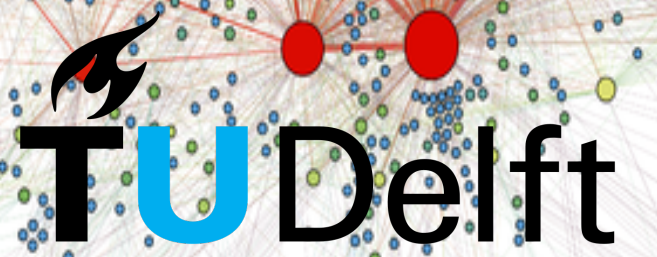
Learning Graphs and Simplicial Complexes from Data

Andrei Buciulea Vlas

Joint work with E. Isufi, G. Leus and A. G. Marques

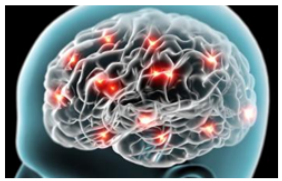


October 5, TU Delft 2023

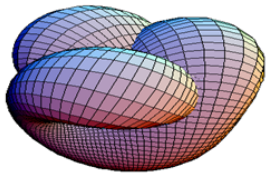
Motivating Examples: Networked Data

- Huge data sets are generated in networks (transportation networks, biological networks, brain networks, computer networks, social networks)
- The data structure carries critical information about the nature of the data
- Modelling the data structure with graphs

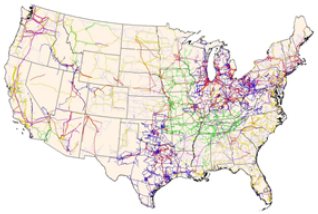
Interpolate a brain signal from local observations



Compress a signal in an irregular domain



Localize the source of a rumor



Smooth an observed network profile

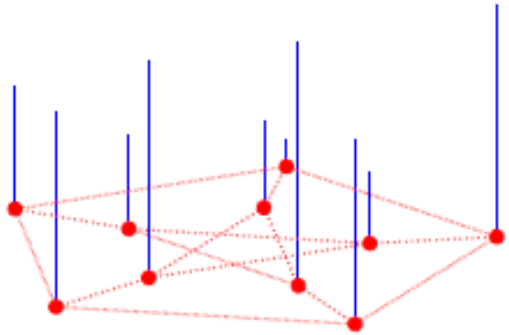
Predict the evolution of a network process

Infer the topology where the signals reside

Graph Signal Processing (GSP)

- Consider an undirected weighted graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{W})$

  $\Rightarrow \mathcal{V}$, $\mathcal{E}$, $\mathcal{W} \rightarrow$ set of nodes, edges, weights



- Define a signal $\mathbf{x} \in \mathbb{R}^N$ on the top of the graph

  $\Rightarrow x_i =$ value of graph signal (GS) at node i

- Associated with $\mathcal{G}$ is the *Graph-Shift Operator* (GSO)

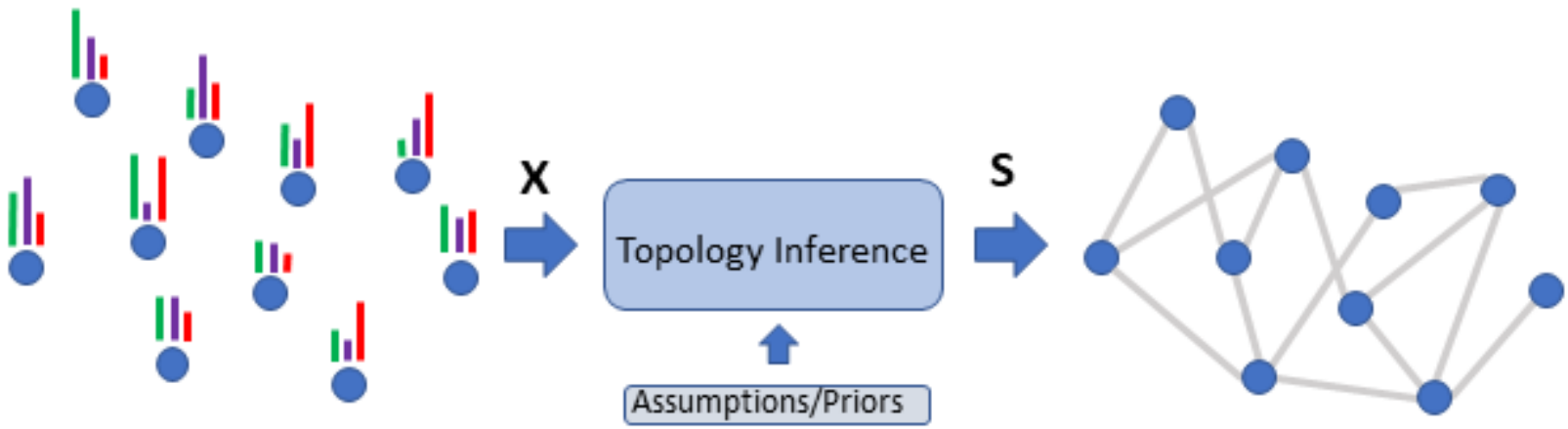
  $\Rightarrow \mathbf{S} \in \mathbb{R}^{N \times N}$, $S_{ij} \neq 0$ for $i = j$ and $(i, j) \in \mathcal{E}$

