Geometric Aspects of Mining Complex Networks

Leo Torres
PhD candidate
Network Science Institute, Northeastern University

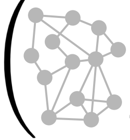
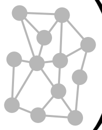
netsi

Geometric Aspects of Mining Complex Networks

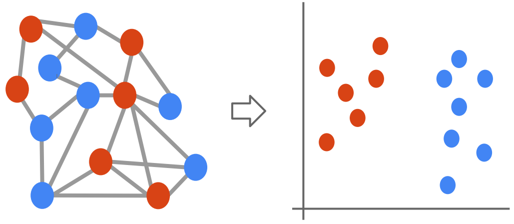
Geometric Aspects of Mining Complex Networks

What concepts and procedures can we take from geometry and topology and apply to mining and learning from complex networks?

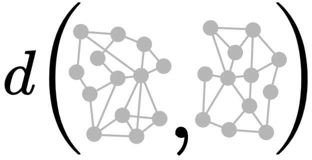
Distances

$d\Bigg($  ,  $\Bigg)$

Embeddings



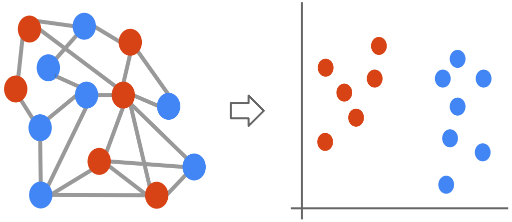
Distances



Non-backtracking cycles: length spectrum theory and graph mining applications

Torres, L., Suárez-Serrato, P. and Eliassi-Rad, T. Appl Netw Sci (2019) 4: 41.

Embeddings

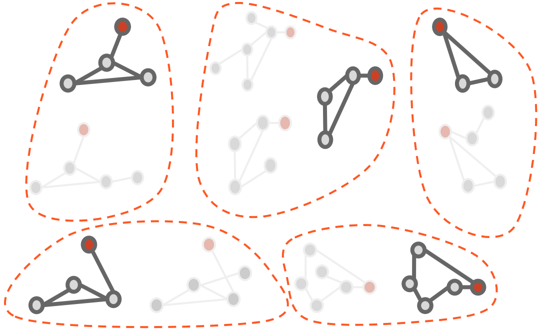


GLEE: Geometric Laplacian Eigenmap Embedding

Torres, L., Chan, K. S. and Eliassi-Rad, T. Journal of Complex Networks, Volume 8, Issue 2, April 2020, cnaa007.

NBD: Non-Backtracking Distance

Pablo Suárez-Serrato, UNAM



Tina Eliassi-Rad, NEU

Torres, L., Suárez-Serrato, P. and Eliassi-Rad, T. **Non-backtracking cycles: length spectrum theory and graph mining applications**. Appl Netw Sci (2019) 4: 41.

Supported by **NSF** CNS-1314603, **NSF** IIS-1741197, and **DTRA** HDTRA1-10-1-0120.

Spoiler Alert!

Networks: the non-backtracking eigenvalues track important descriptors like degree distribution and triangles.

Machine Learning: non-backtracking eigenvalues are a great way of measuring distance.

Mathematics: the length spectrum of an unweighted graph characterizes its 2-core uniquely up to isomorphism.

Spoiler Alert!

Networks: the non-backtracking eigenvalues track important descriptors like degree distribution and triangles.

Machine Learning: non-backtracking eigenvalues are a great way of measuring distance.

Mathematics: the length spectrum of an unweighted graph characterizes its 2-core uniquely up to isomorphism.

Spoiler Alert!

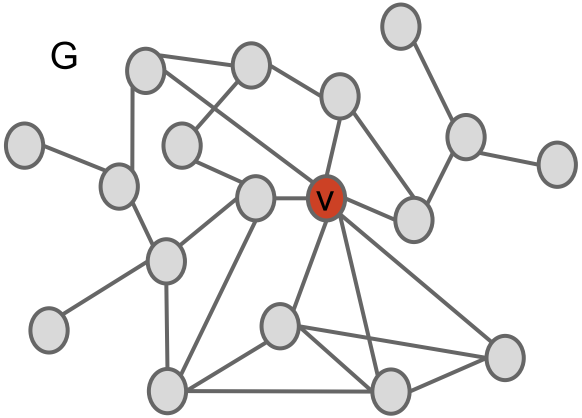
Networks: the non-backtracking eigenvalues track important descriptors like degree distribution and triangles.

Machine Learning: non-backtracking eigenvalues are a great way of measuring distance.

Mathematics: the length spectrum of an unweighted graph characterizes its 2-core uniquely up to isomorphism.

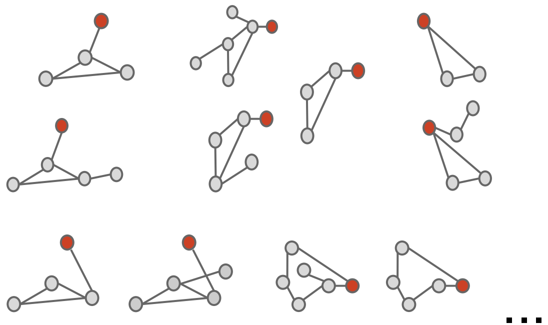
The Length Spectrum

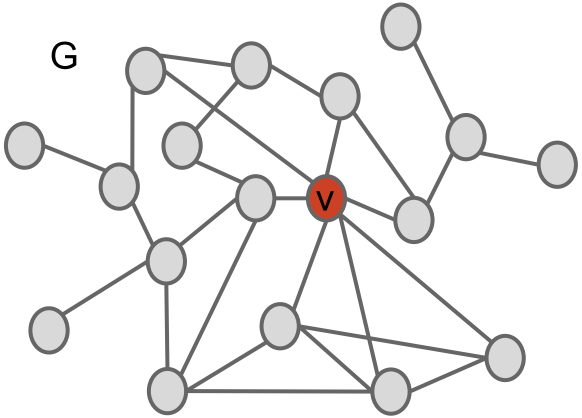
1. Given a graph G = (V, E) and a node v,



The Length Spectrum

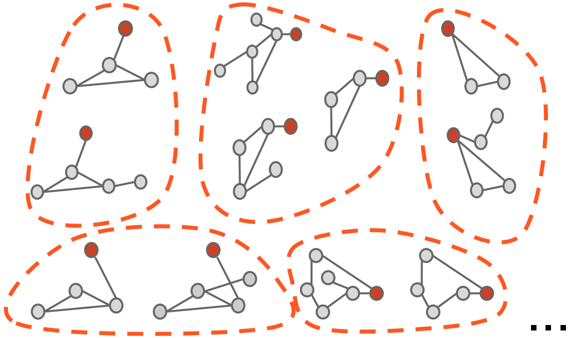
1. Given a graph G = (V, E) and a node v, consider the set of all closed walks that start and end at v.

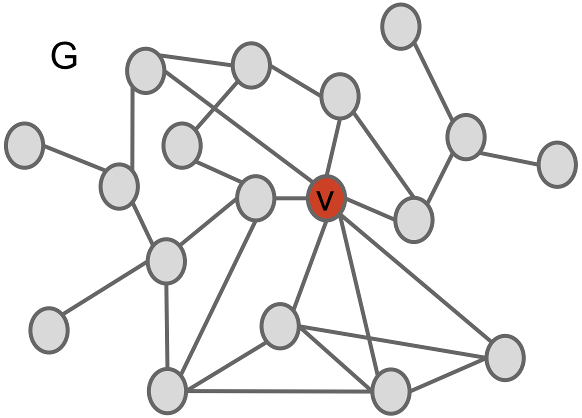




The Length Spectrum

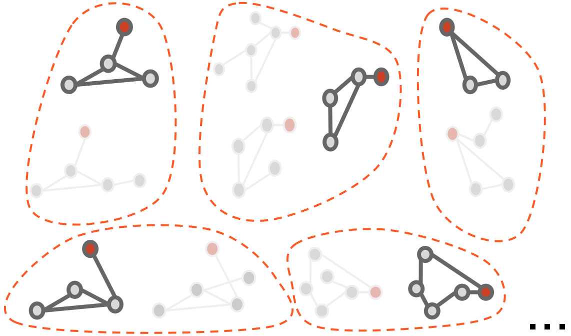
2. Walks are equivalent if they are equal save for **tree-like parts that don't go through the basepoint…**

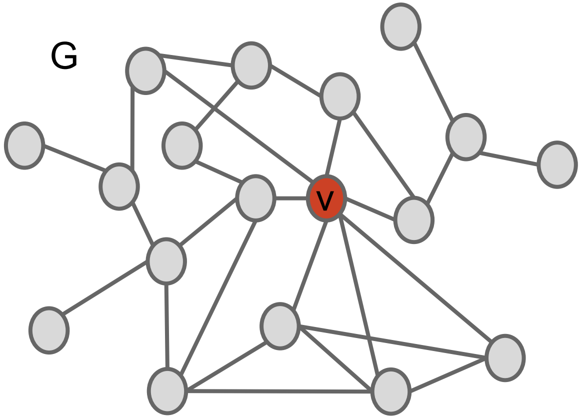




The Length Spectrum

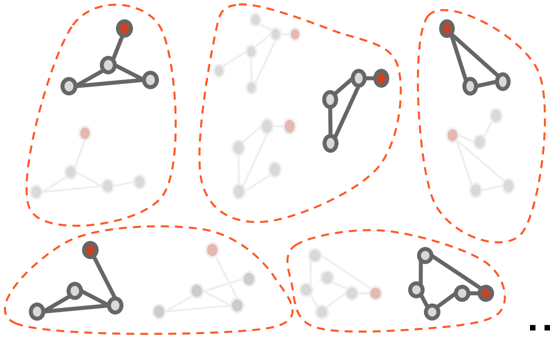
2. … and retain the shortest walk in each subset.





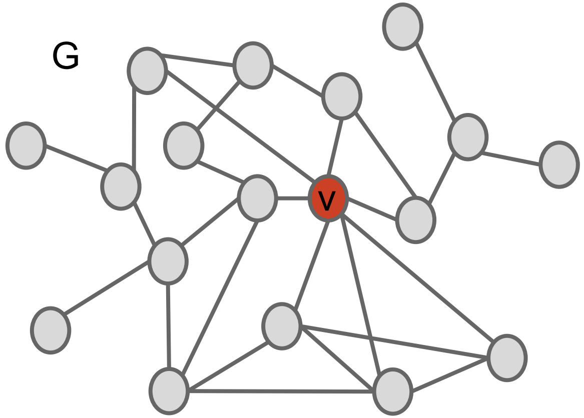
The Length Spectrum

2. This set is the fundamental group of G with basepoint v.



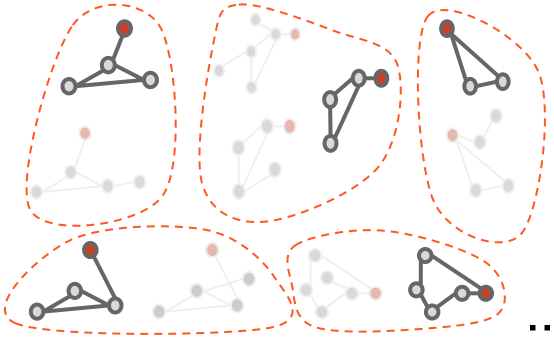
...

$$\pi_1(G, v)$$



The Length Spectrum

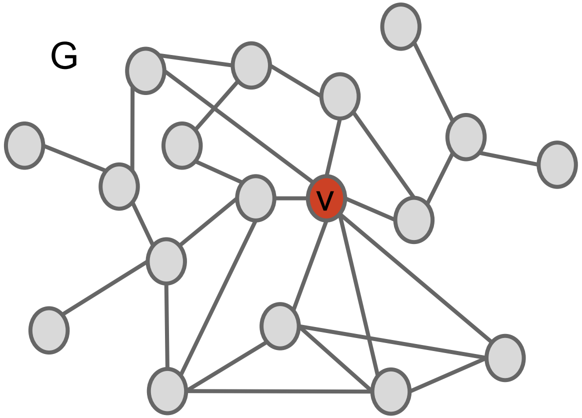
3. Walks are equivalent if they are equal **save for tree-like parts** ~~that don't go through the basepoint~~.

