viability_selection_mr_lifespan

Contents

Libraries 1

Download data and do preliminary parsing 3

Calculating metabolic rates 3

Day 11 Cohort 3

- Creating the time sequence . . . 3
- Calculating start and end time for total amount of hours of CO2 production . . . 4
- Metabolic Rate Calculation . . . 5
- Controlling for blank controls . . . 5
- Figure of Metabolic Rate between Hosts . . . 5
- Calculate eclosion days . . . 7
- Figures of eclosion and metabolic rate data . . . 7
- Test differences in eclosion timing between host and experiment (ANOVA) . . . 11
- Explore relationship between eclosion days and MR . . . 12
- Calculating lifespan . . . 16
- Figures of lifespan between hosts . . . 16
- Figures Associating MR with lifespan . . . 18

Day 15 Cohort 20

- Repeat above, but with day 15, starting with time sequence . . . 20
- Calculating start and end time for total amount of hours of CO2 production . . . 21
- Metabolic Rate Calculation . . . 21
- Controlling for Blanks . . . 21
- Calculate Eclosions Dates . . . 22
- Test differences in eclosion timing between host and experiment for Day 15 (ANOVA) . . . 22
- Figures looking at MR between hosts . . . 23
- Explore relationship between eclosion days and MR . . . 24
- Figures associating MR with lifespan . . . 28
- Double Checking Eclosion for Pupal Deaths . . . 30
- Merging pupal death data sheet lifespans . . . 30
- Negative binomial regression reanalyzed with pupal death data . . . 31
- Kaplan-Meier Survival Estimate . . . 40
- Cox Regression . . . 42

Conclusions 45

Session Info 45

Libraries

```r
library(dplyr) #data parsing library
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(data.table) #data parsing library mainly for reading and writing out
```

```
##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```r
library(ggplot2) #data visualization library
library(lubridate) #package to deal with time variables
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:data.table':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday,
##     week, yday, year

## The following object is masked from 'package:base':
##
##     date
```

```r
library(curl) #incorporate URLs
library(DiagrammeR) #figure aesthetics
library(MASS) #Functions and datasets to support
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```r
library(BBmisc) #Miscellaneous functions
```

```
##
## Attaching package: 'BBmisc'

## The following objects are masked from 'package:dplyr':
##
##     coalesce, collapse

## The following object is masked from 'package:base':
##
##     isFALSE
```

```r
library(survival) #survival data analysis
library(survminer) #survival data analysis
```

```
## Loading required package: ggpubr

## Loading required package: magrittr
```

```
library(gridExtra) #figure aesthetics
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

Download data and do preliminary parsing

Data are publicly available via github

```
data<- fread("https://raw.githubusercontent.com/adnguyen/Circadian_rhythm_runs_seasonal_timing/master/Da

#look at summary of data
#glimpse(data)
```

Calculating metabolic rates

Metabolic Rate = CO2 production/ hours Mass Specific Metabolic Rate = CO2 production/hrs per mass **Problem** with purge_time_1: only start and stop times except for apple cohort 1 **Solution** create time sequence for each cohort from start to end and merge into master data sheet

Day 11 Cohort

Creating the time sequence

```
#glimpse(data$purge_time_1)
#hm(data$purge_time_1)
data$day10purge <- lubridate::hour(hm(data$purge_time_1))+lubridate::minute(hm(data$purge_time_1))/60
```

```
## Warning in .parse_hms(..., order = "HM", quiet = quiet): Some strings
## failed to parse, or all strings are NAs

## Warning in .parse_hms(..., order = "HM", quiet = quiet): Some strings
## failed to parse, or all strings are NAs
```

```
#Obtaining start and end (min and max) of purges and smaple size for each host, cohort day, and tape
param <- data%>%
  group_by(cohort_day, tape)%>%
  summarise(max=max(day10purge, na.rm=TRUE), min=min(day10purge, na.rm=TRUE), n=length(cohort_day))

#goal: for this section, we want a sequence of times for day 10 purge
data2 <- data%>%
  group_by(cohort_day, tape)%>%
  mutate(.,day10purge.trans=seq(from = min(day10purge, na.rm=TRUE), to = max(day10purge, na.rm=TRUE), le
#glimpse(data2)

#cohort 2&3 for apple have the right sequence of purge times so they(day10purge) need to be replaced in
```

```
  #extract number of rows we want to replace
data2[1:500,]%>%
  filter(cohort_day < 4)
```

```
## # A tibble: 211 x 46
## # Groups:   cohort_day, tape [4]
##    Ind_ID tape  Site_name mass_day10 purge_time_1 purge1 collection_date
##     <int> <chr> <chr>          <dbl> <chr>         <dbl> <chr>
##  1      1 blue  OG              6.94 13:38            NA 2017-08-21
##  2      2 blue  Ferris         11.2  13:39            NA 2017-08-21
##  3      3 blue  Ferris          6.72 13:39            NA 2017-08-21
##  4      4 blue  Ferris         10.7  13:40            NA 2017-08-21
##  5      5 blue  OG              3.85 13:41            NA 2017-08-21
##  6      6 blue  OG              7.58 13:37            NA 2017-08-21
##  7      7 blue  OG              6.41 13:35            NA 2017-08-21
##  8      8 blue  OG              9.36 13:57            NA 2017-08-21
##  9      9 blue  Ferris          7.98 13:41            NA 2017-08-21
## 10     10 blue  OG              4.89 13:36            NA 2017-08-21
## # ... with 201 more rows, and 39 more variables: day10 <chr>,
## #   cohort_date <chr>, cohort_day <int>, Host <chr>, `Li-cor_1` <int>,
## #   resp_time_1 <chr>, resp_day11 <dbl>, mass_day14 <dbl>,
## #   purge_time_2 <chr>, resp_time_2 <chr>, resp_day15 <dbl>,
## #   Li_cor2 <int>, treatment_day15 <chr>, exit_fridge_date <chr>,
## #   Eclosion_reference_date <chr>, notes <chr>, Resp_code <int>,
## #   treatment <chr>, uniqueID <chr>, eclosion_date <chr>,
## #   eclosion_days <int>, well_id <chr>, organism <chr>,
## #   Trikinetics_position <int>, Trik_monitor <int>,
## #   Trikinetics_entry_LD_time <chr>, Trikinetic_exit_date <chr>,
## #   Trikinetics_exit_LD_time <chr>, notes_2 <chr>,
## #   Free_run_trik_monitor <int>, Free_run_trik_position <int>,
## #   Free_run_entry_date <chr>, Free_run_entry_time <chr>,
## #   Free_run_exit_date <chr>, Free_run_exit_time <chr>, notes_3 <chr>,
## #   Adult_death_date <chr>, day10purge <dbl>, day10purge.trans <dbl>
```

```
data2[1:211,46]<- data2[1:211,45]
#data2[1:211,46]
```

Calculating start and end time for total amount of hours of CO2 production

```
#glimpse(data$resp_time_1)
#hms(data$resp_time_1)
data2$day10resp <- lubridate::hour(hms(data$resp_time_1))+lubridate::minute(hms(data$resp_time_1))/60
```

```
## Warning in .parse_hms(..., order = "HMS", quiet = quiet): Some strings
## failed to parse, or all strings are NAs

## Warning in .parse_hms(..., order = "HMS", quiet = quiet): Some strings
## failed to parse, or all strings are NAs
```

```
#reformatting to obtain positive values
data2$total_time_day10 <- (24 - data2$day10purge.trans) + data2$day10resp
```

Metabolic Rate Calculation

```
#getting denominator for mass-specific mr
denom2 <- data2$total_time_day10 * data$mass_day10

#Metabolic Rate for day 11
data2$MR11<- data$resp_day11/(data2$total_time_day10)

#Mass specific Metabolic Rate for day 11
data2$msMR11<- data$resp_day11/(denom2)
```

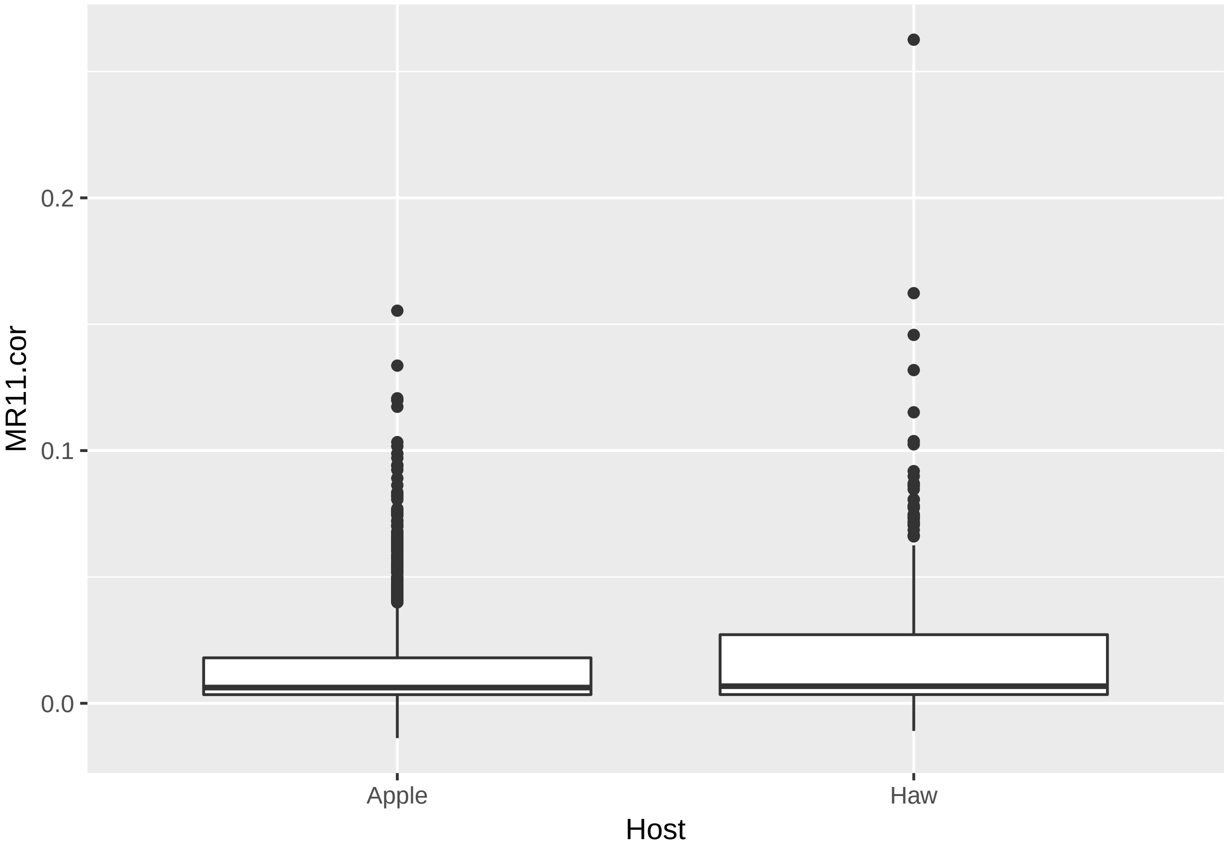
Controlling for blank controls

```
#Assign the blanks to an object
data3 <- data2%>%
  group_by(cohort_day, tape)%>%
  filter(Site_name=="Blank")%>%
  summarise(mean.blank=mean(MR11,na.rm=TRUE))

#check columns
#glimpse(data3)
#data3$mean.blank

#merge data3 and data2 by cohort day and tape and create a mean blanks column
data4 <- inner_join(data2, data3, by=c("cohort_day", "tape"))
#data4$mean.blank

#do some corrections
data5 <- data4%>%
  mutate(MR11.cor = MR11 - mean.blank, msMR11.cor = msMR11 - mean.blank)
#glimpse(data5)

data5.neg <- data5%>%
  filter(MR11.cor<0)
```

Figure of Metabolic Rate between Hosts

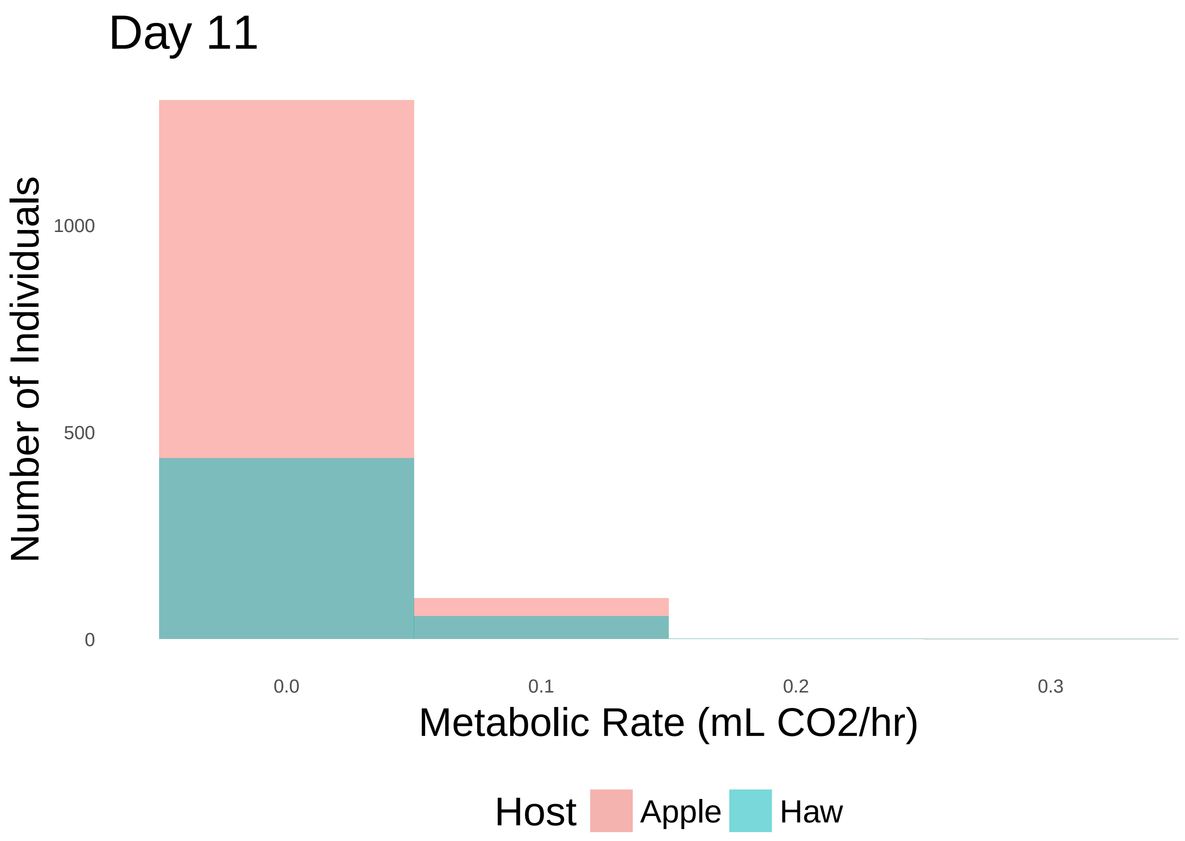
```
#Boxplot
ggplot(data5, aes(x=Host, y=MR11.cor))+geom_boxplot()
```

```
## Warning: Removed 13 rows containing non-finite values (stat_boxplot).
```



```
#Histogram
setup<-theme_bw()+theme(axis.text.x=element_blank(),
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.text.y=element_blank(),
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s

ggplot(data5, aes(x=MR11.cor, fill=Host))+geom_histogram(position = "identity", alpha=.5, binwidth = .1
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Warning: Removed 13 rows containing non-finite values (stat_bin).
```



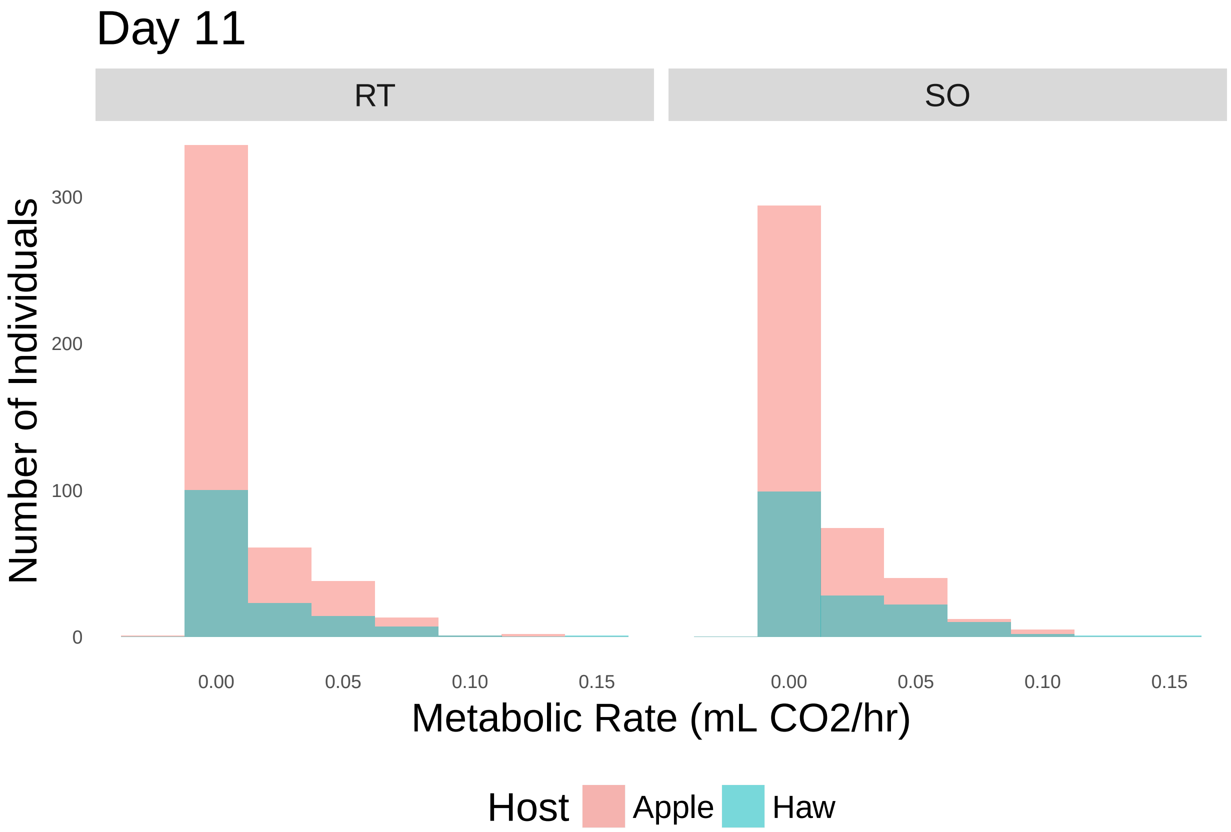
Calculate eclosion days

