

# Waal Intrinsic Variables and Parental Care Coordination Model Script

Fionnuala R McCully

11/08/2021

## Wandering Albatross Model Script

This R Markdown document details the code used to generate the results for a study on the impact of intrinsic variables on parental care coordination in wandering albatrosses. It covers the refinement of the data to create a data set suitable for analysis and the mixed models used to analyse the data.

## Data Setup, Descriptive Statistics and Variable Refinement

The following packages are used in this script. The data are uploaded to R Studio. The categorical variables are defined as factors.

```
#setup
setwd("~/R Practice/Wandering albatross/Model R Markdown")
library(dplyr)
library(MuMIn)
library(ggplot2)
library(lme4)
library(tidyverse)
library(optimx)
library(knitr)
library(car)
library(effects)
library(sjPlot)

# upload data and remove extra variable
total_df2<-read.csv("waal_dataset.csv")
total_df2$X<-NULL

#binary variables as factors
total_df2$sex<- as.factor(total_df2$sex)
total_df2$partner_type<- as.factor(total_df2$partner_type)
total_df2$breed_stage<-as.factor(total_df2$breed_stage)
total_df2$new_part<-as.factor(total_df2$new_part)
```

The full data set is divided into the incubation and brooding breeding stages. Each of these subsets will be analysed in a separate model.

```
# dividing brooding and incubation observations
incu_new<-subset(total_df2, breed_stage == "incu") # 260 observations
brood_new<-subset(total_df2, breed_stage=="brood") # 611 observations
```

The following code relates to generating the descriptive statistics for the incubation and brooding subsets, respectively. This includes examining sample sizes within these subsets and examining measures of centrality.

## Descriptive Statistics: Incubation

Raw sample sizes for the incubation subset:

```
#sample sizes- incubation
unique(incu_new$cycle_pair)
unique(incu_new$pair_id)
unique(incu_new$tag)#62 birds, 31 pairs
unique(incu_new$cycle) #7 breeding seasons
unique(incu_new$age) #no ages
```

Sample sizes per year and age group for the incubation subset:

```
with(incu_new, tapply(pair_id, cycle, FUN = function(x) length(unique(x)))) #
  pair sample size per year
```

```
## 2008 2009 2010 2011 2012 2013 2014
##    1    8    6    6    2    7    5
```

```
with(incu_new, tapply(tag, age, FUN = function(x) length(unique(x)))) # number
  birds per age group (individual birds may appear more than once)
```

```
##  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
## 34
##  1  2  1  1  1  3  2  4  4  2  3  2  4  5  4  3  3  1  2  2  2  3  2  2  1
## 1
## 35 36 37 38 39 41 42
##  1  2  1  1  1  1  2
```

```
table(incu_new$cycle) # no obs per year
```

```
##
## 2008 2009 2010 2011 2012 2013 2014
##   10   70   49   38   14   38   41
```

## Measures of centrality and group comparison for the incubation subset:

```
# measures of centrality- incubation
incu_new%>%summarise(average=mean(no_hours),med=median(no_hours), sd=sd(no_hours))
```

```
##      average      med      sd
## 1 134.3769 124.4167 102.6349
```

```
# males and female comparison
incu_new%>%group_by(sex)%>%summarise(average=mean(no_hours),med=median(no_hours), sd=sd(no_hours))
```

```
## # A tibble: 2 x 4
##   sex      average      med      sd
## * <fct>      <dbl> <dbl> <dbl>
## 1 Female      147.   132.  100.
## 2 Male        122.   112.  104.
```

```
# logged and partner bird (unlogged) comparison
incu_new%>%group_by(partner_type)%>%summarise(average=mean(no_hours),med= median(no_hours), sd=sd(no_hours))
```

```
## # A tibble: 2 x 4
##   partner_type average      med      sd
## * <fct>          <dbl> <dbl> <dbl>
## 1 Logger          130.   123.  96.7
## 2 Partner          139.   125. 109.
```

```
# interannual comparison
incu_new%>%group_by(cycle)%>%summarise(average=mean(no_hours),med=median(no_hours), sd=sd(no_hours))
```

```
## # A tibble: 7 x 4
##   cycle average      med      sd
## * <int>      <dbl> <dbl> <dbl>
## 1  2008     192.   185.  49.9
## 2  2009     143.   137.  111.
## 3  2010     151.   159.  112.
## 4  2011     102.    89.7  82.2
## 5  2012      94.5   103.  72.4
## 6  2013     119.   115.  91.0
## 7  2014     145.   130.  112.
```

## Descriptive Statistics: Brooding

Raw sample sizes for the brooding subset:

```
#sample sizes- brooding
unique(brood_new$cycle_pair)
unique(brood_new$pair_id)
unique(brood_new$tag)#132 birds, 66 pairs
unique(brood_new$cycle) #6 breeding seasons
unique(brood_new$age) #no ages
```

Sample sizes per year and age group for the brooding subset:

```
with(brood_new, tapply(pair_id, cycle, FUN = function(x) length(unique(x))))
# pair sample size per year
```

```
## 2008 2009 2010 2011 2012 2013
##    15    20     9    12    11     8
```

```
with(brood_new, tapply(tag, age, FUN = function(x) length(unique(x)))) # number birds per age group
```

```
##  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
## 32
##  2  3  1  9  4  8  7 10  6  6  7  5  7  4 11  8  6  7  3  4  6  4  5  5  2
## 3
## 33 34 35 37 40
##  1  2  1  2  1
```

```
table(brood_new$cycle) # no obs per year
```

```
##
## 2008 2009 2010 2011 2012 2013
## 137 174  70  81  84  65
```

Measures of centrality and group comparison for the brooding subset:

```
# measures of centrality- brooding
brood_new%>%summarise(average=mean(no_hours),med=median(no_hours), sd=sd(no_hours))
```

```
##      average      med      sd
## 1 65.31315 64.33333 32.59774
```

```
#males and females comparison
brood_new%>%group_by(sex)%>%summarise(average=mean(no_hours),med=median(no_hours), sd=sd(no_hours))
```

```
## # A tibble: 2 x 4
##   sex      average    med    sd
## * <fct>    <dbl> <dbl> <dbl>
## 1 Female      68.3   67.6  36.2
## 2 Male        62.3   58.3  28.3
```

```
# logged and partner birds comparison
brood_new%>%group_by(partner_type)%>%summarise(average=mean(no_hours),med= median(no_hours), sd=sd(no_hours))
```

```
## # A tibble: 2 x 4
##   partner_type average    med    sd
## * <fct>          <dbl> <dbl> <dbl>
## 1 Logger          59.6   57.3  27.7
## 2 Partner          70.3   69.1  35.7
```

```
#interannual comparison
brood_new%>%group_by(cycle)%>%summarise(average=mean(no_hours),med=median(no_hours), sd=sd(no_hours))
```

```
## # A tibble: 6 x 4
##   cycle average    med    sd
## * <int>    <dbl> <dbl> <dbl>
## 1  2008      59.7   58.5  27.9
## 2  2009      61.3   57.4  30.0
## 3  2010      66.5   61.1  35.8
## 4  2011      77.4   70.2  42.4
## 5  2012      69.2   71.4  28.3
## 6  2013      66.7   67.2  32.2
```

## Quadratic Variables, Scaling and Partner's Previous Trip Adjustment

Certain variables need to be adjusted for analysis. The instructions below include making a new data frame, calculating a quadratic age variable, scaling fixed effect variables, adding a categorical boldness variable (for graphical purposes only) and adjusting the partner's previous trip variable to account for each individual's mean trip duration.

## Variable Refinement: Incubation

Step 1: First, a new data frame is created to separate alterations from original data set. A check for NAs is carried out simultaneously (there should be none).The following actions are labeled as steps 1-5 to make comparison with the brooding code easier.

```
incu_new2<-na.omit(incu_new)# double check there are no NAs
```

Step 2: A quadratic variable representing age must be created. To prevent low sample sizes at the extremes of the age spectrum causing inaccuracies in the analyses, the highest and lowest age categories are collapsed so there is a minimum of 5 individuals at each end of the spectrum (see methods for further details and justification). This is done for both the focal bird age and also the partner bird age variables.

```
# collapse age group and create quadratic variable incubation

# focal bird age
with(incu_new2, tapply(tag, age, FUN = function(x) length(unique(x))))
```

```
##  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
34
##  1  2  1  1  1  3  2  4  4  2  3  2  4  5  4  3  3  1  2  2  2  3  2  2  1
1
## 35 36 37 38 39 41 42
##  1  2  1  1  1  1  2
```

```
incu_new2$age[incu_new2$age>36]<-36 # higher end of age spectrum
incu_new2$age[incu_new2$age<12]<-12 # lower end of age spectrum

#partner bird age
with(incu_new2, tapply(tag, p_age, FUN = function(x) length(unique(x))))
```

```
##  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33
34
##  1  2  1  1  1  3  2  4  4  2  3  2  4  5  4  3  3  1  2  2  2  3  2  2  1
1
## 35 36 37 38 39 41 42
##  1  2  1  1  1  1  2
```

```
incu_new2$p_age[incu_new2$p_age>36]<-36 # higher end of age spectrum
incu_new2$p_age[incu_new2$p_age<12]<-12 # lower end of age spectrum

# age quadratic variable conversion
incu_new2$age_sq <-poly(incu_new2$age,2, raw=TRUE)[,2]
incu_new2$p_age_sq <-poly(incu_new2$p_age,2, raw=TRUE)[,2]
```

Comparing the new samples size per age category with those above shows there are now at least five individuals in each of the most extreme age categories.

```
with(incu_new2, tapply(tag, age, FUN = function(x) length(unique(x))))
```

```
## 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
##  5  1  3  2  4  4  2  3  2  4  5  4  3  3  1  2  2  2  3  2  2  1  1  1  6
```

```
with(incu_new2, tapply(tag, p_age, FUN = function(x) length(unique(x))))
```

```
## 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
##  5  1  3  2  4  4  2  3  2  4  5  4  3  3  1  2  2  2  3  2  2  1  1  1  6
```

Step 3: All continuous variables are scaled (mean = 0, SD = 1).

```
#rescaling
incu_new2$age_sc<-scale(incu_new2$age)
incu_new2$p_age_sc<-scale(incu_new2$p_age)
incu_new2$age_sq_sc<-scale(incu_new2$age_sq)
incu_new2$p_age_sq_sc<-scale(incu_new2$p_age_sq)
incu_new2$pers_sc<-scale(incu_new2$pers)
incu_new2$num_days_sc<-scale(incu_new2$NumDays)
incu_new2$p_pers_sc<-scale(incu_new2$p_pers)
```

Step 4: For the purposes of visualisation, a categorical variables representing focal bird and partner boldness are created. Here, any individual whose scaled boldness score ( `pers_sc` or `p_pers_sc` ) is  $>0$  is considered bold, while those with  $< 0$  are considered shy. This variable is then confirmed as a factor in R. **In all analysis, boldness is represented by the continuous variables `pers_sc` and `p_pers_sc` and this categorical variable will only be used during graph creation (see below).**

step 5: In this important section, the partner's previous bout variable is refined to account for each individual's mean foraging trip duration. This helps us to ensure that the focal bird is responding to its partner's most recent trip, rather than their general foraging pattern. This is accomplished by creating a new variable which represents the differences between each partner's previous trip and their mean trip length. Step-by-step instructions for each line of code are included in the comments.

```
# creating final partner's previous bout variable
incu_aves<-incu_new2%>%group_by(cycle_tag)%>%summarise(ppb_ave_no_hours=mean
  (part_prev_bout)) # calculate mean trip duration for each bird in ea
  ch season
incu_avesdf<- data.frame(incu_aves) # convert this into a data frame
incu_new2<-merge(incu_new2, incu_avesdf, by.x = "cycle_tag", by.y="cycle_tag"
  ) # add mean to original data set
incu_new2<-incu_new2%>%mutate(ppb_ave_diff=part_prev_bout-ppb_ave_no_hours) #
  subtract raw partners previous bout from mean
incu_new2$ppb_ave_diff_sc<-scale(incu_new2$ppb_ave_diff) # scale new variable
incu_new2%>%summarise(average=mean(ppb_ave_diff),med=median(ppb_ave_diff), sd
  =sd(ppb_ave_diff))
```

```
##          average      med      sd
## 1 -5.224453e-16 -2.45832 75.10486
```

## Variable Refinement: Brooding

Steps 1-5 above are then repeated, this time using the brooding data subset.

