

Robust Metric Learning based on the Rescaled Hinge Loss

Sumia Abdulhussien Razooqi Al-Obaidi¹, Davood Zabihzadeh*², Hamideh Hajiabadi³

¹ Supervision and Scientific Evaluation Apparatus, Ministry of Higher Education and Scientific Research, Baghdad, Iraq

²Computer Department, Engineering Faculty, Sabzevar University of New Technology, Sabzevar, IRAN

³ Department of Computer Engineering, Birjand University of Technology, Birjand, Iran

* Corresponding Author

sumia.abedai@mohesr.gov.iq, d.zabihzadeh@mail.um.ac.ir, hajiabadi@birjandut.ac.ir

Abstract

Distance/Similarity learning is a fundamental problem in machine learning. For example, kNN classifier or clustering methods are based on a distance/similarity measure. Metric learning algorithms enhance the efficiency of these methods by learning an optimal distance function from data. Most metric learning methods need training information in the form of pair or triplet sets. Nowadays, this training information often is obtained from the Internet via crowdsourcing methods. Therefore, this information may contain label noise or outliers leading to the poor performance of the learned metric. It is even possible that the learned metric functions perform worse than the general metrics such as Euclidean distance. To address this challenge, this paper presents a new robust metric learning method based on the Rescaled Hinge loss. This loss function is a general case of the popular Hinge loss and initially introduced in (Xu et al. 2017) to develop a new robust SVM algorithm. In this paper, we formulate the metric learning problem using the Rescaled Hinge loss function and then develop an efficient algorithm based on HQ (Half-Quadratic) to solve the problem. Experimental results on a variety of both real and synthetic datasets confirm that our new robust algorithm considerably outperforms state-of-the-art metric learning methods in the presence of label noise and outliers.

Keywords: Metric Learning, Rescaled Hinge loss, Robust Algorithm, Label noise, Outlier, Half Quadratic (HQ) optimization

1. Introduction

Similarity/Distance measures are a key component in many machine learning and data mining algorithms. For example, clustering methods or kNN classifier are based on a similarity/distance measure. In addition, information retrieval systems require a measure to sort the retrieved objects based on degrees of relevancy to a query object. However, standard measures such as Euclidean distance or cosine similarity are not appropriate for many applications. For example in Figure 1, the w_1 feature, unlike w_2 , is useful to discriminate data for the classification task while standard measures assign the same weight to both of these features.

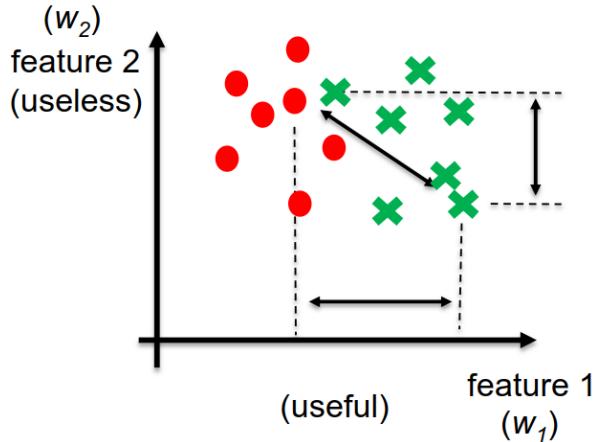


Figure 1: The problem of Standard Similarity/Distance measures. These measures assign the same importance to both w_1 and w_2 features while w_1 is more useful than w_2 to discriminate date items for the classification task. The image downloaded from Google.

Metric learning learns a distance function from data which brings conceptually related data items together while keeps unrelated ones at a distance. Distance Metric Learning (DML) algorithms often require training information in the form of pair or triplet side information as follows:

$$S = \{(x_i, x_j) \mid x_i \text{ and } x_j \text{ are similar}\}$$

$$D = \{(x_i, x_j) \mid x_i \text{ and } x_j \text{ are dissimilar}\}$$

$$T = \{(x_i, x_i^+, x_i^-) \mid x_i \text{ should be more closer to } x_i^+ \text{ than to } x_i^-\}$$

DML successfully applied in many applications such as Content Based Information Retrieval (CBIR) (Chechik et al. 2010; Hao et al. 2014; Li et al. 2017; Wu et al. 2016), Person re-identification (Bak and Carr 2017; Lin et al. 2017), Visual Tracking (Jiang et al. 2012), and Image Annotation (Guillaumin et al. 2009).

Despite achievements of DML in various applications, DML algorithms often are sensitive to outliers which may lead to substantial performance degradation. To address this challenge, Robust DML techniques are developed. These techniques can be classified into three categories

- 1 - Regularization-based,
- 2 - Probabilistic,
- 3 - Methods with robust loss function

A regularization-based approach often helps to avoid over-fitting in a small training dataset or when it contains noisy features. However, this approach is less effective in the presence of label noise.

Probabilistic methods estimate the probability of label noise for the input data. These methods often have non-convex formulation and are computationally expensive.

Another type of methods utilizes robust loss functions against outliers which has not been receiving much attention so far. Many DML algorithms (Chechik et al. 2010; Hao et al. 2014; Shi et al. 2014; Wang et al. 2015; Weinberger and Saul 2009; Wu et al. 2016; Yang et al. 2010) use the margin-based Hinge loss. These algorithms mainly use triplet side information to learn the metric. Since triplet side

information has more emphasis on relative distance instead of the absolute one, these algorithms usually have higher performance compared to those which utilize the pairwise constraints. However, the unboundedness of the Hinge function causes outliers to have a large loss which leads to the poor performance of the DML algorithm in noisy environments.

To address this issue, this work proposes a robust metric learning method based on the Rescaled Hinge loss. We initially formulate the metric learning problem using the Rescaled Hinge loss and then provide an efficient algorithm based on HQ (Half-Quadratic) {Geman, 1995 #127} to solve the problem. The proposed approach is rather general and easily can be applied to any DML algorithm based on the Hinge loss.

The rest of the paper is organized as follows: Section 2 reviews related works. Section 3 presents the formulation of metric learning problem using the Rescaled Hinge loss as well as the development of the proposed algorithm. A comparison with state-of-the-art methods and experimental results are presented in Section 4. Finally, Section 5 concludes with remarks and recommendations for future work.

2. Related Work

Nowadays it is common to use crowdsourcing or similar techniques to harvest data from the Internet, particularly in the field of computer vision and machine learning (Krishna et al. 2016). Hence, one emerging challenge in DML is noisy side information. Robust DML algorithms are developed to address this problem.

Many metric learning algorithms add a regularizer to the objective function. The optimization problem of DML typically has the following general form:

$$\min_{\mathbf{M}} l(\mathbf{M}, \mathcal{C}) + \lambda r(\mathbf{M}) \quad (1)$$

where \mathbf{M} is the metric, \mathcal{C} is a set of input side information and $r(\mathbf{M})$ is a regularizer on the parameters of metric. Seminal regularization term include *Frobenius norm* (Chechik et al. 2010; Hao et al. 2014; Huang et al. 2012; Jin et al. 2009; Nguyen et al. 2017; Wu et al. 2016) *trace* (Niu et al. 2014; Shen et al. 2012) and *logdet* (Davis et al. 2007; Jain et al. 2012). Although a regularizer helps to avoid over-fitting in a small or contaminated with noisy features dataset, this technique is less effective in the presence of label noise.

Most research in Robust DML is dedicated to probabilistic methods. These methods usually have non-convex formulation and are computationally expensive. In the following, we discuss some prominent methods in this domain.

GMENs¹(Yang et al. 2010) proposes a framework for learning from noisy side information based on the generalized maximum entropy model. GMENs initially formulates the DML as a binary classification task. Then, it extends the problem to the case of noisy side information. The authors provide theoretical analysis which indicates that under a certain assumption, the solution found by GMENS in noisy environment converges to the one obtained from perfect side information.

