Graph Learning with Low-pass Graph Signal Processing

Hoang-Son Nguyen

Department of SEEM, The Chinese University of Hong Kong

Thanks: Prof. Hoi-To Wai for the slide materials

13 September 2024, Talk at ORLab-SLSCM Center

Our Group at CUHK

- We're from Department of Systems Engineering and Engineering Management at The Chinese University of Hong Kong.
- Our department focuses on financial engineering, information systems, logistics and supply chain management, and operations research.
- Beautiful campus by the sea, surrounded by lots of greens and hiking trails.

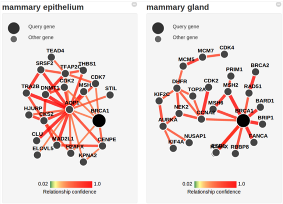
Our Group at CUHK

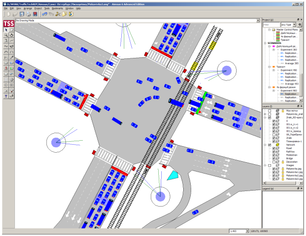
- Hong Kong is a vibrant global financial hub: a mix of Western and Chinese culture, country parks and skyscrapers; just under 2hr of flight from Hanoi.
- Lots of opportunities for funded postgraduate studies for non-local students.
- Come visit us sometime!

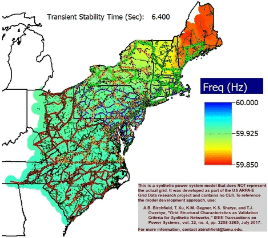
Our Group at CUHK

- Our team is advised by Prof. Hoi-To Wai, in Operations Research group.
- We focus on:
  - graph signal processing and graph learning problem for network science, and
  - stochastic and distributed methods for machine learning, signal processing.
- Today I'll talk about our recent works on graph signal processing.

Motivation: Network (Graph) Data



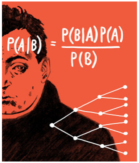




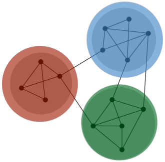
- Graph signal processing (GSP): tool to analyze **network data (graph signals)**.
- Data-generating processes affected by **network structure**: social, economic, biological, energy, transportation, etc.

Dealing with Network Data

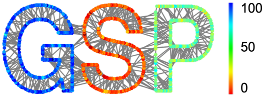
- Statistics: Gaussian Markov random fields, graphical models
  graph – statistical association of data



- Machine learning: dimensionality reduction
  graph – representation of data

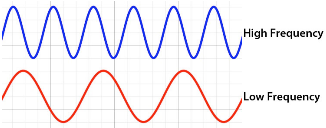


- **SP: Graph Signal Processing**
  graph – input/output association of data
  $\Longrightarrow$ generative, interpretable model

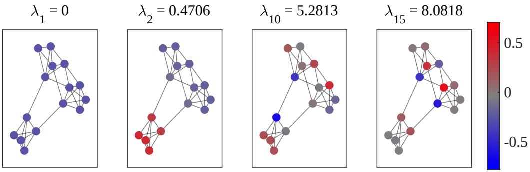


Low Pass GSP

- SP cares about the **frequency content** in a (time domain) signal — *low frequency* vs *high frequency*:



- Similar notion carries over to **graph signal processing (GSP)** — *low pass graph signals* vs *non low pass graph signals*:



Takehome: *Low pass* graph signals are prevalent + entail structure that enables (blind) *graph topology learning*.

Agenda

Agenda

Background

Basics of GSP Models

- A Quick Introduction
- Low Pass Graph Signals

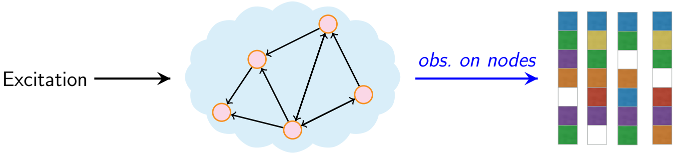
Graph Learning from Network Data

- Smoothness and Graph Learning
- Low-rank Model and Graph Feature Learning
- Learning with Partial Observation

Beyond Inference Problems & Wrapping Up

Graph Data (output) = Filter (system) + Excitation (input)

- Consider a *undirected graph* $G = (V, E, \boldsymbol{A})$ with N nodes



- Graph signals = vectors defined on V, *i.e.*, $\boldsymbol{x} \in \mathbb{R}^N$.

$$\text{excitation} \xrightarrow{\text{`filter'}} \text{signal}$$

†as in signal processing, filter encodes the **responses** of a system to excitation.

- We model the network dynamics generating the graph data by: *linear time invariant* (LTI) filter = 'shift-invariant' + 'linear combination'.

Graph Filters

- Network structure G is encoded in a matrix called *graph shift operator*
  - Common choice is Laplacian matrix $\boldsymbol{L} = \mathrm{Diag}(\boldsymbol{A}\mathbf{1}) - \boldsymbol{A}$
  - The EVD of $\boldsymbol{L}$ is $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\top$ with $0 = \lambda_1 < \cdots < \lambda_N$.
- Consider the **graph filter** as a matrix polynomial of $\boldsymbol{L}$:

$$\mathcal{H}(\boldsymbol{L}) := \sum_{\ell=0}^{+\infty} h_\ell \boldsymbol{L}^\ell.$$

<u>Shift-invariant prop:</u> $\boldsymbol{y} = \mathcal{H}(\boldsymbol{L})\boldsymbol{x} \quad \rightarrow \quad \boldsymbol{L}\boldsymbol{y} = \boldsymbol{L}\mathcal{H}(\boldsymbol{L})\boldsymbol{x} \equiv \mathcal{H}(\boldsymbol{L})\boldsymbol{L}\boldsymbol{x}$

- **GSP Perspective**: network data are **filtered** graph signals,

$$\underbrace{\boldsymbol{y}}_{output} = \underbrace{\mathcal{H}(\boldsymbol{L})}_{system} \underbrace{\boldsymbol{x}}_{input} = \sum_{\ell=0}^{+\infty} h_\ell \, \boldsymbol{L}^\ell \boldsymbol{x}.$$

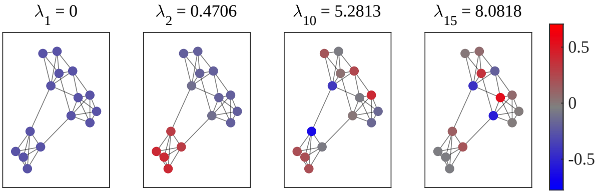
- The signal/observation is $\boldsymbol{y}$ while $\boldsymbol{x}$ is viewed as the **excitation**.