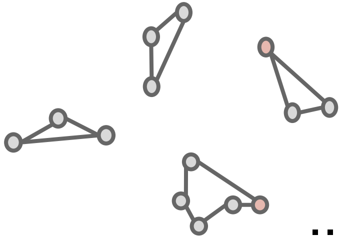


...

$$\pi_1(G, v)$$

G

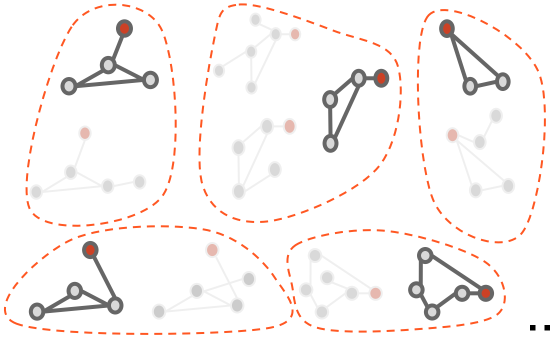




...

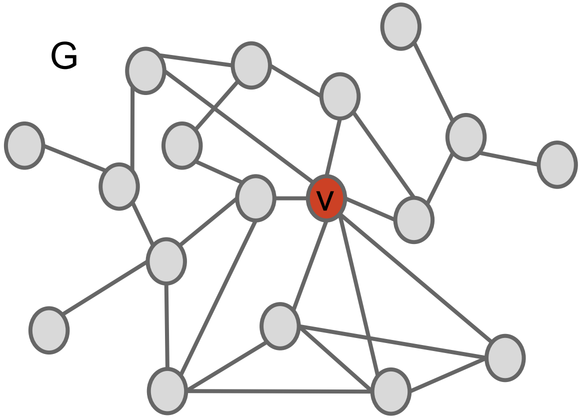
The Length Spectrum

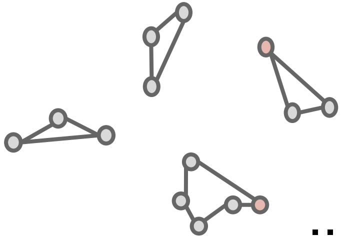
3. This is the set of non-backtracking cycles (NBCs) of G.



...

$\pi_1(G, v)$



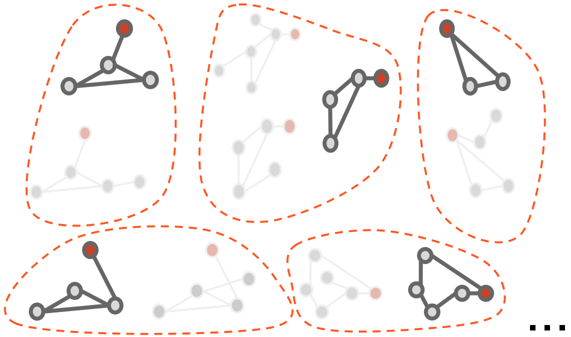


...

$NBCs$

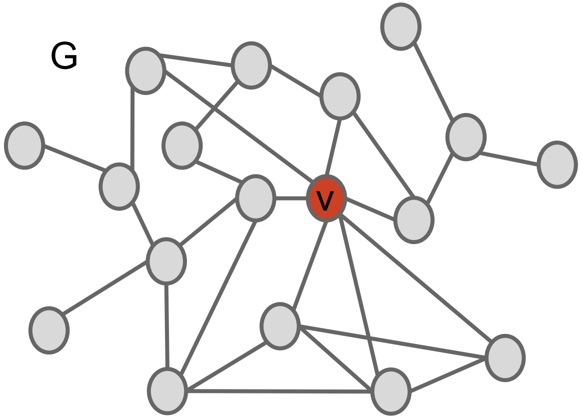
The Length Spectrum

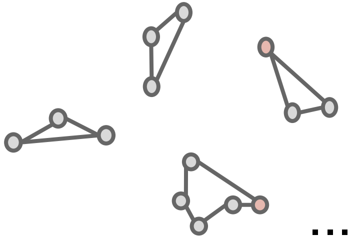
4. $\mathcal{L}$ is defined on $\pi_1(G, v)$ and assigns each walk the length of its “shaved” version.



$\pi_1(G, v)$

$$\boxed{\mathcal{L} : \pi_1(G, v) \to \mathbb{R}}$$

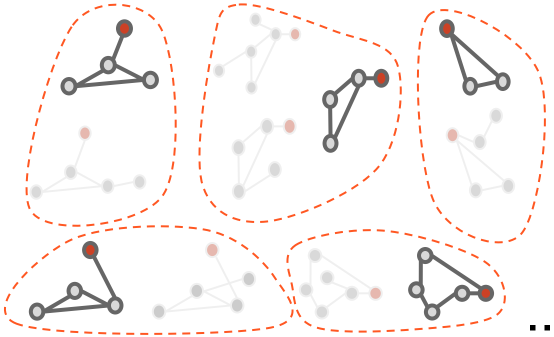




$NBCs$

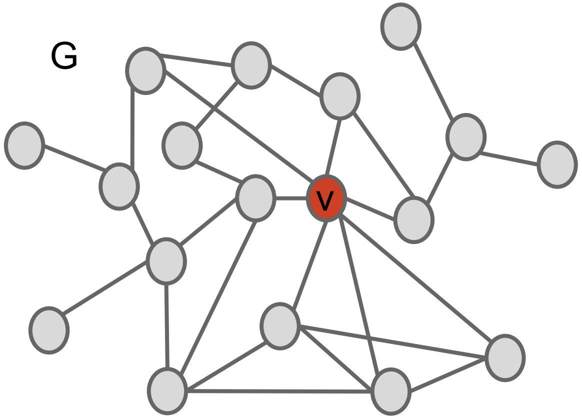
The Length Spectrum

4. $\mathcal{L}$ is defined on $\pi_1(G, v)$ and assigns each walk the length of its “shaved” version.

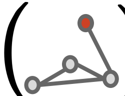


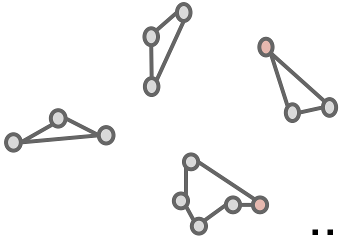
...

$\pi_1(G, v)$



$$\boxed{\mathcal{L} : \pi_1(G, v) \to \mathbb{R}}$$

$\mathcal{L}($  $) = 3$



...

$NBCs$

The Length Spectrum

The Length Spectrum of a graph characterizes **its 2-core** uniquely up to isomorphism.

Constantine, D., and Lafont, J.-F. **Marked Length Rigidity for One-Dimensional Spaces.** Journal of Topology and Analysis, 2018.

Modifying the Length Spectrum

$$d(G, H) = d(\mathcal{L}_G, \mathcal{L}_H)$$

Modifying the Length Spectrum

$$d(G, H) = d(\mathcal{L}_G, \mathcal{L}_H)$$

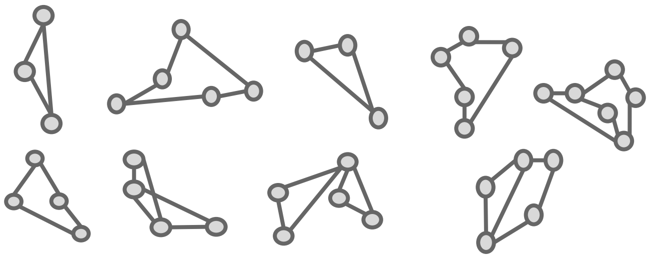
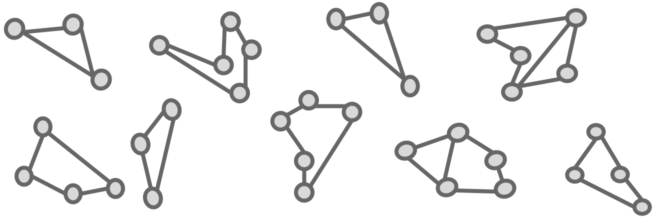
Two assumptions	Two problems
$G \to \mathcal{L}_G$	How to compute?
$d(\mathcal{L}_G, \mathcal{L}_H)$	How to compare?

Modifying the Length Spectrum

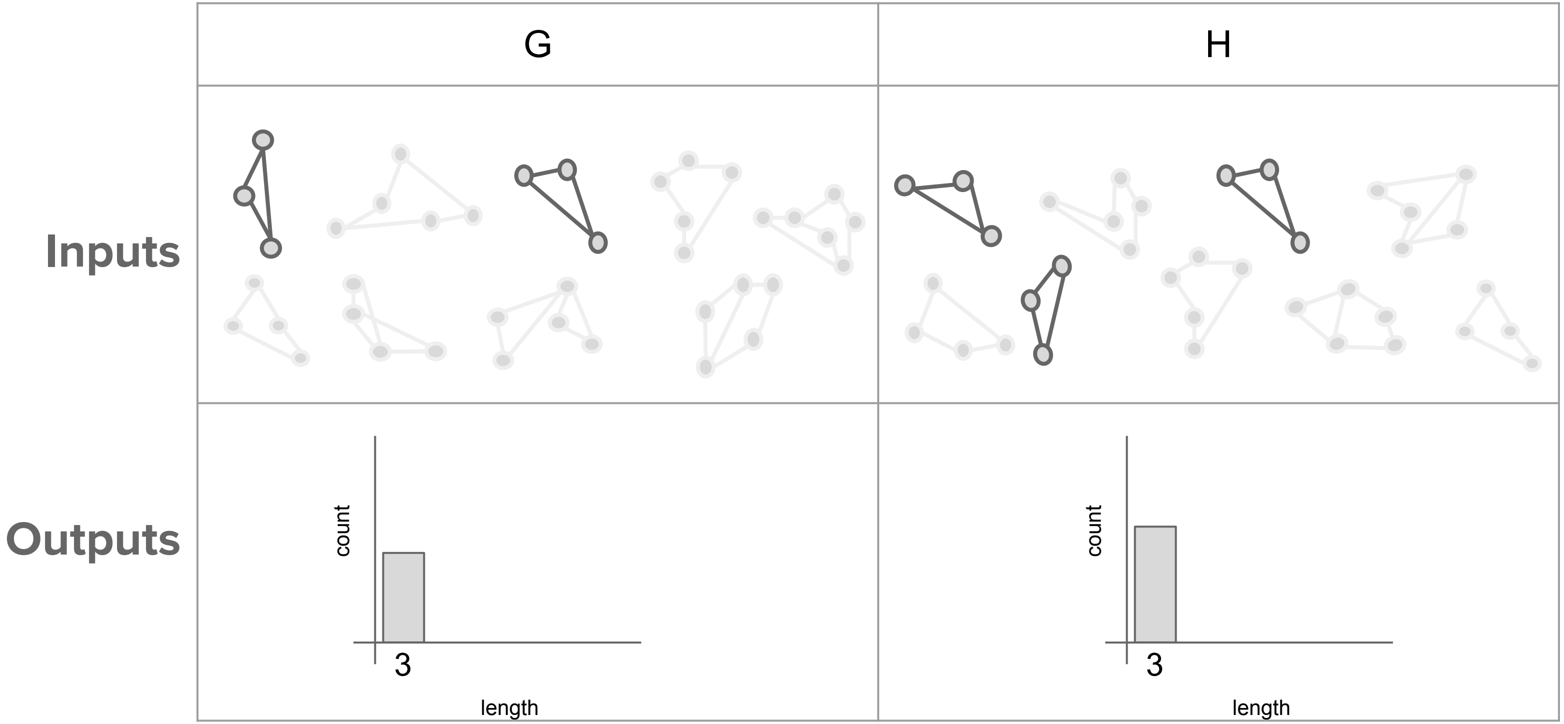
$$d(G, H) = d(\mathcal{L}_G, \mathcal{L}_H)$$

