Statistical and computational challenges in networks and cybersecurity

Hugh Chipman

Acadia University

June 12, 2015

Statistical and computational challenges in networks and cybersecurity

May 4-8, 2015, Centre de recherches mathmatiques, Montreal

Thematic Program on
Statistical Inference, Learning,
and Models for Big Data

January to June, 2015

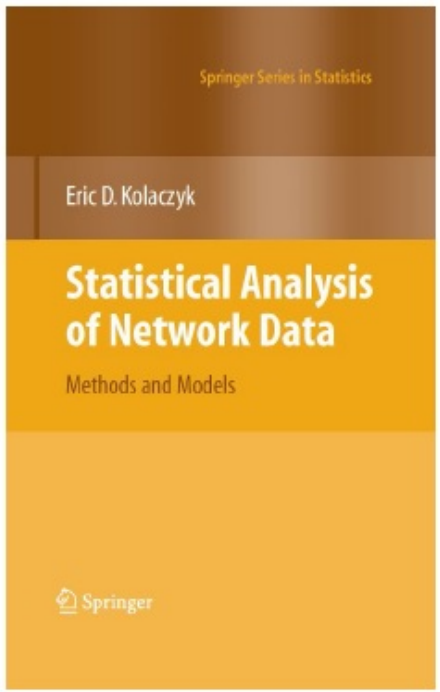
FIELDS

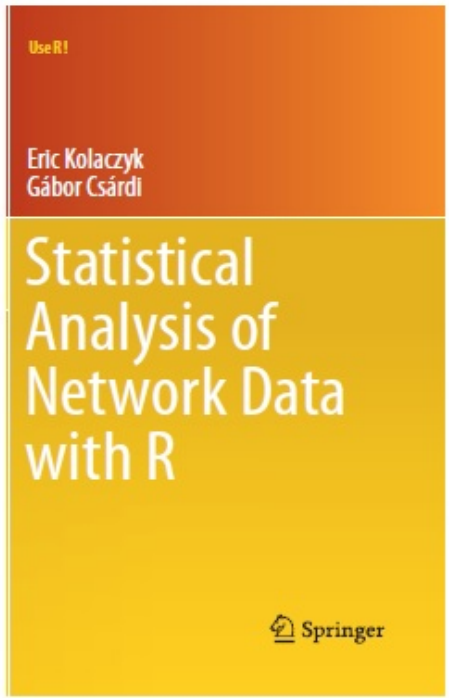
- About 60 participants
- 2-day short course by Eric Kolaczyk (Boston University) on "Statistical Analysis of Network Data",
- followed by 2.5 days of research presentations

Short course: "Statistical Analysis of Network Data"

1-hour overview: watch Eric's talk at the January Opening Conference (Fields Online / Video Archive, or `http://goo.gl/XhX09s`)

Two companion books provide a very good introduction / overview to the area, including R code using the `igraph` package.



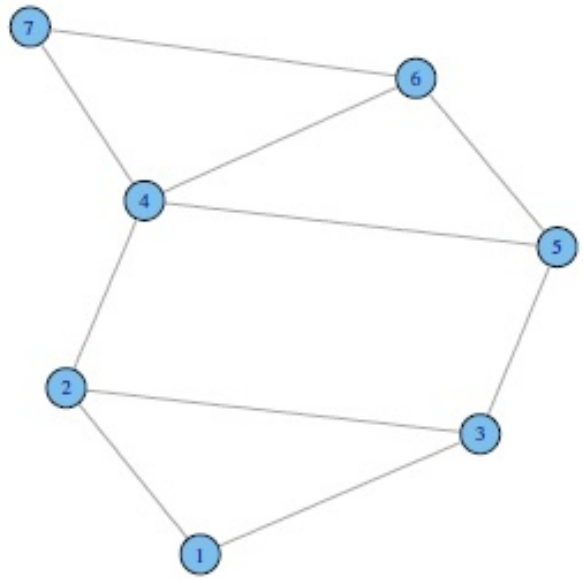


Background on Graphs

A graph $G = (V, E)$ is a mathematical structure with sets:

- V of *vertices* (also called *nodes*)
- E of *edges* (also called *links*)

where elements of E are unordered pairs $\{u, v\}$ of distinct vertices $u, v \in V$

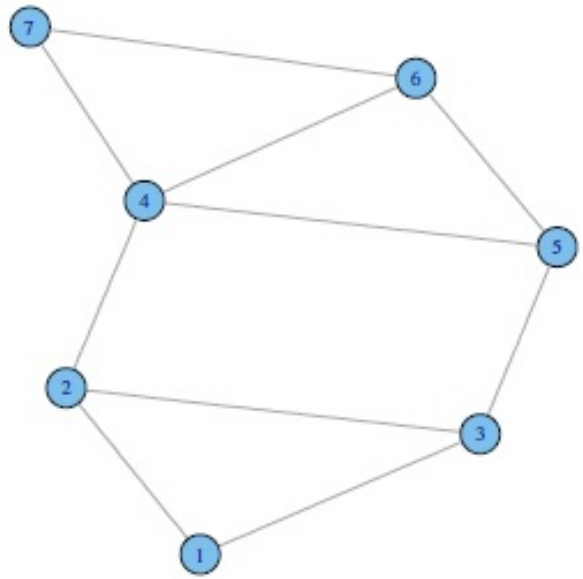


Graphs can be directed or undirected.

Directed graph: A graph G with each edge having an ordering, i.e., $\{u, v\} \neq \{v, u\}$.

Background on Graphs, continued

Graphs can be represented by several data structures:



Adjacency matrix:

```
  1 2 3 4 5 6 7
1 . 1 1 . . . .
2 1 . 1 1 . . .
3 1 1 . . 1 . .
4 . 1 . . 1 1 1
5 . . 1 1 . 1 .
6 . . . 1 1 . 1
7 . . . . 1 1 .
```

Adjacency list:

```
1: 2, 3
2: 1, 3, 4
3: 1, 2, 5
4: 2, 5, 6, 7
5: 3, 4, 6
6: 4, 5, 7
7: 5, 6
```

Edge list:

```
1--2, 1--3, 2--3, 2--4, 3--5, 4--5, 4--6, 4--7, 5--6, 6--7
```

Background on Graphs, continued

- Weighted graph: $G = (V, E)$, and for every edge in E, we have a non-negative weight.
- Dynamic graph: $G(t) = (V(t), E(t))$, allowing the set of vertices and edges to vary over time t.
- Decorated graphs: Covariates or "attributes" associated with
  - Vertices (e.g. gender of actors in a social network)
  - Edges (e.g. number of emails sent between two actors)

Data examples: Social Network

Paramjit Gill discussed a bullying network among 3rd and 4th grade students (21 students).

- Vertices = students
- Edges = bullying relation
- $A \to B$ means "A bullies B"
- ...actually more complicated, data involves assessment of who bullies who by other students.
- Objective: Fit a model characterizing the tendency to have more (less) ties than you'd expect in a random graph.

Data Examples: Technological network

Leman Akoglu described a network traffic flow dataset

- vertices = hosts (125 hosts)
- network flow measured along edges (352 edges)
- this is *dynamic data*: flow changes over time, giving us "snapshots" of the network over time (1304 snapshots)
- Objective: identify anomalies

Data Examples: Biological Networks

John Conroy described a study involving mapping the "connectome" of brains.

- Nodes = locations in brain
- An edge is present if there is a sufficiently high correlation of activation.
- Many biological network examples involve data where the edges are not directly observed.
- Objective: Cluster regions of the brain in terms of connectivity.

Data Examples: Online social network

Carey Priebe discussed the Friendster network.

- Online social network
- $\sim$ 65 million vertices
- $\sim$ 1.8 billion edges
- self-identified groups are "ground truth"
- Objective is to cluster vertices into groups

Challenges and General Observations

Challenges:

- relational aspect to the data;
- complex statistical dependencies;
- high-dimensional and often massive in quantity.
- "sample size of 1"

General observations:

- Many disciplines study graph data: CS, Mathematics, Statistics, Physics, ...
- Statistically there have been success stories, but some basic statistical questions remain unanswered.

