

**Etude des règles d'assemblages des communautés :**  
Développement d'une nouvelle approche méthodologique

Gaëlle Legras  
legras.gaelle@gmail.com

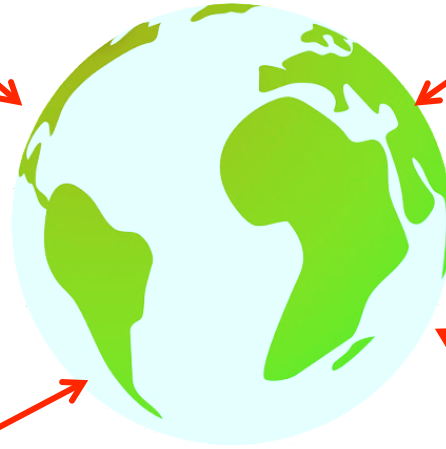
Surexploitation des  
espèces



Modification de  
l'habitat



Introduction  
d'espèces invasives



Changement climatique



- Anticiper et prédire l'impact des perturbations sur les écosystèmes



- Etudier le fonctionnement des écosystèmes



Etude des règles d'assemblages structurant les communautés

Règles d'assemblages des communautés : 3 théories

- « **Limiting similarity** » (MacArthur & Levins, 1967)
  - Seules des espèces dissimilaires fonctionnellement peuvent coexister de manière stable au sein d'un même habitat.
- « **Environmental filtering** » (Zobel, 1997)
  - Habitat exerce un filtre permettant uniquement à certaines espèces de s'y installer.
- **Théorie neutre (Hubbell, 2001)**
  - Espèces vivent et cohabitent indépendamment de leur fonction/rôle dans l'écosystème et des caractéristiques environnementales. Facteurs stochastiques uniquement.



**Nouvelle approche méthodologique**

Approche divisée en 3 étapes majeures:

### **1<sup>ère</sup> étape : Etude de la diversité fonctionnelle**

« la valeur et la gamme des traits fonctionnels dans un écosystème donné » (Diaz & Cabido, 2001 ; Tilman et al. 2001).

→ Plus un écosystème possède d'espèces exerçant des fonctions différentes, plus il est considéré comme stable et résistant face aux perturbations.

### **2<sup>ème</sup> étape : Etude des assemblages d'espèces**

Etude de la fréquence de co-occurrence des espèces afin de déterminer les « groupes » d'espèces se côtoyant le plus.

### **3<sup>ème</sup> étape : Comparaison des résultats des deux premières étapes**

Détermination de la théorie/règle d'assemblage qui prédomine dans la communauté

- Espèces fonctionnellement proches vivent au même endroit → « Environmental filtering »
- Espèces fonctionnellement proches vivent éloignées → « Limiting similarity »
- Pas de lien entre les deux → Théorie neutre

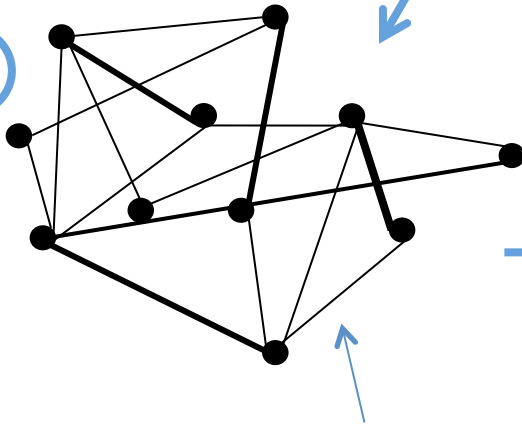
**Etape 1 : Etude de la diversité fonctionnelle**

1

	Trait 1	Trait 2	Trait t
espèce 1	x1	y1	t1
espèce 2	x2	y2	t2
espèce $i$	$x_i$	$y_i$	$t_3$

