GENERAL ASSEMBLY

DATA SCIENCE
10 WEEK PART TIME COURSE

Week 7 – Network Analysis
Wednesday 23rd November 2016

AGENDA

1. Anthony Tockar talk
2. Network Analysis
3. Lab
4. Discussion

WHAT IS NETWORK ANALYSIS?

Many types of real-world problems involve dependencies between observations.

For example:

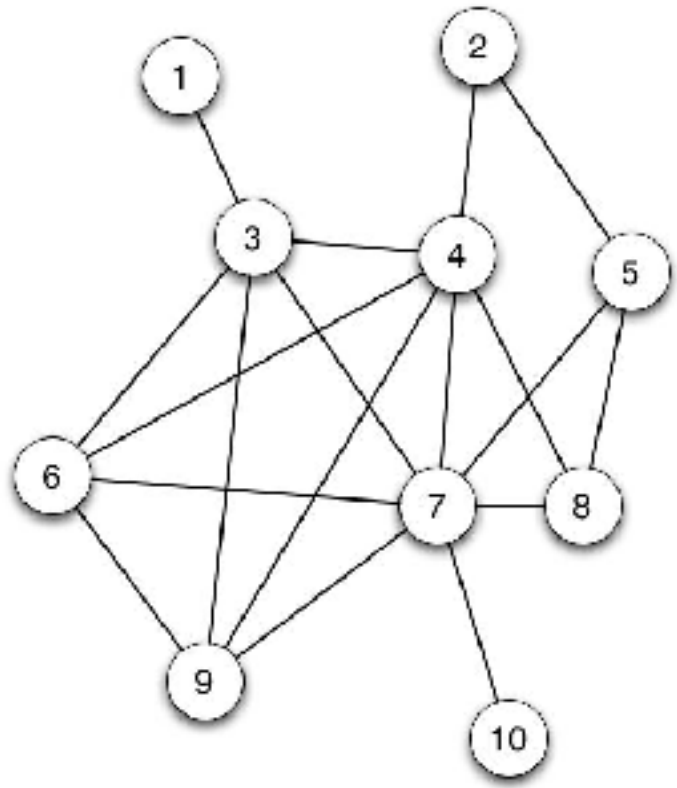
- Town planners are looking at vehicular flows through a city
- Sociologist want to understand how people influence others that they know (if at all)
- Biologists want to know how proteins regulate the actions of other proteins
- Information agencies want to discover groups of adversaries

Problems involving dependencies can often be modelled as graphs, and scientists have developed methods for answering these questions called network analysis.

A graph consists of a nodes (or vertices) and are connected by edges.

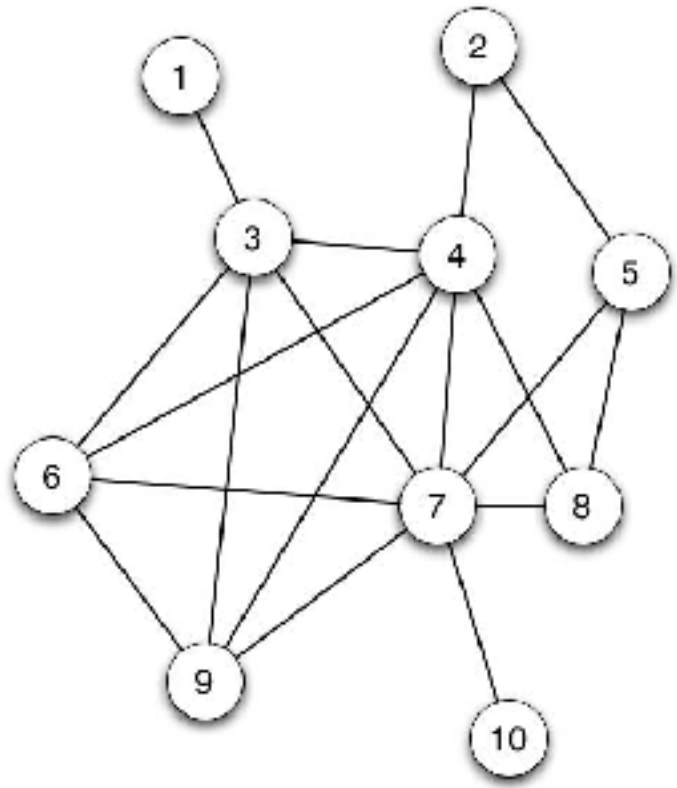
For example the nodes may represent people and the edges are there if a friendship exists.

How many nodes and edges are there?



