Block 8: Network inference

ELEC 573: Network Science and Analytics

Santiago Segarra

Electrical and Computer Engineering
Rice University
segarra@rice.edu

Fall 2021

You are here

Wk.	Date	Topic	HW	Project
1	23-Aug	Introduction to course	HW0 out	
2	30-Aug	Graph theory	HW0 solutions posted	
3	6-Sep	LABOR DAY (no class)	HW1 out	
4	13-Sep	Centrality measures / Community detection		
5	20-Sep	Community detection		
6	27-Sep	Signal Processing and Deep learning for graphs	HW1 due	
7	4-Oct	Signal Processing and Deep learning for graphs	HW2 out	
8	11-Oct	FALL BREAK (no class)		
9	18-Oct	Network models	HW2 due	
10	25-Oct	Network models	HW3 out	Project proposal due
11	1-Nov	Inference of network topologies and features		
12	8-Nov	Inference of network topologies and features	HW3 due	
13	15-Nov	Inference of network topologies and features		
14	22-Nov	Epidemics		Project progress report
15	29-Nov	Inference of network processes		
	13-Dec	Project presentation (video recording) and final report due		

Network sampling

Network sampling and challenges

Background on statistical sampling theory

Graph sampling designs

Estimation of network totals, group size, and degree distributions

Network topology inference problems

Link prediction

Inference of association networks

Tomographic network topology inference

Sampling networks

- Measurements often gathered only from a portion of a complex system
  - Ex: social study of high-school class vs. large corporation, Internet
  - Graph $\rightarrow$ sample from a larger underlying network

Sampling networks

- Measurements often gathered only from a portion of a complex system
  - Ex: social study of high-school class vs. large corporation, Internet
  - Graph $\rightarrow$ sample from a larger underlying network
- Goal: use sampled network data to infer properties of the whole system
  - Approach using principles of statistical sampling theory
- Sampling in network contexts introduces various potential challenges

Sampling networks

- Measurements often gathered only from a portion of a complex system
  - Ex: social study of high-school class vs. large corporation, Internet
  - Graph $\rightarrow$ sample from a larger underlying network
- Goal: use sampled network data to infer properties of the whole system
  - Approach using principles of statistical sampling theory
- Sampling in network contexts introduces various potential challenges

System under study
$G(V, E)$
Population graph

$\xrightarrow{\textit{Random Procedure}}$

Available measurements
$G^*(V^*, E^*)$
Sampled graph

- G^* often a subgraph of G (i.e., $V^* \subseteq V$, $E^* \subseteq E$), but may not be

The fundamental problem

- Suppose a given graph characteristic or summary $\eta(G)$ is of interest
  - Ex: order N_v, size N_e, degree d_v, clustering coefficient $\mathrm{cl}(G)$, ...
- Typically impossible to recover $\eta(G)$ exactly from G^*

  $\Rightarrow$ Q: Can we still form a useful estimate $\hat{\eta} = \hat{\eta}(G^*)$ of $\eta(G)$?

The fundamental problem

- Suppose a given graph characteristic or summary $\eta(G)$ is of interest
  - Ex: order N_v, size N_e, degree d_v, clustering coefficient $\mathrm{cl}(G)$, ...
- Typically impossible to recover $\eta(G)$ exactly from G^*

  $\Rightarrow$ Q: Can we still form a useful estimate $\hat{\eta} = \hat{\eta}(G^*)$ of $\eta(G)$?
- Plug-in estimator $\hat{\eta} := \eta(G^*)$
  - Boils down to computing the characteristic of interest in G^*
  - Many familiar estimators in statistical practice are of this type

    Ex: sample means, standard deviations, covariances, quantiles...

The fundamental problem

- Suppose a given graph characteristic or summary $\eta(G)$ is of interest
  - Ex: order N_v, size N_e, degree d_v, clustering coefficient $\mathrm{cl}(G)$, ...
- Typically impossible to recover $\eta(G)$ exactly from G^*

  $\Rightarrow$ Q: Can we still form a useful estimate $\hat{\eta} = \hat{\eta}(G^*)$ of $\eta(G)$?
- Plug-in estimator $\hat{\eta} := \eta(G^*)$
  - Boils down to computing the characteristic of interest in G^*
  - Many familiar estimators in statistical practice are of this type

    Ex: sample means, standard deviations, covariances, quantiles...
- Oftentimes $\eta(G^*)$ is a poor representation of $\eta(G)$

Example: Estimating average degree

- Let $G(V, E)$ be a network of protein interactions in yeast
  $\Rightarrow$ Characteristic of interest is average degree

$$\eta(G) = \frac{1}{N_v} \sum_{i \in V} d_i$$

- Here $N_v = 5,151$, $N_e = 31,201 \Rightarrow \eta(G) = 12.115$

Example: Estimating average degree

- Let $G(V, E)$ be a network of protein interactions in yeast
  - $\Rightarrow$ Characteristic of interest is average degree

$$\eta(G) = \frac{1}{N_v} \sum_{i \in V} d_i$$

- Here $N_v = 5,151$, $N_e = 31,201 \Rightarrow \eta(G) = 12.115$
- Consider two sampling designs to obtain G^*
  - First sample n vertices $V^* = \{i_1, \ldots, i_n\}$ without replacement
  - Design 1: For each $i \in V^*$, observe incident edges $(i, j) \in E$
  - Design 2: Observe edge (i, j) only when both $i, j \in V^*$

Example: Estimating average degree

- Let $G(V, E)$ be a network of protein interactions in yeast
  - $\Rightarrow$ Characteristic of interest is average degree

$$\eta(G) = \frac{1}{N_v} \sum_{i \in V} d_i$$

- Here $N_v = 5,151$, $N_e = 31,201 \Rightarrow \eta(G) = 12.115$
- Consider two sampling designs to obtain G^*
  - First sample n vertices $V^* = \{i_1, \ldots, i_n\}$ without replacement
  - Design 1: For each $i \in V^*$, observe incident edges $(i, j) \in E$
  - Design 2: Observe edge (i, j) only when both $i, j \in V^*$
- Estimate $\eta(G)$ by averaging the observed degree sequence $\{d_i^*\}_{i \in V^*}$

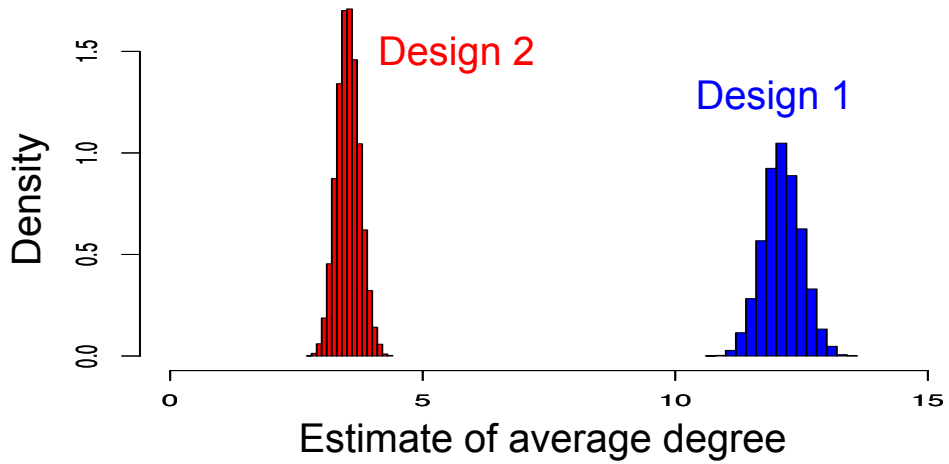
$$\eta(G^*) = \frac{1}{n} \sum_{i \in V^*} d_i^*$$

Example: Estimating average degree (cont.)

- Random sample of $n = 1,500$ vertices, Designs 1 and 2 for edges
  $\Rightarrow$ Process repeated for 10,000 trials $\Rightarrow$ histogram of $\eta(G^*)$

Example: Estimating average degree (cont.)

- Random sample of $n = 1,500$ vertices, Designs 1 and 2 for edges
  $\Rightarrow$ Process repeated for 10,000 trials $\Rightarrow$ histogram of $\eta(G^*)$



- Under-estimate $\eta(G)$ for Design 2, but Design 1 on target. Why?
  - Design 1: sample vertex degree explicitly, i.e., $d_i^* = d_i$
  - Design 2: (implicitly) sample vertex degree with bias, i.e., $d_i^* \approx \frac{n}{N_v} d_i$

Improving estimation accuracy

- In order to do better we need to incorporate the effects of
  - $\Rightarrow$ Random sampling; and/or
  - $\Rightarrow$ Measurement error
- Sampling design, topology of G, nature of $\eta(\cdot)$ all critical

Improving estimation accuracy

- In order to do better we need to incorporate the effects of
  - $\Rightarrow$ Random sampling; and/or
  - $\Rightarrow$ Measurement error
- Sampling design, topology of G, nature of $\eta(\cdot)$ all critical
- Model-based inference $\rightarrow$ Likelihood-based and Bayesian paradigms
- Design-based methods $\rightarrow$ Statistical sampling theory
  - Assume observations made without measurement error
  - Only source of randomness $\rightarrow$ sampling procedure

Improving estimation accuracy

- In order to do better we need to incorporate the effects of
  - $\Rightarrow$ Random sampling; and/or
  - $\Rightarrow$ Measurement error
- Sampling design, topology of G, nature of $\eta(\cdot)$ all critical
- Model-based inference $\rightarrow$ Likelihood-based and Bayesian paradigms
- Design-based methods $\rightarrow$ Statistical sampling theory
  - Assume observations made without measurement error
  - Only source of randomness $\rightarrow$ sampling procedure
- Ex: Estimating average degree
  - Under Design 2 the estimate is biased, with mean of only 3.528
  - Adjusting $\eta(G^*)$ upward by a factor $\frac{N_v}{n} = 3.434$ yields 12,115
- Will see how statistical sampling theory justifies this correction

Background

Network sampling and challenges

Background on statistical sampling theory

Graph sampling designs

Estimation of network totals, group size, and degree distributions

Network topology inference problems

Link prediction

Inference of association networks

Tomographic network topology inference

Statistical sampling theory

- Suppose we have a population $\mathcal{U} = \{1, \ldots, N_u\}$ of N_u units
  - Ex: People, animals, objects, vertices, ...
- A value y_i is associated with each unit $i \in \mathcal{U}$
  - Ex: Height, age, gender, infected, membership, ...

Statistical sampling theory

- Suppose we have a population $\mathcal{U} = \{1, \ldots, N_u\}$ of N_u units
  - Ex: People, animals, objects, vertices, ...
- A value y_i is associated with each unit $i \in \mathcal{U}$
  - Ex: Height, age, gender, infected, membership, ...
- Typical interest in the population **totals** τ and **averages** μ

$$\tau := \sum_{i \in \mathcal{U}} y_i \quad \text{and} \quad \mu := \frac{1}{N_u} \sum_{i \in \mathcal{U}} y_i = \frac{1}{N_u} \tau$$

Statistical sampling theory

- Suppose we have a population $\mathcal{U} = \{1, \ldots, N_u\}$ of N_u units
  - Ex: People, animals, objects, vertices, ...
- A value y_i is associated with each unit $i \in \mathcal{U}$
  - Ex: Height, age, gender, infected, membership, ...
- Typical interest in the population **totals** τ and **averages** μ

$$\tau := \sum_{i \in \mathcal{U}} y_i \quad \text{and} \quad \mu := \frac{1}{N_u} \sum_{i \in \mathcal{U}} y_i = \frac{1}{N_u} \tau$$

- Basic sampling theory paradigm oriented around these steps:

  S1: Randomly sample n units $\mathcal{S} = \{i_1, \ldots, i_n\}$ from $\mathcal{U}$

  S2: Observe the value y_{i_k} for $k = 1, \ldots, n$

  S3: Form an unbiased estimator $\hat{\mu}$ of μ, i.e., $\mathbb{E}[\hat{\mu}] = \mu$

  S4: Evaluate or estimate the variance $\text{var}[\hat{\mu}]$

Inclusion probabilities

- **Def:** For given sampling design, the inclusion probability π_i of unit i is

$$\pi_i := \mathrm{P}\left[\text{unit } i \text{ belongs in the sample } \mathcal{S}\right]$$

Inclusion probabilities

- **Def:** For given sampling design, the inclusion probability π_i of unit i is

$$\pi_i := \mathrm{P}\left[\text{unit } i \text{ belongs in the sample } \mathcal{S}\right]$$

- Simple random sampling (SRS): n units sampled uniformly form $\mathcal{U}$

  Without replacement: i_1 chosen from $\mathcal{U}$, i_2 from $\mathcal{U} \setminus \{i_1\}$, and so on

  $\Rightarrow$ There are $\binom{N_u}{n}$ such possible samples of size n

  $\Rightarrow$ There are $\binom{N_u-1}{n-1}$ samples which include a given unit i

Inclusion probabilities

- **Def:** For given sampling design, the inclusion probability π_i of unit i is

$$\pi_i := \mathrm{P}\left[\text{unit } i \text{ belongs in the sample } \mathcal{S}\right]$$

- Simple random sampling (SRS): n units sampled uniformly form $\mathcal{U}$

  Without replacement: i_1 chosen from $\mathcal{U}$, i_2 from $\mathcal{U} \setminus \{i_1\}$, and so on

  $\Rightarrow$ There are $\binom{N_u}{n}$ such possible samples of size n

  $\Rightarrow$ There are $\binom{N_u-1}{n-1}$ samples which include a given unit i

- The inclusion probability is

$$\pi_i = \frac{\binom{N_u-1}{n-1}}{\binom{N_u}{n}} = \frac{n}{N_u}$$

Sample mean estimator

- Definition of sample mean estimator

$$\hat{\mu} = \frac{1}{n} \sum_{i \in \mathcal{S}} y_i$$

Sample mean estimator

- Definition of sample mean estimator

$$\hat{\mu} = \frac{1}{n} \sum_{i \in \mathcal{S}} y_i$$

- Using indicator RVs $\mathbb{I}\{i \in \mathcal{S}\}$ for $i \in \mathcal{U}$, where $\mathbb{E}[\mathbb{I}\{i \in \mathcal{S}\}] = \pi_i$

$$\Rightarrow \mathbb{E}[\hat{\mu}] = \mathbb{E}\left[\frac{1}{n} \sum_{i \in \mathcal{S}} y_i\right] = \mathbb{E}\left[\frac{1}{n} \sum_{i=1}^{N_u} y_i \mathbb{I}\{i \in \mathcal{S}\}\right]$$

$$= \frac{1}{n} \sum_{i=1}^{N_u} y_i \mathbb{E}[\mathbb{I}\{i \in \mathcal{S}\}] = \frac{1}{n} \sum_{i=1}^{N_u} y_i \pi_i$$

Sample mean estimator

- Definition of sample mean estimator

$$\hat{\mu} = \frac{1}{n} \sum_{i \in \mathcal{S}} y_i$$

- Using indicator RVs $\mathbb{I}\{i \in \mathcal{S}\}$ for $i \in \mathcal{U}$, where $\mathbb{E}[\mathbb{I}\{i \in \mathcal{S}\}] = \pi_i$

$$\Rightarrow \mathbb{E}[\hat{\mu}] = \mathbb{E}\left[\frac{1}{n} \sum_{i \in \mathcal{S}} y_i\right] = \mathbb{E}\left[\frac{1}{n} \sum_{i=1}^{N_u} y_i \mathbb{I}\{i \in \mathcal{S}\}\right]$$

$$= \frac{1}{n} \sum_{i=1}^{N_u} y_i \mathbb{E}[\mathbb{I}\{i \in \mathcal{S}\}] = \frac{1}{n} \sum_{i=1}^{N_u} y_i \pi_i$$

- SRS without replacement $\rightarrow$ unbiased because $\pi_i = \frac{n}{N_u}$
- Unequal probability sampling
  - More common than SRS, especially with networks. (More soon)
  - Sample mean can be a poor (i.e., biased) estimator for μ

Horvitz-Thompson estimation for totals

- Idea: weighted average using inclusion probabilities as weights

Horvitz-Thompson (HT) estimator

$$\hat{\mu}_\pi = \frac{1}{N_u} \sum_{i \in \mathcal{S}} \frac{y_i}{\pi_i} \qquad \text{and} \qquad \hat{\tau}_\pi = N_u \hat{\mu}_\pi$$

Horvitz-Thompson estimation for totals

- Idea: weighted average using inclusion probabilities as weights

Horvitz-Thompson (HT) estimator

$$\hat{\mu}_\pi = \frac{1}{N_u} \sum_{i \in \mathcal{S}} \frac{y_i}{\pi_i} \qquad \text{and} \qquad \hat{\tau}_\pi = N_u \hat{\mu}_\pi$$

- Remedies the bias problem

$$\mathbb{E}\left[\hat{\mu}_\pi\right] = \frac{1}{N_u} \sum_{i=1}^{N_u} \frac{y_i}{\pi_i} \mathbb{E}\left[\mathbb{I}\left\{i \in \mathcal{S}\right\}\right] = \frac{1}{N_u} \sum_{i=1}^{N_u} y_i = \mu$$

$\Rightarrow$ Size of the population N_u assumed known

$\Rightarrow$ Broad applicability, but π_i may be difficult to compute

Horvitz-Thompson estimator variance

- **Def:** Joint inclusion probability π_{ij} of units i and j is

$$\pi_{ij} := \mathrm{P}\left[\text{units } i \text{ and } j \text{ belong in the sample } \mathcal{S}\right]$$

- If inclusion of units i and j are independent events $\Rightarrow \pi_{ij} = \pi_i \pi_j$
- Ex: Simple random sampling without replacement yields

$$\pi_{ij} = \frac{n(n-1)}{N_u(N_u-1)}$$

Horvitz-Thompson estimator variance

- **Def:** Joint inclusion probability π_{ij} of units i and j is

$$\pi_{ij} := \mathrm{P}\left[\text{units } i \text{ and } j \text{ belong in the sample } \mathcal{S}\right]$$

- If inclusion of units i and j are independent events $\Rightarrow \pi_{ij} = \pi_i \pi_j$
- Ex: Simple random sampling without replacement yields

$$\pi_{ij} = \frac{n(n-1)}{N_u(N_u-1)}$$

- Variance of the HT estimator:

$$\mathrm{var}\left[\hat{\tau}_\pi\right] = \sum_{i \in \mathcal{U}} \sum_{j \in \mathcal{U}} y_i y_j \left(\frac{\pi_{ij}}{\pi_i \pi_j} - 1 \right), \quad \mathrm{var}\left[\hat{\mu}_\pi\right] = \frac{\mathrm{var}\left[\hat{\tau}_\pi\right]}{N_u^2}$$

$\Rightarrow$ Typically estimated in an unbiased fashion from the sample $\mathcal{S}$

Probability proportional to size sampling

- Unequal probability sampling
  - $\Rightarrow$ n units selected w.r.t. a distribution $\{p_1, \ldots, p_{N_u}\}$ on $\mathcal{U}$
  - $\Rightarrow$ Uniform sampling: special case with $p_i = \frac{1}{N_u}$ for all $i \in \mathcal{U}$

Probability proportional to size sampling

- Unequal probability sampling
  - $\Rightarrow$ n units selected w.r.t. a distribution $\{p_1, \ldots, p_{N_u}\}$ on $\mathcal{U}$
  - $\Rightarrow$ Uniform sampling: special case with $p_i = \frac{1}{N_u}$ for all $i \in \mathcal{U}$
- Probability proportional to size (PPS) sampling
  - $\Rightarrow$ Probabilities p_i proportional to a characteristic c_i

  Ex: households chosen by drawing names from a database

Probability proportional to size sampling

- Unequal probability sampling
  - $\Rightarrow$ n units selected w.r.t. a distribution $\{p_1, \ldots, p_{N_u}\}$ on $\mathcal{U}$
  - $\Rightarrow$ Uniform sampling: special case with $p_i = \frac{1}{N_u}$ for all $i \in \mathcal{U}$
- Probability proportional to size (PPS) sampling
  - $\Rightarrow$ Probabilities p_i proportional to a characteristic c_i

  Ex: households chosen by drawing names from a database
- If sampling with replacement, PPS inclusion probabilities are

$$\pi_i = 1 - (1 - p_i)^n, \quad \text{where } p_i = \frac{c_i}{\sum_k c_k}$$

- Joint inclusion probabilities for variance calculations

$$\pi_{ij} = \pi_i + \pi_j - [1 - (1 - p_i - p_j)^n]$$

Estimation of group size

- So far implicitly assumed N_u known $\rightarrow$ Often not the case!

  Ex: endangered animal species, people at risk of rare disease
- Special population total often of interest is the group size

$$N_u = \sum_{i \in \mathcal{U}} 1$$

Estimation of group size

- So far implicitly assumed N_u known $\rightarrow$ Often not the case!

  Ex: endangered animal species, people at risk of rare disease
- Special population total often of interest is the group size

$$N_u = \sum_{i \in \mathcal{U}} 1$$

- Suggests the following HT estimator of N_u

$$\hat{N}_u = \sum_{i \in \mathcal{S}} \pi_i^{-1}$$

$\Rightarrow$ Infeasible, since knowledge of N_u needed to compute π_i

Capture-recapture estimator

- Capture-recapture estimators overcome HT limitations in this setting
- Two rounds of SRS without replacement $\Rightarrow$ Two samples $\mathcal{S}_1$, $\mathcal{S}_2$

  Round 1: Mark all units in sample $\mathcal{S}_1$ of size n_1 from $\mathcal{U}$
  - Ex: tagging a fish, noting the ID number...
  - All units in $\mathcal{S}_1$ are returned to the population

  Round 2: Obtain a sample $\mathcal{S}_2$ of size n_2 from $\mathcal{U}$

Capture-recapture estimator

- Capture-recapture estimators overcome HT limitations in this setting
- Two rounds of SRS without replacement $\Rightarrow$ Two samples $\mathcal{S}_1$, $\mathcal{S}_2$

  Round 1: Mark all units in sample $\mathcal{S}_1$ of size n_1 from $\mathcal{U}$
  - Ex: tagging a fish, noting the ID number...
  - All units in $\mathcal{S}_1$ are returned to the population

  Round 2: Obtain a sample $\mathcal{S}_2$ of size n_2 from $\mathcal{U}$

Capture-recapture estimator of N_u

$$\hat{N}_u := \frac{n_2}{m} n_1, \quad \text{where } m := |\mathcal{S}_1 \cap \mathcal{S}_2|$$

- Factor m/n_2 indicative of marked fraction of the overall population

  $\Rightarrow$ Can derive using model-based arguments as an ML estimator

Common graph sampling designs

Network sampling and challenges

Background on statistical sampling theory

Graph sampling designs

Estimation of network totals, group size, and degree distributions

Network topology inference problems

Link prediction

Inference of association networks

Tomographic network topology inference

Graph sampling designs

- Q: What are common designs for sampling a graph G?
- A: Will see a few examples, along with their inclusion probabilities π_i

Graph sampling designs

- Q: What are common designs for sampling a graph G?
- A: Will see a few examples, along with their inclusion probabilities π_i
- Graph-based sampling designs

  $\Rightarrow$ Two inter-related classes of units, vertices i and edges (i, j)
- Often two stages
  - Selection among one class of units (e.g., vertices)
  - Observation of units from the other class (e.g., edges)

Graph sampling designs

- Q: What are common designs for sampling a graph G?
- A: Will see a few examples, along with their inclusion probabilities π_i
- Graph-based sampling designs

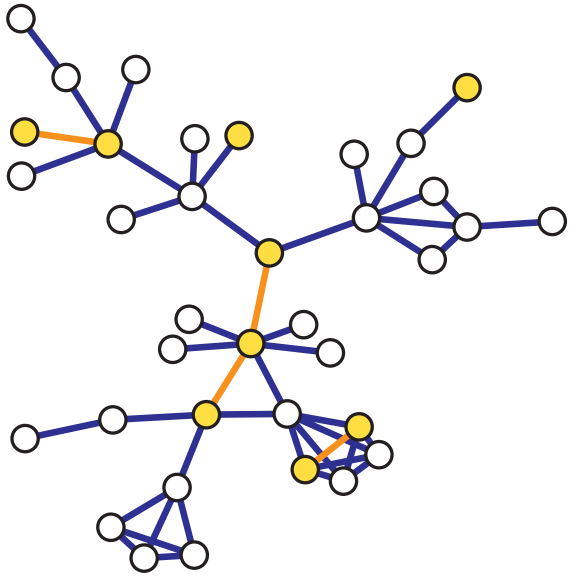
  $\Rightarrow$ Two inter-related classes of units, vertices i and edges (i, j)
- Often two stages
  - Selection among one class of units (e.g., vertices)
  - Observation of units from the other class (e.g., edges)
- Inclusion probabilities offer insight into the nature of the designs

  $\Rightarrow$ Central to HT estimators of network characteristics $\eta(G)$

Induced subgraph sampling

S) Sample n vertices $V^* = \{i_1, \ldots, i_n\}$ without replacement (SRS)

O) Observe edges $(i, j) \in E^*$ only when both $i, j \in V^*$ (induced by V^*)

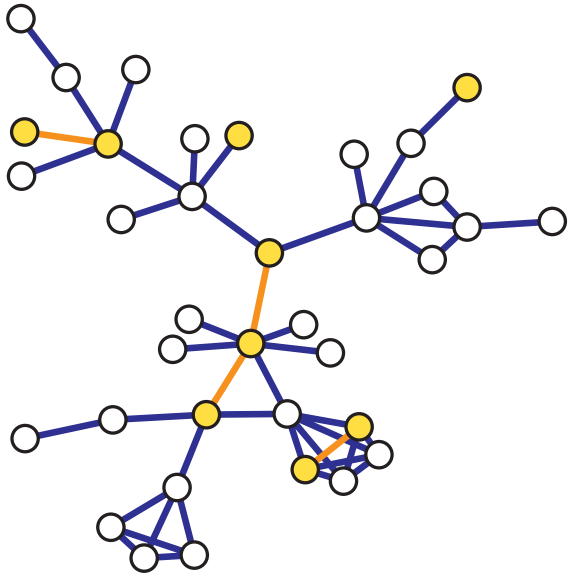


- Ex: construction of contact networks in social network research

Induced subgraph sampling

S) Sample n vertices $V^* = \{i_1, \ldots, i_n\}$ without replacement (SRS)

