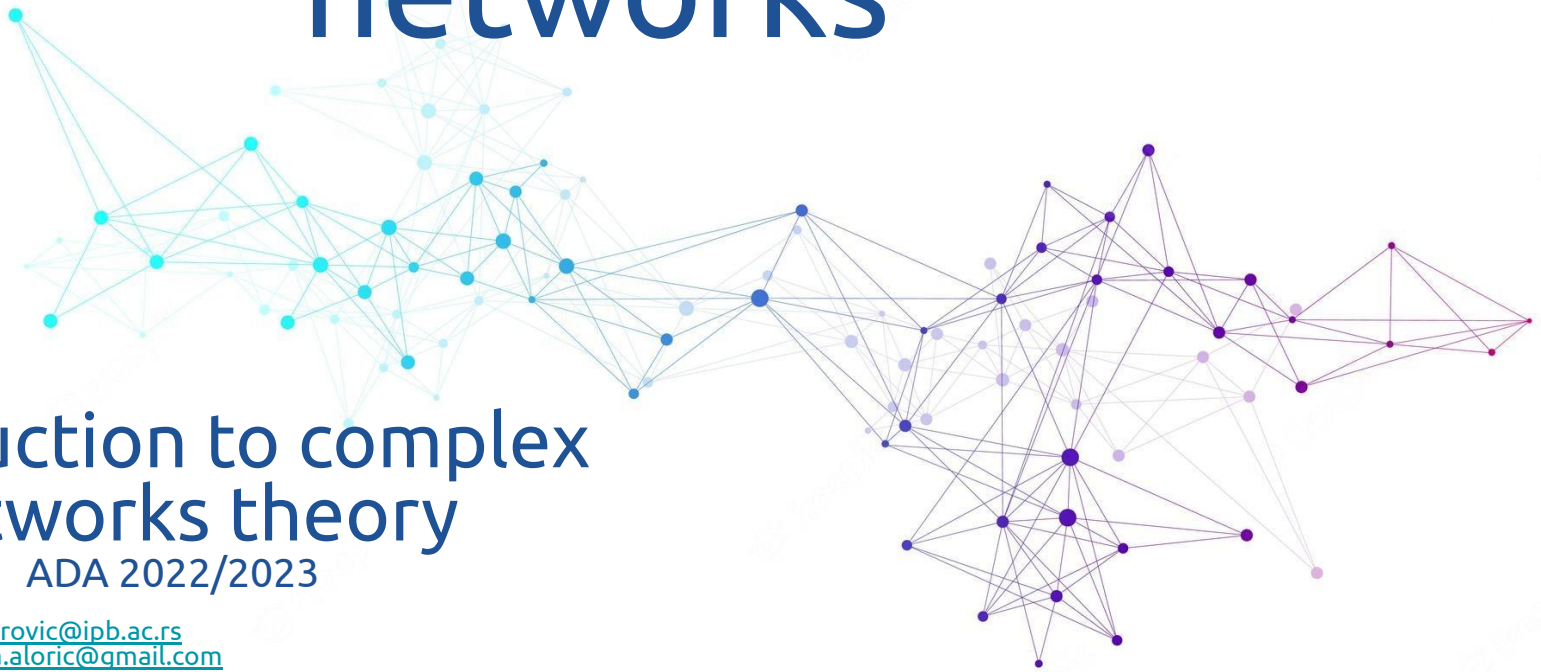
Types of links within networks



Introduction to complex networks theory

ADA 2022/2023

Marija Mitrović Dankulov mitrovic@ipb.ac.rs
Aleksandra Alorić aleksandra.aloric@gmail.com

Topics of this lecture

- Link properties:
  - Direction
  - Weight

- Degree (and analogues) in directed and weighted networks

- Bipartite networks

Directed networks

- Not all interactions are symmetric
- Directed links help us denote one way relationships
- Examples:
  - Network of followers on Instagram/Twitter that contrary to Facebook friend networks are not symmetrical, although follows could be reciprocated
  - Citation networks
- When tracing information flows, malware propagation, disease spread, you can always see direction of spread, or source and a target

Company organisational chart

- Nodes: employees or job positions
- Links: delegation of tasks & responsibilities, could be also who answers to whom relationship