What are low and high frequencies basis on graph?

- High frequency graph signal $\to$ *large variation* in adjacent entries:

$$\mathrm{S}(\boldsymbol{x}) := \sum_{i,j} A_{ij}(x_i - x_j)^2 = \boldsymbol{x}^\top \boldsymbol{L}\boldsymbol{x}.$$

- Intuition: if $\mathrm{S}(\boldsymbol{x})$ is small, the graph signal $\boldsymbol{x}$ is *smooth*. It holds $\mathrm{S}(\boldsymbol{u}_i) = \boldsymbol{u}_i^\top \boldsymbol{L}\boldsymbol{u}_i = \lambda_i$, as seen:



$$\underbrace{\boldsymbol{u}_1}_{\text{lowest frequency}} \quad \cdots \longrightarrow \cdots \quad \underbrace{\boldsymbol{u}_N}_{\text{highest frequency}}$$

$\Longrightarrow \boldsymbol{U} = (\mathbf{u}_1 \ \mathbf{u}_2 \ \cdots \ \mathbf{u}_N)$ form the right basis for graph signals on G.

Frequency Analysis via Graph Fourier Transform

- **Graph Fourier Transform** gives the frequency components of a signal:

$$\tilde{\boldsymbol{y}} = \boldsymbol{U}^\top \boldsymbol{y} \longleftarrow \tilde{y}_i = \langle \boldsymbol{u}_i, \boldsymbol{y} \rangle.$$

- The transfer/frequency response function of the graph filter is:

$$\tilde{\boldsymbol{h}} = h(\boldsymbol{\lambda}) \quad \text{where} \quad \tilde{h}_i = h(\lambda_i) := \textstyle\sum_\ell h_\ell \lambda_i^\ell.$$

<u>Thus:</u> $\mathcal{H}(\boldsymbol{L}) = \boldsymbol{U} h(\boldsymbol{\Lambda}) \boldsymbol{U}^\top, \quad h(\boldsymbol{\Lambda}) = \mathrm{Diag}(h(\lambda_1), ..., h(\lambda_n)).$

- We have the convolution theorem:

$$\boldsymbol{y} = \mathcal{H}(\boldsymbol{L})\boldsymbol{x} \Longleftrightarrow \tilde{\boldsymbol{y}} = \tilde{\boldsymbol{h}} \odot \tilde{\boldsymbol{x}} \quad \leftarrow \odot \text{ is element-wise product.}$$

- Graph filter can be classified as either **low-pass**[1], band-pass, or high-pass, depending on its graph frequency response[2].

---

[1] E.g., an ideal low-pass $\tilde{h}_1, ..., \tilde{h}_K = 1$, $\tilde{h}_{K+1}, ..., \tilde{h}_N = 0$.

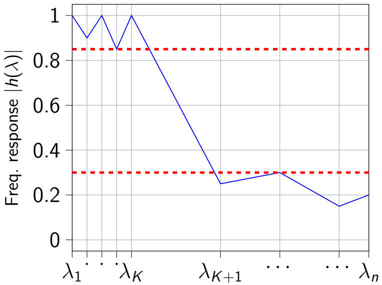
[2] [Isufi et al., 2024] E. Isufi, F. Gama, D. I Shuman, S. Segarra. Graph Filters for Signal Processing and Machine Learning on Graphs. ArXiv, 2022.

Low Pass Graph Filter (LPGF)

Def. For $1 \le K \le N-1$, define

$$\eta_K := \frac{\max\{|h(\lambda_{K+1})|, \ldots, |h(\lambda_N)|\}}{\min\{|h(\lambda_1)|, \ldots, |h(\lambda_K)|\}}.$$

If the low-pass ratio satisfies $\eta_K < 1$, then $\mathcal{H}(\boldsymbol{L})$ is ***K*-low-pass**.



- Integer ***K*** characterizes the *bandwidth*, or the cut-off frequency.
- We say that $\boldsymbol{y}$ is K **low pass signal** provided that

  $\boldsymbol{y} = \mathcal{H}(\boldsymbol{L})\boldsymbol{x}$, where $\mathcal{H}(\boldsymbol{L})$ is K-low pass & $\boldsymbol{x}$ satisfies some mild cond..
- Graph frequencies are **non-uniformly** distributed: $\lambda_K \ll \lambda_{K+1}$ tends to induce ***K***-low-pass filters, e.g., stochastic block model (SBM).

Physical Models lead to Low Pass Signals

Social Network Opinions[3]

- V = individuals, E = friends.
- DeGroot model for opinions:

$$\boldsymbol{y}_{t+1} = (1-\beta)\big(\boldsymbol{I} - \alpha \boldsymbol{L}\big)\boldsymbol{y}_t + \beta \boldsymbol{x}_t.$$

- **Observed** steady state:

$$\boldsymbol{y}_\infty = (\boldsymbol{I} + \widetilde{\alpha}\boldsymbol{L})^{-1}\boldsymbol{x} = \mathcal{H}(\boldsymbol{L})\boldsymbol{x},$$

where $\widetilde{\alpha} = \beta(1-\alpha)/\alpha > 0$.

Prices in Stock Market[4]

- V = financial inst., E = ties.
- Business performances evolve as:

$$\boldsymbol{y}_{t+1} = (1-\beta)\mathcal{H}(\boldsymbol{L})\boldsymbol{y}_t + \beta \boldsymbol{B}\boldsymbol{x},$$

e.g., stock return.

