Waal Intrinsic Variables and Parental Care Coordination Model Script

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 markdown experiment")
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("anon_waal_dataset.csv")
total_df2$X<-NULL

#chategorical 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)
total_df2$cycle<-as.factor(total_df2$cycle)
total_df2$cycle_pair<-as.factor(total_df2$cycle_pair)
```

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:

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

```
## A B C D E F G
## 1 8 6 6 2 7 5
```

```r
# number birds per age group (individual birds may appear more than once)
with(incu_new, 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
```

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

```
##
##  A  B  C  D  E  F  G
## 10 70 49 38 14 38 41
```

Measures of centrality and group comparison for the incubation subset:

```r
# 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
```

```r
# 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.
```

```
# loggered and partner bird (unloggered) 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
## * <fct>   <dbl> <dbl> <dbl>
## 1 A       192.  185.  49.9
## 2 B       143.  137.  111.
## 3 C       151.  159.  112.
## 4 D       102.   89.7  82.2
## 5 E        94.5 103.   72.4
## 6 F       119.  115.   91.0
## 7 G       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:

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

```
##  A  B  C  D  E  F  G
## 15 20  9 12 11  8 NA
```

```
# number birds per age group
with(brood_new, 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
```

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

```
##
##   A   B   C   D   E   F   G
## 137 174  70  81  84  65   0
```

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
```

```
# loggered and partner birds comparison
brood_new%>%group_by(partner_type)%>%summarise(average=mean(no_hours),
                                      med= median(no_hours), sd=sd(n
        o_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_hour
        s))
```

```
## # A tibble: 6 x 4
##   cycle average   med    sd
## * <fct>   <dbl> <dbl> <dbl>
## 1 A        59.7  58.5  27.9
## 2 B        61.3  57.4  30.0
## 3 C        66.5  61.1  35.8
## 4 D        77.4  70.2  42.4
## 5 E        69.2  71.4  28.3
## 6 F        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 below include making a new data frame, calculating a quadratic age variable and adjusting the partner's previous trip variable to account for each individual's mean previous 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 1-4 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: In this important section, the partner's previous bout variable is refined to account for each individual's mean previous 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 previous trip length. Step-by-step instructions for each line of code are included in the comments.

```
# creating final partner's previous bout variable
# calculate mean previous trip duration for each bird's partner in each seaso
     n
incu_aves<-incu_new2%>%group_by(cycle_p_tag)%>%
   summarise(ppb_ave_no_hours=mean(part_prev_bout))

# convert this into a data frame
incu_avesdf<- data.frame(incu_aves)

# add partner mean to original data set
incu_new2<-merge(incu_new2, incu_avesdf, by.x = "cycle_p_tag", by.y="cycle_p_
     tag")

# subtract raw partners previous bout from partner's mean
incu_new2<-incu_new2%>%mutate(ppb_ave_diff=part_prev_bout-ppb_ave_no_hours)

# new variable summary
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 1-3 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 new variable representing partner's previous trip duration is created.

```
# creating final partner's previous bout variable
# calculate mean previous trip duration for each bird's partner in each seaso
       n
brood_aves<-brood_new2%>%group_by(cycle_p_tag)%>%
  summarise(ppb_ave_no_hours=mean(part_prev_bout))

# converted into a dataframe
brood_avesdf<- data.frame(brood_aves)

# add new mean to original data set
brood_new2<-merge(brood_new2, brood_avesdf, by.x = "cycle_p_tag", by.y="cycle
       _p_tag")

# subtract previous partner bout from mean
brood_new2<-brood_new2%>%mutate(ppb_ave_diff=part_prev_bout-ppb_ave_no_hours)

# new variable summary
brood_new2%>%summarise(average=mean(ppb_ave_diff),med=median(ppb_ave_diff), s
       d=sd(ppb_ave_diff))
```

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

The two breeding stage subsets and recombined so that the variables can be scaled across the entire dataset. This will allow coordination strength to be directly compared between models and thus, between breeding stages. The data are then redivided into the subsets for analysis.

```
#recombine into one dataframe for scaling (so coordination is comparable)
# a combined data set of both incubation and brooding data is created
total_df4<-full_join(incu_new2, brood_new2)

#rescaling

total_df4$age_sc<-scale(total_df4$age)
total_df4$p_age_sc<-scale(total_df4$p_age)
total_df4$age_sq_sc<-scale(total_df4$age_sq)
total_df4$p_age_sq_sc<-scale(total_df4$p_age_sq)
total_df4$pers_sc<-scale(total_df4$pers)
total_df4$num_days_sc<-scale(total_df4$NumDays)
total_df4$p_pers_sc<-scale(total_df4$p_pers)
total_df4$ppb_ave_diff_sc<-scale(total_df4$ppb_ave_diff)

