Texas 2022 Analysis

Kirsten Sheehy

2024-05-16

Overview

The following script cleans and analyzes data from Texas 2022. Fish were collected by Kirsten Sheehy and Jon Aguiñaga. Behavioral data was extracted from videos by Nishika Raghavan. Fish length was extracted from photos by Tommy (INSERT LAST NAME). Parasite data were collected by Dr. Jessica Stephenson's lab.

Packages to Load

```
library(dplyr)
library(readr)
library(tidyr)
library(tibble)
library(lubridate)
library(tidyverse)
library(ggplot2)
library(lme4)
library(pscl)
```

```
## Warning: package 'pscl' was built under R version 4.3.2
```

```
library(MASS)
library(lmtest)
library(here)
library(knitr)
```

Raw Data

```
parasite_data <- read.csv(here("data", "copy_RAW_parasite_data_20230428.csv"))
length_data <- read.csv(here("data", "copy_Fish Length Data.csv"))
boris_data <- read.csv(here("data", "copy_RAW_Texas_BORISdata_20240515.csv"))
ID_data <- read.csv(here("data", "copy_RAW_trial_ID_data_completeonly_20240220.csv"))
```

Tidy Data

Parasite Data

```
# rename columns to be consistent across data sets
parasite_data <- parasite_data %>% dplyr::rename(
  fish.ID = fish.id,
  site.ID = collection.site
)
```

```r
# change collection.date and dissection.date to a date format (YYYY-MM-DD)
parasite_data$collection.date <- as.Date(parasite_data$collection.date,
  format = "%m/%d/%y"
)
parasite_data$dissection.date <- as.Date(parasite_data$dissection.date,
  format = "%m/%d/%y"
)

# change site names from abbreviation (WES, BR) to full (Weslaco, Brownsville)
parasite_data$site.ID <- gsub("WES", "Weslaco", parasite_data$site.ID)
parasite_data$site.ID <- gsub("BR-OP", "Brownsville", parasite_data$site.ID)

# note: the 'OP' in Brownsville stands for 'overpass'. We explored several sites
# in Brownsville, but only used the ones from the overpass for this study, so I
# simplified the name to just 'Brownsville'.
```

Length Data

NOTE: I need to go back and check Tommy's labeling and a few measurements to be sure they're accurate.

```r
# rename columns to be consistent across data sets
length_data <- length_data %>% dplyr::rename(
  file.name = file_name,
  date.image = date_image,
  site.ID = site_ID,
  fish.ID = fish_ID
)

# remove spaces in fish.IDs
length_data$fish.ID <- gsub(" ", "", length_data$fish.ID)
```

Boris Data

First, I tidy the raw data. I rename columns and remove unnecessary ones.

```r
# remove unnecessary columns (largely meta data and unused features in BORIS)
boris_data <- boris_data %>% dplyr::select(
  -Observation.date, # this is just the day processed in BORIS
  -Description,
  -FPS,
  -Behavioral.category,
  -Modifiers,
  -Comment.start,
  -Comment.stop
)

# rename columns (to match up across data)
boris_data <- boris_data %>% dplyr::rename(
  pool = Subject,
  trial.length = Total.length,
  start = Start..s.,
  stop = Stop..s.,
  duration = Duration..s.
)
```

```r
# split Media.file into columns to extract file name (could also use
# Observation.id, but figured this would help avoid typos made in Boris)
boris_data <- boris_data %>% tidyr::separate_wider_delim(Media.file,
  delim = "/",
  names = c(
    "file1",
    "file2",
    "file3",
    "file4",
    "file5",
    "video.ID"
  ),
  too_few = "align_end"
)

# remove the excess filepath columns
boris_data <- boris_data %>% dplyr::select(
  -file1,
  -file2,
  -file3,
  -file4,
  -file5
)

# video.id (from the file path split above) is the file name of the recording.
# It decomposes into the site ID, trial number, batch, and date recorded. The
# following code duplicates the column, then splits the information in video.id
# into separate columns.
boris_data$video.ID.split <- boris_data$video.ID
boris_data <- boris_data %>% tidyr::separate_wider_delim(video.ID.split,
  delim = "_",
  names = c(
    "site.ID",
    "trial.ID",
    "batch.ID",
    "trial.date"
  )
)

# remove the file type from the trial.date column
boris_data$trial.date <- gsub(".mov", "", boris_data$trial.date)

# change trial.date from (YYYYMMDD) to a date (YYYY-MM-DD)
boris_data$trial.date <- as.Date(boris_data$trial.date, format = "%Y%m%d")

# remove 'trial' from the data entries for trial.ID
boris_data$trial.ID <- gsub("trial", "", boris_data$trial.ID)
boris_data$trial.ID <- gsub("trail", "", boris_data$trial.ID) # had to find a few where this was a typo

# remove 'pool' from data in pool column
boris_data$pool <- gsub("Pool ", "", boris_data$pool)

# change site ID from abbreviations to full name
```

```r
# note: doing them in this order is important
boris_data$site.ID <- gsub("Wes", "Weslaco", boris_data$site.ID)
boris_data$site.ID <- gsub("WES", "Weslaco", boris_data$site.ID)
boris_data$site.ID <- gsub("BR1", "Brownsville", boris_data$site.ID)
boris_data$site.ID <- gsub("BR2", "Brownsville", boris_data$site.ID)
boris_data$site.ID <- gsub("BR", "Brownsville", boris_data$site.ID)
# note: there are three entry types for Brownsville: BR, BR1, and BR2
# need to revisit lab notebook to confirm, but I believe BR1 and BR2
# are the two sides of the garage (i.e. the two cameras)
```

Now that the columns are all formatted correctly, I need to pull out the behaviors from the Behavior column into their own, separate columns.

```r
# start by duplicating the 'Behavior' column twice. This will be used to extract start and stop times o
boris_data <- boris_data %>%
  dplyr::mutate(behavior.start = Behavior)

boris_data <- boris_data %>%
  dplyr::mutate(behavior.stop = Behavior)

# then pivot_wider with names from behavior.start and values from start
boris_data_wide <- boris_data %>%
  tidyr::pivot_wider(
    names_from = behavior.start,
    values_from = start,
    names_prefix = "start."
  )

# do the same with stop
boris_data_wide <- boris_data_wide %>%
  tidyr::pivot_wider(
    names_from = behavior.stop,
    values_from = stop,
    names_prefix = "stop."
  )

# now, I need to get the duration of each behavior using the start and stop times
boris_data_wide <- boris_data_wide %>%
  tidyr::pivot_wider(
    names_from = Behavior,
    values_from = duration,
    names_prefix = "duration."
  )

# I'll remove stop_Startle and duration_Startle because these are 'points' not 'states' and do not have
boris_data_wide <- boris_data_wide %>% dplyr::select(
  -stop.Startle,
  -duration.Startle
)
```

Now, I need to join the ID_data and boris_data_wide datasets.

```r
# I need a column in both ID_data and boris_data_wide to join by
# I'll create a new column that merges the file name (which already includes
# site, trial, and batch) with pool # for both data sets
```

```
boris_data_wide$merge.ID <- paste(
  boris_data_wide$video.ID,
  boris_data_wide$pool
)

ID_data$merge.ID <- paste(
  ID_data$video.ID,
  ID_data$pool
)

boris_data_merge <- boris_data_wide %>%
  left_join(ID_data, by = "merge.ID")
```

Now we tidy the merged data.

```
# remove duplicate columns
boris_data_merge <- boris_data_merge %>% dplyr::select(
  -merge.ID,
  -video.ID.y,
  -pool.y,
  -site.ID.y,
  -trial.ID.y,
  -batch.ID.y,
  -trial.date.y
)

# rename columns to get rid of .x and .y appendages
boris_data_merge <- boris_data_merge %>% dplyr::rename(
  video.ID = video.ID.x,
  pool = pool.x,
  site.ID = site.ID.x,
  trial.ID = trial.ID.x,
  batch.ID = batch.ID.x,
  trial.date = trial.date.x
)

# add column for species from fish.ID
boris_data_merge <- boris_data_merge %>%
  mutate(species = fish.ID)

boris_data_merge <- boris_data_merge %>%
  separate_wider_delim(species,
    delim = "-",
    names = c(
      "species",
      "junk.num"
    )
  )

boris_data_merge <- boris_data_merge %>% dplyr::select(-junk.num)

# filling the 'start.startle' column based on fish and trial ID
boris_data_merge <- boris_data_merge %>%
  group_by(fish.ID, trial.ID) %>%
```

```
  fill(start.Startle, .direction = "downup")
```

Now, I need to standardize the trial times. When Nishika and I were observing, we sometimes recorded behaviors for longer than the prescribed 15 minutes. The following code finds the earliest behavior observation (either start.hiding or start.open) and then cuts off any observations after 15 minutes.

```
# new columns with the earliest open and hiding value per fish per trial
boris_data_merge <- boris_data_merge %>%
  group_by(fish.ID, trial.ID) %>%
  mutate(
    earliest.open = min(start.Open, na.rm = TRUE),
    earliest.hiding = min(start.Hiding, na.rm = TRUE)
  )
```

```
## Warning: There were 33 warnings in `mutate()`.
## The first warning was:
## i In argument: `earliest.open = min(start.Open, na.rm = TRUE)`.
## i In group 15: `fish.ID = "PF-28"`, `trial.ID = "3"`.
## Caused by warning in `min()`:
## ! no non-missing arguments to min; returning Inf
## i Run `dplyr::last_dplyr_warnings()` to see the 32 remaining warnings.
```

```
# creates a trial cutoff time by taking the earliest behavior time (either open or closed)
# and adding 1200 seconds (20 minutes) to it
boris_data_merge <- boris_data_merge %>%
  group_by(fish.ID, trial.ID) %>%
  mutate(
    trial.end = pmin(earliest.open, earliest.hiding, na.rm = TRUE) + 1200
  )

# now, I need to remove all observations per fish per trial that exceed this cutoff time
boris_data_cutoff <- boris_data_merge %>%
  group_by(fish.ID, trial.ID) %>%
  filter(start.Open <= trial.end |
    start.Hiding <= trial.end)

# now, I need to create an 'end cap' value to replace any 'stop' behaviors
# basically, I need to close the observation (like in Boris)
# this also means I'll need to change the 'duration' columns, which are automatically
# exported from Boris.

# replace stop.Open with trial cutoff if higher than cutoff
boris_data_cutoff <- boris_data_cutoff %>%
  group_by(fish.ID, trial.ID) %>%
  mutate(stop.Open = if_else(stop.Open > trial.end, trial.end, stop.Open))

# replace stop.Open with trial cutoff if higher than cutoff
boris_data_cutoff <- boris_data_cutoff %>%
  group_by(fish.ID, trial.ID) %>%
  mutate(stop.Hiding = if_else(stop.Hiding > trial.end, trial.end, stop.Hiding))

# now, recalculate duration based on new end times
boris_data_cutoff <- boris_data_cutoff %>%
  group_by(fish.ID, trial.ID) %>%
  mutate(duration.Open = stop.Open - start.Open)
```

```
boris_data_cutoff <- boris_data_cutoff %>%
  group_by(fish.ID, trial.ID) %>%
  mutate(duration.Hiding = stop.Hiding - start.Hiding)
```