- **Observed** steady state:

$$\boldsymbol{y}_\infty = \left(\tfrac{1}{\beta}\boldsymbol{I} - \tfrac{\overline{\beta}}{\beta}\mathcal{H}(\boldsymbol{L})\right)^{-1}\boldsymbol{B}\boldsymbol{x} = \widetilde{\mathcal{H}}(\boldsymbol{L})\boldsymbol{B}\boldsymbol{x}.$$

Fact[5]: Both $\mathcal{H}(\boldsymbol{L})$, $\tilde{\mathcal{H}}(\boldsymbol{L})$ are **low pass** graph filters.

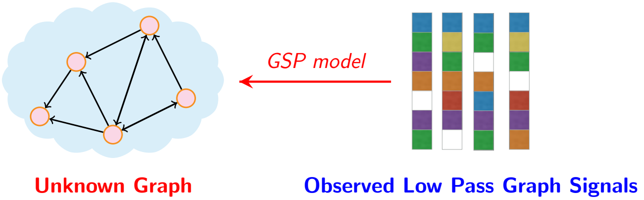
---

[3] [DeGroot, 1974] M. H. DeGroot, Reaching a consensus. JASA, 1974.
[4] [Billio et al., 2012] M. Billio et al., Econometric measures of connectedness and systemic risk in the finance and insurance sectors, Journal of Economics Finance, 2012.
[5] [Ramakrishna et al., 2020] R. Ramakrishna, **H.-T.**, A. Scalgione. A user guide to low-pass graph signal processing and its applications. SPM, 2020.

Agenda

Graph Learning from Network Data

- **Goal:** estimate $\boldsymbol{L}$ or some information about it.
- Working hypothesis: a number of graph signals $\boldsymbol{y}^{(t)}$ are available as



Observed graph signals: $\boldsymbol{y}^{(t)} \approx \mathcal{H}(\boldsymbol{L})\boldsymbol{x}^{(t)} = \mathcal{H}(\boldsymbol{L})\boldsymbol{B}\boldsymbol{z}^{(t)},\ t = 0, \ldots, T-1$

– $\mathcal{H}(\boldsymbol{L})$ is low pass, $\boldsymbol{z}^{(t)}$ is 0-mean, $\boldsymbol{B}$ is **pattern** of excitation

- Graph learning relies on **two properties** of low pass signals:
  - **Smoothness** $\rightarrow$ graph topology learning.
  - **Low-rankness** $\rightarrow$ graph feature learning (e.g., community, centrality)

Smoothness and Graph Learning

- **Insight**: For K-low-pass graph signals ($\eta_K \ll 1$) with **full-rank** excitation satisfying $\boldsymbol{B} = \boldsymbol{I}$, we observe that

$$\mathbb{E}\big[\boldsymbol{y}_\ell^\top \boldsymbol{L} \boldsymbol{y}_\ell\big] \approx \sum_{i=1}^{K} \lambda_i \, |h(\lambda_i)|^2 + \sigma^2 \mathrm{Tr}(\boldsymbol{L}) \overset{\text{low pass filter}}{\approx} 0,$$

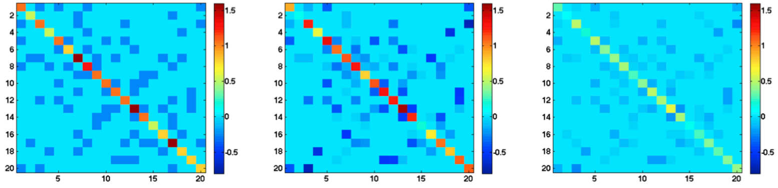
  i.e., the low pass filtered graph signals are *smooth* w.r.t. $\boldsymbol{L}$.

- **Idea**: Fit a graph optimizing for smoothness (GL-SigRep)[6]:

$$\begin{aligned} \min_{\boldsymbol{z}_\ell, \ell=1,\ldots,m, \widehat{\boldsymbol{L}}} \quad & \frac{1}{m} \sum_{\ell=1}^{m} \left\{ \frac{1}{\sigma^2} \|\boldsymbol{z}_\ell - \boldsymbol{y}_\ell\|_2^2 + \boldsymbol{z}_\ell^\top \widehat{\boldsymbol{L}} \boldsymbol{z}_\ell \right\} \leftarrow \text{note } \boldsymbol{z} \approx \boldsymbol{y} \\ \text{s.t.} \quad & \mathrm{Tr}(\widehat{\boldsymbol{L}}) = N, \; \widehat{L}_{ji} = \widehat{L}_{ij} \leq 0, \; \forall \, i \neq j, \; \widehat{\boldsymbol{L}}\boldsymbol{1} = \boldsymbol{0}, \end{aligned}$$

[6] [Dong et al., 2016] X. Dong, D. Thanou, P. Frossard, P. Vandergheynst, "Learning Laplacian matrix in smooth graph signal representations." TSP, 2016.

Numerical Experiment: GL-SigRep



(f) ER: Groundtruth (g) ER: **GL-SigRep** (h) ER: **GL-LogDet**

- Topology learnt[7] using GL-SigRep from the synthetic data generated through a low pass graph filter:

$$\boldsymbol{y}_\ell = \sqrt{\boldsymbol{L}}^{-1} \boldsymbol{x}_\ell, \quad \boldsymbol{x}_\ell \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}),$$

- Alternative approaches:
  - [Friedman et al., 2008] Graphical LASSO: ML estimation for GMRF.
  - [Segarra et al., 2017] Spectral template: stationary graph signals.
  - [Mei and Moura, 2016] Causal modeling: time series data

[7] Image credits: [Dong et al., 2016].

Low-rank-ness and Graph Feature Learning

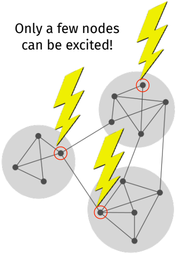
Issue: with low-rank excitation ($\boldsymbol{B} \in \mathbb{R}^{N \times R}$ with $R < N$) $\longrightarrow$ graph learning $=$ difficult $\because$ data is nearly rank deficient...

