

# WK5

2025-07-31

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.5
## v forcats    1.0.0      v stringr   1.5.1
## v ggplot2    3.5.2      v tibble    3.2.1
## v lubridate  1.9.4      v tidyr     1.3.1
## v purrr      1.0.4
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(readxl)
library(lubridate)
library(janitor)
```

```
##
## Attaching package: 'janitor'
##
## The following objects are masked from 'package:stats':
##
##   chisq.test, fisher.test
```

```
library(purrr)
library(readr)
library(ggthemes)
library(ggeffects)
library(lme4)
```

```
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##   expand, pack, unpack
```

```
#setwd("C:/Users/Tobyz/Desktop/S.tort-light-growth/Data")
```

```
import plant data
```

```

plant <- read.csv("Data/WL2-2023_Size_Combined.csv") %>%
  clean_names() %>%
  mutate(survey_date = as.Date(survey_date, format = "%m/%d/%Y"))
summary(plant)

```

```

##   survey_date      block      genotype      pop_mf
##   Min.   :2023-07-03   Length:17336   Length:17336   Length:17336
##   1st Qu.:2023-08-02   Class :character   Class :character   Class :character
##   Median :2023-08-30   Mode  :character   Mode  :character   Mode  :character
##   Mean   :2023-08-28
##   3rd Qu.:2023-09-20
##   Max.   :2023-10-20
##
##   parent_pop      mf      rep      height_cm
##   Length:17336    Min.   : 1.000   Min.   : 1.000   Min.   : 0.100
##   Class :character 1st Qu.: 2.000   1st Qu.: 4.000   1st Qu.: 1.700
##   Mode  :character Median : 5.000   Median : 8.000   Median : 3.100
##                   Mean   : 4.584   Mean   : 7.932   Mean   : 4.491
##                   3rd Qu.: 6.000   3rd Qu.:11.000   3rd Qu.: 5.700
##                   Max.   :14.000   Max.   :27.000   Max.   :39.400
##                                     NA's   :8762
##   long_leaf_cm  survey_notes
##   Min.   :0.100   Length:17336
##   1st Qu.:1.600   Class :character
##   Median :2.500   Mode  :character
##   Mean   :2.599
##   3rd Qu.:3.500
##   Max.   :9.000
##   NA's   :9350

```

*consolidate light measurement to a weekly measurement*

```

#import light data
light_raw <- read_csv("Data/IntBioHalfHourTable_clean.txt")

```

```

## Rows: 4063 Columns: 139
## -- Column specification -----
## Delimiter: ","
## dbl   (138): RECORD, BattV_Max, PTemp_C_Max, SlrW_Avg, SlrW_Max, SlrW_Min, Sl...
## dtm    (1): TIMESTAMP
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

```

```

#weekly measurement
weekly_light <- light_raw %>%
  mutate(
    timestamp = ymd_hms(TIMESTAMP),
    SlrW_Avg = as.numeric(SlrW_Avg), # turn into number format
    week = floor_date(timestamp, "week")
  ) %>%
  group_by(week) %>%

```

```

summarise(
  weekly_avg_SlrW = mean(SlrW_Avg, na.rm = TRUE),
  .groups = "drop"
)

```

```

## Warning: There was 1 warning in 'mutate()'.
## i In argument: 'timestamp = ymd_hms(TIMESTAMP)'.
## Caused by warning:
## ! 84 failed to parse.

```

```

# result
print(weekly_light)

```

```

## # A tibble: 14 x 2
##   week                weekly_avg_SlrW
##   <dtm>                <dbl>
## 1 2023-07-30 00:00:00          280.
## 2 2023-08-06 00:00:00          277.
## 3 2023-08-13 00:00:00          186.
## 4 2023-08-20 00:00:00          184.
## 5 2023-08-27 00:00:00          200.
## 6 2023-09-03 00:00:00          211.
## 7 2023-09-10 00:00:00          204.
## 8 2023-09-17 00:00:00          189.
## 9 2023-09-24 00:00:00          159.
## 10 2023-10-01 00:00:00          138.
## 11 2023-10-08 00:00:00          133.
## 12 2023-10-15 00:00:00          135.
## 13 2023-10-22 00:00:00          116.
## 14 NA                      -0.616

```

*Investigate or filter out plants that show negative growth*

