Machine Learning
https://doi.org/10.1007/s10994-023-06380-3

Boundary-restricted metric learning

Shuo Chen[1] · **Chen Gong**[2] · **Xiang Li**[4] · **Jian Yang**[2] · **Gang Niu**[1] · **Masashi Sugiyama**[1,3]

Received: 12 December 2022 / Revised: 28 April 2023 / Accepted: 17 July 2023

Abstract
Metric learning aims to learn a distance metric to properly measure the similarities between pairwise examples. Most existing learning algorithms are designed to reduce intra-class distances and meanwhile enlarge inter-class distances by critically introducing a *margin* between intra-class and inter-class distances. However, such learning objectives may yield *boundless* (distance) metric space, because their enlargements on inter-class distances are usually unconstrained. In this case, excessively enlarged inter-class distances would relatively reduce the ratio of margin to the whole distance range (i.e., the *margin-range-ratio*), and thus being against the initial large-margin purpose for discriminating the similarities of data pairs. To address this issue, we propose a new *boundary-restricted metric* (BRM), which confines the metric space by a restriction function. Such a restriction function is monotonous and gradually converges to an upper bound, which suppresses excessively large distances of data pairs and concurrently maintains the reliable discriminability. After that, the learned metric can be successfully restricted in a finite region, and thereby avoiding the reduction of margin-range-ratio. Theoretically, we prove that BRM tightens the *generalization error bound* of the traditional learning model without sacrificing the *fitting capability* or destroying the *topological property* of the learned metric, which implies that BRM makes a good *bias-variance tradeoff* for the metric learning task. Extensive experiments on toy data and real-world datasets validate the superiority of our approach over the state-of-the-art metric learning methods.

Keywords Metric learning · Boundary restriction · Generalization ability · Topological property

1 Introduction

Measuring distances/similarities between pairwise examples are required in many pattern recognition and machine learning tasks, such as clustering (Yan et al., 2022), classification (Dong et al., 2019), and verification (Yang et al., 2018). The manually designed simple measures (e.g., the *Euclidean distance* (Meyer, 2000; Yang et al., 2016)) can hardly adapt to diverse scenarios with different data distributions. Thereby, *(distance) metric*

Editor: Zhi-Hua Zhou.

Extended author information available on the last page of the article

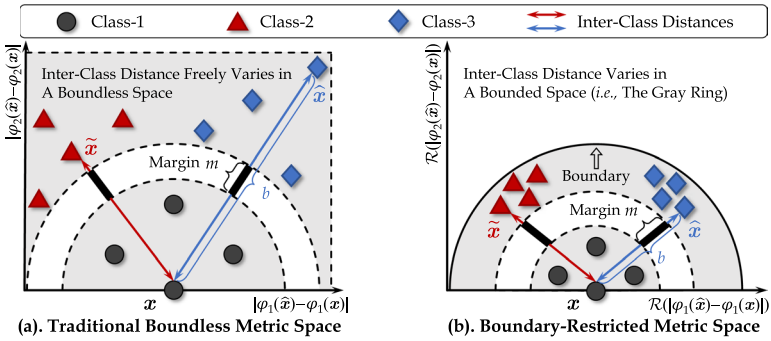


Fig. 1 Conceptual illustrations of the unconstrained boundless metric space and the boundary-restricted metric (BRM) space with the *same margin width*. **a** In the traditional unconstrained metric space, the margin-range-ratio (i.e., the value *m*/*b*) would be relatively reduced by excessively large distances. **b** Under the boundary restriction, the metric space is confined to prevent the reduction of margin-range-ratio, so that the data points could have better separability

learning (Lebanon, 2006; Weinberger et al., 2006; Yoshida et al., 2021) is proposed to learn intrinsic distance functions for various datasets based on some available supervision.

Metric learning usually considers the distance between each pairwise examples in a whole dataset, and determines a non-negative value for any two given examples to measure their similarity. Most existing learning algorithms are designed to enlarge the (*inter-class*) distances between negative pairs of examples and meanwhile reduce the (*intra-class*) distances between positive pairs of examples (Goldberger et al., 2005; Law et al., 2019), so that the similarities of data pairs can be discriminated as far as possible. To this end, loss functions in early learning algorithms punish the cases when inter-class distances are smaller than intra-class ones (Xing et al., 2003; Bar-Hillel et al., 2003). However, such learned metrics may not yield sufficiently large *margin* (i.e., the intermediate distance region for separating positive and negative pairs) to tolerate data variations in test phase (Suarez et al., 2018), and thus hurting the model generalizability. Therefore, some subsequent works set up a positive constant as the margin between their constrained intra-class and inter-class distances, and those methods are commonly dubbed as *contrastive similarity loss* (CSL) (Davis et al., 2007; Zadeh et al., 2016; Harandi et al., 2017). Nevertheless, with the improvement of model fitting capability, the traditional CSL might be over-fitted by trivial pairs (Oh Song et al., 2016). Accordingly, the *relative similarity loss* (RSL) (Sohn, 2016; Qian et al., 2019) is proposed to directly restrict the difference between inter-class and intra-class distances, which merely maintains a fixed margin width instead of controlling the absolute values of distances themselves. In the past, both CSL and RSL have shown promising results in many linear and nonlinear metric learning approaches (Ye et al., 2019b; Yoshida et al., 2021).

Although the above methods employing margin-based loss functions have gained increasing success, their learned metric spaces are usually boundless. In this case, the ratio of width-fixed margin to the whole distance range (i.e., the *margin-range-ratio*) would be reduced by the excessively enlarged distances (as shown in Fig. 1a). It leads to the effect that the margin is suppressed, and thereby the actual power of margin in discriminating data pair similarities is weakened (Xia et al., 2015; Sohn, 2016). A straightforward way to alleviate this issue is utilizing traditional *regularization techniques* (e.g., the ℓ_2-norm regularizer (Zadeh et al., 2016; Chen et al., 2019a)), which constrains the

parameter space of the metric learning model and thus reducing the variation of the predicted distances. However, distances with reduced variation may still yield boundless metric space, so the regularization techniques cannot solve the problem in essence. As an alternative strategy, recent works propose using upper-bounded functions (e.g., the *cosine function* (Xia et al., 2015) or *Lorentzian product* (Law et al., 2019)) to measure the difference between pairwise examples, and successfully improving the final recognition performance of learning algorithms. Nevertheless, this practice can hardly obtain a strict *metric* (Xing et al., 2003; Suarez et al., 2018; Chen et al., 2019b) which well preserves the critical topological properties of the conventional difference-norm based metrics (e.g., the well-known triangle property of the *Mahalanobis distance* (Huo et al., 2016) and the *manifold-based metric* (Zhu et al., 2018)). Thereby, a new metric learning framework is desired to explicitly restrict a bounded metric space and meanwhile preserve the topological property of metrics.

In this paper, we firstly provide the analytical study to understand how the boundless metric weakens the model generalizability, and secondly, we propose a new distance form to confine the metric space for enhancing model generalizability. To be specific, we consider the worst case of the traditional learned metric in the test phase and derive a probability value for *incorrect distance prediction*. Such a probability is monotonically increasing *w.r.t.* the metric space boundary, so it naturally inspires us to build a new *boundary-restricted metric* (BRM) to alleviate the incorrect distance prediction. To this end, we employ a *monotonous and gradually convergent function* to measure the divergence between two examples in each projection. Then, the inter-class distance can be confined in a bounded region, and thereby avoiding the reduction of margin-range-ratio (see Fig. 1b). As a result, the good ability of margin in discriminating the similarities of data pairs is well preserved. Theoretically, we prove that BRM tightens the *generalization error bound* of the learning algorithm without sacrificing the model *fitting capability*. We also reveal that the learned BRM preserves the *topological properties* of metrics, so that the model geometric soundness can be guaranteed. Intensive experiments are conducted on toy datasets and real-world datasets in comparison with both *linear* and *nonlinear* representative metric learning methods, and the results clearly demonstrate the effectiveness and superiority of our approach. The proposed method is simple and generic, and it can be easily deployed in many existing metric learning algorithms. Our main contributions are summarized below:

- We provide a new analytical result, which clearly reveals the quantitative relationship between the probability of incorrect distance prediction and the boundary of metric space.
- By explicitly confining the metric space, we propose a novel boundary-restricted metric (BRM) to enhance the generalizability of the traditional metric learning algorithm, with complete theoretical analyses guaranteeing the model effectiveness.
- Experimental investigations on synthesis datasets and real-world datasets validate the superiority of BRM to the state-of-the-art metric learning methods.

The rest of this paper starts with a brief review on the background in Sect. 2. Then, Sect. 3 details the BRM framework based on our analyses for the boundless metric space. Section 4 provides the theoretical guarantees on the topological property, fitting capability, and generalizability of BRM. Section 5 shows the experimental results on both synthetic and real-world benchmark datasets. Finally, Sect. 6 concludes our paper.

2 Background and related work

In this section, we first introduce some necessary notations. After that, we briefly review the main existing learning metrics and we also introduce the representative loss functions.

2.1 Notations

Throughout this paper, we write matrices and vectors as bold uppercase characters and bold lowercase characters, respectively. Let $\{y_1, y_2, \dots, y_N\}$ be the labels of training data pairs $\mathcal{X} = \{(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i) | i = 1, 2, \dots, N\}$ with $\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i \in \mathbb{R}^d$, where $y_i = 1$ if $\boldsymbol{x}_i$ and $\widehat{\boldsymbol{x}}_i$ are similar, and $y_i = 0$ otherwise. Here d is the data dimensionality, and N is the total number of data pairs. Operators $\|\cdot\|_p$ and $\|\cdot\|_{\mathrm{F}}$ denote the vector ℓ_p-norm ($p = 1$ or 2) and matrix Frobenius-norm, respectively. We use the neighbourhood symbol $\Delta(a, \delta)$ to simply represent all real numbers in $[a - \delta, a + \delta]$. For a random event A, $\mathrm{pr}[A]$ denotes the probability value of A that occurs.

2.2 Metric forms

In the metric learning task, the learnable metric forms are commonly divided into three types: the *linear metric*, the *manifold-based metric*, and the *deep neural network-based (DNN-based) metric* (Ye et al., 2019b; Chen et al., 2019b).

The Linear Metric. Originally, the learning metric is assumed as a *Mahalanobis distance* $\sqrt{(\boldsymbol{x} - \widehat{\boldsymbol{x}})^\top \boldsymbol{M} (\boldsymbol{x} - \widehat{\boldsymbol{x}})} = \|\boldsymbol{L}\boldsymbol{x} - \boldsymbol{L}\widehat{\boldsymbol{x}}\|_2$, where the data points $\boldsymbol{x}$ and $\widehat{\boldsymbol{x}}$ are from the d-dimensional sample space (Xing et al., 2003; Berrendero et al., 2020). Here the learning parameter $\boldsymbol{M} \in \mathbb{R}^{d \times d}$ is *semi-positive definite* (SPD) and can be equivalently decomposed to $\boldsymbol{M} = \boldsymbol{L}^\top \boldsymbol{L}$ with $\boldsymbol{L} \in \mathbb{R}^{r \times d}$ ($r \leq d$). Therefore, the Mahalanobis distance is also interpreted as a projected *Euclidean distance*. Recent works propose to learn a projected *Manhattan distance* $\|\boldsymbol{L}\boldsymbol{x} - \boldsymbol{L}\widehat{\boldsymbol{x}}\|_1$ for reducing the impact from outliers (Lim et al., 2013). Generally, the ℓ_1-norm based (Manhattan) distance has reliable robustness and the ℓ_2-norm based (*Euclidean*) distance is with favorable smoothness (Huo et al., 2016; Suarez et al., 2021). In the past, linear metric learning models have been well-studied with complete theoretical guarantees for both generalizability (Perrot and Habrard, 2015) and topological property (Paassen et al., 2018).

The Manifold-based Metric. As the above linear metrics might suffer from inadequate fitting capabilities, manifold approaches were utilized to enhance the model non-linearity on specific data following *Riemannian manifold* (Huang et al., 2018). They consider the geodesic distance between two d-dimensional square matrix variables $\boldsymbol{X}$ and $\widehat{\boldsymbol{X}}$ with a quadratic form of $\|\boldsymbol{M}^\top \boldsymbol{g}(\boldsymbol{X})\boldsymbol{M} - \boldsymbol{M}^\top \boldsymbol{g}(\widehat{\boldsymbol{X}})\boldsymbol{M}\|_{\mathrm{F}}$, where $\boldsymbol{M} \in \mathbb{R}^{d \times d}$ is the learning parameter, and the mapping $\boldsymbol{g} : \mathbb{R}^{d \times d} \to \mathbb{R}^{d \times d}$ is a transformation from the original space to the manifold space. For example, when the mapping $\boldsymbol{g}(\cdot)$ is instantiated by $\log(\cdot)$ function, it becomes to learn a metric on the SPD manifolds (Horev et al., 2017; Zhu et al., 2018).

The DNN-based Metric. To adaptively learn a nonlinear metric, the *deep neural network* (DNN) is introduced to improve the fitting capability of metric learning models. Specifically, the DNN-based metric intuitively extends the linear projection to a feature-extracted linear projection $\|\boldsymbol{L}\boldsymbol{\mathcal{W}}(\boldsymbol{x}) - \boldsymbol{L}\boldsymbol{\mathcal{W}}(\widehat{\boldsymbol{x}})\|_p$, where the mapping $\boldsymbol{\mathcal{W}} : \mathbb{R}^d \to \mathbb{R}^h$ ($h \in \mathbb{N}_+$) is a feature extractor implemented by typical DNN models such as *convolutional neural network* (CNN) (Zbontar and LeCun, 2016) and *multi-layer perceptron*

(MLP) (Franklin, 2005). Some popular training tricks and novel mechanisms (e.g., *self-attention* (Zhang et al., 2019a) and *adversarial generation* (Goodfellow et al., 2014)) are also utilized in such a learning framework to improve the prediction accuracy. The DNN-based metric learning has been successfully applied in some challenging recognition tasks, especially showing promising results on some image related benchmarks (Lu et al., 2019; Yan et al., 2022).

A Unified Form. Without loss of generality, for the learning parameters in the above three types of distance metrics, we can define a general vector-valued function $\boldsymbol{\varphi} : \mathbb{R}^d \to \mathbb{R}^h$, which consists of h real-valued functions. Specifically, the mapping result $\boldsymbol{\varphi}(\boldsymbol{x}) = [\varphi_1(\boldsymbol{x}), \varphi_2(\boldsymbol{x}), \ldots, \varphi_h(\boldsymbol{x})]^\top$ and the real-valued function $\varphi_i : \mathbb{R}^d \to \mathbb{R}$ is continuous for $i = 1, 2, \ldots, h$. Then the main existing distance metrics listed above could be briefly unified as the following generic ℓ_p-norm based formulation [12]

$$d_{\boldsymbol{\varphi}}(\boldsymbol{x}, \widehat{\boldsymbol{x}}) = \left(\frac{1}{h} \sum_{i=1}^{h} |\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|^p \right)^{1/p}, \tag{1}$$

where $p = 1$ or 2 for common cases, and the general vector-valued function $\boldsymbol{\varphi}$ is learned from given loss functions. Here the non-negative distance $d_{\boldsymbol{\varphi}}(\boldsymbol{x}, \widehat{\boldsymbol{x}})$ is usually boundless, as the ℓ_p-norm can be arbitrarily enlarged with the increasing divergence between $\boldsymbol{\varphi}(\boldsymbol{x})$ and $\boldsymbol{\varphi}(\widehat{\boldsymbol{x}})$. In this paper, we focus on such a unified distance metric [i.e., Eq. (1)] and explore the corresponding issue incurred by its boundless distance form.

2.3 Loss functions

To explicitly learn the metric parameter by the way of *empirical risk minimization* (ERM) (Alpaydin, 2020; Kwon et al., 2020), there are mainly two categories of margin-based loss functions which are usually dubbed as *contrastive similarity loss* (Davis et al., 2007; Xu et al., 2019) and *relative similarity loss* (Oh Song et al., 2016), respectively.

Contrastive Similarity Loss (CSL). For a parameterized distance metric $d_{\boldsymbol{\varphi}}$ in Eq. (1), CSL aims to build evaluation functions $\ell_u^+(\cdot)$ and $\ell_v^-(\cdot)$ for positive and negative pairs, respectively. The function values will increase when the intra-class distance is greater than u or the inter-class distance is smaller than v. Then the corresponding empirical risk on the training dataset $\mathcal{X}$ with N data pairs has a form

$$\mathcal{L}_{\mathrm{c}}(\boldsymbol{\varphi}) = \frac{1}{N} \sum_{i=1}^{N} y_i \ell_u^+(d_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i)) + (1 - y_i)\ell_v^-(d_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i)), \tag{2}$$

where the threshold parameters $v > u > 0$ are pre-set. Here the positive value $v - u$ plays the role of margin, which guarantees a margin space for discriminating the similarities of positive and negative pairs. In the past, a considerable number of metric learning models were learned with the above CSL and achieved promising results by different implementations of $\ell_u^+(\cdot)$ and $\ell_v^-(\cdot)$ (Xie et al., 2018; Law et al., 2019).

Relative Similarity Loss (RSL). For many nonlinear metrics, CSL would incur overfitting due to the high complexity of model nonlinearity, so it cannot always obtain a gener-

[1] Here h adaptively scales the oversized measurement of very high-dimensional projected features.

[2] To include the manifold-based metric, we let $\boldsymbol{\varphi}(\boldsymbol{x}) = \boldsymbol{\varphi}(\mathrm{vec}(\boldsymbol{X})) = d \cdot \mathrm{vec}(\boldsymbol{M}^\top \boldsymbol{g}(\boldsymbol{X})\boldsymbol{M}) \in \mathbb{R}^{d^2}$.

alizable metric for the test phase (Chu et al., 2020). In this case, RSL is proposed to explore the relative similarity between positive pair $(\boldsymbol{x}_i, \boldsymbol{x}_i^+)$ and negative pair $(\boldsymbol{x}_i, \boldsymbol{x}_i^-)$ in the triplet set $\mathcal{T} = \{(\boldsymbol{x}_i, \boldsymbol{x}_i^+, \boldsymbol{x}_i^-) | (\boldsymbol{x}_i, \boldsymbol{x}_i^+), (\boldsymbol{x}_i, \boldsymbol{x}_i^-) \in \mathcal{X}, \text{where} y_{(\boldsymbol{x}_i, \boldsymbol{x}_i^+)} = 1, y_{(\boldsymbol{x}_i, \boldsymbol{x}_i^-)} = 0, \text{and} i = 1, \ldots, T\}$. For this triplet set, the corresponding relative similarities are penalized by the following empirical risk

$$\mathcal{L}_{\mathrm{r}}(\boldsymbol{\varphi}) = \frac{1}{T} \sum_{i=1}^{T} \widetilde{\ell}(\max(d_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \boldsymbol{x}_i^+) - d_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \boldsymbol{x}_i^-) + \tau, 0)), \tag{3}$$

where $\tau = v - u > 0$ is a (relative) margin between positive and negative pairs, and $\widetilde{\ell}(\cdot)$ is implemented by a *monotonically increasing function*. With such relative similarities, nonlinear metric learning models could successfully benefit from the more plentiful supervision compared with the contrastive similarities. Nevertheless, the slow convergence of loss functions might be incurred by the dramatic increase from the pair number N to the triplet number T. Accordingly, recent works proposed various triplet and tuplet sampling techniques (Qian et al., 2019; Sohn, 2016) to improve the convergence speed of RSL and have shown promising results on DNN based metric learning approaches.