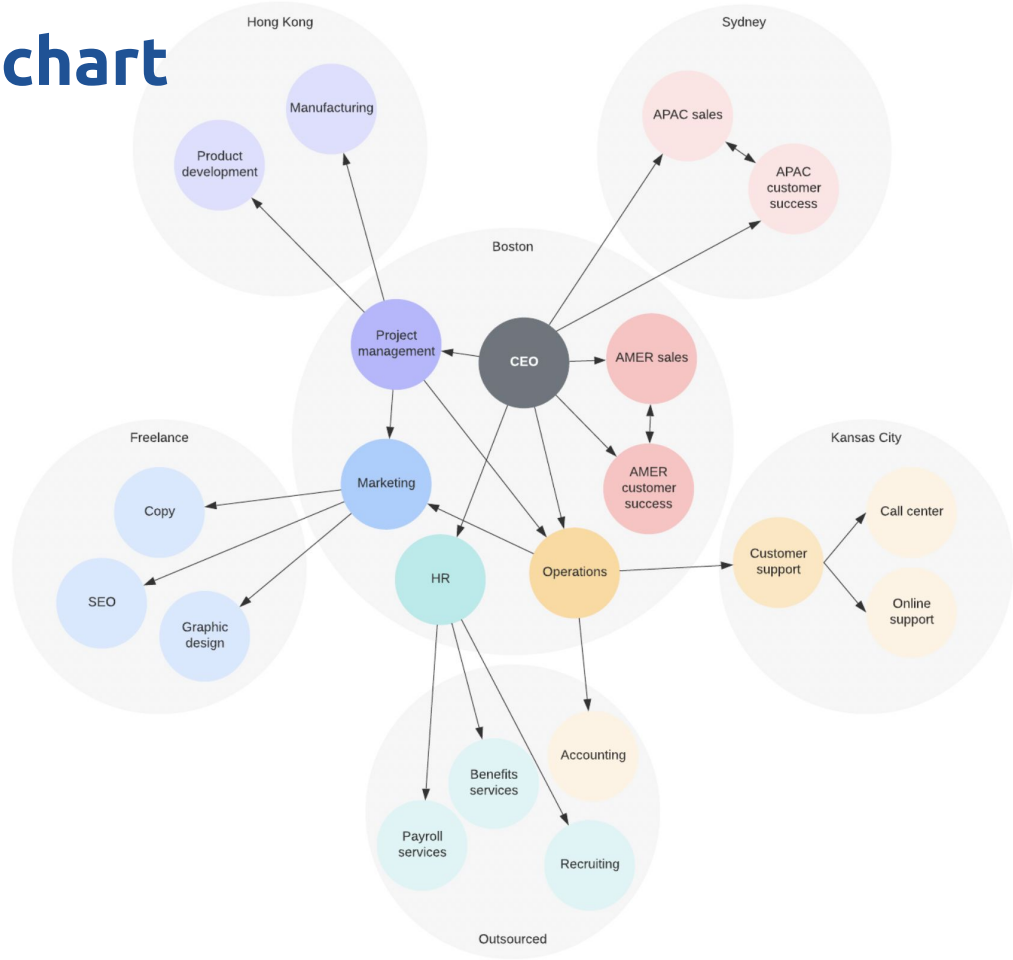


Image source:
https://www.lucidchart.com/blog/types-of-organizational-structures

Food web network

- Nodes - species
- Links - who's eating whom

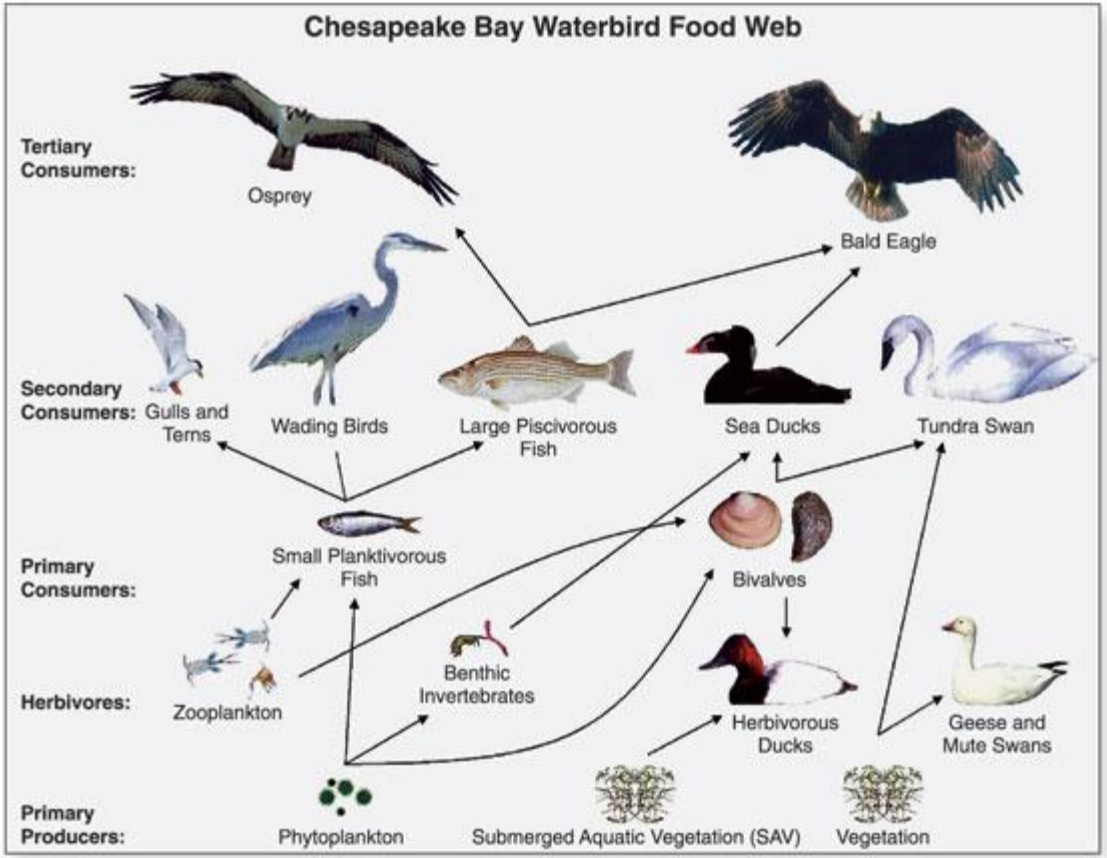


Image source:
https://en.wikipedia.org/wiki/Food_chain

Football pass network

- Nodes: Football players
- Links: ball passess between players

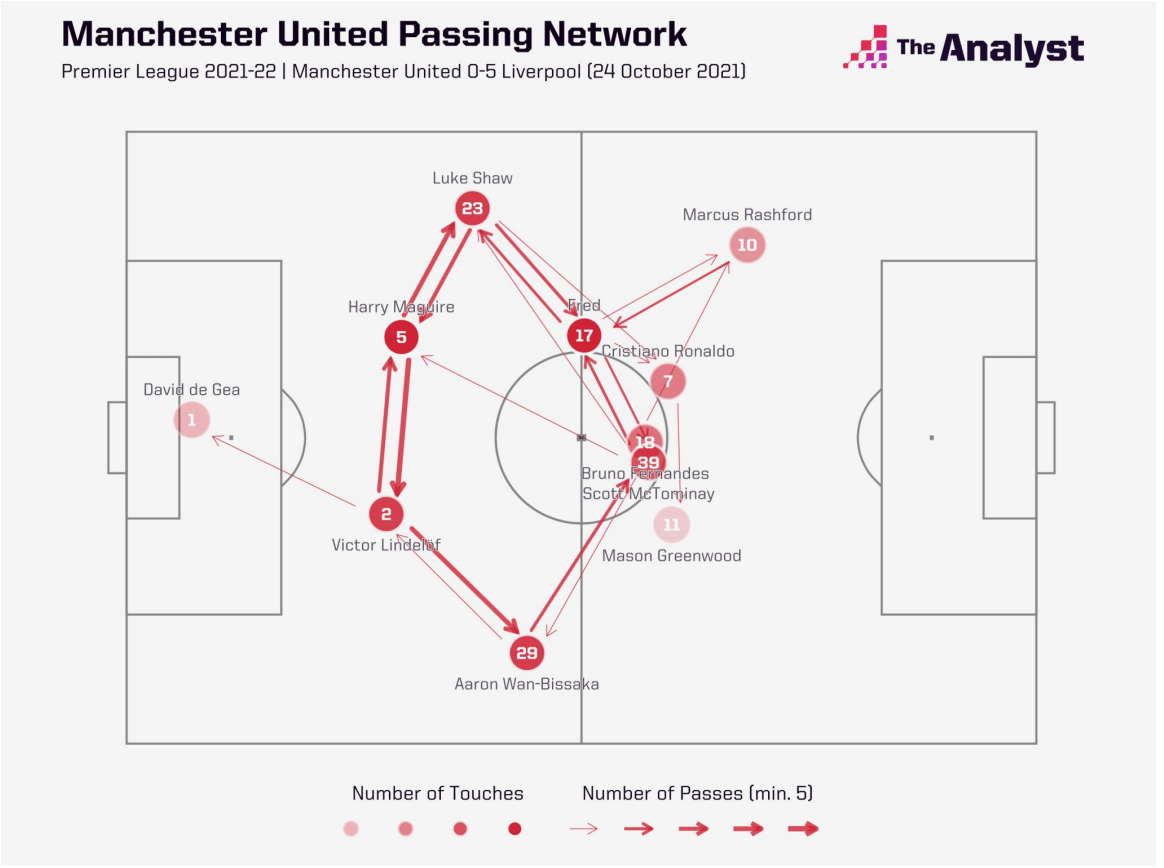
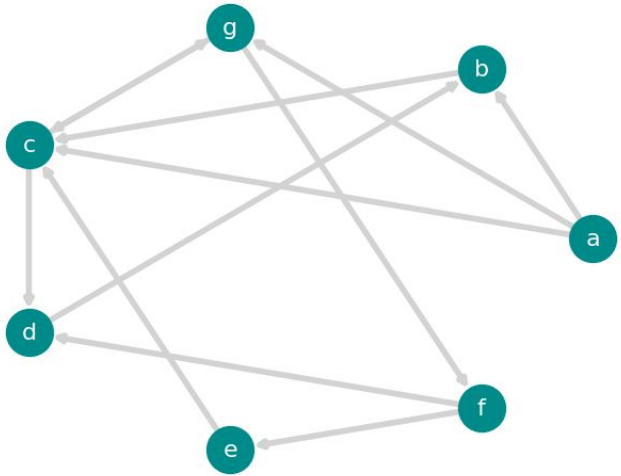


