Social Network Analysis and Visualization in Gephi

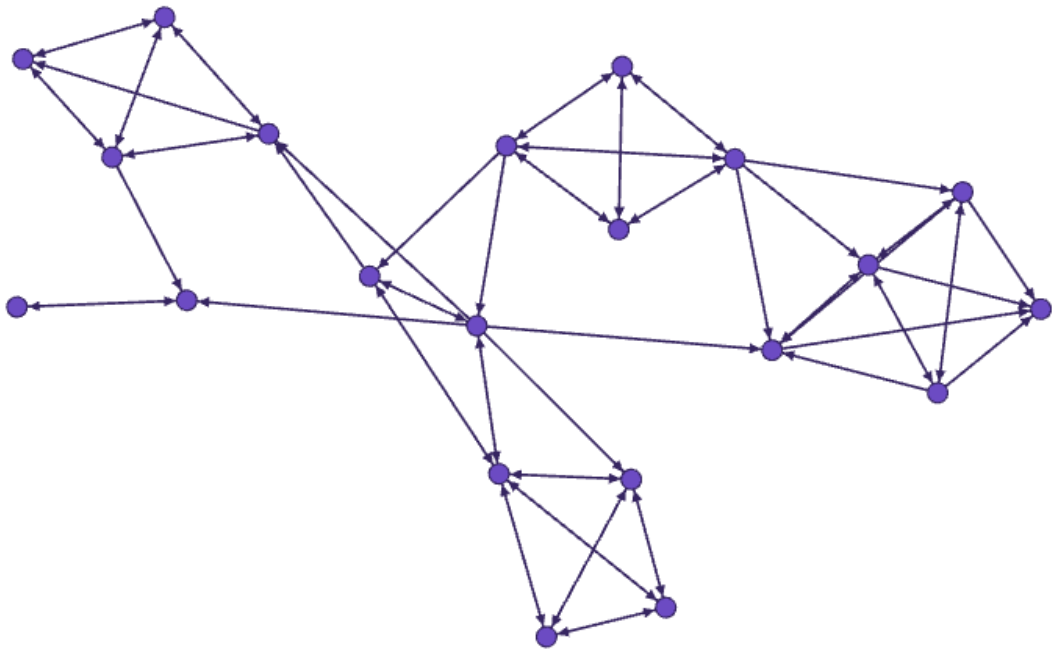
CLUe training June 23, 2017

Rense Corten
(with contributions by Bas Hofstra and Lukas Norbutas)
Department of Sociology

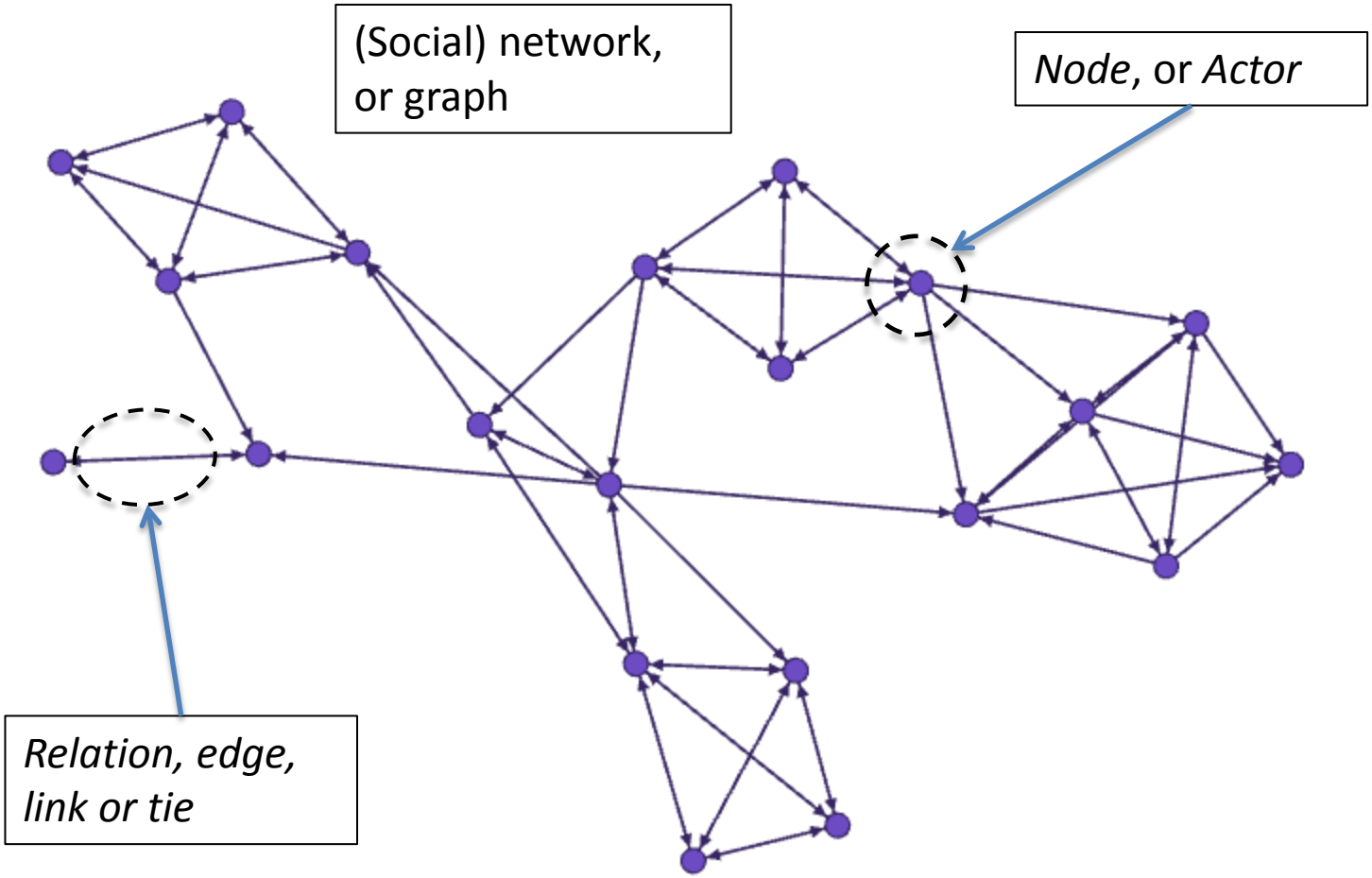
Analysis of the *structure of social interaction*