Considering that the above two loss functions have been widely employed by existing metric learning methods, in this paper, we investigate the effectiveness of our proposed BRM in such two cases.

3 Methodology

In this section, we first theoretically analyze the probability of incorrect distance prediction for the traditional boundless metric. After that, a novel boundary-restricted metric (BRM) is proposed to adaptively restrict the metric space. The learning objective and the corresponding optimization algorithm are finally designed with convergence guarantee.

3.1 False positives on test data

Suppose that the metric parameter [i.e., the h-dimensional mapping $\boldsymbol{\varphi}$ in Eq. (1)] is searched from the hypothesis space $\mathcal{H}$. When the loss function in Eq. (2) or Eq. (3) is employed to learn the metric, $\boldsymbol{\varphi}$ is expected to be contained in an optimized hypothesis set $\mathcal{H}_v^u = \{\boldsymbol{\varphi} \in \mathcal{H} | d_{\boldsymbol{\varphi}}(\boldsymbol{\alpha}, \widehat{\boldsymbol{\alpha}}) \geq v > u \geq d_{\boldsymbol{\varphi}}(\boldsymbol{\beta}, \widehat{\boldsymbol{\beta}}), \forall (\boldsymbol{\alpha}, \widehat{\boldsymbol{\alpha}}) \in \mathcal{X}^- \text{and} (\boldsymbol{\beta}, \widehat{\boldsymbol{\beta}}) \in \mathcal{X}^+\}$, where the negative pair set $\mathcal{X}^-$ and positive pair set $\mathcal{X}^+$ satisfy $\mathcal{X}^- \cup \mathcal{X}^+ = \mathcal{X}$. Here the hypothesis set $\mathcal{H}_v^u \subseteq \mathcal{H}$ might contain more than one optimal hypothesis element, because the inter-class distance $d_{\boldsymbol{\varphi}}(\boldsymbol{\alpha}, \widehat{\boldsymbol{\alpha}})$ can freely vary in the range of $(v, +\infty)$. Therefore, we denote the metric space boundary as $b(\mathcal{H}_v^u) = \sup_{\boldsymbol{\varphi} \in \mathcal{H}_v^u, (\boldsymbol{x}, \widehat{\boldsymbol{x}}) \in \mathcal{X}, i = 1, \ldots, h} \{|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|\}$, and we investigate the relationship between $b(\mathcal{H}_v^u)$ and the generalizability of $\boldsymbol{\varphi}$ learned from $\mathcal{H}_v^u$.

Firstly, here we choose the widely-used triplet loss to empirical reveal the issue of boundless metric space. We employ the triplet loss to learn the distance metric on *CAR-196* dataset (Oh Song et al., 2016), and visualize the two distance distribution results in Fig. 2 by respectively setting the margin parameter of loss function [namely τ in Eq. (3)] to 1 and 3. We can find that the maximal distance (i.e., the metric space boundary $b(\mathcal{H}_v^u)$) is amplified when increasing the margin value from 1 to 3. It means that the conventional method can hardly improve the margin-range-ratio by merely setting a large margin parameter in the loss

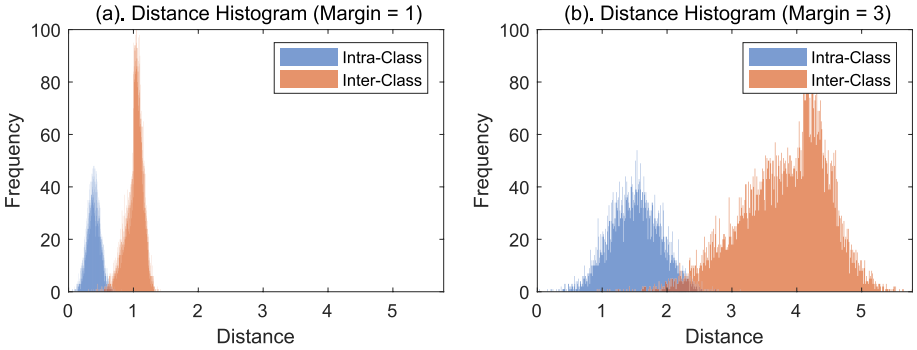


Fig. 2 Distance histograms learned by triplet (relative similarity) loss on *CAR-196* dataset. We set the margin parameter [i.e., the parameter τ in Eq. (3)] as 1 and 3, respectively, and record the learned pairwise distances to plot the histograms

function. This is because the learning algorithm may prefer to enlarge all negative pairs of distances for obtain a large margin between intra-class and inter-class distances. In this case, simply setting a larger margin parameter cannot always achieve better generalizability, and this phenomenon is also observed in some existing works (Zhang et al., 2019b; Yu and Tao, 2019).

For further theoretically evaluating the model generalizability, it is well known that the *generalization error bound* (GEB) (Ye et al., 2019a) measures the bias between the empirical risk and generalization risk for a machine learning model. However, most existing GEB results (Chen et al., 2019b; Luo et al., 2019) mainly focus on the convergence behavior of the error bound that is related to the sample size. They cannot take into account the impact from boundary $b(\mathcal{H}_v^u)$, so here we would like to conduct a new analytical study to connect the model generalizability with the boundary of a metric space. Since most existing metric learning algorithms (Harandi et al., 2017; Xie et al., 2018) are designed to seek for orthogonal projections, we can assume that the h projections are independent to each other. Specifically, as the orthogonal constraint is usually carried on the fully-connected layer of the (non-linear) neural network model or the whole projection matrix of the linear model. Without loss of generality, we can formulate the feature embedding $\boldsymbol{\varphi}(\boldsymbol{x}) \in \mathbb{R}^h$ as $\boldsymbol{\varphi}(\boldsymbol{x}) = \boldsymbol{L}(\boldsymbol{\phi}(\boldsymbol{x}))$, where $\boldsymbol{L} = [\boldsymbol{L}_1^\top, \boldsymbol{L}_2^\top, \ldots, \boldsymbol{L}_h^\top]^\top \in \mathbb{R}^{h \times h'}$ is with h orthogonal rows and $\boldsymbol{\phi}(\boldsymbol{x}) \in \mathbb{R}^{h'}$ is the feature mapping ahead of the fully-connected layer. Then we have that

$$\begin{aligned}
&\mathrm{cov}[\varphi_i(\boldsymbol{x}), \varphi_j(\boldsymbol{x})] \\
&\quad = \mathbb{E}[(\varphi_i(\boldsymbol{x}) - \overline{\varphi}_i)(\varphi_j(\boldsymbol{x}) - \overline{\varphi}_j)] \\
&\quad = \mathbb{E}[\varphi_i(\boldsymbol{x}) \cdot \varphi_j(\boldsymbol{x})] - \overline{\varphi}_i \overline{\varphi}_j \\
&\quad = \mathbb{E}[(\boldsymbol{L}_i \boldsymbol{\phi}(\boldsymbol{x})) \cdot (\boldsymbol{L}_j \boldsymbol{\phi}(\boldsymbol{x}))] - \mathbb{E}[(\boldsymbol{L}_i \boldsymbol{\phi}(\boldsymbol{x}))] \mathbb{E}[(\boldsymbol{L}_j \boldsymbol{\phi}(\boldsymbol{x}))] \\
&\quad = \mathbb{E}[\mathrm{tr}((\boldsymbol{\phi}(\boldsymbol{x}))^\top \boldsymbol{L}_i^\top \boldsymbol{L}_j (\boldsymbol{\phi}(\boldsymbol{x})))] - (\boldsymbol{L}_i \mathbb{E}[\boldsymbol{\phi}(\boldsymbol{x})]) \cdot (\boldsymbol{L}_j \mathbb{E}[\boldsymbol{\phi}(\boldsymbol{x})]) \\
&\quad = \mathbb{E}[(\boldsymbol{L}_j \boldsymbol{L}_i^\top)(\boldsymbol{\phi}(\boldsymbol{x})^\top \boldsymbol{\phi}(\boldsymbol{x}))] - (\mathbb{E}[\boldsymbol{\phi}(\boldsymbol{x})])^\top (\boldsymbol{L}_j \boldsymbol{L}_i^\top)(\mathbb{E}[\boldsymbol{\phi}(\boldsymbol{x})]) \\
&\quad = 0,
\end{aligned} \tag{4}$$

which implies that $\varphi_i(\boldsymbol{x})$ and $\varphi_j(\boldsymbol{x})$ are statistically uncorrelated under the orthogonal constraints. Here we also provide experimental results to further validate the reasonability of this assumption. We select two representative metric learning methods GMML (Zadeh et al., 2016) and SoftTriple (Qian et al., 2019) to visualize the covariance between each

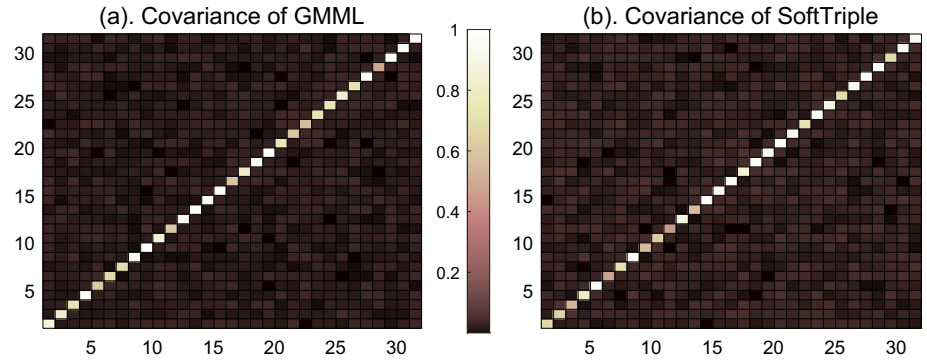


Fig. 3 Visualization for the covariance matrices of the learned feature embedding

two features $\varphi_i(\boldsymbol{x})$ and $\varphi_j(\boldsymbol{x})$. Here GMML is learned on the *PubFig* dataset and SoftTriple is learned on the *CAR-196* dataset. We set the embedding size $h = 32$ for both the linear method (i.e., GMML) and deep method (i.e., SoftTriple). As shown in Fig. 3, we can observe that the diagonal elements (i.e., $\varphi_i(\boldsymbol{x})^\top \varphi_i(\boldsymbol{x})$) of the covariance matrices are significantly larger than the other elements (i.e., $\varphi_i(\boldsymbol{x})^\top \varphi_j(\boldsymbol{x})$ for $i \neq j$), and non-diagonal elements are very close to zero. This clearly demonstrates that each two features $\varphi_i(\boldsymbol{x})$ and $\varphi_j(\boldsymbol{x})$ are statistically uncorrelated.

As each projection would have its own characteristic, projection results are not identically distributed necessarily. It means that $\varphi_1(\boldsymbol{x}), \varphi_2(\boldsymbol{x}), \ldots, \varphi_h(\boldsymbol{x})$ should be independent non-identically distributed (*i.n.i.d.*) random variables (Ralaivola et al., 2010). Then, we investigate the distribution of distance $d_{\boldsymbol{\varphi}}(\boldsymbol{x}, \hat{\boldsymbol{x}})$ in the non-negative real space. As the boundless metric space is incurred by negative pairs, here we consider to investigate the incorrectly predicted negative pairs affected by the boundary $b(\mathcal{H}_v^u)$. Specifically, we consider the distribution of distance $d_{\boldsymbol{\varphi}}(\boldsymbol{x}, \hat{\boldsymbol{x}})$ in the non-negative real space. Specifically, by invoking *Lindeberg central limit theorem* (CLT) (Vershynin, 2018) on $d_{\boldsymbol{\varphi}}(\boldsymbol{x}, \hat{\boldsymbol{x}})$, we can easily have the following assertion in the high-dimensional feature space (i.e., a sufficiently large h)

$$\lim_{h\to\infty}\left(\frac{\sum_{i=1}^{h}\left(|\varphi_i(\boldsymbol{x}) - \varphi_i(\hat{\boldsymbol{x}})|^p - \mu_i\right)}{\sqrt{\sum_{i=1}^{h}\sigma_i^2}}\right) \sim \mathcal{N}(0,1), \tag{5}$$

where $\mu_i \in [\mu_L, \mu_U]$ and $\sigma_i \in [\sigma_L, \sigma_U]$ are the expectation and variance of $|\varphi_i(\boldsymbol{x}) - \varphi_i(\hat{\boldsymbol{x}})|^p$, respectively. Here $\mu_U > \mu_L > 0$ and $\sigma_U > \sigma_L > 0$ ensure the finite expectation/variance. Note that the condition of $h \to \infty$ can be easily satisfied in practical use. This is because the CLT merely needs the dimensionality h to be larger than 30 (Montgomery and Runger, 2010) to take effect, while h is usually set to 128 or 512 in metric learning tasks (Huo et al., 2016; Ye et al., 2019b). The above Eq. (5) reveals that the standardized learning distance approximately obeys the standard normal distribution in the high-dimensional feature space, and thus offering us a effective way to calculate the probability of incorrect prediction result. As the boundless metric space is incurred by negative pairs, here we investigate the incorrectly predicted negative pairs (i.e., the false positive result $d_{\boldsymbol{\varphi}}(\boldsymbol{z}, \hat{\boldsymbol{z}}) < u$ for the negative *test pair* $(\boldsymbol{z}, \hat{\boldsymbol{z}})$) affected by the boundary $b(\mathcal{H}_v^u)$. To be more rigorous, the

corresponding upper bound of the expected *false positive rate* (FPR) on the test data is described below.

Theorem 1 (Upper Bound of FPR) *Assume that the h-dimensional feature mapping* $\boldsymbol{\varphi} \in \mathcal{H}_v^u$ [*i.e., the metric parameter in Eq.* (1)] *is learned from the training data* $\mathcal{X}^- \cup \mathcal{X}^+$ *with N data pairs. Then for a given* $\delta \in (0, \min(1, v^p - u^p))$ *and sufficiently large integers N and h, it holds that with probability at least* $1 - \delta$

$$\sup_{\boldsymbol{\varphi} \in \mathcal{H}_v^u} \left\{ \mathrm{pr}\left[d_{\boldsymbol{\varphi}}(\boldsymbol{z}, \hat{\boldsymbol{z}}) < u \right] \right\} \in \Delta\left(\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\psi[b(\mathcal{H}_v^u)]} \mathrm{e}^{-t^2/2} \mathrm{d}t, \delta \right), \tag{6}$$

where the real-valued function $\psi\left[b(\mathcal{H}_v^u)\right] = \sqrt{h}(1 + 2(v^p - u^p)/(v^p - b^p(\mathcal{H}_v^u)))$ *is strictly monotonically increasing , and the real value* $b(\mathcal{H}_v^u)$ *is the boundary of the hypothesis set* $\mathcal{H}_v^u$. *Here the* (*random*) *test pair* $(\boldsymbol{z}, \hat{\boldsymbol{z}})$ *obeys the same distribution with the training pairs in* $\mathcal{X}^-$.

The proof of Theorem 1 is given in the Appendix. From the upper bound result in the above Eq. (6), we can clearly observe that $(1/\sqrt{2\pi}) \int_{-\infty}^{\psi[b(\mathcal{H}_v^u)]} \mathrm{e}^{-t^2/2} \mathrm{d}t$ is consistently enlarged with the increase of the boundary value $b(\mathcal{H}_v^u)$. This growing integral value is a good approximation (with an arbitrary small δ) to the probability of incorrect distance prediction. However, for most existing metric learning models, the distance metric formulated as Eq. (1) does not have an explicitly finite boundary to prevent the increase of the probability in Eq. (6). Therefore, in Sect. 3.2, we propose a boundary-restricted metric to suppress $b(\mathcal{H}_v^u)$, so that we can guarantee a limited integral fractile $\psi[b(\mathcal{H}_v^u)]$ in Theorem 1. Then the expected FPR could be controllable during the test phase of the learning algorithm.

3.2 Boundary-restricted metric

As we discussed in the previous subsection, the boundless metric space potentially hurts the model generalizability. Consequently, the traditional distance metric in Eq. (1) may not deliver accurate prediction results in some difficult cases. Now we consider to make a straightforward modification on its original metric form.

Actually, the boundless metric space of Eq. (1) results from the ℓ_p-norm computation on $|\varphi_i(\boldsymbol{x}) - \varphi_i(\hat{\boldsymbol{x}})|$. We naturally restrict the difference value $|\varphi_i(\boldsymbol{x}) - \varphi_i(\hat{\boldsymbol{x}})|$ by a specific function instead of the original ℓ_p-norm in each projected direction. To be more specific, we employ a *smooth and monotonically increasing function* $\mathcal{R}(\cdot)$ to restrict the divergence $|\varphi_i(\boldsymbol{x}) - \varphi_i(\hat{\boldsymbol{x}})|$ for $i = 1, 2, \ldots, h$. Based on the unified form in Eq. (1), we propose the following new boundary-restricted metric (BRM).