- **Insight**: Suppose $\mathcal{H}(\boldsymbol{L})$ is (η, K) **low pass**, then

  $$\boldsymbol{C}_y = \mathbb{E}[\boldsymbol{y}\boldsymbol{y}^\top] = \mathcal{H}(\boldsymbol{L})\boldsymbol{U}\boldsymbol{C}_x\boldsymbol{U}^\top\mathcal{H}(\boldsymbol{L})^\top \approx \boldsymbol{U}_K \boldsymbol{C}_{\tilde{x}} \boldsymbol{U}_K^\top \,,$$

  where $\boldsymbol{C}_x = \boldsymbol{B}\boldsymbol{B}^\top$, $\boldsymbol{U}_K = (\boldsymbol{u}_1, ..., \boldsymbol{u}_K) \in \mathbb{R}^{N \times K}$.
  $\Rightarrow$ Thus $\boldsymbol{C}_y$ is also **low rank**!
- *Approximation* holds if $\eta \ll 1 \Rightarrow$ **low rank** $\mathcal{H}(\cdot)$, $\mathrm{rank}(\mathcal{H}(\boldsymbol{L})) \approx K \ll N$ and range space $\approx \boldsymbol{U}_K$.
- **Idea:** spectral method to extract principal components in $\boldsymbol{U}_K$ from $\boldsymbol{C}_y$.

Only a few nodes can be excited!



$\Longrightarrow$ Can (still) learn **communities** and **centrality** of the graph.

Blind community detection (Blind CD)

Idea: spectral clustering applied on empirical covariance $\widehat{\boldsymbol{C}}_y \approx \boldsymbol{C}_y$:

(i) find the top-k $\widehat{\boldsymbol{U}}_K \in \mathbb{R}^{N \times K}$ of $\widehat{\boldsymbol{C}}_y = \frac{1}{m} \sum_{\ell=1}^{m} \boldsymbol{y}_\ell \boldsymbol{y}_\ell^\top$;

(ii) apply k-means on the rows of $\widehat{\boldsymbol{U}}_K$.

- **Theorem**: Denote the detected clusters as $\widehat{\mathcal{N}}_1, \ldots, \widehat{\mathcal{N}}_K$, then[8]

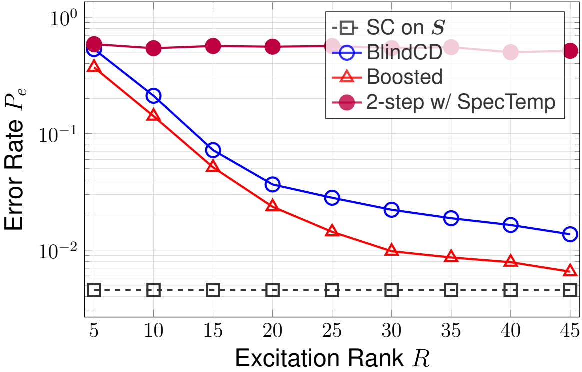
$$\underbrace{\mathbb{K}(\widehat{\mathcal{N}}_1, \ldots, \widehat{\mathcal{N}}_k; \boldsymbol{U}_K)}_{K\text{-means obj. based on } \boldsymbol{U}_K} - \underbrace{\mathbb{K}^\star}_{\text{Optimal } K\text{-means obj.}} = \mathcal{O}(\eta_k + m^{-1/2}).$$

† $\rightarrow$ performance of *spectral clustering (with known topology)* if $\eta_k \rightarrow 0$.

- Learning of high-level structure is **robust** to low-rank excitation.

[8][Wai et al., 2019] **H.-T.**, S. Segarra, A. Ozdaglar, A. Scaglione, A. Jadbabaie, "Blind community detection from low-rank excitations of a graph filter," TSP, 2019.

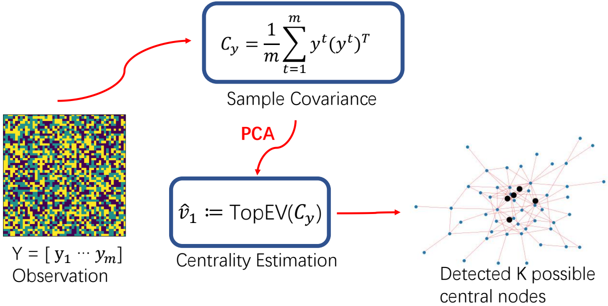
Numerical Experiment: Blind CD (+Boosting)



(a) As $R = \mathrm{rank}(\boldsymbol{C}_x)$ increases, Blind CD approaches the performance of spectral clustering on the true GSO.

Blind Centrality Learning

- Eigen-centrality = TopEV($\boldsymbol{A}$) is revealed by TopEV($\boldsymbol{C}_y$) for **1-low pass** signals $\Longrightarrow$ a simple PCA procedure suffices:

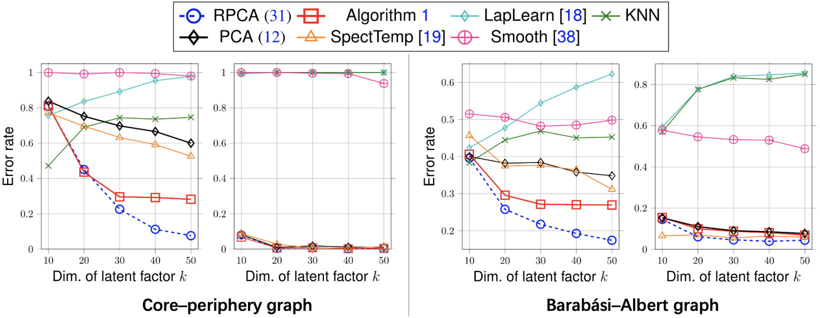


- **Theorem**[9]: let $\boldsymbol{v}_1$ be the true eig. centrality,

$$\|\hat{\boldsymbol{v}}_1 - \boldsymbol{u}_1\|_2 = \mathcal{O}(\eta_1 + m^{-1/2}).$$

