TDA + Machine Learning

Utilizing Topological Structures of Data for Machine Learning

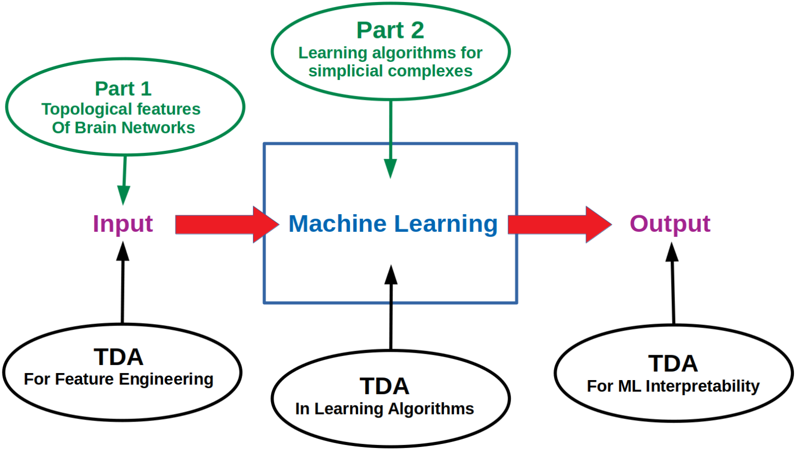
Sourabh Palande

MSU TDA Seminar

October 26, 2020

Vision

Goal: Integrating TDA into different stages of ML pipelines.



Overview of the Talk

1. TDA for Brain Networks.
   1. Statistical Inference with β_0 curves.
   2. Regression with persistence diagrams.
   3. Classification with persistence diagrams.
2. Spectral Algorithms for Simplicial Complexes.
   1. Spectral sparsification.
   2. Learning algorithms (spectral clustering, label propagation).
   3. Random walks on Simplicial Complexes.
3. Ongoing Projects.
4. Conclusion.

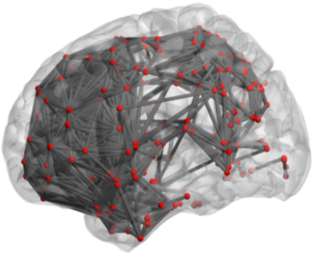
Part 1

Machine Learning with Topological Features of Brain Networks

Motivation: Each data point is a network.

Structural Brain Networks

Functional Brain Networks



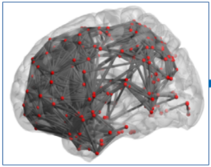
Approach: Extract topological features from brain networks and use them for machine learning.

Contributions

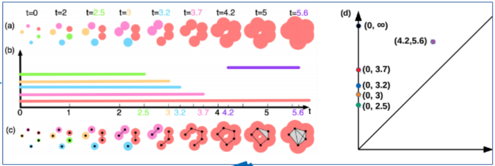
- Statistical inference for structural brain networks.
- Predicting behavioral measures with functional brain networks.
- Classifying functional brain networks.

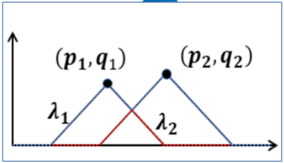
Topological Features

Brain Networks

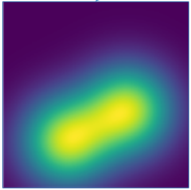


Persistent Homology





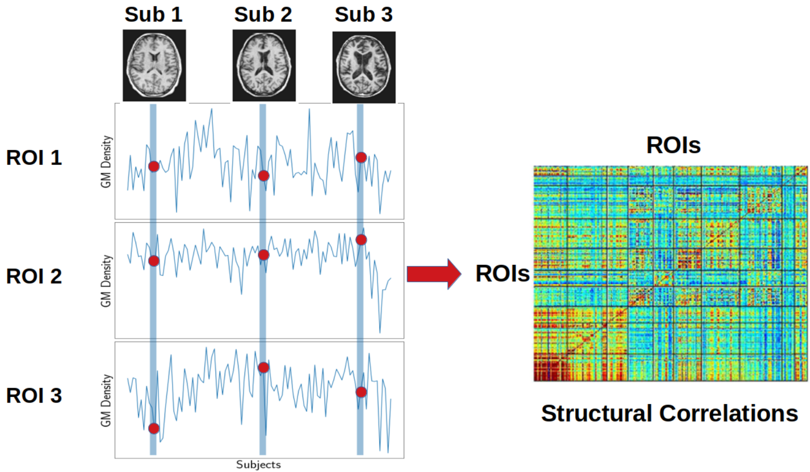
Persistence Landscapes



Persistence Images

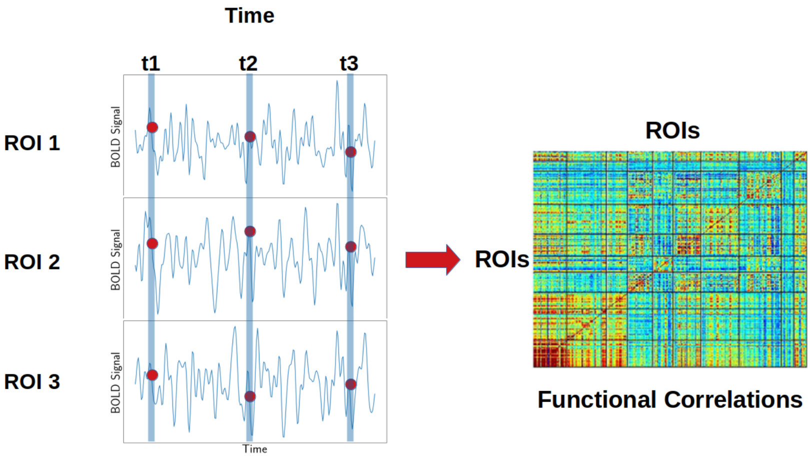
- **Kernels**
- **Projection layer for NN**

Structural Brain Networks



Encode shared structural influences across a group of subjects.