```
data5$neweclosions<-difftime(as.Date(data5$eclosion_date), as.Date(data5$Eclosion_reference_date),units

data5$neweclosions <- as.numeric(data5$neweclosions)
```

Figures of eclosion and metabolic rate data

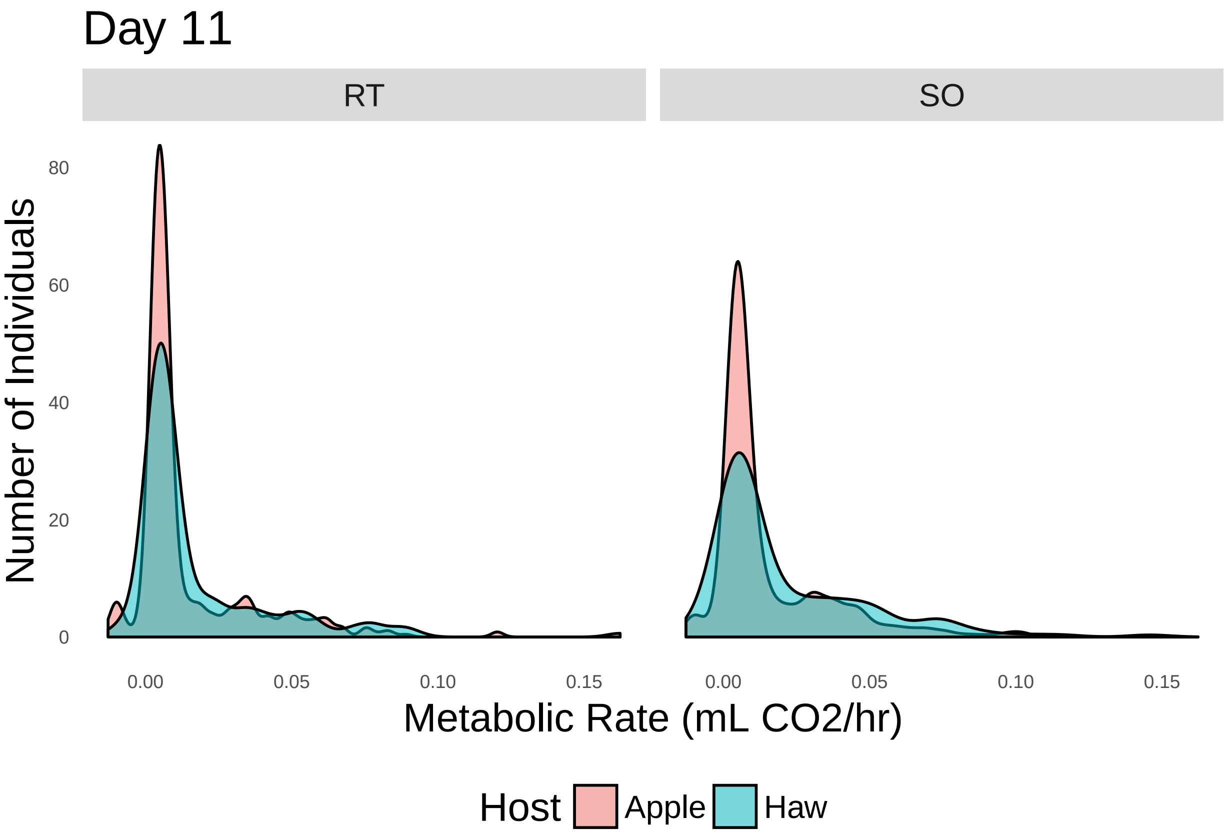
```
data5.treatsub<-data5%>%
  filter(treatment!="GC"&treatment!="")

data5.treatsub$neweclosions<-as.numeric(data5.treatsub$neweclosions)

#Histogram comparison between MR of RT & SO
ggplot(data5.treatsub, aes(x=MR11.cor, fill=Host))+geom_histogram(position = "identity", alpha=.5, binw
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Warning: Removed 4 rows containing non-finite values (stat_bin).
```



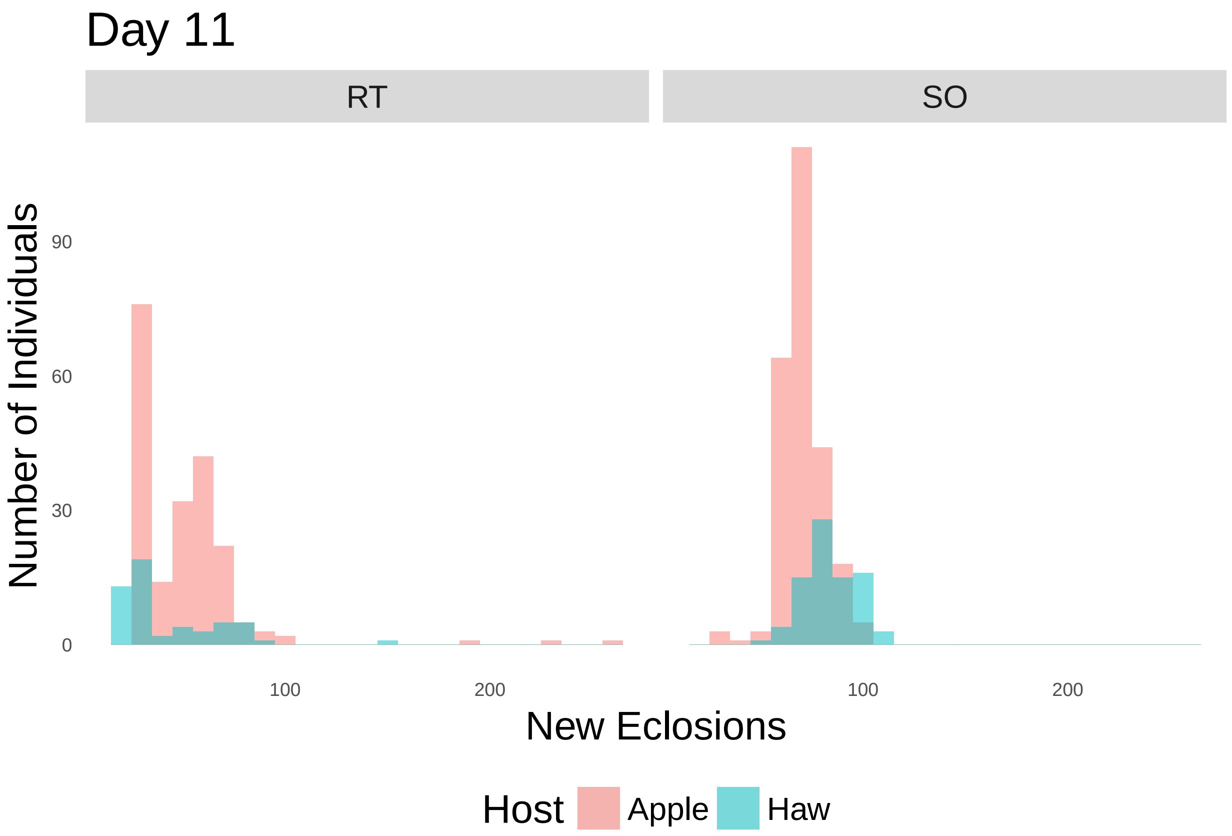
```
#Density comparison between MR of RT & SO
ggplot(data5.treatsub, aes(x=MR11.cor, fill=Host))+geom_density(position = "identity", alpha=.5)+facet_
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Warning: Removed 4 rows containing non-finite values (stat_density).
```



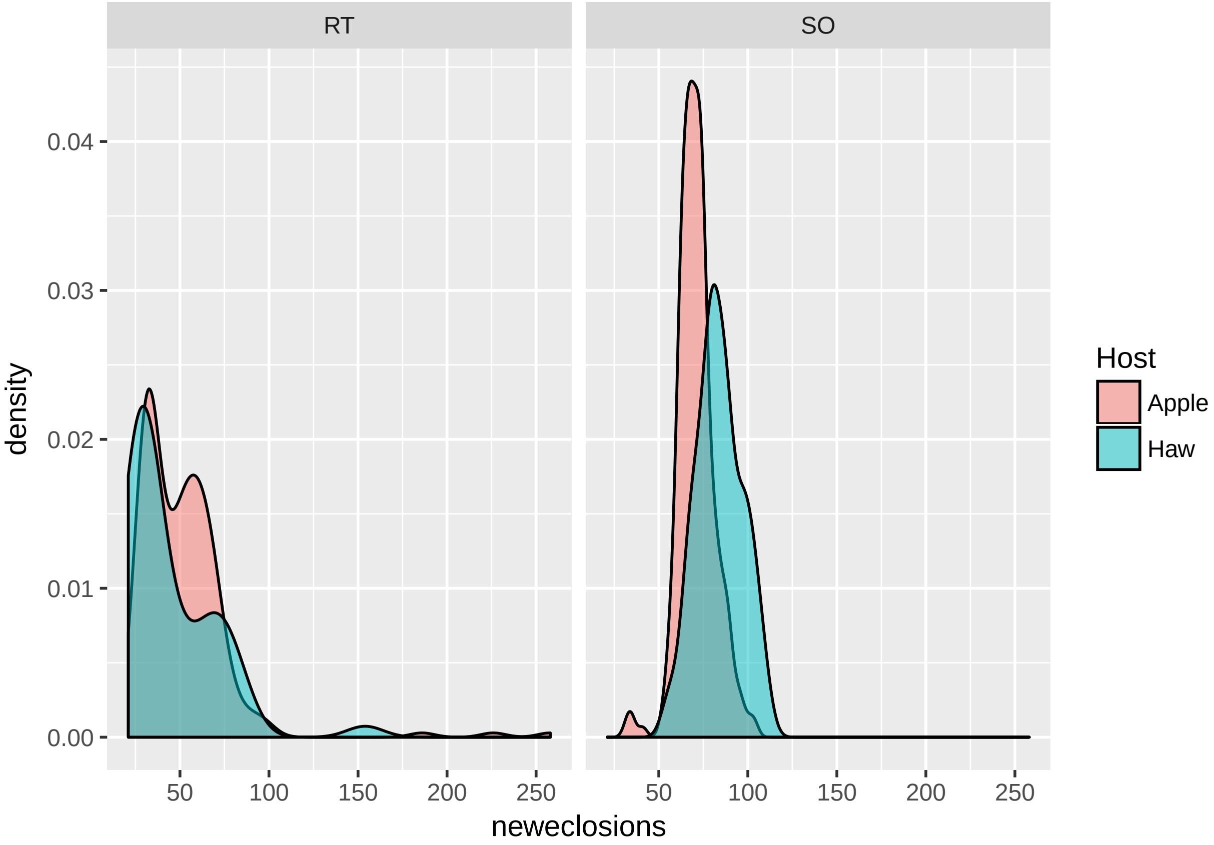
```
#Histogram comparison between eclosion date of RT & SO
ggplot(data5.treatsub, aes(x=neweclosions, fill=Host))+geom_histogram(position = "identity", alpha=.5, 
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Warning: Removed 606 rows containing non-finite values (stat_bin).
```



```
#Density comparison between eclosion date of RT & SO
ggplot(data5.treatsub, aes(x=neweclosions, fill=Host))+geom_density(position = "identity", alpha=.5)+fa
```

```
## Warning: Removed 606 rows containing non-finite values (stat_density).
```



Test differences in eclosion timing between host and experiment (ANOVA)

Using ANOVA which tests differences between two or more means * The null hypothesis would be that the two means are equal * Significance would indicate two means are not equal One Way ANOVA compares two means from two independent groups, in this case eclosion date and host (specific to SO and RT)

Two Way ANOVA compares means of two independent variables affecting one dependent variable, in this case the effect of Host and treatment (interacting) on eclosion

```
#Focus only on RT and SO treatments
data5.treatsub<-data5%>%
  filter(treatment!="GC"&treatment!="")

#Two Way ANOVA
mod1<- aov(neweclosions ~ Host*treatment, data=data5.treatsub)
summary(mod1)
```

```
##                  Df Sum Sq Mean Sq F value   Pr(>F)    
## Host              1   3685    3685   9.705  0.00193 ** 
## treatment         1  82798   82798 218.056  < 2e-16 ***
## Host:treatment    1   9903    9903  26.080 4.45e-07 ***
## Residuals       579 219851     380                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 606 observations deleted due to missingness
```

```
#narrow down to SO & RT
data5.SO<-data5.treatsub%>%
  filter(treatment=="SO")

data5.RT<-data5.treatsub%>%
  filter(treatment=="RT")

#One Way ANOVA for RT & SO
mod2RT<-aov(neweclosions~Host, data=data5.RT)
summary(mod2RT)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## Host          1   2143  2142.9    2.97  0.086 .
## Residuals   250 180353   721.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 346 observations deleted due to missingness
```

```
mod2SO<-aov(neweclosions~Host, data=data5.SO)
summary(mod2SO)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## Host          1  10064   10064   83.83 <2e-16 ***
## Residuals   329  39499     120
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 260 observations deleted due to missingness
```

```
#how balanced it is
data5.RT%>%
  group_by(Host)%>%
  summarise(n=length(Host))
```

```
## # A tibble: 2 x 2
##   Host      n
##   <chr> <int>
## 1 Apple   452
## 2 Haw     146
```

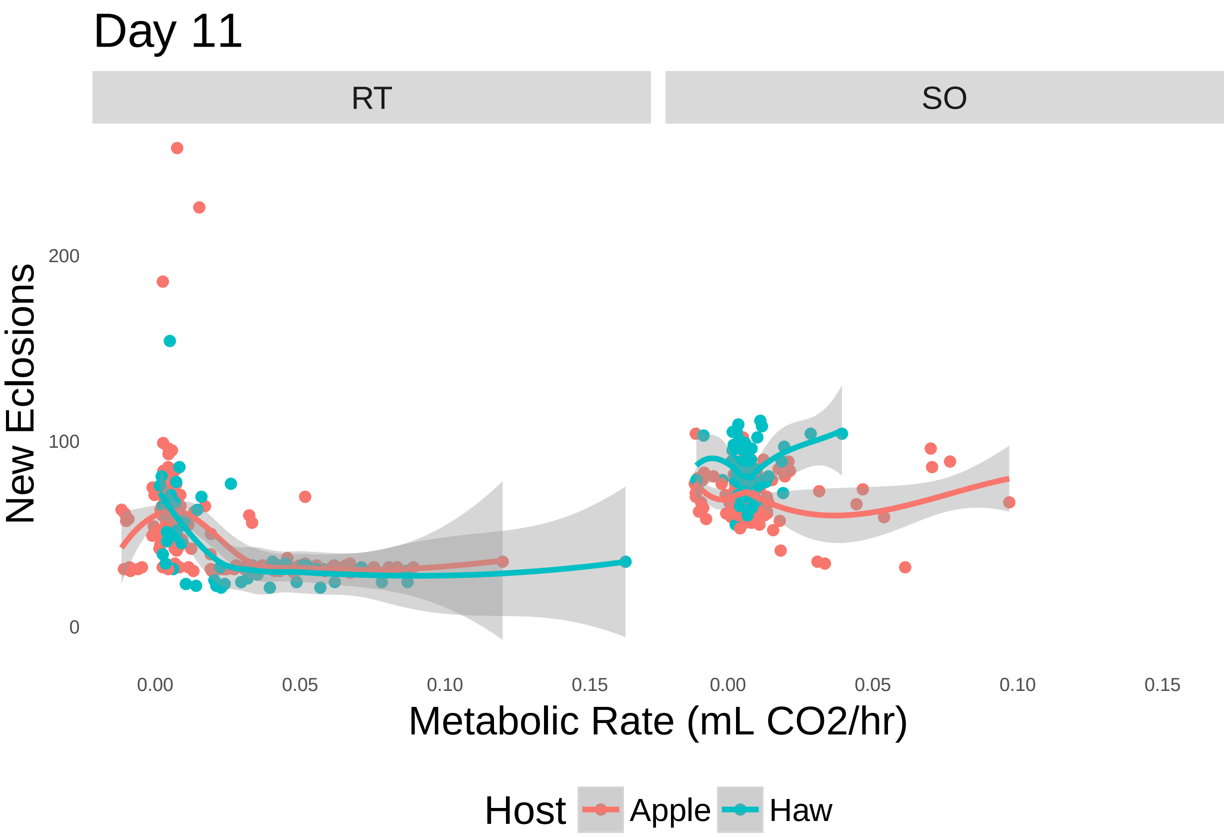
Both the one way and two way ANOVA tests showed significance between the two means (meaning they are not equal). The two way ANOVA suggests that there is a difference in number of days to eclosion based on treatment type - favorable conditions vs. simulated overwintering. Therefore, a one way ANOVA was done for each treatment to test whether there is any significance between host type and days to eclosion. The one way ANOVA showed significance for both treatments; however, the simulated overwintering samples had a lower p value, which indicates more signigicance.

Explore relationship between eclosion days and MR

scatter plots