I know that I won't be using the third trial since most fish didn't get there, so I'm removing that data now.

```
# removing 3rd trial since most fish didn't get three
boris_data_cutoff <- boris_data_cutoff %>%
  filter(trial.ID == "1" |
    trial.ID == "2")
```

Now let's create some columns for summary data (e.g. total time hiding)

```
# time hiding per trial
total_hiding <- boris_data_cutoff %>%
  aggregate(
    duration.Hiding ~ fish.ID + trial.ID,
    sum
  )

total_hiding <- total_hiding %>%
  mutate(duplicate.fish.ID = fish.ID)

total_hiding <- total_hiding %>%
  separate_wider_delim(duplicate.fish.ID,
    delim = "-",
    names = c(
      "species",
      "junk.num"
    )
  )

total_hiding <- total_hiding %>%
  dplyr::select(-junk.num)

# time in open per trial
total_open <- boris_data_cutoff %>%
  aggregate(
    duration.Open ~ fish.ID + trial.ID,
    sum
  )

total_open <- total_open %>%
  mutate(duplicate.fish.ID = fish.ID)

total_open <- total_open %>%
  separate_wider_delim(duplicate.fish.ID,
    delim = "-",
    names = c(
      "species",
      "junk.num"
    )
  )

total_open <- total_open %>%
```

```
  dplyr::select(-junk.num)
```

Now, I want to merge total hiding and open per fish per trial.

```
# I'm going to create a column with both fish ID and trial to create a unique
# row for each fish/trial combination. I'll then use this to join the hiding
# and open datasets
total_open <- total_open %>%
  unite(fish.ID.trial, c(fish.ID, trial.ID))

total_hiding <- total_hiding %>%
  unite(fish.ID.trial, c(fish.ID, trial.ID))

# merge
total_hide_open <- total_hiding %>%
  left_join(total_open, by = "fish.ID.trial")

# get rid of duplicate columns
total_hide_open <- total_hide_open %>%
  dplyr::select(-species.y)

# separate fish ID and trial
total_hide_open <- total_hide_open %>%
  separate_wider_delim(fish.ID.trial,
    delim = "_",
    names = c(
      "fish.ID",
      "trial"
    )
  )

# and for my own sanity, renaming the species column
total_hide_open <- total_hide_open %>%
  rename(species = species.x)

# time hiding and open by trial
total_hide_open <- total_hide_open %>%
  mutate(site.ID = boris_data_cutoff$site.ID[match(fish.ID, boris_data_cutoff$fish.ID)])

# hiding and open by trial, joined with parasite and size data
total_hide_open <- total_hide_open %>%
  left_join(parasite_data, by = "fish.ID", relationship = "many-to-many")

total_hide_open <- total_hide_open %>%
  left_join(length_data, by = "fish.ID", relationship = "many-to-many")

total_hide_open <- total_hide_open %>%
  dplyr::select(
    -site.ID.y,
    -species.y,
    -site.ID,
    -species
  )
```

```r
# rename columns
total_hide_open <- total_hide_open %>% rename(
  species = species.x,
  site.ID = site.ID.x,
  total.parasites = totalpara
)
```

Inspect data

First, some 'dummy checks' to make sure the data make sense

```r
# fish.IDs
unique(ID_data$fish.ID) # 68 unique IDs
```

```
##  [1] "PF-08BR" "PL-25"   "PF-25B"  "PF-26"   "PF-27"   "PF-28"   "PF-30"
##  [8] "PF-31"   "PF-32"   "PF-33"   "PF-35"   "PF-36"   "PF-37"   "PF-38"
## [15] "PF-39"   "PF-40"   "PF-46"   "PF-60"   "PF-61"   "PF-62"   "PF-65"
## [22] "PL-02"   "PL-05"   "PL-07"   "PL-08"   "PL-09"   "PL-11"   "PL-12"
## [29] "PL-13"   "PL-15"   "PL-20"   "PL-21"   "PL-22"   "PL-26"   "PL-28"
## [36] "PL-30"   "PL-31"   "PL-32"   "PL-34"   "PL-35"   "PL-37"   "PL-38"
## [43] "PL-39"   "PL-41"   "PL-43"   "PL-44"   "PL-45"   "PL-46"   "PL-47"
## [50] "PL-48"   "PL-49"   "PL-50"   "PL-51"   "PL-52"   "PL-53"   "PL-54"
## [57] "PL-55"   "PL-56"   "PL-57"   "PL-58"   "PL-59"   "PL-60"   "PL-61"
## [64] "PL-62"   "PL-63"   "PL-64"   "PL-65"   "PL-66"
```

```r
unique(boris_data_cutoff$fish.ID) # 69 unique IDs -> somehow an NA was introduced?
```

```
##  [1] "PF-31"   "PL-61"   "PL-46"   "PF-61"   "PF-26"   "PL-57"   NA
##  [8] "PL-65"   "PL-52"   "PL-53"   "PL-15"   "PL-64"   "PL-49"   "PF-65"
## [15] "PL-58"   "PL-50"   "PF-27"   "PL-12"   "PL-43"   "PL-45"   "PF-33"
## [22] "PF-35"   "PL-63"   "PL-21"   "PL-38"   "PF-62"   "PL-44"   "PF-25B"
## [29] "PL-05"   "PL-08"   "PL-55"   "PL-07"   "PF-30"   "PL-54"   "PL-13"
## [36] "PL-47"   "PF-60"   "PL-25"   "PL-28"   "PL-60"   "PL-09"   "PL-20"
## [43] "PL-22"   "PL-66"   "PL-26"   "PF-46"   "PL-56"   "PF-32"   "PL-59"
## [50] "PL-51"   "PF-08BR" "PL-41"   "PL-34"   "PL-62"   "PF-28"   "PL-48"
## [57] "PL-11"   "PF-39"   "PL-30"   "PL-02"   "PL-35"   "PL-39"   "PL-37"
## [64] "PF-36"   "PF-38"   "PL-31"   "PL-32"   "PF-40"   "PF-37"
```

```r
unique(parasite_data$fish.ID)
```

```
##   [1] "P.formosa"   "P.latipinna" "PF-01"       "PF-02"       "PF-04"
##   [6] "PF-05"       "PF-07"       "PF-08"       "PF-09"       "PF-10"
##  [11] "PF-11"       "PF-12"       "PF-13"       "PF-14"       "PF-15"
##  [16] "PF-16"       "PF-17"       "PF-18"       "PF-19"       "PF-20"
##  [21] "PF-21"       "PF-22"       "PF-23"       "PF-24"       "PF-25B"
##  [26] "PF-26"       "PF-27"       "PF-28"       "PF-29"       "PF-3"
##  [31] "PF-30"       "PF-31"       "PF-32"       "PF-33"       "PF-34"
##  [36] "PF-35"       "PF-36"       "PF-37"       "PF-38"       "PF-39"
##  [41] "PF-40"       "PF-41"       "PF-42"       "PF-43"       "PF-44"
##  [46] "PF-45"       "PF-46"       "PF-47"       "PF-49"       "PF-50"
##  [51] "PF-51"       "PF-52"       "PF-53"       "PF-54"       "PF-55"
##  [56] "PF-56"       "PF-57"       "PF-58"       "PF-59"       "PF-6"
##  [61] "PF-60"       "PF-61"       "PF-62"       "PF-63"       "PF-64"
##  [66] "PF-65"       "PL-01"       "PL-02"       "PL-04"       "PL-05"
##  [71] "PL-07"       "PL-08"       "PL-09"       "PL-10"       "PL-11"
##  [76] "PL-12"       "PL-13"       "PL-14"       "PL-15"       "PL-16"
```

```
##  [81] "PL-17"       "PL-18"       "PL-19"       "PL-20"       "PL-22"
##  [86] "PL-25"       "PL-26"       "PL-27"       "PL-28"       "PL-29"
##  [91] "PL-30"       "PL-31"       "PL-32"       "PL-34"       "PL-35"
##  [96] "PL-36"       "PL-37"       "PL-38"       "PL-40"       "PL-41"
## [101] "PL-42"       "PL-43"       "PL-44"       "PL-46"       "PL-47"
## [106] "PL-48"       "PL-49"       "PL-50"       "PL-51"       "PL-52"
## [111] "PL-53"       "PL-54"       "PL-55"       "PL-56"       "PL-57"
## [116] "PL-58"       "PL-60"       "PL-62"       "PL-63"       "PL-64"
## [121] "PL-66"
```

```
# 121, but 2 are just 'P.formosa' and 'P.latipina' because these are how fish were labeled if they didn
```

