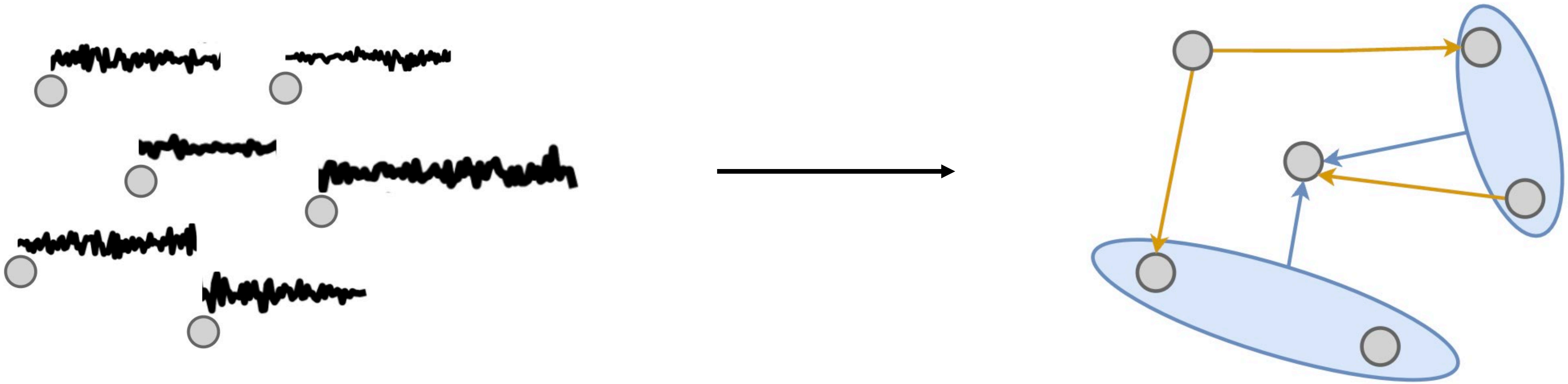
Scalable Higher-Order Topology Identification from Nodal Observations



Geert Leus

Delft University of Technology

Thanks: *Ruben Wijnands, Andrea Cavallo, Borbála Hunyadi, Elvin Isufi*

TU Delft

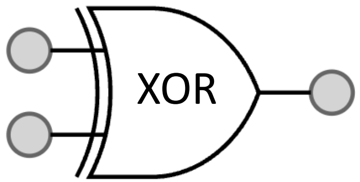
Introduction

Why learn relationships in data?

- Understanding complex systems (brain, social circles,…)
- Use learned relationships as
  - bias (e.g. graph neural networks)
  - features for downstream tasks (e.g. classification)

Why learn higher-order interactions in data?

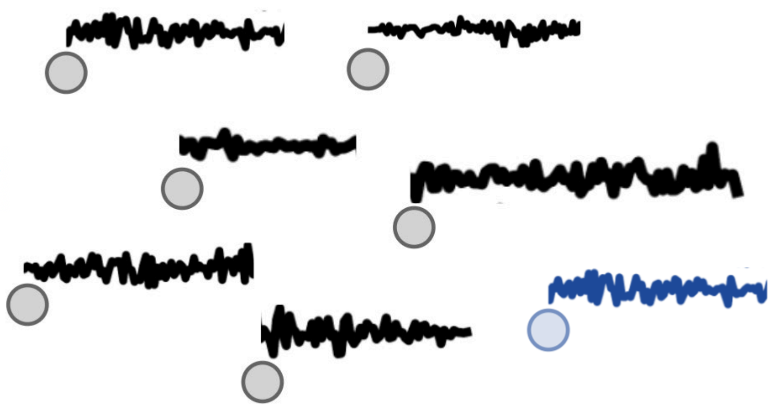
- Many systems involve interactions between multiple entities (nodes)
- Methods exploring pair-wise interactions ignore higher-order interactions



