DISTANCE PRESERVING MODEL ORDER REDUCTION OF GRAPH-LAPLACIANS AND CLUSTER IMAGING

Mikhail Zaslavsky

Vladimir Druskin (Worcester Polytechnic Institute) and Alexander Mamonov (University of Houston)

Outline

Clustering problem. Formulation

- Unsupervised machine learning
- Input: data set $V = \{x_1, \dots x_N\}$ with no labels and number of partitions K
- Output: set of clusters $C_1, \dots C_K$ such that $\forall 1 \leq i \leq N \quad \exists 1 \leq l \leq K$ such that $x_i \in C_l$ based on similarity

Clustering problem. Formulation

- Unsupervised machine learning
- Input: data set $V = \{x_1, \dots x_N\}$ with no labels and number of partitions K
- Output: set of clusters $C_1, \dots C_K$ such that $\forall 1 \leq i \leq N \quad \exists 1 \leq l \leq K$ such that $x_i \in C_l$ based on similarity
- Data dimensionality: $x_i \in \mathbb{R}^d, i = 1, \dots, n$. d varies from $O(1)$ to $O(10^5)$ or even higher
- Number of clusters K is set a priori here

Clustering problem. Formulation

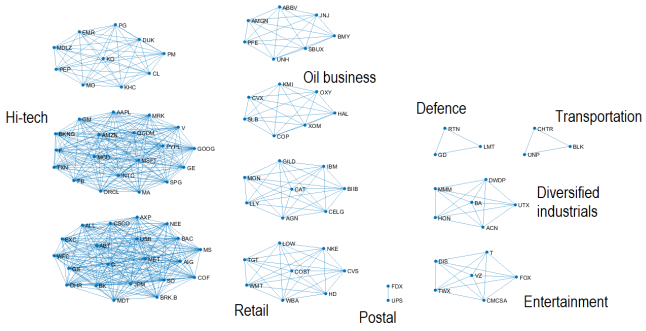
- Unsupervised machine learning
- Input: data set $V = \{x_1, \dots x_N\}$ with no labels and number of partitions K
- Output: set of clusters $C_1, \dots C_K$ such that $\forall 1 \leq i \leq N \quad \exists 1 \leq l \leq K$ such that $x_i \in C_l$ based on similarity
- Data dimensionality: $x_i \in \mathbb{R}^d, i = 1, \dots, n$. d varies from $O(1)$ to $O(10^5)$ or even higher
- Number of clusters K is set a priori here

Types of clustering:

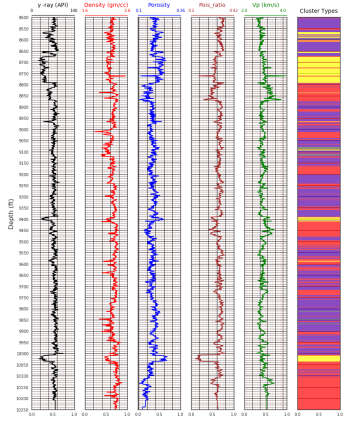
- Soft assignment: l is non-unique, i.e. different degrees (e.g., probability) of cluster membership
- Hard assignment: l is unique, i.e. each element belongs to only one cluster

Clustering problem. Applications

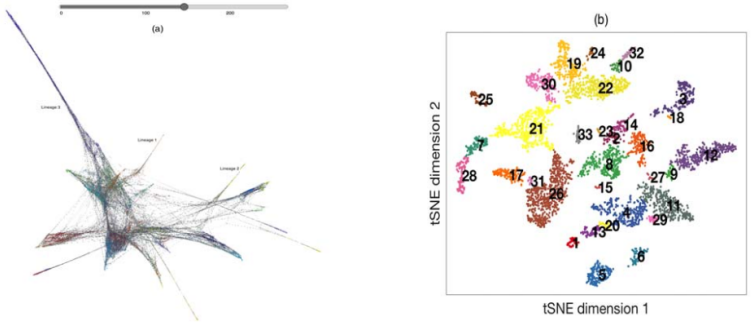
Finances



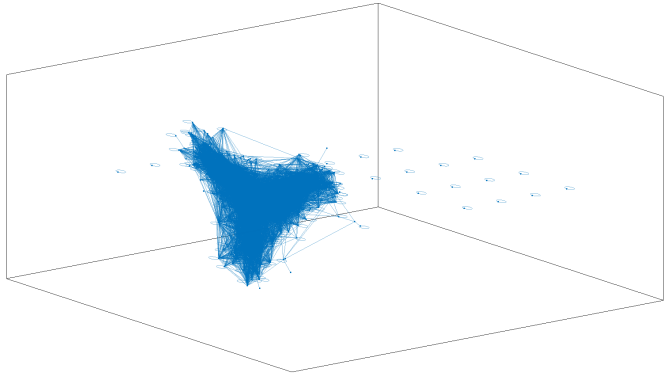
Geophysics



Bioinformatics



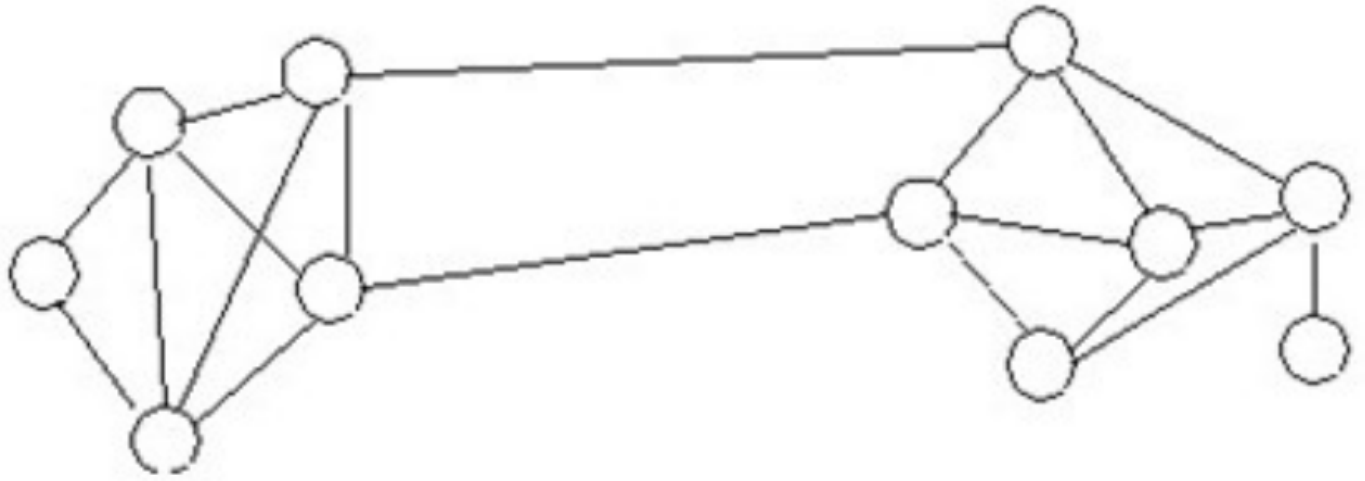
Organizational structure



Outline

Graph-based formulation

- Let $G = (V; E; \mathbf{W})$ be undirected weighted graph with vertices (data) V and edges E, with assigned positive weights W to them.
- Adjacency matrix $\mathbf{W} = (w_{ij})$ is a similarity measure between dataset elements.
- $d_i = \sum_j w_{ij}$ is i-th vertex degree. $\mathbf{D} = diag(d_1, \ldots, d_N)$ is degree matrix
- for $A \subseteq V$ let $vol(A) = sum_{i:x_i \in A} d_i$



Clustering via graph cuts: case of 2 clusters

- Idea: remove some edges to split V into 2 groups with minimum similarities between them

Clustering via graph cuts: case of 2 clusters

- Idea: remove some edges to split V into 2 groups with minimum similarities between them
- Normalized cut:

$$NCut(A,B) = \left(\frac{1}{vol(A)} + \frac{1}{vol(B)}\right) \sum_{x_i \in A, x_j \in B} w_{ij}$$

  - Integer optimization problem: find 2 normalized indicator functions $Z = (z_1, z_2)$ on V such that $tr(\mathbf{Z}^T(\mathbf{D} - \mathbf{W})\mathbf{Z}) \to min$ and $\mathbf{Z}^T\mathbf{D}\mathbf{Z} = \mathbf{I}$
  - Highly non-convex, especially for noisy data
  - Global minimization is combinatorial and NP-hard: cost of exact solution is $O(N^{Kd+1})$
  - Different robust relaxations are still expensive: Semi-Definite Programming (SDP) can handle the problems of size about 200 at reasonable time
  - How to overcome this prohibitive cost? Below we shall attack the problems of reducing both d and N

