

512 Project Part II

Jared Adam

10/31/2024

Part I (512 only, project proposal, 25 pts):

1) Read in your data set and run dim on it:

```
set.seed(654321)
s21 <- read_csv('data/2021 Sentinel Prey Assessment.csv')
s22 <- read_csv("data/PSA_CE2_SentinelPrey.csv")
s23 <- read_csv('data/PSA_Sent.prey.2023.csv')

# I need to get total predation into a column as a binary. 1 = predation, 0 = not

# 2021 cleaning
s21

## # A tibble: 5,281 × 17
##   location date      n.weather d.weather growth_stage plot_id rep.block
##   <chr>    <chr>      <dbl> <chr>      <chr>          <dbl>    <dbl>
##   <dbl>
## 1 PA      6/16/2...    15.3 18.44    V3             101      1
1
## 2 PA      6/16/2...    15.3 18.44    V3             101      1
1
## 3 PA      6/16/2...    15.3 18.44    V3             101      1
1
## 4 PA      6/16/2...    15.3 18.44    V3             101      1
1
## 5 PA      6/16/2...    15.3 18.44    V3             101      1
1
## 6 PA      6/16/2...    15.3 18.44    V3             101      1
1
## 7 PA      6/16/2...    15.3 18.44    V3             102      1
3
## 8 PA      6/16/2...    15.3 18.44    V3             102      1
3
## 9 PA      6/16/2...    15.3 18.44    V3             102      1
3
## 10 PA     6/16/2...    15.3 18.44    V3             102      1
3
## # i 5,271 more rows
## # i 9 more variables: row <dbl>, sample <dbl>, n.absent <chr>, n.partial
```

```

<chr>,
## #   n.predated <chr>, d.absent <chr>, d.partial <chr>, d.predated <chr>,
## #   to.predated <chr>

clean21 <- s21 %>%
  mutate(year = '2021') %>%
  dplyr::select(location, year, growth_stage, plot_id, rep.block, treatment,
to.predated) %>%
  mutate(to.predated = as.double(to.predated)) %>%
  dplyr::rename(block = rep.block) %>%
  group_by(location, year, growth_stage, plot_id, block, treatment) %>%
  # dplyr::summarise(total = sum(to.predated)) %>%
  na.omit() %>%
  mutate(treatment = case_when(
    treatment == '33' ~ '3',
    .default = as.factor(treatment))) %>%
  dplyr::filter(treatment != '6',
    treatment != '7',
    treatment != '8') %>%
  mutate_at(vars(1:6), as.factor)

```

2022 cleaning

s22

A tibble: 3,246 × 19

location date am.weather pm.weather growth_stage plotid block
treatment

<chr> <chr> <dbl> <chr> <chr> <dbl> <dbl>
<dbl>

1 PA 6/22/2022 22.7 26.1 V3 101 1

1

2 PA 6/22/2022 22.7 26.1 V3 101 1

1

3 PA 6/22/2022 22.7 26.1 V3 101 1

1

4 PA 6/22/2022 22.7 26.1 V3 101 1

1

5 PA 6/22/2022 22.7 26.1 V3 101 1

1

6 PA 6/22/2022 22.7 26.1 V3 101 1

1

7 PA 6/22/2022 22.7 26.1 V3 102 1

3

8 PA 6/22/2022 22.7 26.1 V3 102 1

3

9 PA 6/22/2022 22.7 26.1 V3 102 1

3

10 PA 6/22/2022 22.7 26.1 V3 102 1

3

```

## # i 3,236 more rows
## # i 11 more variables: row <dbl>, sample <dbl>, am.absent <chr>,
## #   am.partial <chr>, am.predators <chr>, pm.absent <chr>, pm.partial
## #   <chr>, pm.predators <chr>, to.predated <dbl>, n.predated <dbl>, d.predated
## #   <dbl>

unique(s22$treatment)

## [1] 1 3 2 4

unique(s22$growth_stage)

## [1] "V3" "V5" "R3" "R2"

clean22 <- s22 %>%
  mutate(year = '2022') %>%
  dplyr::select(location, year, growth_stage, plotid, block, treatment,
to.predated) %>%
  dplyr::rename(plot_id = plotid) %>%
  mutate(growth_stage = case_when(growth_stage == 'R2' ~ 'R3',
                                .default = as.character(growth_stage))) %>%
  dplyr::group_by(location, year, growth_stage, plot_id, block, treatment)
%>%
  # dplyr::summarise(total = sum(to.predated)) %>%
  mutate_at(vars(1:6), as.factor)

# 2023 cleaning

clean23 <- s23 %>%
  mutate(year = '2023') %>%
  relocate(am.partial, am.absent, pm.partial, pm.absent) %>%
  mutate_at(vars(1:4), as.double) %>%
  mutate(to.predated = if_else(am.partial | am.absent | pm.partial |
pm.absent == 1, 1, 0)) %>%
  relocate(to.predated) %>%
  mutate(growth_stage = case_when((location == 'NC' & date == '7/20/2023') ~
'R3',
                                .default = as.character(growth_stage))) %>%
  dplyr::select(location, year, growth_stage, plotid, block, treatmetn,
to.predated) %>%
  dplyr::rename(plot_id = plotid,
                treatment = treatmetn) %>%
  group_by(location, year, growth_stage, plot_id, block, treatment) %>%
  na.omit() %>%
  filter(treatment != 5) %>%
  mutate_at(vars(1:6), as.factor)

# and in the darkness, bind them

