Supplement to: On the inference of positive and negative interactions and their relation to abundance

Andrew J. Rominger

This supplement combines a narrative account of how to reproduce the results in the main text as well as additional analyses in support of the conclusions in the main text.

S1 Reproducibility

The results of this study can be fully reproduced by installing the package *RarePlusComMinus* (availible on GitHub) writen in R[1]. Installation requires the package *devtools*[2]:

```r
dt <- require(devtools)
if(!dt) {
    install.packages('devtools')
    library(devtools)
}

thisPack <- require(RarePlusComMinus)
if(!thisPack) install_github('ajrominger/RarePlusComMinus')
```

Two additional custom packages are required, *socorro*[3] for plotting, and *pika*[4] for simulating and analyzing species abundance distributions:

```r
socorroLoad <- require(socorro)
if(!socorroLoad) install_github('ajrominger/socorro')

pikaLoad <- require(pika)
if(!pikaLoad) install_github('ajrominger/pika')
```

All other required packages[5–8] are installed with the installation of *RarePlusComMinus*. The *RarePlusComMinus* package includes documented[9] and unit-tested[10] functions to carry out all analyses. The help documentation explains these functions, for example

```r
?plusMinus
?schoener
```

Now we can set-up our analysis.

```r
library(RarePlusComMinus)
library(pika)
library(socorro)
library(parallel)
library(viridis)

# we can now set caching to be TRUE by default
knitr::opts_chunk$set(cache = TRUE)
```

```r
# threading defaults
nthrd <- detectCores()
nthrd <- ifelse(round(nthrd * 0.8) >= nthrd - 1, nthrd - 1, round(nthrd * 0.8))
if(nthrd < 1) nthrd <- 1
```

```r
# plotting defaults
parArgs <- list(mar = c(3, 3, 0, 0) + 0.5, mgp = c(1.5, 0.30, 0), tcl = -0.25)
cexDefault <- 1.4
lwdDefault <- 2
figW <- 3.75
figH <- 3.75
```

```r
knitr::opts_chunk$set(fig.width = figW, fig.height = figH, fig.align = 'center')
```

S2 Reproducing the results of Calatayud *et al.*

First we reproduce some of the key results of Calatayuda *et al.* (CEA)[11], namely how abundances relate to positive and negative association networks. To do this we first process the data from CEA which I include as `data` in the *RarePlusComMinus* package; more information about the data can be accessed through the R help document via `?abundMats`.

```r
# load data from paper
data('abundMats')

# clean it
obsdat <- lapply(abundMats, function(x) {
    x <- ceiling(x)
    x <- x[rowSums(x) > 0, colSums(x) > 0]

    return(x)
})
```

Now we use the `plusMinus` function to calculate positive and negative association networks and abundances from the observed data:

```r
commStats <- mclapply(obsdat, mc.cores = nthrd,
                      FUN = function(x) unlist(plusMinus(x)))

commStats <- as.data.frame(do.call(rbind, commStats))

# remove studies with too few plus or minus links
commStats[is.na(commStats$pos.rho.rho) | is.na(commStats$neg.rho.rho), ] <- NA
```

The `plusMinus` function is based on, but not copied from, the function the original authors make availible at https://figshare.com/articles/Positive_associations_among_rare_species_and_their_persistence_in_ecological_assemblages/9906092. The new `plusMinus` function is more streamlined to be faster and thus able to be applied time-efficiently to many more simulations. It is unit tested against the authors' original function.

It should be noted that the authors of the original analysis[11] retained 326 studies after filtering, whereas I retain 300. This is because I removed any dataset for which there were fewer than three links in either the postive or negative networks, whereas the authors of the original analysis removed only those datasets with fewer than two links[11].