Relaxation: spectral clustering. Case of 2 clusters

[von Luxburg, 2007]

- $\min_{\mathbf{Z} \in \mathbb{R}^{N \times 2}, \mathbf{Z}^T \mathbf{D} \mathbf{Z} = \mathbf{I}} tr(\mathbf{Z}^T (\mathbf{D} - \mathbf{W}) \mathbf{Z})$
- Graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{W} \geq 0$ mimics Laplace operator with Neumann boundary conditions

Relaxation: spectral clustering. Case of 2 clusters

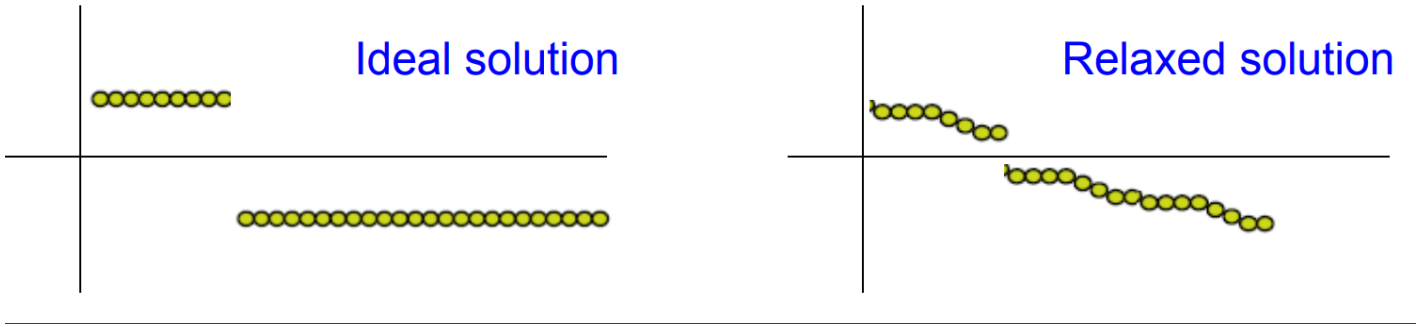
[von Luxburg, 2007]

- $\min_{\mathbf{Z} \in \mathbb{R}^{N \times 2}, \mathbf{Z}^T \mathbf{D} \mathbf{Z} = \mathbf{I}} tr(\mathbf{Z}^T (\mathbf{D} - \mathbf{W}) \mathbf{Z})$
- Graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{W} \geq 0$ mimics Laplace operator with Neumann boundary conditions

Relaxation: spectral clustering. Case of 2 clusters

[von Luxburg, 2007]

- $\min_{\mathbf{Z}\in\mathbb{R}^{N\times 2}, \mathbf{Z}^T\mathbf{D}\mathbf{Z}=\mathbf{I}} tr(\mathbf{Z}^T(\mathbf{D}-\mathbf{W})\mathbf{Z})$
- Graph Laplacian $\mathbf{L}=\mathbf{D}-\mathbf{W}\geq 0$ mimics Laplace operator with Neumann boundary conditions
- Solution are the first two eigenvectors of $\mathbf{L}u=\lambda\mathbf{D}u$. Real-valued and the first one is constant, so we split into clusters according to components sign of the second eigenvector



Spectral clustering. Algorithm for K clusters

[von Luxburg, 2007]

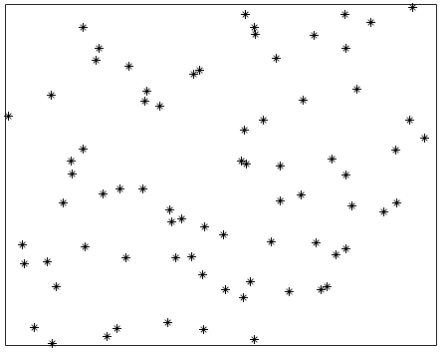
$$\min_{\mathbf{Z}\in\mathbb{R}^{N\times K},\mathbf{Z}^T\mathbf{D}\mathbf{Z}=\mathbf{I}} tr(\mathbf{Z}^T\mathbf{L}\mathbf{Z})$$

1. Compute $K-1$ eigenvectors $\{\mathbf{z}_i\}_{i=1}^{K-1}$ of $\mathbf{L}u = \lambda\mathbf{D}u$ for smallest eigenvalues such that $\mathbf{z}_i \perp \mathbf{e}$
2. Perform K-means on spectral data $\mathbf{Z} = (\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{K-1}) \in \mathbb{R}^{N\times K-1}$

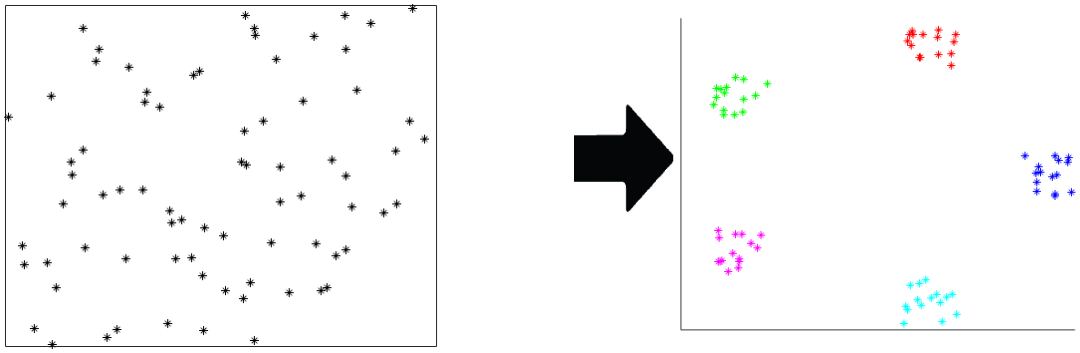
- Provides embedding into $(K-1)$-dimensional manifold
- Physical meaning: late-time asymptotics of diffusion process on graphs
- Spectral gap can provide some intuition on how to choose the number of clusters
- Cost=Linear algebra to construct embedding + K-means on embedded data
- However, size of dataset N is still large
- How can we reduce it?

Outline

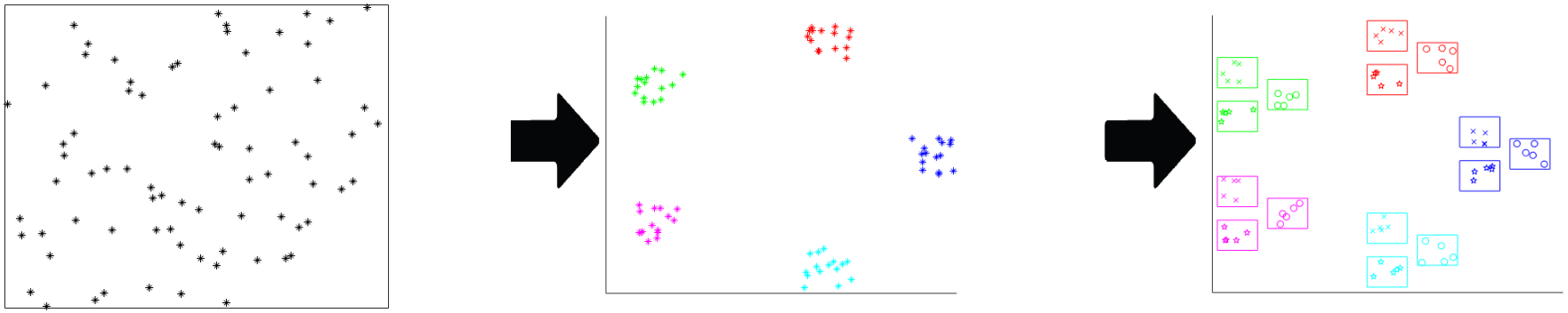
Sketch of the algorithm [Druskin, Mamonov, Z., 2019]