#redivide into incubation and brooding
incu_new2<-subset(total_df4, breed_stage == "incu") # 260 observations
brood_new2<-subset(total_df4, breed_stage== "brood") # 611 observations
```

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 Multicolinearity

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
```
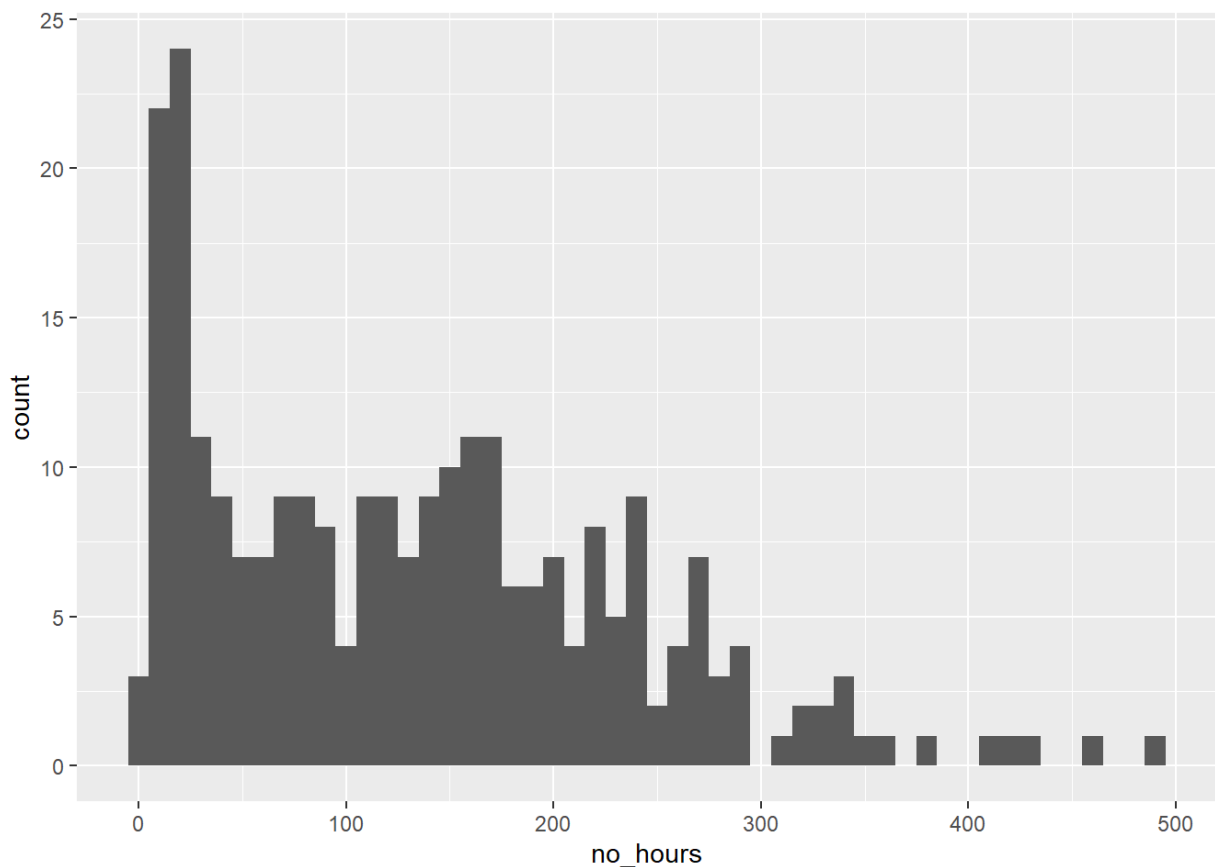


```
# creating new sqrt response variable
incu_new2$no_hours_sqrt<-sqrt(incu_new2$no_hours)

#check success
histogram_hours_sqrt<-incu_new2 %>%ggplot(aes(x=no_hours_sqrt))+
  geom_histogram (binwidth = 0.5)
histogram_hours_sqrt
```
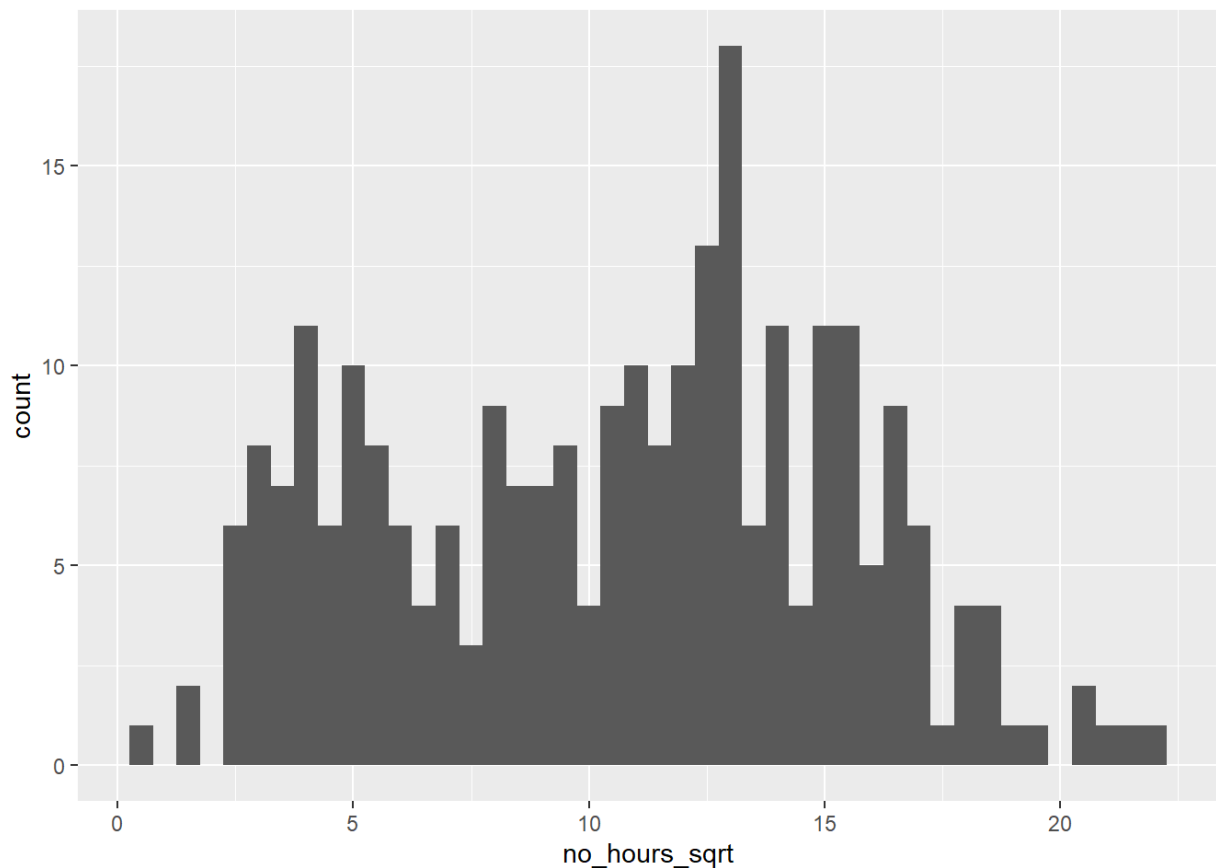
The variable `no_hours_sqrt` will be used as the response variable in the models.