Examples of *structural* questions:

- How “cohesive” is a group?
- Can we identify “cliques”?
- What is the average distance (cf. “six degrees”)?
- Who is most “important” in a group?



Terminology



Nodes

- “Dots” in a network picture
- Individual people, organizations, etc…
- Sociological terminology: “actors”

Relations/ties: examples

- Friendship
- Acquaintance
- Love
- Hate
- Trade/exchange
- Authority
- Proximity
- … almost anything can be a network!
  (But not everything is interesting)

How to measure social networks?

- Etnographic/direct observation (e.g., McFarland 2001)
- Survey methods
  - ... among a sample of individuals from a population → ego networks research
  - ... Among a deliniated subset of individuals → complete networks (sociometric approach)

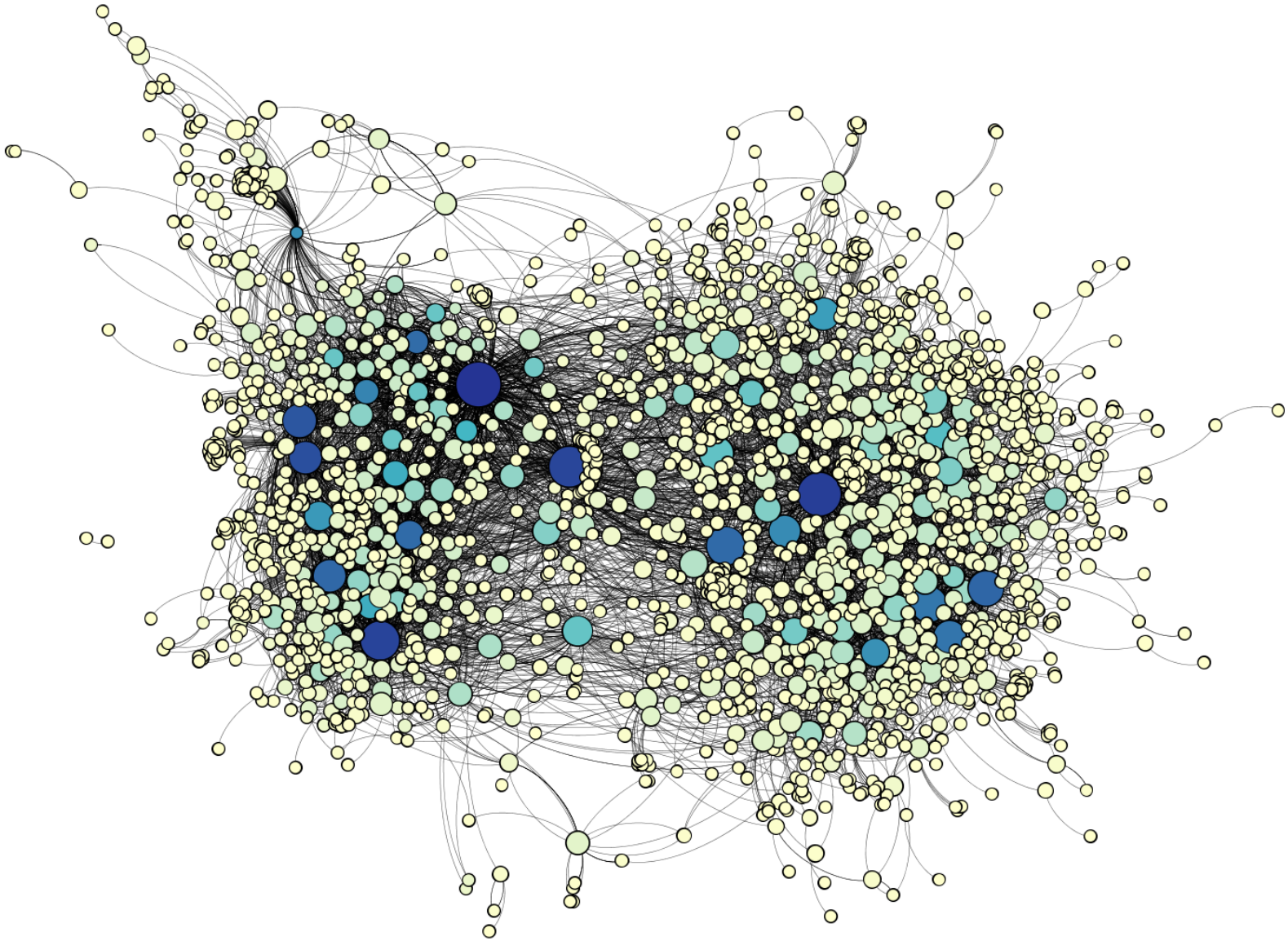
- **From records of behavior → online social networks**