```

```
sent <- rbind(clean21, clean22, clean23)
as_tibble(sent)

## # A tibble: 12,432 × 7
##   location year growth_stage plot_id block treatment to.predated
##   <fct>    <fct> <fct>      <fct>  <fct> <fct>      <dbl>
## 1 PA      2021 V3         101    1    1          0
## 2 PA      2021 V3         101    1    1          0
## 3 PA      2021 V3         101    1    1          0
## 4 PA      2021 V3         101    1    1          0
## 5 PA      2021 V3         101    1    1          0
## 6 PA      2021 V3         101    1    1          0
## 7 PA      2021 V3         102    1    3          1
## 8 PA      2021 V3         102    1    3          1
## 9 PA      2021 V3         102    1    3          0
## 10 PA     2021 V3         102    1    3          1
## # i 12,422 more rows

dim(sent)

## [1] 12432      7
```

- 2) Prepare a short description of your data set (source if published paper exists), especially providing the study design, sample size, and variables of primary interest. If there is random sampling, note the population sampled from. If there is random assignment, note how and for which variable(s).

Data: Sentinel Prey assessment of arthropod-predator activity in corn fields.

These data come from the Precision Sustainable Agriculture effort through the USDA. I am the lead on the entomology component of this project and responsible for analyzing this three year data set which spans multiple states. This effort began during my Master's degree, but I only analyzed Pennsylvania data for my thesis.

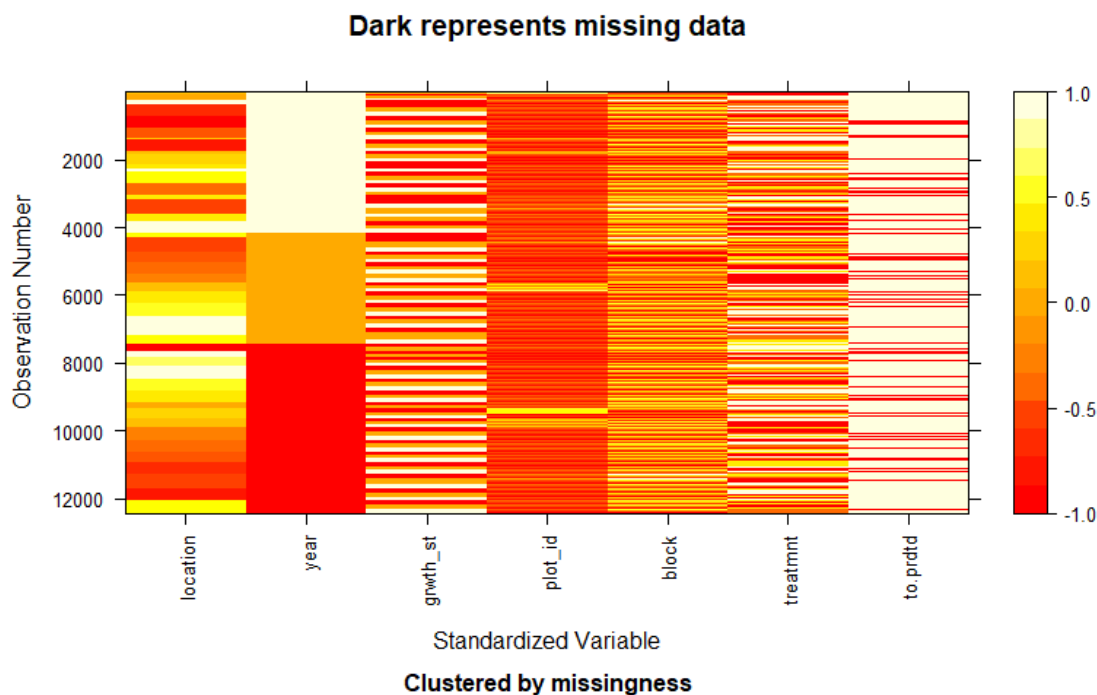
Study design: **Treatments** = 4; No cover crop, early-terminated cover crop, late-terminated cover crop, planting green **Plots** = 20; 5 blocks composed of 4 plots each = 20 plots / study site / year **Years** = 3 (2021,2022,2023) **Locations** = This project comprises 16 states. Not all states collected sentinel prey data every year. Each site year was in a different field. **Effort** = Data were collected at three corn growth stages / year (V3,V5,R3). **Sample** = 6 sentinel prey traps were placed in each plot = 120 samples collected / growth stage. Total sample effort per state per season = 360 samples.

Variables: **Response** = Total level of predation. This is a binomial of 6 traps/ counts per plot. Pseudoreplication is account for in the random term. **Explanatory** = Crop growth stage (timing, three levels) and treatment (four levels). I am not interested in the fixed effects of location. **Random effects** = Plot in block in location, in year. I want to account for pseduoreplication and all of the site/year combinations.

Plots were randomly assigned to each block. Field sites were as random as they could be at each respective research station. Sentinel prey traps were placed between pre-determined rows and at specific length intervals within each plot to maintain consistency.

