

warmXtrophic Project: Greenup Analyses

Kara Dobson, Phoebe Zarnetske

October 04, 2021

Script Details:

```
script_tbl <- data.frame(item = c("OVERVIEW", "COLLABORATORS",
  "REQUIRES", "DATA INPUT", "DATA OUTPUT", "NOTES"), Details = c("This script explores and analyses the
  'Moriah Young, Mark Hammond, Pat Bills', 'Prior to running this script, make sure plant_comp_clean_
  'Data imported as csv files from shared Google drive 'SpaCE_Lab_warmXtrophic' plant comp folder",
  "... a brief description of the data output from through the script, including what format it's in"
  "Each row in 'greenup' is the date at which spp_half_cover_date was recorded, per species. The 'greenup'
  file contains the following information:
  - spp_half_cover_date: the date at which 50% of a species max cover was reached (per plot, per year)",
  "describes each treatment: warmed or ambient"))

kbl(script_tbl) %>% kable_paper(full_width = F) %>% column_spec(1,
  bold = T, border_right = T) %>% column_spec(2, width = "30em",
  background = "lightblue")

metadata_tbl <- data.frame(Variable = c("spp_half_cover_date",
  "plot_half_cover_date", "state"), Definition = c("date at which 50% of a species max cover was reached
  'the date at which 50% of a plot's max cover was reached (per plot, per year)",
  "describes each treatment: warmed or ambient"))

kbl(metadata_tbl) %>% kable_paper(full_width = F) %>% column_spec(1,
  bold = T, border_right = T) %>% column_spec(2, width = "30em",
  background = "lightyellow")

# Clear all existing data
rm(list = ls())

# Load packages
library(tidyverse)
library(ggplot2)
library(lmerTest)
library(olsrr)
library(predictmeans)
library(car)
library(fitdistrplus)
library(ggpubr)
library(rstatix)
library(vegan)
library(interactions)
library(sjPlot)
library(effects)
library(glmTMB)
library(bbmle)
# install.packages('TMB', type='source')
```

```

# Get data
L1_dir <- Sys.getenv("L1DIR")
L2_dir <- Sys.getenv("L2DIR")
greenup <- read.csv(file.path(L2_dir, "greenup/final_greenup_species_L2.csv"))
greenup <- greenup %>% select(-X) # get rid of 'X' column that shows up
greenupp <- read.csv(file.path(L2_dir, "greenup/final_greenup_plot_L2.csv"))
greenupp <- greenupp %>% select(-X) # get rid of 'X' column that shows up

# Set ggplot2 plots to bw: see here for more options:
# http://www.sthda.com/english/wiki/ggplot2-themes-and-background-colors-the-3-elements
theme_set(theme_bw(base_size = 14))
# check variable types
str(greenup)

```

```

## 'data.frame': 2026 obs. of 18 variables:
## $ site : chr "kbs" "kbs" "kbs" "kbs" ...
## $ plot : chr "A1" "A1" "A1" "A1" ...
## $ year : int 2016 2017 2018 2019 2020 2016 2017 2016 2017 2018 ...
## $ species : chr "Acmi" "Acmi" "Acmi" "Acmi" ...
## $ spp_half_cover_date: int 197 101 122 120 127 88 108 97 99 127 ...
## $ min_green_date : int 81 80 122 120 107 81 108 85 80 127 ...
## $ treatment_key : chr "AO" "AO" "AO" "AO" ...
## $ state : chr "ambient" "ambient" "ambient" "ambient" ...
## $ insecticide : chr "no_insects" "no_insects" "no_insects" "no_insects" ...
## $ scientific_name : chr "Achillea millefolium" "Achillea millefolium" "Achillea millefolium" "A
## $ common_name : chr "common yarrow" "common yarrow" "common yarrow" "common yarrow" ...
## $ USDA_species : chr "ACMI2" "ACMI2" "ACMI2" "ACMI2" ...
## $ LTER_species : chr "ACHMI" "ACHMI" "ACHMI" "ACHMI" ...
## $ origin : chr "Native" "Native" "Native" "Native" ...
## $ group : chr "Dicot" "Dicot" "Dicot" "Dicot" ...
## $ family : chr "Fabaceae" "Fabaceae" "Fabaceae" "Fabaceae" ...
## $ duration : chr "Biennial" "Biennial" "Biennial" "Biennial" ...
## $ growth_habit : chr "Forb" "Forb" "Forb" "Forb" ...

```

```

# Order warm and ambient so that warm shows up first in
# plotting (and is default is red = warm; blue = ambient).
# First make it a factor.

```

```

greenup$state <- as.factor(greenup$state)
levels(greenup$state)

```

```

## [1] "ambient" "warmed"

```

```

greenup$state <- factor(greenup$state, levels(greenup$state)[c(2,
1)])
levels(greenup$state)

```

```

## [1] "warmed" "ambient"

```

```

greenupp$state <- as.factor(greenupp$state)
levels(greenupp$state)

```

```

## [1] "ambient" "warmed"

```

```

greenupp$state <- factor(greenupp$state, levels(greenupp$state)[c(2,
1)])
levels(greenupp$state)

```

```
## [1] "warmed" "ambient"

# adding sequential year variable starting at 1: this is
# because 2016... are large numbers compare with other values
# in the dataset. We can always label axes with these real
# years.
greenup$year_factor[greenup$year == 2016] <- 1
greenup$year_factor[greenup$year == 2017] <- 2
greenup$year_factor[greenup$year == 2018] <- 3
greenup$year_factor[greenup$year == 2019] <- 4
greenup$year_factor[greenup$year == 2020] <- 5

greenupp$year_factor[greenupp$year == 2016] <- 1
greenupp$year_factor[greenupp$year == 2017] <- 2
greenupp$year_factor[greenupp$year == 2018] <- 3
greenupp$year_factor[greenupp$year == 2019] <- 4
greenupp$year_factor[greenupp$year == 2020] <- 5

# create dataframes for kbs and umbs - remember that these
# contain species within plots
green_kbs <- subset(greenup, site == "kbs")
green_umbs <- subset(greenup, site == "umbs")

green_kbsp <- subset(greenupp, site == "kbs")
green_umbsp <- subset(greenupp, site == "umbs")
```

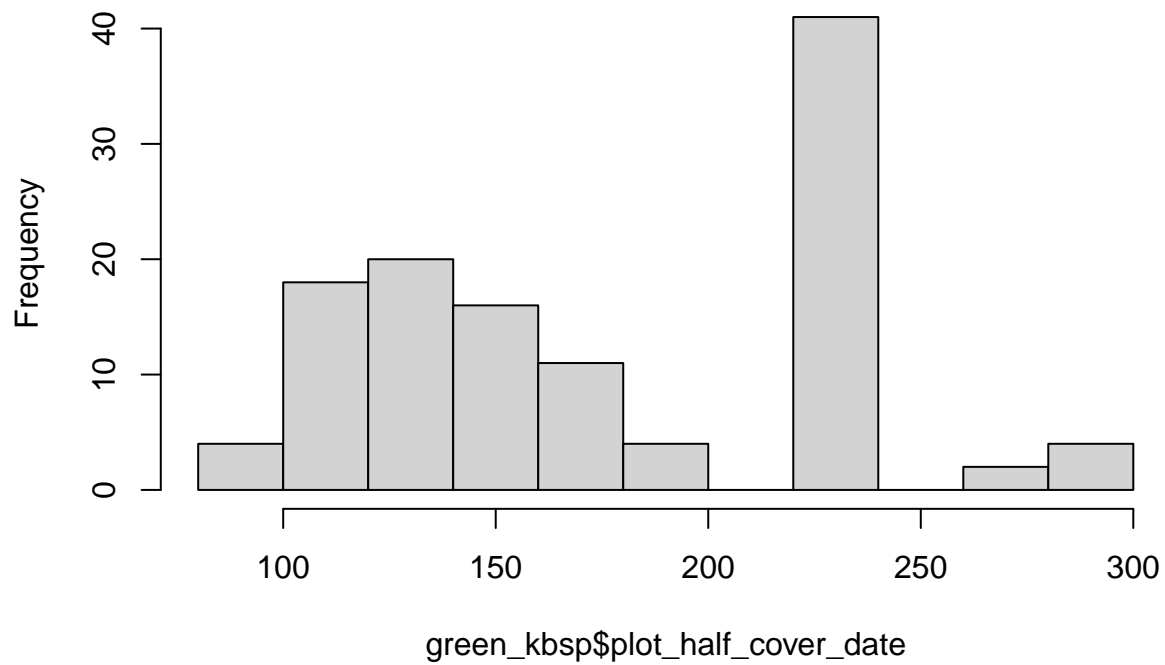
Data Exploration: are there differences between warmed vs. ambient plots when we account for species?

Starting with KBS

First, checking for normality in raw data. It's not going to tell you about normality once you fit a model to these data - that's when you really need to investigate the residuals.

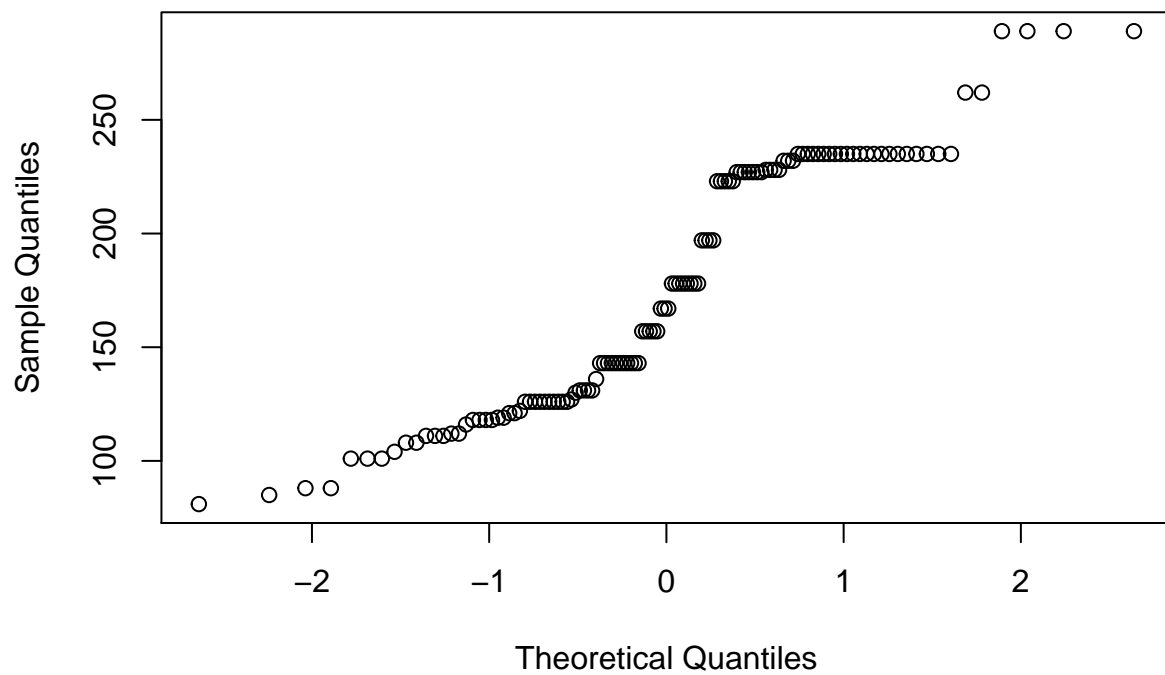
```
hist(green_kbsp$plot_half_cover_date)
```

Histogram of green_kbsp\$plot_half_cover_date



```
qqnorm(green_kbsp$plot_half_cover_date)
```

Normal Q-Q Plot



```
shapiro.test(green_kbsp$plot_half_cover_date)
```

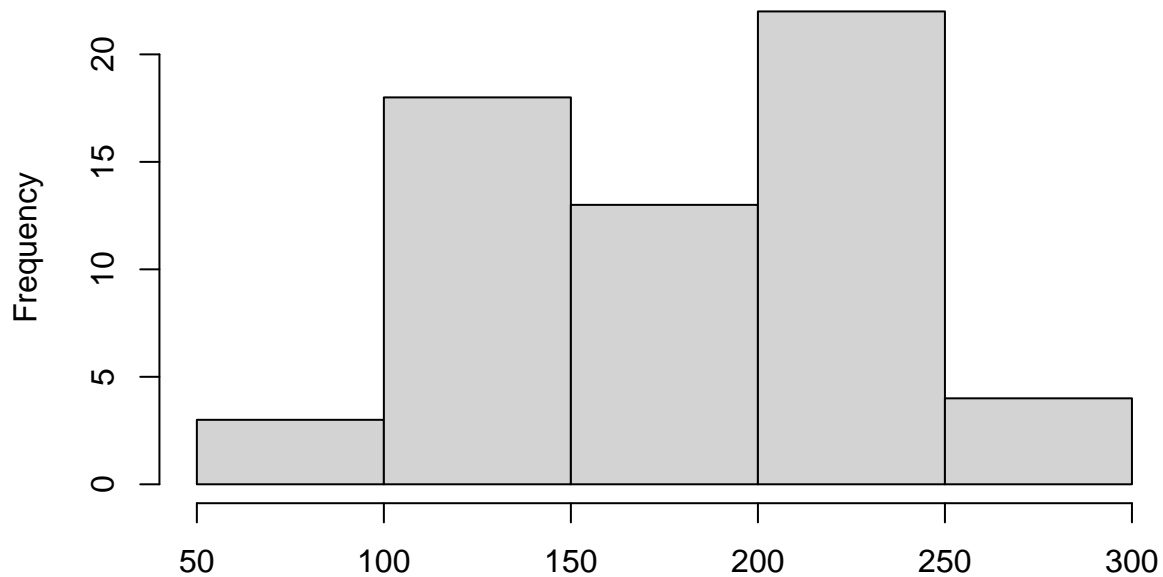
```
##
```

```
## Shapiro-Wilk normality test
##
## data: green_kbsp$plot_half_cover_date
## W = 0.90721, p-value = 4.673e-07
```

```
# histograms for each treatment separately
```

```
hist(green_kbsp$plot_half_cover_date[green_kbsp$state == "ambient"])
```

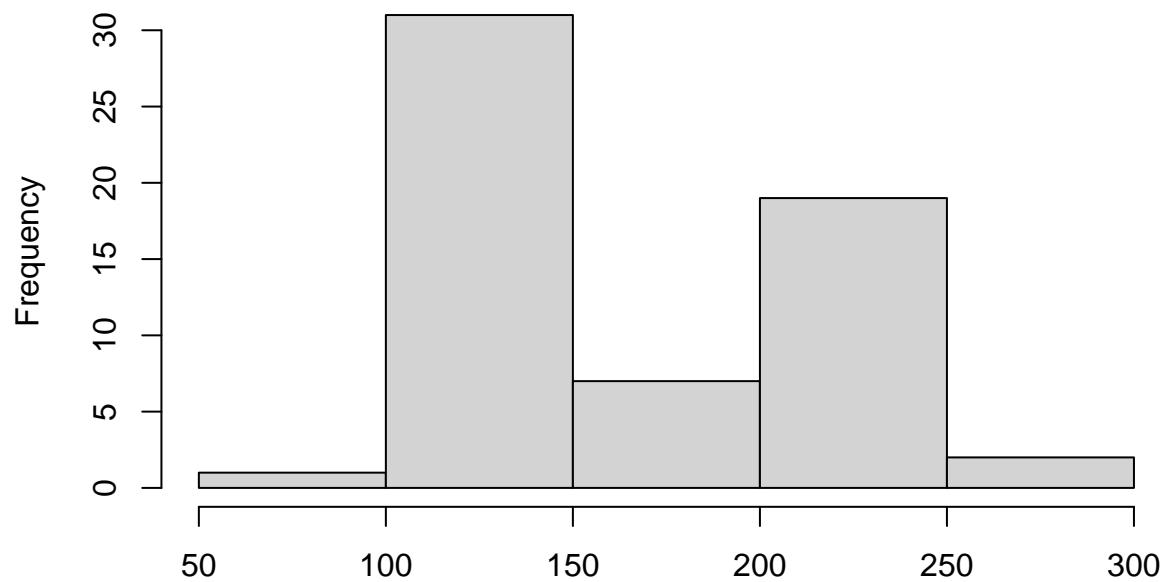
istogram of green_kbsp\$plot_half_cover_date[green_kbsp\$state == "ar



```
green_kbsp$plot_half_cover_date[green_kbsp$state == "ambient"]
```

```
hist(green_kbsp$plot_half_cover_date[green_kbsp$state == "warmed"])
```

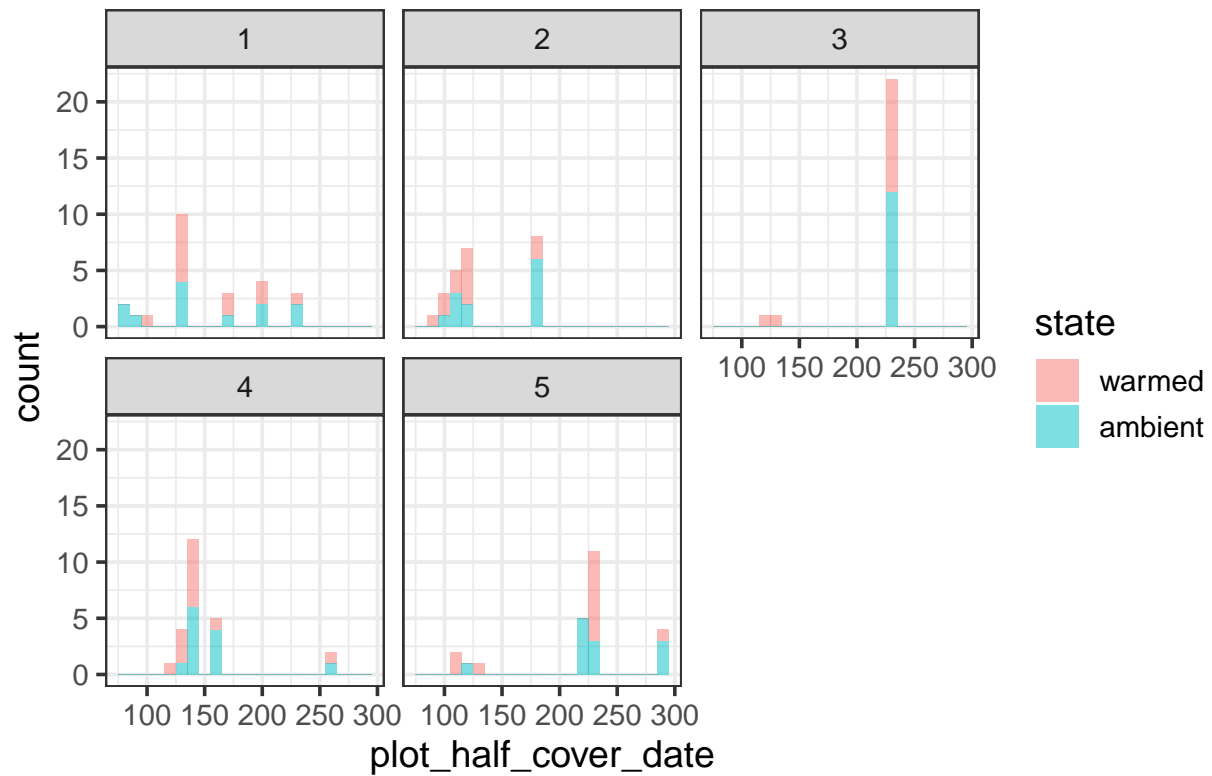
istogram of green_kbsp\$plot_half_cover_date[green_kbsp\$state == "wa



green_kbsp\$plot_half_cover_date[green_kbsp\$state == "warmed"]

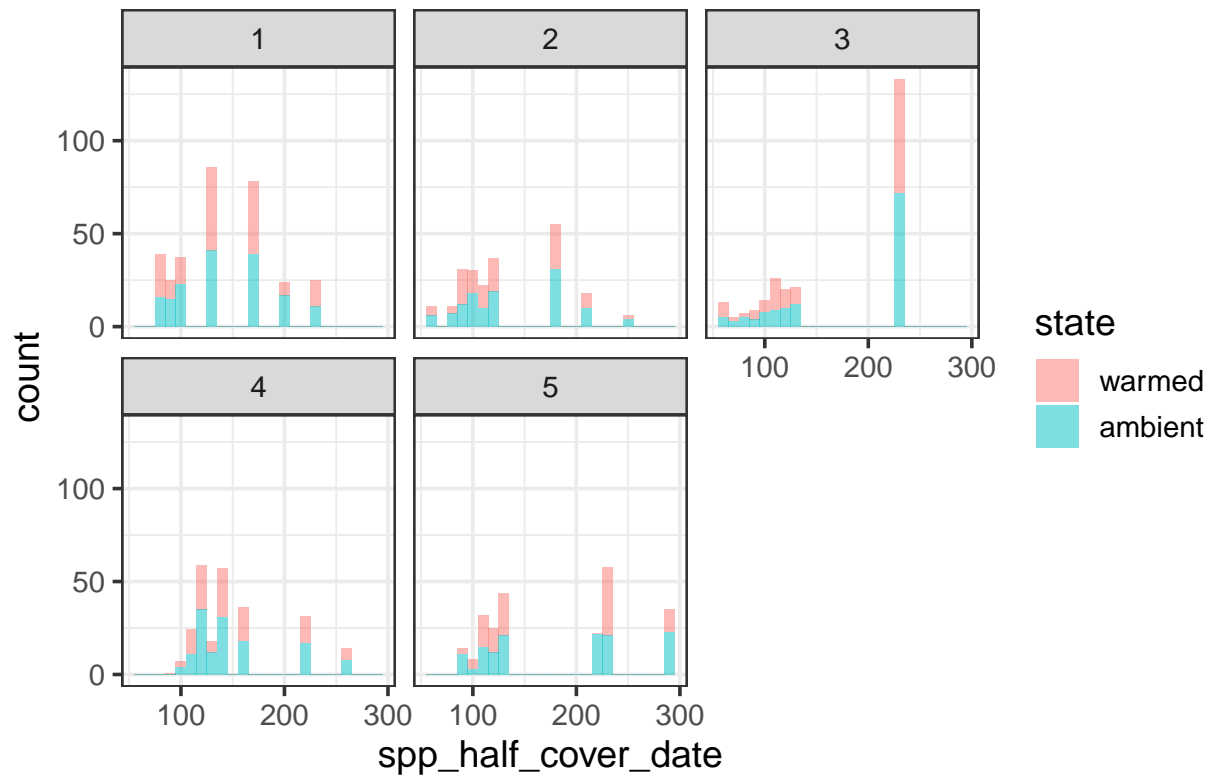
```
# histograms for each year - look at them together:
p1 <- ggplot(data = green_kbsp, aes(x = plot_half_cover_date,
  fill = state)) + geom_histogram(alpha = 0.5, binwidth = 10)
p1 + facet_wrap(~year_factor) + labs(title = "Plot-level half cover date")
```