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 multicolinearity between potentially correlated continuous fixed variables. VIF scores > 3 are considered problematic. Code for the production of graphs has been provided but 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)  10.6981     0.3088  34.641   <2e-16 ***
## age_sc       -0.4176     0.2885  -1.447    0.149
## pers_sc       0.4037     0.3073   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 = 3.5518e-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
## 2.211278e-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)      10.6973     0.3026  35.355  < 2e-16 ***
## age_sc           -0.4056     0.2825  -1.436 0.152350
## ppb_ave_diff_sc   0.6390     0.1805   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)  10.6721     0.3125  34.153   <2e-16 ***
## age_sc       -0.5638     0.3912  -1.441    0.151
## p_age_sc      0.2360     0.3932   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 = pers_sc , y = p_pers_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) 10.57647    0.29408  35.965   <2e-16 ***
## pers_sc      0.03166    0.35741   0.089   0.9295
## p_pers_sc   -0.69353    0.35724  -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
```
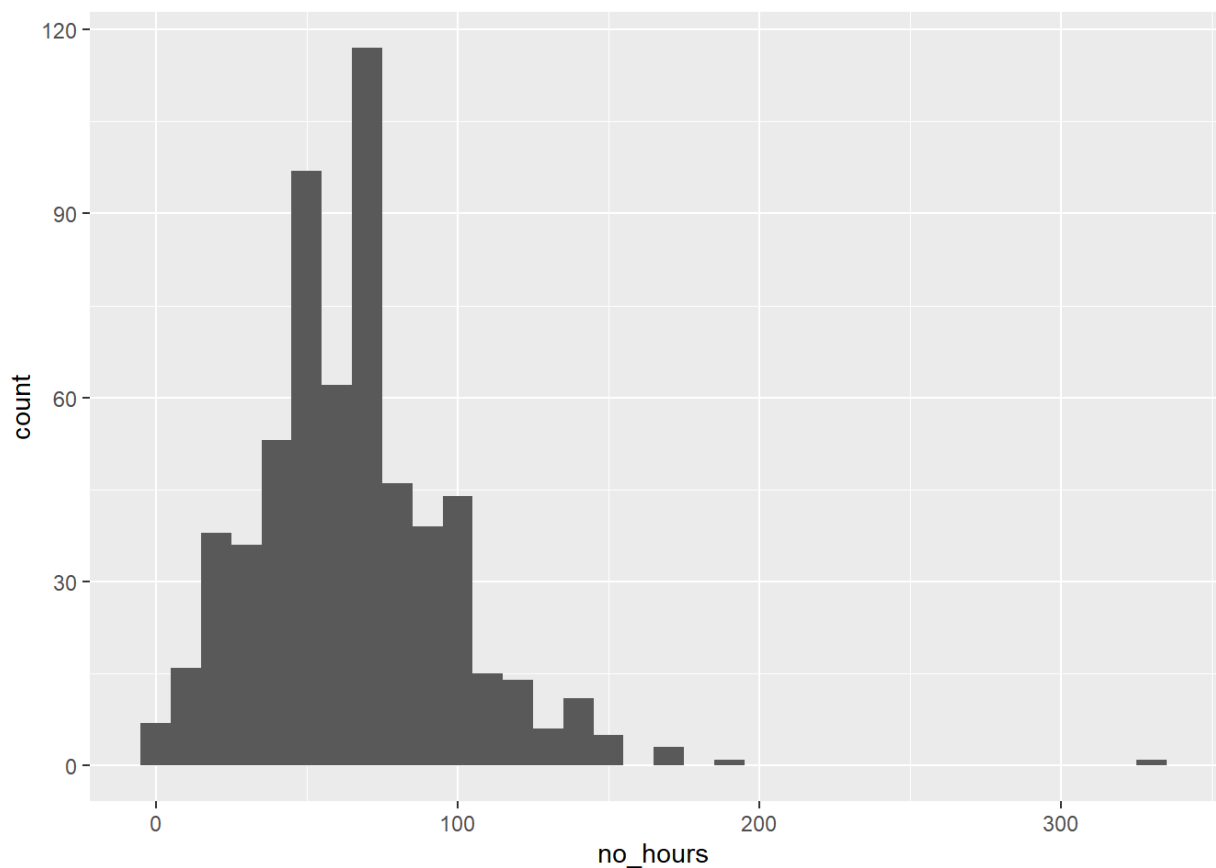


```
# creating new sqrt response variable
brood_new2$no_hours_sqrt<-sqrt(brood_new2$no_hours)