  $\Rightarrow$ Ex: Adjacency $\mathbf{A}$, Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$, random walk...

Graph Learning: Motivation and Context

Network topology inference from nodal observations

"Given a collection $\mathbf{X} := [\mathbf{x}_1, ..., \mathbf{x}_R]$ of graph signal observations supported on the unknown graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathbf{A})$ find an optimal $\mathbf{S}$"



- **This work**:

  $\Rightarrow$ Use data to learn both, the graph and the higher-order interactions

  $\Rightarrow$ Modelling data and graph using Autoregressive Graph Volterra Models

Graph Learning: Related work (I)

- Goal: use $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_R] \in \mathbb{R}^{N \times R}$ to learn $\mathbf{S}$ with $\hat{\mathbf{\Sigma}} = \frac{1}{R}\mathbf{X}\mathbf{X}^\top$
  - Let $\mathbf{X}$ supported on $\mathcal{G} \Rightarrow \{\text{Correlation networks}\}$

    $\hat{\mathbf{S}} \approx \hat{\mathbf{\Sigma}} = \mathbb{E}\left[\mathbf{X}\mathbf{X}^\top\right]$ ($\hat{\mathbf{S}}$ is a thresholded version of $\hat{\mathbf{\Sigma}}$)

Graph Learning: Related work (I)

- Goal: use $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_R] \in \mathbb{R}^{N \times R}$ to learn $\mathbf{S}$ with $\hat{\mathbf{\Sigma}} = \frac{1}{R}\mathbf{X}\mathbf{X}^\top$
  - Let $\mathbf{X}$ supported on $\mathcal{G} \Rightarrow \{\text{Correlation networks}\}$

    $\hat{\mathbf{S}} \approx \hat{\mathbf{\Sigma}} = \mathbb{E}\left[\mathbf{X}\mathbf{X}^\top\right]$ ($\hat{\mathbf{S}}$ is a thresholded version of $\hat{\mathbf{\Sigma}}$)

  - Let $\mathbf{X}$ be i.i.d samples of $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma}) \Rightarrow \{\text{Part. corr. netw.}\}$ GL

    $$\hat{\mathbf{S}} = \underset{\mathbf{S} \succeq 0, \mathbf{S} \in \mathcal{S}_\Theta}{\operatorname{argmin}} \; -\log(\det(\mathbf{S})) + \operatorname{tr}(\hat{\mathbf{\Sigma}}\mathbf{S}) + \rho \mathrm{h}(\mathbf{S})$$ [Fr.08]

Graph Learning: Related work (I)

- Goal: use $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_R] \in \mathbb{R}^{N \times R}$ to learn $\mathbf{S}$ with $\hat{\mathbf{\Sigma}} = \frac{1}{R}\mathbf{X}\mathbf{X}^\top$
  - Let $\mathbf{X}$ supported on $\mathcal{G} \Rightarrow \{$Correlation networks$\}$

