# Social Network Analysis - Heroes

Alexis Buckens

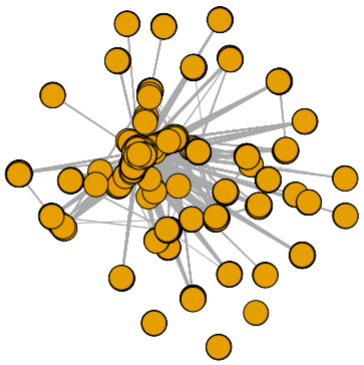
March 16, 2018

## 1 Dataset

Le jeu de donnée provient de Kaggle[1]. Une description complète se trouve sur le site[2]

L'analyse porte uniquement sur le jeu de donnée hero-edge contenant un graphe reprenant les héros apparaissant dans le même comics. Comme certain vertex possèdent plusieurs edges entre eux, il faut d'abord transformer le graphe de manière à ce qu'un seul edge se trouve entre une paire de vertex. Pour cela, on peut ajuster le poids de chaque edge en fonction du nombre de liens se trouvant à l'origine entre deux vertex. Cela peut être fait à l'aide de la fonction "simplify". Le graphe comprend 167207 edges et 6426 vertex. On a désormais un graphe simple, non-dirigé, des poids sont attribués aux edges et le graphe n'est pas connecté. On a donc des portions de graphes qui ne sont pas reliées aux autres. On a cependant un sous-graphe principal, connecté, de 6408 vertex. Les 3 autres sous-graphes connectés contiennent au total 18 vertex. On peut aller plus en profondeur et regarder si on a des points d'articulation, c'est-à-dire des vertex qui déconnectent le graphe si on les supprime. On a 53 points d'articulation dans le graphe, ce qui n'est pas un nombre très élevé.

On peut finalement faire un graphique du réseau, mais le fait que celui-ci comprenne un grand nombre de vertex rend le graphe peu lisible. Le layout layout.drl permet partiellement de circonvenir à ce problème, mais le graphe reste globalement peu lisible :



[1] https://www.kaggle.com/csanhueza/the-marvel-universe-social-network

[2] Marvel Comics, originally called Timely Comics Inc., has been publishing comic books for several decades. "The Golden Age of Comics" name that was given due to the popularity of the books during the first years, was later followed by a period of decline of interest in superhero stories due to World War ref. In 1961, Marvel relaunched its superhero comic books publishing line. This new era started what has been known as the Marvel Age of Comics. Characters created during this period such as Spider-Man, the Hulk, the Fantastic Four, and the X-Men, together with those created during the Golden Age such as Captain America, are known worldwide and have become cultural icons during the last decades. Later, Marvel's characters popularity has been revitalized even more due to the release of several recent movies which recreate the comic books using spectacular modern special effects. Nowadays, it is possible to access the content of the comic books via a digital platform created by Marvel, where it is possible to subscribe monthly or yearly to get access to the comics. More information about the Marvel Universe can be found here. Content

The dataset contains heroes and comics, and the relationship between them. The dataset is divided into three files:

nodes.csv: Contains two columns (node, type), indicating the name and the type (comic, hero) of the nodes. edges.csv: Contains two columns (hero, comic), indicating in which comics the heroes appear. hero-edge.csv: Contains the network of heroes which appear together in the comics. This file was originally taken from http://syntagmatic.github.io/exposedata/marvel/

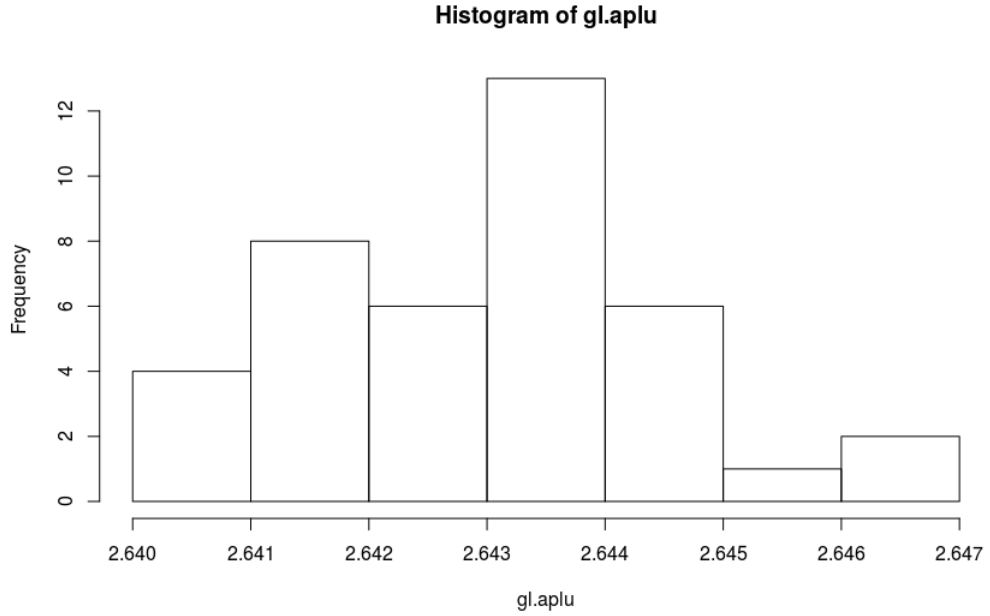
Past Research (Acknowledgements)