Some types of relations in online contexts

- Facebook: friendship, likes, tagging, mentioning, commenting
- Twitter: following, retweeting, replying, mentioning, liking
- Websites (e.g., blogs): hyperlinks
- Forums: reponses to posts
- Online markets: who trades with whom?
- E-mail: messages
- Etc. etc. etc.

Some cautionary notes about interpreting networks

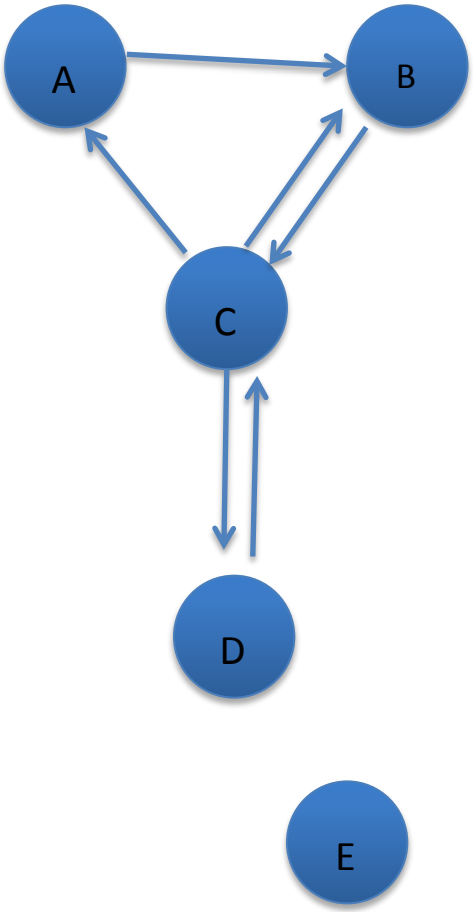
- Again: almost anything can be graphed as a network, but not everything is interesting
- Interpretation of network effects and network structure depends on the *meaning* of the ties
- For example: compare: follower- and retweet networks on Twitter
- Looking at network graphs is only useful with a *theory* about the underlying social process!



Mentioning or quoting in a Dark Web discussion forum (graphic by Lukas Norbutas)

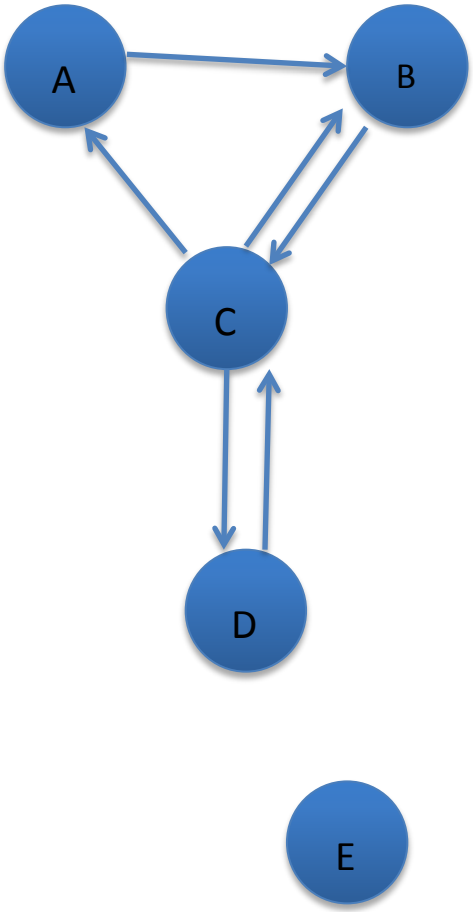
Storing network data 1: the adjacency matrix

