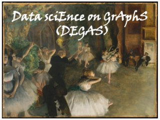
Universidad Rey Juan Carlos

Connecting the dots: Leveraging GSP to learn graphs from nodal observations

Antonio G. Marques
King Juan Carlos University - Madrid, Spain

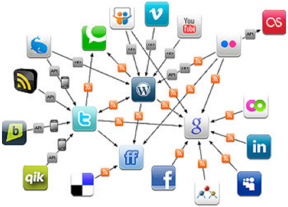
Thanks to: S. Segarra, G. Mateos, A. Ribeiro, C. Uhler, A. Buciulea, S. Rey, R. Shafipour, M. Navarro, (Spanish NSF: PID2019-105032gb-i00)



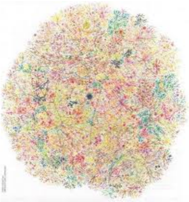
November 18, 2021 – IEEE SPS-DSI DEGAS Webinar Series

Network Science analytics

Online social media



Internet

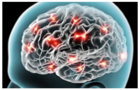


Clean energy and grid analytics

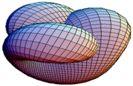
- Network as graph $G = (\mathcal{V}, \mathcal{E})$: encode pairwise relationships
- Desiderata: Process, analyze and learn from network data [Kolaczyk'09]
  - $\Rightarrow$ Use G to study graph signals, data associated with nodes in $\mathcal{V}$
- Ex: Opinion profile, buffer congestion levels, neural activity, epidemic

- **Goal:** Process, analyze and learn from graph signals
  - $\Rightarrow$ **As.:** Signal properties related to topology of G (e.g., locality)
- **Graph SP**: broaden classical SP to graph signals [Shuman'13,Sandryhaila'13]
  - $\Rightarrow$ Main actors: nodal signals $\mathbf{x}$, $\mathbf{y}$, $\mathbf{w}$ and graph shift operator $\mathbf{S}$
  - $\Rightarrow$ Algorithms that fruitfully leverage this relational structure

Interpolate a brain signal from local observations



Compress a signal in an irregular domain

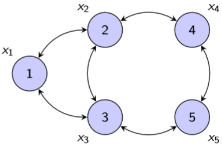


Localize the source of a rumor

Smooth an observed network profile

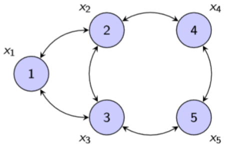
- GSP leverages $\mathbf{S}$ to define: Graph Fourier Transform and Graph Filters

- Graph G with N nodes and adjacency $\mathbf{A}$
  - $\Rightarrow A_{ij} =$ Proximity between i and j
- Define a signal $\mathbf{x} \in \mathbb{R}^N$ on top of the graph
  - $\Rightarrow x_i =$ Signal value at node i



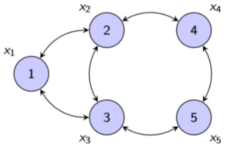
Universidad Rey Juan Carlos

- Graph G with N nodes and adjacency $\mathbf{A}$
  - $\Rightarrow A_{ij}$ = Proximity between i and j
- Define a signal $\mathbf{x} \in \mathbb{R}^N$ on top of the graph
  - $\Rightarrow x_i$ = Signal value at node i



- Associated with G is the graph-shift operator $\mathbf{S} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1} \in \mathbb{R}^{N \times N}$
  - $\Rightarrow S_{ij} = 0$ for $i \neq j$ and $(i, j) \notin \mathcal{E}$ (local structure in G)
  - $\Rightarrow$ Ex: $\mathbf{A}$ and Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$ matrices

- Graph G with N nodes and adjacency $\mathbf{A}$
  - $\Rightarrow A_{ij}$ = Proximity between i and j
- Define a signal $\mathbf{x} \in \mathbb{R}^N$ on top of the graph
  - $\Rightarrow x_i$ = Signal value at node i



- Associated with G is the graph-shift operator $\mathbf{S} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{-1} \in \mathbb{R}^{N \times N}$
  - $\Rightarrow S_{ij} = 0$ for $i \neq j$ and $(i, j) \notin \mathcal{E}$ (local structure in G)
  - $\Rightarrow$ Ex: $\mathbf{A}$ and Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$ matrices
- Graph filters $\rightarrow$ Matrix polynomials: $\mathbf{H} = \sum_{l=0}^{N-1} h_l \mathbf{S}^l = \mathbf{V}\text{diag}(\tilde{\mathbf{h}})\mathbf{V}^{-1}$
- Graph SP $\rightarrow$ Exploit structure encoded in $\mathbf{S}$ to process $\mathbf{x}$
- Take the reverse path. How to use GSP to infer the graph topology?
  - $\Rightarrow$ Talk's key GSP concepts: graph signal smoothness and stationarity

- Total variation of signal $\mathbf{x}$ with respect to $\mathbf{L} = \mathbf{D} - \mathbf{A} = \mathbf{B}\mathbf{B}^T$

$$\mathrm{TV}(\mathbf{x}) = \mathbf{x}^T \mathbf{L} \mathbf{x} = \sum_{i,j=1, j>i}^{N} A_{ij}(x_i - x_j)^2$$

$\Rightarrow$ Smoothness measure on the graph G (Dirichlet energy)

- For $\mathbf{L}$ eigenvecs $\mathbf{V} = [\mathbf{v}_0, ..., \mathbf{v}_{N-1}]$ $\Rightarrow \mathrm{TV}(\mathbf{v}_k) = \lambda_k$ $\Rightarrow \mathrm{TV}(\mathbf{1}) = 0$

  $\Rightarrow \lambda_0 = 0$ and can view $\lambda_0 = 0 \leq \cdots \leq \lambda_{N-1}$ as frequencies

- Total variation of signal $\mathbf{x}$ with respect to $\mathbf{L}=\mathbf{D}-\mathbf{A}=\mathbf{B}\mathbf{B}^T$

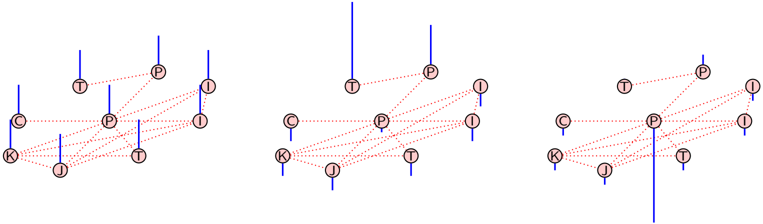
$$\mathrm{TV}(\mathbf{x})=\mathbf{x}^T\mathbf{L}\mathbf{x}=\sum_{i,j=1,j>i}^{N} A_{ij}(x_i-x_j)^2$$

$\Rightarrow$ Smoothness measure on the graph G (Dirichlet energy)

