Neurospora crassa trait models

Kolea Zimmerman

Kolea Zimmerman

14 September, 2015

- Load packages
- Additional functions
  - Deviance calculation for qAIC
  - R2GLMM
  - Mean Center
- Data import and processing
  - Import data
  - Calculations used to infer the number of ascospores and conidia in germinated samples.
  - Create data variables for models (except spore size).
- Trait Heatmaps
  - Data processing
  - Perithecia Count
  - Total Spore Count
  - Percent Pigmented
  - Pigmented Spore Size
  - Pigmented Spore Germination
- Trait Correlogram
  - Supplementary Figure 4
- Model Perithecia Count
  - Full Model
  - Calculate qAIC
  - Selected Model
  - R2GLMM of selected model.
  - Variance of maternal vs. paternal random effect coefficients (BLUPs) of selected model
  - Correlation between maternal and paternal BLUPs of selected model
- Model Total Ascospore Count
  - Full Model
  - Calculate qAIC
  - Selected Model
  - R2GLMM of selected model.
  - Variance of maternal vs. paternal random effect coefficients (BLUPs) of selected model
  - Correlation between maternal and paternal BLUPs of selected model
• Model Proportion Pigmented
  – Full Model
  – Calculate qAIC
  – Selected Model
  – R2GLMM of selected model.
  – Variance of maternal vs. paternal random effect coefficients (BLUPs) of selected model
  – Correlation between maternal and paternal BLUPs of selected model

• Model Pigmented Spore Size
  – Full Model
  – Calculate AIC
  – Selected Model
  – R2GLMM of selected model.
  – Variance of maternal vs. paternal random effect coefficients (BLUPs) of selected model
  – Correlation between maternal and paternal BLUPs of selected model

• Model Pigmented Spore Germination
  – Full Model
  – Calculate qAIC
  – Selected Model
  – R2GLMM of selected model.
  – Variance of maternal vs. paternal random effect coefficients (BLUPs) of selected model
  – Correlation between maternal and paternal BLUPs of selected model

• Plot variance explained by models
• Plot maternal and paternal proportions of random effect variance
• Plot of maternal vs paternal BLUPs
• HPD intervals for parameters
  – MCMC parameter estimation
  * Perithecia Count
  * Spore Count
  * Proportion Pigmented
  * Pigmented Size
  * Pigmented Spore Germination
  – Plot parameters and HPD intervals

This document shows the GLMM analysis used in the paper.

Load packages

ipak function: install and load multiple R packages (Steve Worthington https://gist.github.com/stevenworthington/3178163)

```
#install.packages("glmmADMB", repos=c("http://glmmadmb.r-forge.r-project.org/repos",
#getOption("repos")), type="source")

ipak <- function(pkg){
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
  if (length(new.pkg))
    install.packages(new.pkg, repos="http://cran.us.r-project.org", dependencies = TRUE)
  sapply(pkg, require, character.only = TRUE)
```
packages <- c("glmmADMB", "lme4", "plyr", 
    "ggplot2", "reshape", "car", 
    "MuMIn", "R2admb", "corrgram", "grid")

ipak(packages)

## glmmADMB lme4 plyr ggplot2 reshape car MuMIn R2admb
## TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## corrgram grid
## TRUE TRUE

Additional functions

Deviance calculation for qAIC

```r
#function to calculate deviance for quasi AIC, used in 'chat' arg of qAIC
#calculations see https://cran.r-project.org/web/packages/bbmle/vignettes/quasi.pdf
dfun <- function(object) {
    MM<--Anova(object)
    df_residuals<-max(MM$Df)
    res<--residuals(object)
    res[is.na(res)==T]<-0
    return(sum(1 * res^2)/df_residuals)
}
```

R2GLMM

```r
#function to calculate R2glmm
glmmadmbR2 <- function (model, offset=TRUE, spec_os, err_dist) {
    # Determines the marginal and conditional R2glmm described in:
    # for a glmmADMB model using a negative binomial or gaussian error distribution,
    # with an option for specifying an offset if it is used in the model.
    #
    # Args:
    # model: Full glmmADMB model object.
    # offset: Logical indicating whether the model includes an offset.
    # spec_os: A string specifying the offset as specified in the full model.
    # err_dist: A string specifying either "nbinom" for negative binomial or
    #    "gaussian" for gaussian.
    #
    # Returns:
    # List containing the conditional "R2c" and marginal "R2m" R2glmm values.
    #
    # Reformulate the full model to include only random effects and offset if
    # offset = TRUE, or just the random effects if offset = FALSE. Some code
```
if(offset==TRUE){
  parens <- function(x) paste0("(", x, ")")
  onlyBars <- function(form) reformulate(c(sapply(findbars(form),
    function(x) parens(deparse(x)),
    spec_os),
    response=".")
  null_model <- update(model, onlyBars(formula(model))))
} else {
  parens <- function(x) paste0("(" , x, ")")
  onlyBars <- function(form) reformulate(sapply(findbars(form),
    function(x) parens(deparse(x)),
    response=".")
  null_model <- update(model, onlyBars(formula(model))))
}

dm_M <- model.matrix(model)

Fixed <- vector("list", length(fixef(model)) - 1)

for(i in 2:length(fixef(model))){
  Fixed[[i-1]] <- fixef(model)[[i]] * dm_M[, i]
}

Fixed <- Reduce("+", Fixed)

VarF <- var(Fixed)

VarR <- vector("list", length(ranef(model)))

for(i in 1:length(ranef(model))){
  VarR[[i]] <- VarCorr(model)[[i]][1]
}

VarR <- Reduce("+", VarR)

if(err_dist == "nbinom"){
  R2m <- VarF/(VarF + VarR + log(model$alpha) *
                 log(1 + 1/exp(as.numeric(fixef(null_model)) +
                 1/as.numeric(model$alpha)))
  R2c <- (VarF + VarR)/
        (VarF + VarR + log(model$alpha) *
         log(1 + 1/exp(as.numeric(fixef(null_model)) +
         1/as.numeric(model$alpha)))

  R2 <- list()
  R2$R2m <- R2m
  R2$R2c <- R2c
  return(R2)
} else{
  R2m <- VarF/(VarF + VarR + model$alpha)
  R2c <- (VarF + VarR)/(VarF + VarR + model$alpha)

  R2 <- list()
  R2$R2m <- R2m
  R2$R2c <- R2c

  return(R2)
}
return(R2)
}
}

Mean Center

mean_center <- function(x) {
  (x-mean(x))/sd(x)
}

Data import and processing

Import data

ndata <- read.csv("ncrassa_maternal_effects_data.csv", stringsAsFactors = F)

Calculations used to infer the number of ascospores and conidia in germinated samples.

#total particle count (beads, conidia, ascospores, outliers) in the germinated samples

g.total <- ndata$g_total_count

g.beads<-ndata$g_beads_count #fluorescent beads in the germinated samples

g.germinated <- ndata$g_germinated_count #count of germinated ascospores in germinated sample

g.asco.conidia<-g.total-g.beads #count of particles that are not beads (ascospores and conidia)

a.total <- ndata$Cfreq + ndata$Hfreq + ndata$Pfreq + ndata$Ofreq
a.conidia <- ndata$Cfreq
a.hyaline <- ndata$Hfreq
a.pigmented <- ndata$Pfreq
a.outliers <- ndata$Ofreq

#add columns to "ndata" data frame for the percentage of conidia, beads, hyaline ascospores, pigmented ascospores, and outlier particles in the samples

ndata$a.percent.conidia <- a.conidia / a.total
ndata$a.percent.hyaline <- a.hyaline / a.total
ndata$a.percent.pigmented <- a.pigmented / a.total
ndata$a.percent.outliers <- a.outliers / a.total