- 3) Make a `missing_data.frame` plot of your data set and explain any missing values indicated:

```
library(mi)
# make an object of the missing df and then present the image
tdf <- missing_data.frame(data.frame(sent))
image(tdf)
```



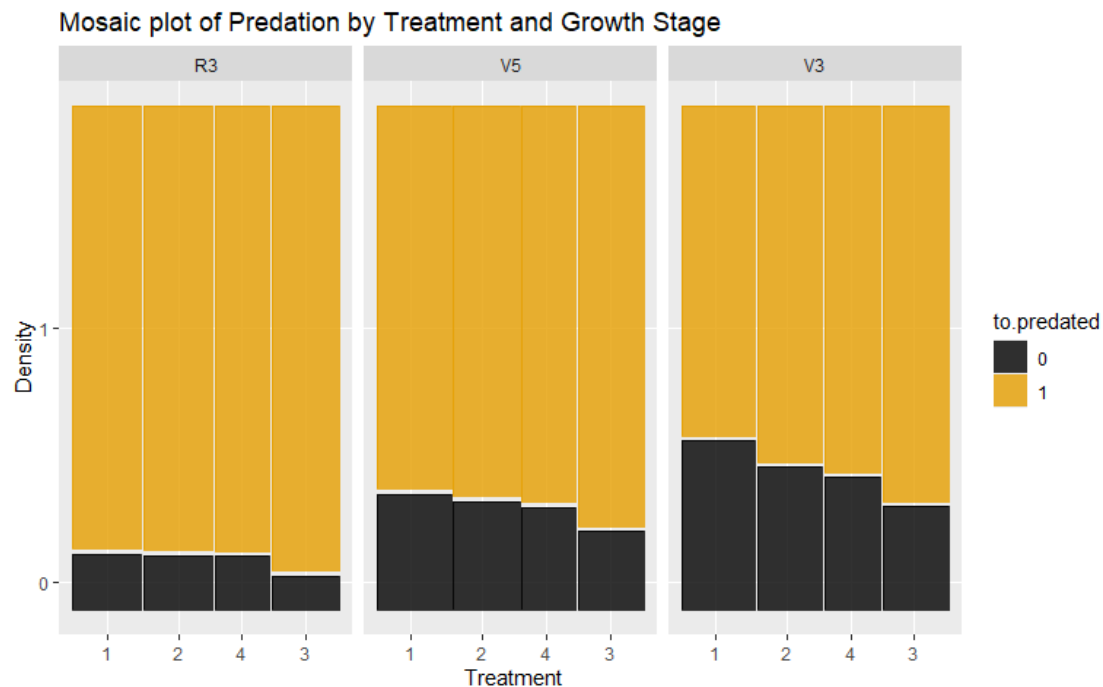
- 4) Discuss any other use in classes or theses for the data - either that you have used it for or are currently working on for future submissions.

I am working on this for a publication. There is no published paper yet. None of the code from that is used here. This analysis is for all of the states combined, but in the future, I plan to run each state individually with their three years of data. I suspect results to differ based on some regional grouping (e.g., growing degree days, growth region, etc.), but am yet to decide what I will use. For now, I am mainly interested in the treatment and growth stage effects on the whole data set.

- 5) Provide at least one display of the data, focusing on the response of interest versus a predictor. If you have multiple predictors, try to plot the response versus those too.

```
sent %>%
  ggplot() +
  geom_mosaic(aes(x = product(treatment), fill = to.predated))+
  facet_wrap(~growth_stage)+
  scale_fill_colorblind()+
```

```
labs(title = 'Mosaic plot of Predation by Treatment and Growth Stage',
     y = 'Density',
     x = 'Treatment')
```



- 6) Provide an initial model you hope to fit (does not need to be fit). If you fit a model, add a model summary and effects plot.

```
sent
## # A tibble: 12,432 × 7
## # Groups:   location, year, growth_stage, plot_id, block, treatment
## [2,091]
##   location year growth_stage plot_id block treatment to.predated
##   <fct>    <fct> <fct>      <fct>  <fct> <fct>      <dbl>
## 1 PA      2021 V3         101    1    1          0
## 2 PA      2021 V3         101    1    1          0
## 3 PA      2021 V3         101    1    1          0
## 4 PA      2021 V3         101    1    1          0
## 5 PA      2021 V3         101    1    1          0
## 6 PA      2021 V3         101    1    1          0
## 7 PA      2021 V3         102    1    3          1
## 8 PA      2021 V3         102    1    3          1
## 9 PA      2021 V3         102    1    3          0
## 10 PA     2021 V3         102    1    3          1
## # i 12,422 more rows

m2 <- glmer(to.predated ~ treatment*growth_stage +
(1|year/location/block/plot_id) , family = binomial, data = sent)
summary(m2)
```

```

## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula:
## to.predated ~ treatment * growth_stage + (1 | year/location/block/plot_id)
## Data: sent
##
##      AIC      BIC   logLik deviance df.resid
##   9135.0   9253.8  -4551.5   9103.0   12416
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -11.4203   0.0801   0.2125   0.4224   2.9757
##
## Random effects:
## Groups                                Name          Variance Std.Dev.
## plot_id:(block:(location:year)) (Intercept) 2.611e-01 0.511004
## block:(location:year)          (Intercept) 1.701e-01 0.412483
## location:year                  (Intercept) 2.689e+00 1.639802
## year                          (Intercept) 1.545e-05 0.003931
## Number of obs: 12432, groups:
## plot_id:(block:(location:year)), 744; block:(location:year), 191;
## location:year, 39; year, 3
##
## Fixed effects:
##
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      3.03307    0.29061  10.437 < 2e-16
## treatment2        0.03549    0.16020   0.222 0.824674
## treatment4        0.22142    0.17252   1.283 0.199352
## treatment3        0.62055    0.17515   3.543 0.000396
## growth_stageV5    -1.15589    0.13170  -8.776 < 2e-16
## growth_stageV3    -1.92785    0.13398 -14.389 < 2e-16
## treatment2:growth_stageV5  0.03110    0.18796   0.165 0.868581
## treatment4:growth_stageV5  0.08776    0.20154   0.435 0.663242
## treatment3:growth_stageV5 -0.02574    0.20392  -0.126 0.899541
## treatment2:growth_stageV3  0.29511    0.18920   1.560 0.118818
## treatment4:growth_stageV3  0.44923    0.20103   2.235 0.025439
## treatment3:growth_stageV3  0.31249    0.20397   1.532 0.125508
##
## Correlation of Fixed Effects:
##      (Intr) trtmn2 trtmn4 trtmn3 grw_V5 grw_V3 t2:_V5 t4:_V5 t3:_V5
## treatment2 -0.272
## treatment4 -0.248 0.458
## treatment3 -0.246 0.453 0.421
## grwth_stgV5 -0.293 0.511 0.472 0.466
## grwth_stgV3 -0.300 0.511 0.474 0.464 0.647
## trtmnt2:_V5 0.199 -0.724 -0.332 -0.327 -0.693 -0.446
## trtmnt4:_V5 0.185 -0.334 -0.719 -0.305 -0.646 -0.415 0.453
## trtmnt3:_V5 0.184 -0.330 -0.306 -0.744 -0.639 -0.412 0.448 0.417
## trtmnt2:_V3 0.203 -0.727 -0.334 -0.330 -0.451 -0.692 0.622 0.294 0.290