Areas of graph analytics

In his short course, Eric Kolaczyk divided things up into 5 main areas:

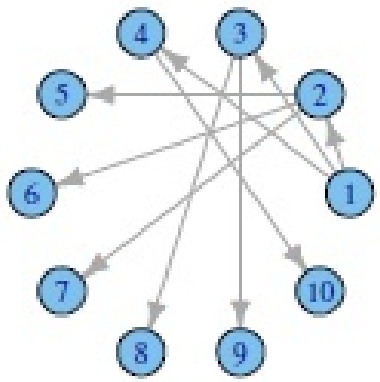
- Visualization (Mapping)
- Descriptive statistics (Characterization)
- Sampling
- Inference
- Modeling

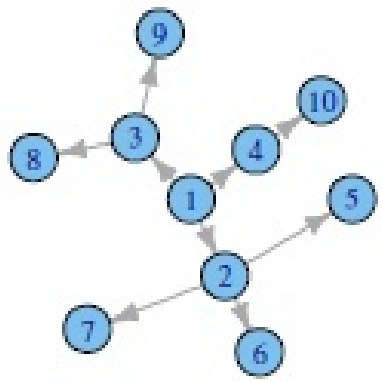
I'll organize my comments mostly under these headings, with reference to research talks presented at the conference.

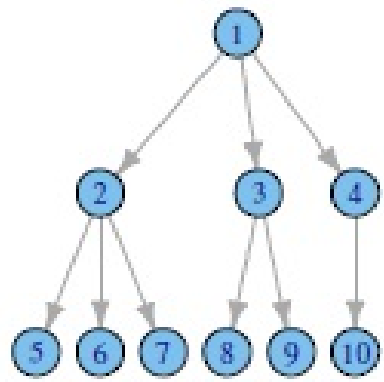
Visualization

Basic objective: draw the graph in a visually appealing way.

Layout matters! Below are three views of the same graph.