Functional Brain Networks



Encode level of synchronicity across time (for a single subject).

Graph Filtration

β_0: # Connected Components.

β_0 Curve: Changes in connectivity across a sequence of thresholds.

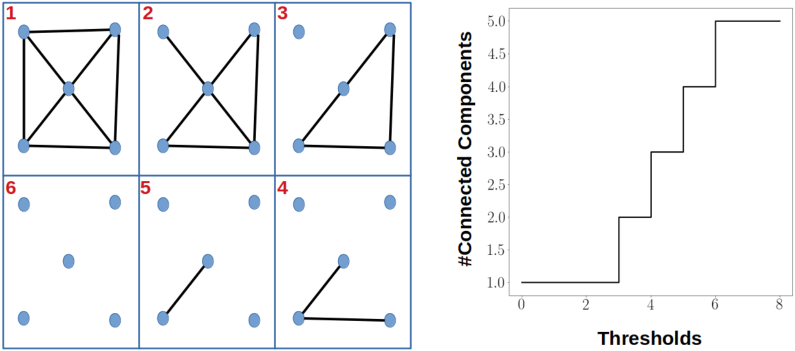


Figure: Graph filtration to compute β_0 curve.

Tracks changes in topology across multiple scales.

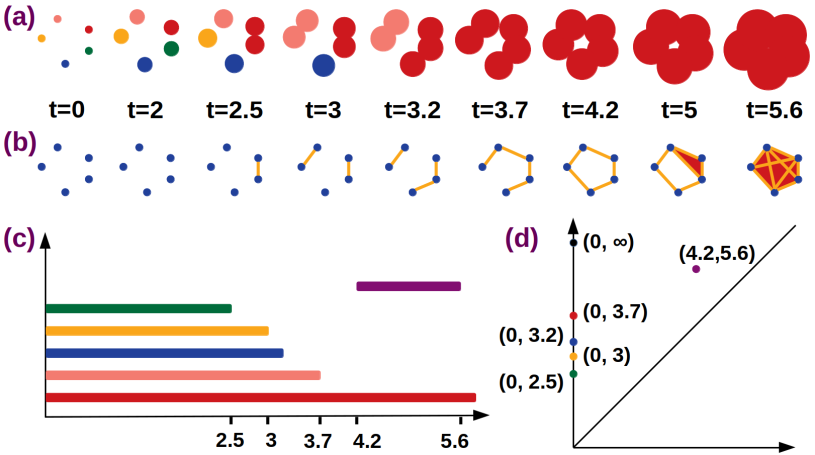
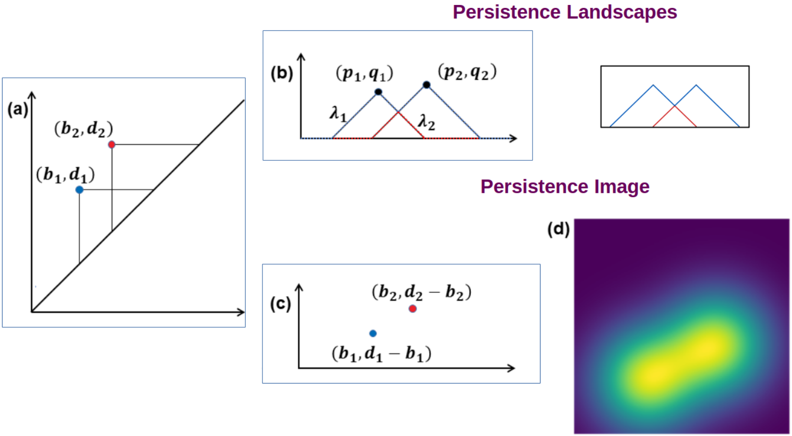


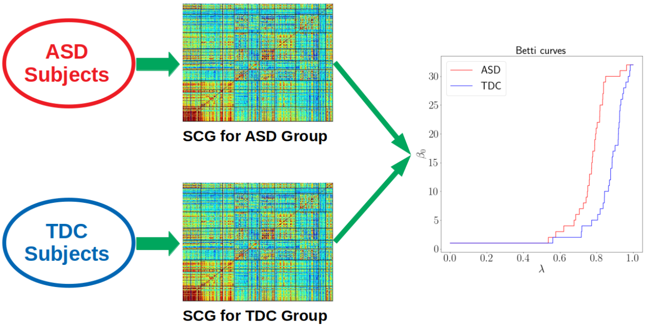
Figure: **(a, b)** Persistent homology computation, **(c)** Persistence barcode, and **(d)** Persistence diagram.

Persistent Homology: Representations

Transform persistence diagrams to vectorizable representations.



