INTERNATIONAL JOURNAL OF CIRCUITS, SYSTEMS AND SIGNAL PROCESSING
DOI: 10.46300/9106.2022.16.26 Volume 16, 2022

Similar Pair-free Partial Label Metric Learning

Houjie Li[1], Min Yang[1,2], Yu Zhou[1], Ruirui Zheng[1], Wenpeng Liu[1], and Jianjun He[1,*]
College of Information and Communication Engineering, Dalian Minzu University, Dalian, China
School of Electronic and Information Engineering, Tongji University, Shanghai, China
*Corresponding author, Email address: jianjunhe@live.com

Received: August 2, 2021. Revised: December 20, 2021. Accepted: January 8, 2022. Published: January 10, 2022.

Abstract- Partial label learning is a new weakly supervised learning framework. In this framework, the real category label of a training sample is usually concealed in a set of candidate labels, which will lead to lower accuracy of learning algorithms compared with traditional strong supervised cases. Recently, it has been found that metric learning technology can be used to improve the accuracy of partial label learning algorithms. However, because it is difficult to ascertain similar pairs from training samples, at present there are few metric learning algorithms for partial label learning framework. In view of this, this paper proposes a similar pair-free partial label metric learning algorithm. The main idea of the algorithm is to define two probability distributions on the training samples, i.e., the probability distribution determined by the distance of sample pairs and the probability distribution determined by the similarity of candidate label set of sample pairs, and then the metric matrix is obtained via minimizing the KL divergence of the two probability distributions. The experimental results on several real-world partial label datasets show that the proposed algorithm can improve the accuracy of k-nearest neighbor partial label learning algorithm (PL-KNN) better than the existing partial label metric learning algorithms, up to 8 percentage points.

Keywords- Partial label learning, Metric learning, Similar pair-free, Weakly supervised data.

I. Introduction

IN the traditional supervised learning framework, training samples usually need to be accurately labeled with category information, which can achieve gratifying results on small sample data. With the advent of the era of big data, data scale is growing rapidly. Sometimes the labeling of sample category information will consume huge manpower and material resources, and the label forms of samples are various, so that the data obtained with accurate category label information are limited. On the contrary, samples with incorrect labels, multiple labels, inadequate labels, and local labels (i.e. weakly supervised data) are usually readily available. In this case, more and more research on weakly supervised learning technology [1] has been conducted. Partial label learning [2] is an important weakly supervised learning framework in which a classifier can be trained by knowing only a candidate set of the real category label of the training samples, so in many practical problems [3] it has a wide range of applications.

Due to the requirement of finding the real category label from several candidate category labels of the training sample, the algorithm of the partial label learning framework is more difficult to be established than the traditional classification framework. In earlier studies, scholars mainly tried to improve the traditional machine learning model to establish the partial label learning algorithms, such as k-nearest neighbor model [4], linear support vector machine [5], maximum margin method [6, 7, 8], offset tree [9], and maximum likelihood estimation [10] are gradually being used to develop partial label learning algorithms. The common idea of the above methods to solve partial label learning problem is to disambiguate the candidate category labels of the sample in the category label space, and have achieved certain effects. In recent years, it has been observed that mining useful information in the feature space of the samples to disambiguate candidate category labels can achieve better results. The literature [11] and [12] developed two partial label learning algorithms using the manifold structure information of training samples respectively. Inspired by the above results, Zhou and Gu [13] proposed a partial label metric learning algorithm based on geometric mean model (PL-GMML) by studying the metric learning problem in the partial label learning framework. As the front end of the PL-KNN algorithm [4], the PL-GMML algorithm can effectively improve the accuracy of PL-KNN. The main idea of the PL-GMML algorithm is as follows. First of all, two samples which both have non-empty intersection of their candidate category label sets and smaller distance are taken as a similar pair, while

which have empty intersection of their candidate category label sets are taken as a dissimilar pair, and then a partial label metric learning algorithm is developed using the obtained similar pairs and dissimilar pairs based on the traditional geometric mean model. Since the two samples with both smaller distance and non-empty intersection of their candidate category label sets may also come from different categories, the construction method of similar pairs in PL-GMML algorithm can not get completely accurate similar pair, which will affect the accuracy of the algorithm to a certain extent. In addition, the geometric mean model makes the distance between samples of different classes as large as possible by minimizing the distance determined by the inverse matrix of the metric matrix. Although an analytical solution of the metric matrix can be obtained, the accuracy of the geometric mean model is usually lower than other metric learning models. In view of the above two reasons, this paper proposes a new similar pair-free partial label metric learning algorithm, which is abbreviated as PL-SPFML algorithm. The main idea of the algorithm is to define two probability distributions on the training samples, i.e., the probability distribution determined by the distance of sample pairs and the probability distribution determined by the similarity of candidate label sets of sample pairs, and then a metric matrix is obtained via minimizing the Kullback-Leibler divergence of the two probability distributions, which makes the samples in the same category collapse to the same point as far as possible, and the distance of two samples come from different category should be as infinite as possible. The simulation results on the UCI dataset and the real-world partial label dataset show that the PL-SPFML algorithm can better improve the classification accuracy of the PL-KNN algorithm compared with the PL-GMML algorithm.

The structure of the paper is as follows. Firstly, the related work required in this paper is introduced. Secondly, the algorithm is described, and the flow chart is given. Then the simulation experiment is carried out to verify the performance of the algorithm and analyze the experimental results. The final section draws conclusions.