Image source:
https://theanalyst.com/eu/2021/10/ole-gunnar-solskjaer-hanging-on/

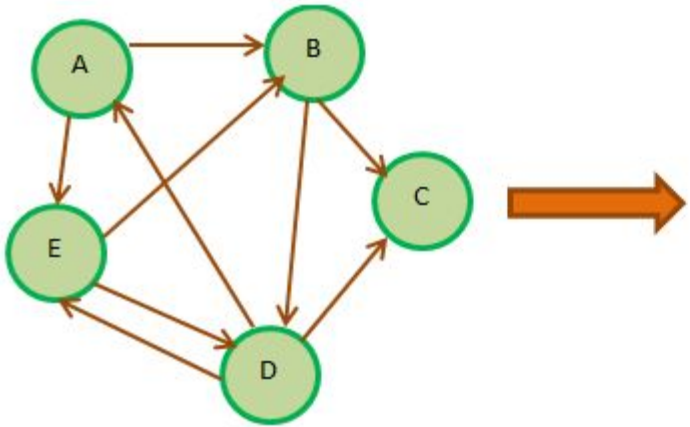
Directed networks

- We need to extend the notion of degree, besides the number of links related to the node, it is important how many of these links are:
  - Starting from the node i - **out degree k_out**
  - Finishing at the node i - **in degree k_in**
- Looking at the graph on the right
  - Which node has the highest indegree?
  - Which node has the highest outdegree?
  - What node would have the highest degree if this was undirected network?



Adjacency matrix of directed networks

- The adjacency matrix is not symmetric!
- A_{ij}=1 means there is a link **from i to j** *



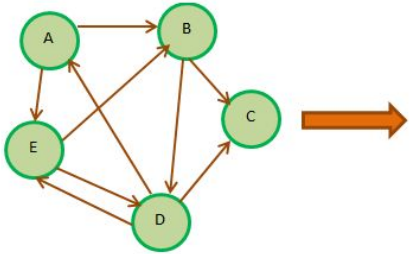
Directed graph

	A	B	C	D	E
A	0	1	0	0	1
B	0	0	1	1	0
C	0	0	0	0	0
D	1	0	1	0	1
E	0	1	0	1	0

Adjacency Matrix

*Pay attention in Barabasi's book, element of adjacency matrix is defined in the other direction!

Adjacency matrix



Directed graph

	A	B	C	D	E
A	0	1	0	0	1
B	0	0	1	1	0
C	0	0	0	0	0
D	1	0	1	0	1
E	0	1	0	1	0

Adjacency Matrix

- Number of links: $L = \sum_{i,j=1}^{4} A_{ij}$
- Node out degree: $k_i^{out} = \sum_{j=1}^{4} A_{ij}$
- Node in degree: $k_i^{in} = \sum_{j=1}^{4} A_{ji}$