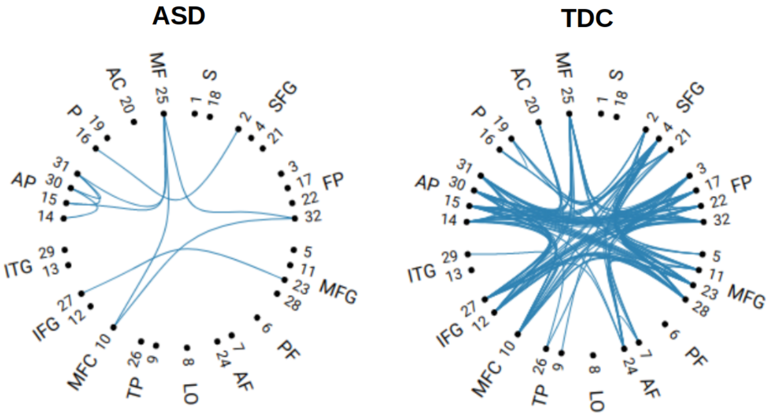
Statistical Inference with Structural Networks

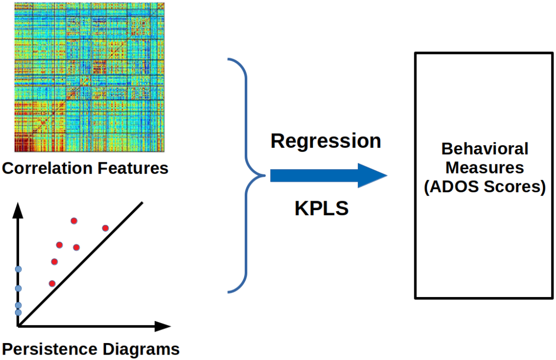


- Permutation, Bootstrap tests.
  - Test statistic: Largest gap between β_0 curves.

Main Result: Evidence of abnormalities in gray matter regions associated with Salience Network (SN). [BrainCon2019]

Difference between ASD and TDC at max-gap threshold for SN.





Method: Kernel Partial Least Squares Regression (KPLS).

Main result: The model augmenting correlations with topological features has the best predictive power and it is the only model that shows statistically significant improvement over other models. [ISBI2016]

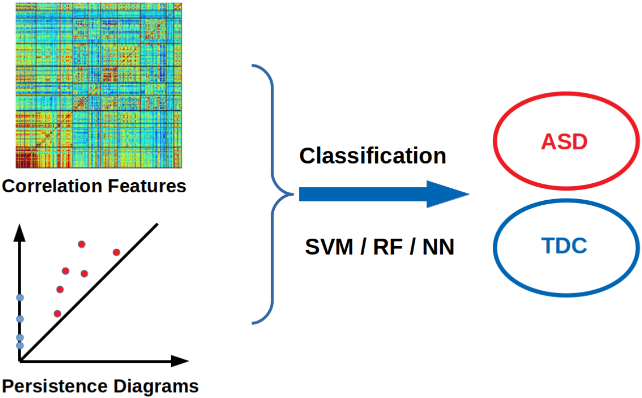
Methods

- Correlation features: Linear kernel (K^{cor}).
- Persistence Diagrams: Scale-space kernel Reininghaus, Huber, Bauer, and Kwitt 2015 (K^{TDA}).
- Combined kernels ($K^{\text{TDA+cor}}$):

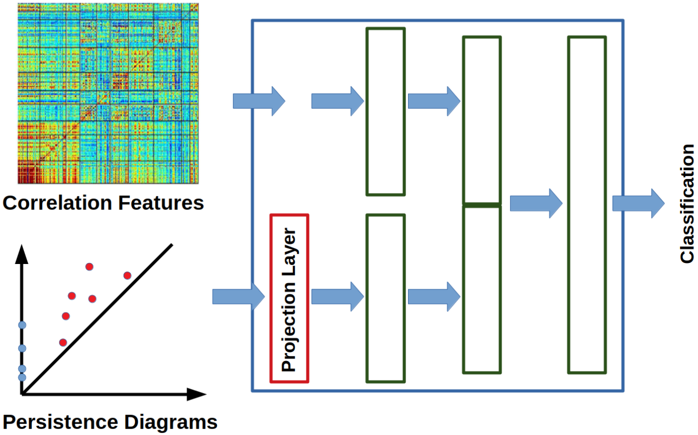
$$K^{\text{TDA+cor}} = w_0 K^{\text{TDA}_0} + w_1 K^{\text{TDA}_1} + (1 - w_0 - w_1) K^{\text{cor}}.$$

Best result with $w_0 = 0.10$, $w_1 = 0.40$.

Autism Classification with Functional Networks



Autism Classification with Functional Networks



Projection layer: Hofer et al.[1]

[1] Hofer, Kwitt, Niethammer, and Uhl 2017.

Results

Model	CC200	CC400	Model	CC200	CC400	Model	CC200	CC400
SVM_{Corr}	65.41	66.33	-	-	-	-	-	-
RF_{Corr}	64.81	63.92	-	-	-	-	-	-
$NN3_{Corr}$	68.35	63.92	-	-	-	-	-	-
$NN5_{Corr}$	68.46	65.58	-	-	-	-	-	-
$NN7_{Corr}$	67.10	62.06	-	-	-	-	-	-
SVM_{PD}	53.03	53.69	SVM_{PI}	54.54	53.76	SVM_{PL}	53.03	53.69
RF_{PD}	-	-	RF_{PI}	52.25	53.04	RF_{PL}	52.51	53.12
$NN3_{PD}$	56.06	55.90	$NN3_{PI}$	58.56	56.10	$NN3_{PL}$	55.36	54.24
$NN5_{PD}$	56.15	56.04	$NN5_{PI}$	59.09	57.39	$NN5_{PL}$	55.18	55.72
$NN7_{PD}$	55.48	54.33	$NN7_{PI}$	56.75	55.58	$NN7_{PL}$	54.85	53.67
$SVM_{PD+Corr}$	65.86	63.36	$SVM_{PI+Corr}$	64.25	62.68	$SVM_{PL+Corr}$	65.65	64.12
$NN3_{PD+Corr}$	**69.19**	**67.84**	**$NN3_{PI+Corr}$**	**67.2**	**67.02**	**$NN3_{PL+Corr}$**	**68.5**	**66.76**
$NN5_{PD+Corr}$	68.20	66.03	$NN5_{PI+Corr}$	66.87	66.23	$NN5_{PL+Corr}$	67.45	66.48
$NN7_{PD+Corr}$	64.47	61.25	$NN7_{PI+Corr}$	65.10	64.16	$NN7_{PL+Corr}$	67.02	65.26