Plot-level half cover date

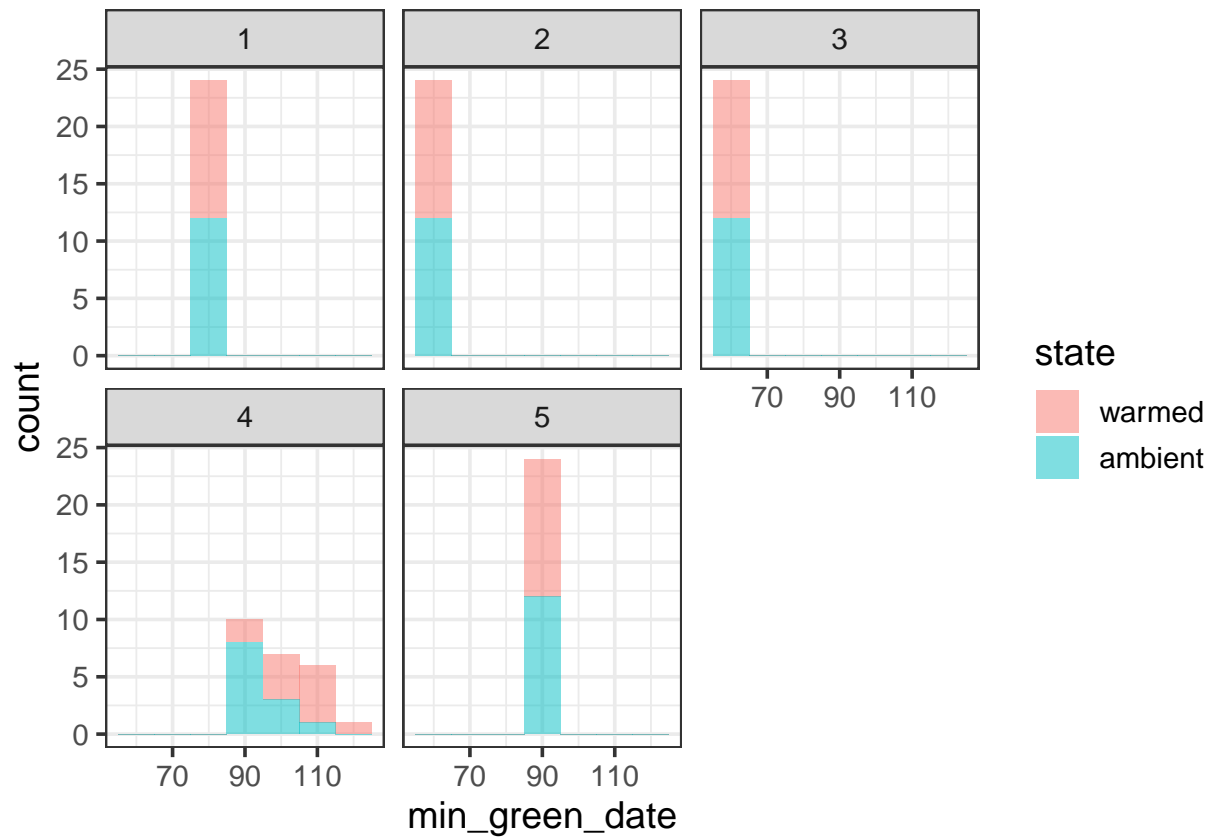


```
p1 <- ggplot(data = green_kbs, aes(x = spp_half_cover_date, fill = state)) +
  geom_histogram(alpha = 0.5, binwidth = 10)
p1 + facet_wrap(~year_factor) + labs(title = "Species-level half cover date")
```

Species-level half cover date

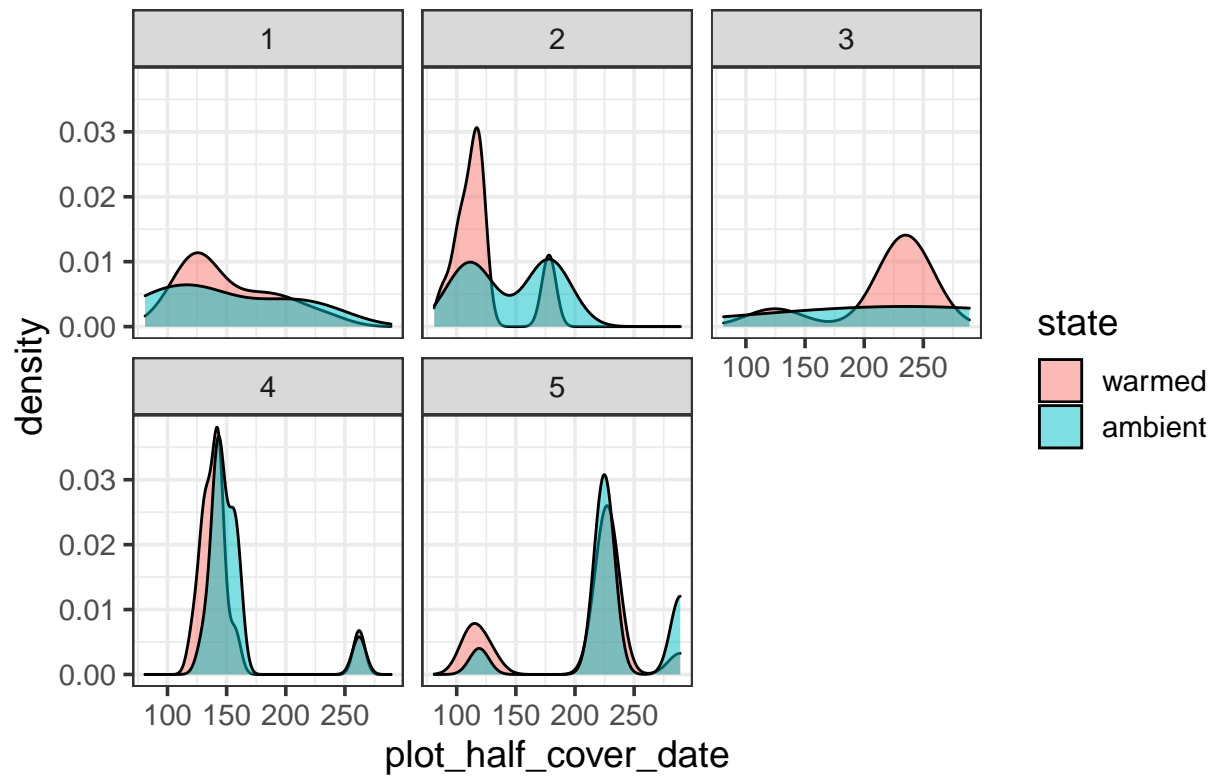


```
# this will just show sampling date artifact
p2 <- ggplot(data = green_kbsp, aes(x = min_green_date, fill = state)) +
  geom_histogram(alpha = 0.5, binwidth = 10)
p2 + facet_wrap(~year_factor)
```

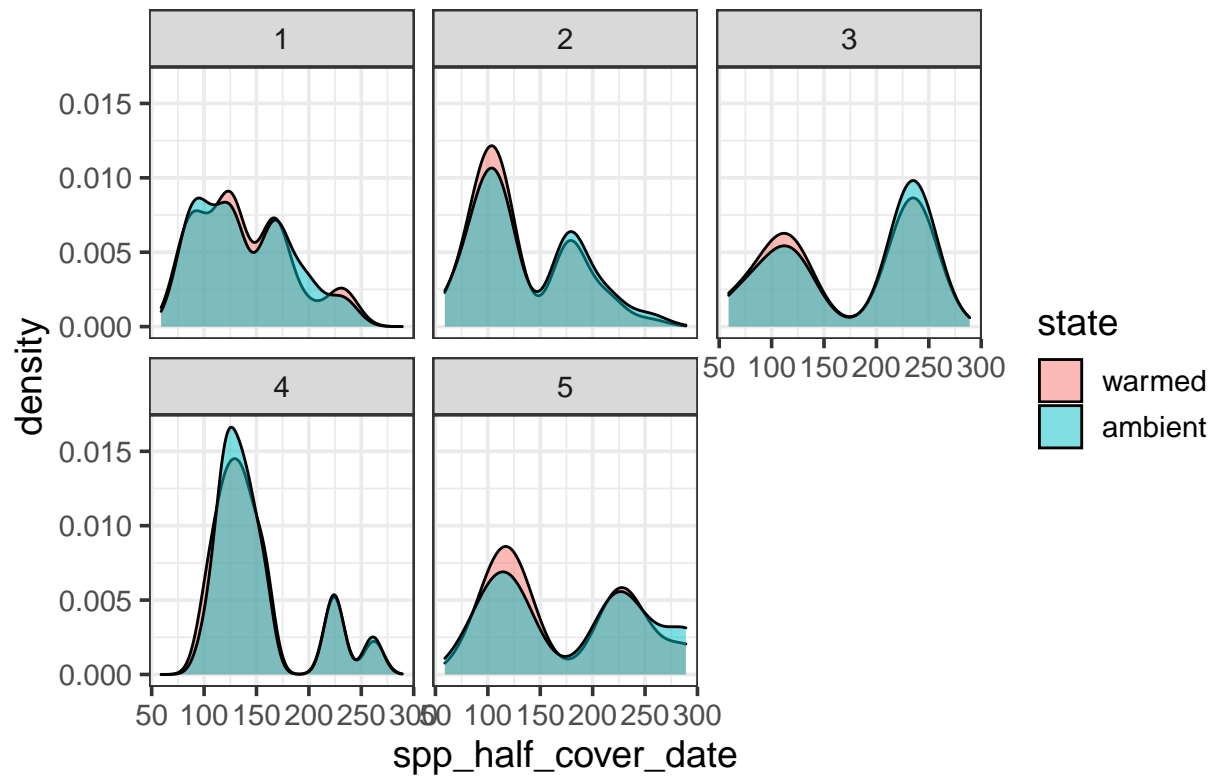
```
# Density plot
p3 <- ggplot(data = green_kbsp, aes(x = plot_half_cover_date,
  fill = state)) + geom_density(alpha = 0.5)
p3 + facet_wrap(~year_factor) + labs(title = "Plot-level half cover date")
```

Plot-level half cover date

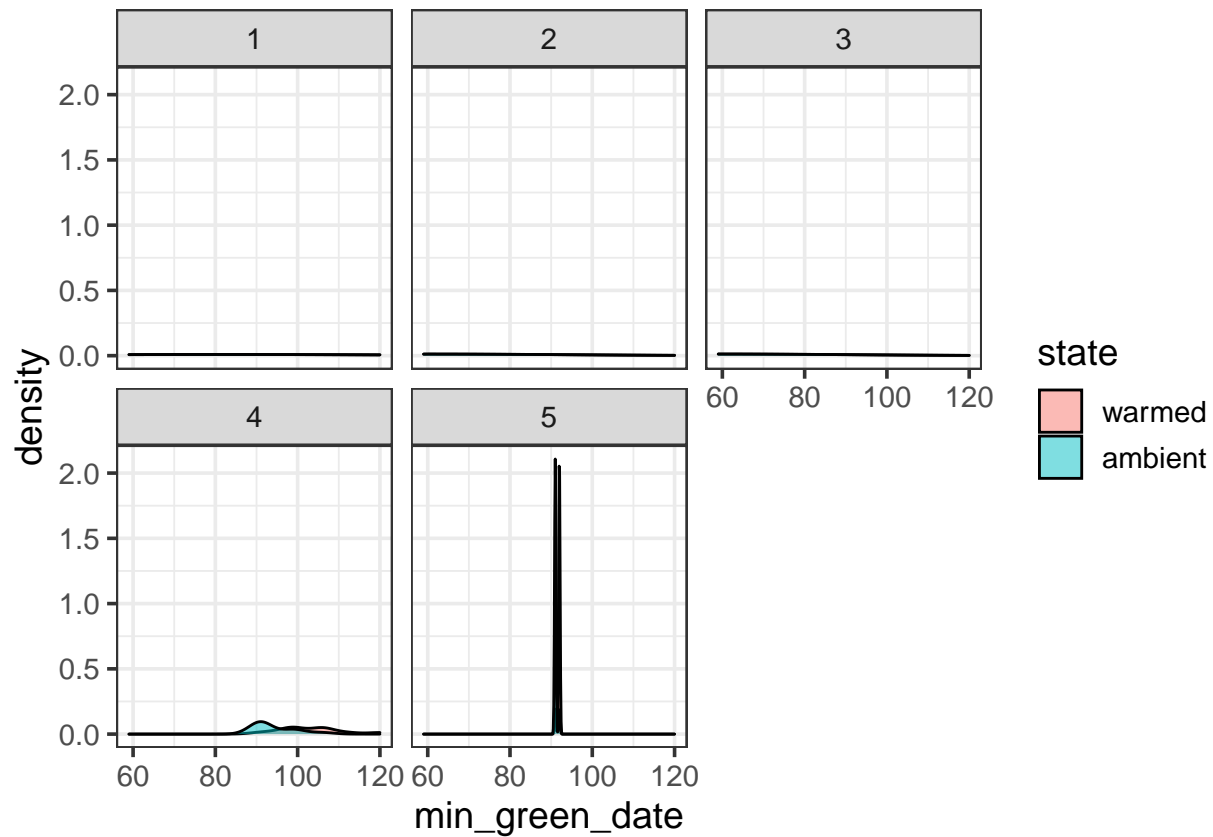


```
p3 <- ggplot(data = green_kbs, aes(x = spp_half_cover_date, fill = state)) +
  geom_density(alpha = 0.5)
p3 + facet_wrap(~year_factor) + labs(title = "Species-level half cover date")
```

Species-level half cover date



```
# this will just show sampling date artifact
p4 <- ggplot(data = green_kbsp, aes(x = min_green_date, fill = state)) +
  geom_density(alpha = 0.5)
p4 + facet_wrap(~year_factor)
```



```
# Or try with tidyverse format
green_kbsp.t <- as_tibble(green_kbsp)
# green_kbsp.t %>% gather(state, plot_half_cover_date,
# year_factor) %>% ggplot(aes(plot_half_cover_date, fill =
# state)) + geom_histogram() + facet_wrap(~year_factor)

# looks like the 225 spike is from 2018 and 2020 - what's
# going on here is that you are treating all species-plot
# records as independent observations, so the influence of
# species differences is likely coming through here.
kbs_2018 <- subset(green_kbs, year == 4) # many records on 235
kbs_2020 <- subset(green_kbs, year == 6) # records from 227 & 228
```

Leverage plots and detecting Outliers. <https://www.statmethods.net/stats/riagnostics.html>

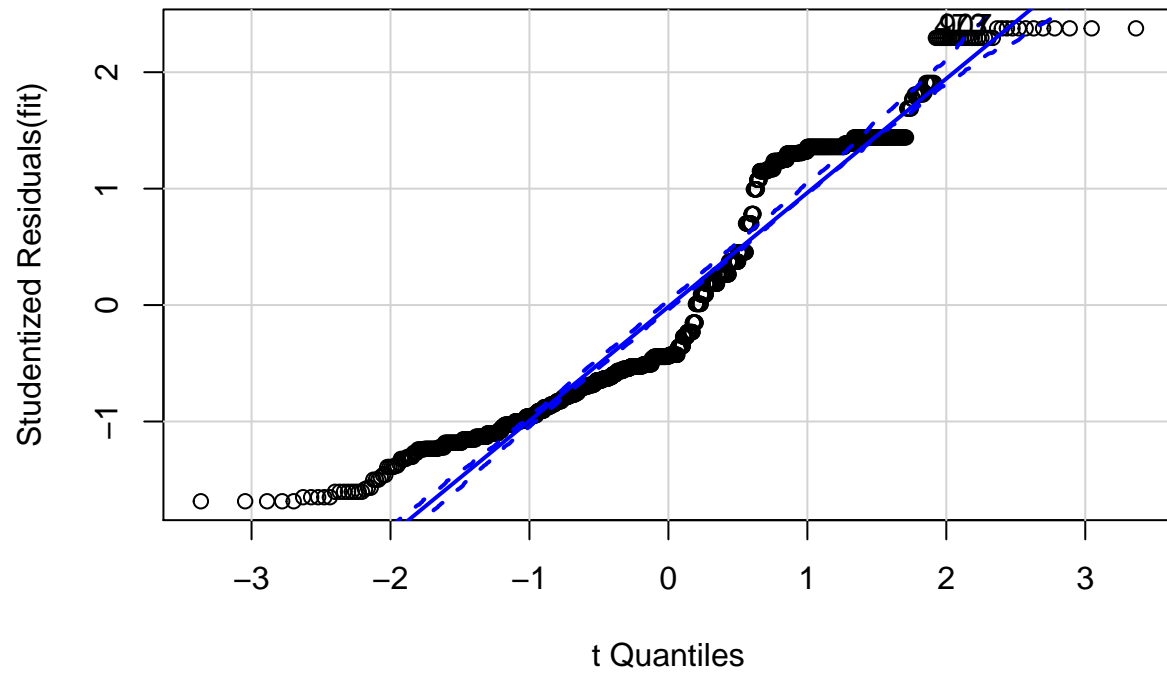
These illustrate whether certain data points have more leverage (more influence), and thus could be outliers. It's a way of detecting outliers. Leverage plots can help identify whether a point has high or low influence, based on its leverage and residual and determining model fit with and without the point in question. Ultimately you decide whether the points are outliers or not, based on the knowledge of the system and how much it changes the model when included vs. excluded from the data used to fit the model. Here is a good overview of the combination of leverage and residual: scroll down to sections beginning at "13.3 Unusual Observations": <https://davidalpiaz.github.io/appliedstats/model-diagnostics.html>

```
# checking fit for date as a function of state and species -  
# bringing in species here makes it obvious that that is  
# explaining some of the variation compared with the  
# state-only model you had previously.
```

```
# State-only model  
fit <- lm(spp_half_cover_date ~ state, data = green_kbs)  
outlierTest(fit) # no outliers
```

```
## No Studentized residuals with Bonferroni p < 0.05  
## Largest |rstudent|:  
##      rstudent unadjusted p-value Bonferroni p  
## 473 2.376821      0.017611      NA  
qqPlot(fit, main = "QQ Plot")
```

QQ Plot

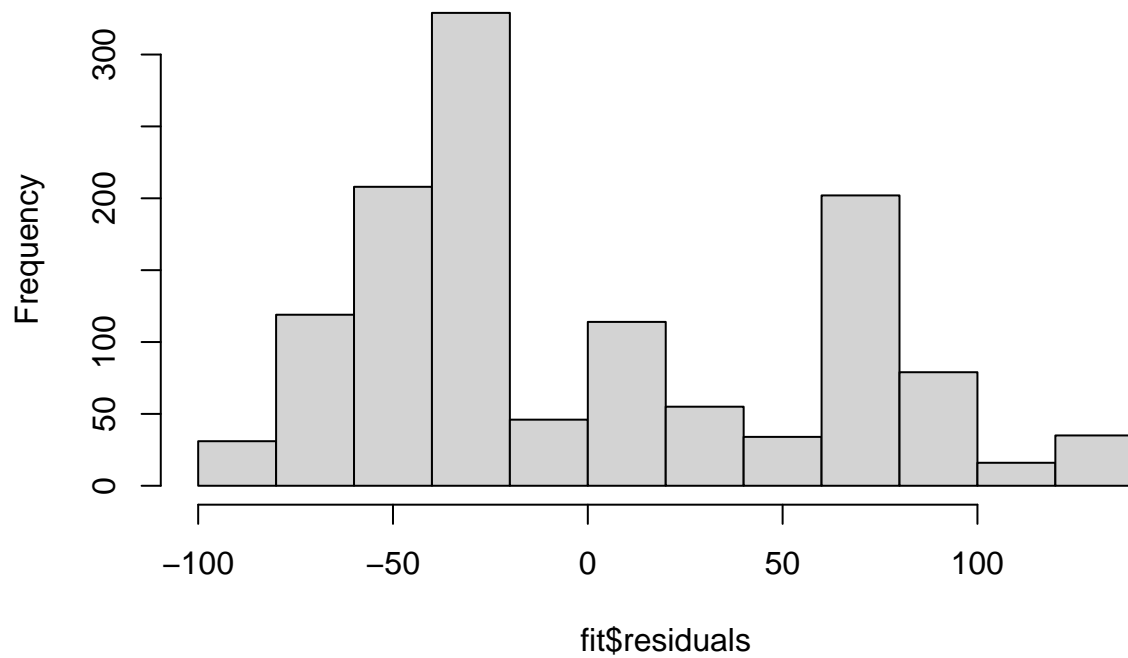


```
## 473 907
```

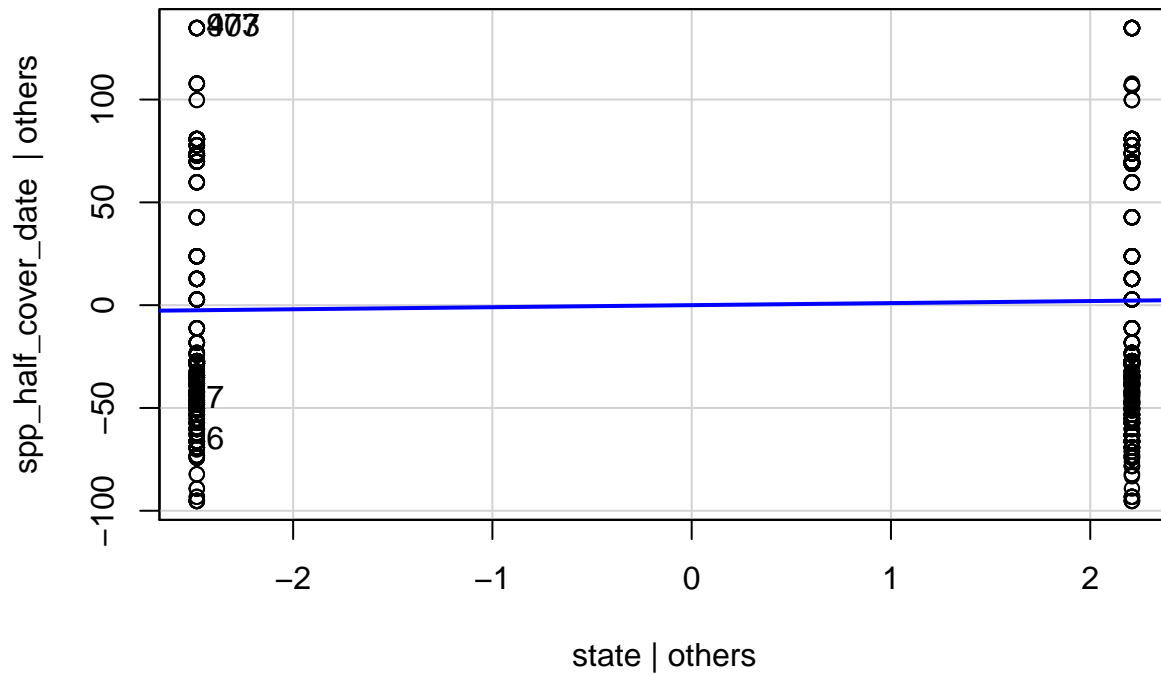
```
## 283 552
```

```
hist(fit$residuals)
```

Histogram of fit\$residuals



```
leveragePlots(fit)
```



```
# State and species model
```

```
fit1 <- lm(spp_half_cover_date ~ state + species, data = green_kbs)
```

```
outlierTest(fit1) # no outliers
```

```
## No Studentized residuals with Bonferroni p < 0.05
```

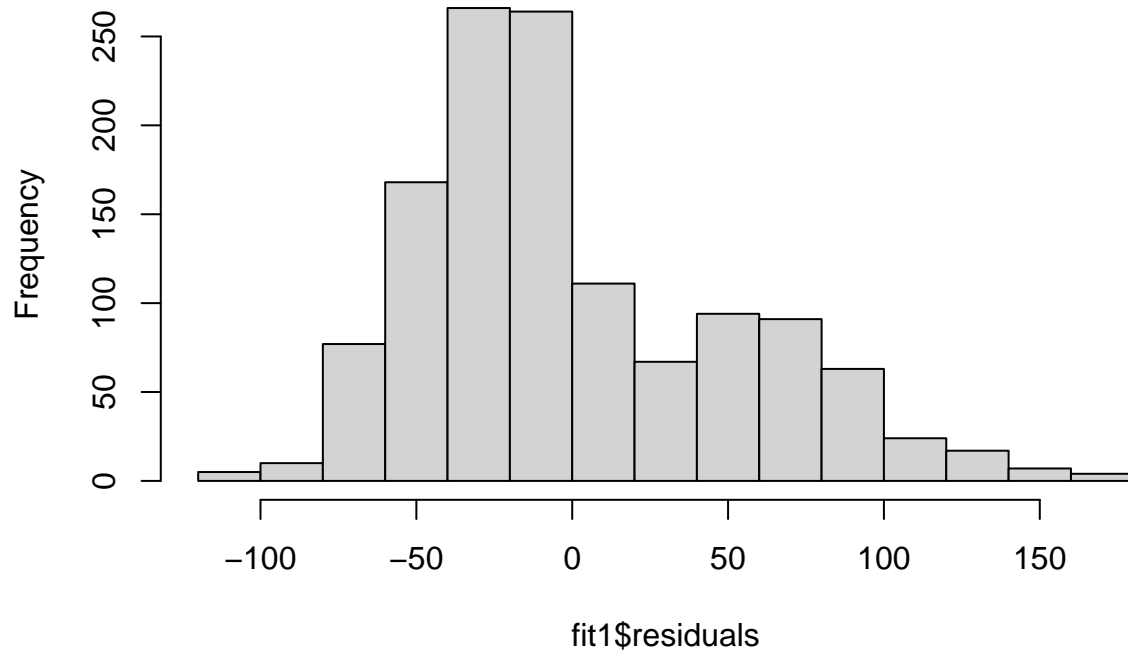
```
## Largest |rstudent|:
```

```
##      rstudent unadjusted p-value Bonferroni p
```

```
## 1910 3.455976      0.00056677      0.71866
```