O) Observe edges $(i, j) \in E^*$ only when both $i, j \in V^*$ (induced by V^*)



- Ex: construction of contact networks in social network research
- Vertex and edge inclusion probabilities are uniformly equal to

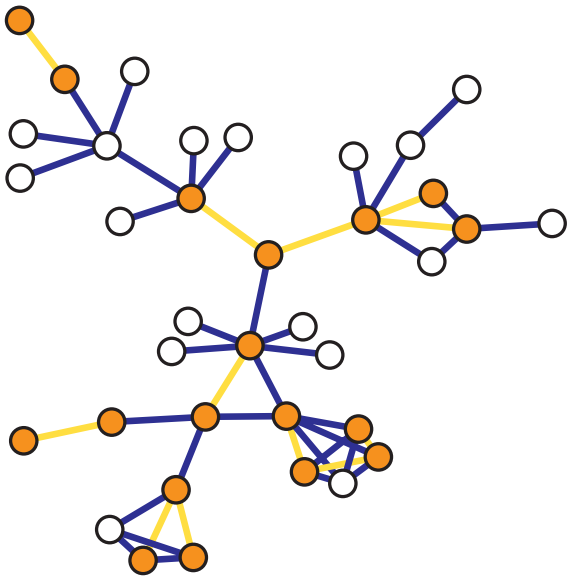
$$\pi_i = \frac{n}{N_v} \quad \text{and} \quad \pi_{\{i,j\}} = \frac{n(n-1)}{N_v(N_v-1)}$$

Incident subgraph sampling

- Consider a complementary design to induced subgraph sampling

S) Sample n edges E^* without replacement (SRS)

O) Observe vertices $i \in V^*$ incident to those selected edges in E^*



- Ex: construction of sampled telephone call graphs

Inclusion probabilities

- For incident subgraph sampling, edge inclusion probabilities are

$$\pi_{\{i,j\}} = \frac{n}{N_e}$$

Inclusion probabilities

- For incident subgraph sampling, edge inclusion probabilities are

$$\pi_{\{i,j\}} = \frac{n}{N_e}$$

- Vertex in V^* if any one or more of its incident edges are sampled

$$\begin{aligned}
\pi_i &= \mathrm{P}\left[\text{vertex } i \text{ is sampled}\right] \\
&= 1 - \mathrm{P}\left[\text{no edge incident to } i \text{ is sampled}\right] \\
&= \begin{cases} 1 - \dfrac{\binom{N_e - d_i}{n}}{\binom{N_e}{n}}, & \text{if } n \leq N_e - d_i \\ 1, & \text{if } n > N_e - d_i \end{cases}
\end{aligned}$$

Inclusion probabilities

- For incident subgraph sampling, edge inclusion probabilities are

$$\pi_{\{i,j\}} = \frac{n}{N_e}$$

- Vertex in V^* if any one or more of its incident edges are sampled

$$\begin{aligned}
\pi_i &= \mathrm{P}\left[\text{vertex } i \text{ is sampled}\right] \\
&= 1 - \mathrm{P}\left[\text{no edge incident to } i \text{ is sampled}\right] \\
&= \begin{cases} 1 - \dfrac{\binom{N_e - d_i}{n}}{\binom{N_e}{n}}, & \text{if } n \leq N_e - d_i \\ 1, & \text{if } n > N_e - d_i \end{cases}
\end{aligned}$$

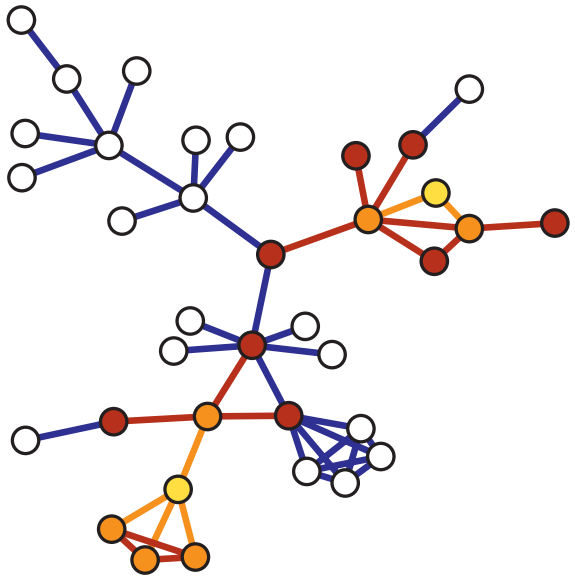
- Vertices included with unequal probs. that depend on their degrees
  - $\Rightarrow$ Probability proportional to size (degree) sampling of vertices
  - $\Rightarrow$ Requires knowledge of N_e and degree sequence $\{d_i\}_{i \in V^*}$

Snowball sampling

S) Sample n vertices $V_0^* = \{i_1, \ldots, i_n\}$ without replacement (SRS)

O1) Observe edges E_0^* incident to each $i \in V_0^*$, forming the initial wave

O2) Observe neighbors $\mathcal{N}(V_0^*)$ of $i \in V_0^*$, i.e., $V_1^* = \mathcal{N}(V_0^*) \cap (V_0^*)^c$



- Iterate to a desired number of e.g., k waves, or until V_k^* empty

  $\Rightarrow$ G^* has $V^* = V_0^* \cup V_1^* \cup \ldots \cup V_k^*$, and their incident edges

- Ex: 'spiders' or 'crawlers' to discover the WWW's structure

Star sampling

- Difficult to compute inclusion probabilities beyond a single wave
  - $\Rightarrow$ Single-wave snowball sampling reduces to star sampling

Star sampling

- Difficult to compute inclusion probabilities beyond a single wave
  - $\Rightarrow$ Single-wave snowball sampling reduces to star sampling
- Unlabeled: $V^* = V_0^*$ and $E^* = E_0^*$ their incident edges
  - Ex: Count all co-authors of n sampled authors
  - Vertex inclusion probabilities are simply $\pi_i = n/N_v$

Star sampling

- Difficult to compute inclusion probabilities beyond a single wave

  $\Rightarrow$ Single-wave snowball sampling reduces to star sampling

- Unlabeled: $V^* = V_0^*$ and $E^* = E_0^*$ their incident edges
  - Ex: Count all co-authors of n sampled authors
  - Vertex inclusion probabilities are simply $\pi_i = n/N_v$
- Labeled: $V^* = V_0^* \cup (\mathcal{N}(V_0^*) \cap (V_0^*)^c)$ and $E^* = E_0^*$
  - Ex: Count and identify all co-authors of n sampled authors
  - Vertex inclusion probabilities can be shown to look like

$$\pi_i = \sum_{L \subseteq \mathcal{N}_i} (-1)^{|L|+1} \mathrm{P}[L], \quad \text{where } \mathrm{P}[L] = \frac{\binom{N_v - |L|}{n - |L|}}{\binom{N_v}{n}}$$

  - Denoted by $\mathcal{N}_i$ the neighborhood of vertex i (including i itself)

Link tracing

- Link-tracing designs
  - $\Rightarrow$ Select an initial sample of vertices V_S^*
  - $\Rightarrow$ Trace edges (links) from V_s^* to another set of vertices V_T^*
- Snowball sampling: special case where all incident edges are traced
- May be infeasible to follow all incident edges to a given vertex

  Ex: lack of recollection/deception in social contact networks

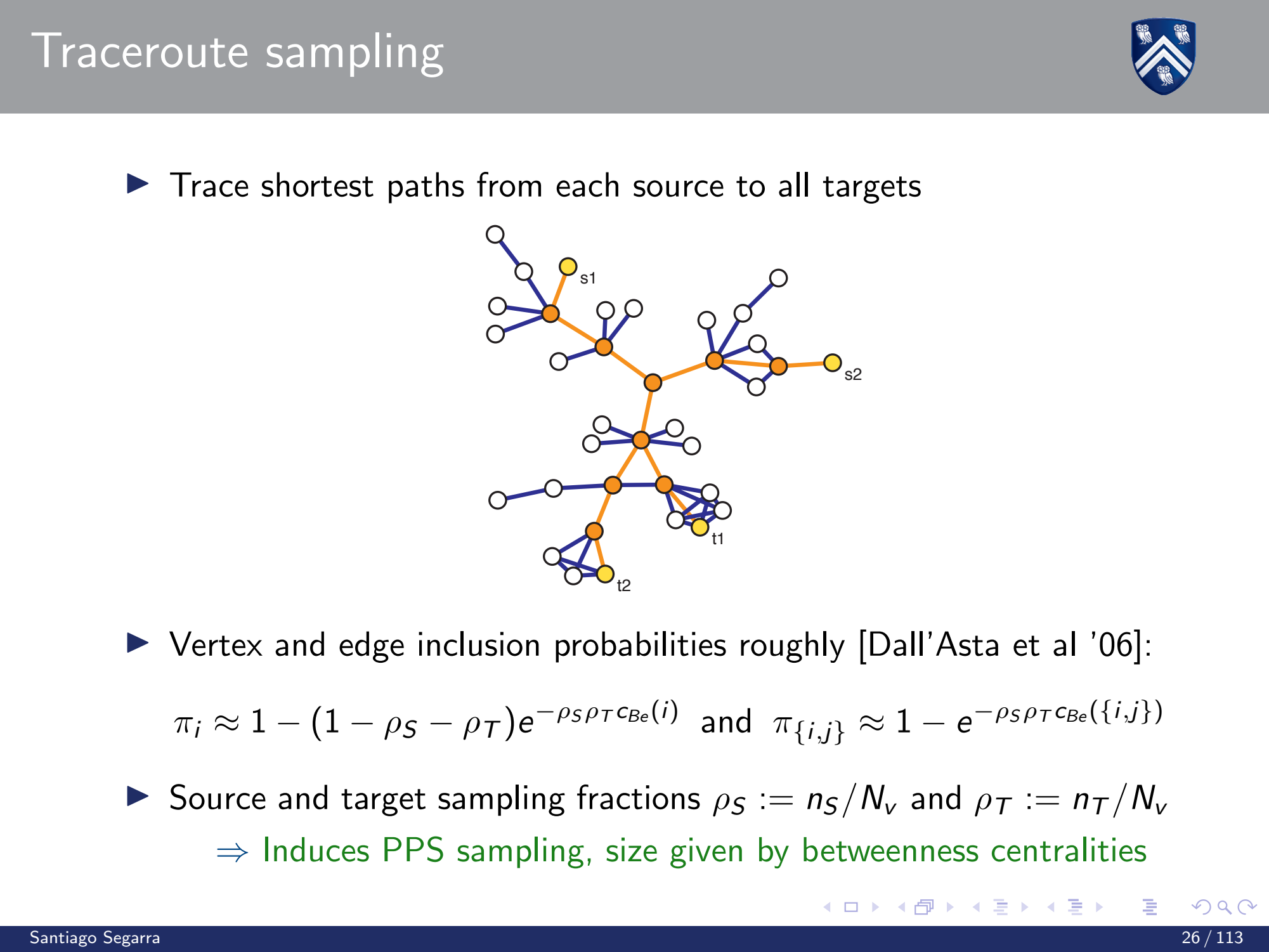
Link tracing

- Link-tracing designs
  - $\Rightarrow$ Select an initial sample of vertices V_S^*
  - $\Rightarrow$ Trace edges (links) from V_s^* to another set of vertices V_T^*
- Snowball sampling: special case where all incident edges are traced
- May be infeasible to follow all incident edges to a given vertex

  Ex: lack of recollection/deception in social contact networks
- Path sampling designs
  - $\Rightarrow$ Source nodes $V_S^* = \{s_1, \ldots, s_{n_S}\} \subset V$
  - $\Rightarrow$ Target nodes $V_T^* = \{t_1, \ldots, t_{n_T}\} \subset V \setminus V_S^*$
  - $\Rightarrow$ Traverse and measure the path between each pair (s_i, t_j)

  Ex: `Traceroute` Internet studies, Milgram's "Six Degrees" experiment

Traceroute sampling



- Trace shortest paths from each source to all targets

- Vertex and edge inclusion probabilities roughly [Dall'Asta et al '06]:

$$\pi_i \approx 1 - (1 - \rho_S - \rho_T)e^{-\rho_S \rho_T c_{Be}(i)} \quad \text{and} \quad \pi_{\{i,j\}} \approx 1 - e^{-\rho_S \rho_T c_{Be}(\{i,j\})}$$

- Source and target sampling fractions $\rho_S := n_S/N_v$ and $\rho_T := n_T/N_v$

  $\Rightarrow$ Induces PPS sampling, size given by betweenness centralities

Estimation of totals in networks

Network sampling and challenges

Background on statistical sampling theory

Graph sampling designs

Estimation of network totals, group size, and degree distributions

Network topology inference problems

Link prediction

Inference of association networks

Tomographic network topology inference

Network summaries as totals

- Various graph summaries $\eta(G)$ are expressible in terms of totals τ

Network summaries as totals

- Various graph summaries $\eta(G)$ are expressible in terms of totals τ

  Average degree: Let $\mathcal{U} = V$ and $y_i = d_i$, then $\eta(G) = \bar{d} \propto \sum_{i \in V} d_i$

Network summaries as totals

- Various graph summaries $\eta(G)$ are expressible in terms of totals τ

  Average degree: Let $\mathcal{U} = V$ and $y_i = d_i$, then $\eta(G) = \bar{d} \propto \sum_{i \in V} d_i$

  Graph size: Let $\mathcal{U} = E$ and $y_{ij} = 1$, then $\eta(G) = N_e = \sum_{(i,j) \in E} 1$

Network summaries as totals

- Various graph summaries $\eta(G)$ are expressible in terms of totals τ

  Average degree: Let $\mathcal{U} = V$ and $y_i = d_i$, then $\eta(G) = \bar{d} \propto \sum_{i \in V} d_i$

  Graph size: Let $\mathcal{U} = E$ and $y_{ij} = 1$, then $\eta(G) = N_e = \sum_{(i,j) \in E} 1$

  Betweenness centrality: Let $\mathcal{U} = V^{(2)}$ (unordered vertex pairs) and $y_{ij} = \mathbb{I}\left\{k \in \mathcal{P}_{(i,j)}\right\}$. For unique shortest $i-j$ paths $\mathcal{P}_{(i,j)}$, then

$$\eta(G) = c_{Be}(k) = \sum_{(i,j) \in V^{(2)}} \mathbb{I}\left\{k \in \mathcal{P}_{(i,j)}\right\}$$

Network summaries as totals

- Various graph summaries $\eta(G)$ are expressible in terms of totals τ

  Average degree: Let $\mathcal{U} = V$ and $y_i = d_i$, then $\eta(G) = \bar{d} \propto \sum_{i \in V} d_i$

  Graph size: Let $\mathcal{U} = E$ and $y_{ij} = 1$, then $\eta(G) = N_e = \sum_{(i,j) \in E} 1$

  Betweenness centrality: Let $\mathcal{U} = V^{(2)}$ (unordered vertex pairs) and $y_{ij} = \mathbb{I}\left\{k \in \mathcal{P}_{(i,j)}\right\}$. For unique shortest $i-j$ paths $\mathcal{P}_{(i,j)}$, then

$$\eta(G) = c_{Be}(k) = \sum_{(i,j) \in V^{(2)}} \mathbb{I}\left\{k \in \mathcal{P}_{(i,j)}\right\}$$

  Clustering coefficient: Let $\mathcal{U} = V^{(3)}$ (unordered vertex triples), then

$$\eta(G) = \mathrm{cl}(G) = 3 \times \frac{\text{total number of triangles}}{\text{total number of connected triples}}$$

Network summaries as totals

- Various graph summaries $\eta(G)$ are expressible in terms of totals τ

  Average degree: Let $\mathcal{U} = V$ and $y_i = d_i$, then $\eta(G) = \bar{d} \propto \sum_{i \in V} d_i$

  Graph size: Let $\mathcal{U} = E$ and $y_{ij} = 1$, then $\eta(G) = N_e = \sum_{(i,j) \in E} 1$

  Betweenness centrality: Let $\mathcal{U} = V^{(2)}$ (unordered vertex pairs) and $y_{ij} = \mathbb{I}\left\{k \in \mathcal{P}_{(i,j)}\right\}$. For unique shortest $i-j$ paths $\mathcal{P}_{(i,j)}$, then

$$\eta(G) = c_{Be}(k) = \sum_{(i,j) \in V^{(2)}} \mathbb{I}\left\{k \in \mathcal{P}_{(i,j)}\right\}$$

  Clustering coefficient: Let $\mathcal{U} = V^{(3)}$ (unordered vertex triples), then

$$\eta(G) = \mathrm{cl}(G) = 3 \times \frac{\text{total number of triangles}}{\text{total number of connected triples}}$$

- Often such totals can be obtained from sampled G^* via HT estimation

Vertex totals

- Vertex totals are of the form $\tau = \sum_{i \in V} y_i$, averages are τ / N_v
  - Ex: average degree where $y_i = d_i$
  - Ex: nodes with characteristic $\mathcal{C}$, where $y_i = \mathbb{I}\{i \in \mathcal{C}\}$

Vertex totals

- Vertex totals are of the form $\tau = \sum_{i \in V} y_i$, averages are τ / N_v
  - Ex: average degree where $y_i = d_i$
  - Ex: nodes with characteristic $\mathcal{C}$, where $y_i = \mathbb{I}\{i \in \mathcal{C}\}$
- Given a sample $V^* \subseteq V$, the HT estimator for vertex totals is

$$\hat{\tau}_\pi = \sum_{i \in V^*} \frac{y_i}{\pi_i}$$

  $\Rightarrow$ Variance expressions carry over, let $\mathcal{U} = V$ and V^* for estimates

- Inclusion probabilities π_i depend on network sampling design

Totals on vertex pairs

- Quantity y_{ij} corresponding to vertex pairs $(i,j) \in V^{(2)}$ of interest
  - $\Rightarrow$ Totals $\tau = \sum_{(i,j) \in V^{(2)}} y_{ij}$ become relevant
    - Ex: graph size N_e and betweenness $c_{Be}(k)$ where $y_{ij} = \mathbb{I}\left\{k \in \mathcal{P}_{(i,j)}\right\}$
    - Ex: shared gender in friendship network, average dissimilarity

Totals on vertex pairs

- Quantity y_{ij} corresponding to vertex pairs $(i,j) \in V^{(2)}$ of interest
  - $\Rightarrow$ Totals $\tau = \sum_{(i,j) \in V^{(2)}} y_{ij}$ become relevant
    - Ex: graph size N_e and betweenness $c_{Be}(k)$ where $y_{ij} = \mathbb{I}\left\{k \in \mathcal{P}_{(i,j)}\right\}$
    - Ex: shared gender in friendship network, average dissimilarity
- The HT estimator in this context is

$$\hat{\tau}_\pi = \sum_{(i,j) \in V^{(2)*}} \frac{y_{ij}}{\pi_{ij}}$$

  - $\Rightarrow$ Edge totals a special case, when $y_{ij} \neq 0$ only for $(i,j) \in E$

Totals on vertex pairs

- Quantity y_{ij} corresponding to vertex pairs $(i,j) \in V^{(2)}$ of interest
  - $\Rightarrow$ Totals $\tau = \sum_{(i,j) \in V^{(2)}} y_{ij}$ become relevant
    - Ex: graph size N_e and betweenness $c_{Be}(k)$ where $y_{ij} = \mathbb{I}\left\{k \in \mathcal{P}_{(i,j)}\right\}$
    - Ex: shared gender in friendship network, average dissimilarity
- The HT estimator in this context is

$$\hat{\tau}_\pi = \sum_{(i,j) \in V^{(2)*}} \frac{y_{ij}}{\pi_{ij}}$$

  - $\Rightarrow$ Edge totals a special case, when $y_{ij} \neq 0$ only for $(i,j) \in E$
- Variance expression increasingly complicated, namely

$$\text{var}\left[\hat{\tau}_\pi\right] = \sum_{(i,j) \in V^{(2)}} \sum_{(k,l) \in V^{(2)}} y_{ik} y_{kl} \left(\frac{\pi_{ijkl}}{\pi_{ij}\pi_{kl}} - 1 \right)$$

  - $\Rightarrow$ Depends on inclusion probabilities π_{ijkl} of vertex quadruples

Example: Estimating network size

- Consider estimating N_e as an edge total, i.e.,

$$N_e = \sum_{(i,j)\in E} 1 = \sum_{(i,j)\in V^{(2)}} A_{ij}$$

Example: Estimating network size

- Consider estimating N_e as an edge total, i.e.,

$$N_e = \sum_{(i,j)\in E} 1 = \sum_{(i,j)\in V^{(2)}} A_{ij}$$

- Bernoulli sampling (BS): $\mathbb{I}\{i \in V^*\} \sim \text{Ber}(p)$ i.i.d. for all $i \in V$

  $\Rightarrow$ Edges E^* obtained via induced subgraph sampling $\Rightarrow \pi_{ij} = p^2$

- The HT estimator of N_e is

$$\hat{N}_e = \sum_{(i,j)\in V^{(2)*}} \frac{A_{ij}}{\pi_{ij}} = p^{-2} N_e^*$$

  $\Rightarrow$ Scales up the empirically observed edge total N_e^* by $p^{-2} > 1$

Example: Estimating network size

- Consider estimating N_e as an edge total, i.e.,

$$N_e = \sum_{(i,j)\in E} 1 = \sum_{(i,j)\in V^{(2)}} A_{ij}$$

- Bernoulli sampling (BS): $\mathbb{I}\{i \in V^*\} \sim \text{Ber}(p)$ i.i.d. for all $i \in V$

  $\Rightarrow$ Edges E^* obtained via induced subgraph sampling $\Rightarrow \pi_{ij} = p^2$

- The HT estimator of N_e is

$$\hat{N}_e = \sum_{(i,j)\in V^{(2)*}} \frac{A_{ij}}{\pi_{ij}} = p^{-2} N_e^*$$

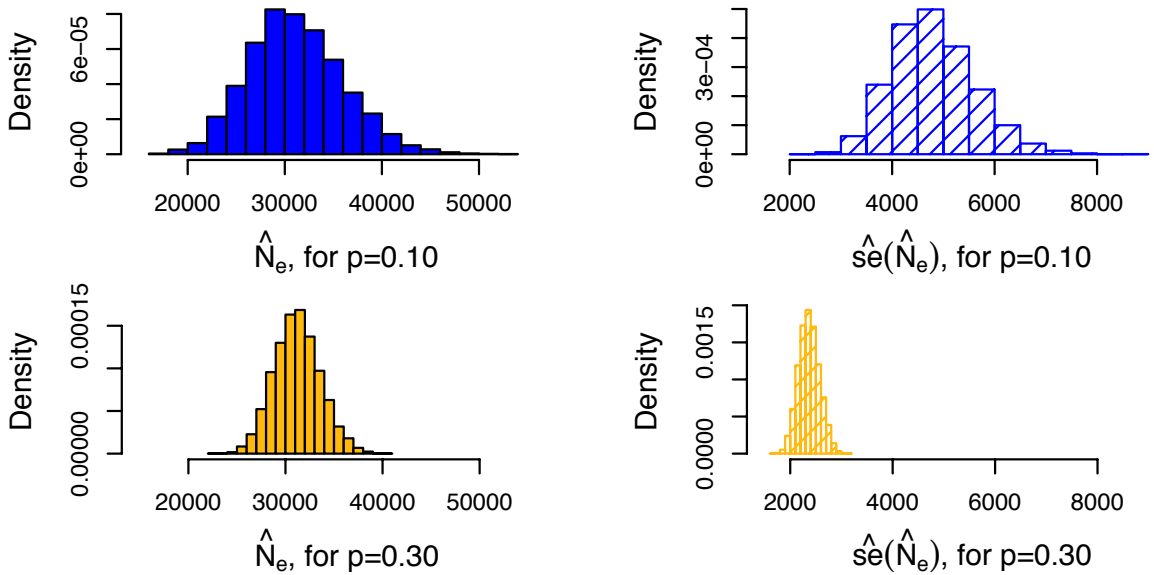
  $\Rightarrow$ Scales up the empirically observed edge total N_e^* by $p^{-2} > 1$

- Variance can be shown to take the form [Frank '77]

$$\text{var}\left[\hat{N}_e\right] = (p^{-1} - 1)\sum_{i\in V} d_i^2 + (p^{-2} - 2p^{-1} + 1)N_e$$

Example: Estimating network size (cont.)

