Article

Kernel Geometric Mean Metric Learning

Yu Zhou *, Zixin Feng , Teligeng Yun , Ruirui Zheng , Jianjun He

Posted Date: 25 July 2023

doi: 10.20944/preprints202307.1650.v1

Keywords: Metric learning; Kernel methods; Weighted geometric mean

Article

Kernel Geometric Mean Metric Learning

Zixin Feng, Teligeng Yun, Yu Zhou *, Ruirui Zheng, Jianjun He

Dalian Minzu University

* Correspondence: yuzhou829@sina.com

Abstract: This paper propose a kernel geometric mean metric learning (KGMML) algorithm. The basic idea is to obtain the closed-form solution of the geometric mean metric learning (GMML) algorithm in the high-dimensional feature space determined by the kernel function. Then, the solution is generalized as a form of kernel matrix by using the integral representation of the weighted geometric mean and the Woodbury matrix in this new feature space. Experimental results on 15 datasets show that the proposed algorithm can effectively improve the accuracy of the GMML algorithm and other metric algorithms.

Keywords: metric learning; kernel methods; weighted geometric mean

1. Introduction

Analyzing the modeling process of machine learning algorithms, it is clear that the construction of a learning algorithm requires a similarity metric between sample pairs given. It is well known that the distance measure is one of the most commonly used measures to describe the similarity between samples. At present various distance metrics have been proposed such as Euclidean distance and Mahalanobis distance. However, these distance metric expressions are fixed, i.e., there are non-adjustable parameters, which result in different effectiveness of dealing with various problems. Thus, an effective distance metric is proposed by constructing learning from training samples. From the definition of the distance metric, it follows that any binary function $d\left(x_i, x_j\right)$ defined in the feature space is called a distance function, provided that the four conditions of symmetry, self-similarity, non-negativity, and trigonometric inequality are satisfied simultaneously. Thus any binary function

$$d_M\left(x_i, x_j\right) = \left(x_i - x_j\right)^{\mathrm{T}} M\left(x_i - x_j\right), \tag{1}$$

is a distance function determined by any symmetric positive definite (SPD) matrix M, where x_i, x_j are two samples from the training set $\mathcal{X}$, and usually M is called a metric matrix. The purpose of metric learning is to use training samples to learn a metric matrix M such that the resulting distance function $d_M\left(x_i, x_j\right)$ can improve the performance of the learning algorithm or satisfy some application requirements. Thus, metric learning has wide applications in many fields, such as pattern recognition [1,2], data mining [3–5], information security [6,7], bioinformatics [8,9], and medical diagnosis [10–12].

Because of the wide application space, metric learning techniques have received a lot of attention and many excellent algorithms have been proposed methods. Xing [13] first proposed a metric learning algorithm. The main idea of the algorithm is to learn a metric matrix so that the distance between similar pairs of samples is small and the distance between dissimilar pairs of samples is large. The algorithm can make the distribution of similar pairs of samples in the new metric space more compact more compact in the new metric space, while the distribution of dissimilar pairs is more discrete. The proposal of this algorithm marked the real development of metric learning, and many subsequent research works were inspired by this algorithm. Davis [14] proposed an information theoretic metric learning algorithm. The basic idea of the algorithm is to assume the existence of an a priori metric matrix M_0, while ensuring that the distance between similar pairs of samples is less than a threshold and the distance between dissimilar pairs of samples is greater than a threshold. Minimize the relative entropy between the multivariate Gaussian distributions corresponding to M and M_0. Wang [15] proposed an information geometry metric learning algorithm. The basic idea of the algorithm is to use the category labeling information of the training samples to construct a kernel matrix that can reflect the desired distance between samples. Construct an actual kernel matrix describing the realistic distance relationship between the samples using the eigenvectors of the samples as well as the metric matrix M. The metric matrix M is then solved by minimizing the distance between these two kernel matrices. Weinberger [16] proposed a metric learning algorithm based on maximum margin. The basic idea is to define an interval function between similar samples of each sample and other classes of samples. The metric matrix is then solved by minimizing the distance between similar pairs of samples and maximizing the defined interval.

Geometric mean metric learning algorithm was proposed by Pourya [17] in 2016. The essence of most metric learning algorithms is to minimize the distance between similar pairs of samples rather than maximize the distance between similar pairs of samples, therefore the sign of the term corresponding to the dissimilar pair of samples in the objective function is usually negative, while the strategy of the geometric mean metric learning model is to use the inverse matrix of the metric matrix M to represent the distance between dissimilar pairs of samples. The advantage of this approach is that the sign of the item corresponding to the dissimilar pair of samples in the objective function will become positive, which reduces the difficulty of solving the model. Its objective function is as follows:

$$\min_{M\succ 0} \sum_{(x_i,x_j)\in D^+} d_M\left(x_i,x_j\right) + \sum_{(x_i,x_j)\in D^-} d_{M^{-1}}\left(x_i,x_j\right), \tag{2}$$

where the similar pair sample set D^+ and the dissimilar pair sample set D^- can be expressed as:.

$$\begin{cases} D^+ = \left\{\left(x_i,x_j\right) \mid x_i,x_j \text{ are in the same class}\right\}, \\ D^- = \left\{\left(x_i,x_j\right) \mid x_i,x_j \text{ are in different class}\right\}. \end{cases} \tag{3}$$

