Varroa sup1

Kevin Healy

2024-03-15

Varroa Supplementary Analysis

This is a repeat of the main analysis but with an offset of 1% of the given response values for a given treatment on a given response target for a given study. This will allow for the inclusion of any ratios with zero's in either the denominator or numerator.

This document outlines the code used for the analysis in O Connell et al 2024 regarding the effects of various treatments on Varroa based on a literature search (see main manuscript for details). The analysis measures the effect of various treatments using the log ratio between the treatment and its associated control. This approach allows us to compared values from various sources and treatments from a wide general perspective.

Load up Packages

We will use the MCMCglmm package which allows us to run mixed effects models in a Bayesian framework.

```
library(MCMCglmm)
```

```
## Loading required package: Matrix
```

```
## Loading required package: coda
```

```
## Loading required package: ape
```

```
library(hdrcde)
```

```
## This is hdrcde 3.4
```

Data

The raw data set Varroa_treatment_database_2023.10.27.csv can be found in the supplementary of the manuscript. Once loaded we will also create a new variable which further splits the category Chemical into "Synthetic" and "Agriculturally_Organic".

```
varroa_data <- read.csv("Varroa_treatment_database_2023.10.27.csv",
                        sep = ",",
                        header = T)

#we will add a breakdown of synthetic versus non synthetic
chem_split <- as.vector(varroa_data$categoryTreatment)

#We will loop around and replace the the term Chemical with its entry in the
#SubCategory1Treatment variable
for(i in 1:length(chem_split)){

  if(chem_split[i] == "Chemical")
    {chem_split[i] <- varroa_data$SubCategory1Treatment[i]}
}

#Set it so Synthetic chemicals are the baseline.
chem_split <- factor(chem_split, levels = c("Synthetic",
                                            "Agriculturally_Organic",
                                            "Biological",
                                            "Physical",
                                            "Mixed"))

#Add this new chem_split variable to the data set.
varroa_data <-  data.frame(varroa_data,
                                     chem_split)
```

log ratio calculations

To calculate the log ratio vales we use a loop so that for every study we calculate the pairwise log ratio between the studies control and each of the treatment measures as log(treatment/control).

There are four different broad measurement types in the analysis. (1) HoneyBeeIncrease: Those that measure aspects of honey bees where an increase in the measure is a measure of the positive effects of the treatment. For example, if the number of bees increases in response to some treatment.

(2) HoneyBeeReduction Those that measure aspects of honey bees where a increase in the measure is a measure of the negative effects of the treatment. For example, if the mortality rate of bees increases in response to some treatment.

(3) VarroaReduction: Those that measure aspects of Varroa where an increase in the measure is a measure of the positive effects of the treatment. For example, if the Varroa mortality rate increases in response to some treatment.

(4) VarroaIncrease: Those that measure aspects of Varroa where an increase in the measure is a measure of the negative effects of the treatment. For example, if the Varroa population size increases in response to some treatment.

In order to include all 4 of these groups together in the main analysis we reversed the sign for the log ratio of HoneyBeeReduction and VarroaIncrease values so that positive values indicate positive outcomes for bee control.

We do not include any infinite ratios caused by either log(1/0) or log(0/1). We change log(0/0) values to zero as while it gives an NA a zero value of no change is comparable to values such as log(1/1).

First we will create a loop for each of the Response Variable Category types (HoneyBeeIncrease, HoneyBeeReduction, VarroaIncrease, VarroaReduction)

HoneyBeeIncrease

Loop matching up all the treatments and controls for measures where an increase is a positive outcome for bees. This loop only compares treatment values within studies and for the same response target (for example, within studies there may be several response targets such as adults, juveniles etc). 3 measures are dropped as the control and treatments have different signs leading to log(-t/c) which cannot be computed.

```
#subset to just responses with HoneyBeeIncrease
HoneyBeeIncrease_data <- varroa_data[varroa_data$ResponseVariableCategory ==
                                       "HoneyBeeIncrease",]

#create a column that has unique treatment and response numbers
HoneyBeeIncrease_data$tre_resp <-  paste(HoneyBeeIncrease_data$Treatment_Group,
                                         "_",
                                         HoneyBeeIncrease_data$ResponseNo.,
                                         sep = "")

#empty list to put the final paired rows into
HoneyBeeIncrease_tret_group_temp <- list()

#Data for every study
for (i in 1:length(unique(HoneyBeeIncrease_data$StudyID))){

  hbi_stud <- HoneyBeeIncrease_data[HoneyBeeIncrease_data$StudyID ==
                                      unique(HoneyBeeIncrease_data$StudyID)[i],]
  #Data for every response target (juvinal. adult etc)
  for(z in 1:length(unique(hbi_stud$ResponseVariableTarget))){

    hbi_stud_res <- hbi_stud[hbi_stud$ResponseVariableTarget ==
                               unique(hbi_stud$ResponseVariableTarget)[z],]

     for(w in 1:length(unique(hbi_stud_res$tre_resp))){

       hbi_tret <- hbi_stud_res[hbi_stud_res$tre_resp ==
                                  unique(hbi_stud_res$tre_resp)[w],]

       #This given the median value across the responses for this study
       median_hbi_tret_res <- median(hbi_tret$specificResponseMean)

#HoneyBeeIncrease_tret_group_temp <- vector()

        for(t in 1:length(hbi_tret[hbi_tret$Status != "control",1])){

#we default to the first control for now.
hbi_control <- hbi_tret[hbi_tret$Status == "control", ][1,]

#rename the colunm names to _control
colnames(hbi_control) <- paste(names(hbi_control),"_control",sep = "")

#we default to the first control for now.
HoneyBeeIncrease_tret_group_temp[[length(HoneyBeeIncrease_tret_group_temp)+1]]<-
  cbind(hbi_tret[hbi_tret$Status != "control",][t,],
                  hbi_control, median_hbi_tret_res)
```

```
        }
    }
  }
  }

#Now we just
HoneyBeeIncrease_paired <- do.call(rbind.data.frame,
                                   HoneyBeeIncrease_tret_group_temp)

#We can add a row of the log ratio of the response mean value
#(specificResponseMean) divided by the control (specificResponseMean_control)

HoneyBeeIncrease_paired$logratio <-
  log(c(HoneyBeeIncrease_paired$specificResponseMean +
          HoneyBeeIncrease_paired$median_hbi_tret_res*0.01)/
        c(HoneyBeeIncrease_paired$specificResponseMean_control +
            HoneyBeeIncrease_paired$median_hbi_tret_res*0.01))
```

```
## Warning in log(c(HoneyBeeIncrease_paired$specificResponseMean +
## HoneyBeeIncrease_paired$median_hbi_tret_res * : NaNs produced
```

```
HoneyBeeIncrease_paired_fin <- HoneyBeeIncrease_paired[!(is.nan(HoneyBeeIncrease_paired$logratio)),]

#Lets just set Chemical as the baseline
HoneyBeeIncrease_paired_fin$categoryTreatment <-
  factor(HoneyBeeIncrease_paired_fin$categoryTreatment,
                                                      levels = c("Chemical",
                                                                 "Physical",
                                                                 "Biological",
                                                                 "Mixed")
                                                      )

#create a variable that gives a unique identify for nested country continent
HoneyBeeIncrease_paired_fin$Cont_Country <-
  paste0(HoneyBeeIncrease_paired_fin$Continent,
          HoneyBeeIncrease_paired_fin$Country)
```

HoneyBeeReduction

Loop matching up all the treatments and controls for measures where a decrease is a positive outcome for bees. This loop only compares treatment values within studies and for the same response target (for example, within studies there may be several response targets such as adults, juveniles etc). After zero adjusting there are 3 NaN values which are caused by the control and treatment having different signs to their values.(59 values where originally dropped)

```
#subset to just responses with HoneyBeeIncrease
HoneyBeeReduction_data <- varroa_data[varroa_data$ResponseVariableCategory ==
                                        "HoneyBeeReduction",]

#create a column that has unique treatment and response numbers
HoneyBeeReduction_data$tre_resp <-  paste(HoneyBeeReduction_data$Treatment_Group,
                                          "_",
```

```
                                        HoneyBeeReduction_data$ResponseNo.,
                                        sep = "")

#empty list to put the final paired rows into
HoneyBeeReduction_tret_group_temp <- list()

#Data for every study
for (i in 1:length(unique(HoneyBeeReduction_data$StudyID))){

  hbr_stud <- HoneyBeeReduction_data[HoneyBeeReduction_data$StudyID ==
                                       unique(HoneyBeeReduction_data$StudyID)[i],]
  #Data for every response target (juvinal. adult etc)
  for(z in 1:length(unique(hbr_stud$ResponseVariableTarget))){

    hbr_stud_res <- hbr_stud[hbr_stud$ResponseVariableTarget ==
                               unique(hbr_stud$ResponseVariableTarget)[z],]

     for(w in 1:length(unique(hbr_stud_res$tre_resp))){

hbr_tret <- hbr_stud_res[hbr_stud_res$tre_resp==unique(hbr_stud_res$tre_resp)[w],]

       #This given the median value across the responses
            median_hbr_tret_res <- median(hbr_tret$specificResponseMean)

        for(t in 1:length(hbr_tret[hbr_tret$Status != "control",1])){


#we default to the first control for now.
hbr_control <- hbr_tret[hbr_tret$Status == "control", ][1,]

#rename the colunm names to _control
colnames(hbr_control) <- paste(names(hbr_control),"_control",sep = "")

#we default to the first control for now.
HoneyBeeReduction_tret_group_temp[[length(HoneyBeeReduction_tret_group_temp)+1]]<-
  cbind(hbr_tret[hbr_tret$Status != "control",][t,],
                  hbr_control, median_hbr_tret_res)

        }
    }
  }
  }

#Now we just
HoneyBeeReduction_paired <- do.call(rbind.data.frame,
                                    HoneyBeeReduction_tret_group_temp)

#We can add a row of the log ratio of the response mean value
#(specificResponseMean) divided by the control (specificResponseMean_control)

HoneyBeeReduction_paired$logratio <-
  log(c(HoneyBeeReduction_paired$specificResponseMean +
          HoneyBeeReduction_paired$median_hbr_tret_res*0.01)/
```

```
        c(HoneyBeeReduction_paired$specificResponseMean_control +
            HoneyBeeReduction_paired$median_hbr_tret_res*0.01))
```

```
## Warning in log(c(HoneyBeeReduction_paired$specificResponseMean +
## HoneyBeeReduction_paired$median_hbr_tret_res * : NaNs produced
```

```
##From looking at the data values for 154,155,156 should have 0 values
#The loop does not work as the zero adjustment is also zero.
HoneyBeeReduction_paired$logratio[154:156] <- c(0)

#This give infinity due to the zero adjustment is also being zero
#So here we will adjust by 1% of the response varible
HoneyBeeReduction_paired$logratio[157] <- log(c(20+ 20*0.01)/c(20*0.01))


HoneyBeeReduction_paired_fin <- HoneyBeeReduction_paired[!(is.nan(HoneyBeeReduction_paired$logratio)),]

#Lets just set Chemical as the baseline
HoneyBeeReduction_paired_fin$categoryTreatment <-
  factor(HoneyBeeReduction_paired_fin$categoryTreatment,
                                                      levels = c("Chemical",
                                                                 "Physical",
                                                                 "Biological",
                                                                 "Mixed")
                                                      )

#create a variable that gives a unique identify for nested country continent
HoneyBeeReduction_paired_fin$Cont_Country <-
  paste0(HoneyBeeReduction_paired_fin$Continent,
         HoneyBeeReduction_paired_fin$Country)


#We can also create a version of the data set
#with the log ratio value flipped so that it can be read as a positive value

HoneyBeeReduction_paired_red_fin <- HoneyBeeReduction_paired_fin

HoneyBeeReduction_paired_red_fin$logratio <-
  -HoneyBeeReduction_paired_red_fin$logratio
```

VarroaIncrease

Loop matching up all the treatments and controls for measures where a decrease is a positive outcome for bees. This loop only compares treatment values within studies and for the same response target (for example, within studies there may be several response targets such as adults, juveniles etc). No values where removed (originally 29 infinite values were removed)

```
#subset to just responses with VarroaIncrease
VarroaIncrease_data <- varroa_data[varroa_data$ResponseVariableCategory ==
                                     "VarroaIncrease",]

#create a column that has unique treatment and response numbers
```

```r
VarroaIncrease_data$tre_resp <-  paste(VarroaIncrease_data$Treatment_Group,
                                       "_",
                                       VarroaIncrease_data$ResponseNo.,
                                       sep = "")

#empty list to put the final paired rows into
VarroaIncrease_tret_group_temp <- list()

#Data for every study
for (i in 1:length(unique(VarroaIncrease_data$StudyID))){

  vi_stud <- VarroaIncrease_data[VarroaIncrease_data$StudyID ==
                                   unique(VarroaIncrease_data$StudyID)[i],]
  #Data for every response target (juvinal. adult etc)
  for(z in 1:length(unique(vi_stud$ResponseVariableTarget))){

    vi_stud_res <- vi_stud[vi_stud$ResponseVariableTarget ==
                             unique(vi_stud$ResponseVariableTarget)[z],]

     for(w in 1:length(unique(vi_stud_res$tre_resp))){

vi_tret <- vi_stud_res[vi_stud_res$tre_resp == unique(vi_stud_res$tre_resp)[w],]


        median_vi_tret_res <- median(vi_tret$specificResponseMean)


        for(t in 1:length(vi_tret[vi_tret$Status != "control",1])){

#we default to the first control for now.
vi_control <- vi_tret[vi_tret$Status == "control", ][1,]

#rename the colunm names to _control
colnames(vi_control) <- paste(names(vi_control),"_control",sep = "")

#we default to the first control for now.
VarroaIncrease_tret_group_temp[[length(VarroaIncrease_tret_group_temp) + 1]]  <-
  cbind(vi_tret[vi_tret$Status != "control",][t,],
                  vi_control, median_vi_tret_res)

        }
    }
  }
  }

#Now we just
VarroaIncrease_paired <- do.call(rbind.data.frame, VarroaIncrease_tret_group_temp)

#We can add a row of the log ratio of the response mean value
#(specificResponseMean) divided by the control (specificResponseMean_control)

VarroaIncrease_paired$logratio <-
  log(c(VarroaIncrease_paired$specificResponseMean +
```

```
          median_vi_tret_res*0.01)/
        c(VarroaIncrease_paired$specificResponseMean_control +
            median_vi_tret_res*0.01))
VarroaIncrease_paired_fin <- VarroaIncrease_paired


#Lets just set Chemical as the baseline
VarroaIncrease_paired_fin$categoryTreatment <-
  factor(VarroaIncrease_paired_fin$categoryTreatment,
                                                    levels = c("Chemical",
                                                               "Physical",
                                                               "Biological",
                                                               "Mixed")
                                                    )

#create a variable that gives a unique identify for nested country continent
VarroaIncrease_paired_fin$Cont_Country <-
  paste0(VarroaIncrease_paired_fin$Continent,
         VarroaIncrease_paired_fin$Country)



#We can also create a version of the dataset
#with the log ratio value flipped so that it can be read as a positive value

VarroaIncrease_paired_red_fin <- VarroaIncrease_paired_fin
VarroaIncrease_paired_red_fin$logratio <- -VarroaIncrease_paired_red_fin$logratio
```