- Régime alimentaire
- Période d'activité
- Taille de l'espèce

2



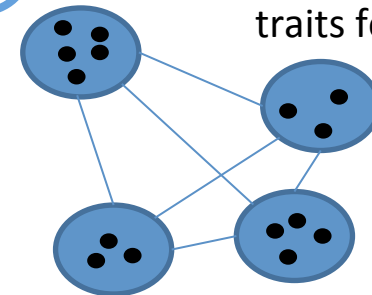
Epaisseur du lien proportionnelle au degré de  
de similarité fonctionnelle

fonction *louvain()* package  
**modMax**

*Optimisation de la  
modularité*

*Algorithme de Louvain  
(Blondel et al. 2008)*

3



**Réseau fonctionnel**

Même groupe  
=  
Valeurs proches de  
traits fonctionnels

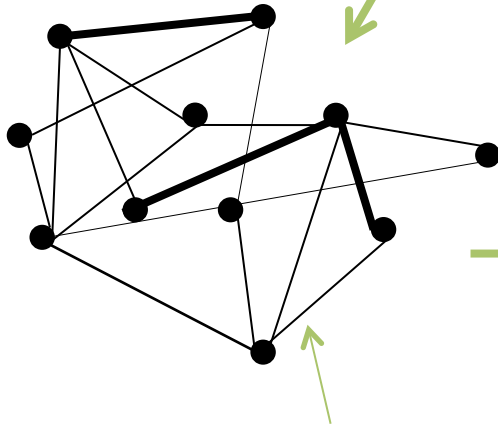
## Etape 2 : Etude des assemblages d'espèces

1

	Lieu A	Lieu B	Lieu X
espèce 1	a1.A	a1.B	a1.X
espèce 2	a2.A	a2.B	a2.X
espèce $i$	$a_i.A$	$a_i.B$	$a_i.X$

Abondance de l'espèce 1 au Lieu B

2



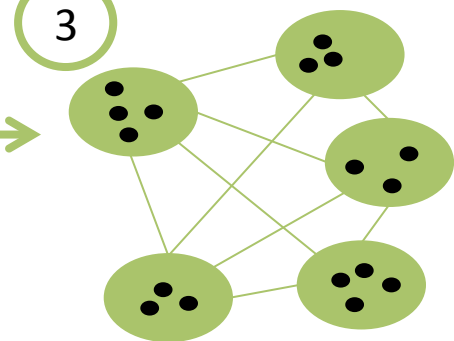
Epaisseur du lien proportionnelle au degré de co-occurrence des espèces

fonction *louvain()* package  
**modMax**

Optimisation de la modularité

Algorithme de Louvain  
(Blondel et al. 2008)

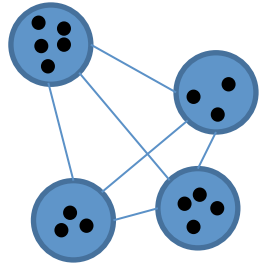
3



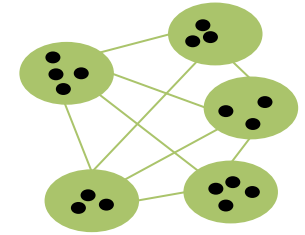
Même groupe  
=  
Vivent au même  
endroit

**Réseau de co-occurrence**

### Etape 3 : Comparaison des deux réseaux obtenus



Réseau fonctionnel



Réseau de co-occurrence

Calcul d'un indice de comparaison :

**Indice  $Dg_M$**

*inspiré de Gauzens et al. (2015)*

```
dgcomput<-function(trM){
  Dgstep<-NULL
  Dg2<-NULL
  trM[,1]<-as.factor(trM[,1])
  trM[,2]<-as.factor(trM[,2])

  for (i in 1:length(levels(trM[,1]))){
    Sub<-subset(trM,trM[,1]==levels(trM[,1])[i])
    for (j in 1:length(levels(trM[,2]))){
      Dg2[j]<-sum(Sub[,2]==j)^2/(dim(Sub)[1])^2
      Dgstep[i]<-1-sum(Dg2)
    }
  }
  DgM<-mean(Dgstep)
  names(DgM)<- "DgM"
  return(DgM)
}
```

$$Dg_M = 1 - \sum_{s=1}^m \left( \frac{g_s}{|g|} \right)^2$$

- $g_s$ : nb d'esp du groupe fonctionnel  $g$  présente aussi dans le groupe de co-occ.  $s$