Nodes = 10

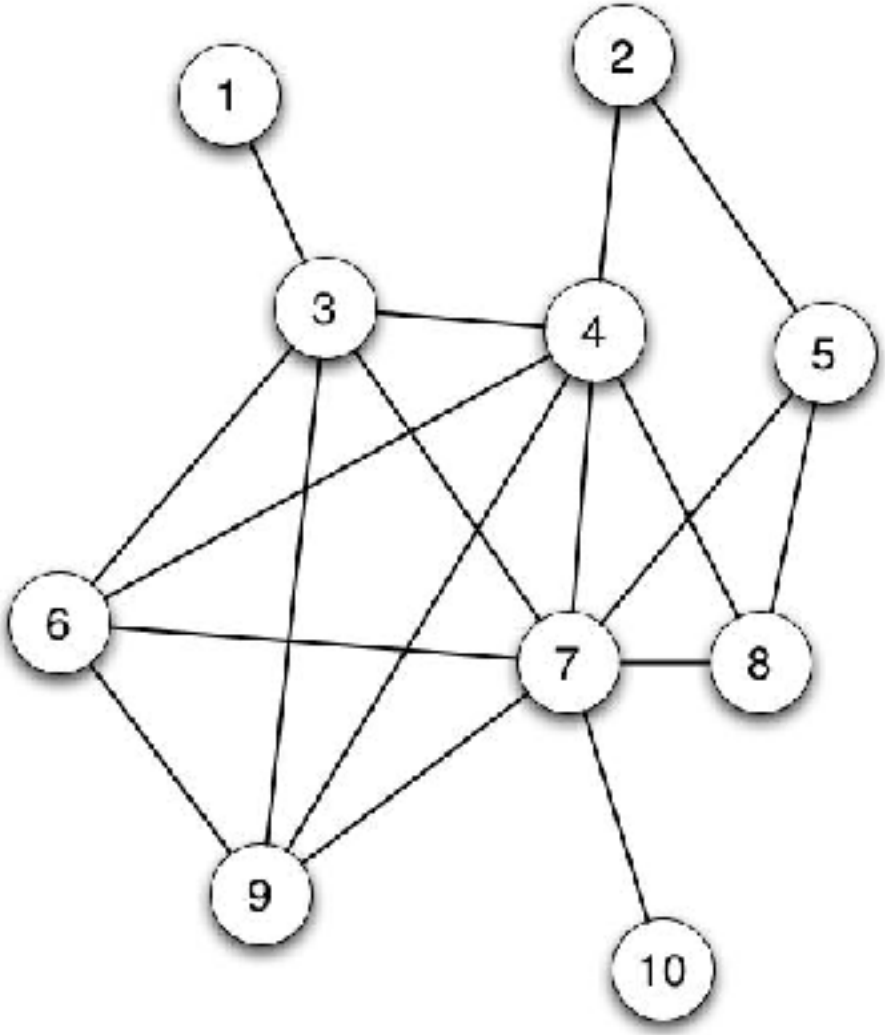
Edges = 18



How do we know who the key actors (nodes) are in our network ?

We need to calculate some measures of importance (or centrality).

What are some of the ways you might measure an individual's centrality (there are multiple measures).


1
2
3
4
5
6
7
8
9
10

- Degree Centrality - number of edges a node has
- Closeness Centrality - the reciprocal of the sum of the shortest path distances from one node to all n-1 other nodes. Since the sum of distances depends on the number of nodes in the graph, closeness is normalized by the sum of minimum possible distances n-1. Higher values of closeness indicate higher centrality
- Betweenness Centrality - the sum of the fraction of all-pairs shortest paths that pass through the node v
- Eigenvector centrality - computes the centrality for a node based on the centrality of its neighbours
- Page Rank - count the number and quality of links to a page to determine a rough estimate of how important the website is. The underlying assumption is that more important websites are likely to receive more links from other websites

```
networkx.degree_centrality(graph)

networkx.closeness_centrality(graph)

networkx.betweenness_centrality(graph)

networkx.eigenvector_centrality(graph)

networkx.pagerank(graph)

sorted(networkx.pagerank(graph).items(),
   key=(lambda x: x[1]), reverse=True)
```

