Graphs in Social Networks

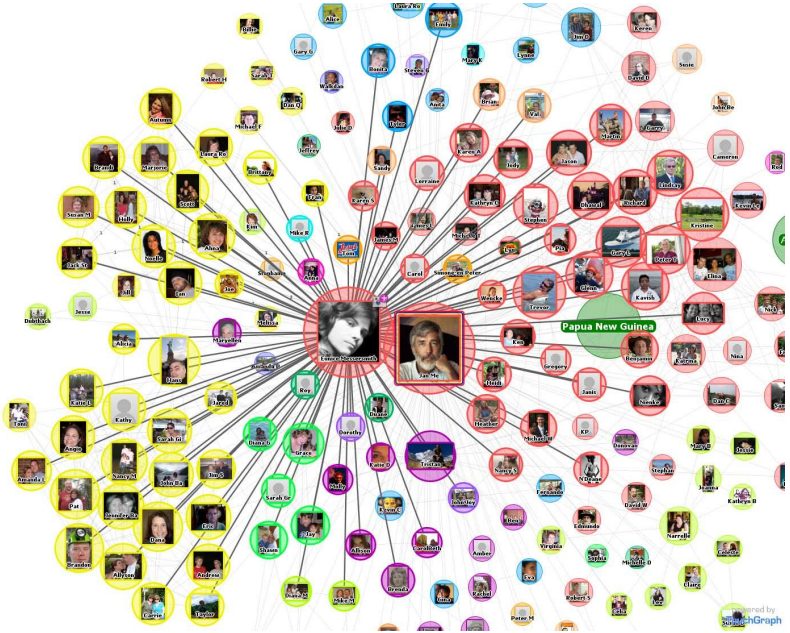
EN 500.111 Week 4

Outline

- ➔ What is a social network graph?
- ➔ Centrality
- ➔ Clustering

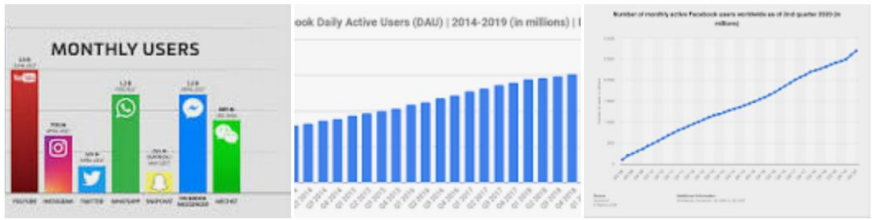
Social Network Graphs

- ➔ Nodes are individual people
- ➔ Edges are some kind of connection between them - can be directed or undirected
- ➔ For example, websites with friend/following relationships or management hierarchies in a workplace
- ➔ Other examples?



(http://www.messersmith.name/wordpress/tag/social-networking)

Social Network - Enormous Datasets



View all

With over 2.7 billion monthly active **users** as of the second quarter of 2020, **Facebook** is the biggest social network worldwide. In the third quarter of 2012, the number of active **Facebook users** surpassed one billion, making it the first social network ever to do so.

Aug 10, 2020

Website category: Social networking service

www.statista.com › statistics › number-of-monthly-active-...

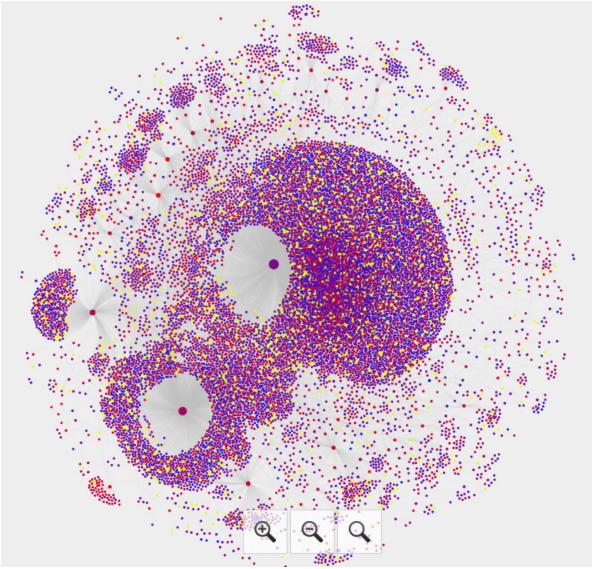
• Facebook: active users worldwide | Statista