Argument de la fonction : *Dataframe* de 2 colonnes

- Colonne 1 : Groupe fonctionnel
- Colonne 2 : Groupe de co-occurrence

### Etape 3 : Comparaison des deux réseaux obtenus

- Comparaison de l'indice  $Dg_M$  obtenu avec ceux d'un modèle nul

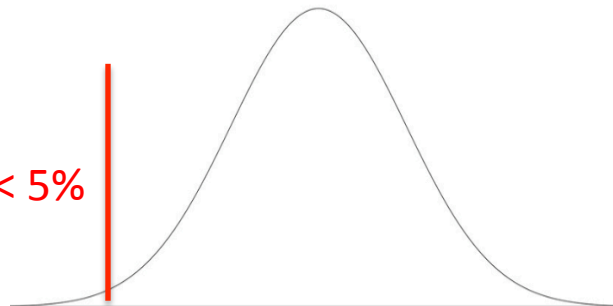
```
dgnull<-NULL  
for (i in 1:1000){  
  trM2<-trM  
  trM[,2]<-sample(trM[,2],length(trM[,2]),replace=FALSE)  
  dgnull[i]<-dgcomput(trM2)}
```

Permet de conserver le même nombre d'espèces par groupe fonctionnel et par groupe de co-occurrence

3 situations possibles

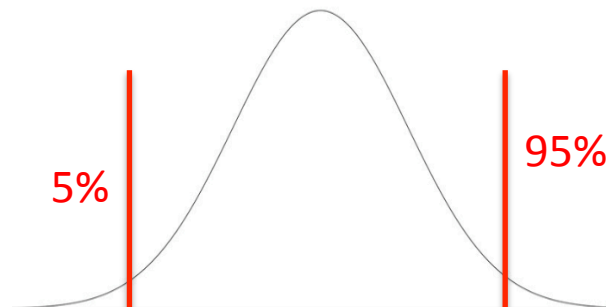


$Dg_M^{obs} < 5\%$  de  $Dg_M^{théo}$



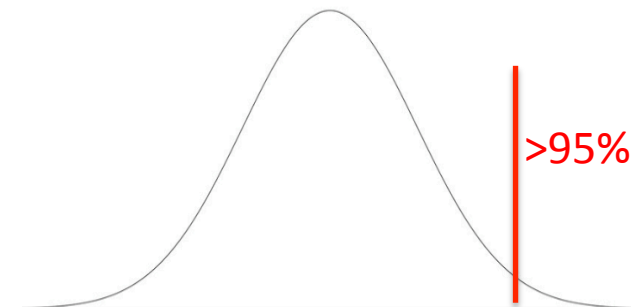
« Environnemental filtering »

$Dg_M^{obs}$  entre 5% et 95% de  $Dg_M^{théo}$



Facteurs stochastiques

$Dg_M^{obs} > 95\%$  de  $Dg_M^{théo}$



« Limiting similarity »



- Composition de communautés d'abeilles selon un degré croissant de modifications de l'habitat, Vallée de Sacramento (Californie – USA).
- Données : Etude menée par Forrest et al. 2015

