Diffusion maps, spectral clustering and reaction coordinates of dynamical systems

Boaz Nadler
Stéphane Lafon
Ronald R. Coifman
Ioannis G. Kevrekidis

Shane McQuarrie

CSE 392: SCIENTIF COMP MACH/DEEP LRN
15 November 2019

Outline

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Appl. Comput. Harmon. Anal. 21 (2006) 113–127

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

Diffusion maps, spectral clustering and reaction coordinates of dynamical systems

Boaz Nadler [a,*], Stéphane Lafon [a,1], Ronald R. Coifman [a], Ioannis G. Kevrekidis [b]

[a] *Department of Mathematics, Yale University, New Haven, CT 06520, USA*
[b] *Chemical Engineering and PACM, Princeton University, Princeton, NJ 08544, USA*

Received 28 October 2004; revised 10 February 2005; accepted 29 July 2005

Available online 9 June 2006

Communicated by the Editors

Overview of Paper

- **Statistics**: published in 2006, ~ 500 citations (companion to a paper with $\sim 2,100$ citations)
- **Problem statement**: high-dimensional data coming from large-scale dynamical systems often have low-dimensional structure, but learning this structure requires identifying slow variables and dynamically meaningful reaction coordinates.
- **Solution**: define a family of *diffusion maps* that embed the data in a low-dimensional space with a desired time scale.

Markov Chains

$$\begin{array}{ccc} & \text{hot tomorrow} & \text{cold tomorrow} \\ \text{hot today} & \left[\begin{array}{c} 0.7 \\ 0.6 \end{array}\right. & \left.\begin{array}{c} 0.3 \\ 0.4 \end{array}\right] = M \\ \text{cold today} & & \end{array}$$

Markov Chains

- Let $\mathbf{x} \in \mathbb{R}^n$ be a state distribution vector, i.e., x_j is the probability of being in state j at the current time.
- $(M^\mathsf{T}\mathbf{x})_j$ is the probability of being in state j after one transition
- $((M^t)^\mathsf{T}\mathbf{x})_j$ is the probability of being in state j after t transitions.

Key observation: M^t describes the dynamics with the time scale t.

Key property: M has a steady-state distribution $\boldsymbol{\psi} \succ \mathbf{0}$, i.e., $M^\mathsf{T}\boldsymbol{\psi} = \boldsymbol{\psi}$.

Markov Chains

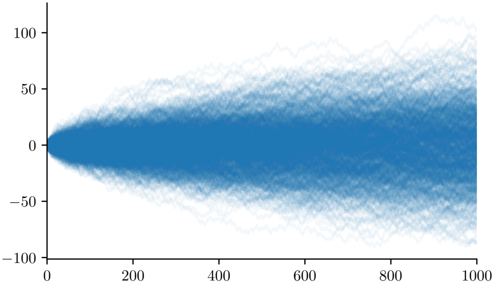
- Discrete case: P_{ij} is the probability of moving from state i to state j. Therefore,

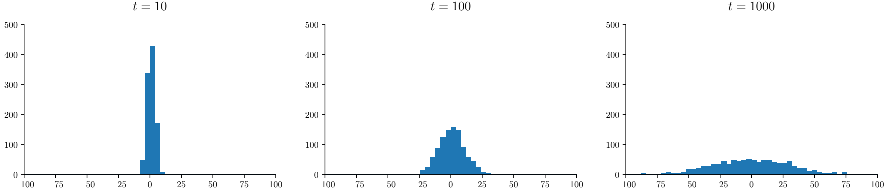
$$\sum_{j=1}^{N} P_{ij} = 1 \qquad \forall j = 1, 2, \ldots, N.$$

- Continuous case: $p(\mathbf{x}, \mathbf{y})$ is a probability density function describing the transition from $\mathbf{x}$ to $\mathbf{y}$, with

$$\int_X p(\mathbf{x}, \mathbf{y}) d\mu(\mathbf{y}) = 1 \qquad \forall \mathbf{x} \in X.$$

Random Walks and Diffusion





From Data to Markov Chain

Let $X = \{\mathbf{x}\}_{i=1}^{N} \subset \mathbb{R}^n$ be the data, and let $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ be a *kernel function* satisfying

$$\begin{aligned} k(\mathbf{x}, \mathbf{y}) &= k(\mathbf{y}, \mathbf{x}) \\ k(\mathbf{x}, \mathbf{y}) &\geq 0 \\ k(\mathbf{x}, \mathbf{x}) &= 0 \qquad \text{(usually)} \end{aligned}$$

Example: Gaussian kernel, $k_\varepsilon(\mathbf{x}, \mathbf{y}) = e^{-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{\varepsilon}}$

Define $M \in \mathbb{R}^{N \times N}$ by

$$M_{ij} = \frac{k(\mathbf{x}_i, \mathbf{x}_j)}{\sum_{\ell=1}^{N} k(\mathbf{x}_i, \mathbf{x}_\ell)}.$$