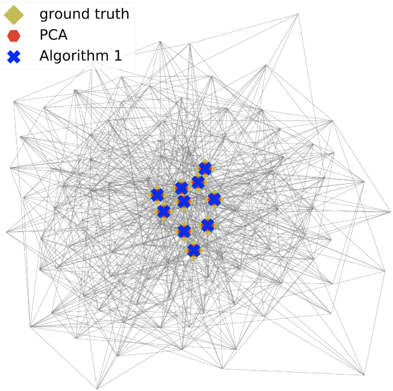
[9] [He and Wai, 2022] Y. He, **H.-T.**, "Detecting central nodes from low-rank excited graph signals via structured factor analysis," TSP, 2022 $\leftarrow$ note GSO = $\boldsymbol{A}$ in this case.

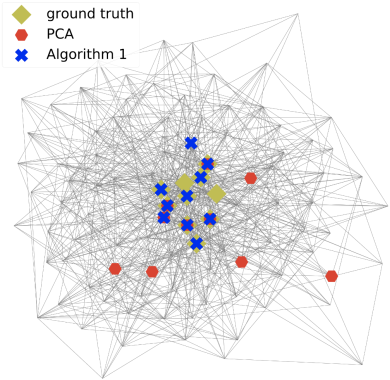
Numerical Experiment: Blind Centrality Learning



- Graph filter $\mathcal{H}(\cdot)$ is (left) 'weak' low pass, i.e., $\eta \approx 1$; (right) 'strong' low pass, i.e., $\eta \ll 1$.
- Proposed **Algorithm 1** with NMF outperforms SOTA in the considered setting for 'weak' low pass; and similarly as PCA for 'strong' low pass.

Numerical Experiment: Blind Centrality Learning





(left) 'Strong' low pass, (right) 'Weak' low pass

Numerical Experiment: Blind Centrality Learning

(a) Stock Dataset†

Method	Top-10 Estimated Central Stocks (sorted left-to-right)									
Algorithm 1	ALL 0.43	ACN 0.56	HON 0.51	AXP 0.72	IBM 0.50	DIS 0.36	ORCL 0.70	MMM 0.33	BRK.B 0.52	COST 0.64
	Average Correlation Score: 0.53 ± 0.133									
PCA (11)	NVDA 0.56	NFLX 0.60	AMZN 0.68	ADBE 0.63	PYPL 0.65	CAT 0.27	MA 0.67	GOOG 0.63	BA 0.28	GOOGL 0.63
	Average Correlation Score: 0.56 ± 0.154									
GL-SigRep [13]	GOOGL 0.63	GOOG 0.63	LLY 0.17	USB 0.43	EMR 0.59	DUK 0.11	ORCL 0.70	GD 0.53	VZ 0.27	V 0.71
	Average Correlation Score: 0.48 ± 0.22									
KNN	ACN 0.56	HON 0.51	ALL 0.43	BRK.B 0.52	IBM 0.50	AXP 0.72	EMR 0.59	MMM 0.33	CSCO 0.63	XOM 0.55
	Average Correlation Score: 0.53 ± 0.107									
SpecTemp [14]	ACN 0.56	ORCL 0.70	PG 0.36	LLY 0.17	SUBX 0.58	PYPL 0.65	MDLZ 0.41	FB 0.61	PFE 0.14	MRK 0.20
	Average Correlation Score: 0.44 ± 0.211									
Kalofolias [44]	ACN 0.56	HON 0.51	BRK.B 0.52	ALL 0.43	AXP 0.72	IBM 0.50	XOM 0.55	KO 0.32	USB 0.43	COST 0.64
	Average Correlation Score: 0.52 ± 0.112									

Information Technology/ Communication Services/ Industrials/ Financials/other sectors.

(b) Senate Dataset†

Method	Top-10 Estimated Central States (sorted left-to-right)									
Algorithm 1	MI 0.79	MT 0.66	KS 0.74	RI 0.67	TN 0.68	MN 0.74	NV 0.43	ME 0.67	MD 0.6	IN 0.62
	Average Correlation Score: 0.66 ± 0.099									
PCA (11)	CA 0.55	DE 0.46	CO 0.54	IL 0.63	ND 0.72	WV 0.52	IA 0.51	VA 0.56	WY 0.59	MA 0.58
	Average Correlation Score: 0.57 ± 0.072									
GL-SigRep [13]	CA 0.55	DE 0.46	WV 0.52	CO 0.54	IL 0.63	VA 0.56	ND 0.72	IA 0.51	WY 0.59	AZ 0.31
	Average Correlation Score: 0.54 ± 0.108									
KNN	ND 0.72	CA 0.55	IL 0.63	WV 0.52	DE 0.46	VA 0.56	AZ 0.31	CO 0.54	WY 0.59	IA 0.51
	Average Correlation Score: 0.54 ± 0.108									
SpecTemp [14]	AL 0.61	ND 0.72	WV 0.52	CA 0.55	DE 0.46	IL 0.63	MO 0.57	MA 0.58	VA 0.56	SD 0.56
	Average Correlation Score: 0.58 ± 0.069									
Kalofolias [44]	AL 0.61	AK 0.63	AZ 0.31	AR 0.47	WV 0.52	VA 0.56	CA 0.55	CO 0.54	CT 0.45	DE 0.46
	Average Correlation Score: 0.51 ± 0.093									

Republican/ Democrat/ Mixed.

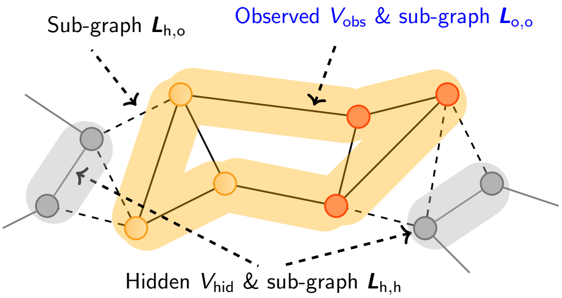
[4]†The number below each stock/state shows its normalized correlation score with the S&P100 index and number of 'Yay's in the voting result [cf. (36)]. The average correlation scores are taken over the set of central nodes found and the number after '±' is the standard deviation.