    $\hat{\mathbf{S}} \approx \hat{\mathbf{\Sigma}} = \mathbb{E}\left[\mathbf{X}\mathbf{X}^\top\right]$ ($\hat{\mathbf{S}}$ is a thresholded version of $\hat{\mathbf{\Sigma}}$)
  - Let $\mathbf{X}$ be i.i.d samples of $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma}) \Rightarrow \{$Part. corr. netw.$\}$ GL

    $\hat{\mathbf{S}} = \underset{\mathbf{S} \succeq 0, \mathbf{S} \in \mathcal{S}_\Theta}{\operatorname{argmin}} \; -\log(\det(\mathbf{S})) + \operatorname{tr}(\hat{\mathbf{\Sigma}}\mathbf{S}) + \rho \mathrm{h}(\mathbf{S})$ [Fr.08]
  - Let $\mathbf{X}$ be stationary w.r.t $\mathbf{S} \Rightarrow \{$Graph-st. diff. process.$\}$ GSR

    $\hat{\mathbf{S}} = \underset{\mathbf{S} \in \mathcal{S}}{\operatorname{argmin}} \; \|\mathbf{S}\|_0 \quad$ s. to $\quad \hat{\mathbf{\Sigma}}\mathbf{S} = \mathbf{S}\hat{\mathbf{\Sigma}}$ [Segarra17]

Graph Learning: Related work (I)

- Goal: use $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_R] \in \mathbb{R}^{N \times R}$ to learn $\mathbf{S}$ with $\hat{\mathbf{\Sigma}} = \frac{1}{R}\mathbf{X}\mathbf{X}^\top$
  - Let $\mathbf{X}$ supported on $\mathcal{G} \Rightarrow$ {Correlation networks}

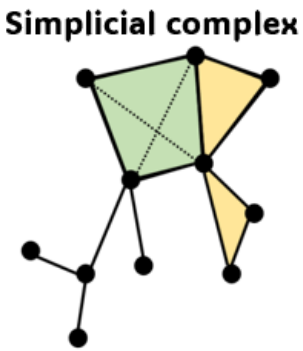
    $\hat{\mathbf{S}} \approx \hat{\mathbf{\Sigma}} = \mathbb{E}\left[\mathbf{X}\mathbf{X}^\top\right]$ ($\hat{\mathbf{S}}$ is a thresholded version of $\hat{\mathbf{\Sigma}}$)
  - Let $\mathbf{X}$ be i.i.d samples of $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma}) \Rightarrow$ {Part. corr. netw.} GL

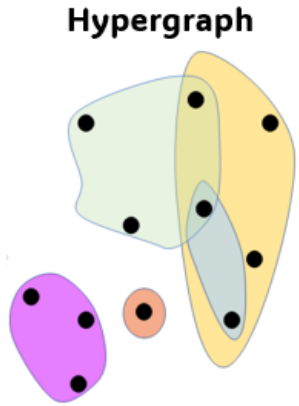
    $\hat{\mathbf{S}} = \underset{\mathbf{S} \succeq 0, \mathbf{S} \in \mathcal{S}_\Theta}{\text{argmin}} \; -\log(\det(\mathbf{S})) + \text{tr}(\hat{\mathbf{\Sigma}}\mathbf{S}) + \rho \text{h}(\mathbf{S})$ [Fr.08]
  - Let $\mathbf{X}$ be stationary w.r.t $\mathbf{S} \Rightarrow$ {Graph-st. diff. process.} GSR

    $\hat{\mathbf{S}} = \underset{\mathbf{S} \in \mathcal{S}}{\text{argmin}} \; \|\mathbf{S}\|_0 \quad \text{s. to} \quad \hat{\mathbf{\Sigma}}\mathbf{S} = \mathbf{S}\hat{\mathbf{\Sigma}}$ [Segarra17]
  - Other approaches:

    Smoothness: $\hat{\mathbf{S}} = \underset{\mathbf{S} \succeq 0, \mathbf{S} \in \mathcal{S}_\mathbf{L}}{\text{argmin}} \; \text{tr}(\mathbf{X}^\top \mathbf{S} \mathbf{X}) + \text{f}(\mathbf{S})$ [Dong17]