- Protein network: $N_v = 5,151$, $N_e = 31,201$
  - $\Rightarrow$ BS of vertices with $p = 0.1$ and $p = 0.3$
  - $\Rightarrow$ Process repeated for 10,000 trials $\Rightarrow$ histogram of $\hat{N}_e$



- Average of $\hat{N}_e$ was $31,116$ and $31,203$ $\Rightarrow$ Unbiasedness supported
  - $\Rightarrow$ Mean and variability of $\hat{se}$ shrinks with p (larger sample)

Example: Estimating clustering coefficient

- Global clustering coefficient $\mathrm{cl}(G)$ can be expressed as

$$\mathrm{cl}(G) = \frac{3\tau_{\triangle}(G)}{3\tau_{\triangle}(G) + \tau_3(G)}$$

Example: Estimating clustering coefficient

- Global clustering coefficient $\mathrm{cl}(G)$ can be expressed as

$$\mathrm{cl}(G) = \frac{3\tau_{\triangle}(G)}{3\tau_{\triangle}(G) + \tau_3(G)}$$

- Involves the quotient of two totals on vertex triples

$$\tau = \sum_{(i,j,k) \in V^{(3)}} y_{ijk} \Rightarrow \hat{\tau}_{\pi} = \sum_{(i,j,k) \in V^{(3)*}} \frac{y_{ijk}}{\pi_{ijk}}$$

Example: Estimating clustering coefficient

- Global clustering coefficient $\mathrm{cl}(G)$ can be expressed as

$$\mathrm{cl}(G) = \frac{3\tau_{\triangle}(G)}{3\tau_{\triangle}(G) + \tau_3(G)}$$

- Involves the quotient of two totals on vertex triples

$$\tau = \sum_{(i,j,k) \in V^{(3)}} y_{ijk} \Rightarrow \hat{\tau}_{\pi} = \sum_{(i,j,k) \in V^{(3)*}} \frac{y_{ijk}}{\pi_{ijk}}$$

- Total number of triangles $\tau_{\triangle}(G)$, where

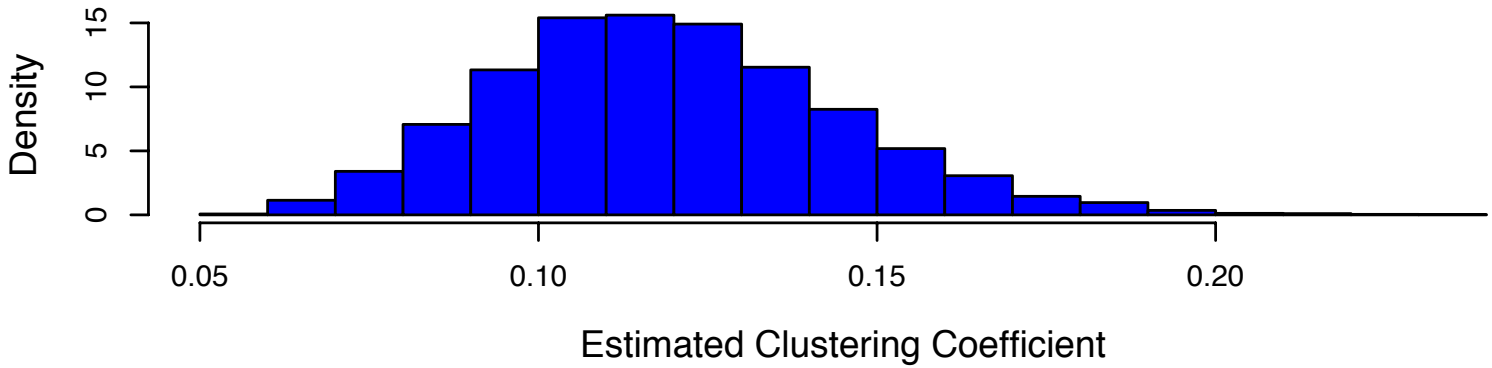
$$y_{ijk} = A_{ij} A_{jk} A_{ki}$$

- Total number of triples connected by exactly two edges $\tau_3(G)$, where

$$y_{ijk} = A_{ij} A_{jk} (1 - A_{ki}) + A_{ij} (1 - A_{jk}) A_{ki} + (1 - A_{ij}) A_{jk} A_{ki}$$

Example: Estimating clustering coefficient (cont.)

- Protein network: $\tau_{\triangle}(G) = 44,858$, $\tau_3(G) \approx 1M$, and $\mathrm{cl}(G) = 0.1179$
  - $\Rightarrow$ BS of vertices with $p = 0.2$
  - $\Rightarrow$ Induced subgraph sampling of edges
  - $\Rightarrow$ Process repeated for 10,000 trials $\Rightarrow$ histogram of $\hat{\mathrm{cl}}(G)$



- Unbiased HT estimators $\hat{\tau}_{\triangle} = p^{-3}\tau_{\triangle}(G^*)$ and $\hat{\tau}_3 = p^{-3}\tau_3(G^*)$
  - $\Rightarrow$ Plug-in estimator $\hat{\mathrm{cl}}(G) = 3\hat{\tau}_{\triangle}/(3\hat{\tau}_{\triangle} + \hat{\tau}_3)$
  - $\Rightarrow$ Quite accurate with mean 0.1191 and $\hat{\mathrm{se}}$ of 0.0251

Caveat emptor

- Horvitz-Thompson framework fairly straightforward in its essence

Caveat emptor

- Horvitz-Thompson framework fairly straightforward in its essence
- Success in network sampling and estimation rests on interaction among
  a) Sampling design;
  b) Measurements taken; and
  c) Total to be estimated

Caveat emptor

- Horvitz-Thompson framework fairly straightforward in its essence
- Success in network sampling and estimation rests on interaction among
  a) Sampling design;
  b) Measurements taken; and
  c) Total to be estimated
- Three basic elements must be present in the problem
  1) Network summary statistic $\eta(G)$ expressible as total;
  2) Values y either observed, or obtainable from measurements; and
  3) Inclusion probabilities π computable for the sampling design

Caveat emptor

- Horvitz-Thompson framework fairly straightforward in its essence
- Success in network sampling and estimation rests on interaction among
  a) Sampling design;
  b) Measurements taken; and
  c) Total to be estimated
- Three basic elements must be present in the problem
  1) Network summary statistic $\eta(G)$ expressible as total;
  2) Values y either observed, or obtainable from measurements; and
  3) Inclusion probabilities π computable for the sampling design
- Unfortunately, often not all three are present at the same time ...

Example: Estimating average degree

- Recall our first example on estimation of average degree $\frac{1}{N_v}\sum_{i\in V} d_i$
  - Design 1: Unlabeled star sampling, observes degrees d_i, $i \in V^*$
  - Design 2: Induced subgraph sampling, does not observe degrees
- Average degree is a scaling of a vertex total (N_v known)

  $\Rightarrow$ HT estimation applicable so long as $y_i = d_i$ observed

Example: Estimating average degree

- Recall our first example on estimation of average degree $\frac{1}{N_v}\sum_{i\in V} d_i$
  - Design 1: Unlabeled star sampling, observes degrees d_i, $i \in V^*$
  - Design 2: Induced subgraph sampling, does not observe degrees
- Average degree is a scaling of a vertex total (N_v known)

  $\Rightarrow$ HT estimation applicable so long as $y_i = d_i$ observed
- True for unlabeled star sampling, and since $\pi_i = n/N_v$ we have

$$\hat{\mu}_{St} = \frac{\hat{\tau}_{St}}{N_v}, \quad \text{where } \hat{\tau}_{St} = \sum_{i\in V^*_{St}} \frac{d_i}{n/N_v}$$

Example: Estimating average degree

- Recall our first example on estimation of average degree $\frac{1}{N_v}\sum_{i\in V} d_i$
  - Design 1: Unlabeled star sampling, observes degrees d_i, $i \in V^*$
  - Design 2: Induced subgraph sampling, does not observe degrees
- Average degree is a scaling of a vertex total (N_v known)

  $\Rightarrow$ HT estimation applicable so long as $y_i = d_i$ observed
- True for unlabeled star sampling, and since $\pi_i = n/N_v$ we have

$$\hat{\mu}_{St} = \frac{\hat{\tau}_{St}}{N_v}, \quad \text{where } \hat{\tau}_{St} = \sum_{i\in V^*_{St}} \frac{d_i}{n/N_v}$$

- We do not observe d_i under induced subgraph sampling

  $\Rightarrow$ Not amenable to HT estimation as vertex total for this design

Example: Estimating average degree (cont.)

- Identity $\mu = \frac{2N_e}{N_v} \Rightarrow$ Tackle instead as estimation of network size N_e

Example: Estimating average degree (cont.)

- Identity $\mu = \frac{2N_e}{N_v} \Rightarrow$ Tackle instead as estimation of network size N_e
- For induced subgraph sampling $\pi_{ij} = \frac{n(n-1)}{N_v(N_v-1)}$, so HT estimator is

$$\hat{N}_{e,IS} = \sum_{(i,j)\in V^{(2)*}} \frac{A_{ij}}{n(n-1)/[N_v(N_v-1)]} = \frac{N_v(N_v-1)}{n(n-1)} N^*_{e,IS}$$

$\Rightarrow$ Desired unbiased estimator for the average degree is

$$\hat{\mu}_{IS} = \frac{2\hat{N}_{e,IS}}{N_v}$$

Example: Estimating average degree (cont.)

- Identity $\mu = \frac{2N_e}{N_v} \Rightarrow$ Tackle instead as estimation of network size N_e
- For induced subgraph sampling $\pi_{ij} = \frac{n(n-1)}{N_v(N_v-1)}$, so HT estimator is

$$\hat{N}_{e,IS} = \sum_{(i,j)\in V^{(2)*}} \frac{A_{ij}}{n(n-1)/[N_v(N_v-1)]} = \frac{N_v(N_v-1)}{n(n-1)} N^*_{e,IS}$$

$\Rightarrow$ Desired unbiased estimator for the average degree is

$$\hat{\mu}_{IS} = \frac{2\hat{N}_{e,IS}}{N_v}$$

- Estimators under both designs can be compared by writing them as

$$\hat{\mu}_{St} = \frac{2N^*_{e,St}}{n} \text{ and } \hat{\mu}_{IS} = \frac{2N^*_{e,IS}}{n} . \frac{N_v-1}{n-1}$$

$\Rightarrow$ Design 1: uses the identity $\mu = \frac{2N_e}{N_v}$ on G^*_{St}

$\Rightarrow$ Design 2: same but inflated by $\frac{N_v-1}{n-1}$, compensates $d^*_{i,IS} < d_i$

Estimation of network group size

- Assuming that N_v is known may not be on safe grounds
  - $\Rightarrow$ Human or animal groups too mobile or elusive to count accurately
  - $\Rightarrow$ All Web pages or Internet routers are too massive and dispersed
- Often estimating N_v may well be the prime objective

Estimation of network group size

- Assuming that N_v is known may not be on safe grounds
  - $\Rightarrow$ Human or animal groups too mobile or elusive to count accurately
  - $\Rightarrow$ All Web pages or Internet routers are too massive and dispersed
- Often estimating N_v may well be the prime objective
- If vertex SRS or BS feasible, could sample twice 'marking' in between
  - $\Rightarrow$ Facilitates usage of capture-recapture estimators 'off-the-shelf'

Estimation of network group size

- Assuming that N_v is known may not be on safe grounds
  - $\Rightarrow$ Human or animal groups too mobile or elusive to count accurately
  - $\Rightarrow$ All Web pages or Internet routers are too massive and dispersed
- Often estimating N_v may well be the prime objective
- If vertex SRS or BS feasible, could sample twice 'marking' in between
  - $\Rightarrow$ Facilitates usage of capture-recapture estimators 'off-the-shelf'
- If sampling infeasible, or capture-recapture performs poorly
  - $\Rightarrow$ Develop estimators of N_v tailored to the graph sampling at hand

Estimating the size of a "hidden population"

- Hidden population: individuals do not wish to expose themselves
  - Ex: humans of socially sensitive status, such as homeless
  - Ex: involved in socially sensitive activities, e.g., drugs, prostitution
- Such groups are often small $\Rightarrow$ Estimating their size is challenging

Estimating the size of a "hidden population"

- Hidden population: individuals do not wish to expose themselves
  - Ex: humans of socially sensitive status, such as homeless
  - Ex: involved in socially sensitive activities, e.g., drugs, prostitution
- Such groups are often small $\Rightarrow$ Estimating their size is challenging
- Snowball sampling used to estimate the size of hidden populations
- O. Frank and T. Snijders, "Estimating the size of hidden populations using snowball sampling," *J. Official Stats.*, vol. 10, pp. 53-67, 1994

Sampling a hidden population

- Directed graph $G(V, E)$, V the members of the hidden population
  - $\Rightarrow$ Graph describing willingness to identify other members
  - $\Rightarrow$ Arc (i, j) when ask individual i, mentions j as a member

Sampling a hidden population

- Directed graph $G(V, E)$, V the members of the hidden population
  - $\Rightarrow$ Graph describing willingness to identify other members
  - $\Rightarrow$ Arc (i, j) when ask individual i, mentions j as a member
- Graph G^* obtained via one-wave snowball sampling, i.e., $V^* = V_0^* \cup V_1^*$
  - $\Rightarrow$ Initial sample V_0^* obtained via BS from V with probability p_0

Sampling a hidden population

- Directed graph $G(V, E)$, V the members of the hidden population
  - $\Rightarrow$ Graph describing willingness to identify other members
  - $\Rightarrow$ Arc (i, j) when ask individual i, mentions j as a member
- Graph G^* obtained via one-wave snowball sampling, i.e., $V^* = V_0^* \cup V_1^*$
  - $\Rightarrow$ Initial sample V_0^* obtained via BS from V with probability p_0
- Consider the following random variables (RVs) of interest
  - $N = |V_0^*|$: size of the initial sample
  - M_1: number of arcs among individuals in V_0^*
  - M_2: number of arcs from individuals in V_0^* to individuals in V_1^*
- Snowball sampling yields measurements n, m_1, and m_2 of these RVs

Method of moments estimator

- Method of moments: equate moments to sample counterparts

$$\mathbb{E}[N] = \mathbb{E}\left[\sum_i \mathbb{I}\{i \in V_0^*\}\right] = N_v p_0 = n$$

$$\mathbb{E}[M_1] = \mathbb{E}\left[\sum_j \sum_{i \neq j} \mathbb{I}\{i \in V_0^*\}\mathbb{I}\{j \in V_0^*\}A_{ij}\right] = N_e p_0^2 = m_1$$

$$\mathbb{E}[M_2] = \mathbb{E}\left[\sum_j \sum_{i \neq j} \mathbb{I}\{i \in V_0^*\}\mathbb{I}\{j \notin V_0^*\}A_{ij}\right] = N_e p_0(1-p_0) = m_2$$

- Expectation w.r.t. randomness in selecting the sample V_0^*.

Method of moments estimator

- Method of moments: equate moments to sample counterparts

$$\mathbb{E}[N] = \mathbb{E}\left[\sum_i \mathbb{I}\{i \in V_0^*\}\right] = N_v p_0 = n$$

$$\mathbb{E}[M_1] = \mathbb{E}\left[\sum_j \sum_{i \neq j} \mathbb{I}\{i \in V_0^*\}\mathbb{I}\{j \in V_0^*\}A_{ij}\right] = N_e p_0^2 = m_1$$

$$\mathbb{E}[M_2] = \mathbb{E}\left[\sum_j \sum_{i \neq j} \mathbb{I}\{i \in V_0^*\}\mathbb{I}\{j \notin V_0^*\}A_{ij}\right] = N_e p_0(1-p_0) = m_2$$

- Expectation w.r.t. randomness in selecting the sample V_0^*. Solution:

$$\hat{N}_v = n\left(\frac{m_1 + m_2}{m_1}\right)$$

$\Rightarrow$ Size of initial sample inflated by estimate of the sampling rate

Estimation of other network characteristics

- Classical sampling theory rests heavily on Horvitz-Thompson framework
  - $\Rightarrow$ Scope limited to network totals
  - $\Rightarrow$ Q: Other network summaries, e.g., degree distributions?

Estimation of other network characteristics

- Classical sampling theory rests heavily on Horvitz-Thompson framework
  - $\Rightarrow$ Scope limited to network totals
  - $\Rightarrow$ Q: Other network summaries, e.g., degree distributions?
- Findings on the effect of sampling on observed degree distributions:
  - Highly unrepresentative of actual degree distributions; and
  - Unhelpful to characterizing heterogeneous distributions

Estimation of other network characteristics

- Classical sampling theory rests heavily on Horvitz-Thompson framework
  - $\Rightarrow$ Scope limited to network totals
  - $\Rightarrow$ Q: Other network summaries, e.g., degree distributions?
- Findings on the effect of sampling on observed degree distributions:
  - Highly unrepresentative of actual degree distributions; and
  - Unhelpful to characterizing heterogeneous distributions
- Ex: Internet `traceroute` sampling [Lakhina et al' 03]
  - $\Rightarrow$ Broad degree distribution in G^*, while concentrated in G

Estimation of other network characteristics

- Classical sampling theory rests heavily on Horvitz-Thompson framework
  - $\Rightarrow$ Scope limited to network totals
  - $\Rightarrow$ Q: Other network summaries, e.g., degree distributions?
- Findings on the effect of sampling on observed degree distributions:
  - Highly unrepresentative of actual degree distributions; and
  - Unhelpful to characterizing heterogeneous distributions
- Ex: Internet `traceroute` sampling [Lakhina et al' 03]
  - $\Rightarrow$ Broad degree distribution in G^*, while concentrated in G
- Ex: Sampling protein-protein interaction networks [Han et al' 05]
  - $\Rightarrow$ Power-law exponent estimate from G^* underestimates α in G

Impact of sampling on degree distribution

- Let $N(d)$ denote the number of vertices with degree d in G
  - $\Rightarrow$ Let $N^*(d)$ be the counterpart in a sampled graph G^*
  - $\Rightarrow$ Introduce vectors $\mathbf{n} = [N(0), \ldots, N(d_{\max})]^\top$ and likewise $\mathbf{n}^*$
- Under a variety of sampling designs, it holds that

$$\mathbb{E}[\mathbf{n}^*] = \mathbf{P}\mathbf{n}$$

  - $\Rightarrow$ Matrix $\mathbf{P}$ depends fully on the sampling, not G itself
  - $\Rightarrow$ Expectation w.r.t. randomness in selecting the sample G^*
- O. Frank, "Estimation of the number of vertices of different degrees in a graph," *J. Stat. Planning and Inference,* vol. 4, pp. 45-50, 1980

An inverse problem

- Recall the identity $\mathbb{E}[\mathbf{n}^*] = \mathbf{P}\mathbf{n}$ $\Rightarrow$ Face a linear inverse problem
- Unbiased estimator of the degree distribution $\mathbf{n}$

$$\hat{\mathbf{n}}_{\text{naive}} = \mathbf{P}^{-1}\mathbf{n}^*$$

- While natural, two problems with this simple solution
  - $\Rightarrow$ Matrix $\mathbf{P}$ typically not invertible in practice; and
  - $\Rightarrow$ Non-negativity of the solution is not guaranteed

An inverse problem

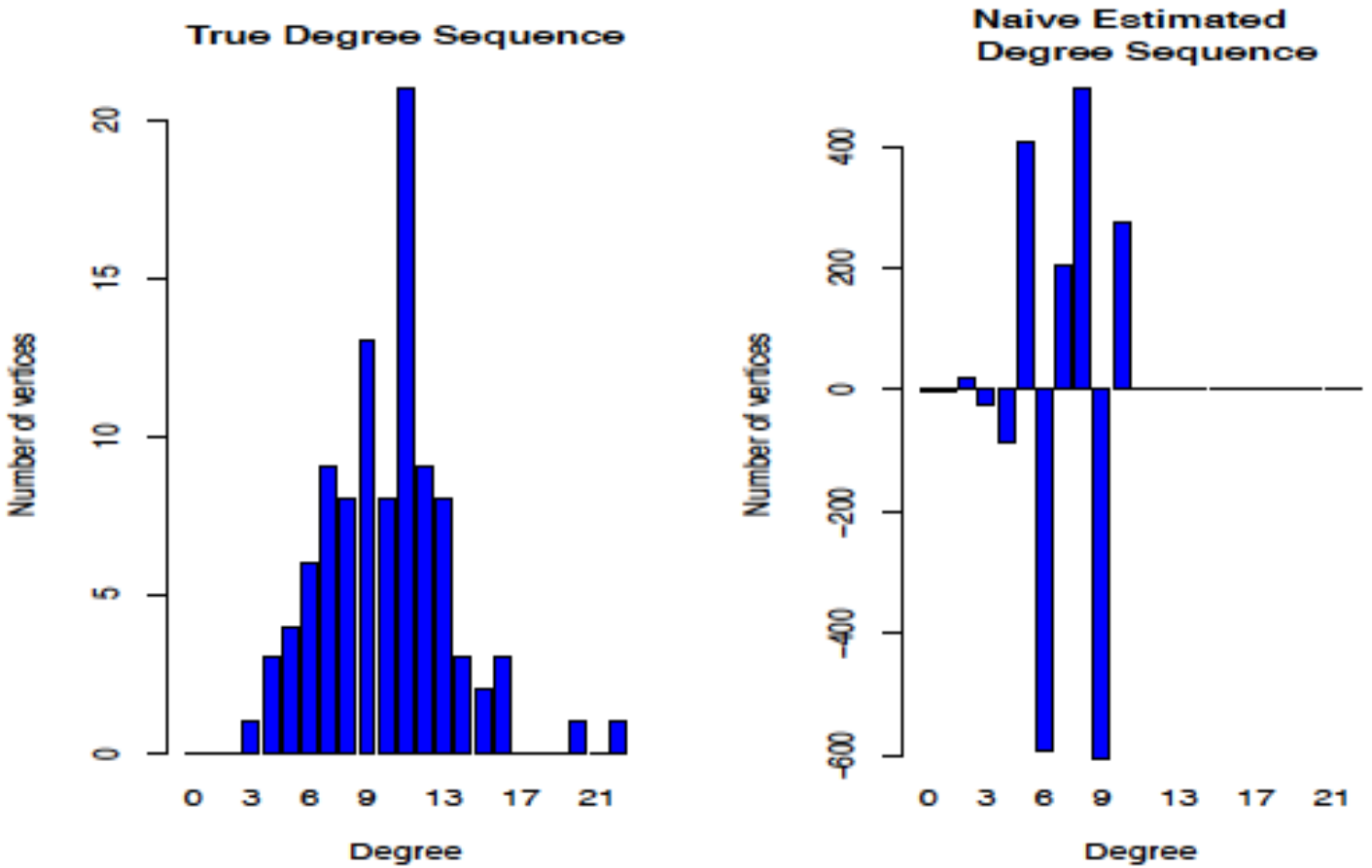
- Recall the identity $\mathbb{E}[\mathbf{n}^*] = \mathbf{P}\mathbf{n}$ $\Rightarrow$ Face a linear inverse problem
- Unbiased estimator of the degree distribution $\mathbf{n}$

$$\hat{\mathbf{n}}_{\text{naive}} = \mathbf{P}^{-1}\mathbf{n}^*$$

- While natural, two problems with this simple solution
  - $\Rightarrow$ Matrix $\mathbf{P}$ typically not invertible in practice; and
  - $\Rightarrow$ Non-negativity of the solution is not guaranteed
- We actually have an ill-posed linear inverse problem

Performance of naive estimator

- Erdős-Rényi graph with $N_v = 100$ and $N_e = 500$
  - $\Rightarrow$ BS of vertices with $p = 0.6$
  - $\Rightarrow$ Induced subgraph sampling of edges



Penalized least-squares formulation

- Constrained, penalized, weighted least-squares [Zhang et al '14]

$$\min_{\mathbf{n}} \ (\mathbf{P}\mathbf{n} - \mathbf{n}^*)^\top \mathbf{C}^{-1} (\mathbf{P}\mathbf{n} - \mathbf{n}^*) + \lambda \text{pen}(\mathbf{n})$$

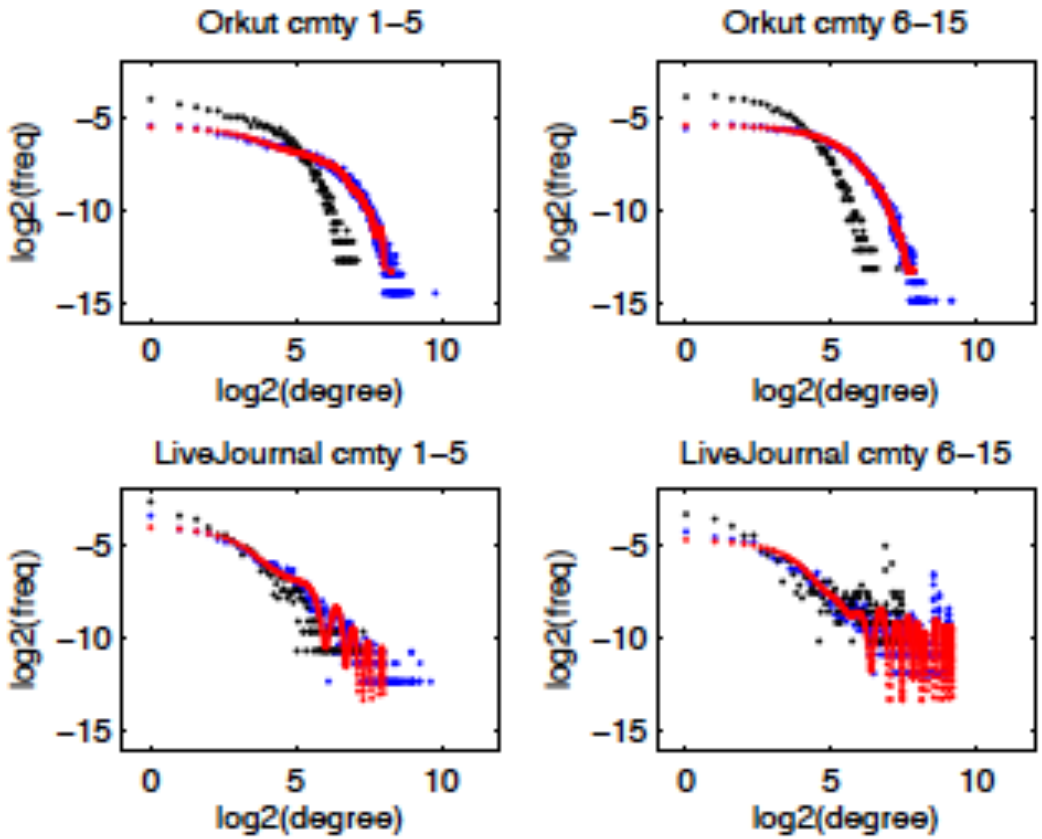
$$\text{s. to} \ \ N(d) \geq 0, \ d = 0, 1, \ldots, d_{\max},$$

$$\sum_{d=1}^{d_{\max}} N(d) = N_v$$

  - $\Rightarrow$ Matrix $\mathbf{C}$ denotes the covariance of $\mathbf{n}^*$
  - $\Rightarrow$ Functional pen($\mathbf{n}$) penalizes complexity in $\mathbf{n}$, tuned by λ
- Constraints
  - $\Rightarrow$ Non-negativity of degree counts
  - $\Rightarrow$ Total degree counts equal the number of vertices
  - $\Rightarrow$ Smoothness: pen($\mathbf{n}$) $= \|\mathbf{D}\mathbf{n}\|^2$, $\mathbf{D}$ differentiating operator

Application to online social networks

- Communities from online social networks Orkut and LiveJournal
  - $\Rightarrow$ BS of vertices with $p = 0.3$
  - $\Rightarrow$ Induced subgraph sampling of edges