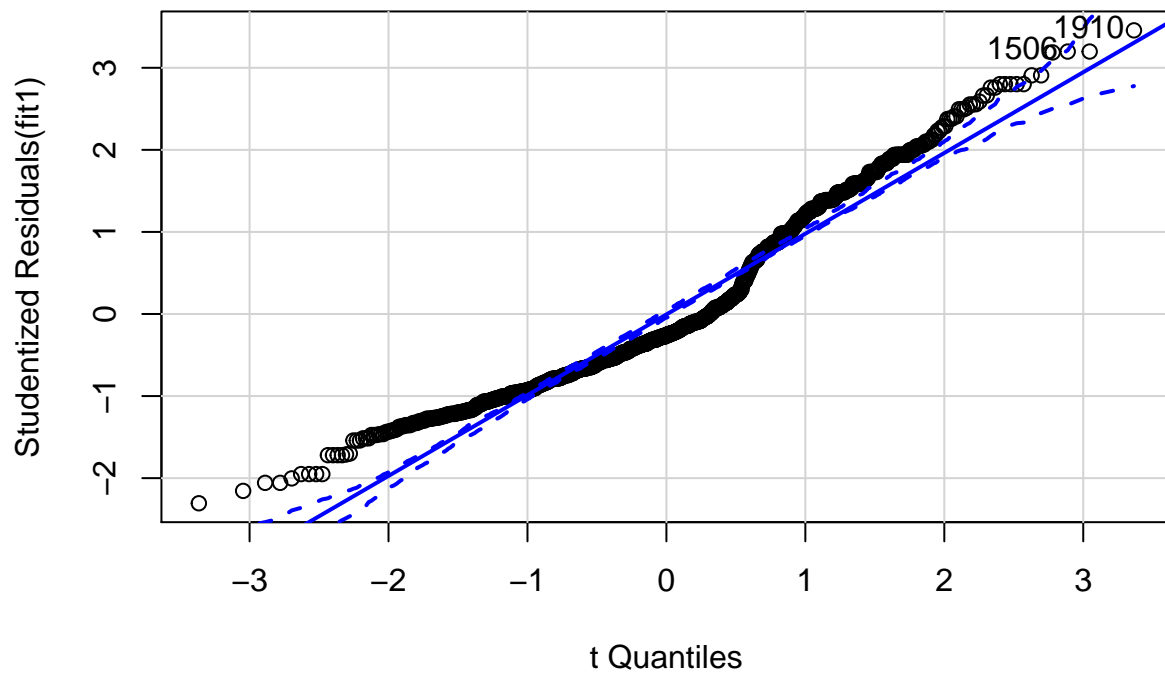
```
hist(fit1$residuals)
```

Histogram of fit1\$residuals



```
qqPlot(fit1, main = "QQ Plot")
```

QQ Plot

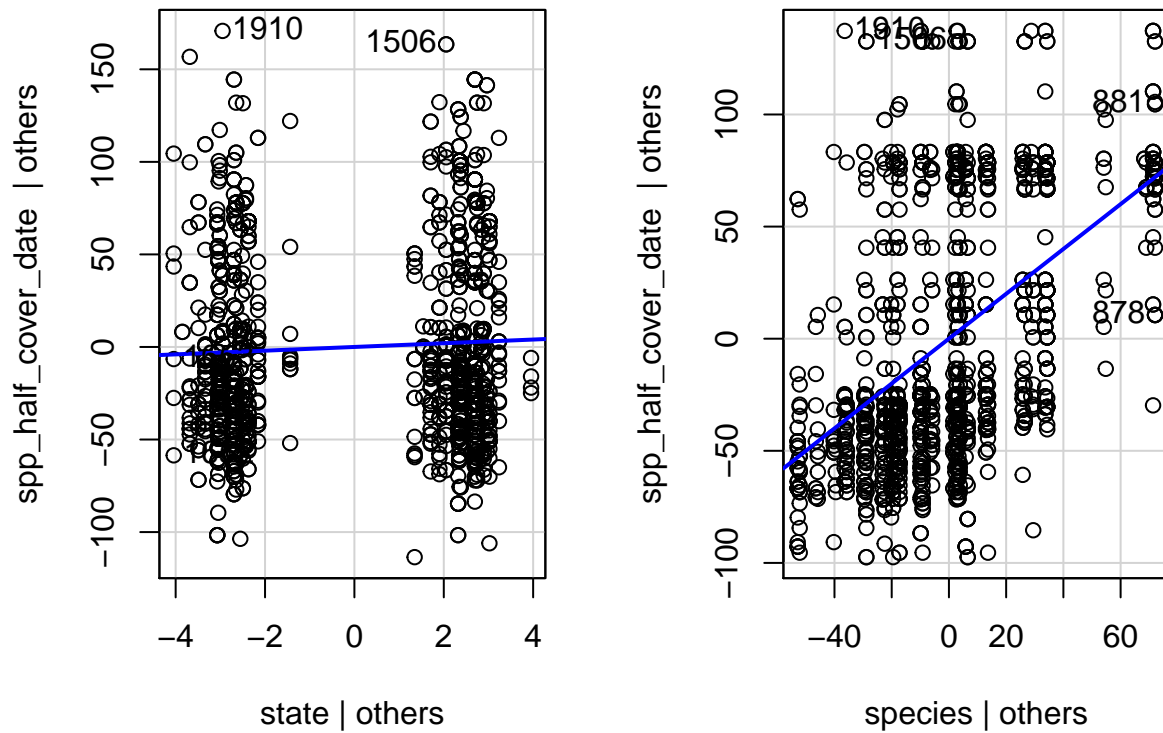


```
## 1506 1910  
## 943 1152
```



```
leveragePlots(fit1)
```

Leverage Plots

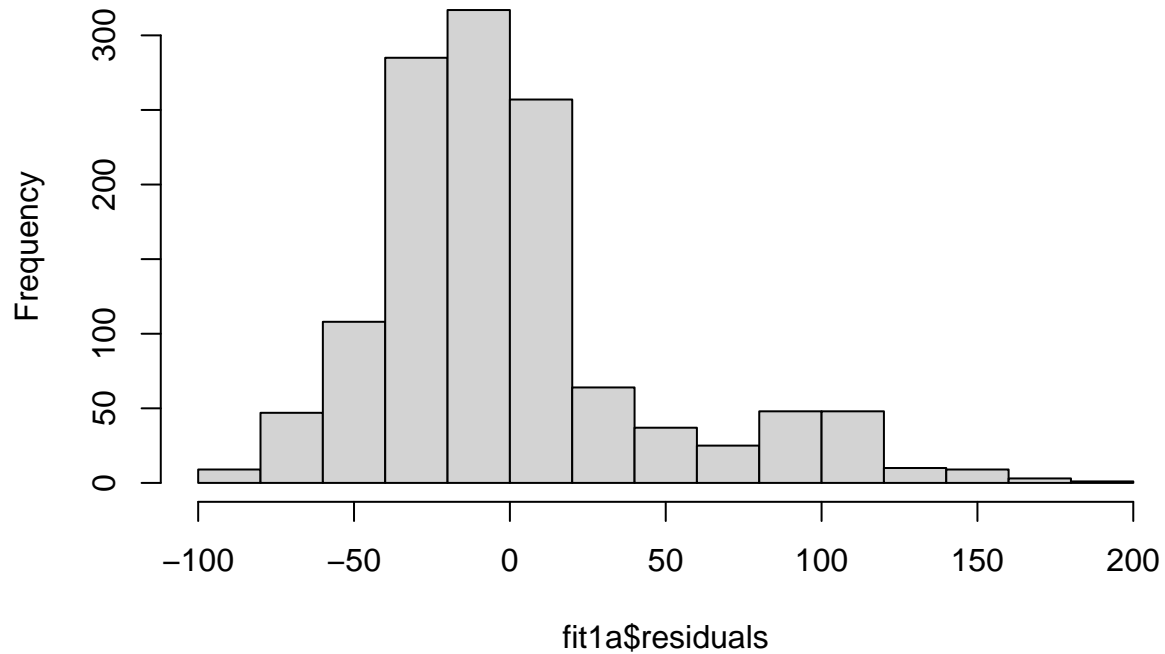


```
fit1a <- lm(min_green_date ~ state + species, data = green_kbs)
outlierTest(fit1a) # no outliers
```

```
##      rstudent unadjusted p-value Bonferroni p
## 473 4.211351      2.7212e-05      0.034505
```

```
hist(fit1a$residuals)
```

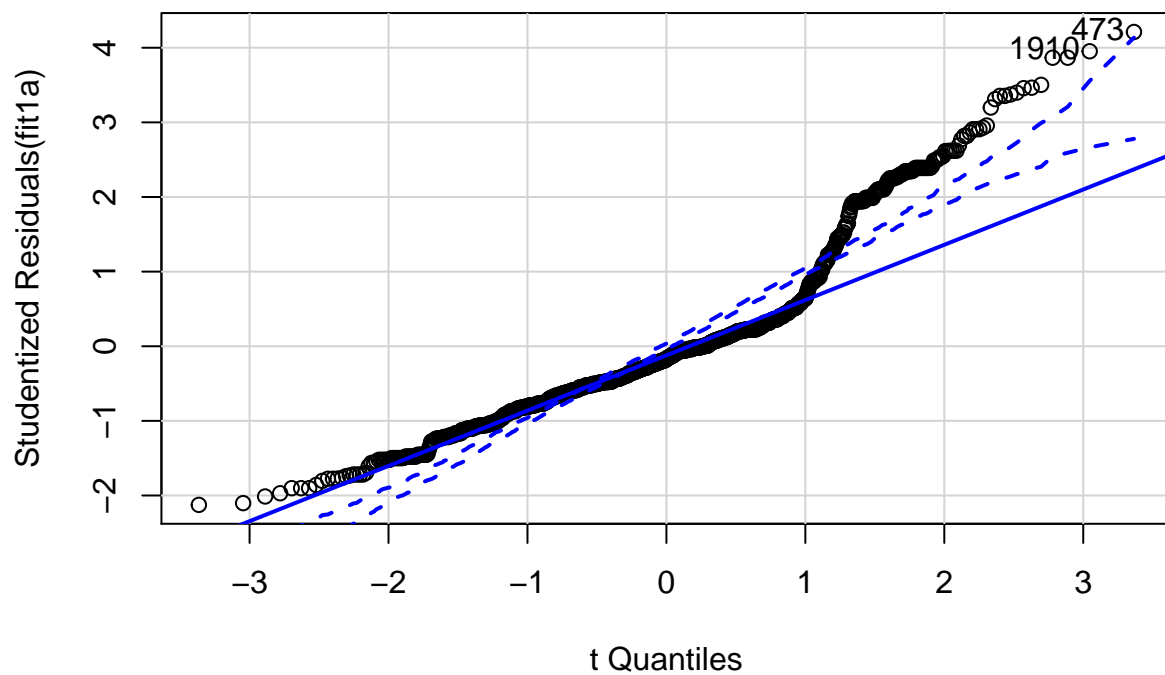
Histogram of fit1a\$residuals



```
qqPlot(fit1a, main = "QQ Plot")
```

```
## Warning in rlm.default(x, y, weights, method = method, wt.method = wt.method, :  
## 'rlm' failed to converge in 20 steps
```

QQ Plot

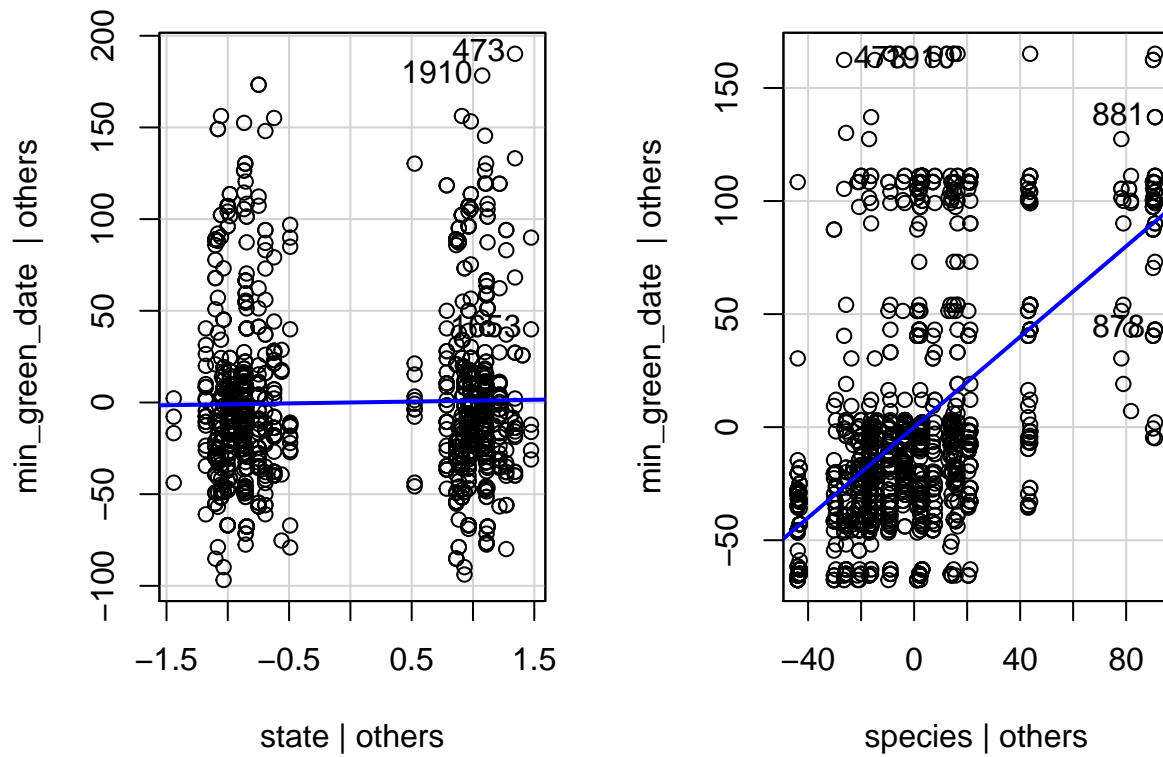


```
## 473 1910
```

```
## 283 1152
```

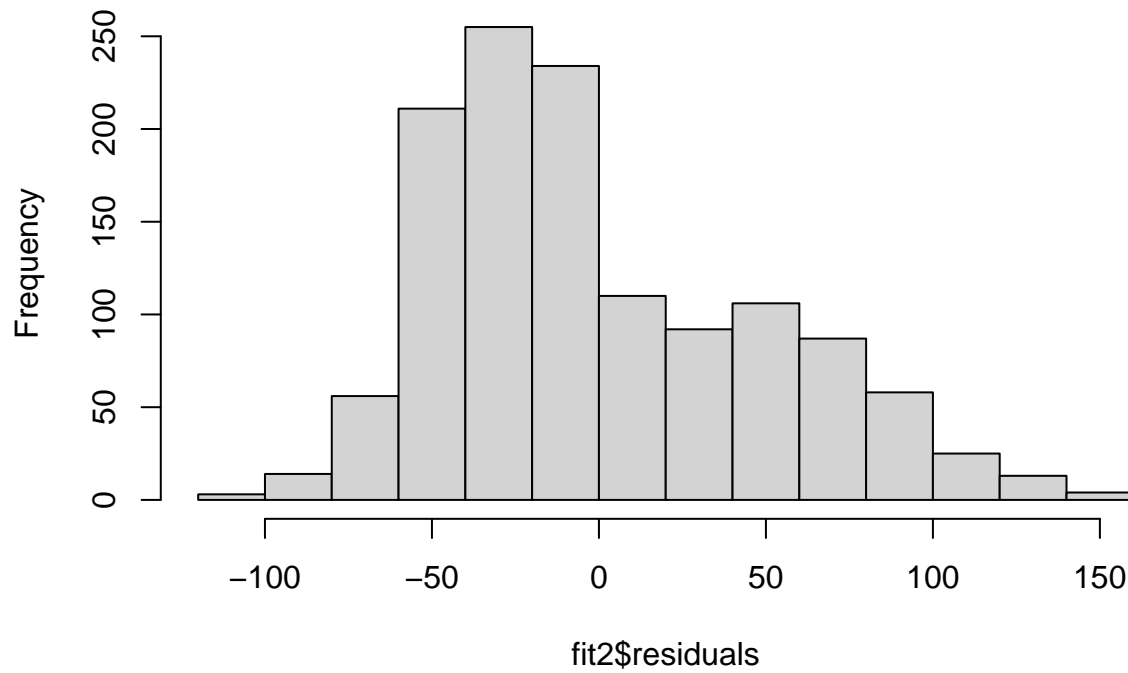
```
leveragePlots(fit1a)
```

Leverage Plots



```
# checking fit for date as a function of state and year  
fit2 <- lm(spp_half_cover_date ~ state + species + year, data = green_kbs)  
hist(fit2$residuals)
```

Histogram of fit2\$residuals

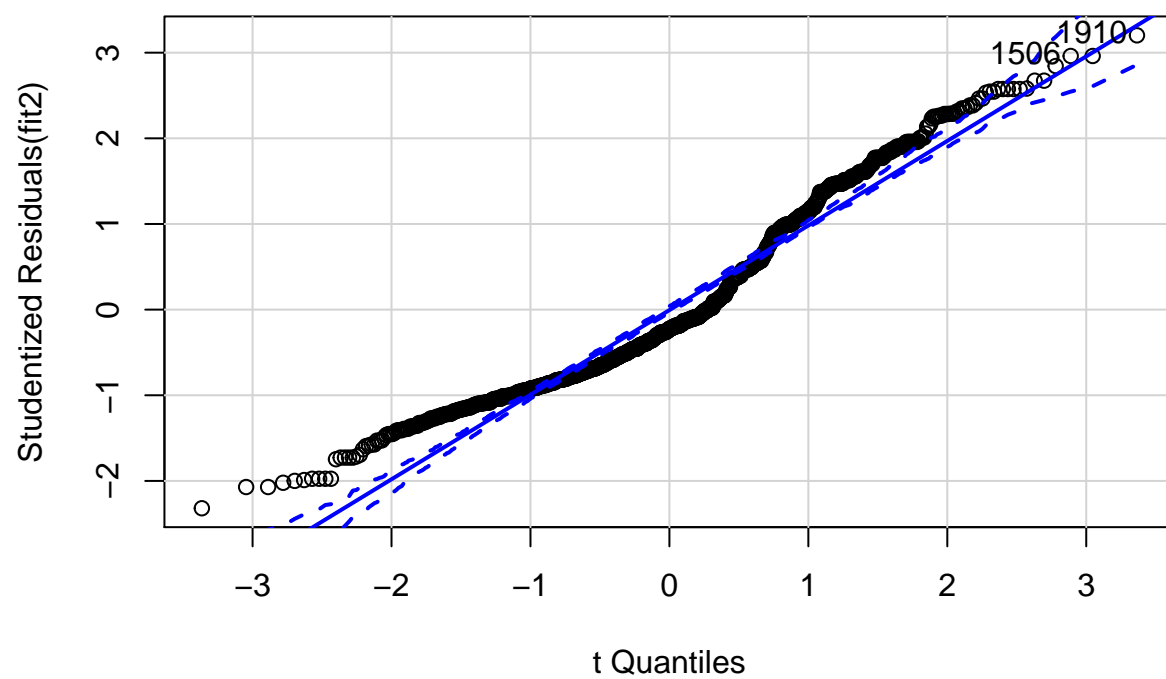


```
outlierTest(fit2) # no outliers
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##      rstudent unadjusted p-value Bonferroni p
## 1910 3.202366      0.0013976      NA
```

```
qqPlot(fit2, main = "QQ Plot")
```

QQ Plot

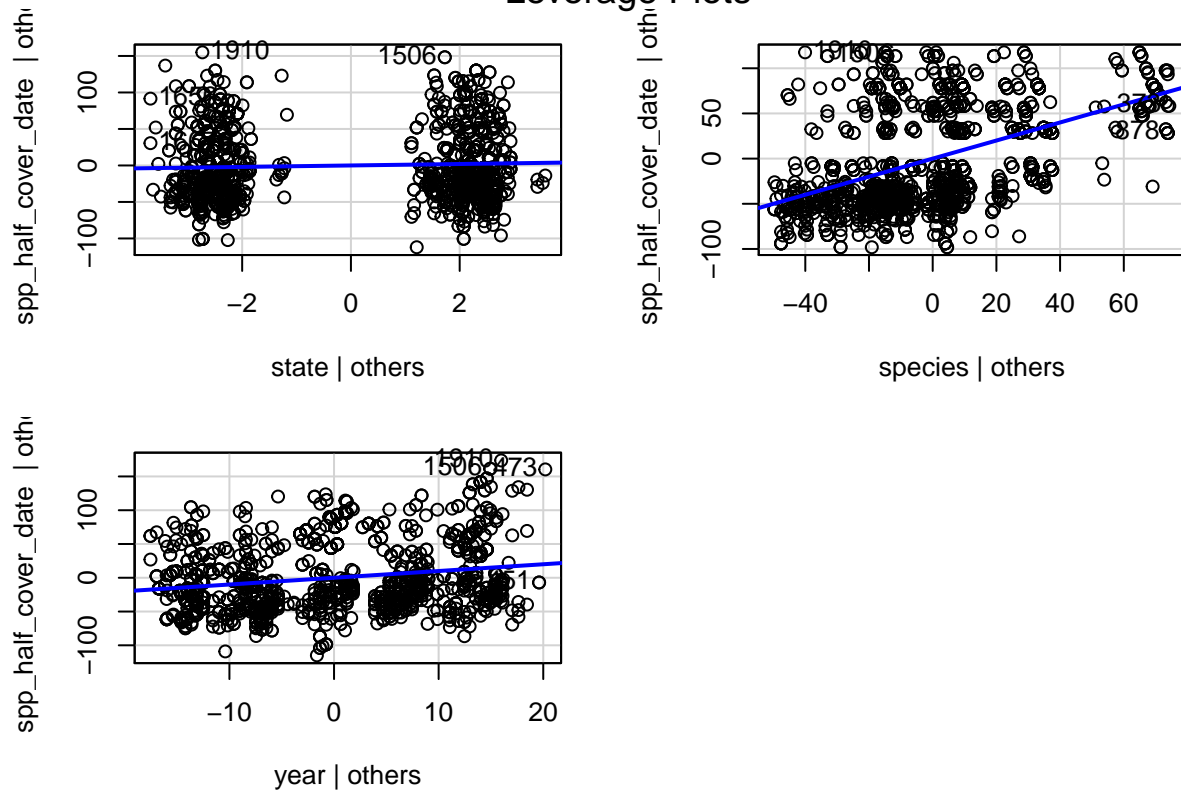


```
## 1506 1910
```

```
## 943 1152
```

```
leveragePlots(fit2)
```