I use these calculations to make Supplementary Figure 1, which reproduce Figure 2 (B-C) from CEA[11].

```r
# ----
# helper function for making fancy histograms
specialHist <- function(x, breaks, col, add = FALSE, ...) {
    h <- hist(x, breaks, plot = FALSE)
    if(!add) plot(range(h$breaks), range(h$counts), type = 'n', ...)

    rect(xleft = h$breaks[-length(h$breaks)], xright = h$breaks[-1],
         ybottom = 0, ytop = h$counts, col = col)
}

# ----
# helper function for labeling subfigures
figLetter <- function(l, lab, bg = 'transparent', cex = 1, ...) {
    legend('topleft', legend = lab, bg = bg, box.col = 'transparent',
           x.intersp = 0, y.intersp = 0.25, adj = c(0.5, 0.5), cex = cex, ...)
}


# ----
# function to remake Fig 2(b-c)
fig2bc <- function(x, breaksRho, breaksWM, addxlab = TRUE, figLabs = LETTERS[1:2],
                   insetxprop = 0.5, insetyprop = 0.5) {

    # ----
    # relative bounds for inset figures
    insetxMax <- 0.975
    insetxMin <- insetxMax - insetxprop

    insetyMax <- 0.975
    insetyMin <- insetyMax - insetyprop

    # ----
    # set up plot
    plot.new()
    par(parArgs)
    par(cex = 1)

    # ----
    # split into two main plots and fill in
    fi <- split.screen(c(1, 2), erase = FALSE)

    # correlation differences
    screen(fi[1], new = FALSE)

    par(parArgs)
    if(addxlab) {
        par(mar = parArgs$mar + c(0, 0, 0, -0.5))
        xlab <- 'Abundance-species degree differences\n(positive - negative)'
    } else {
        par(mar = parArgs$mar + c(-2.25, 0, 0, -0.5))
        xlab <- ''
    }

    specialHist(x$pos.rho.rho - x$neg.rho.rho, xlim = c(-2, 2),
                breaks = breaksRho, col = 'gray90',
                xlab = '', ylab = 'Number of assemblages')
    mtext(xlab, side = 1, line = 2.5)
```

```
abline(v = 0, col = 'red', lty = 2, lwd = 2)
figLetter('topleft', figLabs[1])

# raw correlations
fj <- split.screen(matrix(c(insetxMin + 0.02, insetxMax + 0.02,
                            insetyMin, insetyMax), nrow = 1),
                    erase = FALSE)
screen(fj, new = FALSE)

par(parArgs)
par(cex = 0.75)
par(mgp = par('mgp') * par('cex'))

rhoYmax <- 90
rhoFmax <- max(hist(x$pos.rho.rho, breaks = breaksRho / 2, plot = FALSE)$counts,
               hist(x$neg.rho.rho, breaks = breaksRho / 2, plot = FALSE)$counts)

if(0.75 * rhoYmax < rhoFmax + 0.1 * rhoYmax) {
    rhoYmax <- 1.35 * rhoYmax
}

specialHist(x$pos.rho.rho, xlim = c(-1, 1), ylim = c(0, rhoYmax),
            breaks = breaksRho / 2, col = hsv(0.65, alpha = 0.5),
            xlab = expression("Spearman's"~rho),
            ylab = '')
specialHist(x$neg.rho.rho,
            breaks = breaksRho / 2, col = hsv(0, alpha = 0.5),
            add = TRUE)

usr <- par('usr')

points(usr[1] + c(0.2, 0.5) * diff(usr[1:2]), rep(usr[3] + 0.75 * diff(usr[3:4]), 2),
       cex = 3, bg = hsv(c(0.65, 0), alpha = 0.5), pch = 21)
text(usr[1] + c(0.2, 0.5) * diff(usr[1:2]), rep(usr[3] + 0.75 * diff(usr[3:4]), 2),
     labels = c('+', '-'), cex = 2)


# mean differences
screen(fi[2])
par(parArgs)
if(addxlab) {
    par(mar = parArgs$mar + c(0, -1, 0, +0.5))
    xlab <- 'Abundance differences\n(positive - negative)'
} else {
    par(mar = parArgs$mar + c(-2.25, -1, 0, +0.5))
    xlab <- ''
}

specialHist(x$pos.wm - x$neg.wm, xlim = c(-0.3, 0.3),
            breaks = breaksWM, col = 'gray90',
            xlab = '', ylab = '')
mtext(xlab,
      side = 1, line = 2.5)
    abline(v = 0, col = 'red', lty = 2, lwd = 2)
    figLetter('topleft', figLabs[2])


    # raw means
    fk <- split.screen(matrix(c(insetxMin - 0.04, insetxMax - 0.04,
                                insetyMin, insetyMax), nrow = 1),
                        erase = FALSE)
    screen(fk, new = FALSE)

    par(parArgs)
    par(cex = 0.75)
    par(mgp = par('mgp') * par('cex'))

    specialHist(x$pos.wm, xlim = c(0, 0.3),
                breaks = (breaksWM + 0.3) / 2, col = hsv(0.65, alpha = 0.5),
                xlab = 'Weighted mean abund.',
                ylab = '')
    specialHist(x$neg.wm,
                breaks = (breaksWM + 0.3) / 2, col = hsv(0, alpha = 0.5),
                add = TRUE)
    usr <- par('usr')
    points(usr[1] + (1 - c(0.5, 0.2)) * diff(usr[1:2]),
           rep(usr[3] + 0.75 * diff(usr[3:4]), 2),
           cex = 3, bg = hsv(c(0.65, 0), alpha = 0.5), pch = 21)
    text(usr[1] + (1 - c(0.5, 0.2)) * diff(usr[1:2]),
           rep(usr[3] + 0.75 * diff(usr[3:4]), 2),
         labels = c('+', '-'), cex = 2)

    foo <- close.screen(c(fi, fj, fk))

    invisible(NULL)
}
```

```
fig2bc(commStats, breaksRho = seq(-2, 2, by = 1/5),
       breaksWM = seq(-0.3, 0.3, by = 0.1/3))
```