(a) Detected central nodes with performance measured on correlation of nodes with (left) S&P500 index, (right) voting outcomes.

Leveraging Low-passness with Partial Observation

- In many settings, we do not observe **complete graph signals** on every nodes, e.g., large social network, power network, etc.
- Hidden nodes remain **influential** and affect the observations:

$$\boldsymbol{y} = \mathcal{H}(\boldsymbol{L})\boldsymbol{x} \quad \text{with} \quad \boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_{\rm obs} \\ \boldsymbol{y}_{\rm hid} \end{bmatrix}, \ \boldsymbol{L} = \begin{bmatrix} \boldsymbol{L}_{\rm o,o} & \boldsymbol{L}_{\rm o,h} \\ \boldsymbol{L}_{\rm h,o} & \boldsymbol{L}_{\rm h,h} \end{bmatrix}$$



Learning with Partial Observation

- **Goal**: infer about **the subgraph of observable nodes**, $\boldsymbol{L}_{\mathrm{o,o}}$:

$$\boldsymbol{y} = \mathcal{H}(\boldsymbol{L})\boldsymbol{x} = \begin{bmatrix} \boldsymbol{y}_{\mathrm{obs}} \\ \boldsymbol{y}_{\mathrm{hid}} \end{bmatrix},\ \boldsymbol{C}_y = \begin{bmatrix} \boldsymbol{C}_y^{\mathrm{o}} & \boldsymbol{C}_y^{\mathrm{o,h}} \\ \boldsymbol{C}_y^{\mathrm{h,o}} & \boldsymbol{C}_y^{\mathrm{h}} \end{bmatrix},\ \boldsymbol{L} = \begin{bmatrix} \boxed{\boldsymbol{L}_{\mathrm{o,o}}} & \boldsymbol{L}_{\mathrm{o,h}} \\ \boldsymbol{L}_{\mathrm{h,o}} & \boldsymbol{L}_{\mathrm{h,h}} \end{bmatrix}$$

Leveraging Lowrank-ness: provided $\mathcal{H}(\boldsymbol{L})$ is (η, K) low pass,

$$\boldsymbol{C}_y^o = \boldsymbol{E}_o \boldsymbol{C}_y \boldsymbol{E}_o^\top \approx (\boldsymbol{E}_o \boldsymbol{U}_K) \boldsymbol{C}_{\tilde{x}} (\boldsymbol{E}_o \boldsymbol{U}_K)^\top$$

where $\boldsymbol{E}_o$ is **row-selection** matrix for V_{obs}. $\uparrow$ can estimate $\boldsymbol{E}_o \boldsymbol{U}_K \approx \boldsymbol{U}_{K,o}$

- **Key observation**: low-rankness of $\mathcal{H}(\boldsymbol{L})$ **supersedes** partial obs.
- Straightforward extension for graph feature learning: partial community detection[10], partial centrality inference[11]

[10][Wai et al., 2022] **H.-T.**, Y. Eldar, A. Ozdaglar, A. Scaglione, "Community Inference From Partially Observed Graph Signals: Algorithms and Analysis", TSP, 2022.

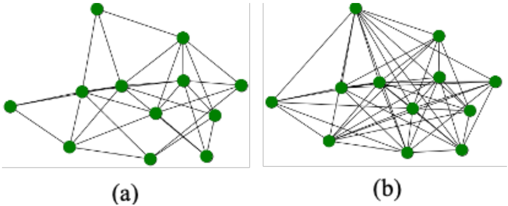
[11][He and Wai, 2023] Y. He, **H.-T.**, Central nodes detection from partially observed graph signals, in ICASSP 2023.

Agenda

Detecting Low-pass Signals

Question: How do we know if a set of graph signals are low pass?

- Topology inferred from non low pass signals can be **deceptive**.

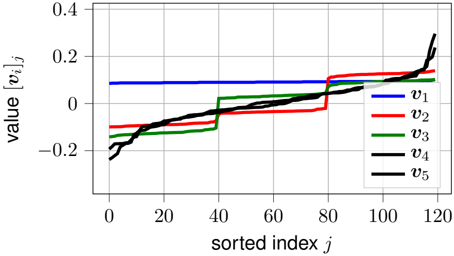


(a) Ground truth. (b) Topology learnt by GL-SigRep on **non-low-pass** signals.

- *Challenges*: graph topology $\boldsymbol{A}$ and filter $\mathcal{H}(\boldsymbol{A})$ are **unknown**.
- Warning: an **ill posed** problem – graph signals is *smooth* on one graph, but *non-smooth* on another.

Detecting Low-pass Signals

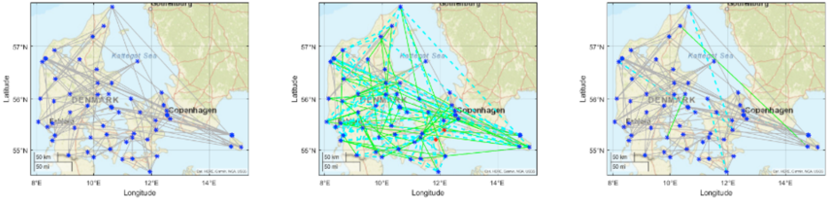
- **Assume**: no. of dense clusters, K, in the graph is known a-priori.
  $\Longrightarrow \lambda_1, \ldots, \lambda_K \approx 0 \Longrightarrow$ if the filter is low pass, it will be K low pass.
- **Observation**: graph signals from K low pass filter exhibit particular *spectral signature*. E.g., SBM graph with $K = 3$ clusters,



Idea: Measure *clusterability* of principal eigenvectors.