    Sparse SEM: $\hat{\mathbf{S}} = \underset{\mathbf{S} \succeq 0, \mathbf{S} \in \mathcal{S}}{\text{argmin}} \; \|\mathbf{X} - \mathbf{S}\mathbf{X}\|_F^2 + \text{g}(\mathbf{S})$ [Bazerque13]





Graph Learning: Related work (II)

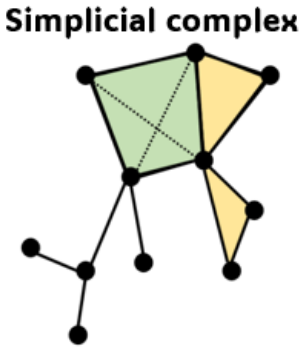
- Goal: use $\mathbf{X}$ and $\mathbf{S}$ to learn higher-order interactions
  - **Vietoris–Rips complex approach** [Zomorodian10] RC
    - $\Rightarrow$ Is defined as a way of forming a topological space from distances in a set of points
    - $\Rightarrow$ Learn simplicial complexes (SCs) from a distance matrix computed from the data (i.e. $\hat{\mathbf{\Sigma}} = \mathbb{E}\left[\mathbf{X}\mathbf{X}^{\mathsf{T}}\right]$)

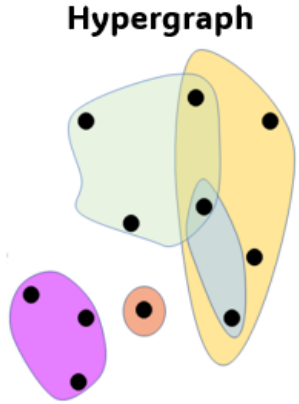
Graph Learning: Related work (II)

- Goal: use $\mathbf{X}$ and $\mathbf{S}$ to learn higher-order interactions
  - **Vietoris–Rips complex approach** [Zomorodian10] RC
    - $\Rightarrow$ Is defined as a way of forming a topological space from distances in a set of points
    - $\Rightarrow$ Learn simplicial complexes (SCs) from a distance matrix computed from the data (i.e. $\hat{\mathbf{\Sigma}} = \mathbb{E}\left[\mathbf{X}\mathbf{X}^{\mathsf{T}}\right]$)
  - **Learning SCs from data** [Barbarossa20] MTV-SC
    - $\Rightarrow$ Assuming specific physical nature for the data defined on the edges of a graph $\mathbf{x}_1 = \mathbf{B}_1^{\top} \mathbf{s}_0 + \mathbf{s}_H + \mathbf{B}_2 \mathbf{s}_2 + \mathbf{w}$
    - $\Rightarrow$ Learning higher-order interactions from data defined on the edges ($\mathbf{X}_1$) and assuming known topology ($\mathbf{B}_1$)

Graph Learning: Related work (II)

- Goal: use $\mathbf{X}$ and $\mathbf{S}$ to learn higher-order interactions
  - **Vietoris–Rips complex approach** [Zomorodian10] RC
    - $\Rightarrow$ Is defined as a way of forming a topological space from distances in a set of points
    - $\Rightarrow$ Learn simplicial complexes (SCs) from a distance matrix computed from the data (i.e. $\hat{\mathbf{\Sigma}} = \mathbb{E}\left[\mathbf{X}\mathbf{X}^{\mathsf{T}}\right]$)
  - **Learning SCs from data** [Barbarossa20] MTV-SC
    - $\Rightarrow$ Assuming specific physical nature for the data defined on the edges of a graph $\mathbf{x}_1 = \mathbf{B}_1^{\top} \mathbf{s}_0 + \mathbf{s}_H + \mathbf{B}_2 \mathbf{s}_2 + \mathbf{w}$
    - $\Rightarrow$ Learning higher-order interactions from data defined on the edges ($\mathbf{X}_1$) and assuming known topology ($\mathbf{B}_1$)
  - **Learning hypergraphs from data** [Tang23] HGSL
    - $\Rightarrow$ Assume that the hypergraph structure is derived from a learnable graph structure obtained from data
    - $\Rightarrow$ The learned higher-order interactions (hyperedges) are obtained based on the learned topology from data