```

```
## trtmnt4:_V3  0.192 -0.339 -0.728 -0.310 -0.424 -0.655  0.297  0.623  0.273
## trtmnt3:_V3  0.191 -0.335 -0.309 -0.747 -0.420 -0.645  0.292  0.272  0.645
##          t2:_V3 t4:_V3
## treatment2
## treatment4
## treatment3
## grwth_stgV5
## grwth_stgV3
## trtmnt2:_V5
## trtmnt4:_V5
## trtmnt3:_V5
## trtmnt2:_V3
## trtmnt4:_V3  0.459
## trtmnt3:_V3  0.454  0.427
## optimizer (Nelder_Mead) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 0.0296809 (tol = 0.002,
component 1)
```

Anova(m2)

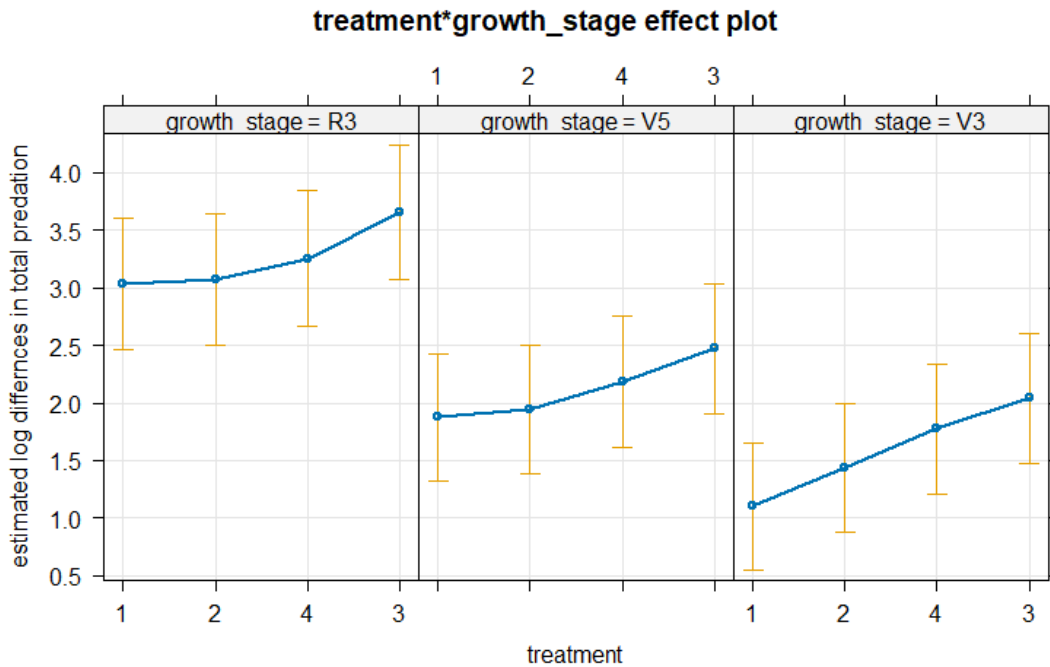
```
## Analysis of Deviance Table (Type II Wald chisquare tests)
```

```
##
```

