**Supplemental Appendix 1.**

# Author: John Nason

# November 15, 2021

###############################

### CODE OBJECTIVES ###

# This R code simulates the null distribution of *Ficophagus* and *Parasitodiplogaster*

# mating pools under the binomial distribution with replacement, in accordance

# with a user-defined sex ratio of 0.5, as expected with chromosomal sex

# determination, or set to match the sex ratio in the observed data. The results

# include a one-one-tailed test of the hypothesis that the observed variance in

# sex ratio is less than that expected under the binomial. The p-value of the test

$ is returned in variable p\_val.

###########

# User settings for simulations:

# Specify the mating pool size, m, sample number of mating pools, N, sex ratio

# (probability of males), p, and observed sample variance in sex ratio, v\_obs.

# Values of m and N will be representative of the actual data, while p may reflect

# an external null hypothesis such as a 1:1 sex ratio (p = 0.5), or the sex ratio in

# the actual data.

m <- 2 # Mating pool size.

N <- 7 # Sample number of mating pools.

p <- 0.5 # User-defined sex ratio: 0.5 or the observed sex ratio.

v\_obs <- 0.036 # Observed variance in sex ratio (provide to five decimal places).

###########

# The expected variance is binomial(m, p) and equals p\*(1-p)/m:

v\_exp <- p\*(1-p)/m

v\_exp

###########

# Simulating the distribution of variance given N mating pools

# of size m and male probability p.

# Simulate N mating pools (given m and p) nrep times under the

# binomial.

nrep <- 10000

Pmale <- matrix(rbinom(N\*nrep, m, p), nrow=N)/m

#Pmale is a matrix with N rows and nrep columns.

# Summarize the distribution of Pmale, the probability of

# males, across all N\*nrep simulated mating pools. For large

# N\*nrep, should be very close to binomial(m,p).

table(Pmale)

# Calculate the simulated sample variance in Pmale for each of

# the nrep simulated mating pools of sample size N.

v\_sim <- apply(Pmale, 2, var)

mean(v\_sim)

# Plot the distribution v\_sim and it's mean value.

mx <- mean(v\_sim) #Approximates p\*(1-p)/m and so is independent of N given binomial(m,p).

hist(v\_sim, xlab = paste("Variance in Pmale"), main = paste("Distribution of simulated sample variance in Pmale"))

abline(v = mx, col = "blue", lwd = 2)

text(x=mx, y=nrep/40 , cex=1.25, pos=4, col="blue", paste("Mean = ", round(mx, 5)))

# The simulated variance v\_sim is a vector of length nrep. The critical

# value, crit\_val, of the one-tailed test of v\_obs is less than or

# equal to the alpha\*nrep lowest value of v\_sim, where alpha=0.05 is

# typical. If v\_obs ≤ crit\_val, then we conclude the observed variance

# is significantly less than expected under the binomial.

alpha <- 0.05

crit\_value <- sort(v\_sim, decreasing=TRUE)[length(v\_sim) - alpha\*nrep]

crit\_value

# The p-value (p\_val) for the one-tailed test is probability of v\_obs under

# the null distribution v\_sim. This is determined by the proportion of v\_sim

# values less than or equal to v\_obs.

p\_val <- sum(round(v\_sim, 5) <= round(v\_obs, 5))/nrep

p\_val

# Plot observed variance p\_val relative to the null distribution v\_sim.

hist(v\_sim, xlab = paste("Variance in Pmale"), main = paste("One-tailed test of the sample variance in Pmale (p = ",round(p\_val, 5), ")"))

if (v\_obs != crit\_value){

abline(v=crit\_value, col = "blue", lwd = 2)

text(x=round(crit\_value, 5), y=nrep/40 , col = "blue", cex=1.25, pos=4, offset=1, paste("Critical value = ", round(crit\_value, 5)))

abline(v = v\_obs, col = "red", lwd = 2)

text(x=round(v\_obs, 5), y=nrep/80 , col = "red", cex=1.25, pos=4, offset=1, paste("Observed variance = ", round(v\_obs, 5)))

} else {

abline(v=crit\_value, col = "blue", lwd = 2)

text(x=round(crit\_value, 5), y=nrep/40 , col = "blue", cex=1.25, pos=4, offset=1, paste("Observed variance = critical value = ", round(crit\_value, 5)))