Problem Formulation: Data Modelling

- Data Modelling: Autoregressive Graph Volterra Model of order 2

$$\mathbf{X} = \mathbf{H}_1\mathbf{X} + \mathbf{H}_2\mathbf{Y} + \mathbf{V} + \mathbf{E}, \text{ with } \mathbf{Y} = \mathbf{X} \odot \mathbf{X} \in \mathbb{R}^{N^2 \times R}$$

$\mathbf{H}_1 \in \mathbb{R}^{N \times N}$ pairwise interactions, $\mathbf{H}_2 \in \mathbb{R}^{N \times N^2}$ node-pair interactions

- $\mathbf{H}_1\mathbf{X}$ is a linear combination of the signals in the other nodes
- $\mathbf{H}_2\mathbf{Y}$ is a product of the signals in the other tuples of nodes

Problem Formulation: Data Modelling

- Data Modelling: Autoregressive Graph Volterra Model of order 2

$$\mathbf{X} = \mathbf{H}_1\mathbf{X} + \mathbf{H}_2\mathbf{Y} + \mathbf{V} + \mathbf{E}, \text{ with } \mathbf{Y} = \mathbf{X} \odot \mathbf{X} \in \mathbb{R}^{N^2 \times R}$$

$\mathbf{H}_1 \in \mathbb{R}^{N \times N}$ pairwise interactions, $\mathbf{H}_2 \in \mathbb{R}^{N \times N^2}$ node-pair interactions

- $\mathbf{H}_1\mathbf{X}$ is a linear combination of the signals in the other nodes
- $\mathbf{H}_2\mathbf{Y}$ is a product of the signals in the other tuples of nodes

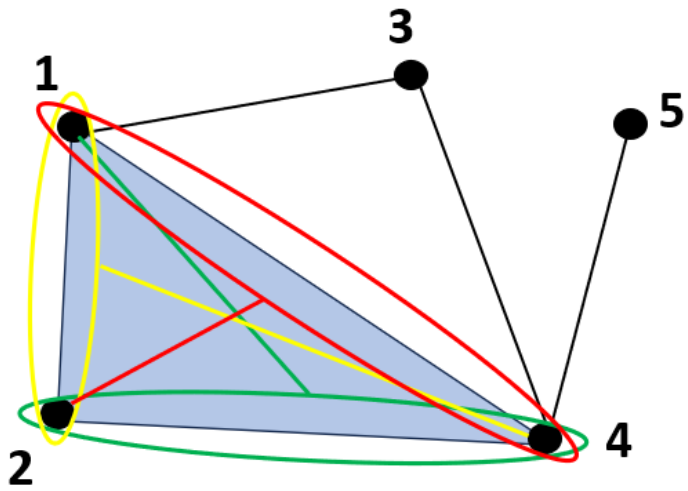
- Example of signal representation in terms of $\mathbf{H}_1$ and $\mathbf{H}_2$

$$x_2 = \mathbf{H}_1[2,1]x_1 + \mathbf{H}_1[2,4]x_4 + \mathbf{H}_2[2,(1,4)]x_1x_4 + \mathbf{H}_2[2,(4,1)]x_1x_4 + v_2 + e_2.$$

Part of x_2 is described by:

$\Rightarrow$ node-to-node interactions ($\mathbf{H}_1$)

$\Rightarrow$ node-to-pair interactions ($\mathbf{H}_2$)



Problem Formulation: Graph Modelling

- Recalling the signal modelling

$$\mathbf{X} = \mathbf{H}_1\mathbf{X} + \mathbf{H}_2\mathbf{Y} + \mathbf{V} + \mathbf{E}, \text{ with } \mathbf{Y} = \mathbf{X} \odot \mathbf{X}.$$

- Graph Modelling: pairwise interactions $\mathbf{H}_1$.