```

#PID
plant_growth <- plant %>%
  unite("PID", genotype:rep, sep = "_", remove = FALSE) %>%
  mutate(survey_date = as.Date(survey_date))

#find out plants with negative growth
plant_growth %>%
  arrange(PID, survey_date) %>% # arrange in time sequence
  group_by(PID) %>%
  mutate(growth = height_cm - lag(height_cm)) %>% # find out the diff btw nearby dates
  summarise(has_negative_growth = any(growth < 0, na.rm = TRUE)) %>%
  filter(has_negative_growth) -> neg_growth_plants
neg_growth_plants

```

```

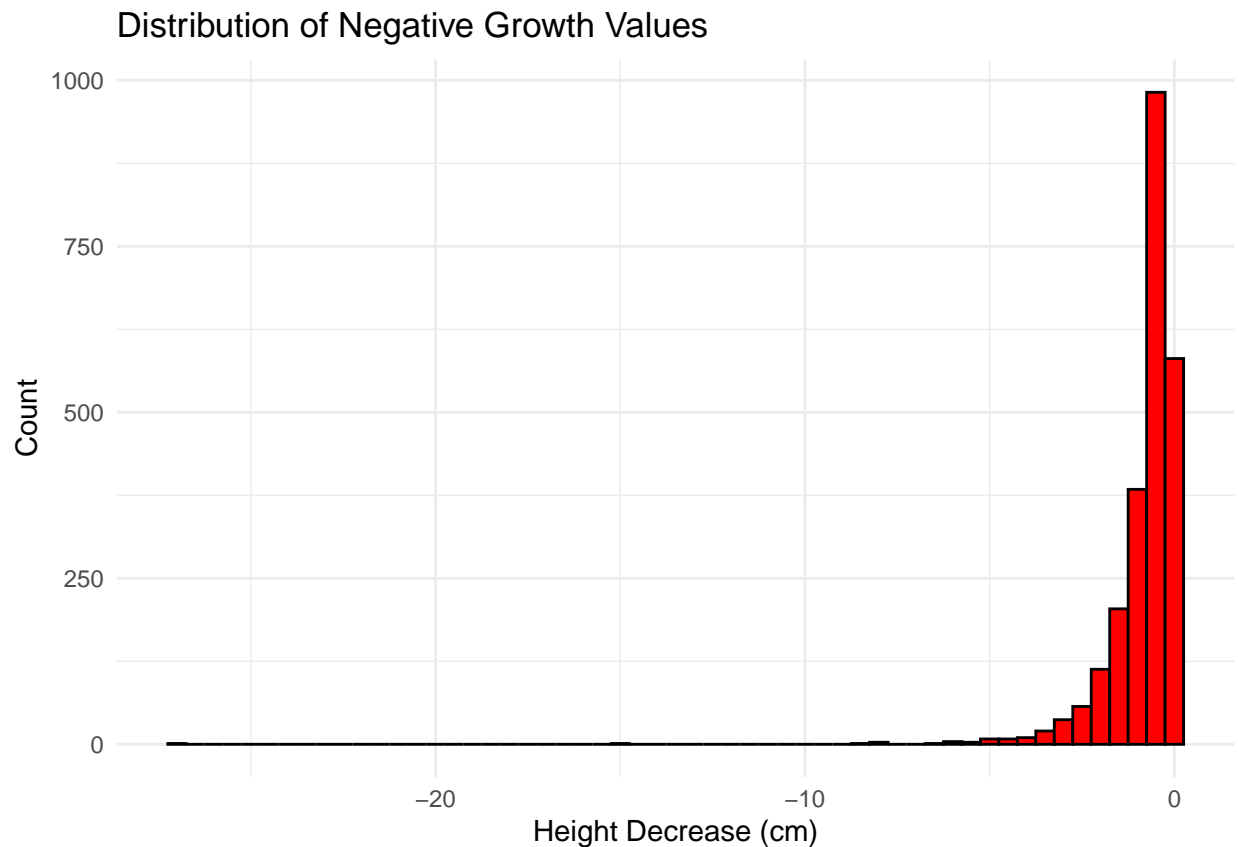
## # A tibble: 826 x 2
##   PID                has_negative_growth
##   <chr>                <lgl>
## 1 BH_1_10_BH_1_10 TRUE
## 2 BH_1_12_BH_1_12 TRUE

```

