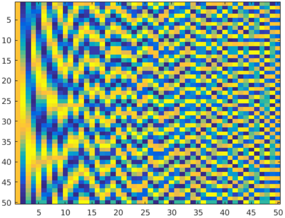
Applications of Diffusion Maps

Geometric Flows, Resampling and Dimensionality Reduction



Orton Babb   Aneesh Malhotra

Advised by Dr. Tyrus Berry

MEGL Symposium, Spring 2018

Outline

Outline

Geometric Flows

- Sampling from a Manifold $\mathcal{M}$
- Constructing the Normalized Discrete Laplacian on a Manifold
- Solving the Heat Equation on a Manifold
- Running a Geometric Heat Flow

Resampling

- Recovering Points from the Normalized Kernel
- Normalizing the Distance to the K-Nearest Neighbors
- Nyström extensions

Dimensionality Reduction

- Gradient Descent Reconstruction

Laplacian on Manifolds

- Laplacian in Euclidean space: $\Delta f = \sum_{i=1}^{n} \frac{\partial^2 f}{\partial x_i^2}$
- Laplacian on a circle: $\Delta f = \frac{d^2 f}{d\theta^2}$
- In generaly, determined by the Riemannian metric, g:

$$\Delta f = \frac{1}{\sqrt{|g|}} \sum_{i,j=1}^{n} \frac{\partial}{\partial x_i} \left(g^{ij} \sqrt{|g|} \frac{\partial f}{\partial x_j} \right)$$

- Laplacian is hard to construct but very useful
- Eigenfunctions are a basis for function space on the manifold
- Defines the heat equation on the manifold $\frac{\partial f}{\partial t} = \Delta f$

Estimating the Laplacian with Diffusion Maps

- Given a set of of N data living in $\mathbb{R}^n$, we can define a kernel matrix to be

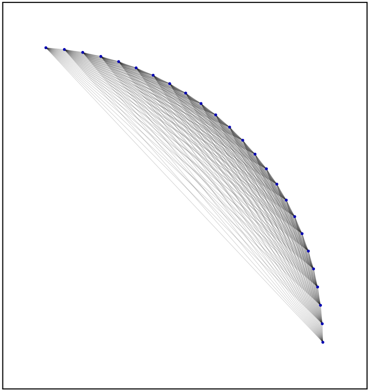
$$K_{ij} = \exp \frac{-||x_i - x_j||^2}{\epsilon}.$$

- If we let D be the diagonal matrix whose entrees are the row sums of K, the graph laplacian is given by

$$L = D - K$$

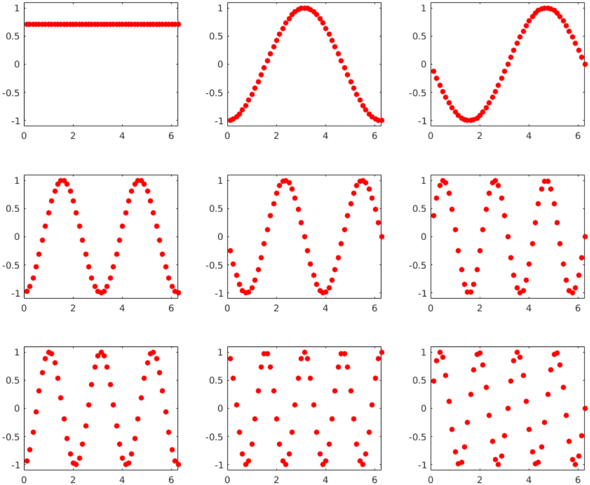
- Diffusion Maps paper shows that L approximates Δ
- In the limit of infinite data, $L \to \Delta$