```
ggplot(data5.treatsub,aes(x=MR11.cor, y=neweclosions, colour=Host))+geom_point()+stat_smooth(method="lo
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

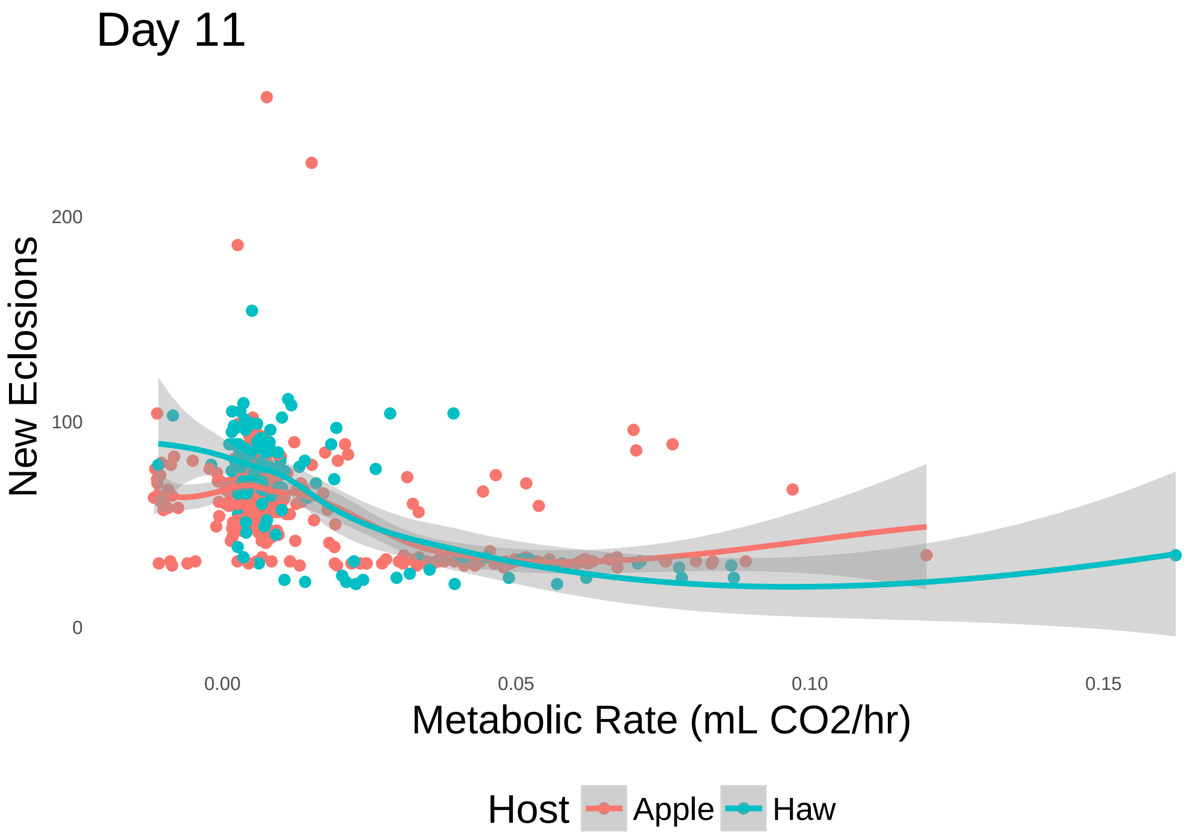
```
## Warning: Removed 609 rows containing non-finite values (stat_smooth).
## Warning: Removed 609 rows containing missing values (geom_point).
```



```
#plotting mr11 on the x axis and new eclosions on the y axis; color points by host, fit curve to data
ggplot(data5.treatsub,aes(x=MR11.cor, y=neweclosions, colour=Host))+geom_point()+stat_smooth(method="lo
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

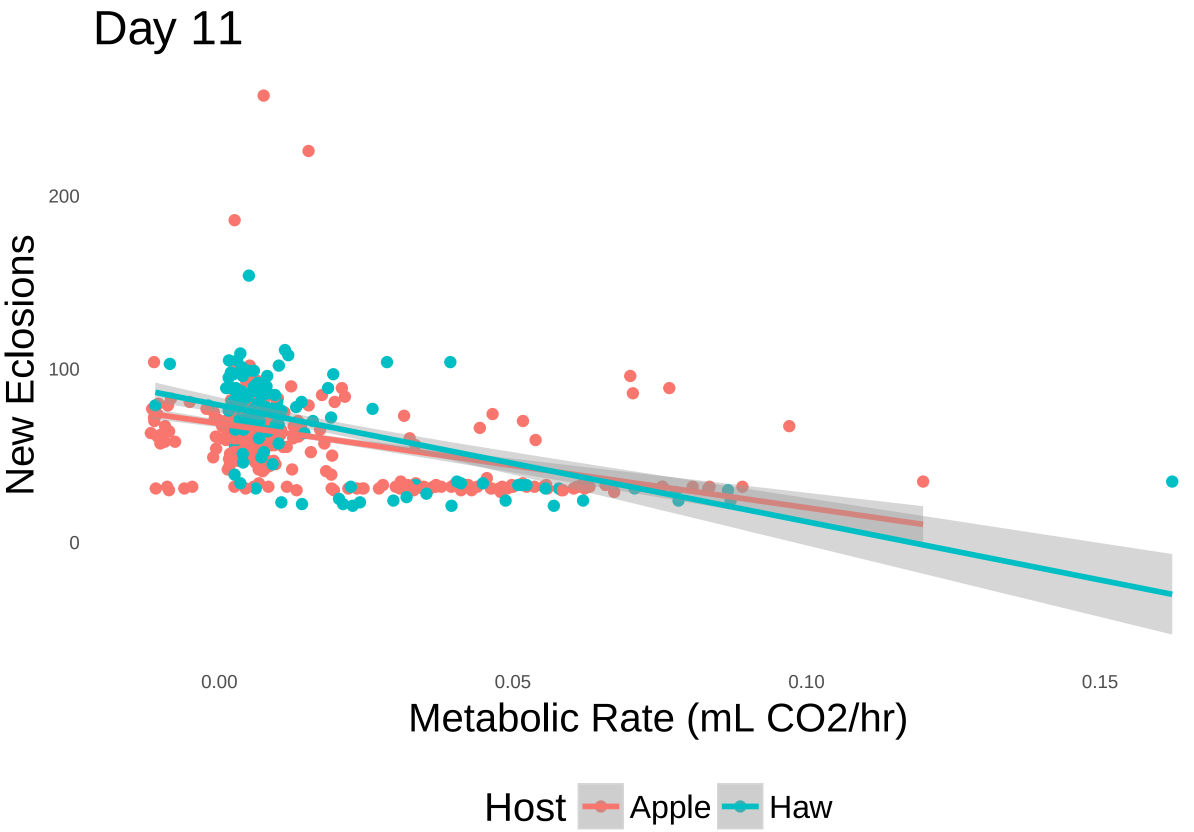
```
## Warning: Removed 609 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 609 rows containing missing values (geom_point).
```



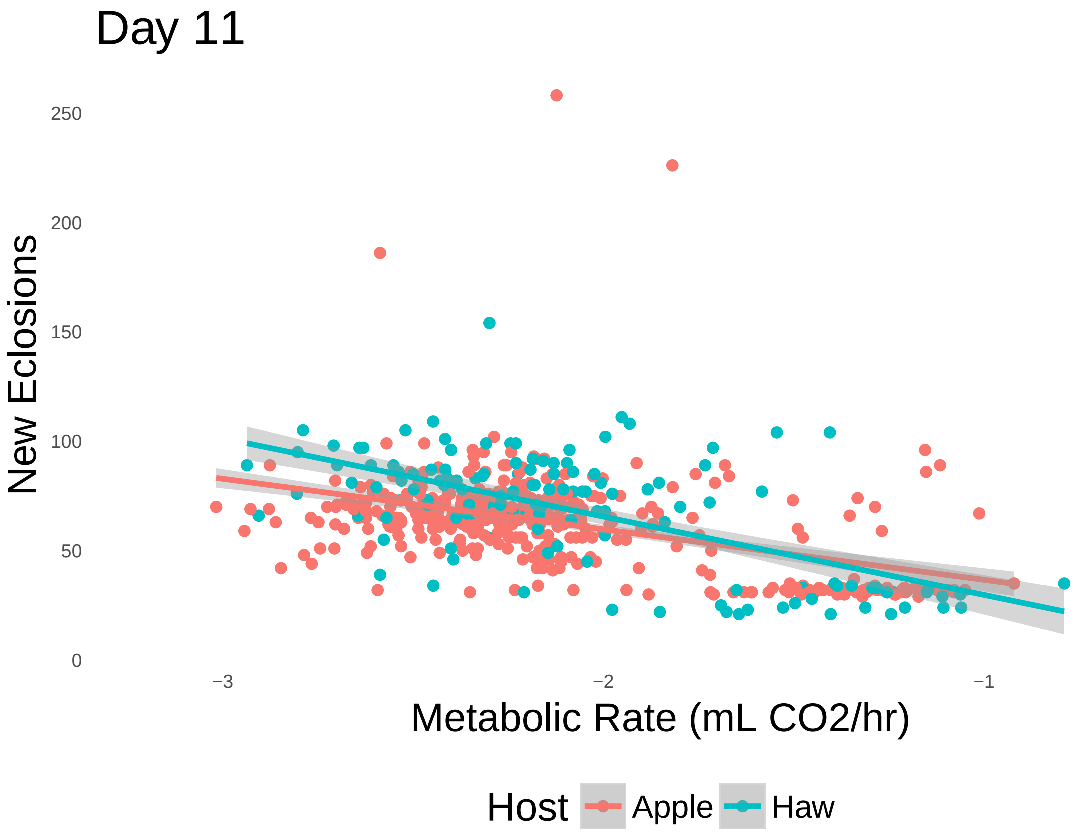
```
#same - fit linear curve to data
ggplot(data5.treatsub,aes(x=MR11.cor, y=neweclosions, colour=Host))+geom_point()+stat_smooth(method="lm
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Warning: Removed 609 rows containing non-finite values (stat_smooth).

## Warning: Removed 609 rows containing missing values (geom_point).
```



```
#could log transform mr to make it more linear
ggplot(data5.treatsub,aes(x=log10(MR11.cor), y=neweclosions, colour=Host))+geom_point()+stat_smooth(met
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Warning in FUN(X[[i]], ...): NaNs produced

## Warning in FUN(X[[i]], ...): NaNs produced

## Warning: Removed 644 rows containing non-finite values (stat_smooth).

## Warning: Removed 644 rows containing missing values (geom_point).
```



Calculating lifespan

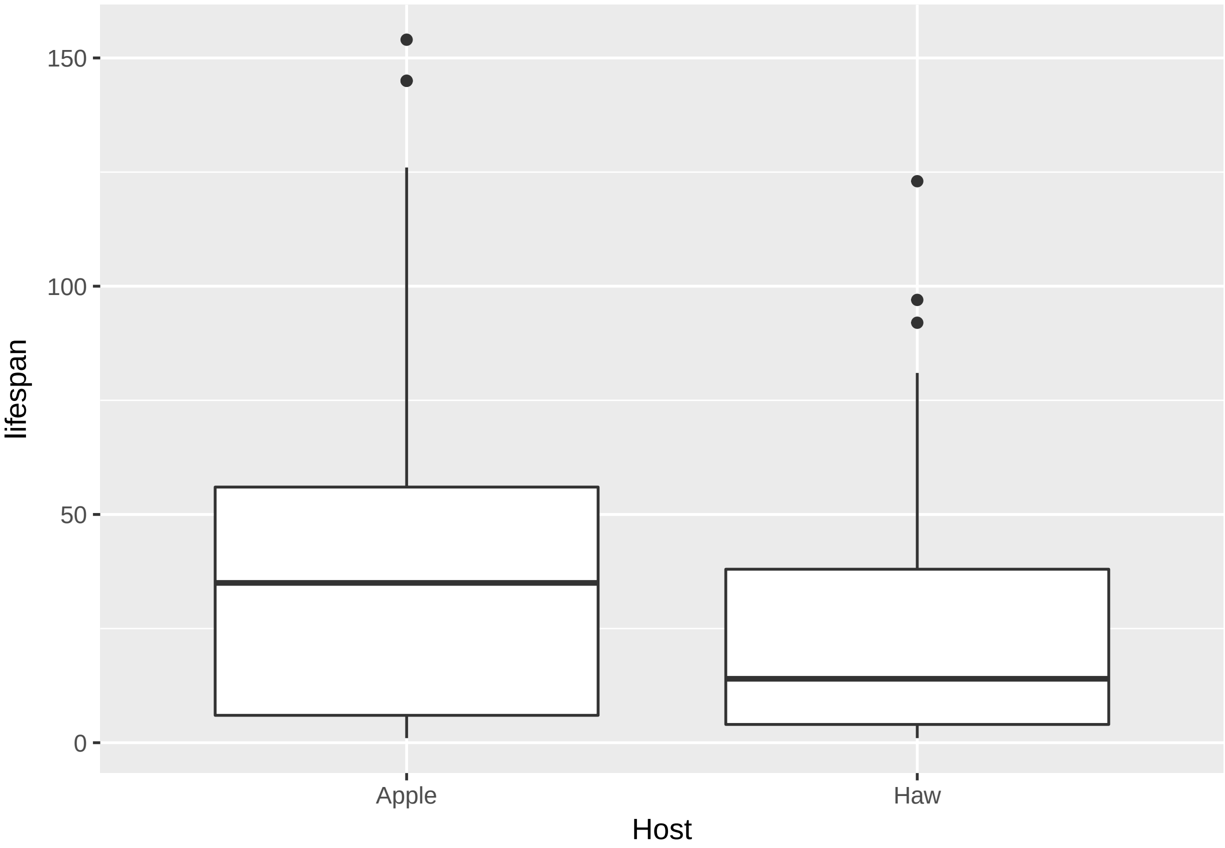
```
data5.treatsub$lifespan<-difftime(as.Date(data5.treatsub$Adult_death_date, na.rm=TRUE), as.Date(data5.t
```

Figures of lifespan between hosts

```
ggplot(data5.treatsub, aes(x=Host, y=lifespan))+geom_boxplot()
```

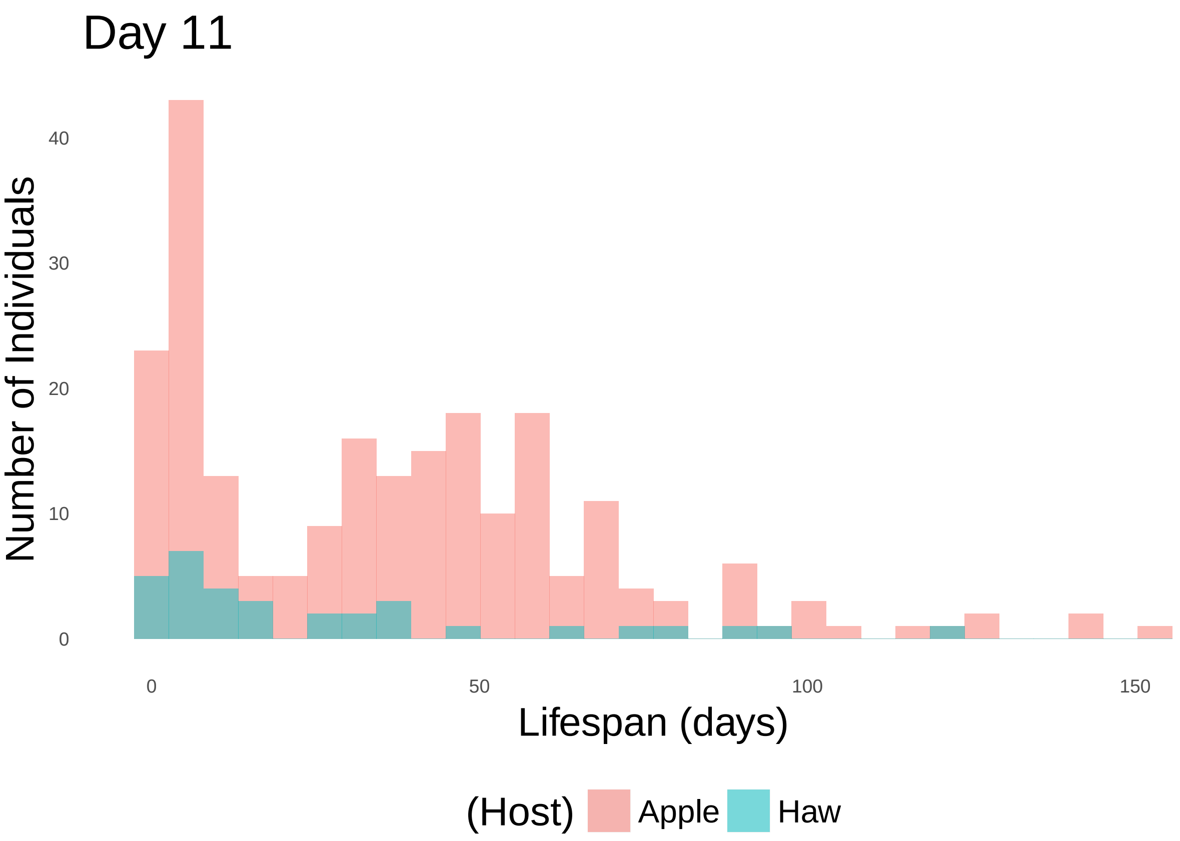
```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.
```