```
## 3 BH_1_13_BH_1_BH_1_13 TRUE
## 4 BH_1_1_BH_1_BH_1_1 TRUE
## 5 BH_1_4_BH_1_BH_1_4 TRUE
## 6 BH_1_7_BH_1_BH_1_7 TRUE
## 7 BH_2_10_BH_2_BH_2_10 TRUE
## 8 BH_2_11_BH_2_BH_2_11 TRUE
## 9 BH_2_12_BH_2_BH_2_12 TRUE
## 10 BH_2_13_BH_2_BH_2_13 TRUE
## # i 816 more rows
```

```
#find out tolerance value
neg_growth_values <- plant_growth %>%
  arrange(PID, survey_date) %>%
  group_by(PID) %>%
  mutate(growth = height_cm - lag(height_cm)) %>%
  ungroup() %>%
  filter(growth < 0)

ggplot(neg_growth_values, aes(x = growth)) +
  geom_histogram(binwidth = 0.5, fill = "red", color = "black") +
  labs(
    title = "Distribution of Negative Growth Values",
    x = "Height Decrease (cm)",
    y = "Count"
  ) +
  theme_minimal()
```



```

#filter out plants with negative growth < -5
plant_growth_cleaned <- plant_growth

repeat {
  plant_growth_cleaned <- plant_growth_cleaned %>%
    arrange(PID, survey_date) %>%
    group_by(PID) %>%
    mutate(growth = height_cm - lag(height_cm)) %>%
    filter(is.na(growth) | growth >= -5) %>%
    select(-growth) %>%
    ungroup()

  check <- plant_growth_cleaned %>%
    arrange(PID, survey_date) %>%
    group_by(PID) %>%
    mutate(growth = height_cm - lag(height_cm)) %>%
    filter(growth < -5)

  if (nrow(check) == 0) break
}

```

#### *Measure Growth via Daily Growth Rate*

```

#define daily growth rate
plant_growth_daily <- plant_growth_cleaned %>%
  arrange(PID, survey_date) %>%
  group_by(PID) %>%
  mutate(
    prev_height = lag(height_cm),
    prev_date = lag(survey_date),
    days_elapsed = as.numeric(survey_date - prev_date),
    daily_growth = (height_cm - prev_height) / days_elapsed
  ) %>%
  ungroup()

```

#### *Correlate Growth with Solar Radiation*

```

#Align plant growth data to week
plant_weekly <- plant_growth_daily %>%
  filter(!is.na(daily_growth), days_elapsed > 0) %>%
  mutate(week = floor_date(survey_date, "week"))

#Adds `weekly_avg_SlrW` to plant data
plant_with_light <- plant_weekly %>%
  left_join(weekly_light, by = "week")

#Calculate correlation
cor_result <- cor(
  plant_with_light$daily_growth,
  plant_with_light$weekly_avg_SlrW,
  use = "complete.obs"
)

