The International Joint
Conference on Neural Networks

Fast, Accurate Spectral Clustering Using Locally Linear Landmarks

Max Vladymyrov
Google Inc.

Miguel Á. Carreira-Perpiñán
EECS, UC Merced

May 17, 2017

Spectral clustering

Focus on the problem of Spectral Clustering, where we try to split the dataset $\mathbf{Y} \in \mathbb{R}^{D \times N}$ into a set of K meaningful clusters.

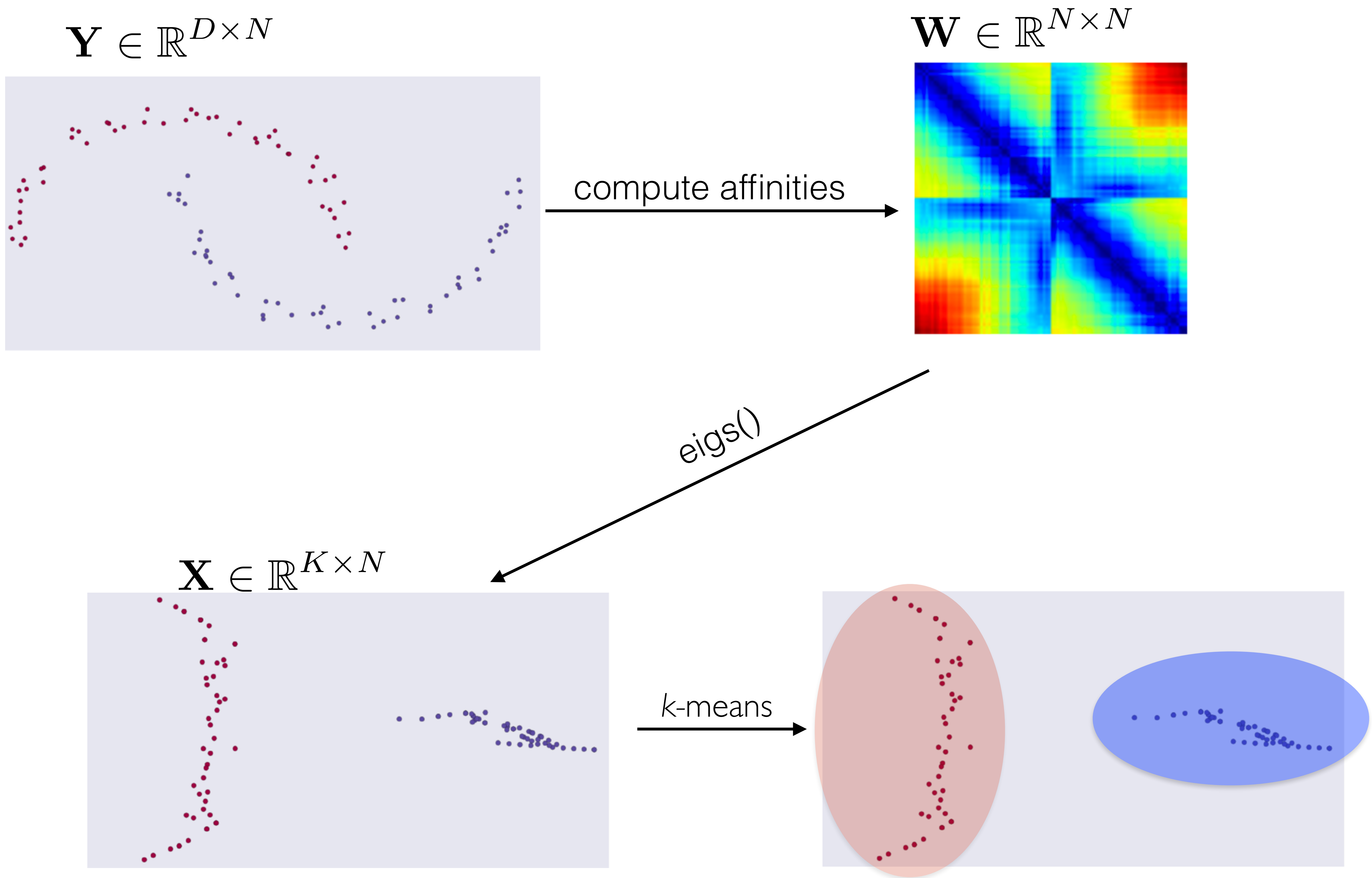
- Compute the affinity matrix $\mathbf{W} \in \mathbb{R}^{N \times N}$ (e.g. Gaussian $w_{ij} = e^{-\|(\mathbf{y}_i - \mathbf{y}_j)^2\|/2\sigma^2}$).
- Construct the degree matrix $\mathbf{D} = \operatorname{diag}(\mathbf{W1})$ and graph Laplacian (e.g. unnormalized $\mathbf{L} = \mathbf{D} - \mathbf{W}$).
- find the low-dimensional projection $\mathbf{X} \in \mathbb{R}^{K \times N}$, by minimizing

$$\min_{\mathbf{X}} \operatorname{tr}\left(\mathbf{XLX}^T\right), \text{ s.t. } \mathbf{XDX}^T = \mathbf{I}, \; \mathbf{XD1} = \mathbf{0}$$

- solution is given in the closed form by trailing eigenvectors of $\mathbf{D}^{-\frac{1}{2}}\mathbf{LD}^{-\frac{1}{2}}$
- apply k-means on normalized projection to achieve the final clustering.

Problem: does not scale when the number of points is large!