Definition 1 For a smooth and monotonically increasing function $\mathcal{R} : [0, +\infty) \to [0, B]$ and an h-dimensional mapping $\boldsymbol{\varphi} : \mathbb{R}^d \to \mathbb{R}^h$, our BRM is defined as

$$\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \left(\frac{1}{h} \sum_{i=1}^{h} \left[\mathcal{R}\left(|\varphi_i(\boldsymbol{x}) - \varphi_i(\hat{\boldsymbol{x}})| \right) \right]^p \right)^{1/p}, \tag{7}$$

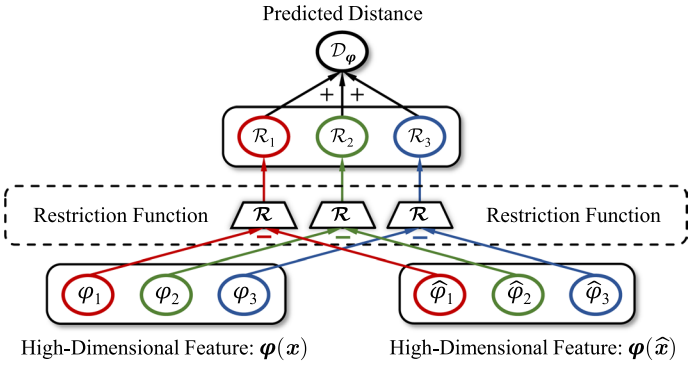


Fig. 4 The overall framework of our proposed BRM (i.e., $\mathcal{D}_{\boldsymbol{\varphi}}$ in Definition 1). For two given examples $\boldsymbol{x}$ and $\widehat{\boldsymbol{x}}$, we employ a bounded and monotonically increasing restriction function $\mathcal{R}$ to measure their divergence on each projected direction (i.e., $|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|$). Then, the accumulated results are summed up as the final predicted distance

where the mapping $\boldsymbol{\varphi}(\boldsymbol{x}) = [\varphi_1(\boldsymbol{x}), \varphi_2(\boldsymbol{x}), \ldots, \varphi_h(\boldsymbol{x})]^\top$ and the data points $\boldsymbol{x}, \widehat{\boldsymbol{x}} \in \mathbb{R}^d$. Here the constant $B > 0$ is the upper bound of the restriction function $\mathcal{R}(\cdot)$ in the domain $[0, +\infty)$.

Figure 4 offers an intuitive visualization for the above Definition 1. It is notable that the restriction function $\mathcal{R}(\cdot)$ plays a critical role in suppressing the distance result within a controllable region. Specifically, as we assumed that the restriction function $\mathcal{R}(\cdot)$ is monotonically increasing in the codomain $[0, B]$, this restriction function will necessarily converge to a definite value which is not larger than B. As shown in Fig. 5, such a gradually convergent function adjusts the excessively large inter-class distances in a constrained region, which is comparable with the intra-class distances. Thereby, it successfully ensures a relatively large margin-range-ratio in the whole metric space.

Bounded FPR. Now we further investigate the expected FPR value of the proposed BRM (i.e., $\mathcal{D}_{\boldsymbol{\varphi}}(\cdot, \cdot)$). Specifically, according to our derived Theorem 1, it follows that for a given $\delta \in (0, \min(1, v - u))$ and sufficiently large integers N and h, with probability at least $1 - \delta$ we have

$$\sup_{\boldsymbol{\varphi} \in \mathcal{H}_v^u} \left\{ \mathrm{pr}\left[\mathcal{D}_{\boldsymbol{\varphi}}(z, \widehat{z}) < u\right] \right\} \in \Delta\left(\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\beta} \mathrm{e}^{-t^2/2} \mathrm{d}t, \delta \right), \tag{8}$$

where the constant $\beta = \sqrt{h}(1 + 2(v^p - u^p)/(v^p - B))$ does not depend on the variation of training data any more. Here the test pair $(z, \widehat{z})$ *obeys the same distribution* with negative training pairs. From the above Eq. (8), we can clearly observe that the expected FPR of BRM is bounded by a constant β, so that the probability supremum the of incorrect prediction can be well controlled.

Finally, we want to briefly discuss about the instantiation of the restriction function $\mathcal{R}(\cdot)$. As stated in Definition 1, the smooth and monotonically increasing function $\mathcal{R} : [0, +\infty) \to [0, B]$ does not have any other specific constraints. Interestingly, many *activation functions* (as listed in Table 1) in neural networks can be directly employed as the restriction function $\mathcal{R}(\cdot)$. That is to say, we can also regard the calculation of BRM as a *feedforward layer* of neural network (see Fig. 4). Correspondingly, in our experiments, the

Table 1 Representative activation functions defined in $(-\infty, +\infty)$ that could be served as the restriction function $\mathcal{R}(\cdot)$ in our proposed BRM

Function	Expression	Codomain
Soft-Sign (Glorot and Bengio, 2010)	$\mathcal{R}(t) = t/(1+\|t\|)$	$(-1, 1)$
Sigmoid (Cybenko, 1989)	$\mathcal{R}(t) = 2/(1+\mathrm{e}^{-t}) - 1$	$(-1, 1)$
ArcTan (Weisstein, 2002)	$\mathcal{R}(t) = \tan^{-1}(t)$	$(-\pi/2, \pi/2)$
TanH (Alpaydin, 2020)	$\mathcal{R}(t) = (\mathrm{e}^{t} - \mathrm{e}^{-t})/(\mathrm{e}^{t} + \mathrm{e}^{-t})$	$(-1, 1)$
ISRU (Carlile et al., 2017)	$\mathcal{R}(t) = t/\sqrt{1+\omega t^2}$ $(\omega > 0)$	$(-1/\sqrt{\omega}, 1/\sqrt{\omega})$

Note here we only need the positive domain and codomain

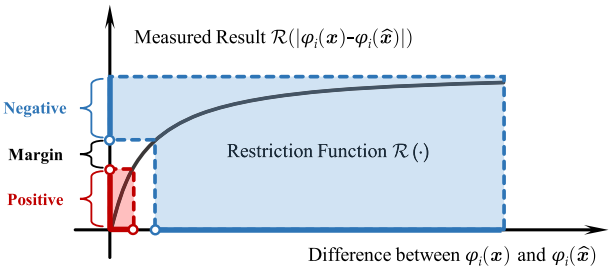


Fig. 5 A visualization of the restriction function $\mathcal{R}(\cdot)$. For the consistently increased divergence between a pair of examples, we utilize a bounded and monotonically increasing function to restrict the excessively large divergence. After that, the boundless distance metric space (in horizontal) can be well adjusted in a finite region (in vertical)

effectiveness of BRM is also validated in an end-to-end neural network architecture. It is worth pointing out that there is no additional learning parameter introduced into the model. Therefore, our proposed BRM maintains the computational complexity of the traditional learning algorithm, which is also discussed in the next subsection.

3.3 Model setup and optimization

This subsection depicts the learning objective and the corresponding optimization algorithm for our proposed BRM. As we discussed in Sect. 2.3, the optimization model of metric learning is usually based on the minimization of the empirical loss (Geng and Chen, 2018; Bian and Tao, 2012). For our proposed BRM, we follow this common practice and also minimize an empirical loss to learn the parameter $\boldsymbol{\varphi}$ of the distance function $\mathcal{D}_{\boldsymbol{\varphi}}(\cdot,\cdot)$.

Without loss of generality, we denote that the function $\ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i); y_i)$ evaluates the inconsistency between the predicted distance $\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i)$, where the label y_i is the similarity (positive/negative) between $\boldsymbol{x}_i$ and $\widehat{\boldsymbol{x}}_i$. By further leveraging the regularizer to reduce over-fitting, the learning objective of BRM can be formulated as

$$\min_{\boldsymbol{\varphi}\in\mathcal{H}} \left\{ \mathcal{F}(\boldsymbol{\varphi}) = \frac{1}{N}\sum_{i=1}^{N} \ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i); y_i) + \lambda\Omega(\boldsymbol{\varphi}) \right\}, \tag{9}$$

where the regularization parameter $\lambda > 0$ is tuned to balance the fitting accuracy and the complexity of hypothesis space. The second term $\Omega(\boldsymbol{\varphi})$ can be implemented by

commonly-used regularization techniques such as the ℓ_2-norm regularizer (Huo et al., 2016). Due to the high sample complexity of metric learning, the number N is usually very large, so we utilize the *stochastic gradient descent* (SGD) method to solve Eq. (9) which picks up a mini-batch in each iteration. Specifically, in the $(t+1)$-th iteration, we conduct an update

$$\boldsymbol{\varphi}^{(t+1)} := \boldsymbol{\varphi}^{(t)} - \eta\left(\frac{1}{K}\sum_{i=1}^{K}\nabla_{\boldsymbol{\varphi}}\ell_{k_i} + \lambda\nabla_{\boldsymbol{\varphi}}\Omega\right), \tag{10}$$

where K is the batch size and $\eta > 0$ is the step size of each iteration (i.e., the learning rate). Here $\ell_{k_i} = \ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_{k_i}, \widehat{\boldsymbol{x}}_{k_i}); y_{k_i})$ is the i-th element function in one single mini-batch ($k_i = 1, 2, \ldots, N$ for $i = 1, 2, \ldots, K$). Based on the above discussion, we summarize the main iteration steps solving Eq. (9) in Algorithm 1.

Gradient Analysis. We denote that $\widetilde{\ell}(\mathcal{D}^p_{\boldsymbol{\varphi}}(\boldsymbol{x}, \widehat{\boldsymbol{x}})) = \ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}, \widehat{\boldsymbol{x}}); y_{(\boldsymbol{x}\widehat{\boldsymbol{x}})})$ for simplicity. Then the gradient of loss function can be calculated as

$$\begin{aligned}
&\partial\widetilde{\ell}(\mathcal{D}^p_{\boldsymbol{\varphi}}(\boldsymbol{x}, \widehat{\boldsymbol{x}}))/\partial\varphi_i \\
&\quad = p/h \cdot [\mathrm{d}\widetilde{\ell}(\mathcal{D}^p_{\boldsymbol{\varphi}}(\boldsymbol{x}, \widehat{\boldsymbol{x}}))/\mathrm{d}\mathcal{D}^p_{\boldsymbol{\varphi}}(\boldsymbol{x}, \widehat{\boldsymbol{x}})] \cdot |\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|^{p-1} \\
&\qquad \cdot \mathrm{sign}(\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}}))[\mathrm{d}(\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}}))/\mathrm{d}\varphi_i] \\
&\qquad \cdot [\mathrm{d}\mathcal{R}(|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|)/\mathrm{d}|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|],
\end{aligned} \tag{11}$$

and we can observe that the restriction function $\mathcal{R}(\cdot)$ will incur an additional term $\mathrm{d}\mathcal{R}(|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|)/\mathrm{d}|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|$ to the original gradient of conventional metric learning objective. As $\mathcal{R}(\cdot)$ is assumed to be monotonically increasing and gradually convergent, such an additional term $\mathrm{d}\mathcal{R}(|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|)/\mathrm{d}|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|$ can be regarded as a scale variable. Specifically, with the increasing of $|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|$, the function $\mathcal{R}(|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|)$ converges to *an upper bound* so that the derivative $\mathrm{d}\mathcal{R}(|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|)/\mathrm{d}|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|$ gradually converges to 0. Then the gradient $\partial\widetilde{\ell}(\mathcal{D}^p_{\boldsymbol{\varphi}}(\boldsymbol{x}, \widehat{\boldsymbol{x}}))/\partial\varphi_i$ is rescaled and converges to 0 to prevent the update of learning parameters, and thus avoiding the extremely large distance determination. From the above dependency relationship between the gradient and restriction function, we can also find that introducing the restriction function $\mathcal{R}(\cdot)$ effectively overcomes the boundless problem in conventional metric learning algorithms.

Algorithm Convergence. Existing convergence analysis of SGD (Reddi et al., 2016) usually focuses on a given optimization objective composed of a series of *Lipschitz-continuous* functions [3]. Here we would like to discuss the Lipschitz-continuity of our final learning objective $\mathcal{F}(\boldsymbol{\varphi}) = \frac{1}{N}\sum_{i=1}^{N}\ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i); y_i) + \lambda\Omega(\boldsymbol{\varphi})$ in detail. Specifically, it has already been well proved that our employed hingle loss $\ell(\cdot)$ and ℓ_2-norm regularizer $\Omega(\cdot)$ are Lipschitz continuous (Kar et al., 2014; Li et al., 2019), respectively. Meanwhile, the Lipschitz-continuity of CNN (namely the feature embedding $\boldsymbol{\varphi}(\boldsymbol{x})$ in our manuscript) is also validated by showing its corresponding Lipschitz constant (Fazlyab et al., 2019), i.e., $\|\boldsymbol{\varphi}(\boldsymbol{x}_i) - \widetilde{\boldsymbol{\varphi}}(\boldsymbol{x}_i)\|_2 \leq L_0\|\boldsymbol{\varphi} - \widetilde{\boldsymbol{\varphi}}\|_2$ for any $i = 1, 2, \ldots, N$. Furthermore, we can observe that all restriction functions implemented in our experiments (e.g., the Sigmoid function) are

[3] A differentiable function $f(\boldsymbol{a})$ defined on the domain $\mathcal{C}$ is Lipschitz-continuous if and only if there exists $L > 0$ such that $|f(\boldsymbol{a}) - f(\boldsymbol{b})| \leq L\|\boldsymbol{a} - \boldsymbol{b}\|_2$ for any $\boldsymbol{a}, \boldsymbol{b} \in \mathcal{C}$.

gradient-bounded, so there exists $L_{\mathcal{R}} > 0$ such that $|\mathcal{R}(t_1) - \mathcal{R}(t_2)| = |t_1 - t_2| \cdot |\mathcal{R}'(\xi)| \leq L_{\mathcal{R}}|t_1 - t_2|$. Next, we consider the Lipschitz-continuity of the function $\omega(\boldsymbol{t}) = \left(\frac{1}{h}\sum_{i=1}^{h}\left[\mathcal{R}(t_i)\right]^p\right)^{1/p}$ by showing its bounded gradient. Specifically,

$$\begin{aligned}\frac{\partial\omega(\boldsymbol{t})}{\partial t_i} &= (1/p)\Big(\frac{1}{h}\sum\nolimits_{i=1}^{h}\left[\mathcal{R}(t_i)\right]^p\Big)^{(1-p)/p}\cdot(1/h)p\left[\mathcal{R}(t_i)\right]^{p-1}\mathcal{R}'(t_i)\\ &\leq B^{-(1-p)^2}\mathcal{R}'(t_i)/h\\ &\leq B^{-(1-p)^2}L_{\mathcal{R}}/h,\end{aligned} \tag{12}$$

where $B > 0$ is the upper bound of the restriction function. Therefore, we have that $\|\nabla\omega(\boldsymbol{t})\|_2 \leq B^{-(1-p)^2}L_{\mathcal{R}}/\sqrt{h} < B^{-(1-p)^2}L_{\mathcal{R}}$, and thus we have $|\omega(\boldsymbol{t}) - \omega(\widetilde{\boldsymbol{t}})| = |(\boldsymbol{t} - \widetilde{\boldsymbol{t}})^\top\nabla\omega(\boldsymbol{t} + \xi(\widetilde{\boldsymbol{t}} - \boldsymbol{t}))| \leq B^{-(1-p)^2}L_{\mathcal{R}}\|\boldsymbol{t} - \widetilde{\boldsymbol{t}}\|_2$. Based on the Lipschitz constants (L_0, L_1, L_2, and $L_{\mathcal{R}}$) of $\boldsymbol{\varphi}(\cdot)$, $\ell(\cdot)$, $\Omega(\cdot)$, and $\mathcal{R}(\cdot)$, we can obtain that our learning objective $\mathcal{F}(\boldsymbol{\varphi})$ is always Lipschitz continuous such that $|\mathcal{F}(\boldsymbol{\varphi}) - \mathcal{F}(\widetilde{\boldsymbol{\varphi}})| \leq (2L_0L_1B^{-(1-p)^2}L_{\mathcal{R}} + \lambda L_2)\|\boldsymbol{\varphi} - \widetilde{\boldsymbol{\varphi}}\|_2$ for any two different $\boldsymbol{\varphi}$ and $\widetilde{\boldsymbol{\varphi}}$ (the detailed calculations are given in the Appendix). Therefore, $\mathcal{F}$ is Lipschitz continuous and the iterations of Algorithm 1 can converge to a stationary point according to the convergence property of SGD. To be more specific, for the iteration sequence $\boldsymbol{\varphi}^{(1)}, \boldsymbol{\varphi}^{(2)}, \ldots, \boldsymbol{\varphi}^{(T)}$ iterated with a specific learning rate η, we have that

$$\min_{1\leq t\leq T}\mathbb{E}[\|\nabla\mathcal{F}(\boldsymbol{\varphi}^{(t)})\|^2] \leq \tau\sqrt{2L(\mathcal{F}(\boldsymbol{\varphi}^{(1)}) - \mathcal{F}(\boldsymbol{\varphi}^*))/T}, \tag{13}$$

where $\tau, L > 0$ are the upper bound and Lipschitz constant of $\mathcal{F}(\cdot)$, respectively. From the above Eq. (13), we know that the gradient gradually approaches to zero with the increase of the iteration number T. The iteration algorithm converges to a stationary point of the learning objective $\mathcal{F}$ with a convergence rate $\mathcal{O}(1/\sqrt{T})$.

Algorithm 1 Boundary-Restricted Metric Learning via SGD.

Input: Training data pairs $\mathcal{X} = \{(\boldsymbol{x}_j, \widehat{\boldsymbol{x}}_j)|j = 1, 2, \ldots, N\}$; labels $\{y_j\}_{j=1}^N$; batch size K; learning rate $\eta > 0$; regularization parameter $\lambda > 0$.
Initialize: $t = 1$; randomize $\boldsymbol{\varphi}^{(1)}$.
Repeat:

1). Uniformly randomly pick K data pairs $\{(\boldsymbol{x}_{k_j}, \widehat{\boldsymbol{x}}_{k_j})|j = 1, 2, \ldots, K\}$ from the training set $\mathcal{X}$;
2). Update the learning parameter $\boldsymbol{\varphi}$ by

$$\boldsymbol{\varphi}^{(t+1)} := \boldsymbol{\varphi}^{(t)} - \eta\left(\frac{1}{K}\sum\nolimits_{i=1}^{K}\nabla_{\boldsymbol{\varphi}}\ell_{k_i} + \lambda\nabla_{\boldsymbol{\varphi}}\Omega\right);$$

3). Update $t := t + 1$;

Until Converge.
Output: The converged $\boldsymbol{\varphi}^*$.

Computational Complexity It is easy to find that the main time consumption in Algorithm 1 is the calculation of K gradients. For each gradient $\nabla_{\boldsymbol{\varphi}}\ell_{k_i} + \lambda\nabla_{\boldsymbol{\varphi}}\Omega$, both two terms $\nabla_{\boldsymbol{\varphi}}\ell_{k_i}$ and $\nabla_{\boldsymbol{\varphi}}\Omega$ have the complexity of $\mathcal{O}(dh)$ because $\boldsymbol{\varphi}$ is a projection from d-dimensional space to h-dimensional space. Therefore, the total computational complexity of each iteration is $\mathcal{O}(Kdh)$, where K, d, h are the batch-size, the dimensionality of sample space, and the dimensionality of feature space, respectively. Such a computational complexity is the same as most existing metric learning algorithms (Sohn, 2016; Chen et al., 2019b).

