

# warmXtrophic Project: Flowering Phenology Analyses

Moriah Young

January 17, 2022

## Load in packages & data

```
# clear all existing data
rm(list = ls())

# Load packages
library(tidyverse)
library(ggplot2)
library(lme4)
library(lmerTest)
library(emmeans)
library(vegan)
library(car)
library(rstatix)
library(scales)
library(fitdistrplus)
library(moments) # for calculating skewness of data
library(ggpubr)
library(jtools) # summ() function
library(predictmeans)
library(olsrr)
library(car)
library(fitdistrplus)
library(ggpubr)
library(interactions)
library(sjPlot)
library(effects)
library(glmmTMB)
library(GGally) # ggpairs() function
library(bbmle) # AICtab() function

# Set working directory
Sys.getenv("L1DIR")

## [1] "/Volumes/GoogleDrive/Shared drives/SpaCE_Lab_warmXtrophic/data/L1"

L0_dir <- Sys.getenv("LODIR")
L1_dir <- Sys.getenv("L1DIR")
L2_dir <- Sys.getenv("L2DIR")
```

```

# Set ggplot2 plotting This code for ggplot2 sets the theme to mostly black and
# white (Arial font, and large font, base size=24)
theme_set(theme_bw(14))
theme_update(axis.text.x = element_text(size = 12, angle = 90), axis.text.y = element_text(size = 12))

# Read in data
flwr_species <- read.csv(file.path(L2_dir, "phenology/final_flwr_species_L2.csv")) # species level data
flwr_plot <- read.csv(file.path(L2_dir, "phenology/final_flwr_plot_L2.csv")) # plot level data
flwr_plot_origin <- read.csv(file.path(L2_dir, "phenology/final_flwr_plot_origin_L2.csv")) # plot level data
flwr_plot_growthhabit <- read.csv(file.path(L2_dir, "phenology/final_flwr_plot_growthhabit_L2.csv")) # plot level data

# get rid of 'X' column that shows up
flwr_species$X <- NULL
flwr_plot$X <- NULL
flwr_plot_origin$X <- NULL
flwr_plot_growthhabit$X <- NULL
# take a look at the data to see if looks good
View(flwr_species)
View(flwr_plot)

# Order warmed and ambient so that warmed shows up first in plotting (and is
# default is red = warmed; blue = ambient). First make it a factor
flwr_species$state <- as.factor(flwr_species$state)
levels(flwr_species$state)

## [1] "ambient" "warmed"

# [1] 'ambient' 'warmed'
flwr_species$state <- factor(flwr_species$state, levels(flwr_species$state)[c(2, 1)])
levels(flwr_species$state)

## [1] "warmed" "ambient"

# [1] 'warmed' 'ambient'

# again for plot level data
flwr_plot$state <- as.factor(flwr_plot$state)
levels(flwr_plot$state)

## [1] "ambient" "warmed"

# [1] 'ambient' 'warmed'
flwr_plot$state <- factor(flwr_plot$state, levels(flwr_plot$state)[c(2, 1)])
levels(flwr_plot$state)

## [1] "warmed" "ambient"

```

```

# [1] 'warmed' 'ambient'

umbs_flwr_plot <- subset(flwr_plot, site == "umbs") # pull out umbs only data at plot level
# delete 2021 data from umbs dataframe (doesn't make sense to have bc we can't
# get min flower/sd, median flower, or flower duration from 2021 UMBS)
umbs_flwr_plot <- subset(umbs_flwr_plot, year != "2021")
umbs_flwr_spp <- subset(flwr_species, site == "umbs") # pull out umbs only data at species level
umbs_flwr_spp <- subset(umbs_flwr_spp, year != "2021") # delete 2021 data
kbs_flwr_plot <- subset(flwr_plot, site == "kbs") # pull out kbs only data at plot level
kbs_flwr_spp <- subset(flwr_species, site == "kbs") # pull out kbs only data at species level
kbs_flwr_plot_origin <- subset(flwr_plot_origin, site == "kbs")
kbs_flwr_plot_growthhabit <- subset(flwr_plot_growthhabit, site == "kbs")
umbs_flwr_plot_origin <- subset(flwr_plot_origin, site == "umbs")
umbs_flwr_plot_origin <- subset(flwr_plot_origin, year != "2021") # delete 2021 data
umbs_flwr_plot_growthhabit <- subset(flwr_plot_growthhabit, site == "umbs")
umbs_flwr_plot_growthhabit <- subset(flwr_plot_growthhabit, year != "2021") # delete 2021 data

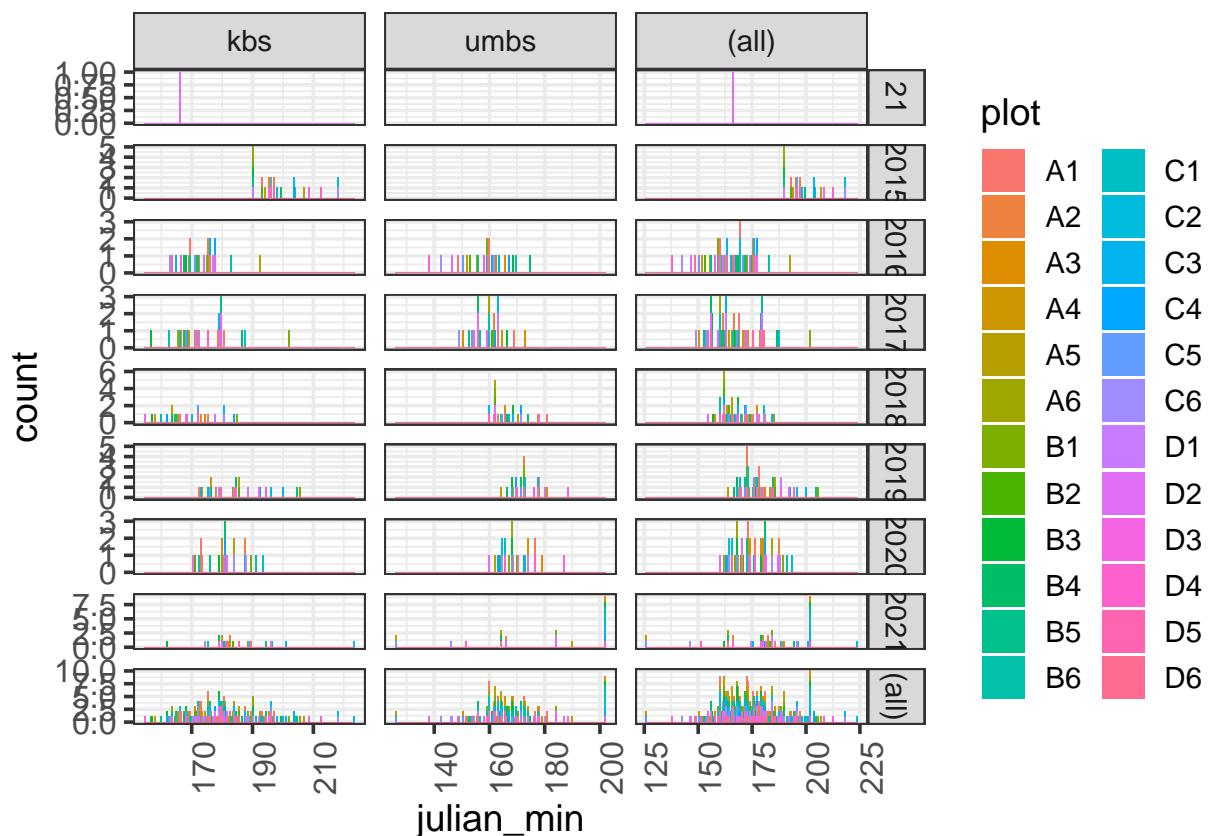
```

## Data exploration for minimum (first) Julian date of flowering at the PLOT for both sites

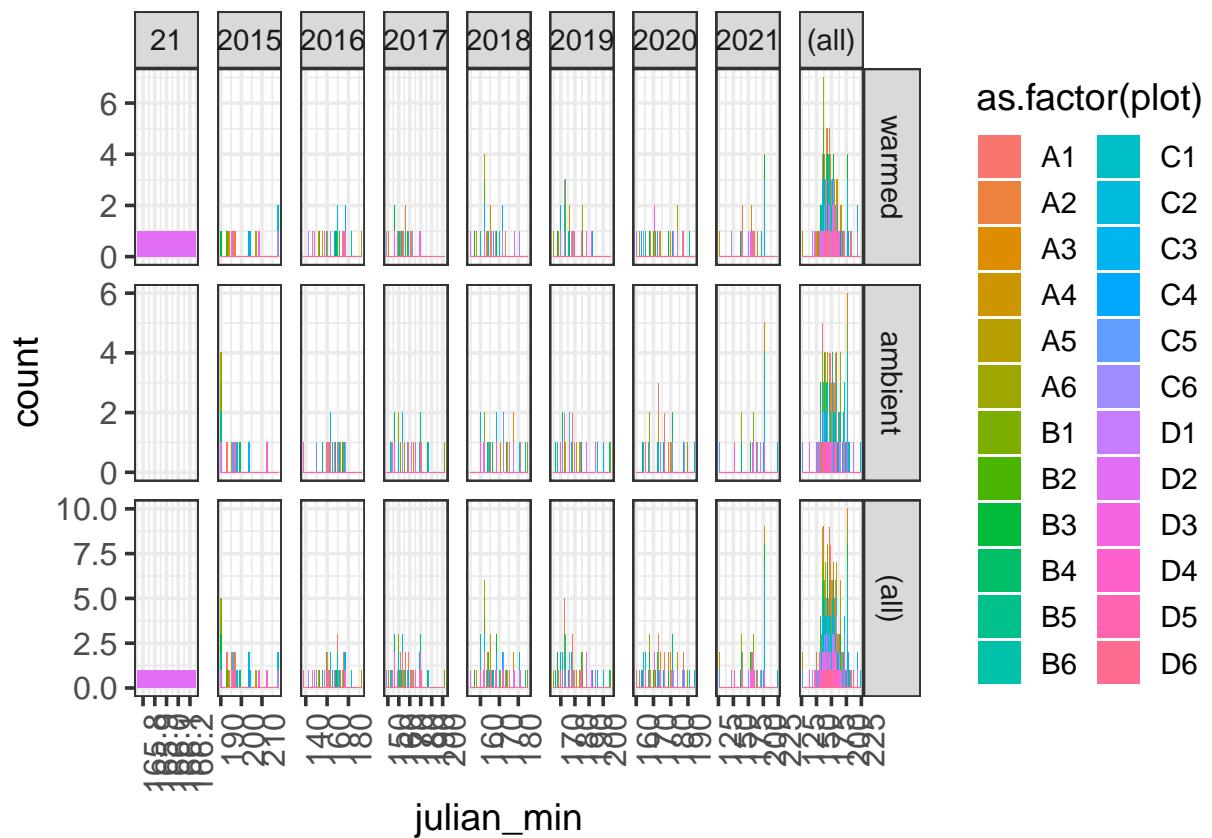
```

# Visualizing avg minimum Julian date for both sites at the PLOT LEVEL
ggplot(fltr_plot, aes(julian_min, fill = plot)) + geom_histogram(binwidth = 0.5) +
  facet_grid(year ~ site, margins = TRUE, scales = "free")

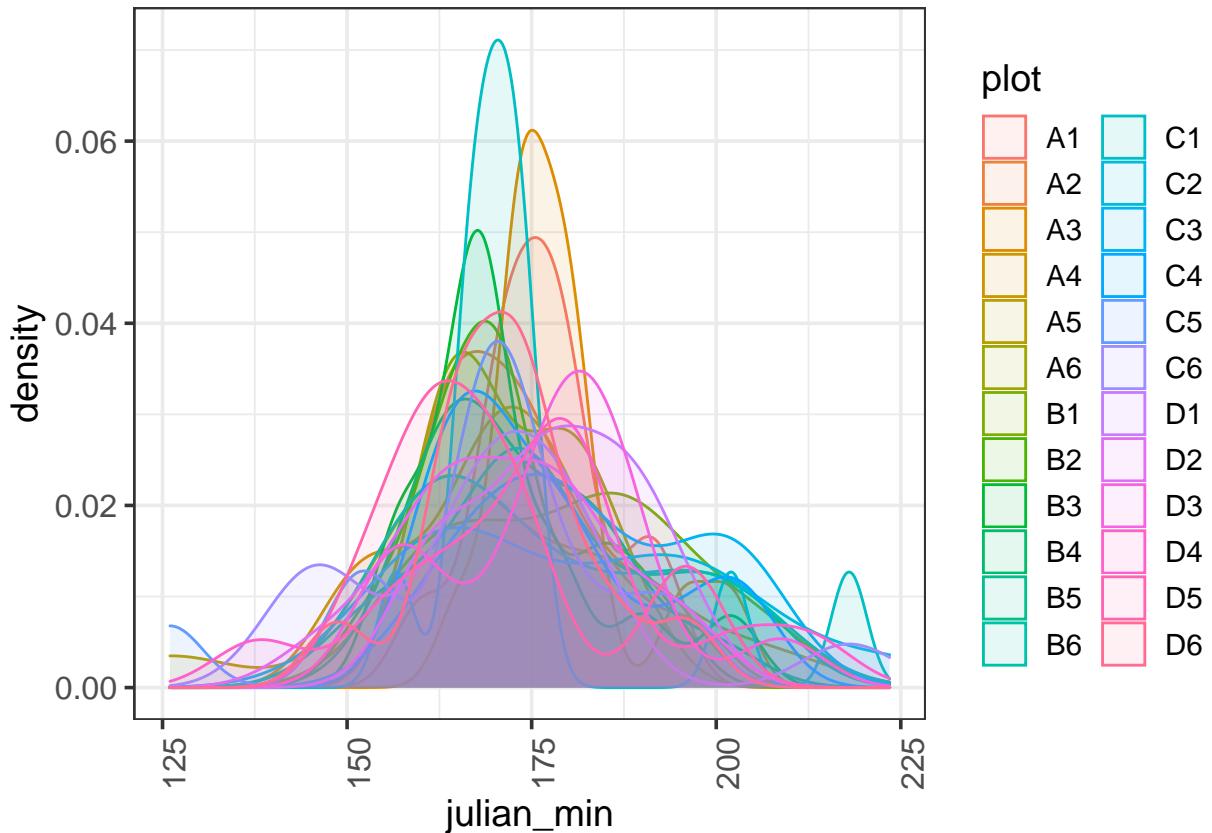
```



```
ggplot(fltrr_plot, aes(julian_min, fill = as.factor(plot))) + geom_histogram(binwidth = 0.5) +
  facet_grid(state ~ year, margins = TRUE, scales = "free")
```



```
ggplot(fltrr_plot, aes(julian_min, fill = plot, color = plot)) + geom_density(alpha = 0.1)
```



```
ggplot(flwrr_plot, aes(julian_min, fill = plot, color = plot)) + geom_density(alpha = 0.1) +
  facet_wrap(~year)
```

```
## Warning: Groups with fewer than two data points have been dropped.
## Warning: Groups with fewer than two data points have been dropped.
## Warning: Groups with fewer than two data points have been dropped.
## Warning: Groups with fewer than two data points have been dropped.
## Warning: Groups with fewer than two data points have been dropped.
## Warning: Groups with fewer than two data points have been dropped.
## Warning: Groups with fewer than two data points have been dropped.
## Warning: Groups with fewer than two data points have been dropped.
## Warning: Groups with fewer than two data points have been dropped.
## Warning: Groups with fewer than two data points have been dropped.
## Warning: Groups with fewer than two data points have been dropped.
## Warning: Groups with fewer than two data points have been dropped.
```



```
## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf
```

```

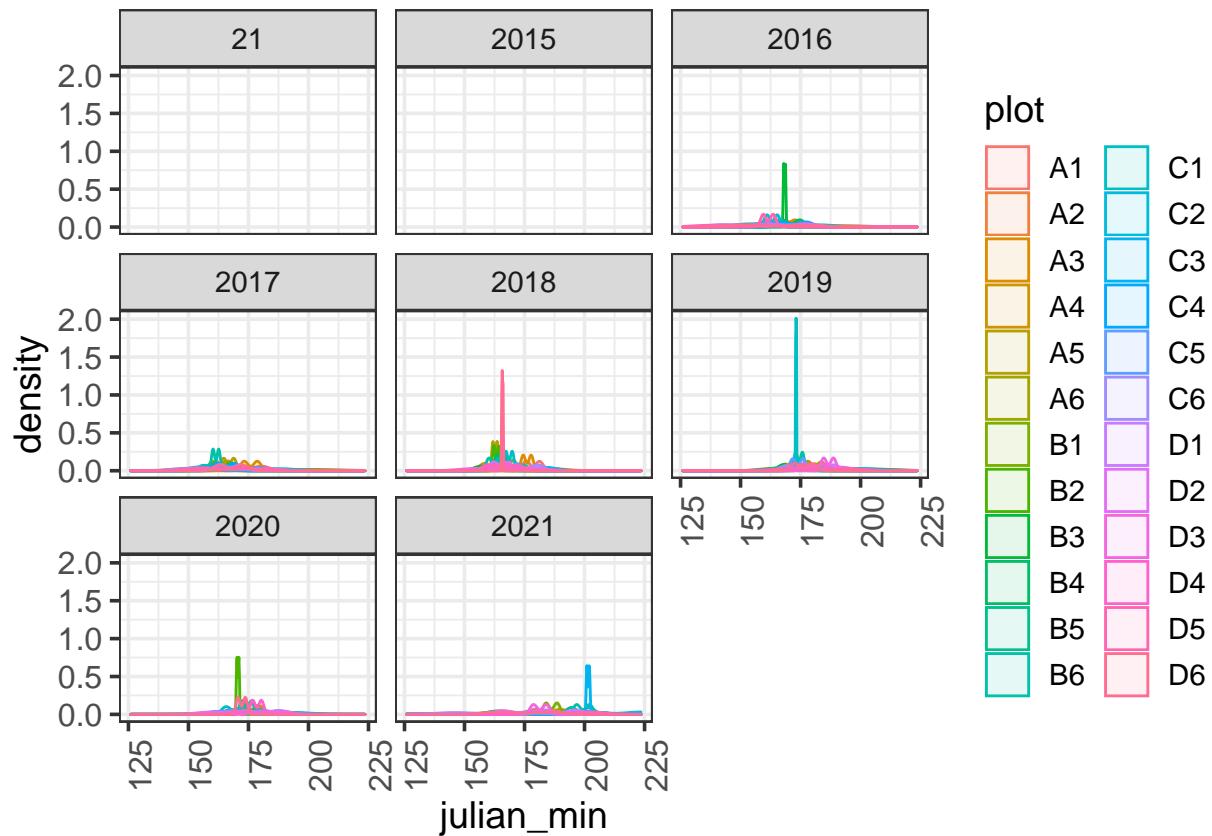
## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

```



```

ggplot(fltrr_plot, aes(julian_min, fill = plot, color = plot)) + geom_density(alpha = 0.1) +
  facet_wrap(~year + plot)

```

```

## Warning: Groups with fewer than two data points have been dropped.

## Warning: Groups with fewer than two data points have been dropped.

## Warning: Groups with fewer than two data points have been dropped.

## Warning: Groups with fewer than two data points have been dropped.

## Warning: Groups with fewer than two data points have been dropped.

```



```
## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf
```

```

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

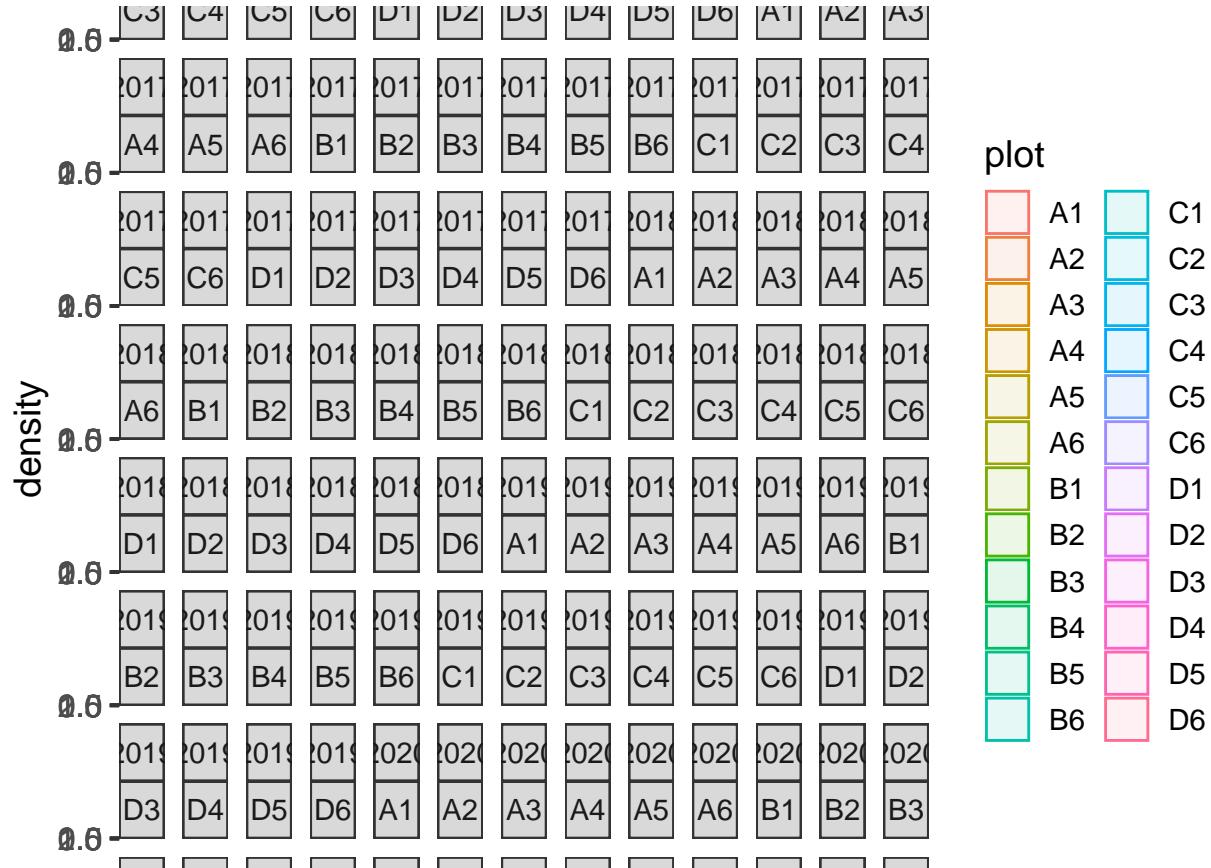
## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

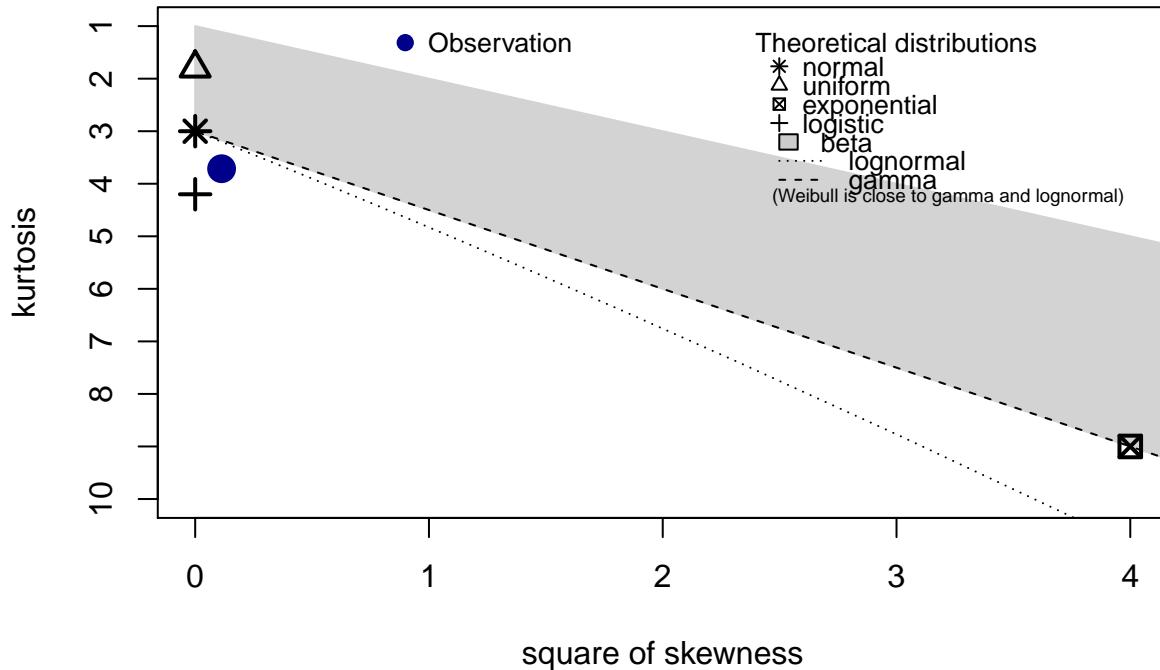
## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max; returning -
## Inf

```



```
descdist(flwr_plot$julian_min, discrete = FALSE)
```

### Cullen and Frey graph

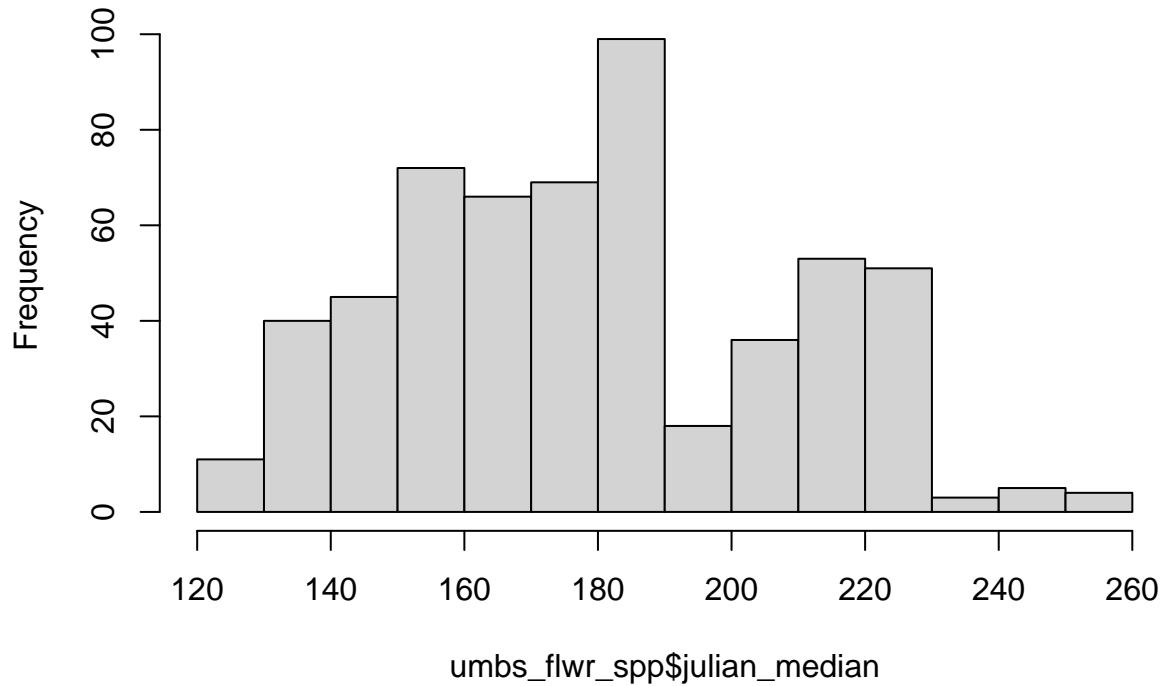


```
## summary statistics
## -----
## min: 126   max: 223.5
## median: 172.7778
## mean: 174.7747
## estimated sd: 14.72798
## estimated skewness: 0.3363232
## estimated kurtosis: 3.71398
```

### UMBS SPECIES LEVEL - Looking at MEDIAN JULIAN DAY

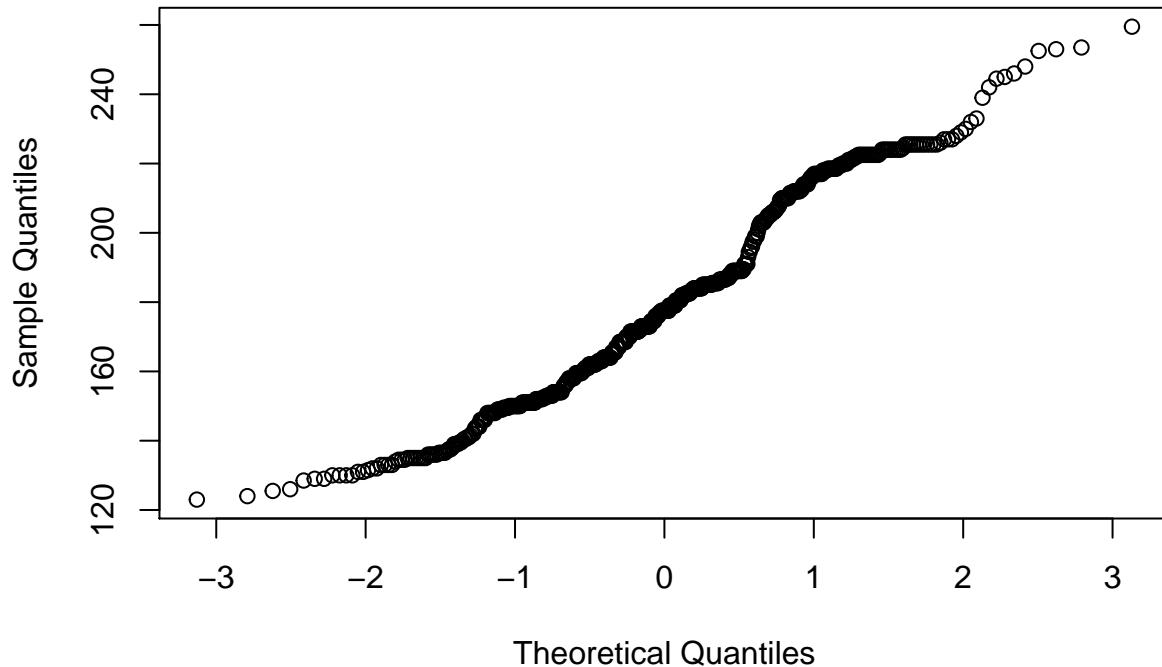
```
### UMBS ####
hist(umbs_flwr_spp$julian_median)
```

### Histogram of umbs\_flwr\_spp\$julian\_median



```
qqnorm(umbs_flwr_spp$julian_median) # this looks good
```

### Normal Q-Q Plot



```

shapiro.test(umbs_flwr_spp$julian_median) # pvalue is < 0.05 so we reject the null hypothesis that the

##  

## Shapiro-Wilk normality test  

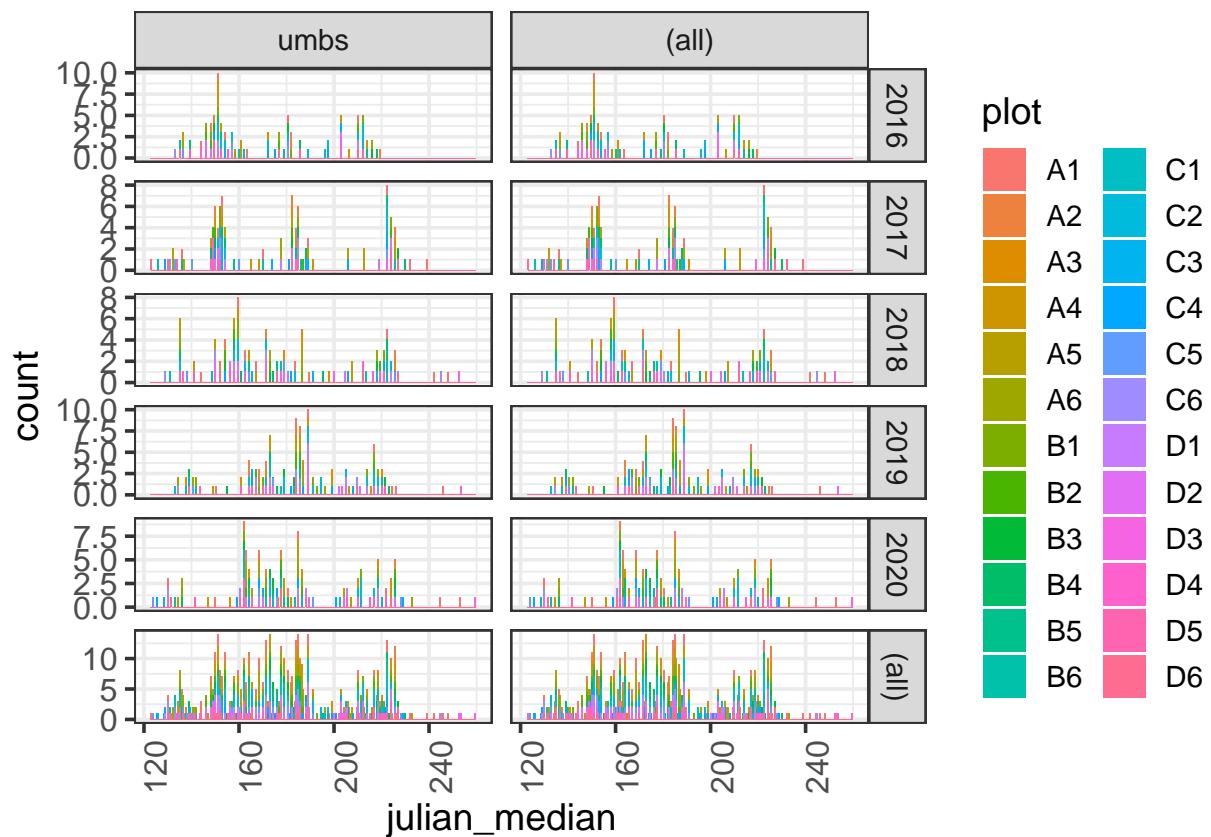
##  

## data: umbs_flwr_spp$julian_median  

## W = 0.9706, p-value = 2.693e-09

# Visualizing median Julian date for umbs at the PLOT LEVEL
ggplot(umbs_flwr_spp, aes(julian_median, fill = plot)) + geom_histogram(binwidth = 0.5) +
  facet_grid(year ~ site, margins = TRUE, scales = "free")

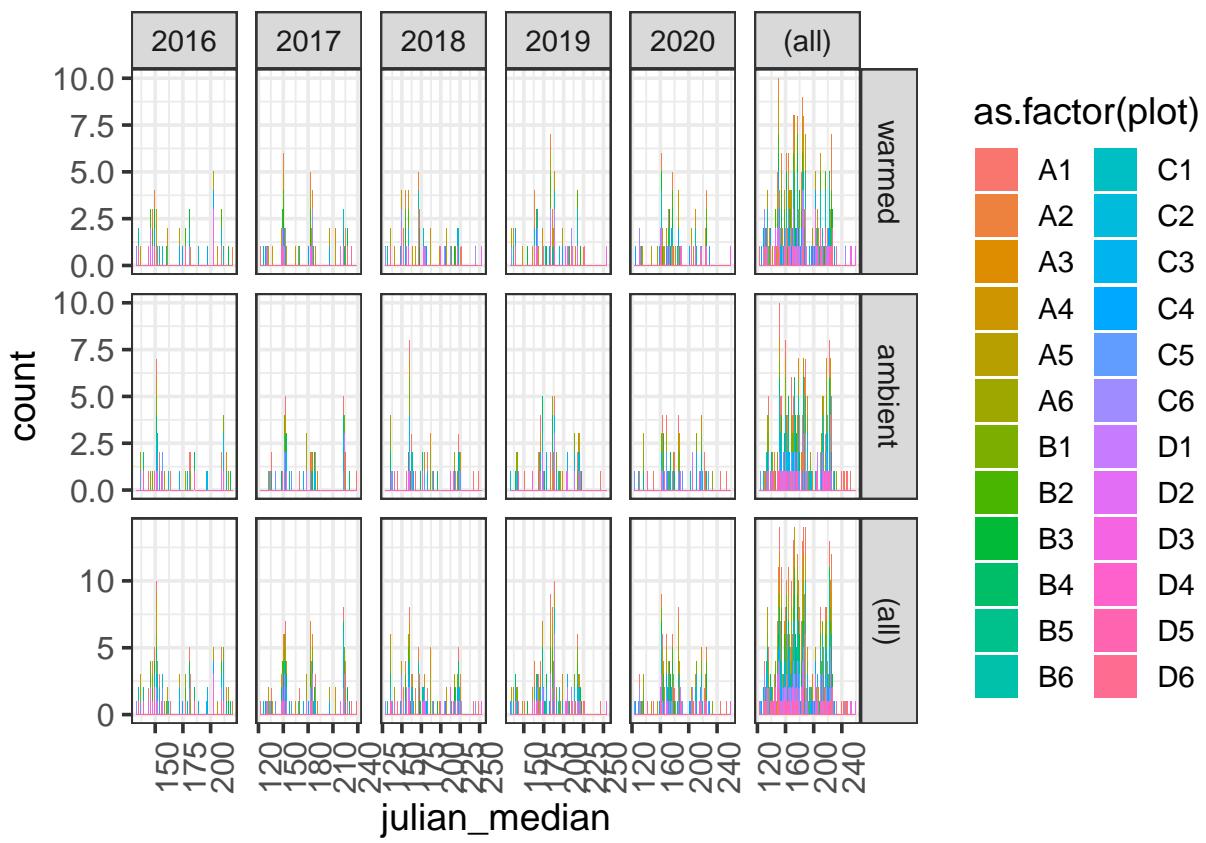
```



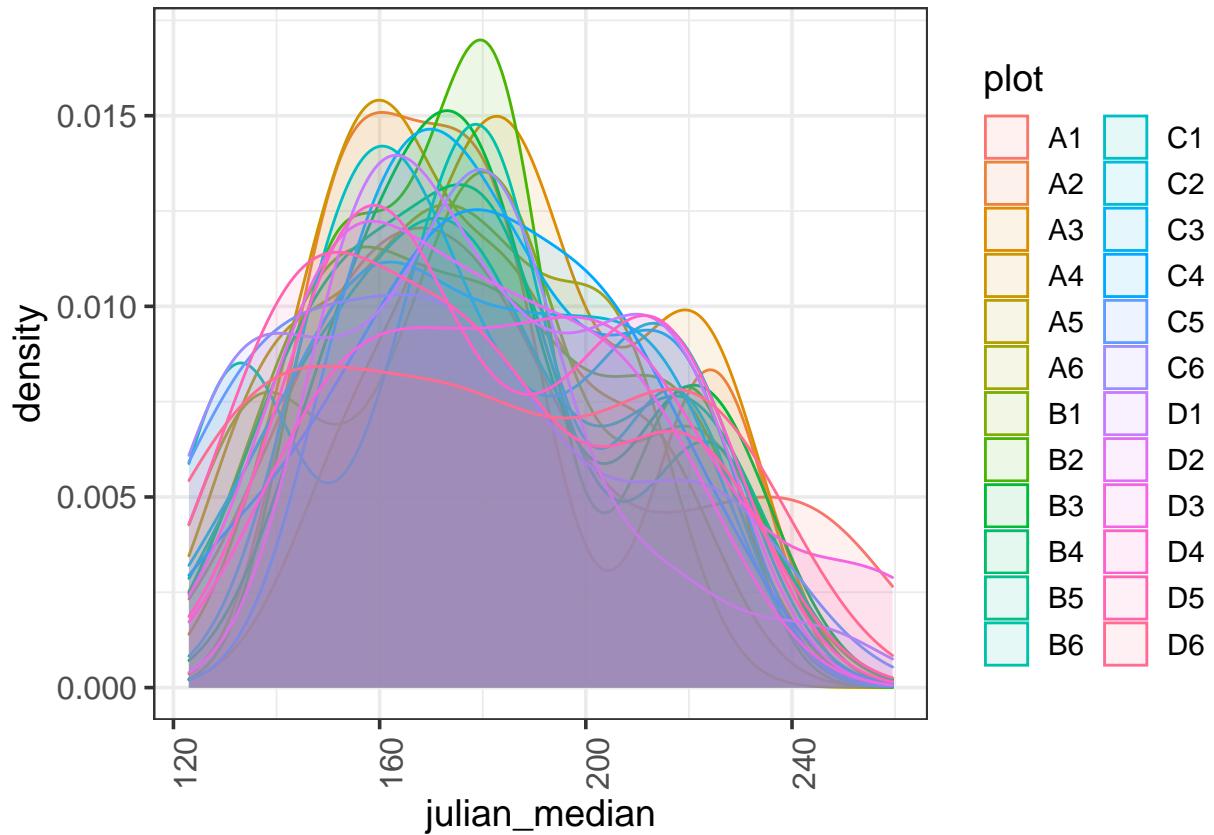
```

ggplot(umbs_flwr_spp, aes(julian_median, fill = as.factor(plot))) + geom_histogram(binwidth = 0.5) +
  facet_grid(state ~ year, margins = TRUE, scales = "free")

```



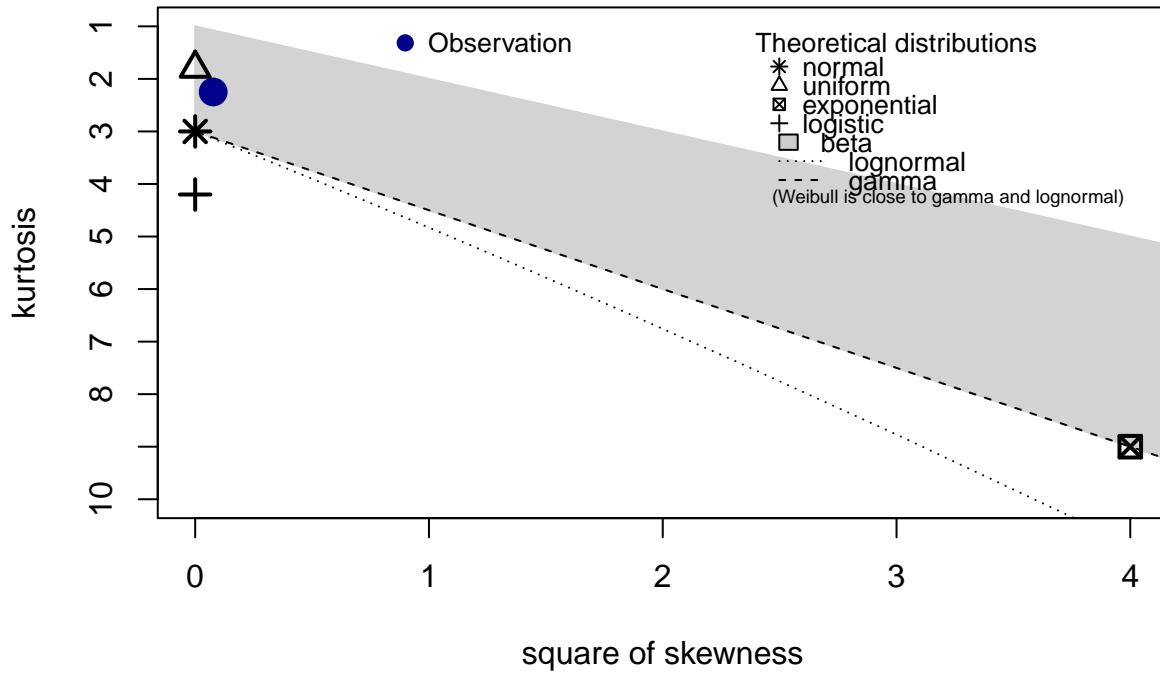
```
ggplot(umbs_flwr_spp, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1)
```



```
# ggplot(umbs_flwr_plot, aes(julian_median, fill = plot, color=plot)) +
# geom_density(alpha = 0.1) + facet_wrap(~year) ggplot(umbs_flwr_plot,
# aes(julian_median, fill = plot, color=plot)) + geom_density(alpha = 0.1) +
# facet_wrap(~year + plot)
```

```
# Exploring distributions for these right-skewed data:
descdist(umbs_flwr_spp$julian_median, discrete = FALSE)
```

## Cullen and Frey graph

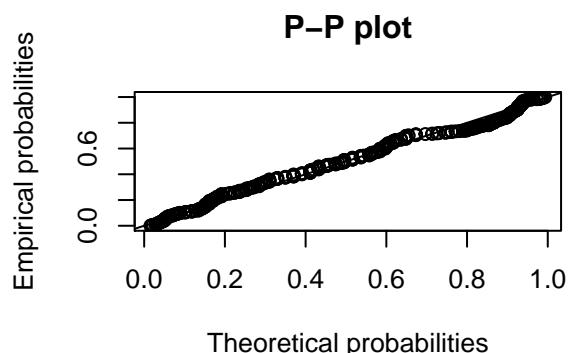
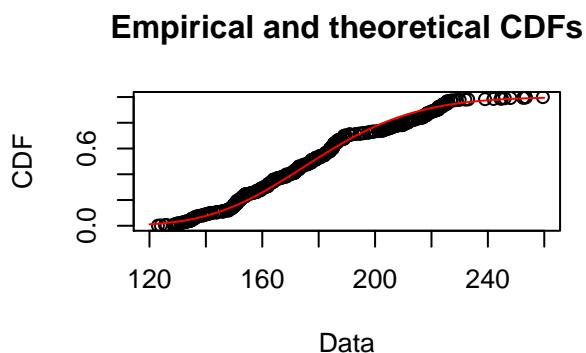
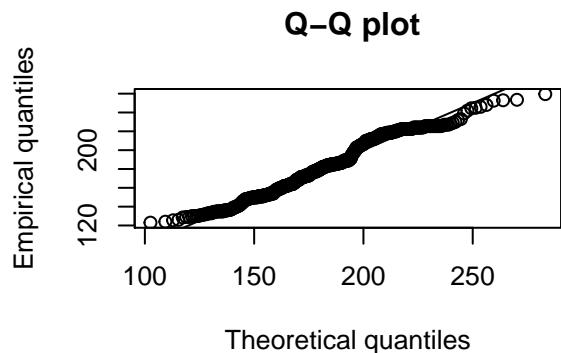
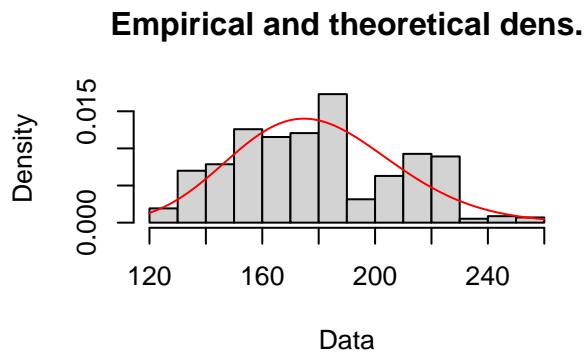


```

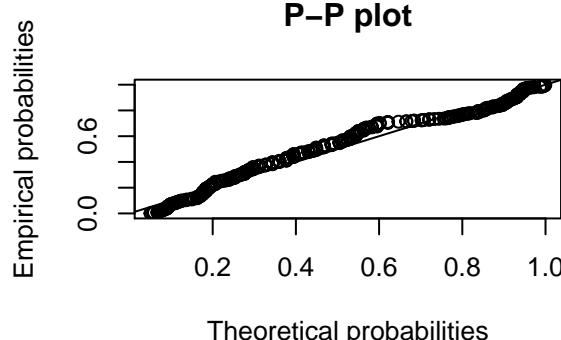
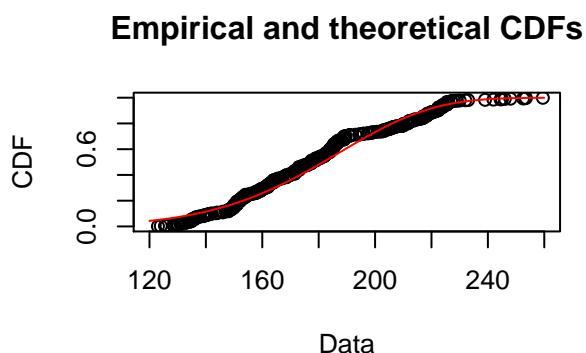
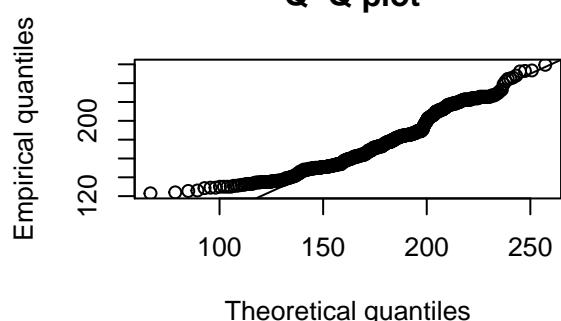
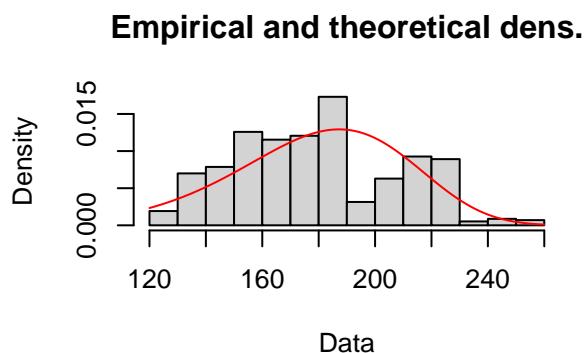
## summary statistics
## -----
## min: 123   max: 259.5
## median: 177.5
## mean: 179.3706
## estimated sd: 28.90832
## estimated skewness: 0.2778292
## estimated kurtosis: 2.249028

# Gamma distribution
fit.gamma <- fitdist(umbs_flwr_spp$julian_median, "gamma")
plot(fit.gamma)

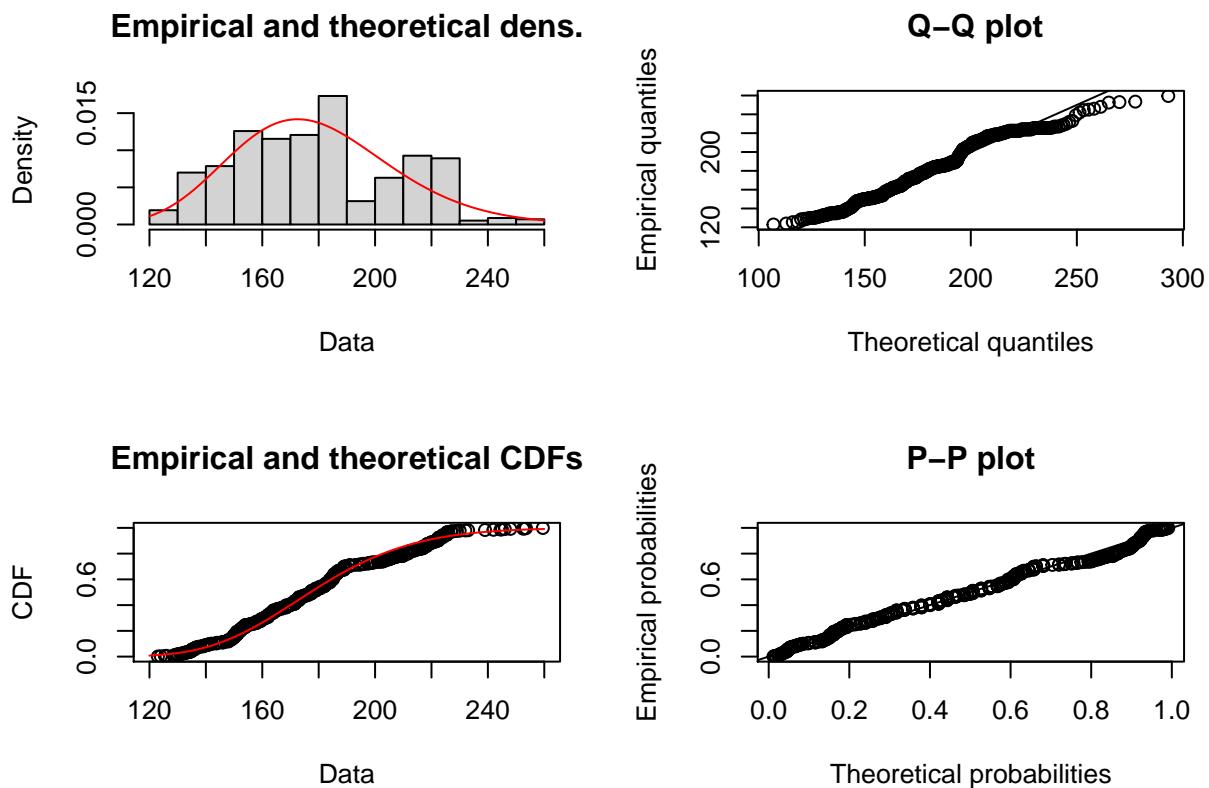
```



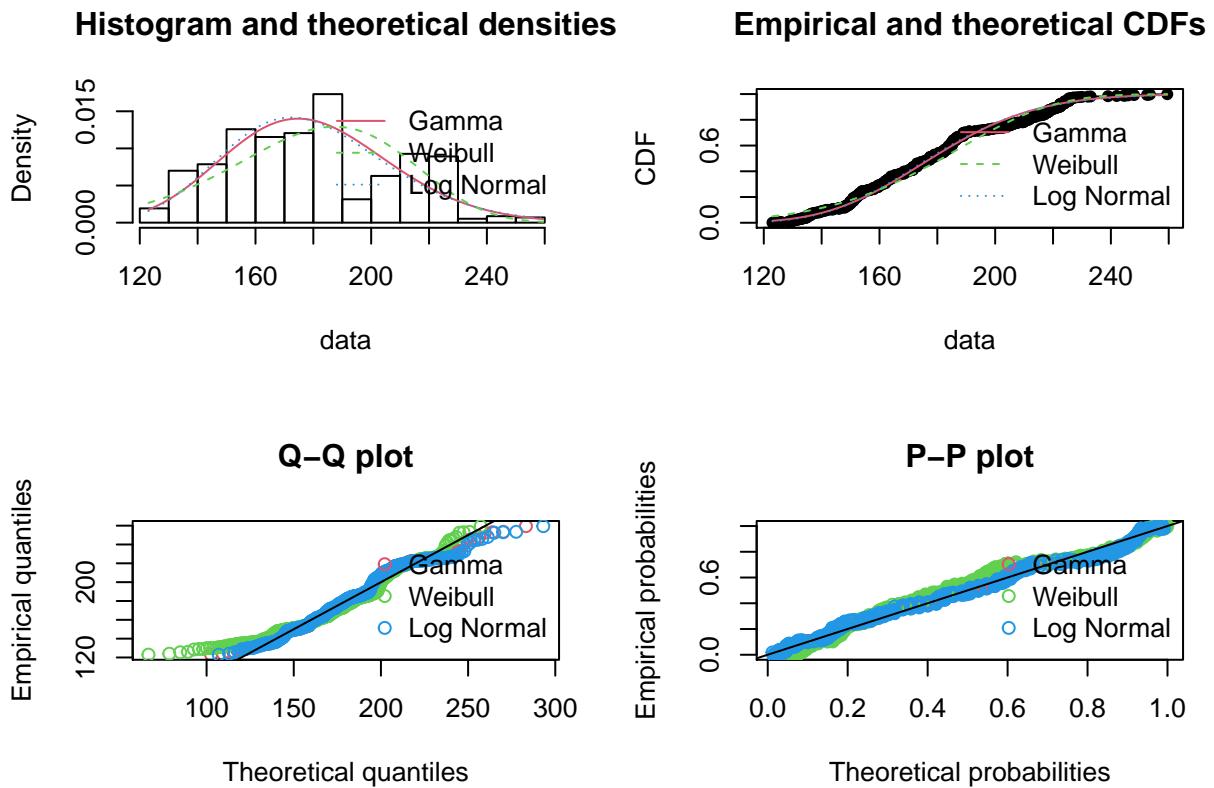
```
# Weibull distribution
fit.weibull <- fitdist(umbs_flwr_spp$julian_median, "weibull")
plot(fit.weibull)
```



```
# Lognormal distribution
fit.ln <- fitdist(umbs_flwr_spp$julian_median, "lnorm")
plot(fit.ln)
```



```
par(mfrow = c(2, 2))
plot.legend <- c("Gamma", "Weibull", "Log Normal")
denscomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
cdfcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
qqcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
ppcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
```



```
# Goodness of fit comparisons across fits (can't include the sqrt normal bc it
# becomes diff response values)
gofstat(list(fit.gamma, fit.weibull, fit.ln), fitnames = c("Gamma", "Weibull", "Log Normal"))

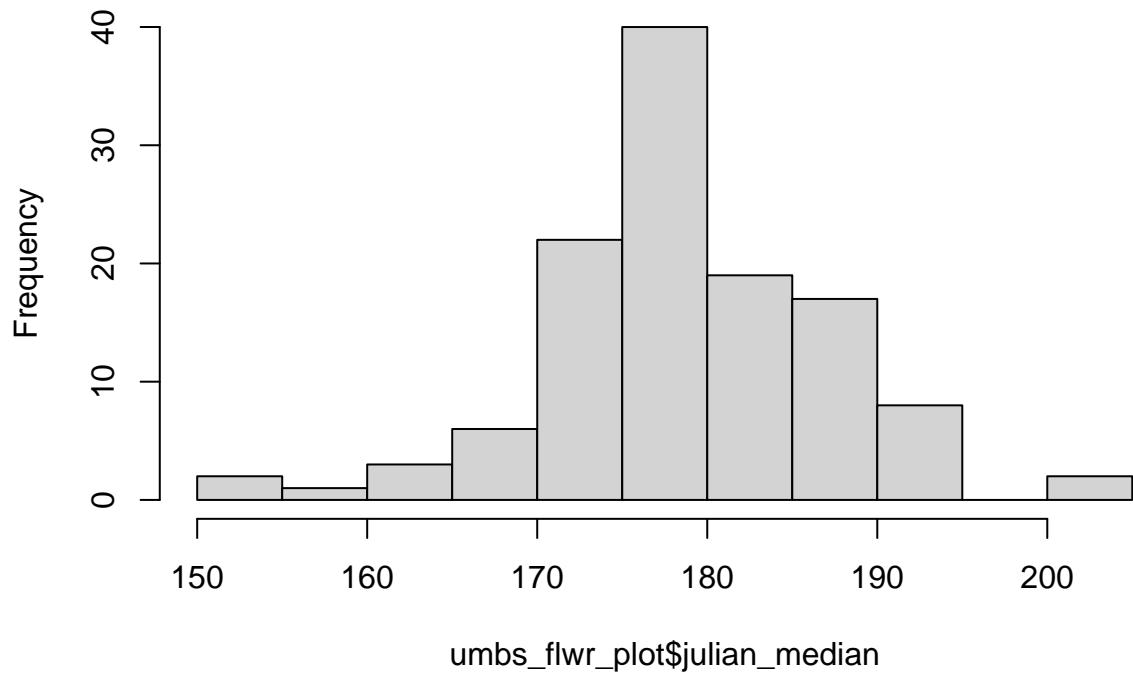
## Goodness-of-fit statistics
##                                     Gamma   Weibull Log Normal
## Kolmogorov-Smirnov statistic 0.07251101 0.1040666 0.07215405
## Cramer-von Mises statistic   0.52738159 1.2693500 0.46115964
## Anderson-Darling statistic   3.91501557 7.8885552 3.58465528
##
## Goodness-of-fit criteria
##                                     Gamma   Weibull Log Normal
## Akaike's Information Criterion 5461.481 5512.411 5460.162
## Bayesian Information Criterion 5470.180 5521.109 5468.860

# Lognormal and gamma are very close, gamma slightly better, but going to go with
# log transforming
```

## UMBS PLOT LEVEL - Looking at MEDIAN JULIAN DAY

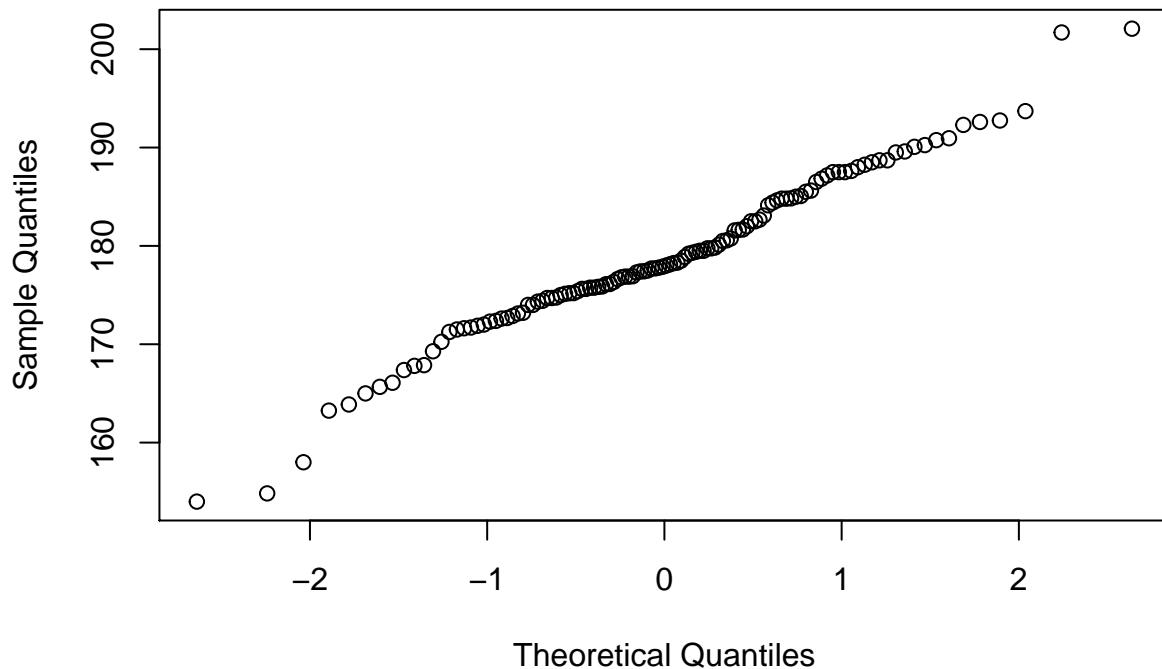
```
### UMBS ####
hist(umbs_flwr_plot$julian_median)
```

### Histogram of umbs\_flwr\_plot\$julian\_median



```
qqnorm(umbs_flwr_plot$julian_median)
```

### Normal Q-Q Plot



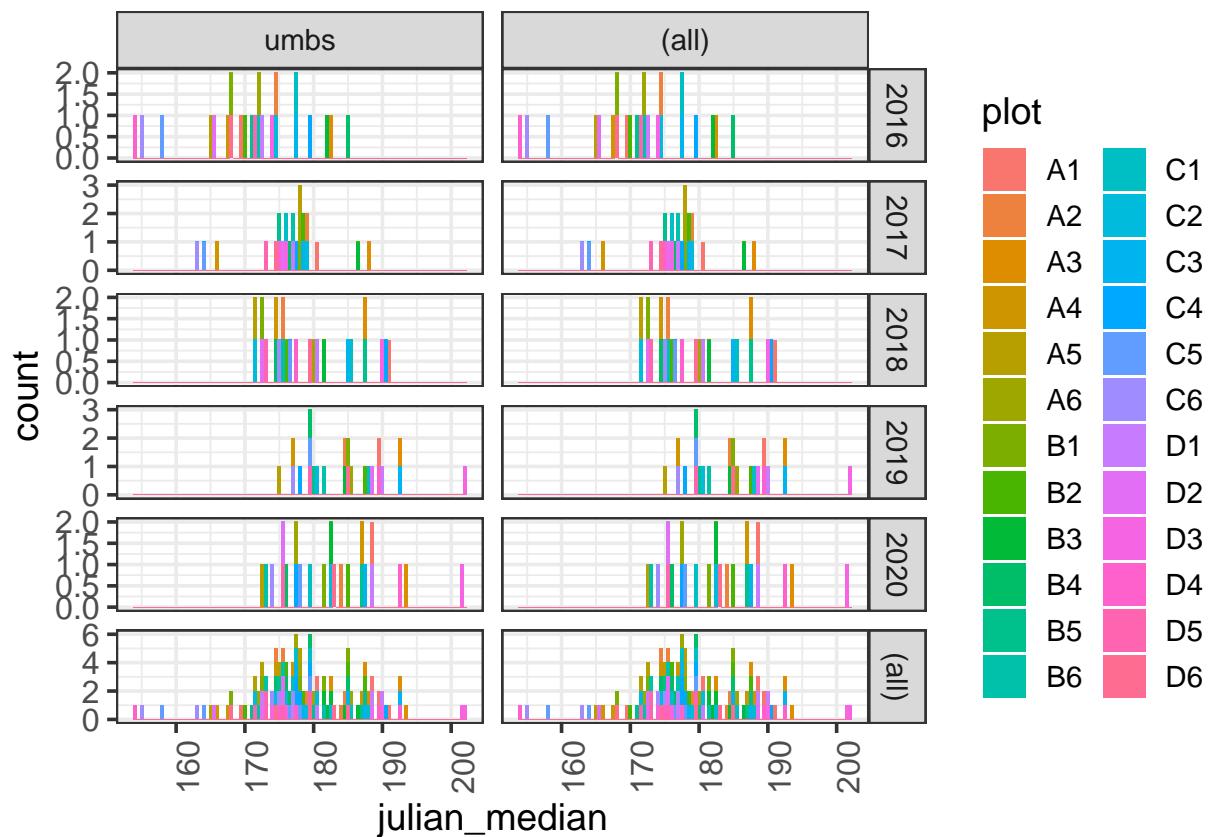
```

shapiro.test(umbs_flwr_plot$julian_median) # pvalue is > 0.05 so we accept the null hypothesis that the data is normally distributed

## 
## Shapiro-Wilk normality test
## 
## data: umbs_flwr_plot$julian_median
## W = 0.9798, p-value = 0.0682

# Visualizing median Julian date for umbs at the PLOT LEVEL
ggplot(umbs_flwr_plot, aes(julian_median, fill = plot)) + geom_histogram(binwidth = 0.5) +
  facet_grid(year ~ site, margins = TRUE, scales = "free")

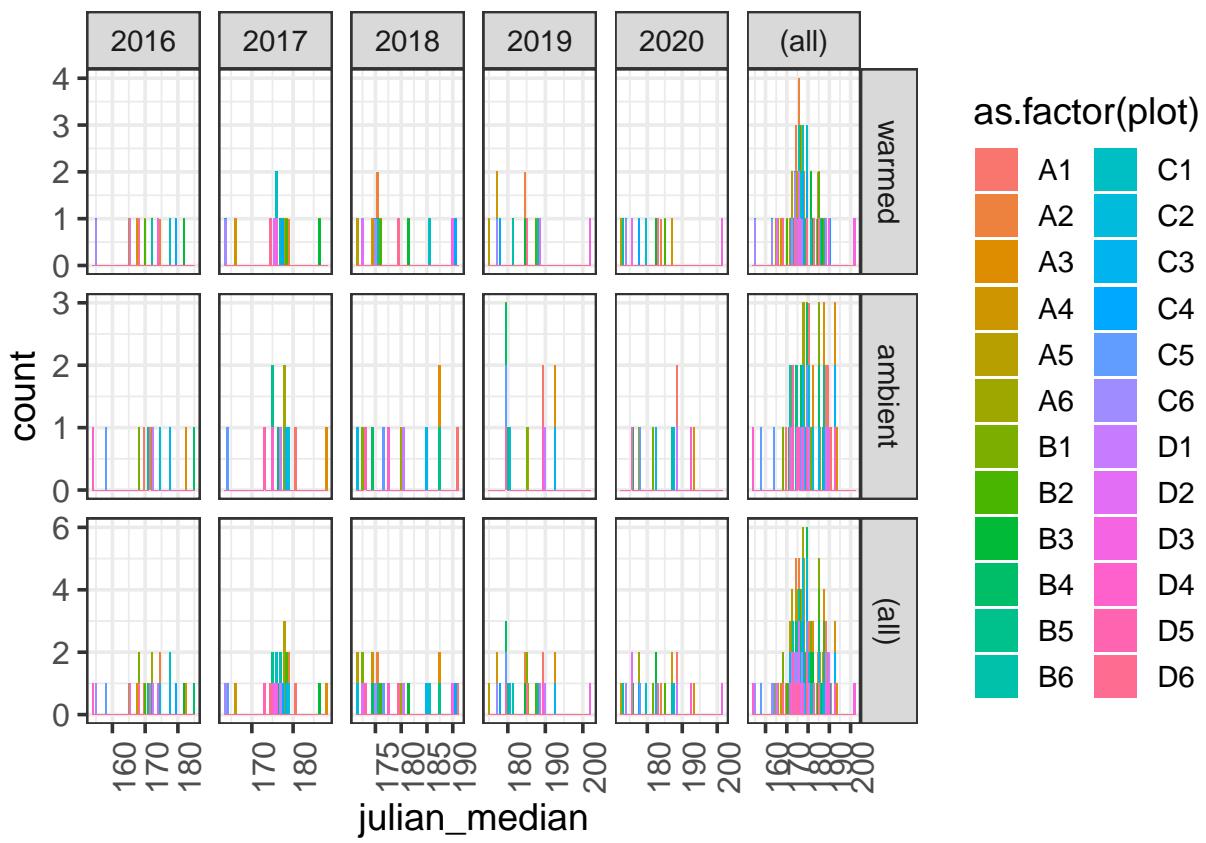
```



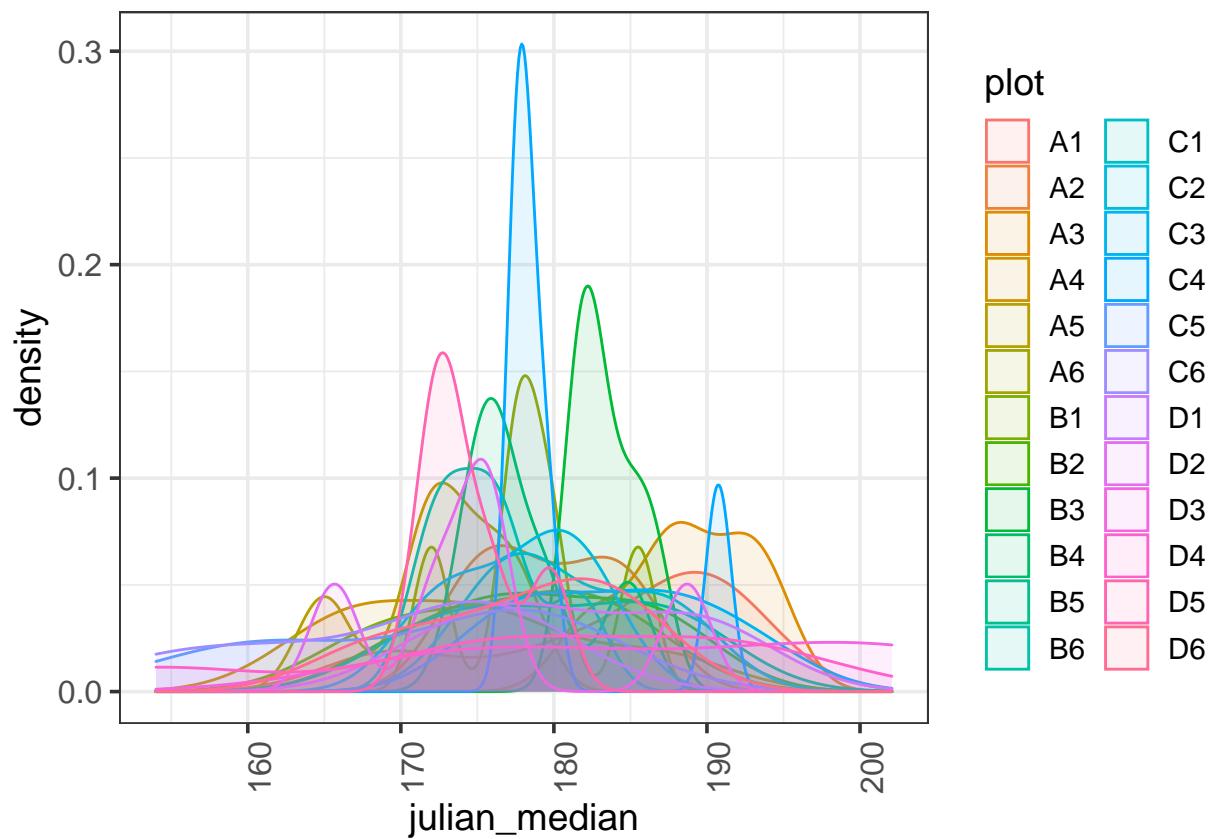
```

ggplot(umbs_flwr_plot, aes(julian_median, fill = as.factor(plot))) + geom_histogram(binwidth = 0.5) +
  facet_grid(state ~ year, margins = TRUE, scales = "free")

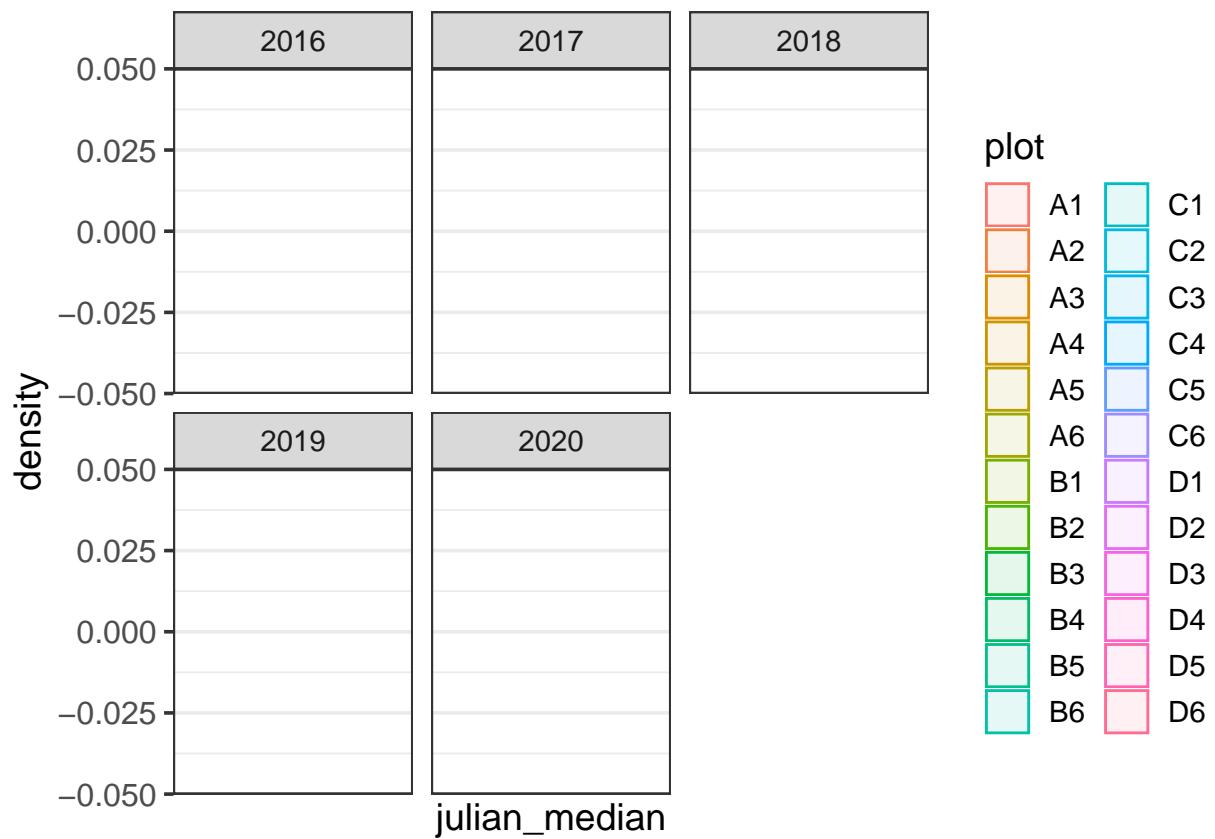
```



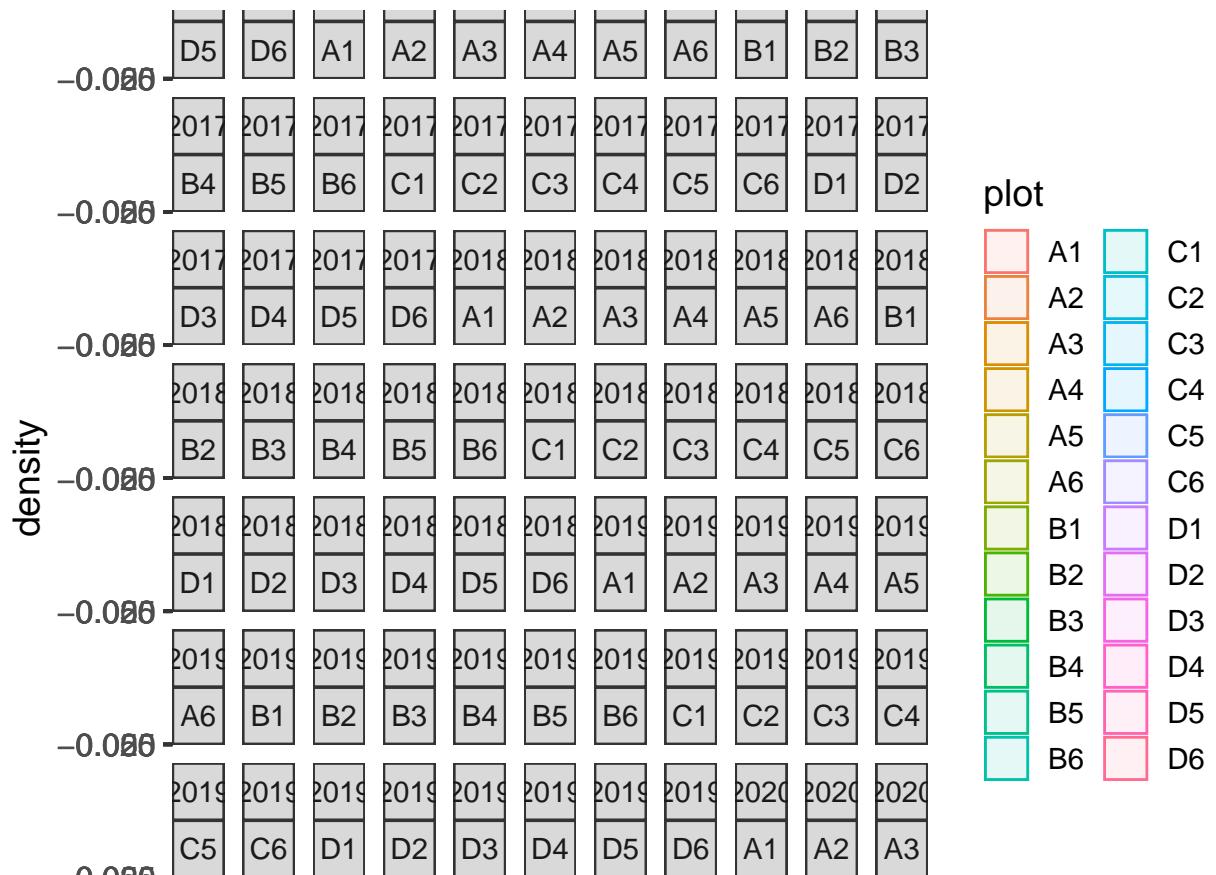
```
ggplot(umbs_flwr_plot, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1)
```



```
ggplot(umbs_flwr_plot, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1) +
  facet_wrap(~year)
```