Two assumptions	Two problems	Two solutions
$G \to \mathcal{L}_G$	How to compute?	Outputs instead of inputs
$d(\mathcal{L}_G, \mathcal{L}_H)$	How to compare?	Partition the set of outputs

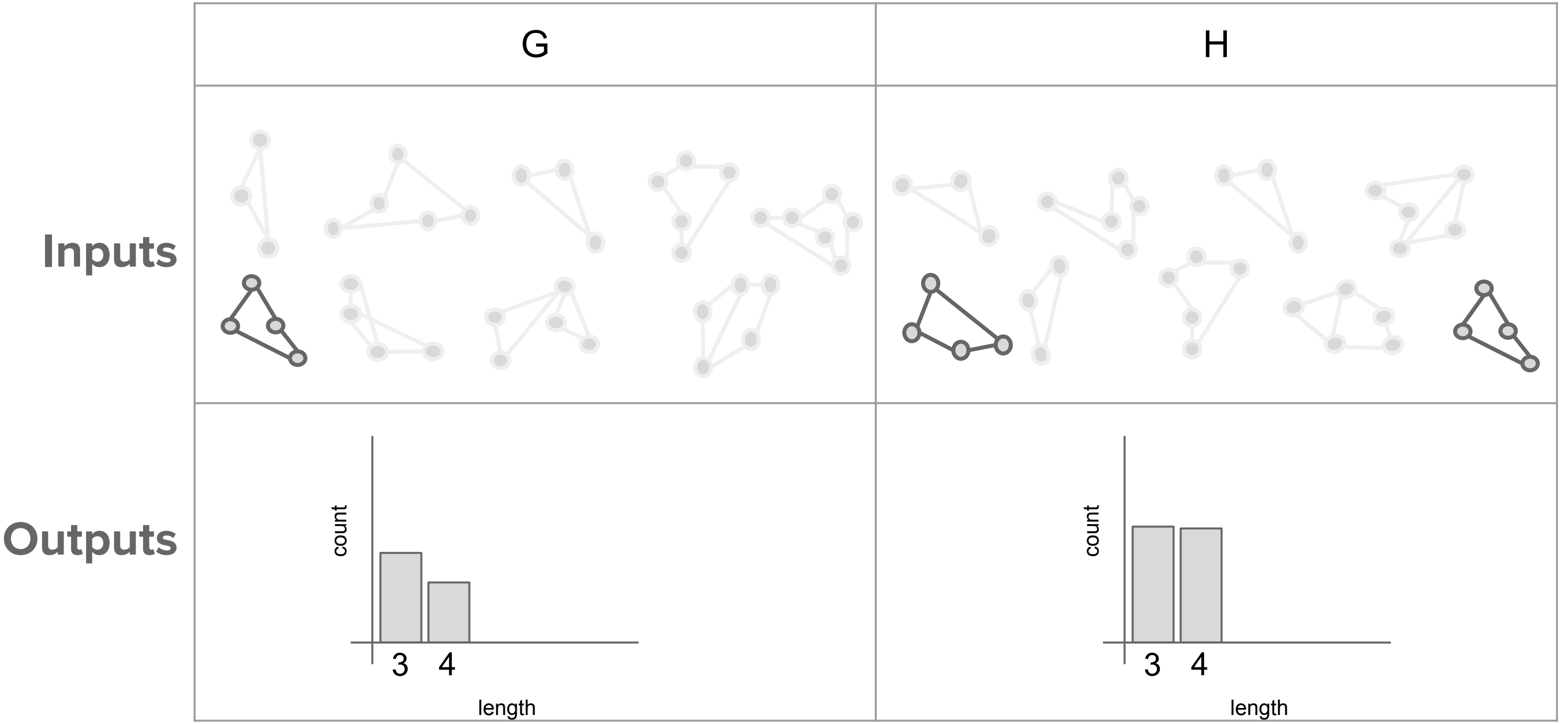
Modifying the Length Spectrum

	G	H
Inputs		
Outputs		

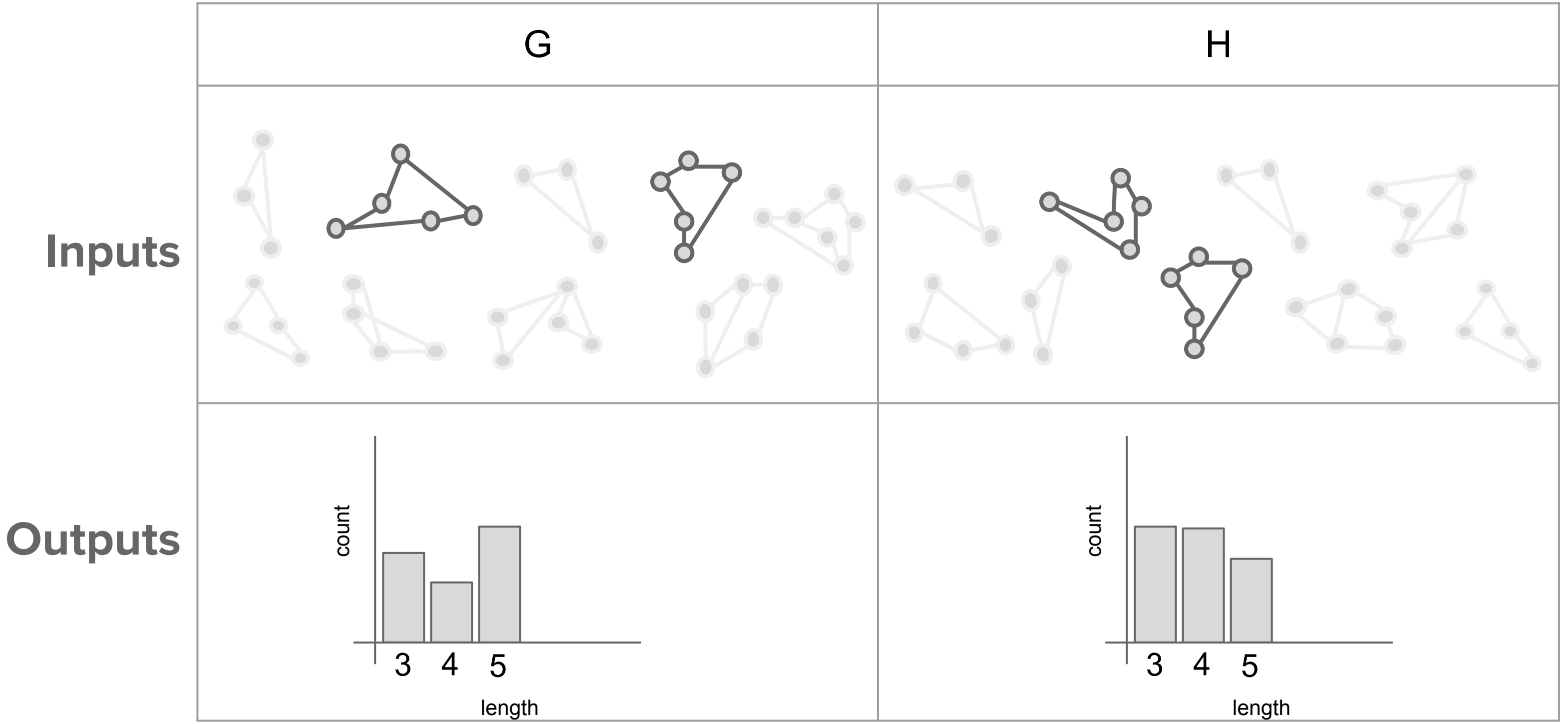
Modifying the Length Spectrum



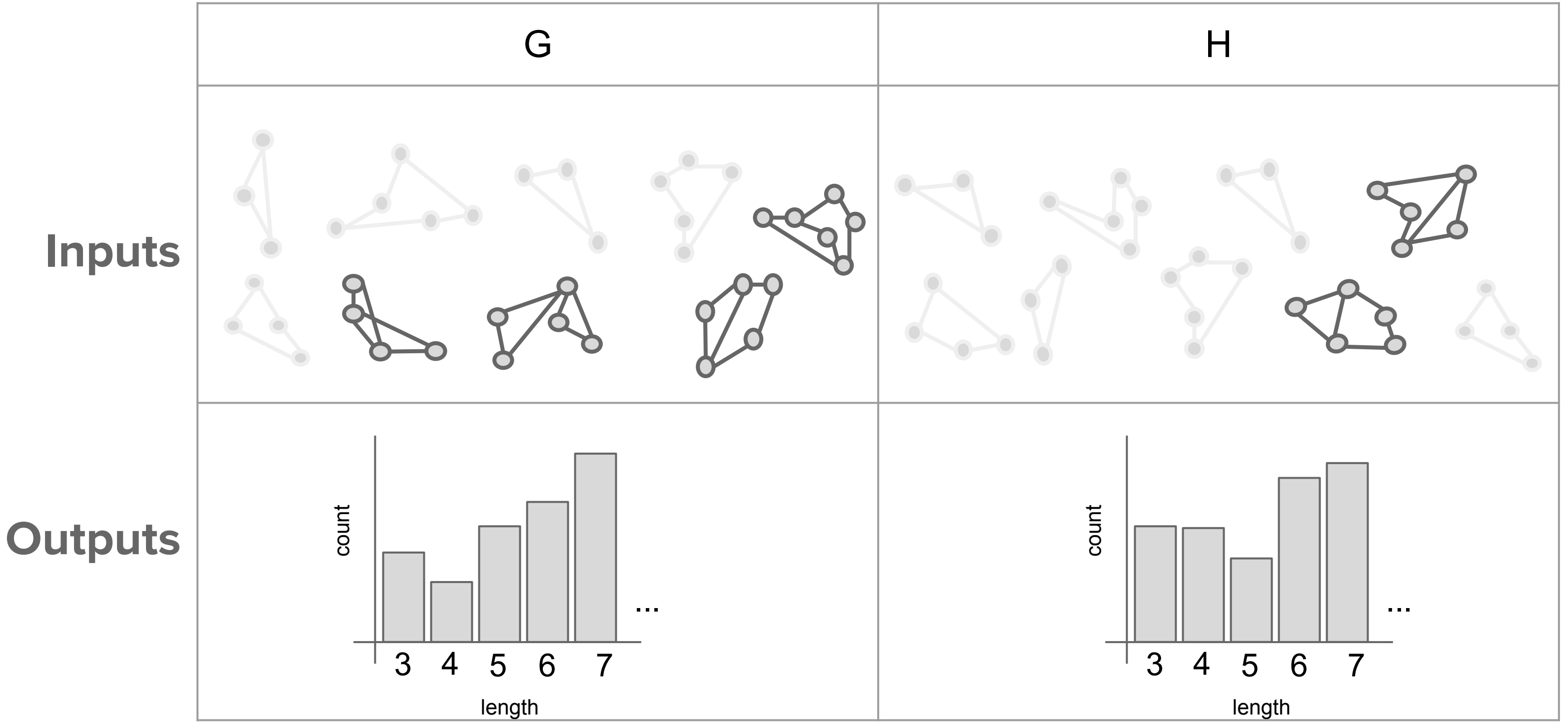
Modifying the Length Spectrum



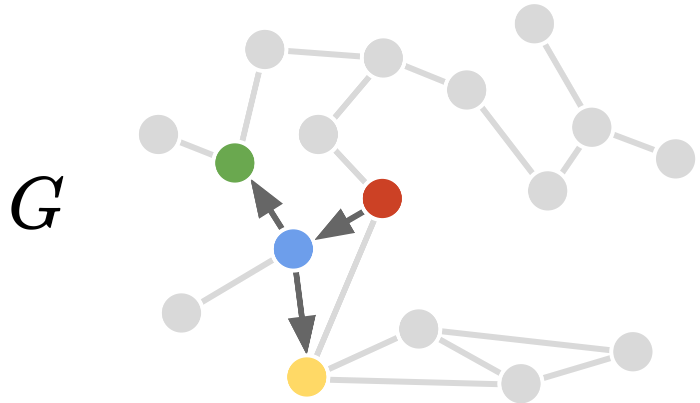
Modifying the Length Spectrum



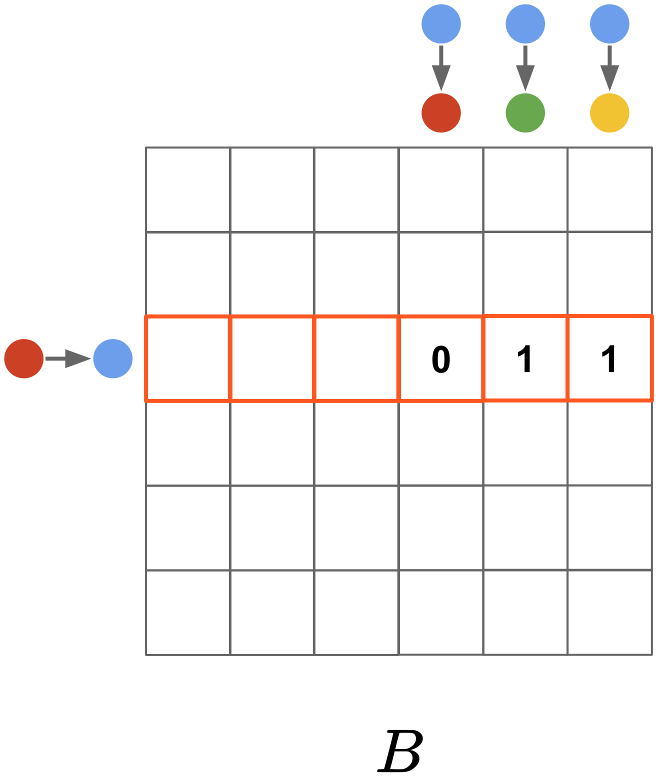
Modifying the Length Spectrum



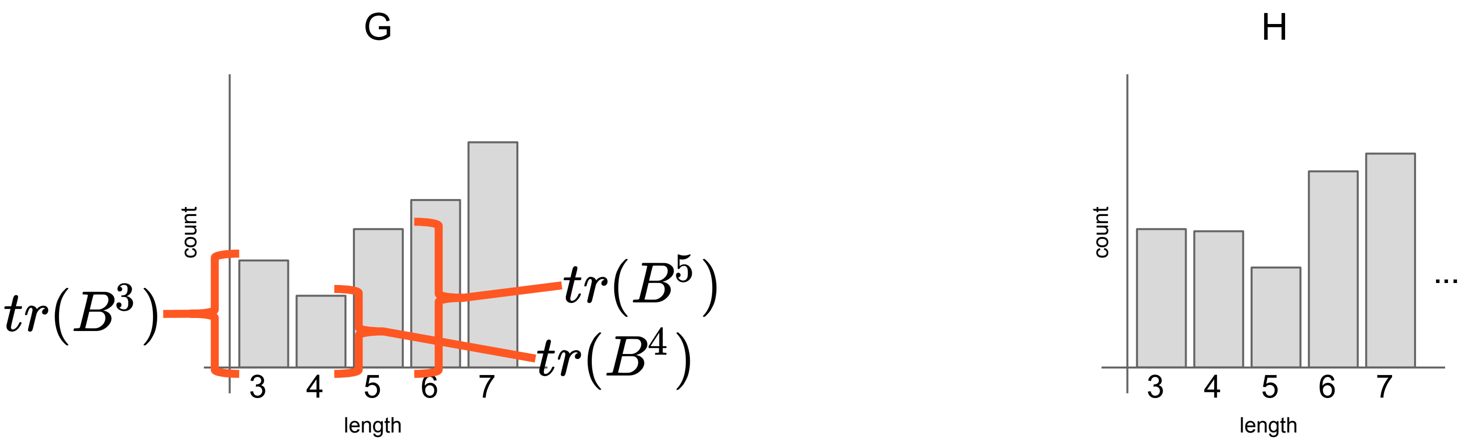
Non-backtracking matrix



$$G = (V, E)$$
$$|E| = m$$



Graph Distance



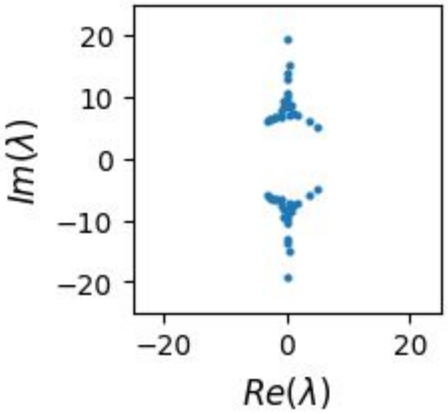
Graph Distance