# calculate mean particle percentages for each cross (mean of 4 replicates per cross)
ndata1 <- ddply(ndata,
 .(FEMALE, MALE),
  transform,
  mean.a.percent.conida = mean(na.omit(a.percent.conidia)),
  mean.a.percent.hyaline = mean(na.omit(a.percent.hyaline)),
  mean.a.percent.pigmented = mean(na.omit(a.percent.pigmented)),
  mean.a.percent.outliers = mean(na.omit(a.percent.outliers)))
# calculate the number of pigmented ascospores, hyaline ascospores, conidia, 
# and outlier particles in the germinated samples using the percentages from 
# the ungerminated samples. Note, g.asco.conidia and g.germinated need to be 
# reassigned because ddply rearranges dataframe.

g.asco.conidia <- ndata1$g_total_count - ndata1$g_beads_count

# percent germinated spores for trait correlogram
ndata1$percent.germinated <- g.germinated/(g.pigmented+g.germinated)

Create data variables for models (except spore size).

perithecia_count <- ndata1$Perithecia_count
pigmented_count <- ndata1$Pfreq
bead_count <- ndata1$Bfreq
pigmented_count_std <- ndata1$Pfreq / ndata1$TIME
percent_pigmented <- ndata1$Pfreq / (ndata1$Pfreq + ndata1$Hfreq) # used in trait correlogram
pigmented_size <- ndata1$PSSCWmean / ndata1$BSSCWmean

Trait Heatmaps

Data processing

seven_variables <- data.frame(ma, pa, matFemale, matMale, perithecia_count, total_ascospores_std, percent_pigmented, pigmented_size, perc.germinated)
perithecia_count = mean(na.omit(perithecia_count)),
asco_count = mean(na.omit(total_ascospores_std)),
percent_pigmented = mean(na.omit(percent_pigmented)),
pigmented_size = mean(na.omit(pigmented_size)),
person_germinated = mean(na.omit(perc_germinated))

mean_traits_s <- mean_traits[with(mean_traits, order(matFemale)), ]
mean_traits_s$sma <- factor(mean_traits_s$sma, levels=unique(as.character(mean_traits_s$sma)) )
mean_traits_s$s(pa <- factor(mean_traits_s$spa, levels=unique(as.character(mean_traits_s$spa)) )

mean_traits_sA <- subset(mean_traits_s, matFemale == "A")
mean_traits_sa <- subset(mean_traits_s, matFemale == "a")

Perithecia Count

sfg_perithecia <- c(min(mean_traits_s$perithecia_count, na.rm = T),
max(mean_traits_s$perithecia_count, na.rm = T))

#mat-A Mother
ggplot(mean_traits_sA, aes(ma, pa)) +
  geom_tile(aes(fill = perithecia_count), colour = "black") +
  scale_fill_gradient(low = "orange", high = "dark blue",
                      limits = sfg_perithecia, trans = "sqrt",
                      breaks = as.integer(c(200, 400, 600, 800, mean(na.omit(perithecia_count)))))+
  xlab(label = "Mother (mat-A)") +
  ylab(label = "Father (mat-a)") +
  labs(fill = "Perithecia\nCount") +
  theme_bw()+
  theme(axis.text.x=element_text(angle = 45, hjust=1, vjust=1))
```r
ggsave("bigA_perithecia_new.pdf", height = 3, width = 4)

# mat-a mother

ggplot(mean_traits_sa, aes(pa, ma)) +
  geom_tile(aes(fill = perithecia_count), colour = "black") +
  scale_fill_gradient(low = "orange", high = "dark blue",
                     limits = sfg_perithecia, trans = "sqrt",
                     breaks = as.integer(c(200, 400, 600, 800, mean(na.omit(perithecia_count))))) +
  xlab(label = "Father (mat-A)") +
  ylab(label = "Mother (mat-a)") +
  labs(fill = "Perithecia\nCount") +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1))
```
Total Spore Count

sfg_ascospore <- c(min(mean_traits_s$asco_count, na.rm = T),
                   max(mean_traits_s$asco_count, na.rm = T))

# mat-A mother
ggplot(mean_traits_sA, aes(ma, pa)) +
  geom_tile(aes(fill = asco_count), colour = "black") +
  scale_fill_gradient(low = "orange", high = "dark blue",
                     limits = sfg_ascospore, trans = "sqrt",
                     breaks = as.integer(c(0, 100, 200, 300, mean(na.omit(total_ascospores_std)))))+
  xlab(label = "Mother (mat-A)") +
  ylab(label = "Father (mat-a)") +
  labs(fill = "Ascospore\nCount") +
  theme_bw()+
  theme(axis.text.x=element_text(angle = 45, hjust=1, vjust=1))
```
#mat-a mother

ggplot(mean_traits_sa, aes(pa, ma)) +
  geom_tile(aes(fill = asco_count), colour = "black") +
  scale_fill_gradient(low = "orange", high = "dark blue",
                     limits = sfg_ascospore, trans = "sqrt",
                     breaks = as.integer(c(0, 100, 200, 300, mean(na.omit(total_ascospores_std))))) +
  xlab(label = "Father (mat-A)") +
  ylab(label = "Mother (mat-a)") +
  labs(fill = "Ascospore\nCount") +
  theme_bw() +
  theme(axis.text.x=element_text(angle = 45, hjust=1, vjust=1))
```
Percent Pigmented

```r
sfg_perc_pig <- c(min(mean_traits_s$percent_pigmented, na.rm = T),
                   max(mean_traits_s$percent_pigmented, na.rm = T))

# mat-A mother
ggplot(mean_traits_sA, aes(ma, pa)) +
  geom_tile(aes(fill = percent_pigmented), colour = "black") +
  scale_fill_gradient(low = "orange", high = "dark blue",
                     limits = sfg_perc_pig, breaks = c(.25, .50, .75, round(mean(na.omit(percent_pigmented)), 2)))+
  xlab(label = "Mother (mat-A)") +
  ylab(label = "Father (mat-a)") +
  labs(fill = "Proportion\nPigmented") +
  theme_bw()+
  theme(axis.text.x=element_text(angle = 45, hjust=1, vjust=1))
```
ggsave("BigA_Perc_Pigmented_new.pdf", height = 3, width = 4)

#mat-a mother

ggplot(mean_traits_sa, aes(pa, ma)) +
  geom_tile(aes(fill = percent_pigmented), colour = "black") +
  scale_fill_gradient(low = "orange", high = "dark blue",
                     limits = sfg_perc_pig,
                     breaks = c(.25, .50, .75, round(mean(na.omit(percent_pigmented)), 2)))+
  xlab(label = "Father (mat-A)") +
  ylab(label = "Mother (mat-a)") +
  labs(fill = "Proportion\nPigmented") +
  theme_bw()+
  theme(axis.text.x=element_text(angle = 45, hjust=1, vjust=1))

Pigmented Spore Size

sfg_pigsize <- c(min(mean_traits_s$pigmented_size, na.rm = T),
               max(mean_traits_s$pigmented_size, na.rm = T))

#mat-A mother
ggplot(mean_traits_sA, aes(ma, pa)) +
  geom_tile(aes(fill = pigmented_size), colour = "black") +
  scale_fill_gradient(low = "orange", high = "dark blue",
                     limits = sfg_pigsize,
                     breaks = c(1.7, 1.9, 2.1, round(mean(na.omit(pigmented_size)), 1))) +
  xlab(label = "Mother (mat-A)") +
  ylab(label = "Father (mat-a)") +
  labs(fill = "Pigmented\nSpore Size") +
  theme_bw() +
  theme(axis.text.x=element_text(angle = 45, hjust=1, vjust=1))
```r
#mat-a mother
ggplot(mean_traits_sa, aes(pa, ma)) +
  geom_tile(aes(fill = pigmented_size), colour = "black") +
  scale_fill_gradient(low = "orange", high = "dark blue",
                     limits = sfg_pigsize,
                     breaks = c(1.7, 1.9, 2.1, round(mean(na.omit(pigmented_size)), 1)))+
  xlab(label = "Father (mat-A)") +
  ylab(label = "Mother (mat-a)") +
  labs(fill = "Pigmented
Spore Size") +
  theme_bw()+
  theme(axis.text.x=element_text(angle = 45, hjust=1, vjust=1))
```
Pigmented Spore Germination

sfg_germ <- c(min(mean_traits_s$percent_germinated, na.rm = T),
             max(mean_traits_s$percent_germinated, na.rm = T))