Leverage Plots



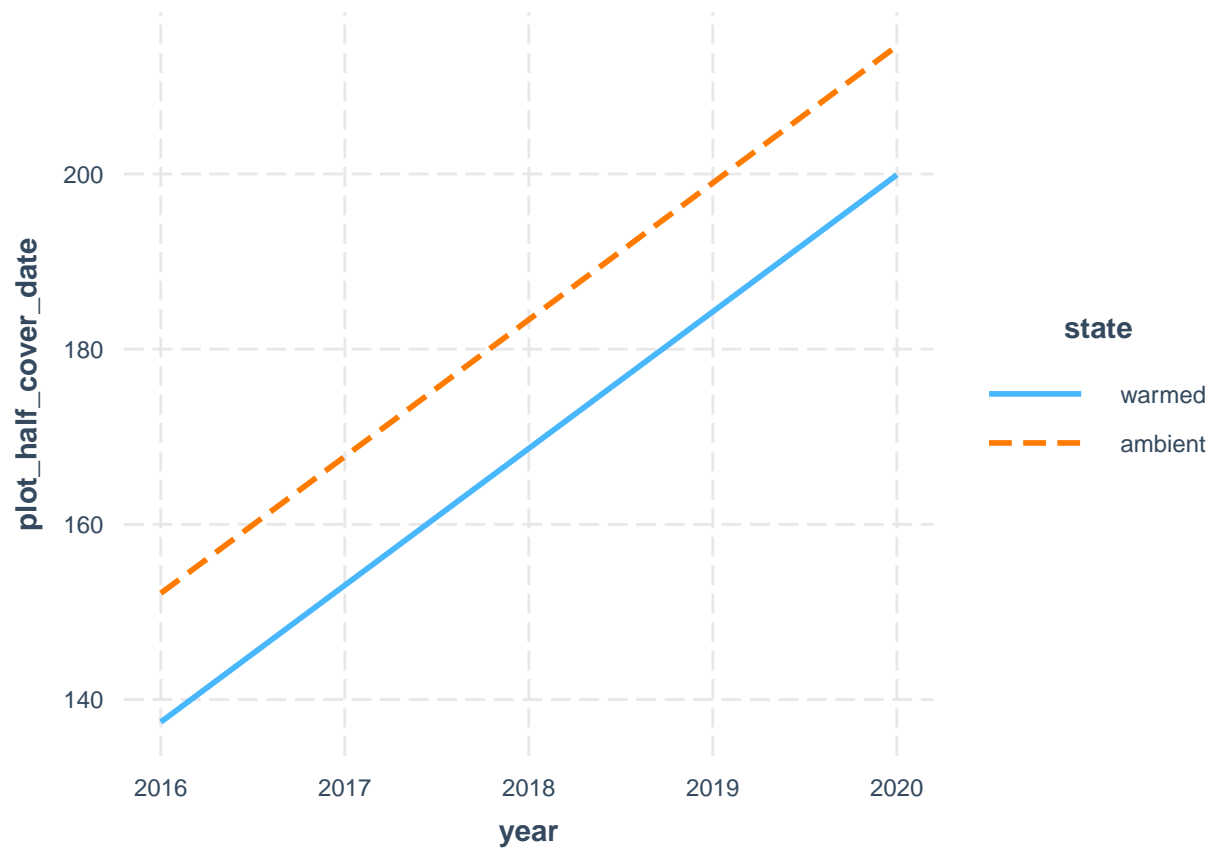
Normal distribution after accounting for species - we will be using species as a random effect to account for their variation. Set up some linear mixed effects models to evaluate. From Ben Bolker: “The traditional view of random effects is as a way to do correct statistical tests when some observations are correlated. ... Random effects are especially useful when we have (1) lots of levels (e.g., many species or blocks), (2) relatively little data on each level (although we need multiple samples from most of the levels), and (3) uneven sampling across levels. People sometimes say that random effects are “factors that you aren’t interested in.” This is not always true. While it is often the case in ecological experiments (where variation among sites is usually just a nuisance), it is sometimes of great interest.” In our case, variation among plots is a nuisance, and not something we’re interested in. For some questions, variation among species is also a nuisance for us. It’s possible that variation among years is a nuisance if we only care about warm vs. ambient, but I think time is an interesting variable to consider with this study.

We should also think about how we’re treating year. Some of the models have a state * year interaction as a fixed effect, which means that the warming or ambient treatment could affect the half_cover_date differently over time (there would be a different slope for each state in the relationship between half_cover_date (y) and year (x)). If we just had state + year, the states would have the same slope, indicating that they have no interaction in their effect on half_cover_date (but they could still have different intercepts).

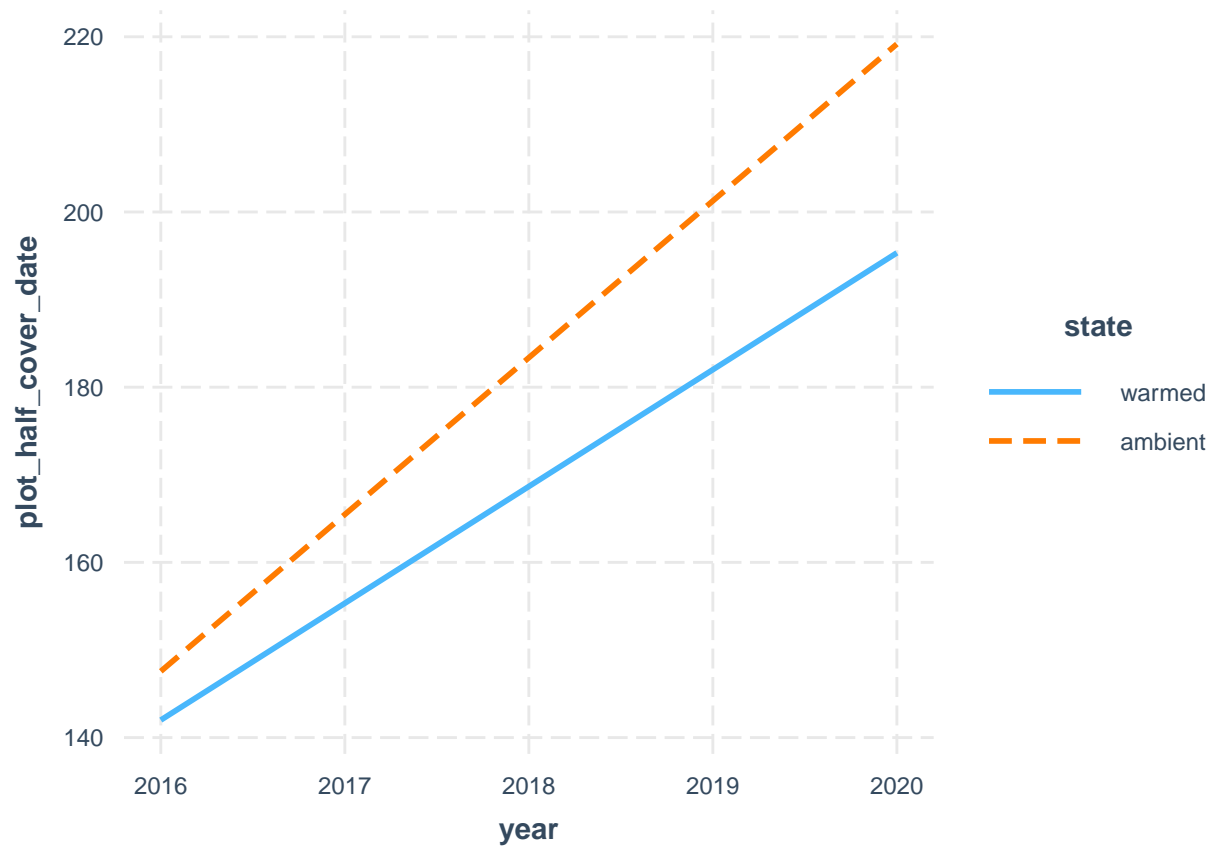
```
# Interaction plot (ignore for now the repeated measures with
# species); see:
# https://cran.r-project.org/web/packages/interactions/vignettes/interactions.html
# and: https://interactions.jacob-long.com/

fit3 <- lm(plot_half_cover_date ~ state + year, data = green_kbsp)
interact_plot(fit3, pred = year, modx = state)

## Warning: year and state are not included in an interaction with one another in the
## model.
```

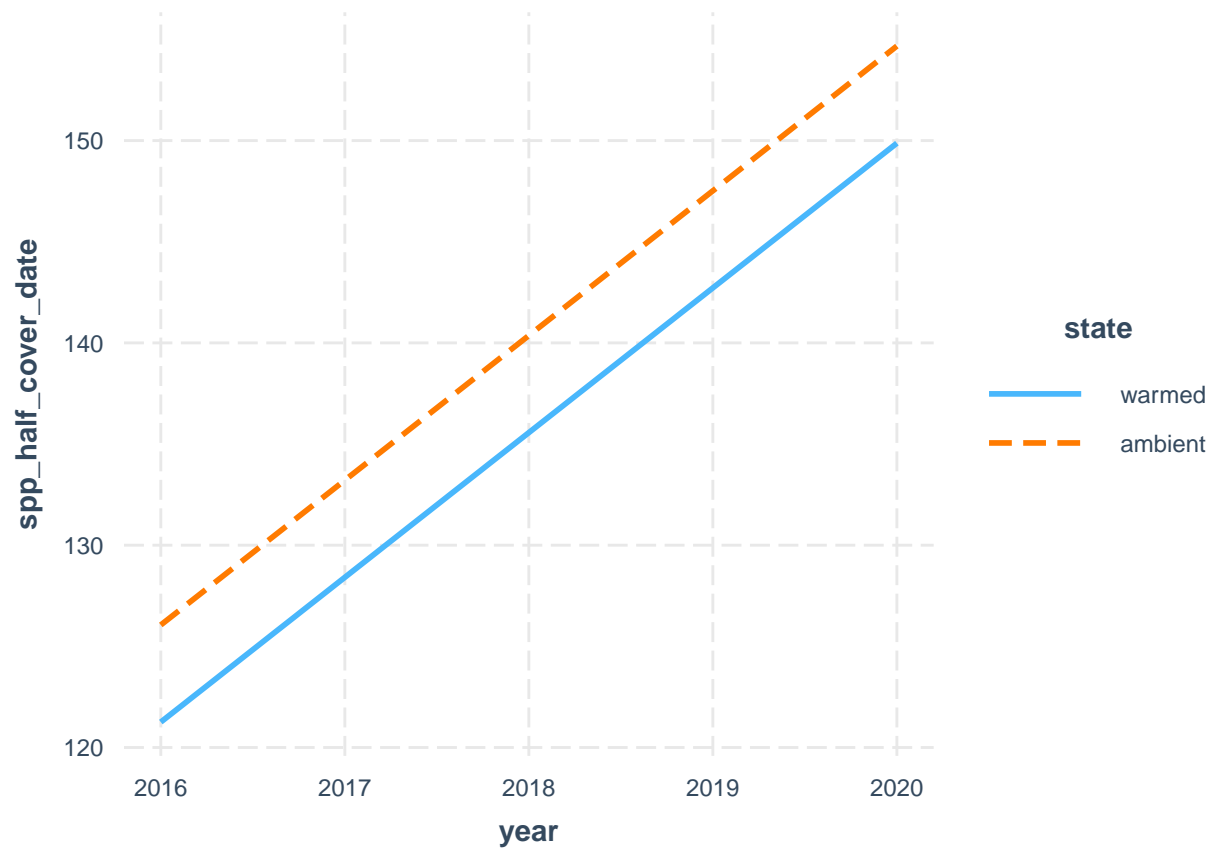


```
fit4 <- lm(plot_half_cover_date ~ state * year, data = green_kbsp)
interact_plot(fit4, pred = year, modx = state)
```

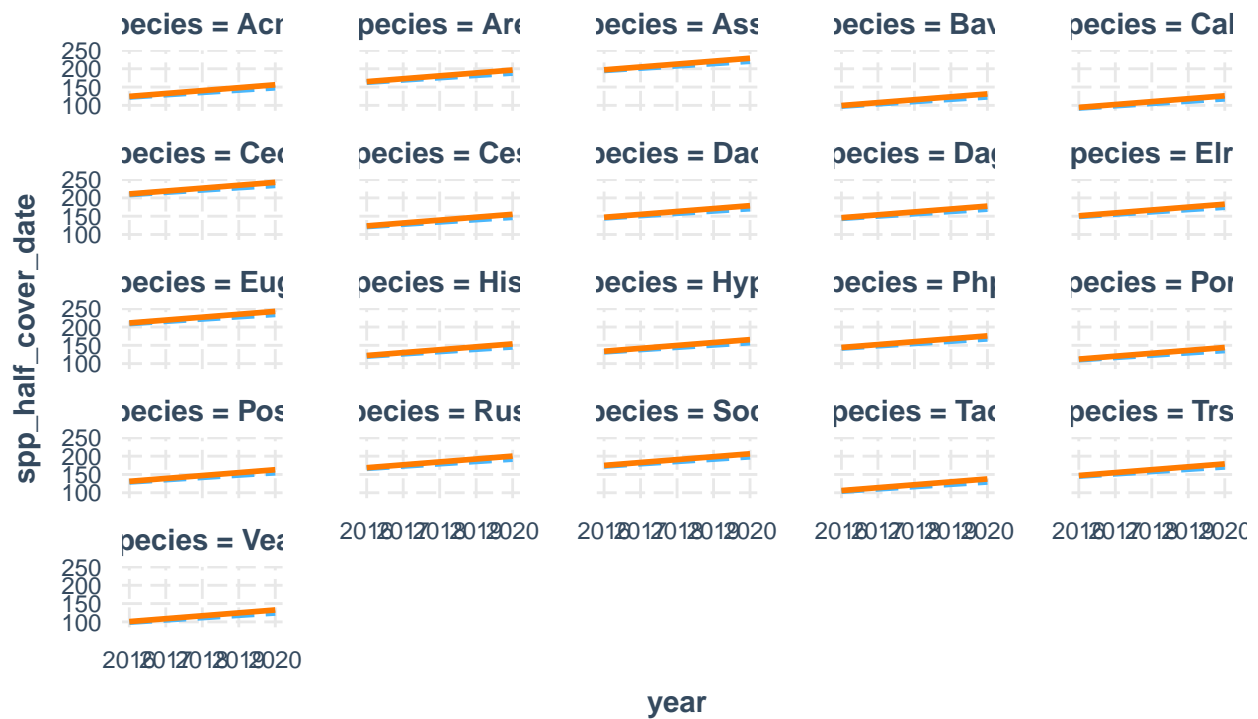
```
fit5 <- lm(spp_half_cover_date ~ state + year + species, data = green_kbs)
interact_plot(fit5, pred = year, modx = state)
```

```
## Warning: year and state are not included in an interaction with one another in the
## model.
```



```
fit6 <- lm(spp_half_cover_date ~ state * year + species, data = green_kbs)
interact_plot(fit6, pred = year, modx = state, mod2 = species)
```

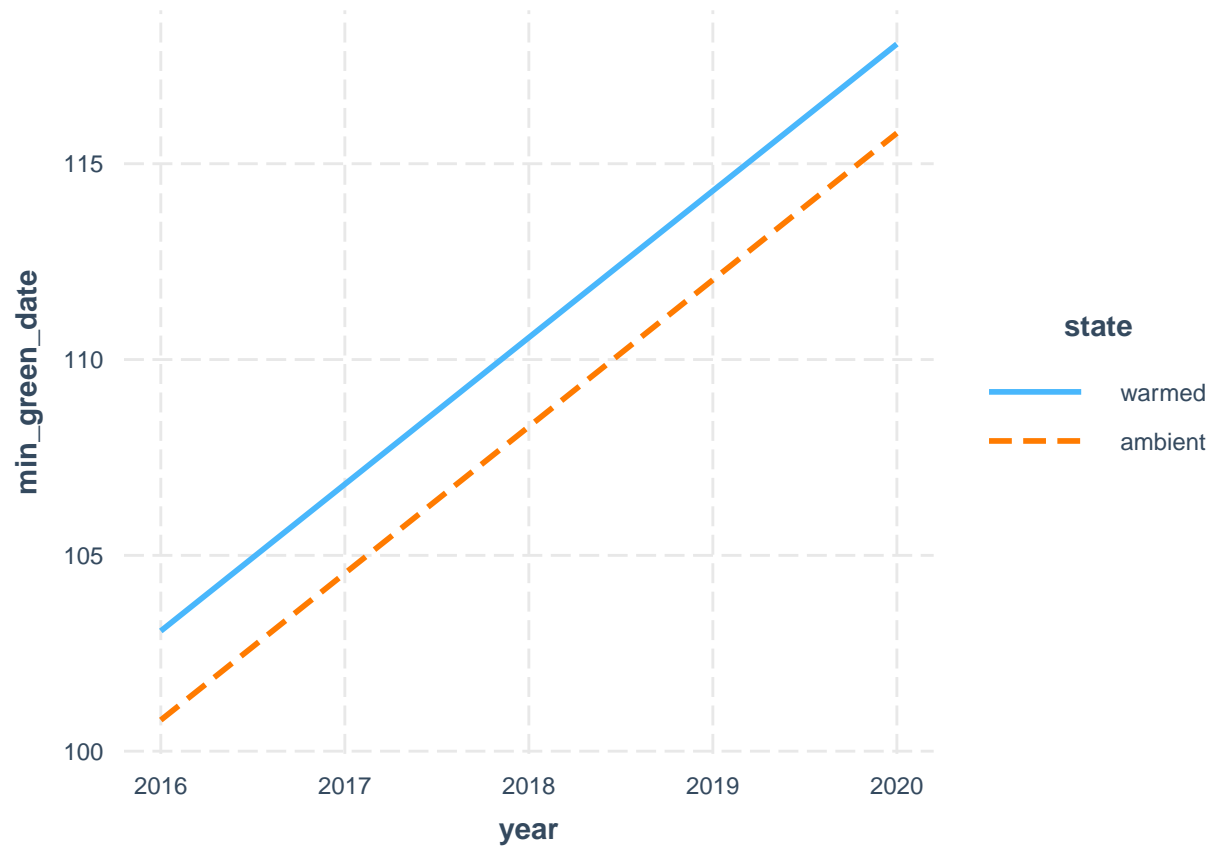
```
## Warning: year and state and species are not included in an interaction with one
## another in the model.
```



state — — — warmed ————— ambient

```
fit7 <- lm(min_green_date ~ state + year + species, data = green_kbs)
interact_plot(fit7, pred = year, modx = state)
```

```
## Warning: year and state are not included in an interaction with one another in the
## model.
```



```
fit8 <- lm(min_green_date ~ state * year + species, data = green_kbs)
interact_plot(fit8, pred = year, modx = state, mod2 = species)
```

```
## Warning: year and state and species are not included in an interaction with one
## another in the model.
```



Mixed Effects Models:

```
# Start by replicating (almost) what we did in the Decologia
# 2018 paper. The only difference here is that we have
# multiple years, so we are also including year as a fixed
# effect and as an interactive term. Our goal here is to find
# a model that is the best fit to the data. We also want to
# find a model that is the most parsimonious (one that has
# the fewest parameters).
```

```
# Do we need to include plot as a random effect with the KBS
# models?
mod1 <- lmer(spp_half_cover_date ~ state * year + insecticide *
  year + (1 | species) + (1 | plot), green_kbs, REML = FALSE)
```

```
## Warning: Some predictor variables are on very different scales: consider
## rescaling
```

```
## Warning: Some predictor variables are on very different scales: consider
## rescaling
```

```
mod2 <- lmer(spp_half_cover_date ~ state * year + insecticide *
  year + (1 | species), green_kbs, REML = FALSE)
```

```
## Warning: Some predictor variables are on very different scales: consider
## rescaling
```

```
## Warning: Some predictor variables are on very different scales: consider
## rescaling
```

```
# Run analysis of variance on each model (see this for more
# explanation on how anova on a linear mixed effects model is
# similar to an anova on a regular linear model:
# https://m-clark.github.io/docs/mixedModels/anovamixed.html)
anova(mod1)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##               Sum Sq Mean Sq NumDF   DenDF F value    Pr(>F)
## state           1272     1272      1 1235.9  0.5193    0.4713
## year          138894  138894      1 1241.4 56.6968 9.744e-14 ***
## insecticide     3587     3587      1 1233.9  1.4642    0.2265
## state:year       1276     1276      1 1235.9  0.5209    0.4706
## year:insecticide 3596     3596      1 1233.9  1.4681    0.2259
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(mod2)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##               Sum Sq Mean Sq NumDF   DenDF F value    Pr(>F)
## state           1312     1312      1 1247.2  0.5293    0.4670
## year          138522  138522      1 1252.3 55.8679 1.451e-13 ***
## insecticide     3950     3950      1 1247.7  1.5932    0.2071
## state:year       1316     1316      1 1247.2  0.5309    0.4664
## year:insecticide 3961     3961      1 1247.7  1.5977    0.2065
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Run an ANOVA to test if 2 models to test whether the more
# complex model is significantly better at capturing the data
# than the simpler model. If the resulting p-value is
# sufficiently low (usually less than 0.05), we conclude that
# the more complex model is significantly better than the
# simpler model, and thus favor the more complex model. If
# the p-value is not sufficiently low (usually greater than
# 0.05), we should favor the simpler model.
# https://bookdown.org/ndphillips/YaRrr/comparing-regression-models-with-anova.html
```

```
anova(mod2, mod1) # They are different so plot as a random effect should stay in the model (we go with
```

```
## Data: green_kbs
## Models:
## mod2: spp_half_cover_date ~ state * year + insecticide * year + (1 |
## mod2:   species)
## mod1: spp_half_cover_date ~ state * year + insecticide * year + (1 |
## mod1:   species) + (1 | plot)
##      npar   AIC   BIC logLik deviance Chisq Df Pr(>Chisq)
## mod2    8 13587 13628 -6785.6   13571
## mod1    9 13586 13632 -6784.0   13568 3.374  1    0.06623 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(mod1)
```

```
## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: spp_half_cover_date ~ state * year + insecticide * year + (1 |
```

```

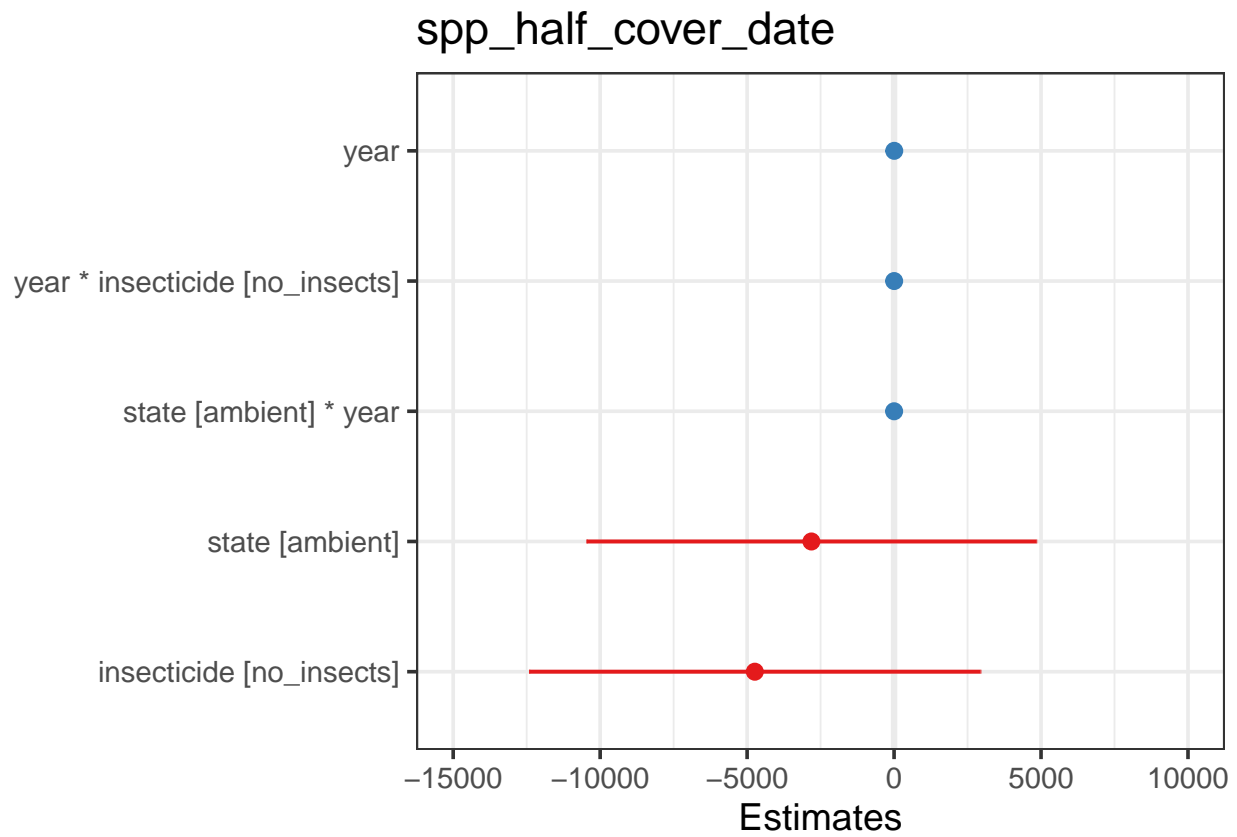
##      species) + (1 | plot)
##      Data: green_kbs
##
##      AIC      BIC    logLik deviance df.resid
## 13585.9 13632.2 -6784.0 13567.9      1259
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.1194 -0.7683 -0.2513  0.6857  3.2607
##
## Random effects:
##      Groups   Name      Variance Std.Dev.
##      plot     (Intercept) 32.77   5.725
##      species  (Intercept) 930.34 30.502
##      Residual                2449.77 49.495
## Number of obs: 1268, groups: plot, 24; species, 21
##
## Fixed effects:
##
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)    -11128.065   3367.723  1236.896  -3.304 0.000979
## stateambient     -2812.392   3902.785  1235.866  -0.721 0.471285
## year              5.589     1.669   1236.892   3.349 0.000836
## insecticideno_insects -4738.174  3915.775  1233.884  -1.210 0.226502
## stateambient:year    1.396     1.934   1235.875   0.722 0.470576
## year:insecticideno_insects 2.351     1.941   1233.896   1.212 0.225879
##
## (Intercept)          ***
## stateambient
## year                  ***
## insecticideno_insects
## stateambient:year
## year:insecticideno_insects
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) sttmbn year   insct_ sttmb:
## stateambint -0.583
## year        -1.000  0.583
## insctcdn_ns -0.519 -0.062  0.519
## statmbnt:yr  0.583 -1.000 -0.583  0.062
## yr:nsctcdn_  0.519  0.062 -0.519 -1.000 -0.062
## fit warnings:
## Some predictor variables are on very different scales: consider rescaling
summary(mod2)

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: spp_half_cover_date ~ state * year + insecticide * year + (1 |
##      species)
##      Data: green_kbs
##
##      AIC      BIC    logLik deviance df.resid
## 13587.3 13628.4 -6785.6 13571.3      1260