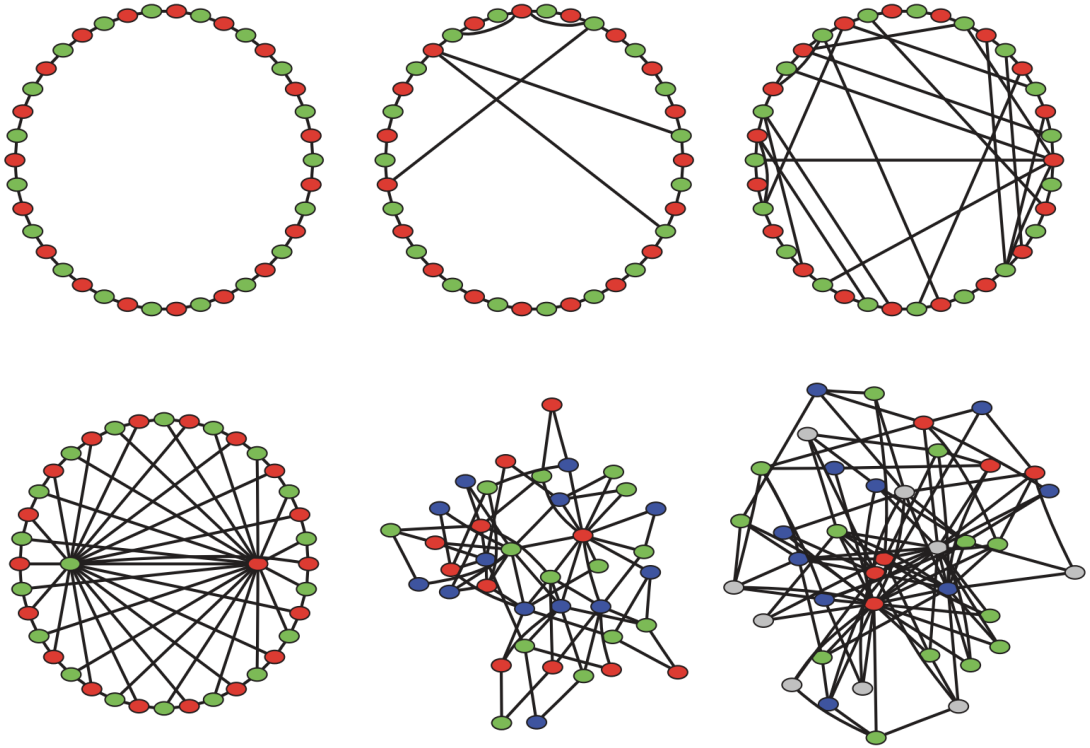


- True, sampled, and estimated degree distribution

Network of the week

Network of the week

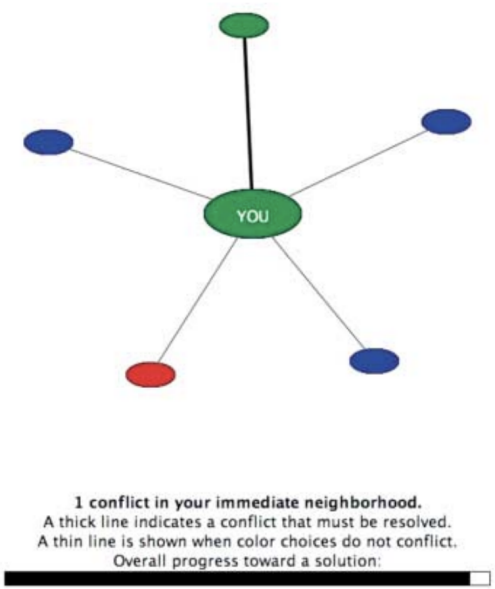
- "An Experimental Study of the Coloring Problem on Human Subject Networks" Kearns et al. (2006)

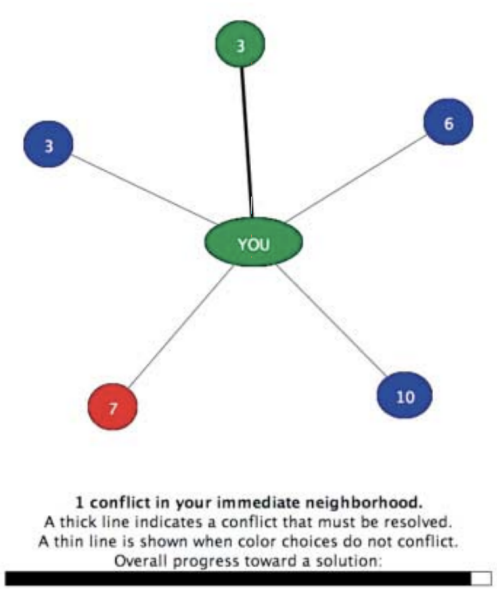


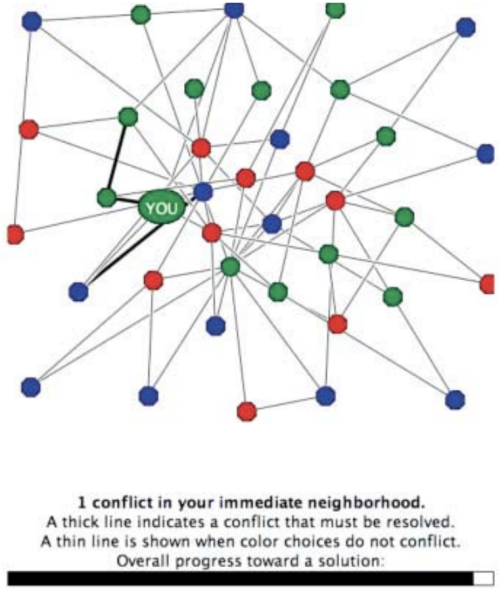
Network of the week

	Graph statistics								
	Colors required (No.)	Min. links (No.)	Max. links (No.)	Avg. links (No.)	SD	Avg. distance (No. of links)	Avg. experiment duration (s) and fraction solved		Distributed heuristic (No. of color changes)
Simple cycle	2	2	2	2	0	9.76	144.17	5/6	378
5-chord cycle	2	2	4	2.26	0.60	5.63	121.14	7/7	687
20-chord cycle	2	2	7	3.05	1.01	3.34	65.67	6/6	8265
Leader cycle	2	3	19	3.84	3.62	2.31	40.86	7/7	8797
Pref. att., $\nu = 2$	3	2	13	3.84	2.44	2.63	219.67	2/6	1744
Pref. att., $\nu = 3$	4	3	22	5.68	4.22	2.08	154.83	4/6	4703

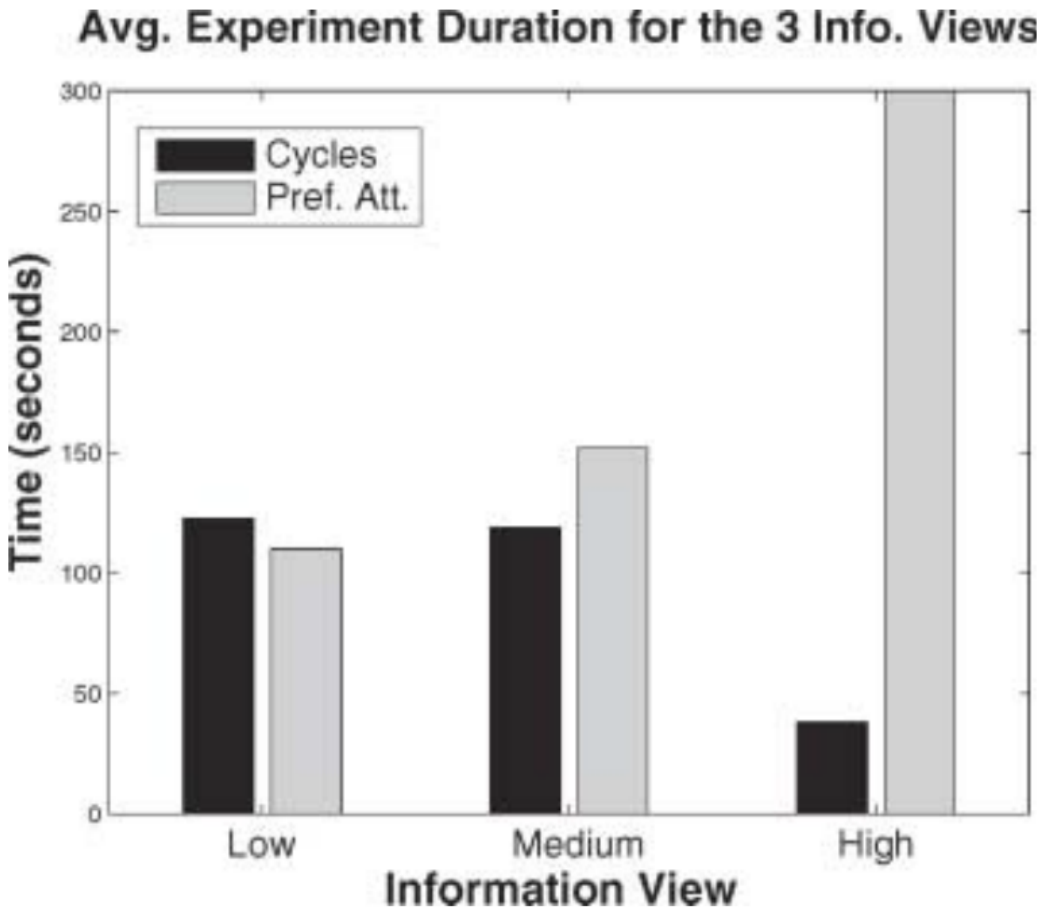
Network of the week







Network of the week



Network topology inference

Network sampling and challenges

Background on statistical sampling theory

Graph sampling designs

Estimation of network totals, group size, and degree distributions

Network topology inference problems

Link prediction

Inference of association networks

Tomographic network topology inference

Network topology inference

- Formulate problem as statistical inference task, i.e. given
  - Measurements x_i of attributes at some or all vertices $i \in V$
  - Indicators y_{ij} of edge status for some vertex pairs $\{i, j\} \in V^{(2)}$
  - A collection $\mathcal{G}$ of candidate graphs G

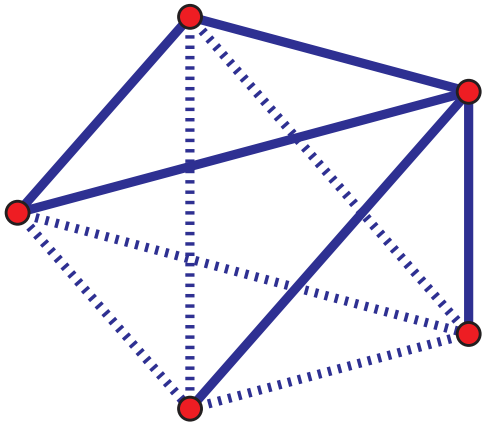
  Goal: infer the topology of the network $G(V, E)$

Network topology inference

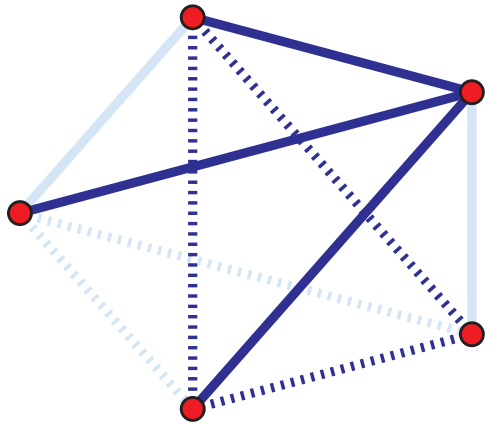
- Formulate problem as statistical inference task, i.e. given
  - Measurements x_i of attributes at some or all vertices $i \in V$
  - Indicators y_{ij} of edge status for some vertex pairs $\{i, j\} \in V^{(2)}$
  - A collection $\mathcal{G}$ of candidate graphs G

  Goal: infer the topology of the network $G(V, E)$
- Three canonical network topology inference problems
  - (i) Link prediction
  - (ii) Association network inference
  - (iii) Tomographic network topology inference

Link prediction



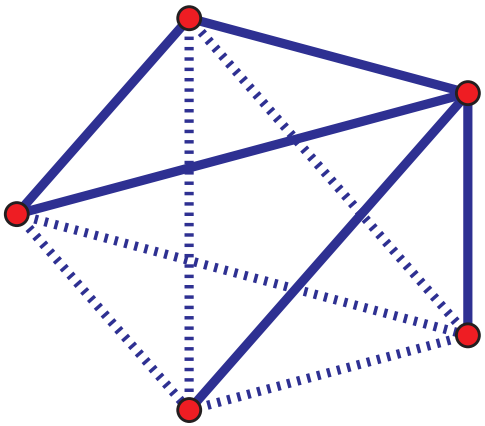
Original graph



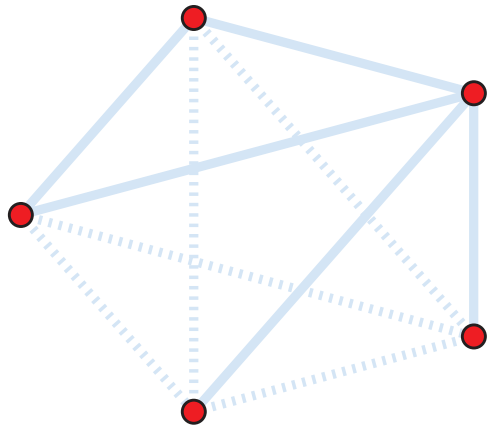
Link prediction

- Suppose we observe vertex attributes $\mathbf{x} = [x_1, \ldots, x_{N_v}]^\top$; and
- Edge status is only observed for some subset of pairs $V^{(2)}_{obs} \subset V^{(2)}$
- **Goal:** predict edge status for all other pairs, i.e., $V^{(2)}_{miss} = V^{(2)} \setminus V^{(2)}_{obs}$

Association network inference



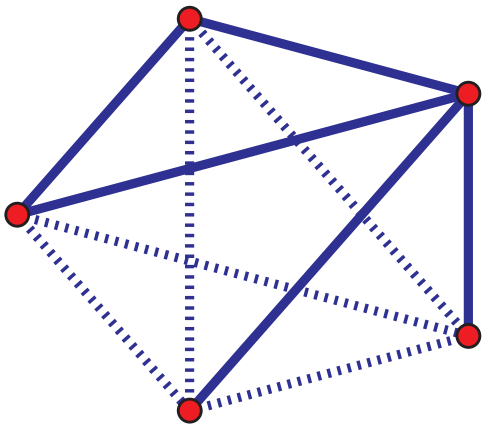
Original graph



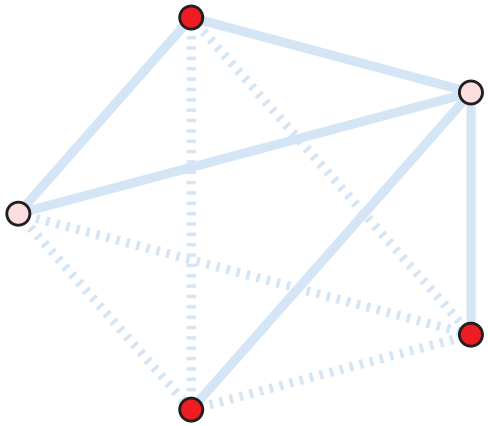
Association network inference

- Suppose we only observe vertex attributes $\mathbf{x} = [x_1, \ldots, x_{N_v}]^\top$; and
- Assume (i, j) defined by nontrivial 'level of association' among x_i, x_j
- **Goal:** predict edge status for all vertex pairs $V^{(2)}$

Tomographic network topology inference



Original graph



Tomographic inference

- Suppose we only observe x_i for vertices $i \subset V$ in the 'perimeter' of G
- **Goal:** predict edge and vertex status in the 'interior' of G

Link prediction

Network sampling and challenges

Background on statistical sampling theory

Graph sampling designs

Estimation of network totals, group size, and degree distributions

Network topology inference problems

Link prediction

Inference of association networks

Tomographic network topology inference

Link prediction

- Let $G(V, E)$ be a random graph, with adjacency matrix $\mathbf{Y} \in \{0, 1\}^{N_v \times N_v}$
  - $\Rightarrow$ $\mathbf{Y}^{obs}$ and $\mathbf{Y}^{miss}$ denote entries in $V^{(2)}_{obs}$ and $V^{(2)}_{miss}$

Link prediction

Predict entries in $\mathbf{Y}^{miss}$, given observations $\mathbf{Y}^{obs} = \mathbf{y}^{obs}$ and possibly various vertex attributes $\mathbf{X} = \mathbf{x} \in \mathbb{R}^{N_v}$

- Edge status information may be missing due to:
  - $\Rightarrow$ Difficulty in observation, issues of sampling
  - $\Rightarrow$ Edge is not yet present, wish to predict future status

Link prediction

- Let $G(V, E)$ be a random graph, with adjacency matrix $\mathbf{Y} \in \{0,1\}^{N_v \times N_v}$
  - $\Rightarrow$ $\mathbf{Y}^{obs}$ and $\mathbf{Y}^{miss}$ denote entries in $V^{(2)}_{obs}$ and $V^{(2)}_{miss}$

Link prediction

Predict entries in $\mathbf{Y}^{miss}$, given observations $\mathbf{Y}^{obs} = \mathbf{y}^{obs}$ and possibly various vertex attributes $\mathbf{X} = \mathbf{x} \in \mathbb{R}^{N_v}$

- Edge status information may be missing due to:
  - $\Rightarrow$ Difficulty in observation, issues of sampling
  - $\Rightarrow$ Edge is not yet present, wish to predict future status
- Given a model for $\mathbf{X}$ and $(\mathbf{Y}^{obs}, \mathbf{Y}^{miss})$, jointly predict $\mathbf{Y}^{miss}$ based on

$$\mathrm{P}\left[\mathbf{Y}^{miss} \,\middle|\, \mathbf{Y}^{obs} = \mathbf{y}^{obs}, \mathbf{X} = \mathbf{x}\right]$$

  - $\Rightarrow$ More manageable to predict the variables Y^{miss}_{ij} individually

Informal scoring methods

- Idea: compute score $s(i,j)$ for missing 'potential edges' $\{i,j\} \in V_{miss}^{(2)}$

  $\Rightarrow$ Predicted edges returned by retaining the top n^* scores

Informal scoring methods

- Idea: compute score $s(i,j)$ for missing 'potential edges' $\{i,j\} \in V_{miss}^{(2)}$

  $\Rightarrow$ Predicted edges returned by retaining the top n^* scores

- Scores designed to assess certain local structural properties of G^{obs}

  $\Rightarrow$ Distance-based, inspired by the small-world principle

$$s(i,j) = -\text{dist}_{G^{obs}}(i,j)$$

  $\Rightarrow$ Neighborhood-based, e.g., the number of common neighbors

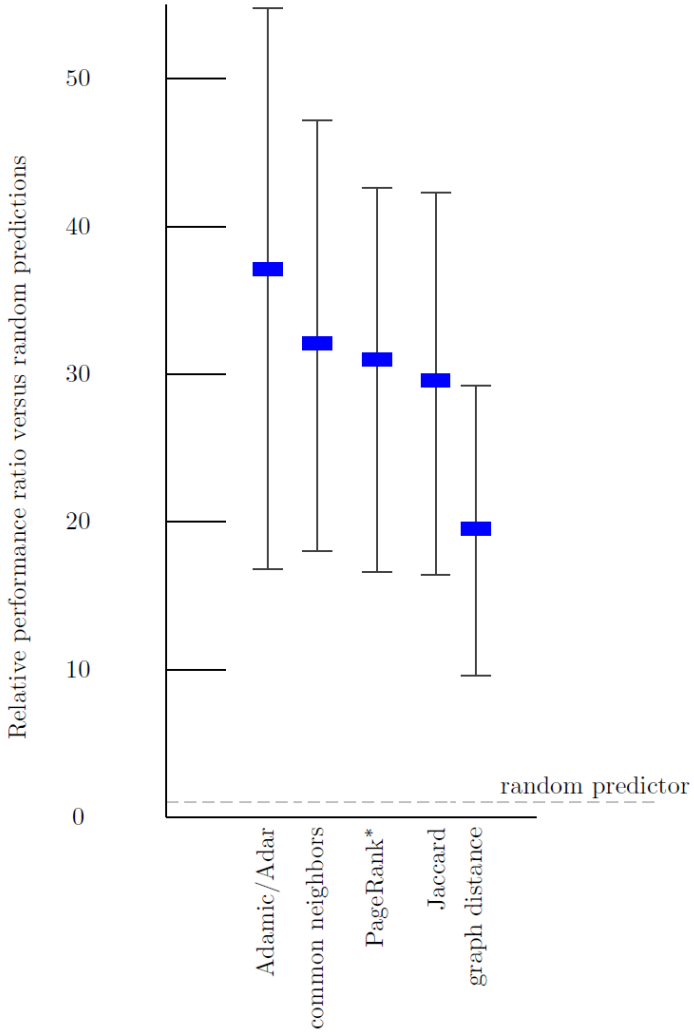
$$s(i,j) = |\mathcal{N}_i^{obs} \cap \mathcal{N}_j^{obs}| \quad \text{or} \quad s(i,j) = \frac{|\mathcal{N}_i^{obs} \cap \mathcal{N}_j^{obs}|}{|\mathcal{N}_i^{obs} \cup \mathcal{N}_j^{obs}|}$$

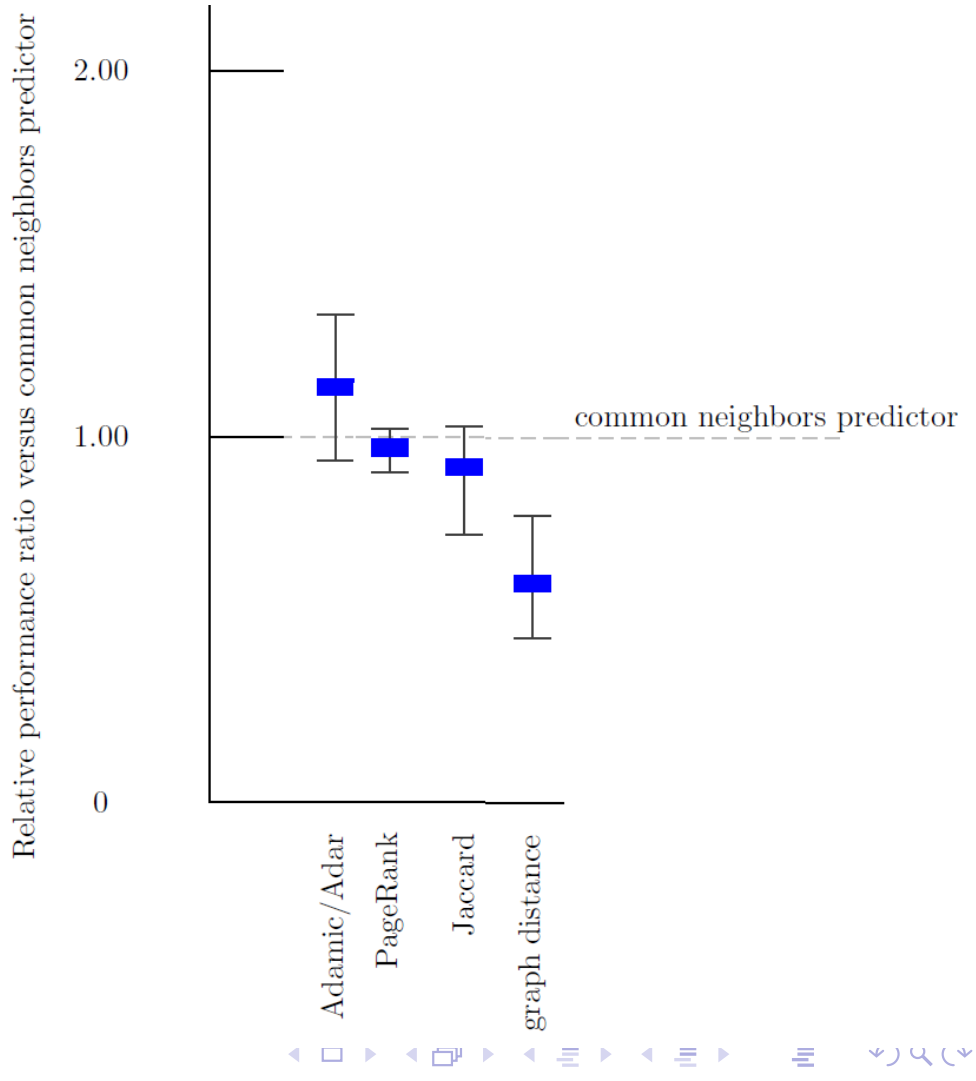
  $\Rightarrow$ Favor loosely-connected common neighbors [Adamic-Adar'03]

$$s(i,j) = \sum_{k \in \mathcal{N}_i^{obs} \cap \mathcal{N}_j^{obs}} \frac{1}{\log |\mathcal{N}_k^{obs}|}$$

Tests on co-authorship networks

- Results from a link prediction study in [Liben Nowell-Kleinberg'03]





Classification methods

- Idea: use training data $\mathbf{y}^{obs}$ and $\mathbf{x}$ to build a binary classifier
  - $\Rightarrow$ Classifier is in turn used to predict the entries in $\mathbf{Y}^{miss}$

- Idea: use training data $\mathbf{y}^{obs}$ and $\mathbf{x}$ to build a binary classifier
  $\Rightarrow$ Classifier is in turn used to predict the entries in $\mathbf{Y}^{miss}$
- Logistic regression classifiers most popular, based on the model

$$\log\left[\frac{\mathrm{P}_\beta(Y_{ij}=1\,|\,\mathbf{Z}_{ij}=\mathbf{z})}{\mathrm{P}_\beta(Y_{ij}=0\,|\,\mathbf{Z}_{ij}=\mathbf{z})}\right]=\boldsymbol{\beta}^\top\mathbf{z},\quad\text{where}$$

(i) $\boldsymbol{\beta}\in\mathbb{R}^K$ is a vector of regression coefficients; and
(ii) $\mathbf{Z}_{ij}$ is a vector of explanatory variables indexed by $\{i,j\}$

$$\mathbf{Z}_{ij}=[g_1(\mathbf{Y}^{obs}_{(-ij)},\mathbf{X}),\ldots,g_K(\mathbf{Y}^{obs}_{(-ij)},\mathbf{X})]^\top$$

- Functions $g_k(\cdot)$ encode useful predictive information in $\mathbf{y}^{obs}_{(-ij)}$ and $\mathbf{x}$
  Ex: vertex attributes, score functions, network statistics in ERGMs