#mat-A mother
ggplot(mean_traits_sA, aes(ma, pa)) +
  geom_tile(aes(fill = percent_germinated), colour = "black") +
  scale_fill_gradient(low = "orange", high = "dark blue",
                     limits = sfg_germ, trans = "sqrt",
                     breaks = c(.25, .5, .75, round(mean(na.omit(perc_germinated)), 2)))+
  xlab(label = "Mother (mat-A)") +
  ylab(label = "Father (mat-a)") +
  labs(fill = "Proportion\nGerminated") +
  theme_bw()+
  theme(axis.text.x=element_text(angle = 45, hjust=1, vjust=1))
ggsave("BigA_spore_germination_new.pdf", height = 3, width = 4)

#mat-a mother
ggplot(mean_traits_sa, aes(pa, ma)) +
geom_tile(aes(fill = percent_germinated), colour = "black") +
scale_fill_gradient(low = "orange", high = "dark blue",
limits = sfg_germ, trans = "sqrt",
breaks = c(.25, .5, .75, round(mean(na.omit(perc_germinated)), 2)))+
  xlab(label = "Father (mat-A)") +
  ylab(label = "Mother (mat-a)") +
  labs(fill = "Proportion\nGerminated") +
  theme_bw() +
  theme(axis.text.x=element_text(angle = 45, hjust=1, vjust=1))
Trait Correlogram

Supplementary Figure 4

```r
five_variables <- data.frame(
  perc_germinated, 
  pigmented_size, 
  percent_pigmented, 
  total_ascospores_std, 
  perithecia_count)

colnames(five_variables) <- c("Pigmented\nSpore\nGermination", 
  "Pigmented\nSpore Size", 
  "Proportion\nPigmented", 
  "Spore\nCount", 
  "Perithecia\nCount")

five_variables_complete <- five_variables[complete.cases(five_variables), ]

corrgram(five_variables_complete, 
  order=F, 
  upper.panel=panel.pts, 
  lower.panel=panel.conf, 
  pch=".")
```
Model Perithecia Count

Full Model

```r
perithecia_data <- data.frame(geo = mean_center(geo),
    geo_sq = mean_center(geo)^2,
    gen_dis = mean_center(gen_dis),
    gen_dis_sq = mean_center(gen_dis)^2,
    ma = ma,
    pa = pa,
    perithecia_count = perithecia_count)
perithecia_data <- perithecia_data[complete.cases(perithecia_data),]
```
#define model
MP1_perithecia <- glmmadmb(formula = perithecia_count ~
geo_sq +
gen_dis*geo +
gen_dis_sq +
(1 | ma) +
(1 | pa),
data = perithecia_data,
zeroInflation = FALSE,
family = "nbinom")

summary(MP1_perithecia)

## Call:
## glmmadmb(formula = perithecia_count ~ geo_sq + gen_dis * geo +
## gen_dis_sq + (1 | ma) + (1 | pa), data = perithecia_data,
## family = "nbinom", zeroInflation = FALSE)
##
## AIC: 10525.4
##
## Coefficients:
##             Estimate Std. Error  z value  Pr(>|z|)  
## (Intercept)   5.5066      0.1393    39.53   <2e-16 ***
## geo_sq       -0.0244      0.0164    -1.49     0.136
## gen_dis       0.0641      0.0378     1.69     0.090 .
## geo           0.0312      0.0347     0.90     0.368
## gen_dis_sq    0.0371      0.0155     2.39     0.017 *
## gen_dis:geo   -0.0015      0.0256    -0.06     0.953
##
## Number of observations: total=884, ma=21, pa=21
## Random effect variance(s):
## Group=ma
## Variance StdDev
## (Intercept)  0.3606  0.6005
## Group=pa
## Variance StdDev
## (Intercept)  0.0316  0.1779
##
## Negative binomial dispersion parameter: 8.5772 (std. err.: 0.42802)
##
## Log-likelihood: -5253.71

Calculate qAIC

chat <- dfun(MP1_perithecia)
dredge_MP1_perithecia <- dredge(MP1_perithecia, rank = "QAI C", chat = chat)

## Fixed term is "(Intercept)"

head(dredge_MP1_perithecia)
## Global model call: glmmadmb(formula = perithecia_count ~ geo_sq + gen_dis +
##     gen_dis_sq + (1 | ma) + (1 | pa), data = perithecia_data,
##     family = "nbinom", zeroInflation = FALSE)
## ---
## Model selection table
##   (Int) gen_dis gen_dis_sq geo geo_sq gen_dis:geo df logLik
## 4   5.478 0.08491 0.04089 6 -5254.84
## 12  5.495 0.07996 0.04014 -0.01657 7 -5254.13
## 16  5.507 0.06475 0.03688 0.03134 8 -5253.72
## 8   5.479 0.08327 0.04052 0.00403 7 -5254.83
## 15  5.540 0.01392 0.06002 -0.03448 7 -5255.32
## 32  5.507 0.06411 0.03712 0.03120 -0.02439 -0.001491 9 -5253.71
## QAIC delta weight
## 4   10521.7 0.00 0.343
## 12  10522.3 0.58 0.257
## 16  10523.4 1.76 0.142
## 8   10523.7 1.98 0.127
## 15  10524.6 2.96 0.078
## 32  10525.4 3.74 0.053
## Models ranked by QAIC(x, chat = 0.957642499870479)
## Random terms (all models):
##   '1 | ma', '1 | pa'

Selected Model

#qAIC includes only gen and gen_dis_sq
MP1_perithecia <- glmmadmb(formula = perithecia_count ~
    gen_dis +
    gen_dis_sq +
    (1 | ma) +
    (1 | pa),
    data = perithecia_data,
    zeroInflation = FALSE,
    family = "nbinom")

summary(MP1_perithecia)

##
## Call:
## glmmadmb(formula = perithecia_count ~ gen_dis +
##     gen_dis_sq +
##     (1 | ma) +
##     (1 | pa),
##     data = perithecia_data, family = "nbinom",
##     zeroInflation = FALSE)
##
## AIC: 10521.7
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 5.4782     0.1373   39.89  <2e-16 ***
## gen_dis    0.0849     0.0319    2.66  0.0078 **
## gen_dis_sq 0.0409     0.0146    2.81  0.0050 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of observations: total=884, ma=21, pa=21
## Random effect variance(s):
## Group=ma
##                  Variance  StdDev
## (Intercept) 0.3566  0.5972
## Group=pa
##                  Variance  StdDev
## (Intercept) 0.03172 0.1781
##
## Negative binomial dispersion parameter: 8.5522 (std. err.: 0.42667)
##
## Log-likelihood: -5254.84

R2GLMM of selected model.