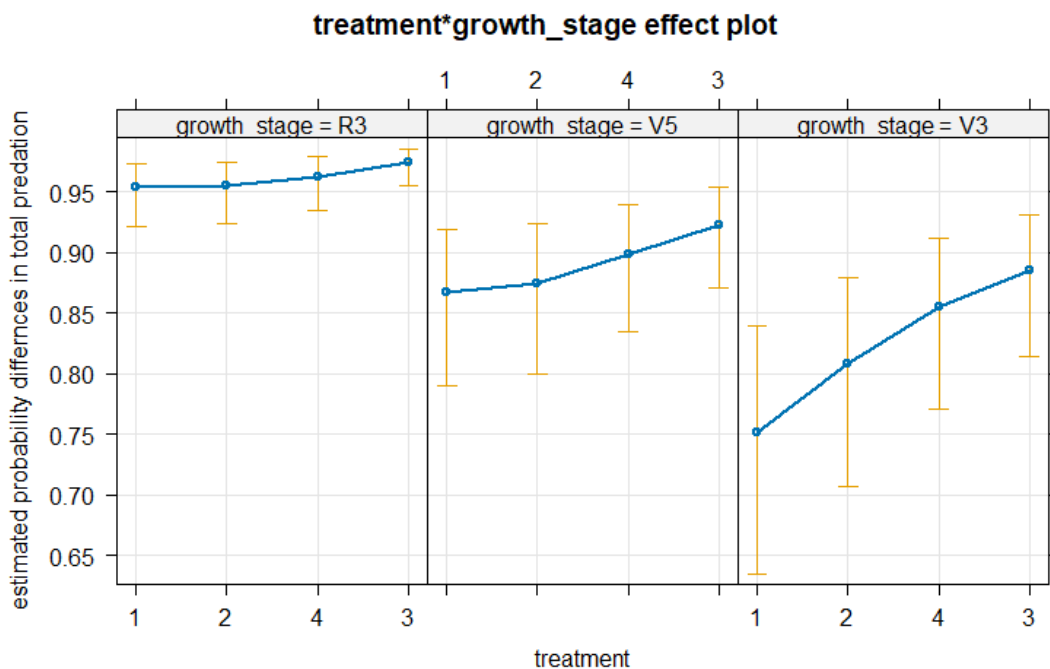
```
## Response: to.predated
```

```
##          Chisq Df Pr(>Chisq)
## treatment          60.9286  3 3.722e-13
## growth_stage       526.0579  2 < 2.2e-16
## treatment:growth_stage   8.2544  6      0.22
```

```
plot(allEffects(m2), type = 'link',ylab = 'estimated log differnces in total
predation', grid = T)
```

```
plot(allEffects(m2), type = 'response', ylab = 'estimated probability
differences in total predation', grid = T)
```



* There is weak evidence against the null of no interaction between treatment and growth stage $\chi^2(6) = 8.25$, $p = 0.22$, after controlling for the random effect of plot nested in block nested in location nested in year, and will remove the interaction term from the model.

Model refinement

```

m3 <- glmer(to.predated ~ treatment+growth_stage +
(1|year/location/block/plot_id) , family = binomial, data = sent)
summary(m3)

## Generalized linear mixed model fit by maximum likelihood (Laplace
## Approximation) [glmerMod]
## Family: binomial ( logit )
## Formula:
## to.predated ~ treatment + growth_stage + (1 | year/location/block/plot_id)
## Data: sent
##
##      AIC      BIC   logLik deviance df.resid
##  9131.1   9205.4  -4555.5   9111.1    12422
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -11.4488   0.0811   0.2124   0.4229   2.9455
##
## Random effects:
## Groups                                Name      Variance Std.Dev.
## plot_id:(block:(location:year)) (Intercept) 2.600e-01 0.50988
## block:(location:year)          (Intercept) 1.700e-01 0.41236
## location:year                  (Intercept) 2.672e+00 1.63473
## year                          (Intercept) 5.569e-06 0.00236
## Number of obs: 12432, groups:
## plot_id:(block:(location:year)), 744; block:(location:year), 191;
## location:year, 39; year, 3
##
## Fixed effects:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    2.92687    0.27926  10.481 < 2e-16
## treatment2      0.15828    0.09462   1.673  0.0944
## treatment4      0.42821    0.10271   4.169 3.06e-05
## treatment3      0.72957    0.09942   7.338 2.16e-13
## growth_stageV5 -1.13027    0.07256 -15.578 < 2e-16
## growth_stageV3 -1.68191    0.07334 -22.932 < 2e-16
##
## Correlation of Fixed Effects:
##              (Intr) trtmn2 trtmn4 trtmn3 grw_V5
## treatment2  -0.161
## treatment4  -0.138  0.451
## treatment3  -0.141  0.468  0.436
## grwth_stgV5 -0.172 -0.003 -0.015 -0.026
## grwth_stgV3 -0.176 -0.007 -0.024 -0.041  0.625
## optimizer (Nelder_Mead) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 0.0104819 (tol = 0.002,
## component 1)

Anova(m3)

```

```
## Analysis of Deviance Table (Type II Wald chisquare tests)
```

```
##
```

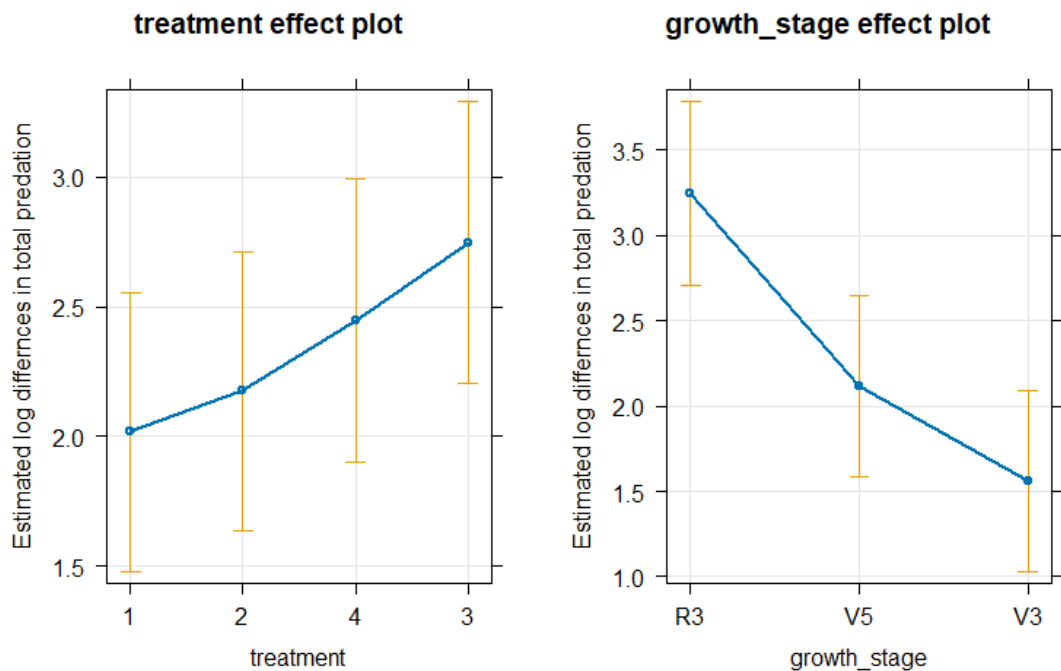
```
## Response: to.predated
```

```
##           Chisq Df Pr(>Chisq)
```

