Recherche de communautés dans les graphes

Maxime CHAZALVIEL

PRISE EN MAIN DU GRAPHE

```
>graph = simplify(graph)
```

Quelle est l'effet de cette fonction ?

Discutez cette observation en la ramenant à la façon dont a été construit le graphe.

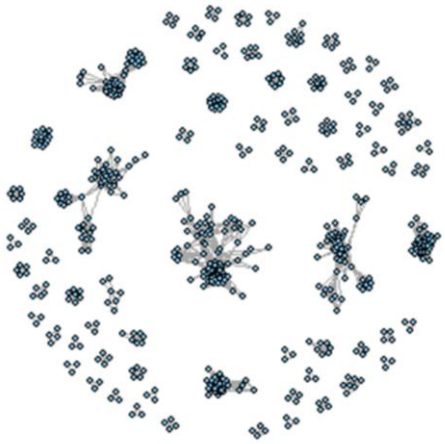
Elle supprime les arêtes multiples entre deux sommets. La moitié des arêtes du graphe sont supprimées. Nous utiliserons par la suite le graphe simplifié.

TAUX DE PARALOGIE

Quelle peut être l'origine de ce taux élevé da paralogie ?

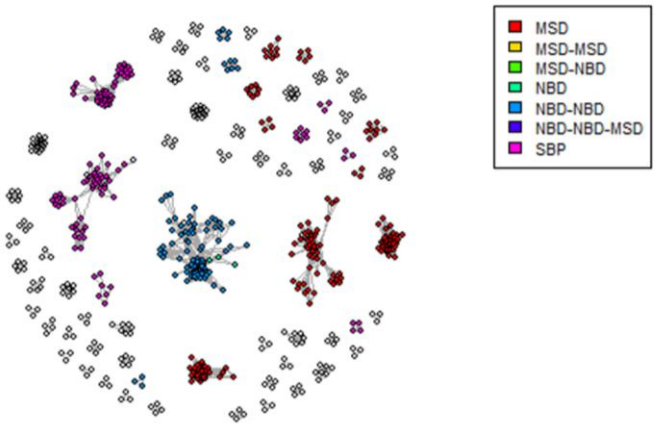
Après avoir codé la fonction donnant le taux de paralogie et l'avoir appliquée au graphe, les résultats montrent des taux de paralogies élevés. Ceci est dû au fait que le jeu de données présente de nombreux gènes pour peu de souches.

REPRESENTATION GRAPHIQUE



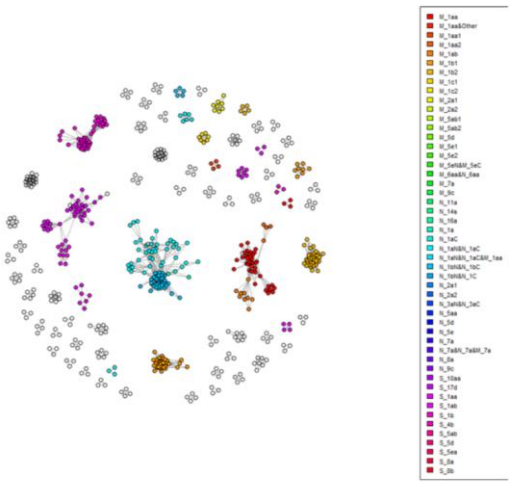
On voit un nombre conséquent de sous graphes isolés. C'est un graphe peu dense, peu connecté avec beaucoup de composantes connexes. On peut penser que ces petites composantes connexes représentent les protéines se situant en amont et en aval de nos protéines du transporteur ABC.

ANNOTATION EN DOMAINE, SOUS FAMILLES



Annotation du graphe en domaine

Remarque : à l'intérieur même d'un domaine nous observons une diversité assez importante au niveau des sous-graphes.



Annotation du graphe en sous familles

Il y a beaucoup de sous familles et la colorisation n'est pas celle attendue. Cette annotation du graphe semble assez peu pertinente.

COMPOSANTES CONNEXES

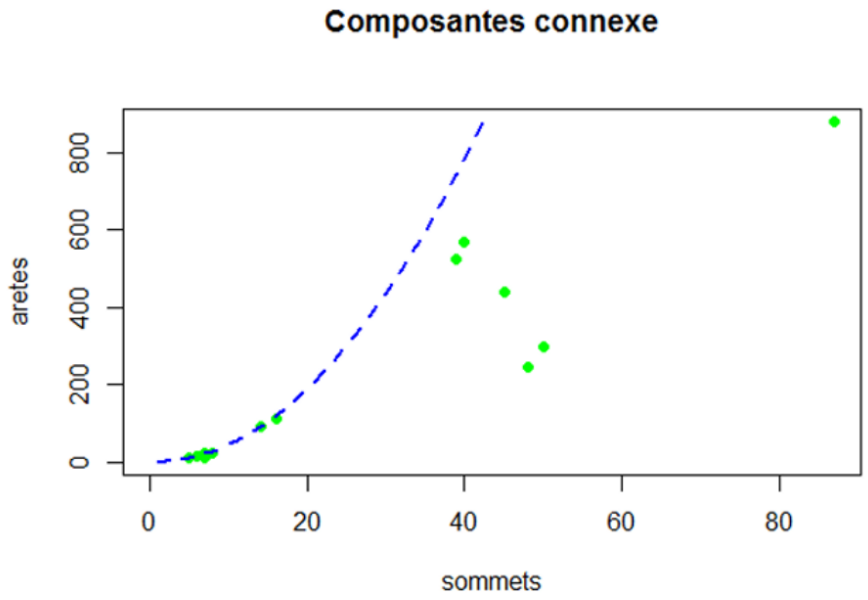
Combien de composantes connexes avez-vous obtenu ?