Example S^1: Kernel Matrix Represents a Weighted Graph



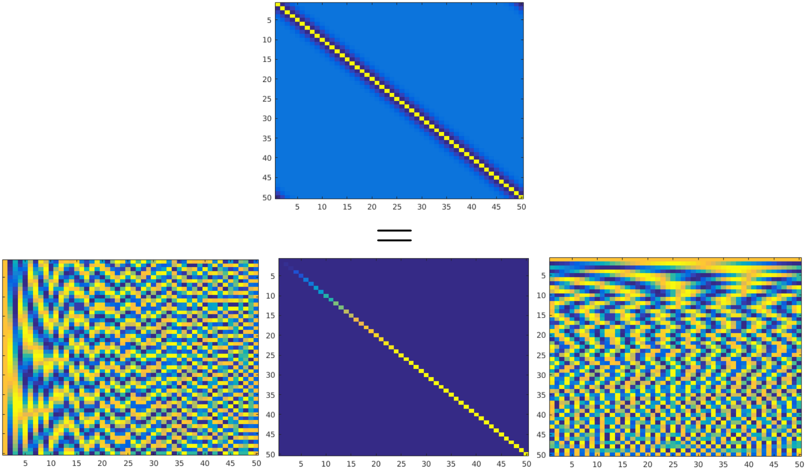
Visualization of a weighted graph with points sampled from a segment on the circle

Example S^1: Eigenfunctions of the Laplacian as Eigenvectors of L



Scatter plots of $\{(\theta_i, \phi_i)\}_{i=1}^{N}$

Example S^1: Eigenvector Decomposition $L = U\Lambda U^\top$



$U \qquad \Lambda \qquad U^\top$

Solving the Heat Equation on a Manifold

- In the limit of infinite data, $L \to \Delta$
- With an expression for the laplacian, we can solve the heat equation on a manifold, which is discretized by a set of data points.
- Heat equation can be expressed as:

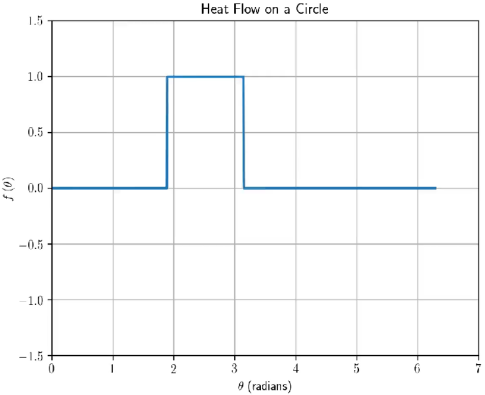
$$\frac{\partial u(x,t)}{\partial t} = -\Delta u(x,t).$$

- Discrete solution $\vec{u}(t)_i = u(x_i, t)$:

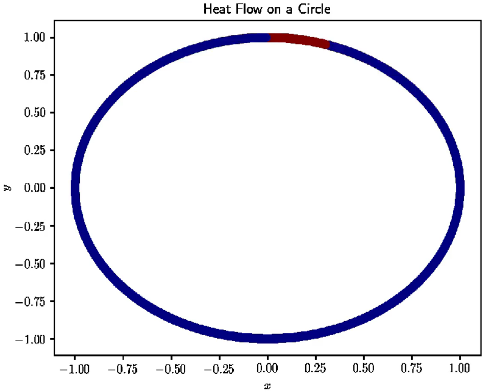
$$\vec{u}(t+\tau) \approx \vec{u}(t) - \tau L \vec{u}(t)$$

- We replaced the time derivative and Δ with discretizations.

Example S^1: Heat flow as function of θ



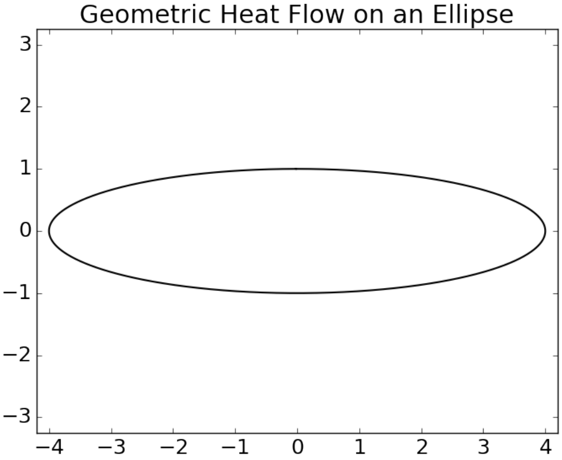
Example S^1: Heat flow shown on the data