Logistic regression classifier

- Train: Obtain MLE $\hat{\boldsymbol{\beta}}$ via iteratively-reweighted LS
- Test: Potential edges (i, j) declared present based on probabilities

$$\mathrm{P}_{\hat{\boldsymbol{\beta}}}(Y_{ij} = 1 \,|\, \mathbf{Z}_{ij} = \mathbf{z}) = \frac{\exp\left(\hat{\boldsymbol{\beta}}^{\top} \mathbf{z}\right)}{1 + \exp\left(\hat{\boldsymbol{\beta}}^{\top} \mathbf{z}\right)}$$

Logistic regression classifier

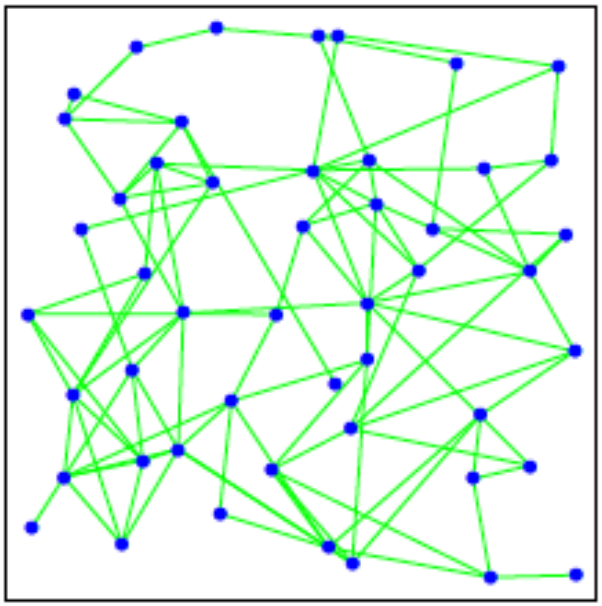
- Train: Obtain MLE $\hat{\boldsymbol{\beta}}$ via iteratively-reweighted LS
- Test: Potential edges (i, j) declared present based on probabilities

$$\mathrm{P}_{\hat{\boldsymbol{\beta}}}(Y_{ij} = 1 \,|\, \mathbf{Z}_{ij} = \mathbf{z}) = \frac{\exp\left(\hat{\boldsymbol{\beta}}^{\top} \mathbf{z}\right)}{1 + \exp\left(\hat{\boldsymbol{\beta}}^{\top} \mathbf{z}\right)}$$

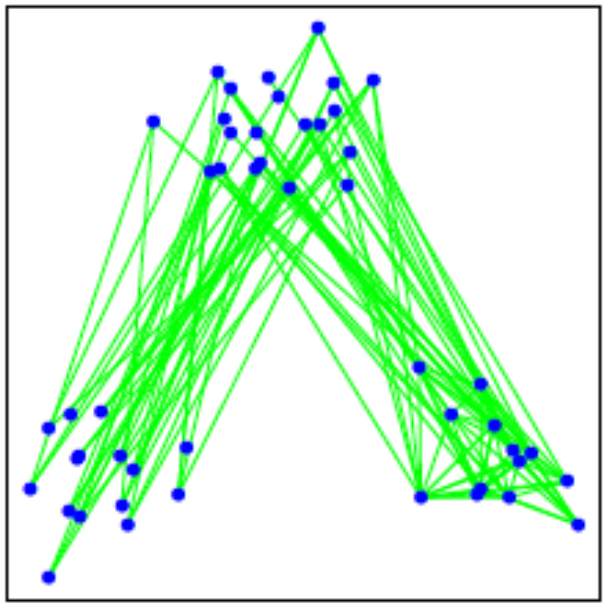
- Logistic regression assumes $\mathbf{Y}_{ij}$ conditionally independent given $\mathbf{z}$
  $\Rightarrow$ Seldom the case with relational network data
- Underlying mechanism of data missingness is important
  $\Rightarrow$ Classification for link prediction reminiscent of cross-validation
  $\Rightarrow$ Assumption that data are missing at random is fundamental

Latent variable models

- In addition to a linear predictor $\boldsymbol{\beta}^\top \mathbf{z}$, latent models describe Y_{ij}
  $\Rightarrow$ As a function of vertex-specific latent variables $\mathbf{u}_i$ and $\mathbf{u}_j$



Homophily



Stochastic equivalence

- Latent models are flexible to capture underlying social mechanisms
  Ex: homophily (transitivity) and stochastic equivalence (groups)

Latent class and distance models

- Latent distance model: node i has unobserved position $\mathbf{U}_i \in \mathbb{R}^d$
  - Positions $\mathbf{U}_i$ in latent space assumed i.i.d. e.g., Gaussian distributed
  - Model cond. probability of edge Y_{ij} as function of $\boldsymbol{\beta}^\top \mathbf{z} - \|\mathbf{u}_i - \mathbf{u}_j\|_2$
  - Homophily: Nearby nodes in latent space more likely to link

Latent class and distance models

- Latent distance model: node i has unobserved position $\mathbf{U}_i \in \mathbb{R}^d$
  - Positions $\mathbf{U}_i$ in latent space assumed i.i.d. e.g., Gaussian distributed
  - Model cond. probability of edge Y_{ij} as function of $\beta^\top \mathbf{z} - \|\mathbf{u}_i - \mathbf{u}_j\|_2$
  - Homophily: Nearby nodes in latent space more likely to link
- Latent class model: node i belongs to unobserved class $U_i \in \{1, \ldots, k\}$
  - Classes U_i assumed i.i.d. e.g., multinomial distributed
  - Model cond. probability of edge Y_{ij} as function of $\beta^\top \mathbf{z} - \theta_{u_i, u_j}$
  - Stochastic equivalence: Nodes in same class equally likely to link
- P. D. Hoff, "Modeling homophily and stochastic equivalence in symmetric relational data," *NIPS*, 2008

Logistic regression with latent variables

- Let $\mathbf{M} \in \mathbb{R}^{N_v \times N_v}$ be unknown, random, and symmetric of the form

$$\mathbf{M} = \mathbf{U}^\top \mathbf{\Lambda} \mathbf{U} + \mathbf{E}, \quad \text{where}$$

(i) $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_{N_v}]$ is a random orthonormal matrix of latent variables;
(ii) $\mathbf{\Lambda}$ is a random diagonal matrix; and
(iii) $\mathbf{E}$ is a symmetric matrix of i.i.d. noise entries ϵ_{ij}

- Latent eigenmodel subsumes the class and distance variants [Hoff'08]

  $\Rightarrow$ Notice that $M_{ij} = \mathbf{u}_i^T \mathbf{\Lambda} \mathbf{u}_j + \epsilon_{ij}$

Logistic regression with latent variables

- Let $\mathbf{M} \in \mathbb{R}^{N_v \times N_v}$ be unknown, random, and symmetric of the form

$$\mathbf{M} = \mathbf{U}^\top \mathbf{\Lambda} \mathbf{U} + \mathbf{E}, \quad \text{where}$$

  (i) $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_{N_v}]$ is a random orthonormal matrix of latent variables;
  (ii) $\mathbf{\Lambda}$ is a random diagonal matrix; and
  (iii) $\mathbf{E}$ is a symmetric matrix of i.i.d. noise entries ϵ_{ij}

- Latent eigenmodel subsumes the class and distance variants [Hoff'08]

  $\Rightarrow$ Notice that $M_{ij} = \mathbf{u}_i^T \mathbf{\Lambda} \mathbf{u}_j + \epsilon_{ij}$

- The logistic regression model with latent variables is

$$\log \left[\frac{\mathrm{P}_\beta(Y_{ij} = 1 \,|\, \mathbf{Z}_{ij} = \mathbf{z}, M_{ij} = m)}{\mathrm{P}_\beta(Y_{ij} = 0 \,|\, \mathbf{Z}_{ij} = \mathbf{z}, M_{ij} = m)} \right] = \boldsymbol{\beta}^\top \mathbf{z} + m$$

- Y_{ij} still assumed conditionally independent given $\mathbf{Z}_{ij}$ and M_{ij}

  $\Rightarrow$ But they are conditionally dependent given only $\mathbf{Z}_{ij}$

Bayesian link prediction

- Specify distributions for $\mathbf{U}, \mathbf{\Lambda}, \mathbf{E}$ to make statistical link predictions
  - Bayesian inference natural $\Rightarrow$ Specify a prior for β as well

Bayesian link prediction

- Specify distributions for $\mathbf{U}, \mathbf{\Lambda}, \mathbf{E}$ to make statistical link predictions
  - Bayesian inference natural $\Rightarrow$ Specify a prior for β as well
- To predict those entries in $\mathbf{Y}^{miss}$, threshold the posterior mean

$$\mathbb{E}\left[\frac{\exp\left(\beta^\top \mathbf{Z}_{ij} + M_{ij}\right)}{1+\exp\left(\beta^\top \mathbf{Z}_{ij} + M_{ij}\right)} \,\middle|\, \mathbf{Y}^{obs} = \mathbf{y}^{obs}, \mathbf{Z}_{ij} = \mathbf{z}\right]$$

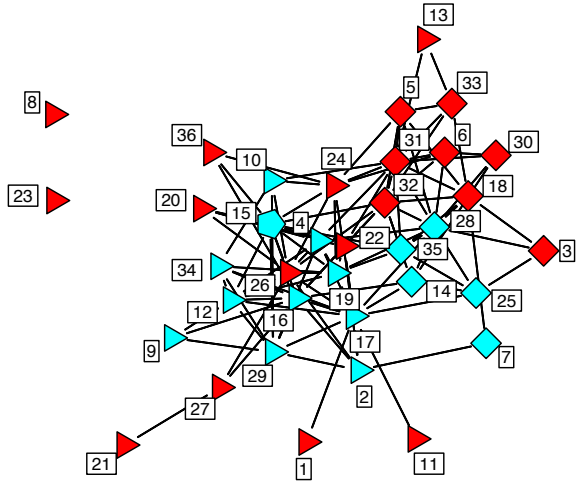
- Use MCMC algorithms to approximate the posterior distribution
  - Gaussian distributions attractive for their conjugacy properties
- Higher complexity than MLE for standard logistic regression

  $\Rightarrow$ Need to generate draws for N_v^2 unobserved variables $\{U_{ij}\}$

  $\Rightarrow$ Major cost reduction with reduced $\text{rank}(\mathbf{U}) = k \ll N_v$ models

Case study: Lawyer collaboration network

- Network G^{obs} of working relationships among lawyers [Lazega'01]
  - Nodes are $N_v = 36$ partners, edges indicate partners worked together



- Data includes various node-level attributes:
  - Seniority (node labels indicate rank ordering)
  - Office location (triangle, square or pentagon)
  - Type of practice, i.e., litigation (red) and corporate (cyan)
  - Gender (three partners are female labeled 27, 29 and 34)
- Goal: predict cooperation among social actors in an organization

Methods to predict lawyer collaboration

- Define the following set of explanatory variables:

$$Z_{ij}^{(1)} = \text{seniority}_i + \text{seniority}_j, \quad Z_{ij}^{(2)} = \text{practice}_i + \text{practice}_j$$
$$Z_{ij}^{(3)} = \mathbb{I}\{\text{practice}_i = \text{practice}_j\}, \quad Z_{ij}^{(4)} = \mathbb{I}\{\text{gender}_i = \text{gender}_j\}$$
$$Z_{ij}^{(5)} = \mathbb{I}\{\text{office}_i = \text{office}_j\}, \quad Z_{ij}^{(6)} = |\mathcal{N}_i^{obs} \cap \mathcal{N}_j^{obs}|$$

Methods to predict lawyer collaboration

- Define the following set of explanatory variables:

$$Z_{ij}^{(1)} = \text{seniority}_i + \text{seniority}_j, \quad Z_{ij}^{(2)} = \text{practice}_i + \text{practice}_j$$
$$Z_{ij}^{(3)} = \mathbb{I}\{\text{practice}_i = \text{practice}_j\}, \quad Z_{ij}^{(4)} = \mathbb{I}\{\text{gender}_i = \text{gender}_j\}$$
$$Z_{ij}^{(5)} = \mathbb{I}\{\text{office}_i = \text{office}_j\}, \quad Z_{ij}^{(6)} = |\mathcal{N}_i^{obs} \cap \mathcal{N}_j^{obs}|$$

Method 1: standard logistic regression with $Z_{ij}^{(1)}, \ldots, Z_{ij}^{(5)}$

Method 2: standard logistic regression with $Z_{ij}^{(1)}, \ldots, Z_{ij}^{(6)}$

Method 3: informal scoring method with $s(i,j) = Z_{ij}^{(6)}$

Method 4: logistic regression with $Z_{ij}^{(1)}, \ldots, Z_{ij}^{(5)}$ and latent eigenmodel

Methods to predict lawyer collaboration

- Define the following set of explanatory variables:

$$Z_{ij}^{(1)} = \text{seniority}_i + \text{seniority}_j, \quad Z_{ij}^{(2)} = \text{practice}_i + \text{practice}_j$$
$$Z_{ij}^{(3)} = \mathbb{I}\{\text{practice}_i = \text{practice}_j\}, \quad Z_{ij}^{(4)} = \mathbb{I}\{\text{gender}_i = \text{gender}_j\}$$
$$Z_{ij}^{(5)} = \mathbb{I}\{\text{office}_i = \text{office}_j\}, \quad Z_{ij}^{(6)} = |\mathcal{N}_i^{obs} \cap \mathcal{N}_j^{obs}|$$

  Method 1: standard logistic regression with $Z_{ij}^{(1)}, \ldots, Z_{ij}^{(5)}$

  Method 2: standard logistic regression with $Z_{ij}^{(1)}, \ldots, Z_{ij}^{(6)}$

  Method 3: informal scoring method with $s(i,j) = Z_{ij}^{(6)}$

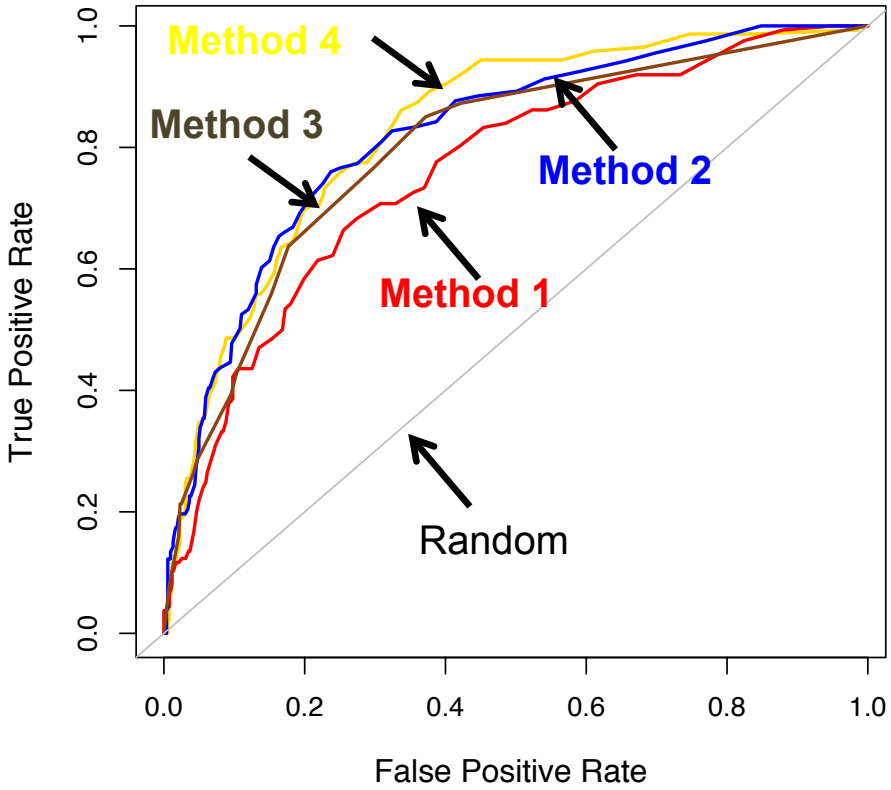
  Method 4: logistic regression with $Z_{ij}^{(1)}, \ldots, Z_{ij}^{(5)}$ and latent eigenmodel

- Five-fold cross-validation over the set of $36(36-1)/2 = 630$ vertex pairs

  $\Rightarrow$ For each fold, $630/5 = 126$ pairs in $\mathbf{Y}^{miss}$ and the rest in $\mathbf{Y}^{obs}$

Receiver operating characteristic

- Receiver operating characteristic curves show predictive performance



- Method 1 performs worst $\Rightarrow$ Agnostic to network structure
- Informal Method 3 yields slightly worst performance than 2 and 4

Inference of association networks

Network sampling and challenges

Background on statistical sampling theory

Graph sampling designs

Estimation of network totals, group size, and degree distributions

Network topology inference problems

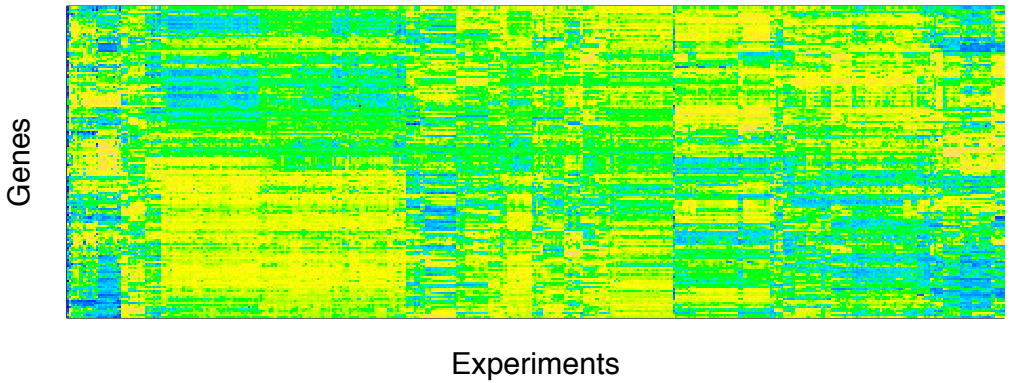
Link prediction

Inference of association networks

Tomographic network topology inference

Association networks

- **Def**: in association networks vertices are linked if there is a sufficient level of 'association' between attributes of vertex pairs



Example

- Scientific citation networks
- Movie networks
- Gene-regulatory networks
- Neuro-functional connectivity networks

Association network inference

- Given a collection of N_v elements represented as vertices $v \in V$
  - Let $\mathbf{x}_i \in \mathbb{R}^m$ be a vector of observed vertex attributes, for all $i \in V$
- User-defined similarity $\texttt{sim}(i,j) = f(\mathbf{x}_i, \mathbf{x}_j)$ specifies edges $(i,j) \in E$
  - Q: What if `sim` values themselves (i.e., edge status) not observable?

Association network inference

- Given a collection of N_v elements represented as vertices $v \in V$
  - Let $\mathbf{x}_i \in \mathbb{R}^m$ be a vector of observed vertex attributes, for all $i \in V$
- User-defined similarity $\texttt{sim}(i,j) = f(\mathbf{x}_i, \mathbf{x}_j)$ specifies edges $(i,j) \in E$
  - Q: What if `sim` values themselves (i.e., edge status) not observable?

Association network inference

Infer non-trivial `sim` values from vertex observations $\{\mathbf{x}_1, \ldots, \mathbf{x}_{N_v}\}$

- Various choices to be made, hence multiple possible approaches
  - Choice of `sim`: correlation, partial correlation, mutual information
  - Choice of inference: hypothesis testing, regression, ad hoc
  - Choice of parameters: testing thresholds, tuning regularization

Correlation networks

- Let $X_i \in \mathbb{R}$ be an RV of interest corresponding to $i \in V$
- Pearson product-moment correlation as $\texttt{sim}$ between vertex pairs

$$\texttt{sim}(i,j) := \rho_{ij} = \frac{\text{cov}[X_i, X_j]}{\sqrt{\text{var}\,[X_i]\,\text{var}\,[X_j]}}, \quad i,j \in V$$

Correlation networks

- Let $X_i \in \mathbb{R}$ be an RV of interest corresponding to $i \in V$
- Pearson product-moment correlation as $\texttt{sim}$ between vertex pairs

$$\texttt{sim}(i,j) := \rho_{ij} = \frac{\text{cov}[X_i, X_j]}{\sqrt{\text{var}\,[X_i]\,\text{var}\,[X_j]}}, \quad i,j \in V$$

- **Def:** the correlation network $G(V,E)$ has edge set

$$E = \left\{(i,j) \in V^{(2)} : \rho_{ij} \neq 0\right\}$$

  - Association network inference $\Leftrightarrow$ Inference of non-zero correlations
- Inference of E typically approached as a testing problem

$$H_0 : \rho_{ij} = 0 \quad \text{versus} \quad H_1 : \rho_{ij} \neq 0$$

Test statistics

- Let $x_{i1}, \ldots, x_{in}$ be observations of zero-mean X_i, for each $i \in V$

  $\Rightarrow$ Common choice of test statistic are empirical correlations

$$\hat{\rho}_{ij} = \frac{\hat{\sigma}_{ij}}{\sqrt{\hat{\sigma}_{ii}\hat{\sigma}_{jj}}}, \quad \text{where } \hat{\mathbf{\Sigma}} = [\hat{\sigma}_{ij}] = \frac{\mathbf{X}^\top \mathbf{X}}{n-1}$$

- Convenient alternative statistic is Fisher's transformation

$$z_{ij} = \frac{1}{2} \log \left(\frac{1+\hat{\rho}_{ij}}{1-\hat{\rho}_{ij}} \right), \quad i,j \in V$$

  $\Rightarrow$ Under H_0 and Gaussian assumption

  $\Rightarrow$ $z_{ij} \sim \mathcal{N}(0, \frac{1}{n-3}) \Rightarrow$ Simple to assess significance

Test statistics

- Let $x_{i1}, \ldots, x_{in}$ be observations of zero-mean X_i, for each $i \in V$

  $\Rightarrow$ Common choice of test statistic are empirical correlations

$$\hat{\rho}_{ij} = \frac{\hat{\sigma}_{ij}}{\sqrt{\hat{\sigma}_{ii}\hat{\sigma}_{jj}}}, \quad \text{where } \hat{\mathbf{\Sigma}} = [\hat{\sigma}_{ij}] = \frac{\mathbf{X}^\top \mathbf{X}}{n-1}$$

- Convenient alternative statistic is Fisher's transformation

$$z_{ij} = \frac{1}{2} \log \left(\frac{1+\hat{\rho}_{ij}}{1-\hat{\rho}_{ij}} \right), \quad i,j \in V$$

  $\Rightarrow$ Under H_0 and Gaussian assumption

  $\Rightarrow$ $z_{ij} \sim \mathcal{N}(0, \frac{1}{n-3}) \Rightarrow$ Simple to assess significance

- Reject H_0 at significance level α, i.e., assign edge (i,j) if $|z_{ij}| > \frac{z_{\alpha/2}}{\sqrt{n-3}}$

$$\text{Error rate control: } \mathrm{P}_{H_0}(\text{false edge}) = \mathrm{P}_{H_0}\left(|z_{ij}| > \frac{z_{\alpha/2}}{\sqrt{n-3}} \right) = \alpha$$

Networks and multiple testing

- Interesting testing challenges emerge with large-scale networks
  - $\Rightarrow$ Suppose we test all $\binom{N_v}{2}$ vertex pairs, each at level α
- Even if the true G is the empty graph, i.e., $E = \emptyset$
  - $\Rightarrow$ We expect to declare $\binom{N_v}{2}\alpha$ spurious edges just by chance!
  - $\Rightarrow$ For a large graph, this number can be considerable

Networks and multiple testing

- Interesting testing challenges emerge with large-scale networks
  - $\Rightarrow$ Suppose we test all $\binom{N_v}{2}$ vertex pairs, each at level α
- Even if the true G is the empty graph, i.e., $E = \emptyset$
  - $\Rightarrow$ We expect to declare $\binom{N_v}{2}\alpha$ spurious edges just by chance!
  - $\Rightarrow$ For a large graph, this number can be considerable
- Ex: For G of order $N_v = 100$ and individual tests at level $\alpha = 0.05$
  - $\Rightarrow$ Expected number of spurious edges is $4950 \times 0.05 \approx 250$
- This predicament known as the multiple testing problem in statistics

Correction for multiple testing

- Idea: Control errors at the level of collection of tests, not individually
- False discovery rate (FDR) control, i.e., for given level γ ensure

$$\text{FDR} = \mathbb{E}\left[\frac{R_{false}}{R} \mid R > 0\right] \text{P}\left[R > 0\right] \leq \gamma$$

  - R is the total number of edges detected; and
  - R_{false} is the total number of false edges detected

Correction for multiple testing

- Idea: Control errors at the level of collection of tests, not individually
- False discovery rate (FDR) control, i.e., for given level γ ensure

$$\text{FDR} = \mathbb{E}\left[\frac{R_{false}}{R} \mid R > 0\right] \text{P}\left[R > 0\right] \leq \gamma$$

  - R is the total number of edges detected; and
  - R_{false} is the total number of false edges detected

- Method of FDR control at level γ [Benjamini-Hochberg'94]

  Step 1: Sort p-values for all $N = \binom{N_v}{2}$ tests, yields $p_{(1)} \leq \ldots \leq p_{(N)}$

  Step 2: Reject H_0, i.e., declare all those edges for which

$$p_{(k)} \leq \left(\frac{k}{N}\right)\gamma$$

Gene-regulatory interactions

- Genes are segments of DNA encoding information about cell functions
- Such information used in the expression of genes
  $\Rightarrow$ Creation of biochemical products, i.e., RNA or proteins

Gene-regulatory interactions

- Genes are segments of DNA encoding information about cell functions
- Such information used in the expression of genes
  - $\Rightarrow$ Creation of biochemical products, i.e., RNA or proteins
- Regulation of a gene refers to the control of its expression

  Ex: regulation exerted during transcription, copy of DNA to RNA
  - $\Rightarrow$ Controlling genes are transcription factors (TFs)
  - $\Rightarrow$ Controlled genes are termed targets
  - $\Rightarrow$ Regulation type: activation or repression