¹ Generalized Maximum Entropy Model for learning from Noisy side information

RML (Huang et al. 2012) initially formulates the DML task as a combinatorial integer optimization problem. It assumes a priori knowledge that at most $1 - \eta$ percent of triplet constraints is noisy, exists. learning task is then formulates as:

$$\begin{aligned} & \min_{t, \mathbf{M} \geq 0} t + \frac{\lambda}{2} \|\mathbf{M}\|_F^2 \\ s.t. \quad & t \geq \sum_{i=1}^N q_i l\left(d_{\mathbf{M}}^2(\mathbf{x}_i, \mathbf{x}_i^+) - d_{\mathbf{M}}^2(\mathbf{x}_i, \mathbf{x}_i^-)\right), \quad \forall \mathbf{q} \in Q(\eta) \end{aligned} \quad (2)$$

where $\{(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-) \mid i = 1, 2, \dots, N\}$ is the set of triplet constraints, $\|\mathbf{M}\|_F$ is the Frobenius norm regularizer and the set $Q(\eta)$ is defined as: $Q(\eta) = \{\mathbf{q} \in [0, 1]^N : \sum_{i=1}^N q_i \leq N\eta\}$.

The above problem is transformed to a semi-infinite programming problem [4] and further to a convex optimization problem. the final problem is solved by Nesterov's smooth optimization method [7] which converges considerably faster than the sub-gradient method.

RNCA extends a previous non-parametric DML algorithm, i.e., NCA. It analyzes the effect of label noise on the derivative of likelihood with respect to the metric. Afterwards, RNCA proposes to model the probability of the true label of each point so as to reduce that effect. The model is then optimized within the EM framework.

Recently, BLMNN¹ (Wang and Tan 2018) proposes metric learning using Bayesian inference.

Instead of the point estimation of the distance matrix, BLMNN estimates the posterior distribution of the metric using SVI (Stochastic Variational Inference) which decreases over-fitting problem in the small or noisy training set. This method extends the popular LMNN algorithm for the Bayesian learning scheme. Let S indicate the training triplet set, $p(\mathbf{M})$ is the *prior* distribution on the metric \mathbf{M} , and $p(S|\mathbf{M})$ denote the *likelihood* defined as follows:

$$\begin{aligned} p(S|\mathbf{M}) &= \prod_{(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-) \in S} p(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^- | \mathbf{M}) \\ &= C \prod_{(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-) \in S} \exp\{-2 \max(1 + d_{\mathbf{M}}^2(\mathbf{x}_i, \mathbf{x}_i^+) - d_{\mathbf{M}}^2(\mathbf{x}_i, \mathbf{x}_i^-), 0)\} \end{aligned} \quad (3)$$

then BLMN provide an algorithm to estimate the *posterior* $p(\mathbf{M}|S)$. Both the theoretical analysis and experimental results show that any outlier has a bounded influence on the learned model.

Some methods utilizes robust loss functions against outliers. For instance, (Wang et al. 2014) use the l_1 norm to deal with noisy features and examples. Since the introduced objective function is very sensitive to the initial value of the solution, this method uses (Xiang et al. 2008) to initialize the metric. Also, the unbounded l_1 norm function has linear growth versus the error caused by an outlier. Hence, this method is not robust against label noise.

The proposed algorithm is based on HQ minimization method that is a fast alternating direction method. Suppose $F(\mathbf{M})$ is a non-quadratic function of \mathbf{M} . The main idea of HQ is to introduce an

¹ Bayesian Large Margin Nearest Neighbor

auxiliary variable \boldsymbol{v} using conjugate theory {Boyd, 2004 #48} and construct a new cost function $F(\mathbf{M}, \boldsymbol{v})$ which is quadratic in \mathbf{M} and such that:

$$(\mathbf{M}^*, \boldsymbol{v}^*) = \arg \min_{\mathbf{M}, \boldsymbol{v}} F(\mathbf{M}, \boldsymbol{v}) \rightarrow \mathbf{M}^* = \arg \min_{\mathbf{M}} F(\mathbf{M})$$

3. Proposed Method

As mentioned, many DML algorithms use the Hinge loss function (l_{hinge}) shown in Figure 2. These methods typically solve an optimization problem with the following general form:

$$\underset{\mathbf{M}}{\text{minimize}} \text{reg}(\mathbf{M}) + C \sum_{i=1}^{|\mathcal{T}|} l_{hinge}(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-) \quad (4)$$

where \mathbf{M} indicate the metric, $\text{reg}(\mathbf{M})$ is a regularizer function and \mathcal{T} is the input triplet side information. The Hinge loss on the training triplet $(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-)$ is defined as:

$$l_{hinge}(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-) = \max(0, 1 + d_{\mathbf{M}}^2(\mathbf{x}_i, \mathbf{x}_i^+) - d_{\mathbf{M}}^2(\mathbf{x}_i, \mathbf{x}_i^-)) \quad (5)$$

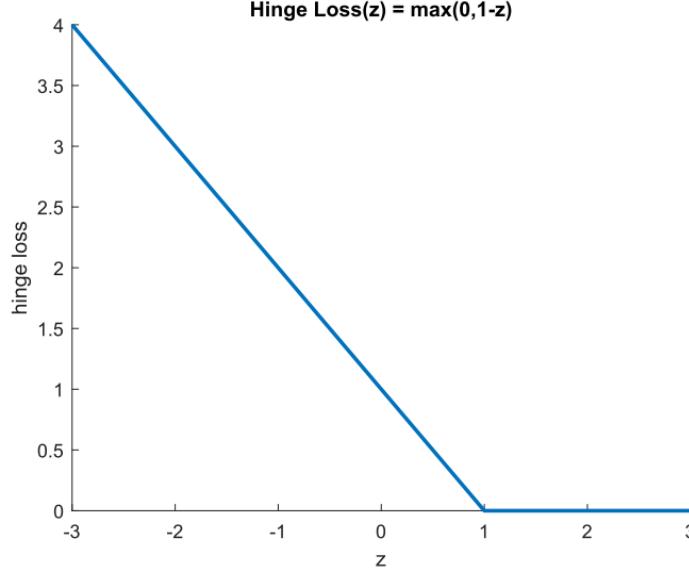


Figure 2: The margin-based Hinge loss function.

Let $z_i = d_{\mathbf{M}}^2(\mathbf{x}_i, \mathbf{x}_i^-) - d_{\mathbf{M}}^2(\mathbf{x}_i, \mathbf{x}_i^+)$, then the optimization problem (4) can be rewritten as:

$$\underset{\mathbf{M} \succcurlyeq 0}{\text{minimize}} \text{reg}(\mathbf{M}) + C \sum_{i=1}^{|\mathcal{T}|} l_{hinge}(z_i) \quad (6)$$

where $l_{hinge}(z) = \max(0, 1 - z)$

As shown in Figure 2, l_{hinge} limitlessly grows versus the variable z . The unboundedness of the l_{hinge} function causes outliers to have a large loss which results in poor performance of the learned metric.

To address this challenge, this paper presents a novel robust DML algorithm with the Rescaled Hinge loss (l_{rhinge}). This loss function is initially introduced in (Xu et al. 2017) to develop a new robust SVM algorithm against outliers. l_{rhinge} is defined as follows:

$$l_{rhinge}(z) = \beta \left[1 - \exp(-\eta l_{hinge}(z)) \right] \quad (7)$$

where η is a rescaling factor and $\beta = 1 / (1 - \exp(-\eta))$ is a normalizing constant which ensures that $l_{rhinge}(0) = 1$ (similar to the Hinge loss).

Figure 3 shows the diagram of the l_{rhinge} loss with different values for η . As seen, the large losses incurred by outliers can be controlled by adjusting the parameter η . Also, (Xu et al. 2017) proves that the Hinge loss is a special case of l_{rhinge} . More precisely, $\lim_{\eta \rightarrow 0} l_{rhinge}(z) = l_{hinge}(z)$.

