Approximated logarithmic maps on Riemannian manifolds and their applications

Jumpei Goto and Hiroyuki Sato

1 Department of Applied Mathematics and Physics, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan
*Corresponding author: goto.junpei.72r@st.kyoto-u.ac.jp

Received December 29, 2020, Accepted January 30, 2021

Abstract

Recently, optimization problems on Riemannian manifolds involving geodesic distances have been attracting considerable research interest. To compute geodesic distances and their Riemannian gradients, we can use logarithmic maps. However, the computational cost of logarithmic maps on Riemannian manifolds is generally higher than that on the Euclidean space. To overcome this computational issue, we propose approximated logarithmic maps. We prove that the definition is closely related to the inverse retractions. Numerical experiments for computing the Riemannian center of mass show that the proposed approximation significantly reduces the computational time while maintaining appropriate precision if the data diameter is sufficiently small.

Keywords Riemannian manifold, logarithmic map, Riemannian optimization, retraction

1. Introduction

In recent years, optimization problems on Riemannian manifolds, especially on matrix manifolds, such as the manifold of symmetric positive definite matrices (SPD manifold) and Grassmann manifold, have gained attention in applications to machine learning. Among them, algorithms using the geodesic distance have been proposed in the studies of clustering on Riemannian manifolds [1], dictionary learning and sparse coding [2], and kernel methods [3].

If a Riemannian manifold $\mathcal{M}$ is geodesically complete, then the geodesic distance between any two points $X, Y \in \mathcal{M}$ can be computed as $\text{dist}(X,Y) = \| \log_X Y \|_X$, where $\log$ denotes the Riemannian logarithmic map at $X$, and $\| \cdot \|_X$ denotes a norm at $X$ derived from the Riemannian metric [4]. Let $\text{dist}_X^2(\cdot) := \text{dist}^2(\cdot,Y)$ be the squared distance function from a fixed point $Y$. Then, we can compute the Riemannian gradient of $\text{dist}_X^2$ as $\text{grad} \text{dist}_X^2(X) = -2 \log_X Y$ using the logarithmic map. However, the computational cost of the logarithmic map on $\mathcal{M}$ is generally high. For instance, the logarithmic map on the SPD manifold is given as $\log_X Y = X^{1/2} \log \left( X^{-1/2} Y X^{-1/2} \right) X^{1/2}$, where $\log$ is the matrix logarithmic function, which is more complex than the map on $\mathbb{R}^n$, i.e., $\log_y y = y - x$.

In this study, we newly define the concept of an approximated logarithmic map on Riemannian manifolds. We attempt to reduce the computational time required to solve optimization problems involving the logarithmic map. In addition, we verify the usefulness of the approximation by numerical experiments on the computation of the Riemannian center of mass [5].

The outline of this paper is as follows. Section 2 provides preliminaries of Riemannian manifolds. In Section 3, we present the approximated logarithmic map and some examples. Applications of logarithmic maps to the computation of geodesic mean are proposed in Section 4. Numerical experiments are described in Section 5. Finally, Section 6 concludes this paper.

2. Preliminaries

In this section, we introduce the fundamental concepts of Riemannian manifolds and optimization before defining and discussing the approximated logarithmic map. For more details, one can refer to [4,6].

A Riemannian manifold is a couple $(\mathcal{M}, g)$, where $\mathcal{M}$ is a $C^\infty$ manifold and $g$ is a Riemannian metric. This metric induces an inner product in each tangent space $T_X \mathcal{M}$ at every $X \in \mathcal{M}$. We denote the inner product of $U,V \in T_X \mathcal{M}$ by $g_X(U,V)$, and the norm of $U \in T_X \mathcal{M}$ is defined by $\|U\|_X := g_X(U,U)^{1/2}$.

For every $U \in T_X \mathcal{M}$, there exists an interval $I$ around 0 and a unique geodesic $\gamma(t;X,U): I \to \mathcal{M}$ that satisfies $\gamma(0;X,U) = X$ and $\gamma'(0) = U$. The exponential map at $X \in \mathcal{M}$ is defined through this geodesic as $\exp_X : T_X \mathcal{M} \to \mathcal{M} : U \mapsto \exp_X(U) = \gamma(1;X,U)$.

