

# Berkeley Analysis of Human Schisto Data

## Background

Longitudinal data analysis of human parasitological data among communities in Senegal with prawn interventions, vegetation removal, and net addition. Three parasitological sampling events in February of 2016, 2017, and 2018 attempted to measure *S. haematobium* eggs/10mL urine and *S. mansoni* EPG among school children in 16 communities. Praziquantel administration occurred in April 2016 and in June of 2017. Vegetation removal and prawn introduction interventions were initiated following PZQ administration in June 2017. Nets were added in 4 villages following PZQ administration in Feb 2016 to assess if there is a net effect independent of prawn addition.

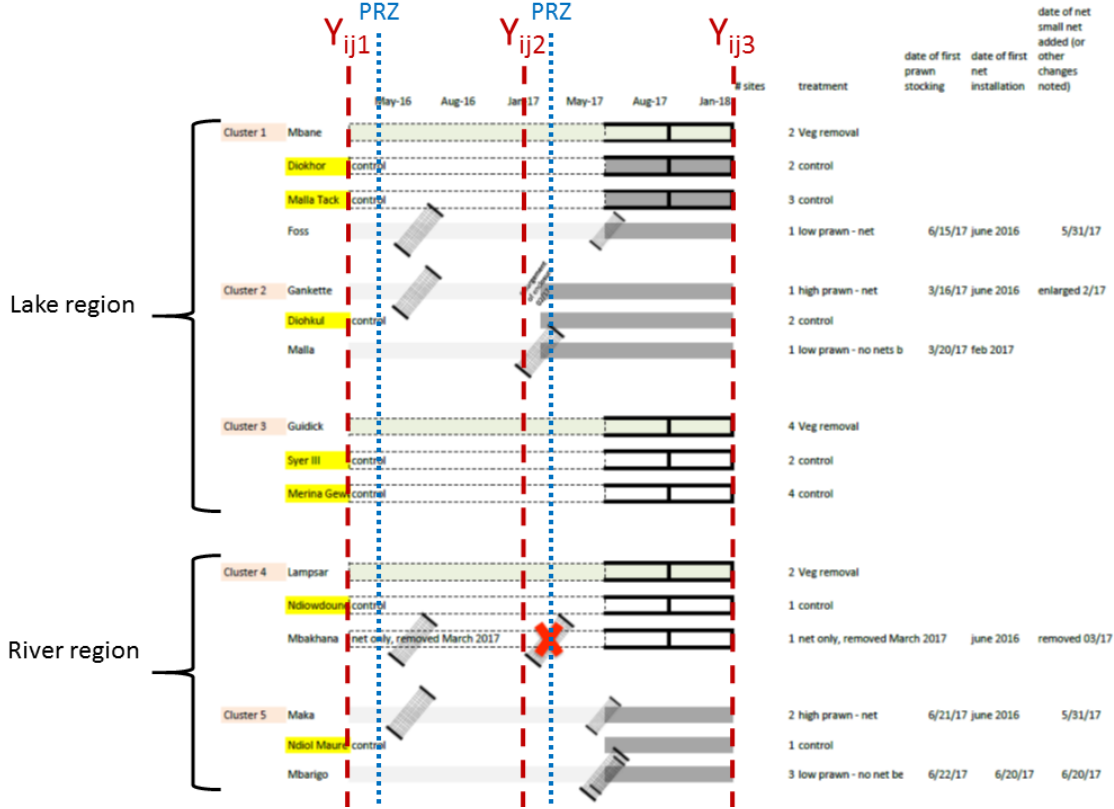


Figure 1: Study design

Our main question is: how do the interventions affect transmission? The interventions were implemented in June/July 2017 following the 2017 PZQ administration and therefore only affect the 2018 survey data. An ideal analysis should therefore leverage the earlier surveys to account for individual, community, and regional variability while relying on the 2018 survey to identify the effects of each intervention. More formally, our statistical question may be framed as:

$$\mathbb{E}[Y_{ij3}|Y_{ij2}, Y_{ij1}] = \mathbb{E}[Y_{ij3}|I_{j3}, Y_{ij2}, Y_{ij1}]$$

where  $Y_{ijt}$  is the egg burden of individual  $i$  in community  $j$  in year  $t$  and  $I_{j3}$  is the intervention that community  $j$  receives in year 3 (2018). We would hope:

$$\mathbb{E}[Y_{ij3}|Y_{ij2}, Y_{ij1}] > \mathbb{E}[Y_{ij3}|I_{j3}, Y_{ij2}, Y_{ij1}]$$

but what we want to be sure of is that the intervention is not harmful or:

$$\mathbb{E}[Y_{ij3}|Y_{ij2}, Y_{ij1}] < \mathbb{E}[Y_{ij3}|I_{j3}, Y_{ij2}, Y_{ij1}]$$

## Clarifying Questions:

Safe to assume all children with egg burden measures were treated with PZQ?

+ Yes

What is the source/point of clusters?

+ From Giulio: “the 5 groups were derived through a rigorous cluster analysis that accounted for location (lake/river), population size and distance (to simply logistic and homogenized microclimate differences through the geographical reason)”

+ From Sanna: “There was a protocol to assign clusters based on matching of several variables for each site: latitude, longitude, water point # and cattle presence (qualitative). Once clusters were assigned, they were used to stratify the randomization of treatment. With such a small sample size (16 sites), randomization might otherwise have failed — ie. all prawn treatments may have ended up on the Lake, or at high cattle use sites, or at places with many water points. Thus, we intended to assign one high prawn and one low prawn village in each cluster randomly and ensure control sites of comparable ecology were also being tracked. Somewhere in the middle of the experiment (at the end of the first follow up year) because we lacked prawns, [difficult] executive decisions were made by Giulio, so he can explain more why one high and one low prawn treatment were applied in each of 3 clusters (rather than one single density (or several densities) of prawns were assigned in each of the 5 clusters). Whether “cluster” remains in the analysis doesn’t matter as much to me as having used the clusters in the first place to ensure a wide stratification of treatments and controls in space and across other influential ecological variables. Note that “high prawn” and “low prawn” should probably be referred to as “very low prawn” and “extremely low prawn” when we look at the actual realized densities across the transmission periods in the prawn sites.”

Better to use mean of multiple data points or single data point for less missingness?

+ Giulio says using a single stool sample for *S. mansoni* underestimates prevalence on the village level, but that for four communities, only 1 stool sample was collected. In our analyses to follow average individual egg burden is estimated for each individual and for each species (mansoni and haematobium) and an additional weight variable is constructed which indicates the number of egg burden estimates that contribute to that mean (max of 2 for haematobium, max of 4 for mansoni)

Giulio says ~80 children in Maka Diama (prawn village) and Ndiol Maure (control village) received PZQ in 2017 outside of our study and should therefore be excluded, but the 2017 dataset and the full dataset shared by Stanford has all children from Diokhor Tack, Malla, Gankette Guinth, Merina Guewal, Maka Diama, and Ndiol Maure coded as receiving extra PZQ in 2017

## Data loading, cleaning, and merging

```
#My interventions data generated from interpretation of spreadsheets shared by Giulio
```

```
interventions <- read_csv("~/RemaisWork/Schisto/Stanford/Human_Parasitology_data/Village_interventions.csv") %>%  
  mutate(Intervention = factor(Intervention, levels = c("control", "net", "prawn", "veg_removal")),  
         Region = factor(Region),  
         Prawn_cat = factor(Prawn_cat, levels = c("No", "low", "high")))
```

```
## Parsed with column specification:
```

```
## cols(  
##   Intervention_2018 = col_character(),  
##   Intervention = col_character(),  
##   Village = col_character(),  
##   School = col_character(),  
##   Region = col_character(),  
##   year = col_integer(),  
##   Veg = col_integer(),  
##   Veg_tons = col_double(),  
##   Net = col_integer(),  
##   Net_months = col_integer(),  
##   Prawn_cat = col_character(),  
##   Prawns = col_integer(),  
##   Prawn_density = col_double(),  
##   Prawn_biomass = col_double()  
## )
```

```
dat1 <- read_csv("~/RemaisWork/Schisto/Stanford/Human_Parasitology_data/First_round_human_subject_analysis_APL")  
#data munging
```

```
#sort(unique(dat1$P2_omega_total))
```

```
# Get means from different samples, rename variables, and reorder
```

```
dat1 <- dat1 %>%
```

```

group_by(Child_ID) %>%
mutate(year = 2016, School = as.character(School),
  # Replace >200 with 200 then coerce to numeric
  P2_omega_total = as.numeric(ifelse(P2_omega_total == ">200", 200, P2_omega_total)),
  #Mean haematobium eggs/10mL
  s_haem_mean2016 = mean(c(P1_omega_total, P2_omega_total)),
  #Mean haematobium eggs/10mL NAs removed
  s_haem_mean2016_narm = mean(c(P1_omega_total, P2_omega_total), na.rm = TRUE),
  #Number of samples contributing to mean
  s_haem_mean2016_w = length(which(!is.na(c(P1_omega_total, P2_omega_total)))),
  #WHO infection intensity labels
  s_haem_mean_cat2016 = cut(s_haem_mean2016_narm,
    breaks = c(0, 1, 50, Inf),
    labels = c("No", "Lo", "Hi"), right = FALSE),
  #Mean mansoni epq from 1st kato katz
  s_mans_mean1 = mean(c(P1_kk1, P1_kk2)),
  ##Mean mansoni epq from 2nd kato katz
  s_mans_mean2 = mean(c(P2_kk1, P2_kk2)),
  #Mean mansoni epq from 1st/2nd kato katz
  s_mans_mean2016 = mean(c(P1_kk1, P1_kk2, P2_kk1, P2_kk2)),
  #Mean mansoni EPG NAs removed
  s_mans_mean2016_narm = mean(c(P1_kk1, P1_kk2, P2_kk1, P2_kk2), na.rm = TRUE),
  #Number of non-NA measurements
  s_mans_mean2016_w = length(which(!is.na(c(P1_kk1, P1_kk2, P2_kk1, P2_kk2)))),
  #WHO infection levels
  s_mans_mean_cat2016 = cut(s_mans_mean2016_narm,
    breaks = c(0,1,100,400,Inf),
    labels = c("No", "Lo", "Md", "Hi"), right = FALSE),

  pzq = 0,
  extra_pzq = 0) %>%
select(Child_ID, year, School, pzq, extra_pzq,
  s_haem1 = P1_omega_total,
  s_haem2 = P2_omega_total,
  s_haem_mean = s_haem_mean2016,
  s_haem_mean_narm = s_haem_mean2016_narm,
  s_haem_mean_w = s_haem_mean2016_w,
  s_haem_mean_cat = s_haem_mean_cat2016,
  s_mans1_1 = P1_kk1,
  s_mans1_2 = P1_kk2,
  s_mans2_1 = P2_kk1,
  s_mans2_2 = P2_kk2,
  s_mans_mean1,
  s_mans_mean2,
  s_mans_mean12 = s_mans_mean2016,
  s_mans_mean_narm = s_mans_mean2016_narm,
  s_mans_mean_w = s_mans_mean2016_w,
  s_mans_mean_cat = s_mans_mean_cat2016) %>% ungroup()

```

*#Data check*

*#make sure each subject only has one record:*

```

bad_id_dat1 <- dat1 %>%
  group_by(Child_ID) %>%
  summarise(nobs = n()) %>%
  filter(nobs > 1) %>%
  pull(Child_ID)

```

```
length(bad_id_dat1)
```

## [1] 0

```
dat2 <- read_csv("~/RemaisWork/Schisto/Stanford/Human_Parasitology_data/Second_round_human_subject_analysis_AP")
```

```
# Get means from different samples, rename variables, and reorder
```

```
dat2 <- dat2 %>%
  group_by(Child_ID) %>%
  mutate(year = 2017, School = as.character(School),
    # Replace >200 with 200 then coerce to numeric
    P4_omega_total = as.numeric(ifelse(P4_omega_total == ">200", 200, P4_omega_total)),
    #Mean haematobium eggs/10mL
    s_haem_mean2017 = mean(c(P3_omega_total, P4_omega_total)),
    #Mean haematobium eggs/10mL NAs removed
    s_haem_mean2017_narm = mean(c(P3_omega_total, P4_omega_total), na.rm = TRUE),
    #Number of samples contributing to mean
    s_haem_mean2017_w = length(which(!is.na(c(P3_omega_total, P4_omega_total)))),
    #WHO infection intensity labels
    s_haem_mean_cat2017 = cut(s_haem_mean2017_narm,
      breaks = c(0, 1, 50, Inf),
      labels = c("No", "Lo", "Hi"), right = FALSE),
    #Mean mansoni epq from 1st kato katz
    s_mans_mean1 = mean(c(P3_kk1, P3_kk2)),
    #Mean mansoni epq from 2nd kato katz
    s_mans_mean2 = mean(c(P4_kk1, P4_kk2)),
    #Mean mansoni epq from 1st/2nd kato katz
    s_mans_mean2017 = mean(c(P3_kk1, P3_kk2, P4_kk1, P4_kk2)),
    #Mean mansoni epq NAs removed
    s_mans_mean2017_narm = mean(c(P3_kk1, P3_kk2, P4_kk1, P4_kk2), na.rm = TRUE),
    #Number of non-NA measurements
    s_mans_mean2017_w = length(which(!is.na(c(P3_kk1, P3_kk2, P4_kk1, P4_kk2)))),
    #Number of samples contributing to mean
    s_mans_mean_cat2017 = cut(s_mans_mean2017_narm,
      breaks = c(0,1,100,400, Inf),
      labels = c("No", "Lo", "Md", "Hi"), right = FALSE),

    pzq = 1,
    extra_pzq = as.numeric(!is.na(PZQ_date_before_analyses))) %>%
  select(Child_ID, year, School, pzq, extra_pzq,
    s_haem1 = P3_omega_total,
    s_haem2 = P4_omega_total,
    s_haem_mean = s_haem_mean2017,
    s_haem_mean_narm = s_haem_mean2017_narm,
    s_haem_mean_w = s_haem_mean2017_w,
    s_haem_mean_cat = s_haem_mean_cat2017,
    s_mans1_1 = P3_kk1,
    s_mans1_2 = P3_kk2,
    s_mans2_1 = P4_kk1,
    s_mans2_2 = P4_kk2,
    s_mans_mean1,
    s_mans_mean2,
    s_mans_mean12 = s_mans_mean2017,
    s_mans_mean_narm = s_mans_mean2017_narm,
    s_mans_mean_w = s_mans_mean2017_w,
    s_mans_mean_cat = s_mans_mean_cat2017) %>% ungroup()
```

```
#Data check
```

```
#make sure each subject only has one record:
```

```
bad_id_dat2 <- dat2 %>%
  group_by(Child_ID) %>%
  summarise(nobs = n()) %>%
  filter(nobs > 1) %>%
  pull(Child_ID)
```

```

length(bad_id_dat2)

## [1] 0

dat3 <- read_csv("~/RemaisWork/Schisto/Stanford/Human_Parasitology_data/Third_round_human_subject_analysis_APL")
# Replace **_ numbers with just number and coerce to numeric
dat3$P5_omega_total <- as.numeric(sub(".*", "", dat3$P5_omega_total, fixed = TRUE))
dat3$P6_omega_total <- sub(".*", "*", dat3$P6_omega_total, fixed = TRUE)
dat3$P6_omega_total <- as.numeric(sub(".*", "", dat3$P6_omega_total, fixed = TRUE))

# Get means from different samples, rename variables, and reorder
dat3 <- dat3 %>%
  group_by(Child_ID) %>%
  mutate(year = 2018,
    School = strsplit(Child_ID, "/")[[1]][1],
    #Mean haematobium eggs/10mL
    s_haem_mean2018 = mean(c(P5_omega_total, P6_omega_total)),
    #Mean haematobium eggs/10mL, NAs removed
    s_haem_mean2018_narm = mean(c(P5_omega_total, P6_omega_total), na.rm = TRUE),
    #Number of samples contributing to mean
    s_haem_mean2018_w = length(which(!is.na(c(P5_omega_total, P6_omega_total)))),
    #WHO infection intensity labels
    s_haem_mean_cat2018 = cut(s_haem_mean2018_narm,
      breaks = c(0, 1, 50, Inf),
      labels = c("No", "Lo", "Hi"), right = FALSE),
    #Mean mansoni epq from 1st kato katz
    s_mans_mean1 = mean(c(P5_kk1_epg, P5_kk2_epg)),
    ##Mean mansoni epq from 2nd kato katz
    s_mans_mean2 = mean(c(P6_kk1_epg, P6_kk2_epg)),
    #Mean mansoni epq from 1st/2nd kato katz
    s_mans_mean2018 = mean(c(P5_kk1_epg, P5_kk2_epg, P6_kk1_epg, P6_kk2_epg)),
    #Mean mansoni epq from 1st/2nd kato katz, NAs removed
    s_mans_mean2018_narm = mean(c(P5_kk1_epg, P5_kk2_epg, P6_kk1_epg, P6_kk2_epg), na.rm = TRUE),
    #Number of samples contributing to mean
    s_mans_mean2018_w = length(which(!is.na(c(P5_kk1_epg, P5_kk2_epg, P6_kk1_epg, P6_kk2_epg)))),
    #WHO infection categories
    s_mans_mean_cat2018 = cut(s_mans_mean2018_narm,
      breaks = c(0, 1, 100, 400, Inf),
      labels = c("No", "Lo", "Md", "Hi"), right = FALSE),
    pzq = 1,
    #Individuals who received additional praziquantel influencing their 2018 egg burden
    extra_pzq = ifelse(Child_ID %in% c("DT/1/008", "DT/1/021", "DT/1/032",
      "MD/3/015", "MD/3/023",
      "MQ/2/013", "MQ/2/014", "MQ/3/030"), 1, 0)) %>%
  select(Child_ID, year, School, pzq, extra_pzq,
    s_haem1 = P5_omega_total,
    s_haem2 = P6_omega_total,
    s_haem_mean = s_haem_mean2018,
    s_haem_mean_narm = s_haem_mean2018_narm,
    s_haem_mean_w = s_haem_mean2018_w,
    s_haem_mean_cat = s_haem_mean_cat2018,
    s_mans1_1 = P5_kk1_epg,
    s_mans1_2 = P5_kk2_epg,
    s_mans2_1 = P6_kk1_epg,
    s_mans2_2 = P6_kk2_epg,
    s_mans_mean1,
    s_mans_mean2,
    s_mans_mean12 = s_mans_mean2018,

```

```

s_mans_mean_narm = s_mans_mean2018_narm,
s_mans_mean_w = s_mans_mean2018_w,
s_mans_mean_cat = s_mans_mean_cat2018) %>% ungroup()

```

*#Data check*

*#make sure each subject only has one record:*

```

bad_id_dat3 <- dat3 %>%
  group_by(Child_ID) %>%
  summarise(nobs = n()) %>%
  filter(nobs > 1) %>%
  pull(Child_ID)

```

```
bad_id_dat3
```

```
## [1] "LR/1/053"
```

*#This person ended up with two records somehow, going to take the safe route: remove both records*

```
dat3 <- dat3 %>% filter(Child_ID != enquo(bad_id_dat3)[[2]])
```

*#Full dataset (in long format) shared by Stanford, constructed by Isabel*

```
stanford <- read_csv("~/RemaisWork/Schisto/Stanford/Human_Parasitology_data/human_data_2018Mat1518.csv") %>%
  mutate(ID = factor(ID),      #Convert some variables to factors
    School = factor(School),
    Village = factor(Village),
    year_fac = factor(year),
    cluster = factor(cluster),
    Intervention.type = factor(Intervention.type),
    pzq = ifelse(year %in% c(2017, 2018), 1, 0))

```

*#same code as above to remove individual with multiple observation in the same year*

```
stanford <- stanford %>% filter(ID != enquo(bad_id_dat3)[[2]])
```

*#Jason's full dataset*

```
jason_dat <- read_csv("~/RemaisWork/Schisto/Stanford/Human_Parasitology_data/Jason_human_data_2018.csv") %>%
  mutate(ID = factor(ID),      #Convert some variables to factors
    School = factor(School),
    Village = factor(Village),
    year_fac = factor(year),
    sex = factor(sex),
    cluster = factor(cluster),
    Intervention.type = factor(Intervention.type),
    pzq = ifelse(year %in% c(2017, 2018), 1, 0))

```

*#Full dataset in long format from data reads/manipulations above*

```
full_long <- rbind(dat1, dat2, dat3) %>%
  full_join(interventions, by = c("School", "year")) %>%
  #Join with stanford dataset to get cluster and sex
  full_join(stanford %>% select(ID, year, sex, cluster),
    by = c("Child_ID" = "ID", "year" = "year")) %>%
  mutate(School = factor(School),
    Child_ID = factor(Child_ID),
    year_fac = factor(year),
    year_2016 = ifelse(year == 2016, 1, 0),
    year_2017 = ifelse(year == 2017, 1, 0),
    year_2018 = ifelse(year == 2018, 1, 0),
    pzq = factor(pzq))

```

*#Full community-level dataset in long format*

```

full_long_comm <- full_long %>% #filter(extra_pzq != 1) %>%
  group_by(School, year) %>%
  summarise(haem_mean = mean(c(s_haem1, s_haem2), na.rm = TRUE),
            haem_samps = length(na.omit(c(s_haem1, s_haem2))),
            haem_pos = length(which(s_haem_mean_narm > 0)),
            haem_prev = haem_pos / length(na.omit(s_haem_mean_narm)),
            mans_mean = mean(c(s_mans1_1, s_mans1_2,
                               s_mans2_1, s_mans2_2), na.rm = TRUE),
            mans_samps = length(na.omit(c(s_mans1_1, s_mans1_2,
                                           s_mans2_1, s_mans2_2))),
            mans_pos = length(which(s_mans_mean_narm > 0)),
            mans_prev = mans_pos / length(na.omit(s_mans_mean_narm)),
            extra_pzq = mean(extra_pzq)) %>%
  inner_join(interventions, by = c("School", "year")) %>%
  group_by(School) %>%
  mutate(vil_haem_mean = mean(haem_mean),
         vil_mans_mean = mean(mans_mean)) %>%
  ungroup() %>%
  mutate(vil_haem_diff = haem_mean - vil_haem_mean,
         vil_mans_diff = mans_mean - vil_mans_mean) %>%
  arrange(year, School)

#Full dataset in wide format
full_wide <- dat1 %>%
  full_join(dat2 %>% filter(extra_pzq != 1),
            by = c("Child_ID", "School"),
            suffix = c("_2016", "_2017")) %>%
  full_join(dat3, by = c("Child_ID", "School")) %>%
  full_join(interventions %>% filter(year == 2018), by = c("School"))

```

## Exploratory

### Check egg burden distribution

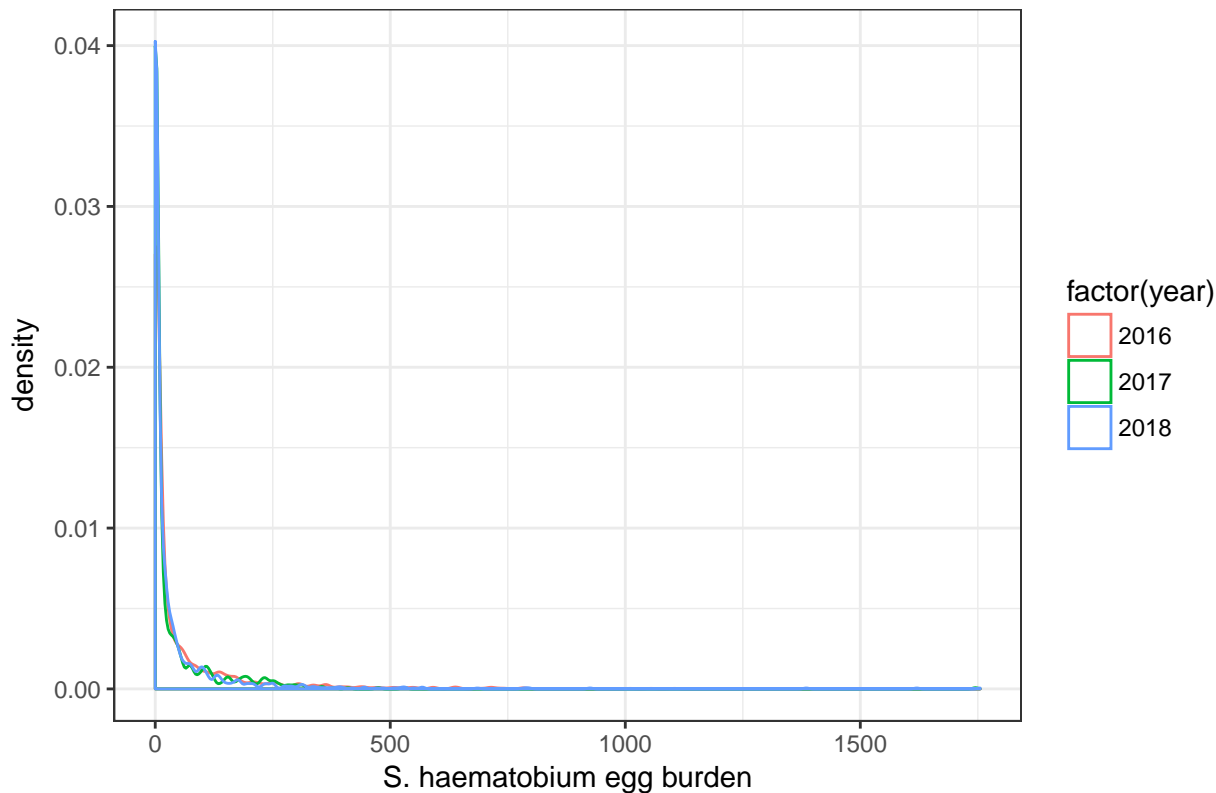
```

full_long %>% filter(extra_pzq != 1) %>%
  ggplot(aes(x = s_haem_mean_narm, col = factor(year))) + geom_density() +
  theme_bw() + xlab("S. haematobium egg burden") + ggtitle("S. haematobium yearly egg burden dist'n")

```



## S. haematobium yearly egg burden dist'n



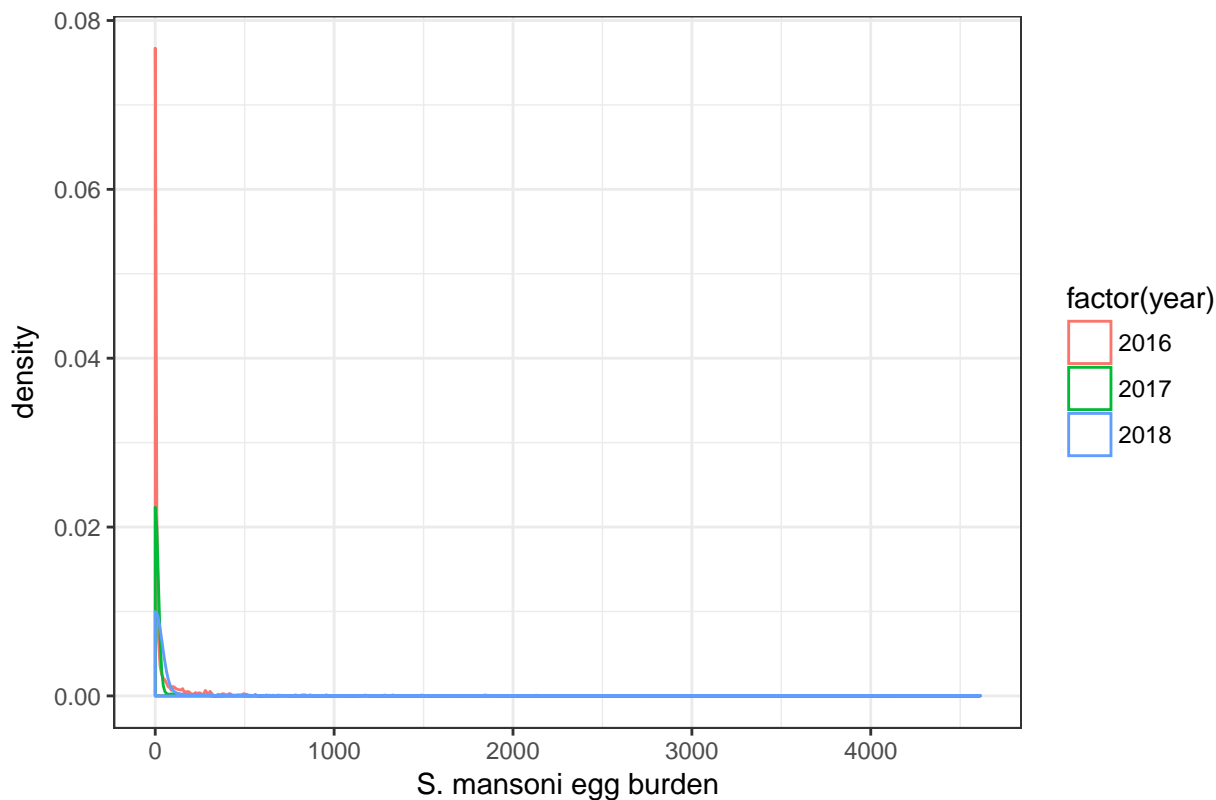
```
full_long %>% filter(extra_pzq != 1) %>%
  group_by(year, School) %>%
  summarise(mean_haem = mean(s_haem_mean_narm),
            var_haem = var(s_haem_mean_narm),
            mean_haem_rmv0 = mean(s_haem_mean_narm[which(s_haem_mean_narm != 0)]),
            var_haem_rmv0 = var(s_haem_mean_narm[which(s_haem_mean_narm != 0)]))
```

```
## # A tibble: 44 x 6
## # Groups:   year [?]
##   year School mean_haem var_haem mean_haem_rmv0 var_haem_rmv0
##   <dbl> <fct>    <dbl>    <dbl>         <dbl>         <dbl>
## 1 2016. DF      7.38      289.         10.4          379.
## 2 2016. DT     66.1    17172.        73.3        18529.
## 3 2016. FS     38.4    6952.         42.5         7544.
## 4 2016. GG     45.0    4629.         53.7         5063.
## 5 2016. GK     52.2   10327.         63.9        11915.
## 6 2016. LR     NaN      NaN          78.3        15122.
## 7 2016. MA    111.    28788.        112.        28991.
## 8 2016. MB      9.29     577.         18.1          971.
## 9 2016. MD     84.1   17075.         95.0        18266.
## 10 2016. ME     91.0   31524.        107.        35288.
## # ... with 34 more rows
```

```
full_long %>% filter(extra_pzq != 1) %>%
  ggplot(aes(x = s_mans_mean12, col = factor(year))) + geom_density() + theme_bw() +
  xlab("S. mansoni egg burden") + ggtitle("S. mansoni yearly egg burden dist'n")
```