Step 1: a new data frame is created:

```
brood_new2<-na.omit(brood_new)# check to ensure no NAs
```

Step 2: the quadratic age and partner age variables are created.

```
# collapse age groups and creating quadratic variable brooding
with(brood_new2, tapply(tag, age, FUN = function(x) length(unique(x))))
```

```
##  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
32
##  2  3  1  9  4  8  7 10  6  6  7  5  7  4 11  8  6  7  3  4  6  4  5  5  2
3
## 33 34 35 37 40
##  1  2  1  2  1
```

```
brood_new2$age[brood_new2$age>34]<-34 # higher end of age spectrum
brood_new2$age[brood_new2$age<8]<-8 # lower end of age spectrum

with(brood_new2, tapply(tag, p_age, FUN = function(x) length(unique(x))))
```

```
##  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
32
##  2  3  1  9  4  8  7 10  6  6  7  5  7  4 11  8  6  7  3  4  6  4  5  5  2
3
## 33 34 35 37 40
##  1  2  1  2  1
```

```
brood_new2$p_age[brood_new2$p_age>34]<-34 # higher end of age spectrum
brood_new2$p_age[brood_new2$p_age<8]<-8 # lower end of age spectrum

# age quadratic variable conversion
brood_new2$age_sq <-poly(brood_new2$age,2, raw=TRUE)[,2]
brood_new2$p_age_sq <-poly(brood_new2$p_age,2, raw=TRUE)[,2]
```

and checked:

```
with(brood_new2, tapply(tag, age, FUN = function(x) length(unique(x))))
```



```
## 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
33
## 5 1 9 4 8 7 10 6 6 7 5 7 4 11 8 6 7 3 4 6 4 5 5 2 3
1
## 34
## 6
```

```
with(brood_new2, tapply(tag, p_age, FUN = function(x) length(unique(x))))
```

```
## 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
33
## 5 1 9 4 8 7 10 6 6 7 5 7 4 11 8 6 7 3 4 6 4 5 5 2 3
1
## 34
## 6
```

Step 3: the continuous variables are scaled.

```
#rescaling
brood_new2$age_sc<-scale(brood_new2$age)
brood_new2$p_age_sc<-scale(brood_new2$p_age)
brood_new2$age_sq_sc<-scale(brood_new2$age_sq)
brood_new2$p_age_sq_sc<-scale(brood_new2$p_age_sq)
brood_new2$pers_sc<-scale(brood_new2$pers)
brood_new2$p_pers_sc<-scale(brood_new2$p_pers)
brood_new2$num_days_sc<-scale(brood_new2$NumDays)
```

Step 4: The categorical boldness variable is created and set aside for later.

Step 5: the new variable representing partner's previous trip is created.

```
# creating final partner's previous bout variable
brood_aves<-brood_new2%>%group_by(cycle_tag)%>%summarise(ppb_ave_no_hours=mean(
  part_prev_bout)) # calculate each individual's mean trip duration
brood_avesdf<- data.frame(brood_aves) # converted into a dataframe
brood_new2<-merge(brood_new2, brood_avesdf, by.x = "cycle_tag", by.y="cycle_tag") # add new mean to original data set
brood_new2<-brood_new2%>%mutate(ppb_ave_diff=part_prev_bout-ppb_ave_no_hours)
# subtract previous partner bout from mean
brood_new2$ppb_ave_diff_sc<-scale(brood_new2$ppb_ave_diff) # scale variable
brood_new2%>%summarise(average=mean(ppb_ave_diff),med=median(ppb_ave_diff), sd=sd(ppb_ave_diff))
```

```
##          average      med      sd
## 1 2.729535e-16 0.20832 27.0139
```

## Assumption Testing

Both data subsets are tested individually to ensure they meet the assumptions required by linear mixed modeling. This involves examining:

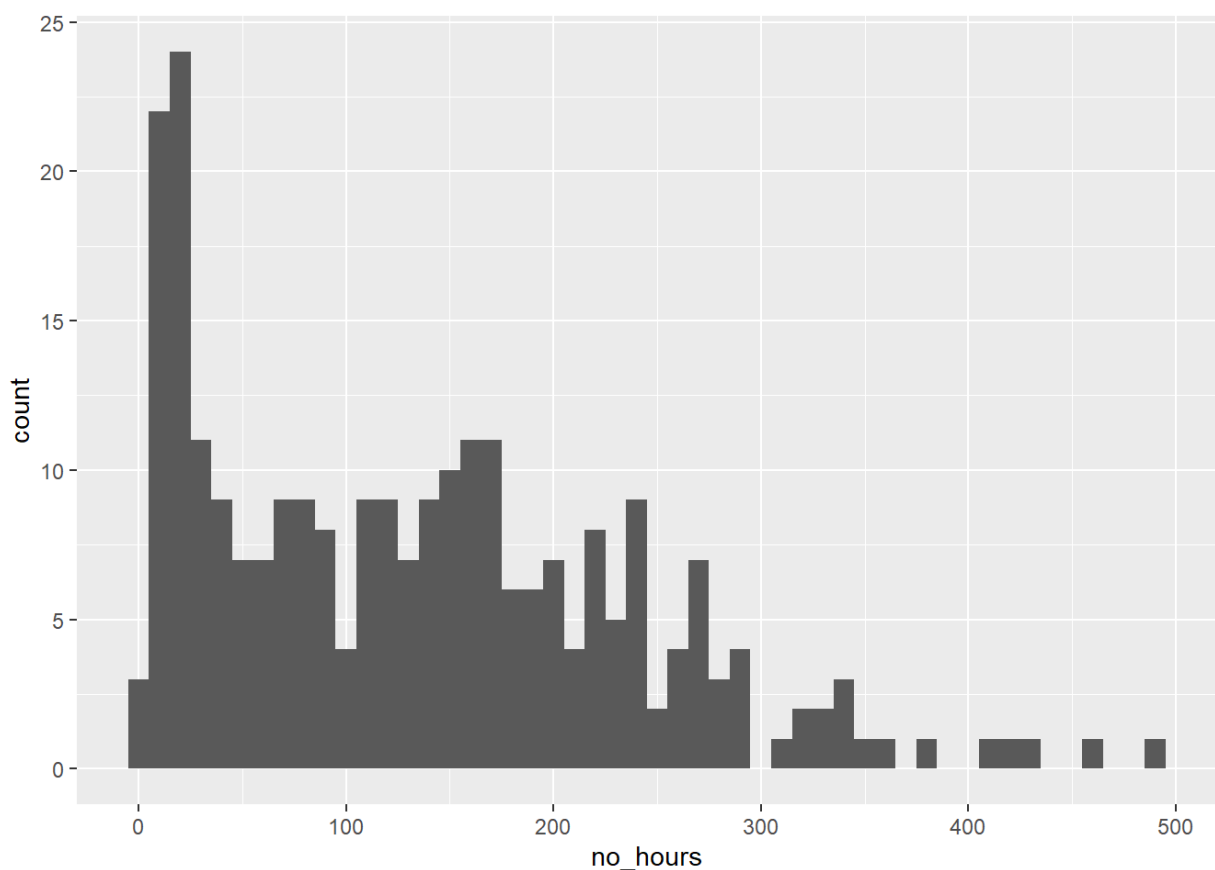
The normality in the response variable Homogeneity of variance Multicollinearity

## Assumption Testing: Incubation

As in the variable refinement section, the assumption tests for incubation will be labeled as steps 1-3 to make comparison with the brooding code easier to follow.

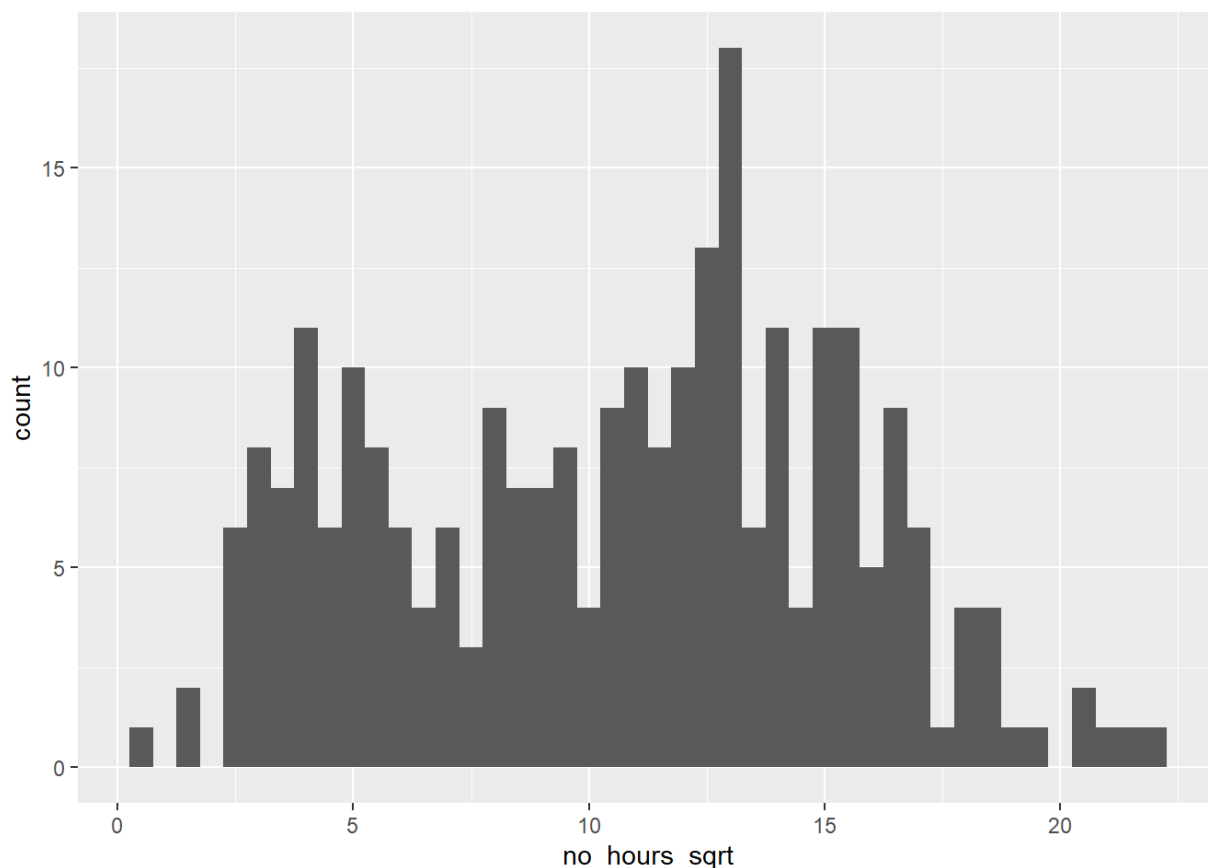
Step 1: the response variable representing the length of the focal bird's foraging trip ( `no_hours` ) is not normally distributed and requires transformation. This can be seen using an initial histogram and is corrected by creating a square root variable ( `no_hours_sqrt` ).

```
#normality
# no_hours is non-normal requires transformation
histogram_hours_sqrt<-incu_new2 %>%ggplot(aes(x=no_hours))+
  geom_histogram (binwidth = 10)
histogram_hours_sqrt
```



```
# creating new sqrt response variable
incu_new2$no_hours_sqrt<-sqrt(incu_new2$no_hours)

#check success
histogram_hours_sqrt<-incu_new2 %>%ggplot(aes(x=no_hours_sqrt))+
  geom_histogram (binwidth = 0.5)
histogram_hours_sqrt
```



