STAT 414 - Part 2 Update

Kyle Nessen, Justin Mai, Aiden Kelly, Arneh Begi

2024-10-28

Introduction

Every fall, monarch butterflies migrate from the western United States to coastal California in search of forested areas where they cluster in large numbers. It is hypothesized that the butterflies seek these habitats or groves because they offer a particular microclimate regarding temperature, humidity, and light. Within each grove, butterflies often cluster at the same tree (and even branch), year after year, spurring scientists to hypothesize that these areas are selected for their microclimatic characteristics over other areas within the same grove. Saniee and Villablanca (2022) tested this hypothesis directly by installing weather station 'arrays' at the butterfly clustering sites ('cluster') and four orthogonal positions from the butterflies (NW, NE, SW, SE). They repeated these weather station arrays at eight overwintering groves along an N-S gradient along the California coast.

stations are or

will include pictu

Our research question investigates whether climatic conditions are significantly different at the cluster site compared to the four controls (NW, NE, SW, SE) and whether they are consistent across groves at cluster sites.

------?

```
df <- suppressMessages(read_csv('allgr_array_KianaRawdat.csv'))
head(df)
```

```
# A tibble: 6 × 21
  ...1 array   month.day temp.avg hum.avg dew.pt.avg light.avg light.min
 <dbl> <chr>   <chr>       <dbl>   <dbl>     <dbl>     <dbl>     <dbl>
1     1 Cluster 01-01        11.7    46.6     -1.45     26005.         0
2     2 Cluster 01-02         9.36   57.2     -0.605    25292.         0
3     3 Cluster 01-03        10.4    62.5      1.62     25430.         0
4     4 Cluster 01-04        10.6    71.1      3.56     22243.         0
5     5 Cluster 01-05         9.77   88.7      7.71       766.         0
6     6 Cluster 01-06        12.4    94.5     10.9      12197.         0
# ℹ 13 more variables: light.max <dbl>, temp.min <dbl>, temp.max <dbl>,
#   temp.std <dbl>, light.std <dbl>, dew.pt.std <dbl>, hum.std <dbl>,
#   hum.max <dbl>, hum.min <dbl>, dp.max <dbl>, dp.min <dbl>, daynum <dbl>,
#   grove <chr>
```

Weight observations based on time of season

The monarch overwintering season occurs from the beginning of October through the end of February. Weather can be unpredictable at the beginning and end of the season (fall and spring), and butterflies either arrive at groves or leave to begin breeding; thus, they generally occur in low numbers. The highest counts of butterflies occur around

Thanksgiving and Christmas, or approximately in the middle of the season. Scientists are usually most "concerned" about storms during this time.

One idea to account for this is to weigh observations based on how far away they are from the middle of the season. We could derive day of the season (seasonDay) from the date column (10/1 = 1, 10/2 = 2), then grand mean center our seasonDay variable. If we take the absolute value of seasonDay, days near the middle of the season would be small, and days toward the beginning or end would be large. We could weigh by dividing one by seasonDay. We are presenting this idea to see if it is worth pursuing or if it is a route we should take with caution.

own idea or fol

weighting the data or weighted least squares? Why not just trim the ends of the time interval?

Not the worst idea :) Can always do both ways and compare. Mi

Unequal samples across groves

Since not all groves have the same number of observations throughout the overwintering season, should we restrict our analysis to dates where the groves have data? A benefit is that it may provide a better comparison of conditions across sites, but it could also reduce our sample size and miss important data in grove specific conditions on unmonitored dates.