Geometric Flows

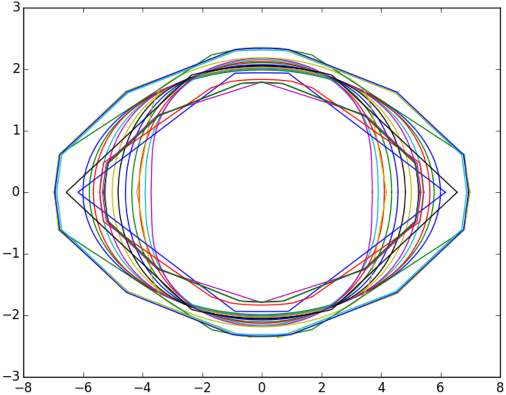
- **Geometric Flow:** Apply the heat flow to the manifold
- Each coordinate of our embedding is a function on the manifold $F = (f_1, f_2, ..., f_n) : \mathcal{M} \to \mathbb{R}^n$
- Apply heat flow to each coordinate independently
- As the embedding evolves, the manifold changes!
- As the manifold changes, the Laplacian changes!
- Must recompute the Laplacian after each small time step

Geometric Flow on Ellipse



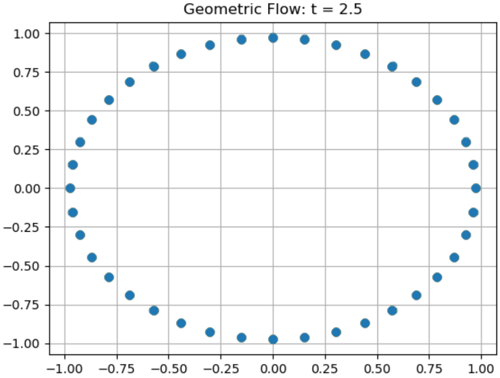
Issues

- As the flow progresses, we eventually get numerical singularities in the kernel matrix



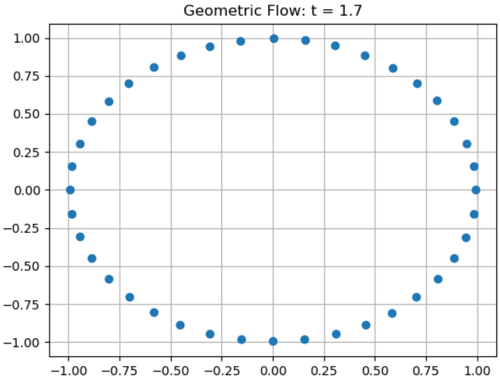
Issues

- The circle should be a steady state solution of the flow



Issues

- The circle should be a steady state solution of the flow



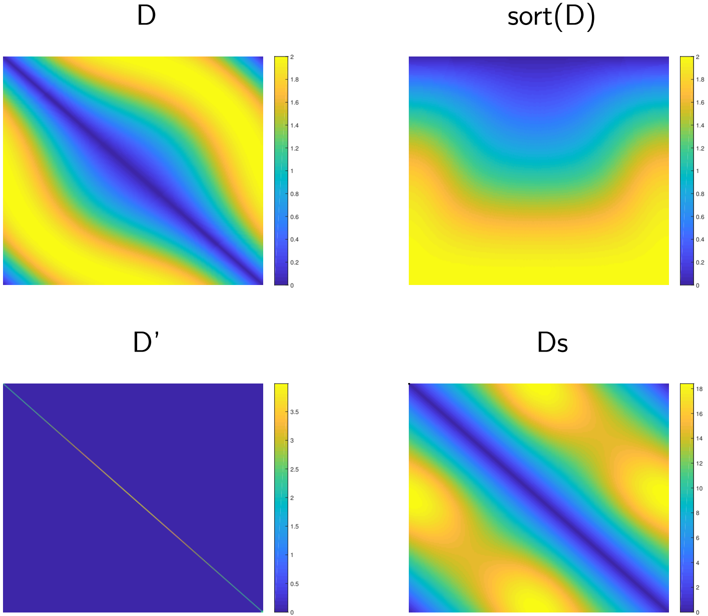
- After many steps, the symmetry breaks in the data points
- We think this is the cause of the instability
- Solution is to 'resample' the points to maintain symmetry