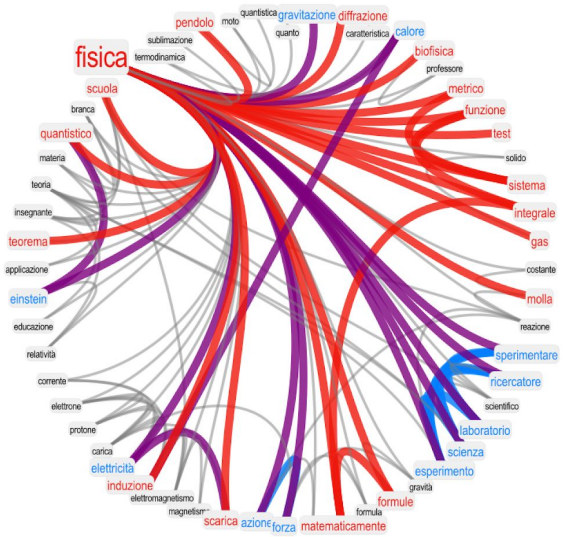
Directed networks in python

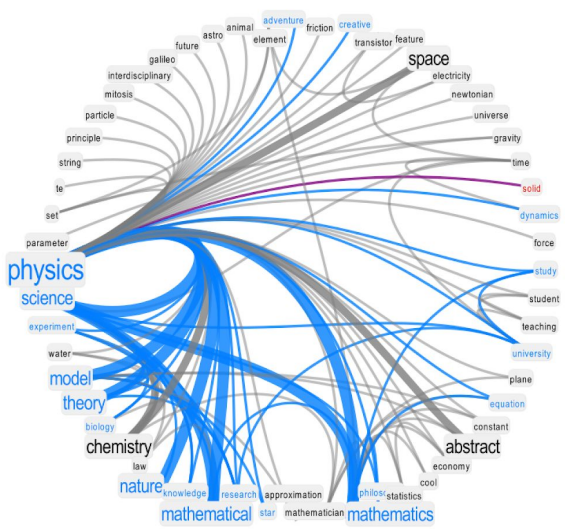
Weighted networks

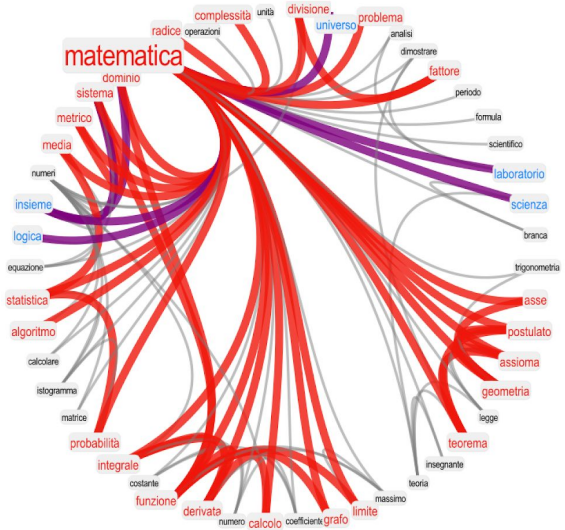
- Not all links are equal!
- If you are considering a road network, should a single lane street be the same as the highway?
- If we’re creating contact network between people, duration of close contact between individuals would be a very important to consider
- Other examples - sewer network, river networks, electrical grid
- We can add weight to both directed and undirected links

Word associations

- Nodes: words
- Links: free associations
- Weight: frequency of given association pair







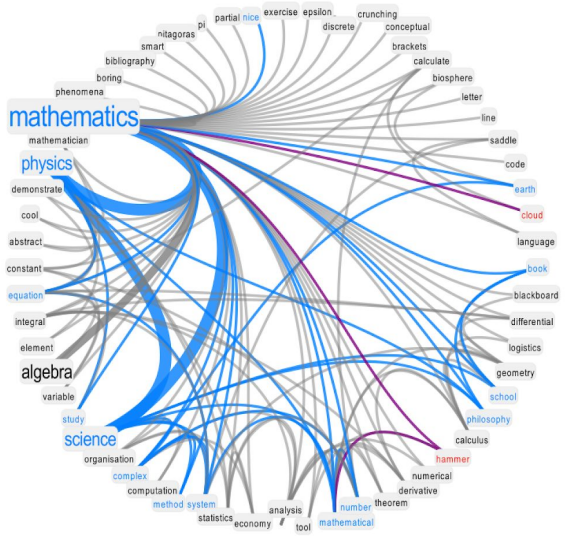


Image source:
https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0222870

A Song of Ice and Fire

- Nodes: book characters
- Links: two characters are in the same location, talk together, name co-occurrence in the text
- Weight: frequency of character

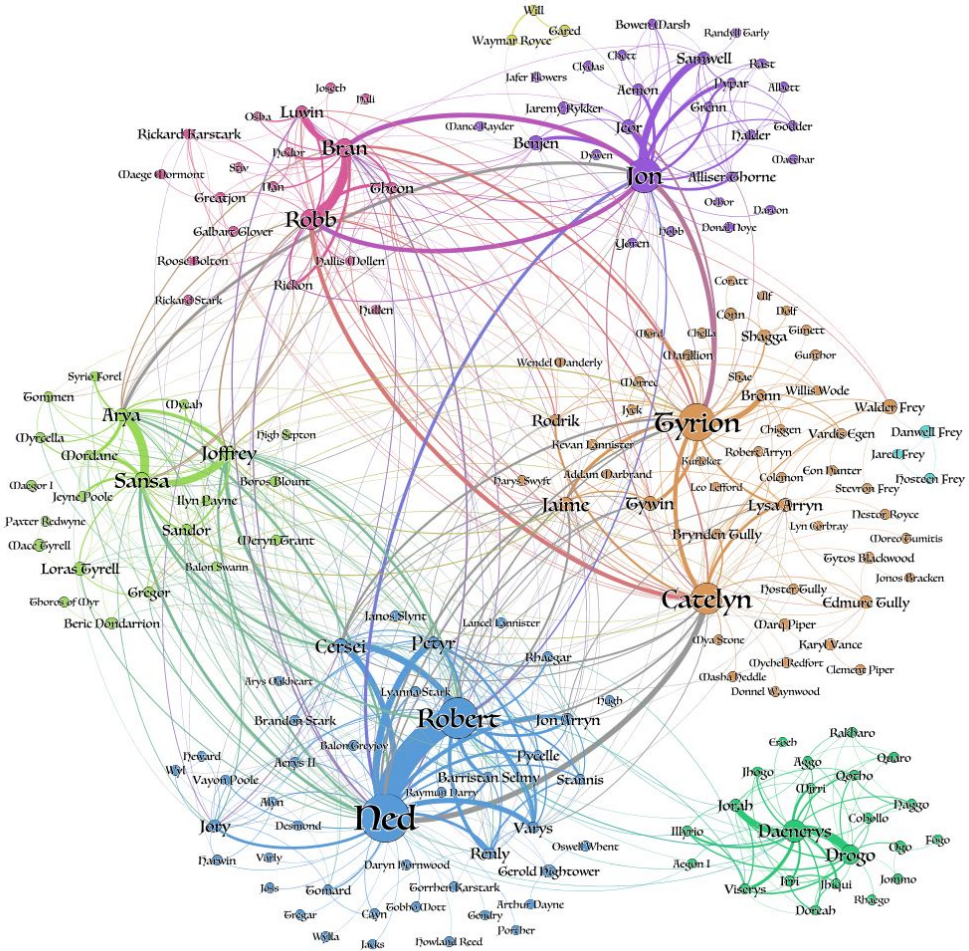
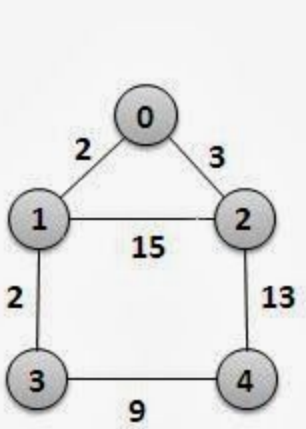