```

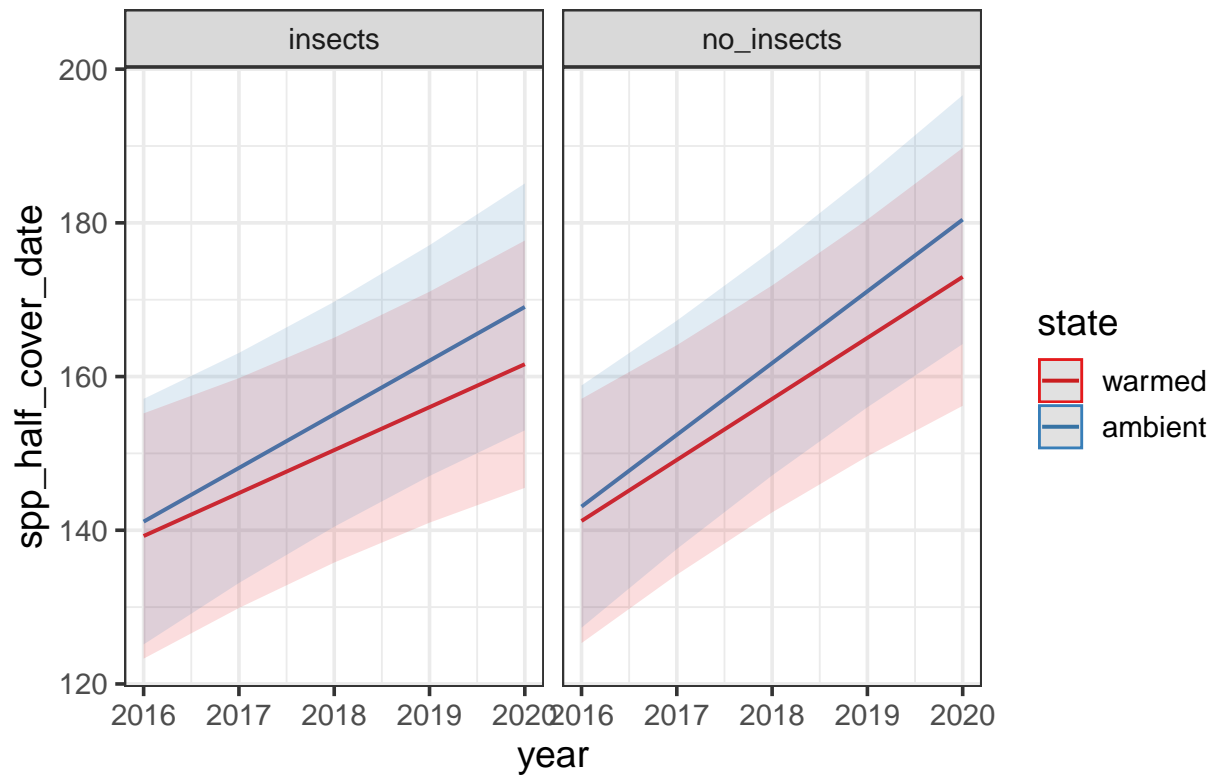
```
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.2600 -0.7665 -0.2441  0.6924  3.2567
##
## Random effects:
##   Groups   Name      Variance Std.Dev.
##  species (Intercept)  951.2    30.84
##  Residual            2479.5    49.79
## Number of obs: 1268, groups: species, 21
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   -10945.494   3381.228  1247.990  -3.237  0.00124 **
## stateambient    -2850.741   3918.330  1247.175  -0.728  0.46703
## year              5.498     1.676   1247.982   3.282  0.00106 **
## insecticideno_insects -4965.169  3933.683  1247.647  -1.262  0.20711
## stateambient:year    1.415     1.942   1247.177   0.729  0.46635
## year:insecticideno_insects  2.464     1.949   1247.650   1.264  0.20647
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) sttmbn year   insct_ sttmb:
## stateambint -0.583
## year        -1.000  0.583
## insctcdn_ns -0.520 -0.061  0.520
## statmbnt:yr  0.583 -1.000 -0.583  0.061
## yr:nsctcdn_  0.520  0.061 -0.520 -1.000 -0.061
## fit warnings:
## Some predictor variables are on very different scales: consider rescaling

# Next, plot the model. There are multiple variables but
# here's one way to do it based on this package sjPlot:
# https://strengjacke.github.io/sjPlot/articles/plot\_model\_estimates.html
# Annoyingly, this package somehow overwrites the factor
# order in its plotting so we will have to modify the code to
# get warmed = red. I haven't figured this out yet. It does
# seem to work on some of the plots. hmm.
`?`(plot_model)
# Plot the fixed effects estimates for different models these
# are the fixed effects estimates from summary(mod5)
plot_model(mod1, sort.est = TRUE)
```

```
# these are the fixed predicted values:  
plot_model(mod1, type = "pred", terms = c("year", "state", "insecticide"))
```

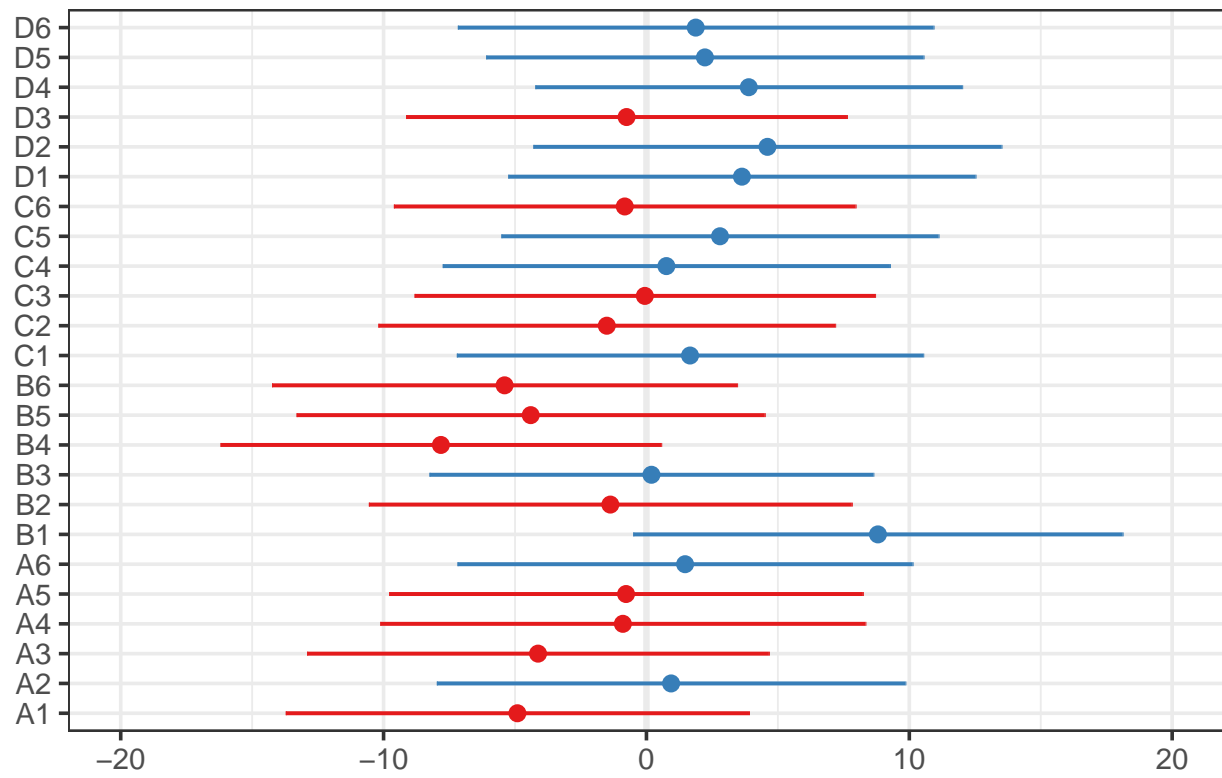
Predicted values of spp_half_cover_date



```
# these are the random effects estimates  
plot_model(mod1, type = "re", terms = c("species", "plot"))
```

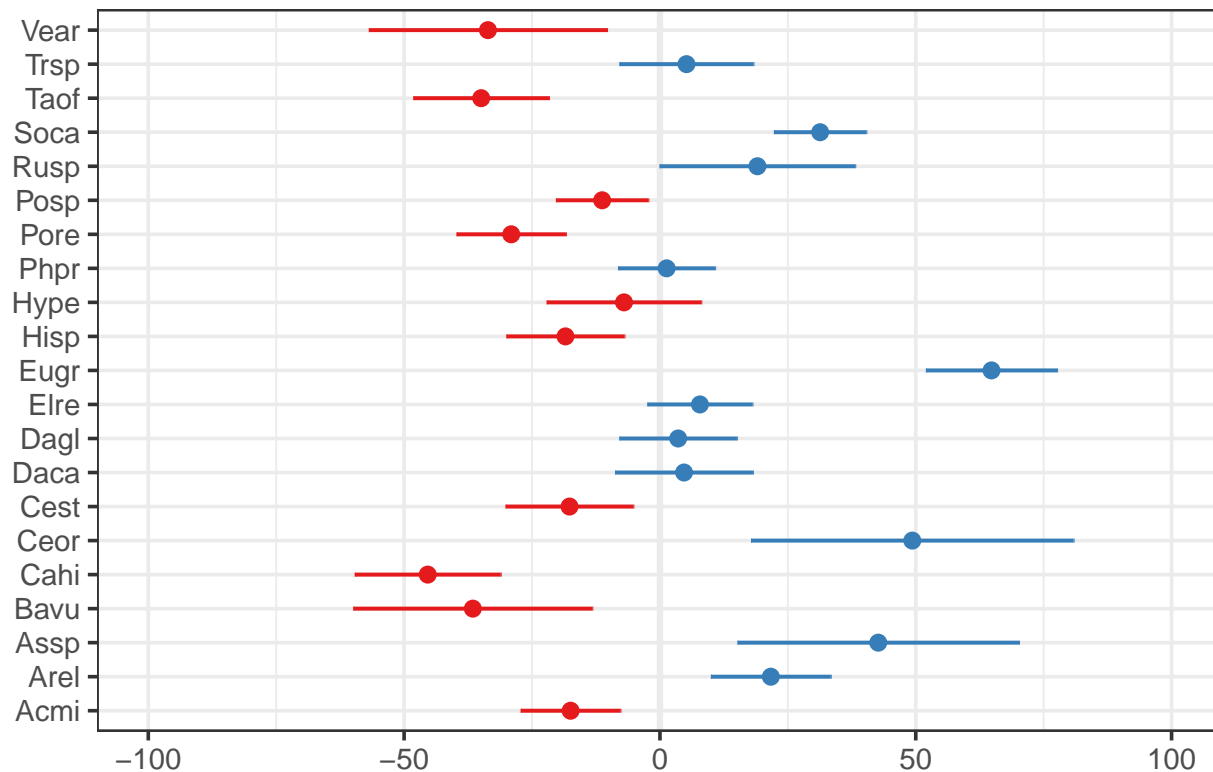
```
## [[1]]
```

Random effects



[[2]]

Random effects



Do we need to include insecticide?

```
mod3 <- lmer(spp_half_cover_date ~ state * year + (1 | species),
  green_kbs, REML = FALSE)
```

```
## Warning: Some predictor variables are on very different scales: consider
## rescaling
```

```
## Warning: Some predictor variables are on very different scales: consider
## rescaling
```

```
anova(mod1, mod3)
```

```
## Data: green_kbs
```

```
## Models:
```

```
## mod3: spp_half_cover_date ~ state * year + (1 | species)
```

```
## mod1: spp_half_cover_date ~ state * year + insecticide * year + (1 |
```

```
## mod1: species) + (1 | plot)
```

```
##      npar   AIC   BIC logLik deviance Chisq Df Pr(>Chisq)
```

```
## mod3    6 13591 13622 -6789.5   13579
```

```
## mod1    9 13586 13632 -6784.0   13568 10.994  3    0.01176 *
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AICctab(mod1, mod3, weights = T)
```

```
##      dAICc df weight
```

```
## mod1 0.0   9  0.921
```

```
## mod3 4.9   6  0.079
```

```

# Looks like yes  $P < 0.05$ , insecticide improves model fit so we
# will continue to include it and stick with mod1

# Does year need to be interactive with insecticide?
mod4 <- lmer(spp_half_cover_date ~ state * year + insecticide +
  (1 | species) + (1 | plot), green_kbs, REML = FALSE)

## Warning: Some predictor variables are on very different scales: consider
## rescaling

## Warning: Some predictor variables are on very different scales: consider
## rescaling

anova(mod1, mod4)

## Data: green_kbs
## Models:
## mod4: spp_half_cover_date ~ state * year + insecticide + (1 | species) +
## mod4: (1 | plot)
## mod1: spp_half_cover_date ~ state * year + insecticide * year + (1 |
## mod1: species) + (1 | plot)
##      npar   AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## mod4     8 13585 13626 -6784.7    13569
## mod1     9 13586 13632 -6784.0    13568 1.4664  1      0.2259

# No,  $P > 0.05$  so insecticide*year doesn't strongly improve
# model fit so we will shift to mod4
anova(mod3, mod4)

## Data: green_kbs
## Models:
## mod3: spp_half_cover_date ~ state * year + (1 | species)
## mod4: spp_half_cover_date ~ state * year + insecticide + (1 | species) +
## mod4: (1 | plot)
##      npar   AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## mod3     6 13591 13622 -6789.5    13579
## mod4     8 13585 13626 -6784.7    13569 9.5277  2    0.008533 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# Yes,  $P < 0.05$  so insecticide still improves model fit so we
# will stay with mod4

# Does year need to be interactive with state?
mod5 <- lmer(spp_half_cover_date ~ state + year + insecticide +
  (1 | species) + (1 | plot), green_kbs, REML = FALSE)
anova(mod4, mod5)

## Data: green_kbs
## Models:
## mod5: spp_half_cover_date ~ state + year + insecticide + (1 | species) +
## mod5: (1 | plot)
## mod4: spp_half_cover_date ~ state * year + insecticide + (1 | species) +
## mod4: (1 | plot)
##      npar   AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## mod5     7 13584 13620 -6785.0    13570

```

```
## mod4      8 13585 13626 -6784.7    13569 0.6369 1      0.4249
AICctab(mod4, mod5, weights = T)

##      dAICc df weight
## mod5 0.0   7  0.67
## mod4 1.4   8  0.33

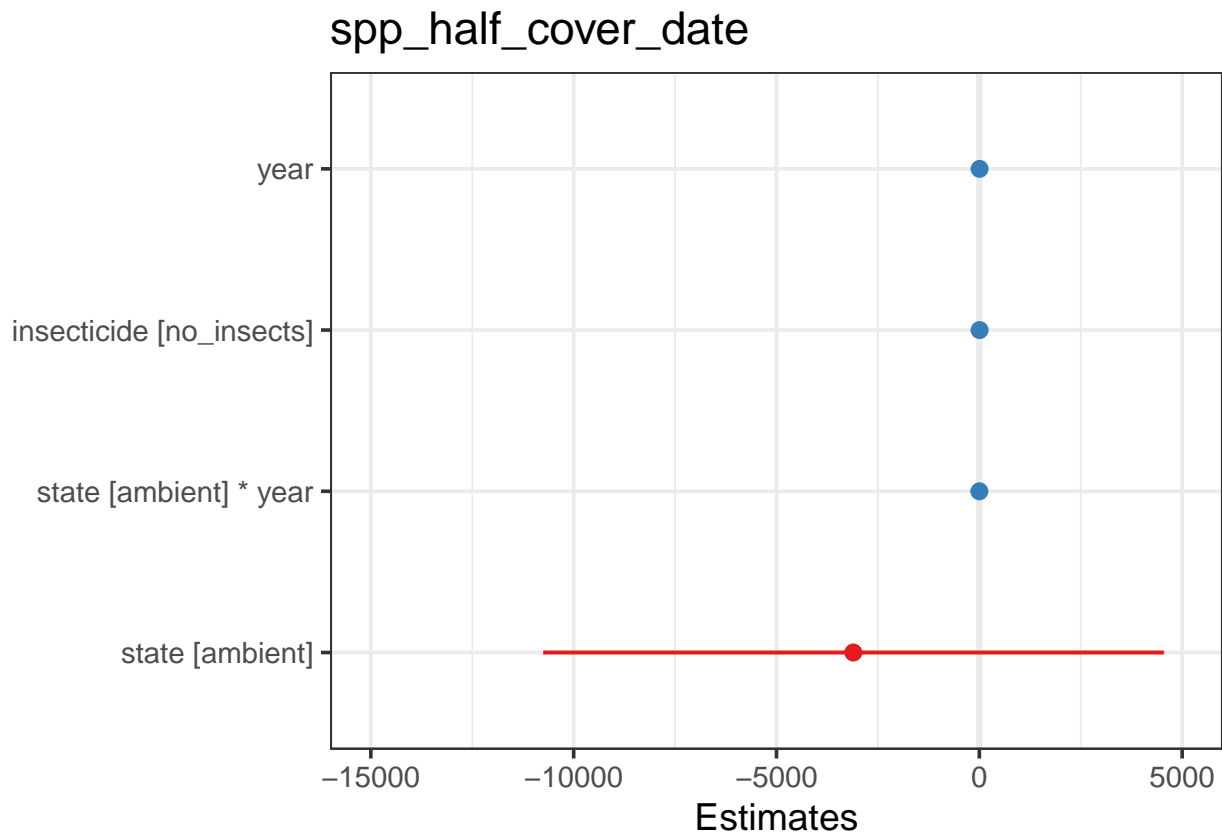
# No, P>0.05 so state*year doesn't improve model fit so we
# could drop it and go with mod5, but note that the AIC
# values are super close. mod4 makes sense, with increased
# divergence between warmed and ambient.
summary(mod4)

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: spp_half_cover_date ~ state * year + insecticide + (1 | species) +
## (1 | plot)
## Data: green_kbs
##
##      AIC      BIC   logLik deviance df.resid
## 13585.4 13626.5 -6784.7 13569.4      1260
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.1148 -0.7659 -0.2457  0.6789  3.2173
##
## Random effects:
## Groups   Name                Variance Std.Dev.
## plot     (Intercept)         33.48    5.786
## species  (Intercept)      928.65   30.474
## Residual                    2452.34  49.521
## Number of obs: 1268, groups: plot, 24; species, 21
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)   -13246.986    2880.397 1242.858  -4.599 4.68e-06 ***
## stateambient   -3106.128    3897.363 1235.902  -0.797  0.4256
## year              6.639      1.427 1242.865   4.651 3.66e-06 ***
## insecticideno_insects  6.358      3.690   21.051   1.723  0.0995 .
## stateambient:year    1.542      1.931 1235.911   0.798  0.4249
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) sttmbn year  insct_
## stateambint -0.721
## year        -1.000  0.721
## insctcdn_ns -0.062  0.029  0.061
## statmbnt:yr  0.721 -1.000 -0.721 -0.029
## fit warnings:
## Some predictor variables are on very different scales: consider rescaling
anova(mod4)

## Type III Analysis of Variance Table with Satterthwaite's method
```

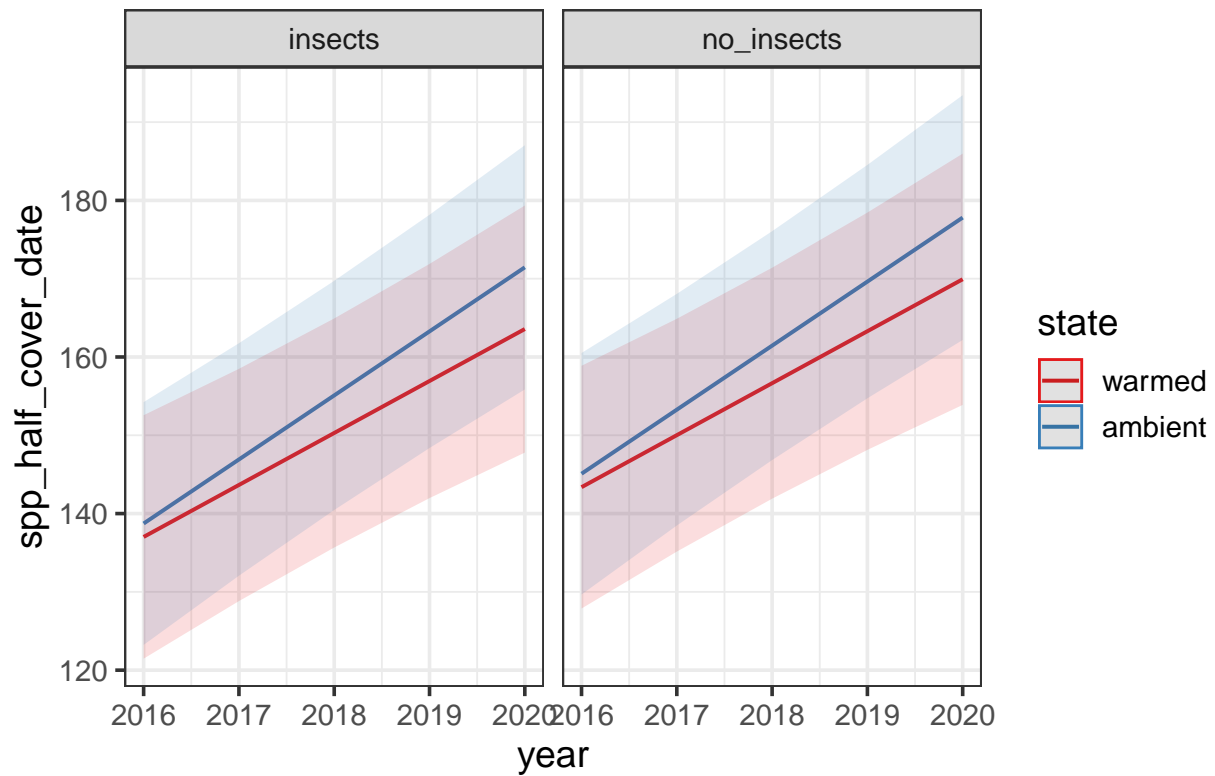
```
##           Sum Sq Mean Sq NumDF   DenDF F value    Pr(>F)
## state           1558      1558     1 1235.90  0.6352  0.42561
## year          137200    137200     1 1241.48 55.9464 1.404e-13 ***
## insecticide      7282       7282     1   21.05  2.9693  0.09952 .
## state:year       1562      1562     1 1235.91  0.6371  0.42493
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# these are the fixed effects estimates from summary(mod4)
plot_model(mod4, sort.est = TRUE)
```



```
# these are the fixed predicted values:
plot_model(mod4, type = "pred", terms = c("year", "state", "insecticide"))
```