Although GMML algorithm has some advantages, such as unconstrained convex objective function, closed form solution and interpretability, and faster calculation speed [18], it is actually a linear learning method and does not work well to address non-linear problems. Kernel methods are key technology for addressing nonlinear problems, and therefore kernel algorithms [19–23] have been proposed. The principle of the kernel method is that the original data transformed from the input space into a higher dimensional feature space by a mapping function. This transformation must be achieved to offer a reliable linear model in the feature space that corresponds to a nonlinear solution in the original data space. Gaussian kernel functions were used to map the data in the feature space. The closed form solution of the GMML algorithm is represented as a kernel matrix by using the integral representation of the weighted geometric mean and the Woodbury matrix. Then the KGMML algorithm is obtained. Thus the nonlinear problem can be handled effectively while retaining the advantages of the GMML algorithm. We define the objective function as the following form:

$$\min_{M_\Phi\succ 0} \sum_{\left(\Phi(x_i),\Phi(x_j)\right)\in D_\Phi^+} d_{M_\Phi}\left(\Phi(x_i),\Phi(x_j)\right) + \sum_{\left((\Phi(x_i)-\Phi(x_j)\right)\in D_\Phi^-} d_{{M_\Phi}^{-1}}\left(\Phi(x_i),\Phi(x_j)\right), \tag{4}$$

where a mapping is $\Phi : \mathbb{R}^m \mapsto \mathcal{H}_\kappa$ (Hilbert Space), i.e., $x \in \mathbb{R}^m \to \Phi(x) \in \mathcal{H}_\kappa$, M_Φ is the metric matrix in $\mathcal{H}_\kappa$ space, the set of similar pairs D_Φ^+ and the set of dissimilar pairs D_Φ^- are

$$\begin{cases} D_\Phi^+ = \left\{\left(\Phi(x_i),\Phi(x_j)\right) \mid \Phi(x_i),\Phi(x_j) \text{are in the same class}\right\}, \\ D_\Phi^- = \left\{\left(\Phi(x_i),\Phi(x_j)\right) \mid \Phi(x_i),\Phi(x_j) \text{are in different class}\right\}. \end{cases} \tag{5}$$

The main innovation of the paper is to study kernel geometric mean metric learning algorithm for nonlinear distance using a kernel function. The key idea is to construct kernel matrices based on the distance metric for the given training data. Secondly, the accuracy of the proposed algorithm is superior to GMML algorithm and other metric algorithms. The structure of this paper is organized as follows: In Sect. 2, some lemmas of weighted geometric mean and Woodbury identity are formulated.

doi:10.20944/preprints202307.1650.v1

Then, in Sect. 3, optimization problem and its solution were discuss, then extension to weighted geometric mean is discussed. The steup of the experiment and the analysis of parameter sensitivity are carried out in Sect. 4. Finally, our results are summarized in Sect. 5.

2. Preliminaries

In this section, three lemmas will be given in order to simplify the objective function (4) more clearly.

Lemma 1. *[24] For any $t \in (0,1)$, A is $n \times n$ positive definite and B is $n \times n$ Hermitian, the following equation holds*

$$A \sharp_t B = \frac{2\sin(\pi t)}{\pi} A \int_{-1}^{1} (1-s)^{-t}(1+s)^{t-1} \left((1-s)I + (1+s)B^{-1}A\right)^{-1} ds,$$

which is the integral representation of the A, B weighted geometric mean, where I is the identity matrix.

Lemma 2. *[25] If A is a $n \times n$ invertible matrix corrected by UCV where U is $n \times k$ matrice, C is $k \times k$ matrice, V is $k \times n$ matrice, then Woodbury identity is*

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U\left(C^{-1} + VA^{-1}U\right)^{-1} VA^{-1}.$$

Lemma 3. *[24] For any $t \in (0,1)$, A is $n \times n$ positive definite and B is $n \times n$ Hermitian, the following equation holds*

$$A \sharp_t B = A\left(B^{-1}A\right)^{-t}.$$

3. Main Results

3.1. Optimization Problem and Its Solution

In the following, the objective function (4) can be simplified in view of Equation (1), one has

$$\begin{aligned} &\min_{M_\Phi > 0} \sum_{(\Phi(x_i),\Phi(x_j)) \in D_\Phi^+} \left(\Phi(x_i) - \Phi(x_j)\right)^{\mathrm{T}} M_\Phi \left(\Phi(x_i) - \Phi(x_j)\right) \\ &+ \sum_{(\Phi(x_i),\Phi(x_j)) \in D_\Phi^-} \left((\Phi(x_i) - \Phi(x_j)\right)^{\mathrm{T}} M_\Phi^{-1} \left((\Phi(x_i) - \Phi(x_j)\right). \end{aligned} \tag{6}$$

Rewriting the Mahalanobis distance uses traces, Equation (4) can be turned into the following optimization problem

$$\begin{aligned} &\min_{M_\Phi > 0} \sum_{(\Phi(x_i),\Phi(x_j)) \in D_\Phi^+} \mathrm{tr}\left(M_\Phi \left(\Phi(x_i) - \Phi(x_j)\right)\left(\Phi(x_i) - \Phi(x_j)\right)^{\mathrm{T}}\right) \\ &+ \sum_{(\Phi(x_i),\Phi(x_j)) \in D_\Phi^-} \mathrm{tr}\left(M_\Phi^{-1} \left(\Phi(x_i) - \Phi(x_j)\right)\left(\Phi(x_i) - \Phi(x_j)\right)^{\mathrm{T}}\right) \\ = &\min_{M_\Phi > 0} \mathrm{tr}(M_\Phi S_\Phi) + \mathrm{tr}\left(M_\Phi^{-1} D_\Phi\right) \\ \overset{\triangle}{=} &\min_{M_\Phi > 0} h(M_\Phi), \end{aligned} \tag{7}$$

where S_Φ , D_Φ are SPD matrixes, and it is a realistic assumption in many situations [17]. Thus S_Φ , D_Φ can be expressed as

$$\begin{aligned} S_\Phi &= \sum_{\left(\Phi(x_i),\Phi(x_j)\right)\in D_\Phi^+} \left(\Phi(x_i)-\Phi(x_j)\right)\left(\Phi(x_i)-\Phi(x_j)\right)^{\mathrm{T}}, \\ D_\Phi &= \sum_{\left(\Phi(x_i),\Phi(x_j)\right)\in D_\Phi^-} \left(\Phi(x_i)-\Phi(x_j)\right)\left(\Phi(x_i)-\Phi(x_j)\right)^{\mathrm{T}}. \end{aligned} \tag{8}$$