4 Theoretical analyses

This section provides further in-depth theoretical analyses for our proposed BRM. In detail, we first consider the geometric property of BRM based on the topological definition of a distance metric. After that, we demonstrate the fitting and generalization capabilities of the BRM based learning model. Overall, BRM could make a good balance between fitting and generalization (i.e., the *bias-variance tradeoff*), and thus successfully guaranteeing the effectiveness of our method. All proofs of theorems are given in the Appendix.

4.1 Topological property preservation

It is well-known that the concept of metric is originally constructed in the topology community (Kelley, 2017), where a *topological metric* [4] is defined as the distance function satisfying the non-negativity, symmetry, triangle, and coincidence properties. As an extended *metric*, the *pseudo-metric* merely has the first three properties as revealed in (Paassen et al., 2018; Ting et al., 2019).

The topological definition of metric intrinsically guarantees the geometric soundness of a distance function. Further speaking, in metric learning algorithms, a well-defined learning metric with geometric soundness could reasonably measure the distances between pairwise examples of real-world data (Yang et al., 2013). Therefore, here we want to demonstrate that our proposed BRM well preserves the topological property by the following Theorem 2, and thereby naturally guaranteeing a strict metric space.

Theorem 2 *For any feature mapping $\boldsymbol{\varphi}$ learned from $\mathcal{H}$ and the corresponding distance function $\mathcal{D}_{\boldsymbol{\varphi}}(\cdot,\cdot)$ defined in Eq. (7), we have that*: *I*). *$\mathcal{D}_{\boldsymbol{\varphi}}(\cdot,\cdot)$ is a pseudo-metric if the derivative of the restriction function $\mathcal{R}$ is monotonically decreasing*; *II*). *$\mathcal{D}_{\boldsymbol{\varphi}}(\cdot,\cdot)$ is a metric if and only if $\mathcal{D}_{\boldsymbol{\varphi}}(\cdot,\cdot)$ is a pseudo-metric with an invertible learned mapping $\boldsymbol{\varphi}$.*

Note that the condition of monotone derivative can be easily satisfied (e.g., all functions listed in Table 1). The above theoretical result reveals that our learned BRM is always a topological pseudo-metric no matter how the training data and learning objective change. This conclusion is completely consistent with the traditional linear and nonlinear metric learning (Suarez et al., 2018). Furthermore, we can find that the 4th topological property (i.e., coincidence property) could also be satisfied when the mapping $\boldsymbol{\varphi}$ is invertible. This

[4] The distance function $D(\cdot,\cdot)$ is a metric if and only if it satisfies the four conditions $\forall\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\boldsymbol{\alpha}_3 \in \mathbb{R}^d$: **(I).** Non-negativity: $D(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2) \geq 0$; **(II).** Symmetry: $D(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2) = D(\boldsymbol{\alpha}_2,\boldsymbol{\alpha}_1)$; **(III).** Triangle: $D(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2) + D(\boldsymbol{\alpha}_2,\boldsymbol{\alpha}_3) \geq D(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_3)$; **(IV).** Coincidence: $D(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2) = 0 \Leftrightarrow \boldsymbol{\alpha}_1 = \boldsymbol{\alpha}_2$.

coincidence property of our BRM is also consistent with the traditional metric. It means that such a property is inherited from the traditional metric and is well preserved in our BRM.

As we already theoretically proved the topological property preservation of BRM, here we would like to further provide some intuitive understandings on how our proposed BRM preserves those topological properties of the original Euclidean distance. Simply speaking, this result mainly benefits from the *monotonicity* and *boundedness* of the restriction function $\mathcal{R}$. Although we use an upper-bounded function to explicitly constrain the divergence between the feature embeddings $\boldsymbol{\varphi}(\boldsymbol{x})$ and $\boldsymbol{\varphi}(\widehat{\boldsymbol{x}})$ on each element (i.e., $|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|$ for $i = 1, 2, \ldots, h$), the restricted result $\mathcal{R}(|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|)$ is still monotonically increasing *w.r.t.* the divergence $|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|$. It means that the size relationship of divergence can be completely preserved after the distance calculation of $\mathcal{R}(|\varphi_i(\boldsymbol{x}) - \varphi_i(\widehat{\boldsymbol{x}})|)$, so that the key topological properties that mainly depend on the size relationship of distance can also be satisfied in our new BRM metric.

The topological properties of metrics ensure the theoretical soundness of metric learning algorithms and are essentially important for properly predicting distances. For example, the distance from $\boldsymbol{x}$ to $\widehat{\boldsymbol{x}}$ and the distance from $\widehat{\boldsymbol{x}}$ to $\boldsymbol{x}$ should be the same value in most cases, or switching the order of distance calculations will lead to (unreasonable) different classification results. Therefore, the symmetry property is necessary for a distance metric. Meanwhile, the non-negativity property ensures that the sum of multiple distances will not be a smaller value which may lead to contradictory results in the loss evaluation. Furthermore, we suppose that examples $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are from the same class, and the example $\boldsymbol{x}_3$ is from another class. As the metric learning algorithm is designed to repel negative pairs of examples, the distances $d(\boldsymbol{x}_1, \boldsymbol{x}_3)$ and $d(\boldsymbol{x}_2, \boldsymbol{x}_3)$ are expected to be as large as possible. However, when the distance function $d(\cdot, \cdot)$ loses its triangle property, it may lead to that $d(\boldsymbol{x}_1, \boldsymbol{x}_2) > d(\boldsymbol{x}_1, \boldsymbol{x}_3) + d(\boldsymbol{x}_2, \boldsymbol{x}_3)$, which unreasonably enlarges the intra-class distance $d(\boldsymbol{x}_1, \boldsymbol{x}_2)$, and this is against the purpose of reducing distances between positive pairs of examples.

4.2 Fitting capability guarantee

In this subsection, we investigate the fitting capability of our proposed BRM. Intuitively, as we mentioned that BRM restricts the learning metric space, the model fitting ability seems to be weakened by the restriction function. Here we prove that although the metric space is restricted, it is still capable of *distinguishing the data pair similarities* and thereby rendering an accurate fitting result.

We suppose that the traditional metric $d_{\boldsymbol{\varphi}}(\cdot, \cdot)$ learned from the hypothesis space $\mathcal{H}$ [the ℓ_p-norm based form in Eq. (1)] is able to fit a given dataset. Then we investigate whether there exists the new learning parameter $\widehat{\boldsymbol{\varphi}}$ such that $\mathcal{D}_{\widehat{\boldsymbol{\varphi}}}(\cdot, \cdot)$ is still capable of distinguishing the data pair similarities correctly. We have the following Theorem 3 revealing the discriminability of BRM on the dataset $\mathcal{X} = \mathcal{X}^+ \cup \mathcal{X}^-$.

Theorem 3 *Assume that the original metric $d_{\boldsymbol{\varphi}}(\cdot, \cdot)$ with a learned $\boldsymbol{\varphi} \in \mathcal{H}$ satisfies that $d_{\boldsymbol{\varphi}}(\boldsymbol{x}^-, \widehat{\boldsymbol{x}}^-) \geq v > u \geq d_{\boldsymbol{\varphi}}(\boldsymbol{x}^+, \widehat{\boldsymbol{x}}^+)$ for any $(\boldsymbol{x}^-, \widehat{\boldsymbol{x}}^-) \in \mathcal{X}^-$, $(\boldsymbol{x}^+, \widehat{\boldsymbol{x}}^+) \in \mathcal{X}^+$, and given $v > u > 0$. If $\mathcal{R}'(0) \neq 0$, then there exists $\widehat{v} > \widehat{u} > 0$, and $\widehat{\boldsymbol{\varphi}}(\cdot) = c\boldsymbol{\varphi}(\cdot)$ such that*

$$\mathcal{D}_{\widehat{\boldsymbol{\varphi}}}(\boldsymbol{x}^-, \widehat{\boldsymbol{x}}^-) \geq \widehat{v} > \widehat{u} \geq \mathcal{D}_{\widehat{\boldsymbol{\varphi}}}(\boldsymbol{x}^+, \widehat{\boldsymbol{x}}^+), \tag{14}$$

where the data pairs $(\boldsymbol{x}^-, \hat{\boldsymbol{x}}^-) \in \mathcal{X}^-$ *and* $(\boldsymbol{x}^+, \hat{\boldsymbol{x}}^+) \in \mathcal{X}^+$. *Here* $c \in \mathbb{R}_+$ *is the rescaling parameter*.

From the above Eq. (14), we can observe an interesting phenomenon that the learning parameter $\boldsymbol{\varphi}$ can be rescaled to the new learning parameter of BRM, and the obtained metric still predicts the pairwise similarities correctly on the training data. Such an important result implies that BRM does not essentially sacrifice the intrinsic fitting capability of learning parameters, even though the metric space is restricted.

4.3 Generalization error bound

In Sect. 3.1 and Sect. 3.2, we have already demonstrated that the expected prediction accuracy can be hurt by the increased boundary of metric space [i.e., the FPR values in Eqs. (6) and (8)]. Now we further investigate the bias between the empirical risk and expected (generalization) risk *w.r.t.* a given loss function. Such a bias could quantitatively evaluate the model generalizability when the metric model is learned with a specific empirical loss function.

As we know that the *generalization error bound* (GEB) usually has a convergence rate of $\mathcal{O}(1/\sqrt{N})$ for an ERM model, where N is the sample size (Ye et al., 2019a; Luo et al., 2019). Here we are not going to offer a faster convergence rate *w.r.t.* the sample size, but showing a tightened GEB result benefited from the bounded metric space for validating the effectiveness of BRM. Specifically, for the underlying data distribution $\mathcal{P}$, we denote the expected risk $\widetilde{\mathcal{L}}(\boldsymbol{\varphi}) = \mathbb{E}_{(\boldsymbol{x},\hat{\boldsymbol{x}})\sim\mathcal{P}}[\ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x},\hat{\boldsymbol{x}});y_{(\boldsymbol{x},\hat{\boldsymbol{x}})})]$ and discuss how far it is from the empirical risk $\mathcal{L}(\boldsymbol{\varphi}) = \frac{1}{N}\sum_{i=1}^{N} \ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i,\hat{\boldsymbol{x}}_i);y_i)$. The corresponding result is described as follow.

Theorem 4 *For any* $\boldsymbol{\varphi}$ *learned within the hypothesis space* $\mathcal{H}$ *and any* $\delta \in (0,1)$, *we have that with probability* $1-\delta$

$$|\mathcal{L}(\boldsymbol{\varphi}) - \widetilde{\mathcal{L}}(\boldsymbol{\varphi})| \leq \theta(B)\sqrt{[\ln(2/\delta)]/(2N)}, \tag{15}$$

where $\theta(B) = \max(\ell((1-c_1)B;1), \ell(-c_0B;0))$ *is monotonically increasing. Here* $0 < c_0 < c_1 < 1$ *determines thresholds* c_1B *and* c_0B *for positive pairs and negative pairs, respectively*.

From the above GEB result in Eq. (15), it is easy to observe that the error bound is dominated by two main aspects. Firstly, the error bound gradually decreases with the increase of the sampling number N, and this is consistent with the traditional GEB result. More importantly, we can also find that such an error bound becomes *tight* when the boundary B is *decreased*, and thereby the bounded metric space would assist the expected risk in converging to the empirical risk. Therefore, Theorem 4 clearly demonstrates the usefulness of metric space restriction for improving the model generalizability during the test phase.

5 Experimental results

In this section, we show experimental results on both synthetic and real-world datasets to validate the effectiveness of BRM. In detail, we first give visualization results and ablation studies on synthetic data to demonstrate the usefulness of BRM. After that, we compare our proposed learning algorithm with existing state-of-the-art linear models on

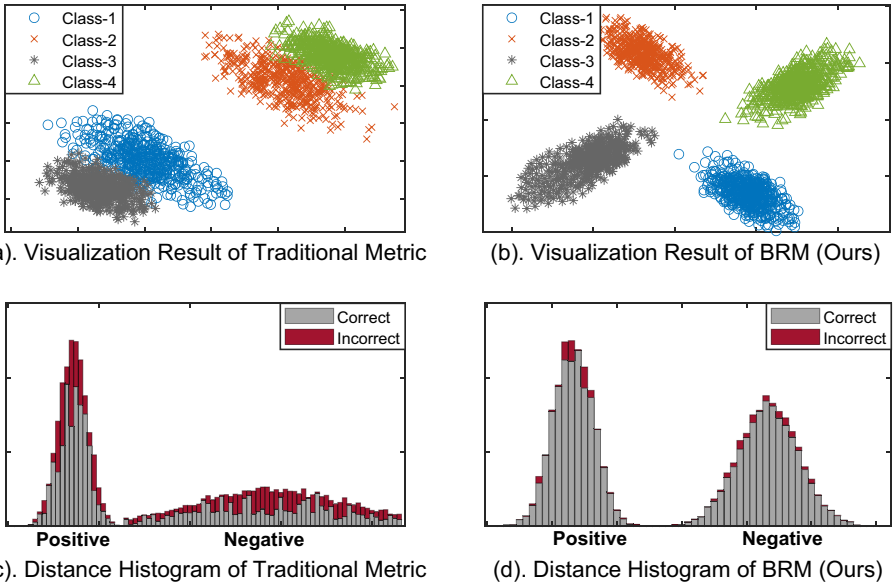


Fig. 6 Visualized learning results [(**a**) and (**b**)] and the corresponding predicted distance histograms [(**c**) and (**d**)] of the traditional method (ITML) and our proposed BRM. Two methods share the same learning objective and learning parameters (i.e., the projection matrix $\boldsymbol{P}$)

classification and verification tasks. Next, we further implement our BRM in an end-to-end neural network and also compare it with representative DNN based metric learning approaches. Finally, we investigate the robustness of our proposed method on noisy (corrupted) datasets. We adopt Sigmoid as the restriction function of BRM and fix the p value in Definition 1 to 2 in our experiments. Here we did not use additional loss functions for our BRM, and our proposed new metric is directly integrated into the existing contrastive similarity loss, relative similarity loss, or Npairs loss for implementations, so we do not need to tune the weights of different loss functions.

It is worth pointing out that the original distance in Eq. (1) parameterized by the learned embedding $\boldsymbol{\varphi}$ is not necessarily bounded, so we still need to calculate the function $\mathcal{R}$ in the test phase to obtain explicitly bounded distances. However, the computational complexity of distance in Eq. (7) is independent of the sample size, so its complexity can be regarded as $\mathcal{O}(1)$. Therefore, such a distance calculation will not affect the efficiency of metric learning during both the training phase and the test phase.

5.1 Experiments on synthetic data

We first construct 4 classes of data points in 10-dimensional space. Each class contains 500 data points sampled from the 10-dimensional normal distribution with a diagonal covariance matrix. On such a synthetic dataset composed of 4×500 data points, we employ the classical *information theory metric learning* (ITML) (Davis et al., 2007; Suárez et al., 2020) as a baseline method to validate the effectiveness of BRM. Specifically, ITML learns a traditional ℓ_p-norm based metric $\|\boldsymbol{Px} - \boldsymbol{P\hat{x}}\|_2^2$ ($\boldsymbol{P} \in \mathbb{R}^{2\times 10}$) to measure the distance

Table 2 Training/test error rates (%, mean ± std, 20 random trials) of the baseline method and our method on the synthetic datasets

Setting	Traditional metric	Boundary-restricted metric	t-test
$\rho = 1$	$6.91 \pm 3.28/9.78 \pm 6.38$	$9.98 \pm 2.98/\mathbf{9.11 \pm 7.11}$	–
$\rho = 2$	$1.12 \pm 0.12/4.25 \pm 2.33$	$1.32 \pm 1.02/\mathbf{2.31 \pm 1.14}$	$\surd$
$\rho = 4$	$0.21 \pm 0.02/3.34 \pm 1.42$	$0.12 \pm 0.07/\mathbf{1.56 \pm 0.96}$	$\surd$

Bold indicates the best test results

between $\boldsymbol{x}$ and $\widehat{\boldsymbol{x}}$, while our method learns the metric $\frac{1}{h}\sum_{i=1}^{h}[\mathcal{R}(|(\boldsymbol{P}\boldsymbol{x})_i - (\boldsymbol{P}\widehat{\boldsymbol{x}})_i|)]^2$ with $h = 2$ for visualizing the projection results in 2D space. Both methods employ the squared hinge loss for model training with same hyper-parameter settings. Here 60% of all data is randomly selected for training, and the rest is used for test.

Visualization. As shown in Fig. 6, four classes of points can be separated by both two methods in their training phases. However, the inter-class distances predicted by the traditional method are imbalanced in Fig. 6a, e.g., the distance between Class-2 and Class-4 is significantly smaller than the one between Class-3 and Class-4. In this case, such a small margin between Class-2 and Class-4 is almost negligible and its discriminability would be weakened in the test phase. As a result, the data points belonging to Class-2 and Class-4 are potentially treated as a single class, and thus incurring incorrect predictions (i.e., small inter-class distances and large intra-class distances) in Fig. 6c. In comparison, our method obtains a more balanced separation, and all inter-class distances are comparable in Fig. 6b. Accordingly, its distance histogram in the test phase is more accurate than ITML, as shown in Fig. 6d.

Ablation Study. We adopt the k-NN classifier ($k = 5$) based on the learned metrics to further investigate the classification accuracy of the baseline method and our method over 20 random trials. Here the distances between the centers of every two classes in original 10-dimensional space are set to ρ. As shown in Table 2, two methods obtain comparable training accuracy. By integrating BRM into the objective of traditional metric learning, the classification performance can be effectively enhanced. We also perform the t-test at significance level 0.05 in the last column. It clearly demonstrates that the introduction of BRM could significantly improve the classification accuracy of the traditional metric learning algorithm.

5.2 Classification and verification experiments

In this subsection, we compare our proposed BRM with existing linear metric learning methods on both classification and verification tasks. The compared methods are *LMNN* (Weinberger et al., 2006), *ITML* (Davis et al., 2007), *GMML* (Zadeh et al., 2016), *CERML* (Huang et al., 2018), *ODML* (Xie et al., 2018), and *UM2L* (Ye et al., 2019b). For a fair comparison, BRM is also implemented by a linear feature mapping in this subsection. Furthermore, both the contrastive similarity loss and relative similarity loss [i.e., Eqs. (2) and (3)] are introduced as the learning objectives. The corresponding distance metrics learned from such two loss functions are denoted as BRM-C and BRM-R, respectively.