## S. mansoni yearly egg burden dist'n



```
full_long %>% filter(extra_pzq != 1) %>%
  group_by(year, School) %>%
  summarise(mean_mans = mean(s_mans_mean_narm),
            var_mans = var(s_mans_mean_narm),
            mean_mans_rmv0 = mean(s_mans_mean_narm[which(s_mans_mean_narm != 0)]),
            var_mans_rmv0 = var(s_mans_mean_narm[which(s_mans_mean_narm != 0)]))
```

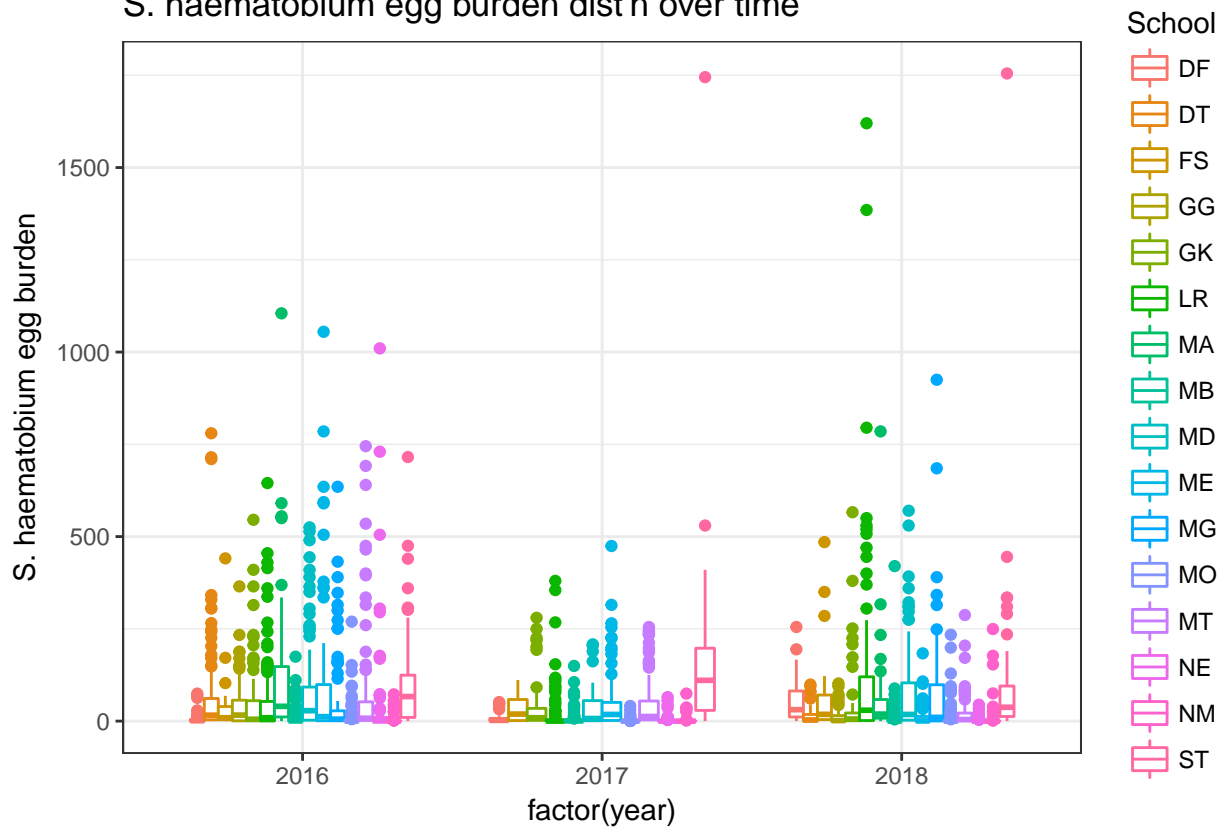
```
## # A tibble: 44 x 6
## # Groups:   year [?]
##   year School mean_mans var_mans mean_mans_rmv0 var_mans_rmv0
##   <dbl> <fct>    <dbl>    <dbl>         <dbl>         <dbl>
## 1 2016. DF      8.25      718.         56.6         2437.
## 2 2016. DT     NaN      NaN          18.0          104.
## 3 2016. FS     0.266      2.20         8.25          NA
## 4 2016. GG    557.    448243.       602.       457359.
## 5 2016. GK      0.        0.          NaN          NA
## 6 2016. LR    130.    274885.       334.       647577.
## 7 2016. MA     NaN      NaN          307.       256249.
## 8 2016. MB     39.1    14983.       103.       33401.
## 9 2016. MD     NaN      NaN          51.6        9415.
## 10 2016. ME     6.49    1504.        37.1       7817.
## # ... with 34 more rows
```

Data is clearly overdispersed. Negative binomial distribution, maybe 0-inflated model will be necessary

## Community level egg burden over time

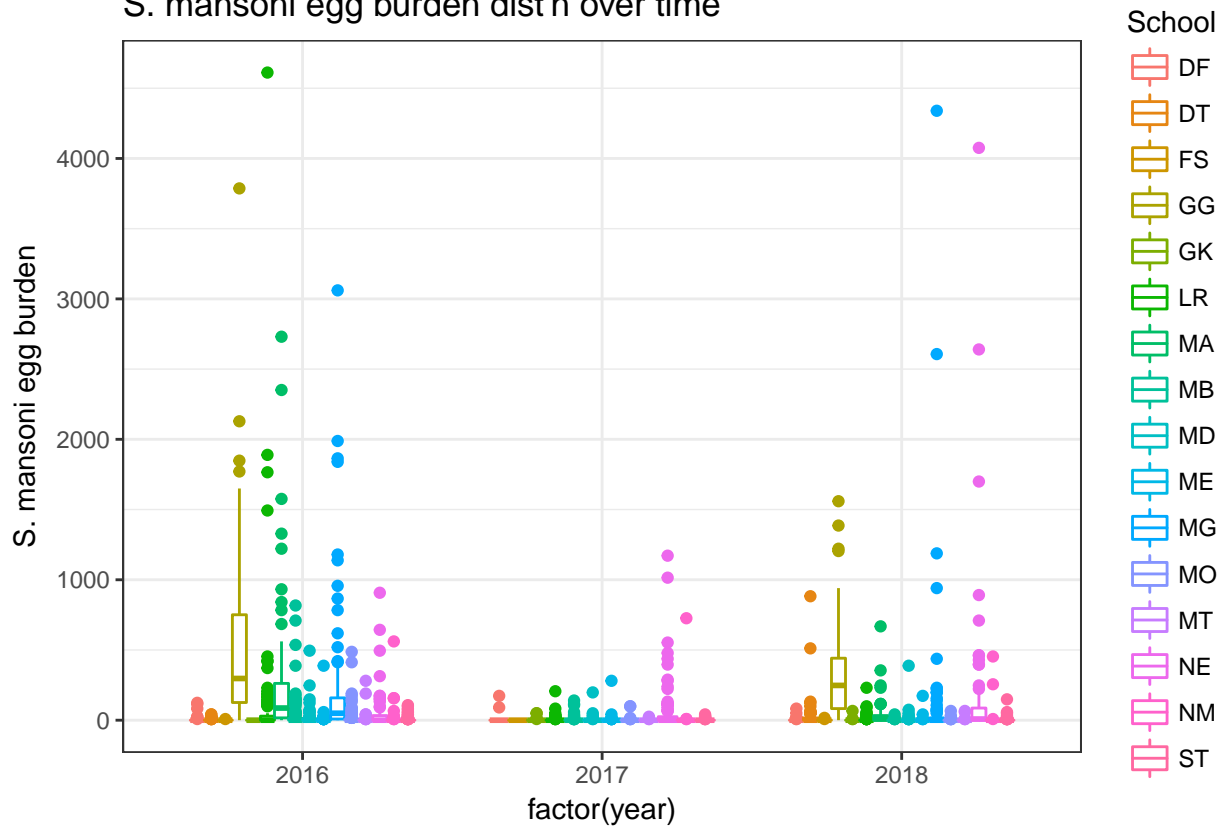
```
full_long %>% filter(extra_pzq != 1) %>%
  ggplot(aes(x = factor(year), y = s_haem_mean_narm, col = School)) + theme_bw() + geom_boxplot() +
  ylab("S. haematobium egg burden") + ggtitle("S. haematobium egg burden dist'n over time")
```

S. haematobium egg burden dist'n over time



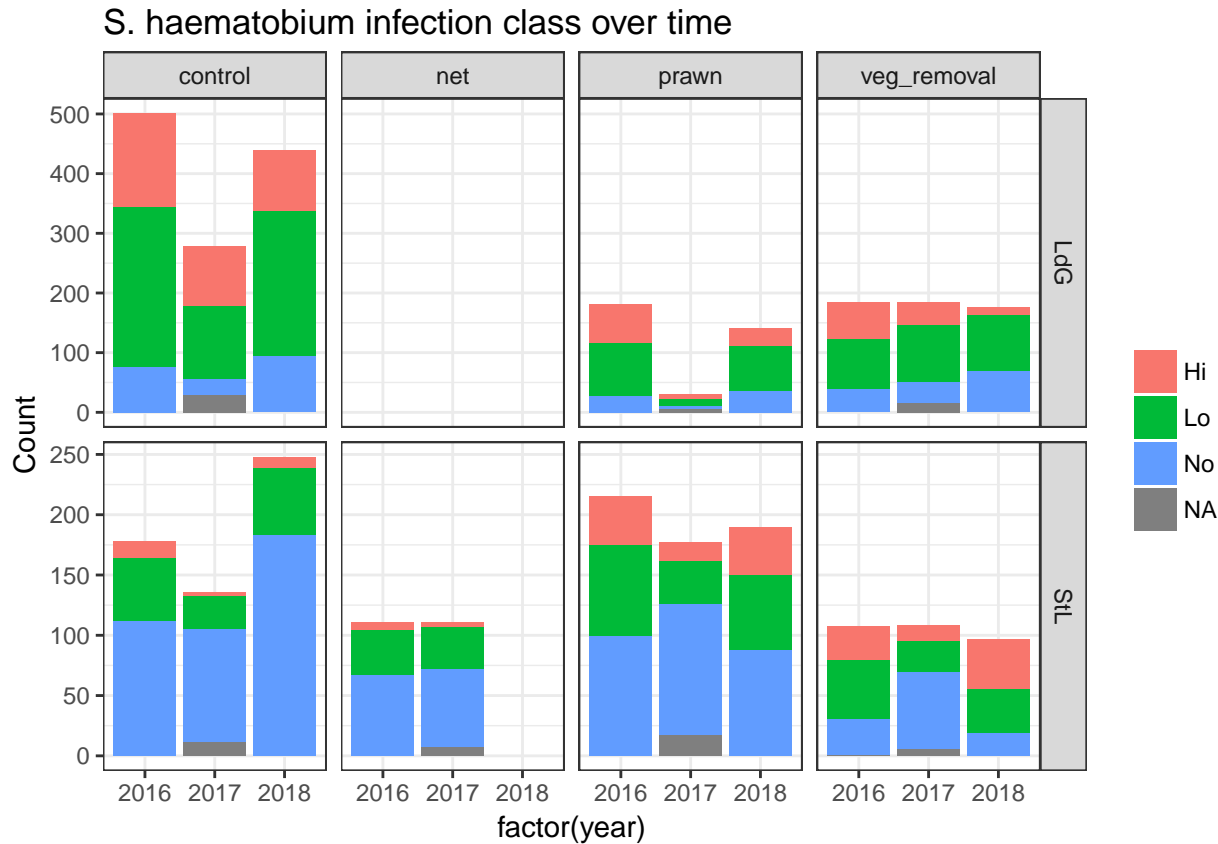
```
full_long %>% filter(extra_pzq != 1) %>%
  ggplot(aes(x = factor(year), y = s_mans_mean_narm, col = School)) + theme_bw() + geom_boxplot() +
  ylab("S. mansoni egg burden") + ggtitle("S. mansoni egg burden dist'n over time")
```

S. mansoni egg burden dist'n over time



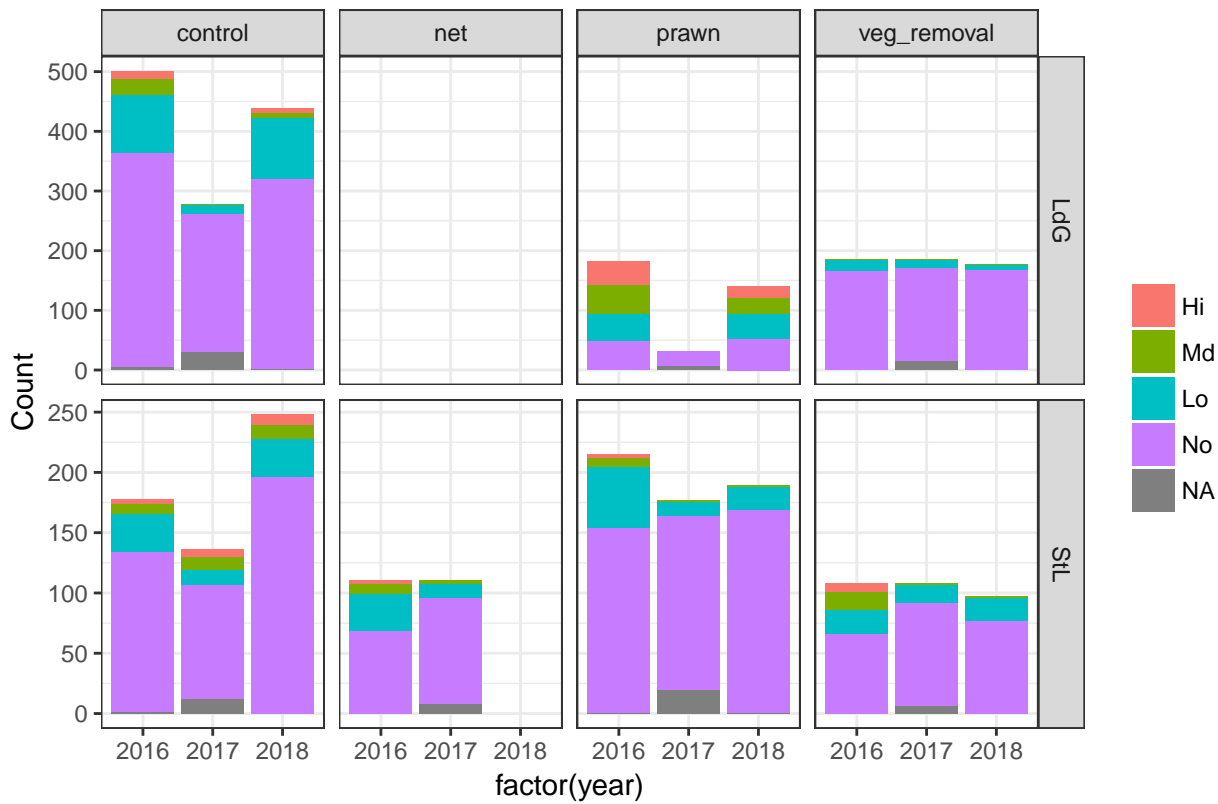
## Intervention grouped infection level (WHO categories) over time

```
full_long %>% filter(extra_pzq != 1) %>%
  ggplot(aes(x = factor(year), fill = factor(s_haem_mean_cat, levels = c("Hi", "Lo", "No")))) +
  geom_bar() +
  theme_bw() + theme(legend.title = element_blank()) +
  ylab("Count") + ggtitle("S. haematobium infection class over time") +
  facet_grid(Region~Intervention_2018, scales = "free_y")
```



```
full_long %>% filter(extra_pzq != 1) %>%
  ggplot(aes(x = factor(year), fill = factor(s_mans_mean_cat, levels = c("Hi", "Md", "Lo", "No")))) +
  geom_bar() +
  theme_bw() + theme(legend.title = element_blank()) +
  ylab("Count") + ggtitle("S. mansoni infection class over time") +
  facet_grid(Region~Intervention_2018, scales = "free_y")
```

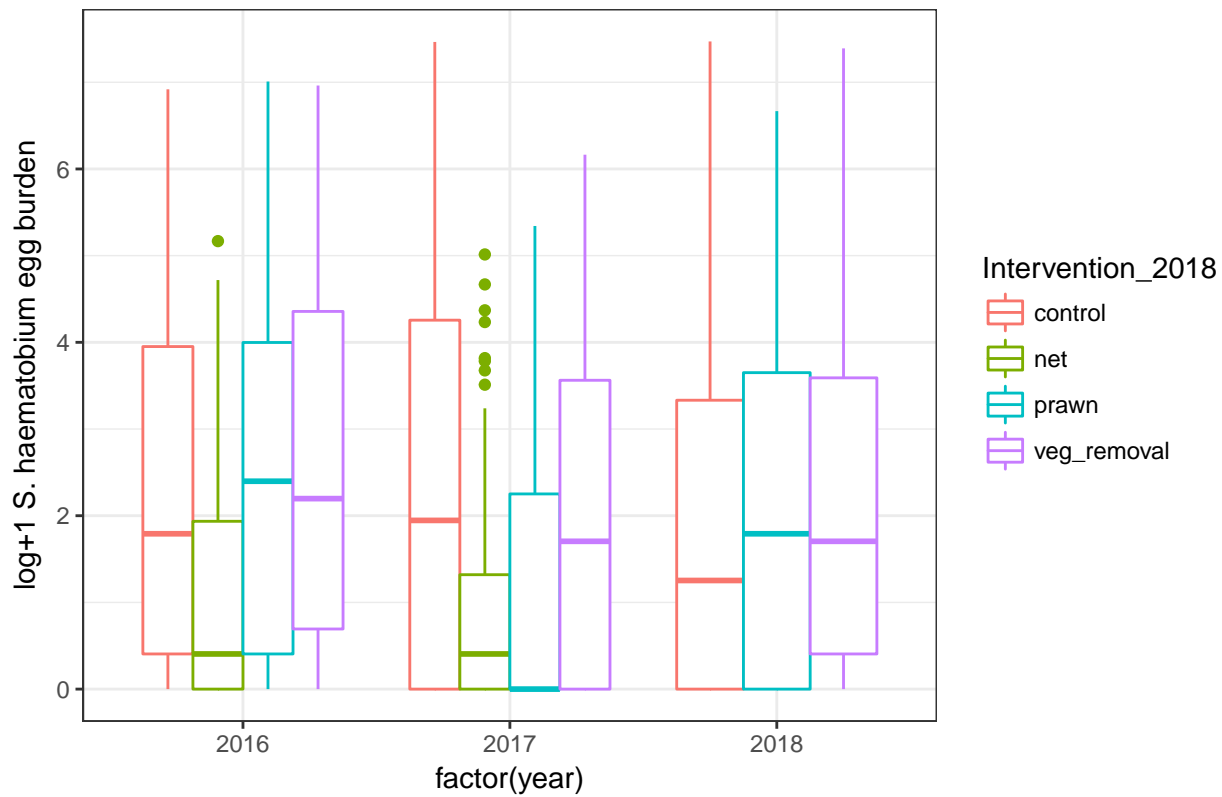
## S. mansoni infection class over time



## Intervention status log+1 egg burden over time

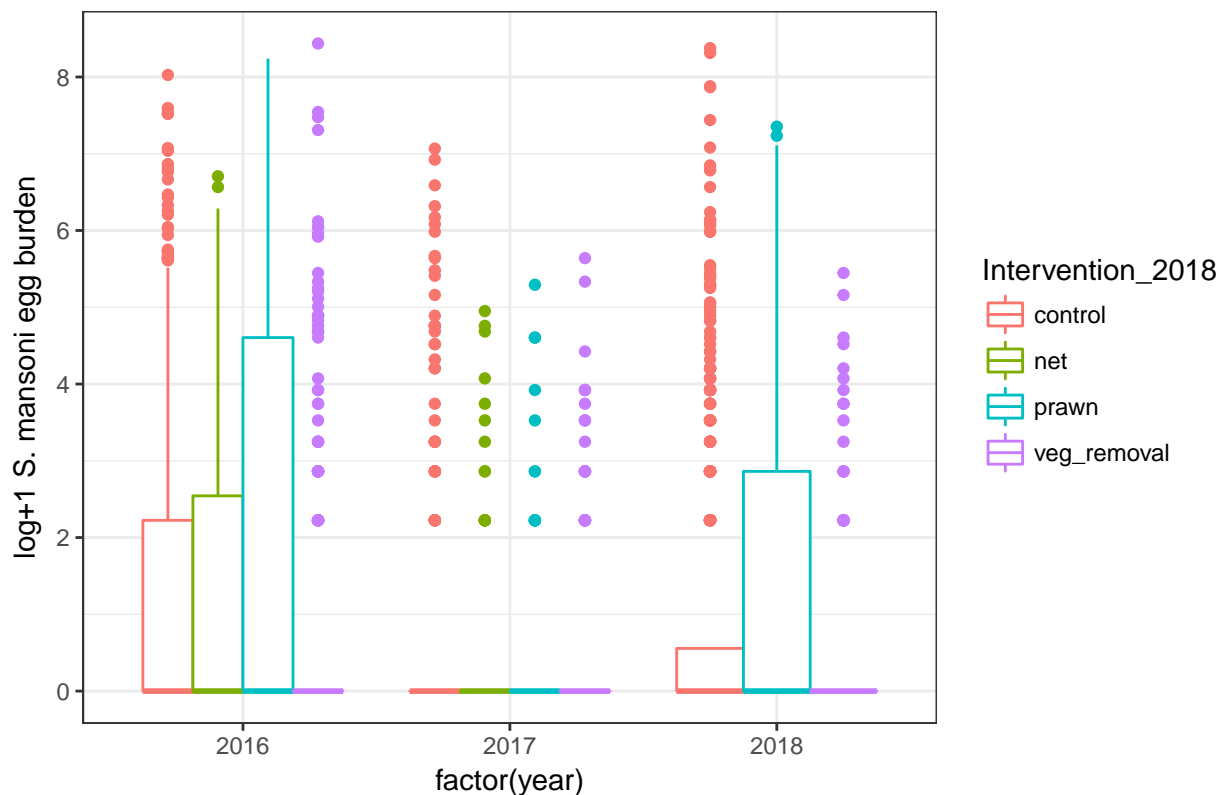
```
full_long %>% filter(extra_pzq != 1) %>%
  ggplot(aes(x = factor(year), y = log(s_haem_mean_narm+1), col = Intervention_2018)) +
  theme_bw() + geom_boxplot() +
  ylab("log+1 S. haematobium egg burden") +
  ggtitle("S. haematobium egg burden dist'n over time")
```

## S. haematobium egg burden dist'n over time



```
full_long %>% filter(extra_pzq != 1) %>%
  ggplot(aes(x = factor(year), y = log(s_mans_mean_narm+1), col = Intervention_2018)) +
  theme_bw() + geom_boxplot() +
  ylab("log+1 S. mansoni egg burden") +
  ggtitle("S. mansoni egg burden dist'n over time")
```

## S. mansoni egg burden dist'n over time

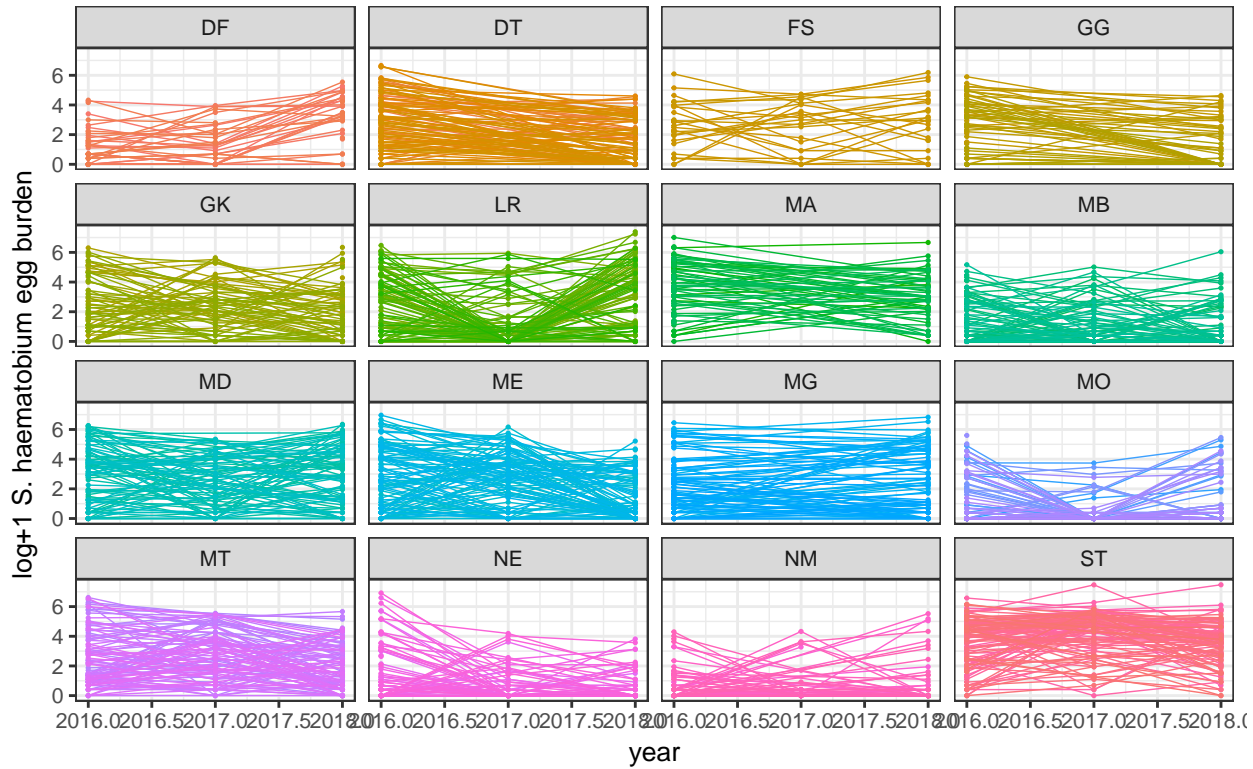


## Individual level mean egg burden over time (spaghetti plots)

```
full_long %>% filter(extra_pzq != 1) %>%
  ggplot(aes(x = year, y = log(s_haem_mean_narm+1), col = Child_ID)) +
  geom_line(size = 0.25) + geom_point(size = 0.25) +
  theme_bw(base_size = 10) + theme(legend.position = "none") +
  facet_wrap(~School, ncol = 4) +
  ylab("log+1 S. haematobium egg burden") +
  ggtitle("Individual egg burden trajectories", subtitle = "S. haematobium")
```