This all makes sense. The number of unique fish IDs from my notebook match up with the boris data. There are more parasite fish IDs because we sent them fish that didn't necessarily go through trials, in addition to trial fish.

```
# video.ID
unique(boris_data_wide$video.ID) # 36
```

```
##  [1] "WES_trial1_03_20220808.mov" "WES_trial2_02_20220812.mov"
##  [3] "WES_trial1_05_20220812.mov" "WES_trial2_01_20220812.mov"
##  [5] "WES_trial1_04_20220812.mov" "BR_trial1_02_20220808.mov"
##  [7] "WES_trial1_03_20220812.mov" "WES_trial3_02_20220809.mov"
##  [9] "WES_trial2_03_20220812.mov" "BR_trial3_02_20220808.mov"
## [11] "WES_trial2_03_20220808.mov" "WES_trial1_02_20220812.mov"
## [13] "WES_trial1_02_20220808.mov" "BR_trial2_02_20220809.mov"
## [15] "BR_trial2_02_20220808.mov"  "BR1_trial2_01_20220810.mov"
## [17] "BR2_trial2_01_20220810.mov" "WES_trial2_02_20220808.mov"
## [19] "WES_trial2_01_20220808.mov" "BR_trial2_01_20220808.mov"
## [21] "BR_trial1_01_20220808.mov"  "WES_trial2_04_20220812.mov"
## [23] "WES_trial1_01_20220808.mov" "BR_trial3_01_20220808.mov"
## [25] "WES_trial2_05_20220812.mov" "WES_trial3_01_20220809.mov"
## [27] "BR1_trial3_01_20220810.mov" "BR2_trial3_01_20220810.mov"
## [29] "BR1_trial3_02_20220810.mov" "BR2_trial3_02_20220810.mov"
## [31] "BR1_trial1_02_20220810.mov" "BR2_trial2_02_20220810.mov"
## [33] "Wes_trial1_01_20220812.mov" "BR2_trial1_02_20220810.mov"
## [35] "BR1_trial1_01_20220810.mov" "BR1_trial2_02_20220810.mov"
```

```
unique(ID_data$video.ID) # 36
```

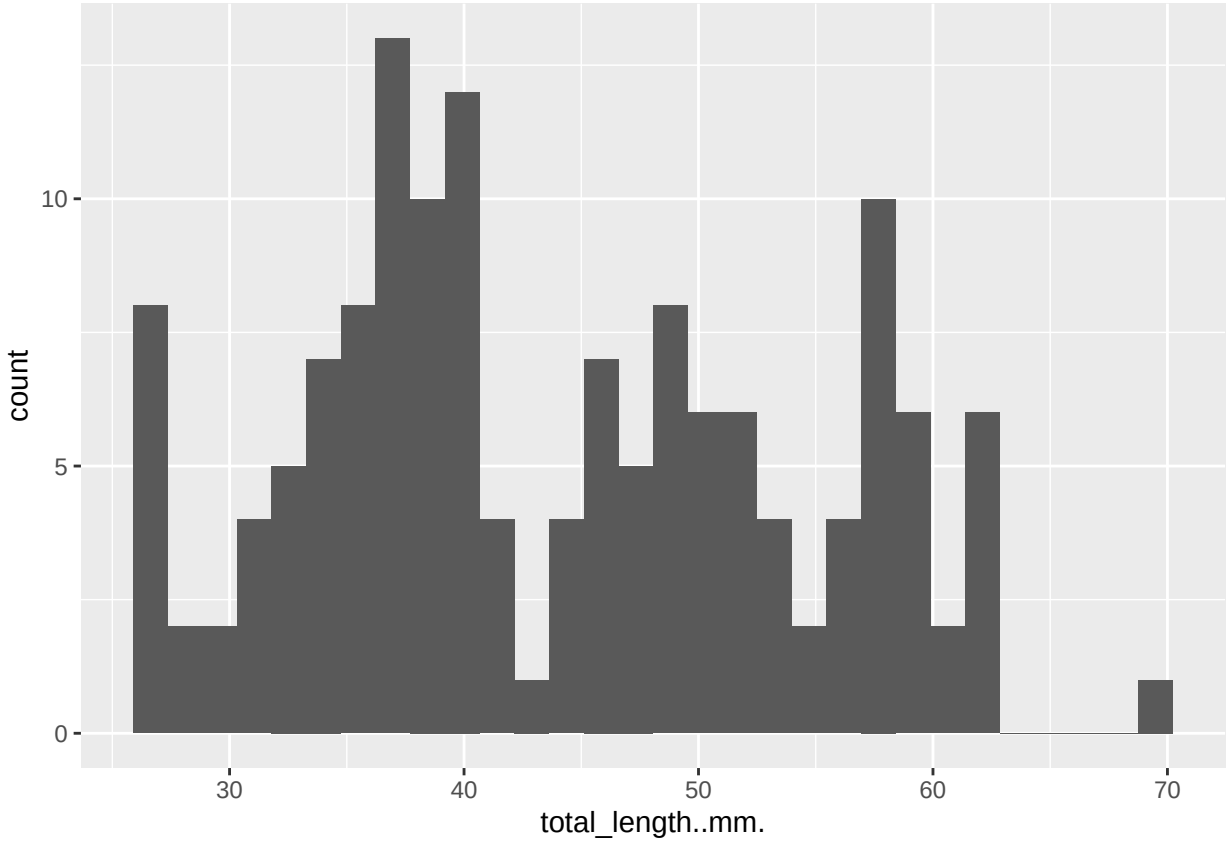
```
##  [1] "BR_trial1_01_20220808.mov"  "BR_trial3_01_20220808.mov"
##  [3] "BR_trial2_01_20220808.mov"  "WES_trial1_01_20220808.mov"
##  [5] "WES_trial2_01_20220808.mov" "WES_trial3_02_20220809.mov"
##  [7] "WES_trial1_02_20220808.mov" "WES_trial2_02_20220808.mov"
##  [9] "WES_trial1_03_20220808.mov" "WES_trial2_03_20220808.mov"
## [11] "WES_trial3_01_20220809.mov" "BR1_trial1_01_20220810.mov"
## [13] "BR1_trial2_01_20220810.mov" "BR2_trial1_01_20220810.mov"
## [15] "BR2_trial2_01_20220810.mov" "BR2_trial1_02_20220810.mov"
## [17] "BR2_trial2_02_20220810.mov" "BR2_trial3_02_20220810.mov"
## [19] "BR2_trial3_01_20220810.mov" "BR1_trial3_01_20220810.mov"
## [21] "BR1_trial1_02_20220810.mov" "BR1_trial2_02_20220810.mov"
## [23] "WES_trial1_04_20220812.mov" "WES_trial2_04_20220812.mov"
## [25] "WES_trial1_05_20220812.mov" "WES_trial2_05_20220812.mov"
## [27] "BR_trial1_02_20220808.mov"  "BR_trial2_02_20220808.mov"
## [29] "BR_trial3_02_20220808.mov"  "BR1_trial3_02_20220810.mov"
## [31] "WES_trial1_02_20220812.mov" "WES_trial2_02_20220812.mov"
```

```
## [33] "WES_trial1_03_20220812.mov" "WES_trial2_03_20220812.mov"
## [35] "Wes_trial1_01_20220812.mov" "WES_trial2_01_20220812.mov"
```

This also makes sense. We have the same number of video IDs in the boris and ID data.

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 46 rows containing non-finite values (`stat_bin()`).
```



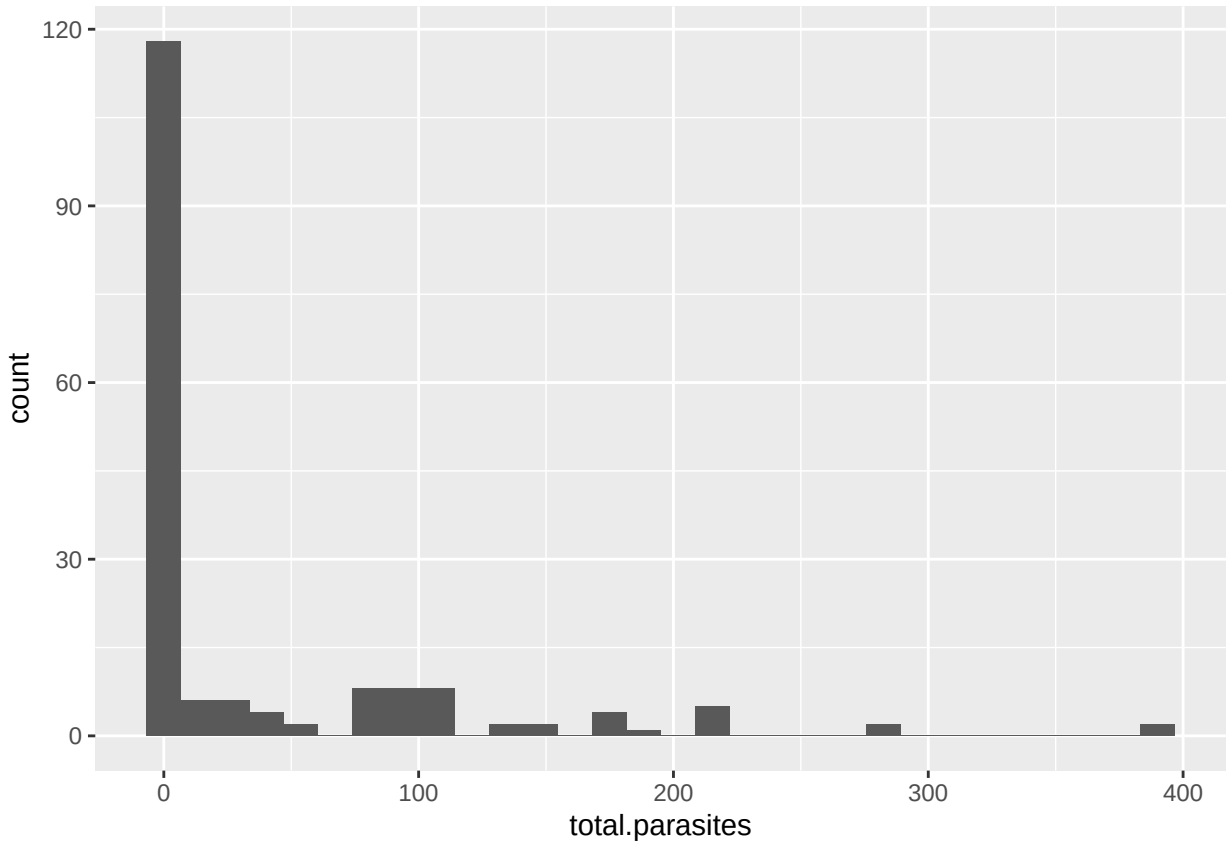
Seems like a fairly normal distribution for length? possibly bimodal?

Let's take a look at the parasite data.