La fonction decompose.graph() permet d'obtenir 26 composantes connexes.

PROPRIETES DES COMPOSANTES CONNEXES

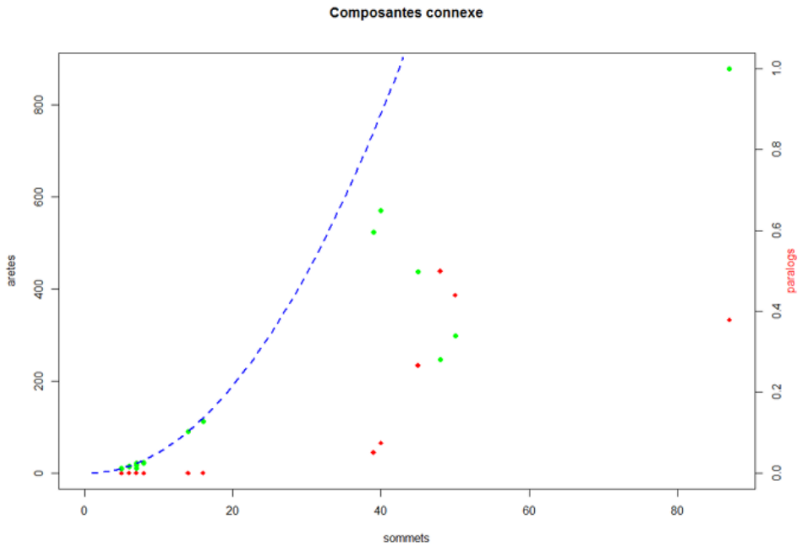
Quelle est le nombre d'arêtes maximum possible dans un graphe de n sommets ?

La formule permettant de calculer le nombre maximum d'arêtes dans un graphe de n sommets est : $(n^2-n)/2$



Représentation du nombre d'arêtes des composantes connexes en F

On ne peut pas comparer toutes les composantes connexes entre elles, seulement quelques-unes suivent le maximum d'arêtes par sommet. Les composantes connexes situées sur la courbe bleue sont donc des cliques et il est inutile de chercher des communautés dans celles-ci. En revanche, les graphes qui s'éloignent de la distribution max sont intéressants car on va pouvoir trouver des communautés du fait qu'ils sont peu denses. Pour compléter l'analyse, nous allons ajouter le taux de paralogies de chaque CC.

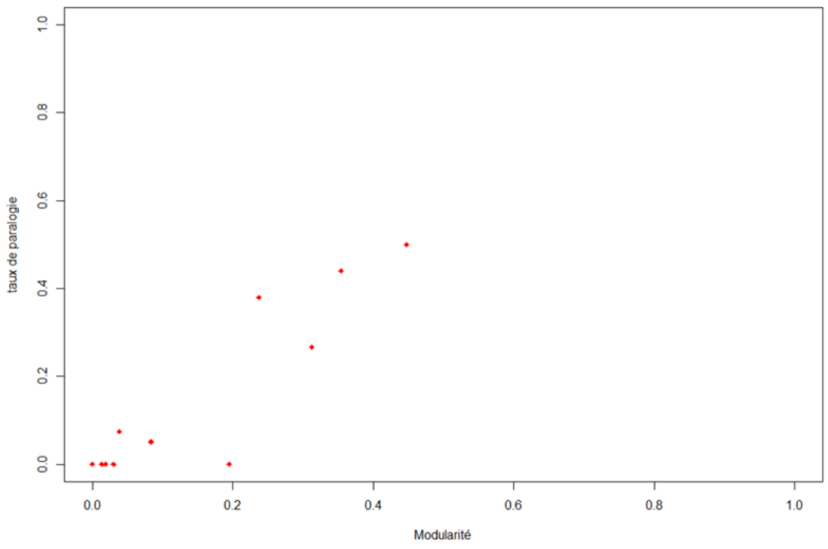


Taux de paralogies en fonction du nombre d'arêtes et de sommets sur les différentes composantes connexes

Lorsque l'on se rapproche de la distribution de densité maximale, on remarque que le taux de paralogie a tendance à être bas voire nul.