Spectral clustering

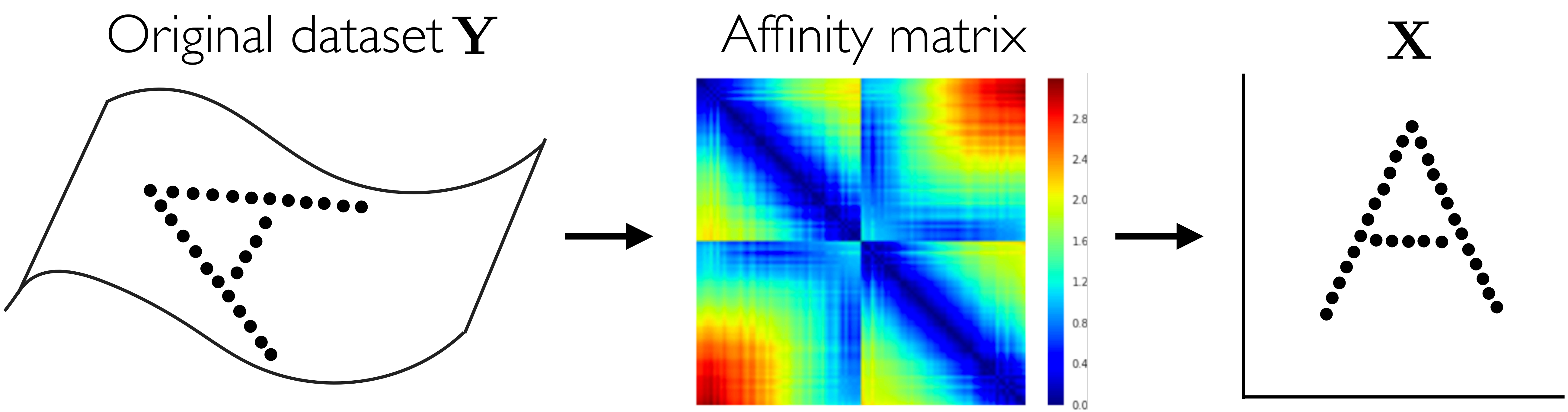


Learning with landmarks

Goal is find a fast, approximate solution for the embedding $\mathbf{X}$ using only the subset $\widetilde{\mathbf{Y}}$ of $L \ll N$ points from $\mathbf{Y}$.

Applications:

- When N is so large that the direct solution is infeasible.
- To select hyperparameters (e.g. for Gaussian kernel: k, σ) efficiently even if N is not large (since a grid search over these requires solving the eigenproblem many times).
- As an out-of-sample extension.