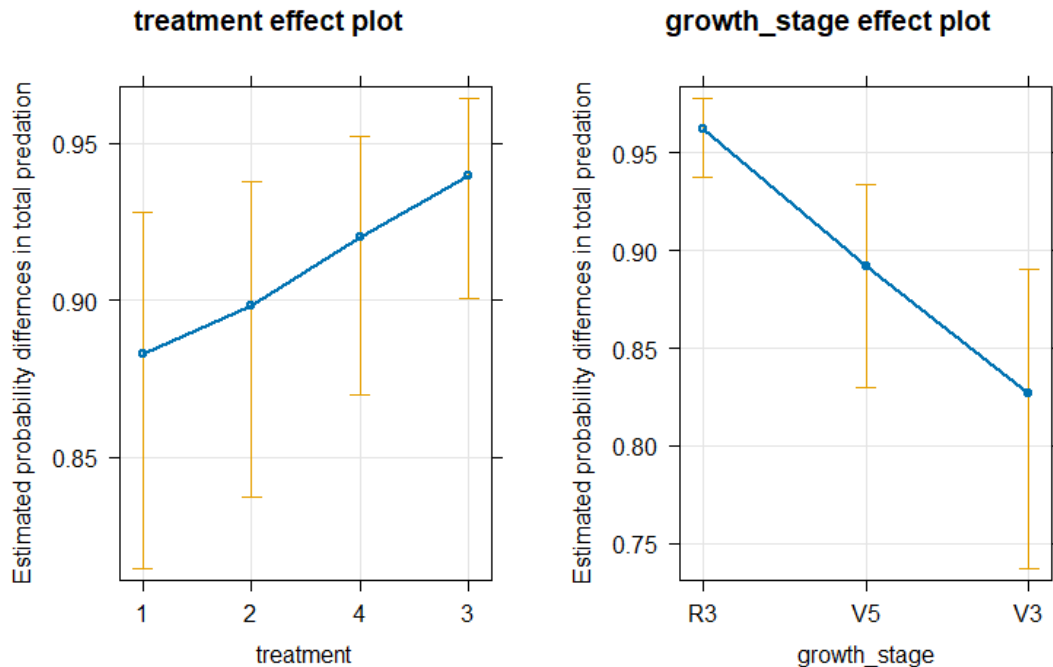
```
## treatment    60.988  3  3.615e-13
```

```
## growth_stage 528.427  2  < 2.2e-16
```

```
plot(allEffects(m3), type = 'link', ylab = 'Estimated log differences in total  
predation', grid = T)
```



```
plot(allEffects(m3), type = 'response', ylab = 'Estimated probability  
differences in total predation', grid = T)
```



```
cld(emmeans(m3, ~treatment), Letters = letters)
```

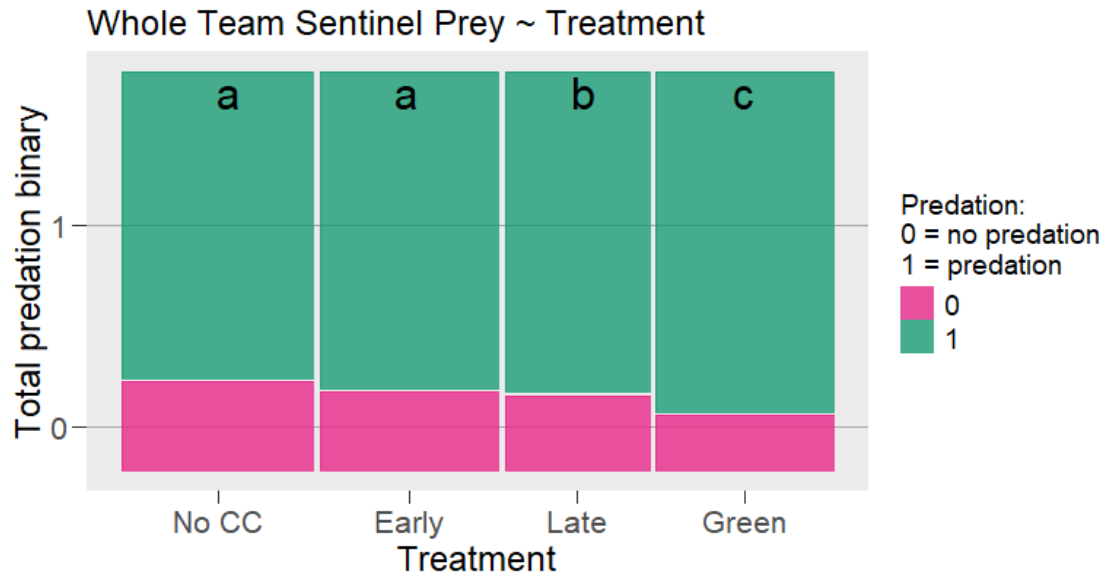
```
## treatment emmean SE df asymp.LCL asymp.UCL .group
## 1 1.99 0.274 Inf 1.45 2.53 a
## 2 2.15 0.275 Inf 1.61 2.69 a
## 4 2.42 0.279 Inf 1.87 2.96 b
## 3 2.72 0.277 Inf 2.18 3.26 c
##
## Results are averaged over the levels of: growth_stage
## Results are given on the logit (not the response) scale.
## Confidence level used: 0.95
## Results are given on the log odds ratio (not the response) scale.
## P value adjustment: tukey method for comparing a family of 4 estimates
## significance level used: alpha = 0.05
## NOTE: If two or more means share the same grouping symbol,
## then we cannot show them to be different.
## But we also did not show them to be the same.
```

```
cld(emmeans(m3, ~growth_stage), Letters = letters)
```

```
## growth_stage emmean SE df asymp.LCL asymp.UCL .group
## V3 1.57 0.271 Inf 1.04 2.10 a
## V5 2.13 0.271 Inf 1.59 2.66 b
## R3 3.26 0.275 Inf 2.72 3.79 c
##
## Results are averaged over the levels of: treatment
## Results are given on the logit (not the response) scale.
## Confidence level used: 0.95
## Results are given on the log odds ratio (not the response) scale.
```