Plus on s'éloigne, plus il augmente Ceci est vrai jusqu'à un certain nombre de sommets au-delà duquel l'éloignement par rapport à la courbe n'est plus proportionnel au taux de paralogie (voir la valeur extrême à 90 sommets).

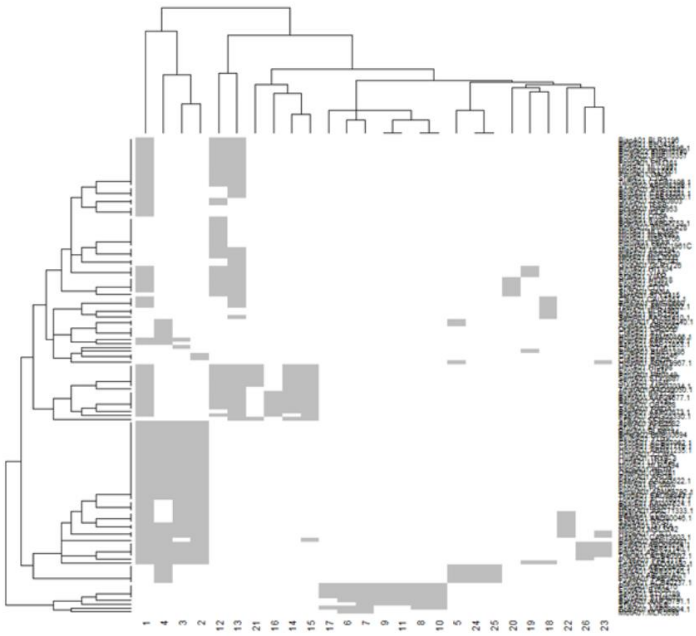
TAUX DE PARALOGIE ET MODULARITE



Evolution du taux de paralogie en fonction de la modularité

La modularité est corrélée positivement avec le taux de paralogie. On observe cependant qu'il existe un ensemble de sommets ayant un taux de paralogie nul et pourtant ayant une modularité positive. Donc sans doute une structure dans ce sous graphe que l'on ne pouvait observer avec le graphique précédent. On doit ainsi pouvoir les séparer en communauté d'orthologues.

RELATION ENTRE VOISINAGE ET DECOUPAGE EN CC



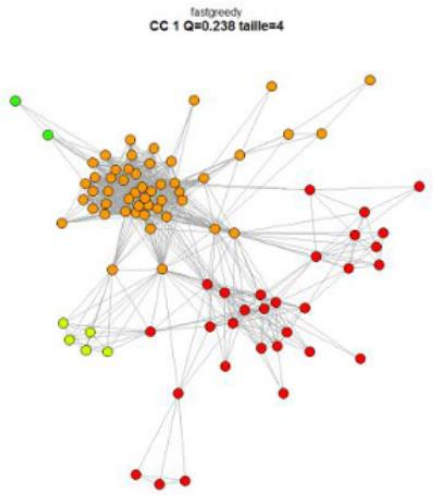
Heatmap de la répartition des gènes dans les composantes connexes

Les composantes connexes 1,3,4 sont souvent ensemble mais la composante 1 est partagée par plusieurs systèmes Il va sans doute falloir la découper car elle doit avoir beaucoup de paralogie. L'hypothèse sous-jacente étant bien entendu que les gènes d'un système Co-évoluent. On voit que pour certains d'entre eux c'est le cas (1 et 3 sont souvent ensemble de même que 12 et 13). Dans la suite de ce TP, nous allons essayer de mieux découper ces composantes connexes.

DECOMPOSITION EN COMMUNAUTES

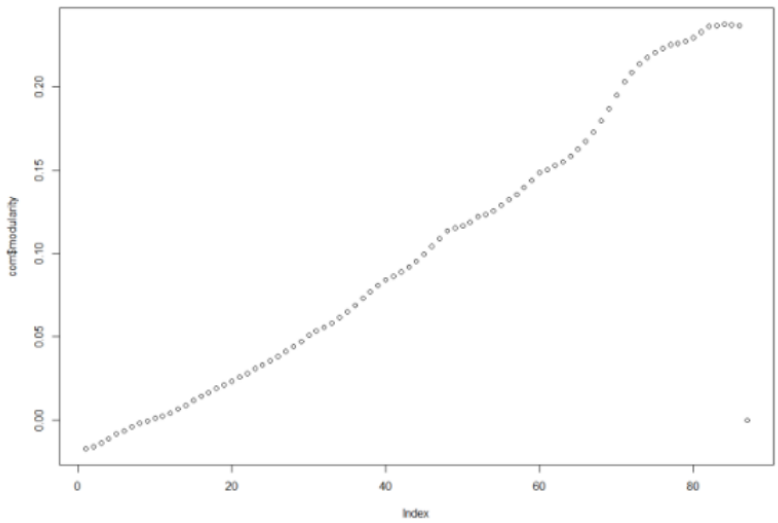
Utilisation de l'algorithme de découpage fastgreedy. Cet algorithme utilise la modularité et améliore de manière itérative la qualité du partitionnement.