Outline

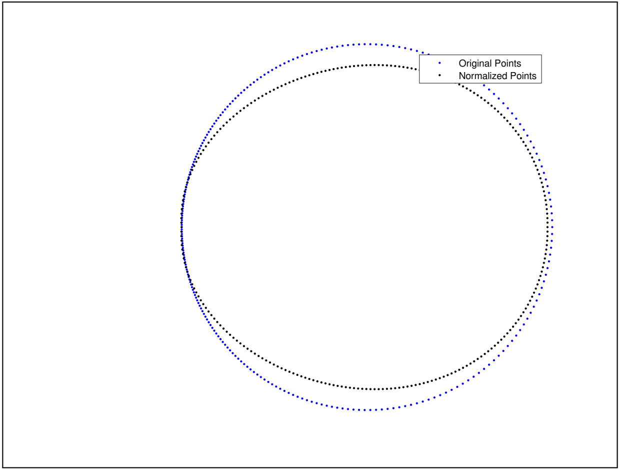
Recovering Points from the Normalized Kernel

1. Build the distance/kernel matrix
2. Perform normaliziation
3. Apply inverse to retrieve distance matrix from K_X
4. Perform MDS to recover center version of the points

K-Nearest Neighbors Normalization



Normalizing Distorts the Shape



Normalized data set of 250 points

Extending the Eigenfunction Basis I

- The Laplacian $\Delta_{\mathcal{M}}$ can be recovered from data
- It has eigenfunctions ϕ_i such that $\Delta_{\mathcal{M}}\phi_i = \lambda_i \phi_i$
- Suppose there is a function, $f : \mathcal{M} \to \mathbb{R}$ on this data
- Since eigenfunctions form a basis for $L^2(\mathcal{M})$, we can write f as

$$f(z) = \sum_{i=1}^{\infty} \underbrace{\langle f, \phi_i \rangle}_{\hat{f}} \phi_i(z)$$

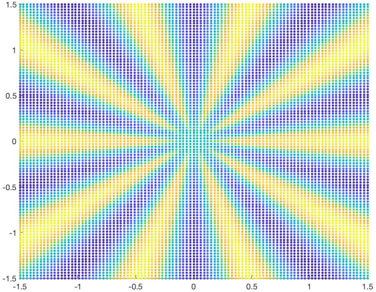
Extending the Eigenfunction Basis II

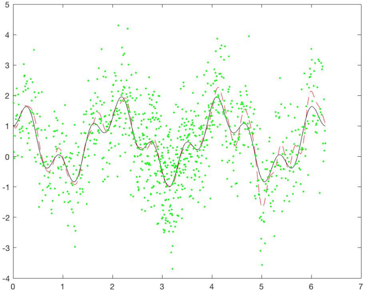
- Since we can describe F in terms of its coordinate functions as $F = (f_1, f_2, ..., f_n)$ with $f_k : \mathcal{M} \to \mathbb{R}$
- There are $c_{kj} = \langle f_k, \phi_j \rangle \cong \mathbf{f}^t \tilde{D} \vec{\phi}_i$ (where c is an $n \times N$ matrix)
- c may be computed in full by $c = X^t \tilde{D} \Phi$
- It remains to show how $\phi_i(z)$ can be computed for $z \neq z_i$ for any i

Nyström Extension

- Reconstructing the eigenfunctions of Laplacian along the entire domain.

$$\phi_j(z) \approx \frac{1}{\lambda_j} \sum \tilde{K}(z, z_j)\phi_j(z_i)$$





Outline

M E G L

Reconstructing in Lower Dimensions

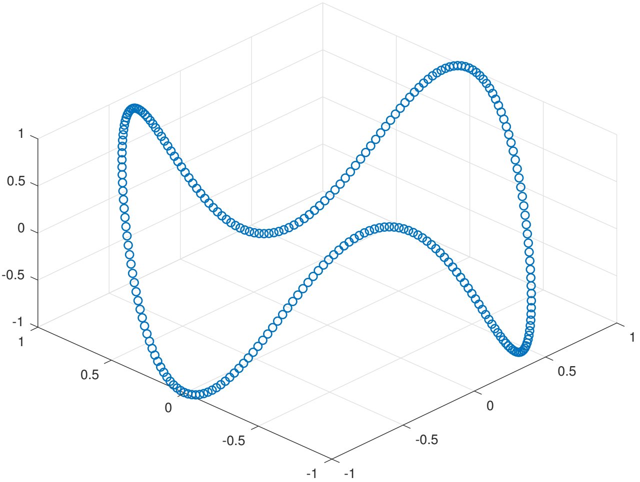
If we can reconstruct the embedding with $X = CU^t$ why not aim for an even better reconstruction that preserves the K_X?

$$\mathcal{M} \to X \quad \longrightarrow \qquad \begin{array}{c} K_X = U \Lambda U^t \\ \downarrow \\ \tilde{X} = \tilde{C} U^t \\ \tilde{C} = \mathrm{argmin}_C \| K_X - K_{\tilde{X}}(C) \|_{fro} \end{array}$$

Gradient Descent for Minimal Embedding

Example: Pringle Chip!