The variable `no_hours_sqrt` will be used as the response variable in the models.

Step 2: the following code is used to check each variable for homogeneity of variance. Output plots are omitted here for brevity.

```
plot(lm(no_hours_sqrt~pers_sc,data=incu_new2))
```

```
plot(lm(no_hours_sqrt~p_pers_sc,data=incu_new2))
```

```
plot(lm(no_hours_sqrt~age_sc,data=incu_new2))
```

```
plot(lm(no_hours_sqrt~age_sq_sc,data=incu_new2))
```

```
plot(lm(no_hours_sqrt~p_age_sc,data=incu_new2))
```

```
plot(lm(no_hours_sqrt~p_age_sq_sc,data=incu_new2))
```

```
plot(lm(no_hours_sqrt~ppb_ave_diff_sc,data=incu_new2))
```

Step 3: this code searches for evidence of multicollinearity between potentially correlated continuous fixed variables. VIF scores > 3 are considered problematic. Code for the production of graphs has been provided by outputs are omitted for brevity.

```
cor.test(incu_new2$age_sc, incu_new2$pers_sc) # view correlation
```

```
##
## Pearson's product-moment correlation
##
## data: incu_new2$age_sc and incu_new2$pers_sc
## t = 0.51054, df = 258, p-value = 0.6101
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.09023343 0.15283197
## sample estimates:
## cor
## 0.03176897
```

```
col_g<-incu_new2%>%ggplot(aes(x =age_sc , y =pers_sc))+
  geom_point(size=2)
col_g
```

```
colin_model <- lm(no_hours_sqrt ~ age_sc +pers_sc, data = incu_new2)
summary(colin_model)
```

```
##
## Call:
## lm(formula = no_hours_sqrt ~ age_sc + pers_sc, data = incu_new2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.0912  -3.9329   0.5991   3.5304  11.0185
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  10.5686     0.2950  35.830  <2e-16 ***
## age_sc       -0.4280     0.2957  -1.447   0.149
## pers_sc       0.3884     0.2957   1.314   0.190
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.756 on 257 degrees of freedom
## Multiple R-squared:  0.01421,    Adjusted R-squared:  0.006535
## F-statistic: 1.852 on 2 and 257 DF,  p-value: 0.159
```

```
vif(colin_model) #>3 = problematic
```

```
## age_sc pers_sc
## 1.00101 1.00101
```

```
#age and ppb
cor.test(incu_new2$age_sc, incu_new2$ppb_ave_diff_sc) #view correlation
```

```
##
## Pearson's product-moment correlation
##
## data: incu_new2$age_sc and incu_new2$ppb_ave_diff_sc
## t = 3.595e-16, df = 258, p-value = 1
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.1216537 0.1216537
## sample estimates:
## cor
## 2.238166e-17
```

```
col_g<-incu_new2%>%ggplot(aes(x =age_sc , y =ppb_ave_diff_sc))+
  geom_point(size=2)
col_g
```

```
colin_model <- lm(no_hours_sqrt ~ age_sc +ppb_ave_diff_sc, data = incu_new2)
summary(colin_model)
```

```
##
## Call:
## lm(formula = no_hours_sqrt ~ age_sc + ppb_ave_diff_sc, data = incu_new2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.1331  -3.6887   0.4586   3.5538  10.1394
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    10.5686     0.2890  36.571 < 2e-16 ***
## age_sc         -0.4157     0.2895  -1.436 0.152350
## ppb_ave_diff_sc  1.0252     0.2895   3.541 0.000473 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.66 on 257 degrees of freedom
## Multiple R-squared:  0.05375,    Adjusted R-squared:  0.04639
## F-statistic: 7.299 on 2 and 257 DF,  p-value: 0.0008254
```

```
vif(colin_model) #>3 = problematic
```

```
##          age_sc ppb_ave_diff_sc
##              1              1
```

```
#age and p_age
cor.test(incu_new2$age_sc, incu_new2$p_age_sc) # view correlation
```

```
##
## Pearson's product-moment correlation
##
## data: incu_new2$age_sc and incu_new2$p_age_sc
## t = 14.636, df = 258, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.6011335 0.7349642
## sample estimates:
## cor
## 0.6735318
```

```
col_g<-incu_new2%>%ggplot(aes(x =age_sc , y =p_age_sc))+
  geom_point(size=2)
col_g
```

```
colin_model <- lm(no_hours_sqrt ~ age_sc +p_age_sc, data = incu_new2)
summary(colin_model)
```

```
##
## Call:
## lm(formula = no_hours_sqrt ~ age_sc + p_age_sc, data = incu_new2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.1740  -4.2080   0.6748   3.5055  10.9966
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  10.5686     0.2957  35.736  <2e-16 ***
## age_sc       -0.5777     0.4009  -1.441   0.151
## p_age_sc      0.2406     0.4009   0.600   0.549
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.769 on 257 degrees of freedom
## Multiple R-squared:  0.008977, Adjusted R-squared: 0.001265
## F-statistic: 1.164 on 2 and 257 DF, p-value: 0.3139
```

```
vif(colin_model) #>3 = problematic
```

```
## age_sc p_age_sc
## 1.830312 1.830312
```

```
#pers and p_pers
cor.test(incu_new2$pers_sc, incu_new2$p_pers_sc) # view correlation
```

```
##
## Pearson's product-moment correlation
##
## data: incu_new2$pers_sc and incu_new2$p_pers_sc
## t = -9.6713, df = 258, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.5998380 -0.4205629
## sample estimates:
## cor
## -0.5158254
```

```
col_g<-incu_new2%>%ggplot(aes(x =age_sc , y =p_age_sc))+
  geom_point(size=2)
col_g
```

```
colin_model <- lm(no_hours_sqrt ~ pers_sc +p_pers_sc, data = incu_new2)
summary(colin_model)
```

```
##
## Call:
## lm(formula = no_hours_sqrt ~ pers_sc + p_pers_sc, data = incu_new2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.0624  -3.8970   0.6179   3.5531  11.4408
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  10.56858    0.29401  35.946  <2e-16 ***
## pers_sc       0.03046    0.34386   0.089   0.9295
## p_pers_sc    -0.66756    0.34386  -1.941   0.0533 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.741 on 257 degrees of freedom
## Multiple R-squared:  0.02053,    Adjusted R-squared:  0.01291
## F-statistic: 2.694 on 2 and 257 DF,  p-value: 0.06953
```

```
vif(colin_model) #>3 = problematic
```

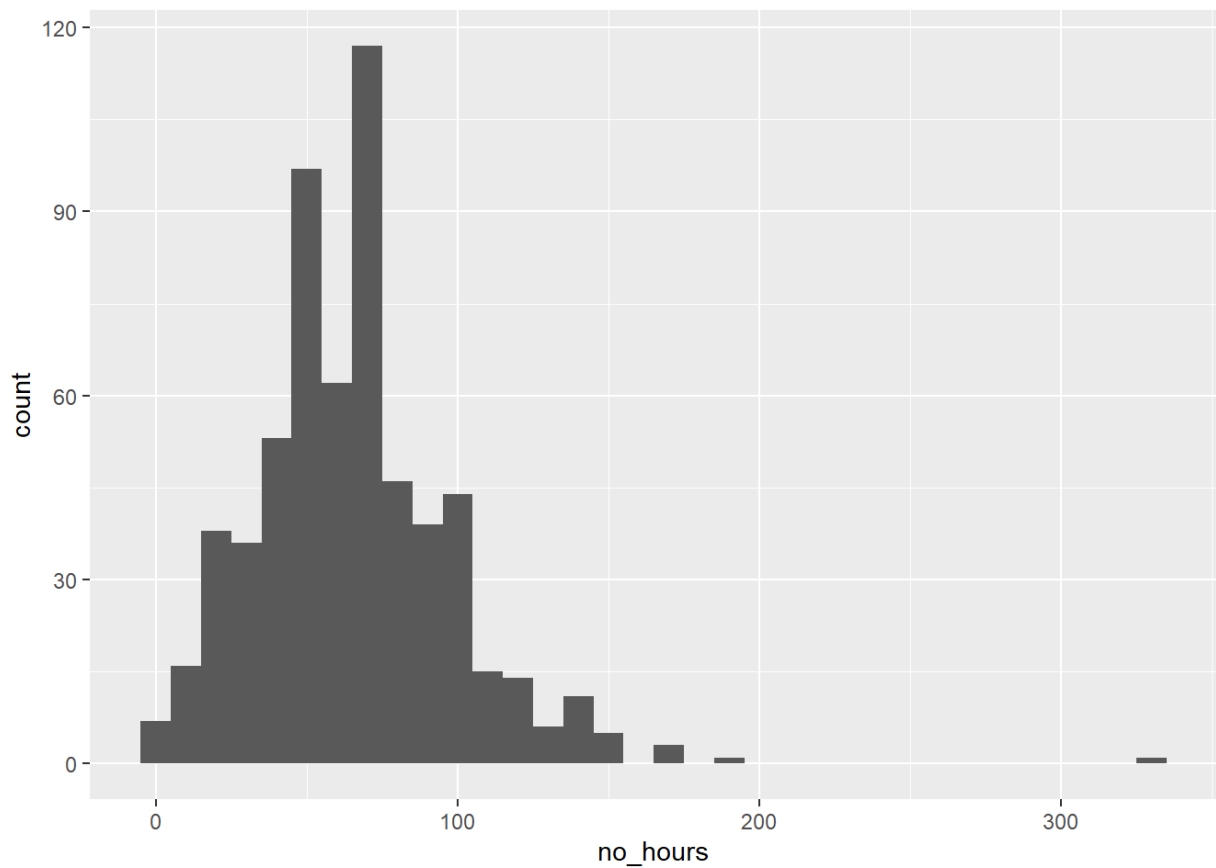
```
## pers_sc p_pers_sc
## 1.362539 1.362539
```

## Assumption Testing: Brooding

Once again, steps 1-3 above are repeated this time for the brooding subset.