VarroaReduction

Loop matching up all the treatments and controls for measures where an increase is a positive outcome for bees. This loop only compares treatment values within studies and for the same response target (for example, within studies there may be several response targets such as adults, juveniles etc). 1 NaN value was removed due to change in sign between control and main value. Originally 300 infinite values were removed.

```
#subset to just responses with VarroaReduction
VarroaReduction_data <- varroa_data[varroa_data$ResponseVariableCategory ==
                                      "VarroaReduction",]

#create a column that has unique treatment and response numbers
VarroaReduction_data$tre_resp <-  paste(VarroaReduction_data$Treatment_Group,
                                        "_",
                                        VarroaReduction_data$ResponseNo.,
                                        sep = "")

#empty list to put the final paired rows into
VarroaReduction_tret_group_temp <- list()

#Data for every study
for (i in 1:length(unique(VarroaReduction_data$StudyID))){

  vr_stud <- VarroaReduction_data[VarroaReduction_data$StudyID ==
```

```
                                    unique(VarroaReduction_data$StudyID)[i],]
  #Data for every response target (juvinal. adult etc)
  for(z in 1:length(unique(vr_stud$ResponseVariableTarget))){

    vr_stud_res <- vr_stud[vr_stud$ResponseVariableTarget ==
                             unique(vr_stud$ResponseVariableTarget)[z],]

     for(w in 1:length(unique(vr_stud_res$tre_resp))){

vr_tret <- vr_stud_res[vr_stud_res$tre_resp == unique(vr_stud_res$tre_resp)[w],]

           median_vr_tret_res <- median(vr_tret$specificResponseMean)

#HoneyBeeIncrease_tret_group_temp <- vector()

        for(t in 1:length(vr_tret[vr_tret$Status != "control",1])){

#we default to the first control for now.
vr_control <- vr_tret[vr_tret$Status == "control", ][1,]

#rename the colunm names to _control
colnames(vr_control) <- paste(names(vr_control),"_control",sep = "")

#we default to the first control for now.
VarroaReduction_tret_group_temp[[length(VarroaReduction_tret_group_temp) + 1]]<-
  cbind(vr_tret[vr_tret$Status != "control",][t,],
                  vr_control, median_vr_tret_res)

        }
    }
  }
  }

#Now we just
VarroaReduction_paired <- do.call(rbind.data.frame,
                                  VarroaReduction_tret_group_temp)

#We can add a row of the log ratio of the response mean value
#(specificResponseMean) divided by the control (specificResponseMean_control)

VarroaReduction_paired$logratio <-
  log(c(VarroaReduction_paired$specificResponseMean +
          VarroaReduction_paired$median_vr_tret_res*0.01)/
        c(VarroaReduction_paired$specificResponseMean_control +
          VarroaReduction_paired$median_vr_tret_res*0.01))
```

```
## Warning in log(c(VarroaReduction_paired$specificResponseMean +
## VarroaReduction_paired$median_vr_tret_res * : NaNs produced
```

```
#From looking at the data the values for 731,732,733,735,736,737,738
#should be zero as all values are zero
VarroaReduction_paired$logratio[c(731,732,733,735,736,737,738)] <- c(0)
```

```
VarroaReduction_paired$logratio[727] <- log(c(VarroaReduction_paired$specificResponseMean[727] +
                                        VarroaReduction_paired$specificResponseMean[727]*0.01)/
                                      c(VarroaReduction_paired$specificResponseMean[727]*0.01))

VarroaReduction_paired$logratio[728] <- log(c(VarroaReduction_paired$specificResponseMean[728] +
                                        VarroaReduction_paired$specificResponseMean[728]*0.01)/
                                      c(VarroaReduction_paired$specificResponseMean[728]*0.01))

VarroaReduction_paired$logratio[729] <- log(c(VarroaReduction_paired$specificResponseMean[729] +
                                        VarroaReduction_paired$specificResponseMean[729]*0.01)/
                                      c(VarroaReduction_paired$specificResponseMean[729]*0.01))

VarroaReduction_paired$logratio[730] <- log(c(VarroaReduction_paired$specificResponseMean[730] +
                                        VarroaReduction_paired$specificResponseMean[730]*0.01)/
                                      c(VarroaReduction_paired$specificResponseMean[730]*0.01))

VarroaReduction_paired$logratio[734] <- log(c(VarroaReduction_paired$specificResponseMean[734] +
                                        VarroaReduction_paired$specificResponseMean[734]*0.01)/
                                      c(VarroaReduction_paired$specificResponseMean[734]*0.01))


VarroaReduction_paired_fin <-
  VarroaReduction_paired[!(is.nan(VarroaReduction_paired$logratio)),]

#lets remove any pairs that are infinite.
VarroaReduction_paired_fin <-
  VarroaReduction_paired[!(is.infinite(VarroaReduction_paired$logratio)),]


#Lets just set Chemical as the baseline
VarroaReduction_paired_fin$categoryTreatment <-
  factor(VarroaReduction_paired_fin$categoryTreatment,
                                                    levels = c("Chemical",
                                                               "Physical",
                                                               "Biological",
                                                               "Mixed"
                                                               )
                                                    )

#create a variable that gives a unique identify for nested country continent
VarroaReduction_paired_fin$Cont_Country <-
  paste0(VarroaReduction_paired_fin$Continent,
         VarroaReduction_paired_fin$Country)
```

We can join all the studies together with the sign reversed so that a positive difference indicates postie outcomes for control

```
#This just puts all the datasets together.

#We need to rename the median response column to match

names(HoneyBeeIncrease_paired_fin)[names(HoneyBeeIncrease_paired_fin) == "median_hbi_tret_res"] <- "medi
names(HoneyBeeReduction_paired_red_fin)[names(HoneyBeeReduction_paired_red_fin) == "median_hbr_tret_res
```

```
names(VarroaIncrease_paired_red_fin)[names(VarroaIncrease_paired_red_fin) == "median_vi_tret_res"] <- "
names(VarroaReduction_paired_fin)[names(VarroaReduction_paired_fin) == "median_vr_tret_res"] <- "median_


#Notice HoneyBeeReduction_paired_red_fin and VarroaIncrease_paired_red_fin
#have their signs reversed
Full_comb_data <- rbind(HoneyBeeIncrease_paired_fin,
                            HoneyBeeReduction_paired_red_fin,
                            VarroaIncrease_paired_red_fin,
                            VarroaReduction_paired_fin)

#just bee data
Full_bees_data <- rbind(HoneyBeeIncrease_paired_fin,
                            HoneyBeeReduction_paired_red_fin)

#Reset the levels for bees so workers are the baseline
Full_bees_data$ResponseVariableTarget <-
  factor(Full_bees_data$ResponseVariableTarget,
                                                levels = c("Honey_bee_worker",
                                                           "Honey_bee_colony",
                                                           "Honey_bee_juvenile",
                                                           "Honey_bee_product",
                                                           "Honey_bee_queen"))

#just vorroa data
Full_varroa_data <- rbind(VarroaIncrease_paired_red_fin,
                            VarroaReduction_paired_fin)
```

MCMCglmm analysis

Prior and parameters

Now that we have a set of log ratios we can run some analysis. We first set up a non-informative prior for our our models, with a flat gamma distribution used as the non-informative prior for each for the random terms. For more info on priors see the Course notes (http://cran.nexr.com/web/packages/MCMCglmm/vignettes/CourseNotes.pdf).

```
prior_d <-list(R = list(V = 1, nu=0.002),
            G = list(G1=list(V = 1, nu=0.002, alpha.mu= 0, alpha.V= 10^3),
                     G2=list(V = 1, nu=0.002, alpha.mu= 0, alpha.V= 10^3),
                     G3=list(V = 1, nu=0.002, alpha.mu= 0, alpha.V= 10^3)
                     ))
```

We will also set the number of iterations (nitt), the burnin (burnin) and the thining (thining).

```
burnin <- c(10000)
nitt <- c(110000)
thining <- c(50)
```

Main model

The first model will include all studies with a positive values associated with a positive outcome for bee health.

We run three chains (mod_full, mod_full2 and mod_full3) so we can test if they converge.

```
mod_full <- MCMCglmm(logratio ~ chem_split
                                + Context,
                                 rcov=~units,
                                 random =~StudyID_control
                                          + Continent
                                          + Cont_Country,
                                 family ="gaussian",
                                 data = Full_comb_data,
                                 nitt = nitt,
                                 thin = thining,
                                 burnin = burnin,
                                 prior = prior_d,
                                 verbose = F
                             )

#second model acts as second chain for convergence
#Add a 3rd later for final check
mod_full2 <- MCMCglmm(logratio ~ chem_split
                                  + Context,
                                 rcov=~units,
                                 random =~StudyID_control
                                          + Continent
                                          + Cont_Country,
                                 family ="gaussian",
                                 data = Full_comb_data,
                                 nitt = nitt,
                                 thin = thining,
                                 burnin = burnin,
                                 prior = prior_d,
                                 verbose = F
                             )

mod_full3 <- MCMCglmm(logratio ~ chem_split
                                  + Context,
                                 rcov=~units,
                                 random =~StudyID_control
                                          + Continent
                                          + Cont_Country,
                                 family ="gaussian",
                                 data = Full_comb_data,
                                 nitt = nitt,
                                 thin = thining,
                                 burnin = burnin,
                                 prior = prior_d,
                                 verbose = F
                             )
```

```r
summary(mod_full)
```

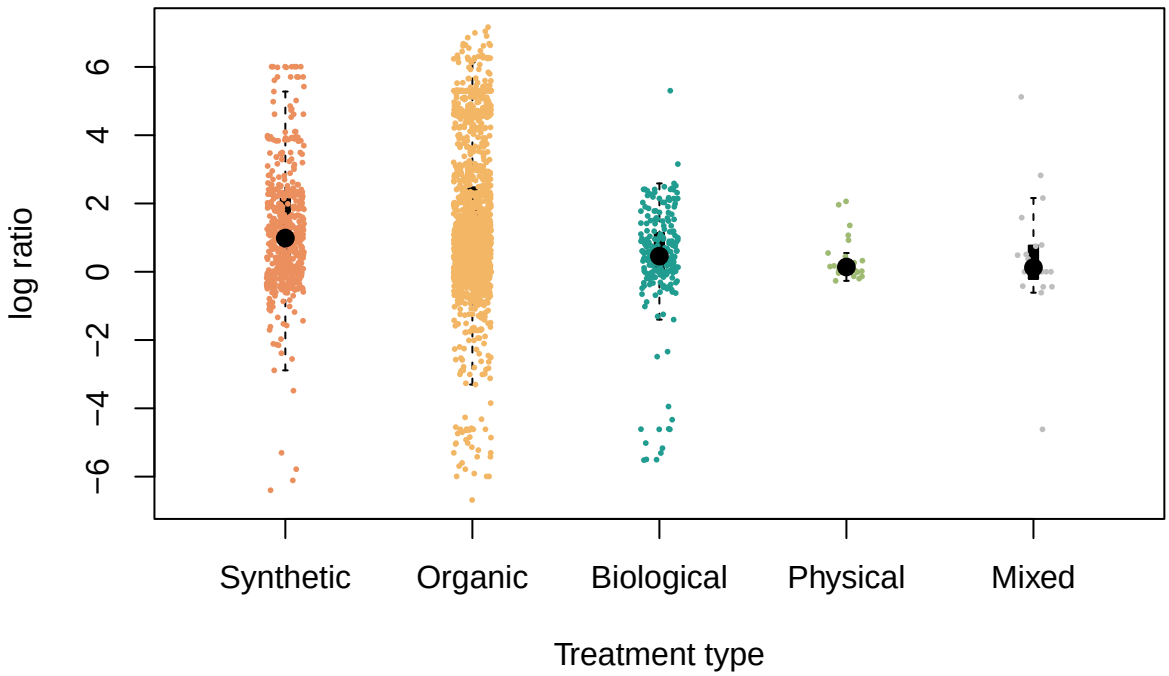
```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
##
##  DIC: 9490.116
##
##  G-structure:  ~StudyID_control
##
##                 post.mean l-95% CI u-95% CI eff.samp
## StudyID_control     1.076   0.7735    1.401     2000
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent    0.2971 1.181e-08    1.087     1375
##
##                ~Cont_Country
##
##              post.mean  l-95% CI u-95% CI eff.samp
## Cont_Country   0.04811 1.722e-11   0.1863     2000
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units     2.634    2.481    2.783     1833
##
##  Location effects: logratio ~ chem_split + Context
##
##                                  post.mean l-95% CI u-95% CI eff.samp  pMCMC
## (Intercept)                        0.97159  0.43684  1.48823     2000  0.008
## chem_splitAgriculturally_Organic   0.02702 -0.19105  0.29697     2000  0.830
## chem_splitBiological              -0.56827 -1.05209 -0.04856     2000  0.029
## chem_splitPhysical                -0.51822 -1.54262  0.55656     2000  0.327
## chem_splitMixed                   -0.30214 -1.75041  1.04906     1786  0.687
## ContextLab                         0.66577  0.32196  0.98817     2000 <5e-04
##
## (Intercept)                       **
## chem_splitAgriculturally_Organic
## chem_splitBiological              *
## chem_splitPhysical
## chem_splitMixed
## ContextLab                        ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We can check for convergence using gelman.diag(). Values

We can do a simple plot of our results. I need to fix this graph up some more.