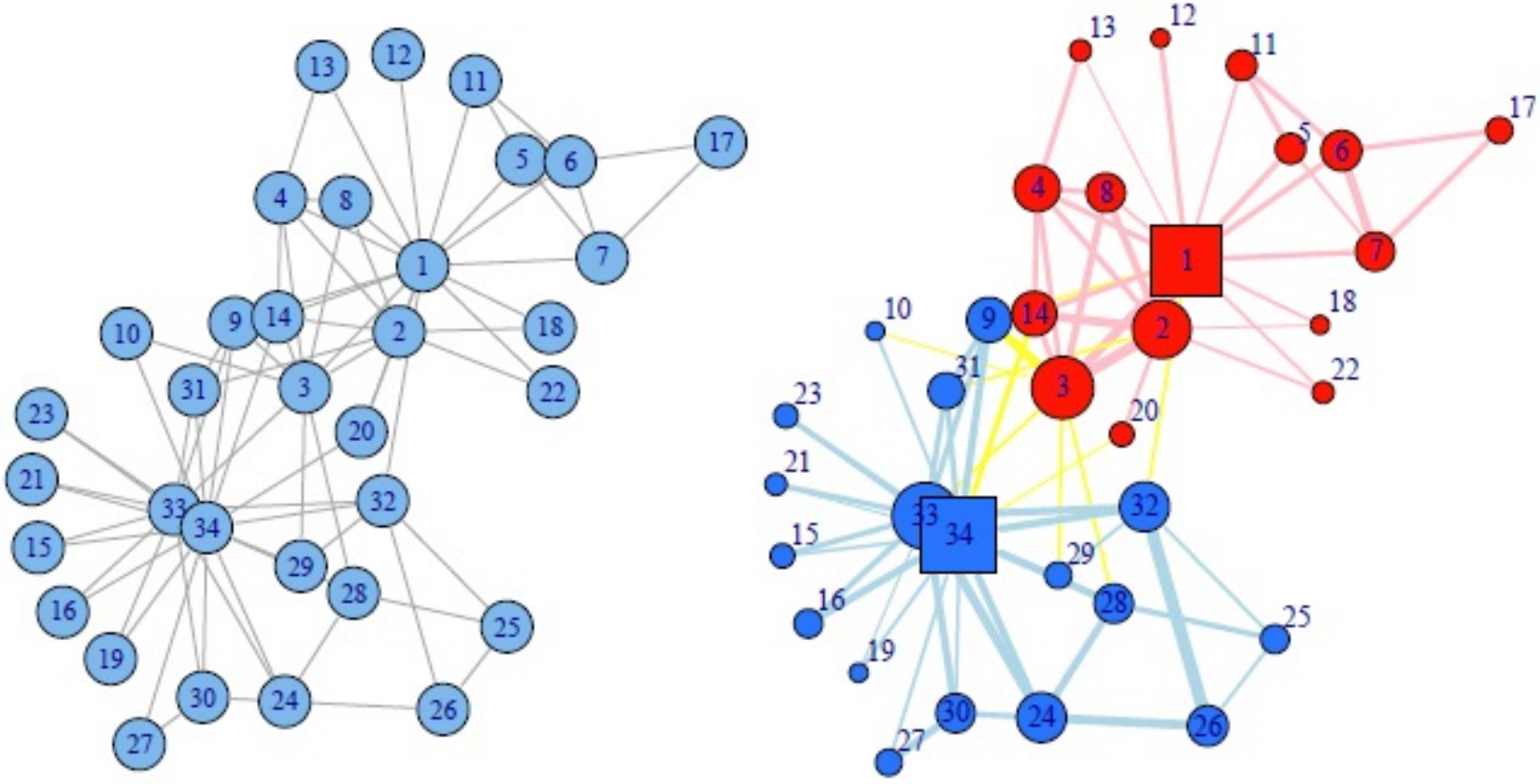


Many algorithms for layout exist, often by analogy to energy / forces between vertices.

Visualization

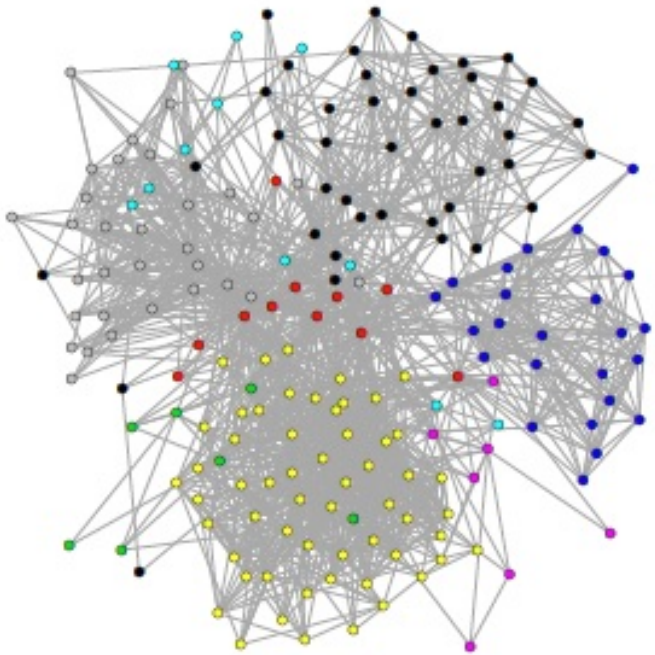
Decoration of edges or vertices can help visualize.

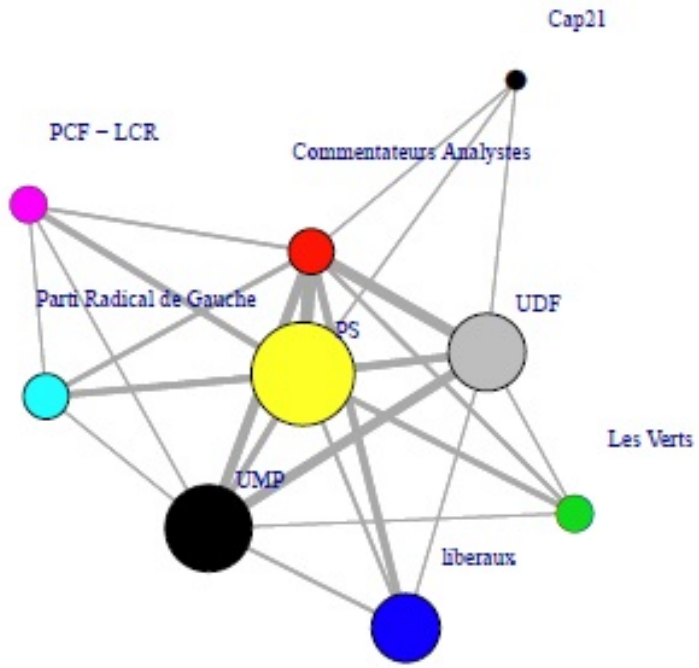
Example below: Karate network with two "factions" and their leaders, using different colors for between and within group edges.