II. Related work

In the partial label learning framework, each training sample has multiple candidate category labels at the same time, but only one of them is the true category label. The mathematical description of the framework is as follows: Let $X = R^d, Y = \{1, 2, ..., Q\}$ be the feature space and the category label space respectively, $S = \{(x_i, Y_i)|i = 1, 2, \cdots, n\}$ be the given training sample set, where $x_i \in X$ represents the feature vector of the i-th training sample, $Y_i \subset Y$ is the candidate category label set of x_i. The task of partial label learning is to learn a multi-class classifier $f : X \to Y$ based on the training set S. In the early researches, people mainly tried to improve the traditional machine learning model to propose the partial label learning algorithm. The literature [4] proposed a partial label learning algorithm based on k-nearest neighbor model; literature [5] proposed a partial label learning algorithm based on the linear support vector machine; literatures [6, 7, 8] proposed partial label learning algorithms based on the maximum margin model; literature [9] proposed a learning algorithm based on offset tree; literature [14] proposed a learning algorithm based on dictionary learning; Gong et al. [15] and Feng and An [16] successively proposed partial label learning algorithms based on regularization method; literature [17] proposed a partial label learning algorithm based on ECOC technology; Liu and Dietterich [18] proposed a conditional multinomial mixture model based partial label learning algorithm; Feng and An [19] proposed a partial label learning algorithm based on leveraging latent label distributions according to different labeling confidence levels of different labels; literature [20] proposed a partial label learning algorithm for structured output data with candidate labels; Tang and Zhang [21] used Boosting techniques to optimize the confidence-rated of candidate label in samples to establish a partial label learning algorithm; literature [22] proposed a partial label learning algorithm based on binary classifier; Wang and Zhang [23] proposed a partial label learning algorithm for class-imbalance data; literatures [24] and [25] proposed two partial label learning algorithms based on Graph model; Zhou et al. [26] proposed a partial label learning algorithm based on Gaussian process model; Xu et al. [27] proposed a partial label learning algorithm based on label enhancement strategy; Lyu et al. [28] proposed a self-paced regularization framework for partial-label learning; Yao et al. [29] proposed a partial label learning algorithm by using CNN model.

Because training data can be used to learn a better distance metric to improve the accuracy of related learning algorithms, metric learning technology has attracted wide attention from machine learning scholars in recent years. Many excellent metric learning algorithms have been proposed for the traditional classification problem. Literature [30] proposed a metric learning model based on information geometry; Weinberger and Saul [31] and Verma and Jawahar [32] proposed a metric learning algorithm based on maximum margin method; literature [33] proposed a metric learning model based on geometric mean; Globerson and Roweis [34] proposed a metric learning algorithm based on collapsing classes; Huai et al. [35] proposed a metric learning from probabilistic labels.

Since the true category label of the training sample in the partial label learning problem is uncertain, it is difficult to accurately determine whether a pair of samples belongs to the same class, the traditional metric learning technology cannot be directly applied to the partial label learning problem. At present, the research of metric learning algorithm for partial label learning problem is rare. The authors only saw two partial label metric learning algorithms proposed by using the geometric mean model [13] and collapsing classes model [34], re-

spectively. However, these two algorithms were proposed by using the same construction method of similar pairs, which can not get completely accurate similar pairs and thus affects the performance of the algorithms.

III. The PL-SPFML Algorithm

A. Modeling

The same as the metric learning algorithm in traditional supervised learning framework, the purpose of this paper is to learn a metric matrix A that the distance

$$d(x, x'|A) = \sqrt{(x - x')^{\mathrm{T}} A(x - x')} \tag{1}$$

determined by it will meet our requirements, where $x, x' \in R^d$ represents the feature vectors of two samples, and A is a symmetric positive definite matrix. In order to avoid the computational difficulty caused by the square root, we will use $d^2(x, x'|A)$ instead of $d(x, x'|A)$ in the process of modeling.

$$d^2(x, x'|A) = (x - x')^{\mathrm{T}} A(x - x') \tag{2}$$

In the partial label learning problem, the true category label of the training sample is unknown but concealed in a candidate category label set, so it is difficult to accurately distinguish whether two samples belong to the same category only according to the candidate category label set. In literature [13], two samples which both have non-empty intersection of their candidate category label sets and smaller distance are taken as a similar pair, but the similar sample pairs constructed in this way are not accurate. Considering that in the partial label learning framework, it is difficult to accurately obtain similar pairs of samples, so this paper will construct a metric learning model avoiding the steps of constructing similar pairs and dissimilar pairs. The main idea is to define two probability distributions on the training samples, i.e., the probability distribution determined by the distance of sample pairs and the probability distribution determined by the similarity of candidate label set of sample pairs, and then a metric matrix is obtained via minimizing the KL divergence of the two probability distributions, as follows.

Inspired by the idea of neighborhood component analysis model [36], for each training sample x_i, in the metric space determined by the metric matrix A , the probability that the samples x_j and x_i are of the same category (represented by $p^A(x_j|x_i)$) can be defined as

$$p^A(x_j|x_i) = \frac{1}{Z_i} e^{-d_{ij}^A} = \frac{e^{-d_{ij}^A}}{\sum\limits_{m \neq i} e^{-d_{im}^A}} \tag{3}$$

where $d_{ij}^A = d^2(x_i, x_j|A) = (x_i - x_j)^{\mathrm{T}} A(x_i - x_j)$. It can be seen that the closer the distance between x_j and x_i , the larger the probability that they are of the same category, and vice versa. In addition, by using the candidate category label set of the sample, we define the probability that the sample x_j and x_i are the same category as follows

