

TP 1 IGRAPH/R - Introduction  
*Rushed KANAWATI*

Arthur JEANNIN      Florent CLAPIÉ

21 septembre 2015

# Table des matières

Introduction . . . . .	2
Explication du TP . . . . .	2
Question 1 . . . . .	2
Question 2 . . . . .	3
Question 3 . . . . .	4
Question 4 . . . . .	5
Question 5 . . . . .	6
Conclusion . . . . .	9

# Introduction

L'objectif de ce TP est de se familiariser avec R et l'outil igraph : un outil d'analyse et de visualisation de graphes.

Ce tp est disponible sur github : <https://github.com/florent1933/tp1-ars>

Pour faire fonctionner le code, il suffit de définir le chemin d'accès vers le dossier du code.

## Explication du TP

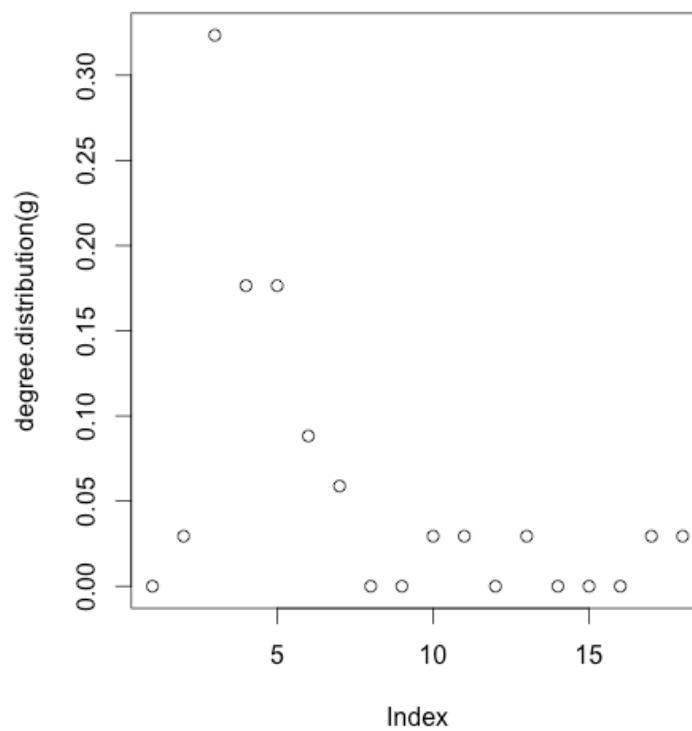
### Question 1

Nous avons écrit une fonction `topology` qui prend en paramètre un graphe `G`. La fonction résume les principales caractéristiques topologiques d'un graphe :

- `vcount(g)` : retourne le nombre de nœuds dans `g`
- `ecount(g)` : retourne le nombre de liens dans `g`
- `graph.density(g)` : donne la densité du graphe `g`
- `diameter(g)` : retourne le diamètre du graphe `g`
- `degree(g)` : retourne le degrés de chaque nœud dans `g`
- `degree.distribution(g)` : calcule la distribution de degrés de `g`
- `transitivity(g)` : calcule le coefficient de clustering du graphe `g`
- `average.path.length(g)` : retourne la moyenne de tous les plus courts chemins dans le graphe.
- `shortest.paths(g)` : retourne une matrice qui donne les longueurs de plus courts chemins entre chaque couple de nœuds.
- `betweenness(g)` : calcule la centralité d'intermédiation
- `closeness(g)` : calcule la centralité de proximité.
- `is.connected(g)` : retourne `TRUE` si le graphe est connexe.
- `clusters(g)` : retourne une liste des composantes connexes dans le graphe

## Question 2

Pour afficher la distribution de degrés d'un graphe, nous avons écrit une fonction `distribution` qui prend en paramètre un graphe `G` et affiche la distribution de degrés d'un graphe.