Image source:
https://networkofthrones.wordpress.com/from-book-to-network/

Adjacency of weighted networks

- Elements of the adjacency matrix are now real numbers

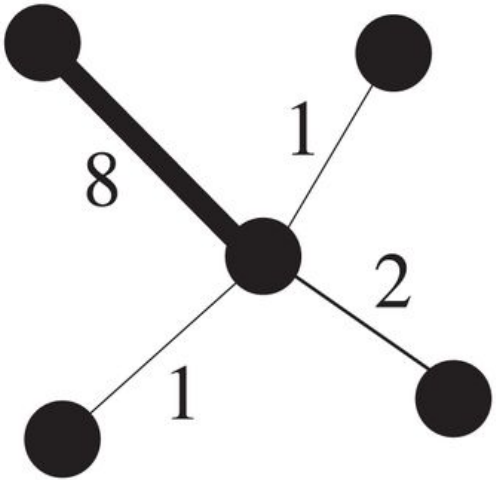


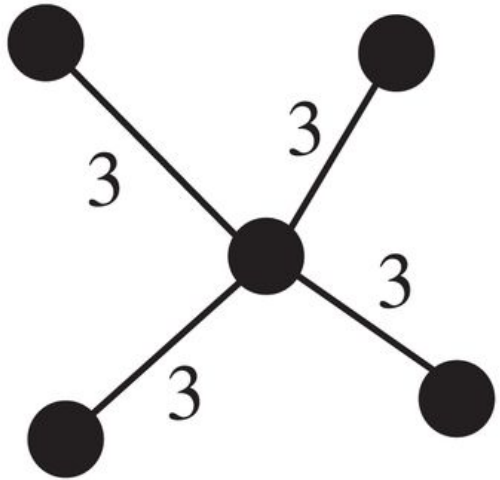
	0	1	2	3	4
0	0	2	3	0	0
1	2	0	15	2	0
2	3	15	0	0	13
3	0	2	0	0	9
4	0	0	13	9	0

Adjacency Matrix Representation of Weighted Graph

Weighted networks

- Besides node degree, it makes sense that we consider also node **strength**, that considers weights of links, not only their number
- How would you calculate it from adjacency matrix?
- Picture shows simple networks in which the middle node has the same degree and strength and degree, yet weight distribution of surrounding links is not the same



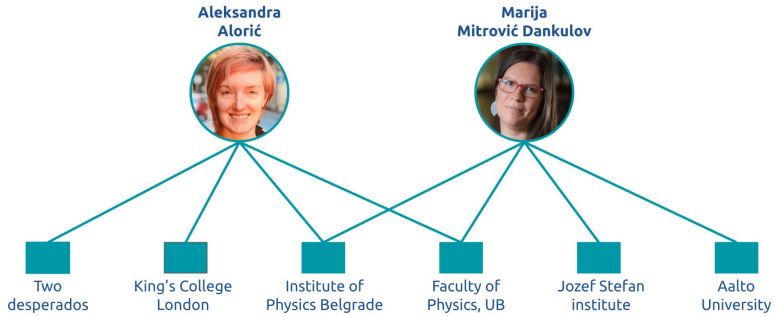


Weighted networks

- Python -> show how we could calculate strength from the pandas dataframe (not only from the networkx)

Weighted networks in python

Bipartite networks



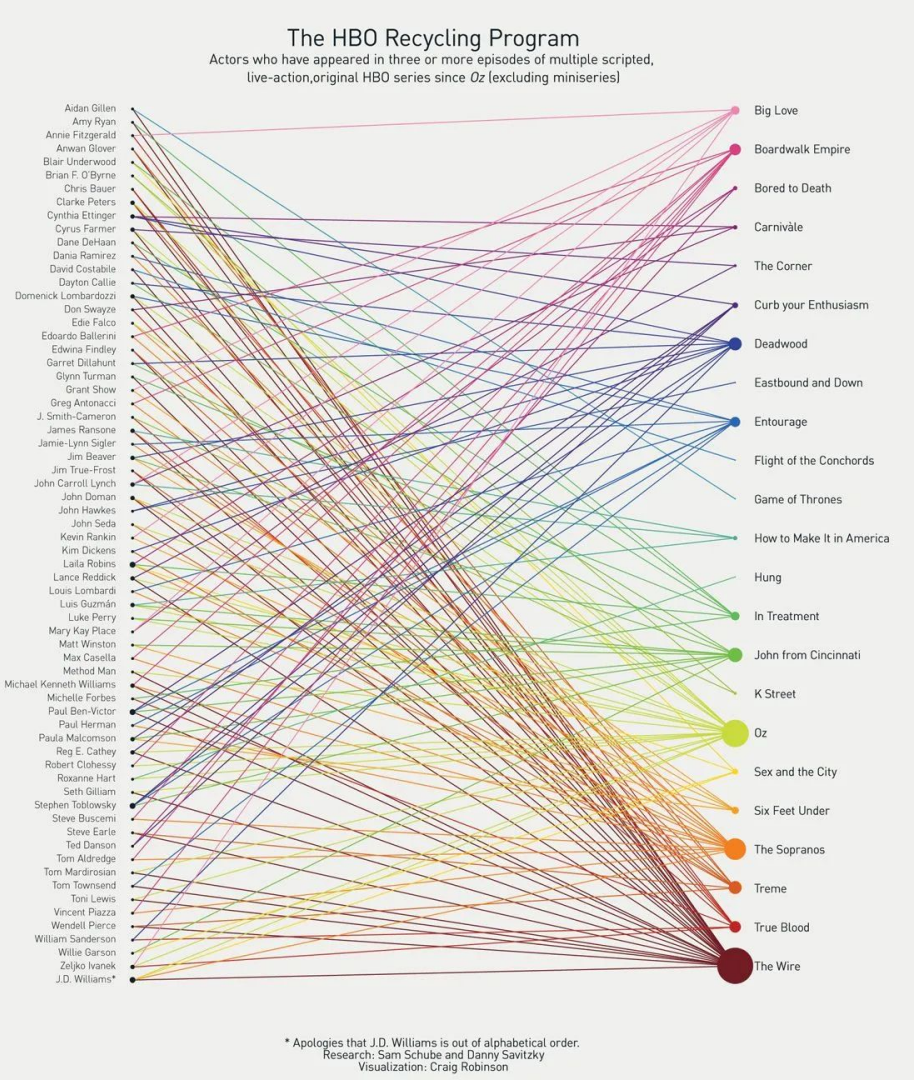
- First network we've seen in this module :)
  - Nodes in two different groups
  - Links can exist only between nodes from different groups
  - G = (U, V, E), where U and V are sets of nodes of different group, while E contains list of links between them

- Most of the examples you've generated in the first homework (and those I've been sharing on the first class) are actually bipartite networks, e.g.
  - Network of mental disorders based on symptoms (one group of nodes disorders, other group symptoms, links showing disorder symptom associations)
  - Network of students based on the classes they listen to at UoB (one group of nodes are students, other classes, links when a student attend a class)