- For $\mathbf{L}$ eigenvecs $\mathbf{V}=[\mathbf{v}_0,\ldots,\mathbf{v}_{N-1}]$ $\Rightarrow$ $\mathrm{TV}(\mathbf{v}_k)=\lambda_k$ $\Rightarrow$ $\mathrm{TV}(\mathbf{1})=0$

  $\Rightarrow$ $\lambda_0=0$ and can view $\lambda_0=0\leq\cdots\leq\lambda_{N-1}$ as frequencies

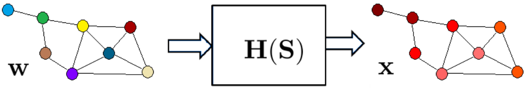
- Ex: gene network, $N=10$, $k=0$, $k=1$, $k=9$



- Random signals over a graph G $\Rightarrow$ (Statistical) Properties related to G
  - $\Rightarrow$ In time, stationarity is a pervasive, tractable and fruitful model

Stationary graph signal

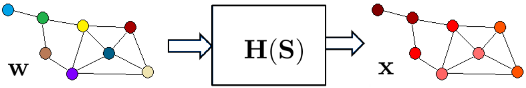
Def: A graph signal $\mathbf{x}$ is stationary with respect to the shift $\mathbf{S}$ if and only if $\mathbf{x} = \mathbf{H}\mathbf{w}$, where $\mathbf{H} = \sum_{l=0}^{L-1} h_l \mathbf{S}^l$ and $\mathbf{w}$ is white.



- Random signals over a graph G $\Rightarrow$ (Statistical) Properties related to G
  - $\Rightarrow$ In time, stationarity is a pervasive, tractable and fruitful model

Stationary graph signal

Def: A graph signal $\mathbf{x}$ is stationary with respect to the shift $\mathbf{S}$ if and only if $\mathbf{x} = \mathbf{H}\mathbf{w}$, where $\mathbf{H} = \sum_{l=0}^{L-1} h_l \mathbf{S}^l$ and $\mathbf{w}$ is white.



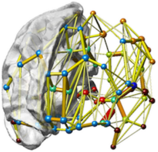
- The covariance matrix of the stationary signal $\mathbf{x}$ is a polynomial on $\mathbf{S}$

$$\mathbf{C}_x = \mathbb{E}\left[\mathbf{Hw}(\mathbf{Hw})^T\right] = \mathbf{H}\mathbb{E}\left[\mathbf{w}\mathbf{w}^T\right]\mathbf{H}^T = \mathbf{H}^2 = h_0\mathbf{I}+2h_0h_1\mathbf{S}+(2h_0h_2+h_1^2)\mathbf{S}^2\ldots$$

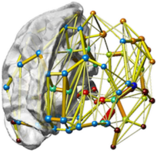
- Key: $\mathbf{C}_x$ and $\mathbf{S}$ simultaneously diagonalizable
  - $\Rightarrow$ eigenvecs($\mathbf{C}_x$)=eigenvecs($\mathbf{S}$) AND $\mathbf{C}_x\mathbf{S}=\mathbf{S}\mathbf{C}_x$

What is this talk about?

- Learning graphs from nodal observations
- Fundamental problem in statistics (later)
- Key in neuroscience [Sporns'10]
  - $\Rightarrow$ Functional network from fMRI signals

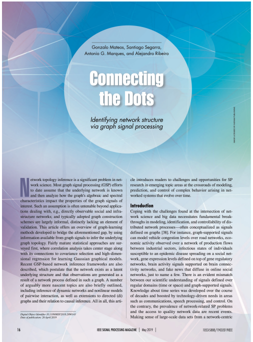


- Learning graphs from nodal observations
- Fundamental problem in statistics (later)
- Key in neuroscience [Sporns'10]
  - $\Rightarrow$ Functional network from fMRI signals



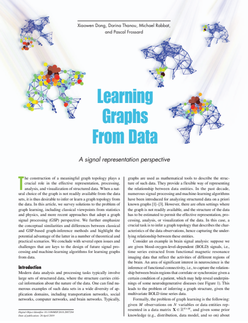
- Most GSP works: how known graph $\mathbf{S}$ affects signals and filters
- Here, reverse path: how to use GSP to infer the graph topology?
  - Graphical models [Egilmez et al'16], [Rabbat'17], [Kumar et al'19], ...
  - Smooth signals [Dong et al'15], [Kalofolias'16], [Sardellitti et al'17], ...
  - Graph filtering models [Shafipour et al'17], [Thanou et al'17], ...
  - Stationary signals [Pasdeloup et al'15], [Segarra et al'16], ...
  - Directed graphs [Mei-Moura'15], [Shen et al'16], ...

- Recent tutorials on learning graphs from data
  - IEEE Signal Processing Magazine and Proceedings of the IEEE


Gonzalo Mateos, Santiago Segarra, Antonio G. Marques, and Alejandro Ribeiro

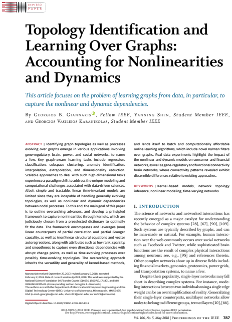
Connecting the Dots

Identifying network structure via graph signal processing


Xiaowen Dong, Dorina Thanou, Michael Rabbat, and Pascal Frossard

Learning Graphs From Data

A signal representation perspective


Topology Identification and Learning Over Graphs: Accounting for Nonlinearities and Dynamics

This article focuses on the problem of learning graphs from data, in particular, to capture the nonlinear and dynamic dependencies.

- IEEE Trans. on Signal and Information Processing over Networks
  - Special issue on Network Topology Inference (2020)

Network topology inference problems

- **Q:** If G (or a portion thereof) is unobserved, can we infer it from data?
- Formulate as a statistical inference task, i.e. given
  - Signal measurements x_i at some or all vertices $i \in \mathcal{V}$
  - Indicators y_{ij} of edge status for some vertex pairs $\{i, j\} \in \mathcal{V}_{obs}^{(2)}$
  - A collection G of candidate graphs G

  Goal: infer the topology of the network graph $G(\mathcal{V}, \mathcal{E})$
- Bring to bear existing statistical concepts and tools

  $\Rightarrow$ Study identifiability, consistency, robustness, complexity

Network topology inference problems

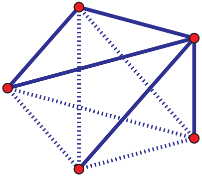
Universidad Rey Juan Carlos

- **Q:** If G (or a portion thereof) is unobserved, can we infer it from data?
- Formulate as a statistical inference task, i.e. given
  - Signal measurements x_i at some or all vertices $i \in \mathcal{V}$
  - Indicators y_{ij} of edge status for some vertex pairs $\{i, j\} \in \mathcal{V}^{(2)}_{obs}$
  - A collection G of candidate graphs G