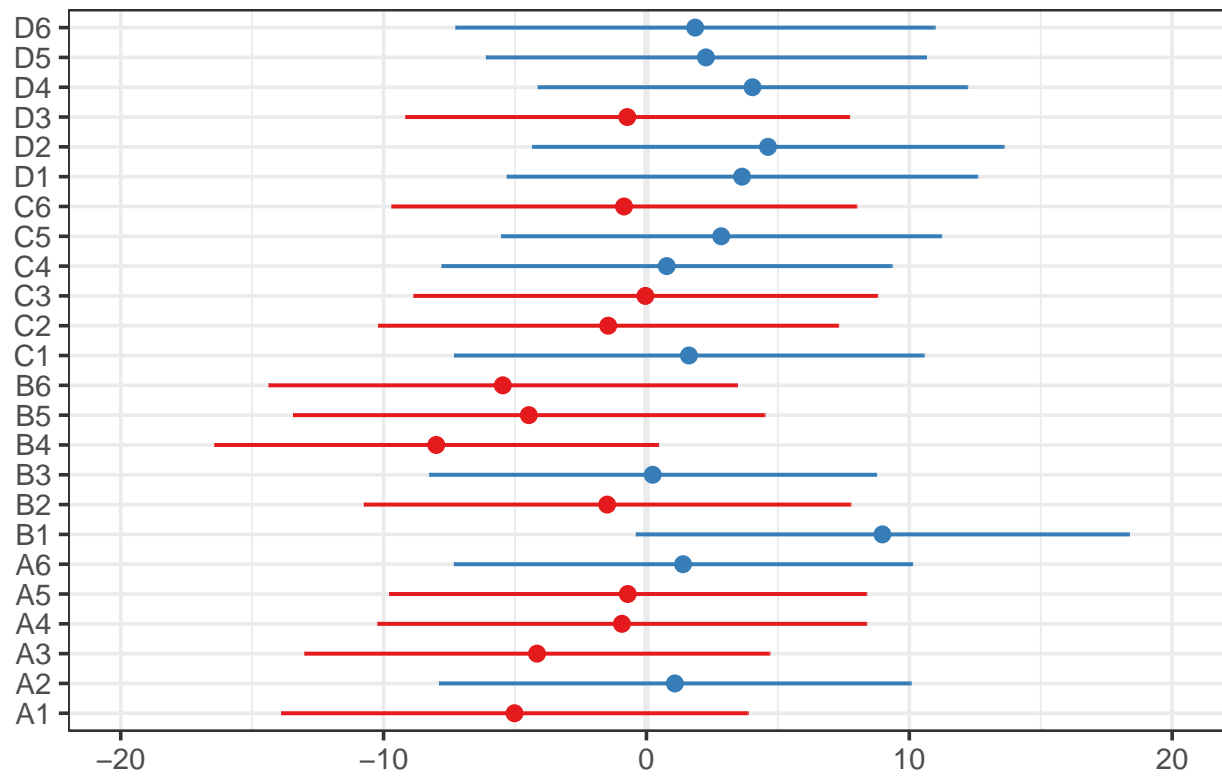
Predicted values of spp_half_cover_date



```
# these are the random effects estimates
plot_model(mod4, type = "re", terms = c("species", "plot"))
```

```
## [[1]]
```

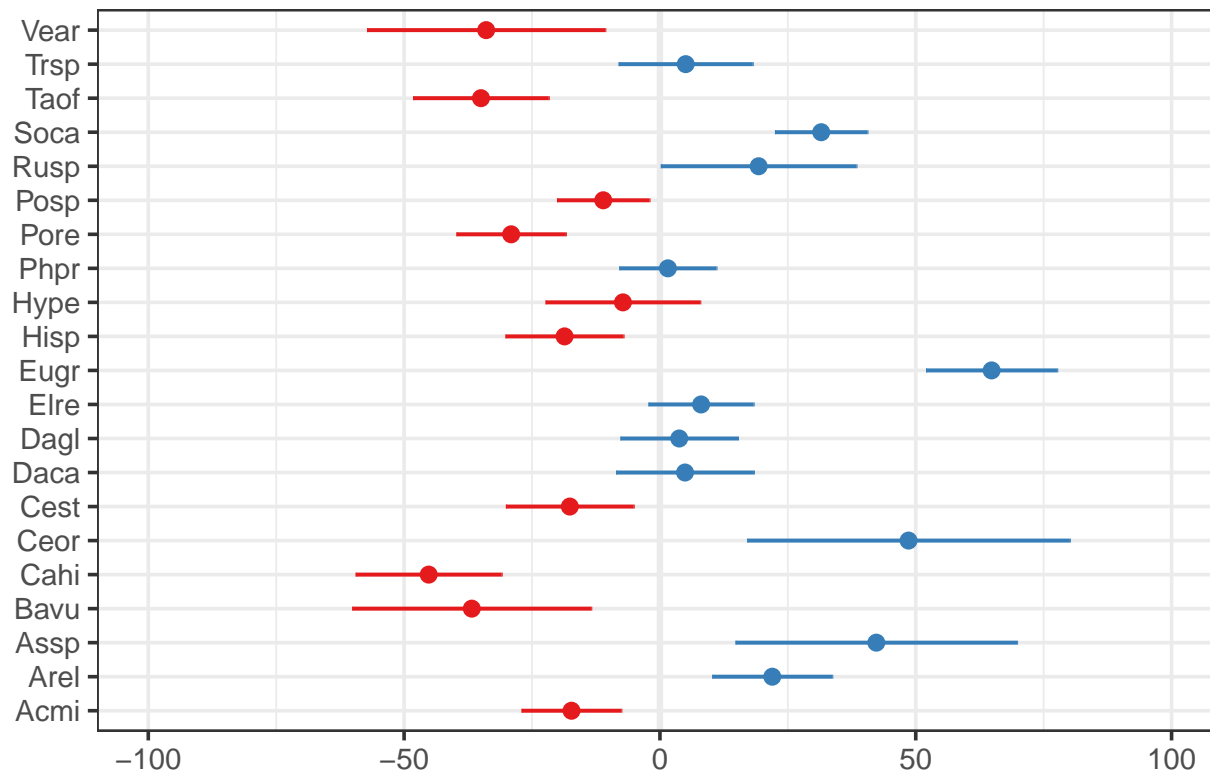

Random effects



##

[[2]]

Random effects



```
# If we wanted to include plots nested within year it would
# look like this:
mod7 <- lmer(spp_half_cover_date ~ state * year + insecticide +
  (1 | species) + (1 + year | plot), green_kbs, REML = FALSE)
```

```
## Warning: Some predictor variables are on very different scales: consider
## rescaling
```

```
## boundary (singular) fit: see ?isSingular
```

```
## Warning: Some predictor variables are on very different scales: consider
## rescaling
```

```
## Warning: Model failed to converge with 1 negative eigenvalue: -3.6e-01
```

```
anova(mod4, mod7)
```

```
## Data: green_kbs
```

```
## Models:
```

```
## mod4: spp_half_cover_date ~ state * year + insecticide + (1 | species) +
```

```
## mod4:      (1 | plot)
```

```
## mod7: spp_half_cover_date ~ state * year + insecticide + (1 | species) +
```

```
## mod7:      (1 + year | plot)
```

```
##      npar   AIC   BIC logLik deviance Chisq Df Pr(>Chisq)
```

```
## mod4      8 13585 13626 -6784.7   13569
```

```
## mod7     10 13596 13647 -6787.8   13576    0  2          1
```

```
anova(mod7)
```

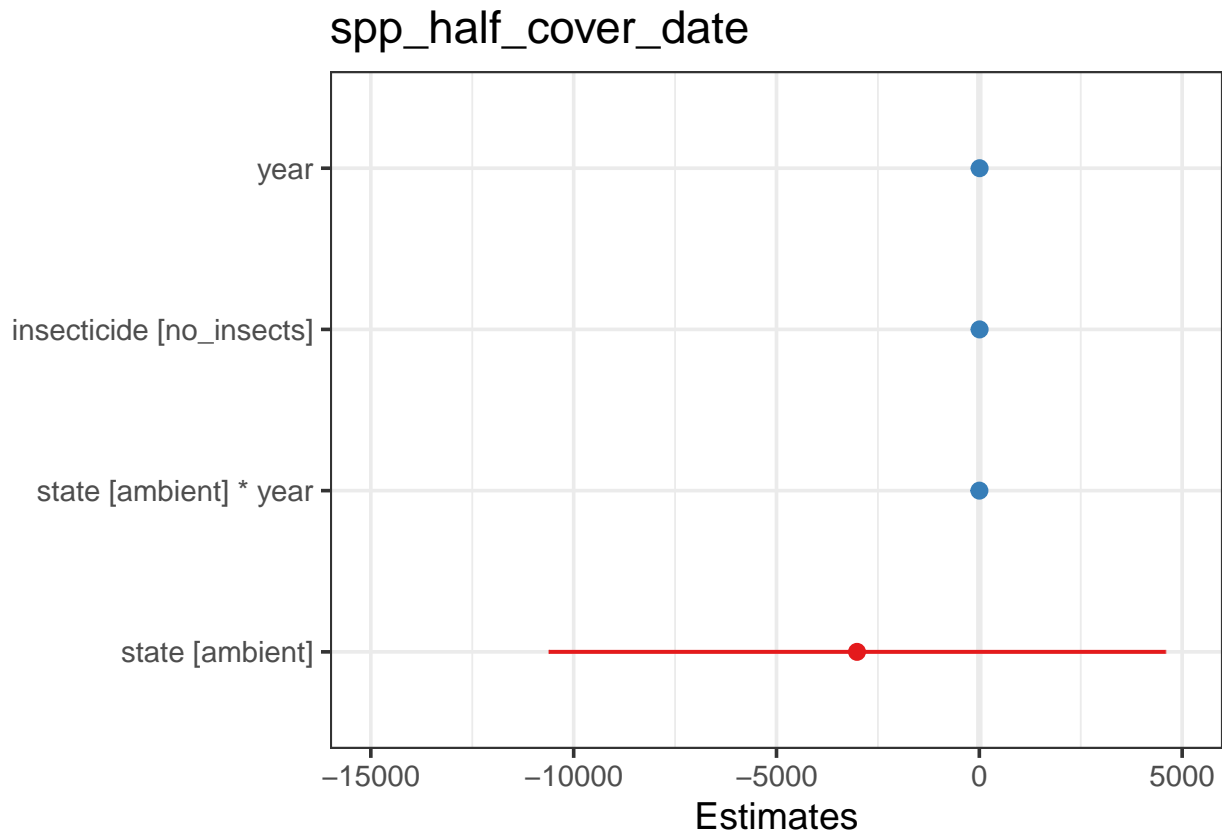
```
## Type III Analysis of Variance Table with Satterthwaite's method
```

```
##      Sum Sq Mean Sq NumDF   DenDF F value    Pr(>F)
```

```
## state      1467      1467      1 1232.41  0.6049   0.4369
## year      134924   134924      1 1238.24 55.6476 1.626e-13 ***
## insecticide 6552      6552      1   19.58  2.7021   0.1162
## state:year  1471      1471      1 1232.41  0.6068   0.4362
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Yup, seems to matter but it is making this more complex,
# though not overly so because it's on the random effects
# structure only.
```

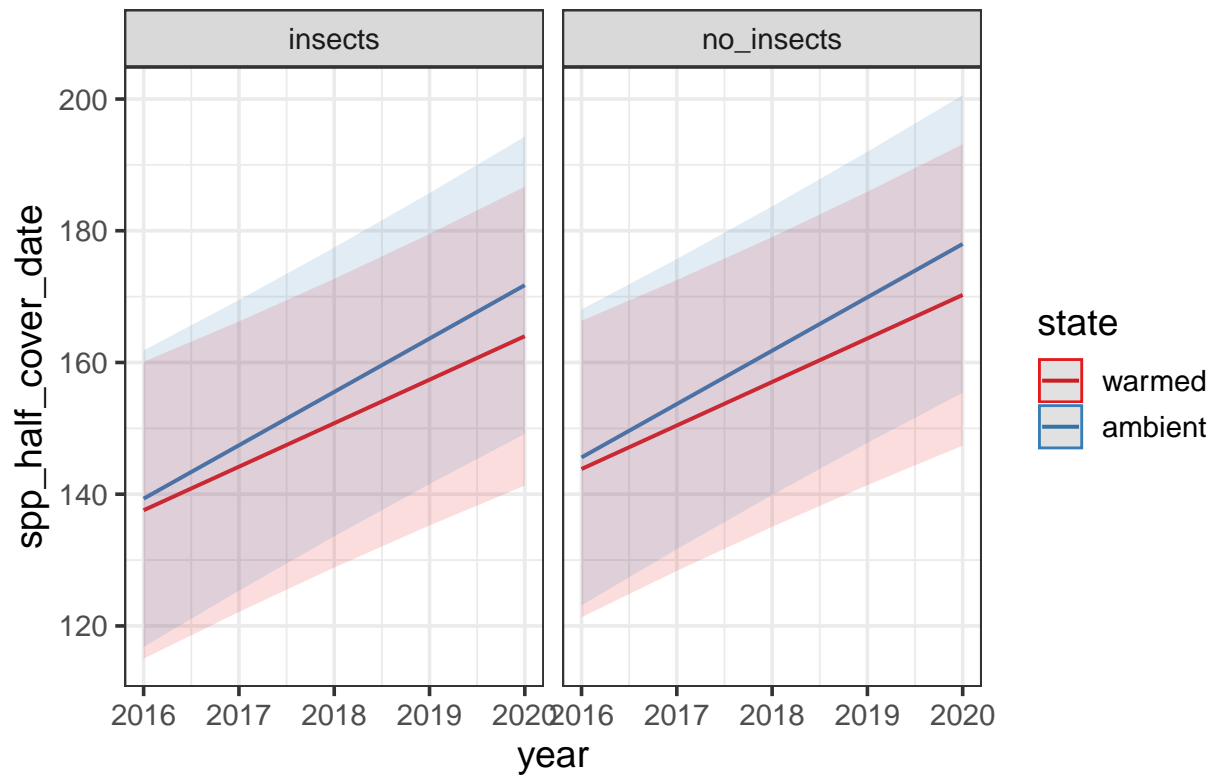
```
plot_model(mod7, sort.est = TRUE)
```



```
# these are the fixed predicted values:
```

```
plot_model(mod7, type = "pred", terms = c("year", "state", "insecticide"))
```

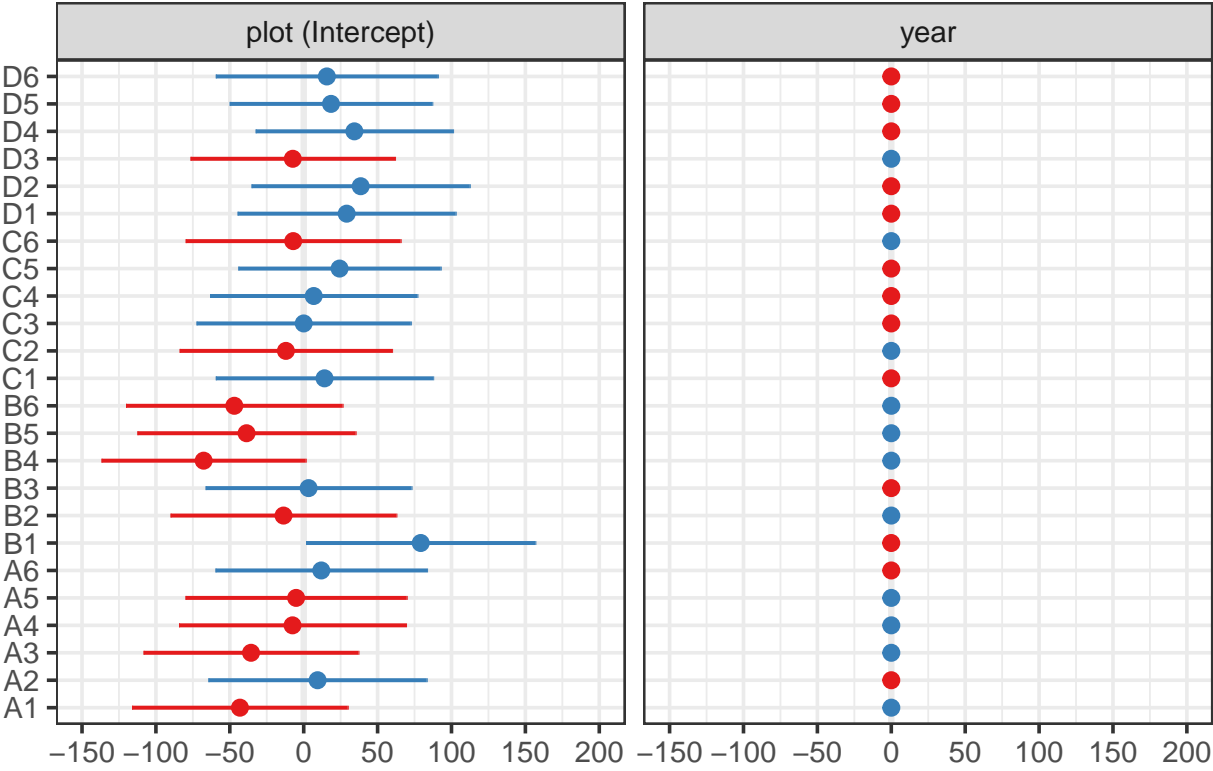
Predicted values of spp_half_cover_date



```
# these are the random effects estimates
plot_model(mod7, type = "re", terms = c("species", "plot"))
```

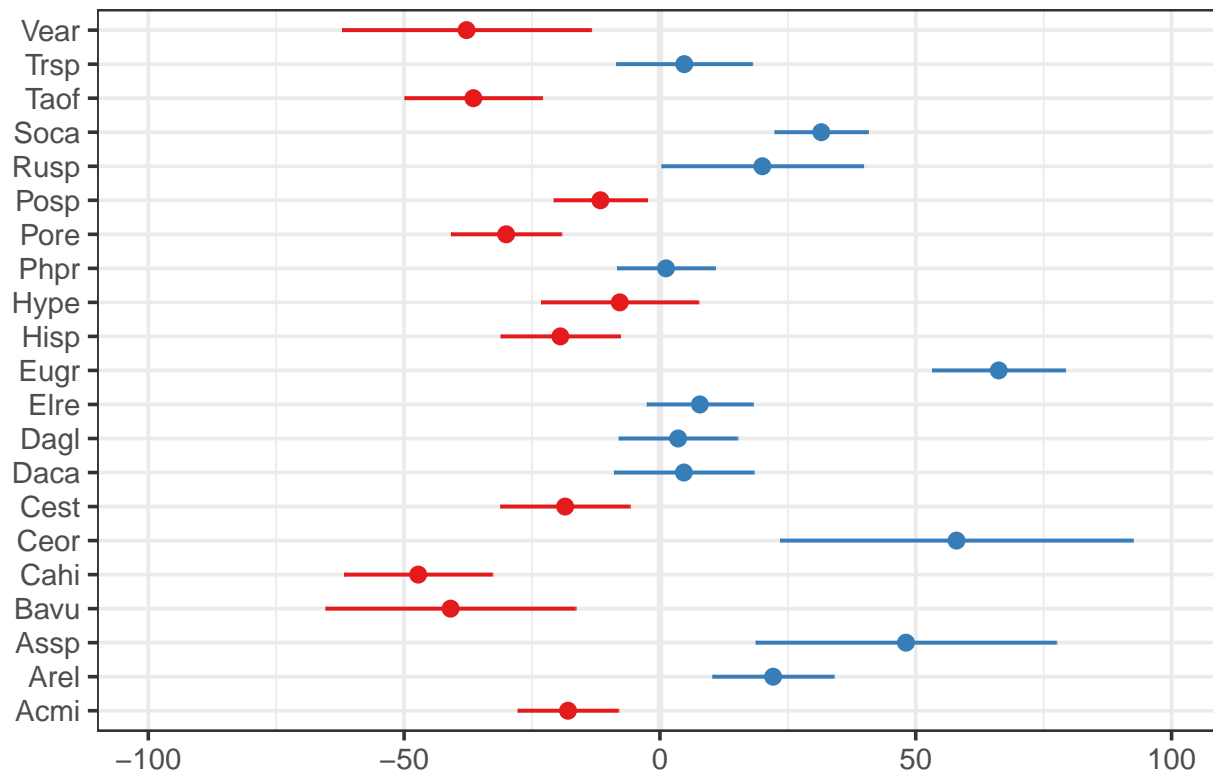
```
## [[1]]
```

Random effects



[[2]]

Random effects



mod4 (and mod7) are pretty complex in terms of interpretation (they actually don't have many parameters though). We could consider an alternative model that's simpler to understand and also one that provides more insight about the species. That would be something like this:

```
mod8 <- lmer(spp_half_cover_date ~ state + species + (1 + year | plot), green_kbs, REML = FALSE)
```

boundary (singular) fit: see ?isSingular

Warning: Model failed to converge with 1 negative eigenvalue: -3.2e-01

```
anova(mod7, mod8) # model 8 is a better fit to data
```

Data: green_kbs

Models:

mod7: spp_half_cover_date ~ state * year + insecticide + (1 | species) +

mod7: (1 + year | plot)

mod8: spp_half_cover_date ~ state + species + (1 + year | plot)

	npar	AIC	BIC	logLik	deviance	Chisq	Df	Pr(>Chisq)
## mod7	10	13596	13647	-6787.8	13576			
## mod8	26	13595	13729	-6771.6	13543	32.465	16	0.008694 **

mod7 10 13596 13647 -6787.8 13576

mod8 26 13595 13729 -6771.6 13543 32.465 16 0.008694 **

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
summary(mod8)
```

Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's

method [lmerModLmerTest]