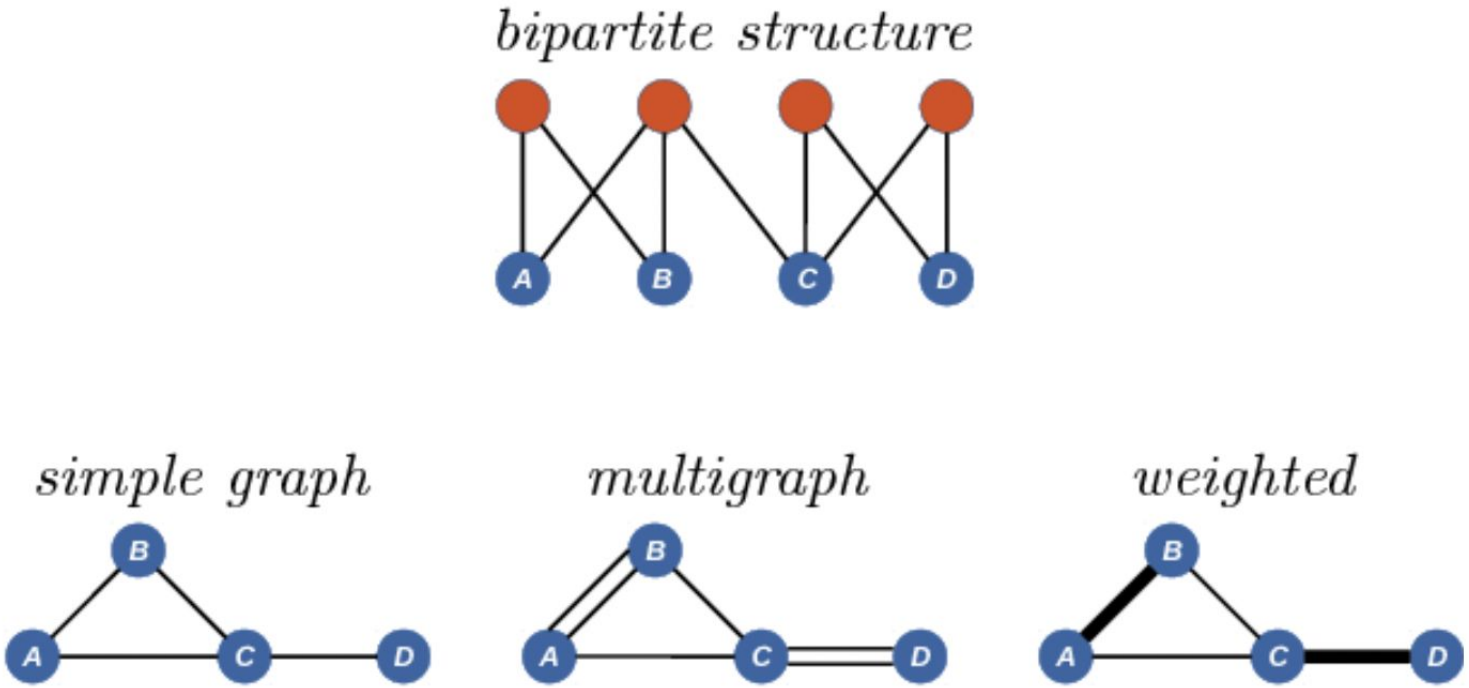
Network of actors and HBO shows

- Nodes: actors & tv shows
- Links: appearance of an actor in a show

Image source:
http://grantland.com/features/the-hbo-recycling-program/
Interactive version:
https://zgrossbart.github.io/hborecycling/



Projections of bipartite networks



Network of movies and their authors

- Bipartite network of movies and authors (directors, writers and composers)
- Movie network projection
- If we made projection to authors will it be connected?

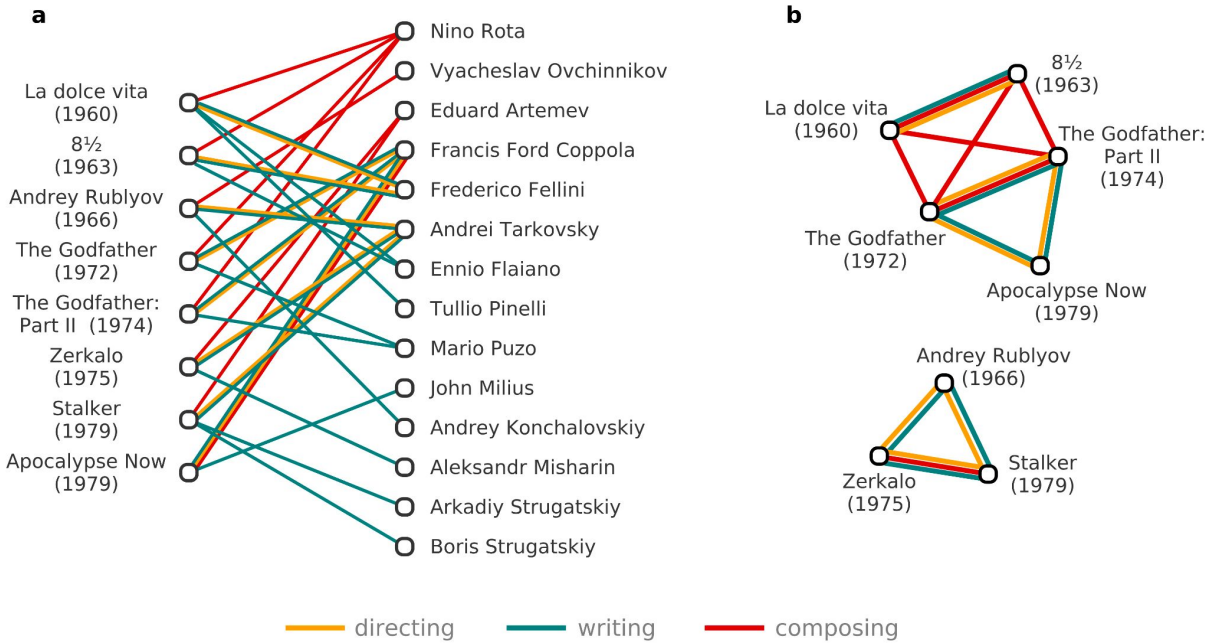
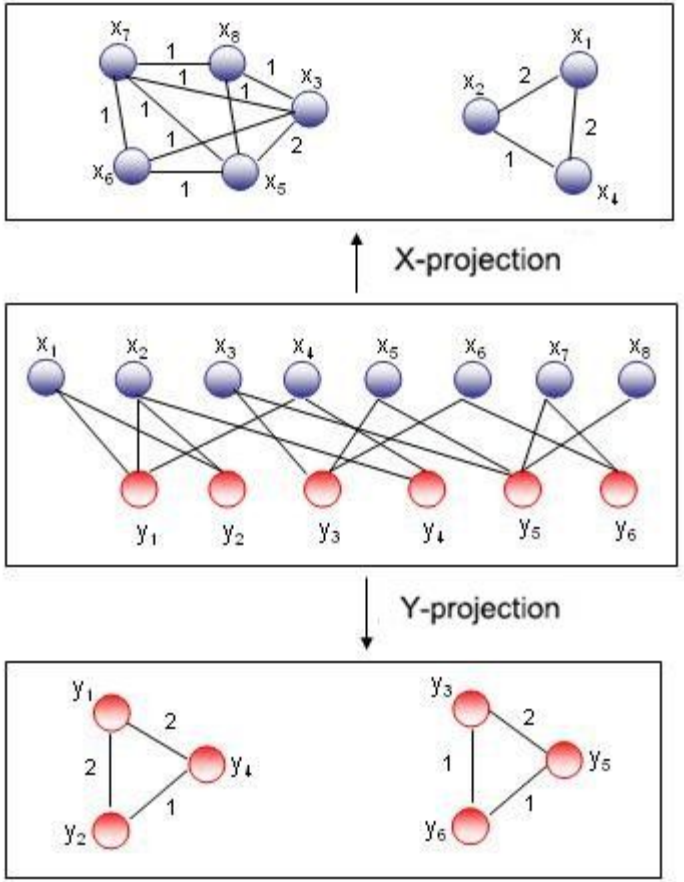