```
# parasite data
parasite_hist <- total_hide_open %>%
  ggplot(mapping = aes(total.parasites)) +
  geom_histogram()
parasite_hist
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 15 rows containing non-finite values (`stat_bin()`).
```

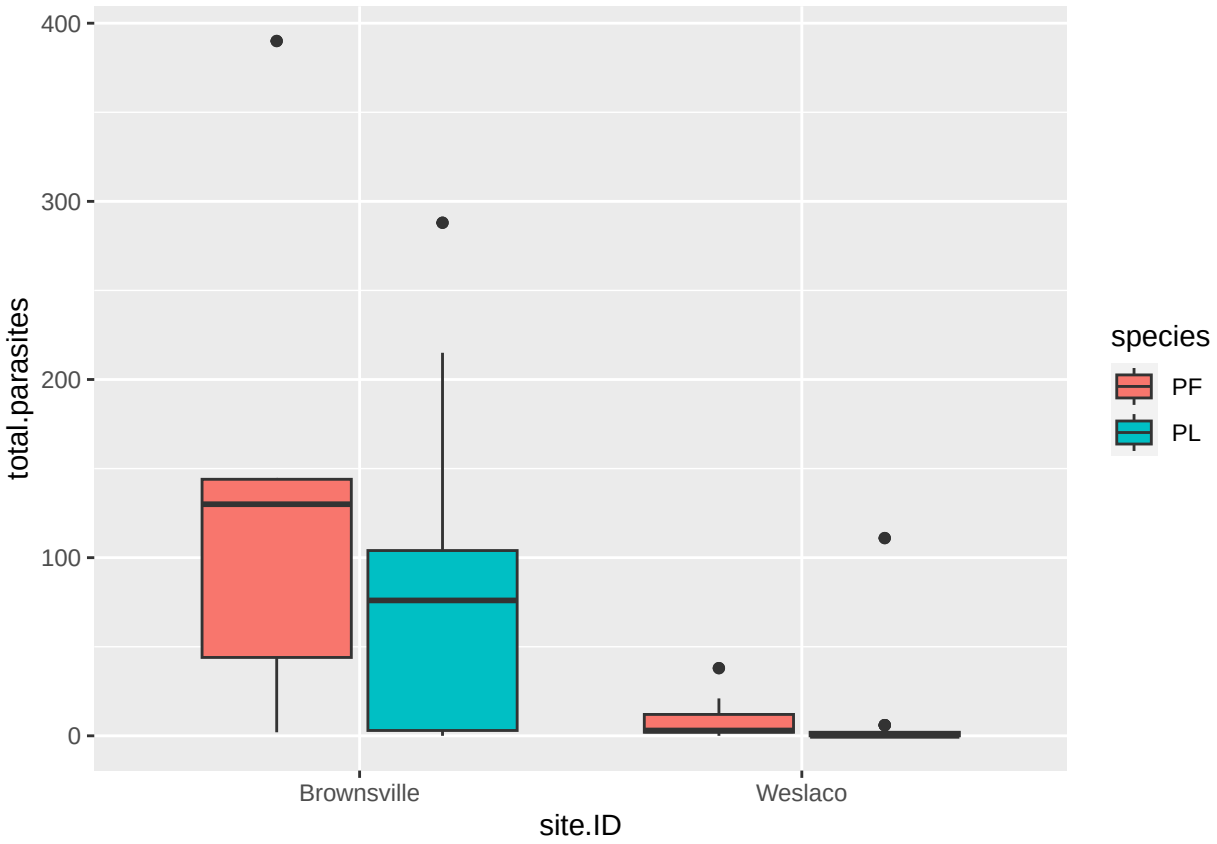


Not a normal distribution. No obvious pattern here, besides a lot of zeros. Should check to see how this matches up with notes in the parasite data about specimen quality).

Let's look at the parasites by species and site.

```
sp_parasite_box <- total_hide_open %>%
  ggplot(mapping = aes(
    fill = species,
    x = site.ID,
    y = total.parasites
  )) +
  geom_boxplot()
sp_parasite_box
```

```
## Warning: Removed 15 rows containing non-finite values (`stat_boxplot()`).
```

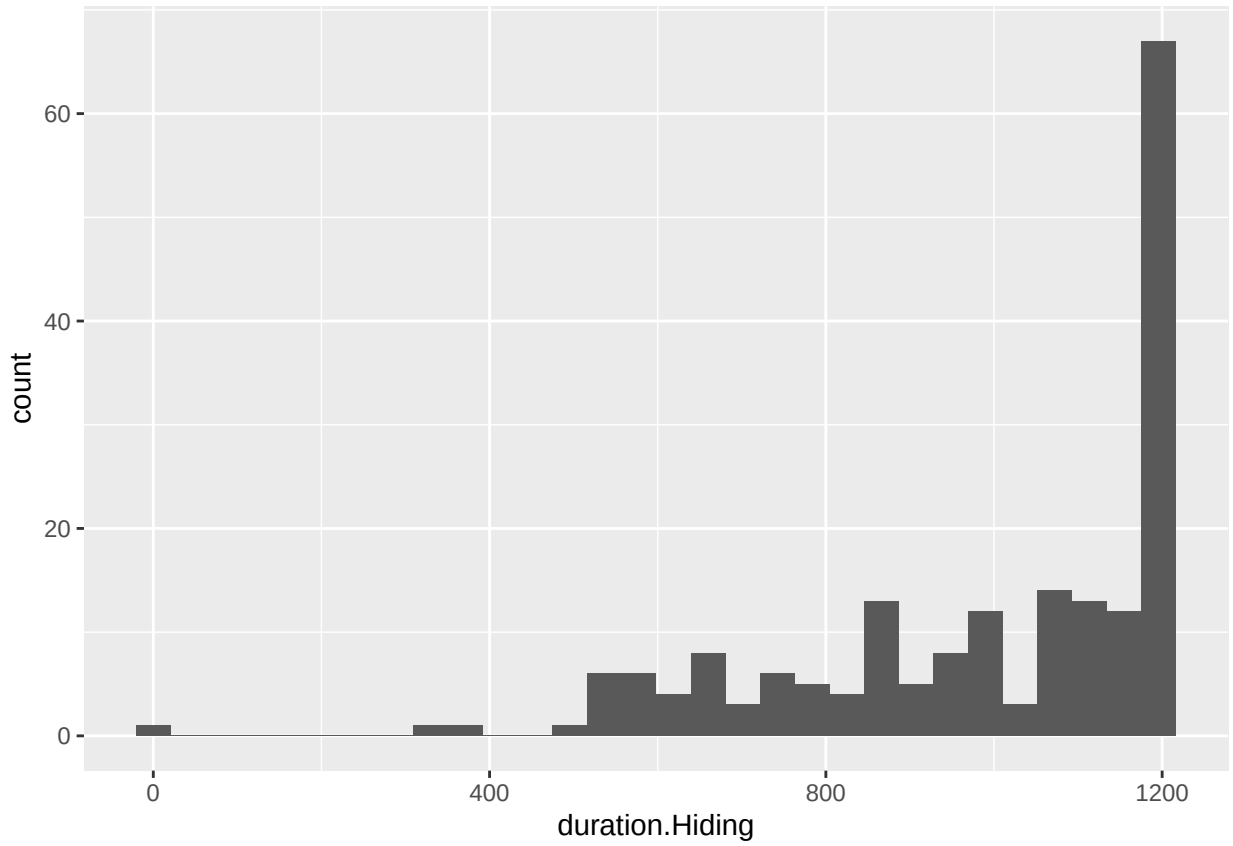


Ok, so there is a clear pattern of way more parasites in Brownsville, generally. It also seems like there may be more parasites on amazons in both sites, but we'll see what the stats say.

Now, let's take a look at the shape of the behavior data.

