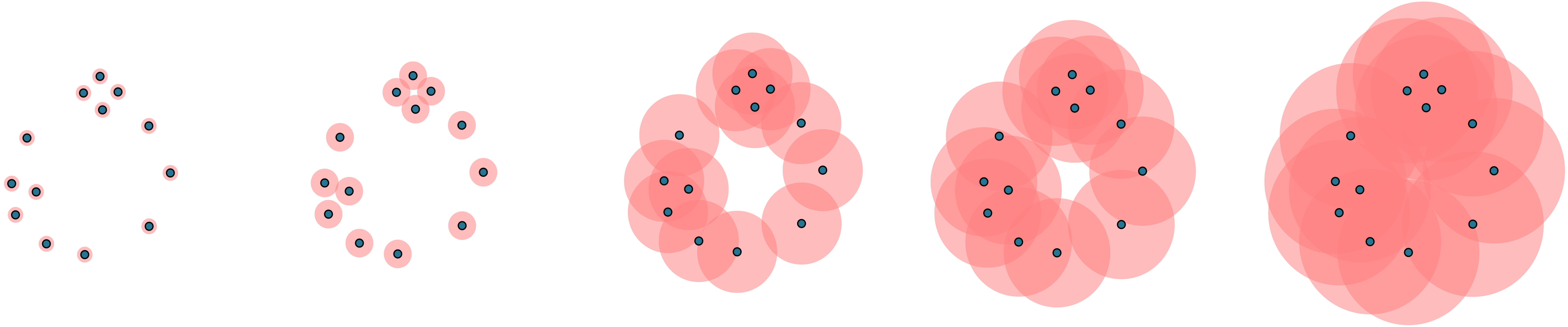
Persistent homology, hypergraphs and geometric cycle matching

Agnese Barbensi

Online Machine Learning Seminar

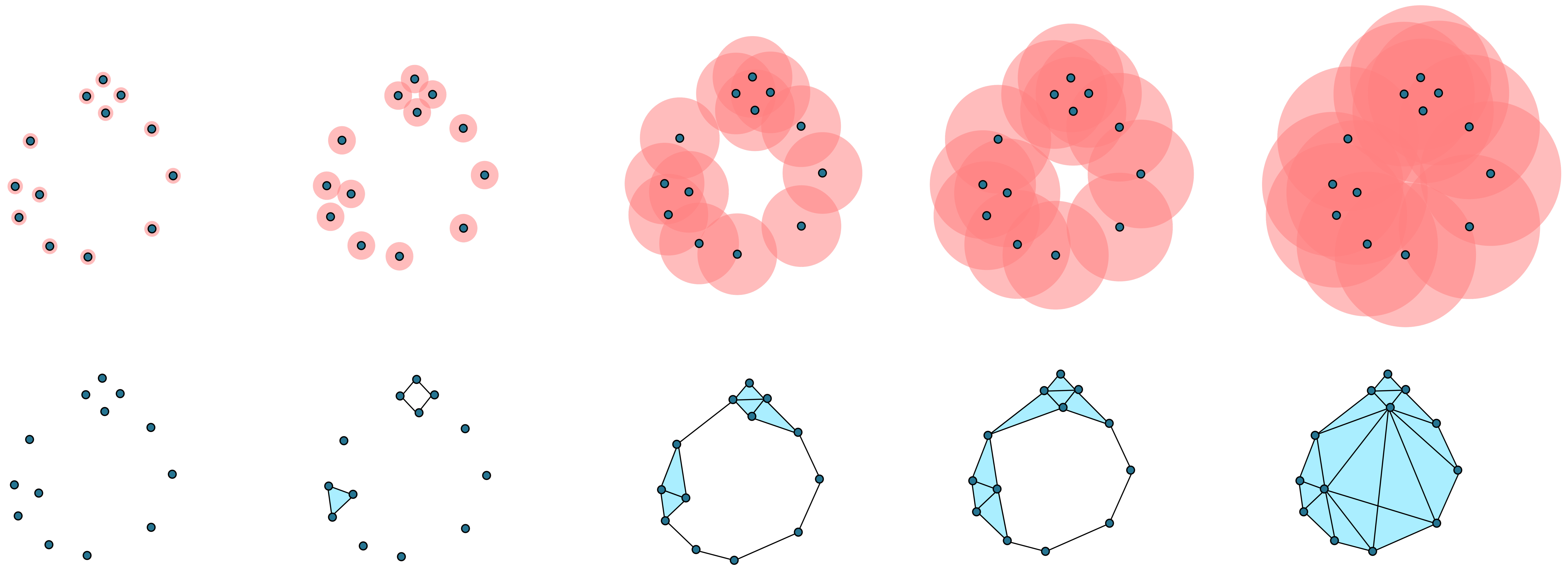
Nov 2023

Topological Data Analysis (TDA)



Builds revealing **shapes** from **data** to find features persisting across **multiple scales**

Topological Data Analysis (TDA)

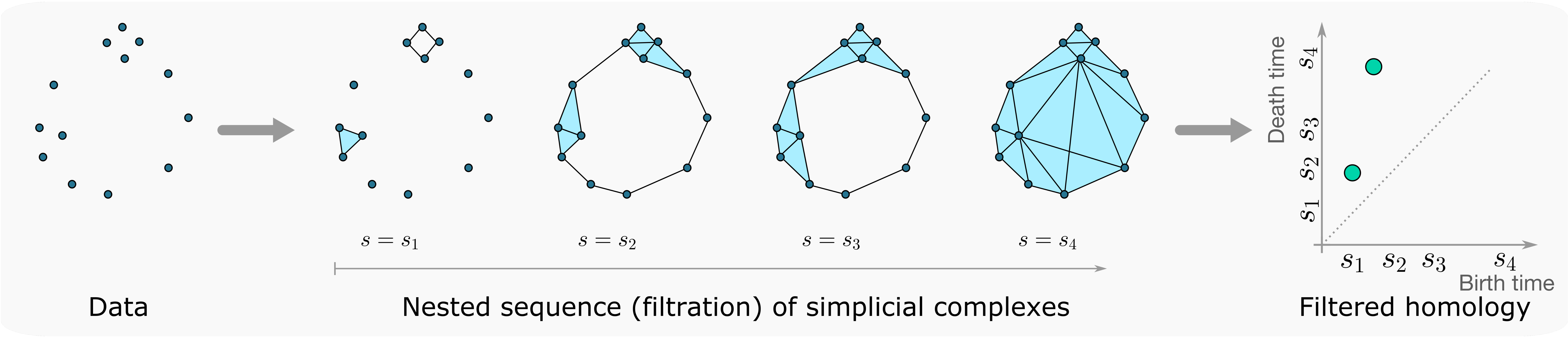


Builds revealing **shapes** from **data** to find features persisting across **multiple scales**

Simplicial complexes: combinatorial approximations of data at different scales

Topological Data Analysis (TDA)

Persistent homology (PH): algebraically describes the structure of data based on *topological* features *persisting* across different scales.

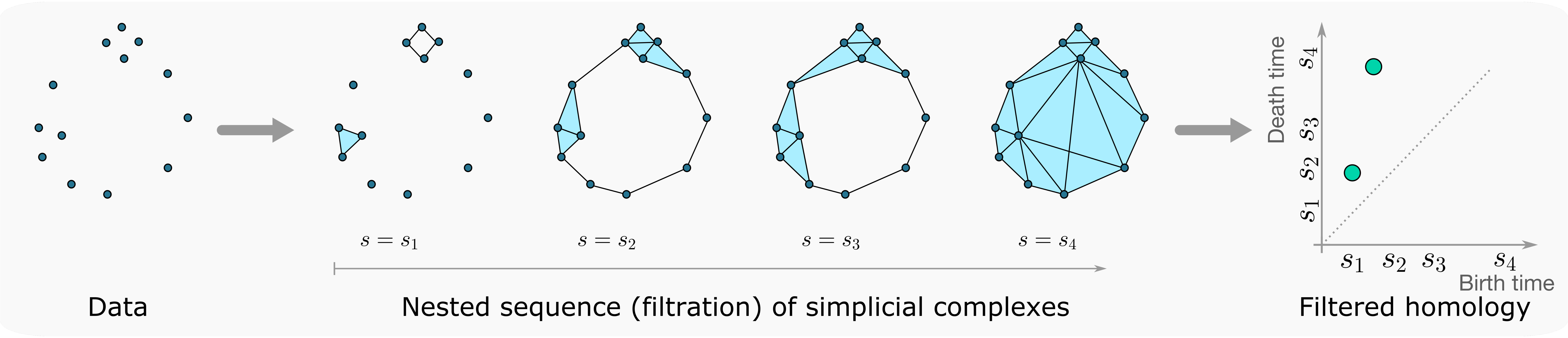


From a **point cloud,** to (filtered) **simplicial complexes,** to **homology**

Features are encoded in a **persistent diagram** (multi-set of topological features)

Topological Data Analysis (TDA)

Persistent homology (PH): algebraically describes the structure of data based on *topological* features *persisting* across different scales.

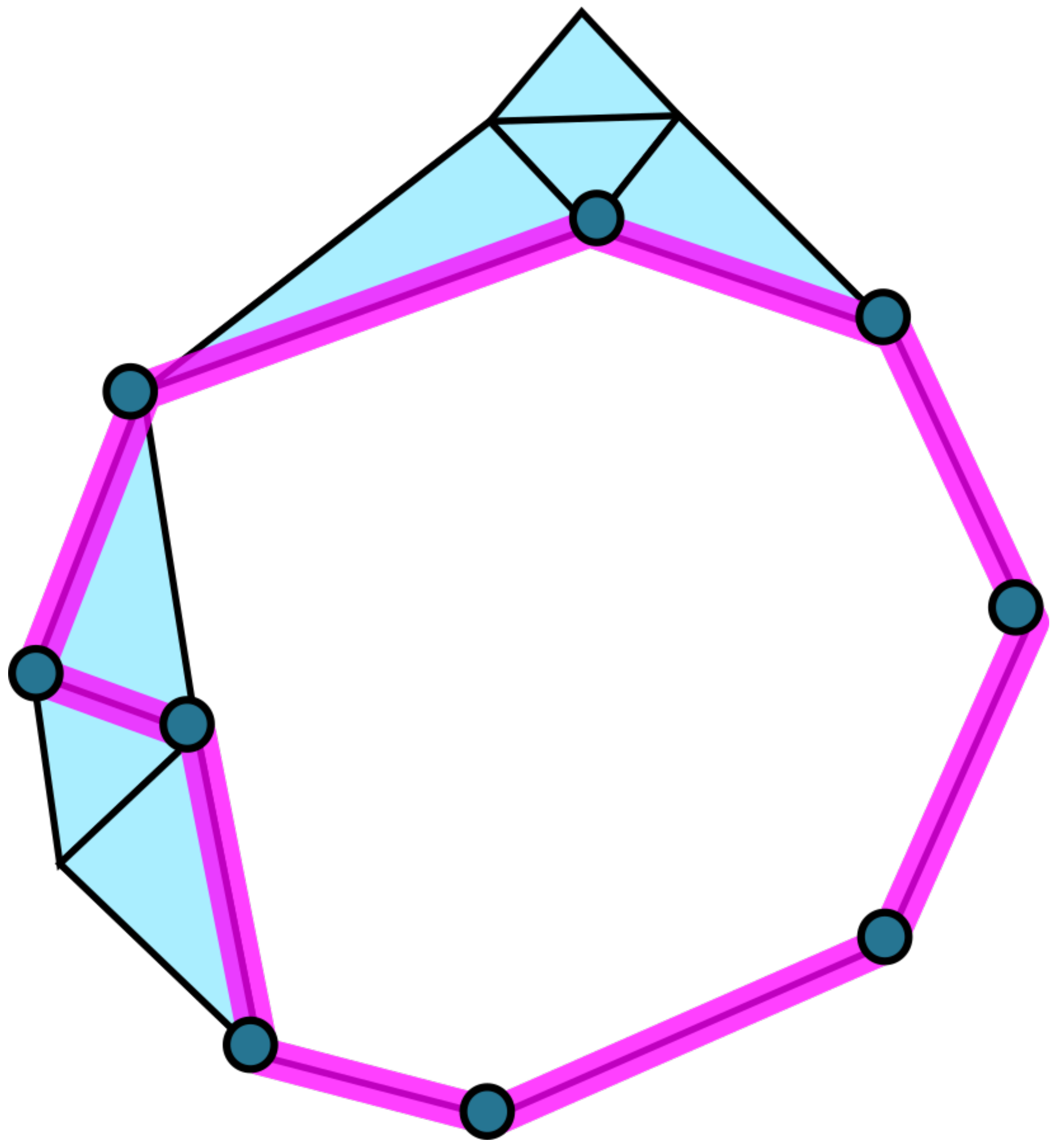


Applications: oncological studies **(1-2)** pathology **(3)** brain **(4-5)** ecology **(6)** materials **(7)**…

1) Bukkuri, et al "Applications of topological data analysis in oncology." Frontiers in artificial intelligence 2021
2) Rabadán, Raúl, et al. "Identification of relevant genetic alterations in cancer using topological data analysis." *Nature communications* 2020
3) Vipond, et al. "Multiparameter persistent homology landscapes identify immune cell spatial patterns in tumors." PNAS 2021
4) Saggar, Manish, et al. "Towards a new approach to reveal dynamical organization of the brain using topological data analysis." *Nature communications* 2018
5) Kanari, et al. "A topological representation of branching neuronal morphologies." Neuroinformatics 2018
6) McGuirl, et al. ”Topological data analysis of zebrafish patterns.” PNAS 2020
7) Sørensen, Søren S., et al. "Revealing hidden medium-range order in amorphous materials using topological data analysis." *Science Advances* 2020

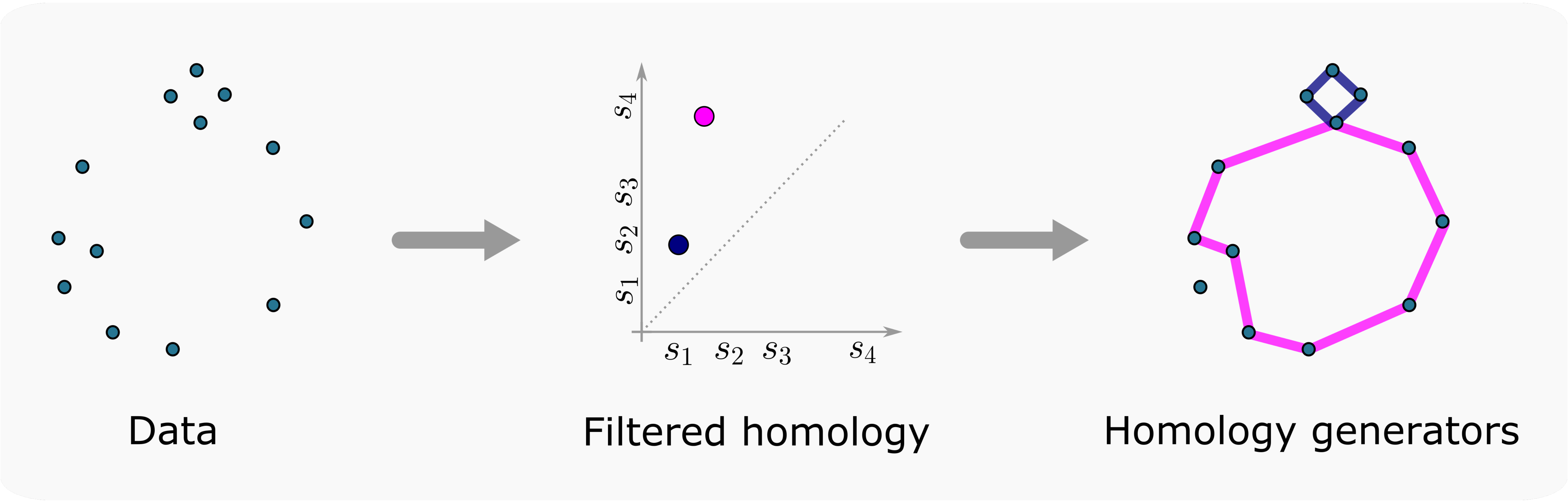
Interpretability: from *algebraic* summary to *local* structure

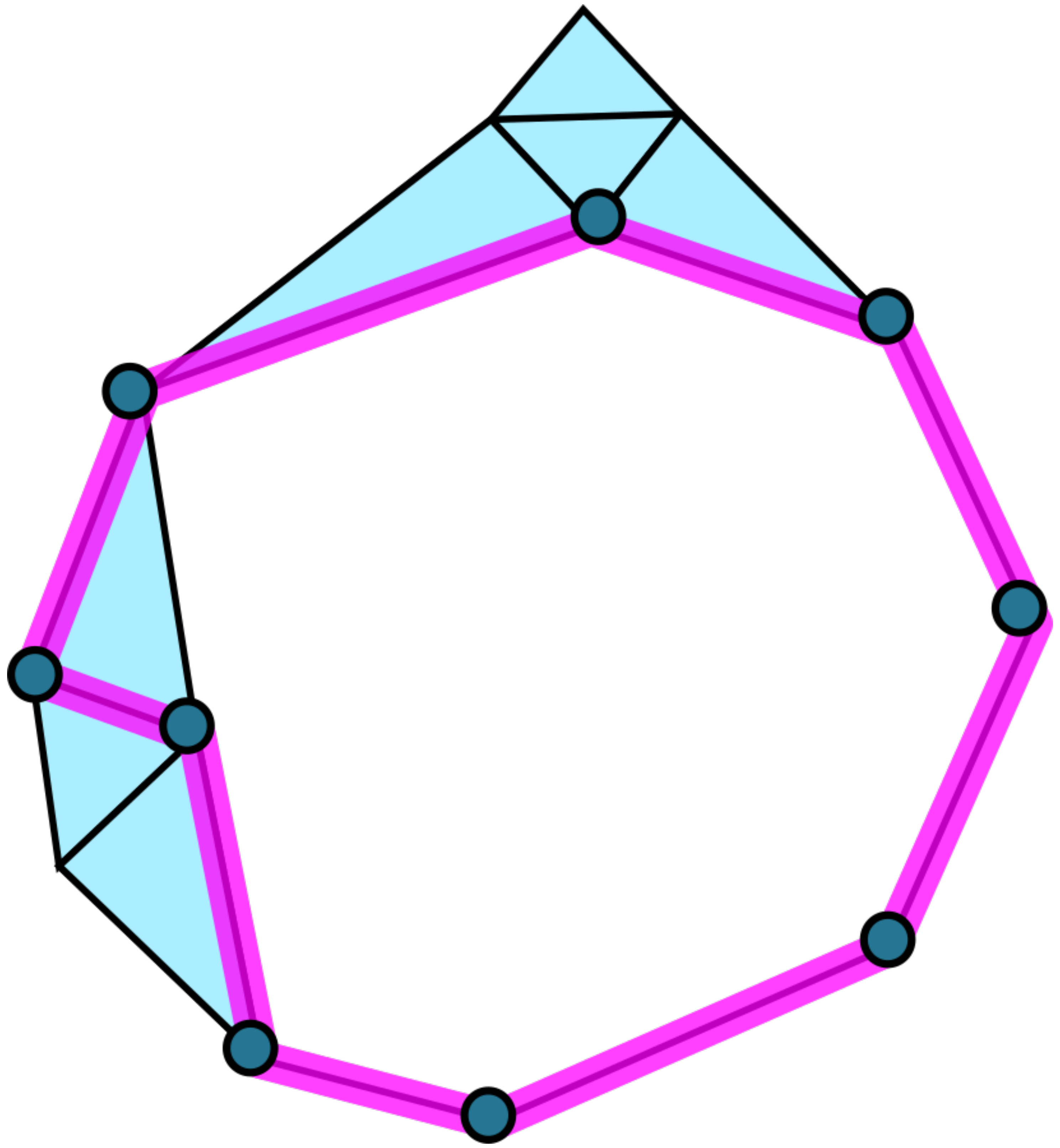
Homology generators: subsets of data giving rise to topological classes, *i.e.* those points forming **cycles** representing **homology classes**



Interpretability: from *algebraic* summary to *local* structure

Homology generators: subsets of data giving rise to <u>topological</u> classes, *i.e.* those points forming **cycles** representing **homology classes**

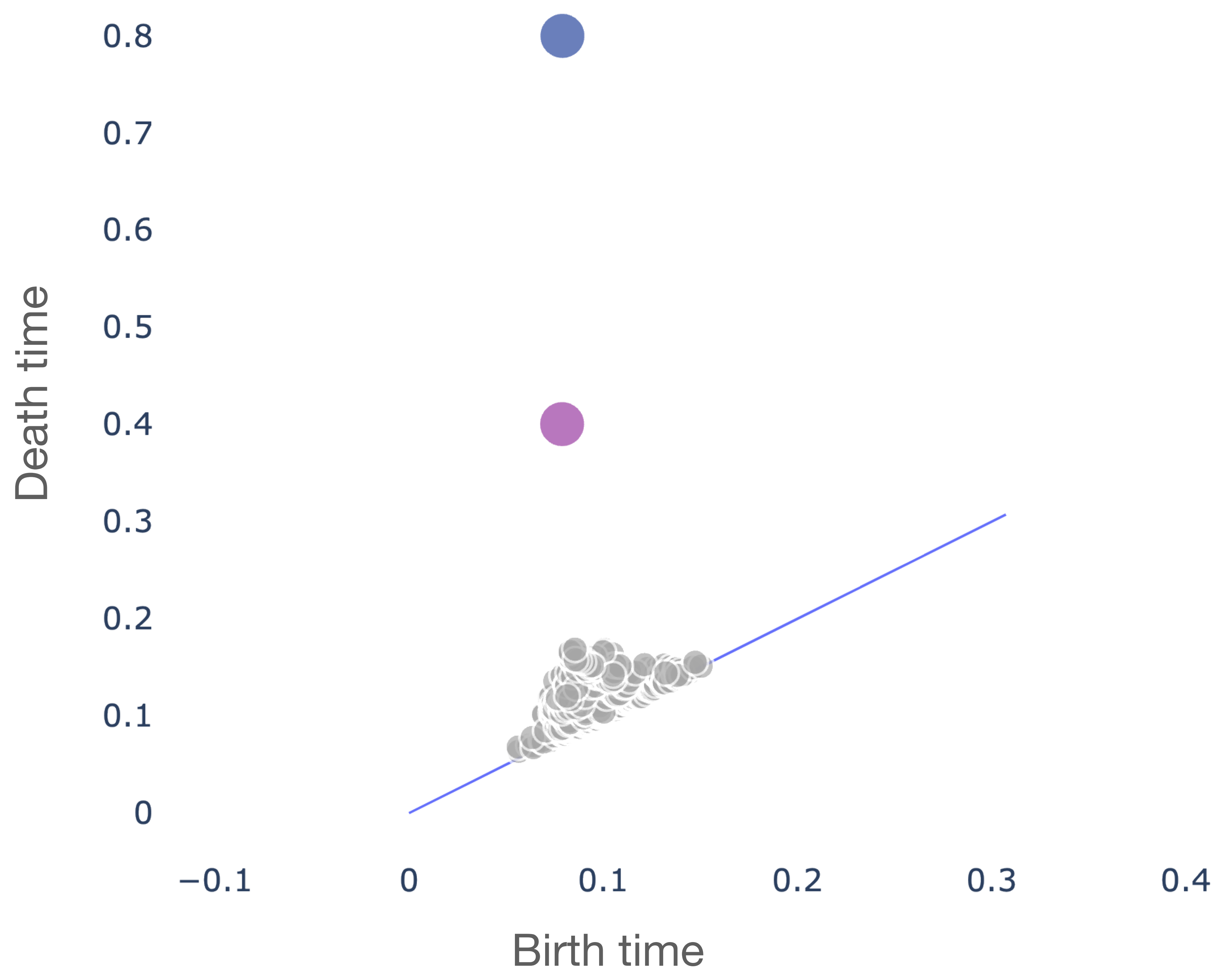


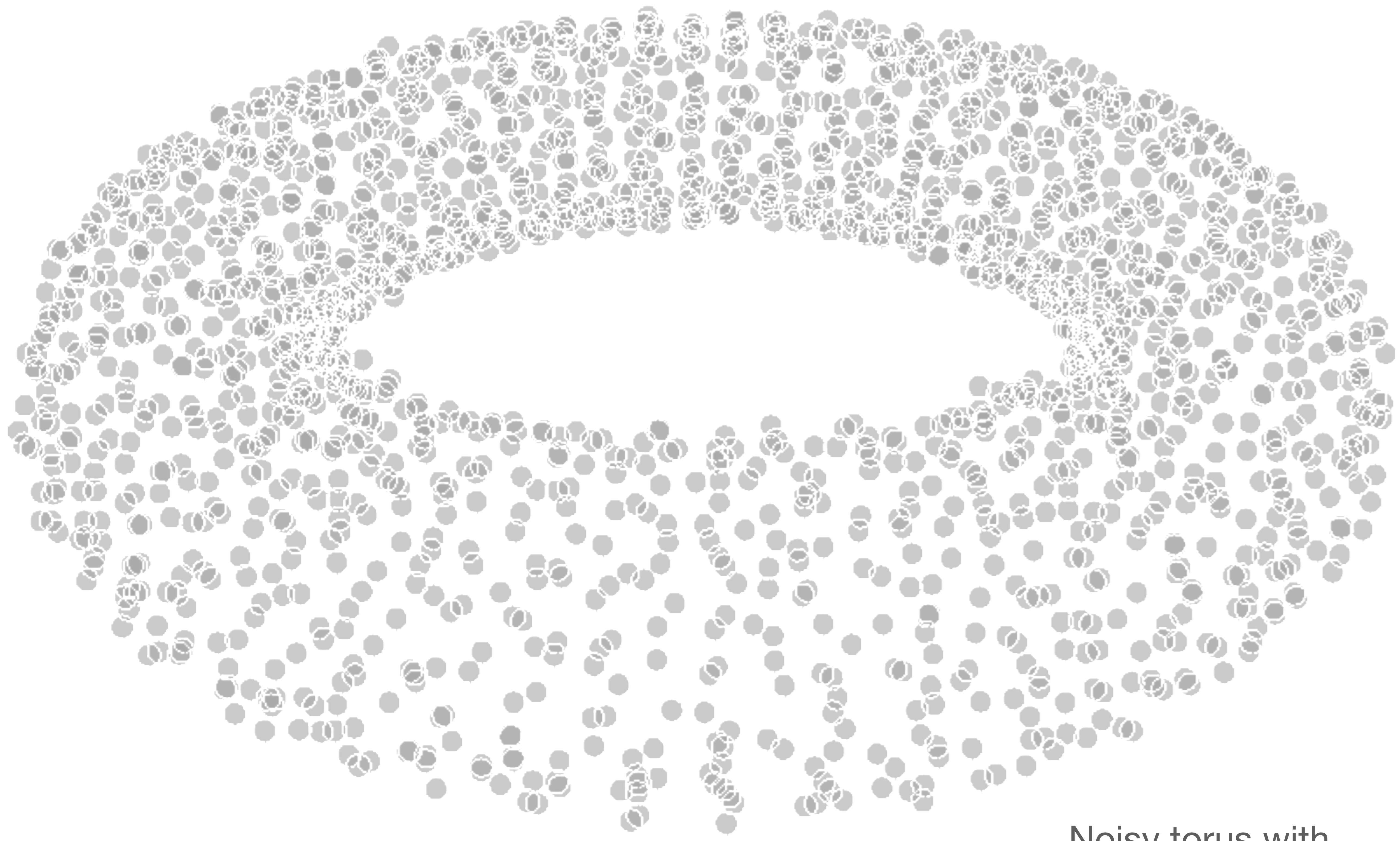


Lead to <u>geometric</u> interpretation of structural features

Interpretability: from *algebraic* summary to *local* structure

Homology generators: subsets of data giving rise to <u>topological</u> classes, *i.e.* those points forming **cycles** representing **homology classes**



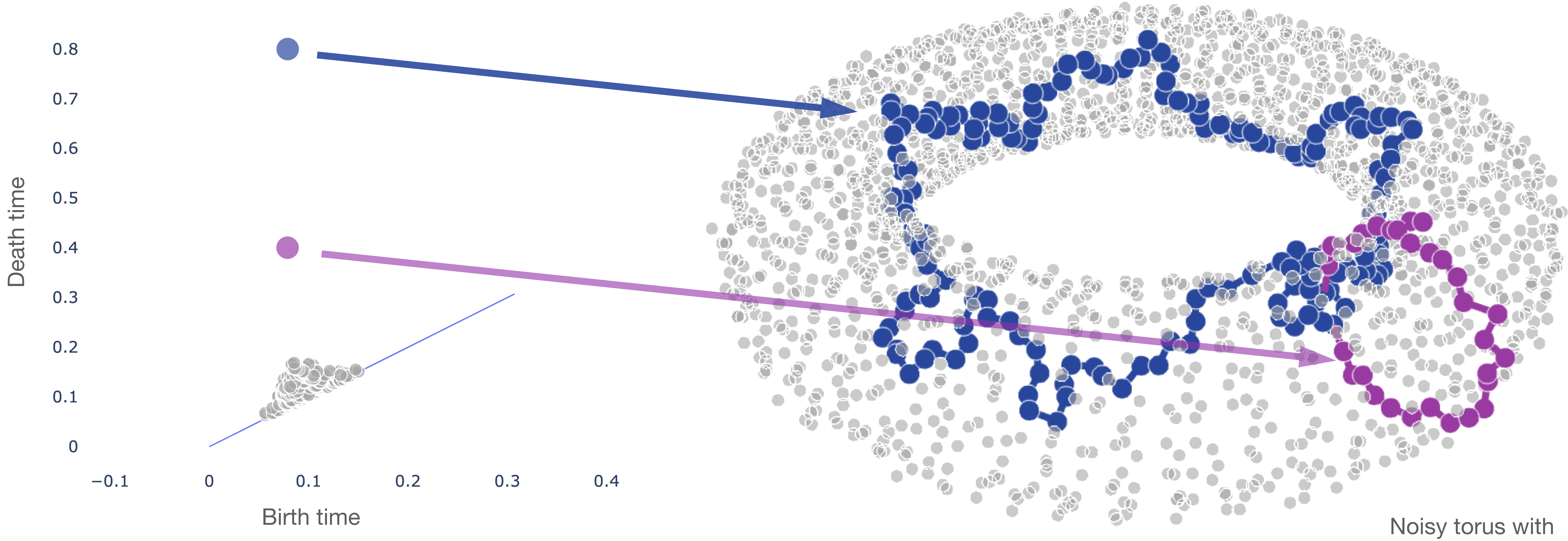


Noisy torus with homology generator

Lead to <u>geometric</u> interpretation of structural features

Interpretability: from *algebraic* summary to *local* structure

Homology generators: subsets of data giving rise to topological classes, *i.e.* those points forming **cycles** representing **homology classes**