Image source:
https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0108857

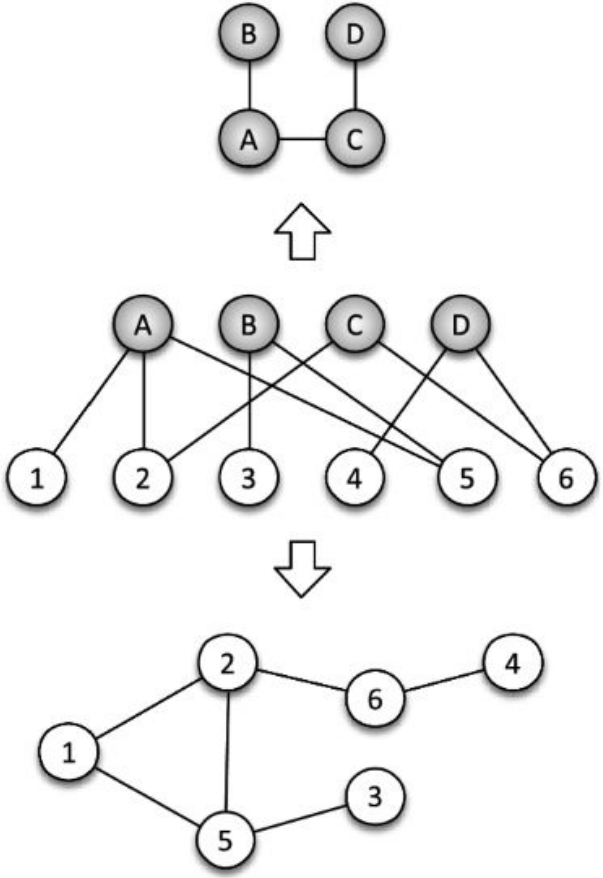
Choice of projection

- Projection can be made on both groups of nodes
  - Nodes are connected in a projection, if both nodes were connected to at least one same node in the other group
  - By default, the projection is a weighted network, but you can transform it to unweighted if that makes more sense for the problem you are studying
  - The resulting projection networks are usually structurally different



Adjacency in Bipartite networks

- Adjacency matrix is not a square matrix any more
- As before it contains info about the number of links, from it we can calculate degrees
- Its elements can be!=1 as bipartite networks can also be weighed



$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array} \begin{array}{c} \begin{array}{cccc} A & B & C & D \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \end{array}$$

Human disease network

- Now we can see how human disease network we saw in first class was constructed
- We can see its complementary network projection to genes