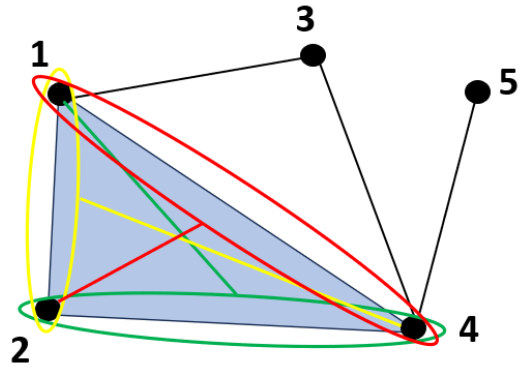
  $\Rightarrow \mathcal{H}_1 = \{\mathbf{H}_1 \geq \mathbf{0}, \mathbf{B}_1 \circ \mathbf{H}_1 = \mathbf{0}, \ \mathbf{H}_1 = \mathbf{H}_1^\top\}$

  $\Rightarrow$ Pos. weights, no self-loops ($\mathbf{B}_1 = \mathbf{I}$), symmetry.

- Graph Modelling: node-to-pair interactions $\mathbf{H}_2$.

  $\Rightarrow \mathcal{H}_2 = \{\mathbf{H}_2 \geq \mathbf{0}, \mathbf{B}_2 \circ \mathbf{H}_2 = \mathbf{0}\}$

  $\Rightarrow$ Positive weights, no self-loops



H1	1	2	3	4	5
1		■	■	■	
2	■			■	
3	■			■	
4	■	■	■		■
5				■	

H2	(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(3,1)	(3,2)	(3,3)	(3,4)	(3,5)	(4,1)	(4,2)	(4,3)	(4,4)	(4,5)	(5,1)	(5,2)	(5,3)	(5,4)	(5,5)
1									■								■								
2				■												■									
3																									
4		■				■																			
5																									

Proposed Approach

Proposed formulation for learning graphs and simplicial complexes

$$(\hat{\mathbf{H}}_1, \hat{\mathbf{H}}_2) = \underset{\mathbf{H}_1 \in \mathcal{H}_1, \mathbf{H}_2 \in \mathcal{H}_2}{\operatorname{argmin}} \quad \|\mathbf{X} - \mathbf{H}_1\mathbf{X} - \mathbf{H}_2\mathbf{Y} - \mathbf{V}\|_F^2 + \alpha\|\mathbf{H}_1\|_1 + \beta\|\mathbf{H}_2\|_1$$

$$\text{s. t.} \qquad \mathbf{H}_2[k,(i,j)] \leq \theta \mathbb{1}(\mathbf{H}_1[k,i]\mathbf{H}_1[k,j]\mathbf{H}_1[i,j]);$$

$\Rightarrow$ Fitting the available data to the autoregressive graph Volterra model

$\Rightarrow$ Controlling the number of node-to-node interactions ($\|\mathbf{H}_1\|_1$) with α

$\Rightarrow$ Controlling the number of node-to-pair interactions ($\|\mathbf{H}_2\|_1$) with β

$\Rightarrow$ Filled triangle can exist if nodes i, j, and k are interconnected

- Non-convex formulation because of the trilinear constraint

  $\Rightarrow$ Next $\rightarrow$ convex formulation to address non-convexities.

Proposed Convex Approach

Convex formulation for learning graphs and simplicial complexes

$$(\hat{\mathbf{H}}_1, \hat{\mathbf{H}}_2) = \underset{\mathbf{H}_1 \in \mathcal{H}_1, \mathbf{H}_2 \in \mathcal{H}_2}{\operatorname{argmin}} \|\mathbf{X} - \mathbf{H}_1\mathbf{X} - \mathbf{H}_2\mathbf{Y} - \mathbf{V}\|_F^2 + \alpha\|\mathbf{H}_1\|_1 + \beta\|\mathbf{H}_2\|_1 + \gamma \sum_{i,j,k=1}^{N} \|\mathbf{Q}^{(i,j,k)} \circ [\mathbf{H}_1, \mathbf{H}_2]\|_F$$