```r
plot(Full_comb_data$logratio ~ Full_comb_data$chem_split,
     col ="black",
     bty = "n",
     boxwex = 0.05,
     ylab = "log ratio",
     xlab = "Treatment type",
     pch = 16,
     cex = 0,
     names = c("Synthetic", "Organic", "Biological", "Physical", "Mixed"))

#We can plot the points for Synthetic
points(Full_comb_data[Full_comb_data$chem_split == "Synthetic", "logratio"] ~ jitter(as.integer(Full_co
                    amount = 0.1),
       pch = 16,
       col = rgb(236, 143, 94, max=255),
       cex = 0.4)

points(median(Full_comb_data[Full_comb_data$chem_split == "Synthetic", "logratio"])
       ~ c(1),
       pch = 16,
       col = "black",
       cex = 1.3)


#We can plot the points for Agriculturally_Organic
points(Full_comb_data[Full_comb_data$chem_split == "Agriculturally_Organic",
                      "logratio"] ~
         jitter(as.integer(Full_comb_data[Full_comb_data$chem_split ==
                                            "Agriculturally_Organic",
       amount = 0.1),
       pch = 16,
       col = rgb(243, 182, 100, max=255),
       cex = 0.4)

points(median(Full_comb_data[Full_comb_data$chem_split == "Agriculturally_Organic",
                             "logratio"]) ~ c(2),
       pch = 16,
       col = "black",
       cex = 1.3)

#We can plot the points for Biological
points(Full_comb_data[Full_comb_data$chem_split == "Biological", "logratio"] ~ jitter(as.integer(Full_c
                    amount = 0.1),
       pch = 16,
       col = rgb(33, 156, 144, max=255),
       cex = 0.4)

points(median(Full_comb_data[Full_comb_data$chem_split == "Biological", "logratio"])
       ~ c(3),
       pch = 16,
       col = "black",
       cex = 1.3)
```

```
#We can plot the points for Physical
points(Full_comb_data[Full_comb_data$chem_split == "Physical", "logratio"] ~ jitter(as.integer(Full_comb
                    amount = 0.1),
       pch = 16,
       col = rgb(159, 187, 115, max=255),
       cex = 0.4)


points(median(Full_comb_data[Full_comb_data$chem_split == "Physical", "logratio"])
       ~ c(4),
       pch = 16,
       col = "black",
       cex = 1.3)


#We can plot the points for Mixed
points(Full_comb_data[Full_comb_data$chem_split == "Mixed", "logratio"] ~ jitter(as.integer(Full_comb_da
                    amount = 0.1),
       pch = 16,
       col = "grey",
       cex = 0.4)

points(median(Full_comb_data[Full_comb_data$chem_split == "Mixed", "logratio"])
       ~ c(5),
       pch = 16,
       col = "black",
       cex = 1.3)
```



Overall Synthetic chemicals have a significant overall positive outcome at a ratio of 2.7/1 when compared to the treatment compared to control. This is significantly higher when compared to biological controls which only have a positive outcome effect at a ratio of 1.3/1 when compared to the control.

There is some weak support that organic chemicals have less of an effect compared to synthetic chemicals with a ratio of 2.4/1, however this is not significantly different.

Both Physical and Mixed treatments are found to have reduced effects on outcomes when compared to synthetic chemicals, however neither are significantly different.

Finally, there is weak support that treatments have higher positive outcomes when tested in lab based setting, however, this is also not significant.

Main model with effects just on bees

We can also repeat the main model with just bees included. This is with both increases and decreases combined but with the decreases pop data sign flipped so now any positive value is also a posative indicator for bee population or health.

```
All_bees_data <- rbind(HoneyBeeReduction_paired_red_fin,
                       HoneyBeeIncrease_paired_fin)

mod_all_bees <- MCMCglmm(logratio ~ chem_split,
                                  rcov=~units,
                                  random =~StudyID_control
                                           + Continent
                                           + Cont_Country,
                                  family ="gaussian",
                                  data = All_bees_data,
                                  nitt = nitt,
                                  thin = thining,
                                  burnin = burnin,
                                  prior = prior_d,
                                  verbose = FALSE
                           )



mod_all_bees2 <- MCMCglmm(logratio ~ chem_split,
                                  rcov=~units,
                                  random =~StudyID_control
                                           + Continent
                                           + Cont_Country,
                                  family ="gaussian",
                                  data = All_bees_data,
                                  nitt = nitt,
                                  thin = thining,
                                  burnin = burnin,
                                  prior = prior_d,
                                  verbose = FALSE
                           )


mod_all_bees3 <- MCMCglmm(logratio ~ chem_split,
                                  rcov=~units,
                                  random =~StudyID_control
                                           + Continent
```

```
                                        + Cont_Country,
                                    family ="gaussian",
                                    data = All_bees_data,
                                    nitt = nitt,
                                    thin = thining,
                                    burnin = burnin,
                                    prior = prior_d,
                                    verbose = FALSE
                                )

#Check the fixed terms
mod_all_bees_Sol_conv <- gelman.diag(mcmc.list(mod_all_bees$Sol,
                                               mod_all_bees2$Sol,
                                               mod_all_bees3$Sol))
mod_all_bees_Sol_conv
```

```
## Potential scale reduction factors:
##
##                                   Point est. Upper C.I.
## (Intercept)                                1          1
## chem_splitAgriculturally_Organic           1          1
## chem_splitBiological                       1          1
## chem_splitPhysical                         1          1
## chem_splitMixed                            1          1
##
## Multivariate psrf
##
## 1
```

```
#Check the random terms
mod_all_bees_VCV_conv <- gelman.diag(mcmc.list(mod_all_bees$VCV,
                                               mod_all_bees2$VCV,
                                               mod_all_bees3$VCV))
mod_all_bees_VCV_conv
```

```
## Potential scale reduction factors:
##
##                  Point est. Upper C.I.
## StudyID_control           1          1
## Continent                 1          1
## Cont_Country              1          1
## units                     1          1
##
## Multivariate psrf
##
## 1
```

```
summary(mod_all_bees)
```

```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
```

```
##  Sample size  = 2000
##
##  DIC: 2505.205
##
##  G-structure:  ~StudyID_control
##
##                 post.mean l-95% CI u-95% CI eff.samp
## StudyID_control     1.272   0.8559    1.807     2000
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent     0.168 5.446e-08   0.6254     2000
##
##                ~Cont_Country
##
##              post.mean  l-95% CI u-95% CI eff.samp
## Cont_Country   0.06762 1.043e-08   0.2691     1740
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units     1.343    1.207    1.481     2000
##
##  Location effects: logratio ~ chem_split
##
##                                  post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)                       -0.64719 -1.15793 -0.04305     2000 0.035 *
## chem_splitAgriculturally_Organic   0.28173 -0.04389  0.57278     2000 0.077 .
## chem_splitBiological               0.44197 -0.26932  1.10433     2156 0.209
## chem_splitPhysical                 0.59960 -0.62008  2.07924     2000 0.382
## chem_splitMixed                    0.67818 -1.19502  2.46573     2000 0.461
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

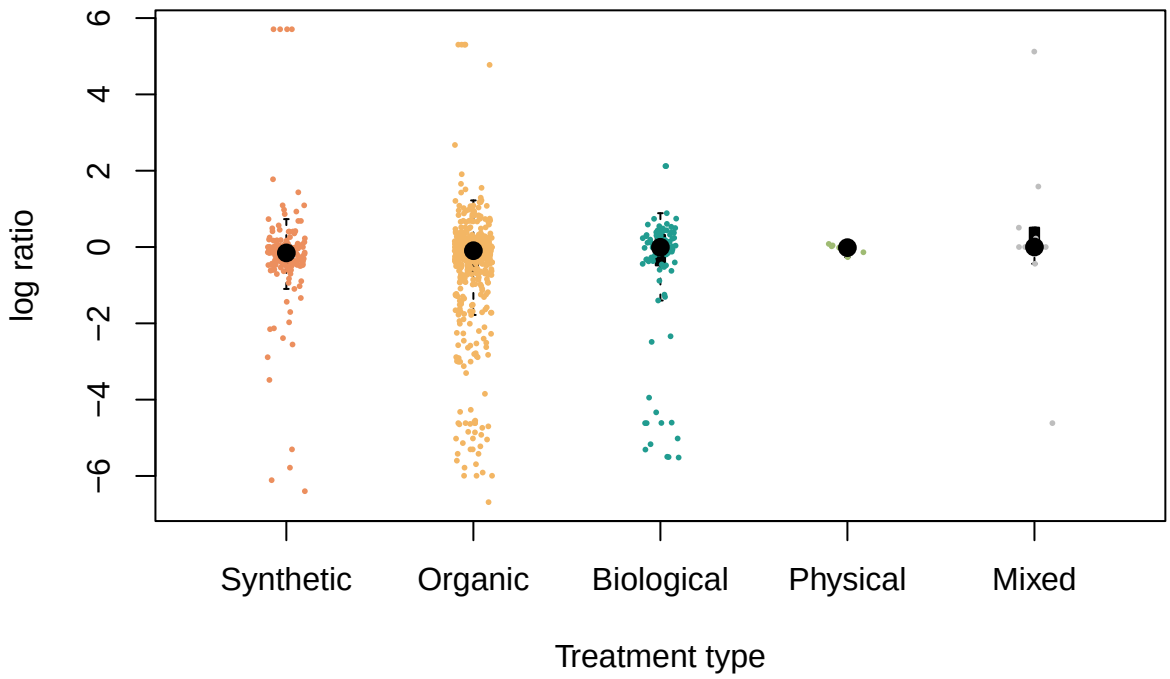
Synthetic chemical are not found to significantly effect bee pops when compared to the Null, but there is some weak evidence they have perform worse than the control at a ratio 0.74/1. Biological treatments have a significant positive effect on bee pops at a ratio of 1.5/1.

We can do a simple plot of our results.

```
plot(All_bees_data$logratio ~ All_bees_data$chem_split,
     col ="black",
     bty = "n",
     boxwex = 0.05,
     ylab = "log ratio",
     xlab = "Treatment type",
     pch = 16,
     cex = 0,
     names = c("Synthetic", "Organic", "Biological", "Physical", "Mixed"))

#We can plot the points for Synthetic
points(All_bees_data[All_bees_data$chem_split == "Synthetic", "logratio"] ~ jitter(as.integer(All_bees_
                  amount = 0.1),
```

```
       pch = 16,
       col = rgb(236, 143, 94, max=255),
       cex = 0.4)

points(median(All_bees_data[All_bees_data$chem_split == "Synthetic", "logratio"])
       ~ c(1),
       pch = 16,
       col = "black",
       cex = 1.3)


#We can plot the points for Agriculturally_Organic
points(All_bees_data[All_bees_data$chem_split == "Agriculturally_Organic", "logratio"] ~ jitter(as.inte
                  amount = 0.1),
       pch = 16,
       col = rgb(243, 182, 100, max=255),
       cex = 0.4)


points(median(All_bees_data[All_bees_data$chem_split == "Agriculturally_Organic",
                            "logratio"])
       ~ c(2),
       pch = 16,
       col = "black",
       cex = 1.3)



#We can plot the points for Biological
points(All_bees_data[All_bees_data$chem_split == "Biological", "logratio"] ~ jitter(as.integer(All_bees
                  amount = 0.1),
       pch = 16,
       col = rgb(33, 156, 144, max=255),
       cex = 0.4)

points(median(All_bees_data[All_bees_data$chem_split == "Biological",
                            "logratio"])
       ~ c(3),
       pch = 16,
       col = "black",
       cex = 1.3)



#We can plot the points for Physical
points(All_bees_data[All_bees_data$chem_split == "Physical", "logratio"] ~ jitter(as.integer(All_bees_da
                  amount = 0.1),
       pch = 16,
       col = rgb(159, 187, 115, max=255),
       cex = 0.4)


points(median(All_bees_data[All_bees_data$chem_split == "Physical",
```

```
                              "logratio"])
       ~ c(4),
       pch = 16,
       col = "black",
       cex = 1.3)



#We can plot the points for Mixed
points(All_bees_data[All_bees_data$chem_split == "Mixed", "logratio"] ~ jitter(as.integer(All_bees_data
                                                        "chem_split"]),
                                                         amount = 0.1),
                                                         pch = 16,
                                                         col = "grey",
                                                         cex = 0.4)

points(median(All_bees_data[All_bees_data$chem_split == "Mixed",
                              "logratio"])
       ~ c(5),
       pch = 16,
       col = "black",
       cex = 1.3)
```