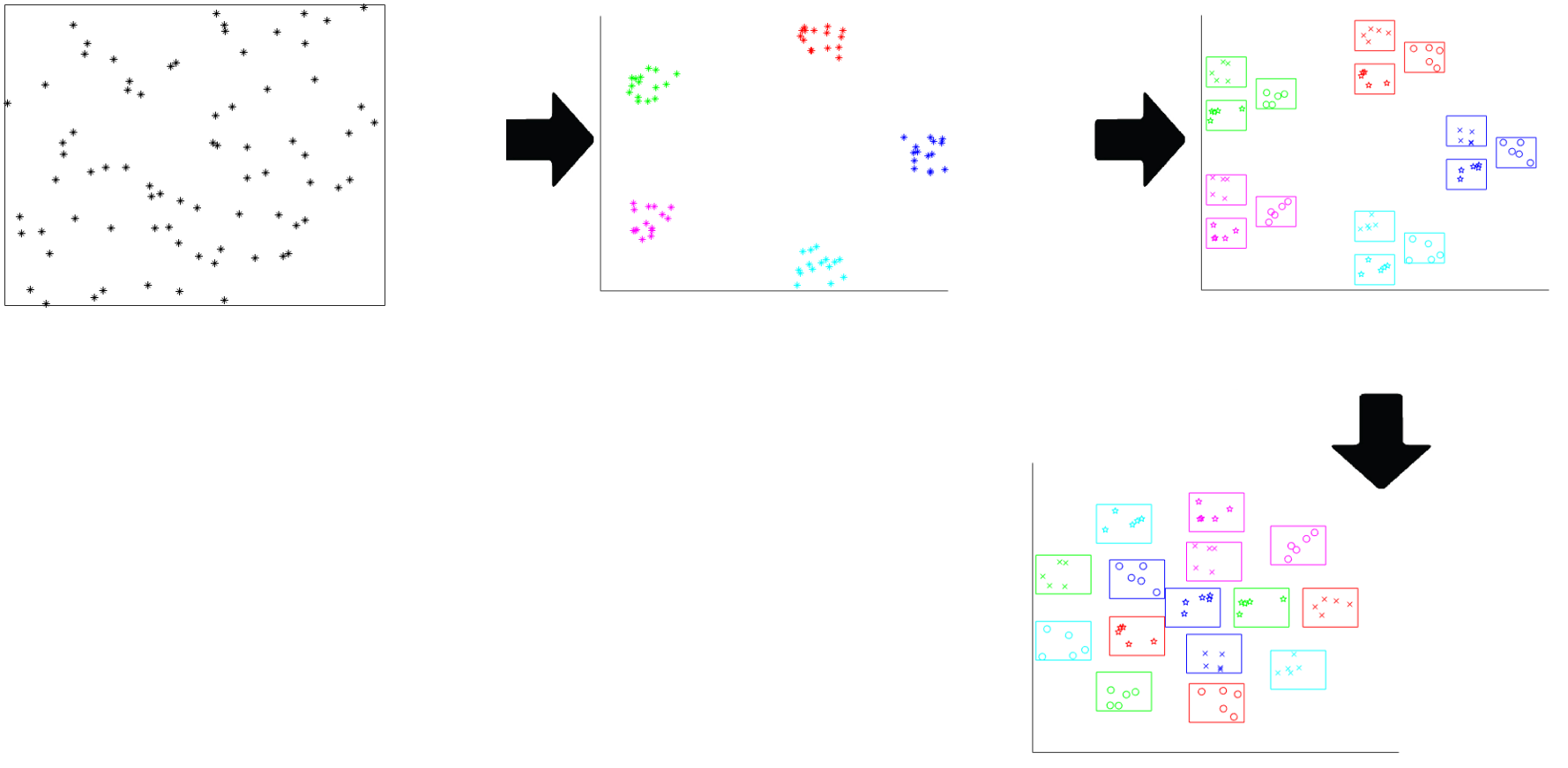
Sketch of the algorithm [Druskin, Mamonov, Z., 2019]



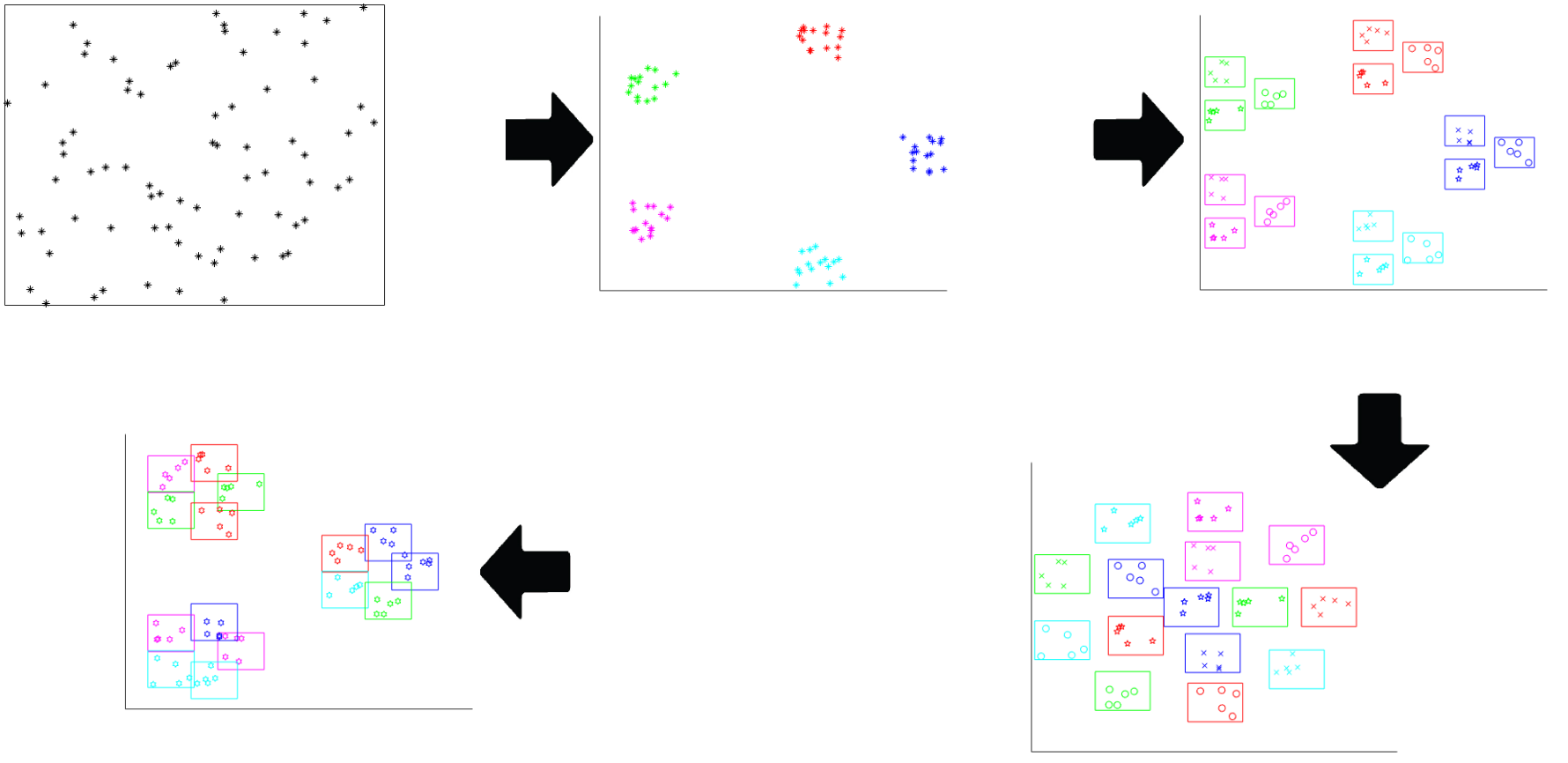
Sketch of the algorithm [Druskin, Mamonov, Z., 2019]