R2m is the marginal (fixed effects) R2glmm value and R2c is the conditional R2glmm value (fixed + random effects)

```r
perithecia_R2glmm <- glmmadmbR2(model = MP1_perithecia,
                                 offset = FALSE,
                                 err_dist = "nbinom")
print(perithecia_R2glmm)
```

```
## $R2m
## [1] 0.00517726
##
## $R2c
## [1] 0.6151382
```

Variance of maternal vs. paternal random effect coefficients (BLUPs) of selected model

```r
MP1_perithecia_ranef <- ranef(MP1_perithecia)

r.pa.peri <- MP1_perithecia_ranef$pa  # paternal random effect coefficients
r.ma.peri <- MP1_perithecia_ranef$ma  # maternal random effect coefficients

var.test(r.pa.peri, r.ma.peri)
```

```
##
## F test to compare two variances
##
## data:  r.pa.peri and r.ma.peri
## F = 0.0756, num df = 20, denom df = 20, p-value = 3.004e-07
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
## 0.03067471 0.18630844
## sample estimates:
## ratio of variances
## 0.07559733
```
Correlation between maternal and paternal BLUPs of selected model

```r
lm_ranef_peri <- lm(r.pa.peri ~ r.ma.peri)
summary(lm_ranef_peri)
```

```
##
## Call:
## lm(formula = r.pa.peri ~ r.ma.peri)
##
## Residuals:
## Min 1Q Median 3Q Max
## -0.28432 -0.11600 -0.06087 0.13873 0.30559
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0001478  0.0372401 -0.004 0.997
## r.ma.peri   0.0225548  0.0628652  0.359 0.724
##
## Residual standard error: 0.1707 on 19 degrees of freedom
## Multiple R-squared: 0.006729, Adjusted R-squared: -0.04555
## F-statistic: 0.1287 on 1 and 19 DF, p-value: 0.7237
```

Model Total Ascospore Count

Full Model

```r
scd <- data.frame(logtime, geo, gen_dis, ma, pa, total_char_ascospores, bead_count)

scd$bead_count[scd$bead_count==0] <- NA
scd <- scd[complete.cases(scd), ]
spore_count_data <- data.frame(logtime = scd$logtime, 
geo = mean_center(scd$geo), 
geo_sq = mean_center(scd$geo)^2, 
gen_dis = mean_center(scd$gen_dis), 
gen_dis_sq = mean_center(scd$gen_dis)^2, 
ma = scd$ma, 
pa = scd$pa, 
total_char_ascospores = scd$total_char_ascospores)

MP1_noff <- glmmadmb(formula = total_char_ascospores ~ 
geo_sq + 
gen_dis*geo + 
gen_dis_sq + 
(1 | ma) + 
(1 | pa) + 
offset(logtime),
```

22
data = spore_count_data,
zeroInflation = FALSE,
family = "nbinom")

summary(MP1_noff)

## Call:
glmmadmb(formula = total_char_ascospores ~ geo_sq + gen_dis *
geo + gen_dis_sq + (1 | ma) + (1 | pa) + offset(logtime),
data = spore_count_data, family = "nbinom", zeroInflation = FALSE)
## AIC: 12643.4
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) 3.4855 0.2192 15.90 <2e-16 ***
## geo_sq -0.0463 0.0361 -1.28 0.2001
## gen_dis -0.1312 0.0827 -1.59 0.1126
## geo 0.2370 0.0754 3.14 0.0017 **
## gen_dis_sq -0.1065 0.0336 -3.17 0.0015 **
## gen_dis:geo 0.0772 0.0551 1.40 0.1612
## ---
## Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
## Number of observations: total=882, ma=21, pa=21
## Random effect variance(s):
## Group=ma
## Variance StdDev
## (Intercept) 0.733 0.8561
## Group=pa
## Variance StdDev
## (Intercept) 0.201 0.4483
## Negative binomial dispersion parameter: 1.9077 (std. err.: 0.087565)
## Log-likelihood: -6312.71

Calculate qAIC

chat <- dfun(MP1_noff)
dredge_MP1_noff <- dredge(MP1_noff, rank = "QAIC", chat = chat)

## Fixed term is ";(Intercept)"

head(dredge_MP1_noff)

## Global model call: glmmadmb(formula = total_char_ascospores ~ geo_sq + gen_dis *
geo + gen_dis_sq + (1 | ma) + (1 | pa) + offset(logtime),
data = spore_count_data, family = "nbinom", zeroInflation = FALSE)
## ---
## Model selection table
## (Int) gen_dis gen_dis_sq geo geo_sq gen_dis:geo off(lgt) df

23
Selected Model

# QAIC includes all but geo_sq and gen_dis*geo
MP1_noff <- glmmadmb(formula = total_char_ascospores ~ geo +
  #geo_sq +
  #gen_dis*geo +
  gen_dis +
  gen_dis_sq +
  (1 | ma) +
  (1 | pa) +
  offset(logtime),
  data = spore_count_data,
  zeroInflation = FALSE,
  family = "nbinom")

summary(MP1_noff)

## Call:
## glmmadmb(formula = total_char_ascospores ~ geo + gen_dis + gen_dis_sq +
##         (1 | ma) + (1 | pa) + offset(logtime), data = spore_count_data,
##         family = "nbinom", zeroInflation = FALSE)
##
## AIC: 12642.9

## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.4318    0.2168  15.83  <2e-16 ***
## geo         0.1847    0.0655   2.82   0.0048 **
## gen_dis     -0.1273    0.0731  -1.74   0.0814 .
## gen_dis_sq  -0.0853    0.0318  -2.68   0.0073 **

## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Number of observations: total=882, ma=21, pa=21
## Random effect variance(s):
## Group=ma
## Variance StdDev
## (Intercept) 0.74 0.8602
## Group=pa
## Variance StdDev
## (Intercept) 0.2105 0.4588
##
## Negative binomial dispersion parameter: 1.903 (std. err.: 0.087327)
##
## Log-likelihood: -6314.43

R2GLMM of selected model.

R2m is the marginal (fixed effects) R2glm value and R2c is the conditional R2glm value (fixed + random effects)

```r
spore_count_R2glm <- glmmadmbR2(model = MP1_noff, 
                               offset=TRUE, 
                               spec_os = "offset(logtime)", 
                               err_dist = "nbinom")
print(spore_count_R2glm)
```

```
## $R2m
## [1] 0.04349842
##
## $R2c
## [1] 0.7785641
```

Variance of maternal vs. paternal random effect coefficients (BLUPs) of selected model

```r
MP1noff_ranef <- ranef(MP1_noff)

r.pa.spore <- MP1noff_ranef$pa
r.ma.spore <- MP1noff_ranef$ma

var.test(r.pa.spore, r.ma.spore)
```

```
## F test to compare two variances
##
## data:  r.pa.spore and r.ma.spore
## F = 0.2582, num df = 20, denom df = 20, p-value = 0.003878
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
## 0.1047543 0.6362447
## sample estimates:
## ratio of variances
## 0.2581654
```
Correlation between maternal and paternal BLUPs of selected model

```r
lm_ranef_spore <- lm(r.pa.spore ~ r.ma.spore)
summary(lm_ranef_spore)
```

```
##
## Call:
## lm(formula = r.pa.spore ~ r.ma.spore)
##
## Residuals:
##    Min     1Q   Median     3Q    Max
## -0.52935 -0.43410  0.09129  0.29621  0.63146
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.000791  0.090887  -0.009  0.993
## r.ma.spore   0.194374  0.107699   1.805  0.087 .
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4165 on 19 degrees of freedom
## Multiple R-squared: 0.1463, Adjusted R-squared: 0.1014
## F-statistic: 3.257 on 1 and 19 DF,  p-value: 0.08698
```

Model Proportion Pigmented

Full Model

```r
# add log of total ascospores for offset to dataframe used for pigmented spore count model
pigmented_count_data <- data.frame(geo,
                                   gen_dis,
                                   ma,
                                   pa,
                                   pigmented_count)

#add log of total ascospores for offset to dataframe used for pigmented spore count model
total_char_ascospores_na<-total_char_ascospores
total_char_ascospores_na[total_char_ascospores_na==0]<-NA
pigmented_count_data$logTotalAscospores <- log(total_char_ascospores_na)
pigmented_count_data <- pigmented_count_data[complete.cases(pigmented_count_data), ]
pigmented_count_data <- data.frame (geo = mean_center(pigmented_count_data$geo),
                                    geo_sq = mean_center(pigmented_count_data$geo)^2,
                                    gen_dis = mean_center(pigmented_count_data$gen_dis),
                                    gen_dis_sq = mean_center(pigmented_count_data$gen_dis)^2,
                                    ma = pigmented_count_data$ma,
                                    pa = pigmented_count_data$pa,
                                    pigmented_count = pigmented_count_data$pigmented_count,
                                    logTotalAscospores = pigmented_count_data$logTotalAscospores)

MP1_perc_pig <- glmmadmb(formula = pigmented_count ~
                          geo_sq +
                          gen_dis*geo +
                          gen_dis_sq +
```
(1 | ma) +
(1 | pa) +
offset(logTotalAscospores),
data = pigmented_count_data,
zeroInflation = FALSE,
family = "nbinom")

summary(MP1_perc_pig)