*Habitat naturel (N)**Polycultures (P)**Monoculture (M)*

Degré croissant d'homogénéisation du paysage

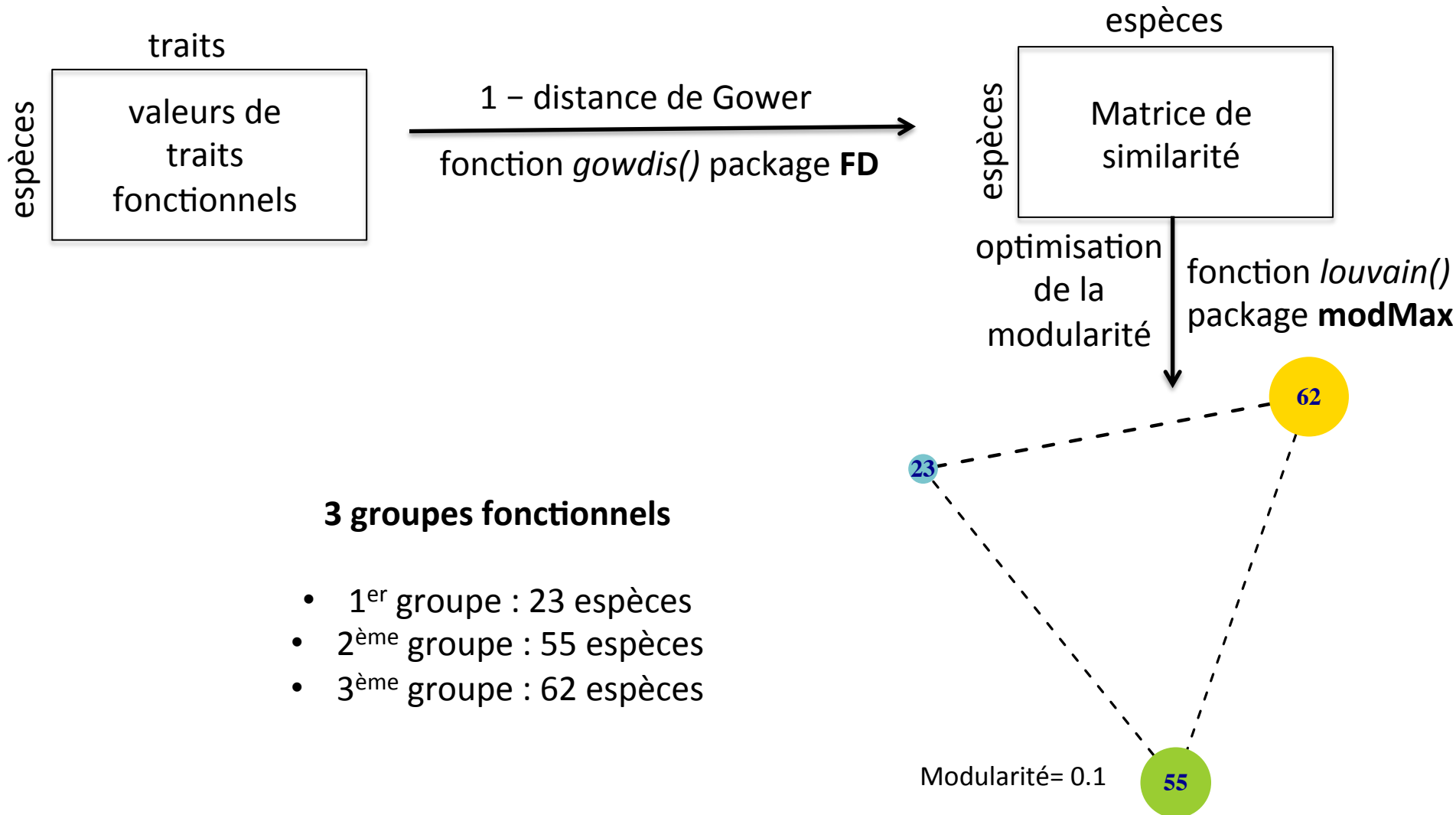


*Forrest et al. 2015 :*

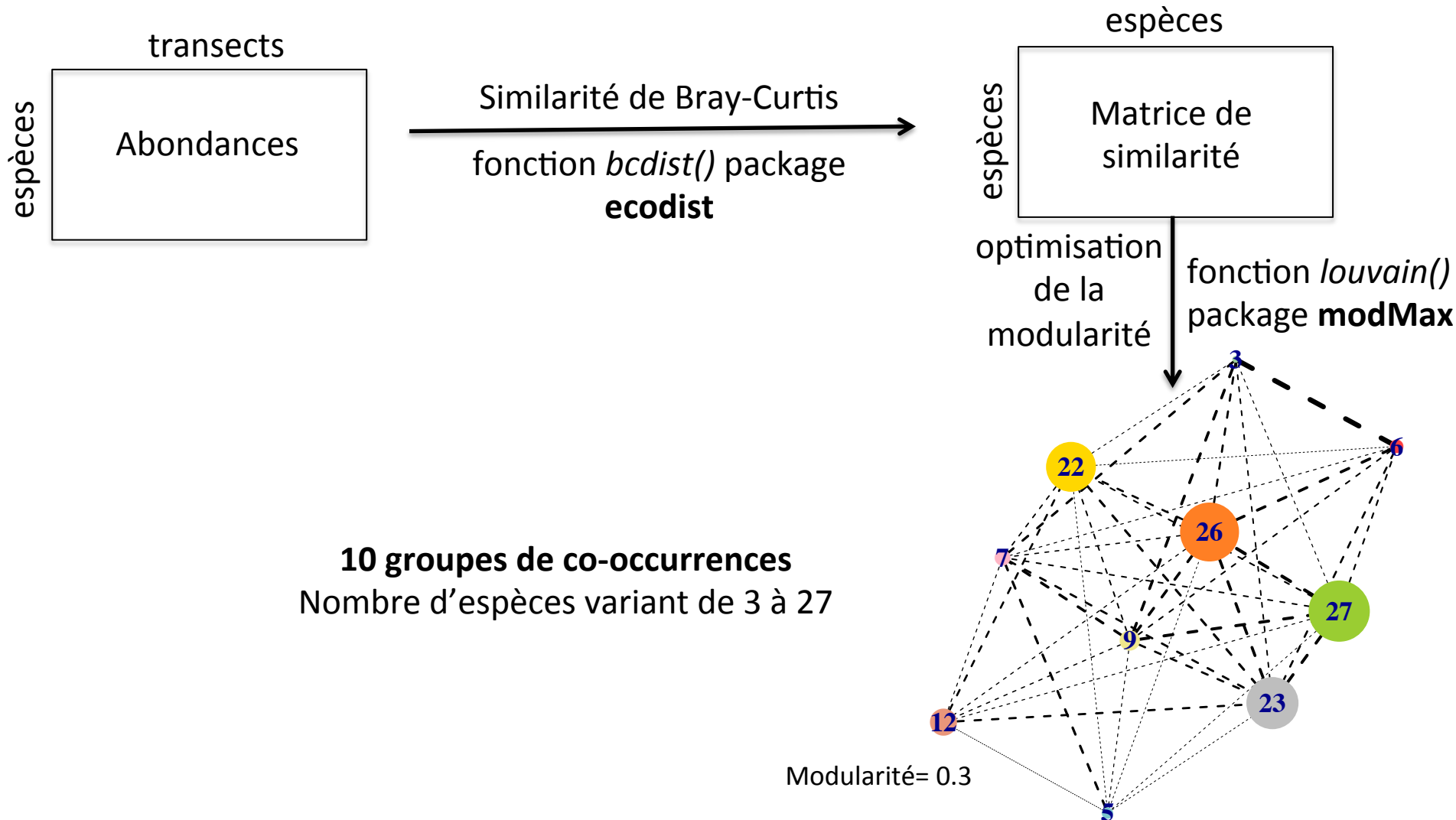
- Différence de composition fonctionnelle entre « Habitat Naturel » (N) et les deux types de fermes (P et M)
- Pas de différence de composition fonctionnelle entre les deux fermes (P et M)

Forrest, J. R., Thorp, R. W., Kremen, C., & Williams, N. M. (2015). Contrasting patterns in species and functional-trait diversity of bees in an agricultural landscape. *Journal of Applied Ecology*, 52(3), 706-715.