print(cor_result)

```

```
## [1] 0.1299395
```

```
#Plot correlation
```

```
ggplot(plant_with_light, aes(x = weekly_avg_SlrW, y = daily_growth)) +  
  geom_point(alpha = 0.2) +  
  geom_smooth(method = "lm", color = "red") +  
  labs(title = "Correlation between Light and Growth", x = "Weekly Avg Light (SlrW)", y = "Daily Growth")  
  theme_bw()
```

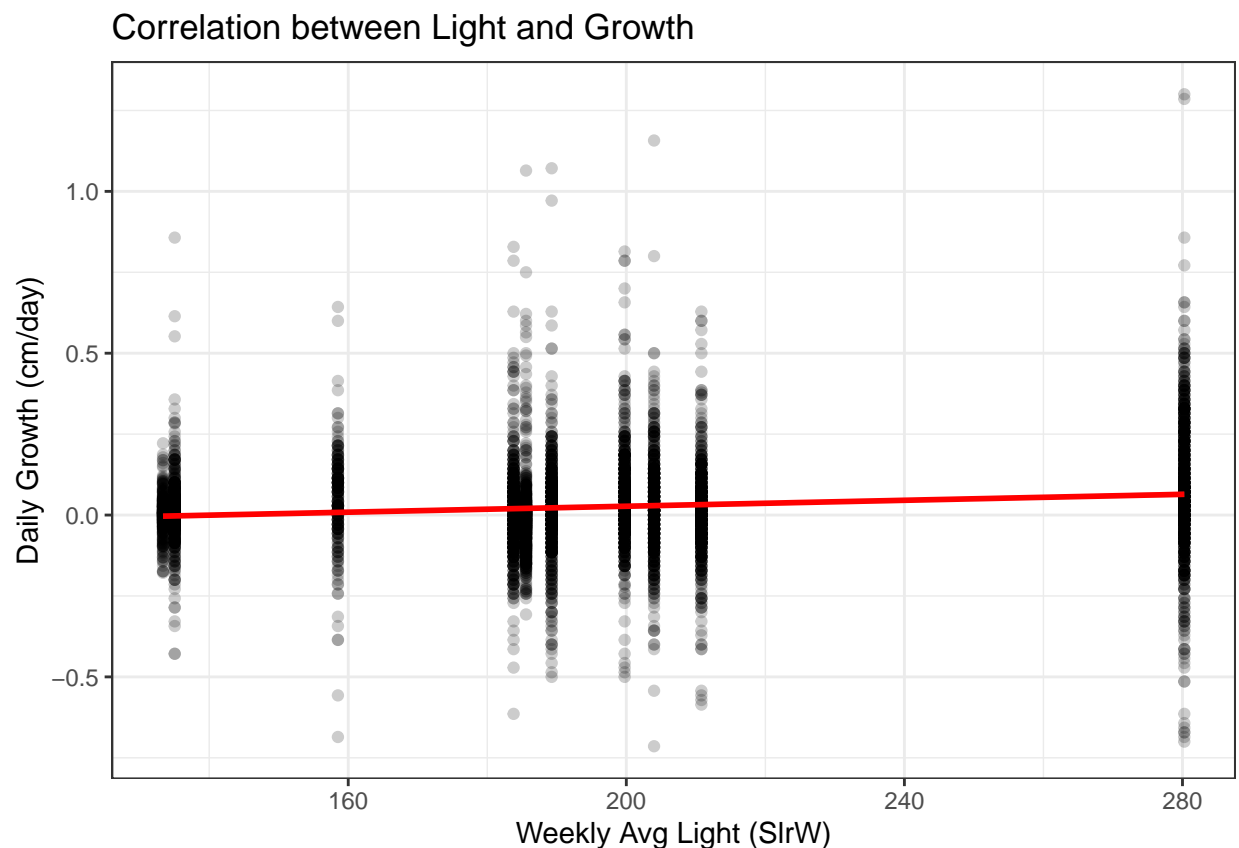
```
## 'geom_smooth()' using formula = 'y ~ x'
```

```
## Warning: Removed 1069 rows containing non-finite outside the scale range
```

```
## ('stat_smooth()').
```

```
## Warning: Removed 1069 rows containing missing values or values outside the scale range
```

```
## ('geom_point()').
```



*Relationship btw Light and Growth by filtering out the effect of block via Mixed-effect Model*

```
#Standardization
```

```
plant_with_light$weekly_avg_SlrW2 <- scale(plant_with_light$weekly_avg_SlrW, center = TRUE, scale = TRUE)
```

```

#Model
growth_block.lmer <- lmer(
  daily_growth ~ weekly_avg_SlrW2 + (1 | block),
  data = plant_with_light
)
summary(growth_block.lmer)

