

Article

Not peer-reviewed version

A K-means Clustering Algorithm with Total Bregman Divergence for Point Cloud Denoising

[Xiaomin Duan](#), [Anqi Mu](#), [Xinyu Zhao](#)^{*}, Yuqi Wu

Posted Date: 30 June 2025

doi: 10.20944/preprints202506.2252.v1

Keywords: Total Bregman divergence; point cloud denoising; K-means clustering; manifold learning; influence function; anisotropic index



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

A K-means Clustering Algorithm with Total Bregman Divergence for Point Cloud Denoising

Xiaomin Duan ¹, Anqi Mu ¹, Xinyu Zhao ^{2,*}, Yuqi Wu ¹

¹ School of Science, Dalian Jiaotong University, Dalian 116028, China

² School of Materials Science and Engineering, Dalian Jiaotong University, Dalian 116028, China

* Correspondence: xyz@djtu.edu.cn; Tel.: +86-411-8410-6287

Abstract

Point cloud denoising is essential for improving 3D data quality, yet traditional K-means methods relying on Euclidean distance struggle with non-uniform noise. This paper proposes a K-means algorithm leveraging Total Bregman Divergence (TBD) to better model geometric structures on manifolds, enhancing robustness against noise. Specifically, TBDs - Total Logarithm, Exponential, and Inverse Divergences - are defined on symmetric positive-definite matrices, each tailored to capture distinct local geometries. Theoretical analysis demonstrates the bounded sensitivity of TBD-induced means to outliers via influence functions, while anisotropy indices quantify structural variations. Numerical experiments validate the methods superiority over Euclidean-based approaches, showing effective noise separation and improved stability. The work bridges geometric insights with practical clustering, offering a robust framework for point cloud preprocessing in vision and robotics applications.

Keywords: total Bregman divergence; point cloud denoising; K-means clustering; manifold learning; influence function; anisotropic index

1. Introduction

In recent years, the rapid advancement of 3D scanning technologies has led to an explosion of point cloud data in various fields, including computer vision, robotics, and autonomous systems [1–4]. As raw digitalizations of physical structures, point clouds often contain inherent noise that degrades segmentation and 3D reconstruction fidelity. Despite their utility, raw point cloud datasets frequently contain inherent imperfections like noise and outliers that compromise the accuracy of downstream applications including segmentation and 3D reconstruction. This inherent data quality challenge necessitates robust denoising algorithms as a fundamental preprocessing requirement for reliable 3D analysis [5–10].

Among the various denoising techniques, clustering-based methods have shown promising results due to their ability to group similar points together and distinguish between signal and noise [11,12]. While conventional K-means clustering demonstrates effectiveness in regular data spaces, its dependence on isotropic distance measures creates fundamental limitations when processing irregular 3D structures. Specifically, the Euclidean metric fails to adequately characterize manifold geometries inherent to point cloud data, particularly under spatially variant noise conditions.

Our methodological contribution resolves this through a density-aware clustering framework that adapts to local topological characteristics [13,14]. Given the differences in local geometric structures between noise and valid data on the manifold, points with similar local features should be grouped together into the same cluster. Previous studies [15,16] have employed the K-means clustering algorithm for point cloud denoising, but they relied solely on a limited number of metrics to measure distances on the manifold. Furthermore, these studies overlooked the influence function, which is crucial for evaluating the robustness of the mean derived from each metric in the presence of outliers. Conversely, [17] proposed a point cloud denoising method leveraging the geometric structure of the

Gaussian distribution manifold, employing five distinct measures: Euclidean distance, Affine-Invariant Riemannian Metric (AIRM), Log-Euclidean Metric (LEM), Kullback-Leibler Divergence (KLD), and its symmetric variant (SKLD). The research evaluated metric robustness by deriving influence functions for mean estimators under outlier conditions. Experimental results revealed that geometric metrics outperformed Euclidean measurements in denoising quality, with geometric means showing greater outlier resistance than arithmetic means. Notably, KLD and SKLD demonstrated computational advantages over AIRM/LEM through reduced complexity. Bregman divergence extends squared Euclidean distances to diverse data-adaptive metrics, enabling more efficient computation of means and influence functions compared to AIRM/LEM frameworks, thus enhancing processing efficiency while maintaining theoretical rigor.

This work introduces a novel divergence class measuring orthogonal deviation between a convex differentiable function's value at one point and its tangent approximation at another, formally termed Total Bregman Divergence (TBD). A schematic comparison in [18] visually clarifies TBD's geometric distinction from conventional Bregman divergence. Notably, the Total Bregman Divergence exhibits heightened robustness and adaptability when managing intricate data structures.

This paper builds upon our previous concept presented in [17] by introducing a novel class of methods specifically designed for point cloud denoising. Furthermore, we implement the idea of K-means clustering, leveraging local statistical features to distinguish between noise and valid data. TBD offers advantages over traditional Bregman divergence by providing invariance to coordinate transformations, measuring orthogonal distance between function values and tangents, potentially improving robustness against noise, and generalizing the standard Bregman divergence while retaining its useful properties, albeit the specific benefits may vary depending on the application context and function choice.

Three fundamental breakthroughs constitute the core innovations of this research:

(1) We introduce a K-means clustering algorithm based on TBDs specifically designed for denoising high-density noise point clouds. This methodology capitalizes on the local statistical differences between noise and valid data within the context of Gaussian distribution families.

(2) Bregman divergences are defined on the manifold of positive definite matrices by employing convex functions. Associated with these divergences, anisotropic indices are proposed, which serve to reflect the local geometric structure surrounding each point on the matrix manifold. Furthermore, TBDs are endowed on the manifold, including the total logarithm divergence (TLD), the total exponent divergence (TED), and the total inverse divergence (TID).

(3) The means of several positive-definite matrices are computed based on TBDs. To evaluate the robustness of these means, approximations and upper bounds for their influence functions are provided. Numerical simulations reveal that the loss function of the TBDs-induced means is lower than that induced by the Euclidean distance, and the weighted logarithm Euclidean divergence exhibits a smaller cost function compared to the mean induced by the logarithmic Euclidean metric, thereby affirming the algorithm's effectiveness and advantage.

The paper is structured into six main sections. Section 1 introduces the importance of point cloud denoising and highlights the limitations of traditional K-means clustering approaches based on Euclidean distance. Section 2 delves into the theoretical foundations, defining TBD and deriving the means of several positive-definite matrices. Section 3 proposes a novel K-means clustering algorithm that leverages TBD for point cloud denoising. Section 4 provides an in-depth analysis of the anisotropy indices of Bregman divergence and the influence functions of the TBD means. Section 5 presents simulation results and performance comparisons, demonstrating the effectiveness of the proposed algorithm. Section 6 concludes by summarizing the main contributions and discussing potential applications and future research directions.

2. Geometry on the Manifold of Symmetric Positive-Definite Matrices

The collection of $n \times n$ real matrices is represented by $M_{n \times n}$, whereas the subset comprising all invertible $n \times n$ real matrices forms the general linear group, denoted as $GL(n, \mathbb{R})$. Notably, $GL(n, \mathbb{R})$ possesses a differentiable manifold structure, with $M_{n \times n}$ functioning as its Lie algebra, denoted as $\mathfrak{gl}(n, \mathbb{R})$. The exchange of information between $GL(n, \mathbb{R})$ and $M_{n \times n}$ is facilitated by exponential and logarithmic mappings. In particular, the exponential map, given by

$$\text{Exp}(X) = \sum_{k=0}^{\infty} \frac{X^k}{k!},$$

converts a matrix X in $M_{n \times n}$ to an element in $GL(n, \mathbb{R})$. Conversely, for an invertible matrix A devoid of eigenvalues on the closed negative real axis, there exists a unique logarithm with eigenvalues within the strip $\{z \in \mathbb{C} \mid -\pi < \text{Im}(z) < \pi\}$. This logarithm, serving as the inverse of the exponential map, is termed the principal logarithm map and symbolized by

$$\text{Log}(A).$$

Let \mathcal{S}_n denote the space of real symmetric $n \times n$ matrices:

$$\mathcal{S}_n = \{X \in \mathbb{R}^{n \times n} \mid X = X^T\}. \quad (1)$$

The subset of symmetric positive-definite matrices forms the Riemannian manifold:

$$\mathcal{P}_n = \{\Sigma \in \mathcal{S}_n \mid \mathbf{v}^T \Sigma \mathbf{v} > 0, \forall \mathbf{v} \in \mathbb{R}^n \setminus \{\mathbf{0}\}\}. \quad (2)$$

Three fundamental metric structures on \mathcal{P}_n are considered:

- (i) **Euclidean (Frobenius) Framework:** The canonical inner product on \mathcal{P}_n is defined as

$$\langle X, Y \rangle = \text{tr}(X^T Y), \quad (3)$$

inducing the norm $\|X\| = \sqrt{\langle X, X \rangle}$ and metric distance

$$d_F(\Sigma_1, \Sigma_2) = \|\Sigma_1 - \Sigma_2\|. \quad (4)$$

The tangent space $T_\Sigma \mathcal{P}_n$ at any $\Sigma \in \mathcal{P}_n$ coincides with \mathcal{S}_n due to \mathcal{P}_n 's open submanifold structure in \mathcal{S}_n .