Given two graphs G, H *and an integer* r*, write* λ_k, μ_k *for the eigenvalues of their corresponding non-backtracking matrices,* $k = 1, 2, \ldots, r$*. Let* Λ *and* M *be the cumulative density function of the respective spectral densities. Define the distance between G and H by*

$$\boxed{d_r(G, H) = \sqrt{\iint |\Lambda(x, y) - M(x, y)|^2 \, dx dy}}$$

Properties: hubs

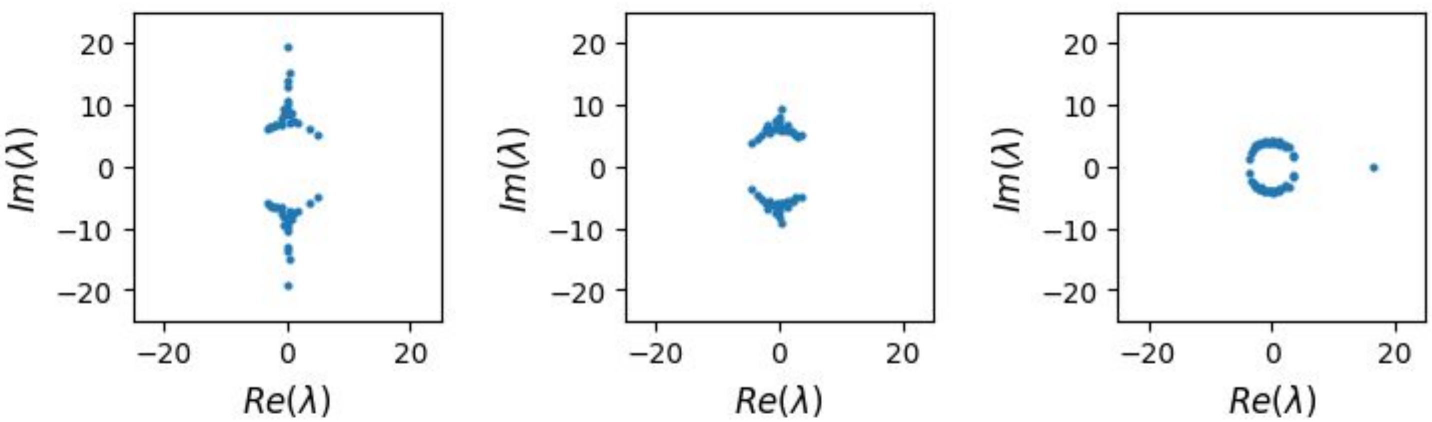
Configuration model ($n = 10k, \langle k \rangle = 10, \gamma = 2.1$)



fewer hubs

Properties: hubs are imaginary

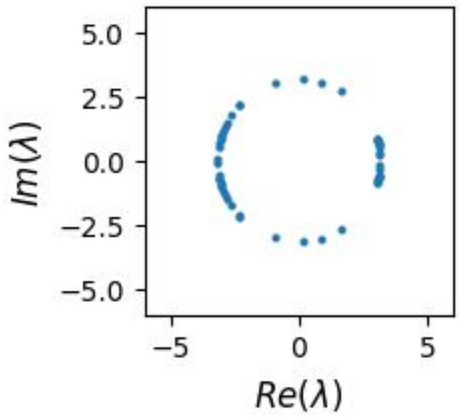
Configuration model ($n = 10k, \langle k \rangle = 10, \gamma = 2.1$)



fewer hubs

Properties: triangles

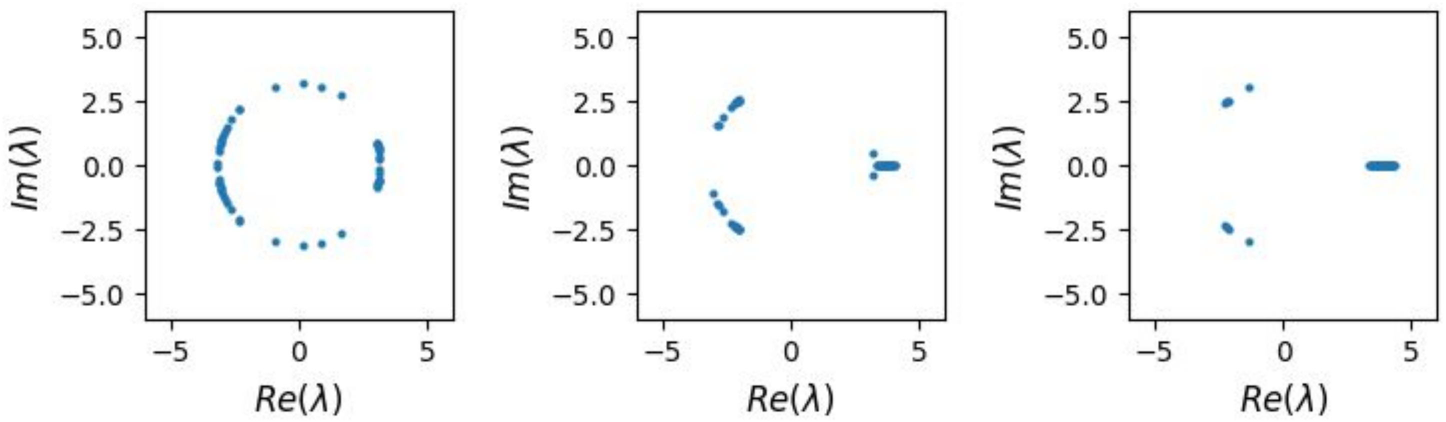
ER graph ($n = 10k$, $p = 0.001$)



more triangles

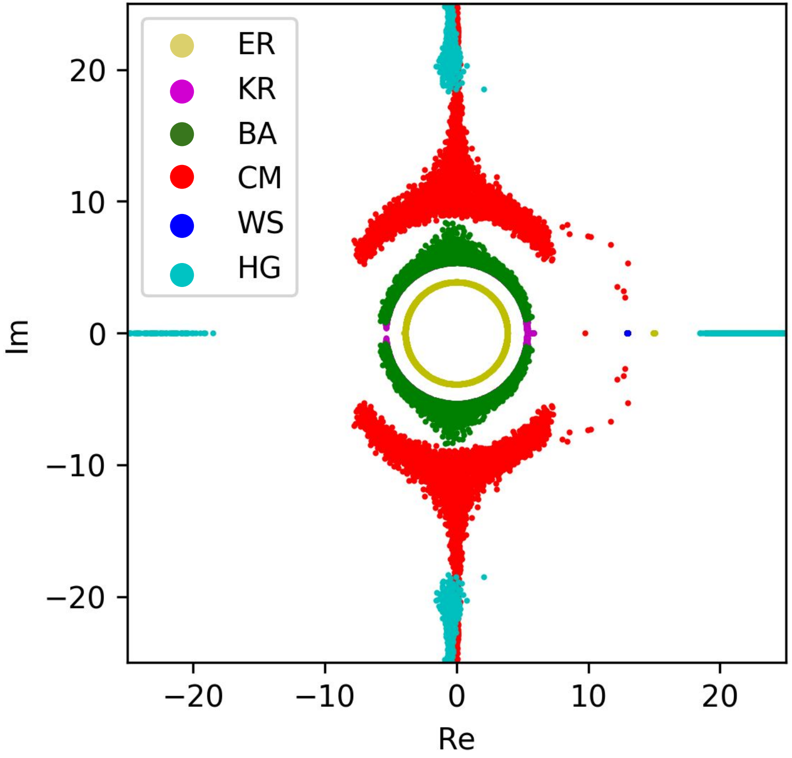
Properties: triangles

ER graph (n = 10k, p = 0.001)