Main model with effects just on Varrora

We can also do the main model with just Varroa data included. Here we include both studies that measure increases or decreases in Varroa population or health data. Again the log ratio is calculated so that a positive value indicates a positive outcome for bee health/populations (that is a reduction in Varroa is indicated in positive terms.).

```r
All_varroa_data <- rbind(VarroaReduction_paired_fin,
                         VarroaIncrease_paired_red_fin)

All_varroa_data_mod <- MCMCglmm(logratio ~ chem_split,
                                rcov=~units,
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = All_varroa_data,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )


All_varroa_data_mod2 <- MCMCglmm(logratio ~ chem_split,
                                rcov=~units,
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = All_varroa_data,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )

All_varroa_data_mod3 <- MCMCglmm(logratio ~ chem_split,
                                rcov=~units,
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = All_varroa_data,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )


#Check the fixed terms
All_varroa_data_mod_Sol_conv <- gelman.diag(mcmc.list(All_varroa_data_mod$Sol,
                                                      All_varroa_data_mod2$Sol,
                                                      All_varroa_data_mod3$Sol))
All_varroa_data_mod_Sol_conv
```

```
## Potential scale reduction factors:
##
##                                   Point est. Upper C.I.
## (Intercept)                                1       1.00
## chem_splitAgriculturally_Organic           1       1.00
## chem_splitBiological                       1       1.01
## chem_splitPhysical                         1       1.00
## chem_splitMixed                            1       1.00
##
## Multivariate psrf
##
## 1
```

```
#Check the random terms
All_varroa_data_mod_VCV_conv <- gelman.diag(mcmc.list(All_varroa_data_mod$VCV,
                                                      All_varroa_data_mod2$VCV,
                                                      All_varroa_data_mod3$VCV))
All_varroa_data_mod_VCV_conv
```

```
## Potential scale reduction factors:
##
##                  Point est. Upper C.I.
## StudyID_control        1.00       1.00
## Continent              1.11       1.11
## Cont_Country           1.00       1.00
## units                  1.00       1.00
##
## Multivariate psrf
##
## 1
```

```
summary(All_varroa_data_mod)
```

```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
##
##  DIC: 5747.984
##
##  G-structure:  ~StudyID_control
##
##                  post.mean l-95% CI u-95% CI eff.samp
## StudyID_control     1.102   0.7946    1.452     2000
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent    0.1388 4.336e-10   0.5216     2000
##
##                ~Cont_Country
##
##              post.mean l-95% CI u-95% CI eff.samp
```

```
## Cont_Country   0.05789 1.68e-08   0.2225     1593
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units     1.667    1.556    1.787     2000
##
##  Location effects: logratio ~ chem_split
##
##                                  post.mean l-95% CI u-95% CI eff.samp  pMCMC
## (Intercept)                         1.7636   1.3451   2.1852     2126  0.002
## chem_splitAgriculturally_Organic    0.0223  -0.2500   0.2568     2000  0.858
## chem_splitBiological               -0.7927  -1.2685  -0.2986     2000 <5e-04
## chem_splitPhysical                 -0.8956  -1.9788   0.1099     1865  0.086
## chem_splitMixed                    -0.6442  -2.0522   0.7053     2000  0.370
##
## (Intercept)                        **
## chem_splitAgriculturally_Organic
## chem_splitBiological               ***
## chem_splitPhysical                 .
## chem_splitMixed
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

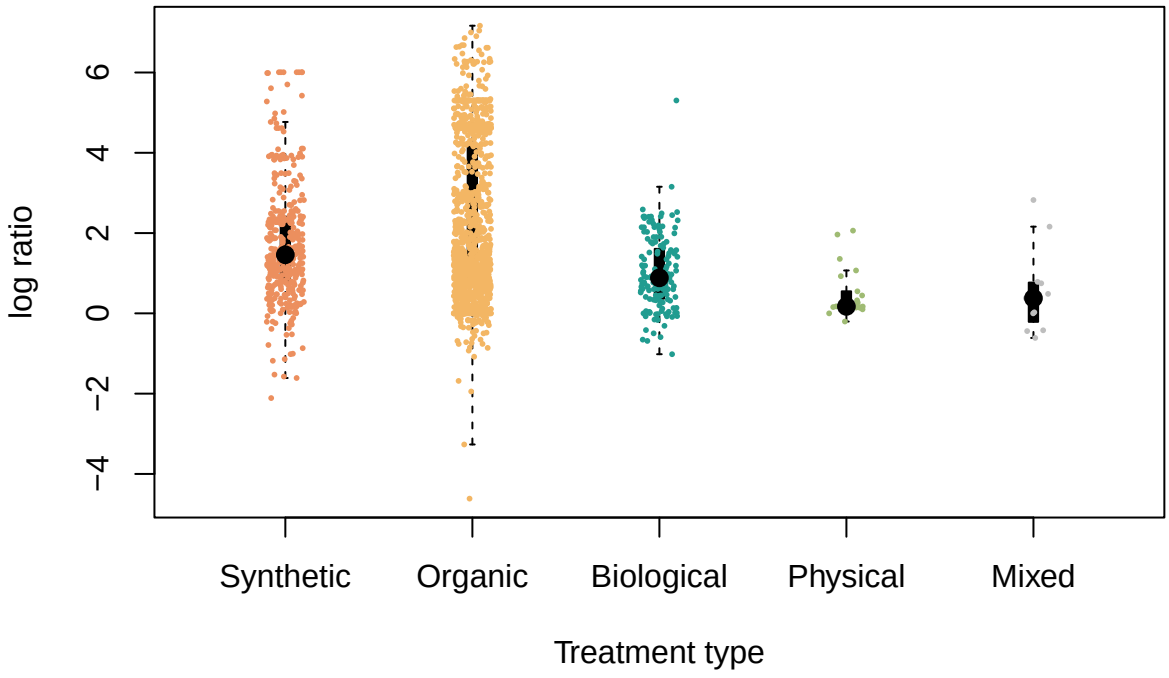
In the main model using just Varroa, synthetic chemicals are 4.7 times better when compared to the control in terms of treatment (B = 1.55, lower 95% CI = 1.21, higher 95% CI = 1.92; Table 3). While Agriculturally Organic have a slightly lower effect size this was not significantly lower (Table 3). Both biological and physical treatments are significantly less effective compared to synthetic chemicals, with biological treatments having a positive effect of 1.7 times that of their controls (B = -1.02, lower 95% CI = 1.33, higher 95% CI = 0.68; Table 3) and physical treatments having a positive effect of 2.1 times that of their controls (B = -1.02, lower 95% CI = 1.33, higher 95% CI = 0.68; Table 3).

We can do a simple plot of our results.

```
plot(All_varroa_data$logratio ~ All_varroa_data$chem_split,
     col ="black",
     bty = "n",
     boxwex = 0.05,
     ylab = "log ratio",
     xlab = "Treatment type",
     pch = 16,
     cex = 0,
    names = c("Synthetic", "Organic", "Biological", "Physical", "Mixed"))

#We can plot the points for Synthetic
points(All_varroa_data[All_varroa_data$chem_split == "Synthetic", "logratio"] ~ jitter(as.integer(All_va
                   amount = 0.1),
       pch = 16,
       col = rgb(236, 143, 94, max=255),
       cex = 0.4)

points(median(All_varroa_data[All_varroa_data$chem_split == "Synthetic",
                              "logratio"])
```

```
       ~ c(1),
       pch = 16,
       col = "black",
       cex = 1.3)


#We can plot the points for Agriculturally_Organic
points(All_varroa_data[All_varroa_data$chem_split == "Agriculturally_Organic",
                       "logratio"]
       ~ jitter(as.integer(All_varroa_data[All_varroa_data$chem_split ==
                                             "Agriculturally_Organic",
                                           "chem_split"]),
                                            amount = 0.1),
                                            pch = 16,
                                            col = rgb(243, 182, 100, max=255),
                                            cex = 0.4)

points(median(All_varroa_data[All_varroa_data$chem_split == "Agriculturally_Organic",
                              "logratio"])
       ~ c(2),
       pch = 16,
       col = "black",
       cex = 1.3)



#We can plot the points for Biological
points(All_varroa_data[All_varroa_data$chem_split == "Biological", "logratio"] ~ jitter(as.integer(All_
                  amount = 0.1),
       pch = 16,
       col = rgb(33, 156, 144, max=255),
       cex = 0.4)

points(median(All_varroa_data[All_varroa_data$chem_split == "Biological",
                              "logratio"])
       ~ c(3),
       pch = 16,
       col = "black",
       cex = 1.3)


#We can plot the points for Physical
points(All_varroa_data[All_varroa_data$chem_split == "Physical", "logratio"] ~ jitter(as.integer(All_va
                  amount = 0.1),
       pch = 16,
       col = rgb(159, 187, 115, max=255),
       cex = 0.4)

points(median(All_varroa_data[All_varroa_data$chem_split == "Physical",
                              "logratio"])
       ~ c(4),
       pch = 16,
       col = "black",
```

```
       cex = 1.3)



#We can plot the points for Mixed
points(All_varroa_data[All_varroa_data$chem_split == "Mixed", "logratio"] ~ jitter(as.integer(All_varroa
                                                                                    "chem_split"]),
                                                                                     amount = 0.1),
                                                                                     pch = 16,
                                                                                     col = "grey",
                                                                                     cex = 0.4)

points(median(All_varroa_data[All_varroa_data$chem_split == "Mixed",
                              "logratio"])
       ~ c(5),
       pch = 16,
       col = "black",
       cex = 1.3)
```



Chemical treatment sub analysis

Dosage dependance

Taking from the Full_comb_data, which has the signs flipped so any positive number is a positive effect for bees (i.e. decreased bee mortality is now a positive number) we create a subset for data that has some measure of dosage. To allow for comparisons we include dosage as low medium high as either described in the paper.

```r
Full_dosage <- data.frame(logratio = Full_comb_data$logratio,
                          StudyID = Full_comb_data$StudyID,
                          SubCa2Treat = Full_comb_data$SubCategory2Treatment,
                          Dosage_level = factor(Full_comb_data$Dosage_level),
                          Continent = Full_comb_data$Continent,
                          Country = Full_comb_data$Country,
                          Cont_Country = Full_comb_data$Cont_Country)

Full_dosage <- na.omit(Full_dosage)
```

Ordinal dosage analysis.

```r
mod_dos <- MCMCglmm(logratio ~ Dosage_level,
                                rcov=~units,
                                random =~StudyID,
                                family ="gaussian",
                                data = Full_dosage,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                verbose = FALSE
                            )

mod_dos2 <- MCMCglmm(logratio ~ Dosage_level,
                                rcov=~units,
                                random =~StudyID ,
                                family ="gaussian",
                                data = Full_dosage,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                verbose = FALSE
                            )

mod_dos3 <- MCMCglmm(logratio ~ Dosage_level,
                                rcov=~units,
                                random =~StudyID ,
                                family ="gaussian",
                                data = Full_dosage,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                verbose = FALSE
                            )


#Check the fixed terms
mod_dos_Sol_conv <- gelman.diag(mcmc.list(mod_dos$Sol,
                                          mod_dos2$Sol,
                                          mod_dos3$Sol))
mod_dos_Sol_conv
```

```
## Potential scale reduction factors:
```

```
##
##                     Point est. Upper C.I.
## (Intercept)                  1          1
## Dosage_levelLOW              1          1
## Dosage_levelMEDIUM           1          1
##
## Multivariate psrf
##
## 1
```

```
#Check the random terms
mod_dos_VCV_conv <- gelman.diag(mcmc.list(mod_dos$VCV,
                                          mod_dos2$VCV,
                                          mod_dos3$VCV))
mod_dos_VCV_conv
```

```
## Potential scale reduction factors:
##
##         Point est. Upper C.I.
## StudyID          1       1.00
## units            1       1.01
##
## Multivariate psrf
##
## 1
```

```
summary(mod_dos)
```

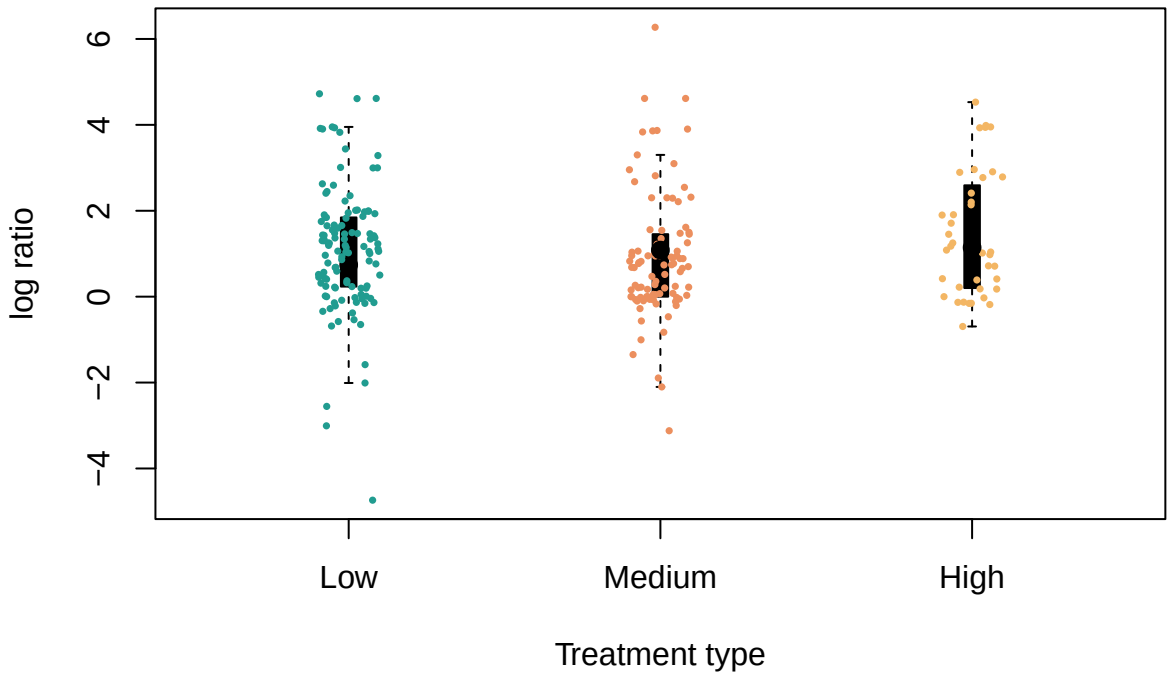
```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
##
##  DIC: 853.4952
##
##  G-structure:  ~StudyID
##
##         post.mean l-95% CI u-95% CI eff.samp
## StudyID    0.5901   0.1602    1.098     2000
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units     1.653    1.375    1.989     2132
##
##  Location effects: logratio ~ Dosage_level
##
##                     post.mean l-95% CI u-95% CI eff.samp  pMCMC
## (Intercept)            1.0844   0.6842   1.4616     2445 <5e-04 ***
## Dosage_levelLOW       -0.2092  -0.5131   0.1736     2000  0.230
## Dosage_levelMEDIUM     0.1855  -0.3142   0.6838     2000  0.468
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

When comparing dosages there is some support that the lowest dosages (2.1/1) are less effective compared to the highest dosages (2.6/1) with no significant difference between the highest and medium levels of dosages. Note that there are only 24 studies in this analysis so it cannot really be broken down more.