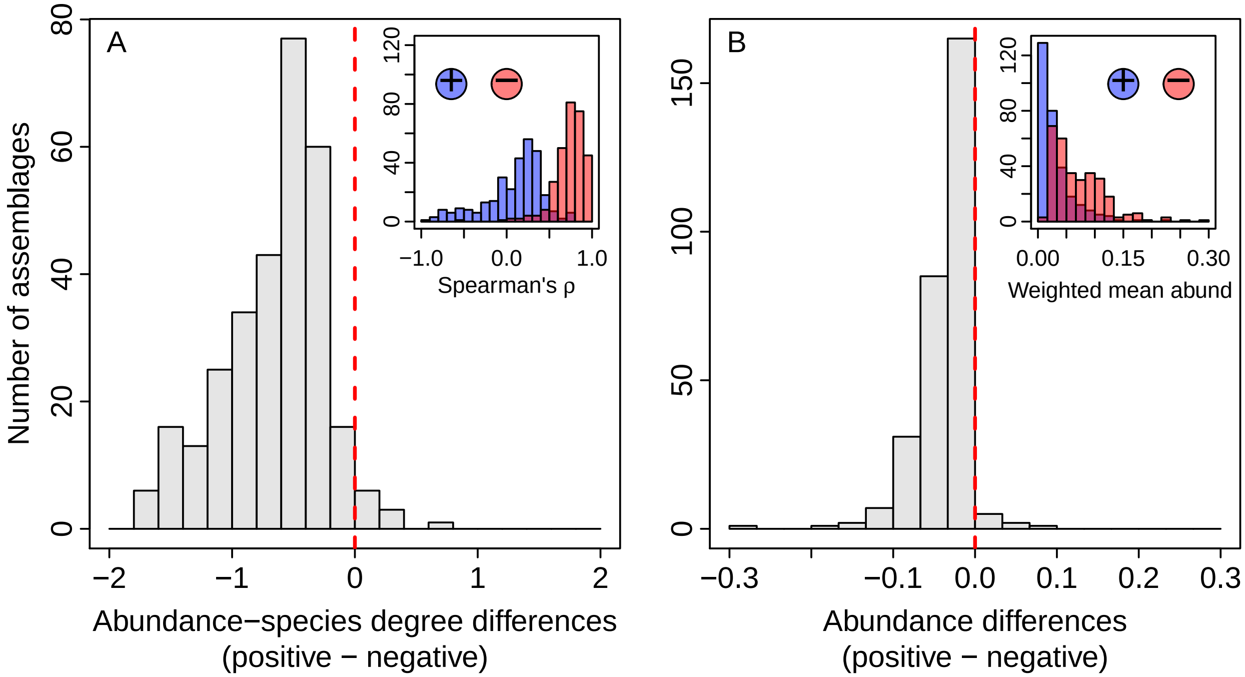
```
foo <- close.screen(all.screens = TRUE)
```

It should be noted that my inset plot of mean weighted abundances (Supplementary Fig. 1 B) differs from the original paper[11] in that their y-axis ranges from 0 to 220, while mine ranges from 0 to 120; I suspected the original authors mislabled their axis, because the scale as presented would suggest over 600 assembledges, while the number should be 326. Rescaling their axis to range from 0 to 120 brings the rough estiamte from their figure more in line with the reported number of assembledges.

S3 Exploring patterns of species abundance across space

Now we explore the shape of the spatial abundance patters in the real data provided by CEA[11]. The main function we use is `nbFit` from the *RarePlusComMinus* package which calculates summary statistics about the negative binomial and Poisson fits to spatial abundance data.

In Supplementary Figure 2 I show that the data are well fit by the negative binomial via a likelihood-based goodness of fit test[12,13]. This test scales the observed likelihood by the sampling distribution of likelihoods