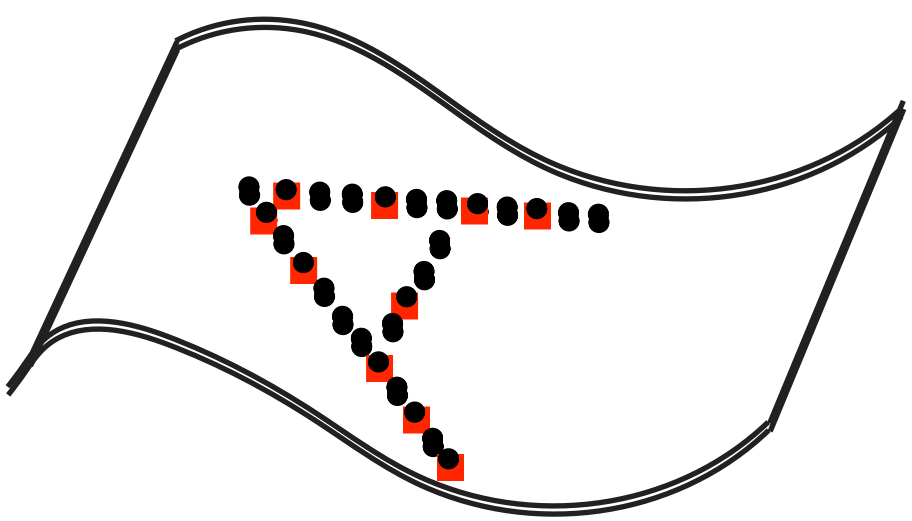


Learning with landmarks

$\Downarrow$

Original dataset $\mathbf{Y}$

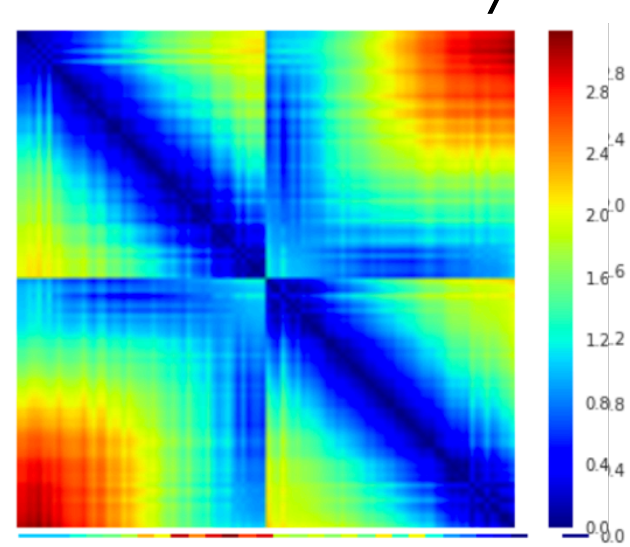
Landmarks $\widetilde{\mathbf{Y}}$



$\Rightarrow$

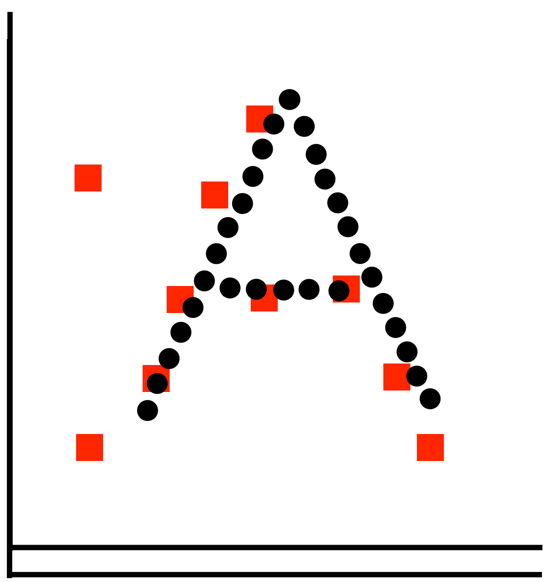
Affinity matrix

Reduced affinity matrix



$\Rightarrow$

$\widetilde{\mathbf{X}}$ $\mathbf{X}$

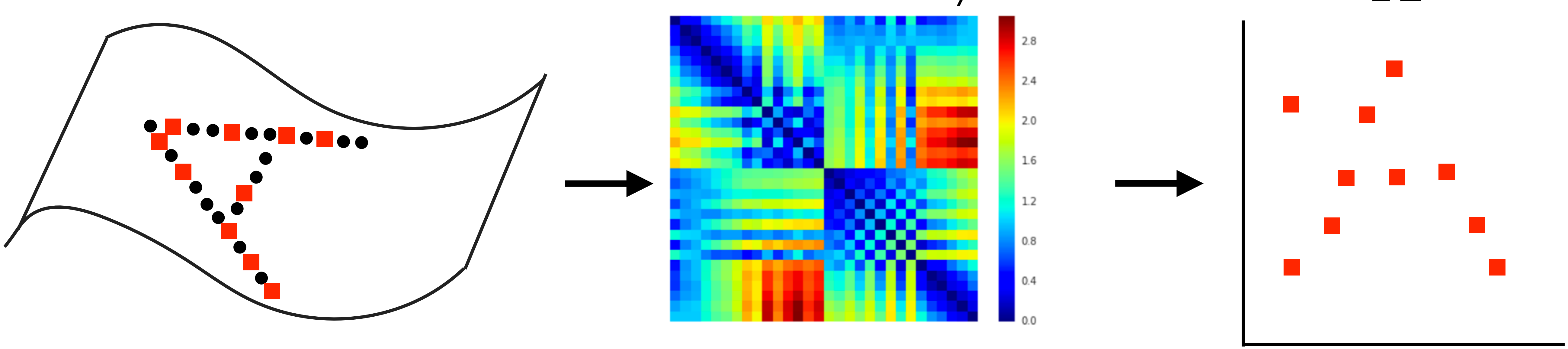


Learning with landmarks

Problems:

- We need a way to project the non-landmark points, e.g. with Nyström method (Talwalkar el at, 2008).
- Solving only on a subset $\widetilde{\mathbf{Y}}$ only, uses the information about the landmarks, but ignores information about non-landmarks. This requires using many landmarks to represent the data manifold well.
- If too few landmarks are used:
  - Bad solution for the landmark projection $\widetilde{\mathbf{X}} = \widetilde{\mathbf{x}}_1 \ldots, \widetilde{\mathbf{x}}_L$
  - …and bad prediction for the non-landmarks.

Landmarks $\widetilde{\mathbf{Y}}$ Reduced affinity matrix $\widetilde{\mathbf{X}}$



Locally Linear Landmarks (LLL)

(Vladymyrov and Carreira-Perpiñán, '13)

- For a set of landmarks $\widetilde{\mathbf{Y}}$ find a local reconstruction matrix $\mathbf{Z} \in \mathbb{R}^{N \times L}$ (e.g. by solving a liner system $\mathbf{Y} = \widetilde{\mathbf{Y}}\mathbf{Z}$). Locality is enforced by using only nearby landmarks when reconstructing $\mathbf{Y}$.

- Each projection then can be interpreted as a locally linear function of the landmarks:

$$\mathbf{x}_n = \sum_{l=1}^{L} z_{ln} \widetilde{\mathbf{x}}, n = 1, \ldots, N \Rightarrow \mathbf{X} = \widetilde{\mathbf{X}}\mathbf{Z}$$

- Solving the original eigenproblem of $N \times N$ with this constraint results in a reduced eigenproblem of the same form but of $L \times L$ on $\widetilde{\mathbf{X}}$:

$$\min_{\widetilde{\mathbf{X}}} \operatorname{tr}\left(\widetilde{\mathbf{X}}\widetilde{\mathbf{A}}\widetilde{\mathbf{X}}^T\right) \text{ s.t. } \widetilde{\mathbf{X}}\widetilde{\mathbf{B}}\widetilde{\mathbf{X}}^T = \mathbf{I}$$

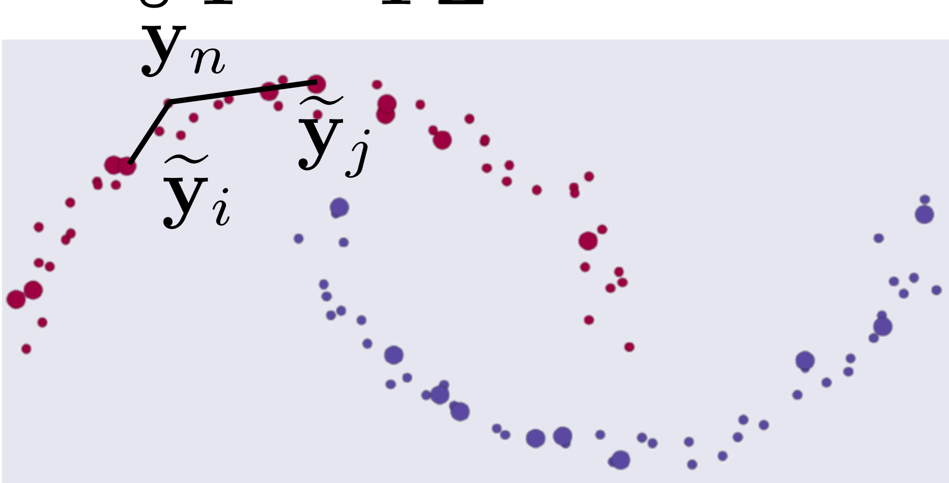
  with reduced affinities $\widetilde{\mathbf{A}} = \mathbf{Z}\mathbf{L}\mathbf{Z}^T, \widetilde{\mathbf{B}} = \mathbf{Z}\mathbf{D}\mathbf{Z}^T$.

- After $\widetilde{\mathbf{X}}$ is found, the non-landmarks are predicted as $\mathbf{X} = \widetilde{\mathbf{X}}\mathbf{Z}$.
  (out-of-sample mapping).

- Final k-means step on $\mathbf{X}$ to find a resulting clusters.

