

Méthodes de randomisation pour générer des modèles nuls dans le cadre d'analyse de sur- et sous dispersion des traits

A. Boyé

24 avril, 2018

Table des matières

Idee principale	1
Méthode	2
Calcul de l'écart avec les communautés simulées	2
Indice fonctionnel à utiliser	2
Analyse <i>single-trait</i> versus <i>mutli-trait</i>	2
Pool d'espèce sur lequel randomiser ?	3
Quelle type de randomisation pour le modèle nul	4
R functions	5
Test de significativité	6
Autres notes importantes	6
Extraits méthodes dans la littérature	6
Test	8
randomizeMatrix	8
Permatswap / Commsimulator	10
Bilbiographie	11

Idée principale

Randomizer nos communautés avec certaines contraintes pour voir l'écart entre FD observée et FD simulée
⇒ écart entre distribution observée des traits et distribution attendu sous hypothèse nulle de distribution random.

- **Overdispersion** : serait témoin de l'effet des interactions biotiques qui a tendance à favoriser la distinction des niches entre espèces ainsi qu'une plus grande régularité dans l'espace des traits (*limiting similarity*)
 - Voir Mouillot *et al.* (2013)
 - Voir discussion D'andrea & Ostling (2016) sur les limites de cette interprétation et sur les patrons à attendre
- **Underdispersion** : témoin de l'effet de filtres environnementaux
 - si c'est le cas on peut pour aller plus loin et comprendre les facteurs à l'origine de cette underdispersion faire :
 - RLQ
 - quatrième coin (lien individuel entre trait et enviro qui colle assez bien à la démarche individuelle de cette analyse)

- ou varpart sur matrice abondance de trait avec *envir* + *MEM* : voir Zhang *et al.* (2018) ou Astor *et al.* (2014) pour CWM-RDA ; dont avantages sont décrits par Kleyer *et al.* (2012)
- ou développer partition variance avec RLQ à partir de RLQ partielle (Wesuls *et al.* 2012)

Attention : cela est différent de modèle neutre qui décrit les espèces comme équivalente et explique leur patrons par les processus de dispersion, l'état démographique des populations et la distribution spatiale des espèces mais qui nécessite un paramétrage impossible à faire dans notre cas. Utiliser des modèles nuls - randomization stat - ou des modèles neutres amènent souvent des résultats différents, et il y a des partisans de chacune de ces méthodes (voir D'andrea & Ostling 2016)

Méthode

Calcul de l'écart avec les communautés simulées

L'écart entre les observations et les modèles nuls peut notamment être calculé par Standard size effect (Gotelli & McCabe 2002) :

$$SES = \frac{FD_{observed} - \mu_{null}}{\sigma_{null}}$$

Indice fonctionnel à utiliser

Le plus souvent cela est fait sur l'indice *RaoQ* (e.g. Astor *et al.* 2014 ; Zhang *et al.* 2018) mais cela a aussi été fait avec le *FRic* et la *FDis* (Laliberté *et al.* 2013 ; Mason *et al.* 2013). Potentiellement cela peut être fait avec l'indice de *distinctiveness* de Violle *et al.* (2017) qui se trouve dans le package *funrar* de Grenié *et al.* (2017)

Functional distinctiveness : mean functional distance of species *i* to the *N* other species. Can be weighted by species relative abundance because a species *i* even more distinct if it does not share traits with the most abundant species within the community (Violle *et al.* 2017). This measure can then be averaged at the scale of the community (see Grenié *et al.* 2017)

Bilan

- Favoriser *RaoQ* ou *FDis* par rapport à *FRic* (podani2009convex_hulls ; in Botta-Dukát & Czúcz 2016)
 - Favoriser *FDis* car moins sensibles aux espèces aux traits extrêmes ? (Laliberté & Legendre 2010 ; Laliberté *et al.* 2013)
 - *RaoQ* pourrait permettre d'évaluer en même temps SES_{FRed} sur la redondance en simulant $1 - \frac{RaoQ}{SimD}$
-

Analyse *single-trait* versus *mutli-trait*

Le calcul de l'indice de diversité sur les données observées ou simulées peut se faire sur chaque trait séparément (e.g. Astor *et al.* 2014 ; ou Zhang *et al.* 2018) ou sur l'ensemble des traits simultanément (voir Perronne *et al.* 2017 pour discussion du sujet)

Assembly processes may act contrastingly on different traits (Spasojevic & Suding 2012). Indeed, Spasojevic & Suding (2012) show that multiple assembly processes (abiotic filtering, aboveground competition, and belowground competition) operated simultaneously to structure plant communities and that they were therefore obscured by a single multivariate trait index and only evident by analysing functional diversity patterns of individual traits.

