Low Pass Graph Signal Processing: Modeling Data, Inference, and Beyond

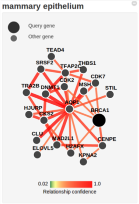
Hoi-To Wai

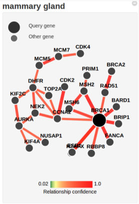
Department of SEEM, The Chinese University of Hong Kong

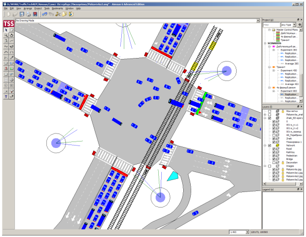
June 12, 2023, GSP Workshop 2023

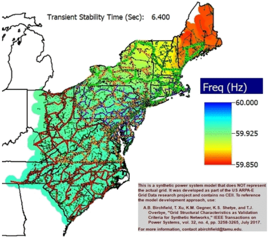
Acknowledgement: CUHK Direct Grant #4055135, HKRGC Project #24203520

Motivation: Network (Graph) Data





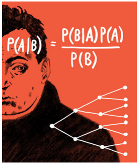




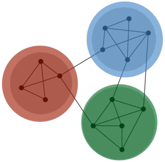
- Graph signal processing (GSP): tool to analyze **network data (graph signals)**.
- Processes affected by **irregular+relational** parameters: social, economic, biological, electric power, transportation, gas, etc.

Dealing with Network Data

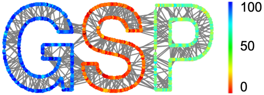
- Statistics: Gauss Markov random fields, graphical models
  — statistical association of data



- Machine learning: dimensionality reduction
  — graph representation of data

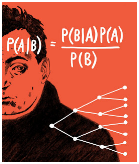


- **SP: Graph Signal Processing**
  — input/output association of data
  $\Longrightarrow$ generative, interpretable model

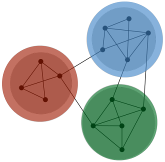


Dealing with Network Data

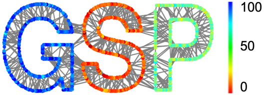
- Statistics: Gauss Markov random fields, graphical models
  — *statistical association of data*



- Machine learning: dimensionality reduction
  — *graph representation of data*

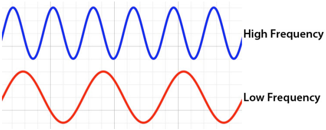


- **SP: Graph Signal Processing**
  — *input/output association of data*
  $\Longrightarrow$ *generative, interpretable model*

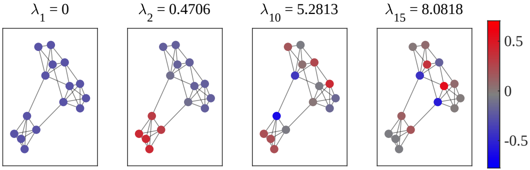


Low Pass GSP

- 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*:



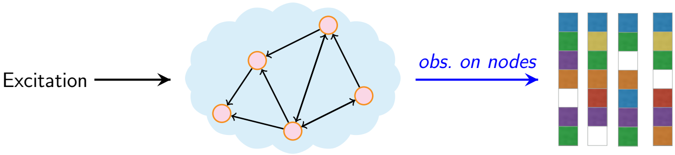
Takehome Point: *Low pass* graph signals are prevalent + entail structure that enables (blind) *graph topology learning*.

Agenda

Agenda

Network Data = Filter + Excitation

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



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

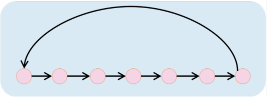
$$\text{excitation} \xrightarrow{\text{‘filter’}} \text{signal}$$

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

- As SP-ers, what is our favorite form of *filter*?
- *Linear time invariant* filter = ‘shifting’ + ‘linear combination’.

Graph Shift Operator (GSO)

- Starting point: Periodic signals $\boldsymbol{x} = (x_1, \ldots, x_N)$ is 'shifted' on a cycle



$$\rightarrow \quad \boldsymbol{A} = \begin{pmatrix} 0 & 0 & 0 & \ldots & 0 & 1 \\ 1 & 0 & 0 & \ldots & 0 & 0 \\ 0 & 1 & 0 & \ldots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \ddots & 1 & 0 & 0 \\ 0 & 0 & \ldots & 0 & 1 & 0 \end{pmatrix}$$

$$\boldsymbol{A} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} = \begin{pmatrix} x_N \\ x_1 \\ \vdots \\ x_{N-1} \end{pmatrix} \Longleftrightarrow \text{Applying } \boldsymbol{A} \text{ is analogous to shifting the signal}$$

- **Generalization to graphs**: GSO mixes adjacent elements on G[1].
- Common choice of GSO: Laplacian matrix, $\boldsymbol{L} = \mathrm{Diag}(\boldsymbol{A}\mathbf{1}) - \boldsymbol{A}$.