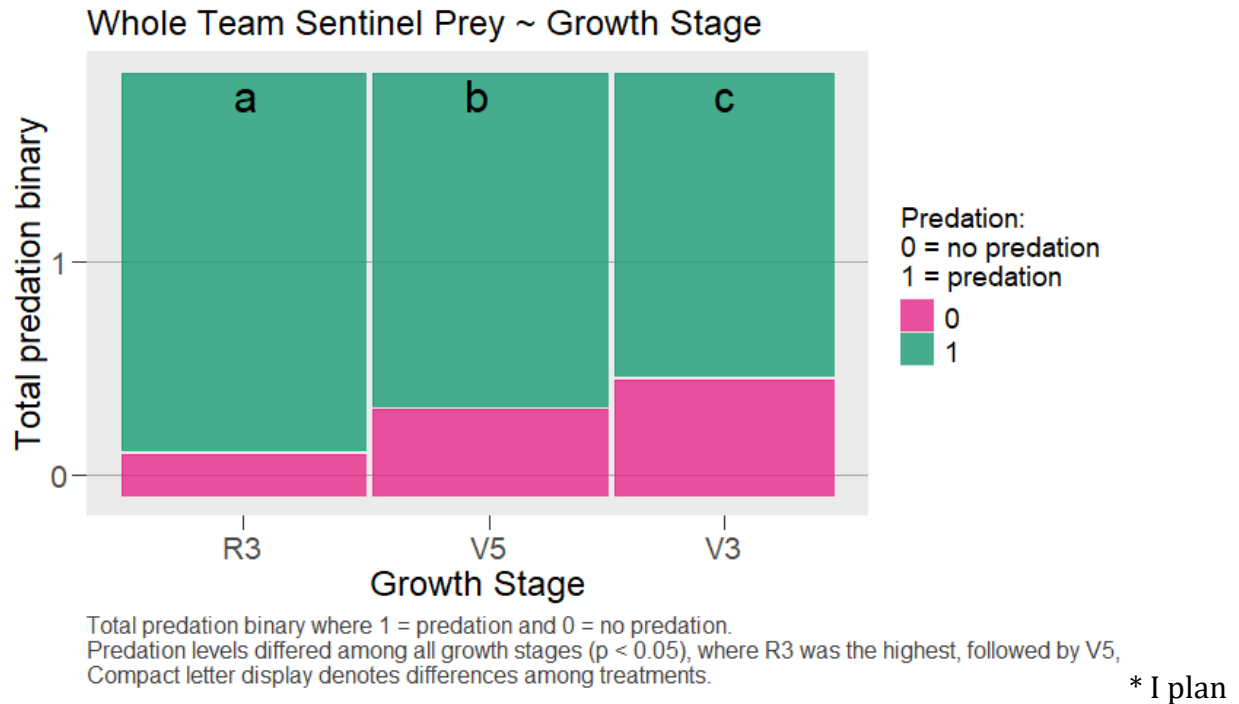
```
## P value adjustment: tukey method for comparing a family of 3 estimates
## significance level used: alpha = 0.05
## NOTE: If two or more means share the same grouping symbol,
##       then we cannot show them to be different.
##       But we also did not show them to be the same.
```

```
sent %>%
  ggplot()+
  geom_mosaic(aes(x = product(treatment), fill = to.predated))+
  scale_fill_manual(values = c("#E7298A", "#1B9E77"))+
  scale_x_productlist(labels=c("No CC", "Early", "Late", "Green"))+
  labs(title = 'Whole Team Sentinel Prey ~ Treatment',
       x = 'Treatment',
       y = 'Total predation binary',
       caption = 'Total predation binary where 1 = predation and 0 = no
predation.\nPredation levels differed between the no-cover and late-
terminated treatment\nand the no-cover and planting-green treatment (p <
0.05).\nCompact letter display denotes differences among treatments.')+
  guides(fill = guide_legend(title = 'Predation:\n0 = no predation\n1 =
predation'))+
  theme(legend.title = element_text(size = 14),
        legend.text = element_text(size = 14),
        axis.text.x = element_text(size=16),
        axis.text.y = element_text(size = 16),
        axis.title = element_text(size = 18),
        plot.title = element_text(size = 18),
        plot.subtitle = element_text(size = 12),
        panel.grid.major.y = element_line(color = "darkgrey"),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        strip.text = element_text(size = 12),
        plot.caption = element_text(hjust = 0, size = 12, color = "grey25"),
        axis.ticks.length=unit(.25, "cm"))+
  annotate('text', x = 0.15, y = 0.95, label = 'a', size = 8)+
  annotate('text', x = .4, y = 0.95, label = 'a', size = 8)+
  annotate('text', x = .65, y = 0.95, label = 'b', size = 8)+
  annotate('text', x = .875, y = 0.95, label = 'c', size = 8)
```