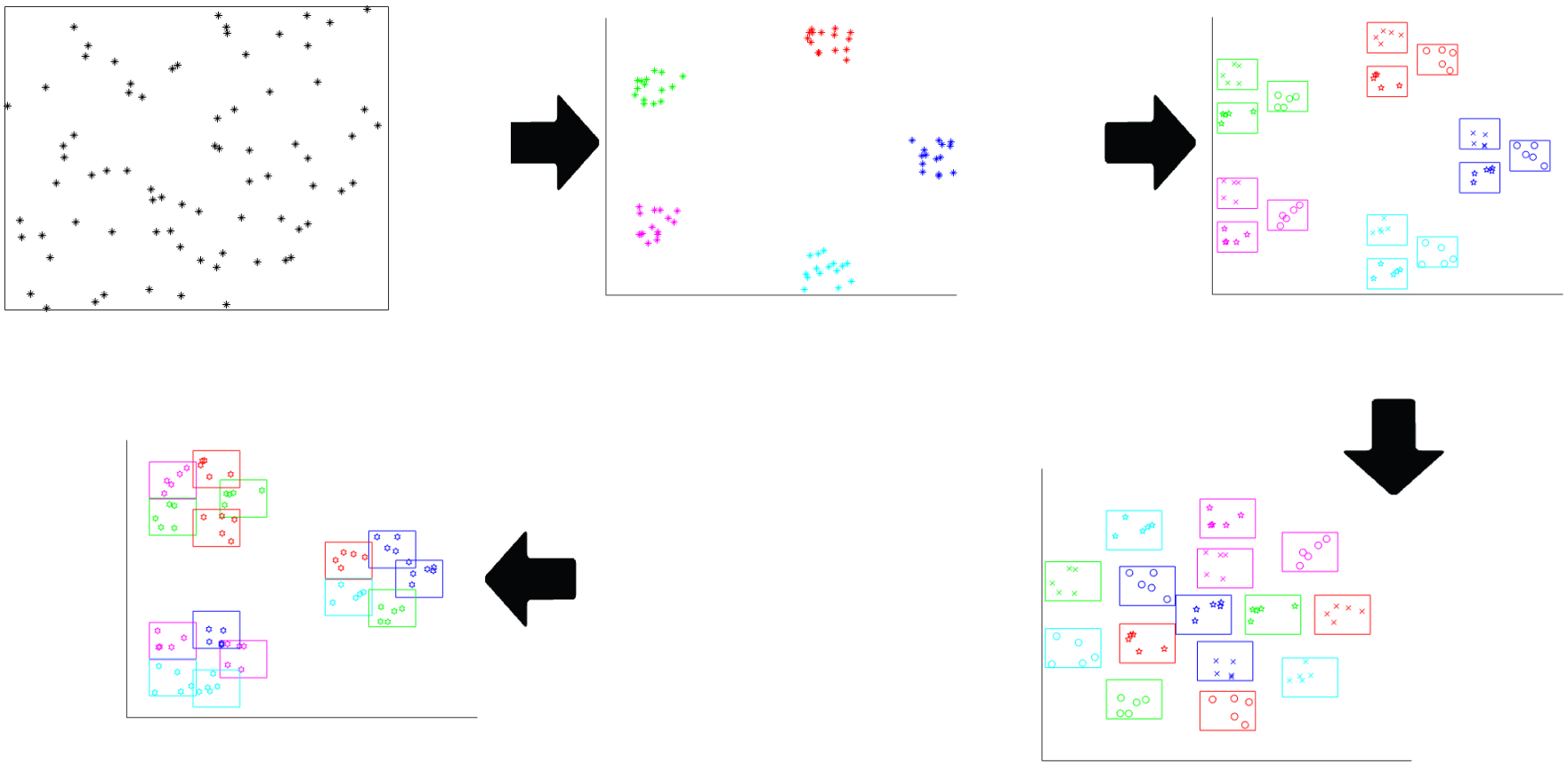
Sketch of the algorithm [Druskin, Mamonov, Z., 2019]



Sketch of the algorithm [Druskin, Mamonov, Z., 2019]

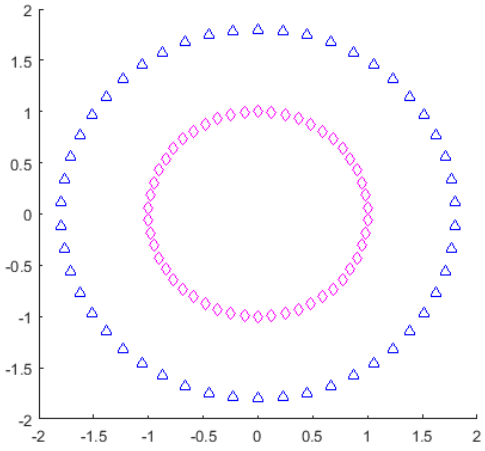


Sketch of the algorithm [Druskin, Mamonov, Z., 2019]

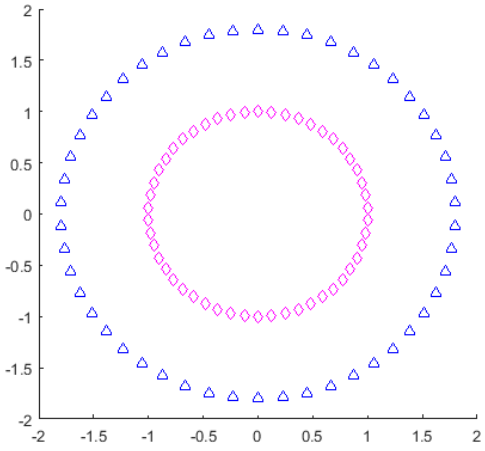


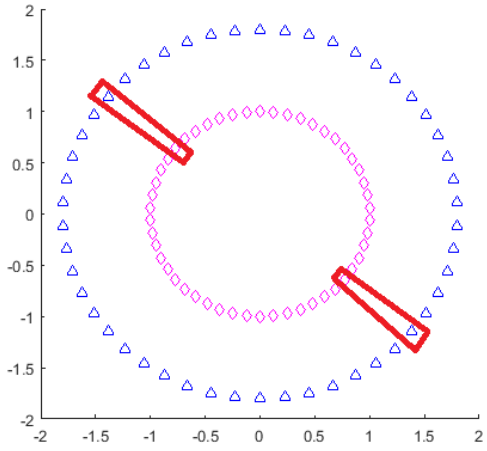
- Replace clustering of full dataset of size N by $\frac{N}{m}$ clusterings of subsets of size $O(m)$ each and one clustering of subset of size at most $K\frac{N}{m}$
- Need to cluster the data subset $V_m = \{x_{i_1}, x_{i_2}, \ldots, x_{i_m}\}$