  — for the rest of the talk, we focus on **undirected** graph.
- Denote the EVD $\boldsymbol{L} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\top$ with $0 = \lambda_1 < \cdots < \lambda_N$.

[1][Sandryhaila and Moura, 2013] A. Sandryhaila, J. M. Moura. Discrete signal processing on graphs. TSP, 2013; also see [Püschel and Moura, 2003].

Graph Filters

- Consider the **graph filter** as a matrix polynomial:

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

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

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

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

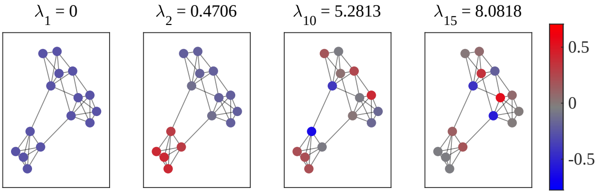
- The signal/observation is $\boldsymbol{y}$ while $\boldsymbol{x}$ is viewed as the **excitation**.

What are low and high frequencies basis on graph?

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

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

- 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 (GFT) calculates 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:

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

- 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**[2], band-pass, or high-pass, depending on its graph frequency response, also see[3].

---

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

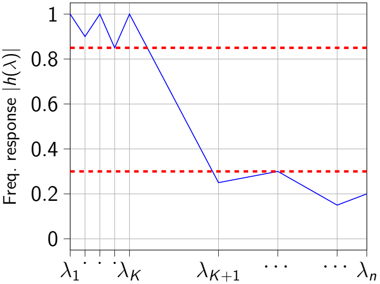
[3] [Isufi et al., 2022] E. Isufi, F. Gama, D. I Shuman, S. Segarra. Graph Filters for Signal Processing and Machine Learning on Graphs. ArXiv, 2022.

Low Pass Graph Filter (LPGF)

Def. For $1 \leq K \leq N-1$, 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

  $$\boldsymbol{y} = \mathcal{H}(\boldsymbol{L})\boldsymbol{x}, \quad \text{where } \mathcal{H}(\boldsymbol{L}) \text{ is } K\text{-low pass \& } \boldsymbol{x} \text{ satisfies some mild cond..}$$

- 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[4]

- 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[5]

- 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:

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

Fact[6]: Both $\mathcal{H}(\boldsymbol{L})$, $\tilde{\mathcal{H}}(\boldsymbol{L})$ are **low pass** graph filters.

[4] [DeGroot, 1974] M. H. DeGroot, Reaching a consensus. JASA, 1974.

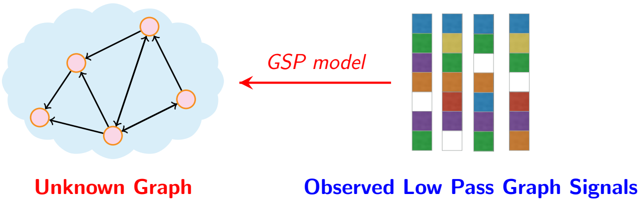
[5] [Billio et al., 2012] M. Billio et al., Econometric measures of connectedness and systemic risk in the finance and insurance sectors, Journal of Economics Finance, 2012.

[6] [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: $\boldsymbol{y}^{(t)} \approx \mathcal{H}(\boldsymbol{L})\boldsymbol{B}\boldsymbol{z}^{(t)}, \; t = 0, ..., T-1$

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

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

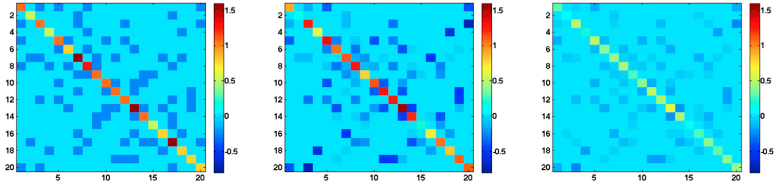
  i.e., the low pass filtered graph signals are *smooth* w.r.t. $\boldsymbol{L}$.

- **Idea**: Fit a graph optimizing for smoothness (GL-SigRep)[7]:

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

[7] [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[8] 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

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

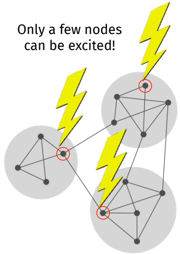
Low-rank-ness and Graph Feature Learning

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

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

  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 $\boldsymbol{C}_y$.



$\Longrightarrow$ 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[9]

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

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

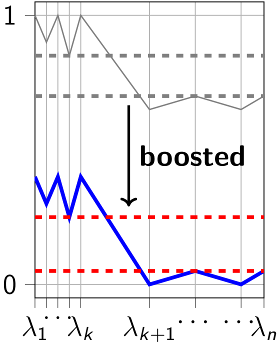
- Learning of high-level structure is **robust** to low-rank excitation.
- **Extensions**: exact community recovery on multi-graphs [Roddenberry et al., 2020], dynamic observations [Schaub et al., 2020], ...