- (ii) **Affine-Invariant Riemannian Metric (AIRM):** The geometry metric at $\Sigma \in \mathcal{P}_n$ is given by

$$g_\Sigma(V, W) = \text{tr}(\Sigma^{-1} V \Sigma^{-1} W), \quad V, W \in T_\Sigma \mathcal{P}_n. \quad (5)$$

This induces the geodesic distance [19,20]:

$$d_{\text{AIRM}}(\Sigma_1, \Sigma_2) = \|\log(\Sigma_1^{-1/2} \Sigma_2 \Sigma_1^{-1/2})\|. \quad (6)$$

- (iii) **Log-Euclidean Metric (LEM):** Through the logarithmic group operation

$$\Sigma_1 \otimes \Sigma_2 := \text{Exp}(\text{Log}(\Sigma_1) + \text{Log}(\Sigma_2)), \quad (7)$$

\mathcal{P}_n becomes a Lie group. The metric at Σ is defined via differential mappings:

$$g_\Sigma(V, W) = \langle D_\Sigma \text{Log}(V), D_\Sigma \text{Log}(W) \rangle, \quad (8)$$

where $D_{\Sigma}\text{Log}$ denotes the differential of the matrix logarithm. The corresponding distance becomes

$$d_{\text{LEM}}(\Sigma_1, \Sigma_2) = \|\text{Log}(\Sigma_1) - \text{Log}(\Sigma_2)\|, \quad (9)$$

effectively Euclideanizing the manifold geometry through logarithmic coordinates.

2.1. Bregman Divergence on Manifold \mathcal{P}_n

For matrices $A, B \in \text{GL}(n, \mathbb{R})$, the Bregman matrix divergence associated with a strictly convex differentiable potential function Φ is given by:

$$\mathcal{D}_{\Phi}(A, B) = \Phi(A) - \Phi(B) - \langle \nabla \Phi(B), A - B \rangle, \quad (10)$$

where $\langle \cdot, \cdot \rangle$ denotes the Frobenius inner product [21].

This divergence can be systematically extended to a total Bregman divergence (TBD) through the subsequent formulation. For invertible matrices $A, B \in \text{GL}(n, \mathbb{R})$, the TBD is defined as [22]

$$\delta_{\Phi}(A, B) = \frac{\Phi(A) - \Phi(B) - \langle \nabla \Phi(B), A - B \rangle}{\sqrt{1 + \|\nabla \Phi(B)\|^2}}. \quad (11)$$

When calculating the divergence associated with a certain convex function $\Phi(A)$, the Riemannian gradient of $\Phi(A)$ is often needed, which can be obtained by the covariant/directional derivative related to the Riemannian metric as follows:

$$\langle \nabla \Phi(A), X \rangle = \left. \frac{d}{d\varepsilon} \right|_{\varepsilon=0} \Phi(\gamma(\varepsilon)), \forall X \in T_A \mathcal{P}_n \quad (12)$$

with the curve $\gamma : [0, 1] \rightarrow \mathcal{P}_n$ and $\gamma(0) = A, \dot{\gamma}(0) = X$. After linearizing the curve γ , (12) can be rewritten as

$$\langle \nabla \Phi(A), X \rangle = \left. \frac{d}{d\varepsilon} \right|_{\varepsilon=0} \Phi(A + \varepsilon X), \forall X \in T_A \mathcal{P}_n. \quad (13)$$

Proposition 1. For any nonsingular matrix $A \in \text{GL}(n, \mathbb{R})$ with spectral exclusion condition $\sigma(A) \cap \mathbb{R}_{\leq 0} = \emptyset$, there exists a well-defined matrix entropy functional:

$$\Phi(A) = \text{tr}(A \text{Log}(A)), \quad (14)$$

which generalizes the scalar function $\phi(x) = x \ln x$ to the matrix setting through trace-theoretic extension [22]. Then, the Riemannian gradient of $\Phi(A)$ is

$$\nabla \Phi(A) = \text{Log}(A) + I. \quad (15)$$

Furthermore, the Bregman divergence is expressed as follows

$$D_{\Phi}(A, B) = \text{tr}(A(\text{Log}(A) - \text{Log}(B)) - (A - B)), \quad (16)$$

and the total logarithm divergence (TLD) is provided by

$$\delta_{\Phi}(A, B) = \frac{\text{tr}(A(\text{Log}(A) - \text{Log}(B)) - (A - B))}{\sqrt{1 + \|\text{Log}(B) + I\|^2}}. \quad (17)$$

Proof. From (13), $\nabla\Phi(B)$ is calculated by

$$\begin{aligned}\langle \nabla\Phi(B), V \rangle &= \left. \frac{d}{d\varepsilon} \right|_{\varepsilon=0} \Phi(B + \varepsilon V) \\ &= \left. \frac{d}{d\varepsilon} \right|_{\varepsilon=0} \text{tr}(\gamma(\varepsilon)\text{Log}\gamma(\varepsilon)) \\ &= \text{tr}(\text{Log}(B)V) - \text{tr}\left(\gamma(\varepsilon)\frac{d}{d\varepsilon}\text{Log}\gamma(\varepsilon)\right)\Bigg|_{\varepsilon=0},\end{aligned}$$

where

$$\gamma(\varepsilon) := B + \varepsilon V. \quad (18)$$

Next, we will calculate the second term of the above expression, namely the differential term. Using Lemmas 3 and 4 in [17], we obtain

$$\begin{aligned}&\text{tr}\left(\gamma(\varepsilon)\frac{d}{d\varepsilon}\text{Log}\gamma(\varepsilon)\right) \\ &= \text{tr}\left(\int_0^1 \gamma(\varepsilon)((\gamma(\varepsilon) - I)t + I)^{-1} \frac{d\gamma(\varepsilon)}{d\varepsilon} ((\gamma(\varepsilon) - I)t + I)^{-1} dt\right) \\ &= \text{tr}\left(\int_0^1 ((\gamma(\varepsilon) - I)t + I)^{-1} \gamma(\varepsilon) \frac{d\gamma(\varepsilon)}{d\varepsilon} ((\gamma(\varepsilon) - I)t + I)^{-1} dt\right) \\ &= \int_0^1 \text{tr}\left(((\gamma(\varepsilon) - I)t + I)^{-2} \gamma(\varepsilon) \frac{d\gamma(\varepsilon)}{d\varepsilon}\right) dt \\ &= \text{tr}\left(\gamma(\varepsilon)^{-1} \gamma(\varepsilon) V\right) \\ &= \text{tr}(V).\end{aligned}$$

Then, we have

$$\langle \nabla\Phi(B), V \rangle = \text{tr}((\text{Log}(B) + I)V) = \langle \text{Log}(B) + I, V \rangle, \quad (19)$$

and thus (15) has been proven. Substituting the result into equation (10), the proof of equation (16) is complete. \square

Similar to the proof methodology employed in Proposition 1, we can derive Proposition 2 and Proposition 3 through analogous applications.