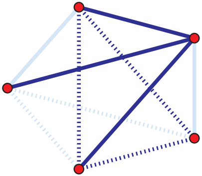
  Goal: infer the topology of the network graph $G(\mathcal{V}, \mathcal{E})$
- Bring to bear existing statistical concepts and tools

  $\Rightarrow$ Study identifiability, consistency, robustness, complexity
- Three canonical network topology inference problems [Kolaczyk'09]
  - (i) Link prediction
  - (ii) Association network inference $\leftarrow$ Focus of this talk
  - (iii) Tomographic network topology inference

Link prediction



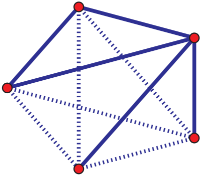
Original graph



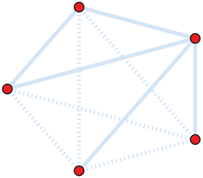
Link prediction

- Edge status is only observed for some subset of pairs $\mathcal{V}^{(2)}_{obs} \subset \mathcal{V}^{(2)}$
- **Goal:** predict edge status for all other pairs, i.e., $\mathcal{V}^{(2)}_{miss} = \mathcal{V}^{(2)} \setminus \mathcal{V}^{(2)}_{obs}$
- Approach address the problem leveraging:
  - a) topological info only (nodal features) and/or
  - b) **nodal signals** $\mathbf{x} = [x_1, \ldots, x_N]^\top$

Association network inference



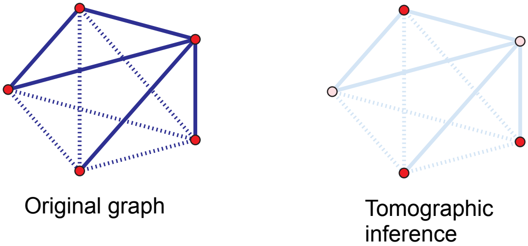
Original graph



Association network inference

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

Universidad Rey Juan Carlos



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

Preliminaries and problem statement

Statistical methods for network topology inference

GSP methods for network topology inference: smoothness

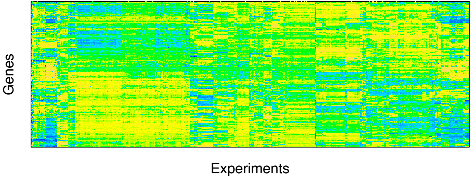
GSP methods for network topology inference: stationarity

Stationarity as an overreaching model

Conclusions and future lines of work

Learning a graph from nodal observations

"Given a collection $\mathbf{X} := [\mathbf{x}_1, ..., \mathbf{x}_P] \in \mathbb{R}^{N \times P}$ of graph signal observations supported on the unknown graph $G(\mathcal{V}, \mathcal{E}, \mathbf{W})$ find an optimal $\mathbf{S}$"



- Ill-posed problem: optimality, priors, regularizations
- Most classical approaches focus on pairwise similarities
  - $\Rightarrow$ User-defined similarity $\texttt{sim}(i,j) = f(x_i, x_j)$ specifies edges $(i,j) \in \mathcal{E}$
- More recent approaches look at G as a whole: mapping from $\mathbf{X}$ to $\mathbf{S}$
- We start by reviewing classical approaches in statistics

- Pearson product-moment correlation as sim between vertex pairs

$$\mathtt{sim}(i,j) := \rho_{ij} = \frac{\mathrm{cov}[x_i, x_j]}{\sqrt{\mathrm{var}\left[x_i\right]\mathrm{var}\left[x_j\right]}}, \quad i,j \in \mathcal{V}$$

- Inference of edges $\mathcal{E}$ $\Leftrightarrow$ Inference of non-zero correlations

   $\Rightarrow$ Typically approached as a testing problem: $H_0 : \rho_{ij} = 0$ vs. $H_1 : \rho_{ij} \neq 0$

Universidad Rey Juan Carlos

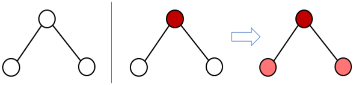
- Pearson product-moment correlation as $\mathtt{sim}$ between vertex pairs

$$\mathtt{sim}(i,j) := \rho_{ij} = \frac{\mathrm{cov}[x_i, x_j]}{\sqrt{\mathrm{var}[x_i]\,\mathrm{var}[x_j]}}, \quad i,j \in \mathcal{V}$$

- Inference of edges $\mathcal{E}$ $\Leftrightarrow$ Inference of non-zero correlations
  - $\Rightarrow$ Typically approached as a testing problem: $H_0 : \rho_{ij} = 0$ vs. $H_1 : \rho_{ij} \neq 0$
- Find sample covariance $\hat{\mathbf{C}} = \mathbf{X}\mathbf{X}^T$, then $\hat{\rho}_{ij} = \hat{C}_{ij} / \sqrt{\hat{C}_{ii}\hat{C}_{jj}}$
  - $\Rightarrow$ Edge exists if: $0.5 \log\left(\frac{1+\hat{\rho}_{ij}}{1-\hat{\rho}_{ij}}\right) > \frac{z_{\alpha/2}}{\sqrt{P-3}}$, with $P_{FA} = \alpha$ [Kol'09]
- Non-zero entries of the GSO $\mathbf{S}$:
  - $\Rightarrow$ $S_{ij} = \hat{\rho}_{ij}, \quad S_{ij} = \hat{C}_{ij}, \quad S_{ij} = 1_{\{H_1\}}, \quad S_{ij} = f(\hat{\rho}_{ij}), \ldots$
  - $\Rightarrow$ Sparsification of the covariance / correlation matrix

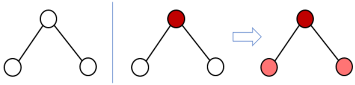
- Use correlations carefully: 'correlation does not imply causation'
  - Vertices $i, j \in \mathcal{V}$ may have high ρ_{ij} because they influence each other
- But ρ_{ij} could be high if both i, j influenced by a third vertex $k \in \mathcal{V}$

  $\Rightarrow$ Correlation networks may declare edges due to confounders



- Use correlations carefully: 'correlation does not imply causation'
  - Vertices $i, j \in \mathcal{V}$ 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 $k \in \mathcal{V}$

  $\Rightarrow$ Correlation networks may declare edges due to confounders



- Partial correlations better capture direct influence among vertices
  - For $i, j \in \mathcal{V}$ consider latent vertices $\mathcal{V}_{-ij} = \mathcal{V} \setminus \{i, j\}$, then partial correlation of x_i and x_j, adjusting for $\mathbf{x}_{-ij} = [x_1, .., x_{i-1}, x_{i+1}, ..., x_{j-1}, x_{j+1}, ..., x_N]^T$ is

$$\rho_{ij|\mathcal{V}_{-ij}} = \frac{\text{cov}[x_i, x_j \mid \mathbf{x}_{-ij}]}{\sqrt{\text{var}\left[x_i \mid \mathbf{x}_{-ij}\right] \text{var}\left[x_j \mid \mathbf{x}_{-ij}\right]}}, \quad i, j \in \mathcal{V}$$

- **Q:** How do we obtain these partial correlations?