Differentiating $h(M_\Phi)$ with respect to M_Φ yields

$$\nabla h\left(M_\Phi\right) = S_\Phi - M_\Phi^{-1} D_\Phi M_\Phi^{-1}. \tag{9}$$

Then, $\nabla h(M)$ is set to 0, which implies that

$$M_\Phi S_\Phi M_\Phi = D_\Phi, \tag{10}$$

it is clear that the above equation is a Riccati equation. Since S_Φ and D_Φ are positive definite matrices, Equation (4) has a unique positive solution which is the midpoint of the geodesic joining ${S_\Phi}^{-1}$ to D_Φ [26], that is

$$M_\Phi = {S_\Phi}^{-1}\sharp_{1/2} D_\Phi = {S_\Phi}^{-1/2}\left({S_\Phi}^{1/2} D_\Phi {S_\Phi}^{1/2}\right)^{1/2} {S_\Phi}^{-1/2}, \tag{11}$$

where ${S_\Phi}^{-1}\sharp_{1/2} D_\Phi$ represents the geometric mean of ${S_\Phi}^{-1}$ and D_Φ.

3.2. Extension to Weighted Geometric Mean

In order to incorporate the weighted geometric mean, it is necessary to take into account the determination of weights for the objective function. When assigning linear weights for ${S_\Phi}^{-1}$ and D_Φ, only the metric matrix M_Φ can be uniformly scaled by a constant factor. Consequently, it is illogical to assign linear weights to the two components in Equation (2). Nevertheless, by employing nonlinear weights derived from SPD manifold Riemannian geometry, the weights can be transformed into trade-offs between the two terms. Thus, the Riemann distance δ_R is introduced, and the problem of finding the minimum value of $h(M_\Phi)$ is equivalent to solving the minimum value of the following optimization problem

$$\min_{M_\Phi \succeq 0} \delta_R^2\left(M_\Phi, {S_\Phi}^{-1}\right) + \delta_R^2\left(M_\Phi, D_\Phi\right), \tag{12}$$

where the Riemannian distance between SPD matrices X and Y is denoted by $\delta_R^2(X,Y) = \left\|\log\left(Y^{-1/2} X Y^{-1/2}\right)\right\|_F$, and $\|.\|_F$ is the Frobenius norm of a matrix. A linear parameter $t \in [0,1]$ is introduced to trade off the relationship of the two terms in the formula

$$\min_{M_\Phi \succeq 0} (1-t)\delta_R^2\left(M_\Phi, {S_\Phi}^{-1}\right) + t\delta_R^2\left(M_\Phi, D_\Phi\right) \triangleq \min_{M_\Phi > 0} h_t(M_\Phi). \tag{13}$$

Because $h_t(M_\Phi)$ is still geodesically convex (see [17]), Equation (13) has a unique positive solution, that is

$$M_\Phi = {S_\Phi}^{-1}\sharp_t D_\Phi. \tag{14}$$

For the convenience of subsequent calculations, the notations are defined as follow

$$\begin{aligned} P_\Phi &= \sum_{\left(\Phi(x_i),\Phi(x_j)\right)\in D_\Phi^+} \Phi\left(x_i\right) - \Phi\left(x_j\right), \\ Q_\Phi &= \sum_{\left(\Phi(x_i),\Phi(x_j)\right)\in D_\Phi^-} \Phi\left(x_i\right) - \Phi\left(x_j\right). \end{aligned}$$

Theorem 1. *The solution $M_\Phi = {S_\Phi}^{-1}\sharp_t D_\Phi$ of the objective function (4) can be rewritten as*

$$M_\Phi = Q_\Phi \left({K_{PQ}}^{\mathrm{T}} K_{PQ}\right)^{t-1} {Q_\Phi}^{\mathrm{T}},$$

where the kernel matrix $K_{PQ} = \langle P_\Phi \cdot Q_\Phi \rangle = P_\Phi^{\mathrm{T}} Q_\Phi$, and $\langle \cdot \rangle$ is denoted inner product.

Proof. According to Lemma 1

$$\begin{aligned} M_\Phi &= S_\Phi^{-1}\sharp_t D_\Phi \\ &= \frac{2sin(\pi t)}{\pi} S_\Phi^{-1} \int_{-1}^{1} (1-s)^{-t}(1+s)^{t-1}((1-s)I + (1+s)D_\Phi^{-1}S_\Phi^{-1})^{-1}\mathrm{d}s \\ &= \frac{2sin(\pi t)}{\pi} \int_{-1}^{1} (1-s)^{-t}(1+s)^{t-1}((1-s)S_\Phi + (1+s)D_\Phi^{-1})^{-1}\mathrm{d}s. \end{aligned} \tag{15}$$