## Individual egg burden trajectories

*S. haematobium*

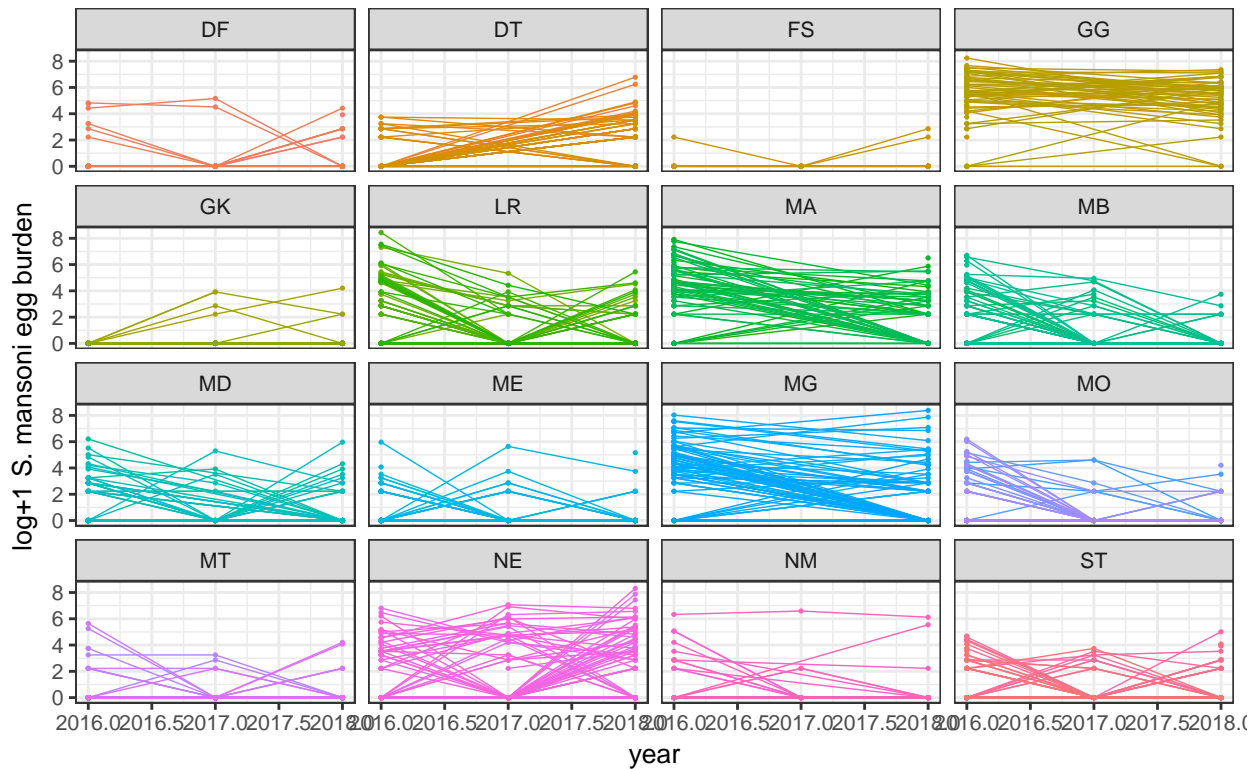


```
full_long %>% filter(extra_pzq != 1) %>%
  ggplot(aes(x = year, y = log(s_mans_mean_narm+1), col = Child_ID)) +
  geom_line(size = 0.25) + geom_point(size = 0.25) +
  theme_bw(base_size = 10) + theme(legend.position = "none") +
  facet_wrap(~School, ncol = 4) +
  ylab("log+1 S. mansoni egg burden") +
  ggtitle("Individual egg burden trajectories", subtitle = "S. mansoni")
```



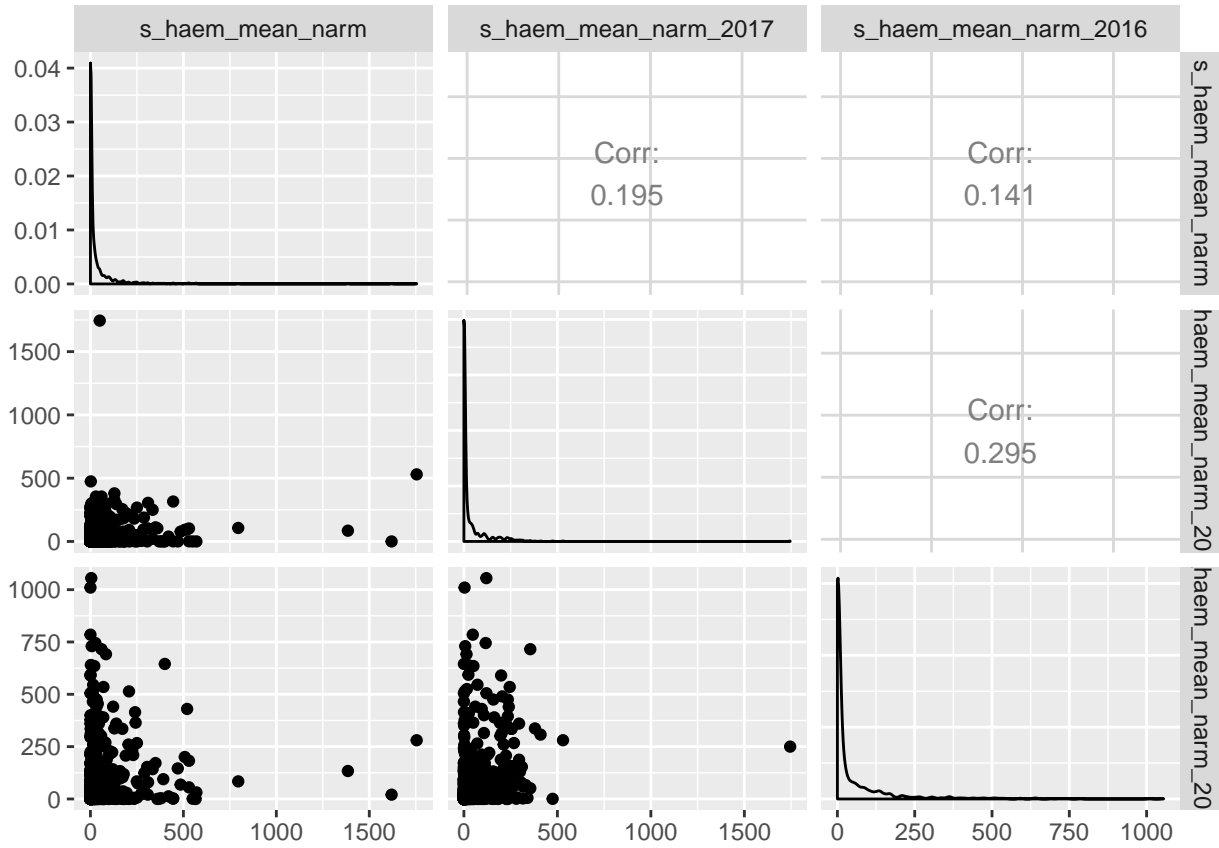
## Individual egg burden trajectories

*S. mansoni*

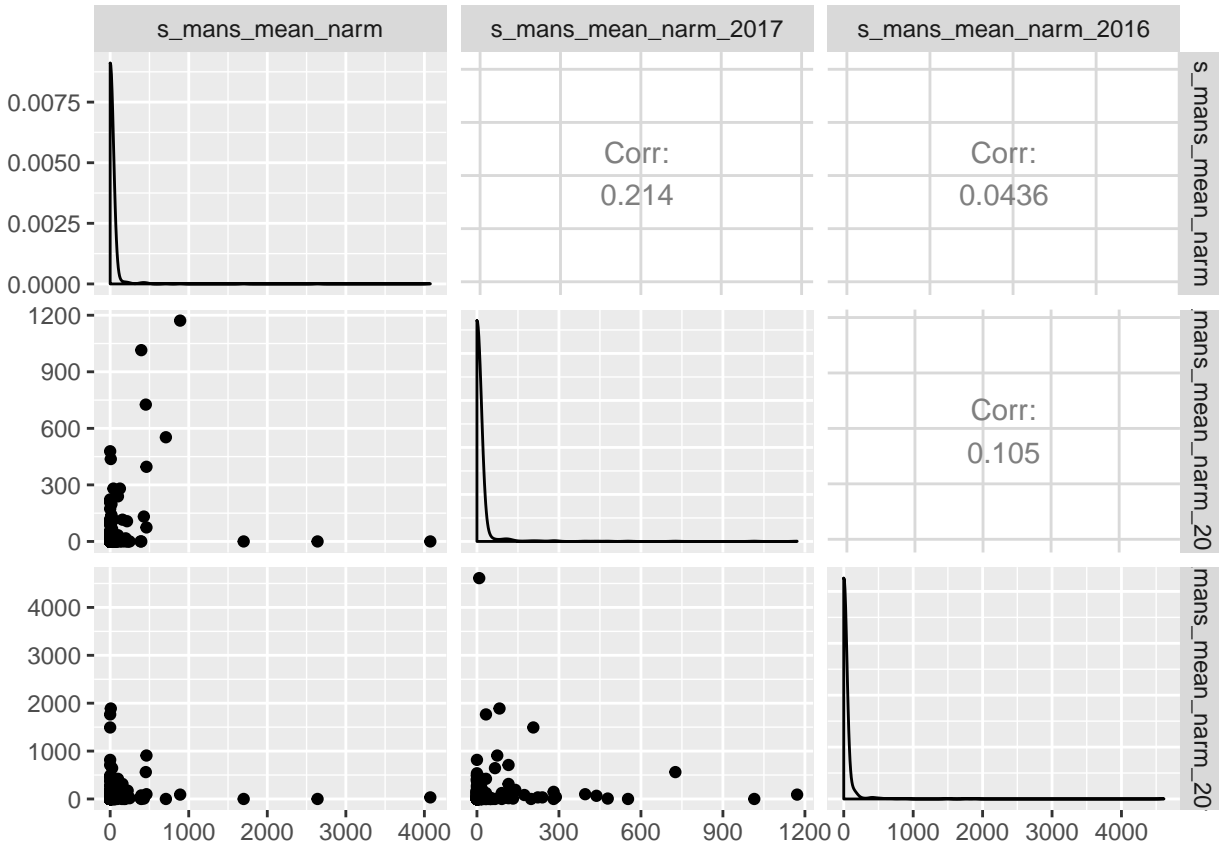


## Correlation between past infection and infection in 2018

```
full_wide %>%
  filter(extra_pzq_2017 != 1) %>%
  select(s_haem_mean_narm, s_haem_mean_narm_2017, s_haem_mean_narm_2016) %>%
  ggpairs()
```

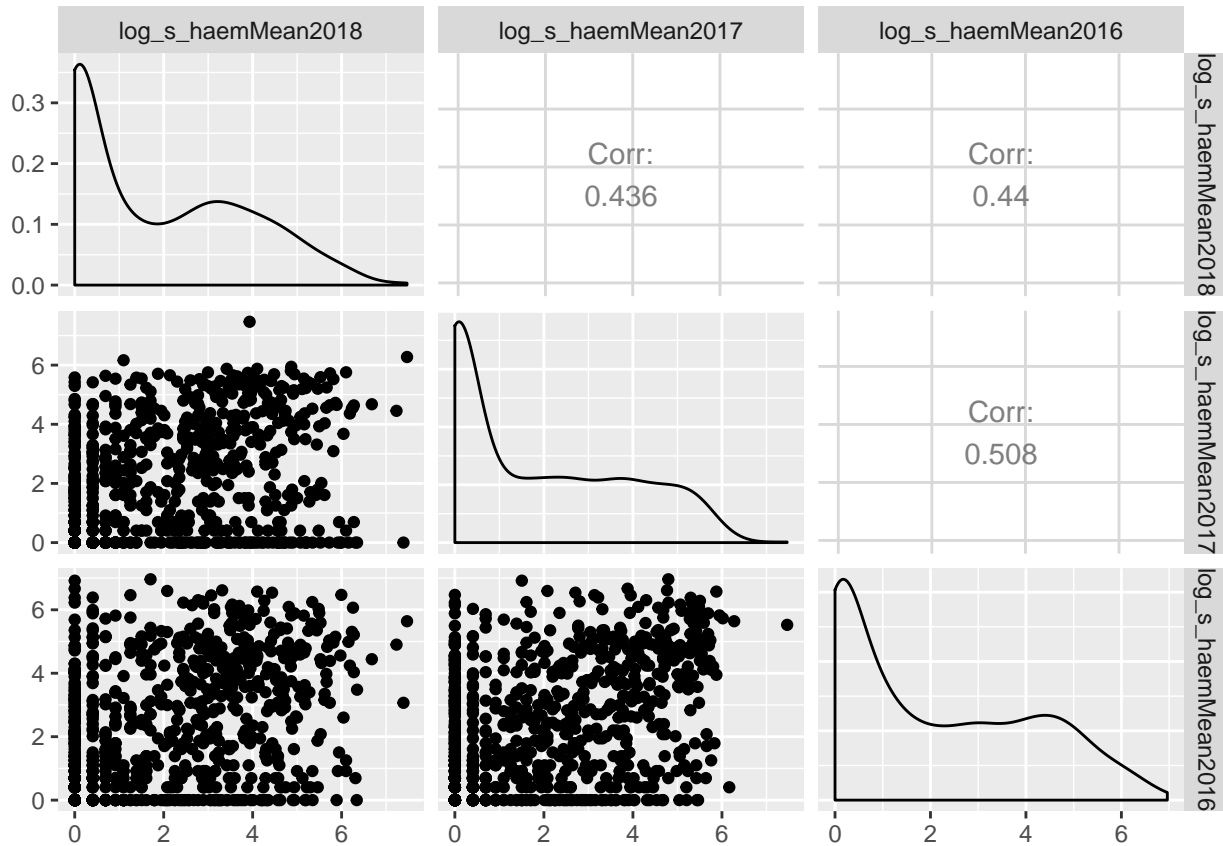


```
full_wide %>%
  filter(extra_pzq_2017 != 1) %>%
  select(s_mans_mean_narm, s_mans_mean_narm_2017, s_mans_mean_narm_2016) %>%
  ggpairs()
```

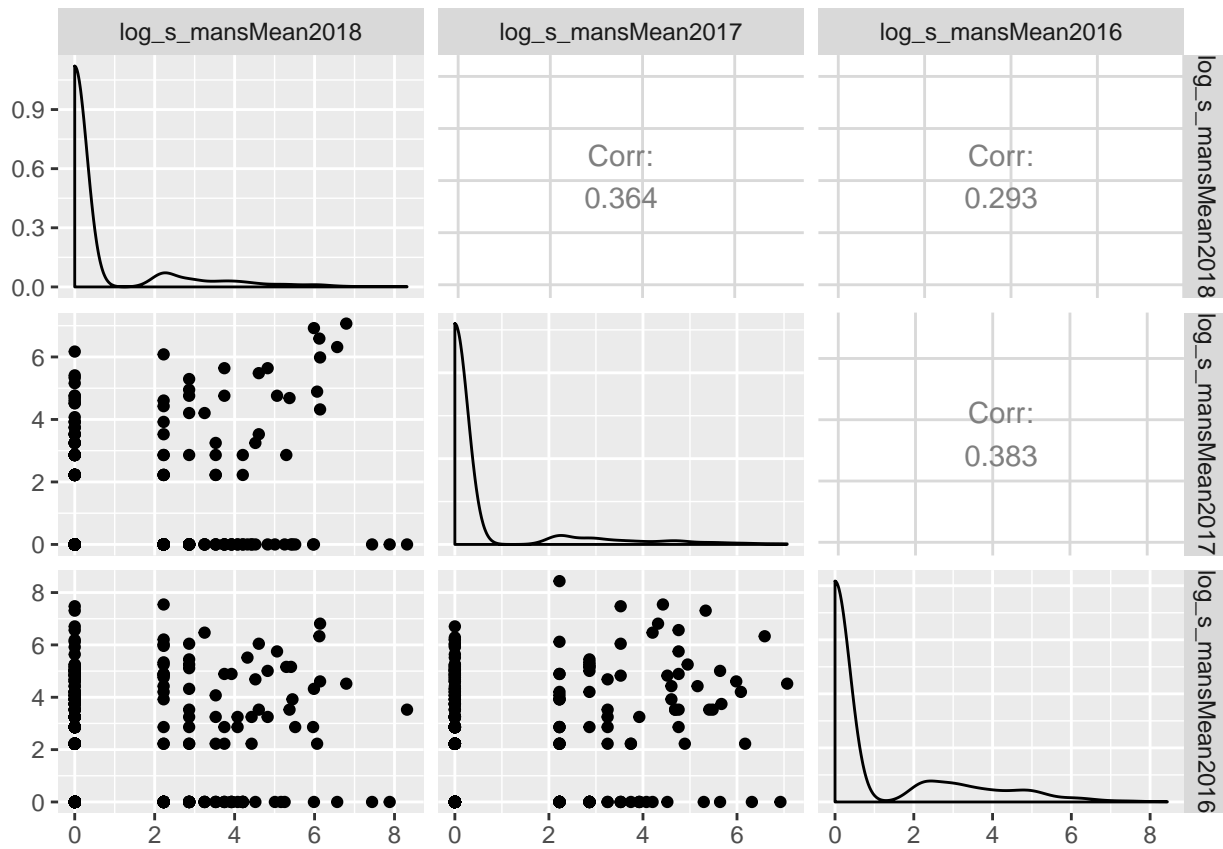


## Correlation between past infection and infection in 2018, log+1 transformed

```
full_wide %>% filter(extra_pzq_2017 != 1) %>%
  mutate(log_s_haemMean2018 = log(s_haem_mean_narm+1),
         log_s_haemMean2017 = log(s_haem_mean_narm_2017+1),
         log_s_haemMean2016 = log(s_haem_mean_narm_2016+1)) %>%
  select(log_s_haemMean2018, log_s_haemMean2017, log_s_haemMean2016) %>%
  ggpairs()
```



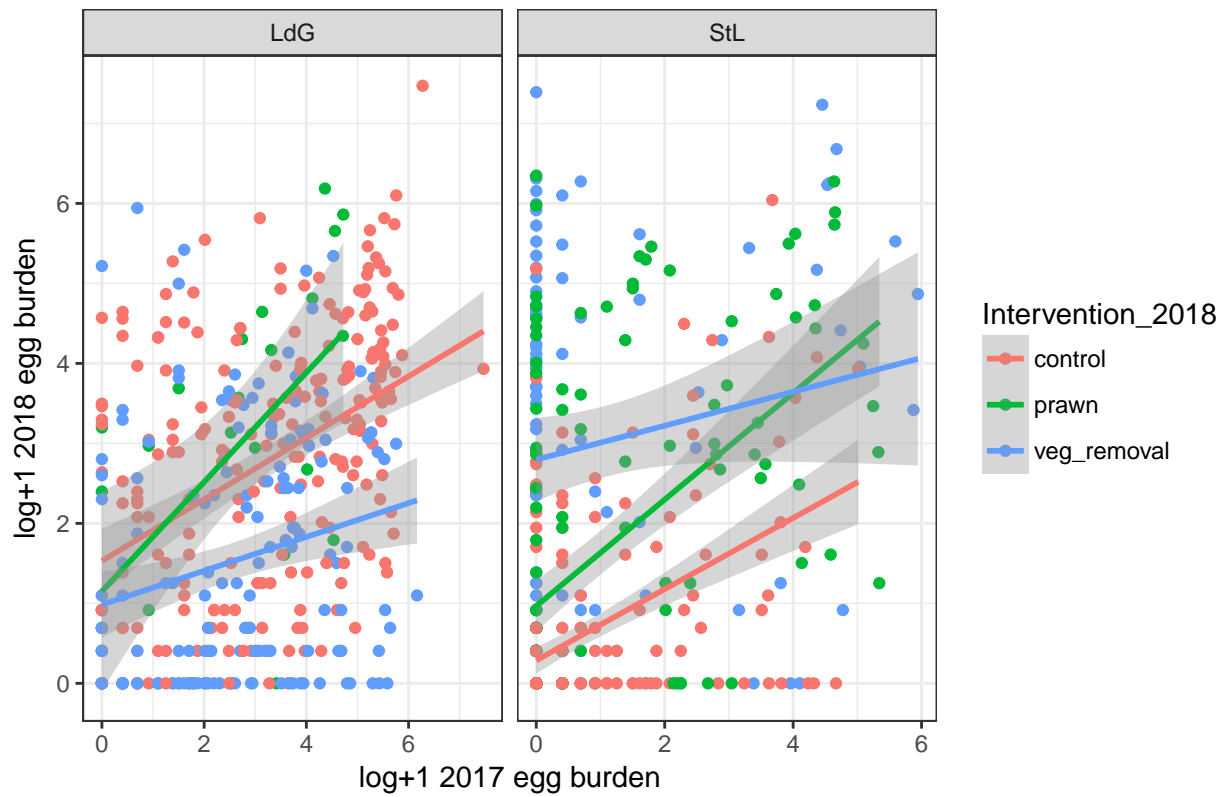
```
full_wide %>% filter(extra_pzq_2017 != 1) %>%
  mutate(log_s_mansMean2018 = log(s_mans_mean_narm+1),
         log_s_mansMean2017 = log(s_mans_mean_narm_2017+1),
         log_s_mansMean2016 = log(s_mans_mean_narm_2016+1)) %>%
  select(log_s_mansMean2018,
         log_s_mansMean2017,
         log_s_mansMean2016) %>%
  ggpairs()
```



Correlations between egg burden in 2017 and egg burden in 2018, stratified by intervention

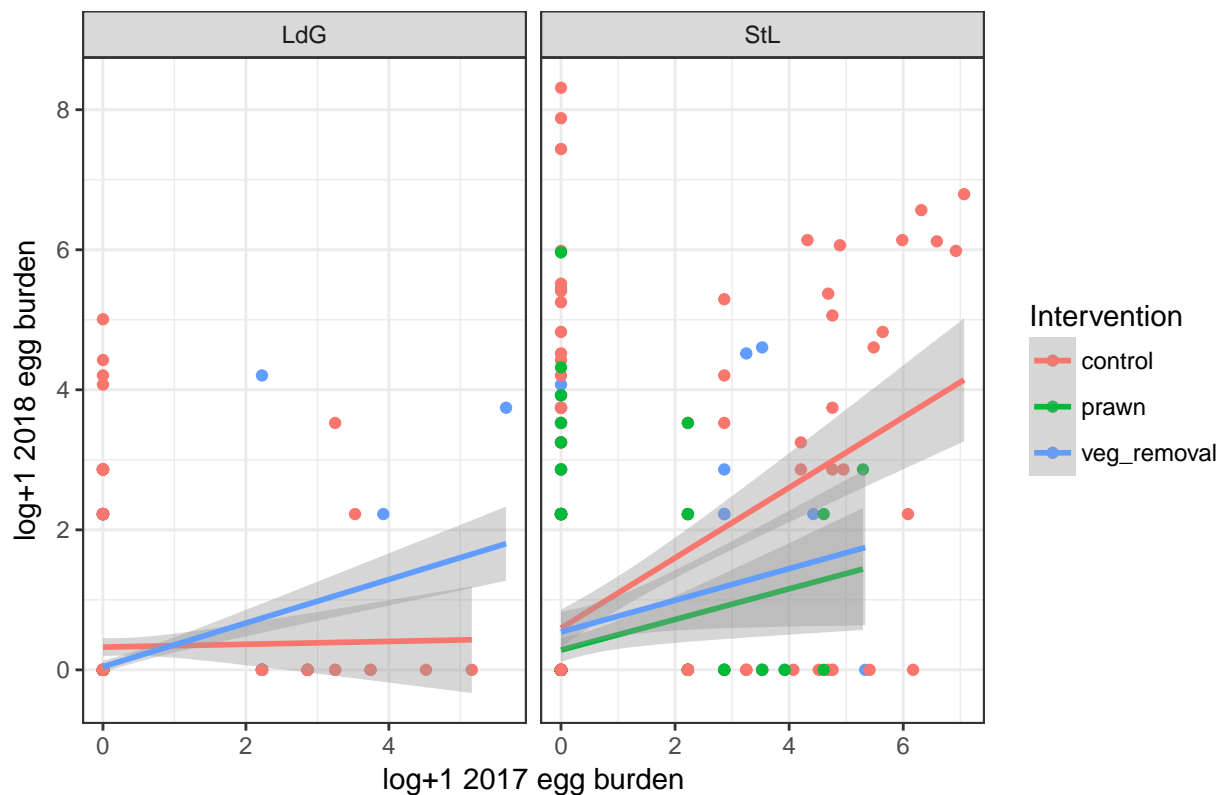
```
full_wide %>% filter(extra_pzq_2017 != 1) %>%
  ggplot(aes(x = log(s_haem_mean_narm_2017+1), y = log(s_haem_mean_narm+1), col = Intervention_2018)) +
  geom_point() + stat_smooth(method = "lm") +
  facet_grid(.~Region, scales = "free") +
  theme_bw() + xlab("log+1 2017 egg burden") + ylab("log+1 2018 egg burden") +
  ggtitle("S. haematobium 2017-2018 egg burden correlation")
```

## S. haematobium 2017–2018 egg burden correlation



```
full_wide %>% filter(extra_pzq_2017 != 1) %>%
  ggplot(aes(x = log(s_mans_mean_narm_2017+1), y = log(s_mans_mean_narm+1), col = Intervention)) +
  geom_point() + stat_smooth(method = "lm") +
  facet_grid(.~Region, scales = "free") +
  theme_bw() + xlab("log+1 2017 egg burden") + ylab("log+1 2018 egg burden") +
  ggtitle("S. mansoni 2017-2018 egg burden correlation")
```

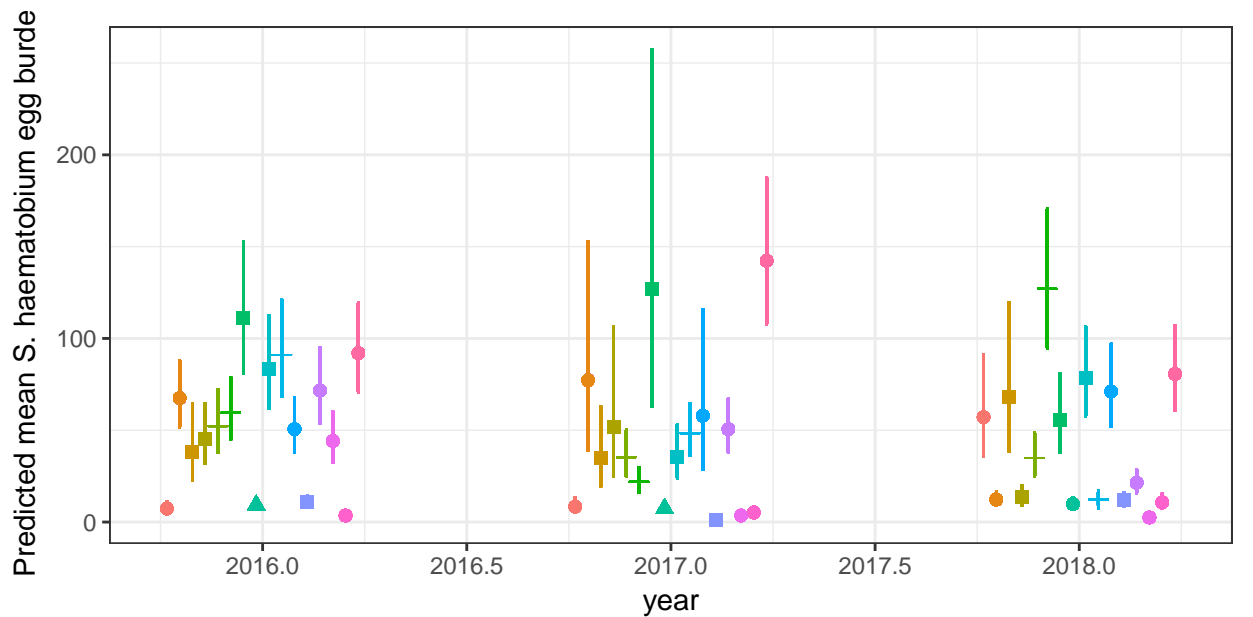
## S. mansoni 2017–2018 egg burden correlation



## Community level mean egg burden across years and communities

```
haem_int_yr <- glm.nb(round(s_haem_mean_narm) ~ year_fac*School,
  weights = s_haem_mean_w,
  data = full_long %>% filter(extra_pzq != 1))

ggpredict(haem_int_yr, c("year_fac", "School")) %>%
  mutate(year = factor(x)) %>%
  full_join(full_long %>% select(year_fac, School, Intervention_2018),
    by = c("year" = "year_fac", "group" = "School")) %>%
  ggplot(aes(x = x, y = predicted, col = group, shape = Intervention_2018)) +
    geom_point(position = position_dodge(width = 0.5), size = 2) +
    geom_errorbar(aes(ymin = conf.low, ymax = conf.high, x = x),
      position = position_dodge(width = 0.5),
      width = 0.1) +
    theme_bw() + theme(legend.position = "bottom") +
    xlab("year") + ylab("Predicted mean S. haematobium egg burden")
```