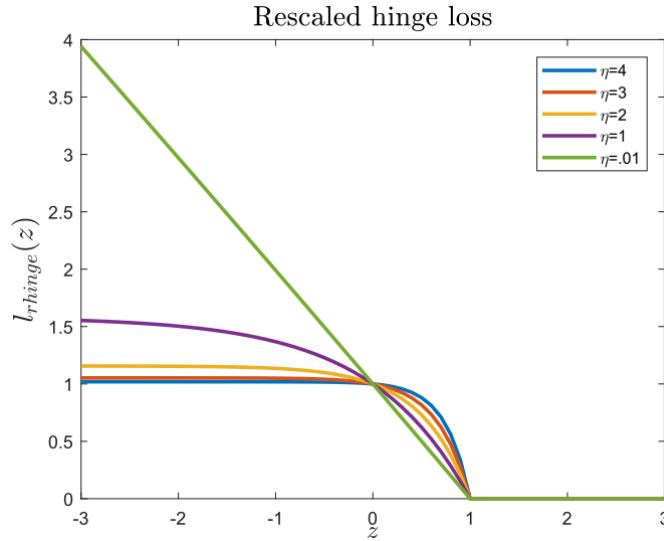


Figure 3: The Rescaled hinge loss function with different η values

The objective function of the proposed robust DML method is formulated as follows:

$$\underset{\mathbf{M} \succcurlyeq 0}{\text{minimize}} \ reg(\mathbf{M}) + C \sum_{i=1}^{|\mathcal{T}|} l_{rhinge}(z_i) \quad (8)$$

where

$$z_i = d_{\mathbf{M}}^2(\mathbf{x}_i, \mathbf{x}_i^-) - d_{\mathbf{M}}^2(\mathbf{x}_i, \mathbf{x}_i^+)$$

In the next subsection, we derive an efficient algorithm based on HQ to solve the above optimization problem.

3.1 RDML¹

Let consider the regularizer term as $\frac{1}{2} \|\mathbf{M}\|_F^2$, then we have

$$\underset{\mathbf{M} \succcurlyeq 0}{\text{minimize}} \ \frac{1}{2} \|\mathbf{M}\|_F^2 + C \sum_{i=1}^{|\mathcal{T}|} l_{rhinge}(z_i) \quad (9)$$

By simple arithmetic modification, (9) can be rewritten as

¹ Robust DML

$$\underset{\mathbf{M} \succcurlyeq \mathbf{0}}{\text{maximize}} -\frac{1}{2} \|\mathbf{M}\|_F^2 + C\beta \sum_{i=1}^{|\mathcal{T}|} \exp(-\eta l_{hinge}(z_i)) \quad (10)$$

As proved in (Xu et al. 2017), according to conjugate function theory

$$\exp(-\eta l_{hinge}(z)) = \sup_{v < 0} (\eta l_{hinge}(z)v - g(v)) \quad (11)$$

where $g(v) = -v \log(-v) + v$, ($v < 0$)

According to relation (11):

$$\begin{aligned} f_2(\mathbf{M}) &= -\frac{1}{2} \|\mathbf{M}\|_F^2 + C\beta \sum_{i=1}^{|\mathcal{T}|} \exp(-\eta l_{hinge}(z_i)) \\ &= -\frac{1}{2} \|\mathbf{M}\|_F^2 + C\beta \sum_{i=1}^{|\mathcal{T}|} \sup_{v_i < 0} (\eta l_{hinge}(z_i)v_i - g(v_i)) \\ &= -\frac{1}{2} \|\mathbf{M}\|_F^2 + C\beta \sup_{\mathbf{v} \prec \mathbf{0}} \left(\sum_{i=1}^{|\mathcal{T}|} (\eta l_{hinge}(z_i)v_i - g(v_i)) \right) \\ &= \sup_{\mathbf{v} \prec \mathbf{0}} \left(-\frac{1}{2} \|\mathbf{M}\|_F^2 + C\beta \sum_{i=1}^{|\mathcal{T}|} (\eta l_{hinge}(z_i)v_i - g(v_i)) \right) \end{aligned} \quad (12)$$

where $\mathbf{v} = [v_1, v_2, \dots, v_N] \prec \mathbf{0}$ ($N = |\mathcal{T}|$). The second equation holds since the functions $\eta l_{hinge}(z_i)v_i - g(v_i)$ $i = 1, 2, \dots, N$ are independent in terms of v_i and the last one establishes since $-\frac{1}{2} \|\mathbf{M}\|_F^2$ is constant respect to \mathbf{v} . Substituting (12) into (10) yields:

$$\underset{\mathbf{M} \succcurlyeq \mathbf{0}, \mathbf{v} \prec \mathbf{0}}{\text{maximize}} f_3(\mathbf{M}, \mathbf{v}) = -\frac{1}{2} \|\mathbf{M}\|_F^2 + C\beta \sum_{i=1}^{|\mathcal{T}|} (\eta l_{hinge}(z_i)v_i - g(v_i)) \quad (13)$$

We solve the above problem by *alternating optimization* method. More precisely, given a current value of \mathbf{v} , (13) is optimized over \mathbf{M} and then given \mathbf{M} , it is optimized over \mathbf{v} . We repeat this procedure until convergence.

Let the superscript s denote the s^{th} iteration, then if we optimize (13) just over \mathbf{v} , it will be equivalent to

$$\underset{\mathbf{v}^s \prec \mathbf{0}}{\text{maximize}} \sum_{i=1}^{|\mathcal{T}|} (\eta l_{hinge}(z_i)v_i - g(v_i)) \quad (14)$$

(14) has a closed-form solution as follows:

$$v_i^s = -\exp(-\eta l_{hinge}(z_i)), \quad i = 1, 2, \dots, |\mathcal{T}| \quad (15)$$

After obtaining \mathbf{v}^s , we can find \mathbf{M}^{s+1} from the following optimization problem.

$$\underset{\mathbf{M} \succcurlyeq \mathbf{0}}{\text{maximize}} -\frac{1}{2} \|\mathbf{M}\|_F^2 + C\beta \sum_{i=1}^{|\mathcal{T}|} (\eta l_{hinge}(z_i)v_i) \quad (16)$$

The above optimization problem is equivalent to

$$\underset{\mathbf{M} \succcurlyeq \mathbf{0}}{\text{minimize}} f(\mathbf{M}) = \frac{1}{2} \|\mathbf{M}\|_F^2 + \sum_{i=1}^{|\mathcal{T}|} C_i \xi_i \quad (17)$$

subject to

$$1 - d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_i^-) + d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_i^+) \leq \xi_i, \quad \xi_i \geq 0 \quad i = 1, 2, \dots, |\mathcal{T}|$$

where $C_i = C\beta\eta(-v_i)$ indicate the weight of the i^{th} triplet.

Since $f(\mathbf{M})$ is convex respect to \mathbf{M} , we can use the stochastic sub-gradient method to find the global minimum of (17). The derivative of $f(\mathbf{M})$ with respect to \mathbf{M} is equal to

$$\frac{\delta f}{\delta \mathbf{M}} = \mathbf{M} + \sum_{i \in \mathcal{I}^s} C_i (1 - \mathbf{A}^s) \quad (18)$$

$$\text{where } \mathbf{A}^s = (\mathbf{x}_i - \mathbf{x}_i^-)(\mathbf{x}_i - \mathbf{x}_i^-)^T + (\mathbf{x}_i - \mathbf{x}_i^+)(\mathbf{x}_i - \mathbf{x}_i^+)^T$$

Here, \mathcal{I}^s denote the set of active constraints in the current batch. According to (18), the gradient step to update \mathbf{M} is

$$\begin{aligned} \mathbf{M}^{(new)} &= \mathbf{M}^{(old)} - \lambda \left(\mathbf{M}^{(old)} - \sum_{i \in \mathcal{I}^s} C_i (1 - \mathbf{A}^s) \right) \\ &= (1 - \lambda) \mathbf{M}^{(old)} - \lambda \sum_{i \in \mathcal{I}^s} C_i (1 - \mathbf{A}^s) \end{aligned} \quad (19)$$

where λ is the learning rate. Algorithm 1 summarizes the steps of RDML

Algorithm1. RDML (Robust Distance Metric Learning)

Input: λ : learning rate

1. Initialize Matrix \mathbf{M} with Identity matrix
2. Initialize weight vector \mathbf{v} : $\mathbf{v} = \mathbf{1}$
3. for $s = 1, 2, \dots, MaxHQIter$
 - 3.1. update weight vector \mathbf{v} from (15)
 - 3.2. **repeat**
 - 3.2.1. update matrix \mathbf{M} from (19)
 - 3.3. **until** convergence
 - 3.3. $\mathbf{M} = psd(\mathbf{M})$

end;

Here, the function $psd(\mathbf{M})$ project \mathbf{M} into the cone of p.s.d (positive semi definite) matrices.