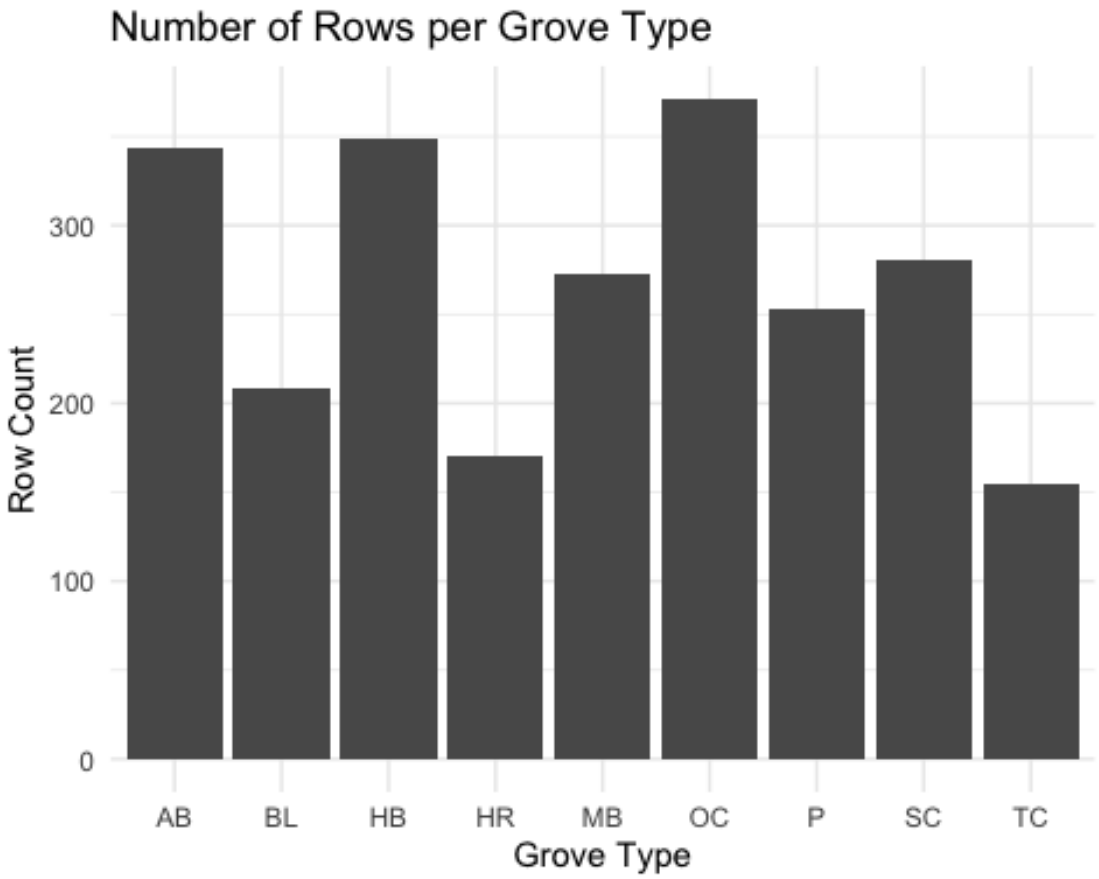
how often does

```
grove_counts <- df %>%
  count(grove)

# Plot the counts as a bar chart
ggplot(grove_counts, aes(x = grove, y = n)) +
  geom_bar(stat = "identity") +
  labs(title = "Number of Rows per Grove Type", x = "Grove Type", y = "Row Count") +
  theme_minimal()
```

Need to clarify level 1 unit and r

Do you mean grove type or just grove (name)?