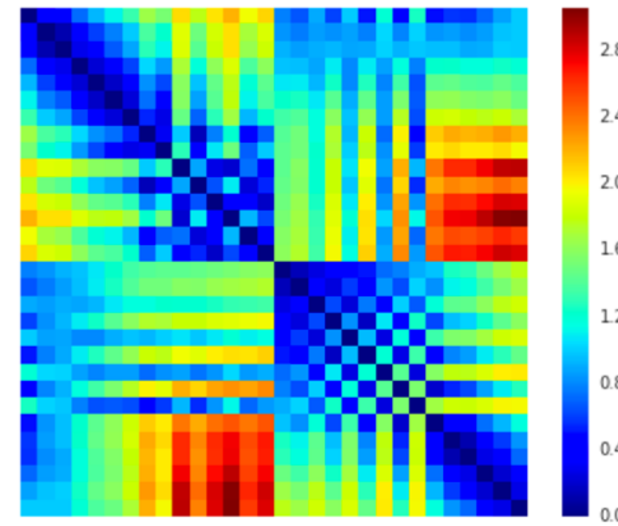
LLL: reduced affinities

compute reconstruction weights $\mathbf{Z}$
using $\mathbf{Y} = \widetilde{\mathbf{Y}}\mathbf{Z}$



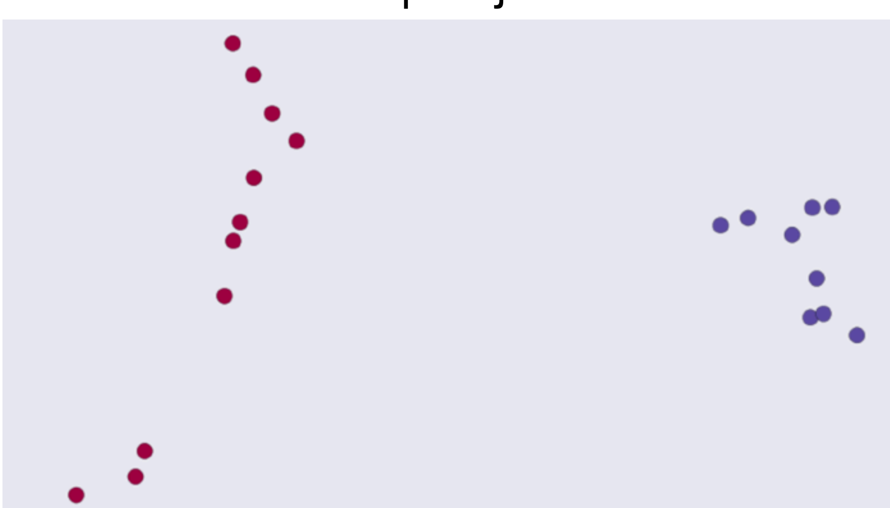
$\widetilde{\mathbf{A}} = \mathbf{ZLZ}$

reduced affinities $\widetilde{\mathbf{A}}$



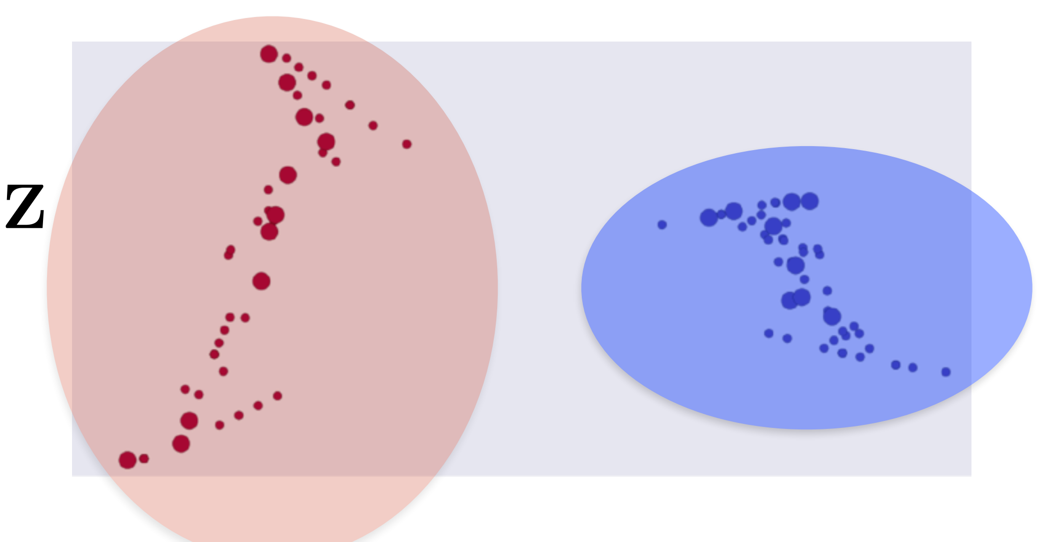
eigs()