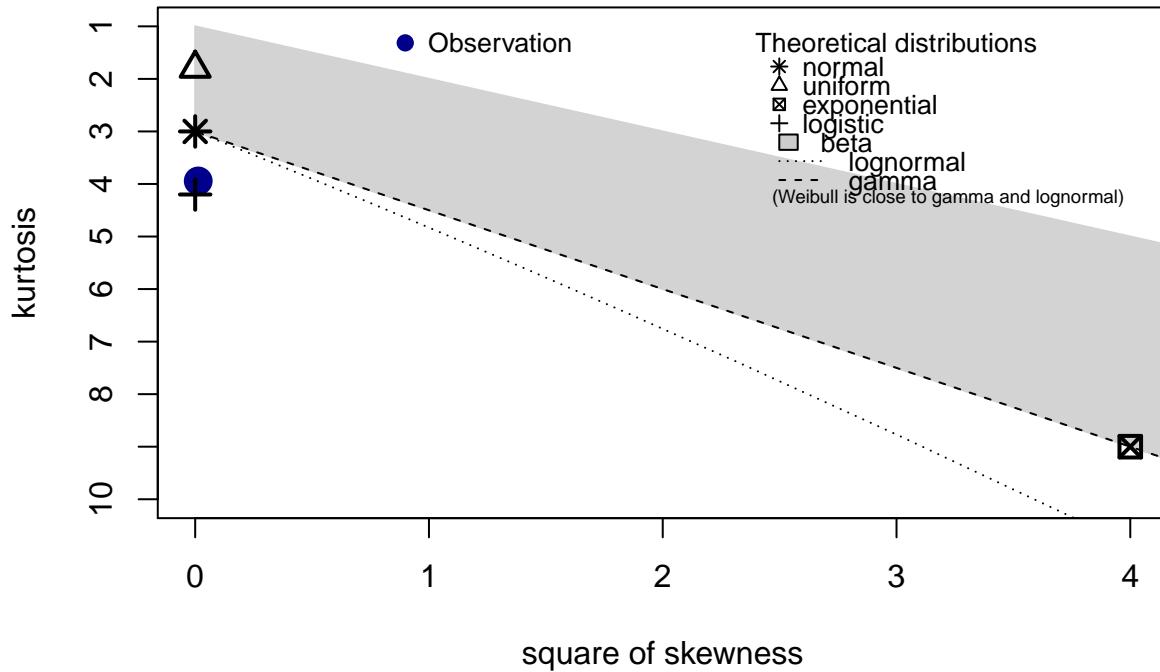


```
ggplot(umbs_flwr_plot, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1) +
  facet_wrap(~year + plot)
```



```
# Exploring distributions for these right-skewed data:  
descdist(umbs_flwr_plot$julian_median, discrete = FALSE)
```

## Cullen and Frey graph

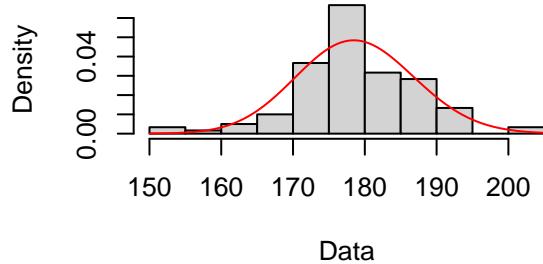
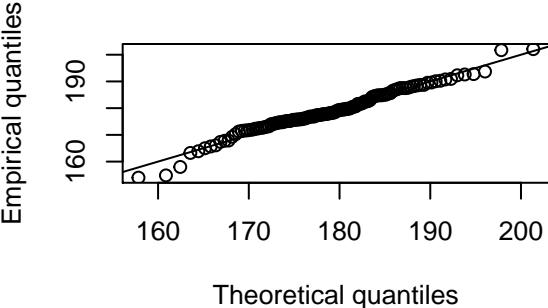
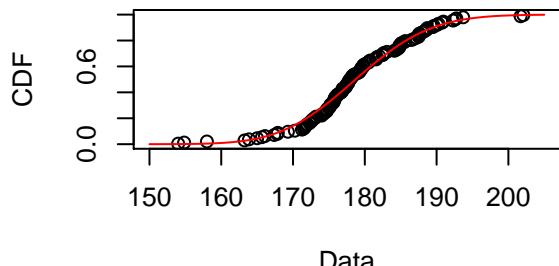
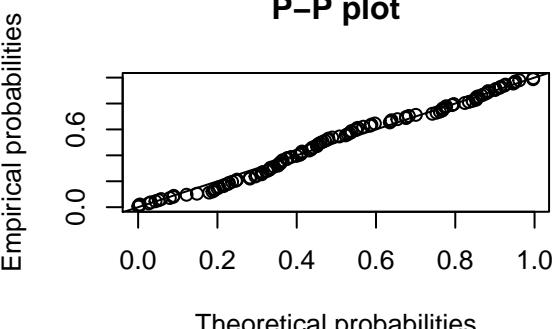


```

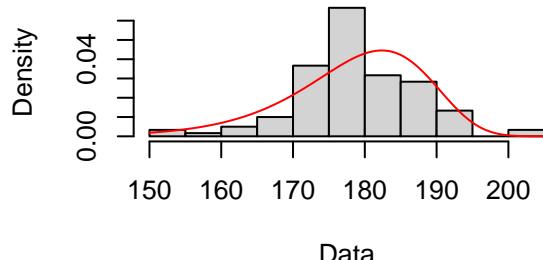
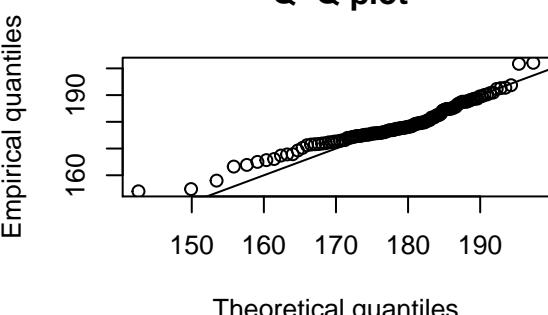
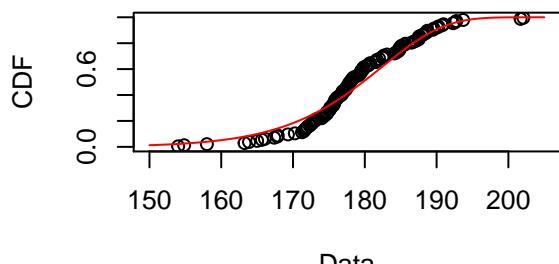
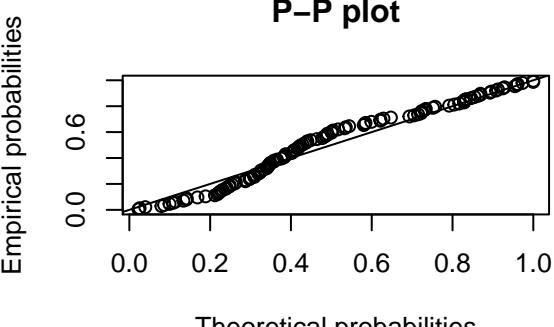
## summary statistics
## -----
## min: 154   max: 202.0833
## median: 177.9375
## mean: 178.8387
## estimated sd: 8.251076
## estimated skewness: -0.1139959
## estimated kurtosis: 3.94229

# Gamma distribution
fit.gamma <- fitdist(umbs_flwr_plot$julian_median, "gamma")
plot(fit.gamma)

```

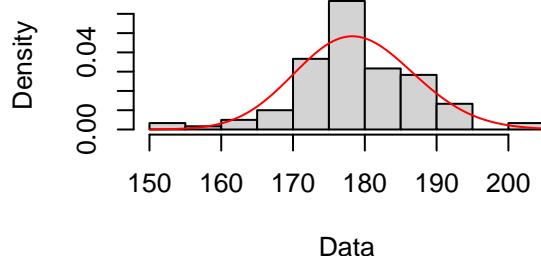
**Empirical and theoretical dens.****Q-Q plot****Empirical and theoretical CDFs****P-P plot**

```
# Weibull distribution
fit.weibull <- fitdist(umbs_flwr_plot$julian_median, "weibull")
plot(fit.weibull)
```

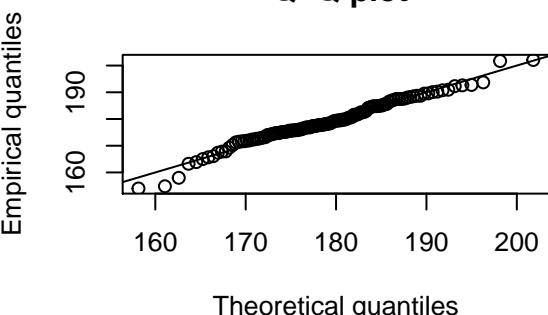
**Empirical and theoretical dens.****Q-Q plot****Empirical and theoretical CDFs****P-P plot**

```
# Lognormal distribution
fit.ln <- fitdist(umbs_flwr_plot$julian_median, "lnorm")
plot(fit.ln)
```

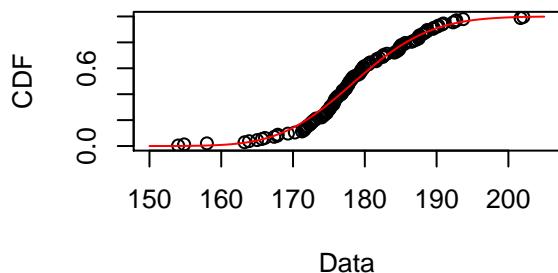
**Empirical and theoretical dens.**



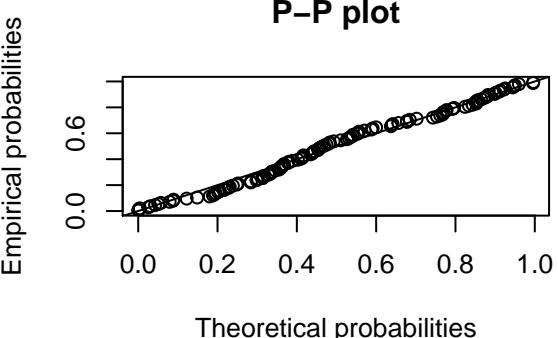
**Q-Q plot**



**Empirical and theoretical CDFs**

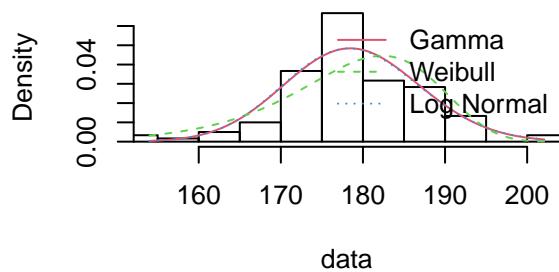


**P-P plot**

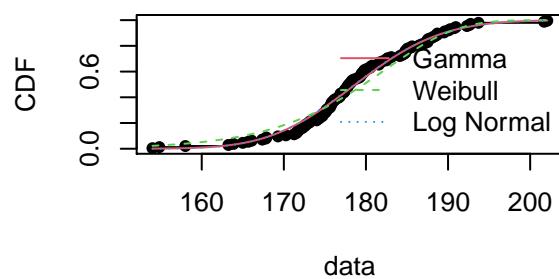


```
par(mfrow = c(2, 2))
plot.legend <- c("Gamma", "Weibull", "Log Normal")
denscomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
cdfcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
qqcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
ppcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
```

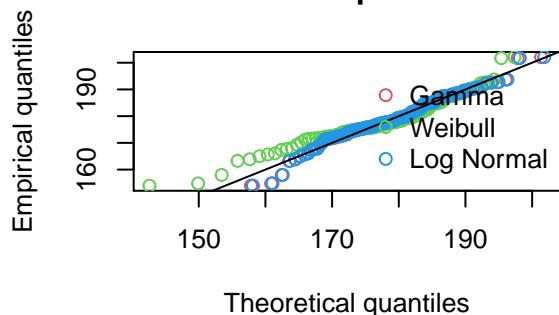
### Histogram and theoretical densities



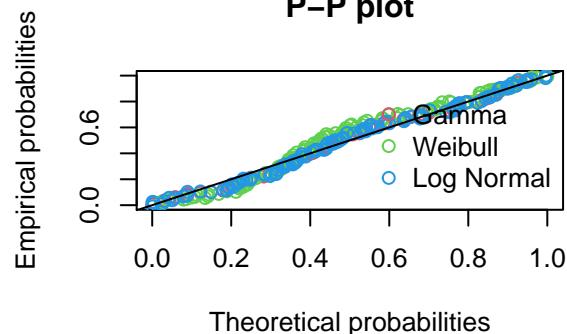
### Empirical and theoretical CDFs



### Q-Q plot



### P-P plot



```
# Goodness of fit comparisons across fits (can't include the sqrt normal bc it
# becomes diff response values)
gofstat(list(fit.gamma, fit.weibull, fit.ln), fitnames = c("Gamma", "Weibull", "Log Normal"))

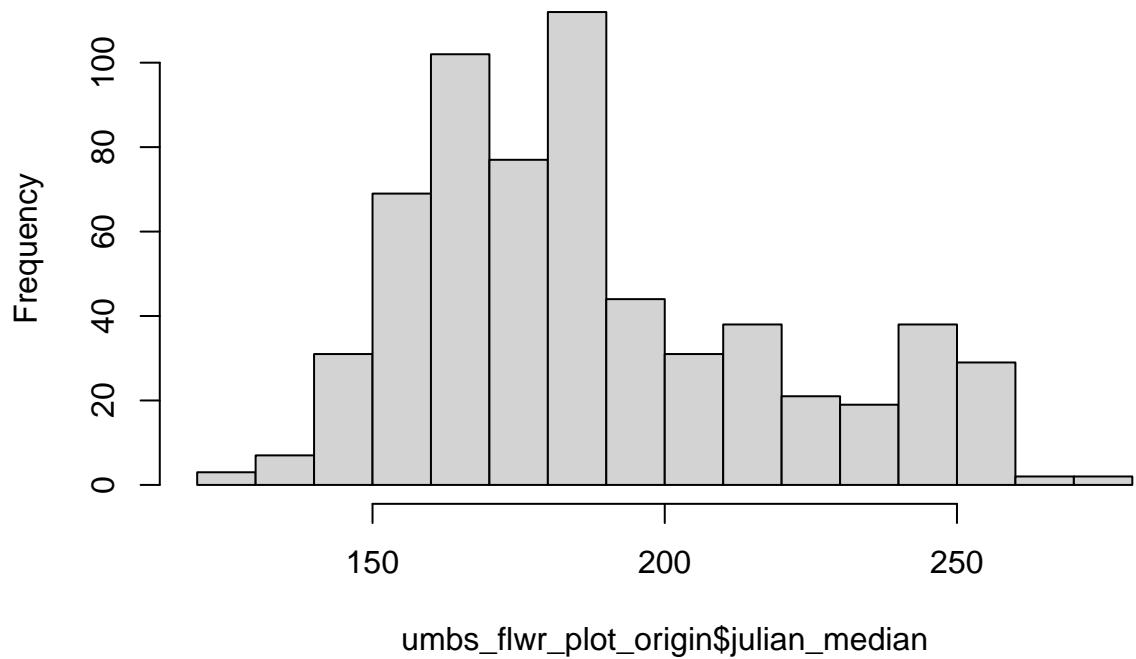
## Goodness-of-fit statistics
##                                     Gamma   Weibull Log Normal
## Kolmogorov-Smirnov statistic 0.07087173 0.1123078 0.07183673
## Cramer-von Mises statistic   0.12153497 0.3827918 0.12108114
## Anderson-Darling statistic   0.76301381 2.1930729 0.78012312
##
## Goodness-of-fit criteria
##                                     Gamma   Weibull Log Normal
## Akaike's Information Criterion 850.722 865.6707    851.238
## Bayesian Information Criterion 856.297 871.2457    856.813

# Log normal is the best fit based on this function or we could go with normal
# distribution
```

## UMBS PLOT LEVEL ORIGIN - Looking at MEDIAN JULIAN DAY

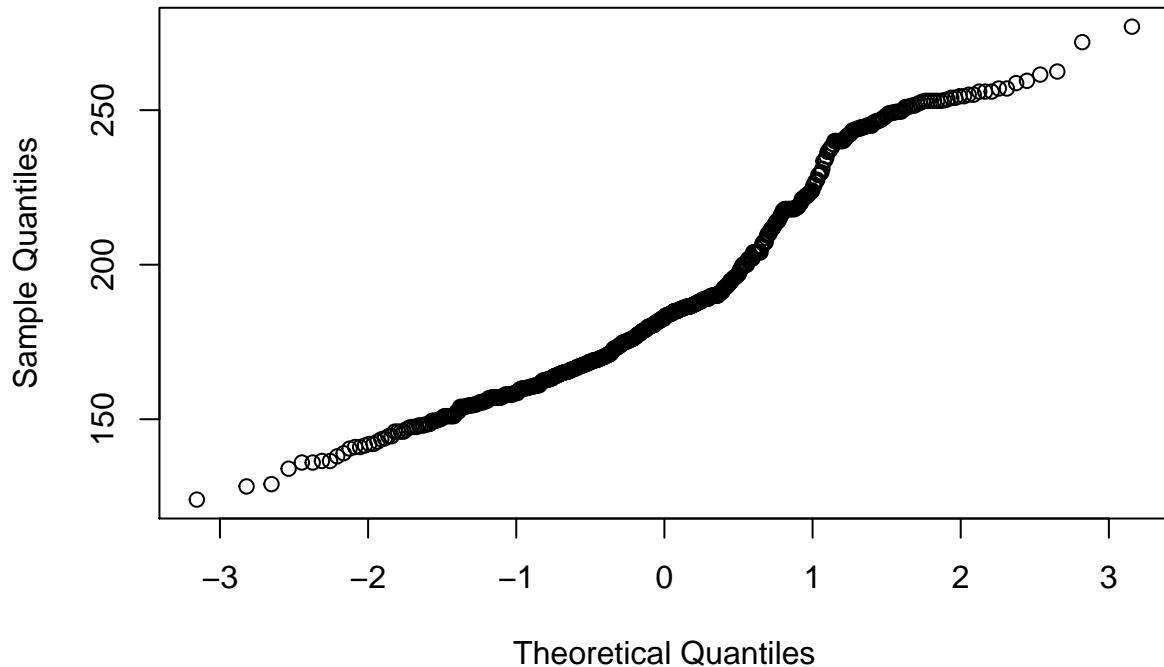
```
### UMBS ####
hist(umbs_flwr_plot_origin$julian_median)
```

### Histogram of umbs\_flwr\_plot\_origin\$julian\_median



```
qqnorm(umbs_flwr_plot_origin$julian_median)
```

### Normal Q-Q Plot



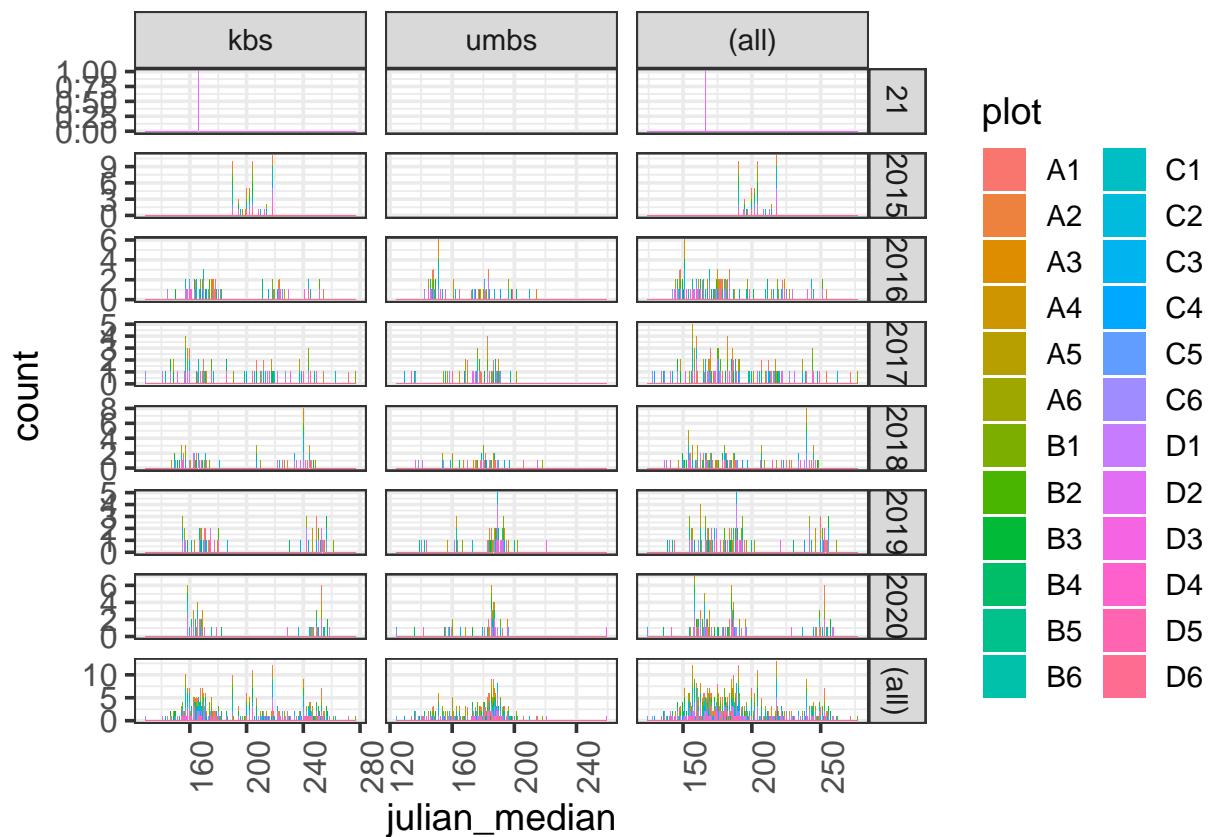
```

shapiro.test(umbs_flwr_plot_origin$julian_median) # pvalue is < 0.05 so we reject the null hypothesis

## 
## Shapiro-Wilk normality test
## 
## data: umbs_flwr_plot_origin$julian_median
## W = 0.9393, p-value = 2.808e-15

# Visualizing median Julian date for umbs at the PLOT LEVEL
ggplot(umbs_flwr_plot_origin, aes(julian_median, fill = plot)) + geom_histogram(binwidth = 0.5) +
  facet_grid(year ~ site, margins = TRUE, scales = "free")

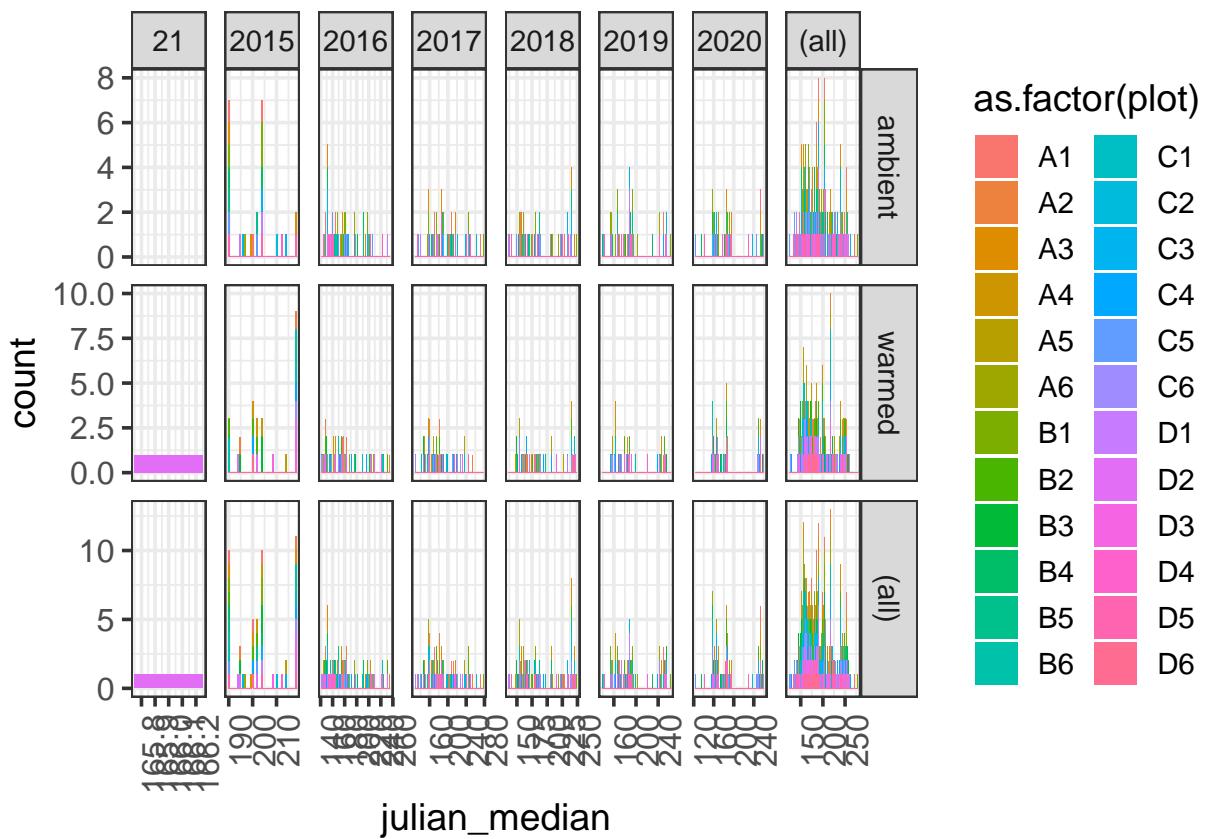
```



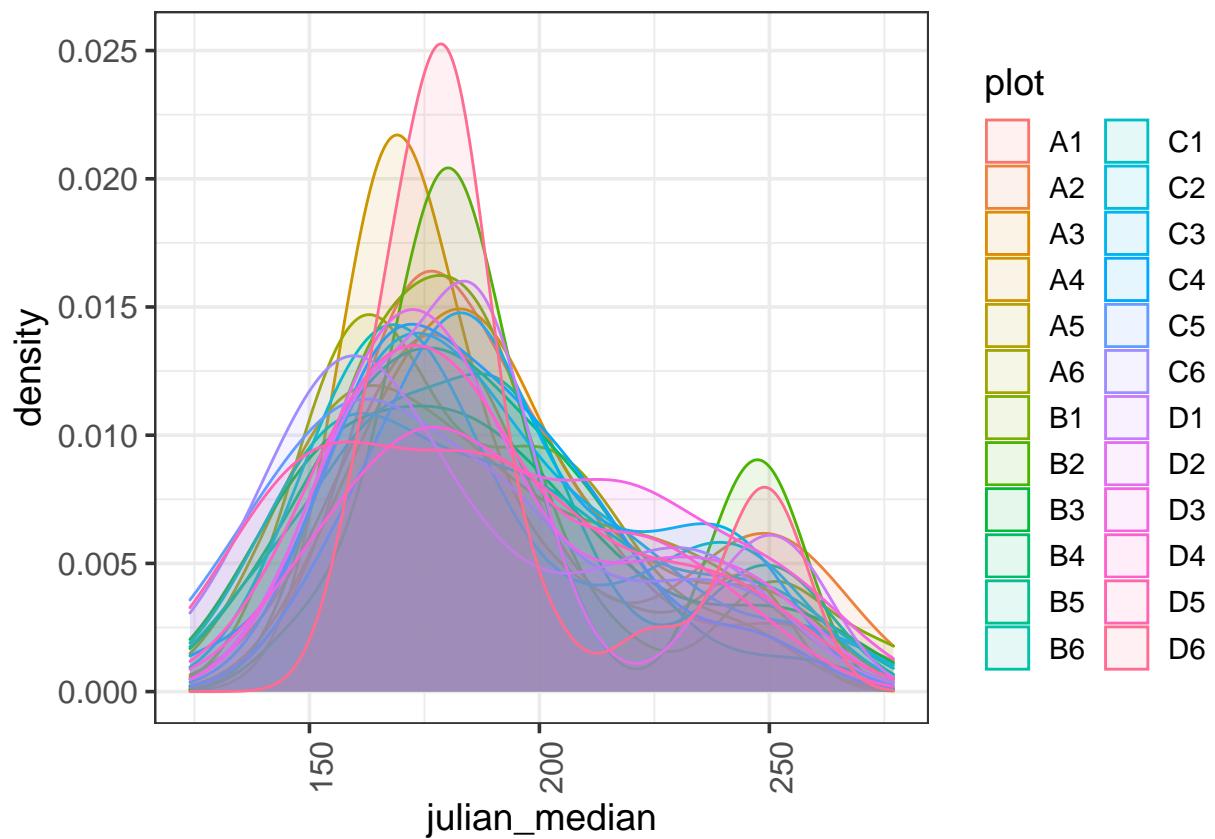
```

ggplot(umbs_flwr_plot_origin, aes(julian_median, fill = as.factor(plot))) + geom_histogram(binwidth = 0.5) +
  facet_grid(state ~ year, margins = TRUE, scales = "free")

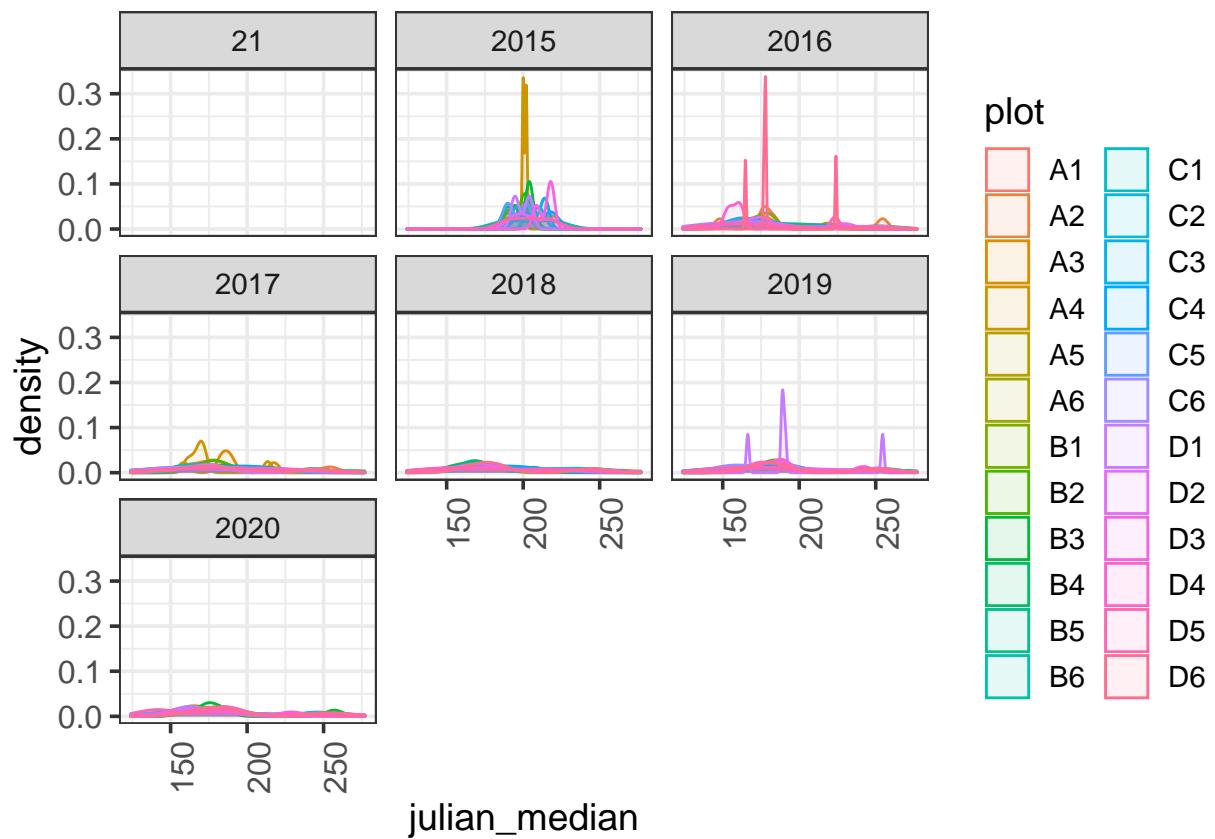
```



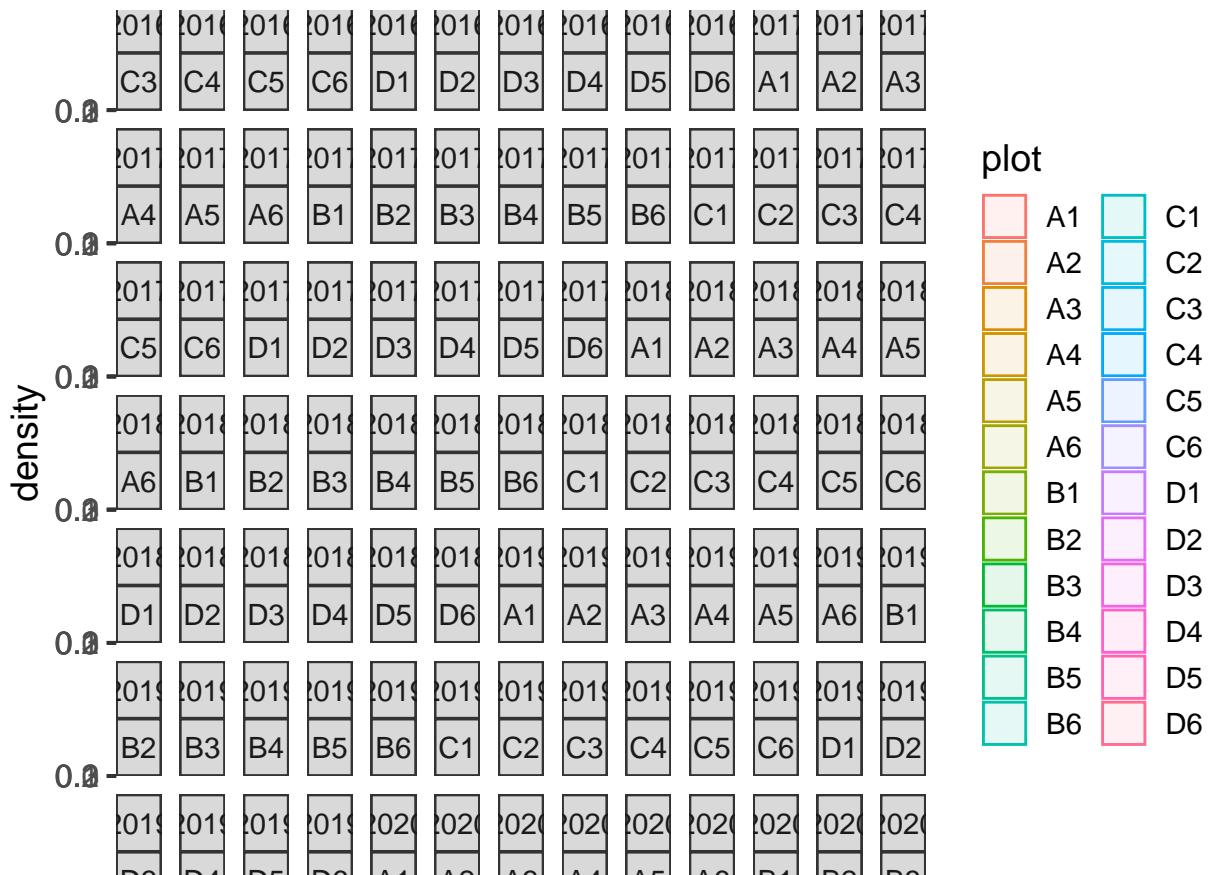
```
ggplot(umbs_flwr_plot_origin, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1)
```



```
ggplot(umbs_flwr_plot_origin, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1)  
  facet_wrap(~year)
```

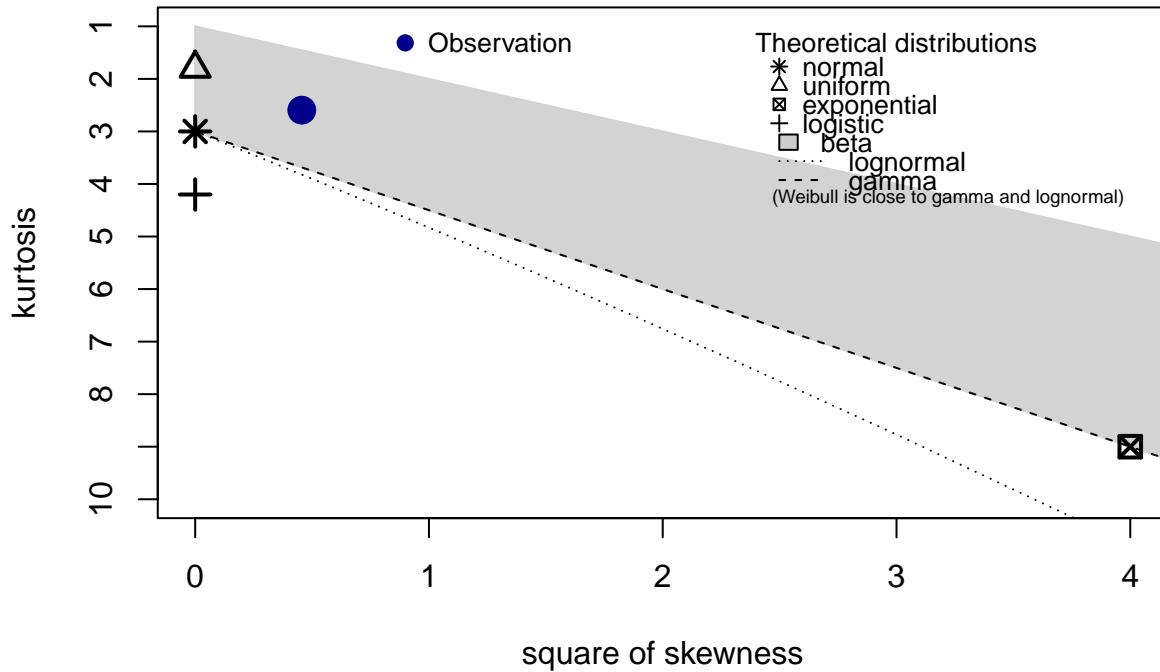


```
ggplot(umbs_flwr_plot_origin, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1)
  facet_wrap(~year + plot)
```



```
# Exploring distributions for these data:  
descdist(umbs_flwr_plot_origin$julian_median, discrete = FALSE)
```

## Cullen and Frey graph

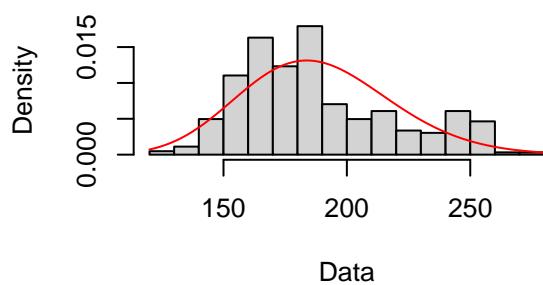
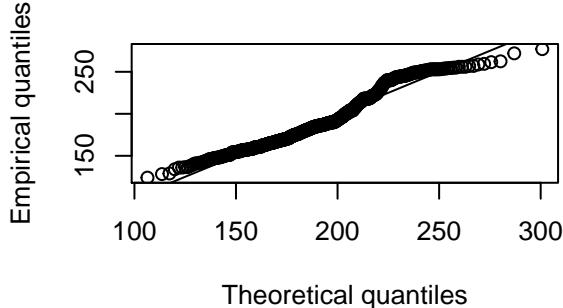
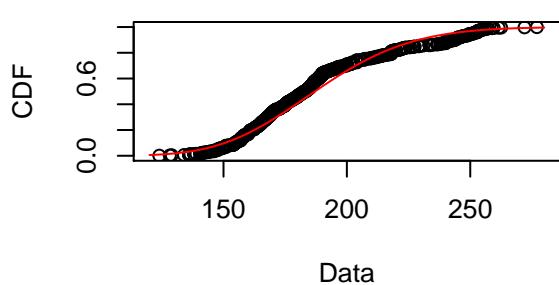
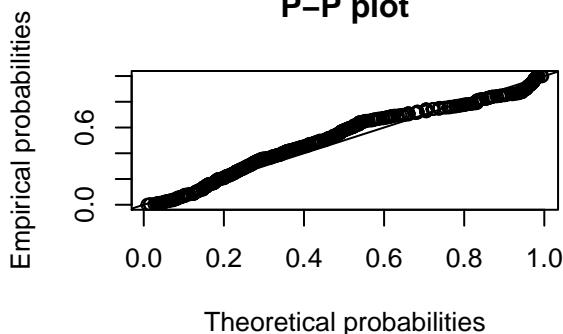


```

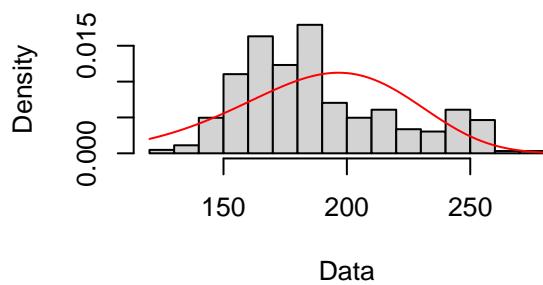
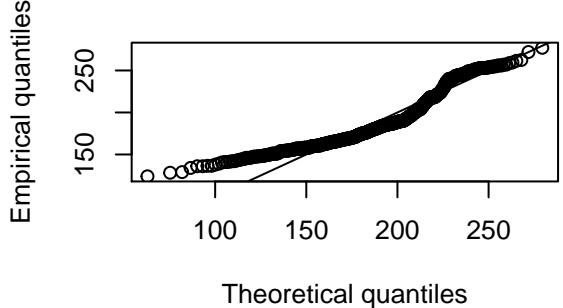
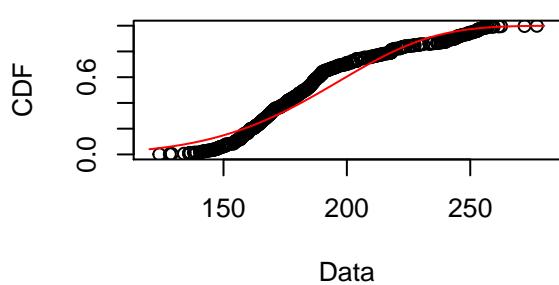
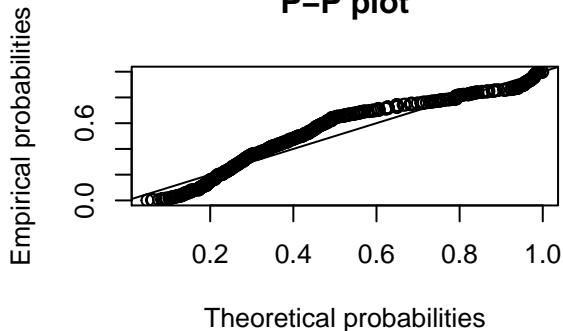
## summary statistics
## -----
## min: 124   max: 277
## median: 183.1667
## mean: 188.7282
## estimated sd: 31.51092
## estimated skewness: 0.6753548
## estimated kurtosis: 2.595054

# Gamma distribution
fit.gamma <- fitdist(umbs_flwr_plot_origin$julian_median, "gamma")
plot(fit.gamma)

```

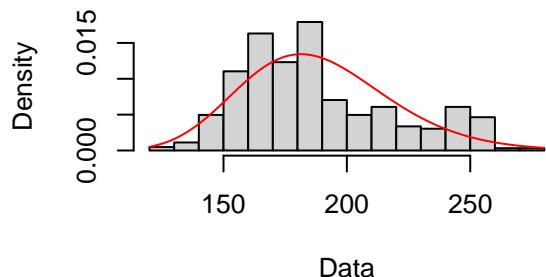
**Empirical and theoretical dens.****Q-Q plot****Empirical and theoretical CDFs****P-P plot**

```
# Weibull distribution
fit.weibull <- fitdist(umbs_flwr_plot_origin$julian_median, "weibull")
plot(fit.weibull)
```

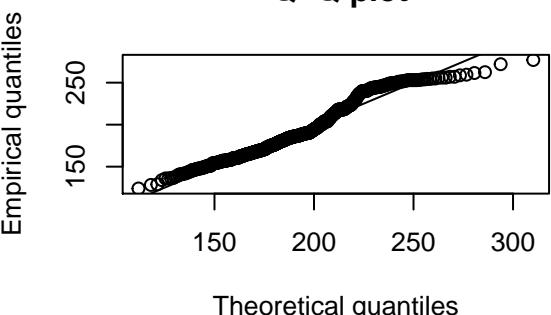
**Empirical and theoretical dens.****Q-Q plot****Empirical and theoretical CDFs****P-P plot**

```
# Lognormal distribution
fit.ln <- fitdist(umbs_flwr_plot_origin$julian_median, "lnorm")
plot(fit.ln)
```

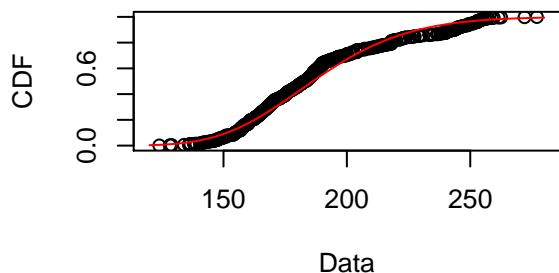
**Empirical and theoretical dens.**



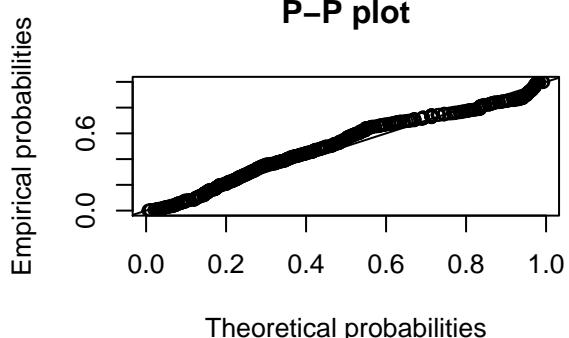
**Q–Q plot**



**Empirical and theoretical CDFs**

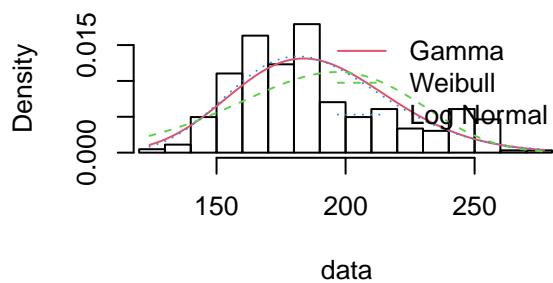


**P–P plot**

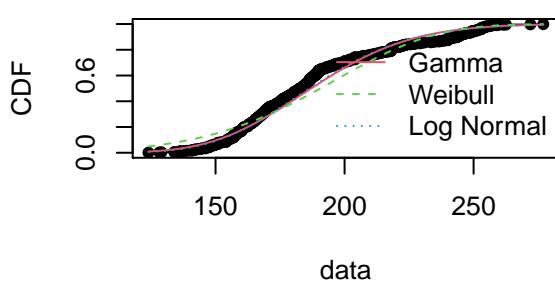


```
par(mfrow = c(2, 2))
plot.legend <- c("Gamma", "Weibull", "Log Normal")
denscomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
cdfcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
qqcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
ppcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
```

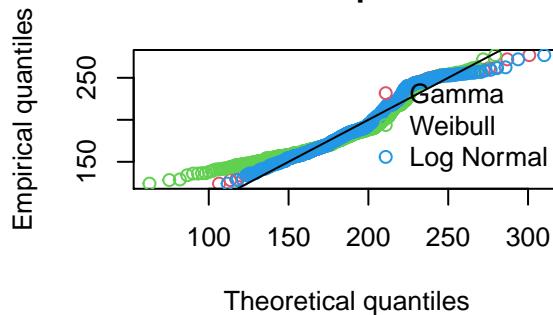
### Histogram and theoretical densities



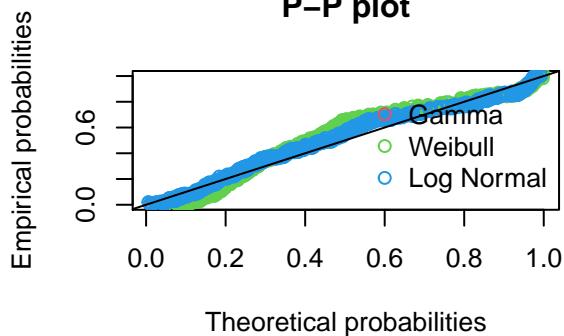
### Empirical and theoretical CDFs



### Q-Q plot



### P-P plot



```
# Goodness of fit comparisons across fits (can't include the sqrt normal bc it
# becomes diff response values)
gofstat(list(fit.gamma, fit.weibull, fit.ln), fitnames = c("Gamma", "Weibull", "Log Normal"))
```

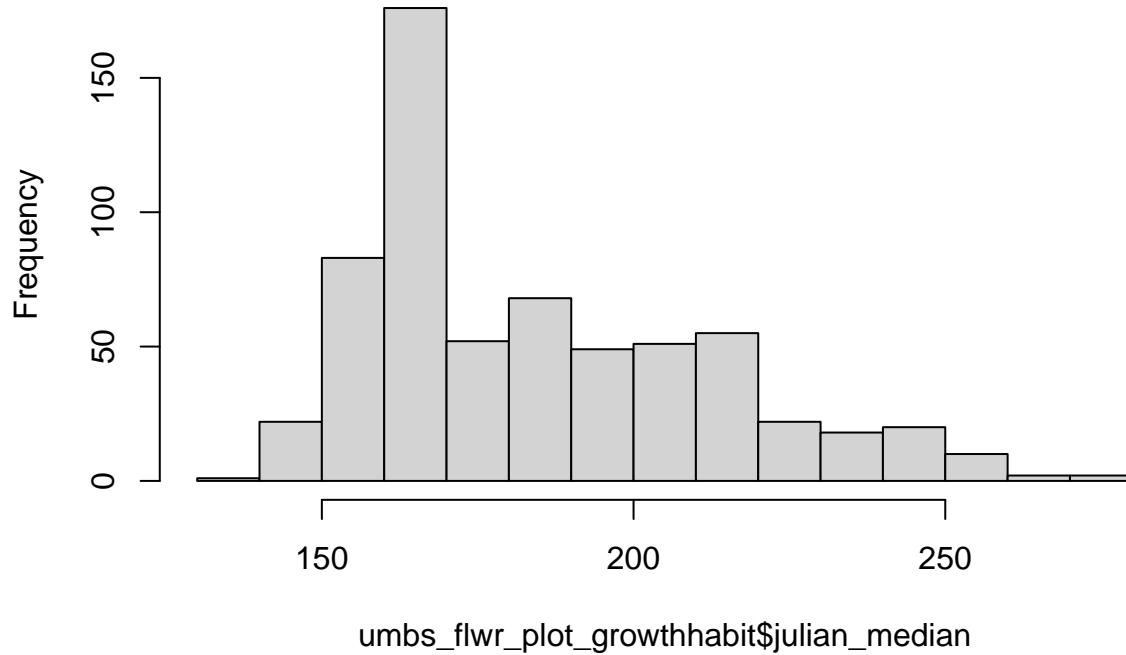
```
## Goodness-of-fit statistics
##                                     Gamma   Weibull Log Normal
## Kolmogorov-Smirnov statistic 0.1033789 0.148877 0.09228639
## Cramer-von Mises statistic   1.4472267 3.289711 1.12667727
## Anderson-Darling statistic   9.1394824 19.530457 7.30206746
##
## Goodness-of-fit criteria
##                                     Gamma   Weibull Log Normal
## Akaike's Information Criterion 6048.141 6165.178 6033.459
## Bayesian Information Criterion 6057.017 6174.053 6042.334
```

*# log normal looks like it's the best fit*

## UMBS PLOT LEVEL GROWTH HABIT - Looking at MEDIAN JULIAN DAY

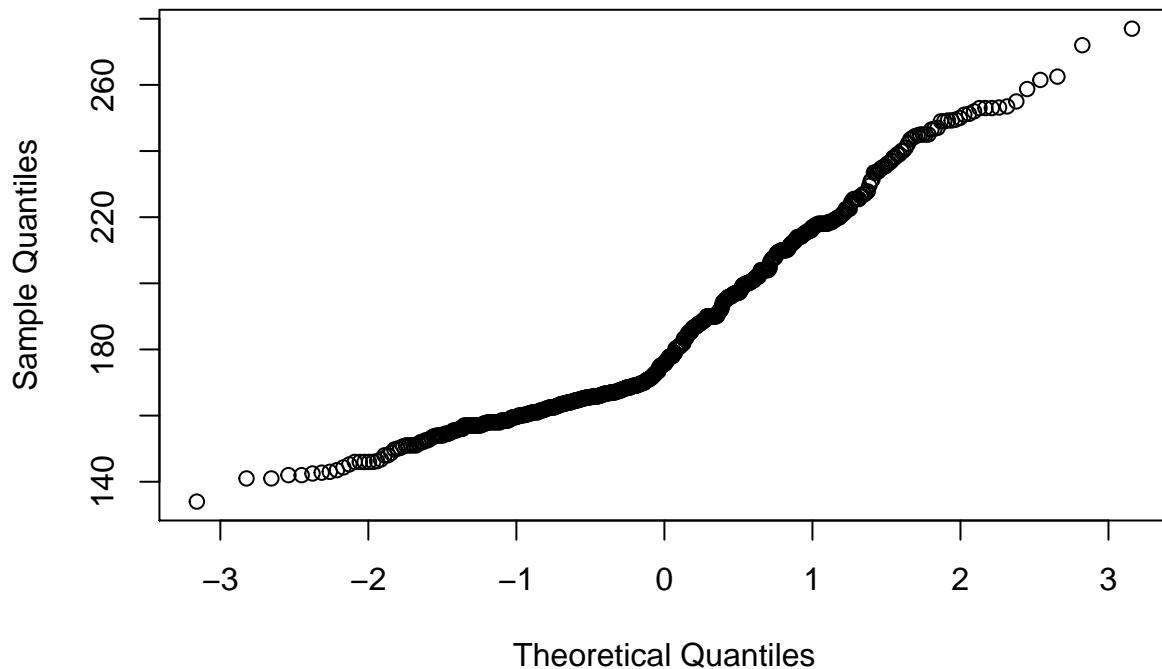
```
### UMBS ####
hist(umbs_flwr_plot_growthhabit$julian_median)
```

### Histogram of umbs\_flwr\_plot\_growthhabit\$julian\_median



```
qqnorm(umb..._flwr..._plot..._growthhabit$julian..._median)
```

### Normal Q-Q Plot



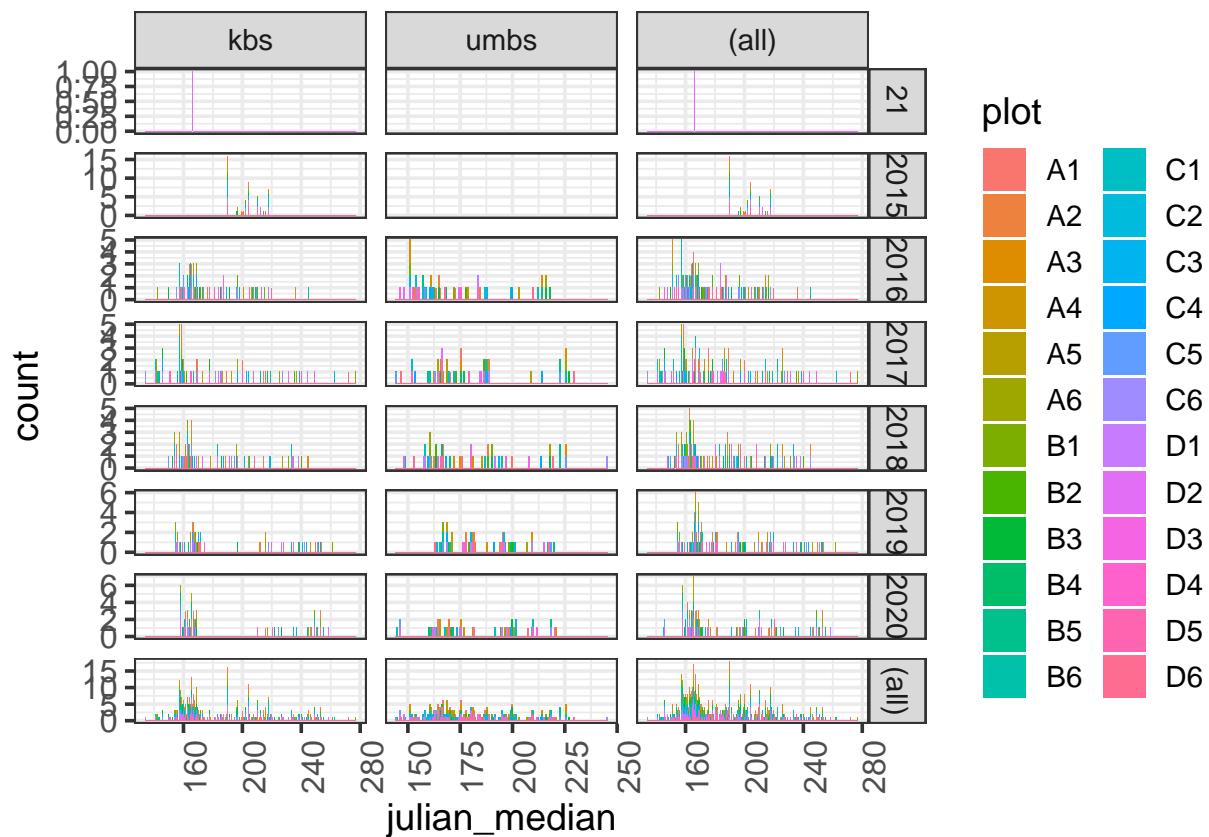
```

shapiro.test(umbs_flwr_plot_growthhabit$julian_median) # pvalue is < 0.05 so we reject the null hypothesis

## 
## Shapiro-Wilk normality test
## 
## data: umbs_flwr_plot_growthhabit$julian_median
## W = 0.92629, p-value < 2.2e-16

# Visualizing median Julian date for umbs at the PLOT LEVEL
ggplot(umbs_flwr_plot_growthhabit, aes(julian_median, fill = plot)) + geom_histogram(binwidth = 0.5) +
  facet_grid(year ~ site, margins = TRUE, scales = "free")

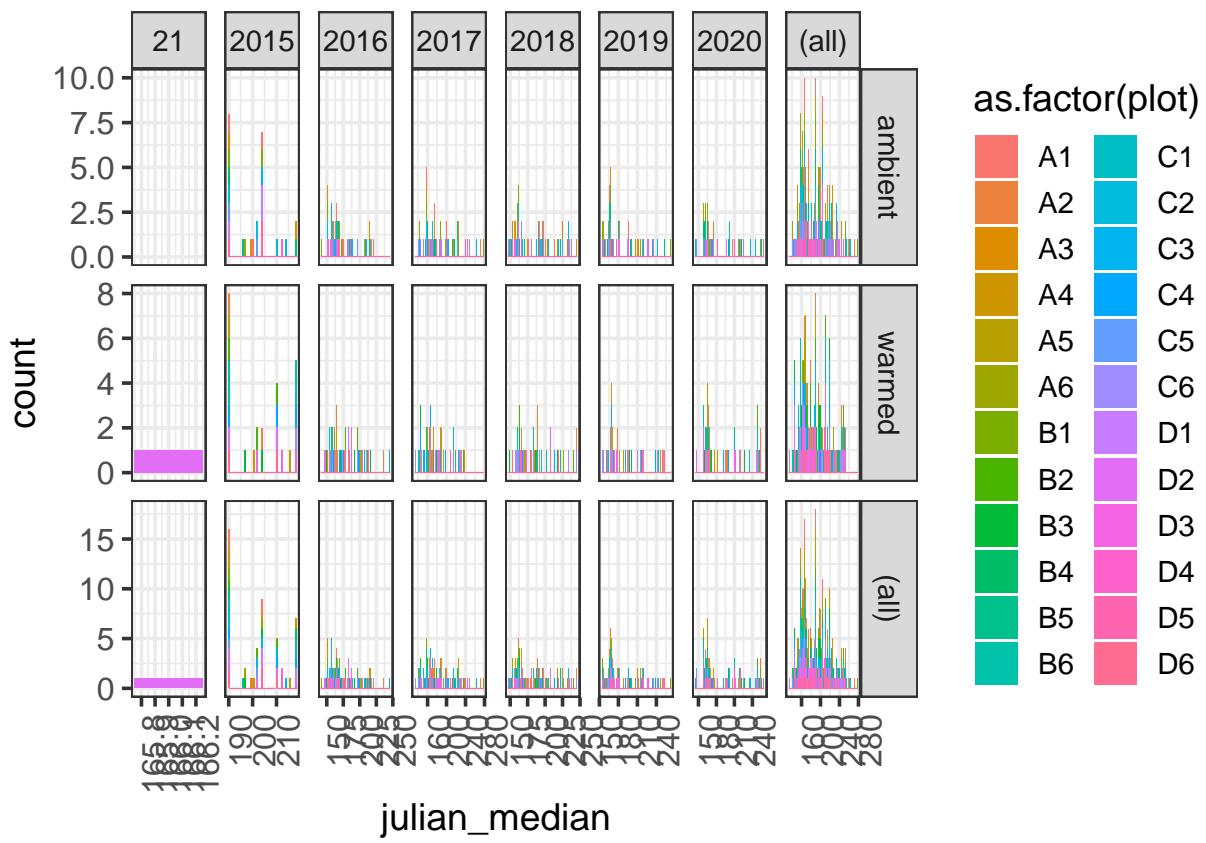
```



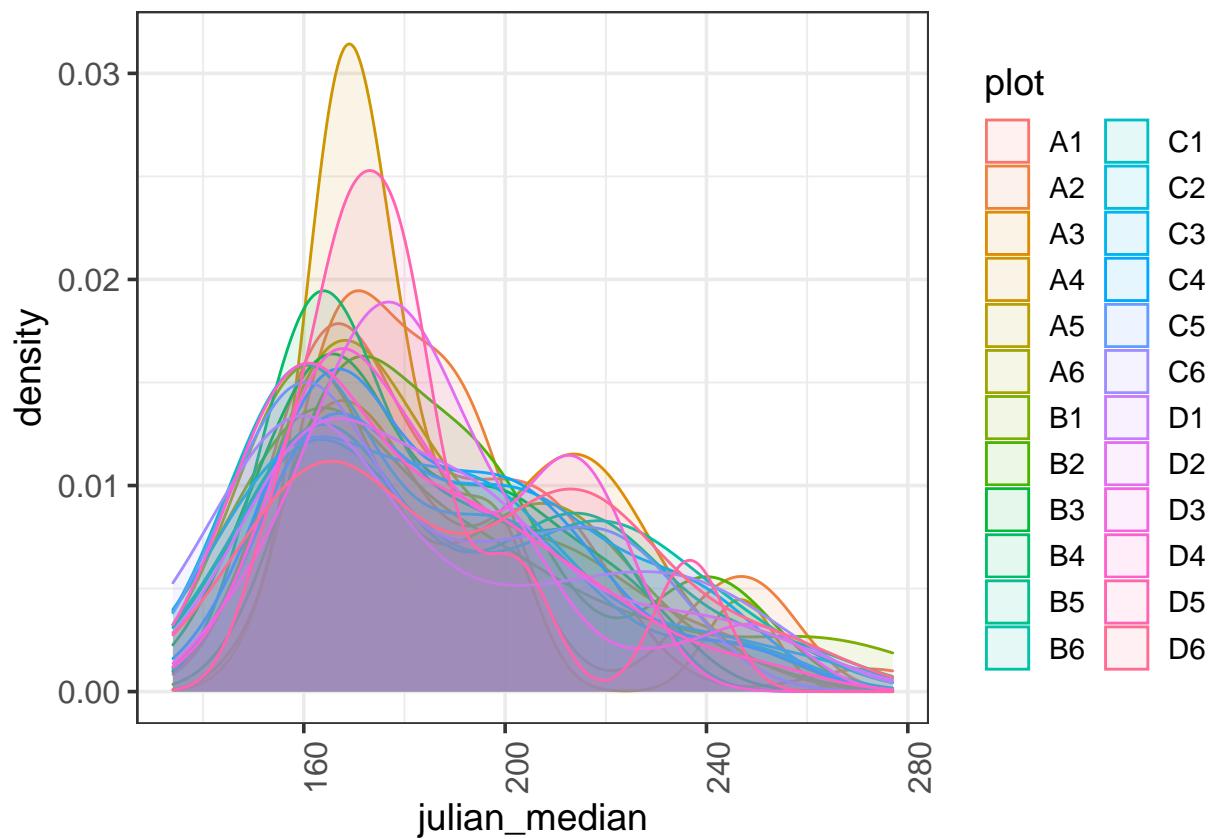
```

ggplot(umbs_flwr_plot_growthhabit, aes(julian_median, fill = as.factor(plot))) +
  geom_histogram(binwidth = 0.5) + facet_grid(state ~ year, margins = TRUE, scales = "free")

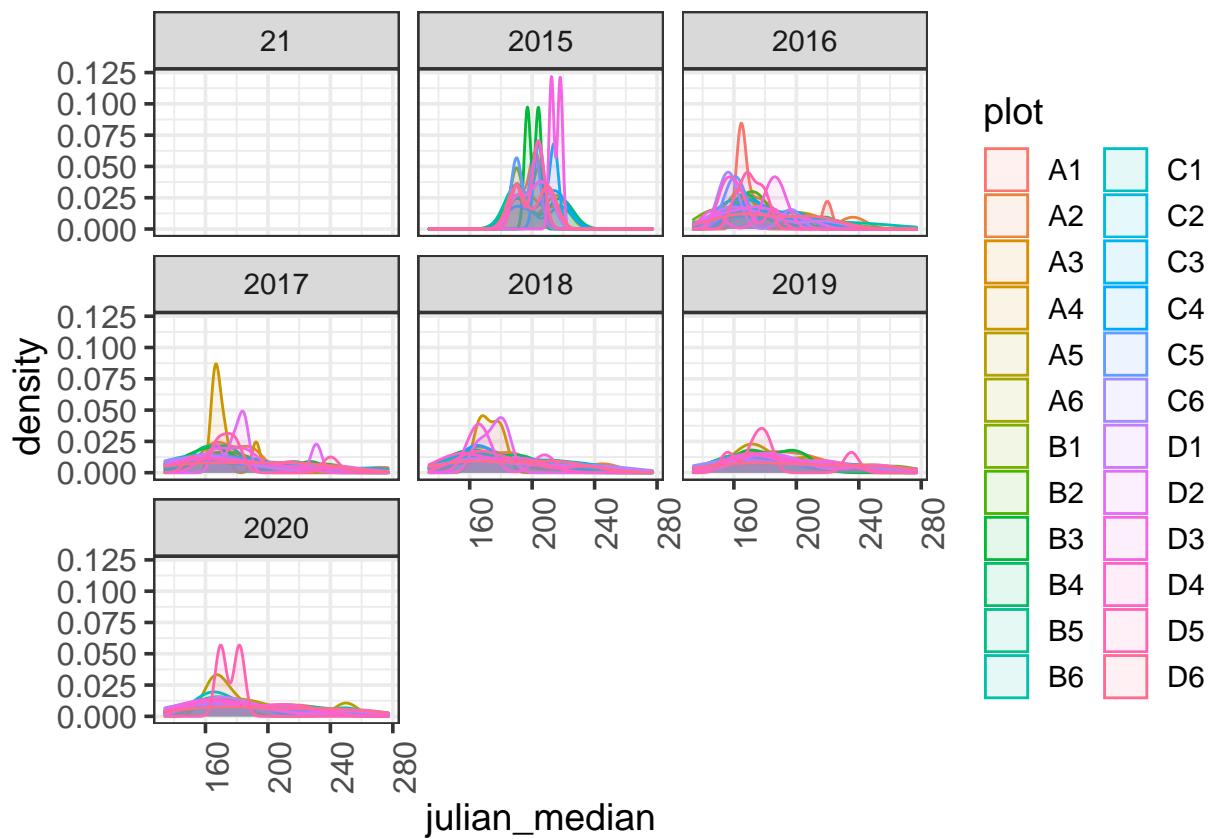
```



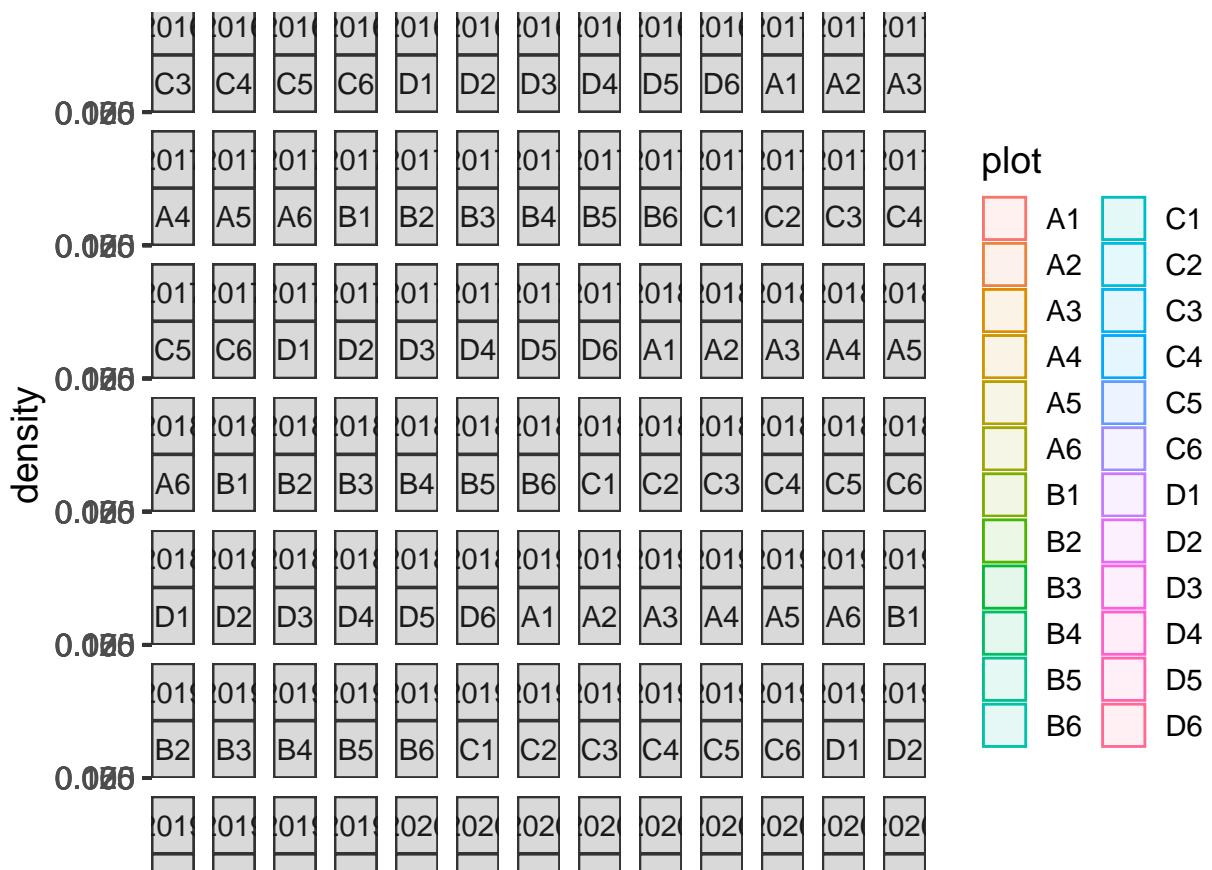
```
ggplot(umbs_flwr_plot_growthhabit, aes(julian_median, fill = plot, color = plot)) +
  geom_density(alpha = 0.1)
```



```
ggplot(umbs_flwr_plot_growthhabit, aes(julian_median, fill = plot, color = plot)) +  
  geom_density(alpha = 0.1) + facet_wrap(~year)
```

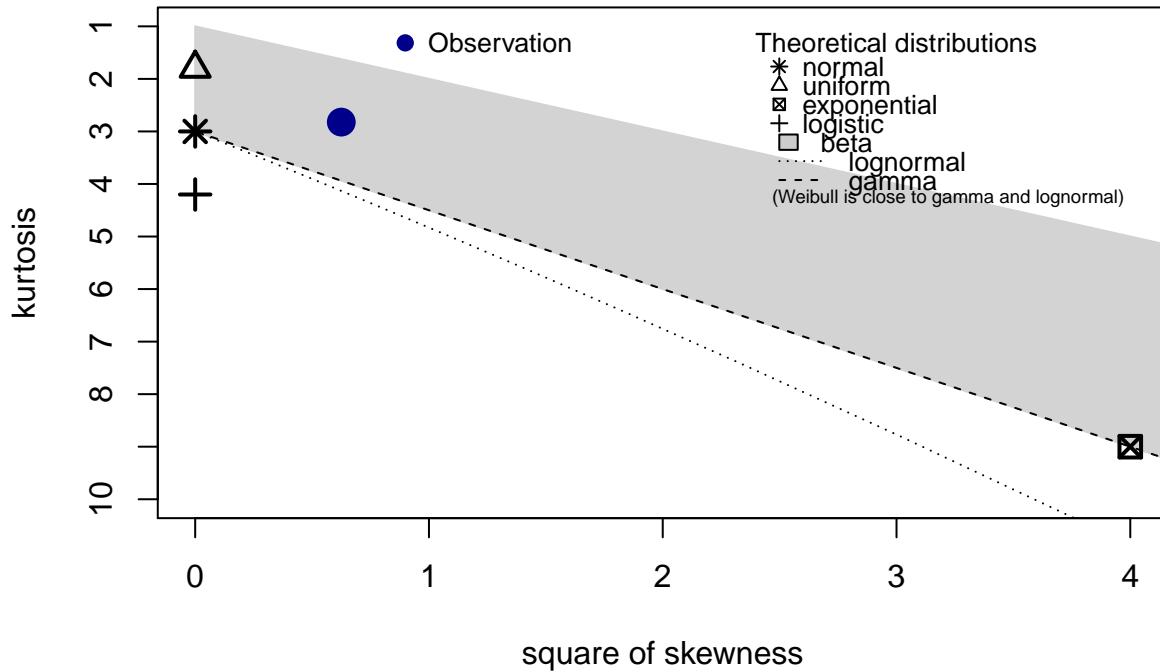


```
ggplot(umbs_flwr_plot_growthhabit, aes(julian_median, fill = plot, color = plot)) +
  geom_density(alpha = 0.1) + facet_wrap(~year + plot)
```



```
# Exploring distributions for these data:  
descdist(umbs_flwr_plot_growthhabit$julian_median, discrete = FALSE)
```

## Cullen and Frey graph



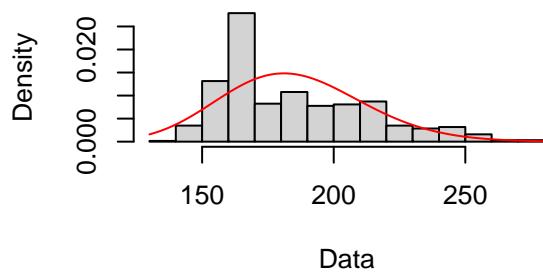
```

## summary statistics
## -----
## min: 134   max: 277
## median: 175.6667
## mean: 185.0703
## estimated sd: 27.98008
## estimated skewness: 0.7906147
## estimated kurtosis: 2.822453

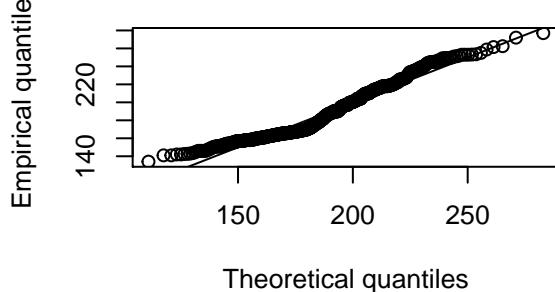
# Gamma distribution
fit.gamma <- fitdist(umbs_flwr_plot_growthhabit$julian_median, "gamma")
plot(fit.gamma)

```

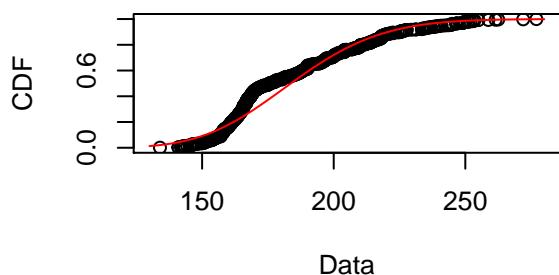
**Empirical and theoretical dens.**



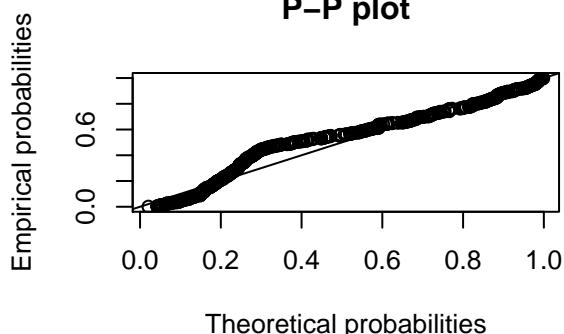
**Q-Q plot**



**Empirical and theoretical CDFs**

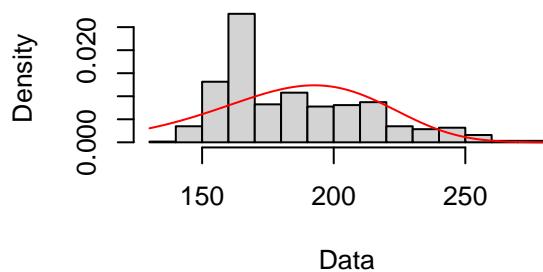


**P-P plot**

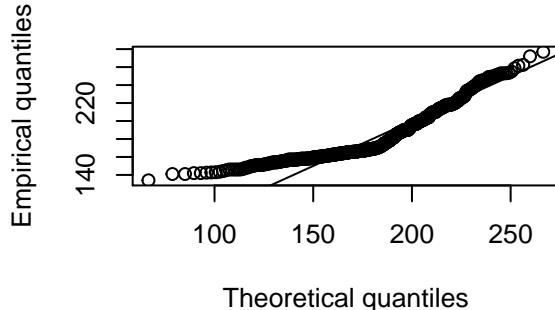


```
# Weibull distribution
fit.weibull <- fitdist(umbs_flwr_plot_growthhabit$julian_median, "weibull")
plot(fit.weibull)
```

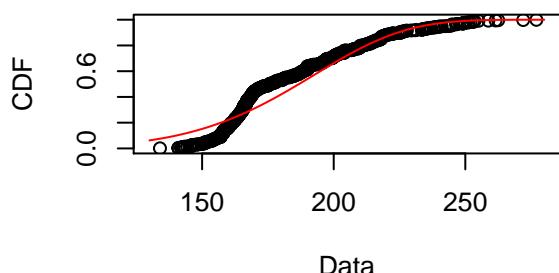
**Empirical and theoretical dens.**



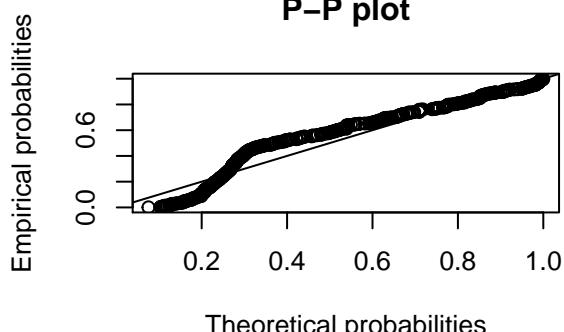
**Q-Q plot**



**Empirical and theoretical CDFs**

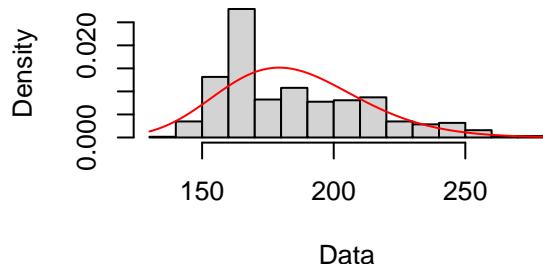


**P-P plot**

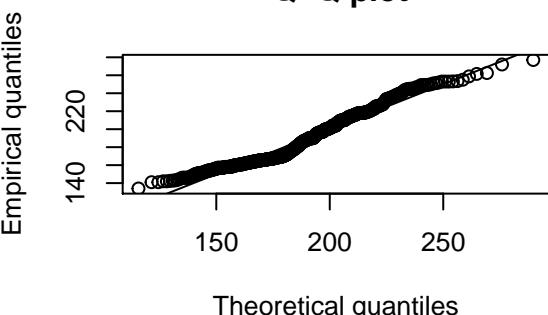


```
# Lognormal distribution
fit.ln <- fitdist(umbs_flwr_plot_growthhabit$julian_median, "lnorm")
plot(fit.ln)
```