```
## Warning: Removed 927 rows containing non-finite values (stat_boxplot).
```



```
ggplot(data5.treatsub, aes(x=lifespan, fill=(Host)))+geom_histogram(position = "identity", alpha=.5) +la
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

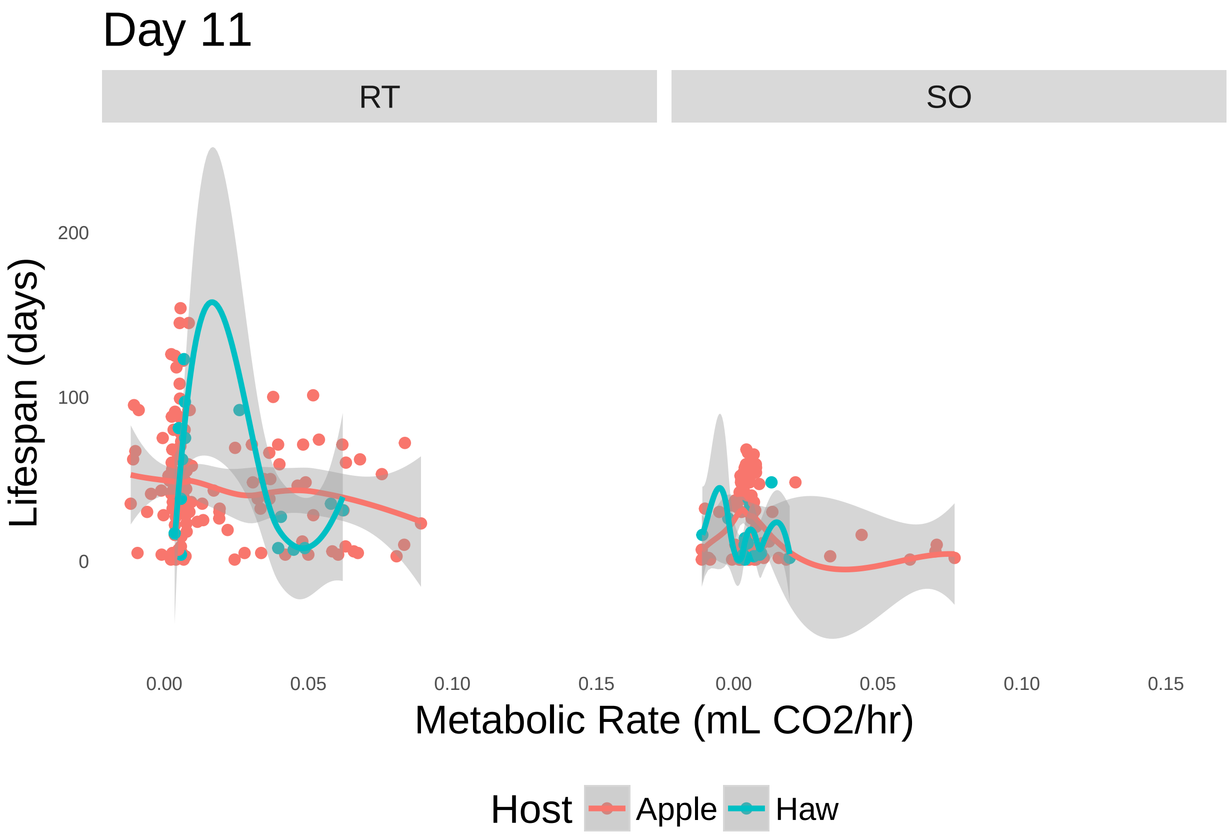
```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.

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

## Warning: Removed 927 rows containing non-finite values (stat_bin).
```



Figures Associating MR with lifespan

```
#Loess
ggplot(data5.treatsub,aes(x= MR11.cor, y=lifespan, colour=Host))+geom_point()+stat_smooth(method="loess
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.

## Warning: Removed 928 rows containing non-finite values (stat_smooth).

## Warning: Removed 928 rows containing missing values (geom_point).
```

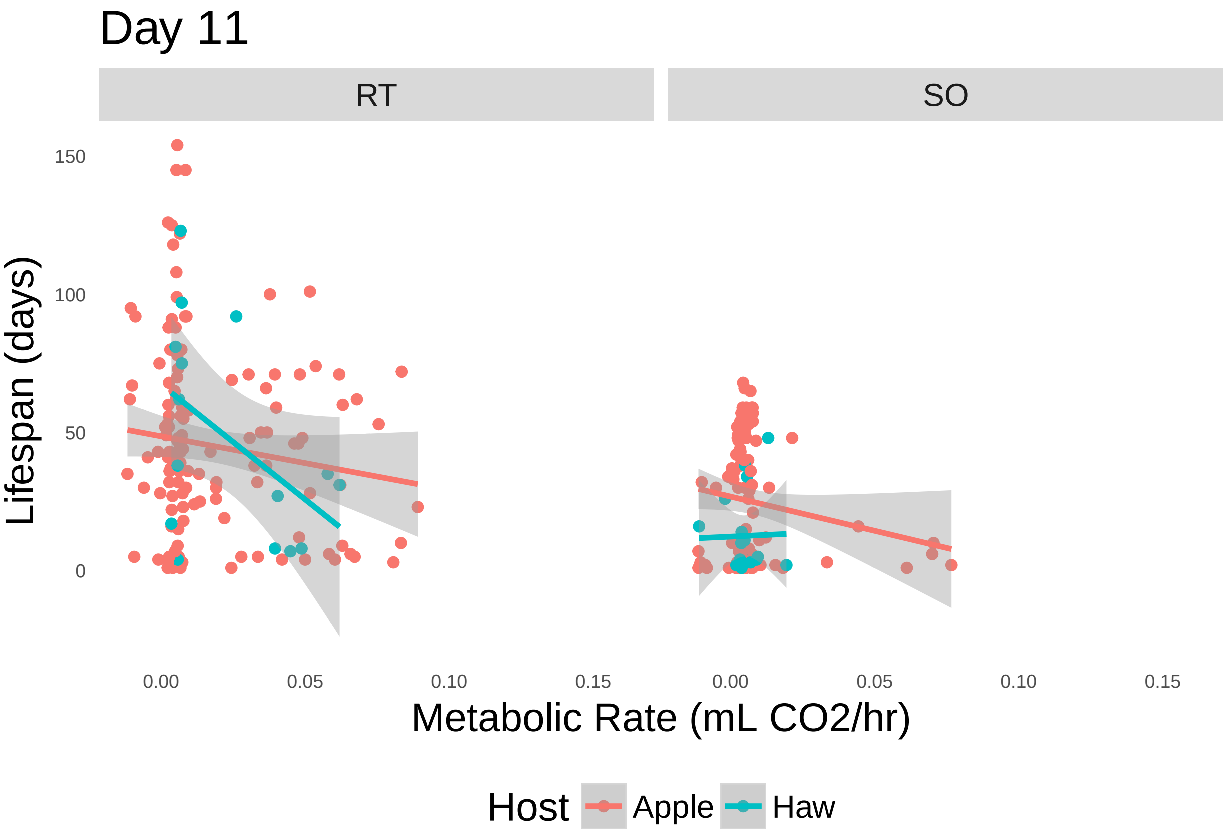


```
#Linear
ggplot(data5.treatsub,aes(x=MR11.cor, y=lifespan, colour=Host))+geom_point()+stat_smooth(method="lm") +
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.
```

```
## Warning: Removed 928 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 928 rows containing missing values (geom_point).
```



Day 15 Cohort

Repeat above, but with day 15, starting with time sequence

```
#glimpse(data$purge_time_2)
#hm(data$purge_time_2)
data$day15purge <- lubridate::hour(hm(data$purge_time_2))+lubridate::minute(hm(data$purge_time_2))/60
```

```
## Warning in .parse_hms(..., order = "HM", quiet = quiet): Some strings
## failed to parse, or all strings are NAs

## Warning in .parse_hms(..., order = "HM", quiet = quiet): Some strings
## failed to parse, or all strings are NAs
```

```
#Getting start and end (min and max) of purges and sample size for each host, cohort day, and tape
param <- data%>%
  group_by(cohort_day, tape)%>%
  summarise(max=max(day15purge, na.rm=TRUE), min=min(day10purge, na.rm=TRUE), n=length(cohort_day))

#goal: for this section, we want a sequence of times for day 15 purge
data2.15 <- data%>%
  group_by(cohort_day, tape)%>%
  mutate(.,day15purge.trans=seq(from = min(day15purge, na.rm=TRUE), to = max(day15purge, na.rm=TRUE), l
#glimpse(data2.15)
```

Calculating start and end time for total amount of hours of CO2 production

```
#glimpse(data$resp_time_2)
#hms(data$resp_time_2)
data2.15$day15resp <- lubridate::hour(hms(data$resp_time_2))+lubridate::minute(hms(data$resp_time_2))/60
```

```
## Warning in .parse_hms(..., order = "HMS", quiet = quiet): Some strings
## failed to parse, or all strings are NAs

## Warning in .parse_hms(..., order = "HMS", quiet = quiet): Some strings
## failed to parse, or all strings are NAs
```

```
#Obtaining total time in hours
data2.15$total_time_day15 <- (24 - data2.15$day15purge.trans) + data2.15$day15resp
```

Metabolic Rate Calculation

```
#getting denominator for mass-specific mr
denom<-data2.15$total_time_day15 * data$mass_day14

#Metabolic Rate
data2.15$MR15<- data$resp_day15/(data2.15$total_time_day15)

#Mass specfic metabolic rate
data2.15$msMR15<- data$resp_day15/(denom)
```

Controlling for Blanks

```
data3.15 <- data2.15%>%
  group_by(cohort_day, tape)%>%
  filter(Site_name=="Blank")%>%
  summarise(mean.blank2=mean(MR15,na.rm=TRUE))

#check columns
#glimpse(data3.15)
#data3.15$mean.blank2

#merge data3 and data2 by cohort day and tape
data4.15 <- inner_join(data2.15, data3.15, by=c("cohort_day", "tape"))
#data4.15$mean.blank2

#do some corrections
data5.15 <- data4.15%>%
  mutate(MR15.cor = MR15 - mean.blank2, msMR15.cor = msMR15 - mean.blank2)
#glimpse(data5.15)

#Filter out negatives
data5.15.neg <- data5.15%>%
  filter(MR15.cor<0)
```

Calculate Eclosions Dates

```
#Focus only on RT and SO treatments
data5.15.treatsub<-data5%>%
  filter(treatment!="GC"&treatment!="")

data5.15$neweclosions <- difftime(as.Date(data5.15$eclosion_date), as.Date(data5.15$Eclosion_reference_

data5.15.treatsub$neweclosions<-difftime(as.Date(data5.15.treatsub$eclosion_date), as.Date(data5.15.trea
```

Test differences in eclosion timing between host and experiment for Day 15 (ANOVA)

```
#Two Way ANOVA
mod1<- aov(neweclosions ~ Host*treatment, data=data5.15.treatsub)
summary(mod1)
```

```
##                  Df Sum Sq Mean Sq F value   Pr(>F)    
## Host              1   3685    3685   9.705  0.00193 ** 
## treatment         1  82798   82798 218.056  < 2e-16 ***
## Host:treatment    1   9903    9903  26.080 4.45e-07 ***
## Residuals       579 219851     380                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 606 observations deleted due to missingness
```

```
#narrow down to SO & RT
data5.15SO<-data5.15.treatsub%>%
  filter(treatment=="SO")

data5.15RT<-data5.15.treatsub%>%
  filter(treatment=="RT")

#One Way ANOVA for RT & SO
mod2RT15<-aov(neweclosions~Host, data=data5.15)
summary(mod2RT15)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)   
## Host          1   3685    3685    6.85 0.0091 **
## Residuals   581 312552     538                  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 1326 observations deleted due to missingness
```

```
mod2SO15<-aov(neweclosions~Host, data=data5.15)
summary(mod2SO15)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)   
## Host          1   3685    3685    6.85 0.0091 **
## Residuals   581 312552     538                  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 1326 observations deleted due to missingness
```

```
#how balanced is it?
data5.SO%>%
  group_by(Host)%>%
  summarise(n=length(Host))
```

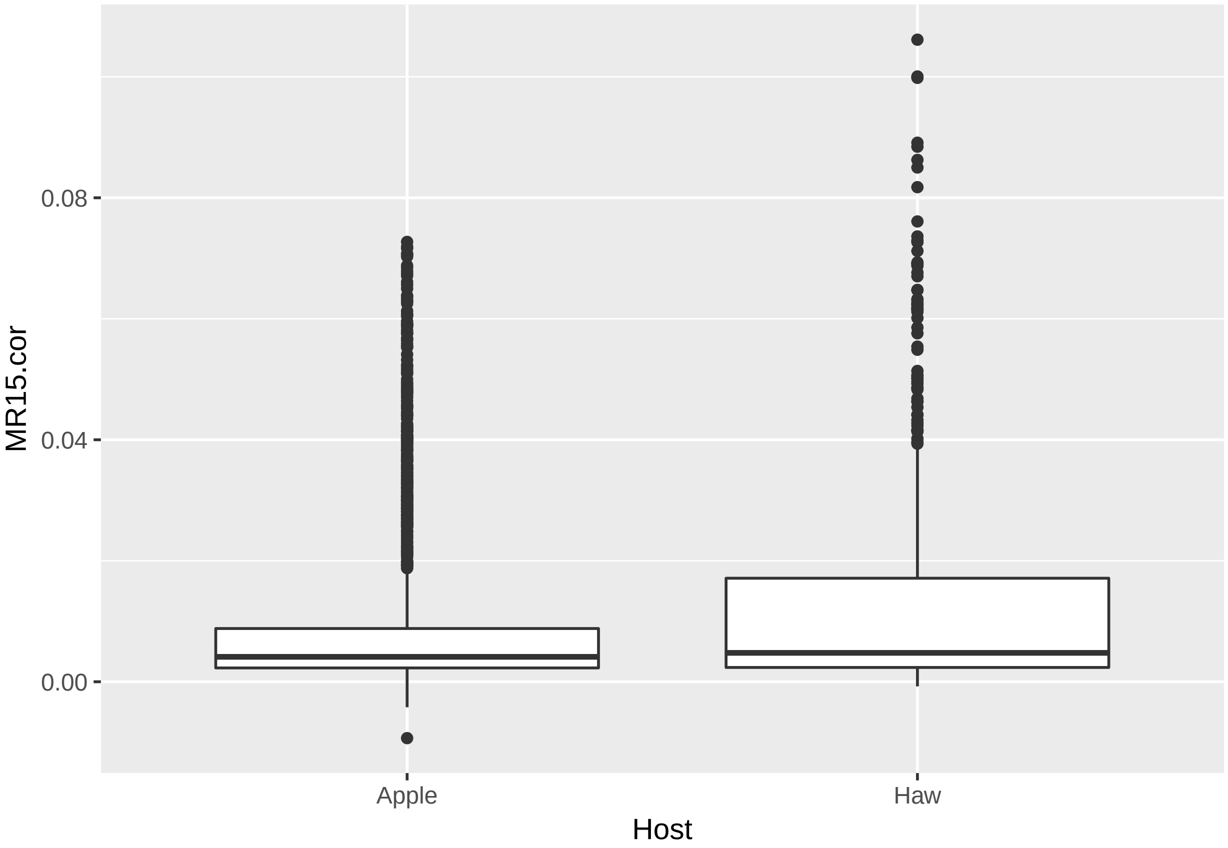
```
## # A tibble: 2 x 2
##   Host      n
##   <chr> <int>
## 1 Apple   428
## 2 Haw     163
```

This ANOVA test reflects the same results from the ANOVA done on Day 11.