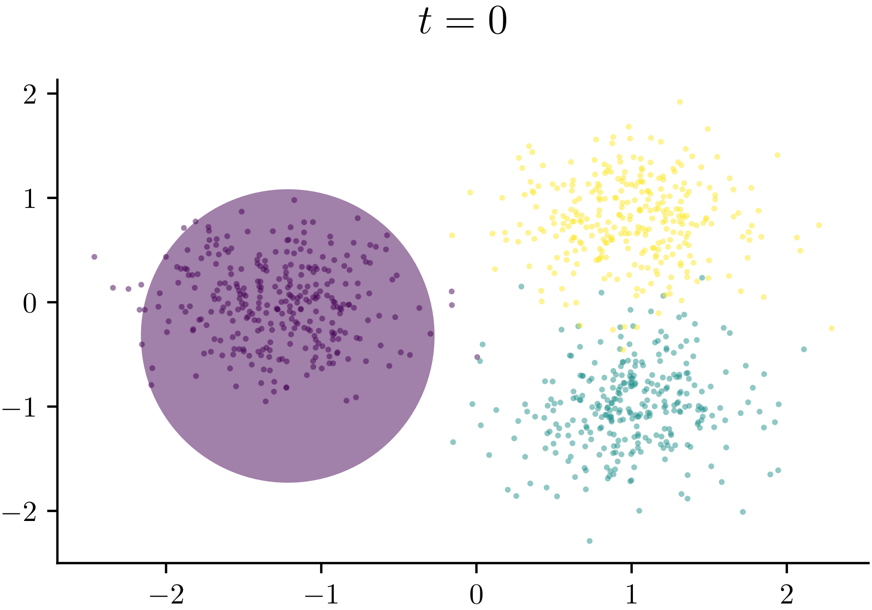
Then M_{ij} is the weighted distance from $\mathbf{x}_i$ to $\mathbf{x}_j$.

Diffusion on Discrete Markov Chains

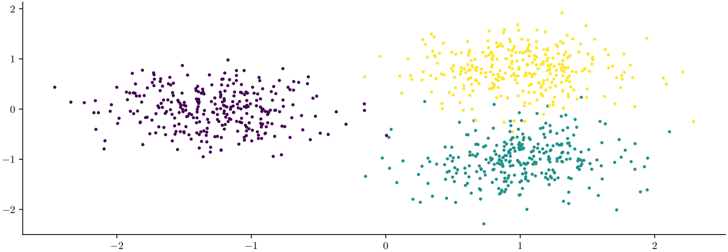
https://tinyurl.com/y5u3jhte

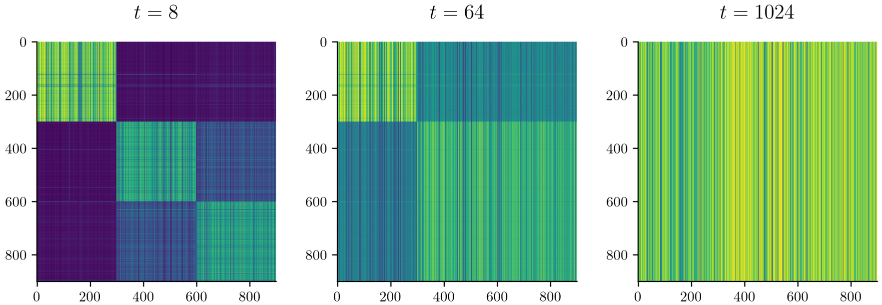
```
import numpy as np

def gaussian_kernel(x, y, eps=.7):
    return np.exp(-np.sum((x - y)**2) / eps)

def markov_transition_matrix(X, k, **kwargs):
    m,n = X.shape
    G = np.zeros((m,m))
    for i in range(m):
        for j in range(i+1,m):
            G[i,j] = k(X[i], X[j], **kwargs)
    G = G + G.T
    return G / G.sum(axis=1).reshape((-1,1))
```

Diffusion on Discrete Markov Chains



Diffusion on Discrete Markov Chains





Diffusion Maps

Definition (Diffusion Distance)

Let M be the row-stochastic Markov transition matrix defined previously. Let $\{\lambda_j\}_{j=0}^{N-1}$ be the nondecreasing eigenvalues of M, with $\lambda_0 = 1$, and let $\{\boldsymbol{\psi}_j\}_{j=0}^{N-1}$ be the corresponding right eigenvectors. The *diffusion distance* at time t is given by

$$\begin{aligned} D_t^2(\mathbf{x}, \mathbf{y}) :&= \|p(\mathbf{z}, t|\mathbf{x}) - p(\mathbf{z}, t|\mathbf{y})\|_w^2 \\ &= \sum_{\mathbf{z}} \big(p(\mathbf{z}, t|\mathbf{x}) - p(\mathbf{z}, t|\mathbf{y})\big)^2 w(\mathbf{z}). \end{aligned}$$

Choosing $w(\mathbf{z}) = 1/p(\mathbf{z})$, we also have

$$D_t^2(\mathbf{x}, \mathbf{y}) = \sum_j \lambda_j^{2t} \big(\boldsymbol{\psi}_j(\mathbf{x}) - \boldsymbol{\psi}_j(\mathbf{y})\big)^2.$$

Wait, what?

