Geodesic Distance Function Learning via Heat Flow on Vector Fields

1 Binbin Lin 2 Ji Yang 2 Xiaofei He 1 Jieping Ye

1 Arizona State University
2 Zhejiang University

June 23, 2014
Problem 1.1 (Distance Metric Learning)

Given a manifold $\mathcal{M}$. We aim to learn a desired distance function $d(x, y)$ such that it provides a natural measure of the similarity between two data points $x$ and $y$ on the manifold $\mathcal{M}$.

- **Supervised**: label information available
- **Unsupervised**: no label information
Manifold learning can be viewed as an alternative way of unsupervised distance metric learning.

**Problem 1.2 (Manifold Learning)**

Given a $d$-dimensional manifold $\mathcal{M}$ embedded in $\mathbb{R}^m$, $d \ll m$. We aim to learn a mapping $F : \mathbb{R}^m \rightarrow \mathbb{R}^d$ such that

$$d(F(x), F(y)) = d(x, y).$$

Pro: Easy to compute the distance in the target Euclidean space

Con: The original distance may not be faithfully preserved, e.g., $\mathcal{M} = S^d$, $F : \mathcal{M} \rightarrow \mathbb{R}^d$
**Our goal**: learn the geodesic distance directly on the manifold.

- Geodesic distance is indeed a metric: it satisfies the usual axioms of metrics - positivity, symmetry and triangle inequality (Jost 2008)
- Geodesic distance is an intrinsic distance: depends on the metric of the manifold
- Geodesic distance is a fundamental distance: many variations like diffusion distance
Classic methods

- Compute the shortest path distance (Tenenbaum et al. 2000)
  - Very time consuming
  - Can’t handle the case that the manifold is not geodesically convex
- Solve the Hamilton-Jacobi equation $\|\nabla r\| = 1$ (Mémoli & Sapiro, 2001)
  - Iterative method
  - The computational cost depends on the ambient dimension

Recent approaches

- Represent the vector field by local tangent spaces (Lin et al. 2011, Singer & Wu 2012)
  - The computational cost is reduced due to the fact that it relies on the intrinsic dimension of the manifold
- Solve the heat equation on scalar field (Crane et al., 2013)
  - Solving linear systems is efficient
Main Contributions

▷ Provide a theoretical characterization of the distance function by its gradient field
▷ Propose to learn the vector field via heat flow on vector fields
▷ Establish the connection between heat flow on vector fields and vector field regularization
Let $(\mathcal{M}, g)$ be a $d$-dimensional Riemannian manifold embedded in a much higher dimensional Euclidean space $\mathbb{R}^m$, where $g$ is a Riemannian metric tensor on $\mathcal{M}$. Given a point $p$ on the manifold, we aim to learn the distance function $f_p(x) = d(p, x)$.

**Initial Vector Field.**

- Let $U_\epsilon := \{x : d(p, x) \leq \epsilon\} \subset \mathcal{M}$ be a geodesic ball around $p$.
- Let $f^0$ denote a local distance function on $U$. That is, $f^0(x) = d(p, x)$ if $p \in U$ and 0 otherwise.
- Let $V^0$ denote the gradient field of $f^0$, i.e., $V^0 = \nabla f^0$. 
Geodesic Distance Learning (GDL):

- Vector field regularization:

\[
\min_{V} E(V) := \int_{\mathcal{M}} \|V - V^0\|^2 dx + t \int_{\mathcal{M}} \|\nabla V\|_{\text{HS}}^2 dx,
\]

where \(\| \cdot \|_{\text{HS}}\) denotes the Hilbert-Schmidt tensor norm and \(t > 0\) is a parameter.

- Normalization:

\[
\hat{V}_x := \begin{cases} 
  \frac{V_x}{\|V_x\|}, & x \neq p; \\
  0, & x = p.
\end{cases}
\]

- Learn the distance function \(f\):

\[
\min_{f} \Phi(f) := \int_{\mathcal{M}} \|\nabla f - \hat{V}\|^2 dx, \quad \text{s.t.} \quad f(p) = 0.
\]
Algorithm overview. The base point is on the top of the manifold.