If each user has an average of 100 friends, that would take up:

3 billion users
...times 100 friends
...times 4 bytes
= 1.2 TB (most than most computers today hold)

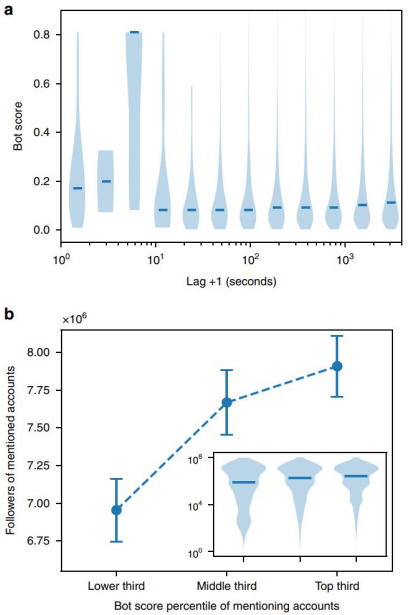
And that's just to store friendships, let alone any interactions or edge weights

Example: Using Graphs to Study Bot Activity



Graph shows over 30,000 accounts which Tweeted a fabricated headline in the days leading up to the 2016 election colored blue (likely human), red (likely bot), or yellow (deleted/suspended): “SPIRIT COOKING”: CLINTON CAMPAIGN CHAIRMAN PRACTICES BIZARRE OCCULT RITUAL

By looking at timestamps and connections to other accounts, researchers could analyze strategies used for bot-assisted misinformation sharing.

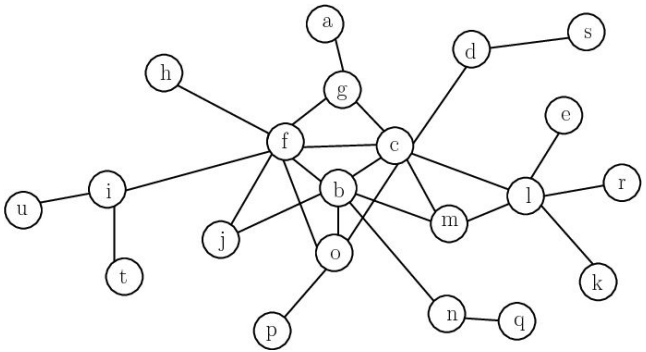


Shao et. al. The spread of low-credibility content by social bots. Nature Communications, 2018.

What kind of information in a social network might we care about?

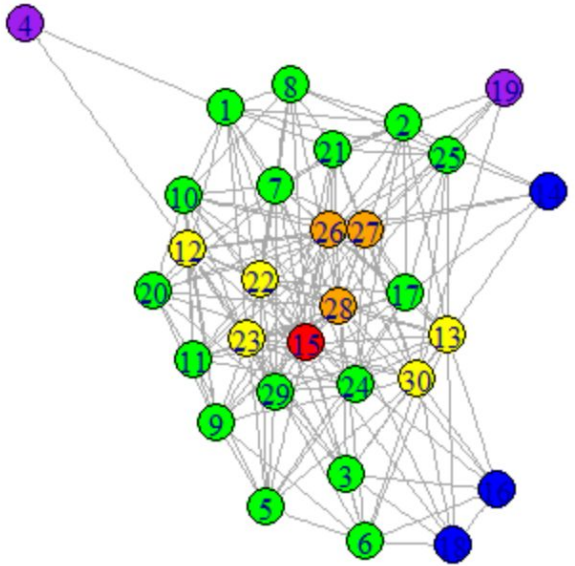
Finding important people

- ➔ Imagine you have a graph of social media users, with edges between people who are friends
- ➔ If you're a company and can pay one person to post about your product, who would you choose and why?
- ➔ What information would you want besides friendships when making this decision?