Partial correlations and covariance selection

- **Def:** the precision matrix of $\mathbf{x}$ is $\boldsymbol{\Theta} := \mathbf{C}^{-1}$, with $\mathbf{C}$ being its covariance
- Key result: The partial correlations can be expressed as

$$\rho_{ij|\mathcal{V}_{-ij}} = -\frac{\Theta_{ij}}{\sqrt{\Theta_{ii}\Theta_{jj}}}$$

- Edges $\mathcal{E}$ in the graph $G \Leftrightarrow$ Non-zero entries in $\boldsymbol{\Theta}$

Partial correlations and covariance selection

- **Def:** the precision matrix of $\mathbf{x}$ is $\boldsymbol{\Theta} := \mathbf{C}^{-1}$, with $\mathbf{C}$ being its covariance
- Key result: The partial correlations can be expressed as

$$\rho_{ij|\mathcal{V}_{-ij}} = -\frac{\Theta_{ij}}{\sqrt{\Theta_{ii}\Theta_{jj}}}$$

- Edges $\mathcal{E}$ in the graph $G \Leftrightarrow$ Non-zero entries in $\boldsymbol{\Theta}$
  - $\Rightarrow$ Inferring G from $\mathbf{X}$ known as covariance selection [Dempster'74]
  - $\Rightarrow$ Classical methods are 'network-agnostic,' and effectively test

$$H_0 : \rho_{ij|\mathcal{V}_{-ij}} = 0 \quad \text{vs.} \quad H_1 : \rho_{ij|\mathcal{V}_{-ij}} \neq 0$$

  - $\Rightarrow$ Often not scalable, and $P \ll N$ so estimation of $\hat{\mathbf{C}}$ challenging
- Under Gaussianity $\rho_{ij|\mathcal{V}_{-ij}} = 0$ iff x_i and x_j are conditionally independent
  - $\Rightarrow$ Also known as Gaussian Markov random field (GMRF)
  - $\Rightarrow$ A popular particular instance of partial correlation networks

- Sparsity-regularized maximum-likelihood estimator of $\mathbf{\Theta}$ [Yuan'07]

$$\hat{\mathbf{\Theta}} = \arg\max_{\mathbf{\Theta} \succeq \mathbf{0}} \left\{ \log\det \mathbf{\Theta} - \operatorname{trace}(\hat{\mathbf{C}}\mathbf{\Theta}) - \lambda \|\mathbf{\Theta}\|_1 \right\}$$

$\Rightarrow$ Effective when $P \ll N$, encourages interpretable models

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

Graphical Lasso (GL)

- Sparsity-regularized maximum-likelihood estimator of $\mathbf{\Theta}$ [Yuan'07]

$$\hat{\mathbf{\Theta}} = \arg\max_{\mathbf{\Theta}\succeq\mathbf{0}} \left\{ \log\det\mathbf{\Theta} - \text{trace}(\hat{\mathbf{C}}\mathbf{\Theta}) - \lambda\|\mathbf{\Theta}\|_1 \right\}$$

$\Rightarrow$ Effective when $P \ll N$, encourages interpretable models

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

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

$$\|\hat{\mathbf{\Theta}} - \mathbf{\Theta}_0\|_2 \leq \sqrt{\frac{d_{\max}^2 \log N}{P}} \quad \text{w.h.p.}$$

$\Rightarrow$ Ground-truth $\mathbf{\Theta}_0$, maximum nodal degree $d_{\max}$

- Support consistency for $P = \Omega(d_{\max}^2 \log N)$ [Ravikumar'11]
- Partial correlation / GL: estimate GSO $\mathbf{S}$ sparsifying $\mathbf{C}^{-1}$

Case study: Inferring 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

- 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

Case study: Inferring gene-regulatory interactions

Universidad Rey Juan Carlos

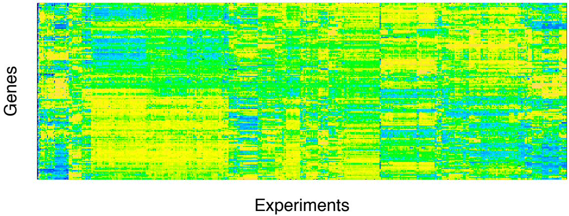
- Genes are segments of DNA encoding information about cell functions
- Such information used in the expression of genes
  - $\Rightarrow$ Creation of biochemical products, i.e., RNA or proteins
- Regulation of a gene refers to the control of its expression

  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

Regulatory interactions among E. coli genes

Universidad Rey Juan Carlos

- 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

  ⇒ 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

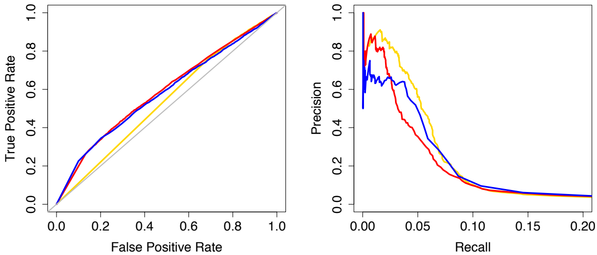
- In all cases applied Fisher transformation to obtain z-scores

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

- 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'07]

Rationale

- Seek graphs on which data admit certain regularities
  - Nearest-neighbor prediction (a.k.a. graph smoothing)
  - Semi-supervised learning
- Many real-world graph signals are smooth
  - Graphs based on similarities among vertex attributes
  - Network formation driven by homophily, proximity in latent space

Universidad Rey Juan Carlos

Rationale

- Seek graphs on which data admit certain regularities
  - Nearest-neighbor prediction (a.k.a. graph smoothing)
  - Semi-supervised learning
- Many real-world graph signals are smooth
  - Graphs based on similarities among vertex attributes
  - Network formation driven by homophily, proximity in latent space

Problem statement

Given observations $\mathbf{X} := [\mathbf{x}_1, ..., \mathbf{x}_P] \in \mathbb{R}^{N \times P}$, identify a graph G such that signals in $\mathbf{X}$ are smooth on G.