The inverse of the exponential map is called a logarithmic map, which is denoted by $\log_X$ at $X \in \mathcal{M}$. If a manifold $\mathcal{M}$ is geodesically complete, i.e., the exponential map $\exp_X$ is defined at any point $X \in \mathcal{M}$ for any tangent vector in $T_X \mathcal{M}$, the geodesic distance between $X, Y \in \mathcal{M}$ can be computed as $\text{dist}(X,Y) = \| \log_X Y \|_X$.

The update formula of line-search methods on $\mathbb{R}^n$ is written as $x_{t+1} = x_t + c_t u_t$, where $u_t$ is in $\mathbb{R}^n$ and $c_t$ is a positive step size. However, on Riemannian manifold $\mathcal{M}$,
this formula cannot be applied due to the nonlinearity of \( M \). Instead, the update is formulated as
\[
X_{t+1} = R_{X_t}(c_t U_t),
\]
where \( U_t \in T_{X_t}M \) is a search direction, \( c_t \) is a positive step size, and \( R_X \) is a mapping called retraction defined as follows [6, Definition 4.1.1].

**Definition 1 (Retraction)** A retraction on a manifold \( M \) is a smooth mapping \( R \) from the tangent bundle \( TM \) to \( M \) with the following properties. Let \( R_X \) denote the restriction of \( R \) to \( T_XM \).

1. \( R_X(0_X) = X \), where \( 0_X \) denotes the zero vector of \( T_XM \);
2. \( R_X \) satisfies \( D R_X(0_X) = \text{id}_{T_XM} \), where \( \text{id}_{T_XM} \) denotes the identity mapping on \( T_XM \).

**3. Approximated logarithmic map**

In this section, we define the approximated logarithmic map following the definition of a retraction, and we introduce its properties and examples.

**Definition 2 (Approximated logarithmic map)** Let \( X \) be an arbitrary point on a Riemannian manifold \( M \). An approximated logarithmic map \( R_X^{d} \) from \( M \) onto the tangent space \( T_XM \) with the following properties:

1. \( R_X^{d}(X) = 0_X \),
2. \( D R_X^{d}(X) = \text{id}_{T_XM} \).

We originally proposed the concept of an approximated logarithmic map in a domestic conference [7]. In [8], Schiela and Ortiz independently proposed a similar concept, which they call stratification. Note that we require an approximated logarithmic map to be a smooth mappings following the definition of a retraction in [6] while Schiela and Ortiz defined a stratification as a \( C^2 \)-mapping.

Using Definitions 1 and 2 and the inverse function theorem [9], we can prove that the inverse retraction is locally equivalent to the approximated logarithmic map and vice versa, as the following theorems imply.

**Theorem 3** Let \( X \) be an arbitrary point on a Riemannian manifold \( M \) and \( R_X^{d} \) be the restriction of a retraction \( R \) to the tangent space \( T_XM \). Then, there exists an open neighborhood \( U \subseteq T_XM \) around \( 0_X \) and the inverse map \( R_X^{-d} : R_X(U) \rightarrow U \), and \( R_X^{-d} \) satisfies the two conditions of Definition 2.

**Proof** From the inverse function theorem and condition (2) of Definition 1, the smooth inverse map \( R_X^{-d} \) of the retraction \( R_X \) exists in an open neighborhood \( U \subseteq T_XM \) of \( 0_X \). Below, we prove that \( R_X^{-d} := R_X^{-d} \) satisfies the two conditions of Definition 2.

1. \( R_X^{-d}(X) = R_X^{-d}(R_X(0_X)) = 0_X \).
2. Since \( R_X^{-d} \circ R_X = \text{id}_{T_XM} \), we have
   \[
   \text{id}_{T_XM} = D(R_X^{-d} \circ R_X)(0_X)
   = D R_X^{-d}(R_X(0_X)) \circ D R_X(0_X)
   = D R_X^{-d}(X) \circ \text{id}_{T_XM}.
   \]

The third equation holds from the two conditions of Definition 1. This implies \( D R_X^{-d}(X) = \text{id}_{T_XM} \).