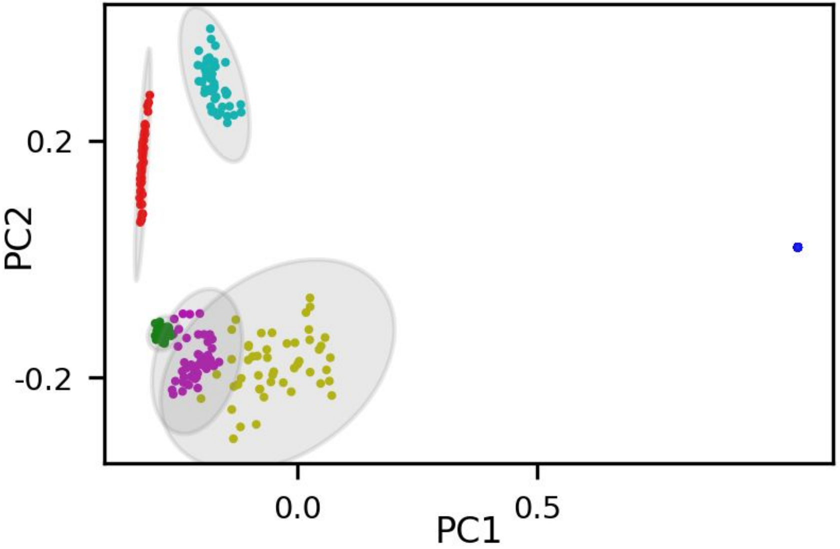


more triangles

Examples: clustering

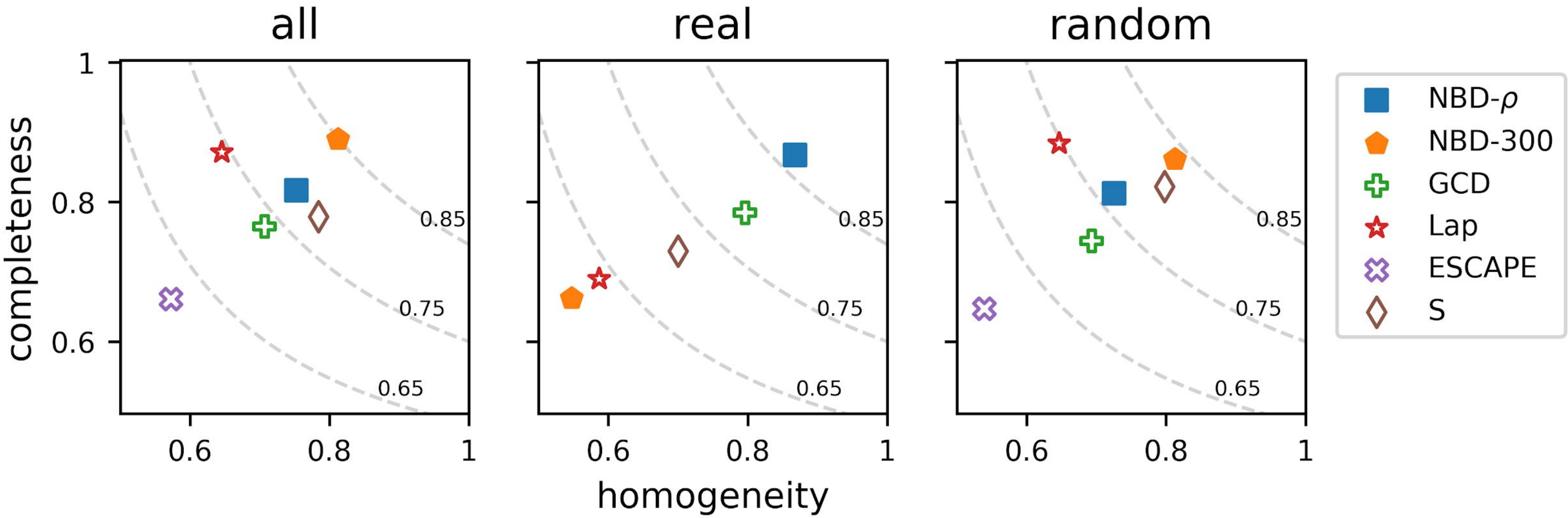


1 dot = 1 eigenvalue



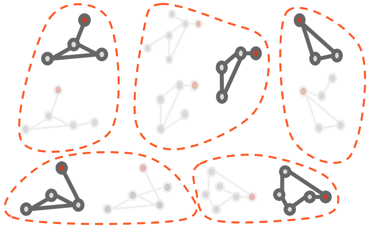
1 dot = 1 graph

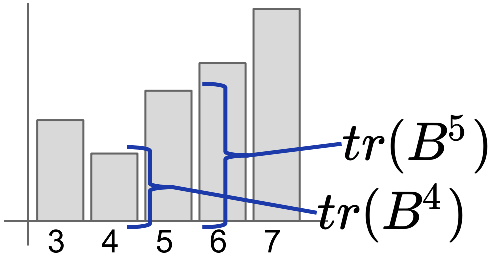
Examples: clustering

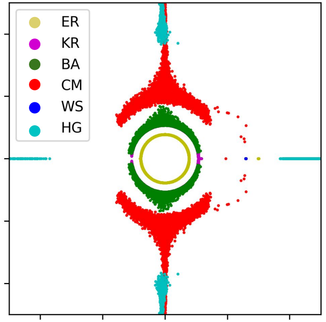


Summary

The non-backtracking eigenvalues track descriptors like degree distribution and triangles and can find patterns and anomalies; THEREFORE they are a great way of measuring distance BECAUSE they contain similar information to the length spectrum, which characterizes the 2-core of an unweighted graph uniquely.



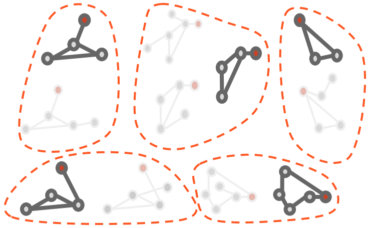


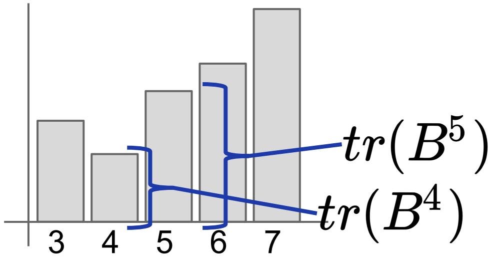


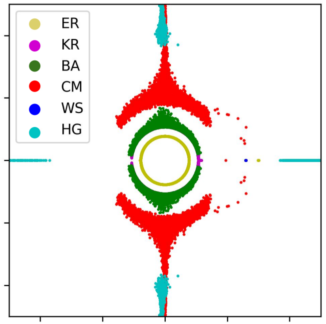
Supported by **NSF** CNS-1314603, **NSF** IIS-1741197, and **DTRA** HDTRA1-10-1-0120.

Summary: geometry

- The derivation and algorithm are based on **algebraic topology**.
  - Intrinsic topology/geometry of each graph.
- The set of eigenvalues can be considered as a form of "**graph embedding**".
  - Geometry of the set of all graphs, as represented by their eigenvalues.



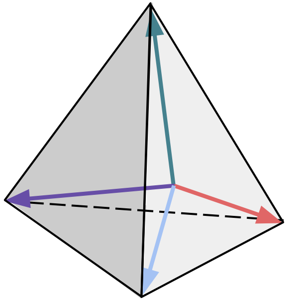


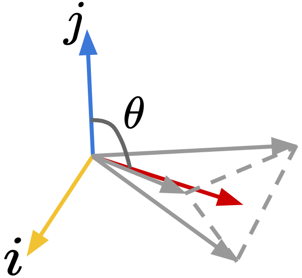


Supported by **NSF** CNS-1314603, **NSF** IIS-1741197, and **DTRA** HDTRA1-10-1-0120.

GLEE: Geometric Laplacian Eigenmap Embedding

Kevin S. Chan, ARL





Tina Eliassi-Rad, NEU

Leo Torres, K. S. Chan and T. Eliassi-Rad. **GLEE: Geometric Laplacian Eigenmap Embedding**. J. of Comp. Net., Volume 8, Issue 2, April 2020, cnaa007.

Work supported by **NSF** CNS-1314603, **NSF** IIS-1741197, Army Research Laboratory **Cooperative Agreement** W911NF-13-2-0045.

Spoiler Alert!

Mathematics: there is a bijection between undirected graphs on n nodes and n-1 dimensional simplices.

Networks: we can encode graph structure in geometric terms using the simplex geometry of the Laplacian.

Machine Learning: some embedding methods perform well only when clustering coefficient is high.

Spoiler Alert!

Mathematics: there is a bijection between undirected graphs on n nodes and n-1 dimensional simplices.

Networks: we can encode graph structure in geometric terms using the simplex geometry of the Laplacian.

Machine Learning: some embedding methods perform well only when clustering coefficient is high.

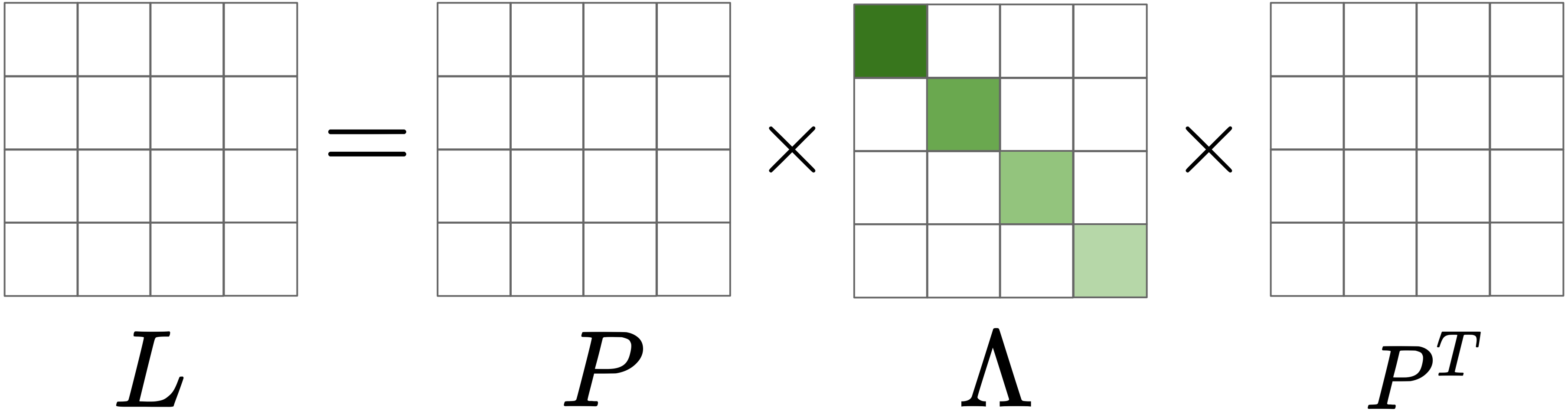
Spoiler Alert!

Mathematics: there is a bijection between undirected graphs on n nodes and n-1 dimensional simplices.