[9] [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.

Blind community detection (Blind CD)

Problem: What if $\eta_K \approx 1$? Let's try $\underbrace{\widetilde{\mathcal{H}}_\rho(\boldsymbol{L}) := \mathcal{H}(\boldsymbol{L}) - \rho \boldsymbol{I}}_{\text{'boosted' filter}}$ $(\rho > 0)$.

Freq. response



- Original ratio: $\eta_K = \frac{0.7}{0.85} \approx 0.82$.
- **Boosted** ratio: $\tilde{\eta}_K = \frac{0.05}{0.25} = 0.2$.

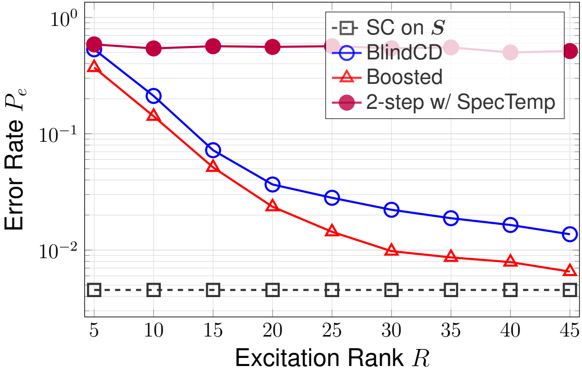
Suppose that $\boldsymbol{Z}$ is known,

$$\boldsymbol{YZ}^\dagger = \mathcal{H}(\boldsymbol{L})\boldsymbol{B} = \underbrace{\widetilde{\mathcal{H}}_\rho(\boldsymbol{L})\boldsymbol{B}}_{\text{low-rank}} + \rho \boldsymbol{B}$$

- Typically, $\boldsymbol{B}$ is sparse
  $\Longrightarrow$ low-rank + sparse
  decomposition!

Robust PCA formulation: $\min_{\boldsymbol{L},\boldsymbol{B}} \; \|\boldsymbol{YZ}^\dagger - \boldsymbol{L} - \boldsymbol{B}\|_F^2 + \gamma \|\boldsymbol{L}\|_\star + \mu \|\boldsymbol{B}\|_1$

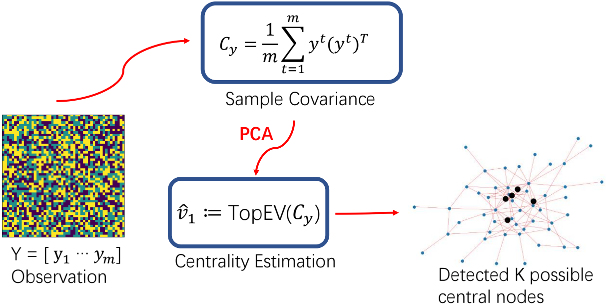
Numerical Experiment: Blind CD (+Boosting)



(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**[10]: let $\boldsymbol{v}_1$ be the true eig. centrality,

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

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

Blind Centrality Learning (cont'd)

- To obtain a robust formulation against $\eta_1 \approx 1$, assume that $\boldsymbol{B}$ is *sparse* and use similar idea as Blind CD:

$$\boldsymbol{Y} = \mathcal{H}(\boldsymbol{A})\boldsymbol{B}\boldsymbol{Z} = \big((\mathcal{H}(\boldsymbol{A}) - \rho \boldsymbol{I})\boldsymbol{B} + \rho \boldsymbol{B}\big)\boldsymbol{Z}$$
$$= (\text{Low-rank} + \text{Sparse}) \times \boldsymbol{Z}$$

- **Structured factor analysis**: if $\boldsymbol{Z}$ is *unknown*,

  Step 1. decompose $\boldsymbol{Y}$ via NMF, Step 2. Robust PCA.

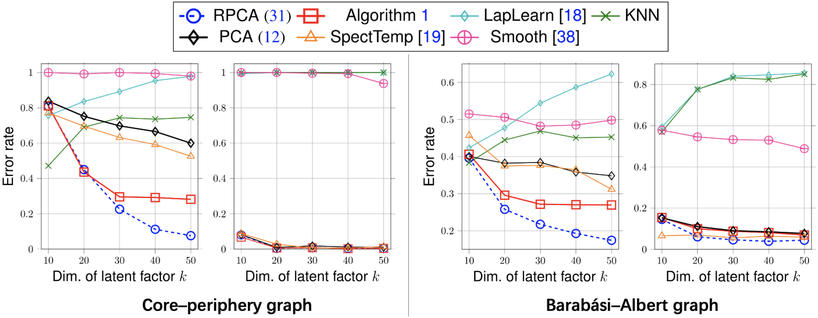
- **Theoretical analysis** (for NMF): good performance if (i) $N/\mathrm{rank}(\boldsymbol{Z})$ is large, (ii) $\mathrm{rank}(\boldsymbol{Z})$ is large.