```

## Formula: spp_half_cover_date ~ state + species + (1 + year | plot)
## Data: green_kbs
##
##      AIC      BIC   logLik deviance df.resid
## 13595.1 13728.9 -6771.6 13543.1    1242
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.1232 -0.7237 -0.2572  0.7334  3.4828
##
## Random effects:
## Groups   Name      Variance Std.Dev. Corr
## plot     (Intercept) 2.537e+03 50.36370
##          year        4.741e-04 0.02177 -1.00
## Residual          2.518e+03 50.17603
## Number of obs: 1268, groups: plot, 24
##
## Fixed effects:
##              Estimate Std. Error      df t value Pr(>|t|)
## (Intercept) 133.901      5.546   302.106  24.143 < 2e-16 ***
## stateambient  5.412      3.897    21.438   1.389 0.179229
## speciesArel  44.169      7.844   1260.160   5.631 2.21e-08 ***
## speciesAssp  70.545     16.757   1267.942   4.210 2.73e-05 ***
## speciesBavu -22.949     13.996   1267.915  -1.640 0.101318
## speciesCahi -35.193      9.011   1257.335  -3.906 9.90e-05 ***
## speciesCeor  84.001     19.822   1261.449   4.238 2.42e-05 ***
## speciesCest  -5.209      8.183   1253.784  -0.637 0.524527
## speciesDaca  19.856      8.627   1256.246   2.302 0.021515 *
## speciesDagl  24.693      7.743   1258.750   3.189 0.001462 **
## speciesElre  30.733      7.252   1252.267   4.238 2.42e-05 ***
## speciesEugr  88.527      8.335   1264.020  10.621 < 2e-16 ***
## speciesHisp  -1.916      7.762   1250.982  -0.247 0.805081
## speciesHype  12.271      9.416   1266.314   1.303 0.192754
## speciesPhpr  20.661      6.936   1250.109   2.979 0.002948 **
## speciesPore -11.775      7.410   1256.420  -1.589 0.112266
## speciesPosp   7.985      6.762   1247.878   1.181 0.237868
## speciesRusp  43.145     11.525   1262.439   3.744 0.000189 ***
## speciesSoca  51.527      6.762   1247.878   7.620 4.99e-14 ***
## speciesTaof -19.080      8.535   1265.042  -2.236 0.025558 *
## speciesTrsp  20.983      8.462   1254.243   2.480 0.013279 *
## speciesVear -27.931     13.943   1264.466  -2.003 0.045361 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation matrix not shown by default, as p = 22 > 12.
## Use print(x, correlation=TRUE) or
##      vcov(x)          if you need it
##
## convergence code: 0
## boundary (singular) fit: see ?isSingular
anova(mod8) # investigates whether at least one of the levels within each factor is significantly diff
## Type III Analysis of Variance Table with Satterthwaite's method
##      Sum Sq Mean Sq NumDF   DenDF F value Pr(>F)

```

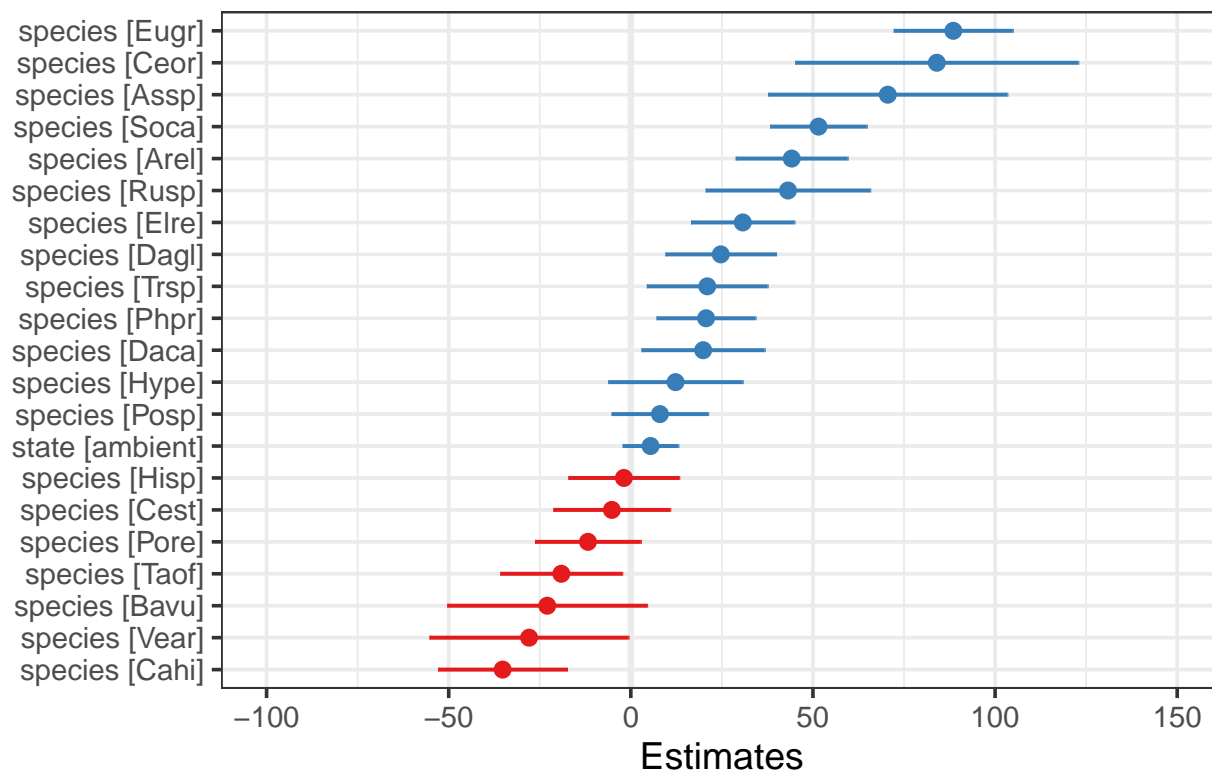
```
## state      4854      4854      1  21.44  1.9281 0.1792
## species 989680   49484     20 1260.49 19.6550 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

*# Yes, at least one of the species is different (they do not
all have the same half cover dates).*

*# Take a look at the estimates for each fixed effect. These
are the estimates from summary(mod8). You'll see that
species vary a lot - and many of them are different from
zero (meaning their half cover date is significantly
different from zero).*

```
plot_model(mod8, sort.est = TRUE)
```

spp_half_cover_date

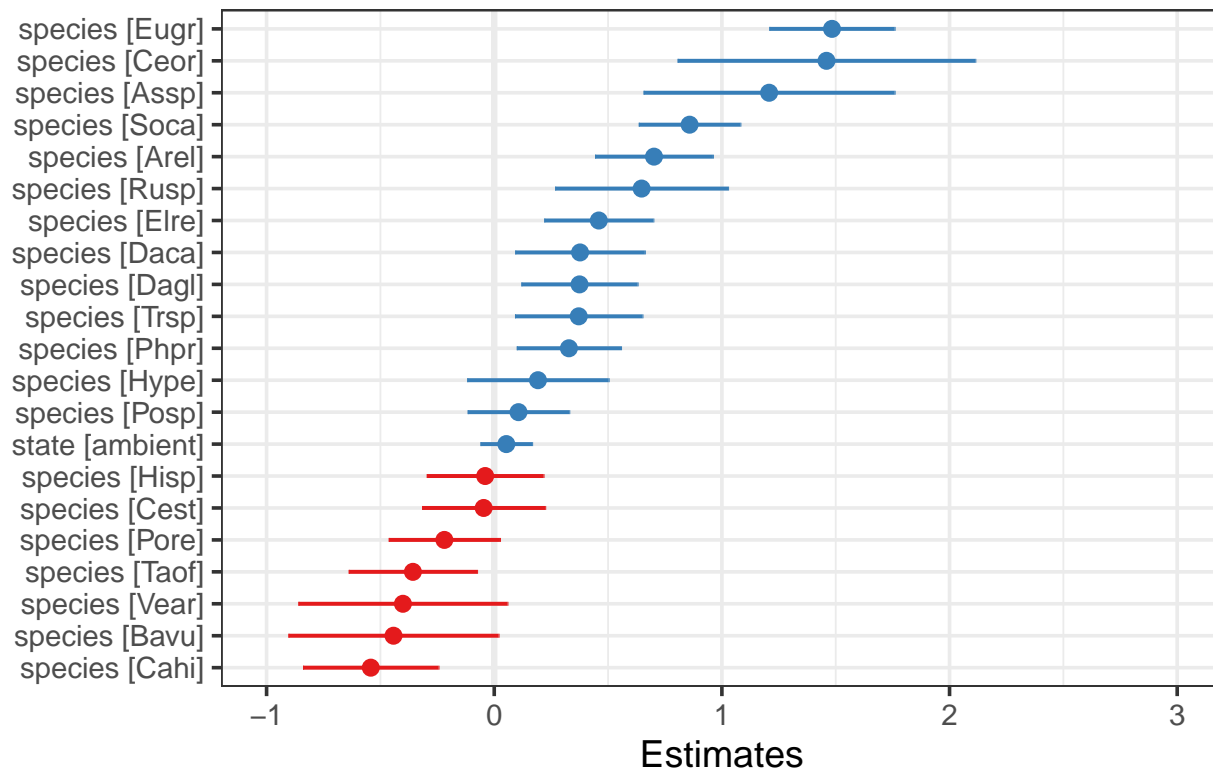


if you want to standardize the estimates:

```
plot_model(mod8, sort.est = TRUE, type = "std")
```

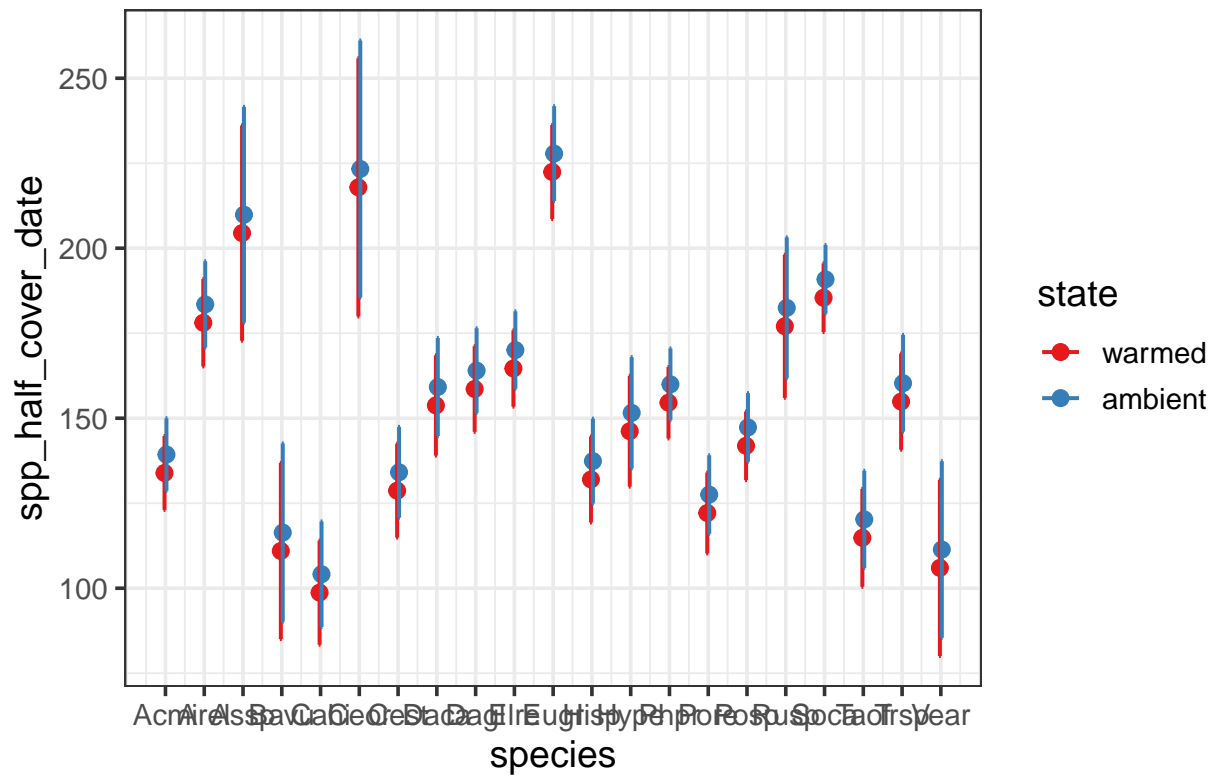
```
## boundary (singular) fit: see ?isSingular
```


spp_half_cover_date



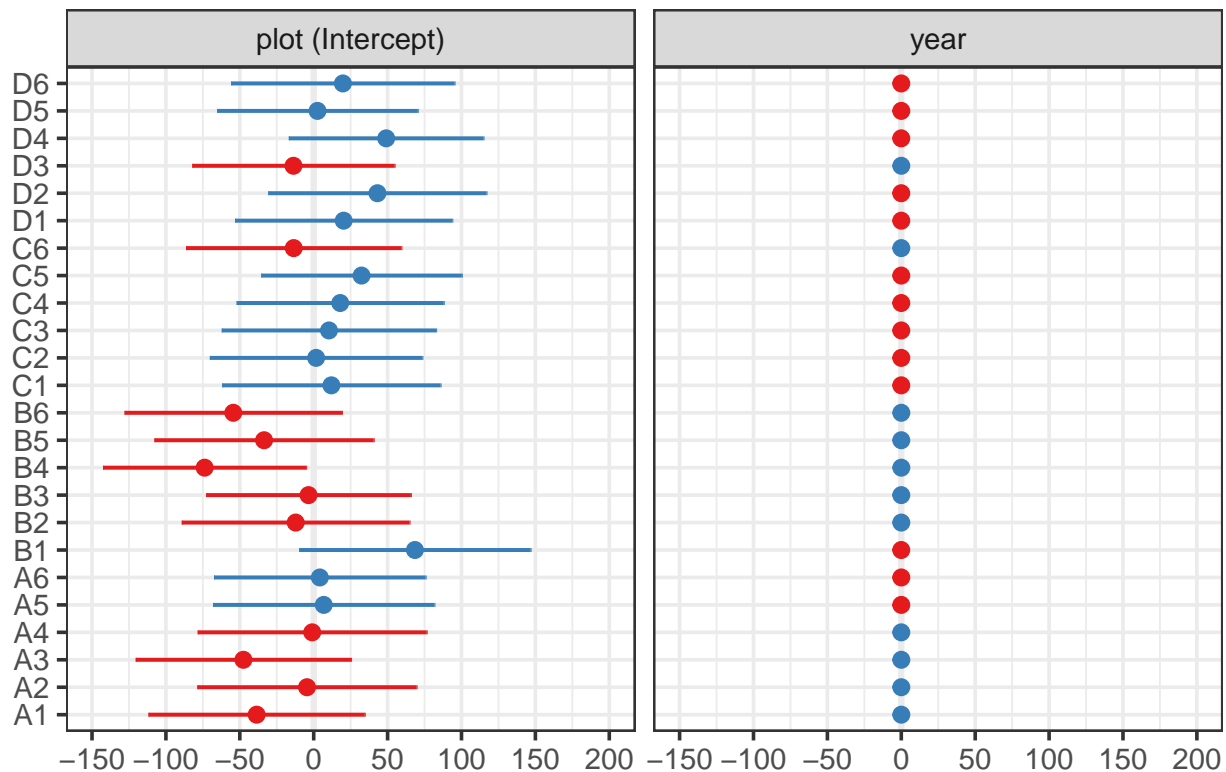
```
# these are the fixed predicted values: - note this is a new
# plot
plot_model(mod8, type = "pred", terms = c("species", "state"))
```

Predicted values of spp_half_cover_date



```
# these are the random effects estimates
plot_model(mod8, type = "re")
```

Random effects



```
# You could now run some post hoc tests on these (see:
# https://stats.stackexchange.com/questions/169543/output-of-fixed-effects-summary-in-lmertest-in-r-and

# Here are some other options for plotting these plots above:
# https://stackoverflow.com/questions/31075407/plot-mixed-effects-model-in-ggplot

# Here's another approach:
# https://stats.stackexchange.com/questions/98958/plots-to-illustrate-results-of-linear-mixed-effect-mo
# Not quite working yet:
newdat <- expand.grid(state = unique(green_kbs$state), year = c(min(green_kbs$year),
  max(green_kbs$year)), insecticide = unique(green_kbs$insecticide))
# p <- ggplot(green_kbs, aes(x=year, y=spp_half_cover_date,
# colour=state, shape=insecticide)) + geom_point(size=3) +
# geom_line(aes(y=predict(mod5), group=species,
# size='species')) + geom_line(data=newdat,
# aes(y=predict(mod5, level=0, newdata=newdat),
# size='Population')) + scale_size_manual(name='Predictions',
# values=c('species'=0.5, 'Population'=3)) +
# #facet_wrap(~insecticide) + theme_bw(base_size=22) print(p)

# New version of our model incorporating interaction term and
# species within year so that there is a separate intercept
# and slope for each species. The issue here is that there
# are some species that are not found each year. Easiest to
# remove those from another version of this dataframe before
# running below. Otherwise, it's not a balanced design.
# updated mod4
mod9 <- lmer(spp_half_cover_date ~ state * year + (1 + year |
```

```

species), green_kbs)

## Warning: Some predictor variables are on very different scales: consider
## rescaling

## boundary (singular) fit: see ?isSingular

## Warning: Some predictor variables are on very different scales: consider
## rescaling

# So another version of this model would include the
# interaction but not include the nesting (and thus would
# assume that species aren't observed ea yr) updated mod5
mod10 <- lmer(spp_half_cover_date ~ state * year + (1 | species),
  green_kbs)

## Warning: Some predictor variables are on very different scales: consider
## rescaling

## Warning: Some predictor variables are on very different scales: consider
## rescaling

summary(mod5)

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: spp_half_cover_date ~ state + year + insecticide + (1 | species) +
## (1 | plot)
## Data: green_kbs
##
##      AIC      BIC   logLik deviance df.resid
##  13584   13620   -6785   13570     1261
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.1127 -0.7727 -0.2469  0.6665  3.1808
##
## Random effects:
## Groups   Name      Variance Std.Dev.
## plot     (Intercept)  33.61    5.798
## species  (Intercept) 931.11   30.514
## Residual                2453.43  49.532
## Number of obs: 1268, groups: plot, 24; species, 21
##
## Fixed effects:
##              Estimate Std. Error    df t value Pr(>|t|)
## (Intercept)   -1.491e+04  1.995e+03 1.241e+03  -7.470 1.51e-13 ***
## stateambient    4.622e+00  3.701e+00 2.128e+01   1.249  0.2253
## year           7.461e+00  9.888e-01 1.241e+03   7.545 8.70e-14 ***
## insecticideno_insects 6.441e+00  3.692e+00 2.101e+01   1.745  0.0956 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) sttmbn year
## stateambint  0.025

```

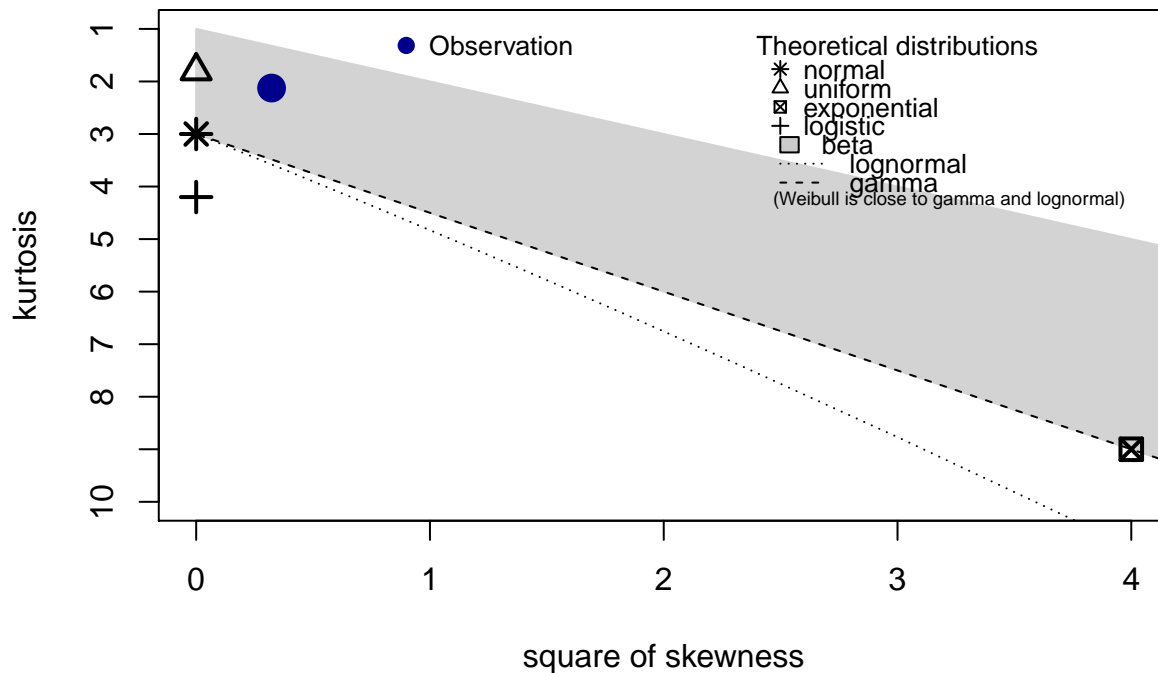
```
## year          -1.000 -0.026
## insctcdn_ns -0.060 -0.038  0.059
```

ORIGINAL CODE BELOW; not edited by Phoebe

Seeing what other distribution could fit

```
descdist(green_kbs$spp_half_cover_date, discrete = FALSE)
```

Cullen and Frey graph

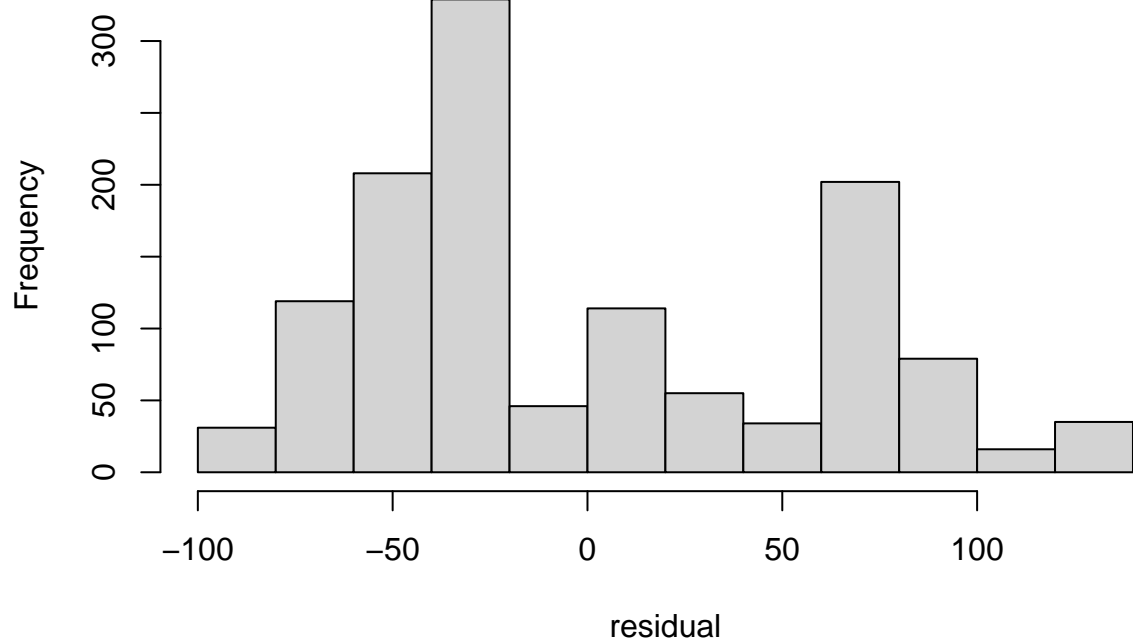


```
## summary statistics
## -----
## min: 59   max: 289
## median: 127
## mean: 154.2169
## estimated sd: 57.9311
## estimated skewness: 0.5680173
## estimated kurtosis: 2.125259
```