Noisy torus with homology generator

Lead to geometric interpretation of structural features

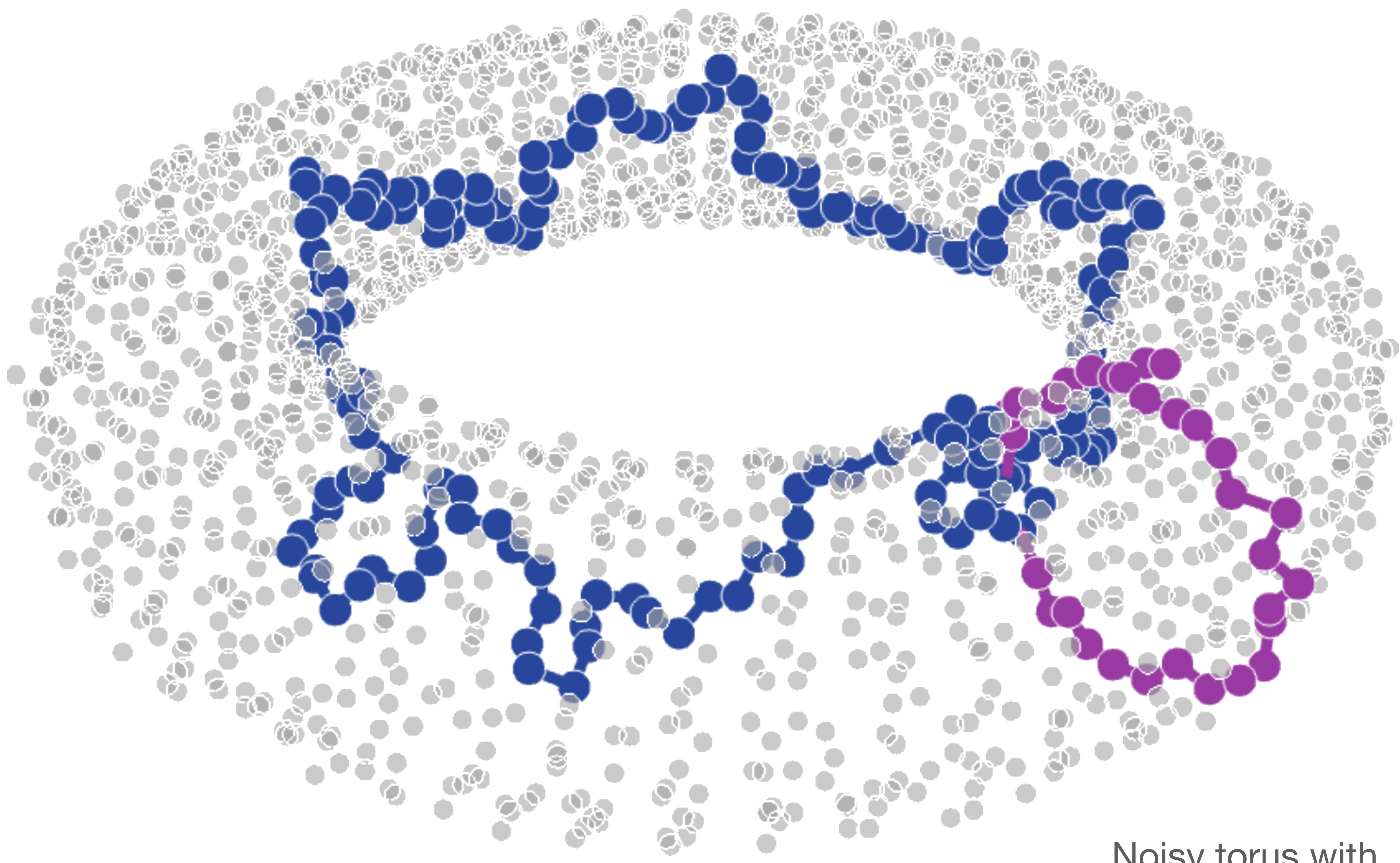
Interpretability: from *algebraic* summary to *local* structure

Homology generators: subsets of data giving rise to topological classes, *i.e.* those points forming **cycles** representing **homology classes**

Challenge 1

a) Homology generators are **not unique**: their analysis might introduce **biases**

b) Finding **optimal** cycles is **NP-hard**[1]: there is no natural **preferred choice**



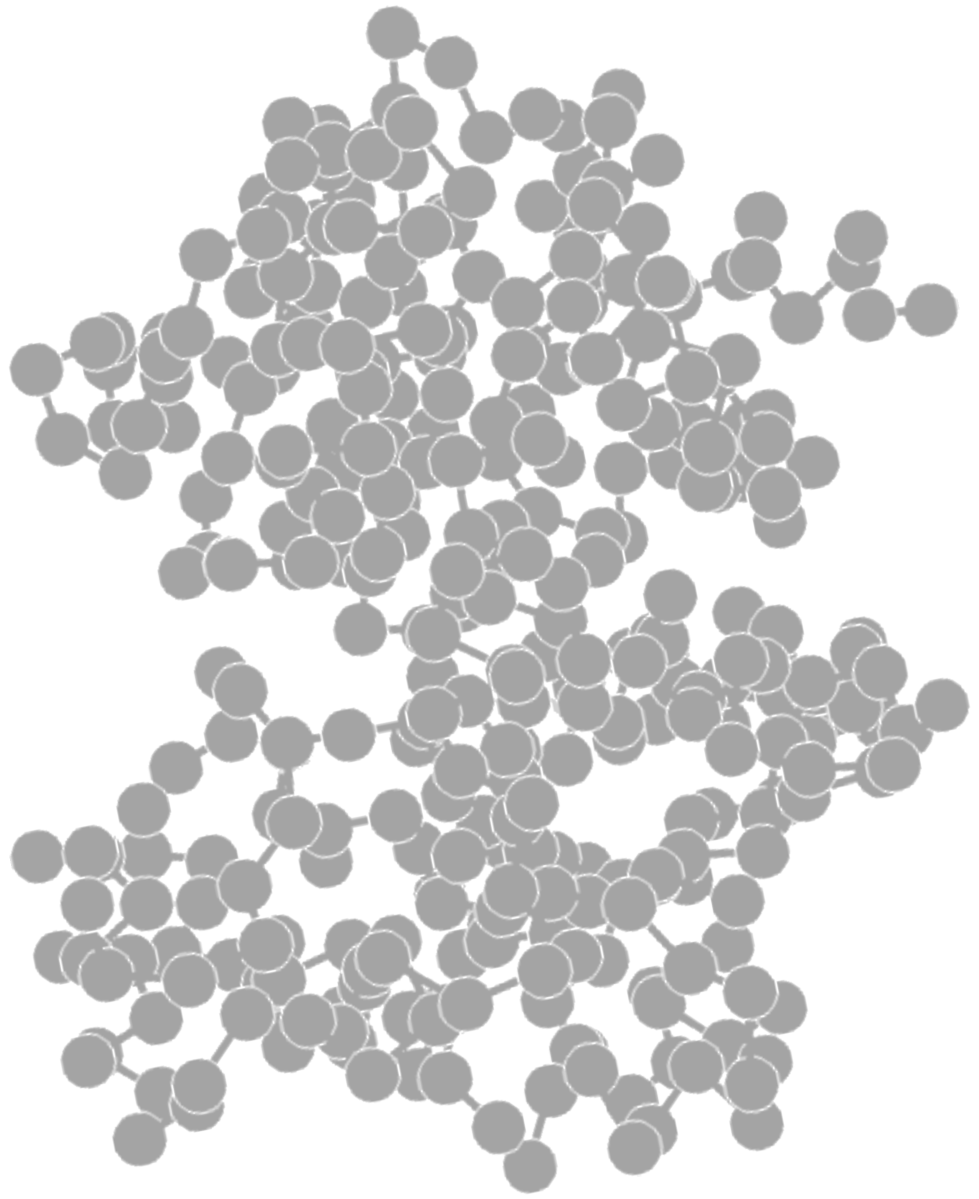
Noisy torus with homology generator

Lead to geometric interpretation of structural features

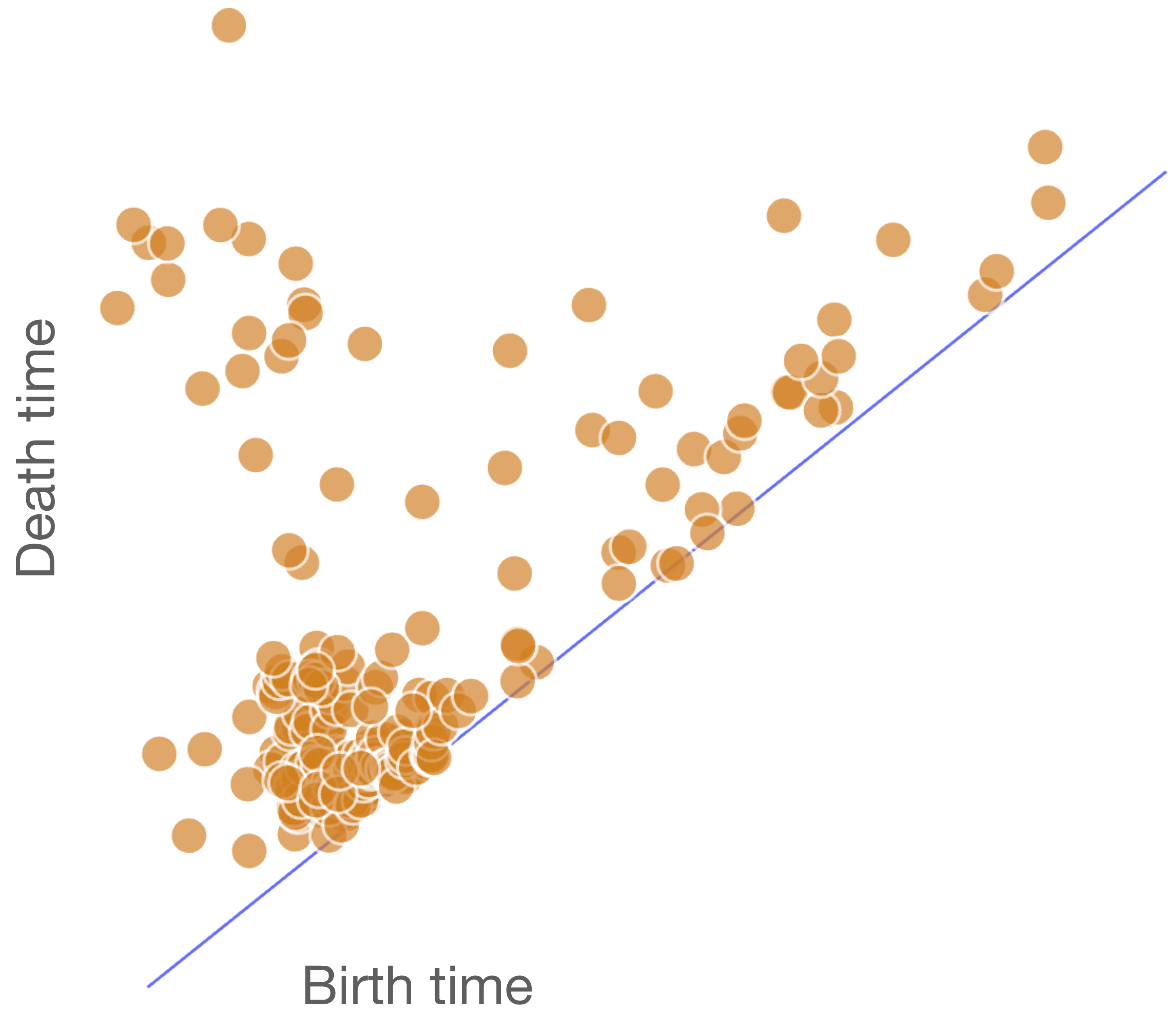
1) Li, Lu, et al. *Frontiers in artificial intelligence* (2021): 73.

Interpretability: noisy homology classes

Protein structure



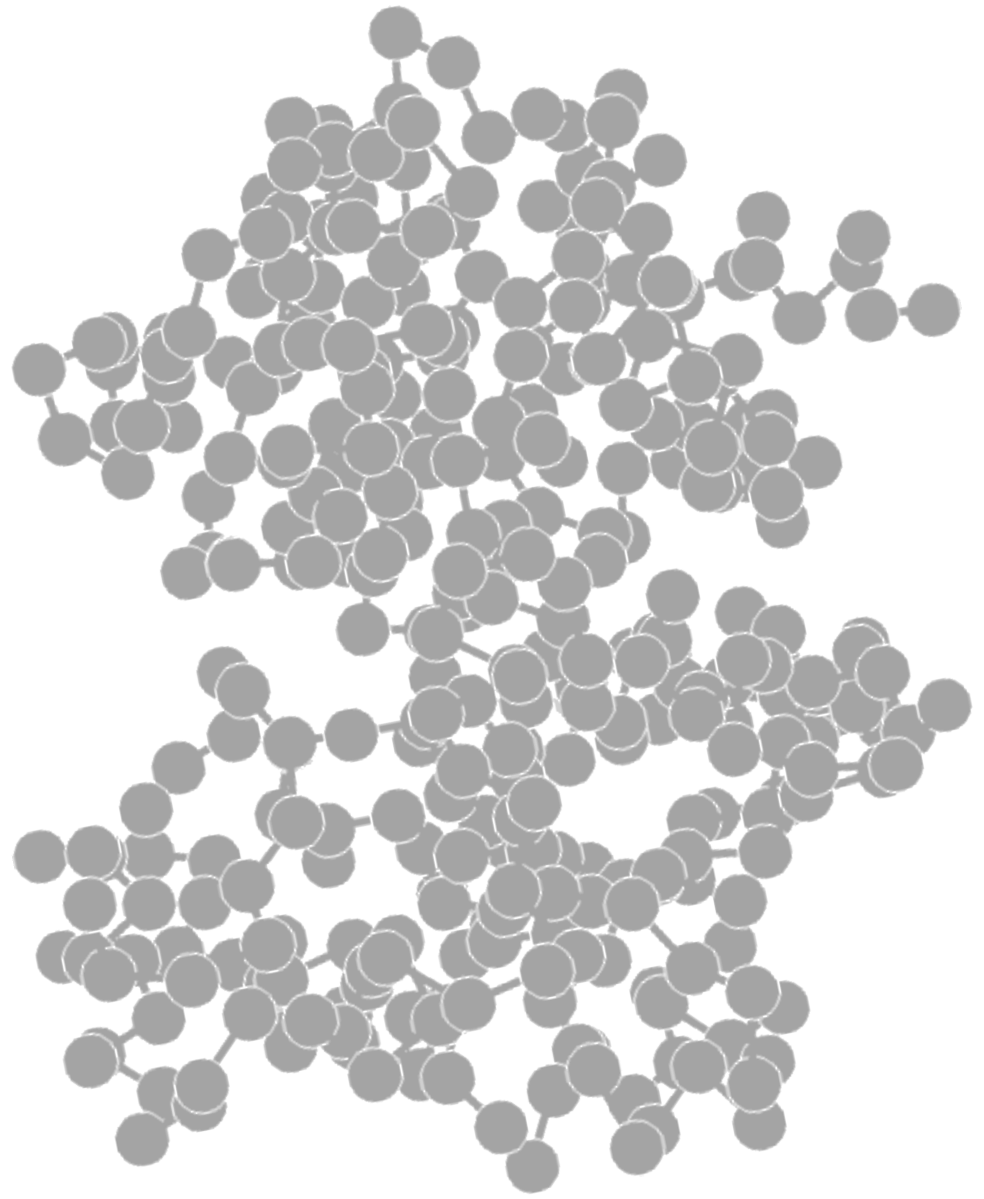
Point Cloud



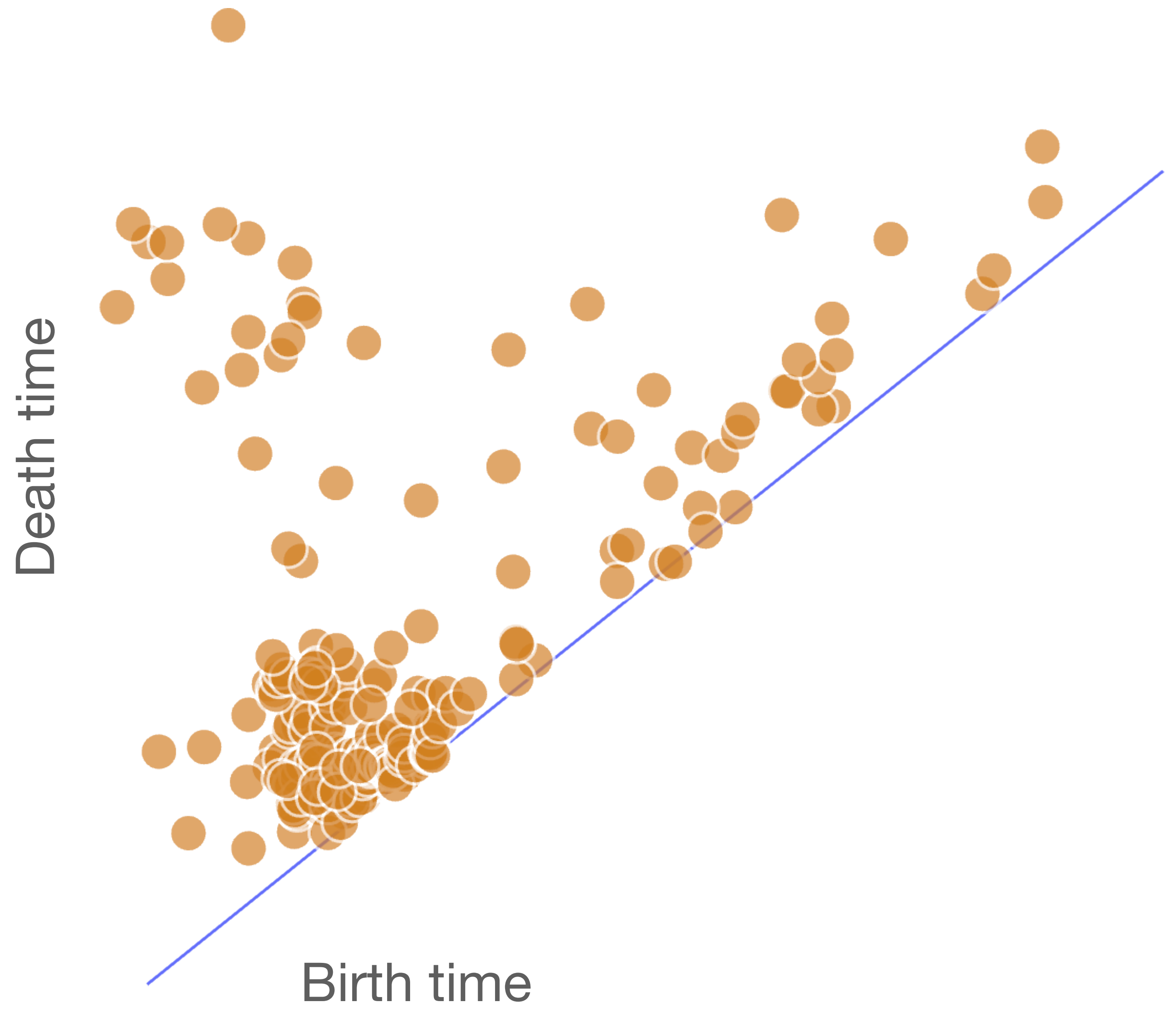
Persistent diagram

Interpretability: noisy homology classes

Protein structure



Point Cloud



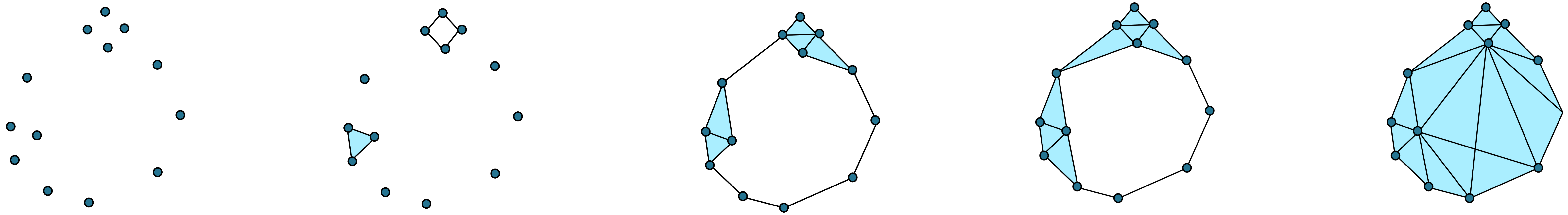
Persistent diagram

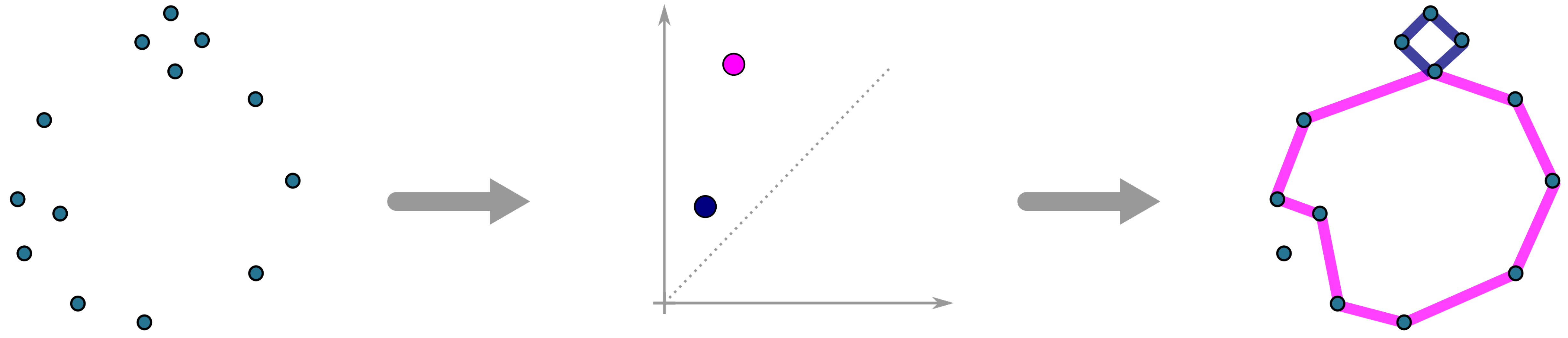
Challenge 2

a) How to interpret **complicated** and **diffused** persistence diagrams?

b) How to **capture** information from **noisy homology classes**?

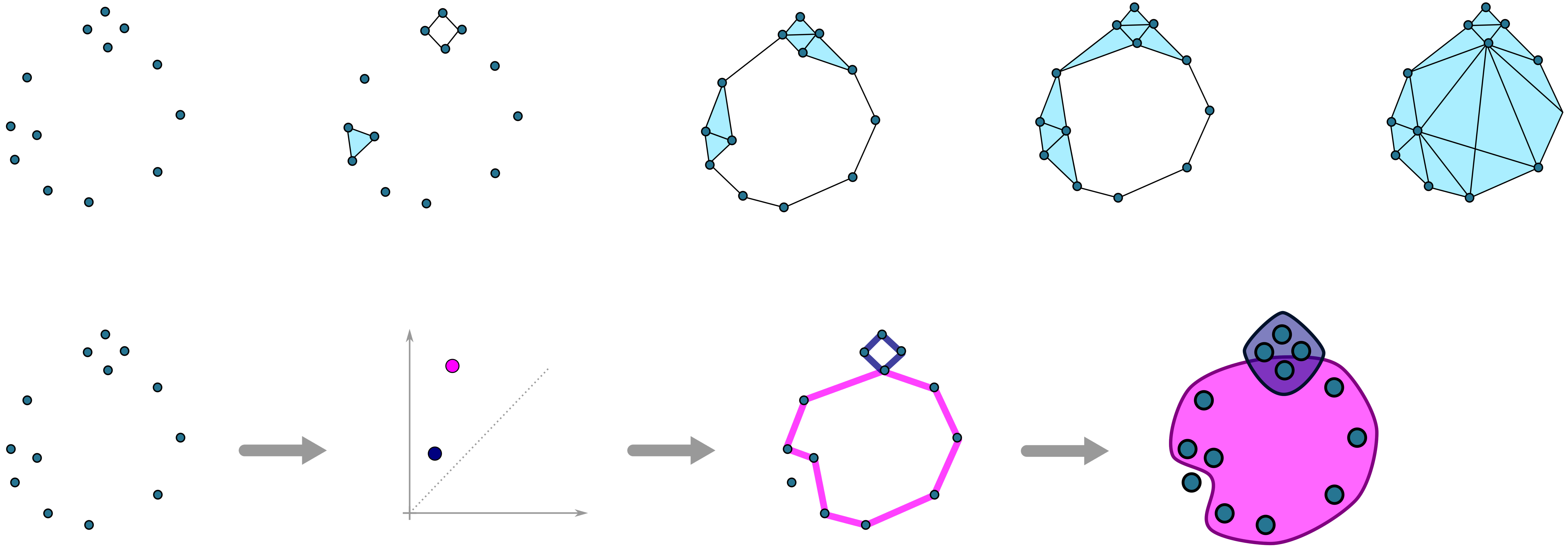
Persistent homology and the PH-hypergraph





Data Persistent diagram Homology generators

Persistent homology and the PH-hypergraph



Data | Persistent diagram | Homology generators | PH-hypergraph

CD.Madsen

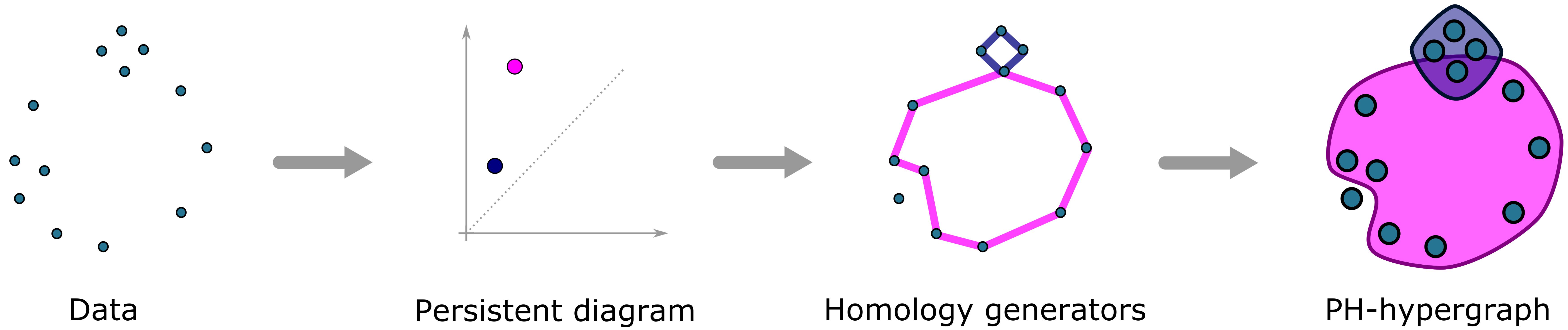
HR.Yoon

DO.Ajayi

MPH.Stumpf

HA.Harrington

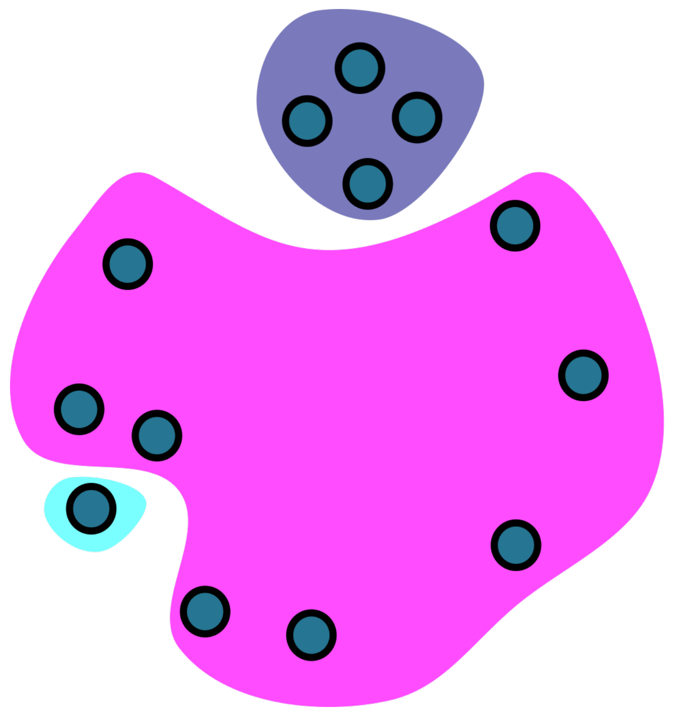
Persistent homology and the PH-hypergraph