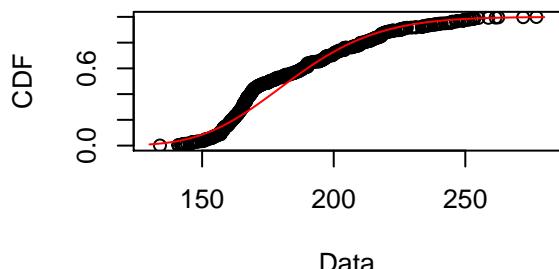
**Empirical and theoretical dens.**



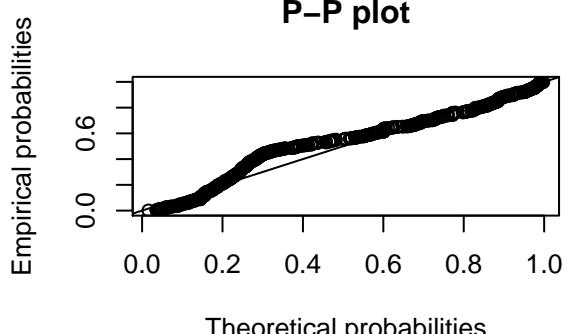
**Q-Q plot**



**Empirical and theoretical CDFs**

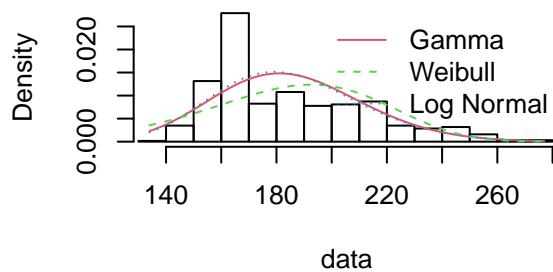


**P-P plot**

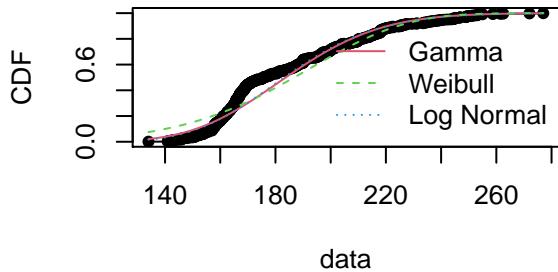


```
par(mfrow = c(2, 2))
plot.legend <- c("Gamma", "Weibull", "Log Normal")
denscomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
cdfcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
qqcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
ppcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
```

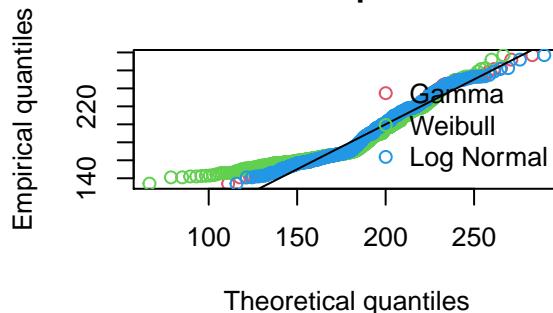
### Histogram and theoretical densities



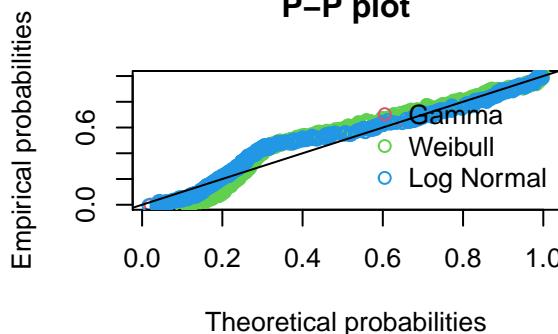
### Empirical and theoretical CDFs



### Q-Q plot



### P-P plot



```
# Goodness of fit comparisons across fits (can't include the sqrt normal bc it
# becomes diff response values)
gofstat(list(fit.gamma, fit.weibull, fit.ln), fitnames = c("Gamma", "Weibull", "Log Normal"))
```

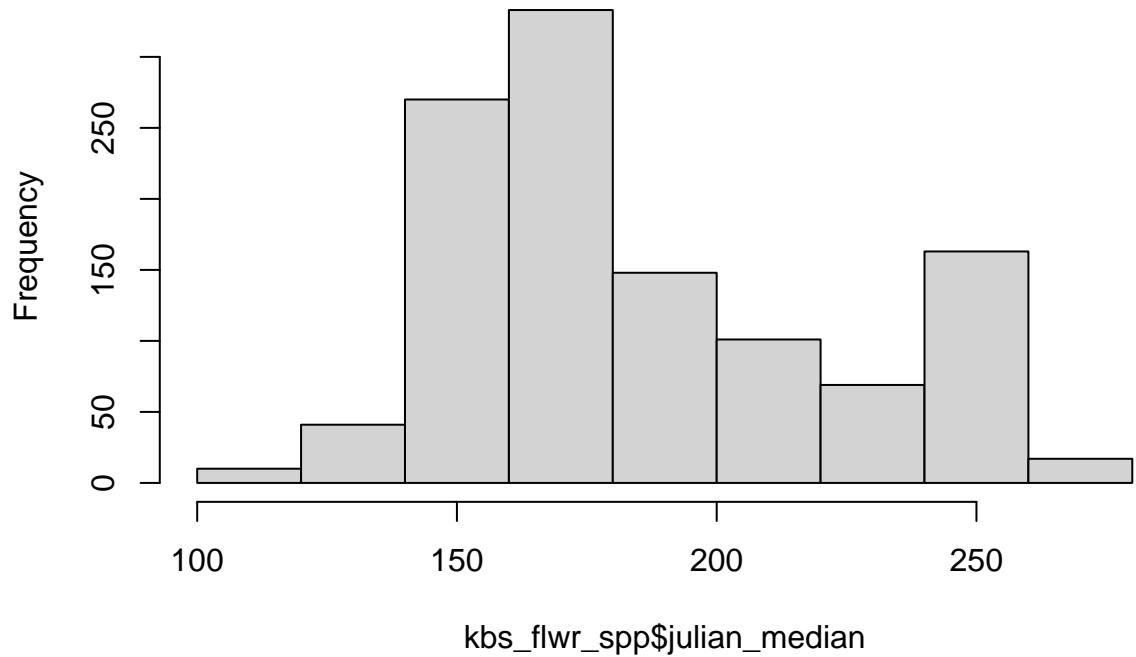
```
## Goodness-of-fit statistics
##                                     Gamma   Weibull Log Normal
## Kolmogorov-Smirnov statistic 0.1457577 0.1392539 0.1420043
## Cramer-von Mises statistic  2.4904042 3.5666090 2.2843640
## Anderson-Darling statistic 13.8740882 21.4965609 12.5741740
##
## Goodness-of-fit criteria
##                                     Gamma   Weibull Log Normal
## Akaike's Information Criterion 5952.729 6095.992 5934.791
## Bayesian Information Criterion 5961.624 6104.887 5943.686
```

```
# Log normal looks like it's the best fit
```

## KBS SPECIES LEVEL - Looking at MEDIAN JULIAN DAY

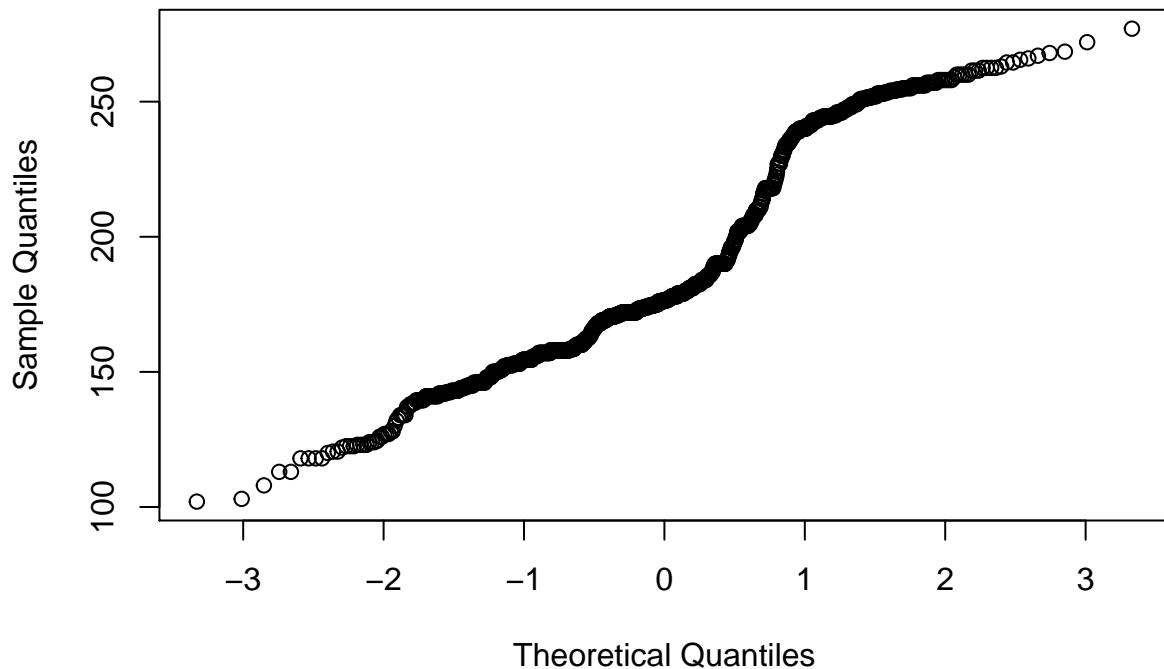
```
### KBS ####
hist(kbs_flwr_spp$julian_median)
```

### Histogram of kbs\_flwr\_spp\$julian\_median



```
qqnorm(kbs_flwr_spp$julian_median)
```

### Normal Q-Q Plot



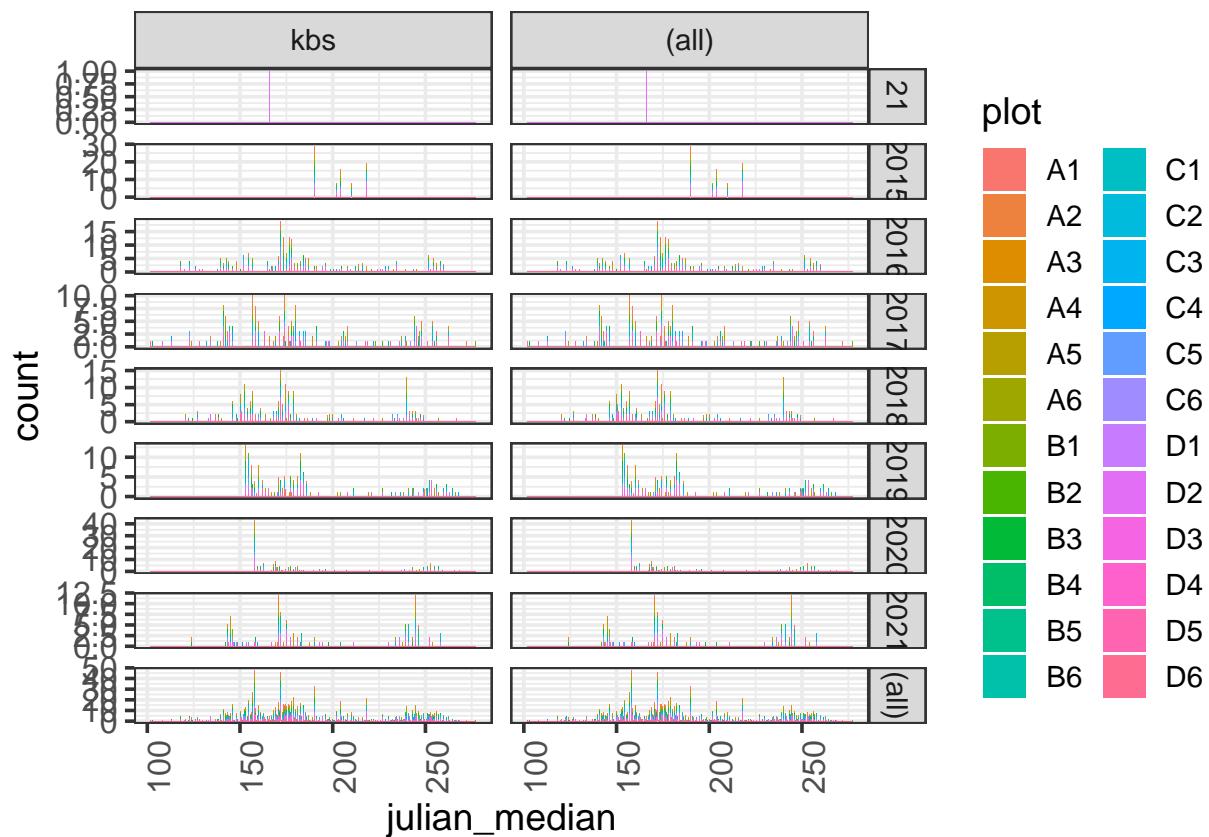
```

shapiro.test(kbs_flwr_spp$julian_median) # pvalue is < 0.05 so we reject the null hypothesis that the

##
## Shapiro-Wilk normality test
##
## data: kbs_flwr_spp$julian_median
## W = 0.92791, p-value < 2.2e-16

# Visualizing median Julian date for kbs at the SPECIES LEVEL
ggplot(kbs_flwr_spp, aes(julian_median, fill = plot)) + geom_histogram(binwidth = 0.5) +
  facet_grid(year ~ site, margins = TRUE, scales = "free")

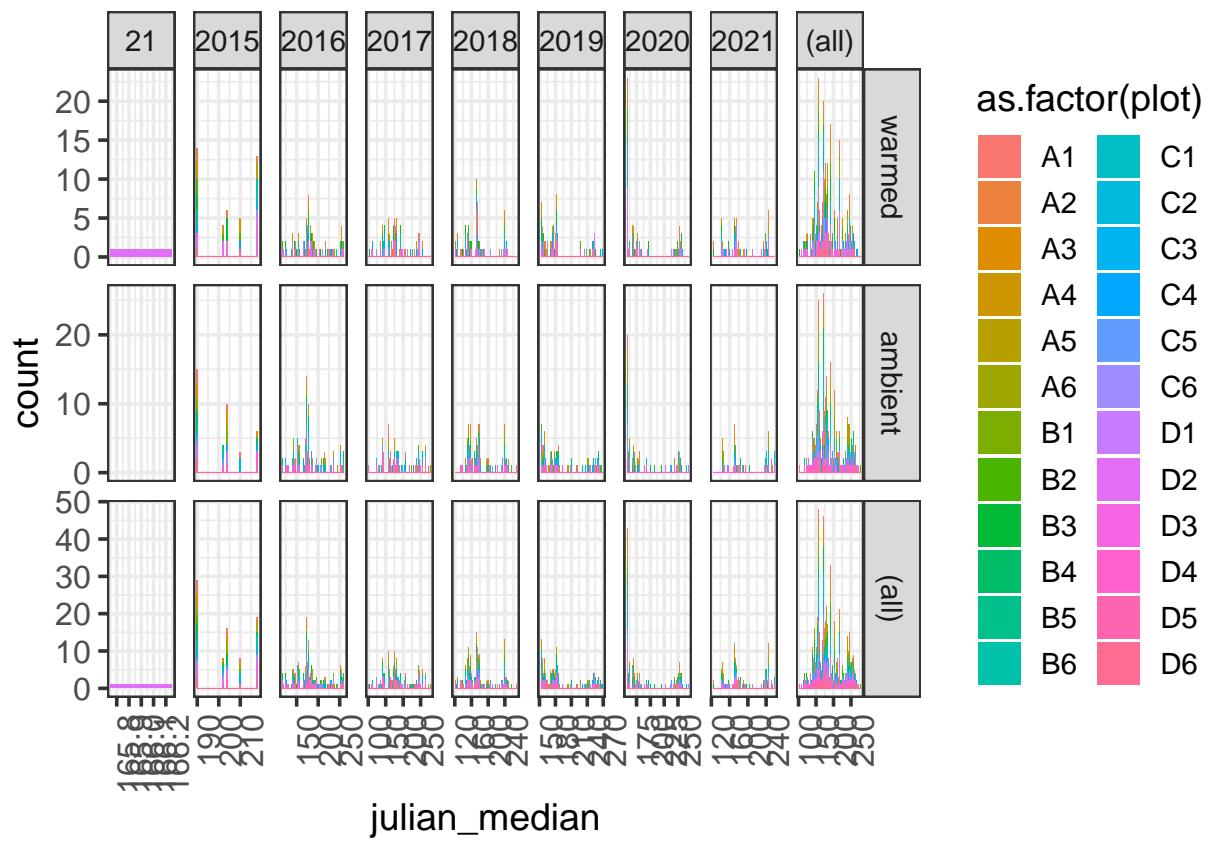
```



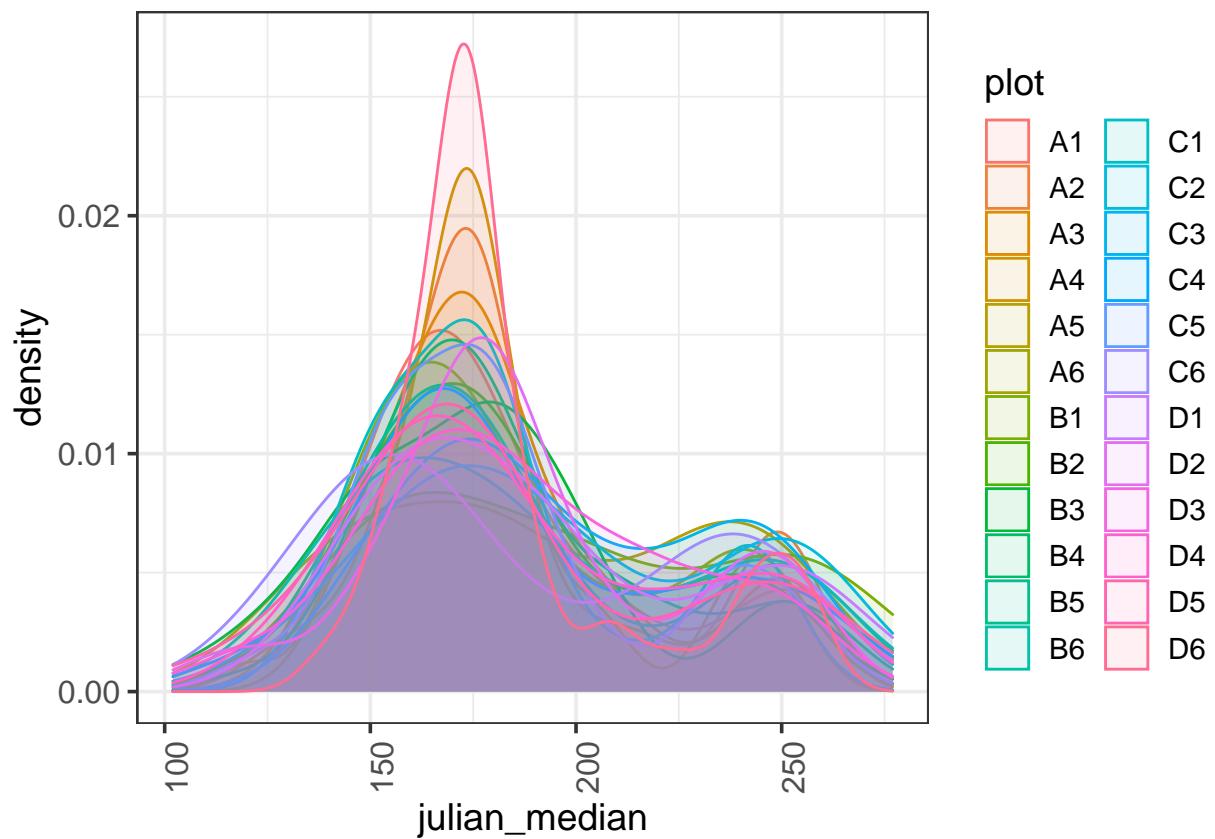
```

ggplot(kbs_flwr_spp, aes(julian_median, fill = as.factor(plot))) + geom_histogram(binwidth = 0.5) +
  facet_grid(state ~ year, margins = TRUE, scales = "free")

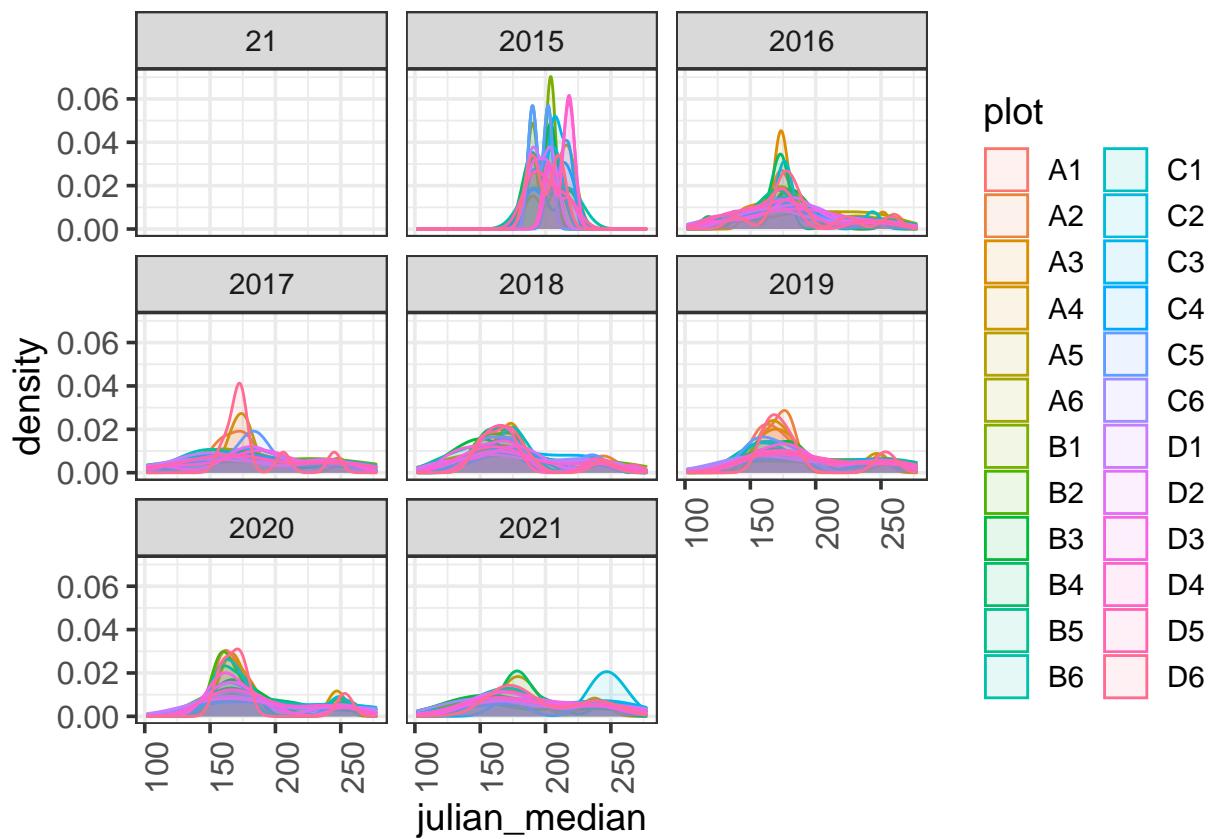
```



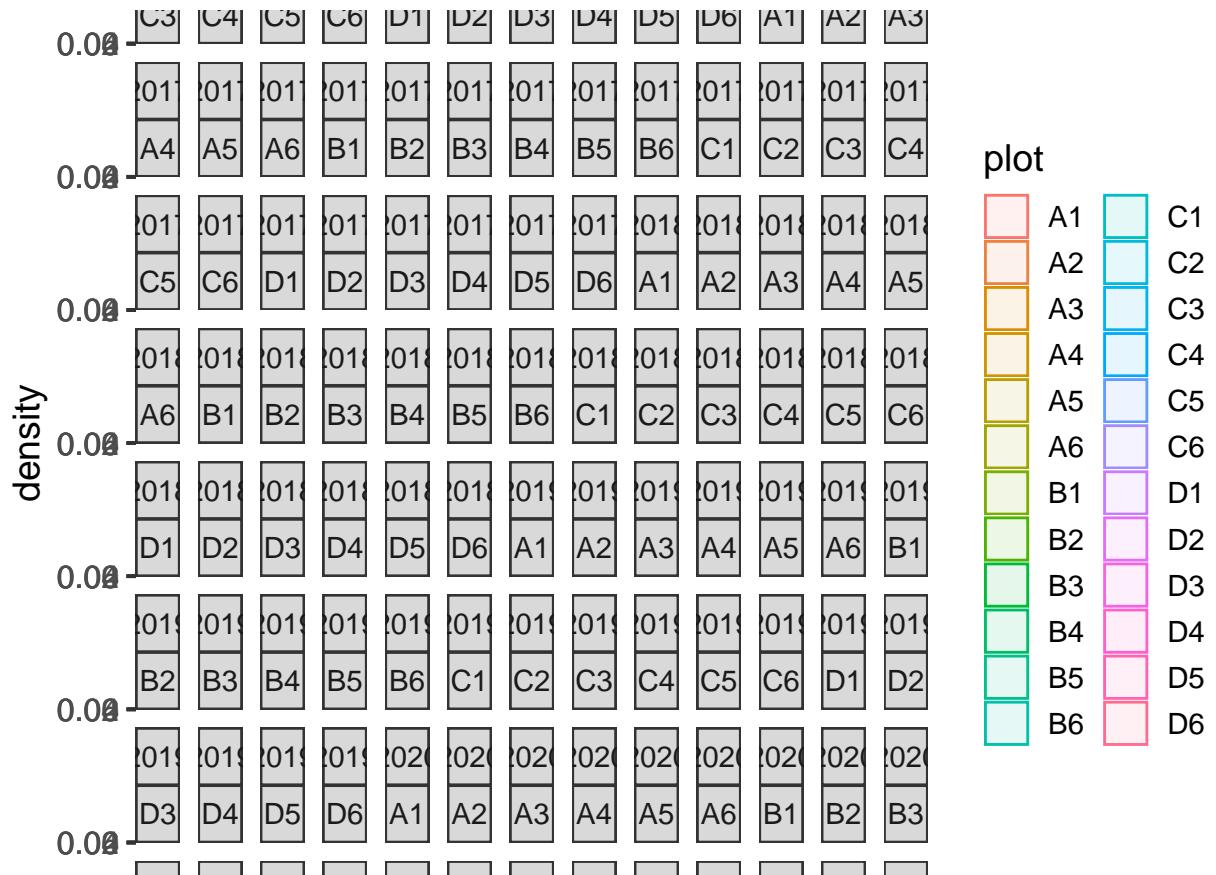
```
ggplot(kbs_flwr_spp, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1)
```



```
ggplot(kbs_flwr_spp, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1) +
  facet_wrap(~year)
```

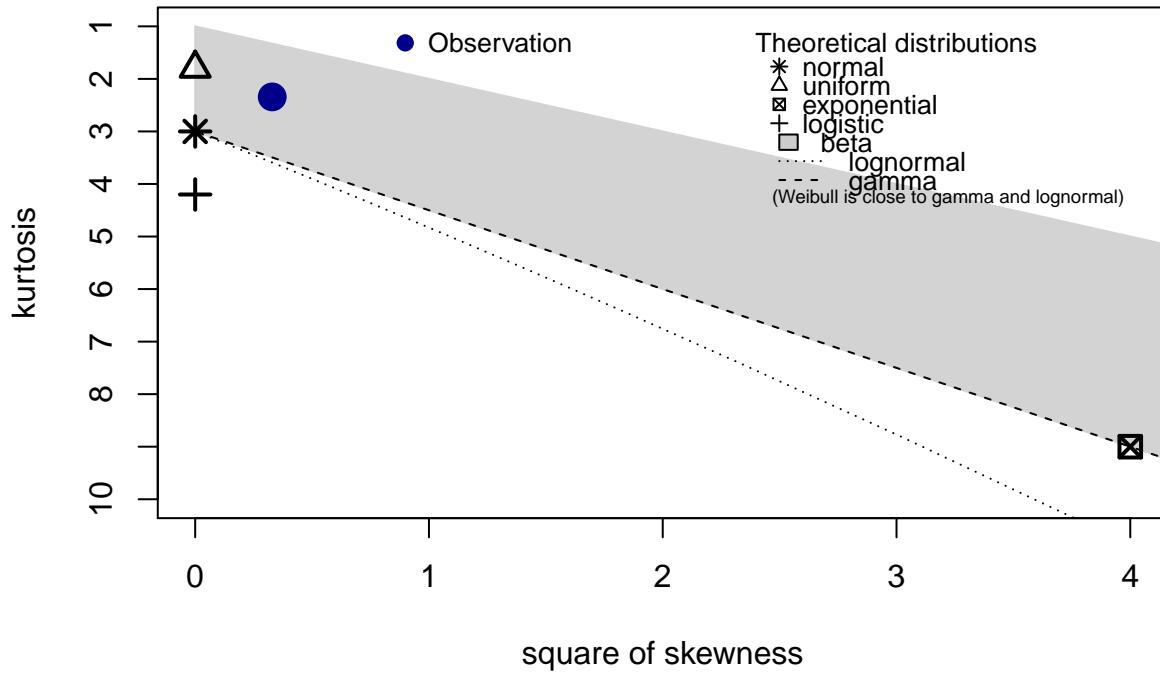


```
ggplot(kbs_flwr_spp, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1) +
  facet_wrap(~year + plot)
```



```
# Exploring distributions for these right-skewed data:  
descdist(kbs_flwr_spp$julian_median, discrete = FALSE)
```

## Cullen and Frey graph

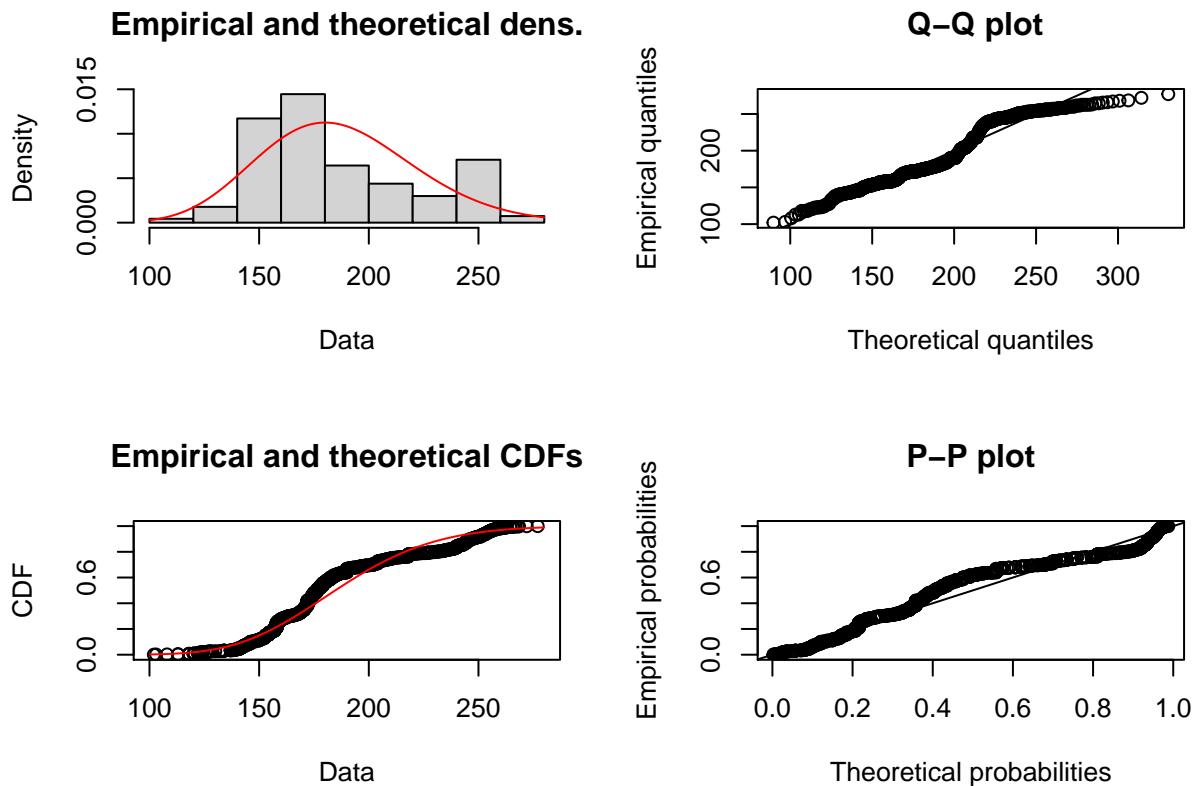


```

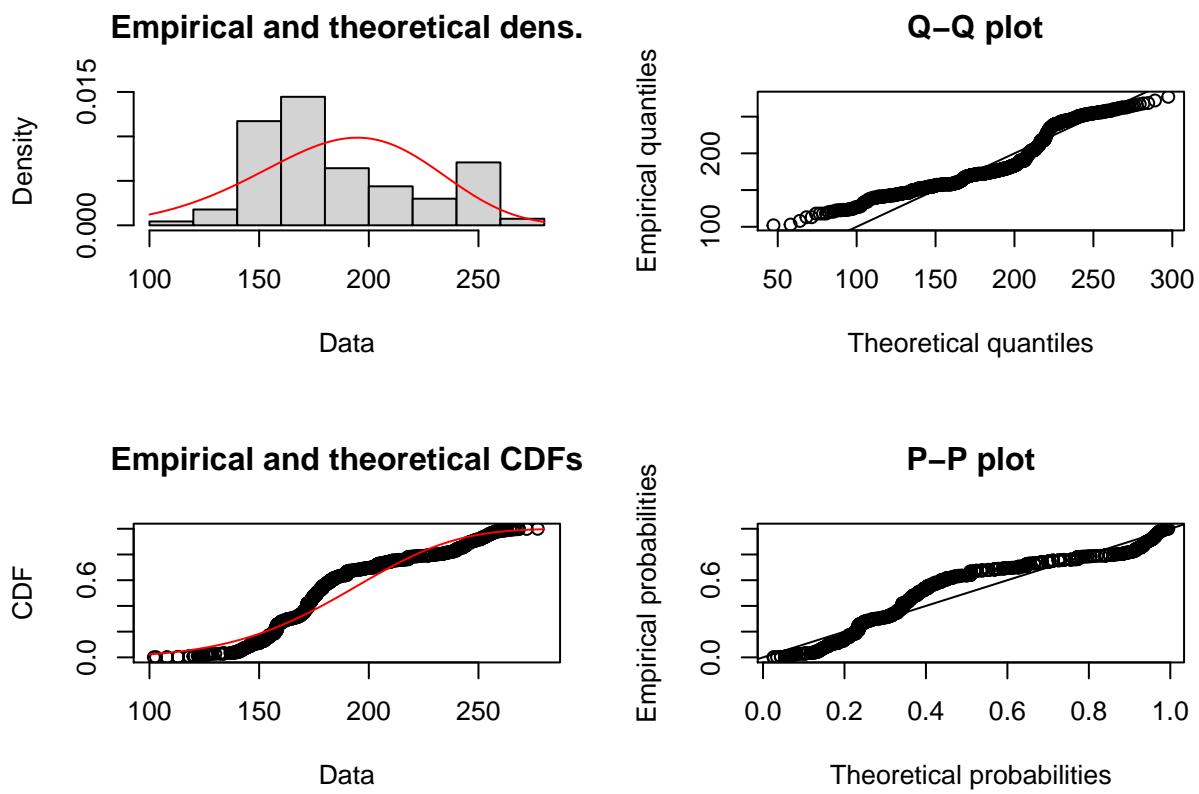
## summary statistics
## -----
## min: 102   max: 277
## median: 176.5
## mean: 187.0304
## estimated sd: 36.75702
## estimated skewness: 0.5740771
## estimated kurtosis: 2.342844

# Gamma distribution
fit.gamma <- fitdist(kbs_flwr_spp$julian_median, "gamma")
plot(fit.gamma)

```

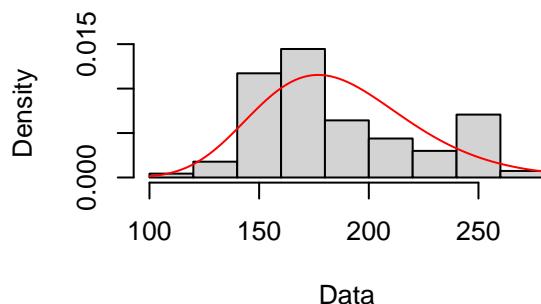


```
# Weibull distribution
fit.weibull <- fitdist(kbs_flwr_spp$julian_median, "weibull")
plot(fit.weibull)
```

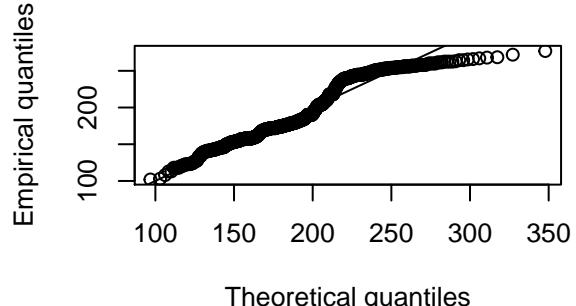


```
# Lognormal distribution
fit.ln <- fitdist(kbs_flwr_spp$julian_median, "lnorm")
plot(fit.ln)
```

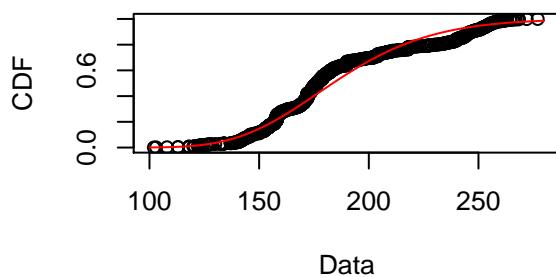
**Empirical and theoretical dens.**



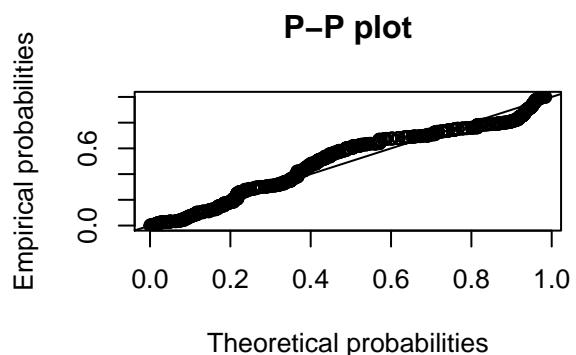
**Q-Q plot**



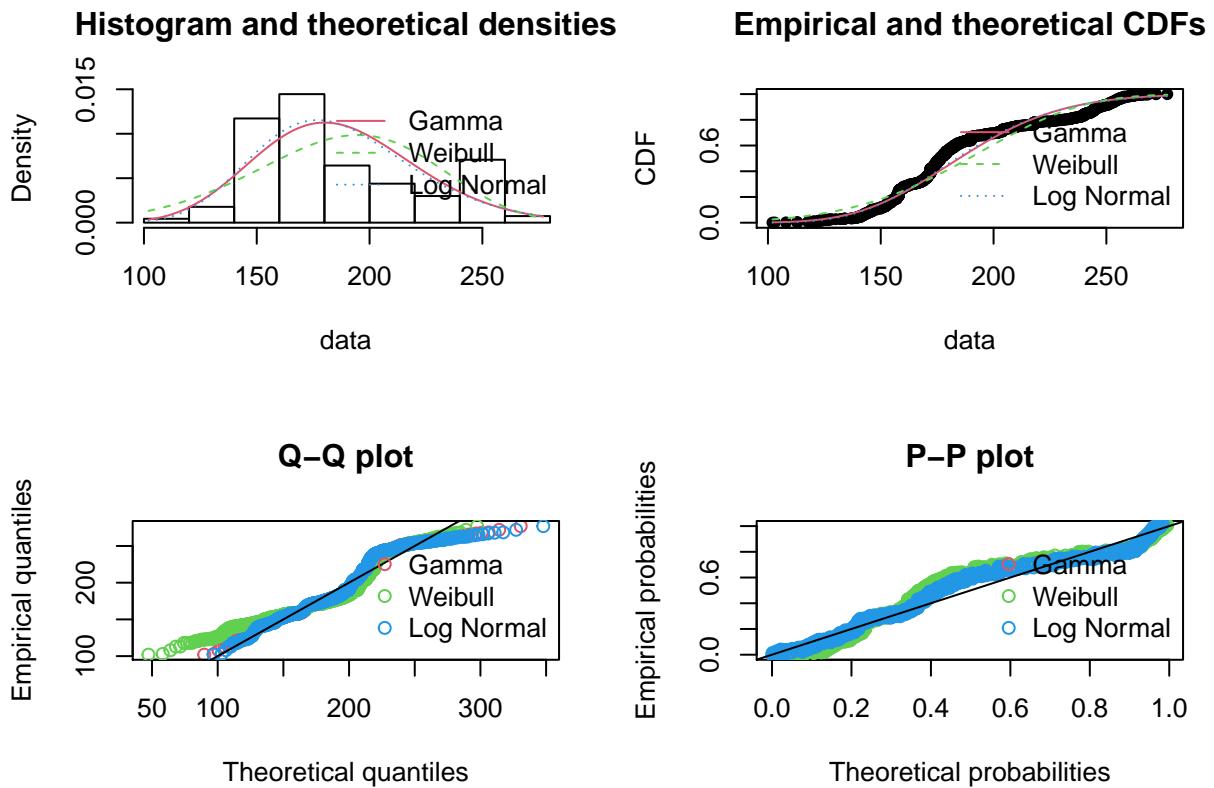
**Empirical and theoretical CDFs**



**P-P plot**



```
par(mfrow = c(2, 2))
plot.legend <- c("Gamma", "Weibull", "Log Normal")
denscomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
cdfcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
qqcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
ppcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
```



```
# Goodness of fit comparisons across fits (can't include the sqrt normal bc it
# becomes diff response values)
gofstat(list(fit.gamma, fit.weibull, fit.ln), fitnames = c("Gamma", "Weibull", "Log Normal"))
```

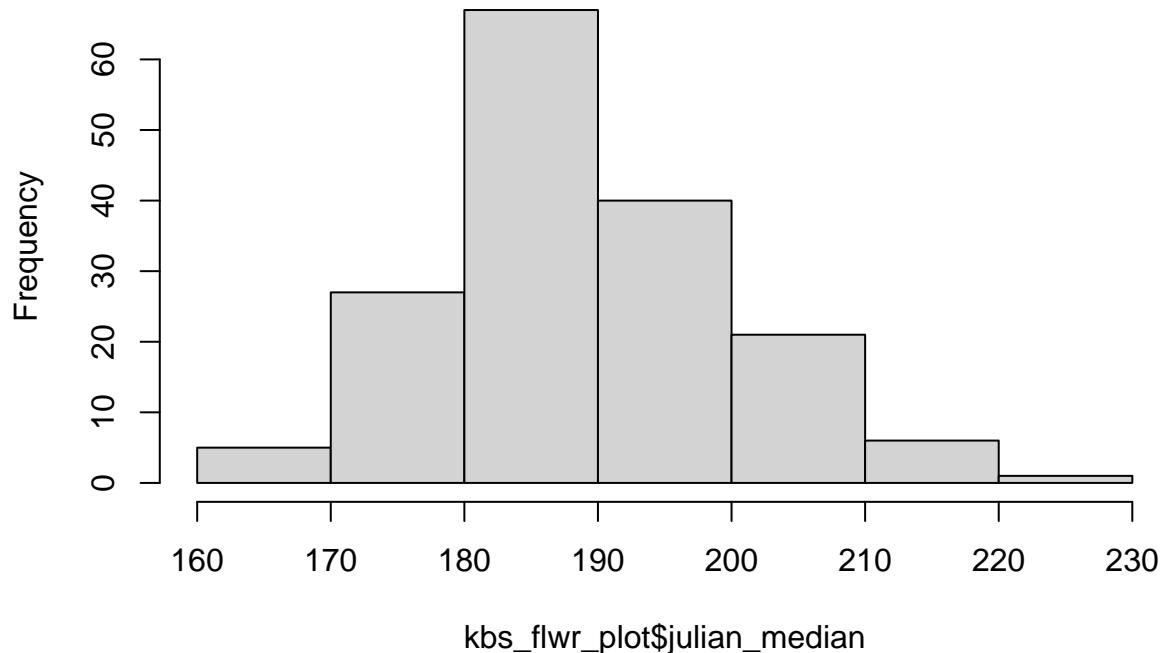
```
## Goodness-of-fit statistics
##                                     Gamma   Weibull Log Normal
## Kolmogorov-Smirnov statistic 0.1255364 0.1657855 0.112895
## Cramer-von Mises statistic  4.2217015 7.6018720 3.401248
## Anderson-Darling statistic 24.6691234 42.0641014 20.431315
##
## Goodness-of-fit criteria
##                                     Gamma   Weibull Log Normal
## Akaike's Information Criterion 11501.63 11668.78 11480.01
## Bayesian Information Criterion 11511.73 11678.88 11490.10
```

```
# Lognormal is the best so going to log transform response variable julian_median
```

## KBS PLOT LEVEL - Looking at MEDIAN JULIAN DAY

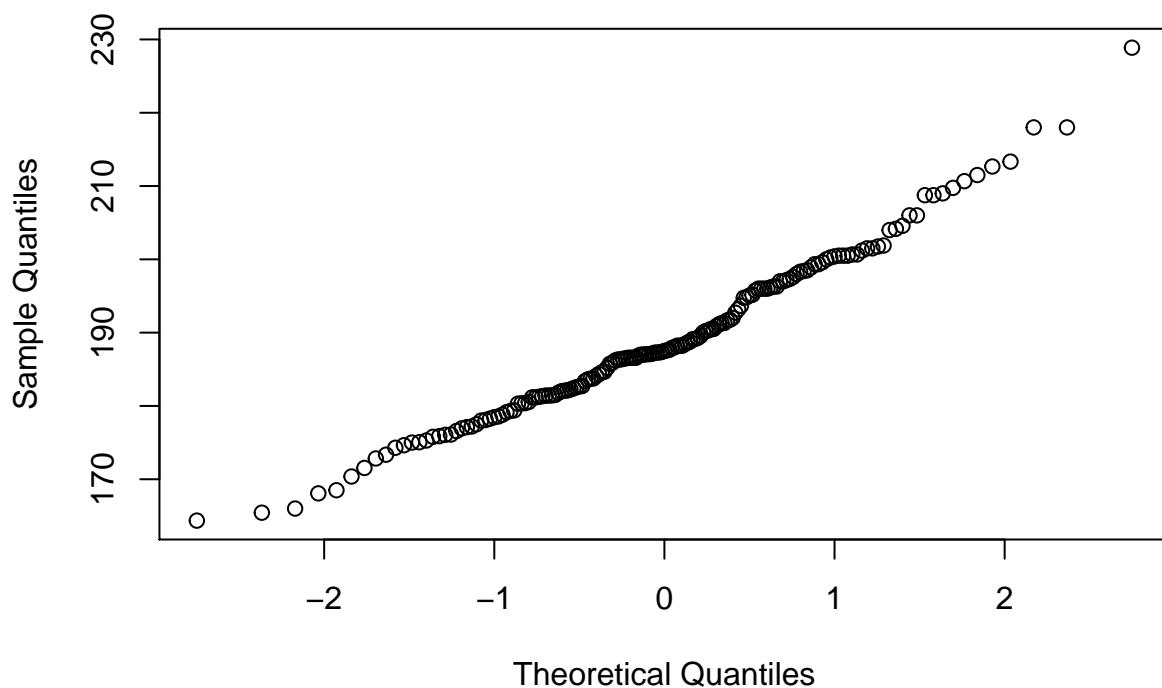
```
### UMBS ####
hist(kbs_flwr_plot$julian_median)
```

### Histogram of kbs\_flwr\_plot\$julian\_median



```
qqnorm(kbs_flwr_plot$julian_median)
```

### Normal Q-Q Plot



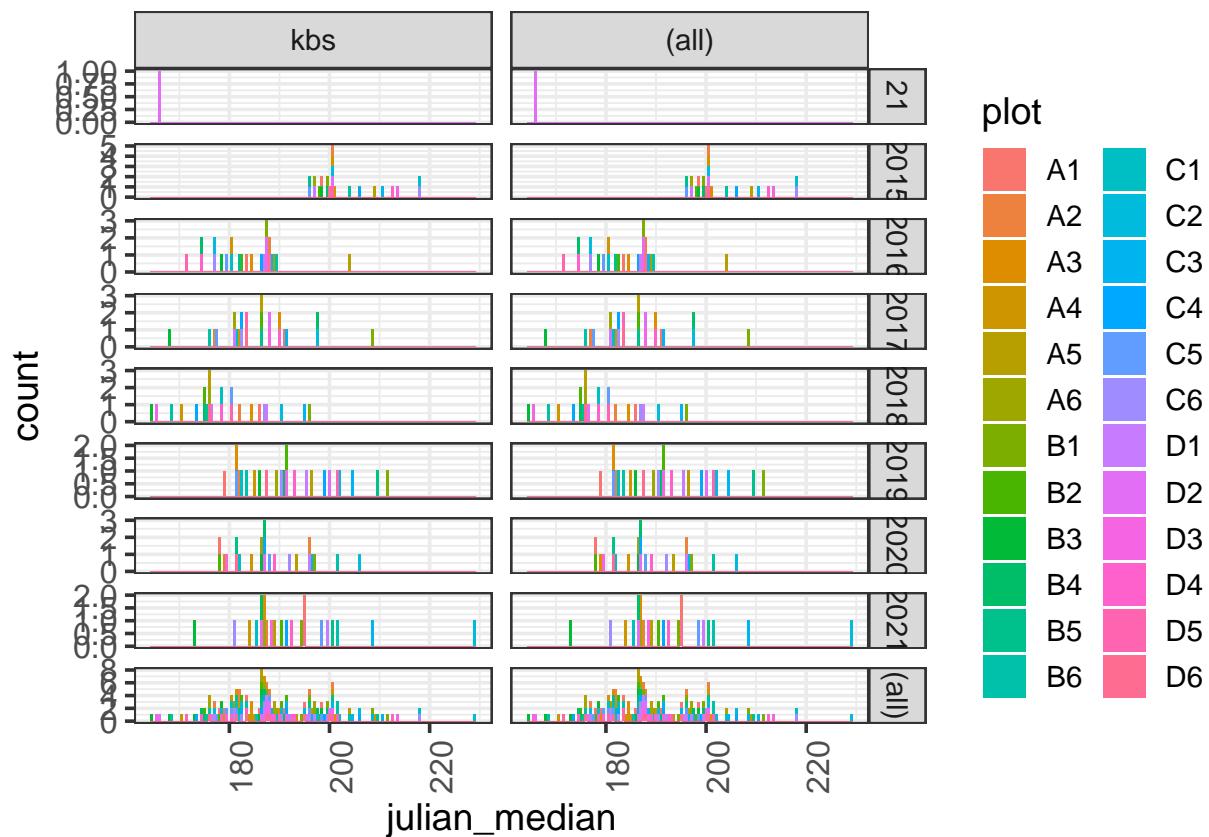
```

shapiro.test(kbs_flwr_plot$julian_median) # pvalue is < 0.05 so we reject the null hypothesis that the

## 
## Shapiro-Wilk normality test
## 
## data: kbs_flwr_plot$julian_median
## W = 0.98221, p-value = 0.03084

# Visualizing median Julian date for kbs at the PLOT LEVEL
ggplot(kbs_flwr_plot, aes(julian_median, fill = plot)) + geom_histogram(binwidth = 0.5) +
  facet_grid(year ~ site, margins = TRUE, scales = "free")

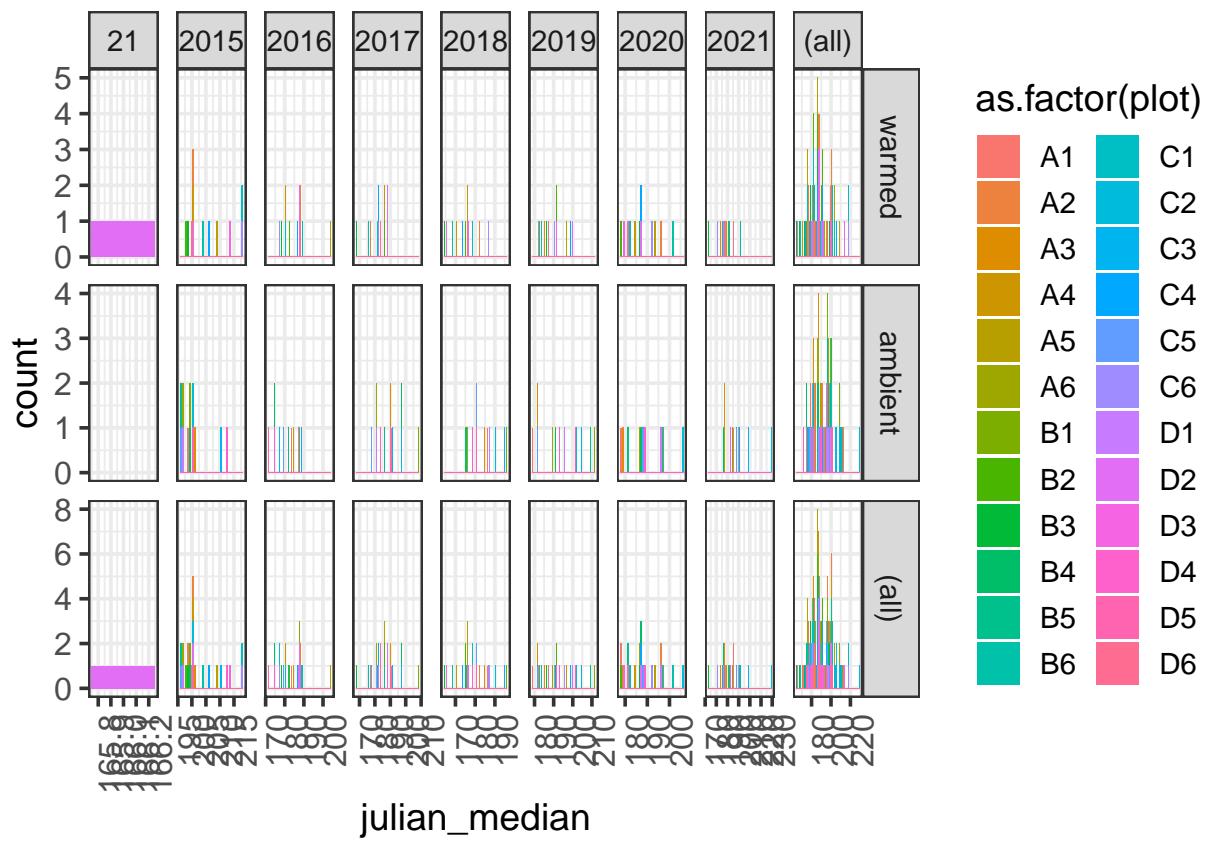
```



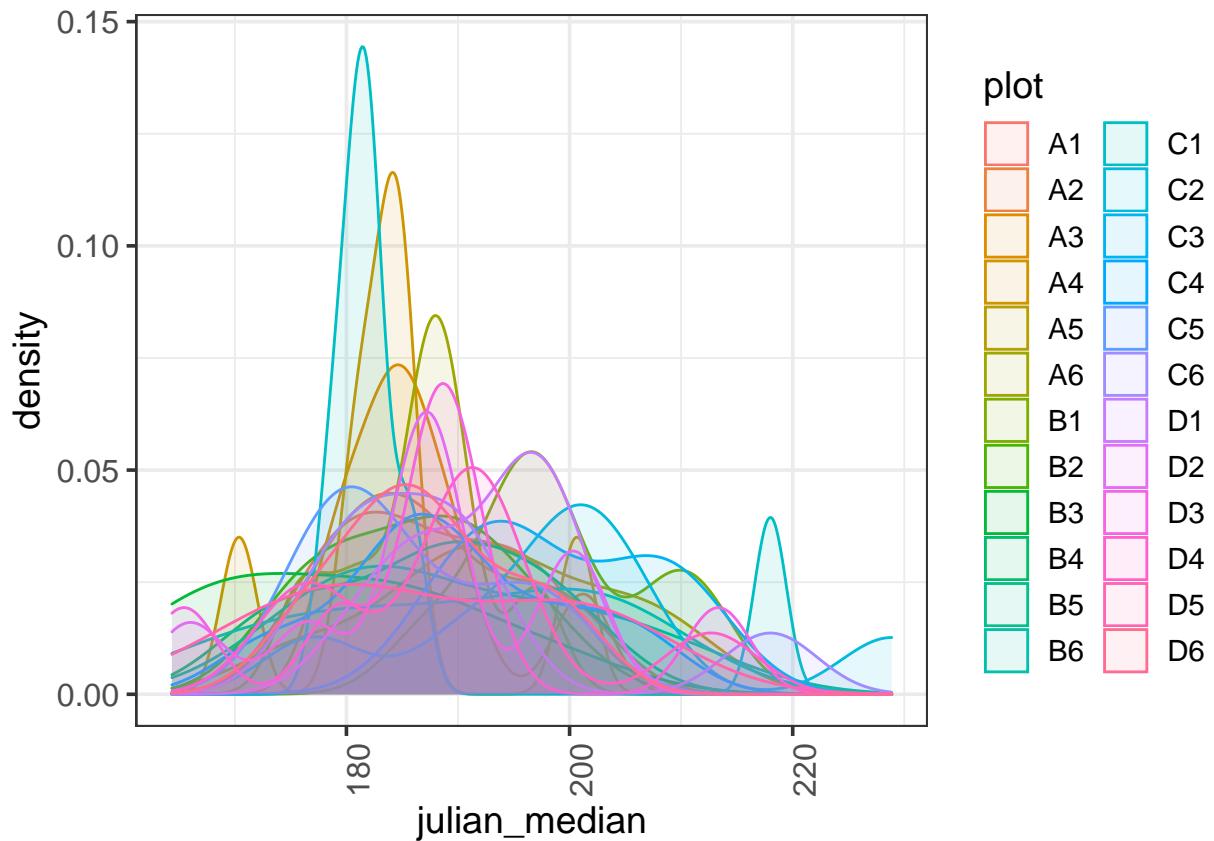
```

ggplot(kbs_flwr_plot, aes(julian_median, fill = as.factor(plot))) + geom_histogram(binwidth = 0.5) +
  facet_grid(state ~ year, margins = TRUE, scales = "free")

```



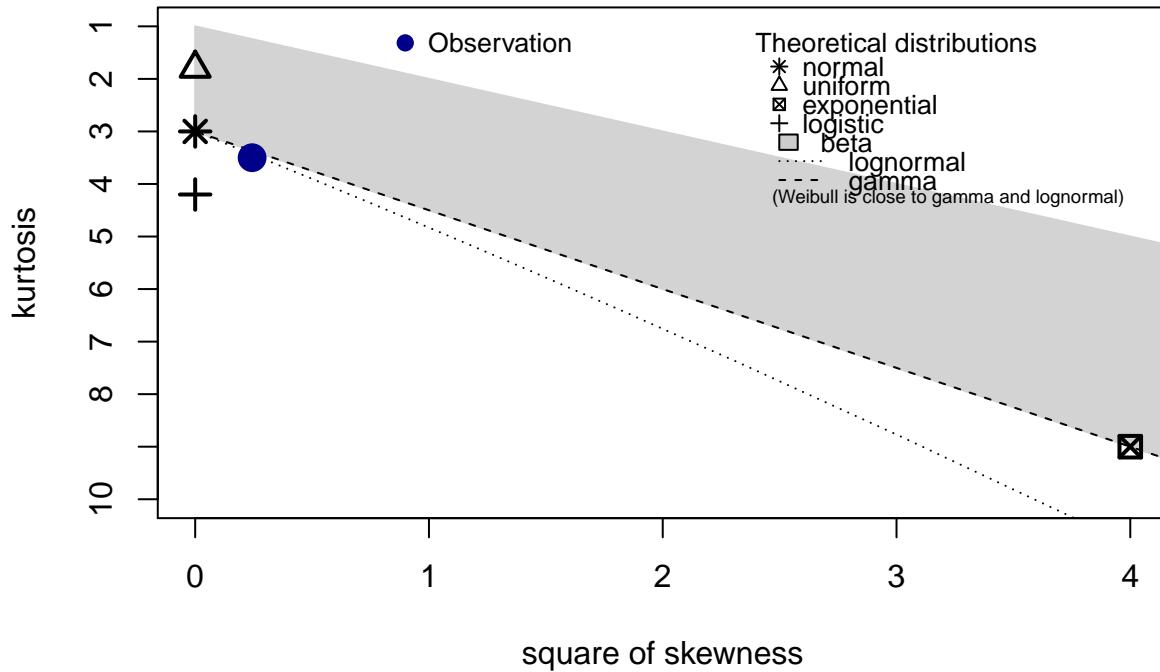
```
ggplot(kbs_flwr_plot, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1)
```



```
# ggplot(umbs_flwr_plot, aes(julian_median, fill = plot, color=plot)) +
# geom_density(alpha = 0.1) + facet_wrap(~year) ggplot(umbs_flwr_plot,
# aes(julian_median, fill = plot, color=plot)) + geom_density(alpha = 0.1) +
# facet_wrap(~year + plot)
```

```
# Exploring distributions for these right-skewed data:
descdist(kbs_flwr_plot$julian_median, discrete = FALSE)
```

## Cullen and Frey graph



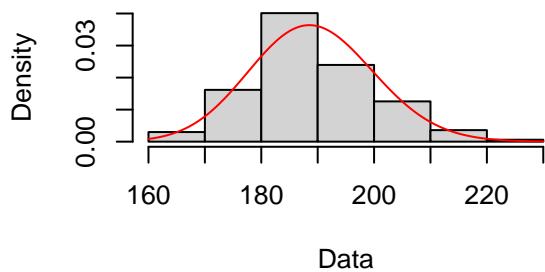
```

## summary statistics
## -----
## min: 164.3571   max: 228.875
## median: 187.5
## mean: 189.1667
## estimated sd: 11.09836
## estimated skewness: 0.4934932
## estimated kurtosis: 3.499969

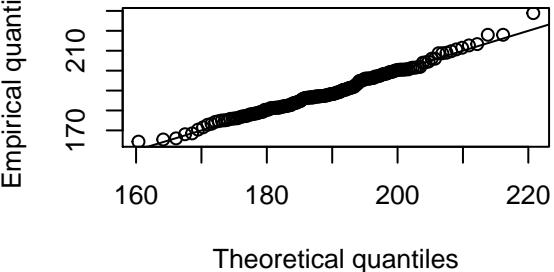
# Gamma distribution
fit.gamma <- fitdist(kbs_flwr_plot$julian_median, "gamma")
plot(fit.gamma)

```

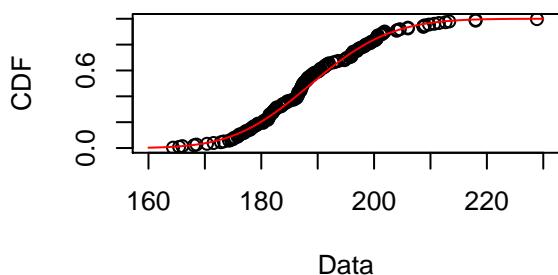
**Empirical and theoretical dens.**



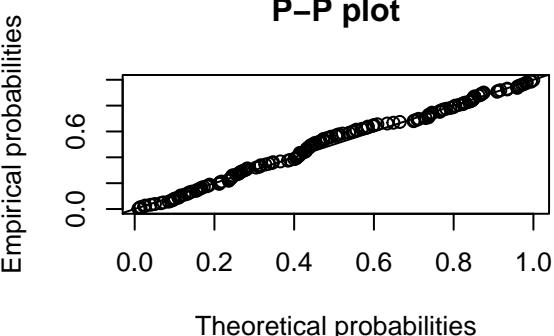
**Q-Q plot**



**Empirical and theoretical CDFs**

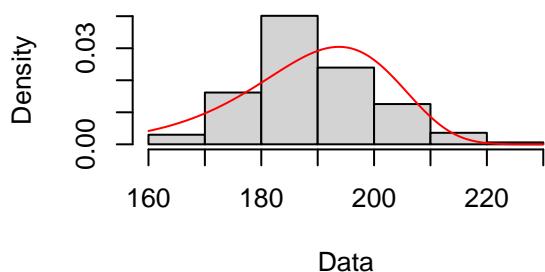


**P-P plot**

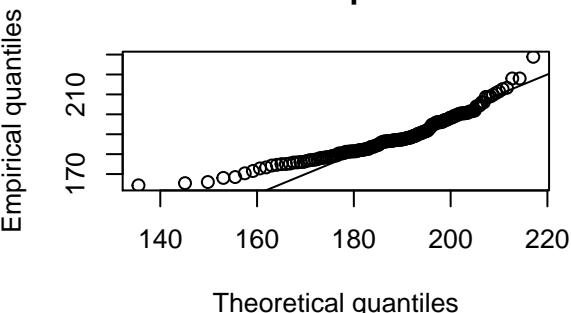


```
# Weibull distribution
fit.weibull <- fitdist(kbs_flwr_plot$julian_median, "weibull")
plot(fit.weibull)
```

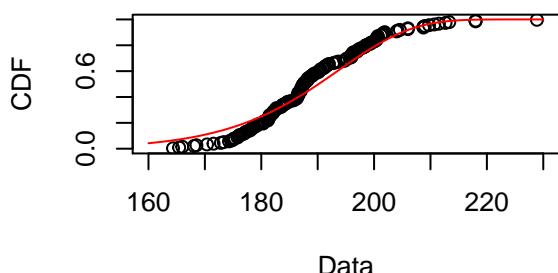
**Empirical and theoretical dens.**



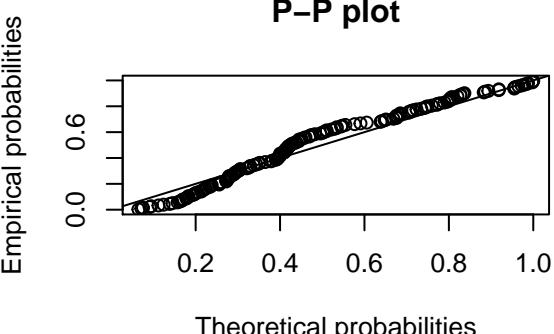
**Q-Q plot**



**Empirical and theoretical CDFs**

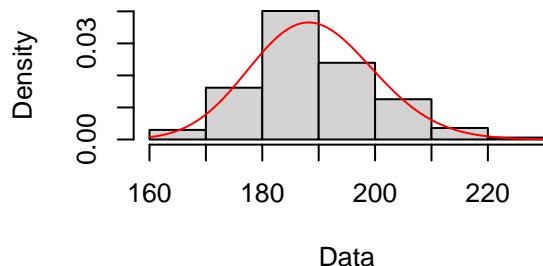


**P-P plot**

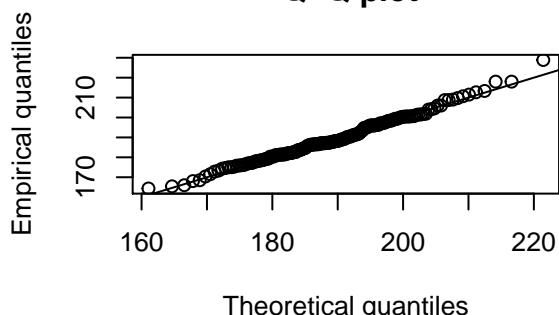


```
# Lognormal distribution
fit.ln <- fitdist(kbs_flwr_plot$julian_median, "lnorm")
plot(fit.ln)
```

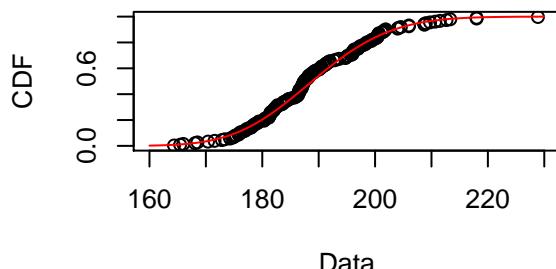
**Empirical and theoretical dens.**



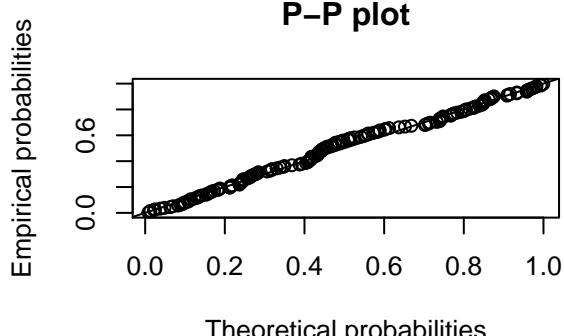
**Q–Q plot**



**Empirical and theoretical CDFs**

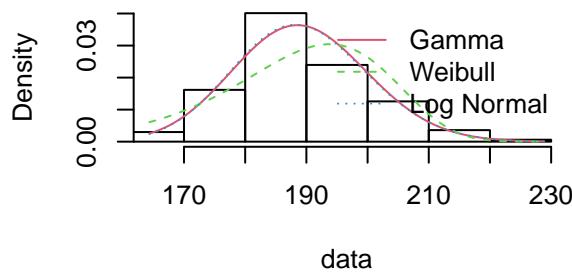


**P–P plot**

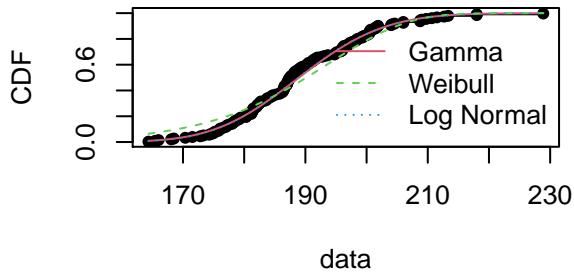


```
par(mfrow = c(2, 2))
plot.legend <- c("Gamma", "Weibull", "Log Normal")
denscomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
cdfcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
qqcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
ppcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
```

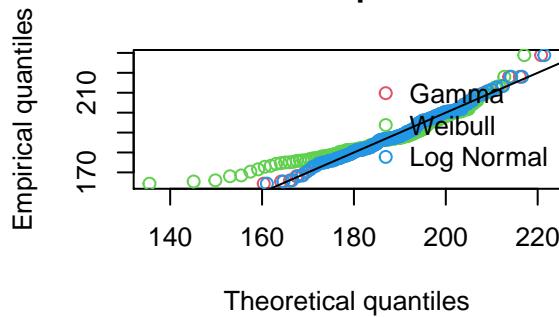
### Histogram and theoretical densities



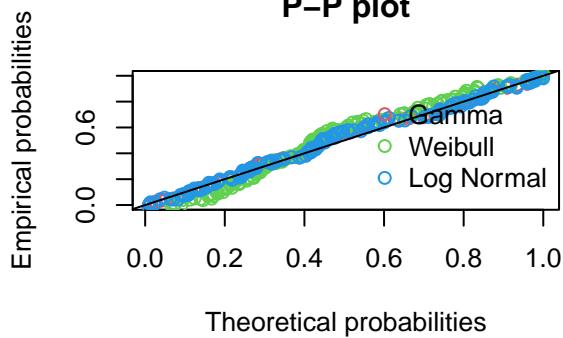
### Empirical and theoretical CDFs



### Q-Q plot



### P-P plot



```
# Goodness of fit comparisons across fits (can't include the sqrt normal bc it
# becomes diff response values)
gofstat(list(fit.gamma, fit.weibull, fit.ln), fitnames = c("Gamma", "Weibull", "Log Normal"))
```

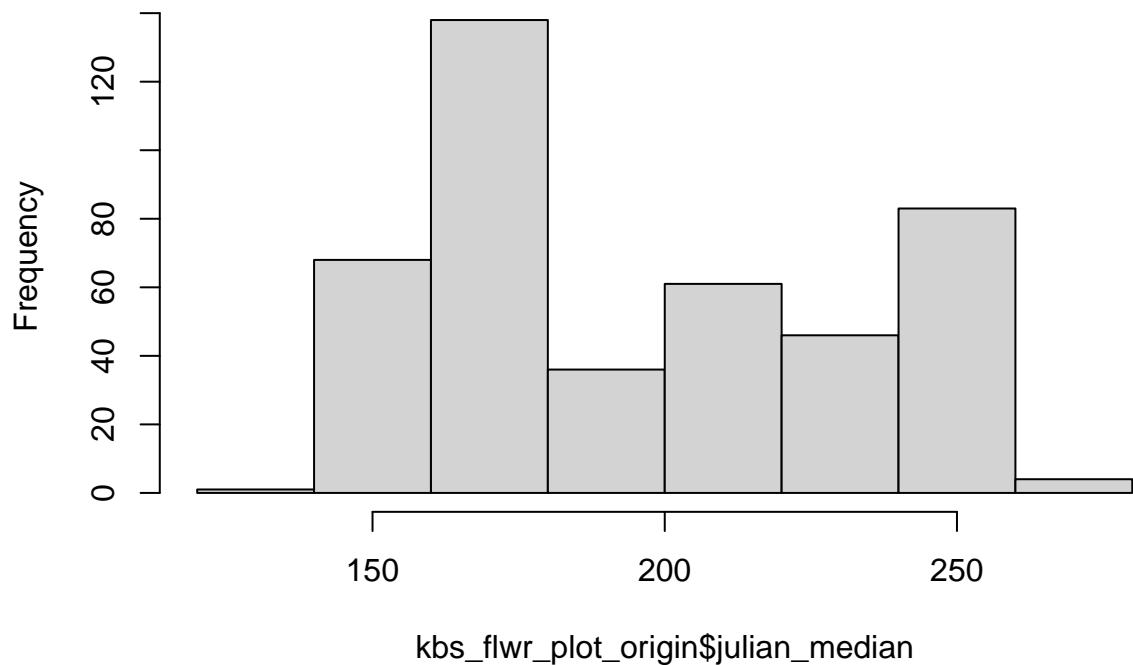
```
## Goodness-of-fit statistics
##                                     Gamma   Weibull Log Normal
## Kolmogorov-Smirnov statistic 0.07157104 0.1067166 0.06774769
## Cramer-von Mises statistic  0.12663515 0.6064607 0.11174987
## Anderson-Darling statistic  0.66002222 3.8934673 0.57853177
##
## Goodness-of-fit criteria
##                                     Gamma   Weibull Log Normal
## Akaike's Information Criterion 1278.054 1320.667 1276.977
## Bayesian Information Criterion 1284.290 1326.903 1283.213
```

```
# Log normal is the best first so going to log transform response variable
# julian_median
```

## KBS PLOT LEVEL ORIGIN - Looking at MEDIAN JULIAN DAY

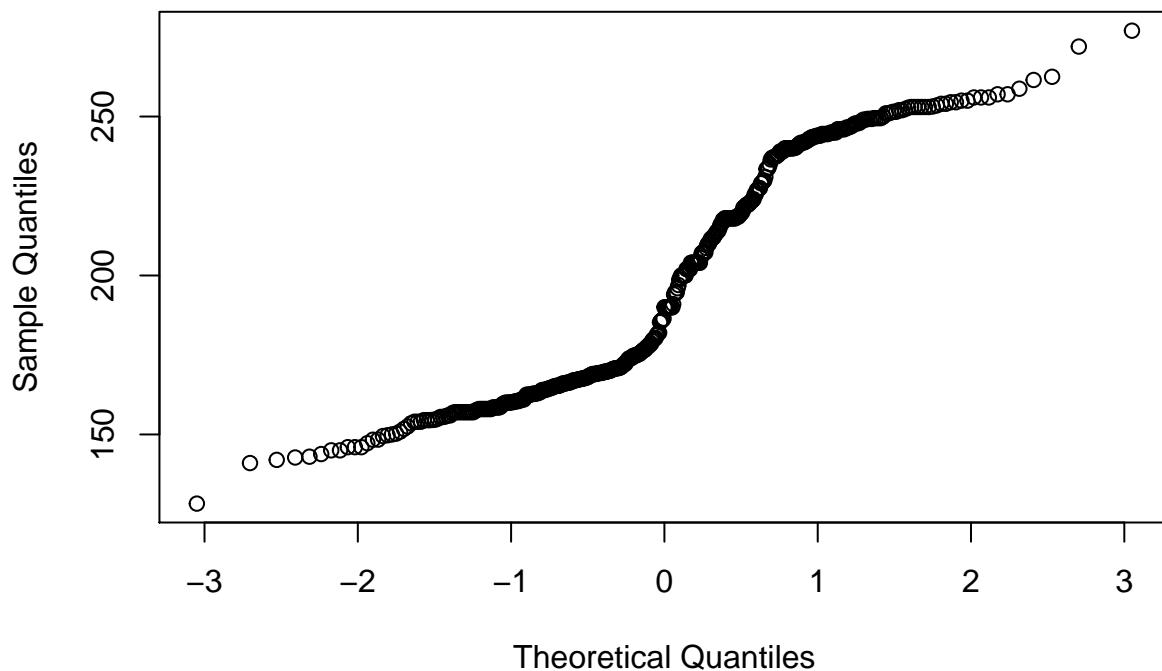
```
### KBS ####
hist(kbs_flwr_plot_origin$julian_median)
```

### Histogram of kbs\_flwr\_plot\_origin\$julian\_median



```
qqnorm(kbs_flwr_plot_origin$julian_median)
```

### Normal Q-Q Plot



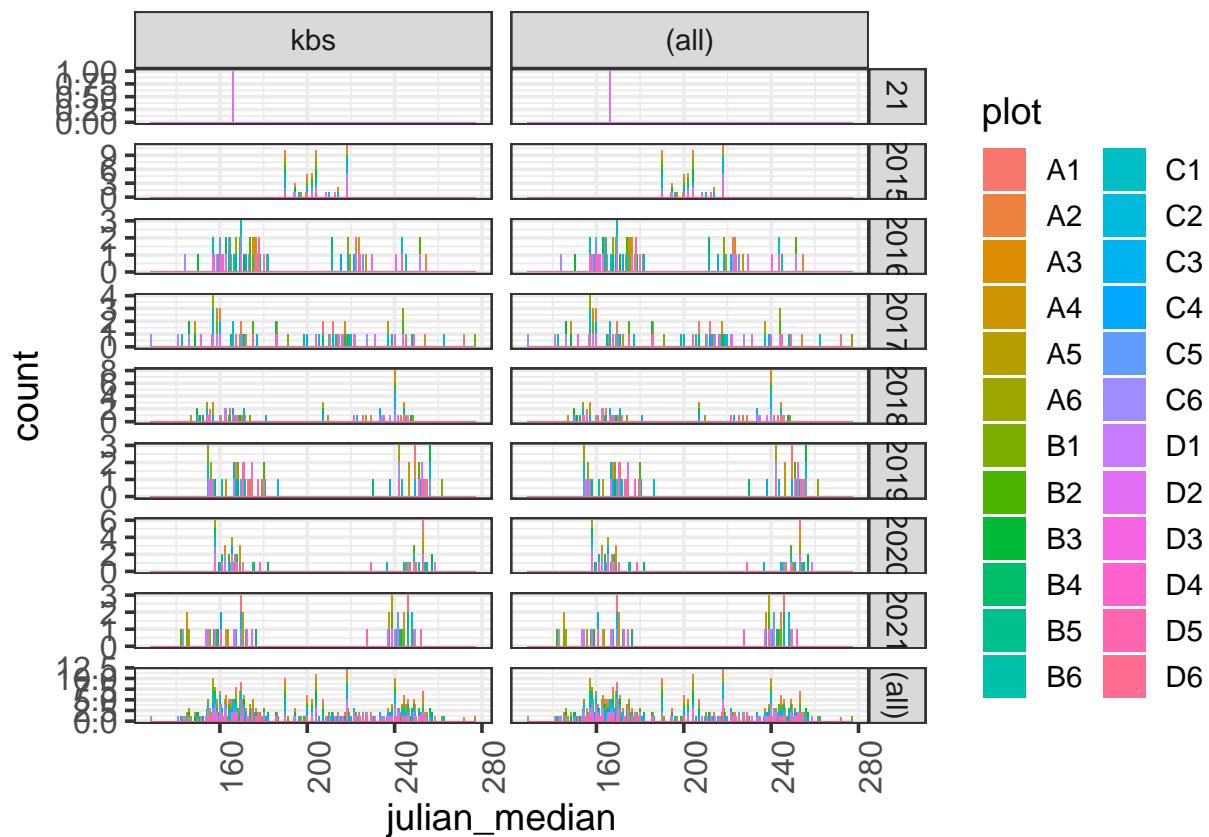
```

shapiro.test(umbs_flwr_plot_origin$julian_median) # pvalue is < 0.05 so we reject the null hypothesis

## 
## Shapiro-Wilk normality test
## 
## data: umbs_flwr_plot_origin$julian_median
## W = 0.9393, p-value = 2.808e-15

# Visualizing median Julian date for kbs at the PLOT LEVEL
ggplot(kbs_flwr_plot_origin, aes(julian_median, fill = plot)) + geom_histogram(binwidth = 0.5) +
  facet_grid(year ~ site, margins = TRUE, scales = "free")

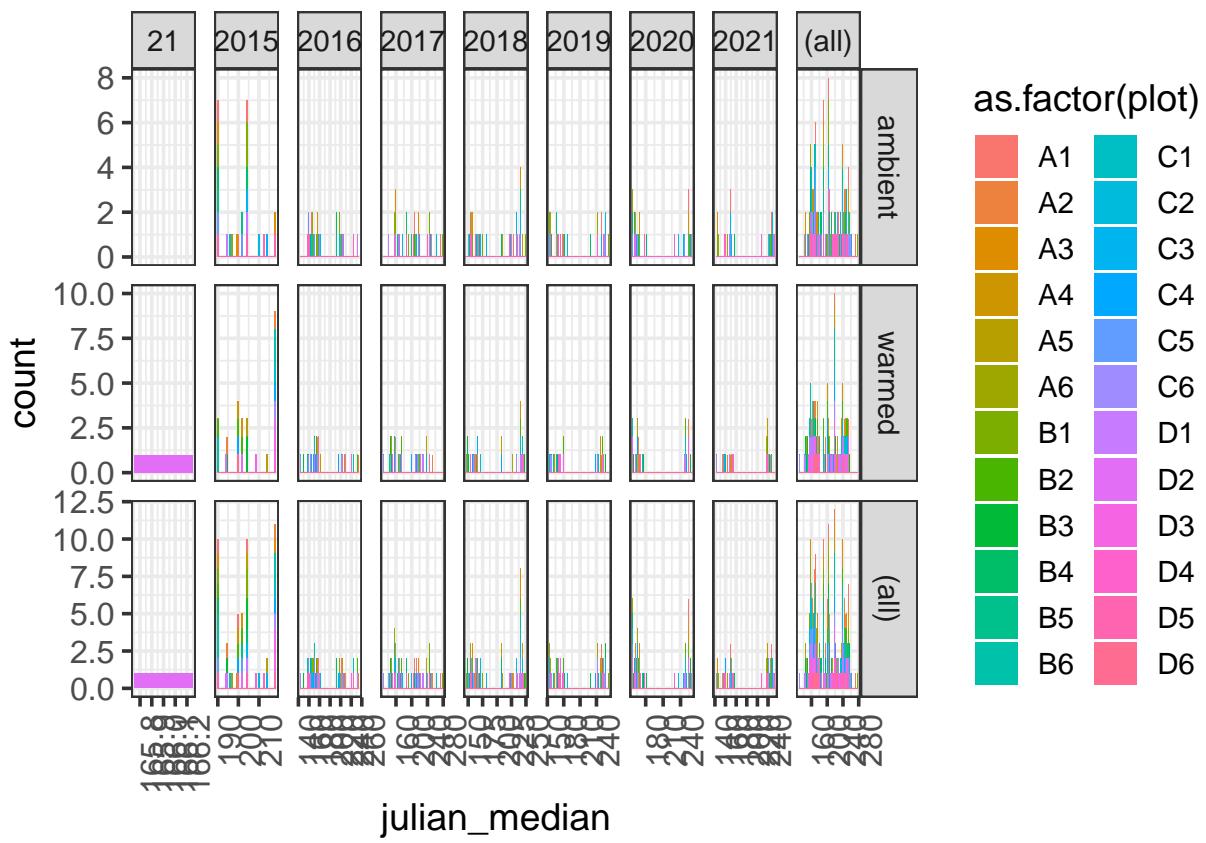
```