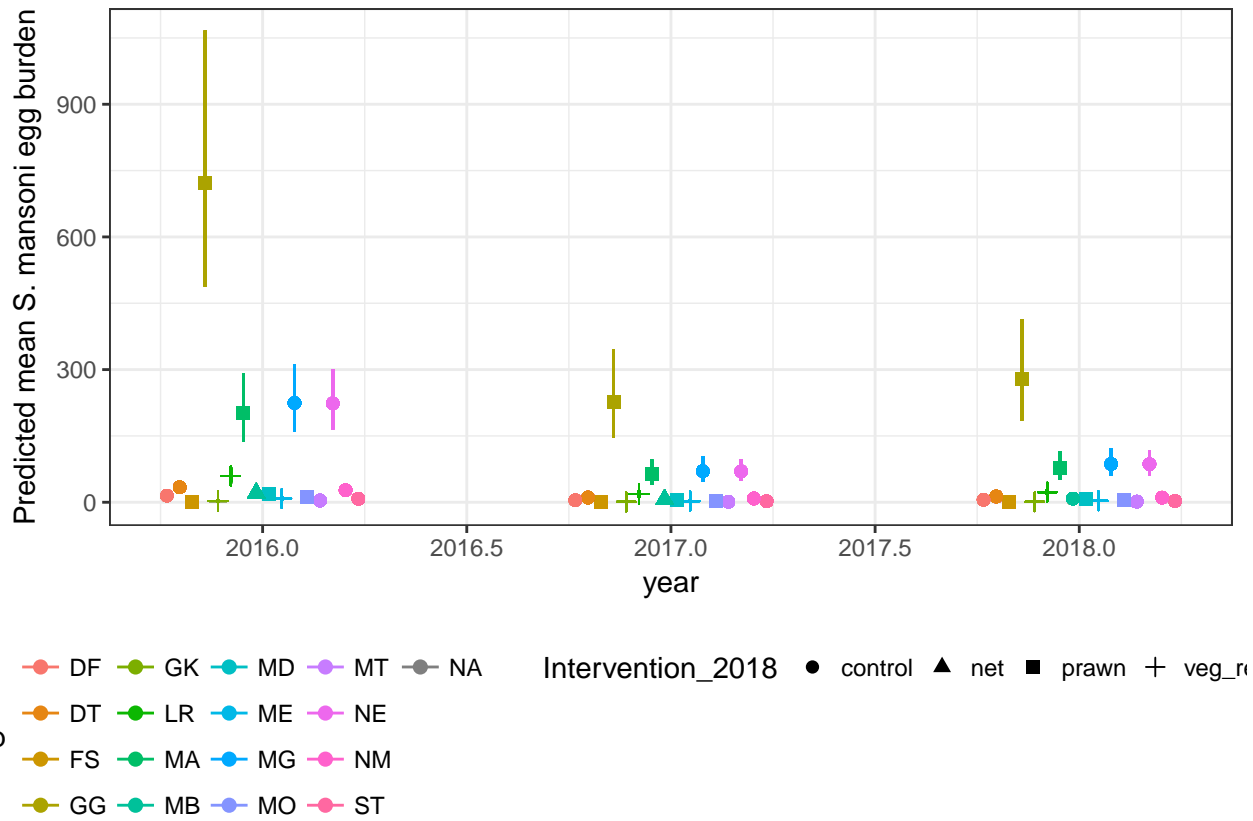


DF GK MD MT NA Intervention\_2018 ● control ▲ net ■ prawn + veg\_r  
 DT LR ME NE  
 FS MA MG NM  
 GG MB MO ST

```
mans_int_yr <- glm.nb(round(s_mans_mean_narm) ~ year_fac+School,
  weights = s_mans_mean_w,
  data = full_long %>% filter(extra_pzq != 1))
```

```
ggpredict(mans_int_yr, c("year_fac", "School")) %>%
mutate(year = factor(x)) %>%
full_join(full_long %>% select(year_fac, School, Intervention_2018),
  by = c("year" = "year_fac", "group" = "School")) %>%
ggplot(aes(x = x, y = predicted, col = group, shape = Intervention_2018)) +
  geom_point(position = position_dodge(width = 0.5), size = 2) +
  geom_errorbar(aes(ymin = conf.low, ymax = conf.high, x = x),
    position = position_dodge(width = 0.5),
    width = 0.1) +
  theme_bw() + theme(legend.position = "bottom") +
  xlab("year") + ylab("Predicted mean S. mansoni egg burden")
```





## Others' models

Sanna: neg binomial egg output w/ individual and community (school) random effects

$$Y_{ijt} = \beta_0 + \epsilon_{0i} + \epsilon_{0j} + \beta_1 year_t + \beta_2 Intervention_{jt} + \beta_3 year_t \times Intervention_{jt} + \beta_4 sex_i$$

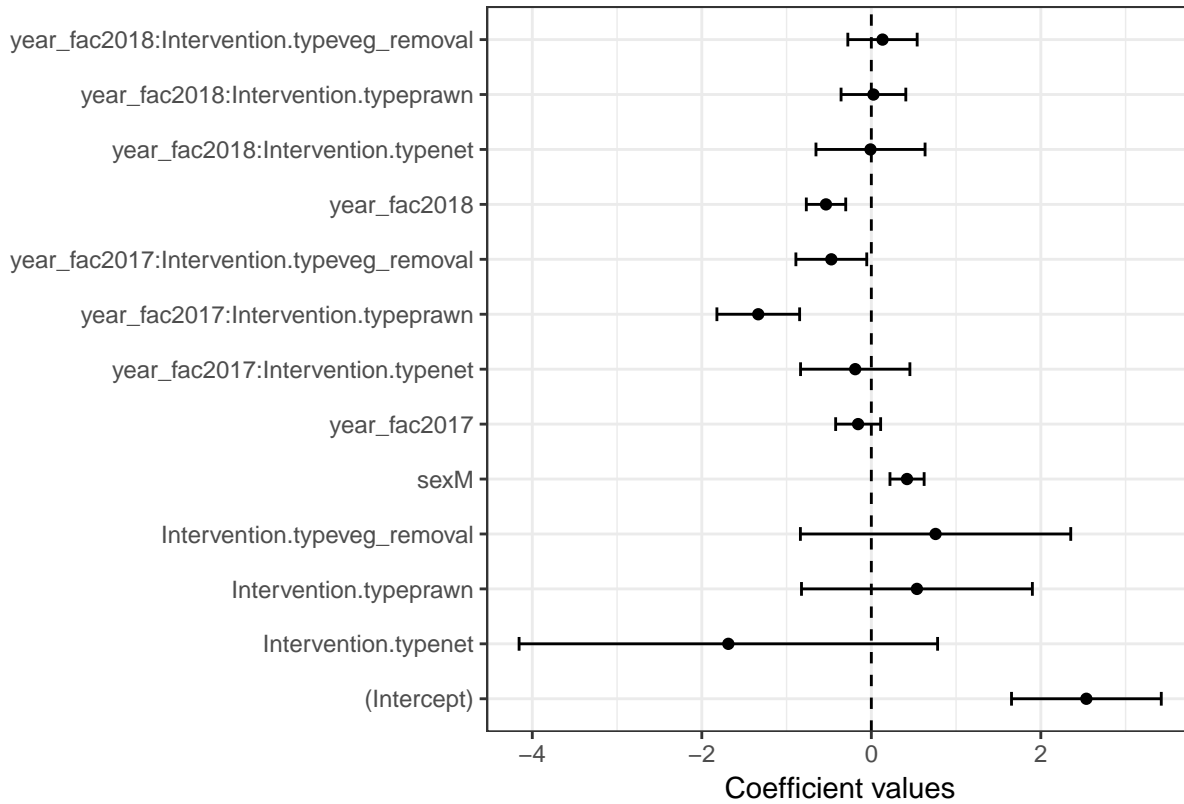
where  $\beta_{0i}$  and  $\beta_{0j}$  are individual and community level random effects

```
#Original code from Sanna's script:
#nbinom.SH<-glmmadmb(Data~(1|ID)+(1|Village)+Label*treatment+sex, data = SH.df, family="nbinom")

#glmmadmb throws an error for me, use updated version, glmmTMB
#Run original model with stanford data:
mods_haem_stan <- glmmTMB(ShW~year_fac*Intervention.type+sex + (1|ID) + (1|School),
  data = stanford %>% filter(Tx_early != 1),
  family="nbinom2")

as.data.frame(round(cbind(summary(mods_haem_stan)$coef[[1]],
  confint(mods_haem_stan)[1:13,]), 3)[,c(1,2,5,6,3,4)]) %>%
  rename("lower" = !!names(. [3]),
    "upper" = !!names(. [4])) %>%
  mutate(coefficient = rownames(summary(mods_haem_stan)$coef[[1]])) %>%
  ggplot(aes(x = Estimate, y = coefficient)) + theme_bw() +
  geom_point() + geom_errorbarh(aes(xmin = lower, xmax = upper, y = coefficient), height = 0.25) +
  geom_vline(xintercept = 0, lty = 2) +
  ylab("") + xlab("Coefficient values") +
  ggtitle("Sanna model coefficients")
```

## Sanna model coefficients



```
#Stanford summary of interventions in villages
stanford %>% group_by(School, year) %>%
  summarise(net01 = mean(net),
            prawn01 = mean(prawn),
            veg01 = mean(veg_removal),
            Intervention = unique(Intervention.type))
```

```
## # A tibble: 48 x 6
## # Groups:   School [?]
##   School year net01 prawn01 veg01 Intervention
##   <fct> <int> <dbl> <dbl> <dbl> <fct>
## 1 DF 2016 0. 0. 0. control
## 2 DF 2017 0. 0. 0. control
## 3 DF 2018 0. 0. 0. control
## 4 DT 2016 0. 0. 0. control
## 5 DT 2017 0. 0. 0. control
## 6 DT 2018 0. 0. 0. control
## 7 FS 2016 1. 1. 0. prawn
## 8 FS 2017 1. 1. 0. prawn
## 9 FS 2018 1. 1. 0. prawn
## 10 GG 2016 1. 1. 0. prawn
## # ... with 38 more rows
```

I think there's an issue with how the dataset is formatted here: since the intervention is assigned to the community and doesn't vary across years in the dataset, this model is trying to estimate the effect of intervention across all three years, when it was only acting on the third (2018) data point in reality

**Jason: BACI analysis with individual and year random effects nested within community (village) random effects**

$$Y_{ijt} = \beta_0 + \epsilon_{0j} + \epsilon_{0ij} + \epsilon_{0jt} + \beta_1 year_t + \beta_2 Intervention_{jt} + \beta_3 year_t \times Intervention_{jt} + \beta_4 sex_i + \beta_5 lake_j$$

```
jas_mod <- glmer(Sh ~ Intervention.type*year + sex + lake + (1|Village/ID) + (1|Village/year), family="binomial")
###NOT CONVERGING###
```

I can't get this model to converge. I'm also not sure it's getting the most out of our data: surely 2016 holds valuable information for us. We're also parsing egg burden down to a binary presence-absence variable and using a logistic model which causes us to lose quite a lot of information on infection intensity.

I also think we need to think critically about our use of random effects model. After all, a random effect is an assumption placed on the distribution of the variable for which we are estimating a random effect. Here we're assuming that communities' mean egg burden is normally distributed around some global mean and within that community, each individual's egg burden is normally distributed around the village level mean, and each year

## Berkeley approach

We propose a sequence of analyses with increasingly *fewer* assumptions in order to estimate the effect of interventions on egg burden. We assume individual egg burden follows a negative binomial distribution  $Y_{ijt} \sim NB(\mu, \theta)$

**Model 1:** Individual egg burden,  $Y_{ijt}$  is determined by a common transmission intensity shared by all communities,  $T$ , praziquantel administration,  $P_t$ , and intervention in a given year,  $I_{jt}$

**Proposed DAG:**

Graphical representation of the data-generating mechanism corresponding to assumptions of Model 1.  $Y_{ijt}$  is the measured egg burden at each time point,  $T$  is the transmission intensity that produces  $Y_{ijt}$  at each time point,  $I_{jt}$  is the intervention in each community at time  $t$ , and  $P_t$  is praziquantel administration

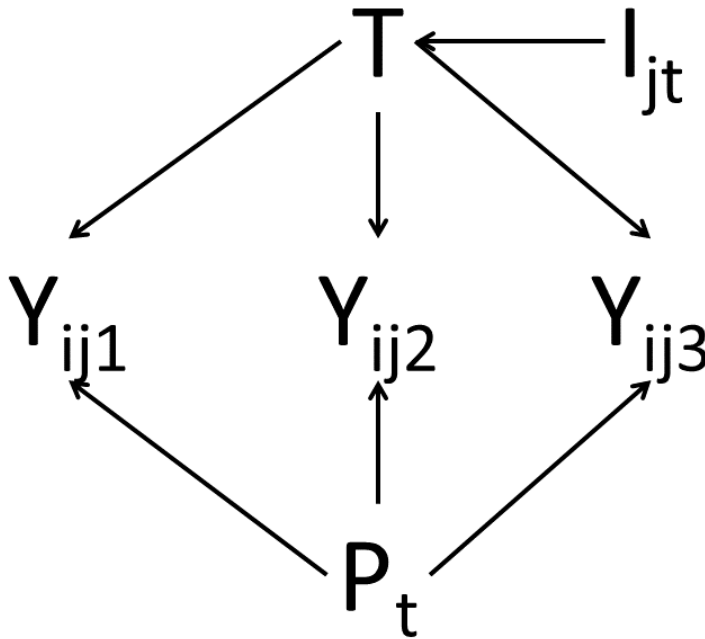


Figure 2: Proposed DAG for model 1

**Proposed Model:**

$$Y_{ijt} = \beta_0 + \beta_1 I_{jt} + \beta_2 P_t$$

Therefore in year 1 with no Praziquantel and no intervention:

$$Y_{ij1} = \beta_0$$

In year 2 with praziquantel, but no intervention:

$$Y_{ij2} = \beta_0 + \beta_2$$

And in year 3 with both praziquantel and intervention:

$$Y_{ij3} = \beta_0 + \beta_1 I_{jt} + \beta_2$$

This model will be fit by stratifying on intervention groups and comparing them to control communities, making  $I_{jt}$  a simple binary variable defined at the community level with control communities as the reference group and estimate  $\beta_1$  as our parameter of interest using a negative binomial GLM

Since this model fails to account for variability in transmission between communities other than that which is presumed to be caused by the intervention, associations may be confounded by baseline variability in community-level transmission: e.g. intervention communities have inherently higher transmission intensity than control communities. Model 2 addresses this by including main effects of communities.

**Model 2: Individual egg burden,  $Y_{ijt}$  is determined by each community's unique transmission intensity,  $T_j$ , praziquantel administration,  $P_t$ , and interventions,  $I_{jt}$**

**Proposed DAG:**

Graphical representation of the data-generating mechanism corresponding to assumptions of Model 2.  $Y_{ijt}$  is the measured egg burden at each time point,  $T_j$  is the transmission intensity unique to each community that produces  $Y_{ijt}$  at each time point,  $I_{jt}$  is the intervention in each community, and  $P_t$  is praziquantel administration

**Proposed Model:**

$$Y_{ijt} = \beta_0 + \beta_1 I_{jt} + \beta_2 P_t + \beta_3 T_j$$

Therefore in year 1 with no Praziquantel and no intervention:

$$Y_{ij1} = \beta_0 + \beta_3 T_j$$

In year 2 with praziquantel, but no intervention:

$$Y_{ij2} = \beta_0 + \beta_2 + \beta_3 T_j$$

And in year 3 with both praziquantel and intervention:

$$Y_{ij3} = \beta_0 + \beta_1 I_{jt} + \beta_2 + \beta_3 T_j$$

We'll again fit this model with  $I_{jt}$  as a simple binary variable defined at the community-year level with control communities as the reference group and estimate  $\beta_1$  as our parameter of interest using a negative binomial GLM.  $\beta_0 + \beta_3$  can be interpreted as the community-level transmission intensity,  $T_j$

This is better since we're treating communities as their own entities and therefore controlling for inter-community variability in transmission, but we're still treating *individual* egg burdens as independent when we know that individuals are likely correlated. Model 3 will address this concern by controlling for individuals' previous egg burden. In addition to controlling for confounding by inter-community and inter-individual variability, conditioning on prior outcome is also a good way to control for potential unmeasured confounding from other sources

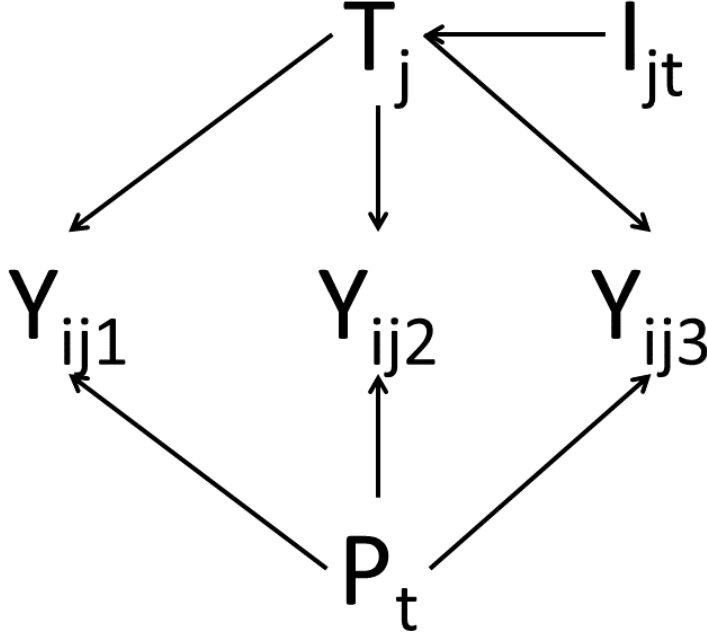


Figure 3: Proposed DAG for model 2

**Model 3:** Individual egg burden,  $Y_{ijt}$  is determined by each community's unique transmission intensity that varies by year,  $T_{jt}$ , praziquantel administration,  $P_t$ , and interventions,  $I_{jt}$

**Proposed DAG:**

Graphical representation of the data-generating mechanism corresponding to assumptions of Model 3.  $Y_{ijt}$  is the measured egg burden at each time point,  $T_j$  is the transmission intensity unique to each community that produces  $Y_{ijt}$  at each time point,  $I_{jt}$  is the intervention in each community, and  $P_t$  is praziquantel administration

**Proposed Model:**

$$Y_{ijt} = \beta_0 + \beta_1 I_{jt} + \beta_2 P_t + \beta_3 T_j + \beta_4 Year_t + \beta_5 T_j \times Year_t$$

Therefore in year 1 with no Praziquantel and no intervention:

$$Y_{ij1} = \beta_0 + \beta_3 T_j$$

In year 2 with praziquantel, but no intervention:

$$Y_{ij2} = \beta_0 + \beta_2 + \beta_3 T_j + \beta_4 Year_2 + \beta_5 T_j \times Year_2$$

And in year 3 with both praziquantel and intervention:

$$Y_{ij3} = \beta_0 + \beta_1 I_{jt} + \beta_2 + \beta_3 T_j + \beta_4 Year_3 + \beta_5 T_j \times Year_3$$

We'll again fit this model with  $I_{jt}$  as a simple binary variable defined at the community-year level with control communities as the reference group and estimate  $\beta_1$  as our parameter of interest using a negative binomial GLM.  $\beta_0 + \beta_3 + \beta_4 + \beta_5$  can be interpreted as the community-level transmission intensity in year  $t$ ,  $T_{jt}$

This model is still not accounting for potential correlation between individuals that is not explained at the community level, which is what models 4 and 5 will account for

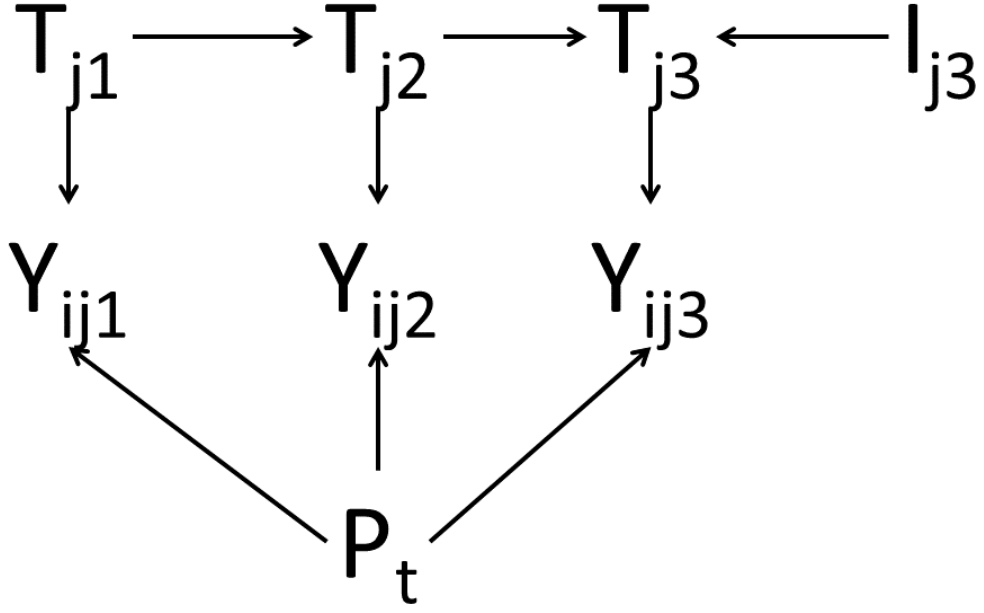


Figure 4: Proposed DAG for model 3

**Model 4: Transition model in which individual egg burden,  $Y_{ijt}$  is associated with previous infection,  $Y_{ijt-1}$ , and determined by each community's transmission intensity,  $T_j$  and interventions,  $I_{jt}$**

**Proposed DAG:**

Graphical representation of the data-generating mechanism corresponding to assumptions of Model 4.  $Y_{ijt}$  is the measured egg burden at each time point,  $T_j$  is the transmission intensity unique to each community that produces  $Y_{ijt}$  at each time point, and  $I_{jt}$  is the intervention in each community. Including  $Y_{ijt-1}$  controls for confounding by  $T_j$  regardless of whether dummy variables on communities serve as a good proxy for  $T_j$ .

**Proposed Model:**

$$Y_{ijt} = \beta_0 + \beta_1 I_{jt} + \beta_2 \log(Y_{ijt-1} + 1) + \beta_3 T_j$$

Therefore in year 2 with no intervention:

$$Y_{ij2} = \beta_0 + \beta_2 \log(Y_{ij1} + 1) + \beta_3 T_j$$

And in year 3 with intervention:

$$Y_{ij3} = \beta_0 + \beta_1 I_{jt} + \beta_2 \log(Y_{ij2} + 1) + \beta_3 T_j$$

We'll again fit this model with  $I_{jt}$  as a simple binary variable defined at the community level with control communities as the reference group and estimate  $\beta_1$  as our parameter of interest using a negative binomial GLM. Since the outcome is only defined in 2017 and 2018, no need to control for praziquantel administration which occurred in both years.

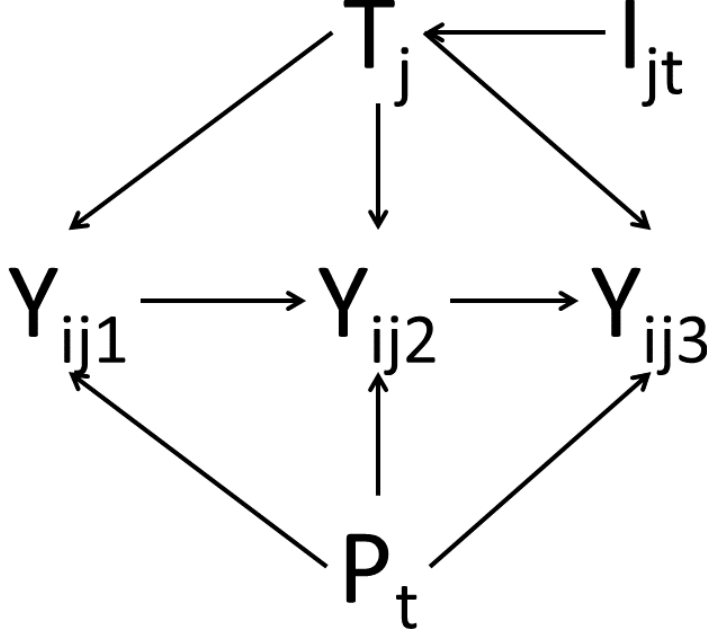


Figure 5: Proposed DAG for model 4

**Model 5: Transition model in which individual egg burden,  $Y_{ijt}$  is associated with previous infection,  $Y_{ijt-1}$ , and determined by each community's transmission intensity that varies with year,  $T_jt$ , and interventions,  $I_{jt}$**

**Proposed DAG:**

Graphical representation of the data-generating mechanism corresponding to assumptions of Model 5.  $Y_{ijt}$  is the measured egg burden at each time point,  $T_jt$  is the transmission intensity unique to each community in each year that produces  $Y_{ijt}$  at each time point, and  $I_{jt}$  is the intervention in each community. Including  $Y_{ijt-1}$  controls for confounding by  $T_j$  regardless of whether dummy variables on communities serve as a good proxy for  $T_j$ .

**Proposed Model:**

$$Y_{ijt} = \beta_0 + \beta_1 I_{jt} + \beta_2 \log(Y_{ijt-1} + 1) + \beta_3 T_j + \beta_4 Year_t + \beta_5 T_j \times Year_t$$

Therefore in year 2 with no intervention (and year 2 as reference for the  $Year_t$  variable):

$$Y_{ij2} = \beta_0 + \beta_2 \log(Y_{ij1} + 1) + \beta_3 T_j$$

And in year 3 with intervention:

$$Y_{ij3} = \beta_0 + \beta_1 I_{jt} + \beta_2 \log(Y_{ij2} + 1) + \beta_3 T_j + \beta_4 Year_t + \beta_5 T_j \times Year_t$$

We'll again fit this model with  $I_{jt}$  as a simple binary variable defined at the community level with control communities as the reference group and estimate  $\beta_1$  as our parameter of interest using a negative binomial GLM. Since the outcome is only defined in 2017 and 2018, no need to control for praziquantel administration which occurred in both years.



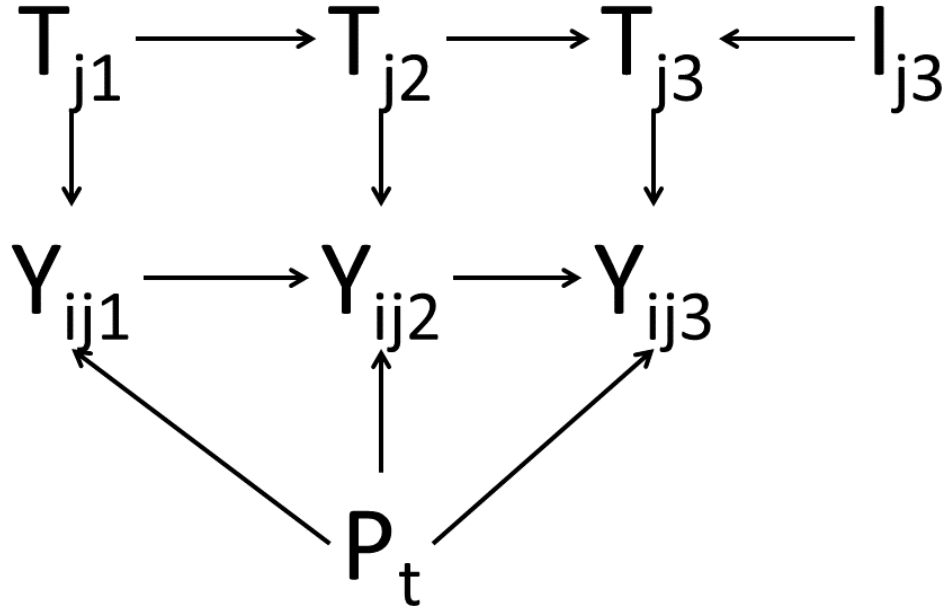


Figure 6: Proposed DAG for model 5

## Implementation

Models 4 and 5 actually require a bit more data wrangling to add the lag variables:

```

mod_comm <- full_long_comm %>% arrange(School, year) %>% group_by(School) %>%
  mutate(haem_mean_l1 = lag(haem_mean),
         haem_diff_l1 = lag(vil_haem_diff),
         haem_2016 = haem_mean[which(year == 2016)],
         log_haem_mean_l1 = lag(log(haem_mean+1)),
         log_haem_2016 = log(haem_mean[which(year == 2016)]+1),
         mans_mean_l1 = lag(mans_mean),
         mans_diff_l1 = lag(vil_mans_diff),
         mans_2016 = mans_mean[which(year == 2016)],
         log_mans_mean_l1 = lag(log(mans_mean+1)),
         log_mans_2016 = log(mans_mean[which(year == 2016)]+1),
         year_fac = factor(as.character(year), levels = c("2016", "2017", "2018")),
         pzq = ifelse(year != 2016, 1, 0))

mod4_dat <- full_long %>% arrange(Child_ID, year) %>% group_by(Child_ID) %>%
  mutate(s_haem_mean_l1 = lag(s_haem_mean),
         log_s_haem_mean_l1 = lag(log(s_haem_mean+1)),
         s_haem_mean_narm_l1 = lag(s_haem_mean_narm),
         log_s_haem_mean_narm_l1 = lag(log(s_haem_mean_narm+1)),
         s_mans_mean1_l1 = lag(s_mans_mean1),
         log_s_mans_mean1_l1 = lag(log(s_mans_mean1+1)),
         s_mans_mean2_l1 = lag(s_mans_mean2),
         log_s_mans_mean2_l1 = lag(log(s_mans_mean2+1)),
         s_mans_mean12_l1 = lag(s_mans_mean12),
         log_s_mans_mean12_l1 = lag(log(s_mans_mean12+1)),
         s_mans_mean_narm_l1 = lag(s_mans_mean_narm),
         log_s_mans_mean_narm_l1 = lag(log(s_mans_mean_narm+1))) %>%

```

```
full_join(mod_comm %>% select(School, year,
                             haem_mean_l1, haem_2016,
                             mans_mean_l1, mans_2016),
          by = c("School", "year"))
```

```
## Warning: Column `School` joining factor and character vector, coercing into
## character vector
```

```
#function to get robust standard errors and spit out data frame with 95%CI
robust_fx <- function(mod_obj){
  df_init <- as.data.frame(summary(mod_obj)$coef)

  rob_se <- sqrt(diag(vcovHC(mod_obj, type = "HCO")))
  rob_lo <- rob_se * qnorm(0.025) + df_init[, "Estimate"]
  rob_up <- rob_se * qnorm(0.975) + df_init[, "Estimate"]

  out <- cbind(df_init, rob_se, rob_lo, rob_up)

  return(out)
}
```