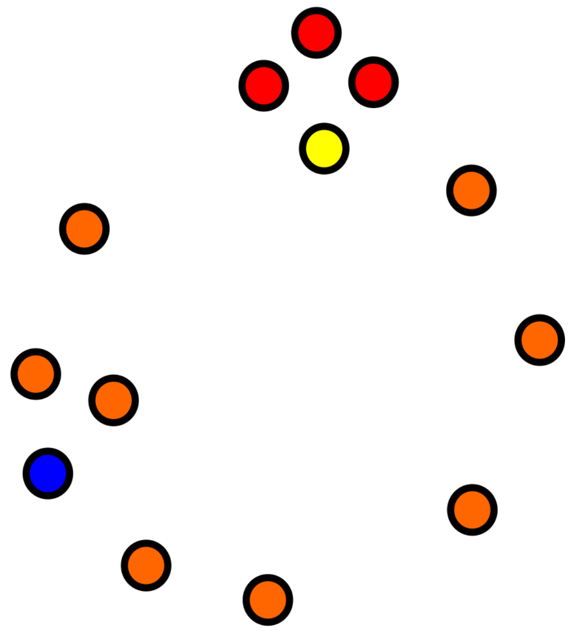


PH-communities

Community detection: find **densely-connected** groups of nodes

These correspond to a partition induced by **higher-order interactions**





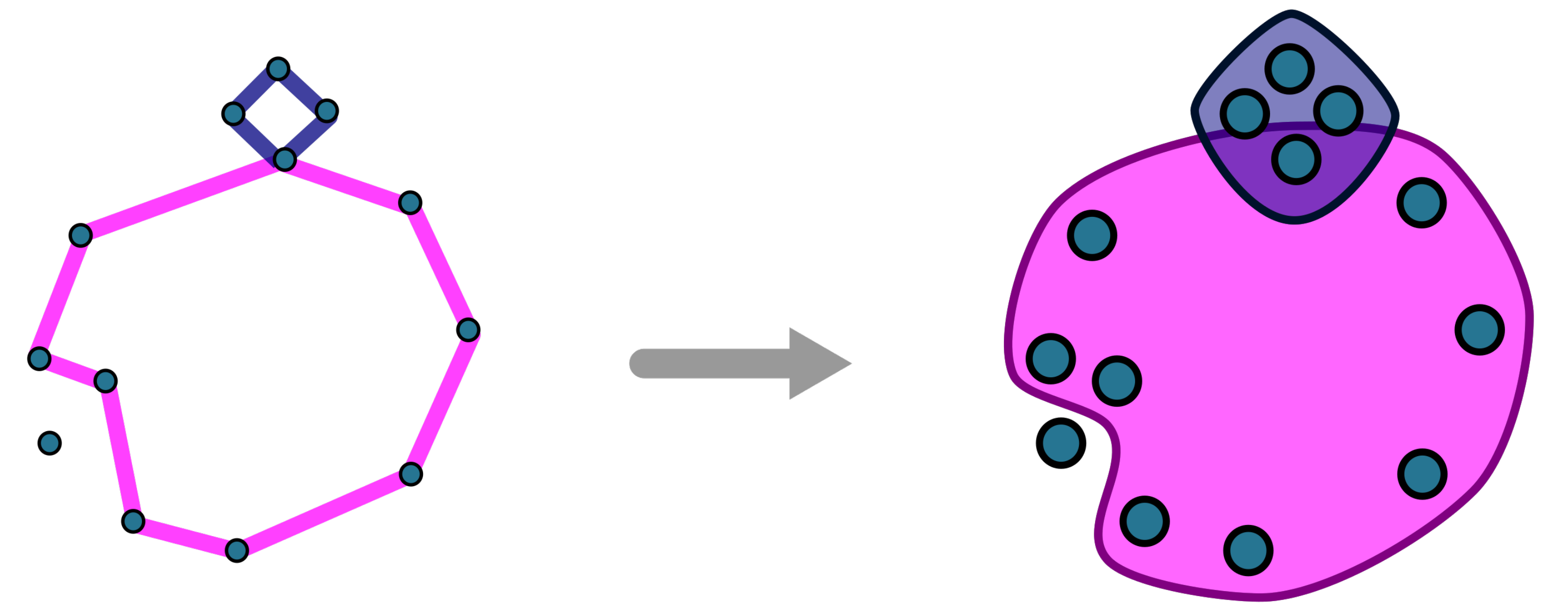
Node centrality

Rankings of nodes based on hyperedge membership and significance

The importance of a **node** depends on the importance of its **connections**

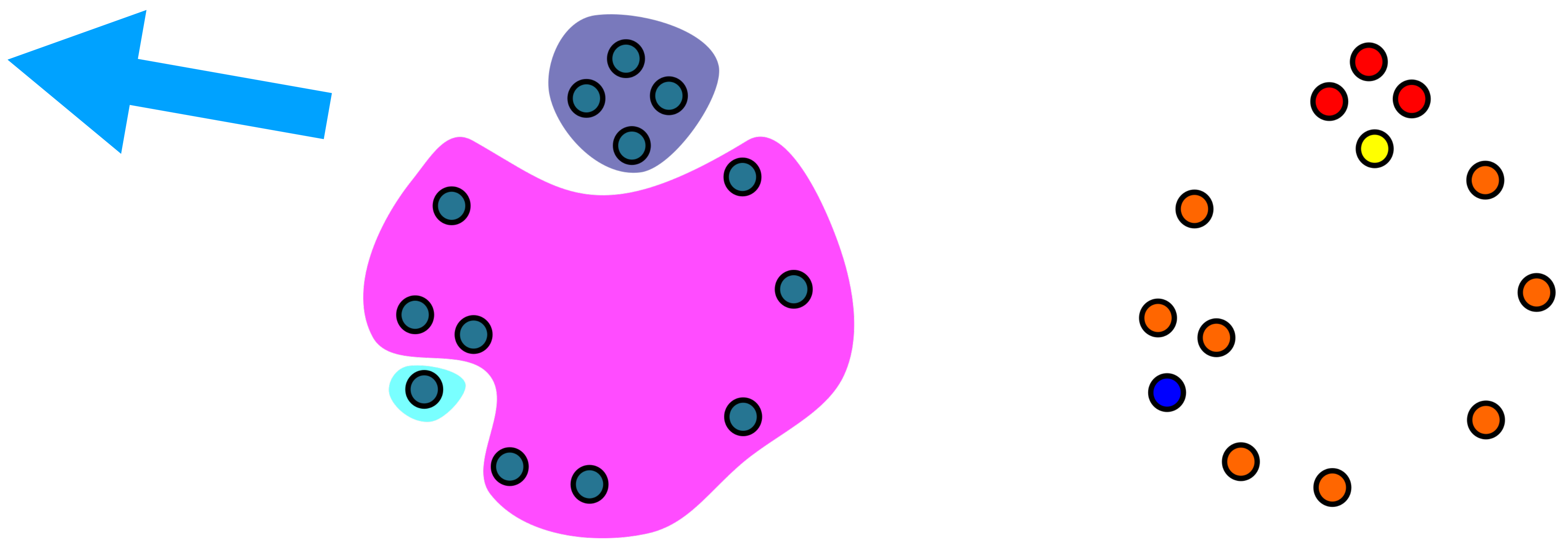
Persistent homology and the PH-hypergraph

PH-communities & **centrality** are **robust** to **noisy** data



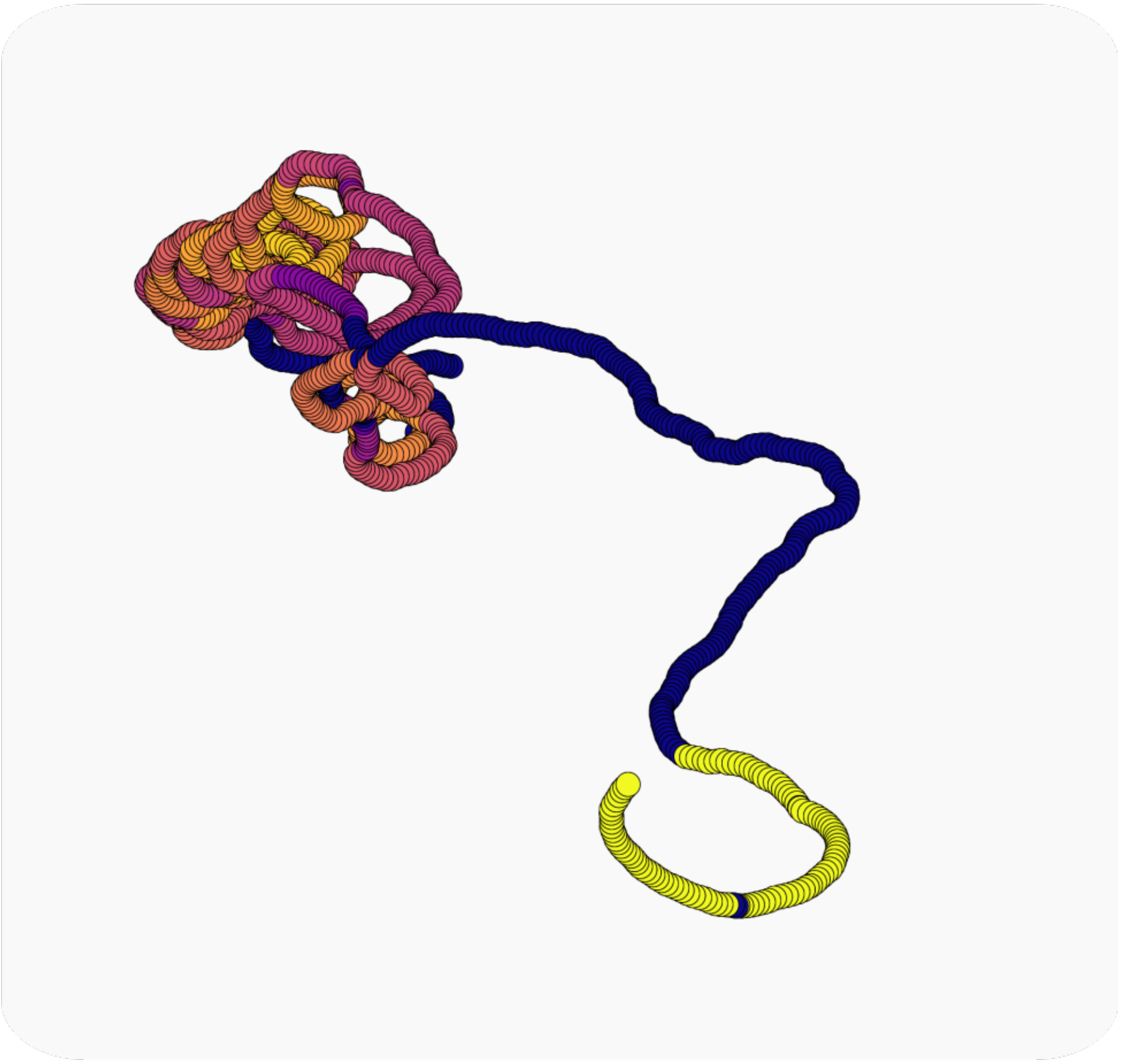
Homology generators PH-hypergraph

PH-communities & **centrality** are **stable** under different choices of homology **generators**



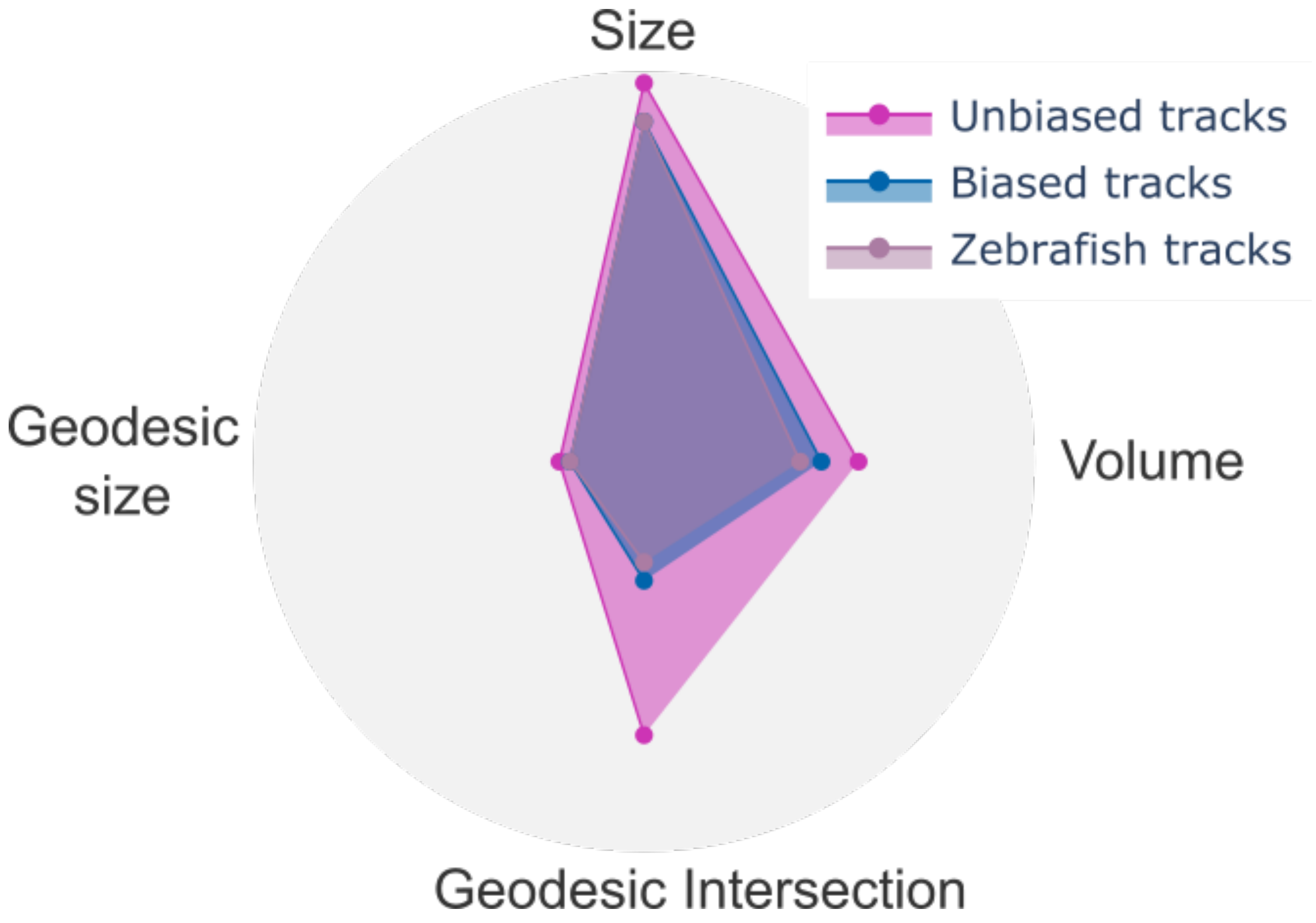
Persistent homology and the PH-hypergraph

Problem 1: fragment animal trajectories into **behavioural nodes**



Node centrality distinguishes **different behaviours** in terms of **1)** intensity of **local searches**, **2) looping** behaviour and **3) relocation**

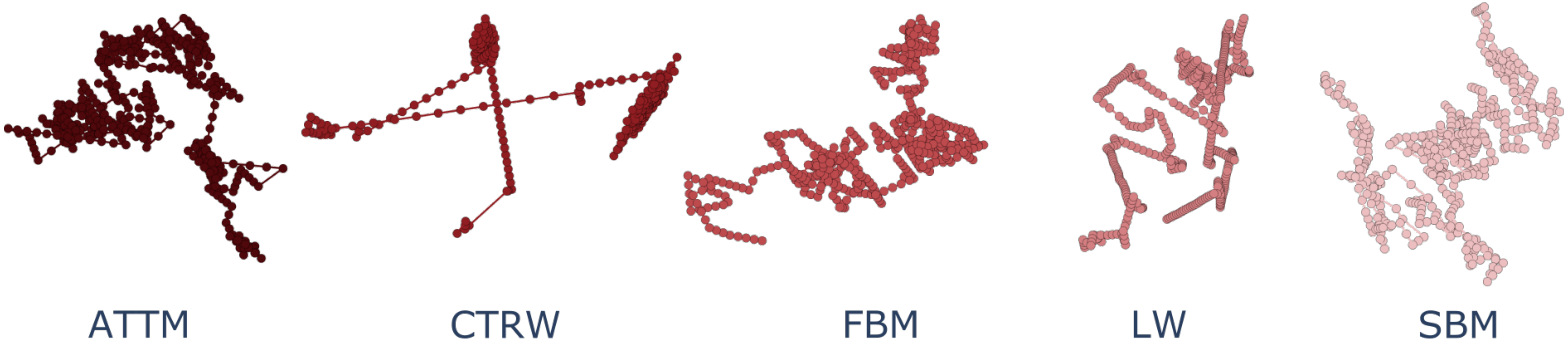
Problem 2: **detect** underlying random walk and **quantify** bias in movement



Communities analysis **1)** identifies **directional bias** in neutrophils migration towards a wound (zebrafish) and **2)** distinguishes **anomalous diffusion trajectories**

Anomalous diffusion trajectories

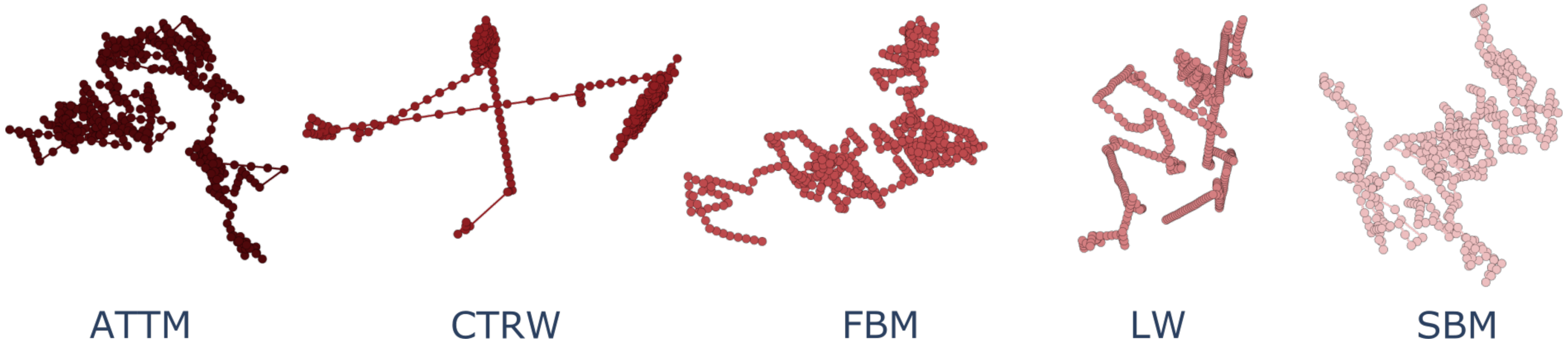
Anomalous diffusion: transport with MSD $\sim t^{\alpha}$: ubiquitous in nature

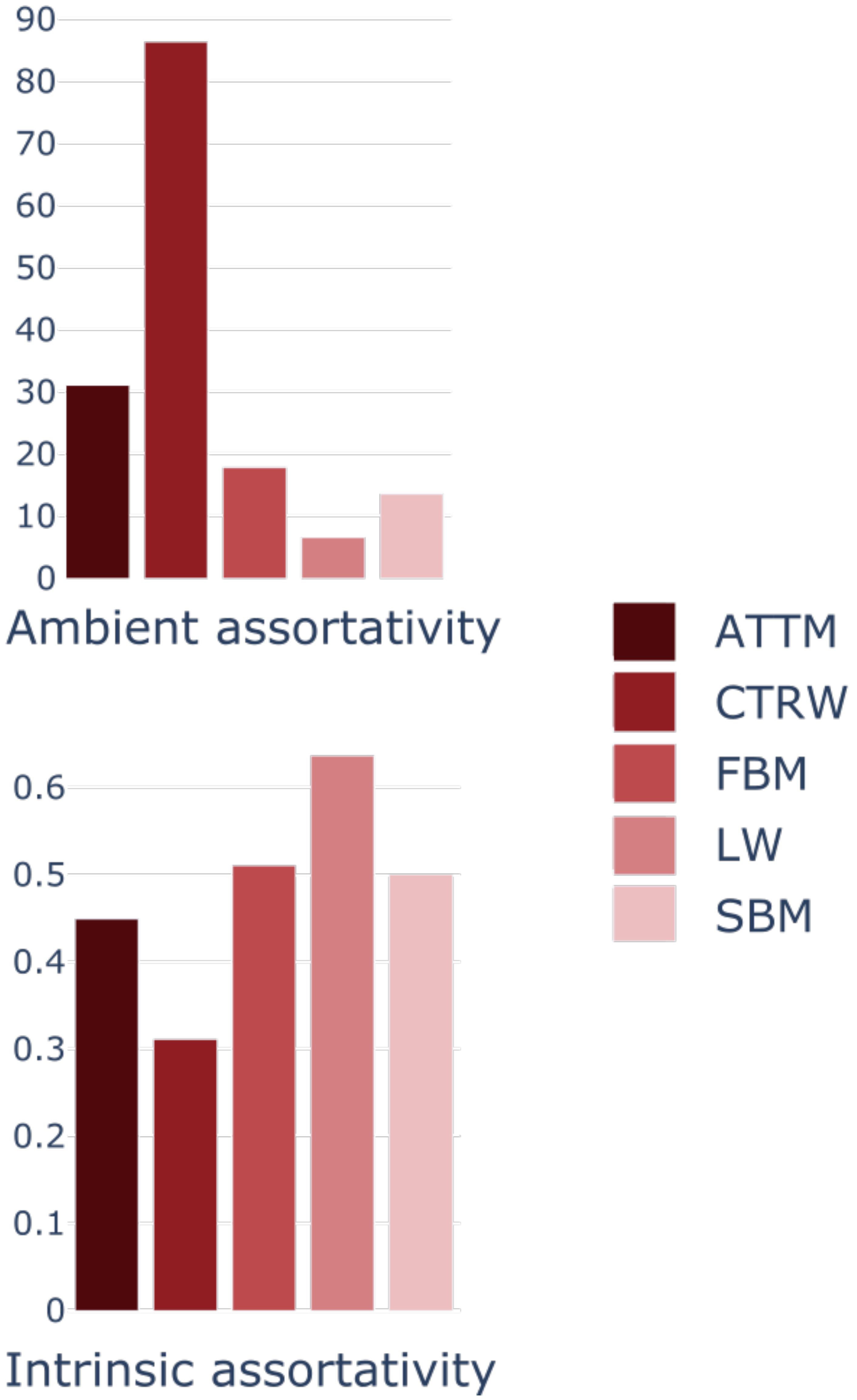


Challenge: distinguish different models from the trajectory (AnDi challenge[1,2])

1) Munoz-Gil *et al.* "**The anomalous diffusion challenge: single trajectory characterisation as a competition**" In Emerging Topics in Artificial Intelligence 2020, SPIE, 2020.
2) Munoz-Gil *et al.* "**Objective comparison of methods to decode anomalous diffusion.**" *Nature communications* 12.1

Application: AnDi models



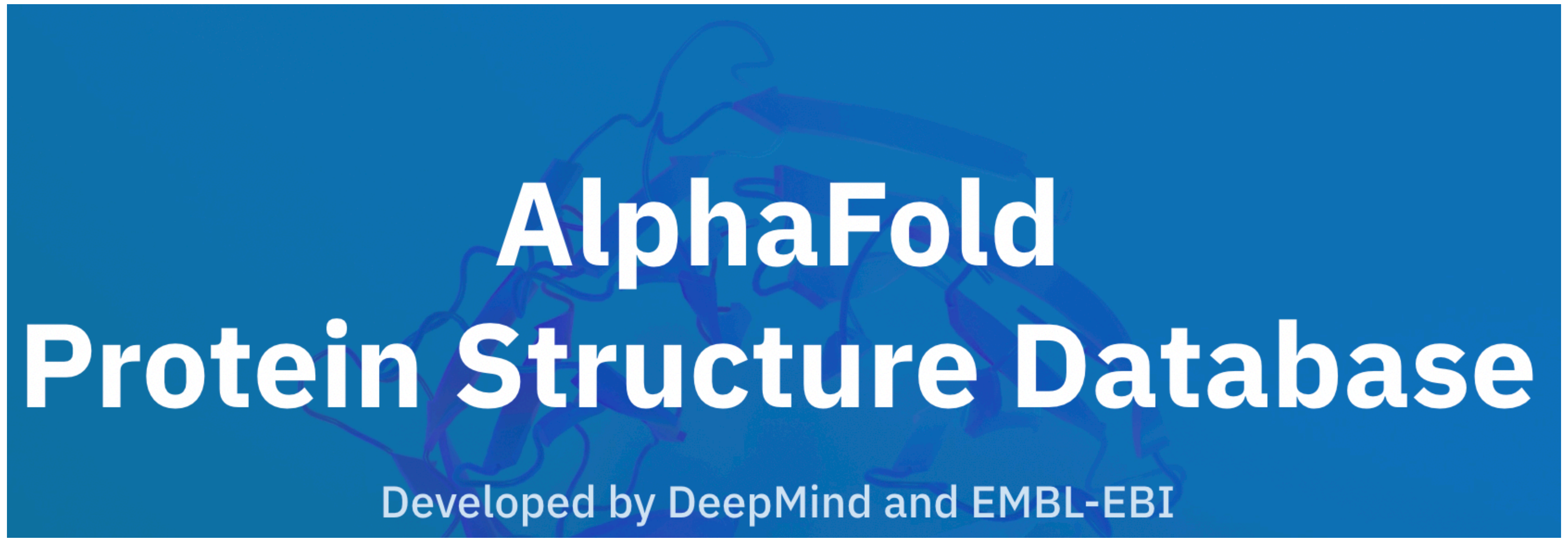


1. PH-community analysis detects **model specific** differences
2. Interpretation as **local structural features**
3. PH-communities and centrality fed to CNN **predict** underlying diffusion model with high accuracy (comparable to ranked participants in AnDi challenge[1,2])

1) Munoz-Gil *et al.* "**The anomalous diffusion challenge: single trajectory characterisation as a competition**" In Emerging Topics in Artificial Intelligence 2020, SPIE, 2020.
2) Munoz-Gil *et al.* "**Objective comparison of methods to decode anomalous diffusion.**" *Nature communications* 12.1

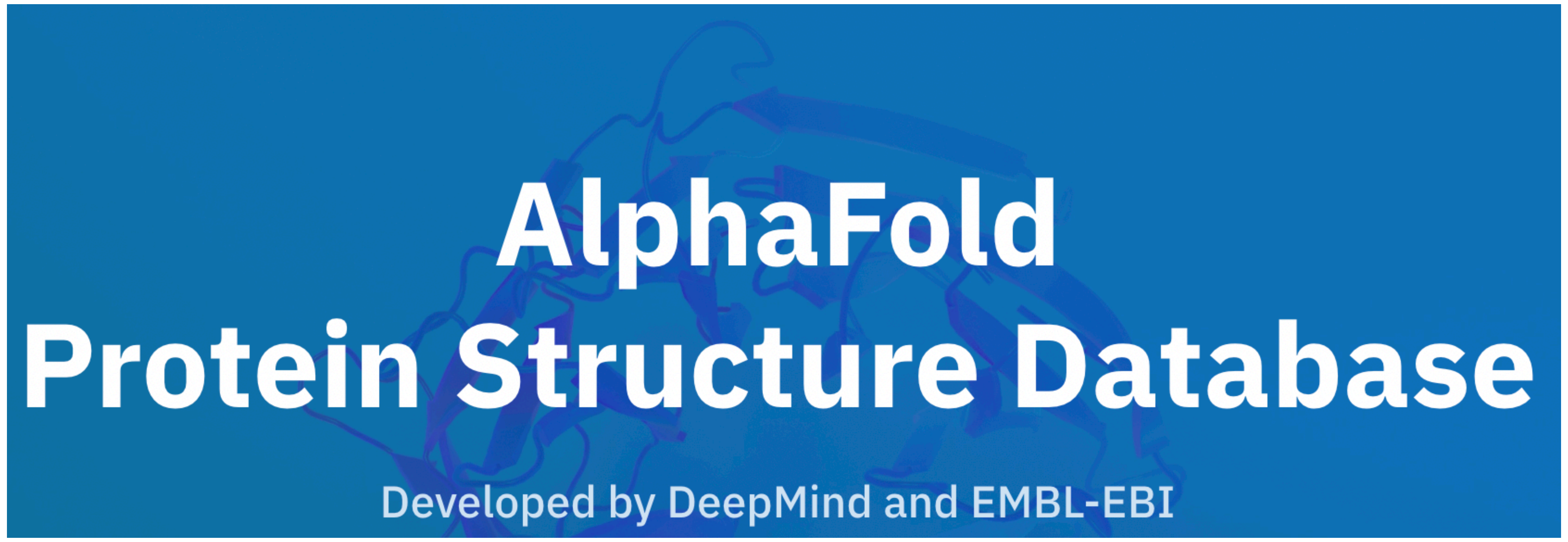
Persistent homology and the PH-hypergraph

Topological Analysis of the Protein Universe

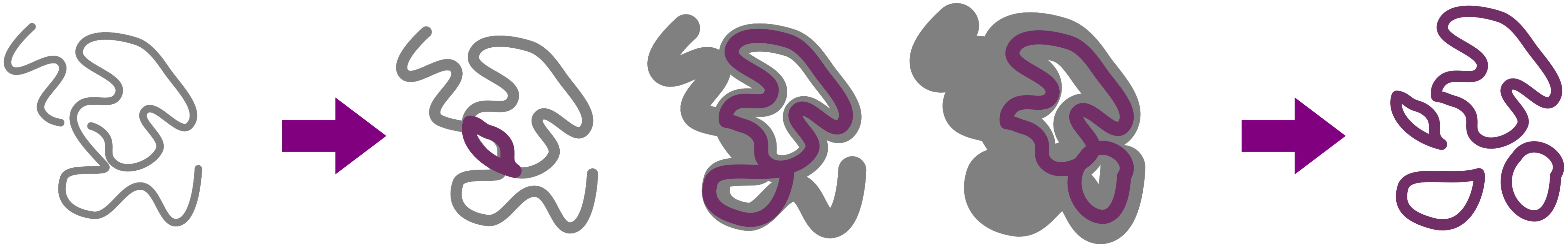


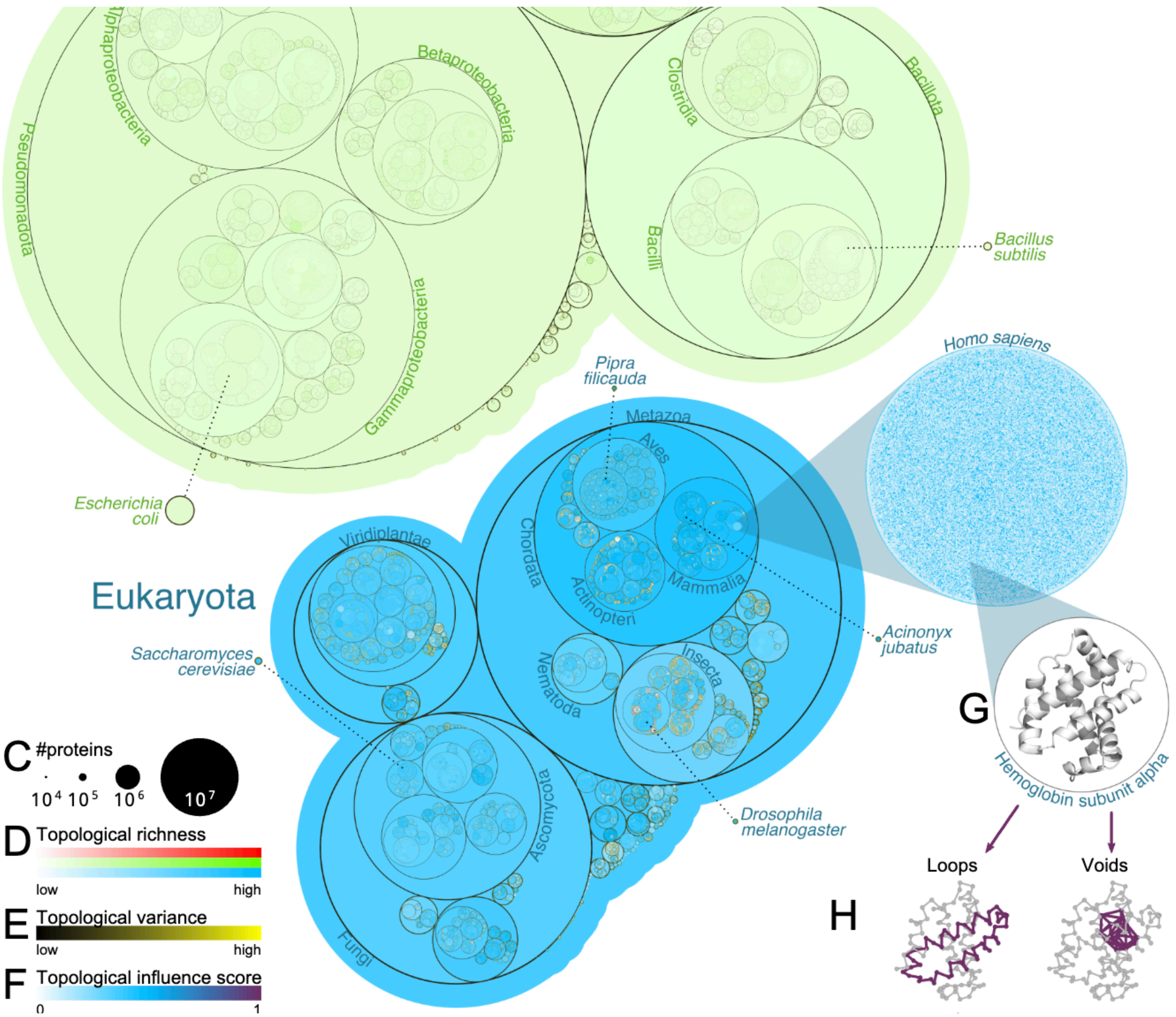
AlphaFold2: ~220 million predicted protein structures