**Theorem 4** Let \( X \) be an arbitrary point on a Riemannian manifold \( M \) and \( R_X^{d} \) be an approximated logarithmic map on \( M \). Then, there exists an open neighborhood \( V \subseteq M \) around \( X \) and the inverse map \( R_X^{-d} : R_X(V) \rightarrow V \), and \( R_X^{-d} \) satisfies the two conditions of Definition 1, i.e., \( R_X := R_X^{-d} \) is locally a retraction on \( M \) restricted to \( T_XM \).

**Proof** From the inverse function theorem and condition (2) of Definition 2, the smooth inverse map \( R_X^{-d} \) of the approximated logarithmic map \( R_X^{d} \) exists in an open neighborhood \( V \subseteq M \) of \( X \). Below, we prove that \( R_X^{-d} \) satisfies the two conditions of Definition 1.

1. \( R_X^{-d}(0_X) = R_X^{-d}(R_X(X)) = X \).
2. Since \( R_X^{-d} \circ R_X = \text{id}_M \), we have
   \[
   \text{id}_M = D(R_X^{-d} \circ R_X)(X)
   = D R_X^{-d}(R_X(X)) \circ D R_X(X)
   = D R_X^{-d}(0_X) \circ \text{id}_{T_XM}.
   \]

The third equation holds from the two conditions of Definition 2. This implies \( D R_X^{-d}(0_X) = \text{id}_{T_XM} \).

In particular, since the exponential map is a kind of retraction, the logarithmic map satisfies Definition 2 from Theorem 3.

We show examples of geodesic distances, logarithmic maps, and their approximations on the SPD manifold:

\[
\text{Sym}_d^+ := \{ X \in \mathbb{R}^{d \times d} \mid X = X^T, \ X > O \},
\]
and the Grassmann manifold:

\[
\mathcal{G}_n := \{ \text{span}(X) \mid X \in \mathbb{R}^{n \times r}, \ X^T X = I_r \}.
\]

To compare the computational costs of the logarithmic map and its approximation, we count floating point operations (flops) used in [10].

**Example 5 (Logarithmic map and its approximation on \( \text{Sym}_d^+ \))** Let \( X, Y \) be points on \( \text{Sym}_d^+ \). The logarithmic map on \( \text{Sym}_d^+ \) is given as follows [11]:

\[
\log_X Y = X^{\dagger} \log (X^{-\dagger} Y X^{-\dagger}) X^{\dagger}
= L \log (L^{-1} Y L^{-T}) L^T,
\]

where \( L \in \mathbb{R}^{d \times d} \) is a lower triangular matrix obtained by the Cholesky decomposition \( X = LL^T \). The triangular matrix multiplication costs \( d^3 + O(d^2) \) flops, the inverse triangular matrix multiplication costs \( d^3 + O(d^2) \) flops, the Cholesky decomposition costs \( (1/3)d^3 + O(d^2) \) flops, and the matrix logarithmic function costs \( 6d^3 + O(d^2) \) (one eigendecomposition and one matrix multiplication), the cost to compute \( \log_X Y \) is \((1 \times 2 + 1 \times 2 + 1/3 + 6)d^3 + O(d^2) = (31/3)d^3 + O(d^2).\) From
the Riemannian metric on $X \in \text{Sym}^+_{d}$ that is defined by $g_X(U, V) = \text{tr}(X^{-1}UX^{-1}V)$, we can compute the geodesic distance between $X$ and $Y$ as

$$\text{dist}(X, Y) = \left\| \log \left( X^{-\frac{1}{2}} Y X^{-\frac{1}{2}} \right) \right\|_F.$$ 

We also define

$$R^d_X(Y) := -\frac{1}{2} Y X^{-1} Y + 2Y - \frac{3}{2} X,$$

which satisfies the two conditions of Definition 2 and is thus an approximated logarithmic map on $\text{Sym}^+_{d}$. This follows from its directional derivative:

$$D R^d_X(Y)[U] = -\frac{1}{2} (U X^{-1} Y + Y X^{-1} U) + 2U.$$

The cost to compute $R^d_X(Y)$ is $(10/3)d^3 + O(d^2)$.