→ likes	A	B	C	D	E
A					
B					
C					
D					
E					



Storing network data 1: the adjacency matrix

→ likes	A	B	C	D	E
A	0	**1**	0	0	0
B	0	0	**1**	0	0
C	**1**	**1**	0	**1**	0
D	0	0	**1**	0	0
E	0	0	0	0	0

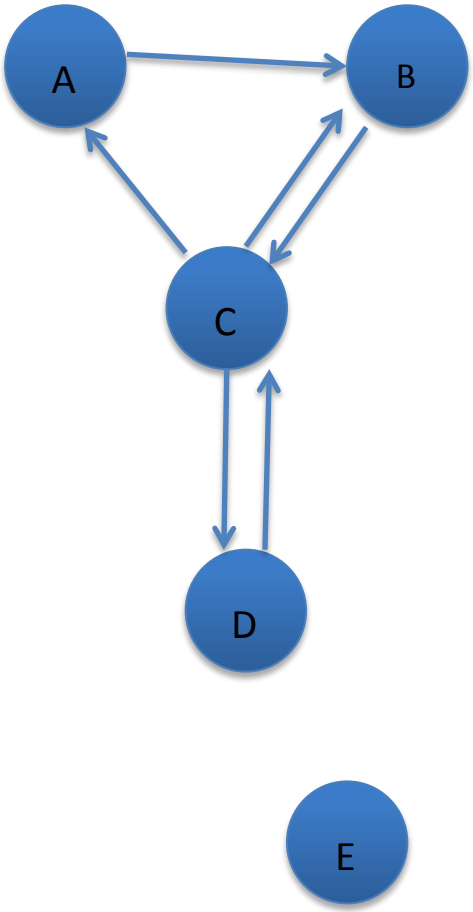


Storing network data 2: ties-as-cases (aka the edgelist or arclist)

→ **Likes**

A	**B**
B	**C**
C	**B**
C	**A**
C	**D**
D	**C**

E does not appear in this list! (Why not?)



A typical SNA project uses two types of data:

Attribute data

ID	Male	Age
A	1	26
B	0	23
C	0	30
D	1	22
E	1	28

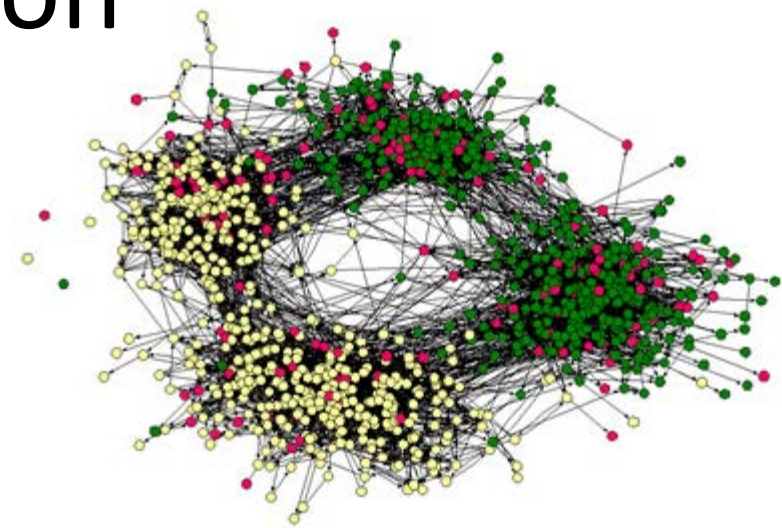
Relational data

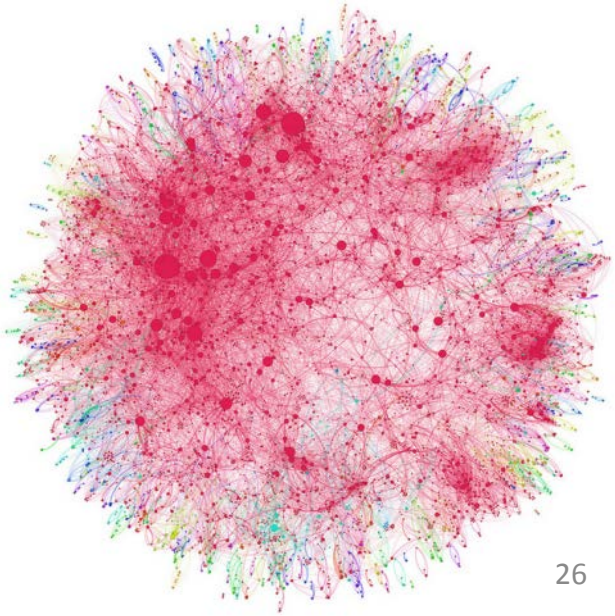
→ **Likes**

A	B
B	C
C	B
C	A
C	D
D	C

Visualization

- Goal: to highlight the structure of the network visually
- Typically only a first step, before further analysis
- Principal problem to solve: where to place the nodes in a two-dimensional space
- Many algorithms: most try to minimize crossing edges and keep edge lenght constant
- Result depends crucially on the algorithm!
- Extensions: 3D and animation





Software for network analysis

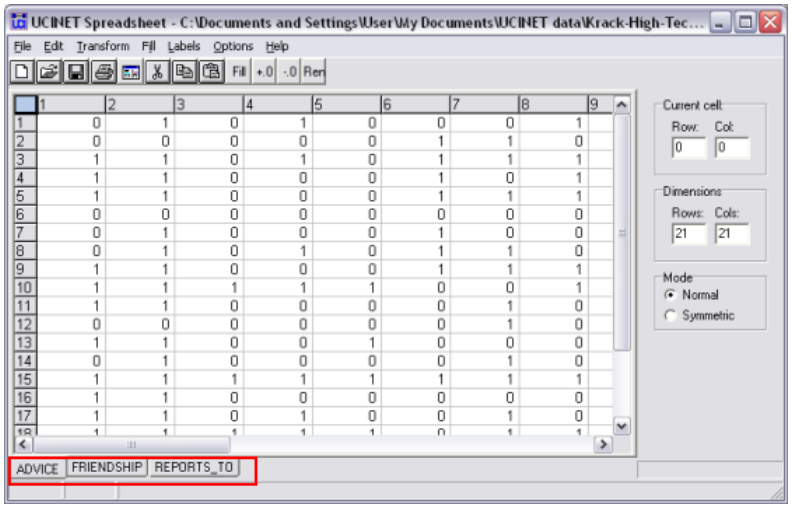
- “Conventional” generic statistics packages (SPSS, Stata): not designed for handling relational data, no or few facilities for visualization

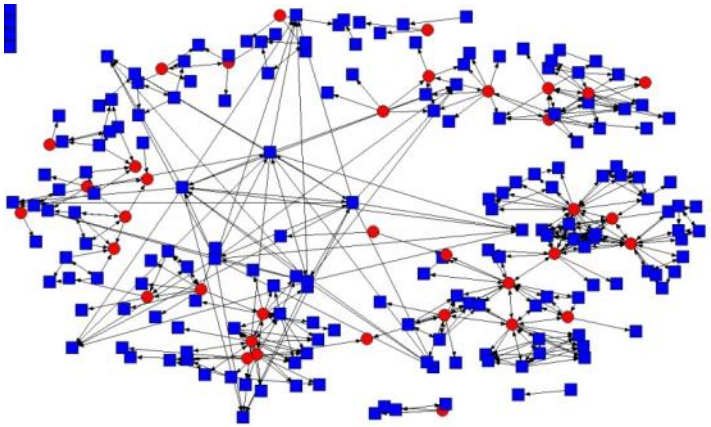
- Dedicated SNA software (UCINet, Pajek, Gephi, Visone): good at handling relational data, great for visualization, often not very flexible

- (Statistical) Programming languages (R, Python): powerfull and flexible, but steeper learning curve. Dedicated SNA packages available.

UCInet + Netdraw

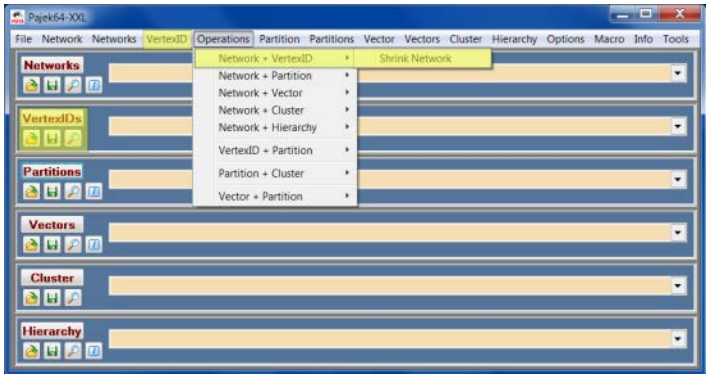
- Around since 1980's
- Developed by sociology SNA crowd (Borgatti, Everet, Freeman)
- Non-free
- MS Windows only
- Many options for analysis
- Not great for large networks
- Decent visualization, but not great
- Relatively user friendly
- Somewhat scriptable