- Criterion: Dirichlet energy on the graph G with Laplacian $\mathbf{L}$

  $\Rightarrow$ Search for the GSO $\mathbf{S} = \mathbf{L}$ such that $\text{TV}(\mathbf{x}) = \mathbf{x}^T \mathbf{L} \mathbf{x}$ small

$$\text{TV}(\mathbf{X}) = \sum_{p=1}^{P} \mathbf{x}_p^T \mathbf{L} \mathbf{x}_p$$

Formulation and algorithm

Universidad Rey Juan Carlos

- Noiseless obs. $\Rightarrow$ Objective: Smoothness + graph regularization [Dong16]

$$\mathbf{L}^* = \arg\min_{\mathbf{L}} \left\{ \sum_{p=1}^{P} \mathbf{x}_p^T \mathbf{L} \mathbf{x}_p + \frac{\beta}{2} \|\mathbf{L}\|_F^2 \right\}$$

$$\text{s. to} \quad \text{trace}(\mathbf{L}) = N,\ \mathbf{L1} = \mathbf{0},\ L_{ij} = L_{ji} \leq 0,\ i \neq j$$

$\Rightarrow$ Sparsity $\|\mathbf{L}\|_1$ redundant due to linear constraints

Formulation and algorithm

- Noiseless obs. $\Rightarrow$ Objective: Smoothness + graph regularization [Dong16]

$$\mathbf{L}^* = \arg\min_{\mathbf{L}} \left\{ \sum_{p=1}^{P} \mathbf{x}_p^T \mathbf{L} \mathbf{x}_p + \frac{\beta}{2} \|\mathbf{L}\|_F^2 \right\}$$

$$\text{s. to} \quad \text{trace}(\mathbf{L}) = N,\ \mathbf{L}\mathbf{1} = \mathbf{0},\ L_{ij} = L_{ji} \leq 0,\ i \neq j$$

$\Rightarrow$ Sparsity $\|\mathbf{L}\|_1$ redundant due to linear constraints

- Noisy obs. $\Rightarrow$ Objective must include fidelity term [Dong16]

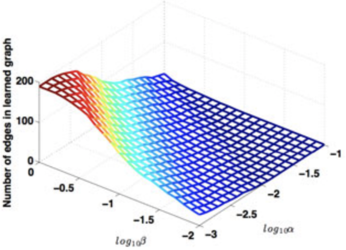
$$\mathbf{L}^* = \arg\min_{\mathbf{L},\mathbf{Y}} \left\{ \|\mathbf{X} - \mathbf{Y}\|_F^2 + \alpha \sum_{p=1}^{P} \mathbf{y}_p^T \mathbf{L} \mathbf{y}_p + \frac{\beta}{2} \|\mathbf{L}\|_F^2 \right\} \quad \text{s. to} \ \ \text{trace}(\mathbf{L}) = N, \ldots$$

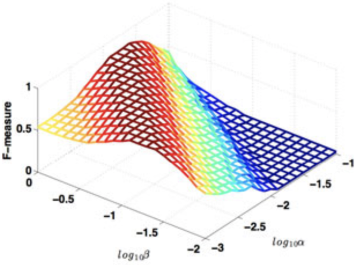
$\Rightarrow$ Not jointly convex in $\mathbf{L}$ and $\mathbf{Y}$, but bi-convex

- Algorithmic approach: alternating minimization (AM), $O(N^3)$ cost
  - (S1) Fixed $\mathbf{Y}$: solve for $\mathbf{L}$ via interior-point method, ADMM
  - (S2) Fixed $\mathbf{L}$: low-pass graph-filter smoother $\mathbf{Y} = (\mathbf{I} + \alpha \mathbf{L})^{-1} \mathbf{X}$

Impact of regularizers on sparsity and accuracy

- Generate multiple signals on a synthetic Erdős-Rényi graph
- Recover the graph for different values of α and β





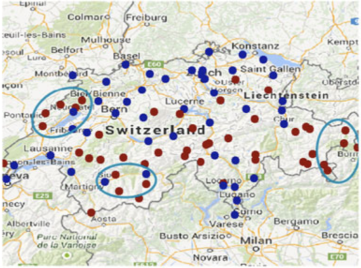
- More edges promoted by increasing β and decreasing α
- In the low noise regime, the ratio β/α determines behavior

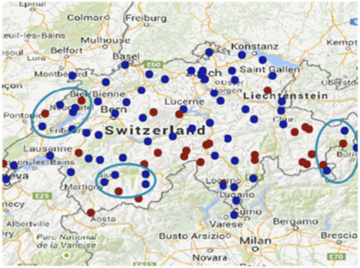
Learning a temperature graph in Switzerland

Universidad Rey Juan Carlos

- 89 stations measuring monthly temperatures (1981-2010) [Meteoswiss]
- Learn a graph on which the temperatures vary smoothly
- Geographical distance not a good idea $\Rightarrow$ different altitudes
- Recover altitude partition from spectral clustering

  $\Rightarrow$ Red (high stations) and blue (low stations) clusters
- k-means applied directly to the temperatures (right) fails





- Smoothness is a deterministic metric and graph regularizers are needed
  - $\Rightarrow$ Note that $\sum_{p=1}^{P} \mathbf{x}_p^T \mathbf{L} \mathbf{x}_p = \sum_{p=1}^{P} \text{trace}(\mathbf{x}_p \mathbf{x}_p^T \mathbf{L}) = P\text{trace}(\hat{\mathbf{C}}\mathbf{L})$
  - $\Rightarrow$ Use as regularizer $\log\det(\mathbf{L}) - \lambda \|\mathbf{L}\|_1$

$$\mathbf{L}^* = \arg \max_{\mathbf{L} \succeq \mathbf{0}, \gamma \geq 0} \left\{ \log\det \mathbf{L} - \text{trace}(\hat{\mathbf{C}}\mathbf{L}) - \lambda \|\mathbf{L}\|_1 \right\}$$
$$\text{s. to } \mathbf{L1} = \mathbf{0},\ L_{ij} \leq 0,\ i \neq j$$

- $\mathbf{\Theta} = \mathbf{L}$ GMRF with Laplacian constraints!!
  - $\Rightarrow$ KO: $\mathbf{L}$ singular (improper GMRF)
  - $\Rightarrow$ Use $\mathbf{\Theta} = \mathbf{L} + \gamma \mathbf{I}$ $\Rightarrow$ Proper GMRF via diagonal loading [Lake'07]
- GMRF with Laplacian constr. favors graphs over which $\mathbf{X}$ is smooth
  - $\Rightarrow$ Efficient algorithms, topological constraints [Pavez'17], [Zhao'19]

Connecting the dots: GSP methods

- **Def:** A graph signal $\mathbf{x}$ is stationary with respect to the shift $\mathbf{S}$ if and only if $\mathbf{x} = \mathbf{H}\mathbf{w}$, where $\mathbf{H} = \sum_{l=0}^{L-1} h_l \mathbf{S}^l$ and $\mathbf{w}$ is white.
  - $\Rightarrow$ **Coro:** The covariance matrix $\mathbf{C} = \mathbb{E}\left[\mathbf{x}\mathbf{x}^T\right]$ is a polynomial on $\mathbf{S}$.