	RF_{Corr}	SVM_{Corr}	$SVM_{PD+Corr}$	$NN3_{Corr}$
SVM_{Corr}	0.1502			
$SVM_{PD+Corr}$	0.1943	0.4213		
$NN3_{Corr}$	**0.0461**	**0.0480**	0.0631	
$NN3_{PD+Corr}$	**0.0406**	**0.0446**	**0.0414**	0.1894
	RF_{Corr}	SVM_{Corr}	$SVM_{PI+Corr}$	$NN3_{Corr}$
$SVM_{PI+Corr}$	0.1943	0.4213		
$NN3_{Corr}$	-	-	**0.0420**	
$NN3_{PI+Corr}$	**0.0493**	0.0763	0.0734	0.7432
	RF_{Corr}	SVM_{Corr}	$SVM_{PL+Corr}$	$NN3_{Corr}$
$SVM_{PL+Corr}$	0.1623	0.3513		
$NN3_{Corr}$	-	-	0.0581	
$NN3_{PL+Corr}$	**0.0467**	0.0683	0.0717	0.3524

Kernel SVM	CC-200	CC-400
K_S	53.03	53.69
K_S + Corr	65.86	63.36
K_G	52.51	53.12
K_G+ Corr	62.98	61.41
K_W	55.36	54.24
K_W + Corr	64.73	64.12
K_F	55.18	55.72
K_F + Corr	61.48	60.25

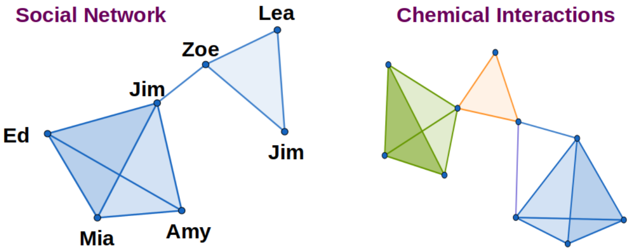
Main Results:

- Hybrid models typically outperform.
- Best accuracy: 69.19% (3-layer hybrid NN).
- Improvement is not always statistically significant.

Part 2

Spectral Algorithms for Simplicial Complexes

Motivation: Data modeled as a simplicial complex



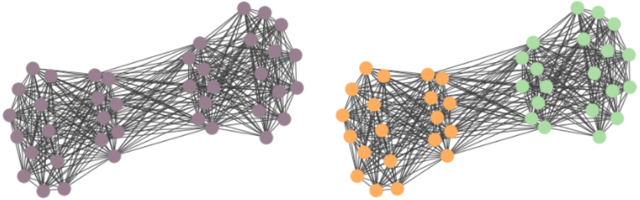
Approach: Leverage topological structures encoded by higher order interactions in machine learning algorithms.

Contributions

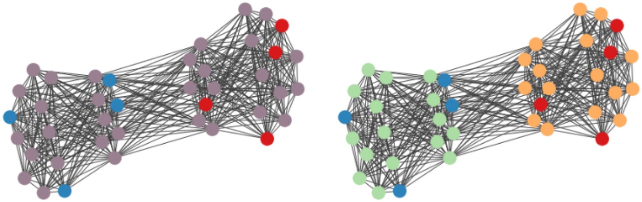
- Label propagation and spectral clustering algorithms for simplicial complexes.
- Spectral sparsification algorithm for simplicial complexes.
- Some perspectives on random walks on simplicial complexes.

Graphs

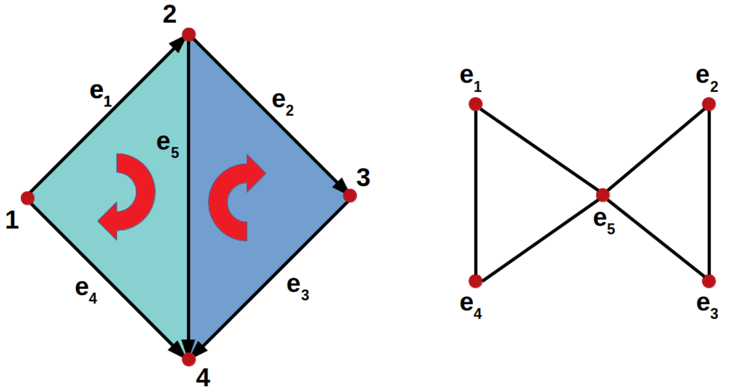
Spectral Clustering



Label Propagation

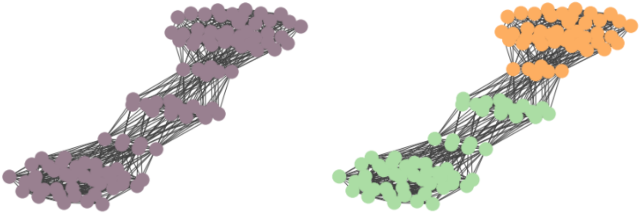


Dual graph at dimension 1.