While uniform looks the closest, I'll try poisson

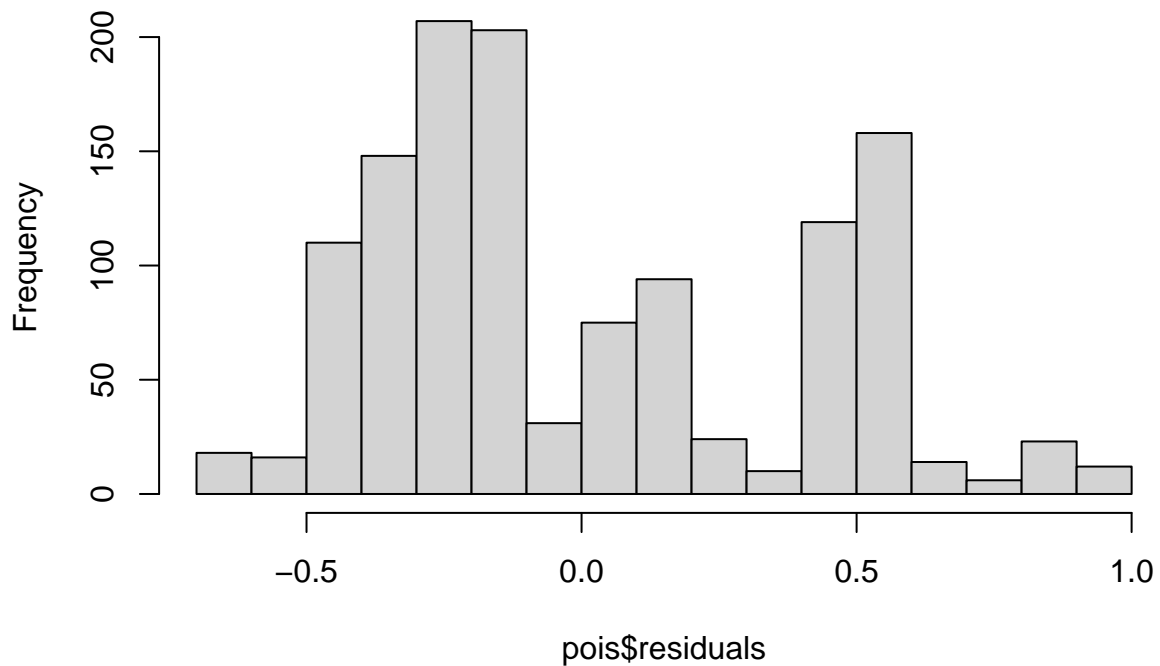
```
fit <- lm(spp_half_cover_date ~ state, data = green_kbs)
residual <- fit$residuals
hist(residual, main = "Raw residuals")
```

Raw residuals



```
pois <- glm(spp_half_cover_date ~ state, data = green_kbs, family = "poisson")  
hist(pois$residuals, main = "Poisson glm residuals")
```

Poisson glm residuals



Below I try a few different generalized linear models with poisson distribution:

An interaction between state and year, plus insecticide as a fixed effect and species and plot as random effects

```
moda <- glmer(spp_half_cover_date ~ state * year + insecticide +
  (1 | species) + (1 | plot), data = green_kbs, family = poisson)
```

```
## Warning: Some predictor variables are on very different scales: consider
## rescaling
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.0232094 (tol = 0.002, component 1)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model is nearly unidentifiable:
## - Rescale variables?;Model is nearly unidentifiable: large eigenvalue ratio
## - Rescale variables?
```

```
summary(moda)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: poisson ( log )
## Formula: spp_half_cover_date ~ state * year + insecticide + (1 | species) +
## (1 | plot)
## Data: green_kbs
```

```
##      AIC      BIC    logLik deviance df.resid
## 27686.8 27722.8 -13836.4 27672.8      1261
```

```
## Scaled residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -8.3234 -3.0092 -0.9901  2.4141 14.4417
```

```
## Random effects:
```

```
## Groups Name      Variance Std.Dev.
## plot   (Intercept) 0.003601 0.06001
## species (Intercept) 0.043398 0.20832
```

```
## Number of obs: 1268, groups: plot, 24; species, 21
```

```
## Fixed effects:
```

```
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -82.699485   4.741076 -17.443  <2e-16 ***
## stateambient    -15.581997   6.330178  -2.462   0.0138 *
## year              0.043454   0.002349  18.497  <2e-16 ***
## insecticideno_insects 0.036029   0.024939   1.445   0.1485
## stateambient:year  0.007736   0.003137   2.466   0.0137 *
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
```

```
## Correlation of Fixed Effects:
```

```
##              (Intr) sttmbn year  insct_
## stateambint -0.727
## year        -1.000  0.727
```

```
## insctcdn_ns -0.017  0.007  0.015
## statmbnt:yr  0.727 -1.000 -0.727 -0.007
## fit warnings:
## Some predictor variables are on very different scales: consider rescaling
## convergence code: 0
## Model failed to converge with max|grad| = 0.0232094 (tol = 0.002, component 1)
## Model is nearly unidentifiable: very large eigenvalue
## - Rescale variables?
## Model is nearly unidentifiable: large eigenvalue ratio
## - Rescale variables?
```


No interaction between state and year, but with state and insecticide as fixed effects and species and plot as random effects

```
modb <- glmer(spp_half_cover_date ~ state + year + insecticide +
  (1 | species) + (1 | plot), data = green_kbs, family = poisson)
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.00310689 (tol = 0.002, component 1)

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model is nearly unidentifiable:
## - Rescale variables?;Model is nearly unidentifiable: large eigenvalue ratio
## - Rescale variables?
```

```
summary(modb)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: poisson ( log )
## Formula: spp_half_cover_date ~ state + year + insecticide + (1 | species) +
## (1 | plot)
## Data: green_kbs
##
##      AIC      BIC   logLik deviance df.resid
## 27690.8 27721.7 -13839.4 27678.8      1262
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -8.3309 -3.0222 -0.9997  2.3954 14.2765
##
## Random effects:
## Groups Name Variance Std.Dev.
## plot (Intercept) 0.003606 0.06005
## species (Intercept) 0.043488 0.20854
## Number of obs: 1268, groups: plot, 24; species, 21
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -91.116751   3.172907 -28.717 <2e-16 ***
## stateambient      0.030236   0.024956  1.212 0.226
## year             0.047625   0.001572 30.293 <2e-16 ***
## insecticideno_insects 0.036450   0.024954  1.461 0.144
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##              (Intr) sttmbn year
## stateambint  0.002
## year        -1.000 -0.006
## insctcdn_ns -0.018 -0.002  0.014
## convergence code: 0
## Model failed to converge with max|grad| = 0.00310689 (tol = 0.002, component 1)
## Model is nearly unidentifiable: very large eigenvalue
## - Rescale variables?
## Model is nearly unidentifiable: large eigenvalue ratio
## - Rescale variables?
```

State and insecticide as fixed effects & year, species and plot as random effects

```
modc <- glmer(spp_half_cover_date ~ state + insecticide + (1 |  
  year) + (1 | species) + (1 | plot), data = green_kbs, family = poisson)  
summary(modc)
```

Because no distributions seems to match well, I'll try a Friedman's test

```
# friedman_kbs <- green_kbs %>%  
# friedman_test(spp_half_cover_date ~ state)
```

Error: Must extract column with a single valid subscript. x Subscript var can't be NA

Can't figure out what this means

If I include the blocks portion of the formula (from the documentation) I get this error

```
# friedman_kbs <- green_kbs %>%  
# friedman_test(spp_half_cover_date ~ state | plot)
```

Error in friedman.test.default(c(141L, 202L, 122L, 101L, 127L, 120L, 197L, : not an unreplicated complete block design

Permanova?

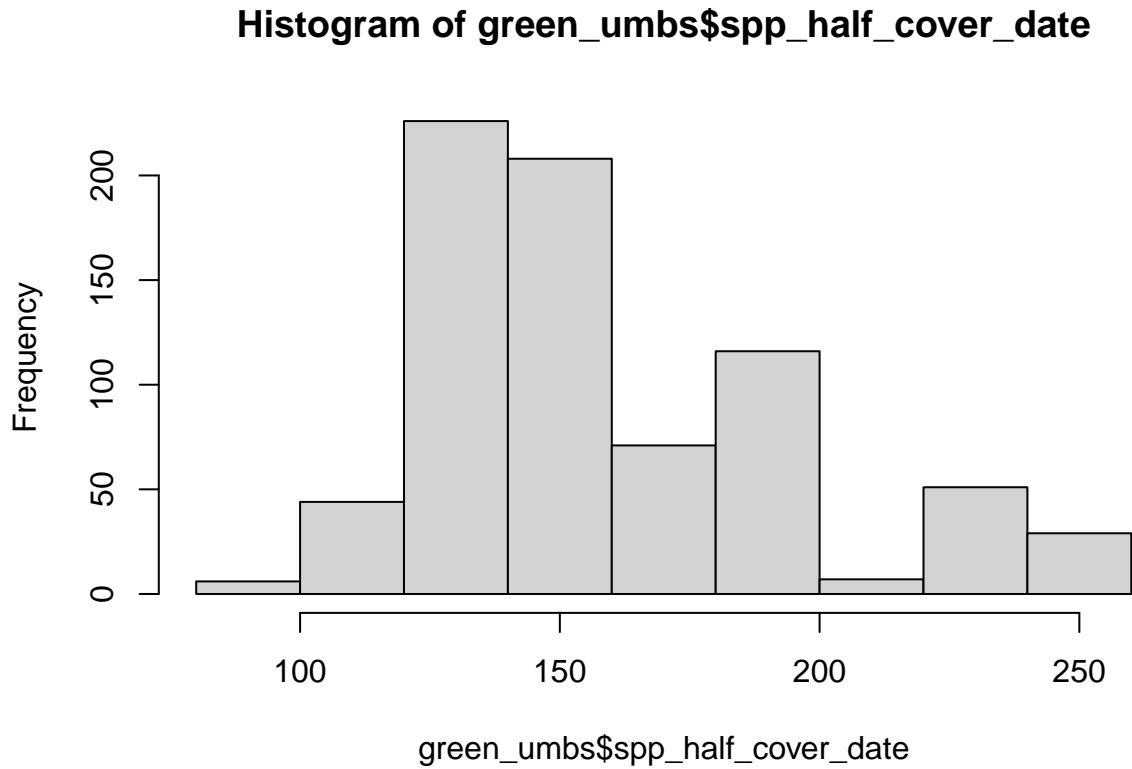
```
per1 <- adonis2(green_kbs$spp_half_cover_date ~ state * year +  
  insecticide, data = green_kbs)  
per1  
per2 <- adonis(formula = green_kbs$spp_half_cover_date ~ state *  
  year + insecticide, strata = green_kbs$plot, data = green_kbs)  
per2
```

With per2, when controlling for “plot”, there is a difference btwn treatments

UMBS

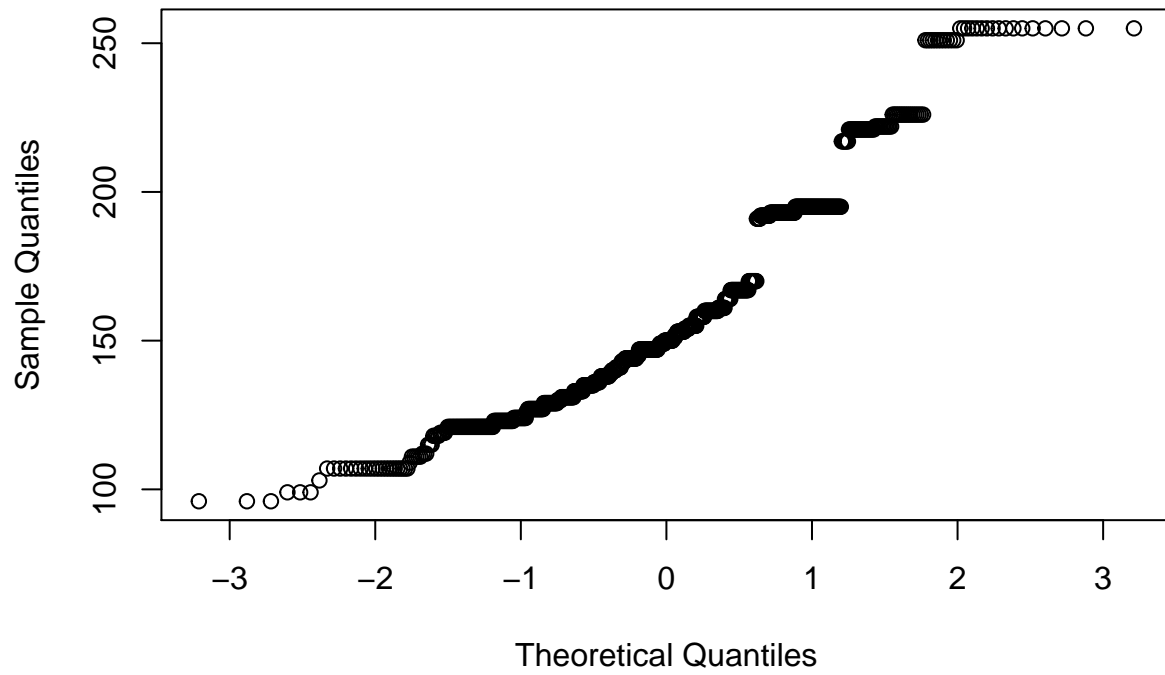
Checking for normality

```
hist(green_umbs$spp_half_cover_date)
```



```
qqnorm(green_umbs$spp_half_cover_date)
```

Normal Q-Q Plot

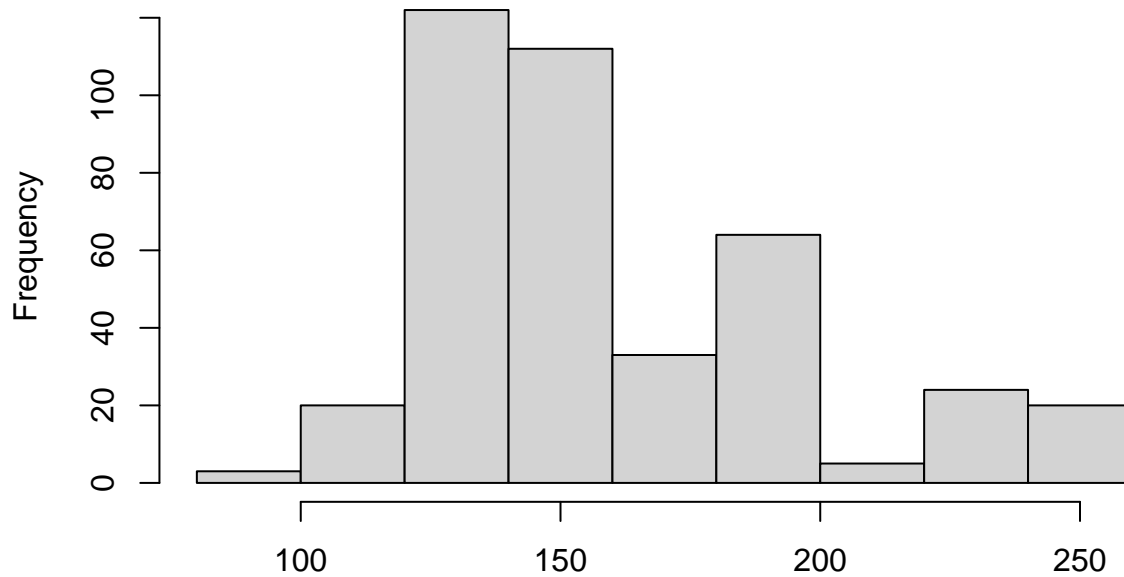


```
shapiro.test(green_umbs$spp_half_cover_date)
```

```
##  
##  Shapiro-Wilk normality test  
##  
## data:  green_umbs$spp_half_cover_date  
## W = 0.92247, p-value < 2.2e-16
```

```
hist(green_umbs$spp_half_cover_date[green_kbs$state == "ambient"])
```

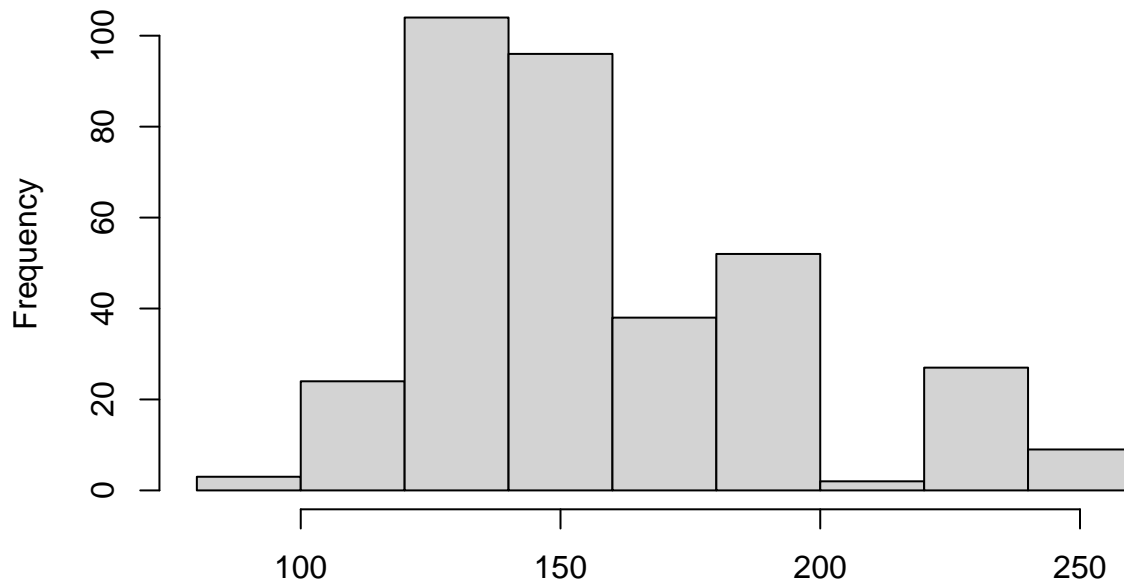
istogram of green_umbs\$spp_half_cover_date[green_kbs\$state == "am



green_umbs\$spp_half_cover_date[green_kbs\$state == "ambient"]

```
hist(green_umbs$spp_half_cover_date[green_kbs$state == "warmed"])
```

istogram of green_umbs\$spp_half_cover_date[green_kbs\$state == "wa



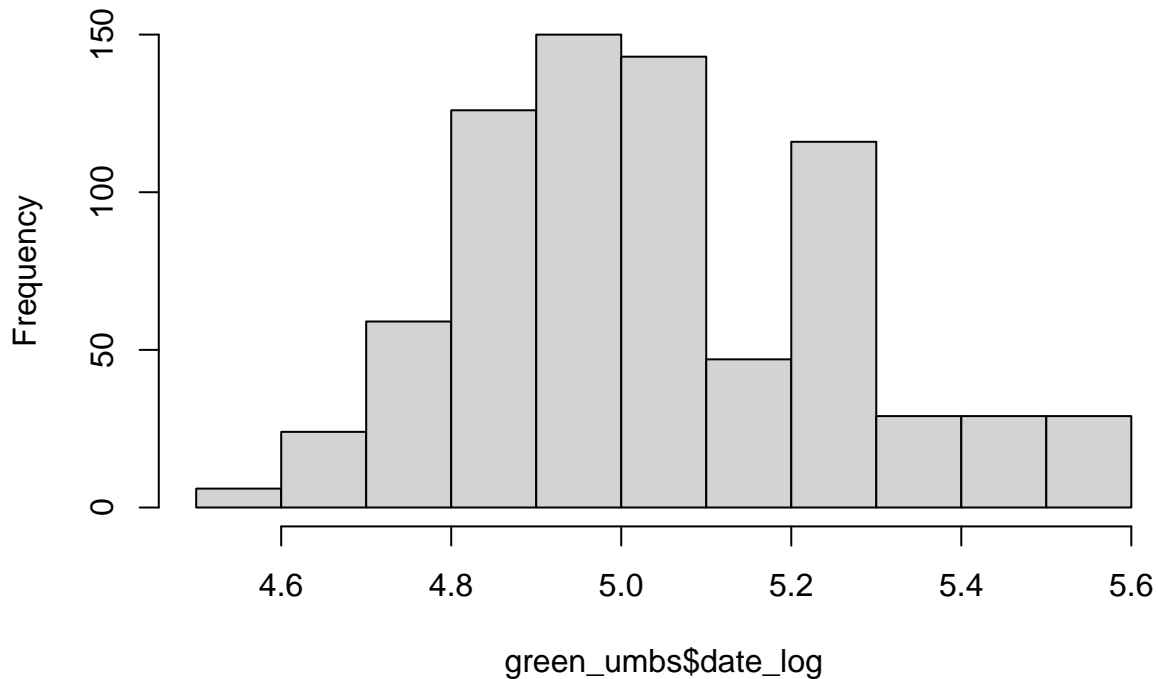
green_umbs\$spp_half_cover_date[green_kbs\$state == "warmed"]

These look pretty good

Trying log transformation

```
green_umbs$date_log <- log(green_umbs$spp_half_cover_date)
hist(green_umbs$date_log)
```

Histogram of green_umbs\$date_log



```
shapiro.test(green_umbs$date_log)
```

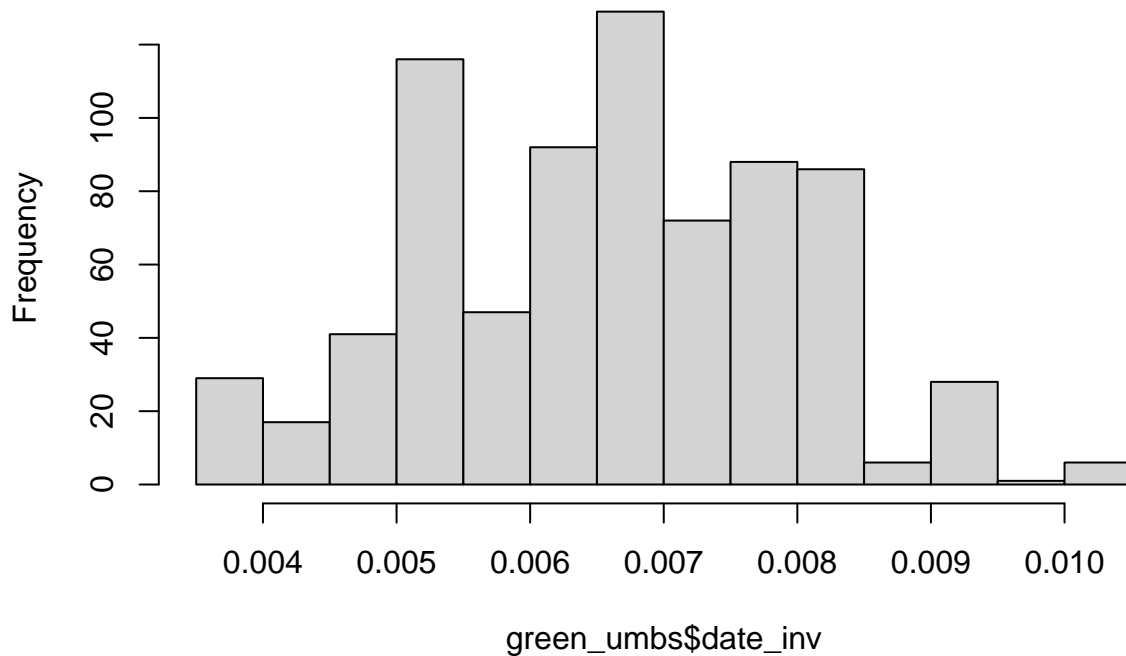
```
##
##  Shapiro-Wilk normality test
##
## data:  green_umbs$date_log
## W = 0.96356, p-value = 8.516e-13
```

I think this looks good but shapiro-wilk is lower than 0.05

Trying inverse tranformation

```
green_umbs$date_inv <- 1/(green_umbs$spp_half_cover_date)
hist(green_umbs$date_inv)
```

Histogram of green_umbs\$date_inv



```
shapiro.test(green_umbs$date_inv)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  green_umbs$date_inv
## W = 0.97928, p-value = 6.952e-09
```

This also looks good but is also still low for shapiro-wilk