landmark projection $\widetilde{\mathbf{X}}$



projection $\mathbf{X} = \widetilde{\mathbf{X}}\mathbf{Z}$

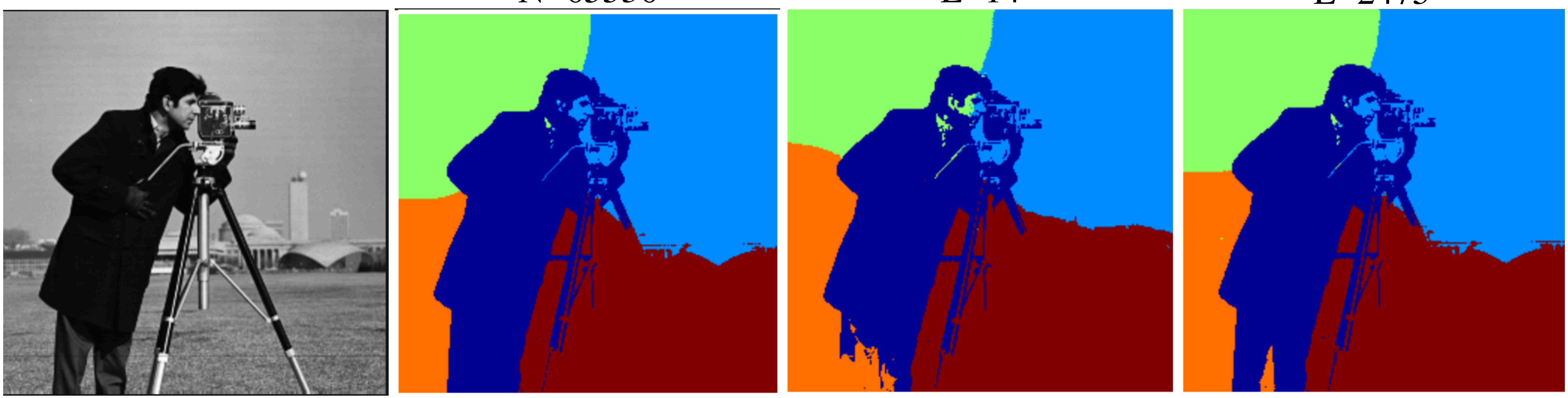
k-means



LLL for spectral clustering

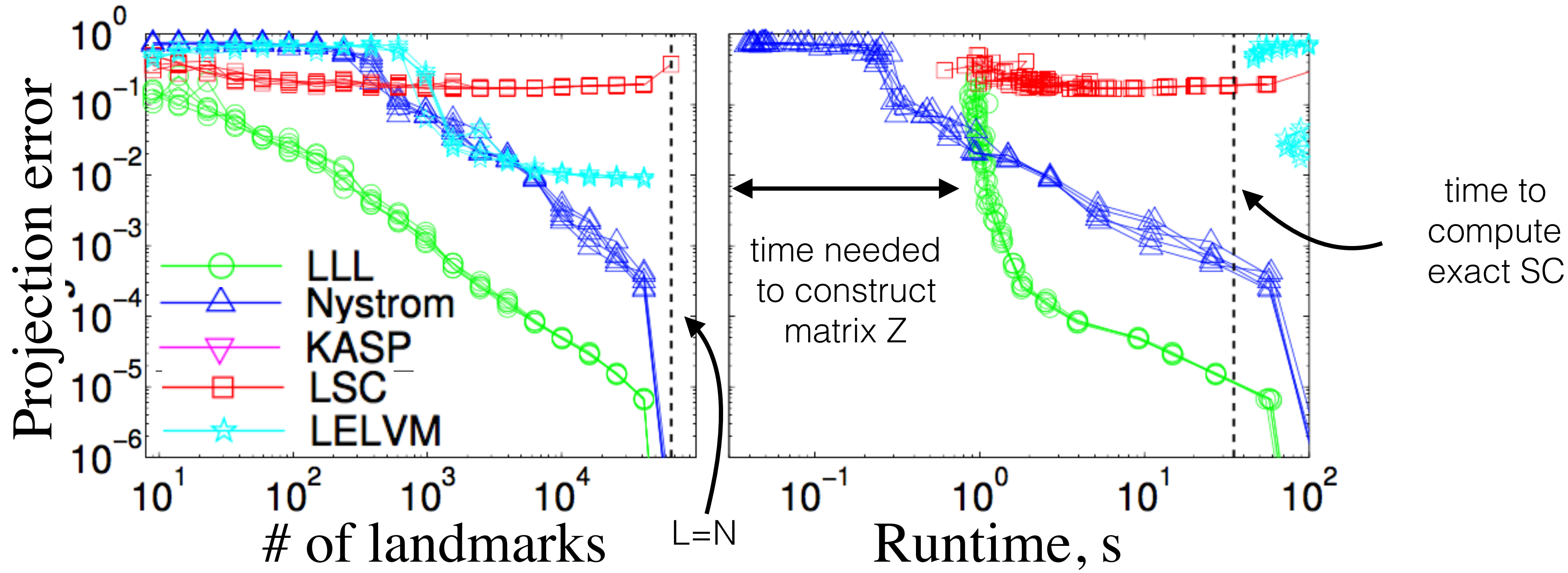
- Applied spectral clustering on 256x256 *cameraman* image (N=65536, D=3) with different number of landmarks.

Original | Exact SL, t=66 s, N=65536 | LLL, e=10%, t=0.9 s, L=14 | LLL, e=1%, t=1.82 s, L=2475

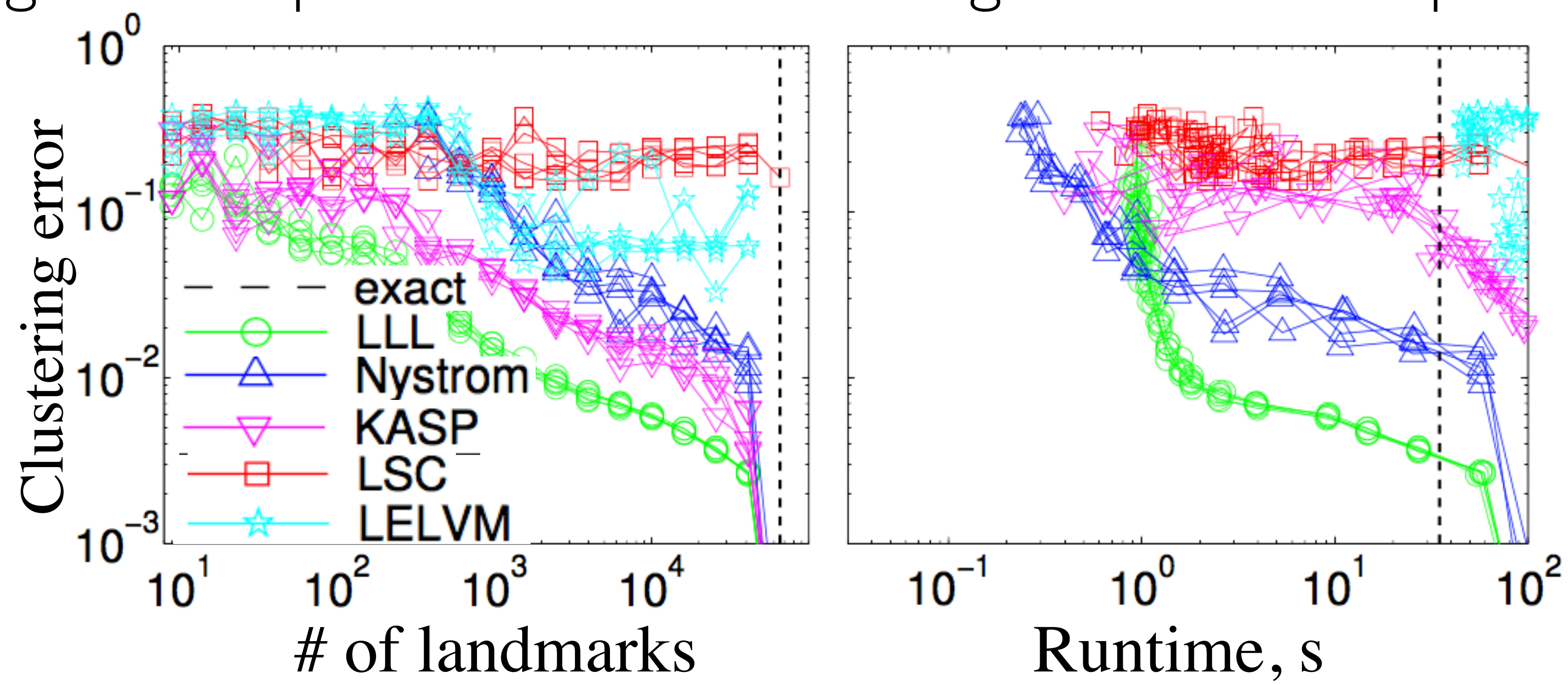


LLL for spectral clustering

- Projection error reports the error of the projection matrix X with respect to the projection matrix of the exact LE.



