On Laplacian Eigenmaps for Dimensionality Reduction

Dr. Juan Orduz

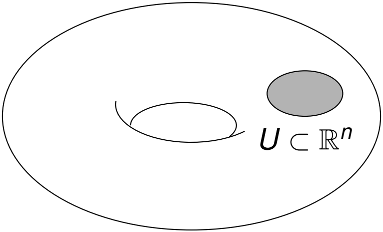
PyData Berlin 2018

Overview

Can One Hear the Shape of a Drum?

[Kac66]

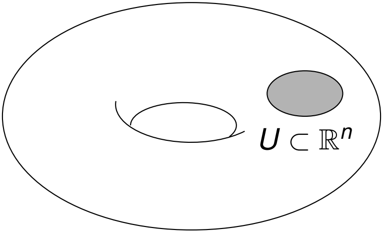
A **differentiable manifold** is a type of manifold that is locally similar enough to a linear space to allow one to do calculus. A (Riemannian) metric g allow us to measure distances.



Can One Hear the Shape of a Drum?

[Kac66]

A **differentiable manifold** is a type of manifold that is locally similar enough to a linear space to allow one to do calculus. A (Riemannian) metric g allow us to measure distances.

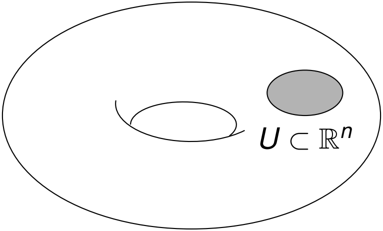


We can consider the **Laplacian** $L : C^\infty(M) \longrightarrow C^\infty(M)$ and its **spectrum** $\text{spec}(L) = \{\lambda_0, \lambda_1, \cdots, \lambda_k, \cdots \longrightarrow \infty\}$.

Can One Hear the Shape of a Drum?

[Kac66]

A **differentiable manifold** is a type of manifold that is locally similar enough to a linear space to allow one to do calculus. A (Riemannian) metric g allow us to measure distances.



We can consider the **Laplacian** $L : C^{\infty}(M) \longrightarrow C^{\infty}(M)$ and its **spectrum** $\text{spec}(L) = \{\lambda_0, \lambda_1, \cdots, \lambda_k, \cdots \longrightarrow \infty\}$.

- If we are given spec(L) we can infer the dimension of M, its volume and its total scalar curvature.

Spectral Geometry for Dimensionality Reduction?

Let us assume we have data points $x_1, \cdots, x_k \in \mathbb{R}^N$ which lie on an unknown submanifold $M \subset \mathbb{R}^N$.

Key Observation

- Eigenfunctions of L on M can be used to define lower dimensional embeddings.

Idea ([BN03])

- Model M by constructing a graph $G = (V, E)$ where close data points are connected by edges.

Spectral Geometry for Dimensionality Reduction?

Let us assume we have data points $x_1, \cdots, x_k \in \mathbb{R}^N$ which lie on an <u>unknown</u> submanifold $M \subset \mathbb{R}^N$.

Key Observation

- Eigenfunctions of L on M can be used to define lower dimensional embeddings.

Idea ([BN03])

- Model M by constructing a graph $G = (V, E)$ where close data points are connected by edges.
- Construct the graph Laplacian L on G.

Spectral Geometry for Dimensionality Reduction?

Let us assume we have data points $x_1, \cdots, x_k \in \mathbb{R}^N$ which lie on an unknown submanifold $M \subset \mathbb{R}^N$.

Key Observation

- Eigenfunctions of L on M can be used to define lower dimensional embeddings.

Idea ([BN03])

- Model M by constructing a graph $G = (V, E)$ where close data points are connected by edges.
- Construct the graph Laplacian L on G.
- Compute spec(L) and the corresponding eigenfunctions.

Spectral Geometry for Dimensionality Reduction?

Let us assume we have data points $x_1, \cdots, x_k \in \mathbb{R}^N$ which lie on an unknown submanifold $M \subset \mathbb{R}^N$.

Key Observation

- Eigenfunctions of L on M can be used to define lower dimensional embeddings.

Idea ([BN03])

- Model M by constructing a graph $G = (V, E)$ where close data points are connected by edges.
- Construct the graph Laplacian L on G.
- Compute spec(L) and the corresponding eigenfunctions.
- Use these eigenfunctions to construct an embedding $F : V \longrightarrow \mathbb{R}^m$ for $m < N$.