  — trade-off between low-pass-ness and NMF performance.

  — derived from [Fu et al., 2019].

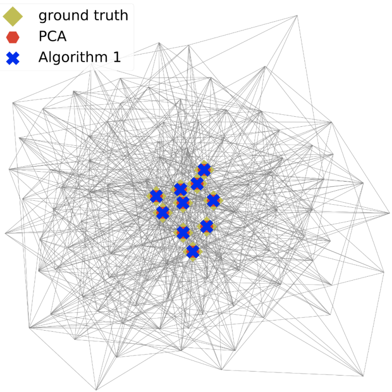
- **Related Works**: centrality ranking [Roddenberry and Segarra, 2021].

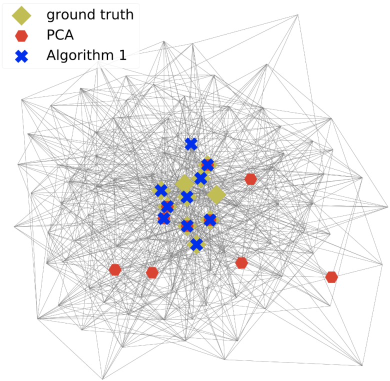
Numerical Experiment: Blind Centrality Learning



- 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	ACN	HON	AXP	IBM	DIS	ORCL	MMM	BRK.B	COST
	0.43	0.56	0.51	0.72	0.50	0.36	0.70	0.33	0.52	0.64
	Average Correlation Score: 0.53 ± 0.133									
PCA (11)	NVDA	NFLX	AMZN	ADBE	PYPL	CAT	MA	GOOG	BA	GOOGL
	0.56	0.60	0.68	0.63	0.65	0.27	0.67	0.63	0.28	0.63
	Average Correlation Score: 0.56 ± 0.154									
GL-SigRep [13]	GOOGL	GOOG	LLY	USB	EMR	DUK	ORCL	GD	VZ	V
	0.63	0.63	0.17	0.43	0.59	0.11	0.70	0.53	0.27	0.71
	Average Correlation Score: 0.48 ± 0.22									
KNN	ACN	HON	ALL	BRK.B	IBM	AXP	EMR	MMM	CSCO	XOM
	0.56	0.51	0.43	0.52	0.50	0.72	0.59	0.33	0.63	0.55
	Average Correlation Score: 0.53 ± 0.107									
SpecTemp [14]	ACN	ORCL	PG	LLY	SUBX	PYPL	MDLZ	FB	PFE	MRK
	0.56	0.70	0.36	0.17	0.58	0.65	0.41	0.61	0.14	0.20
	Average Correlation Score: 0.44 ± 0.211									
Kalofolias [44]	ACN	HON	BRK.B	ALL	AXP	IBM	XOM	KO	USB	COST
	0.56	0.51	0.52	0.43	0.72	0.50	0.55	0.32	0.43	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	MT	KS	RI	TN	MN	NV	ME	MD	IN
	0.79	0.66	0.74	0.67	0.68	0.74	0.43	0.67	0.6	0.62
	Average Correlation Score: 0.66 ± 0.099									
PCA (11)	CA	DE	CO	IL	ND	WV	IA	VA	WY	MA
	0.55	0.46	0.54	0.63	0.72	0.52	0.51	0.56	0.59	0.58
	Average Correlation Score: 0.57 ± 0.072									
GL-SigRep [13]	CA	DE	WV	CO	IL	VA	ND	IA	WY	AZ
	0.55	0.46	0.52	0.54	0.63	0.56	0.72	0.51	0.59	0.31
	Average Correlation Score: 0.54 ± 0.108									
KNN	ND	CA	IL	WV	DE	VA	AZ	CO	WY	IA
	0.72	0.55	0.63	0.52	0.46	0.56	0.31	0.54	0.59	0.51
	Average Correlation Score: 0.54 ± 0.108									
SpecTemp [14]	AL	ND	WV	CA	DE	IL	MO	MA	VA	SD
	0.61	0.72	0.52	0.55	0.46	0.63	0.57	0.58	0.56	0.56
	Average Correlation Score: 0.58 ± 0.069									
Kalofolias [44]	AL	AK	AZ	AR	WV	VA	CA	CO	CT	DE
	0.61	0.63	0.31	0.47	0.52	0.56	0.55	0.54	0.45	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.

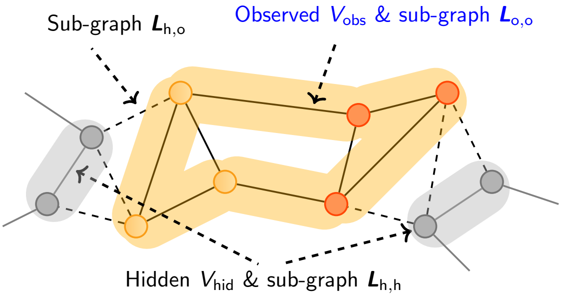
Extension: *Multiple graph* learning from streaming data[11].