Substituting $S_\Phi = P_\Phi P_\Phi^{\mathrm{T}}, D_\Phi = Q_\Phi Q_\Phi^{\mathrm{T}}$ into Equation (15), it is clearly that

$$M_\Phi = \frac{2sin(\pi t)}{\pi} \int_{-1}^{1} (1-s)^{-t}(1+s)^{t-1}((1-s)P_\Phi P_\Phi^{\mathrm{T}} + (1+s)(Q_\Phi Q_\Phi^{\mathrm{T}})^{-1})^{-1}\mathrm{d}s. \tag{16}$$

For convenience in the following discussion, we introduce the notation

$$G := \left((1-s)\,P_\Phi P_\Phi^{\mathrm{T}} + (1+s)\left(Q_\Phi Q_\Phi^{\mathrm{T}}\right)^{-1}\right)^{-1}.$$

From Lemma 2, it follows that

$$G = (A+UCV)^{-1} = A^{-1} - A^{-1}U\left(C^{-1} + VA^{-1}U\right)^{-1} VA^{-1}, \tag{17}$$

where $A := (1+s)\left(Q_\Phi Q_\Phi^{\mathrm{T}}\right)^{-1}$, $U := P_\Phi$, $C := (1-s)I$, $V := P_\Phi^{\mathrm{T}}$. Then, taking it into Equation (16), one has

$$\begin{aligned} M_\Phi &= \frac{2sin(\pi t)}{\pi} \int_{-1}^{1} (1-s)^{-t}(1+s)^{t-1}[\frac{Q_\Phi Q_\Phi^{\mathrm{T}}}{1+s} - \frac{Q_\Phi Q_\Phi^{\mathrm{T}}}{1+s} \cdot P_\Phi \left(\frac{I^{-1}}{1-s} + P_\Phi^{\mathrm{T}} \cdot \frac{Q_\Phi Q_\Phi^{\mathrm{T}}}{1+s} \cdot P_\Phi\right)^{-1} P_\Phi^{\mathrm{T}} \frac{Q_\Phi Q_\Phi^{\mathrm{T}}}{1+s}]\mathrm{d}s. \\ &= \frac{2sin(\pi t)}{\pi} \int_{-1}^{1} (1-s)^{-t}(1+s)^{t-1}[\frac{Q_\Phi Q_\Phi^{\mathrm{T}}}{1+s} - \frac{Q_\Phi}{1+s} \cdot {K_{PQ}}^{\mathrm{T}} \left(\frac{I^{-1}}{1-s} + \frac{K_{PQ}{K_{PQ}}^{\mathrm{T}}}{1+s}\right)^{-1} \frac{K_{PQ}{Q_\Phi}^{\mathrm{T}}}{1+s}]\mathrm{d}s \\ &= \frac{2sin(\pi t)}{\pi} \int_{-1}^{1} (1-s)^{-t}(1+s)^{t-1} Q_\Phi[\frac{I}{1+s} - \frac{{K_{PQ}}^{\mathrm{T}}}{1+s} \left(\frac{I^{-1}}{1-s} + \frac{K_{PQ}{K_{PQ}}^{\mathrm{T}}}{1+s}\right)^{-1} \frac{K_{PQ}}{1+s}]Q_\Phi^{\mathrm{T}}\mathrm{d}s, \end{aligned} \tag{18}$$

where $K_{PQ} = P_\Phi^{\mathrm{T}} Q_\Phi$, and set

$$\mathcal{G} := \frac{I}{1+s} - \frac{{K_{PQ}}^{\mathrm{T}}}{1+s}\left(\frac{I^{-1}}{1-s} + \frac{K_{PQ}{K_{PQ}}^{\mathrm{T}}}{1+s}\right)^{-1} \frac{K_{PQ}}{1+s}.$$

In view of Lemma 2

$$\mathcal{G} = \mathcal{A}^{-1} - \mathcal{A}^{-1}\mathcal{U}\left(\mathcal{C}^{-1} + \mathcal{V}\mathcal{A}^{-1}\mathcal{U}\right)^{-1} \mathcal{V}\mathcal{A}^{-1} = (\mathcal{A} + \mathcal{U}\mathcal{C}\mathcal{V})^{-1}, \tag{19}$$

doi:10.20944/preprints202307.1650.v1

where $\mathcal{A} := [(1+s)I], \mathcal{U} := K_{PQ}^{\mathrm{T}}, \mathcal{C} := (1-s)I, \mathcal{V} := K_{PQ}$. Thus,

$$\begin{aligned} M_{\Phi} =& \frac{2sin(\pi t)}{\pi} \int_{-1}^{1} (1-s)^{-t}(1+s)^{t-1} Q_{\Phi}((1+s)I + (1-s)K_{PQ}{}^{\mathrm{T}}K_{PQ})^{-1} Q_{\Phi}^{\mathrm{T}} \mathrm{d}s \\ =& Q_{\Phi} \frac{2sin(\pi t)}{\pi} \int_{-1}^{1} (1-s)^{-t}(1+s)^{t-1} \left((1+s) \left(K_{PQ}{}^{\mathrm{T}}K_{PQ} \right)^{-1} \left(K_{PQ}{}^{\mathrm{T}}K_{PQ} \right) + (1-s) \left(K_{PQ}{}^{\mathrm{T}}K_{PQ} \right) \right)^{-1} \mathrm{d}s Q_{\Phi}^{\mathrm{T}} \\ =& Q_{\Phi} \left(K_{PQ}{}^{\mathrm{T}}K_{PQ} \right)^{-1} \frac{2sin(\pi t)}{\pi} \int_{-1}^{1} (1-s)^{-t}(1+s)^{t-1} \left((1-s)\, I + (1+s) \left(K_{PQ}{}^{\mathrm{T}}K_{PQ} \right)^{-1} \right)^{-1} \mathrm{d}s Q_{\Phi}^{\mathrm{T}} \\ =& Q_{\Phi} \left(K_{PQ}{}^{\mathrm{T}}K_{PQ} \right)^{-1} I \sharp_t (K_{PQ}{}^{\mathrm{T}}K_{PQ}) Q_{\Phi}^{\mathrm{T}}. \end{aligned} \tag{20}$$