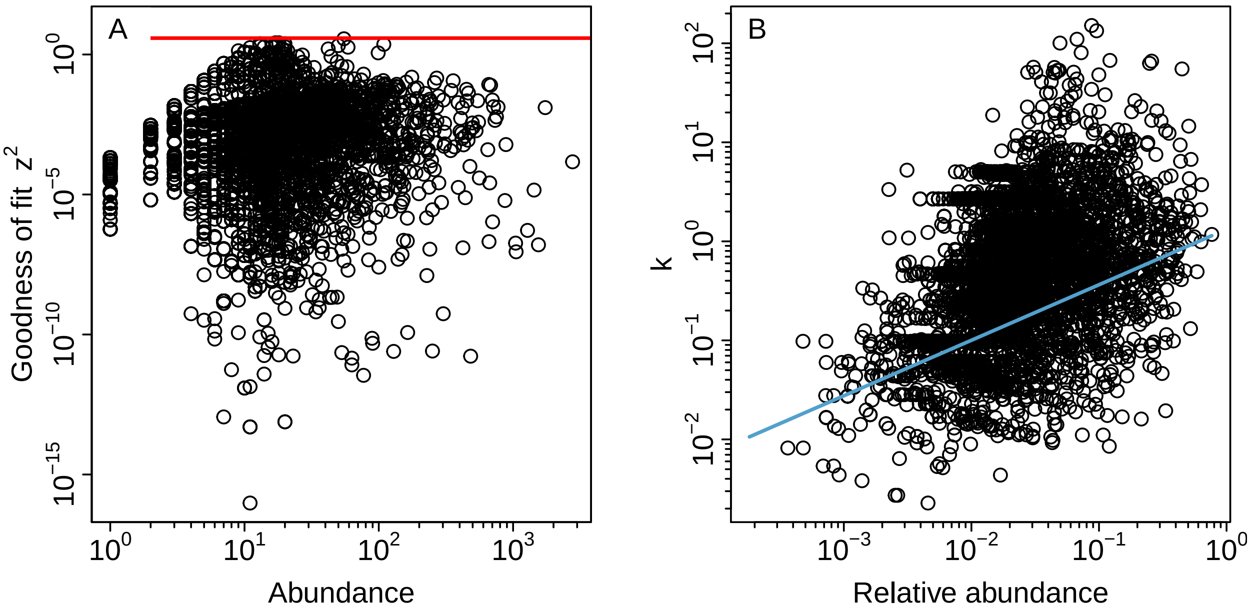
Supplementary Figure 1: Observed association between abundance and infered positive and negative interactions, reproducing Figure 2(B-C) from CEA. This figure is also found in the main text and discussed further there.

given the hypothesis that the negative binomial is the correct distribution. This test stastistic, when squared, follows a χ-squared distribution with 1 degree of freedom[13], allowing us to make a parametric cuttoff of when the data are not well represented by a negative binomial. No assembledge analyzed rejected the negative binomial distribution. In Supplementary Figure 2 I also explore the relationship of the clustering parameter k with species relative abundance. The purpose of this later analysis is to be able to simulate random but realistic data.

```
# limit to only those communities that yielded meaningful networks
summStats <- mclapply(obsdat[rownames(commStats[!is.na(commStats$pos.n), ])],
                      mc.cores = nthrd, FUN = function(x) {
    nbInfo <- nbFit(x)
    cbind(nsite = nrow(x), nspp = ncol(x), J = sum(x), nbInfo)
})

summStats <- data.frame(study = rep(rownames(commStats[!is.na(commStats$pos.n), ]),
                                    sapply(summStats, nrow)),
                        do.call(rbind, summStats))
```

```
# data for linear model of k, excluding sites that didn't produce a good network
dat4k <- with(summStats[summStats$study %in%
                            rownames(commStats[!is.na(commStats$pos.n), ]) &
                            is.finite(summStats$size), ],
              data.frame(logk = log(size),
                         loga = log(abund / J),
                         logS = log(nspp))
)
```



Supplementary Figure 2: Observed spatial species abundance distributions (SSAD) as characterized by (A) the goodness of fit of the negative binomial distribution and (B) the relationship between a given species' relative abundance and its clustering parameter k. The horizontal red line in (A) indicates the critical value above which we would reject the negative binomial; no points are above this line. The blue regression line in (B) shows the best fit log-log linear model.

```
kMod <- lm(logk ~ loga, data = dat4k)

kfun <- function(nspp, abund) {
    J <- sum(abund)

    exp(predict(kMod, newdata = data.frame(loga = log(abund / J))) +
            rnorm(nspp, sd = summary(kMod)$sigma))
}
```

```
layout(matrix(1:2, nrow = 1))

par(parArgs)
plot(summStats$abund, summStats$z, log = 'xy',
     ylim = range(summStats$z, qchisq(0.999, 1)),
     xaxt = 'n', yaxt = 'n',
     xlab = 'Abundance', ylab = expression('Goodness of fit '~z^2))
logAxis(1:2, expLab = TRUE)
abline(h = qchisq(0.95, 1), col = 'red', lwd = 2)
figLetter('topleft', 'A', bg = 'white')
box()

plot(summStats$abund / summStats$J, summStats$size, log = 'xy', xaxt = 'n', yaxt = 'n',
     xlab = 'Relative abundance', ylab = expression(k))
curve(kMod$coefficients[1] * x^kMod$coefficients[2], col = hsv(0.56, 0.6, 0.8),
      lwd = 2, add = TRUE)
logAxis(1:2, expLab = TRUE)
figLetter('topleft', 'B')
```