[11][He and Wai, 2023b] Y. He, **H.-T.**, "Online Inference for Mixture Model of Streaming Graph Signals with Non-White Excitation", TSP, 2023.

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:

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



Learning with Partial Observation

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

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

I. Leveraging Smoothness: observing that[12]

$$\frac{1}{m}\sum_{i=1}^{m} \mathbf{y}_\ell^\top \mathbf{L}\mathbf{y}_\ell \approx \mathrm{Tr}(\mathbf{C}_y^o \mathbf{L}_{\text{o,o}}) + \mathrm{Tr}(\underbrace{2\mathbf{C}_y^{o,h}\mathbf{L}_{\text{o,h}}^\top}_{\text{low rank if } |\mathcal{V}_{\text{hid}}| \ll N}) + \underbrace{\mathrm{Tr}(\mathbf{C}_y^h \mathbf{L}_{\text{h,h}})}_{\geq 0} \geq 0$$

$$\implies \min_{\mathbf{L}_{o,o},\mathbf{K},\mathbf{R}} \ \mathrm{Tr}(\mathbf{C}_y^o \mathbf{L}_{\text{o,o}}) + \mathrm{Tr}(\mathbf{K}) + \mathrm{Tr}(\mathbf{R}) + \alpha g(\mathbf{L}_{\text{o,o}}) + \gamma\|\mathbf{K}\|_\star$$
$$\text{s.t.} \ \mathrm{Tr}(\mathbf{C}_y^o \mathbf{L}_{\text{o,o}}) + \mathrm{Tr}(\mathbf{K}) + \mathrm{Tr}(\mathbf{R}) \geq 0, \ \mathrm{Tr}(\mathbf{R}) \geq 0, \mathbf{L}_{\text{o,o}} \in \mathcal{L},$$

where $g(\cdot)$, $\mathcal{L}$ are respectively regularization, constraint for $\mathbf{L}_{\text{o,o}}$ to be a proper sub-matrix of Laplacian.

[12] [Buciulea et al., 2022] A. Buciulea, S. Rey, A. G. Marques. Learning graphs from smooth and graph-stationary signals with hidden variables. TSIPN, 2022.

Learning with Partial Observation

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

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

I. Leveraging Smoothness: observing that[12]

$$\frac{1}{m}\sum_{i=1}^{m} \boldsymbol{y}_\ell^\top \boldsymbol{L}\boldsymbol{y}_\ell \approx \mathrm{Tr}(\boldsymbol{C}_y^o \boldsymbol{L}_{\mathrm{o,o}}) + \mathrm{Tr}(\underbrace{2\boldsymbol{C}_y^{o,h}\boldsymbol{L}_{\mathrm{o,h}}^\top}_{\text{low rank if } |\mathcal{V}_{\mathrm{hid}}| \ll N}) + \underbrace{\mathrm{Tr}(\boldsymbol{C}_y^h \boldsymbol{L}_{\mathrm{h,h}})}_{\geq 0} \geq 0$$

$$\Longrightarrow \quad \begin{aligned} \min_{\boldsymbol{L}_{o,o},\boldsymbol{K},\boldsymbol{R}} \quad & \mathrm{Tr}(\boldsymbol{C}_y^o \boldsymbol{L}_{\mathrm{o,o}}) + \mathrm{Tr}(\boldsymbol{K}) + \mathrm{Tr}(\boldsymbol{R}) + \alpha g(\boldsymbol{L}_{\mathrm{o,o}}) + \gamma \|\boldsymbol{K}\|_\star \\ \text{s.t.} \quad & \mathrm{Tr}(\boldsymbol{C}_y^o \boldsymbol{L}_{\mathrm{o,o}}) + \mathrm{Tr}(\boldsymbol{K}) + \mathrm{Tr}(\boldsymbol{R}) \geq 0, \; \mathrm{Tr}(\boldsymbol{R}) \geq 0, \boldsymbol{L}_{\mathrm{o,o}} \in \mathcal{L}, \end{aligned}$$

where $g(\cdot)$, $\mathcal{L}$ are respectively regularization, constraint for $\boldsymbol{L}_{\mathrm{o,o}}$ to be a proper sub-matrix of Laplacian.

[12] [Buciulea et al., 2022] A. Buciulea, S. Rey, A. G. Marques. Learning graphs from smooth and graph-stationary signals with hidden variables. TSIPN, 2022.