Visualization

Large networks present a challenge. Tricks include shifting the "focus" to a specific part of the network or (below for a French political blog network) clustering the vertices and then drawing a graph for the clusters.





Visualization

Dynamic networks are also a challenge for visualization.

- Steve Thompson demonstrated a dynamic visualization of the spread of a disease over a network.
- Visualizing a simulation provoked many interesting questions about the underlying model.

Characterization

("descriptive statistics for graphs")

A wide range of numeric summaries exist for graphs. For simplicitly I'll mention just a few.

They can be at the level of an individual element:

- e.g. "degree" of a vertex = number of edges.
- proportion of closed triangles (if A and B are friends, and B and C are friends, then are A and C friends?)
- closeness centrality $c(v) = \frac{1}{\sum_{u \neq v} \text{dist}(v,u)}$, where $\text{dist}(v, u)$ is the shortest path distance between vertices u, v.

Or they can be at the level of a graph:

- average degree
- degree distribution

Characterization

Computation:

- Many summaries can be calculated from adjacency matrix A
  - Degree = row sum
- Some computing (e.g., shortest path) can be time-consuming.

Sampling

Typical network analysis approach:

- Interested in a system of elements and their interactions
- Collect elements and relations among them
- Represent data as network
- Characterize properties of network

But, what are we interested in?

1. The properties of the actual network we collected, or
2. An underlying "true" network, which our collected data represent

for #2, statistical sampling theory is relevant.

Sampling: Notation

Let

- $G = (V, E)$ be a network graph
- $G^* = (V^*, E^*)$ be a sampled subgraph of G
- $\eta(G)$ be a summary characteristic of graph G

Goal: accurate estimation of $\eta = \eta(G)$ by some $\hat{\eta} = \hat{\eta}(G^*)$.

For example, we might be interested in average degree of the graph.

Sampling example

Illustrative example from Eric's presentation: real data from biological network.

Goal to estimate average degree

Entire graph has average degree 12.115

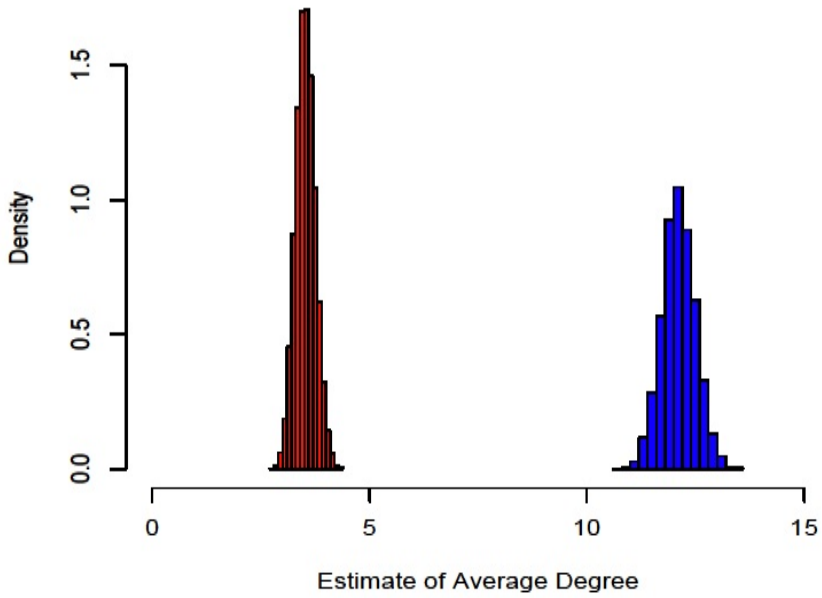
Two sampling strategies:

1. Sample n vertices and all their edges.
2. Sample n vertices and only edge between them.

Note: #2 will omit some edges that are included in #1 (e.g., edges from a sampled vertex to an unsampled vertex).