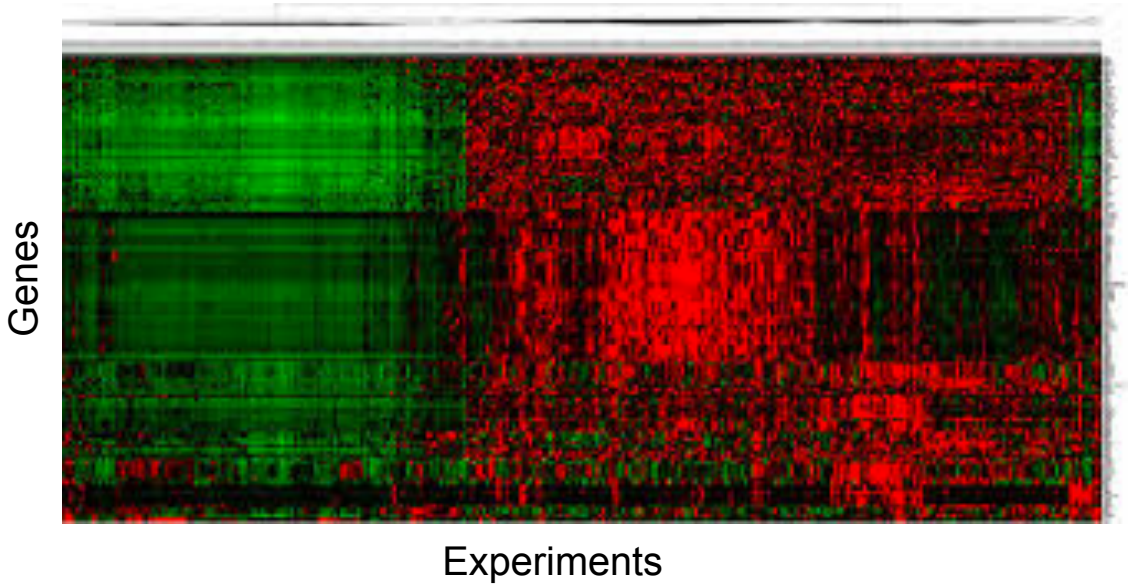
Gene-regulatory interactions

- Genes are segments of DNA encoding information about cell functions
- Such information used in the expression of genes
  - $\Rightarrow$ Creation of biochemical products, i.e., RNA or proteins
- Regulation of a gene refers to the control of its expression

  Ex: regulation exerted during transcription, copy of DNA to RNA
  - $\Rightarrow$ Controlling genes are transcription factors (TFs)
  - $\Rightarrow$ Controlled genes are termed targets
  - $\Rightarrow$ Regulation type: activation or repression
- Regulatory interactions among genes basic to the workings of organisms
  - $\Rightarrow$ Inference of interactions $\rightarrow$ Finding TF/target gene pairs
- Such relational information summarized in gene-regulatory networks

Microarray data

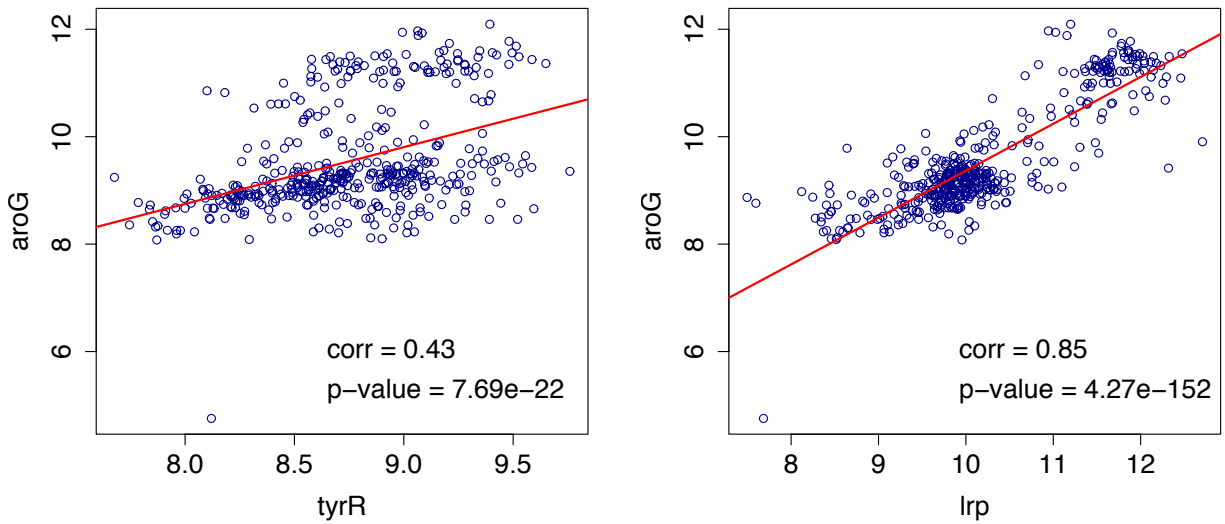
- Relative levels of gene expression in the cell can be measured
  - $\Rightarrow$ Genome-wide scale data obtained using microarray technologies



- For each gene $i \in V$, measure an expression profile $\mathbf{x}_i \in \mathbb{R}^n$
  - Vector $\mathbf{x}_i$ has gene expression levels under n different conditions
  - Ex: change in pH, heat level, oxygen concentrations
- Microarray data commonly used to infer gene regulatory interactions

Example: gene expression level correlations

- Microarray data for the bacteria Escherichia coli (E. coli)
  - Two TFs *tyrR* and *lrp*, potential target *aroG* over $n = 445$ experiments
  - Ground truth: *aroG* is regulated by *tyrR* but not *lrp*



- Fisher scores: $z_{tyrR}^{aroG} = 0.4599$ and $z_{lrp}^{aroG} = 1.2562$. Both p-values small
- Based on correlations, *aroG* strongly associated with both *tyrR* and *lrp*

Partial correlations

- Use correlations carefully: 'correlation does not imply causation'
  - Vertices $i, j \in V$ may have high ρ_{ij} because they influence each other
- But ρ_{ij} could be high if both i, j influenced by a third vertex $k \in V$

  $\Rightarrow$ Correlation networks may declare edges due to latent variables

Partial correlations

- Use correlations carefully: 'correlation does not imply causation'
  - Vertices $i, j \in V$ may have high ρ_{ij} because they influence each other
- But ρ_{ij} could be high if both i, j influenced by a third vertex $k \in V$

  $\Rightarrow$ Correlation networks may declare edges due to latent variables
- Partial correlations better capture direct influence among vertices
  - For $i, j \in V$ consider latent vertices $S_m = \{k_1, \ldots, k_m\} \subset V \setminus \{i, j\}$
- Partial correlation of X_i and X_j, adjusting for $\mathbf{X}_{S_m} = [X_{k_1}, \ldots, X_{k_m}]^\top$ is

$$\rho_{ij|S_m} = \frac{\text{cov}[X_i, X_j \,|\, \mathbf{X}_{S_m}]}{\sqrt{\text{var}\left[X_i \,|\, \mathbf{X}_{S_m}\right] \text{var}\left[X_j \,|\, \mathbf{X}_{S_m}\right]}}, \quad i, j \in V$$

- Q: How do we obtain these partial correlations?

Computing partial correlations

- Given $\mathbf{X}_{S_m} = [X_{k_1}, \ldots, X_{k_m}]^\top$, the partial correlation of X_i and X_j is

$$\rho_{ij|S_m} = \frac{\text{cov}[X_i, X_j \,|\, \mathbf{X}_{S_m}]}{\sqrt{\text{var}\left[X_i \,|\, \mathbf{X}_{S_m}\right] \text{var}\left[X_j \,|\, \mathbf{X}_{S_m}\right]}} = \frac{\sigma_{ij|S_m}}{\sqrt{\sigma_{ii|S_m}\sigma_{jj|S_m}}}$$

Computing partial correlations

- Given $\mathbf{X}_{S_m} = [X_{k_1}, \ldots, X_{k_m}]^\top$, the partial correlation of X_i and X_j is

$$\rho_{ij|S_m} = \frac{\text{cov}[X_i, X_j \,|\, \mathbf{X}_{S_m}]}{\sqrt{\text{var}\left[X_i \,|\, \mathbf{X}_{S_m}\right] \text{var}\left[X_j \,|\, \mathbf{X}_{S_m}\right]}} = \frac{\sigma_{ij|S_m}}{\sqrt{\sigma_{ii|S_m}\sigma_{jj|S_m}}}$$

- Here $\sigma_{ii|S_m}, \sigma_{jj|S_m}$ and $\sigma_{ij|S_m}$ are diagonal and off-diagonal elements of

$$\mathbf{\Sigma}_{11|2} := \mathbf{\Sigma}_{11} - \mathbf{\Sigma}_{12}\mathbf{\Sigma}_{22}^{-1}\mathbf{\Sigma}_{21} \in \mathbb{R}^{2\times 2}$$

- Matrices $\mathbf{\Sigma}_{11}$, $\mathbf{\Sigma}_{22}$ and $\mathbf{\Sigma}_{21} = \mathbf{\Sigma}_{12}^\top$ are blocks of the covariance matrix

$$\text{cov}\begin{bmatrix} \mathbf{W}_1 \\ \mathbf{W}_2 \end{bmatrix} = \begin{pmatrix} \mathbf{\Sigma}_{11} & \mathbf{\Sigma}_{12} \\ \mathbf{\Sigma}_{21} & \mathbf{\Sigma}_{22} \end{pmatrix}, \text{ where } \mathbf{W}_1 = [X_i, X_j]^\top \text{ and } \mathbf{W}_2 = \mathbf{X}_{S_m}$$

Partial correlation networks

- Various ways to use partial correlations to define edges in G

  Ex: X_i, X_j correlated regardless of what m vertices we condition upon

$$E = \left\{ (i,j) \in V^{(2)} : \rho_{ij|S_m} \neq 0, \text{ for all } S_m \in V^{(m)}_{\setminus\{i,j\}} \right\}$$

Partial correlation networks

- Various ways to use partial correlations to define edges in G

  Ex: X_i, X_j correlated regardless of what m vertices we condition upon

$$E = \left\{(i,j) \in V^{(2)} : \rho_{ij|S_m} \neq 0, \text{ for all } S_m \in V^{(m)}_{\setminus\{i,j\}}\right\}$$

- Inference of potential edge (i,j) as a testing problem

$$H_0 : \rho_{ij|S_m} = 0 \text{ for some } S_m \in V^{(m)}_{\setminus\{i,j\}}$$

$$H_1 : \rho_{ij|S_m} \neq 0 \text{ for all } S_m \in V^{(m)}_{\setminus\{i,j\}}$$

- Again, given measurements $x_{i1}, \ldots, x_{in}$ for each $i \in V$ need to:
  - Select a test statistic
  - Construct an appropriate null distribution
  - Adjust for multiple testing

Testing partial correlations

- Often consider a collection (over S_m) of smaller testing sub-problems

$$H_0' : \rho_{ij|S_m} = 0 \quad \text{versus} \quad H_1' : \rho_{ij|S_m} \neq 0$$

Testing partial correlations

- Often consider a collection (over S_m) of smaller testing sub-problems

$$H_0' : \rho_{ij|S_m} = 0 \quad \text{versus} \quad H_1' : \rho_{ij|S_m} \neq 0$$

- Statistic: empirical partial correlations $\hat{\rho}_{ij|S_m}$, or Fisher's z-scores

$$z_{ij|S_m} = \frac{1}{2} \log \left(\frac{1 + \hat{\rho}_{ij|S_m}}{1 - \hat{\rho}_{ij|S_m}} \right)$$

  $\Rightarrow$ From asymptotic theory, under H_0' then $z_{ij|S_m} \sim \mathcal{N}(0, \frac{1}{n-m-3})$

- Multiple tests for each $\{i, j\} \in V^{(2)}$. How do we combine p-values?

Testing partial correlations

- Often consider a collection (over S_m) of smaller testing sub-problems

$$H_0' : \rho_{ij|S_m} = 0 \quad \text{versus} \quad H_1' : \rho_{ij|S_m} \neq 0$$

- Statistic: empirical partial correlations $\hat{\rho}_{ij|S_m}$, or Fisher's z-scores

$$z_{ij|S_m} = \frac{1}{2} \log \left(\frac{1 + \hat{\rho}_{ij|S_m}}{1 - \hat{\rho}_{ij|S_m}} \right)$$

  $\Rightarrow$ From asymptotic theory, under H_0' then $z_{ij|S_m} \sim \mathcal{N}(0, \frac{1}{n-m-3})$

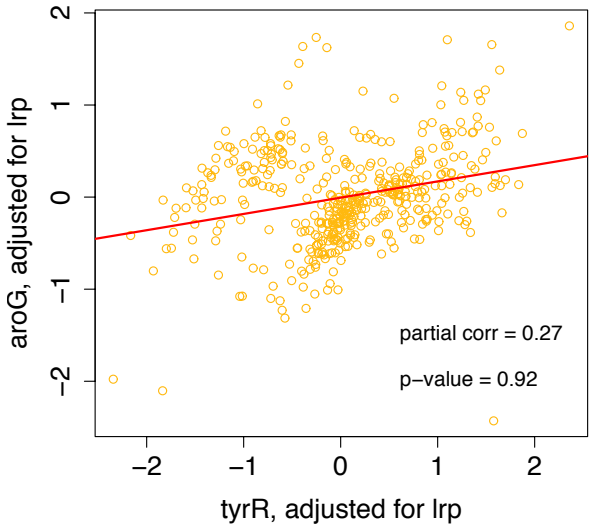
- Multiple tests for each $\{i, j\} \in V^{(2)}$. How do we combine p-values?
  - If $p_{ij|S_m}$ is the p-value for testing H_0' versus H_1' for $\{i, j\}$, use

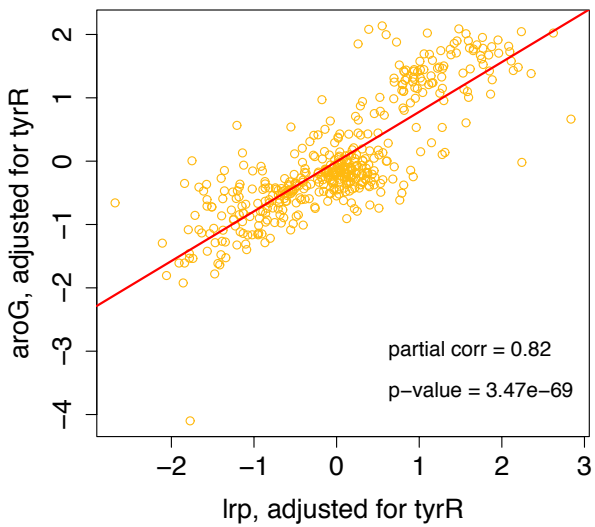
$$p_{ij}^{\max} = \max \left\{ p_{ij|S_m} : S_m \in V_{\setminus\{i,j\}}^{(m)} \right\}$$

  - FDR control possible from collection $\{p_{ij}^{\max}\}_{i,j}$ [Wille-Bühlmann'06]

Example: gene expression level partial correlations

- Nontrivial questions about measured TF/target gene pair correlation
  $\Rightarrow$ TF may be a target gene of another TF'
- Q: Direct influence or result from regulation of TF by other TF'?
- Partial correlation may sort out such confouding among variables
  - Partial correlations $\rho_{aroG,tyrR|lrp}$ and $\rho_{aroG,lrp|tyrR}$ for E. coli data

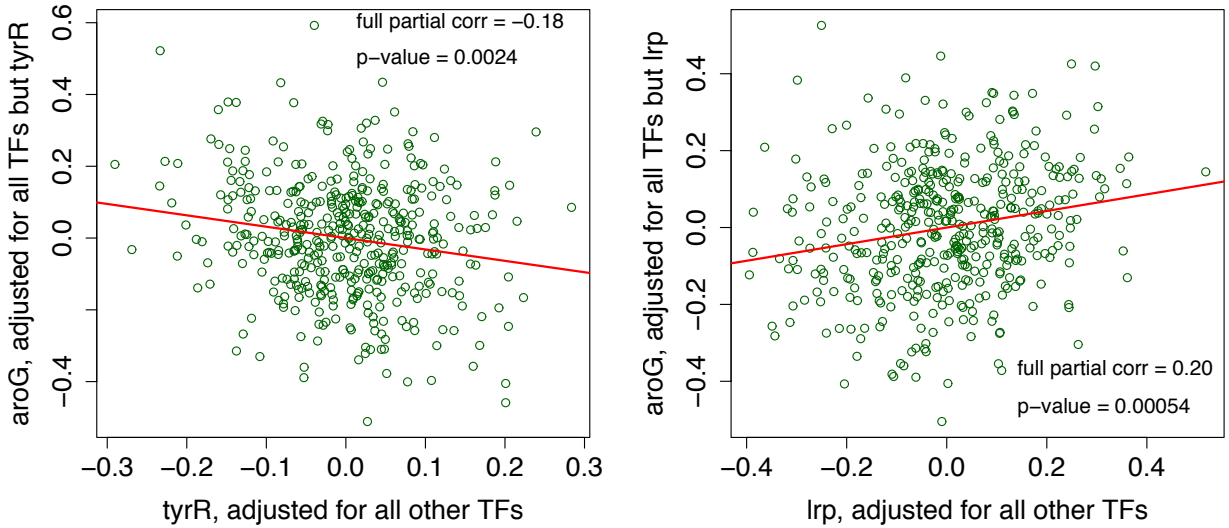




- Major drop $\rho_{aroG,tyrR|lrp} < \rho_{aroG,tyrR}$, no edge based on p-value 0.92

Full partial correlations

- Recompute partial correlations adjusting for all other $m = 152$ TFs



- Moderately strong evidence of association for both pairs
- The sign of the association between *aroG* and *tyrR* changed

  $\Rightarrow$ Suggests a repressive role of *tyrR* in regulating *aroG*
- Choices matter, e.g., the test statistic here. Interpret results carefully

- Suppose variables $\{X_i\}_{i \in V}$ have multivariate Gaussian distribution
  - $\Rightarrow$ Consider $\rho_{ij|V\setminus\{i,j\}}$ conditioning on all other vertices ($m = N_v - 2$)

Theorem

Under the Gaussian assumption, vertices $i, j \in V$ have partial correlation

$$\rho_{ij|V\setminus\{i,j\}} = 0$$

if and only if X_i and X_j are conditionally independent given $\{X_k\}_{k \in V\setminus\{i,j\}}$

- Suppose variables $\{X_i\}_{i \in V}$ have multivariate Gaussian distribution
  - $\Rightarrow$ Consider $\rho_{ij|V\setminus\{i,j\}}$ conditioning on all other vertices $(m = N_v - 2)$

Theorem

Under the Gaussian assumption, vertices $i, j \in V$ have partial correlation

$$\rho_{ij|V\setminus\{i,j\}} = 0$$

if and only if X_i and X_j are conditionally independent given $\{X_k\}_{k \in V\setminus\{i,j\}}$

- **Def:** the conditional independence graph $G(V, E)$ has edge set

$$E = \left\{(i,j) \in V^{(2)} : \rho_{ij|V\setminus\{i,j\}} \neq 0\right\}$$

  - $\Rightarrow$ A special and popular case of partial correlation networks
- Gaussian graphical model (GGM): Gaussian assumption along with G

Concentration matrix

- Let $\mathbf{\Sigma}$ be the covariance matrix of $\mathbf{X} = [X_1, \ldots, X_{N_v}]^T$

  Def: the concentration matrix is $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$ with entries ω_{ij}

- Key result: For GGMs, the partial correlations can be expressed as

$$\rho_{ij|V\setminus\{i,j\}} = -\frac{\omega_{ij}}{\sqrt{\omega_{ii}\omega_{jj}}}$$

  $\Rightarrow$ Non-zero entries in $\mathbf{\Omega} \Leftrightarrow$ Edges in the graph G

Concentration matrix

- Let $\mathbf{\Sigma}$ be the covariance matrix of $\mathbf{X} = [X_1, \ldots, X_{N_v}]^T$
  Def: the concentration matrix is $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$ with entries ω_{ij}
- Key result: For GGMs, the partial correlations can be expressed as

$$\rho_{ij|V\setminus\{i,j\}} = -\frac{\omega_{ij}}{\sqrt{\omega_{ii}\omega_{jj}}}$$

  $\Rightarrow$ Non-zero entries in $\mathbf{\Omega} \Leftrightarrow$ Edges in the graph G
- Inferring G from data in this context known as covariance selection
  $\Rightarrow$ Classical methods are 'network-agnostic,' and effectively test

$$H_0 : \rho_{ij|V\setminus\{i,j\}} = 0 \quad \text{versus} \quad H_1 : \rho_{ij|V\setminus\{i,j\}} \neq 0$$

  $\Rightarrow$ Often not scalable, and $n \ll N_v$ so estimation of $\hat{\mathbf{\Sigma}}$ challenging
- A. Dempster, "Covariance selection," *Biometrics,* vol. 28, pp. 157-175, 1974

Covariance selection meets linear regression

- Suppose the random vector $\mathbf{X} = [X_1, \ldots, X_{N_v}]^\top \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$
- Conditional mean of X_i given $\mathbf{X}_{(-i)} = [X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_{N_v}]^\top$ is

$$\mathbb{E}\left[X_i \,\middle|\, \mathbf{X}_{(-i)} = \mathbf{x}_{(-i)}\right] = \boldsymbol{\beta}_{(-i)}^\top \mathbf{x}_{(-i)}$$

Covariance selection meets linear regression

- Suppose the random vector $\mathbf{X} = [X_1, \ldots, X_{N_v}]^\top \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$
- Conditional mean of X_i given $\mathbf{X}_{(-i)} = [X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_{N_v}]^\top$ is

$$\mathbb{E}\left[X_i \,\middle|\, \mathbf{X}_{(-i)} = \mathbf{x}_{(-i)}\right] = \boldsymbol{\beta}_{(-i)}^\top \mathbf{x}_{(-i)}$$

- Entries of $\boldsymbol{\beta}_{(-i)}$ expressible in terms of those in $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$, namely

$$\beta_{(-i),j} = -\frac{\omega_{ij}}{\omega_{ii}}$$

$\Rightarrow$ Non-zero $\beta_{(-i),j} \Leftrightarrow$ Non-zero ω_{ij} in $\mathbf{\Omega} \Leftrightarrow$ Edge (i, j) in G

Covariance selection meets linear regression

- Suppose the random vector $\mathbf{X} = [X_1, \ldots, X_{N_v}]^\top \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$
- Conditional mean of X_i given $\mathbf{X}_{(-i)} = [X_1, \ldots, X_{i-1}, X_{i+1}, \ldots, X_{N_v}]^\top$ is

$$\mathbb{E}\left[X_i \,\middle|\, \mathbf{X}_{(-i)} = \mathbf{x}_{(-i)}\right] = \boldsymbol{\beta}_{(-i)}^\top \mathbf{x}_{(-i)}$$

- Entries of $\boldsymbol{\beta}_{(-i)}$ expressible in terms of those in $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$, namely

$$\beta_{(-i),j} = -\frac{\omega_{ij}}{\omega_{ii}}$$

$\Rightarrow$ Non-zero $\beta_{(-i),j} \Leftrightarrow$ Non-zero ω_{ij} in $\mathbf{\Omega} \Leftrightarrow$ Edge (i, j) in G

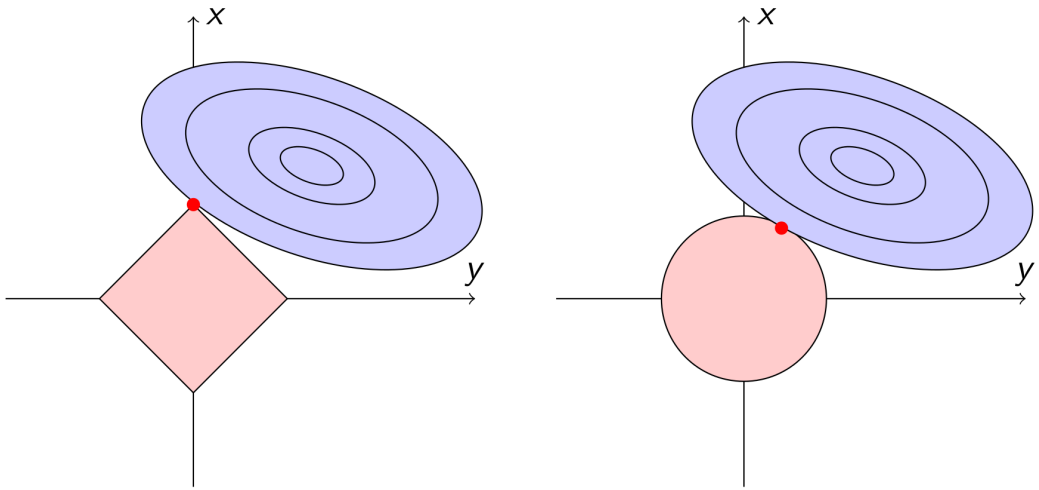
- Suggests inference of G via least-squares (LS) regression, to estimate

$$\boldsymbol{\beta}_{(-i)} = \arg\min_{\boldsymbol{\theta}} \mathbb{E}\left[(X_i - \boldsymbol{\theta}^\top \mathbf{X}_{(-i)})^2\right]$$

$\Rightarrow$ Looking for zeros in $\boldsymbol{\beta}_{(-i)}$, so should encourage sparse solutions

Sparsity and the ℓ_1 norm

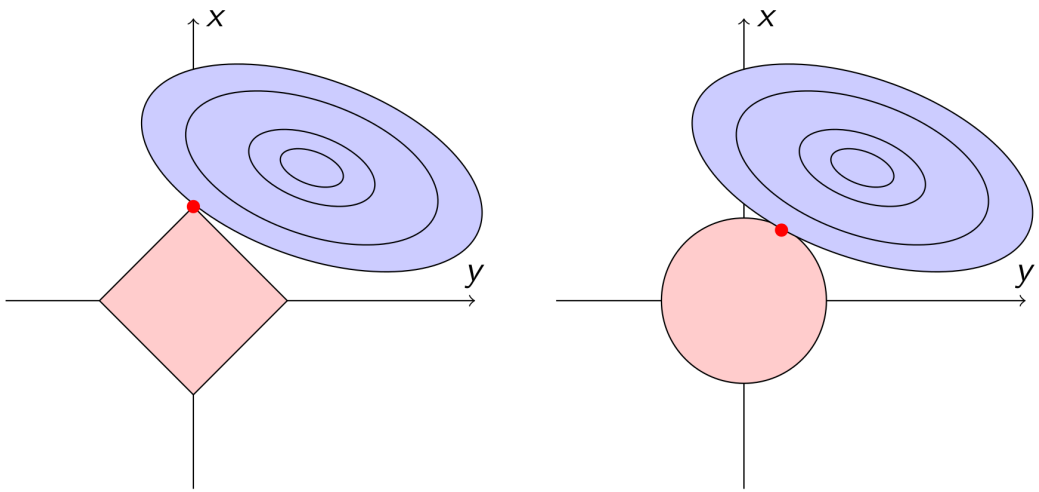
- Consider minimizing a quadratic function of $\boldsymbol{\theta}$ as in LS or ridge
- Q: What is the effect of an ℓ_1-norm constraint, i.e., $\|\boldsymbol{\theta}\|_1 = \sum_i |\theta_i| \leq \tau$?