The Marvel Comics character collaboration graph was originally constructed by Cesc Rosselló, Ricardo Alberich, and Joe Miro from the University of the Balearic Islands. They compare the characteristics of this universe to real-world collaboration networks, such as the Hollywood network, or the one created by scientists who work together in producing research papers. Their original sources can be found here. With this dataset, the authors published the paper titled: "Marvel Universe looks almost like a real social network".

## 2 Caractéristiques générales

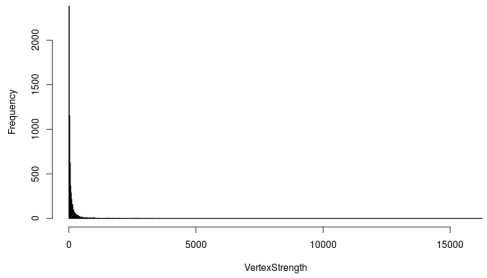
On peut commencer à s'intéresser aux caractéristiques générales du graphe. On peut commencer par regarder la densité du graphe. La fonction edge_ density donne une valeur de 0.008. Ce nombre est le rapport du nombre de edges présent dans le graphe, divisé parce le nombre maximal d'edges possible si chaque vertice du graphe était connecté à tout les autres vertex.

Une autre mesure possible et le chemin moyen entre deux vertex. La valeur pour le graphe peut être obtenue en utilisant la fonction mean_ distance. La valeur est de 2.638. Pour savoir si cette valeur est élevée ou basse, on peut simuler d'autres graphes ayant le même nombre de vertex et la même densité que celui-ci, et voir si notre graphe semble avoir une valeur similaire aux graphes simulés. Bien qu'il soit préférable de simuler un grand nombre de graphes pour plus de puissance, seuls 40 graphes peuvent etre simulé sur cet ordinateur. On peut afficher un histogramme des valeurs obtenues pour les différentes simulations comparées à valeur de notre graphe.