Degree Centrality

- ➔ One of many ways to measure how “central” a node is in a network
- ➔ Defined as a value proportional to the degree of each node (also potentially normalized based on the network size)
- ➔ Great for finding people who are highly visible
- ➔ Can you think of other measures we may want to highlight important people?

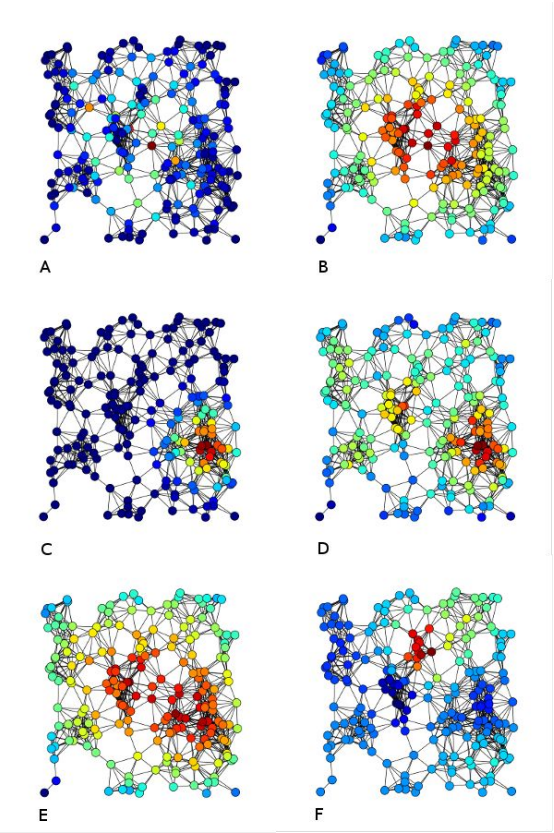


(stackoverflow.com)

Other Measures of Centrality

A lot of different ways to measure it:

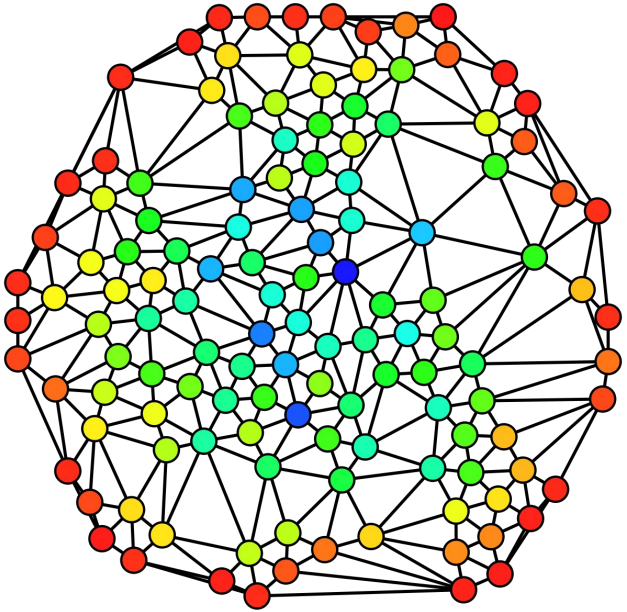
A. Betweenness Centrality
B. Closeness Centrality
C. Eigenvector Centrality
D. Degree Centrality
E. Harmonic Centrality
F. Katz Centrality



(Wikipedia)