Application: Robustifying Graph Learning

What if graph signals are corrupted with non-low-pass observations? $\Longrightarrow$ **screen them out** by a blind detector and apply [Dong et al., 2016].



(a) (b) (c)

(a) Ground truth graph learnt from clean data.
(b) Graph learnt from **corrupted** data (mixed w/ high-pass signals).
(c) Graph learnt after the **pre-screening** procedure.

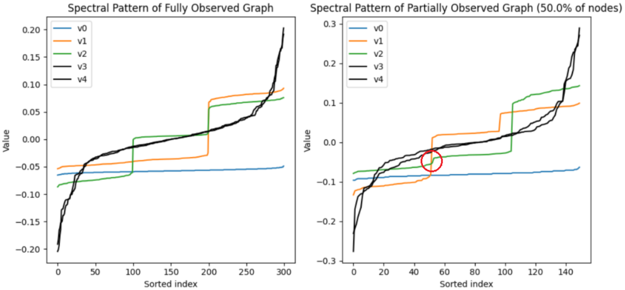
- **Other applications:** blind detection of network dynamics, blind anomaly detection, etc.[12]

[12] [Zhang et al., 2023] C. Zhang, Y. He, **H.-T.**. Detecting Low Pass Graph Signals via Spectral Pattern: Sampling Complexity and Applications. ArXiv, 2023.

Detecting Low-pass Signals w/ Partial Observations

$$\boldsymbol{y} = \mathcal{H}(\boldsymbol{L})\boldsymbol{x} = \begin{bmatrix} \boldsymbol{y}_{\text{obs}} \\ \boldsymbol{y}_{\text{hid}} \end{bmatrix}, \; \boldsymbol{C}_y = \begin{bmatrix} \boldsymbol{C}_y^{\text{o}} & \boldsymbol{C}_y^{\text{o,h}} \\ \boldsymbol{C}_y^{\text{h,o}} & \boldsymbol{C}_y^{\text{h}} \end{bmatrix}, \; \boldsymbol{L} = \begin{bmatrix} \boxed{\boldsymbol{L}_{\text{o,o}}} & \boldsymbol{L}_{\text{o,h}} \\ \boldsymbol{L}_{\text{h,o}} & \boldsymbol{L}_{\text{h,h}} \end{bmatrix}$$

- **Observation**: the *spectral signature* is preserved even in partially observed low-pass graph signals, E.g., SBM graph with $K = 3$ clusters,



Measuring *clusterability* of principal eigenvectors **still works**.

Application: Robustifying Partial Blind CD

What if partial graph signals are corrupted with non-low-pass observations?
$\Longrightarrow$ **screen them out** by a blind detector and apply [Wai et al., 2022].

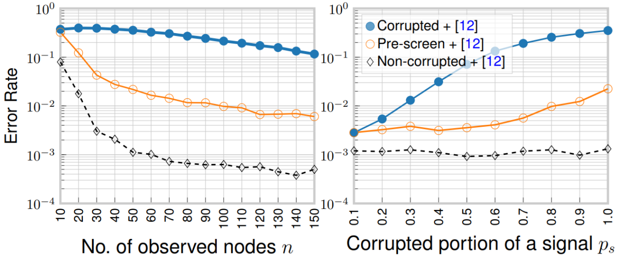


Fig. 3. Comparing blind community detection performance vs. (left) no. of observed nodes n ($p_s = 1$), (right) corrupted portion of signals p_s ($n = 50$).

- **Other applications:** blind detection of network dynamics, blind anomaly detection, etc. with only partial observations[13]

[13][Nguyen and Wai, 2024] **H.-S.**, **H.-T.**, “On Detecting Low-pass Graph Signals under Partial Observations”, in SAM, 2024.

Stability of Graph Filter with Edge Rewiring

- Graph filter is an important building block of *Graph Convolutional Neural Network (GCN)* $\rightarrow$ trained on $\mathcal{H}(\boldsymbol{L})$, but applied on $\mathcal{H}(\hat{\boldsymbol{L}})$.
- **Stability**[14] is related to *transferability* of GCNs. Existing results require small no. of edge rewires.

Frequency-domain bound: If $\mathcal{H}(\boldsymbol{L})$ is **low pass**, then

$$\|\mathcal{H}(\boldsymbol{L}) - \mathcal{H}(\hat{\boldsymbol{L}})\| = \mathcal{O}(\eta + \|\boldsymbol{U}_k - \hat{\boldsymbol{U}}_k\| + \|\boldsymbol{\Lambda}_k - \hat{\boldsymbol{\Lambda}}_k\|),$$

where $\boldsymbol{U}_k - \hat{\boldsymbol{U}}_k$, $\boldsymbol{\Lambda}_k - \hat{\boldsymbol{\Lambda}}_k$ are perturbations of top eigenvectors/values.

- Residuals $\rightarrow 0$ for edge rewiring on SBMs perturbations[15].

[14] [Gama et al., 2020] F. Gama, J. Bruna, A. Ribeiro. Stability properties of graph neural networks. TSP, 2020.

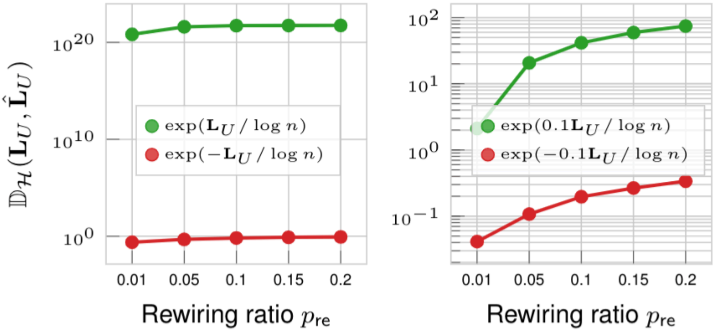
[15] [Nguyen et al., 2022] **H.-S.**, Y. He, **H.-T.**, "On the stability of low pass graph filter with a large number of edge rewires," in ICASSP, 2022.