$$p_0^A(x_j|x_i) \propto \frac{|Y_i \cap Y_j|}{|Y_i| \cdot |Y_j|} \tag{4}$$

It can be seen that if the intersection of the candidate category label set of the two samples is an empty set, i.e. the two samples do not belong to the same category, then $p_0^A(x_j|x_i)$=0; if the candidate category label set of both samples has only one element and the intersection is non-empty (that indicate the two samples come from the same category), then $p_0^A(x_j|x_i)$= 1; other cases $p_0^A(x_j|x_i)$ will be between 0 and 1. The purpose of this paper is to find a metric matrix A such that the samples of the same categories collapse into the same point as much as possible, and the distance of two samples come from different categories is as infinite as possible. To achieve this purpose, we can look for a metric matrix A such that $p^A(x_j|x_i)$ is as close as possible to $p_0^A(x_j|x_i)$, so the following objective function can be used

$$\min_A \sum_i KL[p_0^A(x_j|x_i)|p^A(x_j|x_i)] \tag{5}$$

where $KL[\cdot|\cdot]$ denotes the KL divergence between two probability distributions. To prevent overfitting, we add a regularization term $\lambda tr((A-I)^T(A-I))$ to the objective function, where λ is the regularization parameter, I is an identity matrix. Therefore, the following new objective function is established.

$$\begin{aligned} \min_A \quad & \sum_i KL\left[p_0^A(x_j|x_i)|p^A(x_j|x_i)\right] \\ & +\lambda tr((A-I)^T(A-I)) \triangleq \min_A f(A) \end{aligned} \tag{6}$$

In the case of discrete random variables, the KL divergence is defined as:

$$KL[P(x)|Q(x)] = \sum_{x \in X} P(x) \log \frac{P(x)}{Q(x)} \tag{7}$$

where $P(x)$, $Q(x)$ are the two probability distributions on the random variable X. The objective function of formula (7) can be expressed as formula (8) based on the formula (7).

Since $(x_i - x_j)^{\mathrm{T}} A(x_i - x_j)$ is a linear function, it can be proved that the objective function $f(A)$ in equation (8) is a smooth convex function. The optimal solution of the objective function (8) can be obtained by using the gradient descent method, the Newton method and other

common optimization methods [37].

$$
\begin{aligned}
&\min_A f(A)\\
&=\min_A \sum_i \left\{KL\left[p_0^A(x_j|x_i)|p^A(x_j|x_i)\right]\right\}+\lambda tr((A-I)^T(A-I))\\
&=\min_A \left\{\sum_{\{i,j|i,j=1,2,\cdots,n,i\neq j\}} p_0^A(x_j|x_i)\log\frac{p_0^A(x_j|x_i)}{p^A(x_j|x_i)}\right.\\
&\quad\left.+\lambda tr((A-I)^T(A-I))\right\}\\
&=\min_A \left\{\sum_{\{i,j|i,j=1,2,\cdots,n,i\neq j\}} \frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\log\left[\frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\cdot\frac{\sum\limits_{m\neq i}e^{-d_{im}^A}}{e^{-d_{ij}^A}}\right]\right.\\
&\quad\left.+\lambda tr((A-I)^T(A-I))\right\}\\
&=\min_A \left\{\sum_{\{i,j|i,j=1,2,\cdots,n,i\neq j\}} \frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\log\frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\right.\\
&\quad+\sum_{\{i,j|i,j=1,2,\cdots,n,i\neq j\}} \frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\log\frac{\sum\limits_{m\neq i}e^{-d_{im}^A}}{e^{-d_{ij}^A}}\\
&\quad\left.+\lambda tr((A-I)^T(A-I))\right\}\\
&=\min_A \left\{\sum_{\{i,j|i,j=1,2,\cdots,n,i\neq j\}} \frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\log\frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\right.\\
&\quad+\sum_{\{i,j|i,j=1,2,\cdots,n,i\neq j\}} \frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\log\frac{\sum\limits_{m\neq i}e^{-(x_i-x_m)^TA(x_i-x_m)}}{e^{-(x_i-x_j)^TA(x_i-x_j)}}\\
&\quad\left.+\lambda tr((A-I)^T(A-I))\right\}\\
&=\min_A \left\{\sum_{\{i,j|i,j=1,2,\cdots,n,i\neq j\}} \frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\log\frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\right.\\
&\quad+\sum_{\{i,j|i,j=1,2,\cdots,n,i\neq j\}} \frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\log\sum_{m\neq i}e^{-(x_i-x_m)^TA(x_i-x_m)}\\
&\quad+\sum_{\{i,j|i,j=1,2,\cdots,n,i\neq j\}} \frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}(x_i-x_j)^TA(x_i-x_j)\\
&\quad\left.+\lambda tr((A-I)^T(A-I))\right\}
\end{aligned}
\tag{8}
$$

B. Model Solving and Algorithm Implementation

In this paper, the gradient descent method is used to solve the optimal solution of the objective function (8). The iterative formula is as follows

$$A_{t+1}=A_t-lr_{t+1}\nabla f(A_t) \tag{9}$$

where t is the number of iterations; lr_{t+1} is the step size of the t+1 th step, lr_{t+1} is updated by the following formula

$$lr_{t+1}=\rho\times\frac{1}{1+decay\times t} \tag{10}$$

where ρ is the learning rate and $decay$ is the attenuation rate. $decay$ is set to a fixed value (0.01) in our experiments. $\nabla f(A_t)$ is the gradient of $f(A)$ at point A_t. By deriving the objective function $f(A)$, we can get the expression of $\nabla f(A)$ as follows

$$
\begin{aligned}
&\nabla f(A)=\\
&\sum_{\{i,j|i,j=1,2,\cdots,n,i\neq j\}} \frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}\frac{-\sum\limits_{m\neq i}e^{-(x_i-x_m)^TA(x_i-x_m)}(x_i-x_m)(x_i-x_m)^T}{\sum\limits_{m\neq i}e^{-(x_i-x_m)^TA(x_i-x_m)}}\\
&+\sum_{\{i,j|i,j=1,2,\cdots,n,i\neq j\}} \frac{|Y_i\cap Y_j|}{|Y_i|\cdot|Y_j|}(x_i-x_j)(x_i-x_j)^T+2\lambda(A-I)
\end{aligned}
\tag{11}
$$