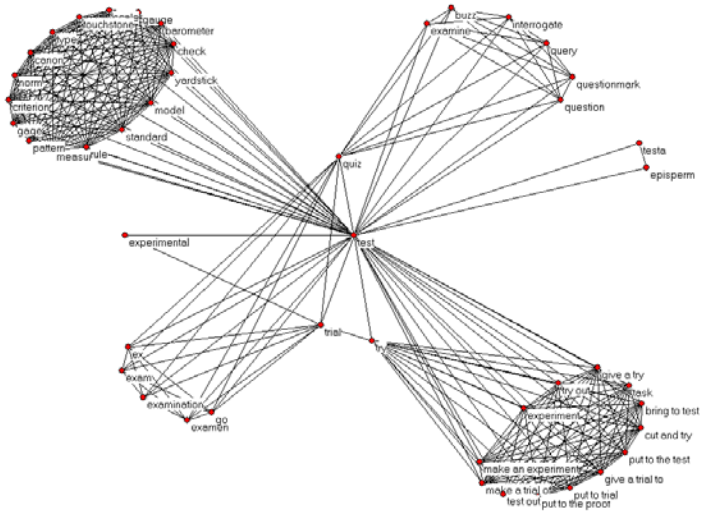




Pajek (XXL)

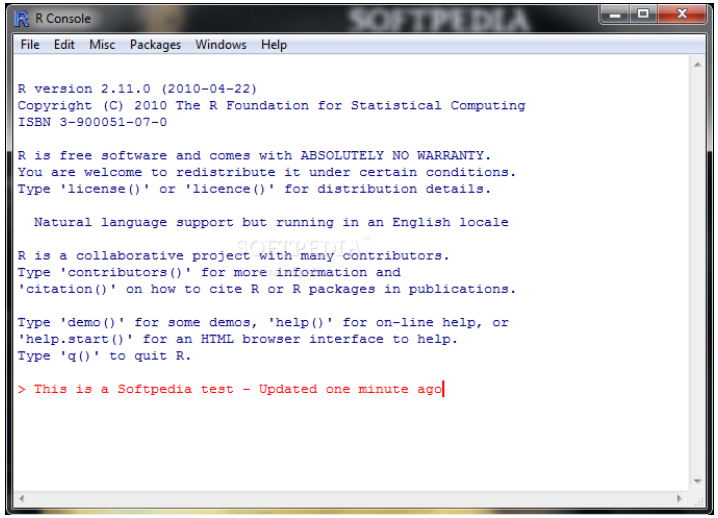
- Developed in Slovenia by Bagatelj et al.
- Free, MS Windows only
- Many options for data management and analysis
- Interface takes getting used to
- Relatively good at visualizing larger networks
- Pajek XXL for large networks
- Somewhat scriptable

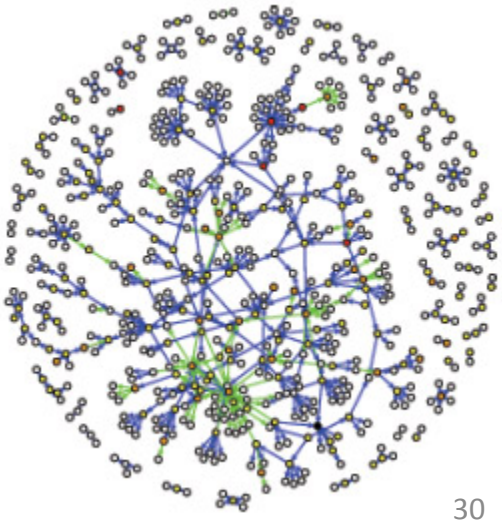




R: statnet, igraph, & RSiena packages

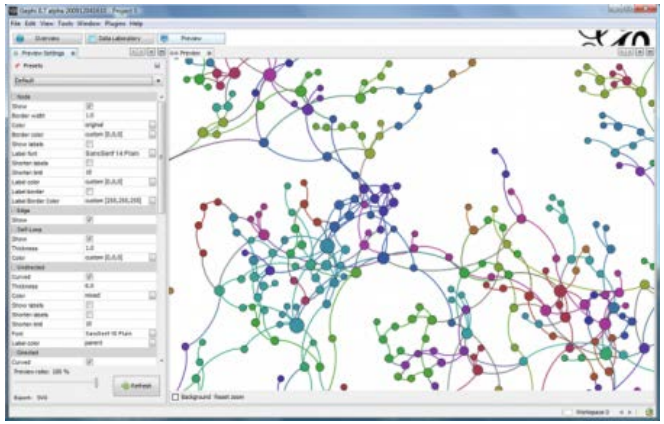
- Embedded in R language, so:
  - Very flexible
  - Handles many types of data
  - Easy access to other types of analyses
- Cross-platform
- Free (beer & speech)
- Potentially handles large networks
- RSiena for dynamic network analysis (panel data)
- Syntax **only**
- Steep(er) learning curve