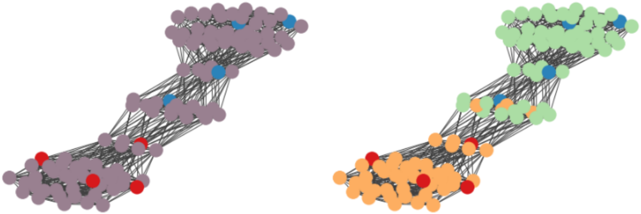


Edges in SC become vertices in dual graph.
If edges in SC share a triangle, vertices in dual graph are connected.

Spectral Clustering

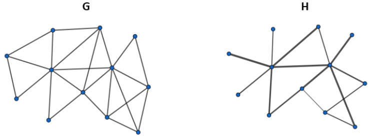


Label Propagation



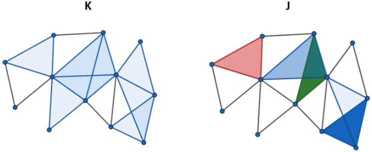
Sparsification: Preserving Spectral Properties

Graphs:



$$(1-\epsilon)L_G \preceq \; L_H \preceq (1+\epsilon)L_G.$$

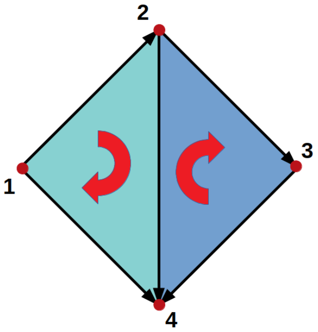
Simplicial Complexes:



$$(1-\epsilon)\mathcal{L}_K \preceq \; \mathcal{L}_J \preceq (1+\epsilon)\mathcal{L}_K.$$

$$(1-\epsilon)x^T\mathcal{L}_K x \leq x^T\mathcal{L}_J x \leq (1+\epsilon)x^T\mathcal{L}_K x.$$

- *Oriented* simplicial complex K: Every simplex is oriented.
- n_k: Number of k-simplices.
- S_k: set of all k-simplices in K. w_f: weight of simplex $f \in K$.
- W_k: diagonal matrix, $W_k(f,f) = w_f$ where $f \in S_k$.

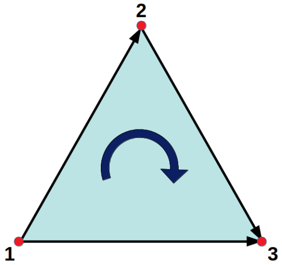


$V = \{1,2,3,4\}$, $E = \{12,23,34,14,24\}$, $T = \{124,234\}$.

Simplicial Complex: Incidence, up-Adjacency

D_k: Incidence matrix ($\mathbb{R}^{n_{k+1}} \times \mathbb{R}^{n_k}$).

$$D_k(i,j) = \begin{cases} 0 & \text{if } \sigma_j^k \text{ is not on the boundary of } \sigma_i^{k+1}, \\ 1 & \text{if orientation of } \sigma_j^k \text{ agrees with the} \\ & \text{orientation induced by } \sigma_i^{k+1}, \\ -1 & \text{if orientation of } \sigma_j^k \text{ does not agree with the} \\ & \text{orientation induced by } \sigma_i^{k+1}. \end{cases}$$



A_k^{up}: *up*-Adjacency matrix ($\mathbb{R}^{n_k} \times \mathbb{R}^{n_k}$).

$$A_k^{\text{up}}(i,j) = \begin{cases} -w_f & \sigma_i \text{ and } \sigma_j \text{ are both faces of the same } (k+1)\text{-simplex } f \in S_{k+1}(K) \\ & \text{and both agree or disagree with the orientation of } f, \\ w_f & \sigma_i \text{ and } \sigma_j \text{ are both faces of the same } (k+1)\text{-simplex } f \in S_{k+1}(K) \\ & \text{and either } \sigma_i \text{ or } \sigma_j \text{ (but not both) agree with the orientation of } f, \\ 0 & \text{if } \sigma_i \text{ and } \sigma_j \text{ are not faces of the same } (k+1)\text{-simplex } f \\ & \text{for any } f \in S_{k+1}(K)\ . \end{cases}$$

Simplicial Complex: up-Laplacian

- $\deg(\sigma_i)$: Degree of a k-simplex σ_i:

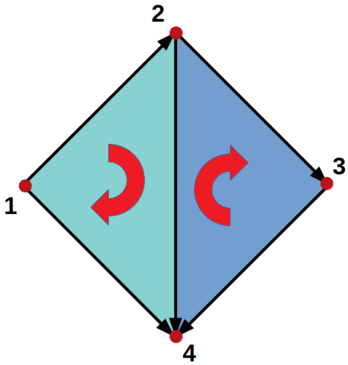
$$\deg(\sigma_i) = \sum_{\substack{f \in S_{k+1} \\ \sigma_i \subset f}} w_f .$$