# check success
histogram_hours_sqrt<-brood_new2 %>%ggplot(aes(x=no_hours_sqrt))+
geom_histogram (binwidth = 0.5)
histogram_hours_sqrt
```
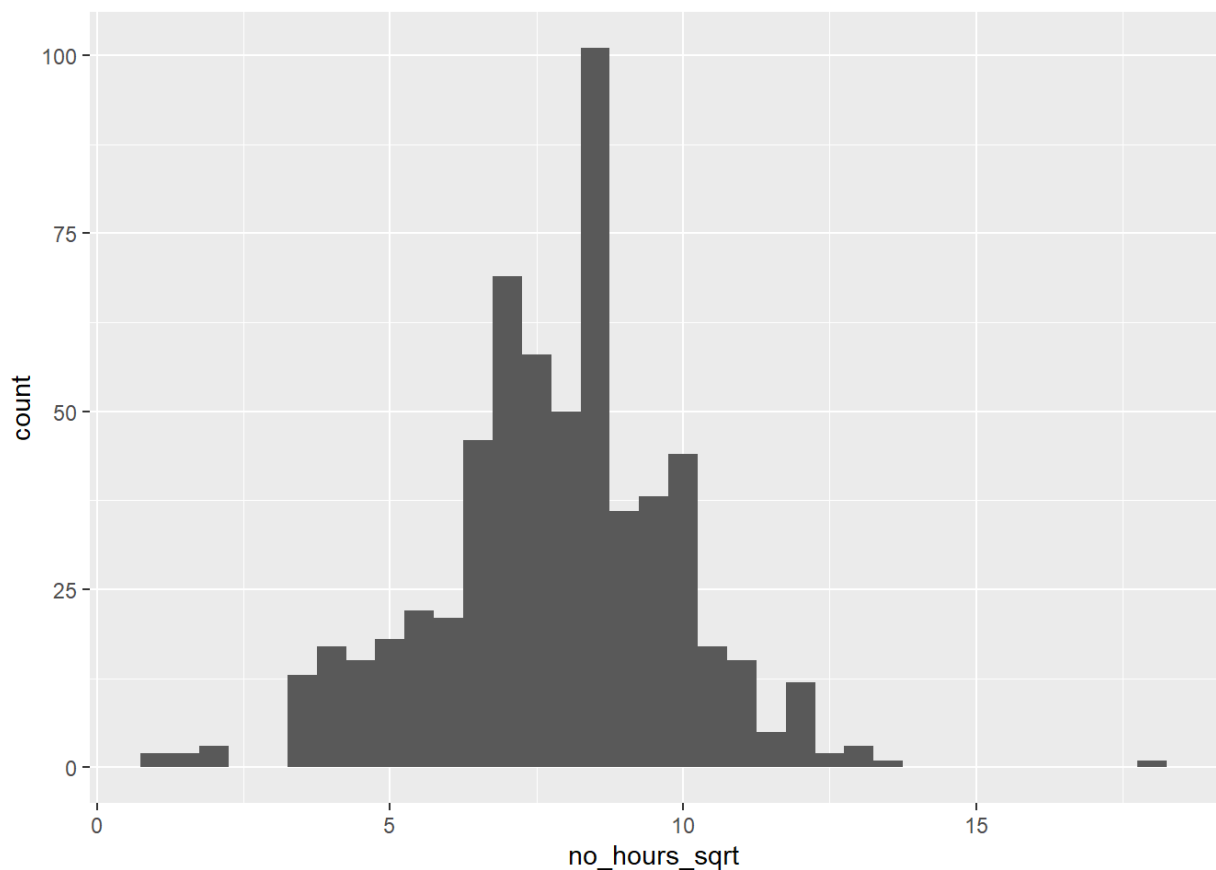
The variable `no_hours_sqrt` will be used as the response variable in the models.

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 multicolinearity between potentially correlated continuous fixed variables. VIF scores > 3 are considered problematic. Code for the production of graphs has been provided but 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)  7.81635    0.08354  93.565  < 2e-16 ***
## age_sc       0.00511    0.08733   0.059  0.95336
## pers_sc     -0.23716    0.08238  -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
```

```
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 = -2.1681e-16, 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
## -8.785404e-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)      7.81199    0.08379  93.233   <2e-16 ***
## age_sc          -0.03277    0.08661  -0.378   0.7053
## ppb_ave_diff_sc  0.30211    0.14388   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
```

```
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)  7.81998    0.08419  92.890   <2e-16 ***
## age_sc      -0.11746    0.10762  -1.091    0.276
## p_age_sc     0.14297    0.10740   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
```

```
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 =pers_sc , y =p_pers_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)  7.81435    0.08204  95.250  < 2e-16 ***
## pers_sc     -0.24924    0.08088  -3.082  0.00215 **
## p_pers_sc   -0.25543    0.08089  -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
```

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

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

Create a total dataset and save files

An updated, combined data set of both the incubation and brooding data is created.

```
# an updated combined data set of both incubation and brooding data is create
    d
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$age) #no ages
# pair sample size per year
with(total_df4, tapply(pair_id, cycle, FUN = function(x) length(unique(x))))
with(total_df4, tapply(tag, age, FUN = function(x) length(unique(x))))
total_df4$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(n
        o_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_hour
        s))
```