Brute-force approach doesn't work

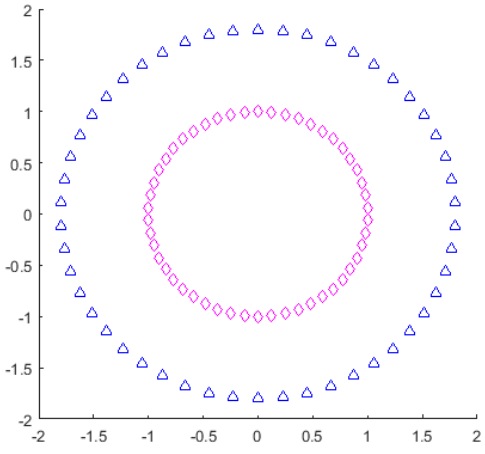


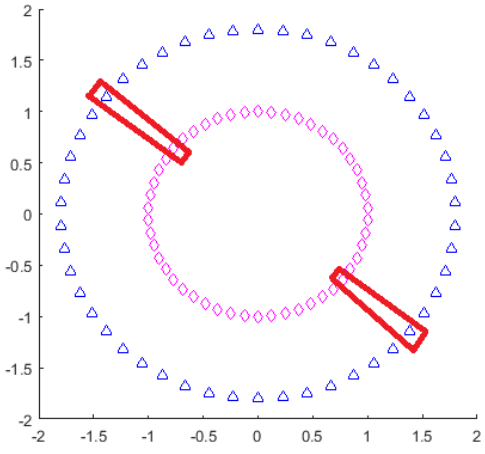
Brute-force approach doesn't work

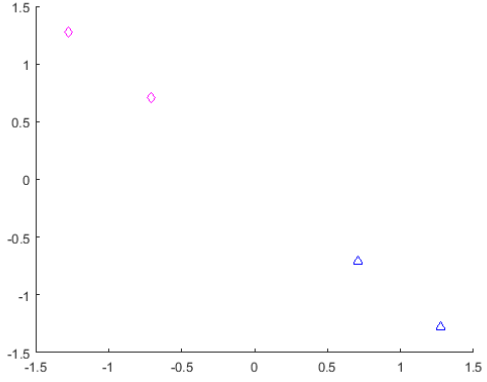




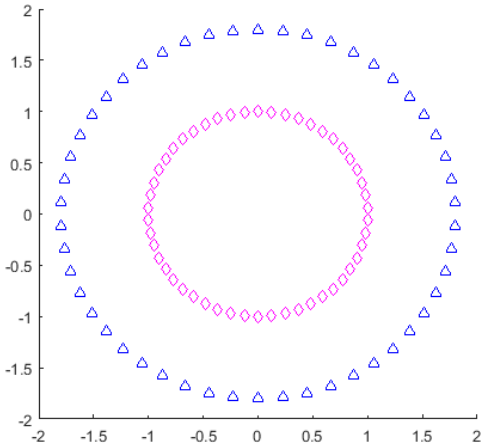
Brute-force approach doesn't work

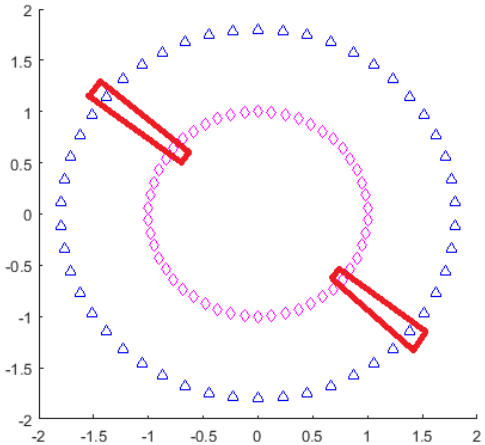


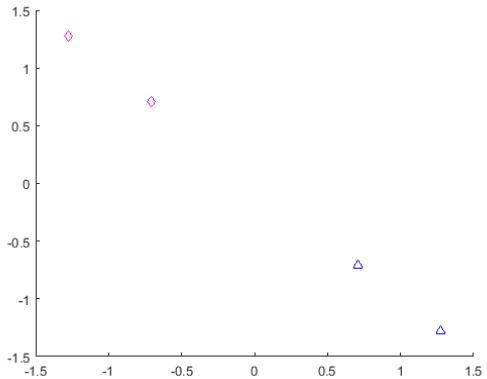




Brute-force approach doesn't work







Need to be aware of the FULL data manifold for diffusion on graph

Reduced-order model for data subset clustering

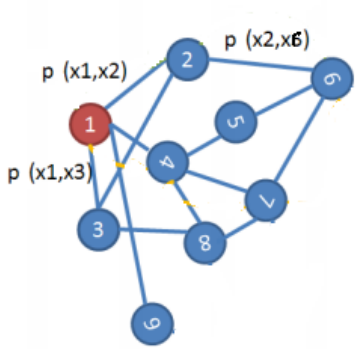
- Action of the subset complement onto the subset is governed by multi-input-multi-output transfer function
- Can be compressed with exponential accuracy via well-developed theory of model-order reduction: employ Krylov subspace $K_m((\mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2})^{\dagger}, \mathbf{D}^{1/2}\mathbf{B})$ for projection (Moore-Penrose inverse powers of symmetric normalized graph-Laplcian): $\widetilde{\mathbf{T}} = \mathbf{U}^T\mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2}\mathbf{U}$
- Can be diagonally transformed to sparse reduced-order graph Laplacian $\widetilde{\mathbf{L}} = \widetilde{\mathbf{D}}^{1/2}\widetilde{\mathbf{T}}\widetilde{\mathbf{D}}^{1/2}$

Advantages:

- Full dataset is replaced by ROM of size from $3m$ to $6m$
- Typically even lower, especially with multiple connected components
- Allows to employ more efficient relaxations, like SDP

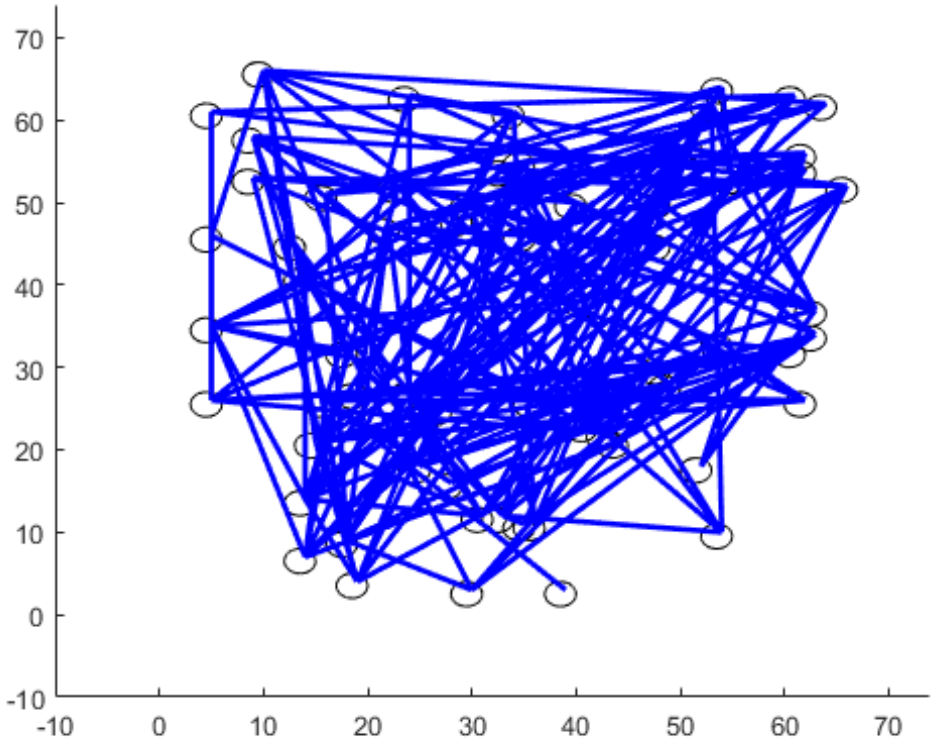
Random walks on graph and associated distances

- $\mathbf{L}_{RW} = \mathbf{I} - \mathbf{D}^{-1}\mathbf{W}$
- $\mathbf{P} = \mathbf{D}^{-1}\mathbf{W}$ is Markov matrix. It is transition matrix of random walk on graph: $p_{ij} = P(\mathbf{x}^{k+1} = x_j | \mathbf{x}^k = x_i)$

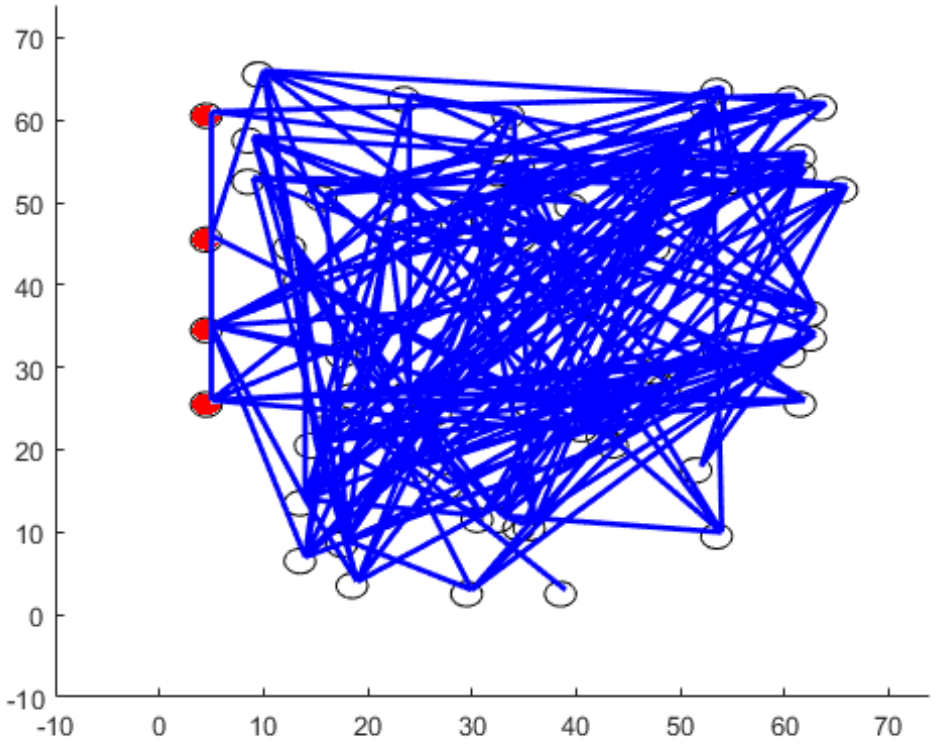


- Diffusion distance: distance between probability clouds originated at nodes i and j and measured after step l:
$$\left(D^l_{jk}(G)\right)^2 = \|P^l_j(G) - P^l_k(G)\|^2_{\mathbf{D}}$$
- Commute-time (or resistance) distance: $C_{ij} = H_{ij} + H_{ji}$ where $H_{ij} = p_{ij} + \sum_{k \neq j} p_{ik}(1 + H_{kj})$. It can be computed as $C^2_{jk}(G) = (\mathbf{e}_j - \mathbf{e}_k)^T \mathbf{L}^\dagger (\mathbf{e}_j - \mathbf{e}_k)$ where $\mathbf{L}^\dagger$ is Moore-Penrose pseudo-inverse of $\mathbf{L}$.
- For elements i and j from target set V_m commute-time distance C_{ij} is preserved and diffusion distance D^l_{ij} is approximated with exponential accuracy