- Δ_k: Degree matrix $(\mathbb{R}^{n_k} \times \mathbb{R}^{n_k})$, $\Delta_k(i,i) = \deg(\sigma_i)$.
- The k-dimensional up-Laplacian is:

$$\mathcal{L}_k^{\mathrm{up}} = W_k^{-1} \left(D_k^T W_{k+1} D_k \right),$$

$$= W_k^{-1} \left(\Delta_k - A_k^{\mathrm{up}} \right).$$

Running Example



$$D_1 = \begin{array}{c} \\ (124) \\ (234) \end{array} \begin{array}{c} \begin{array}{ccccc} (12) & (23) & (34) & (14) & (24) \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & -1 & 1 \\ 0 & 1 & 1 & 0 & -1 \end{pmatrix} \end{array}$$

$$A_1^{\mathrm{up}} = \begin{array}{c|ccccc} & (12) & (23) & (34) & (14) & (24) \\ \hline (12) & 0 & 0 & 0 & 1 & -1 \\ (23) & 0 & 0 & -1 & 0 & 1 \\ (34) & 0 & -1 & 0 & 0 & 1 \\ (14) & 1 & 0 & 0 & 0 & 1 \\ (24) & -1 & 1 & 1 & 1 & 0 \end{array}$$

$$\Delta_1 = \begin{array}{c|ccccc} & (12) & (23) & (34) & (14) & (24) \\ \hline (12) & 1 & & & & \\ (23) & & 1 & & & \\ (34) & & & 1 & & \\ (14) & & & & 1 & \\ (24) & & & & & 2 \end{array}$$

$$\mathcal{L}_1^{\mathrm{up}} = \begin{array}{c|ccccc} & (12) & (23) & (34) & (14) & (24) \\ \hline (12) & 1 & 0 & 0 & -1 & 1 \\ (23) & 0 & 1 & 1 & 0 & -1 \\ (34) & 0 & 1 & 1 & 0 & -1 \\ (14) & -1 & 0 & 0 & 1 & -1 \\ (24) & 1 & -1 & -1 & -1 & 2 \end{array}$$

- $(\mathcal{L}^{\mathrm{up}}_{k-1})^+$: Moore-Penrose pseudoinverse.

$$
\begin{aligned}
R_k &= D_{k-1}(\mathcal{L}^{\mathrm{up}}_{k-1})^+ D^T_{k-1} \\
&= D_{k-1}\left(W^{-1}_{k-1} D^T_{k-1} W_k D_{k-1}\right)^+ D^T_{k-1}.
\end{aligned}
$$

- **Generalized effective resistance** of k-simplex $f = R_k(f, f)$.
- In case of graphs: Effective resistance is the voltage drop between vertices for a unit current.

Algorithm 1: $J = \text{Sparsify}(K, k, q)$

Data: A weighted, oriented simplicial complex K, dimension k (where $1 \leq k \leq \dim K$), and an integer q.

Result: A weighted, oriented simplicial complex J which is sparsified at dimension k, with equivalent $(k-1)$-skeleton to K and $\dim J = k$.

$J := K^{(k-1)}$

Sample q k-dimensional simplices independently with replacement according to the probability

$$p_f = \frac{w(f)R_k(f,f)}{\sum_f w(f)R_k(f,f)},$$

and add sampled simplices to J with weight $w(f)/qp_f$. If a simplex is chosen more than once, the weights are summed.

Algorithm 2: y = Cluster(G, d)[a]

Data: $G(V, E, W)$: A weighted, undirected graph with $|V| = n$, d: the number of clusters

Result: y: A vector of cluster assignments $l \in \{1, 2, \ldots, d\}$ for the vertices of G.

Construct matrix A, $A(i, j)$: weight of edge $e_{i,j}$.

Compute diagonal matrix D, $D(j, j) = \sum_i A(i, j)$.

Compute $\mathrm{M} = \mathrm{D}^{-1/2}\mathrm{A}\mathrm{D}^{-1/2}$.

Construct matrix $X = [u_1 u_2 \cdots u_d] \in \mathbb{R}^{n \times d}$ where u_i's are the eigenvectors corresponding to the d largest eigenvalues of M (chosen to be orthogonal to each other in the case of repeated eigenvalues).

$Y_{ij} = X_{ij} / \left(\sum_j X_{ij}^2\right)^{1/2}$ (normalize rows of X to unit length).

y = kMeans(Y, d).

Return y as cluster assignments for vertices of G.

[a] Ng, Jordan, and Weiss 2001.

Graph: Label Propagation (ZG)

Algorithm 3: $\mathrm{y} = \mathrm{PropagateLabels}(G, \mathrm{y_l})$

[a] **Data:** $G(V, E, W)$: A weighted, undirected graph with $|V| = n$, $\mathrm{y_l}$: A vector containing labels $\in \{+1, -1\}$ of first l vertices.

Result: y: A vector of label assignments $l \in \{+1, -1\}$ for all the vertices of G.

Construct matrix A, $A(i, j)$: weight of edge $e_{i,j}$.

Compute diagonal matrix D, $D(j, j) = \sum_i A(i, j)$.

$\mathrm{P} = \mathrm{AD}^{-1}$.

Initialize $\mathrm{y}^{(0)} = (\mathrm{y}_l, 0)$, $t = 0$.

Repeat until convergence:

$$\mathrm{y}^{(t+1)} = P\mathrm{y}^{(t)},$$

$$\mathrm{y}_l^{(t+1)} = \mathrm{y}_l^{(t)}.$$

Return $\mathrm{sgn}(\mathrm{y}^{(\mathrm{t})})$ as label assignments for vertices of G.

[a] Zhu and Ghahramani 2002.