Perronne *et al.* (2017) : although metrics have been proposed for both single-trait and multi-trait approaches, results from multivariate approaches may be more difficult to understand, especially when several processes can lead to fitness equalization via trade-offs among traits in multi-dimensional niches (Clark *et al.*, 2007). In addition, the use of a multi-trait approach reveals several practical restrictions (Leps *et al.*, 2006) leading to a lower detection of the signature of the underlying mechanisms (Aiba *et al.*, 2013). For instance, correlations between traits may require removing some redundant traits, and using different types of variables needs appropriate standardization or transformation (Leps *et al.*, 2006; Petchey and Gaston, 2006; Bernhardt-Römermann *et al.*, 2008).

Bilan

- le faire en *single_trait* pour *RaoQ* et en *multi-trait* pour *FRed*, potentiellement en séparant trait effet et réponse ?

Pool d'espèce sur lequel randomiser ?

- Randomiser au sein de chaque habitat avec le pool d'espèce capable de vivre dans cet habitat **ou** en intertidal et subtidal séparément mais en randomisant entre les habitats biogéniques et nus au sein de chaque frange tidale ?
- Le faire aussi pour chaque année séparément ou randomiser toutes années confondus en considérant qu'il y a un pool d'espèces régional et que certaines années on en loupe ?

Selon Perronne *et al.* (2017 : p.38) "*a reference species pool includes all species that may potentially disperse to a specific site*". Certaines études prennent même en compte dans leur randomisation des espèces issues d'inventaires régionaux mais pas échantillonnées dans leur analyse.

The study of environmental filtering should be done at broad spatial scale, encompassing contrasted environments and with randomization technique that randomize species accross the different environmental conditions (Götzenberger *et al.* 2016; Perronne *et al.* 2017)

Bilan

- Prendre donc dans la randomisation toutes les années confondus pour représenter ainsi le pool d'espèces trouvé dans la région ainsi que pour chaque niveau tidal les deux habitats ensemble dans la randomisation *i.e.* randomiser entre HI et IM ensemble et entre SM et MA ensemble car les espèces trouvées dans chaque habitat aurait le potentiel de se trouver dans l'autre si ce n'était du fait de filtre environnementaux ou de competition (*i.e.* les processus qu'on test)

Attention : Cette méthode permet surtout de tester l'effet des filtres environnementaux, pour tester l'effet de la competition *i.e.* **limiting similarity**, il faut randomiser les espèces avec un pool d'espèces qui se limite aux espèces pouvant persister dans l'environnement. Cela peut être fait en randomisant au sein unité géographique définies par exemple (Perronne *et al.* 2017 : p.38) ⇒ **randomiser au sein des sites**

Chalmandrier *et al.* (2013) : Large scales thereby reinforce the detection of the effect of environmental gradients, while a small spatial scale is better suited to detect competition (Thuiller *et al.* 2010; Mouquet *et al.* 2012).

Limiting similarity and competitive dominance are harder to detect and should be assessed in homogenous environments at fine scale (Botta-Dukát & Czucz 2016; Götzenberger *et al.* 2016; Perronne *et al.* 2017). Species should not be randomized between sites from different environmental conditions (Götzenberger *et al.* 2016)