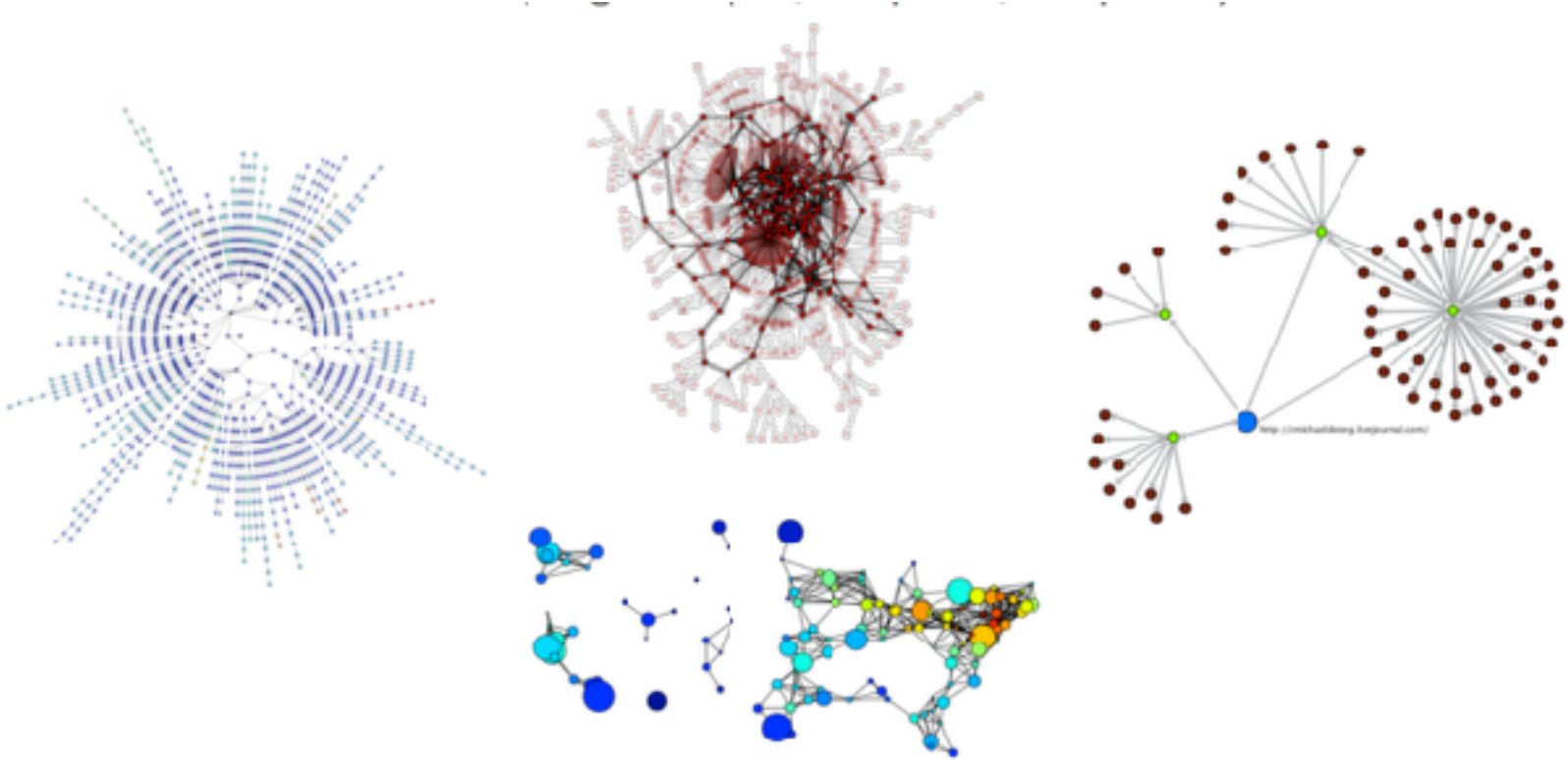
NetworkX

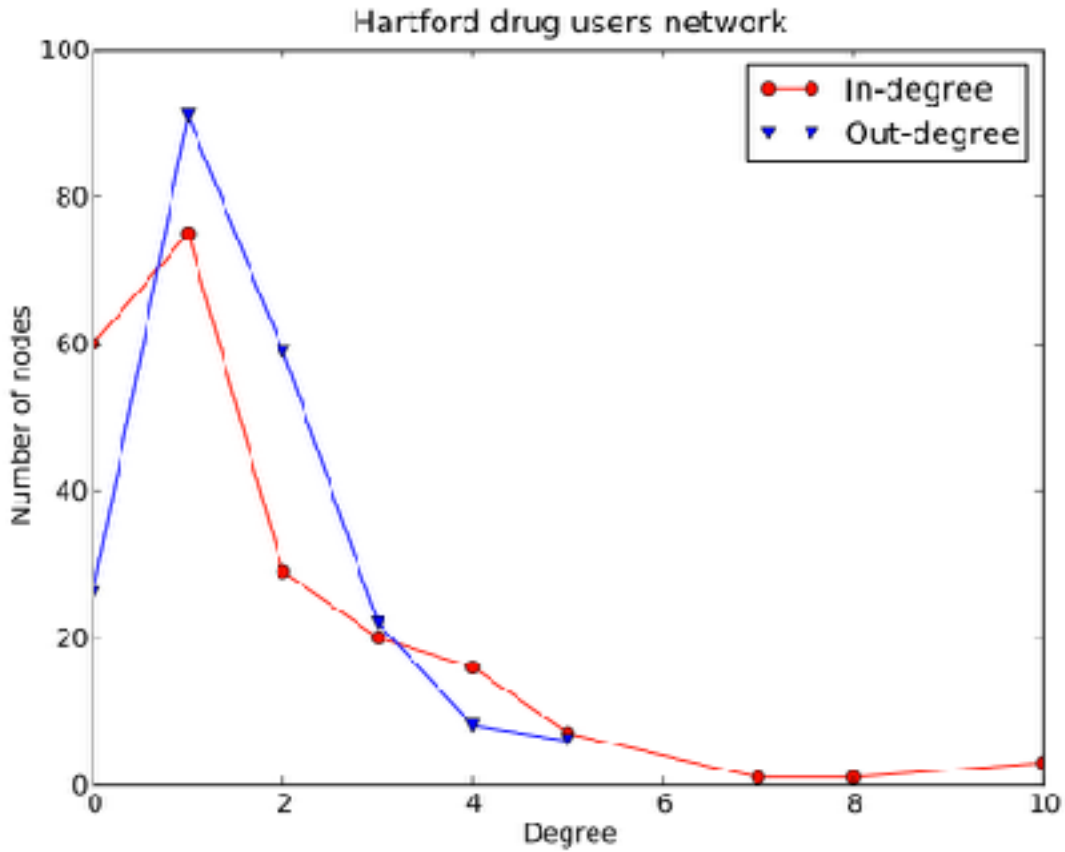
- Tool to study the structure and dynamics of social, biological, and infrastructure networks
- Ease-of-use and rapid development
- Open-source tool base that can easily grow in a multidisciplinary environment with non-expert users and developers
- An easy interface to existing code bases written in C, C++, and FORTRAN
- To painlessly slurp in relatively large nonstandard data sets