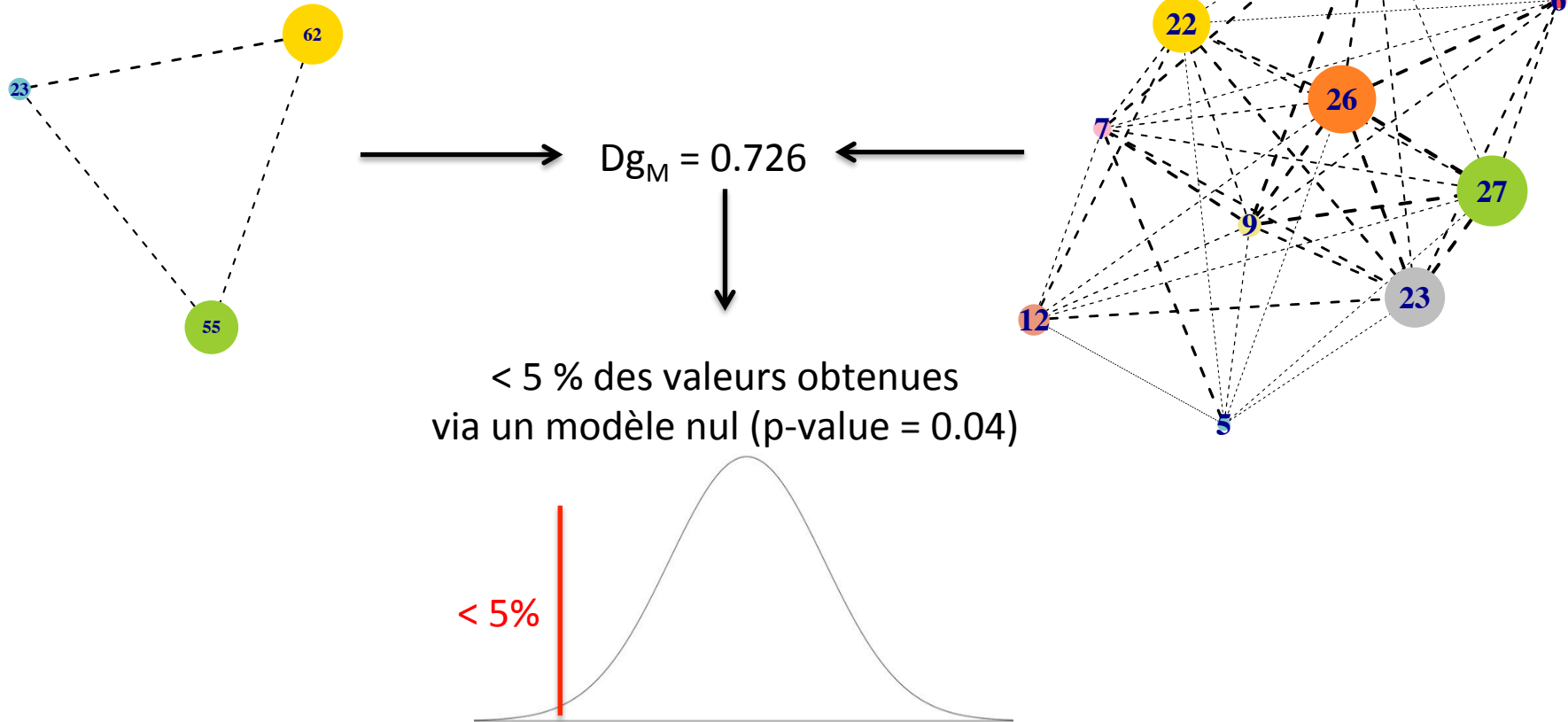
## Etape 1 : Construction du réseau fonctionnel



## Etape 2 : Construction du réseau de co-occurrence



### Etape 3 : Calcul de l'indice $Dg_M$ et comparaison avec modèle nul



Filtre environnemental (homogénéisation du paysage) agit sur la composition fonctionnelle des communautés d'abeilles

→ Résultats concordant avec ceux de Forrest et al. (2015)

## Pour aller plus loin...

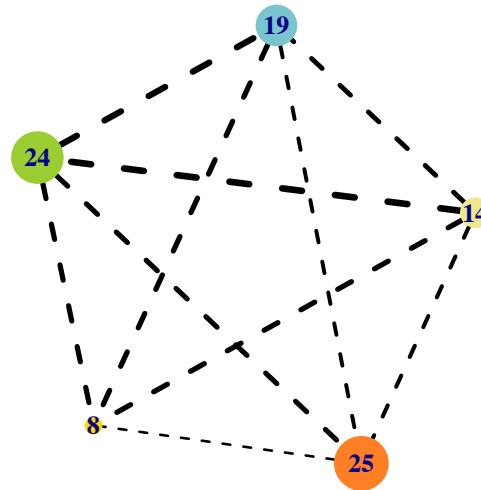
Rappel : *Forrest et al. (2015)*

- Pas de mise en évidence de différence dans la composition fonctionnelle des communautés d'abeilles entre les deux types de fermes (polycultures et monocultures)

→ Ré-application de l'algorithme de Louvain sur les deux communautés en question  
(Polycultures et Monocultures)

*Résultat :*

- Monocultures : pas de structure modulaire trouvée -> Toutes les espèces d'abeilles présentes sont fonctionnellement proches
- Polycultures : Détection d'une structure modulaire trouvée → 5 groupes fonctionnels distincts identifiés → Meilleure diversité fonctionnelle

