data.frame(Dates, Before, After)
```

Test if difference between growth rates for CCBER and BBS is significant for california towhee

```
#three tests: before, after 2005, after 2010.

t.test(ct_ccber_before$r, ct_bbs_before$r, conf.level = 0.95, paired= TRUE)
```

```
##
##  Paired t-test
##
## data:  ct_ccber_before$r and ct_bbs_before$r
## t = 0.89219, df = 9, p-value = 0.3955
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.01661345  0.03825263
## sample estimates:
## mean of the differences
##              0.01081959
```

```
t.test(ct_ccber_after$r,ct_bbs_after$r, conf.level = 0.95, paired = TRUE)
```

```
##
##  Paired t-test
##
## data:  ct_ccber_after$r and ct_bbs_after$r
## t = -1.6071, df = 9, p-value = 0.1425
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.06486451  0.01098165
## sample estimates:
## mean of the differences
##             -0.02694143
```

```
t.test(ct_ccber_after_10$r,ct_bbs_after_10$r, conf.level = 0.95, paired = TRUE)
```

```
##
##  Paired t-test
##
## data:  ct_ccber_after_10$r and ct_bbs_after_10$r
## t = -1.3797, df = 4, p-value = 0.2398
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.13144980  0.04417711
## sample estimates:
## mean of the differences
##             -0.04363634
```