DATA SCIENCE PART TIME COURSE

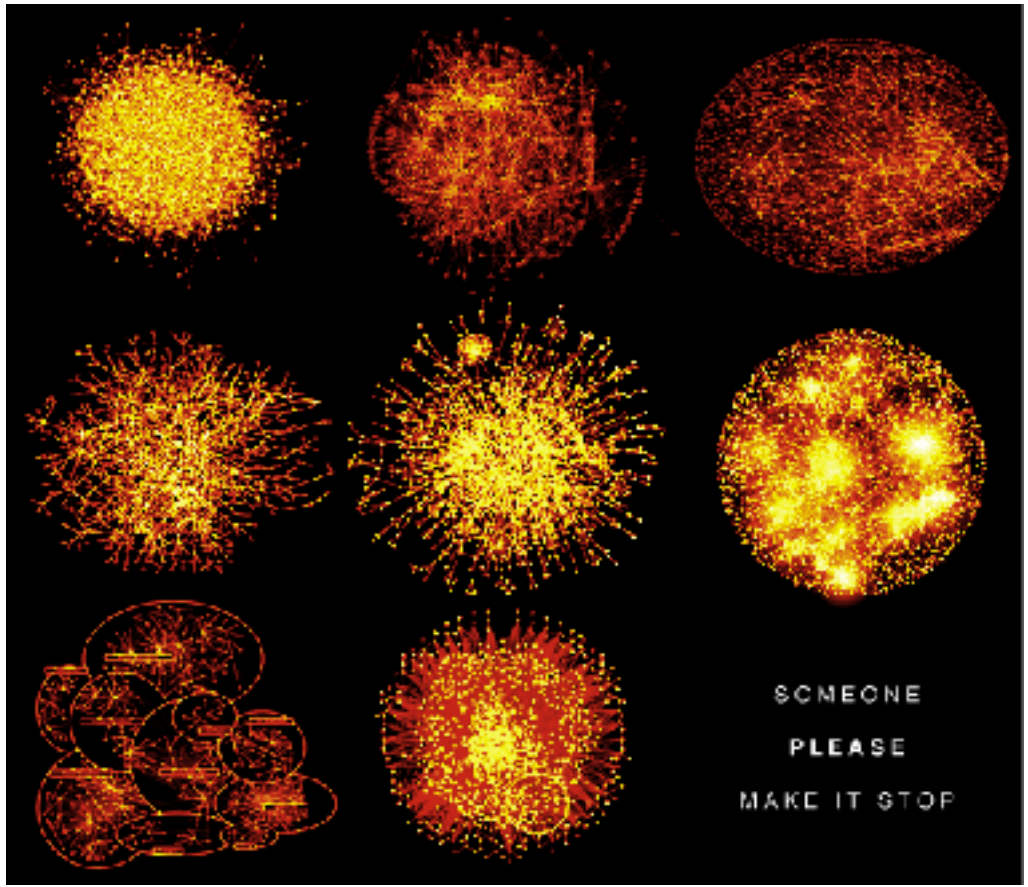
VISUALISING GRAPHS

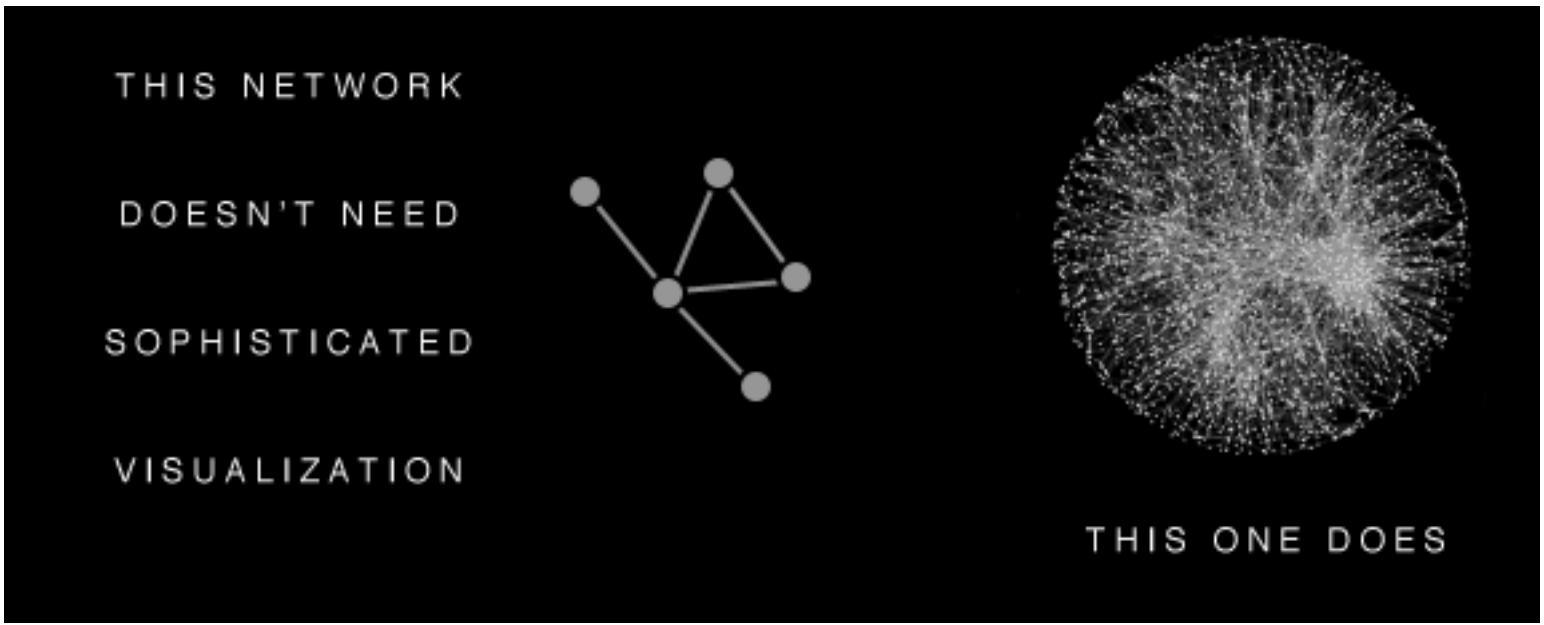
HOW DO WE VISUALISE A GRAPH?