Proposition 2. Consider $\Phi(A) = \text{tr}(\text{Exp}(A))$, which is induced from the function e^x [21]. Then, the Riemannian gradient of $\Phi(A)$ is

$$\nabla\Phi(B) = \text{Exp}(A). \quad (20)$$

Furthermore, the Bregman divergence is expressed as

$$D_\Phi(A, B) = \text{tr}(\text{Exp}(A) - \text{Exp}(B) - \text{Exp}(B)(A - B)), \quad (21)$$

and the total exponential divergence (TED) is provided by

$$\delta_\Phi(A, B) = \frac{\text{tr}(\text{Exp}(A) - \text{Exp}(B) - \text{Exp}(B)(A - B))}{\sqrt{1 + \|\text{Exp}(B)\|^2}}. \quad (22)$$

Proposition 3. When A is invertible and let $\Phi(A) = \text{tr}(A^{-1})$, which is induced from the function $\frac{1}{x}$ [21]. Then, the Riemannian gradient of $\Phi(A)$ is

$$\nabla\Phi(B) = -B^{-2}. \quad (23)$$

Furthermore, the Bregman divergence is expressed as follows

$$D_{\Phi}(A, B) = \text{tr}(A^{-1} + AB^{-2} - 2B^{-1}), \quad (24)$$

and the total inverse divergence (TID) is provided by

$$\delta_{\Phi}(A, B) = \frac{\text{tr}(A^{-1} + AB^{-2} - 2B^{-1})}{\sqrt{1 + \|B^{-2}\|^2}}. \quad (25)$$

To analyze the differences among the TBD divergences defined on the positive-definite matrix manifold, Figure 1 shows 3-dimensional isosurfaces centered at the identity matrix, which are induced by TLD, TED, and TID, respectively. All of these isosurfaces are convex balls with non-spherical shapes, differing completely from the spherical isosurfaces in the context of the Euclidean metric.

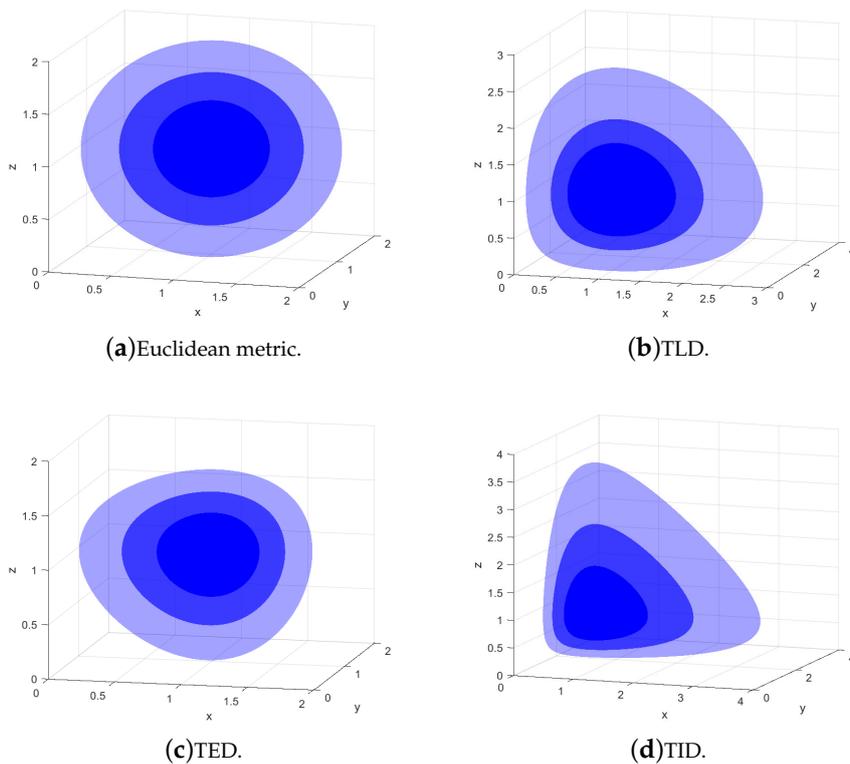


Figure 1. Isosurfaces induced by Euclidean metric and TBDs.

2.2. TBDs Means on \mathcal{P}_n

In this subsection, we study the geometric mean of several symmetric positive-definite matrices induced by the TBD, by considering the minimization problem of an objective function.

For m positive real numbers $\{a_i\}$, the arithmetic mean is often denoted as \hat{a} and expressed as

$$\bar{a} := \frac{1}{m} \sum_{i=1}^m a_i = \arg \min_{a>0} \sum_{i=1}^m |a - a_i|^2, \quad (26)$$

where $|a - a_i|$ represents the absolute difference between a and a_i , which signifies the distance separating the two points on the real number line. The mean of m symmetric positive-definite matrices $\{A_1, A_2, \dots, A_m\} \in \mathcal{P}_n$ is the solution to

$$\bar{A} := \arg \min_{A \in \mathcal{P}_n} \sum_{i=1}^m d^2(A, A_i). \quad (27)$$

Thus, the arithmetic mean of $\{A_1, A_2, \dots, A_m\}$ endowed by the Euclidean metric can be represented as

$$\bar{A} = \frac{1}{m} \sum_{i=1}^m A_i. \quad (28)$$

And, the mean of $\{A_1, A_2, \dots, A_m\}$ with LEM is [16]

$$\bar{A} = \text{Exp} \left(\frac{1}{m} \sum_{i=1}^m \text{Log}(A_i) \right). \quad (29)$$

The result indicates that the Log-Euclidean mean has Euclidean characteristics when considered in the logarithmic domain, whereas AIRM's mean lacks these properties [17].

The concept extends naturally to the TBD mean. For a strictly convex differentiable function Φ and matrices $\{A_1, A_2, \dots, A_m\}$, the TBD mean is defined as

$$\bar{A} := \arg \min_{A \in \mathcal{P}_n} \sum_{i=1}^m \delta_{\Phi}(A, A_i). \quad (30)$$

The strict convexity of $\Phi(A)$ ensures the Bregman divergence $\sum_{i=1}^m \delta_{\Phi}(A, A_i)$ retains strict convexity in A . Consequently, uniqueness of the TBD mean (30) follows directly from convex optimization principles, provided such a mean exists. To guarantee existence, $\delta_{\Phi}(A)$ must operate within a compact subset of the symmetric positive-definite matrix manifold \mathcal{P}_n . Compactness ensures adherence to the Weierstrass extreme value theorem, which mandates attainment of minima for continuous functions over closed and bounded domains conditions inherently satisfied by the convex functional on this Riemannian manifold. Let's define $H(A)$ as

$$H(A) := \frac{1}{m} \sum_{i=1}^m \delta_{\Phi}(A, A_i). \quad (31)$$

According to (13), the Riemannian gradient of $H(A)$ can be obtained as

$$\nabla H(A) = \frac{1}{m} \sum_{i=1}^m \frac{\nabla \Phi(A) - \nabla \Phi(A_i)}{\sqrt{1 + \|\nabla \Phi(A_i)\|^2}}. \quad (32)$$

Then, by solving $\nabla H(A) = 0$, the TBD mean can be expressed as

$$\nabla \Phi(\bar{A}) = \sum_{i=1}^m \Omega_i \nabla \Phi(A_i) \quad (33)$$

with the weight

$$\Omega_i = \frac{1}{\sqrt{1 + \|\nabla \Phi(A_i)\|^2}} \bigg/ \sum_{j=1}^m \frac{1}{\sqrt{1 + \|\nabla \Phi(A_j)\|^2}}. \quad (34)$$

Next, by substituting (15), (20), and (23) into (33) respectively, it is straightforward to get explicit expressions for the means corresponding to TLD, TED, and TID.

Proposition 4. The TLD mean of m symmetric positive-definite matrices $\{A_1, A_2, \dots, A_m\}$ is provided by

$$\bar{A} = \text{Exp} \left(\sum_{i=1}^m \Omega_i \text{Log}(A_i) \right), \quad (35)$$

where

$$\Omega_i = \frac{1}{\sqrt{1 + \|\text{Log}(A_i) + I\|^2}} \bigg/ \sum_{j=1}^m \frac{1}{\sqrt{1 + \|\text{Log}(A_j) + I\|^2}}. \quad (36)$$

By comparing (29) and (35), it can be seen that the TLD mean is essentially a weighted version of the LEM mean.