I don't think I know wl

Exploratory plots of response variables

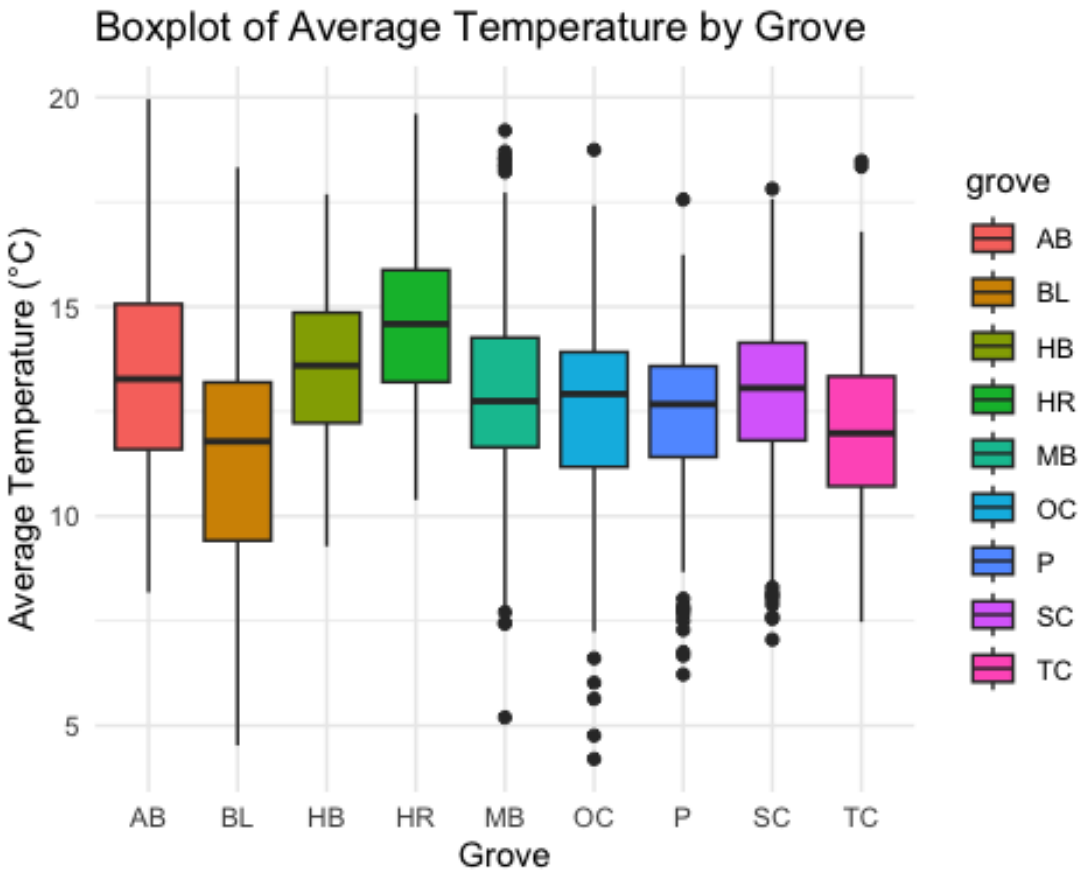
Below are box plots of the response variables we plan to test by grove.

```
#Reponse variables by Grove

# Boxplot for Average Temperature by Grove
ggplot(df, aes(x = grove, y = temp.avg, fill = grove)) +
  geom_boxplot() +
  labs(title = "Boxplot of Average Temperature by Grove",
       x = "Grove", y = "Average Temperature (°C)") +
  theme_minimal()
```

do you really mean respons

```
Warning: Removed 31 rows containing non-finite outside the scale range
(`stat_boxplot()`).
```