Figures looking at MR between hosts

```
ggplot(data5.15, aes(x=Host, y=MR15.cor))+geom_boxplot()
```

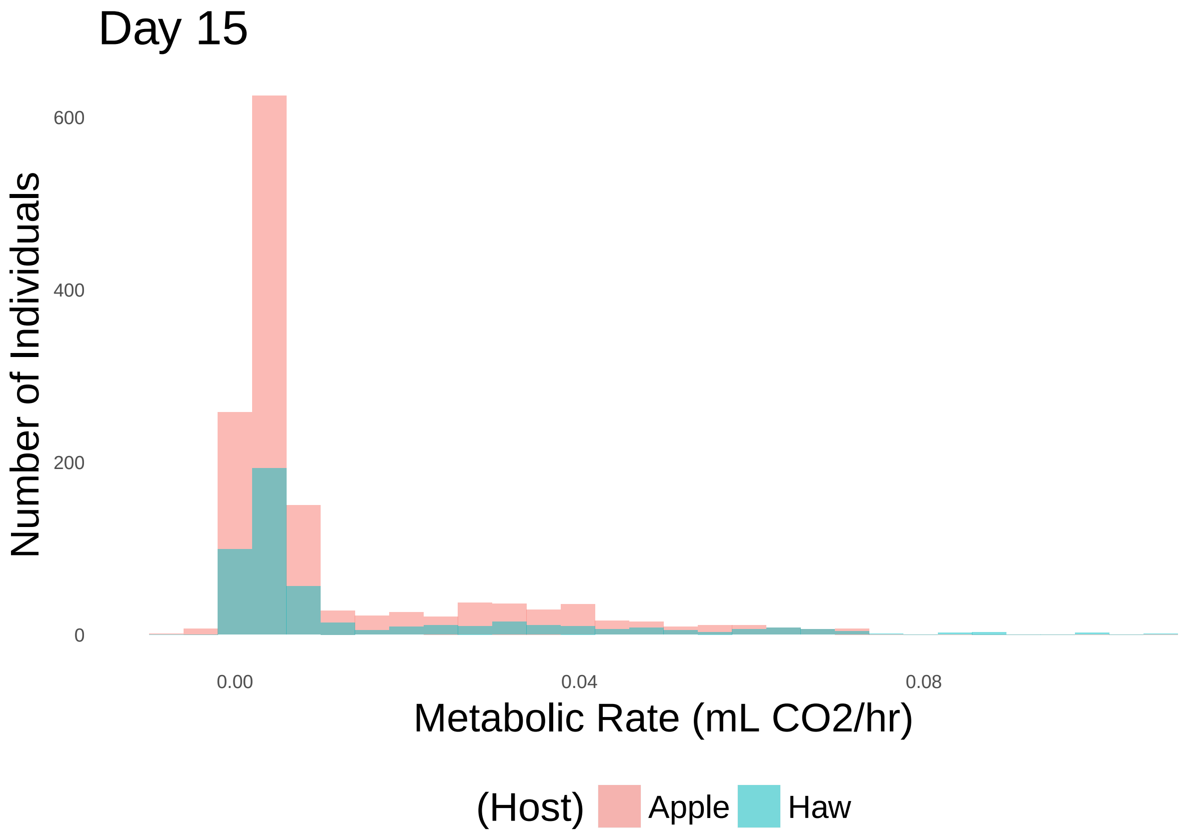
```
## Warning: Removed 63 rows containing non-finite values (stat_boxplot).
```



```
ggplot(data5.15, aes(x=MR15.cor, fill=(Host)))+geom_histogram(position = "identity", alpha=.5) +labs(x =
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

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

```
## Warning: Removed 63 rows containing non-finite values (stat_bin).
```

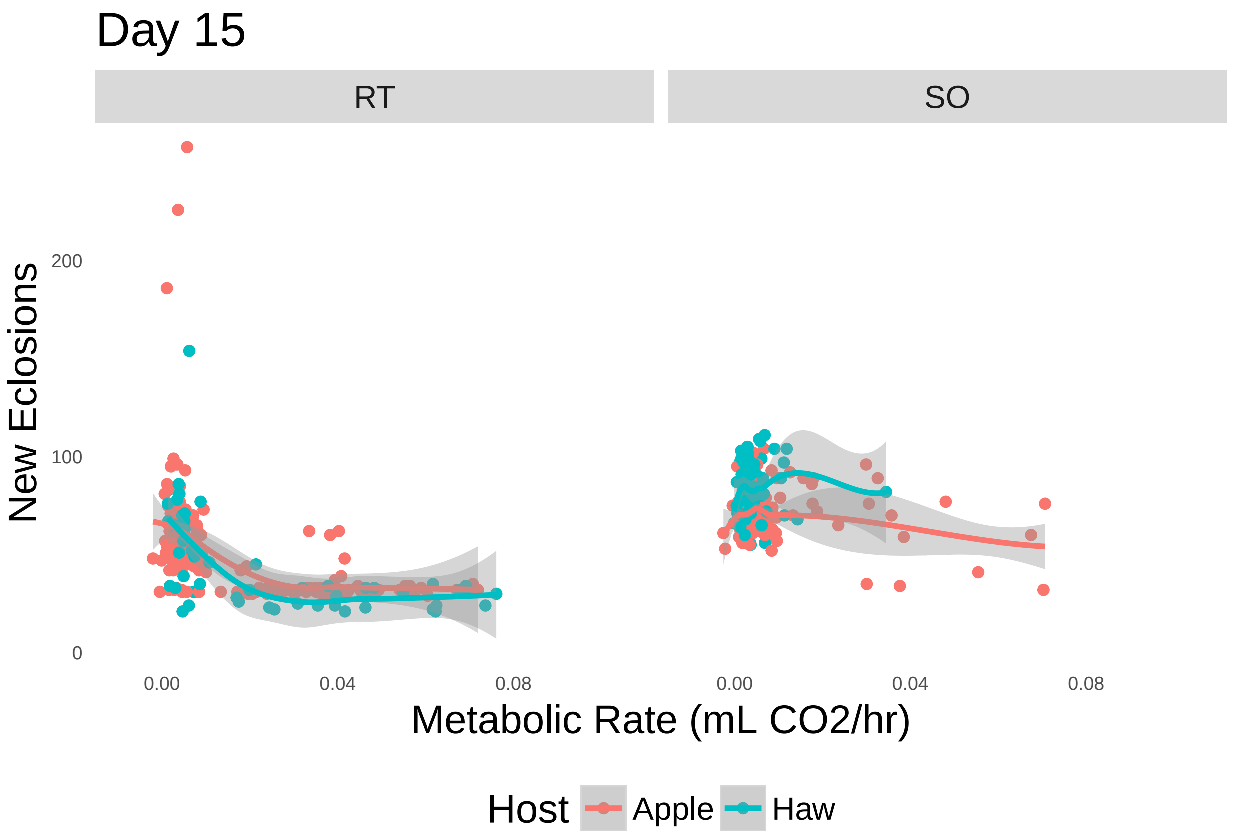


Explore relationship between eclosion days and MR

scatter plots

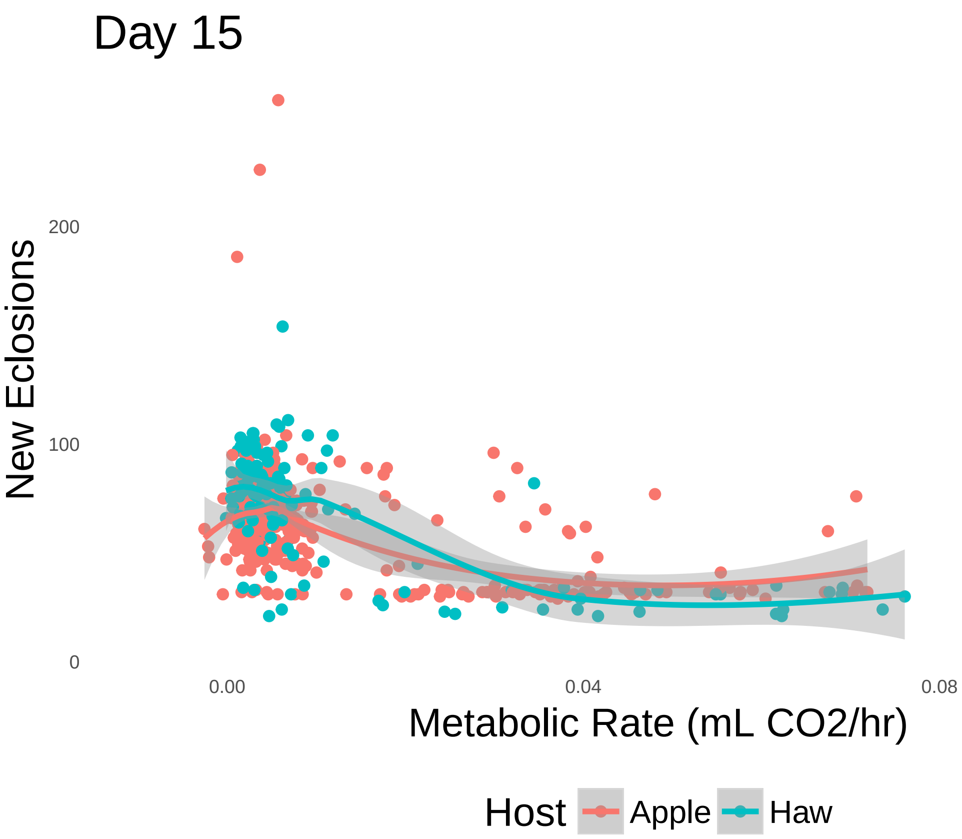
```
#Filter out GC and blank treatments
data5.15.treatsub<-data5.15%>%
  filter(treatment!="GC"&treatment!="")

ggplot(data5.15.treatsub, aes(x=MR15.cor, y=neweclosions, colour=Host))+geom_point()+stat_smooth(method=
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.

## Warning: Removed 607 rows containing non-finite values (stat_smooth).

## Warning: Removed 607 rows containing missing values (geom_point).
```



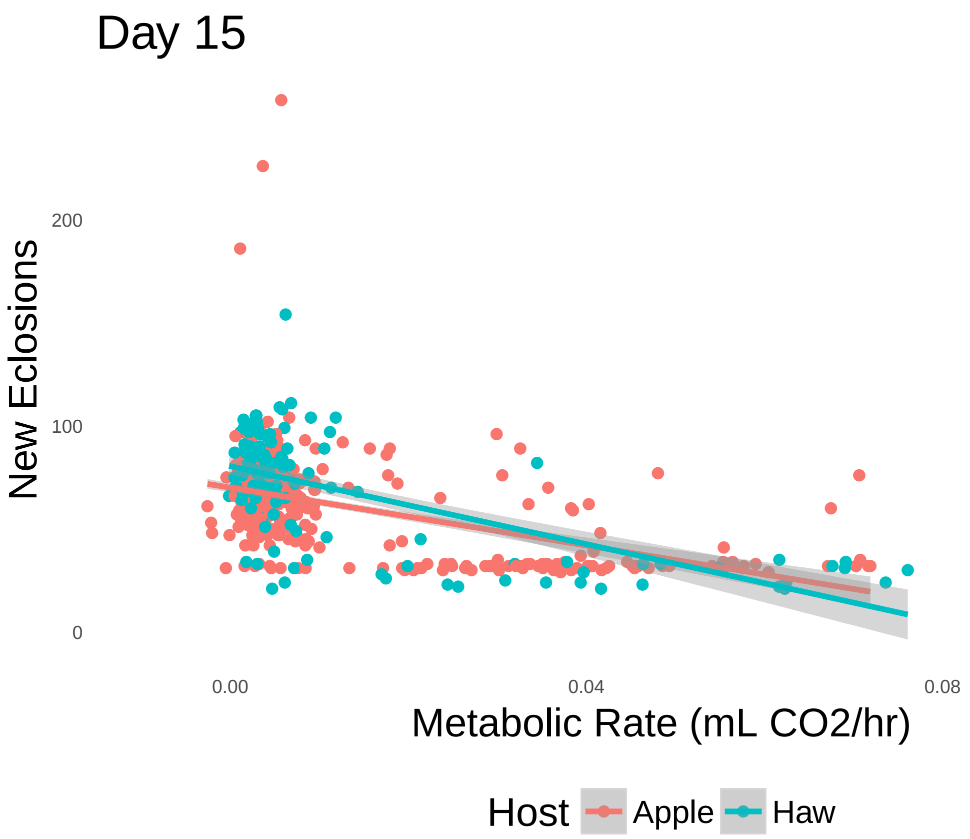
```
#plotting mr15 on the x axis and new eclosions on the y axis; color points by host, fit curve to data
ggplot(data5.15.treatsub,aes(x=MR15.cor, y=neweclosions, colour=Host))+geom_point()+stat_smooth(method=
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.
## Warning: Removed 607 rows containing non-finite values (stat_smooth).
## Warning: Removed 607 rows containing missing values (geom_point).
```



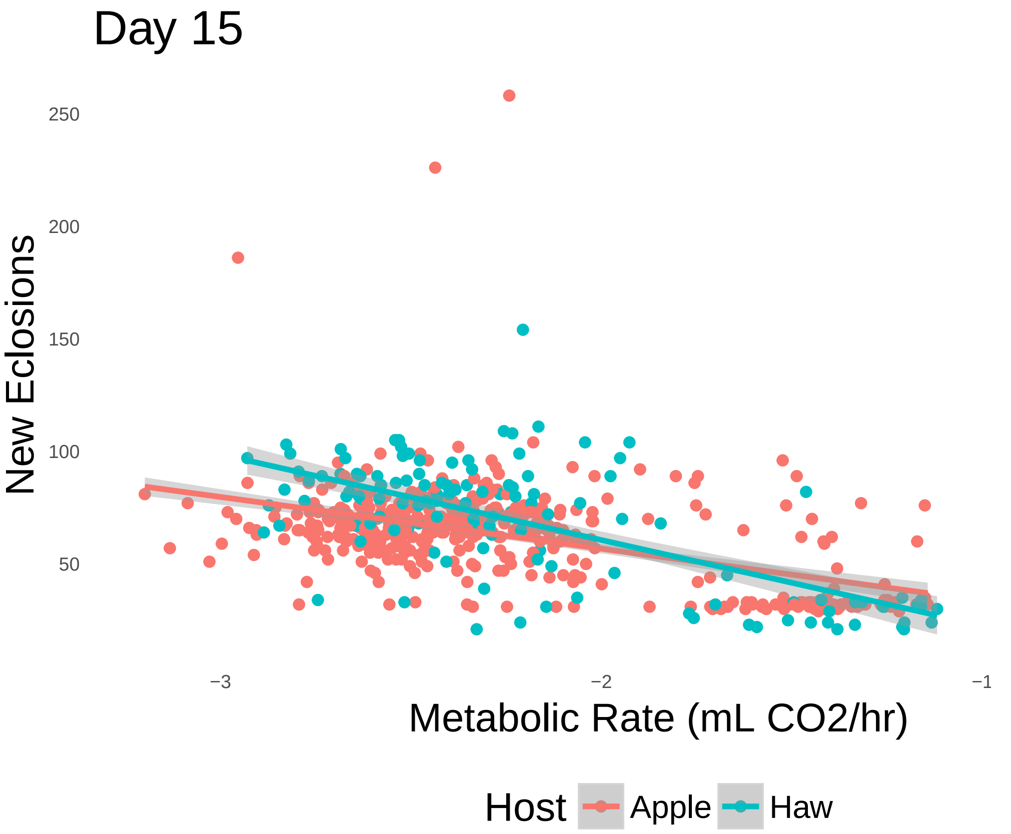
```
#same - fit linear curve to data
ggplot(data5.15.treatsub,aes(x=MR15.cor, y=neweclosions, colour=Host))+geom_point()+stat_smooth(method=
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.