**Example 6** (Logarithmic map and its approximation on $\mathcal{G}^+_n$). We use an orthonormal matrix $X \in \mathbb{R}^{n \times n}$ to represent a point on $\mathcal{G}^+_n$, which is the equivalence class of $X$. Let $X, Y$ be representations of points on $\mathcal{G}^+_n$. The logarithmic map on $\mathcal{G}^+_n$ is given as

$$\log_X Y = U \Sigma V^\top,$$

where $(U, \Sigma, V)$ is the solution of the following equation [12]:

$$\begin{bmatrix} X^\top Y \\ (I_n - XX^\top) Y \end{bmatrix} = \begin{bmatrix} V \cos(\Sigma) V^\top \\ U \sin(\Sigma) V^\top \end{bmatrix} .$$

The solution is obtained by the singular value decomposition (SVD):

$$(I_n - XX^\top) Y (X^\top Y)^{-1} = U \tan(\Sigma) V^\top .$$

From the Riemannian metric on $X \in \mathcal{G}^+_n$ that is defined by $g_X(U, V) = \text{tr}(U^\top V)$, we can compute the geodesic distance between $X$ and $Y$ as

$$\text{dist}(X, Y) = \left\| \arccos(\text{diag} \Sigma) \right\|_2 .$$

We also define

$$R^d_X(Y) := (I_n - XX^\top) Y (X^\top Y)^{-1} ,$$

which is an inverse map of the retraction:

$$R_X(U) = \text{qf}(X + U),$$

where $\text{qf}(\cdot)$ denotes the map which returns the $Q$ factor of its QR decomposition [6, Example 4.1.2]. From Theorem 3, $R^d_X$ is an approximated logarithmic map on $\mathcal{G}^+_n$. Compared with the logarithmic map, we can avoid the SVD computation by this approximation.

4. Application to geodesic mean

This section introduces the optimization problem to compute the Riemannian center of mass as an application of approximated logarithmic maps. The geodesic mean of $N$ points $A_1, A_2, \ldots, A_N \in \mathcal{M}$ is computed as the solution to the following optimization problem:

$$\minimize_{X \in \mathcal{M}} \ f(X) = \frac{1}{2N} \sum_{i=1}^N \left\| \log_X(A_i) \right\|_X^2 .$$

**Algorithm 1** Gradient descent (GD)

**Require:** $A_1, A_2, \ldots, A_N \in \mathcal{M}$

1. Initialize $X_0 = A_1$.
2. for $t = 1, 2, \ldots$ do
3. Calculate $c_t > 0$.
4. $X_t = R_{X_{t-1}} \left( \frac{2}{N} \sum_{i=1}^N \log_{X_{t-1}}(A_i) \right)$.
5. end for
6. return $X_t$.

**Algorithm 2** Approximated gradient descent (AGD)

**Require:** $A_1, A_2, \ldots, A_N \in \mathcal{M}$

1. Initialize $X_0 = A_1$.
2. for $t = 1, 2, \ldots$ do
3. Calculate $c_t > 0$.
4. $X_t = R_{X_{t-1}} \left( \frac{2}{N} \sum_{i=1}^N \log_{X_{t-1}}(A_i) \right)$.
5. end for
6. return $X_t$.

| Table 1. Computational time [s]. |
|-------------------------------|
| $\text{Sym}_{10}^+$       | 1.02435  | 0.54569 |
| $\mathbb{G}^{20}_{1000}$   | 7.44599  | 2.40129 |

The gradient $\nabla f(X)$ of the objective function $f$ at $X \in \mathcal{M}$ is computed by the following formula [5]:

$$\nabla f(X) = -\frac{1}{N} \sum_{i=1}^N \log_X A_i .$$

Adapting $-\nabla f(X)$ as the search direction leads to the gradient descent algorithm (Algorithm 1).

By replacing the logarithmic map with its approximation, we approximate the gradient as

$$\nabla f(X) \approx -\frac{1}{N} \sum_{i=1}^N R^d_X A_i ,$$

and we obtain Algorithm 2.

5. Numerical Experiments

In this section, we address the Riemannian mean problem on $\text{Sym}_{10}^+$ and $\mathbb{G}^{20}_{1000}$ using Algorithms 1 and 2, i.e., the gradient descent (GD) and approximated gradient descent (AGD) algorithms introduced in the previous section. All the experiments were implemented in Python 3.7.3 and run on 2.3 GHz dual-core Intel Core i5.

5.1 Parameters and datasets

The step size $c_t$ is set to a constant, selected from $\{1, 0.1\}$. Both algorithms are terminated when the number of iterations reaches 300. Datasets are generated using random values according to Algorithms 3 and 4. The number of data points is set to 30.