Classification. For the classification task, we adopt the k-NN classifier ($k = 5$) based on the learned metrics to investigate the classification error rates of various methods. The datasets are from the well-known UCI machine learning repository (Asuncion and Newman, 2007), including *Vowel, Vehicle, MNIST, German, Australia, Pima, Segment,*

Table 3 Classification error rates (%, mean ± std, 20 random trials) of all compared methods on the 10 real-world datasets

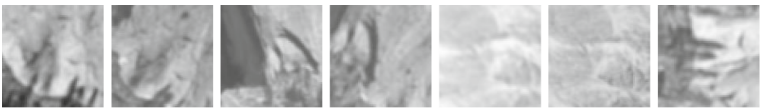
Dataset	LMNN	ITML	GMML	CERML	ODML	UM2L	BRM-C	BRM-R
Vowel (990, 14, 11)	47.46 ± 7.32	43.32 ± 5.32	41.56 ± 2.07	39.26 ± 5.12	42.12 ± 5.22	38.36 ± 2.32	**35.86 ± 3.12**	**34.86 ± 1.22**
Vehicle (846, 18, 4)	23.92±3.32	33.62±3.21	24.95±1.52	19.02±3.01	25.32 ± 3.22	20.36 ± 3.02	**15.51 ± 3.12**	**15.51 ± 2.11**
MNIST (4000, 784, 10)	17.46 ± 5.32	14.32 ± 7.32	11.26 ± 1.07	12.32 ± 5.34	12.09 ± 0.22	13.36 ± 2.32	**8.86 ± 1.12**	**8.01 ± 0.82**
German (1000, 24, 2)	28.51 ± 2.53	28.52 ± 2.13	27.12 ± 5.11	25.54 ± 1.23	26.23 ± 4.12	28.26 ± 6.22	**22.67 ± 0.34**	**21.11 ± 0.25**
Australia (690, 14, 2)	15.51 ± 2.53	17.52 ± 2.13	**14.36 ± 2.17**	18.26 ± 6.22	16.23 ± 4.12	17.67 ± 6.34	15.72 ± 6.11	15.98 ± **7.12**
Pima (768, 8, 2)	27.12 ± 2.32	28.11 ± 3.28	27.01 ± 4.72	24.02 ± 1.22	28.26 ± 1.22	25.11 ± 0.12	**20.31 ± 1.72**	**21.31 ± 0.18**
Segment (2310, 19, 7)	2.73 ± 0.82	5.16 ± 2.22	**2.77 ± 0.32**	5.36 ± 3.12	3.76 ± 1.34	2.91 ± 1.12	2.81 ± 0.02	**2.21 ± 0.12**
USPS (9298, 256, 10)	2.93 ± 0.62	2.31 ± 0.21	2.12 ± 0.97	2.12 ± 0.22	2.63 ± 0.22	2.45 ± 0.45	**1.98 ± 0.05**	**1.98 ± 0.03**
IsoLet (7794, 617, 26)	3.23 ± 1.32	8.33 ± 2.12	6.43 ± 2.12	5.49 ± 1.82	**2.58 ± 0.82**	3.26 ± 0.92	3.17 ± 0.78	3.03 ± 0.97
Letters (20000, 16, 26)	3.51 ± 2.05	6.24 ± 0.32	3.44 ± 1.04	4.67 ± 1.32	4.88 ± 0.89	4.32 ± 2.22	**1.52 ± 0.22**	**1.42 ± 0.02**
Win/Tie/Loss	7/3/0	7/3/0	8/2/0	6/4/0	7/3/0	6/4/0	BRM-C v.s. Others	
Win/Tie/Loss	7/3/0	8/2/0	9/1/0	6/4/0	7/3/0	7/3/0	BRM-R v.s. Others	

The best results are marked in bold

The last two rows list the Win/Tie/Loss counts of our BRM versus other baseline methods with t-test at significance level 0.05

(a). Example images from "*Statue of Liberty*"

(b). Example images from "*Notre Dame*"



(c). Example images from "*Half Dome in Yosemite*"

Fig. 7 Example images from *MVS* dataset with 3 different scenarios

USPS, Isolet, and *Letters*. The sample size, data dimensionality, and class number of each dataset are shown in the first column of Table 3. We compare all methods over 20 random trials. In each trial, 80% of examples are randomly selected as the training examples, and the rest is used for testing. By following the experimental settings in (Zadeh et al., 2016), the training pairs are generated by randomly picking up $1000C(C-1)$ pairs among the training examples, where C is the number of classes. Based on the k-NN prediction results of all compared methods, the average classification error rates of all compared methods are shown in Table 3. We can find that our method outperforms most of the compared methods on the ten datasets. We further analyze the accuracy improvements from the statistical perspective. Specifically, we perform the t-test (significance level 0.05) to validate the superiority of our method to all baseline methods on each dataset. From the t-test results (Win/Tie/Loss), we can clearly observe that our method obtains significant improvements on most datasets, which demonstrates the effectiveness of our proposed BRM.

Verification. We evaluate the capabilities of all compared methods on the verification task. For each method, the learned metric is used to calculate the distance value between two test examples, and then we obtain the verification results based on some distance thresholds. Our experiments are performed on the following two datasets:

- The *PubFig* dataset includes 2×10^4 pairs of face images belonging to 140 people (Huo et al., 2016), where the first 80% pairs are selected for training and the rest is used for test.
- The *MVS* dataset (Brown et al., 2010) consists of 3×10^4 grayscale patches sampled from 3D reconstructions of the *Statue of Liberty*, *Notre Dame,* and *Half Dome in Yosemite*, which are shown in Fig. 7. We randomly sample 10^5 pairs of patches to form the training set and 10^4 pairs to form the test set.

For all compared methods, we use the high-level relative attribute descriptor (Biswas and Parikh, 2013) to extract the image features of *PubFig* face data, and employ the fully connected layer features of Siamese-CNN (Zagoruyko and Komodakis, 2015) for

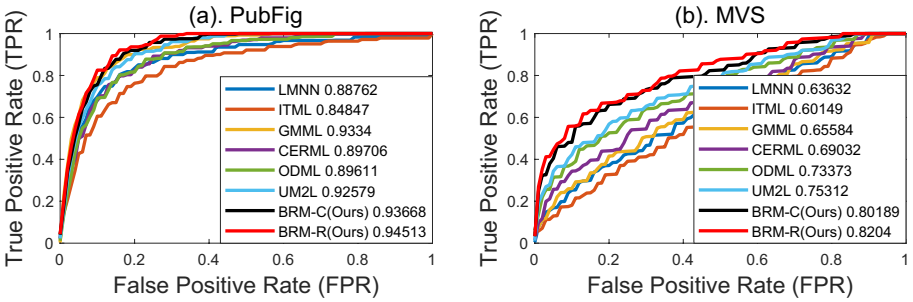


Fig. 8 ROC curves of all compared methods on the 2 verification datasets. The corresponding AUC values are presented in the legends

MVS dataset. By changing the verification thresholds of the learned distance metrics, we plot the *receiver operator characteristic* (ROC) curves of the verification results on the three datasets. The values of *area under curve* (AUC) are also calculated to evaluate the performances of all comparators quantitatively. The corresponding ROC curves and AUC values are shown in Fig. 8. It is clear to see that ODML, UM2L, and BRM-C/BRM-R consistently outperform other methods. Furthermore, we can find that our proposed BRM-C/BRM-R obtain obviously better ROC results than the best baseline method. To be specific, for the two datasets, we achieve 1–6% AUC improvements over the best baseline method.

5.3 Retrieval and clustering experiments

In this subsection, we investigate the capability of BRM on more challenging object recognition datasets. Considering the difficulty of the object recognition task, here we compare our method with representative deep neural network based metric learning approaches instead of the linear learning algorithms in the previous subsection. The compared methods are *Npairs* (Sohn, 2016), *MDR* (Kim and Park, 2021), *MS* (Wang et al., 2019), *SoftTriple* (Qian et al., 2019), *JDR* (Chu et al., 2020), and *NASA* (Li et al., 2022). To fairly evaluate the performance of all compared methods, we conduct K-means and k-NN on the learned distance metrics for clustering and retrieval tasks, respectively. By following the experimental protocol in (Sohn, 2016), we compare all methods on the two benchmark datasets below:

- Stanford *CAR-196* (Krause et al., 2013) dataset is composed of 16185 car images from 196 categories. The first 98 categories are used for training and the rest is for testing.
- Caltech-UCSD Birds (*CUB-200*) (Welinder et al., 2010) dataset is composed of 11,788 images of birds from 200 different species. Similarly, we use the first 100 categories for training and the rest for testing.
- Stanford Online Product (*SOP*) (Oh Song et al., 2016) dataset is composed of 120,053 images from 22,634 categories, and is partitioned into 11,318 categories for training and 11,316 categories for testing.

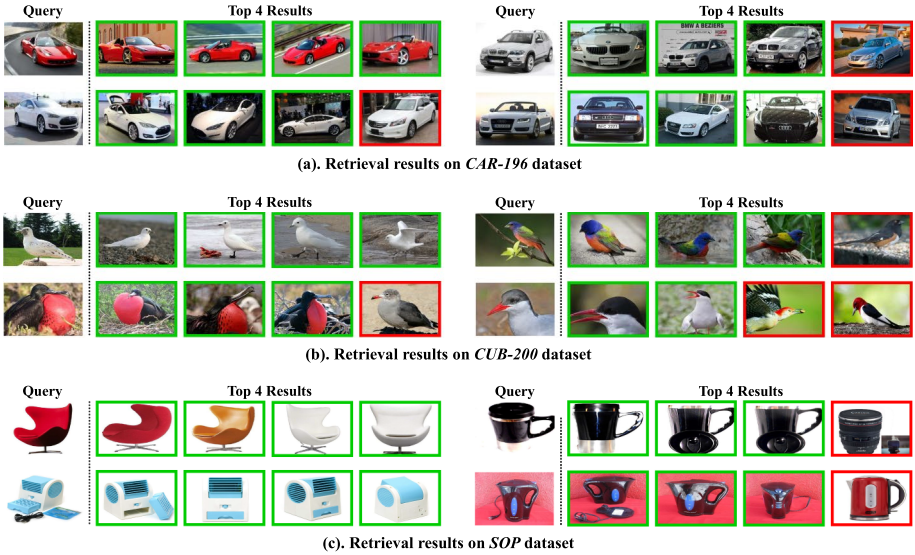


Fig. 9 The 4 nearest neighbor (top 4) results of our method BRM on *CAR-196* and *CUB-200* datasets for image retrieval. The green box means that the retrieval result is correct, and the red box denotes that the retrieval result is incorrect (Color figure online)

In our experiments, BRM is incorporated into the BN-Inception backbone (Ioffe and Szegedy, 2015) to train an end-to-end network as the final distance metric. Here we employ contrastive similarity loss, relative similarity loss, and npairs loss for learning objectives of our proposed metric, and the learned metrics are denoted as BRM-C, BRM-R, and BRM-N, respectively. For pre-processing, input images are resized to 256×256, randomly cropped to size 227×227. The dimensionality of embedding is set to 512. The learning rate and the batch size of SGD algorithm are set to 0.01 and 256,respectively. For each dataset, we perform K-means based on the learned distance metric of each method. Since the performance of K-means heavily depends on the initialization, we repeat the clustering 20 times independently and record the average results. To evaluate the consistency between clustering results Y and given class labels Y_0, we use the popular *normalized mutual information* (NMI) score

$$\mathrm{NMI}(Y_0, Y) = 2I(Y_0, Y)/(H(Y_0) + H(Y)), \tag{16}$$

where $H(\cdot)$ and $I(\cdot, \cdot)$ are entropy and mutual information, respectively. After that, we evaluate the retrieval performance of all compared methods by calculating the k nearest neighbors based on their learned distance metrics. We record the percentage of the testing examples whose k nearest neighbors contain at least one example of the same class. This quantity is also known as Recall@R (Ye et al., 2019b). Some retrieval results of our BRM on two datasets are illustrated in Fig. 9, where the hard examples can be successfully retrieved by our method. In summary, the NMI and Recall@R scores of all compared methods are shown in Table 4. From the quantitative results, we clearly observe that MDR, NASA, and our methods obtain better results than other baseline methods. Compared with both the Euclidean and cosine based approaches, our methods (i.e., BRM-C, BRM-R, and

Table 4 Clustering and retrieval performance (NMI and Recall@*R* scores) of all methods implemented by BN-Inception network on *CAR-196*, *CUB-200*, and *SOP* datasets

Method	Dim.	*CAR-196*				*CUB-200*				*SOP*			
		NMI	R@1	R@4	R@8	NMI	R@1	R@4	R@8	NMI	R@1	R@10	R@100
Npairs (Sohn, 2016)	128	58.12	68.16	80.13	84.50	59.12	58.12	71.56	78.72	80.12	69.88	81.12	85.86
	512	69.20	81.87	93.47	96.54	69.73	64.58	84.73	91.12	91.11	75.11	88.43	91.15
Npairs (min-max normalization)	128	58.22	68.26	82.81	86.55	60.23	58.22	72.66	78.52	82.01	71.05	81.02	86.21
	512	68.89	82.57	94.97	96.92	69.53	64.52	85.63	91.15	91.07	75.12	88.23	91.25
Npairs (cosine)	128	58.32	68.36	82.97	86.01	58.88	58.12	71.56	77.92	80.12	70.23	81.12	85.27
	512	69.50	82.37	94.55	95.92	69.21	64.48	84.25	90.89	91.09	76.21	87.49	92.08
MDR (Kim and Park, 2021)	128	70.12	84.21	94.11	96.91	68.98	63.32	83.82	90.40	91.01	77.01	89.63	95.45
	512	71.20	88.41	95.37	97.10	70.73	68.51	86.52	91.22	91.11	80.11	91.43	96.41
MS (Wang et al., 2019)	128	61.12	73.77	81.01	84.23	59.32	57.62	68.78	78.12	81.05	70.43	82.12	87.02
	512	70.23	84.07	94.12	96.53	70.57	66.14	85.43	91.26	91.42	76.29	89.38	95.58
SoftTriple (Qian et al., 2019)	128	60.52	73.69	81.21	83.96	64.98	57.86	69.21	77.01	81.20	70.92	82.27	86.86
	512	70.21	84.49	94.51	96.90	65.02	65.38	84.52	90.37	92.02	78.30	90.28	95.88
JRD (Chu et al., 2020)	128	59.42	72.83	80.23	84.64	59.69	58.12	70.12	75.56	82.32	71.21	81.45	86.78
	512	69.45	84.74	94.41	97.21	65.21	67.94	86.23	91.31	90.21	79.21	90.53	96.01
NASA (Li et al., 2022)	128	70.22	84.01	94.31	96.71	68.98	67.02	86.01	90.14	91.01	78.11	91.43	96.45
	512	70.22	88.91	96.21	**97.35**	69.98	68.80	86.91	91.04	91.01	78.51	90.63	96.24
BRM-C (Ours)	128	70.82	83.67	94.31	96.25	68.93	67.23	85.87	90.89	91.12	78.19	91.62	96.74
	512	71.12	89.12	96.10	97.34	70.02	69.44	87.01	91.33	**92.55**	80.44	91.51	97.15
BRM-R (Ours)	128	70.32	83.94	93.86	96.20	68.65	67.35	86.12	90.89	91.32	79.21	91.21	96.78
	512	**72.28**	89.22	**97.49**	97.30	70.82	70.97	**87.54**	**92.39**	92.49	80.04	91.54	97.15
BRM-N (Ours)	128	70.28	85.21	94.32	96.30	69.26	68.03	86.22	90.01	91.56	79.23	83.78	97.01
	512	**71.82**	**89.29**	97.29	**97.35**	**71.21**	**71.14**	87.53	91.05	92.21	**80.55**	**92.52**	**97.75**

The best two results are bolded and underlined, respectively

BRM-N) can achieve further better or competitive NMI and Recall@R scores on the three datasets.

Here we also conduct experiments to compare the learned distance distributions of the traditional cosine metric and our proposed new metric BRM. As shown in Fig. 10, for the cosine metric, the variance of inter-class distance is much larger than the variance of intra-class distance, which makes the corresponding margin-range-ratio not discriminative. In comparison, for our proposed BRM, the variance of inter-class distance is effectively reduced, so a large margin-range-ratio is ensured and the final recognition performance is improved.

We also adopt ResNet-50 (with 512-dimensional features) (He et al., 2016) as an additional backbone of our method, and we compare the results with recent works including MetricFormer (Yan et al., 2022), AVSL (Zhang et al., 2022), IBC (Seidenschwarz et al., 2021), and the baseline method ProxyAnchor (Kim et al., 2020). For fair comparisons, all methods are trained by the proxy anchor loss (Kim et al., 2020), where the temperature, positive margin, and negative margin are set to 16, 1.8, and 2.2, respectively. We train all networks 100 epochs using SGD with learning rate 0.001 and batch size is fixed to 256. Based on the new implementations, we record the corresponding NMI and Recall@R scores of all compared methods in Table 5.

On the three popular benchmark datasets, our BRM successfully improves the baseline method ProxyAnchor by 1–6%. Meanwhile, we can observe that BRM can outperform the compared methods MetricFormer, AVSL, and IBC in most cases, which clearly demonstrates the superiority of our method.

Now we further validate the effectiveness of our method on the popular *vision transformer* (ViT) backbone (Dosovitskiy et al., 2020). Specifically, we adopt the ViT-S framework pre-trained on ImageNet-21k as our encoder network (Steiner et al., 2021) and its output dimensionality is 384. After that, it is further plugged into a linear projection head to obtain the 128-dimensional features. In other words, our feature embedding $\boldsymbol{\varphi}$ consists of the ViT-S encoder and an additional projection layer. Meanwhile, we adopt the well-known pairwise cross-entropy as our loss function to calculate the empirical risk between the supervisory information and the distances predicted by our BRM. Finally, we use the Adam optimizer with a learning rate 1×10^{-5}, and the batch size is set to 900. The number of optimizer steps depends on the dataset: 200 for *CUB-200*, 600 for *Cars-196*, and 25,000 for *SOP*, respectively.

