Network Topology Inference

Gonzalo Mateos
Dept. of ECE and Goergen Institute for Data Science
University of Rochester
gmateosb@ece.rochester.edu
`http://www.hajim.rochester.edu/ece/sites/gmateos/`

March 28, 2023

Network topology inference

Network topology inference problems

Link prediction

Case study: Predicting lawyer collaboration

Inference of association networks

Case study: Inferring genetic regulatory interactions

Tomographic network topology inference

Case study: Computer network topology identification

Network topology inference

- So far dealt with modeling and inference of observed network graphs

  $\Rightarrow$ Q: If a portion of G is unobserved, can we infer it from data?

- Discussed construction of representations $G(V, E)$ for network mapping

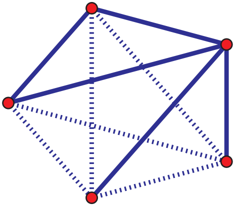
  $\Rightarrow$ Largely informal methodology, lacking an element of validation

- Formulate instead 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

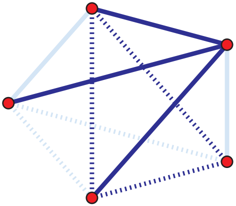
  Goal: infer the topology of the network graph $G(V, E)$

- Three canonical network topology inference problems
  - (i) Link prediction
  - (ii) Association network inference
  - (iii) Tomographic network topology inference

Link prediction



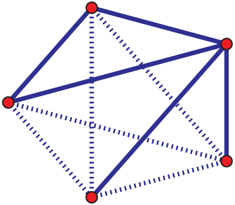
Original graph



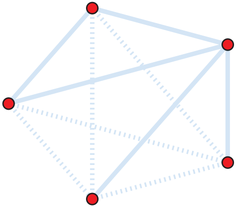
Link prediction

- 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



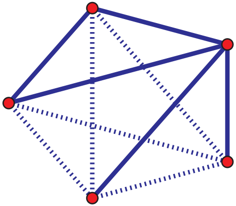
Original graph



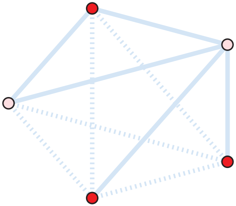
Association network inference

- 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 topology inference problems

Link prediction

Case study: Predicting lawyer collaboration

Inference of association networks

Case study: Inferring genetic regulatory interactions

Tomographic network topology inference

Case study: Computer network topology identification

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

- 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

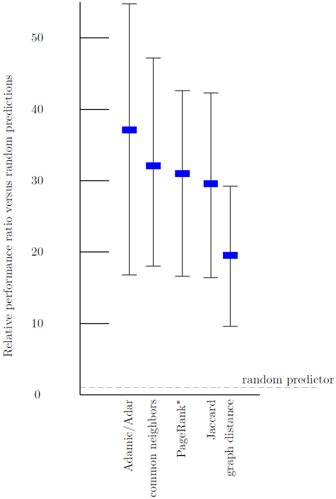
$$s(i,j) = |\mathcal{N}_i^{obs} \cap \mathcal{N}_j^{obs}| \text{ or } s(i,j) = \frac{|\mathcal{N}_i^{obs} \cap \mathcal{N}_j^{obs}|}{|\mathcal{N}_i^{obs} \cup \mathcal{N}_j^{obs}|}$$

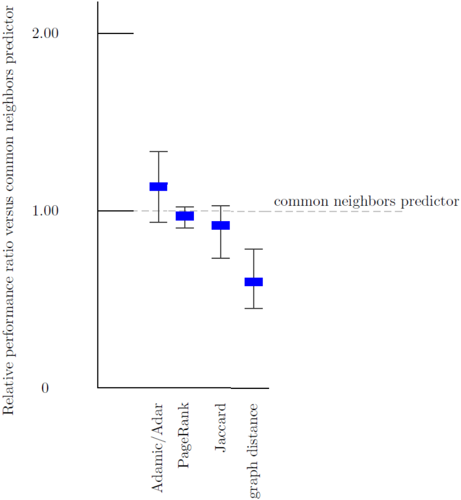
  $\Rightarrow$ Favor loosely-connected common neighbors [Adamic-Adar'03]

