Supplementary online material

Article: Defaunation effects on plant recruitment depend on size matching and size trade-offs in seed-dispersal networks

Authors: Isabel Donoso, Matthias Schleuning, Daniel García, Jochen Fründ.

Journal: Proceedings of the Royal Society B

Appendix S3. Source Code for simulations in the R language for statistical computing

Explanatory comments (#)

```
## 1) R version 3.3.1 (2016-06-21)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 7 x64 (build 7601) Service Pack 1
##
## 2) Load Packages
## require(vegan)# to calculate Shannon diversity index
## require(Hmisc)# to calculate the community weighted mean of seed size

# -----------------------------------------------------------------------------------------------------------
# --------------- Functions to create our simulated seed dispersal networks ------------
# ------------------------ Modified from Fründ et. al. (2016) ----------------------------------
# -----------------------------------------------------------------------------------------------------------

#---- web generator (quantitative (weighted) niche model) ----
makeweb <- function(specpar = 1, birdtraits, planttraits, nicheshape="normal"){
  # function to generate web with defined "specialization", based on a trait matching concept
  # the output web has interaction probabilities for all species (assuming equal abundances)
  # specpar is the specialization parameter (always increases with specialization)
  # nicheshape: the function calculating pref.values from trait-differences; the default 'normal' uses gaussian/bell-shaped niches
  fun_pref <- function(traitdif){
    if (nicheshape == "normal") {
      prefs <- dnorm(traitdif,mean=0,sd=1/specpar)
    }
    if (nicheshape == "skewed"){
   # a simple skewed function, lognormal shifted to mode=0 (scaling by stretching x)
      prefs <- dlnorm(traitdif * specpar + exp(-1))
    }
    prefs
  }
  Nplant <- length(planttraits)
  Nbird <- length(birdtraits)
  web <- fun_pref(outer(planttraits,birdtraits,"-")*(-1))  # adjusted this so that traitdif is defined as birdtrait-planttrait
  web <- web / matrix(colSums(web),nrow=Nplant,ncol=Nbird,byrow=TRUE)  # standardize link weights to probability;
  web
}

#---- create skewed traits ---- (same as get_skewabuns function in Fründ et. al. (2016))
get_skewtr <- function(myN, tr_meanlog=2, tr_sdlog=1.5){
  # generate traits that match a log-normal distribution (but without introducing noise):
  # divide quantile distribution in N+1 regular intervals, and take the N non-0or1 intvl borders as trait values
  # it is rescaled in the second step (line70) to really have the intended mean trait (not log-mean)
  tr <- qlnorm(seq(0, 1, length.out=myN+2), tr_meanlog, tr_sdlog)[-c(1,myN+2)]  # takes equidistant points of the quantile function, removing the extremes that would be 0 and Inf
  tr <- sort(tr, decr=TRUE)
}

#---- make true web from preferences and interaction frequencies ----
make_trueweb <- function(web_p, plantfreq, birdfreq){
  # first step: prepare a web that's used to multiply with the web_p;
  web_relfreq <- (plantfreq %*% t(birdfreq)) / mean(plantfreq)
```

```
62    # second step: multiply preference web with interaction frequencies
63      web_p * web_relfreq
64  }
65
66
67  #---------------------------------------------------------------------------------------------------------------------
68  #-------------------------- STEP 1: GENERATE SEED DISPERSAL NETWORKS -------------------
69  #---------------------------------------------------------------------------------------------------------------------
70
71  #----(I) Generate the simulated sps pool ----
72  Nbird <- 60
73  Nplant <- 50
74
75  #----(II) Draw trait (SIZE) values using lognormal distributions but with parameters matching the empirical distributions of
76  fruit volume and bird body mass presented in Dehling et. al. (2014) ----
77  ## For the sake of simplicity, we will make available just the final fitted traits (lines 100-115). However, from lines 80-95, we show
78  how to reproduce it with other empirical trait data.