The Spectral Theorem

Let $A \in M_{n\times n}(\mathbb{R})$ be a symmetric matrix, i.e. $A = A^\dagger$.

The Spectral Theorem

Let $A \in M_{n\times n}(\mathbb{R})$ be a symmetric matrix, i.e. $A = A^\dagger$.

Recall

- $\lambda \in \mathbb{C}$ is an **eigenvalue** for A with **eigenvector** $f \in \mathbb{R}^n$, $f \neq 0$, if
$$Af = \lambda f.$$
- A set of vectors $\mathcal{B} = \{f_1, f_2, \cdots, f_n\}$ is a **basis** for $\mathbb{R}^n$ if:
  - They are linearly independent.
  - They generate $\mathbb{R}^n$.
- $\mathcal{B}$ is said to be an **orthonormal** basis if $\langle f_i, f_j\rangle = \delta_{ij}$.

The Spectral Theorem

Let $A \in M_{n\times n}(\mathbb{R})$ be a symmetric matrix, i.e. $A = A^\dagger$.

Recall

- $\lambda \in \mathbb{C}$ is an **eigenvalue** for A with **eigenvector** $f \in \mathbb{R}^n$, $f \neq 0$, if
$$Af = \lambda f.$$
- A set of vectors $\mathcal{B} = \{f_1, f_2, \cdots, f_n\}$ is a **basis** for $\mathbb{R}^n$ if:
  - They are linearly independent.
  - They generate $\mathbb{R}^n$.
- $\mathcal{B}$ is said to be an **orthonormal** basis if $\langle f_i, f_j\rangle = \delta_{ij}$.

Spectral Theorem

There exists an orthonormal basis of $\mathbb{R}^n$ consisting of eigenvectors of A. Each eigenvalue is real.

Min(Max)imizing Properties of Eigenvalues

Let $A \in M_n(\mathbb{R})$ be a symmetric matrix with spectral decomposition $\lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_n$.

For later purposes, we would like to find

$$\underset{||f||=1}{\arg\max} \langle Af, f \rangle.$$

PyData Berlin 2018

Min(Max)imizing Properties of Eigenvalues

Let $A \in M_n(\mathbb{R})$ be a symmetric matrix with spectral decomposition $\lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_n$.

For later purposes, we would like to find

$$\underset{||f||=1}{\arg\max} \langle Af, f \rangle.$$

- Define the associated Lagrange optimization problem

$$\mathcal{L}(f, \lambda) = \langle Af, f \rangle - \lambda(||f||^2 - 1).$$

Min(Max)imizing Properties of Eigenvalues

Let $A \in M_n(\mathbb{R})$ be a symmetric matrix with spectral decomposition $\lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_n$.

For later purposes, we would like to find

$$\operatorname*{arg\,max}_{||f||=1} \langle Af, f \rangle.$$

- Define the associated Lagrange optimization problem

$$\mathcal{L}(f, \lambda) = \langle Af, f \rangle - \lambda(||f||^2 - 1).$$

- Take the derivative with respect to f

$$\frac{\partial}{\partial f}\mathcal{L}(f, \lambda) = 2(Af - \lambda f) \stackrel{!}{=} 0.$$

Min(Max)imizing Properties of Eigenvalues

Let $A \in M_n(\mathbb{R})$ be a symmetric matrix with spectral decomposition $\lambda_0 \leq \lambda_1 \leq \cdots \leq \lambda_n$.

For later purposes, we would like to find

$$\underset{||f||=1}{\arg\max} \langle Af, f \rangle.$$

- Define the associated Lagrange optimization problem

$$\mathcal{L}(f, \lambda) = \langle Af, f \rangle - \lambda(||f||^2 - 1).$$

- Take the derivative with respect to f

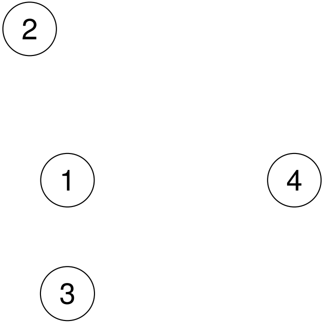
$$\frac{\partial}{\partial f} \mathcal{L}(f, \lambda) = 2(Af - \lambda f) \overset{!}{=} 0.$$

- Hence,

$$\underset{||f||=1}{\arg\max} \langle Af, f \rangle = f_n \quad \text{and} \quad \underset{||f||=1}{\arg\min} \langle Af, f \rangle = f_0.$$

Step 0: Understand the Problem

Consider the problem of mapping these points to a line so that close points stay as together as possible.