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

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 $\mathbf{Y}^{miss}$

- 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

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

- Logistic regression assumes $\mathbf{Y}_{ij}$ conditionally independent given $\mathbf{z}$

  $\Rightarrow$ Seldom the case with relational network data

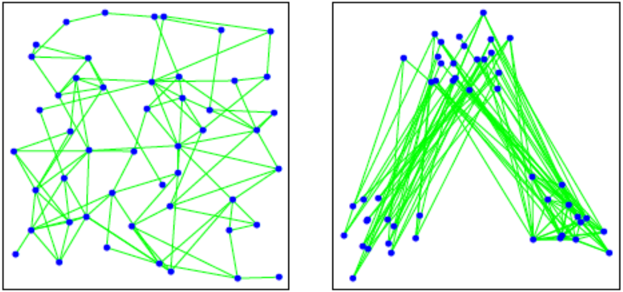
- Underlying mechanism of data missingness is important

  $\Rightarrow$ 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 lineal 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 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 $\boldsymbol{\beta}^\top \mathbf{z} - \theta_{u_i, u_j}$
  - Stochastic equivalence: Nodes in same class equally likely to link

P. D. Hoff, "Modeling homophily and stochastic equivalence in symmetric relational data," *NeurIPS*, 2008

Logistic regression with latent variables

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

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

  (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 $M_{ij} = \mathbf{u}_i^\top \mathbf{\Lambda} \mathbf{u}_j + \epsilon_{ij}$

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

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

- 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

Network topology inference problems

Link prediction

Case study: Predicting lawyer collaboration

Inference of association networks

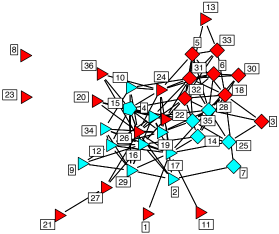
Case study: Inferring genetic regulatory interactions

Tomographic network topology inference

Case study: Computer network topology identification

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:

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

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

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

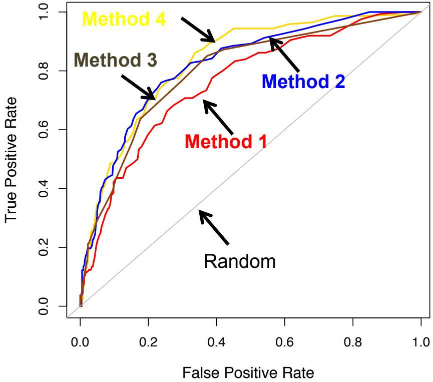
Method 3 informal scoring method with $s(i,j) = Z_{ij}^{(6)}$

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 ⇒ Agnostic to network structure
- Informal Method 3 yields slightly worst performance than 2 and 4

Inference of association networks

Network topology inference problems

Link prediction

Case study: Predicting lawyer collaboration

Inference of association networks

Case study: Inferring genetic regulatory interactions

Tomographic network topology inference

Case study: Computer network topology identification

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 $\mathtt{sim}(i,j) = f(\mathbf{x}_i, \mathbf{x}_j)$ 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 `sim` between vertex pairs

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

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

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

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

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

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
  - ⇒ 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., $E = \emptyset$
  - ⇒ We expect to declare $\binom{N_v}{2}\alpha$ spurious edges just by chance!
  - ⇒ For a large graph, this number can be considerable
- Ex: For G of order $N_v = 100$ and individual tests at level $\alpha = 0.05$
  - ⇒ 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

$$\text{FDR} = \mathbb{E}\left[\frac{R_{false}}{R} \,\Big|\, R > 0\right] \mathrm{P}\left(R > 0\right) \leq \gamma$$

  - 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

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

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

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

- 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

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

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

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

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

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

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

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

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

$$H_0 : \rho_{ij|S_m} = 0 \text{ for some } S_m \in V_{\setminus\{i,j\}}^{(m)}$$
$$H_1 : \rho_{ij|S_m} \neq 0 \text{ for all } S_m \in V_{\setminus\{i,j\}}^{(m)}$$