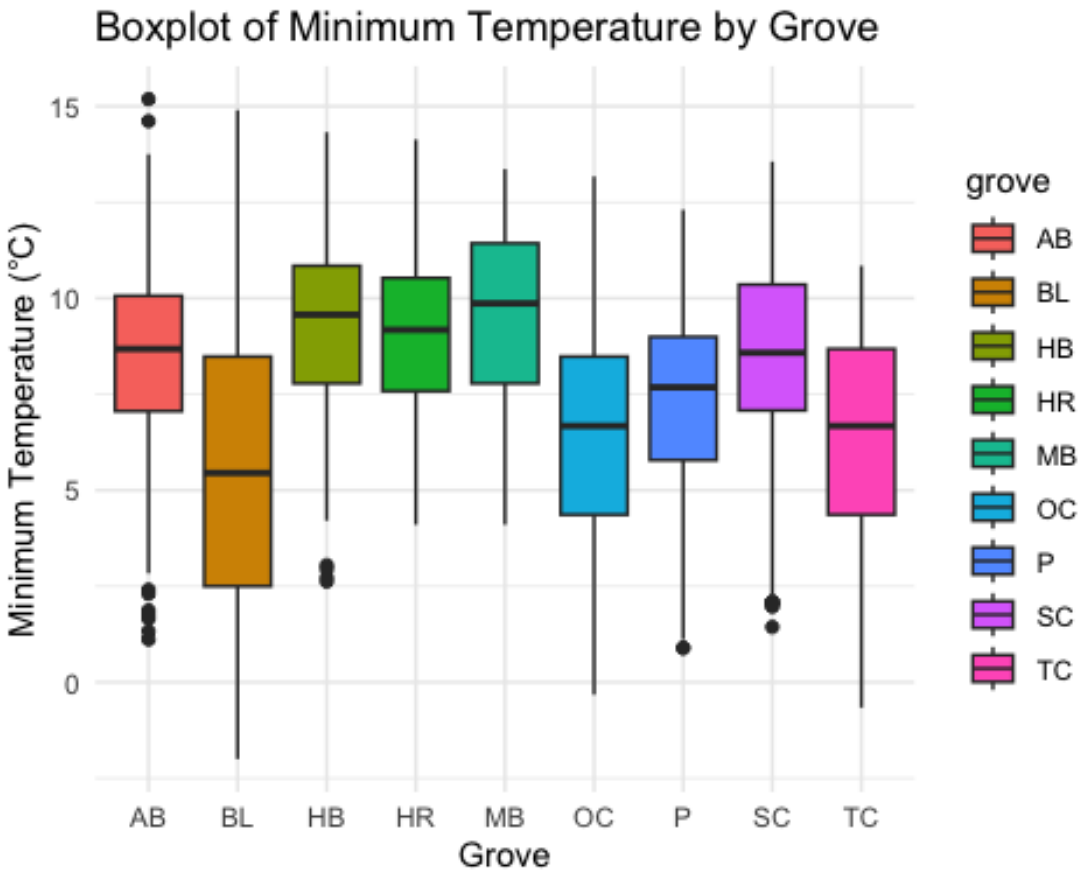


and of course not c

so you are getting

```
# Boxplot for Minimum Temperature by Grove
ggplot(df, aes(x = grove, y = temp.min, fill = grove)) +
  geom_boxplot() +
  labs(title = "Boxplot of Minimum Temperature by Grove",
       x = "Grove", y = "Minimum Temperature (°C)") +
  theme_minimal()
```

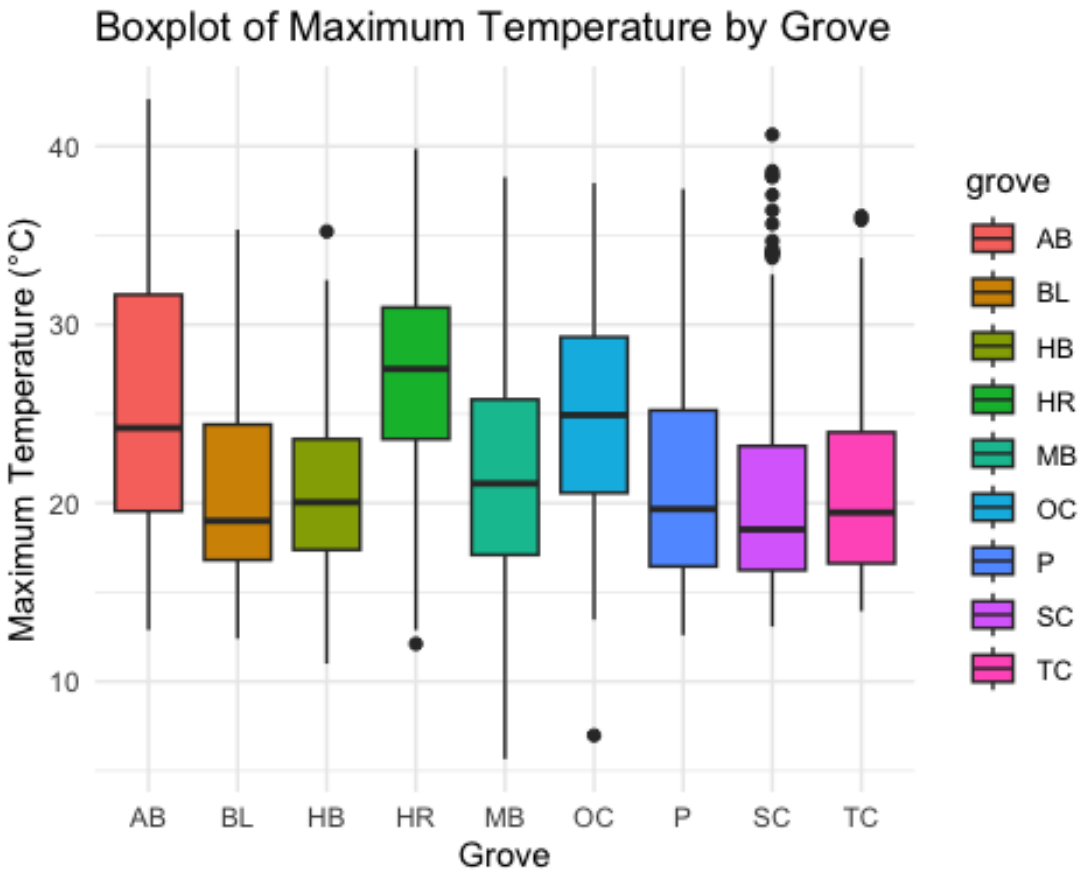
```
Warning: Removed 31 rows containing non-finite outside the scale range
(`stat_boxplot()`).
```




```
# Boxplot for Maximum Temperature by Grove
ggplot(df, aes(x = grove, y = temp.max, fill = grove)) +
  geom_boxplot() +
  labs(title = "Boxplot of Maximum Temperature by Grove",
       x = "Grove", y = "Maximum Temperature (°C)") +
  theme_minimal()
```