- (a) shows the initial vector field $V^0$
- (b) shows the vector field $V$ after transporting $V^0$ to the whole manifold using heat flows on vector fields
- (c) shows the normalized vector field $\hat{V}$ of $V$
- (d) shows the final distance function learned via requiring its gradient field to be close to $\hat{V}$

The red color indicates small distance function values and the blue color indicates large distance function values.
Given \( n \) data points \( x_i, i = 1, \ldots, n \in \mathcal{M} \subset \mathbb{R}^m \). Let \( x_q \) denote the base point. We aim to learn the distance function \( f : \mathcal{M} \rightarrow \mathbb{R} \) based at \( x_q \), i.e., \( f(x_i) = d(x_q, x_i), i = 1, \ldots, n \).

- Construct an undirected nearest neighbour graph by either \( \epsilon \)-neighbourhood or \( k \)-nn
- For each point \( x_i \), estimate its tangent space \( T_{x_i} \mathcal{M} \) by performing PCA on its neighborhood
- Let \( T_i \in \mathbb{R}^{m \times d} \) denote the matrix whose columns are constituted by the \( d \) principal components and Let \( V \) be a vector field on the manifold
Initialization

\[ v_j^0 = \begin{cases} \frac{T_j^T(x_j - x_q)}{\|T_j T_j^T(x_j - x_q)\|}, & \text{if } j \sim q \\ 0, & \text{otherwise} \end{cases} \]  

Discretization (Lin et al. 2011)

\[ E(V) = V^T V - 2V^0^T V + V^0^T V^0 + tV^T B V, \]
\[ \Phi(f) = 2f^T L f + \hat{V}^T G \hat{V} - 2\hat{V}^T C f, \]

where \( L \) is the graph Laplacian matrix and \( B \) is a \( dn \times dn \) block sparse matrix

The block matrix \( B \) provides a discrete approximation of the connection Laplacian operator.
Compute $V$:

$$\frac{\partial E(V)}{\partial V} = 0 \implies (I + tB)V = V^0. \quad (6)$$

Normalize $V$:

$$\hat{v}_i = \begin{cases} 
\frac{v_i}{\|v_i\|}, & \text{if } i \neq q \\
0, & \text{if } i = q 
\end{cases} \quad (7)$$

Compute $f$:

$$\frac{\partial \Phi(f)}{\partial f} = 0 \implies 2Lf = C^T\hat{V}, \quad (8)$$

where we restrict $f_q = 0$ when solving Eq. (8).
GDL (Geodesic Distance Learning) Algorithm

**Require:** Data sample \( X = (x_1, \ldots, x_n) \in \mathbb{R}^{m \times n} \) and a base point \( x_q, 1 \leq q \leq n \).

**Ensure:** \( f = (f_1, \ldots, f_n) \in \mathbb{R}^n \)

\begin{algorithmic}
  \FOR {i = 1 \text{ to } n}
    \STATE Compute the tangent space coordinates \( T_i \in \mathbb{R}^{m \times d} \) by using PCA
  \ENDFOR

  Set an initial vector field \( V^0 \) and construct block sparse matrices \( B \) and \( C \)

  Solve \((I + tB)V = V^0\) to obtain \( V \)

  Normalize each vector in \( V \) to obtain \( \hat{V} \)

  Solve \(2Lf = C^T\hat{V}\) to obtain \( f \)

  \RETURN \( f \)
\end{algorithmic}
Computation Complexity Analysis: (1) searching for $k$-nearest neighbors, (2) computing local tangent spaces, (3) computing $Q_{ij}$ and solving the sparse linear system Eq. (6).

