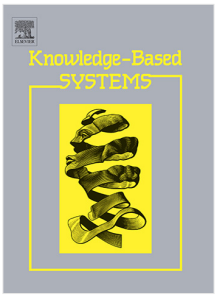
Journal Pre-proof



Two-stage label distribution learning with label-independent prediction based on label-specific features

Gui-Lin Li, Heng-Ru Zhang, Fan Min, Yu-Nan Lu

PII: S0950-7051(23)00176-4
DOI: https://doi.org/10.1016/j.knosys.2023.110426
Reference: KNOSYS 110426

To appear in: *Knowledge-Based Systems*

Received date : 12 September 2022
Revised date : 17 February 2023
Accepted date : 23 February 2023

Please cite this article as: G.-L. Li, H.-R. Zhang, F. Min et al., Two-stage label distribution learning with label-independent prediction based on label-specific features, *Knowledge-Based Systems* (2023), doi: https://doi.org/10.1016/j.knosys.2023.110426.

Revised Manuscript (Clean Version)

Two-Stage Label Distribution Learning with Label-Independent Prediction Based on Label-Specific Features

Gui-Lin Li[a], Heng-Ru Zhang[a,*], Fan Min[a,b], Yu-Nan Lu[c]

[a]School of Computer Science, Southwest Petroleum University, Chengdu 610500, China
[b]Institute for Artificial Intelligence, Southwest Petroleum University, Chengdu 610500, China
[c]School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, 210094, China

Abstract

Label distribution learning (LDL) explicitly models label ambiguity based on the degree to which each label describes an instance. Existing LDL algorithms typically consider all features shared by all class labels. However, irrelevant or redundant information in the feature space can degrade the performance of the model. In this study, a novel two-stage LDL algorithm with label-independent prediction based on label-specific features (TSLIP) is proposed. The first stage combines sparse feature selection and instance correlation constraints. The original high-dimensional input feature space is mapped to a derived low-dimensional label-independent prediction space. The mapping operation not only supports the extraction of label-specific features but also brings the derived space closer to the label distribution space. The second stage employs the Pearson coefficient to explore the label correlation to enhance the performance of the model. Subsequently, the label-independent predictions are normalized to obtain the final label distribution. Extensive experiments are conducted on twelve real-world datasets. The results of the seven measures demonstrate the superiority of our algorithm compared with several state-of-the-art methods.

Keywords: Label correlations, Label distribution learning, Label-independent prediction, Label-specific features.

1. Introduction

Label ambiguity [1] is a popular topic in the fields of machine learning and artificial intelligence. Early label ambiguity is dominated by single-label learning (SLL), that is, assigning a unique label to an instance, for example, assigning "dog" to a picture or rating a paper "7." However, in several scenarios, an instance may contain numerous labels, e.g., a photo is both a landscape and portrait. Thus, multilabel learning (MLL) [2] has been designed to handle such situations. MLL can further address label ambiguity in applications such as image classification [3, 4, 5, 6] and text recognition [7, 8, 9] by assigning several labels for each instance.

In addition to whether a label belongs to an instance, the importance of the label to the instance is crucial. Label distribution learning (LDL) [10] encodes labels in the form of histograms or probability distributions, essentially aiming to answer the question, "how much does each label describe the instance?" LDL uses more general semantics compared with MLL to address label ambiguity more effectively. Geng et al. [11] introduced the concept of label distribution in the age-estimation problem and proposed an IIS-LLD algorithm. Zhang et al. [12] introduced LDL in crowd-counting problems to reduce the impact of data imbalance in the training set. Ling et al. [13] proposed a bidirectional sliding window method to generate label distributions for individual video segments.

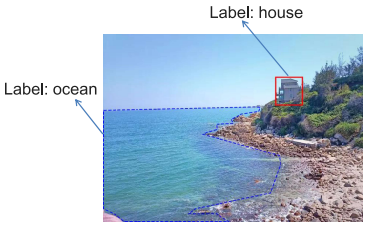


Figure 1: Description of label-specific features. A specific label is typically derived from a subset of all features. The label "house" is associated only with the features in the red box, whereas "ocean" is associated only with the features in the blue dashed frame.