## Community level

```
#Model 1
mod1_haem_comm_all <- glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq,
                             weights = haem_samps,
                             data = mod_comm)

mod1_haem_comm_all_res <- robust_fx(mod1_haem_comm_all)
mod1_haem_comm_all_res <- cbind(mod1_haem_comm_all_res, confint(mod1_haem_comm_all))
```

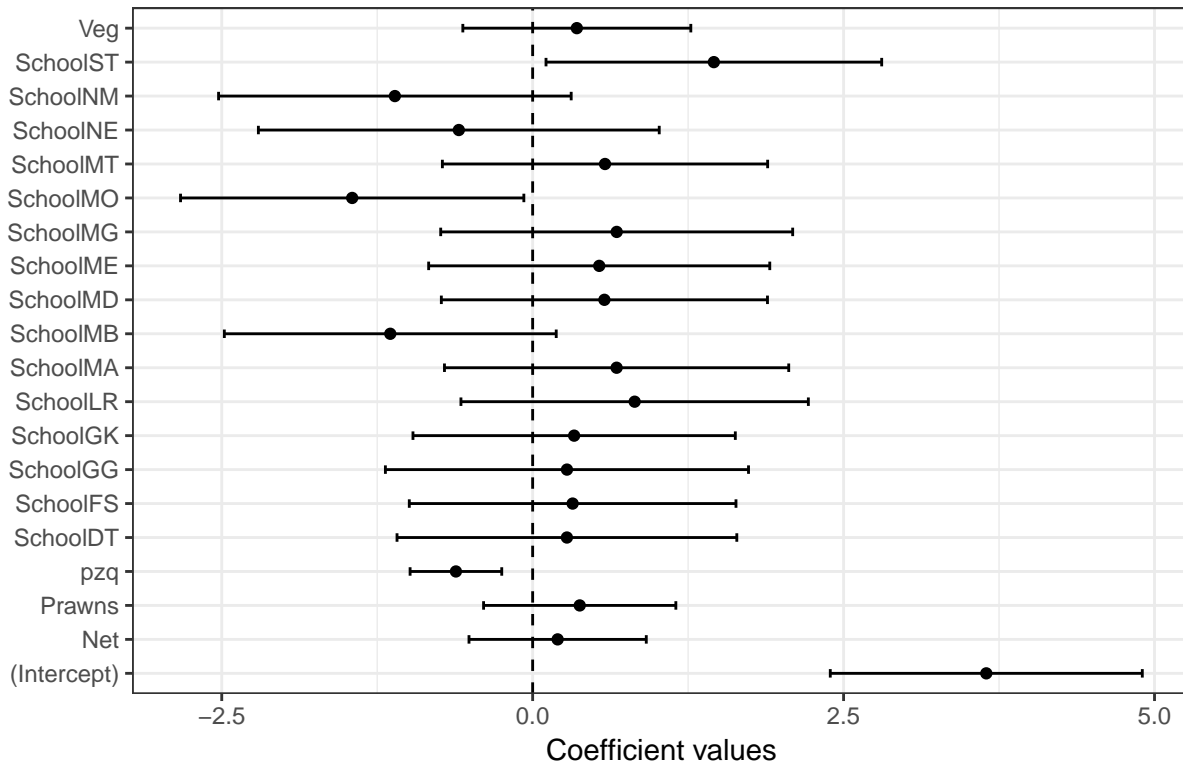
```
## Waiting for profiling to be done...
```

```
mod1_haem_comm_all_res$model <- "Model 1"

mod1_haem_comm_all_res %>%
  mutate(coefficient = rownames(mod1_haem_comm_all_res)) %>%
  ggplot(aes(x = Estimate, y = coefficient)) + theme_bw() +
  geom_point() +
  geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = coefficient), height = 0.25) +
  geom_vline(xintercept = 0, lty = 2) +
  ylab("") + xlab("Coefficient values") +
  ggtitle("Berkeley model coefficients",
          subtitle = "95% CIs derived from robust standard errors")
```

## Berkeley model coefficients

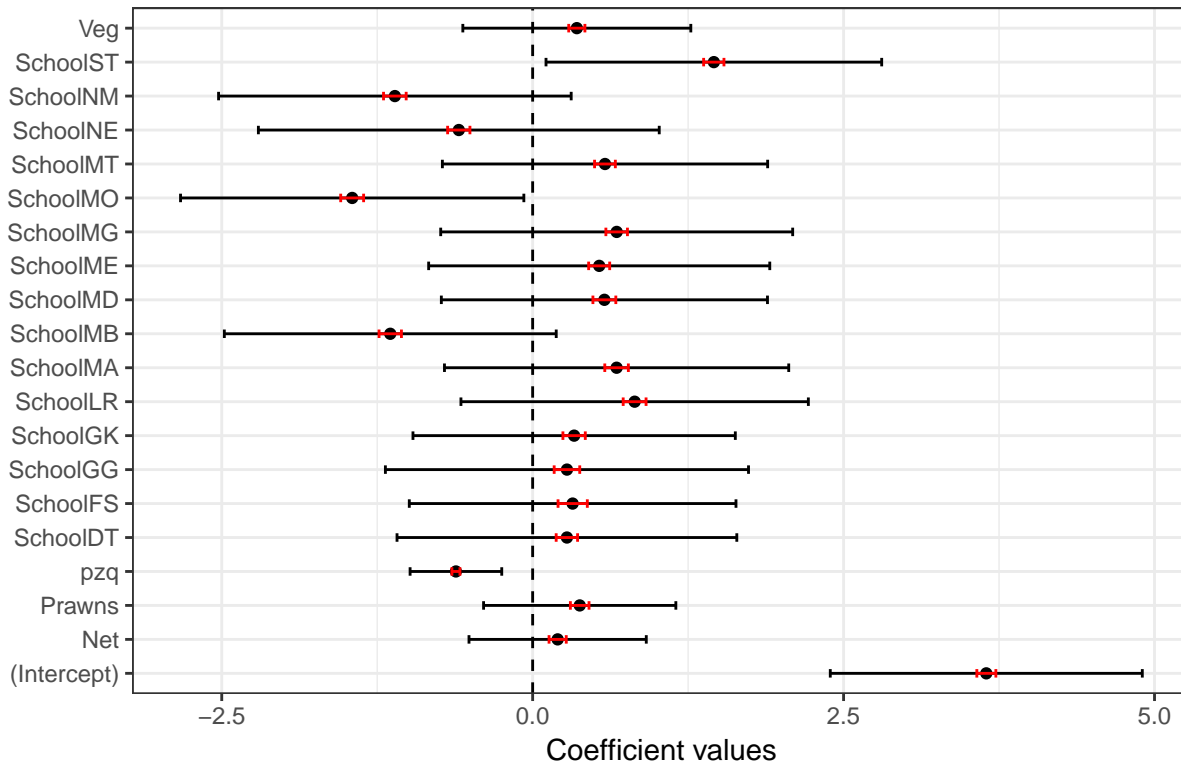
95% CIs derived from robust standard errors



```
mod1_haem_comm_all_res %>%
  mutate(coefficient = rownames(mod1_haem_comm_all_res)) %>%
  rename("naive_lo" = !!names(.[8]),
         "naive_up" = !!names(.[9])) %>%
  ggplot(aes(x = Estimate, y = coefficient)) + theme_bw() +
    geom_point() +
    geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = coefficient), height = 0.25) +
    geom_errorbarh(aes(xmin = naive_lo, xmax = naive_up, y = coefficient), height = 0.25, col = 2) +
    geom_vline(xintercept = 0, lty = 2) +
    ylab("") + xlab("Coefficient values") +
    ggtitle("Berkeley model coefficients",
            subtitle = "95% CIs derived from robust standard errors")
```

## Berkeley model coefficients

95% CIs derived from robust standard errors



*#Model 2*

```
mod2_haem_comm_all <- glm.nb(round(haem_mean) ~ Net + Veg + Prawns + year_fac + haem_mean_l1,
                             weights = haem_samps,
                             data = mod_comm)
```

```
mod2_haem_comm_all_res <- robust_fx(mod2_haem_comm_all)
```

```
mod2_haem_comm_all_res <- cbind(mod2_haem_comm_all_res, confint(mod2_haem_comm_all))
```

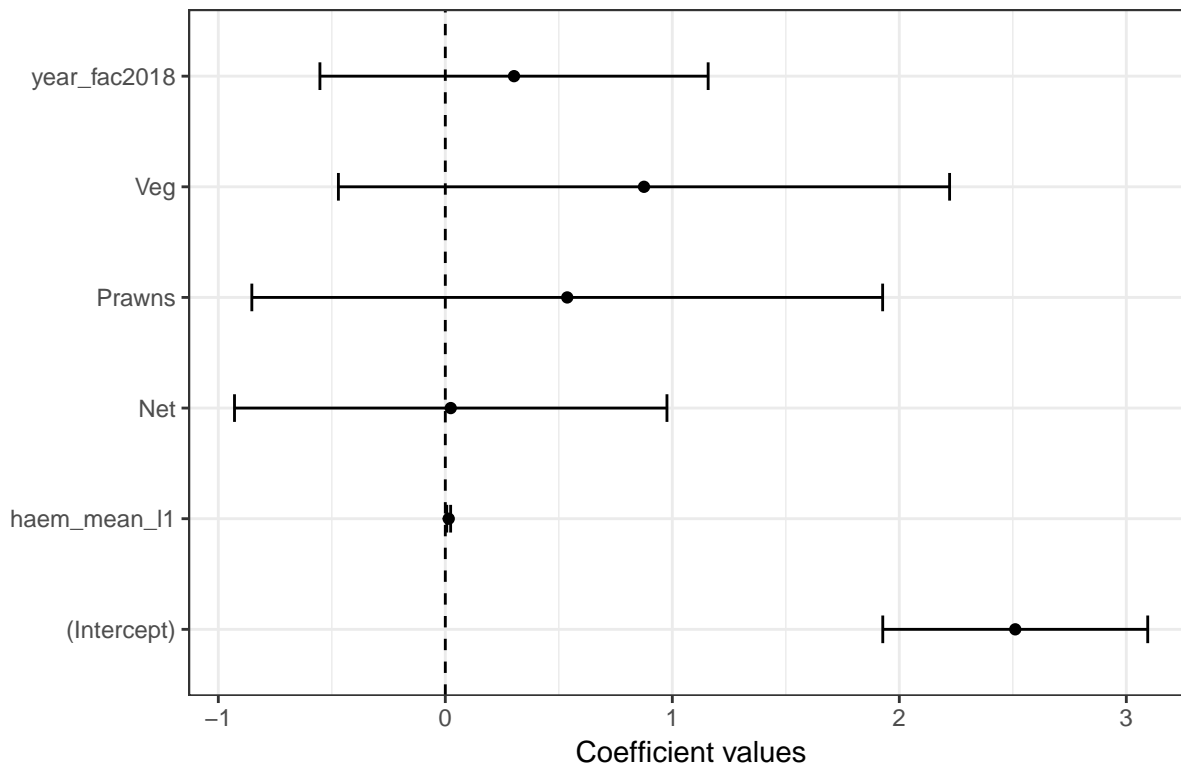
## Waiting for profiling to be done...

```
mod2_haem_comm_all_res$model <- "Model 2"
```

```
mod2_haem_comm_all_res %>%
  mutate(coefficient = rownames(mod2_haem_comm_all_res)) %>%
  ggplot(aes(x = Estimate, y = coefficient)) + theme_bw() +
  geom_point() +
  geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = coefficient), height = 0.25) +
  geom_vline(xintercept = 0, lty = 2) +
  ylab("") + xlab("Coefficient values") +
  ggtitle("Berkeley model2 coefficients",
          subtitle = "95% CIs derived from robust standard errors")
```

## Berkeley model2 coefficients

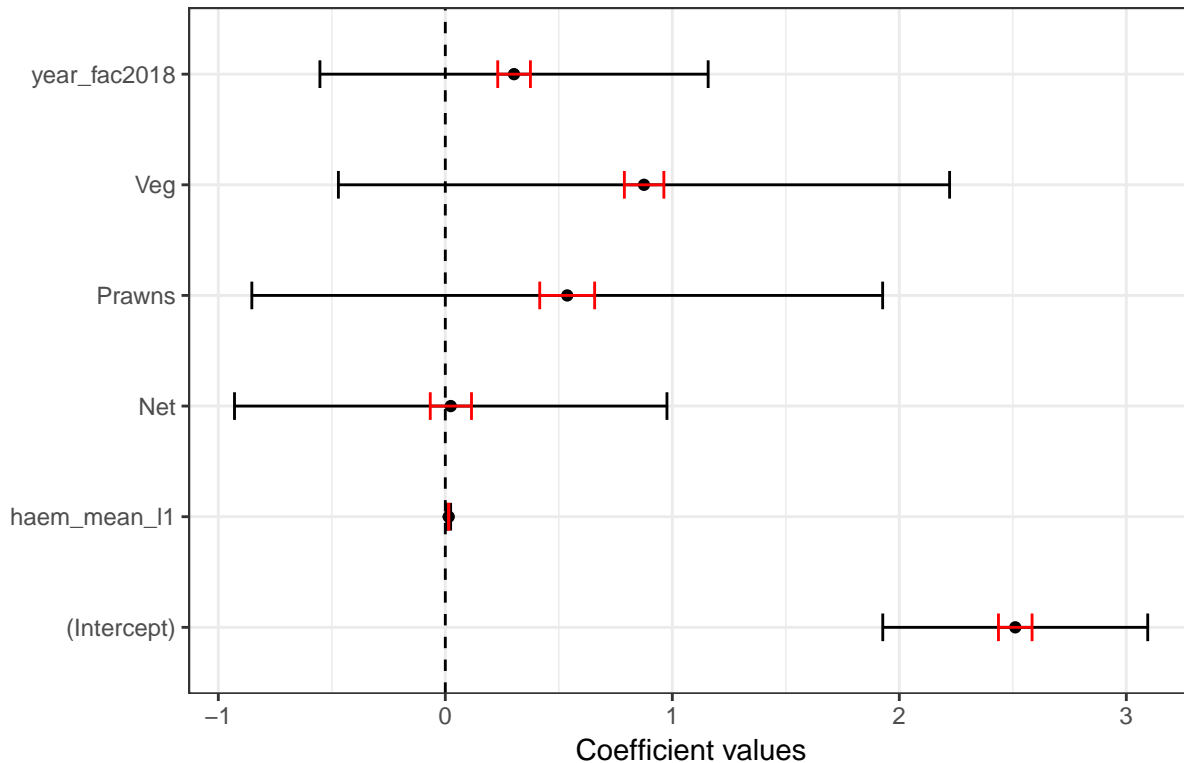
95% CIs derived from robust standard errors



```
mod2_haem_comm_all_res %>%
  mutate(coefficient = rownames(mod2_haem_comm_all_res)) %>%
  rename("naive_lo" = !!names(.[8]),
         "naive_up" = !!names(.[9])) %>%
  ggplot(aes(x = Estimate, y = coefficient)) + theme_bw() +
  geom_point() +
  geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = coefficient), height = 0.25) +
  geom_errorbarh(aes(xmin = naive_lo, xmax = naive_up, y = coefficient), height = 0.25, col = 2) +
  geom_vline(xintercept = 0, lty = 2) +
  ylab("") + xlab("Coefficient values") +
  ggtitle("Berkeley model2 coefficients",
         subtitle = "95% CIs derived from robust standard errors")
```

## Berkeley model2 coefficients

95% CIs derived from robust standard errors



### #Model 0

```
mod0_haem_comm_all <- glm.nb(round(haem_mean) ~ year_fac + extra_pzq,
                             weights = haem_samps,
                             data = mod_comm)
```

```
mod0_haem_comm_all_res <- robust_fx(mod0_haem_comm_all)
mod0_haem_comm_all_res$model <- "Model 0"
```

### #Model 00

```
mod00_haem_comm_all <- glm.nb(round(haem_mean) ~ year_fac + School + extra_pzq,
                              weights = haem_samps,
                              data = mod_comm)
```

```
mod00_haem_comm_all_res <- robust_fx(mod00_haem_comm_all)
mod00_haem_comm_all_res$model <- "Model 0"
```

### #Model 1

```
mod1_haem_comm_all <- glm.nb(round(haem_mean) ~ Net + Veg + Prawns + year_fac + extra_pzq,
                              weights = haem_samps,
                              data = mod_comm)
```

```
mod1_haem_comm_all_res <- robust_fx(mod1_haem_comm_all)
mod1_haem_comm_all_res$model <- "Model 1"
```

### #Model 2

```
mod2_haem_comm_all <- glm.nb(round(haem_mean) ~ Net + Veg + Prawns + year_fac + School + extra_pzq,
                              weights = haem_samps,
                              data = mod_comm)
```

```
mod2_haem_comm_all_res <- robust_fx(mod2_haem_comm_all)
mod2_haem_comm_all_res$model <- "Model 2"
```

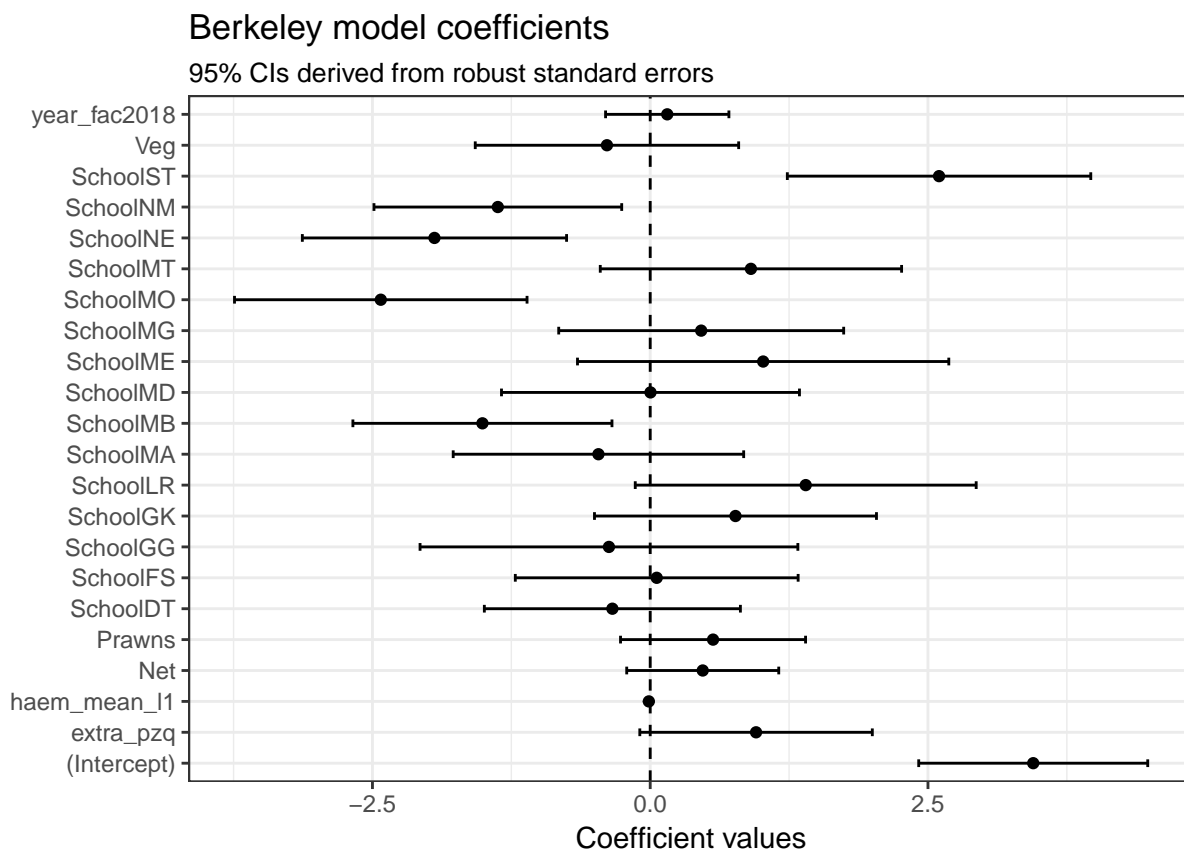
```

#Model 4
mod4_haem_comm_all <- glm.nb(round(haem_mean) ~ Net + Veg + Prawns + year_fac + School + haem_mean_l1 + extra_pzq,
                             weights = haem_samps,
                             data = mod_comm)

mod4_haem_comm_all_res <- robust_fx(mod4_haem_comm_all)
mod4_haem_comm_all_res$model <- "Model 4"

mod4_haem_comm_all_res %>%
  mutate(coefficient = rownames(mod4_haem_comm_all_res)) %>%
  ggplot(aes(x = Estimate, y = coefficient)) + theme_bw() +
  geom_point() +
  geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = coefficient), height = 0.25) +
  geom_vline(xintercept = 0, lty = 2) +
  ylab("") + xlab("Coefficient values") +
  ggtitle("Berkeley model coefficients",
          subtitle = "95% CIs derived from robust standard errors")

```



```
anova(mod0_haem_comm_all, mod1_haem_comm_all, mod2_haem_comm_all, mod4_haem_comm_all)
```

```

## Likelihood ratio tests of Negative Binomial Models
##
## Response: round(haem_mean)
##
## 1 year_fac + extra_pzq
## 2 Net + Veg + Prawns + year_fac + extra_pzq
## 3 Net + Veg + Prawns + year_fac + School + extra_pzq
## 4 Net + Veg + Prawns + year_fac + School + haem_mean_l1 + extra_pzq
##   theta Resid. df    2 x log-lik.  Test  df  LR stat. Pr(Chi)
## 1 1.241787    44    -74432.44
## 2 1.271622    41    -74222.48 1 vs 2    3   209.9508    0

```



## 3	3.601229	26	-65911.59	2 vs 3	15	8310.8925	0
## 4	4.759839	10	-37678.44	3 vs 4	16	28233.1539	0

*#Model deviation from the village level mean egg burden  
#rather than the egg burden iteself. Allows for increased efficiency since  
#the village effects are encapsulated in the data. Also allows for coefficient  
#estimates on a linear rather than log scale*

*#Model 2*

```
mod2_haem_diff_comm_all <- glm(vil_haem_diff ~ Net + Veg + Prawns + year_fac,
                               weights = haem_samps,
                               data = mod_comm)
```

```
mod2_haem_diff_comm_all_res <- robust_fx(mod2_haem_diff_comm_all)
mod2_haem_diff_comm_all_res$model <- "Model 2"
```

*#Model 4*

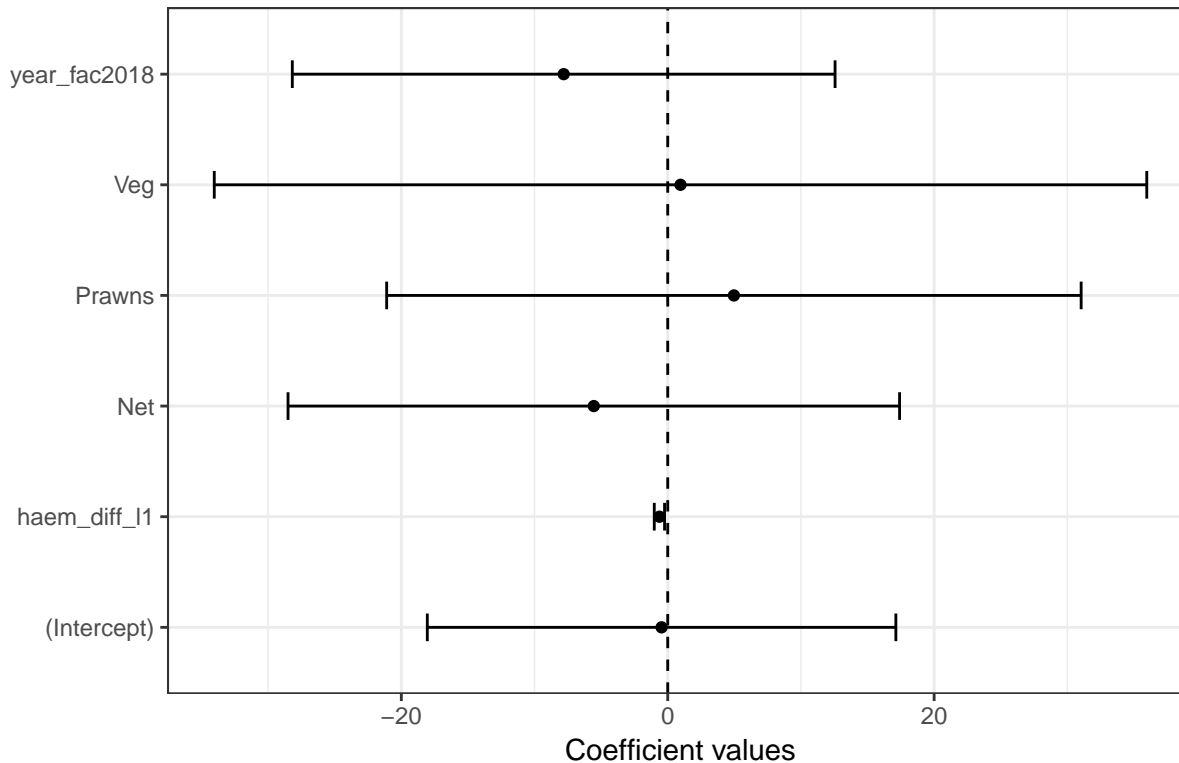
```
mod4_haem_diff_comm_all <- glm(vil_haem_diff ~ Net + Veg + Prawns + year_fac + haem_diff_l1,
                               weights = haem_samps,
                               data = mod_comm)
```

```
mod4_haem_diff_comm_all_res <- robust_fx(mod4_haem_diff_comm_all)
mod4_haem_diff_comm_all_res$model <- "Model 4"
```

```
mod4_haem_diff_comm_all_res %>%
  mutate(coefficient = rownames(mod4_haem_diff_comm_all_res)) %>%
  ggplot(aes(x = Estimate, y = coefficient)) + theme_bw() +
  geom_point() +
  geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = coefficient), height = 0.25) +
  geom_vline(xintercept = 0, lty = 2) +
  ylab("") + xlab("Coefficient values") +
  ggtitle("Berkeley model coefficients",
          subtitle = "95% CIs derived from robust standard errors")
```

## Berkeley model coefficients

95% CIs derived from robust standard errors



```
anova(mod0_haem_comm_all, mod00_haem_comm_all,
      mod1_haem_comm_all, mod2_haem_comm_all, mod4_haem_comm_all)
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: round(haem_mean)
##
##                                     Model
## 1                                     year_fac + extra_pzq
## 2               Net + Veg + Prawns + year_fac + extra_pzq
## 3               year_fac + School + extra_pzq
## 4               Net + Veg + Prawns + year_fac + School + extra_pzq
## 5 Net + Veg + Prawns + year_fac + School + haem_mean_l1 + extra_pzq
##      theta Resid. df    2 x log-lik. Test    df  LR stat. Pr(Chi)
## 1 1.241787    44      -74432.44
## 2 1.271622    41      -74222.48 1 vs 2     3    209.9508    0
## 3 3.509871    29      -66134.50 2 vs 3    12   8087.9835    0
## 4 3.601229    26      -65911.59 3 vs 4     3    222.9090    0
## 5 4.759839    10      -37678.44 4 vs 5    16  28233.1539    0
```

## Permutation test

Conduct permutation test to determine, given our data, the probability of finding a significant effect of treatment regardless of how the interventions were assigned to communities. Basically determines if any finding of a significant association is a true effect or was pre-ordained given our data.

```
#Function to randomly reallocate treatment among communities
#Made tricky by the fact that treatments vary each year
#and there's dependence between years since the four communities
#that received nets in 2017 also received prawns in 2018
```

```

permute <- function(){
  #Get full treatment vector
  treats <- mod_comm %>%
    unite(int_vec, c(Veg, Net, Prawns)) %>%
    group_by(School) %>%
    summarise(full_hist = paste(int_vec, collapse = "-")) %>%
    pull(full_hist)

  #Resample to scramble treatment
  mix <- sample(treats, replace = FALSE)

  #Unlist to rebind with the full df
  mix2 <- unlist(strsplit(mix, "-"))

  mod_perm <- cbind(mod_comm, mix2 = mix2) %>%
    #Replace old values with randomly scrambled ones
    select(-c(Veg, Net, Prawns)) %>%
    separate(mix2, c("Veg", "Net", "Prawns"))

  mod1_perm <- glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq,
    weights = haem_samps,
    data = mod_perm)

  mod1_int_coef <- summary(mod1_perm)$coef[, "Estimate"] [c("Veg1", "Net1", "Prawns1")]

  mod2_perm <- glm.nb(round(haem_mean) ~ Net + Veg + Prawns + year_fac + haem_mean_l1,
    weights = haem_samps,
    data = mod_perm)

  mod2_int_coef <- summary(mod2_perm)$coef[, "Estimate"] [c("Veg1", "Net1", "Prawns1")]

  return(c(mod1_int_coef, mod2_int_coef))
}

permute()

```

```

##      Veg1      Net1      Prawns1      Veg1      Net1      Prawns1
## 0.6900560 -0.6228587 0.2492713 1.2626181 -0.1681823 0.4973604

```

```

set.seed(43093)
perm_result <- replicate(10000, permute())