Note that, the proposed approach is rather general and can be easily applied to any DML algorithm with the Hinge loss. Here, we simply use the Frobenius norm as a regularizer and the stochastic gradient method to optimize the metric.

3.2 Run Time Analysis

To analyze the time complexity of our model, let T_h denote the time complexity of a DML algorithm based on the Hinge loss. If we apply our method to this algorithm, in addition of optimizing the metric, our method computes the vector \boldsymbol{v} in each iteration according to equation (15). This computation requires $O(p|\mathcal{T}|)$ where $p \ll d$ is the rank of matrix \mathbf{M} and $|\mathcal{T}|$ is the number of training triplets. Hence, the overall time complexity of our method is $O(MAXHQIter \times (T_h + p|\mathcal{T}|))$. In practice the vector \boldsymbol{v} is computed very faster compared to the optimization of metric \mathbf{M} with $O(d^2)$ parameters (d is the dimension of input data). So, $|\mathcal{T}| \ll T_h$. Moreover, in our experiments always $MAXHQIter \leq 3$. Therefore, our method can be efficiently applied to any DML algorithm based on the Hinge loss.

3.3 Convergence Analysis

Following the similar analysis in {Xu, 2017 #105}, the convergence property of the RDML algorithm can be established. According to (13):

$$f_3(\mathbf{M}, \boldsymbol{v}) \leq f_2(\mathbf{M}) = -\frac{1}{2} \|\mathbf{M}\|_F^2 + C\beta \sum_{i=1}^{|\mathcal{T}|} \exp(-\eta l_{hinge}(z_i)) \leq C\beta |\mathcal{T}|$$

The second inequality holds since $\|\mathbf{M}\|_F^2 \geq 0$ and $\exp(-\eta l_{hinge}(z_i)) \leq 1$. Hence, $f_3(\mathbf{M}, \boldsymbol{v})$ is upper bounded. In addition, according to (14) and (16) we have

$$f_3(\mathbf{M}^s, \boldsymbol{v}^s) \leq f_3(\mathbf{M}^s, \boldsymbol{v}^{s+1}) \leq f_3(\mathbf{M}^{s+1}, \boldsymbol{v}^{s+1})$$

Consequently, the sequence $\{f_3(\mathbf{M}^s, \boldsymbol{v}^s), s = 1, 2, \dots\}$ converges.

4. Experimental Results

In this section, we evaluate the performance of the proposed method on both synthetic and real datasets in presence of label noise. The results are compared with some peer DML methods.

In the first experiment, we investigate the ability of the proposed method to find outliers as well as reducing their effects on the learning process. For this purpose, a synthetic dataset with label noise is generated as shown in Figure 4(a). According to equation (17), $C_i = C\beta\eta(-v_i)$ indicates the weight of triplet $(\mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^-)$. We define the weight of an instance as the sum of weights of triplets in which it participates as the first element. In Figure 4(a), the normalized weight of each outlier is indicated. It is calculated by dividing the weight by the total weights. Note that, the normalized weights in this figure are multiplied by 1000 for a better representation. Figure 4(c) show the results after 3 iterations of RDML. As the results indicate, the weight of all instances with label noise is gradually decreased. The result confirm that RDML can effectively reduce the effects of outliers on the metric learning process.

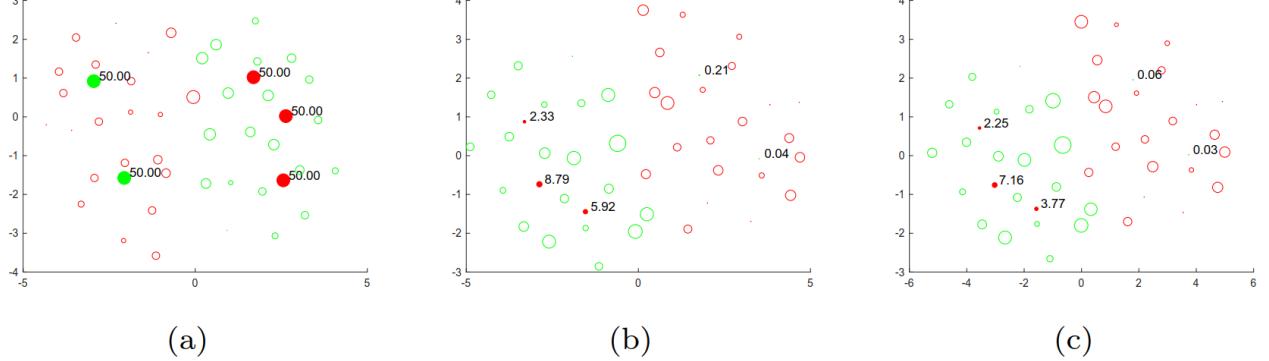


Figure 4: (a) Initial normalized weights of outliers in the synthetic dataset. (b), (c) The normalized weights after 2 and 3 iterations of the RDML algorithm

We also evaluate the performance of RDML versus outliers in a synthetic dataset with two classes sampled from two Gaussian distributions. The generation of the outliers is controlled by two parameters: *outlier_ratio* and *outlier_intensity*. More precisely, we sample outliers in the i^{th} class ($i = 1, 2$) from $\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_{oi}, \boldsymbol{\Sigma}_{oi})$ where

$$\boldsymbol{\mu}_{oi} = (1 - \text{outlier_intensity}) \times \boldsymbol{\mu}_i + \text{outlier_intensity} \times \boldsymbol{\mu}_j$$

$$\boldsymbol{\Sigma}_{oi} = 10 \times \text{outlier_intensity} \times \boldsymbol{\Sigma}_i$$

Here, $\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ is a probability distribution of the i^{th} class and $\boldsymbol{\mu}_j$ represents the mean of other class. Figure 5 shows the generated synthetic data where *outlier_ratio* = 0.2 and *outlier_intensity* is changed from 0.1 to 0.3. The outliers can be identified by filled circles in this figure.

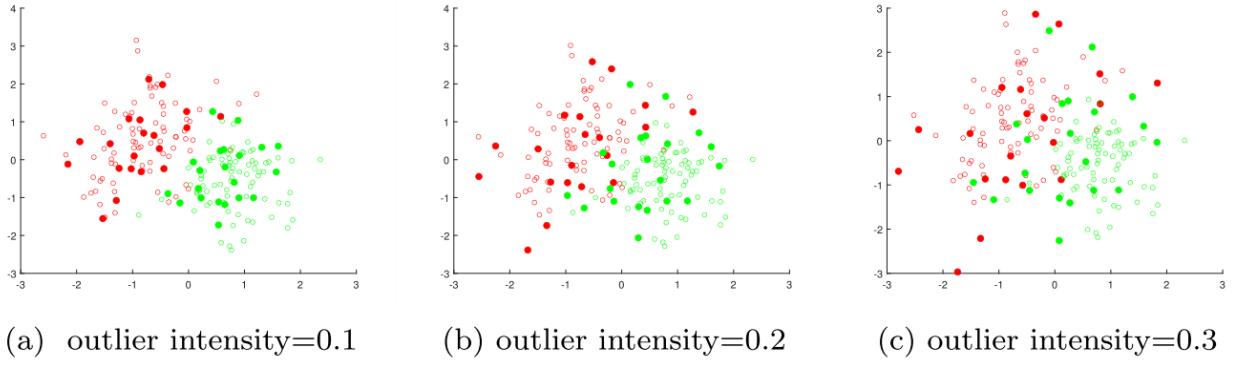


Figure 5: Generated synthetic data where *outlier_ratio* = 0.2 and *outlier_intensity* is changed from 0.1 to 0.3.