From Lemma 3, it follows that

$$\begin{aligned} M_{\Phi} =& Q_{\Phi} \left(K_{PQ}{}^{\mathrm{T}}K_{PQ} \right)^{-1} \left(\left(K_{PQ}{}^{\mathrm{T}}K_{PQ} \right)^{-1} \right)^{-t} Q_{\Phi}^{\mathrm{T}} \\ =& Q_{\Phi} \left(K_{PQ}{}^{\mathrm{T}}K_{PQ} \right)^{t-1} Q_{\Phi}{}^{\mathrm{T}}. \end{aligned} \tag{21}$$

□

From the definition of the kernel matrix yields

$$\begin{aligned} K_{iQ} &= \langle \Phi(x_i) \cdot Q_{\Phi} \rangle , \\ K_{jQ} &= \langle \Phi(x_j) \cdot Q_{\Phi} \rangle . \end{aligned} \tag{22}$$

Theorem 2. *The distance $d_{M_{\Phi}}(x_i, x_j) = (\Phi(x_i) - \Phi(x_j))^{\mathrm{T}} M_{\Phi} (\Phi(x_i) - \Phi(x_j))$ can be rewritten as*

$$d_{M_{\Phi}}(x_i, x_j) = (K_{iQ} - K_{jQ}) \left(K_{PQ}{}^{T}K_{PQ} \right)^{t-1} (K_{iQ}{}^{\mathrm{T}} - K_{jQ}{}^{\mathrm{T}}).$$

Proof. According to Theorem 1

$$\begin{aligned} d_{M_{\Phi}}(x_i, x_j) =& (\Phi(x_i) - \Phi(x_j))^{\mathrm{T}} Q_{\Phi} \left(K_{PQ}{}^{\mathrm{T}}K_{PQ} \right)^{t-1} Q_{\Phi}^{\mathrm{T}} (\Phi(x_i) - \Phi(x_j)) \\ =& (K_{iQ} - K_{jQ}) \left(K_{PQ}{}^{\mathrm{T}}K_{PQ} \right)^{t-1} (K_{iQ}{}^{\mathrm{T}} - K_{jQ}{}^{\mathrm{T}}). \end{aligned} \tag{23}$$

□

Equation (23) can be taken as the final result of kernel geometric mean metric learning algorithm. The algorithm is summarized in Algorithm 1.

Algorithm 1 Kernel Geometric Mean Metric Learning Algorithm

```
Input: Training set $\mathcal{S} = \{(x_i, x_j) | x_i, x_j \in \mathcal{X}\}$.
Parameter: $t$: $t \in [0, 1]$, the weight coefficient in Equation (14),
$p$: kernel parameters of Gaussian kernel function.
Output: $d_{M_{\Phi}}$ the distance learned for KGMML,
Step1. According to Equation (5), construct $D_{\Phi}^{+}$ and $D_{\Phi}^{-}$.
Step2. Compute kernel matrices $K_{PQ}$ , $K_{iQ}$ and $K_{jQ}$ according to Theorem 1 and Theorem 2.
Step3. Compute $d_{M_{\Phi}}(x_i, x_j) = (K_{iQ} - K_{jQ}) \left( K_{PQ}{}^{\mathrm{T}}K_{PQ} \right)^{t-1} (K_{iQ}{}^{\mathrm{T}} - K_{jQ}{}^{\mathrm{T}})$.
```

4. Experiment

4.1. Experimental setup

To verify the effectiveness of algorithm 1, simulation experiments will be conducted on 15 UCI [27] datasets, where the basic information is shown in Table 1.

Table 1. Characteristics of experimental datasets.

	Data sets	Of features	Of instances	Of classes
1	Pima	8	768	2
2	Vehicle	18	846	4
3	German	24	1000	2
4	Segment	18	2310	7
5	Usps	256	9298	10
6	Mnist	784	4000	10
7	Glasses	9	214	6
8	DNA	180	3186	2
9	Heart-Disease	13	270	2
10	Lymphgraphy	18	148	4
11	Liver-Disorders	6	345	2
12	Hages-Roth	4	160	3
13	Ionosphere	34	351	2
14	Spambase	57	4601	2
15	Balance-Scale	4	625	3

Next, some efficient algorithms are described for distance metric learning, and the proposed method is compared with existing excellent classical algorithms. The specific introduction will be given in Table 2. For the KGMML, the setting of t and p will be given in detail in the next subsection.

Table 2. Briefly describes the distance metric learning method used in this paper

	Name	Description
1	Euclidean	The Euclidean distance metric [28].
2	DMLMJ	Distance metric learning through maximization of the Jeffrey divergence [28].
3	LMNN	Large margin nearest neighbor classification [16].
4	GB-LMNN	Non-linear Transformations with Gradient Boosting [29].
5	GMML	Geometric Mean Metric Learning [17].
6	Low-rank	Low-rank geometric mean metric learning [30].
7	KGMML	The kernelized version of GMML

4.2. Parameter Sensitivity Analysis

It can be seen from Algorithm1 that the values of t and p are specified before use. To determine the impact of these two parameters on the KGMML algorithm, the experiments are conducted on five datasets selected from 15 datasets. 5-fold cross-validation is used to choose the best t-value, and the two-step method is used to test different t-values. The above approach is used to find the best t-value, and its precision can be verified in Figures 1 and 2. Firstly, obtain the optimal t in the set $\{0.1, 0.3, 0.5, 0.7, 0.9\}$, and the result is shown in Figure 1. Secondly, test 5 t-values using intervals with a step size of 0.02. As shown in Figure 2, the variation of accuracy in the test interval with a step size of 0.02 is not significant, so we chose the middle t-value of 0.05. When the p-value is 10, the precision has an inflection point in Figure 3, and the precision is higher at the inflection point. Thus, the p- value is chosen as 10. Vary t in the set $\{0.1, 0.3, 0.5, 0.7, 0.9\}$, and p is fixed.

