

TD Biogéographie Fonctionnelle

Matthias Grenié

29 novembre 2018

Contexte et questions

L'idée de la biogéographie fonctionnelle est d'étudier la distribution spatiale des différents traits afin de mieux comprendre les processus qui façonnent ces traits (Violle et al. 2014).

Les plantes représentent un compartiment majeur des écosystèmes terrestres. Les traits foliaires sont donc particulièrement important pour la compréhension des cycles biogéochimiques par exemple de l'eau ou du carbone. Afin de mieux comprendre le fonctionnement des écosystèmes, il est important d'élucider les relations entre la variation des traits foliaires et leur environnement.

Plusieurs études globales sur les traits foliaires ont montré des relations contrastées (Ordoñez et al. 2009; Reich 2014). Montrant parfois des relations de certains traits positives avec la température, d'autres négatives. En outre, ces études globales n'ont pas fait de différence entre les espèces rares et communes.

Durant ce TD nous allons reproduire les analyses de (Borgy et al. 2017) pour relier les traits foliaires aux conditions environnementales en France métropolitaine, à partir de différentes bases de données.

Chargement des données

Le dossier `data` contient trois fichiers qui décrivent chacun un aspect des données :

- `divgrass_env_subset.csv` est un tableau des variables environnementales décrivant les sites, il contient les identifiants des sites ainsi qu'un certain nombre de variables environnementales que l'on décrira plus bas,
- `divgrass_relevés_subset.csv` est un tableau contenant les relevés proprement dit, avec une colonne indiquant le site, une autre l'espèce décrite et la troisième l'abondance locale de l'espèce,
- `divgrass_traits.csv` contient des valeurs moyennes de traits par espèces issues de différentes bases de données.

```
# On charge les données environnementales
divgrass_env = read.csv("data/divgrass_env.csv")

# Puis les relevés
divgrass_relevés = read.csv("data/divgrass_relevés.csv")

# Puis les traits
divgrass_traits = read.csv("data/divgrass_traits.csv")
```

On peut alors regarder la structure des données afin de les décrire. On utilisera les fonctions `head()` pour afficher les premières lignes, `summary()` pour afficher un résumé des données et `dim()` pour obtenir les dimensions des tableaux.

On peut alors décrire les colonnes des différents fichiers.

Pour `divgrass_env.csv`:

- `site`, l'identifiant du site ;
- `longitude`, la longitude du site échantillonné ;
- `latitude`, la latitude du site échantillonné ;

- `annual_temp`, la température annuelle moyenne (en $10 * ^\circ\text{C}$) du site échantillonné d'après les modélisations WorldClim (Hijmans et al. 2005) ;
- `annual_prec`, les précipitations annuelles en mm telles d'après les modélisation WorldClim (Hijmans et al. 2005) ;
- `gsl`, la longueur de la saison de croissance, en nombre de jours (voir dernière partie pour l'explication du calcul).

Pour `divgrass_relevés`:

- `site`, l'identifiant du site ;
- `species`, le nom binomial de l'espèce observée dans le site ;
- `abund`, l'abondance locale de l'espèce dans le site observée.

Enfin pour `divgrass_traits`:

- `species`, le nom binomial de l'espèce ;
- `sla`, la surface spécifique foliaire en mètre carrés par kilogramme (*Specific Leaf Area*, SLA) moyenne de l'espèce d'après la base de données TRY (Kattge et al. 2011) ;
- `ldmc`, la teneur en matière sèche des feuilles en gramme par gramme de feuille (*Leaf Dry Matter Content*, LDMC) moyenne de l'espèce d'après la base de données TRY (Kattge et al. 2011) ;
- `lnc_m`, la teneur en azote foliaire par unité de masse en mg d'azote par gramme de feuille (*Leaf Nitrogen Content per Mass*, LNC_m), d'après la base de données TRY (Kattge et al. 2011) ;
- `lpc_m`, la teneur en phosphore foliaire par unité de masse en mg de phosphore par gramme de feuille (*Leaf Phosphorus Content per Mass*, LPC_m), d'après la base de données TRY (Kattge et al. 2011).

Avant de pouvoir calculer des indices de diversité fonctionnelle il faut faire correspondre les jeux données afin de "coller" les valeurs de traits aux espèces présentes, on peut utiliser la fonction `merge()` pour le faire :

```
relevés_traits = merge(divgrass_relevés, divgrass_traits, by = "species")
head(relevés_traits)
```

On pourra procéder de la même façon pour joindre les variables environnementales identifiants de sites.