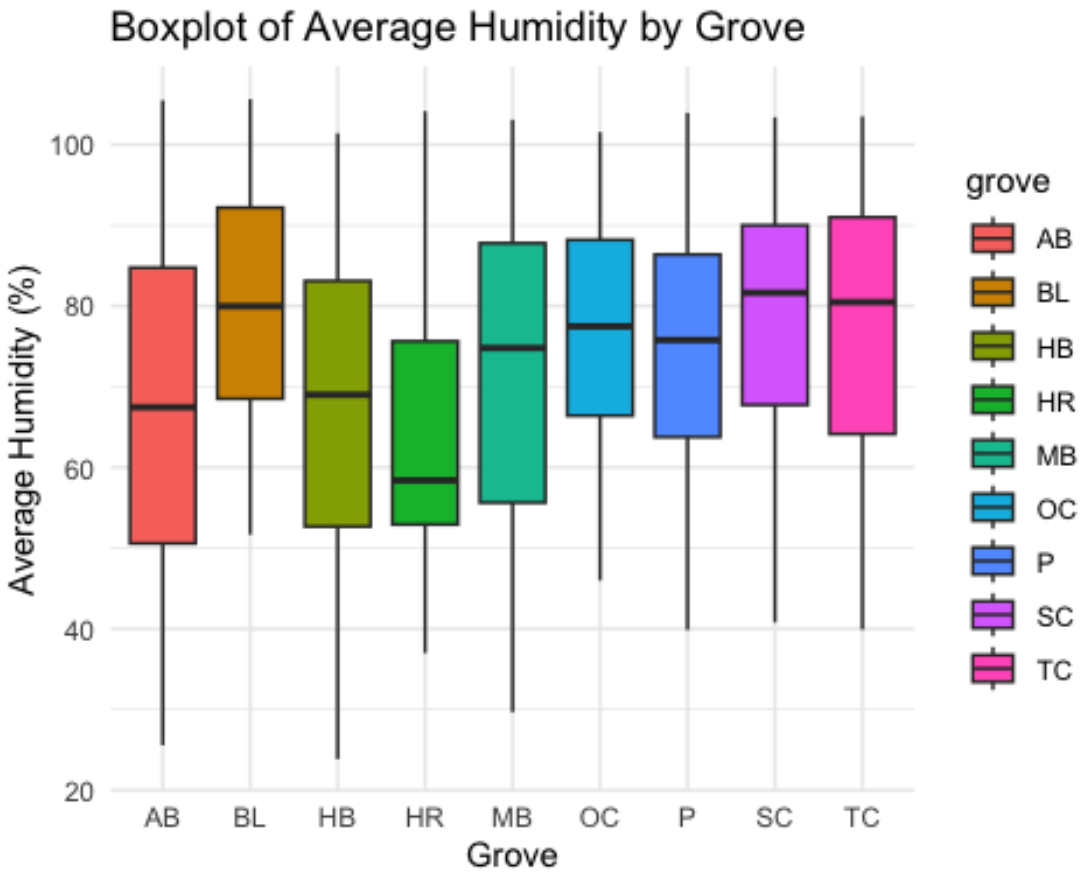
Warning: Removed 31 rows containing non-finite outside the scale range (`stat_boxplot()`).