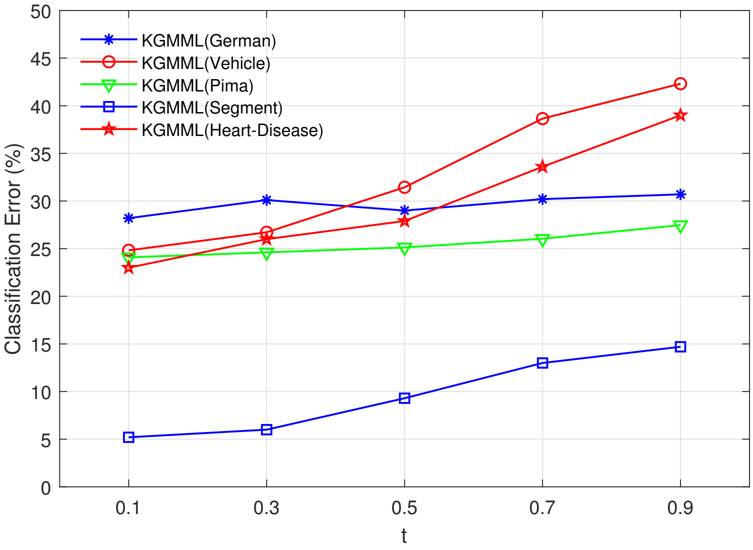

Figure 1. Vary t in the set $\{0.1, 0.3, 0.5, 0.7, 0.9\}$, and p is fixed.

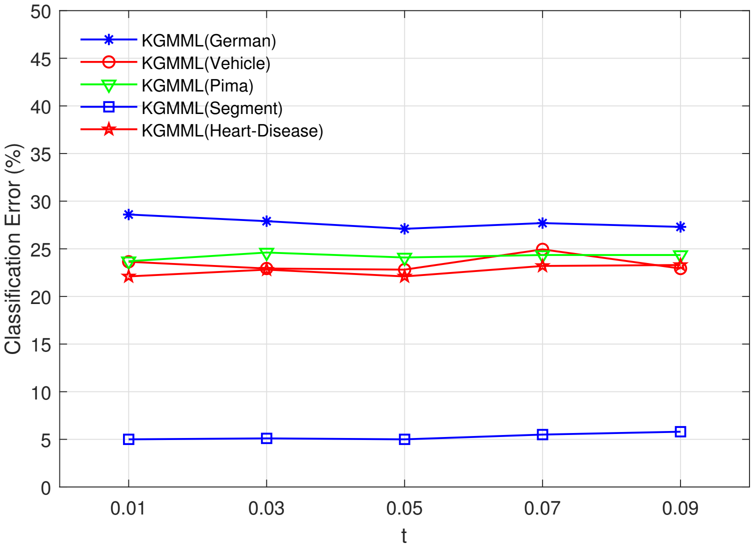

Figure 2. Vary t in the set $\{0.01, 0.03, 0.05, 0.07, 0.09\}$, and p is fixed

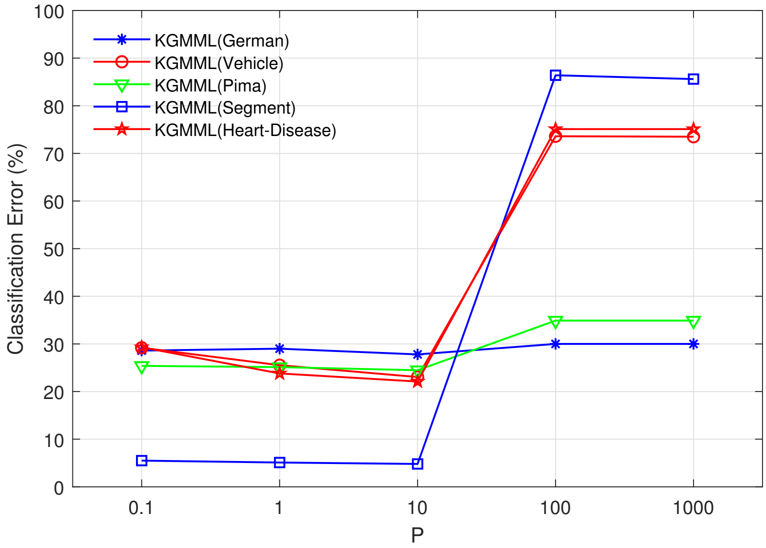

Figure 3. Vary p in the set $\{0.01, 1, 10, 100, 1000\}$, and t is fixed

4.3. Experimental Results

The accuracy of each compared algorithm on 15 datasets is shown in Table 3, where the best result on each dataset is shown in boldface. Compared with the other six algorithms, the highest accuracy is achieved from KGMML algorithm on the 8 datasets. All experimental methods are implemented on MATLABR2018b (64-bit), and the simulations are run on a laptop with an Intel Core i5 (2.5GHz) processor.

Table 3. Error rate results are shown on UCI dataset, and the best result is shown in bold.