FASTGREEDY



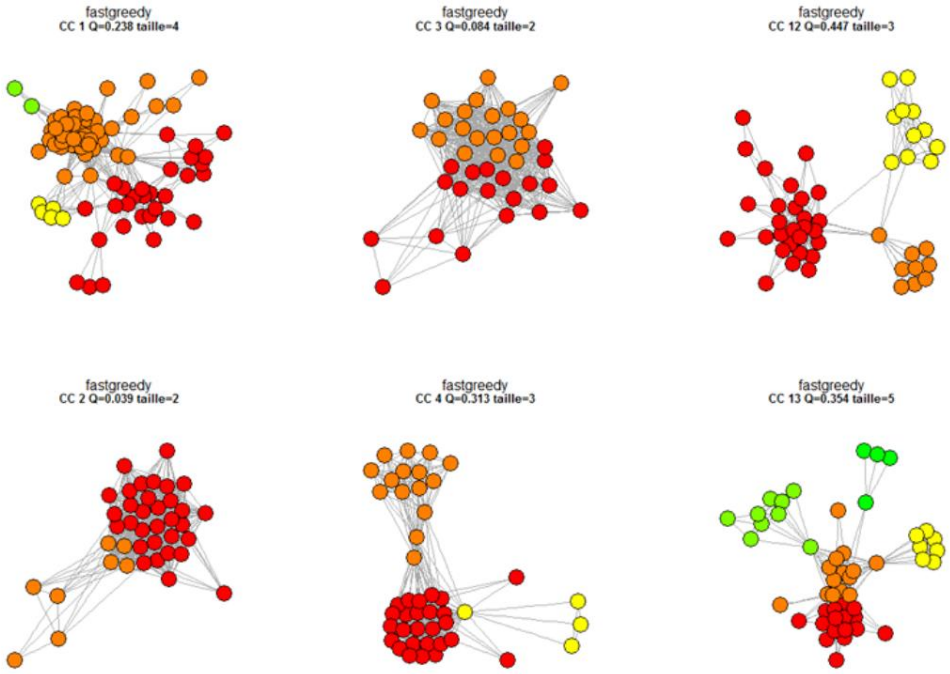
Découpage ne communautés pour la CC_1

On remarque que le groupe de deux vert est peut être sur-découpé. Le groupe de jaune possède des relations avec les rouges et les oranges. Finalement, c'est assez difficile à interpréter.



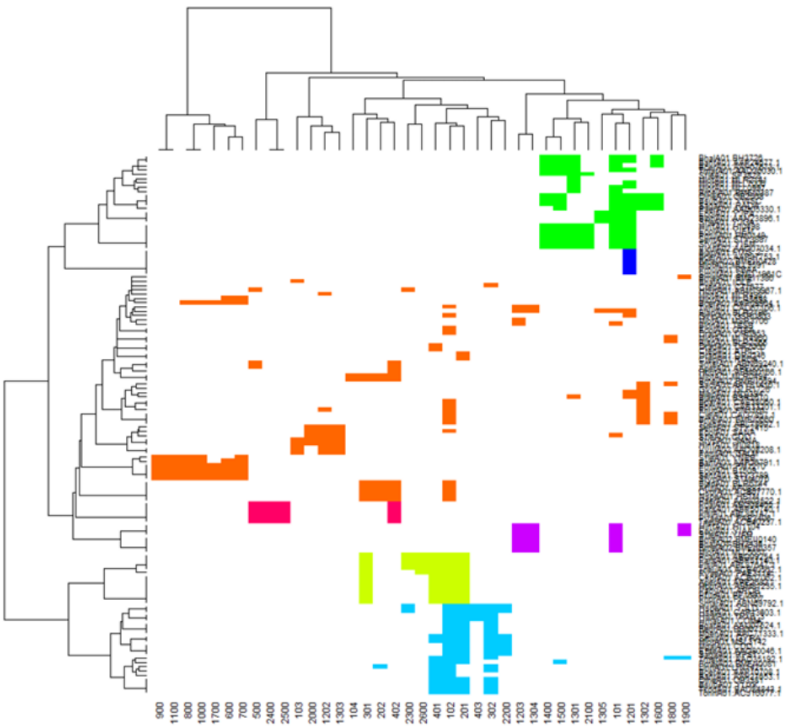
Evolution de la modularité pas à pas

Voyons maintenant le découpage en communauté de toutes les composantes connexes ayant un taux de paralogie supérieur à 0.