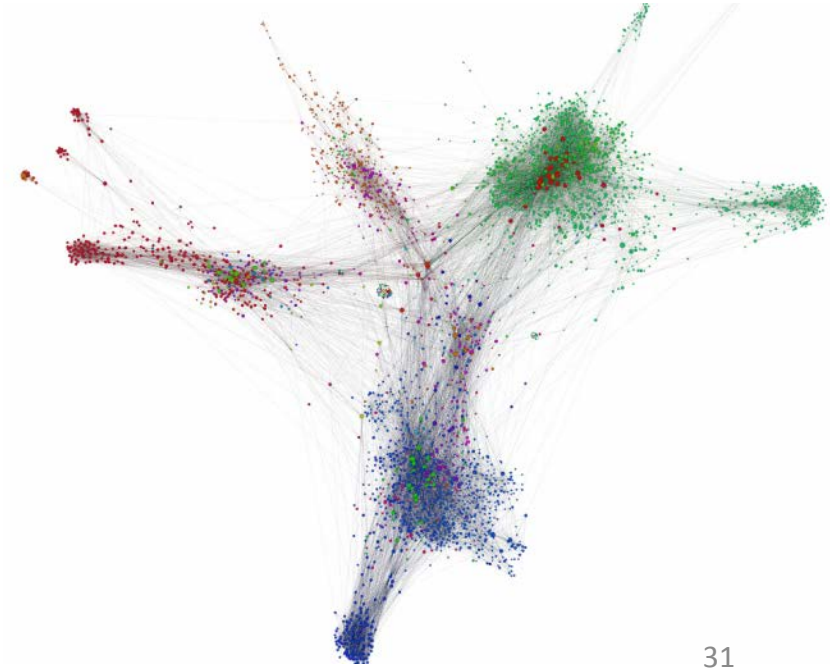




Gephi

- Relatively new
- Cross-platform (Java)
- Free (as in beer)
- Free (as in speech)
- Plug-ins available from large user community
- Good at visualization (incl animation of dynamic data)
- Analysis options relatively limited (but expanding)
- Scriptable via Python plugin





Relational data for Gephi

Attribute data (nodes table):
characteristics of individuals

ID	Male	Age
Anna	1	26
Bob	0	23
Casper	0	30
Dirk	1	22
Ernie	1	28

Relational data (edges table) :
characteristics of relations between individuals

→ **Who likes whom?**

Source	Target
Anna	Bob
Bob	Casper
Casper	Bob
Casper	Anna
Casper	Dirk
Dirk	Casper

A toy example: Knecht's highshool data

- Larger project: surveys in 128 first-grade classes in 14 Dutch highschools in 2003
- Longitudinal data: 4 waves
- Many different types of networks (name generators), attributes and behavior → great data, publicly available in DANS
- This example: 1 class, 1 wave, 30 kids, gender, homework behavior, mother's country of origin (coded)

Steps

- Import network data
- Add labels
- Visualize the network
  - Different algorithms, expand
- Filter: giant component
- Analysis:
  - Degree distribution
  - Distances
- Add actor attributes
- Exporting

Collecting network data from Twitter

- Access via Twitter API, e.g. using TwitteR package for R
- Max 9 days retrospectively
- Easy: download tweets based on search query (e.g., a hashtag), construct networks based on retweets & mentions
- More involved: downloading follower networks → requires snowballing based on set of seeds