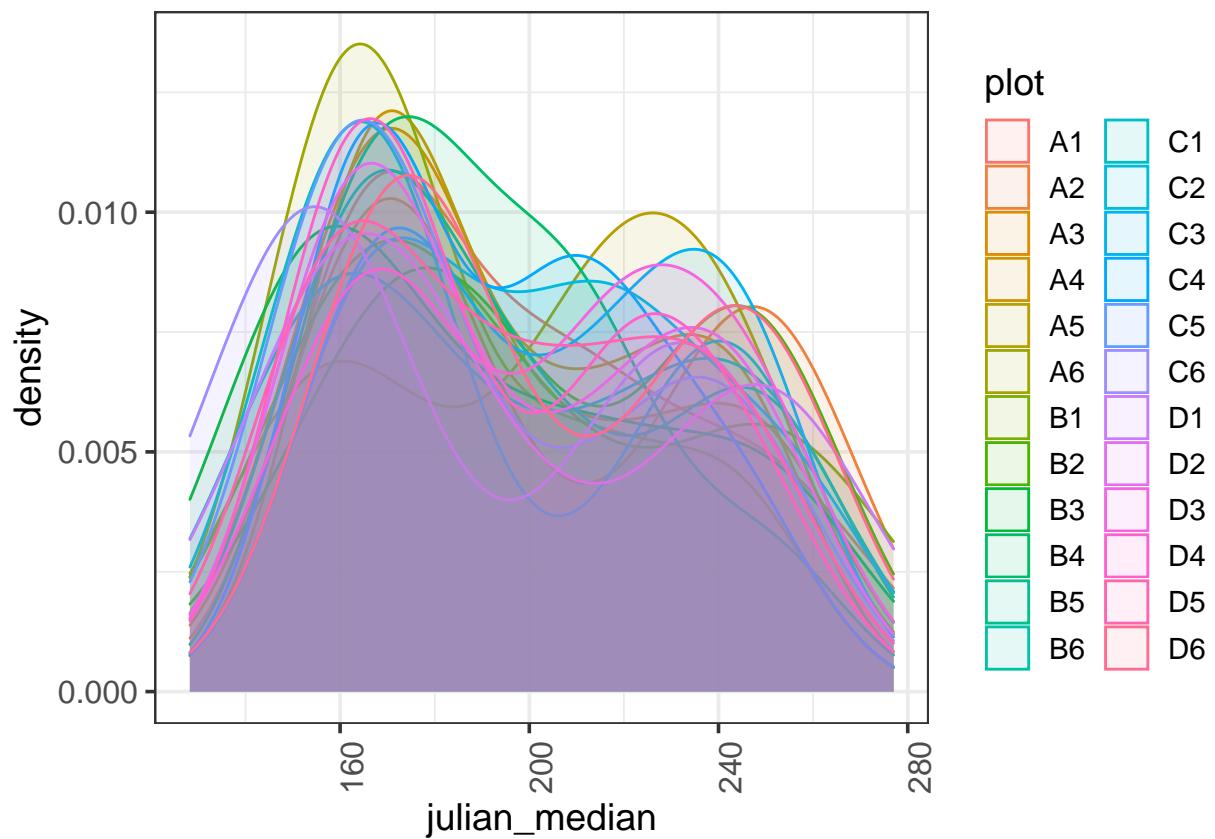
```

ggplot(kbs_flwr_plot_origin, aes(julian_median, fill = as.factor(plot))) + geom_histogram(binwidth = 0.5) +
  facet_grid(state ~ year, margins = TRUE, scales = "free")

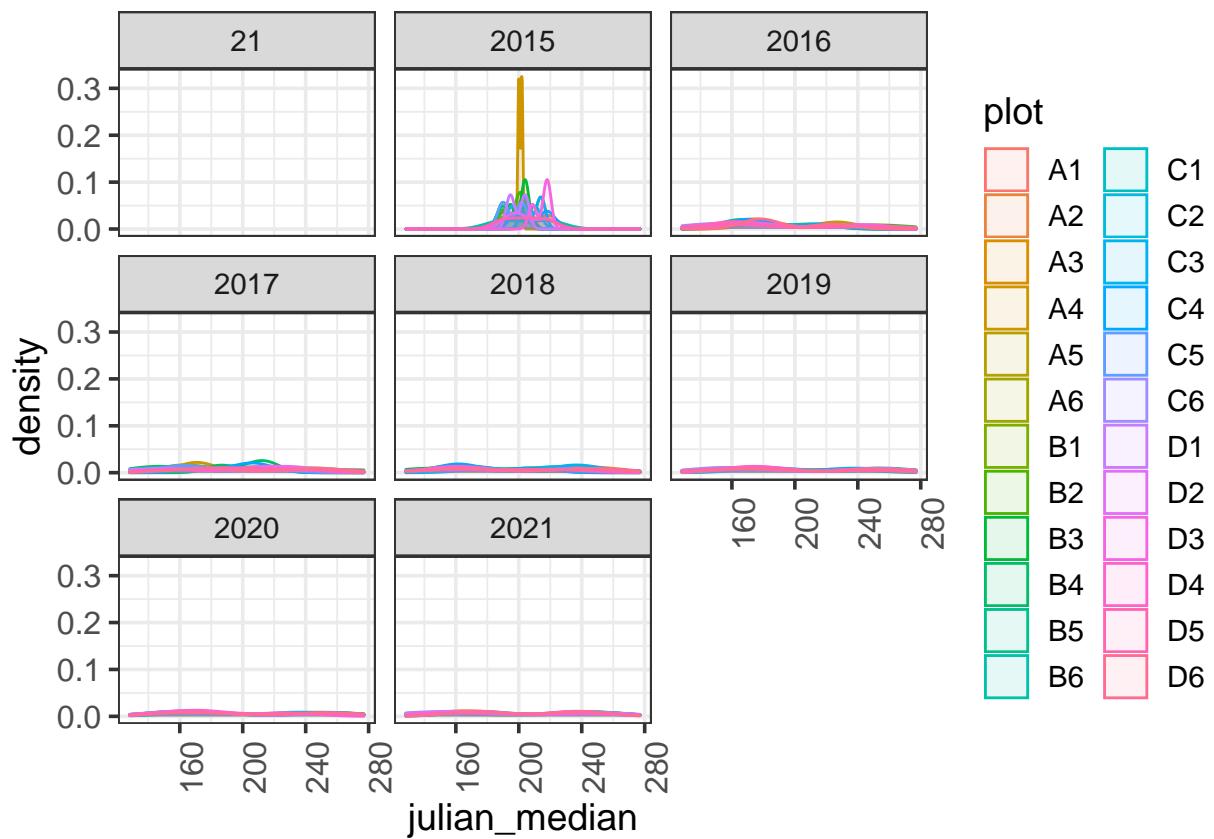
```



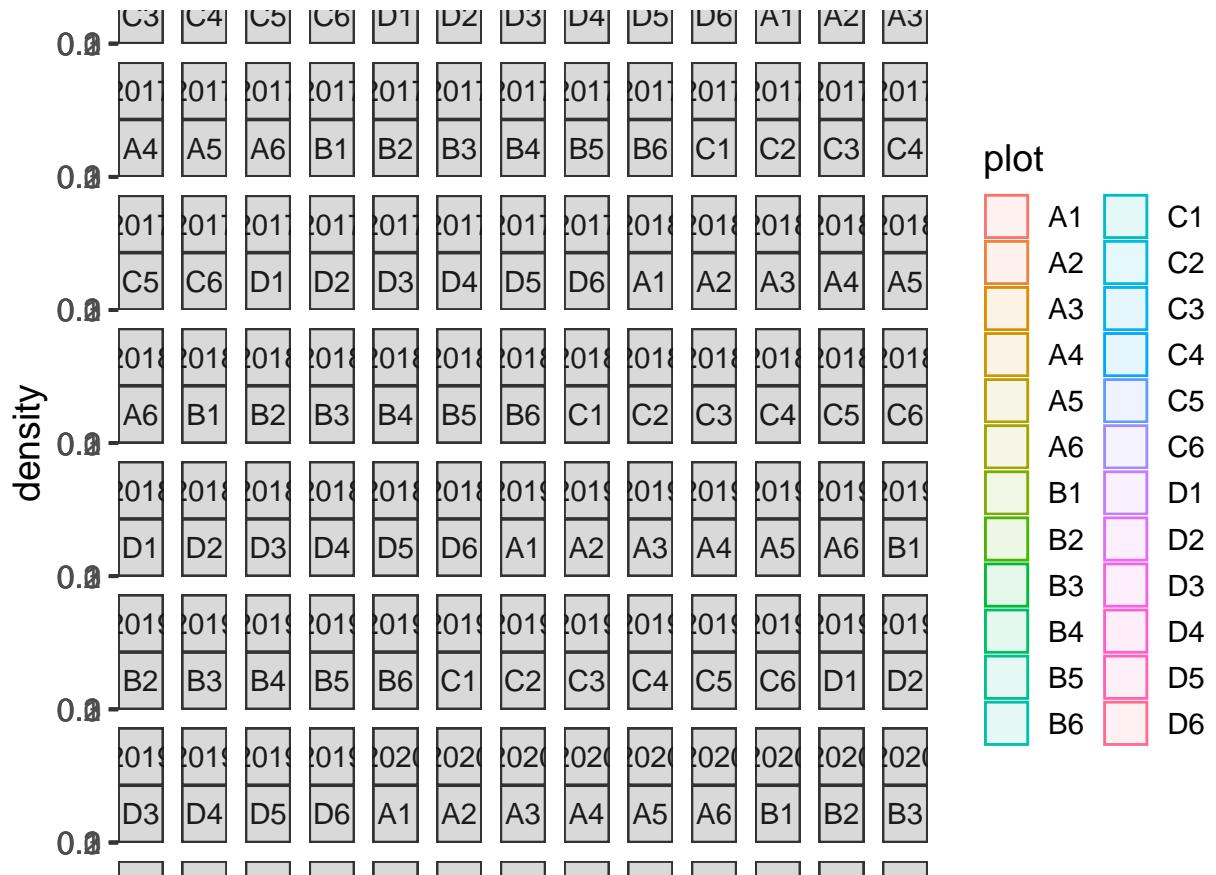
```
ggplot(kbs_flwr_plot_origin, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1)
```



```
ggplot(kbs_flwr_plot_origin, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1)  
  facet_wrap(~year)
```

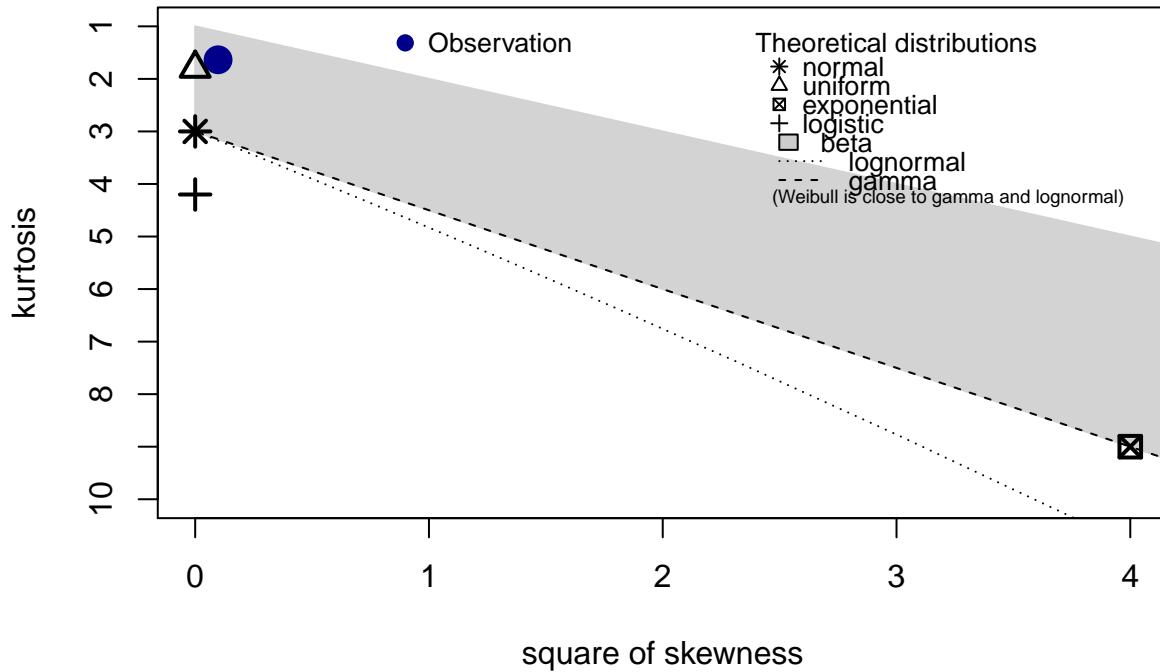


```
ggplot(kbs_flwr_plot_origin, aes(julian_median, fill = plot, color = plot)) + geom_density(alpha = 0.1)
  facet_wrap(~year + plot)
```



```
# Exploring distributions for these data:  
descdist(kbs_flwr_plot_origin$julian_median, discrete = FALSE)
```

## Cullen and Frey graph

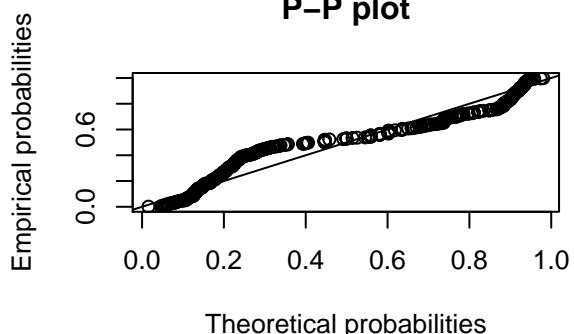
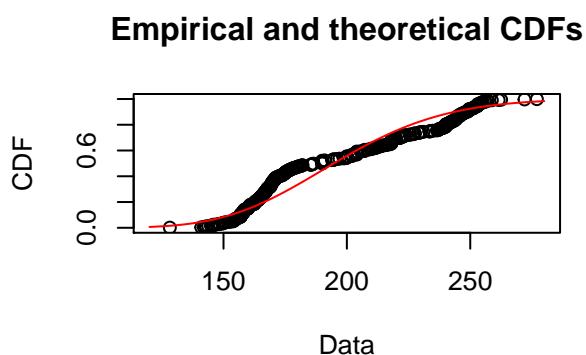
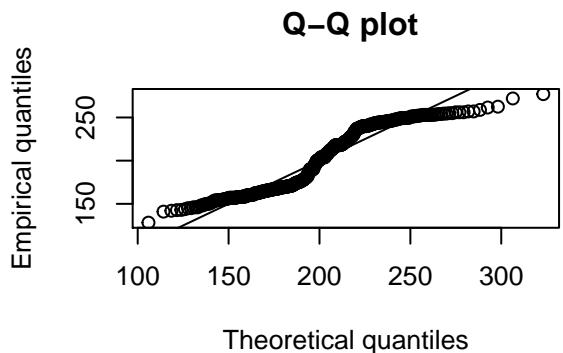
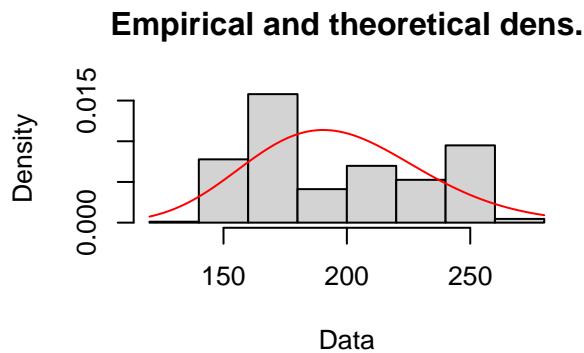


```

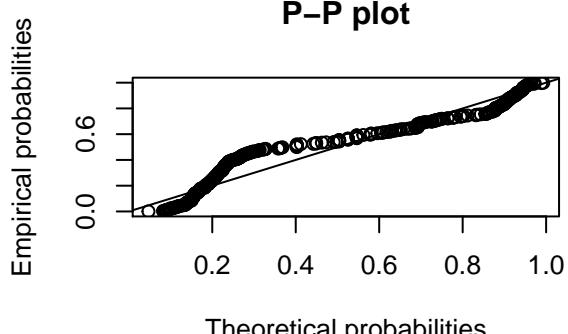
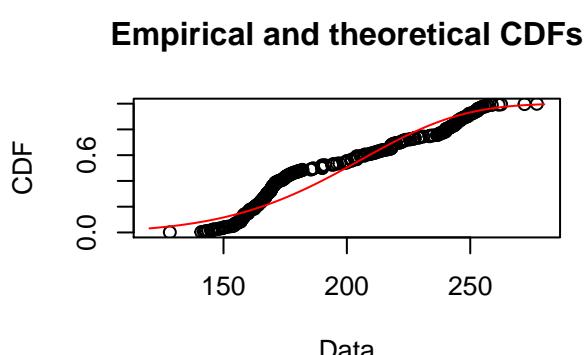
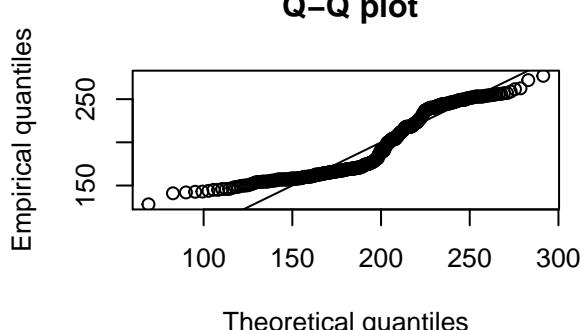
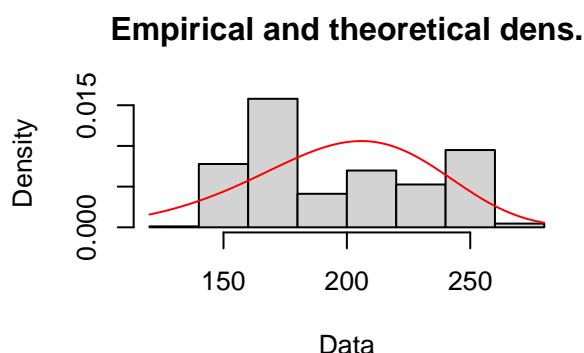
## summary statistics
## -----
## min: 128.25   max: 277
## median: 190
## mean: 196.8577
## estimated sd: 35.88364
## estimated skewness: 0.3131168
## estimated kurtosis: 1.636482

# Gamma distribution
fit.gamma <- fitdist(kbs_flwr_plot_origin$julian_median, "gamma")
plot(fit.gamma)

```

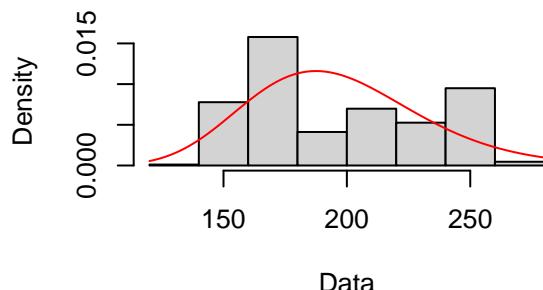


```
# Weibull distribution
fit.weibull <- fitdist(kbs_flwr_plot_origin$julian_median, "weibull")
plot(fit.weibull)
```

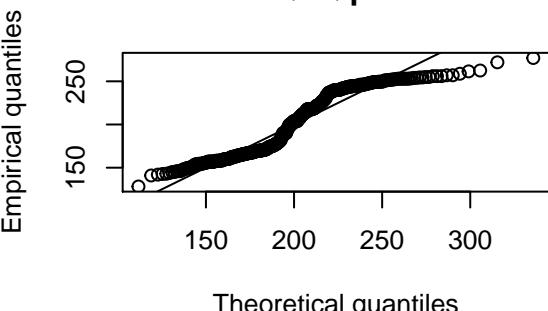


```
# Lognormal distribution
fit.ln <- fitdist(kbs_flwr_plot_origin$julian_median, "lnorm")
plot(fit.ln)
```

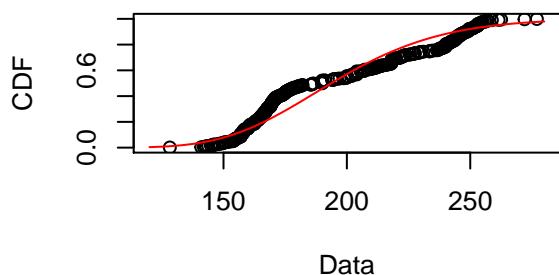
**Empirical and theoretical dens.**



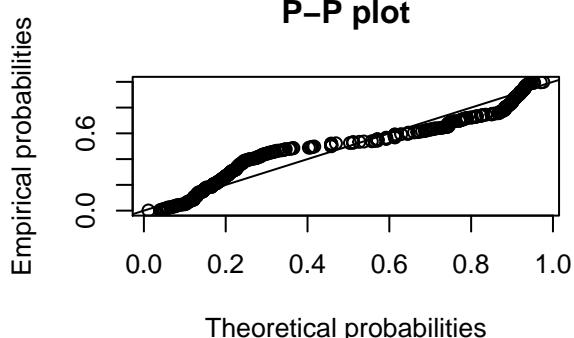
**Q–Q plot**



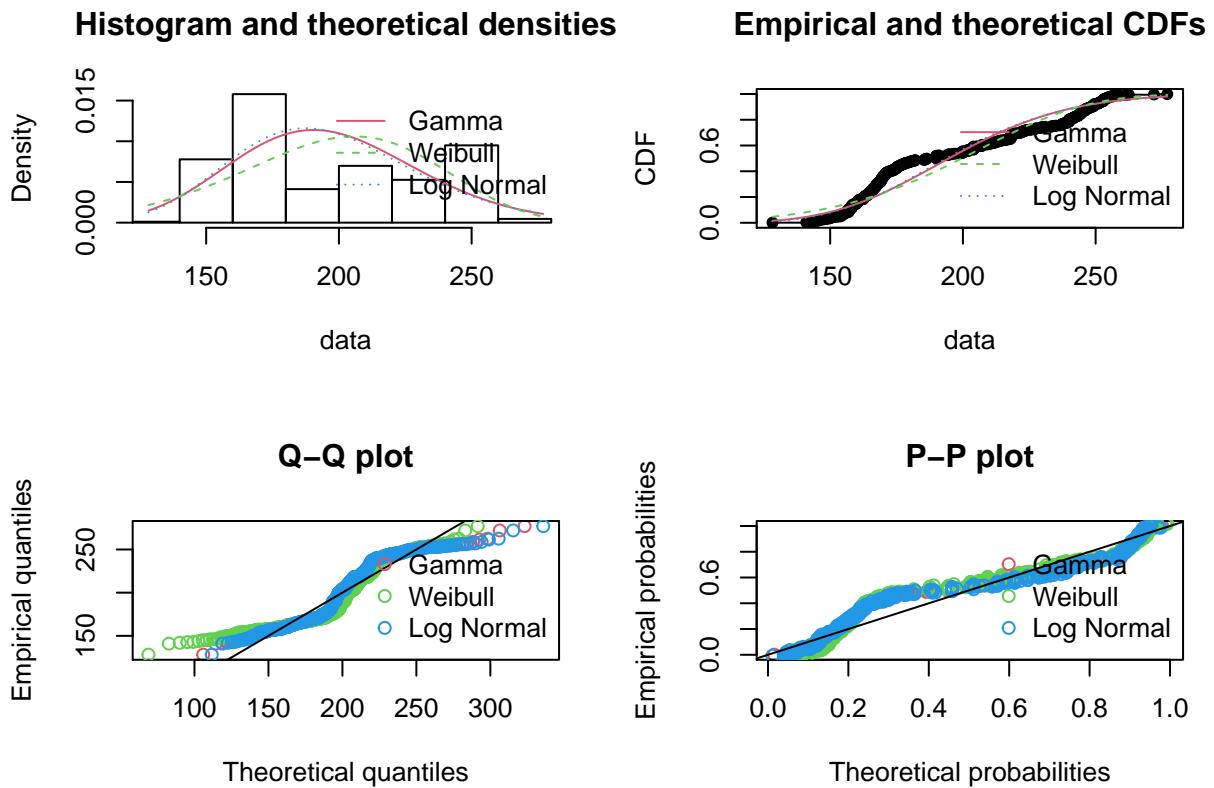
**Empirical and theoretical CDFs**



**P–P plot**



```
par(mfrow = c(2, 2))
plot.legend <- c("Gamma", "Weibull", "Log Normal")
denscomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
cdfcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
qqcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
ppcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
```



```
# Goodness of fit comparisons across fits (can't include the sqrt normal bc it
# becomes diff response values)
gofstat(list(fit.gamma, fit.weibull, fit.ln), fitnames = c("Gamma", "Weibull", "Log Normal"))

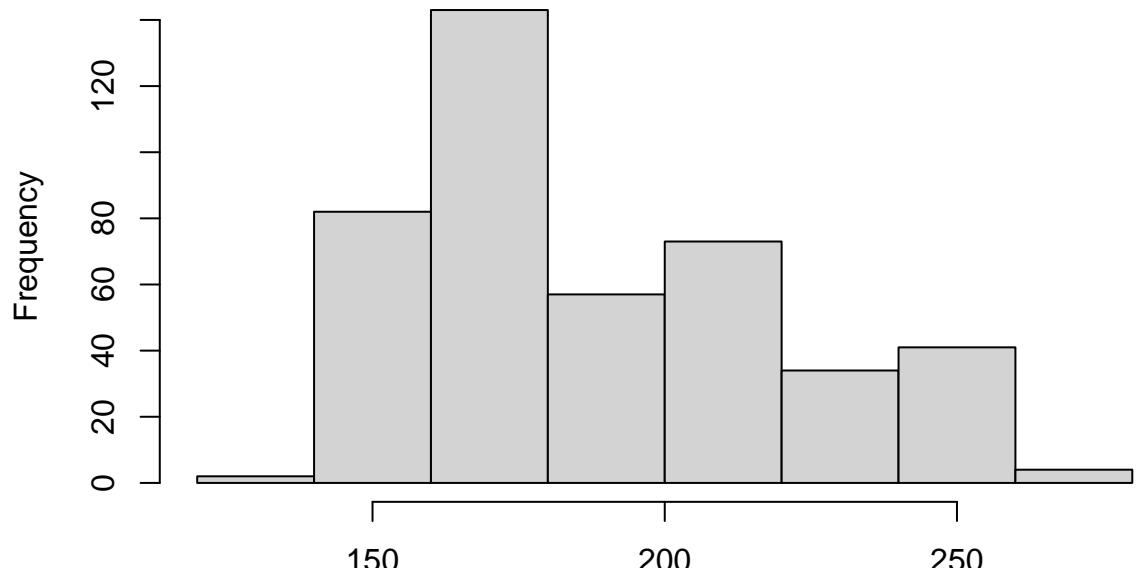
## Goodness-of-fit statistics
##                                     Gamma   Weibull Log Normal
## Kolmogorov-Smirnov statistic 0.1520427 0.1686025 0.1455288
## Cramer-von Mises statistic  2.4745439 2.6896906 2.3902748
## Anderson-Darling statistic 14.7981691 16.2267847 14.2550989
##
## Goodness-of-fit criteria
##                                     Gamma   Weibull Log Normal
## Akaike's Information Criterion 4355.035 4390.966 4350.107
## Bayesian Information Criterion 4363.195 4399.126 4358.267

# log normal looks like it's the best fit
```

## KBS PLOT LEVEL GROWTH HABIT - Looking at MEDIAN JULIAN DAY

```
### KBS ####
hist(kbs_flwr_plot_growthhabit$julian_median)
```

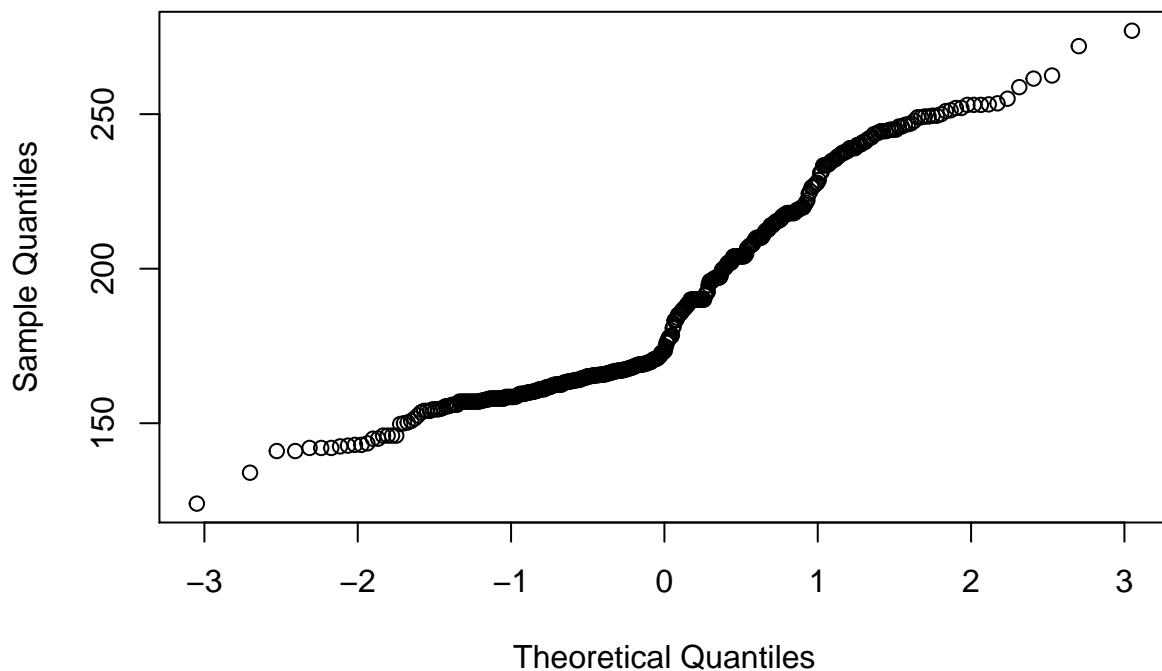
## Histogram of kbs\_flwr\_plot\_growthhabit\$julian\_median



kbs\_flwr\_plot\_growthhabit\$julian\_median

```
qqnorm(kbs_flwr_plot_growthhabit$julian_median)
```

## Normal Q-Q Plot



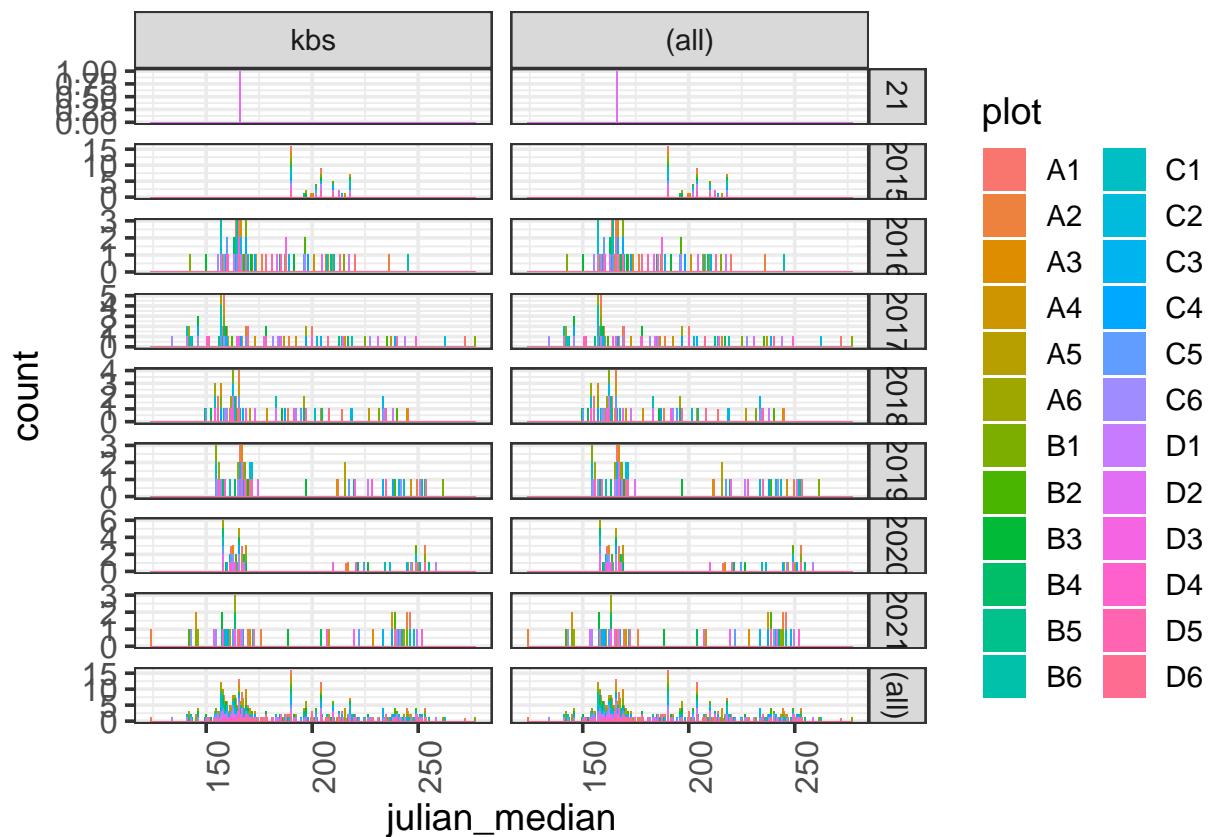
```

shapiro.test(kbs_flwr_plot_growthhabit$julian_median) # pvalue is < 0.05 so we reject the null hypothesis

## 
## Shapiro-Wilk normality test
## 
## data: kbs_flwr_plot_growthhabit$julian_median
## W = 0.91399, p-value = 4.797e-15

# Visualizing median Julian date for kbs at the PLOT LEVEL
ggplot(kbs_flwr_plot_growthhabit, aes(julian_median, fill = plot)) + geom_histogram(binwidth = 0.5) +
  facet_grid(year ~ site, margins = TRUE, scales = "free")

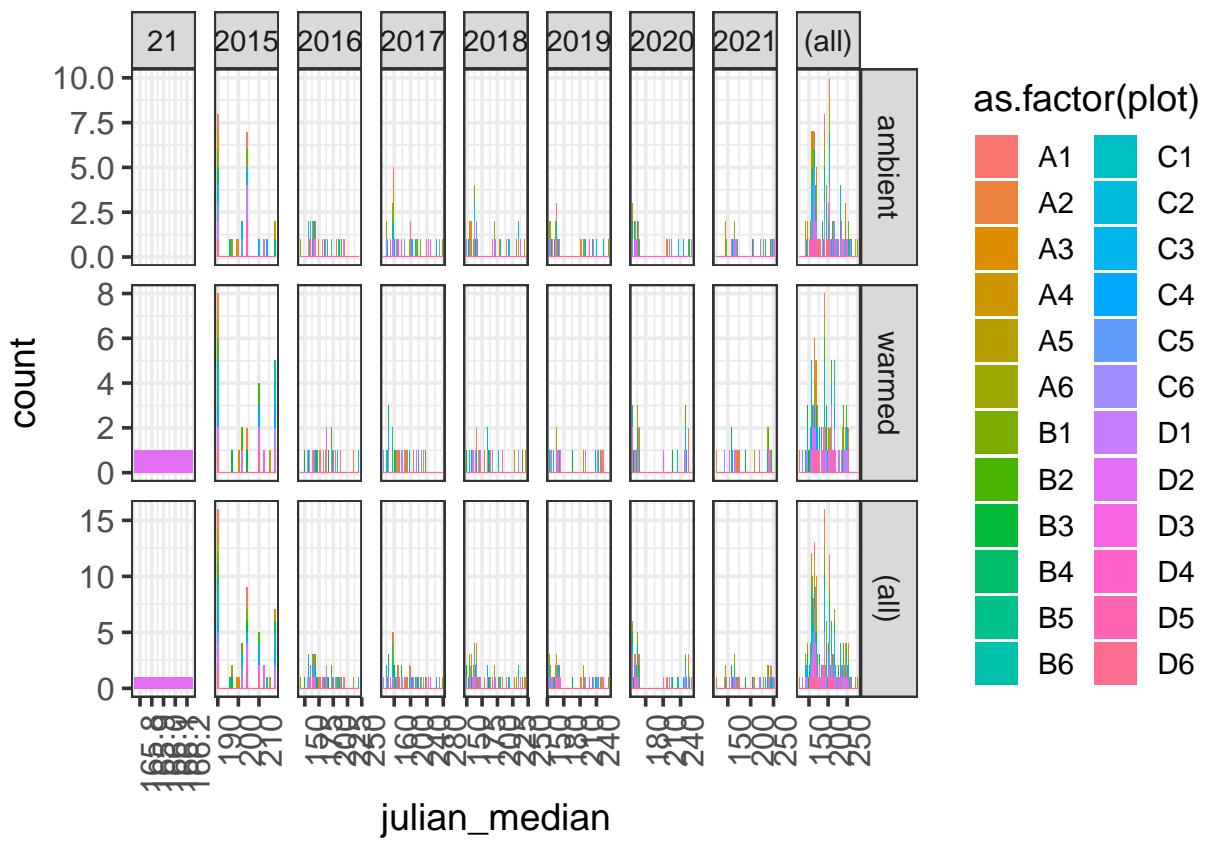
```



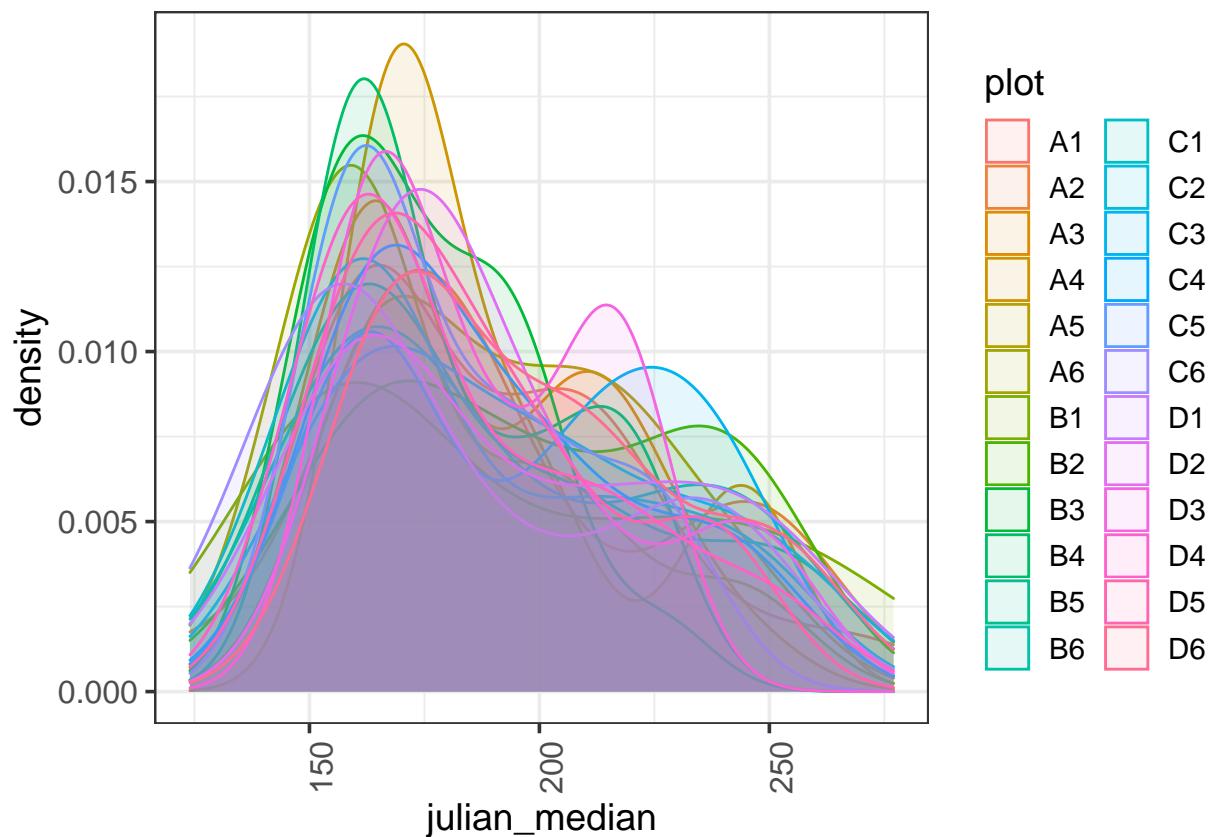
```

ggplot(kbs_flwr_plot_growthhabit, aes(julian_median, fill = as.factor(plot))) + geom_histogram(binwidth = 0.5) +
  facet_grid(state ~ year, margins = TRUE, scales = "free")

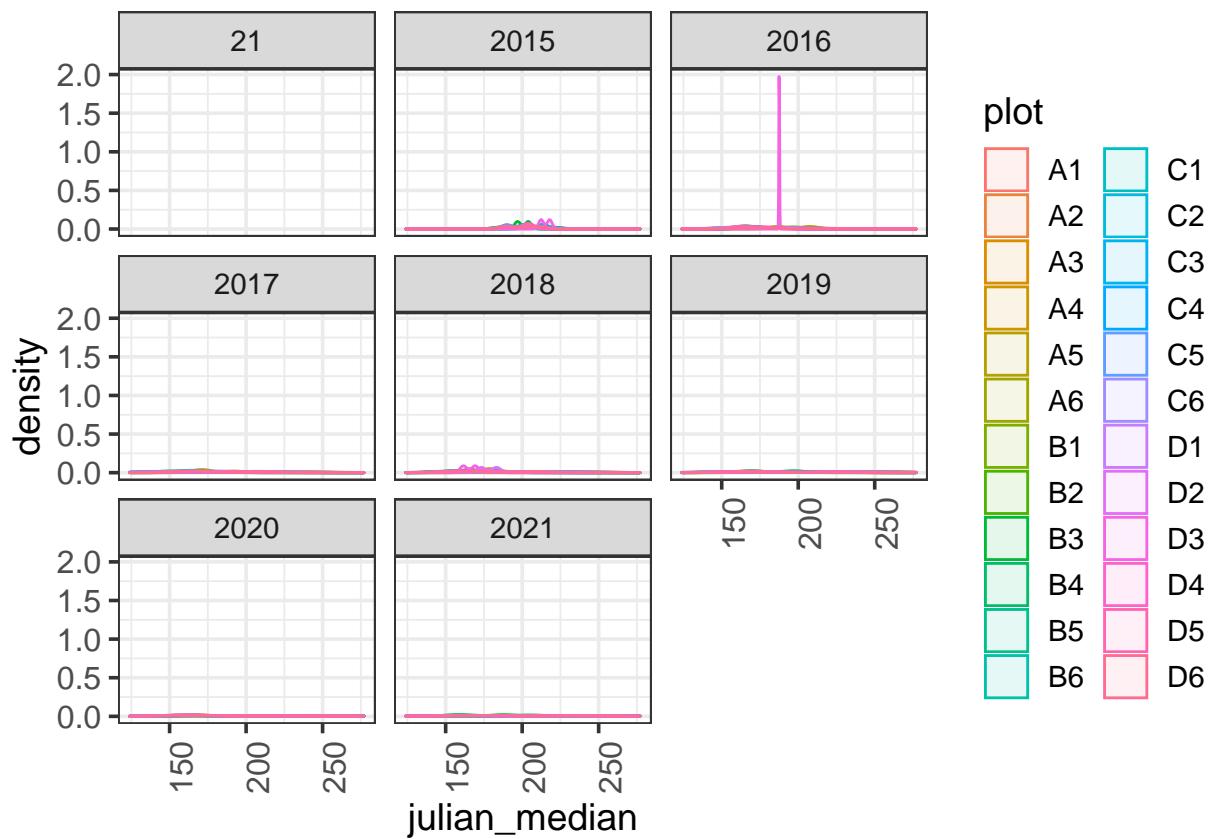
```



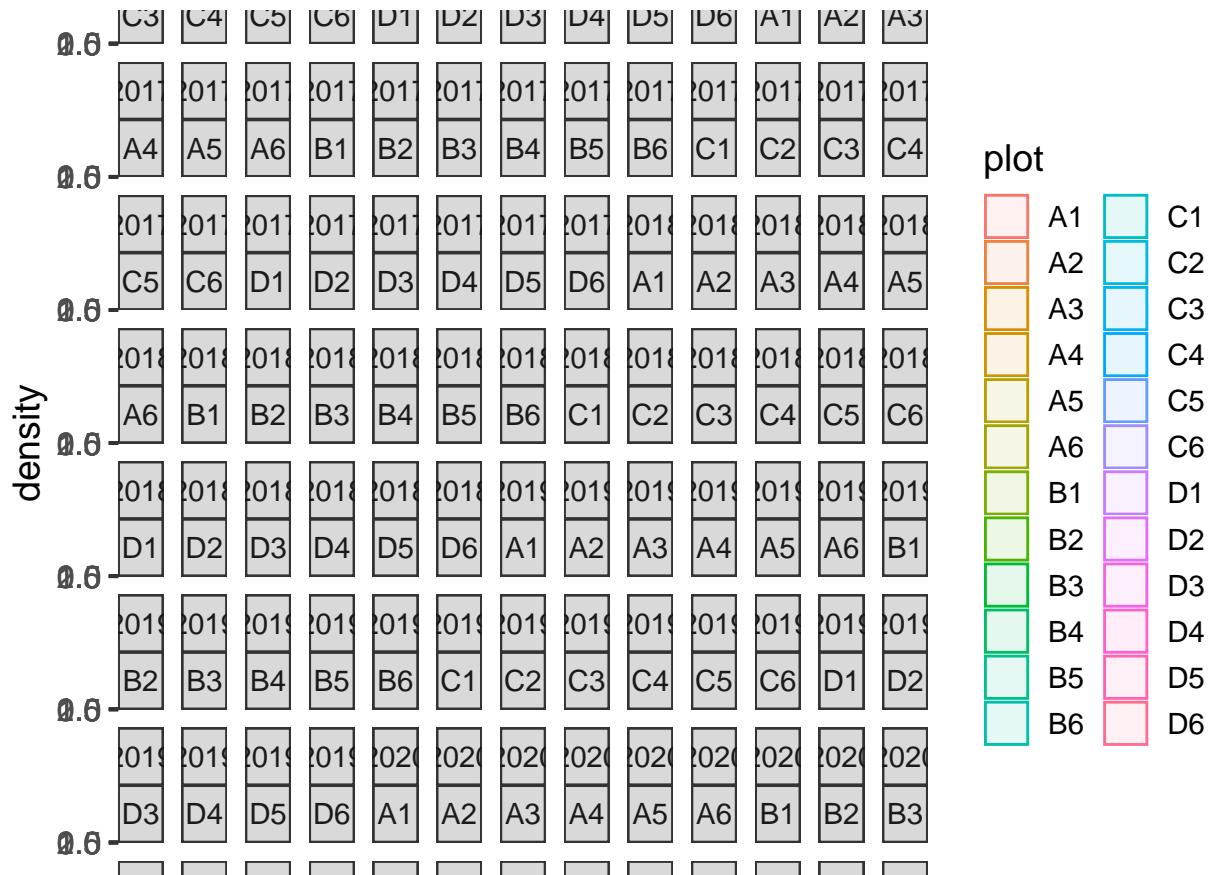
```
ggplot(kbs_flwr_plot_growthhabit, aes(julian_median, fill = plot, color = plot)) +
  geom_density(alpha = 0.1)
```



```
ggplot(kbs_flwr_plot_growthhabit, aes(julian_median, fill = plot, color = plot)) +  
  geom_density(alpha = 0.1) + facet_wrap(~year)
```

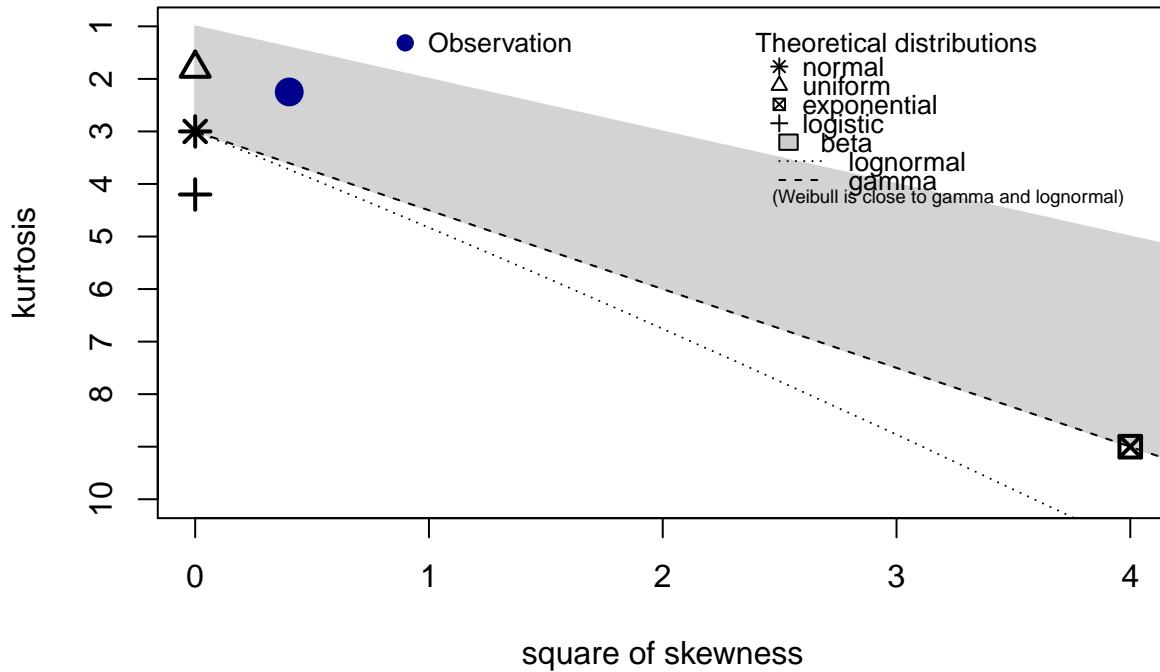


```
ggplot(kbs_flwr_plot_growthhabit, aes(julian_median, fill = plot, color = plot)) +
  geom_density(alpha = 0.1) + facet_wrap(~year + plot)
```



```
# Exploring distributions for these data:  
descdist(kbs_flwr_plot_growthhabit$julian_median, discrete = FALSE)
```

## Cullen and Frey graph



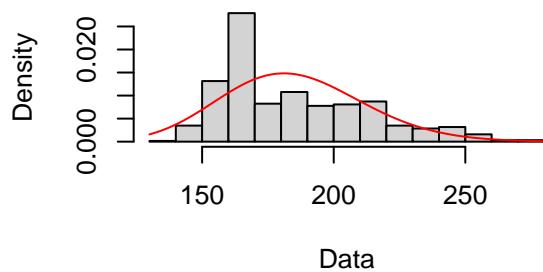
```

## summary statistics
## -----
## min: 124   max: 277
## median: 173.375
## mean: 188.2971
## estimated sd: 32.24529
## estimated skewness: 0.6342817
## estimated kurtosis: 2.248348

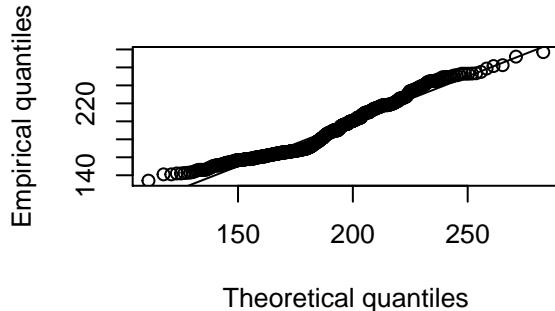
# Gamma distribution
fit.gamma <- fitdist(umbs_flwr_plot_growthhabit$julian_median, "gamma")
plot(fit.gamma)

```

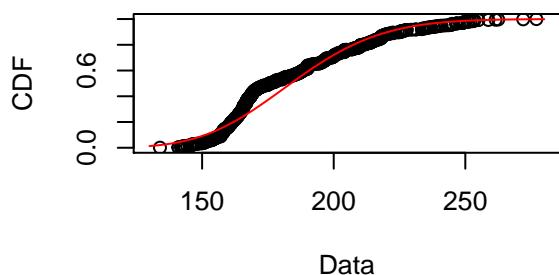
**Empirical and theoretical dens.**



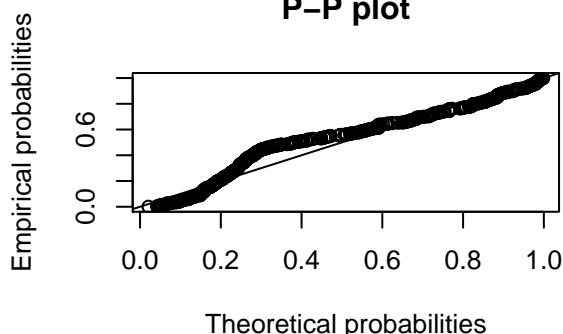
**Q-Q plot**



**Empirical and theoretical CDFs**

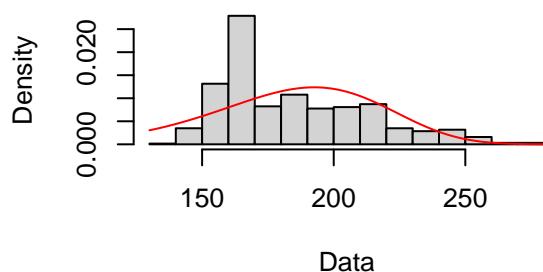


**P-P plot**

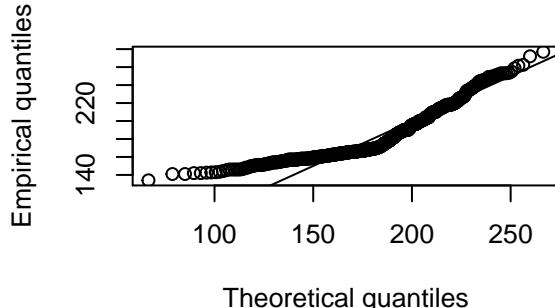


```
# Weibull distribution
fit.weibull <- fitdist(umbs_flwr_plot_growthhabit$julian_median, "weibull")
plot(fit.weibull)
```

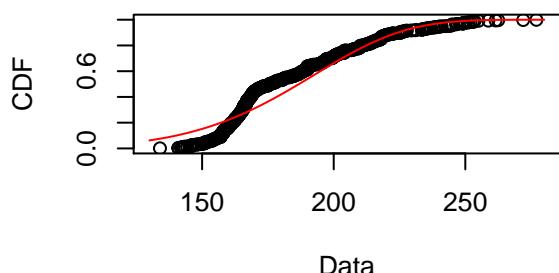
**Empirical and theoretical dens.**



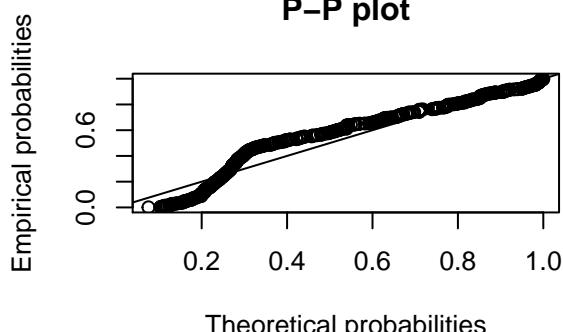
**Q-Q plot**



**Empirical and theoretical CDFs**

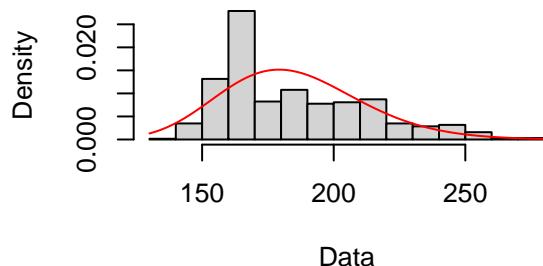


**P-P plot**

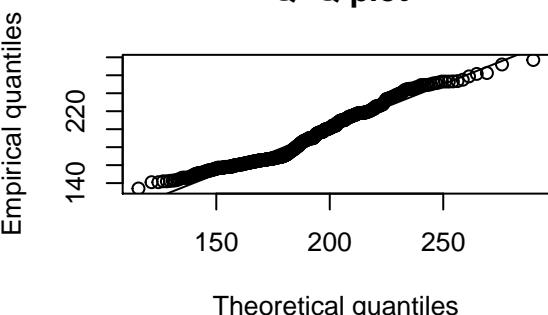


```
# Lognormal distribution
fit.ln <- fitdist(umbs_flwr_plot_growthhabit$julian_median, "lnorm")
plot(fit.ln)
```

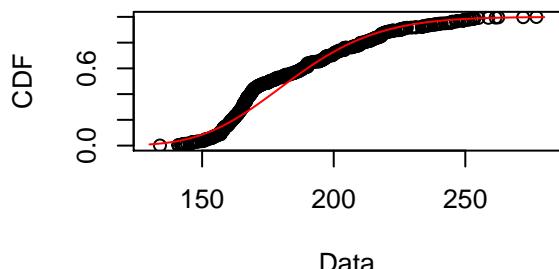
**Empirical and theoretical dens.**



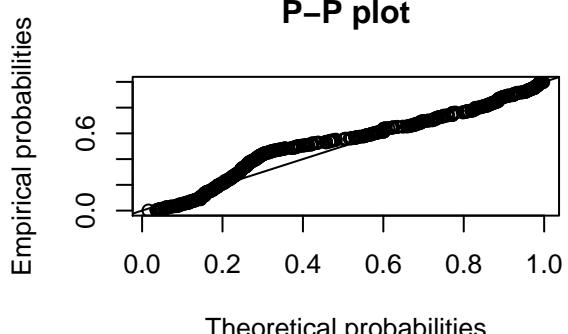
**Q-Q plot**



**Empirical and theoretical CDFs**

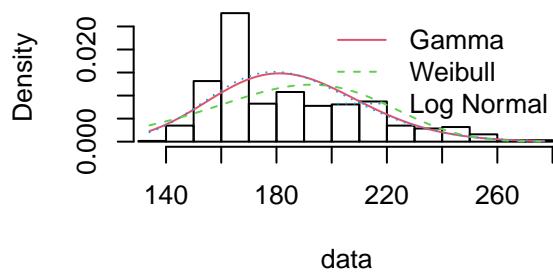


**P-P plot**

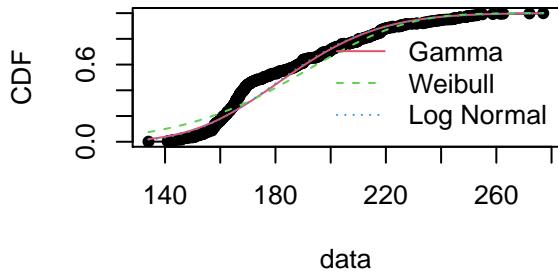


```
par(mfrow = c(2, 2))
plot.legend <- c("Gamma", "Weibull", "Log Normal")
denscomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
cdfcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
qqcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
ppcomp(list(fit.gamma, fit.weibull, fit.ln), legendtext = plot.legend)
```

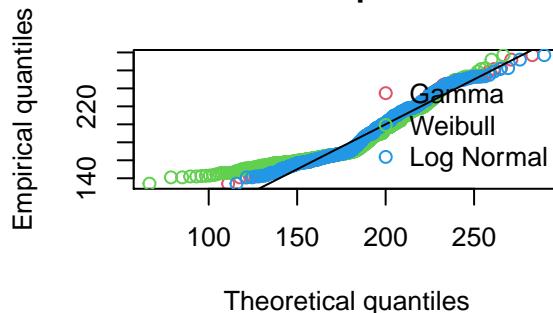
### Histogram and theoretical densities



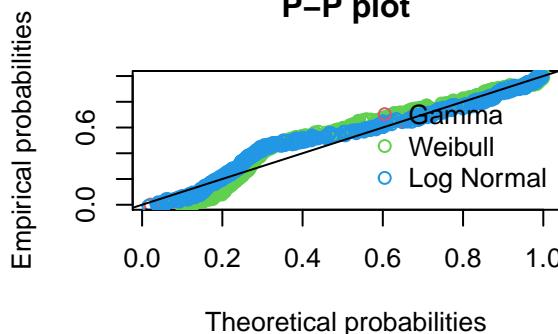
### Empirical and theoretical CDFs



### Q-Q plot



### P-P plot



```
# Goodness of fit comparisons across fits (can't include the sqrt normal bc it
# becomes diff response values)
gofstat(list(fit.gamma, fit.weibull, fit.ln), fitnames = c("Gamma", "Weibull", "Log Normal"))
```

```
## Goodness-of-fit statistics
##                                     Gamma   Weibull Log Normal
## Kolmogorov-Smirnov statistic 0.1457577 0.1392539 0.1420043
## Cramer-von Mises statistic  2.4904042 3.5666090 2.2843640
## Anderson-Darling statistic 13.8740882 21.4965609 12.5741740
##
## Goodness-of-fit criteria
##                                     Gamma   Weibull Log Normal
## Akaike's Information Criterion 5952.729 6095.992 5934.791
## Bayesian Information Criterion 5961.624 6104.887 5943.686

# Log normal looks like it's the best fit
```

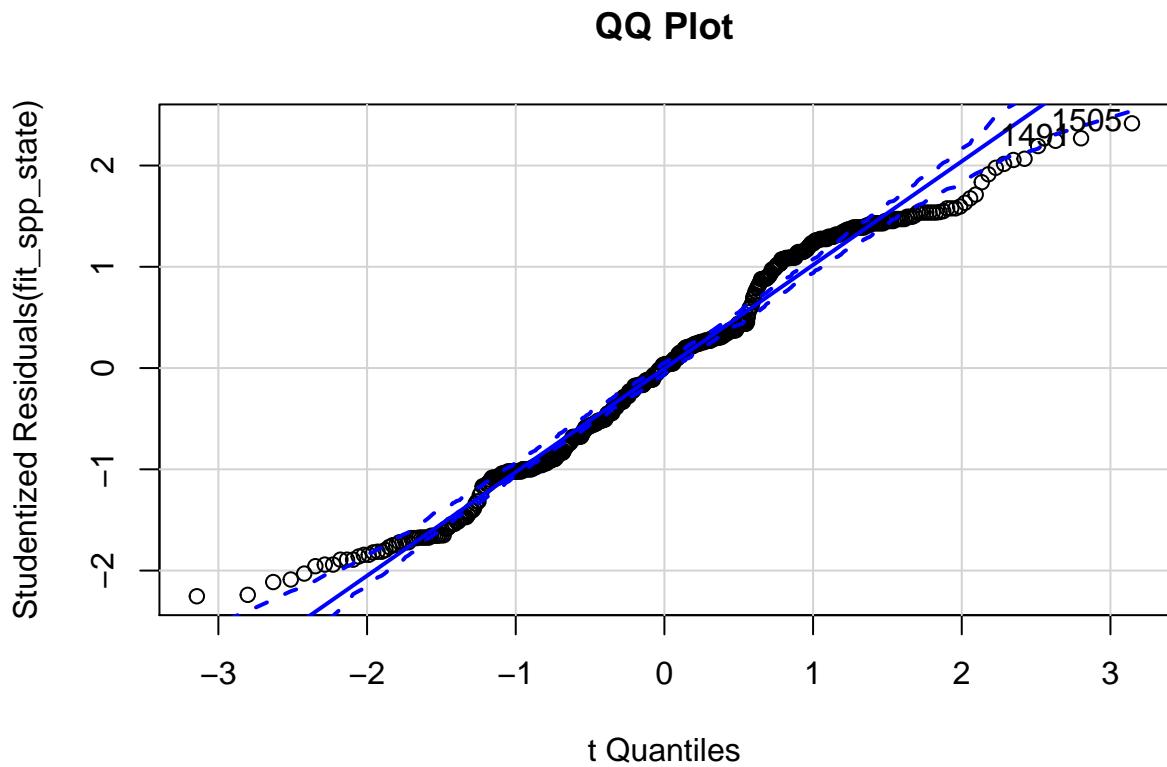
Leverage plots and detecting Outliers. <https://www.statmethods.net/stats/rdiagnostics.html>

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://daviddalpiaz.github.io/appliedstats/model-diagnostics.html>

```
# species level data UMBS State-only model
fit_spp_state <- lm(log(julian_median) ~ state, data = umbs_flwr_spp)
outlierTest(fit_spp_state) # no outliers

## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##      rstudent unadjusted p-value Bonferroni p
## 1505 2.416487          0.015985         NA

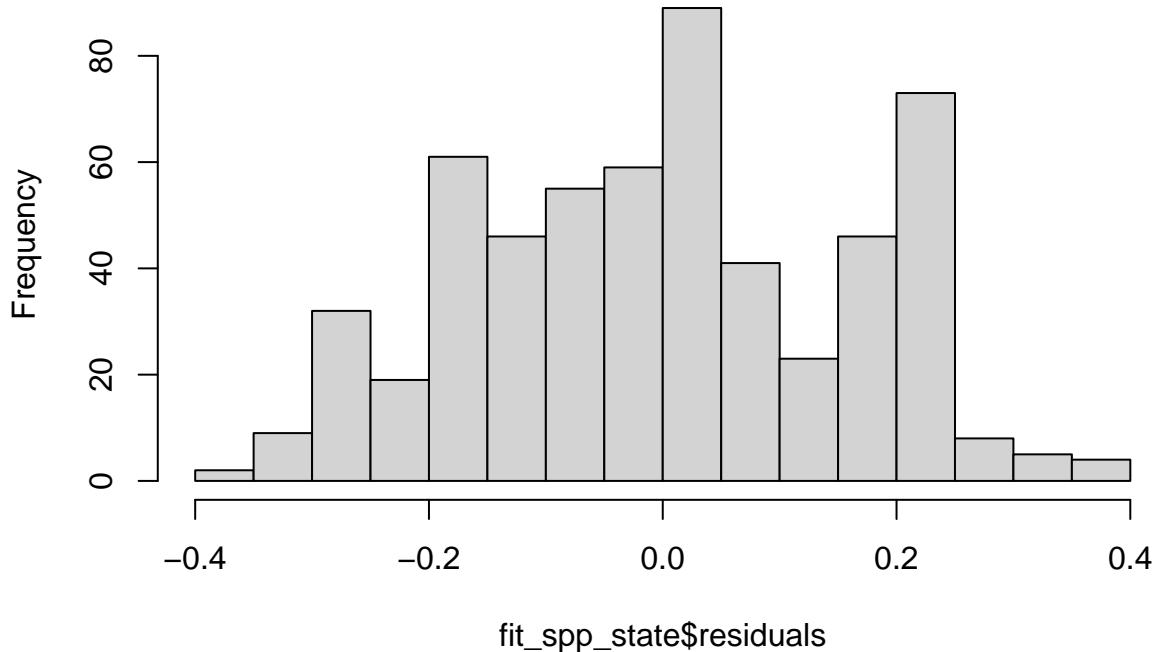
qqPlot(fit_spp_state, main = "QQ Plot")
```



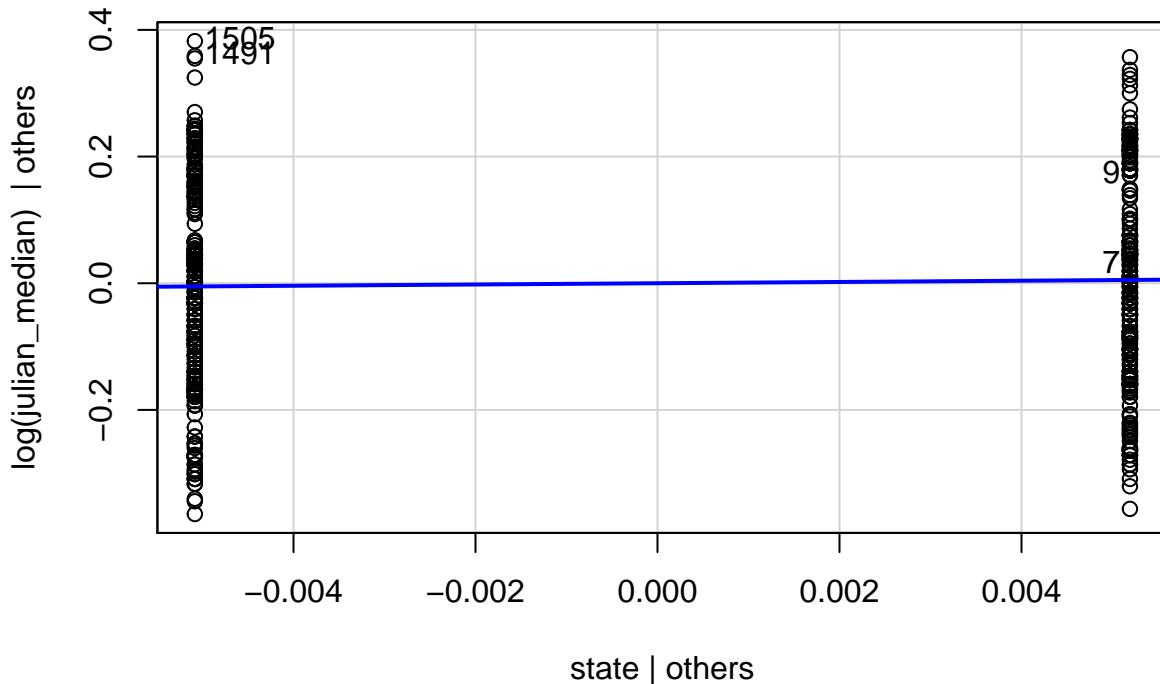
```
## 1491 1505
## 502 507
```

```
hist(fit_spp_state$residuals)
```

**Histogram of fit\_spp\_state\$residuals**



```
leveragePlots(fit_spp_state)
```



```
ols_test_normality(fit_spp_state)
```

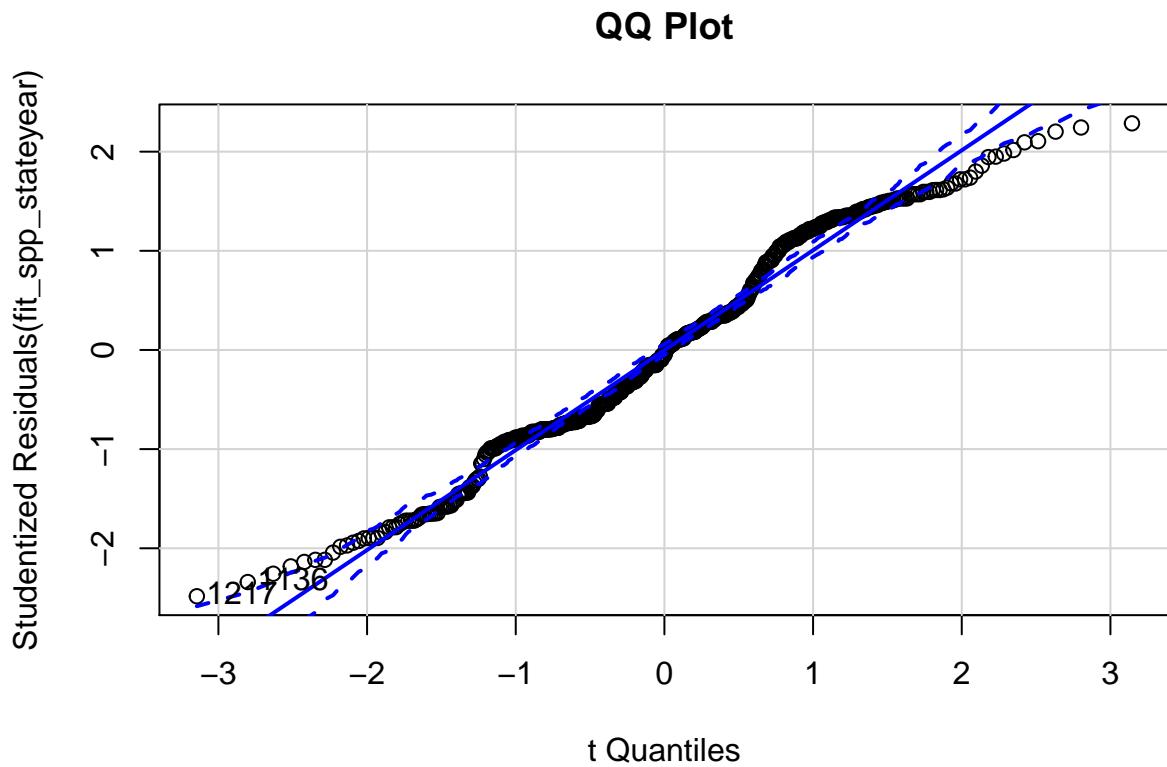
```
## Warning in ks.test(y, "pnorm", mean(y), sd(y)): ties should not be present for
## the Kolmogorov-Smirnov test
```

```
## -----
##      Test          Statistic       pvalue
## -----
## Shapiro-Wilk      0.9777       0.0000
## Kolmogorov-Smirnov 0.0747       0.0034
## Cramer-von Mises 133.5002      0.0000
## Anderson-Darling   3.6609       0.0000
## -----
```

```
# UMBS State and year model
fit_spp_stateyear <- lm(log(julian_median) ~ state + year_factor, data = umbs_flwr_spp)
outlierTest(fit_spp_stateyear) # no outliers
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##           rstudent unadjusted p-value Bonferroni p
## 1217    -2.483867          0.013284        NA
```

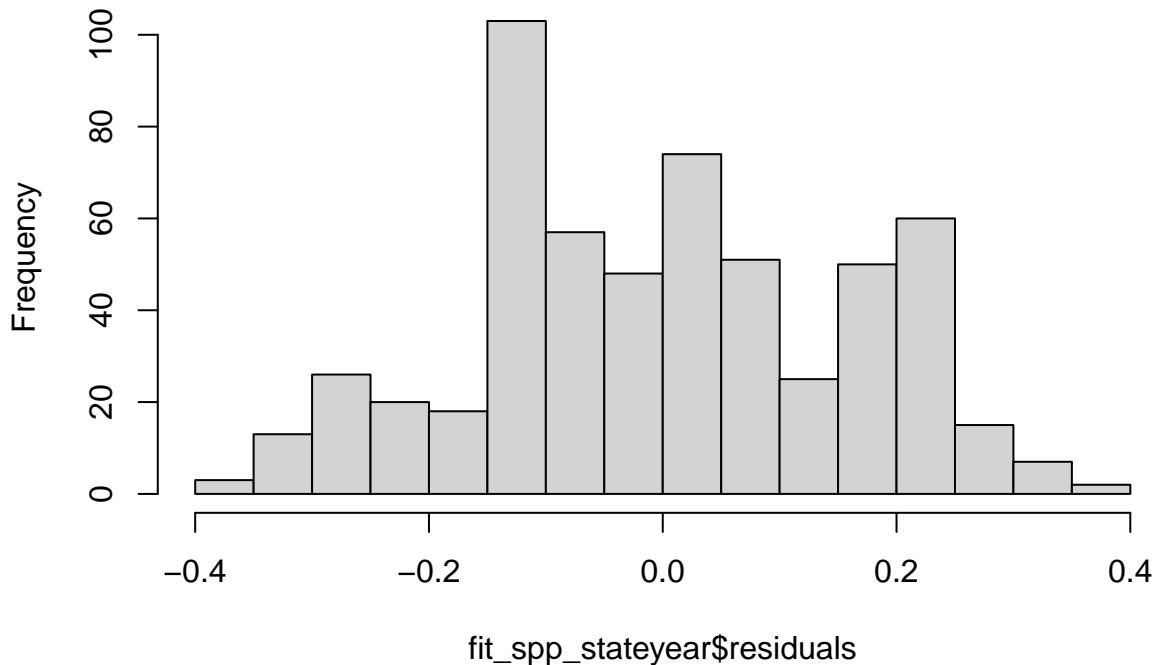
```
qqPlot(fit_spp_stateyear, main = "QQ Plot")
```