```
# boris_data, distributions
hiding_hist <- total_hide_open %>%
  ggplot(mapping = aes(duration.Hiding)) +
  geom_histogram()
hiding_hist
```

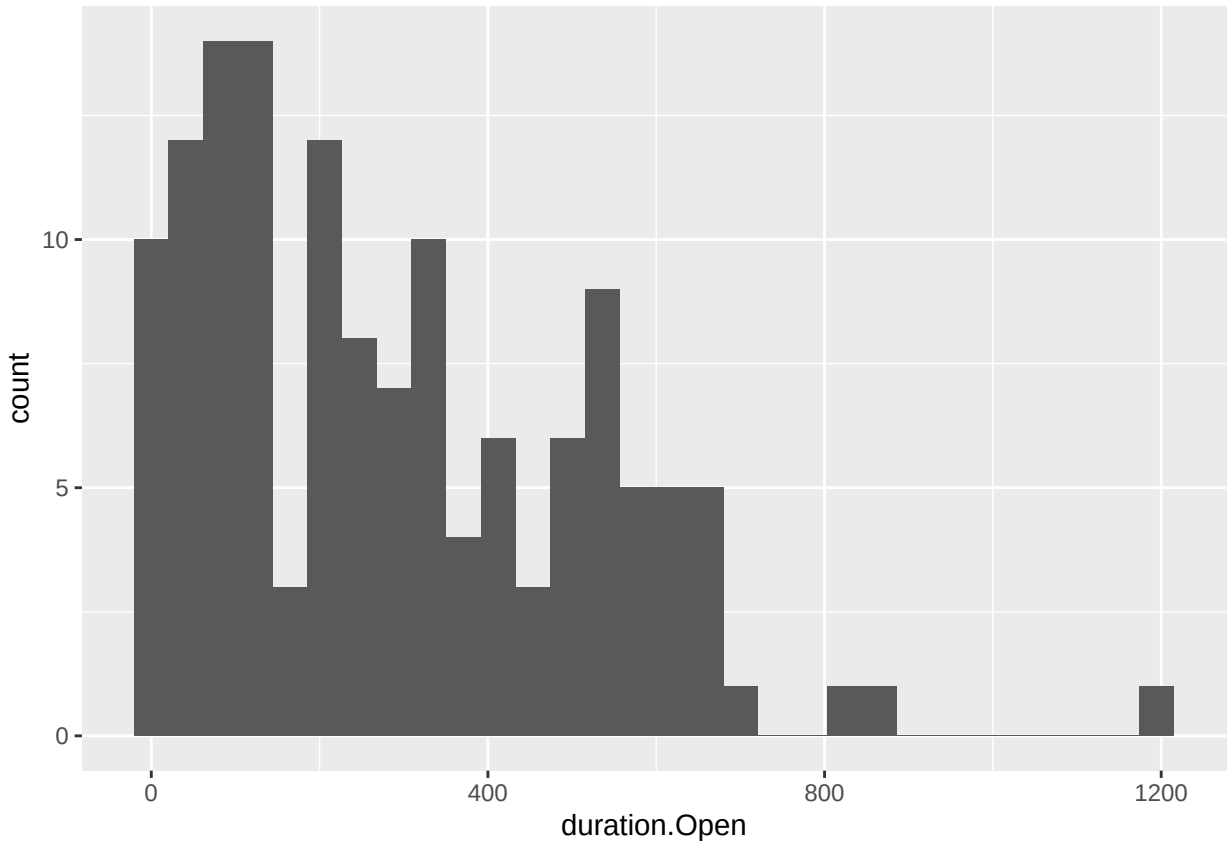
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
open_hist <- total_hide_open %>%
  ggplot(mapping = aes(duration.Open)) +
  geom_histogram()
open_hist
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

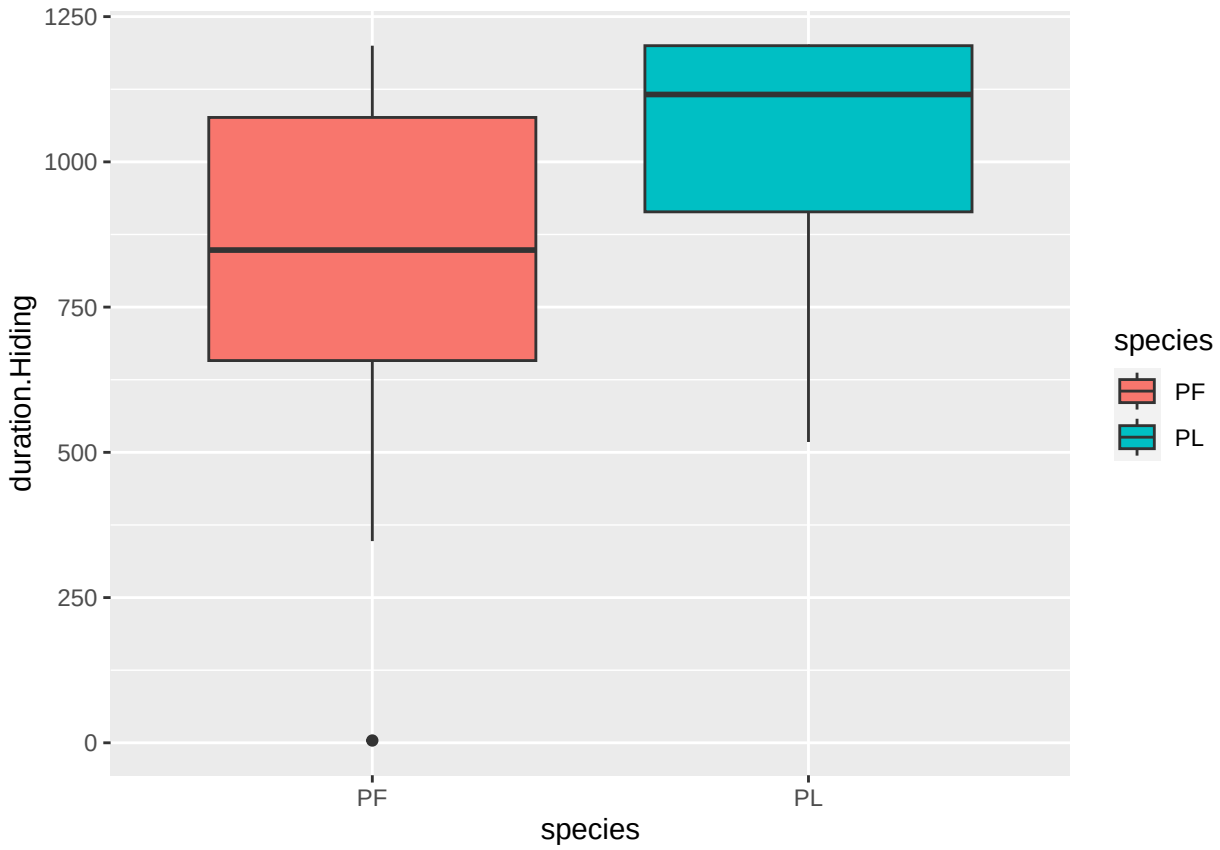
```
## Warning: Removed 56 rows containing non-finite values (`stat_bin()`).
```



Ok, so it looks like we have a fairly normal distribution for duration hiding, with a right skew. For time in the open, we have a floor of zero, so a strong left skew. This indicates that many fish spent the entire, or most of the trial hiding.

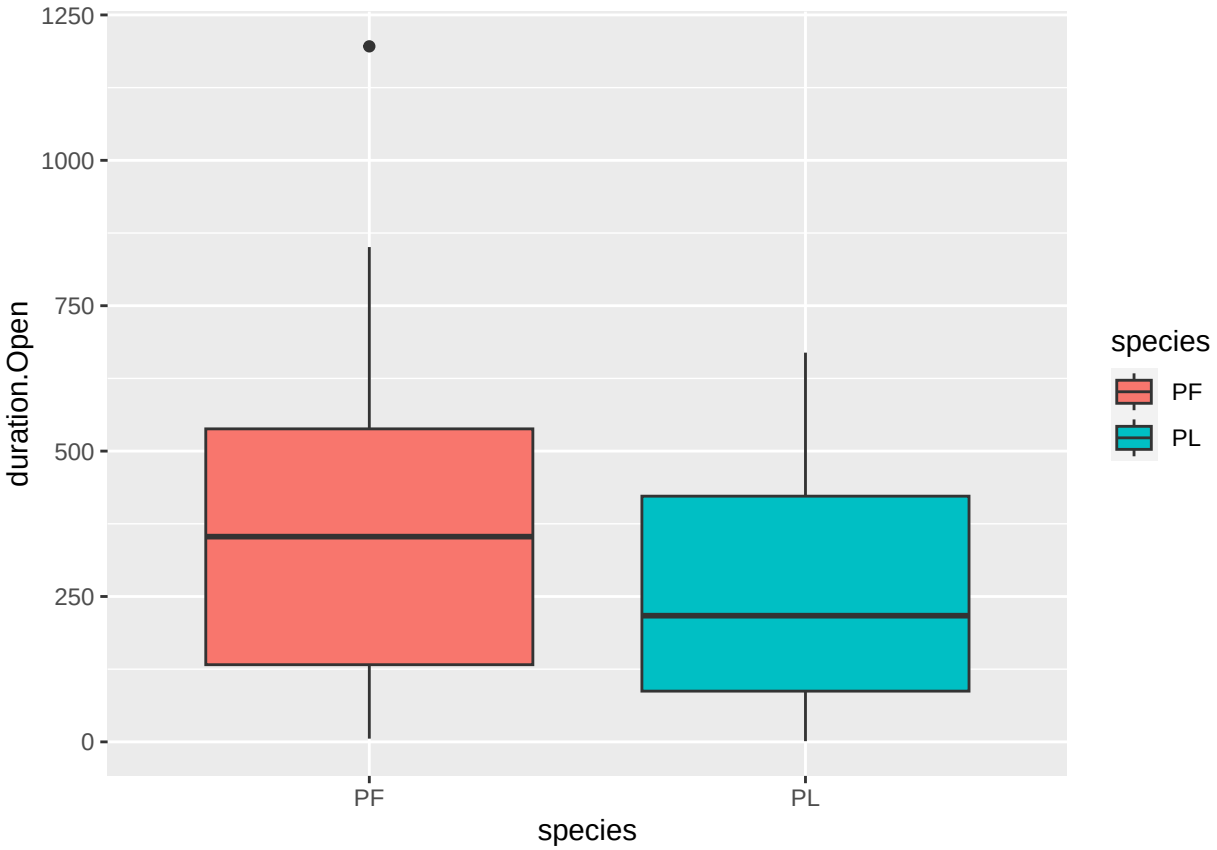
Now, let's take a look at the open vs. hiding data by species.

```
# diff in hiding between species
species_hiding <- total_hide_open %>%
  ggplot(mapping = aes(
    x = species,
    y = duration.Hiding,
    fill = species
  )) +
  geom_boxplot()
species_hiding
```




```
# diff in open between species
species_open <- total_hide_open %>%
  ggplot(mapping = aes(
    x = species,
    y = duration.Open,
    fill = species
  )) +
  geom_boxplot()
species_open
```

```
## Warning: Removed 56 rows containing non-finite values (`stat_boxplot()`).
```



Again, we'll see how the stats pan out, but it looks like Amazons might spend less time hiding and more time in the open than sailfins.

Models

Parasites

First, I want to see if there is a difference in parasite load between Amazons and Sailfins.