Topological Analysis of the Protein Universe



AlphaFold2: ~220 million predicted protein structures





CD.Madsen

S.Zhang

L.Ham

AlphaFold2: ~220 million predicted protein structures

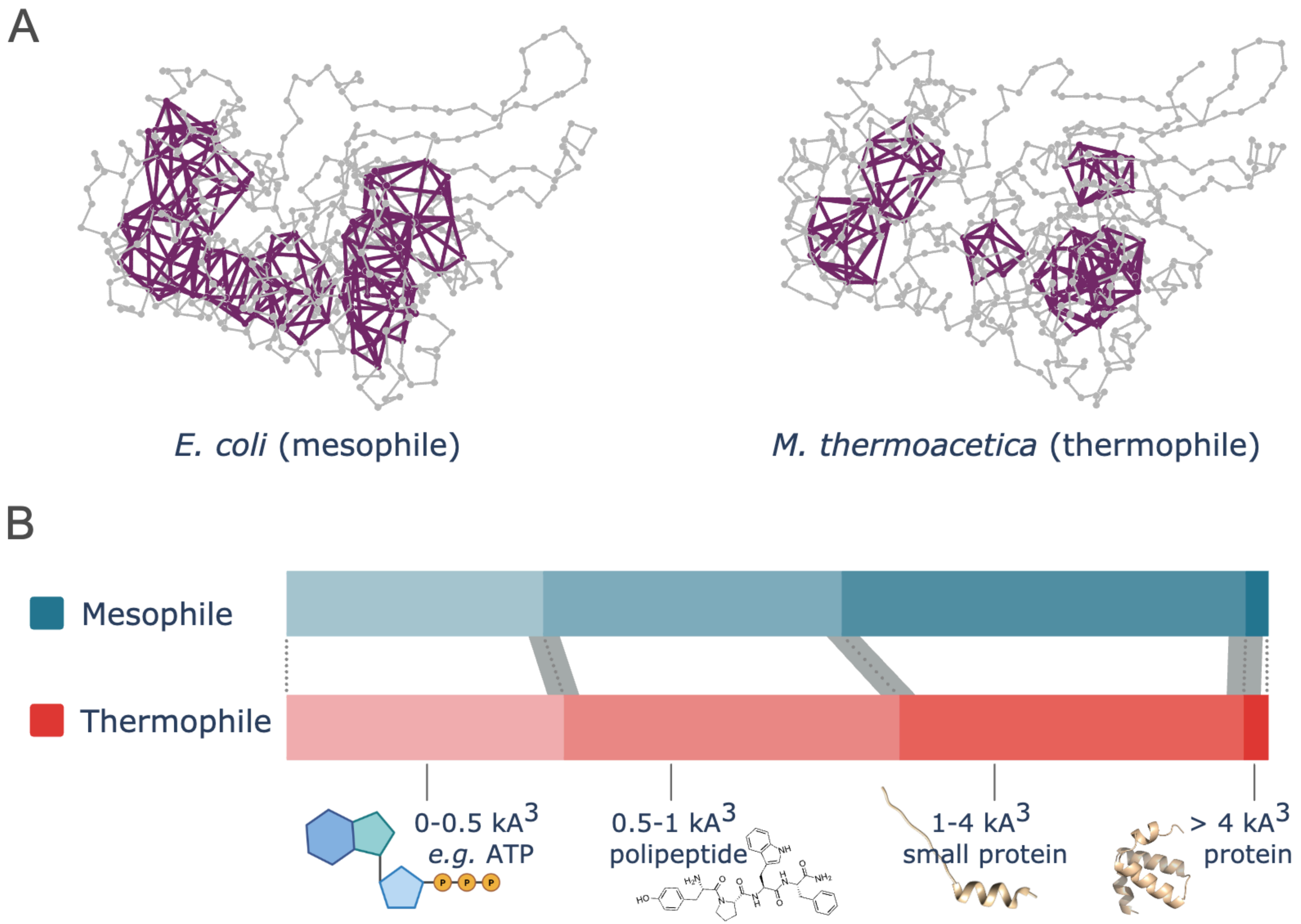
MPH.Stumpf

Analysed using persistent homology and PH-hypergraphs

D.Pires

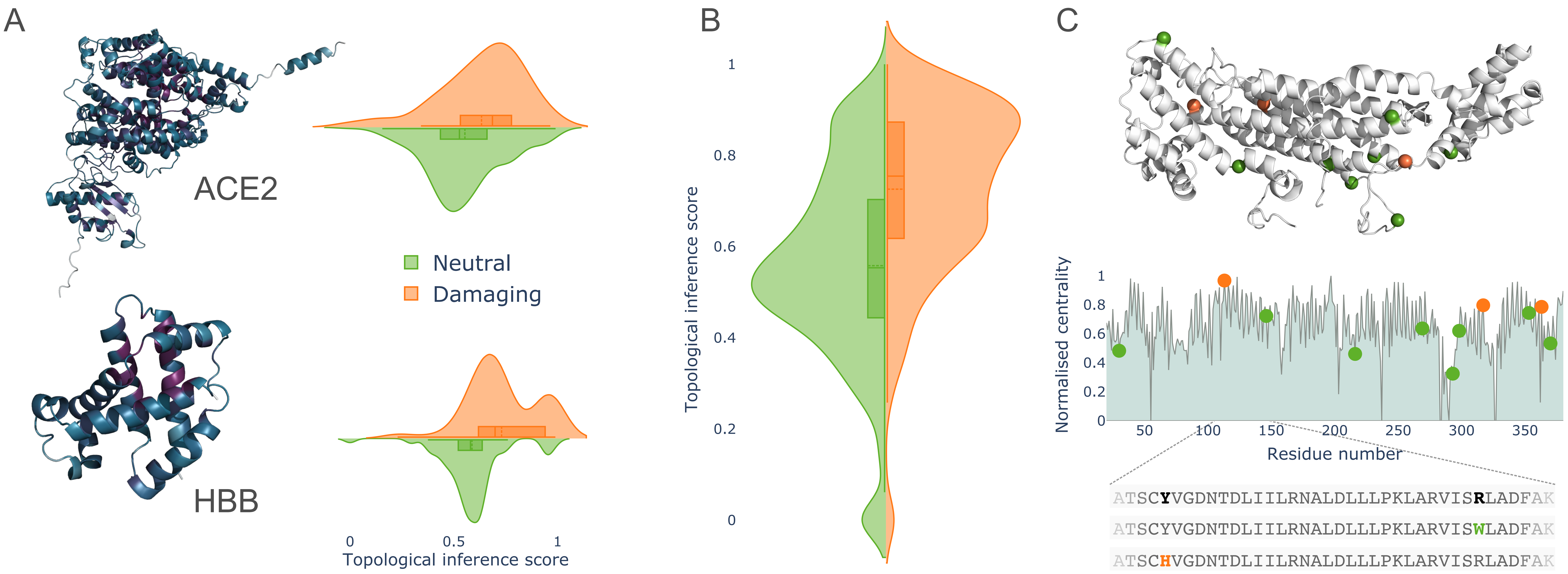
A.David

Thermophilic and mesophilic proteins are topologically different



The volume of 2-dimensional persistent classes is smaller in thermophile enzymes

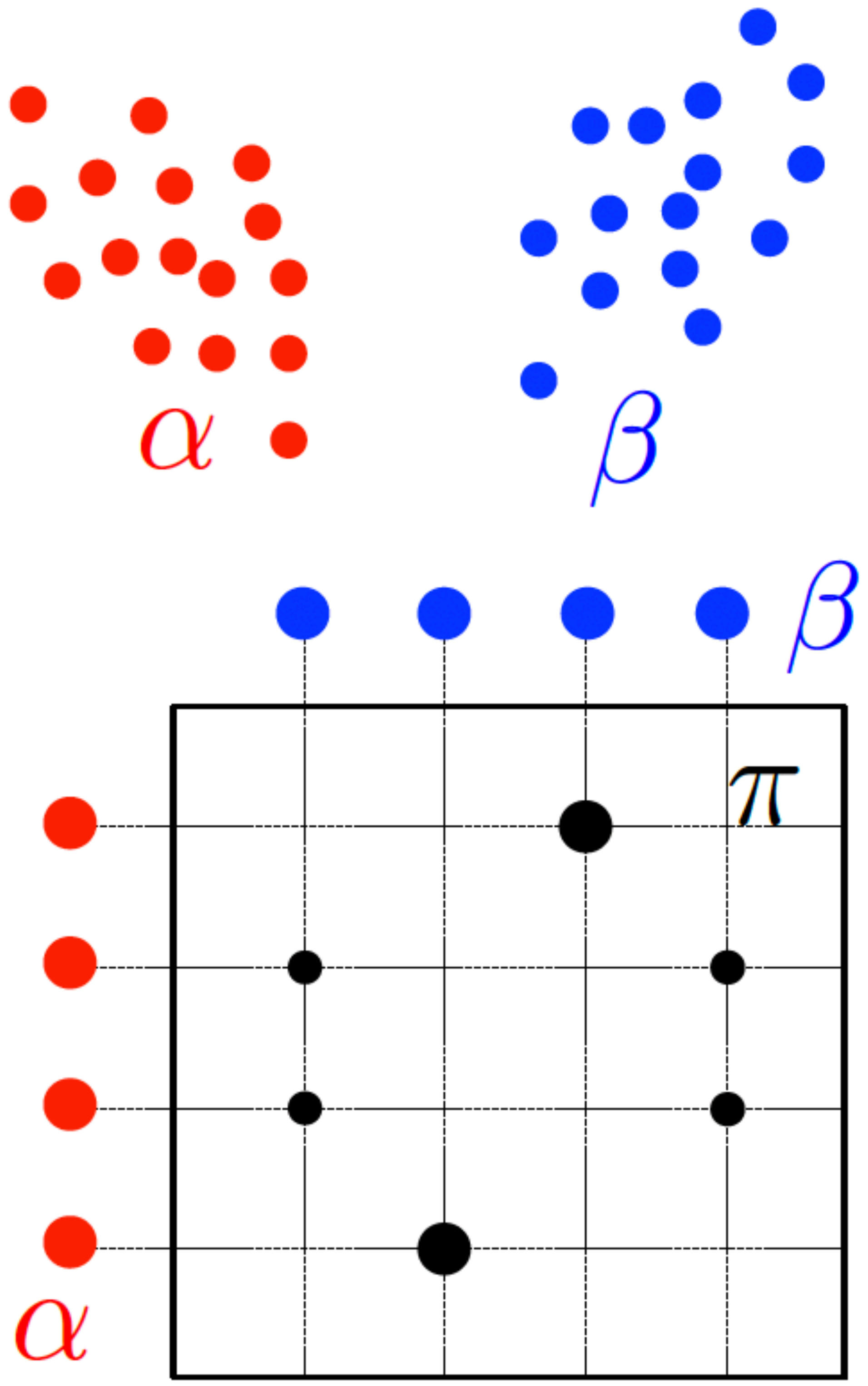
Topological features are enriched in damaging variants



Centrality is higher in residues accommodating (structurally) damaging mutations

Optimal transport

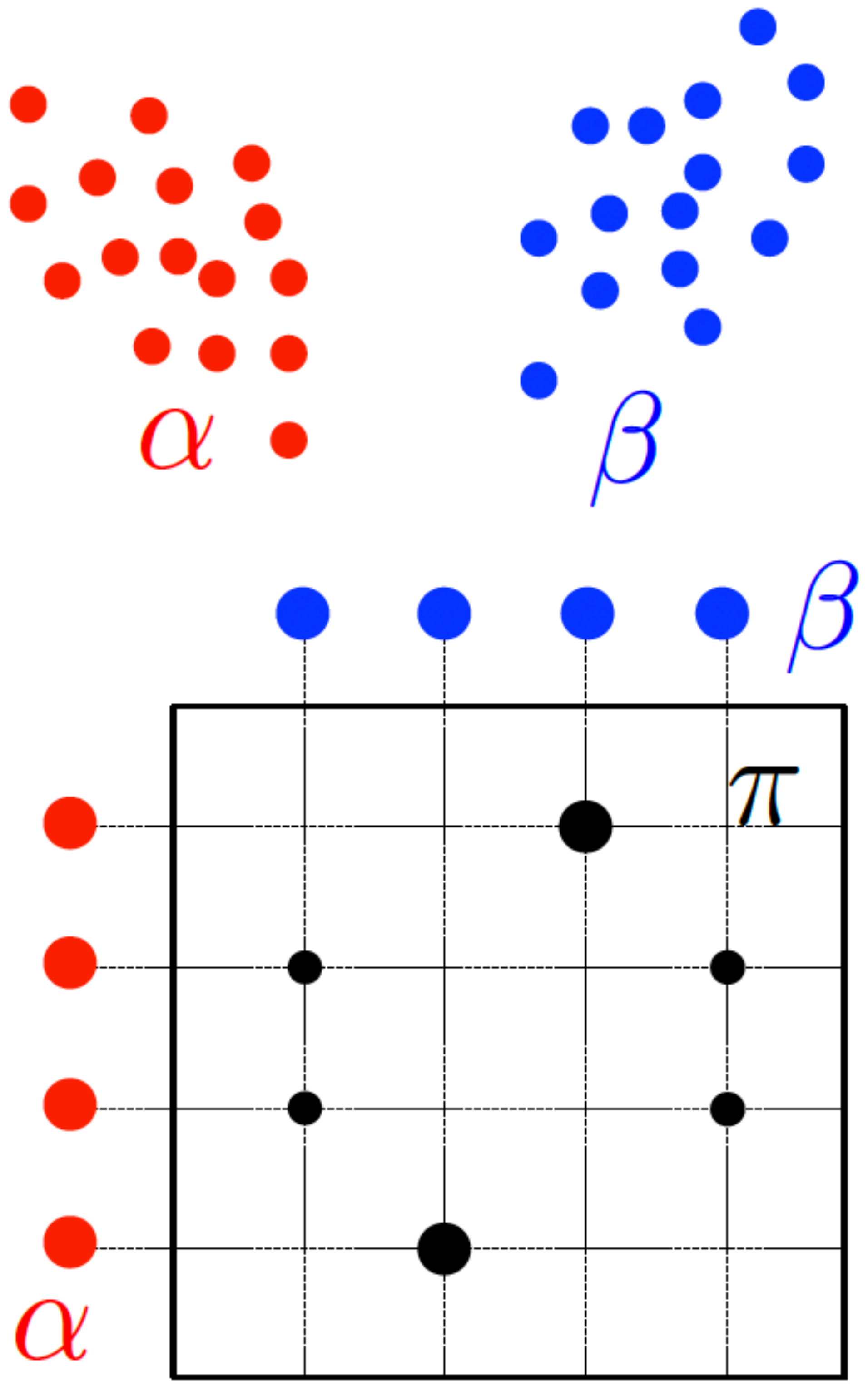
Optimal transport

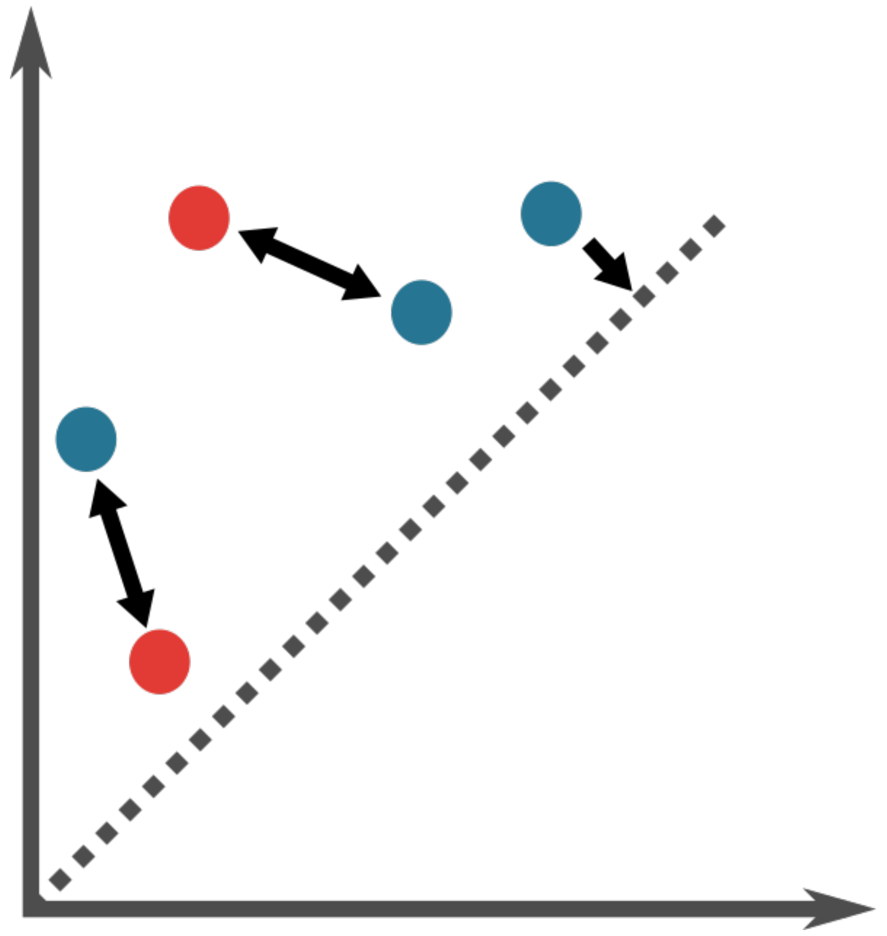


Wasserstein distances/matchings: find the matching between two distributions (point clouds) that minimises "cost" (earth mover's distance)

Figures taken from Vayer, Titouan, et al. *arXiv preprint arXiv:1811.02834* (2018) and Peyré, Gabriel, and Marco Cuturi. *Center for Research in Economics and Statistics Working Papers* 2017-86 (2017).

Optimal transport



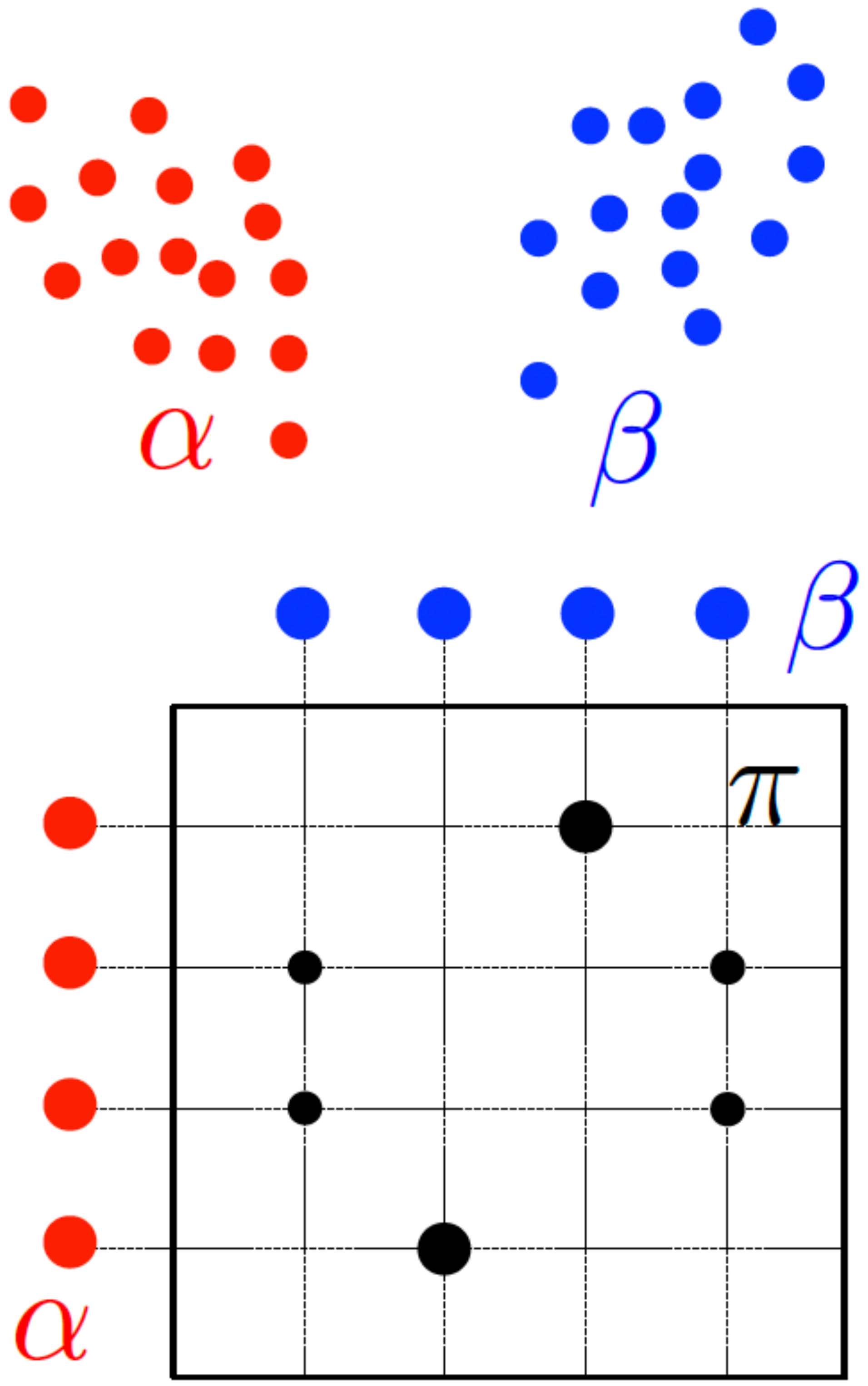


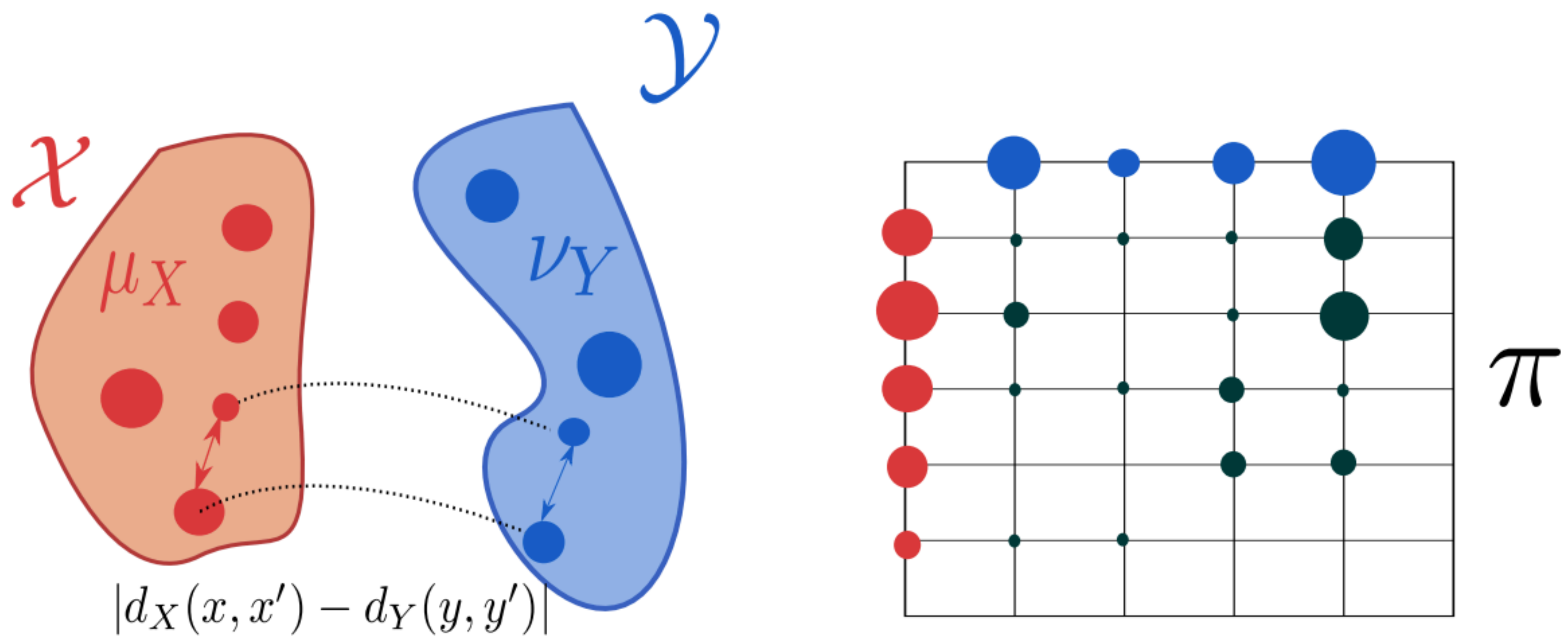
Wasserstein distances **for persistent diagrams**: matching between homology classes that minimises total distance.

Points are allowed to be matched to the diagonal

Figures taken from Vayer, Titouan, et al. *arXiv preprint arXiv:1811.02834* (2018) and Peyré, Gabriel, and Marco Cuturi. *Center for Research in Economics and Statistics Working Papers* 2017-86 (2017).