Proposition 5. The TED mean of m symmetric positive-definite matrices $\{A_1, A_2, \dots, A_m\}$ is provided by

$$\bar{A} = \text{Log} \left(\sum_{i=1}^m \Omega_i \text{Exp}(A_i) \right), \quad (37)$$

where

$$\Omega_i = \frac{1}{\sqrt{1 + \|\text{Exp}(A_i)\|^2}} \bigg/ \sum_{j=1}^m \frac{1}{\sqrt{1 + \|\text{Exp}(A_j)\|^2}}. \quad (38)$$

Proposition 6. The TID mean of m positive definite matrices $\{A_1, A_2, \dots, A_m\}$ is provided by

$$\bar{A} = \sum_{i=1}^m \Omega_i A_i^{-2}, \quad (39)$$

where

$$\Omega_i = \frac{1}{\sqrt{1 + \|A_i^{-2}\|^2}} \bigg/ \sum_{j=1}^m \frac{1}{\sqrt{1 + \|A_j^{-2}\|^2}}. \quad (40)$$

3. K-Means Clustering Algorithm with TBDs

In the n -dimensional Euclidean space R^n , we denote a point cloud of size ι as

$$\mathcal{B} := \{\mathbf{b}_i \in R^n \mid i = 1, \dots, \iota\}. \quad (41)$$

For $\forall \mathbf{b} \in \mathcal{B}$, we first identify its local neighborhood $\mathcal{S} \equiv \Delta(\mathbf{b}, q)$ using the k -nearest neighbors algorithm. Subsequently, the intrinsic geometry of \mathcal{S} is characterized by computing two statistical descriptors:

1. The centered mean vector:

$$\boldsymbol{\mu}(\mathcal{S}) := \mathbb{E}[\mathcal{S}] - \mathbf{b},$$

where $\mathbb{E}[\cdot]$ denotes empirical expectation;

2. The covariance operator:

$$\boldsymbol{\Sigma}(\mathcal{S}) := \text{Cov}(\mathcal{S}, \mathcal{S})$$

quantifying pairwise positional deviations.

These descriptors induce a parameterization of \mathcal{S} as points on the statistical manifold of n -dimensional Gaussian distributions

$$\mathcal{S}_n = \left\{ f \mid f(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^n \det(\boldsymbol{\Sigma})}} \text{Exp} \left(-\frac{(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})}{2} \right), \mathbf{x} \in R^n \right\}, \quad (42)$$

where f denotes the probability density function. This geometric embedding facilitates subsequent analysis within the framework of information geometry. The point cloud \mathcal{B} undergoes local statistical encoding through operator Y , generating its parametric representation $\tilde{\mathcal{B}} := Y(\mathcal{B}) \subset \mathcal{S}_n$ in symmetric matrix space. This mapped ensemble, termed the statistical parameter point cloud, preserves neighborhood geometry through covariance descriptors while enabling manifold-coordinate analysis.

Due to the topological homeomorphism between \mathcal{S}_n and $R^n \times \mathcal{P}_n$ [23], the geometric structure on \mathcal{S}_n can be induced by assigning metrics on $R^n \times \mathcal{P}_n$. Additionally, the distance on \mathcal{S}_n is denoted as

$$d((\mu_1, \Sigma_1), (\mu_2, \Sigma_2)) = \|\mu_1 - \mu_2\| + \text{diff}(\Sigma_1, \Sigma_2), \quad (43)$$

where diff stands for the difference between Σ_1 and Σ_2 . The mean of $(\mu_i, \Sigma_i)_1^l$ on the parameter point cloud $\tilde{\mathcal{B}}$ is denoted as $(\mathbf{c}_R, \mathbf{c}_P)$, where

$$\mathbf{c}_R = \frac{1}{l} \sum_{i=1}^l \mu_i, \mathbf{c}_P = \arg \min_{\Sigma} \sum_{i=1}^l d^2(\Sigma_i, \Sigma). \quad (44)$$

The definitions of both distance and mean operators on $\tilde{\mathcal{B}}$ are contingent upon the specific metric structure imposed on $\tilde{\mathcal{B}}$. In our proposed algorithm, these fundamental statistical measures will be directly computed through TBDs, which play a crucial role in establishing the geometric framework for subsequent computations.

The intrinsic statistical properties of valid and stochastic noise components exhibit fundamental divergences in their local structural organizations. This statistical separation forms the basis for our implementation of a K-means clustering framework to partition the complete dataset $\tilde{\mathcal{B}}$ into distinct phenomenological categories: structured information carriers and unstructured random perturbations. The formal procedure for this discriminative clustering operation is systematically outlined in Algorithm 1.

Algorithm 1 Signal-Noise Discriminative Clustering Framework.

1. **Point Cloud Parametrization:**

- Statistically encode \mathcal{B} into $\tilde{\mathcal{B}} \subset \mathcal{S}_n$ via covariance descriptors.

2. **Applying the K-means Algorithm:**

Step a: Barycenters Setup

- Assign barycenters (cluster centers) for the two clusters as $(\mathbf{c}_R^{1(i)}, \mathbf{c}_P^{1(i)})$ and $(\mathbf{c}_R^{2(i)}, \mathbf{c}_P^{2(i)})$.

Step b: Grouping with K-means

- Use the K-means algorithm with the TBDs to group the points in the parameter point cloud $\tilde{\mathcal{B}}$ into two clusters based on their proximity to the current barycenters.

Step c: Updating Barycenters

- Recalculate the barycenters for the two clusters based on the new clustering results, updating them to $(\mathbf{c}_R^{1(i+1)}, \mathbf{c}_P^{1(i+1)})$ and $(\mathbf{c}_R^{2(i+1)}, \mathbf{c}_P^{2(i+1)})$.

Step d: Convergence Check

- Set a convergence threshold ϵ_0 and a convergence condition $d((\mathbf{c}_R^{1(i)}, \mathbf{c}_P^{1(i)}), (\mathbf{c}_R^{2(i+1)}, \mathbf{c}_P^{2(i+1)})) < \epsilon_0$.
 - Upon satisfying convergence criteria, project the partitioned clusters in $\tilde{\mathcal{B}}$ to \mathcal{B} , then finalize the computation with categorical assignments.
 - If the convergence condition is not met, repeat steps b, c, and d.
-

4. Anisotropy Index and Influence Functions

This chapter establishes a unified analytical framework for evaluating geometric sensitivity and algorithmic robustness on \mathcal{P}_n . Section 4.1 introduces the anisotropy index as a key geometric descriptor, formally defining its relationship with fundamental metrics including the Euclidean metric, AIRM, and the Bregman divergences. Through variational optimization, we derive closed-form expressions for these indices, revealing their intrinsic connections to matrix spectral properties. Section 4.2 advances robustness analysis through the influence function theory, developing perturbation models for three central tensor means: TLD, TED, and TID. By quantifying sensitivity bounds under outlier contamination, we establish theoretical guarantees for each mean operator.

4.1. The Anisotropy Index Related to Some Metrics

The discriminatory capacity of weighted positive definite matrices manifests through their associated anisotropy measures. Defined intrinsically on the matrix manifold \mathcal{P}_n , the anisotropy index quantifies local geometric distortion relative to isotropic configurations. For $A \in \mathcal{P}_n$, the anisotropy measure relative to the Riemannian metric is

$$\alpha(A) := \inf_{\varepsilon > 0} d^2(A, \varepsilon I). \quad (45)$$

This index corresponds to the squared minimal projection distance from A to the scalar curvature subspace $\{\varepsilon I | \varepsilon \in \mathbb{R}^+\}$. Larger $\alpha(A)$ values indicate stronger anisotropic characteristics. For explicit computation, we minimize the metric-specific functional:

$$\phi_A(\varepsilon) = d^2(A, \varepsilon I), \quad \varepsilon > 0. \quad (46)$$

Next, we systematically investigate anisotropy indices induced by three fundamental geometries: the Frobenius metric representing Euclidean structure, AIRM characterizing curved manifold topology, and Bregman divergences rooted in information geometry. This trichotomy reveals how metric selection governs directional sensitivity analysis on the symmetric positive-definite matrix manifold.