It can be seen that $\nabla f(A_t)$ is a real symmetric matrix, which could ensure that A_{t+1} obtained in each iteration of equation (9) is a real symmetric matrix. Since it is necessary to ensure that the obtained distance matrix A is a positive definite matrix (at least a semi-positive definite matrix), after calculating A_{t+1} using equation (9), we have to calculate the eigenvalue and eigenvector of A_{t+1}, replace the negative eigenvalues with 0, and then get the final A_{t+1}. The detailed flow chart of the proposed algorithm is as follows:

Input: training set $S=\{(x_i,Y_i)|i=1,2,\cdots,n\}$,
hyperparameters λ and ρ,
maximum number of iterations t_{max}.

Step:

Step 1. Initialize $t=0$, $A_t=I$;

Step 2. Calculate the gradient $\nabla f(A_t)$ of $f(A)$ at point A_t according to (11);

Step 3. Update lr_{t+1} according to (10);

Step 4. Update A_{t+1}: $A_{t+1}=A_t-lr_{t+1}\nabla f(A_t)$;

Step 5. Calculate the eigenvalues $\{\lambda_l\}$ and the corresponding eigenvectors $\{u_l\}$ of A_{t+1}, update A_{t+1}: $A_{t+1}=\sum_l\max(\lambda_l,0)u_lu_l^T$;

Step 6. If $t>t_{max}$, then output A_{t+1}, otherwise $t=t+1$, go to **Step 2**;

Output: A_{t+1}

It can be seen that the computational complexity of training model is mainly dominated by computing the gradient (11) which takes about $O(n^2d^2)$ operations, where n is the number of training samples and d is the dimension of the feature vector.

IV. Experiments and Analysis

A. Experimental Setup

In order to verify the performance of the PL-SPFML algorithm proposed in this paper, we compare it with the PL-GMML algorithm [13], and both of them are used as the front end of the PL-KNN algorithm [4]. The process is as follows: firstly, the PL-SPFML and PL-GMML algorithms are used to learns a metric matrix A from the training set, and then the Cholesky decomposition $A=L^{\mathrm{T}}L$ of A is calculated; secondly, the feature vector x of each sample in the original training set and the test set is transformed into a new feature vector Lx to obtain a new training set and test set; finally, the PL-KNN algorithm runs on the new training set and test set. Experiments were conducted on 5 UCI data sets [38] and 4

Table 1: Characteristics of the experimental data sets.

Data sets		#Samples	#Features	#Classes	#Candidate labels		
					min	max	mean
UCI	Ecoli	336	7	8	-	-	-
	Movement	360	90	15	-	-	-
	Vehicle	846	18	4	-	-	-
	CTG	2126	21	10	-	-	-
	Segment	2310	19	7	-	-	-
Real-world	Lost	1122	108	16	1	3	2.23
	FG-NET	1002	262	78	2	11	7.48
	MSRCv2	1758	48	23	1	7	3.16
	BirdSong	4998	38	13	1	4	2.18

real-world partial label problem data sets [17]. Table 1 gives characteristics of the experimental data sets. Since the UCI data sets are traditional multi-class dataset, we first transform them into partial label data sets by two parameters p and r before conducting experiments, where p denotes the proportion of partial label samples in the data set, and r denotes the number of candidate category label for each partial label sample except for the true category label. The specific transformation process is described in [13]. In the experiments of this paper, we take two values for p and r respectively, $p = 0.3$ or 0.6, $r = 1$ or 0.6. Therefore, for each UCI data set, 4 partial label data sets are generated by different combinations of (p, r). The real-world partial label data set Lost is composed of 1122 face images of 16 people cutting from TV series, and each image is represented by 108 features obtained by the PCA method [5]; the FG-NET data set comes from an age recognition problem based on facial images [11]; the MSRCv2 data set contains 1758 segmented image regions from 23 categories, and each region consists of a 48-dimensional histogram and gradient attributes [17]; the Birdsong data set contains syllables of 4998 birds in 13 species, each syllable consists of 38 attributes, and birds appearing within 10 seconds of the syllable are placed in the candidate label set of the syllable [18].

For the PL-SPFML algorithm, on each data set, the values of the parameters λ and ρ will be selected from the set $\{0, 1 \times 10^{-6}, 1 \times 10^{-5}, \cdots, 1 \times 10^{6}\}$ and $\{0, 1 \times 10^{-6}, 1 \times 10^{-5}, \cdots, 1 \times 10^{-1}\}$ by cross validation method; the value of the parameter k on the UCI data set and the real-world partial label data set will be obtained respectively from the sets $\{k|k = 5, 8, 11, 14, 17\}$ and $\{k|k = 3, 5, 7, 9, 11, 13, 15\}$ by cross validation method, where k is the value of k in the PL-KNN algorithm. For the PL-GMML algorithm, all parameters are set and selected according to the requirements in [13]. All experimental results in this section were computed based on 10 times of 5-fold cross-validation, and all experimental data sets were normalized in the pre-processing stage.