In the first experiment, we study the effect of the increasing the ratio of outliers (*outlier_ratio*) on the classification accuracy of kNN-RDML (kNN classifier with the metric learned by RDML). We changed the *outlier_ratio* from 0 to 40% and keep the *outlier_intensity* fixed equal to 0.3 . The results are plotted in Figure 12

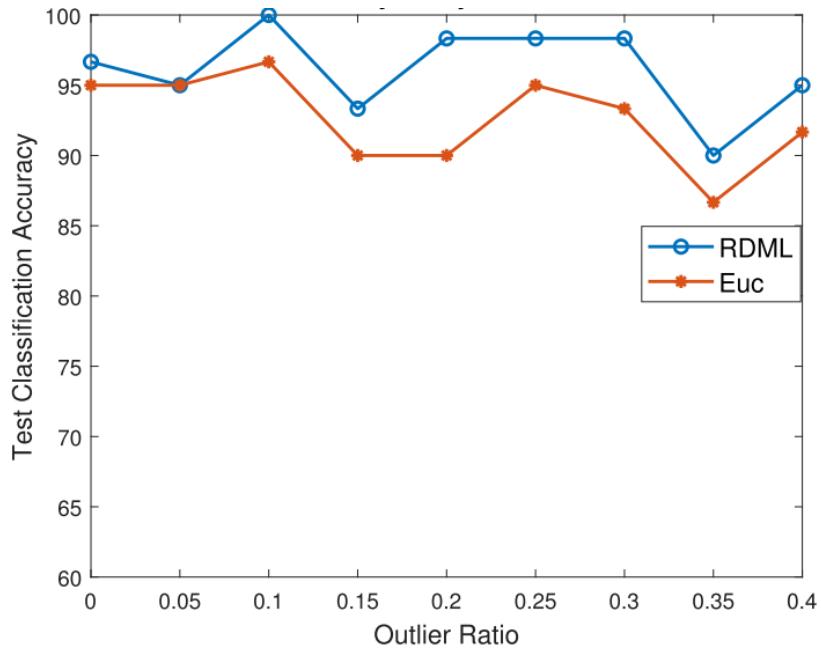


Figure 6: Classification Accuracy of kNN using RDML and Euclidean metric on the synthetic data versus outlier ratio

In the first experiment, we study the effect of the increasing (*outlier_intensity*) on the classification accuracy of kNN-RDML. We changed the *outlier_intensity* from 0 to 0.70 and keep the *outlier_ratio* fixed equal to 0.3 . The results are depicted in Figure 7.

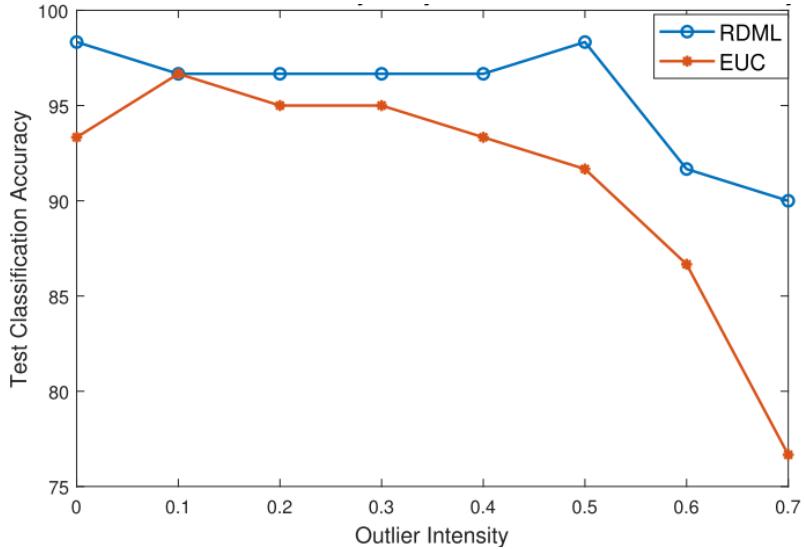


Figure 7: Classification Accuracy of kNN using RDML and Euclidean metric on the synthetic data versus outlier intensity

As the results in Figure 6 and Figure 7 indicate, the proposed method is robust against both the *outlier_ratio* and *outlier_intensity* and its performance is acceptable even when the large number of extreme outliers exists in the dataset.

In the next experiment, we evaluate the performance of the proposed method on real datasets contaminated with label noise. The results are compared with some peer DML methods. Table 1 reports the statistics of evaluated datasets in our experiments.

Table 1-Statistics and explanations of evaluated datasets

Data Set	#classes	#samples	#dim	Feature extraction	d	Description
Wine (Lichman 2013)	3	178	13	None	13	Standard UCI classification dataset. https://archive.ics.uci.edu/ml/datasets/wine
Letters (Lichman 2013)	26	20,000	16	None	16	This dataset contains 20,000 examples of 26 English capital letters. Images of letters are generated from 20 different fonts. Then, 16 numerical attributes are extracted from these images. https://archive.ics.uci.edu/ml/datasets/letter+recognition
YaleFaceB (Lee et al. 2005)	38	2,414	1,024	PCA	200	This standard face recognition dataset includes 2,414 images of 38 classes. For each person, at most 64 images are taken under extreme illumination conditions. http://vision.ucsd.edu/~iskwak/ExtYaleDatabase/ExtYaleB.html
USPS(Hull 1994)	10	9,289	256	PCA	100	USPS is a handwritten digit dataset from envelopes by the U.S. Postal Service. The originally scanned digits are binary with different sizes and orientations. The images have been normalized which results in 16×16 grayscale images. USPS consists of 7291 training observations and 2007 test instances. https://www.kaggle.com/bistaumanga/usps-dataset
MNIST(LeCun et al. 1998)	10	60,000	784	PCA	164	MNIST is a popular handwritten digits dataset. It contains 10 categories (one per digit), 60,000 training examples and 10,000 test instances. The size of each image is 28×28 pixels. http://yann.lecun.com/exdb/mnist/
Caltech101 (Li et al. 2004)	101	9,146	variable	CNN Resnet152	1000	Caltech101 is a standard machine vision classification dataset. The images belong to 101 categories. Each class contains about 40 to 800 images and the size of each image is roughly 300×200 pixels. The images have no clutter and the objects often are centered in each image. http://www.vision.caltech.edu/Image_Datasets/Caltech101/

4.1 Experimental Setup

The data in the Wine, Letters, and YaleFaceB datasets are normalized so that the mean and variance of each attribute becomes 0 and 1 respectively. Also, to decrease the noise effect, PCA is utilized to reduce the dimension of images in the YaleFaceB, USPS and MNIST datasets. The parameter d in Table 1 denotes the input dimension after applying PCA.

In the Caltech101 dataset, the feature extraction is done using the Imagenet-resnet-152-dag deep network¹. For this purpose, the images are initially rescaled to 224×224 and then subtracted from the mean image of the network. Subsequently, a total of 1000 features per image are extracted from the second last layer.

¹ downloaded from <http://www.vlfeat.org/matconvnet/pretrained/>

In the experiments, triplet side information is generated as follows. A training instance x is set as similar to k 's nearest neighbors with the same label (*target neighbors*). Also, x is considered as dissimilar to any *imposters*. These are any observations of a different class which violate the margin specified by x 's target neighbors. Finally, the triplet set is generated by the natural join of the similar and dissimilar sets. Figure 8 illustrates the concepts of *target neighbors* and *imposters* for a training example.

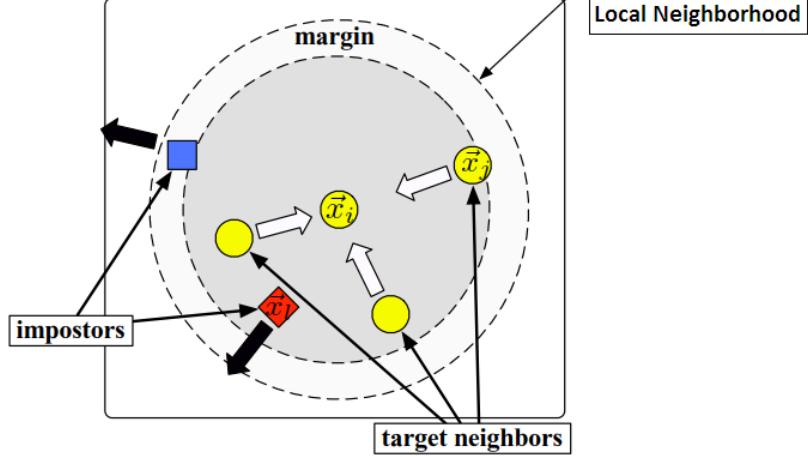


Figure 8: The illustration of the target neighbors and imposters of x_i . Reprinted from (Weinberger and Saul 2009)

We randomly split instances of datasets into training/test sets (70/30) at each run, except for USPS and MNIST which have predefined training/test sets. The results are compared with peer robust DML methods: BLMNN¹(Wang and Tan 2018), L1-DML² (Wang et al. 2014) and also with sparsity-based DML ones: SVM-trip³ (Wang et al. 2015) and SCML-G⁴ (Shi et al. 2014).