Optimal transport

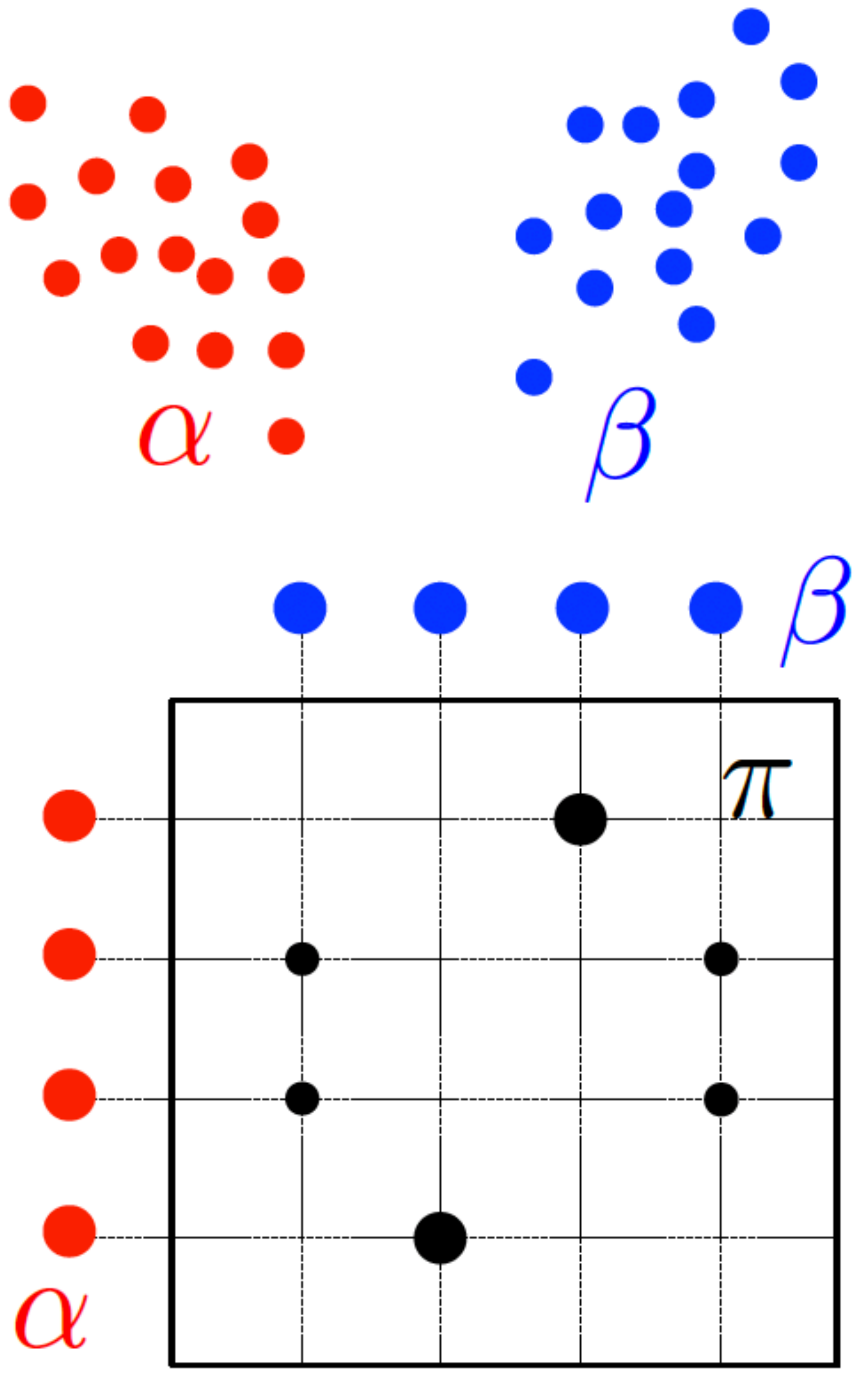


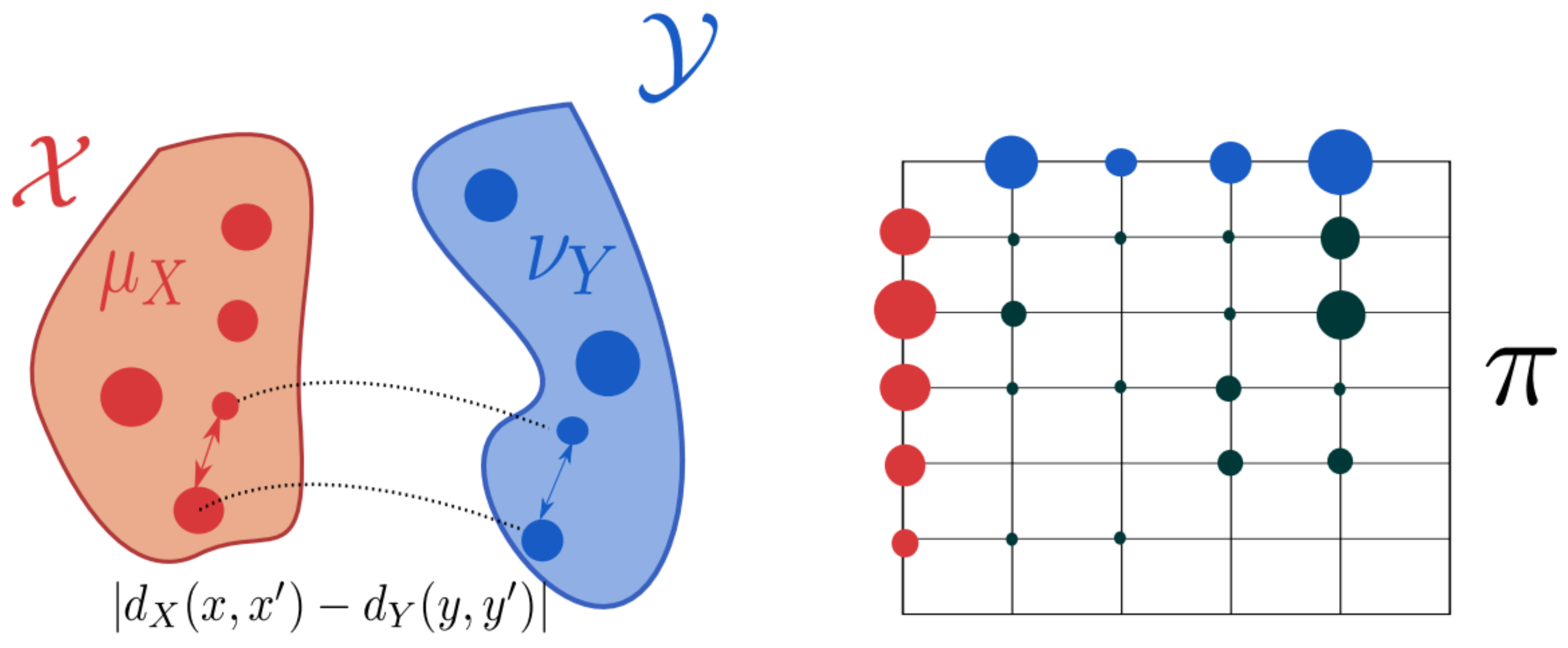


Gromov-Wasserstein:
find matching that optimally
preserves pairwise distances

Figures taken from Vayer, Titouan, et al. *arXiv preprint arXiv:1811.02834* (2018) and Peyré, Gabriel, and Marco Cuturi. *Center for Research in Economics and Statistics Working Papers* 2017-86 (2017).

Optimal transport



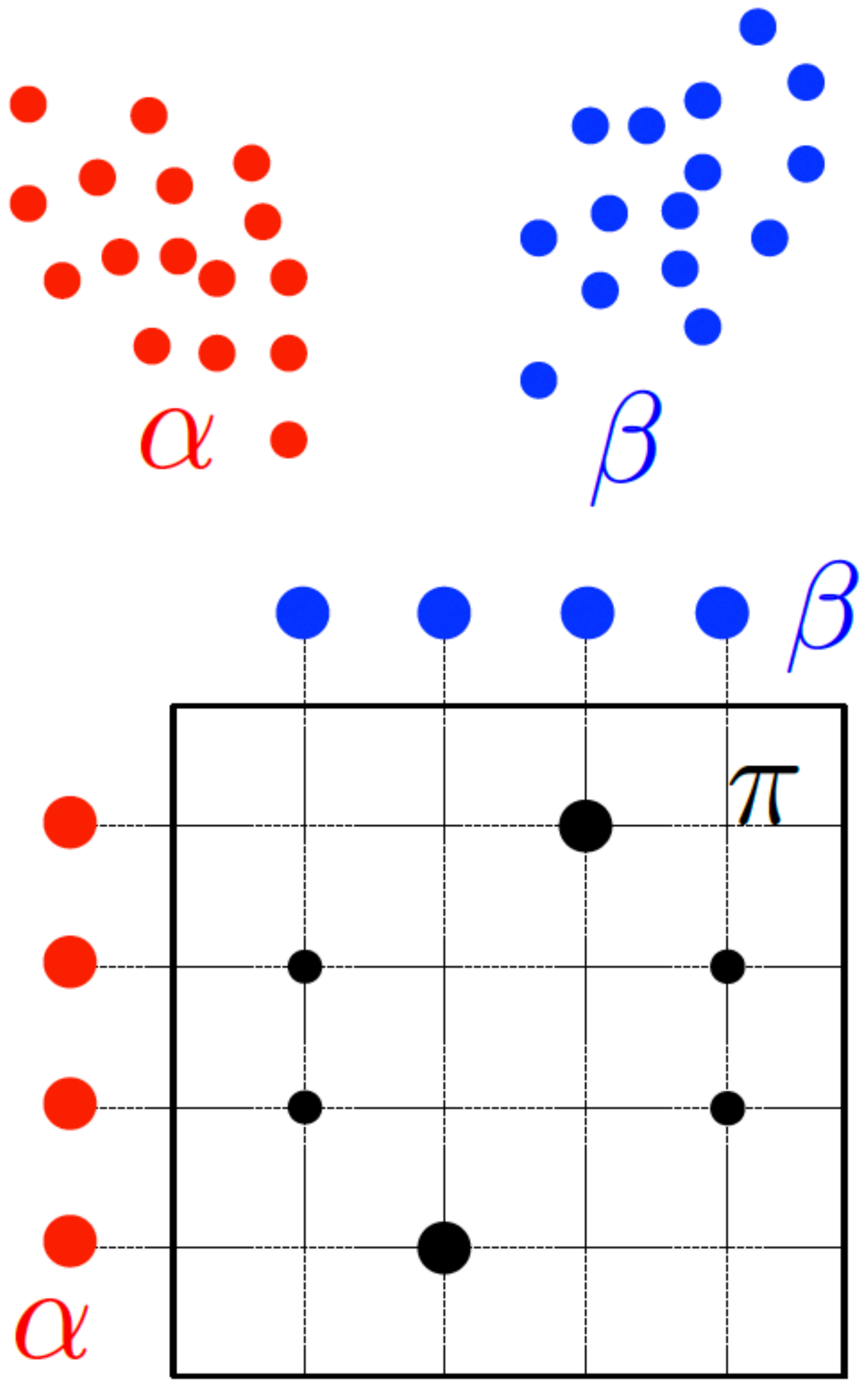


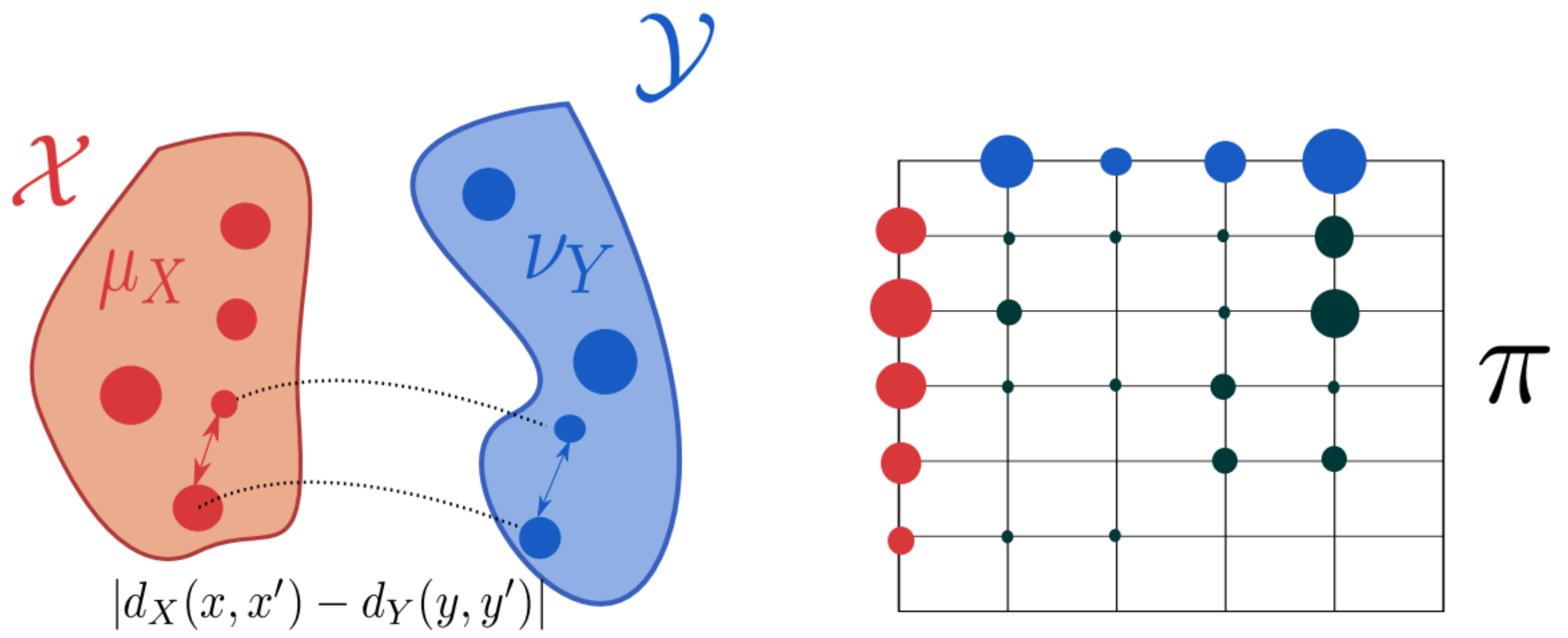
Gromov-Wasserstein:
find matching that optimally preserves pairwise distances

On **graphs**, the output is matching between **vertices** optimally preserving graph structure

Figures taken from Vayer, Titouan, et al. *arXiv preprint arXiv:1811.02834* (2018) and Peyré, Gabriel, and Marco Cuturi. *Center for Research in Economics and Statistics Working Papers* 2017-86 (2017).

Optimal transport



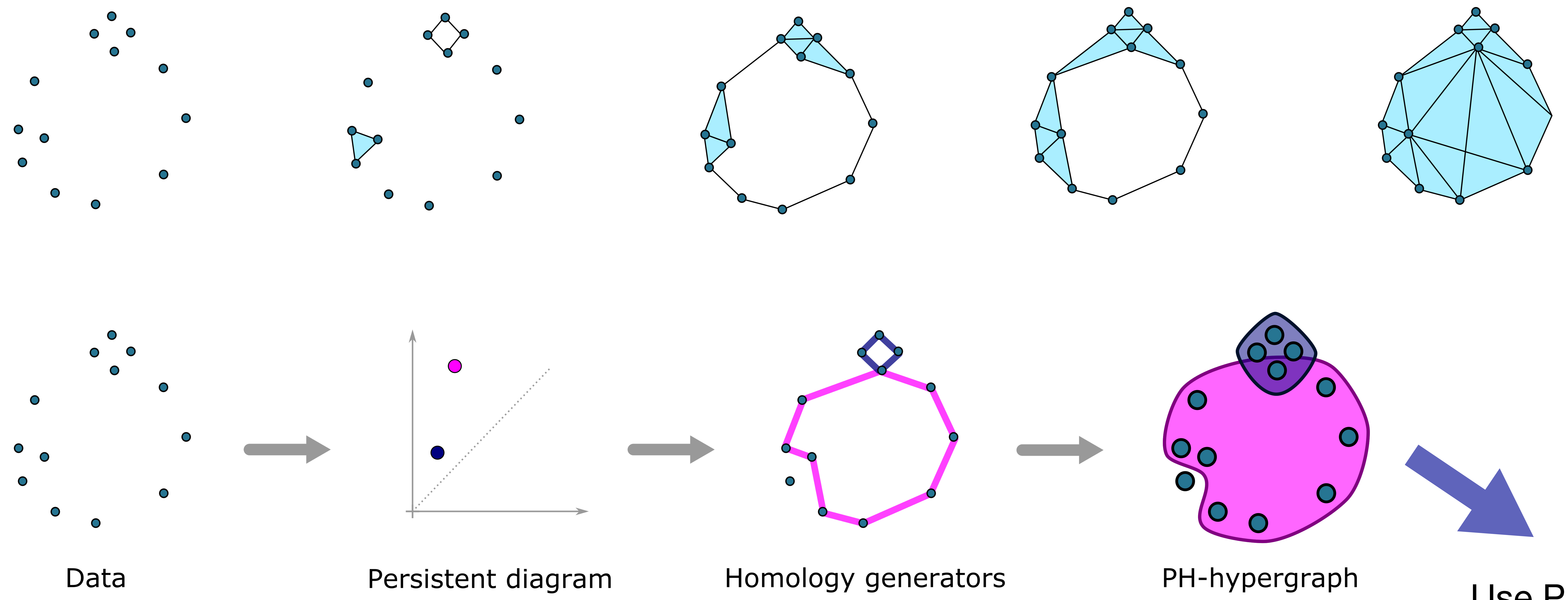


Recently generalised to **hypergraphs**: **hyperCOT** outputs **coupled** matchings of **vertices** and **edges**

Chowdhury, Samir, et al. "**Hypergraph co-optimal transport**: Metric and categorical properties." *arXiv preprint arXiv:2112.03904* (2021).

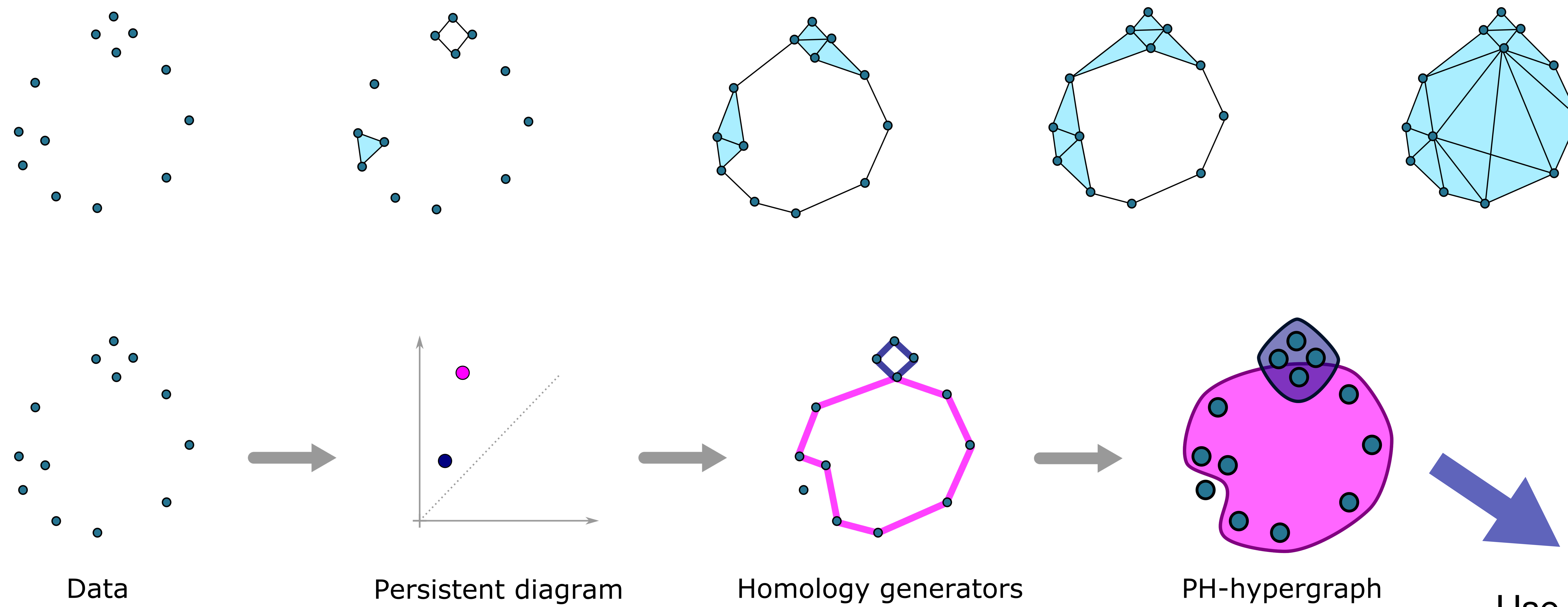
Figures taken from Vayer, Titouan, et al. *arXiv preprint arXiv:1811.02834* (2018) and Peyré, Gabriel, and Marco Cuturi. *Center for Research in Economics and Statistics Working Papers* 2017-86 (2017).

Persistent homology and the PH-hypergraph



Use PH-hypergraph for **Topological Optimal Transport** theory

Persistent homology and the PH-hypergraph

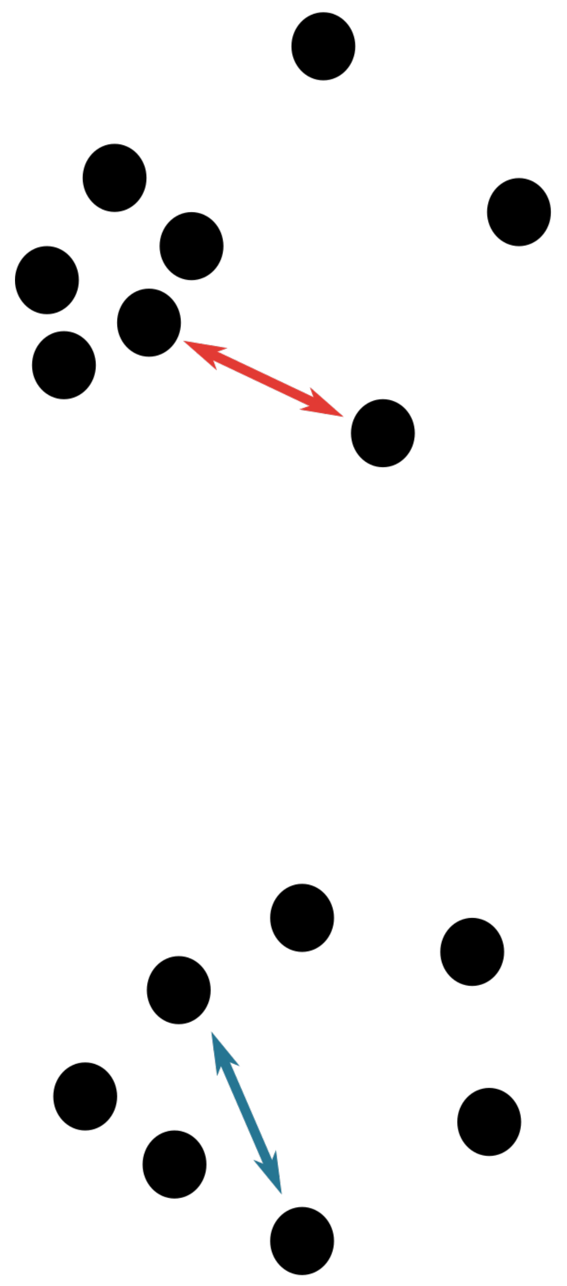



T.Needham


S.Zhang

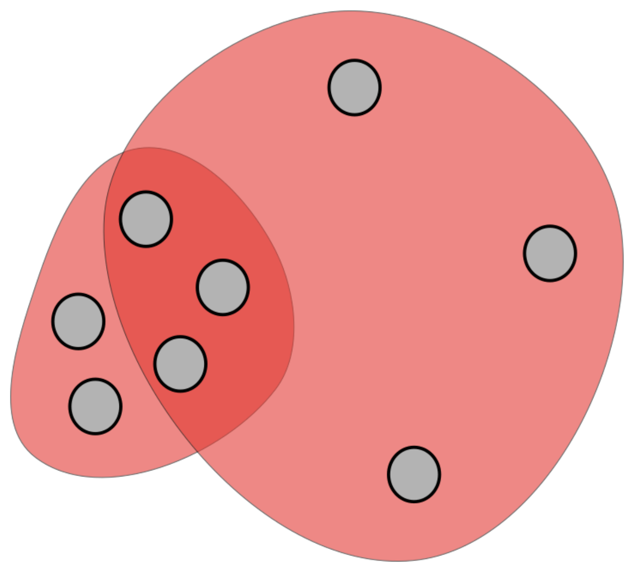
Use PH-hypergraph for **Topological Optimal Transport** theory

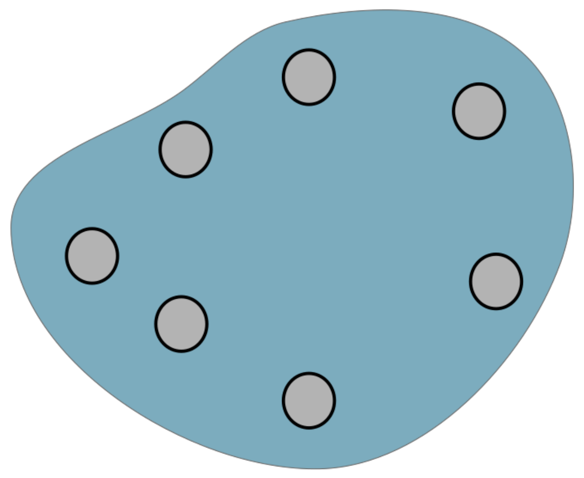
Topological Optimal Transport (tPOT)



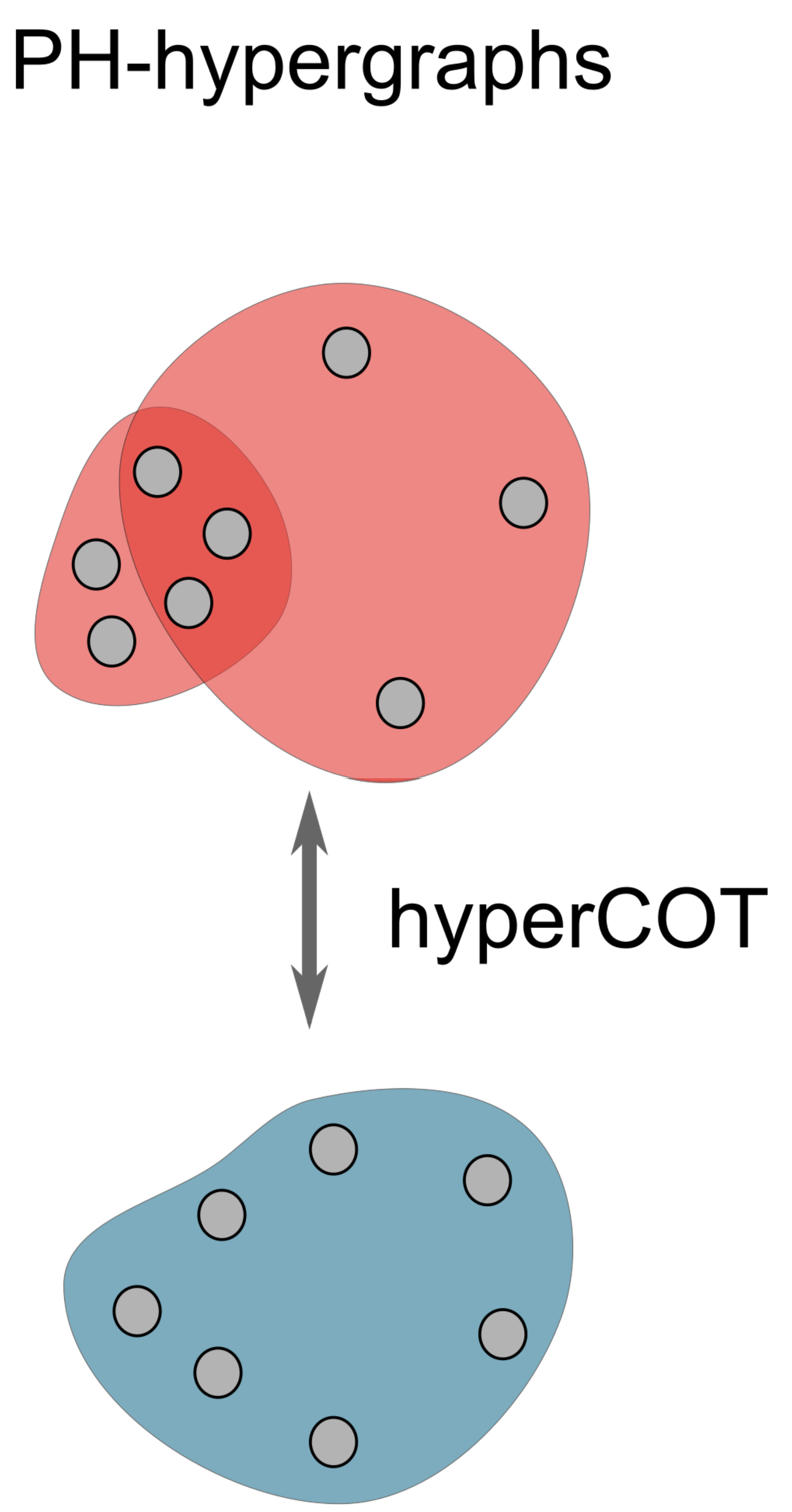
Topological Optimal Transport (tPOT)

PH-hypergraphs





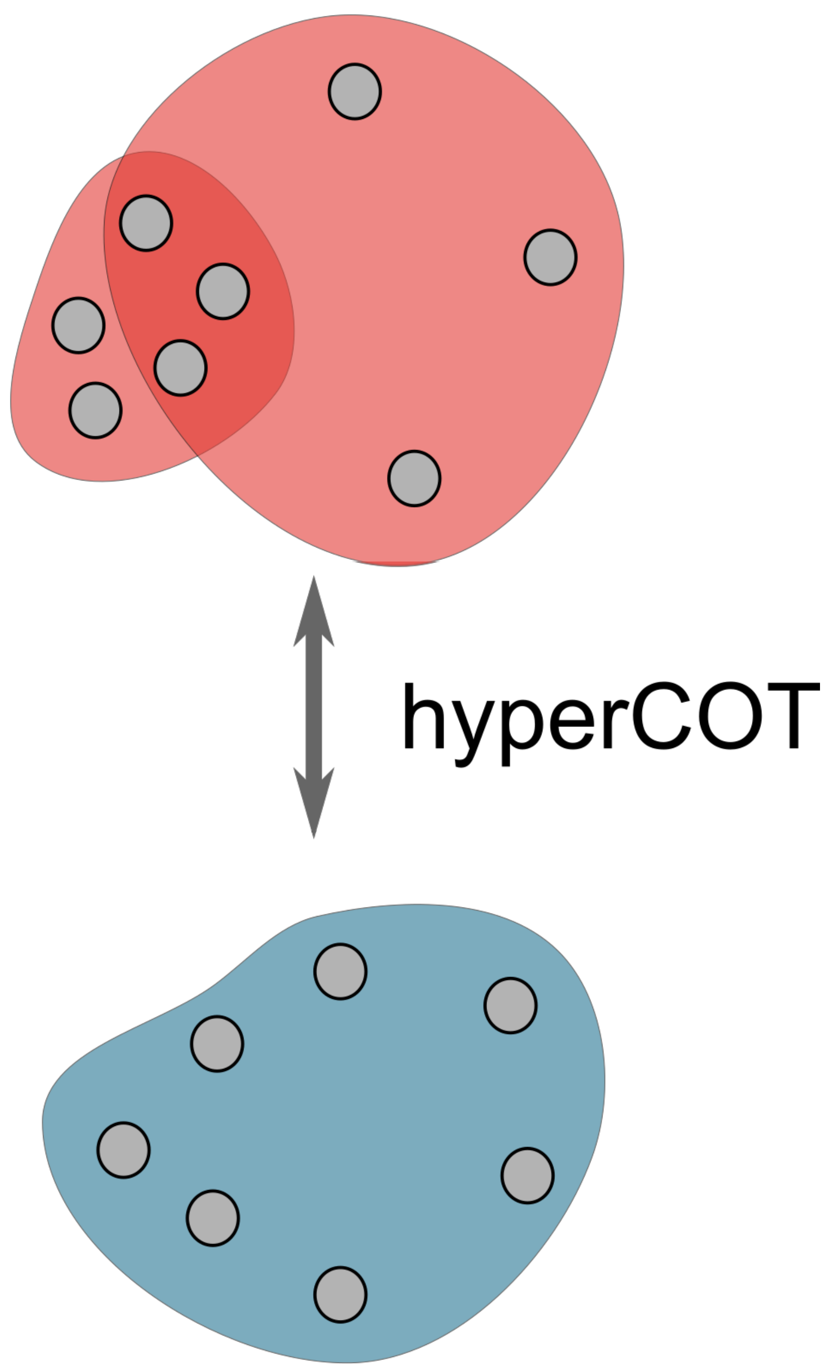
Topological Optimal Transport (tPOT)



Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

Topological Optimal Transport (tPOT)

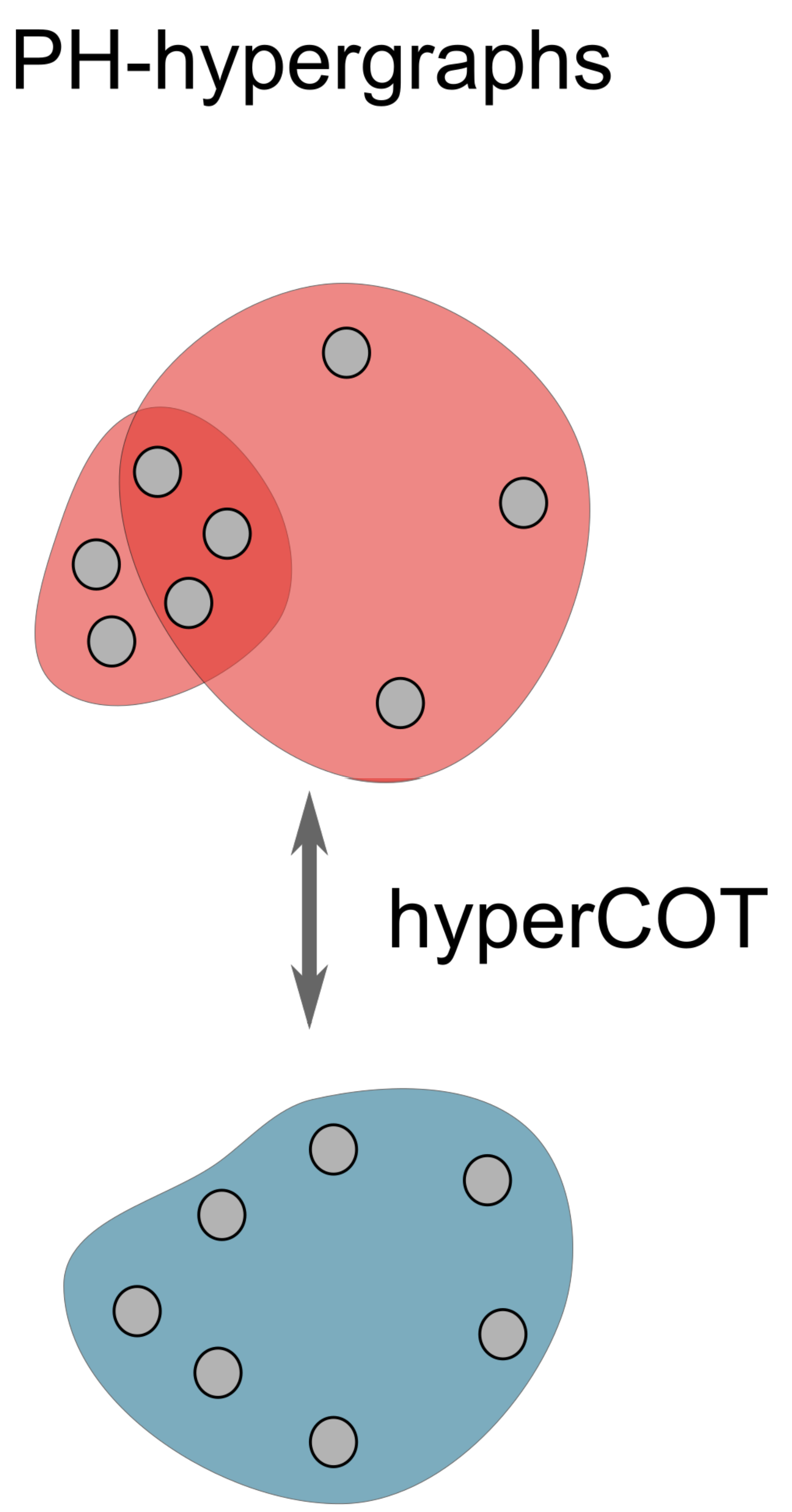
PH-hypergraphs



Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

Problem 1: how to accurately *match* edges (= features)?