We can do a simple plot of our results.

```
plot(Full_dosage$logratio ~ factor(Full_dosage$Dosage_level),
     col ="black",
     bty = "n",
     boxwex = 0.05,
     ylab = "log ratio",
     xlab = "Treatment type",
     pch = 16,
     cex = 0,
     names = c("Low", "Medium", "High"))

#We can plot the points for LOW
points(Full_dosage[Full_dosage$Dosage_level == "LOW", "logratio"] ~
         jitter(as.integer(Full_dosage[Full_dosage$Dosage_level == "LOW",
                                       "Dosage_level"]),
                amount = 0.1),
       pch = 16,
       col = rgb(236, 143, 94, max=255),
       cex = 0.5)

points(median(Full_dosage[Full_dosage$Dosage_level == "LOW", "logratio"])
       ~ c(1),
       pch = 16,
       col = "black",
       cex = 1.3)


#We can plot the points for MEDIUM
points(Full_dosage[Full_dosage$Dosage_level == "MEDIUM", "logratio"] ~
         jitter(as.integer(Full_dosage[Full_dosage$Dosage_level == "MEDIUM",
                                       "Dosage_level"]),
       amount = 0.1),
       pch = 16,
       col = rgb(243, 182, 100, max=255),
       cex = 0.5)

points(median(Full_dosage[Full_dosage$Dosage_level == "MEDIUM", "logratio"])
       ~ c(2),
       pch = 16,
       col = "black",
       cex = 1.3)

#We can plot the points for HIGH
points(Full_dosage[Full_dosage$Dosage_level == "HIGH", "logratio"] ~
         jitter(as.integer(Full_dosage[Full_dosage$Dosage_level == "HIGH",
                  amount = 0.1),
       pch = 16,
       col = rgb(33, 156, 144, max=255),
       cex = 0.5)
```

```
points(median(Full_dosage[Full_dosage$Dosage_level == "HIGH", "logratio"])
       ~ c(3),
       pch = 16,
       col = "black",
       cex = 1.3)
```



break down each of the chemicals

We can compare each of the specific chemicals for which we had enough data.

```
temp_Sub_chem <- Full_comb_data

#combine Flumethrin and Fluvalinate and call them Pyrethroid
temp_Sub_chem[temp_Sub_chem$broadTreatment %in% c("Flumethrin",
                                                  "Fluvalinate"),
              "broadTreatment"] <- "Pyrethroid"

Sub_chem <- temp_Sub_chem[temp_Sub_chem$broadTreatment %in%
c("Amitraz",
  "Coumaphos",
  "Thymol",
  "Oxalic_acid",
  "Formic_acid",
  "Pyrethroid"), ]

Sub_chem$broadTreatment <- factor(Sub_chem$broadTreatment)
```

We can now run the model comparing each of the size chemical groups to a baseline. We use Amitraz as the baseline here as its got a large sample size and as its one of the most effective trreatments making the contrasts a little easier to interpreate.

```
mod_spec_chem <- MCMCglmm(logratio ~ broadTreatment,
                                rcov=~units,
                                random =~StudyID
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = Sub_chem,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )

mod_spec_chem2 <- MCMCglmm(logratio ~ broadTreatment,
                                rcov=~units,
                                random =~StudyID
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = Sub_chem,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )


mod_spec_chem3 <- MCMCglmm(logratio ~ broadTreatment,
                                rcov=~units,
                                random =~StudyID
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = Sub_chem,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )



#Check the fixed terms
mod_spec_chem_Sol_conv <- gelman.diag(mcmc.list(mod_spec_chem$Sol,
                                                mod_spec_chem2$Sol,
                                                mod_spec_chem3$Sol))
mod_spec_chem_Sol_conv
```

```
## Potential scale reduction factors:
##
```

```
##                            Point est. Upper C.I.
## (Intercept)                         1       1.00
## broadTreatmentCoumaphos             1       1.00
## broadTreatmentFormic_acid           1       1.00
## broadTreatmentOxalic_acid           1       1.00
## broadTreatmentPyrethroid            1       1.00
## broadTreatmentThymol                1       1.01
##
## Multivariate psrf
##
## 1
```

```
#Check the random terms
mod_spec_chem_VCV_conv <- gelman.diag(mcmc.list(mod_spec_chem$VCV,
                                                mod_spec_chem2$VCV,
                                                mod_spec_chem3$VCV))
mod_spec_chem_VCV_conv
```

```
## Potential scale reduction factors:
##
##              Point est. Upper C.I.
## StudyID            1.00       1.01
## Continent          1.05       1.06
## Cont_Country       1.00       1.00
## units              1.00       1.01
##
## Multivariate psrf
##
## 1
```

```
summary(mod_spec_chem)
```

```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
##
##  DIC: 3510.851
##
##  G-structure:  ~StudyID
##
##         post.mean l-95% CI u-95% CI eff.samp
## StudyID    0.4745   0.2485   0.7637     2000
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent    0.8985 1.193e-05    2.985     2000
##
##                ~Cont_Country
##
##              post.mean  l-95% CI u-95% CI eff.samp
## Cont_Country    0.1746 3.703e-09    0.536     2000
```

```
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units     1.965    1.783    2.152     2000
##
##  Location effects: logratio ~ broadTreatment
##
##                            post.mean l-95% CI u-95% CI eff.samp  pMCMC
## (Intercept)                  1.45685  0.53638  2.33684     1640  0.010 *
## broadTreatmentCoumaphos     -0.98248 -1.50221 -0.50476     2000 <5e-04 ***
## broadTreatmentFormic_acid   -0.34798 -0.75121  0.04187     2000  0.082 .
## broadTreatmentOxalic_acid   -0.28178 -0.75927  0.18043     2000  0.238
## broadTreatmentPyrethroid    -0.29015 -0.66512  0.07251     2000  0.125
## broadTreatmentThymol        -0.13755 -0.52385  0.21644     2000  0.455
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

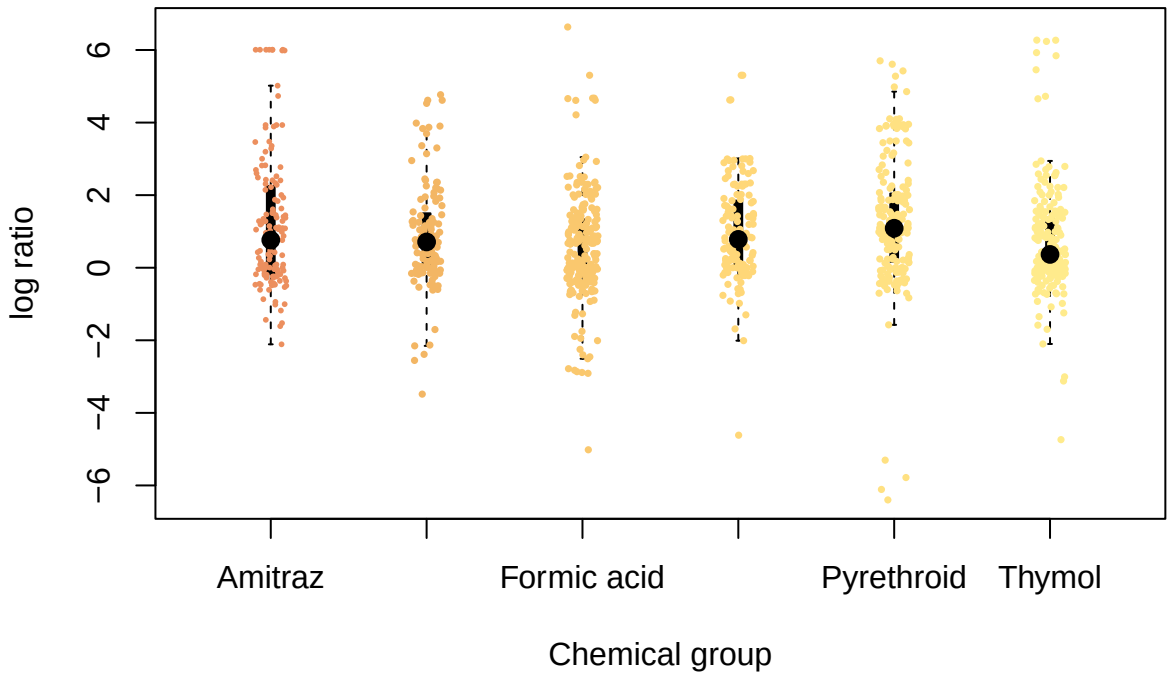
Coumaphos is significantly less effective compared to Amitraz at a ratio of 0.6/1 with all other chemicals not different.

We can do a simple plot of our results.

```
plot(Sub_chem$logratio ~ Sub_chem$broadTreatment,
     col ="black",
     bty = "n",
     boxwex = 0.05,
     ylab = "log ratio",
     xlab = "Chemical group",
     pch = 16,
     cex = 0,
     names = c("Amitraz",
               "Coumaphos",
               "Formic acid",
               "Oxalic acid",
               "Pyrethroid",
               "Thymol"))

#We can plot the points for Amitraz
points(Sub_chem[Sub_chem$broadTreatment == "Amitraz", "logratio"] ~
         jitter(as.integer(Sub_chem[Sub_chem$broadTreatment == "Amitraz",
                                    "broadTreatment"]),
                amount = 0.1),
       pch = 16,
       col = rgb(236, 143, 94, max=255),
       cex = 0.4)

points(median(Sub_chem[Sub_chem$broadTreatment == "Amitraz", "logratio"])
       ~ c(1),
       pch = 16,
       col = "black",
       cex = 1.3)
```

```
#We can plot the points for Coumaphos
points(Sub_chem[Sub_chem$broadTreatment == "Coumaphos", "logratio"] ~
         jitter(as.integer(Sub_chem[Sub_chem$broadTreatment == "Coumaphos",
                                    "broadTreatment"]),amount = 0.1),
       pch = 16,col = rgb(243, 182, 100, max=255),
       cex = 0.5)

points(median(Sub_chem[Sub_chem$broadTreatment == "Coumaphos", "logratio"])
       ~ c(2),
       pch = 16,
       col = "black",
       cex = 1.3)



#We can plot the points for Formic_acid
points(Sub_chem[Sub_chem$broadTreatment == "Formic_acid", "logratio"] ~
         jitter(as.integer(Sub_chem[Sub_chem$broadTreatment == "Formic_acid",
                                    "broadTreatment"]),amount = 0.1),
       pch = 16,
       col = rgb(250, 200, 110, max=255),
       cex = 0.5)

points(median(Sub_chem[Sub_chem$broadTreatment == "Formic_acid", "logratio"])
       ~ c(3),
       pch = 16,
       col = "black",
       cex = 1.3)


#We can plot the points for Oxalic_acid
points(Sub_chem[Sub_chem$broadTreatment == "Oxalic_acid", "logratio"] ~
         jitter(as.integer(Sub_chem[Sub_chem$broadTreatment == "Oxalic_acid",
                                    "broadTreatment"]),
                amount = 0.1),
       pch = 16,
       col = rgb(255, 215, 120, max=255),
       cex = 0.5)

points(median(Sub_chem[Sub_chem$broadTreatment == "Oxalic_acid", "logratio"])
       ~ c(4),
       pch = 16,
       col = "black",
       cex = 1.3)



#We can plot the points for Pyrethroid
points(Sub_chem[Sub_chem$broadTreatment == "Pyrethroid", "logratio"] ~
         jitter(as.integer(Sub_chem[Sub_chem$broadTreatment == "Pyrethroid",
                                    "broadTreatment"]),
                amount = 0.1),
       pch = 16,
```

```
       col = rgb(255, 225, 130, max=255),
       cex = 0.5)

points(median(Sub_chem[Sub_chem$broadTreatment == "Pyrethroid", "logratio"])
       ~ c(5),
       pch = 16,
       col = "black",
       cex = 1.3)


#We can plot the points for Thymol
points(Sub_chem[Sub_chem$broadTreatment == "Thymol", "logratio"] ~
         jitter(as.integer(Sub_chem[Sub_chem$broadTreatment == "Thymol",
                                    "broadTreatment"]),
                amount = 0.1),
       pch = 16,
       col = rgb(255, 235, 140, max=255),
       cex = 0.5)

points(median(Sub_chem[Sub_chem$broadTreatment == "Thymol", "logratio"])
       ~ c(6),
       pch = 16,
       col = "black",
       cex = 1.3)
```



Biological

Lets look at the Biological sub category

```r
Full_bio <-  Full_comb_data[Full_comb_data$categoryTreatment == "Biological",]

Full_bio$SubCategory1Treatment <- factor(Full_bio$SubCategory1Treatment)

nitt_b <- 2200000
thining_b <- 1000
burnin_b <- 200000

 mod_Full_bio <- MCMCglmm(logratio ~ SubCategory1Treatment,
                                rcov=~units,
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = Full_bio,
                                nitt = nitt_b,
                                thin = thining_b,
                                burnin = burnin_b,
                                verbose = FALSE
                            )


mod_Full_bio2 <- MCMCglmm(logratio ~ SubCategory1Treatment,
                                rcov=~units,
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = Full_bio,
                                nitt = nitt_b,
                                thin = thining_b,
                                burnin = burnin_b,
                                verbose = FALSE
                            )


mod_Full_bio3 <- MCMCglmm(logratio ~ SubCategory1Treatment,
                                rcov=~units,
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = Full_bio,
                                nitt = nitt_b,
                                thin = thining_b,
                                burnin = burnin_b,
                                verbose = FALSE
                            )

#Check the fixed terms
mod_Full_bio_Sol_conv <- gelman.diag(mcmc.list(mod_Full_bio$Sol,
                                               mod_Full_bio2$Sol,
                                               mod_Full_bio3$Sol))
```

```
mod_Full_bio_Sol_conv
```

```
## Potential scale reduction factors:
##
##                                       Point est. Upper C.I.
## (Intercept)                                   1          1
## SubCategory1TreatmentNatural_enemies          1          1
##
## Multivariate psrf
##
## 1
```

```
#Check the random terms
mod_Full_bio_VCV_conv <- gelman.diag(mcmc.list(mod_Full_bio$VCV,
                                               mod_Full_bio2$VCV,
                                               mod_Full_bio3$VCV))
mod_Full_bio_VCV_conv
```

```
## Potential scale reduction factors:
##
##                  Point est. Upper C.I.
## StudyID_control        1.01       1.02
## Continent              1.03       1.03
## Cont_Country           1.01       1.01
## units                  1.00       1.00
##
## Multivariate psrf
##
## 1
```