Learning Algorithms for Simplicial Complex

We perform learning on the dual graph at dimension k.

$$A_k^{\text{dual}}(i,j) = \begin{cases} w_f & \sigma_i \text{ and } \sigma_j \text{ are both faces} \\ & \text{of the same } (k+1)\text{-simplex } f \in S_{k+1}(K), \\ 0 & \text{otherwise.} \end{cases} .$$

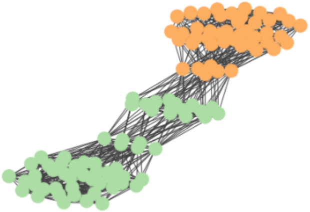
Spectral Clustering:

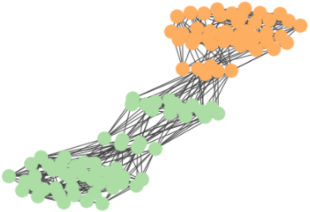
$$M = \frac{1}{k+1} \Delta_k^{-1/2} A_k^{\text{dual}} \Delta_k^{-1/2}.$$

Label Propagation:

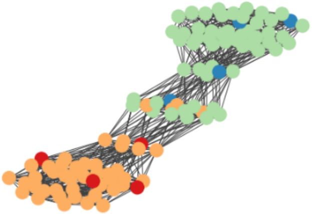
$$P = \frac{1}{k+1} A_k^{\text{dual}} \Delta_k^{-1}.$$

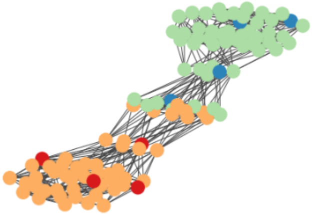
Spectral Clustering:



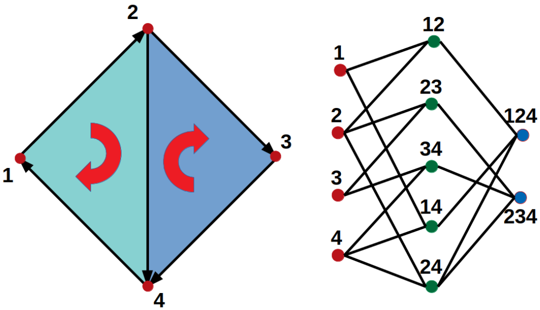


Label Propagation:





Discussion: Multidimensional Random Walks



- Random walk on vertices:
  - Usually defined through edges.
  - Can also be defined through triangles.
- Random walk on k-simplices:
  - through $(k + j)$-simplices.
  - through $(k - j)$-simplices.
- Random walk spanning multiple dimensions.

Discussion: Multilevel Sparsification

- Sparsify at dimension k:
  - Copy $(k-1)$-simplices.
  - Reduce number of k-simplices.
- Theorem guarantees:

$$(1-\epsilon)\mathcal{L}_K \preceq \mathcal{L}_J \preceq (1+\epsilon)\mathcal{L}_K$$

- **Problem:** Higher order simplices are broken.
- How are up-Laplacians for dimension $\geq k+1$ affected?
- Can we sparsify several dimensions, preserving spectra of up-Laplacians at every dimension?

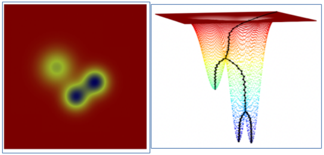
Part 3

Ongoing Projects

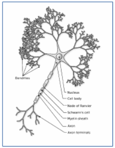
Sketching an Ensemble of Merge Trees

Motivation: Dataset is a large collection of trees.

Scalar Fields



Neurons



Trees

Goals:

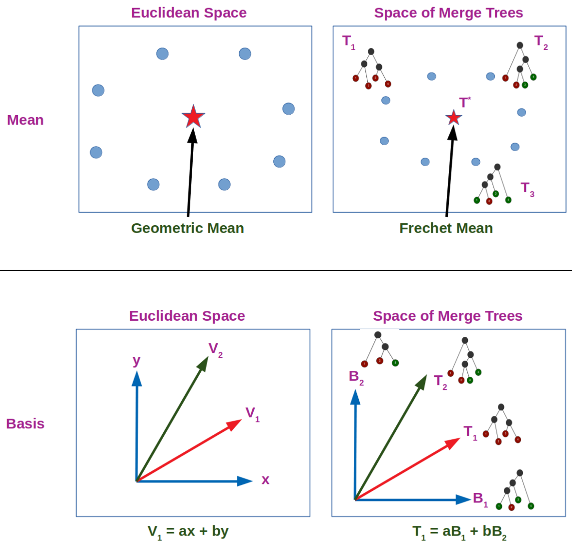
- Compute a structural average of trees.
- Compute a basis set of trees.

Approach: Adapt the Gromov-Wasserstein framework[2].

Scalar field image taken from [YanWangMunch2020].

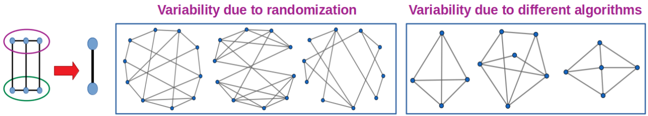
[2] Chowdhury and Needham 2019.

Fréchet Mean and Basis



Structural Variability in Graph Reduction

Motivation: Dataset is a collection of graphs output by different graph reduction algorithms or by same randomized algorithm over multiple runs.

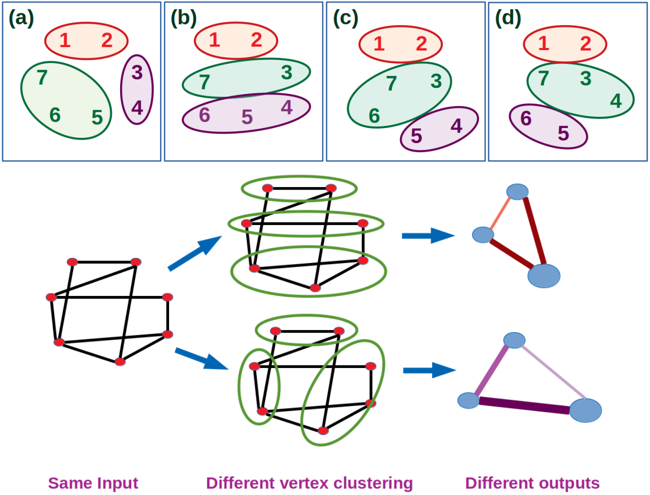


Goal: Measure the structural variability in supervertices of the reduced graphs for uncertainty quantification and visualization.