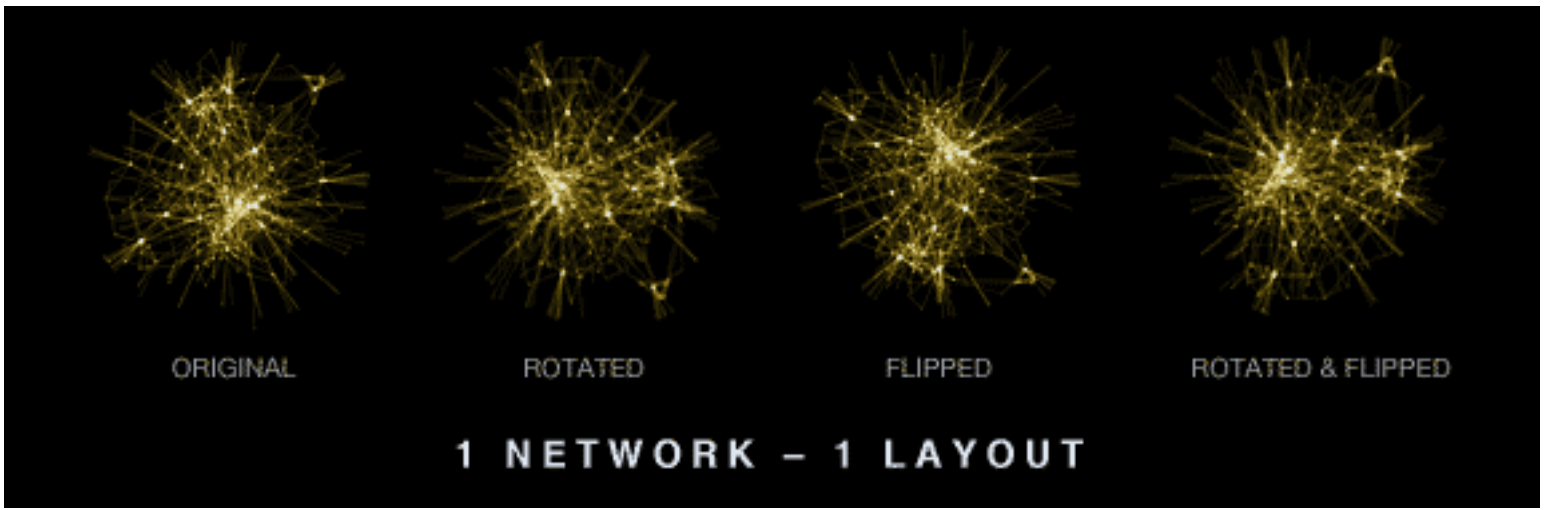



Hartford drug users network
In-degree
Out-degree
Number of nodes
Degree
0
20
40
60
80
100
0
2
4
6
8
10

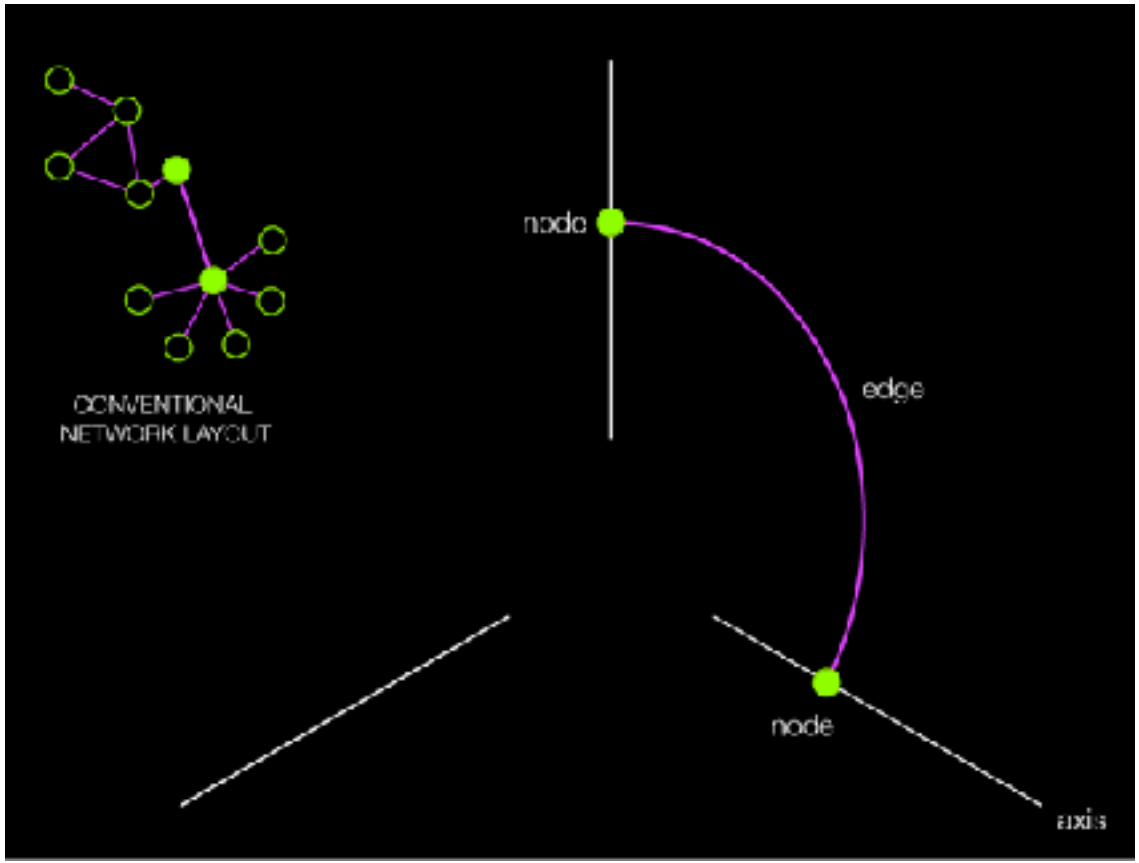
HOW DO WE VISUALISE A GRAPH?