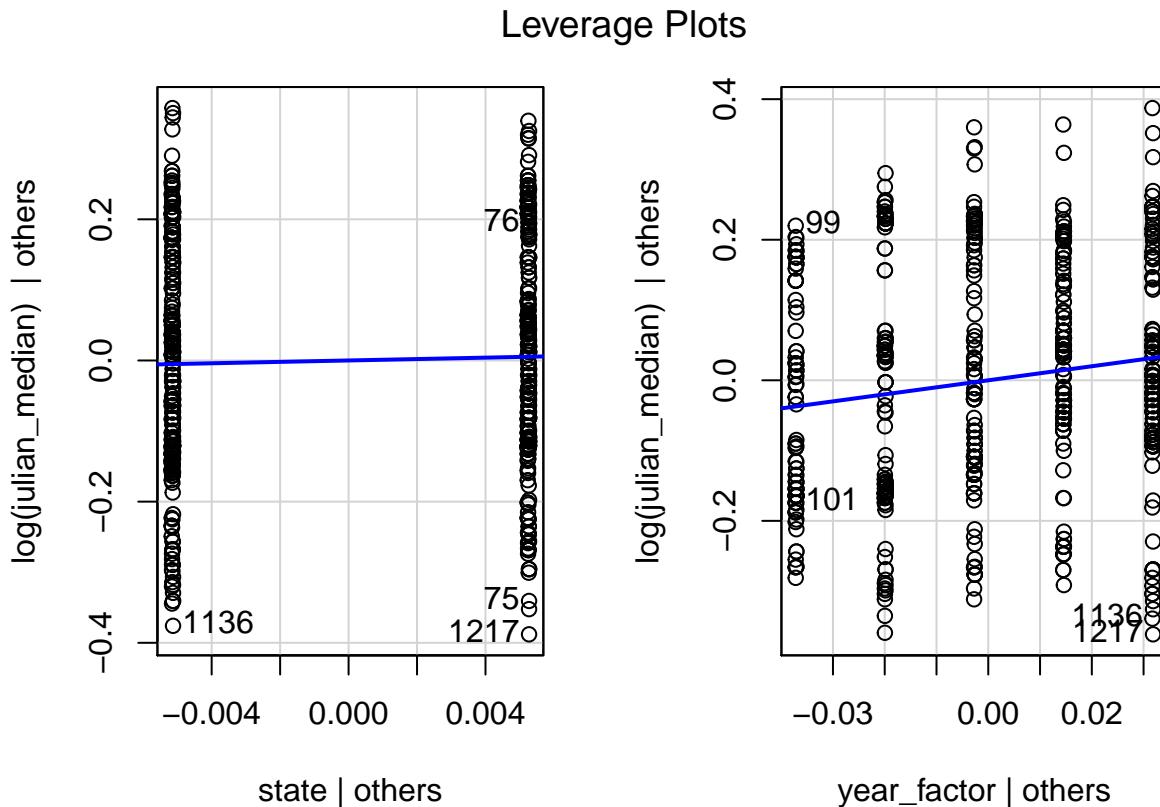
```
## 1136 1217
## 392 414
```

```
hist(fit_spp_stateyear$residuals)
```

**Histogram of fit\_spp\_stateyear\$residuals**



```
leveragePlots(fit_spp_stateyear)
```



```
ols_test_normality(fit_spp_stateyear)
```

```
## Warning in ks.test(y, "pnorm", mean(y), sd(y)): ties should not be present for
## the Kolmogorov-Smirnov test
```

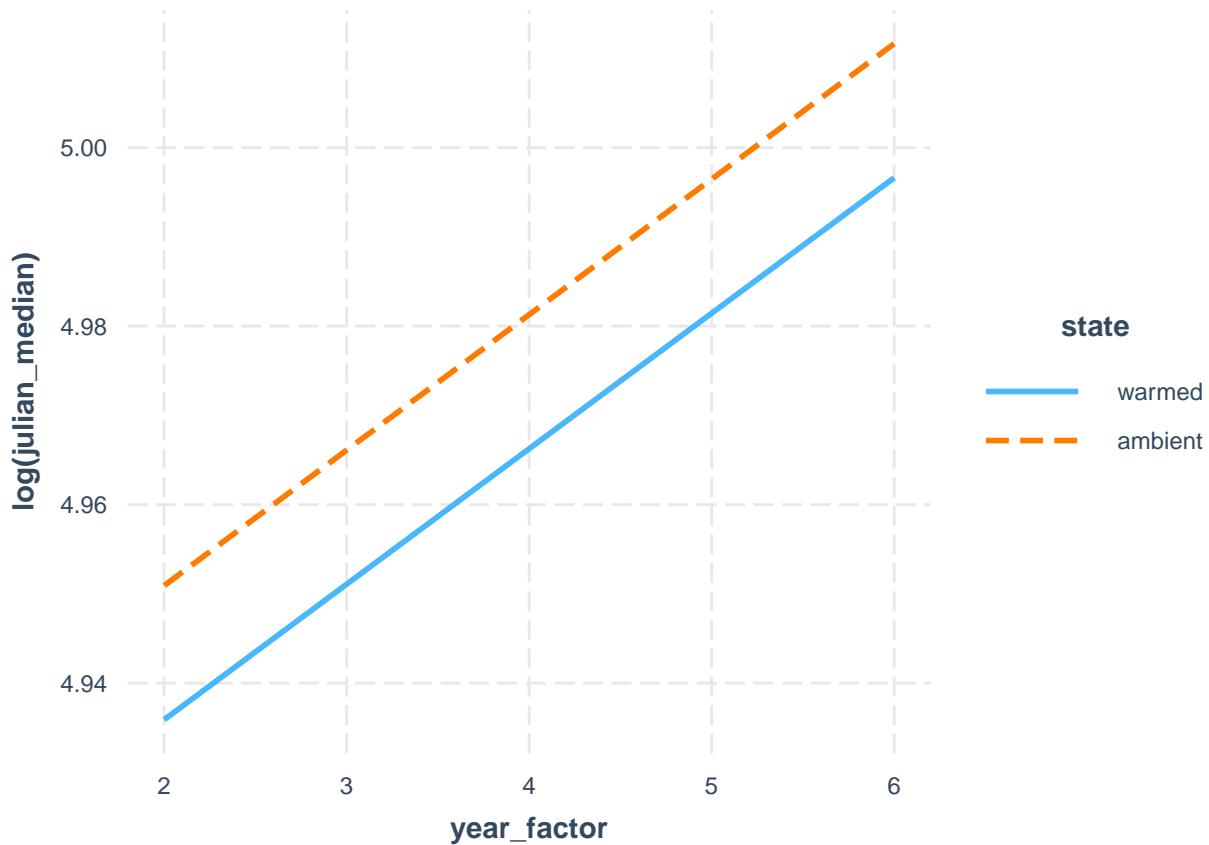
```
## -----
##      Test       Statistic     pvalue
## -----
## Shapiro-Wilk      0.9794    0.0000
## Kolmogorov-Smirnov 0.0715    0.0057
## Cramer-von Mises   133.7468   0.0000
## Anderson-Darling     3.8514    0.0000
## -----
```

```
# 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(log(julian_median) ~ state + year_factor + species, data = umbs_flwr_spp)
interact_plot(fit3, pred = year_factor, modx = state)
```

```
## Using data umbs_flwr_spp from global environment. This could cause
## incorrect results if umbs_flwr_spp has been altered since the model was
## fit. You can manually provide the data to the "data =" argument.
```

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



```
fit4 <- lm(log(julian_median) ~ state * year_factor + species, data = umbs_flwr_spp)
interact_plot(fit4, pred = year_factor, modx = state, mod2 = species)
```

```
## Using data umbs_flwr_spp from global environment. This could cause
## incorrect results if umbs_flwr_spp has been altered since the model was
## fit. You can manually provide the data to the "data =" argument.
```

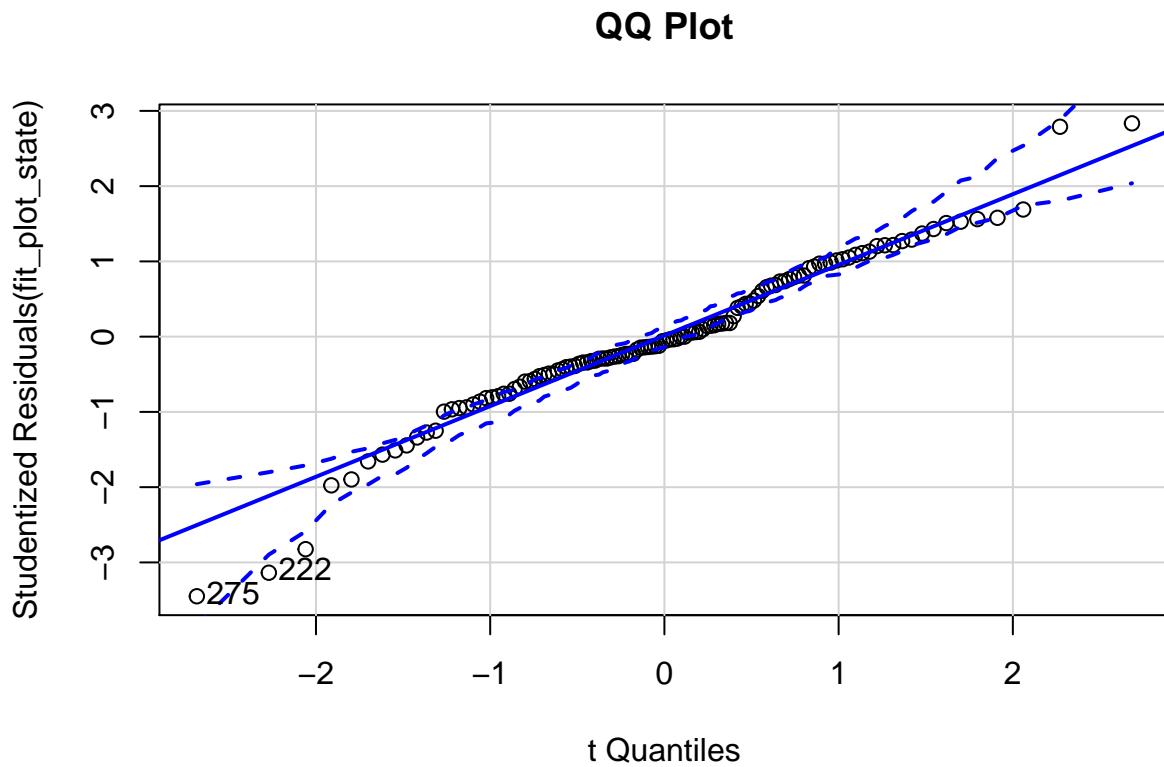
```
## Warning: year_factor and state and species are not included in an interaction with
## one another in the model.
```



```
# Plot level data UMBS State-only model
fit_plot_state <- lm(log(julian_median) ~ state, data = umbs_flwr_plot)
outlierTest(fit_plot_state) # no outliers
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##      rstudent unadjusted p-value Bonferroni p
## 275 -3.449752          0.00078104     0.093725
```

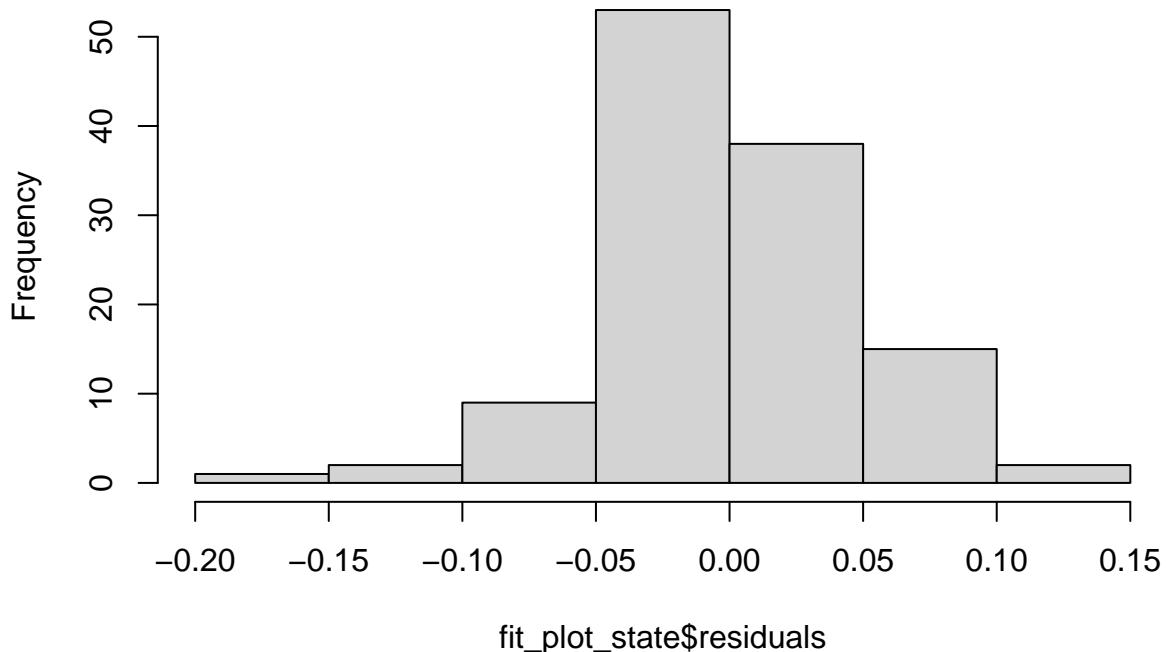
```
qqPlot(fit_plot_state, main = "QQ Plot")
```



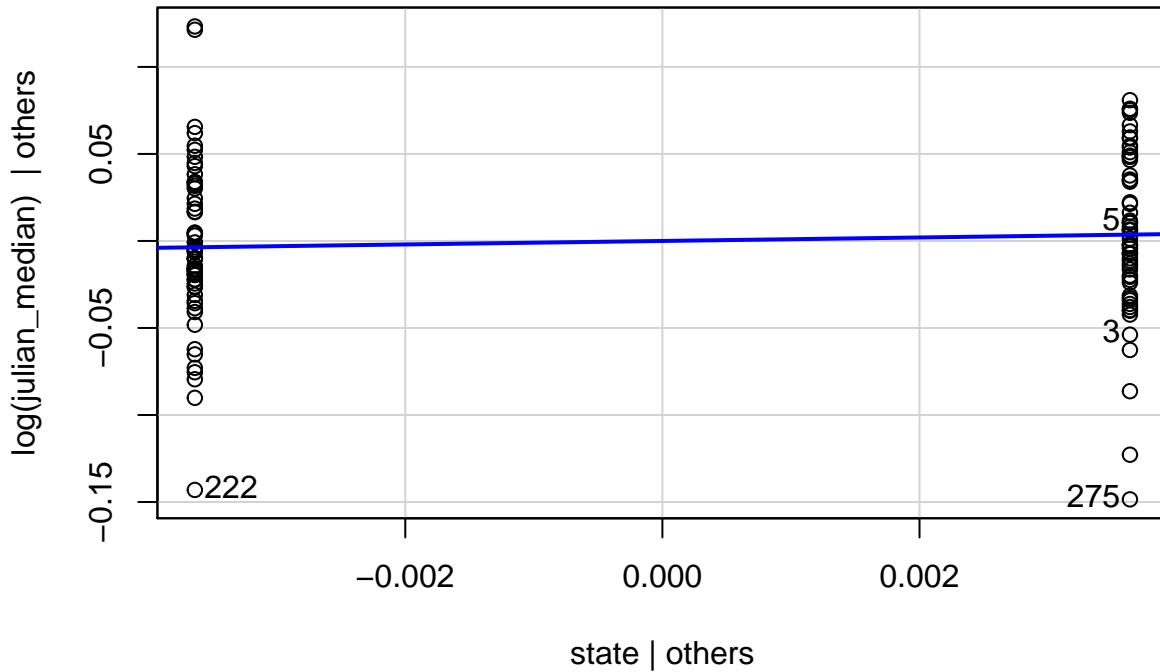
```
## 222 275
## 86 106
```

```
hist(fit_plot_state$residuals)
```

**Histogram of fit\_plot\_state\$residuals**



```
leveragePlots(fit_plot_state)
```



```
ols_test_normality(fit_plot_state) # looks ok besides Kolmogorov-Smirnov test
```

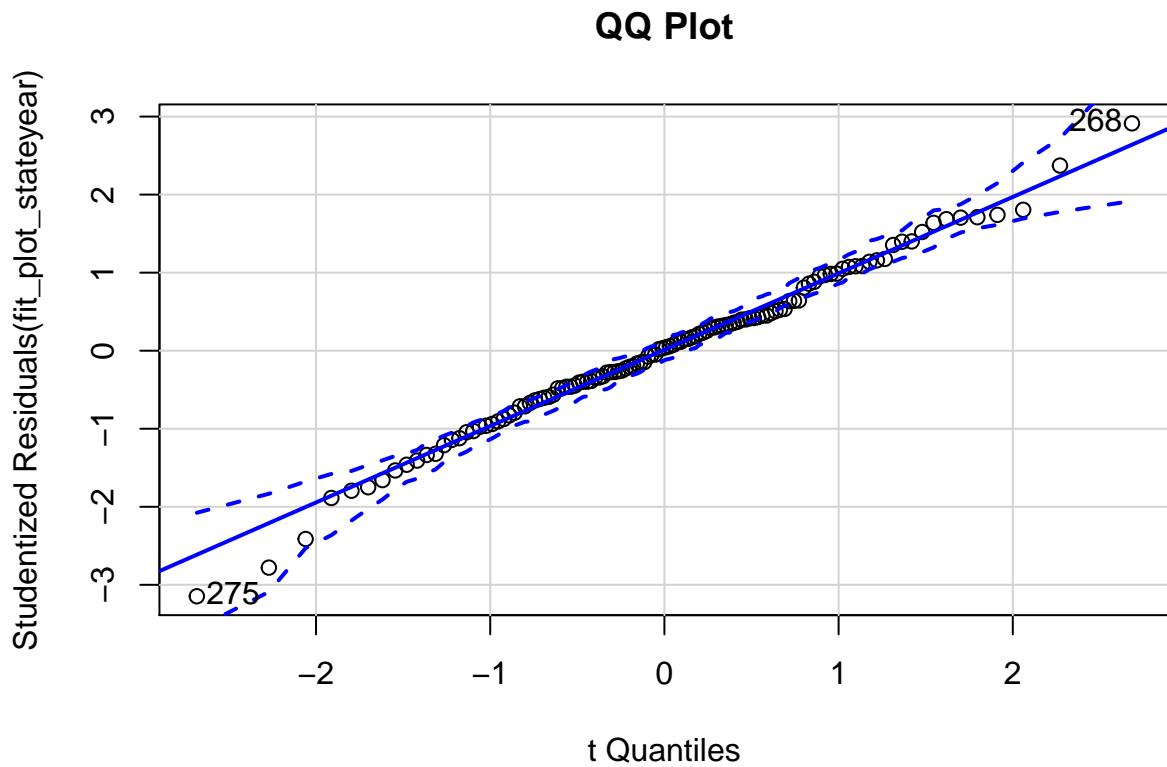
```
## Warning in ks.test(y, "pnorm", mean(y), sd(y)): ties should not be present for
## the Kolmogorov-Smirnov test
```

```
## -----
##      Test          Statistic       pvalue
## -----
## Shapiro-Wilk      0.9733     0.0172
## Kolmogorov-Smirnov 0.0781     0.4577
## Cramer-von Mises  36.5315    0.0000
## Anderson-Darling   0.8575     0.0268
## -----
```

```
# UMBS State and year model
fit_plot_stateyear <- lm(log(julian_median) ~ state + year_factor, data = umbs_flwr_plot)
outlierTest(fit_plot_stateyear) # no outliers
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##      rstudent unadjusted p-value Bonferroni p
## 275 -3.146336      0.0021011     0.25213
```

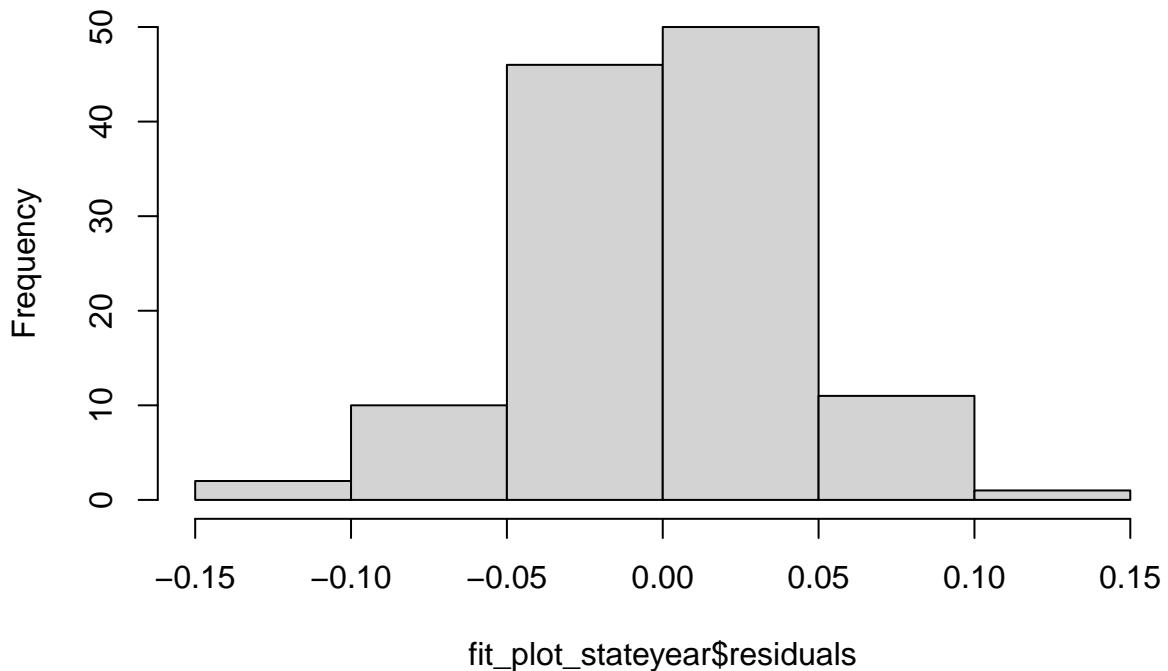
```
qqPlot(fit_plot_stateyear, main = "QQ Plot")
```



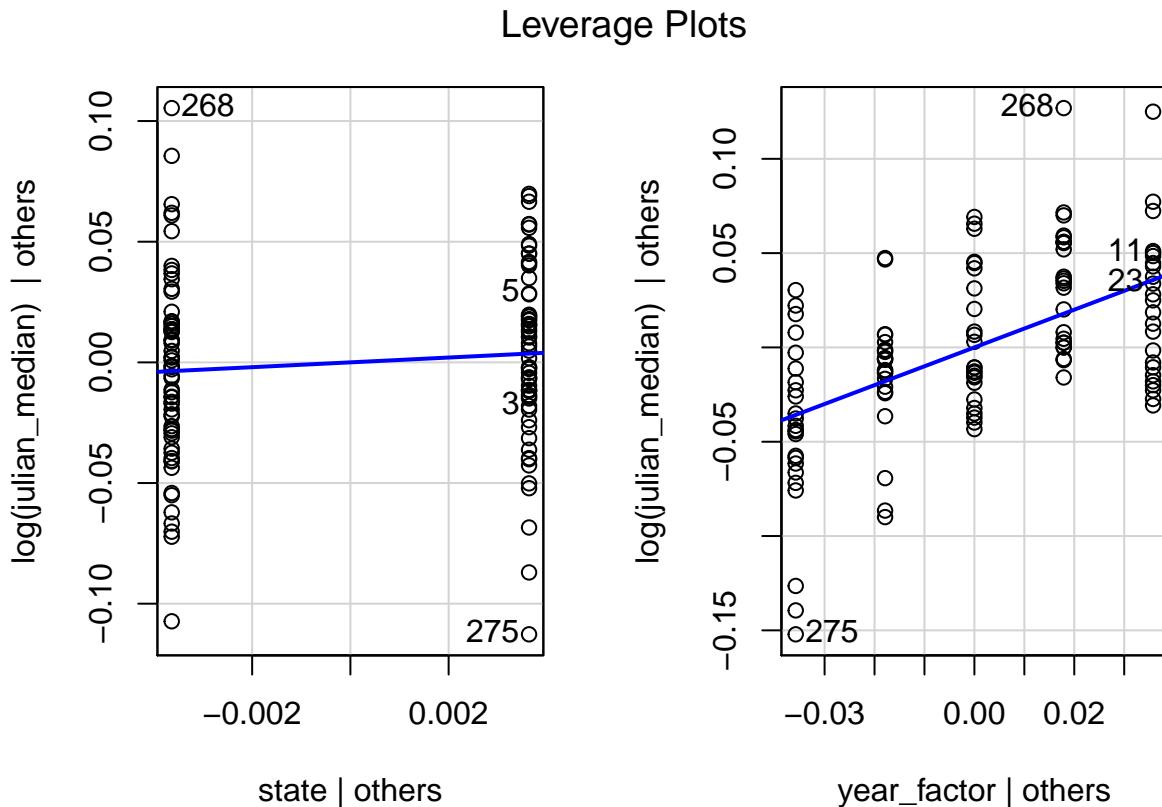
```
## 268 275
## 104 106
```

```
hist(fit_plot_stateyear$residuals)
```

**Histogram of fit\_plot\_stateyear\$residuals**



```
leveragePlots(fit_plot_stateyear)
```



```
ols_test_normality(fit_plot_stateyear) # all good
```

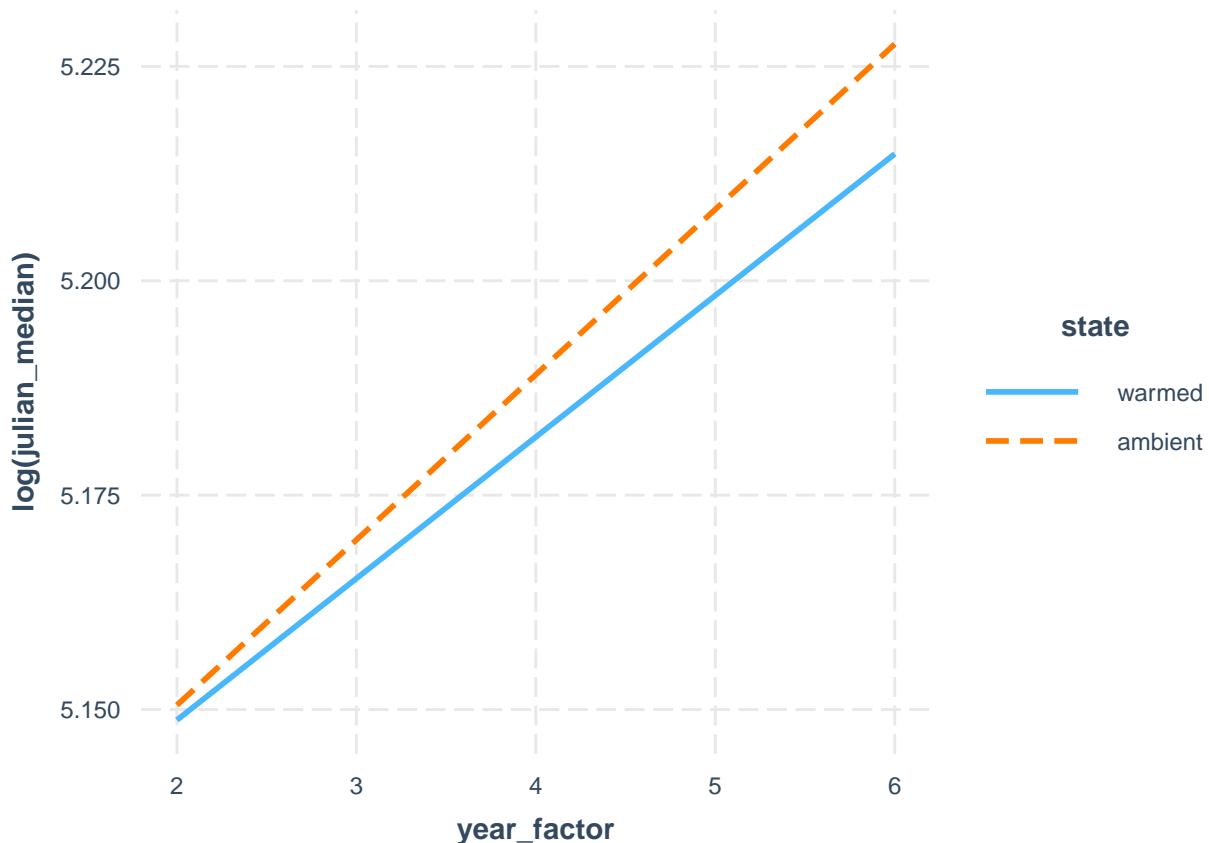
```
## Warning in ks.test(y, "pnorm", mean(y), sd(y)): ties should not be present for
## the Kolmogorov-Smirnov test
```

```
## -----
##      Test       Statistic     pvalue
## -----
## Shapiro-Wilk      0.9929    0.8000
## Kolmogorov-Smirnov 0.0546    0.8662
## Cramer-von Mises   37.0156   0.0000
## Anderson-Darling    0.2734    0.6606
## -----
```

```
# 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(log(julian_median) ~ state * year_factor, data = umbs_flwr_plot)
interact_plot(fit3, pred = year_factor, modx = state)
```

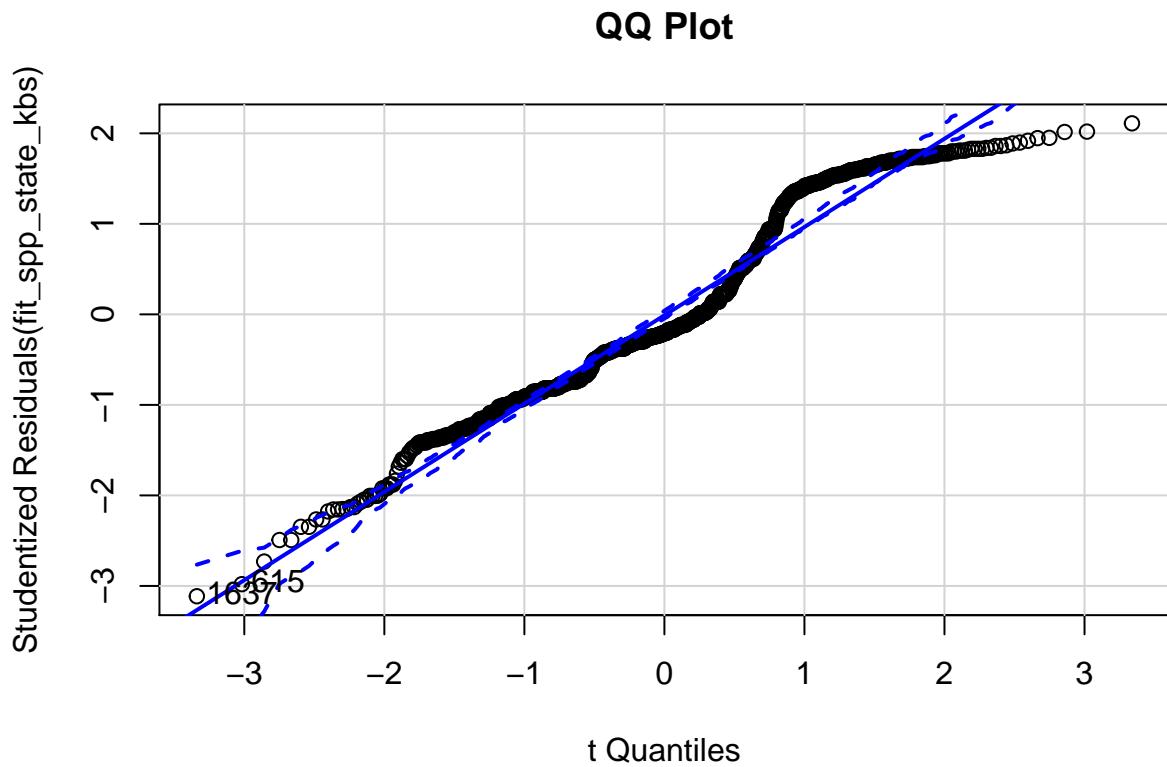
```
## Using data umbs_flwr_plot from global environment. This could cause
## incorrect results if umbs_flwr_plot has been altered since the model was
## fit. You can manually provide the data to the "data =" argument.
```



```
# species level data KBS State-only model
fit_spp_state_kbs <- lm(log(julian_median) ~ state, data = kbs_flwr_spp)
outlierTest(fit_spp_state_kbs) # no outliers
```

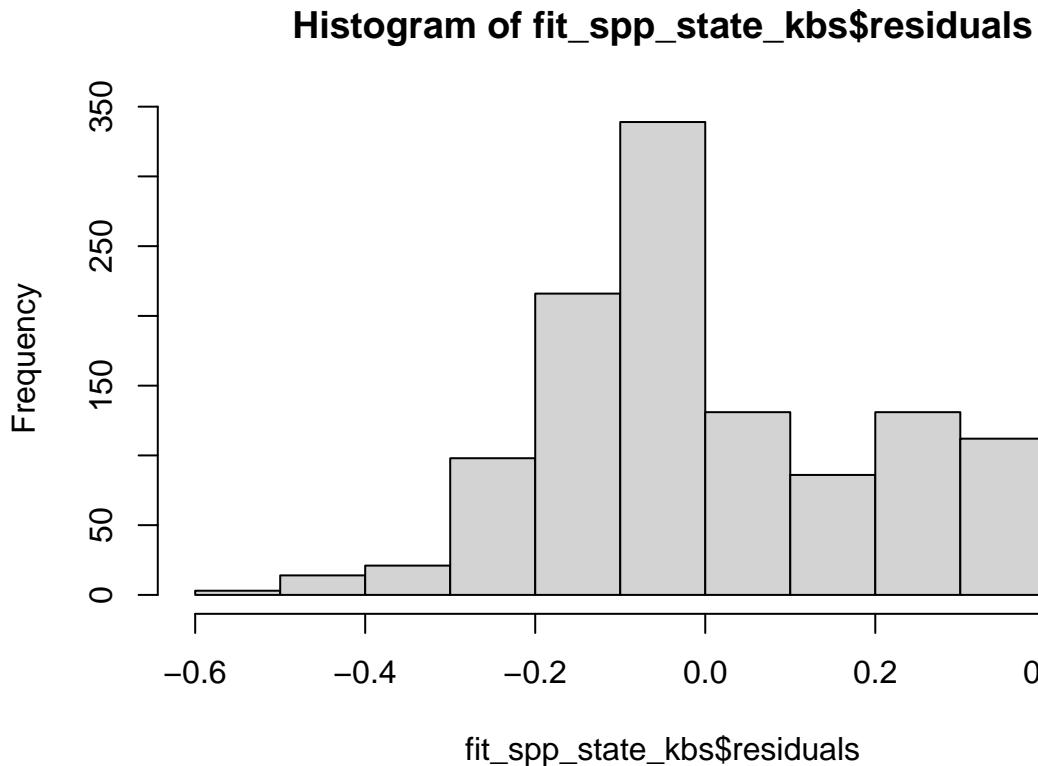
```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##      rstUDENT unadjusted p-value Bonferroni p
## 1637 -3.11402          0.0018912        NA

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

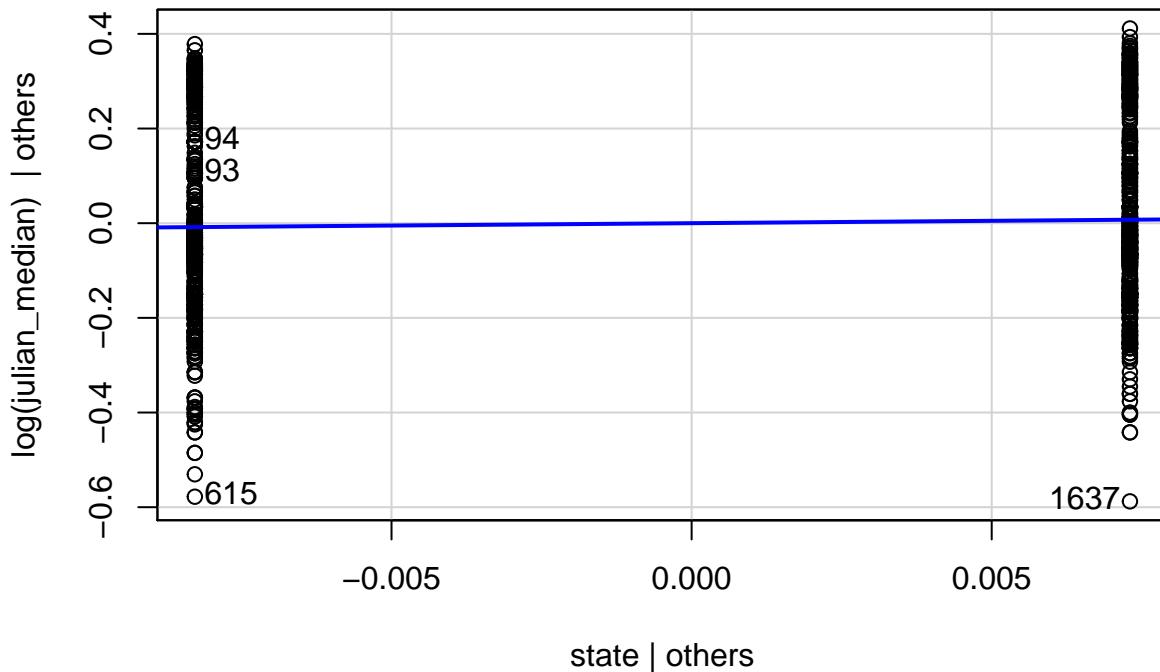


```
##   615 1637
##  383 1077
```

```
hist(fit_spp_state_kbs$residuals)
```



```
leveragePlots(fit_spp_state_kbs)
```



```
ols_test_normality(fit_spp_state_kbs)
```

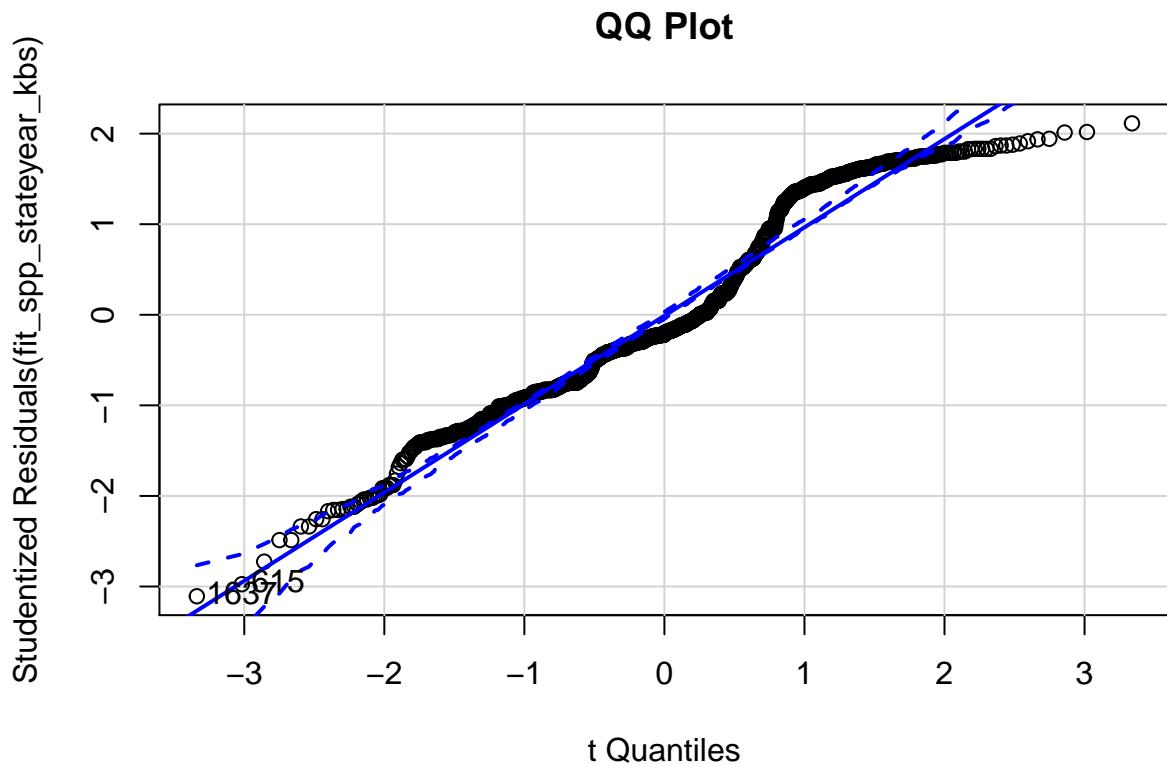
```
## Warning in ks.test(y, "pnorm", mean(y), sd(y)): ties should not be present for
## the Kolmogorov-Smirnov test
```

```
## -----
##           Test          Statistic      pvalue
## -----
## Shapiro-Wilk       0.9563      0.0000
## Kolmogorov-Smirnov 0.1089      0.0000
## Cramer-von Mises   251.7392     0.0000
## Anderson-Darling    19.9905     0.0000
## -----
```

```
# KBS State and year model
fit_spp_stateyear_kbs <- lm(log(julian_median) ~ state + year_factor, data = kbs_flwr_spp)
outlierTest(fit_spp_stateyear_kbs) # no outliers
```

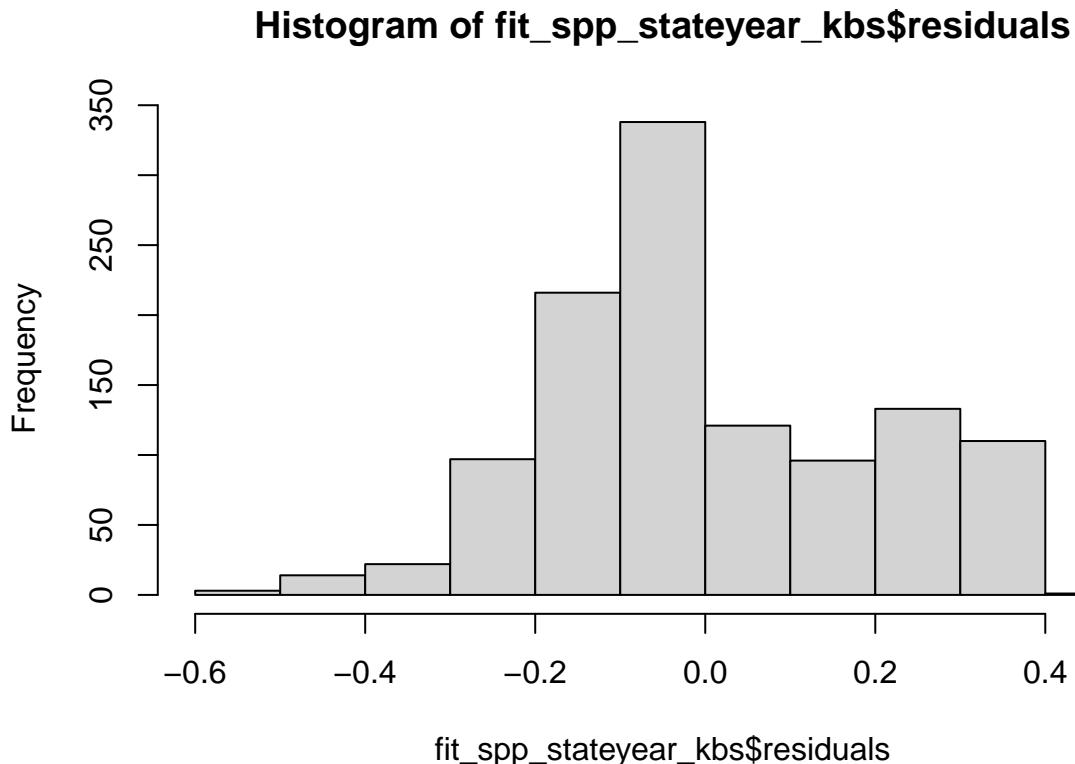
```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##           rstudent unadjusted p-value Bonferroni p
## 1637    -3.10761          0.0019324        NA
```

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



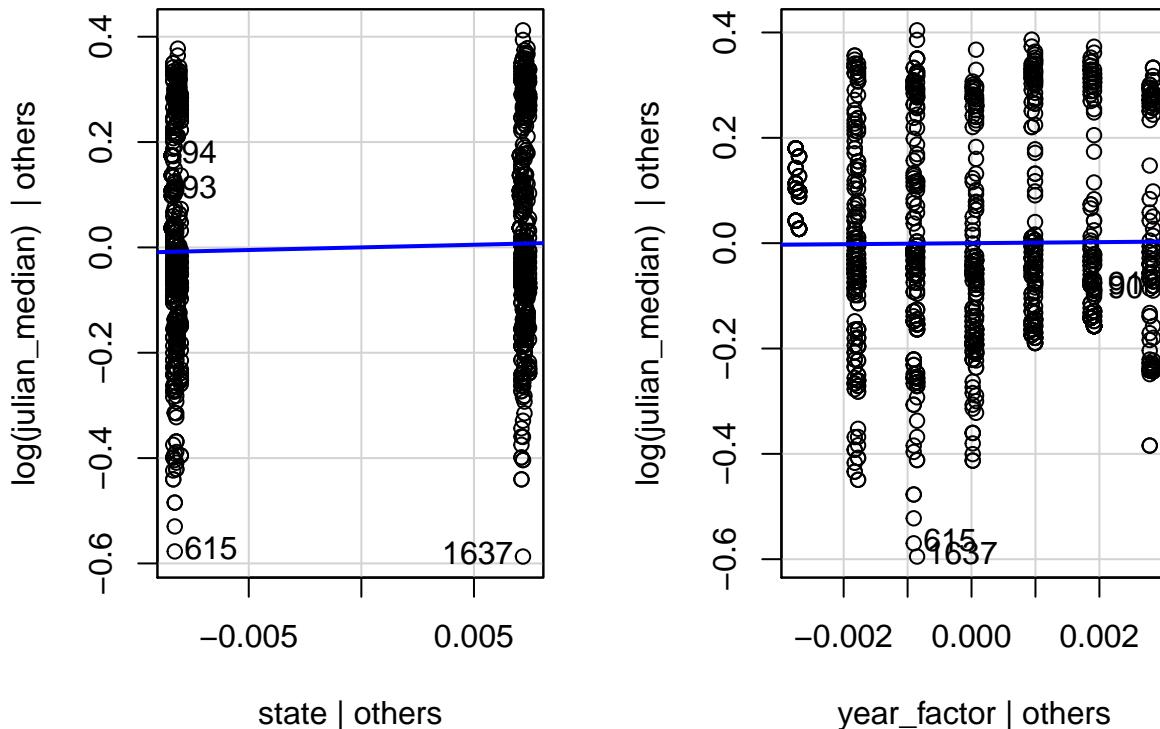
```
##   615 1637
##   383 1077
```

```
hist(fit_spp_stateyear_kbs$residuals)
```



```
leveragePlots(fit_spp_stateyear_kbs)
```

## Leverage Plots



```
ols_test_normality(fit_spp_stateyear_kbs)
```

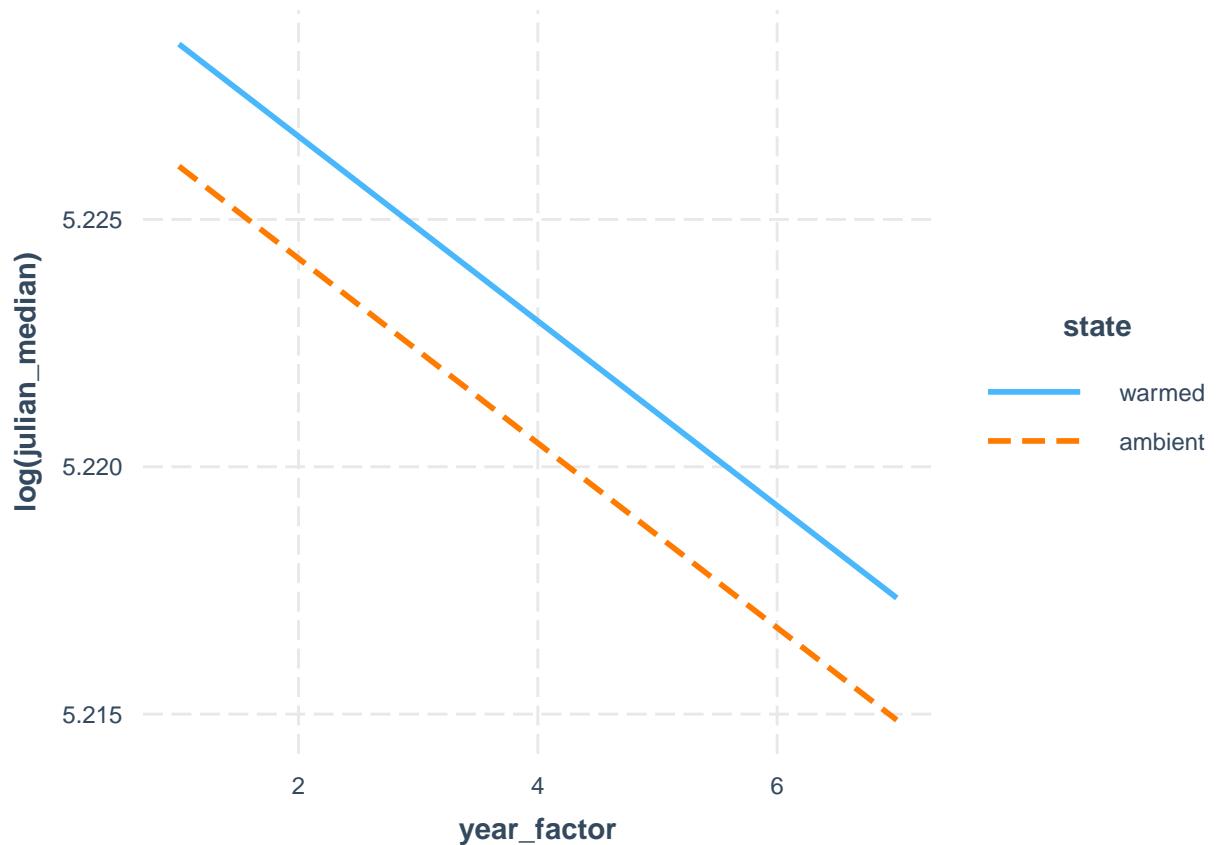
```
## Warning in ks.test(y, "pnorm", mean(y), sd(y)): ties should not be present for
## the Kolmogorov-Smirnov test
```

```
## -----
##           Test          Statistic       pvalue
## -----
## Shapiro-Wilk      0.9565       0.0000
## Kolmogorov-Smirnov 0.1101       0.0000
## Cramer-von Mises 251.3042      0.0000
## Anderson-Darling 19.7878       0.0000
## -----
```

```
# 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/
```

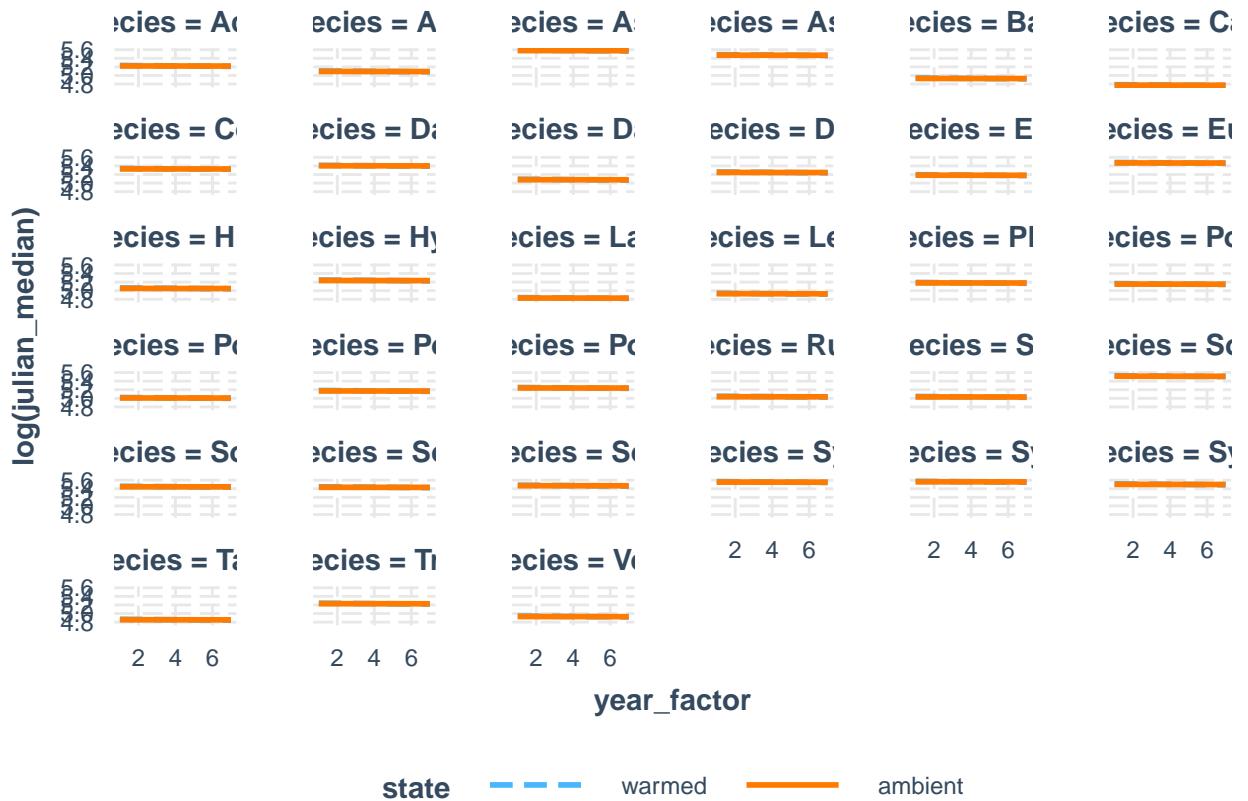
```
fit3_spp_kbs <- lm(log(julian_median) ~ state + year_factor + species, data = kbs_flwr_spp)
interact_plot(fit3_spp_kbs, pred = year_factor, modx = state, data = kbs_flwr_spp)
```

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



```
fit4_spp_kbs <- lm(log(julian_median) ~ state * year_factor + species, data = kbs_flwr_spp)
interact_plot(fit4_spp_kbs, pred = year_factor, modx = state, mod2 = species, data = kbs_flwr_spp)
```

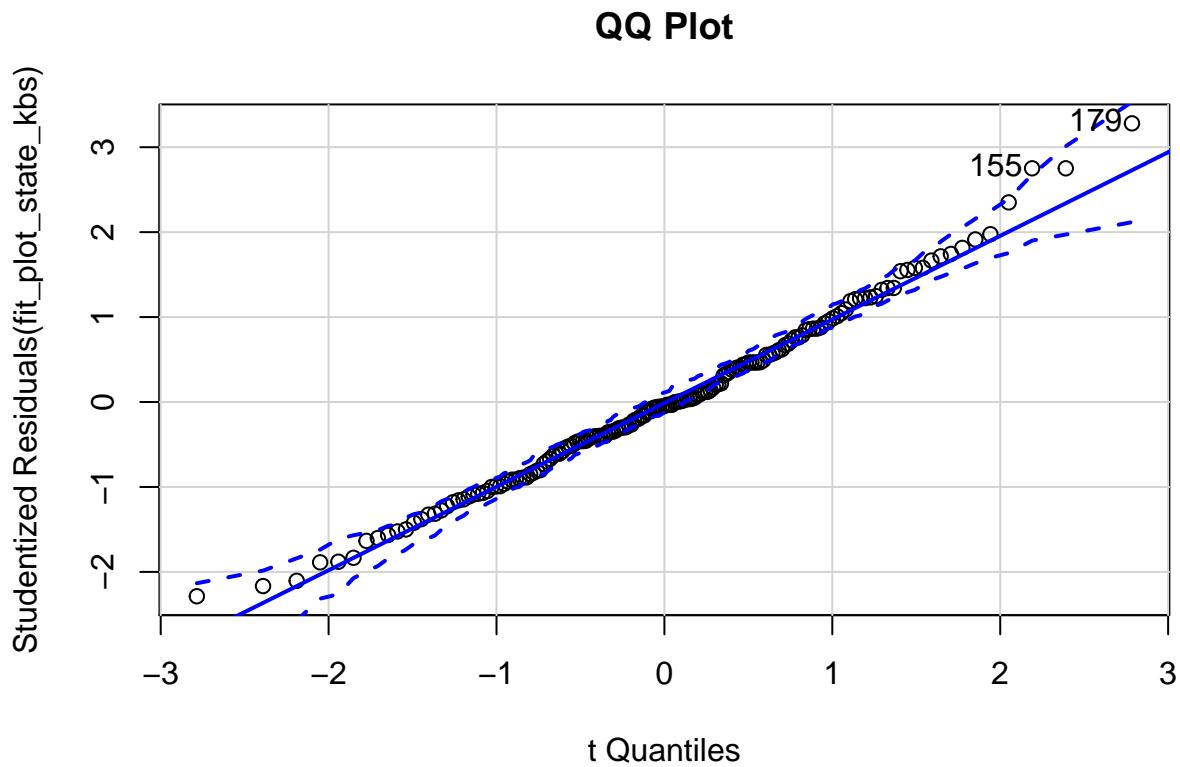
```
## Warning: year_factor and state and species are not included in an interaction with
## one another in the model.
```



```
# Plot level data KBS State-only model
fit_plot_state_kbs <- lm(log(julian_median) ~ state, data = kbs_flwr_plot)
outlierTest(fit_plot_state_kbs) # no outliers
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##      rstudent unadjusted p-value Bonferroni p
## 179 3.280787       0.0012646     0.21119
```

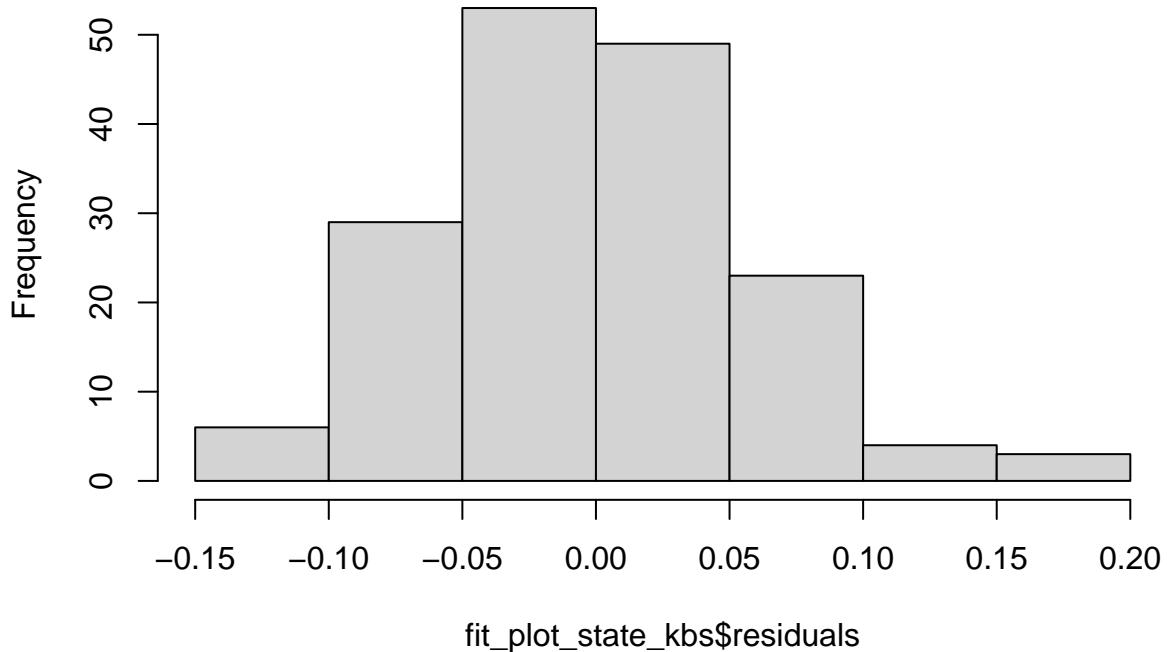
```
qqPlot(fit_plot_state_kbs, main = "QQ Plot")
```



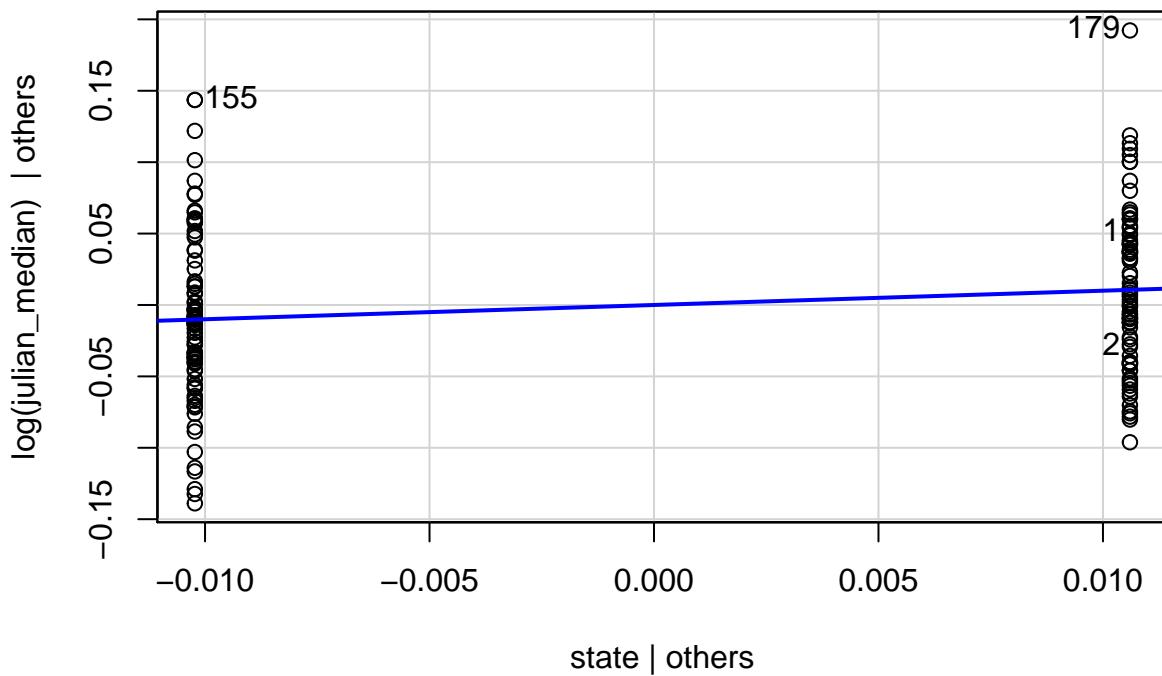
```
## 155 179
## 85 98
```

```
hist(fit_plot_state_kbs$residuals)
```

**Histogram of fit\_plot\_state\_kbs\$residuals**



```
leveragePlots(fit_plot_state_kbs)
```



```
ols_test_normality(fit_plot_state_kbs)
```

```
## Warning in ks.test(y, "pnorm", mean(y), sd(y)): ties should not be present for
## the Kolmogorov-Smirnov test
```

```
## -----
##      Test          Statistic       pvalue
## -----
## Shapiro-Wilk      0.9902      0.3090
## Kolmogorov-Smirnov 0.0567      0.6567
## Cramer-von Mises  49.4835      0.0000
## Anderson-Darling   0.3582      0.4490
## -----
```

```
# KBS State and year model
fit_plot_stateyear_kbs <- lm(log(julian_median) ~ state + year_factor, data = kbs_flwr_plot)
outlierTest(fit_plot_stateyear_kbs) # outliers
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##      rstudent unadjusted p-value Bonferroni p
## 179  3.517051        0.00056629     0.094005
```

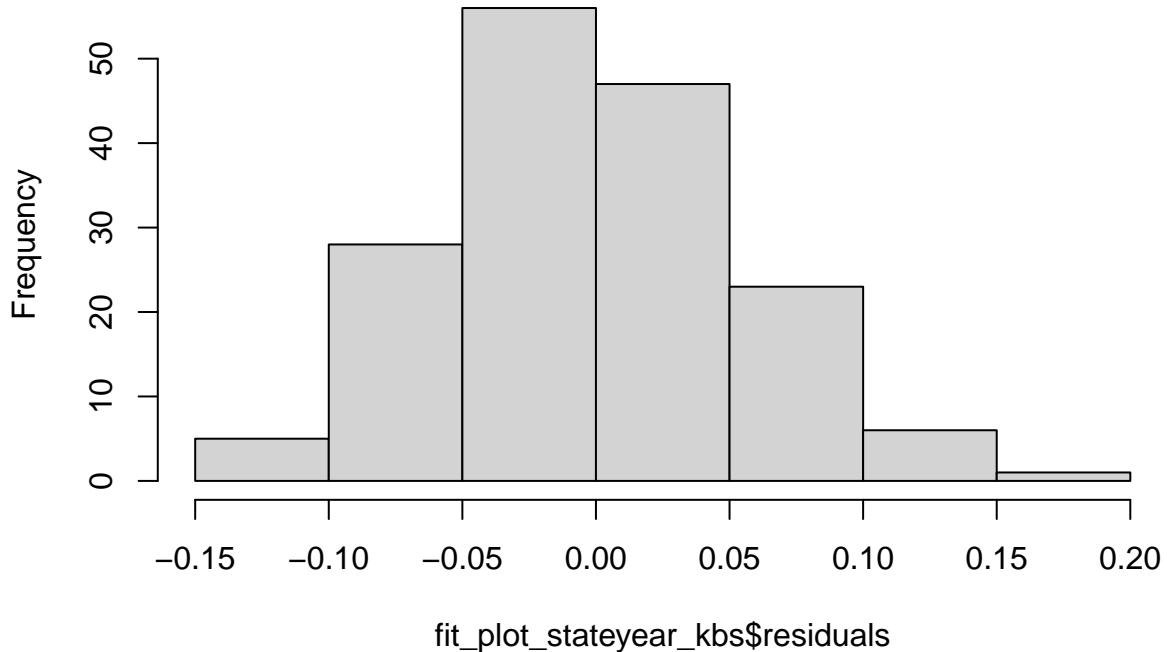
```
qqPlot(fit_plot_stateyear_kbs, main = "QQ Plot")
```



```
## 155 179
## 85 98
```

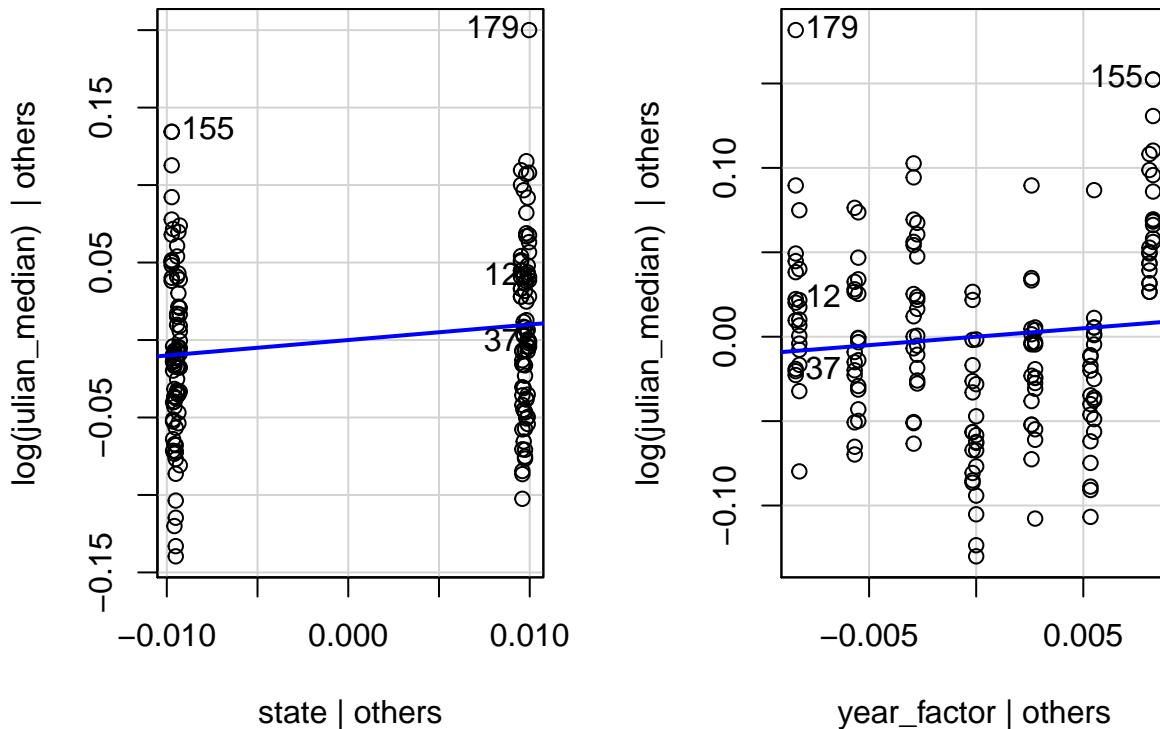
```
hist(fit_plot_stateyear_kbs$residuals)
```

**Histogram of fit\_plot\_stateyear\_kbs\$residuals**



```
leveragePlots(fit_plot_stateyear_kbs)
```

## Leverage Plots



```
ols_test_normality(fit_plot_stateyear_kbs)
```

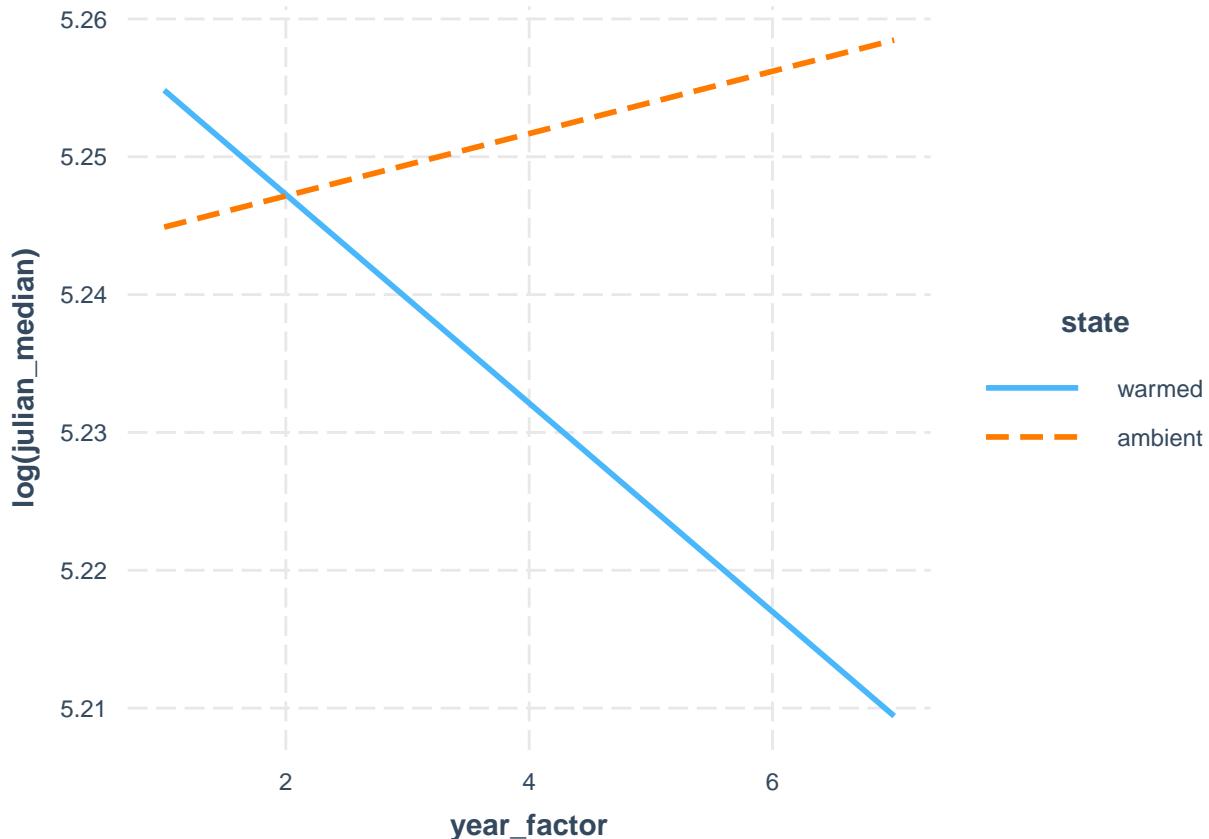
```
## Warning in ks.test(y, "pnorm", mean(y), sd(y)): ties should not be present for
## the Kolmogorov-Smirnov test
```

```
## -----
##      Test          Statistic       pvalue
## -----
## Shapiro-Wilk      0.9918      0.4642
## Kolmogorov-Smirnov 0.0626      0.5333
## Cramer-von Mises  49.2703      0.0000
## Anderson-Darling   0.2922      0.6016
## -----
```

```
# 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_plot_kbs <- lm(log(julian_median) ~ state * year_factor, data = kbs_flwr_plot)
interact_plot(fit3_plot_kbs, pred = year_factor, modx = state)
```

```
## Using data kbs_flwr_plot from global environment. This could cause
## incorrect results if kbs_flwr_plot has been altered since the model was
## fit. You can manually provide the data to the "data =" argument.
```



## MIXED EFFECT MODELS

UMBS Species-level Mixed Effects Models

```
# 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 UMBS models?
mod1 <- lmer(log(julian_median) ~ state * year_factor + insecticide * year_factor +
  (1 | species) + (1 | plot), umbs_flwr_spp, REML = FALSE)
```

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

```
mod2 <- lmer(log(julian_median) ~ state * year_factor + insecticide * year_factor +
  (1 | species), umbs_flwr_spp, REML = FALSE)
# 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                           0.002935 0.002935     1 558.00  1.0207 0.3128
## year_factor                     0.240534 0.240534     1 558.43 83.6531 <2e-16 ***
## insecticide                      0.002560 0.002560     1 558.09  0.8904 0.3458
## state:year_factor                0.000006 0.000006     1 557.98  0.0021 0.9635
```

```

## year_factor:insecticide 0.001030 0.001030      1 558.11  0.3581 0.5498
## ---
## 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                      0.002935 0.002935      1 558.00  1.0207 0.3128
## year_factor                 0.240534 0.240534      1 558.43 83.6531 <2e-16 ***
## insecticide                  0.002560 0.002560      1 558.09  0.8904 0.3458
## state:year_factor            0.000006 0.000006      1 557.98  0.0021 0.9635
## year_factor:insecticide     0.001030 0.001030      1 558.11  0.3581 0.5498
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

# Run an ANOVA to test the 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(mod1, mod2) # Go with model 2 since pvalue >0.05, aka more complex model does not have something
```

```

## Data: umbs_flwr_spp
## Models:
## mod2: log(julian_median) ~ state * year_factor + insecticide * year_factor +
## mod2:      (1 | species)
## mod1: log(julian_median) ~ state * year_factor + insecticide * year_factor +
## mod1:      (1 | species) + (1 | plot)
##      npar    AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## mod2     8 -1631.8 -1597.0 823.91   -1647.8
## mod1     9 -1629.8 -1590.7 823.91   -1647.8      0  1          1

```

```
summ(mod1)
```

Observations	572
Dependent variable	log(julian_median)
Type	Mixed effects linear regression

AIC	-1629.82
BIC	-1590.68
Pseudo-R <sup>2</sup> (fixed effects)	0.02
Pseudo-R <sup>2</sup> (total)	0.92

```
summ(mod2)
```

Fixed Effects					
	Est.	S.E.	t val.	d.f.	p
(Intercept)	5.12	0.05	105.12	15.97	0.00
stateambient	0.01	0.01	1.01	558.00	0.31
year_factor	0.01	0.00	4.84	558.06	0.00
insecticideno_insects	-0.01	0.01	-0.94	558.09	0.35
stateambient:year_factor	0.00	0.00	0.05	557.98	0.96
year_factor:insecticideno_insects	0.00	0.00	0.60	558.11	0.55

p values calculated using Satterthwaite d.f.

Random Effects		
Group	Parameter	Std. Dev.
plot	(Intercept)	0.00
species	(Intercept)	0.18
	Residual	0.05

Grouping Variables		
Group	# groups	ICC
plot	24	0.00
species	14	0.91

Observations	572
Dependent variable	log(julian_median)
Type	Mixed effects linear regression

AIC	-1631.82
BIC	-1597.03
Pseudo-R <sup>2</sup> (fixed effects)	0.02
Pseudo-R <sup>2</sup> (total)	0.92

Fixed Effects					
	Est.	S.E.	t val.	d.f.	p
(Intercept)	5.12	0.05	105.12	15.97	0.00
stateambient	0.01	0.01	1.01	558.00	0.31
year_factor	0.01	0.00	4.84	558.06	0.00
insecticideno_insects	-0.01	0.01	-0.94	558.09	0.35
stateambient:year_factor	0.00	0.00	0.05	557.98	0.96
year_factor:insecticideno_insects	0.00	0.00	0.60	558.11	0.55

p values calculated using Satterthwaite d.f.