S4 Exploring the species abundance distribution

We also need to know the shapes of the species abundance distributions (SAD) of each assembledge to simulate realistic data. I do this using the function `fitSAD` from the custom *pika* package. I preform model selection on three standard SAD forms: the log-series, Poisson log-normal, and zero-truncated negative binomial, and record the best fit model and its parameter(s) for each assembledge.

```
mods <- c('fish', 'plnorm', 'tnegb')
sadStats <- mclapply(obsdat, mc.cores = nthrd, FUN = function(x) {
    nsite <- nrow(x)
    x <- colSums(x)
    s <- fitSAD(x, mods)

    i <- which.min(sapply(s, AIC))
    o <- s[[i]]$MLE
    if(i == 1) o <- c(o, NA)

    o <- c(i, o, sum(x), length(x), nsite)
    names(o) <- NULL

    return(o)
})

sadStats <- as.data.frame(do.call(rbind, sadStats))
names(sadStats) <- c('mod', 'par1', 'par2', 'J', 'nspp', 'nsite')
sadStats$mod <- mods[sadStats$mod]

# limit to only those sites that produced good networks
sadStats <- sadStats[rownames(sadStats) %in%
                         rownames(commStats[!is.na(commStats$pos.n), ]), ]
```

```
# helper function to make a rank abundance dist for a hypothetical community of
# `S` species given model `m` and parameters `p`

hypRAD <- function(m, p, S) {
    x <- sad(model = m, par = p[!is.na(p)])

    r <- sad2Rank(x, S)

    return(r / sum(r))
}

S <- 100

allRAD <- mclapply(1:nrow(sadStats), mc.cores = nthrd, FUN = function(i) {
    hypRAD(sadStats$mod[i], as.numeric(sadStats[i, c('par1', 'par2')]), S)
})
allRAD <- do.call(rbind, allRAD)

radEnv <- lapply(unique(sadStats$mod), function(m) {
    apply(allRAD[sadStats$mod == m, ], 2, quantile, probs = c(0.025, 0.975))
})
```

To demonstrate the marked unevenness of the SADs, in Supplementary Figure 3 we look at the outline of the shapes of all the rank abundance distributions for a hypothetical community of 100 species.

```
# ----
# helper function to make overlapping polygons more clear
specialPoly <- function(x, y, col) {
    polygon(x, y, col = colAlpha(col, 0.4), border = NA)
    polygon(x, y, border = col)
}


# ----
# plotting

par(parArgs)

plot(1, xlim = c(1, S), ylim = range(unlist(radEnv)), log = 'y', yaxt = 'n', type = 'n',
     xlab = 'Species rank', ylab = 'Relative abundance')
logAxis(2, expLab = TRUE)

polygon(c(1:S, S:1), c(radEnv[[1]][1, ], rev(radEnv[[1]][2, ])),
        col = hsv(0.56, 1, 0.8, 0.4))
polygon(c(1:S, S:1), c(radEnv[[2]][1, ], rev(radEnv[[2]][2, ])),
        col = hsv(0.05, 1, 1, 0.4))
polygon(c(1:S, S:1), c(radEnv[[3]][1, ], rev(radEnv[[3]][2, ])),
        col = hsv(0.75, 1, 0.8, 0.4))
polygon(c(1:S, S:1), c(radEnv[[1]][1, ], rev(radEnv[[1]][2, ])),
        border = hsv(0.56, 1, 0.8))
polygon(c(1:S, S:1), c(radEnv[[2]][1, ], rev(radEnv[[2]][2, ])),
        border = hsv(0.05, 1, 0.8))
polygon(c(1:S, S:1), c(radEnv[[3]][1, ], rev(radEnv[[3]][2, ])),
        border = hsv(0.75, 1, 0.8))

legend('topright',
       legend = c('Log-series', 'Poisson log-norm', 'zero-trunc. nbinom'),
       pch = 22, pt.cex = 2, pt.lwd = 1.5,
       col = hsv(c(0.56, 0.05, 0.75), 1, 0.8),
       pt.bg = hsv(c(0.56, 0.05, 0.75), 1, c(0.8, 1, 0.8), 0.4),
       bty = 'n')
```

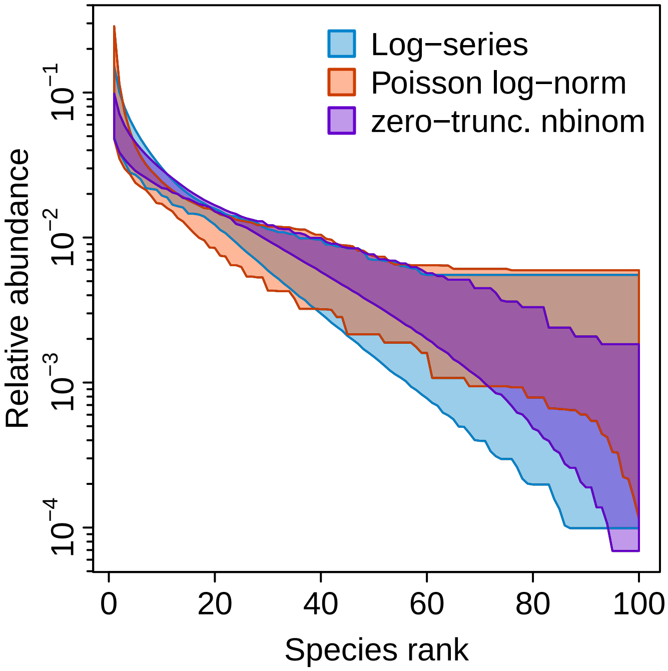
S5 Simulating random data and artifactual interactions