SOMEONE
PLEASE
MAKE IT STOP

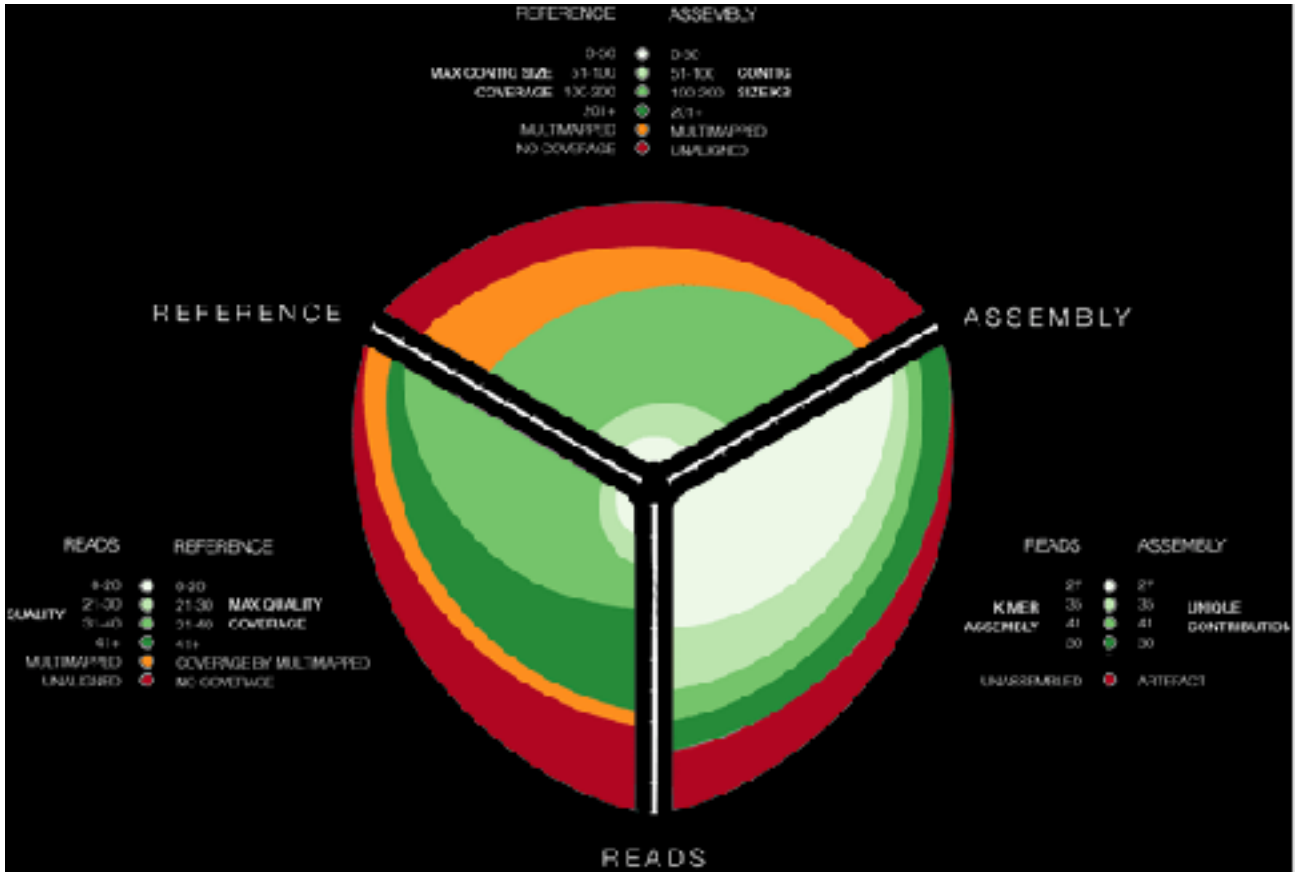

THIS NETWORK
DOESN'T NEED
SOPHISTICATED
VISUALIZATION
THIS ONE DOES


ORIGINAL
ROTATED
FLIPPED
ROTATED & FLIPPED
1 NETWORK – 1 LAYOUT

HOW DO WE VISUALISE A GRAPH?



HOW DO WE VISUALISE A GRAPH?



DATA SCIENCE PART TIME COURSE

FINDING COMMUNITIES

We all know on our Facebook page we have friends we made at different places and stages in our lives. We might have some work friends, school friends, some friends we made travelling, etc…

How might we find good groupings of communities?

The criteria for finding good communities is similar to that for finding good clusters.

We want to maximize intra-community edges while minimizing inter-community edges.

Formally, the algorithm tries to maximize the modularity of network, or the fraction of edges that fall within the community minus the expected fraction of edges if the edges were distributed by random. Good communities should have a high number of intra-community edges, so by maximizing the modularity, we detect dense communities that have a high fraction of intra-community edges.