- 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

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

- 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]

Case study

Network topology inference problems

Link prediction

Case study: Predicting lawyer collaboration

Inference of association networks

Case study: Inferring genetic regulatory interactions

Tomographic network topology inference

Case study: Computer network topology identification

Case study: Gene-regulatory interactions

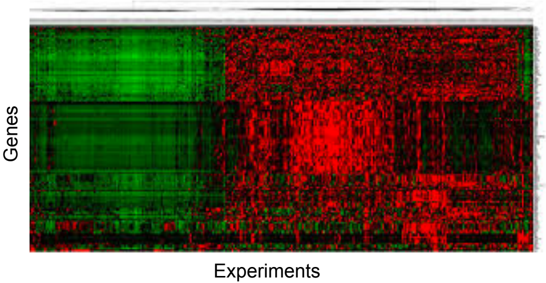
- Genes are segments of DNA encoding information about cell functions
- Such information used in the expression of genes
  - $\Rightarrow$ Creation of biochemical products, i.e., RNA or proteins
- Regulation of a gene refers to the control of its expression

  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

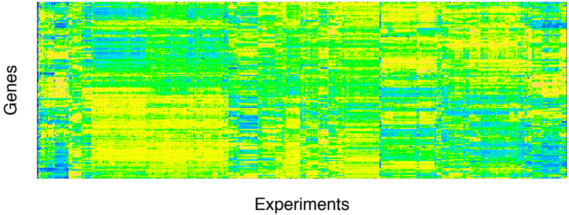
  $\Rightarrow$ Genome-wide scale data obtained using microarray technologies



- 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

Regulatory interactions in E. coli

- 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

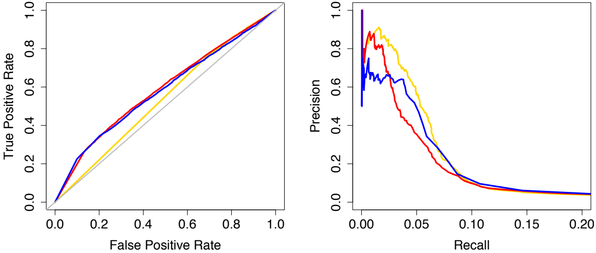
- In all cases applied Fisher transformation to obtain z-scores

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

- Compared inferred graphs to ground-truth network from RegulonDB

Performance comparisons

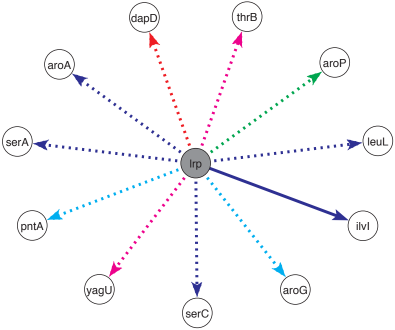
- ROC and Precision/Recall curves for Methods 1, 2, and 3
  - ⇒ Precision: fraction of predicted links that are true
  - ⇒ Recall: fraction of true links that are correctly predicted



- 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)
  - ⇒ 5 interacting target genes were new (magenta, red, cyan)
  - ⇒ 4 present in RegulonDB (magenta, cyan), but not as *lrp* targets

Gaussian graphical model networks

- 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

- Also known as Gaussian Markov random field (GMRF)

Covariance selection

- 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, 1974

Graphical Lasso

- Sparsity-regularized maximum-likelihood estimator of $\boldsymbol{\Omega}$ [Yuan-Lin'07]

$$\hat{\boldsymbol{\Omega}} \in \arg\max_{\boldsymbol{\Omega} \succeq \mathbf{0}} \left\{ \log\det \boldsymbol{\Omega} - \text{trace}(\hat{\boldsymbol{\Sigma}}\boldsymbol{\Omega}) - \lambda \|\boldsymbol{\Omega}\|_1 \right\}$$