## Linear mixed model fit by REML ['lmerMod']
## Formula: daily_growth ~ weekly_avg_SlrW2 + (1 | block)
## Data: plant_with_light
##
## REML criterion at convergence: -5986.6
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -5.2983 -0.4760 -0.0364  0.4131  8.4083
##
## Random effects:
## Groups Name Variance Std.Dev.
## block (Intercept) 0.0001089 0.01044
## Residual 0.0209954 0.14490
## Number of obs: 5870, groups: block, 13
##
## Fixed effects:
##              Estimate Std. Error t value
## (Intercept)  0.026553  0.003498  7.592
## weekly_avg_SlrW2 0.018923  0.001893  9.997
##
## Correlation of Fixed Effects:
##              (Intr)
## wkly_vg_SW2 -0.003

#Faceting
(mmblock <- ggplot(plant_with_light, aes(x = weekly_avg_SlrW2, y = daily_growth)) +
  facet_wrap(~block, nrow = 2) +
  geom_point(alpha = 0.3) +
  geom_line(
    data = cbind(plant_with_light, pred = predict(growth_block.lmer, na.action = na.exclude)),
    aes(y = pred),
    size = 1
  ) +
  theme_bw() +
  theme(
    legend.position = "none",
    panel.spacing = unit(2, "lines"),
    axis.text.x = element_text(angle = 45, hjust = 1)
  ) +
  labs(title = "Effect of Light on Daily Growth Rate in each Block",
    x = "Weekly Avg Light (SlrW)",
    y = "Daily Growth Rate (cm/day)")
)