B. Parameter Sensitivity Analysis

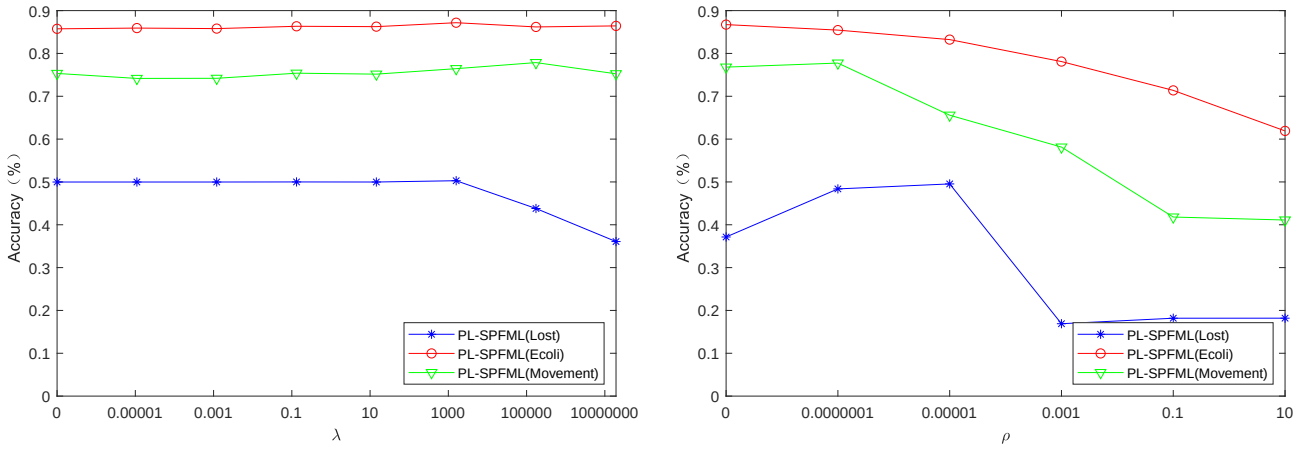
In order to verify the influence of hyper-parameters λ and ρ on the accuracy of PL-SPFML algorithm , we first conduct experiments on one real-world partial label data set (Lost) and two UCI data sets (Ecoli and Movement)($r = 1$, $p = 0.3$). Fig.1 shows the experimental results on these three data sets when one of the parameters is changed while the other one remains fixed. It can be seen from Fig. 1(a) that when λ changes, the highest accuracy of the algorithm on each data set is higher than the accuracy at $\lambda = 0$, so this shows that it is useful to add a regularization term $tr((A - I)^T(A - I))$ to the objective function. It can be seen from Fig. 1(b) that the learning rate ρ has a greater impact on the algorithm, and the impact on different data sets is different. Because the impact of λ and ρ on the accuracy of the algorithm is different on different data sets and there is no rule to follow, therefore, in the experiments of this paper, the values of the parameters λ and ρ are obtained by using cross validation method. The detailed description can be found in the previous subsection.

C. Experimental Results

Table 2 shows the classification accuracy of the original PL-KNN algorithm, and the PL-KNN algorithm with the PL-SPFML algorithm and the PL-GMML algorithm as its front-end on the 5 UCI data sets. The best result on each dataset has been shown in boldface. It can be seen from Table 2 that both the PL-SPFML algorithm and the PL-GMML algorithm can significantly improve the prediction accuracy of the PL-KNN algorithm, while compared the PL-SPFML algorithm with the PL-GMML algorithm, the former is better on about half of these data sets, and the latter is better on others.

Table 3 presents the experimental results of each algorithm on the real-world partial label data sets. The same as in Table 2, the best result on each data set has been shown in boldface. It can be seen from Table 3 that the PL-SPFML algorithm is better than the PL-GMML algorithm on the three data sets. On the Lost data set, the PL-SPFML algorithm can be nearly 8 percentage points higher than the PL-GMML algorithm; On the FG-NET data set, the PL-SPFML algorithm can be nearly 2 percentage points higher than the PL-GMML algorithm.

Based on the above analysis, PL-SPFML algorithm and PL-GMML algorithm have their own advantages on UCI dataset, while PL-SPFML algorithm is superior to PL-GMML algorithm on the real-world partial label data



(a) varying λ (ρ is fixed) (b) varying ρ (λ is fixed)

Fig. 1: Parameter sensitivity analysis on the Lost, Ecoli and Movement data sets.

Table 2: Classification accuracy comparison on UCI data sets(%)

Data sets	Algorithms	Accuracy($mean\pm std.$)			
		$r=1,p=0.3$	$r=1,p=0.6$	$r=3,p=0.3$	$r=3,p=0.6$
Ecoli	PL-KNN	86.30±0.61	86.04±0.71	85.42±0.99	82.42±1.06
	PL-GMML+ PL-KNN	85.98±0.88	84.46±1.32	85.15±0.80	82.11±1.97
	PL-SPFML+PL-KNN	**86.90±0.99**	**86.66±0.87**	**86.04±0.76**	**82.86±1.15**
Movement	PL-KNN	75.58±1.36	75.22±1.02	74.69±1.41	74.08±1.10
	PL-GMML+ PL-KNN	**79.81±1.21**	**78.33±0.96**	**78.83±1.44**	**76.56±1.98**
	PL-SPFML+PL-KNN	77.86±1.43	76.75±1.33	76.61±1.92	75.31±1.35
Vehicle	PL-KNN	71.75±0.71	68.04±0.96	69.04±0.88	65.27±0.91
	PL-GMML+ PL-KNN	**78.92±0.74**	**78.00±1.03**	**78.05±0.81**	**74.86±1.13**
	PL-SPFML+PL-KNN	75.64±0.62	73.40±1.32	73.90±0.96	69.27±1.15
CTG	PL-KNN	74.69±0.28	74.27±0.45	73.87±0.62	72.13±0.61
	PL-GMML+ PL-KNN	**76.81±0.52**	**75.22±0.79**	75.80±0.35	**70.95±0.63**
	PL-SPFML+PL-KNN	74.88±0.56	74.57±0.60	**76.23±0.54**	69.95±0.50
Segment	PL-KNN	94.39±0.22	94.27±0.22	94.39±0.10	92.81±0.19
	PL-GMML+ PL-KNN	95.67±0.23	95.27±0.45	95.55±0.33	**93.84±0.40**
	PL-SPFML+PL-KNN	**97.00±0.21**	**95.97±0.16**	**96.65±0.32**	93.67±0.46