Total predation binary where 1 = predation and 0 = no predation.
Predation levels differed between the no-cover and late-terminated treatment and the no-cover and planting-green treatment ($p < 0.05$).
Compact letter display denotes differences among treatments.

```
sent %>%
  ggplot()+
  geom_mosaic(aes(x = product(growth_stage), fill = to.predated))+
  scale_fill_manual(values = c("#E7298A", "#1B9E77"))+
  labs(title = 'Whole Team Sentinel Prey ~ Growth Stage',
       x = 'Growth Stage',
       y = 'Total predation binary',
       caption = 'Total predation binary where 1 = predation and 0 = no
predation.\nPredation levels differed among all growth stages (p < 0.05),
where R3 was the highest, followed by V5, and then V3.\nCompact letter
display denotes differences among treatments.')+
  guides(fill = guide_legend(title = 'Predation:\n0 = no predation\n1 =
predation'))+
  theme(legend.title = element_text(size = 14),
        legend.text = element_text(size = 14),
        axis.text.x = element_text(size=16),
        axis.text.y = element_text(size = 16),
        axis.title = element_text(size = 18),
        plot.title = element_text(size = 18),
        plot.subtitle = element_text(size = 12),
        panel.grid.major.y = element_line(color = "darkgrey"),
        panel.grid.major.x = element_blank(),
        panel.grid.minor = element_blank(),
        strip.text = element_text(size = 12),
        plot.caption = element_text(hjust = 0, size = 12, color = "grey25"),
        axis.ticks.length=unit(.25, "cm"))+
  annotate('text', x = 0.175, y = 0.95, label = 'a', size = 8)+
  annotate('text', x = .5, y = 0.95, label = 'b', size = 8)+
  annotate('text', x = .85, y = 0.95, label = 'c', size = 8)
```



to make additional plots with the proportion of predation.

- Used https://haleyjeppson.github.io/ggmosaic/reference/geom_mosaic.html for the `scale_x_productlist` function.

How to I reorder my x axis? I was not able to find the syntax online. I tried `limits`, but it seems `limits` in the `scale_x_productlist` does not work like it does in `scale_x_discrete`. I would like it to do V3, V5, R3. I tried functions from the `forcats` package, but it seems `ggmosaic` is overriding those or is not compatible with them in the same line.

- 7) Start to work on a Table 1 that summarizes variables of interest, possibly by groups of interest. At a minimum, summarize the response variable, by a grouping variable if one exists.

```
# table as a proportion
sent %>%
  group_by(location, treatment, growth_stage) %>%
  summary()
```

##	location	year	growth_stage	plot_id	block	treatment
##	IL	:1296	2021:5004	R3:4334	203 : 608	1:2681 1:3439
##	OH	:1173	2022:3246	V5:4196	103 : 607	2:2491 2:3203
##	VT	:1080	2023:4182	V3:3902	303 : 607	3:2493 4:2605
##	PA	:1074		401 : 603	4:2514	3:3185
##	KY	:1054		102 : 591	5:1969	
##	KS	: 848		202 : 589	6: 284	
##	(Other):5907			(Other):8827		
##	to.predated					
##	Min.	:0.0000				

```
## 1st Qu.:1.0000
## Median :1.0000
## Mean   :0.8103
## 3rd Qu.:1.0000
## Max.   :1.0000
##

sent %>%
  group_by(growth_stage) %>%
  summary()

##      location      year growth_stage  plot_id  block  treatment
## IL      :1296    2021:5004    R3:4334    203    : 608    1:2681    1:3439
## OH      :1173    2022:3246    V5:4196    103    : 607    2:2491    2:3203
## VT      :1080    2023:4182    V3:3902    303    : 607    3:2493    4:2605
## PA      :1074                                401    : 603    4:2514    3:3185
## KY      :1054                                102    : 591    5:1969
## KS      : 848                                202    : 589    6: 284
## (Other):5907                                (Other):8827
## to.predated
## Min.     :0.0000
## 1st Qu.  :1.0000
## Median   :1.0000
## Mean     :0.8103
## 3rd Qu.  :1.0000
## Max.     :1.0000
##

tally(treatment ~ growth_stage, data = sent)

##      growth_stage
## treatment  R3   V5   V3
##      1 1146 1195 1098
##      2 1149 1068  986
##      4  898  871  836
##      3 1141 1062  982
```

8) Research question:

How do sentinel prey rates (1,0) change among treatments and growth stages and is there an interaction between treatment and growth stage?

9) SOI

G: The locations chosen for this experiment were not randomly selected (Universities volunteered to join the project), and thus the results would not generalize beyond the locations present. Additionally, the results only apply to years 2021-2023 in a corn monoculture following a cereal-rye cover crop. Data were collected at three, non-random time points throughout the season, and we cannot generalize beyond those time points.

C: The treatments and the trap location were randomly assigned in the field and thus, we can make causal inference on the predation levels within different treatments and parts of the field.