##
## Call:
## glmmadmb(formula = pigmented_count ~ geo_sq + gen_dis * geo +
## gen_dis_sq + (1 | ma) + (1 | pa) + offset(logTotalAscospores),
## data = pigmented_count_data, family = "nbinom", zeroInflation = FALSE)
##
## AIC: 10085.6
##
## Coefficients:
##            Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.3997    0.0948  -4.22  2.5e-05 ***
## geo_sq      -0.0717    0.0159   -4.50  6.9e-06 ***
## gen_dis     -0.1814    0.0359  -5.05   4.5e-07 ***
## geo         0.1033    0.0335   3.09   0.0020 **
## gen_dis_sq  -0.0540    0.0145  -3.72   0.0002 ***
## gen_dis:geo -0.0614    0.0234  -2.63   0.0086 **
## ---
## Signif. codes:  
##  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of observations: total=881, ma=21, pa=21
## Random effect variance(s):
## Group=ma
##  Variance StdDev
## (Intercept)  0.09062  0.301
## Group=pa
##  Variance StdDev
## (Intercept)  0.08404  0.2899
##
## Negative binomial dispersion parameter: 11.013 (std. err.: 0.61238)
##
## Log-likelihood: -5033.79

Calculate qAIC

cchat <- dfun(MP1_perc_pig)
ddredge_MP1_perc_pig <- dredge(MP1_perc_pig, rank = "QAIC", chat = cchat)

## Fixed term is "(Intercept)"

head(ddredge_MP1_perc_pig)

## Global model call: glmmadmb(formula = pigmented_count ~ geo_sq + gen_dis * geo +
##   gen_dis_sq + (1 | ma) + (1 | pa) + offset(logTotalAscospores),
##   data = pigmented_count_data, family = "nbinom", zeroInflation = FALSE)
## ---
## Model selection table
## (Int) gen_dis gen_dis_sq geo geo_sq gen_dis:geo off(lTA) df
## 64 -0.3997 -0.18140 -0.05401 0.10330 -0.07171 -0.06141 + 9
## 48 -0.4017 -0.15210 -0.06349 0.10600 -0.07016 + 8
## 62 -0.4592 -0.08598 0.06909 -0.06193 -0.08255 + 8
## 44 -0.4385 -0.09904 -0.05141 -0.04509 + 7
## 56 -0.4812 -0.12580 -0.04331 0.02772 -0.05956 + 8
## 36 -0.4845 -0.08565 -0.04962 + 6
## logLik QAIC delta weight
## 64 -5033.79 10085.6 0.00 0.915
## 48 -5037.22 10090.4 4.86 0.081
## 62 -5040.56 10097.1 11.54 0.003
## 44 -5042.15 10098.3 12.72 0.002
## 56 -5043.79 10103.6 18.00 0.000
## 36 -5047.40 10106.8 21.22 0.000
## Models ranked by QAIC(x, chat = 0.823629062022813)
## Random terms (all models):
## '1 | ma', '1 | pa'

**Selected Model**

Selected model includes all terms.

**R2GLMM of selected model.**

R2m is the marginal (fixed effects) R2glmm value and R2c is the conditional R2glmm value (fixed + random effects)

```r
perc_pig_R2glmm<-glmmadmbR2(model = MP1_perc_pig, 
offset = TRUE, 
spec_os = "offset(logTotalAscospores)", 
err_dist = "nbinom")
print(perc_pig_R2glmm)
```

```r
## $R2m
## [1] 0.005010005
##
## $R2c
## [1] 0.07081422
```

**Variance of maternal vs. paternal random effect coefficients (BLUPs) of selected model**

```r
MP1perc_pig_ranef<-ranef(MP1_perc_pig)
r.pa.pig.perc<-MP1perc_pig_ranef$pa
r.ma.pig.perc<-MP1perc_pig_ranef$ma
var.test(r.pa.pig.perc, r.ma.pig.perc)
```

```r
## F test to compare two variances
```
Correlation between maternal and paternal BLUPs of selected model

```r
lm_ranef_pig_perc<-lm(r.pa.pig.perc ~ r.ma.pig.perc)
summary(lm_ranef_pig_perc)
```

Model Pigmented Spore Size

Full Model

```r
PSSCW_std <- pigmented_size

PSSCWdf <- data.frame(PSSCW_std, 
    gen_dis, 
    geo, 
    ma, 
    pa)

PSSCWdf <- PSSCWdf[complete.cases(PSSCWdf), ]

PSSCWdf <- data.frame(PSSCW_std = PSSCWdf$PSSCW_std, 
    gen_dis = mean_center(PSSCWdf$gen_dis), 
    # other transformations as needed)
```
$$\text{gen\_dis\_sq} = \text{mean\_center}(\text{PSSCWdf}\$\text{gen\_dis})^2,$$
$$\text{geo} = \text{mean\_center}(\text{PSSCWdf}\$\text{geo}),$$
$$\text{geo\_sq} = \text{mean\_center}(\text{PSSCWdf}\$\text{geo})^2,$$
$$\text{ma} = \text{PSSCWdf}\$\text{ma},$$
$$\text{pa} = \text{PSSCWdf}\$\text{pa})$$

```r
MP1_Pigmented_size <- \text{glmmadmb(formula = PSSCW\_std -}
geo*gen\_dis +
geo\_sq +
gen\_dis\_sq +
(1 | ma) +
(1 | pa),
data = PSSCWdf,
zeroInflation=FALSE,
family="\text{gaussian}")
```

```r
summary(MP1_Pigmented_size)
```

```r
## Call:
## \text{glmmadmb(formula = PSSCW\_std - geo \ast gen\_dis + geo\_sq + gen\_dis\_sq +}
## (1 | ma) + (1 | pa), data = PSSCWdf, family = "\text{gaussian}",
## zeroInflation = FALSE)
## ## AIC: -1298.6
## ## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) 1.852074 0.028322 65.39 <2e-16 ***
## geo 0.025042 0.010888 2.30 0.021 *
## gen\_dis -0.007022 0.011542 -0.61 0.543
## geo\_sq -0.012859 0.005148 -2.50 0.012 *
## gen\_dis\_sq -0.005114 0.004667 -1.10 0.273
## geo:gen\_dis -0.000276 0.007779 -0.04 0.972
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## ## Number of observations: total=880, ma=21, pa=21
## ## Random effect variance(s):
## Group=ma
## Variance StdDev
## (Intercept) 0.008371 0.09149
## Group=pa
## Variance StdDev
## (Intercept) 0.00703 0.08384
## ## Residual variance: 0.10611 (std. err.: 0.0025908)
## ## Log-likelihood: 658.319
```
Calculate AIC

dredge_MP1_Pigmented_Size <- dredge(MP1_Pigmented_size)

## Fixed term is "(Intercept)"

head(dredge_MP1_Pigmented_Size)

## Global model call: glmmadmb(formula = PSSCW_std ~ geo * gen_dis + geo_sq + gen_dis_sq +
## (1 | ma) + (1 | pa), data = PSSCWdf, family = "gaussian",
## zeroInflation = FALSE)

## ---
## Model selection table
## (Int) gen_dis gen_dis_sq geo geo_sq gen_dis:geo df logLik
## 13 1.848 0.02589 -0.013430 6 657.416
## 15 1.849 -0.002727 0.02178 -0.011750 7 658.122
## 14 1.846 0.003946 0.02192 -0.011940 7 657.663
## 16 1.852 -0.006897 -0.005157 0.02506 -0.012860 8 658.319
## 30 1.846 0.001969 0.02199 -0.011960 -0.00252 8 657.720
## 11 1.846 -0.004615 -0.007029 6 655.528

## AICc delta weight
## 13 -1302.7 0.00 0.351
## 15 -1302.1 0.62 0.258
## 14 -1301.2 1.54 0.163
## 16 -1300.5 2.26 0.113
## 30 -1299.3 3.46 0.062
## 11 -1299.0 3.78 0.053

## Models ranked by AICc(x)

## Random terms (all models):
## '1 | ma', '1 | pa'

Selected Model

#only includes geo and geo_sq

MP1_Pigmented_size <- glmmadmb(formula = PSSCW_std -
                           #geo*gen_dis +
                           geo +
                           geo_sq +
                           #gen_dis_sq +
                           (1 | ma) +
                           (1 | pa),
                           data = PSSCWdf,
                           zeroInflation=FALSE,
                           family="gaussian")

R2GLMM of selected model.