Découpage en communautés des CC (taux de paralogies > 0)avec la méthode fastgreedy

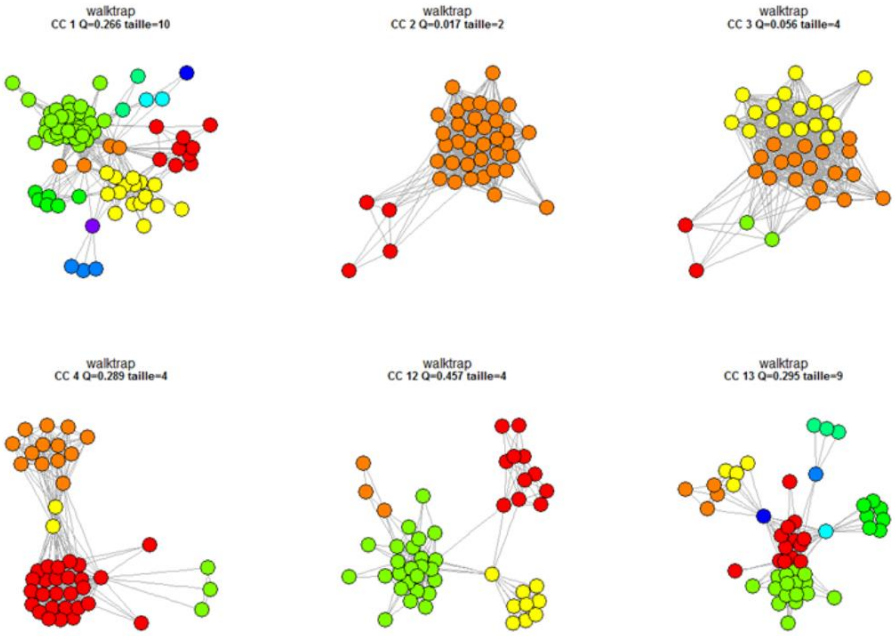
Et la représentation sous forme de heatmap avec la méthode Ward et l'utilisation du coefficient de Jaccard pour le calcul de la distance :



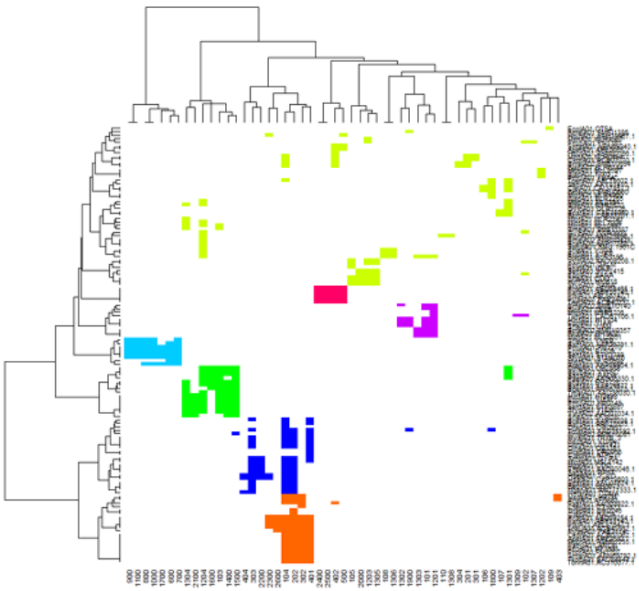
Heatmap des CC avec la méthode de partitionnement de Ward et la distance de Jaccard

C'est encore un peu trop découpé, on retrouve des éclatements. Toutefois, on retrouve les corrélations entre partenaires. On peut arranger le sur/sous découpage de la méthode fastgreedy en modifiant les paramètres et les méthodes utilisés. Il existe plusieurs méthodes de partitions en communautés telles que walktrap, la betweeness ou encore spinglass. Ci-dessous, les résultats de ces différentes méthodes :