```
# FULL DATA (both sites)

# the parasite count data is zero inflated and overdispersed, so I'm going to use a zero-inlfated negat

# I'm going to use backwards elimination

# interaction model
mod_para_interaction <- zeroinfl(totalpara ~ species * site.ID,
  dist = "negbin",
  lin = "logit",
  data = parasite_data
)

# combined model
mod_para_combined <- zeroinfl(totalpara ~ species + site.ID,
  dist = "negbin",
  lin = "logit",
  data = parasite_data
)
```

```r
# species model
mod_para_species <- zeroinfl(totalpara ~ species,
  dist = "negbin",
  lin = "logit",
  data = parasite_data
)

# site model
mod_para_site <- zeroinfl(totalpara ~ site.ID,
  dist = "negbin",
  lin = "logit",
  data = parasite_data
)

# test 2-way with log likelihood ratio test
lrtest(mod_para_combined, mod_para_interaction) # no significant difference
```

```
## Likelihood ratio test
##
## Model 1: totalpara ~ species + site.ID
## Model 2: totalpara ~ species * site.ID
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -474.97
## 2   9 -474.74  2 0.4566     0.7959
```

```r
# test species effect
lrtest(mod_para_site, mod_para_combined) # the combined model fits the data much better
```

```
## Likelihood ratio test
##
## Model 1: totalpara ~ site.ID
## Model 2: totalpara ~ species + site.ID
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   5 -479.64
## 2   7 -474.97  2 9.3294   0.009422 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
lrtest(mod_para_species, mod_para_combined) # same results, the combined model is better
```

```
## Likelihood ratio test
##
## Model 1: totalpara ~ species
## Model 2: totalpara ~ species + site.ID
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   5 -508.08
## 2   7 -474.97  2 66.222   4.17e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(mod_para_combined) # but there is no significant effect of species or site.ID
```

```
##
## Call:
## zeroinfl(formula = totalpara ~ species + site.ID, data = parasite_data,
```

```
##     dist = "negbin", link = "logit")
##
## Pearson residuals:
##     Min      1Q  Median      3Q     Max
## -0.6715 -0.5375 -0.4157 -0.1079 10.3690
##
## Count model coefficients (negbin with log link):
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       4.84359    0.31721  15.269  < 2e-16 ***
## specieslatipinna -0.05922    0.31583  -0.188    0.851
## site.IDWeslaco   -2.46194    0.32398  -7.599 2.98e-14 ***
## Log(theta)       -0.79177    0.13760  -5.754 8.71e-09 ***
##
## Zero-inflation model coefficients (binomial with logit link):
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -16.658     60.684  -0.275    0.784
## specieslatipinna    9.443     55.611   0.170    0.865
## site.IDWeslaco      6.742     24.210   0.278    0.781
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Theta = 0.453
## Number of iterations in BFGS optimization: 36
## Log-likelihood:  -475 on 7 Df
```

```r
## Now, just with the WESLACO site ##

# filter data to just Weslaco
parasite_data_wes <- parasite_data %>%
  filter(site.ID == "Weslaco")

# species model
mod_para_species_wes <- zeroinfl(totalpara ~ species,
  dist = "negbin",
  lin = "logit",
  data = parasite_data_wes
)

summary(mod_para_species_wes)
```

```
##
## Call:
## zeroinfl(formula = totalpara ~ species, data = parasite_data_wes, dist = "negbin",
##     link = "logit")
##
## Pearson residuals:
##     Min      1Q  Median      3Q     Max
## -0.5821 -0.4700 -0.4520 -0.2490  9.5692
##
## Count model coefficients (negbin with log link):
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       2.33983    0.22557  10.373  < 2e-16 ***
## specieslatipinna  0.01086    0.46684   0.023    0.981
## Log(theta)       -1.04896    0.17916  -5.855 4.77e-09 ***
##
```

```
## Zero-inflation model coefficients (binomial with logit link):
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -11.70      81.48  -0.144    0.886
## specieslatipinna    10.99      81.48   0.135    0.893
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Theta = 0.3503
## Number of iterations in BFGS optimization: 36
## Log-likelihood: -252.3 on 5 Df
```

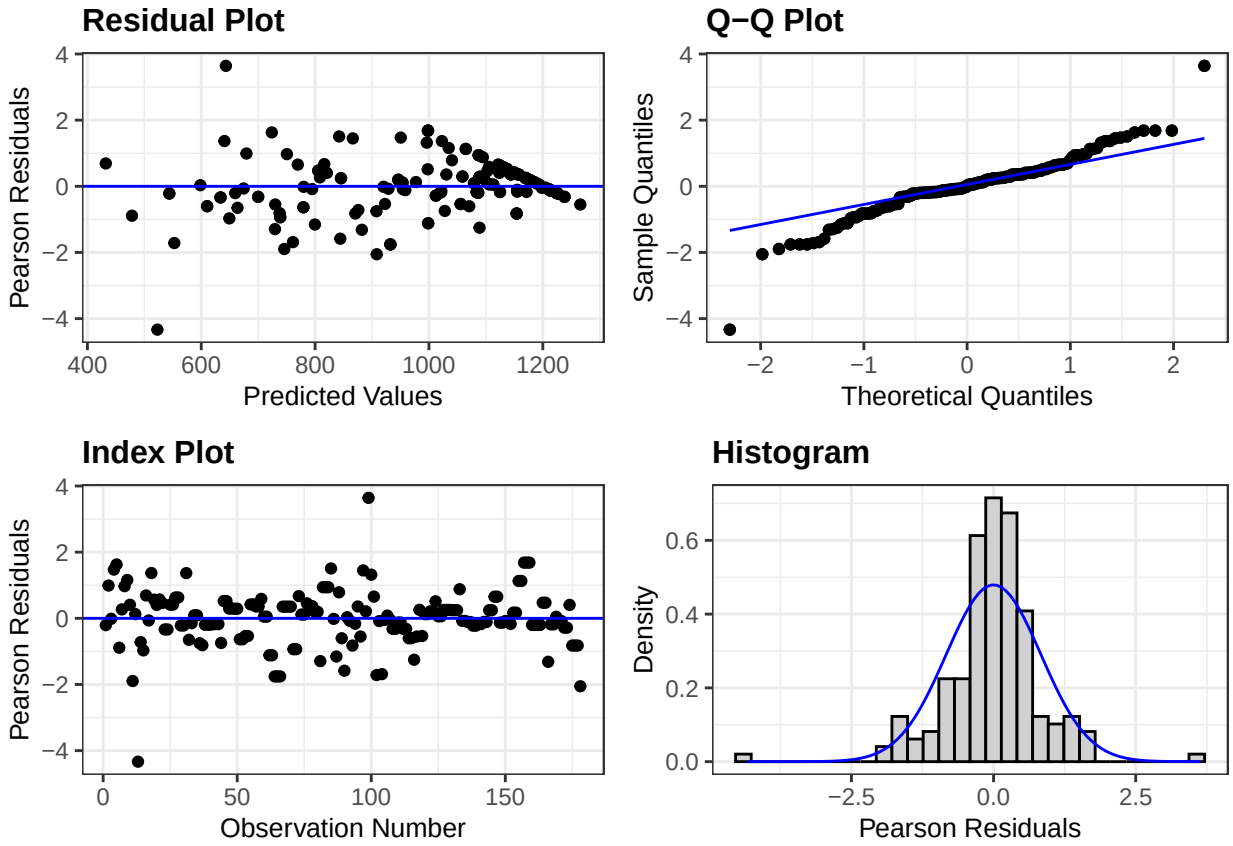
I want to see if the time spent hiding is predicted by parasites, species, trial number, fish.ID or their interaction.

```
# full model
mod_full <- lmer(duration.Hiding ~ total.parasites * species * trial + (1 | fish.ID),
  data = total_hide_open
)
summary(mod_full)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: duration.Hiding ~ total.parasites * species * trial + (1 | fish.ID)
##    Data: total_hide_open
##
## REML criterion at convergence: 2282.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.3327 -0.2824  0.0098  0.3952  3.6412
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  fish.ID  (Intercept) 32702    180.8
##  Residual             14350    119.8
## Number of obs: 178, groups:  fish.ID, 60
##
## Fixed effects:
##                                    Estimate Std. Error t value
## (Intercept)                       7.094e+02  5.655e+01  12.545
## total.parasites                   1.217e+00  5.469e-01   2.225
## speciesPL                         2.821e+02  6.746e+01   4.182
## trial2                            1.203e+02  4.436e+01   2.711
## total.parasites:speciesPL        -1.392e+00  7.294e-01  -1.909
## total.parasites:trial2            5.686e-04  4.278e-01   0.001
## speciesPL:trial2                 -5.421e+01  5.017e+01  -1.080
## total.parasites:speciesPL:trial2  2.262e-01  5.671e-01   0.399
##
## Correlation of Fixed Effects:
##             (Intr) ttl.pr spcsPL trial2 tt.:PL ttl.:2 spPL:2
## total.prsts -0.451
## speciesPL   -0.838  0.378
## trial2      -0.401  0.168  0.336
## ttl.prst:PL  0.338 -0.750 -0.453 -0.126
## ttl.prsts:2  0.168 -0.394 -0.141 -0.443  0.296
## spcsPL:trl2  0.355 -0.149 -0.383 -0.884  0.179  0.392
```