Hmm, maybe even conside



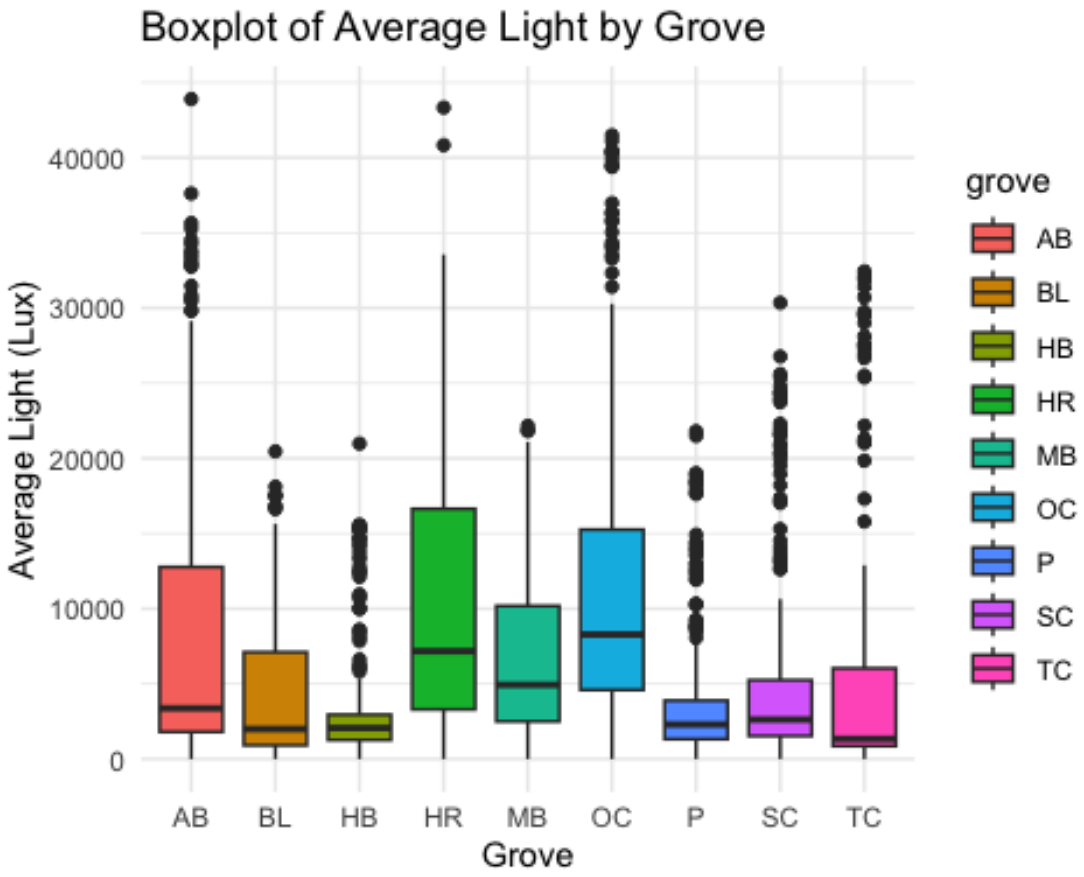

```
# Boxplot for Average Humidity by Grove
ggplot(df, aes(x = grove, y = hum.avg, fill = grove)) +
  geom_boxplot() +
  labs(title = "Boxplot of Average Humidity by Grove",
       x = "Grove", y = "Average Humidity (%)") +
  theme_minimal()
```

Warning: Removed 618 rows containing non-finite outside the scale range (`stat_boxplot()`).