```
AICctab(mod1, mod2, weights = T) # yup, model 2
```

```
##      dAICc df weight
## mod2  0.0   8  0.74
## mod1  2.1   9  0.26
```

Random Effects		
Group	Parameter	Std. Dev.
species	(Intercept)	0.18
Residual		0.05

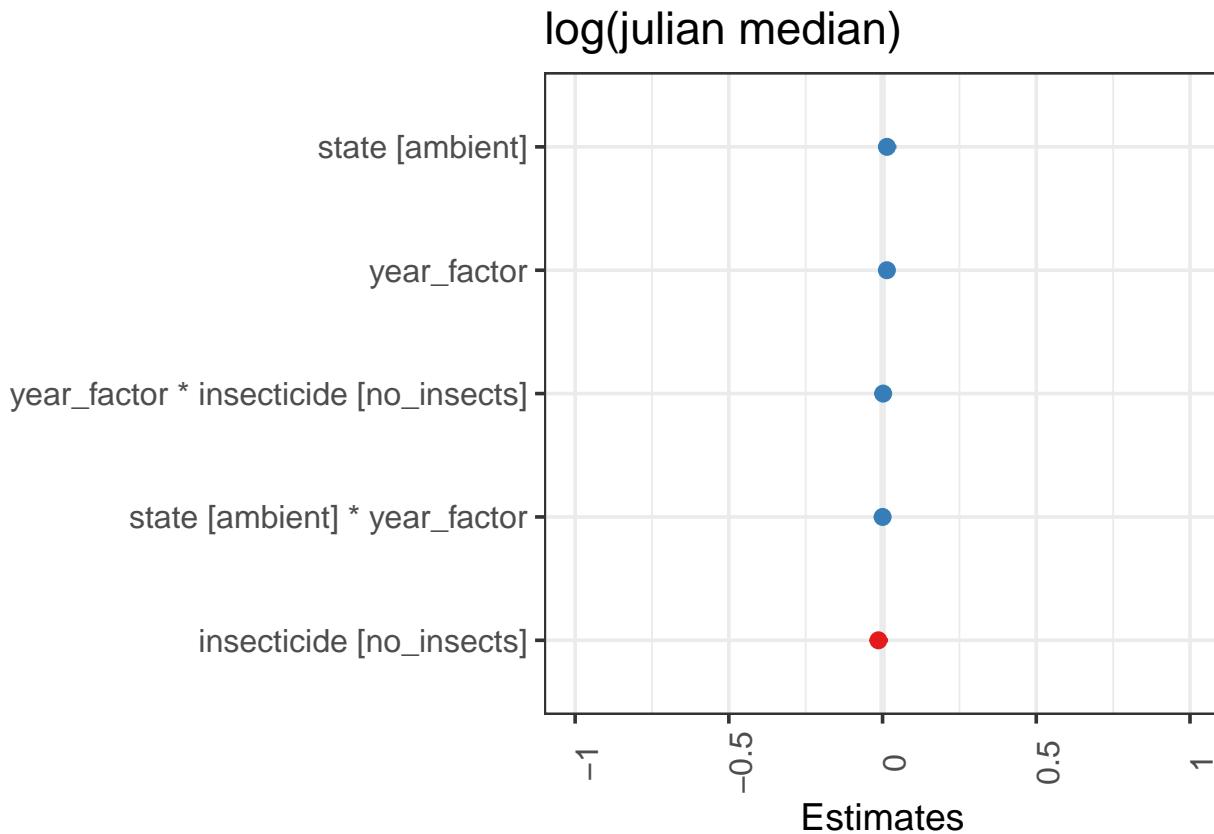
  

Grouping Variables		
Group	# groups	ICC
species	14	0.91

```

# Next, plot the model. There are multiple variables but here's one way to do it
# based on this package sjPlot:
# https://strengejacke.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(mod1)
plot_model(mod2, sort.est = TRUE)

```



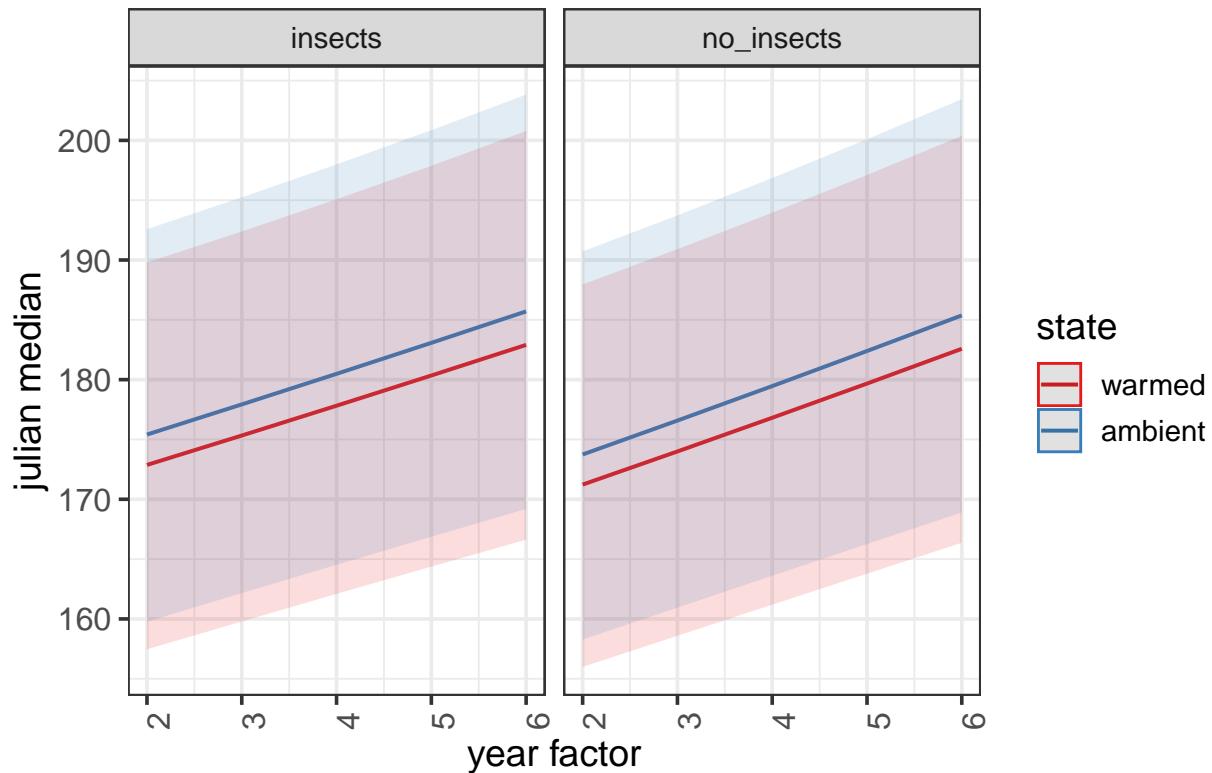
```

# these are the fixed predicted values:
plot_model(mod2, type = "pred", terms = c("year_factor", "state", "insecticide"))

```

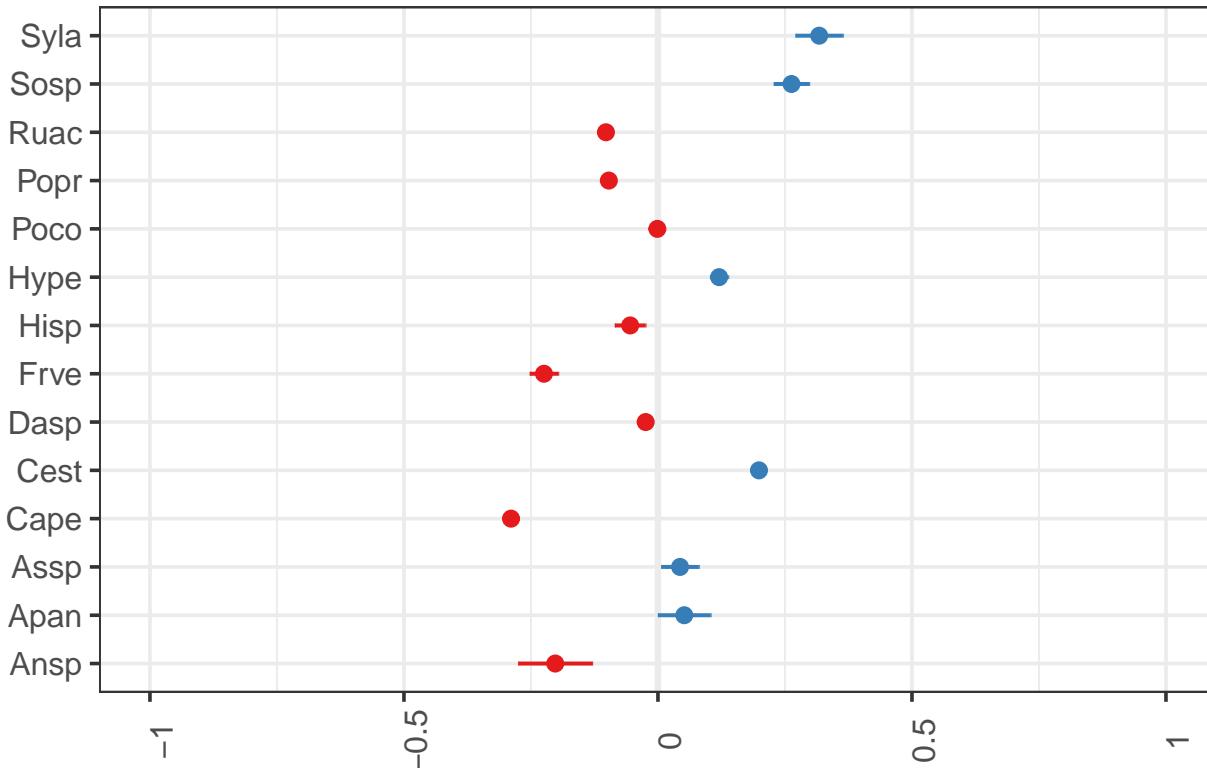
## Model has log-transformed response. Back-transforming predictions to original response scale. Standardized residuals.

## Predicted values of julian median



```
# these are the random effects estimates  
plot_model(mod2, type = "re", terms = c("species"))
```

## Random effects



```
# Do we need to include insecticide? (dropping insecticide from the model)
mod3 <- lmer(log(julian_median) ~ state * year_factor + (1 | species), umbs_flwr_spp,
    REML = FALSE)
anova(mod2, mod3) # Looks like no pvalue > 0.05, so insecticide does not improve model fit so we will ...
```

```
## Data: umbs_flwr_spp
## Models:
## mod3: log(julian_median) ~ state * year_factor + (1 | species)
## mod2: log(julian_median) ~ state * year_factor + insecticide * year_factor +
## mod2: (1 | species)
##      npar     AIC     BIC logLik deviance Chisq Df Pr(>Chisq)
## mod3     6 -1634.1 -1608  823.05   -1646.1
## mod2     8 -1631.8 -1597  823.91   -1647.8 1.7163  2      0.4239
```

```
AICctab(mod2, mod3, weights = T)
```

```
##      dAICc df weight
## mod3 0.0   6  0.77
## mod2 2.4   8  0.23
```

```
# Does year need to be interactive with insecticide? - already removed
# insecticide mod4 <- lmer(log(julian_median) ~ state*year_factor + insecticide +
# (1/species) + (1/plot), umbs_flwr_spp, REML=FALSE)
```

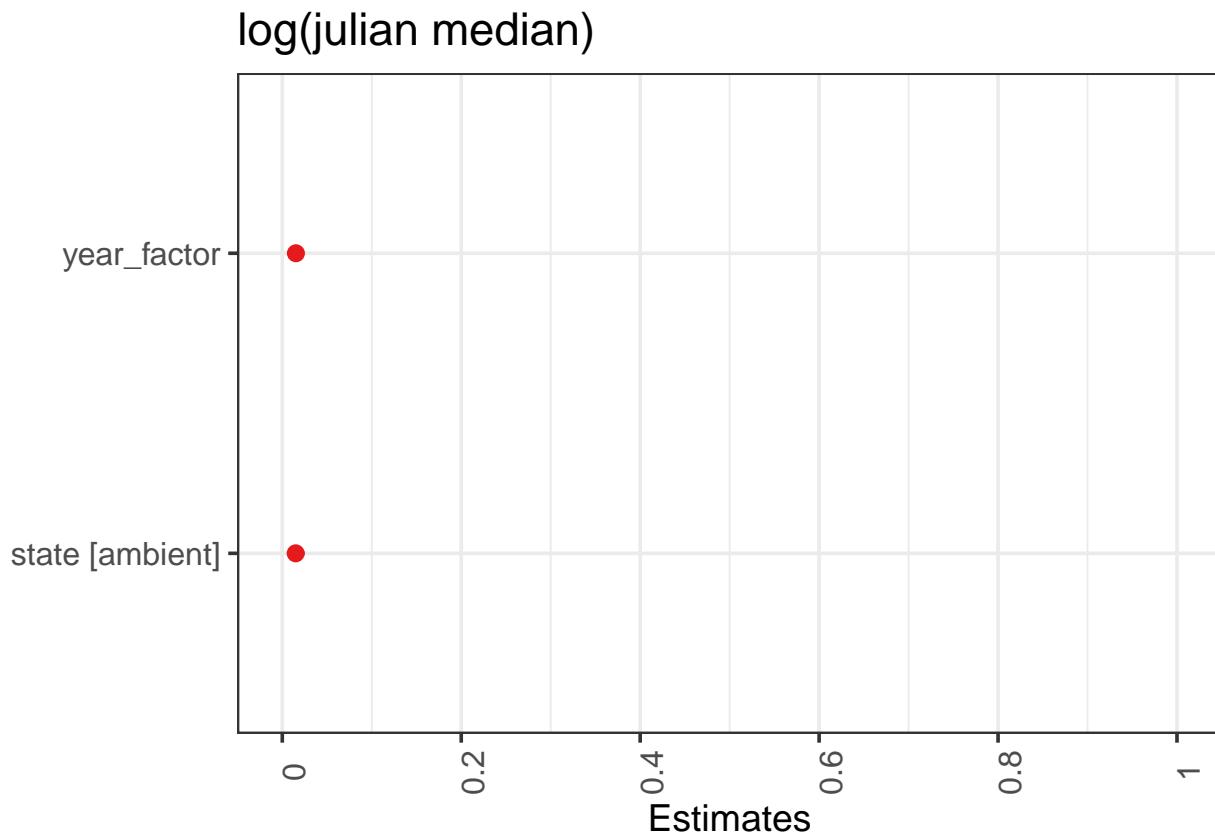
```
# Does year need to be interactive with state?
```

```
mod5 <- lmer(log(julian_median) ~ state + year_factor + (1 | species), umbs_flwr_spp,
  REML = FALSE)
anova(mod3, mod5)
```

```
## Data: umbs_flwr_spp
## Models:
## mod5: log(julian_median) ~ state + year_factor + (1 | species)
## mod3: log(julian_median) ~ state * year_factor + (1 | species)
##      npar    AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## mod5     5 -1636.1 -1614.3  823.05   -1646.1
## mod3     6 -1634.1 -1608.0  823.05   -1646.1 0.0045  1      0.9463
```

```
# No, P>0.05 so state*year_factor doesn't strongly improve model fit so we will
# shift to mod5, but AIC values are super close!!
```

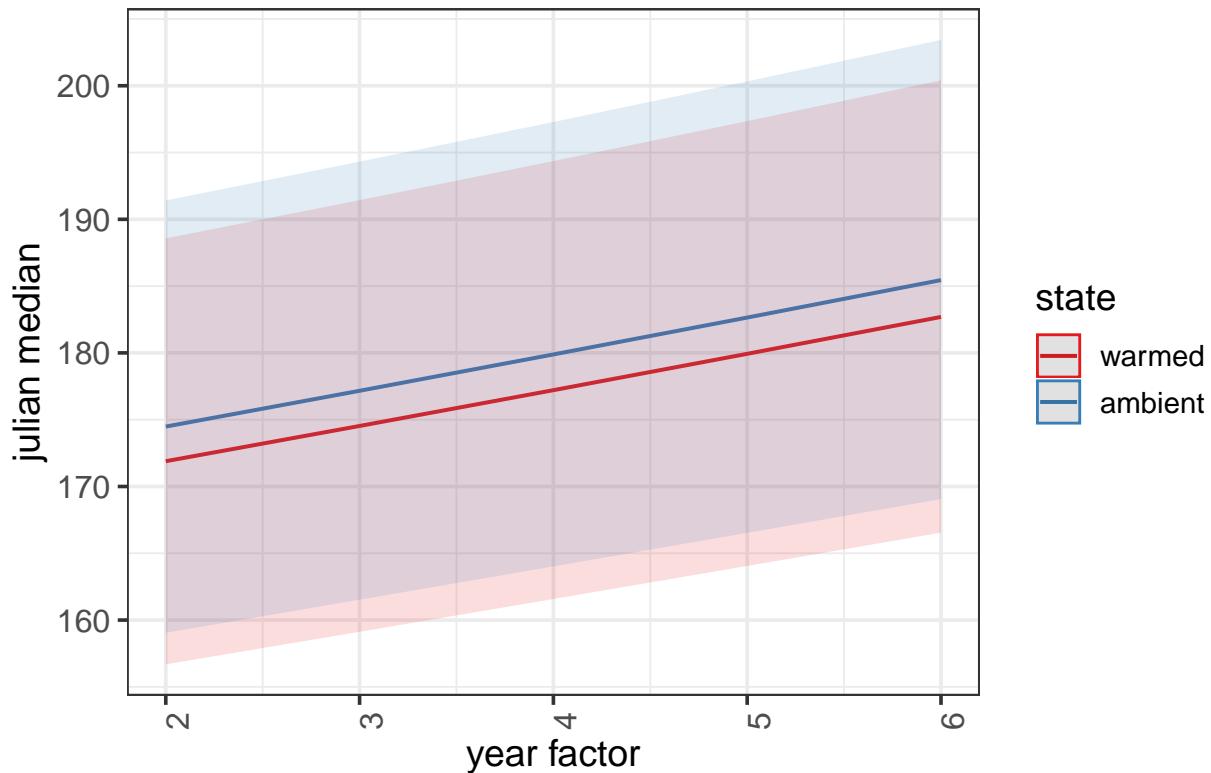
```
# Plot the fixed effects estimates for different models these are the fixed
# effects estimates from summary(mod5)
plot_model(mod5, sort.est = TRUE)
```



```
# these are the fixed predicted values:
plot_model(mod5, type = "pred", terms = c("year_factor", "state"))
```

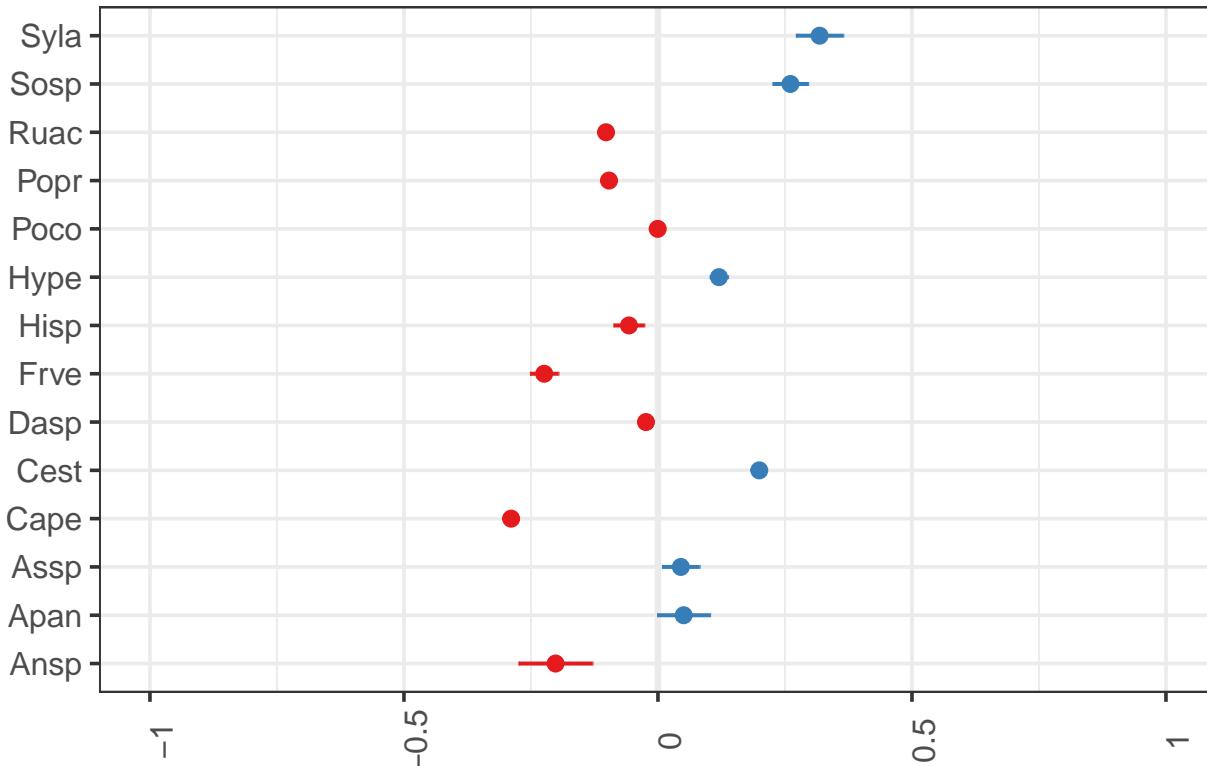
```
## Model has log-transformed response. Back-transforming predictions to original response scale. Standard
```

## Predicted values of julian median



```
# these are the random effects estimates
plot_model(mod5, type = "re", terms = c("species"))
```

## Random effects



```
# If we wanted to include plots nested within year it would look like this:
mod6 <- lmer(log(julian_median) ~ state + year_factor + (1 | species) + (1 + year |
  plot), umbs_flwr_spp, REML = FALSE)
```

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

## Warning: Model failed to converge with 1 negative eigenvalue: -9.0e-02

# model failed to converge...
anova(mod5, mod6)
```

```
## Data: umbs_flwr_spp
## Models:
## mod5: log(julian_median) ~ state + year_factor + (1 | species)
## mod6: log(julian_median) ~ state + year_factor + (1 | species) + (1 +
##       year | plot)
##      npar    AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## mod5     5 -1636.1 -1614.3  823.05   -1646.1
## mod6     8 -1620.1 -1585.3  818.04   -1636.1      0  3          1
```

```
anova(mod5)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##           Sum Sq  Mean Sq NumDF DenDF F value    Pr(>F)
## state      0.031325 0.031325     1 558.16 10.860 0.001045 **
```

```

## year_factor 0.244442 0.244442      1 558.46  84.747 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# mod4 (and mod5) 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:
mod7 <- lmer(log(julian_median) ~ state + species + (1 + factor(year_factor) | plot),
  umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod7a <- lmer(log(julian_median) ~ state + species + factor(year_factor) + (1 | plot),
  umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod7b <- lmer(log(julian_median) ~ state * factor(year_factor) + species + (1 | plot),
  umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod7c <- lmer(log(julian_median) ~ state + species + factor(year_factor) + insecticide +
  (1 | plot), umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

anova(mod5, mod7)  # model 7 is a better fit to data

## Data: umbs_flwr_spp
## Models:
## mod5: log(julian_median) ~ state + year_factor + (1 | species)
## mod7: log(julian_median) ~ state + species + (1 + factor(year_factor) |
## mod7:   plot)
##     npar      AIC      BIC logLik deviance  Chisq Df Pr(>Chisq)
## mod5    5 -1636.1 -1614.3 823.05   -1646.1
## mod7   31 -1661.8 -1527.0 861.91  -1723.8 77.714 26  4.696e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

anova(mod7, mod7a)  #mod 7a

## Data: umbs_flwr_spp
## Models:
## mod7a: log(julian_median) ~ state + species + factor(year_factor) +
## mod7a:   (1 | plot)
## mod7: log(julian_median) ~ state + species + (1 + factor(year_factor) |
## mod7:   plot)
##     npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod7a   21 -1724.2 -1632.9 883.11   -1766.2
## mod7   31 -1661.8 -1527.0 861.91  -1723.8      0 10          1

```

```

anova(mod7a, mod7b) #mod 7a - interaction between state and year does not improve model, but pvalue = 0.9131

## Data: umbs_flwr_spp
## Models:
## mod7a: log(julian_median) ~ state + species + factor(year_factor) +
## mod7a:      (1 | plot)
## mod7b: log(julian_median) ~ state * factor(year_factor) + species +
## mod7b:      (1 | plot)
##      npar      AIC      BIC logLik deviance  Chisq Df Pr(>Chisq)
## mod7a   21 -1724.2 -1632.9 883.11  -1766.2
## mod7b   25 -1717.2 -1608.5 883.60  -1767.2 0.9783  4      0.9131

```

```
anova(mod7a, mod7c) # model 7a
```

```

## Data: umbs_flwr_spp
## Models:
## mod7a: log(julian_median) ~ state + species + factor(year_factor) +
## mod7a:      (1 | plot)
## mod7c: log(julian_median) ~ state + species + factor(year_factor) +
## mod7c:      insecticide + (1 | plot)
##      npar      AIC      BIC logLik deviance  Chisq Df Pr(>Chisq)
## mod7a   21 -1724.2 -1632.9 883.11  -1766.2
## mod7c   22 -1723.8 -1628.1 883.89  -1767.8 1.5511  1      0.213

```

```
summ(mod7a)
```

Observations	572
Dependent variable	log(julian_median)
Type	Mixed effects linear regression

AIC	-1724.23
BIC	-1632.90
Pseudo-R <sup>2</sup> (fixed effects)	0.90
Pseudo-R <sup>2</sup> (total)	0.90

```
anova(mod7a) # investigates whether at least one of the levels within each factor is significantly different from the others
```

```

## Type III Analysis of Variance Table with Satterthwaite's method
##          Sum Sq Mean Sq NumDF DenDF F value    Pr(>F)
## state       0.0316 0.03162     1    572 11.844 0.0006208 ***
## species     12.8865 0.99127    13    572 371.266 < 2.2e-16 ***
## factor(year_factor) 0.3246 0.08114     4    572 30.391 < 2.2e-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 median first flwr dates).
```

```
emmeans(mod7a, list(pairwise ~ state + year_factor), adjust = "tukey")
```

Fixed Effects					
	Est.	S.E.	t val.	d.f.	p
(Intercept)	4.92	0.04	131.84	572.00	0.00
stateambient	0.02	0.00	3.44	572.00	0.00
speciesApan	0.27	0.05	5.97	572.00	0.00
speciesAssp	0.26	0.04	6.25	572.00	0.00
speciesCape	-0.08	0.04	-2.00	572.00	0.05
speciesCest	0.41	0.04	11.16	572.00	0.00
speciesDasp	0.19	0.04	5.14	572.00	0.00
speciesFrve	-0.01	0.04	-0.26	572.00	0.80
speciesHisp	0.16	0.04	4.08	572.00	0.00
speciesHype	0.34	0.04	8.84	572.00	0.00
speciesPoco	0.21	0.04	5.69	572.00	0.00
speciesPopr	0.12	0.04	3.19	572.00	0.00
speciesRuac	0.11	0.04	3.00	572.00	0.00
speciesSosp	0.48	0.04	11.76	572.00	0.00
speciesSyla	0.54	0.04	12.42	572.00	0.00
factor(year_factor)3	0.03	0.01	3.68	572.00	0.00
factor(year_factor)4	0.04	0.01	4.87	572.00	0.00
factor(year_factor)5	0.07	0.01	10.22	572.00	0.00
factor(year_factor)6	0.06	0.01	7.67	572.00	0.00

p values calculated using Satterthwaite d.f.

Random Effects		
Group	Parameter	Std. Dev.
plot	(Intercept)	0.00
Residual		0.05

Grouping Variables		
Group	# groups	ICC
plot	24	0.00

```

## boundary (singular) fit: see ?isSingular

## $`emmeans of state, year_factor'
##   state  year_factor emmean      SE    df lower.CL upper.CL
##   warmed          2  5.139 0.007277 313.4    5.124   5.153
##   ambient          2  5.154 0.007218 273.1    5.139   5.168
##   warmed          3  5.166 0.006848 254.7    5.152   5.179
##   ambient          3  5.181 0.006755 219.1    5.168   5.194
##   warmed          4  5.175 0.006843 246.3    5.161   5.188
##   ambient          4  5.190 0.006766 209.8    5.176   5.203
##   warmed          5  5.212 0.006645 240.3    5.199   5.225
##   ambient          5  5.227 0.006577 212.6    5.214   5.240
##   warmed          6  5.194 0.006561 226.6    5.181   5.207
##   ambient          6  5.209 0.006488 192.4    5.196   5.222
##
## Results are averaged over the levels of: species

```

```

## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $`pairwise differences of state, year_factor`
##   1           estimate      SE    df t.ratio p.value
## warmed 2 - ambient 2 -0.01505 0.00447 21.2 -3.366 0.0683
## warmed 2 - warmed 3 -0.02732 0.00755 574.5 -3.617 0.0119
## warmed 2 - ambient 3 -0.04237 0.00875 242.1 -4.840 0.0001
## warmed 2 - warmed 4 -0.03608 0.00754 579.1 -4.785 0.0001
## warmed 2 - ambient 4 -0.05112 0.00875 235.1 -5.840 <.0001
## warmed 2 - warmed 5 -0.07361 0.00733 577.9 -10.037 <.0001
## warmed 2 - ambient 5 -0.08866 0.00859 229.9 -10.325 <.0001
## warmed 2 - warmed 6 -0.05535 0.00734 578.9 -7.538 <.0001
## warmed 2 - ambient 6 -0.07040 0.00859 224.6 -8.194 <.0001
## ambient 2 - warmed 3 -0.01227 0.00880 239.6 -1.394 0.9280
## ambient 2 - ambient 3 -0.02732 0.00755 574.5 -3.617 0.0119
## ambient 2 - warmed 4 -0.02103 0.00878 235.5 -2.396 0.3329
## ambient 2 - ambient 4 -0.03608 0.00754 579.1 -4.785 0.0001
## ambient 2 - warmed 5 -0.05857 0.00859 217.9 -6.816 <.0001
## ambient 2 - ambient 5 -0.07361 0.00733 577.9 -10.037 <.0001
## ambient 2 - warmed 6 -0.04031 0.00860 222.3 -4.685 0.0002
## ambient 2 - ambient 6 -0.05535 0.00734 578.9 -7.538 <.0001
## warmed 3 - ambient 3 -0.01505 0.00447 21.2 -3.366 0.0683
## warmed 3 - warmed 4 -0.00876 0.00715 572.8 -1.226 0.9681
## warmed 3 - ambient 4 -0.02381 0.00844 215.0 -2.820 0.1364
## warmed 3 - warmed 5 -0.04630 0.00694 572.0 -6.667 <.0001
## warmed 3 - ambient 5 -0.06134 0.00828 209.2 -7.407 <.0001
## warmed 3 - warmed 6 -0.02804 0.00692 572.5 -4.051 0.0023
## warmed 3 - ambient 6 -0.04308 0.00826 202.7 -5.216 <.0001
## ambient 3 - warmed 4 0.00629 0.00842 217.7 0.747 0.9991
## ambient 3 - ambient 4 -0.00876 0.00715 572.8 -1.226 0.9681
## ambient 3 - warmed 5 -0.03125 0.00824 199.6 -3.794 0.0074
## ambient 3 - ambient 5 -0.04630 0.00694 572.0 -6.667 <.0001
## ambient 3 - warmed 6 -0.01299 0.00822 202.5 -1.580 0.8564
## ambient 3 - ambient 6 -0.02804 0.00692 572.5 -4.051 0.0023
## warmed 4 - ambient 4 -0.01505 0.00447 21.2 -3.366 0.0683
## warmed 4 - warmed 5 -0.03754 0.00686 573.7 -5.476 <.0001
## warmed 4 - ambient 5 -0.05258 0.00819 203.7 -6.418 <.0001
## warmed 4 - warmed 6 -0.01928 0.00681 575.5 -2.830 0.1290
## warmed 4 - ambient 6 -0.03432 0.00815 195.2 -4.209 0.0016
## ambient 4 - warmed 5 -0.02249 0.00818 191.8 -2.751 0.1614
## ambient 4 - ambient 5 -0.03754 0.00686 573.7 -5.476 <.0001
## ambient 4 - warmed 6 -0.00423 0.00814 192.5 -0.519 1.0000
## ambient 4 - ambient 6 -0.01928 0.00681 575.5 -2.830 0.1290
## warmed 5 - ambient 5 -0.01505 0.00447 21.2 -3.366 0.0683
## warmed 5 - warmed 6 0.01826 0.00658 572.3 2.777 0.1467
## warmed 5 - ambient 6 0.00321 0.00795 177.5 0.404 1.0000
## ambient 5 - warmed 6 0.03331 0.00795 186.4 4.187 0.0017
## ambient 5 - ambient 6 0.01826 0.00658 572.3 2.777 0.1467
## warmed 6 - ambient 6 -0.01505 0.00447 21.2 -3.366 0.0683
##
## Results are averaged over the levels of: species
## Degrees-of-freedom method: kenward-roger

```

```

## Results are given on the log (not the response) scale.
## P value adjustment: tukey method for comparing a family of 10 estimates

emmeans(mod7a, list(pairwise ~ year_factor), adjust = "tukey")

## boundary (singular) fit: see ?isSingular

## $`emmeans of year_factor`
##   year_factor emmean      SE    df lower.CL upper.CL
##       2     5.146 0.006894 445.6    5.133    5.160
##       3     5.173 0.006424 380.4    5.161    5.186
##       4     5.182 0.006427 364.5    5.170    5.195
##       5     5.220 0.006221 373.0    5.208    5.232
##       6     5.202 0.006130 346.2    5.189    5.214
##
## Results are averaged over the levels of: state, species
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $`pairwise differences of year_factor`
##   1     estimate      SE  df t.ratio p.value
##   2 - 3 -0.02732 0.00755 575 -3.617 0.0030
##   2 - 4 -0.03608 0.00754 579 -4.785 <.0001
##   2 - 5 -0.07361 0.00733 578 -10.037 <.0001
##   2 - 6 -0.05535 0.00734 579 -7.538 <.0001
##   3 - 4 -0.00876 0.00715 573 -1.226 0.7363
##   3 - 5 -0.04630 0.00694 572 -6.667 <.0001
##   3 - 6 -0.02804 0.00692 573 -4.051 0.0006
##   4 - 5 -0.03754 0.00686 574 -5.476 <.0001
##   4 - 6 -0.01928 0.00681 575 -2.830 0.0386
##   5 - 6  0.01826 0.00658 572  2.777 0.0447
##
## Results are averaged over the levels of: state, species
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## P value adjustment: tukey method for comparing a family of 5 estimates

```

```
emmeans(mod7a, list(pairwise ~ species), adjust = "tukey")
```

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

```
## $`emmeans of species`
##   species emmean      SE    df lower.CL upper.CL
##   Ansp    4.970 0.037582 591.4    4.896    5.044
##   Apan    5.239 0.026624 530.3    5.187    5.291
##   Assp    5.228 0.018847 573.3    5.191    5.265
##   Cape    4.895 0.007577 399.5    4.880    4.910
##   Cest    5.384 0.004904 336.4    5.375    5.394
##   Dasp    5.161 0.005742 397.7    5.150    5.173
##   Frve    4.960 0.014315 398.6    4.932    4.988
##   Hisp    5.132 0.015477 527.3    5.101    5.162
```

```

##   Hype      5.305 0.009155 485.1      5.287      5.323
##   Poco      5.182 0.006874 450.5      5.169      5.196
##   Popr      5.088 0.005144 360.2      5.078      5.098
##   Ruac      5.082 0.006370 363.0      5.069      5.094
##   Sosp      5.447 0.017854 357.1      5.412      5.483
##   Syla      5.511 0.023853 520.0      5.465      5.558
##
## Results are averaged over the levels of: state, year_factor
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $'pairwise differences of species'
##   1           estimate      SE  df t.ratio p.value
## Ansp - Apan -0.26930 0.04618 583  -5.831 <.0001
## Ansp - Assp -0.25816 0.04220 590  -6.118 <.0001
## Ansp - Cape  0.07511 0.03836 592   1.958 0.7932
## Ansp - Cest -0.41446 0.03790 592 -10.936 <.0001
## Ansp - Dasp -0.19148 0.03801 592  -5.038 0.0001
## Ansp - Frve  0.01007 0.04029 585   0.250 1.0000
## Ansp - Hisp -0.16210 0.04058 591  -3.995 0.0056
## Ansp - Hype -0.33557 0.03879 591  -8.651 <.0001
## Ansp - Poco -0.21266 0.03816 592  -5.572 <.0001
## Ansp - Popr -0.11837 0.03792 592  -3.121 0.1025
## Ansp - Ruac -0.11220 0.03811 592  -2.944 0.1623
## Ansp - Sosp -0.47768 0.04162 576 -11.476 <.0001
## Ansp - Syla -0.54167 0.04465 583 -12.131 <.0001
## Apan - Assp  0.01114 0.03263 537   0.341 1.0000
## Apan - Cape  0.34441 0.02764 548  12.461 <.0001
## Apan - Cest -0.14516 0.02706 547  -5.364 <.0001
## Apan - Dasp  0.07782 0.02728 545   2.853 0.2019
## Apan - Frve  0.27937 0.03022 495   9.246 <.0001
## Apan - Hisp  0.10721 0.03092 556   3.467 0.0366
## Apan - Hype -0.06627 0.02807 551  -2.361 0.5102
## Apan - Poco  0.05664 0.02748 528   2.062 0.7273
## Apan - Popr  0.15094 0.02710 545   5.570 <.0001
## Apan - Ruac  0.15711 0.02734 558   5.747 <.0001
## Apan - Sosp -0.20837 0.03208 479  -6.495 <.0001
## Apan - Syla -0.27236 0.03568 526  -7.633 <.0001
## Assp - Cape  0.33327 0.02032 583  16.405 <.0001
## Assp - Cest -0.15629 0.01948 588  -8.024 <.0001
## Assp - Dasp  0.06668 0.01966 590   3.391 0.0465
## Assp - Frve  0.26823 0.02349 577  11.418 <.0001
## Assp - Hisp  0.09607 0.02435 590   3.945 0.0068
## Assp - Hype -0.07741 0.02095 583  -3.695 0.0170
## Assp - Poco  0.04550 0.02017 580   2.256 0.5881
## Assp - Popr  0.13980 0.01956 586   7.147 <.0001
## Assp - Ruac  0.14597 0.01988 590   7.343 <.0001
## Assp - Sosp -0.21951 0.02592 552  -8.468 <.0001
## Assp - Syla -0.28350 0.03039 577  -9.328 <.0001
## Cape - Cest -0.48957 0.00902 580  -54.265 <.0001
## Cape - Dasp -0.26659 0.00952 588  -28.003 <.0001
## Cape - Frve -0.06504 0.01620 463  -4.015 0.0053
## Cape - Hisp -0.23720 0.01728 562  -13.730 <.0001

```

```

##  Cape - Hype -0.41068 0.01184 552 -34.675 <.0001
##  Cape - Poco -0.28777 0.01017 588 -28.285 <.0001
##  Cape - Popr -0.19347 0.00915 580 -21.153 <.0001
##  Cape - Ruac -0.18730 0.00992 498 -18.887 <.0001
##  Cape - Sosp -0.55278 0.01937 446 -28.540 <.0001
##  Cape - Syla -0.61677 0.02503 515 -24.642 <.0001
##  Cest - Dasp  0.22298 0.00756 583  29.482 <.0001
##  Cest - Frve  0.42453 0.01513 481  28.062 <.0001
##  Cest - Hisp  0.25236 0.01625 570  15.527 <.0001
##  Cest - Hype  0.07889 0.01037 588   7.605 <.0001
##  Cest - Poco  0.20180 0.00842 590  23.963 <.0001
##  Cest - Popr  0.29609 0.00710 575  41.703 <.0001
##  Cest - Ruac  0.30226 0.00803 592  37.637 <.0001
##  Cest - Sosp -0.06322 0.01851 407  -3.414 0.0441
##  Cest - Syla -0.12721 0.02434 543  -5.226 <.0001
##  Dasp - Frve 0.20155 0.01542 485  13.070 <.0001
##  Dasp - Hisp  0.02939 0.01643 579   1.789 0.8805
##  Dasp - Hype -0.14409 0.01083 590  -13.301 <.0001
##  Dasp - Poco -0.02118 0.00902 592  -2.348 0.5202
##  Dasp - Popr  0.07311 0.00774 587   9.451 <.0001
##  Dasp - Ruac  0.07929 0.00858 591   9.236 <.0001
##  Dasp - Sosp -0.28620 0.01876 422  -15.255 <.0001
##  Dasp - Syla -0.35019 0.02456 548  -14.256 <.0001
##  Frve - Hisp -0.17216 0.02097 588  -8.209 <.0001
##  Frve - Hype -0.34564 0.01699 479  -20.341 <.0001
##  Frve - Poco -0.22273 0.01589 518  -14.017 <.0001
##  Frve - Popr -0.12843 0.01521 491  -8.446 <.0001
##  Frve - Ruac -0.12226 0.01567 463  -7.802 <.0001
##  Frve - Sosp -0.48774 0.02276 514  -21.429 <.0001
##  Frve - Syla -0.55173 0.02772 563  -19.903 <.0001
##  Hisp - Hype -0.17348 0.01806 577  -9.605 <.0001
##  Hisp - Poco -0.05057 0.01706 565  -2.964 0.1545
##  Hisp - Popr  0.04373 0.01635 570   2.675 0.2962
##  Hisp - Ruac  0.04990 0.01675 577   2.979 0.1489
##  Hisp - Sosp -0.31558 0.02357 524  -13.389 <.0001
##  Hisp - Syla -0.37957 0.02846 566  -13.336 <.0001
##  Hype - Poco  0.12291 0.01140 559   10.783 <.0001
##  Hype - Popr  0.21720 0.01048 588   20.731 <.0001
##  Hype - Ruac  0.22338 0.01111 590   20.105 <.0001
##  Hype - Sosp -0.14210 0.02010 384  -7.069 <.0001
##  Hype - Syla -0.20610 0.02546 557  -8.095 <.0001
##  Poco - Popr  0.09429 0.00853 589   11.055 <.0001
##  Poco - Ruac  0.10047 0.00936 556   10.739 <.0001
##  Poco - Sosp -0.26502 0.01907 444  -13.896 <.0001
##  Poco - Syla -0.32901 0.02477 534  -13.285 <.0001
##  Popr - Ruac  0.00617 0.00818 591   0.755 1.0000
##  Popr - Sosp -0.35931 0.01856 421  -19.357 <.0001
##  Popr - Syla -0.42330 0.02438 542  -17.362 <.0001
##  Ruac - Sosp -0.36548 0.01895 399  -19.286 <.0001
##  Ruac - Syla -0.42947 0.02464 557  -17.429 <.0001
##  Sosp - Syla -0.06399 0.02962 550  -2.160 0.6586
## 

## Results are averaged over the levels of: state, year_factor
## Degrees-of-freedom method: kenward-roger

```

```
## Results are given on the log (not the response) scale.  
## P value adjustment: tukey method for comparing a family of 14 estimates
```

```
emmeans(mod7a, list(pairwise ~ insecticide), adjust = "tukey")
```

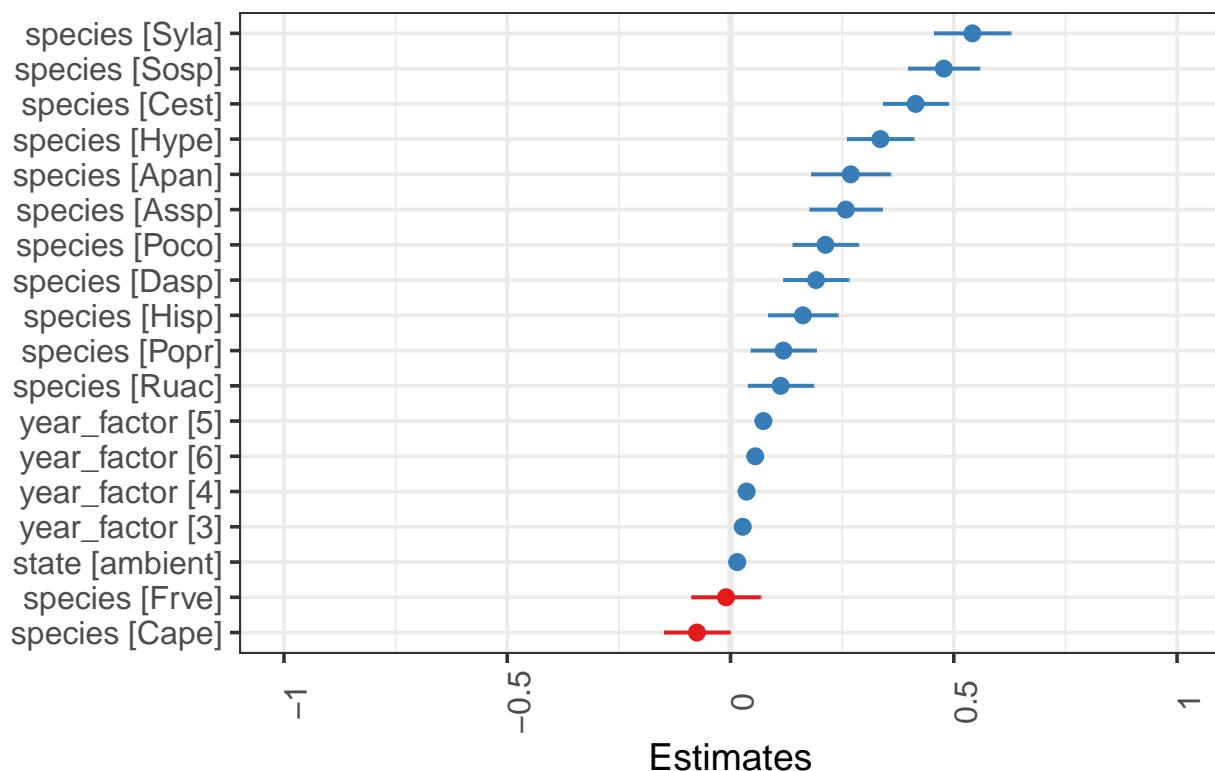
```
## boundary (singular) fit: see ?isSingular
```

```
## Error in emmeans(object = object, specs = specs[[i]], name = contr.name, : No variable named insectici
```

```
# Take a look at the estimates for each fixed effect. These are the estimates  
# from summary(mod7a). 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(mod7a, sort.est = TRUE)
```

log(julian median)

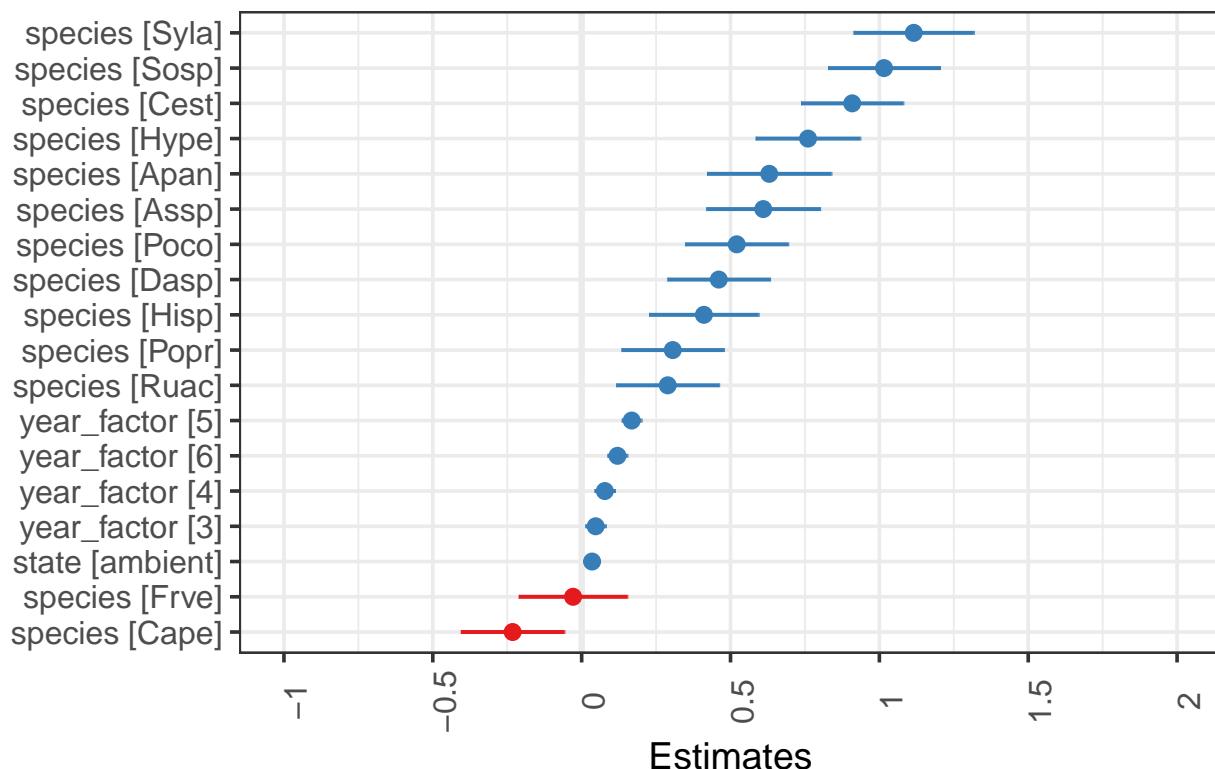


```
# if you want to standardize the estimates:
```

```
plot_model(mod7a, sort.est = TRUE, type = "std")
```

```
## Formula contains log- or sqrt-terms. See help("standardize") for how such terms are standardized.  
## boundary (singular) fit: see ?isSingular
```

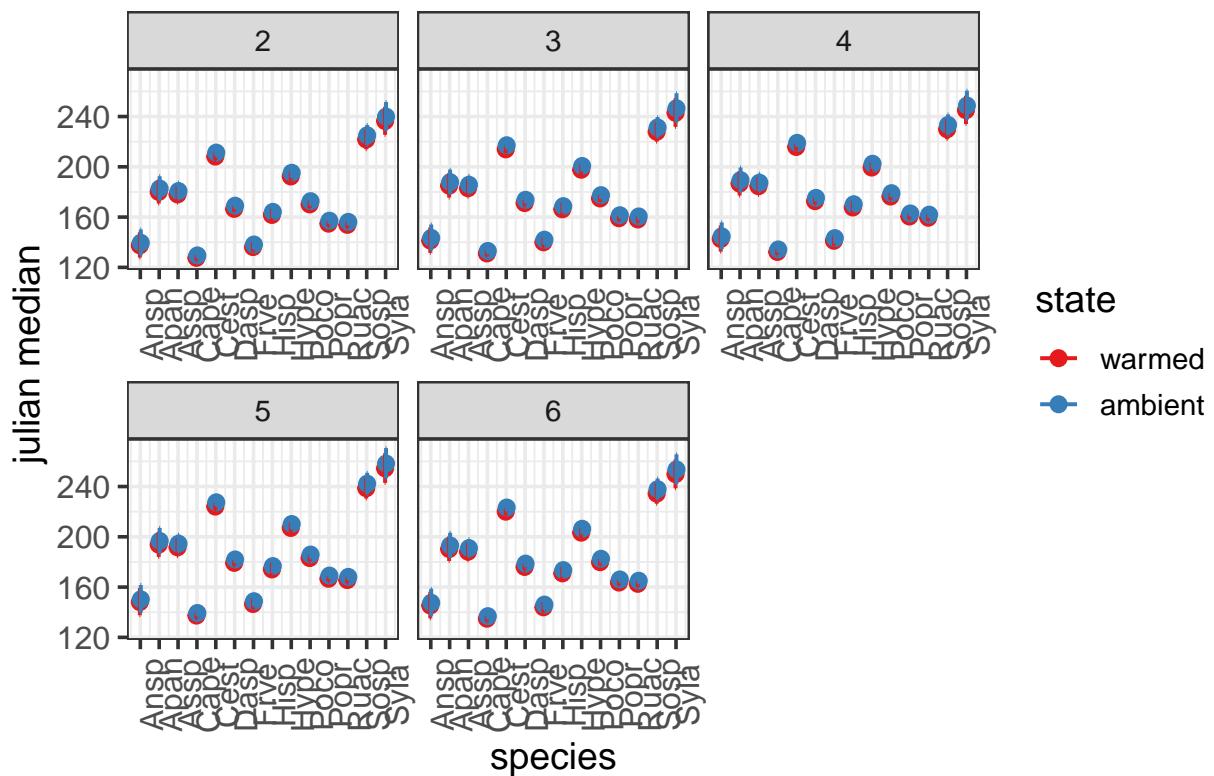
## log(julian median)



```
# these are the fixed predicted values: - note this is a new plot
plot_model(mod7a, type = "pred", terms = c("species", "state", "year_factor"))
```

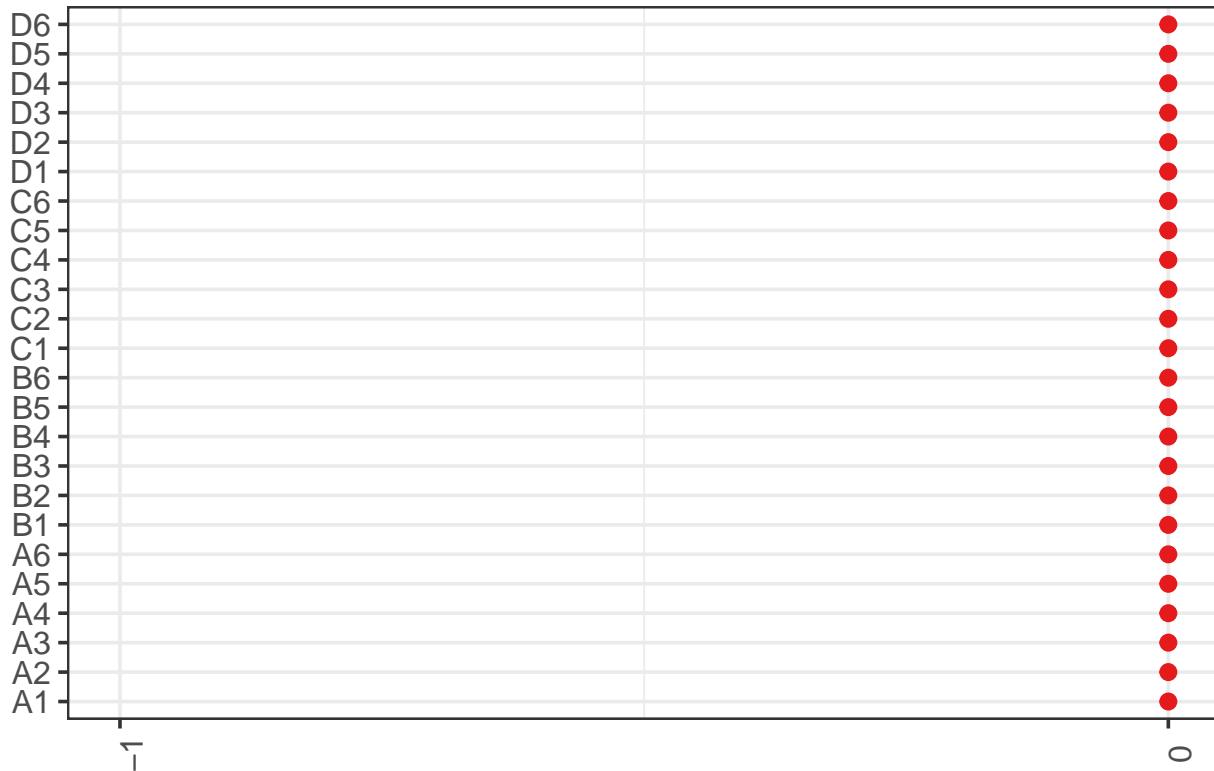
```
## Model has log-transformed response. Back-transforming predictions to original response scale. Standard
```

## Predicted values of julian median



```
# these are the random effects estimates  
plot_model(mod7a, type = "re")
```

## Random effects



```
# including native vs. exotic
umbss_flwr_spp <- within(umbss_flwr_spp, origin <- relevel(factor(origin), ref = "Native")) # releveling
mod8 <- lmer(log(julian_median) ~ state * origin + (1 + factor(year_factor) | plot),
umbss_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod9 <- lmer(log(julian_median) ~ state + origin + (1 + factor(year_factor) | plot),
umbss_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod9a <- lmer(log(julian_median) ~ state + origin + factor(year_factor) + (1 | plot),
umbss_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod9b <- lmer(log(julian_median) ~ state + origin + insecticide + factor(year_factor) +
(1 | plot), umbss_flwr_spp, REML = FALSE)

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

```
anova(mod8, mod9) # model 9 is a better fit to data
```

```
## Data: umbs_flwr_spp
## Models:
## mod9: log(julian_median) ~ state + origin + (1 + factor(year_factor) |
## mod9:     plot)
## mod8: log(julian_median) ~ state * origin + (1 + factor(year_factor) |
## mod8:     plot)
##      npar      AIC      BIC logLik deviance  Chisq Df Pr(>Chisq)
## mod9   20 -497.84 -410.85 268.92   -537.84
## mod8   22 -494.95 -399.26 269.47   -538.95 1.1094  2      0.5742
```

```
anova(mod9, mod9a) # mod 9a
```

```
## Data: umbs_flwr_spp
## Models:
## mod9a: log(julian_median) ~ state + origin + factor(year_factor) + (1 |
## mod9a:     plot)
## mod9: log(julian_median) ~ state + origin + (1 + factor(year_factor) |
## mod9:     plot)
##      npar      AIC      BIC logLik deviance  Chisq Df Pr(>Chisq)
## mod9a   10 -530.70 -487.21 275.35   -550.70
## mod9    20 -497.84 -410.85 268.92   -537.84      0 10          1
```

```
anova(mod9a, mod9b) # mod 9a
```

```
## Data: umbs_flwr_spp
## Models:
## mod9a: log(julian_median) ~ state + origin + factor(year_factor) + (1 |
## mod9a:     plot)
## mod9b: log(julian_median) ~ state + origin + insecticide + factor(year_factor) +
## mod9b:     (1 | plot)
##      npar      AIC      BIC logLik deviance  Chisq Df Pr(>Chisq)
## mod9a   10 -530.70 -487.21 275.35   -550.70
## mod9b   11 -528.83 -480.99 275.41   -550.83 0.1286  1      0.7199
```

```
summary(mod9a)
```

```
## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + origin + factor(year_factor) + (1 |
##     plot)
## Data: umbs_flwr_spp
##
##      AIC      BIC      logLik deviance df.resid
##    -530.7   -487.2    275.3   -550.7      562
##
## Scaled residuals:
##      Min      1Q Median      3Q      Max
## -1.9687 -0.9202 -0.1915  0.8779  3.0575
##
```

```

## Random effects:
## Groups   Name        Variance Std.Dev.
## plot     (Intercept) 0.00000  0.0000
## Residual          0.02236  0.1495
## Number of obs: 572, groups: plot, 24
##
## Fixed effects:
##                               Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)                5.05556  0.01892 572.00000 267.204 < 2e-16 ***
## stateambient               0.01304  0.01251 572.00000  1.042  0.29770
## originBoth                 0.07581  0.03562 572.00000  2.128  0.03375 *
## originExotic               0.11855  0.01390 572.00000  8.527 < 2e-16 ***
## factor(year_factor)3       0.01108  0.02118 572.00000  0.523  0.60099
## factor(year_factor)4       0.02887  0.02116 572.00000  1.364  0.17308
## factor(year_factor)5       0.06335  0.02056 572.00000  3.082  0.00216 **
## factor(year_factor)6       0.04603  0.02051 572.00000  2.244  0.02520 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
## (Intr) sttmbn orgnBt orgnEx fc(_)3 fc(_)4 fc(_)5
## stateambint -0.338
## originBoth  -0.276 -0.013
## originExotc -0.445  0.025  0.257
## fctr(yr_f)3 -0.583  0.001  0.090 -0.069
## fctr(yr_f)4 -0.577 -0.002  0.087 -0.082  0.549
## fctr(yr_f)5 -0.598 -0.002  0.097 -0.077  0.565  0.567
## fctr(yr_f)6 -0.598  0.000  0.108 -0.088  0.568  0.570  0.587
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular

```

```
anova(mod9a)
```

```

## Type III Analysis of Variance Table with Satterthwaite's method
##                   Sum Sq Mean Sq NumDF DenDF F value    Pr(>F)
## state            0.02429 0.02429     1    572  1.0864  0.2977
## origin           1.62573 0.81287     2    572 36.3592 1.366e-15 ***
## factor(year_factor) 0.29150 0.07288     4    572  3.2597  0.0117 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
emmeans(mod9a, list(pairwise ~ state * origin), adjust = "tukey")
```

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

```

## $`emmeans of state, origin`
##   state   origin emmean      SE      df lower.CL upper.CL
##   warmed  Native  5.085 0.013383  82.27    5.059    5.112
##   ambient Native  5.098 0.013272  75.69    5.072    5.125
##   warmed  Both    5.161 0.034968 438.54    5.093    5.230
##   ambient Both   5.174 0.034798 409.14    5.106    5.243
##   warmed  Exotic  5.204 0.009929  32.60    5.184    5.224

```

```

## ambient Exotic 5.217 0.010184 35.42    5.196    5.238
##
## Results are averaged over the levels of: year_factor
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $'pairwise differences of state, origin'
## 1
##   estimate      SE   df t.ratio p.value
## warmed Native - ambient Native -0.0130 0.0127 21.0 -1.029 0.9028
## warmed Native - warmed Both -0.0758 0.0361 571.6 -2.100 0.2889
## warmed Native - ambient Both -0.0888 0.0381 377.8 -2.329 0.1850
## warmed Native - warmed Exotic -0.1186 0.0141 561.4 -8.409 <.0001
## warmed Native - ambient Exotic -0.1316 0.0192 90.4 -6.865 <.0001
## ambient Native - warmed Both -0.0628 0.0384 418.7 -1.635 0.5755
## ambient Native - ambient Both -0.0758 0.0361 571.6 -2.100 0.2889
## ambient Native - warmed Exotic -0.1055 0.0187 80.0 -5.630 <.0001
## ambient Native - ambient Exotic -0.1186 0.0141 561.4 -8.409 <.0001
## warmed Both - ambient Both -0.0130 0.0127 21.0 -1.029 0.9028
## warmed Both - warmed Exotic -0.0427 0.0353 550.8 -1.210 0.8319
## warmed Both - ambient Exotic -0.0558 0.0377 379.4 -1.478 0.6787
## ambient Both - warmed Exotic -0.0297 0.0373 330.5 -0.796 0.9680
## ambient Both - ambient Exotic -0.0427 0.0353 550.8 -1.210 0.8319
## warmed Exotic - ambient Exotic -0.0130 0.0127 21.0 -1.029 0.9028
##
## Results are averaged over the levels of: year_factor
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## P value adjustment: tukey method for comparing a family of 6 estimates

# including growth form - first with interaction term
umbs_flwr_spp <- within(umbs_flwr_spp, growth_habit <- relevel(factor(growth_habit),
  ref = "Forb")) # releveling so forb is the reference
mod10 <- lmer(log(julian_median) ~ state * growth_habit + (1 + factor(year_factor) |
  plot), umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod11 <- lmer(log(julian_median) ~ state + growth_habit + (1 + factor(year_factor) |
  plot), umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod11a <- lmer(log(julian_median) ~ state + growth_habit + factor(year_factor) +
  (1 | plot), umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod11b <- lmer(log(julian_median) ~ state + growth_habit + factor(year_factor) +
  insecticide + (1 | plot), umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

```

```

anova(mod10, mod11) # model 11 is a better fit to data

## Data: umbs_flwr_spp
## Models:
## mod11: log(julian_median) ~ state + growth_habit + (1 + factor(year_factor) | 
## mod11:     plot)
## mod10: log(julian_median) ~ state * growth_habit + (1 + factor(year_factor) | 
## mod10:     plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod11   20 -610.61 -523.62 325.30  -650.61
## mod10   22 -607.47 -511.79 325.74  -651.47 0.8678  2      0.648

```

```

anova(mod11, mod11a) # model 11a is still a better fit to data

```

```

## Data: umbs_flwr_spp
## Models:
## mod11a: log(julian_median) ~ state + growth_habit + factor(year_factor) + 
## mod11a:     (1 | plot)
## mod11: log(julian_median) ~ state + growth_habit + (1 + factor(year_factor) | 
## mod11:     plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod11a  10 -648.44 -604.95 334.22  -668.44
## mod11   20 -610.61 -523.62 325.30  -650.61      0 10          1

```

```

anova(mod11a, mod11b) # model 11a still a better fit to data

```

```

## Data: umbs_flwr_spp
## Models:
## mod11a: log(julian_median) ~ state + growth_habit + factor(year_factor) + 
## mod11a:     (1 | plot)
## mod11b: log(julian_median) ~ state + growth_habit + factor(year_factor) + 
## mod11b:     insecticide + (1 | plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod11a  10 -648.44 -604.95 334.22  -668.44
## mod11b  11 -647.84 -600.00 334.92  -669.84 1.4004  1      0.2367

```

```

summary(mod11a)

```

```

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
##   method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + growth_habit + factor(year_factor) +
##           (1 | plot)
## Data: umbs_flwr_spp
##
##      AIC      BIC      logLik deviance df.resid
##    -648.4   -605.0    334.2    -668.4      562
## 
## Scaled residuals:
##       Min     1Q Median     3Q    Max
## -2.6685 -0.5723  0.1949  0.8177  2.0564
## 

```

```

## Random effects:
## Groups   Name        Variance Std.Dev.
## plot     (Intercept) 0.0000   0.0000
## Residual          0.0182   0.1349
## Number of obs: 572, groups: plot, 24
##
## Fixed effects:
##                         Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)            5.222936  0.016510 572.000000 316.344 < 2e-16 ***
## stateambient          0.002125  0.011307 572.000000  0.188 0.850983
## growth_habit          -0.135338  0.040330 572.000000 -3.356 0.000844 ***
## growth_habitGraminoid -0.169526  0.011457 572.000000 -14.796 < 2e-16 ***
## factor(year_factor)3   0.032751  0.019018 572.000000  1.722 0.085589 .
## factor(year_factor)4   0.041751  0.018997 572.000000  2.198 0.028361 *
## factor(year_factor)5   0.074759  0.018530 572.000000  4.035 6.21e-05 ***
## factor(year_factor)6   0.058423  0.018401 572.000000  3.175 0.001579 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) sttmbn grwth_G fc(_)3 fc(_)4 fc(_)5
## stateambint -0.366
## growth_habt -0.227  0.055
## grwth_hbtGr -0.379  0.046  0.155
## fctr(yr_f)3 -0.617  0.007  0.070 -0.038
## fctr(yr_f)4 -0.635  0.007  0.095  0.001  0.543
## fctr(yr_f)5 -0.656  0.009  0.135 -0.002  0.560  0.563
## fctr(yr_f)6 -0.661  0.011  0.121  0.003  0.562  0.566  0.584
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular

```

```
anova(mod11a)
```

```

## Type III Analysis of Variance Table with Satterthwaite's method
##           Sum Sq Mean Sq NumDF DenDF F value    Pr(>F)
## state       0.0006 0.00064     1    572  0.0353  0.850983
## growth_habit 4.0049 2.00243     2    572 110.0400 < 2.2e-16 ***
## factor(year_factor) 0.3363 0.08408     4    572  4.6207  0.001117 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
emmeans(mod11a, list(pairwise ~ factor(year_factor) * growth_habit), adjust = "tukey")
```

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

```

## $`emmeans of year_factor, growth_habit`
##   year_factor growth_habit emmean      SE      df lower.CL upper.CL
##   2 Forb        5.224 0.01551 421.9    5.194    5.254
##   3 Forb        5.257 0.01470 389.6    5.228    5.286
##   4 Forb        5.266 0.01431 355.3    5.238    5.294
##   5 Forb        5.299 0.01354 331.4    5.272    5.325
##   6 Forb        5.282 0.01338 320.9    5.256    5.309

```

```

##          2      5.089 0.04043 535.4      5.009      5.168
##          3      5.121 0.04150 531.3      5.040      5.203
##          4      5.130 0.04184 531.7      5.048      5.213
##          5      5.163 0.04225 537.5      5.080      5.246
##          6      5.147 0.04194 531.0      5.065      5.229
## 2 Graminoid  5.054 0.01528 437.0      5.024      5.085
## 3 Graminoid  5.087 0.01386 371.1      5.060      5.114
## 4 Graminoid  5.096 0.01406 381.8      5.069      5.124
## 5 Graminoid  5.129 0.01325 351.4      5.103      5.155
## 6 Graminoid  5.113 0.01317 335.3      5.087      5.139
##
## Results are averaged over the levels of: state
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## ## $'pairwise differences of year_factor, growth_habit'
##   1           estimate      SE  df t.ratio p.value
## 2 Forb - 3 Forb -0.03275 0.0192 564 -1.709  0.9298
## 2 Forb - 4 Forb -0.04175 0.0191 567 -2.181  0.6787
## 2 Forb - 5 Forb -0.07476 0.0187 564 -4.004  0.0062
## 2 Forb - 6 Forb -0.05842 0.0185 564 -3.151  0.1055
## 2 Forb - 2      0.13534 0.0409 570  3.308  0.0675
## 2 Forb - 3      0.10259 0.0464 575  2.211  0.6570
## 2 Forb - 4      0.09359 0.0468 575  1.999  0.7984
## 2 Forb - 5      0.06058 0.0472 576  1.283  0.9945
## 2 Forb - 6      0.07692 0.0469 575  1.639  0.9495
## 2 Forb - 2 Graminoid 0.16953 0.0116 564 14.605 <.0001
## 2 Forb - 3 Graminoid 0.13678 0.0220 577  6.209 <.0001
## 2 Forb - 4 Graminoid 0.12778 0.0224 576  5.707 <.0001
## 2 Forb - 5 Graminoid 0.09477 0.0220 576  4.314  0.0018
## 2 Forb - 6 Graminoid 0.11110 0.0219 577  5.072  0.0001
## 3 Forb - 4 Forb -0.00900 0.0183 563 -0.492  1.0000
## 3 Forb - 5 Forb -0.04201 0.0177 563 -2.367  0.5408
## 3 Forb - 6 Forb -0.02567 0.0176 563 -1.455  0.9817
## 3 Forb - 2      0.16809 0.0439 578  3.827  0.0120
## 3 Forb - 3      0.13534 0.0409 570  3.308  0.0675
## 3 Forb - 4      0.12634 0.0453 577  2.792  0.2543
## 3 Forb - 5      0.09333 0.0457 577  2.044  0.7709
## 3 Forb - 6      0.10967 0.0454 576  2.417  0.5032
## 3 Forb - 2 Graminoid 0.20228 0.0228 575  8.882 <.0001
## 3 Forb - 3 Graminoid 0.16953 0.0116 564 14.605 <.0001
## 3 Forb - 4 Graminoid 0.16053 0.0221 574  7.278 <.0001
## 3 Forb - 5 Graminoid 0.12752 0.0216 575  5.910 <.0001
## 3 Forb - 6 Graminoid 0.14385 0.0215 576  6.678 <.0001
## 4 Forb - 5 Forb -0.03301 0.0177 564 -1.867  0.8684
## 4 Forb - 6 Forb -0.01667 0.0176 565 -0.949  0.9998
## 4 Forb - 2      0.17709 0.0435 577  4.074  0.0047
## 4 Forb - 3      0.14434 0.0444 576  3.252  0.0794
## 4 Forb - 4      0.13534 0.0409 570  3.308  0.0675
## 4 Forb - 5      0.10233 0.0452 577  2.264  0.6179
## 4 Forb - 6      0.11867 0.0449 575  2.642  0.3441
## 4 Forb - 2 Graminoid 0.21128 0.0224 578  9.436 <.0001
## 4 Forb - 3 Graminoid 0.17853 0.0213 579  8.387 <.0001

```

## 4 Forb - 4 Graminoid	0.16953	0.0116	564	14.605	<.0001
## 4 Forb - 5 Graminoid	0.13652	0.0211	578	6.462	<.0001
## 4 Forb - 6 Graminoid	0.15285	0.0211	579	7.247	<.0001
## 5 Forb - 6 Forb	0.01634	0.0170	561	0.963	0.9998
## 5 Forb - 2	0.21010	0.0426	576	4.932	0.0001
## 5 Forb - 3	0.17735	0.0435	574	4.076	0.0047
## 5 Forb - 4	0.16835	0.0439	574	3.832	0.0118
## 5 Forb - 5	0.13534	0.0409	570	3.308	0.0675
## 5 Forb - 6	0.15167	0.0440	574	3.444	0.0444
## 5 Forb - 2 Graminoid	0.24429	0.0220	578	11.102	<.0001
## 5 Forb - 3 Graminoid	0.21153	0.0208	579	10.154	<.0001
## 5 Forb - 4 Graminoid	0.20253	0.0212	577	9.568	<.0001
## 5 Forb - 5 Graminoid	0.16953	0.0116	564	14.605	<.0001
## 5 Forb - 6 Graminoid	0.18586	0.0206	578	9.021	<.0001
## 6 Forb - 2	0.19376	0.0428	578	4.527	0.0007
## 6 Forb - 3	0.16101	0.0437	576	3.682	0.0201
## 6 Forb - 4	0.15201	0.0441	576	3.444	0.0444
## 6 Forb - 5	0.11900	0.0445	577	2.672	0.3249
## 6 Forb - 6	0.13534	0.0409	570	3.308	0.0675
## 6 Forb - 2 Graminoid	0.22795	0.0218	577	10.436	<.0001
## 6 Forb - 3 Graminoid	0.19520	0.0207	578	9.436	<.0001
## 6 Forb - 4 Graminoid	0.18620	0.0210	576	8.855	<.0001
## 6 Forb - 5 Graminoid	0.15319	0.0205	576	7.475	<.0001
## 6 Forb - 6 Graminoid	0.16953	0.0116	564	14.605	<.0001
## 2 - 3	-0.03275	0.0192	564	-1.709	0.9298
## 2 - 4	-0.04175	0.0191	567	-2.181	0.6787
## 2 - 5	-0.07476	0.0187	564	-4.004	0.0062
## 2 - 6	-0.05842	0.0185	564	-3.151	0.1055
## 2 - 2 Graminoid	0.03419	0.0408	557	0.838	1.0000
## 2 - 3 Graminoid	0.00144	0.0436	572	0.033	1.0000
## 2 - 4 Graminoid	-0.00756	0.0434	571	-0.174	1.0000
## 2 - 5 Graminoid	-0.04057	0.0425	569	-0.955	0.9998
## 2 - 6 Graminoid	-0.02423	0.0427	571	-0.567	1.0000
## 3 - 4	-0.00900	0.0183	563	-0.492	1.0000
## 3 - 5	-0.04201	0.0177	563	-2.367	0.5408
## 3 - 6	-0.02567	0.0176	563	-1.455	0.9817
## 3 - 2 Graminoid	0.06694	0.0465	568	1.440	0.9834
## 3 - 3 Graminoid	0.03419	0.0408	557	0.838	1.0000
## 3 - 4 Graminoid	0.02519	0.0445	570	0.566	1.0000
## 3 - 5 Graminoid	-0.00782	0.0436	566	-0.179	1.0000
## 3 - 6 Graminoid	0.00852	0.0438	569	0.194	1.0000
## 4 - 5	-0.03301	0.0177	564	-1.867	0.8684
## 4 - 6	-0.01667	0.0176	565	-0.949	0.9998
## 4 - 2 Graminoid	0.07594	0.0467	568	1.625	0.9527
## 4 - 3 Graminoid	0.04319	0.0450	569	0.960	0.9998
## 4 - 4 Graminoid	0.03419	0.0408	557	0.838	1.0000
## 4 - 5 Graminoid	0.00118	0.0438	566	0.027	1.0000
## 4 - 6 Graminoid	0.01752	0.0441	568	0.398	1.0000
## 5 - 6	0.01634	0.0170	561	0.963	0.9998
## 5 - 2 Graminoid	0.10895	0.0471	570	2.311	0.5829
## 5 - 3 Graminoid	0.07620	0.0454	571	1.679	0.9389
## 5 - 4 Graminoid	0.06720	0.0451	571	1.489	0.9774
## 5 - 5 Graminoid	0.03419	0.0408	557	0.838	1.0000
## 5 - 6 Graminoid	0.05052	0.0445	571	1.136	0.9985

```

##   6 - 2 Graminoid      0.09261 0.0468 568  1.977  0.8111
##   6 - 3 Graminoid      0.05986 0.0451 569  1.328  0.9923
##   6 - 4 Graminoid      0.05086 0.0448 569  1.135  0.9985
##   6 - 5 Graminoid      0.01785 0.0439 566  0.406  1.0000
##   6 - 6 Graminoid      0.03419 0.0408 557  0.838  1.0000
## 2 Graminoid - 3 Graminoid -0.03275 0.0192 564 -1.709  0.9298
## 2 Graminoid - 4 Graminoid -0.04175 0.0191 567 -2.181  0.6787
## 2 Graminoid - 5 Graminoid -0.07476 0.0187 564 -4.004  0.0062
## 2 Graminoid - 6 Graminoid -0.05842 0.0185 564 -3.151  0.1055
## 3 Graminoid - 4 Graminoid -0.00900 0.0183 563 -0.492  1.0000
## 3 Graminoid - 5 Graminoid -0.04201 0.0177 563 -2.367  0.5408
## 3 Graminoid - 6 Graminoid -0.02567 0.0176 563 -1.455  0.9817
## 4 Graminoid - 5 Graminoid -0.03301 0.0177 564 -1.867  0.8684
## 4 Graminoid - 6 Graminoid -0.01667 0.0176 565 -0.949  0.9998
## 5 Graminoid - 6 Graminoid  0.01634 0.0170 561  0.963  0.9998
##
## Results are averaged over the levels of: state
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## P value adjustment: tukey method for comparing a family of 15 estimates

```

```

# 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
mod12 <- lmer(log(julian_median) ~ state * factor(year_factor) + (1 + factor(year_factor) |
  species), umbs_flwr_spp)

```

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

```

# 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
mod13 <- lmer(log(julian_median) ~ state * factor(year_factor) + (1 | species), umbs_flwr_spp)

# All the models ran:
mod1 <- lmer(log(julian_median) ~ state * year_factor + insecticide * year_factor +
  (1 | species) + (1 | plot), umbs_flwr_spp, REML = FALSE)

```

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

```

mod2 <- lmer(log(julian_median) ~ state * year_factor + insecticide * year_factor +
  (1 | species), umbs_flwr_spp, REML = FALSE)
mod3 <- lmer(log(julian_median) ~ state * year_factor + (1 | species), umbs_flwr_spp,
  REML = FALSE)
mod4 <- lmer(log(julian_median) ~ state * year_factor + insecticide + (1 | species) +
  (1 | plot), umbs_flwr_spp, REML = FALSE)

```

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

```

mod5 <- lmer(log(julian_median) ~ state + year_factor + (1 | species), umbs_flwr_spp,
  REML = FALSE)
mod6 <- lmer(log(julian_median) ~ state + year_factor + (1 | species) + (1 + year |
  plot), umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

## Warning: Model failed to converge with 1 negative eigenvalue: -9.0e-02

mod7 <- lmer(log(julian_median) ~ state + species + (1 + factor(year_factor) | plot),
  umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod7a <- lmer(log(julian_median) ~ state + species + factor(year_factor) + (1 | plot),
  umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod7b <- lmer(log(julian_median) ~ state * factor(year_factor) + species + (1 | plot),
  umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod7c <- lmer(log(julian_median) ~ state + species + factor(year_factor) + insecticide +
  (1 | plot), umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod8 <- lmer(log(julian_median) ~ state * origin + (1 + factor(year_factor) | plot),
  umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod9 <- lmer(log(julian_median) ~ state + origin + (1 + factor(year_factor) | plot),
  umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod9a <- lmer(log(julian_median) ~ state + origin + factor(year_factor) + (1 | plot),
  umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod10 <- lmer(log(julian_median) ~ state * growth_habit + (1 + factor(year_factor) | plot),
  umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

```

```

mod11 <- lmer(log(julian_median) ~ state + growth_habit + (1 + factor(year_factor) | plot), umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod11a <- lmer(log(julian_median) ~ state + growth_habit + factor(year_factor) + (1 | plot), umbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod12 <- lmer(log(julian_median) ~ state * factor(year_factor) + (1 + factor(year_factor) | species), umbs_flwr_spp)

## boundary (singular) fit: see ?isSingular

mod13 <- lmer(log(julian_median) ~ state * factor(year_factor) + (1 | species), umbs_flwr_spp)
AICctab(mod1, mod2, mod3, mod5, mod6, mod7, mod7a, mod7b, mod7c, mod8, mod9, mod9a, mod10, mod11, mod11a, mod13, weights = T) #mod7a is the best fitting model (took out mod12 bc it w

##          dAICc   df weight
## mod7a      0.0  21  0.569
## mod7c      0.6  22  0.419
## mod7b     7.7  25  0.012
## mod7      64.4 31 <0.001
## mod5      86.6  5 <0.001
## mod3      88.6  6 <0.001
## mod2      91.0  8 <0.001
## mod1      93.0  9 <0.001
## mod6     102.7  8 <0.001
## mod13    145.5 12 <0.001
## mod11a   1074.5 10 <0.001
## mod11    1113.5 20 <0.001
## mod10    1116.9 22 <0.001
## mod9a    1192.2 10 <0.001
## mod9     1226.2 20 <0.001
## mod8     1229.4 22 <0.001

summary(mod7a)

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + species + factor(year_factor) +
##           (1 | plot)
## Data: umbs_flwr_spp
##
##          AIC      BIC      logLik deviance df.resid
## -1724.2 -1632.9    883.1   -1766.2      551
##
## Scaled residuals:
##       Min     1Q Median     3Q    Max

```

```

## -6.3936 -0.5403 -0.0484  0.6020  3.2104
##
## Random effects:
## Groups   Name        Variance Std.Dev.
## plot     (Intercept) 0.00000  0.00000
## Residual           0.00267  0.05167
## Number of obs: 572, groups: plot, 24
##
## Fixed effects:
##                               Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)                4.923754  0.037345 572.00000 131.844 < 2e-16 ***
## stateambient               0.015047  0.004372 572.00000  3.441 0.000621 ***
## speciesApan                0.269304  0.045135 572.00000  5.967 4.25e-09 ***
## speciesAssp                0.258165  0.041286 572.00000  6.253 7.88e-10 ***
## speciesCape                -0.075106 0.037567 572.00000 -1.999 0.046051 *
## speciesCest                0.414459  0.037123 572.00000 11.165 < 2e-16 ***
## speciesDasp                0.191482  0.037233 572.00000  5.143 3.72e-07 ***
## speciesFrve                -0.010067 0.039373 572.00000 -0.256 0.798296
## speciesHisp                0.162096  0.039753 572.00000  4.078 5.20e-05 ***
## speciesHype                0.335573  0.037967 572.00000  8.838 < 2e-16 ***
## speciesPoco                0.212661  0.037383 572.00000  5.689 2.04e-08 ***
## speciesPopr                0.118368  0.037146 572.00000  3.187 0.001518 **
## speciesRuac                0.112195  0.037338 572.00000  3.005 0.002774 **
## speciesSosp                0.477677  0.040627 572.00000 11.757 < 2e-16 ***
## speciesSyla                0.541668  0.043618 572.00000 12.418 < 2e-16 ***
## factor(year_factor)3       0.027319  0.007423 572.00000  3.680 0.000255 ***
## factor(year_factor)4       0.036078  0.007406 572.00000  4.871 1.44e-06 ***
## factor(year_factor)5       0.073615  0.007206 572.00000 10.216 < 2e-16 ***
## factor(year_factor)6       0.055354  0.007214 572.00000  7.673 7.27e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ',' 1

##
## Correlation matrix not shown by default, as p = 19 > 12.
## Use print(x, correlation=TRUE)  or
##      vcov(x)      if you need it

## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular

anova(mod7a)

## Type III Analysis of Variance Table with Satterthwaite's method
##                               Sum Sq Mean Sq NumDF DenDF F value    Pr(>F)
## state                      0.0316 0.03162     1    572 11.844 0.0006208 ***
## species                     12.8865 0.99127    13    572 371.266 < 2.2e-16 ***
## factor(year_factor)        0.3246 0.08114     4    572 30.391 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ',' 1

```

## UMBS Plot-level Mixed Effects Models:

```

mod1p <- lmer(log(julian_median) ~ state + (1 | plot), umbs_flwr_plot, REML = FALSE)
mod2p <- lmer(log(julian_median) ~ insecticide + (1 | plot), umbs_flwr_plot, REML = FALSE)
mod3p <- lmer(log(julian_median) ~ insecticide + state + (1 | plot), umbs_flwr_plot,
    REML = FALSE)
mod4p <- lmer(log(julian_median) ~ insecticide * state + (1 | plot), umbs_flwr_plot,
    REML = FALSE)
mod5p <- lmer(log(julian_median) ~ state + year_factor + (1 | plot), umbs_flwr_plot,
    REML = FALSE)
mod6p <- lmer(log(julian_median) ~ state + year_factor + insecticide + (1 | plot),
    umbs_flwr_plot, REML = FALSE)
mod7p <- lmer(log(julian_median) ~ state * year_factor + (1 | plot), umbs_flwr_plot,
    REML = FALSE)
mod8p <- lmer(log(julian_median) ~ state * year_factor + insecticide + (1 | plot),
    umbs_flwr_plot, REML = FALSE)
mod9p <- lmer(log(julian_median) ~ state * insecticide + year_factor + (1 | plot),
    umbs_flwr_plot, REML = FALSE)
mod10p <- lmer(log(julian_median) ~ state + insecticide * year_factor + (1 | plot),
    umbs_flwr_plot, REML = FALSE)
mod11p <- lmer(log(julian_median) ~ state * year_factor * insecticide + (1 | plot),
    umbs_flwr_plot, REML = FALSE)
AICctab(mod1p, mod2p, mod3p, mod4p, mod5p, mod6p, mod7p, mod8p, mod9p, mod10p, mod11p,
    weights = T) # model 5p and 7p are the top models

```

```

##      dAICc df weight
## mod5p   0.0  5  0.278
## mod10p  0.1  7  0.258
## mod6p   1.3  6  0.143
## mod7p   1.7  6  0.116
## mod11p  2.2 10  0.093
## mod8p   3.1  7  0.059
## mod9p   3.3  7  0.053
## mod2p   55.4 4 <0.001
## mod1p   55.9 4 <0.001
## mod3p   57.2 5 <0.001
## mod4p   59.1 6 <0.001

```

```
anova(mod5p, mod7p) # go with simpler model 5p
```

```

## Data: umbs_flwr_plot
## Models:
## mod5p: log(julian_median) ~ state + year_factor + (1 | plot)
## mod7p: log(julian_median) ~ state * year_factor + (1 | plot)
##          npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod5p     5 -451.46 -437.52 230.73  -461.46
## mod7p     6 -449.94 -433.21 230.97  -461.94 0.4777  1     0.4895

```

```
AICctab(mod5p, mod7p, weights = T)
```

```
##      dAICc df weight
```

```
## mod5p 0.0   5   0.7
## mod7p 1.7   6   0.3
```

```
summ(mod5p)
```

Observations	120
Dependent variable	log(julian_median)
Type	Mixed effects linear regression

AIC	-451.46
BIC	-437.52
Pseudo-R <sup>2</sup> (fixed effects)	0.31
Pseudo-R <sup>2</sup> (total)	0.55

Fixed Effects					
	Est.	S.E.	t val.	d.f.	p
(Intercept)	5.11	0.01	459.60	80.45	0.00
stateambient	0.01	0.01	0.67	24.00	0.51
year_factor	0.02	0.00	8.93	96.00	0.00

p values calculated using Satterthwaite d.f.

Random Effects		
Group	Parameter	Std. Dev.
plot	(Intercept)	0.02
	Residual	0.03

Grouping Variables		
Group	# groups	ICC
plot	24	0.35

```
summary(mod5p)
```

```
## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
##   method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + year_factor + (1 | plot)
##   Data: umbs_flwr_plot
##
##      AIC      BIC      logLik deviance df.resid
## -451.5 -437.5    230.7    -461.5      115
##
## Scaled residuals:
##      Min      1Q Median      3Q      Max 
## -3.4711 -0.5205 -0.0592  0.6232  2.2821 
##
## Random effects:
```

```

## Groups      Name          Variance Std.Dev.
## plot      (Intercept) 0.0005201 0.02281
## Residual           0.0009634 0.03104
## Number of obs: 120, groups: plot, 24
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)
## (Intercept) 5.110223  0.011119 80.446797 459.601 < 2e-16 ***
## stateambient 0.007274  0.010900 23.999999  0.667   0.511
## year_factor   0.017890  0.002004 96.000000  8.929 2.97e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##            (Intr) sttmbn
## stateambient -0.490
## year_factor   -0.721  0.000

emmeans(mod5p, list(pairwise ~ state + year_factor), adjust = "tukey")

## $`emmeans of state, year_factor`
## state  year_factor emmean      SE      df lower.CL upper.CL
## warmed           4  5.182 0.00805 26.18    5.165  5.198
## ambient          4  5.189 0.00805 26.18    5.173  5.206
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $`pairwise differences of state, year_factor`
## 1                  estimate      SE      df t.ratio p.value
## warmed 4 - ambient 4 -0.00727 0.0114 26.2 -0.639  0.5284
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.