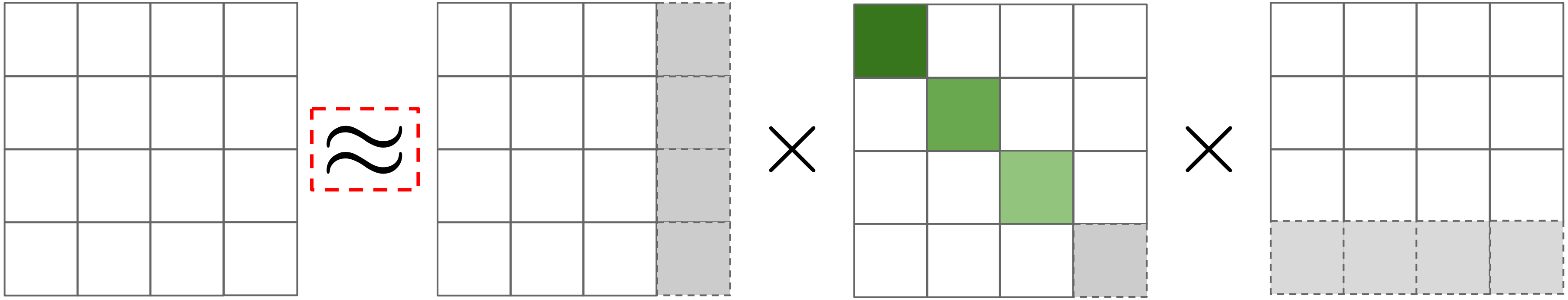
Networks: we can encode graph structure in geometric terms using the simplex geometry of the Laplacian.

Machine Learning: some embedding methods perform well only when clustering coefficient is high.

GLEE: Geometric Laplacian Eigenmap Embedding



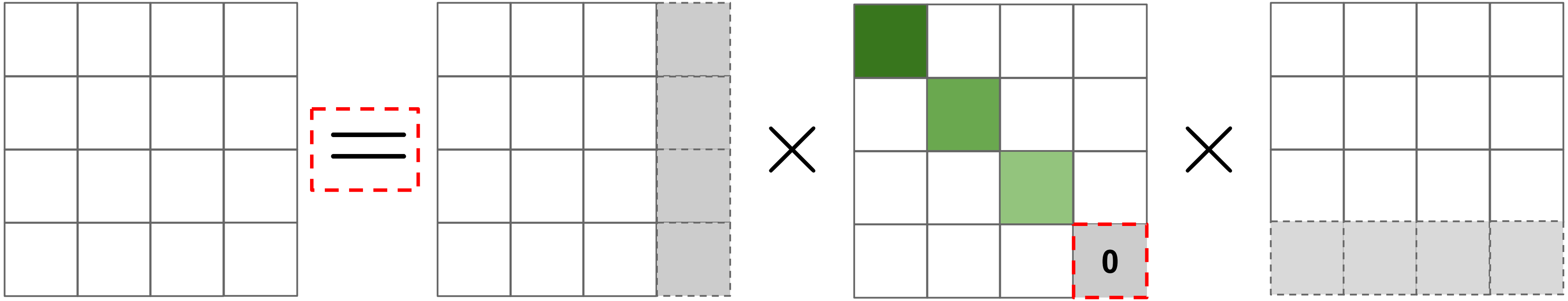
GLEE: Geometric Laplacian Eigenmap Embedding



$$L \approx P \times \Lambda \times P^T$$

Singular Value Decomposition says that eliminating the rows and columns corresponding to the lowest singular values give a good approximation of ***L***.

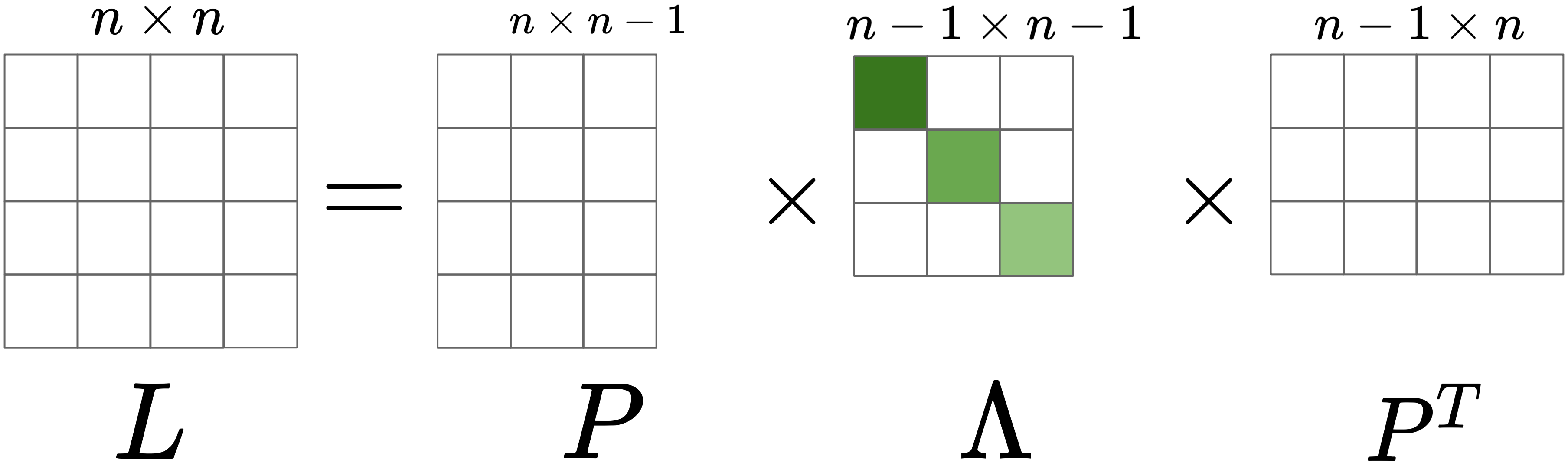
GLEE: Geometric Laplacian Eigenmap Embedding



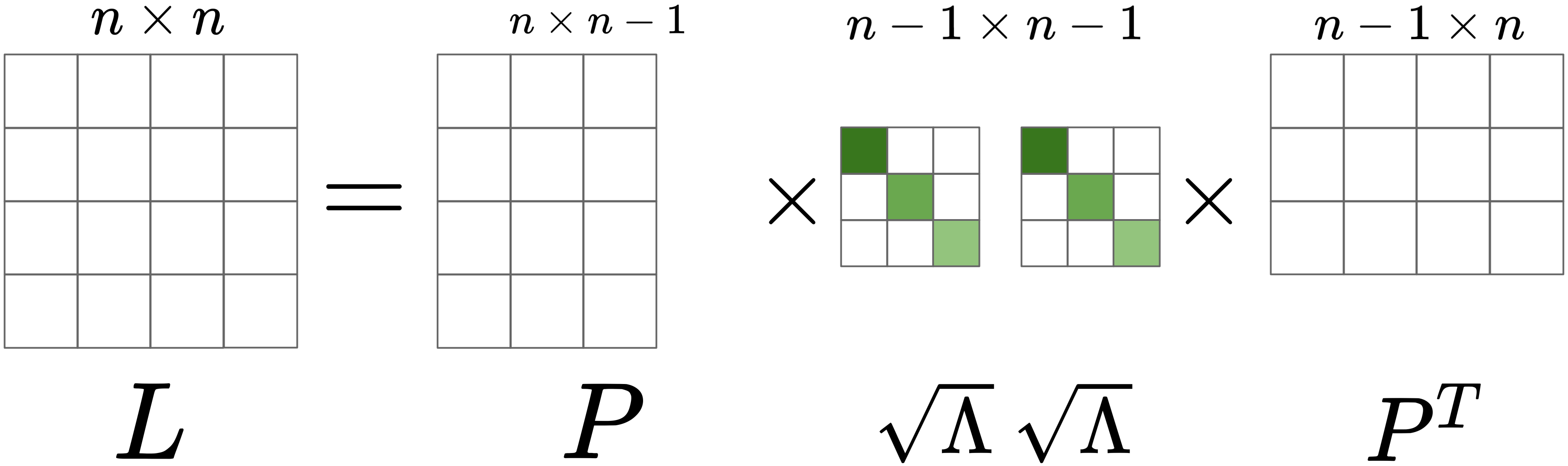
$$L = P \times \Lambda \times P^T$$

However, the last eigenvalue of ***L*** is always **0**, which implies exact equality.

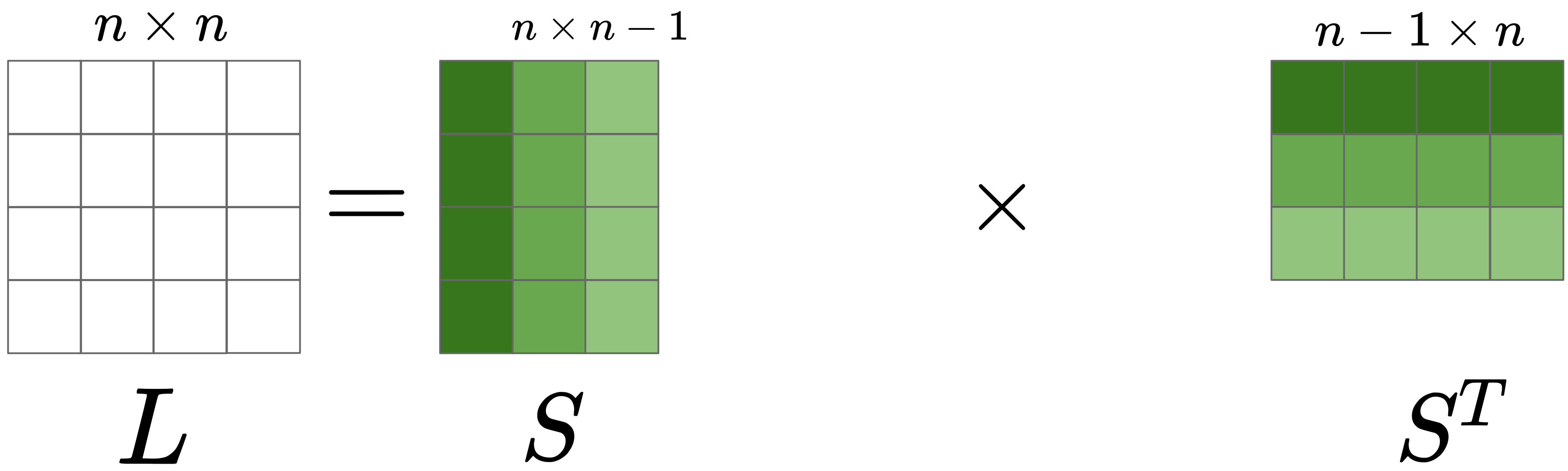
GLEE: Geometric Laplacian Eigenmap Embedding