WALKTRAP

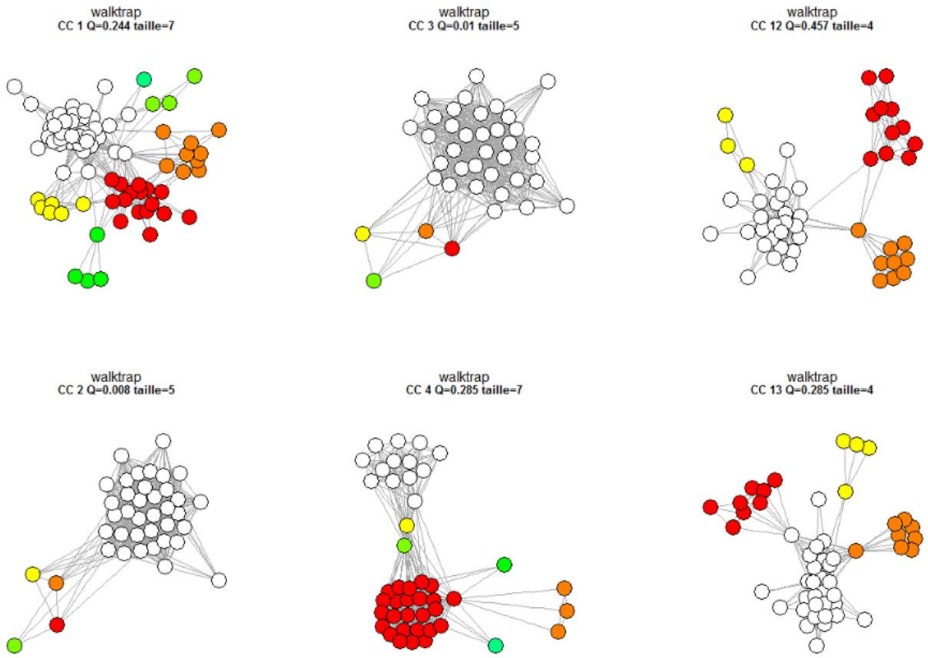


Découpage en communautés des CC (taux de paralogies > 0)avec la méthode walktrap



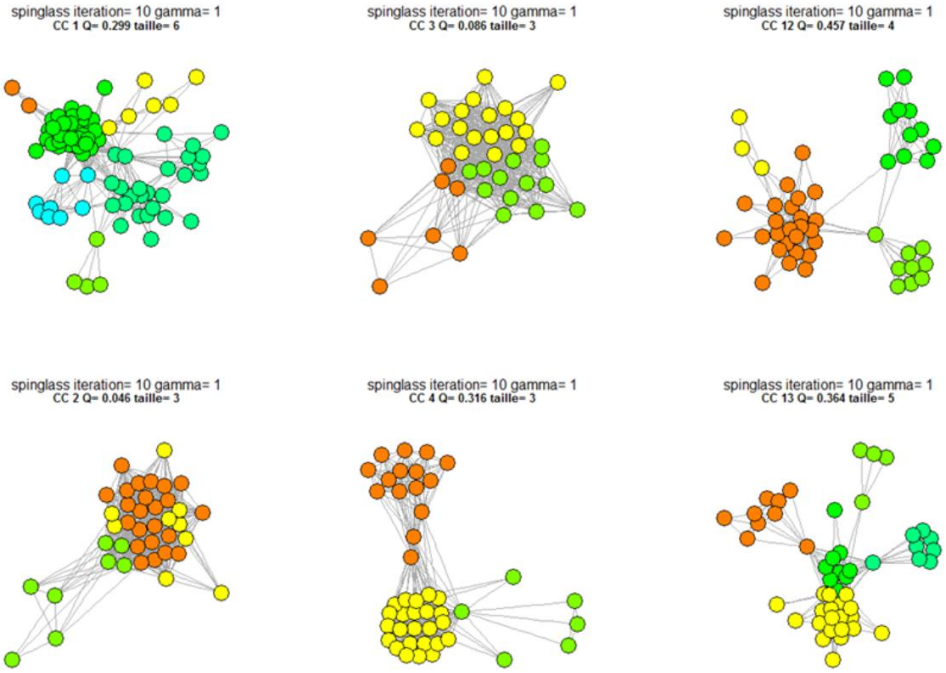
Heatmap des CC avec la méthode de partitionnement de Ward et la distance du Phi de Pearson

BETWEENESS



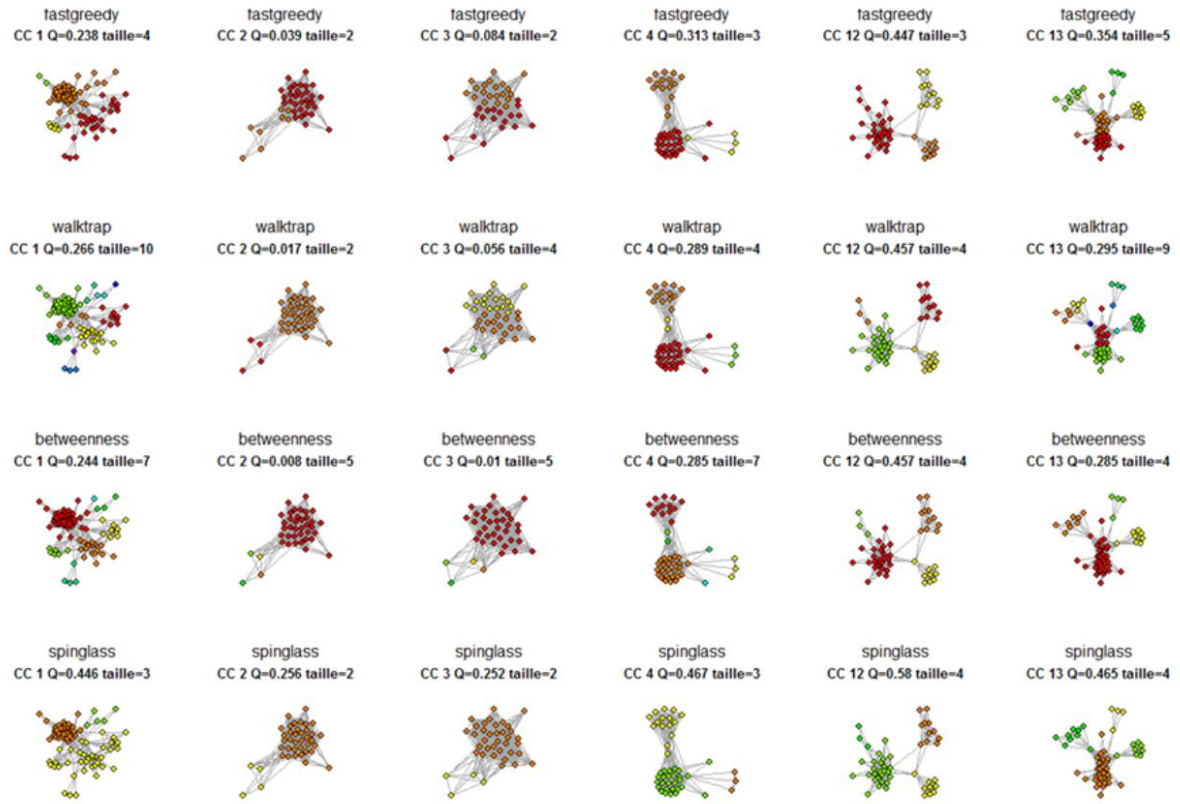
Découpage en communautés des CC (taux de paralogies > 0)avec la méthode betweeness