```
summary(mod_Full_bio)
```

```
##
##  Iterations = 200001:2199001
##  Thinning interval  = 1000
##  Sample size  = 2000
##
##  DIC: 880.9511
##
##  G-structure:  ~StudyID_control
##
##                  post.mean  l-95% CI u-95% CI eff.samp
## StudyID_control  0.007654 6.994e-17  0.05205     1743
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent   0.01105 1.818e-17  0.02537     2000
##
##                ~Cont_Country
##
##              post.mean  l-95% CI u-95% CI eff.samp
```

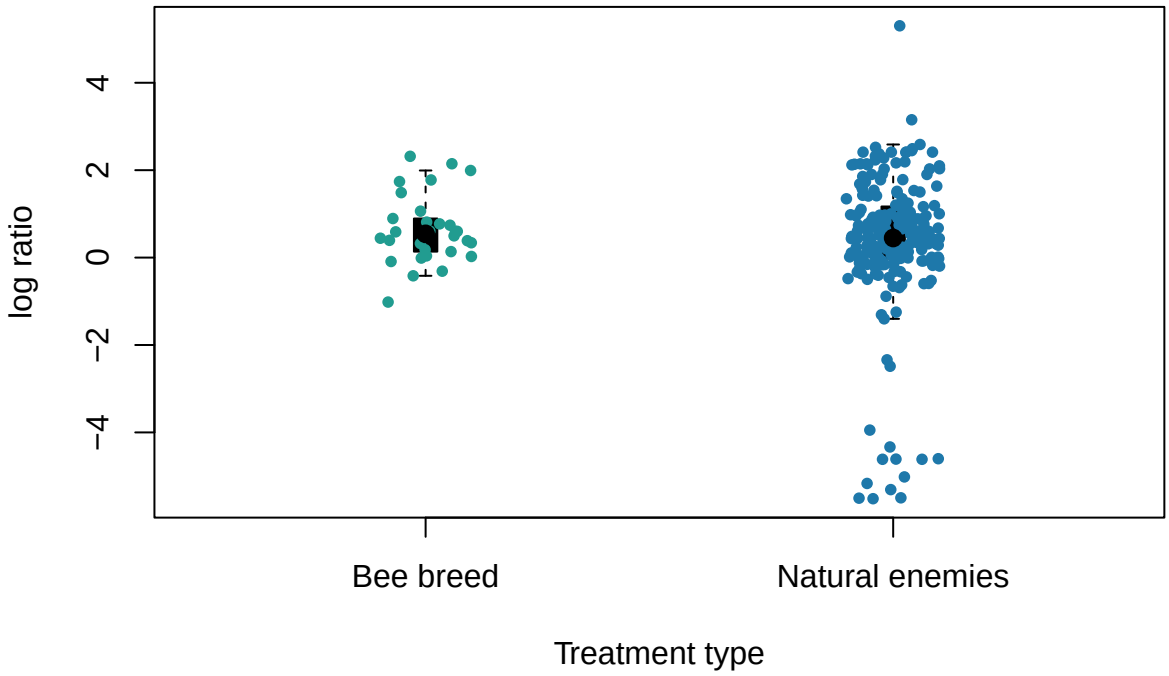
```
## Cont_Country   0.03504 5.636e-17   0.1866     2000
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units     2.402    1.976    2.836     2000
##
##  Location effects: logratio ~ SubCategory1Treatment
##
##                                     post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)                           0.58512  0.01582  1.22757     2000 0.066
## SubCategory1TreatmentNatural_enemies -0.24929 -0.81904  0.40176     2000 0.415
##
## (Intercept)                          .
## SubCategory1TreatmentNatural_enemies
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Here bee bread has a significant effect compared to the null of about 2/1 with no significant difference of the effect of Natural enemies.

We can do a simple plot of our results.

```
plot(Full_bio$logratio ~ Full_bio$SubCategory1Treatment,
     col ="black",
     bty = "n",
     boxwex = 0.05,
     ylab = "log ratio",
     xlab = "Treatment type",
     pch = 16,
     cex = 0,
     names = c("Bee breed", "Natural enemies"))


#We can plot the points for Bee_breed
points(Full_bio[Full_bio$SubCategory1Treatment == "Bee_breed", "logratio"] ~
         jitter(as.integer(Full_bio[Full_bio$SubCategory1Treatment == "Bee_breed",
                  amount = 0.1),
       pch = 16,
       col = rgb(33, 156, 144, max=255),
       cex = 0.8)

points(median(Full_bio[Full_bio$SubCategory1Treatment == "Bee_breed", "logratio"])
       ~ c(1),
       pch = 16,
       col = "black",
       cex = 1.3)


#We can plot the points for Natural_enemies
points(Full_bio[Full_bio$SubCategory1Treatment == "Natural_enemies", "logratio"] ~
         jitter(as.integer(Full_bio[Full_bio$SubCategory1Treatment == "Natural_enemies",
                  amount = 0.1),
       pch = 16,
```

```
       col = rgb(30, 120, 170, max=255),
       cex = 0.8)


points(median(Full_bio[Full_bio$SubCategory1Treatment == "Natural_enemies",
                       "logratio"]) ~ c(2),
       pch = 16,
       col = "black",
       cex = 1.3)
```



Life stage

Models of life stage target

We will look at life stage separately for bees and Varroa.

Bees life stage

First lets do it for bees. We will create 6 groups Synthetic chemical treatment on Honey bee worker, Synthetic chemical treatment on juvenile, Organic chemical treatment on Honey bee worker, Organic chemical treatment on juvenile, Biological treatment on Honey bee worker and Biological treatment on juveniles.

```
life_bees_data <- Full_bees_data[Full_bees_data$ResponseVariableTarget %in%
                                   c("Honey_bee_worker",
                                     "Honey_bee_juvenile"),]

life_bees_data <- life_bees_data[life_bees_data$chem_split %in%
                                   c("Agriculturally_Organic",
```

```
                                    "Synthetic",
                                    "Biological"),]

life_bees_data$life_treat <- paste(life_bees_data$chem_split,
                                   life_bees_data$ResponseVariableTarget,
                                   sep = "_")

life_bees_data$life_treat <- factor(life_bees_data$life_treat,
                                    levels = c("Synthetic_Honey_bee_worker",
                                               "Synthetic_Honey_bee_juvenile",
                                               "Agriculturally_Organic_Honey_bee_worker",
                                               "Agriculturally_Organic_Honey_bee_juvenile",
                                               "Biological_Honey_bee_worker",
                                               "Biological_Honey_bee_juvenile"))

mod_bees_life <- MCMCglmm(logratio ~ life_treat,
                                   rcov=~units,
                                   random =~StudyID_control
                                            + Continent
                                            + Cont_Country,
                                   family ="gaussian",
                                   data = life_bees_data,
                                   nitt = nitt,
                                   thin = thining,
                                   burnin = burnin,
                                   prior = prior_d,
                                   verbose = FALSE
                                  )


mod_bees_life2 <- MCMCglmm(logratio ~ life_treat,
                                   rcov=~units,
                                   random =~StudyID_control
                                            + Continent
                                            + Cont_Country,
                                   family ="gaussian",
                                   data = life_bees_data,
                                   nitt = nitt,
                                   thin = thining,
                                   burnin = burnin,
                                   prior = prior_d,
                                   verbose = FALSE
                                  )

mod_bees_life3 <- MCMCglmm(logratio ~ life_treat,
                                   rcov=~units,
                                   random =~StudyID_control
                                            + Continent
                                            + Cont_Country,
                                   family ="gaussian",
                                   data = life_bees_data,
                                   nitt = nitt,
                                   thin = thining,
```

```
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )


#Check the fixed terms
mod_bees_life_Sol_conv <- gelman.diag(mcmc.list(mod_bees_life$Sol,
                                                mod_bees_life2$Sol,
                                                mod_bees_life3$Sol))
mod_bees_life_Sol_conv
```

```
## Potential scale reduction factors:
##
##                                                        Point est. Upper C.I.
## (Intercept)                                                    1          1
## life_treatSynthetic_Honey_bee_juvenile                         1          1
## life_treatAgriculturally_Organic_Honey_bee_worker              1          1
## life_treatAgriculturally_Organic_Honey_bee_juvenile            1          1
## life_treatBiological_Honey_bee_worker                          1          1
## life_treatBiological_Honey_bee_juvenile                        1          1
##
## Multivariate psrf
##
## 1
```

```
#Check the random terms
mod_bees_life_VCV_conv <- gelman.diag(mcmc.list(mod_bees_life$VCV,
                                                mod_bees_life2$VCV,
                                                mod_bees_life3$VCV))
mod_bees_life_VCV_conv
```

```
## Potential scale reduction factors:
##
##                  Point est. Upper C.I.
## StudyID_control          1          1
## Continent                1          1
## Cont_Country             1          1
## units                    1          1
##
## Multivariate psrf
##
## 1
```

```
summary(mod_bees_life)
```

```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
##
##  DIC: 2013.38
```

```
##
##  G-structure:  ~StudyID_control
##
##                 post.mean l-95% CI u-95% CI eff.samp
## StudyID_control     1.362   0.7741    1.949     2000
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent    0.2337 2.513e-08   0.9421     2000
##
##                ~Cont_Country
##
##              post.mean  l-95% CI u-95% CI eff.samp
## Cont_Country    0.1554 1.277e-09   0.6336     1789
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units      1.18    1.049    1.318     1926
##
##  Location effects: logratio ~ life_treat
##
##                                                    post.mean  l-95% CI
## (Intercept)                                        -0.797431 -1.427465
## life_treatSynthetic_Honey_bee_juvenile              0.518714 -0.225104
## life_treatAgriculturally_Organic_Honey_bee_worker   0.309497 -0.005108
## life_treatAgriculturally_Organic_Honey_bee_juvenile 0.356450 -0.069111
## life_treatBiological_Honey_bee_worker               0.073563 -0.674428
## life_treatBiological_Honey_bee_juvenile             1.549998  0.723729
##                                                    u-95% CI eff.samp  pMCMC
## (Intercept)                                        -0.196140     2000  0.030
## life_treatSynthetic_Honey_bee_juvenile              1.284196     2000  0.186
## life_treatAgriculturally_Organic_Honey_bee_worker   0.712953     2398  0.076
## life_treatAgriculturally_Organic_Honey_bee_juvenile 0.841152     2000  0.130
## life_treatBiological_Honey_bee_worker               0.808894     1848  0.883
## life_treatBiological_Honey_bee_juvenile             2.374100     2000 <5e-04
##
## (Intercept)                                        *
## life_treatSynthetic_Honey_bee_juvenile
## life_treatAgriculturally_Organic_Honey_bee_worker  .
## life_treatAgriculturally_Organic_Honey_bee_juvenile
## life_treatBiological_Honey_bee_worker
## life_treatBiological_Honey_bee_juvenile            ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

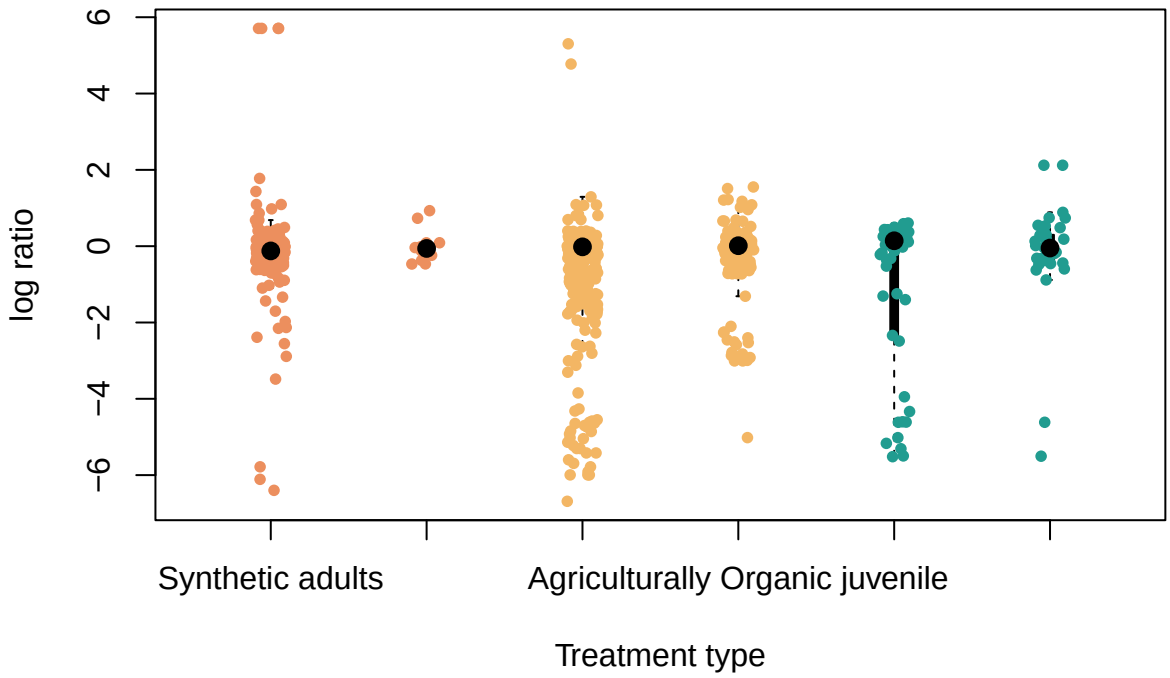
Overall synthetic chemical had a significant negative effect on adult bees with outcomes in controls 1.62 times better than the corresponding chemical treatment. There was some weak support for a less negative effect for synthetic chemical on juveniles, however, controls where still 1.12 times better than synthetic chemical treatments. Organic chemicals showed a similar response to synthetic chemicals on adults and also showed a significantly less negative effect on juveniles, however, controls still performed 1.28 times better. While biological treatments were not significantly different to synthetic treatments on adults, they were significant more positive, at a ratio of 2.57 when the treatment was applied to juveniles.

We can do a simple plot of our results.