- Clustering error reports the final clustering error with respect to the exact LE.



LLL for custom affinities

1. Constrained spectral clustering. User provides additional must- and cannot- constrains.
2. Affinity aggregation. Using a linear combination of different affinities.
3. Motion Segmentation. affinity based on the spatio-temporal graph.

---

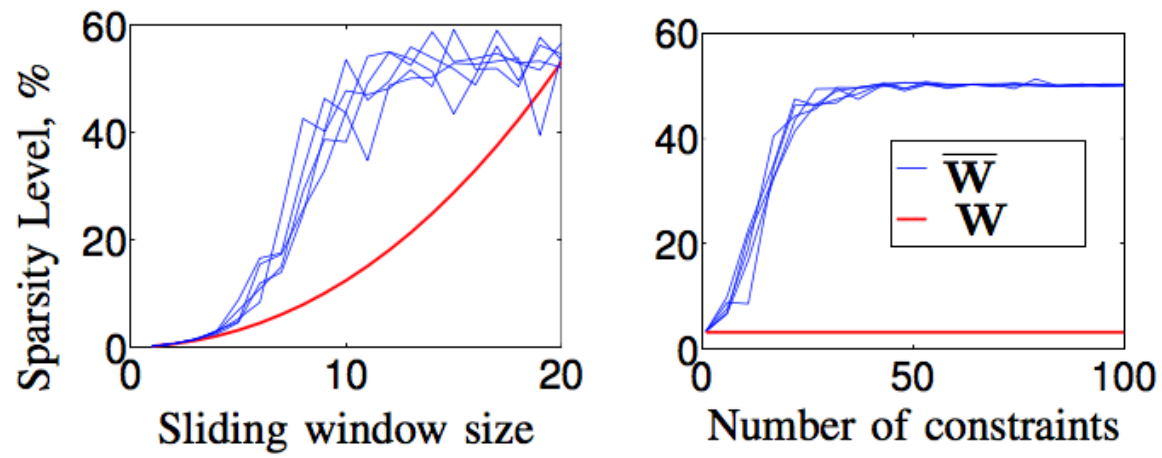
4. Proximity graph. Non-Gaussian affinity built using MST on multiple affinities.
5. LLL for hyperparameter selection. Fast search of the parameter space for the optimal set of hyperharapeters.
6. Out of sample extension. On-the-fly clustering assignment (using the landmark re-projection formula $\mathbf{X} = \widetilde{\mathbf{X}}\mathbf{Z}$).

Constrained spectral clustering

- We follow framework from (Lu and Carreira-Perpiñán, 2008).
- User additionally provides must and cannot link constraints using the sparse matrix $\mathbf{M}$. New affinities are computed as:

$$\overline{\mathbf{W}} = (\mathbf{W}^{-1} + \mathbf{M})^{-1} = \mathbf{W} - \mathbf{W}(\mathbf{I} + \mathbf{M}\mathbf{W})^{-1}\mathbf{M}\mathbf{W},$$

- Problems:
  - inverse $\mathbf{Q} = (\mathbf{I} + \mathbf{M}\mathbf{W})^{-1}\mathbf{M}$ is slow: solve by rearranging the elements inside sparse matrix $\mathbf{M}$.
  - $\overline{\mathbf{W}}$ is much denser:



- Using LLL reduced affinities:

$$\mathbf{Z}\overline{\mathbf{L}}\mathbf{Z}^T = \mathbf{Z}\overline{\mathbf{D}}\mathbf{Z}^T - \mathbf{Z}\overline{\mathbf{W}}\mathbf{Z}^T$$

$$= \mathbf{Z}\,\mathrm{diag}\left(\mathbf{W}\mathbf{1}\right)\mathbf{Z}^T - \mathbf{Z}\,\mathrm{diag}\left(\mathbf{W}\mathbf{Q}\mathbf{W}\mathbf{1}\right)\mathbf{Z}^T - \mathbf{Z}\mathbf{W}\mathbf{Z}^T + \mathbf{Z}\mathbf{W}\mathbf{Q}\mathbf{W}\mathbf{Z}^T$$

By precomputing $\mathbf{Z}\mathbf{W}$ and rearranging terms the overall complexity is $\mathcal{O}(cNL)$ and we don't need to compute $\overline{\mathbf{W}}$.

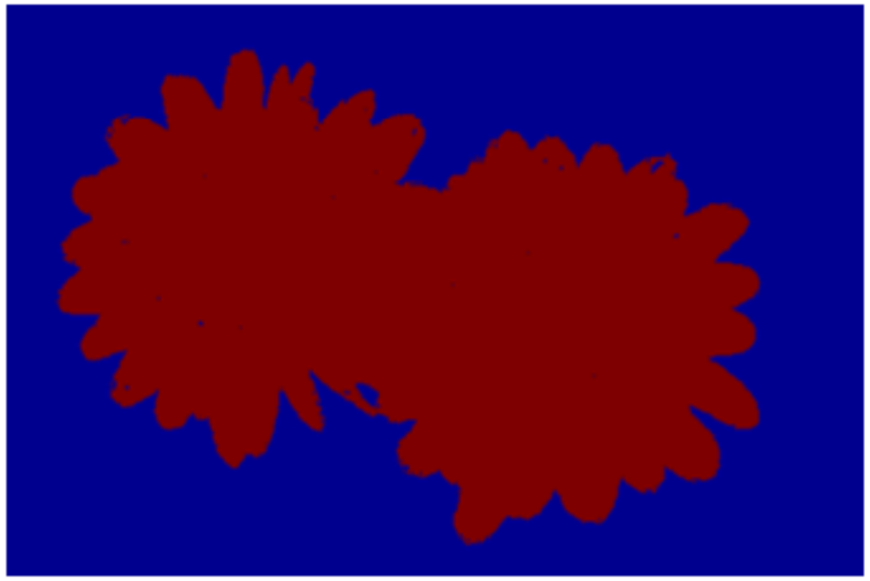
Constrained spectral clustering

- Problem of foreground segmentation:
  - added few constraints for an image of three different sizes,
  - run Spectral Clustering (SC) and Constrained Spectral Clustering (CSC).
- LLL achieves 10x speedup for SC and >20x for CSC.