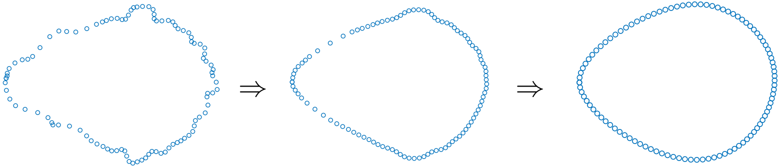
$x_i = F(\theta_i) = [cos(\theta_i), sin(\theta_i), cos(k\theta_i), sin(k\theta_i)]$ for some $k \in \mathbb{Z}$



Gradient Descent for Minimal Embedding I

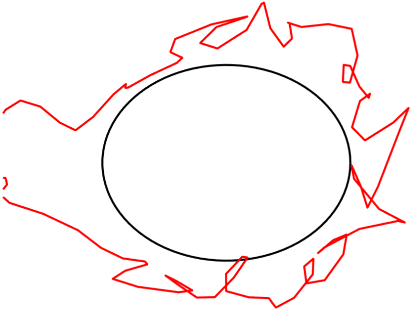
1. Start with random guess for C
2. Define reconstructErr(C) $= \|K_X - K_{\tilde{X}}(C)\|_{fro}$
3. Do $C := C - \eta * \nabla reconstructErr(C)$ until $\|C' - C\| < TOL$

Running the Gradient Descent



Progressive improvements in reconstruction for $m = 2$

Running the Gradient Descent

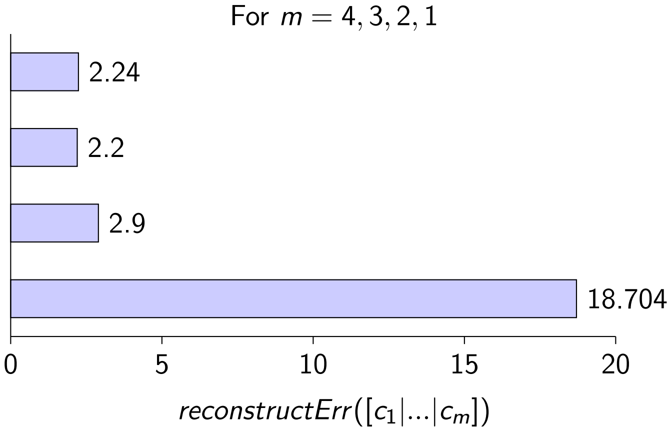


Black = Optimal Embedding
Red = Gradient Descent from Random Initial Embedding

Gradient Descent for Minimal Embedding II

1. Solve $X = CU^t$ for $C = [c_1|c_2|...|c_m]$ as initial guess.
2. Define reconstructErr(C) $= \|K_X - K_{\tilde{X}}(C)\|_{fro}$
3. Do $C := C - \eta * \nabla reconstructErr(C)$ until $\|C' - C\| < TOL$
4. Define reconstructErr(C) $= \|K_X - K_{\tilde{X}}(C)\|_{fro}\|c_m\|$
5. Do $C := C - \eta * \nabla reconstructErr(C)$ until $\|C' - C\| < TOL$
6. Throw away the end column of C and repeat from #2

Gradient Descent for Minimal Embedding II



Summary

- The geometric heat flow failed due to breakdown of symmetry.
- The resampling methods attempt to resolve these issues.
- The reconstruction of the data, could be used and this also helps with dimensionality reduction.
- Outlook
  - Dimensionality reduction could be improved for efficiency and preventing loss of information.

For Further Reading I

R. Coifman and S. Lafon.
Diffusion maps.
Applied and Computational Harmonic Analysis, 21(1):5–30, 2006.