SPINGLASS



Découpage en communautés des CC (taux de paralogies > 0)avec la méthode spinglass

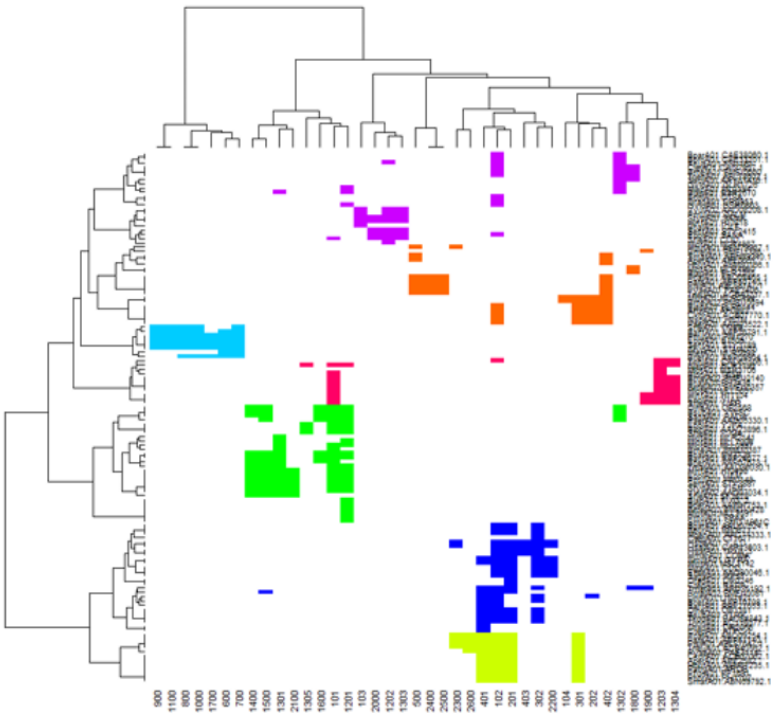
COMPARAISON DES METHODES



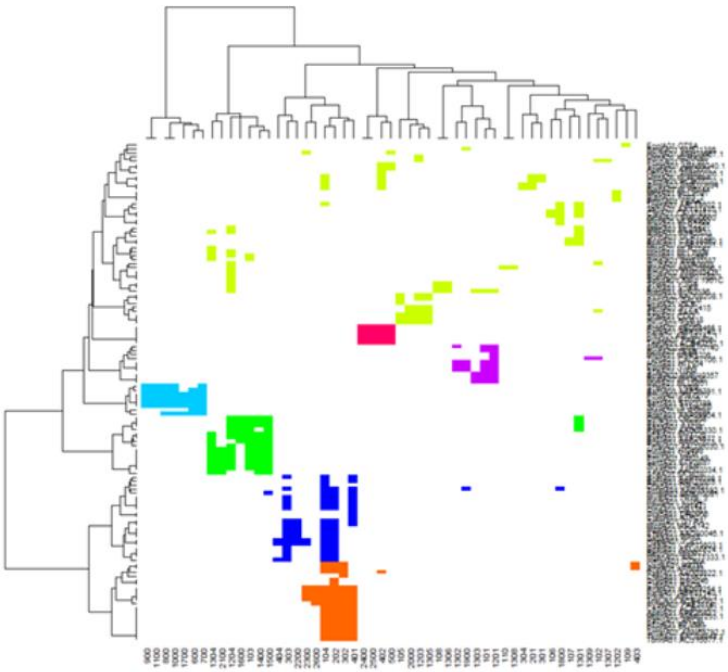
Comparaison des découpages en communautés des CC (taux de paralogies > 0)pour toutes les méthodes