Topological Optimal Transport (tPOT)

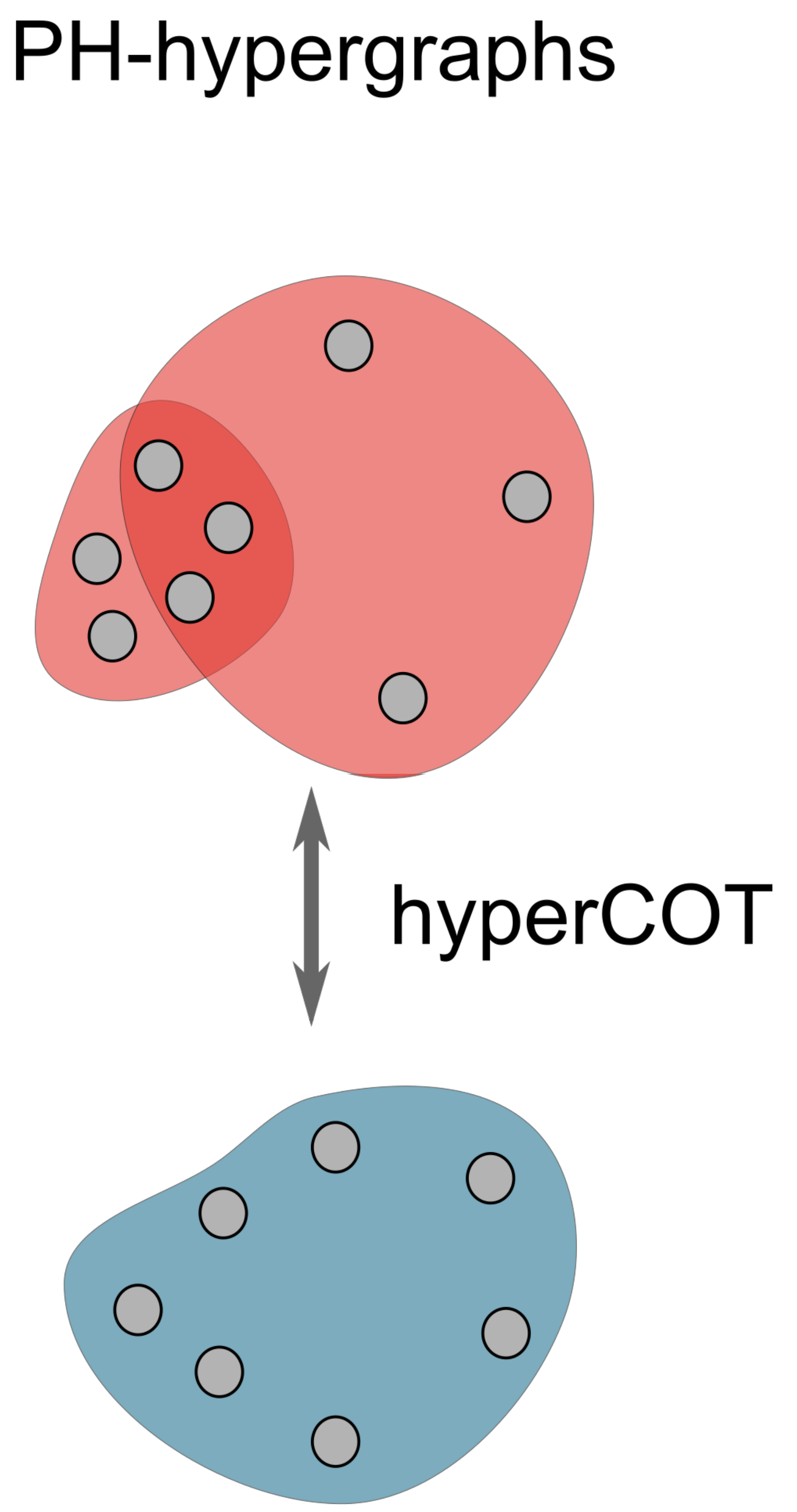


Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

Problem 1: how to accurately *match* edges (= features)?

Note: weighting by persistence does not work!!

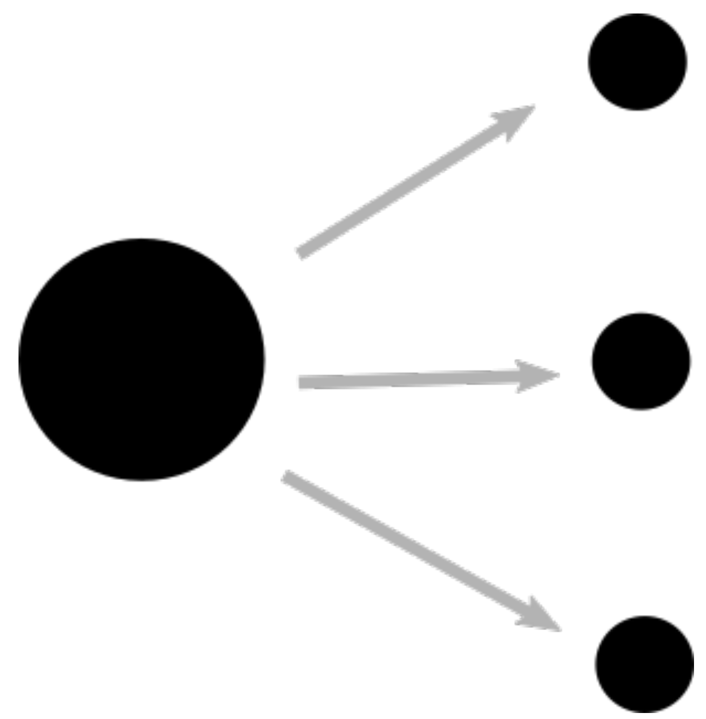
Topological Optimal Transport (tPOT)



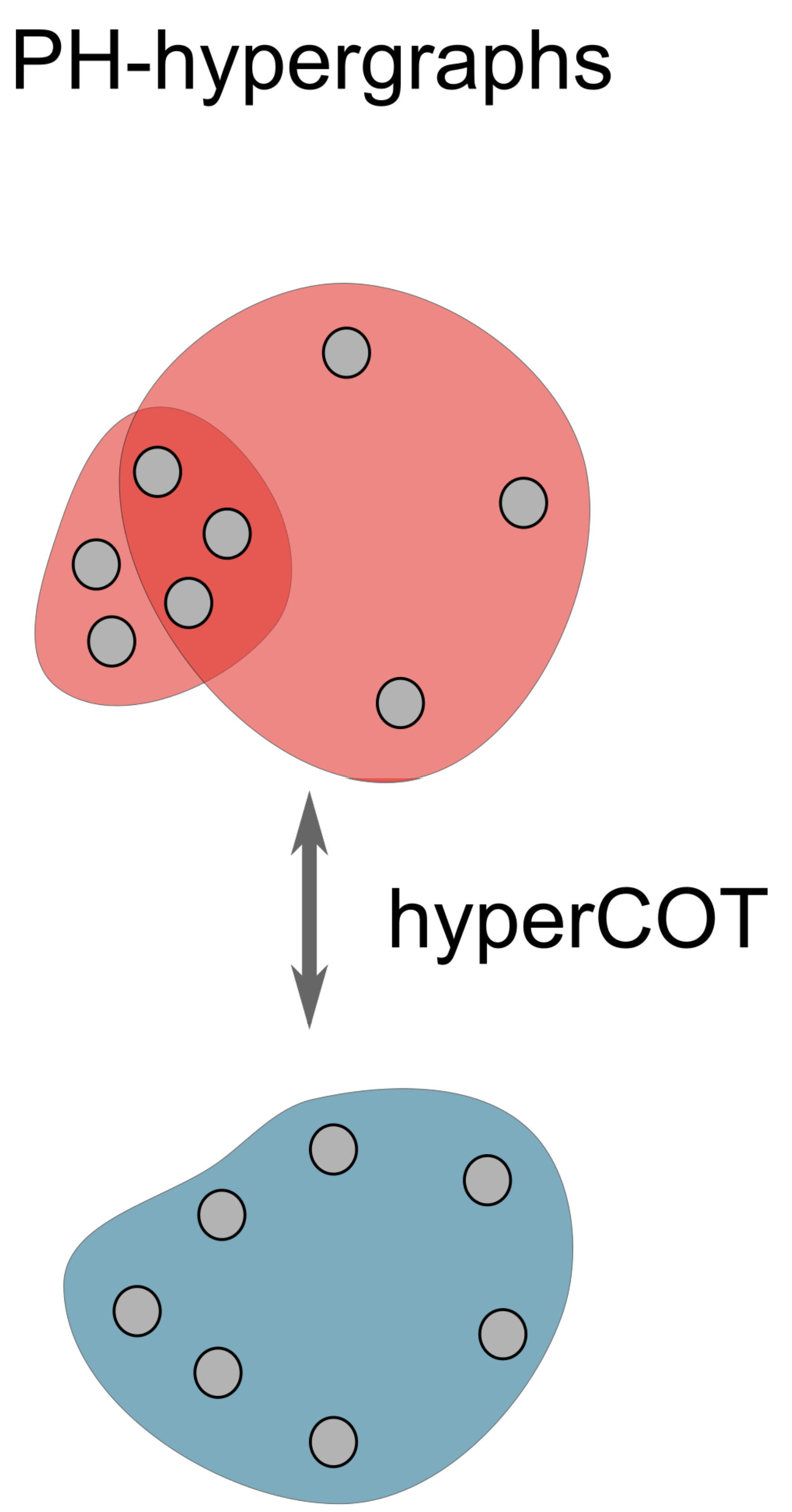
Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

Problem 1: how to accurately *match* edges (= features)?

Note: weighting by persistence does not work!!



Topological Optimal Transport (tPOT)



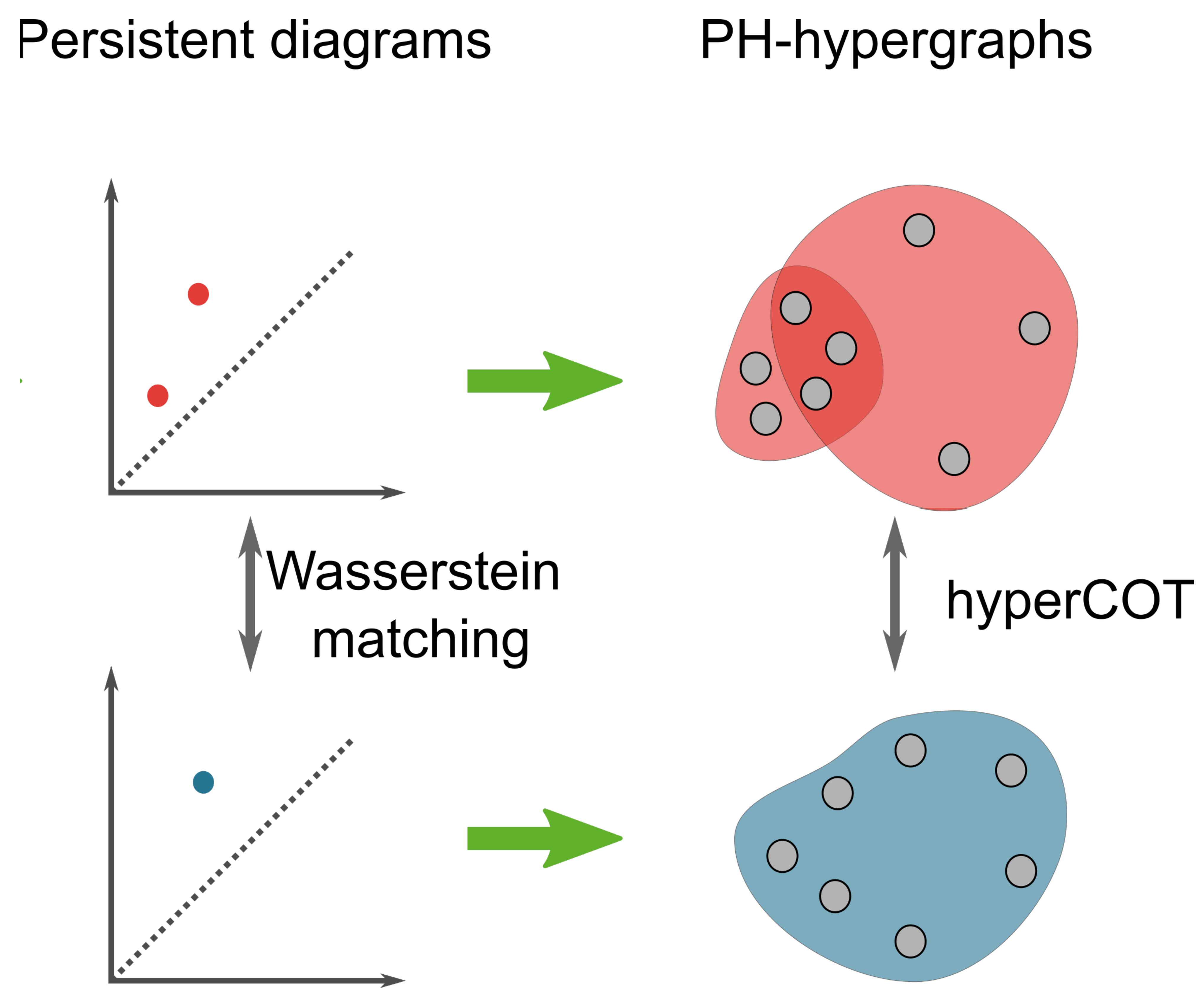
Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

Problem 1: how to accurately *match* edges (= features)?

Note: weighting by persistence does not work!!

Problem 2: what about points not involved in any homology cycle?

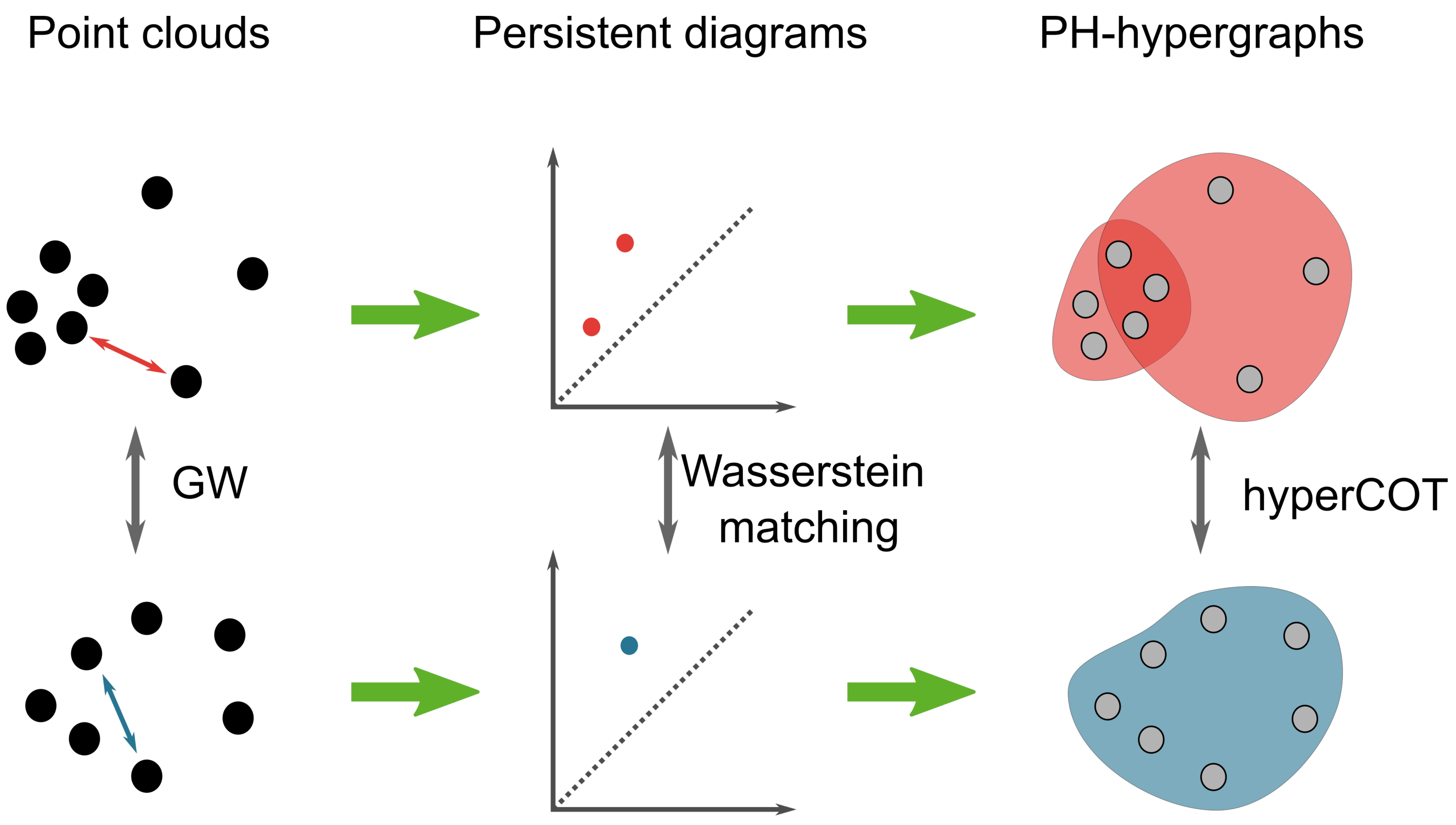
Topological Optimal Transport (tPOT)



Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

Solution 1: couple with Wasserstein matching on PDs!!

Topological Optimal Transport (tPOT)

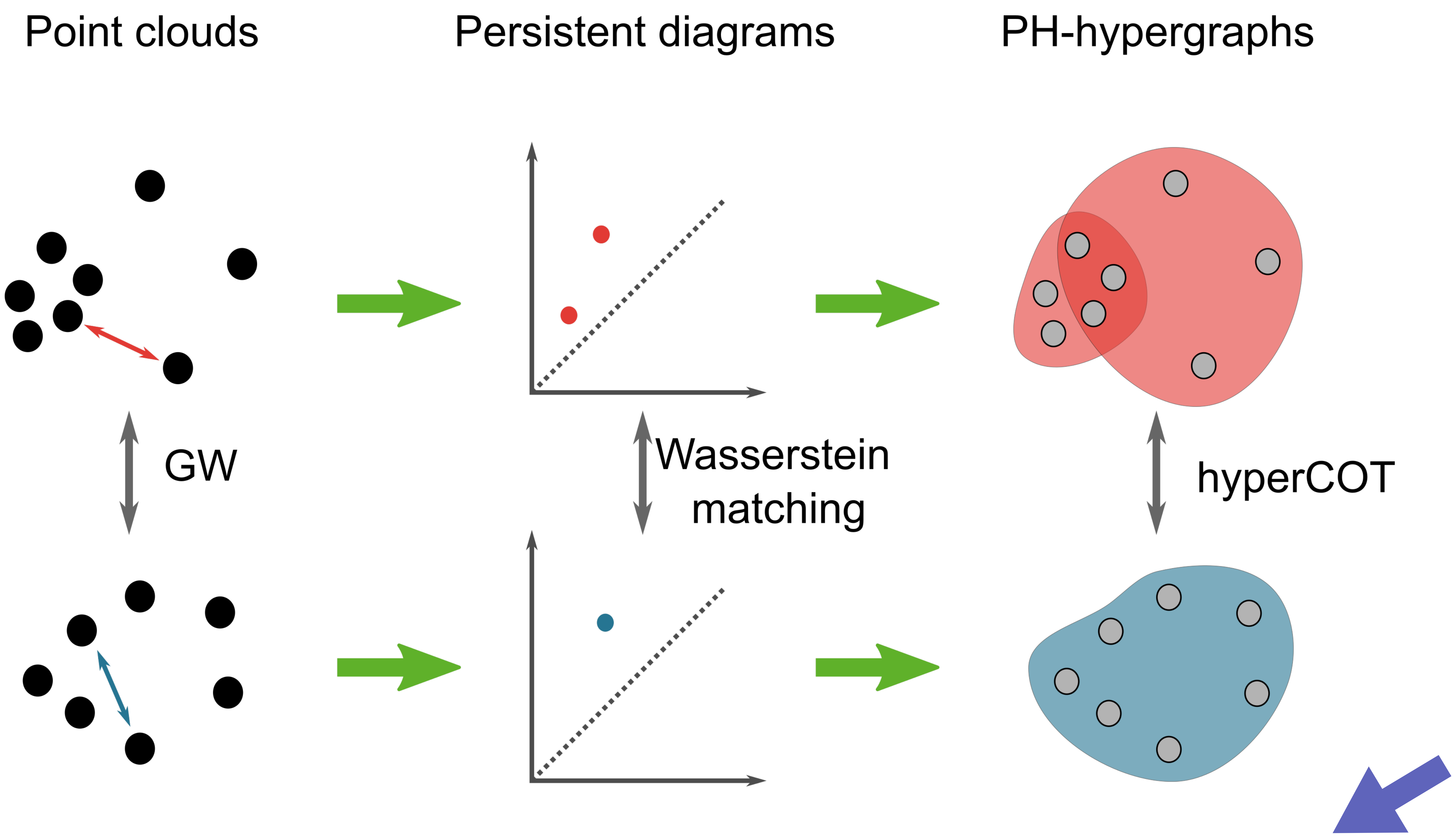


Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

Solution 1: couple with Wasserstein matching on PDs!!

Solution 2: Couple with GW on point clouds!

Topological Optimal Transport (tPOT)



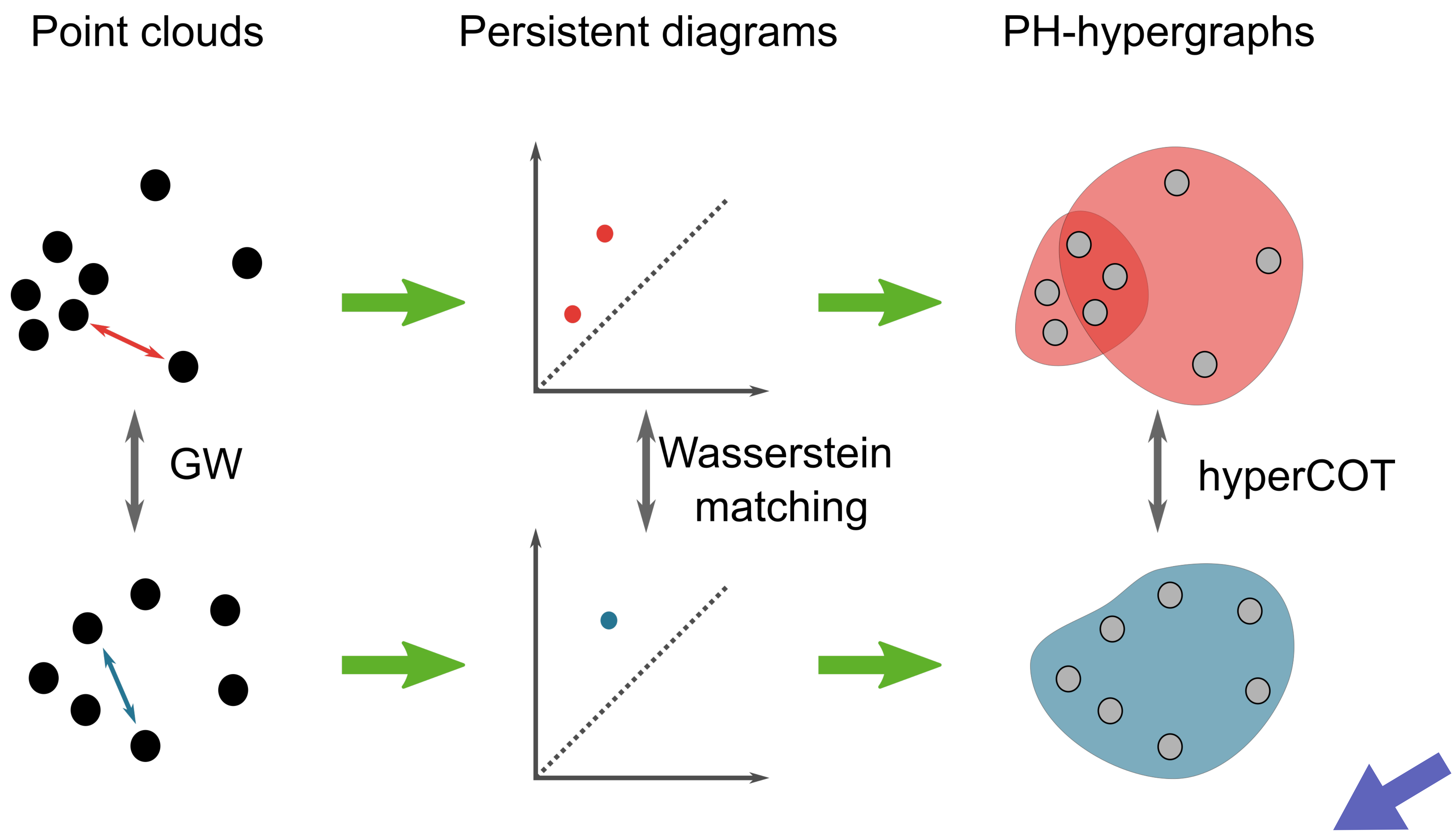
Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

Solution 1: couple with Wasserstein matching on PDs!!

Solution 2: Couple with GW on point clouds!

$$\min_{\pi^v,\pi^e} \lambda\langle \tilde{L}(X,X'), \pi^v \otimes \pi^e\rangle + (\alpha\langle L(C,C'), \pi^v \otimes \pi^v\rangle + (1-\alpha)\langle \tilde{C}[D,D'], \pi^e\rangle)$$

Topological Optimal Transport (tPOT)



Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

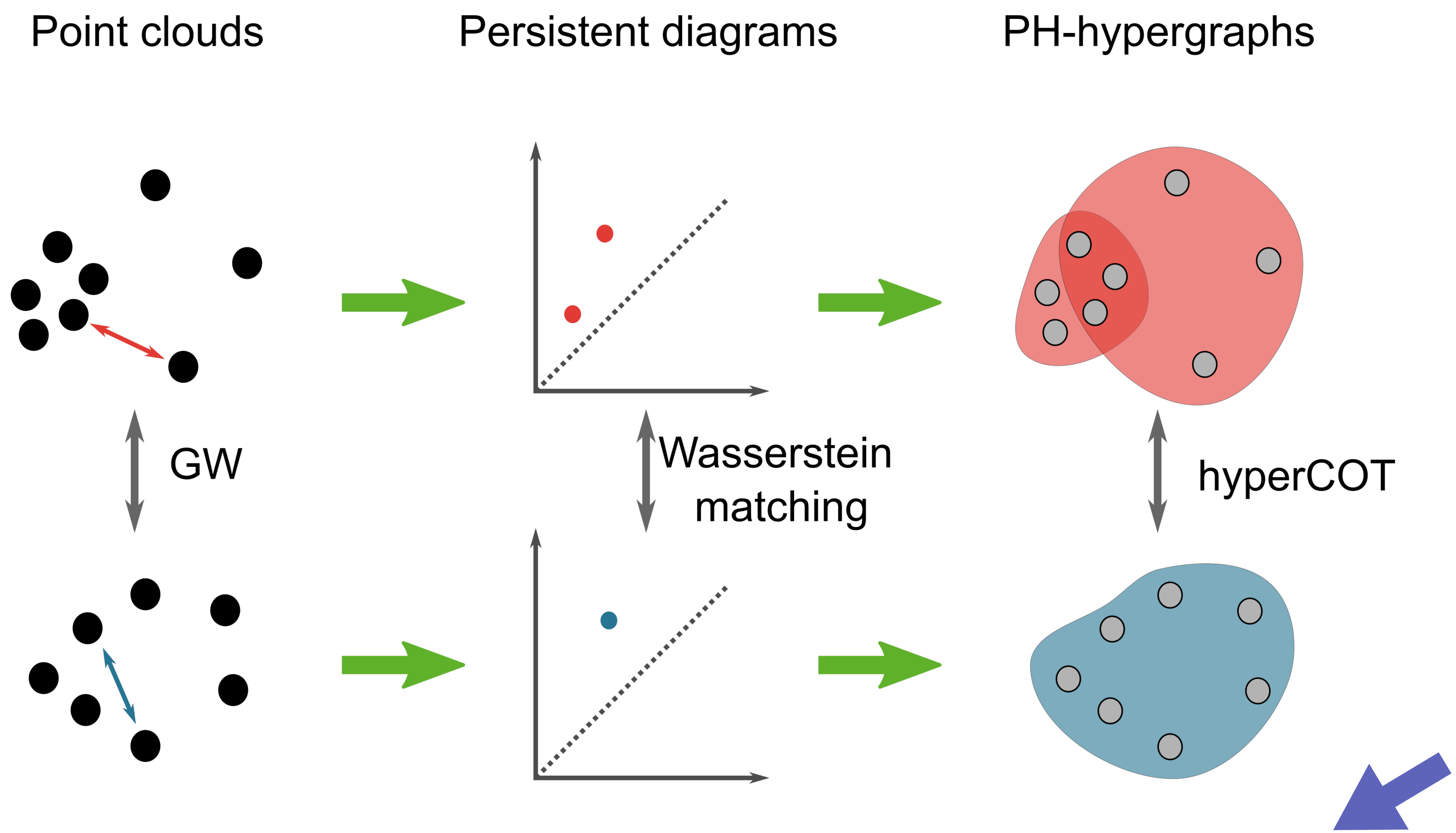
Solution 1: couple with Wasserstein matching on PDs!!