$\Rightarrow$ Level sets touch constrain set in a kink $\rightarrow$ Sparse solution

Sparsity and the ℓ_1 norm

- Consider minimizing a quadratic function of $\boldsymbol{\theta}$ as in LS or ridge
- Q: What is the effect of an ℓ_1-norm constraint, i.e., $\|\boldsymbol{\theta}\|_1 = \sum_i |\theta_i| \leq \tau$?



$\Rightarrow$ Level sets touch constrain set in a kink $\rightarrow$ Sparse solution

- Lasso estimator enables estimation and variable selection [Tibshirani'94]

$$\hat{\boldsymbol{\theta}}_{Lasso} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^\top \boldsymbol{\theta})^2, \text{ s. to } \|\boldsymbol{\theta}\|_1 \leq \tau$$

Penalized linear regression

- Given data $\{x_{ik}\}_{k=1}^{n}$, ordinary LS not satisfactory for inference of G

$$\hat{\beta}_{(-i)}^{LS} = \arg\min_{\boldsymbol{\theta}} \sum_{k=1}^{n} (x_{ik} - \boldsymbol{\theta}^{\top} \mathbf{x}_{(-i),k})^2$$

  - If $n \ll N_v - 1$, the LS estimation problem is underdetermined
  - For finite n, LS yields non-zero estimates a.s. $\Rightarrow$ Full graph G

Penalized linear regression

- Given data $\{x_{ik}\}_{k=1}^{n}$, ordinary LS not satisfactory for inference of G

$$\hat{\boldsymbol{\beta}}_{(-i)}^{LS} = \arg\min_{\boldsymbol{\theta}} \sum_{k=1}^{n} (x_{ik} - \boldsymbol{\theta}^{\top} \mathbf{x}_{(-i),k})^2$$

  - If $n \ll N_v - 1$, the LS estimation problem is underdetermined
  - For finite n, LS yields non-zero estimates a.s. $\Rightarrow$ Full graph G

- Overcome these limitations using ℓ_1-norm penalized LS regression

$$\hat{\boldsymbol{\beta}}_{(-i)}^{PLS} = \arg\min_{\boldsymbol{\theta}} \sum_{k=1}^{n} (x_{ik} - \boldsymbol{\theta}^{\top} \mathbf{x}_{(-i),k})^2 + \lambda \|\boldsymbol{\theta}\|_1$$

  - Convex problem, tuning λ controls the sparsity level in $\hat{\boldsymbol{\beta}}_{(-i)}^{PLS}$
  - Theoretical guarantees: consistency [Meinshausen-Bühlmann'06]
  - Fast algorithms: graphical Lasso [Friedman et al'07]

Summary of logical roadmap

- Inference of GGMs with edges $E = \left\{(i,j) \in V^{(2)} : \rho_{ij|V\setminus\{i,j\}} \neq 0\right\}$

Association network inference:

Find pairs $\{i,j\}$ for which $\rho_{ij|V\setminus\{i,j\}} \neq 0$

$$\rho_{ij|V\setminus\{i,j\}} = -\frac{\omega_{ij}}{\sqrt{\omega_{ii}\omega_{ij}}}$$

Covariance selection:

Find non-zero entries $\omega_{ij} \neq 0$ in the concentration matrix $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$

$$\beta_{(-i),j} = -\frac{\omega_{ij}}{\omega_{ii}}$$

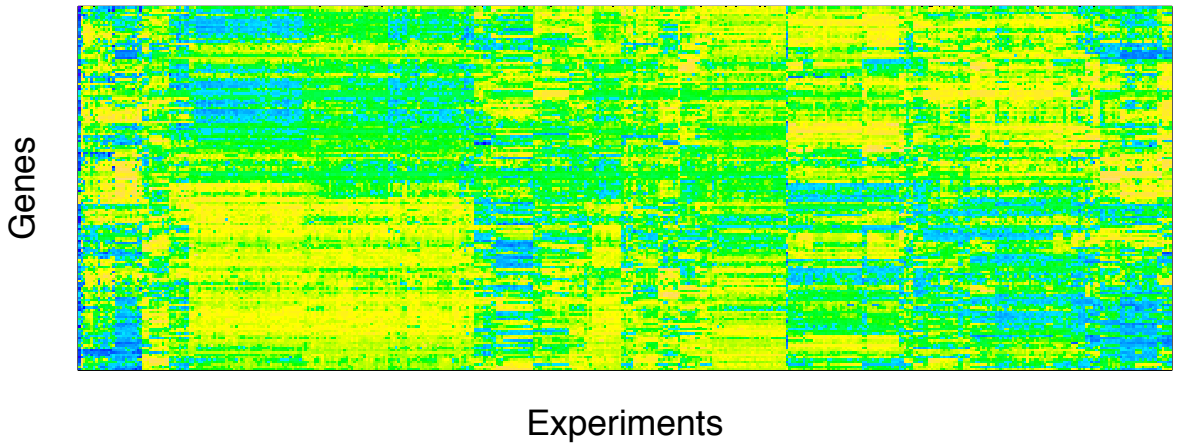
Variable selection in linear regression:

Find non-zero regression coefficients in

$$\boldsymbol{\beta}_{(-i)} = \arg\min_{\boldsymbol{\theta}} \mathbb{E}\left[(X_i - \boldsymbol{\theta}^\top \mathbf{X}_{(-i)})^2\right]$$

Case study: Interactions among E. coli genes

- Use microarray data and correlation methods to infer TF/target pairs



- Dataset: relative log expression RNA levels, for genes in E. coli
  - 4,345 genes measured under 445 different experimental conditions
- Ground truth: 153 TFs, and TF/target pairs from database RegulonDB

Methods to infer TF/target gene pairs

- Three correlation based methods to infer TF/target gene pairs

  $\Rightarrow$ Interactions declared if suitable p-values fall below a threshold

  Method 1: Pearson correlation between TF and potential target gene

  Method 2: Partial correlation, controlling for shared effects of one ($m = 1$) other TF, across all 152 other TFs

  Method 3: Full partial correlation, simultaneously controlling for shared effects of all ($m = 152$) other TFs

Methods to infer TF/target gene pairs

- Three correlation based methods to infer TF/target gene pairs

  $\Rightarrow$ Interactions declared if suitable p-values fall below a threshold

  Method 1: Pearson correlation between TF and potential target gene

  Method 2: Partial correlation, controlling for shared effects of one ($m = 1$) other TF, across all 152 other TFs

  Method 3: Full partial correlation, simultaneously controlling for shared effects of all ($m = 152$) other TFs

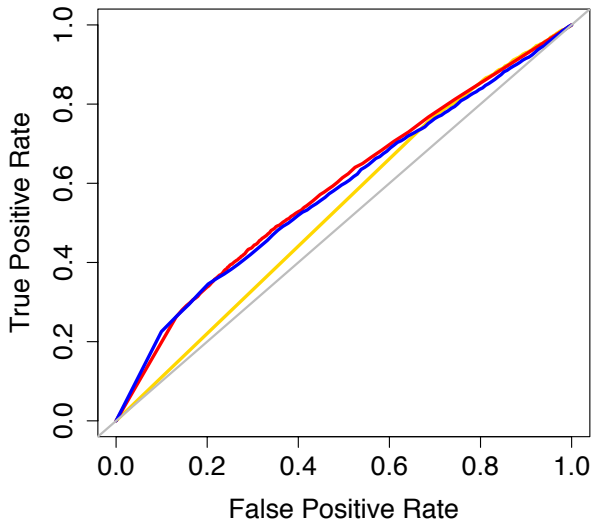
- In all cases applied Fisher transformation to obtain z-scores

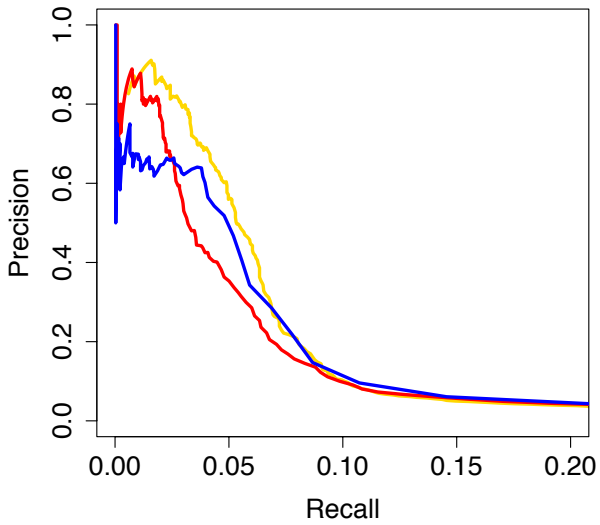
  $\Rightarrow$ Asymptotic Gaussian distributions for p-values, with $n = 445$

- Compared inferred graphs to ground-truth network from RegulonDB

Performance comparisons

- ROC and Precision/Recall curves for Methods 1, 2, and 3
  - ⇒ Precision: fraction of predicted links that are true
  - ⇒ Recall: fraction of true links that are correctly predicted

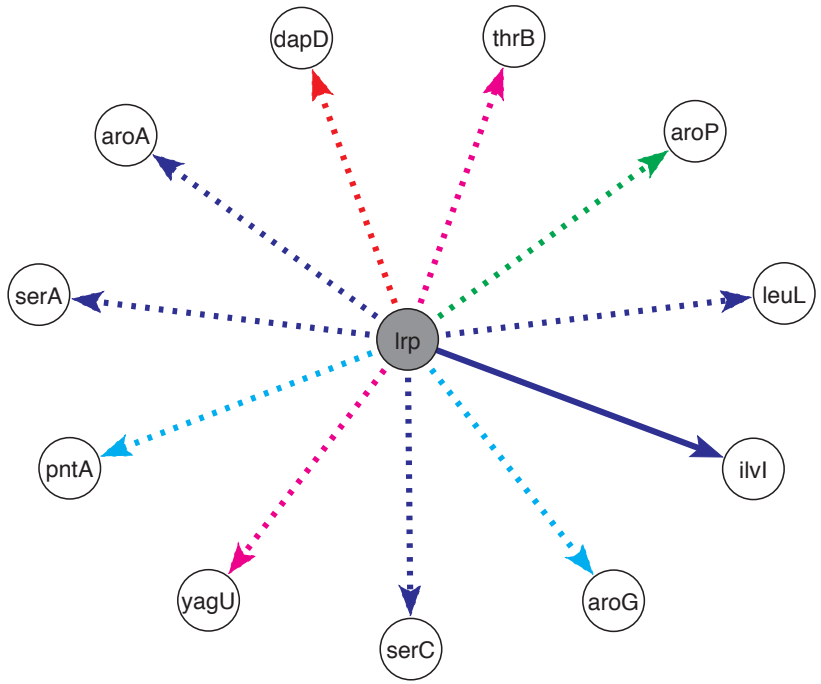




- Method 1 performs worst, but none is stellar
  - ⇒ Correlation not strong indicator of regulation in this data
- All methods share a region of high precision, but a very small recall
  - ⇒ Limitations in number/diversity of profiles [Faith et al'07]

Predicting new TF/target gene pairs

- In biology, often interest is in predicting new interactions



- 11 interactions found for TF *lrp*, 10 experimentally confirmed (dotted)
  - $\Rightarrow$ 5 interacting target genes were new (magenta, red, cyan)
  - $\Rightarrow$ 4 present in RegulonDB (magenta, cyan), but not as *lrp* targets

Importing slides from a tutorial

- Parts of tutorial talk by Xiaowen Dong (Oxford)
  - $\Rightarrow$ 2017 Graph Signal Processing workshop
  - $\Rightarrow$ Thank you Xiaowen!
- We also thank again Gonzalo Mateos (U. Rochester)!

Tomographic inference

Network sampling and challenges

Background on statistical sampling theory

Graph sampling designs

Estimation of network totals, group size, and degree distributions

Network topology inference problems

Link prediction

Inference of association networks

Tomographic network topology inference

Tomographic network topology inference

- In imaging, tomography refers to imaging by sections (e.g., MRI)
  - Reconstruction algorithms relate 'external data' to internal structure

Goal: create images of internal aspects of the human body

Tomographic network topology inference

- In imaging, tomography refers to imaging by sections (e.g., MRI)
  - Reconstruction algorithms relate 'external data' to internal structure

Goal: create images of internal aspects of the human body

Tomographic network topology inference

Predict edge and vertex status in the 'interior' of G, given only observations x_i for vertices $i \in V$ in the 'exterior' of G

- Most difficult case of topology inference. An ill-posed inverse problem
  $\Rightarrow$ Inverse problem: invert mapping from 'internal' to 'external'
  $\Rightarrow$ Ill-posed: the mapping is many-to-one

Tomographic network topology inference

- In imaging, tomography refers to imaging by sections (e.g., MRI)
  - Reconstruction algorithms relate 'external data' to internal structure

  Goal: create images of internal aspects of the human body

Tomographic network topology inference

Predict edge and vertex status in the 'interior' of G, given only observations x_i for vertices $i \in V$ in the 'exterior' of G

- Most difficult case of topology inference. An ill-posed inverse problem

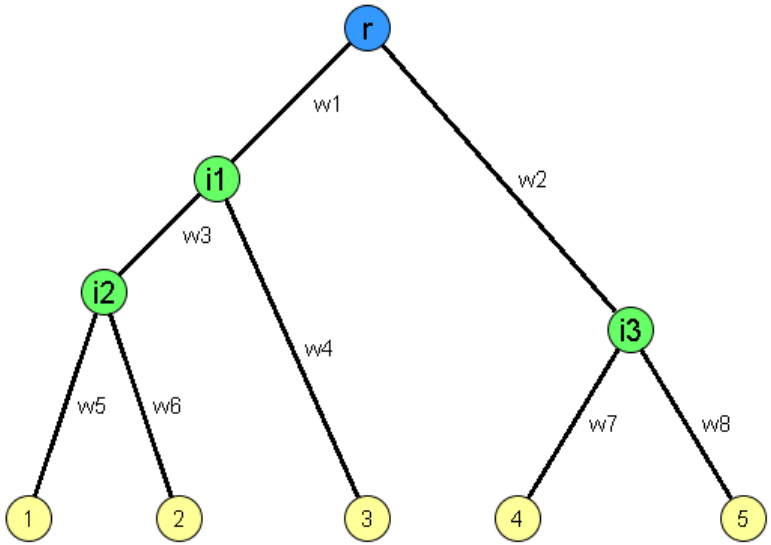
  $\Rightarrow$ Inverse problem: invert mapping from 'internal' to 'external'

  $\Rightarrow$ Ill-posed: the mapping is many-to-one

- Most work has dealt with inference of tree topologies

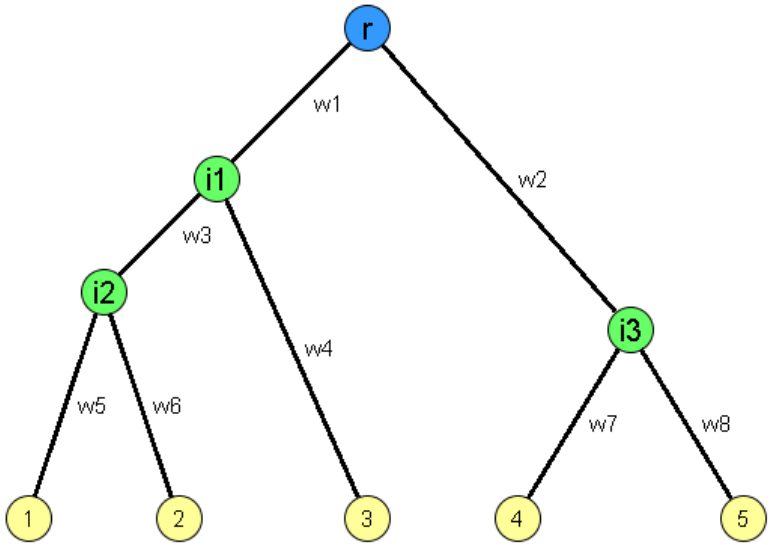
  Ex: computer network topologies, phylogenetic tree, media cascades

- **Def:** an undirected tree $T = (V_T, E_T)$ is a connected acyclic graph



- Nomenclature:
  - Rooted tree: tree with a single vertex $r \in V_T$ singled out
  - Leaves: subset of vertices $L \subset V_T$ of degree one
  - Internal vertices: those vertices in $V_T \setminus \{\{r\} \cup L\}$
  - Binary tree: root and internal vertices have at most two children

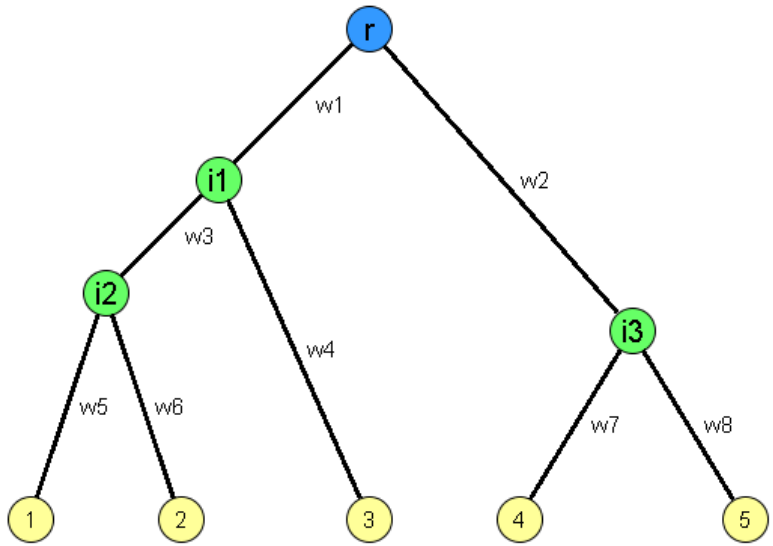
- Given n i.i.d. measurements of RVs $\{X_1, \ldots, X_{N_L}\}$ on N_L vertices



- Consider the family $\mathcal{T}_{N_L}$ of binary trees with N_L labeled leaves
  $\Rightarrow$ If we know r then all trees in $\mathcal{T}_{N_L}$ will be rooted at r

Tomographic inference of tree topologies

- Given n i.i.d. measurements of RVs $\{X_1, \ldots, X_{N_L}\}$ on N_L vertices



- Consider the family $\mathcal{T}_{N_L}$ of binary trees with N_L labeled leaves
  $\Rightarrow$ If we know r then all trees in $\mathcal{T}_{N_L}$ will be rooted at r

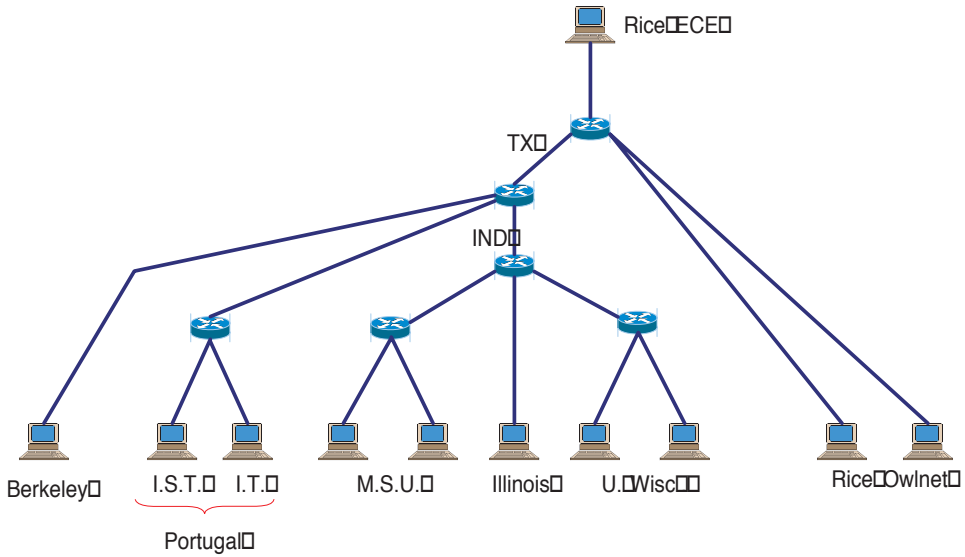
Tomographic tree topology inference

Find a tree $\hat{T} \in \mathcal{T}_{N_L}$ that 'best' explains the data $\{\mathbf{x}_1, \ldots, \mathbf{x}_{N_L}\}$

- Often of interest to infer a set of branch weights as well

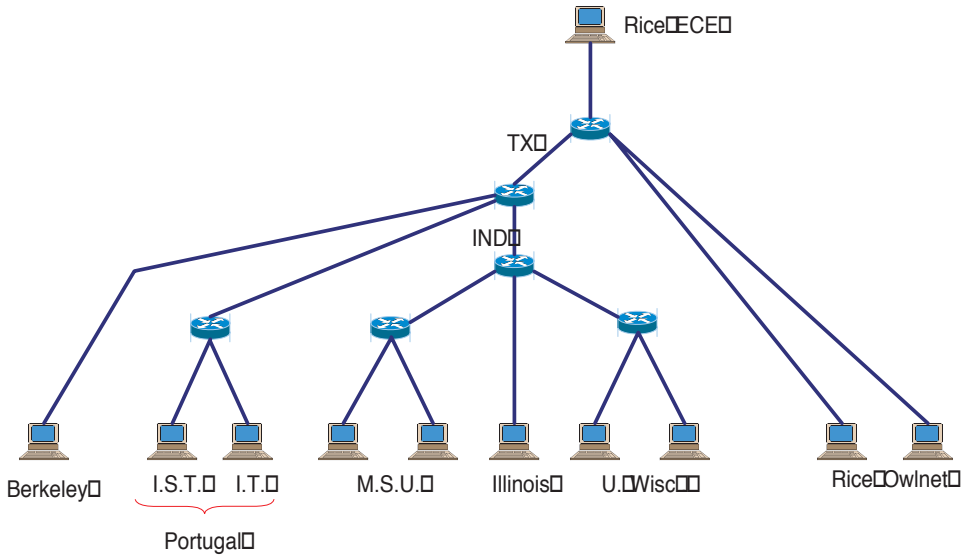
Multicast probes: measurements

- Ex: Consider inference of computer network topologies, e.g., Internet
- Multicast packets sent from a node (r) to multiple destinations (L)
  - $\Rightarrow$ Probes forwarded at routing devices, could be lost en route



Multicast probes: measurements

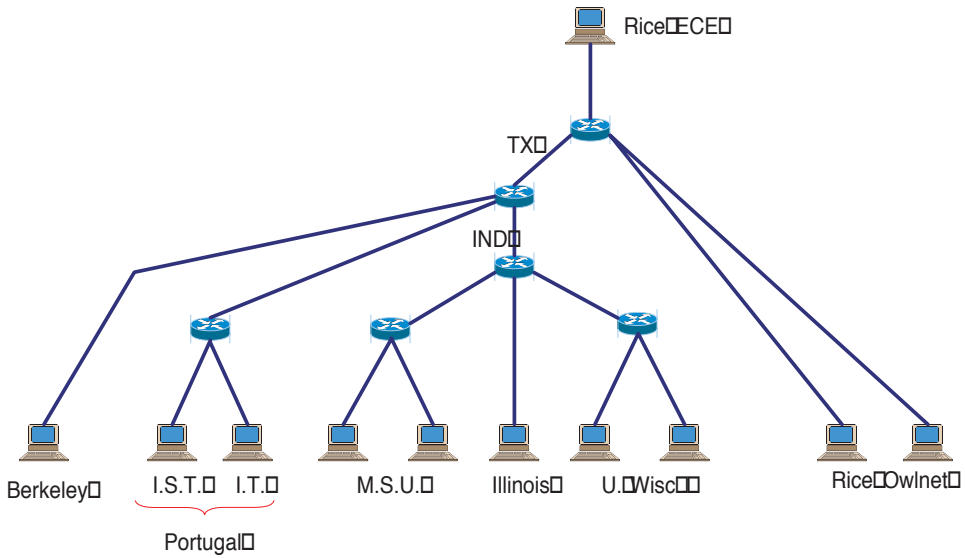
- Ex: Consider inference of computer network topologies, e.g., Internet
- Multicast packets sent from a node (r) to multiple destinations (L)
  - $\Rightarrow$ Probes forwarded at routing devices, could be lost en route



- For leaves $\ell \in L$, consider the indicator $X_\ell = \mathbb{I}\{\ell \text{ received the probe}\}$
  - $\Rightarrow$ Send n multicast probes to yield data $\{\mathbf{x}_\ell \in \{0,1\}^n\}_{\ell \in L}$

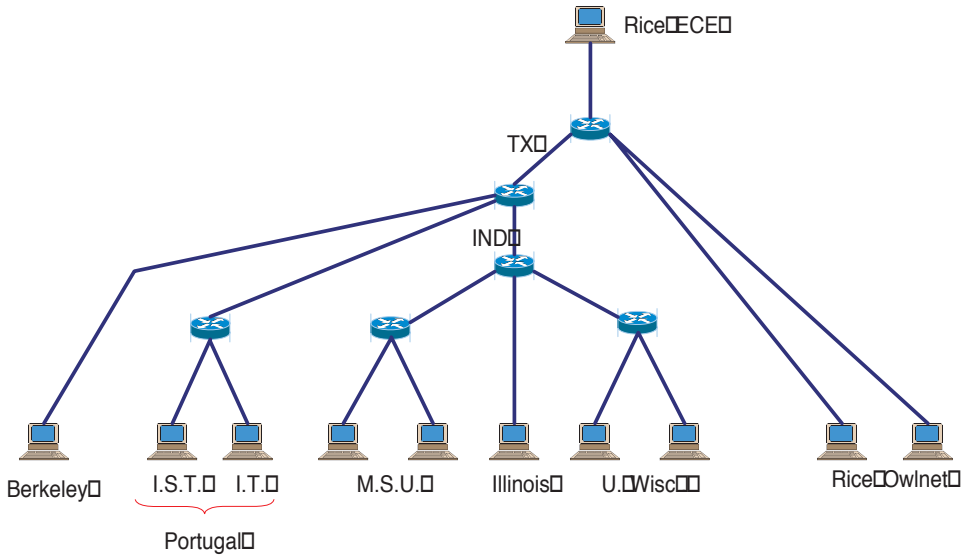
Multicast probes: structure

- Think of leaf RVs $\{X_1, \ldots, X_{N_L}\}$ as samples of a process $\{X_j\}_{j \in V_T}$
- Useful notation to describe process' structure
  - **Def:** closest common ancestor $a(U)$ to a set of leaves $U \subseteq L$
  - **Def:** set $d(j)$ of all immediate descendants of internal vertex j