Betweenness Centrality

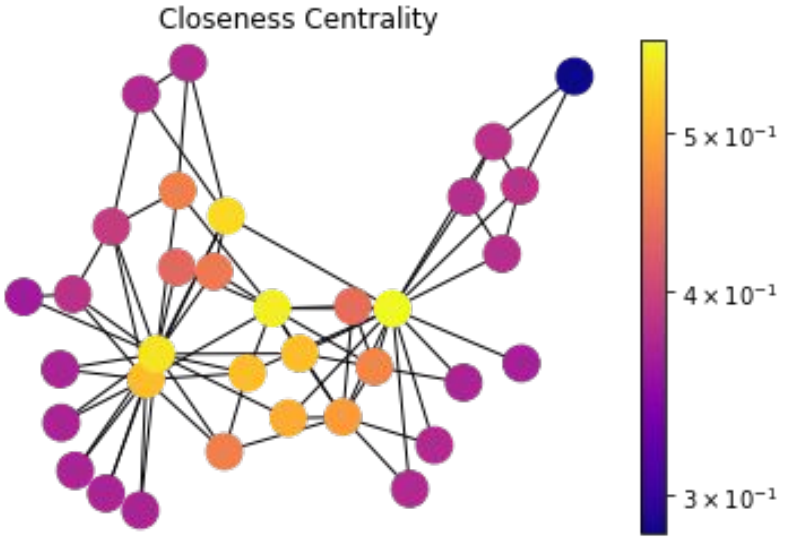
- ➔ Measures centrality as a function of **how many shortest paths in the graph each vertex is a part of**
- ➔ Exact formula for betweenness_centrality(v)
  - ◆ For all pairs of vertices (a, b) besides v, compute all shortest paths from a to b and calculate the proportion of them which go through v.
  - ◆ Add up these proportions for all pairs
- ➔ Quite slow to determine because it requires finding all-pairs shortest paths



(Wikipedia)

Closeness Centrality

➔ Measures the centrality of each vertex based on the **sum of the lengths of the shortest paths to all other vertices**

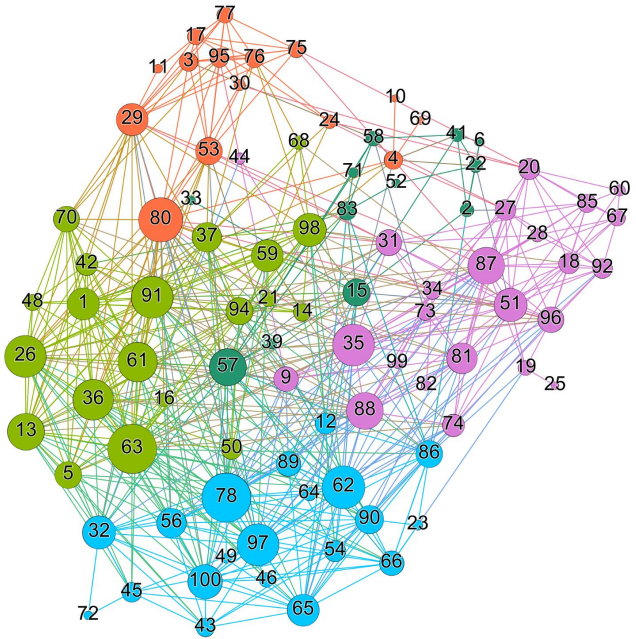


Exercises

1. Construct examples of graphs with the following. If you're having trouble coming up with example, try to describe what structures of graphs would lead to high vs. low values for the different measures:
    a. A node with high degree centrality, but low betweenness and closeness centrality
    b. A node with high betweenness centrality, but low degree and closeness centrality
    c. A node with high closeness centrality, but low degree and betweenness centrality
2. In what real-world situations might each centrality measure be useful?

Community Detection

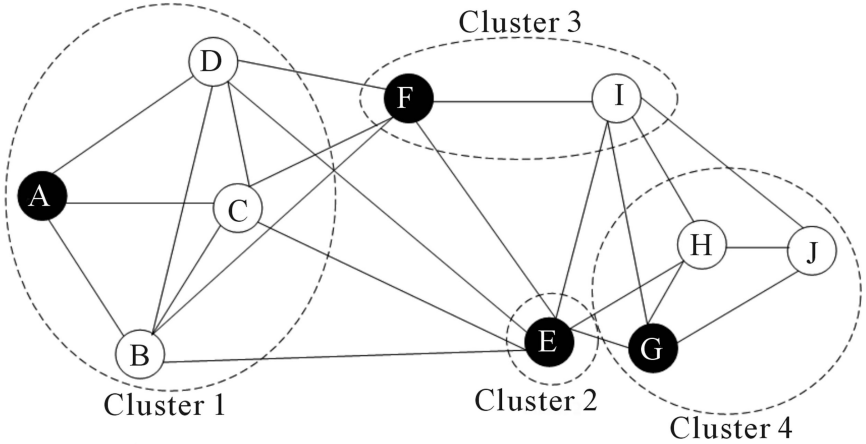
- ➔ Many social networks have underlying communities - groups who interact often with each other a lot and not very much with others outside the group
- ➔ For example, my social media friends may be divided up into these subgroups:
  - Co-workers
  - Neighbors
  - Family
  - College friends
  - High school friends