Learning with Partial Observation

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

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

II. 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 $V_{\rm obs}$. ↑ 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[12], partial centrality inference[13]

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

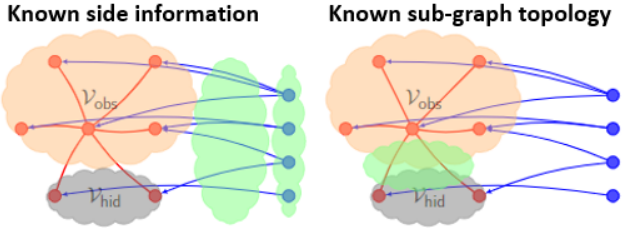
[13] [He and Wai, 2023a] Y. He, **H.-T.**, Central nodes detection from partially observed graph signals, in ICASSP 2023.

Complete Learning with Partial Observation

- **Goal**: inferring the graph features of **the whole $\boldsymbol{A}$**,

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

- Requires *side information or sub-graph topology*:



- We rely on *low-rankness* and aim to learn community or centrality.

Complete Learning with Partial Observation

- **Goal**: inferring the graph features of **the whole $\boldsymbol{A}$**,

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

- Requires *side information or sub-graph topology*:

(I) **If $\boldsymbol{A}_{o,h}$ is known**[14]: Nyström method [Fowlkes et al., 2004] to 'interpolate' eigenvectors,

$$\text{(i) top-}K\ \widehat{\boldsymbol{U}}_K \text{ of } \widehat{\boldsymbol{C}}_y^{\rm obs}, \quad \text{(ii) } \widehat{\boldsymbol{V}}_K := \begin{pmatrix} \widehat{\boldsymbol{U}}_K \\ \boldsymbol{A}_{\rm h,o}\widehat{\boldsymbol{U}}_K / \widehat{\lambda} \end{pmatrix}, \quad \text{(iii) } k\text{-means on } \widehat{\boldsymbol{V}}_K.$$

  - Assume that $V_{\rm obs}$ is chosen at random, then w.h.p.,

$$\underbrace{F(\widehat{\mathcal{N}}_1, \ldots, \widehat{\mathcal{N}}_k; \boldsymbol{V}_K)}_{K\text{-means obj. on whole graph.}} - F^\star = \mathcal{O}\left(\eta_K + \frac{1}{\sqrt{m}} + \frac{1}{\sqrt{|V_{\rm obs}|}} + \frac{|V_{\rm hid}|}{|V|} \right).$$

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

Complete Learning with Partial Observation

- **Goal**: inferring the graph features of **the whole $\boldsymbol{A}$**,

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

- Requires *side information or sub-graph topology*:

(II) Excitation signal is known[14]: recall $\boldsymbol{x}^{(t)} = \boldsymbol{B}\boldsymbol{z}^{(t)}$ and we know $\boldsymbol{B}, \boldsymbol{z}^{(t)}$.

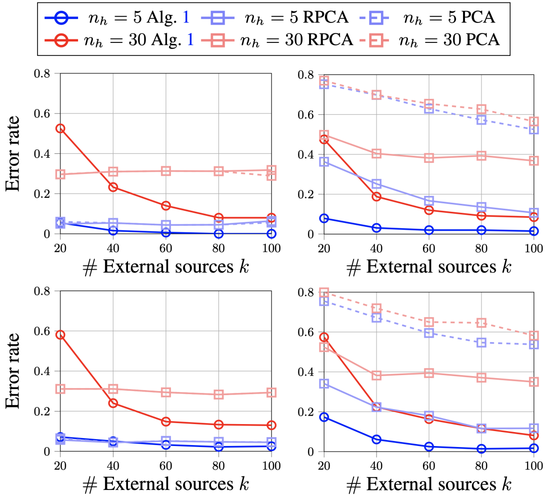
$$\boldsymbol{Y}_{\rm obs}\boldsymbol{Z}^{\dagger} = \underbrace{\widetilde{h}_{\rho}(\lambda_1)\boldsymbol{c}_{\rm obs}\boldsymbol{c}^{\top}\boldsymbol{B}}_{\text{rank-1 w/ eig.-centrality}} + \underbrace{\rho\boldsymbol{E}_{\rm o}\boldsymbol{B}}_{\text{sparse}} + \mathcal{O}(\tilde{\eta}), \quad \text{holds} \quad \underbrace{\forall\ \rho > 0}_{\text{`boosting'}}$$

- **Full eigen-centrality $\boldsymbol{c}$** can be estimated if

$$\text{Excitation rank} = \mathrm{rank}(\boldsymbol{B}) = K \geq |V_{\rm hid}| + 1$$

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

Numerical Experiment: Complete Graph Learning



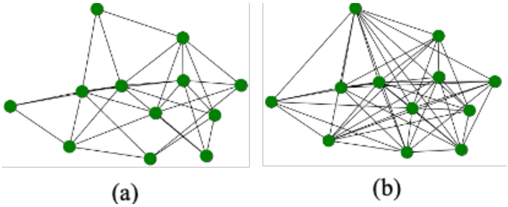
- Increasing the excitation rank K improves the detection performances.