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)                15.32418    0.75270  20.359  < 2e-16 ***
## sexMale                    -1.38103    0.61076  -2.261   0.0247 *
## pers_sc                    -0.63694    0.38494  -1.655   0.0993 .
## p_pers_sc                  -0.80265    0.38713  -2.073   0.0392 *
## age_sc                     -0.65813    2.13447  -0.308   0.7581
## age_sq_sc                   0.24411    1.94267   0.126   0.9001
## p_age_sc                    4.26069    2.12966   2.001   0.0466 *
## p_age_sq_sc                -3.83024    1.93356  -1.981   0.0488 *
## num_days_sc                 2.83202    0.43610   6.494 4.83e-10 ***
## new_part1                   0.87002    1.84494   0.472   0.6377
## ppb_ave_diff_sc             0.24523    0.19887   1.233   0.2187
## pers_sc:age_sc              1.58068    2.73867   0.577   0.5644
## pers_sc:age_sq_sc          -1.40377    2.44800  -0.573   0.5669
## age_sc:ppb_ave_diff_sc      0.05428    1.37083   0.040   0.9684
## age_sq_sc:ppb_ave_diff_sc  -0.08801    1.25451  -0.070   0.9441
## pers_sc:ppb_ave_diff_sc    -0.09331    0.24201  -0.386   0.7002
## new_part1:ppb_ave_diff_sc   0.78805    0.84127   0.937   0.3498
## p_pers_sc:p_age_sc          3.30004    2.76944   1.192   0.2346
## p_pers_sc:p_age_sq_sc      -3.24049    2.46855  -1.313   0.1905
## p_age_sc:ppb_ave_diff_sc   -0.67920    1.16801  -0.582   0.5615
## p_age_sq_sc:ppb_ave_diff_sc 0.66851    1.09266   0.612   0.5412
## p_pers_sc:ppb_ave_diff_sc   0.10792    0.24089   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+
                  p_pers_sc:sex+
                  pers_sc:sex +
                  (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 +
##       p_pers_sc:sex + pers_sc:sex + (1 | cycle) + (1 + ppb_ave_diff_sc |
##       cycle_pair)
##     Data: incu_new2
## Control: lmerControl(optimizer = "optimx", optCtrl = list(method = "nlmin
b"))
##
##      AIC      BIC   logLik deviance df.resid
##   1505.2   1608.4   -723.6   1447.2      231
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.76315 -0.63795  0.00059  0.62819  2.38054
##
## Random effects:
##  Groups     Name             Variance  Std.Dev. Corr
##  cycle_pair (Intercept)      2.464e+00 1.56987
##             ppb_ave_diff_sc  3.758e-04 0.01938  1.00
##  cycle      (Intercept)      0.000e+00 0.00000
##  Residual                    1.371e+01 3.70235
## Number of obs: 260, groups:  cycle_pair, 35; cycle, 7
##
## Fixed effects:
##                            Estimate Std. Error t value
## (Intercept)                15.04740    0.85441  17.612
## sexMale                    -1.37140    0.53587  -2.559
## pers_sc                    -0.98388    0.54175  -1.816
## p_pers_sc                  -0.59000    0.65383  -0.902
## age_sc                      0.53765    2.42071   0.222
## age_sq_sc                  -0.77316    2.18984  -0.353
## p_age_sc                    4.77325    2.42416   1.969
## p_age_sq_sc                -4.21704    2.19027  -1.925
## num_days_sc                 2.49371    0.45661   5.461
## new_part1                  -0.12620    2.39419  -0.053
## ppb_ave_diff_sc             0.28698    0.17794   1.613
## pers_sc:age_sc              0.72996    3.24522   0.225
## pers_sc:age_sq_sc          -0.71652    2.93251  -0.244
## age_sc:ppb_ave_diff_sc      0.05619    1.20103   0.047
## age_sq_sc:ppb_ave_diff_sc  -0.09725    1.09919  -0.088
## pers_sc:ppb_ave_diff_sc    -0.09157    0.21208  -0.432
## new_part1:ppb_ave_diff_sc   0.77405    0.73742   1.050
## p_pers_sc:p_age_sc          2.67887    3.27022   0.819
## p_pers_sc:p_age_sq_sc      -2.73887    2.95049  -0.928
## p_age_sc:ppb_ave_diff_sc   -0.69698    1.02343  -0.681
```

```
## p_age_sq_sc:ppb_ave_diff_sc    0.68918      0.95741    0.720
## p_pers_sc:ppb_ave_diff_sc       0.10428      0.21108    0.494
## sexMale:p_pers_sc              -0.42250      0.75921   -0.557
## sexMale:pers_sc                 0.82259      0.75777    1.086
## optimizer (optimx) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

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


Histogram of resid(co_ml_incu)
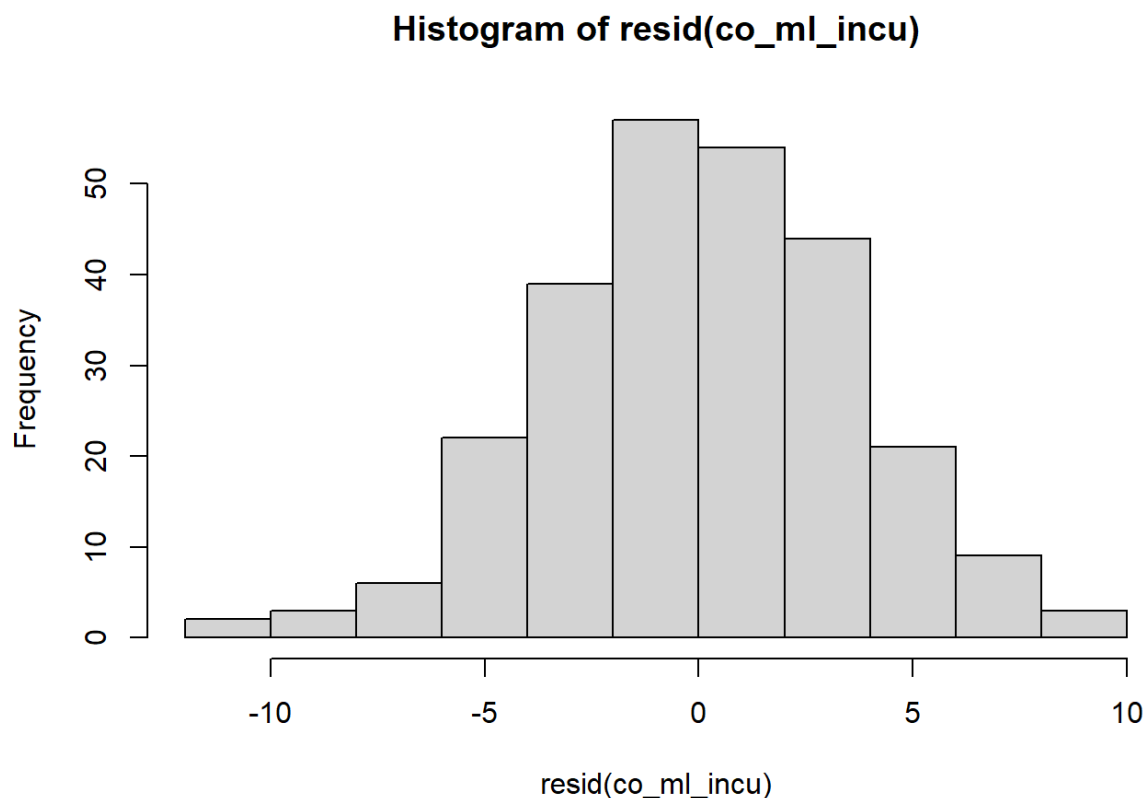
Step 3: A model set is created from the global model. This creates all possible versions of the model to establish the best fit. As the focal bird age and its quadratic equivalent (age_sc and age_sq_sc) must be retained or discarded together, additional code is added to ensure this is reflected in the 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
        ither 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 subsetted models
```