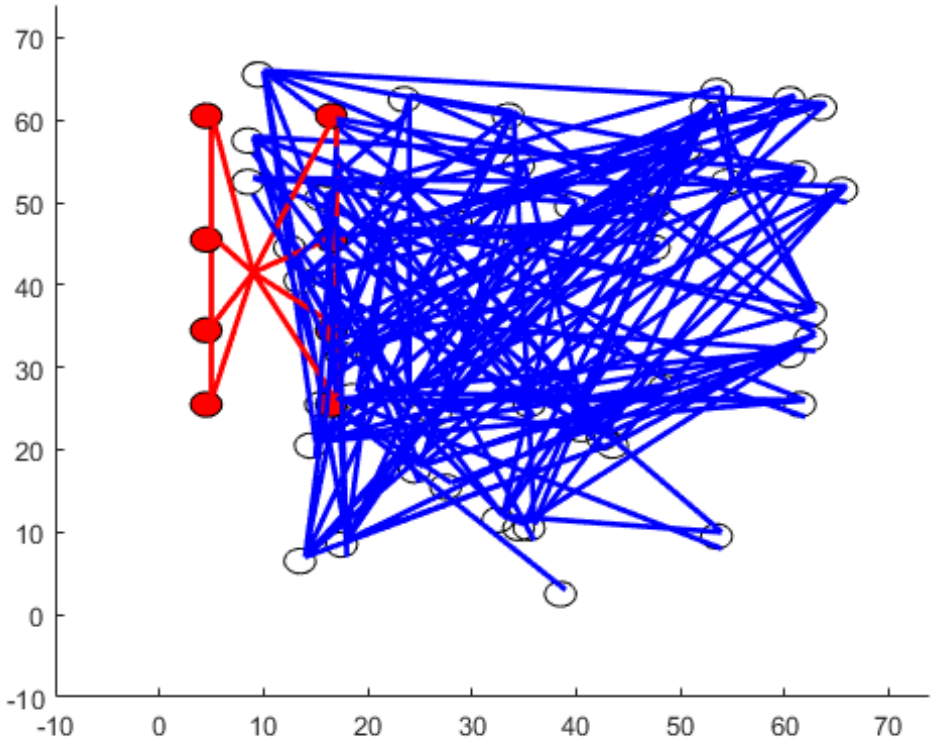
Geometrical interpretation of the reduced graph $\widetilde{G}$



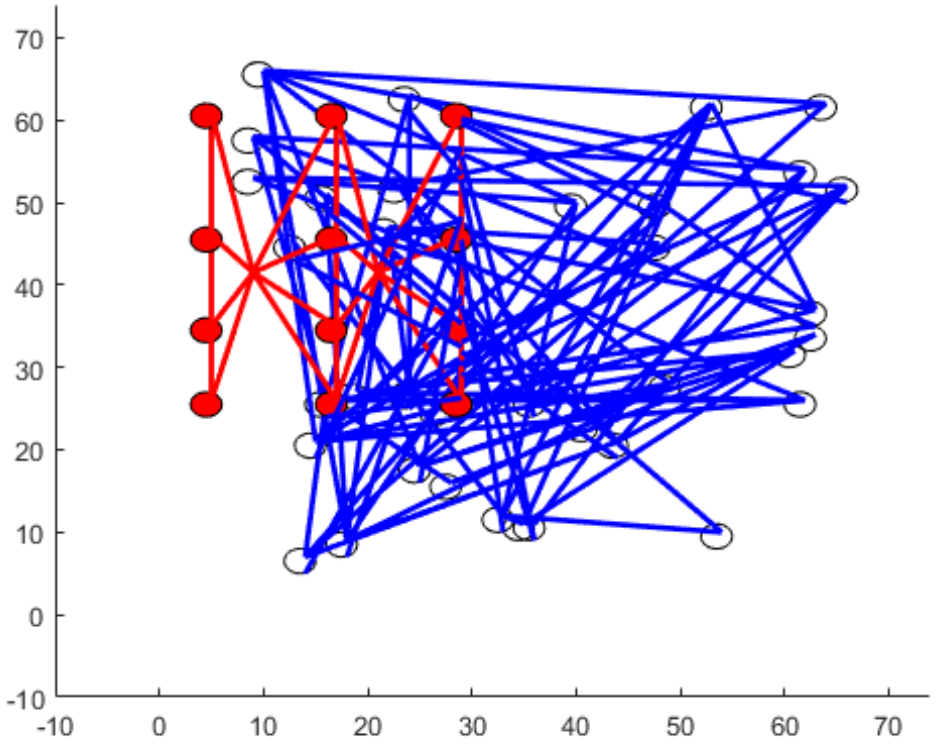
Geometrical interpretation of the reduced graph $\widetilde{G}$



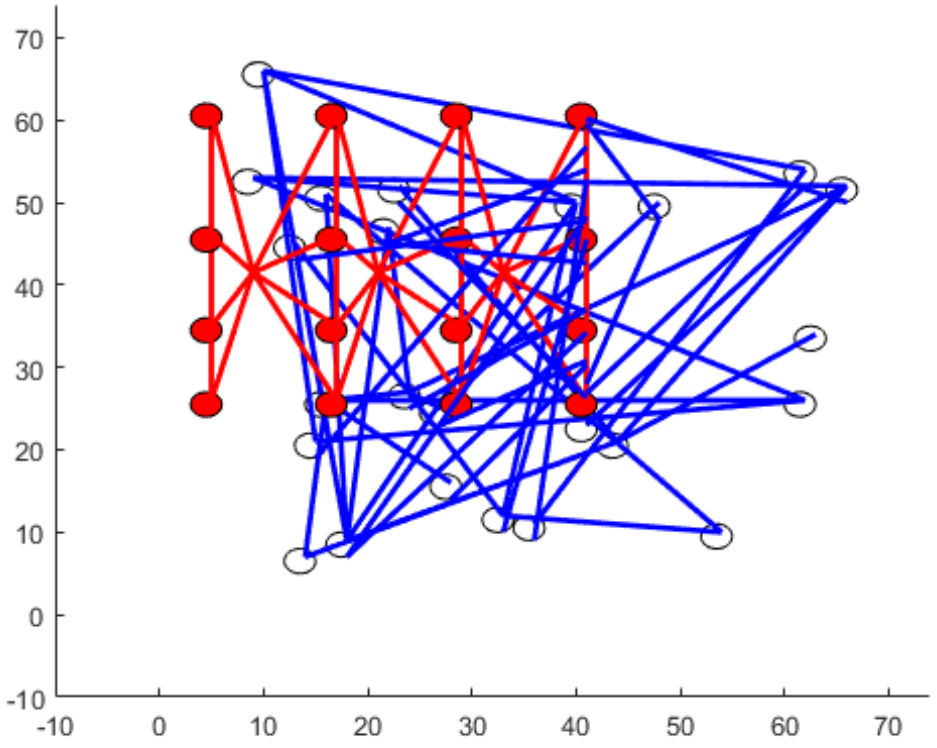
Geometrical interpretation of the reduced graph $\widetilde{G}$