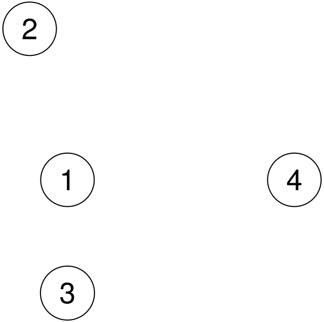


Step 1: From Data to Adjacency Graph

- Define a distance function: first nearest neighbour.

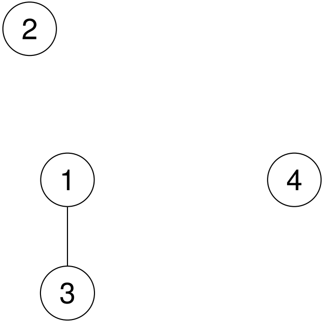
Step 1: From Data to Adjacency Graph

- Define a distance function: first nearest neighbour.
- For each node, attach an edge for close points.



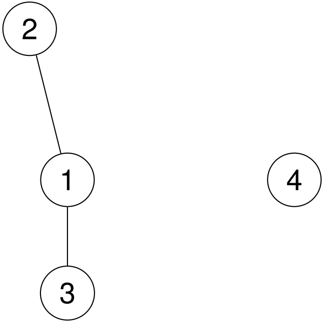
Step 1: From Data to Adjacency Graph

- Define a distance function: first nearest neighbour.
- For each node, attach an edge for close points.



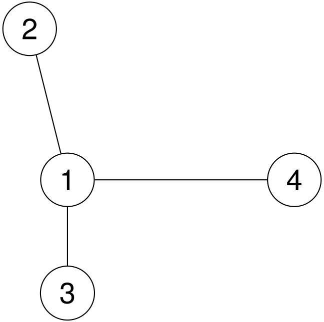
Step 1: From Data to Adjacency Graph

- Define a distance function: first nearest neighbour.
- For each node, attach an edge for close points.

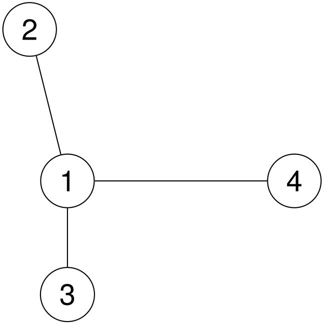


Step 1: From Data to Adjacency Graph

- Define a distance function: first nearest neighbour.
- For each node, attach an edge for close points.



Step 2: Construct the Adjacency and Degree Matrices



$$W = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \quad D = \begin{pmatrix} 3 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Step 3: Spectrum of the Graph Laplacian

- Construct the operator L defined by

$$L := D - W = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

- Consider the generalized eigenvalue problem

$$Lf = \lambda Df.$$

Equivalently, $D^{-1}Lf = \lambda f$.

Step 3: Spectrum of the Graph Laplacian

- Construct the operator L defined by

$$L := D - W = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

- Consider the generalized eigenvalue problem

$$Lf = \lambda Df.$$

  Equivalently, $D^{-1}Lf = \lambda f$.
- Eigenvalues: $\lambda_0 = 0, \lambda_1 = 1, \lambda_2 = 1, \lambda_3 = 2$.

Step 3: Spectrum of the Graph Laplacian

- Construct the operator L defined by

$$L := D - W = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

- Consider the generalized eigenvalue problem

$$Lf = \lambda Df.$$

  Equivalently, $D^{-1}Lf = \lambda f$.
- Eigenvalues: $\lambda_0 = 0, \lambda_1 = 1, \lambda_2 = 1, \lambda_3 = 2$.
- An eigenvector for $\lambda_1 = 1$ is $y := f_1 = (0, -3, 1, 2)$.

Step 3: Spectrum of the Graph Laplacian

- Construct the operator L defined by

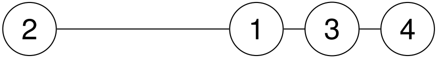
$$L := D - W = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

- Consider the generalized eigenvalue problem

$$Lf = \lambda Df.$$

Equivalently, $D^{-1}Lf = \lambda f$.