Sampling example, continued

Estimate average degree using $\hat{\eta}(G^*) = \eta(G^*) =$ average degree of sampled network. This is a "plug in estimator".



1. (blue) Sample n vertices and all their edges.
2. (red) Sample n vertices and only edges between them.

True average degree $= 12.115$

Red plug-in estimator underestimates average degree, since the number of sampled edges is biased downward for every sampled vertex.

Sampling

The accuracy of the estimator depends on how the data were collected.

How do you correct such problems?

General approach: adjust for the sampling design.

Horvitz-Thompson estimator utilizes the inclusion probabilities of elements (vertices or edges)

- Design-based inference,
- General approach that can be used for many network statistics.
- Inferential statements possible.
- Inclusion probabilities (including joint inclusion probabilities) may be difficult to compute.

Sampling Methods

Commonly used methods to sample a network:

1. Sample vertices, then edges between sampled vertices.
2. Sample edges, and all vertices attached to them
3. Snowball sampling: sample vertices, then follow all their edges out to vertices, and repeat these "waves"
4. Path sampling: sample a set of source vertices and target vertices. For each source/target pair, sample a path between them.

As you can imagine, the sampling method affects the form of the estimator used.

Inference

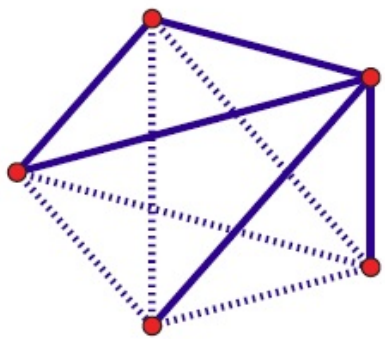
In discussing Sampling, we've seen that inferential statements may be possible based on samples.

Slava Lyubchich discussed a bootstrap-based method of inference (e.g., for mean degree):

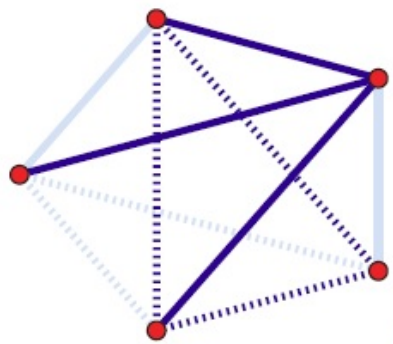
- Bootstrap developed that samples small "patches" of subgraph via snowball sampling
- HT estimator used for inference.
- Two bootstrap parameters must be chosen.
- Seems to give good inference for a wide variety of networks.

Inference

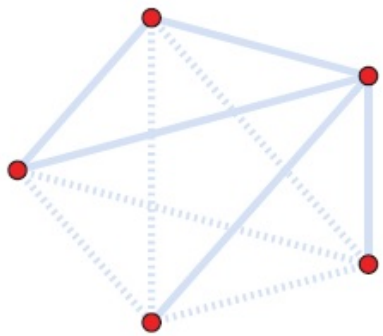
Eric Kolaczyk defined network inference as follows:
Problem Statement: Given measurements x_i of attributes at some or all vertices $i \in V$, and/or observations y_{ij} of 'edge status' for some vertex pairs $\{i, j\} \in V^{(2)}$, infer the topology of $G = (V, E)$.



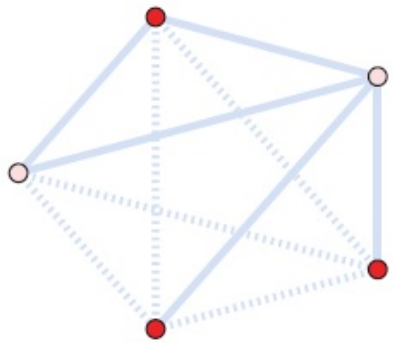
Original Network



Link Prediction



Assoc. Network Inf.