Perronne *et al.* (2017) : Alternatively, when investigating the signatures of more local ecological processes in a particular environmental context, e.g. studying trait distribution patterns associated with fine-scale disturbances and interactions, the species pool needs to be limited to a subset of species able to persist in these particular environmental conditions, i.e. presenting specific functional traits and properties, such as an appropriate phenology or some tolerance or avoidance abilities regarding a key abiotic factor (de Bello *et al.*, 2012; Lessard *et al.*, 2012). The pool can then be restricted to the species observed in each sampling unit and ignoring the species from the other sites that constitute the matrix (Bernard-Verdier *et al.*, 2012; Table 4; see Supplementary Appendix B and C for details), potentially including species from neighboring sources (Lessard *et al.*, 2012; Lessard *et al.*, 2016), or species occurring at regional scale presenting values within the trait range of each particular site, i.e. assuming that species with intermediate trait values could be adapted to local conditions of the site considered (e.g. Gotelli and Ellison, 2002; Table 4; Fig. 3A). The pool can also be defined based on operational submatrices consisting of environmentally uniform groups of sites, such as types of vegetation (Ding *et al.*, 2012; Pipenbaher *et al.*, 2013) or locations (Richardson *et al.*, 2012), also termed geographical units (Cornell and Harrison, 2014). These options specifically address the influence of local niche-based mechanisms while controlling the influence of environmental filtering playing at a larger habitat scale

Quelle type de randomisation pour le modèle nul

- Maintenir richesse fixe

*“For instance, modifying species richness at a site, i.e. row marginal sums, will impact the values of most functional metrics currently available (Mouchet *et al.*, 2010), leading to interpretations which are not only influenced by the mechanism studied, but also by the differences in species richness between observed and virtual communities caused by a methodological choice. In this case, the explicit (and more often implicit) principle of not holding constant the species richness of each community can be seen as a computational bias”* (Perronne *et al.* 2017)

- Best model for testing environmental filtering : C4, C5, T3 and T4 (Götzenberger *et al.* 2016)

C4 : randomize sample abundances within each species across all samples (‘frequency’ randomization *sensu* Picante)

- Richness of samples **is not** fixed
- Total abundances per sample **is not** fixed
- Matrix position **is not** fixed (“when matrix position is fixed in a randomization, the species that occur in a local community do not change, only their abundances do”)
- Number of samples inhabited by each species is fixed (“fixed frequency”)
- Total species abundance is fixed (“abondance totale de chaque espèce sur l’ensemble des échantillons”)

C5 : randomize species’ presence with their abundances

- Richness of samples is fixed
- Total abundances per sample **is not** fixed
- Matrix position **is not** fixed (“when matrix position is fixed in a randomization, the species that occur in a local community do not change, only their abundances do”)
- Number of samples inhabited by each species is fixed (“fixed frequency”)

- Total species abundance is fixed (“*abondance totale de chaque espèce sur l’ensemble des échantillons*”)

Algorithme Trialswap de la fonction `randomizeMatrix` de `Picante` \Rightarrow **fixed richness** \oplus **fixed colSums : total abundances of species** \oplus **fixed number of occurrence per species**

- Best model for testing limiting similarity : C2, T1 and T2 (**ATTENTION** : “In this case, one should never use randomizations that swap abundances or occurrences between sites from different environments, or trait values between species that occur in these different environments”, Götzenberger *et al.* 2016)

A DECREIRE

R functions

- `permatswap` / `commsimulator`

In `permatswap`, a special swap algorithm (‘swapcount’) is implemented that results in permuted matrices with fixed marginals and matrix fill at the same time \Rightarrow fixed total abundances per samples \oplus fixed total abundances per species \oplus fixed proportion of empty cells. **WARNING** : according to simulations, this algorithm seems to be biased and non random, thus its use should be avoided !

The algorithm “abuswap” produces two kinds of null models (based on `fixedmar`=“columns” or `fixedmar`=“rows”) as described in Hardy (2008 ; randomization scheme 2x and 3x, respectively). These preserve column and row occurrences, and column or row sums at the same time. Note that similar constraints can be achieved by the non sequential “swsh” algorithm with `fixedmar` argument set to “columns” or “rows”, respectively. \Rightarrow **pour faire modèle C5 de Götzenberger *et al.* (2016) ?**

Exemples d’utilisation

Bello (2011) : “The results presented refer to the quasiswap method of Miklos & Podani (2004), with 499 randomizations, which satisfies the requirements for equidistribution of species. The `commsimulator` function was used to generate null communities (vegan library ; R Development Core Team, 2009). The quasiswap method randomizes species composition while keeping the number of species per plot in the randomized data fixed (see de Bello *et al.*, 2009, for details).”