R2m is the marginal (fixed effects) R2glmm value and R2c is the conditional R2glmm value (fixed + random effects)
pigm_size_R2glmm <- glmmadmbR2(model = MP1_Pigmented_size, 
                             offset = FALSE, 
                             err_dist = "gaussian")

print(pigm_size_R2glmm)

## $R2m
## [1] 0.002374187
##
## $R2c
## [1] 0.1283417

Variance of maternal vs. paternal random effect coefficients (BLUPs) of selected model

MP1_Pigmented_size_ranef <- ranef(MP1_Pigmented_size)

r.pa.psize <- MP1_Pigmented_size_ranef$pa
r.ma.psize <- MP1_Pigmented_size_ranef$ma

var.test(r.pa.psize, r.ma.psize)

## F test to compare two variances
## data: r.pa.psize and r.ma.psize
## F = 0.8206, num df = 20, denom df = 20, p-value = 0.6626
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
## 0.332972 2.022367
## sample estimates:
## ratio of variances
## 0.8206044

Correlation between maternal and paternal BLUPs of selected model

lm_ranef_pigsize <- lm(r.pa.psize ~ r.ma.psize)
summary(lm_ranef_pigsize)

## Call:
## lm(formula = r.pa.psize ~ r.ma.psize)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -0.082840 -0.024166  0.005716  0.027876  0.078675
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.790e-07  9.449e-03  0.000 1
## r.ma.psize   7.760e-01  1.072e-01  7.238 7.18e-07 ***
## ---

32
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.0433 on 19 degrees of freedom
Multiple R-squared: 0.7338, Adjusted R-squared: 0.7198
F-statistic: 52.38 on 1 and 19 DF,  p-value: 7.175e-07

Model Pigmented Spore Germination

Full Model

g_pigmented <- g.pigmented + g.germinated
g_pigmented[g_pigmented==0] <- NA
Log_pigmented <- log(g_pigmented)
gd <- data.frame(germ, geo, gen_dis, ma, pa, Log_pigmented)
gd <- gd[complete.cases(gd),]

germ_data <- data.frame(germ = gd$germ,
geo = mean_center(gd$geo),
geo_sq = mean_center(gd$geo)^2,
gen_dis = mean_center(gd$gen_dis),
gen_dis_sq = mean_center(gd$gen_dis)^2,
ma = gd$ma,
pa = gd$pa,
Log_pigmented = gd$Log_pigmented)

M2_germ_pigmented<- glmmadmb(formula = germ ~
geo_sq +
gen_dis*geo +
gen_dis_sq +
(1 | ma) +
(1 | pa) +
offset(Log_pigmented),
data = germ_data,
zeroInflation=F,
family="nbinom")

summary(M2_germ_pigmented)

# Call:
glmmadmb(formula = germ ~ geo_sq + gen_dis * geo + gen_dis_sq +
(1 | ma) + (1 | pa) + offset(Log_pigmented), data = germ_data,
family = "nbinom", zeroInflation = F)

# AIC: 8756.4

33
## Coefficients:

| Estimate | Std. Error | z value | Pr(>|z|) |
|----------|------------|---------|----------|
| -1.0478  | 0.1345     | -7.79   | 6.8e-15 *** |
| 0.0214   | 0.0265     | 0.81    | 0.419    |
| -0.0314  | 0.0585     | -0.54   | 0.591    |
| -0.0577  | 0.0584     | -0.99   | 0.323    |
| -0.0526  | 0.0262     | -2.01   | 0.044 *  |
| 0.0112   | 0.0426     | 0.26    | 0.792    |

---

Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

Number of observations: total=875, ma=21, pa=21

Random effect variance(s):

<table>
<thead>
<tr>
<th>Group</th>
<th>Variance</th>
<th>StdDev</th>
</tr>
</thead>
<tbody>
<tr>
<td>ma</td>
<td>0.3055</td>
<td>0.5528</td>
</tr>
<tr>
<td>pa</td>
<td>0.03329</td>
<td>0.1825</td>
</tr>
</tbody>
</table>

Negative binomial dispersion parameter: 3.1676 (std. err.: 0.16625)

Log-likelihood: -4369.18

Calculate qAIC

```r
dfun <- dfun(M2_germ_pigmented)
dredge_M2_germ_pigmented <- dredge(M2_germ_pigmented, rank = "QAIC", chat = chat)
```

Fixed term is "(Intercept)"

```r
head(dredge_M2_germ_pigmented)
```

Global model call: glmmadmb(formula = germ ~ geo_sq + gen_dis * geo + gen_dis_sq + (1 | ma) + (1 | pa) + offset(Log_pigmented), data = germ_data, family = "nbinom", zeroInflation = F)

---

Model selection table

<table>
<thead>
<tr>
<th>(Int)</th>
<th>gen_dis</th>
<th>gen_dis_sq</th>
<th>geo</th>
<th>geo_sq</th>
<th>off(Log_pgm)</th>
<th>df</th>
<th>logLik</th>
</tr>
</thead>
<tbody>
<tr>
<td>35</td>
<td>-1.047</td>
<td>-0.02932</td>
<td>+</td>
<td></td>
<td></td>
<td>5</td>
<td>-4370.49</td>
</tr>
<tr>
<td>36</td>
<td>-1.021</td>
<td>-0.06147</td>
<td>-0.05528</td>
<td>+</td>
<td></td>
<td>6</td>
<td>-4369.75</td>
</tr>
<tr>
<td>39</td>
<td>-1.044</td>
<td>-0.03252</td>
<td>-0.03915</td>
<td>+</td>
<td></td>
<td>6</td>
<td>-4370.07</td>
</tr>
<tr>
<td>43</td>
<td>-1.054</td>
<td>-0.02984</td>
<td>0.007514</td>
<td>+</td>
<td></td>
<td>6</td>
<td>-4370.43</td>
</tr>
<tr>
<td>47</td>
<td>-1.068</td>
<td>-0.03727</td>
<td>-0.07481</td>
<td>0.027840</td>
<td>+</td>
<td>7</td>
<td>-4369.43</td>
</tr>
<tr>
<td>40</td>
<td>-1.023</td>
<td>-0.05278</td>
<td>-0.05392</td>
<td>-0.02858</td>
<td>+</td>
<td>7</td>
<td>-4369.54</td>
</tr>
</tbody>
</table>
```

QAIC delta weight

| 35    | 8751.0  | 0.000 | 0.289 |
| 36    | 8751.5  | 0.052 | 0.223 |
| 39    | 8752.1  | 1.16  | 0.162 |
| 43    | 8752.9  | 1.88  | 0.113 |
| 47    | 8752.9  | 1.88  | 0.113 |
| 40    | 8753.1  | 2.10  | 0.101 |

34
## Models ranked by QAIC(x, chat = 0.759348818743698)
## Random terms (all models):
## '1 | ma', '1 | pa'

Selected Model

`#qAIC indicates only to keep gen_dis_sq`