5.2 Computational time

Table 1 shows the 10-time average of the computational time to compute the Riemannian center when
the step size is selected as $c_1 = 0.1$ and the parameter $\sigma$ in Algorithms 3 and 4 is set to 1. We can observe that the computational time is significantly reduced by the approximation in both $\operatorname{Sym}^+_d$ and $G^{20}_{1000}$. Especially on $\operatorname{Sym}^+_d$, the ratio of the computational time is $1.02435/0.34560 \approx 2.96398$, which is close to the theoretical ratio $(31/3)/(10/3) = 3.1$ described in Section 3.

5.3 Precision of approximation

We conducted several experiments while changing the data diameter (maximum distance between points) by selecting $\sigma > 0$ from $[0.01, 1]$ in Algorithms 3 and 4. Fig. 1 shows the results of objective function values. We can observe that the approximation works well if the data diameter is sufficiently small. However, when the diameter exceeds 3 in $\operatorname{Sym}^+_d$ or exceeds 6.3 in $G^{20}_{1000}$, the objective function values of AGD are higher than those of GD because of the approximation error.

6. Conclusion

In this paper, we newly proposed the approximated logarithmic maps on Riemannian manifolds and proved their relation with retractions, which are approximations of the exponential maps. Our approach allows algorithms that require inverse retractions (such as [13]) to use approximated logarithmic maps instead even without knowing the retraction itself.

In addition, the Riemannian center of mass is appropriately computed using the approximation if the data diameter is sufficiently small. In future works, convergence and error bounds should be analyzed to confirm the results. Performing numerical experiments with real-world data (instead of random-valued data) is also a future task.

Acknowledgments

This study was supported by JSPS KAKENHI Grant Number JP20K14359. The authors would like to thank the reviewer for his/her helpful comments.

References

[1] A. Goh, Riemannian manifold clustering and dimensionality reduction for vision-based analysis, in: Machine Learning for Vision-Based Motion Analysis, L. Wang, G. Zhao, L. Cheng and M. Pietikäinen eds., pp.27–53, Springer, London, 2011.
[2] Y. Xie, J. Ho and B. Vemuri, On a nonlinear generalization of sparse coding and dictionary learning, in: Proc. of the 30th International Conference on Machine Learning, Vol. 28, pp.1480–1488, 2013.
[3] S. Jayasumana, R. Hartley, M. Salzmann, H. Li and M. Harandi, Kernel methods on Riemannian manifolds with Gaussian RBF kernels, IEEE Transactions on Pattern Analysis and Machine Intelligence, 37 (2015), 2464–2477.
[4] M. Spivak, A Comprehensive Introduction to Differential Geometry, Vol. 1, 3rd edn., Publish or Perish, Inc., Houston, 1999.
[5] H. Karcher, Riemannian center of mass and mollifier smoothing, Commun. Pure Appl. Math., 30 (1977), 509–541.
[6] P.-A. Absil, R. Mahony and R. Sepulchre, Optimization Algorithms on Matrix Manifolds, Princeton University Press, Princeton, 2008.
[7] J. Goto and H. Sato, Approximated logarithmic maps on Riemannian manifolds and their applications to optimization problems (in Japanese), in: Abstract of The 2020 Spring National Conference of Operations Research Society of Japan, pp.72–73, March, 2020.
[8] A. Schiela and J. Ortiz, An SQP method for equality constrained optimization on manifolds, arXiv:2005.06844 [math.OC], May, 2020.
[9] R. S. Hamilton, The inverse function theorem of Nash and Moser, Bull. Am. Math. Soc., 7 (1982), 65–222.
[10] J. M. Demmel, Applied Numerical Linear Algebra, SIAM, Philadelphia, 1997.
[11] S. Sra and R. Hosseini, Conic geometric optimization on the manifold of positive definite matrices, SIAM J. Control Optim., 25 (2015), 713–739.
[12] J. Zhang, G. Zhu, R. W. Heath Jr. and K. Huang, Grassmannian learning; Embedding geometry awareness in shallow and deep learning, arXiv:1808.02229 [cs.LG].
[13] X. Zhu and H. Sato, Riemannian conjugate gradient methods with inverse retraction, Comput. Optim. Appl., 77 (2020), 779–810.