Graph learning based on stationarity

Find the sparsest GSO such that $\mathbf{S}$ can be (approximately) mapped to $\hat{\mathbf{C}} = \frac{1}{P}\mathbf{X}\mathbf{X}^T$ by a polynomial

Observations

(a) Our approach says mapping $\mathbf{C} \to \mathbf{S}$ is polynomial (analytic)

(b) Correlation methods $\Rightarrow \mathbf{C} = \mathbf{S}$ eigenvalues are kept unchanged

(c) Precision methods $\Rightarrow \mathbf{C} = \mathbf{S}^{-1}$ eigenvalues are inverted

- Sparsifying entries of $\mathbf{C}$ or $\mathbf{C}^{-1}$ vs sparsest transformation (more ill posed)

- Finding $\mathbf{S}$ from $\mathbf{C} = h_0^2\mathbf{I} + 2h_0h_1\mathbf{S} + (2h_0h_2 + h_1^2)\mathbf{S}^2$ non-convex but...

- Finding $\mathbf{S}$ from $\mathbf{C} = h_0^2\mathbf{I} + 2h_0h_1\mathbf{S} + (2h_0h_2 + h_1^2)\mathbf{S}^2$ non-convex but...
- Approach 1 [Segarra'16],[Pasdeloup'16]: $[\mathbf{v}_1, ..., \mathbf{v}_N] := \mathrm{eig}(\hat{\mathbf{C}})$ and

$$\mathbf{S}^* = \underset{\boldsymbol{\lambda}}{\operatorname{argmin}} \ \|\mathbf{S}\|_0 \quad \text{s. to} \quad \mathbf{S} = \sum_{k=1}^{N} \lambda_k \mathbf{v}_k \mathbf{v}_k^T, \ \ \mathbf{S} \in \mathcal{S}$$

$\Rightarrow$ Set $\mathcal{S}$ contains all admissible scaled adjacency (Laplacian) matrices

- Finding $\mathbf{S}$ from $\mathbf{C} = h_0^2\mathbf{I} + 2h_0h_1\mathbf{S} + (2h_0h_2 + h_1^2)\mathbf{S}^2$ non-convex but...
- Approach 1 [Segarra'16],[Pasdeloup'16]: $[\mathbf{v}_1, ..., \mathbf{v}_N] := \text{eig}(\hat{\mathbf{C}})$ and

$$\mathbf{S}^* = \underset{\boldsymbol{\lambda}}{\text{argmin}} \ \|\mathbf{S}\|_0 \quad \text{s. to} \quad \mathbf{S} = \sum_{k=1}^{N} \lambda_k \mathbf{v}_k \mathbf{v}_k^T, \ \ \mathbf{S} \in \mathcal{S}$$

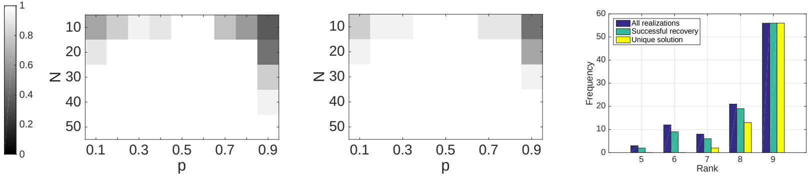
$\Rightarrow$ Set $\mathcal{S}$ contains all admissible scaled adjacency (Laplacian) matrices

- Approach 2 [Segarra'17]: Use $\hat{\mathbf{C}}$ directly and

$$\mathbf{S}^* = \underset{\mathbf{S}}{\text{argmin}} \ \|\mathbf{S}\|_0 \quad \text{s. to} \quad \hat{\mathbf{C}}\mathbf{S} = \mathbf{S}\hat{\mathbf{C}}, \ \ \mathbf{S} \in \mathcal{S}$$

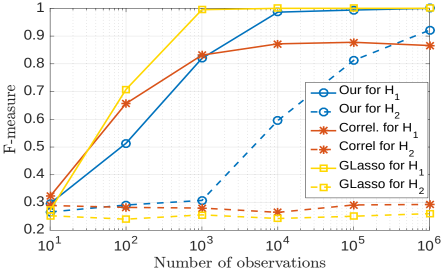
$\Rightarrow$ Equivalent if $\mathbf{S}$ and $\hat{\mathbf{C}}$ have non-repeated eigenvalues

- More ill-posed than (partial) correlation nets $\Rightarrow$ Theoretical results for:
  - $\Rightarrow$ Identifiability under perfect observations [Segarra'17]
  - $\Rightarrow$ Errors in the covariance, incomplete eigenvectors (singular $\hat{\mathbf{C}}$)
- Recovery rates: Erdős-Rényi varying N and edge probability p
  - $\Rightarrow$ Adjacency (left), Laplacian (mid), theoretical guarantees (right)
  - $\Rightarrow$ Works very well in random graphs (also in real datasets)



Performance comparisons

- Comparison with graphical lasso and sparse correlation methods
  - Evaluated on 100 realizations of ER graphs with $N = 20$ and $p = 0.2$



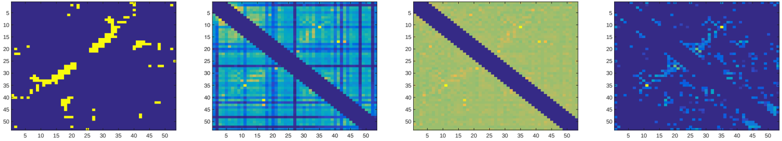
- Graphical lasso implicitly assumes a filter $\mathbf{H}_1 = (\rho\mathbf{I} + \mathbf{S})^{-1/2}$

  $\Rightarrow$ For this filter spectral templates work, but not as well

- For general diffusion filters $\mathbf{H}_2$ spectral templates still work fine

- Our method can be used to sparsify a given network
  - $\Rightarrow$ Keep direct and important edges or relations
  - $\Rightarrow$ Discard indirect relations that can be explained by direct ones
- Use eigenvectors $\hat{\mathbf{V}}$ of given network as noisy eigenvectors of $\mathbf{S}$

Ex: Infer contact between amino-acid residues in BPT1 BOVIN
  - $\Rightarrow$ Use mutual information of amino-acid covariation as input



Ground truth | Mutual info. | Network deconv. | Our approach

- Network deconvolution assumes a specific filter model [Feizi'13]
  - $\Rightarrow$ We achieve better performance by being agnostic to this

Stationarity-based generalizations

Stationarity as an overreaching model

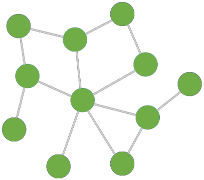
- Assuming $\mathbf{C} \to \mathbf{S}$ endows the problem with a flexible structure
  - $\Rightarrow$ It can be combined with smoothness (TV regularizer)
  - $\Rightarrow$ Graph regularizers for scenarios where # obs. P is limited

$$\max_{\boldsymbol{\Theta}\succeq\mathbf{0},\mathbf{S}} \left\{\log\det\boldsymbol{\Theta} - \mathrm{trace}(\hat{\mathbf{C}}\boldsymbol{\Theta}) - \lambda\|\mathbf{S}\|_1\right\} \quad \text{s. to} \;\; \mathbf{S}\boldsymbol{\Theta} = \boldsymbol{\Theta}\mathbf{S},\; \mathbf{S}\in\mathcal{S}$$