Les partitions obtenues avec les différents algorithmes ne peuvent être interprétées que de manière qualitative. Les grosses communautés sont réalisées sensiblement de la même manière quelque soit la méthode. L'algorithme fastgreedy parait un peu limité dans la décomposition en communautés par rapport aux autres méthodes et à ce que l'on attend de ce découpage car il sur-découpe certaine communauté. De manière générale, toutes les méthodes ont du mal à classer les sommets se trouvant aux intersections de communautés ainsi que les sommets peu connectés sauf peut être pour spinglass avec un gamma de 0.75 qui rend des résultats intéressants (au dépend d'un temps de calcul non négligeable). Ces algorithmes sont des outils d'aides à la décision et à la recherche de gènes orthologues, mais à ce niveau de traitement, il est impossible de conclure sur leur efficacité absolue.

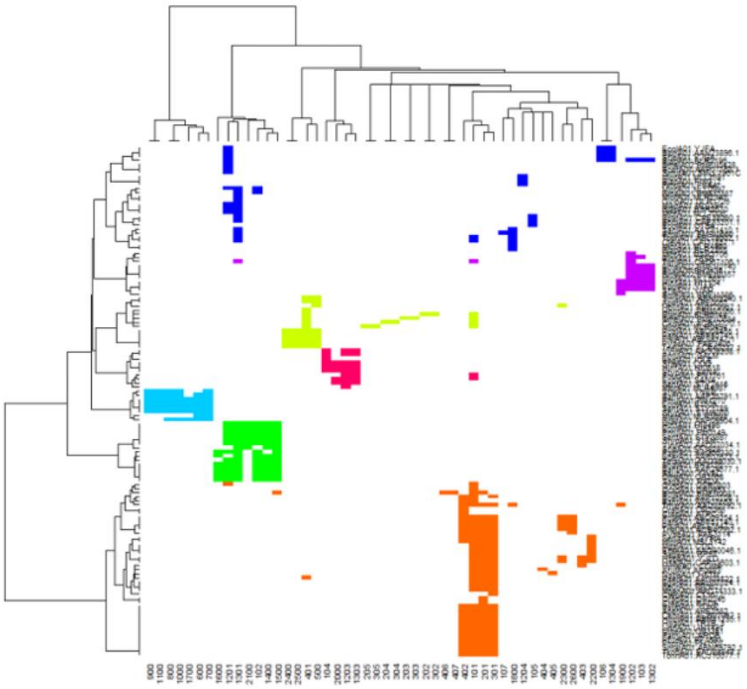
HEATMAP



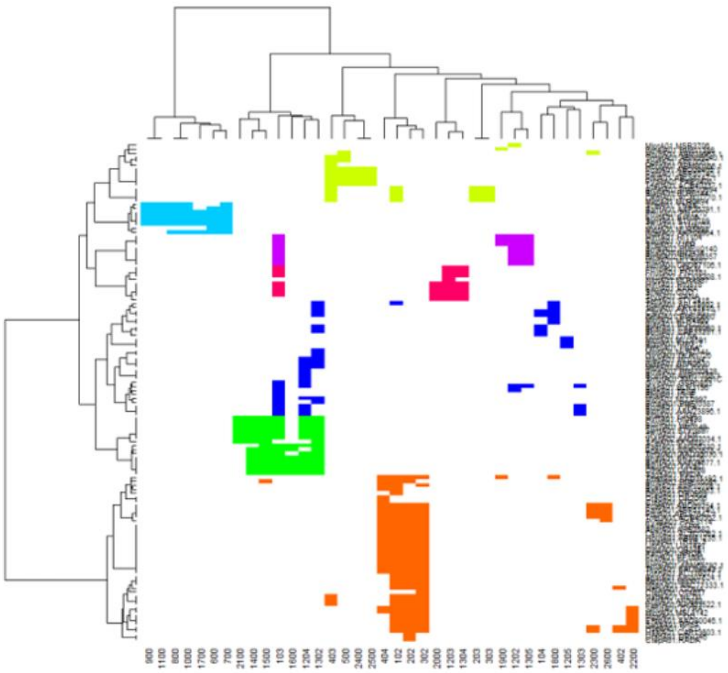
Heatmap des CC avec la méthode de partitionnement de Ward et la distance de Jaccard(fastgreedy)



Heatmap des CC avec la méthode de partitionnement de Ward et la distance de Jaccard (walktrap)



Heatmap des CC avec la méthode de partitionnement de Ward et la distance de Jaccard (betweeness)



Heatmap des CC avec la méthode de partitionnement de Ward et la distance de Jaccard (spinglass)

Visuellement, on peut voir que les méthodes spinglass et betweeness séparent mieux nos groupes mais qu'il reste encore des gènes dont on ne peut affirmer l'appartenance à un groupe.

CROISEMENT DES PARTITIONS