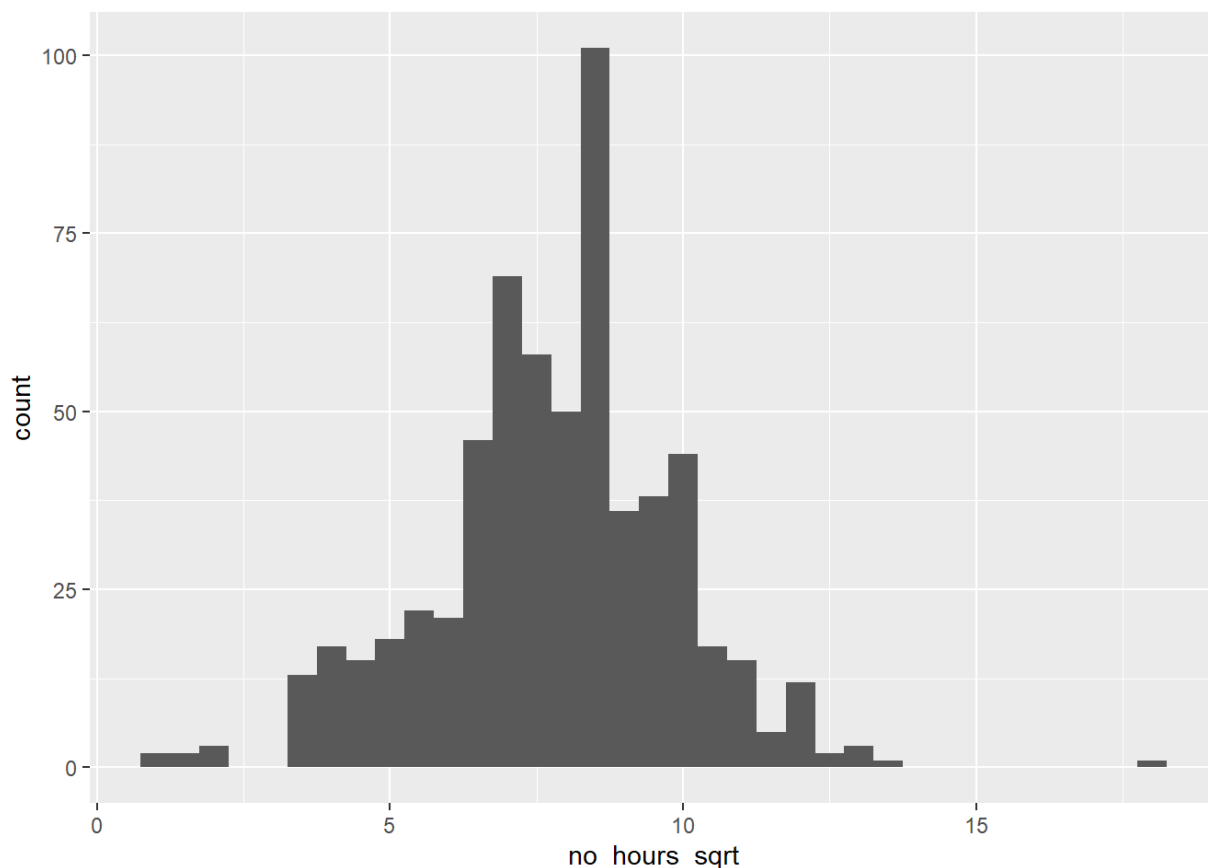
Step 1: transformation of the response variable into `no_hour_sqrt`.

```
#no_hour is non normal and requires transformation  
histogram_hours_sqrt<-brood_new2 %>%ggplot(aes(x=no_hours))+  
  geom_histogram (binwidth = 10)  
histogram_hours_sqrt
```



```
# creating new sqrt response variable  
brood_new2$no_hours_sqrt<-sqrt(brood_new2$no_hours)  
  
# check success  
histogram_hours_sqrt<-brood_new2 %>%ggplot(aes(x=no_hours_sqrt))+  
  geom_histogram (binwidth = 0.5)  
histogram_hours_sqrt
```





The variable `no_hours_sqrt` will be used as the response variable in the models.

Step 2: check each variable for homogeneity of variance. Output plots are omitted here for brevity.

```
plot(lm(no_hours_sqrt~pers_sc,data=brood_new2))
```

```
plot(lm(no_hours_sqrt~age_sc,data=brood_new2))
```

```
plot(lm(no_hours_sqrt~age_sq_sc,data=brood_new2))
```

```
plot(lm(no_hours_sqrt~p_pers_sc,data=brood_new2))
```

```
plot(lm(no_hours_sqrt~p_age_sc,data=brood_new2))
```

```
plot(lm(no_hours_sqrt~p_age_sq_sc,data=brood_new2))
```

```
plot(lm(no_hours_sqrt~ppb_ave_diff_sc,data=brood_new2))
```

Step 3: examine evidence of multicollinearity between potentially correlated continuous fixed variables. VIF scores > 3 are considered problematic. Code for the production of graphs has been provided by outputs are omitted for brevity.

```
#pers and age
cor.test(brood_new2$age_sc, brood_new2$pers_sc)
```

```
##
## Pearson's product-moment correlation
##
## data: brood_new2$age_sc and brood_new2$pers_sc
## t = 3.7603, df = 609, p-value = 0.0001861
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.07217681 0.22723939
## sample estimates:
## cor
## 0.1506345
```

```
col_g<-brood_new2%>%ggplot(aes(x =age_sc , y =pers_sc))+
  geom_point(size=2)
col_g
```

```
colin_model <- lm(no_hours_sqrt ~ age_sc +pers_sc, data = brood_new2)
summary(colin_model)
```

```
##
## Call:
## lm(formula = no_hours_sqrt ~ age_sc + pers_sc, data = brood_new2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.6498 -1.0646  0.1118  1.2052 10.2963
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.816415   0.082708  94.506 < 2e-16 ***
## age_sc       0.004899   0.083731   0.059  0.95336
## pers_sc     -0.241064   0.083731  -2.879  0.00413 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.044 on 608 degrees of freedom
## Multiple R-squared:  0.01368, Adjusted R-squared:  0.01044
## F-statistic: 4.216 on 2 and 608 DF, p-value: 0.01519
```

```
library(car)
vif(colin_model) #>3 = problematic
```

```
## age_sc pers_sc
## 1.023218 1.023218
```

```
#age and ppb
cor.test(brood_new2$age_sc, brood_new2$ppb_ave_diff_sc)
```

```
##
## Pearson's product-moment correlation
##
## data: brood_new2$age_sc and brood_new2$ppb_ave_diff_sc
## t = -8.1004e-17, df = 609, p-value = 1
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.07932005 0.07932005
## sample estimates:
## cor
## -3.282467e-18
```

```
col_g<-brood_new2%>%ggplot(aes(x =age_sc , y =ppb_ave_diff_sc))+
  geom_point(size=2)
col_g
```

```
colin_model <- lm(no_hours_sqrt ~ age_sc +ppb_ave_diff_sc, data = brood_new2)
summary(colin_model)
```

```
##
## Call:
## lm(formula = no_hours_sqrt ~ age_sc + ppb_ave_diff_sc, data = brood_new2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.9504 -1.1158  0.2231  1.1766 10.1607
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    7.81642    0.08297   94.208  <2e-16 ***
## age_sc         -0.03141    0.08304   -0.378  0.7053
## ppb_ave_diff_sc 0.17436    0.08304    2.100  0.0362 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.051 on 608 degrees of freedom
## Multiple R-squared:  0.007431, Adjusted R-squared:  0.004166
## F-statistic: 2.276 on 2 and 608 DF, p-value: 0.1036
```

```
library(car)
vif(colin_model) #>3 = problematic
```

```
##          age_sc ppb_ave_diff_sc
##          1          1
```

```
#age and p_age
cor.test(brood_new2$age_sc, brood_new2$p_age_sc)
```

```
##
## Pearson's product-moment correlation
##
## data: brood_new2$age_sc and brood_new2$p_age_sc
## t = 18.09, df = 609, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5370693 0.6404886
## sample estimates:
##          cor
## 0.5912039
```

```
col_g<-brood_new2%>%ggplot(aes(x =age_sc , y =p_age_sc))+
  geom_point(size=2)
col_g
```

```
colin_model <- lm(no_hours_sqrt ~ age_sc +p_age_sc, data = brood_new2)
summary(colin_model)
```

```
##
## Call:
## lm(formula = no_hours_sqrt ~ age_sc + p_age_sc, data = brood_new2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.8015 -1.1213  0.2078  1.1079 10.0146
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  7.81642    0.08315  94.005  <2e-16 ***
## age_sc       -0.11262    0.10318  -1.091   0.276
## p_age_sc      0.13735    0.10318   1.331   0.184
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.055 on 608 degrees of freedom
## Multiple R-squared:  0.003139,    Adjusted R-squared:  -0.0001402
## F-statistic: 0.9572 on 2 and 608 DF,  p-value: 0.3845
```

```
library(car)
vif(colin_model) #>3 = problematic
```

```
## age_sc p_age_sc
## 1.537331 1.537331
```

```
#pers and p_pers
cor.test(brood_new2$pers_sc, brood_new2$p_pers_sc)
```

```
##
## Pearson's product-moment correlation
##
## data: brood_new2$pers_sc and brood_new2$p_pers_sc
## t = -1.2389, df = 609, p-value = 0.2159
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.12894722 0.02929656
## sample estimates:
## cor
## -0.05014001
```

```
col_g<-brood_new2%>%ggplot(aes(x =age_sc , y =p_age_sc))+
  geom_point(size=2)
col_g
```

```
colin_model <- lm(no_hours_sqrt ~ pers_sc + p_pers_sc, data = brood_new2)
summary(colin_model)
```

```
##
## Call:
## lm(formula = no_hours_sqrt ~ pers_sc + p_pers_sc, data = brood_new2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.8613 -1.1276  0.1368  1.2093 10.0270
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   7.81642    0.08204  95.277  < 2e-16 ***
## pers_sc       -0.25334    0.08221  -3.082  0.00215 **
## p_pers_sc      -0.25958    0.08221  -3.158  0.00167 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.028 on 608 degrees of freedom
## Multiple R-squared:  0.02959,    Adjusted R-squared:  0.0264
## F-statistic: 9.269 on 2 and 608 DF,  p-value: 0.0001083
```

```
library(car)
vif(colin_model) #>3 = problematic
```

```
##   pers_sc p_pers_sc
##   1.00252  1.00252
```

## Create a total dataset and save files

A combined data set of both the incubation and brooding data is created for use later (visualisation)

```
# a combined data set of both incubation and brooding data is created
total_df4<-full_join(incu_new2, brood_new2)
```