GLEE: Geometric Laplacian Eigenmap Embedding

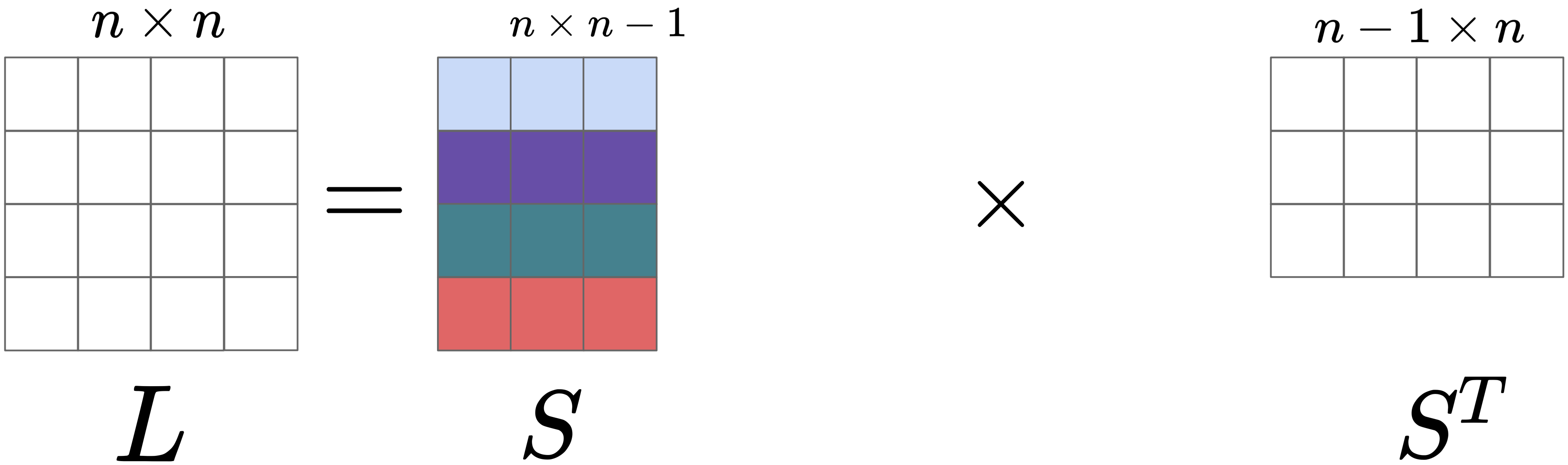


GLEE: Geometric Laplacian Eigenmap Embedding



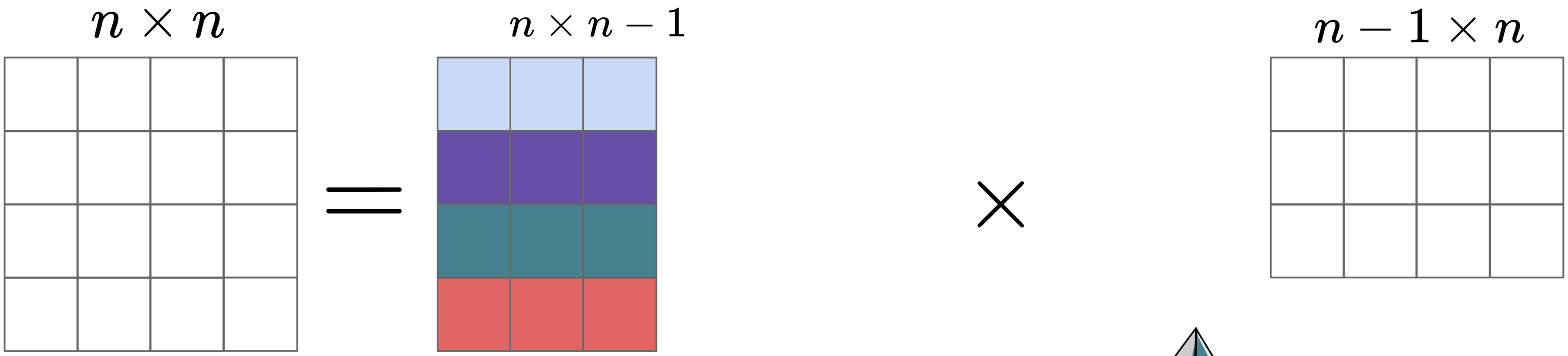
$S = P\sqrt{\Lambda}$

GLEE: Geometric Laplacian Eigenmap Embedding

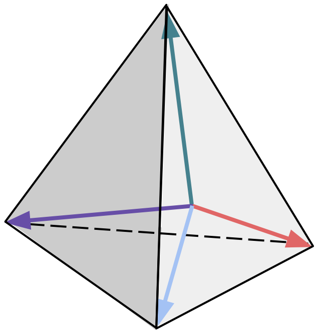


In a connected graph, ***L*** has rank ***n-1,*** and only one eigenvalue equal to ***0.***
This implies that ***S*** has full rank, i.e., rank ***n-1.***

GLEE: Geometric Laplacian Eigenmap Embedding

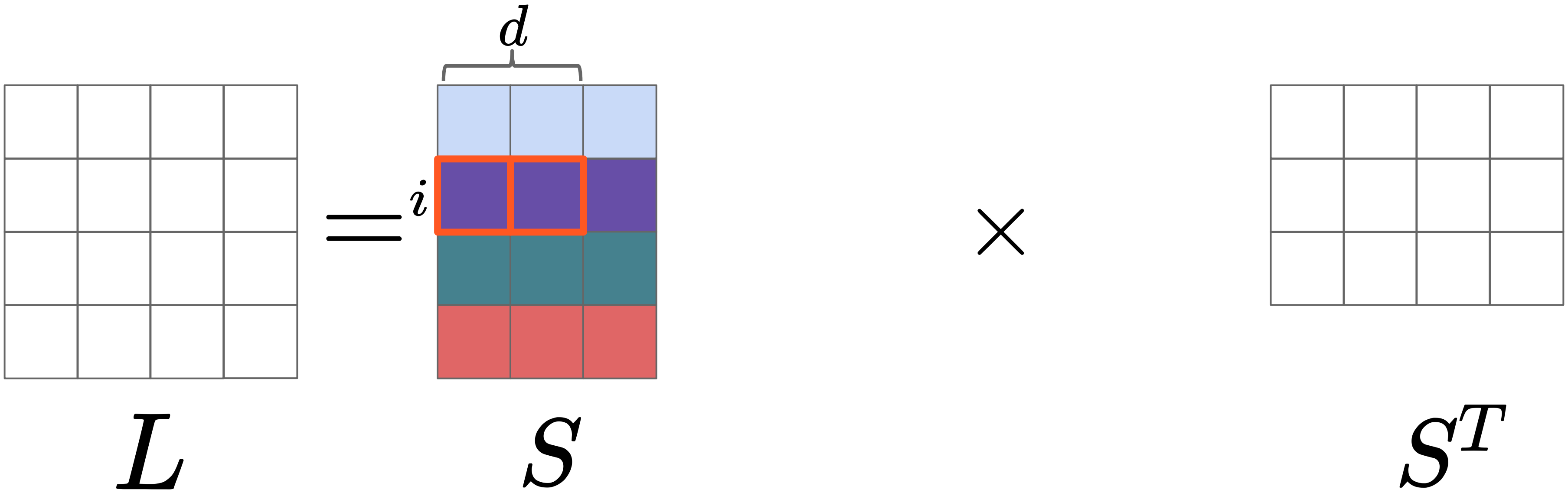


This implies that the rows of ***S*** point to the vertices of an **(n-1)**-D **simplex**.



K. Devriendt and P. Van Mieghem. **The simplex geometry of graphs**. The Journal of Complex Networks, 2019.

GLEE: Geometric Laplacian Eigenmap Embedding



Given a graph **G = (V, E)**, define the ***d***-dimensional **GLEE** of a node ***i*** as the first ***d*** columns of the ***i***-th row of $\boldsymbol{S} = \boldsymbol{P}\boldsymbol{\Lambda}^{1/2}$, and is denote it by $\boldsymbol{s}_i$.

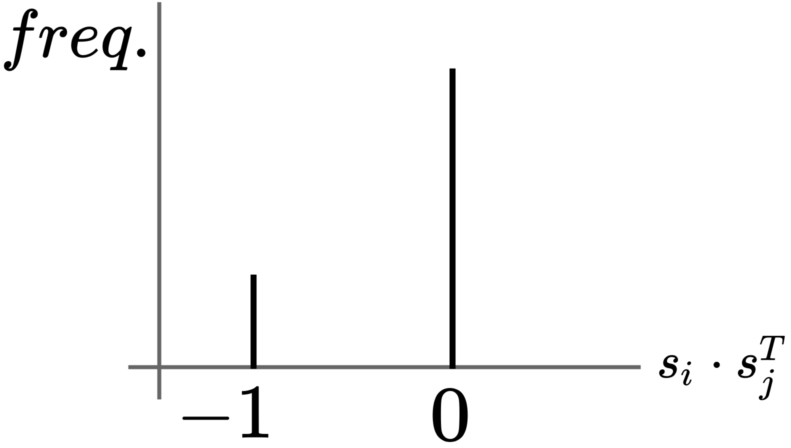
Graph Reconstruction

Given the matrix $\boldsymbol{S}$ whose rows are $\boldsymbol{s}_i$, how do we reconstruct the graph?

Graph Reconstruction

Given the matrix ***S*** whose rows are $\boldsymbol{s}_i$, how do we reconstruct the graph?

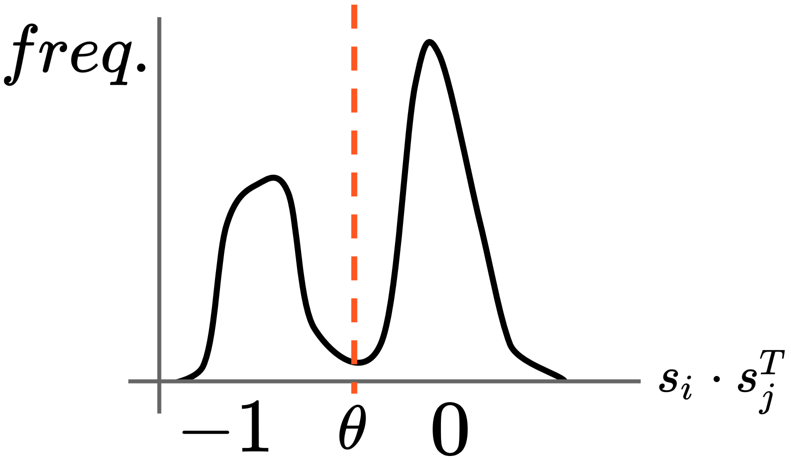
- Assume ***d = n-1***. In this case, we simply have $\boldsymbol{L} = \boldsymbol{S}\,\boldsymbol{S}^T$.



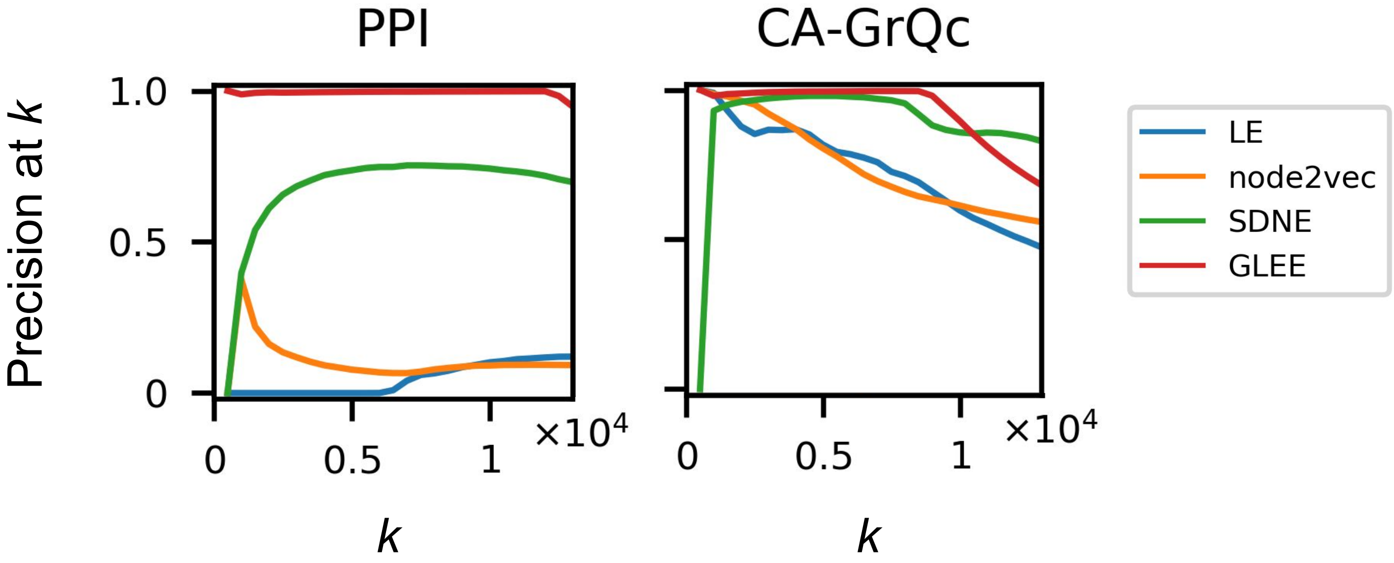
Graph Reconstruction

Given the matrix ***S*** whose rows are $\boldsymbol{s}_i$, how do we reconstruct the graph?

- Assume ***d = n-1***. In this case, we simply have $\boldsymbol{L} = \boldsymbol{S}\,\boldsymbol{S}^T$.
- If ***d < n***, then $\boldsymbol{S}\,\boldsymbol{S}^T$ is the best rank-d approximation of ***L***.