```
plot(life_bees_data$logratio ~ life_bees_data$life_treat,
     col ="black",
     bty = "n",
     boxwex = 0.05,
     ylab = "log ratio",
     xlab = "Treatment type",
     pch = 16,
     cex = 0,
     names = c("Synthetic adults",
               "Synthetic juvenile",
               "Agriculturally Organic worker",
               "Agriculturally Organic juvenile",
               "Biological adults",
               "Biological juvenile"))


#We can plot the points for Synthetic_Honey_bee_worker
points(life_bees_data[life_bees_data$life_treat == "Synthetic_Honey_bee_worker", "logratio"] ~
         jitter(as.integer(life_bees_data[life_bees_data$life_treat == "Synthetic_Honey_bee_worker",
                  amount = 0.1),
       pch = 16,
       col = rgb(236, 143, 94, max=255),
       cex = 0.8)

points(hdr(life_bees_data[life_bees_data$life_treat == "Synthetic_Honey_bee_worker",
                          "logratio"])$mode
       ~ c(1),
       pch = 16,
       col = "black",
       cex = 1.3)


#We can plot the points for Synthetic_Honey_bee_juvenile
points(life_bees_data[life_bees_data$life_treat == "Synthetic_Honey_bee_juvenile",
                        "logratio"] ~
         jitter(as.integer(life_bees_data[life_bees_data$life_treat == "Synthetic_Honey_bee_juvenile",
                  amount = 0.1),
       pch = 16,
       col = rgb(236, 143, 94, max=255),
       cex = 0.8)

points(hdr(life_bees_data[life_bees_data$life_treat == "Synthetic_Honey_bee_juvenile",
                          "logratio"])$mode
       ~ c(2),
       pch = 16,
       col = "black",
       cex = 1.3)



#We can plot the points for Agriculturally_Organic_Honey_bee_worker
points(life_bees_data[life_bees_data$life_treat == "Agriculturally_Organic_Honey_bee_worker",
```

```
                    "logratio"] ~
        jitter(as.integer(life_bees_data[life_bees_data$life_treat == "Agriculturally_Organic_Honey_bee
                 amount = 0.1),
       pch = 16,
       col = rgb(243, 182, 100, max=255),
       cex = 0.8)

points(hdr(life_bees_data[life_bees_data$life_treat == "Agriculturally_Organic_Honey_bee_worker",
                    "logratio"])$mode
       ~ c(3),
       pch = 16,
       col = "black",
       cex = 1.3)



#We can plot the points for Agriculturally_Organic_Honey_bee_juvenile
points(life_bees_data[life_bees_data$life_treat == "Agriculturally_Organic_Honey_bee_juvenile",
                    "logratio"] ~
        jitter(as.integer(life_bees_data[life_bees_data$life_treat == "Agriculturally_Organic_Honey_bee
                 amount = 0.1),
       pch = 16,
       col = rgb(243, 182, 100, max=255),
       cex = 0.8)

points(hdr(life_bees_data[life_bees_data$life_treat == "Agriculturally_Organic_Honey_bee_juvenile",
                    "logratio"])$mode
       ~ c(4),
       pch = 16,
       col = "black",
       cex = 1.3)



#We can plot the points for Biological_Honey_bee_worker
points(life_bees_data[life_bees_data$life_treat == "Biological_Honey_bee_worker",
                    "logratio"] ~
        jitter(as.integer(life_bees_data[life_bees_data$life_treat == "Biological_Honey_bee_worker", "
                 amount = 0.1),
       pch = 16,
       col = rgb(33, 156, 144, max=255),
       cex = 0.8)

points(hdr(life_bees_data[life_bees_data$life_treat == "Biological_Honey_bee_worker",
                    "logratio"])$mode
       ~ c(5),
       pch = 16,
       col = "black",
       cex = 1.3)
```

```
#We can plot the points for Biological_Honey_bee_juvenile
points(life_bees_data[life_bees_data$life_treat == "Biological_Honey_bee_juvenile",
                      "logratio"] ~
         jitter(as.integer(life_bees_data[life_bees_data$life_treat == "Biological_Honey_bee_juvenile",
                  amount = 0.1),
       pch = 16,
       col = rgb(33, 156, 144, max=255),
       cex = 0.8)

points(hdr(life_bees_data[life_bees_data$life_treat == "Biological_Honey_bee_juvenile",
                      "logratio"])$mode
       ~ c(6),
       pch = 16,
       col = "black",
       cex = 1.3)
```



Supplementary analaysis

We can also look at the main model when split into just studies measruing increases in bees, which is for 44 studies for this analysis

```
mod_HoneyBeeIncrease <- MCMCglmm(logratio ~ chem_split,
                                 rcov=~units,
                                 random =~StudyID_control
                                          + Continent
                                          + Cont_Country,
                                 family ="gaussian",
                                 data = HoneyBeeIncrease_paired_fin,
```

```
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )

mod_HoneyBeeIncrease2 <- MCMCglmm(logratio ~ chem_split,
                                rcov=~units,
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = HoneyBeeIncrease_paired_fin,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )

mod_HoneyBeeIncrease3 <- MCMCglmm(logratio ~ chem_split,
                                rcov=~units,
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = HoneyBeeIncrease_paired_fin,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )

#Check the fixed terms
mod_HoneyBeeIncrease_Sol_conv <- gelman.diag(mcmc.list(mod_HoneyBeeIncrease$Sol,
                                      mod_HoneyBeeIncrease2$Sol,
                                      mod_HoneyBeeIncrease3$Sol))
mod_HoneyBeeIncrease_Sol_conv
```

```
## Potential scale reduction factors:
##
##                                   Point est. Upper C.I.
## (Intercept)                                1          1
## chem_splitAgriculturally_Organic           1          1
## chem_splitBiological                       1          1
## chem_splitPhysical                         1          1
## chem_splitMixed                            1          1
##
## Multivariate psrf
##
## 1
```

```r
#Check the random terms
mod_HoneyBeeIncrease_VCV_conv <- gelman.diag(mcmc.list(mod_HoneyBeeIncrease$VCV,
                                     mod_HoneyBeeIncrease2$VCV,
                                     mod_HoneyBeeIncrease3$VCV))
mod_HoneyBeeIncrease_VCV_conv
```

```
## Potential scale reduction factors:
##
##                 Point est. Upper C.I.
## StudyID_control       1.00       1.00
## Continent             1.08       1.08
## Cont_Country          1.00       1.01
## units                 1.00       1.00
##
## Multivariate psrf
##
## 1
```

```r
summary(mod_HoneyBeeIncrease)
```

```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
##
##  DIC: 648.0591
##
##  G-structure:  ~StudyID_control
##
##                  post.mean l-95% CI u-95% CI eff.samp
## StudyID_control    0.1428   0.0415   0.2612     1798
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent    0.2891 4.683e-06   0.9942     1472
##
##                ~Cont_Country
##
##              post.mean  l-95% CI u-95% CI eff.samp
## Cont_Country   0.05574 1.038e-08    0.189     1701
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units    0.2204   0.1891   0.2508     2000
##
##  Location effects: logratio ~ chem_split
##
##                                  post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)                       -0.09049 -0.59992  0.46888     1873 0.650
## chem_splitAgriculturally_Organic   0.16377  0.02858  0.30504     2000 0.017 *
```

```
## chem_splitBiological               0.50865  0.21070  0.82232     2000 0.001 ***
## chem_splitPhysical                 0.28217 -0.22749  0.90030     2214 0.297
## chem_splitMixed                   -0.04596 -1.16043  0.98001     2000 0.935
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

While not significant synthetic chemicals have weak support for a negative effect on bee pops, but organic chemical have a significantly positive effect compared to synthetic chemicals. This difference shows that organic chemicals have a neutral effect on bee pops while synthetic are likely worse than controls.Biological treatments have a positive effect on bee pops at a ratio of 1.6/1 with no other significant effects.

Studies that measure a decrease bee pops

This is with the original data so a positive value here is something that decreases bee pops and health. There is 36 studies here.

```
mod_HoneyBeeReduction <- MCMCglmm(logratio ~ chem_split,
                                  rcov=~units,
                                  random =~StudyID_control
                                           + Continent
                                           + Cont_Country,
                                  family ="gaussian",
                                  data = HoneyBeeReduction_paired_fin,
                                  nitt = nitt,
                                  thin = thining,
                                  burnin = burnin,
                                  prior = prior_d,
                                  verbose = FALSE
                              )
```

```
## Warning in MCMCglmm(logratio ~ chem_split, rcov = ~units, random =
## ~StudyID_control + : some fixed effects are not estimable and have been
## removed. Use singular.ok=TRUE to sample these effects, but use an informative
## prior!
```

```
mod_HoneyBeeReduction2 <- MCMCglmm(logratio ~ chem_split,
                                  rcov=~units,
                                  random =~StudyID_control
                                           + Continent
                                           + Cont_Country,
                                  family ="gaussian",
                                  data = HoneyBeeReduction_paired_fin,
                                  nitt = nitt,
                                  thin = thining,
                                  burnin = burnin,
                                  prior = prior_d,
                                  verbose = FALSE
                              )
```

```
## Warning in MCMCglmm(logratio ~ chem_split, rcov = ~units, random =
## ~StudyID_control + : some fixed effects are not estimable and have been
```

```
## removed. Use singular.ok=TRUE to sample these effects, but use an informative
## prior!
```

```
mod_HoneyBeeReduction3 <- MCMCglmm(logratio ~ chem_split,
                                   rcov=~units,
                                   random =~StudyID_control
                                            + Continent
                                            + Cont_Country,
                                   family ="gaussian",
                                   data = HoneyBeeReduction_paired_fin,
                                   nitt = nitt,
                                   thin = thining,
                                   burnin = burnin,
                                   prior = prior_d,
                                   verbose = FALSE
                                )
```

```
## Warning in MCMCglmm(logratio ~ chem_split, rcov = ~units, random =
## ~StudyID_control + : some fixed effects are not estimable and have been
## removed. Use singular.ok=TRUE to sample these effects, but use an informative
## prior!
```

```
#Check the fixed terms
mod_HoneyBeeReduction_Sol_conv<- gelman.diag(mcmc.list(mod_HoneyBeeReduction$Sol,
                                      mod_HoneyBeeReduction2$Sol,
                                      mod_HoneyBeeReduction3$Sol))
mod_HoneyBeeReduction_Sol_conv
```

```
## Potential scale reduction factors:
##
##                                   Point est. Upper C.I.
## (Intercept)                                1          1
## chem_splitAgriculturally_Organic           1          1
## chem_splitBiological                       1          1
## chem_splitMixed                            1          1
##
## Multivariate psrf
##
## 1
```

```
#Check the random terms
mod_HoneyBeeReduction_VCV_conv <- gelman.diag(mcmc.list(mod_HoneyBeeReduction$VCV,
                                      mod_HoneyBeeReduction2$VCV,
                                      mod_HoneyBeeReduction3$VCV))
mod_HoneyBeeReduction_VCV_conv
```

```
## Potential scale reduction factors:
##
##                  Point est. Upper C.I.
## StudyID_control        1.00       1.00
## Continent              1.04       1.04
## Cont_Country           1.01       1.01
```

```
## units                 1.00       1.01
##
## Multivariate psrf
##
## 1
```

```
summary(mod_HoneyBeeReduction)
```

```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
##
##  DIC: 1271.961
##
##  G-structure:  ~StudyID_control
##
##                  post.mean l-95% CI u-95% CI eff.samp
## StudyID_control     1.893   0.6985     3.25     2000
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent    0.9579 5.341e-08    3.411     2000
##
##                ~Cont_Country
##
##              post.mean  l-95% CI u-95% CI eff.samp
## Cont_Country     0.503 7.694e-09    1.886     1847
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units     2.828     2.42    3.348     1868
##
##  Location effects: logratio ~ chem_split
##
##                                  post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)                         1.6162   0.2363   3.0299     2000 0.024 *
## chem_splitAgriculturally_Organic   -0.7152  -1.7415   0.2058     2000 0.148
## chem_splitBiological               -0.3979  -1.9885   1.2143     2000 0.616
## chem_splitMixed                    -1.8689  -4.8116   1.3808     2669 0.238
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We find no significant effects, likely due to the low sample size.

Studies that measure a decreases in varroa pops

Analysis just including measures of Varroa decreases.

```
mod_VarroaReduction <- MCMCglmm(logratio ~ chem_split,
                                rcov=~units,
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = VarroaReduction_paired_fin,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE)


mod_VarroaReduction2 <- MCMCglmm(logratio ~ chem_split,
                                rcov=~units,
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = VarroaReduction_paired_fin,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE)

mod_VarroaReduction3 <- MCMCglmm(logratio ~ chem_split,
                                rcov=~units,
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = VarroaReduction_paired_fin,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE)

#Check the fixed terms
mod_VarroaReduction_Sol_conv<- gelman.diag(mcmc.list(mod_VarroaReduction$Sol,
                                      mod_VarroaReduction2$Sol,
                                      mod_VarroaReduction3$Sol))
mod_VarroaReduction_Sol_conv
```

```
## Potential scale reduction factors:
##
##                                 Point est. Upper C.I.
## (Intercept)                              1          1
## chem_splitAgriculturally_Organic         1          1
## chem_splitBiological                     1          1
## chem_splitPhysical                       1          1
```

```
## chem_splitMixed                          1          1
##
## Multivariate psrf
##
## 1
```

```
#Check the random terms
mod_VarroaReduction_VCV_conv <- gelman.diag(mcmc.list(mod_VarroaReduction$VCV,
                                      mod_VarroaReduction2$VCV,
                                      mod_VarroaReduction3$VCV))
mod_VarroaReduction_VCV_conv
```

```
## Potential scale reduction factors:
##
##                  Point est. Upper C.I.
## StudyID_control       1.00       1.00
## Continent             1.02       1.02
## Cont_Country          1.00       1.01
## units                 1.00       1.00
##
## Multivariate psrf
##
## 1
```

```
summary(mod_VarroaReduction)
```

```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
##
##  DIC: 4399.34
##
##  G-structure:  ~StudyID_control
##
##                 post.mean l-95% CI u-95% CI eff.samp
## StudyID_control     1.255    0.856    1.685     2000
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent    0.1486 3.593e-10   0.5385     1775
##
##                ~Cont_Country
##
##              post.mean l-95% CI u-95% CI eff.samp
## Cont_Country   0.06476 1.67e-08   0.2475     2000
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units     1.737    1.597    1.888     2000
##
```

```
##  Location effects: logratio ~ chem_split
##
##                                  post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)                        1.97562  1.48730  2.48675     2000 0.001 ***
## chem_splitAgriculturally_Organic  -0.18144 -0.54730  0.13673     2159 0.301
## chem_splitBiological              -0.75383 -1.40689 -0.00974     2844 0.032 *
## chem_splitPhysical                -1.22758 -2.93877  0.51035     2000 0.153
## chem_splitMixed                    0.32030 -2.57061  3.60742     2000 0.834
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Chemical studies are found to reduce Varroa at a ratio of about 5 to 1 when compared to controls. Organic chemical are slightly less effective a reducing varrora with a ratio of 4 to 1 while biological data reduce Varroa at a ratio of 2.5 to 1.