79
80  # Get the mean and sdlog for any empirical traits (parameters to change afterwards within the functions). "volume" and "Bodymass"
81  would be the empirical variables
82
83  # FRUIT SIZE
84  # vol_meanlog <- mean(log(volume^(1/3)))
85  # vol_sdlog <- sd(log(volume^(1/3)))
86  # BIRD BODY MASS
87  # mass_meanlog <- mean(log(Bodymass^(1/3))) # taking the cubic root of mass translates it into a linear (instead of volume)
88  measure
89  # mass_sdlog <- sd(log(Bodymass^(1/3)))
90
91  # Change the values of the parameters mean and sdLog within the function by those empirical ones
92  # PLANTS
93  # fit_pltr <- get_skewtr(Nplant, vol_meanlog, vol_sdlog) ## vector x in the main text
94  # BIRDS
95  # fit_birdtr <- get_skewtr(Nbird, mass_meanlog, mass_sdlog) ## vector y in the main text
96
97  # These are the trait values we obtained (matching the empirical distributions of fruit volume and bird body mass presented in
98  Dehling et al. 2014)
99  # PLANTS
100 fit_pltr<-c(23.716429, 19.578933, 17.298204, 15.737042, 14.556170, 13.609303, 12.820320,
101 12.144684, 11.554152, 11.029717, 10.557994, 10.129207, 9.736007, 9.372729, 9.034917,
102 8.719000, 8.422074, 8.141743, 7.876006, 7.623172, 7.381797, 7.150638, 6.928613,
103 6.714771, 6.508270, 6.308358, 6.114356, 5.925645, 5.741655, 5.561857, 5.385750,
104 5.212858, 5.042716, 4.874868, 4.708854, 4.544203, 4.380421, 4.216975, 4.053279,
105 3.888665, 3.722353, 3.553398, 3.380615, 3.202455, 3.016797, 2.820557, 2.608908,
106 2.373454, 2.096973, 1.731142)
107 # BIRDS
108 fit_birdtr<-c(7.037466, 6.334532, 5.920422, 5.623856, 5.391486, 5.199626, 5.035663,
109 4.892077, 4.764019, 4.648179, 4.542195, 4.444321, 4.353231, 4.267890, 4.187480,
110 4.111336, 4.038914, 3.969761, 3.903497, 3.839797, 3.778383, 3.719013, 3.661475,
111 3.605583, 3.551170, 3.498088, 3.446201, 3.395388, 3.345534, 3.296537, 3.248297,
112 3.200724, 3.153729, 3.107227, 3.061138, 3.015381, 2.969875, 2.924540, 2.879294,
113 2.834051, 2.788723, 2.743215, 2.697425, 2.651240, 2.604538, 2.557178, 2.508999,
114 2.459812, 2.409396, 2.357479, 2.303726, 2.247709, 2.188872, 2.126459, 2.059404,
115 1.986119, 1.904055, 1.808677, 1.690437, 1.521589)
116
117 #---- (III) Once having the trait values… Estimation of the interaction frequency for birds and plants if there is a negative
118 relationship ("YES" scenarios; following g=(1/y)+b for birds, and following f=1/x in the case of plants) or if there is no
119 relationship between size-int freq ("NO" scenarios). All the outputs were scale dividing by the mean ----
120
121 ### "YES"
122 # PLANTS. We assume the relationship: y=1/x
123 YES_pl_freq <- (1/fit_pltr)/mean((1/fit_pltr)) ## vector fi in the main text
124 # BIRDS. We assume a negative relationship: y=(1/x)+b (where b is the undercompensation parameter set to the 10% of the
125 maximum value of 1/x
126 v10 <- max(1/fit_birdtr)/10
127 YES_bird_freq <- ((1/fit_birdtr)+v10)/mean((1/fit_birdtr)+v10) ## vector gj in the main
128 text
129
130 ### "NO". Fixed to the value representing the mean freq of the YES scenarios but as we scale dividing by the mean, the mean freq
131 value is 1
132 # PLANTS
133 mfreqY_pl <- 1
134 NO_pl_freq <- rep(mfreqY_pl,50)
135 # BIRDS
136 mfreqY_bird <- 1
137 NO_bird_freq <- rep(mfreqY_bird,60)
```

```
#------ (IV) Generate the final simulated seed dispersal networks ----

# TWO INTERACTION RULES
# 1. NO SIZE MATCHING (neutral case)
web_neutral <- matrix(1,Nplant,Nbird)
# 2. SIZE MATCHING
web_p <- makeweb(specpar=10, birdtraits=as.vector(decostand(fit_birdtr,"range")),
planttraits=as.vector(decostand(fit_pltr,"range")),nicheshape="skewed")

#-- Calculate final seed dispersal networks incorporating interaction frequencies according to different scenarios: no size matching
vs. size matching, crossed with "YES" and "NO" scenarios for the size-interaction frequency relationship in plants only, birds only,
none or both. We further fixed the exact bird frequencies and let plant frequencies vary in each scenario.