Following Equations (3) and (45), the anisotropy index associated with the Euclidean metric can be derived through direct computation.

Proposition 7. *The anisotropy index according to Euclidean metric (3) at a point $A \in \mathcal{P}_n$ is given by*

$$a_F(A) = \text{tr}(A - \varepsilon^* I)^2, \quad (47)$$

where

$$\varepsilon^* = \frac{\text{tr}(A)}{n}. \quad (48)$$

Proposition 8. *The anisotropy index associated with AIRM at a point $A \in \mathcal{P}_n$ is formulated as*

$$a_{\text{AIRM}}(A) = \text{tr}(\text{Log}^2(\varepsilon^* A^{-1})), \quad (49)$$

where

$$\varepsilon^* = \sqrt[n]{\det A}. \quad (50)$$

Proof. Following (6) and (45), we have

$$\phi_A(\varepsilon) = d^2(A, \varepsilon I) = \text{tr}(\text{Log}^2(\varepsilon A^{-1})). \quad (51)$$

Then, differentiating $\phi_A(\varepsilon)$ with respect to ε yields the expression that

$$\phi'_A(\varepsilon) = \text{tr}\left(2\text{Log}(\varepsilon A^{-1})\frac{d}{d\varepsilon}\text{Log}(\varepsilon A^{-1})\right). \quad (52)$$

$$(53)$$

Let

$$\gamma(\varepsilon) = \varepsilon A^{-1}. \quad (54)$$

Applying Lemma 14 and 15 from Reference [24], we derive

$$\begin{aligned} \phi'_A(\varepsilon) &= \text{tr}\left(2\text{Log}(\gamma(\varepsilon))\frac{d}{d\varepsilon}\text{Log}(\gamma(\varepsilon))\right) \\ &= 2\text{tr}\left(\text{Log}(\gamma(\varepsilon))\int_0^1((\gamma(\varepsilon) - I)s + I)^{-1}\frac{d\gamma(\varepsilon)}{d\varepsilon}((\gamma(\varepsilon) - I)s + I)^{-1}ds\right) \\ &= 2\text{tr}\left(\gamma(\varepsilon)^{-1}\text{Log}(\gamma(\varepsilon)A^{-1})\right) \\ &= 2\text{tr}\left(\varepsilon^{-1}\text{Log}(\varepsilon A^{-1})\right). \end{aligned} \quad (55)$$

With the eigenvalue of A denoted as $\lambda_1, \lambda_2, \dots, \lambda_n$, (55) can be written as

$$\phi'_A(\varepsilon) = \frac{2}{\varepsilon}(n \ln \varepsilon - \ln(\lambda_1 \lambda_2 \dots \lambda_n)). \quad (56)$$

By solving $\phi'_A(\varepsilon) = 0$, the proof of equation (49) is complete. \square

Proposition 9. The anisotropy index according to the Bregman divergence (16) at a point $A \in \mathcal{P}_n$ is formulated as

$$a_L(A) = \text{tr}(A\text{Log}(A) - A\text{Log}(\varepsilon^* I) - (A - \varepsilon^* I)), \quad (57)$$

where

$$\varepsilon^* = \frac{\text{tr}(A)}{n}. \quad (58)$$

Proof. Via (16) and (45), we have

$$\phi_A(\varepsilon) = \text{tr}(A\text{Log}(A) - A\text{Log}(\varepsilon I) - A + \varepsilon I), \quad (59)$$

and

$$\phi'_A(\varepsilon) = \text{tr}(I - \frac{1}{\varepsilon}A). \quad (60)$$

By solving $\phi'_A(\varepsilon) = 0$, the proof of equation (57) is complete. \square

Proposition 10. The anisotropy index according to the Bregman divergence (21) at a point $A \in \mathcal{P}_n$ is formulated as

$$a_E(A) = \text{tr}(\text{Exp}(A) + \text{Exp}(\varepsilon^* I)((\varepsilon^* - 1)I - A)), \quad (61)$$

where

$$\varepsilon^* = \frac{\text{tr}(A)}{n}. \quad (62)$$

Proof. Via (21) and (45), we have

$$\phi_A(\varepsilon) = \text{tr}(\text{Exp}(A) + \text{Exp}(\varepsilon I)((\varepsilon - 1)I - A)), \quad (63)$$

and

$$\phi'_A(\varepsilon) = e^\varepsilon(n - \text{tr}(A)). \quad (64)$$

By solving $\phi'_A(\varepsilon) = 0$, the proof of equation (61) is complete. \square

Proposition 11. *The anisotropy index associated with the Bregman divergence (24) at a point $A \in \mathcal{P}_n$ is given by*

$$a_I(A) = \text{tr}(A^{-1} + \varepsilon^{*-2}A) - 2n\varepsilon^{*-1}, \quad (65)$$

where

$$\varepsilon^* = \frac{\text{tr}(A)}{n}. \quad (66)$$

Proof. Via (24) and (45), we have

$$\phi_A(\varepsilon) = \text{tr}(A^{-1} + \varepsilon^{-2}A) - 2n\varepsilon^{-1}, \quad (67)$$

and

$$\phi'_A(\varepsilon) = 2n\varepsilon^{-2} - 2\varepsilon^{-3}\text{tr}(A). \quad (68)$$

By solving $\phi'_A(\varepsilon) = 0$, the proof of equation (65) is complete. \square

4.2. Influence Functions

This subsection develops a robustness analysis framework through influence functions for symmetric positive-definite matrix-valued data. We systematically quantify the susceptibility of the TBD mean estimators under outlier contamination by deriving closed-form expressions of influence functions. Furthermore, we establish operator norm bounds of influence functions, thereby characterizing their stability margins in perturbed manifold learning scenarios.

Let \bar{A} denote the TBDs mean of m symmetry positive-definite matrices $\{A_1, A_2, \dots, A_m\}$. Let \hat{A} denote the mean after adding a set of l outliers $\{R_1, R_2, \dots, R_l\}$ with a weight $\varepsilon(0 < \varepsilon \ll 1)$ to $\{A_1, A_2, \dots, A_m\}$ [25,26]. Therefore, \hat{A} can be defined as follows:

$$\hat{A} = \bar{A} + \varepsilon F(\bar{A}, R_1, R_2, \dots, R_l) + O(\varepsilon^2), \quad (69)$$

which shows that \hat{A} is a perturbation of \bar{A} and $\|F\|$ is defined as the influence function. Let $\Psi(A)$ denote the objective function to be minimized over the $m + l$ symmetric positive-definite matrices, formulated as follows:

$$\Psi(A) := (1 - \varepsilon) \frac{1}{m} \sum_{i=1}^m \delta_\Phi(A, A_i) + \varepsilon \frac{1}{l} \sum_{j=1}^l \delta_\Phi(A, R_j). \quad (70)$$

Given that \hat{A} denotes the mean of $m + l$ symmetric positive-definite matrices, the optimality condition requires $\nabla \Psi(\hat{A}) = 0$, which gives:

$$(1 - \varepsilon) \frac{1}{m} \sum_{i=1}^m \frac{\nabla \Phi(\hat{A}) - \nabla \Phi(A_i)}{\sqrt{1 + \|\nabla \Phi(A_i)\|^2}} + \varepsilon \frac{1}{l} \sum_{j=1}^l \frac{\nabla \Phi(\hat{A}) - \nabla \Phi(R_j)}{\sqrt{1 + \|\nabla \Phi(R_j)\|^2}} = 0. \quad (71)$$

By taking the derivative of the equation $\nabla \Psi(\hat{A}) = 0$ with respect to ε and evaluating it at $\varepsilon = 0$, we obtain

$$\frac{1}{m} \sum_{i=1}^m \frac{1}{\sqrt{1 + \|\nabla \Phi(A_i)\|^2}} \frac{d}{d\varepsilon} \Big|_{\varepsilon=0} \nabla \Phi(\hat{A}) + \frac{1}{l} \sum_{j=1}^l \frac{\nabla \Phi(\bar{A}) - \nabla \Phi(R_j)}{\sqrt{1 + \|\nabla \Phi(R_j)\|^2}} = 0. \quad (72)$$

Next, the influence functions of TLD mean, TED mean, and TID mean are given by the following properties.