# including native vs. exotic
umbss_flwr_plot_origin <- within(umbss_flwr_plot_origin, origin <- relevel(factor(origin),
  ref = "Native")) # releveling so native is the reference
mod12p <- lmer(log(julian_median) ~ state * origin + (1 + year_factor | plot), umbss_flwr_plot_origin,
  REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod13p <- lmer(log(julian_median) ~ state + origin + (1 + year_factor | plot), umbss_flwr_plot_origin,
  REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod14p <- lmer(log(julian_median) ~ state + origin + year_factor + (1 | plot), umbss_flwr_plot_origin,
  REML = FALSE)

```

```

## boundary (singular) fit: see ?isSingular

anova(mod12p, mod13p) # go with model 13p

## Data: umbs_flwr_plot_origin
## Models:
## mod13p: log(julian_median) ~ state + origin + (1 + year_factor | plot)
## mod12p: log(julian_median) ~ state * origin + (1 + year_factor | plot)
##      npar    AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## mod13p     9 -606.74 -567.06 312.37   -624.74
## mod12p    12 -606.66 -553.75 315.33   -630.66 5.9171  3      0.1157

anova(mod13p, mod14p) # mod 14p

## Data: umbs_flwr_plot_origin
## Models:
## mod14p: log(julian_median) ~ state + origin + year_factor + (1 | plot)
## mod13p: log(julian_median) ~ state + origin + (1 + year_factor | plot)
##      npar    AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## mod14p     8 -608.90 -573.63 312.45   -624.90
## mod13p     9 -606.74 -567.06 312.37   -624.74      0  1           1

summary(mod14p)

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + origin + year_factor + (1 | plot)
## Data: umbs_flwr_plot_origin
##
##      AIC    BIC logLik deviance df.resid
## -608.9 -573.6 312.5   -624.9      599
##
## Scaled residuals:
##      Min    1Q Median    3Q    Max
## -3.2499 -0.5014  0.0136  0.6508  1.8993
##
## Random effects:
## Groups   Name        Variance Std.Dev.
## plot     (Intercept) 0.00000  0.0000
## Residual            0.02091  0.1446
## Number of obs: 607, groups: plot, 24
##
## Fixed effects:
##             Estimate Std. Error       df t value Pr(>|t|)    
## (Intercept) 5.299293  0.018166 607.000000 291.708 <2e-16 ***
## statewarmed 0.003932  0.011780 607.000000   0.334   0.739    
## origin      -0.052729  0.073673 607.000000  -0.716   0.474    
## originBoth   -0.159738  0.017073 607.000000  -9.356 <2e-16 ***
## originExotic -0.109924  0.012996 607.000000  -8.458 <2e-16 ***
## year_factor -0.001506  0.003741 607.000000  -0.403   0.687    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

## 
## Correlation of Fixed Effects:
##          (Intr) sttfrm origin orgnBt orgnEx
## statewarmed -0.340
## origin      -0.179  0.003
## originBoth   -0.400  0.064  0.088
## originExotc -0.393 -0.005  0.096  0.403
## year_factor -0.792  0.025  0.143  0.119  0.023
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular

anova(mod14p)

## Type III Analysis of Variance Table with Satterthwaite's method
##           Sum Sq Mean Sq NumDF DenDF F value Pr(>F)
## state      0.00233 0.00233     1    607  0.1114 0.7387
## origin     2.38287 0.79429     3    607 37.9800 <2e-16 ***
## year_factor 0.00339 0.00339     1    607  0.1620 0.6875
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

emmeans(mod14p, list(pairwise ~ state + origin + year_factor), adjust = "tukey")

## boundary (singular) fit: see ?isSingular

## $`emmeans of state, origin, year_factor`
##   state origin year_factor emmean       SE     df lower.CL upper.CL
##   ambient Native      3.662  5.294 0.01116  68.83    5.272  5.316
##   warmed  Native      3.662  5.298 0.01123  70.15    5.275  5.320
##   ambient      Both      3.662  5.241 0.07379 611.44    5.096  5.386
##   warmed      Both      3.662  5.245 0.07383 611.76    5.100  5.390
##   ambient Exotic      3.662  5.134 0.01500 175.37    5.104  5.164
##   warmed Exotic      3.662  5.138 0.01588 209.87    5.107  5.169
##   ambient Exotic      3.662  5.184 0.01076  60.65    5.162  5.205
##   warmed Exotic      3.662  5.188 0.01076  60.62    5.166  5.209
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $`pairwise differences of state, origin, year_factor`

|                                                                   | estimate |
|-------------------------------------------------------------------|----------|
| ambient Native 3.66227347611203 - warmed Native 3.66227347611203  | -0.00393 |
| ambient Native 3.66227347611203 - ambient Both 3.66227347611203   | 0.05273  |
| ambient Native 3.66227347611203 - warmed Both 3.66227347611203    | 0.04880  |
| ambient Native 3.66227347611203 - ambient Exotic 3.66227347611203 | 0.15974  |
| ambient Native 3.66227347611203 - warmed Exotic 3.66227347611203  | 0.15581  |
| ambient Native 3.66227347611203 - ambient Both 3.66227347611203   | 0.10992  |
| ambient Native 3.66227347611203 - warmed Both 3.66227347611203    | 0.10599  |
| warmed Native 3.66227347611203 - ambient Both 3.66227347611203    | 0.05666  |
| warmed Native 3.66227347611203 - warmed Both 3.66227347611203     | 0.05273  |
| warmed Native 3.66227347611203 - ambient Exotic 3.66227347611203  | 0.16367  |


```

```

##  warmed Native 3.66227347611203 - warmed Both 3.66227347611203      0.15974
##  warmed Native 3.66227347611203 - ambient Exotic 3.66227347611203     0.11386
##  warmed Native 3.66227347611203 - warmed Exotic 3.66227347611203     0.10992
##  ambient  3.66227347611203 - warmed  3.66227347611203      -0.00393
##  ambient  3.66227347611203 - ambient Both 3.66227347611203      0.10701
##  ambient  3.66227347611203 - warmed Both 3.66227347611203     0.10308
##  ambient  3.66227347611203 - ambient Exotic 3.66227347611203     0.05719
##  ambient  3.66227347611203 - warmed Exotic 3.66227347611203     0.05326
##  warmed   3.66227347611203 - ambient Both 3.66227347611203     0.11094
##  warmed   3.66227347611203 - warmed Both 3.66227347611203     0.10701
##  warmed   3.66227347611203 - ambient Exotic 3.66227347611203     0.06113
##  warmed   3.66227347611203 - warmed Exotic 3.66227347611203     0.05719
##  ambient Both 3.66227347611203 - warmed Both 3.66227347611203     -0.00393
##  ambient Both 3.66227347611203 - ambient Exotic 3.66227347611203     -0.04981
##  ambient Both 3.66227347611203 - warmed Exotic 3.66227347611203     -0.05375
##  warmed Both 3.66227347611203 - ambient Exotic 3.66227347611203     -0.04588
##  warmed Both 3.66227347611203 - warmed Exotic 3.66227347611203     -0.04981
##  ambient Exotic 3.66227347611203 - warmed Exotic 3.66227347611203     -0.00393
##          SE    df t.ratio p.value
##  0.0118  22.5 -0.332  1.0000
##  0.0743 612.8  0.710  0.9967
##  0.0752 606.7  0.649  0.9981
##  0.0172 606.9  9.296 <.0001
##  0.0215 191.6  7.253 <.0001
##  0.0131 594.6  8.414 <.0001
##  0.0176 103.1  6.026 <.0001
##  0.0752 605.3  0.754  0.9952
##  0.0743 612.8  0.710  0.9967
##  0.0202 157.6  8.085 <.0001
##  0.0172 606.9  9.296 <.0001
##  0.0177 104.5  6.439 <.0001
##  0.0131 594.6  8.414 <.0001
##  0.0118 22.5  -0.332  1.0000
##  0.0748 613.0  1.431  0.8425
##  0.0758 603.9  1.359  0.8752
##  0.0742 612.8  0.771  0.9945
##  0.0751 605.5  0.710  0.9967
##  0.0755 604.9  1.468  0.8240
##  0.0748 613.0  1.431  0.8425
##  0.0751 606.8  0.814  0.9924
##  0.0742 612.8  0.771  0.9945
##  0.0118 22.5  -0.332  1.0000
##  0.0169 607.0  -2.950  0.0647
##  0.0199 150.6  -2.694  0.1325
##  0.0213 185.9  -2.156  0.3833
##  0.0169 607.0  -2.950  0.0647
##  0.0118 22.5  -0.332  1.0000
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## P value adjustment: tukey method for comparing a family of 8 estimates

```

*# including growth form - first with interaction term*

```
umbs_flwr_plot_growthhabit <- within(umbs_flwr_plot_growthhabit, growth_habit <- relevel(factor(growth_habit), ref = "Native"))
```

```

ref = "Forb")) # releveling so forb is the reference
mod15p <- lmer(log(julian_median) ~ state * growth_habit + (1 + year_factor | plot),
  umbs_flwr_plot_growthhabit, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod16p <- lmer(log(julian_median) ~ state + growth_habit + (1 + year_factor | plot),
  umbs_flwr_plot_growthhabit, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod17p <- lmer(log(julian_median) ~ state + growth_habit + year_factor + (1 | plot),
  umbs_flwr_plot_growthhabit, REML = FALSE)
anova(mod15p, mod16p) # go with model 16pu

## Data: umbs_flwr_plot_growthhabit
## Models:
## mod16p: log(julian_median) ~ state + growth_habit + (1 + year_factor | plot)
## mod15p: log(julian_median) ~ state * growth_habit + (1 + year_factor | plot)
## mod15p: npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod16p    8 -1259.6 -1224.3 637.81   -1275.6
## mod15p   10 -1256.4 -1212.2 638.19   -1276.4 0.7531  2     0.6862

anova(mod16p, mod17p) # mod 17pu

## Data: umbs_flwr_plot_growthhabit
## Models:
## mod17p: log(julian_median) ~ state + growth_habit + year_factor + (1 | plot)
## mod16p: log(julian_median) ~ state + growth_habit + (1 + year_factor | plot)
## mod16p: npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod17p    7 -1262.5 -1231.6 638.26   -1276.5
## mod16p    8 -1259.6 -1224.3 637.81   -1275.6      0  1           1

summary(mod17p)

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + growth_habit + year_factor + (1 | plot)
## Data: umbs_flwr_plot_growthhabit
##
##          AIC      BIC      logLik deviance df.resid
## -1262.5 -1231.6    638.3   -1276.5      606
##
## Scaled residuals:
##       Min     1Q Median     3Q    Max
## -1.0000 -0.7500 -0.5000  0.2500  1.5000
## 
## Correlation of Fixed Effects:
##            (Intercept) state:growth_habit
## state:growth_habit -0.5000
## 
```

```

## -3.2072 -0.5881 -0.0492  0.4711  3.2640
##
## Random effects:
## Groups   Name        Variance Std.Dev.
## plot     (Intercept) 3.925e-05 0.006265
## Residual           7.260e-03 0.085206
## Number of obs: 613, groups: plot, 24
##
## Fixed effects:
##                               Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)                5.318954  0.010535 279.429978 504.873 <2e-16 ***
## statewarmed                 0.005897  0.007356  23.828948   0.802   0.431
## growth_habit              -0.219484  0.010089 604.507276 -21.755 <2e-16 ***
## growth_habitGraminoid    -0.223689  0.007534 589.779435 -29.689 <2e-16 ***
## year_factor                  0.002480  0.002194 598.154763   1.131   0.259
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) sttwrm grwth_ grwt_G
## statewarmed -0.352
## growth_habit -0.372  0.047
## grwth_hbtGr -0.334 -0.004  0.366
## year_factor  -0.782  0.006  0.118 -0.024

```

```
anova(mod17p)
```

```

## Type III Analysis of Variance Table with Satterthwaite's method
##                   Sum Sq Mean Sq NumDF DenDF F value Pr(>F)
## state            0.0047  0.0047     1   23.83  0.6426 0.4307
## growth_habit    7.3938  3.6969     2 598.85 509.2054 <2e-16 ***
## year_factor     0.0093  0.0093     1 598.15  1.2782 0.2587
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
emmeans(mod17p, list(pairwise ~ state + growth_habit + year_factor), adjust = "tukey")
```

```

## $`emmeans of state, growth_habit, year_factor`
## state  growth_habit year_factor emmean       SE      df lower.CL upper.CL
## ambient Forb          3.69  5.328 0.006763  66.27    5.315  5.342
## warmed  Forb          3.69  5.334 0.006776  65.90    5.320  5.348
## ambient
## warmed
## ambient Graminoid     3.69  5.109 0.009392 202.23    5.090  5.127
## warmed Graminoid      3.69  5.115 0.009770 224.53    5.095  5.134
## ambient Graminoid     3.69  5.104 0.006833  68.97    5.091  5.118
## warmed Graminoid      3.69  5.110 0.006815  67.66    5.097  5.124
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $`pairwise differences of state, growth_habit, year_factor`
## 1
## ambient Forb 3.69004893964111 - warmed Forb 3.69004893964111

```

```

## ambient Forb 3.69004893964111 - ambient 3.69004893964111
## ambient Forb 3.69004893964111 - warmed 3.69004893964111
## ambient Forb 3.69004893964111 - ambient Graminoid 3.69004893964111
## ambient Forb 3.69004893964111 - warmed Graminoid 3.69004893964111
## warmed Forb 3.69004893964111 - ambient 3.69004893964111
## warmed Forb 3.69004893964111 - warmed 3.69004893964111
## warmed Forb 3.69004893964111 - ambient Graminoid 3.69004893964111
## warmed Forb 3.69004893964111 - warmed Graminoid 3.69004893964111
## ambient 3.69004893964111 - warmed 3.69004893964111
## ambient 3.69004893964111 - ambient Graminoid 3.69004893964111
## ambient 3.69004893964111 - warmed Graminoid 3.69004893964111
## warmed 3.69004893964111 - ambient Graminoid 3.69004893964111
## warmed 3.69004893964111 - warmed Graminoid 3.69004893964111
## ambient Graminoid 3.69004893964111 - warmed Graminoid 3.69004893964111
## estimate SE df t.ratio p.value
## -0.00590 0.00769 26.3 -0.767 0.9708
## 0.21948 0.01013 608.0 21.657 <.0001
## 0.21359 0.01300 192.2 16.432 <.0001
## 0.22369 0.00755 593.0 29.612 <.0001
## 0.21779 0.01076 104.8 20.237 <.0001
## 0.22538 0.01244 167.7 18.114 <.0001
## 0.21948 0.01013 608.0 21.657 <.0001
## 0.22959 0.01080 105.8 21.257 <.0001
## 0.22369 0.00755 593.0 29.612 <.0001
## -0.00590 0.00769 26.3 -0.767 0.9708
## 0.00421 0.01019 607.4 0.413 0.9985
## -0.00169 0.01247 169.6 -0.136 1.0000
## 0.01010 0.01306 195.1 0.774 0.9716
## 0.00421 0.01019 607.4 0.413 0.9985
## -0.00590 0.00769 26.3 -0.767 0.9708
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## P value adjustment: tukey method for comparing a family of 6 estimates

```

## KBS Species-level Mixed Effects Models

```

# Do we need to include plot as a random effect with the KBS models?
mod1k <- lmer(log(julian_median) ~ state * year_factor + insecticide * year_factor +
  (1 | species) + (1 | plot), kbs_flwr_spp, REML = FALSE)
mod2k <- lmer(log(julian_median) ~ state * year_factor + insecticide * year_factor +
  (1 | species), kbs_flwr_spp, REML = FALSE)

# 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(mod1k)

```

```

## Type III Analysis of Variance Table with Satterthwaite's method
##                                     Sum Sq  Mean Sq NumDF DenDF F value    Pr(>F)
## state                           0.0072778 0.0072778     1   290.93  3.5053 0.06218 .
## year_factor                     0.0090678 0.0090678     1  1119.40  4.3674 0.03686 *
## insecticide                      0.0005264 0.0005264     1   286.47  0.2536 0.61497
## state:year_factor               0.0067478 0.0067478     1  1115.43  3.2500 0.07169 .
## year_factor:insecticide          0.0000022 0.0000022     1  1113.37  0.0011 0.97394

```

```

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

anova(mod2k)

## Type III Analysis of Variance Table with Satterthwaite's method
##                               Sum Sq   Mean Sq NumDF DenDF F value Pr(>F)
## state                      0.0085263 0.0085263     1 1118.3  4.0635 0.04406 *
## year_factor                 0.0096984 0.0096984     1 1119.5  4.6221 0.03178 *
## insecticide                  0.0004668 0.0004668     1 1118.3  0.2225 0.63725
## state:year_factor           0.0070479 0.0070479     1 1118.5  3.3589 0.06711 .
## year_factor:insecticide    0.0000007 0.0000007     1 1118.5  0.0004 0.98498
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# Run an ANOVA to test the 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(mod1k, mod2k) # models are not that different from each other so go with simpler model = model 2

```

```

## Data: kbs_flwr_spp
## Models:
## mod2k: log(julian_median) ~ state * year_factor + insecticide * year_factor +
## mod2k:      (1 | species)
## mod1k: log(julian_median) ~ state * year_factor + insecticide * year_factor +
## mod1k:      (1 | species) + (1 | plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod2k     8 -3622.0 -3581.7 1819.0   -3638.0
## mod1k     9 -3622.2 -3576.7 1820.1   -3640.2 2.1282  1     0.1446

```

```
summary(mod1k)
```

```

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
##   method [lmerModLmerTest]
## Formula:
## log(julian_median) ~ state * year_factor + insecticide * year_factor +
##      (1 | species) + (1 | plot)
## Data: kbs_flwr_spp
##
##      AIC      BIC logLik deviance df.resid
## -3622.2 -3576.7 1820.1   -3640.2      1142
##
## Scaled residuals:
##      Min      1Q Median      3Q      Max
## -5.7106 -0.5897 -0.0644  0.5676  4.0825
##
## Random effects:
## Groups   Name        Variance Std.Dev.

```

```

##  species  (Intercept) 5.308e-02 0.230388
##  plot      (Intercept) 2.348e-05 0.004846
##  Residual          2.076e-03 0.045566
## Number of obs: 1151, groups:  species, 33; plot, 24
##
## Fixed effects:
##                               Estimate Std. Error      df t value
## (Intercept)                5.229e+00 4.072e-02 3.416e+01 128.401
## stateambient              -1.289e-02 6.882e-03 2.909e+02 -1.872
## year_factor                -3.110e-03 1.350e-03 1.114e+03 -2.304
## insecticideno_insects      3.458e-03 6.868e-03 2.865e+02  0.504
## stateambient:year_factor   2.739e-03 1.520e-03 1.115e+03  1.803
## year_factor:insecticideno_insects -4.960e-05 1.518e-03 1.113e+03 -0.033
##                               Pr(>|t|)
## (Intercept)                  <2e-16 ***
## stateambient                 0.0622 .
## year_factor                  0.0214 *
## insecticideno_insects       0.6150
## stateambient:year_factor    0.0717 .
## year_factor:insecticideno_insects 0.9739
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) sttmbn yr_fct insct_ sttm:_ 
## stateambint -0.081
## year_factor  -0.128  0.479
## insctcdn_ns -0.082 -0.007  0.484
## sttmbnt:yr_  0.069 -0.866 -0.549  0.014
## yr_fctr:ns_  0.069  0.010 -0.541 -0.869 -0.024

summary(mod2k)

```

```

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula:
## log(julian_median) ~ state * year_factor + insecticide * year_factor +
##           (1 | species)
## Data: kbs_flwr_spp
##
##      AIC      BIC      logLik deviance df.resid
## -3622.0 -3581.7   1819.0   -3638.0     1143
##
## Scaled residuals:
##      Min      1Q Median      3Q      Max
## -5.6933 -0.5952 -0.0649  0.5687  4.1371
##
## Random effects:
## Groups   Name        Variance Std.Dev.
## species  (Intercept) 0.052985 0.23019
## Residual            0.002098 0.04581
## Number of obs: 1151, groups:  species, 33
##
## Fixed effects:

```

```

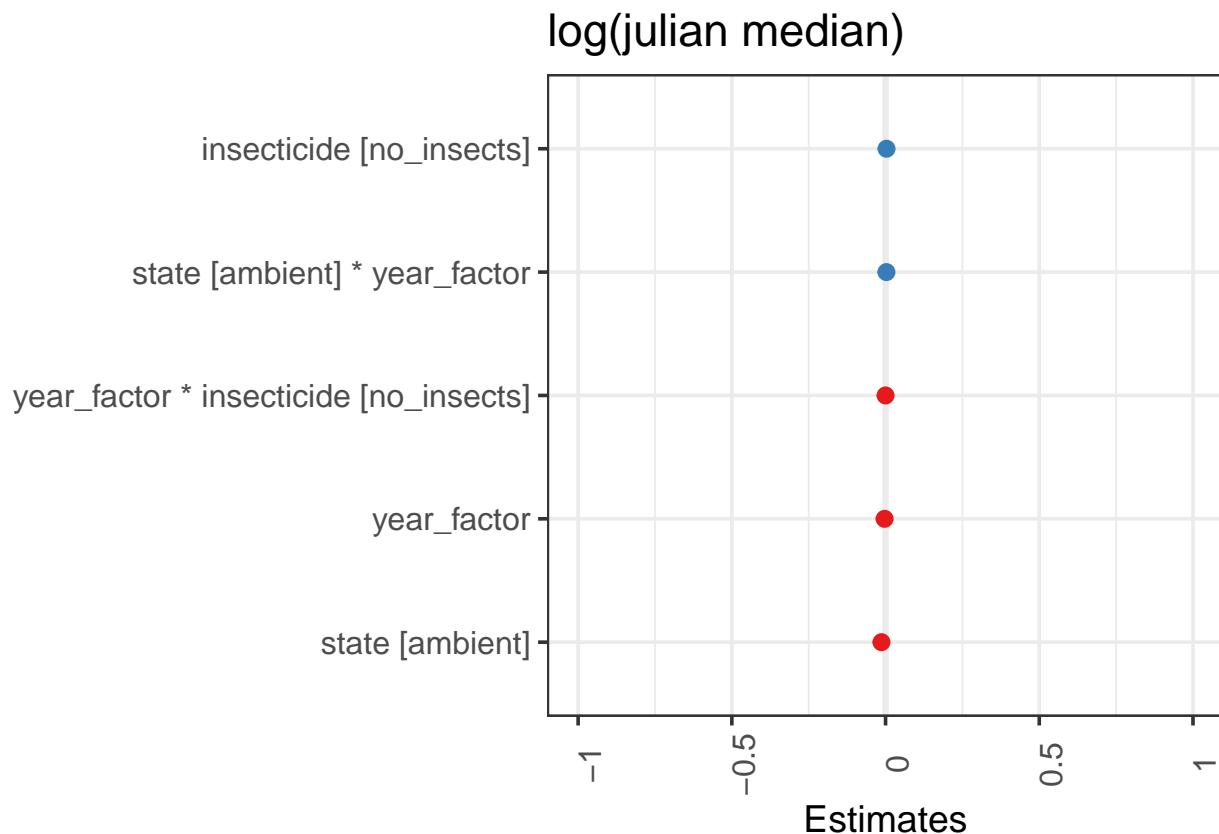
##                                     Estimate Std. Error      df t value
## (Intercept)                  5.230e+00  4.066e-02 3.405e+01 128.633
## stateambient                -1.332e-02  6.606e-03 1.118e+03 -2.016
## year_factor                 -3.197e-03  1.353e-03 1.119e+03 -2.363
## insecticideno_insects       3.108e-03  6.589e-03 1.118e+03  0.472
## stateambient:year_factor    2.790e-03  1.522e-03 1.119e+03  1.833
## year_factor:insecticideno_insects -2.865e-05 1.521e-03 1.119e+03 -0.019
##                                     Pr(>|t|)
## (Intercept)                  <2e-16 ***
## stateambient                 0.0441 *
## year_factor                  0.0183 *
## insecticideno_insects        0.6373
## stateambient:year_factor     0.0671 .
## year_factor:insecticideno_insects  0.9850
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) sttmbn yr_fct insct_ sttm:_
## stateambint -0.079
## year_factor -0.129  0.505
## insctcdn_ns -0.079 -0.004  0.505
## sttmbnt:yr_  0.070 -0.905 -0.554  0.011
## yr_fctr:ns_  0.069  0.007 -0.540 -0.908 -0.020

```

```

# Next, plot the model. There are multiple variables but here's one way to do it
# based on this package sjPlot:
# https://strengejacke.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
plot_model(mod2k, sort.est = TRUE)

```

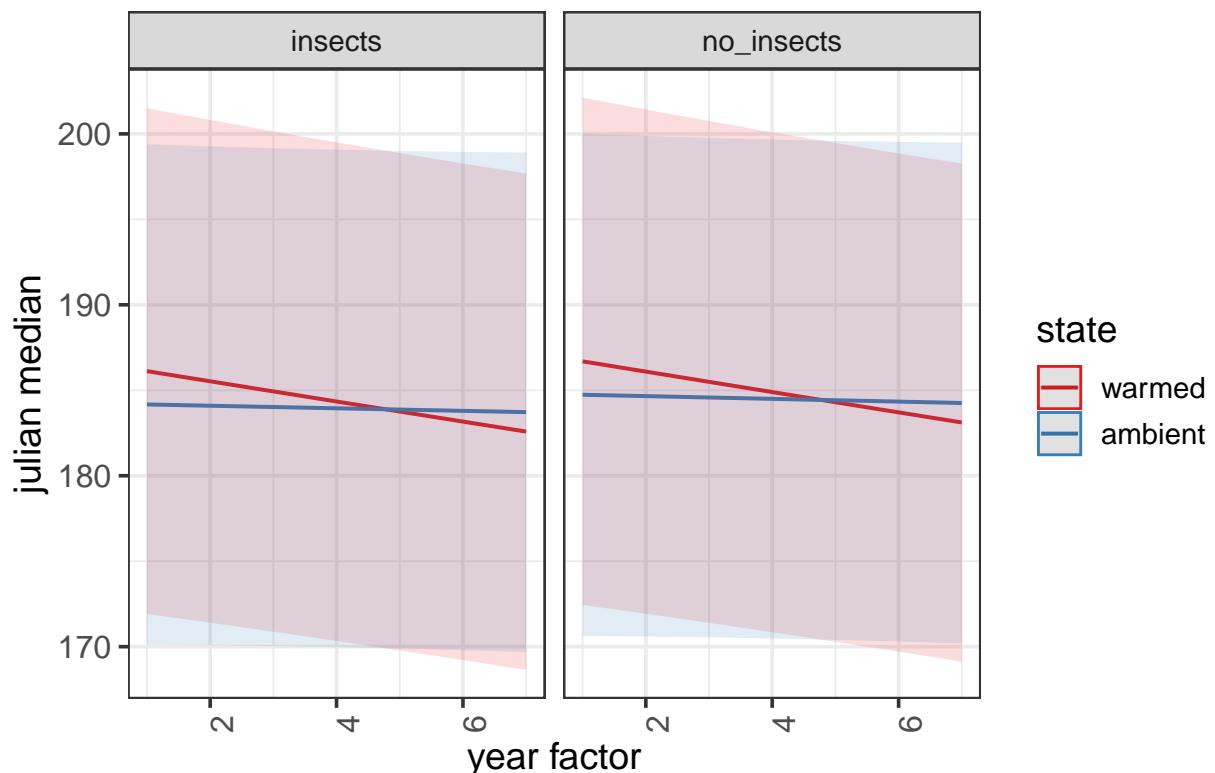


```
# these are the fixed predicted values:
```

```
plot_model(mod2k, type = "pred", terms = c("year_factor", "state", "insecticide"))
```

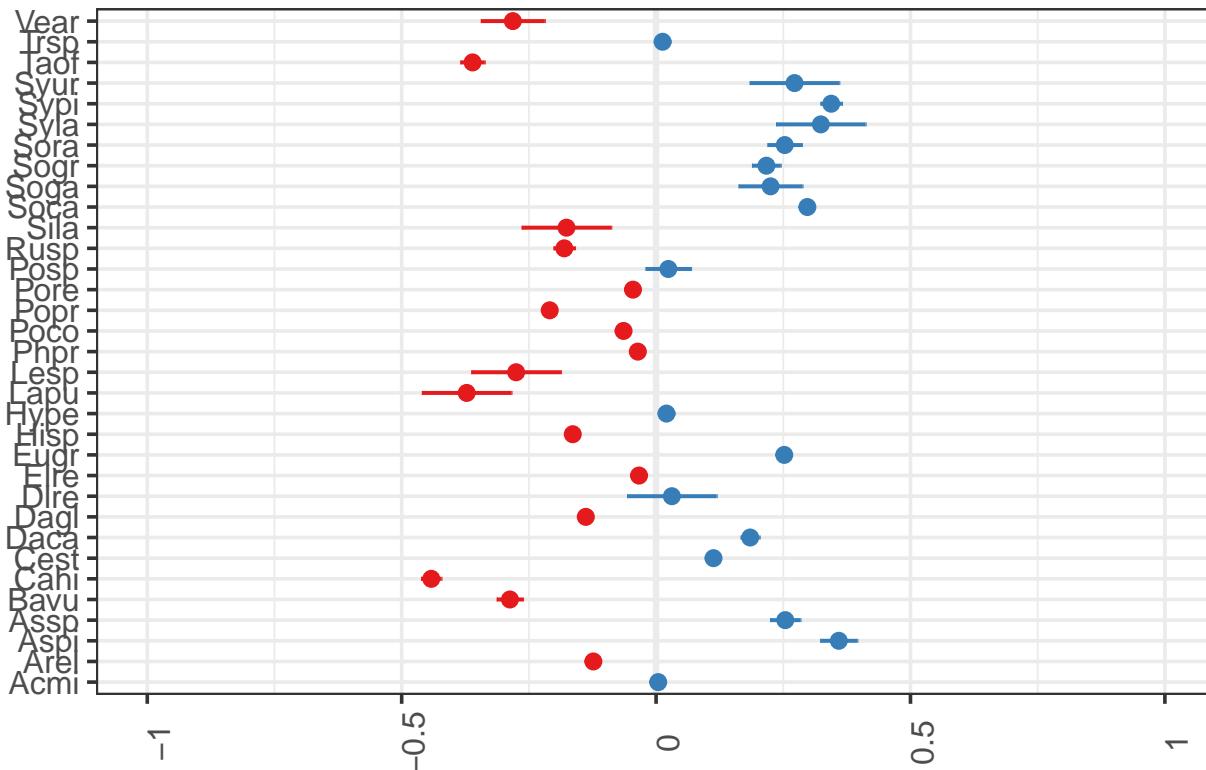
```
## Model has log-transformed response. Back-transforming predictions to original response scale. Standard
```

## Predicted values of julian median



```
# these are the random effects estimates
plot_model(mod2k, type = "re", terms = c("species"))
```

## Random effects



```
# Do we need to include insecticide?
mod3k <- lmer(log(julian_median) ~ state * year_factor + (1 | species), kbs_flwr_spp,
  REML = FALSE)
anova(mod2k, mod3k) # Don't need insecticide, continue with mod3 - the simpler model
```

```
## Data: kbs_flwr_spp
## Models:
## mod3k: log(julian_median) ~ state * year_factor + (1 | species)
## mod2k: log(julian_median) ~ state * year_factor + insecticide * year_factor +
## mod2k: (1 | species)
##      npar     AIC     BIC logLik deviance Chisq Df Pr(>Chisq)
## mod3k     6 -3624.9 -3594.6 1818.4   -3636.9
## mod2k     8 -3622.0 -3581.7 1819.0   -3638.0 1.1814  2     0.5539
```

```
# Does year need to be interactive with insecticide? - already removed
# insecticide mod4k <- lmer(log(julian_median) ~ state*year_factor + insecticide
# + (1/species) + (1/plot), kbs_flwr_spp, REML=FALSE)
```

```
# Does year need to be interactive with state?
mod5k <- lmer(log(julian_median) ~ state + year_factor + (1 | species) + (1 | plot),
  kbs_flwr_spp, REML = FALSE)
anova(mod3k, mod5k) # models are pretty similar
```

```
## Data: kbs_flwr_spp
## Models:
## mod3k: log(julian_median) ~ state * year_factor + (1 | species)
```

```

## mod5k: log(julian_median) ~ state + year_factor + (1 | species) + (1 |
## mod5k:     plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod3k     6 -3624.9 -3594.6 1818.4   -3636.9
## mod5k     6 -3624.0 -3593.7 1818.0   -3636.0     0   0

```

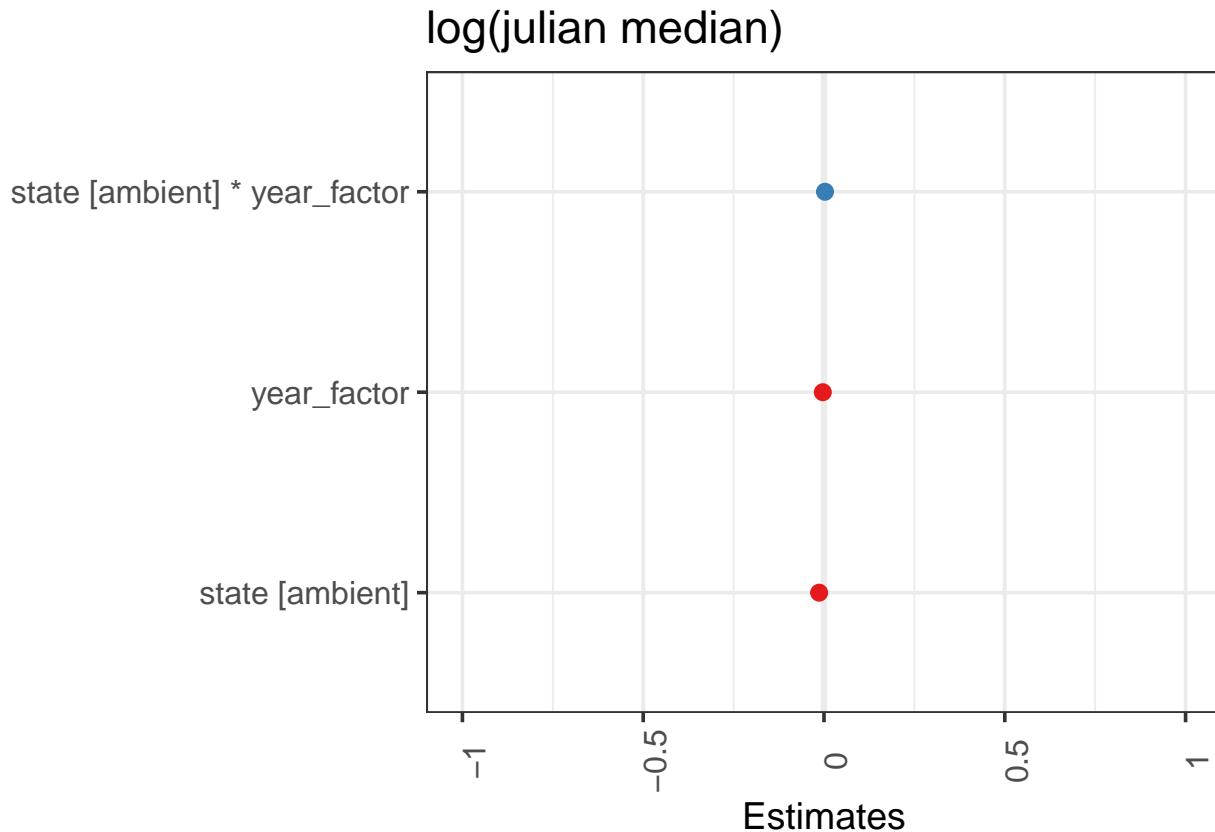
```
AICctab(mod3k, mod5k, weights = T) # go with mod3k
```

```

##          dAICc df weight
## mod3k 0.0    6  0.61
## mod5k 0.9    6  0.39

```

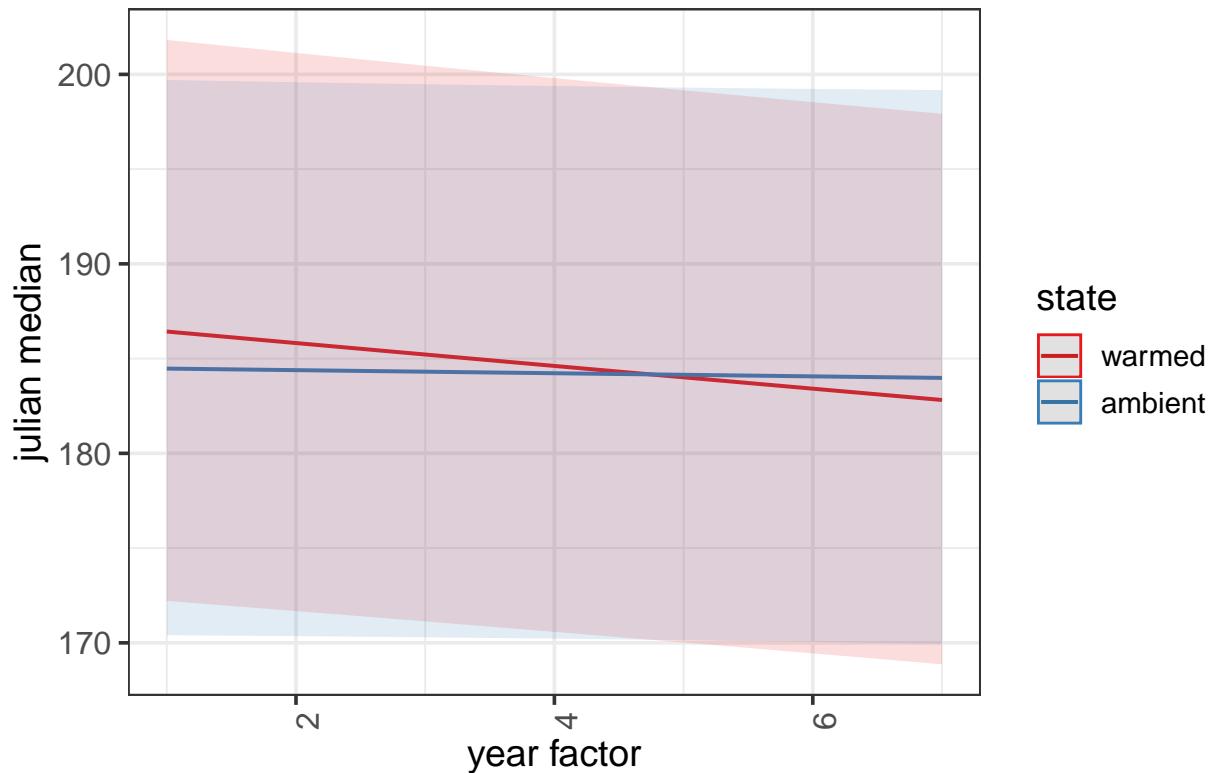
```
# Plot the fixed effects estimates for different models these are the fixed
# effects estimates from summary(mod3k)
plot_model(mod3k, sort.est = TRUE)
```



```
# these are the fixed predicted values:
plot_model(mod3k, type = "pred", terms = c("year_factor", "state"))
```

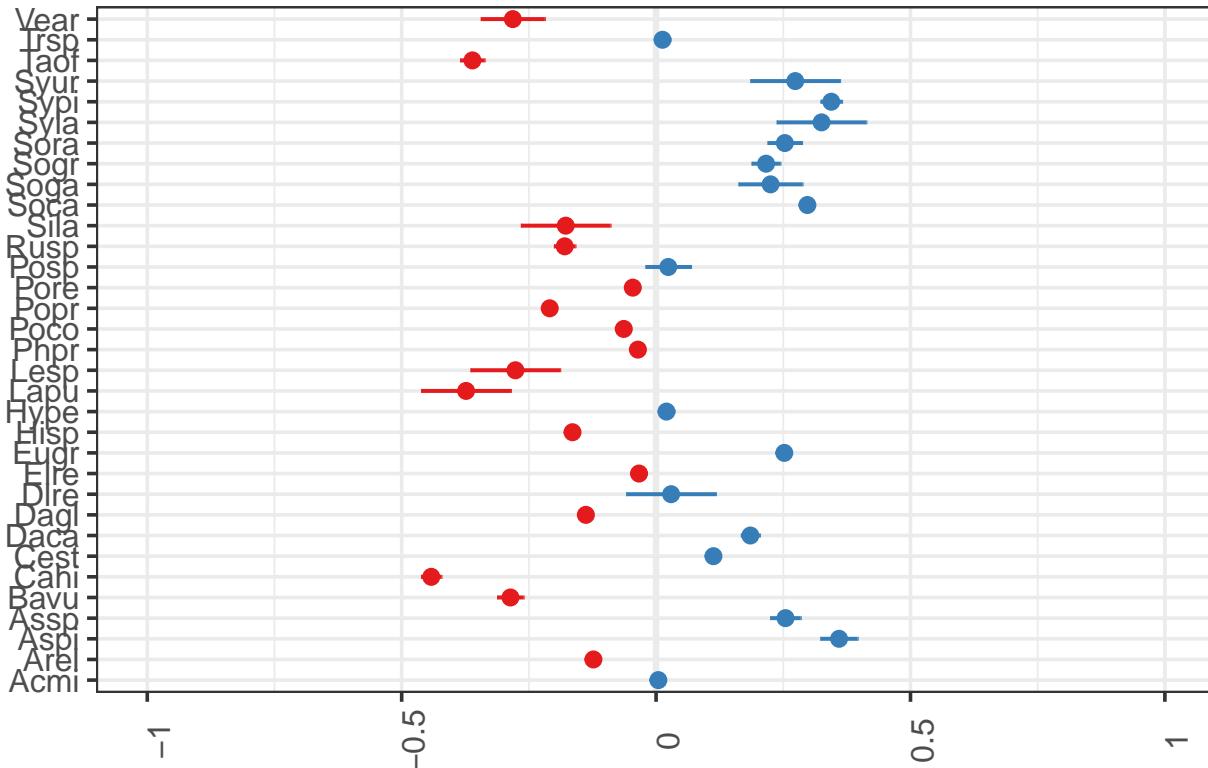
```
## Model has log-transformed response. Back-transforming predictions to original response scale. Standard
```

## Predicted values of julian median



```
# these are the random effects estimates
plot_model(mod3k, type = "re", terms = c("species"))
```

## Random effects



```
# If we wanted to include plots nested within year it would look like this:
mod6k <- lmer(log(julian_median) ~ state + year_factor + (1 | species) + (1 + year | plot), kbs_flwr_spp, REML = FALSE)
```

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

```
anova(mod3k, mod6k) #stick with mod3k
```

```
## Data: kbs_flwr_spp
## Models:
## mod3k: log(julian_median) ~ state * year_factor + (1 | species)
## mod6k: log(julian_median) ~ state + year_factor + (1 | species) + (1 +
## mod6k:      year | plot)
##          npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod3k     6 -3624.9 -3594.6 1818.4   -3636.9
## mod6k     8 -3618.3 -3578.0 1817.2   -3634.3      0    2            1
```

```
anova(mod3k)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##                   Sum Sq  Mean Sq NumDF DenDF F value  Pr(>F)
## state             0.0085836 0.0085836     1 1118.4  4.0868 0.04346 *
## year_factor       0.0101096 0.0101096     1 1119.5  4.8133 0.02845 *
## state:year_factor 0.0071908 0.0071908     1 1118.6  3.4237 0.06453 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ',' 1
```

```

# 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:
mod7k <- lmer(log(julian_median) ~ state + species + (1 + year_factor | plot), kbs_flwr_spp,
  REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod7ak <- lmer(log(julian_median) ~ state + species + year_factor + (1 | plot), kbs_flwr_spp,
  REML = FALSE)
mod7bk <- lmer(log(julian_median) ~ state * year_factor + species + (1 | plot), kbs_flwr_spp,
  REML = FALSE)
mod7ck <- lmer(log(julian_median) ~ state + species + year_factor + insecticide +
  (1 | plot), kbs_flwr_spp, REML = FALSE)
anova(mod5k, mod7k) # model 7k is a better fit to data

## Data: kbs_flwr_spp
## Models:
## mod5k: log(julian_median) ~ state + year_factor + (1 | species) + (1 |
## mod5k:           plot)
## mod7k: log(julian_median) ~ state + species + (1 + year_factor | plot)
##       npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod5k     6 -3624.0 -3593.7 1818.0   -3636.0
## mod7k    38 -3784.6 -3592.8 1930.3   -3860.6 224.63 32  < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

anova(mod7k, mod7ak) #mod 7ak

## Data: kbs_flwr_spp
## Models:
## mod7ak: log(julian_median) ~ state + species + year_factor + (1 | plot)
## mod7k: log(julian_median) ~ state + species + (1 + year_factor | plot)
##       npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod7ak   37 -3788.7 -3601.9 1931.4   -3862.7
## mod7k    38 -3784.6 -3592.8 1930.3   -3860.6      0  1          1

anova(mod7ak, mod7bk) #mod 7ak

## Data: kbs_flwr_spp
## Models:
## mod7ak: log(julian_median) ~ state + species + year_factor + (1 | plot)
## mod7bk: log(julian_median) ~ state * year_factor + species + (1 | plot)
##       npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod7ak   37 -3788.7 -3601.9 1931.4   -3862.7
## mod7bk   38 -3789.9 -3598.1 1933.0   -3865.9 3.2171  1   0.07287 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

anova(mod7ak, mod7ck) #mod7ak

## Data: kbs_flwr_spp
## Models:
## mod7ak: log(julian_median) ~ state + species + year_factor + (1 | plot)
## mod7ck: log(julian_median) ~ state + species + year_factor + insecticide +
## mod7ck: (1 | plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod7ak   37 -3788.7 -3601.9 1931.4   -3862.7
## mod7ck   38 -3787.7 -3595.8 1931.8   -3863.7 0.9445  1     0.3311

```

```
summary(mod7ak)
```

```

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + species + year_factor + (1 | plot)
## Data: kbs_flwr_spp
##
##      AIC      BIC logLik deviance df.resid
## -3788.7 -3601.9 1931.4   -3862.7     1114
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -5.8546 -0.5925 -0.0495  0.5869  4.2188
##
## Random effects:
## Groups   Name        Variance Std.Dev.
## plot     (Intercept) 2.316e-05 0.004812
## Residual           2.024e-03 0.044984
## Number of obs: 1151, groups: plot, 24
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)    
## (Intercept) 5.230e+00 6.825e-03 7.066e+02 766.331 < 2e-16 ***
## stateambient -2.263e-03 3.403e-03 2.271e+01 -0.665   0.5128  
## speciesArel -1.272e-01 8.066e-03 1.148e+03 -15.769 < 2e-16 ***
## speciesAspi  3.567e-01 1.941e-02 1.141e+03 18.376 < 2e-16 ***
## speciesAssp  2.506e-01 1.619e-02 1.136e+03 15.475 < 2e-16 ***
## speciesBavu -2.929e-01 1.450e-02 1.148e+03 -20.195 < 2e-16 ***
## speciesCahi -4.469e-01 1.180e-02 1.142e+03 -37.868 < 2e-16 ***
## speciesCest  1.084e-01 8.062e-03 1.142e+03 13.446 < 2e-16 ***
## speciesDaca  1.820e-01 1.119e-02 1.145e+03 16.272 < 2e-16 ***
## speciesDagl -1.436e-01 8.630e-03 1.149e+03 -16.640 < 2e-16 ***
## speciesDlre  2.914e-02 4.556e-02 1.145e+03  0.640   0.5226  
## speciesElre -3.871e-02 8.034e-03 1.149e+03 -4.819  1.64e-06 ***
## speciesEugr  2.484e-01 8.970e-03 1.149e+03 27.697 < 2e-16 ***
## speciesHisp -1.681e-01 8.134e-03 1.146e+03 -20.664 < 2e-16 ***
## speciesHype  1.657e-02 1.007e-02 1.149e+03  1.645   0.1003  
## speciesLapu -4.010e-01 4.571e-02 1.147e+03 -8.773 < 2e-16 ***
## speciesLesp -2.915e-01 4.558e-02 1.147e+03 -6.395  2.33e-10 ***
## speciesPhpr -4.053e-02 7.502e-03 1.144e+03 -5.402  8.01e-08 ***
## speciesPoco -6.881e-02 8.979e-03 1.149e+03 -7.663  3.84e-14 ***

```

```

## speciesPopr -2.135e-01 7.356e-03 1.145e+03 -29.026 < 2e-16 ***
## speciesPore -5.001e-02 8.765e-03 1.147e+03 -5.706 1.47e-08 ***
## speciesPosp 2.079e-02 2.341e-02 1.147e+03 0.888 0.3748
## speciesRusp -1.838e-01 1.234e-02 1.147e+03 -14.901 < 2e-16 ***
## speciesSila -1.848e-01 4.556e-02 1.145e+03 -4.055 5.35e-05 ***
## speciesSoca 2.930e-01 7.253e-03 1.144e+03 40.392 < 2e-16 ***
## speciesSoga 2.300e-01 3.250e-02 1.147e+03 7.076 2.57e-12 ***
## speciesSogr 2.139e-01 1.552e-02 1.148e+03 13.778 < 2e-16 ***
## speciesSora 2.508e-01 1.808e-02 1.141e+03 13.873 < 2e-16 ***
## speciesSyla 3.357e-01 4.568e-02 1.147e+03 7.349 3.79e-13 ***
## speciesSypi 3.414e-01 1.241e-02 1.149e+03 27.510 < 2e-16 ***
## speciesSyur 2.820e-01 4.568e-02 1.147e+03 6.172 9.32e-10 ***
## speciesTaof -3.656e-01 1.352e-02 1.151e+03 -27.038 < 2e-16 ***
## speciesTrsp 7.973e-03 9.936e-03 1.137e+03 0.802 0.4225
## speciesVear -2.927e-01 3.248e-02 1.145e+03 -9.011 < 2e-16 ***
## year_factor -1.811e-03 8.338e-04 1.151e+03 -2.172 0.0301 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Correlation matrix not shown by default, as p = 35 > 12.
## Use print(x, correlation=TRUE) or
## vcov(x) if you need it

anova(mod7ak) # investigates whether at least one of the levels within each factor is significantly di

## Type III Analysis of Variance Table with Satterthwaite's method
## Sum Sq Mean Sq NumDF DenDF F value Pr(>F)
## state 0.001 0.00089 1 22.71 0.4421 0.51282
## species 39.817 1.24428 32 1144.22 614.9026 < 2e-16 ***
## year_factor 0.010 0.00955 1 1150.95 4.7180 0.03005 *
## ---
## 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 min
# first seed dates).
emmeans(mod7ak, list(pairwise ~ state + year_factor), adjust = "tukey")

## $`emmeans of state, year_factor`
## state year_factor emmean SE df lower.CL upper.CL
## warmed 3.95 5.219 0.004586 262.2 5.210 5.228
## ambient 3.95 5.216 0.004552 263.9 5.207 5.225
##
## Results are averaged over the levels of: species
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $`pairwise differences of state, year_factor`
## 1 estimate SE df t.ratio
## warmed 3.9496090356212 - ambient 3.9496090356212 0.00226 0.00361 29.1 0.627
## p.value
```

```

##  0.5358
##
## Results are averaged over the levels of: species
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.

emmeans(mod7ak, list(pairwise ~ year_factor), adjust = "tukey")

## Note: Use 'contrast(regrid(object), ...)' to obtain contrasts of back-transformed estimates

## $`emmeans of year_factor`
##   year_factor emmean      SE   df lower.CL upper.CL
##       3.95  5.217 0.004197 532.9    5.209    5.226
##
## Results are averaged over the levels of: state, species
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $` of year_factor`
##   1      estimate SE df z.ratio p.value
##   (nothing)  nonEst NA NA NA      NA
##
## Results are averaged over the levels of: state, species
## Note: contrasts are still on the log scale
## Degrees-of-freedom method: kenward-roger

emmeans(mod7ak, list(pairwise ~ species), adjust = "tukey")

## $`emmeans of species`
##   species emmean      SE   df lower.CL upper.CL
##   Acmi    5.222 0.006312 1015.0    5.209    5.234
##   Arel    5.095 0.005379  871.9    5.084    5.105
##   Aspi    5.578 0.018799 1185.9    5.542    5.615
##   Assp    5.472 0.015340 1187.1    5.442    5.502
##   Bavu    4.929 0.013465 1124.4    4.902    4.955
##   Cahi    4.775 0.010400 1169.5    4.754    4.795
##   Cest    5.330 0.005691  926.2    5.319    5.341
##   Daca    5.404 0.009677 1159.1    5.385    5.423
##   Dagl    5.078 0.006304  983.3    5.066    5.090
##   Dlre    5.251 0.045902 1181.4    5.161    5.341
##   Elre    5.183 0.005293  878.3    5.173    5.193
##   Eugr    5.470 0.006704  999.2    5.457    5.483
##   Hisp    5.054 0.005611  918.0    5.043    5.065
##   Hype    5.238 0.008266 1044.0    5.222    5.255
##   Lapu    4.821 0.045979 1182.7    4.731    4.911
##   Lesp    4.930 0.045913 1183.4    4.840    5.020
##   Phpr    5.181 0.004595  698.4    5.172    5.190
##   Poco    5.153 0.006640 1006.4    5.140    5.166
##   Popr    5.008 0.004221  629.0    5.000    5.017
##   Pore    5.172 0.006573 1030.8    5.159    5.185
##   Posp    5.243 0.023090 1185.5    5.197    5.288

```

```

##  Rusp      5.038 0.010998 1122.6    5.016   5.059
##  Sila     5.037 0.045928 1181.6    4.947   5.127
##  Soca     5.515 0.004016  566.4    5.507   5.523
##  Soga     5.452 0.032458 1184.3    5.388   5.515
##  Sogr     5.436 0.014557 1186.4    5.407   5.464
##  Sora     5.473 0.017375 1186.6    5.438   5.507
##  Syla     5.557 0.045990 1183.7    5.467   5.648
##  Sypi     5.563 0.010984 1176.3    5.542   5.585
##  Syur     5.504 0.045990 1183.7    5.413   5.594
##  Taof     4.856 0.012404 1167.4    4.832   4.880
##  Trsp     5.230 0.008266 1160.3    5.213   5.246
##  Vear     4.929 0.032454 1183.3    4.865   4.993
##
## Results are averaged over the levels of: state
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $'pairwise differences of species'
##   1           estimate      SE   df t.ratio p.value
## Acmi - Arel  0.127189 0.00820 1183  15.515 <.0001
## Acmi - Aspi -0.356668 0.01971 1176 -18.092 <.0001
## Acmi - Assp -0.250600 0.01644 1171 -15.240 <.0001
## Acmi - Bavu  0.292851 0.01476 1186  19.835 <.0001
## Acmi - Cahi  0.446892 0.01199 1177  37.276 <.0001
## Acmi - Cest -0.108402 0.00819 1177 -13.237 <.0001
## Acmi - Daca -0.182035 0.01137 1180 -16.013 <.0001
## Acmi - Dagl  0.143605 0.00877 1184  16.368 <.0001
## Acmi - Dlre -0.029137 0.04629 1179 -0.629 1.0000
## Acmi - Elre  0.038714 0.00817 1184   4.740 0.0011
## Acmi - Eugr -0.248436 0.00912 1184 -27.244 <.0001
## Acmi - Hisp  0.168075 0.00826 1181  20.337 <.0001
## Acmi - Hype -0.016568 0.01026 1187 -1.615 0.9996
## Acmi - Lapu  0.401013 0.04646 1182   8.631 <.0001
## Acmi - Lesp  0.291467 0.04632 1181   6.293 <.0001
## Acmi - Phpr  0.040529 0.00762 1179   5.317 0.0001
## Acmi - Poco  0.068806 0.00913 1184   7.538 <.0001
## Acmi - Popr  0.213503 0.00747 1180  28.567 <.0001
## Acmi - Pore  0.050011 0.00891 1182   5.614 <.0001
## Acmi - Posp -0.020788 0.02379 1182 -0.874 1.0000
## Acmi - Rusp  0.183821 0.01256 1186  14.630 <.0001
## Acmi - Sila  0.184753 0.04630 1180   3.990 0.0259
## Acmi - Soca -0.292961 0.00737 1179 -39.757 <.0001
## Acmi - Soga -0.229988 0.03303 1182 -6.963 <.0001
## Acmi - Sogr -0.213876 0.01578 1183 -13.554 <.0001
## Acmi - Sora -0.250825 0.01837 1176 -13.658 <.0001
## Acmi - Syla -0.335713 0.04643 1182 -7.231 <.0001
## Acmi - Sypi -0.341372 0.01261 1184 -27.062 <.0001
## Acmi - Syur -0.281969 0.04643 1182 -6.073 <.0001
## Acmi - Taof  0.365594 0.01376 1187  26.573 <.0001
## Acmi - Trsp -0.007973 0.01009 1172 -0.790 1.0000
## Acmi - Vear  0.292661 0.03300 1179   8.868 <.0001
## Arel - Aspi -0.483856 0.01949 1180 -24.827 <.0001
## Arel - Assp -0.377789 0.01618 1176 -23.356 <.0001

```

```

## Arel - Bavu 0.165662 0.01443 1166 11.481 <.0001
## Arel - Cahí 0.319703 0.01165 1184 27.441 <.0001
## Arel - Cest -0.235591 0.00776 1186 -30.372 <.0001
## Arel - Daca -0.309223 0.01097 1184 -28.189 <.0001
## Arel - Dagl 0.016417 0.00813 1184 2.020 0.9844
## Arel - Dlre -0.156326 0.04621 1180 -3.383 0.1801
## Arel - Elre -0.088474 0.00736 1183 -12.018 <.0001
## Arel - Eugr -0.375625 0.00842 1187 -44.611 <.0001
## Arel - Hisp 0.040887 0.00761 1183 5.375 <.0001
## Arel - Hype -0.143757 0.00973 1187 -14.775 <.0001
## Arel - Lapu 0.273824 0.04624 1182 5.921 <.0001
## Arel - Lesp 0.164278 0.04619 1181 3.557 0.1104
## Arel - Phpr -0.086660 0.00691 1185 -12.539 <.0001
## Arel - Poco -0.058382 0.00832 1186 -7.019 <.0001
## Arel - Popr 0.086315 0.00662 1177 13.041 <.0001
## Arel - Pore -0.077178 0.00835 1184 -9.240 <.0001
## Arel - Posp -0.147976 0.02370 1179 -6.243 <.0001
## Arel - Rusp 0.056632 0.01214 1181 4.667 0.0016
## Arel - Sila 0.057565 0.04625 1181 1.245 1.0000
## Arel - Soca -0.420149 0.00648 1177 -64.788 <.0001
## Arel - Soga -0.357176 0.03285 1181 -10.873 <.0001
## Arel - Sogr -0.341065 0.01545 1183 -22.077 <.0001
## Arel - Sora -0.378014 0.01816 1181 -20.819 <.0001
## Arel - Syla -0.462902 0.04624 1182 -10.012 <.0001
## Arel - Sypi -0.468560 0.01206 1182 -38.839 <.0001
## Arel - Syur -0.409157 0.04624 1182 -8.849 <.0001
## Arel - Taof 0.238405 0.01345 1187 17.730 <.0001
## Arel - Trsp -0.135162 0.00984 1175 -13.733 <.0001
## Arel - Vear 0.165472 0.03286 1180 5.035 0.0003
## Aspi - Assp 0.106067 0.02411 1170 4.400 0.0051
## Aspi - Bavu 0.649519 0.02307 1185 28.155 <.0001
## Aspi - Cahí 0.803559 0.02145 1178 37.455 <.0001
## Aspi - Cest 0.248265 0.01955 1180 12.700 <.0001
## Aspi - Daca 0.174633 0.02102 1177 8.308 <.0001
## Aspi - Dagl 0.500273 0.01976 1179 25.317 <.0001
## Aspi - Dlre 0.327531 0.04961 1179 6.603 <.0001
## Aspi - Elre 0.395382 0.01952 1180 20.254 <.0001
## Aspi - Eugr 0.108232 0.01993 1181 5.432 <.0001
## Aspi - Hisp 0.524743 0.01952 1179 26.876 <.0001
## Aspi - Hype 0.340100 0.02050 1186 16.588 <.0001
## Aspi - Lapu 0.757681 0.04970 1181 15.246 <.0001
## Aspi - Lesp 0.648135 0.04963 1182 13.060 <.0001
## Aspi - Phpr 0.397197 0.01929 1179 20.596 <.0001
## Aspi - Poco 0.425474 0.01988 1178 21.407 <.0001
## Aspi - Popr 0.570171 0.01921 1178 29.683 <.0001
## Aspi - Pore 0.406679 0.01981 1181 20.533 <.0001
## Aspi - Posp 0.335880 0.02975 1183 11.290 <.0001
## Aspi - Rusp 0.540488 0.02163 1180 24.992 <.0001
## Aspi - Sila 0.541421 0.04963 1180 10.910 <.0001
## Aspi - Soca 0.063707 0.01917 1178 3.324 0.2099
## Aspi - Soga 0.126680 0.03750 1183 3.378 0.1825
## Aspi - Sogr 0.142792 0.02373 1181 6.017 <.0001
## Aspi - Sora 0.105842 0.02553 1175 4.146 0.0144
## Aspi - Syla 0.020955 0.04959 1179 0.423 1.0000

```

```

## Aspi - Sypi  0.015296 0.02161 1173   0.708 1.0000
## Aspi - Syur  0.074699 0.04959 1179   1.506 0.9999
## Aspi - Taof  0.722262 0.02239 1178  32.265 <.0001
## Aspi - Trsp  0.348695 0.02044 1176  17.061 <.0001
## Aspi - Vear  0.649329 0.03744 1177  17.343 <.0001
## Assp - Bavu  0.543451 0.02036 1187  26.686 <.0001
## Assp - Cahи  0.697492 0.01844 1172  37.829 <.0001
## Assp - Cest  0.142198 0.01624 1177   8.759 <.0001
## Assp - Daca  0.068566 0.01799 1173   3.811 0.0490
## Assp - Dagl  0.394206 0.01649 1174  23.911 <.0001
## Assp - Dlre  0.221463 0.04834 1176   4.582 0.0023
## Assp - Elre  0.289315 0.01620 1179  17.864 <.0001
## Assp - Eugr  0.002164 0.01668 1179   0.130 1.0000
## Assp - Hisp  0.418676 0.01623 1178  25.790 <.0001
## Assp - Hype  0.234032 0.01736 1185  13.484 <.0001
## Assp - Lapu  0.651613 0.04850 1181  13.436 <.0001
## Assp - Lesp  0.542067 0.04841 1182  11.197 <.0001
## Assp - Phpr  0.291129 0.01592 1174  18.289 <.0001
## Assp - Poco  0.319407 0.01665 1177  19.180 <.0001
## Assp - Popr  0.464104 0.01584 1177  29.293 <.0001
## Assp - Pore  0.300611 0.01655 1175  18.159 <.0001
## Assp - Posp  0.229813 0.02765 1181   8.312 <.0001
## Assp - Rusp  0.434421 0.01875 1183  23.164 <.0001
## Assp - Sila  0.435354 0.04836 1177   9.003 <.0001
## Assp - Soca -0.042360 0.01579 1177 -2.682 0.6896
## Assp - Soga  0.020613 0.03589 1183   0.574 1.0000
## Assp - Sogr  0.036724 0.02109 1181   1.741 0.9986
## Assp - Sora -0.000225 0.02309 1175 -0.010 1.0000
## Assp - Syla -0.085113 0.04843 1180 -1.757 0.9983
## Assp - Sypi -0.090771 0.01875 1176 -4.841 0.0007
## Assp - Syur -0.031368 0.04843 1180 -0.648 1.0000
## Assp - Taof  0.616194 0.01960 1181  31.433 <.0001
## Assp - Trsp  0.242627 0.01729 1173  14.033 <.0001
## Assp - Vear  0.543261 0.03581 1175  15.171 <.0001
## Bavu - Cahи  0.154041 0.01690 1186   9.117 <.0001
## Bavu - Cest  -0.401253 0.01458 1171 -27.514 <.0001
## Bavu - Daca -0.474886 0.01652 1182 -28.743 <.0001
## Bavu - Dagl -0.149246 0.01479 1170 -10.089 <.0001
## Bavu - Dlre -0.321988 0.04782 1184 -6.733 <.0001
## Bavu - Elre -0.254137 0.01434 1176 -17.724 <.0001
## Bavu - Eugr -0.541287 0.01491 1180 -36.305 <.0001
## Bavu - Hisp -0.124775 0.01454 1169 -8.583 <.0001
## Bavu - Hype -0.309419 0.01569 1178 -19.718 <.0001
## Bavu - Lapu  0.108162 0.04787 1185   2.259 0.9344
## Bavu - Lesp -0.001384 0.04774 1181 -0.029 1.0000
## Bavu - Phpr -0.252322 0.01415 1171 -17.835 <.0001
## Bavu - Poco -0.224045 0.01491 1170 -15.021 <.0001
## Bavu - Popr -0.079348 0.01402 1172 -5.660 <.0001
## Bavu - Pore -0.242840 0.01496 1170 -16.234 <.0001
## Bavu - Posp -0.313639 0.02675 1187 -11.727 <.0001
## Bavu - Rusp -0.109030 0.01727 1183 -6.314 <.0001
## Bavu - Sila -0.108098 0.04785 1184 -2.259 0.9346
## Bavu - Soca -0.585812 0.01395 1173 -41.996 <.0001
## Bavu - Soga -0.522839 0.03513 1186 -14.883 <.0001

```

```

##  Bavu - Sogr -0.506727 0.01976 1187 -25.641 <.0001
##  Bavu - Sora -0.543677 0.02187 1184 -24.856 <.0001
##  Bavu - Syla -0.628564 0.04791 1185 -13.119 <.0001
##  Bavu - Sypi -0.634223 0.01735 1182 -36.564 <.0001
##  Bavu - Syur -0.574820 0.04791 1185 -11.997 <.0001
##  Bavu - Taof  0.072743 0.01827 1179   3.981 0.0269
##  Bavu - Trsp -0.300824 0.01580 1176 -19.040 <.0001
##  Bavu - Vear -0.000190 0.03513 1186  -0.005 1.0000
##  Cahи - Cest -0.555294 0.01168 1184 -47.525 <.0001
##  Cahи - Daca -0.628926 0.01410 1182 -44.603 <.0001
##  Cahи - Dagl -0.303286 0.01204 1183 -25.188 <.0001
##  Cahи - Dlre -0.476029 0.04696 1177 -10.137 <.0001
##  Cahи - Elre -0.408178 0.01155 1182 -35.353 <.0001
##  Cahи - Eogr -0.695328 0.01226 1182 -56.696 <.0001
##  Cahи - Hisp -0.278816 0.01173 1184 -23.762 <.0001
##  Cahи - Hype -0.463460 0.01314 1185 -35.270 <.0001
##  Cahи - Lapu -0.045879 0.04717 1183  -0.973 1.0000
##  Cahи - Lesp -0.155425 0.04698 1180  -3.308 0.2185
##  Cahи - Phpr -0.406363 0.01124 1181 -36.152 <.0001
##  Cahи - Poco -0.378085 0.01230 1183 -30.748 <.0001
##  Cahи - Popr -0.233388 0.01115 1184 -20.933 <.0001
##  Cahи - Pore -0.396881 0.01222 1183 -32.488 <.0001
##  Cahи - Posp -0.467679 0.02519 1181 -18.566 <.0001
##  Cahи - Rusp -0.263071 0.01504 1185 -17.494 <.0001
##  Cahи - Sila -0.262138 0.04697 1178  -5.581 <.0001
##  Cahи - Soca -0.739852 0.01107 1183 -66.841 <.0001
##  Cahи - Soga -0.676879 0.03407 1184 -19.866 <.0001
##  Cahи - Sogr -0.660768 0.01781 1184 -37.098 <.0001
##  Cahи - Sora -0.697717 0.02011 1175 -34.687 <.0001
##  Cahи - Syla -0.782605 0.04721 1183 -16.579 <.0001
##  Cahи - Sypi -0.788264 0.01516 1183 -52.005 <.0001
##  Cahи - Syur -0.728861 0.04721 1183 -15.440 <.0001
##  Cahи - Taof -0.081298 0.01609 1187  -5.052 0.0002
##  Cahи - Trsp -0.454865 0.01306 1178 -34.816 <.0001
##  Cahи - Vear -0.154231 0.03401 1179  -4.535 0.0028
##  Cest - Daca -0.073632 0.01103 1180  -6.677 <.0001
##  Cest - Dagl  0.252008 0.00834 1185  30.226 <.0001
##  Cest - Dlre  0.079265 0.04620 1179   1.716 0.9989
##  Cest - Elre  0.147117 0.00771 1183  19.090 <.0001
##  Cest - Eogr -0.140034 0.00874 1187 -16.023 <.0001
##  Cest - Hisp  0.276478 0.00779 1178  35.482 <.0001
##  Cest - Hype  0.091834 0.00989 1184  9.283 <.0001
##  Cest - Lapu  0.509415 0.04637 1181  10.985 <.0001
##  Cest - Lesp  0.399869 0.04625 1182  8.646 <.0001
##  Cest - Phpr  0.148931 0.00711 1178  20.945 <.0001
##  Cest - Poco  0.177209 0.00875 1187  20.254 <.0001
##  Cest - Popr  0.321906 0.00696 1178  46.222 <.0001
##  Cest - Pore  0.158413 0.00846 1180  18.717 <.0001
##  Cest - Posp  0.087615 0.02360 1180  3.713 0.0678
##  Cest - Rusp  0.292223 0.01232 1174  23.722 <.0001
##  Cest - Sila  0.293156 0.04621 1179  6.344 <.0001
##  Cest - Soca -0.184558 0.00686 1178 -26.912 <.0001
##  Cest - Soga -0.121585 0.03293 1183  -3.693 0.0725
##  Cest - Sogr -0.105473 0.01555 1185  -6.782 <.0001

```

```

## Cest - Sora -0.142423 0.01818 1181 -7.835 <.0001
## Cest - Syla -0.227311 0.04636 1182 -4.903 0.0005
## Cest - Sypi -0.232969 0.01234 1185 -18.884 <.0001
## Cest - Syur -0.173566 0.04636 1182 -3.744 0.0614
## Cest - Taof 0.473996 0.01349 1187 35.126 <.0001
## Cest - Trsp 0.100429 0.00966 1169 10.392 <.0001
## Cest - Vear 0.401064 0.03289 1180 12.195 <.0001
## Daca - Dagl 0.325640 0.01141 1180 28.530 <.0001
## Daca - Dlre 0.152897 0.04686 1178 3.263 0.2441
## Daca - Elre 0.220749 0.01097 1185 20.126 <.0001
## Daca - Eogr -0.066402 0.01170 1187 -5.676 <.0001
## Daca - Hisp 0.350110 0.01105 1185 31.672 <.0001
## Daca - Hype 0.165466 0.01261 1187 13.123 <.0001
## Daca - Lapu 0.583047 0.04702 1182 12.401 <.0001
## Daca - Lesp 0.473501 0.04692 1183 10.091 <.0001
## Daca - Phpr 0.222564 0.01057 1181 21.048 <.0001
## Daca - Poco 0.250841 0.01166 1184 21.515 <.0001
## Daca - Popr 0.395538 0.01045 1183 37.837 <.0001
## Daca - Pore 0.232045 0.01153 1183 20.125 <.0001
## Daca - Posp 0.161247 0.02491 1178 6.473 <.0001
## Daca - Rusp 0.365855 0.01453 1187 25.185 <.0001
## Daca - Sila 0.366788 0.04687 1178 7.825 <.0001
## Daca - Soca -0.110926 0.01038 1183 -10.688 <.0001
## Daca - Soga -0.047953 0.03385 1183 -1.417 1.0000
## Daca - Sogr -0.031841 0.01742 1186 -1.828 0.9967
## Daca - Sora -0.068791 0.01979 1178 -3.475 0.1398
## Daca - Syla -0.153678 0.04699 1183 -3.270 0.2398
## Daca - Sypi -0.159337 0.01454 1185 -10.957 <.0001
## Daca - Syur -0.099934 0.04699 1183 -2.127 0.9685
## Daca - Taof 0.547628 0.01562 1187 35.063 <.0001
## Daca - Trsp 0.174061 0.01254 1178 13.879 <.0001
## Daca - Vear 0.474696 0.03382 1180 14.037 <.0001
## Dagl - Dlre -0.172742 0.04629 1179 -3.732 0.0638
## Dagl - Elre -0.104891 0.00807 1184 -12.996 <.0001
## Dagl - Eogr -0.392041 0.00906 1187 -43.263 <.0001
## Dagl - Hisp 0.024470 0.00830 1187 2.948 0.4706
## Dagl - Hype -0.160174 0.01026 1186 -15.606 <.0001
## Dagl - Lapu 0.257408 0.04635 1179 5.554 <.0001
## Dagl - Lesp 0.147861 0.04633 1183 3.192 0.2885
## Dagl - Phpr -0.103076 0.00761 1175 -13.547 <.0001
## Dagl - Poco -0.074799 0.00897 1176 -8.336 <.0001
## Dagl - Popr 0.069898 0.00742 1185 9.415 <.0001
## Dagl - Pore -0.093595 0.00894 1177 -10.472 <.0001
## Dagl - Posp -0.164393 0.02387 1179 -6.888 <.0001
## Dagl - Rusp 0.040215 0.01256 1186 3.201 0.2822
## Dagl - Sila 0.041148 0.04632 1179 0.888 1.0000
## Dagl - Soca -0.436566 0.00730 1184 -59.764 <.0001
## Dagl - Soga -0.373593 0.03304 1183 -11.307 <.0001
## Dagl - Sogr -0.357481 0.01578 1183 -22.660 <.0001
## Dagl - Sora -0.394431 0.01842 1180 -21.419 <.0001
## Dagl - Syla -0.479318 0.04642 1184 -10.327 <.0001
## Dagl - Sypi -0.484977 0.01259 1187 -38.528 <.0001
## Dagl - Syur -0.425574 0.04642 1184 -9.169 <.0001
## Dagl - Taof 0.221988 0.01382 1187 16.061 <.0001

```