# NO SIZE MATCHING SCENARIOS
# NONE
sc1 <- make_trueweb(web_p=web_neutral, birdfreq=NO_bird_freq, plantfreq=NO_pl_freq)
sc1b <- sc1/ matrix(colSums(sc1)/NO_bird_freq,nrow=Nplant, ncol=Nbird,byrow=TRUE)
# in BIRDS ONLY
sc2 <- make_trueweb(web_p=web_neutral, birdfreq=YES_bird_freq, plantfreq=NO_pl_freq)
sc2b <- sc2/ matrix(colSums(sc2)/YES_bird_freq, nrow=Nplant, ncol=Nbird,byrow=TRUE)
# in PLANTS ONLY
sc3 <- make_trueweb(web_p=web_neutral, birdfreq=NO_bird_freq, plantfreq=YES_pl_freq)
sc3b <-sc3/ matrix(colSums(sc3)/NO_bird_freq,nrow=Nplant, ncol=Nbird,byrow=TRUE)
# in BOTH
sc4 <- make_trueweb(web_p=web_neutral, birdfreq=YES_bird_freq, plantfreq=YES_pl_freq)
sc4b <- sc4/matrix(colSums(sc4)/YES_bird_freq,nrow=Nplant, ncol=Nbird,byrow=TRUE)

# SIZE MATCHING SCENARIOS
# NONE
sc5 <- make_trueweb(web_p, birdfreq=NO_bird_freq, plantfreq=NO_pl_freq)
sc5b <- sc5/matrix(colSums(sc5)/NO_bird_freq,nrow=Nplant, ncol=Nbird,byrow=TRUE)
# in BIRDS ONLY
sc6 <- make_trueweb(web_p, birdfreq=YES_bird_freq, plantfreq=NO_pl_freq)
sc6b <- sc6/matrix(colSums(sc6)/YES_bird_freq,nrow=Nplant, ncol=Nbird,byrow=TRUE)
# in PLANTS ONLY
sc7 <- make_trueweb(web_p, birdfreq=NO_bird_freq, plantfreq=YES_pl_freq)
sc7b <- sc7/matrix(colSums(sc7)/NO_bird_freq,nrow=Nplant, ncol=Nbird,byrow=TRUE)
# in BOTH
sc8 <- make_trueweb(web_p, birdfreq=YES_bird_freq, plantfreq=YES_pl_freq)
sc8b <- sc8/matrix(colSums(sc8)/YES_bird_freq,nrow=Nplant, ncol=Nbird,byrow=TRUE)


#-----------------------------------------------------------------------------------------------------------------------------------
#------------------ STEP 2: GENERATE 3 DIMENSSIONS OF SEEDLING RECRUITMENT -------------
#-----------------------------------------------------------------------------------------------------------------------------------

#---- (V) From seed dispersal networks to seedling recruitment networks ----

# Estimation of seed fate for birds and plants if there is a positive relationship which exactly cancels out the negative interaction
frequency-size relationships from step 1 ("YES" scenarios; following y = ax/(1+bx)for birds, and following y=x in the case of
plants) or if there is no relationship between size-seed fate ("NO" scenarios). These outputs were also scaled by the respective
mean.

### "YES"
# PLANTS: we assume y=x
YES_fate_pl <- fit_pltr  ## vector qi in the main text
# BIRDS: we assume y = ax/(1+bx)
YES_fate_bird <- fit_birdtr/(1+v10*fit_birdtr) ## vector rj in the main text

### "NO"
# PLANTS
NO_fate_pl <- rep(mean(YES_fate_pl),50)
# BIRDS
NO_fate_bird <- rep(mean(YES_fate_bird),60)

# GENERATE the final 6 FULL SCENARIOS
# Size-quantity (int freq) and size-quality (seed fate) relationships together constituted a potential quantity-quality trade-off for both
plants and birds. For simplicity, we finally excluded "plants only" and "birds only" trade-off scenarios for the no size matching case,
as these did not differ from the other neutral scenarios (i.e. those potentially calculate with sc2 and sc3 of step1).