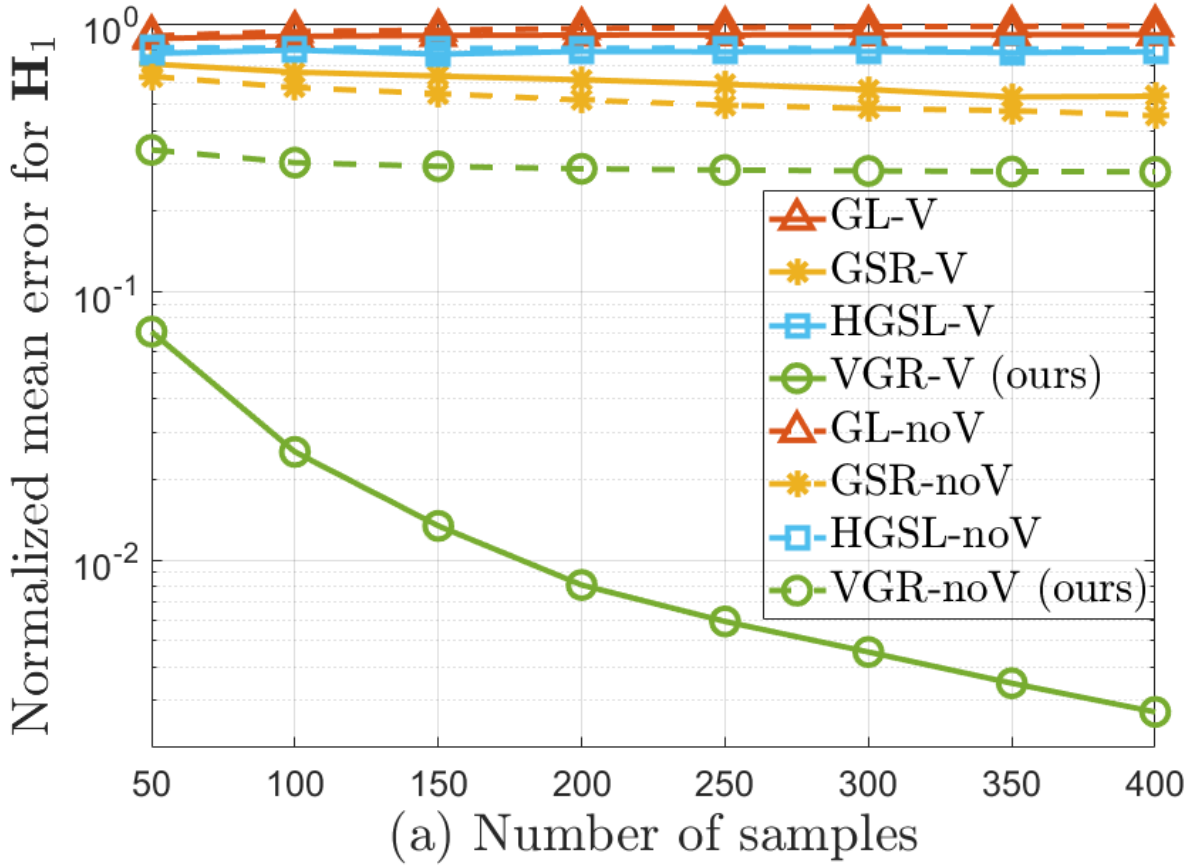
- Binary matrix $\mathbf{Q}^{(i,j,k)} \in \mathbb{R}^{N \times (N+N^2)}$ involving three nodes
  - $\Rightarrow$ Edges between the three nodes

    $\mathbf{Q}^{(i,j,k)}[i,j] = 1, \qquad \mathbf{Q}^{(i,j,k)}[i,k] = 1, \qquad \mathbf{Q}^{(i,j,k)}[j,k] = 1$
  - $\Rightarrow$ Node-pair interactions between the three nodes

    $\mathbf{Q}^{(i,j,k)}[i, Nj+k] = 1,\ \mathbf{Q}^{(i,j,k)}[j, Ni+k] = 1,\ \mathbf{Q}^{(i,j,k)}[k, Ni+j] = 1$
- Group entries of $\mathbf{H}_1$ and $\mathbf{H}_2$ that participate in a triangle using $\mathbf{Q}^{(i,j,k)}$
- Controlling the number of filled triangles ($\mathbf{H}_2$) with β

Synthetic Data Results

- Estimation performance ($\mathrm{err}(\mathbf{H}_1)$) of different algorithms as R increases