```
## 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 = incu_new2, REML = FALSE, control = lmerControl(optimizer = "optimx",
##     optCtrl = list(method = "nlminb")), na.action = na.fail)
## ---
## Model selection table
```

	(Int)	num_dys_sc	p_age_sc	p_age_sq_sc	p_prs_sc	prs_sc	ppb_ave_dff_sc
## 841	14.72	2.479			-0.5237		0.2997
## 132089	14.65	2.472	5.610	-5.005	-0.7032	-0.6852	0.2974
## 1017	14.65	2.481	4.711	-4.229	-0.8962	-0.6365	0.2954
## 33785	14.63	2.478	5.615	-4.998	-0.6900	-0.6796	0.2964
## 889	14.44	2.388	4.179	-3.740	-0.4489		0.3059
## 131961	14.41	2.373	4.987	-4.434	-0.2359		0.3086
## 969	14.91	2.561			-0.8887	-0.4978	0.2903
## 825	14.52	2.336	4.649	-4.202			0.3116
## 585	15.06	2.768			-0.5294		
## 777	14.84	2.434					0.3049
## 33657	14.40	2.379	4.994	-4.430	-0.2272		0.3076
## 131833	15.00	2.766	5.477	-4.883	-0.7198	-0.6961	
## 4195273	15.16	2.632			-0.8553	-0.8846	0.2831
## 761	15.00	2.769	4.608	-4.133	-0.9086	-0.6475	
## 713	15.27	2.851			-0.9018	-0.5130	
## 33529	14.99	2.771	5.486	-4.880	-0.7062	-0.6906	
## 2229241	14.66	2.483	5.609	-5.003	-0.7089	-0.6911	0.2841
## 633	14.77	2.677	4.065	-3.633	-0.4580		
## 525129	14.72	2.481				-0.5235	0.3034
## 164857	14.71	2.447	5.225	-4.731	-0.8465	-0.6941	0.3018

```
## 4326393 14.84    2.535    5.168    -4.595  -0.6764 -0.9412       0.2
910
## 131705  14.76    2.669    4.840    -4.299  -0.2502
## 569     14.85    2.629    4.530    -4.090
## 521     15.17    2.725
## 4195017 15.52    2.920                        -0.8695 -0.9132
## 2098169 14.67    2.492    4.707    -4.225  -0.9021 -0.6422       0.2
828
##          sex p_age_sc:p_prs_sc p_age_sq_sc:p_prs_sc p_prs_sc:ppb_ave_dff_sc
## 841       +
## 132089    +                                  -0.3842
## 1017      +
## 33785     +          -0.4214
## 889       +
## 131961    +                                  -0.3568
## 969       +
## 825       +
## 585       +
## 777       +
## 33657     +          -0.3913
## 131833    +                                  -0.3777
## 4195273   +
## 761       +
## 713       +
## 33529     +          -0.4153
## 2229241   +                                  -0.3847
## 633       +
## 525129    +                                                          0.1084
## 164857    +           2.1150                 -2.1590
## 4326393   +                                  -0.3899
## 131705    +                                  -0.3489
## 569       +
## 521       +
## 4195017   +
## 2098169   +
##          prs_sc:ppb_ave_dff_sc prs_sc:sex df    logLik    AICc delta weight
## 841                                        10 -730.731 1482.3  0.00  0.070
## 132089                                     14 -726.473 1482.7  0.31  0.060
## 1017                                       13 -727.747 1483.0  0.63  0.051
## 33785                                      14 -726.653 1483.0  0.67  0.050
## 889                                        12 -728.906 1483.1  0.73  0.049
## 131961                                     13 -727.814 1483.1  0.76  0.048
## 969                                        11 -730.046 1483.2  0.81  0.047
## 825                                        11 -730.055 1483.2  0.83  0.046
## 585                                         9 -732.228 1483.2  0.83  0.046
## 777                                         9 -732.322 1483.4  1.02  0.042
## 33657                                      13 -727.969 1483.4  1.07  0.041
## 131833                                     13 -728.054 1483.6  1.24  0.038
## 4195273                                  + 12 -729.231 1483.7  1.38  0.035
## 761                                        12 -729.274 1483.8  1.47  0.034
## 713                                        10 -731.491 1483.9  1.52  0.033
## 33529                                      13 -728.220 1483.9  1.57  0.032
## 2229241               -0.1750              15 -726.027 1484.0  1.67  0.030
```

```
## 633                                          11 -730.485 1484.0  1.69  0.030
## 525129                                       11 -730.528 1484.1  1.77  0.029
## 164857                                       15 -726.080 1484.1  1.78  0.029
## 4326393                                    + 15 -726.087 1484.1  1.80  0.028
## 131705                                       12 -729.450 1484.2  1.82  0.028
## 569                                          10 -731.670 1484.2  1.88  0.027
## 521                                           8 -733.836 1484.2  1.90  0.027
## 4195017                                    + 11 -730.626 1484.3  1.97  0.026
## 2098169                    -0.1742           14 -727.308 1484.3  1.98  0.026
## 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] 26
```

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 +
##      p_pers_sc:sex + pers_sc:sex + (1 | cycle) + (1 + ppb_ave_diff_sc |
##      cycle_pair), data = incu_new2, REML = FALSE, control = lmerControl(opt
imizer = "optimx",
##      optCtrl = list(method = "nlminb")), 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 14.72      2.479                          -0.5237        0.2997   + 10
## 825 14.52      2.336    4.649      -4.202                     0.3116   + 11
## 585 15.06      2.768                          -0.5294                 +  9
## 777 14.84      2.434                                         0.3049   +  9
## 569 14.85      2.629    4.530      -4.090                              + 10
## 521 15.17      2.725                                                  +  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    14.71946    2.478942 -0.5236819       0.2997056 -1.207431       NA
## 825    14.51523    2.336453         NA       0.3116017 -1.481906 4.648746
## 585    15.05950    2.768288 -0.5294161              NA -1.213488       NA
## 777    14.83676    2.433595         NA       0.3048772 -1.600663       NA
## 569    14.84952    2.628743         NA              NA -1.495424 4.530132
## 521    15.17452    2.724689         NA              NA -1.611292       NA
##      p_age_sq_sc
## 841          NA
## 825   -4.202089
## 585          NA
## 777          NA
## 569   -4.089666
## 521          NA
```