Bad Notation

If the data set is $\{\mathbf{x}_i\}_{i=1^N}$, then $\boldsymbol{\psi}_j(\mathbf{x}_i) = (\boldsymbol{\psi}_j)_i$.

(We can't apply this mapping to arbitrary data.)

Diffusion Maps

Definition (Diffusion Map)

Let $\{(\lambda_j, \boldsymbol{\psi}_j)\}_{j=0}^{N-1}$, be the right eigenpairs of M as before. For some $k < N$, we define the family of *diffusion maps* $\Psi_t : X \to \mathbb{R}^{k+1}$ parametrized by t:

$$\Psi_t(\mathbf{x}) = \begin{bmatrix} \lambda_0^t \boldsymbol{\psi}_0(\mathbf{x}) \\ \lambda_1^t \boldsymbol{\psi}_1(\mathbf{x}) \\ \vdots \\ \lambda_k^t \boldsymbol{\psi}_k(\mathbf{x}) \end{bmatrix}, \qquad \text{i.e.,} \qquad \Psi_t(\mathbf{x}_i) = \begin{bmatrix} \lambda_0^t (\boldsymbol{\psi}_0)_i \\ \lambda_1^t (\boldsymbol{\psi}_1)_i \\ \vdots \\ \lambda_k^t (\boldsymbol{\psi}_k)_i \end{bmatrix}.$$

Note that $\|\Psi_t(\mathbf{x}) - \Psi_t(\mathbf{y})\| = D_t(\mathbf{x}, \mathbf{y})$ (the diffusion distance).

Diffusion Maps

```python
from scipy import linalg as la

class DiffusionMap:
    def __init__(self, X, k, kernel=gaussian_kernel, **kwargs):
        self.M = markov_transition_matrix(X, kernel, **kwargs)
        vals, vecs = la.eig(self.M)
        index = vals.real.argsort()[::-1][:s]
        self.vals = np.minimum(vals[index].real, 1)
        self.vecs = vecs[:,index].real

    def __call__(self, t):
        return self.vals**t * self.vecs
```

Special Cases

- $\mathbf{x}_i \sim p(\mathbf{x})$ with $p(\mathbf{x}) = e^{-U(\mathbf{x})}$ for some potential $U(\mathbf{x})$.
- $\mathbf{x}_i$ is a state of the stochastic equation $\dot{\mathbf{x}} = -\nabla U(\mathbf{x}) + \sqrt{2}\dot{\mathbf{w}}$.

Main result: Depending on how the transition matrix M is normalized, the eigenvectors of M approach the eigenfunctions of an appropriate diffusion operator as the number of samples increases.
(But this doesn't matter for implementation.)

Summary

Given data $X = \{\mathbf{x}_i\}_{i=1}^N \subset \mathbb{R}^n$ and a kernel $k : X \times X \to \mathbb{R}$,

1. Calculate the Markov transition matrix M with
$$M_{ij} = \frac{k(\mathbf{x}_i, \mathbf{x}_j)}{\sum_{\ell=1}^N k(\mathbf{x}_i, \mathbf{x}_\ell)}.$$
2. Compute the eigenvalue-eigenvector pairs $\{(\lambda_j, \boldsymbol{\psi}_j)\}_{j=0}^{N-1}$ of M
3. Define the diffusion map (parametrized by time)
$$\Psi_t(\mathbf{x}_i) = \begin{bmatrix} \lambda_0^t(\boldsymbol{\psi}_0)_i & \lambda_1^t(\boldsymbol{\psi}_1)_i & \cdots & \lambda_k^t(\boldsymbol{\psi}_k)_i \end{bmatrix}^\mathsf{T}.$$

Cost: $\mathcal{O}(N^2) + \mathcal{O}(kN^2) + \mathcal{O}(Nk) = \mathcal{O}(kN^2)$
(cost of computing k eigenpairs).

Judgements

Pros

- Correlations between all data points are treated simultaneously
- Gracefully exposes the low-dimensional geometry of a dataset
- Provides interpretation for various timescales
- Relatively cheap (eigenvalue problem)
- Robust to noise

Cons

- Diffusion map only defined on X, not $\mathbb{R}^n$, so it's hard to deal with new points; projection is possible but hard
- Sometimes the parametrization by time doesn't pay off (but this can also be informative)
- Sometimes not quite so robust to noise (?)

Numerical Examples

https://tinyurl.com/y5u3jhte

References

Ronald R Coifman and Stéphane Lafon.
Diffusion maps.
Applied and computational harmonic analysis, 21(1):5–30, 2006.

Diffusion map.
https://en.wikipedia.org/wiki/Diffusion_map.
Accessed: 2019-11-08.

Boaz Nadler, Stéphane Lafon, Ronald R Coifman, and Ioannis G Kevrekidis.
Diffusion maps, spectral clustering and reaction coordinates of dynamical systems.
Applied and Computational Harmonic Analysis, 21(1):113–127, 2006.