```

## Dagl - Trsp -0.151579 0.01027 1178 -14.760 <.0001
## Dagl - Vear 0.149056 0.03299 1178 4.518 0.0031
## Dlre - Elre 0.067851 0.04617 1180 1.470 0.9999
## Dlre - Eugr -0.219299 0.04635 1179 -4.731 0.0012
## Dlre - Hisp 0.197213 0.04622 1180 4.266 0.0089
## Dlre - Hype 0.012569 0.04657 1178 0.270 1.0000
## Dlre - Lapu 0.430150 0.06499 1182 6.618 <.0001
## Dlre - Lesp 0.320604 0.06490 1182 4.940 0.0004
## Dlre - Phpr 0.069666 0.04609 1179 1.511 0.9999
## Dlre - Poco 0.097943 0.04640 1181 2.111 0.9714
## Dlre - Popr 0.242640 0.04608 1180 5.266 0.0001
## Dlre - Pore 0.079148 0.04633 1178 1.708 0.9990
## Dlre - Posp 0.008349 0.05132 1180 0.163 1.0000
## Dlre - Rusp 0.212958 0.04720 1182 4.512 0.0032
## Dlre - Sila 0.213890 0.06455 1160 3.313 0.2157
## Dlre - Soca -0.263824 0.04606 1180 -5.728 <.0001
## Dlre - Soga -0.200851 0.05621 1181 -3.573 0.1051
## Dlre - Sogr -0.184739 0.04810 1178 -3.841 0.0442
## Dlre - Sora -0.221689 0.04899 1177 -4.525 0.0030
## Dlre - Syla -0.306576 0.06503 1182 -4.714 0.0013
## Dlre - Sypi -0.312235 0.04723 1180 -6.611 <.0001
## Dlre - Syur -0.252832 0.06503 1182 -3.888 0.0375
## Dlre - Taof 0.394731 0.04753 1181 8.305 <.0001
## Dlre - Trsp 0.021164 0.04656 1178 0.455 1.0000
## Dlre - Vear 0.321798 0.05601 1168 5.745 <.0001
## Elre - Eugr -0.287150 0.00834 1186 -34.421 <.0001
## Elre - Hisp 0.129361 0.00759 1183 17.042 <.0001
## Elre - Hype -0.055282 0.00970 1184 -5.700 <.0001
## Elre - Lapu 0.362299 0.04621 1181 7.840 <.0001
## Elre - Lesp 0.252753 0.04618 1182 5.474 <.0001
## Elre - Phpr 0.001815 0.00683 1178 0.266 1.0000
## Elre - Poco 0.030092 0.00827 1183 3.640 0.0857
## Elre - Popr 0.174789 0.00656 1174 26.645 <.0001
## Elre - Pore 0.011297 0.00835 1179 1.354 1.0000
## Elre - Posp -0.059502 0.02369 1181 -2.512 0.8122
## Elre - Rusp 0.145106 0.01210 1187 11.992 <.0001
## Elre - Sila 0.146039 0.04621 1180 3.161 0.3093
## Elre - Soca -0.331675 0.00642 1173 -51.674 <.0001
## Elre - Soga -0.268702 0.03285 1182 -8.180 <.0001
## Elre - Sogr -0.252590 0.01541 1182 -16.391 <.0001
## Elre - Sora -0.289540 0.01810 1178 -15.992 <.0001
## Elre - Syla -0.374427 0.04627 1183 -8.093 <.0001
## Elre - Sypi -0.380086 0.01211 1181 -31.384 <.0001
## Elre - Syur -0.320683 0.04627 1183 -6.931 <.0001
## Elre - Taof 0.326880 0.01344 1187 24.325 <.0001
## Elre - Trsp -0.046687 0.00979 1174 -4.769 0.0010
## Elre - Vear 0.253947 0.03286 1181 7.728 <.0001
## Eugr - Hisp 0.416512 0.00862 1187 48.338 <.0001
## Eugr - Hype 0.231868 0.01053 1184 22.026 <.0001
## Eugr - Lapu 0.649449 0.04640 1181 13.998 <.0001
## Eugr - Lesp 0.539903 0.04639 1184 11.639 <.0001
## Eugr - Phpr 0.288965 0.00798 1187 36.198 <.0001
## Eugr - Poco 0.317242 0.00925 1187 34.315 <.0001
## Eugr - Popr 0.461939 0.00774 1187 59.656 <.0001

```

```

## Eugr - Pore 0.298447 0.00930 1187 32.097 <.0001
## Eugr - Posp 0.227648 0.02405 1182 9.467 <.0001
## Eugr - Rusp 0.432257 0.01274 1187 33.917 <.0001
## Eugr - Sila 0.433189 0.04639 1180 9.339 <.0001
## Eugr - Soca -0.044525 0.00762 1187 -5.841 <.0001
## Eugr - Soga 0.018448 0.03308 1180 0.558 1.0000
## Eugr - Sogr 0.034560 0.01591 1176 2.172 0.9589
## Eugr - Sora -0.002390 0.01853 1170 -0.129 1.0000
## Eugr - Syla -0.087277 0.04645 1184 -1.879 0.9948
## Eugr - Sypi -0.092936 0.01277 1187 -7.277 <.0001
## Eugr - Syur -0.033533 0.04645 1184 -0.722 1.0000
## Eugr - Taof 0.614030 0.01402 1186 43.806 <.0001
## Eugr - Trsp 0.240463 0.01062 1180 22.649 <.0001
## Eugr - Vear 0.541097 0.03311 1181 16.343 <.0001
## Hisp - Hype -0.184644 0.00986 1184 -18.717 <.0001
## Hisp - Lapu 0.232937 0.04631 1181 5.030 0.0003
## Hisp - Lesp 0.123391 0.04623 1181 2.669 0.6999
## Hisp - Phpr -0.127547 0.00708 1185 -18.013 <.0001
## Hisp - Poco -0.099269 0.00857 1187 -11.580 <.0001
## Hisp - Popr 0.045428 0.00684 1176 6.640 <.0001
## Hisp - Pore -0.118065 0.00845 1179 -13.979 <.0001
## Hisp - Posp -0.188863 0.02371 1181 -7.967 <.0001
## Hisp - Rusp 0.015745 0.01224 1181 1.287 1.0000
## Hisp - Sila 0.016678 0.04625 1180 0.361 1.0000
## Hisp - Soca -0.461036 0.00672 1178 -68.573 <.0001
## Hisp - Soga -0.398063 0.03288 1180 -12.105 <.0001
## Hisp - Sogr -0.381952 0.01551 1182 -24.622 <.0001
## Hisp - Sora -0.418901 0.01819 1179 -23.027 <.0001
## Hisp - Syla -0.503789 0.04631 1183 -10.879 <.0001
## Hisp - Sypi -0.509447 0.01221 1185 -41.717 <.0001
## Hisp - Syur -0.450044 0.04631 1183 -9.718 <.0001
## Hisp - Taof 0.197518 0.01349 1187 14.641 <.0001
## Hisp - Trsp -0.176049 0.00983 1169 -17.907 <.0001
## Hisp - Vear 0.124586 0.03289 1180 3.788 0.0530
## Hype - Lapu 0.417581 0.04670 1182 8.942 <.0001
## Hype - Lesp 0.308035 0.04655 1178 6.617 <.0001
## Hype - Phpr 0.057097 0.00932 1185 6.130 <.0001
## Hype - Poco 0.085375 0.01051 1182 8.126 <.0001
## Hype - Popr 0.230072 0.00915 1184 25.133 <.0001
## Hype - Pore 0.066579 0.01044 1184 6.375 <.0001
## Hype - Posp -0.004220 0.02444 1182 -0.173 1.0000
## Hype - Rusp 0.200389 0.01363 1186 14.703 <.0001
## Hype - Sila 0.201322 0.04659 1178 4.321 0.0071
## Hype - Soca -0.276392 0.00906 1184 -30.511 <.0001
## Hype - Soga -0.213420 0.03343 1182 -6.384 <.0001
## Hype - Sogr -0.197308 0.01664 1185 -11.855 <.0001
## Hype - Sora -0.234257 0.01916 1183 -12.227 <.0001
## Hype - Syla -0.319145 0.04673 1184 -6.830 <.0001
## Hype - Sypi -0.324804 0.01369 1187 -23.730 <.0001
## Hype - Syur -0.265401 0.04673 1184 -5.679 <.0001
## Hype - Taof 0.382162 0.01482 1185 25.789 <.0001
## Hype - Trsp 0.008595 0.01156 1187 0.744 1.0000
## Hype - Vear 0.309229 0.03344 1181 9.248 <.0001
## Lapu - Lesp -0.109546 0.06496 1183 -1.686 0.9992

```

```

## Lapu - Phpr -0.360484 0.04618 1181 -7.806 <.0001
## Lapu - Poco -0.332207 0.04635 1180 -7.168 <.0001
## Lapu - Popr -0.187510 0.04613 1182 -4.065 0.0196
## Lapu - Pore -0.351002 0.04644 1180 -7.558 <.0001
## Lapu - Posp -0.421801 0.05157 1182 -8.179 <.0001
## Lapu - Rusp -0.217192 0.04726 1183 -4.595 0.0022
## Lapu - Sila -0.216259 0.06505 1182 -3.325 0.2095
## Lapu - Soca -0.693974 0.04610 1181 -15.053 <.0001
## Lapu - Soga -0.631001 0.05627 1182 -11.213 <.0001
## Lapu - Sogr -0.614889 0.04818 1180 -12.761 <.0001
## Lapu - Sora -0.651838 0.04919 1182 -13.253 <.0001
## Lapu - Syla -0.736726 0.06494 1182 -11.345 <.0001
## Lapu - Sypi -0.742385 0.04723 1181 -15.718 <.0001
## Lapu - Syur -0.682982 0.06494 1182 -10.517 <.0001
## Lapu - Taof -0.035419 0.04760 1179 -0.744 1.0000
## Lapu - Trsp -0.408986 0.04680 1181 -8.738 <.0001
## Lapu - Vear -0.108352 0.05630 1182 -1.925 0.9924
## Lesp - Phpr -0.250938 0.04612 1182 -5.441 <.0001
## Lesp - Poco -0.222660 0.04637 1183 -4.802 0.0008
## Lesp - Popr -0.077964 0.04608 1182 -1.692 0.9991
## Lesp - Pore -0.241456 0.04638 1182 -5.207 0.0001
## Lesp - Posp -0.312254 0.05139 1183 -6.076 <.0001
## Lesp - Rusp -0.107646 0.04721 1184 -2.280 0.9273
## Lesp - Sila -0.106713 0.06493 1182 -1.644 0.9995
## Lesp - Soca -0.584427 0.04606 1182 -12.689 <.0001
## Lesp - Soga -0.521455 0.05622 1183 -9.275 <.0001
## Lesp - Sogr -0.505343 0.04816 1183 -10.493 <.0001
## Lesp - Sora -0.542292 0.04908 1183 -11.049 <.0001
## Lesp - Syla -0.627180 0.06500 1182 -9.650 <.0001
## Lesp - Sypi -0.632839 0.04722 1182 -13.402 <.0001
## Lesp - Syur -0.573436 0.06500 1182 -8.823 <.0001
## Lesp - Taof 0.074127 0.04756 1184 1.559 0.9998
## Lesp - Trsp -0.299440 0.04664 1182 -6.421 <.0001
## Lesp - Vear 0.001194 0.05622 1182 0.021 1.0000
## Phpr - Poco 0.028277 0.00790 1177 3.579 0.1033
## Phpr - Popr 0.172974 0.00604 1178 28.616 <.0001
## Phpr - Pore 0.009482 0.00783 1175 1.211 1.0000
## Phpr - Posp -0.061317 0.02348 1182 -2.611 0.7442
## Phpr - Rusp 0.143292 0.01180 1185 12.140 <.0001
## Phpr - Sila 0.144224 0.04612 1180 3.127 0.3325
## Phpr - Soca -0.333490 0.00590 1175 -56.513 <.0001
## Phpr - Soga -0.270517 0.03276 1183 -8.257 <.0001
## Phpr - Sogr -0.254405 0.01519 1184 -16.751 <.0001
## Phpr - Sora -0.291355 0.01790 1179 -16.279 <.0001
## Phpr - Syla -0.376242 0.04620 1183 -8.144 <.0001
## Phpr - Sypi -0.381901 0.01182 1185 -32.318 <.0001
## Phpr - Syur -0.322498 0.04620 1183 -6.981 <.0001
## Phpr - Taof 0.325065 0.01314 1187 24.743 <.0001
## Phpr - Trsp -0.048502 0.00930 1171 -5.217 0.0001
## Phpr - Vear 0.252132 0.03273 1179 7.704 <.0001
## Poco - Popr 0.144697 0.00766 1186 18.896 <.0001
## Poco - Pore -0.018796 0.00922 1181 -2.038 0.9823
## Poco - Posp -0.089594 0.02408 1181 -3.720 0.0664
## Poco - Rusp 0.115014 0.01267 1187 9.075 <.0001

```

```

## Poco - Sila  0.115947 0.04644 1182   2.497 0.8220
## Poco - Soca -0.361767 0.00753 1185 -48.041 <.0001
## Poco - Soga -0.298794 0.03310 1183  -9.027 <.0001
## Poco - Sogr -0.282682 0.01595 1185 -17.727 <.0001
## Poco - Sora -0.319632 0.01859 1182 -17.191 <.0001
## Poco - Syla -0.404519 0.04638 1183  -8.722 <.0001
## Poco - Sypi -0.410178 0.01266 1186 -32.387 <.0001
## Poco - Syur -0.350775 0.04638 1183  -7.564 <.0001
## Poco - Taof  0.296787 0.01403 1187  21.148 <.0001
## Poco - Trsp -0.076780 0.01069 1181  -7.184 <.0001
## Poco - Vear  0.223855 0.03311 1182   6.761 <.0001
## Popr - Pore -0.163493 0.00767 1182 -21.327 <.0001
## Popr - Posp -0.234291 0.02345 1180  -9.990 <.0001
## Popr - Rusp -0.029683 0.01168 1181  -2.542 0.7924
## Popr - Sila -0.028750 0.04611 1180  -0.623 1.0000
## Popr - Soca -0.506464 0.00558 1161 -90.819 <.0001
## Popr - Soga -0.443491 0.03269 1181 -13.567 <.0001
## Popr - Sogr -0.427379 0.01508 1183 -28.335 <.0001
## Popr - Sora -0.464329 0.01783 1179 -26.038 <.0001
## Popr - Syla -0.549216 0.04614 1182 -11.904 <.0001
## Popr - Sypi -0.554875 0.01163 1182 -47.713 <.0001
## Popr - Syur -0.495472 0.04614 1182 -10.739 <.0001
## Popr - Taof  0.152090 0.01304 1187  11.668 <.0001
## Popr - Trsp -0.221477 0.00922 1168 -24.016 <.0001
## Popr - Vear  0.079158 0.03270 1181   2.421 0.8653
## Pore - Posp -0.070799 0.02392 1182  -2.959 0.4611
## Pore - Rusp  0.133810 0.01269 1183  10.545 <.0001
## Pore - Sila  0.134743 0.04635 1179   2.907 0.5044
## Pore - Soca -0.342971 0.00756 1182 -45.356 <.0001
## Pore - Soga -0.279999 0.03309 1183  -8.462 <.0001
## Pore - Sogr -0.263887 0.01589 1183 -16.612 <.0001
## Pore - Sora -0.300836 0.01851 1181 -16.251 <.0001
## Pore - Syla -0.385724 0.04644 1183  -8.306 <.0001
## Pore - Sypi -0.391383 0.01269 1187 -30.843 <.0001
## Pore - Syur -0.331980 0.04644 1183  -7.149 <.0001
## Pore - Taof  0.315583 0.01390 1187  22.698 <.0001
## Pore - Trsp -0.057984 0.01037 1175  -5.594 <.0001
## Pore - Vear  0.242650 0.03305 1179   7.342 <.0001
## Posp - Rusp  0.204608 0.02554 1185   8.012 <.0001
## Posp - Sila  0.205541 0.05131 1180   4.006 0.0245
## Posp - Soca -0.272173 0.02343 1180 -11.617 <.0001
## Posp - Soga -0.209200 0.03976 1178  -5.262 0.0001
## Posp - Sogr -0.193088 0.02725 1183  -7.087 <.0001
## Posp - Sora -0.230038 0.02881 1183  -7.983 <.0001
## Posp - Syla -0.314925 0.05158 1183  -6.105 <.0001
## Posp - Sypi -0.320584 0.02564 1184 -12.501 <.0001
## Posp - Syur -0.261181 0.05158 1183  -5.063 0.0002
## Posp - Taof  0.386381 0.02609 1184  14.809 <.0001
## Posp - Trsp  0.012814 0.02429 1181   0.528 1.0000
## Posp - Vear  0.313449 0.03979 1182   7.877 <.0001
## Rusp - Sila  0.000933 0.04723 1182   0.020 1.0000
## Rusp - Soca -0.476781 0.01160 1183 -41.109 <.0001
## Rusp - Soga -0.413808 0.03419 1181 -12.104 <.0001
## Rusp - Sogr -0.397697 0.01814 1186 -21.918 <.0001

```

```

## Rusp - Sora -0.434646 0.02046 1179 -21.247 <.0001
## Rusp - Syla -0.519534 0.04726 1185 -10.993 <.0001
## Rusp - Sypi -0.525192 0.01541 1187 -34.077 <.0001
## Rusp - Syur -0.465789 0.04726 1185 -9.856 <.0001
## Rusp - Taof 0.181773 0.01640 1185 11.081 <.0001
## Rusp - Trsp -0.191794 0.01369 1187 -14.009 <.0001
## Rusp - Vear 0.108841 0.03426 1185 3.177 0.2981
## Sila - Soca -0.477714 0.04609 1180 -10.364 <.0001
## Sila - Soga -0.414741 0.05623 1181 -7.375 <.0001
## Sila - Sogr -0.398629 0.04812 1179 -8.284 <.0001
## Sila - Sora -0.435579 0.04900 1177 -8.889 <.0001
## Sila - Syla -0.520467 0.06508 1182 -7.997 <.0001
## Sila - Sypi -0.526125 0.04727 1181 -11.130 <.0001
## Sila - Syur -0.466722 0.06508 1182 -7.171 <.0001
## Sila - Taof 0.180840 0.04754 1182 3.804 0.0501
## Sila - Trsp -0.192727 0.04655 1178 -4.140 0.0147
## Sila - Vear 0.107908 0.05602 1168 1.926 0.9923
## Soca - Soga 0.062973 0.03267 1182 1.928 0.9922
## Soca - Sogr 0.079085 0.01503 1183 5.263 0.0001
## Soca - Sora 0.042135 0.01778 1178 2.369 0.8910
## Soca - Syla -0.042752 0.04611 1182 -0.927 1.0000
## Soca - Sypi -0.048411 0.01155 1181 -4.190 0.0121
## Soca - Syur 0.010992 0.04611 1182 0.238 1.0000
## Soca - Taof 0.658554 0.01297 1187 50.764 <.0001
## Soca - Trsp 0.284987 0.00915 1168 31.162 <.0001
## Soca - Vear 0.585622 0.03267 1180 17.926 <.0001
## Soga - Sogr 0.016112 0.03546 1175 0.454 1.0000
## Soga - Sora -0.020838 0.03676 1179 -0.567 1.0000
## Soga - Syla -0.105725 0.05629 1183 -1.878 0.9948
## Soga - Sypi -0.111384 0.03426 1184 -3.251 0.2511
## Soga - Syur -0.051981 0.05629 1183 -0.924 1.0000
## Soga - Taof 0.595581 0.03462 1174 17.205 <.0001
## Soga - Trsp 0.222014 0.03345 1181 6.637 <.0001
## Soga - Vear 0.522649 0.04589 1182 11.388 <.0001
## Sogr - Sora -0.036950 0.02254 1169 -1.639 0.9995
## Sogr - Syla -0.121837 0.04824 1183 -2.525 0.8036
## Sogr - Sypi -0.127496 0.01819 1184 -7.009 <.0001
## Sogr - Syur -0.068093 0.04824 1183 -1.411 1.0000
## Sogr - Taof 0.579470 0.01896 1172 30.563 <.0001
## Sogr - Trsp 0.205903 0.01663 1179 12.379 <.0001
## Sogr - Vear 0.506537 0.03553 1181 14.258 <.0001
## Sora - Syla -0.084888 0.04920 1183 -1.725 0.9988
## Sora - Sypi -0.090546 0.02054 1179 -4.409 0.0049
## Sora - Syur -0.031143 0.04920 1183 -0.633 1.0000
## Sora - Taof 0.616419 0.02122 1178 29.054 <.0001
## Sora - Trsp 0.242852 0.01909 1175 12.723 <.0001
## Sora - Vear 0.543487 0.03675 1178 14.789 <.0001
## Syla - Sypi -0.005659 0.04712 1179 -0.120 1.0000
## Syla - Syur 0.053744 0.06455 1160 0.833 1.0000
## Syla - Taof 0.701307 0.04759 1180 14.735 <.0001
## Syla - Trsp 0.327740 0.04680 1182 7.004 <.0001
## Syla - Vear 0.628374 0.05630 1183 11.161 <.0001
## Sypi - Syur 0.059403 0.04712 1179 1.261 1.0000
## Sypi - Taof 0.706966 0.01650 1186 42.855 <.0001

```

```

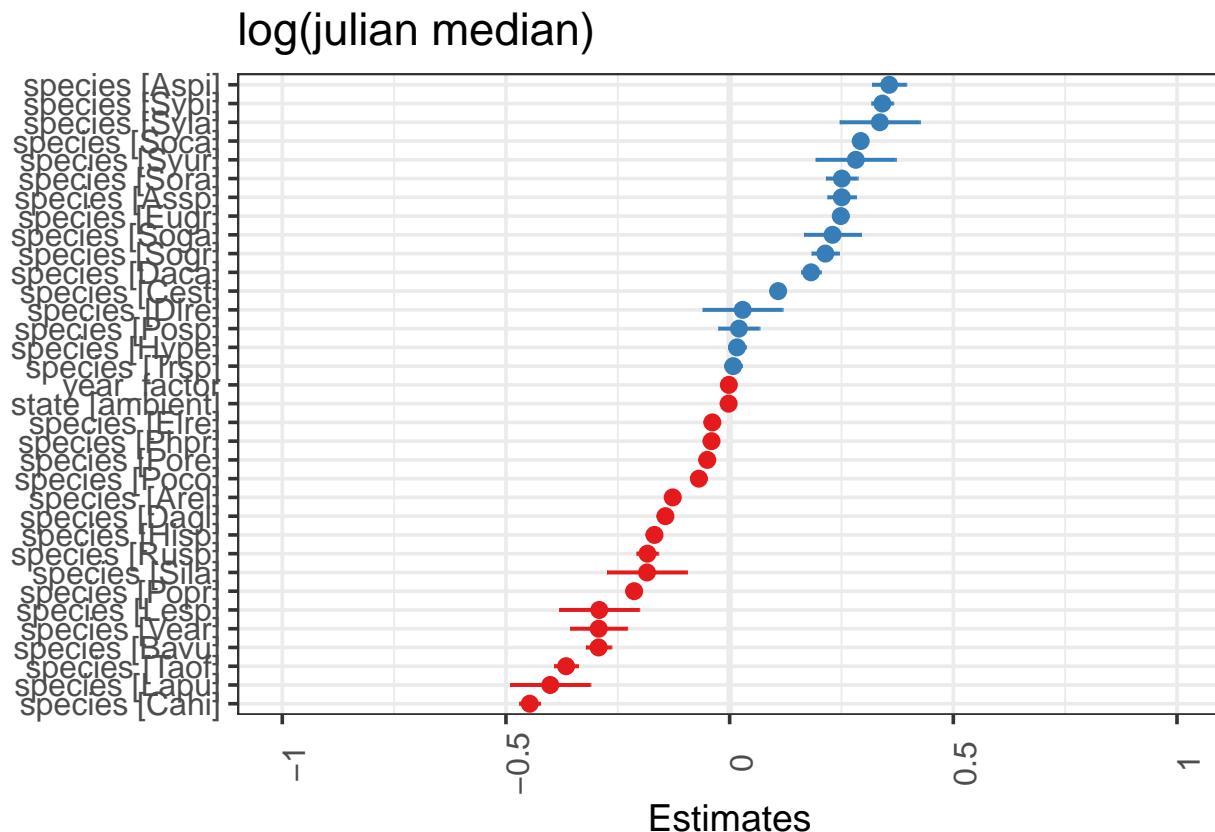
##   Sypi - Trsp  0.333398 0.01377 1178  24.213 <.0001
##   Sypi - Vear  0.634033 0.03425 1182  18.515 <.0001
##   Syur - Taof  0.647563 0.04759 1180  13.606 <.0001
##   Syur - Trsp  0.273996 0.04680 1182   5.855 <.0001
##   Syur - Vear  0.574630 0.05630 1183  10.206 <.0001
##   Taof - Trsp -0.373567 0.01470 1186 -25.412 <.0001
##   Taof - Vear -0.072933 0.03472 1184 -2.101 0.9732
##   Trsp - Vear  0.300634 0.03341 1178   8.998 <.0001
##
## Results are averaged over the levels of: state
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## P value adjustment: tukey method for comparing a family of 33 estimates

```

```
emmeans(mod7ak, list(pairwise ~ insecticide), adjust = "tukey")
```

```
## Error in emmeans(object = object, specs = specs[[i]], name = contr.name, : No variable named insectici
```

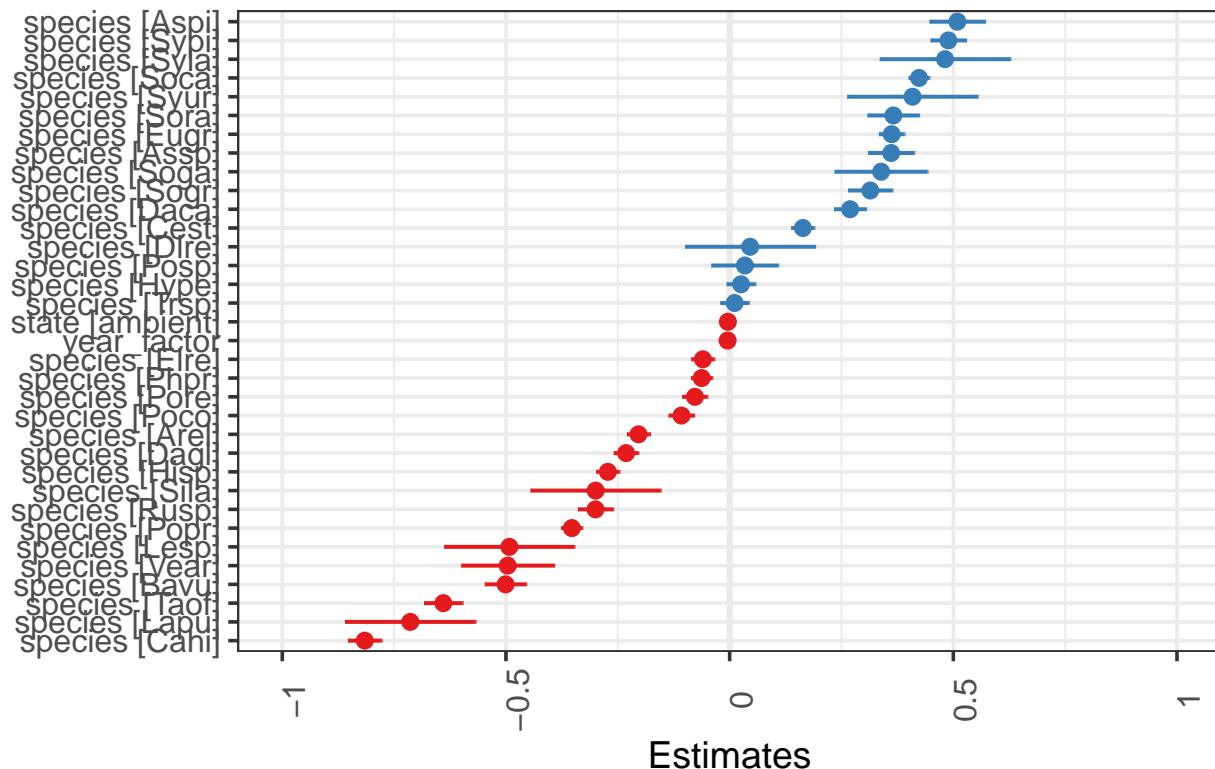
# Take a look at the estimates for each fixed effect. These are the estimates  
# from summary(mod7a). You'll see that species vary a lot - and many of them are  
# different from zero (meaning their half cover date is significantly different  
# from zero).  
plot\_model(mod7ak, sort.est = TRUE)



```
# if you want to standardize the estimates:
plot_model(mod7ak, sort.est = TRUE, type = "std")
```

## Formula contains log- or sqrt-terms. See help("standardize") for how such terms are standardized.

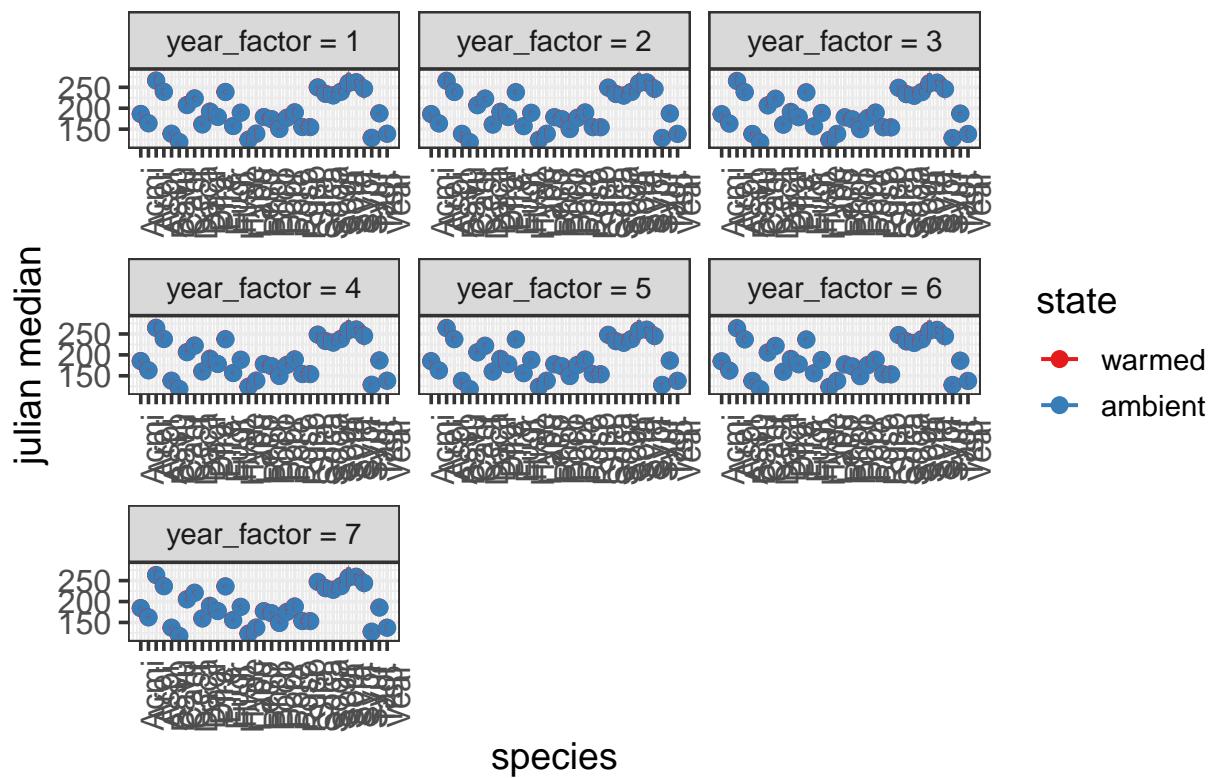
### log(julian median)



```
# these are the fixed predicted values: - note this is a new plot
plot_model(mod7ak, type = "pred", terms = c("species", "state", "year_factor"))
```

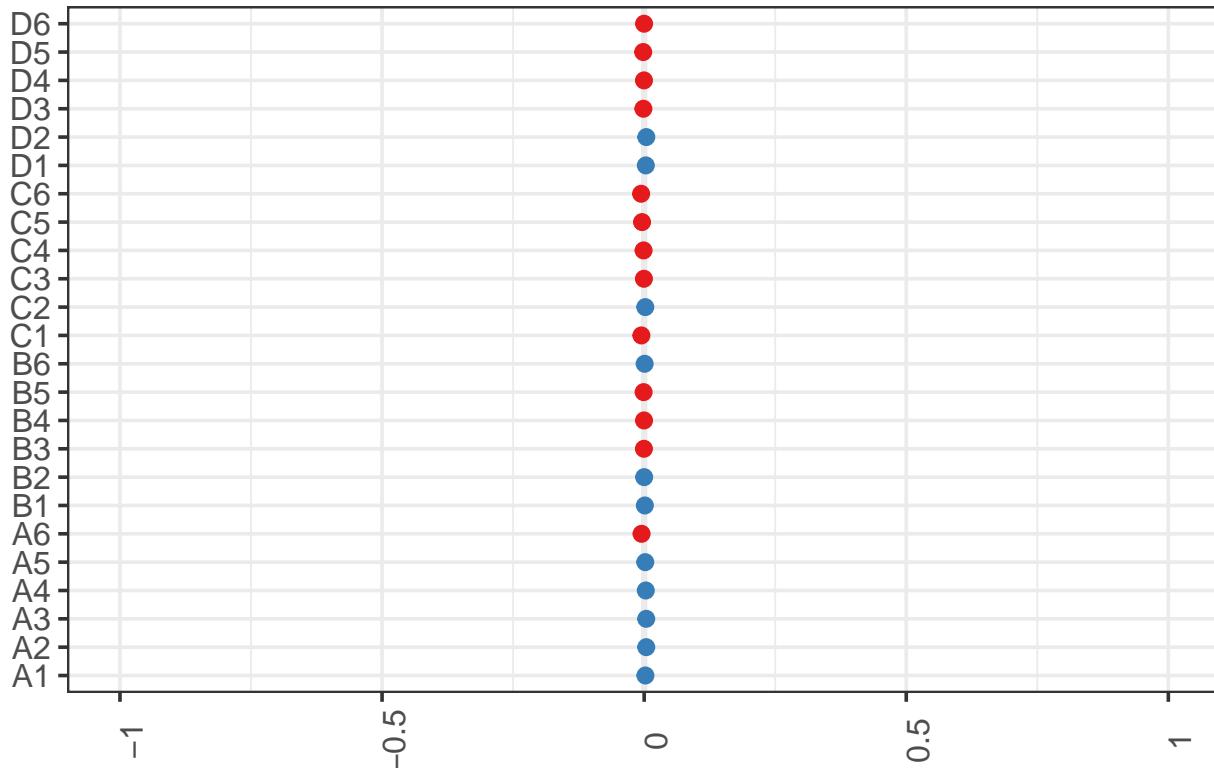
## Model has log-transformed response. Back-transforming predictions to original response scale. Standard

## Predicted values of julian median



```
# these are the random effects estimates
plot_model(mod7ak, type = "re")
```

## Random effects



```
AICctab(mod3k, mod7ak, weights = T)
```

```
##          dAICc df weight
## mod7ak     0.0 37  1
## mod3k   161.4  6 <0.001

# including native vs. exotic
kbs_flwr_spp <- within(kbs_flwr_spp, origin <- relevel(factor(origin), ref = "Native")) # releveling s
mod8k <- lmer(log(julian_median) ~ state * origin + (1 + year_factor | plot), kbs_flwr_spp,
               REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod9k <- lmer(log(julian_median) ~ state + origin + (1 + year_factor | plot), kbs_flwr_spp,
               REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod9ak <- lmer(log(julian_median) ~ state + origin + year_factor + (1 | plot), kbs_flwr_spp,
                 REML = FALSE)
anova(mod8k, mod9k) # model 9k is a better fit to data

## Data: kbs_flwr_spp
## Models:
```

```

## mod9k: log(julian_median) ~ state + origin + (1 + year_factor | plot)
## mod8k: log(julian_median) ~ state * origin + (1 + year_factor | plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod9k     9 -1323.7 -1278.5 670.84   -1341.7
## mod8k    12 -1321.2 -1260.9 672.62   -1345.2 3.5539  3      0.3138

```

```
anova(mod9k, mod9ak) # mod 9ak
```

```

## Data: kbs_flwr_spp
## Models:
## mod9ak: log(julian_median) ~ state + origin + year_factor + (1 | plot)
## mod9k: log(julian_median) ~ state + origin + (1 + year_factor | plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod9ak     8 -1328.4 -1288.2 672.21   -1344.4
## mod9k     9 -1323.7 -1278.5 670.84   -1341.7      0  1      1

```

```
AICctab(mod8k, mod9k, mod9ak, weights = T)
```

```

##      dAICc df weight
## mod9ak  0.0  8  0.895
## mod9k   4.8  9  0.082
## mod8k   7.4 12  0.023

```

```
summary(mod9ak)
```

```

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + origin + year_factor + (1 | plot)
## Data: kbs_flwr_spp
##
##      AIC      BIC      logLik deviance df.resid
## -1328.4 -1288.2     672.2   -1344.4      1117
##
## Scaled residuals:
##      Min      1Q      Median      3Q      Max
## -3.8675 -0.5422   0.1600   0.5615   3.2103
##
## Random effects:
## Groups   Name        Variance Std.Dev.
## plot     (Intercept) 2.407e-05 0.004906
## Residual           1.770e-02 0.133038
## Number of obs: 1125, groups: plot, 24
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)    
## (Intercept) 5.458e+00 1.323e-02 4.039e+02 412.471 < 2e-16 ***
## stateambient 1.188e-02 8.229e-03 2.130e+01  1.444  0.16333  
## origin      -2.132e-01 6.743e-02 1.124e+03 -3.162  0.00161 ** 
## originBoth   -3.283e-01 1.436e-02 1.122e+03 -22.870 < 2e-16 ***
## originExotic -3.162e-01 9.546e-03 1.125e+03 -33.122 < 2e-16 ***
## year_factor -3.896e-03 2.228e-03 1.124e+03 -1.749  0.08061 .
## ---
## 
```

```

## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) sttmbn origin orgnBt orgnEx
## stateambint -0.338
## origin      -0.151  0.006
## originBoth   -0.430 -0.051  0.083
## originExotc -0.566  0.021  0.109  0.489
## year_factor -0.711  0.009  0.106  0.138  0.048

```

```
anova(mod9ak)
```

```

## Type III Analysis of Variance Table with Satterthwaite's method
##           Sum Sq Mean Sq NumDF DenDF F value Pr(>F)
## state       0.0369  0.0369     1    21.3  2.0846 0.16333
## origin      20.4576  6.8192     3 1123.7 385.2847 < 2e-16 ***
## year_factor  0.0541  0.0541     1 1124.2  3.0581 0.08061 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```
emmeans(mod9ak, list(pairwise ~ state * origin), adjust = "tukey")
```

```

## $`emmeans of state, origin`
##   state  origin emmean      SE   df lower.CL upper.CL
##   warmed Native  5.443 0.009539 157.51  5.424  5.461
##   ambient Native  5.454 0.009294 149.46  5.436  5.473
##   warmed        5.229 0.067283 1131.24  5.097  5.361
##   ambient        5.241 0.067315 1131.17  5.109  5.373
##   warmed Both    5.114 0.013017 421.52  5.089  5.140
##   ambient Both   5.126 0.012367 362.86  5.102  5.150
##   warmed Exotic  5.126 0.006850 41.76  5.113  5.140
##   ambient Exotic 5.138 0.006775 39.48  5.125  5.152
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $`pairwise differences of state, origin`
##   1                  estimate      SE   df t.ratio p.value
##   warmed Native - ambient Native -0.011882 0.00868 26.4 -1.369  0.8629
##   warmed Native - warmed          0.213202 0.06770 1129.5  3.149  0.0357
##   warmed Native - ambient         0.201320 0.06832 1126.0  2.947  0.0646
##   warmed Native - warmed Both   0.328342 0.01441 1127.2 22.791 <.0001
##   warmed Native - ambient Both  0.316460 0.01646 303.0 19.223 <.0001
##   warmed Native - warmed Exotic 0.316183 0.00958 1130.3 32.989 <.0001
##   warmed Native - ambient Exotic 0.304301 0.01307 135.0 23.285 <.0001
##   ambient Native - warmed        0.225084 0.06819 1129.1  3.301  0.0222
##   ambient Native - ambient       0.213202 0.06770 1129.5  3.149  0.0357
##   ambient Native - warmed Both  0.340224 0.01717 359.3 19.818 <.0001
##   ambient Native - ambient Both 0.328342 0.01441 1127.2 22.791 <.0001
##   ambient Native - warmed Exotic 0.328065 0.01279 130.5 25.651 <.0001
##   ambient Native - ambient Exotic 0.316183 0.00958 1130.3 32.989 <.0001
##   warmed - ambient            -0.011882 0.00868 26.4 -1.369  0.8629

```

```

##  warmed - warmed Both          0.115140 0.06804 1129.8  1.692  0.6925
##  warmed - ambient Both        0.103258 0.06844 1128.7  1.509  0.8029
##  warmed - warmed Exotic      0.102981 0.06733 1128.8  1.530  0.7914
##  warmed - ambient Exotic     0.091099 0.06784 1129.6  1.343  0.8823
##  ambient - warmed Both        0.127022 0.06875 1125.9  1.848  0.5874
##  ambient - ambient Both       0.115140 0.06804 1129.8  1.692  0.6925
##  ambient - warmed Exotic      0.114863 0.06792 1127.0  1.691  0.6932
##  ambient - ambient Exotic     0.102981 0.06733 1128.8  1.530  0.7914
##  warmed Both - ambient Both   -0.011882 0.00868 26.4   -1.369  0.8629
##  warmed Both - warmed Exotic  -0.012158 0.01282 1130.1  -0.948  0.9811
##  warmed Both - ambient Exotic -0.024040 0.01597 278.2  -1.505  0.8043
##  ambient Both - warmed Exotic -0.000277 0.01497 222.8  -0.018  1.0000
##  ambient Both - ambient Exotic -0.012158 0.01282 1130.1  -0.948  0.9811
##  warmed Exotic - ambient Exotic -0.011882 0.00868 26.4   -1.369  0.8629
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## P value adjustment: tukey method for comparing a family of 8 estimates

# including growth form - first with interaction term
kbs_flwr_spp <- within(kbs_flwr_spp, growth_habit <- relevel(factor(growth_habit),
  ref = "Forb")) # releveling so forb is the reference
mod10k <- lmer(log(julian_median) ~ state * growth_habit + (1 + year_factor | plot),
  kbs_flwr_spp, REML = FALSE)

## fixed-effect model matrix is rank deficient so dropping 1 column / coefficient

## boundary (singular) fit: see ?isSingular

mod11k <- lmer(log(julian_median) ~ state + growth_habit + (1 + year_factor | plot),
  kbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod11ak <- lmer(log(julian_median) ~ state + growth_habit + year_factor + (1 | plot),
  kbs_flwr_spp, REML = FALSE)
anova(mod10k, mod11k) # model 11k is a better fit to data

## Data: kbs_flwr_spp
## Models:
## mod11k: log(julian_median) ~ state + growth_habit + (1 + year_factor |
## mod11k:      plot)
## mod10k: log(julian_median) ~ state * growth_habit + (1 + year_factor |
## mod10k:      plot)
##      npar    AIC    BIC logLik deviance Chisq Df Pr(>Chisq)
## mod11k    9 -1004.6 -959.41 511.32   -1022.6
## mod10k   11 -1003.9 -948.62 512.95   -1025.9 3.2552  2      0.1964

anova(mod11k, mod11ak) # model 11ak is a better fit to data

```

```

## Data: kbs_flwr_spp
## Models:
## mod11ak: log(julian_median) ~ state + growth_habit + year_factor + (1 |
## mod11ak:      plot)
## mod11k: log(julian_median) ~ state + growth_habit + (1 + year_factor | 
## mod11k:      plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod11ak     8 -1017.8 -977.57 516.89   -1033.8
## mod11k      9 -1004.6 -959.41 511.32   -1022.6      0   1          1

AICctab(mod10k, mod11k, mod11ak, weights = T)

##      dAICc df weight
## mod11ak  0.0  8  0.9977
## mod11k  13.2  9  0.0014
## mod10k  14.0 11 <0.001

summary(mod11ak)

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + growth_habit + year_factor + (1 |
##      plot)
## Data: kbs_flwr_spp
##
##      AIC      BIC  logLik deviance df.resid
## -1017.8 -977.6    516.9   -1033.8      1117
##
## Scaled residuals:
##      Min      1Q Median      3Q      Max
## -4.5492 -0.4798  0.0751  0.6355  2.0532
##
## Random effects:
## Groups   Name        Variance Std.Dev.
## plot     (Intercept) 0.0001551 0.01246
## Residual           0.0232239 0.15239
## Number of obs: 1125, groups: plot, 24
##
## Fixed effects:
##                   Estimate Std. Error      df t value Pr(>|t|)    
## (Intercept)      5.297e+00  1.334e-02  2.087e+02 397.218 < 2e-16 ***
## stateambient    2.757e-03  1.051e-02  2.272e+01   0.262  0.795385  
## growth_habit    -2.276e-01  1.547e-02  1.121e+03 -14.717 < 2e-16 ***
## growth_habitForb/herb -5.426e-01  1.532e-01  1.123e+03  -3.541 0.000415 ***
## growth_habitGraminoid -2.262e-01  9.804e-03  1.125e+03 -23.072 < 2e-16 ***
## year_factor      8.679e-03  2.575e-03  1.124e+03   3.371 0.000774 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##            (Intr) sttmbn grwth_ grw_F/ grwt_G
## stateambint -0.426
## growth_habt -0.263 -0.034

```

```

## grwth_hbtF/  0.004  0.029  0.014
## grwth_hbtGr -0.270  0.063  0.292  0.043
## year_factor -0.720 -0.003  0.073 -0.055 -0.158

anova(mod11ak)

## Type III Analysis of Variance Table with Satterthwaite's method
##           Sum Sq Mean Sq NumDF DendDF F value Pr(>F)
## state      0.0016  0.0016     1    22.72  0.0688 0.7953853
## growth_habit 14.1323  4.7108     3 1122.82 202.8404 < 2.2e-16 ***
## year_factor  0.2639  0.2639     1 1123.59 11.3638 0.0007744 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

emmeans(mod11ak, list(pairwise ~ factor(year_factor) * growth_habit), adjust = "tukey")

## $`emmeans of year_factor, growth_habit`
##   year_factor growth_habit emmean       SE   df lower.CL upper.CL
##   3.96 Forb        5.333 0.007540  94.05   5.318   5.348
##   3.96          5.105 0.014353 667.19   5.077   5.134
##   3.96 Forb/herb  4.790 0.153604 1128.46   4.489   5.092
##   3.96 Graminoid  5.107 0.007566  96.34   5.092   5.122
##
## Results are averaged over the levels of: state
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $`pairwise differences of year_factor, growth_habit`
##   1                   estimate       SE   df t.ratio p.value
##   3.96 Forb - 3.96      0.22761 0.01552 1126 14.668 <.0001
##   3.96 Forb - (3.96 Forb/herb) 0.54257 0.15378 1128  3.528  0.0025
##   3.96 Forb - 3.96 Graminoid  0.22620 0.00985 1131 22.974 <.0001
##   3.96 - (3.96 Forb/herb)    0.31496 0.15434 1128  2.041  0.1738
##   3.96 - 3.96 Graminoid     -0.00142 0.01577 1130 -0.090  0.9997
##   (3.96 Forb/herb) - 3.96 Graminoid -0.31638 0.15366 1127 -2.059  0.1674
##
## Results are averaged over the levels of: state
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## P value adjustment: tukey method for comparing a family of 4 estimates

# 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
mod12k <- lmer(log(julian_median) ~ state * factor(year_factor) + (1 + factor(year_factor) |
  species), kbs_flwr_spp)

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

```

# 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
mod13k <- lmer(log(julian_median) ~ state * factor(year_factor) + (1 | species),
  kbs_flwr_spp)

# All the models ran:
mod1k <- lmer(log(julian_median) ~ state * year_factor + insecticide * year_factor +
  (1 | species) + (1 | plot), kbs_flwr_spp, REML = FALSE)
mod2k <- lmer(log(julian_median) ~ state * year_factor + insecticide * year_factor +
  (1 | species), kbs_flwr_spp, REML = FALSE)
mod3k <- lmer(log(julian_median) ~ state * year_factor + (1 | species), kbs_flwr_spp,
  REML = FALSE)
mod4k <- lmer(log(julian_median) ~ state * year_factor + insecticide + (1 | species) +
  (1 | plot), kbs_flwr_spp, REML = FALSE)
mod5k <- lmer(log(julian_median) ~ state + year_factor + (1 | species), kbs_flwr_spp,
  REML = FALSE)
mod6k <- lmer(log(julian_median) ~ state + year_factor + (1 | species) + (1 + year |
  plot), kbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod7k <- lmer(log(julian_median) ~ state + species + (1 + factor(year_factor) | plot),
  kbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod7ak <- lmer(log(julian_median) ~ state + species + factor(year_factor) + (1 |
  plot), kbs_flwr_spp, REML = FALSE)
mod7bk <- lmer(log(julian_median) ~ state * factor(year_factor) + species + (1 |
  plot), kbs_flwr_spp, REML = FALSE)
mod7ck <- lmer(log(julian_median) ~ state + species + factor(year_factor) + insecticide +
  (1 | plot), kbs_flwr_spp, REML = FALSE)
mod8k <- lmer(log(julian_median) ~ state * origin + (1 + factor(year_factor) | plot),
  kbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

## Warning: Model failed to converge with 1 negative eigenvalue: -4.7e+00

mod9k <- lmer(log(julian_median) ~ state + origin + (1 + factor(year_factor) | plot),
  kbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod9ak <- lmer(log(julian_median) ~ state + origin + factor(year_factor) + (1 | plot),
  kbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

```

```

mod10k <- lmer(log(julian_median) ~ state * growth_habit + (1 + factor(year_factor) |
  plot), kbs_flwr_spp, REML = FALSE)

## fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
## boundary (singular) fit: see ?isSingular

mod11k <- lmer(log(julian_median) ~ state + growth_habit + (1 + factor(year_factor) |
  plot), kbs_flwr_spp, REML = FALSE)

## boundary (singular) fit: see ?isSingular

## Warning: Model failed to converge with 1 negative eigenvalue: -2.8e+01

mod11ak <- lmer(log(julian_median) ~ state + growth_habit + factor(year_factor) +
  (1 | plot), kbs_flwr_spp, REML = FALSE)
mod12k <- lmer(log(julian_median) ~ state * factor(year_factor) + (1 + factor(year_factor) |
  species), kbs_flwr_spp)

## boundary (singular) fit: see ?isSingular

mod13k <- lmer(log(julian_median) ~ state * factor(year_factor) + (1 | species),
  kbs_flwr_spp)
AICctab(mod1k, mod2k, mod3k, mod5k, mod6k, mod7k, mod7ak, mod7bk, mod7ck, mod8k,
  mod9k, mod9ak, mod10k, mod11k, mod11ak, mod13k, weights = T) #mod7a is the best fitting model (too

## Error in ICtab(..., mnames = mnames, type = "AICc"): nobs different: must have identical data for all

summary(mod7ak)

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
##   method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + species + factor(year_factor) +
##   (1 | plot)
## Data: kbs_flwr_spp
##
##      AIC      BIC      logLik deviance df.resid
## -3917.8 -3705.8    2000.9   -4001.8     1109
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -6.6968 -0.5912 -0.0204  0.5359  4.4994
##
## Random effects:
##   Groups   Name        Variance Std.Dev.
##   plot     (Intercept) 1.535e-05 0.003918
##   Residual           1.797e-03 0.042388
## Number of obs: 1151, groups:  plot, 24
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)

```

```

## (Intercept)      5.274e+00  7.634e-03  9.249e+02 690.861 < 2e-16 ***
## stateambient   -1.121e-03  3.069e-03  2.272e+01 -0.365    0.718
## speciesArel    -1.292e-01  7.661e-03  1.149e+03 -16.865 < 2e-16 ***
## speciesAspi     3.706e-01  1.841e-02  1.142e+03 20.135 < 2e-16 ***
## speciesAssp     2.608e-01  1.539e-02  1.137e+03 16.952 < 2e-16 ***
## speciesBavu    -2.865e-01  1.367e-02  1.146e+03 -20.957 < 2e-16 ***
## speciesCahi    -4.363e-01  1.117e-02  1.145e+03 -39.050 < 2e-16 ***
## speciesCest     9.901e-02  7.651e-03  1.143e+03 12.941 < 2e-16 ***
## speciesDaca     1.809e-01  1.055e-02  1.146e+03 17.146 < 2e-16 ***
## speciesDagl    -1.403e-01  8.164e-03  1.150e+03 -17.188 < 2e-16 ***
## speciesDlre     4.734e-02  4.299e-02  1.146e+03 1.101    0.271
## speciesElre    -3.841e-02  7.582e-03  1.150e+03 -5.066   4.73e-07 ***
## speciesEugr     2.391e-01  8.512e-03  1.150e+03 28.089 < 2e-16 ***
## speciesHisp    -1.642e-01  7.683e-03  1.147e+03 -21.370 < 2e-16 ***
## speciesHype     7.909e-03  9.513e-03  1.147e+03 0.831    0.406
## speciesLapu   -3.979e-01  4.314e-02  1.148e+03 -9.225 < 2e-16 ***
## speciesLesp    -2.758e-01  4.303e-02  1.148e+03 -6.410   2.12e-10 ***
## speciesPhpr    -4.332e-02  7.083e-03  1.146e+03 -6.116   1.32e-09 ***
## speciesPoco    -6.833e-02  8.483e-03  1.150e+03 -8.055   1.97e-15 ***
## speciesPopr    -2.109e-01  6.946e-03  1.146e+03 -30.364 < 2e-16 ***
## speciesPore    -4.862e-02  8.266e-03  1.148e+03 -5.883   5.29e-09 ***
## speciesPosp    -2.603e-02  2.252e-02  1.149e+03 -1.156    0.248
## speciesRusp    -1.832e-01  1.163e-02  1.143e+03 -15.758 < 2e-16 ***
## speciesSila    -1.818e-01  4.297e-02  1.146e+03 -4.230   2.52e-05 ***
## speciesSoca     2.951e-01  6.848e-03  1.145e+03 43.091 < 2e-16 ***
## speciesSoga     2.430e-01  3.075e-02  1.148e+03 7.903   6.35e-15 ***
## speciesSogr     2.288e-01  1.476e-02  1.149e+03 15.508 < 2e-16 ***
## speciesSora     2.678e-01  1.723e-02  1.142e+03 15.545 < 2e-16 ***
## speciesSyla     3.376e-01  4.312e-02  1.148e+03 7.829   1.11e-14 ***
## speciesSypi    3.383e-01  1.179e-02  1.150e+03 28.708 < 2e-16 ***
## speciesSyur     2.838e-01  4.312e-02  1.148e+03 6.583   7.01e-11 ***
## speciesTaof    -3.556e-01  1.276e-02  1.151e+03 -27.867 < 2e-16 ***
## speciesTrsp    -1.124e-02  9.539e-03  1.138e+03 -1.179    0.239
## speciesVear    -2.763e-01  3.063e-02  1.146e+03 -9.018 < 2e-16 ***
## factor(year_factor)2 -4.998e-02  5.932e-03  1.143e+03 -8.424 < 2e-16 ***
## factor(year_factor)3 -6.703e-02  6.294e-03  1.143e+03 -10.648 < 2e-16 ***
## factor(year_factor)4 -6.659e-02  6.298e-03  1.146e+03 -10.573 < 2e-16 ***
## factor(year_factor)5 -4.218e-02  6.438e-03  1.145e+03 -6.551   8.59e-11 ***
## factor(year_factor)6 -4.371e-02  6.555e-03  1.146e+03 -6.669   3.99e-11 ***
## factor(year_factor)7 -5.933e-02  6.823e-03  1.147e+03 -8.696 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## Correlation matrix not shown by default, as p = 40 > 12.
## Use print(x, correlation=TRUE) or
##      vcov(x)      if you need it

```

```
summ(mod7ak)
```

```
anova(mod7ak)
```

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

Observations	1151
Dependent variable	log(julian_median)
Type	Mixed effects linear regression
AIC	-3917.79
BIC	-3705.76
Pseudo-R <sup>2</sup> (fixed effects)	0.95
Pseudo-R <sup>2</sup> (total)	0.95

```
##          Sum Sq Mean Sq NumDF DenDF F value Pr(>F)
## state        0.000 0.00024     1    22.72  0.1334 0.7183
## species      38.670 1.20845    32 1143.67 672.5861 <2e-16 ***
## factor(year_factor) 0.275 0.04591     6 1140.60 25.5533 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

## KBS Plot-level Mixed Effects Models:

```
mod1pk <- lmer(log(julian_median) ~ state + (1 | plot), kbs_flwr_plot, REML = FALSE)
mod2pk <- lmer(log(julian_median) ~ insecticide + (1 | plot), kbs_flwr_plot, REML = FALSE)
mod3pk <- lmer(log(julian_median) ~ insecticide + state + (1 | plot), kbs_flwr_plot,
REML = FALSE)
mod4pk <- lmer(log(julian_median) ~ insecticide * state + (1 | plot), kbs_flwr_plot,
REML = FALSE)
mod5pk <- lmer(log(julian_median) ~ state + year_factor + (1 | plot), kbs_flwr_plot,
REML = FALSE)
mod6pk <- lmer(log(julian_median) ~ state + year_factor + insecticide + (1 | plot),
kbs_flwr_plot, REML = FALSE)
mod7pk <- lmer(log(julian_median) ~ state * year_factor + (1 | plot), kbs_flwr_plot,
REML = FALSE)
mod8pk <- lmer(log(julian_median) ~ state * year_factor + insecticide + (1 | plot),
kbs_flwr_plot, REML = FALSE)
mod9pk <- lmer(log(julian_median) ~ state * insecticide + year_factor + (1 | plot),
kbs_flwr_plot, REML = FALSE)
mod10pk <- lmer(log(julian_median) ~ state + insecticide * year_factor + (1 | plot),
kbs_flwr_plot, REML = FALSE)
mod11pk <- lmer(log(julian_median) ~ state * year_factor * insecticide + (1 | plot),
kbs_flwr_plot, REML = FALSE)
AICctab(mod1pk, mod2pk, mod3pk, mod4pk, mod5pk, mod6pk, mod7pk, mod8pk, mod9pk, mod10pk,
mod11pk, weights = T)
```

```
## Error in ICtab(..., mnames = mnames, type = "AICc"): nobs different: must have identical data for all
```

```
anova(mod5pk, mod7pk) # go with more complex model mod7pk
```

```
## Data: kbs_flwr_plot
## Models:
## mod5pk: log(julian_median) ~ state + year_factor + (1 | plot)
```

Fixed Effects					
	Est.	S.E.	t val.	d.f.	p
(Intercept)	5.27	0.01	690.86	924.91	0.00
stateambient	-0.00	0.00	-0.37	22.72	0.72
speciesArel	-0.13	0.01	-16.87	1148.93	0.00
speciesAspi	0.37	0.02	20.14	1141.55	0.00
speciesAssp	0.26	0.02	16.95	1136.51	0.00
speciesBavu	-0.29	0.01	-20.96	1145.51	0.00
speciesCahi	-0.44	0.01	-39.05	1144.50	0.00
speciesCest	0.10	0.01	12.94	1143.30	0.00
speciesDaca	0.18	0.01	17.15	1146.37	0.00
speciesDagl	-0.14	0.01	-17.19	1150.17	0.00
speciesDlre	0.05	0.04	1.10	1146.47	0.27
speciesElre	-0.04	0.01	-5.07	1149.98	0.00
speciesEugr	0.24	0.01	28.09	1149.99	0.00
speciesHisp	-0.16	0.01	-21.37	1146.54	0.00
speciesHype	0.01	0.01	0.83	1147.07	0.41
speciesLapu	-0.40	0.04	-9.22	1148.39	0.00
speciesLesp	-0.28	0.04	-6.41	1147.86	0.00
speciesPhpr	-0.04	0.01	-6.12	1145.55	0.00
speciesPoco	-0.07	0.01	-8.06	1150.05	0.00
speciesPopr	-0.21	0.01	-30.36	1146.16	0.00
speciesPore	-0.05	0.01	-5.88	1148.24	0.00
speciesPosp	-0.03	0.02	-1.16	1148.77	0.25
speciesRusp	-0.18	0.01	-15.76	1143.23	0.00
speciesSila	-0.18	0.04	-4.23	1146.48	0.00
speciesSoca	0.30	0.01	43.09	1144.93	0.00
speciesSoga	0.24	0.03	7.90	1147.68	0.00
speciesSogr	0.23	0.01	15.51	1148.66	0.00
speciesSora	0.27	0.02	15.55	1142.45	0.00
speciesSyla	0.34	0.04	7.83	1148.24	0.00
speciesSypi	0.34	0.01	28.71	1149.78	0.00
speciesSyur	0.28	0.04	6.58	1148.24	0.00
speciesTaof	-0.36	0.01	-27.87	1150.66	0.00
speciesTrsp	-0.01	0.01	-1.18	1137.98	0.24
speciesVear	-0.28	0.03	-9.02	1146.20	0.00
factor(year_factor)2	-0.05	0.01	-8.42	1143.14	0.00
factor(year_factor)3	-0.07	0.01	-10.65	1142.85	0.00
factor(year_factor)4	-0.07	0.01	-10.57	1145.89	0.00
factor(year_factor)5	-0.04	0.01	-6.55	1144.73	0.00
factor(year_factor)6	-0.04	0.01	-6.67	1145.93	0.00
factor(year_factor)7	-0.06	0.01	-8.70	1147.24	0.00

p values calculated using Satterthwaite d.f.

```

## mod7pk: log(julian_median) ~ state * year_factor + (1 | plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod5pk     5 -475.50 -459.94 242.75   -485.50
## mod7pk     6 -478.74 -460.07 245.37   -490.74 5.2475  1    0.02198 *
## ---

```

Random Effects		
Group	Parameter	Std. Dev.
plot	(Intercept)	0.00
Residual		0.04

Grouping Variables		
Group	# groups	ICC
plot	24	0.01

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

```
AICctab(mod5pk, mod7pk, weights = T)
```

```
##      dAICc df weight
## mod7pk 0.0   6  0.82
## mod5pk 3.1   5  0.18
```

```
summ(mod5pk)
```

Observations	166
Dependent variable	log(julian_median)
Type	Mixed effects linear regression

AIC	-475.50
BIC	-459.94
Pseudo-R <sup>2</sup> (fixed effects)	0.04
Pseudo-R <sup>2</sup> (total)	0.07

Fixed Effects					
	Est.	S.E.	t val.	d.f.	p
(Intercept)	5.24	0.01	481.58	114.94	0.00
stateambient	0.02	0.01	2.03	24.06	0.05
year_factor	-0.00	0.00	-1.29	143.51	0.20

p values calculated using Satterthwaite d.f.

Random Effects		
Group	Parameter	Std. Dev.
plot	(Intercept)	0.01
Residual		0.06

```
summary(mod5pk)
```

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

Grouping Variables		
Group	# groups	ICC
plot	24	0.03

```

##   method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + year_factor + (1 | plot)
##   Data: kbs_flwr_plot
##
##      AIC      BIC  logLik deviance df.resid
## -475.5   -459.9    242.7    -485.5     161
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -2.2343 -0.6811 -0.0614  0.6005  3.2920
##
## Random effects:
##   Groups   Name        Variance Std.Dev.
##   plot      (Intercept) 9.393e-05 0.009692
##   Residual           3.057e-03 0.055288
## Number of obs: 166, groups: plot, 24
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)    
## (Intercept) 5.243228  0.010888 114.941988 481.581 <2e-16 ***
## stateambient 0.019174  0.009454  24.058249  2.028  0.0538 .
## year_factor -0.002775  0.002155 143.505555 -1.288  0.1999
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr) sttmbn
## stateambint -0.441
## year_factor -0.792  0.015

emmeans(mod5pk, list(pairwise ~ state + year_factor), adjust = "tukey")

## $`emmeans of state, year_factor`
##   state   year_factor emmean       SE      df lower.CL upper.CL
##   warmed         3.97  5.232 0.006948 25.75    5.218    5.246
##   ambient        3.97  5.251 0.007019 26.48    5.237    5.266
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $`pairwise differences of state, year_factor`
##   1                           estimate       SE      df
##   warmed 3.96987951807229 - ambient 3.96987951807229 -0.0192 0.00988 26.1
##   t.ratio p.value
##   -1.941  0.0631
##
## Degrees-of-freedom method: kenward-roger

```

```

## Results are given on the log (not the response) scale.

# including native vs. exotic
kbs_flwr_plot_origin <- within(kbs_flwr_plot_origin, origin <- relevel(factor(origin),
  ref = "Native")) # releveling so native is the reference
mod12pk <- lmer(log(julian_median) ~ state * origin + (1 + year_factor | plot), kbs_flwr_plot_origin,
  REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod13pk <- lmer(log(julian_median) ~ state + origin + (1 + year_factor | plot), kbs_flwr_plot_origin,
  REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod14pk <- lmer(log(julian_median) ~ state + origin + year_factor + (1 | plot), kbs_flwr_plot_origin,
  REML = FALSE)

## boundary (singular) fit: see ?isSingular

anova(mod12pk, mod13pk) # go with model 12pk

## Data: kbs_flwr_plot_origin
## Models:
## mod13pk: log(julian_median) ~ state + origin + (1 + year_factor | plot)
## mod12pk: log(julian_median) ~ state * origin + (1 + year_factor | plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod13pk     9 -834.73 -798.41 426.37   -852.73
## mod12pk    12 -833.06 -784.64 428.53   -857.06 4.3311   3      0.2279

anova(mod12pk, mod14pk) # mod 14pk

## Data: kbs_flwr_plot_origin
## Models:
## mod14pk: log(julian_median) ~ state + origin + year_factor + (1 | plot)
## mod12pk: log(julian_median) ~ state * origin + (1 + year_factor | plot)
##      npar      AIC      BIC logLik deviance Chisq Df Pr(>Chisq)
## mod14pk     8 -849.87 -817.59 432.94   -865.87
## mod12pk    12 -833.06 -784.64 428.53   -857.06      0  4          1

AICctab(mod12pk, mod13pk, mod14pk, weights = T)

##      dAICc df weight
## mod14pk  0.0  8    1
## mod13pk 15.2  9  <0.001
## mod12pk 17.2 12  <0.001

```

```

summary(mod14pk)

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
##   method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + origin + year_factor + (1 | plot)
##   Data: kbs_flwr_plot_origin
##
##      AIC      BIC  logLik deviance df.resid
## -849.9   -817.6    432.9   -865.9     410
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -3.4493 -0.7075 -0.0040  0.6229  2.8178
##
## Random effects:
##   Groups   Name        Variance Std.Dev.
##   plot      (Intercept) 0.000000 0.00000
##   Residual           0.007377 0.08589
## Number of obs: 418, groups: plot, 24
##
## Fixed effects:
##             Estimate Std. Error      df t value Pr(>|t|)    
## (Intercept) 5.495453  0.012365 418.00000 444.421 < 2e-16 ***
## statewarmed -0.004467  0.008442 418.00000 -0.529 0.596989  
## origin      -0.238104  0.044070 418.00000 -5.403 1.1e-07 ***
## originBoth   -0.340426  0.011357 418.00000 -29.975 < 2e-16 ***
## originExotic -0.316462  0.009636 418.00000 -32.843 < 2e-16 ***
## year_factor -0.008092  0.002207 418.00000 -3.666 0.000278 *** 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##            (Intr) sttwrm origin orgnBt orgnEx
## statewarmed -0.353
## origin      -0.209  0.003
## originBoth   -0.475  0.084  0.118
## originExotic -0.442  0.009  0.121  0.445
## year_factor -0.751  0.003  0.160  0.140  0.049
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular

```

```

summ(mod14pk)

```

Observations	418
Dependent variable	log(julian_median)
Type	Mixed effects linear regression

```

anova(mod14pk)

```

```

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

```

AIC	-849.87
BIC	-817.59
Pseudo-R <sup>2</sup> (fixed effects)	0.77
Pseudo-R <sup>2</sup> (total)	0.77

Fixed Effects					
	Est.	S.E.	t val.	d.f.	p
(Intercept)	5.50	0.01	444.42	418.00	0.00
statewarmed	-0.00	0.01	-0.53	418.00	0.60
origin	-0.24	0.04	-5.40	418.00	0.00
originBoth	-0.34	0.01	-29.97	418.00	0.00
originExotic	-0.32	0.01	-32.84	418.00	0.00
year_factor	-0.01	0.00	-3.67	418.00	0.00

p values calculated using Satterthwaite d.f.

Random Effects		
Group	Parameter	Std. Dev.
plot	(Intercept)	0.00
Residual		0.09

Grouping Variables		
Group	# groups	ICC
plot	24	0.00

```
##          Sum Sq Mean Sq NumDF DenDF F value    Pr(>F)
## state      0.0021  0.0021     1    418   0.280 0.5969889
## origin     10.1259  3.3753     3    418 457.527 < 2.2e-16 ***
## year_factor  0.0992  0.0992     1    418  13.441 0.0002779 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(mod14pk, list(pairwise ~ state + origin + year_factor), adjust = "tukey")
```

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

```
## $`emmeans of state, origin, year_factor`
## state origin year_factor emmean       SE      df lower.CL upper.CL
## ambient Native      3.923  5.464 0.008253  76.41     5.447  5.480
## warmed  Native      3.923  5.459 0.008134  73.12     5.443  5.475
## ambient      Native      3.923  5.226 0.044143 422.50     5.139  5.312
## warmed      Native      3.923  5.221 0.044140 422.89     5.134  5.308
## ambient Both       3.923  5.123 0.009582 114.33     5.104  5.142
## warmed Both       3.923  5.119 0.010322 137.30     5.098  5.139
## ambient Exotic     3.923  5.147 0.007984  67.82     5.131  5.163
## warmed Exotic     3.923  5.143 0.007954  67.73     5.127  5.159
##
## Degrees-of-freedom method: kenward-roger
```

```

## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
##
## $`pairwise differences of state, origin, year_factor`
## 1
##   ambient Native 3.92344497607656 - warmed Native 3.92344497607656   estimate
##   ambient Native 3.92344497607656 - ambient    3.92344497607656   0.00447
##   ambient Native 3.92344497607656 - warmed    3.92344497607656   0.23810
##   ambient Native 3.92344497607656 - warmed Both 3.92344497607656   0.24257
##   ambient Native 3.92344497607656 - ambient Both 3.92344497607656   0.34043
##   ambient Native 3.92344497607656 - warmed Both 3.92344497607656   0.34489
##   ambient Native 3.92344497607656 - ambient Exotic 3.92344497607656   0.31646
##   ambient Native 3.92344497607656 - warmed Exotic 3.92344497607656   0.32093
##   warmed Native 3.92344497607656 - ambient    3.92344497607656   0.23364
##   warmed Native 3.92344497607656 - warmed    3.92344497607656   0.23810
##   warmed Native 3.92344497607656 - ambient Both 3.92344497607656   0.33596
##   warmed Native 3.92344497607656 - warmed Both 3.92344497607656   0.34043
##   warmed Native 3.92344497607656 - ambient Exotic 3.92344497607656   0.31200
##   warmed Native 3.92344497607656 - warmed Exotic 3.92344497607656   0.31646
##   ambient    3.92344497607656 - warmed    3.92344497607656   0.00447
##   ambient    3.92344497607656 - ambient Both 3.92344497607656   0.10232
##   ambient    3.92344497607656 - warmed Both 3.92344497607656   0.10679
##   ambient    3.92344497607656 - ambient Exotic 3.92344497607656   0.07836
##   ambient    3.92344497607656 - warmed Exotic 3.92344497607656   0.08282
##   warmed    3.92344497607656 - ambient Both 3.92344497607656   0.09786
##   warmed    3.92344497607656 - warmed Both 3.92344497607656   0.10232
##   warmed    3.92344497607656 - ambient Exotic 3.92344497607656   0.07389
##   warmed    3.92344497607656 - warmed Exotic 3.92344497607656   0.07836
##   ambient Both 3.92344497607656 - warmed Both 3.92344497607656   0.00447
##   ambient Both 3.92344497607656 - ambient Exotic 3.92344497607656   -0.02396
##   ambient Both 3.92344497607656 - warmed Exotic 3.92344497607656   -0.01950
##   warmed Both 3.92344497607656 - ambient Exotic 3.92344497607656   -0.02843
##   warmed Both 3.92344497607656 - warmed Exotic 3.92344497607656   -0.02396
##   ambient Exotic 3.92344497607656 - warmed Exotic 3.92344497607656   0.00447
##   SE      df t.ratio p.value
##   0.00851 22.6  0.525  0.9994
##   0.04458 423.8  5.341 <.0001
##   0.04541 418.5  5.342 <.0001
##   0.01147 419.0 29.682 <.0001
##   0.01485 152.2 23.218 <.0001
##   0.00971 402.6 32.603 <.0001
##   0.01297 110.7 24.752 <.0001
##   0.04537 416.9  5.150 <.0001
##   0.04458 423.8  5.341 <.0001
##   0.01369 125.7 24.548 <.0001
##   0.01147 419.0 29.682 <.0001
##   0.01285 107.1 24.274 <.0001
##   0.00971 402.6 32.603 <.0001
##   0.00851 22.6  0.525  0.9994
##   0.04471 424.0  2.288  0.3024
##   0.04568 416.0  2.338  0.2757
##   0.04447 423.7  1.762  0.6461
##   0.04527 417.3  1.829  0.6003
##   0.04535 417.7  2.158  0.3798
##   0.04471 424.0  2.288  0.3024

```

```

##  0.04528 418.7  1.632  0.7307
##  0.04447 423.7  1.762  0.6461
##  0.00851  22.6  0.525  0.9994
##  0.01126 419.2 -2.128  0.3987
##  0.01357 122.6 -1.437  0.8382
##  0.01465 145.8 -1.941  0.5251
##  0.01126 419.2 -2.128  0.3987
##  0.00851  22.6  0.525  0.9994
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## P value adjustment: tukey method for comparing a family of 8 estimates

# including growth form - first with interaction term
kbs_flwr_plot_growthhabit <- within(kbs_flwr_plot_growthhabit, growth_habit <- relevel(factor(growth_habit,
  ref = "Forb")) # releveling so forb is the reference
mod15pk <- lmer(log(julian_median) ~ state * growth_habit + (1 + year_factor | plot),
  umbs_flwr_plot_growthhabit, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod16pk <- lmer(log(julian_median) ~ state + growth_habit + (1 + year_factor | plot),
  umbs_flwr_plot_growthhabit, REML = FALSE)

## boundary (singular) fit: see ?isSingular

mod17pk <- lmer(log(julian_median) ~ state + growth_habit + year_factor + (1 | plot),
  umbs_flwr_plot_growthhabit, REML = FALSE)
anova(mod15pk, mod16pk) # go with model 16pk

## Data: umbs_flwr_plot_growthhabit
## Models:
## mod16pk: log(julian_median) ~ state + growth_habit + (1 + year_factor |
## mod16pk:      plot)
## mod15pk: log(julian_median) ~ state * growth_habit + (1 + year_factor |
## mod15pk:      plot)
##      npar     AIC     BIC logLik deviance Chisq Df Pr(>Chisq)
## mod16pk    8 -1259.6 -1224.3 637.81   -1275.6
## mod15pk   10 -1256.4 -1212.2 638.19   -1276.4 0.7531   2      0.6862

anova(mod16pk, mod17pk) # mod 17pk

## Data: umbs_flwr_plot_growthhabit
## Models:
## mod17pk: log(julian_median) ~ state + growth_habit + year_factor + (1 |
## mod17pk:      plot)
## mod16pk: log(julian_median) ~ state + growth_habit + (1 + year_factor |
## mod16pk:      plot)
##      npar     AIC     BIC logLik deviance Chisq Df Pr(>Chisq)
## mod17pk    7 -1262.5 -1231.6 638.26   -1276.5
## mod16pk    8 -1259.6 -1224.3 637.81   -1275.6      0  1

```

```

AICctab(mod15pk, mod16pk, mod17pk, weights = T)

##          dAICc df weight
## mod17pk   0.0   7  0.787
## mod16pk   3.0   8  0.180
## mod15pk   6.3  10  0.033

summary(mod17pk)

## Linear mixed model fit by maximum likelihood . t-tests use Satterthwaite's
## method [lmerModLmerTest]
## Formula: log(julian_median) ~ state + growth_habit + year_factor + (1 |
##     plot)
## Data: umbs_flwr_plot_growthhabit
##
##          AIC      BIC    logLik deviance df.resid
## -1262.5 -1231.6    638.3   -1276.5      606
##
## Scaled residuals:
##      Min      1Q Median      3Q      Max
## -3.2072 -0.5881 -0.0492  0.4711  3.2640
##
## Random effects:
## Groups   Name        Variance Std.Dev.
## plot     (Intercept) 3.925e-05 0.006265
## Residual           7.260e-03 0.085206
## Number of obs: 613, groups: plot, 24
##
## Fixed effects:
##                   Estimate Std. Error      df t value Pr(>|t|)
## (Intercept)      5.318954  0.010535 279.429978 504.873 <2e-16 ***
## statewarmed      0.005897  0.007356  23.828948  0.802   0.431
## growth_habit     -0.219484  0.010089 604.507276 -21.755 <2e-16 ***
## growth_habitGraminoid -0.223689  0.007534 589.779435 -29.689 <2e-16 ***
## year_factor       0.002480  0.002194 598.154763   1.131   0.259
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##            (Intr) sttwrm grwth_ grwt_G
## statewarmed -0.352
## growth_habit -0.372  0.047
## grwth_hbtGr -0.334 -0.004  0.366
## year_factor  -0.782  0.006  0.118 -0.024

summ(mod17pk)

```

Observations	613
Dependent variable	log(julian_median)
Type	Mixed effects linear regression

AIC	-1262.53
BIC	-1231.60
Pseudo-R <sup>2</sup> (fixed effects)	0.62
Pseudo-R <sup>2</sup> (total)	0.63

Fixed Effects					
	Est.	S.E.	t val.	d.f.	p
(Intercept)	5.32	0.01	504.87	279.43	0.00
statewarmed	0.01	0.01	0.80	23.83	0.43
growth_habit	-0.22	0.01	-21.76	604.51	0.00
growth_habitGraminoid	-0.22	0.01	-29.69	589.78	0.00
year_factor	0.00	0.00	1.13	598.15	0.26

p values calculated using Satterthwaite d.f.

Random Effects		
Group	Parameter	Std. Dev.
plot	(Intercept)	0.01
	Residual	0.09

Grouping Variables		
Group	# groups	ICC
plot	24	0.01

```
anova(mod17pk)
```

```
## Type III Analysis of Variance Table with Satterthwaite's method
##          Sum Sq Mean Sq NumDF DenDF F value Pr(>F)
## state      0.0047  0.0047     1   23.83  0.6426 0.4307
## growth_habit 7.3938  3.6969     2 598.85 509.2054 <2e-16 ***
## year_factor  0.0093  0.0093     1 598.15  1.2782 0.2587
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
emmeans(mod17pk, list(pairwise ~ state + growth_habit + year_factor), adjust = "tukey")
```

```
## $`emmeans of state, growth_habit, year_factor`
## state  growth_habit year_factor emmean       SE    df lower.CL upper.CL
## ambient Forb            3.69 5.328 0.006763 66.27   5.315 5.342
## warmed  Forb            3.69 5.334 0.006776 65.90   5.320 5.348
## ambient
## warmed
## ambient Graminoid        3.69 5.109 0.009392 202.23   5.090 5.127
## warmed
## ambient Graminoid        3.69 5.115 0.009770 224.53   5.095 5.134
## warmed
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## Confidence level used: 0.95
```

```

## 
## $`pairwise differences of state, growth_habit, year_factor`
## 1
## ambient Forb 3.69004893964111 - warmed Forb 3.69004893964111
## ambient Forb 3.69004893964111 - ambient 3.69004893964111
## ambient Forb 3.69004893964111 - warmed 3.69004893964111
## ambient Forb 3.69004893964111 - ambient Graminoid 3.69004893964111
## ambient Forb 3.69004893964111 - warmed Graminoid 3.69004893964111
## warmed Forb 3.69004893964111 - ambient 3.69004893964111
## warmed Forb 3.69004893964111 - warmed 3.69004893964111
## warmed Forb 3.69004893964111 - ambient Graminoid 3.69004893964111
## warmed Forb 3.69004893964111 - warmed Graminoid 3.69004893964111
## ambient 3.69004893964111 - warmed 3.69004893964111
## ambient 3.69004893964111 - ambient Graminoid 3.69004893964111
## ambient 3.69004893964111 - warmed Graminoid 3.69004893964111
## warmed 3.69004893964111 - ambient Graminoid 3.69004893964111
## warmed 3.69004893964111 - warmed Graminoid 3.69004893964111
## ambient Graminoid 3.69004893964111 - warmed Graminoid 3.69004893964111
## estimate SE df t.ratio p.value
## -0.00590 0.00769 26.3 -0.767 0.9708
## 0.21948 0.01013 608.0 21.657 <.0001
## 0.21359 0.01300 192.2 16.432 <.0001
## 0.22369 0.00755 593.0 29.612 <.0001
## 0.21779 0.01076 104.8 20.237 <.0001
## 0.22538 0.01244 167.7 18.114 <.0001
## 0.21948 0.01013 608.0 21.657 <.0001
## 0.22959 0.01080 105.8 21.257 <.0001
## 0.22369 0.00755 593.0 29.612 <.0001
## -0.00590 0.00769 26.3 -0.767 0.9708
## 0.00421 0.01019 607.4 0.413 0.9985
## -0.00169 0.01247 169.6 -0.136 1.0000
## 0.01010 0.01306 195.1 0.774 0.9716
## 0.00421 0.01019 607.4 0.413 0.9985
## -0.00590 0.00769 26.3 -0.767 0.9708
##
## Degrees-of-freedom method: kenward-roger
## Results are given on the log (not the response) scale.
## P value adjustment: tukey method for comparing a family of 6 estimates

```