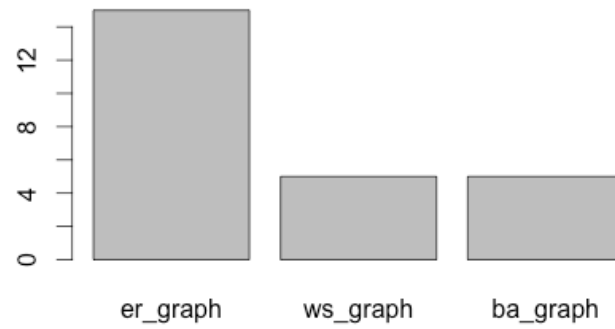


- Distribution de Karaté

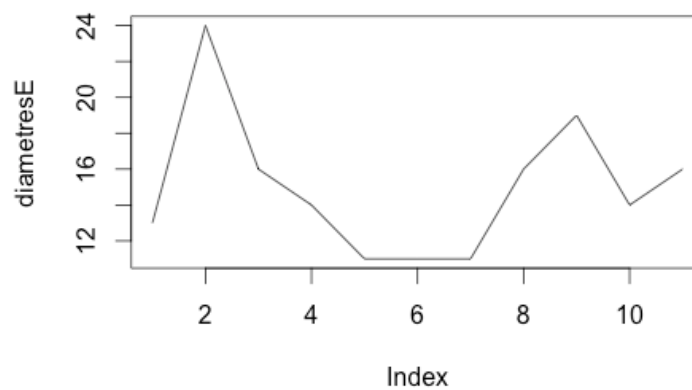
-

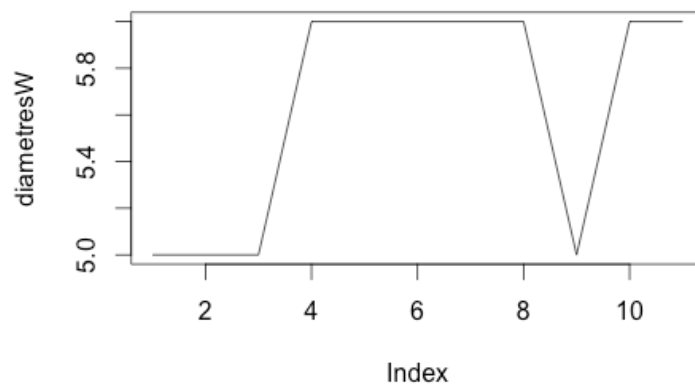
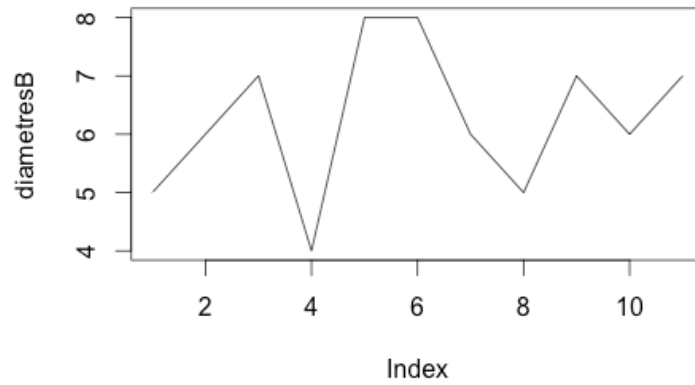
### Question 3

Comparaison des diamètres entre les différents algos



Pour étudier les variations de diamètres et de transitivité en fonction du nombre de noeuds, nous avons tracé un graphe avec des noeuds qui vont de 50 à 100 avec un pas de 5 pour chaque algo.

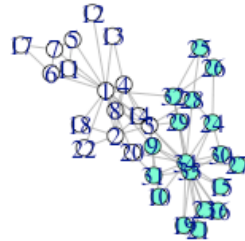




## Question 4

Pour la coloration des graphes, nous avons colorié par rapport à la valeur des noeuds.

Nous avons utilisé un vecteur de couleur contenant le nom de nos couleurs.



### Communauté karaté

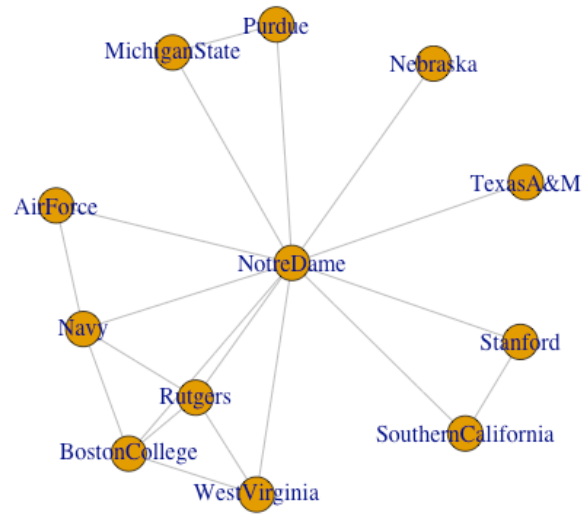
Pour voir les autres communautés, il faut lancer le programme car il y a trop de noeuds.

### Question 5

Il y a 12 graphes, merci de vous référer au code.

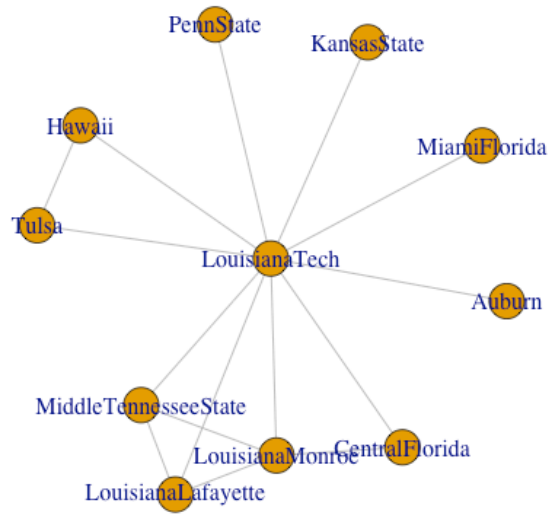
Exemple pour le graphe **football**

### Centralité de l'intermédiarité

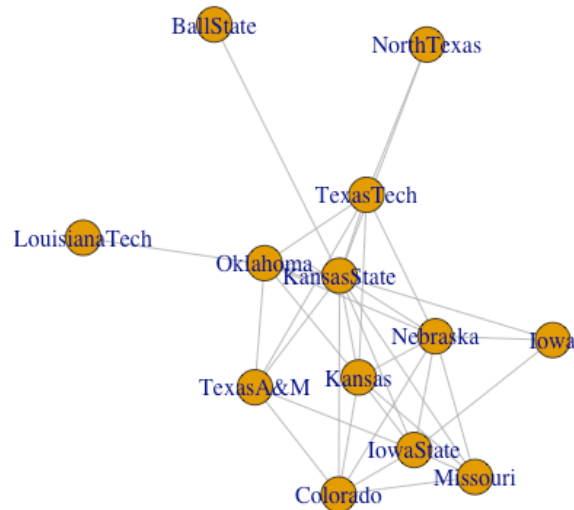




### Centralité de proximité



## Degrés



## Conclusion

Ce TP fut très intéressant, il nous a permis de nous familiariser avec R bien que nous ayons eu des difficultés.

Nous n'avons pas trouvé comment faire des commentaires multilignes, concaténer une chaîne + variable de façon correcte :

```
cat("nombre de noeuds : ", vcount(g), "\n")
```

Il nous a aussi permis d'apprendre à utiliser igraph