Sample size and centrality measures can be calculated for the record.

```
# sample sizes of total data set for the record
unique(total_df4$cycle_pair) # 95 breeding attempts
unique(total_df4$pair_id)
unique(total_df4$tag)# 142 birds, 71 pairs
unique(total_df4$cycle) #6 breeding seasons
unique(total_df4$age) #no ages
with(total_df4, tapply(pair_id, cycle, FUN = function(x) length(unique(x))))
# pair sample size per year
with(total_df4, tapply(tag, age, FUN = function(x) length(unique(x))))
total_df4$for_trip<-total_df4$part_prev_bout

# means, medians and SD full dataset
mean(total_df4$no_hours)
sd(total_df4$no_hours)
mean(total_df4$age)
sd(total_df4$age)
total_df4%>%group_by(sex)%>%summarise(average=mean(no_hours),med=median(no_hours), sd=sd(no_hours))
total_df4%>%group_by(partner_type)%>%summarise(average=mean(no_hours),med=median(no_hours), sd=sd(no_hours))
total_df4%>%group_by(breed_stage)%>%summarise(average=mean(no_hours),med=median(no_hours), sd=sd(no_hours))
total_df4%>%group_by(cycle)%>%summarise(average=mean(no_hours),med=median(no_hours), sd=sd(no_hours))
```

All files (incubation, brooding and total observations) are saved for later use in graph creation.

```
write.csv(incu_new2, "incu_new2.csv")# incubation data
write.csv(brood_new2, "brood_new2.csv")# brooding data
write.csv(total_df4, "total_df4.csv") # all (combined) data
```

## Linear Mixed Models and Model Selection

The following code outlines how the linear mixed models (LMMs) were created and how information theory (specifically AIC and model averaging) was applied for the purpose of model selection. Two models, one for the incubation subset and one for the brooding subset, were created. Steps 1-6 will be used to aid comparison between models.

### Incubation Models

Step 1: As suggested by Bolker et al (2009), a basic linear model was created ahead of the LMM.

```
lin_model <- lm(no_hours_sqrt ~
               sex +
               pers_sc +
               p_pers_sc+
               age_sc +
               age_sq_sc +
               p_age_sc+
               p_age_sq_sc+
               num_days_sc+
               new_part+
               ppb_ave_diff_sc+
               age_sc:pers_sc+
               age_sq_sc:pers_sc+
               age_sc:ppb_ave_diff_sc+
               age_sq_sc:ppb_ave_diff_sc+
               pers_sc:ppb_ave_diff_sc+
               new_part:ppb_ave_diff_sc+
               p_age_sc:p_pers_sc+
               p_age_sq_sc:p_pers_sc+
               p_age_sc:ppb_ave_diff_sc+
               p_age_sq_sc:ppb_ave_diff_sc+
               p_pers_sc:ppb_ave_diff_sc,
               data = incu_new2)
summary(lin_model)
```



```
##
## Call:
## lm(formula = no_hours_sqrt ~ sex + pers_sc + p_pers_sc + age_sc +
##      age_sq_sc + p_age_sc + p_age_sq_sc + num_days_sc + new_part +
##      ppb_ave_diff_sc + age_sc:pers_sc + age_sq_sc:pers_sc + age_sc:ppb_ave_
##      diff_sc +
##      age_sq_sc:ppb_ave_diff_sc + pers_sc:ppb_ave_diff_sc + new_part:ppb_ave_
##      _diff_sc +
##      p_age_sc:p_pers_sc + p_age_sq_sc:p_pers_sc + p_age_sc:ppb_ave_diff_sc
##      +
##      p_age_sq_sc:ppb_ave_diff_sc + p_pers_sc:ppb_ave_diff_sc,
##      data = incu_new2)
##
## Residuals:
##      Min        1Q    Median        3Q        Max
## -10.7661  -2.6938  -0.0042   2.8126   9.4865
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    11.31936    0.41012  27.600 < 2e-16 ***
## sexMale        -1.38103    0.61076  -2.261  0.0247 *
## pers_sc        -0.56541    0.33079  -1.709  0.0887 .
## p_pers_sc      -0.76699    0.33081  -2.319  0.0213 *
## age_sc         -0.66229    2.18281  -0.303  0.7618
## age_sq_sc       0.26480    2.19916   0.120  0.9043
## p_age_sc       4.38323    2.16385   2.026  0.0439 *
## p_age_sq_sc    -4.36434    2.17651  -2.005  0.0461 *
## num_days_sc     1.88731    0.29063   6.494 4.83e-10 ***
## new_part1       0.87002    1.84494   0.472  0.6377
## ppb_ave_diff_sc  0.37543    0.29298   1.281  0.2013
## pers_sc:age_sc   1.55841    2.70007   0.577  0.5644
## pers_sc:age_sq_sc -1.53100    2.66986  -0.573  0.5669
## age_sc:ppb_ave_diff_sc  0.08925    2.25404   0.040  0.9684
## age_sq_sc:ppb_ave_diff_sc -0.16008    2.28189  -0.070  0.9441
## pers_sc:ppb_ave_diff_sc -0.14404    0.37359  -0.386  0.7002
## new_part1:ppb_ave_diff_sc  1.26446    1.34986   0.937  0.3498
## p_pers_sc:p_age_sc  3.23841    2.71773   1.192  0.2346
## p_pers_sc:p_age_sq_sc -3.51919    2.68086  -1.313  0.1905
## p_age_sc:ppb_ave_diff_sc -1.11107    1.91069  -0.582  0.5615
## p_age_sq_sc:ppb_ave_diff_sc  1.21024    1.97810   0.612  0.5412
## p_pers_sc:ppb_ave_diff_sc  0.16668    0.37205   0.448  0.6546
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.226 on 238 degrees of freedom
## Multiple R-squared:  0.2791, Adjusted R-squared:  0.2155
## F-statistic: 4.388 on 21 and 238 DF, p-value: 5.721e-09
```

Step 2: The global model is then created. It features fixed effects, relevant interactions and random intercepts and slopes (see methods for details). Optimix optimizer is included to aid convergence. The residuals are also checked at this point.

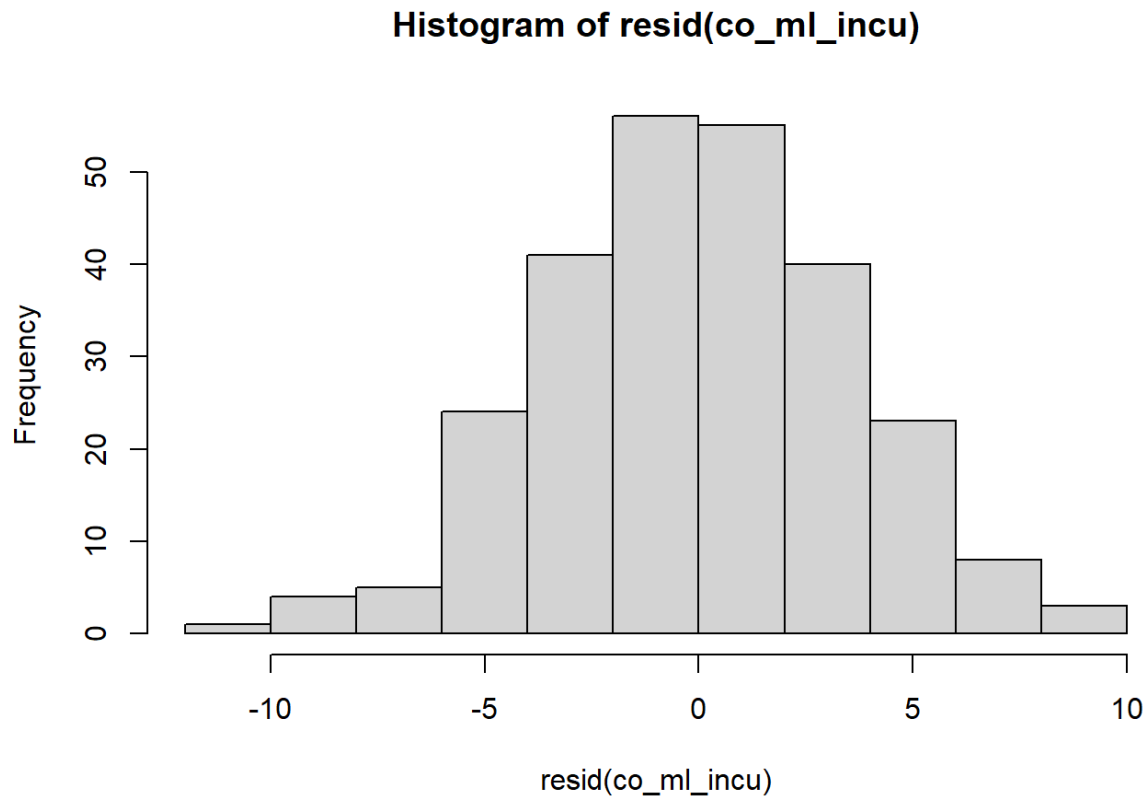
```
co_ml_incu<- lmer(no_hours_sqrt ~
  sex +
  pers_sc +
  p_pers_sc+
  age_sc +
  age_sq_sc +
  p_age_sc+
  p_age_sq_sc+
  num_days_sc+
  new_part+
  ppb_ave_diff_sc+
  age_sc:pers_sc+
  age_sq_sc:pers_sc+
  age_sc:ppb_ave_diff_sc+
  age_sq_sc:ppb_ave_diff_sc+
  pers_sc:ppb_ave_diff_sc+
  new_part:ppb_ave_diff_sc+
  p_age_sc:p_pers_sc+
  p_age_sq_sc:p_pers_sc+
  p_age_sc:ppb_ave_diff_sc+
  p_age_sq_sc:ppb_ave_diff_sc+
  p_pers_sc:ppb_ave_diff_sc+
  (1|cycle) + (1+ppb_ave_diff_sc|cycle_pair),
  data = incu_new2, na.action=na.fail, REML=FALSE,
  control=lmerControl (optimizer="optimx",optCtrl=list(method
='nlminb'))))

summary(co_ml_incu)
```

```
## Linear mixed model fit by maximum likelihood ['lmerMod']
## Formula: no_hours_sqrt ~ sex + pers_sc + p_pers_sc + age_sc + age_sq_sc +
##   p_age_sc + p_age_sq_sc + num_days_sc + new_part + ppb_ave_diff_sc +
##   age_sc:pers_sc + age_sq_sc:pers_sc + age_sc:ppb_ave_diff_sc +
##   age_sq_sc:ppb_ave_diff_sc + pers_sc:ppb_ave_diff_sc + new_part:ppb_ave
##   _diff_sc +
##   p_age_sc:p_pers_sc + p_age_sq_sc:p_pers_sc + p_age_sc:ppb_ave_diff_sc
##   +
##   p_age_sq_sc:ppb_ave_diff_sc + p_pers_sc:ppb_ave_diff_sc +
##   (1 | cycle) + (1 + ppb_ave_diff_sc | cycle_pair)
## Data: incu_new2
## Control: lmerControl(optimizer = "optimx", optCtrl = list(method = "nlnm
##   b"))
##
##      AIC      BIC   logLik deviance df.resid
##  1502.7   1598.8   -724.4   1448.7     233
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.76367 -0.66165 -0.01616  0.65268  2.29527
##
## Random effects:
##   Groups      Name                Variance Std.Dev. Corr
##   cycle_pair (Intercept)          2.718e+00 1.64852
##              ppb_ave_diff_sc      9.706e-04 0.03115  1.00
##   cycle      (Intercept)          0.000e+00 0.00000
##   Residual                                1.369e+01 3.69948
## Number of obs: 260, groups:  cycle_pair, 35; cycle, 7
##
## Fixed effects:
##
##              Estimate Std. Error t value
## (Intercept)      11.3666      0.4666  24.360
## sexMale          -1.3728      0.5349  -2.566
## pers_sc           -0.6381      0.4018  -1.588
## p_pers_sc         -0.8510      0.4021  -2.116
## age_sc            0.3762      2.3914   0.157
## age_sq_sc         -0.6816      2.3823  -0.286
## p_age_sc          5.2572      2.3680   2.220
## p_age_sq_sc       -5.1560      2.3569  -2.188
## num_days_sc       1.6158      0.3057   5.286
## new_part1         0.4495      2.4048   0.187
## ppb_ave_diff_sc    0.4545      0.2625   1.731
## pers_sc:age_sc     0.3351      3.2508   0.103
## pers_sc:age_sq_sc -0.3519      3.2477  -0.108
## age_sc:ppb_ave_diff_sc 0.1016      1.9733   0.051
## age_sq_sc:ppb_ave_diff_sc -0.1912      1.9978  -0.096
## pers_sc:ppb_ave_diff_sc -0.1397      0.3271  -0.427
## new_part1:ppb_ave_diff_sc 1.2360      1.1823   1.045
## p_pers_sc:p_age_sc 2.1787      3.2615   0.668
## p_pers_sc:p_age_sq_sc -2.4869      3.2549  -0.764
## p_age_sc:ppb_ave_diff_sc -1.1393      1.6729  -0.681
## p_age_sq_sc:ppb_ave_diff_sc 1.2493      1.7319   0.721
```

```
## p_pers_sc:ppb_ave_diff_sc    0.1599    0.3257    0.491
## optimizer (optimx) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

