

glms for proportion abscised

E.E. Jackson

13 May, 2020

```
rm(list = ls())
require(tidyverse)
require(lme4)
library(knitr)
library(report)
require(MuMIn)
require(ggpubr)
require(MASS)
require(broom)
require(ggpubr)
opts_chunk$set(tidy.opts=list(width.cutoff=50),tidy=TRUE)
```

Read dataset

```
fruitDat <- read.csv("../output/tables/cleanData.csv")
dm <- read.csv("../output/tables/dispersalMode.csv")

fruitDat <- fruitDat %>% mutate(seedpred_pres = factor(ifelse(seedpred_pres ==
  0, "No", "Yes"))) %>% left_join(dplyr::select(dm,
  dispersal_mode = bio, sp), by = "sp")

## Warning: Column `sp` joining factors with different levels, coercing to
## character vector

fruitDat$seed_dry_log <- log10(fruitDat$seed_dry)
fruitDat$cofruit_log <- log10(fruitDat$cofruit)
fruitDat$bcireproductive_log <- log10(fruitDat$bcireproductive)
```

function to test for overdispersion

```
overdisp_fun <- function(model) {
  rdf <- df.residual(model)
  rp <- residuals(model, type = "pearson")
  Pearson.chisq <- sum(rp^2)
  prat <- Pearson.chisq/rdf
  pval <- pchisq(Pearson.chisq, df = rdf, lower.tail = FALSE)
  c(chisq = Pearson.chisq, ratio = prat, rdf = rdf,
    p = pval)
}
```