Multicast probes: structure

- Think of leaf RVs $\{X_1, \ldots, X_{N_L}\}$ as samples of a process $\{X_j\}_{j \in V_T}$
- Useful notation to describe process' structure
  - **Def:** closest common ancestor $a(U)$ to a set of leaves $U \subseteq L$
  - **Def:** set $d(j)$ of all immediate descendants of internal vertex j



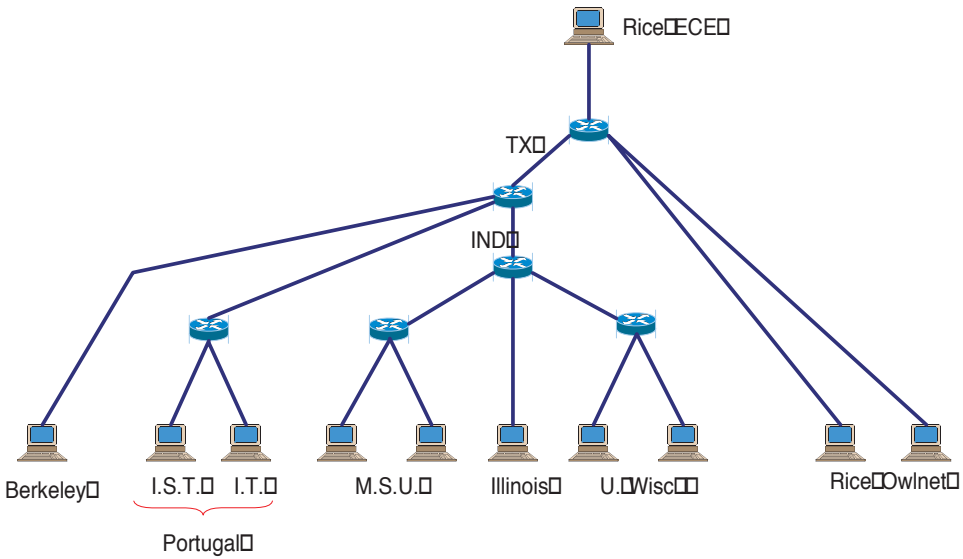
- Multicast tree enforces hereditary constraints

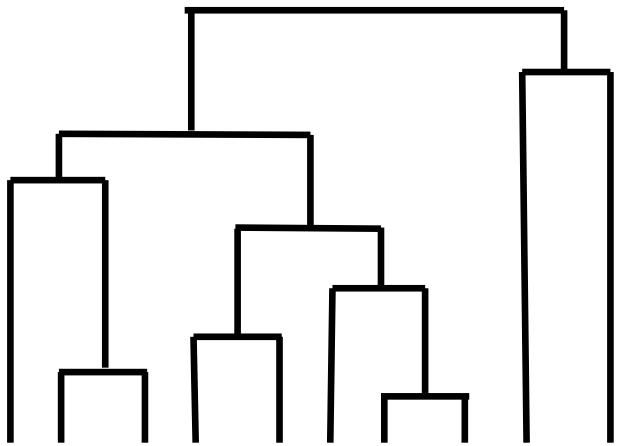
  $\Rightarrow X_{a(U)} = 0$ implies $X_j = 0$ for all $j \in U$

  $\Rightarrow$ If $X_j = 1$ for at least one $j \in d(k)$, then $X_k = 1$

Hierarchical clustering-based methods

- Hierarchical clustering groups N_L objects based on (dis)similarity
  - $\Rightarrow$ Entire hierarchy of nested partitions obtained $\rightarrow$ dendrogram





- Natural tool for tomographic inference of tree topologies
  - $\Rightarrow$ N_L leaves as 'objects', dendrogram as the inferred tree $\hat{T}$
- Tailor a (dis)similarity to the tomographic inference problem at hand

Multicast probes: dissimilarity

- Shared packet loss rate indicative of close leaves in a multicast tree
- Two types of shared loss between a pair of leaves $j, k \in L$
  - **True:** loss of packets in the path common to vertices j and k
  - **False:** losses on paths after the closest common ancestor $a(\{j, k\})$
- Net shared loss rate includes both effects $\Rightarrow$ misleading similarity

  $\Rightarrow$ Can obtain true shared loss rates via simple packet-loss model
- N. G. Duffield et al, "Multicast topology inference from measured end-to-end loss," *IEEE Trans. Info. Theory,* vol. 48, pp. 26-45, 2002

Multicast probes: packet-loss model

- Recall the cascade process $\{X_j\}_{j \in V_T}$ induced by multicast probing
- Specify a Markov model down the tree
  - Root r: set $X_r = 1$
  - Internal vertex k: if $X_k = 0$, then $X_j = 0$ for all $j \in d(k)$. Otherwise,

$$\mathrm{P}\left[X_j = 1 \,\middle|\, X_k = 1\right] = 1 - \mathrm{P}\left[X_j = 0 \,\middle|\, X_k = 1\right] = \alpha_j, \;\; j \in d(k)$$

$\Rightarrow$ Probes successfully transmitted through link (k, j) w.p. α_j

Multicast probes: packet-loss model

- Recall the cascade process $\{X_j\}_{j \in V_T}$ induced by multicast probing
- Specify a Markov model down the tree
  - Root r: set $X_r = 1$
  - Internal vertex k: if $X_k = 0$, then $X_j = 0$ for all $j \in d(k)$. Otherwise,

$$\mathrm{P}\left[X_j = 1 \,\middle|\, X_k = 1\right] = 1 - \mathrm{P}\left[X_j = 0 \,\middle|\, X_k = 1\right] = \alpha_j, \;\; j \in d(k)$$

  $\Rightarrow$ Probes successfully transmitted through link (k, j) w.p. α_j
- Probe successfully transmited from r to k w.p.

$$\mathrm{P}\left[X_k = 1 \,\middle|\, X_r = 1\right] := A(k) = \prod_{j \succ k} \alpha_j$$

  $\Rightarrow$ $j \succ k$ denotes ancestral vertices of k in path from r

Multicast probes: packet-loss model

- Recall the cascade process $\{X_j\}_{j \in V_T}$ induced by multicast probing
- Specify a Markov model down the tree
  - Root r: set $X_r = 1$
  - Internal vertex k: if $X_k = 0$, then $X_j = 0$ for all $j \in d(k)$. Otherwise,

$$\mathrm{P}\left[X_j = 1 \,\middle|\, X_k = 1\right] = 1 - \mathrm{P}\left[X_j = 0 \,\middle|\, X_k = 1\right] = \alpha_j, \;\; j \in d(k)$$

  $\Rightarrow$ Probes successfully transmitted through link (k, j) w.p. α_j
- Probe successfully transmited from r to k w.p.

$$\mathrm{P}\left[X_k = 1 \,\middle|\, X_r = 1\right] := A(k) = \prod_{j \succ k} \alpha_j$$

  $\Rightarrow$ $j \succ k$ denotes ancestral vertices of k in path from r
- True shared loss rate for two leaf vertices $j, k \in L$ is $1 - A(a(\{j, k\}))$

Estimating shared loss rates

- Let $L(k)$ be the set of leaves that are descendants of k
  - Probability that at least one descendant leaf of k received a packet

$$\gamma(k) = \mathsf{P}\left[\bigcup_{j \in L(k)} \{X_j = 1\}\right]$$

Estimating shared loss rates

- Let $L(k)$ be the set of leaves that are descendants of k
  - Probability that at least one descendant leaf of k received a packet

$$\gamma(k) = \mathrm{P}\left[\bigcup_{j \in L(k)} \{X_j = 1\}\right]$$

- Key: Using probabilistic arguments, can establish the relation

$$1 - \frac{\gamma(k)}{A(k)} = \prod_{j \in d(k)} \left[1 - \frac{\gamma(j)}{A(k)}\right]$$

$\Rightarrow$ Given values $\{\gamma(k)\}_{k \in V_T}$, can solve for the $\{A(k)\}_{k \in V_T}$

Estimating shared loss rates

- Let $L(k)$ be the set of leaves that are descendants of k
  - Probability that at least one descendant leaf of k received a packet

$$\gamma(k) = \mathrm{P}\left[\bigcup_{j \in L(k)} \{X_j = 1\}\right]$$

- Key: Using probabilistic arguments, can establish the relation

$$1 - \frac{\gamma(k)}{A(k)} = \prod_{j \in d(k)} \left[1 - \frac{\gamma(j)}{A(k)}\right]$$

  $\Rightarrow$ Given values $\{\gamma(k)\}_{k \in V_T}$, can solve for the $\{A(k)\}_{k \in V_T}$

- But $\{\gamma(k)\}_{k \in V_T}$ unknown! Use leaf measurements to form estimates

$$\hat{\gamma}(k) = \frac{1}{n} \sum_{i=1}^{n} \max_{j \in L(k)} (x_{ji})$$

Agglomerative hierarchical clustering algorithm

- Greedy, agglomerative algorithm based on shared loss similarities

S1: Estimate packet losses $\hat{\gamma}(j)$ at the leaves $j \in L$

S2: Estimate shared loss $1 - \hat{A}(a(\{j,k\}))$ for all pairs $j,k \in L$

$$\text{Estimate: } \hat{\gamma}(a(\{j,k\})) = \frac{1}{n}\sum_{i=1}^{n} \max_{s\in\{j,k\}}(x_{si}), \;\; j,k \in L$$

$$\text{Solve: } 1 - \frac{\hat{\gamma}(a(\{j,k\}))}{\hat{A}(a(\{j,k\}))} = \prod_{i\in\{j,k\}}\left[1 - \frac{\hat{\gamma}(i)}{\hat{A}(a(\{j,k\}))}\right]$$

S3: Merge pair $\{j^*,k^*\} = \arg\max_{j,k}[1 - \hat{A}(a(\{j,k\}))]$

S4: Exchange $\{j^*,k^*\}$ for $a(\{j^*,k^*\})$ in L and go back to S2

Agglomerative hierarchical clustering algorithm

- Greedy, agglomerative algorithm based on shared loss similarities

  S1: Estimate packet losses $\hat{\gamma}(j)$ at the leaves $j \in L$

  S2: Estimate shared loss $1 - \hat{A}(a(\{j,k\}))$ for all pairs $j,k \in L$

$$\text{Estimate: } \hat{\gamma}(a(\{j,k\})) = \frac{1}{n}\sum_{i=1}^{n} \max_{s\in\{j,k\}}(x_{si}), \;\; j,k \in L$$

$$\text{Solve: } 1 - \frac{\hat{\gamma}(a(\{j,k\}))}{\hat{A}(a(\{j,k\}))} = \prod_{i\in\{j,k\}}\left[1 - \frac{\hat{\gamma}(i)}{\hat{A}(a(\{j,k\}))}\right]$$

  S3: Merge pair $\{j^*,k^*\} = \arg\max_{j,k}[1 - \hat{A}(a(\{j,k\}))]$

  S4: Exchange $\{j^*,k^*\}$ for $a(\{j^*,k^*\})$ in L and go back to S2

- Can establish theoretical consistency guarantees for recovering T

Likelihood-based methods

- Probability models of leaf RVs $\{X_\ell\}_{\ell \in L}$ used for defining (dis)similarities
  - $\Rightarrow$ But having such models $f(\mathbf{x} \,|\, T)$ also enables ML inference

Likelihood-based methods

- Probability models of leaf RVs $\{X_\ell\}_{\ell \in L}$ used for defining (dis)similarities
  - $\Rightarrow$ But having such models $f(\mathbf{x} \,|\, T)$ also enables ML inference
- If the n observations $\{\mathbf{x}_i\}_{i=1}^n$ are independent, the likelihood is

$$\mathcal{L}_n(T) = \prod_{i=1}^{n} f(\mathbf{x}_i \,|\, T)$$

Likelihood-based methods

- Probability models of leaf RVs $\{X_\ell\}_{\ell \in L}$ used for defining (dis)similarities
  - $\Rightarrow$ But having such models $f(\mathbf{x} \,|\, T)$ also enables ML inference
- If the n observations $\{\mathbf{x}_i\}_{i=1}^n$ are independent, the likelihood is

$$\mathcal{L}_n(T) = \prod_{i=1}^{n} f(\mathbf{x}_i \,|\, T)$$

- Models often include other parameters $\boldsymbol{\theta}$ (e.g., the α_j) beyond T
  - $\Rightarrow$ In this case $\mathcal{L}_n(T)$ is an integrated likelihood, namely

$$\mathcal{L}_n(T) = \prod_{i=1}^{n} \int_{\theta \in \Theta} f(\mathbf{x}_i \,|\, T, \boldsymbol{\theta}) f(\boldsymbol{\theta} \,|\, T) d\boldsymbol{\theta}$$

Likelihood-based methods

- Probability models of leaf RVs $\{X_\ell\}_{\ell \in L}$ used for defining (dis)similarities
  $\Rightarrow$ But having such models $f(\mathbf{x} \,|\, T)$ also enables ML inference
- If the n observations $\{\mathbf{x}_i\}_{i=1}^n$ are independent, the likelihood is

$$\mathcal{L}_n(T) = \prod_{i=1}^{n} f(\mathbf{x}_i \,|\, T)$$

- Models often include other parameters $\boldsymbol{\theta}$ (e.g., the α_j) beyond T
  $\Rightarrow$ In this case $\mathcal{L}_n(T)$ is an integrated likelihood, namely

$$\mathcal{L}_n(T) = \prod_{i=1}^{n} \int_{\theta \in \Theta} f(\mathbf{x}_i \,|\, T, \boldsymbol{\theta}) f(\boldsymbol{\theta} \,|\, T) d\boldsymbol{\theta}$$

- Integrals may be computationally challenging. The ML estimate is

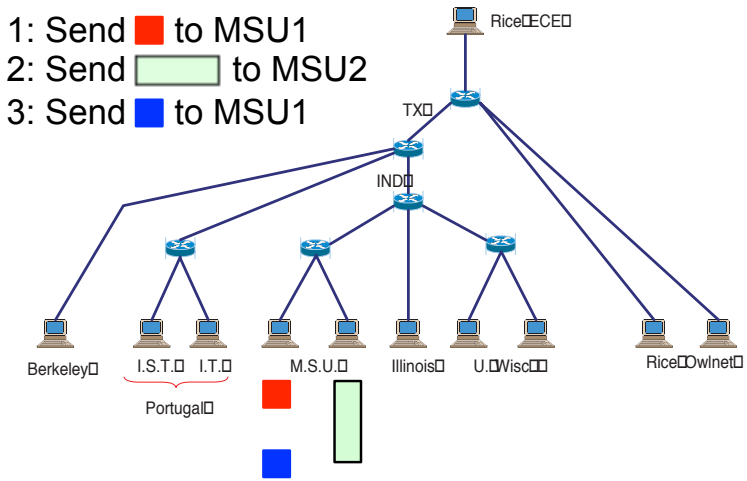
$$\hat{T}_{ML} = \arg \max_{T \in \mathcal{T}_{N_L}} \mathcal{L}_n(T)$$

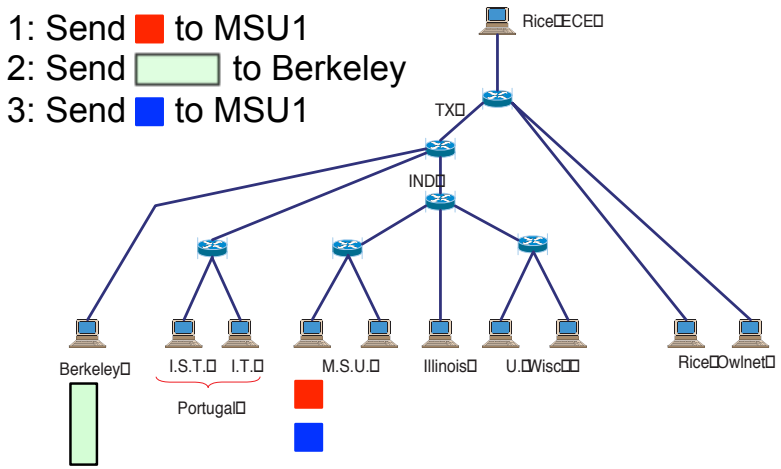
Case study: Sandwich probing

- Consider network tree topology inference via end-to-end probing
  - Packet drops rare (i.e., drop rate $< 2\%$) $\Rightarrow$ Shared loss rates ineffective

Case study: Sandwich probing

- Consider network tree topology inference via end-to-end probing
  - Packet drops rare (i.e., drop rate $< 2\%$) $\Rightarrow$ Shared loss rates ineffective
- Alternative measuring time-delay differences: sandwich probes
  - Send small probe to i, then large probe to j, other small probe to i last
  - Measure time-delay difference (TDD) between small packets



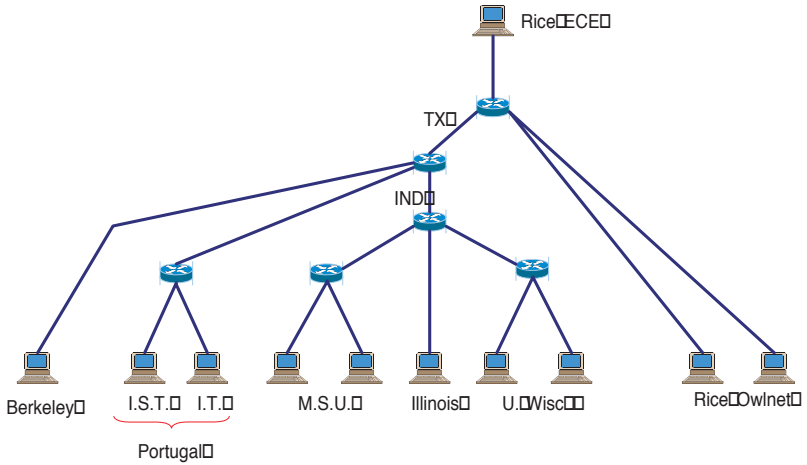


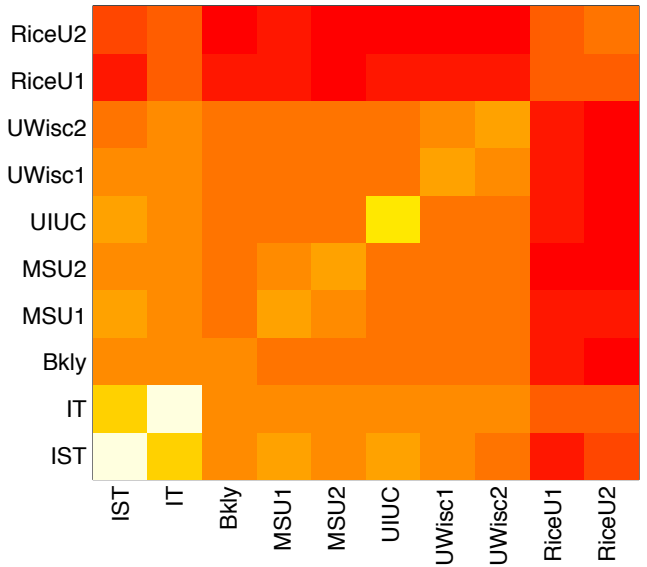
- If paths overlap, large probe induces high delay in the second small one

  $\Rightarrow$ Large TDD values indicative of close leaves in the tree topology

Modeling delay differences

- Sent sandwich probes every 50 *ms* to random pairs $j, k \in L$
  - $\Rightarrow$ Total of $9,567$ measured delay differences over 8 minutes





- For each pair $j, k \in L$, let x_{jk} be the average TDD
  - $\Rightarrow$ The Central Limit Theorem suggests $x_{jk} \sim \mathcal{N}(\mu_{jk}, \sigma^2_{jk})$
  - $\Rightarrow$ Independence of the x_{jk} reasonable by experimental setup

Agglomerative likelihood tree (ALT) algorithm

- Hierarchical clustering with likelihood-based similarity measure

Agglomerative likelihood tree (ALT) algorithm

- Hierarchical clustering with likelihood-based similarity measure
- Let $\ell_{ij}(\mu) = \log f(x_{ij}|\mu)$ be the Gaussian log-likelihood (σ_{ij}^2 known)

Agglomerative likelihood tree (ALT) algorithm

- Hierarchical clustering with likelihood-based similarity measure
- Let $\ell_{ij}(\mu) = \log f(x_{ij}|\mu)$ be the Gaussian log-likelihood (σ_{ij}^2 known)
- Initialize a set of vertices S with the leaves, i.e., $S = L$

  Def: similarity among leaves is estimated mean TDD

$$\hat{\mu}_{ij} = \hat{\mu}_{ji} = \arg\max_{\mu} \left[\ell_{ij}(\mu) + \ell_{ji}(\mu)\right], \quad i, j \in L$$

Agglomerative likelihood tree (ALT) algorithm

- Hierarchical clustering with likelihood-based similarity measure
- Let $\ell_{ij}(\mu) = \log f(x_{ij}|\mu)$ be the Gaussian log-likelihood (σ_{ij}^2 known)
- Initialize a set of vertices S with the leaves, i.e., $S = L$

  Def: similarity among leaves is estimated mean TDD

$$\hat{\mu}_{ij} = \hat{\mu}_{ji} = \arg\max_{\mu} \left[\ell_{ij}(\mu) + \ell_{ji}(\mu)\right], \quad i, j \in L$$

- Merge $\{i^*, j^*\} = \arg\max_{i,j} \hat{\mu}_{ij}$. Exchange $\{i^*, j^*\}$ for $a(\{i^*, j^*\})$ in S

Agglomerative likelihood tree (ALT) algorithm

- Hierarchical clustering with likelihood-based similarity measure
- Let $\ell_{ij}(\mu) = \log f(x_{ij}|\mu)$ be the Gaussian log-likelihood (σ_{ij}^2 known)
- Initialize a set of vertices S with the leaves, i.e., $S = L$

  Def: similarity among leaves is estimated mean TDD

$$\hat{\mu}_{ij} = \hat{\mu}_{ji} = \arg\max_{\mu} \left[\ell_{ij}(\mu) + \ell_{ji}(\mu)\right], \quad i, j \in L$$

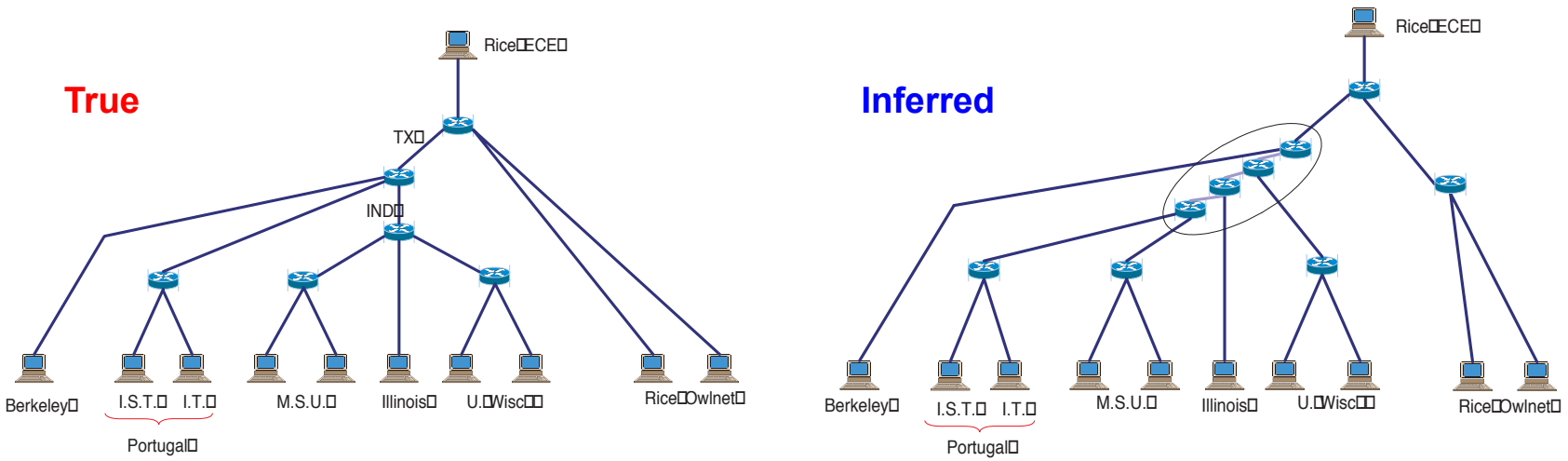
- Merge $\{i^*, j^*\} = \arg\max_{i,j} \hat{\mu}_{ij}$. Exchange $\{i^*, j^*\}$ for $a(\{i^*, j^*\})$ in S
- Algorithm then iterates until $|S| = 1$, by merging after calculating

$$\hat{\mu}_{kl} = \hat{\mu}_{lk} = \arg\max_{\mu} \sum_{m \in L(k)} \sum_{p \in L(l)} \left[\ell_{mp}(\mu) + \ell_{pm}(\mu)\right], \quad k, l \in S$$

$\Rightarrow$ Recall $L(k)$ is the set of leaves descended by k

Inferred topology

- Ground-truth topology obtained via `traceroute` probing
  - $\Rightarrow$ `traceroute` replies often 'turned-off' for security
  - $\Rightarrow$ Tomographic topology inference approaches relevant!



- ALT-inferred topology binary by construction $\Rightarrow$ introduces artifacts
- R. Castro et al, "Likelihood-based hierarchical clustering," *IEEE Trans. Signal Process.*, vol. 52, pp. 2308-2321, 2004