Variables environnementales

Pour une première approche des gradients environnementaux, on utilise uniquement deux variables environnementales : la température moyenne annuelle d'un site et les précipitations annuelles d'un site.

On peut commencer par représenter des histogrammes des variables environnementales.

```
# On change les paramètres graphiques pour avoir deux graphiques l'un au-dessus de l'autre
par(mfrow = c(2, 1))

# Graphe des températures
hist(divgrass_env$annual_temp / 10, col = "gray50", border = "white",
     main = "Température Annuelle Moyenne des sites de divgrass",
     xlab = "Température Annuelle Moyenne", ylab = "Effectifs")

# Graphe des précipitations
hist(divgrass_env$annual_prec / 10, col = "gray50", border = "white",
     main = "Précipitations Annuelles Moyennes des sites de divgrass",
     xlab = "Précipitations Annuelles Moyennes", ylab = "Effectifs")
```

Cartes

L'intérêt de la biogéographie fonctionnelle est aussi de visualiser les différentes variables sur une carte. Pour cela on va se servir du package `sf` qui permet de manipuler des données spatialisées.

```
library("sf")

divgrass_sf = st_as_sf(divgrass_env, coords = c("longitude", "latitude"),
                      crs = st_crs(4326))

# On change le système de coordonnées pour correspondre à la carte de France
divgrass_sf = st_transform(divgrass_sf, st_crs(2154))

france_map = read_sf("data/fr_departement/DEPARTEMENT.shp", crs = st_crs(2154))

# Carte des températures
# Le fond de carte d'abord
plot(france_map[0], reset = FALSE, main = "Température Annuelle Moyenne")
# Les points montrant les communautés
plot(st_transform(divgrass_sf["annual_temp"], st_crs(france_map)),
     pch = 16, key.pos = NULL, add = TRUE)
```

Représenter la même carte pour les précipitations.

Description de la variation des traits

Afin de voir la variation des traits on peut regarder la distribution au sein d'une seule communauté pour commencer. Par exemple le site `_00300000063701031712`

```
un_site = subset(relevés_traits, relevés_traits$site == "_00300000063701031712")

hist(un_site$sla, main = "Distribution de la surface spécifique foliaire dans une communauté",
     xlab = "Surface Spécifique Foliaire", ylab = "Effectifs",
     col = "gray50", border = "white")
```

Représenter les histogrammes pour les autres traits. **Que remarquez-vous ?**

La variance des traits dans une communauté peut nous renseigner sur les processus contraignant cette communauté (filtre environnemental particulier par exemple). Cependant selon l'hypothèse de *mass-ratio* les organismes les plus abondants (en nombre ou en biomasse) ont le plus gros impact sur les écosystèmes que les espèces les moins abondantes (Garnier et al. 2004; Grime 1998). On peut donc s'intéresser au trait moyen pondéré par les abondances relatives des espèces, c'est le *Community-Weighted Mean* (CWM), il se calcule de la façon suivante :

$$CWM = \sum_{i=1}^S p_i T_i,$$

avec i l'indice d'espèces, $p_i = \frac{N_i}{N}$ l'abondance relative de l'espèce i (N_i l'abondance de l'espèce i et N l'abondance totale dans la communauté) et T_i le trait moyen de l'espèce i .

Par exemple pour notre site préalablement sélectionné on doit d'abord calculer l'abondance totale du site N pour calculer l'abondance relative de chaque espèce :

```
# Abondance totale
total_abund = sum(un_site$abund)
```

```

# Calcul de l'abondance relative des espèces
un_site$rel_abund = un_site$abund / total_abund

# Calcul du CWM du SLA
cwm_sla = sum(un_site$rel_abund * un_site$sla)

# On peut représenter à nouveau la distribution du SLA avec le CWM
hist(un_site$sla, main = "Distribution de la surface spécifique foliaire dans une communauté",
     xlab = "Surface Spécifique Foliaire", ylab = "Effectifs",
     col = "gray50", border = "white")
abline(v = cwm_sla, col = "red", lty = 2)

```

Calculer et représenter le CWM pour les autres traits. On pourra utiliser `par(mfrow = c(2, 2))` pour obtenir 4 graphiques sur la même fenêtre.