Controls that measure increases in varroa

Analysis just including measures of Varroa increases. Here a negative estimates indicate posative outcomes for bee populations and health.

```
mod_VarroaIncrease <- MCMCglmm(logratio ~ chem_split,
                               rcov=~units,
                               random =~StudyID_control
                                        + Continent
                                        + Cont_Country,
                               family ="gaussian",
                               data = VarroaIncrease_paired_fin,
                               nitt = nitt,
                               thin = thining,
                               burnin = burnin,
                               prior = prior_d,
                               verbose = FALSE
                           )


mod_VarroaIncrease2 <- MCMCglmm(logratio ~ chem_split,
                               rcov=~units,
                               random =~StudyID_control
                                        + Continent
                                        + Cont_Country,
                               family ="gaussian",
                               data = VarroaIncrease_paired_fin,
                               nitt = nitt,
                               thin = thining,
                               burnin = burnin,
                               prior = prior_d,
                               verbose = FALSE
                           )


mod_VarroaIncrease3 <- MCMCglmm(logratio ~ chem_split,
                               rcov=~units,
```

```
                                random =~StudyID_control
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = VarroaIncrease_paired_fin,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )



#Check the fixed terms
mod_VarroaIncrease_Sol_conv <- gelman.diag(mcmc.list(mod_VarroaIncrease$Sol,
                                        mod_VarroaIncrease2$Sol,
                                        mod_VarroaIncrease3$Sol))
mod_VarroaIncrease_Sol_conv
```

```
## Potential scale reduction factors:
##
##                                   Point est. Upper C.I.
## (Intercept)                                1       1.00
## chem_splitAgriculturally_Organic           1       1.00
## chem_splitBiological                       1       1.00
## chem_splitPhysical                         1       1.01
## chem_splitMixed                            1       1.00
##
## Multivariate psrf
##
## 1
```

```
#Check the random terms
mod_VarroaIncrease_VCV_conv <- gelman.diag(mcmc.list(mod_VarroaIncrease$VCV,
                                        mod_VarroaIncrease2$VCV,
                                        mod_VarroaIncrease3$VCV))
mod_VarroaIncrease_VCV_conv
```

```
## Potential scale reduction factors:
##
##                  Point est. Upper C.I.
## StudyID_control        1.00       1.01
## Continent              1.23       1.24
## Cont_Country           1.00       1.01
## units                  1.00       1.00
##
## Multivariate psrf
##
## 1
```

```
summary(mod_VarroaIncrease)
```

```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
##
##  DIC: 1322.524
##
##  G-structure:  ~StudyID_control
##
##                  post.mean l-95% CI u-95% CI eff.samp
## StudyID_control    0.7315   0.3395    1.191     2000
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent     0.277 1.117e-08   0.9556     2000
##
##                ~Cont_Country
##
##              post.mean  l-95% CI u-95% CI eff.samp
## Cont_Country    0.1437 9.006e-07   0.5914     1703
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units      1.33    1.129    1.511     2000
##
##  Location effects: logratio ~ chem_split
##
##                                  post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)                        -1.2743  -1.9116  -0.5389     2278 0.002 **
## chem_splitAgriculturally_Organic   -0.1109  -0.4557   0.2332     2000 0.540
## chem_splitBiological                0.8377   0.2600   1.3896     1823 0.005 **
## chem_splitPhysical                  0.5213  -0.6118   1.6718     2000 0.385
## chem_splitMixed                     0.7361  -0.6783   2.1278     2000 0.299
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Similar to the results for studies measuring Varroa decrease but in reverse. Synthetic chemicals are found to have a decrease in Varroa at a ratio of 4.6/1 when compared to controls and is significantly different compared to the Null. Organic chemicals are not significantly different compared to synthetic. Biological treatments are significantly less effective when compared to synthetic chemicals with only a ratio of 1.15/1 when compared to controls.

Chem specific effect on bees

Model comparing the chemical groups for just studies on bees

```
temp_Sub_chem_HI <- rbind(HoneyBeeReduction_paired_red_fin,
                          HoneyBeeIncrease_paired_fin)

#combine Flumethrin and Fluvalinate and call them Pyrethroid
temp_Sub_chem_HI[temp_Sub_chem_HI$broadTreatment %in% c("Flumethrin",
                                                        "Fluvalinate"),
                 "broadTreatment"] <- "Pyrethroid"

Sub_chem_HI <- temp_Sub_chem_HI[temp_Sub_chem_HI$broadTreatment %in%
c("Amitraz",
  "Coumaphos",
  "Thymol",
  "Oxalic_acid",
  "Formic_acid",
  "Pyrethroid"), ]
```

Now we can run the model comparing chemicals for studies that measured bees.

```
mod_spec_chem_HI <- MCMCglmm(logratio ~ broadTreatment,
                                rcov=~units,
                                random =~StudyID
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = Sub_chem_HI,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                             )

mod_spec_chem_HI2 <- MCMCglmm(logratio ~ broadTreatment,
                                rcov=~units,
                                random =~StudyID
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = Sub_chem_HI,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                             )


mod_spec_chem_HI3 <- MCMCglmm(logratio ~ broadTreatment,
                                rcov=~units,
                                random =~StudyID
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
```

```
                                data = Sub_chem_HI,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )

#Check the fixed terms
mod_spec_chem_HI_Sol_conv <- gelman.diag(mcmc.list(mod_spec_chem_HI$Sol,
                                        mod_spec_chem_HI2$Sol,
                                        mod_spec_chem_HI3$Sol))
mod_spec_chem_HI_Sol_conv
```

```
## Potential scale reduction factors:
##
##                            Point est. Upper C.I.
## (Intercept)                         1          1
## broadTreatmentCoumaphos             1          1
## broadTreatmentFormic_acid           1          1
## broadTreatmentOxalic_acid           1          1
## broadTreatmentPyrethroid            1          1
## broadTreatmentThymol                1          1
##
## Multivariate psrf
##
## 1
```

```
#Check the random terms
mod_spec_chem_HI_VCV_conv <- gelman.diag(mcmc.list(mod_spec_chem_HI$VCV,
                                        mod_spec_chem_HI2$VCV,
                                        mod_spec_chem_HI3$VCV))
mod_spec_chem_HI_VCV_conv
```

```
## Potential scale reduction factors:
##
##              Point est. Upper C.I.
## StudyID            1.00       1.00
## Continent          1.26       1.28
## Cont_Country       1.00       1.01
## units              1.00       1.00
##
## Multivariate psrf
##
## 1
```

```
summary(mod_spec_chem_HI)
```

```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
```

```
##
##  DIC: 1017.798
##
##  G-structure:  ~StudyID
##
##         post.mean l-95% CI u-95% CI eff.samp
## StudyID    0.8994   0.3929    1.543     1602
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent    0.4255 1.701e-08    1.412     1490
##
##                ~Cont_Country
##
##              post.mean  l-95% CI u-95% CI eff.samp
## Cont_Country    0.3286 6.723e-11   0.9888     1775
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units    0.7622   0.6542   0.8877     2165
##
##  Location effects: logratio ~ broadTreatment
##
##                            post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)                 -0.42893 -1.20053  0.31795     2012 0.226
## broadTreatmentCoumaphos     -0.07394 -0.68847  0.49481     2000 0.815
## broadTreatmentFormic_acid    0.14224 -0.29909  0.53187     2000 0.480
## broadTreatmentOxalic_acid    0.40672 -0.13758  1.00625     2000 0.184
## broadTreatmentPyrethroid    -0.34259 -0.76426  0.09856     2000 0.117
## broadTreatmentThymol         0.13001 -0.19440  0.51679     2000 0.455
```

Chem specific effect on varroa

Model comparing the chemical groups for just studies on varroa

```
temp_Sub_chem_Vb <- rbind(VarroaReduction_paired_fin,
                          VarroaIncrease_paired_red_fin)

#combine Flumethrin and Fluvalinate and call them Pyrethroid
temp_Sub_chem_Vb[temp_Sub_chem_Vb$broadTreatment %in% c("Flumethrin",
                                                        "Fluvalinate"),
                 "broadTreatment"] <- "Pyrethroid"

Sub_chem_Vb <- temp_Sub_chem_Vb[temp_Sub_chem_Vb$broadTreatment %in%
c("Amitraz",
  "Coumaphos",
  "Thymol",
  "Oxalic_acid",
  "Formic_acid",
  "Pyrethroid"), ]
```

```
mod_spec_chem_Vb <- MCMCglmm(logratio ~ broadTreatment,
                             rcov=~units,
                             random =~StudyID
                                      + Continent
                                      + Cont_Country,
                             family ="gaussian",
                             data = Sub_chem_Vb,
                             nitt = nitt,
                             thin = thining,
                             burnin = burnin,
                             prior = prior_d,
                             verbose = FALSE
                            )


mod_spec_chem_Vb2 <- MCMCglmm(logratio ~ broadTreatment,
                             rcov=~units,
                             random =~StudyID
                                      + Continent
                                      + Cont_Country,
                             family ="gaussian",
                             data = Sub_chem_Vb,
                             nitt = nitt,
                             thin = thining,
                             burnin = burnin,
                             prior = prior_d,
                             verbose = FALSE
                            )


mod_spec_chem_Vb3 <- MCMCglmm(logratio ~ broadTreatment,
                             rcov=~units,
                             random =~StudyID
                                      + Continent
                                      + Cont_Country,
                             family ="gaussian",
                             data = Sub_chem_Vb,
                             nitt = nitt,
                             thin = thining,
                             burnin = burnin,
                             prior = prior_d,
                             verbose = FALSE
                            )

#Check the fixed terms
mod_spec_chem_Vb_Sol_conv <- gelman.diag(mcmc.list(mod_spec_chem_Vb$Sol,
                                       mod_spec_chem_Vb2$Sol,
                                       mod_spec_chem_Vb3$Sol))
mod_spec_chem_Vb_Sol_conv
```

```
## Potential scale reduction factors:
##
##                           Point est. Upper C.I.
```

```
## (Intercept)                          1       1.00
## broadTreatmentCoumaphos              1       1.01
## broadTreatmentFormic_acid            1       1.01
## broadTreatmentOxalic_acid            1       1.01
## broadTreatmentPyrethroid             1       1.01
## broadTreatmentThymol                 1       1.00
##
## Multivariate psrf
##
## 1
```

```
#Check the random terms
mod_spec_chem_Vb_VCV_conv <- gelman.diag(mcmc.list(mod_spec_chem_Vb$VCV,
                                      mod_spec_chem_Vb2$VCV,
                                      mod_spec_chem_Vb3$VCV))
mod_spec_chem_Vb_VCV_conv
```

```
## Potential scale reduction factors:
##
##              Point est. Upper C.I.
## StudyID            1.00       1.00
## Continent          1.00       1.01
## Cont_Country       1.01       1.01
## units              1.00       1.00
##
## Multivariate psrf
##
## 1
```

```
summary(mod_spec_chem_Vb)
```

```
##
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
##
##  DIC: 1972.905
##
##  G-structure:  ~StudyID
##
##         post.mean l-95% CI u-95% CI eff.samp
## StudyID    0.6995   0.3889    1.059     2000
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent    0.5676 9.041e-07    1.894     1867
##
##                ~Cont_Country
##
##              post.mean  l-95% CI u-95% CI eff.samp
## Cont_Country   0.07227 3.095e-09   0.2929     1872
##
```

```
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units     1.403     1.23    1.561     2000
##
##  Location effects: logratio ~ broadTreatment
##
##                             post.mean l-95% CI u-95% CI eff.samp  pMCMC
## (Intercept)                   2.06237  1.31136  2.80222     2000  0.004 **
## broadTreatmentCoumaphos      -1.13913 -1.64999 -0.63639     2418 <5e-04 ***
## broadTreatmentFormic_acid    -0.33822 -0.81579  0.08118     2255  0.142
## broadTreatmentOxalic_acid    -0.37834 -0.93635  0.13686     2000  0.162
## broadTreatmentPyrethroid     -0.32301 -0.72300  0.04070     2000  0.110
## broadTreatmentThymol         -0.09454 -0.53021  0.30171     2000  0.652
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Dosage dependance

Taking from the Full_comb_data, which has the signs flipped so any positive number is a positive effect for bees (i.e. decreased bee mortality is now a positive number).

```
Full_dosage <- data.frame(logratio = Full_comb_data$logratio,
                          StudyID = Full_comb_data$StudyID,
                          SubCa2Treat = Full_comb_data$SubCategory2Treatment,
                          Dosage_level = Full_comb_data$Dosage_level,
                          Continent = Full_comb_data$Continent,
                          Country = Full_comb_data$Country,
                          Cont_Country = Full_comb_data$Cont_Country)

Full_dosage <- na.omit(Full_dosage)
```

Ordinal dosage analysis.

```
mod_dos <- MCMCglmm(logratio ~ Dosage_level,
                                rcov=~units,
                                random =~StudyID
                                         + Continent
                                         + Cont_Country,
                                family ="gaussian",
                                data = Full_dosage,
                                nitt = nitt,
                                thin = thining,
                                burnin = burnin,
                                prior = prior_d,
                                verbose = FALSE
                            )

summary(mod_dos)
```

```
##
```

```
##  Iterations = 10001:109951
##  Thinning interval  = 50
##  Sample size  = 2000
##
##  DIC: 854.6991
##
##  G-structure:  ~StudyID
##
##         post.mean  l-95% CI u-95% CI eff.samp
## StudyID    0.3374 1.748e-06   0.8278     2153
##
##                ~Continent
##
##           post.mean  l-95% CI u-95% CI eff.samp
## Continent    0.6803 1.266e-07    2.404     2000
##
##                ~Cont_Country
##
##              post.mean  l-95% CI u-95% CI eff.samp
## Cont_Country    0.4024 7.815e-07    1.287     1823
##
##  R-structure:  ~units
##
##       post.mean l-95% CI u-95% CI eff.samp
## units     1.654    1.373    1.994     2000
##
##  Location effects: logratio ~ Dosage_level
##
##                    post.mean l-95% CI u-95% CI eff.samp pMCMC
## (Intercept)           1.1481   0.2658   2.0058     2214 0.015 *
## Dosage_levelLOW      -0.1994  -0.5612   0.1515     1739 0.282
## Dosage_levelMEDIUM    0.1332  -0.3629   0.6331     1875 0.611
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```