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       14.82403    2.539267 -0.2360045       0.1862005 -1.394078 1.3089
21
## subset     14.82403    2.539267 -0.5259621       0.3045553 -1.394078 4.6046
54
##         p_age_sq_sc
## full      -1.182609
## subset    -4.160299
```

```
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)     14.8240     0.8234
## num_days_sc      2.5393     0.4762
## p_pers_sc       -0.2360     0.3273
## ppb_ave_diff_sc  0.1862     0.1938
## sexMale         -1.3941     0.5243
## p_age_sc         1.3089     2.3783
## p_age_sq_sc     -1.1826     2.1511
```

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

```
##                      2.5 %      97.5 %
## (Intercept)     13.2101537 16.4378999
## num_days_sc      1.6059071  3.4726267
## p_pers_sc       -0.8775593  0.4055503
## ppb_ave_diff_sc -0.1937015  0.5661025
## sexMale         -2.4216408 -0.3665152
## p_age_sc        -3.3524302  5.9702729
## p_age_sq_sc     -5.3987615  3.0335444
```

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.

```r
#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)               8.714918   0.292430  29.802  < 2e-16 ***
## sexMale                  -0.377874   0.181854  -2.078  0.03815 *
## pers_sc                  -0.504386   0.122373  -4.122  4.3e-05 ***
## p_pers_sc                -0.294552   0.131380  -2.242  0.02534 *
## age_sc                    0.328068   0.603438   0.544  0.58688
## age_sq_sc                -0.397948   0.622487  -0.639  0.52289
## p_age_sc                  0.190640   0.611653   0.312  0.75539
## p_age_sq_sc              -0.006675   0.629309  -0.011  0.99154
## num_days_sc              -1.179798   0.436499  -2.703  0.00707 **
## new_part1                 0.734362   0.273943   2.681  0.00755 **
## ppb_ave_diff_sc           0.203408   0.155648   1.307  0.19178
## pers_sc:age_sc           -0.266528   0.580449  -0.459  0.64628
## pers_sc:age_sq_sc         0.059374   0.609413   0.097  0.92242
## age_sc:ppb_ave_diff_sc    0.950273   1.123253   0.846  0.39790
## age_sq_sc:ppb_ave_diff_sc -0.648237   1.158477  -0.560  0.57599
## pers_sc:ppb_ave_diff_sc  -0.282788   0.163399  -1.731  0.08404 .
## new_part1:ppb_ave_diff_sc 0.558494   0.468433   1.192  0.23364
## p_pers_sc:p_age_sc        1.188637   0.580812   2.047  0.04115 *
## p_pers_sc:p_age_sq_sc    -1.295600   0.606569  -2.136  0.03310 *
## p_age_sc:ppb_ave_diff_sc  0.115182   1.117657   0.103  0.91795
## p_age_sq_sc:ppb_ave_diff_sc -0.489899 1.155649  -0.424  0.67178
## sexMale:p_pers_sc        -0.013630   0.172335  -0.079  0.93699
## sexMale:pers_sc           0.369495   0.172496   2.142  0.03260 *
## p_pers_sc:ppb_ave_diff_sc 0.149624   0.158050   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='nlminb')))

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 = "nlmin
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 0.547733
##             ppb_ave_diff_sc 1.446e-05 0.003803 1.00
##  cycle      (Intercept)     1.260e-01 0.354911
##  Residual                   3.396e+00 1.842829
## Number of obs: 611, groups:  cycle_pair, 75; cycle, 6
##
## Fixed effects:
##                            Estimate Std. Error t value
## (Intercept)                 8.76470    0.32443  27.015
## sexMale                    -0.37383    0.16879  -2.215
## pers_sc                    -0.54614    0.13758  -3.969
## p_pers_sc                  -0.43932    0.16087  -2.731
## age_sc                      0.65901    0.62821   1.049
## age_sq_sc                  -0.77897    0.65160  -1.195
## p_age_sc                    0.48828    0.63420   0.770
## p_age_sq_sc                -0.36363    0.65668  -0.554
## num_days_sc                -1.19646    0.41370  -2.892
## new_part1                   0.76443    0.33360   2.291
## ppb_ave_diff_sc             0.20285    0.14425   1.406
## pers_sc:age_sc             -0.05670    0.62451  -0.091
## pers_sc:age_sq_sc          -0.11868    0.66212  -0.179
## age_sc:ppb_ave_diff_sc      0.94697    1.04057   0.910
## age_sq_sc:ppb_ave_diff_sc  -0.64492    1.07322  -0.601
## pers_sc:ppb_ave_diff_sc    -0.28309    0.15137  -1.870
## new_part1:ppb_ave_diff_sc   0.55754    0.43396   1.285
## p_pers_sc:p_age_sc          1.45691    0.62587   2.328
## p_pers_sc:p_age_sq_sc      -1.53504    0.66061  -2.324
## p_age_sc:ppb_ave_diff_sc    0.11927    1.03534   0.115
```

```
## p_age_sq_sc:ppb_ave_diff_sc  -0.49417      1.07054   -0.462
## p_pers_sc:ppb_ave_diff_sc     0.14973      0.14641    1.023
## sexMale:p_pers_sc             0.08494      0.19851    0.428
## sexMale:pers_sc               0.29358      0.19853    1.479
## optimizer (optimx) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