sets. In addition, the performance of PL-SPFML algorithm on the real-world partial label data sets is better than that of on the UCI data sets. In the partial label data sets transformed by UCI data sets, there is no correlation among the candidate labels of each training sample, which may be the reason why the performance of the proposed algorithm on the UCI data sets is not as good as that of on the real-world partial label data sets. In the next section, we will try to analyze the reason of this phenomenon.

D. Reason Analysis of the Phenomenon That the Performance of PL-SPFML Algorithm on Real-world Partial Label Data Sets is Better Than That of on UCI Data Sets

We know that in the real partial label problems, the candidate labels of the samples are usually related. In order to verify whether the proposed algorithm is capable of mining the correlation between candidate labels and thereby resulting its performance on real-world partial label data sets is better than that of on the UCI data sets, we use two dimensional Gaussian distribution randomly generate a partial label data set (named Normal-UCI data set) with no correlation between the candidate labels of the samples and a partial label data set (named Normal-partial data set) with a certain correlation between the candidate labels of the samples. The two data sets are generated as follows: first we generate three sets of points based on three two-dimensional Gaussian distributions with different mean and variance, and these three point sets are combined to form a standard three-category data set called Normal data set, Fig.2 shows the sample distribution of the Normal data set; Then 45% of the samples in the Normal data set are randomly added one other label to form a partial label data set, called the Normal-UCI data set, whose sample has no correlation among the candidate labels since the candidate labels of each partial label sample in this data set are randomly generated; In addition, for every two categories of the Normal data set, 45% samples located at

Table 3: Classification accuracy comparison on real-world partial label data sets(%)

Algorithms	Lost	FG-NET	MSRCv2	BirdSong
PL-KNN	36.10±0.73	4.65±0.40	44.27±0.54	63.11±0.19
PL-GMML+ PL-KNN	42.60±0.80	5.65±0.46	44.25±0.42	**66.57±0.35**
PL-SPFML+ PL-KNN	**50.86±0.34**	**7.37±1.12**	**44.62±0.54**	63.13±0.17

or near the junction of these two categories in Normal data set are annotated as partial label samples and the labels of these two categories are annotated as the candidate labels of these partial label samples, which can also form a partial label data set, called the Normal-partial data set. It is obvious that there is a certain correlation between the candidate category labels of the samples in Normal-partial data set. Fig.3 and Fig.4 respectively show the sample distribution of the Normal-UCI and Normal-partial data sets. It can be seen that the sample distribution, number of points and proportion of partial label samples in the two data sets are the same, except that the partial label samples in the Normal-UCI data set are evenly distributed, while the partial label samples in the Normal-partial data set are only distributed in junction of every two categories.

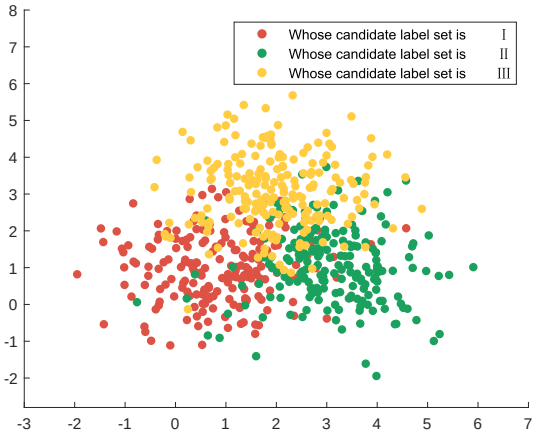


Fig. 2: Sample distribution of Normal data set.

Fig.5 shows the prediction accuracy of the PL-SPFML algorithm on the Normal-UCI and Normal-partial data sets as the hyper-parameters λ and ρ values increase. When one of the two parameters is changed while the other parameter remains fixed. It can be seen from Fig.5, the accuracy of the PL-SPFML algorithm on the Normal-partial data set is higher than that of the Normal-UCI data set. Therefore, this can explain to some extent the phenomenon that the results of the PL-SPFML algorithm on the real partial label data sets are better than the results on the UCI data set, because it can mine the correlation between the candidate labels but the candidate labels of the samples on the UCI data set usually have no correlation. Since the candidate labels of the samples are usually correlated in the real