- (1) For the $k$ nearest neighbor search, the complexity is $O((m + k)n^2)$
- (2) For local PCA, the complexity is $O(mnk^2)$.
- (3) The computation complexity of computing all $Q_{ij}$’s is $O(knm^2)$. We use LSQR package to solve Eq. (6) which has a complexity of $O(Imn^2)$, where $I$ is the number of iterations.
In summary, the overall computational cost for one base point is

\[ O((m + k)n^2 + mndk + kmnd^2 + Iknd^2). \]

For \( p \) base points, the extra cost is to solve Eq. (6) by adding \( p - 1 \) columns which has a complexity of \( O(pIknd^2) \). Empirically, \( d \ll m \) and \( k \ll n \). So the total computational cost for \( p \) base points could be \( O(mn^2 + pIn) \).
Characterization of Distance Functions using Gradient Fields

Theorem 5.1
Let $\mathcal{M}$ be a complete manifold. A continuous function $r : \mathcal{M} \rightarrow \mathbb{R}$ is a distance function on $\mathcal{M}$ based at $p$ if and only if

(a) $r(x) = \| \exp_p^{-1}(x) \|$ holds for a neighborhood of $p$.
(b) $\nabla r \partial r = 0$ holds on the manifold $\mathcal{M}$ except for $p \cup \text{Cut}(p)$.

Theorem 5.2
Let $\mathcal{M}$ be a complete manifold. A continuous function $r : \mathcal{M} \rightarrow \mathbb{R}$ is a distance function on $\mathcal{M}$ based at $p$ if and only if

(a) $r(x) = \| \exp_p^{-1}(x) \|$ holds for a neighborhood of $p$.
(b) $\| \partial_r \| = 1$ holds on the manifold $\mathcal{M}$ except for $p \cup \text{Cut}(p)$. 

B. Lin
Geodesic Distance Function Learning via Heat Flow on Vector Fields
Different color indicates different distance value.

- (a) shows the distance function $r_p(x)$ on the sphere, where the base point $p$ is marked in black
- (b) shows the gradient field $\nabla r_p$
- (c) shows the geodesics passing through $p$ which are denoted by the green lines

B. Lin

Geodesic Distance Function Learning via Heat Flow on Vector Fields
Vector field regularization vs. heat flow on vector fields

- Vector field regularization:

\[ \frac{\delta E(V)}{\delta V} = 0 \iff V = (I + t\nabla^*\nabla)^{-1}V^0. \quad (9) \]

- Heat flow on vector fields: Let \( X(t) \) be a vector field valued function. Given an initial vector field \( X(t)|_{t=0} = X_0 \), we have

\[ \frac{\partial X(t)}{\partial t} + \nabla^*\nabla X(t) = 0 \quad (10) \]

When \( t \) is small,

\[ \frac{X(t) - X_0}{t} + \nabla^*\nabla X(t) = 0 \iff X(t) = (I + t\nabla^*\nabla)^{-1}X_0. \quad (11) \]

If \( X_0 = V^0 \) and \( t \) is sufficiently small, performing vector field regularization is essentially solving the heat equation on vector fields.
The heat equation transfers the initial vector field primarily along geodesics

\[ X(t)(x) = \int_M k(t, x, y)X_0(y)dy \]

\[ k(t, x, y) \approx \left(\frac{1}{4\pi t}\right)^{\frac{d}{2}} e^{-d(x,y)^2/4t} \tau(x, y) \]
Synthetic example

(a) Ground truth  (b) GDL (0.02)  (c) PFRank (0.20)  (d) MR (0.38)

(e) Vector field by GDL  (f) HLLE (0.07)  (g) LE (0.11)  (h) MVU (0.05)

Figure: The base point is marked in black. Different colors indicates different distance values. The number in the brackets measures the difference between the learned distance function and the ground truth.
Ranking results on PIE and Corel data sets

Precision-scope curves.

(a) PIE
(b) Corel
Conclusion and future work

- We provide theoretical analysis to precisely characterize the geodesic distance function
- We propose a novel heat flow on vector fields approach
- Extend single manifold case to multiple manifolds case
- Incorporate the density information of the data manifold
- Develop more efficient algorithms using heat flow on vector fields
- Explore other general partial differential equations in machine learning