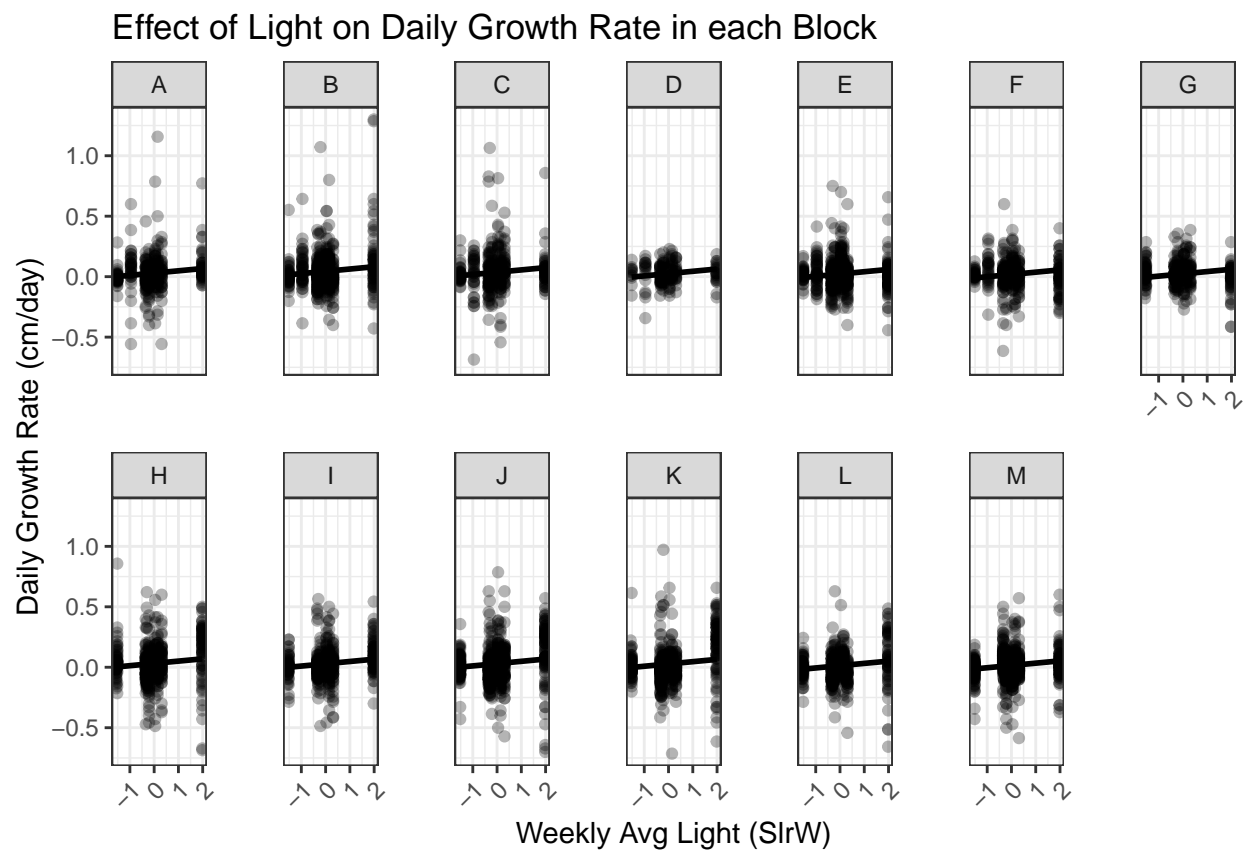
```

```
## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
```

```
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.

## Warning: Removed 1069 rows containing missing values or values outside the scale range
## ('geom_point()').

## Warning: Removed 109 rows containing missing values or values outside the scale range
## ('geom_line()').
```

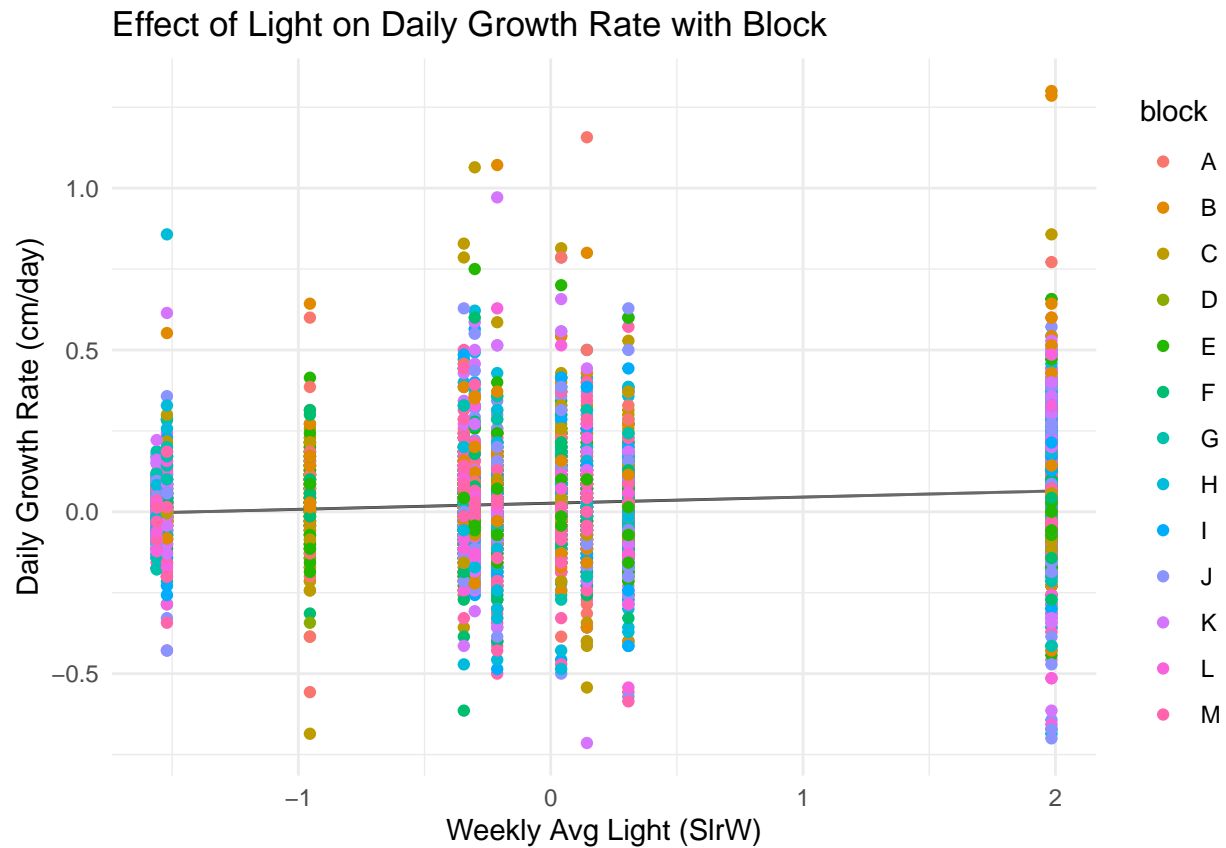