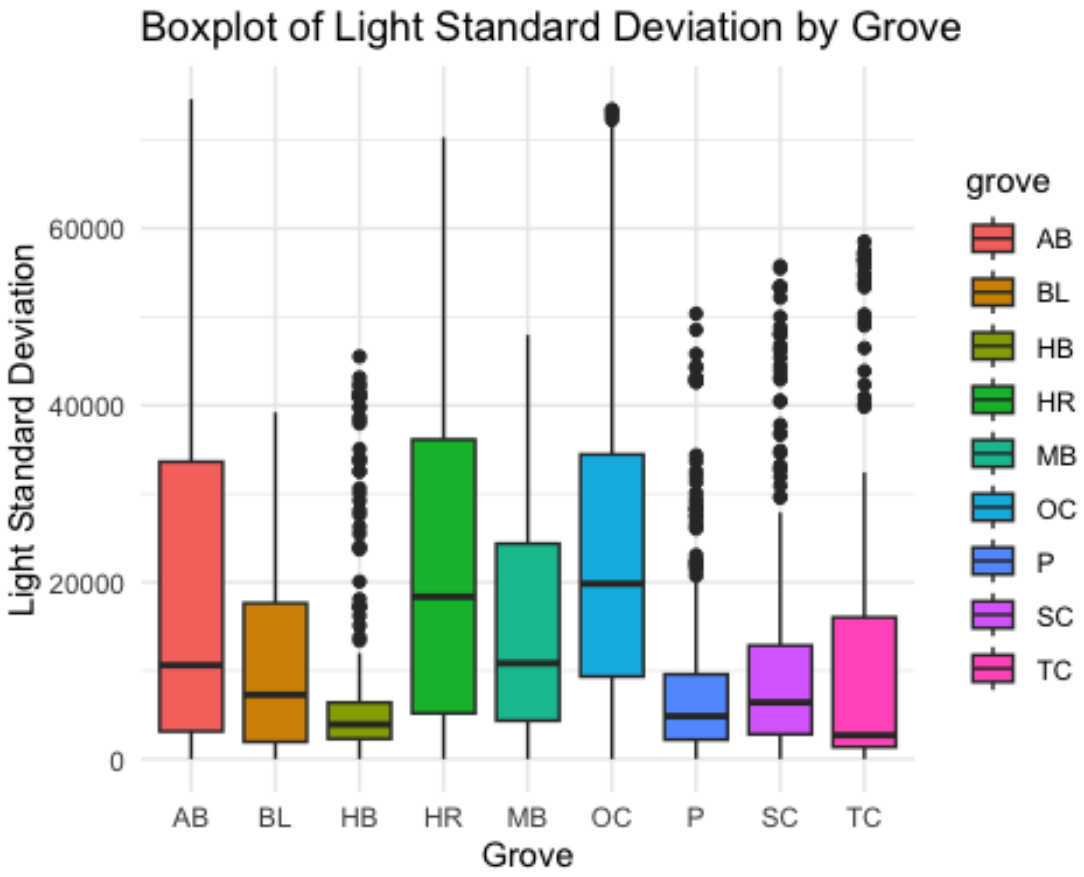

```
# Boxplot for Average Light by Grove
ggplot(df, aes(x = grove, y = light.avg, fill = grove)) +
  geom_boxplot() +
  labs(title = "Boxplot of Average Light by Grove",
       x = "Grove", y = "Average Light (Lux)") +
  theme_minimal()
```

Warning: Removed 109 rows containing non-finite outside the scale range (`stat_boxplot()`).




```
# Boxplot for Light Variability (Std Dev) by Grove
ggplot(df, aes(x = grove, y = light.std, fill = grove)) +
  geom_boxplot() +
  labs(title = "Boxplot of Light Standard Deviation by Grove",
       x = "Grove", y = "Light Standard Deviation") +
  theme_minimal()
```

Warning: Removed 109 rows containing non-finite outside the scale range (`stat_boxplot()`).



Ooh, do have an S

Preliminary models

Below are a few preliminary models to test our questions. These likely will change in the final report, but we provide them here to give a sense of how we plan to approach the investigation.

Arrays as fixed effect

```
model1 <- lmer(temp.avg ~ array + (1 | grove), data = df)
summary(model1, corr = FALSE)

Linear mixed model fit by REML ['lmerMod']
Formula: temp.avg ~ array + (1 | grove)
   Data: df

REML criterion at convergence: 10386.3

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.9984 -0.6595  0.0637  0.6528  3.3479

Random effects:
 Groups   Name        Variance Std.Dev.
 grove    (Intercept) 0.7796   0.8829
```