Stability of Graph Filter with Edge Rewiring

Frequency-domain bound: If $\mathcal{H}(\boldsymbol{L})$ is **low pass**, then

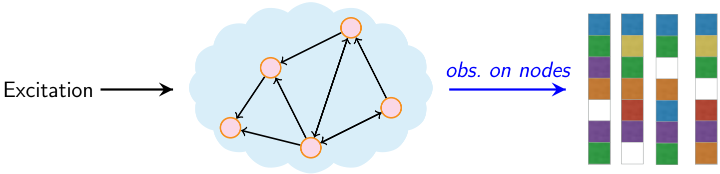
$$\|\mathcal{H}(\boldsymbol{L}) - \mathcal{H}(\hat{\boldsymbol{L}})\| = \mathcal{O}(\eta + \|\boldsymbol{U}_k - \hat{\boldsymbol{U}}_k\| + \|\boldsymbol{\Lambda}_k - \hat{\boldsymbol{\Lambda}}_k\|),$$

where $\boldsymbol{U}_k - \hat{\boldsymbol{U}}_k$, $\boldsymbol{\Lambda}_k - \hat{\boldsymbol{\Lambda}}_k$ are perturbations of top eigenvectors/values.



- Low pass filters are *insensitive* to no. of rewiring vs. high pass filters.

Wrapping Up



- **Takehome Point**: *Low pass* graph signals are prevalent + entail structure that enables (blind) *graph topology learning*.
  - **Smoothness** → graph topology learning.
  - **Low-rankness** → topology **feature** learning (centrality, community).
  - also for learning from partial observation, ...
- Related problems: how to detect low pass signals, application to machine learning on graph, ...

Thank you!

Questions & comments are welcomed.

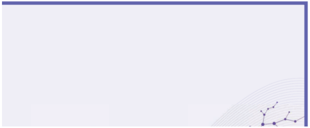
An (old) tutorial can be found here: arxiv.org/abs/2008.01305

**GRAPH SIGNAL PROCESSING:
FOUNDATIONS AND EMERGING DIRECTIONS**

Raksha Ramakrishna, Hoi-To Wai, and Anna Scaglione

A User Guide to Low-Pass Graph Signal Processing and Its Applications

Tools and applications



The notion of graph filters can be used to define generative models for graph data. In fact, the data obtained from many examples of network dynamics may be viewed as the output of a graph filter. With this interpretation, classical signal processing tools, such as frequency analysis, have been successfully applied with analogous interpretation to graph data, generating new insights for data science. What follows is a user guide on a specific class of graph data, where the generating graph filters are low pass; i.e., the filter attenuates contents in the higher

References

[Billio et al., 2012] Billio, M., Getmansky, M., Lo, A. W., and Pelizzon, L. (2012).
Econometric measures of connectedness and systemic risk in the finance and insurance sectors.
Journal of financial economics, 104(3):535–559.

[DeGroot, 1974] DeGroot, M. H. (1974).
Reaching a consensus.
Journal of the American Statistical association, 69(345):118–121.

[Dong et al., 2016] Dong, X., Thanou, D., Frossard, P., and Vandergheynst, P. (2016).
Learning Laplacian matrix in smooth graph signal representations.
IEEE Trans. Signal Process., 64(23):6160–6173.

[Friedman et al., 2008] Friedman, J., Hastie, T., and Tibshirani, R. (2008).
Sparse inverse covariance estimation with the graphical lasso.
Biostatistics, 9(3):432–441.

[Gama et al., 2020] Gama, F., Bruna, J., and Ribeiro, A. (2020).
Stability properties of graph neural networks.
IEEE Transactions on Signal Processing, 68:5680–5695.

[He and Wai, 2022] He, Y. and Wai, H.-T. (2022).
Detecting central nodes from low-rank excited graph signals via structured factor analysis.
IEEE Transactions on Signal Processing.

[He and Wai, 2023] He, Y. and Wai, H.-T. (2023).
Central nodes detection from partially observed graph signals.
In *ICASSP*. IEEE.

[Isufi et al., 2024] Isufi, E., Gama, F., Shuman, D. I., and Segarra, S. (2024).
Graph filters for signal processing and machine learning on graphs.
IEEE Transactions on Signal Processing, pages 1–32.

[Mei and Moura, 2016] Mei, J. and Moura, J. M. (2016).
Signal processing on graphs: Causal modeling of unstructured data.
IEEE Transactions on Signal Processing, 65(8):2077–2092.

[Nguyen et al., 2022] Nguyen, H.-S., He, Y., and Wai, H.-T. (2022).
On the stability of low pass graph filter with a large number of edge rewires.
In *IEEE ICASSP*.

[Nguyen and Wai, 2024] Nguyen, H.-S. and Wai, H.-T. (2024).
On detecting low-pass graph signals under partial observations.
In *2024 IEEE 13rd Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pages 1–5.

[Ramakrishna et al., 2020] Ramakrishna, R., Wai, H.-T., and Scaglione, A. (2020).
A user guide to low-pass graph signal processing and its applications.
IEEE Signal Processing Magazine.

[Segarra et al., 2017] Segarra, S., Marques, A. G., Mateos, G., and Ribeiro, A. (2017).
Network topology inference from spectral templates.
IEEE Trans. Signal and Info. Process. over Networks, 3(3):467–483.

[Wai et al., 2022] Wai, H.-T., Eldar, Y. C., Ozdaglar, A. E., and Scaglione, A. (2022).
Community inference from partially observed graph signals: Algorithms and analysis.
IEEE Transactions on Signal Processing, 70:2136–2151.

[Wai et al., 2019] Wai, H.-T., Segarra, S., Ozdaglar, A. E., Scaglione, A., and Jadbabaie, A. (2019).
Blind community detection from low-rank excitations of a graph filter.
IEEE Transactions on signal processing, 68:436–451.

[Zhang et al., 2023] Zhang, C., He, Y., and Wai, H.-T. (2023).

Detecting low pass graph signals via spectral pattern: Sampling complexity and applications.
ArXiv.