$\Rightarrow$ Effective when $n \ll N_v$, encourages interpretable models

$\Rightarrow$ Scalable solvers using coordinate-descent [Friedman et al'08]

- Performance guarantee: Graphical lasso with $\lambda = 2\sqrt{\frac{\log N_v}{n}}$ satisfies

$$\|\hat{\boldsymbol{\Omega}} - \boldsymbol{\Omega}_0\|_2 \leq \sqrt{\frac{d_{\max}^2 \log N_v}{n}} \quad \text{w.h.p.}$$

$\Rightarrow$ Ground-truth $\boldsymbol{\Omega}_0$, maximum nodal degree $d_{\max}$

- Support consistency for $n = O(d_{\max}^2 \log N_v)$ [Ravikumar et al'11]

Covariance selection meets linear regression

- **Idea:** separately estimate neighborhoods $\mathcal{N}_i := \{j : (i,j) \in E\}$, $i \in \mathcal{V}$
- 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 $\boldsymbol{\Omega} = \boldsymbol{\Sigma}^{-1}$, namely

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

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

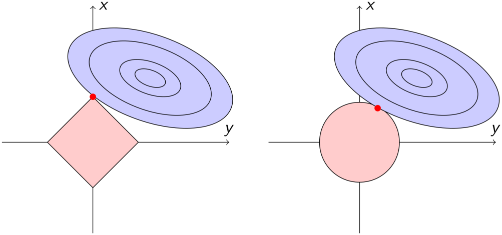
$\Rightarrow$ In other words, $\text{supp}(\boldsymbol{\beta}_{(-i)}) := \{j : \beta_{(-i),j} \neq 0\} \equiv \mathcal{N}_i$

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

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

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 → Sparse solution

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

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

Neighborhood-based sparse regression

- Cycle over vertices $i \in V$ and estimate $\hat{\mathcal{N}}_i = \text{supp}(\hat{\boldsymbol{\beta}}_{(-i)})$, where

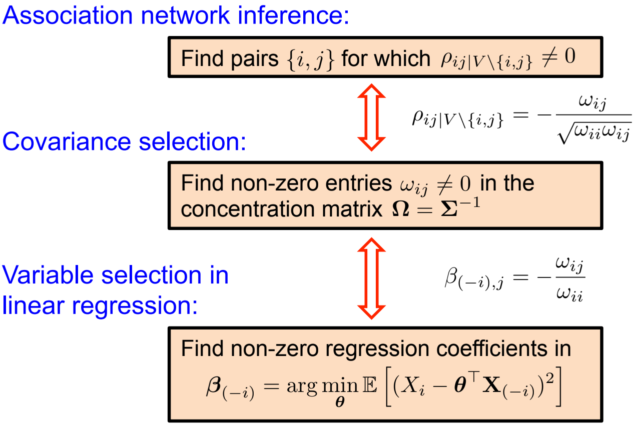
$$\hat{\boldsymbol{\beta}}_{(-i)} \in \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^{N_v - 1}} \left\{ \sum_{p=1}^{n} (x_{pi} - \mathbf{x}_{p,\setminus i}^{\top} \boldsymbol{\beta})^2 + \lambda \|\boldsymbol{\beta}\|_1 \right\}$$

$\Rightarrow$ Separable lasso problems per vertex

- No guarantee that $\hat{\beta}_{(-i),j} \neq 0$ implies $\hat{\beta}_{(-j),i} \neq 0$ and vice versa
  - Combine information in $\hat{\mathcal{N}}_i$ and $\hat{\mathcal{N}}_j$ to enforce symmetry
  - OR rule: $(i,j) \in E$ if $\beta_{(-i),j} \neq 0$ or $\beta_{(-j),i} \neq 0$. Likewise, AND rule
- Support consistency for either rule [Meinshausen-Bühlmann'06]
  - Suitable choice of λ, sparsity of $\boldsymbol{\Omega}_0$, and sample complexity $n \ll N_v$