- Eigenvalues: $\lambda_0 = 0, \lambda_1 = 1, \lambda_2 = 1, \lambda_3 = 2$.
- An eigenvector for $\lambda_1 = 1$ is $y := f_1 = (0, -3, 1, 2)$.
- The vector $y : V \longrightarrow \mathbb{R}$ defines and embedding.



The Algorithm

Let $x_1, \cdots, x_k \in \mathbb{R}^N$.

1. **Construct a weighted graph** $G = (V, E)$ with k nodes, one for each point, and a set of edges connecting neighbouring points. **Select a distance function**:
   - (Euclidean Distance) Let $\varepsilon > 0$. We connect and edge between i and j if $||x_i - x_j||^2 < \varepsilon$.
   - n nearest neighbours.

The Algorithm

Let $x_1, \cdots, x_k \in \mathbb{R}^N$.

1. **Construct a weighted graph** $G = (V, E)$ with k nodes, one for each point, and a set of edges connecting neighbouring points. **Select a distance function**:
   - (Euclidean Distance) Let $\varepsilon > 0$. We connect and edge between i and j if $||x_i - x_j||^2 < \varepsilon$.
   - n nearest neighbours.
2. **Choose Weights.** If nodes i and j are connected, put
   - $W_{ij} = 1$.
   - (Heat Kernel) $W_{ij} := e^{-\frac{||x_i - x_j||^2}{t}}$ for some $t > 0$.

The Algorithm

Let $x_1, \cdots, x_k \in \mathbb{R}^N$.

1. **Construct a weighted graph** $G = (V, E)$ with k nodes, one for each point, and a set of edges connecting neighbouring points. **Select a distance function**:
   - (Euclidean Distance) Let $\varepsilon > 0$. We connect and edge between i and j if $||x_i - x_j||^2 < \varepsilon$.
   - n nearest neighbours.
2. **Choose Weights.** If nodes i and j are connected, put
   - $W_{ij} = 1$.
   - (Heat Kernel) $W_{ij} := e^{-\frac{||x_i - x_j||^2}{t}}$ for some $t > 0$.
3. Assume G is connected. **Compute the eigenvalues** of the generalized eigenvector problem $Lf = \lambda Df$, where
   - D is the diagonal weight matrix, $D_{ii} = \sum_{j=1}^{k} W_{ij}$.
   - $L := D - W$ is the graph Laplacian.

The Algorithm

Let $x_1, \cdots, x_k \in \mathbb{R}^N$.

1. **Construct a weighted graph** $G = (V, E)$ with k nodes, one for each point, and a set of edges connecting neighbouring points. **Select a distance function**:
   - (Euclidean Distance) Let $\varepsilon > 0$. We connect and edge between i and j if $||x_i - x_j||^2 < \varepsilon$.
   - n nearest neighbours.
2. **Choose Weights**. If nodes i and j are connected, put
   - $W_{ij} = 1$.
   - (Heat Kernel) $W_{ij} := e^{-\frac{||x_i - x_j||^2}{t}}$ for some $t > 0$.
3. Assume G is connected. **Compute the eigenvalues** of the generalized eigenvector problem $Lf = \lambda Df$, where
   - D is the diagonal weight matrix, $D_{ii} = \sum_{j=1}^{k} W_{ij}$.
   - $L := D - W$ is the graph Laplacian.
4. **Construct Embedding**. Let $f_0, f_1, \cdots, f_{k-1}$ be the corresponding eigenvectors ordered according to their eigenvalues ($\lambda_0 = 0$). For $m < N$, set

$$F(i) := (f_1(i), \cdots, f_m(i)).$$

Why does it work?

$m = 1$

Assume you have constructed the weighted graph $G = (V, E)$.
We want to construct an embedding $F : V \longrightarrow \mathbb{R}$.

<u>Hint:</u> Minimize

$$J(y) := \sum_{i,j=1}^{k} (y_i - y_j)^2 W_{ij} \stackrel{*}{=} 2y^\dagger L y.$$

Why does it work?

$m = 1$

Assume you have constructed the weighted graph $G = (V, E)$.
We want to construct an embedding $F : V \longrightarrow \mathbb{R}$.