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**.

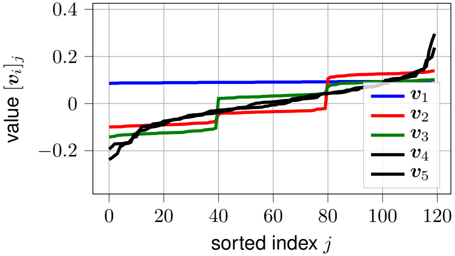


(a) Ground truth. (b) Topology learnt by GL-SigRep on **non-low-pass** signals.

- *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

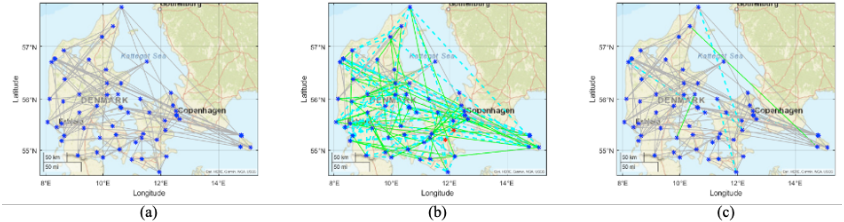
- **Assume**: no. of dense clusters, K, in the graph is known a-priori.
  $\Longrightarrow \lambda_1, \ldots, \lambda_K \approx 0 \Longrightarrow$ if the filter is low pass, it will be K low pass.
- **Observation**: graph signals from K low pass filter exhibit particular *spectral signature*. E.g., SBM 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) 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.[15]

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

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**[16] 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[17].
  — Proof: depends on convergence of graph filter on SBM.

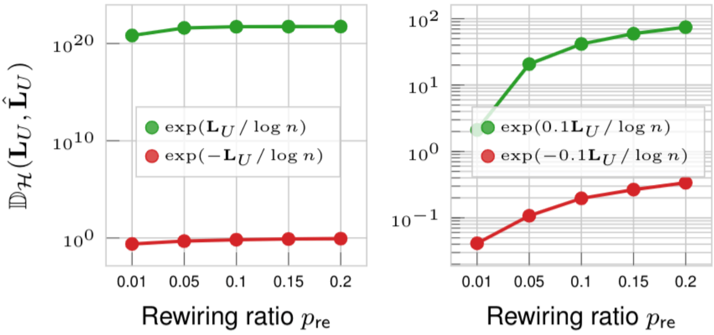
[16] [Gama et al., 2020] F. Gama, J. Bruna, A. Ribeiro. Stability properties of graph neural networks. TSP, 2020.
[17] [Nguyen et al., 2022] H. Nguyen, Y. He, **H.-T.**, "On the stability of low pass graph filter with a large number of edge rewires," in ICASSP, 2022.

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.

Generalization Bound

- **Sample complexity** of MPNN (GCN) learning[18] analyzed via

$$\mathcal{E}_m^n = \mathbb{E}_{\mu_{\mathcal{G}}^m}\Big[\sup_{\Theta}\Big(\underbrace{\frac{1}{m}\sum_{i=1}^{m}\mathcal{L}(\Theta_{G^i}(\boldsymbol{x}^i),\boldsymbol{y}^i)}_{\text{empirical risk}} - \underbrace{\mathbb{E}_{\mu_{\mathcal{G}}}[\mathcal{L}(\Theta_G(\boldsymbol{x}),\boldsymbol{y})]}_{\text{expected risk}}\Big)^2\Big] \le \frac{C}{m} n^{-\frac{1}{D_{\mathcal{X}}+1}}$$

  where $m =$ no. of training sets, $n =$ no. of nodes, and $G^i, \boldsymbol{x}^i, \boldsymbol{y}^i$ is the ith training set of graph, attributes (signals), labels.
- **Proof**: MPNN $\to$ graphon limit as $n \to \infty$ [Keriven et al., 2020].
- C depends on Lipschitz-ness of message (activation) functions, etc. $\leftarrow$ no explicit dependence on graph filter.

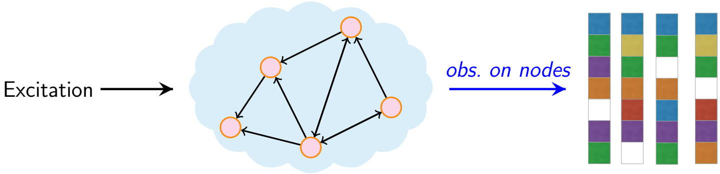
- Recent work[19] provide transferability bound utilizing the spectrum of graph filter similar to [Keriven et al., 2020] $\leftarrow$ *open problem?*

---

[18] [Maskey et al., 2022] S. Maskey, R. Levie, Y. Lee, and G. Kutyniok. Generalization analysis of message passing neural networks on large random graphs. in NeurIPS, 2022.