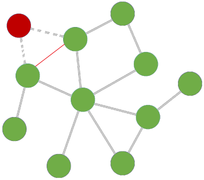
- Algorithms and theoretical results in a number of scenarios
  - $\Rightarrow$ Non-white inputs giving rise to $\mathbf{C}_x = \mathbf{H}(\mathbf{S})\mathbf{C}_w\mathbf{H}(\mathbf{S})$ [Shafipour'18]
  - $\Rightarrow$ Directed networks [Shafipour'18]
  - $\Rightarrow$ Online streaming signals [Shafipour'20]
  - $\Rightarrow$ Multi-relational graphs [Segarra'17,Navarro'20]
  - $\Rightarrow$ Hidden/latent nodes [Buciulea'19,Buciulea'21]

The case of hidden vars. (latent nodes)

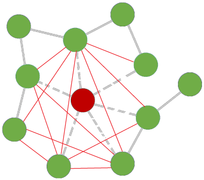
- In many relevant scenarios not all nodes are observed $N = o + h$
  - $\Rightarrow$ Can the $o \times o$ submatrix of $\mathbf{S}$ be recovered?
  - $\Rightarrow$ Can the full $N \times N$ matrix $\mathbf{S}$ be recovered (network tomography)?
  - $\Rightarrow$ How to modify the optimization?
  - $\Rightarrow$ How much does the recovery performance degrade?



$o = 11, \ h = 0$

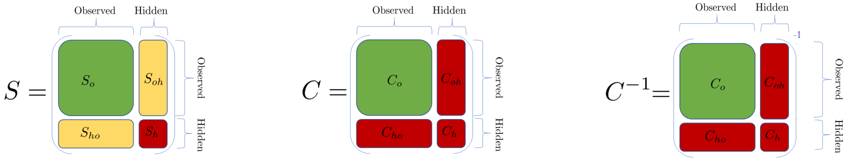


$o = 10, \ h = 1$



$o = 10, \ h = 1$

- Assume for simplicity observed nodes are the first h ones



- Correlation assume direct relation $\Rightarrow$ Trivial to generalize if hidden vars
  $\Rightarrow$ Find $\hat{\mathbf{C}}_o = \frac{1}{P}\mathbf{X}_o\mathbf{X}_o^T$, set $\hat{\mathbf{S}}_o = \hat{\mathbf{C}}_o \Rightarrow$ Network tomo not feasible

- Precision challenging [Chandrasekaran'12], key when $\mathbf{S} = \mathbf{C}^{-1}$:
  $\Rightarrow (\mathbf{C}_o)^{-1} = \mathbf{S}_o - \mathbf{R}$ with $\mathbf{R} := \mathbf{S}_{oh}(\mathbf{S}_h)^{-1}\mathbf{S}_{ho}$ having rank h

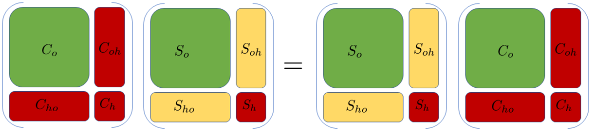
$$\hat{\mathbf{S}}_o = \arg \max_{\mathbf{S}_o - \mathbf{R} \succeq \mathbf{0},\, \mathbf{R} \succeq \mathbf{0}} \log\det(\mathbf{S}_o - \mathbf{R}) - \mathrm{trace}(\hat{\mathbf{C}}_o(\mathbf{S}_o - \mathbf{R})) - \lambda\|\mathbf{S}_o\|_1 + \alpha\|\mathbf{R}\|_*$$

Hidden vars: polynomial covariances

- Two approaches if fully observed, what if hidden nodes?
  - $\Rightarrow$ Estimation of eigenvectors at observed nodes very challenging
  - $\Rightarrow$ What about $\hat{\mathbf{C}}\mathbf{S} = \mathbf{S}\hat{\mathbf{C}}$?

Hidden vars: polynomial covariances

- Two approaches if fully observed, what if hidden nodes?
  - $\Rightarrow$ Estimation of eigenvectors at observed nodes very challenging
  - $\Rightarrow$ What about $\hat{\mathbf{C}}\mathbf{S} = \mathbf{S}\hat{\mathbf{C}}$?



$$\hat{\mathbf{C}}_o\mathbf{S}_o + \hat{\mathbf{C}}_{oh}\mathbf{S}_{ho} = \mathbf{S}_o\hat{\mathbf{C}}_o + \mathbf{S}_{oh}\hat{\mathbf{C}}_{ho}$$

- Leverage structure:

$$\mathrm{rank}(\hat{\mathbf{C}}_{oh}\mathbf{S}_{ho}) = h \ll o \qquad \hat{\mathbf{C}}_{oh}\mathbf{S}_{ho} = (\mathbf{S}_{oh}\hat{\mathbf{C}}_{ho})^T \qquad \|\mathbf{S}_{ho}\|_0 \ll ho$$

- Approach I: Convex relaxation

$$\mathbf{S}_o^* = \underset{\mathbf{R},\, \mathbf{S}_o \in \mathcal{S}_o}{\operatorname{argmin}} \ \|\mathbf{S}_o\|_1 + \eta \|\mathbf{R}\|_* \quad \text{s. to} \quad \hat{\mathbf{C}}_o \mathbf{S}_o + \mathbf{R} = \mathbf{S}_o \hat{\mathbf{C}}_o + \mathbf{R}^T$$

$\Rightarrow$ Re-weighted versions for ℓ_0 and nuclear norms are prudent

- Approach I: Convex relaxation

$$\mathbf{S}_o^* = \underset{\mathbf{R},\, \mathbf{S}_o \in \mathcal{S}_o}{\operatorname{argmin}} \; \|\mathbf{S}_o\|_1 + \eta \|\mathbf{R}\|_* \quad \text{s. to} \quad \hat{\mathbf{C}}_o \mathbf{S}_o + \mathbf{R} = \mathbf{S}_o \hat{\mathbf{C}}_o + \mathbf{R}^T$$