Maintain row and column totals selon l’aide de la fonction : sur présence absence cela préserve la richesse des échantillons mais les données d’abondance, cela devrait préserver les abondances totales des échantillons. Comportement de cette fonction à vérifier

Laliberté *et al.* (2013) : “The randomization was done using the ‘`commsimulator`’ function (‘r1 algorithm’) in the vegan R package (R Foundation for Statistical Computing, Vienna, AT).”

“r1” : non-sequential algorithm for binary matrices that preserves the site (row) frequencies, but uses column marginal frequencies as probabilities of selecting species.

- `randomizeMatrix` du package `Picante` (voir aussi `phylostruct`)

Currently implemented null models (arguments to `null.model`) : `> frequency` : Randomize community data matrix abundances within species (maintains species occurrence frequency) `> richness` : Randomize community data matrix abundances within samples (maintains sample species richness) `> independentswap` : Randomize community data matrix with the independent swap algorithm (Gotelli 2000) maintaining species occurrence frequency and sample species richness `> trialswap` : Randomize community data matrix with the trial-swap algorithm (Miklos & Podani 2004) maintaining species occurrence frequency and sample species richness

Le comportement de ces différents algorithmes est testé dans la partie **Test**

Test de significativité

Zhang *et al.* (2018) utilisent un one-sample Wilcoxon signed-rank test. Ils ne précisent par-contre pas si le test est fait en uni- ou bi-latéral. Dans Götzenberger *et al.* (2016) il le font (avec un test différent) en uni-latéral (*i.e.* one-sided) bien qu'ils test en unilatéral des deux cotés avec la même distribution. Cette méthode semble donc inappropriée car elle accroit le risque α pour ce qui paraît être un test bilatérale.

```
wilcox.test(x, y = NULL, alternative = c("two.sided", "less", "greater"), mu = 0)
```

- ☐ Vérifier comment fonctionne se test
 - ☐ Analyser comment le test de significativité se fait dans les autres papiers
-

Autres notes importantes

When trait en related to both filtering and competition, it is virtually impossible to tease environmental filtering or competitive dominance apart (Götzenberger *et al.* 2016) \Rightarrow **HI** ?

Utiliser `set.seed` : “Initially, there is no seed ; a new one is created from the current time and the process ID when one is required. Hence different sessions will give different simulation results, by default. However, the seed might be restored from a previous session if a previously saved workspace is restored.”, this is why you would want to call `set.seed()` with same integer values the next time you want a same sequence of random sequence.”

Extraits méthodes dans la littérature

— Zhang *et al.* (2018)

Reshuffling the species \times quadrat matrices in each spatial scale was done with three constraints simultaneously, following the method of Zhang *et al.* (2015), *i.e.*, keeping : i) the same number of species (species richness) per plot in the simulated and observed data ; ii) the same number of total species occurrences per region (*i.e.*, number of plots where the species occur in each group of the five spatial scales) ; and iii) the total abundance of species in a region constant (*i.e.*, the sum of the number of quadrats occupied in all plots). We implemented this using the function “randomizeMatrix” in the “picante” package in R (Kembel *et al.* 2010). We then compared the observed FD to the FD simulated in 1000 randomly assembled communities. The Standard Effect Size index (SES) following Gotelli & McCabe (2002) was used as a measure for FD patterns.[...] We used the Wilcoxon signed-rank tests to examine whether SES is significantly more than, less than or approximately equal to zero, which indicates the prevalence of significant trait divergence, trait convergence, and random distribution, respectively.

- Keep site richness + same total species occurrences + same species total abundance \Rightarrow model C5 of Götzenberger *et al.* (2016).
 - This model appears the best to evaluate the signature of environmental filtering according to Götzenberger *et al.* (2016) with the model C4 but is potentially more appropriate than C4 as it keeps richness constant contrary to model C4, which is a desirable property for the randomization (see appendix 1 of Perronne *et al.* 2017)
- Fait avec la fonction `randomizeMatrix` du package `Picante`. Dans Zhang *et al.* (2015), ils ont utilisé la fonction `commsimulator` de `vegan` with the “trialswap” method. Existe aussi dans `randomizeMatrix` mais selon l’aide préserve seulement species occurrence et sample richness.