Maintenant qu'on a pu calculer le CWM des traits foliaires sur une communauté on peut généraliser le calcul à toute les communautés.

```

# On utilise la fonction 'aggregate()' pour sommer les abondances par site
all_abund = aggregate(abund ~ site, releves_traits, sum)
colnames(all_abund)[2] = "tot_abund"

# On ajoute la colonne d'abondance totale pour chaque espèce dans chaque site
releves_traits = merge(releves_traits, all_abund, by = "site")

# On calcule l'abondance relative
releves_traits$rel_abund = releves_traits$abund / releves_traits$tot_abund

# On calcule les CWM pour chaque communauté
all_cwm_sla = aggregate(rel_abund * sla ~ site, releves_traits, sum)
colnames(all_cwm_sla)[2] = "cwm_sla"

all_cwm_ldmc = aggregate(rel_abund * ldmc ~ site, releves_traits, sum,
                        na.rm = TRUE)
colnames(all_cwm_ldmc)[2] = "cwm_ldmc"

all_cwm_lnc_m = aggregate(rel_abund * lnc_m ~ site, releves_traits, sum,
                        na.rm = TRUE)
colnames(all_cwm_lnc_m)[2] = "cwm_lnc_m"

all_cwm_lpc_m = aggregate(rel_abund * lpc_m ~ site, releves_traits, sum,
                        na.rm = TRUE)
colnames(all_cwm_lpc_m)[2] = "cwm_lpc_m"

# On colle toutes valeurs de CWM par site
all_cwm = Reduce(function(x, y) merge(x, y, by = "site"), list(all_cwm_sla,
                                                                all_cwm_ldmc,
                                                                all_cwm_lnc_m,
                                                                all_cwm_lpc_m))

```

De la même façon que plus haut on peut représenter des cartes de CWM :

```

all_cwm_sf = merge(divgrass_sf, all_cwm, by = "site")

plot(france_map[0], reset = FALSE, main = "CWM SLA")

```

```
plot(all_cwm_sf["cwm_sla"], add = TRUE, pch = 16, key.pos = NULL)
```

Représenter les cartes pour les autres traits retenus.

Relation du CWM aux variables environnementales classiques

Après avoir calculé le CWM des traits par communauté on peut maintenant les relier aux variables environnementales des sites associés aux CWM.

```
# On ajoute les variables environnementales aux CWM
all_cwm = merge(all_cwm, divgrass_env, by = "site")
```

Ici s'intéresse à la relation entre température annuelle moyenne, précipitations annuelles moyennes et CWM. On peut commencer par représenter la relation entre ces variables :

```
# Graphique simple
plot(all_cwm$annual_temp, all_cwm$cwm_sla, main = "Relation CWM SLA <-> Température Annuelle Moyenne",
      xlab = "Température Annuelle Moyenne",
      ylab = "Surface Spécifique Foliaire")
```

Le résultat est assez fouilli même si l'on a l'impression de discerner une tendance linéaire. On peut le vérifier en utilisant des points transparents :

```
plot(all_cwm$annual_temp, all_cwm$cwm_sla,
      main = "Relation CWM SLA <-> Température Annuelle Moyenne",
      xlab = "Température Annuelle Moyenne",
      ylab = "Surface Spécifique Foliaire", col = rgb(28, 134, 238, alpha = 33,
                                                    maxColorValue = 255),
      pch = 16)
```

Calculer la corrélation entre les deux variables en utilisant la fonction `cor.test()`. **Puis utiliser** la fonction `lm()` avec la syntaxe suivante `lm(mon_cwm ~ ma_variable_climatique, data = all_cwm)` pour faire une régression linéaire entre la valeur de CWM et la variable climatique.

Comparer ces relations avec celles obtenues avec les précipitations.

Un meilleur prédicteur des traits foliaires : la longueur de la saison de croissance

Les relations observées avec la température annuelle moyenne et les précipitations annuelles moyennes existent mais sont faibles. Même si à l'échelle mondiale il a été montré que ces variables climatiques déterminent la variation observée des traits de feuille à l'échelle mondiale (Ordoñez et al. 2009), elles ne sont pas complètement pertinente biologiquement. En effet, une plante sera sensible non pas uniquement aux conditions moyennes de la communauté au cours de l'année, mais aux conditions tout au long de l'année (voir par exemple Körner and Hiltbrunner (2017) pour la description de la température chez une plante).