Proposition 12. TLD mean of m symmetry positive-definite matrices $\{A_1, A_2, \dots, A_m\}$ and l outliers $\{R_1, R_2, \dots, R_l\}$ is $\|F\|$ with

$$F = \left(\frac{1}{m} \sum_{i=1}^m \frac{\bar{A}^{-1}}{\sqrt{1 + \|\text{Log}(A_i) + I\|^2}} \right)^{-1} \left(\frac{1}{l} \sum_{j=1}^l \frac{\text{Log}(\bar{A}) - \text{Log}(R_j)}{\sqrt{1 + \|\text{Log}(R_j) + I\|^2}} \right). \quad (73)$$

Furthermore,

$$\|F\| \leq \|\bar{A}^{-1}\|^{-1} (\|\text{Log}(\bar{A}) + I\| + 1). \quad (74)$$

Proof. Following from (15), we derive

$$\text{tr} \left(\frac{d}{d\varepsilon} \Big|_{\varepsilon=0} \nabla \Phi(\hat{A}) \right) = \text{tr} \left(\frac{d}{d\varepsilon} \Big|_{\varepsilon=0} \text{Log} \hat{A} + I \right) = \text{tr}(\bar{A}^{-1} F) \quad (75)$$

Then, substituting (75) into (72) and computing the trace on both sides, we obtain

$$\text{tr} \left(\frac{1}{m} \sum_{i=1}^m \frac{\bar{A}^{-1} F}{\sqrt{1 + \|\text{Log} A_i + I\|^2}} + \frac{1}{l} \sum_{j=1}^l \frac{\text{Log} \bar{A} - \text{Log} R_j}{\sqrt{1 + \|\text{Log} R_j + I\|^2}} \right) = 0 \quad (76)$$

By considering the arbitrariness of \bar{A} , we derive (73) for the TLD mean. Furthermore, it can be deduced that the influence function $\|F\|$ has an upper bound (74), and is independent of outliers $\{R_1, R_2, \dots, R_l\}$. \square

Proposition 13. TED mean of m symmetric positive-definite matrices A_1, A_2, \dots, A_m and l outliers $\{R_1, R_2, \dots, R_l\}$ is $\|F\|$, where

$$F = \left(\frac{1}{m} \sum_{i=1}^m \frac{\text{Exp}(A)}{\sqrt{1 + \|\text{Exp}(A_i)\|^2}} \right)^{-1} \left(\frac{1}{l} \sum_{j=1}^l \frac{\text{Exp}(R_j) - \text{Exp}(\bar{A})}{\sqrt{1 + \|\text{Exp}(R_j)\|^2}} \right). \quad (77)$$

Furthermore,

$$\|F\| \leq \|\text{Exp}(\bar{A})\|^{-1} + 1. \quad (78)$$

Proof. Following from (20), we derive

$$\text{tr} \left(\frac{d}{d\varepsilon} \Big|_{\varepsilon=0} \nabla \Phi(\hat{A}) \right) = \text{tr} \left(\frac{d}{d\varepsilon} \Big|_{\varepsilon=0} \text{Exp}(\hat{A}) \right) = \text{tr}(\text{Exp}(\bar{A})). \quad (79)$$

Then, substituting (79) into (72) and computing the trace on both sides, we obtain

$$\text{tr} \left(\frac{1}{m} \sum_{i=1}^m \frac{\text{Exp}(\bar{A})}{\sqrt{1 + \|\text{Exp}(A_i)\|^2}} F + \frac{1}{l} \sum_{j=1}^l \frac{\text{Exp}(\bar{A}) - \text{Exp}(R_j)}{\sqrt{1 + \|\text{Exp}(R_j)\|^2}} \right) = 0. \quad (80)$$

By considering the arbitrariness of \bar{A} , we derive (77) for the TED mean. Furthermore, it can be deduced that the influence function $\|F\|$ has an upper bound (78), and is independent of outliers $\{R_1, R_2, \dots, R_l\}$. \square

Proposition 14. TID mean of m symmetry positive-definite matrices A_1, A_2, \dots, A_m and n outliers R_1, R_2, \dots, R_l is $\|F\|$, where

$$F = \left(\frac{1}{m} \sum_{i=1}^m \frac{2\bar{A}^{-3}}{\sqrt{1 + \|A_i^{-2}\|^2}} \right)^{-1} \cdot \left(\frac{1}{l} \sum_{j=1}^l \frac{\bar{A}^{-2} - R_j^{-2}}{\sqrt{1 + \|R_j^{-2}\|^2}} \right). \quad (81)$$

Furthermore,

$$\|F\| \leq \|2\bar{A}^{-3}\|^{-1}(\|\bar{A}^{-2}\| + 1). \quad (82)$$

Proof. Following from (23), we derive

$$\left. \frac{d}{d\varepsilon} \right|_{\varepsilon=0} \nabla \Phi(\hat{A}) = - \left. \frac{d}{d\varepsilon} \right|_{\varepsilon=0} (\hat{A}^{-1})(\hat{A}^{-1}) = \bar{A}^{-1}F\bar{A}^{-2} + \bar{A}^{-2}F\bar{A}^{-1}. \quad (83)$$

Then, substituting (83) into (72) and computing the trace on both sides, we obtain

$$\text{tr} \left(\frac{1}{m} \sum_{i=1}^m \frac{2 \cdot \bar{A}^{-3}F}{\sqrt{1 + \|A_i^{-2}\|^2}} + \frac{1}{l} \sum_{j=1}^l \frac{-\bar{A}^{-2} + R_j^{-2}}{\sqrt{1 + \|R_j^{-2}\|^2}} \right) = 0. \quad (84)$$

By considering the arbitrariness of \bar{A} , we get (81) for the TED mean. Furthermore, it can be deduced that the influence function $\|F\|$ has an upper bound (82), and is independent of outliers $\{R_1, R_2, \dots, R_l\}$. \square

While the influence function of the AIRM Mean demonstrates unboundedness with respect to its input matrices [24], all of the TBD means exhibit bounded influence functions under equivalent conditions.

5. Simulations and Analysis

In the following simulations, the symmetric positive-definite matrices used in Algorithm 1 are generated by

$$\text{Exp} \left(\frac{X + X^T}{2} \right), \quad (85)$$

where $X \in \mathbb{R}^{n \times n}$ is a square matrix randomly generated by MATLAB.

A. Simulations and Results

In this example, we apply Algorithm 1 to denoise the point cloud of a teapot, employing TLD as the distance metric. The signal-to-noise ratio (SNR) is specified as 4137:1000, with parameters $q = 5$ and $\varepsilon_0 = 0.001$. The experiments utilize MATLAB's built-in Teapot.ply dataset, a standard PLY-format 3D point cloud that serves as a benchmark resource for validating graphics processing algorithms and visualization techniques. To optimize indicator weights and enhance data visualization, we utilize Principal Component Analysis to provide a holistic view of the covariance matrix encompassing all data. Figure 2 (a) displays the initial distribution of valid data and noise prior to denoising. Following that, Figure 2 (b) exhibits the transformed data distribution after applying TLD for denoising. Figure 2 (c) and (d) respectively show the results before and after denoising the teapot. In these figures, blue dots signify noise data, whereas green dots indicate valid data. Figure 2 demonstrates that Algorithm 1 effectively partitions data points into two discrete clusters, achieving explicit separation between valid signals and noise components.