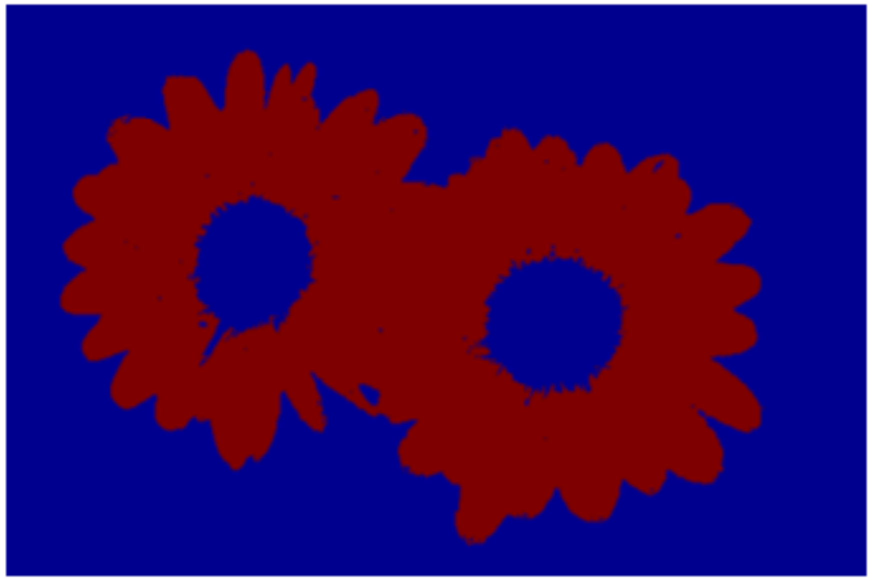
Image size	N	Time, seconds			
		SC,Exact	SC,LLL	CSC,Exact	CSC,LLL
Small (64×94)	6 016	4.47	0.87	5.14	0.51
Medium (160×240)	38 400	44	4.66	104.49	6.51
Large (321×481)	154 401	512.01	48.19	*out-of-memory*	59.98

Original image

SC with LLL



CSC with LLL



Affinity aggregation for spectral clustering

We follow (Huang et al, 2012) that propose to combine many affinities together.

- For a weighted affinity matrix $\overline{\mathbf{W}} = \sum_{k=1}^{K} v_k^2 \mathbf{W}^{(k)}$ minimize:

$$\min_{\mathbf{X},\mathbf{v}} \mathbf{X}\overline{\mathbf{L}}\mathbf{X}^T, \text{s.t. } \mathbf{X}\overline{\mathbf{D}}\mathbf{X}^T = \mathbf{I}, \mathbf{v}^T\mathbf{1} = 1,$$

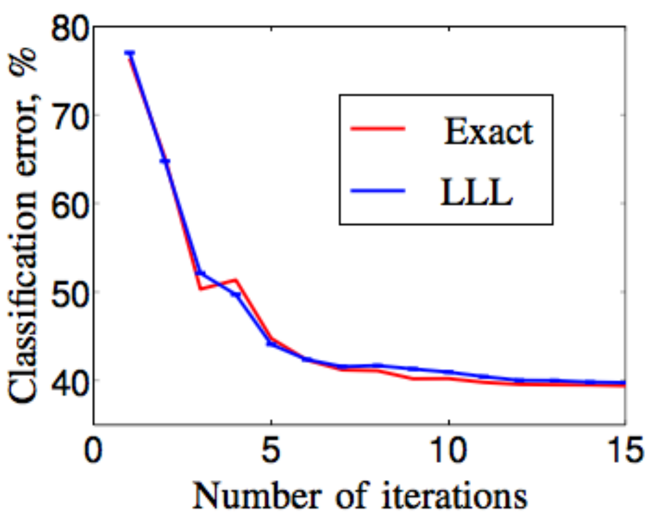
- Solve by alternating minimization between $\mathbf{v}$ and $\mathbf{X}$. For each iteration:
  - $\mathbf{v}$: 1D root-finding that can be solved in a few iterations.
  - $\mathbf{X}$: eigendecomposition of $\overline{\mathbf{L}}$.
- Using LLL:

$$\widetilde{\mathbf{L}} = \mathbf{Z}\overline{\mathbf{D}}\mathbf{Z}^T - \mathbf{Z}\overline{\mathbf{W}}\mathbf{Z}^T = \sum_{k=1}^{K} v_k^2 \mathbf{Z}\mathbf{D}^{(k)}\mathbf{Z}^T - \sum_{k=1}^{K} v_k^2 \mathbf{Z}\mathbf{W}^{(k)}\mathbf{Z}^T.$$

Each of $\mathbf{Z}\mathbf{W}^{(k)}\mathbf{Z}^T$ can be precomputed just once and are $L \times L$. The rest of operations are independent from N.

Affinity aggregation for spectral clustering

- We used N=11368 faces from CMU-PIE dataset (each face is an 64x64 image in near frontal position).
- We used 3 types of affinities:
  - Local Binary Pattern (LBP),
  - Gabor texture,
  - Eigenfaces.