Here we include two baseline methods (ViT-S + Euclidean distance, and ViT-S + cosine dissimilarity) and two state-of-the-art ViT based metric learning methods (ViT-S + hyperbolic embedding, and ViT-S + hypersphere embedding) (Ermolov et al., 2022). All methods are fairly trained with the same pairwise cross-entropy loss with the same optimizer, learning rate, and batch size. From the above Table 6, we can observe the solid performance of our method with ViT encoder, where our BRM consistently improves the two baseline methods on all three datasets by at least 3%. Meanwhile, our method can outperform the compared methods hyperbolic embedding and hypersphere embedding on NMI scores and Recall@R scores in most cases. The above experiments clearly demonstrate that our method is suitable for the ViT architecture, and it can successfully make the ViT based metric learning approaches better on image classification tasks.

Table 5 Clustering and retrieval performance of our method and baseline methods implemented by ResNet-50

Method	*CAR-196*				*CUB-200*				*SOP*			
	NMI	R@1	R@4	R@8	NMI	R@1	R@4	R@8	NMI	R@1	R@10	R@100
ProxyAnchor (Kim et al., 2020)	75.72	87.71	95.76	97.89	72.31	69.72	87.01	92.41	91.02	78.39	90.48	96.16
MetricFormer (Yan et al., 2022)	<u>76.23</u>	**91.76**	96.31	97.21	<u>75.41</u>	<u>74.42</u>	85.75	92.53	<u>92.71</u>	**82.23**	<u>92.62</u>	<u>96.33</u>
AVSL (Zhang et al., 2022)	75.86	91.51	<u>97.02</u>	**98.41**	73.21	71.91	<u>88.11</u>	<u>93.21</u>	91.21	79.61	91.40	96.40
IBC (Seidenschwarz et al., 2021)	74.8	88.11	96.21	98.21	74.01	70.32	87.61	92.72	92.61	<u>81.42</u>	91.32	95.89
BRM-Proxy (ours)	**76.53**	<u>91.62</u>	**97.21**	<u>98.32</u>	**75.89**	**75.51**	**89.20**	**95.31**	**93.30**	<u>81.42</u>	**93.45**	**97.86**

The best two results are bolded and underlined, respectively

Table 6 Clustering and retrieval performance of our method and compared methods implemented by ViT on *CAR-196, CUB-200, and SOP* datasets

Method	*CAR-196*				*CUB-200*				*SOP*			
	NMI	R@1	R@4	R@8	NMI	R@1	R@4	R@8	NMI	R@1	R@10	R@100
ViT-S (Euclidean) (Steiner et al., 2021)	70.21	76.20	91.21	92.92	67.94	80.25	91.89	93.46	90.21	82.21	91.87	95.45
ViT-S (cosine) (Steiner et al., 2021)	70.82	76.22	90.81	93.01	68.80	81.60	92.21	93.36	91.01	82.32	92.32	96.24
ViT-S + sphere (Ermolov et al., 2022)	71.12	78.51	90.93	94.32	69.44	83.21	93.62	95.85	92.46	82.53	92.95	97.44
ViT-S + hyperbolic (Ermolov et al., 2022)	71.82	82.72	93.61	**96.22**	70.91	84.00	94.21	96.41	92.55	**85.54**	94.93	98.17
BRM-C (ViT-S based, ours)	73.25	82.92	93.56	96.11	70.21	83.65	94.21	96.12	93.25	85.40	94.95	97.52
BRM-R (ViT-S based, ours)	**73.35**	82.11	93.60	96.11	71.08	83.65	94.12	96.46	93.30	85.41	95.21	98.45
BRM-N (ViT-S based, ours)	73.26	**83.11**	**94.25**	96.21	**71.10**	**84.64**	**95.02**	**96.47**	**93.32**	85.48	**95.32**	**98.50**

The best two results are bolded and underlined, respectively

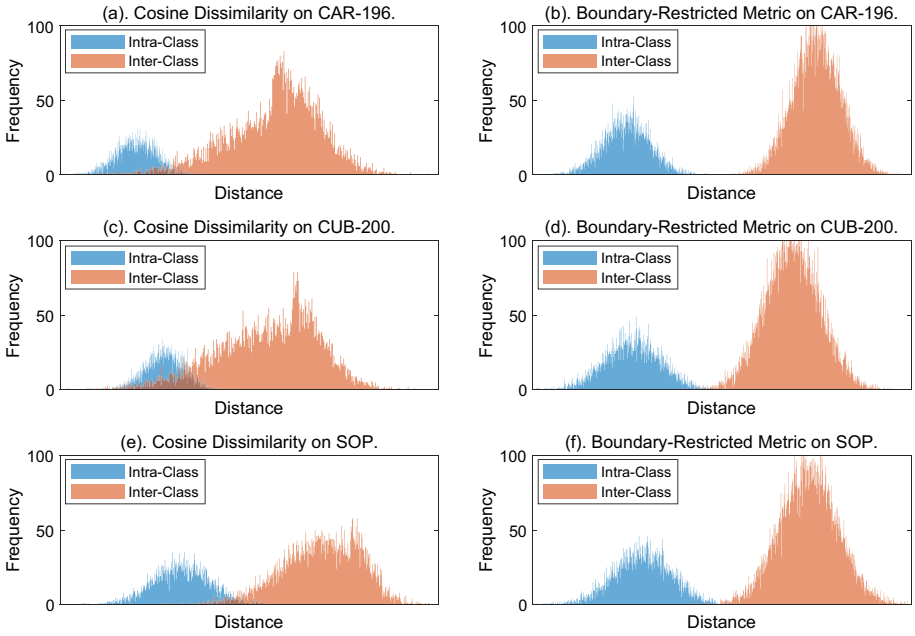


Fig. 10 Distance distributions of cosine similarity metric and our proposed boundary-restricted metric

5.4 Investigation on model robustness

In this subsection, we investigate the model robustness from two aspects, including the recognition accuracy under noisy conditions and the influence of choosing different restriction functions.

5.4.1 Robustness under noisy condition

Now we study the robustness of BRM on the corrupted data with random feature noise. Here we select three classical methods (LMNN (Weinberger et al., 2006), ITML (Davis et al., 2007), and GMML (Zadeh et al., 2016)) and three representative robust methods (*L1-ML* (Wang et al., 2014), *CAP-L1-ML* (Huo et al., 2016), and *BDML* (Xu et al., 2018)) for comparisons. We corrupt the *MNIST* and *Letters* datasets by adding Gaussian noise with 0 mean and 0.1 variance to normalized examples. After that, we conduct the classification experiments by using the same settings in Sect. 5.2 and record the classification error rates for all compared methods.

By following the experiments in existing robust metric learning methods (Xu et al., 2018), we evaluate the performance of all compared methods with clean training data and corrupted test data. After that, both training and test data are corrupted to form more challenging datasets. As shown by the error bars in Fig. 11, we can clearly observe that such noisy data is rather difficult to handle, and the classification performance of all compared methods consistently descends on the noisy data. For example, the classical methods LMNN, ITML, and GMML become significantly worse, although they have shown promising results on the clean data. In comparison, our method BRM can still achieve robust

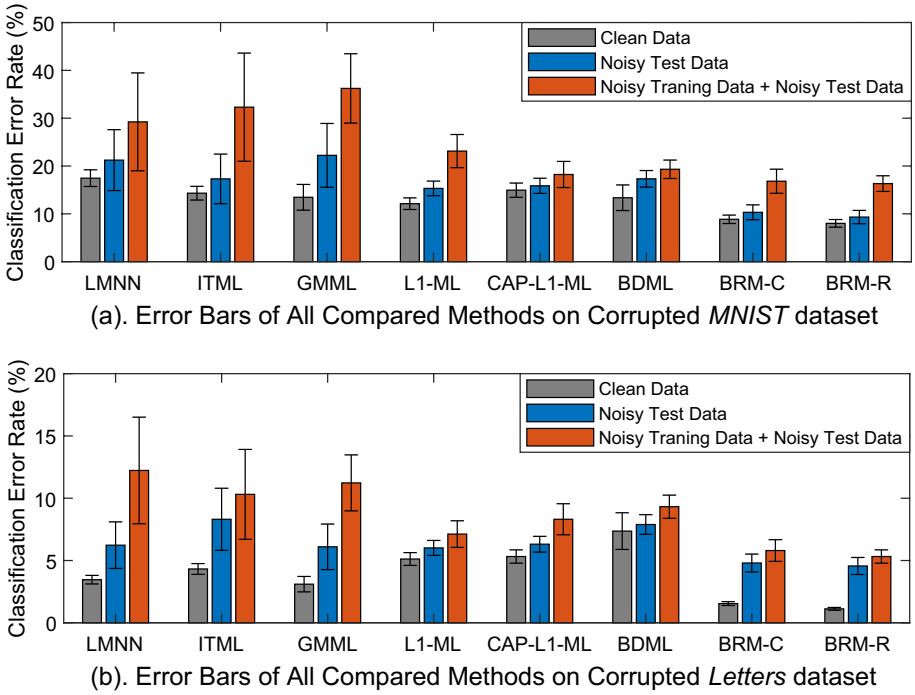


Fig. 11 Classification error rates (%) of all compared methods on the noisy *MNIST* dataset and *Letters* dataset

classification performance, especially when the training data and test data are both corrupted by random noise. This is because the restriction function confines the metric space, and the excessive distance results affected by outliers can be effectively controlled. Therefore, our proposed BRM can successfully enhance the model discriminability and obtain reliable prediction results under noisy conditions.

5.4.2 Influence of different restriction functions

In previous subsections, we fixed the restriction function $\mathcal{R}(\cdot)$ to Sigmoid for all experiments. To further investigate the model robustness, here we explore the influence from different restriction functions. Specifically, we fix the network backbone and loss function, and evaluate the recognition accuracy of our method implemented by different restriction functions listed in Table 1. For different restriction functions, here we adopt the same SGD parameters including batch size and learning rate.

Here we conduct experiments on the three benchmark datasets (*CAR-196*, *CUB-200*, and *SOP*) to investigate the influence of restriction function $\mathcal{R}$. For each dataset, we fix the network backbone (BN-Inception) and loss function (Npairs loss), and we evaluate the recognition accuracy rates of our method implemented by different restriction functions (including Sigmoid, Soft-Sign, ArcTan, TanH, and ISRU). In Table 7, we clearly observe that Sigmoid and Soft-Sign can obtain slightly higher accuracy than the other three restriction functions on all three datasets. Meanwhile, we find that all five restriction functions

Table 7 Clustering and retrieval performance of our method equipped with different restriction functions on *CAR-196, CUB-200, and SOP* datasets

Function	Dim.	*CAR-196*				*CUB-200*				*SOP*			
		NMI	R@1	R@10	R@100	NMI	R@1	R@4	R@8	NMI	R@1	R@10	R@100
Sigmoid	128	70.28	85.21	94.32	96.30	69.26	68.03	86.22	90.01	91.56	79.23	83.78	97.01
	512	**71.82**	**89.29**	97.29	**97.35**	71.21	**71.14**	**87.53**	**91.05**	**92.21**	**80.55**	**92.52**	**97.75**
Soft-Sign	128	70.12	85.01	93.72	95.94	69.36	67.93	86.24	89.92	91.55	79.02	83.58	97.11
	512	71.28	88.82	**97.43**	97.09	**71.81**	71.13	**87.53**	90.85	92.13	80.19	92.49	97.25
ArcTan	128	69.92	84.86	93.65	95.21	68.42	67.04	85.12	90.01	90.26	77.89	82.18	95.51
	512	70.89	88.79	97.01	96.85	70.02	70.41	86.31	90.66	91.12	79.21	91.22	96.25
TanH	128	69.96	84.91	93.56	95.32	68.16	66.55	84.98	90.01	90.56	78.23	81.98	96.11
	512	70.87	88.92	96.35	96.21	69.96	69.86	87.01	90.96	91.21	79.55	92.02	96.75
ISRU	128	70.02	84.92	92.99	94.96	69.95	67.35	85.22	90.11	90.56	78.99	82.21	95.55
	512	71.01	88.59	95.98	95.89	70.21	70.96	86.53	90.80	91.25	80.08	90.98	96.75

The best two results are bolded and underlined, respectively

can achieve relatively stable and comparable performance when they are adopted to implement BRM. These results are consistent with our expectations, as we merely assume that the function $\mathcal{R}$ is smooth and monotonically increasing. It means that we do not need any other preconditions for $\mathcal{R}$ to ensure the effectiveness of our method. This makes the design/selection of the function $\mathcal{R}$ much easy in practical uses.

Although the classification accuracy rates are not significantly influenced by different restriction functions, here we would still like to further discuss the selection/design of the function $\mathcal{R}$. Since we do not make additional assumptions on such an abstract function $\mathcal{R}$, it is pretty hard to theoretically analyze the superiority of each specific restriction function. However, empirically, the Sigmoid and Soft-Sign functions have shown slightly higher accuracy rates on the three benchmark datasets, so the two functions have stronger generalizability than the others. Meanwhile, the Sigmoid function has better smoothness than the Soft-Sign function (around $\mathcal{R}(0)$), and the derivative of Sigmoid is also more computation-friendly (as $\mathcal{R}' = \mathcal{R}(1 - \mathcal{R})$) during the training phase. Therefore, we encourage to use the Sigmoid function as a default setting in the implementations.

6 Conclusion and future work

This paper first derived an analytical result to reveal that the expected false positive rate would be magnified by the expansion of the metric space boundary. To overcome this issue, we proposed a new boundary-restricted metric (BRM) to confine the metric space, and we employed a monotonous function with saturation property to suppress excessively large distances and concurrently maintain the useful topological property. We conducted intensive theoretical analyses to guarantee the model effectiveness and soundness. Visualization experiments on toy data and comparison experiments on real-world datasets indicate that our learning algorithm acquires the more reliable and precise metric than state-of-the-art methods.

Nowadays, self-supervised *contrastive learning* (CL) which is based on pairwise similarities in an unsupervised manner, has attracted lots of attention and shown very powerful performance in several downstream tasks. It is still unclear whether our method can be directly applied in the self-supervised CL, but we think this is a potential improvement. It would be interesting future work if we could further consider the boundary issue in self-supervised CL approaches and investigate the effectiveness of our method in the unsupervised (self-supervised) learning scenario.

Appendix A

This section provids the detailed proofs for all theorems in Sect. 3.1 and Sect. 4.

A.1 Proof for Theorem 1

We first introduce the following *Lindeberg central limit theorem* (CLT) as a Lemma to prove our Theorem 1.

Lemma 1 (Lindeberg CLT (Vershynin, 2018)) *Suppose* $\{X_1, \ldots, X_h\}$ *is a sequence of independent random variables, each with finite expected value* μ_i *and variance* σ_i^2. *If for any given* $\epsilon > 0$

$$\lim_{h\to\infty}\frac{1}{S_h^2}\sum_{i=1}^{h}\mathbb{E}[(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>\epsilon S_n\}}]=0, \tag{17}$$

then the distribution of the standardized sum $1/S_h\sum_{i=1}^{h}(X_i-\mu_i)$ *converges to the standard normal distribution* $\mathcal{N}(0,1)$*, where* $S_h^2=\sum_{i=1}^{h}\sigma_i^2$ *and* $1_{\{\cdot\}}$ *is the indicator function.*

Based on the above conclusion on *i.n.i.d.* random variables $X_1, X_2, \ldots, X_h$, here we prove Theorem 1 by investigating the probability of that the distance value crosses a given upper-bound. We show that such a probability is mainly determined by the boundary of the metric space.

Proof We let $X_i=|\varphi_i(\boldsymbol{x})-\varphi_i(\widehat{\boldsymbol{x}})|^p$ for $i=1,2,\ldots,h$ and we can obtain that $\mathbb{E}[(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>\epsilon S_h\}}]\le\mathbb{E}[(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>b^p(\mathcal{H}_v^u)-\mu_i\}}]=0$ for sufficiently large h. Specifically, we denote $V^2=1/h\sum_{i=1}^{h}\sigma_i^2>0$ and let $h=\left\lceil(b^p(\mathcal{H}_v^u)-\mu_i)^2/(\epsilon V)^2\right\rceil$, and we have that

$$\epsilon S_h=\epsilon\sqrt{h}V\ge\epsilon V\sqrt{(b^p(\mathcal{H}_v^u)-\mu_i)^2/(\epsilon V)^2}=b^p(\mathcal{H}_v^u)-\mu_i, \tag{18}$$

where $b^p(\mathcal{H}_v^u)$ is the upper-bound of X_i so that $b^p(\mathcal{H}_v^u)-\mu_i$ is always non-negative (μ_i is the mean of X_i). Then, if $|X_i-\mu_i|<b^p(\mathcal{H}_v^u)-\mu_i$, we have

$$(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>\epsilon S_h\}}=0=(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>b^p(\mathcal{H}_v^u)-\mu_i\}}. \tag{19}$$

If $b^p(\mathcal{H}_v^u)-\mu_i\le|X_i-\mu_i|\le\epsilon S_h$, we have

$$(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>\epsilon S_h\}}=0\le(X_i-\mu_i)^2=(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>b^p(\mathcal{H}_v^u)-\mu_i\}}. \tag{20}$$

Finally, if $\epsilon S_h<|X_i-\mu_i|$, we have

$$(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>\epsilon S_h\}}=(X_i-\mu_i)^2=(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>b^p(\mathcal{H}_v^u)-\mu_i\}}, \tag{21}$$

and thus we have $\mathbb{E}[(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>\epsilon S_h\}}]\le\mathbb{E}[(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>b^p(\mathcal{H}_v^u)-\mu_i\}}]$ for sufficiently large h. Furthermore, as $|X_i-\mu_i|$ is always small than its upper bound $b^p(\mathcal{H}_v^u)-\mu_i$, we have $\mathbb{E}[(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>b^p(\mathcal{H}_v^u)-\mu_i\}}]=0$. Therefore, we have

$$\lim_{h\to\infty}\frac{\sum_{i=1}^{h}\mathbb{E}[(X_i-\mu_i)^2\cdot 1_{\{|X_i-\mu_i|>\epsilon S_h\}}]}{S_h^2}\le\lim_{h\to\infty}\frac{h\cdot 0}{h\sigma_L^2}=0, \tag{22}$$

which implies that the *Lindeberg condition* in Eq. (17) is satisfied. Therefore, the standardized sum $\sum_{i=1}^{h}|\varphi_i(\boldsymbol{x})-\varphi_i(\widehat{\boldsymbol{x}})|^p/\overline{\sigma}$ converges to the standard normal distribution .[5] Then, for any given $\epsilon_1>0$, there exists sufficiently large h such that

[5] For simplicity, here $\overline{\sigma}^2=\frac{1}{h}\sum_{i=1}^{h}\sigma_i^2$ and $\overline{\mu}=\frac{1}{h}\sum_{i=1}^{h}\mu_i$.