```
# Extract the prediction data frame
pred.block <- ggpredict(growth_block.lmer, terms = c("weekly_avg_SlrW2")) # this gives overall predict

# Plot the predictions
(ggplot(pred.block) +
  geom_line(aes(x = x, y = predicted)) + # slope
  geom_ribbon(aes(x = x, ymin = predicted - std.error, ymax = predicted + std.error),
    fill = "lightgrey", alpha = 0.5) + # error band
  geom_point(data = plant_with_light, # adding the raw data (scaled values)
    aes(x = weekly_avg_SlrW2, y = daily_growth, colour = block)) +
  labs(x = "Weekly Avg Light (SlrW)", y = "Daily Growth Rate (cm/day)",
    title = "Effect of Light on Daily Growth Rate with Block") +
  theme_minimal()
)
```



```
## Warning: Removed 1069 rows containing missing values or values outside the scale range
## ('geom_point()').
```



*Relationship btw Light and Growth by filtering out the effect of population via Mixed-effect Model*

```
#Model
growth_pop.lmer <- lmer(
  daily_growth ~ weekly_avg_SlrW2 + (1|parent_pop),
  data = plant_with_light
)
summary(growth_pop.lmer)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: daily_growth ~ weekly_avg_SlrW2 + (1 | parent_pop)
## Data: plant_with_light
##
## REML criterion at convergence: -6406
##
## Scaled residuals:
##    Min      1Q  Median      3Q     Max
## -5.8571 -0.4648  0.0064  0.4301  8.2806
##
## Random effects:
## Groups      Name                Variance Std.Dev.
## parent_pop (Intercept) 0.001572 0.03965
## Residual              0.019390 0.13925
```