Hint: Minimize

$$J(y) := \sum_{i,j=1}^{k} (y_i - y_j)^2 W_{ij} \overset{*}{=} 2y^\dagger L y.$$

Thus, the problem reduces to find

$$\underset{\substack{y^\dagger D y = 1 \\ y^\dagger D 1 = 0}}{\arg\min} \, y^\dagger L y = \underset{\substack{y^\dagger D y = 1 \\ y^\dagger D 1 = 0}}{\arg\min} \langle L y, y \rangle$$

- $y^\dagger D y = 1$ fixes the scale.
- $y^\dagger D 1 = 0$ eliminates the trivial solution $y = 1$.

Why does it work?

$m = 1$

Assume you have constructed the weighted graph $G = (V, E)$.
We want to construct an embedding $F : V \longrightarrow \mathbb{R}$.

Hint: Minimize

$$J(y) := \sum_{i,j=1}^{k} (y_i - y_j)^2 W_{ij} \stackrel{*}{=} 2y^\dagger L y.$$

Thus, the problem reduces to find

$$\underset{\substack{y^\dagger D y = 1 \\ y^\dagger D 1 = 0}}{\arg\min}\, y^\dagger L y = \underset{\substack{y^\dagger D y = 1 \\ y^\dagger D 1 = 0}}{\arg\min} \langle Ly, y \rangle$$

- $y^\dagger D y = 1$ fixes the scale.
- $y^\dagger D 1 = 0$ eliminates the trivial solution $y = 1$.

This translates to finding the minimum non-zero eigenvalue and eigenvector of

$$Ly = \lambda D y.$$

Why does it work?

$m > 1$ (Vectorize)

Assume you have constructed the weighted graph $G = (V, E)$.
We want to construct an embedding $F : V \longrightarrow \mathbb{R}^m$.

Hint: Minimize, for $Y = (y_1 \cdots y_m) \in M_{k\times m}(\mathbb{R})$,

$$J(Y) := \sum_{i,j=1}^{k} ||Y_i - Y_j||^2 W_{ij} = \mathrm{tr}(Y^\dagger LY).$$

Thus, the problem reduces to find

$$\underset{\mathrm{tr}(Y^\dagger DY = I)}{\arg\min} \ \mathrm{tr}(Y^\dagger LY)$$

This translates to finding the minimum non-zero eigenvalues and eigenvectors of

$$Lf = \lambda Dy.$$

Examples: Scikit-Learn

Let us go to a Jupyter notebook to see some examples.

The Laplacian

Second order differential operator $L : C_c^\infty(M) \longrightarrow C_c^\infty(M)$.

- For $M = \mathbb{R}^n$,

$$L = -\sum_{i=1}^{n} \frac{\partial^2}{\partial x_i^2}$$

- For (M, g) Riemannian manifold,

$$L = -\sum_{i=1}^{n}\sum_{j=1}^{n} g^{ij} \frac{\partial^2}{\partial x_i \partial x_j} + \text{lower order terms.}$$

The Laplacian

Second order differential operator $L : C_c^\infty(M) \longrightarrow C_c^\infty(M)$.

- For $M = \mathbb{R}^n$,

$$L = -\sum_{i=1}^{n} \frac{\partial^2}{\partial x_i^2}$$

- For (M, g) Riemannian manifold,

$$L = -\sum_{i=1}^{n}\sum_{j=1}^{n} g^{ij} \frac{\partial^2}{\partial x_i \partial x_j} + \text{lower order terms.}$$

Spectral Theorem ([Ros97])

L is symmetric with respect to the inner product in $C_c^\infty(M)$,

$$(f, g)_{L^2} = \int_M f(x)g(x)dx.$$

If M is compact, there exists an orthonormal basis of $L^2(M)$ consisting of eigenvectors of L. Each eigenvalue is real.

Embedding trough Eigenmaps

Let (M, g) be a compact Riemannian manifold and $f : M \longrightarrow \mathbb{R}$.

- If $x, z \in M$ are close, then

$$|f(x) - f(z)| \leq \text{dist}_M(x, z)||\nabla f|| + o(\text{dist}_M(x, z)).$$

Embedding trough Eigenmaps

Let (M, g) be a compact Riemannian manifold and $f : M \longrightarrow \mathbb{R}$.