(https://medium.com/@figarrikeisha/community-detection-of-my-network-d0977bfdee4e)

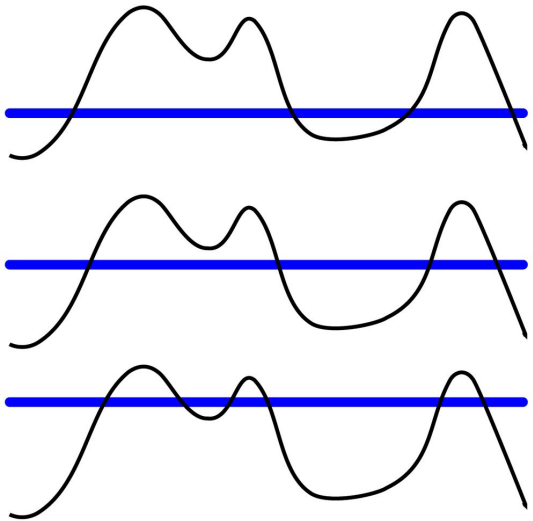
How to detect communities?

- ➔ Much harder than simply detecting connected components because we need more fine-grained measures of connectedness
- ➔ Any ideas for clustering?



https://chanzuckerberg.github.io/scRNA-python-workshop/analysis/04-clustering.html

One idea: Turn it into finding connected components



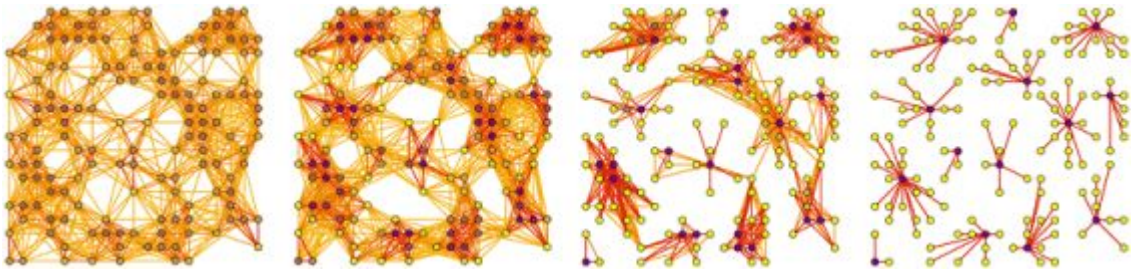
(https://www.oreilly.com/library/view/social-network-analysis/9781449311377/ch04.html)

- ➔ Think of nodes as a landscape (where height is degree or other centrality measure)
- ➔ Add “tide” by removing below a chosen degree threshold so that graph becomes disconnected
- ➔ Let each component be its own group
- ➔ Optionally add the removed nodes back into the clusters

What benefits/downsides would this approach have?

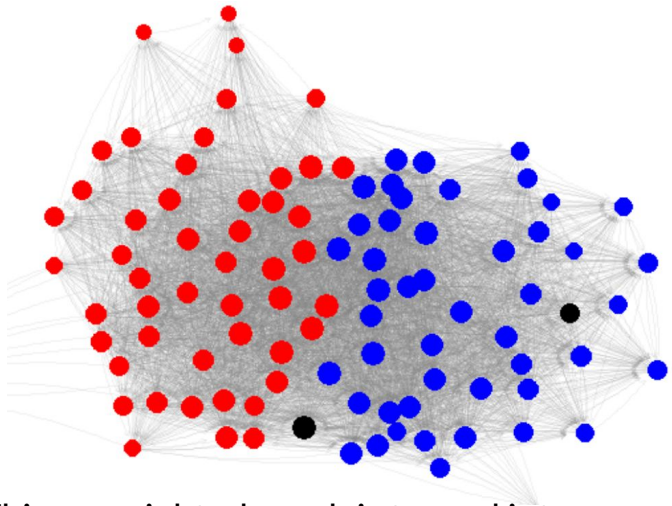
More Sophisticated Clustering

- Clustering generally fall into two main categories
  - **Agglomerative clustering**: start with all of the nodes in their own cluster and repeatedly merge groups together
  - **Divisive clustering**: start with the whole graph and gradually remove edges to break the graph into clusters