```

```

## Warning: glm.fit: algorithm did not converge

```

```

## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + year_fac +
## haem_mean_l1, : alternation limit reached

```

```

## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq, :
## alternation limit reached

```

```

## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq, :
## alternation limit reached

```

```

## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq, :
## alternation limit reached

```

```

## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq, :
## alternation limit reached

```

```
## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq, :
## alternation limit reached

## Warning: glm.fit: algorithm did not converge

## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq, :
## alternation limit reached

## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq, :
## alternation limit reached

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: algorithm did not converge

## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + year_fac +
## haem_mean_l1, : alternation limit reached

## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq, :
## alternation limit reached

## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq, :
## alternation limit reached

## Warning: glm.fit: algorithm did not converge

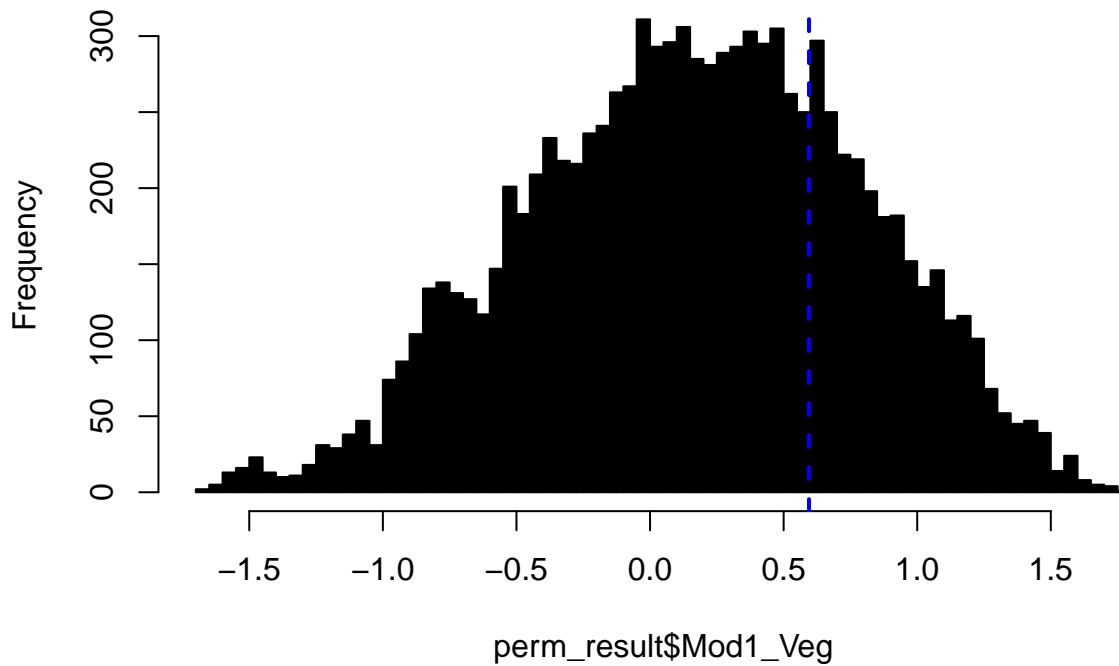
## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + year_fac +
## haem_mean_l1, : alternation limit reached

## Warning in glm.nb(round(haem_mean) ~ Net + Veg + Prawns + School + pzq, :
## alternation limit reached
```

```
perm_result <- as.data.frame(t(perm_result))
colnames(perm_result) <- c("Mod1_Veg", "Mod1_Net", "Mod1_Prawns",
                          "Mod2_Veg", "Mod2_Net", "Mod2_Prawns")

hist(perm_result$Mod1_Veg, col = "black", breaks = 100)
abline(v = summary(mod1_haem_comm_all)$coef[, "Estimate"]["Veg"],
       col = 4, lty = 2, lwd = 2)
```

## Histogram of perm\_result\$Mod1\_Veg

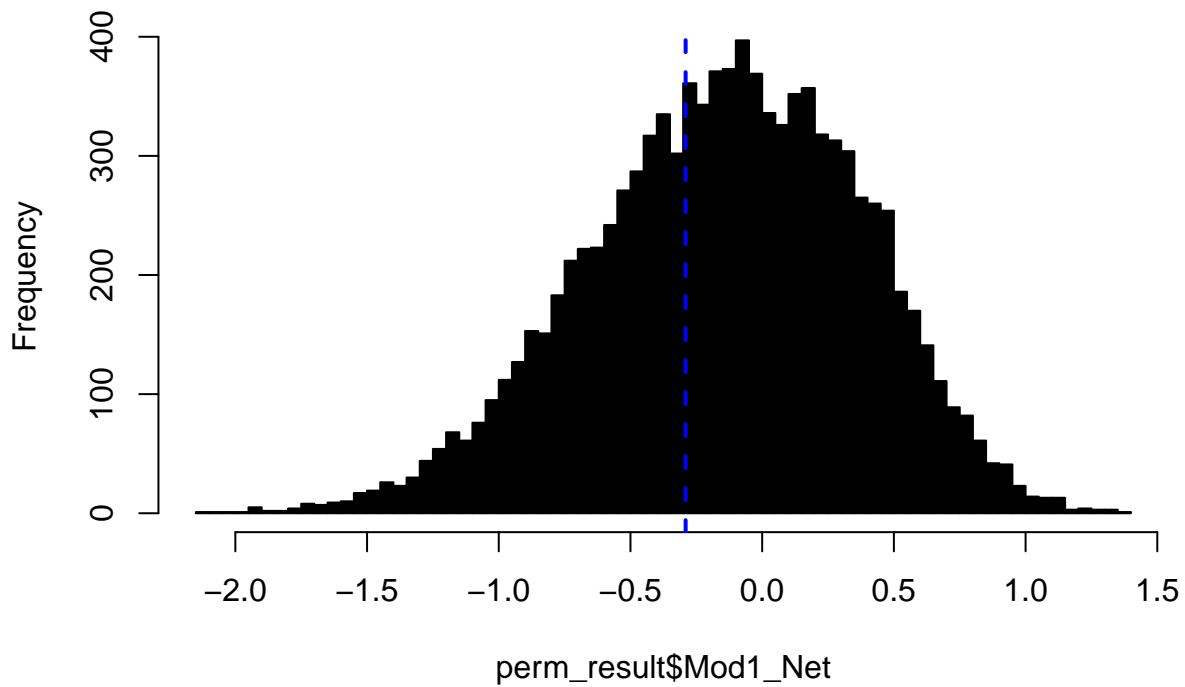


```
length(which(perm_result$Mod1_Veg >
             summary(mod1_haem_comm_all)$coef[, "Estimate"] ["Veg"]))) / 10000
```

```
## [1] 0.2651
```

```
hist(perm_result$Mod1_Net, col = "black", breaks = 100)
abline(v = summary(mod1_haem_comm_all)$coef[, "Estimate"] ["Net"],
       col = 4, lty = 2, lwd = 2)
```

## Histogram of perm\_result\$Mod1\_Net

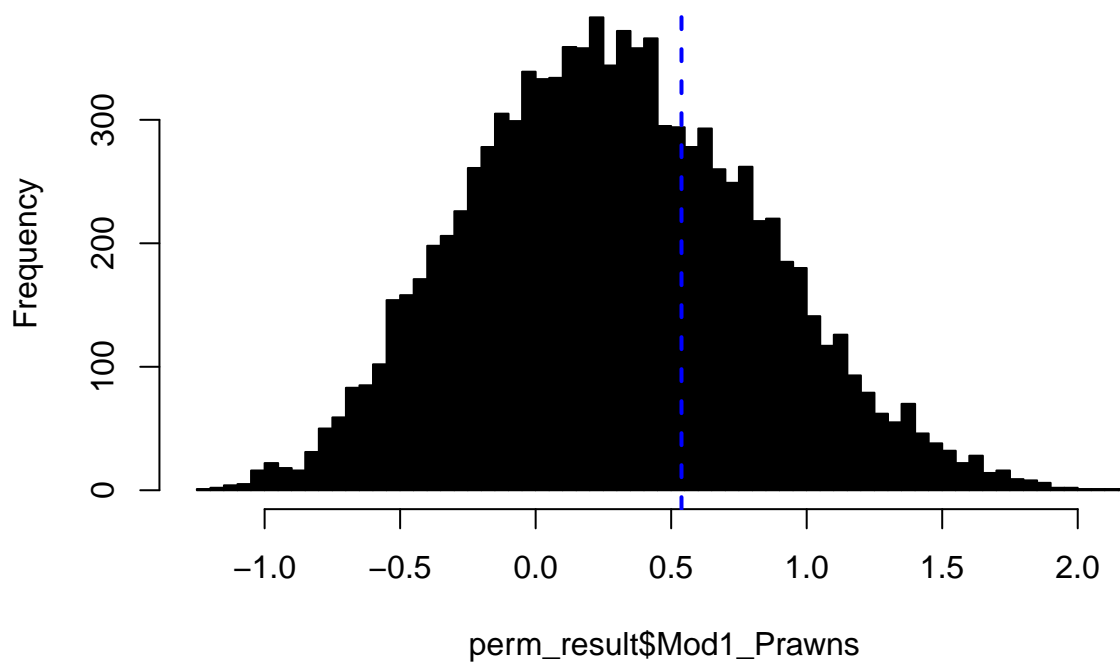


```
length(which(perm_result$Mod1_Net >
             summary(mod1_haem_comm_all)$coef[, "Estimate"] ["Net"]))) / 10000
```

```
## [1] 0.6234
```

```
hist(perm_result$Mod1_Prawns, col = "black", breaks = 100)
abline(v = summary(mod1_haem_comm_all)$coef[, "Estimate"] ["Prawns"],
       col = 4, lty = 2, lwd = 2)
```

## Histogram of perm\_result\$Mod1\_Prawns

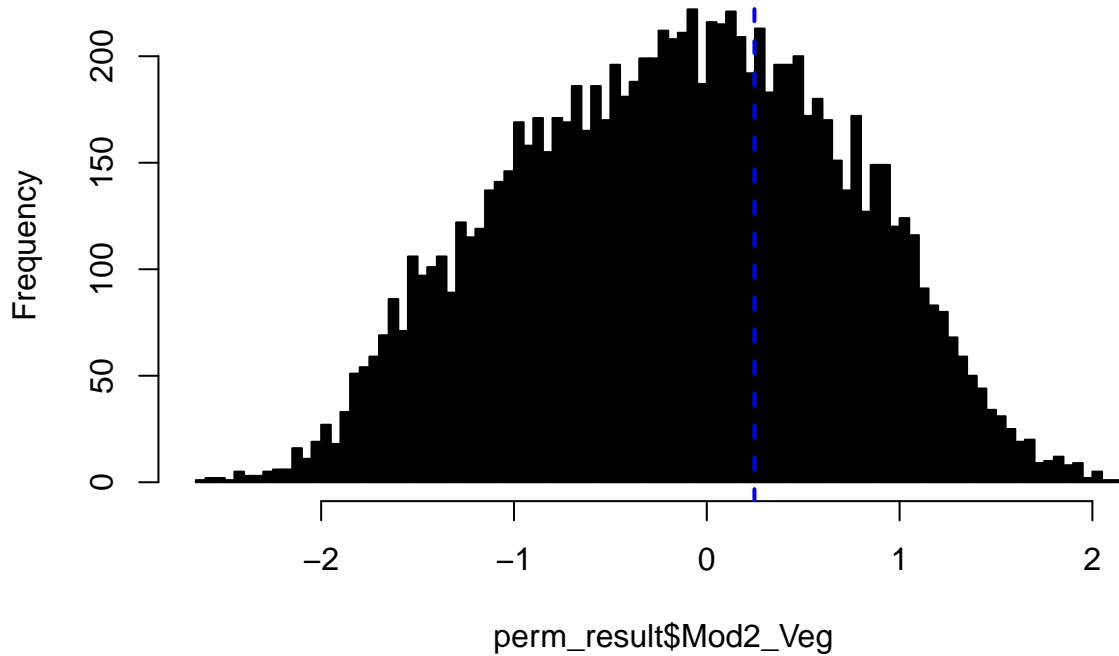


```
length(which(perm_result$Mod1_Prawns >
  summary(mod1_haem_comm_all)$coef[, "Estimate"] ["Prawns"]))) / 10000
```

```
## [1] 0.3192
```

```
hist(perm_result$Mod2_Veg, col = "black", breaks = 100)
abline(v = summary(mod2_haem_comm_all)$coef[, "Estimate"] ["Veg"],
  col = 4, lty = 2, lwd = 2)
```

## Histogram of perm\_result\$Mod2\_Veg



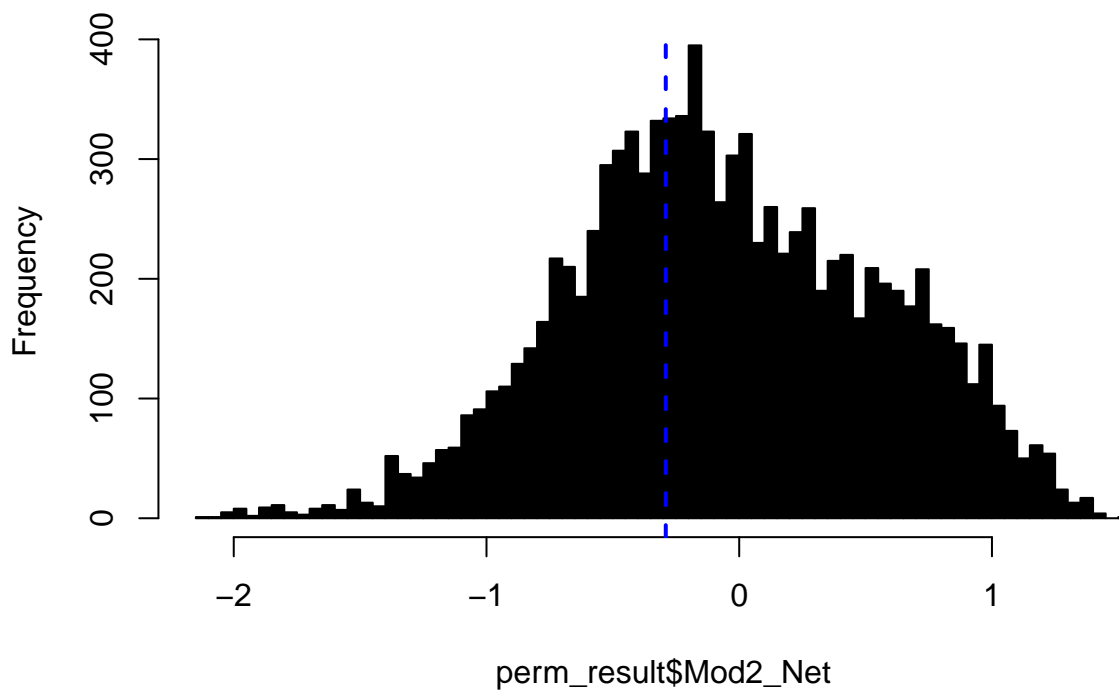
```
length(which(perm_result$Mod2_Veg >
  summary(mod2_haem_comm_all)$coef[, "Estimate"] ["Veg"])) / 10000
```

```
## [1] 0.3429
```

```
hist(perm_result$Mod2_Net, col = "black", breaks = 100)
abline(v = summary(mod1_haem_comm_all)$coef[, "Estimate"] ["Net"],
  col = 4, lty = 2, lwd = 2)
```



## Histogram of perm\_result\$Mod2\_Net

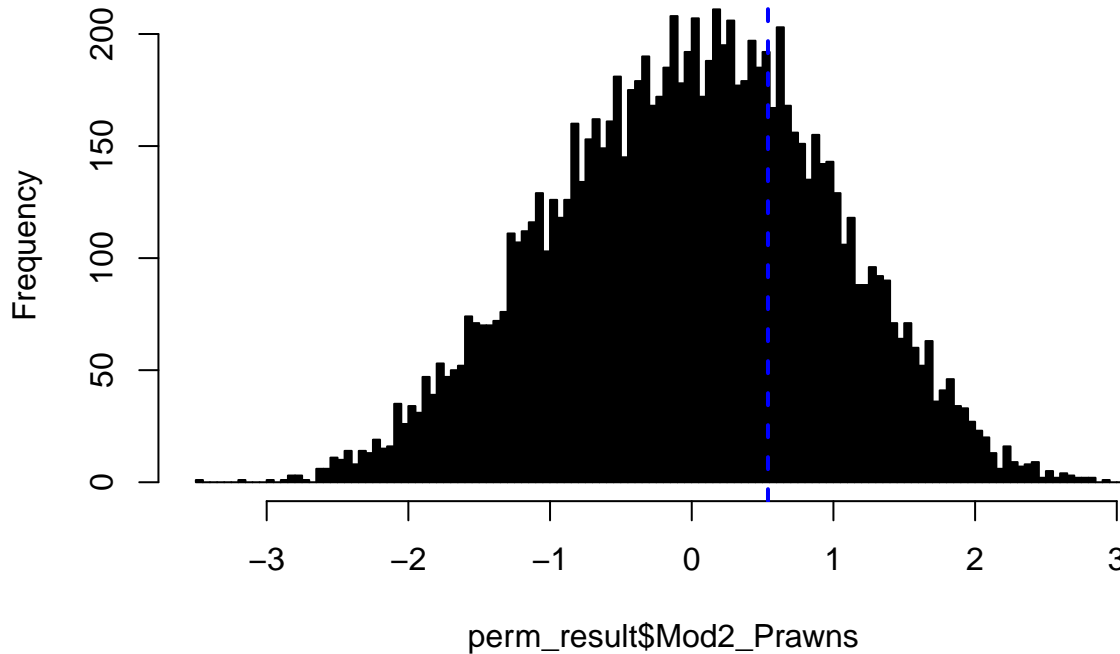


```
length(which(perm_result$Mod2_Net >
             summary(mod2_haem_comm_all)$coef[, "Estimate"] ["Net"]))) / 10000
```

```
## [1] 0.3199
```

```
hist(perm_result$Mod2_Prawns, col = "black", breaks = 100)
abline(v = summary(mod1_haem_comm_all)$coef[, "Estimate"] ["Prawns"],
       col = 4, lty = 2, lwd = 2)
```

## Histogram of perm\_result\$Mod2\_Prawns



```
length(which(perm_result$Mod2_Prawns >
             summary(mod2_haem_comm_all)$coef[, "Estimate"] ["Prawns"]))) / 10000
```

```
## [1] 0.4042
```

Individual level (not really valid since intervention is not defined at individual level)

Schistosoma haematobium

Net effect

To test for a net effect, we'll compare communities which had a net installed in 2016 (therefore influencing egg burden in 2017) to control communities which had no net leading up to 2017

```
#Model 1
mod1_haem_mean_narm_net <- glm.nb(round(s_haem_mean_narm) ~ Net + pzq + extra_pzq,
                                   weights = s_haem_mean_w,
                                   data = full_long %>% filter(extra_pzq != 1 &
                                                                Intervention %in% c("control", "net")))

mod1_haem_narm_net_res <- robust_fx(mod1_haem_mean_narm_net)
mod1_haem_narm_net_res$model <- "Model 1"

#Model 2
mod2_haem_mean_narm_net <- glm.nb(round(s_haem_mean_narm) ~ Net + pzq + School + extra_pzq,
                                   weights = s_haem_mean_w,
                                   data = full_long %>% filter(extra_pzq != 1 &
                                                                Intervention %in% c("control", "net")))

mod2_haem_narm_net_res <- robust_fx(mod2_haem_mean_narm_net)
mod2_haem_narm_net_res$model <- "Model 2"
```

### *#Model 3*

```
mod3_haem_mean_narm_net <- glm.nb(round(s_haem_mean_narm) ~ Net + pzq + School + year_2018 + extra_pzq,  
  weights = s_haem_mean_w,  
  data = full_long %>% filter(extra_pzq != 1 &  
    Intervention %in% c("control", "net")))
```

```
mod3_haem_narm_net_res <- robust_fx(mod3_haem_mean_narm_net)  
mod3_haem_narm_net_res$model <- "Model 3"
```

### *#Model 4*

```
mod4_haem_mean_narm_net <- glm.nb(round(s_haem_mean_narm) ~ Net + School + log_s_haem_mean_narm_l1 + extra_pzq,  
  weights = s_haem_mean_w,  
  data = mod4_dat %>% filter(extra_pzq != 1 &  
    Intervention %in% c("control", "net")))
```

```
mod4_haem_narm_net_res <- robust_fx(mod4_haem_mean_narm_net)  
mod4_haem_narm_net_res$model <- "Model 4"
```

### *# Model 5*

```
mod5_haem_mean_narm_net <- glm.nb(round(s_haem_mean_narm) ~ Net + School + year_2018 + log_s_haem_mean_narm_l1,  
  weights = s_haem_mean_w,  
  data = mod4_dat %>% filter(extra_pzq != 1 &  
    Intervention %in% c("control", "net")))
```

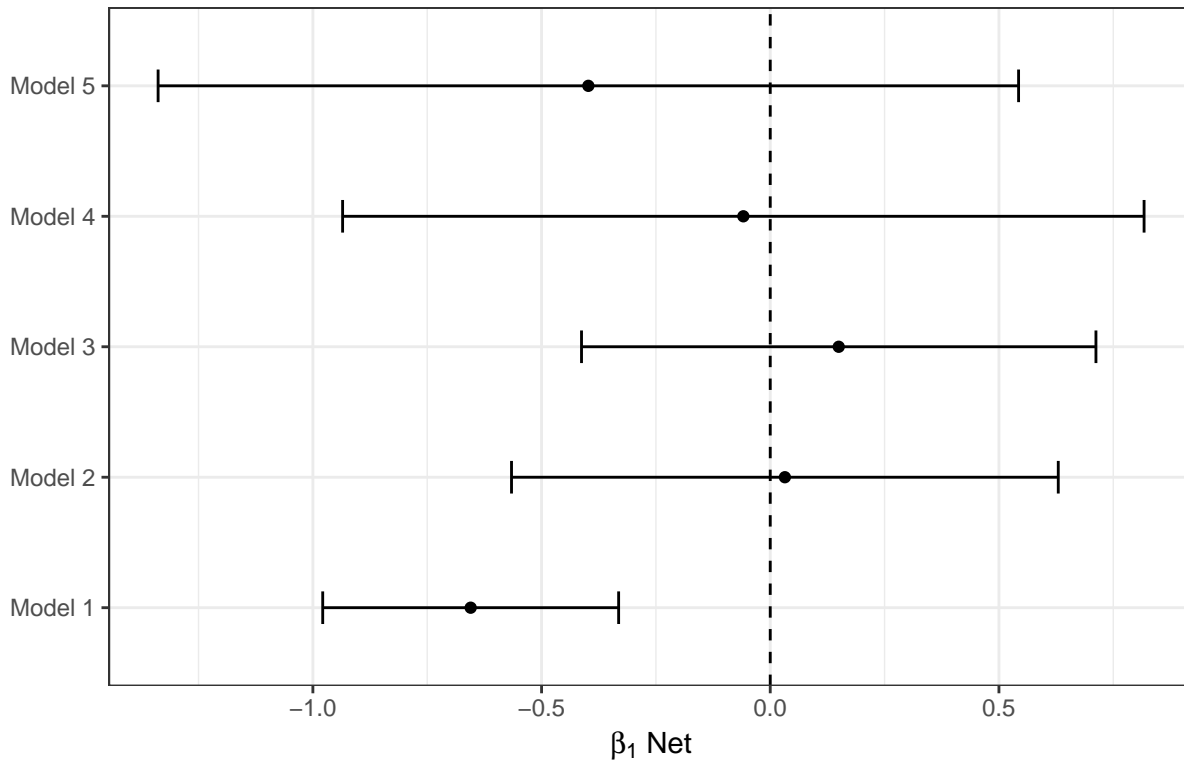
```
mod5_haem_narm_net_res <- robust_fx(mod5_haem_mean_narm_net)  
mod5_haem_narm_net_res$model <- "Model 5"
```

### *#Plot net effect estimate*

```
rbind(mod1_haem_narm_net_res["Net",],  
  mod2_haem_narm_net_res["Net",],  
  mod3_haem_narm_net_res["Net",],  
  mod4_haem_narm_net_res["Net",],  
  mod5_haem_narm_net_res["Net",]) %>%  
ggplot(aes(x = Estimate, y = model)) + theme_bw() +  
  geom_point() +  
  geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = model), height = 0.25) +  
  geom_vline(xintercept = 0, lty = 2) +  
  ylab("") + xlab(expression(paste(beta[1], " Net"))) +  
  ggtitle("S. haematobium net effect",  
    subtitle = "95% CIs derived from robust standard errors")
```

## S. haematobium net effect

95% CIs derived from robust standard errors



## Prawn effect

To test for a prawn effect, we'll compare communities which had prawns introduced in 2017 to control communities which had no net in 2016

```
#Model 1
mod1_haem_mean_narm_prawn <- glm.nb(round(s_haem_mean_narm) ~ Prawns + pzq + extra_pzq,
  data = full_long %>% filter(extra_pzq != 1 &
    Intervention %in% c("control", "prawn")))

mod1_haem_narm_prawn_res <- robust_fx(mod1_haem_mean_narm_prawn)
mod1_haem_narm_prawn_res$model <- "Model 1"

#Model 2
mod2_haem_mean_narm_prawn <- glm.nb(round(s_haem_mean_narm) ~ Prawns + pzq + School + extra_pzq,
  data = full_long %>% filter(extra_pzq != 1 &
    Intervention %in% c("control", "prawn")))

mod2_haem_narm_prawn_res <- robust_fx(mod2_haem_mean_narm_prawn)
mod2_haem_narm_prawn_res$model <- "Model 2"

#Model 3
mod3_haem_mean_narm_prawn <- glm.nb(round(s_haem_mean_narm) ~ Prawns + pzq + School + year_2018 + extra_pzq,
  data = full_long %>% filter(extra_pzq != 1 &
    Intervention %in% c("control", "prawn")))

mod3_haem_narm_prawn_res <- robust_fx(mod3_haem_mean_narm_prawn)
mod3_haem_narm_prawn_res$model <- "Model 3"

#Model 4
mod4_haem_mean_narm_prawn <- glm.nb(round(s_haem_mean_narm) ~ Prawns + School + log_s_haem_mean_narm_l1 + extr
```

```

data = mod4_dat %>% filter(extra_pzq != 1 &
                           Intervention %in% c("control", "prawn"))

mod4_haem_narm_prawn_res <- robust_fx(mod4_haem_mean_narm_prawn)
mod4_haem_narm_prawn_res$model <- "Model 4"

#Model 5
mod5_haem_mean_narm_prawn <- glm.nb(round(s_haem_mean_narm) ~ Prawns + School + year_2018 + log_s_haem_mean_na
data = mod4_dat %>% filter(extra_pzq != 1 &
                           Intervention %in% c("control", "prawn"))

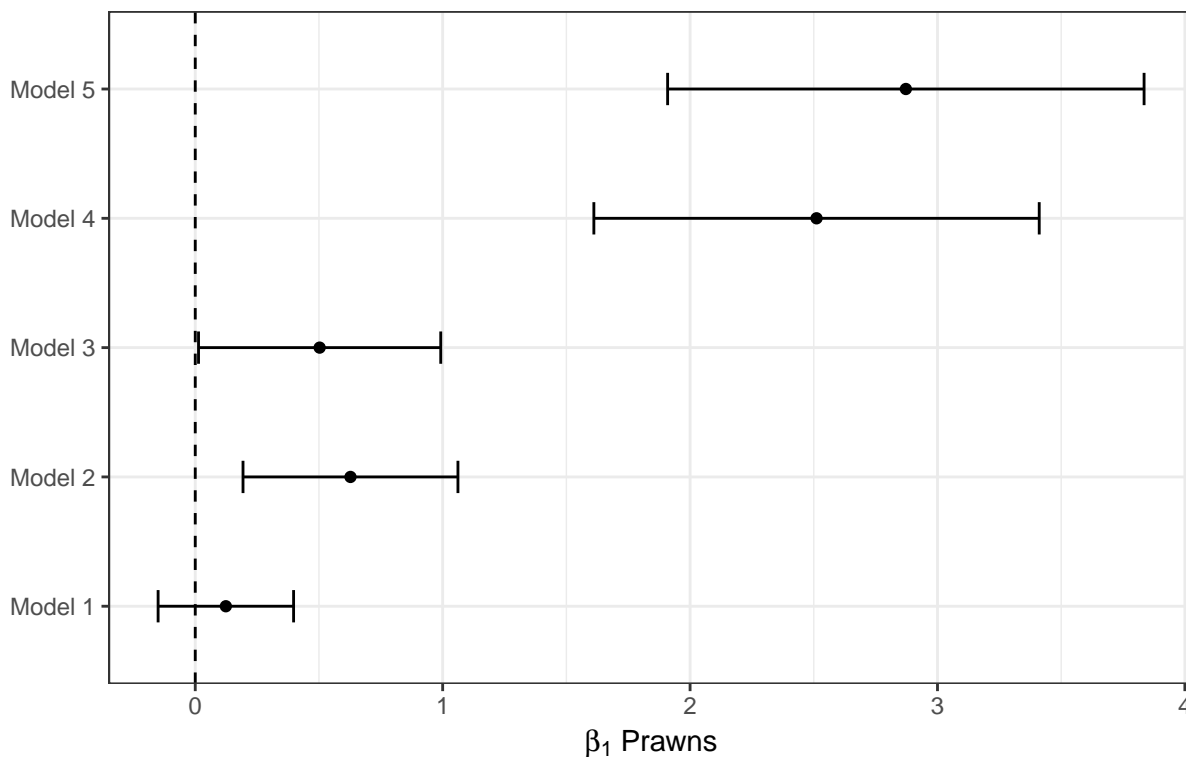
mod5_haem_narm_prawn_res <- robust_fx(mod5_haem_mean_narm_prawn)
mod5_haem_narm_prawn_res$model <- "Model 5"

#Plot prawn effect estimate
rbind(mod1_haem_narm_prawn_res["Prawns",],
      mod2_haem_narm_prawn_res["Prawns",],
      mod3_haem_narm_prawn_res["Prawns",],
      mod4_haem_narm_prawn_res["Prawns",],
      mod5_haem_narm_prawn_res["Prawns",]) %>%
ggplot(aes(x = Estimate, y = model)) + theme_bw() +
  geom_point() +
  geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = model), height = 0.25) +
  geom_vline(xintercept = 0, lty = 2) +
  ylab("") +
  xlab(expression(paste(beta[1], " Prawns"))) +
  ggtitle("S. haematobium prawn effect",
          subtitle = "95% CIs derived from robust standard errors")