Dans les milieux tempérés, l'un des déterminants les plus importants pour la croissance des plantes est la disponibilité en eau dans le sol. Cette disponibilité est liée à la capacité de rétention en eau du sol c'est-à-dire la quantité d'eau que le sol peut retenir. Plus le sol aura une forte capacité de rétention plus la disponibilité en eau sera grande. La disponibilité en eau du sol dépend aussi de l'évapotranspiration, c'est-à-dire l'eau transpirée par les plantes tout au long de la journée. En outre cette disponibilité en eau dépend des précipitations de la communauté, plus elles sont importantes plus l'eau sera disponible pour les plantes.

Un autre déterminant important de la croissance des plantes est la saisonnalité du climat. Pour qu'une plante puisse grandir il faut que la température de l'air ne soit pas trop froide. Dans milieux de prairies tempérées, la plupart des plantes ne montrent plus de croissance quand la température descend en dessous de 5°C.

En combinant des modèles de bilan hydrique qui prédisent la disponibilité en eau le long de l'année aux températures journalières modélisées on peut obtenir la durée de la saison favorable à la croissance des plantes. C'est ce qu'on fait Borgy et collab. (2017) pour créer une nouvelle variable synthétique qui résume ces contraintes : la longueur de la saison favorable (*Growing Season Length*, GSL).

Étant donné le plus grand réalisme biologique impliqué dans la construction de cette variable on s'attend à ce qu'elle détermine mieux les traits foliaires observés que la température annuelle moyenne ou les précipitations annuelles.

Utiliser la variable `gsl` disponible dans le tableau sites-environnement faire une carte de GSL à l'échelle de la France. Puis étudier la relation avec les CWM des différents traits foliaires et comparer aux autres variables climatiques.

Une autre façon d'étudier les relations traits-environnement : l'analyse en quatrième-coin

Plusieurs articles très récents (Miller, Damschen, and Ives 2018; Muscarella and Uriarte 2016; Peres-Neto, Dray, and ter Braak 2016) remettent en cause les bases statistiques de l'utilisation des relations CWM <-> environnement pour déterminer les traits importants dans certains environnements. Les critiques portent non pas sur l'utilisation du CWM en tant que telle (liée au *mass ratio hypothesis*), mais sur la manière de quantifier le lien qui existe avec les variables environnementales.

L'idée est de coupler trois tableaux (R : sites-environnement, L : sites-espèces, Q : espèces-traits) afin de comprendre si des variations des traits sont bien associées à des variations de l'environnement, c'est-à-dire, de trouver le quatrième coin d'association (voir figure RLQ au tableau).

On peut utiliser la fonction `fourthcorner` du package `ade4` pour la mettre en œuvre. La première étape est de mettre en forme les tableaux comme demandés par `ade4` :

```
# Tableau de variables environnementales
env_tab = divgrass_env[, c("annual_temp", "annual_prec", "gsl")]
rownames(env_tab) = divgrass_env$site

# Il faut ensuite reformater le tableau site-espèces
# On doit obtenir les sites en lignes, les espèces en colonnes et à l'intersection
# l'abondance locale des espèces
# pour nous aider on peut utiliser la fonction 'spread()' du package 'tidyr'
site_sp_tab = tidyr::spread(divgrass_relevés, species, abund)
rownames(site_sp_tab) = site_sp_tab$site
site_sp_tab = site_sp_tab[, -1] # On enlève la colonne site

# Le tableau de traits
trait_tab = divgrass_traits
rownames(trait_tab) = trait_tab$species
trait_tab = trait_tab[, c("sla", "ldmc", "lnc_m", "lpc_m")]
```

Maintenant que les trois tableaux sont formatés on peut procéder à l'analyse en quatrième coin :

```
divgrass_rlq = ade4::fourthcorner(env_tab, site_sp_tab, trait_tab)
```

Comme certains traits sont manquants l'analyse ne peut fonctionner il ne faut donc sélectionner que les espèces pour lesquelles on a les traits.

```
# On sélectionne les espèces pour lesquelles on a des traits
trait_tab_no_missing = trait_tab[complete.cases(trait_tab),]

# On fait un sous-ensemble d'espèce pour nous assurer qu'on a bien les espèces
```

```
no_missing_species = intersect(rownames(trait_tab_no_missing), colnames(site_sp_tab))

trait_tab_no_missing = trait_tab_no_missing[no_missing_species, ]

site_sp_tab_no_missing = site_sp_tab[, no_missing_species]
site_sp_tab_no_missing[is.na(site_sp_tab_no_missing)] = 0
```