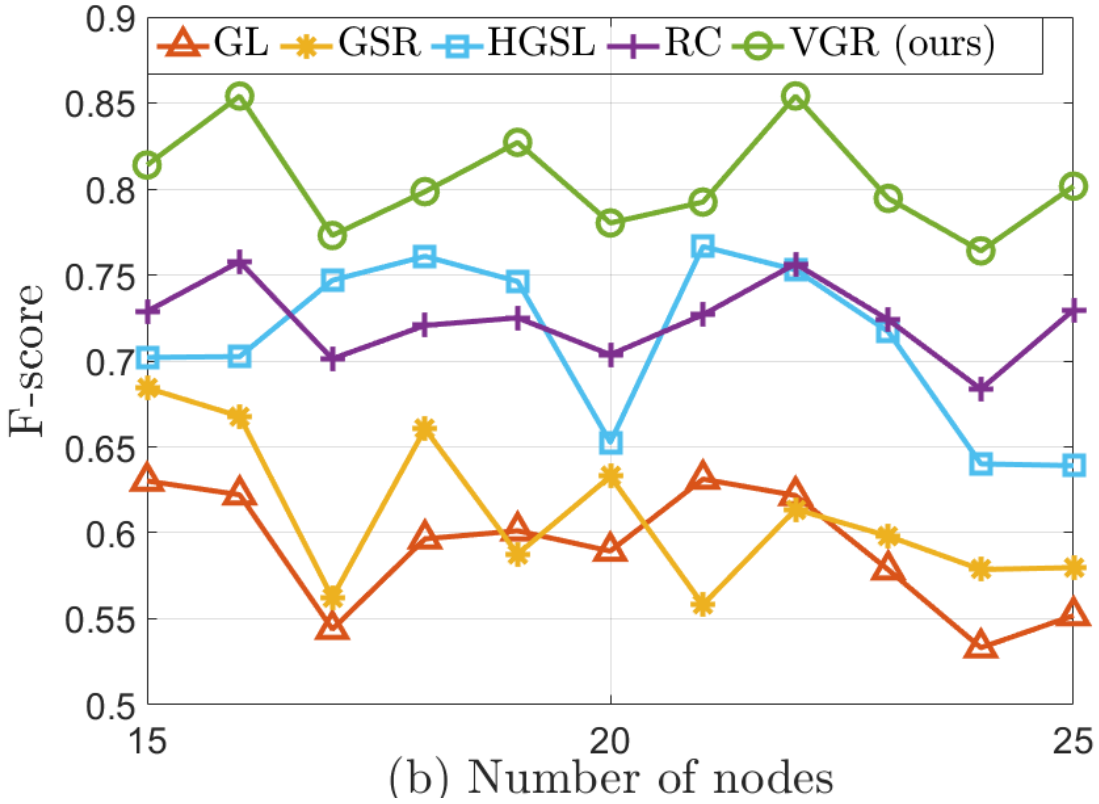


- Normalized error when estimating filled triangles ($\mathrm{err}(\mathbf{H}_2)$)

Alg. \ R	50	100	200	300	400	500
MTV-SC	1.505	1.496	1.497	1.493	1.494	1.490
RC	0.790	0.767	0.761	0.753	0.748	0.751
VGR	0.559	0.428	0.294	0.214	0.165	0.133

Real Data Results

- Estimation performance (F-score) of different algorithms as N increases



- F-score and $\mathrm{err}(\mathbf{H}_2)$ when estimating filled triangles

	F-score			Error		
Alg. \ N	15	20	25	15	20	25
MTV-SC	0.093	0.058	0.056	7.418	7.536	7.530
RC	0.667	0.650	0.585	1.350	2.101	2.837
VGR	0.718	0.676	0.625	0.548	0.558	0.649

Conclusions

- New scheme that jointly learns graphs and simplicial complexes
- Key **assumptions**:
  - $\Rightarrow$ Model data using autoregressive graph Volterra models
  - $\Rightarrow$ Model network as graph ($\mathbf{H}_1$) and simplicial complexes ($\mathbf{H}_2$)
- Jointly learn from data node-pair interactions and filled triangles
- Challenge: non-convex approach due to filled triangle modelling
  - $\Rightarrow$ Convex approach using group sparsity term
- Encouraging results in both synthetic and real data sets

- **THANKS!**
  - $\Rightarrow$ Feel free to contact me for questions and code andrei.buciulea@urjc.es