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

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

Comparative summary

- Parallelizable neighborhood-based regression (NBR)
  - $\Rightarrow$ Conditional likelihood per vertex $i \in V$, disregards $\mathbf{\Omega} \succeq \mathbf{0}$
  - $\Rightarrow$ Tends to be computationally faster
- Graphical Lasso minimizes a (regularized) global likelihood

$$\mathcal{L}(\mathbf{\Omega}) = \log\det\mathbf{\Omega} - \text{trace}(\hat{\mathbf{\Sigma}}\mathbf{\Omega})$$

  - $\Rightarrow$ Tends to be (statistically) more efficient
- NBR method tractable even for discrete or mixed graphical models
  - $\Rightarrow$ Ising-model selection for $\mathbf{X} \in \{-1, +1\}^{N_v}$

P. Ravikumar et al, "High-dimensional Ising model selection using ℓ_1-regularized logistic regression," *Ann. Statist.*, 2010

Tomographic inference

Network topology inference problems

Link prediction

Case study: Predicting lawyer collaboration

Inference of association networks

Case study: Inferring genetic regulatory interactions

Tomographic network topology inference

Case study: Computer network topology identification

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 inference problem

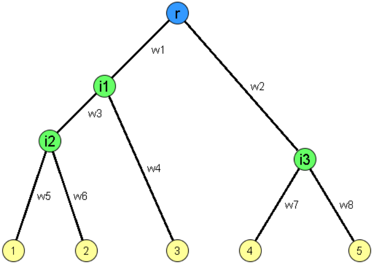
  $\Rightarrow$ Inverse problem: invert mapping from ‘internal’ to ‘external’

  $\Rightarrow$ 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

Trees

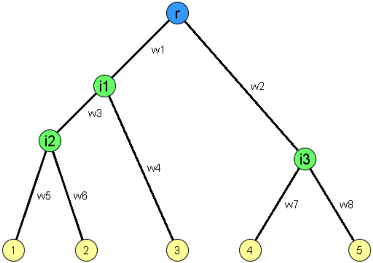
- **Def:** an undirected tree $T = (V_T, E_T)$ is a connected acyclic graph



- 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

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

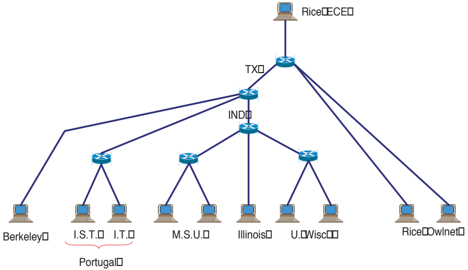
Find a tree $\hat{T} \in \mathcal{T}_{N_L}$ that 'best' explains the data $\{\mathbf{x}_1, \ldots, \mathbf{x}_{N_L}\}$

- 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

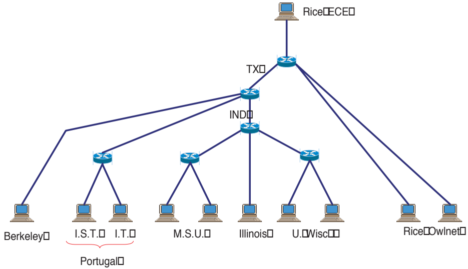


- For leaves $\ell \in L$, consider the indicator $X_\ell = \mathbb{I}\{\ell \text{ received the probe}\}$

  $\Rightarrow$ 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 tree enforces hereditary constraints

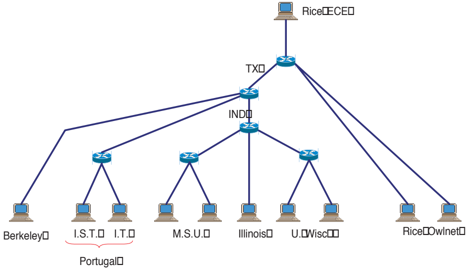
$\Rightarrow$ $X_{a(U)} = 0$ implies $X_j = 0$ for all $j \in U$