```
## Number of obs: 5870, groups:  parent_pop, 22
##
## Fixed effects:
##               Estimate Std. Error t value
## (Intercept)    0.017534   0.008756   2.003
## weekly_avg_SlrW2 0.021165   0.001824  11.605
##
## Correlation of Fixed Effects:
##               (Intr)
## wkly_vg_SW2 -0.005
```

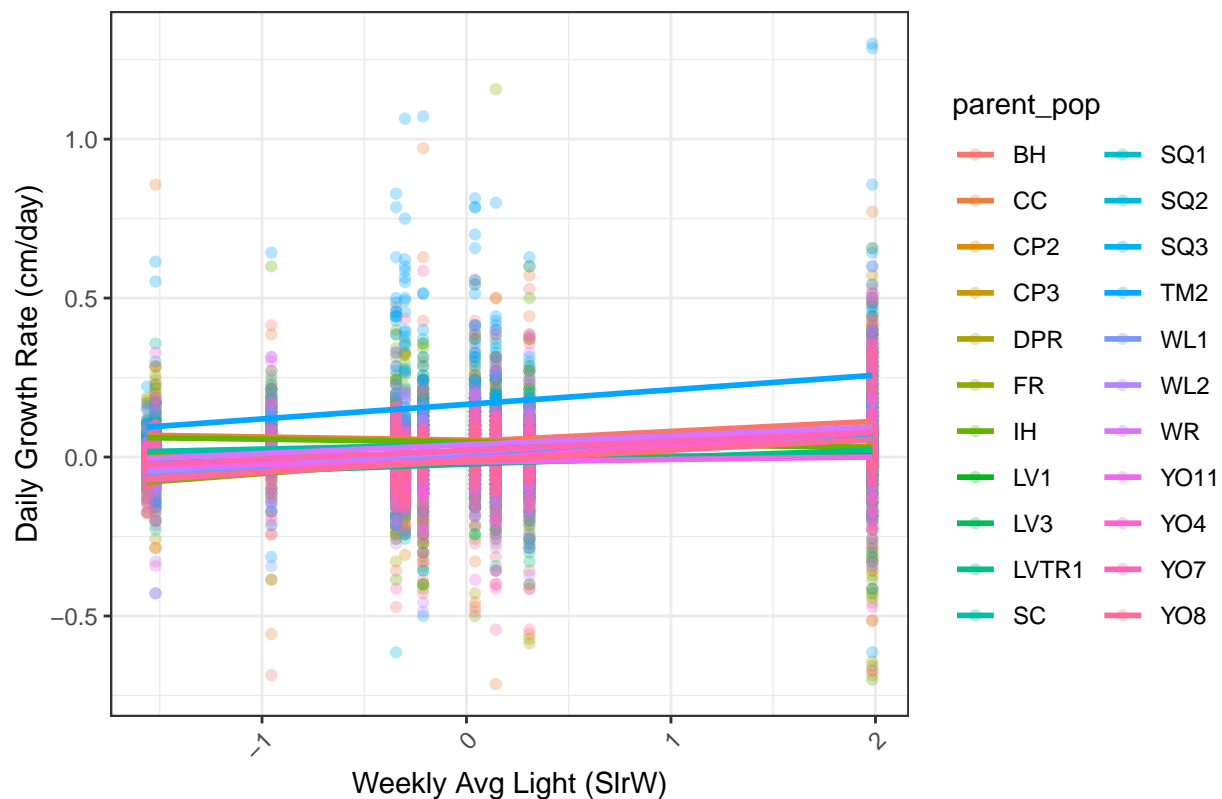
```
#Smooth
(mmblock <- ggplot(plant_with_light, aes(x = weekly_avg_SlrW2, y = daily_growth, colour = parent_pop)) +
  #facet_wrap(~parent_pop) +
  geom_point(alpha = 0.3) +
  #geom_line(
  # data = cbind(plant_with_light, pred = predict(growth_pop.lmer,na.action = na.exclude)),
  # aes(y = pred),
  # size = 1
  #) +
  geom_smooth(method="lm",se=FALSE)+
  theme_bw() +
  theme(
    legend.position = "right",
    panel.spacing = unit(2, "lines"),
    axis.text.x = element_text(angle = 45, hjust = 1)
  ) +
  labs(title = "Effect of Light on Daily Growth Rate in each Popolation",
    x = "Weekly Avg Light (SlrW)",
    y = "Daily Growth Rate (cm/day)")
)
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```

```
## Warning: Removed 1069 rows containing non-finite outside the scale range
## ('stat_smooth()').
```

```
## Warning: Removed 1069 rows containing missing values or values outside the scale range
## ('geom_point()').
```

Effect of Light on Daily Growth Rate in each Population



```
#Prediction
pred.pop <- ggpredict(growth_pop.lmer, terms = c("weekly_avg_SlrW2"))

(ggplot(pred.pop) +
  geom_line(aes(x = x, y = predicted)) + # slope
  geom_ribbon(aes(x = x, ymin = predicted - std.error, ymax = predicted + std.error),
    fill = "lightgrey", alpha = 0.5) + # error band
  geom_point(data = plant_with_light, # adding the raw data (scaled values)
    aes(x = weekly_avg_SlrW2, y = daily_growth, colour = parent_pop)) +
  labs(x = "Weekly Avg Light (SlrW)", y = "Daily Growth Rate (cm/day)",
    title = "Effect of Light on Daily Growth Rate with Population") +
  theme_minimal()
)
```

```
## Warning: Removed 1069 rows containing missing values or values outside the scale range
## ('geom_point()').
```

Effect of Light on Daily Growth Rate with Population