Now we can simulate abundance data matching the overall shapes of the observed SAD's and SSAD's but with absolutely no real correlation or interaction between species.

```
# number of simulations to run
nsim <- round(1.25 * sum(!is.na(commStats$pos.wm)))
```

I simulate 375 random assembledges, slightly more than the 300 observed assemledges because some simulated assembledges will be rejected based on the data standards used. In Figure 1 of the main text I plot these simulated results alongside the results from the real data.

```
# loop over simulation replicates
simPMData <- simPlusMinus(sadStats = sadStats, mcCores = nthrd,
                          ssadType = 'nbinom', kfun = kfun, nsim = nsim)
```



Supplementary Figure 3: Graphical summary of the shapes of the best-fit species abundance distribution models. Polygons represent the 95% conidence envelope of the rank abundance plots for each of the three SAD models considered.

Now we want to figure out if the correspondence between real and simulated results is specifically because of the shape of the SSAD (i.e. negative binomial) or if a spatially unclustered SSAD would also yield this close of a match. We do this by modeling the SSAD with a Poisson distribution.

```
# loop over simulation replicates
simPMDataPois <- simPlusMinus(sadStats = sadStats, mcCores = nthrd,
                              ssadType = 'pois', kfun = NULL, nsim = nsim)
```

Lastly we might be interested in whether we can produce a spurious association between abundance and interaction type without any reference to the observed data. For this experiment I imagine one arbitrary SAD and combine it with one of two arbitrary SSADs (either negative binomial or Poisson) and see if qualitatively similar results are found.

```
oneK <- 0.1
b <- 0.01
nsiteSimp <- 20
nsppSimp <- 50
nsimSimp <- nsim
```

In this case I use a log-series SAD with $\beta = 0.01$, a negative binomial with \$k = \$0.1, and consider an assembledge with on average 20 sites and 50 species.

```
simPMSimpNB <- simpleSim(nsiteSimp, nsppSimp, nthrd, sadfun = function(n) rfish(n, b),
                         ssadfun = function(n, mu) rnbinom(n, oneK, mu = mu),
                         nsim = nsimSimp)
```

```
simPMSimpPo <- simpleSim(nsiteSimp, nsppSimp, nthrd, sadfun = function(n) rfish(n, b),
                         ssadfun = function(n, mu) rpois(n, lambda = mu),
                         nsim = nsimSimp)
```

In Supplementary Figure 4 we see that indeed, when we use a realistic SAD and pair it with a negative binomial SSAD, we reconstruct a spurious association between rare species and positive interactions versus common species and negative interactions. When this same SAD shape is paired with a Poisson SSAD, we see this spurious association is substantially reduced.

```
# calculate breaks
wmmax <- ceiling(max(simPMSimpNB$pos.wm, simPMSimpNB$neg.wm,
                     simPMSimpPo$pos.wm, simPMSimpPo$neg.wm,
                     na.rm = TRUE) * 9) / 3

split.screen(c(2, 1))
```

```
## [1] 1 2
```

```
screen(1)
fig2bc(simPMSimpNB, breaksRho = seq(-2, 2, by = 1/5),
       breaksWM = seq(-wmmax, wmmax, by = 0.1/3), addxlab = FALSE, figLabs = c('A', 'B'))

screen(2)
fig2bc(simPMSimpPo, breaksRho = seq(-2, 2, by = 1/5),
       breaksWM = seq(-wmmax, wmmax, by = 0.1/3), figLabs = c('C', 'D'))

foo <- close.screen(all.screens = TRUE)
```

To understand why these sperious results occur we must understand what the fixed-fixed null model[14,15] does to the underlying SSAD. We know that the fixed-fixed algorithm preserves, by definition, the SAD and the total abundances across sites, but within any given species, the allocation of its abundances has a potentially large combinatorial space. In Supplementary Figure **??** I show that the fixed-fixed permutation algorithm has no net effect on Poisson SSAD's, but when the original SSAD's come from a negative binomial, the fixed-fixed algorithm tends to spread out abundances acorss more sites, and consiquently inflates the inferred k parameters of those SSAD's. This has the overall effect of pushing observed negative binomial SSAD's more toward a Poisson shape.