Solution 2: Couple with GW on point clouds!

$$\min_{\pi^v,\pi^e} \lambda\langle \tilde{L}(X,X'), \pi^v\otimes\pi^e\rangle + (\alpha\langle L(C,C'),\pi^v\otimes\pi^v\rangle + (1-\alpha)\langle \tilde{C}[D,D'],\pi^e\rangle)$$

hyperCOT on PH-hypergraphs

Topological Optimal Transport (tPOT)



Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

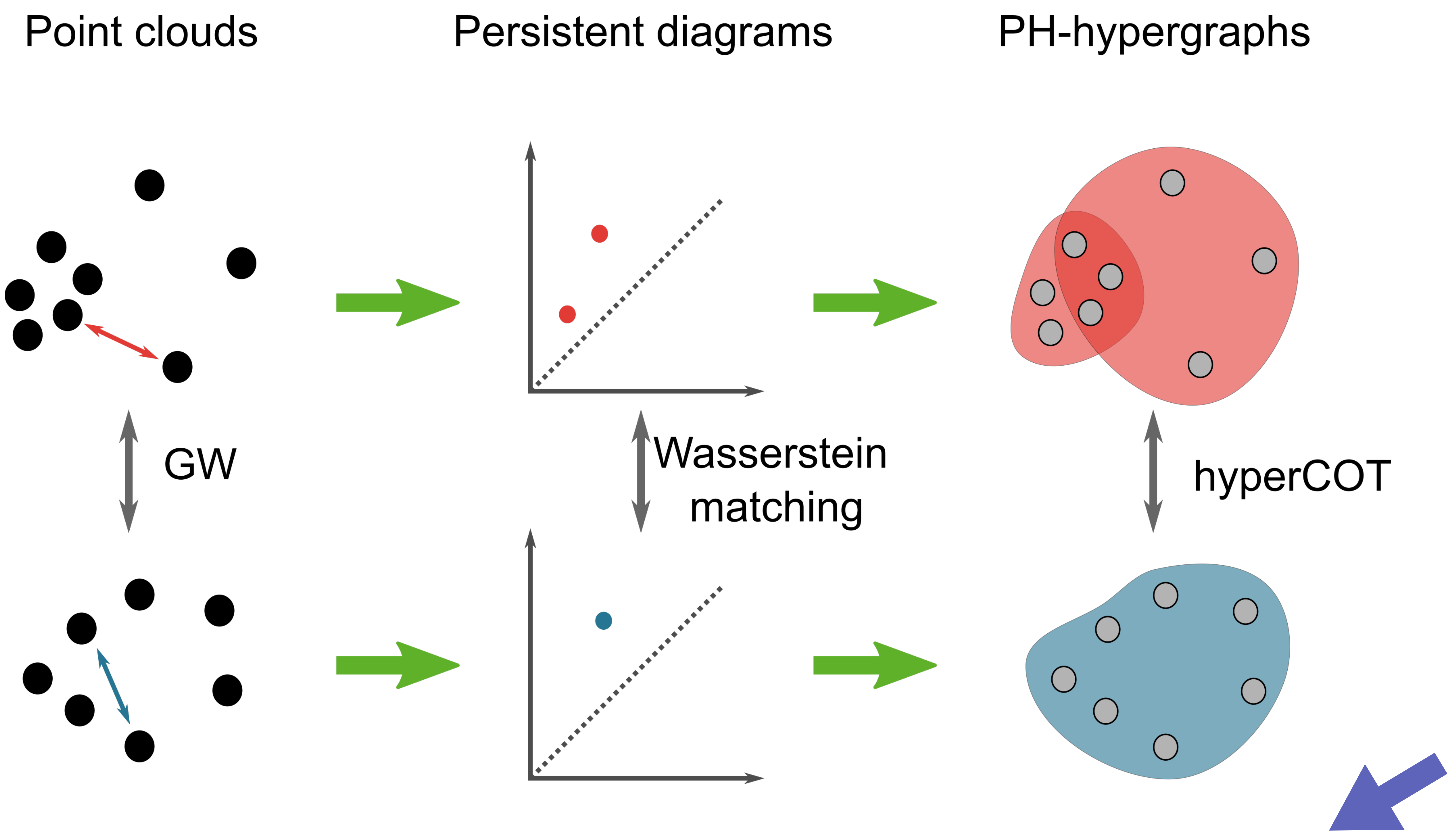
Solution 1: couple with Wasserstein matching on PDs!!

Solution 2: Couple with GW on point clouds!

$$\min_{\pi^v,\pi^e} \lambda\langle \tilde{L}(X,X'), \pi^v \otimes \pi^e\rangle + (\alpha\langle L(C,C'), \pi^v \otimes \pi^v\rangle + (1-\alpha)\langle \tilde{C}[D,D'], \pi^e\rangle)$$

hyperCOT on PH-hypergraphs **GW on point clouds**

Topological Optimal Transport (tPOT)



Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

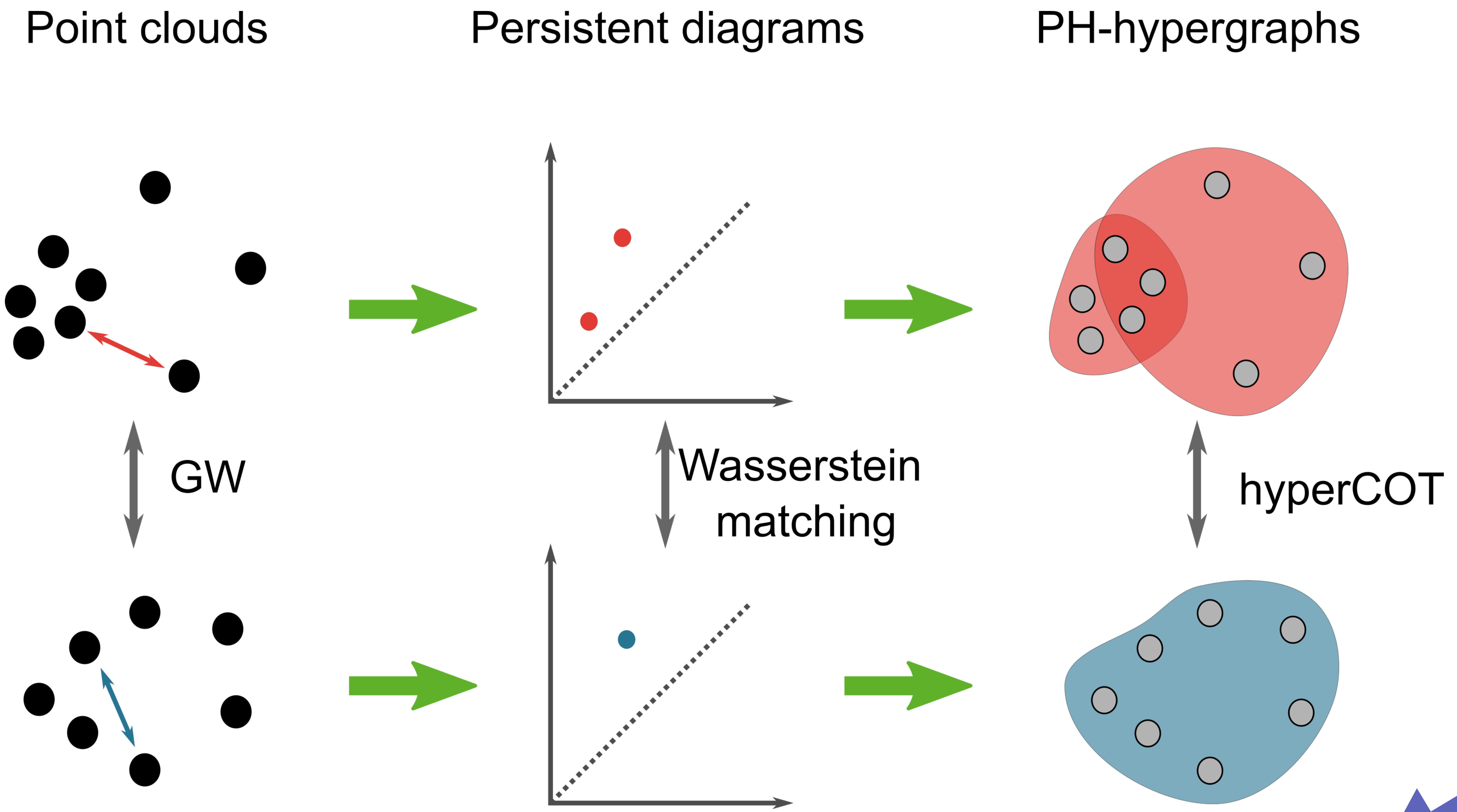
Solution 1: couple with Wasserstein matching on PDs!!

Solution 2: Couple with GW on point clouds!

$$\min_{\pi^v,\pi^e} \lambda\langle \tilde{L}(X,X'), \pi^v \otimes \pi^e\rangle + (\alpha\langle L(C,C'), \pi^v \otimes \pi^v\rangle + (1-\alpha)\langle \tilde{C}[D,D'], \pi^e\rangle)$$

hyperCOT on PH-hypergraphs | **GW on point clouds** | **W metric on PDs**

Topological Optimal Transport (tPOT)



Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

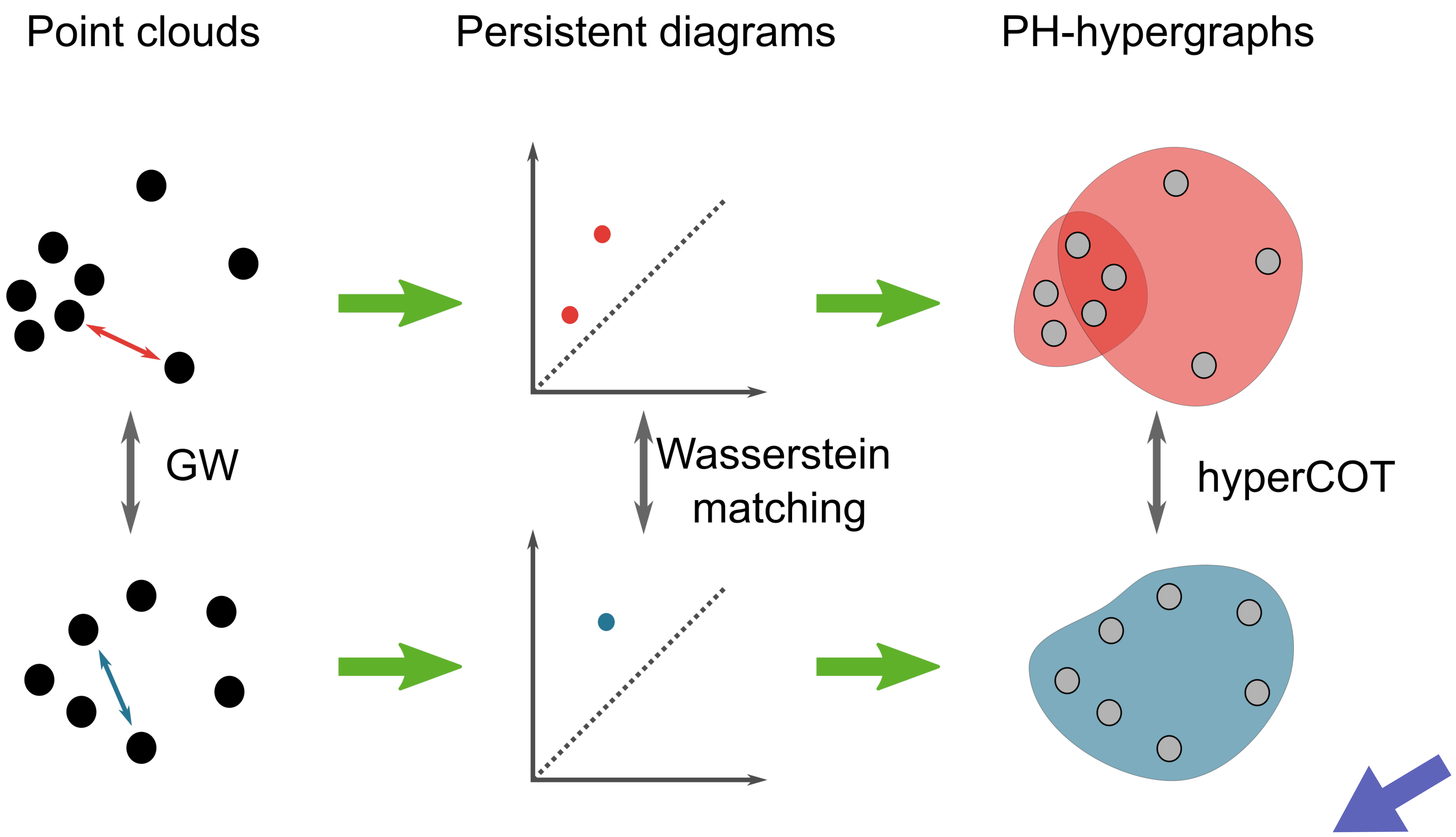
Solution 1: couple with Wasserstein matching on PDs!!

Solution 2: Couple with GW on point clouds!

$$\min_{\pi^v,\pi^e} \lambda\langle \tilde{L}(X,X'), \pi^v\otimes\pi^e\rangle + (\alpha\langle L(C,C'),\pi^v\otimes\pi^v\rangle + (1-\alpha)\langle \tilde{C}[D,D'],\pi^e\rangle)$$

The parameter **α** interpolates between GW and W on PDs

Topological Optimal Transport (tPOT)



Idea: use transport theories (*e.g.* hyperCOT) on PH-hgs to transport point clouds based on topological features

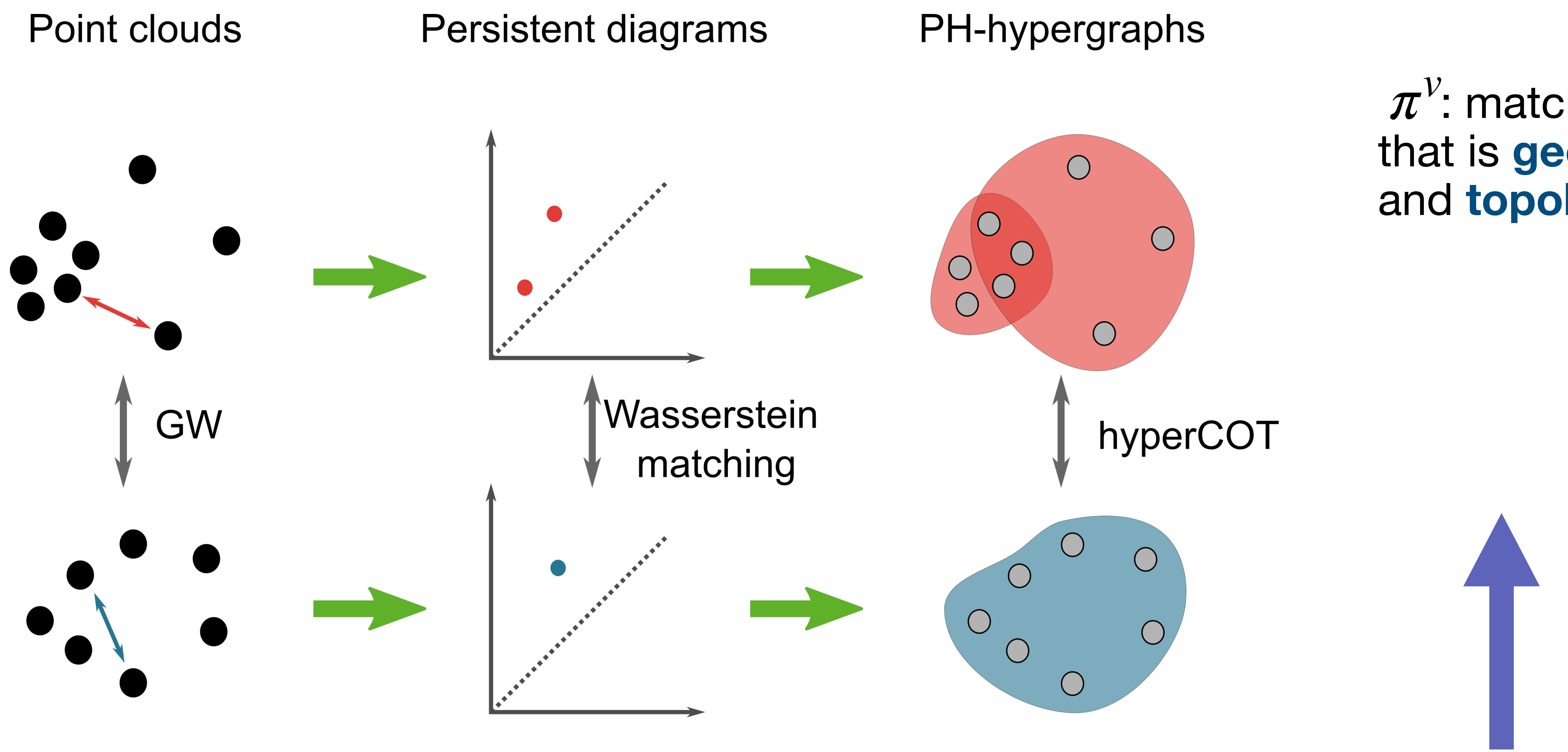
Solution 1: couple with Wasserstein matching on PDs!!

Solution 2: Couple with GW on point clouds!

$$\min_{\pi^v,\pi^e} \lambda\langle \tilde{L}(X,X'), \pi^v \otimes \pi^e\rangle + (\alpha\langle L(C,C'), \pi^v \otimes \pi^v\rangle + (1-\alpha)\langle \tilde{C}[D,D'], \pi^e\rangle)$$

Coupling with hyperCOT

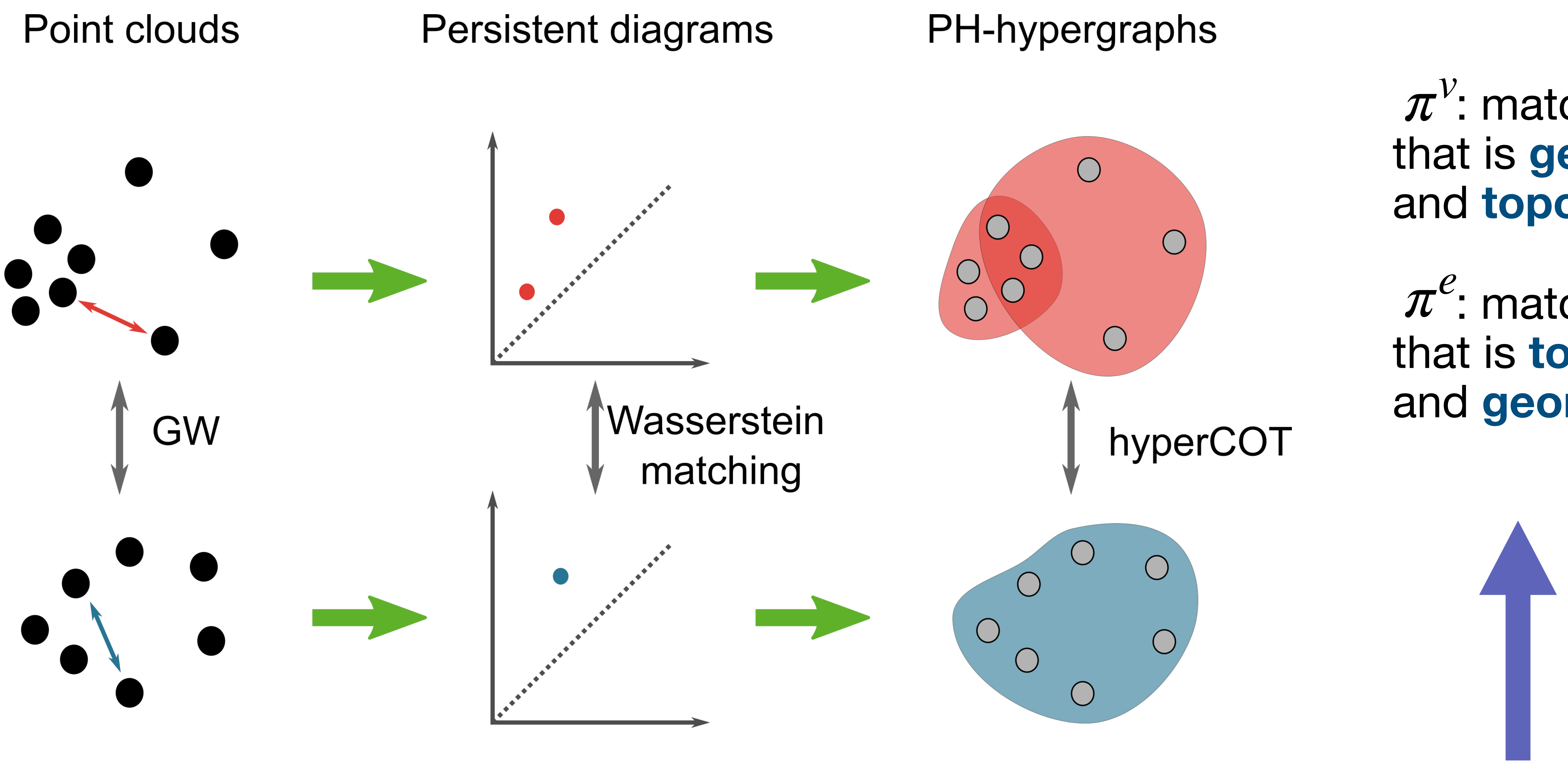
Topological Optimal Transport (tPOT)



π^v: matching between points that is **geometrically driven** and **topologically informed**

$$\min_{\pi^v,\pi^e} \lambda\langle \tilde{L}(X,X'), \pi^v \otimes \pi^e\rangle + (\alpha\langle L(C,C'), \pi^v \otimes \pi^v\rangle + (1-\alpha)\langle \tilde{C}[D,D'], \pi^e\rangle)$$

Topological Optimal Transport (tPOT)

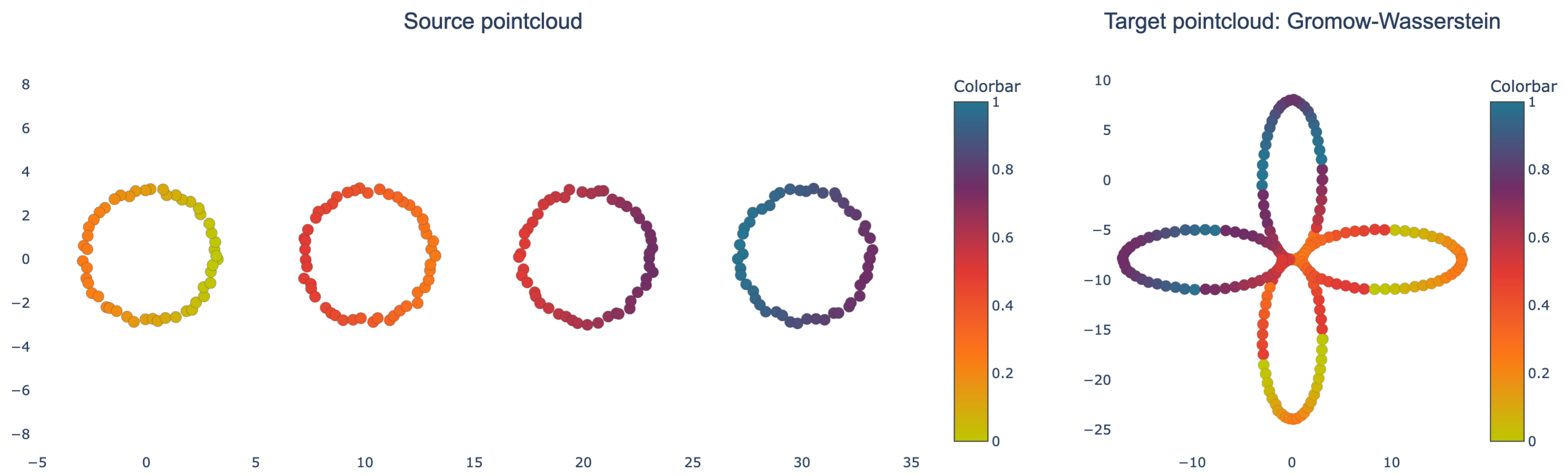


π^v: matching between points that is **geometrically driven** and **topologically informed**

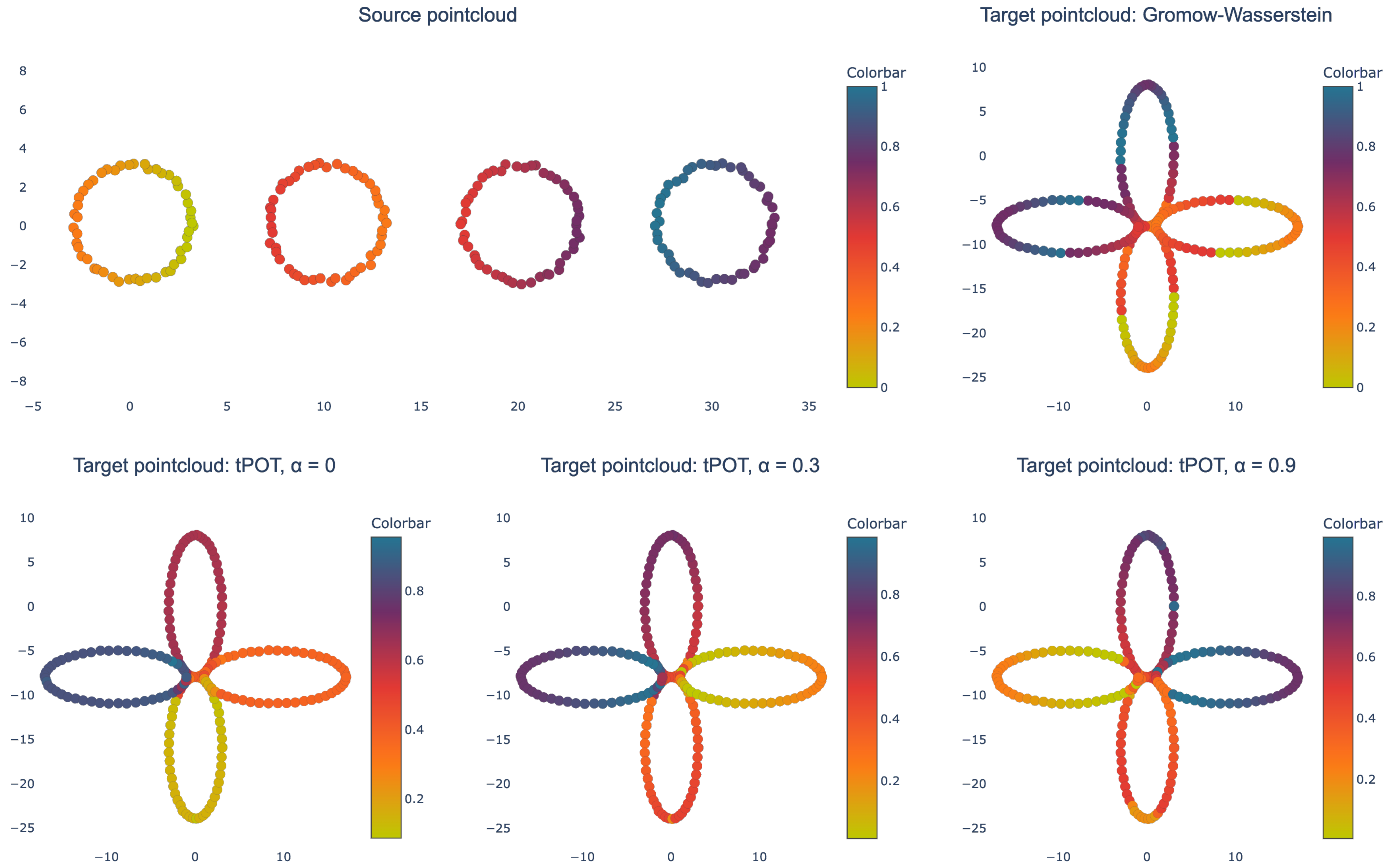
π^e: matching between edges that is **topologically driven** and **geometrically informed**

$$\min_{\pi^v,\pi^e} \lambda\langle \tilde{L}(X, X'), \pi^v \otimes \pi^e\rangle + (\alpha\langle L(C, C'), \pi^v \otimes \pi^v\rangle + (1-\alpha)\langle \tilde{C}[D, D'], \pi^e\rangle)$$