}

**Supplemental Appendix 2.**

# Male-Female\_Combo\_Test.R

# Author: John Nason

# November 15, 2021

###############################

### CODE OBJECTIVES ###

# This R code simulates the frequency of the nematode near-mean sex ratio

# male-female combination expected under genetic sex chromosomes in order

# to test the hypothesis that the observed near-mean combination frequency

# is significantly greater than the simulated null distribution of this

# frequency at a user-specified alpha level (one-tailed test). This

# simulation draws from the Binomial distribution with replacement, with

# a user-defined sex ratio set to 0.5 or to match that in the observed

# input data.

#

# Data is read from a user-specified input Excel file in a particular format,

# as described below. Results are written to a new output Excel file, with

# frequencies and p-values modeled under chromosomal sex determination

# appended as columns to data from the input data file.

# Input Excel file format (separate files by nematode genus and single- vs

# multi-foundress fruits):

# Column headers and data:

# A. Nematode mating pool size (m)

# B. Sample size (n)

# C. Overall mean sex ratio for the single- or multi-foundress data set

# D. Sample sex ratio (pm)

# E. Near-mean sex ratio male-female combination

# F. Number of males in near-mean sex ratio combination

# G. Near-mean sex ratio combination frequency

# H. Near-mean sex ratio combination count

# I. Expected near-mean sex ratio frequency given pm=0.5 (header only, no data)

# J. p-value given pm=0.5 (header only, no data)

# K. Critical value given pm=0.5 (header only, no data)

# L. Expected near-mean sex ratio frequency given the genus mean sex ratio (header only, no data)

# M. p-value given genus mean sex ratio (header only, no data)

# N. Critical value given genus mean sex ratio (header only, no data)

# Rows:

# Rows organized by mating pool size m = 1 to 10.

# The output file will contain the above, with results added to columns I through N.

# Set USER-DEFINED PARAMETERS below.

# Runtime is dependent on sample sizes and computer. The single-foundress

# Parasitidiplogaster data set takes 27 seconds to run on a standard 2017

# Apple MacBook.

###############################

### PRELIMINARIES ###

# Optional: Clear all variables from the current R working environment.

rm(list=ls(all=TRUE))

# Install package to read input Excel data file into a dataframe.

#install.packages("readxl")

library(readxl)

# Install package to write final output dataframe to Excel data file.

#install.packages("writexl")

library(writexl)

###############################

### USER-DEFINED PARAMETERS ###

# Set path to directory where the input data file is located.

# For Parasitodiplogaster:

#cwd <- "~/Dropbox/Manuscripts/Van Goor, Justin/Sex Ratio Paper/Figures 2020-04-19/Figure 2/Data/Parasitodiplogaster/"

cwd <- "~/Dropbox/Manuscripts/Van Goor, Justin/Sex Ratio Paper/Figures/Figures 2021-11/Figure 4, Dom Combo/Test of Dominant male-female combo/2021-11-15/"

# For Ficophagus:

#cwd <- "~/Dropbox/Manuscripts/Van Goor, Justin/Sex Ratio Paper/Figures 2020-04-19/Figure 2/Data/Ficophagus/"

# Set input Excel file name.

#input.file.name <- "Parasitodiplogaster\_Dom\_Freq\_Data.xlsx"

input.file.name <- "Ficophagus\_Dom\_Freq\_Data.xlsx"

# Set output file name.

#output.file.name <- "Parasitodiplogaster\_Dom\_Freq\_Test\_Out\_X.xlsx"

output.file.name <- "Ficophagus\_Dom\_Freq\_Test\_Out\_XX.xlsx"

# Number of simulated replicates.

n.reps <- 10000

# Critical value of one-tailed test of the hypothesis that the observed

# frequency of the near-mean sex ratio male-female combination is higher

# than expected under GSD.

alpha <- 0.05

###############################

### CODE ###

# Start clock

ptm <- proc.time()

# Set output file path

output.file.path <- paste(cwd, output.file.name, sep="")

output.file.path

# Import input data file into a data frame.

# Get the path to the input Excel data file.

input.file.path <- paste(cwd, input.file.name, sep="")

input.file.path

# Read input data from Excel file.

df <- read\_excel(input.file.path, skip = 1, col\_names = TRUE)

#sapply(df, mode)

View(df)

# Convert df to matrix.

df.mat <- as.matrix(df)

# Get mating pool sizes from df.mat and store in a vector.

mating.pool.size.vec <- as.integer(df.mat[,1])

# Get mean observed sex ratio (is constant across rows of df.mat column 3).

p.bar <- as.numeric(df.mat[1,3])

# Create a vector of sex ratios 0.5 and the mean observed ratio, p.bar.

p.gsd <- c(0.5, p.bar)