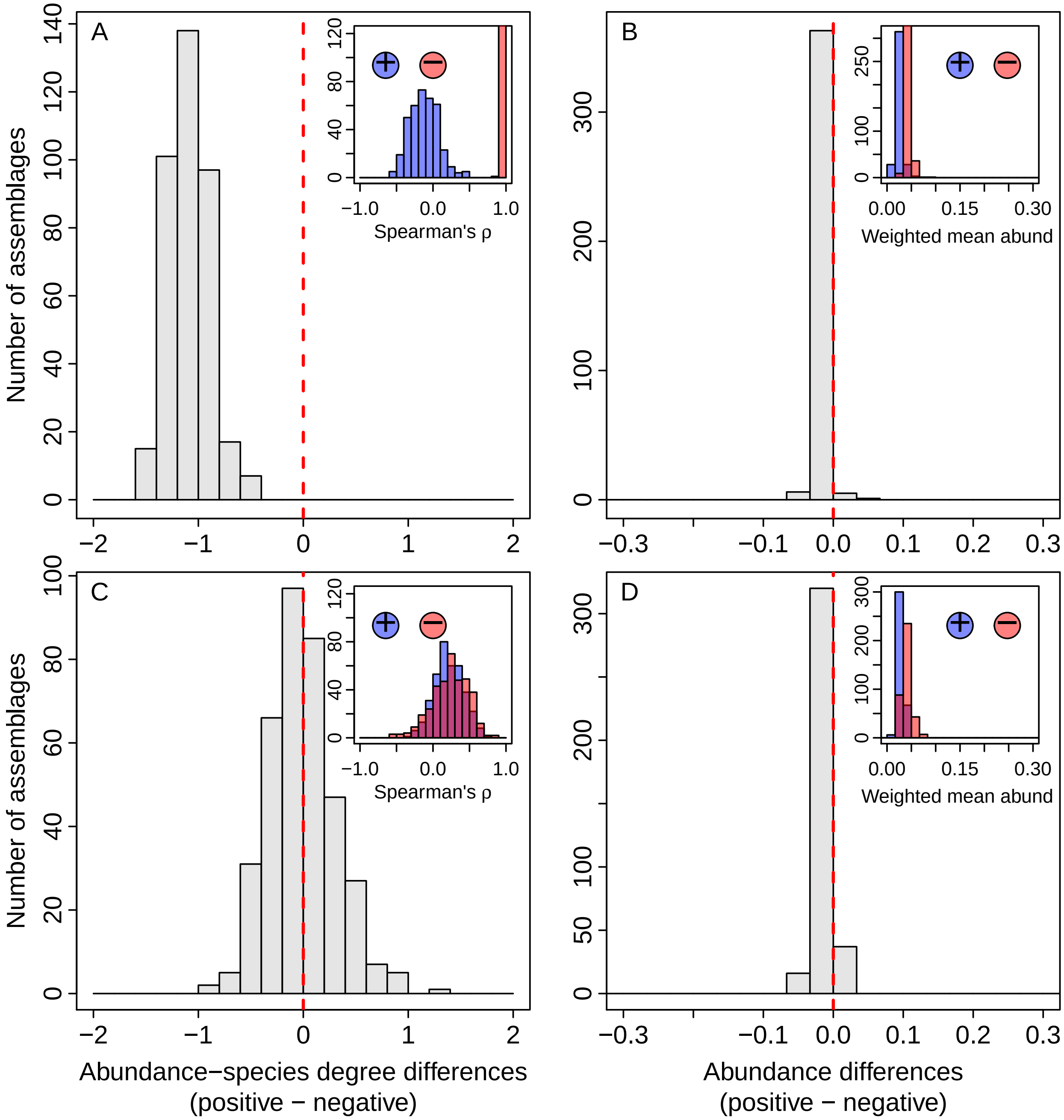
```
nsite <- 30
nspp <- 50
nsim <- 100

simNBPerm <- ssadSim(nsite, nspp, nthrd, function(n) {rfish(n, b)},
                     function(n, mu) {rnbinom(n, oneK, mu = mu)}, nsim = nsim)

simPoPerm <- ssadSim(nsite, nspp, nthrd, function(n) {rfish(n, b)},
                     function(n, mu) {rpois(n, lambda = mu)}, nsim = nsim)
```

```
layout(matrix(c(1, 1, 2, 3), nrow = 2, byrow = TRUE), heights = c(1, 5))
par(mar = rep(0, 4))
plot(1, type = 'n', axes = FALSE)
legend(1, 1, legend = 'temp', pch = 1, bty = 'n')

par(parArgs)
plot(simPoPerm$nocc, simPoPerm$null_nocc, pch = 21,
     col = 'red', bg = hsv(0, alpha = 0.1),
     xlim = c(1, nsite), ylim = c(1, nsite),
     xlab = 'True num. occupied sites',
     ylab = 'Permuted num. occupied sites')
points(simNBPerm$nocc, simNBPerm$null_nocc,
       pch = 21, col = hsv(0.52, 1, 0.8), bg = hsv(0.52, 1, 0.8, 0.1))
```



Supplementary Figure 4: Results from simulated abundances and interaction networks when only a single SAD shape and a single SSAD shape are used (as described in the text). Panel (A) shows correlations between abundance and centrality in positive or negative associations networks; panel (B) shows frequencies of different abundances in the two types of interaction networks.

```
abline(0, 1, col = 'black', lwd = 2)


foo <- rbind(simNBPerm, simPoPerm)
foo <- foo[is.finite(foo$size) & is.finite(foo$null_size), ]

par(parArgs)
plot(simPoPerm$size, simPoPerm$null_size, log = 'xy', xaxt = 'n', yaxt = 'n',
     pch = 21, col = 'red', bg = hsv(0, alpha = 0.1),
     xlim = range(foo[, c('size', 'null_size')]),
     ylim = range(foo[, c('size', 'null_size')]),
     xlab = expression('True'~k),
     ylab = expression('Permuted'~k))
logAxis(1:2, expLab = TRUE)
points(simNBPerm$size, simNBPerm$null_size,
       pch = 21, col = hsv(0.52, 1, 0.8), bg = hsv(0.52, 1, 0.8, 0.1))
abline(0, 1, lwd = 1.5)
```