## Warning: Removed 607 rows containing non-finite values (stat_smooth).

## Warning: Removed 607 rows containing missing values (geom_point).
```



```
#could log transform mr to make it more linear
ggplot(data5.15.treatsub,aes(x=log10(MR15.cor), y=neweclosions, colour=Host))+geom_point()+stat_smooth(
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

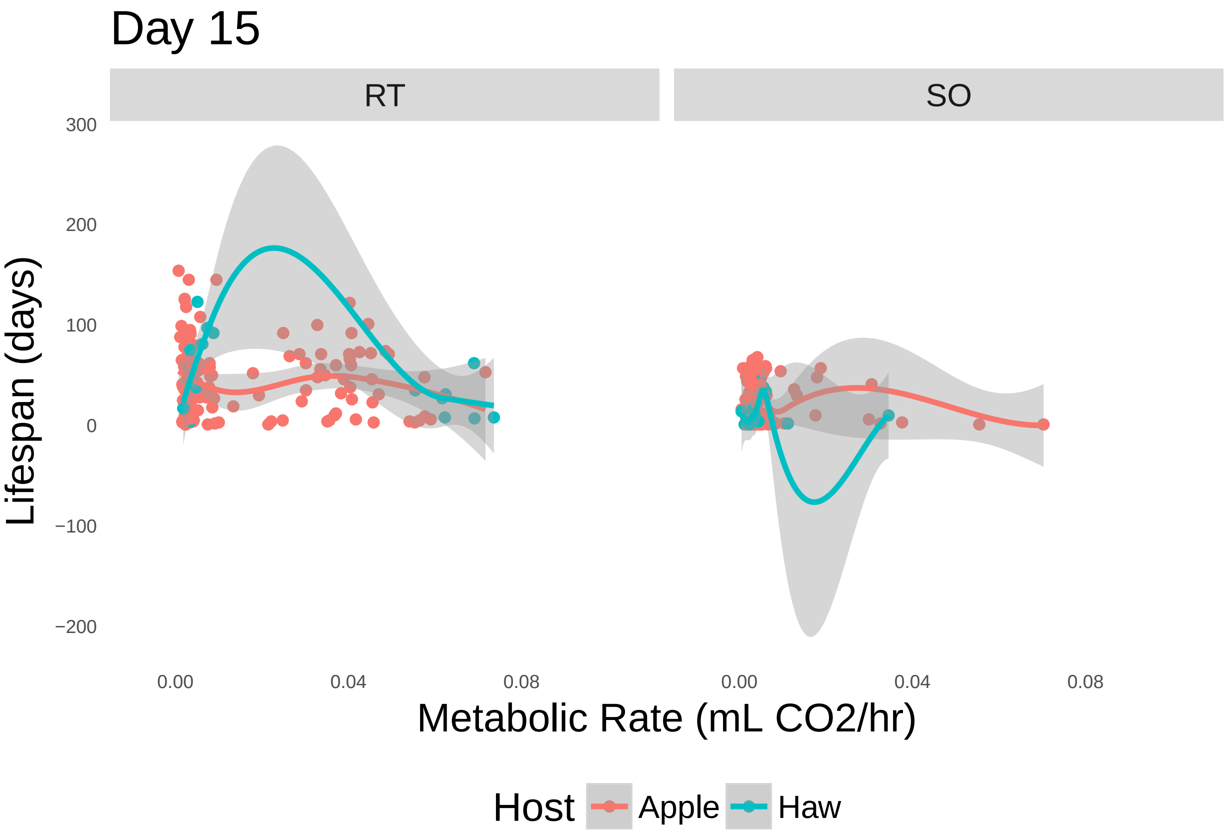
```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.

## Warning in FUN(X[[i]], ...): NaNs produced

## Warning in FUN(X[[i]], ...): NaNs produced

## Warning: Removed 619 rows containing non-finite values (stat_smooth).

## Warning: Removed 619 rows containing missing values (geom_point).
```



Figures associating MR with lifespan

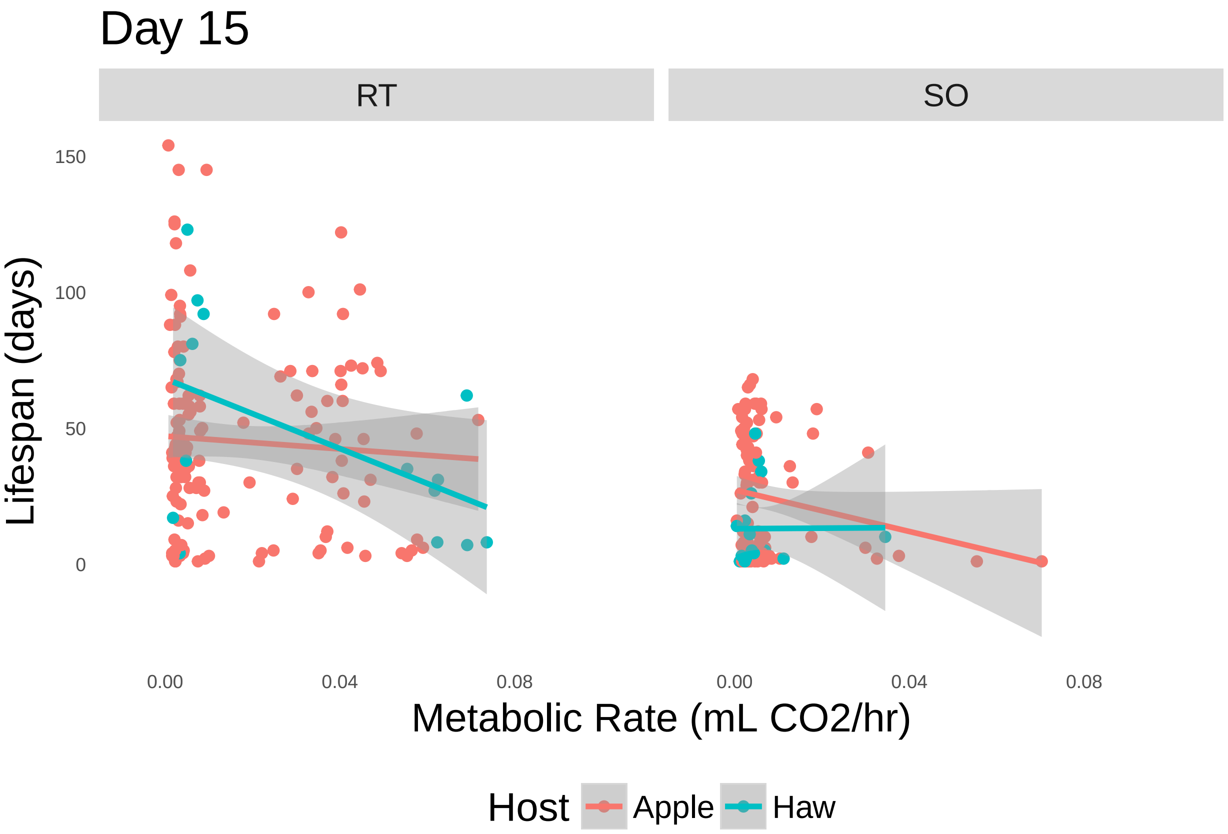
```
#mr15 w lifespan
data5.15.treatsub$lifespan<-difftime(as.Date(data5.15.treatsub$Adult_death_date, na.rm=TRUE), as.Date(da

ggplot(data5.15.treatsub,aes(x= MR15.cor, y=lifespan, colour=Host))+geom_point()+stat_smooth(method="lo
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.

## Warning: Removed 928 rows containing non-finite values (stat_smooth).

## Warning: Removed 928 rows containing missing values (geom_point).
```



```
ggplot(data5.15.treatsub,aes(x=MR15.cor, y=lifespan, colour=Host))+geom_point()+stat_smooth(method="lm"
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```

```
## Don't know how to automatically pick scale for object of type difftime. Defaulting to continuous.

## Warning: Removed 928 rows containing non-finite values (stat_smooth).

## Warning: Removed 928 rows containing missing values (geom_point).
```



Double Checking Eclosion for Pupal Deaths

```
#Assign names to each column and column bind in order to simplify and find data easily
host <- data[,11]
cd <- data[,10]
wid <- data[,29]
unid<-data[,26]
edate <- data[,27]
dcheck<-cbind(host, cd, wid, unid, edate)
```

Merging pupal death data sheet lifespans

Data available on github

```
pupdeaths<- fread("https://raw.githubusercontent.com/adnguyen/Circadian_rhythm_runs_seasonal_timing/mas

#Filter out non-eclosers
data5.na <- data5.15%>%
  mutate(neweclosions=as.numeric(neweclosions))%>%
  #glimpse()
  filter(is.na(neweclosions))

#Filter out individuals with adult lifespans
data5.ls <- data5.15.treatsub%>%
  filter(!is.na(lifespan))
```

```r
#Merge the two sets of data (lifespan and non-eclosers) by "Host", "cohort_day", "well_id", and "unique
mergedat1 <- inner_join(pupdeaths, data5.na, by = c("Host", "cohort_day", "well_id", "uniqueID"))

#column bind the merged data with the lifespan data to check same column dim
#cbind(names(data5.ls), names(mergedat1)[-56])

#once the column dimensions are the same, merge the two sets of data
mergedata<- merge(data5.ls, mergedat1)

#Finally, row bind the data with the desired conditions
finalmerge<- rbind(data5.ls[,c("Host", "cohort_day", "well_id", "uniqueID", "lifespan", "MR15.cor","msMI

#glimpse(finalmerge)
```

Negative binomial regression reanalyzed with pupal death data

```r
#Filtering done to separate the two treatments
RT15.n <- finalmerge%>%
    filter(treatment=="RT")

SO15.n <- finalmerge%>%
    filter(treatment=="SO")

#RT summary (no stat significance)
RT15.n$lifespan <- as.numeric(RT15.n$lifespan)
mod3.n<- glm.nb(lifespan~MR15.cor*Host, data=RT15.n)
summary(mod3.n)
```

```
##
## Call:
## glm.nb(formula = lifespan ~ MR15.cor * Host, data = RT15.n, init.theta = 0.4064870624,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9351  -1.3934  -0.1234   0.3376   1.2813
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)        3.5189     0.1575  22.345   <2e-16 ***
## MR15.cor           3.6204     7.0097   0.516    0.606
## HostHaw           -0.1046     0.4437  -0.236    0.814
## MR15.cor:HostHaw  -7.0638    13.6632  -0.517    0.605
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.4065) family taken to be 1)
##
##     Null deviance: 231.07  on 195  degrees of freedom
## Residual deviance: 230.28  on 192  degrees of freedom
## AIC: 1705
##
```

```
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.4065
##           Std. Err.:  0.0419
##
##  2 x log-likelihood:  -1694.9600
```

```
#summary : HostHaw (in reference to apple) - as lifespan increases, the MR decreases by the estimate (l
#Coefficient describes relationship, i.e. for mod4SO, higher mass=higher life span according to this mo
  #mod4 shows interaction b/w host and mr
mod4.nRT<-glm.nb(lifespan~MR15.cor*Host + mass_day14, data=RT15.n)
summary(mod4.nRT)
```

```
##
## Call:
## glm.nb(formula = lifespan ~ MR15.cor * Host + mass_day14, data = RT15.n,
##     init.theta = 0.4249870689, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0723  -1.3914  -0.2088   0.3326   1.4933
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)        2.1633     0.3769   5.740 9.49e-09 ***
## MR15.cor           1.4843     7.0176   0.212 0.832488
## HostHaw           -0.4048     0.4368  -0.927 0.353964
## mass_day14         0.2047     0.0553   3.702 0.000214 ***
## MR15.cor:HostHaw  -3.2856    13.4285  -0.245 0.806711
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.425) family taken to be 1)
##
##     Null deviance: 239.80  on 195  degrees of freedom
## Residual deviance: 230.27  on 191  degrees of freedom
## AIC: 1698.5
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.4250
##           Std. Err.:  0.0442
##
##  2 x log-likelihood:  -1686.4600
```

```
SO15.n$lifespan <- as.numeric(SO15.n$lifespan)
mod4.nSO<-glm.nb(lifespan~MR15.cor*Host + mass_day14, data=SO15.n)
summary(mod4.nSO)
```

```
##
## Call:
## glm.nb(formula = lifespan ~ MR15.cor * Host + mass_day14, data = SO15.n,
##     init.theta = 0.8729265479, link = log)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2740  -1.2062  -0.2847   0.5513   1.6254
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        2.55542    0.42475   6.016 1.78e-09 ***
## MR15.cor         -30.63664   11.28046  -2.716  0.00661 **
## HostHaw           -1.13706    0.37668  -3.019  0.00254 **
## mass_day14         0.11472    0.05573   2.058  0.03956 *
## MR15.cor:HostHaw  33.18788   37.14903   0.893  0.37166
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.8729) family taken to be 1)
##
##     Null deviance: 139.26  on 111  degrees of freedom
## Residual deviance: 126.36  on 107  degrees of freedom
##   (1 observation deleted due to missingness)
## AIC: 923.59
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.873
##           Std. Err.:  0.112
##
##  2 x log-likelihood:  -911.594
```

```
figmod4.nSO<-glm.nb(lifespan~MR15.cor*Host, data=SO15.n)
summary(figmod4.nSO)
```

```
##
## Call:
## glm.nb(formula = lifespan ~ MR15.cor * Host, data = SO15.n, init.theta = 0.8485587795,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3936  -1.2487  -0.4853   0.5522   1.5282
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)        3.3912     0.1375  24.659  < 2e-16 ***
## MR15.cor         -32.9921    11.4113  -2.891  0.00384 **
## HostHaw           -0.8280     0.3668  -2.257  0.02400 *
## MR15.cor:HostHaw  34.0782    37.5036   0.909  0.36353
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.8486) family taken to be 1)
##
##     Null deviance: 135.74  on 111  degrees of freedom
## Residual deviance: 126.80  on 108  degrees of freedom
##   (1 observation deleted due to missingness)
## AIC: 925.26
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.849
##           Std. Err.:  0.108
##
##  2 x log-likelihood:  -915.265
```

```
figmod4.nRT<-glm.nb(lifespan~MR15.cor*Host, data=RT15.n)
summary(figmod4.nRT)
```

```
##
## Call:
## glm.nb(formula = lifespan ~ MR15.cor * Host, data = RT15.n, init.theta = 0.4064870624,
##     link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9351  -1.3934  -0.1234   0.3376   1.2813
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)        3.5189     0.1575  22.345   <2e-16 ***
## MR15.cor           3.6204     7.0097   0.516    0.606
## HostHaw           -0.1046     0.4437  -0.236    0.814
## MR15.cor:HostHaw  -7.0638    13.6632  -0.517    0.605
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.4065) family taken to be 1)
##
##     Null deviance: 231.07  on 195  degrees of freedom
## Residual deviance: 230.28  on 192  degrees of freedom
## AIC: 1705
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.4065
##           Std. Err.:  0.0419
##
##  2 x log-likelihood:  -1694.9600
```

```
  #mod4.1 shows interaction b/w host and msmr
mod4.1.nSO<-glm.nb(lifespan~msMR15.cor*Host, data=SO15.n)
summary(mod4.1.nSO)
```

```
##
## Call:
## glm.nb(formula = lifespan ~ msMR15.cor * Host, data = SO15.n,
##     init.theta = 0.848594471, link = log)
##
```

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3703  -1.2461  -0.3624   0.6299   1.5013
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)            3.2505     0.1172  27.734  < 2e-16 ***
## msMR15.cor          -182.2773    62.8655  -2.899  0.00374 **
## HostHaw               -0.6846     0.2971  -2.304  0.02123 *
## msMR15.cor:HostHaw   224.5755   356.0600   0.631  0.52822
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.8486) family taken to be 1)
##
##     Null deviance: 135.75  on 111  degrees of freedom
## Residual deviance: 126.85  on 108  degrees of freedom
##   (1 observation deleted due to missingness)
## AIC: 925.31
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.849
##           Std. Err.:  0.108
##
##  2 x log-likelihood:  -915.308
```

```
mod4.1.nRT<-glm.nb(lifespan~msMR15.cor*Host, data=RT15.n)
summary(mod4.1.nRT)
```

```
##
## Call:
## glm.nb(formula = lifespan ~ msMR15.cor * Host, data = RT15.n,
##     init.theta = 0.4071281808, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9507  -1.3966  -0.1284   0.3380   1.2736
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)          3.53929    0.13867  25.524   <2e-16 ***
## msMR15.cor          21.20632   44.94809   0.472    0.637
## HostHaw             -0.08528    0.39366  -0.217    0.828
## msMR15.cor:HostHaw -67.07964   78.15514  -0.858    0.391
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.4071) family taken to be 1)
##
##     Null deviance: 231.38  on 195  degrees of freedom
## Residual deviance: 230.28  on 192  degrees of freedom
## AIC: 1704.7
##
```

```
## Number of Fisher Scoring iterations: 1
##
##
##                Theta:  0.4071
##            Std. Err.:  0.0420
##
##  2 x log-likelihood:  -1694.6580
```

```
#Likelihood ratio tests of Negative Binomial Models
m1.nRT <- update(mod4.nRT, . ~ . - prog)
anova(mod4.nRT, m1.nRT)
```

```
## Likelihood ratio tests of Negative Binomial Models
##
## Response: lifespan
##                                           Model     theta Resid. df
## 1                  MR15.cor * Host + mass_day14 0.4249871       191
## 2 MR15.cor + Host + mass_day14 + MR15.cor:Host 0.4249871       191
##      2 x log-lik.   Test    df      LR stat. Pr(Chi)
## 1        -1686.46
## 2        -1686.46 1 vs 2     0 -3.760033e-08       1
```

```
#Checking model assumption -- is negative binomial regression (NBR) a good model for this data?
##Values close to 0 (estimates the dispersion parameter) strongly suggest the NBR model is more appropr
mod5.n <- glm(lifespan ~ Host*MR15.cor, family = "poisson", data = SO15.n)
pchisq(2 * (logLik(mod4.nRT) - logLik(mod5.n)), df = 1, lower.tail = FALSE)
```

```
## 'log Lik.' 6.470061e-238 (df=6)
```

```
mod5.n <- glm(lifespan ~ Host*MR15.cor, family = "poisson", data = SO15.n)
pchisq(2 * (logLik(mod4.nSO) - logLik(mod5.n)), df = 1, lower.tail = FALSE)
```

```
## 'log Lik.' 0 (df=6)
```

```
#output below indicates that Hawthorn is 0.432 times less likely to live than apple
#confidence interval
(est <- cbind(Estimate = coef(mod4.nRT), confint(mod4.nRT)))
```