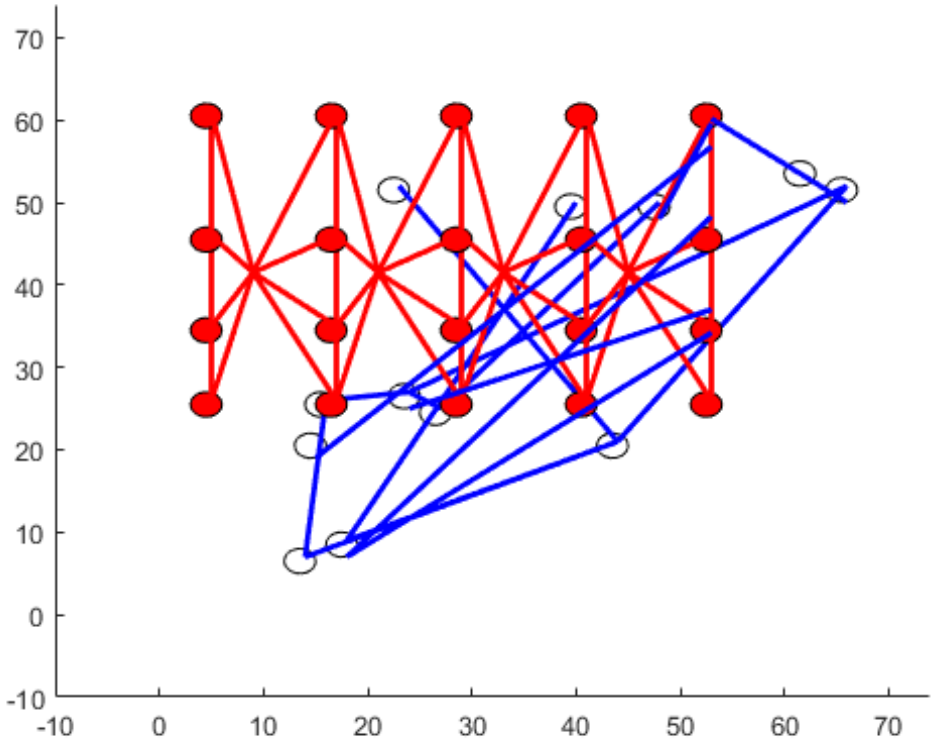
Geometrical interpretation of the reduced graph $\widetilde{G}$



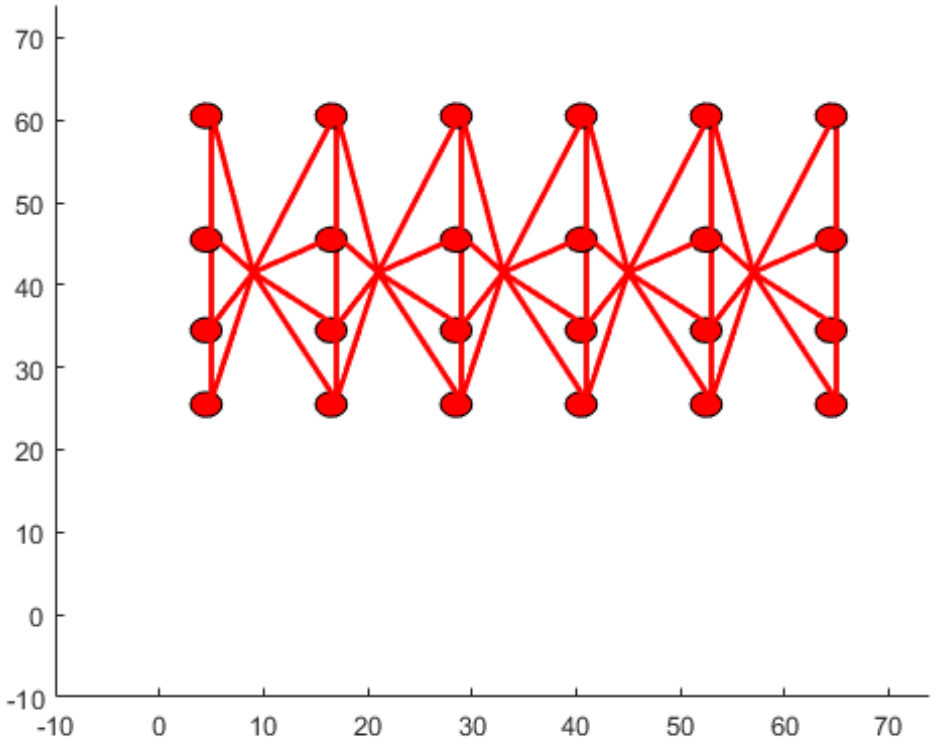
Geometrical interpretation of the reduced graph $\widetilde{G}$



Geometrical interpretation of the reduced graph $\widetilde{G}$



Geometrical interpretation of the reduced graph $\widetilde{G}$

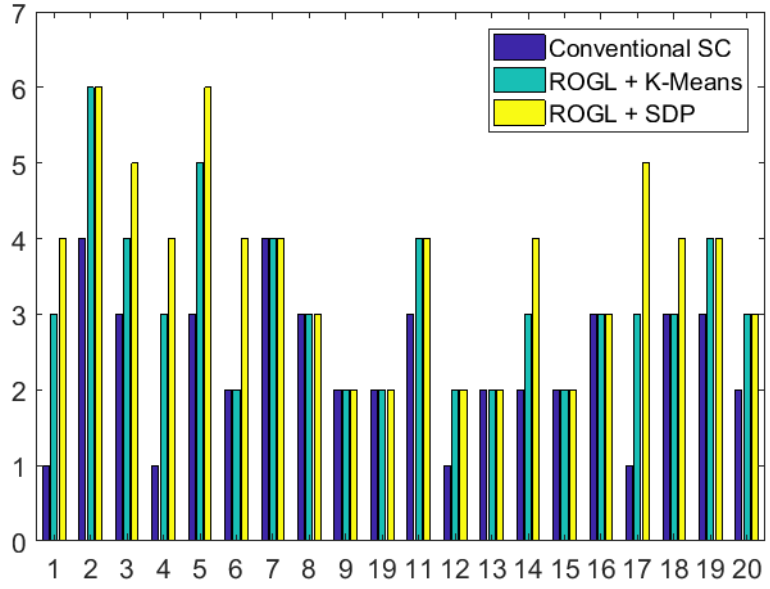


Outline

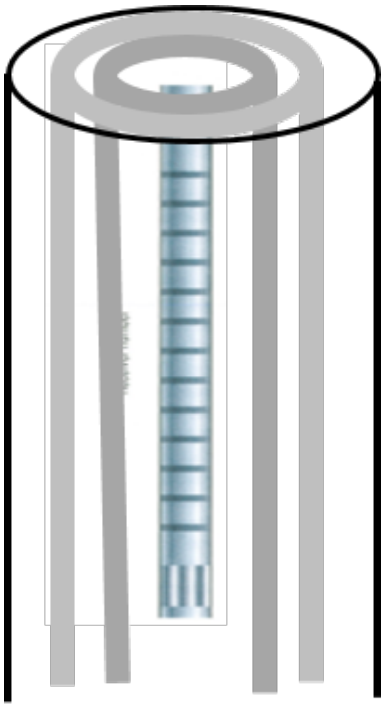
EMail communication network from SNAP repository

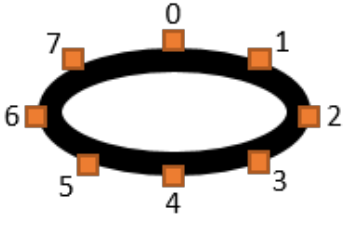
- Goal: identify people from the same organizations
- N=1005. 42 ground-truth communities (European organizations). Not enough data to resolve some of them