```
hist(resid(co_ml_incu))
```



Step 3: A model set is created from the global model. This creates all possible versions of the model to establish the best fit. As the focal bird age and its quadratic equivalent ( `age_sc` and `age_sq_sc` ) must be retained or discarded together, additional code is added to ensure this is reflected in the model set. Similarly, the partner bird age variable and its quadratic equivalent( `p_age_sc` and `p_age_sq_sc` ) receive the same treatment. This portion of the code may take several hours to run.

```
# dredging to create model set
# linking age variables with their quadratic counterpart so either both or ne
  iter are retained
dr_cmin<-dredge(co_ml_incu, subset = (dc(age_sc, age_sq_sc) &&
                                     dc(p_age_sc, p_age_sq_sc) &&
                                     dc(age_sq_sc, age_sc) &&
                                     dc(p_age_sq_sc, p_age_sc)))
dr_cmin # view full model list
```

Step 4: The models are ranked according to AIC. Next, those models with  $\Delta < 2$  are subsetted. Another piece of code indicates how many models have been retained following this action.

```
sub_m1_incu<-subset(dr_cmin, delta<2) ##subsets models delta <2  
sub_m1_incu # view subsetting models
```

```
## Global model call: lmer(formula = no_hours_sqrt ~ sex + pers_sc + p_pers_s
c + age_sc +
##      age_sq_sc + p_age_sc + p_age_sq_sc + num_days_sc + new_part +
##      ppb_ave_diff_sc + age_sc:pers_sc + age_sq_sc:pers_sc + age_sc:ppb_ave_
diff_sc +
##      age_sq_sc:ppb_ave_diff_sc + pers_sc:ppb_ave_diff_sc + new_part:ppb_ave_
_diff_sc +
##      p_age_sc:p_pers_sc + p_age_sq_sc:p_pers_sc + p_age_sc:ppb_ave_diff_sc
+
##      p_age_sq_sc:ppb_ave_diff_sc + p_pers_sc:ppb_ave_diff_sc +
##      (1 | cycle) + (1 + ppb_ave_diff_sc | cycle_pair), data = incu_new2,
##      REML = FALSE, control = lmerControl(optimizer = "optimx",
##      optCtrl = list(method = "nlnminb")), na.action = na.fail)
## ---
## Model selection table
##      (Int) num_dys_sc p_age_sc p_age_sq_sc p_pers_sc  pers_sc ppb_ave_dff
_sc
## 841      11.23      1.652                      -0.5041          0.4
809
## 132089 11.33      1.647      5.719      -5.652 -0.7965 -0.6592          0.4
773
## 1017    11.29      1.654      4.803      -4.772 -0.8627 -0.6124          0.4
739
## 33785   11.31      1.651      5.719      -5.639 -0.7937 -0.6538          0.4
757
## 889     11.20      1.591      4.261      -4.220 -0.4321          0.4
908
## 131961 11.23      1.581      5.084      -5.008 -0.3381          0.4
952
## 969     11.30      1.707                      -0.8554 -0.4789          0.4
657
## 825     11.35      1.557      4.739      -4.741          0.5
000
## 585     11.16      1.845                      -0.5096
## 777     11.41      1.622          0.4
892
## 33657   11.22      1.585      5.086      -4.998 -0.3390          0.4
935
## 131833 11.27      1.844      5.584      -5.515 -0.8105 -0.6697
## 761     11.23      1.845      4.698      -4.663 -0.8746 -0.6229
## 713     11.24      1.900                      -0.8680 -0.4935
## 33529   11.25      1.846      5.588      -5.506 -0.8074 -0.6644
## 1180665 11.33      1.655      5.719      -5.650 -0.8021 -0.6649          0.4
538
## 633     11.12      1.784      4.144      -4.099 -0.4408
## 525129 11.23      1.653                      -0.5039          0.4
889
## 164857 11.39      1.631      5.352      -5.366 -0.8369 -0.6678          0.4
842
## 131705 11.15      1.778      4.934      -4.855 -0.3495
## 569     11.28      1.752      4.618      -4.614
## 521     11.34      1.816
```

```
## 1049593 11.29      1.661    4.799      -4.767  -0.8684 -0.6179      0.4
518
##          sex p_age_sc:p_prs_sc p_age_sq_sc:p_prs_sc p_prs_sc:ppb_ave_dff_sc
## 841      +
## 132089   +
##          -0.4172
## 1017     +
## 33785    +
##          -0.4135
## 889      +
## 131961   +
##          -0.3874
## 969      +
## 825      +
## 585      +
## 777      +
## 33657    +
##          -0.3840
## 131833   +
##          -0.4102
## 761      +
## 713      +
## 33529    +
##          -0.4076
## 1180665  +
##          -0.4177
## 633      +
## 525129   +
##          0.1674
## 164857   +
##          2.0750
##          -2.3450
## 131705   +
##          -0.3790
## 569      +
## 521      +
## 1049593  +
##          prs_sc:ppb_ave_dff_sc df    logLik    AICc delta weight
## 841              10 -730.731 1482.3  0.00  0.077
## 132089           14 -726.473 1482.7  0.31  0.066
## 1017             13 -727.747 1483.0  0.63  0.056
## 33785            14 -726.653 1483.0  0.67  0.055
## 889              12 -728.906 1483.1  0.73  0.053
## 131961           13 -727.814 1483.1  0.76  0.052
## 969              11 -730.046 1483.2  0.81  0.051
## 825              11 -730.055 1483.2  0.83  0.051
## 585               9 -732.228 1483.2  0.83  0.051
## 777               9 -732.322 1483.4  1.02  0.046
## 33657            13 -727.969 1483.4  1.07  0.045
## 131833           13 -728.054 1483.6  1.24  0.041
## 761              12 -729.274 1483.8  1.47  0.037
## 713              10 -731.491 1483.9  1.52  0.036
## 33529            13 -728.220 1483.9  1.57  0.035
## 1180665          -0.2701 15 -726.027 1484.0  1.67  0.033
## 633              11 -730.485 1484.0  1.69  0.033
## 525129           11 -730.528 1484.1  1.77  0.032
## 164857           15 -726.080 1484.1  1.78  0.031
## 131705           12 -729.450 1484.2  1.82  0.031
## 569             10 -731.670 1484.2  1.88  0.030
## 521              8 -733.836 1484.2  1.90  0.030
## 1049593          -0.2689 14 -727.308 1484.3  1.98  0.028
## Models ranked by AICc(x)
## Random terms (all models):
## '1 | cycle', '1 + ppb_ave_diff_sc | cycle_pair'
```

```
no_m1_incu<-dim(as.data.frame(sub_m1_incu))[1] # how many models are in the s
ubsetted model set
no_m1_incu # number of models retained
```

```
## [1] 23
```

Step 5: The nested models are removed (see methods for justification).

```
nest_m1_incu <- subset(sub_m1_incu, !nested(.)) # removes nested models
nest_m1_incu # view retained models
```

```
## Global model call: lmer(formula = no_hours_sqrt ~ sex + pers_sc + p_pers_s
c + age_sc +
##   age_sq_sc + p_age_sc + p_age_sq_sc + num_days_sc + new_part +
##   ppb_ave_diff_sc + age_sc:pers_sc + age_sq_sc:pers_sc + age_sc:ppb_ave_
diff_sc +
##   age_sq_sc:ppb_ave_diff_sc + pers_sc:ppb_ave_diff_sc + new_part:ppb_ave
_diff_sc +
##   p_age_sc:p_pers_sc + p_age_sq_sc:p_pers_sc + p_age_sc:ppb_ave_diff_sc
+
##   p_age_sq_sc:ppb_ave_diff_sc + p_pers_sc:ppb_ave_diff_sc +
##   (1 | cycle) + (1 + ppb_ave_diff_sc | cycle_pair), data = incu_new2,
##   REML = FALSE, control = lmerControl(optimizer = "optimx",
##     optCtrl = list(method = "nlnminb")), na.action = na.fail)
## ---
## Model selection table
##   (Int) num_dys_sc p_age_sc p_age_sq_sc p_prs_sc ppb_ave_dff_sc sex df
## 841 11.23      1.652      4.739      -4.741      0.5041      0.4809 + 10
## 825 11.35      1.557      4.739      -4.741      0.5000      0.5000 + 11
## 585 11.16      1.845      4.739      -0.5096      0.4892 + 9
## 777 11.41      1.622      4.618      -4.614      0.4892 + 9
## 569 11.28      1.752      4.618      -4.614      0.4892 + 10
## 521 11.34      1.816      4.618      -4.614      0.4892 + 8
##   logLik   AICc delta weight
## 841 -730.731 1482.3  0.00  0.270
## 825 -730.055 1483.2  0.83  0.179
## 585 -732.228 1483.2  0.83  0.178
## 777 -732.322 1483.4  1.02  0.163
## 569 -731.670 1484.2  1.88  0.106
## 521 -733.836 1484.2  1.90  0.105
## Models ranked by AICc(x)
## Random terms (all models):
## '1 | cycle', '1 + ppb_ave_diff_sc | cycle_pair'
```

```
no_m12_incu<-dim(as.data.frame(nest_m1_incu))[1] # how many models retained w
hen nested models removed
no_m12_incu # number of models
```



```
## [1] 6
```

```
# see all coefficients for the top model set
in_coeff <- coef(nest_m1_incu, full=TRUE)
in_coeff
```

```
##      (Intercept) num_days_sc  p_pers_sc ppb_ave_diff_sc  sexMale p_age_sc
## 841      11.22709      1.652019 -0.5040743      0.4808967 -1.207430      NA
## 825      11.35491      1.557058      NA      0.4999803 -1.481906  4.739413
## 585      11.16014      1.844831 -0.5095877      NA -1.213491      NA
## 777      11.41431      1.621791      NA      0.4891824 -1.600663      NA
## 569      11.27661      1.751839      NA      NA -1.495424  4.618492
## 521      11.34269      1.815789      NA      NA -1.611292      NA
##      p_age_sq_sc
## 841      NA
## 825     -4.741028
## 585      NA
## 777      NA
## 569     -4.614189
## 521      NA
```

```
nest_m1_dfin<- as.data.frame(in_coeff)
write.csv(nest_m1_dfin, file = "incubation_models.csv")
```

Step 6: The remaining then undergo model averaging to create a final coefficient table.