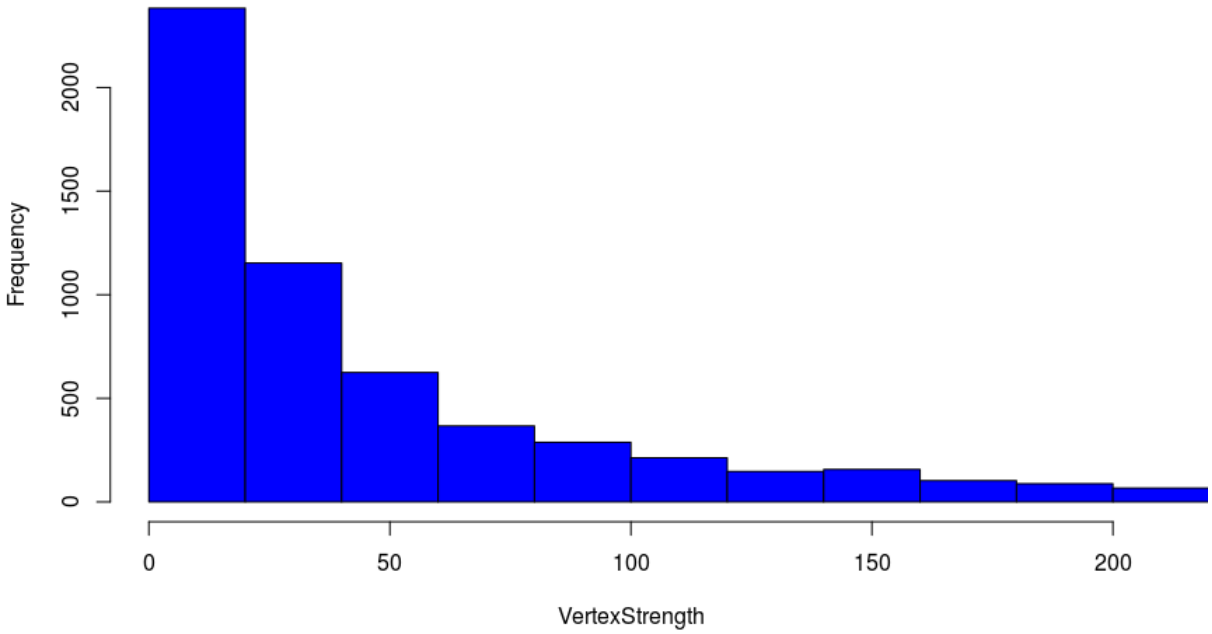



Notre graphique semble définitivement avoir un chemin moyen plus court que la grande majorité des graphes. Il est donc mieux connecté que la moyenne.

On peut également voir si le graphe est bien connecté en fonction du nombre de edges moyen connecté à chaque vertice. Pour aller plus loin, on peut même voir la répartition des vertices en fonction du nombre d'edges connecté. Le nombre d'edge connecté à un vertice est le degré d'un vertice. Le degré peut être pondéré par le poids de chaque vertex. Pour visualiser cette répartition, on peut dresser un histogramme reprenant le nombre de vertex ayant un certain degré.

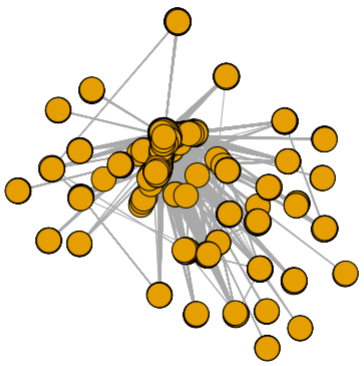



On peut voir que si certains vertex ont un grand nombre de connections, la plupart d'entre eux se trouvent dans la partie gauche de l'histogramme, sur laquelle nous pouvons nous focaliser.




## 3 Centralités

On peut s'intéresser à la centralité de certains vertex. Cela peut nous permettre de voir quels super héros sont les mieux connecté, les plus centraux, et par conséquent les plus populaires. Différentes méthodes permettent de mesurer la centralité des vertex. Le plus simple consiste à regarder le degré de chaque vertice. En utilisant cette méthode, le super héro le plus populaire semble être CAPTAIN AMERICA. Une autre mesure est utilisée par betweeness. Il s'agit du nombre de fois qu'un vertice se trouve dans le chemin aléatoire entre deux vertex. Si le vertice est central, il se trouvera régulièrement entre deux vertex. Cette seconde méthode est plus lente et donne un résultat légèrement différent, le héro le plus populaire étant cette fois SPIDER-MAN/PETER PAR. Finalement, une troisième méthode basée sur les vecteur propres donne le même résultat que la méthode basée sur le degré. En regardant les 5 vertex les plus centraux pour les 3 méthodes, SPIDER-MAN/PETER PAR, CAPTAIN AMERICA et IRON MAN/TONY STARK semblent être les plus centraux. Pour visualiser cette centralité, on peut faire un ego graph autour de Captain America, mais comme le graphe est un grand graphe, celui-ci reste toujours assez peu lisible.



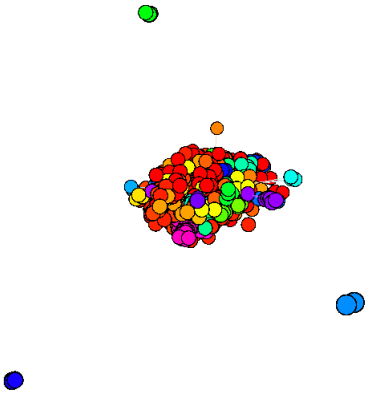
Une meilleure visualisation de la centralité peut être réalisée via get.adjacency et gplot.target, mais l'ordinateur ne semble pas être suffisamment performant.

# 4 Cliques et communautés

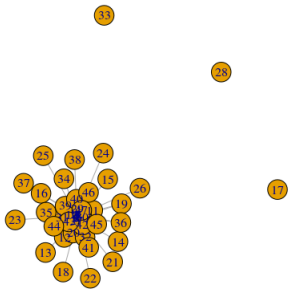
On peut désormais passer à l'analyse de portions du graphe. Les cliques sont des sous-graphes dans lesquels chaque vertice est connecté à tout les autres vertex. Évidemment, il s'agit en général de petit sous-graphes. Malheureusement, l'ordinateur n'est pas assez performant que pour exécuter la fonction "cliques".