```r
M2_germ_pigmented <- glmmadmb(formula = germ ~
  # geo +
  # geo_sq +
  # gen_dis*geo +
  # gen_dis +
  gen_dis_sq +
  (1 | ma) +
  (1 | pa) +
  offset(Log_pigmented),
  data = germ_data,
  zeroInflation = FALSE,
  family = "nbinom")
```

```r
summary(M2_germ_pigmented)
```

```
## Call:
## glmmadmb(formula = germ ~ gen_dis_sq + (1 | ma) + (1 | pa) +
##     offset(Log_pigmented), data = germ_data, family = "nbinom",
##     zeroInflation = FALSE)
##
## AIC: 8751
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -1.0470    0.1303  -8.045  9.2e-16 ***
## gen_dis_sq  -0.0293    0.0133  -2.200    0.028 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Number of observations: total=875, ma=21, pa=21
## Random effect variance(s):
## Group=ma
##   Variance StdDev  
## (Intercept)  0.309  0.5559
## Group=pa
##   Variance StdDev  
## (Intercept)  0.0352 0.1877

## Negative binomial dispersion parameter: 3.1591 (std. err.: 0.16545)
## Log-likelihood:  -4370.49
```

35
R2GLMM of selected model.

R2m is the marginal (fixed effects) R2glmm value and R2c is the conditional R2glmm value (fixed + random effects)

germ_pig_R2glmm <- glmmadmbR2(model = M2_germ_pigmented,
                             offset = TRUE,
                             spec_os = "offset(Log_pigmented)",
                             err_dist = "nbinom")

print(germ_pig_R2glmm)

## $R2m
## [1] 0.0015357
##
## $R2c
## [1] 0.172754

Variance of maternal vs. paternal random effect coefficients (BLUPs) of selected model

M2_germ_pigmented_ranef <- ranef(M2_germ_pigmented)

r.pa.germ.pig <- M2_germ_pigmented_ranef$pa
r.ma.germ.pig <- M2_germ_pigmented_ranef$ma

var.test(r.pa.germ.pig, r.ma.germ.pig)

## F test to compare two variances
##
## data: r.pa.germ.pig and r.ma.germ.pig
## F = 0.0874, num df = 20, denom df = 20, p-value = 1.051e-06
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
## 0.03545748 0.21535748
## sample estimates:
## ratio of variances
## 0.0873844

Correlation between maternal and paternal BLUPs of selected model

lm_ranef_germ_pig <- lm(r.pa.germ.pig ~ r.ma.germ.pig)
summary(lm_ranef_germ_pig)

## Call:
## lm(formula = r.pa.germ.pig ~ r.ma.germ.pig)
##
## Residuals:
##   Min     1Q Median     3Q    Max
## -0.40296 -0.06875  0.01323  0.10096  0.34925
##
## Coefficients:
Plot variance explained by models

r2glmm_list <- list(perithecia_R2glmm, spore_count_R2glmm, perc_pig_R2glmm, pigm_size_R2glmm, germ_pig_R2glmm)

fixed_effects_parser <- function(x){return(x[[1]])}
fixed_effects_variance <- unlist(lapply(r2glmm_list, fixed_effects_parser))

random_effects_parser <- function(x){return(x[[2]]-x[[1]])}
random_effects_variance <- unlist(lapply(r2glmm_list, random_effects_parser))

explained_variance <- fixed_effects_variance + random_effects_variance
unexplained_variance <- 1 - explained_variance

all_models <- list(MP1_perithecia, MP1_noff, MP1_perc_pig, MP1_Pigmented_size, M2_germ_pigmented)

ma_getter <- function(x){x$S$ma[1,1]}
ma_variances <- unlist(lapply(X=all_models, ma_getter))

pa_getter <- function(x){x$S$pa[1,1]}
pa_variances <- unlist(lapply(X=all_models, pa_getter))

model_names<-'c("Perithecia \nCount", "Spore \nCount", "Proportion \nPigmented", "Pigmented \nSpore Size", "Pigmented \nSpore \nGermination")

ma_pa_variances <- data.frame(Female=ma_variances, Male=pa_variances, Trait=model_names)
ma_pa_variances <- melt(ma_pa_variances, value.name=Trait)

## Using Trait as id variables

ma_pa_variances$Trait <- factor(ma_pa_variances$Trait, as.character(ma_pa_variances$Trait))

## Warning in `levels<-`(`*tmp*`, value = if (nl == nL) as.character(labels)
## else paste0(labels, : duplicated levels in factors are deprecated
proportion_ma_pa_variance<-ddply(ma_pa_variances,
  .(Trait),
  transform,
  prop=value/sum(value))

## Warning in `levels<-`(`*tmp*`, value = if (nl == nL) as.character(labels)
## else paste0(labels, : duplicated levels in factors are deprecated

colnames(proportion_ma_pa_variance) <- c("Trait",
  "Sex",
  "value",
  "prop")

fixed_other<-data.frame(Fixed.Effects = fixed_effects_variance,
  Unexplained.Variance = unexplained_variance,
  Trait = model_names)

fixed_other$Female <- subset(proportion_ma_pa_variance, Sex == "Female")$prop * random_effects_variance
fixed_other$Male <- subset(proportion_ma_pa_variance, Sex == "Male")$prop * random_effects_variance
fixed_other_melt <- melt(fixed_other, id.vars = c("Trait"))
fixed_other_melt$variable <- factor(fixed_other_melt$variable, c("Female",
  "Male",
  "Fixed.Effects",
  "Unexplained.Variance"))

fixed_other_melt$Trait <- factor(fixed_other_melt$Trait, c("Perithecia \nCount",
  "Spore \nCount",
  "Proportion \nPigmented",
  "Pigmented \nSpore Size",
  "Pigmented \nSpore \nGermination"))

colnames(fixed_other_melt) <- c("Trait", "Component", "value")

cbbPalette <- c("#000000", "#E69F00", "#9F00E6", "#B8B8B8")

ggplot(fixed_other_melt, aes(x=Trait, y=value)) +
  geom_bar(stat="identity",aes(fill = Component, order = Component), width=.75) +
  theme_bw() +
  scale_fill_manual(values=cbbPalette, labels = c("Female",
    "Male",
    "Crossing Distance",
    "Unexplained Variance")) +
  xlab("Trait") +
  ylab("Proportion of Variance")
Plot maternal and paternal proportions of random effect variance

```r
ggplot(proportion_ma_pa_variance, aes(x=Trait, y=prop)) + geom_bar(stat="identity",aes(fill = Sex), width=.75) + theme_bw() + scale_fill_manual(values=cbbPalette, name="Component") + annotate("text", x=1, y=1, label="***") + annotate("text", x=2, y=1, label="**") + annotate("text", x=5, y=1, label="***") + xlab("Trait") + ylab("Proportion of Variance of Random Effects")
```
Plot of maternal vs paternal BLUPs