mixed effect models

```
# write a loop to run all the univariate models

vars = names(fruitDat)[c(13, 14, 17, 21, 25:28)]

models <- lapply(setNames(vars, vars), function(var) {
  formula = paste("cbind(abscised_seeds,viable_seeds)~",
    var, "+ (1|sp)")
  glmer(formula, family = binomial(logit), data = fruitDat)
})

sapply(models, overdisp_fun) # check for overdispersion, want to see the ratio is not over 1

##          height_avg          cvseed endocarp_investment seedpred_pres
## chisq    7.89457289   10.06526306          9.12975006   10.10821347
## ratio    0.04485553    0.05057921          0.04286268    0.04636795
## rdf     176.00000000  199.00000000          213.00000000  218.00000000
## p         1.00000000    1.00000000          1.00000000    1.00000000
##          dispersal_mode seed_dry_log  cofruit_log bcireproductive_log
## chisq     17.83493004   17.55701694    9.95874222          6.88913193
## ratio     0.05698061    0.05775335    0.04589282          0.03959271
## rdf      313.00000000  304.00000000  217.00000000          174.00000000
## p         1.00000000    1.00000000    1.00000000          1.00000000

sapply(models, r.squaredGLMM) # calculate R squared, row 1 is marginal (fixed effects alone), 3 is con

##          height_avg          cvseed endocarp_investment seedpred_pres dispersal_mode
## [1,] 0.04021311 0.02855874          0.04630999    0.04701081   1.415040e-05
## [2,] 0.04021292 0.02855856          0.04630955    0.04701053   1.415032e-05
## [3,] 0.99998414 0.99997804          0.99996691    0.99997916   9.999801e-01
## [4,] 0.99997959 0.99997174          0.99995741    0.99997318   9.999743e-01
##          seed_dry_log cofruit_log bcireproductive_log
## [1,] 0.07902491 0.03635080          0.04031373
## [2,] 0.07902426 0.03635058          0.04031351
## [3,] 0.99997141 0.99997949          0.99998043
## [4,] 0.99996320 0.99997360          0.99997481

# compute per-model statistics
purrr::map_dfr(models, broom::glance, conf.int = TRUE,
  .id = "vars")

## # A tibble: 8 x 7
##   vars          sigma logLik   AIC   BIC deviance df.residual
##   <chr>          <dbl>  <dbl> <dbl> <dbl>   <dbl>      <int>
## 1 height_avg        1 -1163. 2332. 2341.    15.0        176
## 2 cvseed            1 -1349. 2703. 2713.    19.3        199
## 3 endocarp_investment 1 -1407. 2819. 2829.    16.8        213
## 4 seedpred_pres      1 -1458. 2921. 2931.    19.2        218
## 5 dispersal_mode     1 -1925. 3855. 3867.    34.3        313
## 6 seed_dry_log       1 -1848. 3702. 3714.    33.4        304
## 7 cofruit_log        1 -1449. 2905. 2915.    18.9        217
## 8 bcireproductive_log 1 -1148. 2302. 2311.    12.6        174

# compute statistics about each of the coefficients
# for each model
```

```
res_anova <- purrr::map_dfr(models, broom::tidy, conf.int = TRUE,
  .id = "vars")
res_anova
```

```
## # A tibble: 24 x 9
##   vars      term estimate std.error statistic p.value conf.low conf.high group
##   <chr>   <chr>     <dbl>    <dbl>    <dbl>   <dbl>   <dbl>   <dbl> <chr>
## 1 height~ (Inter~ -1.46     0.457    -3.20   1.35e-3  -2.36    -0.569  fixed
## 2 height~ height~  0.0398    0.0187     2.13   3.29e-2  0.00322  0.0764  fixed
## 3 height~ sd_(I~  1.82      NA        NA      NA      NA      NA      sp
## 4 cvseed~ (Inter~ -0.0452    0.351    -0.129  8.98e-1  -0.732    0.642  fixed
## 5 cvseed~ cvseed~ -0.483     0.239    -2.02   4.30e-2  -0.952   -0.0151 fixed
## 6 cvseed~ sd_(I~  1.72      NA        NA      NA      NA      NA      sp
## 7 endoca~ (Inter~ -0.0257    0.256    -0.100  9.20e-1  -0.527    0.475  fixed
## 8 endoca~ endoc~ -1.43      0.567    -2.51   1.19e-2  -2.54   -0.314  fixed
## 9 endoca~ sd_(I~  1.60      NA        NA      NA      NA      NA      sp
## 10 seedpr~ (Inter~ -1.23      0.278    -4.42   9.70e-6  -1.77   -0.685  fixed
## # ... with 14 more rows
```

```
# we are interested in rows where term = the
# variable
results <- res_anova[!grepl("(Intercept)", res_anova$term),
  ]
results
```

```
## # A tibble: 8 x 9
##   vars      term estimate std.error statistic p.value conf.low conf.high group
##   <chr>   <chr>     <dbl>    <dbl>    <dbl>   <dbl>   <dbl>   <dbl> <chr>
## 1 height~ height~  0.0398    0.0187     2.13   3.29e-2  0.00322  0.0764  fixed
## 2 cvseed~ cvseed~ -0.483     0.239    -2.02   4.30e-2  -0.952   -0.0151 fixed
## 3 endocar~ endoca~ -1.43      0.567    -2.51   1.19e-2  -2.54   -0.314  fixed
## 4 seedpre~ seedpr~  0.857     0.329     2.60   9.31e-3  0.211    1.50    fixed
## 5 dispers~ disper~  0.0208    0.0652     0.319  7.50e-1  -0.107    0.149  fixed
## 6 seed_dr~ seed_d~  0.628     0.156     4.02   5.72e-5  0.322    0.934  fixed
## 7 cofruit~ cofrui~  2.98      1.34      2.22   2.63e-2  0.350    5.60    fixed
## 8 bcirepr~ bcirep~  0.575     0.272     2.11   3.46e-2  0.0417    1.11    fixed
```

plot models

```
# need to remove these two for plotting
models$seedpred_pres <- NULL
models$dispersal_mode <- NULL

plot_data_var <- function(model, modelname) {
  output = as.data.frame(effects::effect(term = modelname,
    mod = model))
  ggplot() + geom_point(data = fruitDat, aes_string(x = modelname,
    y = "proportion_abscised"), size = 0.5) + geom_line(data = output,
    aes_string(x = modelname, y = "fit"), colour = "blue") +
    geom_ribbon(data = output, aes_string(x = modelname,
      ymin = "lower", ymax = "upper"), alpha = 0.1,
      fill = "blue") + xlab(modelname) + theme_bw(base_size = 10)
}
```

```
myplots <- imap(models, plot_data_var)
```

```
ggpubr::ggarrange(plotlist = myplots)
```