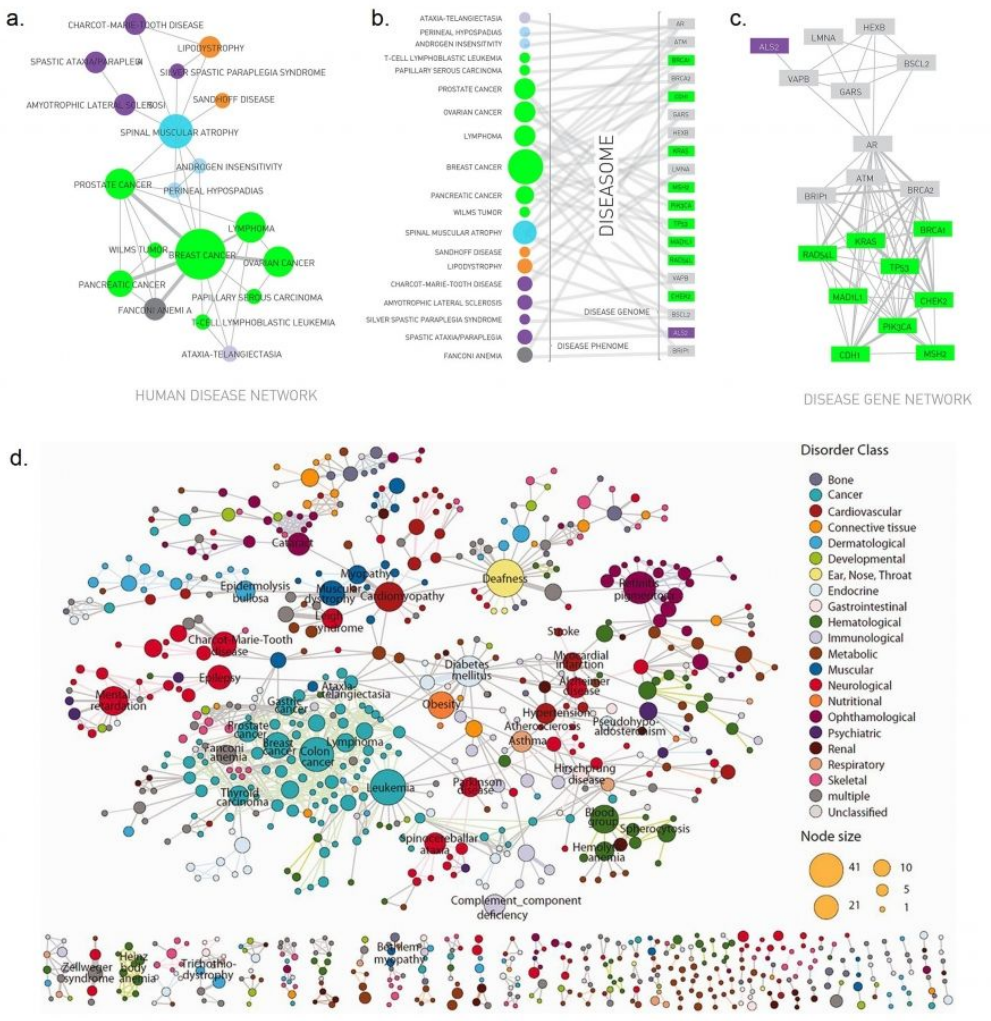


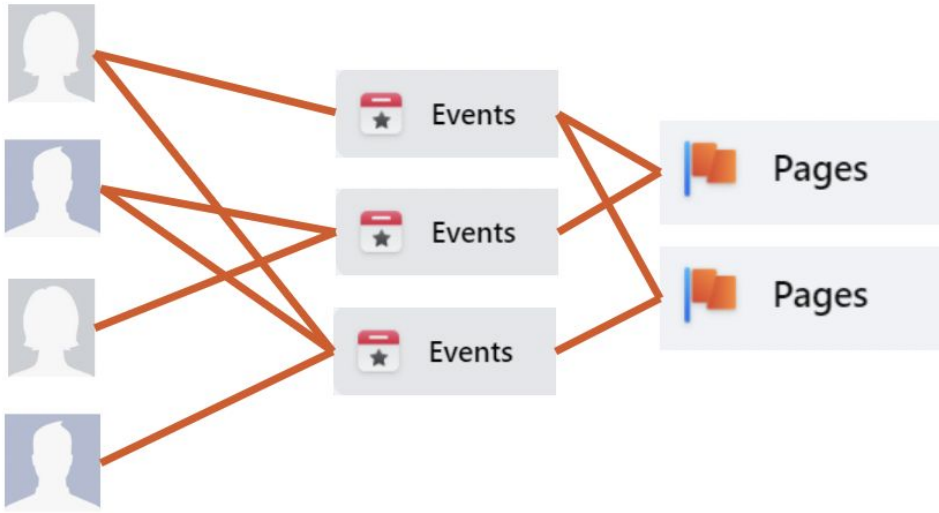
Image source: Fig2.10 in Barabasi's book, while the interactive version should be here: https://archive.nytimes.com/www.nytimes.com/interactive/2008/05/05/science/20080506_DISEASE.html?ref=health

Bipartite networks - projection

- Weighted networks -> unweighted
- Examples where projection creates fully connected and why we wouldn’t want that
- Explain filtering intuitively

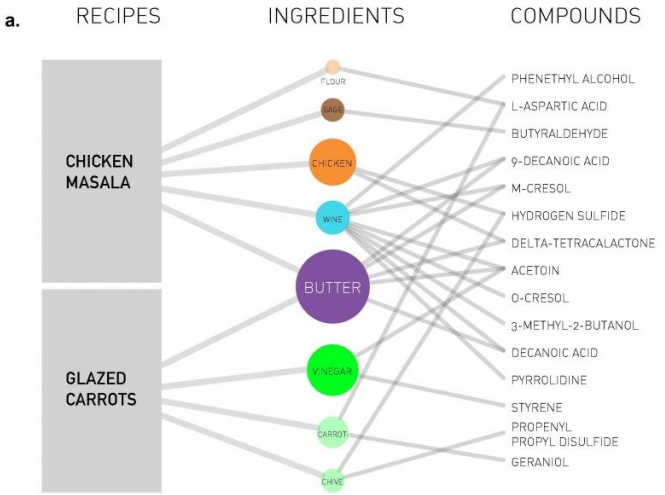
Multipartite networks

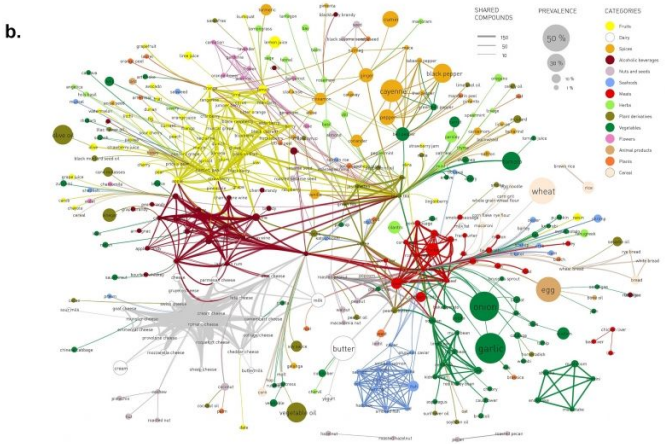
- The notion of bipartite networks can be easily extended to more than two groups of nodes
  - Profiles
  - Events
  - Pages that created events



Cuisine network

- Nodes:
  - Dishes
  - Ingredients
  - Compounds

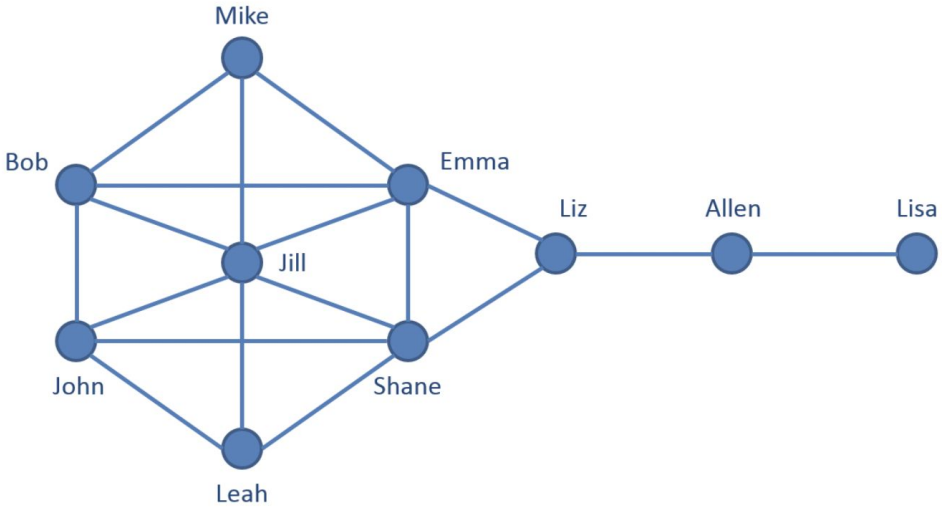




Bipartite networks in python

HW - comments

- First homework was really nice, some of your network examples were inspiration for figures in this presentation
  - Pay attention that it is not enough to say ‘road network’ or links between the nodes are ‘interactions’, it is important to specify what are nodes and links in the network, and what type of interaction is the link standing for
- Third homework
  - Let’s solve it together :)



Homework

Submission link