- J'ai testé cette méthode (voir [Test](#)), et elle semble bien aussi préservé l'abondance totale de chaque espèce.
- Laliberté *et al.* ([2013](#))
 - *FRic*
 - We used a null model approach to test whether the observed functional richness in each plot was lower or higher than expected if metacommunity assembly was not influenced by species traits (i.e. leaf [N] and plant height). To do so, we created 999 random species composition matrices where species richness per plot (i.e. row) was preserved, and where column marginal frequencies were used as probabilities. The randomization was done using the `commsimulator` function ('r1 algorithm) in the `vegan` R package (R Foundation for Statistical Computing, Vienna, AT). We then computed standardized deviations from the null expectation (`FRicdev`) for each plot
 - *FDis*
 - *FDis* is related to Rao's quadratic entropy (Botta-Dukat 2005; Laliberte & Legendre 2010), which is the multivariate analogue of the weighted variance, but ***FDis* is by construction less sensitive than *RaosQ* to species with extreme trait values.**
 - To test for trait divergence, we used a different null model than that described above for functional richness. First, and most importantly, we restricted the randomization between 1 m 9 1-m quadrats within each 8 m 9 50-m plot, in order to detect local-scale assembly processes independent of any filtering effect on species pools (Mason et al. 2011; de Bello et al. 2012). To do so, we kept species abundance matrices for each plot (i.e. 20 quadrats as rows) constant but randomized rows within the species 9 trait matrix (Stubbs & Wilson 2004), using 999 randomizations.
 - Astor *et al.* ([2014](#))
 - To test for deviations from random assembly, we used three null models and calculated the standard effect size (SES; Gotelli and McCabe 2002) as (observed a-Rao minus mean of expected a-Rao) divided by standard deviation of expected a-Rao. The observed and expected values were compared, and the significance was tested with one-sided permutation tests (with 999 randomizations) using the function "as.randtest" of the package "ade4" (Dray and Dufour 2007). In one-tailed null model tests, observed values of $SES < 1.55$ (underdispersion) or > 1.55 (overdispersion) indicate significant ($\alpha = 0.05$) assembly pattern.
 - The randomization procedure was carried out with the trial swap method of Miklos and Podani (2004) implemented in R (R core team) with the `randomizeMatrix` function of the package `picante` (Kembel et al. 2010) with 999 randomizations.
 - **Model 1** : In the first null model, we randomized communities (species x plots matrix) by reshuffling the species identity among islands while keeping the same number of species per site and the same total species occurrence frequency in the whole region and calculated the abundance-weighted a-Rao for each random community. This represents the original Rao index comprising both functional richness and functional divergence.
 - **Model 2** : For the second null model, we randomized the abundances among species within communities and calculated the abundance-weighted Rao. This converts the Rao into a pure divergence component.
 - Voir Botta-Dukát & Czúcz ([2016](#)) pour pertinence de ce modèle
 - **Model 3** : For the third null model, we again used the trial swap randomization, but calculated the Rao based on species occurrences (presence/ absence) only. This resembles the functional richness component.
 - Marteinsdóttir *et al.* ([2018](#))

For trait-divergence, we used randomization that swaps abundances across all species occurring in the observed species pool

- Correspond à modèle C2 de Götzenberger *et al.* (2016) et au modèle “richness” du package **Picante** qui est approprié pour détecter la divergence lié à l’effet d’une *limiting similarity* si la randomization est faite au sein de chaque site.

For trait-convergence, we used a model that exchanges species abundances across plots.

- Correspond au modèle C4 de Götzenberger *et al.* (2016) et au modèle “frequency” du package **Picante** qui est approprié pour détecter la convergence lié à l’effet des filtres environnementaux si les gradients environnementaux sont suffisant et si le pool d’espèce randomisé permet d’inclure des espèces dans un habitat dans laquelle elle sont exclue en réalité du fait des processus de filtration

Test

randomizeMatrix

Randomisation avec le modèle “trialswap”

```
data(mite)

## Diversité des sites

#specnumber(mite)

## Abondance par site

#rowSums(mite)

## Abondance par espèce

#colSums(mite)

## Occurrence par espèces

#specnumber(mite, MARGIN=2)

# Trialswap
#-----
tmp <- randomizeMatrix(mite, null.model="trialswap", iterations = 10000)
```

La matrice randomisée est-elle identique à l’originale ?

```
# La matrice randomisée est-elle identique à l'originale?
isTRUE(all.equal(mite,tmp))
```