*Conclusion :*

- Composition fonctionnelle plus importante dans les fermes de type « polycultures » que dans les monocultures
- Homogénéisation du paysage conduit à un appauvrissement de la composition fonctionnelle des communautés d'abeille

## Réalisation des graphes « réseaux » : package **igraph**

Etape 1 : Créer la matrice d'adjacence du graphe à partir des résultats précédents

→ Création d'une fonction nommée *mat.adj ()*

arguments nécessaires : matrice de similarité initiale et appartenance des espèces aux différents groupes (obtenue avec la fonction *louvain ()* précédemment appliquée)

```
mat.adj<-function(sim,group){...}
```

Éléments importants créés :

- *madj* : matrice d'adjacence
- *edgw* : contient les « poids » des différents liens entre les groupes correctement classés.
- *nclus* : contient le nombre d'éléments (i.e. d'espèces) par groupe

Etape 2 : Créer le graphe

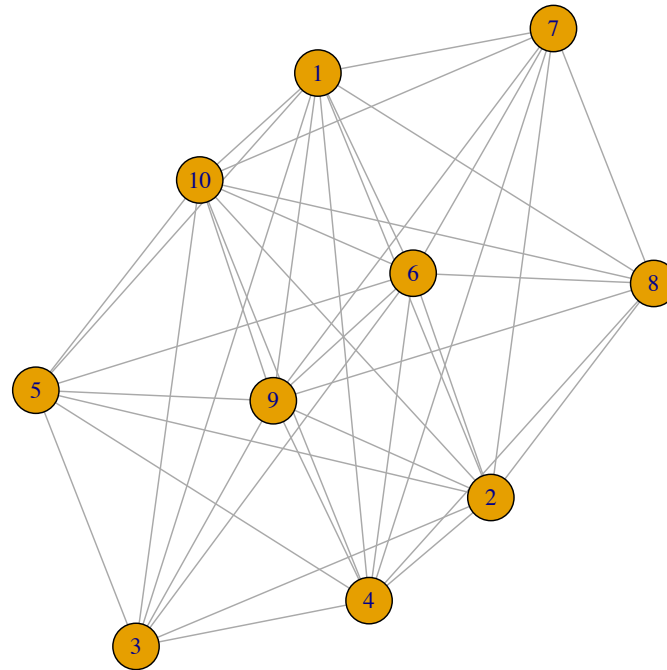
→ Fonction *graph\_from\_adjacency\_matrix ()* du package **igraph**

arguments nécessaires : matrice d'adjacence calculée dans l'étape 1

```
graph1<-graph_from_adjacency_matrix(result$madj,weighted=TRUE,mode="undirected")
```