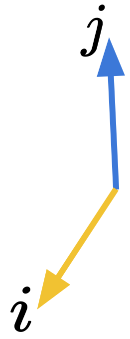
Graph Reconstruction: results



Same embedding dimension, similar network size, but **different average clustering**.

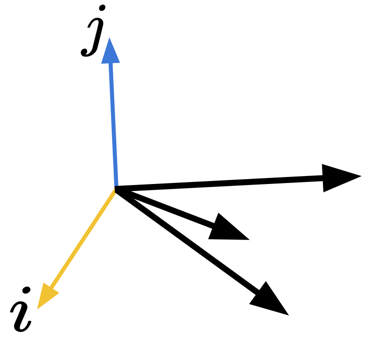
Link Prediction: common neighbors

In many networks (e.g. social networks), the number of common neighbors is an excellent predictor of links because of **triadic closure**.



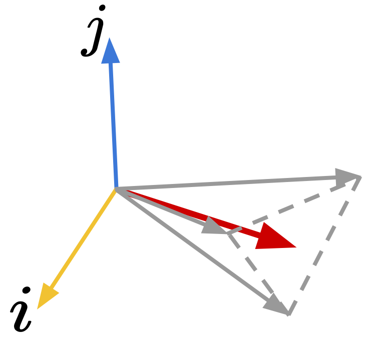
Link Prediction: common neighbors

In many networks (e.g. social networks), the number of common neighbors is an excellent predictor of links because of **triadic closure**.



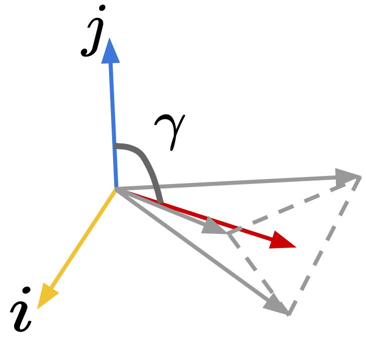
Link Prediction: common neighbors

In many networks (e.g. social networks), the number of common neighbors is an excellent predictor of links because of **triadic closure**.



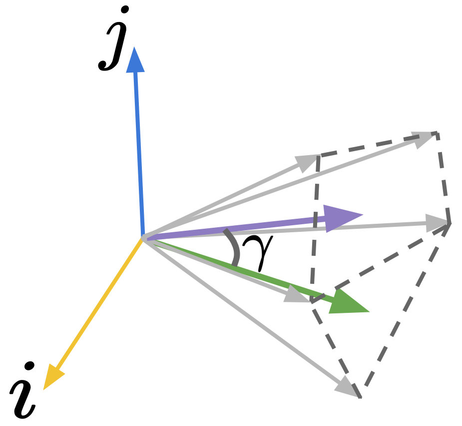
Link Prediction: common neighbors

In many networks (e.g. social networks), the number of common neighbors is an excellent predictor of links because of **triadic closure**.

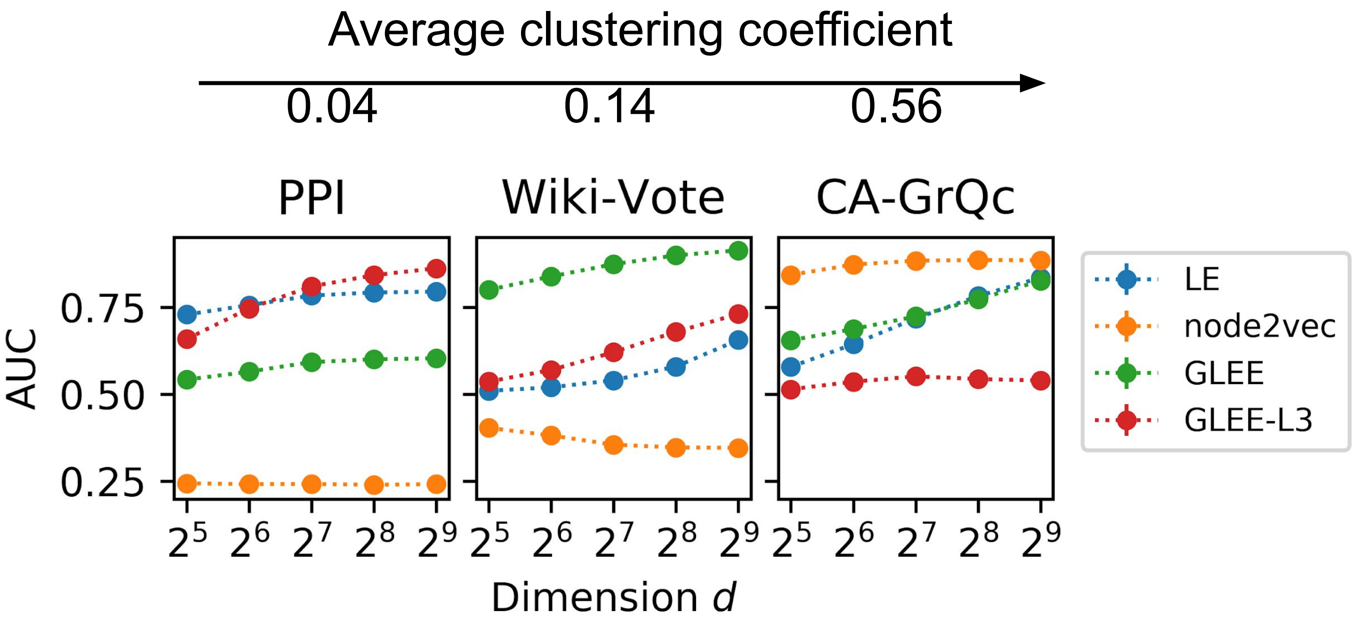


Link Prediction: 3-paths

In other networks with **low clustering** (e.g. PPI networks), a better predictor is the number of paths of length 3.

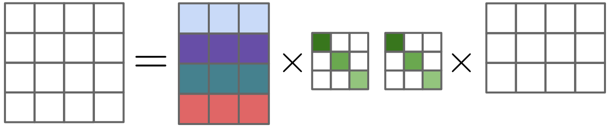


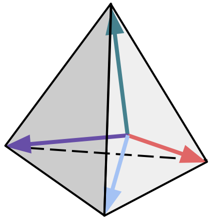
Link prediction: results

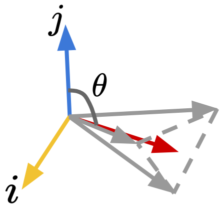


Summary

There is a bijection between undirected graphs and simplices, THEREFORE we can encode graph structure in geometric terms using **GLEE**. IN CONTRAST, other methods usually make assumptions about the structure of the graph and therefore perform well only when those assumptions hold (e.g. high clustering coefficient).



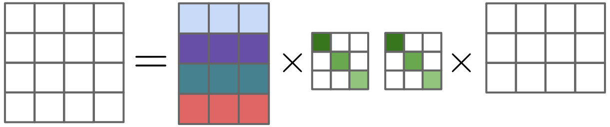


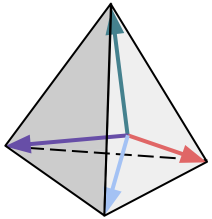


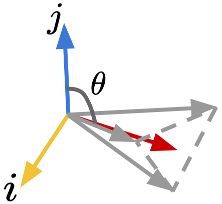
Work supported by **NSF** CNS-1314603, **NSF** IIS-1741197, Army Research Laboratory **Cooperative Agreement** W911NF-13-2-0045.

Summary: Geometry

- What else can we do with the geometry of embeddings?
- How can graphs be encoded geometrically?

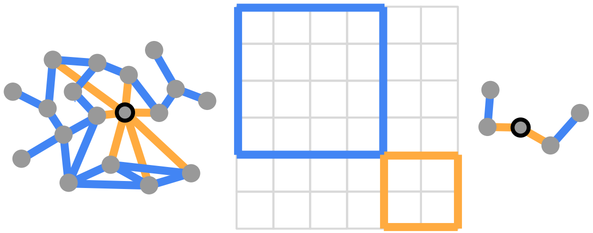






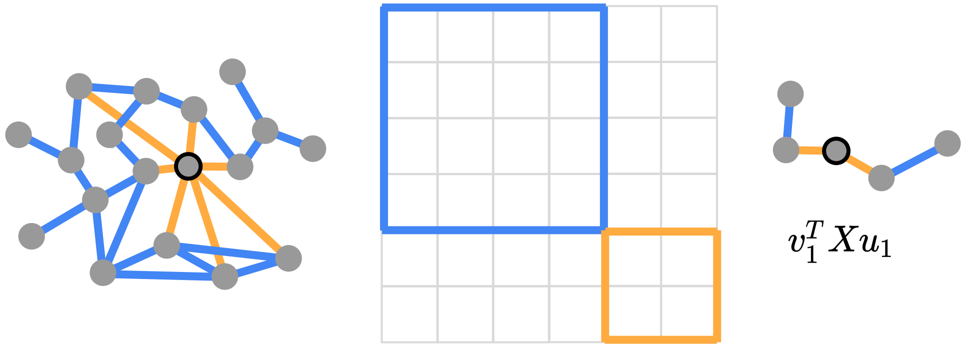
Work supported by **NSF** CNS-1314603, **NSF** IIS-1741197, Army Research Laboratory **Cooperative Agreement** W911NF-13-2-0045.

Perturbations



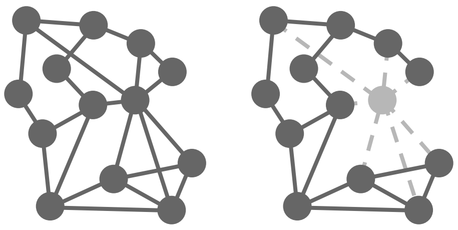
Summary

The largest eigenvalue behaves in predictable ways. THEREFORE, monitoring it should provide a good defense against adversarial attack. FOR EXAMPLE, to immunize against certain recurrent state dynamics, first remove hubs, then break up the cliques.



$v_1^T X u_1$

Perturbations



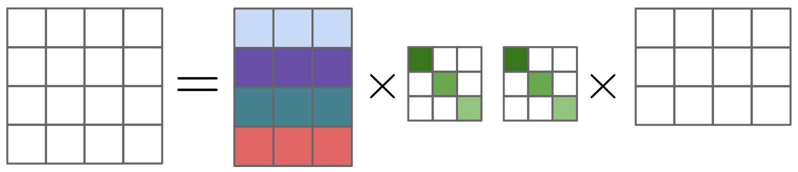
Node Immunization with Non-backtracking Eigenvalues

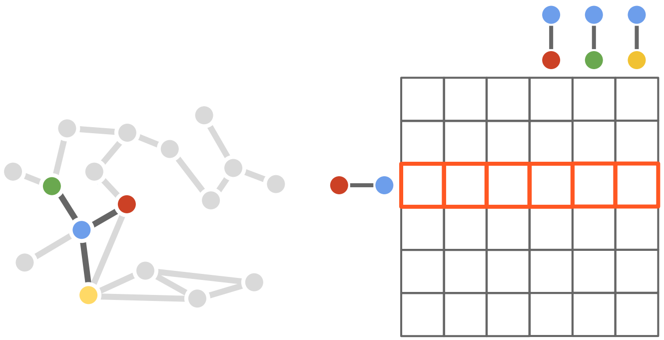
Torres, L., Chan, K.S., Tong, H., Eliassi-Rad, T. Preprint. arXiv:2002.12309 (2020).

Optimizing Graph Structure for Targeted Diffusion

Yu, S., Torres, L., Alfeld, S., Eliassi-Rad, T., and Vorobeychik, Y. Preprint. arXiv:2008.05589 (2020).

Geometry...?





Work supported by **NSF** CNS-1314603, **NSF** IIS-1741197, Army Research Laboratory **Cooperative Agreement** W911NF-13-2-0045.

Gracias!

Currently on the **job market** as a **postdoc** or **assistant professor** at the intersection of network science, computer science, and mathematics. Please get in touch!

leo@leotrs.com

www.leotrs.com

@_leotrs

/ leotrs