[19] [Ruiz et al., 2021] L. Ruiz, L. F. Chamon, A. Ribeiro. Transferability properties of graph neural networks. ArXiv, 2021

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 graph ML, ...

Perspectives

- Graph learning from partial observations with **many hidden nodes**.
  — it is the case for observations on social/economics networks.

- Learning from **multi-attribute signal**: graphs do not live in isolation, e.g., multiplex networks in ecology, social systems, etc.

  — needs new notion for graph filter:

  $$\text{Prod-Graph Filter}: \quad \mathcal{H}(\boldsymbol{L}^{\mathtt{C}}, \boldsymbol{L}^{\mathtt{G}}) = \sum_{i,j} h_{ij} (\boldsymbol{L}^{\mathtt{C}})^i \otimes (\boldsymbol{L}^{\mathtt{G}})^j,$$

  and **interpretation** for low pass multi-layer graph filter [Zhang et al., 2023b, Kadambari and Chepuri, 2021, Einizade and Sardouie, 2022].

Thank you!
Questions & comments are welcomed.

Agenda

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.

[Buciulea et al., 2022] Buciulea, A., Rey, S., and Marques, A. G. (2022).
Learning graphs from smooth and graph-stationary signals with hidden variables.
IEEE Transactions on Signal and Information Processing over Networks, 8:273–287.

[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.

[Einizade and Sardouie, 2022] Einizade, A. and Sardouie, S. H. (2022).
Learning product graphs from spectral templates.
arXiv preprint arXiv:2211.02893.

[Fowlkes et al., 2004] Fowlkes, C., Belongie, S., Chung, F., and Malik, J. (2004).
Spectral grouping using the nystrom method.
IEEE transactions on pattern analysis and machine intelligence, 26(2):214–225.

[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.

[Fu et al., 2019] Fu, X., Huang, K., Sidiropoulos, N. D., and Ma, W.-K. (2019).
Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications.
IEEE Signal Process. Mag., 36(2):59–80.

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

[He and Wai, 2023b] He, Y. and Wai, H.-T. (2023b).
Online inference for mixture model of streaming graph signals with non-white excitation.
IEEE Trans. Signal Processing.

[Isufi et al., 2022] Isufi, E., Gama, F., Shuman, D. I., and Segarra, S. (2022).
Graph filters for signal processing and machine learning on graphs.
arXiv preprint arXiv:2211.08854.

[Kadambari and Chepuri, 2021] Kadambari, S. K. and Chepuri, S. P. (2021).
Product graph learning from multi-domain data with sparsity and rank constraints.
IEEE Transactions on Signal Processing, 69:5665–5680.

[Keriven et al., 2020] Keriven, N., Bietti, A., and Vaiter, S. (2020).
Convergence and stability of graph convolutional networks on large random graphs.
Advances in Neural Information Processing Systems, 33:21512–21523.

[Maskey et al., 2022] Maskey, S., Levie, R., Lee, Y., and Kutyniok, G. (2022).
Generalization analysis of message passing neural networks on large random graphs.
In *Advances in Neural Information Processing Systems*.

[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*.

[Püschel and Moura, 2003] Püschel, M. and Moura, J. M. (2003).
The algebraic approach to the discrete cosine and sine transforms and their fast algorithms.
SIAM Journal on Computing, 32(5):1280–1316.

[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.

[Roddenberry et al., 2020] Roddenberry, T. M., Schaub, M. T., Wai, H.-T., and Segarra, S. (2020).
Exact blind community detection from signals on multiple graphs.
IEEE Transactions on Signal Processing, 68:5016–5030.

[Roddenberry and Segarra, 2021] Roddenberry, T. M. and Segarra, S. (2021).
Blind inference of eigenvector centrality rankings.
IEEE Transactions on Signal Processing, 69:3935–3946.

[Ruiz et al., 2021] Ruiz, L., Chamon, L. F., and Ribeiro, A. (2021).
Transferability properties of graph neural networks.
arXiv preprint arXiv:2112.04629.

[Sandryhaila and Moura, 2013] Sandryhaila, A. and Moura, J. M. (2013).

Discrete signal processing on graphs.
IEEE transactions on signal processing, 61(7):1644–1656.

[Schaub et al., 2020] Schaub, M. T., Segarra, S., and Tsitsiklis, J. N. (2020).
Blind identification of stochastic block models from dynamical observations.
SIAM Journal on Mathematics of Data Science, 2(2):335–367.

[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., 2023a] Zhang, C., He, Y., and Wai, H.-T. (2023a).
Detecting low pass graph signals via spectral pattern: Sampling complexity and applications.
ArXiv.

[Zhang et al., 2023b] Zhang, C., He, Y., and Wai, H.-T. (2023b).
Product graph learning from multi-attribute graph signals with inter-layer coupling.
In *ICASSP*.