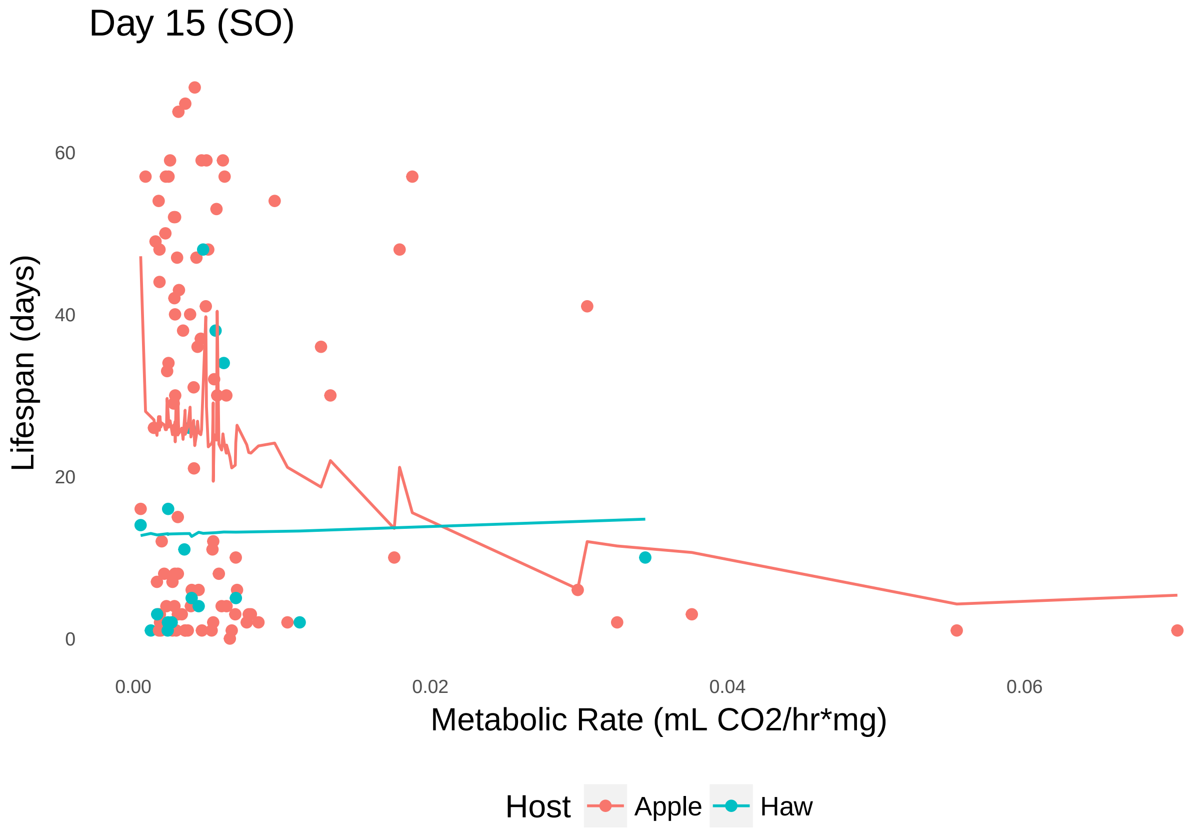
```
## Waiting for profiling to be done...
```

```
##                     Estimate       2.5 %     97.5 %
## (Intercept)        2.1633428   1.2237554  3.1119932
## MR15.cor           1.4842992 -11.7040163 15.8181228
## HostHaw           -0.4048505  -1.2336952  0.5650535
## mass_day14         0.2047244   0.0676875  0.3465155
## MR15.cor:HostHaw -3.2855872 -29.3106300 25.7786008
```

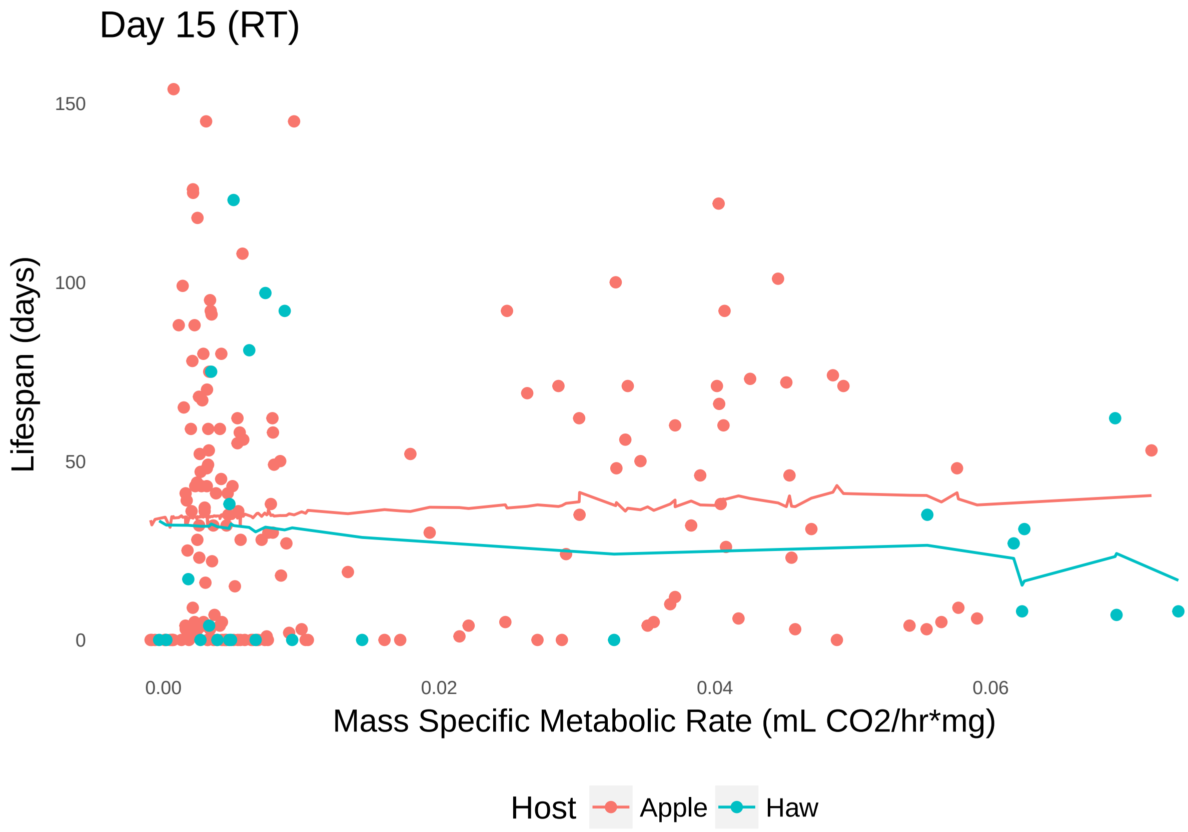
```
#incident rate ratio - lifespan ratio
exp(est)
```

```
##                     Estimate        2.5 %       97.5 %
## (Intercept)       8.70017236 3.399932e+00 2.246578e+01
## MR15.cor          4.41187261 8.260576e-06 7.408384e+06
## HostHaw           0.66707656 2.912145e-01 1.759542e+00
## mass_day14        1.22718686 1.070031e+00 1.414131e+00
## MR15.cor:HostHaw 0.03741861 1.864469e-13 1.568571e+11
```

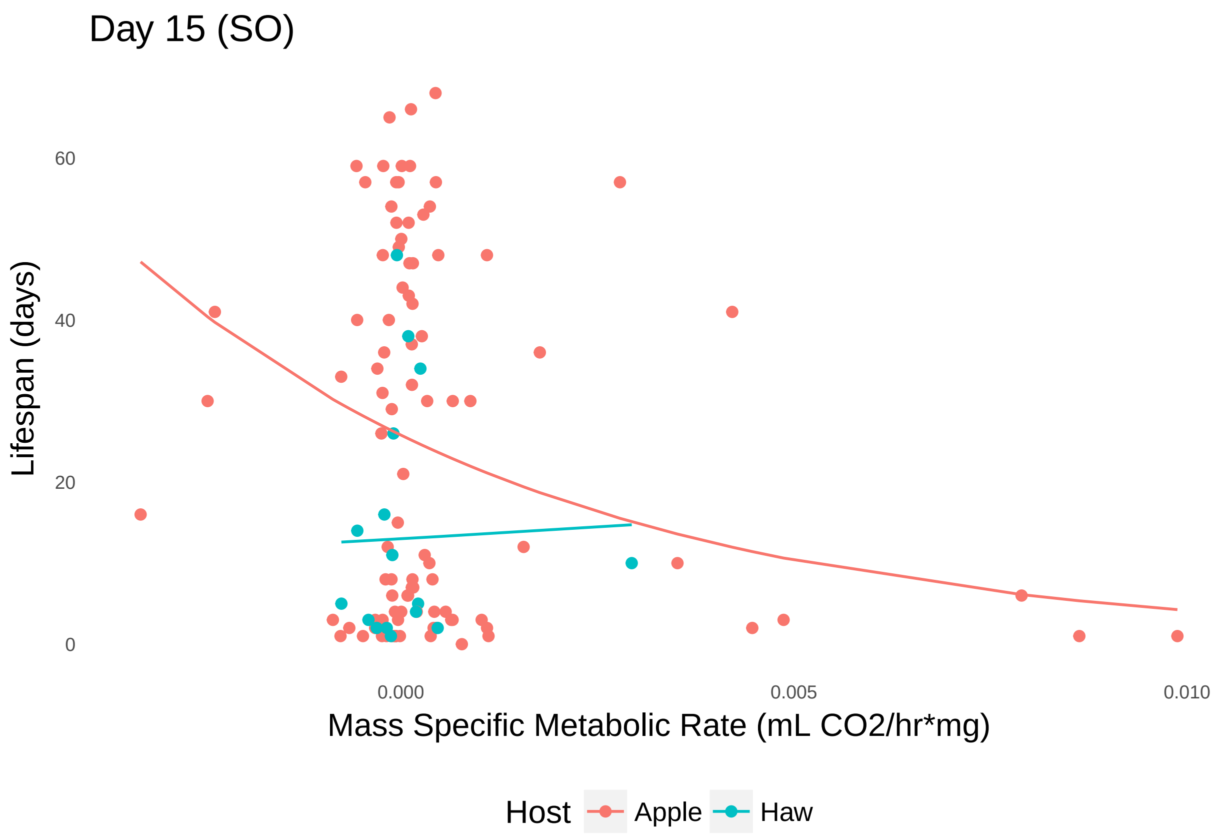
```
#Filter out all the NAs in order to have same dimensions for predvalues
filSO15.ms<- SO15.n%>%
  filter(msMR15.cor!= "NA"&lifespan !="NA")

filRT15.ms<- RT15.n%>%
  filter(msMR15.cor!= "NA"&lifespan !="NA")

#Calculate the predicted values for msMR15
filSO15.ms$predvalues <- predict(mod4.1.nSO, type = "response")

filRT15.ms$predvalues <- predict(mod4.1.nRT, type = "response")

#Filter out all the NAs in order to have same dimensions for predvalues
filSO15.ms<- SO15.n%>%
  filter(msMR15.cor!= "NA"&lifespan !="NA")

filRT15.ms<- RT15.n%>%
  filter(msMR15.cor!= "NA"&lifespan !="NA")

#Calculate the predicted values
filSO15.ms$predvalues <- predict(mod4.1.nSO, type = "response")

filRT15.ms$predvalues <- predict(mod4.1.nRT, type = "response")

#overlay of real data(points)  with predicted values(line) for MR
ggplot(filSO15.ms,aes(x=MR15.cor, y=lifespan, colour=Host))+geom_point()+geom_line(data=filSO15.ms, aes
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```



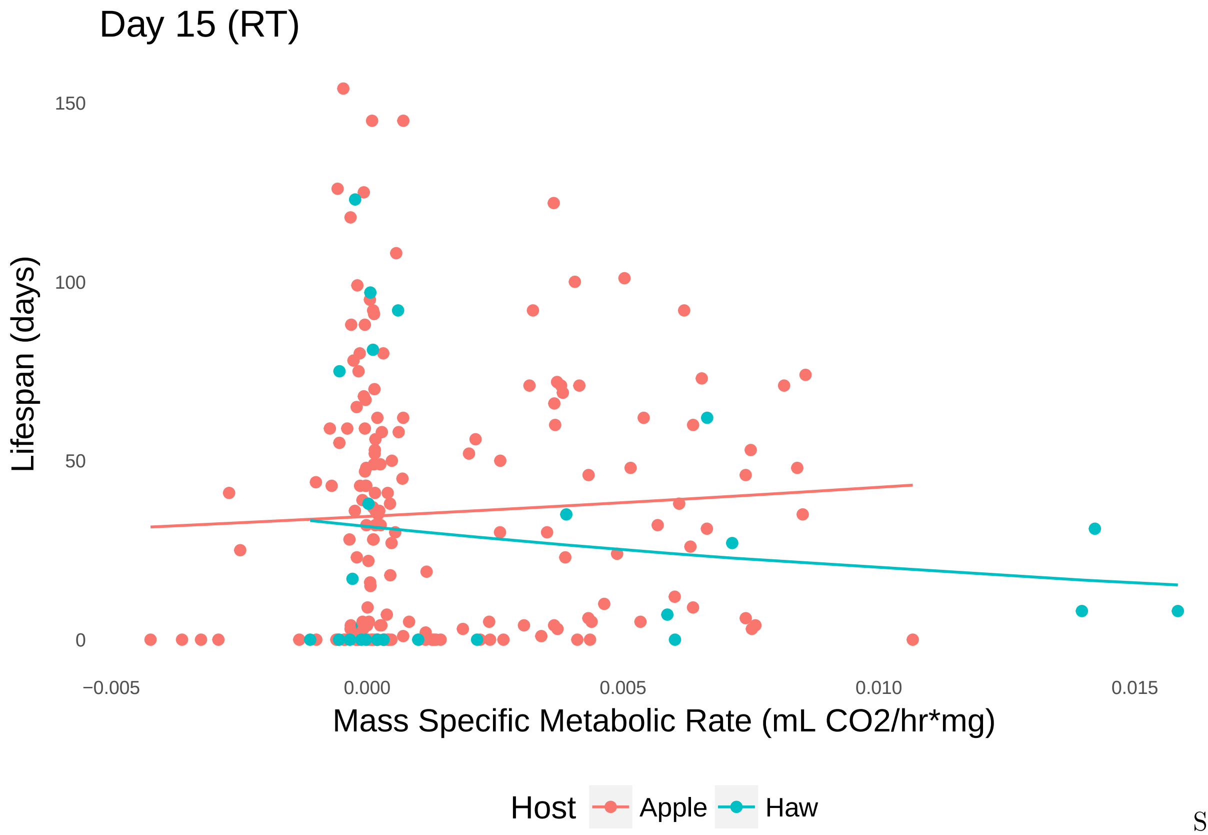
```
ggplot(filRT15.ms,aes(x=MR15.cor, y=lifespan, colour=Host))+geom_point()+geom_line(data=filRT15.ms, aes
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```



```
#overlay of real data(points)  with predicted values(line) for msMR
ggplot(filSO15.ms,aes(x=msMR15.cor, y=lifespan, colour=Host))+geom_point()+geom_line(data=filSO15.ms, a
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```



```
ggplot(filRT15.ms,aes(x=msMR15.cor, y=lifespan, colour=Host))+geom_point()+geom_line(data=filRT15.ms, a
        axis.ticks.x=element_blank(),legend.position="bottom",
        axis.ticks.y=element_blank(),panel.background = element_blank(),
        panel.grid.major = element_blank(), panel.grid.minor = element_blank(),axis.text=element_text(s
```



Samples were split into two treatments, RT (favorable conditions) and SO (simulated overwintering). The dependent variable was lifespan, while the independent variables were metabolic rate in relation to host and mass. For this model, a negative coefficient of the log value would indicate a shorter lifespan since it produces a log-linear model in which a positive coefficient would indicate a (larger x and y value).

Based on the RT treatment, only mass showed significance which suggests an individual with a larger mass would have a longer lifespan (positive coefficient). Analysis of the SO treatment showed significance in host type, metabolic rate, and mass, in which a lower metabolic rate would suggest longer lifespan (neg. coefficient), and larger mass would suggest longer lifespan(pos. coefficient). Hawthorn individuals also had an overall lower lifespan compared to apple.

Negative binomial regression was shown to be a good model by comparing its log likelihood to a poisson distribution. Values close to 0 (which estimates the dispersion parameter) strongly suggests the NBR model is more appropriate than the Poisson model for both treatments.

Kaplan-Meier Survival Estimate

```
#install.packages(c("survival", "survminer"))
#install.packages(c("ggpubr", "magrittr"))
#Assign censored data and create new column
finalmerge$status[finalmerge$lifespan >= 0] <- 1
```