Diversité des sites préservée ?

```
# Diversité des sites préservée?
all(specnumber(tmp) == specnumber(mite))
```

Abondance par site préservée ?

```
# Abondance par site préservée?
all(rowSums(tmp) == rowSums(mite))
```


Abondance par espèce ?

```
# Abondance par espèce?  
all(colSums(tmp) == colSums(mite))
```

Occurrence des espèces préservée ?

```
# Occurrence des espèces préservée?  
all(specnumber(tmp, MARGIN=2)==specnumber(mite, MARGIN=2))
```

Randomisation avec le modèle “richness”

```
# Richness  
#-----  
tmp <- randomizeMatrix(mite, null.model="richness", iterations = 10000)
```

La matrice randomisée est-elle identique à l'originale ?

```
# La matrice randomisée est-elle identique à l'originale?  
isTRUE(all.equal(mite,tmp))
```

Diversité des sites préservée ?

```
# Diversité des sites préservée?  
all(specnumber(tmp) == specnumber(mite))
```

Abondance par site préservée ?

```
# Abondance par site préservée?  
all(rowSums(tmp) == rowSums(mite))
```

Abondance par espèce ?

```
# Abondance par espèce?  
all(colSums(tmp) == colSums(mite))
```

Occurrence des espèces préservée ?

```
# Occurrence des espèces préservée?  
all(specnumber(tmp, MARGIN=2)==specnumber(mite, MARGIN=2))
```

Randomisation avec le modèle “frequency”

```
# Frequency  
#-----  
tmp <- randomizeMatrix(mite, null.model="frequency", iterations = 10000)
```

La matrice randomisée est-elle identique à l'originale ?

```
# La matrice randomisée est-elle identique à l'originale?  
isTRUE(all.equal(mite,tmp))
```

Diversité des sites préservée ?

```
# Diversité des sites préservée?  
all(specnumber(tmp) == specnumber(mite))
```

Abondance par site préservée ?

```
# Abondance par site préservée?
all(rowSums(tmp) == rowSums(mite))
```

Abondance par espèce?

```
# Abondance par espèce?
all(colSums(tmp) == colSums(mite))
```

Occurrence des espèces préservée?

```
# Occurrence des espèces préservée?
all(specnumber(tmp, MARGIN=2)==specnumber(mite, MARGIN=2))
```

Bialn

Permatswap / Commsimulator

```
data(mite)

tmp <- permatswap(mite,method="abuswap", fixedmar="columns",times=1)
```

La matrice randomisée est-elle identique à l'originale?

```
# La matrice randomisée est-elle identique à l'originale?
isTRUE(all.equal(mite,tmp$perm[[1]]))
```

Diversité des sites préservée?

```
# Diversité des sites préservée?
all(specnumber(tmp$perm[[1]]) == specnumber(mite))
```

Abondance par site préservée?

```
# Abondance par site préservée?
all(rowSums(tmp$perm[[1]]) == rowSums(mite))
```

Abondance par espèce?

```
# Abondance par espèce?
all(colSums(tmp$perm[[1]]) == colSums(mite))
```

Occurrence des espèces préservée?

```
# Occurrence des espèces préservée?
all(specnumber(tmp$perm[[1]], MARGIN=2)==specnumber(mite, MARGIN=2))
```

permatswap(x,method="abuswap", fixedmar="columns") équivaut à randomizeMatrix(x,null.model=="trialswap")
Est-ce que ça donne la même

```
commsimulator(mite,method="r1")

tmp <- simulate(nullmodel(mite,method="r1"), seed=1, nsim=10)

my_rao <- function(x){
  tmp <- rao.diversity(x)
}

oecosimu(comm = sipoo, nestfun = , method = "swap",
burnin = 100, thin = 10, statistic = "evals")
```