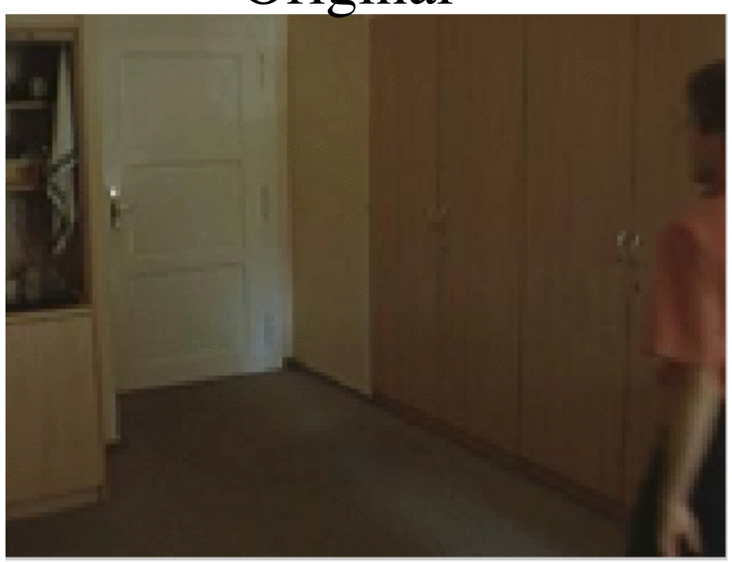
Features		Single affinities LBP	Single affinities Gabor filter	Single affinities Eigenface	Combined AASC
Class. error, %	Exact	43	48	56	39
	LLL	44 ± 1	49 ± 2	56 ± 1	39 ± 3
Runtime, s	Exact	78	85	105	1 063
	LLL	8.15 ± 0.5	8.76 ± 0.7	8.23 ± 0.8	28.2 ± 2.1
Proj. error, %		4 ± 0.8	3 ± 1	1 ± 0.5	3 ± 2

- Combined affinities achieve lower error than any affinity on its own.
- LLL achieves similar accuracy with 35x speedup.

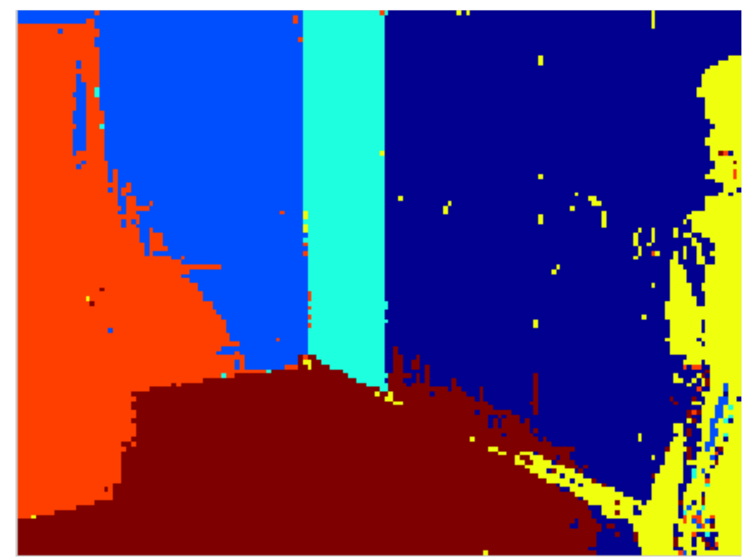
Motion segmentation

- 41 frames with each frame 120x160 image. Spatio-temporal affinities with 2 spatial, 3 color and 1 for a frame number. Overall N=787200 with D=6.
- Used L=5000 landmarks and t=3 minutes.

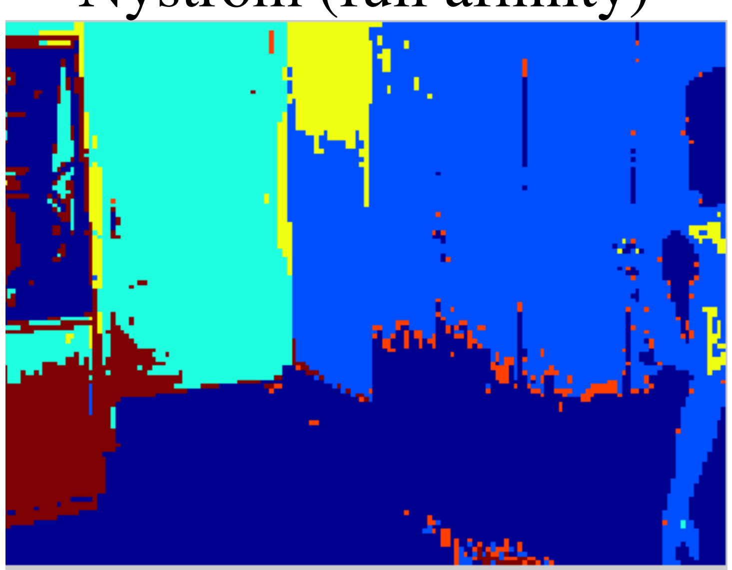
Original



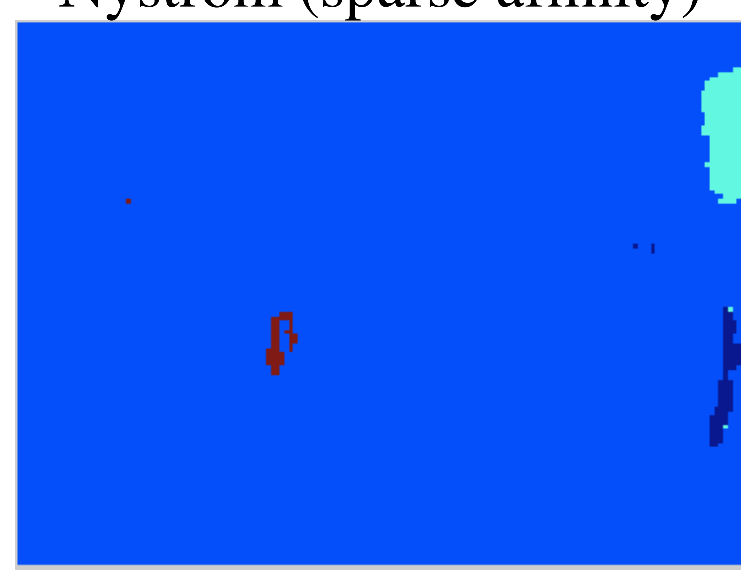
LLL



Nystrom (full affinity)



Nystrom (sparse affinity)



Conclusions

- Spectral clustering is a useful framework for clustering data, however, it does not scale well due to a eigendecomposition of a large NxN affinity matrix.
- Local Linear Landmarks provides a simple approximation algorithm that constructs smaller LxL affinities by incorporating local connection between *all* the affinities.
- The algorithm robustly shows >10x speedup with error within 1% of the exact method.
- It can be easily extended to a variety of useful settings:
  - constrained spectral clustering,
  - affinity aggregation,
  - motion segmentation,
  - etc.