```
modavm1_incu<-model.avg(nest_m1_incu, delta<2, fit = TRUE) # model averaging
modavm1_incu
```

```
##
## Call:
## model.avg(object = get.models(object = nest_m1_incu, subset = delta <
##      2))
##
## Component models:
## '1456' '12356' '146' '156' '1236' '16'
##
## Coefficients:
##      (Intercept) num_days_sc  p_pers_sc ppb_ave_diff_sc  sexMale p_age_
##      sc
## full      11.28571      1.692215 -0.2271670      0.2987677 -1.394078  1.3344
## 51
## subset    11.28571      1.692215 -0.5062667      0.4886737 -1.394078  4.6944
## 64
##      p_age_sq_sc
## full      -1.334284
## subset    -4.693879
```

```
Weights(modavm1_incu)
```

```
## AICc model weights
## [1] 0.270 0.179 0.178 0.163 0.106 0.105
```

```
tablem1a_incu<-coefTable(modavm1_incu, full=TRUE)
tablem1a_incu # final coefficient table
```

```
##              Estimate Std. Error
## (Intercept)   11.28571     0.4749
## num_days_sc    1.69222     0.3174
## p_pers_sc     -0.22717     0.3151
## ppb_ave_diff_sc 0.29877     0.3110
## sexMale       -1.39408     0.5243
## p_age_sc       1.33445     2.4247
## p_age_sq_sc    -1.33428     2.4270
```

```
# confidence intervals
br_CI <- confint(modavm1_incu, full=TRUE)
br_CI
```

```
##              2.5 %    97.5 %
## (Intercept)  10.3549893 12.2164377
## num_days_sc   1.0702081  2.3142224
## p_pers_sc    -0.8446979  0.3903639
## ppb_ave_diff_sc -0.3108046  0.9083401
## sexMale      -2.4216409 -0.3665152
## p_age_sc     -3.4178145  6.0867158
## p_age_sq_sc  -6.0911801  3.4226112
```

The variables listed immediately above are highlighted as having the most significant impact on the response variable following model averaging.

## Brooding Models

The process is then repeated in full with the brooding subset.

Step 1: A linear model is created for initial analysis.

```
#bolker suggested linear model BROODING
lbr_model <- lm(no_hours_sqrt ~
               sex +
               pers_sc +
               p_pers_sc+
               age_sc +
               age_sq_sc +
               p_age_sc+
               p_age_sq_sc+
               num_days_sc+
               new_part+
               ppb_ave_diff_sc+
               age_sc:pers_sc+
               age_sq_sc:pers_sc+
               age_sc:ppb_ave_diff_sc+
               age_sq_sc:ppb_ave_diff_sc+
               pers_sc:ppb_ave_diff_sc+
               new_part:ppb_ave_diff_sc+
               p_age_sc:p_pers_sc+
               p_age_sq_sc:p_pers_sc+
               p_age_sc:ppb_ave_diff_sc+
               p_age_sq_sc:ppb_ave_diff_sc+
               p_pers_sc:sex+
               pers_sc:sex +
               p_pers_sc:ppb_ave_diff_sc,
               data = brood_new2)
summary(lbr_model)
```

```
##
## Call:
## lm(formula = no_hours_sqrt ~ sex + pers_sc + p_pers_sc + age_sc +
##     age_sq_sc + p_age_sc + p_age_sq_sc + num_days_sc + new_part +
##     ppb_ave_diff_sc + age_sc:pers_sc + age_sq_sc:pers_sc + age_sc:ppb_ave_
##     _diff_sc +
##     age_sq_sc:ppb_ave_diff_sc + pers_sc:ppb_ave_diff_sc + new_part:ppb_ave_
##     _diff_sc +
##     p_age_sc:p_pers_sc + p_age_sq_sc:p_pers_sc + p_age_sc:ppb_ave_diff_sc
##     +
##     p_age_sq_sc:ppb_ave_diff_sc + p_pers_sc:sex + pers_sc:sex +
##     p_pers_sc:ppb_ave_diff_sc, data = brood_new2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -7.0156 -1.1302  0.0577  1.2156  9.5144
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      7.9970624   0.1318999   60.630 < 2e-16 ***
## sexMale          -0.3789842   0.1818514   -2.084  0.03759 *
## pers_sc          -0.4843854   0.1232296   -3.931 9.48e-05 ***
## p_pers_sc        -0.2822854   0.1310929   -2.153  0.03170 *
## age_sc           0.3153475   0.5782460    0.545  0.58572
## age_sq_sc        -0.3599843   0.5625395   -0.640  0.52247
## p_age_sc         0.1774525   0.5870956    0.302  0.76256
## p_age_sq_sc      -0.0001924   0.5701235    0.000  0.99973
## num_days_sc      -0.2237092   0.0827674   -2.703  0.00707 **
## new_part1        0.7343620   0.2739425    2.681  0.00755 **
## ppb_ave_diff_sc   0.1246048   0.0900825    1.383  0.16712
## pers_sc:age_sc    -0.2597383   0.5656615   -0.459  0.64628
## pers_sc:age_sq_sc  0.0545676   0.5600836    0.097  0.92242
## age_sc:ppb_ave_diff_sc 0.5258047   0.6215181    0.846  0.39790
## age_sq_sc:ppb_ave_diff_sc -0.3382662   0.6045221   -0.560  0.57599
## pers_sc:ppb_ave_diff_sc -0.1658922   0.0958548   -1.731  0.08404 .
## new_part1:ppb_ave_diff_sc 0.3223241   0.2703472    1.192  0.23364
## p_pers_sc:p_age_sc  1.1604861   0.5670560    2.047  0.04115 *
## p_pers_sc:p_age_sq_sc -1.1938733   0.5589427   -2.136  0.03310 *
## p_age_sc:ppb_ave_diff_sc 0.0638631   0.6196891    0.103  0.91795
## p_age_sq_sc:ppb_ave_diff_sc -0.2563705   0.6047665   -0.424  0.67178
## sexMale:p_pers_sc -0.0138514   0.1751353   -0.079  0.93699
## sexMale:pers_sc    0.3755772   0.1753355    2.142  0.03260 *
## p_pers_sc:ppb_ave_diff_sc 0.0877555   0.0926974    0.947  0.34419
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.989 on 587 degrees of freedom
## Multiple R-squared:  0.09835,    Adjusted R-squared:  0.06302
## F-statistic: 2.784 on 23 and 587 DF,  p-value: 2.057e-05
```

Step 2: The global model is created. Sex:pers interaction is included in the brooding model.

```

co_ml_brood<- lmer(no_hours_sqrt ~
                    sex +
                    pers_sc +
                    p_pers_sc+
                    age_sc +
                    age_sq_sc +
                    p_age_sc+
                    p_age_sq_sc+
                    num_days_sc+
                    new_part+
                    ppb_ave_diff_sc+
                    age_sc:pers_sc+
                    age_sq_sc:pers_sc+
                    age_sc:ppb_ave_diff_sc+
                    age_sq_sc:ppb_ave_diff_sc+
                    pers_sc:ppb_ave_diff_sc+
                    new_part:ppb_ave_diff_sc+
                    p_age_sc:p_pers_sc+
                    p_age_sq_sc:p_pers_sc+
                    p_age_sc:ppb_ave_diff_sc+
                    p_age_sq_sc:ppb_ave_diff_sc+
                    p_pers_sc:ppb_ave_diff_sc+
                    p_pers_sc:sex +
                    pers_sc:sex +
                    (1|cycle) + (1+ppb_ave_diff_sc|cycle_pair),
                    data = brood_new2, na.action=na.fail, REML=FALSE,
                    control=lmerControl (optimizer="optimx",optCtrl=list(metho
d='nlsminb'))))