```r
predf <- data.frame(r.ma.peri, r.pa.peri,
                     Model = "Perithecia\nCount")
colnames(predf) <- c("Female", "Male", "Model")

taredf <- data.frame(r.ma.spore, r.pa.spore,
                     Model = "Spore\nCount")
colnames(taredf) <- c("Female", "Male", "Model")

ppredf <- data.frame(r.ma.pig.perc, r.pa.pig.perc,
                     Model = "Proportion\nPigmented")
colnames(ppredf) <- c("Female", "Male", "Model")

psredf <- data.frame(r.ma.psize, r.pa.psize,
                     Model = "Pigmented Spore\nSize")
colnames(psredf) <- c("Female", "Male", "Model")

ppgredf <- data.frame(r.ma.germ.pig, r.pa.germ.pig,
                      Model = "Pigmented Spore\nGermination")
colnames(ppgredf) <- c("Female", "Male", "Model")
```
melt_ranef_facets <- rbind(predf, taredf, ppredf, psredf, ppgredf)

ggplot(melt_ranef_facets, aes(x=Male, y=Female)) +
  geom_point(size=1.5) +
  theme_bw(10) +
  stat_smooth(geom = "smooth",
               method = "lm",
               formula = y~x,
               se = TRUE,
               colour = "black") +
  facet_wrap(~Model, ncol = 5, scales = "free") +
  theme(strip.background = element_blank(),
        axis.text.x=element_text(angle = 45, hjust=1, vjust=1))

HPD intervals for parameters

MCMC parameter estimation

Perithecia Count

MCMC_perithecia <- glmmadmb(formula = perithecia_count ~
    gen_dis +
    gen_dis_sq +
    (1 | ma) +
    (1 | pa),
    data = perithecia_data,
    zeroInflation = FALSE,
    family = "nbinom",
    mcmc = TRUE,
    mcmc.opts=mcmcControl(mcmc=10000))

## Loading required package: coda
## Loading required package: lattice

## Warning: package 'lattice' was built under R version 3.1.2

perithecia_mcmc_coda <- as.mcmc(MCMC_perithecia$mcmc)
Spore Count

MCMC_noff <- glmmadmb(formula = total_char_ascospores ~ geo +
    #geo_sq +
    #gen_dis*geo +
    gen_dis +
    gen_dis_sq +
    (1 | ma) +
    (1 | pa) +
    offset(logtime),
    data = spore_count_data,
    zeroInflation = FALSE,
    family = "nbinom",
    mcmc = TRUE,
    mcmc.opts=mcmcControl(mcmc=10000))

noff_mcmc_coda <- as.mcmc(MCMC_noff$mcmc)

Proportion Pigmented

MCMC_perc_pig <- glmmadmb(formula = pigmented_count ~
    geo_sq +
    gen_dis*geo +
    gen_dis_sq +
    (1 | ma) +
    (1 | pa) +
    offset(logTotalAscospores),
    data = pigmented_count_data,
    zeroInflation = FALSE,
    family = "nbinom",
    mcmc = TRUE,
    mcmc.opts=mcmcControl(mcmc=10000))

perc_pig_mcmc_coda <- as.mcmc(MCMC_perc_pig$mcmc)

Pigmented Size

MCMC_Pigmented_size <- glmmadmb(formula = PSSCW_std ~
    #geo*gen_dis +
    geo +
    geo_sq +
    #gen_dis_sq +
    (1 | ma) +
    (1 | pa),
    data = PSSCWdf,
    zeroInflation=FALSE,
    family="gaussian",
    mcmc = TRUE,
    mcmc.opts=mcmcControl(mcmc=10000))

pigmented_size_mcmc_coda <- as.mcmc(MCMC_Pigmented_size$mcmc)
Pigmented Spore Germination

MCMC_germ_pigmented <- glmmadmb(formula= germ ~
    #geo +
    #geo_sq +
    #gen_dis*geo +
    #gen_dis +
    gen_dis_sq +
    (1 | ma) +
    (1 | pa) +
    offset(Log_pigmented),
    data = germ_data,
    zeroInflation = FALSE,
    family = "nbinom",
    mcmc = TRUE,
    mcmc.opts=mcmcControl(mcmc=10000))

germ_mcmc_coda <- as.mcmc(MCMC_germ_pigmented$mcmc)

Plot parameters and HPD intervals

hpd_perithecia <- HPDinterval(perithecia_mcmc_coda)
hpd_spores <- HPDinterval(noff_mcmc_coda)
hpd_perc_pig <- HPDinterval(perc_pig_mcmc_coda)
hpd_pigm_size <- HPDinterval(pigmented_size_mcmc_coda)
hpd_germination <- HPDinterval(germ_mcmc_coda)

model_output <- data.frame(Parameter = rep(c("(Intercept)",
    "gen_dis",
    "geo",
    "gen_dis_sq",
    "geo_sq",
    "gen_dis:geo"), 5),
    Model = rep(c("PeritheciaCount",
    "TotalAscospores",
    "ProportionPigmented",
    "PigmentedSize",
    "PigmentedGermination"), each = 6))

model_output$Parameter <- factor(model_output$Parameter,
    levels = c("(Intercept)",
    "gen_dis",
    "geo",
    "gen_dis_sq",
    "geo_sq",
    "gen_dis:geo"))

model_output$Model <- ordered(model_output$Model,
    levels = c("PeritheciaCount",
    "TotalAscospores",
    "ProportionPigmented",
    "PigmentedSize",
    "PigmentedGermination")
)
parameter_getter <- function(model_object, modeled_trait, hpd) {
  model_parameters <- as.data.frame(model_object$b)
  model_parameters <- cbind(row.names(model_parameters), model_parameters, Model = modeled_trait)
  row.names(model_parameters) <- c("Parameter", "Value", "Model")
  row.names(model_parameters) <- NULL
  hpd <- as.data.frame(hpd)
  hpd_rn <- cbind(row.names(hpd), hpd)
  model_parameters <- merge(model_parameters, hpd_rn, by.x="Parameter", by.y ="row.names(hpd)"
  return(model_parameters)
}

perithecia_parameters <- parameter_getter(MP1_perithecia, "PeritheciaCount", hpd_perithecia)
spore_count_parameters <- parameter_getter(MP1_noff, "TotalAscospores", hpd_spores)
perc_pig_parameters <- parameter_getter(MP1_perc_pig, "ProportionPigmented", hpd_perc_pig)
pigm_size_parameters <- parameter_getter(MP1_Pigmented_size, "PigmentedSize", hpd_pigm_size)
germ_parameters <- parameter_getter(M2_germ_pigmented, "PigmentedGermination", hpd_germination)
all_parameters <- rbind(perithecia_parameters,
                        spore_count_parameters,
                        perc_pig_parameters,
                        pigm_size_parameters,
                        germ_parameters)

model_results <- merge(all_parameters, model_output, by.x=c("Parameter", "Model"),
                         by.y=c("Parameter", "Model"), all.y=TRUE)
model_results$lower <- as.numeric(model_results$lower)
model_results$upper <- as.numeric(model_results$upper)

model_results$Parameter <- factor(model_results$Parameter,
                                   levels = c("(Intercept)",
                                              "gen_dis",
                                              "gen disap",
                                              "geo",
                                              "geo sq",
                                              "gen dis:geo")
})
model_results$Model <- ordered(model_results$Model,
                                levels = rev(c("PeritheciaCount",
                                               "TotalAscospores",
                                               "ProportionPigmented",
                                               "PigmentedSize",
                                               "PigmentedGermination"))
})
limits <- aes(xmax = upper, xmin= lower)

ggplot(model_results, aes(x = Value, y = Model)) +
  geom_point() +
  geom_errorbarh(limits, height = .2, colour = "black") +
  geom_vline(xintercept = 0, colour = "black", linetype = "dotted") +
  facet_wrap(~Parameter, scales = "free_x", ncol = 6) +
  theme_bw(10) +
  scale_y_discrete(labels = c("PeritheciaCount" = "Perithecia\nCount",
                         "TotalAscospores" = "Spore\nCount",
                         "ProportionPigmented" = "ProportionPigmented",
                         "PigmentedSize" = "PigmentedSize",
                         "PigmentedGermination" = "PigmentedGermination")

44
scale_x_continuous("Parameter Value") +
theme(
    panel.margin = unit(.5, "lines"),
    strip.background = element_blank(),
    #panel.border = element_blank(),
    axis.line = element_line(color="black"),
    axis.line.y = element_blank(),
    axis.text.x = element_text(angle = 45, hjust=1, vjust=1))

## Warning: Removed 2 rows containing missing values (geom_point).
## Warning: Removed 1 rows containing missing values (geom_point).
## Warning: Removed 2 rows containing missing values (geom_point).
## Warning: Removed 3 rows containing missing values (geom_point).
## Warning: Removed 4 rows containing missing values (geom_point).