	Data sets	GMML	DMLMJ	LMNN	GB-LMNN	Educlidean	Low-rank	KGMML
1	Pima	27.66	30.18	33.82	37.14	27.27	29.58	**25.17**
2	Vehicle	22.09	25.75	46.53	41.24	33.53	41.32	**21.21**
3	German	27.41	24.79	30.50	29.32	31.53	26.96	**24.40**
4	Segment	4.13	3.64	5.19	4.55	6.93	5.59	**3.22**
5	Usps	3.72	2.88	33.11	10.60	10.55	4.08	**2.82**
6	Mnist	9.65	16.44	86.80	82.62	17.12	75.34	**8.32**
7	Glasses	36.96	33.08	30.20	**23.30**	30.23	33.64	32.47
8	DNA	23.65	**21.75**	22.32	22.84	27.63	26.24	23.05
9	Heart-Disease	20.82	19.73	31.52	**18.51**	33.33	22.52	18.97
10	Lymphography	56.88	73.62	70.76	60.21	75.13	57.57	**53.75**
11	Liver-Disorders	35.00	**30.05**	30.46	34.88	31.88	41.86	30.17
12	Hages-Roth	37.69	16.84	31.33	31.35	**16.67**	39.55	19.52
13	Ionosphere	15.34	11.27	5.71	4.29	**1.43**	17.26	11.54
14	Spambase	19.33	18.27	38.60	15.80	16.09	11.43	**11.20**
15	Balance-Scale	12.84	**8.62**	12.80	15.20	14.40	12.31	9.19

In order to show the performance advantages of the KGMML algorithm, a score statistic is performed on the KGMML algorithm and other classic algorithms. The scoring process is as follows: assuming that A is the result of using the KGMML algorithm on a certain data set, and B is the result of uniting the KGMML algorithm on this data set (that is, using other algorithms). Firstly, comparative analysis of A and B is performed by using a 5% significance level t test. In a statistical sense, if $A > B$, it is considered that the result A obtained by using the KGMML algorithm on this dataset wins the result B using other algorithms. Thus, the statistical result of the score on this data set is recorded as "$1/0/0$". If $A < B$, the score statistics result is recorded as "$0/0/1$". If $A = B$, it means that A and B are the same in the statistical sense. Thus, it is considered that they are tied and recorded as "$0/1/0$". It is evident that the notation "$5/0/0$" signifies that the outcomes achieved by the KGMML algorithm outperform those of other algorithms across five datasets. On each data set and the overall score, the statistical results of the KGMML algorithm are listed in Table 4.

Table 4. The score statistics of the comparison between the results obtained by the KGMML algorithm and other classical algorithms.

Datasets	KGMML					
	GMML	DMLMJ	LMNN	GB-LMNN	Educlidean	Low-rank
Pima	1/0/0	1/0/0	1/0/0	1/0/0	1/0/0	1/0/0
Vehicle	0/1/0	1/0/0	1/0/0	1/0/0	1/0/0	1/0/0
German	1/0/0	0/1/0	1/0/0	1/0/0	1/0/0	1/0/0
Segment	1/0/0	0/1/0	1/0/0	1/0/0	1/0/0	1/0/0
Usps	1/0/0	0/1/0	1/0/0	1/0/0	1/0/0	1/0/0
Mnist	0/1/0	1/0/0	1/0/0	1/0/0	1/0/0	1/0/0
Glasses	1/0/0	0/1/0	0/0/1	0/0/1	0/1/0	1/0/0
DNA	0/1/0	0/0/1	0/0/1	0/1/0	1/0/0	1/0/0
Heart-Disease	1/0/0	0/1/0	1/0/0	0/1/0	1/0/0	1/0/0
Lymphography	1/0/0	1/0/0	1/0/0	1/0/0	1/0/0	1/0/0
Liver-Disorders	1/0/0	0/1/0	0/1/0	1/0/0	0/1/0	1/0/0
Hages-Roth	1/0/0	0/0/1	1/0/0	1/0/0	0/0/1	1/0/0
Ionosphere	1/0/0	0/1/0	0/0/1	0/0/1	0/0/1	1/0/0
Spambase	1/0/0	1/0/0	1/0/0	1/0/0	1/0/0	0/1/0
Balance	1/0/0	0/1/0	1/0/0	1/0/0	1/0/0	1/0/0
Total	12/3/0	5/8/2	11/1/3	9/2/2	11/2/2	14/1/0

5. Conclusions

Kernel geometric mean metric learning is proposed for nonlinear distance metric with the introduction of a kernel function. Traditional metric learning approaches aim to learn a global linear metric, which is not well-suited for nonlinear problems. The experimental results on the UCI dataset show that the algorithm can effectively improve the accuracy of GMML algorithm, and the nonlinear problems can be addressed by the proposed algorithm. In future work, the problem of the inaccurate similarity pair will be tried to improve where it exists in the kernel geometric mean metric learning algorithm. The partial labeling metric learning algorithm has been proposed in recent years, and a partial labeling algorithm based on kernel geometric mean metric learning will be proposed in the future.

Author Contributions: Conceptualization, J.H. and R.Z.; methodology, J.H.; software, Z.F. and T.Y.; validation, Z.F., T.Y. and Y.Z.; formal analysis, Z.F.; investigation, Y.Z.; resources, J.H.; data curation, R.Z.; writing—original draft preparation, Z.F.; writing—review and editing, Z.F.; visualization, Z.F.; supervision, R.Z.; project administration, J.H.; funding acquisition, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (62102062), the Humanities and Social Science Research Project of Ministry of Education(21YJCZH037), the Natural Science Foundation of Liaoning Province (2020-MS-134, 2020-MZLH-29).