```

## S. haematobium prawn effect

95% CIs derived from robust standard errors



In addition to a binary prawn intervention variable, we'll also investigate the continuous effect of prawn biomass measured as kg prawn/m<sup>2</sup>/month

```

#Model 1
mod1_haem_mean_narm_prawn_cat <- glm.nb(round(s_haem_mean_narm) ~ Prawn_cat + pzq,
    data = full_long %>% filter(extra_pzq != 1 &
        Intervention %in% c("control", "prawn")))

mod1_haem_narm_prawn_res_cat <- robust_fx(mod1_haem_mean_narm_prawn_cat)
mod1_haem_narm_prawn_res_cat$model <- "Model 1"

#Model 2
mod2_haem_mean_narm_prawn_cat <- glm.nb(round(s_haem_mean_narm) ~ Prawn_cat + pzq + School,
    data = full_long %>% filter(extra_pzq != 1 &
        Intervention %in% c("control", "prawn")))

mod2_haem_narm_prawn_res_cat <- robust_fx(mod2_haem_mean_narm_prawn_cat)
mod2_haem_narm_prawn_res_cat$model <- "Model 2"

#Model 3
mod3_haem_mean_narm_prawn_cat <- glm.nb(round(s_haem_mean_narm) ~ Prawn_cat + pzq + School + year_2018,
    data = full_long %>% filter(extra_pzq != 1 &
        Intervention %in% c("control", "prawn")))

mod3_haem_narm_prawn_res_cat <- robust_fx(mod3_haem_mean_narm_prawn_cat)
mod3_haem_narm_prawn_res_cat$model <- "Model 3"

#Model 4
mod4_haem_mean_narm_prawn_cat <- glm.nb(round(s_haem_mean_narm) ~ Prawn_cat + School + log_s_haem_mean_narm_l1,
    data = mod4_dat %>% filter(extra_pzq != 1 &
        Intervention %in% c("control", "prawn")))

mod4_haem_narm_prawn_res_cat <- robust_fx(mod4_haem_mean_narm_prawn_cat)
mod4_haem_narm_prawn_res_cat$model <- "Model 4"

#Model 5
mod5_haem_mean_narm_prawn_cat <- glm.nb(round(s_haem_mean_narm) ~ Prawn_cat + School + year_2018 + log_s_haem_mean_narm_l1,
    data = mod4_dat %>% filter(extra_pzq != 1 &
        Intervention %in% c("control", "prawn")))

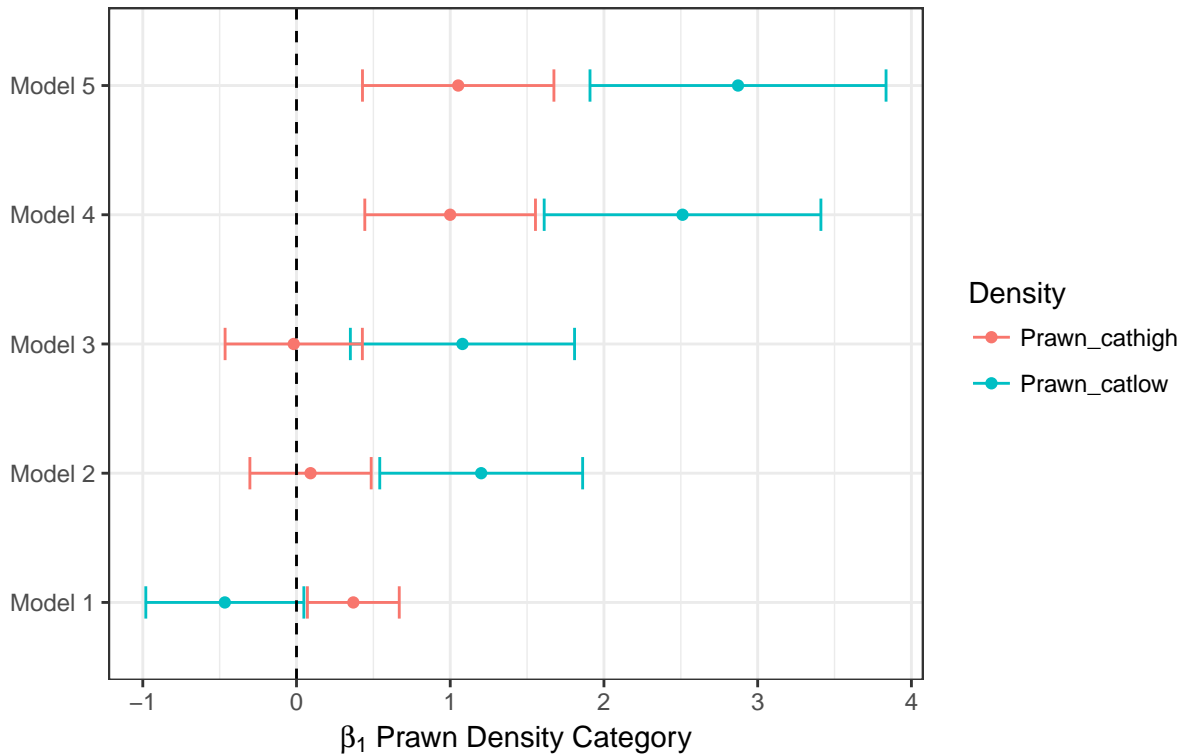
mod5_haem_narm_prawn_res_cat <- robust_fx(mod5_haem_mean_narm_prawn_cat)
mod5_haem_narm_prawn_res_cat$model <- "Model 5"

#Plot prawn effect estimate
rbind(mod1_haem_narm_prawn_res_cat[c("Prawn_catlow", "Prawn_cathigh"),],
    mod2_haem_narm_prawn_res_cat[c("Prawn_catlow", "Prawn_cathigh"),],
    mod3_haem_narm_prawn_res_cat[c("Prawn_catlow", "Prawn_cathigh"),],
    mod4_haem_narm_prawn_res_cat[c("Prawn_catlow", "Prawn_cathigh"),],
    mod5_haem_narm_prawn_res_cat[c("Prawn_catlow", "Prawn_cathigh"),]) %>%
mutate(Density = rep(c("Prawn_catlow", "Prawn_cathigh"), 5)) %>%
ggplot(aes(x = Estimate, y = model, col = Density)) + theme_bw() +
  geom_point() +
  geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = model), height = 0.25) +
  geom_vline(xintercept = 0, lty = 2) +
  ylab("") +
  xlab(expression(paste(beta[1], " Prawn Density Category"))) +
  ggtitle("S. haematobium prawn category effect",
    subtitle = "95% CIs derived from robust standard errors")

```

## S. haematobium prawn category effect

95% CIs derived from robust standard errors



*#Model 1*

```
mod1_haem_mean_narm_prawn_c <- glm.nb(round(s_haem_mean_narm) ~ Prawn_biomass + pzq,
  data = full_long %>% filter(extra_pzq != 1 &
    Intervention %in% c("control", "prawn")))
```

```
mod1_haem_narm_prawn_res_c <- robust_fx(mod1_haem_mean_narm_prawn_c)
mod1_haem_narm_prawn_res_c$model <- "Model 1"
```

*#Model 2*

```
mod2_haem_mean_narm_prawn_c <- glm.nb(round(s_haem_mean_narm) ~ Prawn_biomass + pzq + School,
  data = full_long %>% filter(extra_pzq != 1 &
    Intervention %in% c("control", "prawn")))
```

```
mod2_haem_narm_prawn_res_c <- robust_fx(mod2_haem_mean_narm_prawn_c)
mod2_haem_narm_prawn_res_c$model <- "Model 2"
```

*#Model 3*

```
mod3_haem_mean_narm_prawn_c <- glm.nb(round(s_haem_mean_narm) ~ Prawn_biomass + pzq + School + year_2018,
  data = full_long %>% filter(extra_pzq != 1 &
    Intervention %in% c("control", "prawn")))
```

```
mod3_haem_narm_prawn_res_c <- robust_fx(mod3_haem_mean_narm_prawn_c)
mod3_haem_narm_prawn_res_c$model <- "Model 3"
```

*#Model 4*

```
mod4_haem_mean_narm_prawn_c <- glm.nb(round(s_haem_mean_narm) ~ Prawn_biomass + School + log_s_haem_mean_narm,
  data = mod4_dat %>% filter(extra_pzq != 1 &
    Intervention %in% c("control", "prawn")))
```

```
mod4_haem_narm_prawn_res_c <- robust_fx(mod4_haem_mean_narm_prawn_c)
mod4_haem_narm_prawn_res_c$model <- "Model 4"
```

```

#Model 5
mod5_haem_mean_narm_prawn_c <- glm.nb(round(s_haem_mean_narm) ~ Prawn_biomass + School + year_2018 + log_s_haem
                                data = mod4_dat %>% filter(extra_pzq != 1 &
                                                            Intervention %in% c("control", "prawn")))

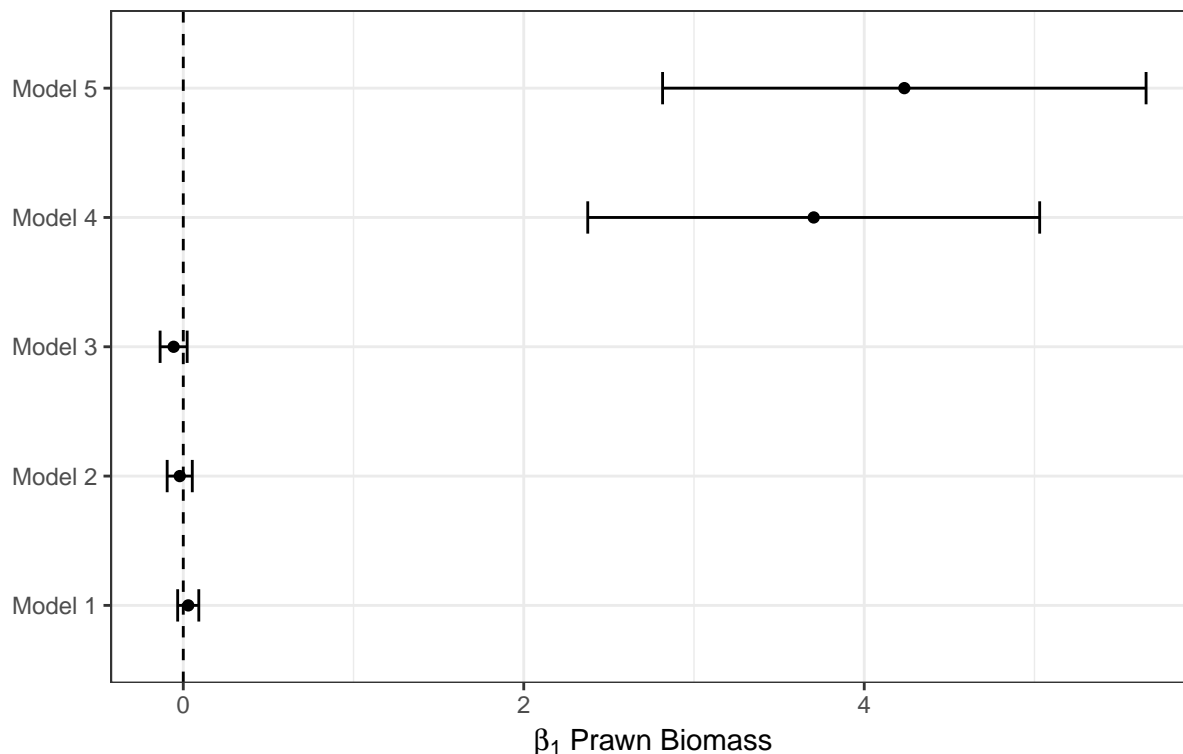
mod5_haem_narm_prawn_res_c <- robust_fx(mod5_haem_mean_narm_prawn_c)
mod5_haem_narm_prawn_res_c$model <- "Model 5"

#Plot prawn effect estimate
rbind(mod1_haem_narm_prawn_res_c["Prawn_biomass",],
      mod2_haem_narm_prawn_res_c["Prawn_biomass",],
      mod3_haem_narm_prawn_res_c["Prawn_biomass",],
      mod4_haem_narm_prawn_res_c["Prawn_biomass",],
      mod5_haem_narm_prawn_res_c["Prawn_biomass",]) %>%
ggplot(aes(x = Estimate, y = model)) + theme_bw() +
  geom_point() +
  geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = model), height = 0.25) +
  geom_vline(xintercept = 0, lty = 2) +
  ylab("") +
  xlab(expression(paste(beta[1], " Prawn Biomass"))) +
  ggtitle("S. haematobium prawn effect",
          subtitle = "95% CIs derived from robust standard errors")

```

## S. haematobium prawn effect

95% CIs derived from robust standard errors



## Vegetation removal effect

```

#Model 1
mod1_haem_mean_narm_veg <- glm.nb(round(s_haem_mean_narm) ~ Veg + pzq,
                                data = full_long %>% filter(extra_pzq != 1 &
                                                            Intervention %in% c("control", "veg_removal")))

```



```

mod1_haem_narm_veg_res <- robust_fx(mod1_haem_mean_narm_veg)
mod1_haem_narm_veg_res$model <- "Model 1"

#Model 2
mod2_haem_mean_narm_veg <- glm.nb(round(s_haem_mean_narm) ~ Veg + pzq + School,
                                   data = full_long %>% filter(extra_pzq != 1 &
                                                             Intervention %in% c("control", "veg_removal")))

mod2_haem_narm_veg_res <- robust_fx(mod2_haem_mean_narm_veg)
mod2_haem_narm_veg_res$model <- "Model 2"

#Model 3
mod3_haem_mean_narm_veg <- glm.nb(round(s_haem_mean_narm) ~ Veg + pzq + School + year_2018,
                                   data = full_long %>% filter(extra_pzq != 1 &
                                                             Intervention %in% c("control", "veg_removal")))

mod3_haem_narm_veg_res <- robust_fx(mod3_haem_mean_narm_veg)
mod3_haem_narm_veg_res$model <- "Model 3"

#Model 4
mod4_haem_mean_narm_veg <- glm.nb(round(s_haem_mean_narm) ~ Veg + School + log_s_haem_mean_narm_l1,
                                   data = mod4_dat %>% filter(extra_pzq != 1 &
                                                             Intervention %in% c("control", "veg_removal")))

mod4_haem_narm_veg_res <- robust_fx(mod4_haem_mean_narm_veg)
mod4_haem_narm_veg_res$model <- "Model 4"

#Model 5
mod5_haem_mean_narm_veg <- glm.nb(round(s_haem_mean_narm) ~ Veg + School + year_2018 + log_s_haem_mean_narm_l1,
                                   data = mod4_dat %>% filter(extra_pzq != 1 &
                                                             Intervention %in% c("control", "veg_removal")))

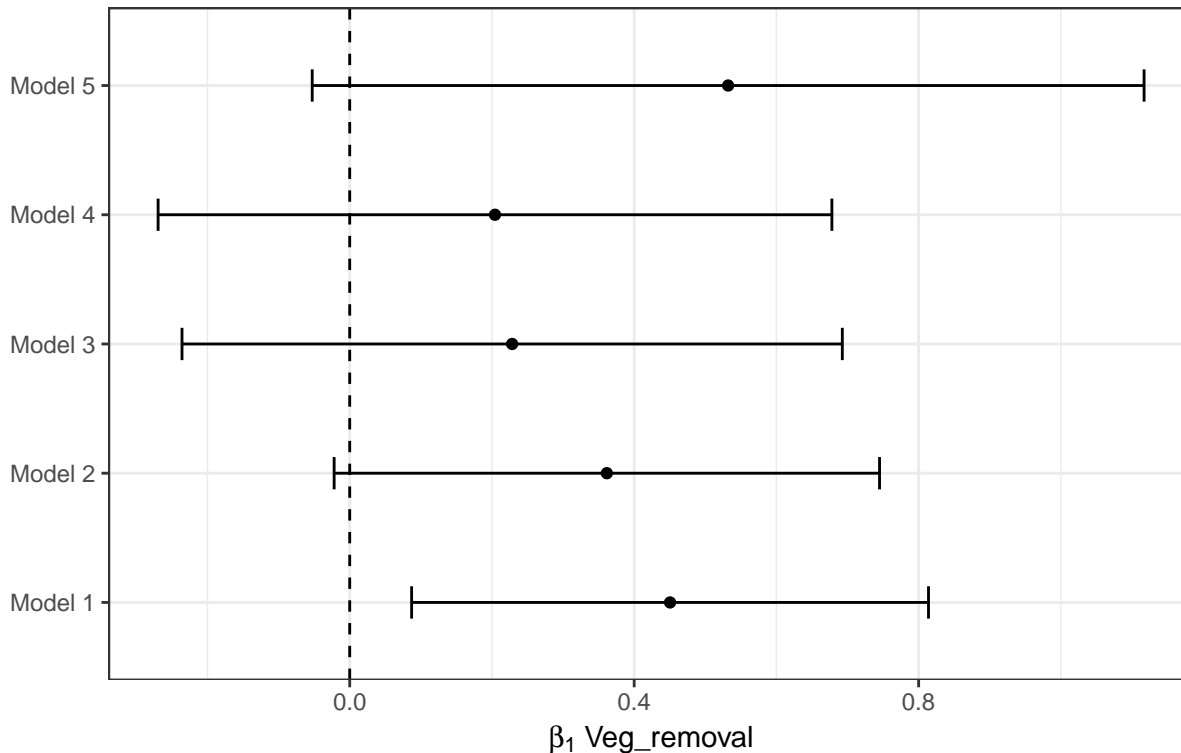
mod5_haem_narm_veg_res <- robust_fx(mod5_haem_mean_narm_veg)
mod5_haem_narm_veg_res$model <- "Model 5"

#Plot net effect estimate
rbind(mod1_haem_narm_veg_res["Veg",],
      mod2_haem_narm_veg_res["Veg",],
      mod3_haem_narm_veg_res["Veg",],
      mod4_haem_narm_veg_res["Veg",],
      mod5_haem_narm_veg_res["Veg",]) %>%
ggplot(aes(x = Estimate, y = model)) + theme_bw() +
  geom_point() +
  geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = model), height = 0.25) +
  geom_vline(xintercept = 0, lty = 2) +
  ylab("") +
  xlab(expression(paste(beta[1], " Veg_removal"))) +
  ggtitle("S. haematobium vegetation removal effect",
          subtitle = "95% CIs derived from robust standard errors")

```

## S. haematobium vegetation removal effect

95% CIs derived from robust standard errors



Also investigate veg removal as a continuous variable: tons of vegetation removed

*#Model 1*

```
mod1_haem_mean_narm_veg_c <- glm.nb(round(s_haem_mean_narm) ~ Veg_tons + pzq,
  weights = s_haem_mean_w,
  data = full_long %>% filter(extra_pzq != 1 &
    Intervention %in% c("control", "veg_removal"))
```

```
mod1_haem_narm_veg_res_c <- robust_fx(mod1_haem_mean_narm_veg_c)
mod1_haem_narm_veg_res_c$model <- "Model 1"
```

*#Model 2*

```
mod2_haem_mean_narm_veg_c <- glm.nb(round(s_haem_mean_narm) ~ Veg_tons + pzq + School,
  weights = s_haem_mean_w,
  data = full_long %>% filter(extra_pzq != 1 &
    Intervention %in% c("control", "veg_removal"))
```

```
mod2_haem_narm_veg_res_c <- robust_fx(mod2_haem_mean_narm_veg_c)
mod2_haem_narm_veg_res_c$model <- "Model 2"
```

*#Model 3*

```
mod3_haem_mean_narm_veg_c <- glm.nb(round(s_haem_mean_narm) ~ Veg_tons + pzq + School + year_2018,
  weights = s_haem_mean_w,
  data = full_long %>% filter(extra_pzq != 1 &
    Intervention %in% c("control", "veg_removal"))
```

```
mod3_haem_narm_veg_res_c <- robust_fx(mod3_haem_mean_narm_veg_c)
mod3_haem_narm_veg_res_c$model <- "Model 3"
```

*#Model 4*

```
mod4_haem_mean_narm_veg_c <- glm.nb(round(s_haem_mean_narm) ~ Veg_tons + School + log_s_haem_mean_narm_l1,
  weights = s_haem_mean_w,
```

```

data = mod4_dat %>% filter(extra_pzq != 1 &
                           Intervention %in% c("control", "veg_removal"))

mod4_haem_narm_veg_res_c <- robust_fx(mod4_haem_mean_narm_veg_c)
mod4_haem_narm_veg_res_c$model <- "Model 4"

#Model 5
mod5_haem_mean_narm_veg_c <- glm.nb(round(s_haem_mean_narm) ~ Veg_tons + School + year_2018 + log_s_haem_mean_narm,
                                     weights = s_haem_mean_w,
                                     data = mod4_dat %>% filter(extra_pzq != 1 &
                                                               Intervention %in% c("control", "veg_removal")))

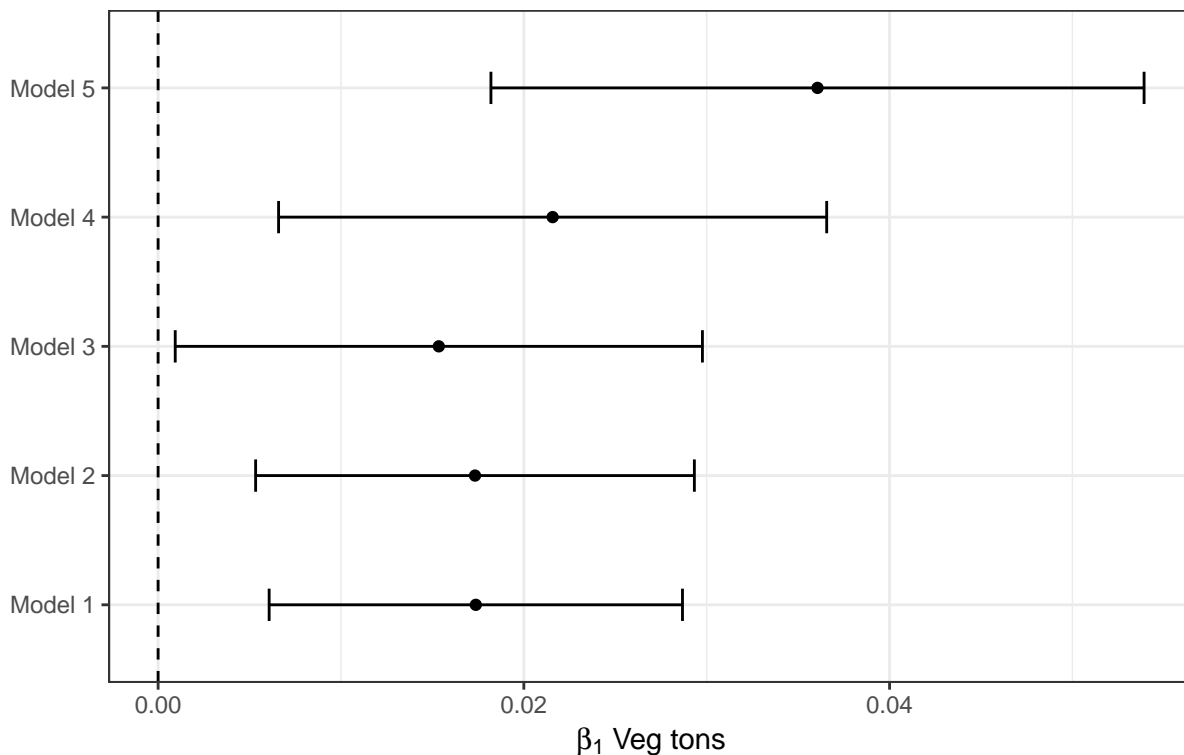
mod5_haem_narm_veg_res_c <- robust_fx(mod5_haem_mean_narm_veg_c)
mod5_haem_narm_veg_res_c$model <- "Model 5"

#Plot net effect estimate
rbind(mod1_haem_narm_veg_res_c["Veg_tons",],
      mod2_haem_narm_veg_res_c["Veg_tons",],
      mod3_haem_narm_veg_res_c["Veg_tons",],
      mod4_haem_narm_veg_res_c["Veg_tons",],
      mod5_haem_narm_veg_res_c["Veg_tons",]) %>%
ggplot(aes(x = Estimate, y = model)) + theme_bw() +
  geom_point() +
  geom_errorbarh(aes(xmin = rob_lo, xmax = rob_up, y = model), height = 0.25) +
  geom_vline(xintercept = 0, lty = 2) +
  ylab("") +
  xlab(expression(paste(beta[1], " Veg tons"))) +
  ggtitle("S. haematobium (continuous) vegetation removal effect",
          subtitle = "95% CIs derived from robust standard errors")

```

## S. haematobium (continuous) vegetation removal effect

95% CIs derived from robust standard errors



All intervention variables

```

#Model 0
mod0_haem_mean_narm_all <- glm.nb(round(s_haem_mean_narm) ~ Net + Veg + Prawns + School + year_fac,
                                   weights = s_haem_mean_w,
                                   data = full_long %>% filter(extra_pzq != 1))

mod0_haem_narm_all_res <- robust_fx(mod0_haem_mean_narm_all)
mod0_haem_narm_all_res$model <- "Model 0"

#Model 1
mod1_haem_mean_narm_all <- glm.nb(round(s_haem_mean_narm) ~ Net + Veg + Prawns + pzq,
                                   weights = s_haem_mean_w,
                                   data = full_long %>% filter(extra_pzq != 1))

mod1_haem_narm_all_res <- robust_fx(mod1_haem_mean_narm_all)
mod1_haem_narm_all_res$model <- "Model 1"

#Model 2
mod2_haem_mean_narm_all <- glm.nb(round(s_haem_mean_narm) ~ Net + Veg + Prawns + pzq + School,
                                   weights = s_haem_mean_w,
                                   data = full_long %>% filter(extra_pzq != 1))

mod2_haem_narm_all_res <- robust_fx(mod2_haem_mean_narm_all)
mod2_haem_narm_all_res$model <- "Model 2"

#Model 3
mod3_haem_mean_narm_all <- glm.nb(round(s_haem_mean_narm) ~ Net + Veg + Prawns + pzq + School + year_2018,
                                   weights = s_haem_mean_w,
                                   data = full_long %>% filter(extra_pzq != 1))

mod3_haem_narm_all_res <- robust_fx(mod3_haem_mean_narm_all)
mod3_haem_narm_all_res$model <- "Model 3"

#Model 4
mod4_haem_mean_narm_all <- glm.nb(round(s_haem_mean_narm) ~ Net + Veg + Prawns + School + log_s_haem_mean_narm,
                                   weights = s_haem_mean_w,
                                   data = mod4_dat %>% filter(extra_pzq != 1))

mod4_haem_narm_all_res <- robust_fx(mod4_haem_mean_narm_all)
mod4_haem_narm_all_res$model <- "Model 4"

#Model 5
mod5_haem_mean_narm_all <- glm.nb(round(s_haem_mean_narm) ~ Net + Veg + Prawns + School + year_2018 + log_s_haem_mean_narm,
                                   weights = s_haem_mean_w,
                                   data = mod4_dat %>% filter(extra_pzq != 1))

mod5_haem_narm_all_res <- robust_fx(mod5_haem_mean_narm_all)
mod5_haem_narm_all_res$model <- "Model 5"

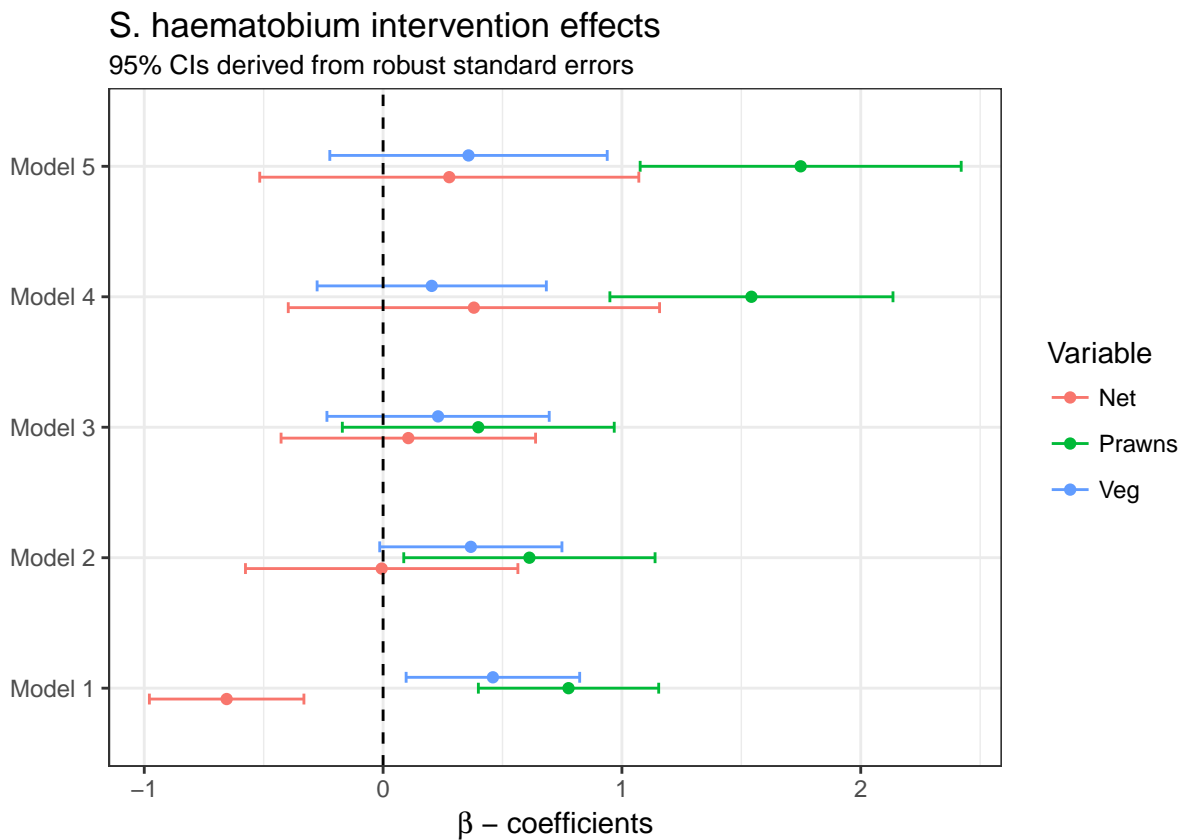
#Plot intervention effect estimates
rbind(mod1_haem_narm_all_res[c("Net", "Veg", "Prawns"),],
      mod2_haem_narm_all_res[c("Net", "Veg", "Prawns"),],
      mod3_haem_narm_all_res[c("Net", "Veg", "Prawns"),],
      mod4_haem_narm_all_res[c("Net", "Veg", "Prawns"),],
      mod5_haem_narm_all_res[c("Net", "Veg", "Prawns"),]) %>%
mutate(Variable = rep(c("Net", "Veg", "Prawns"), 5)) %>%
ggplot(aes(y = Estimate, x = model, col = Variable)) + theme_bw() +
  geom_point(position = position_dodge(width = 0.25)) +
  geom_errorbar(aes(ymin = rob_lo, ymax = rob_up, x = model),