DATA SCIENCE PART TIME COURSE

CASE STUDY BONAFIDE

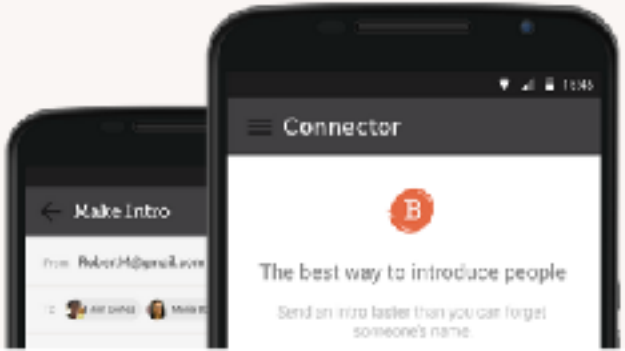
Bonafide

Login

Find your dream team.

Bonafide helps you find great people to work with via trusted introductions.

Sign up for Bonafide Connector



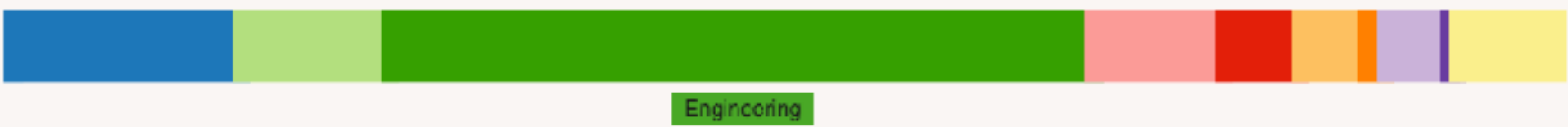
Connector

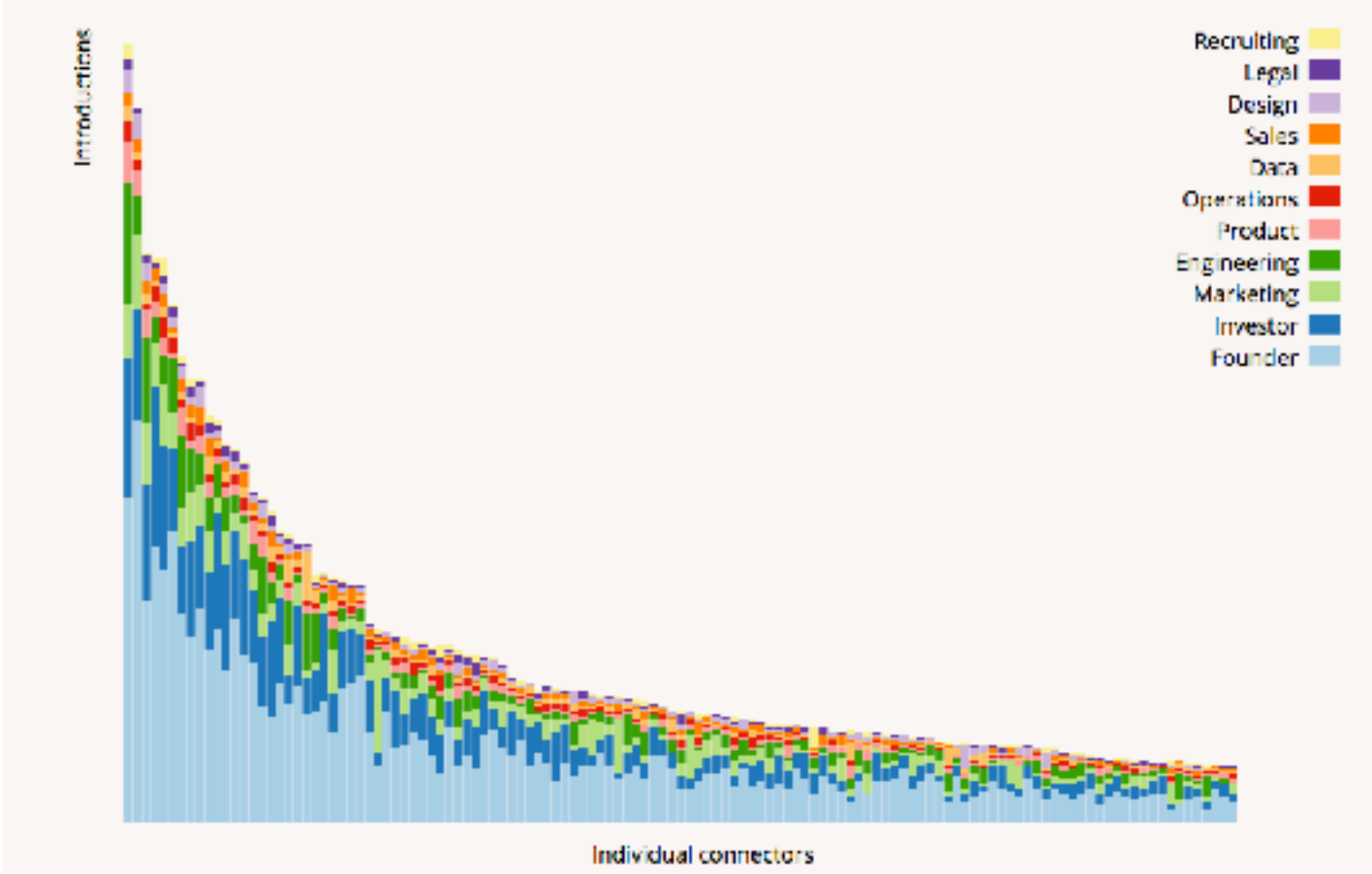
Make an intro in less time than it takes to forget a name.

Connect people in seconds in your own words from your own email address. Bonafide connects with Gmail to find quotes you've written about people in previous intros, letting you write intros more quickly.