# NO SIZE MATCHING SCENARIOS
recr.web_1 <- sc1b * outer(NO_fate_pl,NO_fate_bird)
recr.web_2 <- sc4b * outer(YES_fate_pl,YES_fate_bird)
# SIZE MATCHING SCENARIOS
```

```
recr.web_3 <- sc5b * outer(NO_fate_pl,NO_fate_bird)
recr.web_4 <- sc6b * outer(NO_fate_pl,YES_fate_bird)
recr.web_5 <- sc7b * outer(YES_fate_pl,NO_fate_bird)
recr.web_6 <- sc8b * outer(YES_fate_pl,YES_fate_bird)

# create a list with the 6 final webs of seedling recruitment
recr.webs_tot <- list(recr.web_1,recr.web_2,recr.web_3,recr.web_4,recr.web_5,recr.web_6)
names(recr.webs_tot) <- paste("Full_Scenario", 1:6, sep = "-")

#---- (VI) Calculate the 3 dimensions of seedling recruitment (Abundance and Diversity of seedlings and mean seed seed)----

dimensionsSdl <- list()
for (i in 1:6){
  Abund <- sum(recr.webs_tot[[i]]) # abundance
  Div <- diversity(rowSums(recr.webs_tot[[i]]), index = "shannon") # Shannon diversity (for plants)
  mean_size <- wtd.mean (fit_pltr,rowSums(recr.webs_tot[[i]])) # mean seed size
  dimensionsSdl[[i]] <- cbind(Abund, Div, mean_size)
}
# table with values of the three dimensions of seedling recruitment for each of the 6 scenarios
scn_recr.values <- as.data.frame(do.call("rbind", dimensionsSdl))


#--------------------------------------------------------------------------------------------------------------------
#------------------------ STEP 3: GENERATE THE DEFAUNATION SCENARIOS -----------------
#--------------------------------------------------------------------------------------------------------------------

###########################################
#---- RANDOM EXTINCTION SCENARIO -----
###########################################

# Create an array with 4 dimensions=> [1] 6 Scenarios; [2] bird richness; [3] The 3 dimensions of Sdl recruitment; [4] number of replicates in the case of random extinctions

Nrep=10000 # number of replicates
alldata <- array(NA, dim=c(6,60,3,Nrep))
# Define the names of the dimensions of the arrays
dimnames(alldata)[[1]] <- paste('Full_Scn',1:6,sep='')
dimnames(alldata)[[2]] <- paste('Bird_Rich',60:1,sep='')
dimnames(alldata)[[3]] <- c("abun","div","meansize")

# Create a unique loop for all the scenarios
## put the names of the birds and plants for the whole list with all the recruitment webs
for (i in 1:6) {
  dimnames(recr.webs_tot[[i]]) <- list(paste('p',1:50,sep=''),paste('b',1:60,sep=''))
}

# Calculate 3 dimensions of seedling recruitment for each value of species richness along a random extinction sequence
for (k in 1:6){
  for (n in 1:Nrep){
    seq.ran <- sample(colnames(recr.webs_tot[[k]]))
    web.old <- recr.webs_tot[[k]]
    for(i in 1:60){
      alldata[k, i, "abun", n] <- sum(web.old)
      alldata[k, i, "div", n] <- diversity(rowSums(web.old), index="shannon")
      alldata[k, i, "meansize", n] <- wtd.mean(fit_pltr,rowSums(web.old))
      web.old <- web.old[ ,-which(colnames(web.old)==seq.ran[i]), drop=FALSE]
    }
  }
}

# mean and confidence intervals of all the replicates for each scenario, bird richness and dimension of seedling recruitment.
alldata.ranmean <- apply(alldata, 1:3, mean) ## mean
alldata.ranCI_low <- apply(alldata, 1:3, quantile, probs= 0.025,na.rm=T) ## L CI
alldata.ranCI_high <- apply(alldata, 1:3, quantile, probs= 0.975,na.rm=T) ## H CI


####################################################################
#---- DETERMINISTIC EXTINCTION SCENARIO (size-structured defaunation) ------
#------------------ removing bird species from the largest to the smallest) --------------------
####################################################################

# Create an array with 3 dimensions=> [1] 6 Scenarios; [2] bird richness; [3] The 3 dimensions of Sdl recruitment
alldata_def <- array(NA, dim=c(6,60,3))
# Define the names of the dimensions of the arrays
dimnames(alldata_def)[[1]] <- paste('Full_Scn',1:6,sep='')
dimnames(alldata_def)[[2]] <- paste('Bird_Rich',60:1,sep='')
```

```r
dimnames(alldata_def)[[3]] <- c("abun","div","meansize")

for (k in 1:6){
  seq_def <- colnames(recr.webs_tot[[k]])
  web.old <- recr.webs_tot[[k]]
  for(i in 1:60){
    alldata_def[k, i, "abun"] <- sum(web.old)
    alldata_def[k, i, "div"] <- diversity(rowSums(web.old))
    alldata_def[k, i, "meansize"] <- wtd.mean(fit_pltr,rowSums(web.old))
    web.old <- web.old[ ,-which(colnames(web.old)==seq_def[i]), drop=FALSE]
  }
}

#---- References ----
# Dehling M, Töpfer T, Schaefer M, Jordano P, Böhning-Gaese K, Schleuning M. 2014 Functional relationships beyond species
richness patterns: trait matching in plant–bird mutualisms across scales. Global Ecol. Biogeogr. 23, 1085–1093
(doi:10.1111/geb.12193)

# Fründ J, McCann K, Williams N. 2016 Sampling bias is a challenge for quantifying specialization and network structure: lessons
from a quantitative niche model. Oikos 125, 502–513. (doi:10.1111/oik.02256)
```