The hyperparameters of the competing methods are adjusted by 5-fold cross-validation as follows. The parameter C in RDML and SVM-trip is selected from $\{.1, .5, 1, 3, 10\}$. The learning rate (λ) in RDML is chosen from the range $(10^{-6}, 10^{-2})$. Also, we select the number of bases in SCML-G from $\{400, 1000\}$, as recommended by its authors.

The kNN classifier $k = 3$ is adopted to evaluate the performance of the learned metrics. Figure 9 depicts the system flow of the proposed learning scheme which consists of two phases namely the *learning* and *test*.

¹ http://parnec.nuaa.edu.cn/xtan/data/BLMNN_demo.zip

² http://parnec.nuaa.edu.cn/xtan/data/BLMNN_demo.zip

³ <http://www.cs.cmu.edu/~deyum/Publications.htm>

⁴ <http://mloss.org/software/view/553/>

Training Phase

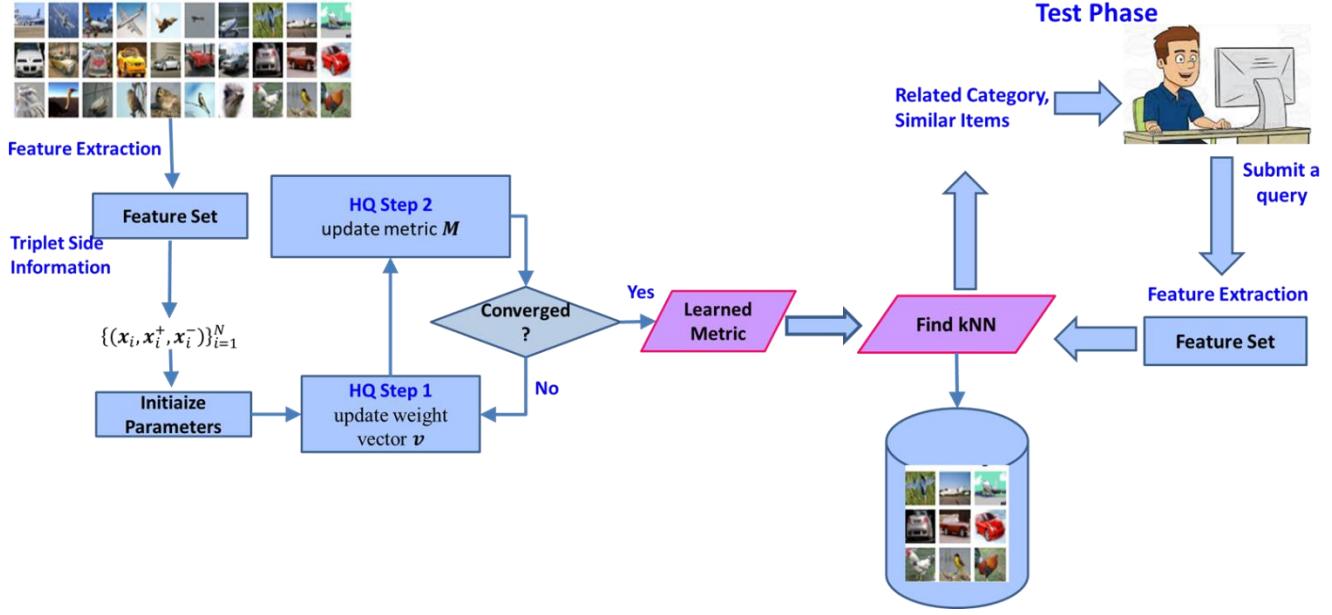


Figure 9: The system flow of the proposed metric learning scheme

4.2 Results and Analysis

The classification rate of the kNN classifier using the learned metric of the competing methods is reported in Table 2. Here, the parameter nl indicates the percent of noise level in datasets. The results are obtained by averaging over 10 runs on these datasets. Also, Figure 7 depicts the accuracy of competing methods versus label noise (ranging from 0% to 20%). To make the comparison meaningful, we perform statistical analysis with p-value = 5% on the obtained results. The results, which have statistically significant differences are marked by * in Table 2. Also, these results are shown using box-plots in Figure 11. In addition, the running time of RDML and other DML methods are compared together on evaluated datasets and the results are shown in Figure 12.

Table 2- The classification rate of the kNN classifier using the learned metric of the competing methods

Data Set	nl %	RDML	BLMNN	L1_DML	SCML_G	SVM_TRIP
Wine	0	98.11±1.33	97.55±2.00	97.36±1.03	98.49±0.84	96.60±2.67
	5	97.74±2.46	96.60±2.19	96.98±1.03	96.60±2.07	94.34±3.77
	10	96.98±1.58	96.23±3.54	95.47±1.03	93.21±3.16	93.96±4.50
	20	95.09±3.91	92.83±3.65	92.45±3.77	93.96±4.50	93.21±4.46
Letters	0	97.44±0.17*	94.10±0.39	94.12±0.26	95.88±0.12	94.28±0.21
	5	93.68±0.20	93.79±0.25	89.38±0.38	89.71±1.30	93.19±0.16
	10	92.02±0.45	91.51±0.55	87.84±0.42	88.58±0.99	91.69±0.14
	20	88.08±0.34	87.43±0.57	84.06±0.55	84.29±1.40	87.43±0.48
YaleB32	0	96.72±0.84	90.51±0.97	90.21±0.50	95.86±1.00	87.54±1.04
	5	95.72±0.59*	86.73±1.59	87.37±0.80	93.30±0.41	86.48±0.62
	10	93.94±1.00*	87.90±1.90	86.40±0.87	92.71±0.40	84.87±1.13
	20	89.21±0.82*	83.56±1.32	81.53±2.08	86.43±0.45	80.81±1.50
USPS	0	95.45±0.03*	94.42±0.00	94.91±0.06	93.54±0.31	94.97±0.00
	5	94.90±0.22*	94.13±0.20	91.66±0.33	91.08±0.42	94.44±0.10
	10	93.80±0.29	92.75±0.47	90.11±0.28	90.60±0.92	93.56±0.20
	20	90.55±0.80	89.69±0.44	86.51±0.56	86.74±0.26	89.99±0.49
MNIST	0	97.84±0.08	97.77±0.00	97.09±0.01	97.45±0.06	97.40±0.00
	5	97.06±0.14	96.70±0.06	96.32±0.10	90.84±0.99	95.20±0.14
	10	95.59±0.34	95.53±0.13	95.08±0.23	88.65±1.15	94.15±0.24
	20	91.91±0.13	91.83±0.20	91.24±0.17	85.61±0.65	90.50±0.17
Caltech101	0	90.29±0.00	88.73±0.64	82.39±0.48	90.71±0.40	89.48±0.28
	5	89.49±0.19*	86.52±0.37	73.77±0.58	83.21±0.88	84.00±0.33
	10	89.08±0.43*	85.80±0.64	71.58±0.85	81.75±0.46	82.92±0.98
	20	86.63±0.67*	81.46±0.16	66.64±1.08	74.59±0.53	79.82±0.79

As the results in Table 2 and Figure 10 indicate, RDML and BLMNN significantly outperform other DML methods and the performance of these two methods declines slowly than other ones with the increase of noise level. Also, RDML is consistently better than BLMNN in most of the evaluated datasets.