Eva Ho, Investor/Advisor/Board Member:

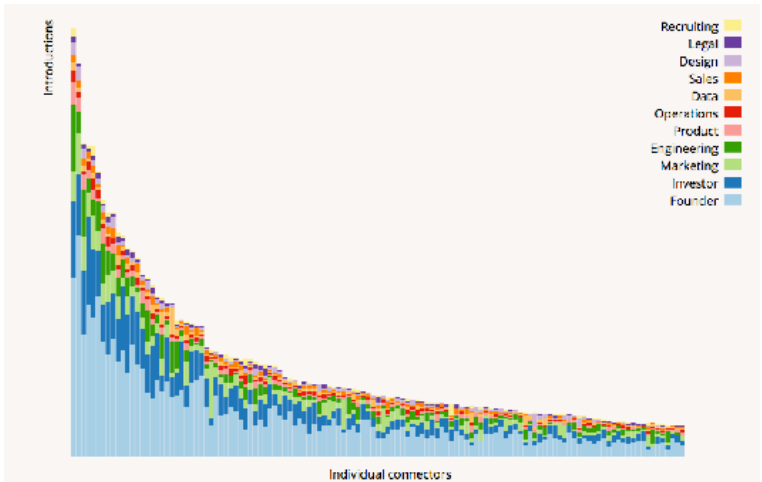
Tony Wu, Engineer at Uber:




Introductions
Recruiting
Legal
Design
Sales
Data
Operations
Product
Engineering
Marketing
Investor
Founder
Individual connectors

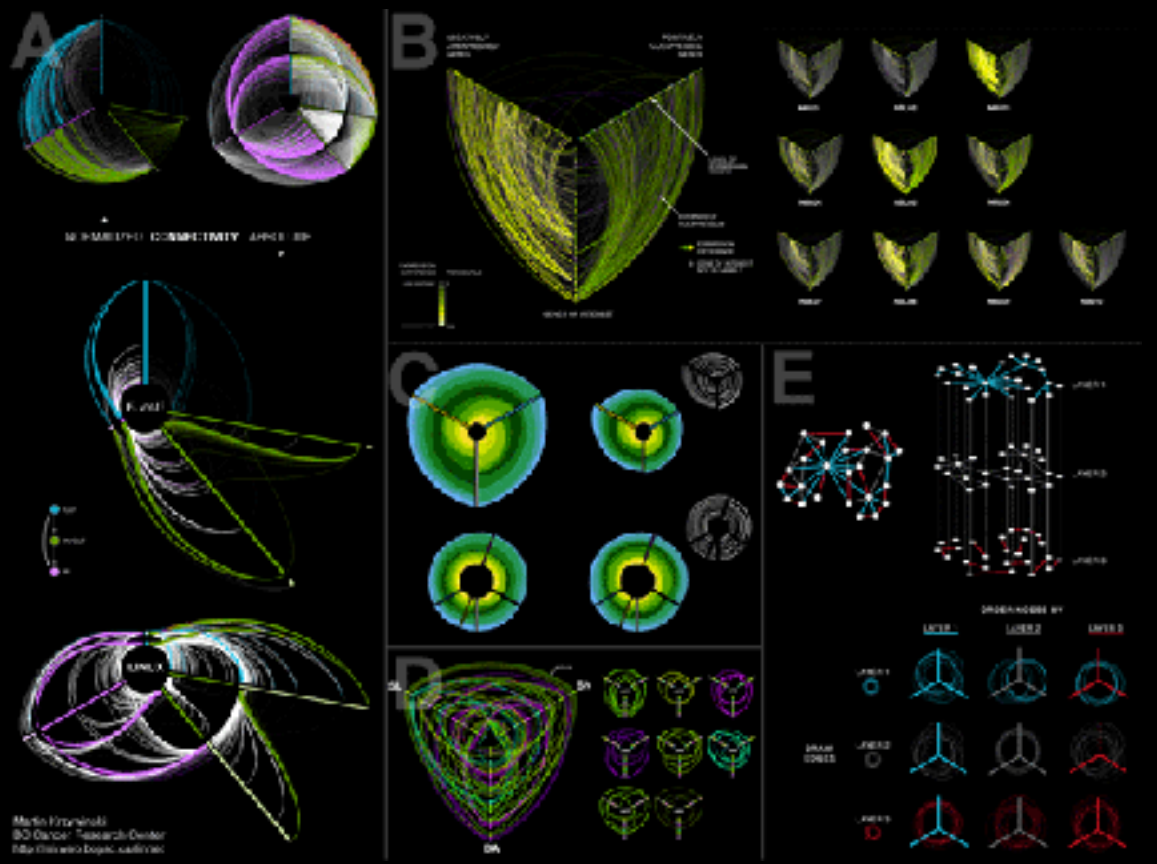
Two things stand out from the figure:

- A lot of people introduce founders
- Heavy connectors disproportionately introduce investors



https://bonafide.co/blog/which-professions-top-connectors-introduce

LAB



SYNCHING YOUR FORK WITH THE COURSE REPO

1. re-name your labs with lab_name.<yourname>.ipynb (to prevent a conflict)
2. cd <path to the root of your SYD_DAT_6 local repo>
3. commit your changes ahead of sync
   - git status
   - git add .
   - git commit -m "descriptive label for the commit"
   - git status
4. download new material from official course repo (upstream) and merge it
   - git checkout master (ensures you are in the master branch)
   - git fetch upstream
   - git merge upstream/master

HOMEWORK

Homework

- Download and install R from CRAN website
- Download and install RStudio

Revision class this weekend!

- Saturday 26th of November 11am–2pm – here at General Assembly