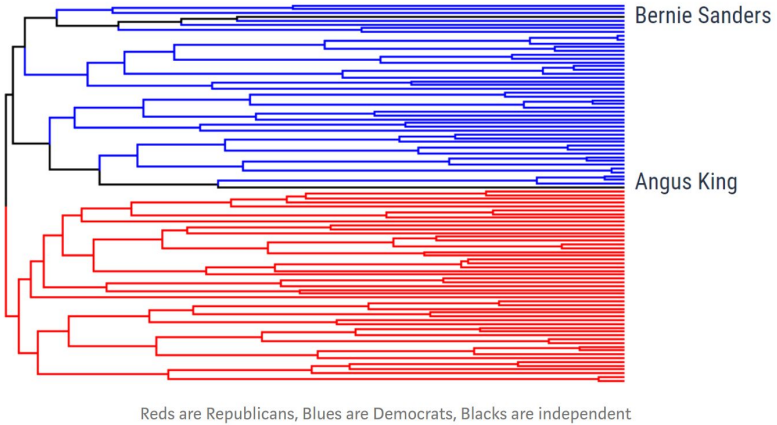


Markov Cluster Algorithm (https://micans.org/mcl/)

Hierarchical Clustering



This unweighted graph is turned into a weighted graph of proximity by a random walk algorithm



Graph of which Senators follow which other Senators on Twitter (https://towardsdatascience.com/hierarchical-clustering-and-its-applications-41c1ad4441a6)

Louvain Algorithm

- Defines a formula for the “modularity” of a graph given an assignment of the nodes into communities
- English version of formula:
  1. Add the weights of all within-community edges
  2. Subtract the total weight of within-community edges you’d expect to get on average in a random graph with the same number of edges
  3. Normalize to get a value in [-1, 1]
  4. Higher modularity means higher-weight edges connect nodes in the same community, which represents a better clustering

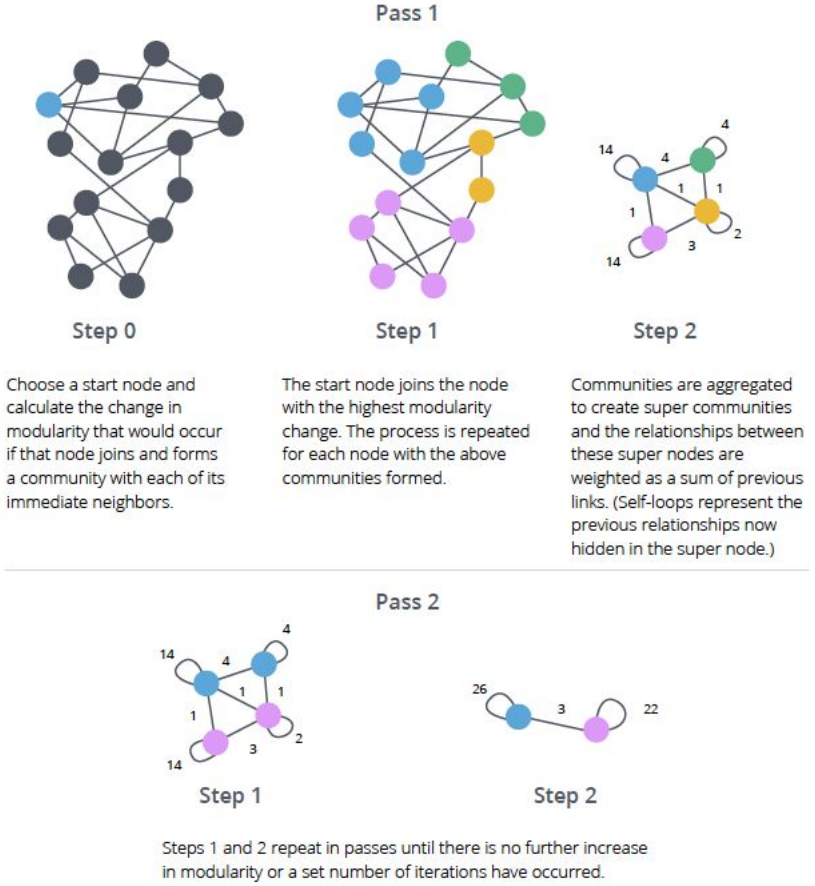
$$Q = \frac{1}{2m} \sum_{i,j} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j),$$