```
## ttl.pr:PL:2 -0.127  0.297  0.173  0.334 -0.452 -0.754 -0.435
```

```
## evaluating assumptions
ggResidpanel::resid_panel(mod_full)
```



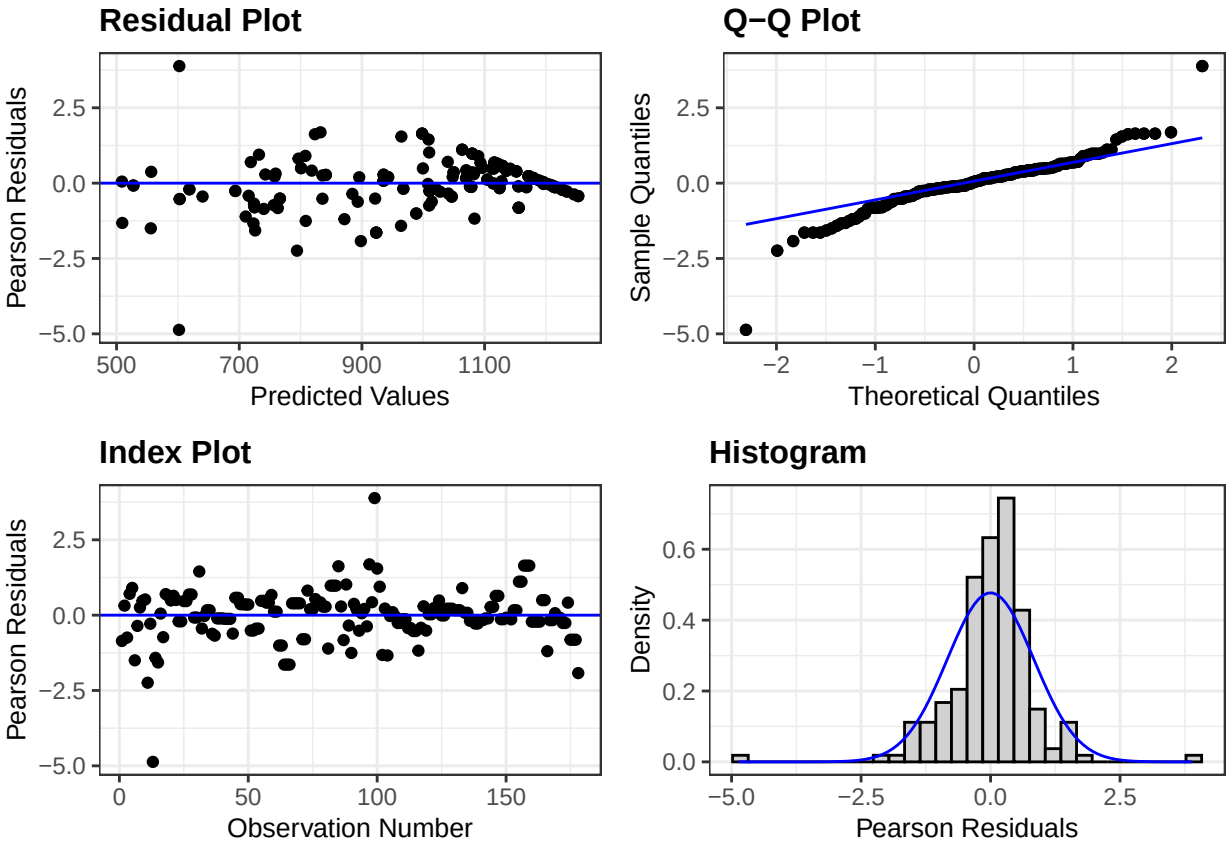
Looking at our residual panel, most assumptions look ok! Residuals vs predicted might be a bit trumet-y? Q-Q looks nice and linear. Index plot is an even scatter. Residual histogram looks pretty normal.

```
# decompose to two way
mod_2way <- lmer(duration.Hiding ~ total.parasites:species + total.parasites:trial + species:trial + (1
  data = total_hide_open
)
summary(mod_2way)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: duration.Hiding ~ total.parasites:species + total.parasites:trial +
##     species:trial + (1 | fish.ID)
##    Data: total_hide_open
##
## REML criterion at convergence: 2321.1
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.8680 -0.2830  0.0408  0.4166  3.8861
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  fish.ID  (Intercept) 39959    199.9
##  Residual             15076    122.8
```

```
## Number of obs: 178, groups:  fish.ID, 60
##
## Fixed effects:
##                             Estimate Std. Error t value
## (Intercept)                920.4339    32.4630  28.353
## total.parasites:speciesPF    0.4140     0.5312   0.779
## total.parasites:speciesPL    0.1357     0.4662   0.291
## total.parasites:trial2       0.2954     0.2737   1.080
## speciesPL:trial2            74.8811    22.4423   3.337
##
## Correlation of Fixed Effects:
##             (Intr) tt.:PF tt.:PL ttl.:2
## ttl.prst:PF -0.299
## ttl.prst:PL -0.388  0.208
## ttl.prsts:2  0.100 -0.293 -0.384
## spcsPL:trl2 -0.247  0.157  0.068 -0.336
```

```
## evaluating assumptions
ggResidpanel::resid_panel(mod_2way)
```




```
anova(mod_full, mod_2way)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: total_hide_open
## Models:
## mod_2way: duration.Hiding ~ total.parasites:species + total.parasites:trial + species:trial + (1 | f
## mod_full: duration.Hiding ~ total.parasites * species * trial + (1 | fish.ID)
##          npar    AIC    BIC  logLik deviance  Chisq Df Pr(>Chisq)
```

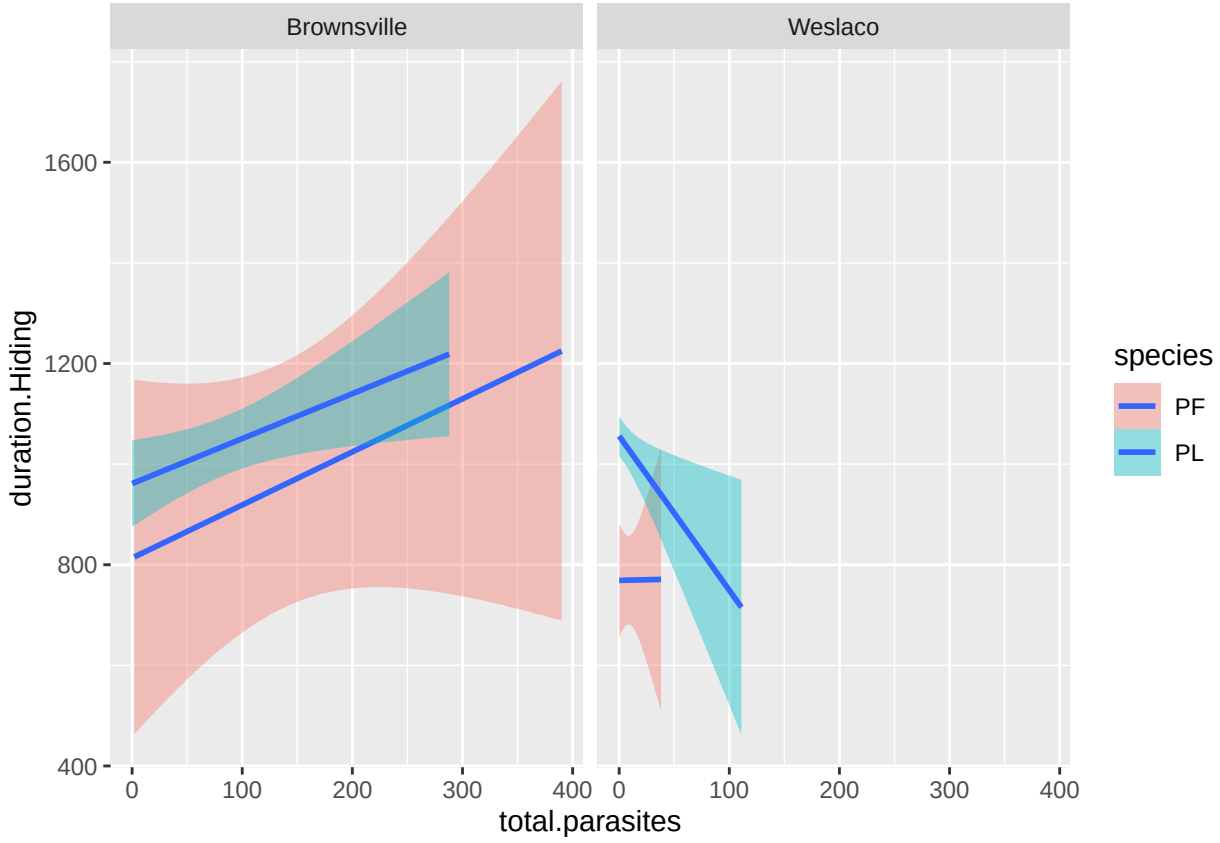
```
## mod_2way    7 2351.3 2373.6 -1168.7   2337.3
## mod_full   10 2337.7 2369.5 -1158.8   2317.7 19.648  3  0.0002008 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ok now I'm going to make a plot to disentangle the pairwise interaction we've got going on.

```
# This plots number of parasites vs time spent hiding. Each line represents a species.
# I've also split the plot into the two sites: Brownsville and Weslaco.
pairwise_species_plot <- total_hide_open %>%
  ggplot(mapping = aes(
    y = duration.Hiding,
    x = total.parasites,
    fill = species
  )) +
  geom_smooth(method = "lm") +
  facet_wrap(vars(site.ID))
pairwise_species_plot
```

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

```
## Warning: Removed 15 rows containing non-finite values (`stat_smooth()`).
```