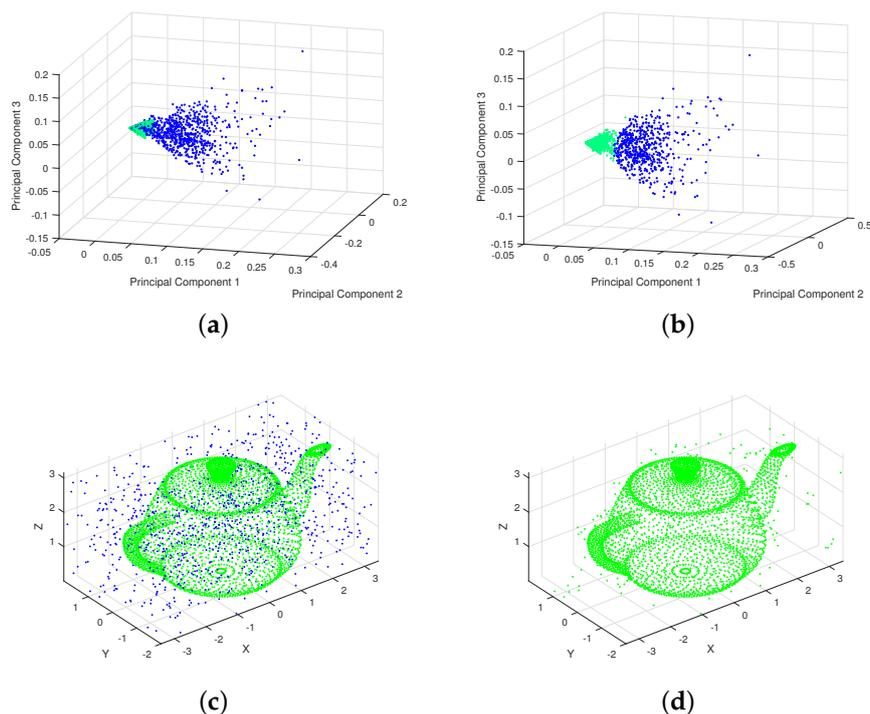


Figure 2. Denoised-raw point cloud contrast.

B. Analysis of TBDs Mean Stability

In Figure 3, for the 10 sample data randomly generated by (85) when $n = 4$, we use 100 experiments to compare the loss functions of the means for the Euclidean metric, TLD, TED, and TID. It is shown that the loss function of the mean for the Euclidean distance is much larger than those of the TBDs. Among the TBDs, the loss function of the TED mean is slightly larger than those of the TLD and TID means. A more minor loss function generally indicates that the algorithm is more stable during the training process, less prone to issues such as gradient explosion or vanishing gradients. This aids the algorithm in finding a better solution within a limited time. Therefore, this indicates that on \mathcal{P}_n , the means induced by TLD, TED, and TID are more suitable than those induced by the Euclidean metric.

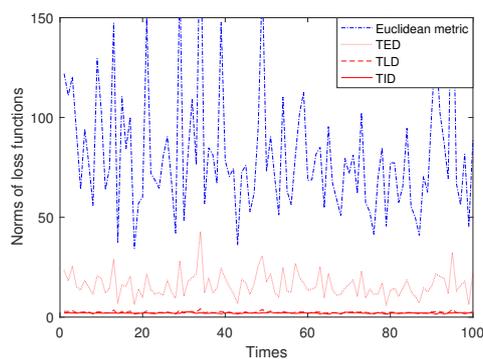


Figure 3. Comparison of loss functions.

From (29) and (35), it is shown that the TLD mean is essentially a weighted version of the LEM. Figure 4 compares the loss functions of the TLD mean and that of the LEM mean based on 100 repeated trials conducted on 50 samples generated using (85). As shown in Figure 4, the loss function of the TLD mean is significantly smaller compared to that of the LEM mean. Therefore, by assigning different weights to the elements of the matrix, the weighted mean further reduces the loss function. In practical applications such as image processing, signal analysis, and machine learning, using the weighted mean can be a valuable tool for achieving more accurate and efficient data processing and analysis.

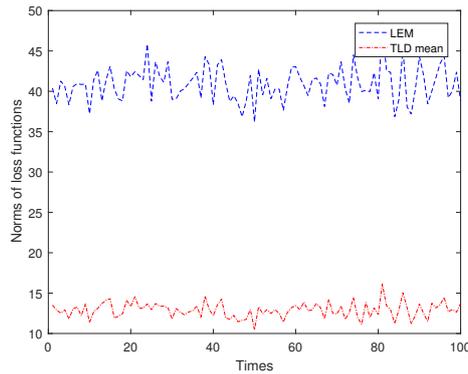


Figure 4. Comparison of the loss functions between LEM and TLD mean.

Table 1 benchmarks the denoising performance of Algorithm 1 across four metrics (Euclid, TLD, TED, TID), evaluated through True Positive Rate (TPR), False Positive Rate (FPR), and Signal-to-Noise Ratio Growth (SNRG) under three degradation levels (SNR=10/2/1). At SNR=10, all algorithms hit 100% TPR, but Euclid’s FPR (80.24%) is highest, making it less effective than TLD, TED, and TID. TLD and TED excel in SNRG, with TID leading at 112.82%. As SNR drops to 2, TED stands out with perfect TPR and high SNRG, followed by TLD. TID remains competitive despite TPR decline. Euclid lags in FPR and SNRG. At SNR=1, TED leads in TPR and SNRG, while TID tops FPR and SNRG among all. Euclid fails to match TLD, TED, and TID, especially in FPR minimization and SNRG maximization. In summary, Euclid underperforms TLD, TED, and TID in FPR and SNRG, highlighting their effectiveness in point cloud denoising.

Table 1. Comparison of denoising results.

Metric	SNR=10			SNR=2			SNR=1		
	TPR	FPR	SNRG	TPR	FPR	SNRG	TPR	FPR	SNRG
Euclid	100%	80.24%	24.62%	96.17%	69.14%	39.09%	92.38%	62.32%	48.24%
TLD	100%	51.81%	93.02%	99.54%	38.19%	160.67%	93.76%	38.86%	141.25%
TED	100%	57.59%	73.64%	100%	44.65%	123.97%	97.40%	47.52%	104.97%
TID	100%	46.99%	112.82%	90.24%	26.81%	236.60%	87.01%	27.41%	217.41%

C. Analysis of Computational Complexity

We first analyze the computational complexity of mean operators induced by four metrics: Euclidean, TLD, TED, and TID, mathematically defined in equations (28), (35), (37), and (39). Subsequently, we quantify the computational load for corresponding influence functions, with formulations specified in (73), (77), (81), and Proposition 1 of [17].

Under the framework of m symmetric positive-definite matrices and l outliers, we systematically evaluate computational costs. Considering single-element operations as $O(1)$ and matrix $A \in R^{n \times n}$, fundamental operations reveal critical patterns: Matrix inversion A^{-1} and logarithmic operations $\text{Log}(A)$ both require $O(n^3)$ computations. Operations involving matrix exponentials $\text{Exp}(A)$ and matrix roots for symmetric positive-definite matrices necessitate eigenvalue decomposition, consequently maintaining $O(n^3)$ complexity.

Detailed analysis reveals the arithmetic mean (28) exhibits $O(n^3)$ complexity. For influence functions, Euclidean-based estimation requires $O((m+l)n^3)$ operations. This establishes a critical trade-off: While Algorithm 1’s Euclidean metric demonstrates inferior denoising efficacy compared to TLD/TED/TID variants, its computational economy in mean matrix calculation $O(n^3)$ and influence function estimation $O((m+l)n^3)$ surpasses geometric counterparts. The complexity disparity stems from TLD/TED/TID’s intrinsic requirements for iterative matrix decompositions and manifold optimizations absent in Euclidean frameworks.

6. Conclusions

This study introduces a novel K-means clustering algorithm based on Total Bregman Divergence for robust point cloud denoising. Traditional Euclidean-based K-means methods often fail to address non-uniform noise distributions due to their limited geometric sensitivity. To overcome this, TBDs-Total Logarithm Divergence, Total Exponential Divergence, and Total Inverse Divergence - are proposed on the manifold of symmetric positive-definite matrices. These divergences are designed to model distinct local geometric structures, enabling more effective separation of noise from valid data. Theoretical contributions include the derivation of anisotropy indices to quantify structural variations and the analysis of influence functions, which demonstrate the bounded sensitivity of TBD-induced means to outliers.