Tomography

Modelling

Statisticians demand a great deal of their modeling:

1. theoretically plausible
2. estimable from data
3. computationally feasible estimation strategies
4. quantification of uncertainty in estimates (e.g., confidence intervals)
5. assessment of goodness-of-fit
6. understanding of the statistical properties of the overall procedure

Modelling

Eric Kolaczyk characterized two main areas of modelling:

1. we observe a network G (and possibly attributes X) and we wish to model G (and X)
   $\Rightarrow$ *describe the network with a model.*
2. we observe the network G, but lack some or all of the attributes X, and we wish to infer X
   $\Rightarrow$ *Use network to predict other variables.*

Nearly all the talks focused on problem #1:

- Stochastic block models
- clustering

Steve Thompson's talk about modelling the spread of a disease on a network was closer to #2: A model for a process on a network.

Stochastic Blockmodel

Stochastic block models explicitly parameterize the notion of groups/modules, labeled (say) $1, \ldots, Q$, with different rates of connections between/within.

More specifically, this is a generative model, where

- Each vertex independently belongs to a group q with probability α_q, where $\sum_{q=1}^{Q} \alpha_q = 1$.
- For vertices $i, j \in V$, with i in group q and j in group r, the probability that we have an edge between i and j is π_{qr}.

This is, effectively, a mixture of classical random graphs.

Stochastic Blockmodel

Perhaps easier to explain in terms of the process of generating a graph.

1. For each vertex, pick a group label at random using probabilities $\alpha_1, ..., \alpha_Q$.
2. For each potential edge between vertices i and j, belonging to groups q and r, respectively, carry out a Bernoulli trial with probability π_{qr}. 0 = no edge, 1=edge.

It's a probabilistic clustering model, with clusters defined by differing rates of between- and within-group communication.

It can be modified to simply identify groups with high within-group connections (make π_{qr} = some small value ϵ for $q \neq r$).

Estimation via EM, MCMC, or Variational Bayes

Stochastic Blockmodel

Blei described an implementation of a *Mixed Membership Stochastic Blockmodel*, in which each vertex can belong to multiple groups.

- Novelty in his approach was the ability to scale to large (1,000,000 vertex) networks.
- Another extension is to have Poisson counts on the edges $\Rightarrow$ seems to do better with varying degree vertices.

Carey Priebe discussed a hierarchical blockmodel, in which the same recurring graph structures ("motifs") were re-used at different levels.

Mu Zhu discussed a continuous-time stochastic blockmodel for a transactional network: basketball games (union of continuous time multistate models and stochastic blockmodels).

Algorithmic models

The stochastic blockmodel and its variants are built on a probability model. Other approaches to clustering can be more algorithmic.

- Spectral methods look at the eigenvectors of the adjacency matrix or other matrices (e.g. "Graph Laplacian", defined as $D - A$, with $A =$ adjacency matrix and $D =$ diagonal matrix of degrees.)
- Danny Dunlavy discussed extensions of spectral methods to time varying networks, using tensors to "stack up" networks observed over time.
- Geoffrey Sanders looked at numerical analysis in accurate computation of spectral decompositions for clustering.
- Francois Theberge took a very fast and scalable hierarchical clustering algorithm and generated an "ensemble" version. (Find multiple clusterings, reweight original edges, then use weighted graph to find a more stable clustering.)

Other models

George Michalidis and Leman Akoglu both considered sequences of graphs measured over time

- Changepoint detection
- Akoglu: algorithmic approach using ensembles of models
- Michalidis: leverages existing work on changepoints, combined with a statistical graph model.

David Banks considered enriching text data (wikipedia) with network information on links, combining topic models with network analysis

Erico de Souza had interesting time series geolocated data on cell phone calls, it would be interesting to include social network information.

Conclusions

- Network modelling has been around for several decades
- ... but some areas are still in their infancy (e.g. goodness-of-fit diagnostics.)
- Diverse groups tackling research problems
- ... and perhaps breakthroughs will happen with collaborations?
- Statistically, the most headway has been made in areas where a connection can be made to existing statistical technology.