$$\mathrm{pr}\left[\frac{1}{\overline{\sigma}\sqrt{h}}\sum\nolimits_{i=1}^{h}|\varphi_i(\boldsymbol{x})-\varphi_i(\widehat{\boldsymbol{x}})|^p-\mu_i\le Z\right]\in\Delta(\phi(Z),\epsilon_1),\tag{23}$$

where $Z\in\mathbb{R}$ and $\phi(\cdot)$ is the cumulative distribution function of the standard norm distribution. Meanwhile, we have

$$\mathrm{pr}[d_{\boldsymbol{\varphi}}(z,\widehat{z})<u]=\mathrm{pr}\Big[\sum\nolimits_{i=1}^{h}|\varphi_i(\boldsymbol{x})-\varphi_i(\widehat{\boldsymbol{x}})|^p<hu^p\Big],\tag{24}$$

so for any $\epsilon_1>0$, there exists sufficiently large h such that

$$\mathrm{pr}[d_{\boldsymbol{\varphi}}(z,\widehat{z})<u]\in\Delta(\phi(\sqrt{h}(u^p-\overline{\mu})/\overline{\sigma}),\epsilon_1).\tag{25}$$

As $\phi(\cdot)$ is monotonically increasing and ϵ_1 is a given sufficiently small number, $\sup_{\boldsymbol{\varphi}\in\mathcal{H}_v^u}\{\mathrm{pr}[d_{\boldsymbol{\varphi}}(z,\widehat{z})<u]\}$ is dominated by $\sqrt{h}(u^p-\overline{\mu})/\overline{\sigma}$. According to the *law of large numbers* (Vershynin, 2018), it follows that for any $\boldsymbol{\varphi}\in\mathcal{H}_v^u$ there exists a sufficiently large N making

$$|\overline{\mu}-m_N|<\epsilon_2\text{and}|\overline{\sigma}\sqrt{h}-\Sigma_N|<\epsilon_2,\tag{26}$$

with at least probability $1-\epsilon_2$, where the sample mean $m_N=(1/N)\sum_{j=1}^{N}d_{\boldsymbol{\varphi}}(z_j,\widehat{z}_j)$ and sample variance $\Sigma_N^2=((1/N))\sum_{j=1}^{N}(d_{\boldsymbol{\varphi}}(z_j,\widehat{z}_j)-m_N)^2$. Then we have that there exists sufficiently small ϵ_1 and ϵ_2 such that for any given $\delta\in\min(1,v^p-u^p)$

$$\sup_{\boldsymbol{\varphi}\in\mathcal{H}_v^u}\{\mathrm{pr}[d_{\boldsymbol{\varphi}}(z,\widehat{z})<u]\}\in\Delta\Big(\phi\Big[\sqrt{h}\big(u^p-m_N\big)/\Sigma_N\Big],\delta\Big).\tag{27}$$

Now we only have to consider the minimal value of the positive term $\big(m_N-u^p\big)/\Sigma_N$ under the constraint $d_{\boldsymbol{\varphi}}(z_j,\widehat{z}_j)>v>u$ for $j=1,2,\ldots,N$. To be specific, it holds that

$$\begin{aligned}
&\big(m_N-u^p\big)/\Sigma_N\\
&\quad=\big(m_N-v^p+v^p-u^p\big)/\Sigma_N\\
&\quad\ge\big(m_N-v^p+v^p-u^p\big)/\min(m_N-v^p,b^p(\mathcal{H}_v^u)-m_N)\\
&\quad\ge\big(m_N-v^p+v^p-u^p\big)/(m_N-v^p)\\
&\quad=(t+v^p-u^p)/t\\
&\quad=(1+(v^p-u^p)/t)\\
&\quad\ge(1+2(v^p-u^p)/(b^p(\mathcal{H}_v^u)-v^p)),
\end{aligned}\tag{28}$$

where $t=m_N-v^p\in(0,\frac{1}{2}(b^p(\mathcal{H}_v^u)-v^p)]$, and m_N is necessarily included in $(v^p,\frac{1}{2}(b^p(\mathcal{H}_v^u)+v^p))$. By combining the results in Eqs. (27) and (28), we thus get

$$\sup_{\boldsymbol{\varphi}\in\mathcal{H}_v^u}\big\{\mathrm{pr}\big[d_{\boldsymbol{\varphi}}(z,\widehat{z})<u\big]\big\}\in\Delta\big(\phi\big\{\psi\big[b(\mathcal{H}_v^u)\big]\big\},\delta\big),\tag{29}$$

where $\psi\big[b(\mathcal{H}_v^u)\big]=[\sqrt{h}(1+2(v^p-u^p)/(v^p-b^p(\mathcal{H}_v^u)))]$. Here $\psi\big[b(\mathcal{H}_v^u)\big]$ is a monotonically increasing function *w.r.t.* the boundary $b(\mathcal{H}_v^u)$. The proof is completed. □

A.2 Proof for Theorem 2

Proof The Non-negativity and Symmetry can be trivially achieved by the definition of BRM. Here we prove that $\mathcal{D}_{\boldsymbol{\varphi}}(\cdot,\cdot)$ has the triangle property when $\mathcal{R}'$ is monotonically decreasing. Specifically, for any given $\boldsymbol{\alpha},\boldsymbol{\beta},\boldsymbol{\gamma}\in\mathbb{R}^d$, we invoke the *mean value theorem* (Rudin, 1964) and have that

$$\begin{aligned}
&\mathcal{R}(|\varphi_i(\boldsymbol{\alpha})-\varphi_i(\boldsymbol{\beta})|)+\mathcal{R}(|\varphi_i(\boldsymbol{\beta})-\varphi_i(\boldsymbol{\gamma})|)\\
&\quad=\mathcal{R}(Q_i)+\mathcal{R}(T_i)-\mathcal{R}(0)\\
&\quad=\mathcal{R}(\max(Q_i,T_i))+\min(Q_i,T_i)\mathcal{R}'(\xi_1)\\
&\quad\geq\mathcal{R}(\max(Q_i,T_i))+\min(Q_i,T_i)\mathcal{R}'(\max(Q_i,T_i))\\
&\quad\geq\mathcal{R}(\max(Q_i,T_i))+\min(Q_i,T_i)\mathcal{R}'(\max(Q_i,T_i))+\Theta(\xi_2)\\
&\quad=\mathcal{R}(\max(Q_i,T_i)+\min(Q_i,T_i))\\
&\quad=\mathcal{R}(|\varphi_i(\boldsymbol{\alpha})-\varphi_i(\boldsymbol{\beta})|+|\varphi_i(\boldsymbol{\beta})-\varphi_i(\boldsymbol{\gamma})|)\\
&\quad\geq\mathcal{R}(|\varphi_i(\boldsymbol{\alpha})-\varphi_i(\boldsymbol{\gamma})|),
\end{aligned}\tag{30}$$

where the real numbers $Q_i=|\varphi_i(\boldsymbol{\alpha})-\varphi_i(\boldsymbol{\beta})|$, $T_i=|\varphi_i(\boldsymbol{\beta})-\varphi_i(\boldsymbol{\gamma})|$, $\xi_1\in[0,\min(Q_i,T_i)]$, $\xi_2\in[0,\max(Q_i,T_i)]$, and $\Theta(\xi_2)=(1/2)\min(Q_i^2,T_i^2)\mathcal{R}''(\xi_2)\leq 0$. Then we have that

$$\begin{aligned}
&\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{\alpha},\boldsymbol{\beta})+\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{\beta},\boldsymbol{\gamma})\\
&\quad=\left(\frac{1}{h}\sum\nolimits_{i=1}^{h}\left[\mathcal{R}(Q_i)\right]^p\right)^{1/p}+\left(\frac{1}{h}\sum\nolimits_{i=1}^{h}\left[\mathcal{R}(T_i)\right]^p\right)^{1/p}\\
&\quad\geq\left(\frac{1}{h}\sum\nolimits_{i=1}^{h}\left[\mathcal{R}(Q_i)+\mathcal{R}(T_i)\right]^p\right)^{1/p}\\
&\quad\geq\left(\frac{1}{h}\sum\nolimits_{i=1}^{h}\left[\mathcal{R}(|\varphi_i(\boldsymbol{\alpha})-\varphi_i(\boldsymbol{\gamma})|)\right]^p\right)^{1/p}\\
&\quad=\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{\alpha},\boldsymbol{\gamma}).
\end{aligned}\tag{31}$$

Finally, for any given $\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{\alpha},\boldsymbol{\beta})=0$, and any given $k\in\mathbb{N}_h$, we have $\mathcal{R}(|\varphi_k(\boldsymbol{\alpha})-\varphi_k(\boldsymbol{\beta})|)=0$, and thus it holds that $[\varphi_1(\boldsymbol{\alpha}),\ldots,\varphi_h(\boldsymbol{\alpha})]=[\varphi_1(\boldsymbol{\beta}),\ldots,\varphi_h(\boldsymbol{\beta})]$. By further invoking the invertibility of the mapping $\boldsymbol{\varphi}$, we have that $\boldsymbol{\alpha}=\boldsymbol{\beta}$ which completes the proof. □

A.3 Proof for Theorem 3

Proof We let $\widehat{\boldsymbol{\varphi}}(\cdot)=c\boldsymbol{\varphi}(\cdot)$ $(c>0)$ and employ the *Taylor expansion* (Rudin, 1964) on each restriction function, and we get

$$\begin{aligned}
&\mathcal{D}_{\widehat{\boldsymbol{\varphi}}}(\boldsymbol{x},\widehat{\boldsymbol{x}})\\
&\quad=\left(\frac{1}{h}\sum\nolimits_{i=1}^{h}\left[\mathcal{R}(c|\varphi_i(\boldsymbol{x})-\varphi_i(\widehat{\boldsymbol{x}})|)\right]^p\right)^{1/p}\\
&\quad=\left(\frac{1}{h}\sum\nolimits_{i=1}^{h}\left[\mathcal{R}(0)+c|\varphi_i(\boldsymbol{x})-\varphi_i(\widehat{\boldsymbol{x}})|\mathcal{R}'(0)+o(c)\right]^p\right)^{1/p}\\
&\quad=\frac{c}{h}\mathcal{R}'(0)\|\boldsymbol{\varphi}(\boldsymbol{x})-\boldsymbol{\varphi}(\widehat{\boldsymbol{x}})\|_1+o(c).
\end{aligned}\tag{32}$$

According to the *homogeneity* of vector norm (Meyer, 2000), it follows that there exists $a_1 > a_0 > 0$ such that $\forall \boldsymbol{x}, \widehat{\boldsymbol{x}} \in \mathbb{R}^d$

$$\frac{a_0}{h}\|\boldsymbol{\varphi}(\boldsymbol{x}) - \boldsymbol{\varphi}(\widehat{\boldsymbol{x}})\|_1 \leq d_{\boldsymbol{\varphi}}(\boldsymbol{x}, \widehat{\boldsymbol{x}}) \leq \frac{a_1}{h}\|\boldsymbol{\varphi}(\boldsymbol{x}) - \boldsymbol{\varphi}(\widehat{\boldsymbol{x}})\|_1, \tag{33}$$

so we have that

$$\begin{cases} \mathcal{D}_{\widehat{\boldsymbol{\varphi}}}(\boldsymbol{x}^-, \widehat{\boldsymbol{x}}^-) \geq \frac{c}{a_1}\mathcal{R}'(0) d_{\boldsymbol{\varphi}}(\boldsymbol{x}^-, \widehat{\boldsymbol{x}}^-) + o(c), \\ \mathcal{D}_{\widehat{\boldsymbol{\varphi}}}(\boldsymbol{x}^+, \widehat{\boldsymbol{x}}^+) \leq \frac{c}{a_0}\mathcal{R}'(0) d_{\boldsymbol{\varphi}}(\boldsymbol{x}^+, \widehat{\boldsymbol{x}}^+) + o(c). \end{cases} \tag{34}$$

Then for $u = a_0/\mathcal{R}'(0)$ and $v = a_1/\mathcal{R}'(0)$, we have

$$\mathcal{D}_{\widehat{\boldsymbol{\varphi}}}(\boldsymbol{x}^-, \widehat{\boldsymbol{x}}^-) \geq cv + o(c) > cu + o(c) \geq \mathcal{D}_{\widehat{\boldsymbol{\varphi}}}(\boldsymbol{x}^+, \widehat{\boldsymbol{x}}^+), \tag{35}$$

which completes the proof by letting c be sufficiently small. □

A.4 Proof for Theorem 4

We first introduce the following *McDiarmids inequality* as a Lemma to prove our Theorem 4.

Lemma 2 (McDiarmid Inequality (Meyer, 2000)) *For independent random variables $t_1, t_2, \ldots, t_n \in \mathcal{T}$ and a given function $\omega : \mathcal{T}^n \to \mathbb{R}$, if $\forall v_i' \in \mathcal{T}$ $(i = 1, 2, \ldots, n)$, the function satisfies*

$$|\omega(t_1, \ldots, t_i, \ldots, t_n) - \omega(t_1, \ldots, t_i', \ldots, t_n)| \leq \rho_i, \tag{36}$$

then for any given $\mu > 0$, it holds that $\mathrm{pr}\{|\omega(t_1, \ldots, t_n) - \mathbb{E}[\omega(t_1, \ldots, t_n)]| > \mu\} \leq 2\mathrm{e}^{-2\mu^2/\sum_{i=1}^n \rho_i^2}$.

We prove Theorem 4 by analyzing the perturbation [i.e., ρ_i in the above Eq. (36)] of the loss function $\mathcal{L}$.

Proof Firstly, we denote that

$$\omega = \frac{1}{N}\sum_{i=1}^{N} \ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i); y_i), \tag{37}$$

and

$$\omega_{(k)} = \frac{1}{N}\left[\sum_{i=1, i\neq k}^{N} \ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i); y_i) + \ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{a}_k, \widehat{\boldsymbol{a}}_k); b_k)\right], \tag{38}$$

where $(\boldsymbol{a}_k, \widehat{\boldsymbol{a}}_k)$ is an arbitrary data pair from the sample space with similarity label b_k. Then we have that

$$
\begin{aligned}
&|\omega - \omega_{(k)}| \\
&\quad = \frac{1}{N}|\ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_k, \widehat{\boldsymbol{x}}_k); y_k) - \ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{a}_k, \widehat{\boldsymbol{a}}_k); b_k)| \\
&\quad \leq \frac{1}{N}\max(\ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_k, \widehat{\boldsymbol{x}}_k); y_k), \ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{a}_k, \widehat{\boldsymbol{a}}_k); b_k)) \\
&\quad \leq \frac{1}{N}\max(\ell((1-c_1)B; 1), \ell((0-c_0)B; 0)).
\end{aligned} \tag{39}
$$

Meanwhile, we have

$$
\begin{aligned}
&\frac{1}{N}\sum_{i=1}^{N}\ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i); y_i) - \mathbb{E}_{\mathcal{X}}\left(\frac{1}{N}\sum_{i=1}^{N}\ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i); y_i)\right) \\
&\quad = \frac{1}{N}\sum_{i=1}^{N}\ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i); y_i) - \mathbb{E}_{(\boldsymbol{x}, \widehat{\boldsymbol{x}})}\left[\ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}, \widehat{\boldsymbol{x}}); y_{(\boldsymbol{x}, \widehat{\boldsymbol{x}})})\right] \\
&\quad = \mathcal{L}(\boldsymbol{\varphi}) - \widetilde{\mathcal{L}}(\boldsymbol{\varphi}).
\end{aligned} \tag{40}
$$

By Lemma 2, we let that

$$
\rho_i = \frac{1}{N}\max(\ell((1-c_1)B; 1), \ell((0-c_0)B; 0)), \tag{41}
$$

for all $i = 1, 2, \ldots, N$, and we get

$$
\begin{aligned}
&\mathrm{pr}\left\{|\mathcal{L}(\boldsymbol{\varphi}) - \widetilde{\mathcal{L}}(\boldsymbol{\varphi})| < \theta(B)\sqrt{[\ln(2/\delta)]/(2N)}\right\} \\
&\quad = 1 - 2\mathrm{e}^{-2\mu^2/\sum_{i=1}^{n}\rho_i^2} \\
&\quad \geq 1 - 2\mathrm{e}^{\frac{-2N(\theta(B)\sqrt{[\ln(2/\delta)]/(2N)})^2}{\max^2(\ell((1-c_1)B;1), \ell((0-c_0)B;0))}} \\
&\quad = 1 - 2\mathrm{e}^{-2N\left(\sqrt{[\ln(2/\delta)]/(2N)}\right)^2} \\
&\quad = 1 - 2\mathrm{e}^{-\ln(2/\delta)} \\
&\quad = 1 - \delta,
\end{aligned} \tag{42}
$$

where the real-valued monotonically increasing function $\theta(B) = \max(\ell((1-c_1)B; 1), \ell(-c_0 B; 0))$. The proof is completed. □