- If $x, z \in M$ are close, then

$$|f(x) - f(z)| \leq \mathrm{dist}_M(x, z)||\nabla f|| + o(\mathrm{dist}_M(x, z)).$$

- We want a map that best preserves locality on average,

$$\underset{||f||_{L^2(M)}=1}{\arg\min} \int_M ||\nabla f||^2 dx. \tag{1}$$

Embedding trough Eigenmaps

Let (M, g) be a compact Riemannian manifold and $f : M \longrightarrow \mathbb{R}$.

- If $x, z \in M$ are close, then

$$|f(x) - f(z)| \leq \mathrm{dist}_M(x, z)||\nabla f|| + o(\mathrm{dist}_M(x, z)).$$

- We want a map that best preserves locality on average,

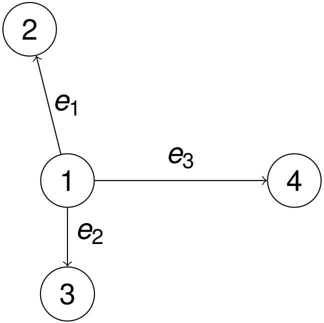
$$\underset{||f||_{L^2(M)}=1}{\arg\min} \int_M ||\nabla f||^2 dx. \tag{1}$$

- By Stokes' Theorem

$$\int_M ||\nabla f||^2 dx = \int_M (Lf) f dx = (Lf, f)_{L^2}.$$

- (1) must be an eigenvalue of the Laplacian.

The Graph Laplacian as a Differential Operator



$$\nabla = \begin{pmatrix} -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix} \quad \Rightarrow \quad \nabla^\dagger \nabla = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{pmatrix}$$

So we see,

$$L = \nabla^\dagger \nabla.$$

The Heat Kernel

Let $f : M \longrightarrow \mathbb{R}$. Consider the **Heat Equation** on M,

$$(\partial_t + L)\, u(x,t) = 0 \quad \text{with intitial condition} \quad u(x,0) = f(x).$$

The Heat Kernel

Let $f : M \longrightarrow \mathbb{R}$. Consider the **Heat Equation** on M,

$$(\partial_t + L)\, u(x,t) = 0 \quad \text{with intitial condition} \quad u(x,0) = f(x).$$

- The solution is given by ([Ros97])

$$u(x,t) = \int_M H_t(x,y) f(y) dy,$$

where the **Heat Kernel** has the form

$$H_t(x,y) = (4\pi t)^{-\dim(M)/2} e^{-\frac{\mathrm{dist}_M(x,y)^2}{4t}} (\phi(x,y) + O(t)),$$

for certain ϕ is a smooth function with $\phi(x,x) = 1$.

The Heat Kernel

Let $f : M \longrightarrow \mathbb{R}$. Consider the **Heat Equation** on M,

$$(\partial_t + L)\, u(x,t) = 0 \quad \text{with intitial condition} \quad u(x,0) = f(x).$$

- The solution is given by ([Ros97])

$$u(x,t) = \int_M H_t(x,y) f(y) dy,$$

  where the **Heat Kernel** has the form

$$H_t(x,y) = (4\pi t)^{-\dim(M)/2} e^{-\frac{\mathrm{dist}_M(x,y)^2}{4t}} (\phi(x,y) + O(t)),$$

  for certain ϕ is a smooth function with $\phi(x,x) = 1$.
- It can be shown that, for $x_1, \cdots, x_k \in M$ and $t > 0$ small,

$$Lf(x_i) \approx \frac{1}{t}\left(f(x_i) - \frac{\sum_{0<||x_i - x_j||^2<\varepsilon} e^{-\frac{||x_i-x_j||^2}{4t}} f(x_j)}{\sum_{0<||x_i - x_j||^2<\varepsilon} e^{-\frac{||x_i-x_j||^2}{4t}}} \right)$$

  which justifies $W_{ij} = e^{-\frac{||x_i-x_j||^2}{4t}}$.

References

Slides and notebook available at juanitorduz.github.io

Mikhail Belkin and Partha Niyogi.
Laplacian eigenmaps for dimensionality reduction and data representation.
Neural Computation, 15(6):1373–1396, 2003.

Mark Kac.
Can one hear the shape of a drum?
The American Mathematical Monthly, 73(4):1–23, 1966.

Steven Rosenberg.
The Laplacian on a Riemannian Manifold: An Introduction to Analysis on Manifolds.
London Mathematical Society Student Texts. Cambridge University Press, 1997.