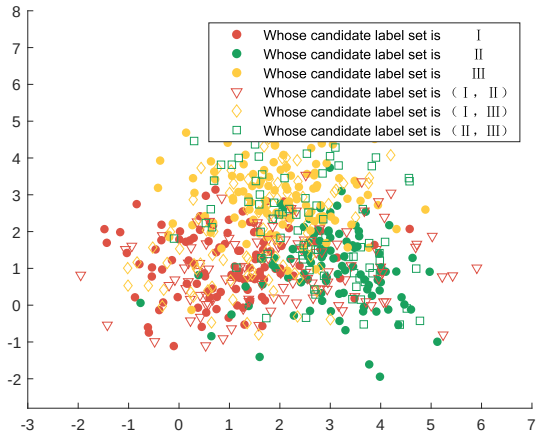
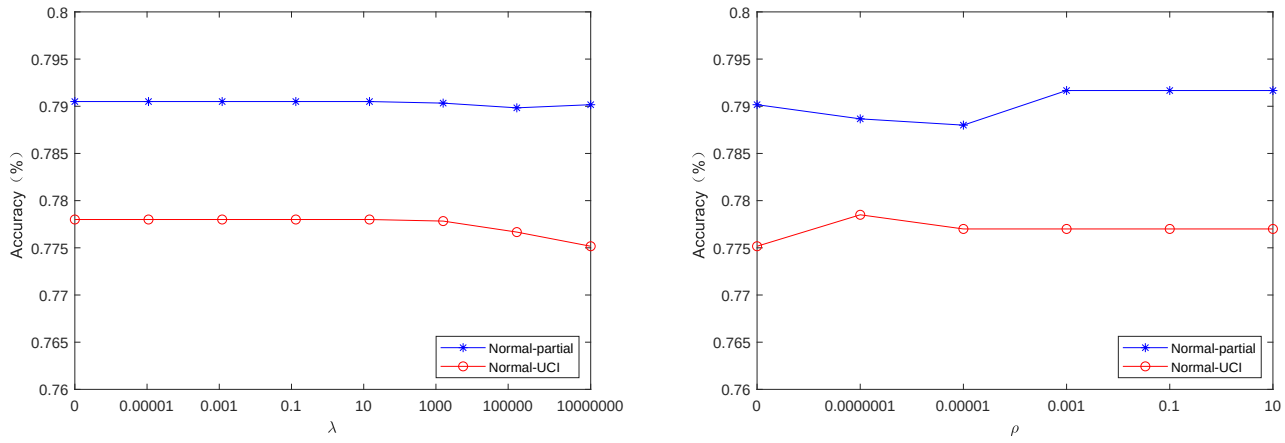


Fig. 3: Sample distribution of Normal-UCI data set.

partial label problems, the algorithm of this paper can deal with the real partial label problem better than the PL-GMML algorithm.

V. Conclusion

Partial label learning is a new weakly supervised learning framework with broad application prospects. It has recently been discovered that metric learning techniques can effectively improve the accuracy of partial label learning algorithms. However, because it is difficult to ascertain similar pairs from training samples, at present there are few metric learning algorithms for partial label learning framework. In view of this, this paper proposes a similar pair-free partial label metric learning algorithm. The experimental results show that the proposed algorithm can improve the accuracy of PL-KNN algorithm better than existing algorithms on most of the real-world partial label data sets. Of course, the proposed algorithm still has a lot of room for improvement. For example, the users need to extract the features of the samples based on some feature extraction technologies themselves before using the proposed algorithms to deal with their application problems. Recently, with the emergence of deep learning technologies, end-to-end learning algorithms have received widespread attention due to the ability of integrating the processes of feature extraction and learning algorithm construction. In the future work, we will try to develop an end-to-end partial label metric learning algorithm by replacing the model used in this paper with a deep learning model. In addition, we will try other solving methods to quickly



(a) varying λ (ρ is fixed) (b) varying ρ (λ is fixed)

Fig. 5: Results of PL-SPFML algorithm on Normal-UCI and Normal-partial data sets.

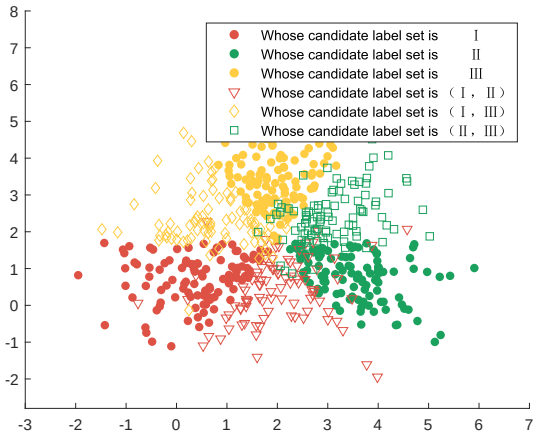


Fig. 4: Sample distribution of Normal-partial data set.

calculate the optimal solution of objective function.

Acknowledgment

This research was funded by the National Natural Science Foundation of China (62102061, 61972068, 62072152), the Natural Science Foundation of Liaoning Province (2020-MS-134, 2020-MZLH-29, 20180550625), and the Scientific Research Funding Project of the Educational Department of Liaoning Province (LJKZ0022).

References

[1] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy. Learning from crowds. *Journal of Machine Learning Research*, 11(Apr):1297–1322, 2010.

[2] N. Nguyen and R. Caruana. Classification with partial labels. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 551–559. ACM, 2008.

[3] C.-H. Chen, V. M. Patel, and R. Chellappa. Learning from ambiguously labeled face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(7):1653–1667, 2017.

[4] E. Hüllermeier and J. Beringer. Learning from ambiguously labeled examples. *Intelligent Data Analysis*, 10(5):419–439, 2006.

[5] T. Cour, B. Sapp, and B. Taskar. Learning from partial labels. *Journal of Machine Learning Research*, 12(May):1501–1536, 2011.

