Network Analysis and Modeling

Aaron Clauset
@aaronclauset
Assistant Professor of Computer Science
University of Colorado Boulder
External Faculty, Santa Fe Institute

lecture 0: what are networks and how do we talk about them?
who are network scientists?

- Physicists
- Computer Scientists
- Applied Mathematicians
- Statisticians
- Biologists
- Ecologists
- Sociologists
- Political Scientists

it’s a big community!
who are network scientists?

- Physicists
- Computer Scientists
- Applied Mathematicians
- Statisticians
- Biologists
- Ecologists
- Sociologists
- Political Scientists

it’s a big community!

- different traditions
- different tools
- different questions
who are network scientists?

Physicists
Computer Scientists
Applied Mathematicians
Statisticians
Biologists
Ecologists
Sociologists
Political Scientists

it’s a big community!
- different traditions
- different tools
- different questions

increasingly, not ONE community, but MANY, only loosely interacting communities
who are network scientists?

- Physicists: phase transitions, universality
- Computer Scientists: data / algorithm oriented, predictions
- Applied Mathematicians: dynamical systems, diff. eq.
- Statisticians: inference, consistency, covariates
- Biologists: experiments, causality, molecules
- Ecologists: observation, experiments, species
- Sociologists: individuals, differences, causality
- Political Scientists: rationality, influence, conflict
what are networks?
what are networks?

• an approach
• a mathematical representation
• provide structure to complexity
• structure above
  individuals / components
• structure below
  system / population

system / population

individuals / components
CSCI 5352 Network Analysis and Modeling:

1. develop a *network intuition* for reasoning about how structural patterns are related, and how they influence dynamics in / on networks

2. master *basic terminology and concepts*

3. master *practical tools* for analyzing / modeling structure of network data

4. build *familiarity with advanced techniques* for exploring / testing hypotheses about networks
Course schedule (roughly):

1. network basics
2. centrality measures
3. random graphs (simple)
4. configuration model
5. large-scale structure (communities, hierarchies, etc.)
6. probabilistic generative models (SBMs)
7. metadata, label and link prediction
8. spreading processes (social, biological, SI-type)
9. data wrangling + data sampling (artifacts)
10. role of statistics in hypothesis generation / testing
11. spatial networks
12. citations networks, dynamics, preferential attachment
13. temporal networks
14. student project presentations
Course webpage: http://santafe.edu/~aaronc/courses/5352/
Network data for assignments

The Colorado Index of Complex Networks (ICON)

ICON is a comprehensive index of research-quality network data sets from all domains of network science, including social, web, information, biological, ecological, connectome, transportation, and technological networks.

Entries found: 553 Networks found: 4096
lessons learned from past instances

what’s difficult:

1. students need to know many different things:
   - some probability Erdos-Renyi, configuration, calculations
   - some mathematics physics-style calculations, phase transitions
   - some statistics basic data analysis, correlations, distributions
   - some machine learning prediction, likelihoods, features, estimation algorithms
   - some programming data wrangling, coding up measures and algorithms

2. can’t teach all of these things to all types of students!
   - vast amounts of advanced material in each of these directions
   - students have little experience / intuition of what makes good science
lessons learned from past instances

what works well:

1. simple mathematical problems
   build intuition + practice with concepts

- calculate the diameter
- closeness centrality
- modularity of a line graph $Q(r)$
- betweenness of $A$
lessons learned from past instances

what works well:

2. analyze real networks
test understanding + practice with implementing methods

mean geodesics and $O(\log n)$

node centrality vs. configuration model
(when is a pattern interesting?)

attribute assortativity
lessons learned from past instances

what works well:

3. simple prediction tasks
   test intuition + run numerical experiments

label prediction via homophily

link prediction via heuristic
lessons learned from past instances

what works well:

4. simple simulations
   explore dynamics vs. structure + numerical experiments

simulate epidemics (SIR) on planted partitions
simulate Price's model
what works well:

5. team projects
   teamwork + exploring their own ideas
key takeaways
key takeaways

- **network intuition is hard to develop!**
  
  good intuition draws on many skills
  (probability, statistics, computation, causal dynamics, etc.)

- best results come from
  1. exercises to get practice with calculations
  2. practice analyzing diverse real-world networks
  3. conducting out numerical experiments & simulations

- practical tasks are a pedagogical tool (e.g., link and label prediction)

- interpreting the results requires a good intuition

- null models are key conceptual idea: *is a pattern interesting?*

- networks are fun!
1. defining a network
2. describing a network
what is a vertex?  
\[ V \] distinct objects (vertices / nodes / actors)

when are two vertices connected?  
\[ E \subseteq V \times V \]  
pairwise relations (edges / links / ties)
<table>
<thead>
<tr>
<th>network</th>
<th>vertex</th>
<th>edge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet(1)</td>
<td>computer</td>
<td>IP network adjacency</td>
</tr>
<tr>
<td>Internet(2)</td>
<td>autonomous system (ISP)</td>
<td>BGP connection</td>
</tr>
<tr>
<td>software</td>
<td>function</td>
<td>function call</td>
</tr>
<tr>
<td>World Wide Web</td>
<td>web page</td>
<td>hyperlink</td>
</tr>
<tr>
<td>documents</td>
<td>article, patent, or legal case</td>
<td>citation</td>
</tr>
<tr>
<td>power grid transmission</td>
<td>generating or relay station</td>
<td>transmission line</td>
</tr>
<tr>
<td>rail system</td>
<td>rail station</td>
<td>railroad tracks</td>
</tr>
<tr>
<td>road network(1)</td>
<td>intersection</td>
<td>pavement</td>
</tr>
<tr>
<td>road network(2)</td>
<td>named road</td>
<td>intersection</td>
</tr>
<tr>
<td>airport network</td>
<td>airport</td>
<td>non-stop flight</td>
</tr>
<tr>
<td>friendship network</td>
<td>person</td>
<td>friendship</td>
</tr>
<tr>
<td>sexual network</td>
<td>person</td>
<td>intercourse</td>
</tr>
<tr>
<td>metabolic network</td>
<td>metabolite</td>
<td>metabolic reaction</td>
</tr>
<tr>
<td>protein-interaction</td>
<td>protein</td>
<td>bonding</td>
</tr>
<tr>
<td>gene regulatory network</td>
<td>gene</td>
<td>regulatory effect</td>
</tr>
<tr>
<td>neuronal network</td>
<td>neuron</td>
<td>synapse</td>
</tr>
<tr>
<td>food web</td>
<td>species</td>
<td>predation or resource transfer</td>
</tr>
</tbody>
</table>
social networks