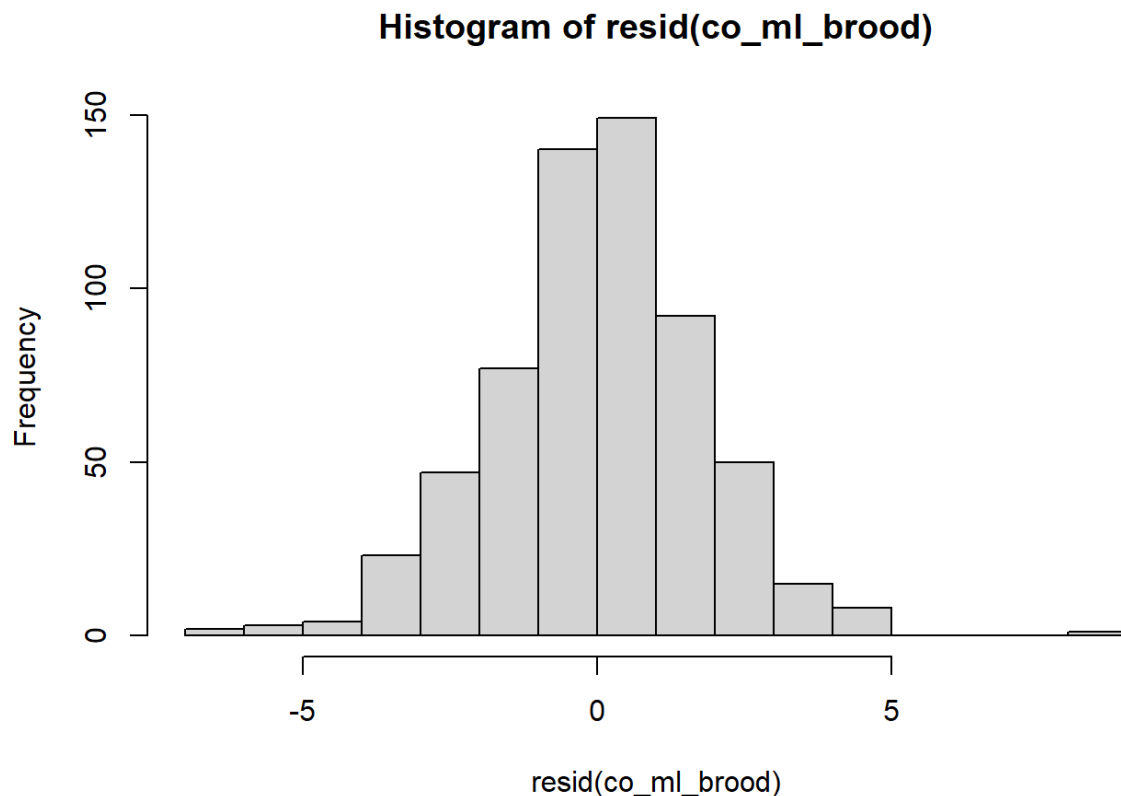
summary(co_ml_brood)

```

```
## Linear mixed model fit by maximum likelihood ['lmerMod']
## Formula: no_hours_sqrt ~ sex + pers_sc + p_pers_sc + age_sc + age_sq_sc +
##   p_age_sc + p_age_sq_sc + num_days_sc + new_part + ppb_ave_diff_sc +
##   age_sc:pers_sc + age_sq_sc:pers_sc + age_sc:ppb_ave_diff_sc +
##   age_sq_sc:ppb_ave_diff_sc + pers_sc:ppb_ave_diff_sc + new_part:ppb_ave
##   _diff_sc +
##   p_age_sc:p_pers_sc + p_age_sq_sc:p_pers_sc + p_age_sc:ppb_ave_diff_sc
##   +
##   p_age_sq_sc:ppb_ave_diff_sc + p_pers_sc:ppb_ave_diff_sc +
##   p_pers_sc:sex + pers_sc:sex + (1 | cycle) + (1 + ppb_ave_diff_sc |
##   cycle_pair)
## Data: brood_new2
## Control: lmerControl(optimizer = "optimx", optCtrl = list(method = "nllmin
##   b"))
##
##      AIC      BIC   logLik deviance df.resid
## 2585.7  2713.7 -1263.8  2527.7      582
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.3787 -0.5637  0.0321  0.5900  4.6325
##
## Random effects:
## Groups      Name                Variance Std.Dev. Corr
## cycle_pair  (Intercept)          3.000e-01 0.547734
##              ppb_ave_diff_sc    4.835e-06 0.002199 1.00
## cycle       (Intercept)          1.260e-01 0.354899
## Residual                    3.396e+00 1.842829
## Number of obs: 611, groups:  cycle_pair, 75; cycle, 6
##
## Fixed effects:
##
##              Estimate Std. Error t value
## (Intercept)      8.05400    0.21000  38.353
## sexMale          -0.37519    0.16879  -2.223
## pers_sc          -0.53081    0.13797  -3.847
## p_pers_sc        -0.43296    0.16056  -2.697
## age_sc           0.63198    0.60196   1.050
## age_sq_sc        -0.70397    0.58882  -1.196
## p_age_sc         0.46210    0.60870   0.759
## p_age_sq_sc      -0.32278    0.59488  -0.543
## num_days_sc      -0.22687    0.07844  -2.892
## new_part1        0.76443    0.33360   2.291
## ppb_ave_diff_sc   0.12430    0.08348   1.489
## pers_sc:age_sc    -0.05526    0.60860  -0.091
## pers_sc:age_sq_sc -0.10907    0.60853  -0.179
## age_sc:ppb_ave_diff_sc 0.52398    0.57577   0.910
## age_sq_sc:ppb_ave_diff_sc -0.33653    0.56003  -0.601
## pers_sc:ppb_ave_diff_sc -0.16607    0.08880  -1.870
## new_part1:ppb_ave_diff_sc 0.32177    0.25045   1.285
## p_pers_sc:p_age_sc 1.42240    0.61105   2.328
## p_pers_sc:p_age_sq_sc -1.41451    0.60874  -2.324
## p_age_sc:ppb_ave_diff_sc 0.06613    0.57405   0.115
```

```
## p_age_sq_sc:ppb_ave_diff_sc -0.25861    0.56023   -0.462
## p_pers_sc:ppb_ave_diff_sc    0.08782    0.08587    1.023
## sexMale:p_pers_sc            0.08632    0.20174    0.428
## sexMale:pers_sc              0.29842    0.20179    1.479
## optimizer (optimx) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

```
hist(resid(co_ml_brood))
```



Step 3: The model set is created and the age and partner age variables are linked to their quadratic counterparts. As the brooding data set is larger than the incubation data set, this action may take much longer than in the previous section.

```
#dredge creating model set
# linking age variables to their equivalent quadratic version so they can only
# be discarded or retained together
dr_cmbr<-dredge(co_ml_brood, subset = (dc(age_sc, age_sq_sc) &&
                                         dc(p_age_sc, p_age_sq_sc) &&
                                         dc(age_sq_sc, age_sc) &&
                                         dc(p_age_sq_sc, p_age_sc)))

dr_cmbr # view full model list
```

Step 4: Model selection of those delta < 2

```
sub_m1_brood<-subset(dr_cmbr, delta<2) ##subsets models delta <2
no_m1_brood<-dim(as.data.frame(sub_m1_brood))[1] # how many models are in the
subsetting model
no_m1_brood
```

```
## [1] 14
```

Step 5: Nested models are removed.

```
nest_m1_brood <- subset(sub_m1_brood, !nested(.))# remove nested models
nest_m1_brood
```

```
## Global model call: lmer(formula = no_hours_sqrt ~ sex + pers_sc + p_pers_sc
+ age_sc +
##   age_sq_sc + p_age_sc + p_age_sq_sc + num_days_sc + new_part +
##   ppb_ave_diff_sc + age_sc:pers_sc + age_sq_sc:pers_sc + age_sc:ppb_ave_
diff_sc +
##   age_sq_sc:ppb_ave_diff_sc + pers_sc:ppb_ave_diff_sc + new_part:ppb_ave
_diff_sc +
##   p_age_sc:p_pers_sc + p_age_sq_sc:p_pers_sc + p_age_sc:ppb_ave_diff_sc
+
##   p_age_sq_sc:ppb_ave_diff_sc + p_pers_sc:ppb_ave_diff_sc +
##   p_pers_sc:sex + pers_sc:sex + (1 | cycle) + (1 + ppb_ave_diff_sc |
##   cycle_pair), data = brood_new2, REML = FALSE, control = lmerControl(op
timizer = "optimx",
##   optCtrl = list(method = "nloptnll")), na.action = na.fail)
## ---
## Model selection table
##      (Int) new_prt num_dys_sc p_pers_sc  pers_sc ppb_ave_diff_sc sex pers_s
c:sex
## 4195277 8.093      +   -0.2249  -0.2476 -0.4971      0.1386  +
+
## 4195273 8.165      -0.2154  -0.2438 -0.4930      0.1409  +
+
## 973      8.041      +   -0.2262  -0.2617 -0.3612      0.1371  +
## 4195021 8.090      +   -0.2504  -0.2467 -0.4953      0.1371  +
+
## 969      8.113      -0.2168  -0.2569 -0.3549      0.1394  +
##      df    logLik  AICc delta weight
## 4195277 13 -1273.506 2573.6  0.00  0.312
## 4195273 12 -1275.014 2574.6  0.93  0.196
## 973      12 -1275.030 2574.6  0.96  0.193
## 4195021 12 -1275.110 2574.7  1.12  0.178
## 969      11 -1276.547 2575.5  1.91  0.120
## Models ranked by AICc(x)
## Random terms (all models):
## '1 | cycle', '1 + ppb_ave_diff_sc | cycle_pair'
```



```
no_m12_brood<-dim(as.data.frame(nest_m1_brood))[1] # how many models when whe
n nested models removed
no_m12_brood
```

```
## [1] 5
```

```
# see all coefficients for the top model set
br_coeff <- coef(nest_m1_brood, full=TRUE)
br_coeff
```

```
##      (Intercept) new_part1 num_days_sc  p_pers_sc    pers_sc ppb_ave_di
ff_sc
## 4195277      8.093260 0.5679546  -0.2248601 -0.2476188 -0.4971034      0.13
86318
## 4195273      8.165250          NA  -0.2153570 -0.2437881 -0.4929566      0.14
09049
## 973          8.041277 0.5703402  -0.2262287 -0.2616834 -0.3611505      0.13
70506
## 4195021      8.089625 0.5795633  -0.2503917 -0.2467217 -0.4953460
NA
## 969          8.113420          NA  -0.2168025 -0.2569485 -0.3549041      0.13
94184
##          sexMale pers_sc:sexMale
## 4195277 -0.3396971      0.3275564
## 4195273 -0.3376497      0.3302206
## 973      -0.3490129          NA
## 4195021 -0.3395663      0.3237265
## 969      -0.3468372          NA
```

```
nest_m1_df<- as.data.frame(br_coeff)
write.csv(nest_m1_df, file = "brooding_models.csv")
```

Step 6: Final coefficient table created via model averaging.

```
modavm1_brood<-model.avg(nest_m1_brood, fit = TRUE) # model averaging
modavm1_brood
```

```
##
## Call:
## model.avg(object = get.models(object = nest_m1_brood, subset = NA))
##
## Component models:
## '1234567' '234567' '123456' '123467' '23456'
##
## Coefficients:
##      (Intercept) new_part1 num_days_sc  p_pers_sc    pers_sc ppb_ave_diff_sc
## full      8.099115 0.3908448 -0.2268441 -0.2505449 -0.4526344      0.114
## subset    8.099115 0.5716560 -0.2268441 -0.2505449 -0.4526344      0.138
##      sexMale pers_sc:sexMale
## full    -0.3419293      0.2247778
## subset  -0.3419293      0.3273235
```

```
Weights(modavm1_brood)
```

```
## AICc model weights
## [1] 0.312 0.196 0.193 0.178 0.120
```

```
tablem1a_brood<-coefTable(modavm1_brood, full=TRUE)
tablem1a_brood # final coefficients table
```

```
##      Estimate Std. Error
## (Intercept)      8.09911      0.1896
## new_part1        0.39084      0.3763
## num_days_sc     -0.22684      0.0800
## p_pers_sc       -0.25054      0.1020
## pers_sc        -0.45263      0.1369
## ppb_ave_diff_sc  0.11415      0.0878
## sexMale        -0.34193      0.1693
## pers_sc:sexMale  0.22478      0.2168
```

```
# confidence intervals
br_CI <- confint(modavm1_brood, full=TRUE)
br_CI
```

##	2.5 %	97.5 %
## (Intercept)	7.72754168	8.47068751
## new_part1	-0.34669553	1.12838520
## num_days_sc	-0.38355860	-0.07012965
## p_pers_sc	-0.45041008	-0.05067980
## pers_sc	-0.72093046	-0.18433833
## ppb_ave_diff_sc	-0.05796129	0.28626393
## sexMale	-0.67375057	-0.01010806
## pers_sc:sexMale	-0.20018958	0.64974517

The variables listed above are highlighted as the most influential on the response variable following model averaging.