Numerical experiments on synthetic and real-world datasets (e.g., 3D teapot point clouds) validate the algorithms excellence over Euclidean-based approaches. Results highlight improved noise separation, enhanced stability, and adaptability to complex noise patterns. The proposed framework bridges geometric insights from information geometry with practical clustering techniques, offering a scalable and robust preprocessing solution for applications in computer vision, robotics, and autonomous systems. This work underscores the potential of manifold-aware metrics in advancing point cloud processing and opens avenues for further exploration of divergence-based methods in high-dimensional data analysis.

Acknowledgments: This research was supported by the National Natural Science Foundation of China (No. 61401058), the General Program of Liaoning Natural Science Foundation (No. 2024-MS-166), and the Basic Scientific Research Project of the Liaoning Provincial Department of Education (No. JYTMS20230010).

Author Contributions: Investigation, Xiaomin Duan; Methodology, Xiaomin Duan and Anqi Mu; Software, Xinyu Zhao; Writing—original draft, Xiaomin Duan; Writing—review and editing, Xiaomin Duan, Anqi Mu, Xinyu Zhao, and Yuqi Wu.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Xiao, K.; Li, T.; Li, J.; Huang, D.; Peng, Y. Equal emphasis on data and network: A two-stage 3D point cloud object detection algorithm with feature alignment. *Remote Sens.* **2024**, *16*, 249. <https://doi.org/10.3390/rs16020249>
2. Zhang, S.; Zhu, Y.; Xiong, W.; Rong, X.; Zhang, J. Bridge substructure feature extraction based on the underwater sonar point cloud data. *Ocean Eng.* **2024**, *294*, 116700. <https://doi.org/10.1016/j.oceaneng.2024.116770>
3. Jiang, J.; Lu, X.; Ouyang, W.; Wang, M. Unsupervised contrastive learning with simple transformation for 3D point cloud data. *Vis. Comput.* **2024**, *40*, 5169-5186. <https://doi.org/10.1007/s00371-023-02921-y>
4. Akhavan, J.; Lyu, J.; Manoochehri, S. A deep learning solution for real-time quality assessment and control in additive manufacturing using point cloud data. *J. Intell. Manuf.* **2024**, *35*, 1389-1406. <https://doi.org/10.1007/s10845-023-02121-4>
5. Ran, C.; Zhang, X.; Han, S.; Yu, H.; Wang, S. TPDNet: a point cloud data denoising method for offshore drilling platforms and its application. *Measurement.* **2025**, *241*, 115671. <https://doi.org/10.1016/j.measurement.2024.115671>
6. Bao, Y.; Wen, Y.; Tang, C.; Sun, Z.; Meng, X.; Zhang, D.; Wang, L. Three-dimensional point cloud denoising for tunnel data by combining intensity and geometry information. *Sustainability.* **2024**, *16*, 2077. <https://doi.org/10.3390/su16052077>
7. Fu, X.; Zhang, G.; Kong, T.; Zhang, Y.; Jing, L.; Li, Y. A segmentation and topology denoising method for three-dimensional (3-D) point cloud data obtained from Laser scanning. *Lasers Eng.* **2024**, *57*, 191-207. <https://doi.org/10.36108/lasers/4232.70.0170>
8. Nurunnabi, A.; West, G.; Belton, D. Outlier detection and robust normal-curvature estimation in mobile laser scanning 3D point cloud data. *Pattern Recog.* **2015**, *48*, 1404-1419. <https://doi.org/10.1016/j.patcog.2014.10.014>

9. Zeng, J.; Cheung, G.; Ng, M.; Pang, J.; Yang, C. 3D point cloud denoising using graph Laplacian regularization of a low dimensional manifold model. *IEEE Trans. Image Process.* **2019**, *29*, 3474-3489. <https://doi.org/10.1109/TIP.2019.2961429>
10. Xu, X.; Geng, G.; Cao, X.; Li, K.; Zhou, M. TDNet: transformer-based network for point cloud denoising. *Appl. Opt.* **2022**, *61*, 80-88. <https://doi.org/10.1364/AO.438396>
11. Jain, A. K. Data clustering: 50 years beyond K-means. *Pattern Recognit. Lett.* **2010**, *31*, 651-666. <https://doi.org/10.1016/j.patrec.2009.09.011>
12. Maronna, R.; Aggarwal, C. C.; Reddy, C. K. Data clustering: algorithms and applications. *Statistical Papers.* **2015**, *57*, 565-566. <https://link.springer.com/article/10.1007/s00362-015-0661-7>
13. Zhu, Y.; Ting, K.; Carman, M. J. Density-ratio based clustering for discovering clusters with varying densities. *Pattern Recognit.* **2016**, *60*, 983-997. <https://doi.org/10.1016/j.patcog.2016.07.007>
14. Rodriguez, A.; Laio, A. Clustering by fast search and find of density peaks. *Science.* **2014**, *334*, 1494-1496. <https://doi.org/10.1126/science.1242072>
15. Sun, H.; Song, Y.; Luo, Y.; Sun, F. A Clustering Algorithm Based on Statistical Manifold. *Trans. Beijing Inst. Technol.* **2021**, *41*, 226-230. <https://dx.doi.org/10.15918/j.tbit1001-0645.2019.304>
16. Duan, X.; Ji, X.; Sun, H.; Guo, H. A non-iterative method for the difference of means on the Lie Group of symmetric positive-definite matrices. *Mathematics.* **2022**, *10*, 255. <https://doi.org/10.3390/math10020255>
17. Duan, X.; Feng, L.; Zhao, X. Point cloud denoising algorithm via geometric metrics on the statistical manifold. *Appl. Sci.* **2023**, *13*, 8264. <https://doi.org/10.3390/app13148264>
18. Liu, M.; Vemuri, B. C.; Amari, S. I.; Nielsen, F. Total Bregman divergence and its applications to shape retrieval. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition(CVPR), San Francisco, CA, USA, 2010, pp. 3463-3468. <https://doi.org/10.1109/CVPR.2010.5539979>
19. Yair, O.; Ben-Chen, M.; Talmon, R. Parallel transport on the cone manifold of SPD matrices for domain adaptation. *IEEE Trans. Signal Process.* **2019**, *67*, 1797-1811. <https://doi.org/10.1109/TSP.2019.2894801>
20. Luo, G.; Wei, J.; Hu, W.; Maybank, S. J. Tangent Fisher vector on matrix manifolds for action recognition. *IEEE Trans. Image Process.* **2020**, *29*, 3052-3064. <https://doi.org/10.1109/TIP.2019.2955561>
21. Dhillon, I. S.; Tropp, J. A. Matrix nearness problems with Bregman divergences. *SIAM J. Matrix Anal. Appl.* **2007**, *29*, 1120-1146. <https://doi.org/10.1137/060649021>
22. Liu, M.; Vemuri, B. C.; Amari, S. I.; Nielsen, F. Shape retrieval using hierarchical total Bregman soft clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2407-2419. <https://doi.org/10.1109/TPAMI.2012.44>
23. Amari, S. I.; Information Geometry and its Applications. *Springer*. Tokyo, Japan, **2016**.
24. Hua, X.; Ono, Y.; Peng, L.; Cheng, Y.; Wang, H. Target detection within nonhomogeneous clutter via total Bregman divergence-based matrix information geometry detectors. *IEEE Trans. Signal Process.* **2021**, *69*, 4326-4340. <https://doi.org/10.1109/TSP.2021.3095725>
25. Hua, X.; Peng, L.; Liu, W.; Cheng, Y.; Wang, H.; Sun, H.; Wang, Z. LDA-MIG detectors for maritime targets in nonhomogeneous sea clutter. *IEEE Trans. Geosci. Remote Sens.* **2023**, *16*, 5101815. <https://doi.org/10.1109/TGRS.2023.3250990>
26. Hua, X.; Cheng, Y.; Wang, H.; Qin, Y. Information geometry for covariance estimation in heterogeneous clutter with total Bregman divergence. *Entropy.* **2018**, *20*, 258. <https://doi.org/10.3390/e20040258>

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.