Approach:

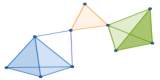
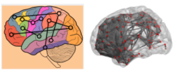
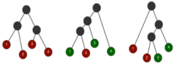
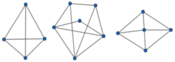
- Local similarity scores for vertices based on clustering comparisons.
- An alternative approach based on co-clustering probabilities.

Graph Reduction

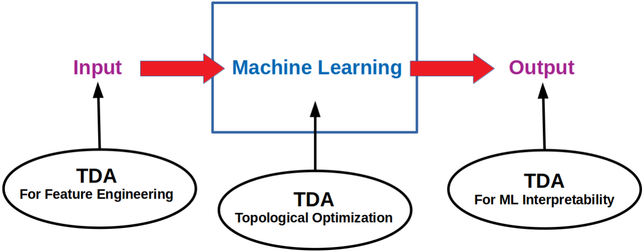


Conclusion

Summary

	Data	Utilization	Machine Learning (ML)
	Simplicial Complexes	Sparsification →	Label Propagation Spectral Clustering [JoCG'20]
	Networks	Topological Features →	Statistical Inference [BrainCon'19] Regression [ISBI'16] Classification [MICCAI'19]
	Ensembles of Merge Trees	Sketching →	Data Reduction [PalandeBhaskaraWang'20]
	Ensembles of Graphs from Graph Reduction	Structural Variability →	Uncertainty Quantification [LanPalandeYoung'20]

Vision



- Inputs: TDA for feature engineering
  - Topological summaries other than persistent homology
- Learning: Algorithms that leverage topological structure
  - Topological optimization: Priors and constraints.
- Output: Understanding / interpreting learned models
  - Understanding model behavior
  - Mapper to visualize parameter and activation spaces.

Related Publications I

[JoCG2020]: Braxton Osting, Sourabh Palande, and Bei Wang. "Spectral sparsification of simplicial complexes for clustering and label propagation.". Journal of Computational Geometry (JoCG), to appear. 2020

[BrainCon2019]: Sourabh Palande, Vipin Jose, Brandon Zielinski, Jeffrey Anderson, P. Thomas Fletcher, and Bei Wang. "Revisiting abnormalities in brain network architecture underlying autism using topology-inspired statistical inference.". In: *Brain Connectivity* 9.1 (2019), pp. 13–21

[MICCAI2019]: Archit Rathore, Sourabh Palande, Jeffrey Anderson, Brandon Zielinski, P. Thomas Fletcher, and Bei Wang. "Autism classification using topological features and deep learning: a cautionary tale.". In: *Medical Image Computing and Computer Assisted Intervention (MICCAI)*. Springer International Publishing, 2019, pp. 736–744

[ISBI2016]: Eleanor Wong, Sourabh Palande, Bei Wang, Brandon Zielinski, Jeffrey Anderson, and P. Thomas Fletcher. "Kernel partial least squares regression for relating functional brain network topology to clinical measures of behavior". In: *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2016, pp. 1303–1306. DOI: 10.1109/isbi.2016.7493506

References I

Samir Chowdhury and Tom Needham. "Gromov-wasserstein averaging in a riemannian framework". preprint, arXiv:1910.04308. 2019.

Christoph Hofer, Roland Kwitt, Marc Niethammer, and Andreas Uhl. "Deep learning with topological signatures". In: *Advances in Neural Information Processing Systems* (2017), pp. 1634–1644.

Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. "On spectral clustering: Analysis and an algorithm". In: *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*. NIPS'01. Vancouver, British Columbia, Canada: MIT Press, 2001, pp. 849–856.

References II

Braxton Osting, Sourabh Palande, and Bei Wang. "Spectral sparsification of simplicial complexes for clustering and label propagation.". Journal of Computational Geometry (JoCG), to appear. 2020.

Sourabh Palande, Vipin Jose, Brandon Zielinski, Jeffrey Anderson, P. Thomas Fletcher, and Bei Wang. "Revisiting abnormalities in brain network architecture underlying autism using topology-inspired statistical inference.". In: *Brain Connectivity* 9.1 (2019), pp. 13–21.

Archit Rathore, Sourabh Palande, Jeffrey Anderson, Brandon Zielinski, P. Thomas Fletcher, and Bei Wang. "Autism classification using topological features and deep learning: a cautionary tale.". In: *Medical Image Computing and Computer Assisted Intervention (MICCAI)*. Springer International Publishing, 2019, pp. 736–744.

References III

Jan Reininghaus, Stefan Huber, Ulrich Bauer, and Roland Kwitt. “A stable multi-scale kernel for topological machine learning”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 4741–4748.

Eleanor Wong, Sourabh Palande, Bei Wang, Brandon Zielinski, Jeffrey Anderson, and P. Thomas Fletcher. “Kernel partial least squares regression for relating functional brain network topology to clinical measures of behavior”. In: *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2016, pp. 1303–1306. DOI: 10.1109/isbi.2016.7493506.

Xiaojin Zhu and Zoubin Ghahramani. *Learning from labeled and unlabeled data with label propagation*. Tech. rep. CMU-CALD-02-107. Carnegie Mellon University, 2002.