Other advantages of the proposed method over BLMNN include lower run time (refer to Figure 12) as well as its remarkable flexibility so that it can be easily applied to any DML algorithm based on the Hinge loss.

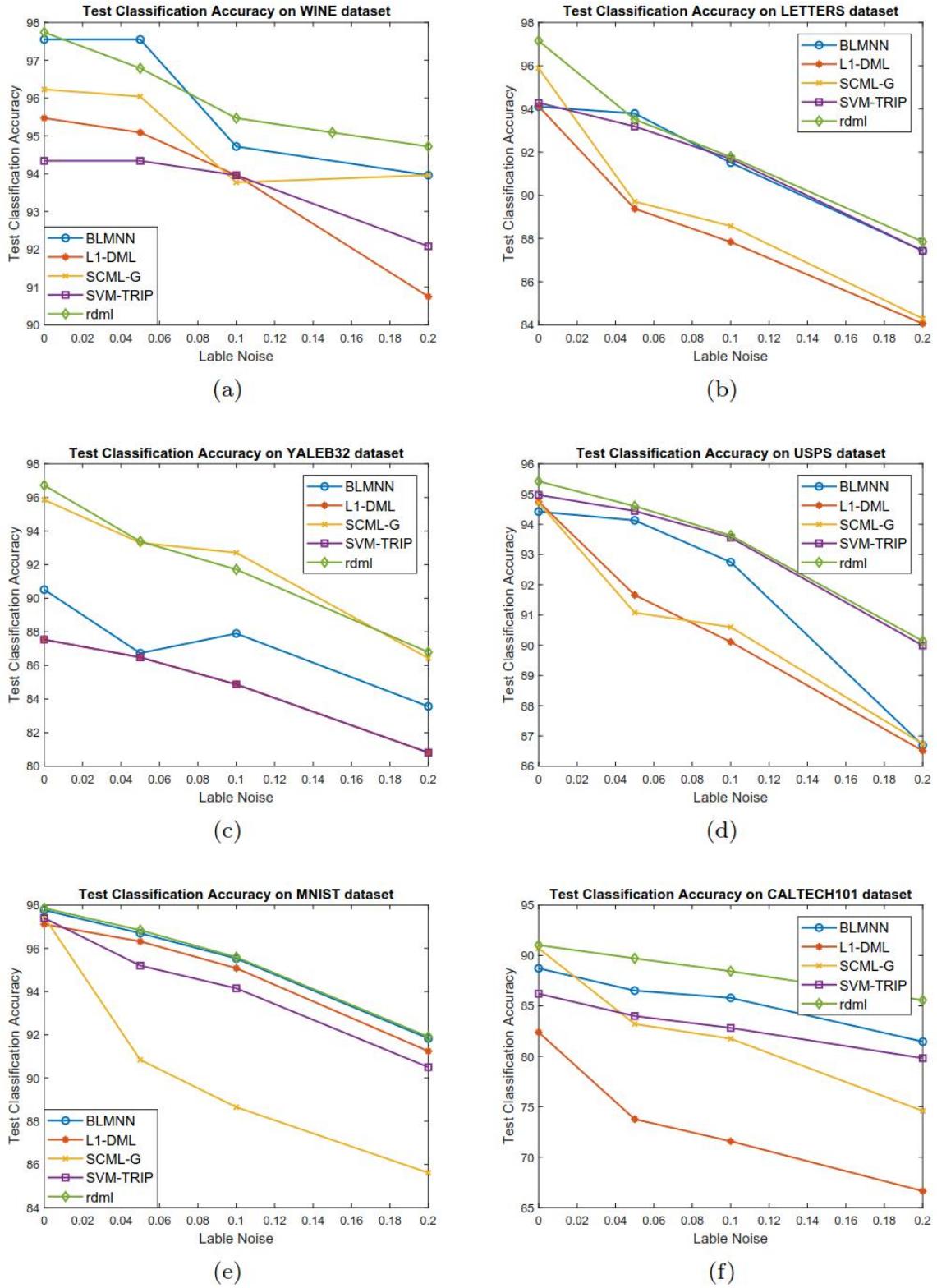


Figure 10: Comparison of the classification accuracy of RDML with other DML methods versus label noise.

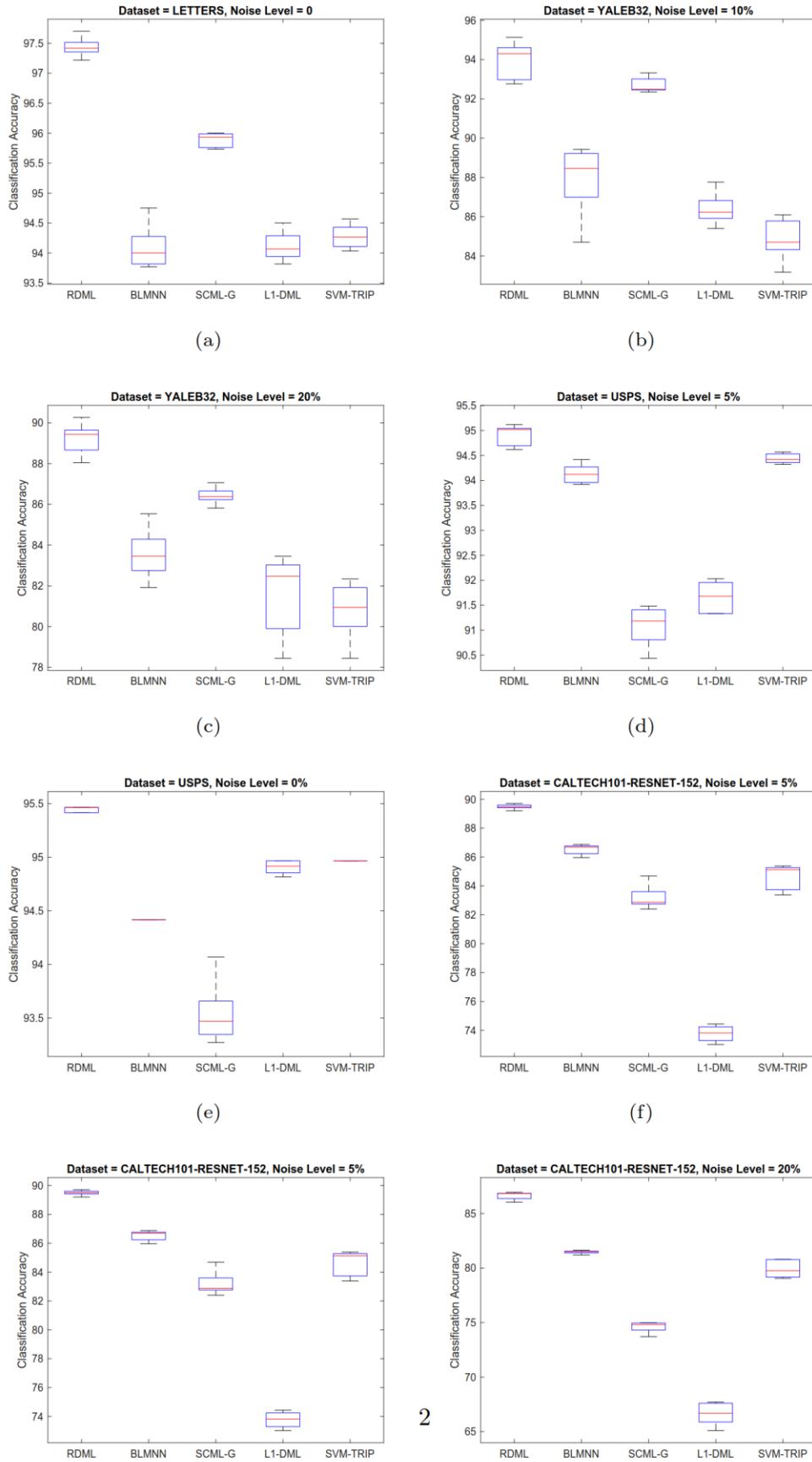


Figure 11: Box-Plots of the results, which have statistically significant differences.