Bibliographie

- Astor, T., Strengbom, J., Berg, M.P., Lenoir, L., Marteinsdóttir, B. & Bengtsson, J. (2014). Underdispersion and overdispersion of traits in terrestrial snail communities on islands. *Ecology and evolution*, 4, 2090–2102.
- Bello, F. de. (2011). The quest for trait convergence and divergence in community assembly : Are null-models the magic wand ? *Global Ecology and Biogeography*, 21, 312–317.
- Botta-Dukát, Z. & Czúcz, B. (2016). Testing the ability of functional diversity indices to detect trait convergence and divergence using individual-based simulation. *Methods in Ecology and Evolution*, 7, 114–126.
- Chalmandrier, L., Münkemüller, T., Gallien, L., Bello, F., Mazel, F., Lavergne, S. & Thuiller, W. (2013). A family of null models to distinguish between environmental filtering and biotic interactions in functional diversity patterns. *Journal of Vegetation Science*, 24, 853–864.
- D’andrea, R. & Ostling, A. (2016). Challenges in linking trait patterns to niche differentiation. *Oikos*, 125, 1369–1385.
- Gotelli, N.J. & McCabe, D.J. (2002). Species co-occurrence : A meta-analysis of jm diamond’s assembly rules model. *Ecology*, 83, 2091–2096.
- Götzenberger, L., Botta-Dukát, Z., Lep, J., Pärtel, M., Zobel, M. & Bello, F. (2016). Which randomizations detect convergence and divergence in trait-based community assembly ? A test of commonly used null models. *Journal of Vegetation Science*, 27, 1275–1287.
- Grenié, M., Denelle, P., Tucker, C.M., Munoz, F. & Violle, C. (2017). Funrar : An r package to characterize functional rarity. *Diversity and Distributions*, 23, 1365–1371.
- Kleyer, M., Dray, S., Bello, F., Lep, J., Pakeman, R.J., Strauss, B., Thuiller, W. & Lavorel, S. (2012). Assessing species and community functional responses to environmental gradients : Which multivariate methods ? *Journal of Vegetation Science*, 23, 805–821.
- Laliberté, E. & Legendre, P. (2010). A distance-based framework for measuring functional diversity from multiple traits. *Ecology*, 91, 299–305.
- Laliberté, E., Norton, D.A. & Scott, D. (2013). Contrasting effects of productivity and disturbance on plant functional diversity at local and metacommunity scales. *Journal of Vegetation Science*, 24, 834–842.
- Marteinsdóttir, B., Svavarsdóttir, K. & Thórhallsdóttir, T.E. (2018). Multiple mechanisms of early plant community assembly with stochasticity driving the process. *Ecology*, 99, 91–102.
- Mason, N.W., Bello, F., Mouillot, D., Pavoine, S. & Dray, S. (2013). A guide for using functional diversity indices to reveal changes in assembly processes along ecological gradients. *Journal of Vegetation Science*, 24, 794–806.
- Mouillot, D., Graham, N.A., Villéger, S., Mason, N.W. & Bellwood, D.R. (2013). A functional approach reveals community responses to disturbances. *Trends in ecology & evolution*, 28, 167–177.
- Perronne, R., Munoz, F., Borgy, B., Reboud, X. & Gaba, S. (2017). How to design trait-based analyses of community assembly mechanisms : Insights and guidelines from a literature review. *Perspectives in Plant Ecology, Evolution and Systematics*, 25, 29–44.
- Spasojevic, M.J. & Suding, K.N. (2012). Inferring community assembly mechanisms from functional diversity patterns : The importance of multiple assembly processes. *Journal of Ecology*, 100, 652–661.
- Violle, C., Thuiller, W., Mouquet, N., Munoz, F., Kraft, N.J., Cadotte, M.W., Livingstone, S.W. & Mouillot, D. (2017). Functional rarity : The ecology of outliers. *Trends in ecology & evolution*, 32, 356–367.
- Wesuls, D., Oldeland, J. & Dray, S. (2012). Disentangling plant trait responses to livestock grazing from

spatio-temporal variation : The partial rlq approach. *Journal of Vegetation Science*, 23, 98–113.

Zhang, H., Chen, H.Y., Lian, J., John, R., Li, R., Liu, H., Ye, W., Berninger, F. & Ye, Q. (2018). Using functional trait diversity patterns to disentangle the scale-dependent ecological processes in a subtropical forest. *Functional Ecology*.

Zhang, H., Qi, W., John, R., Wang, W., Song, F. & Zhou, S. (2015). Using functional trait diversity to evaluate the contribution of multiple ecological processes to community assembly during succession. *Ecography*, 38, 1176–1186.