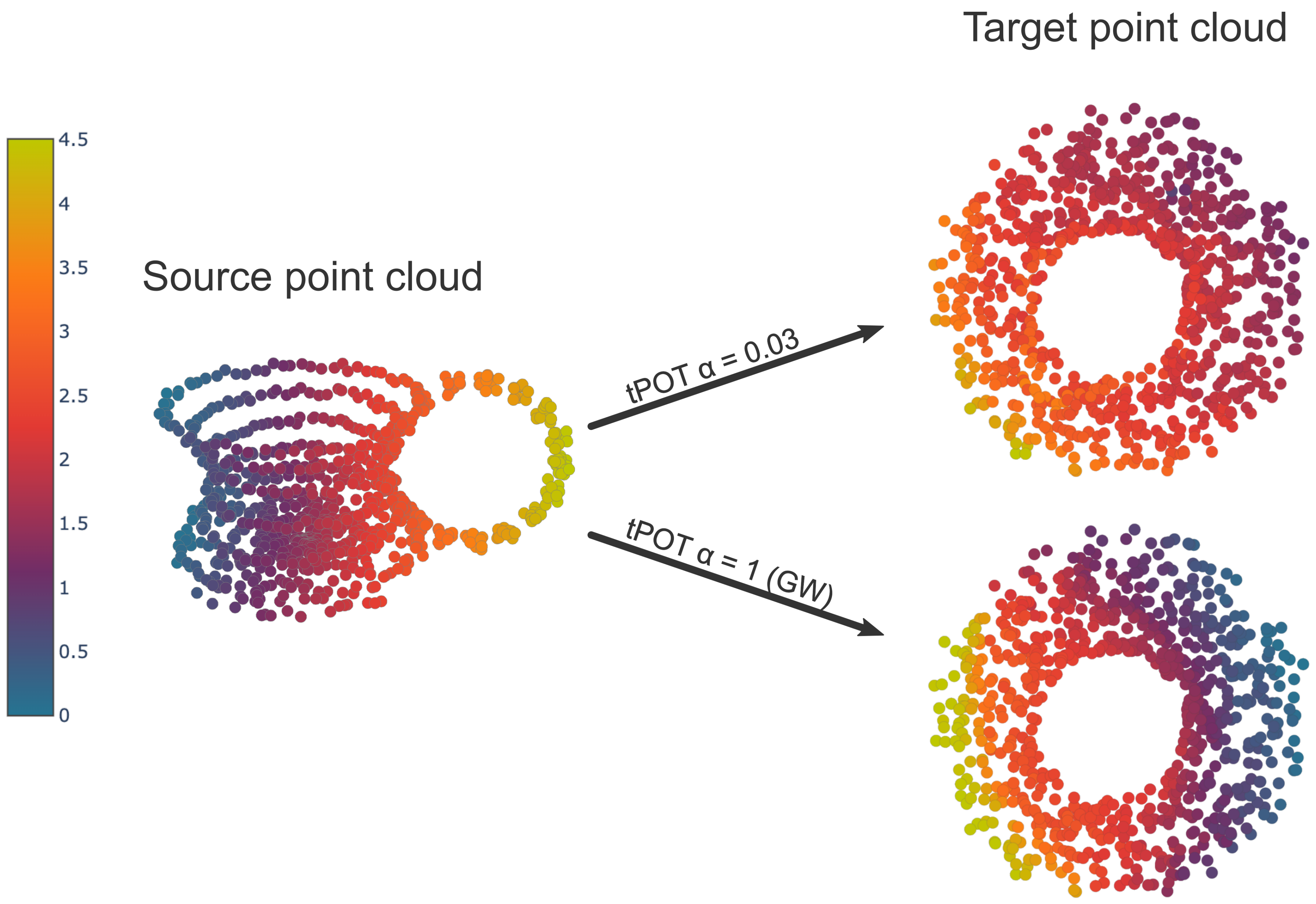
Topological Optimal Transport (tPOT): examples

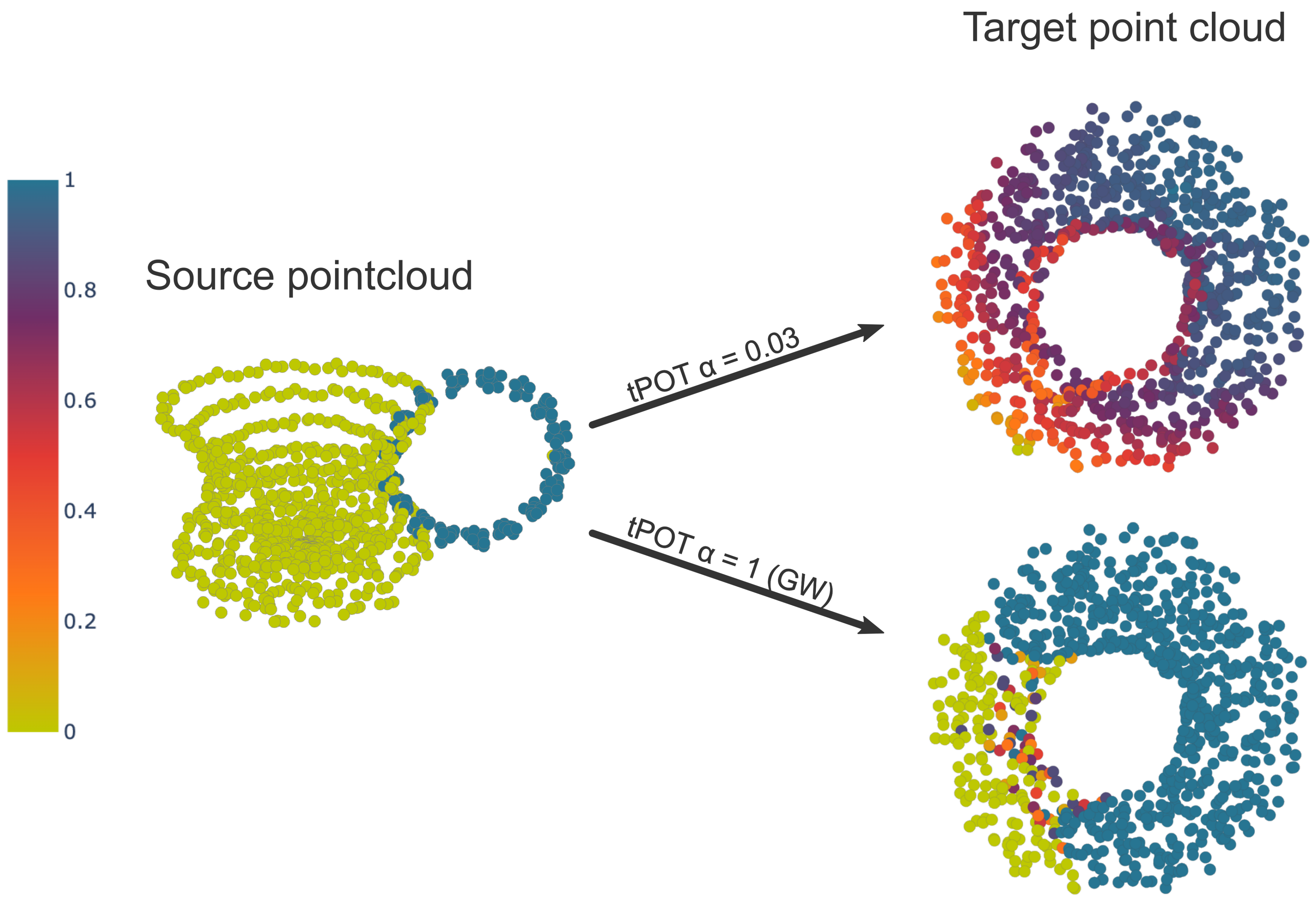


Topological Optimal Transport (tPOT): examples

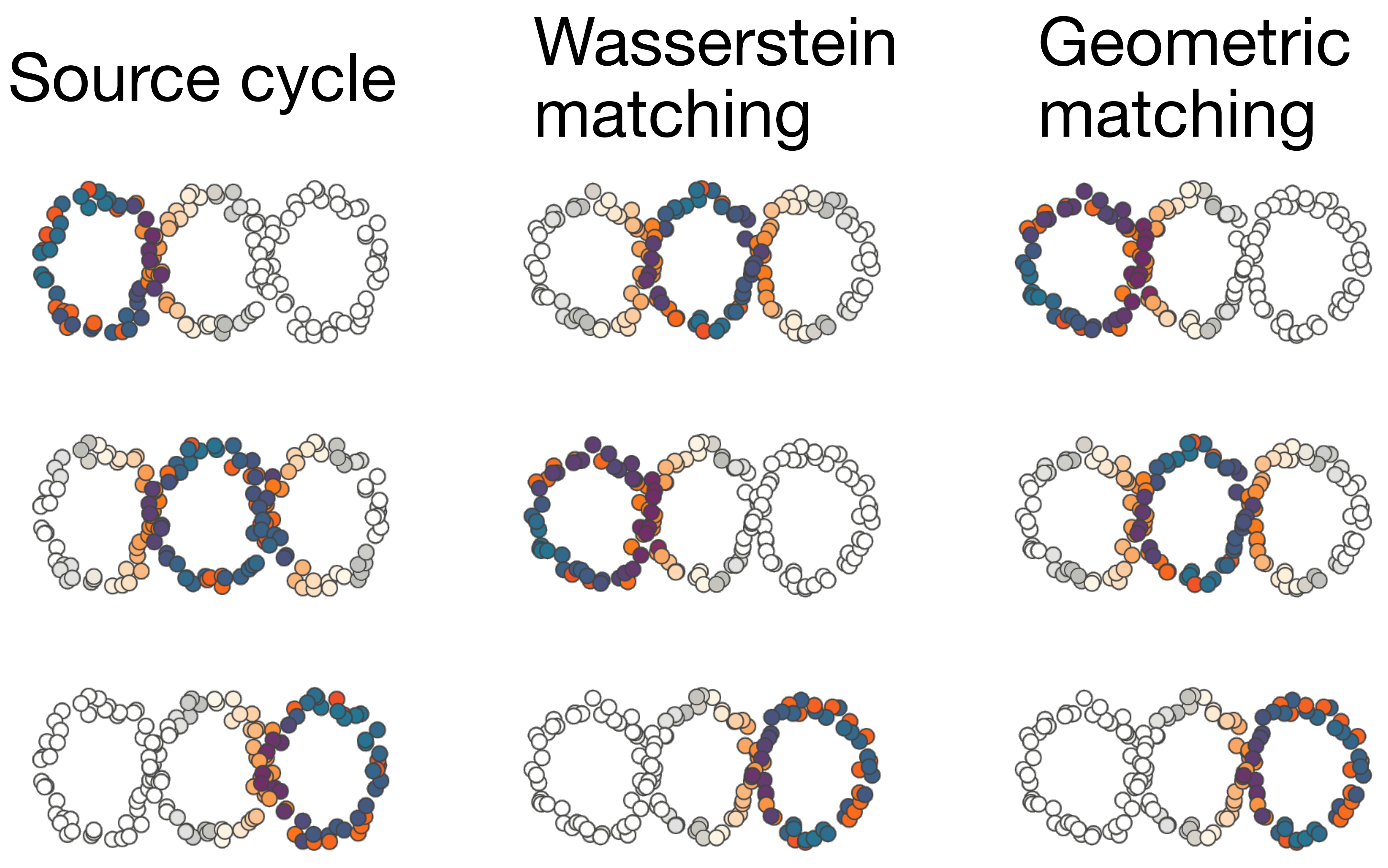


Topological Optimal Transport (tPOT): examples

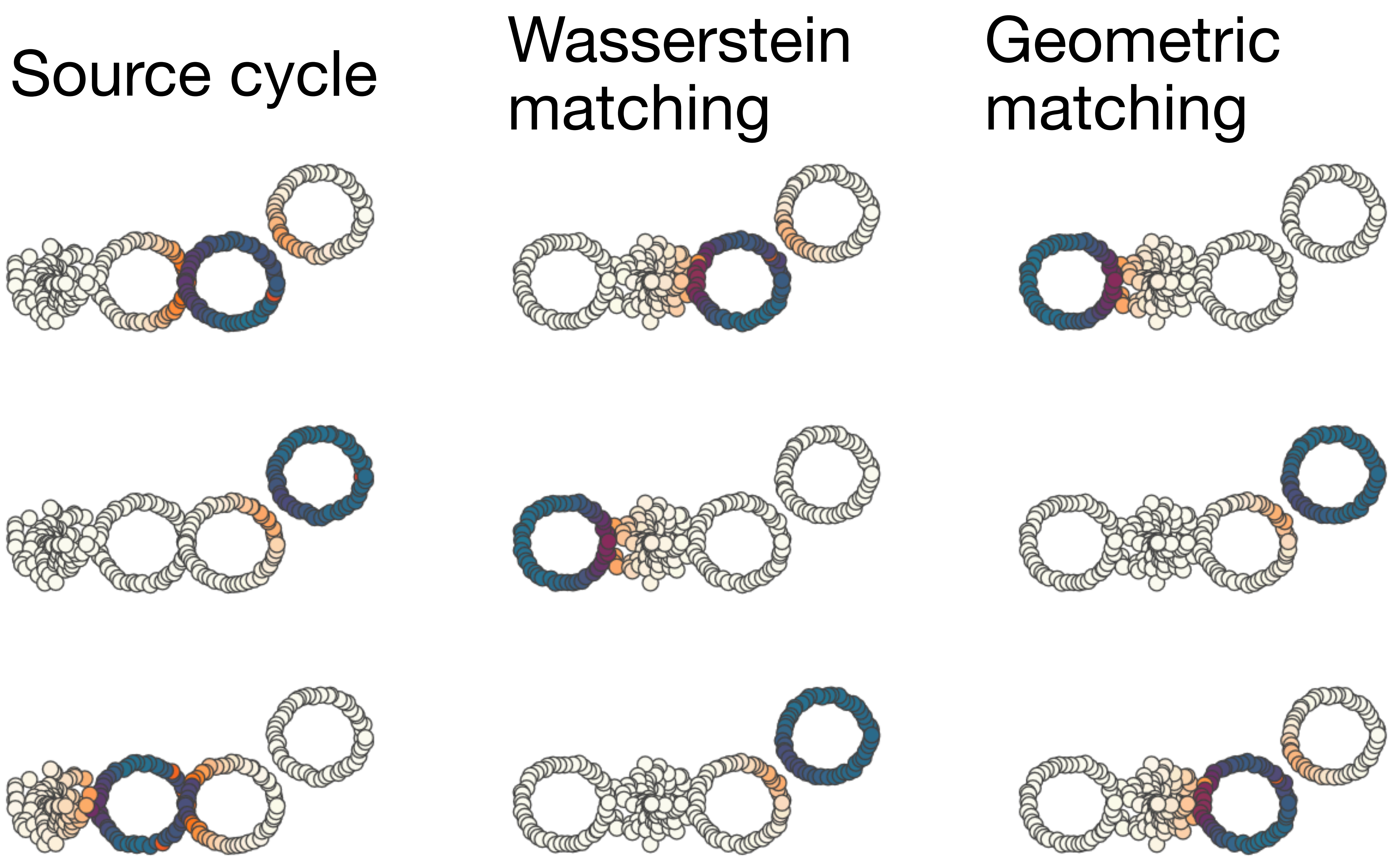




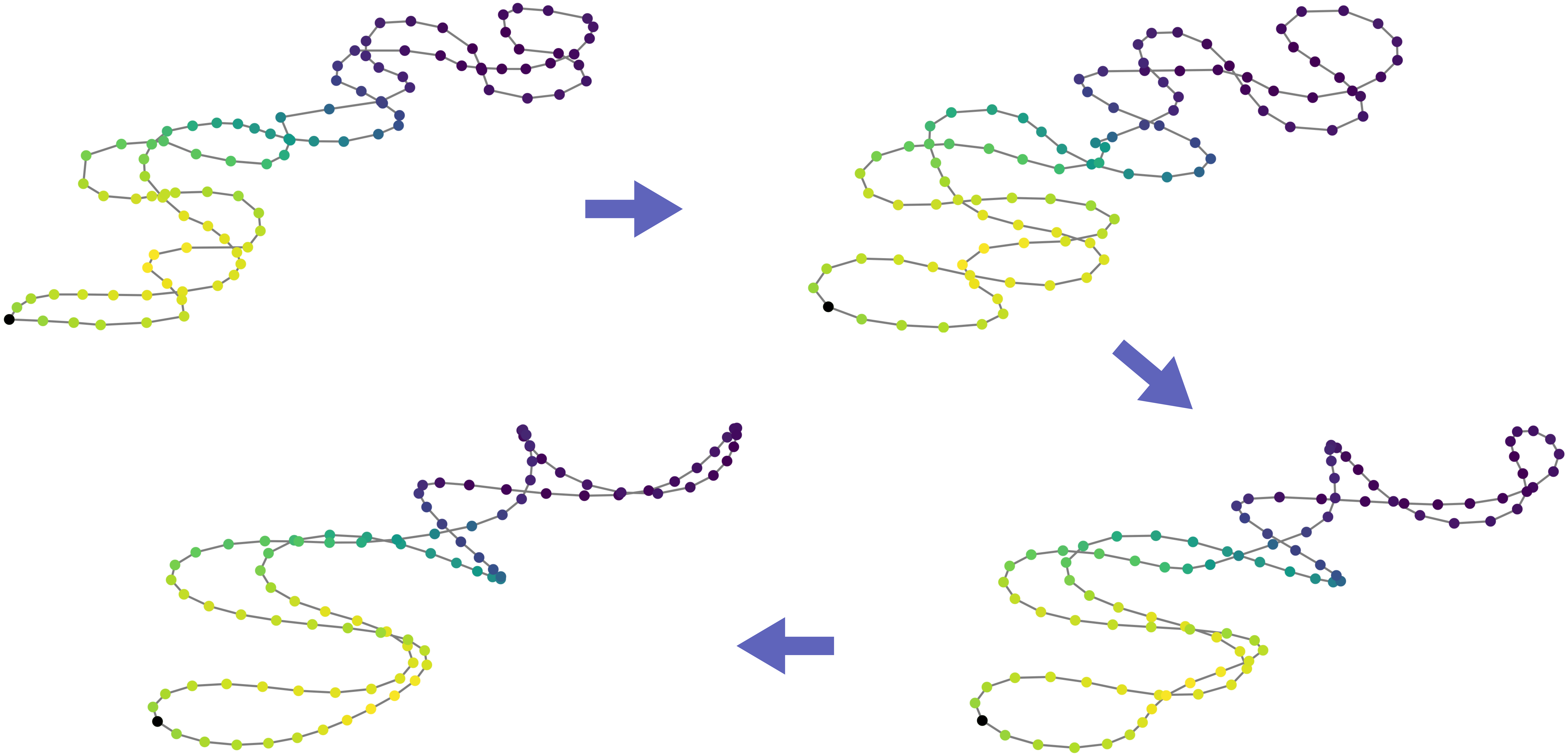
Geometric cycle matching



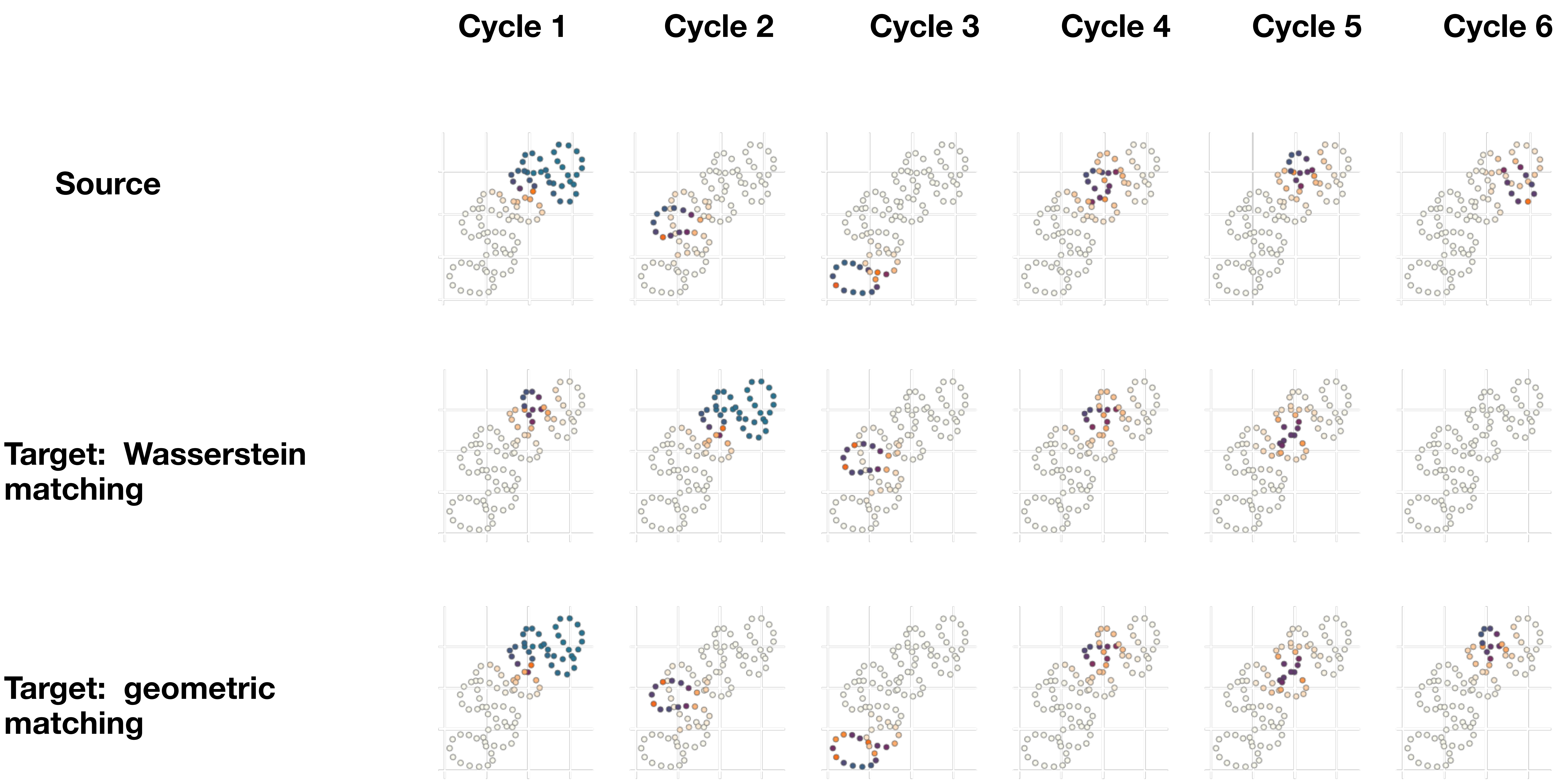
The **GW** component helps matching when classes are **topologically indistinguishable**



Geometric cycle matching



Geometric cycle matching



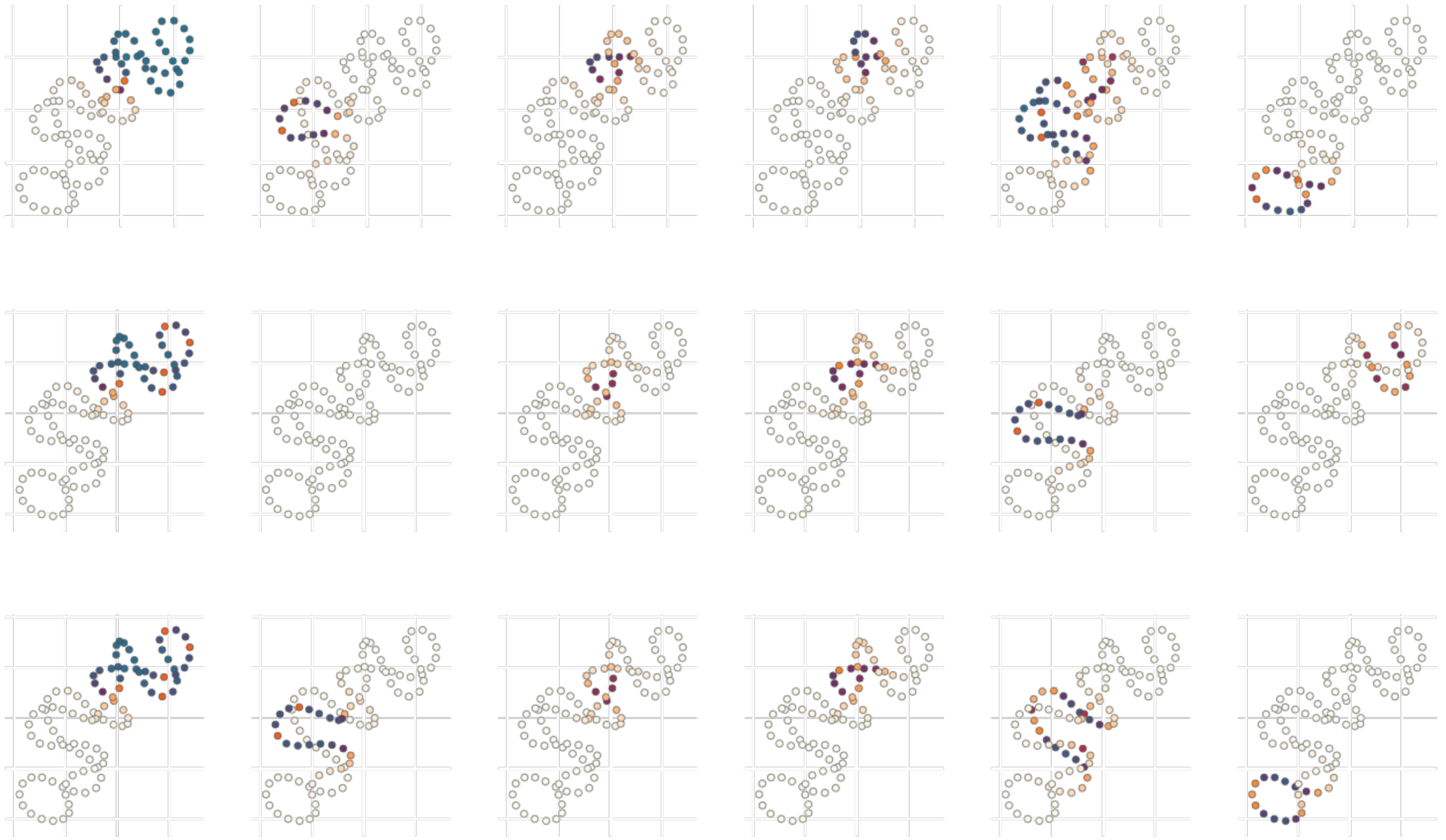
Geometric cycle matching

Cycle 1 | Cycle 2 | Cycle 3 | Cycle 4 | Cycle 5 | Cycle 6

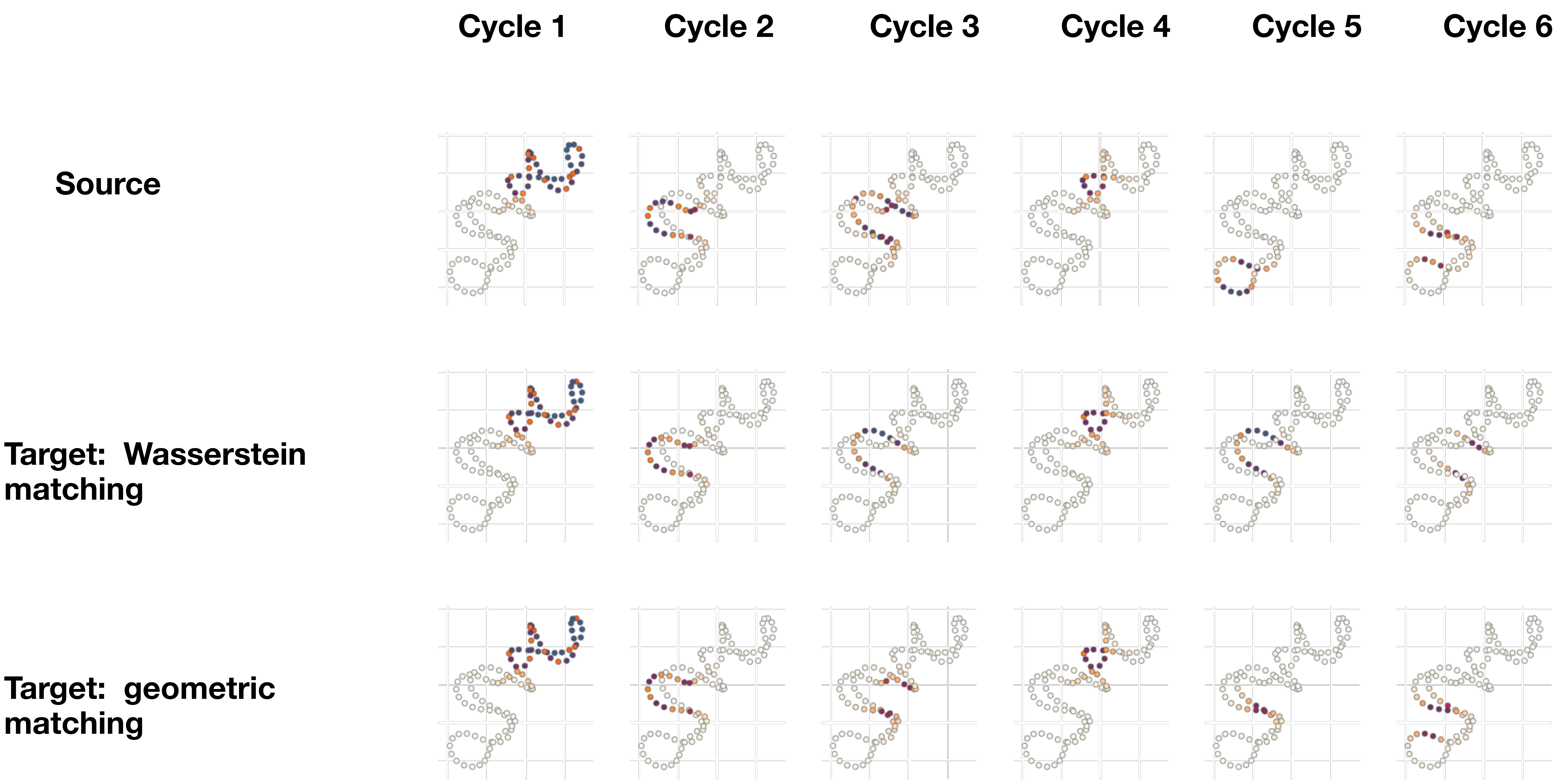
Source

Target: Wasserstein matching

Target: geometric matching



Geometric cycle matching



Geometric cycle matching

Cycle 1 | Cycle 2 | Cycle 3 | Cycle 4 | Cycle 5 | Cycle 6

Source

Target: Wasserstein matching

Target: geometric matching

Thanks!!