Et on peut relancer l'analyse en quatrième-coin (attention cette analyse peut être assez longue et gourmande en calcul !).

```
divgrass_rfq = ade4::fourthcorner(env_tab, site_sp_tab_no_missing,
                                trait_tab_no_missing, nrepet = 99)

summary(divgrass_rfq)
plot(divgrass_rfq)
```

Références

- Borgy, Benjamin, Cyrille Violle, Philippe Choler, Pierre Denelle, François Munoz, Jens Kattge, Sandra Lavorel, et al. 2017. "Plant Community Structure and Nitrogen Inputs Modulate the Climate Signal on Leaf Traits." *Global Ecology and Biogeography* 26 (10): 1138–52. <https://doi.org/10.1111/geb.12623>.
- Garnier, Eric, Jacques Cortez, Georges Billès, Marie-Laure Navas, Catherine Roumet, Max Debussche, Gérard Laurent, et al. 2004. "Plant Functional Markers Capture Ecosystem Properties During Secondary Succession." *Ecology* 85 (9): 2630–7. <https://doi.org/10.1890/03-0799>.
- Grime, J. P. 1998. "Benefits of Plant Diversity to Ecosystems: Immediate, Filter and Founder Effects." *Journal of Ecology* 86 (6): 902–10. <https://doi.org/10.1046/j.1365-2745.1998.00306.x>.
- Hijmans, Robert J., Susan E. Cameron, Juan L. Parra, Peter G. Jones, and Andy Jarvis. 2005. "Very High Resolution Interpolated Climate Surfaces for Global Land Areas." *International Journal of Climatology* 25 (15): 1965–78. <https://doi.org/10.1002/joc.1276>.
- Kattge, J., S. Diaz, S. Lavorel, I. C. Prentice, P. Leadley, G. Bönisch, E. Garnier, et al. 2011. "TRY a Global Database of Plant Traits." *Global Change Biology* 17 (9): 2905–35. <https://doi.org/10.1111/j.1365-2486.2011.02451.x>.
- Körner, Christian, and Erika Hiltbrunner. 2017. "The 90 Ways to Describe Plant Temperature." *Perspectives in Plant Ecology, Evolution and Systematics*, May. <https://doi.org/10.1016/j.ppees.2017.04.004>.
- Miller, Jesse E. D., Ellen I. Damschen, and Anthony R. Ives. 2018. "Functional Traits and Community Composition: A Comparison Among Community-Weighted Means, Weighted Correlations, and Multilevel Models." *Methods in Ecology and Evolution*, November. <https://doi.org/10.1111/2041-210X.13119>.
- Muscarella, Robert, and María Uriarte. 2016. "Do Community-Weighted Mean Functional Traits Reflect Optimal Strategies?" *Proc. R. Soc. B* 283 (1827): 20152434. <https://doi.org/10.1098/rspb.2015.2434>.
- Ordoñez, Jenny C., Peter M. Van Bodegom, Jan-Philip M. Witte, Ian J. Wright, Peter B. Reich, and Rien Aerts. 2009. "A Global Study of Relationships Between Leaf Traits, Climate and Soil Measures of Nutrient Fertility." *Global Ecology and Biogeography* 18 (2): 137–49. <https://doi.org/10.1111/j.1466-8238.2008.00441.x>.
- Peres-Neto, Pedro R., Stéphane Dray, and Cajo JF ter Braak. 2016. "Linking Trait Variation to the Environment: Critical Issues with Community-Weighted Mean Correlation Resolved by the Fourth-Corner Approach." *Ecography*.
- Reich, Peter B. 2014. "The World-Wide 'FastSlow' Plant Economics Spectrum: A Traits Manifesto." *Journal of Ecology* 102 (2): 275–301. <https://doi.org/10.1111/1365-2745.12211>.

Violle, Cyrille, Peter B. Reich, Stephen W. Pacala, Brian J. Enquist, and Jens Kattge. 2014. “The Emergence and Promise of Functional Biogeography.” *Proceedings of the National Academy of Sciences* 111 (38): 13690–6. <https://doi.org/10.1073/pnas.1415442111>.