**vertex:** a person

**edge:** friendship, collaborations, sexual contacts, communication, authority, exchange, etc.
information networks

vertex: books, blogs, webpages, etc.

edge: citations, hyperlinks, recommendations, similarity, etc.
communication networks

**vertex:** network router, ISP, email address, mobile phone number, etc.

**edge:** exchange of information
transportation networks

**vertex:** city, airport, junction, railway station, river confluence, etc.

**edge:** physical transportation of material
biological networks

**vertex:** species, metabolic, protein, gene, neuron, etc.

**edge:** predation, chemical reaction, binding, regulation, activation, etc.

**grassland foodweb**

**core metabolism**
what’s a network?

pop quiz
what’s a network?

Andromeda galaxy
what’s a network?

cauliflower fractal
what’s a network?
diamond lattice
representing networks
a simple network

undirected
unweighted
no self-loops
a *simple* network

adjacency matrix

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

adjacency list

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>{2, 5}</td>
</tr>
<tr>
<td>2</td>
<td>{1, 3, 4}</td>
</tr>
<tr>
<td>3</td>
<td>{2, 4, 5, 6}</td>
</tr>
<tr>
<td>4</td>
<td>{2, 3}</td>
</tr>
<tr>
<td>5</td>
<td>{1, 3}</td>
</tr>
<tr>
<td>6</td>
<td>{3}</td>
</tr>
</tbody>
</table>

undirected
unweighted
no self-loops
a less simple network

- Directed edge
- Weighted edge
- Multi-edge
- Weighted node
- Self-loop
- Undirected
- Unweighted
- No self-loops
a less simple network

adjacency matrix

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>{1, 1, 2}</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>(\frac{1}{2})</td>
<td>{2, 1}</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>{2, 1}</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>{1, 1, 2}</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

adjacency list

<table>
<thead>
<tr>
<th></th>
<th>(A)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>({(5, 1), (5, 1), (5, 2)})</td>
</tr>
<tr>
<td>2</td>
<td>({(1, 1), (2, \frac{1}{2}), (3, 2), (3, 1), (4, 1)})</td>
</tr>
<tr>
<td>3</td>
<td>({(2, 2), (2, 1), (4, 2), (5, 4), (6, 4)})</td>
</tr>
<tr>
<td>4</td>
<td>({(2, 1), (3, 2)})</td>
</tr>
<tr>
<td>5</td>
<td>({(1, 1), (1, 1), (1, 2), (3, 4)})</td>
</tr>
<tr>
<td>6</td>
<td>({(3, 4), (6, 2)})</td>
</tr>
</tbody>
</table>
directed networks

\[ A_{ij} \neq A_{ji} \]

citation networks
foodwebs*
epidemiological
others?
directed acyclic graph
directed graph
WWW
friendship?
flows of goods, information
economic exchange
dominance
neuronal
transcription
time travelers
bipartite networks

Authors & papers
Actors & movies/scenes
Musicians & albums
People & online groups
People & corporate boards

People & locations (checkins)
Metabolites & reactions
Genes & substrings
Words & documents
Plants & pollinators

No within-type edges
bipartite networks

no within-type edges

one type only

authors & papers
actors & movies/scenes
musicians & albums
people & online groups
people & corporate boards

people & locations (checkins)
metabolites & reactions
genes & substrings
words & documents
plants & pollinators
temporal networks

any network over time

- discrete time (snapshots), edges \((i, j, t)\)
- continuous time, edges \((i, j, t_s, \Delta t)\)
describing networks

what networks look like
describing networks

what networks look like

questions:
- how are the edges organized?
- how do vertices differ?
- does network location matter?
- are there underlying patterns?

what we want to know
- what processes shape these networks?
- how can we tell?
describing networks

a first step: describe its features
describing networks

a first step: describe its features

\[ f : G \rightarrow \{ x_1, \ldots, x_k \} \]

- degree distributions
- short-loop density (triangles, etc.)
- shortest paths (diameter, etc.)
- vertex positions
- correlations between these
describing networks

a first step: describe its features

\[ f : \text{object} \rightarrow \{x_1, \ldots, x_k\} \]
describing networks

a first step: describe its features

\[ f : \text{object} \rightarrow \{ \theta_1, \ldots, \theta_k \} \]

- physical dimensions
- material density, composition
- radius of gyration
- correlations between these

helpful for exploration, but not what we want…
what we want: understand its structure

\[ f : \text{object} \rightarrow \{\theta_1, \ldots, \theta_k\} \]

• what are the fundamental parts?
• how are these parts organized?
• where are the degrees of freedom \( \tilde{\theta} \)?
• how can we define an abstract class?
• structure — dynamics — function?

what does local-level structure look like?
what does large-scale structure look like?
how does structure constrain function?