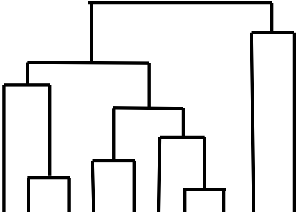
$\Rightarrow$ If $X_j = 1$ for at least one $j \in d(k)$, then $X_k = 1$

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

- 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

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

- 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

- 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

- 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

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

Case study

Network topology inference problems

Link prediction

Case study: Predicting lawyer collaboration

Inference of association networks

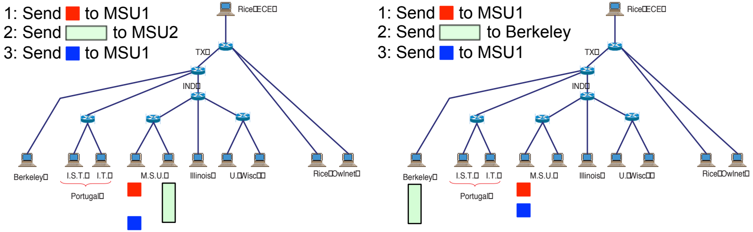
Case study: Inferring genetic regulatory interactions

Tomographic network topology inference

Case study: Computer network topology identification

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



1: Send 🟥 to MSU1
2: Send 🟩 to MSU2
3: Send 🟦 to MSU1

1: Send 🟥 to MSU1
2: Send 🟩 to Berkeley
3: Send 🟦 to MSU1

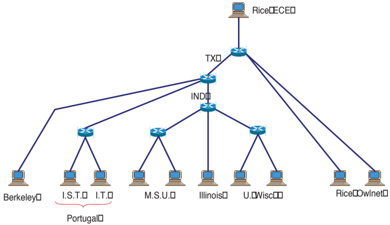
- If paths overlap, large probe induces high delay in the second small one

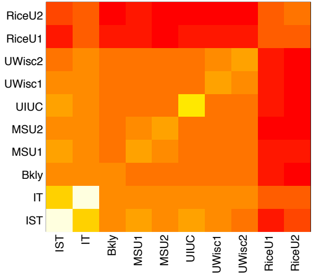
  $\Rightarrow$ Large TDD values indicative of close leaves in the tree topology

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 $x_{jk} \sim \mathcal{N}(\mu_{jk}, \sigma_{jk}^2)$

  $\Rightarrow$ Independence of the x_{jk} reasonable by experimental setup

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

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

- 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

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

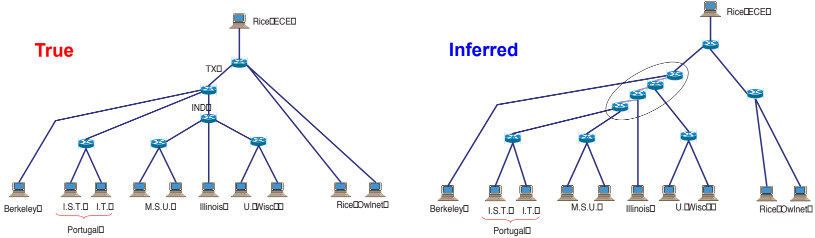
$\Rightarrow$ Recall $L(k)$ is the set of leaves descended by k

Inferred topology

- Ground-truth topology obtained via `traceroute` probing

  ⇒ `traceroute` replies often ‘turned-off’ for security

  ⇒ Tomographic topology inference approaches relevant!



- ALT-inferred topology binary by construction ⇒ introduces artifacts

- R. Castro et al, “Likelihood-based hierarchical clustering,” *IEEE Trans. Signal Process.*, vol. 52, pp. 2308-2321, 2004

Glossary

- Topology inference
- Link prediction
- Scoring methods
- Logistic regression
- Missing data
- Latent variable models
- Latent eigenmodel
- Association networks
- Correlation networks
- Pearson correlation
- Fisher's transformation
- Multiple testing
- False discovery rate
- Gene-regulatory networks
- Microarray data
- Partial correlation
- Gaussian graphical models
- Concentration matrix
- Variable selection
- Network tomography
- Muticast probing
- Shared packet loss
- Sandwich probing
- Time-delay difference