A.5 Proof of Lipschitz-Continuity

Here we demonstrate that our learning objective $\mathcal{F}(\boldsymbol{\varphi})$ is always Lipschitz continuous based on the Lipschitz-continuity of $\boldsymbol{\varphi}(\cdot)$, $\ell(\cdot)$, $\Omega(\cdot)$, and $\mathcal{R}(\cdot)$. To be more specific, for any two given $\boldsymbol{\varphi}$ and $\widetilde{\boldsymbol{\varphi}}$, we have

$$
\begin{aligned}
&|\mathcal{F}(\boldsymbol{\varphi}) - \mathcal{F}(\widetilde{\boldsymbol{\varphi}})| \\
&\quad = |1/N \sum\nolimits_{i=1}^{N} \ell(\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i); y_i) + \lambda \Omega(\boldsymbol{\varphi}) \\
&\qquad - 1/N \sum\nolimits_{i=1}^{N} \ell(\mathcal{D}_{\widetilde{\boldsymbol{\varphi}}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i); y_i) - \lambda \Omega(\widetilde{\boldsymbol{\varphi}})| \\
&\quad \leq L_1 \frac{1}{N} \sum\nolimits_{i=1}^{N} |\mathcal{D}_{\boldsymbol{\varphi}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i) - \mathcal{D}_{\widetilde{\boldsymbol{\varphi}}}(\boldsymbol{x}_i, \widehat{\boldsymbol{x}}_i)| + \lambda L_2 \|\boldsymbol{\varphi} - \widetilde{\boldsymbol{\varphi}}\|_2 \\
&\quad = L_1 \frac{1}{N} \sum\nolimits_{i=1}^{N} |\Big(1/h \sum\nolimits_{i=1}^{h} \left[\mathcal{R}\big(|\varphi_i(\boldsymbol{x}_i) - \varphi_i(\widehat{\boldsymbol{x}}_i)|\big)\right]^p\Big)^{1/p} \\
&\qquad - \Big(1 \sum\nolimits_{i=1}^{h} \left[\mathcal{R}\big(|\widetilde{\varphi}_i(\boldsymbol{x}_i) - \widetilde{\varphi}_i(\widehat{\boldsymbol{x}}_i)|\big)\right]^p\Big)^{1/p}| + \lambda L_2 \|\boldsymbol{\varphi} - \widetilde{\boldsymbol{\varphi}}\|_2 \\
&\quad = L_1 \frac{1}{N} \sum\nolimits_{i=1}^{N} |\omega(|\boldsymbol{\varphi}(\boldsymbol{x}_i) - \boldsymbol{\varphi}(\widehat{\boldsymbol{x}}_i)|) - \omega(|\widetilde{\boldsymbol{\varphi}}(\boldsymbol{x}_i) - \widetilde{\boldsymbol{\varphi}}(\widehat{\boldsymbol{x}}_i)|)| + \lambda L_2 \|\boldsymbol{\varphi} - \widetilde{\boldsymbol{\varphi}}\|_2 \\
&\quad \leq L_1 \frac{1}{N} \sum\nolimits_{i=1}^{N} B^{-(1-p)^2} L_{\mathcal{R}} \| |\boldsymbol{\varphi}(\boldsymbol{x}_i) - \boldsymbol{\varphi}(\widehat{\boldsymbol{x}}_i)| - |\widetilde{\boldsymbol{\varphi}}(\boldsymbol{x}_i) - \widetilde{\boldsymbol{\varphi}}(\widehat{\boldsymbol{x}}_i)| \|_2 \\
&\qquad + \lambda L_2 \|\boldsymbol{\varphi} - \widetilde{\boldsymbol{\varphi}}\|_2 \\
&\quad \leq L_1 \frac{1}{N} \sum\nolimits_{i=1}^{N} B^{-(1-p)^2} L_{\mathcal{R}} \|\boldsymbol{\varphi}(\boldsymbol{x}_i) - \boldsymbol{\varphi}(\widehat{\boldsymbol{x}}_i) + \widetilde{\boldsymbol{\varphi}}(\widehat{\boldsymbol{x}}_i) - \widetilde{\boldsymbol{\varphi}}(\boldsymbol{x}_i)\|_2 \\
&\qquad + \lambda L_2 \|\boldsymbol{\varphi} - \widetilde{\boldsymbol{\varphi}}\|_2 \\
&\quad \leq L_1 \frac{1}{N} \sum\nolimits_{i=1}^{N} B^{-(1-p)^2} L_{\mathcal{R}} \big(\|\boldsymbol{\varphi}(\boldsymbol{x}_i) - \widetilde{\boldsymbol{\varphi}}(\boldsymbol{x}_i)\|_2 + \|\boldsymbol{\varphi}(\widehat{\boldsymbol{x}}_i) - \widetilde{\boldsymbol{\varphi}}(\widehat{\boldsymbol{x}}_i)\|_2\big) \\
&\qquad + \lambda L_2 \|\boldsymbol{\varphi} - \widetilde{\boldsymbol{\varphi}}\|_2 \\
&\quad \leq 2L_0 L_1 B^{-(1-p)^2} L_{\mathcal{R}} \|\boldsymbol{\varphi} - \widetilde{\boldsymbol{\varphi}}\|_2 + \lambda L_2 \|\boldsymbol{\varphi} - \widetilde{\boldsymbol{\varphi}}\|_2 \\
&\quad = (2L_0 L_1 B^{-(1-p)^2} L_{\mathcal{R}} + \lambda L_2) \|\boldsymbol{\varphi} - \widetilde{\boldsymbol{\varphi}}\|_2,
\end{aligned}
\tag{43}
$$

which implies that $(2L_0 L_1 B^{-(1-p)^2} L_{\mathcal{R}} + \lambda L_2)$ is a valid Lipschitz constant of our learning objective $\mathcal{F}$.

Author Contributions (All authors contributed to the algorithm design and analysis. The first draft of the manuscript was written by Shuo Chen, and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.)

Funding (S.C., G.N., and M.S. were supported by JST AIP Acceleration Research Grant Number JPMJCR20U3, Japan. M.S. was also supported by the Institute for AI and Beyond, UTokyo. C.G., J.L., and J.Y. were supported by NSF of China (Nos: U1713208, 61973162, 62072242), NSF of Jiangsu Province (No: BZ2021013), NSF for Distinguished Young Scholar of Jiangsu Province (No: BK20220080), and the Fundamental Research Funds for the Central Universities (Nos: 30920032202, 30921013114).)

Data availability (The data used in this work is all public.)

Code availability The codes of the proposed method will be released after publishing.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

Alpaydin, E. (2020). *Introduction to machine learning*. MIT Press.

Asuncion, A., & Newman, D. (2007). *Uci machine learning repository*.

Bar-Hillel, A., Hertz, T., Shental, N., & Weinshall, D. (2003). Learning distance functions using equivalence relations. In *ICML* (pp. 11–180).

Berrendero, J. R., Bueno-Larraz, B., & Cuevas, A. (2020). On mahalanobis distance in functional settings. *Journal of Machine Learning Research, 21*(9), 1–33.

Bian, W., & Tao, D. (2012). Constrained empirical risk minimization framework for distance metric learning. *IEEE Transactions on Neural Networks and Learning System, 23*(8), 1194–1205.

Biswas, A., & Parikh, D. (2013). Simultaneous active learning of classifiers & attributes via relative feedback. In *CVPR* (pp. 644–651).

Brown, M., Hua, G., & Winder, S. (2010). Discriminative learning of local image descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 33*(1), 43–57.

Carlile, B., Delamarter, G., Kinney, P., Marti, A., & Whitney, B. (2017). *Improving deep learning by inverse square root linear units (isrlus)*. arXiv:1710.09967

Chen, S., Gong, C., Yang, J., Tai, Y., Hui, L., & Li, J. (2019a). Data-adaptive metric learning with scale alignment. In *AAAI* (pp. 3347–3354).

Chen, S., Luo, L., Yang, J., Gong, C., Li, J., & Huang, H. (2019b). Curvilinear distance metric learning. In *NeurIPS* (pp. 4223–4232).

Chu, X., Lin, Y., Wang, Y., Wang, X., Yu, H., Gao, X., & Tong, Q. (2020). Distance metric learning with joint representation diversification. In *ICML* (pp. 1962–1973).

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems, 2*(4), 303–314.

Davis, J. V., Kulis, B., Jain, P., Sra, S., & Dhillon, I.S. (2007). Information-theoretic metric learning. In *ICML* (pp. 209–216).

Dong, M., Wang, Y., Yang, X., & Xue, J. H. (2019). Learning local metrics and influential regions for classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 42*, 1522.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., & Uszkoreit, J. (2020). *An image is worth 16x16 words: Transformers for image recognition at scale*. arXiv preprint arXiv:2010.11929

Ermolov, A., Mirvakhabova, L., Khrulkov, V., Sebe, N., & Oseledets, I. (2022). Hyperbolic vision transformers: Combining improvements in metric learning. In *CVPR* (pp. 7409–7419).

Fazlyab, M., Robey, A., Hassani, H., Morari, M., & Pappas, G. (2019). Efficient and accurate estimation of Lipschitz constants for deep neural networks. In *NeurIPS*.

Franklin, J. (2005). The elements of statistical learning: Data mining, inference and prediction. *The Mathematical Intelligencer, 27*(2), 83–85.

Geng, C., & Chen, S. (2018). Metric learning-guided least squares classifier learning. *IEEE Transactions on Neural Networks and Learning System, 29*(12), 6409–6414.

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *AISTATS* (pp. 249–256).

Goldberger, J., Hinton, G. E., Roweis, S. T., & Salakhutdinov, R. R. (2005). Neighbourhood components analysis. In *NeurIPS* (pp. 513–520).

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial networks. In *NeurIPS* (pp. 2672–2680).

Harandi, M., Salzmann, M., & Hartley, R. (2017). Joint dimensionality reduction and metric learning: A geometric take. In *ICML* (pp. 1404–1413).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *CVPR* (pp. 770–778).

Horev, I., Yger, F., & Sugiyama, M. (2017). Geometry-aware principal component analysis for symmetric positive definite matrices. *Machine Learning, 66*, 493–522.

Huang, Z., Wang, R., Shan, S., Van Gool, L., & Chen, X. (2018). Cross Euclidean-to-Riemannian metric learning with application to face recognition from video. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 40*(12), 2827–2840.

Huo, Z., Nie, F., & Huang, H. (2016). Robust and effective metric learning using capped trace norm: Metric learning via capped trace norm. In *SIGKDD* (pp. 1605–1614).

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML* (pp. 448–456).

Kar, P., Narasimhan, H., & Jain, P. (2014). Online and stochastic gradient methods for non-decomposable loss functions. In *NeurIPS*.

Kelley, J. L. (2017). *General topology*. Courier Dover Publications.
Kim, S., Kim, D., Cho, M., & Kwak, S. (2020). Proxy anchor loss for deep metric learning. In *CVPR* (pp. 3238–3247).
Kim, Y., & Park, W. (2021). Multi-level distance regularization for deep metric learning. In *AAAI* (pp. 1827–1835).
Krause, J., Stark, M., Deng, J., & Fei-Fei, L. (2013). 3d object representations for fine-grained categorization. In *3dRR*.
Kwon, Y., Kim, W., Sugiyama, M., & Paik, M. C. (2020). Principled analytic classifier for positive-unlabeled learning via weighted integral probability metric. *Machine Learning, 66*, 513–532.
Law, M., Liao, R., Snell, J., & Zemel, R. (2019). Lorentzian distance learning for hyperbolic representations. In *ICML* (pp. 3672–3681).
Lebanon, G. (2006). Metric learning for text documents. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 28*(4), 497–508.
Li, P., Li, Y., Xie, H., & Zhang, L. (2022). Neighborhood-adaptive structure augmented metric learning. In *AAAI*.
Li, Q., Haque, S., Anil, C., Lucas, J., Grosse, R. B., & Jacobsen, J. H. (2019). Preventing gradient attenuation in Lipschitz constrained convolutional networks. In *NeurIPS*.
Lim, D., Lanckriet, G., & McFee, B. (2013). Robust structural metric learning. In *ICML* (pp. 615–623).
Lu, J., Xu, C., Zhang, W., Duan, L. Y., & Mei, T. (2019). Sampling wisely: Deep image embedding by top-k precision optimization. In *ICCV* (pp. 7961–7970).
Luo, L., Xu, J., Deng, C., & Huang, H. (2019). Robust metric learning on grassmann manifolds with generalization guarantees. In *AAAI* (pp. 4480–4487).
Meyer, C. D. (2000). *Matrix analysis and applied linear algebra* (vol. 71). SIAM.
Montgomery, D. C., & Runger, G. C. (2010). *Applied statistics and probability for engineers*. Wiley.
Oh Song, H., Xiang, Y., Jegelka, S., & Savarese, S. (2016). Deep metric learning via lifted structured feature embedding. In *CVPR* (pp. 4004–4012).
Paassen, B., Gallicchio, C., Micheli, A., & Hammer, B. (2018). Tree edit distance learning via adaptive symbol embeddings. In *ICML*.
Perrot, M., & Habrard, A. (2015). Regressive virtual metric learning. In *NeurIPS* (pp. 1810–1818).
Qian, Q., Shang, L., Sun, B., Hu, J., Li, H., & Jin, R. (2019). Softtriple loss: Deep metric learning without triplet sampling. In *CVPR*, (pp. 6450–6458).
Ralaivola, L., Szafranski, M., & Stempfel, G. (2010). Chromatic pac-bayes bounds for non-iid data: Applications to ranking and stationary β-mixing processes. *Journal of Machine Learning Research, 11*, 1927–1956.
Reddi, S. J., Hefny, A., Sra, S., Poczos, B., & Smola, A. (2016). Stochastic variance reduction for nonconvex optimization. In *ICML* (pp. 314–323).
Rudin, W. (1964). *Principles of mathematical analysis*. McGraw-Hill.
Seidenschwarz, J. D., Elezi, I., & Leal-Taixe, L. (2021). Learning intra-batch connections for deep metric learning. In *ICML*.
Sohn, K. (2016). Improved deep metric learning with multi-class n-pair loss objective. In *NeurIPS* (pp. 1857–1865).
Steiner, A., Kolesnikov, A., Zhai, X., Wightman, R., Uszkoreit, J., & Beyer, L. (2021). *How to train your vit? Data, augmentation, and regularization in vision transformers*. arXiv preprint arXiv:2106.10270
Suarez, J. L., Garcia, S., & Herrera, F. (2018). *A tutorial on distance metric learning: Mathematical foundations, algorithms and software*. arXiv:1812.05944
Suárez, J. L., Garcia, S., & Herrera, F. (2020). pydml: A python library for distance metric learning. *Journal of Machine Learning Research, 21*(96), 1–7.
Suarez, J. L., Garcia, S., & Herrera, F. (2021). Ordinal regression with explainable distance metric learning based on ordered sequences. *Machine Learning, 66*, 2729–2762.
Ting, K. M., Zhu, Y., Carman, M., Zhu, Y., Washio, T., & Zhou, Z. H. (2019). Lowest probability mass neighbour algorithms: Relaxing the metric constraint in distance-based neighbourhood algorithms. *Machine Learning, 108*(2), 331–376.
Vershynin, R. (2018). *High-dimensional probability: An introduction with applications in data science* (vol. 47). Cambridge University Press.
Wang, H., Nie, F., & Huang, H. (2014). Robust distance metric learning via simultaneous l1-norm minimization and maximization. In *ICML* (pp. 1836–1844).
Wang, X., Han, X., Huang, W., Dong, D., & Scott, M. R. (2019). Multi-similarity loss with general pair weighting for deep metric learning. In *CVPR* (pp. 173–182).

Weinberger, K. Q., Blitzer, J., & Saul, L. K. (2006). Distance metric learning for large margin nearest neighbor classification. In *NeurIPS* (pp. 1473–1480).
Weisstein, E. W. (2002). *Inverse trigonometric functions*. https://mathworldwolfram.com/
Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., & Perona, P. (2010). *Caltech-UCSD Birds 200*. Tech. Rep. CNS-TR-2010-001, California Institute of Technology.
Xia, P., Zhang, L., & Li, F. (2015). Learning similarity with cosine similarity ensemble. *Information Sciences, 307*, 39–52.
Xie, P., Wu, W., Zhu, Y., & Xing, E. (2018). Orthogonality-promoting distance metric learning: Convex relaxation and theoretical analysise. In *ICML* (pp. 2404–2413).
Xing, E. P., Jordan, M. I., Russell, S. J., & Ng, A. (2003). Distance metric learning with application to clustering with side-information. In *NeurIPS* (pp. 521–528).
Xu, J., Luo, L., Deng, C., & Huang, H. (2018). Bilevel distance metric learning for robust image recognition. In *NeurIPS* (pp. 4198–4207).
Xu, X., Yang, Y., Deng, C., & Zheng, F. (2019). Deep asymmetric metric learning via rich relationship mining. In *CVPR* (pp. 4076–4085).
Yan, J., Yang, E., Deng, C., & Huang, H. (2022). Metricformer: A unified perspective of correlation exploring in similarity learning. In *NeurIPS*.
Yang, J., Luo, L., Qian, J., Tai, Y., Zhang, F., & Xu, Y. (2016). Nuclear norm based matrix regression with applications to face recognition with occlusion and illumination changes. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(1), 156–171.
Yang, P., Huang, K., & Liu, C. L. (2013). Geometry preserving multi-task metric learning. *Machine Learning, 66*, 133–175.
Yang, X., Zhou, P., & Wang, M. (2018). Person reidentification via structural deep metric learning. *IEEE Transactions on Neural Networks and Learning System, 30*(10), 2987–2998.
Ye, H. J., Zhan, D. C., & Jiang, Y. (2019). Fast generalization rates for distance metric learning. *Machine Learning, 66*, 267–295.
Ye, H. J., Zhan, D. C., Jiang, Y., Si, X. M., & Zhou, Z. H. (2019). What makes objects similar: A unified multi-metric learning approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 41*(5), 1257–1270.
Yoshida, T., Takeuchi, I., & Karasuyama, M. (2021). Distance metric learning for graph structured data. *Machine Learning, 66*, 1765–1811.
Yu, B., & Tao, D. (2019). Deep metric learning with tuplet margin loss. In *ICCV* (pp. 6490–6499).
Zadeh, P., Hosseini, R., & Sra, S. (2016). Geometric mean metric learning. In *ICML* (pp. 2464–2471).
Zagoruyko, S., & Komodakis, N. (2015). Learning to compare image patches via convolutional neural networks. In *ICCV* (pp. 4353–4361).
Zbontar, J., & LeCun, Y. (2016). Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research, 17*(1), 2287–2318.
Zhang, B., Zheng, W., Zhou, J., & Lu, J. (2022). Attributable visual similarity learning. In *CVPR*.
Zhang, S., Tay, Y., Yao, L., Sun, A., & An, J. (2019a). Next item recommendation with self-attentive metric learning. In *AAAI*.
Zhang, Y., Zhong, Q., Ma, L., Xie, D., & Pu, S (2019b). Learning incremental triplet margin for person re-identification. In *AAAI* (pp. 9243–9250).
Zhu, P., Cheng, H., Hu, Q., Wang, Q., & Zhang, C. (2018). Towards generalized and efficient metric learning on riemannian manifold. In *IJCAI* (pp. 192–199).

Authors and Affiliations

Shuo Chen[1] · Chen Gong[2] · Xiang Li[4] · Jian Yang[2] · Gang Niu[1] · Masashi Sugiyama[1,3]

✉ Shuo Chen
shuo.chen.ya@riken.jp

Chen Gong
chen.gong@njust.edu.cn

Xiang Li
xiang.li.implus@nankai.edu.cn

Jian Yang
csjyang@njust.edu.cn

Gang Niu
gang.niu.ml@gmail.com

Masashi Sugiyama
sugi@k.u-tokyo.ac.jp

1 Center for Advanced Intelligence Project, RIKEN, Tokyo, Japan

2 School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China

3 Graduate School of Frontier Sciences, The University of Tokyo, Tokyo, Japan

4 College of Computer Science, Nankai University, Tianjin, China