$\Rightarrow$ Re-weighted versions for ℓ_0 and nuclear norms are prudent

- Approach II: Additional structure, but convexity sacrificed

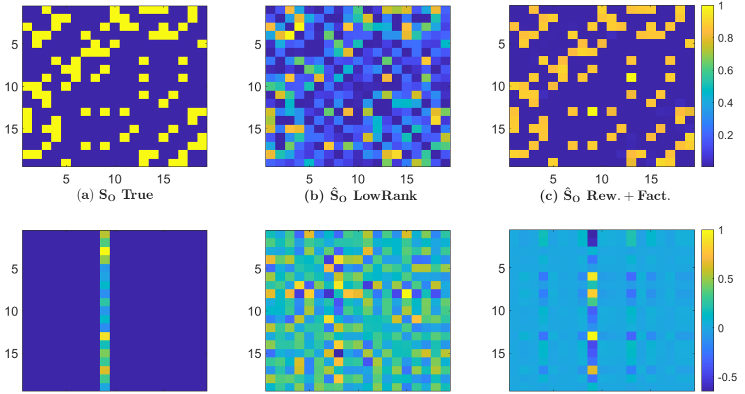
$$\mathbf{S}_o^* = \underset{\mathbf{C}_{oh} \in \mathcal{C}_{oh},\, \mathbf{S}_{oh} \in \mathcal{S}_{oh}\, \mathbf{S}_o \in \mathcal{S}_o}{\operatorname{argmin}} \; \|\mathbf{S}_o\|_1 + \alpha \|\mathbf{S}_{oh}\|_1$$
$$\text{s. to} \quad \hat{\mathbf{C}}_o \mathbf{S}_o + \hat{\mathbf{C}}_{oh} \mathbf{S}_{ho} = \mathbf{S}_o \hat{\mathbf{C}}_o + \mathbf{S}_{oh} \hat{\mathbf{C}}_{ho}$$

$\Rightarrow$ Alternating min, priors on $\mathbf{C}_{oh}$ and $\mathbf{S}_{oh}$ can be accommodated

$\Rightarrow$ $\mathbf{S}_{oh}$ as byproduct (network tomography)

Gaining insights

- Recovery with $N = 20$, $o = 19$, $h = 1$ for an ER graph



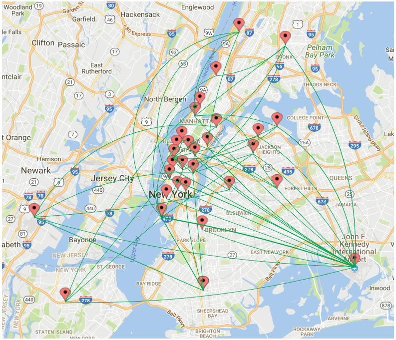
(a) $\mathbf{S}_{\mathrm{O}}$ **True** (b) $\hat{\mathbf{S}}_{\mathrm{O}}$ **LowRank** (c) $\hat{\mathbf{S}}_{\mathrm{O}}$ **Rew. + Fact.**

(d) $\mathbf{R}$ **True** (e) $\hat{\mathbf{R}}$ **LowRank** (f) $\hat{\mathbf{R}}$ **Rew. + Fact.**

$\Rightarrow$ Non-convex formulation does a better job unveiling structure

- What if h varies? Sensitivity to particular nodes?...

- Unveiling urban mobility patterns from Uber pickups in NYC
  - $\Rightarrow$ Times and locations: 1-1-15 to 6-29-15 and 263 locations ($N = 30$)
  - $\Rightarrow$ `https://github.com/fivethirtyeight/uber-tlc-foil-response`
- Input/output aggregated pickups 6am to 11am, 3pm to 8pm ($\mathbf{x}=\mathbf{H}\mathbf{w}$)
  - $\Rightarrow$ $M = 2$ graph processes: $m = 1$ weekday, $m = 2$ weekends



- Most edges connect Manhattan with the other boroughs $\Rightarrow$ Uber used to commute to/from suburbs
- Airports (Kennedy, Newark and LaGuardia) high degree nodes

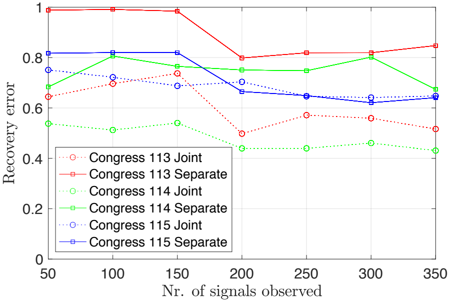
Joint Inference for US Senate Networks

- 2 US senators per state ($N = 50$) for 3 congresses (113th, 114th, 115th)
- Nodes are states, graph signals as congressional votes:

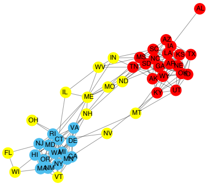
$$x_i = s_i^1 + s_i^2, \quad s_i^n = \begin{Bmatrix} 1, & yea \\ -1, & nay \\ 0, & ow \end{Bmatrix}$$

- **True 3 graphs:** Separately infer each graph with all available votes (657 for 113th, 502 for 114th, 599 for 115th)
- Compare separate and joint inference to true graphs for increasing number of randomly selected signals $n \in \{50, 100, \cdots, 350\}$ for ten trials of randomized subsets
- Joint inference assumes signals are stationary on each graph
  - $\Rightarrow$ Graphs $\{\mathbf{S}_1, \mathbf{S}_2, \mathbf{S}_3\}$ are relatively close
  - $\Rightarrow$ Promotes smoothness via $TV(\mathbf{X})$

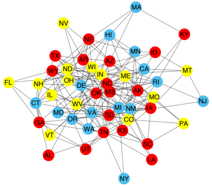
For most cases, joint inference outperforms separate inference:



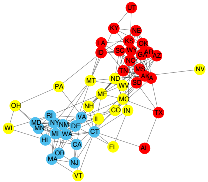
Joint Inference for US Senate Networks



True graph



Separately inferred



Jointly inferred

Closing remarks

- How to use the information in $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_P]$ to identify $G(\mathcal{V}, \mathcal{E})$
  - $\Rightarrow$ Focus on static and undirected graphs
  - $\Rightarrow$ GSP offers some novel insights and tools
- Focus of the talk
  - $\Rightarrow$ Links with classical methods, intuition and problem formulation
  - $\Rightarrow$ Not on algorithms and theoretical results (happy to discuss)
  - $\Rightarrow$ Polynomial mappings (i.e., stationarity) as a flexible model
- Emerging topic areas we did not cover
  - $\Rightarrow$ Network tomography
  - $\Rightarrow$ Directed graphs and causal structure identification
  - $\Rightarrow$ Dynamic networks and multi-layer graphs
  - $\Rightarrow$ Nonlinear models of interaction
  - $\Rightarrow$ Many excellent works we did not mention (cf. SPMag)!

Some open research directions

Universidad Rey Juan Carlos

- Relevance to applications: How to choose a graph learning method?
  - ⇒ Data itself as well as P, N, noise...
  - ⇒ Dependent also on the SP/ML task?
- Additional research directions
  - ⇒ Discrete and heterogeneous signals
  - ⇒ Tractable graph priors, Bayesian methods
  - ⇒ Non-homogeneous nodes

THANKS!

*Feel free to ask for the papers and/or the slides

[Segarra et al. "Network topology inference from spectral templates" IEEE TSIPN 2017.]