Le graphe peut cependant est diviser en communauté d'héros fortement connectés entre eux, et peu connectés avec les autres communautés. De manière similaire aux algorithme de clustering, deux types de méthodes principales existent : celles qui partent de chaque vertex et réunissent les vertex ou groupes de vertex les plus proches, et celles qui partent du graphe et le divisent systématiquement en sous groupes déconnectés. une fonction rapide du premier type est "fastgreedy.community". Pour comparaison, une autre fonction sera utilisée, basé sur une marche aléatoire de 5 étapes.

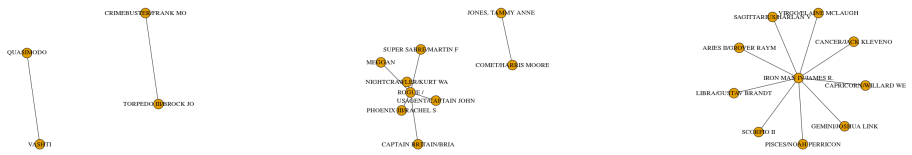
La fonction repère 47 communautés, dont les 3 plus grosses comprennent 4242 vertex, c'est-a-dire 66 % du graphe. On peut représenter le graphe avec des vertex de couleur différente en fonction de la communauté à laquelle ils appartiennent.



On peut aussi représenter un nouveau graphe des communautés en représentant chaque communauté sous forme d'une vertice.



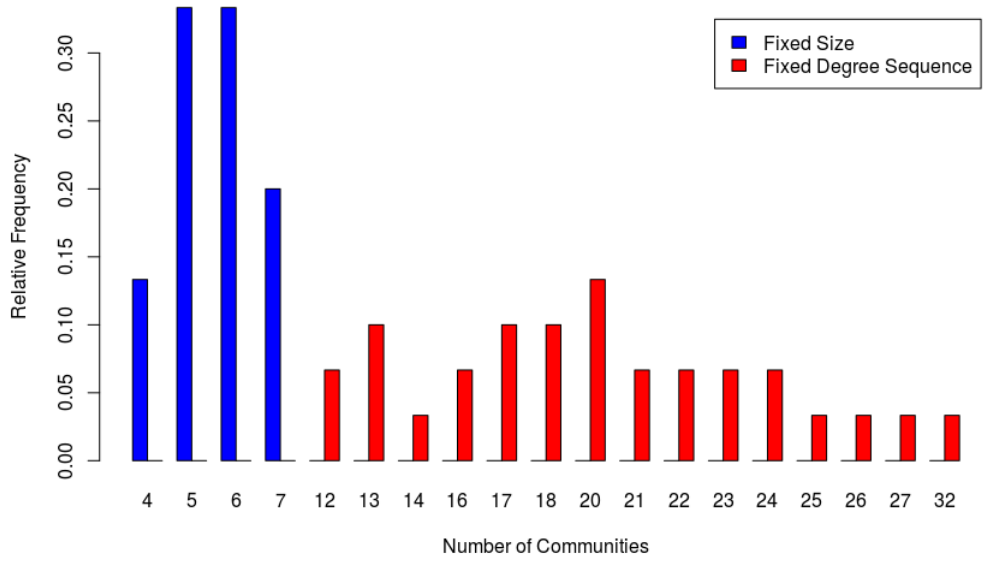

Les 3 communautés disconnectées sont les communautés 17, 28, et 33. On peut produire un graphe de chacune d'entre elles.




Les communautés disjointes ne correspondent pas exactement aux sous-graphes disjoints.

On peut simuler une série d'autres graphes ayant des caractéristiques similaires et voir si le nombre de communautés détectées par le même algorithme est similaire au nombre de communautés détectées pour notre graphe ou si notre graphe possède un nombre anormalement élevé ou anormalement bas de communautés. 30 graphes de la même taille que notre graphe et 30 autres possédant des degrés similaires sont simulés. On peut tracer un

histogramme reprenant la quantité de graphe simulés possédant un certain nombre de communautés selon les deux méthodes, et comparé avec les 47 communautés découvertes dans notre graphe.




Les 47 communautés de notre graphe semblent donc être surprenantes!