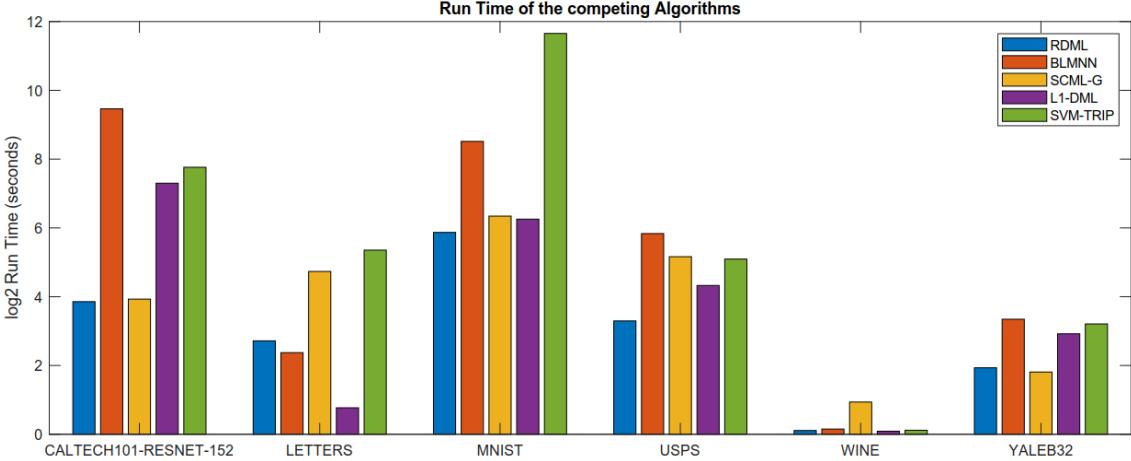


Figure 12: Comparison of the run time of RDML with other DML methods.

5. Conclusion and Future Work

Datasets with noisy information are common in today's applications. The main reason is the fact that many datasets are collected from the Internet using crowdsourcing or similar techniques. The wrong side information significantly diminishes the performance of DML algorithms.

To address this emerging challenge, the paper presents a robust DML method (named RDML) based on the Rescaled Hinge loss named RDML. Specifically, we initially formulate the DML problem with the robust loss function and then develop an efficient algorithm base on HQ to solve the problem. The proposed approach is rather general and one can easily apply it to any DML algorithm based on the Hinge loss.

Several experiments are conducted on both synthetic and real datasets to measure the performance of the proposed method. Experimental results indicate that RDML can effectively identify the noisy side information and reduce their influences in the metric learning process. Also, the results confirm that RDML significantly surpasses other peer DML methods in many of the evaluated datasets and its performance degrades slowly than other ones with the increase of noise level.

Some directions for future work in this area include:

- I. Investigating the performance of the proposed methods in other applications like *CBIR* (*Content Based Information Retrieval*).
- II. Extension of the proposed method for *semi-supervised* learning.
- III. Examining other robust loss functions in the proposed framework.

Acknowledgment

We would like to acknowledge the Machine Learning Lab. in Engineering Faculty of FUM for their kind and technical support.

References

- Bak S, Carr P One-shot metric learning for person re-identification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017. pp 2990-2999
- Chechik G, Sharma V, Shalit U, Bengio S (2010) Large Scale Online Learning of Image Similarity Through Ranking J Mach Learn Res 11:1109-1135
- Davis JV, Kulis B, Jain P, Sra S, Dhillon IS (2007) Information-theoretic metric learning. Paper presented at the Proceedings of the 24th international conference on Machine learning, Corvalis, Oregon, USA,
- Guillaumin M, Mensink T, Verbeek J, Schmid C Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In: Computer Vision, 2009 IEEE 12th International Conference on, 2009. IEEE, pp 309-316
- Hao X, Hoi SCH, Rong J, Peilin Z (2014) Online Multiple Kernel Similarity Learning for Visual Search Pattern Analysis and Machine Intelligence, IEEE Transactions on 36:536-549 doi:10.1109/TPAMI.2013.149
- Huang K, Jin R, Xu Z, Liu C-L (2012) Robust metric learning by smooth optimization arXiv preprint arXiv:12033461
- Hull JJ (1994) A database for handwritten text recognition research IEEE Transactions on Pattern Analysis and Machine Intelligence 16:550-554 doi:10.1109/34.291440
- Jain P, Kulis B, Davis JV, Dhillon IS (2012) Metric and kernel learning using a linear transformation J Mach Learn Res 13:519-547
- Jiang N, Liu W, Wu Y Order determination and sparsity-regularized metric learning adaptive visual tracking. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, 2012. IEEE, pp 1956-1963
- Jin R, Wang S, Zhou Y Regularized distance metric learning: Theory and algorithm. In: Advances in neural information processing systems, 2009. pp 862-870
- Krishna RA, Hata K, Chen S, Kravitz J, Shamma DA, Fei-Fei L, Bernstein MS Embracing error to enable rapid crowdsourcing. In: Proceedings of the 2016 CHI conference on human factors in computing systems, 2016. ACM, pp 3167-3179
- LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition Proceedings of the IEEE 86:2278-2324
- Lee K-C, Ho J, Kriegman DJ (2005) Acquiring linear subspaces for face recognition under variable lighting IEEE Transactions on pattern analysis and machine intelligence 27:684-698
- Li F-F, Fergus R, Perona P Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. In: 2004 Conference on Computer Vision and Pattern Recognition Workshop, 27-02 June 2004 2004. pp 178-178. doi:10.1109/CVPR.2004.109
- Li J, Xu C, Yang W, Sun C, Tao D (2017) Discriminative multi-view interactive image re-ranking IEEE Transactions on Image Processing 26:3113-3127
- Lichman M (2013) UCI Machine Learning Repository
- Lin L, Wang G, Zuo W, Feng X, Zhang L (2017) Cross-domain visual matching via generalized similarity measure and feature learning IEEE transactions on pattern analysis and machine intelligence 39:1089-1102
- Nguyen B, Morell C, Baets BD (2017) Supervised distance metric learning through maximization of the Jeffrey divergence Pattern Recogn 64:215-225 doi:10.1016/j.patcog.2016.11.010
- Niu G, Dai B, Yamada M, Sugiyama M (2014) Information-Theoretic Semi-Supervised Metric Learning via Entropy Regularization Neural Computation 26:1717-1762 doi:10.1162/NECO_a_00614
- Shen C, Kim J, Wang L, Hengel AVD (2012) Positive semidefinite metric learning using boosting-like algorithms J Mach Learn Res 13:1007-1036
- Shi Y, Bellet A, Sha F Sparse Compositional Metric Learning. In: AAAI, 2014. pp 2078-2084

- Wang D, Tan X (2018) Robust Distance Metric Learning via Bayesian Inference IEEE Transactions on Image Processing 27:1542-1553
- Wang F, Zuo W, Zhang L, Meng D, Zhang D (2015) A kernel classification framework for metric learning IEEE transactions on neural networks and learning systems 26:1950-1962
- Wang H, Nie F, Huang H Robust Distance Metric Learning via Simultaneous L1-Norm Minimization and Maximization. In: Jebara T, Xing EP (eds) Proceedings of the 31st International Conference on Machine Learning (ICML-14), 2014. JMLR Workshop and Conference Proceedings, [Formatter not found: ResolvePDF], pp 1836-1844
- Weinberger KQ, Saul LK (2009) Distance Metric Learning for Large Margin Nearest Neighbor Classification J Mach Learn Res 10:207-244
- Wu P, Hoi SCH, Zhao P, Miao C, Liu ZY (2016) Online Multi-Modal Distance Metric Learning with Application to Image Retrieval IEEE Transactions on Knowledge and Data Engineering 28:454-467 doi:10.1109/TKDE.2015.2477296
- Xiang S, Nie F, Zhang C (2008) Learning a Mahalanobis distance metric for data clustering and classification Pattern Recogn 41:3600-3612 doi:10.1016/j.patcog.2008.05.018
- Xu G, Cao Z, Hu B-G, Principe JC (2017) Robust support vector machines based on the rescaled hinge loss function Pattern Recognition 63:139-148
- Yang T, Jin R, Jain AK Learning from noisy side information by generalized maximum entropy model. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10), 2010. Citeseer, pp 1199-1206