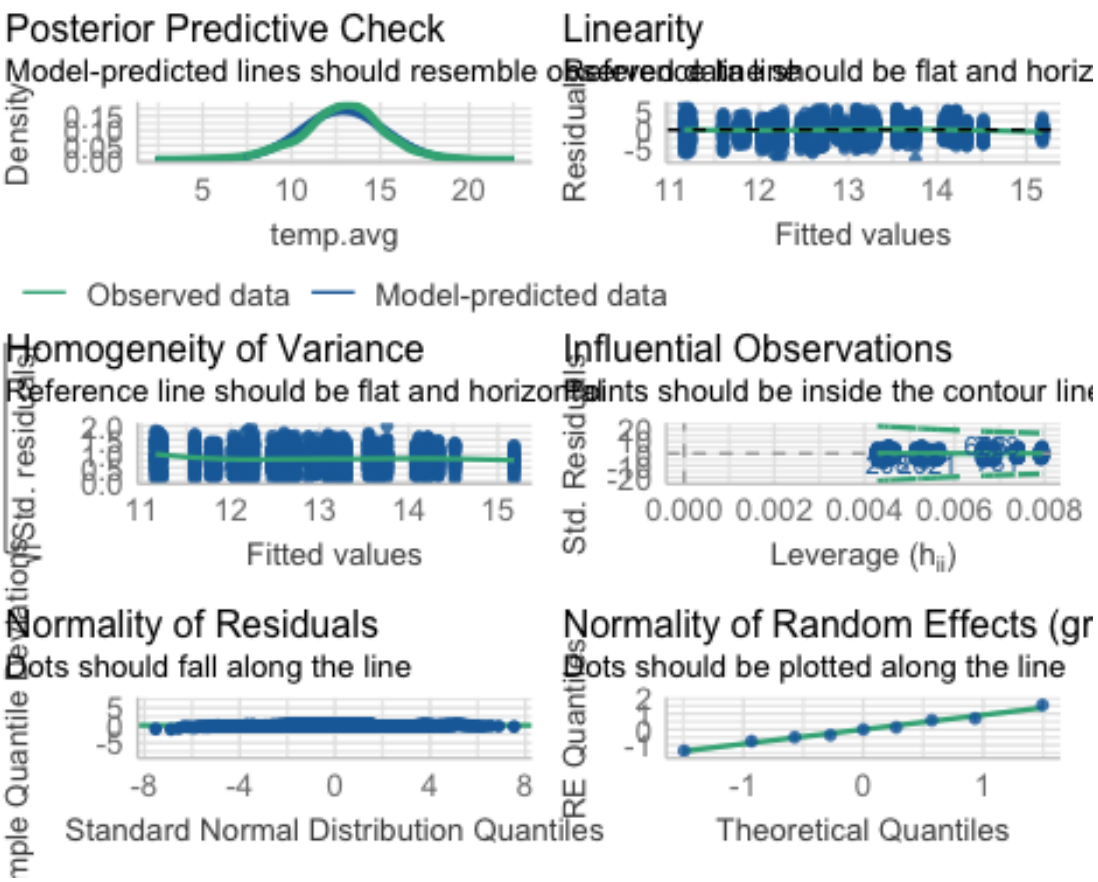
what is array?

so basically you are trying to see whether the

```
 Residual             4.5834   2.1409
Number of obs: 2373, groups:  grove, 9

Fixed effects:
            Estimate Std. Error t value
(Intercept)  12.9618     0.3104  41.763
arrayNE      -0.3667     0.1391  -2.636
arrayNW      -0.4395     0.1378  -3.190
arraySE       0.6640     0.1388   4.782
arraySW      -0.4016     0.1390  -2.889
```

```
check_model(model1)
```



Grove as fixed effect

```
model2 <- lmer(temp.avg ~ grove + (1 | array), data = df)
summary(model2, corr = FALSE)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: temp.avg ~ grove + (1 | array)
   Data: df

REML criterion at convergence: 10370

Scaled residuals:
```

```
   Min     1Q  Median     3Q    Max 
-3.9835 -0.6658  0.0623  0.6524  3.3600 

Random effects:
 Groups   Name        Variance Std.Dev.
 array    (Intercept) 0.208    0.4561  
 Residual             4.583    2.1409  
Number of obs: 2373, groups:  array, 5

Fixed effects:
            Estimate Std. Error t value
(Intercept)  13.4506     0.2344  57.392
groveBL      -2.0033     0.1923 -10.420
groveHB       0.1209     0.1639   0.738
groveHR       1.0124     0.2007   5.044
groveMB      -0.4655     0.1735  -2.682
groveOC      -0.9669     0.1607  -6.017
groveP       -1.1043     0.1780  -6.206
groveSC      -0.6006     0.1722  -3.489
groveTC      -1.3787     0.2071  -6.657
```

```
check_model(model2)
```