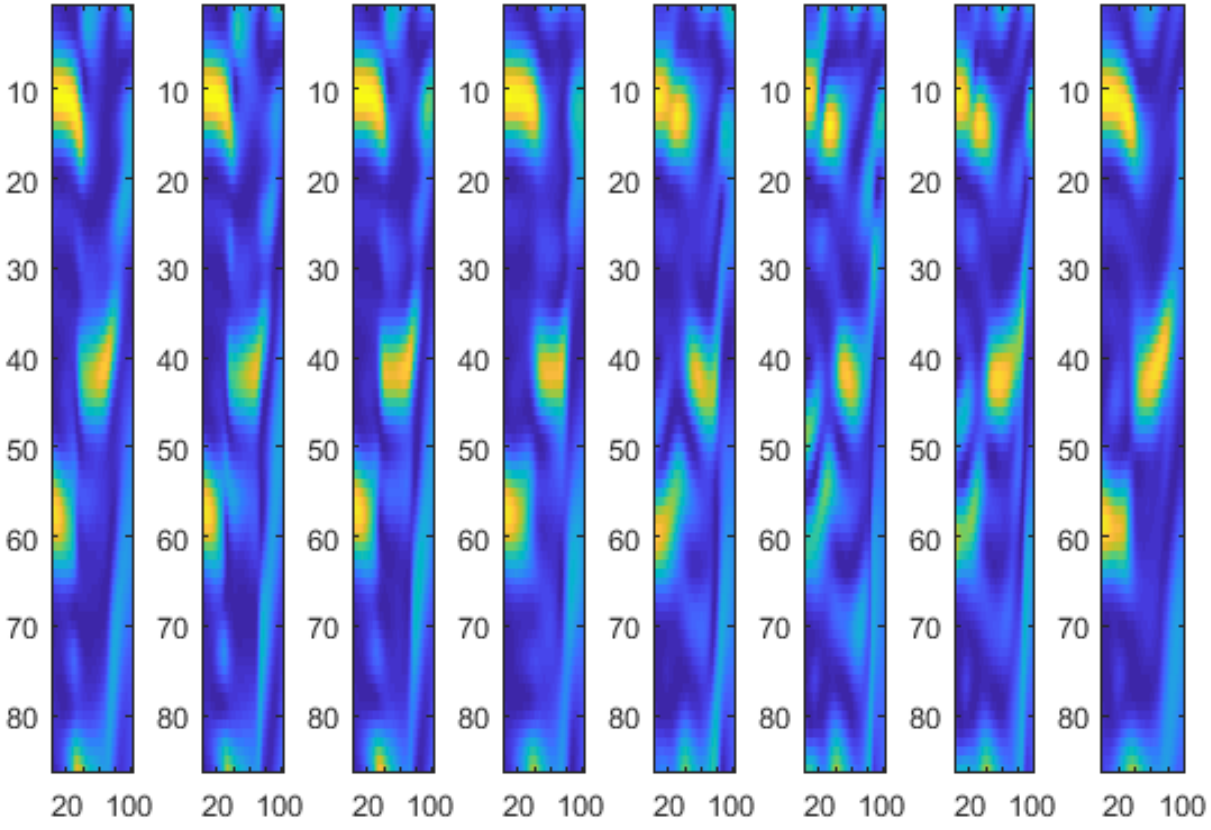
- Statistics of reproducing random elements from 10 random ground-truth communities
- Semi-definite programming shows the best results. Can be applied to ROGL only due to harsh computational cost



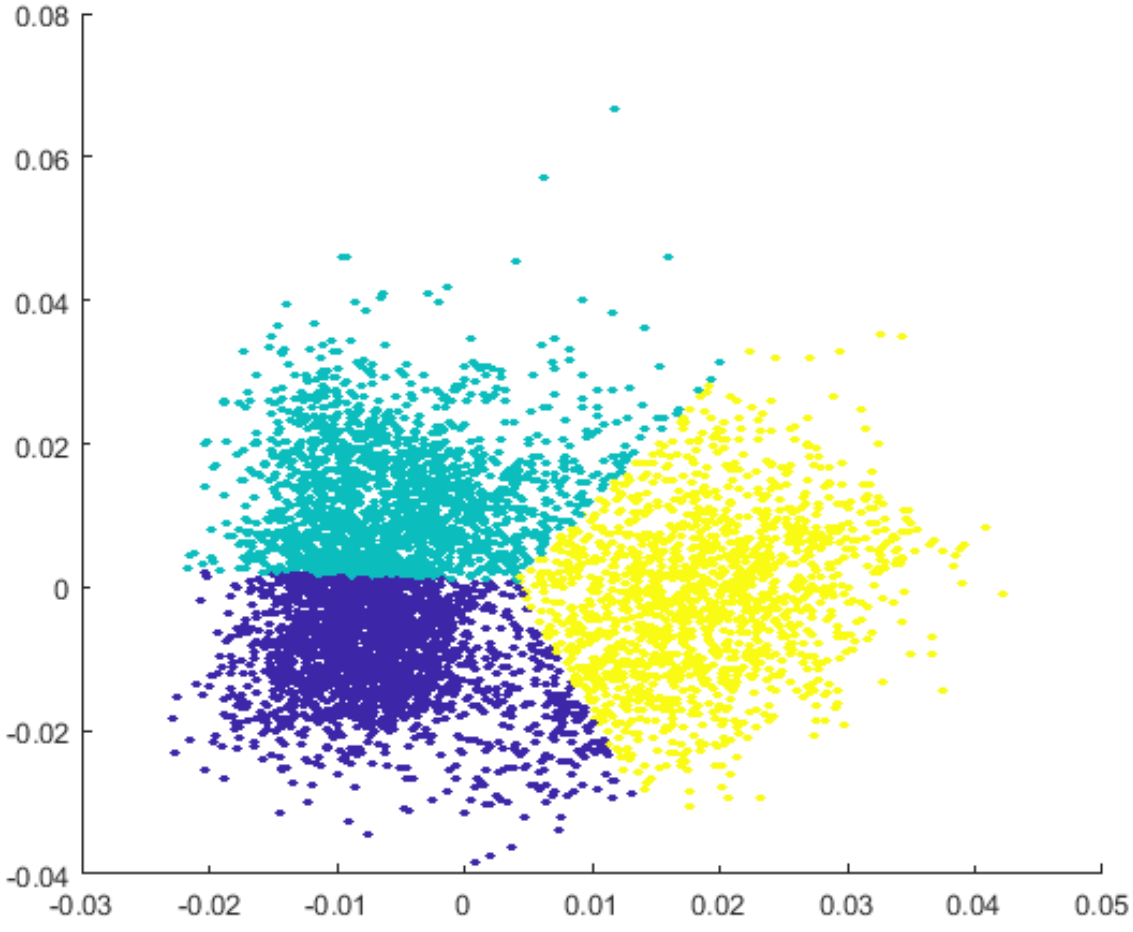
Sonic data





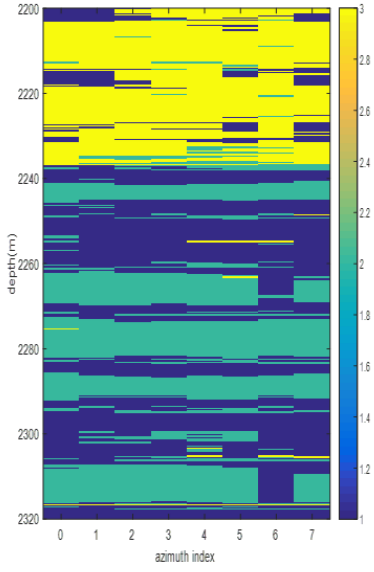


Spectral embedding

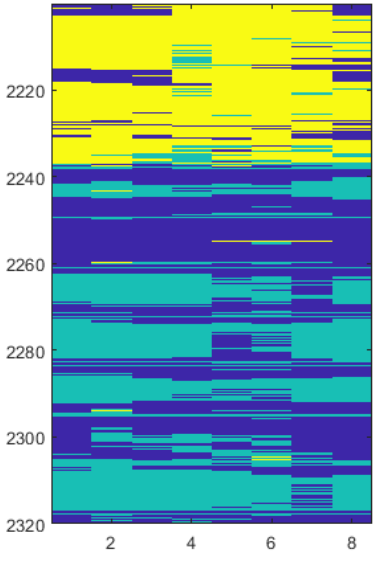


Clustering results

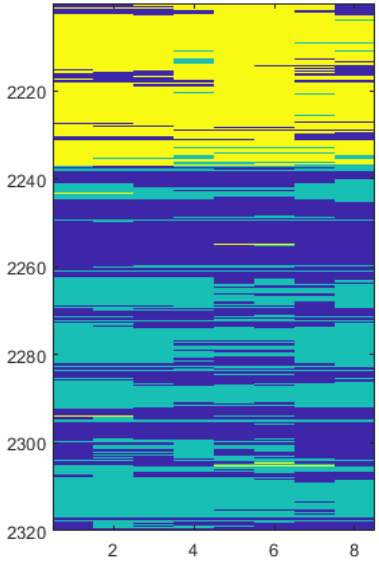
SOM



Spectral



Two-level



- Computational time on desktop: 15sec for SC and 50min for SOM

Outline

Conclusions

- We developed two-level clustering algorithm based on reduced-order model of graph-Laplacian for clustering of data subsets
- ROGL preserves with exponential accuracy such important physical features of the problem as transfer function and late-time diffusion distances for given data subset
- Significantly reduced size of the problem enables not only to speed up the computations but also to obtain qualitatively better results by exploiting NP-hard-type algorithms
- ROGL-based data subset clustering can be used as stand-alone when remaining data is not interesting
- Can be implemented in partial data-driven way via Loewner matrices framework (crucial for truly big data when entire graph is not directly accessible)

Future work

- Feasibility study on truly large datasets
- Multi-level approach
- Distance learning (in progress). Can be done in supervised and unsupervised manner