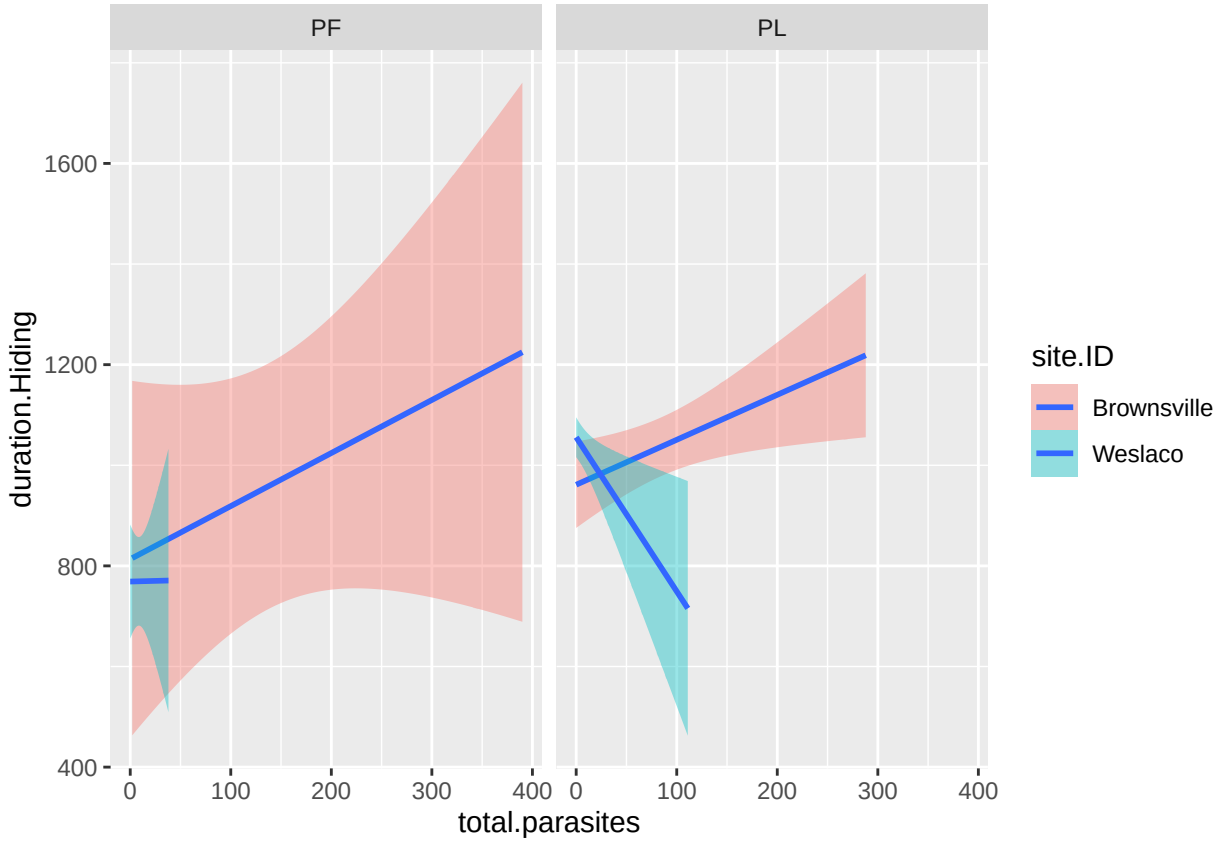



```
pairwise_site_plot <- total_hide_open %>%
  ggplot(mapping = aes(
    y = duration.Hiding,
    x = total.parasites,
    fill = site.ID
  )) +
```

```
  geom_smooth(method = "lm") +
  facet_wrap(vars(species))
pairwise_site_plot
```

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

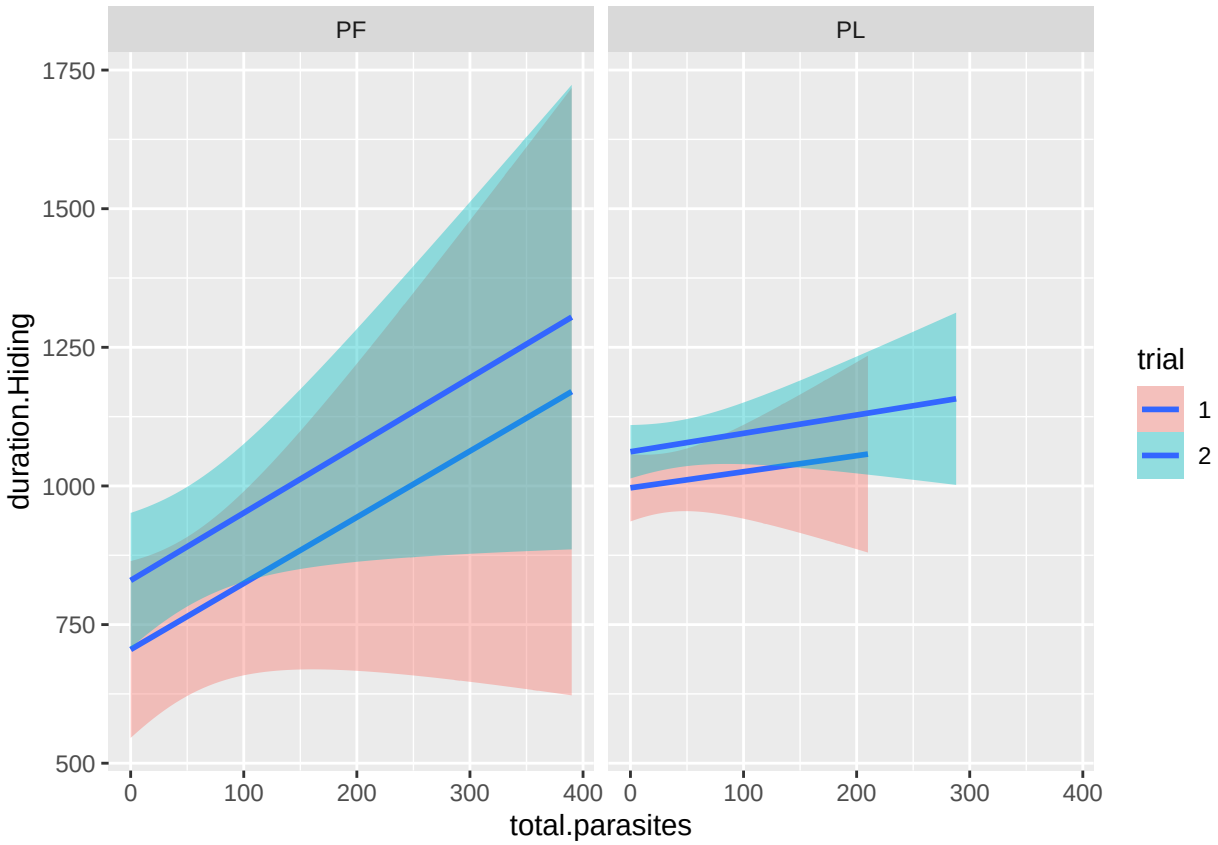
```
## Warning: Removed 15 rows containing non-finite values (`stat_smooth()`).
```




```
pairwise_trial_plot <- total_hide_open %>%
  ggplot(mapping = aes(
    y = duration.Hiding,
    x = total.parasites,
    fill = trial
  )) +
  geom_smooth(method = "lm") +
  facet_wrap(vars(species))
pairwise_trial_plot
```

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

```
## Warning: Removed 15 rows containing non-finite values (`stat_smooth()`).
```




```
# note from Kate: add raw data points
```

PICK UP HERE

Also, check out how long the trials should be and export the observations from Boris cut off at that length (25 minutes, I think?). Double check that observations that start before the cutoff are assigned a new cutoff at the end trial time and not removed or left longer.

SCRAPS

NOTE

There were lots of other plots I made examining site by site or trial by trial patterns as well. They are copied below, but have not been reviewed/error checked.

```
# change in hiding over trials by species
trial_hiding <- total_hide_open %>%
  ggplot(mapping = aes(
    x = trial.ID,
    y = duration_Hiding,
    fill = species,
    dodge = species
  )) +
  geom_boxplot()

# change in hiding over trials by species
trial_open <- total_hide_open %>%
  ggplot(mapping = aes(
```

```
    x = trial.ID,
    y = duration_Open,
    fill = species,
    dodge = species
  )) +
  geom_boxplot()

# diff in open between sites
site_open <- total_hide_open %>%
  ggplot(mapping = aes(
    x = site.ID.x,
    y = duration_Open
  )) +
  geom_boxplot()

# diff in hiding between sites
site_hiding <- total_hide_open %>%
  ggplot(mapping = aes(
    x = site.ID.x,
    y = duration_Hiding
  )) +
  geom_boxplot()
# diff in opn between sites by species
site_spp_open <- total_hide_open %>%
  ggplot(mapping = aes(
    x = site.ID.x,
    y = duration_Open,
    fill = species.x,
    dodge = species.x
  )) +
  geom_boxplot()

# diff in hiding between sites by species
site_spp_hiding <- total_hide_open %>%
  ggplot(mapping = aes(
    x = site.ID.x,
    y = duration_Hiding,
    fill = species.x,
    dodge = species.x
  )) +
  geom_boxplot()

# diff in total parasites by species
parasites_spp <- total_hide_open %>%
  ggplot(mapping = aes(
    x = species.x,
    y = totalpara,
    fill = species.x
  )) +
  geom_boxplot()

# diff in total parasites by site
parasites_site <- total_hide_open %>%
```

```
  ggplot(mapping = aes(
    x = site.ID.x,
    y = totalpara
  )) +
  geom_boxplot()

# diff in total parasites by site and spp
parasites_site_spp <- total_hide_open %>%
  ggplot(mapping = aes(
    x = site.ID.x,
    y = totalpara,
    fill = species.x,
    dodge = species.x
  )) +
  geom_boxplot()

# variation in parasites with open beh
parasites_open <- total_hide_open %>%
  ggplot(mapping = aes(
    x = totalpara,
    y = duration_Open,
    color = species.x
  )) +
  geom_point()

# variation in parasites with hiding beh
parasites_hiding <- total_hide_open %>%
  ggplot(mapping = aes(
    x = totalpara,
    y = duration_Hiding,
    fill = species.x
  )) +
  geom_smooth()

# avg length at each site by spp.
length_site <- total_hide_open %>%
  ggplot(mapping = aes(
    x = site.ID.x,
    y = total_length..mm.,
    fill = species.x,
    dodge = species.x
  )) +
  geom_boxplot()
```