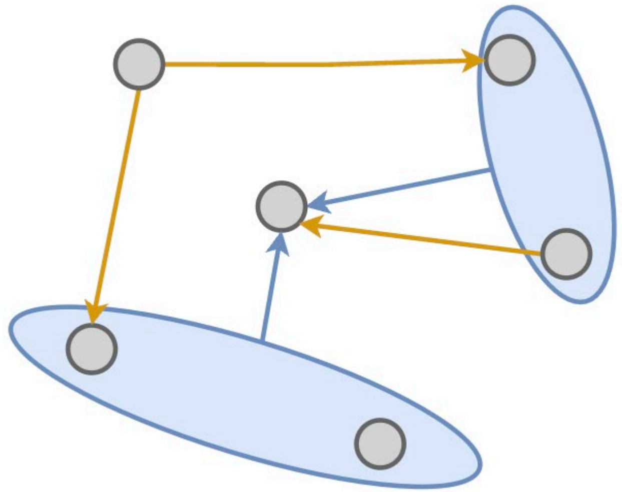
TU Delft

Problem formulation

Given a set of N nodes, a target node signal $x \in \mathbb{R}$
and all node signals excluding the target node $\tilde{\mathbf{x}} = [\tilde{x}_1, \ldots, \tilde{x}_{N-1}]^{\mathsf{T}} \in \mathbb{R}^{N-1}$



Goal
Infer the higher-order topology with interactions up to order D



The autoregressive graph Volterra model (AGVM)

Structural equation model (SEM)

$$x = \sum_{j=1}^{N-1} a_j^{(1)} \tilde{x}_j + \epsilon$$

AVGM

$$x = a^{(0)} + \sum_{j=1}^{N-1} a_j^{(1)} \tilde{x}_j + \sum_{d=2}^{D} A^{(d)}(\tilde{\mathbf{x}}) + \epsilon$$

$$A^{(d)}(\tilde{\mathbf{x}}) = \sum_{j_1=1}^{N-1} \cdots \sum_{j_d=j_{d-1}}^{N-1} a^{(d)}_{j_1,j_2,\ldots,j_d} g(\{\tilde{x}_{j_q}\}_{q=1}^{d})$$

Example

$$x = a_5^{(1)} \tilde{x}_5 + a_{1,2}^{(2)} \tilde{x}_1 \tilde{x}_2 + a_{3,4,5}^{(3)} \tilde{x}_3 \tilde{x}_4 \tilde{x}_5$$

model parameters $P = \binom{N+D-1}{D}$

M. Coutino, G. Karanikolas, G. Leus, and G. Giannakis, *"Self-Driven Graph Volterra Models for Higher-Order Link Prediction,"* ICASSP 2020, pp. 3887–3891, 2020.

The canonical polyadic Volterra model (CPVM)

Interaction tensor

$$\mathcal{Z} = \mathbf{z} \circ \cdots \circ \mathbf{z} \in \mathbb{R}^{N \times \cdots \times N}$$

with $\mathbf{z} = \begin{bmatrix} 1 \\ \tilde{\mathbf{x}} \end{bmatrix} \in \mathbb{R}^N$

CPVM using weight tensor

$$x = \langle \mathcal{W}, \mathcal{Z} \rangle_{\mathrm{F}}$$

Low-rank weight tensor

$$\mathcal{W} = \sum_{r=1}^{R} \mathbf{w}_r^{(1)} \circ \cdots \circ \mathbf{w}_r^{(D)} \in \mathbb{R}^{N \times \cdots \times N}$$

Example

$$x = a_5^{(1)} \tilde{x}_5 + a_{1,2}^{(2)} \tilde{x}_1 \tilde{x}_2 + a_{3,4,5}^{(3)} \tilde{x}_3 \tilde{x}_4 \tilde{x}_5$$

$$= \sum_{\pi \in \Pi(\{0,0,5\})} w_\pi \tilde{x}_5 + \sum_{\pi \in \Pi(\{0,1,2\})} w_\pi \tilde{x}_1 \tilde{x}_2 + \sum_{\pi \in \Pi(\{3,4,5\})} w_\pi \tilde{x}_3 \tilde{x}_4 \tilde{x}_5$$

model parameters $P = \binom{N+D-1}{D}$ ➡ N^D ➡ RDN

F. Wesel and K. Batselier, "*Large-Scale Learning with Fourier Features and Tensor Decompositions*," NeurIPS, vol. 34, pp. 17 543–17 554, 2021.