### Etape 3 : Représentation graphique du graphe

→ fonction `plot()` : `plot(graph1)`



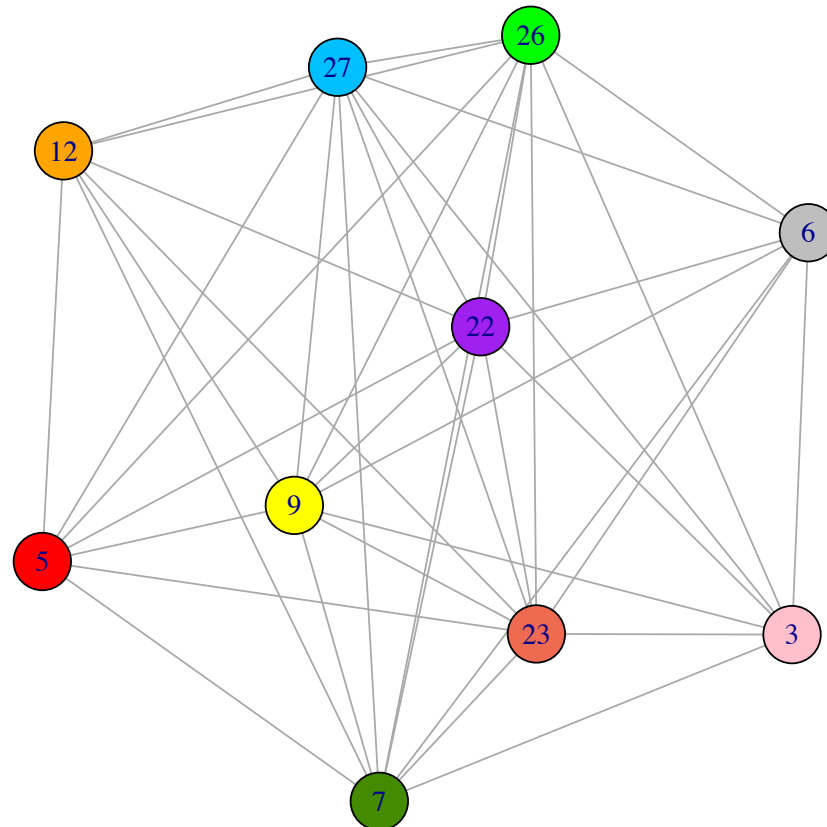


### Etape 3 : Représentation graphique du graphe

→ fonction `plot()` :

1. Ajout du nombre d'éléments par groupe et couleur

```
plot(graph1, vertex.label=result$nclus, vertex.color=c("yellow", "deepskyblue", "red",  
  , "green", "orange", "purple", "pink", "grey", "chartreuse4", "coral2"))
```

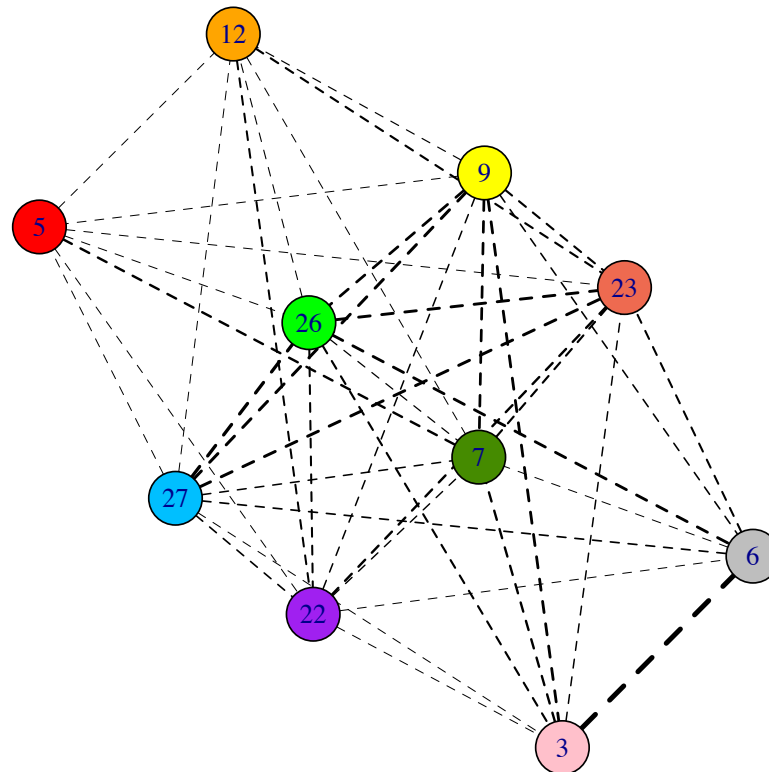


### Etape 3 : Représentation graphique du graphe

→ fonction `plot()` :

1. Ajout du nombre d'éléments par groupe et couleur
2. Epaisseur des liens entre groupes selon leur valeur

```
plot(graph1, vertex.label=result$inclus,  
vertex.color=c("yellow", "deepskyblue", "red", "green", "orange", "purple", "pink", "grey",  
"chartreuse4", "coral2"),  
edge.width=result$edgw*20, edge.lty=2, edge.color="black »)
```



→ Nombreuses autres options ... (cf. doc du package **igraph**)

## Conclusion générale

- Développement d'une nouvelle approche méthodologique permettant une meilleure étude des règles d'assemblages structurant les communautés
- Approche pouvant être totalement réalisée sous le logiciel R
  - Utilisation de fonctions pré-existantes : package modMax et igraph
  - Création de trois nouvelles fonctions :
    - 2 dédiées à l'analyse : *dgcomput ()* et *dgnull()*
    - 1 dédiée à la représentation graphique : *mat.adj()*

# Etude des règles d'assemblages des communautés :

## Développement d'une nouvelle approche méthodologique

Gaëlle Legras  
legras.gaelle@gmail.com

Merci pour votre attention !