Data Availability Statement: The data used to support the findings of this study are available from the corresponding author upon request.

Acknowledgments: The authors would like to thank the School of Information and Communication Engineering, Dalian Minzu University for assistance with simulation verifications related to this work.

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

References

1. Lu J, Wang R, Mian A, et al. Distance metric learning for pattern recognition[J]. Pattern recognition, 2018, 75: 1-3.
2. Wei Z, Cui Y, Zhou X, et al. A research on metric learning in computer vision and pattern recognition[C]//2018 Tenth International Conference on Advanced Computational Intelligence (ICACI). IEEE, 2018: 254-259.
3. Yan Y, Xia J, Sun D, et al. Research on combination evaluation of operational stability of energy industry innovation ecosystem based on machine learning and data mining algorithms[J]. Energy Reports, 2022, 8: 4641-4648.
4. Wang F, Sun J. Survey on distance metric learning and dimensionality reduction in data mining[J]. Data mining and knowledge discovery, 2015, 29(2): 534-564.
5. Yan M, Zhang Y, Wang H. Tree-Based Metric Learning for Distance Computation in Data Mining[C]//Asia-Pacific Web Conference. Cham: Springer International Publishing, 2015: 377-388.
6. Mojisola F O, Misra S, Febisola C F, et al. An improved random bit-stuffing technique with a modified RSA algorithm for resisting attacks in information security (RBMRSA)[J]. Egyptian Informatics Journal, 2022, 23(2): 291-301.
7. Kraeva I, Yakhyaeva G. Application of the metric learning for security incident playbook recommendation[C]//2021 IEEE 22nd International Conference of Young Professionals in Electron Devices and Materials (EDM). IEEE, 2021: 475-479.
8. Bennett J, Pomaznoy M, Singhania A, et al. A metric for evaluating biological information in gene sets and its application to identify co-expressed gene clusters in PBMC[J]. PLoS Computational Biology, 2021, 17(10): e1009459.
9. Makrodimitris S, Reinders M J T, Van Ham R C H J. Metric learning on expression data for gene function prediction[J]. Bioinformatics, 2020, 36(4): 1182-1190.
10. Yuan T, Dong L, Liu B, et al. Deep Metric Learning by Exploring Confusing Triplet Embeddings for COVID-19 Medical Images Diagnosis[C]//Workshop on Healthcare AI and COVID-19. PMLR, 2022: 1-10..
11. Jin Y, Lu H, Li Z, et al. A cross-modal deep metric learning model for disease diagnosis based on chest x-ray images[J]. Multimedia Tools and Applications, 2023: 1-22.
12. Xing Y, Meyer B J, Harandi M, et al. Multimorbidity Content-Based Medical Image Retrieval and Disease Recognition Using Multi-label Proxy Metric Learning[J]. IEEE Access, 2023.
13. Xing E, Jordan M, Russell S J, et al. Distance metric learning with application to clustering with side-information[J]. Advances in neural information processing systems, 2002, 15.
14. Davis J V, Kulis B, Jain P, et al. Information-theoretic metric learning[C]//Proceedings of the 24th international conference on Machine learning. 2007: 209-216.
15. Wang S, Jin R. An information geometry approach for distance metric learning[C]//Artificial intelligence and statistics. PMLR, 2009: 591-598.
16. Weinberger K Q, Saul L K. Distance metric learning for large margin nearest neighbor classification[J]. Journal of machine learning research, 2009, 10(2).
17. Zadeh P, Hosseini R, Sra S. Geometric mean metric learning[C]//International conference on machine learning. PMLR, 2016: 2464-2471.
18. Zhou Y, Gu H. Geometric mean metric learning for partial label data[J]. Neurocomputing, 2018, 275: 394-402.
19. Mika S, Ratsch G, Weston J, et al. Fisher discriminant analysis with kernels[C]//Neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop (cat. no. 98th8468). Ieee, 1999: 41-48.
20. Li Z, Kruger U, Xie L, et al. Adaptive KPCA modeling of nonlinear systems[J]. IEEE Transactions on Signal Processing, 2015, 63(9): 2364-2376.
21. Lee J M, Qin S J, Lee I B. Fault detection of non-linear processes using kernel independent component analysis[J]. The Canadian Journal of Chemical Engineering, 2007, 85(4): 526-536.
22. Zhang L, Zhou W D, Jiao L C. Kernel clustering algorithm[J]. CHINESE JOURNAL OF COMPUTERS-CHINESE EDITION-, 2002, 25(6): 587-590.
23. Choi H, Choi S. Kernel isomap[J]. Electronics letters, 2004, 40(25): 1612-1613.

24. Fasi M, Iannazzo B. Computing the weighted geometric mean of two large-scale matrices and its inverse times a vector[J]. SIAM Journal on Matrix Analysis and Applications, 2018, 39(1): 178-203.
25. Hager W W. Updating the inverse of a matrix[J]. SIAM review, 1989, 31(2): 221-239.
26. Bhatia R. Positive definite matrices[M]. Princeton university press, 2009.
27. Asuncion A, Newman D. UCI machine learning repository[J]. 2007.
28. Nguyen B, Morell C, De Baets B. Supervised distance metric learning through maximization of the Jeffrey divergence[J]. Pattern Recognition, 2017, 64: 215-225.
29. Kedem D, Tyree S, Sha F, et al. Non-linear metric learning[J]. Advances in neural information processing systems, 2012, 25.
30. Bhutani M, Jawanpuria P, Kasai H, et al. Low-rank geometric mean metric learning[J]. arXiv preprint arXiv:1806.05454, 2018.

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.