where

Aij is the weight of the edge between i and j.
ki is the sum of weights of the vertex attached to the vertex I, also called as degree of the node
ci is the community to which vertex i is assigned
δ(x,y) is 1 if x = y and 0 otherwise
m = (1/2)$\sum_{ij}$ Aij i.e number of links

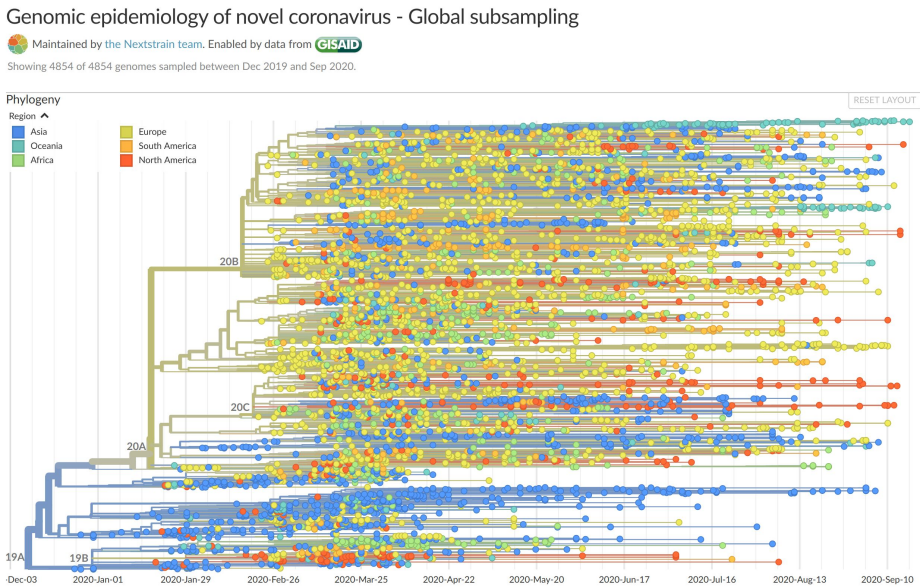
Modularity Definition

Louvain Algorithm



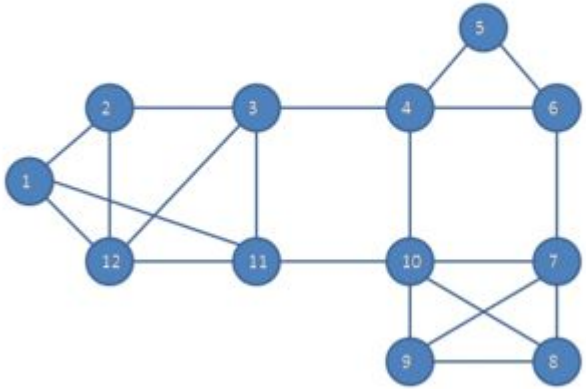
(https://neo4j.com/blog/graph-algorithms-neo4j-louvain-modularity)

Not just for social networks!



(https://nextstrain.org/ncov/global?c=region)

Divisive Clustering Example



(Wikipedia)

Highly Connected Subgraph (HCS) Clustering:

1. Check if the graph is highly connected (high enough ratio of edges to nodes)
2. If so, then stop. Otherwise, find the smallest set of edges that can be removed and disconnect the graph, and return to step 1 for each smaller graph.

Conclusion

- ➔ Graphs can be used to represent relationships between people
- ➔ Algorithms can help us to identify important people or communities within these networks
- ➔ Metrics such as centrality and clustering are loosely defined and vary a lot by application

Any questions?