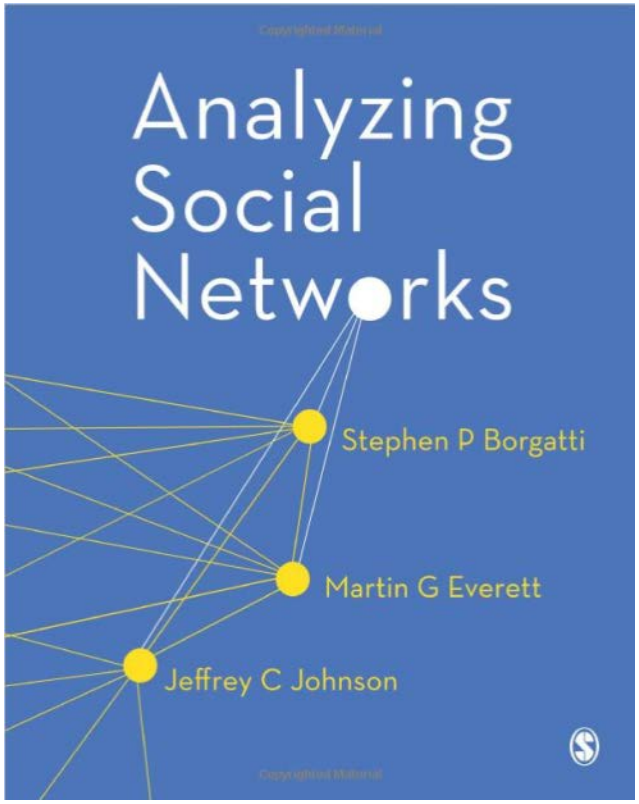
Practice data

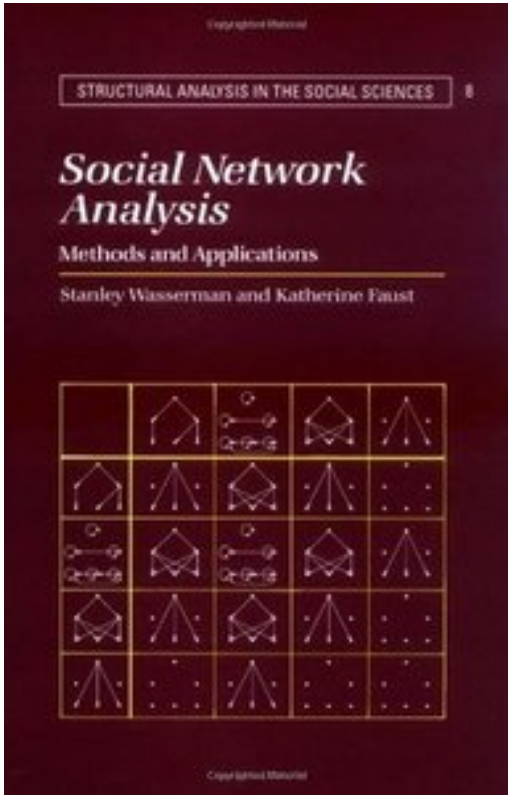
- Tweets with hashtag #kominverzet (“resist”) as coined by @geertwilderspvv (G. Wilders)
- Collected by RC during peak of the refugee crisis, from mid January 2016
- Full dataset: a few months of data, complete tweets, timestamps, sender, receiver, some sender attributes
- Here: sender-receiver, actor attribute: device OS
- Full data still waiting to be analyzed – collaborators welcome!

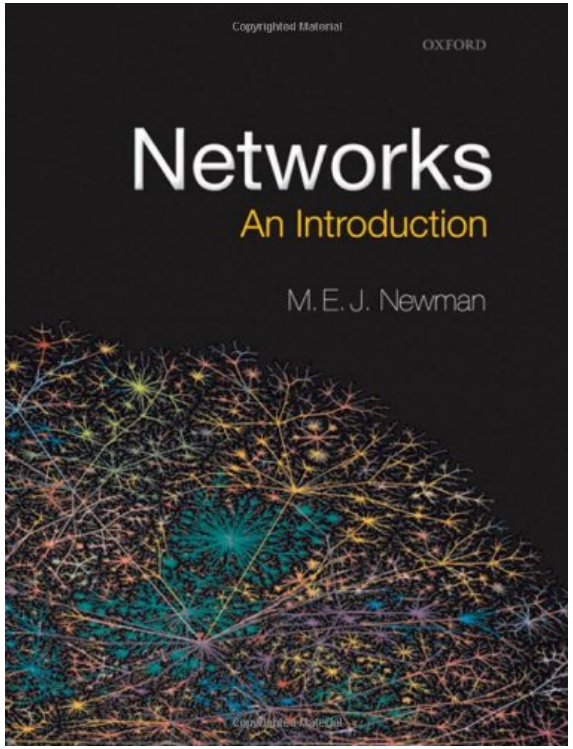
Practice: Steps

- Import network data (kominverzet_ties.txt): new project>data lab>import spreadsheet (as edges table!)
- Visualize the network
  - Try different algorithms: random, Force Atlas 2, Yifan Hu (order matters!)
- Filter giant component
- Look at degree distribution
- Add actor attributes (kominverzet_actors.txt): data lab>import spreadsheet (as nodes table!) → any pattern?
- Try setting node size by degree
- Add labels: can you make sense of the subgroups?
- Export graphic (and tables)

Further reading







(... And much more, ask me if you're interested)