```
## Warning: Unknown or uninitialised column: 'status'.
```

```
#filter out unwanted treatments
fmerge <- finalmerge %>%
      filter(treatment != "")
```

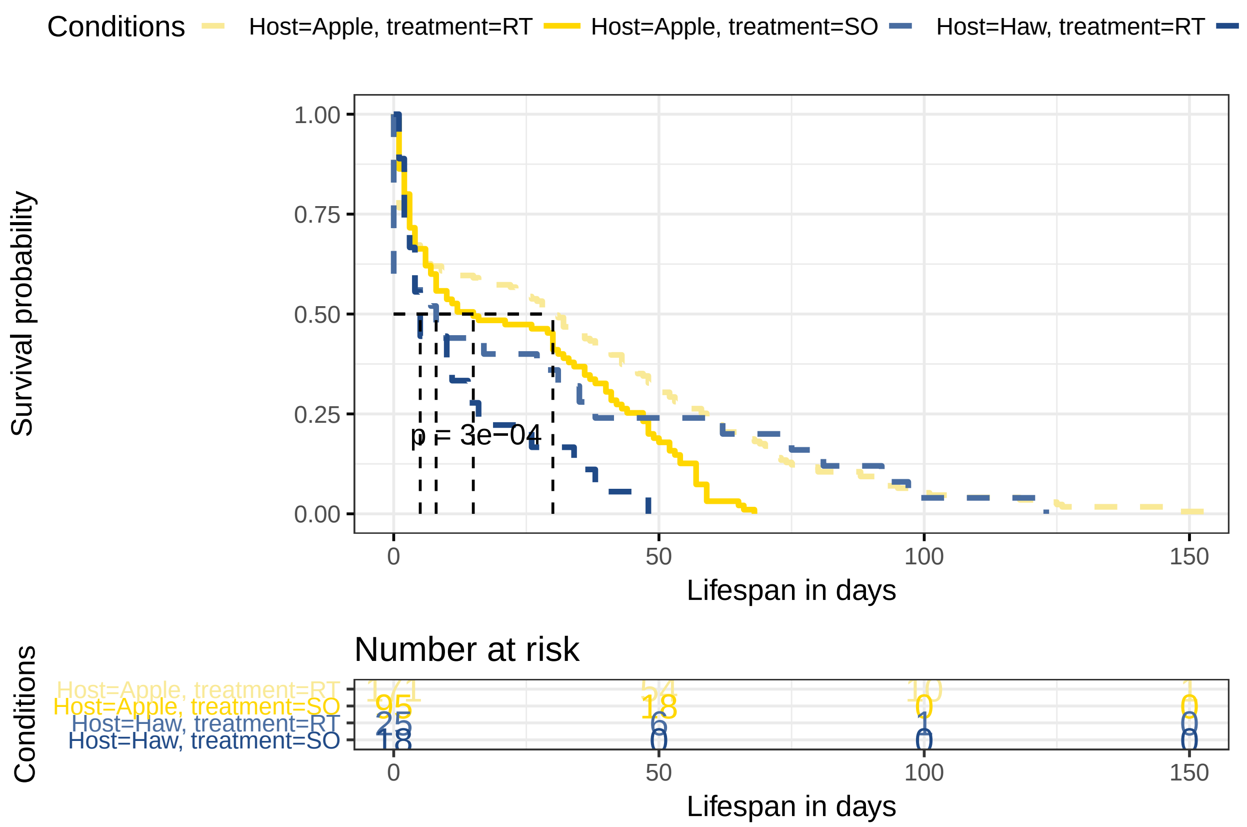
```
#Filter out unwanted treatments
finalmerge <- finalmerge %>%
  filter(treatment != "")

#compute kaplan-Meier survival estimate-so compute the survival probability by host & treatment:
  #fit1 computes survival probability by Host + Treatment

fit1 <- survfit(Surv(lifespan, status) ~ Host + treatment, data = finalmerge)
print(fit1)
```

```
## Call: survfit(formula = Surv(lifespan, status) ~ Host + treatment,
##     data = finalmerge)
##
##                              n events median 0.95LCL 0.95UCL
## Host=Apple, treatment=RT 171    171     30      22      39
## Host=Apple, treatment=SO  95     95     15       8      33
## Host=Haw, treatment=RT    25     25      8       0      38
## Host=Haw, treatment=SO    18     18      5       3      26
```

```
# Access to the sort summary table
summary(fit1)$table
```

```
##                          records n.max n.start events   *rmean *se(rmean)
## Host=Apple, treatment=RT     171   171     171    171 33.91520   2.439589
## Host=Apple, treatment=SO      95    95      95     95 24.31579   2.271454
## Host=Haw, treatment=RT        25    25      25     25 27.04000   6.846483
## Host=Haw, treatment=SO        18    18      18     18 12.55556   3.296296
##                          median 0.95LCL 0.95UCL
## Host=Apple, treatment=RT     30      22      39
## Host=Apple, treatment=SO     15       8      33
## Host=Haw, treatment=RT        8       0      38
## Host=Haw, treatment=SO        5       3      26
```

```
# Visualize curves (Change color, linetype by strata, risk.table color by strata)
ggsurvplot(fit1,
          pval = TRUE, conf.int = FALSE,
          risk.table = TRUE, # Add risk table
          risk.table.col = "strata", # Change risk table color by groups
          linetype = c("dashed","solid", "dashed", "dashed","solid"), # Change line type
          surv.median.line = "hv", # Specify median survival
          ggtheme = theme_bw(), # Change ggplot2 theme
          palette = c("#F9E996","#FFD700","#496DA1","#204A87"),
          xlab = "Lifespan in days",
          legend.title = "Conditions",
          pval.size=4)
```



The Kaplan-Meier Survival Estimate predicts the probability of survival from a specific time, so for this data, the estimates predict the probability of host survival at each time point in their lifespan. According to the figure, Hawthorn flies have an overall lower probability of survival over apple flies, which agrees with the previous analyses. This estimate is best used for categorical values, so MR and mass (which were all unique to the individul) were not good estimators and not used.

Cox Regression

```
#Split up treatments into: SO & RT
fmergeSO <- fmerge%>%
  filter(treatment == "SO")

fmergeRT <- fmerge%>%
  filter(treatment == "RT")

#Merge the data for censoring (all labeled as status 1 for dead)
covariates <- c("host", "treatment", "MR15.cor","mass_day14")
#For RT
univ_formulasRT <- sapply(covariates,
                        function(x) as.formula(paste('Surv(lifespan, status)~', fmergeRT)))
#For SO
univ_formulasSO <- sapply(covariates,
                        function(x) as.formula(paste('Surv(lifespan, status)~', fmergeSO)))

univ_modelsRT <- lapply(univ_formulasRT, function(x){coxph(x, data = fmergeRT)})
univ_modelsSO <- lapply(univ_formulasSO, function(x){coxph(x, data = fmergeSO)})
```

```r
#Multivariate case
#Testing interaction between Host, Treatment, and MR -- Shows interaction b/w treatment & Host
res.cox <- coxph(Surv(lifespan, status) ~ Host*treatment*MR15.cor + mass_day14, data =  fmerge)
summary(res.cox)
```

```
## Call:
## coxph(formula = Surv(lifespan, status) ~ Host * treatment * MR15.cor +
##     mass_day14, data = fmerge)
##
##   n= 308, number of events= 308
##    (1 observation deleted due to missingness)
##
##                                    coef  exp(coef)   se(coef)      z
## HostHaw                       2.178e-01  1.243e+00  2.887e-01  0.754
## treatmentSO                   3.257e-01  1.385e+00  1.698e-01  1.918
## MR15.cor                     -1.494e+00  2.244e-01  4.562e+00 -0.328
## mass_day14                   -1.362e-01  8.727e-01  3.457e-02 -3.940
## HostHaw:treatmentSO           6.938e-01  2.001e+00  4.437e-01  1.564
## HostHaw:MR15.cor              5.393e+00  2.198e+02  8.805e+00  0.612
## treatmentSO:MR15.cor          1.858e+01  1.170e+08  1.163e+01  1.597
## HostHaw:treatmentSO:MR15.cor -2.660e+01  2.804e-12  3.695e+01 -0.720
##                              Pr(>|z|)
## HostHaw                        0.4506
## treatmentSO                    0.0551 .
## MR15.cor                       0.7432
## mass_day14                   8.15e-05 ***
## HostHaw:treatmentSO            0.1179
## HostHaw:MR15.cor               0.5402
## treatmentSO:MR15.cor           0.1102
## HostHaw:treatmentSO:MR15.cor   0.4716
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                              exp(coef) exp(-coef) lower .95 upper .95
## HostHaw                      1.243e+00  8.043e-01 7.060e-01 2.190e+00
## treatmentSO                  1.385e+00  7.220e-01 9.929e-01 1.932e+00
## MR15.cor                     2.244e-01  4.457e+00 2.935e-05 1.715e+03
## mass_day14                   8.727e-01  1.146e+00 8.155e-01 9.338e-01
## HostHaw:treatmentSO          2.001e+00  4.997e-01 8.388e-01 4.775e+00
## HostHaw:MR15.cor             2.198e+02  4.550e-03 7.032e-06 6.870e+09
## treatmentSO:MR15.cor         1.170e+08  8.550e-09 1.469e-02 9.314e+17
## HostHaw:treatmentSO:MR15.cor 2.804e-12  3.566e+11 9.964e-44 7.892e+19
##
## Concordance= 0.611  (se = 0.021 )
## Rsquare= 0.107   (max possible= 1 )
## Likelihood ratio test= 34.75  on 8 df,   p=3e-05
## Wald test            = 36.24  on 8 df,   p=2e-05
## Score (logrank) test = 37.24  on 8 df,   p=1e-05
```

```r
#Coefficient describes relationship: for Cox Regression, coefficient describes relationship to likeliho
#Test interaction between Host, RT, and MR
res.coxRT <- coxph(Surv(lifespan, status) ~ Host*MR15.cor + mass_day14, data =  fmergeRT)
summary(res.coxRT)
```

```
## Call:
```

```
## coxph(formula = Surv(lifespan, status) ~ Host * MR15.cor + mass_day14,
##     data = fmergeRT)
##
##   n= 196, number of events= 196
##
##                      coef exp(coef) se(coef)      z Pr(>|z|)
## HostHaw           0.26512   1.30358  0.29147  0.910 0.363049
## MR15.cor         -0.50118   0.60582  4.57571 -0.110 0.912782
## mass_day14       -0.15524   0.85621  0.04451 -3.488 0.000488 ***
## HostHaw:MR15.cor  2.90449  18.25589  8.85836  0.328 0.743002
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                  exp(coef) exp(-coef) lower .95 upper .95
## HostHaw             1.3036    0.76712 7.363e-01 2.308e+00
## MR15.cor            0.6058    1.65067 7.718e-05 4.755e+03
## mass_day14          0.8562    1.16794 7.847e-01 9.343e-01
## HostHaw:MR15.cor   18.2559    0.05478 5.262e-07 6.334e+08
##
## Concordance= 0.624  (se = 0.028 )
## Rsquare= 0.068   (max possible= 1 )
## Likelihood ratio test= 13.77  on 4 df,   p=0.008
## Wald test            = 13.57  on 4 df,   p=0.009
## Score (logrank) test = 13.62  on 4 df,   p=0.009
```

```
#Test interaction between Host, SO, and MR
res.coxSO <- coxph(Surv(lifespan, status) ~ Host*MR15.cor + mass_day14, data =  fmergeSO)
summary(res.coxSO)
```

```
## Call:
## coxph(formula = Surv(lifespan, status) ~ Host * MR15.cor + mass_day14,
##     data = fmergeSO)
##
##   n= 112, number of events= 112
##    (1 observation deleted due to missingness)
##
##                        coef  exp(coef)   se(coef)      z Pr(>|z|)
## HostHaw           1.175e+00  3.238e+00  3.826e-01  3.071  0.00213 **
## MR15.cor          3.096e+01  2.780e+13  1.176e+01  2.632  0.00849 **
## mass_day14       -1.032e-01  9.019e-01  5.506e-02 -1.874  0.06088 .
## HostHaw:MR15.cor -3.888e+01  1.305e-17  3.748e+01 -1.037  0.29962
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                  exp(coef) exp(-coef) lower .95 upper .95
## HostHaw          3.238e+00  3.088e-01 1.530e+00 6.853e+00
## MR15.cor         2.780e+13  3.597e-14 2.707e+03 2.855e+23
## mass_day14       9.019e-01  1.109e+00 8.097e-01 1.005e+00
## HostHaw:MR15.cor 1.305e-17  7.663e+16 1.627e-49 1.047e+15
##
## Concordance= 0.593  (se = 0.035 )
## Rsquare= 0.122   (max possible= 0.999 )
## Likelihood ratio test= 14.57  on 4 df,   p=0.006
## Wald test            = 16.07  on 4 df,   p=0.003
## Score (logrank) test = 16.72  on 4 df,   p=0.002
```

Samples were split into two treatments, RT (favorable conditions) and SO (simulated overwintering). The dependent variable was lifespan, while the independent variables were metabolic rate in relation to host and mass. For this model, the coefficient relates x variables (metabolic rate, host, and mass) to likelihood of death (hazard rate). The dependent variable is lifespan again.

The RT treatment only showed significance for mass (negative coefficient), in which a lower mass suggests a higher likelihood of death (i.e. longer lifespan).

The SO treatment showed significance for metabolic rate, host, and mass. The positive coefficient for the hawthorn flies and metabolic rate indicate higher likelihood of death if the individual is a host to hawthorn and/or the individual has a high metabolic rate. This agrees with the NBR model. However, the negative coefficient for mass indicates a higher likelihood of death with smaller masses which agrees with the NBR model.

Conclusions

Evolutionarily speaking, linking lower metabolic rate to longer lifespan makes sense since an organism who reserves its energy would take longer to develop, thus longer to die. For example early eclosers tend to have higher metabolic rate vs. late eclosers, so if a group of insects undergo diapause at the same time, those with higher metabolic rates risk eclosing too early in the season, before optimal conditions are available. Late eclosers have a higher energy reserve because they don't need as much energy with lower metabolic rates. This allows them to eclose at suitable seasonal time, lengthening their lifespan/hazard rate.

From this data, the cohorts placed in favorable conditions showed no significance when relating metabolic rate and lifespan. This is most likely because these cohorts had no survival need to undergo diapause which would have separated out the individuals with high and low metabolic rates. Those placed in simulated overwintering conditions would be affected by their metabolic rates – individuals with high metabolic rates would be more likely to eclose earlier and die earlier. Individuals with low metabolic rates would spend more time in diapause and eclose later, extending their lifespan.

According to the evolution theory described in Olshansky & Rattan (2009), higher metabolic rates are liked to species with "high extrinsic mortality" (e.g. targeted prey) since they must quickly develop and reproduce in order to pass down their genes. Lower metabolic rates were linked to individuals with lower extrinsic mortalities and could take their time to gradually develop into larger sizes and live longer. For this theory to hold true, lifespan would need to be considered as a function of metabolic rate.

However, the evolution theory does not consider seasonal timing in this case. It may be possible that both early and late eclosers miss the optimal season for mating by eclosing before or after resource availability. Therefore, individuals who have metabolic rates that are too high or too low would not be able to pass down their genes. However, if we assume these individuals are still able to live, but not reproduce, their lifespans would not be an ideal indicator for fitness.

Session Info

```r
sessionInfo()
```

```
## R version 3.5.0 (2018-04-23)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS High Sierra 10.13.5
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib
```

```
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] bindrcpp_0.2.2    gridExtra_2.3     survminer_0.4.2
##  [4] ggpubr_0.1.6      magrittr_1.5      survival_2.42-3
##  [7] BBmisc_1.11       MASS_7.3-49       DiagrammeR_1.0.0
## [10] curl_3.2          lubridate_1.7.4   ggplot2_2.2.1
## [13] data.table_1.11.4 dplyr_0.7.5
##
## loaded via a namespace (and not attached):
##  [1] viridis_0.5.1      tidyr_0.8.1        jsonlite_1.5
##  [4] viridisLite_0.3.0  splines_3.5.0      assertthat_0.2.0
##  [7] yaml_2.1.19        pillar_1.2.3       backports_1.1.2
## [10] lattice_0.20-35    glue_1.2.0         downloader_0.4
## [13] digest_0.6.15      RColorBrewer_1.1-2 checkmate_1.8.5
## [16] colorspace_1.3-2   cmprsk_2.2-7       htmltools_0.3.6
## [19] Matrix_1.2-14      plyr_1.8.4         psych_1.8.4
## [22] XML_3.98-1.11      pkgconfig_2.0.1    broom_0.4.4
## [25] purrr_0.2.5        xtable_1.8-2       scales_0.5.0
## [28] brew_1.0-6         km.ci_0.5-2        KMsurv_0.1-5
## [31] tibble_1.4.2       influenceR_0.1.0   lazyeval_0.2.1
## [34] cli_1.0.0          mnormt_1.5-5       rgexf_0.15.3
## [37] crayon_1.3.4       evaluate_0.10.1    nlme_3.1-137
## [40] foreign_0.8-70     Rook_1.1-1         tools_3.5.0
## [43] hms_0.4.2          stringr_1.3.1      munsell_0.4.3
## [46] compiler_3.5.0     rlang_0.2.0        grid_3.5.0
## [49] rstudioapi_0.7     htmlwidgets_1.2    visNetwork_2.0.3
## [52] igraph_1.2.1       labeling_0.3       rmarkdown_1.9
## [55] gtable_0.2.0       reshape2_1.4.3     R6_2.2.2
## [58] zoo_1.8-1          knitr_1.20         survMisc_0.5.4
## [61] utf8_1.1.4         bindr_0.1.1        rprojroot_1.3-2
## [64] readr_1.1.1        stringi_1.2.2      parallel_3.5.0
## [67] Rcpp_0.12.17       tidyselect_0.2.4
```