Finally we very simply want to understand at a mathematical level why a spatially clustered verusus spatially even SSAD would lead to these results. To explore this we consider a very simple example of two species and their Sch similarity.

```
N <- 5
Nrare <- 5
Ncomm <- 50
nsite <- 5

rarePair <- cbind(c(1, rep(0, nsite - 1)), c(Nrare, rep(0, nsite - 1)))
commPair <- cbind(c(Ncomm, rep(0, nsite - 1)), c(0, Ncomm, rep(0, nsite - 2)))
commPairMax <- matrix(rep(Ncomm / nsite, nsite * 2), ncol = 2)
```

First we consider two rare species, one with abundance 1, and the other with abundance 5. If we consider a dataset with 5 total sites, then the configuration that maximizes the Schoener similarity between these two species is

$$\begin{bmatrix} 1 & 5 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

We can calcuate how likely this configuration is under a negative binomial versus a Poisson SSAD like this

```
k <- 0.1
nbRareP <- prod(dnbinom(rarePair[, 2], k, mu = Nrare / nsite))
poRareP <- prod(dpois(rarePair[, 2], Nrare / nsite))
```

We see that the negative binomial SSAD is more likely to maximize the Schoener similarity compared to the Poisson, and thus compared to the null model rare species will appear to be aggregated with each other.

Conversely for the common species, in our simple example represented by two species both with abundance 50, we want to compare the probabilities of minimizing the Schoener similarity between them as derived from the negative binomial versus the Poisson SSAD. This occurs in any configuration such as this one where their total abundances fail to overlap

$$\begin{bmatrix} 50 & 0 \\ 0 & 50 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

We can calculate the probabilities of any such configuration like this

```
nbCommP <- prod(dnbinom(commPair[, 1], k, mu = Ncomm / nsite)) * (nsite - 1) *
    prod(dnbinom(commPair[, 2], k, mu = Ncomm / nsite))

poCommP <- prod(dpois(commPair[, 1], Ncomm / nsite)) * (nsite - 1) *
    prod(dpois(commPair[, 2], Ncomm / nsite))
```

We can further compare this to a scenario that would maximize the Schoener similarity between common species:

$$\begin{bmatrix} 10 & 10 \\ 10 & 10 \\ 10 & 10 \\ 10 & 10 \\ 10 & 10 \end{bmatrix}$$

We calculate the probability of this configuration like

```
nbCommPMax <- prod(dnbinom(as.vector(commPairMax), k, mu = Ncomm / nsite))
poCommPMax <- prod(dpois(as.vector(commPairMax), Ncomm / nsite))
```

The conclusions of these probability calculations are disscussed in the main text.

References

1. R Core Team. *R: A language and environment for statistical computing.* (R Foundation for Statistical Computing, 2018).

2. Wickham, H., Hester, J. & Chang, W. devtools*: Tools to make developing R Packages easier.* (2018).

3. Rominger, A. J. socorro*: Helper functions for R.* (2016).

4. Rominger, A. J. pika*: Tools for macroecology.* (2016).

5. Oksanen, J. *et al.* vegan*: Community ecology package.* (2019).

6. Garnier, S. viridis*: Default color maps from 'matplotlib'.* (2018).

7. Csardi, G. & Nepusz, T. The igraph software package for complex network research. *InterJournal, Complex Systems* **1695**, 1–9 (2006).

8. Zhang, J. spaa*: Species association analysis.* (2016).

9. Wickham, H., Danenberg, P. & Eugster, M. roxygen2*: In-line documentation for R.* (2018).

10. Wickham, H. *testthat*: Get started with testing. *The R Journal* **3**, 5–10 (2011).

11. Calatayud, J. *et al.* Positive associations among rare species and their persistence in ecological assemblages. *Nat Ecol Evol* (2019).

12. Etienne, R. S. A neutral sampling formula for multiple samples and an 'exact'test of neutrality. *Ecology*

letters **10**, 608–618 (2007).

13. Rominger, A. J. & Merow, C. *meteR*: An R package for testing the maximum entropy theory of ecology. *Methods in Ecology and Evolution* **8**, 241–247.

14. Ulrich, W. & Gotelli, N. J. Null model analysis of species associations using abundance data. *Ecology* **91**, 3384–3397 (2010).

15. Patefield, W. M. Algorithm as 159: An efficient method of generating random r× c tables with given row and column totals. *Journal of the Royal Statistical Society. Series C (Applied Statistics)* **30**, 91–97 (1981).