###########

for(g in p.gsd){

#browser()

# Create a vector to store simulated values of the expected frequency of

# the near-mean sex ratio male-female combination.

f.dom.exp.vec <- vector()

crit.value.high.vec <- vector()

p.val.vec <- vector()

for(i in mating.pool.size.vec){

#browser()

# Read in required input data from dataframe df

# Mating pool size.

m <- as.integer(df.mat[i,1])

# Sample number of mating pools.

N <- as.integer(df.mat[i,2])

# Mean sex ratio over all sample mating pools.

p.bar <- as.numeric(df.mat[i,3])

# Sex ratio at current mating pool size.

p <- as.numeric(df.mat[i,4])

# Number of males in near-mean sex ratio male-female combination.

n.males <- as.integer(df.mat[i,6])

# Observed frequency of near-mean sex ratio combination.

f.dom <- as.numeric(df.mat[i,7])

# Number of times the near-mean sex ratio combination was observed

# across N sample mating pools.

n.dom <- as.numeric(df.mat[i,8])

#print(n.dom)

###########

# Expected frequency of the near-mean sex ratio male-female combination

# under GSD assuming a binomial with mating pool size m, n\_males\_dom,

# and sex ratio p.gsd. For each m, store exp.freq in a vector.

exp.freq <- dim(combn(m,n.males))[2]\*(g^n.males)\*((1-g)^(m-n.males))

f.dom.exp.vec <- c(f.dom.exp.vec, exp.freq)

# Number of times the near-mean sex ratio male-female combination is

# expected across N sample mating pools under GSD.

# N\*dim(combn(m,n.males))[2]\*(p.gsd^n.males)\*((1-p.gsd)^(m-n.males))

###########

# Simulate counts of the near-mean sex ratio male-female combination

# expected under GSD. Test the observed count of the near-mean sex ratio

# male-female combination relative to the simulated null distribution.

# Simulate N mating pools (given m and p) n.reps times under the

# binomial assuming GSD.

Pmale.mat <- matrix(rbinom(N\*n.reps, m, p.gsd), nrow=N)/m

#Pmale.mat is a matrix with N rows and n.reps columns.

target\_freq <- toString(n.males/m)

sim.dom.combo.count <- vector()

x\_sum <- 0

for(j in 1:ncol(Pmale.mat)){

x <- table(Pmale.mat[,j]) # Table of counts of male frequencies by nrep

x\_sum <- x\_sum + 1

y <- x[names(x) == target\_freq] # Counts in the table at target\_freq

if (length(y)==0){

y <- 0

}

sim.dom.combo.count <- c(sim.dom.combo.count, as.integer(y))

}

sim.dom.combo.freq <- sim.dom.combo.count/N

# The simulated male-female combos sim.dom.combo.count is a vector of length

# n.reps. The critical values of the two-tailed test of n.dom are the alpha\*n.reps

# lowest and (1-alpha)\*n.reps highest values sim.dom.combo.count,

# where alpha=0.025 is typical. If obs\_dom\_combo\_count lies beyond these two

# critical values then we conclude obs\_dom\_combo\_count is significantly

# different than expected under the GSD binomial model.

#crit.value.low <- sort(sim.dom.combo.freq)[length(sim.dom.combo.freq) - (1-alpha)\*n.reps]

crit.value.high <- sort(sim.dom.combo.freq)[length(sim.dom.combo.freq) - alpha\*n.reps]

crit.value.high.vec <- c(crit.value.high.vec, crit.value.high)

# The p-value (p\_val) for the two-tailed test is the probability of

# obs\_dom\_combo\_count under the null distribution sim\_dom\_combo\_count.

# This is determined by the lower of the proportion of sim\_dom\_combo\_count

# values less than or equal to crit\_value\_low or greater than or equal

# to crit\_value\_high.

p.val.high <- sum(sim.dom.combo.freq >= f.dom)/n.reps

p.val.vec <- c(p.val.vec, p.val.high)

}

if (g == 0.5){

df[,9] <- matrix(f.dom.exp.vec)

df[,10] <- matrix(p.val.vec)

df[,11] <- matrix(crit.value.high.vec)

} else{

df[,12] <- matrix(f.dom.exp.vec)

df[,13] <- matrix(p.val.vec)

df[,14] <- matrix(crit.value.high.vec)

}

}

# Write results to new Excel file (this takes 15-20 seconds).

write\_xlsx(df, output.file.path, col\_names = TRUE, format\_headers = TRUE)

# Elapsed time.

proc.time() - ptm

###############################