Methodology – (CPV)

Optimization problem

$$\min_{\mathcal{W}} \sum_{t=0}^{T-1} (x_t - \langle \mathcal{W}, \mathcal{Z}_t \rangle_{\mathrm{F}})^2 + \lambda \mathcal{R}(\mathcal{W})$$
$$\text{s.t.} \quad \text{CPD-rank}(\mathcal{W}) = R,$$

Solve D convex problems per iteration

$$\min_{\mathrm{vec}(\mathbf{W}^{(d)})} ||\mathbf{m} - \mathbf{G}^{(d)\top} \mathrm{vec}(\mathbf{W}^{(d)})||_2^2 + \lambda \mathcal{R}(\mathbf{W}^{(d)})$$

Take one factor matrix out

$$\langle \mathcal{W}, \mathcal{Z}_t \rangle_{\mathrm{F}} = [(\mathbf{W}^{(1)\top} \mathbf{z}_t \circledast \cdots \circledast \mathbf{W}^{(D)\top} \mathbf{z}_t) \otimes \mathbf{z}_t]^\top \mathrm{vec}(\mathbf{W}^{(d)})$$

For multiple observations of target node

$$\mathbf{m} \approx \mathbf{G}^{(d)\top} \mathrm{vec}(\mathbf{W}^{(d)})$$

$$\mathbf{m} = [x_0, \ldots x_{T-1}]^\top$$

$$\mathbf{G}^{(d)} = (\mathbf{W}^{(1)\top} \mathbf{Z} \circledast \cdots \circledast \mathbf{W}^{(D)\top} \mathbf{Z}) \odot \mathbf{Z}$$

<u>Computational complexity</u>

CPV $\mathcal{O}(DTN^2R^2 + DN^3R^3)$

AGVM solved using ridge regression (RV) $\mathcal{O}(TP^2 + P^3)$

AGVM solved using lasso regression (LV) $\mathcal{O}(kTP)$

F. Wesel and K. Batselier, "*Large-Scale Learning with Fourier Features and Tensor Decompositions*," NeurIPS, vol. 34, pp. 17 543–17 554, 2021.

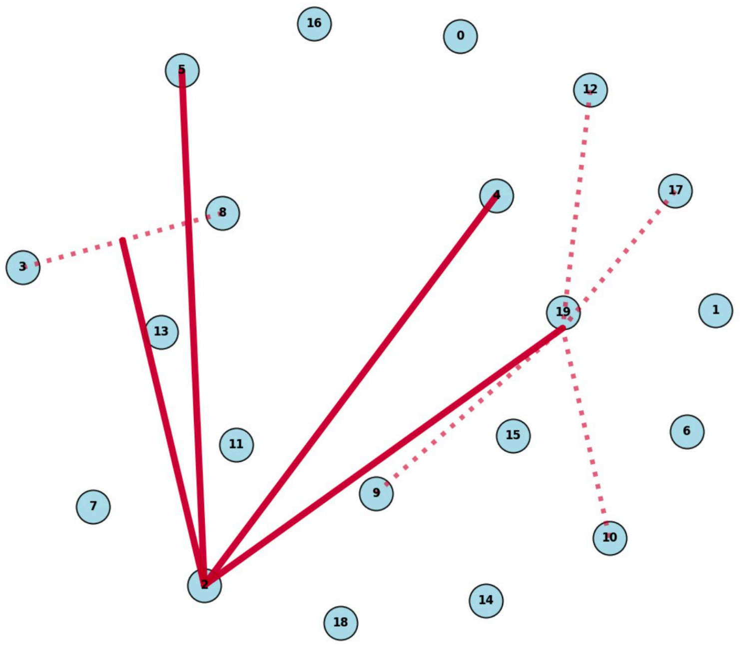
Simulations

$N = 20$ nodes

$R = 3.$ rank

$P = 8855$ model parameters

$D = 4$ order of interactions



$T = 5000$

Experiments

Co-authorship dataset consisting of 12499 papers, 17431 authors, 1903 keywords

R=4

Conclusion

The CPVM is a scalable model
Requires storing and estimating only RND parameters
The low-rank constraint acts as regularizer in ill-posed conditions

Future work
Account for temporal memory
Dynamic scenarios

TU Delft