```

```

width = 0.25, position = position_dodge(width = 0.25)) +
geom_hline(yintercept = 0, lty = 2) + coord_flip() +
xlab("") +
ylab(expression(paste(beta, " - coefficients")) +
ggtitle("S. haematobium intervention effects",
        subtitle = "95% CIs derived from robust standard errors")

```



## Schistosoma mansoni

### Net effect

```

#Model 1
mod1_mans_mean_narm_net <- glm.nb(round(s_mans_mean_narm) ~ Net + pzq,
                                   weights = s_mans_mean_w,
                                   data = full_long %>% filter(extra_pzq != 1 &
                                                                Intervention %in% c("control", "net") &
                                                                year != 2018))

mod1_mans_narm_net_res <- as.data.frame(round(cbind(summary(mod1_mans_mean_narm_net)$coef,
                                                    confint(mod1_mans_mean_narm_net)), 3)[,c(1,2,5,6,3,4)])
mod1_mans_narm_net_res$model <- "Model 1"

#Model 2
mod2_mans_mean_narm_net <- glm.nb(round(s_mans_mean_narm) ~ Net + pzq + School,
                                   weights = s_mans_mean_w,
                                   data = full_long %>% filter(extra_pzq != 1 &
                                                                Intervention %in% c("control", "net") &
                                                                year != 2018))

```

```

mod2_mans_narm_net_res <- as.data.frame(round(cbind(summary(mod2_mans_mean_narm_net)$coef,
                                                    confint(mod2_mans_mean_narm_net)), 3)[,c(1,2,5,6,3,4)])

mod2_mans_narm_net_res$model <- "Model 2"

#Model 3
mod3_mans_mean_narm_net <- glm.nb(round(s_mans_mean_narm) ~ Net + School + log_s_mans_mean12_l1,
                                   weights = s_mans_mean_w,
                                   data = mod4_dat %>% filter(extra_pzq != 1 &
                                                             Intervention %in% c("control", "net") &
                                                             year != 2018),
                                   control = glm.control(maxit = 300))

mod3_mans_narm_net_res <- as.data.frame(round(cbind(summary(mod3_mans_mean_narm_net)$coef,
                                                    na.omit(confint(mod3_mans_mean_narm_net))), 3)[,c(1,2,5,6,3,4)])

mod3_mans_narm_net_res$model <- "Model 3"

#Plot net effect estimate
rbind(mod1_mans_narm_net_res["Net",], mod2_mans_narm_net_res["Net",], mod3_mans_narm_net_res["Net",]) %>%
  rename("lower" = !!names(.[3]),
         "upper" = !!names(.[4])) %>%
  ggplot(aes(x = exp(Estimate), y = model)) + theme_bw() +
  geom_point() + geom_errorbarh(aes(xmin = exp(lower), xmax = exp(upper), y = model), height = 0.25) +
  geom_vline(xintercept = 1, lty = 2) +
  ylab("") + xlab(expression(paste(beta[1], " - Additional eggs associated with net presence"))) +
  ggtitle("S. mansoni net effect")

```

## Prawn effect

```

#Model 1
mod1_mans_mean_narm_prawn <- glm.nb(round(s_mans_mean_narm) ~ Prawns + pzq,
                                   weights = s_mans_mean_w,
                                   data = full_long %>% filter(extra_pzq != 1 &
                                                             Intervention %in% c("control", "prawn")))

mod1_mans_narm_prawn_res <- as.data.frame(round(cbind(summary(mod1_mans_mean_prawn)$coef,
                                                    confint(mod1_mans_mean_prawn)), 3)[,c(1,2,5,6,3,4)])

mod1_mans_narm_prawn_res$model <- "Model 1"

#Model 2
mod2_mans_mean_narm_prawn <- glm.nb(round(s_mans_mean_narm) ~ Prawns + pzq + School,
                                   weights = s_mans_mean_w,
                                   data = full_long %>% filter(extra_pzq != 1 &
                                                             Intervention %in% c("control", "prawn")))

mod2_mans_narm_prawn_res <- as.data.frame(round(cbind(summary(mod2_mans_mean_narm_prawn)$coef,
                                                    confint(mod2_mans_mean_narm_prawn)), 3)[,c(1,2,5,6,3,4)])

mod2_mans_narm_prawn_res$model <- "Model 2"

#Model 3
mod3_mans_mean_narm_prawn <- glm.nb(round(s_mans_mean_narm) ~ Prawns + School + log_s_mans_mean_narm_l1,
                                   weights = s_mans_mean_w,
                                   data = mod4_dat %>% filter(extra_pzq != 1 &
                                                             Intervention %in% c("control", "prawn")),
                                   control = glm.control(maxit = 200))

mod3_mans_narm_prawn_res <- as.data.frame(round(cbind(summary(mod3_mans_mean_narm_prawn)$coef,
                                                    confint(mod3_mans_mean_narm_prawn)), 3)[,c(1,2,5,6,3,4)])

```

```

mod3_mans_narm_prawn_res$model <- "Model 3"

#Plot prawn effect estimate
rbind(mod1_mans_narm_prawn_res["Prawns",],
      mod2_mans_narm_prawn_res["Prawns",],
      mod3_mans_narm_prawn_res["Prawns",]) %>%
  rename("lower" = !!names(.[3]),
         "upper" = !!names(.[4])) %>%
  ggplot(aes(x = exp(Estimate), y = model)) + theme_bw() +
  geom_point() + geom_errorbarh(aes(xmin = exp(lower), xmax = exp(upper), y = model), height = 0.25) +
  geom_vline(xintercept = 1, lty = 2) +
  ylab("") + xlab(expression(paste(beta[1], " - Additional eggs associated with prawn addition"))) +
  ggtitle("S. mansoni prawn effect")

#Model 1
mod1_mans_mean_prawn_c <- glm.nb(round(s_mans_mean12) ~ Prawn_biomass + pzq + sex,
                                data = full_long %>% filter(extra_pzq != 1 & Intervention %in% c("control", "prawn")))

mod1_mans_prawn_res_c <- as.data.frame(round(cbind(summary(mod1_mans_mean_prawn_c)$coef,
                                                  confint(mod1_mans_mean_prawn_c)), 3)[,c(1,2,5,6,3,4)])

mod1_mans_prawn_res_c$model <- "Model 1"

#Model 2
mod2_mans_mean_prawn_c <- glm.nb(round(s_mans_mean12) ~ Prawn_biomass + pzq + sex + School,
                                data = full_long %>% filter(extra_pzq != 1 & Intervention %in% c("control", "prawn")))

mod2_mans_prawn_res_c <- as.data.frame(round(cbind(summary(mod2_mans_mean_prawn_c)$coef,
                                                  confint(mod2_mans_mean_prawn_c)), 3)[,c(1,2,5,6,3,4)])

mod2_mans_prawn_res_c$model <- "Model 2"

#Model 3
mod3_mans_mean_prawn_c <- glm.nb(round(s_mans_mean12) ~ Prawn_biomass + sex + School + log_s_mans_mean12_l1,
                                data = mod4_dat %>% filter(extra_pzq != 1 & Intervention %in% c("control", "prawn")))

mod3_mans_prawn_res_c <- as.data.frame(round(cbind(summary(mod3_mans_mean_prawn_c)$coef,
                                                  confint(mod3_mans_mean_prawn_c)), 3)[,c(1,2,5,6,3,4)])

mod3_mans_prawn_res_c$model <- "Model 3"

#Plot prawn effect estimate
rbind(mod1_mans_prawn_res_c["Prawn_biomass",], mod2_mans_prawn_res_c["Prawn_biomass",], mod3_mans_prawn_res_c["Prawn_biomass",]) %>%
  rename("lower" = !!names(.[3]),
         "upper" = !!names(.[4])) %>%
  ggplot(aes(x = exp(Estimate), y = model)) + theme_bw() +
  geom_point() + geom_errorbarh(aes(xmin = exp(lower), xmax = exp(upper), y = model), height = 0.25) +
  geom_vline(xintercept = 1, lty = 2) +
  ylab("") + xlab(expression(paste(beta[1], " - Additional eggs associated with unit increase in prawn biomass"))) +
  ggtitle("S. mansoni prawn effect")

```

## Scratch

```

#Global model
mod1_haem_mean <- glm.nb(round(s_haem_mean) ~ Intervention + pzq + sex,
                        data = full_long %>% filter(extra_pzq != 1))

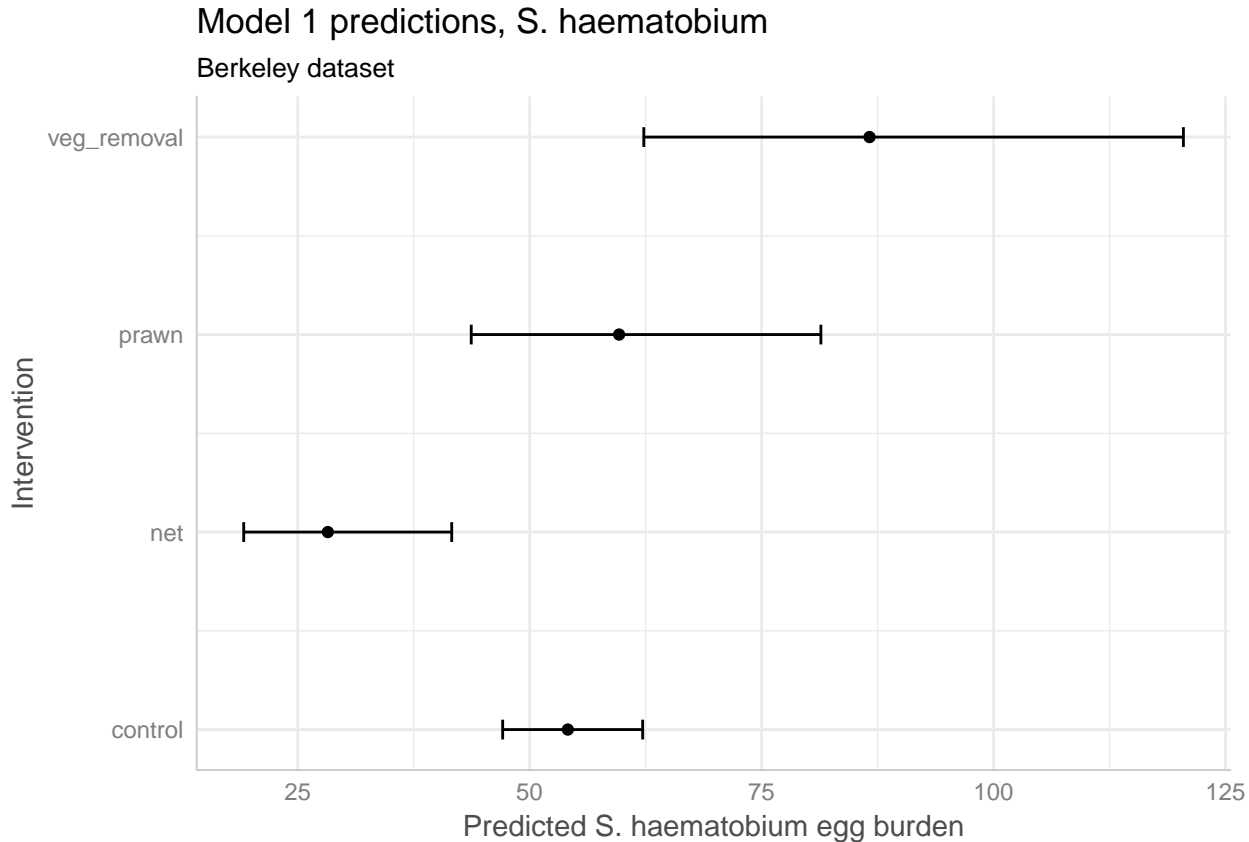
round(cbind(summary(mod1_haem_mean)$coef, confint(mod1_haem_mean)), 3)[,c(1,2,5,6,3,4)]

```

```
## Waiting for profiling to be done...
```

```
##           Estimate Std. Error  2.5 % 97.5 % z value Pr(>|z|)
## (Intercept)      4.013      0.072   3.873  4.158  55.554   0.000
## Interventionnet   -0.650      0.184  -0.995 -0.272  -3.535   0.000
## Interventionprawn  0.097      0.142  -0.173  0.383   0.687   0.492
## Interventionveg_removal 0.471      0.153  0.179  0.782   3.069   0.002
## pzq1             -0.380      0.086  -0.549 -0.211  -4.405   0.000
## sexM             -0.022      0.077  -0.173  0.128  -0.289   0.772
```

```
plot(ggpredict(mod1_haem_mean, c("Intervention"))) + coord_flip() +
  ylab("Predicted S. haematobium egg burden") + ggtitle("Model 1 predictions, S. haematobium", subtitle = "B
```



```
#Compare prawn biomass to only controls
mod1_haem_mean_prawn_bm <- glm.nb(round(s_haem_mean) ~ Prawn_biomass + year_fac + sex,
                                   data = full_long %>% filter(extra_pzq != 1 & Intervention %in% c("control",
round(cbind(summary(mod1_haem_mean_prawn_bm)$coef, confint(mod1_haem_mean_prawn_bm)), 3)[,c(1,2,5,6,3,4)]
```

```
## Waiting for profiling to be done...
```

```
##           Estimate Std. Error  2.5 % 97.5 % z value Pr(>|z|)
## (Intercept)      4.025      0.074   3.882  4.173  54.726   0.000
## Prawn_biomass     0.038      0.036  -0.030  0.115   1.061   0.289
## year_fac2017     -0.271      0.104  -0.473 -0.064  -2.599   0.009
## year_fac2018     -0.473      0.100  -0.669 -0.274  -4.722   0.000
## sexM             -0.044      0.082  -0.204  0.117  -0.534   0.593
```



```
#Compare prawn biomass to only controls with net effect
mod1_haem_mean_prawn_bm_net <- glm.nb(round(s_haem_mean) ~ Prawn_biomass + Net + year_fac + sex,
                                     data = full_long %>% filter(extra_pzq != 1 & Intervention %in% c("control",
round(cbind(summary(mod1_haem_mean_prawn_bm_net)$coef, confint(mod1_haem_mean_prawn_bm_net)), 3)[,c(1,2,5,6,
```

```
## Waiting for profiling to be done...
```

	Estimate	Std. Error	2.5 %	97.5 %	z value	Pr(> z )
## (Intercept)	4.026	0.073	3.884	4.174	55.011	0.000
## Prawn_biomass	0.074	0.041	-0.008	0.168	1.797	0.072
## Net	-0.238	0.141	-0.519	0.055	-1.695	0.090
## year_fac2017	-0.353	0.101	-0.547	-0.157	-3.488	0.000
## year_fac2018	-0.419	0.104	-0.627	-0.205	-4.021	0.000
## sexM	-0.046	0.080	-0.203	0.110	-0.579	0.562

```
#Compare veg removal to controls
mod1_haem_mean_veg <- glm.nb(round(s_haem_mean) ~ Veg + year_fac + sex,
                             data = full_long %>% filter(extra_pzq != 1 & Intervention %in% c("control", "veg_
round(cbind(summary(mod1_haem_mean_veg)$coef, confint(mod1_haem_mean_veg)), 3)[,c(1,2,5,6,3,4)]
```

```
## Waiting for profiling to be done...
```

	Estimate	Std. Error	2.5 %	97.5 %	z value	Pr(> z )
## (Intercept)	3.998	0.074	3.856	4.146	54.285	0.000
## Veg	0.605	0.166	0.286	0.938	3.651	0.000
## year_fac2017	-0.268	0.104	-0.470	-0.062	-2.577	0.010
## year_fac2018	-0.517	0.107	-0.725	-0.304	-4.819	0.000
## sexM	0.006	0.082	-0.155	0.168	0.078	0.938

```
#Rerun with stanford variables
mod1_haem_mean_stan <- glm.nb(round(ShW) ~ pzq + net + prawn + veg_removal + sex,
                              data = stanford %>% filter(Tx_early != 1))
round(cbind(summary(mod1_haem_mean_stan)$coef, confint(mod1_haem_mean_stan)), 3)[,c(1,2,5,6,3,4)]
```

```
## Waiting for profiling to be done...
```

	Estimate	Std. Error	2.5 %	97.5 %	z value	Pr(> z )
## (Intercept)	3.960	0.082	3.802	4.123	48.535	0.000
## pzq	-0.323	0.078	-0.478	-0.170	-4.154	0.000
## net	-0.367	0.105	-0.644	-0.090	-3.499	0.000
## prawn	0.208	0.099	-0.051	0.475	2.100	0.036
## veg_removal	0.241	0.097	0.055	0.432	2.478	0.013
## sexM	-0.020	0.076	-0.171	0.130	-0.264	0.792

For *S. haematobium*, these results show a significant effect of praziquantel administration, as we'd expect, but show no significant effect of the prawn or vegetation removal interventions. A protective effect of prawns and a harmful effect of vegetation removal are borderline insignificant. Individuals in the single community that had a net intervention (e.g. net but no prawns) is driving the significant net effect, but this community is a low transmission environment in general, so this is not surprising

## Schistosoma mansoni

```
mod1_mans_mean <- glm.nb(round(s_mans_mean12) ~ pzq + Intervention + sex,
  data = full_long %>% filter(extra_pzq != 1))

round(cbind(summary(mod1_mans_mean)$coef, confint(mod1_mans_mean)), 3)[,c(1,2,5,6,3,4)]

plot(ggpredict(mod1_mans_mean, c("Intervention")) +
  ylab("Predicted S. mansoni egg burden") + ggtitle("Model 1 predictions, S. mansoni"))
```

For *S. mansoni*, these results again show a significant protective effect of praziquantel administration, as we'd expect. They also show the same net effect, that should not be interpreted too strongly. However, these results do imply a significant protective effect of vegetation removal and significant harmful effect of prawn addition.

### Schistosoma haematobium

```
mod2_haem_mean <- glm.nb(round(s_haem_mean) ~ pzq + Intervention + School + sex,
  data = full_long %>% filter(extra_pzq != 1))

round(cbind(summary(mod2_haem_mean)$coef, na.omit(confint(mod2_haem_mean))), 3)[,c(1,2,5,6,3,4)]

plot(ggpredict(mod2_haem_mean, c("Intervention")) +
  ylab("Predicted S. haematobium egg burden") + ggtitle("Model 2 predictions, S. haematobium"))

#Compare prawn biomass to only controls
mod2_haem_mean_prawn_bm <- glm.nb(round(s_haem_mean) ~ Prawn_biomass + year_fac + sex + School,
  data = full_long %>% filter(extra_pzq != 1 & Intervention %in% c("control",

round(cbind(summary(mod2_haem_mean_prawn_bm)$coef, confint(mod2_haem_mean_prawn_bm)), 3)[,c(1,2,5,6,3,4)]

#Compare prawn biomass to only controls with net effect
mod2_haem_mean_prawn_bm_net <- glm.nb(round(s_haem_mean) ~ Prawn_biomass + Net + year_fac + sex + School,
  data = full_long %>% filter(extra_pzq != 1 & Intervention %in% c("control",

round(cbind(summary(mod2_haem_mean_prawn_bm_net)$coef, confint(mod2_haem_mean_prawn_bm_net)), 3)[,c(1,2,5,6,3,4)]

#Compare veg removal to controls
mod2_haem_mean_veg <- glm.nb(round(s_haem_mean) ~ Veg + year_fac + sex + School,
  data = full_long %>% filter(extra_pzq != 1 & Intervention %in% c("control", "veg_

round(cbind(summary(mod2_haem_mean_veg)$coef, confint(mod2_haem_mean_veg)), 3)[,c(1,2,5,6,3,4)]
```

For *S. haematobium* these results show that there is a large degree of inter-community variability in transmission that influences estimates of  $\beta_1$  our target parameter. Adding the term on communities accounts for baseline differences in transmission intensity that may obscure our ability to detect an effect of a particular intervention. Accounting for this variability leads to a significantly protective prawn effect and significantly harmful vegetation effect. The net effect also remains, implying there may actually be a protective net effect that is not explained by that community's low egg burden alone. This may also imply that the prawn effect is actually driven by a net effect. To dig into this a bit deeper, we'll add another model that considers prawn density as a continuous variable (model 2.1 below). This is better since we're treating villages as their own entities, but we're still treating individual egg burdens as independent when we know that individuals are likely correlated. Model 3 will address this concern.

```
mod2_mans_mean <- glm.nb(round(s_mans_mean12) ~ pzq + Intervention + School + sex,
  data = full_long %>% filter(extra_pzq != 1))

round(cbind(summary(mod2_mans_mean)$coef, na.omit(confint(mod2_mans_mean))), 3)[,c(1,2,5,6,3,4)]

plot(ggpredict(mod2_mans_mean, c("Intervention")) +
  ylab("Predicted S. mansoni egg burden") + ggtitle("Model 2 predictions, S. mansoni"))
```

For *S. mansoni*, the protective effect of vegetation removal and harmful effect of prawn addition are both eliminated when including community effects, implying that variability in transmission intensity between communities may explain these effects.

This is better since we're treating villages as their own entities, but we're still treating individual egg burdens as independent when we know that individuals are likely correlated. Model 3 will address this concern.

## Model 2.1: Further exploration of effect of interventions on egg burden controlling for inter-village variability

*#Note: observations from 2016 should automatically be omitted since the lag variable will be NA, but let's check*  
 mod4\_dat %>% group\_by(year) %>% summarise(sum(!is.na(log\_s\_haem\_mean\_l1)))

*#Confirmed, no non-NA observations in 2016*

```
mod3_haem_mean <- glm.nb(round(s_haem_mean) ~ Intervention + School + log_s_haem_mean_l1, data = mod4_dat)
round(cbind(summary(mod3_haem_mean)$coef, na.omit(confint(mod3_haem_mean))), 3)[,c(1,2,5,6,3,4)]
plot(ggpredict(mod3_haem_mean, c("Intervention")) +
  ylab("Predicted S. haematobium egg burden") + ggtitle("Model 3 predictions, S. haematobium"))
plot(ggpredict(mod3_haem_mean, c("log_s_haem_mean_l1", "Intervention"))) +
  ylab("Predicted S. haematobium egg burden") + ggtitle("Model 3 predictions, S. haematobium")
```

```
mod3_haem_mean <- glm.nb(round(s_haem_mean) ~ Veg + Net + Prawn_biomass + School + log_s_haem_mean_l1, data = mod4_dat)
round(cbind(summary(mod3_haem_mean)$coef, na.omit(confint(mod3_haem_mean))), 3)[,c(1,2,5,6,3,4)]
plot(ggpredict(mod3_haem_mean, c("Prawn_biomass"))) +
  ylab("Predicted S. haematobium egg burden") + ggtitle("Model 3 predictions, S. haematobium")
```

*#Compare prawn biomass to only controls*

```
mod3_haem_mean_prawn_bm <- glm.nb(round(s_haem_mean) ~ Prawn_biomass + sex + School + log_s_haem_mean_l1,
  data = mod4_dat %>% filter(extra_pzq != 1 & Intervention %in% c("control", "veg_r"))
round(cbind(summary(mod3_haem_mean_prawn_bm)$coef, confint(mod3_haem_mean_prawn_bm)), 3)[,c(1,2,5,6,3,4)]
```

*#Compare prawn biomass to only controls with net effect*

```
mod3_haem_mean_prawn_bm_net <- glm.nb(round(s_haem_mean) ~ Prawn_biomass + Net + sex + School + log_s_haem_mean_l1,
  data = mod4_dat %>% filter(extra_pzq != 1 & Intervention %in% c("control", "veg_r")))
round(cbind(summary(mod3_haem_mean_prawn_bm_net)$coef, confint(mod3_haem_mean_prawn_bm_net)), 3)[,c(1,2,5,6,3,4)]
```

*#Compare veg removal to controls*

```
mod3_haem_mean_veg <- glm.nb(round(s_haem_mean) ~ Veg + sex + School + log_s_haem_mean_l1,
  data = mod4_dat %>% filter(extra_pzq != 1 & Intervention %in% c("control", "veg_r")))
round(cbind(summary(mod3_haem_mean_veg)$coef, confint(mod3_haem_mean_veg)), 3)[,c(1,2,5,6,3,4)]
```

*#see if using only first haematobium urine sample makes a difference since using mean introduces some missing*

```
mod3_haem1 <- glm.nb(s_haem1 ~ Intervention + School + log_s_haem1_l1, data = mod4_dat)
round(cbind(summary(mod3_haem1)$coef, na.omit(confint(mod3_haem1))), 3)[,c(1,2,5,6,3,4)]
```

*#Doesn't qualitatively change results*

```
mod3_mans_mean <- glm.nb(s_mans_mean12 ~ Intervention + School + log_s_mans_mean12_l1, data = mod4_dat)
round(cbind(summary(mod3_mans_mean)$coef, na.omit(confint(mod3_mans_mean))), 3)[,c(1,2,5,6,3,4)]
plot(ggpredict(mod3_mans_mean, c("log_s_mans_mean12_l1", "Intervention"))) +
  ylab("Predicted S. mansoni egg burden") + ggtitle("Model 3 predictions, S. mansoni")
```

For each independent variable of interest (e.g. interventions) there are a couple of measurements of interest we might consider:

## **Net addition**

- Binary (net installed or no)
- Continuous-ish (months with net installed over reinfection period)
- Continuous (area enclosed by net)

I think binary makes the most sense here as I'm not sure how different n

## **Prawn stocking**

- Binary (prawns stocked or no)
- Categorical (no prawns, low or high stocking density)
- Continuous (mean monthly prawn density over reinfection period)

I think it's worth exploring each of these sequentially

## **Vegetation removal**

- Binary (Vegetation removal or no)
- Continuous (tons of vegetation removed over reinfection period)

I think binary makes sense here since I believe all vegetation possible was removed at each site

We'll treat egg burden of each Schistosome species as our dependent variable in separate analyses

## **Old Models**

## **Appendix**

A litany of data cleaning/aggregating/integrating notes:

- + Months during reinfection with net added were calculated as inclusive of the month the net was added, but excluded the month when infection was estimated
- + Mean monthly prawn density was estimated as the mean monthly density over the number of months in the reinfection period
- + Mean monthly prawn density for communities with more than one water access point were estimated as an average of all water access points
- + Vegetation removal in communities with multiple water contact sites was estimated as the mean total tons of vegetation removed per site