[6] J. Luo and F. Orabona. Learning from candidate labeling sets. In *Advances in Neural Information Processing Systems*, pages 1504–1512, 2010.

[7] F. Yu and M.-L. Zhang. Maximum margin partial label learning. In *Asian Conference on Machine Learning*, pages 96–111, 2016.

[8] S.-j. Zhang and J. Chai. Partial label learning algorithm based on maximum margin. *Science Technology and Engineering*, 18(28):109–115, 2018.

[9] A. Beygelzimer and J. Langford. The offset tree for learning with partial labels. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 129–138. ACM, 2009.

[10] E. Côme, L. Oukhellou, T. Denoeux, and P. Aknin. Learning from partially supervised data using mixture models and belief functions. *Pattern Recognition*, 42(3):334–348, 2009.

[11] M.-L. Zhang, B.-B. Zhou, and X.-Y. Liu. Partial label learning via feature-aware disambiguation. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1335–1344. ACM, 2016.

[12] M.-L. Zhang and F. Yu. Solving the partial label learning problem: An instance-based approach. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[13] Y. Zhou and H. Gu. Geometric mean metric learning for partial label data. *Neurocomputing*, 275:394–402, 2018.

[14] Y.-C. Chen, V. M. Patel, R. Chellappa, and P. J. Phillips. Ambiguously labeled learning using dictio-

naries. *IEEE Transactions on Information Forensics and Security*, 9(12):2076–2088, 2014.

[15] C. Gong, T. Liu, Y. Tang, J. Yang, J. Yang, and D. Tao. A regularization approach for instance-based superset label learning. *IEEE Transactions on Cybernetics*, 48(3):967–978, 2017.

[16] L. Feng and B. An. Partial label learning with self-guided retraining. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3542–3549, 2019.

[17] M.-L. Zhang, F. Yu, and C.-Z. Tang. Disambiguation-free partial label learning. *IEEE Transactions on Knowledge and Data Engineering*, 29(10):2155–2167, 2017.

[18] L. Liu and T. G. Dietterich. A conditional multinomial mixture model for superset label learning. In *Advances in Neural Information Processing Systems*, pages 548–556, 2012.

[19] L. Feng and B. An. Leveraging latent label distributions for partial label learning. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 2107–2113, 2018.

[20] C. Li, J. Zhang, and Z. Chen. Structured output learning with candidate labels for local parts. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 336–352. Springer, 2013.

[21] C.-Z. Tang and M.-L. Zhang. Confidence-rated discriminative partial label learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[22] X. Wu and M.-L. Zhang. Towards enabling binary decomposition for partial label learning. In *IJCAI*, pages 2868–2874, 2018.

[23] J. Wang and M.-L. Zhang. Towards mitigating the class-imbalance problem for partial label learning. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2427–2436. ACM, 2018.

[24] G. Lyu, S. Feng, T. Wang, C. Lang, and Y. Li. Gm-pll: graph matching based partial label learning. *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[25] D.-B. Wang, L. Li, and M.-L. Zhang. Adaptive graph guided disambiguation for partial label learning. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 83–91, 2019.

[26] Y. Zhou, J. He, and H. Gu. Partial label learning via gaussian processes. *IEEE Transactions on Cybernetics*, 47(12):4443–4450, 2016.

[27] N. Xu, J. Lv, and X. Geng. Partial label learning via label enhancement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5557–5564, 2019.

[28] G. Lyu, S. Feng, T. Wang, and C. Lang. A self-paced regularization framework for partial-label learning. *IEEE Transactions on Cybernetics*, 2020.

[29] Y. Yao, J. Deng, X. Chen, C. Gong, J. Wu, and J. Yang. Deep discriminative cnn with temporal ensembling for ambiguously-labeled image classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12669–12676, 2020.

[30] S. Wang and R. Jin. An information geometry approach for distance metric learning. In *Artificial intelligence and statistics*, pages 591–598. PMLR, 2009.

[31] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of machine learning research*, 10(2), 2009.

[32] Y. Verma and C. Jawahar. Image annotation using metric learning in semantic neighbourhoods. In *European Conference on Computer Vision*, pages 836–849. Springer, 2012.

[33] P. Zadeh, R. Hosseini, and S. Sra. Geometric mean metric learning. In *International Conference on Machine Learning*, pages 2464–2471, 2016.

[34] A. Globerson and S. T. Roweis. Metric learning by collapsing classes. In *Advances in Neural Information Processing Systems*, pages 451–458, 2006.

[35] M. Huai, C. Miao, Y. Li, Q. Suo, L. Su, and A. Zhang. Metric learning from probabilistic labels. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1541–1550, 2018.

[36] J. Goldberger, G. E. Hinton, S. Roweis, and R. R. Salakhutdinov. Neighbourhood components analysis. *Advances in neural information processing systems*, 17:513–520, 2004.

[37] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004.

[38] A. Asuncion and D. Newman. Uci machine learning repository, 2007.

[39] A. Almutairi, A. Gegov, M. Adda, and F. Arabikhan. Conceptual artificial intelligence framework to improving English as second language. *WSEAS Transactions on Advances in Engineering Education*, 17: 87-91, 2020.

[40] P. Lokkas, E. Papadimitriou, N. Alamanis, G. Papageorgiou, D. Christodoulou, T. Chrisanidis. Significant foundation techniques for education: a critical analysis. *WSEAS Transactions on Advances in Engineering Education*, 18: 7-26, 2021.

Creative Commons Attribution License 4.0 (Attribution 4.0 International , CC BY 4.0)
This article is published under the terms of the Creative Commons Attribution License 4.0

https://creativecommons.org/licenses/by/4.0/deed.en_US