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


Histogram of resid(co_ml_brood)
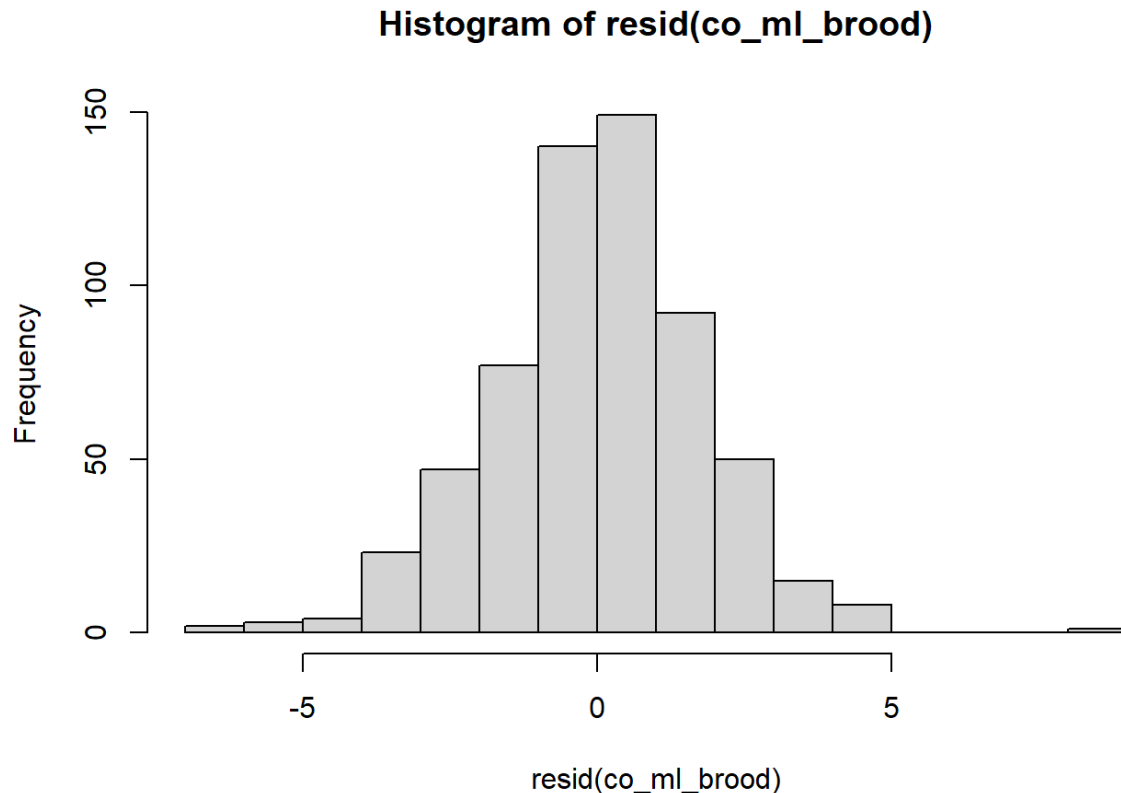
```
typeof(brood_new2$cycle)
```

```
## [1] "integer"
```

```
brood_new2$cycle_pair<-as.factor(brood_new2$cycle_pair)
print(class(brood_new2$cycle))
```

```
## [1] "factor"
```

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 onl
        y 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
        subsetted 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_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 +
##       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 = "nlminb")), na.action = na.fail)
## ---
## Model selection table
##          (Int) new_prt num_dys_sc p_prs_sc  prs_sc ppb_ave_dff_sc sex prs_s
c:sex
## 4195277 8.800       +      -1.186 -0.2437 -0.4891         0.2402   +
+
## 4195273 8.842             -1.136 -0.2399 -0.4850         0.2441   +
+
## 973     8.753       +      -1.193 -0.2575 -0.3553         0.2375   +
## 4195021 8.877       +      -1.321 -0.2428 -0.4873                  +
+
## 969     8.795             -1.143 -0.2528 -0.3492         0.2416   +
##          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     8.800156 0.5679543   -1.185867 -0.2436598 -0.4890533       0.24
02089
## 4195273     8.842187        NA   -1.135754 -0.2398903 -0.4849722       0.24
41454
## 973         8.752849 0.5703423   -1.193085 -0.2575016 -0.3553042       0.23
74691
## 4195021     8.877111 0.5795640   -1.320517 -0.2427785 -0.4873253
NA
## 969         8.795286        NA   -1.143372 -0.2528385 -0.3491549       0.24
15725
##            sexMale pers_sc:sexMale
## 4195277 -0.3386695       0.3222521
## 4195273 -0.3366140       0.3248698
## 973     -0.3490135              NA
## 4195021 -0.3385509       0.3184832
## 969     -0.3468370              NA
```

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_dif
f_sc
## full        8.812398 0.3908453   -1.196331 -0.2465396 -0.4453045       0.197
7910
## subset      8.812398 0.5716566   -1.196331 -0.2465396 -0.4453045       0.240
7039
##            sexMale pers_sc:sexMale
## full    -0.3412243       0.2211370
## subset  -0.3412243       0.3220218
```

```
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)        8.81240     0.3123
## new_part1          0.39085     0.3763
## num_days_sc       -1.19633     0.4217
## p_pers_sc         -0.24654     0.1003
## pers_sc           -0.44530     0.1347
## ppb_ave_diff_sc    0.19779     0.1522
## sexMale           -0.34122     0.1693
## pers_sc:sexMale    0.22114     0.2133
```

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

```
##                        2.5 %        97.5 %
## (Intercept)        8.2002328   9.424562759
## new_part1         -0.3466940   1.128384518
## num_days_sc       -2.0228129  -0.369850021
## p_pers_sc         -0.4432089  -0.049870259
## pers_sc           -0.7092551  -0.181353889
## ppb_ave_diff_sc   -0.1004302   0.496012163
## sexMale           -0.6730953  -0.009353258
## pers_sc:sexMale   -0.1969476   0.639221628
```

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