However, the aforementioned studies do not consider label-specific features [14, 15]; This implies that a label is typically associated with only some features and has no clear relationship with others. Figure 1 shows an example of label-specific features; here, all the pixels constituting the image can be considered as the feature set. Intuitively, this instance could be assigned the label of "house" as the red box indicates that there is a house here. Similarly, the content of the image marked by the blue dashed frame indicates that the instance should have the label "ocean." A specific label of the picture is often determined only by the part of the pixel value of the picture. The

*Corresponding author.
E-mail addresses: lgl@live.com (Gui-Lin Li), zhanghrswpu@163.com (Heng-Ru Zhang), minfan@swpu.edu.cn (Fan Min), luyn@njust.edu.cn (Yu-Nan Lu).

features that can infer specific labels are label-specific features.

In this regard, LDL can mine more information and significantly improve model performance by considering label-specific features. Ren et al. [16] attempted to learn specific labels for a single label and common features for all labels to enhance the model. To improve tolerance to noise, the ProLSFEO-LDL [17] method utilizes both label-specific features and prototype selection. However, in most real-world applications, even if all instances have the same values for the specific features of a given label, different instances are characterized by the label in variable degrees. Figure 2 presents an example.

Two persons may have the same feature values for the label "obesity" but different feature values for the label "cancer" (one person has cancer, whereas the other does not). When assessing their health, the obesity distribution in cancer patients is most likely to be markedly lower than that in others.

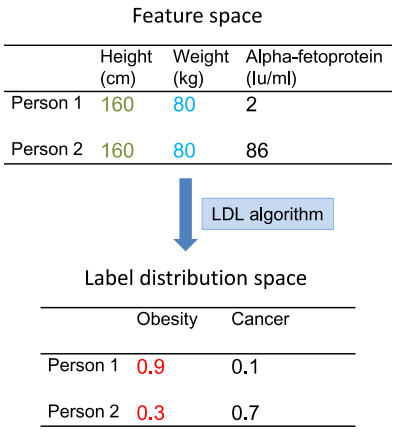


Figure 2: Common phenomenon in LDL. Different instances have the same feature values (height and weight) but different label distribution values (obesity).

To address this problem, this study proposed a two-stage LDL method with label-independent prediction based on label-specific features (TSLIP). Instead of directly generating distributions from features, a derived label-independent prediction space was injected between the feature space and label distribution space, and a two-stage learning model was built. Figure 3 shows the framework of this algorithm.

Label-independent prediction is a prediction value that considers only the specific feature of each label and foretells that label in the derived space. Instances should have the same independent prediction value on a label if they have the same specific feature value. Herein, a label-independent prediction space was introduced to effectively mine and preserve the data information in the feature space. Additionally, the label-independent prediction satisfied the instance correlation constraint. Essentially, instances with similar label-specific features should have similar label-independent prediction values. Furthermore, in the final label distribution space, the label correlation constraint was adopted to improve the model performance.

The main contributions of this study are summarized as follows.

1) The phenomenon that different instances have specific feature values with the same label but different distribution values was highlighted. To the best of our knowledge, this problem has never been raised before.

2) The above problem was addressed by deriving a label-independent prediction space between the feature space and the label distribution space. Based on this space, a new two-stage learning model was proposed.

3) A sparse weighting matrix was designed to transform the feature space into a label-independent prediction space. This matrix not only selects label-specific features but also assigns different weights to each feature.

4) Instance correlation and label correlation were incorporated in these two stages, respectively, thereby effectively improving the generalization ability of the model.

The remainder of this paper is organized as follows. Section 2 provides a brief review of related studies. Section 3 presents the details of the proposed method. Section 4 analyzes the experimental results. Finally, Section 5 concludes the paper and provides suggestions for future work. The implementation of TSLIP is available at https://github.com/zhanghrswpu/TSLIP.

2. Related works

This section presents relevant work, including design strategies of LDL algorithms, introduction to label-specific features and presentation of label enhancement.

2.1. Label distribution learning

LDL generally involves three strategies in algorithm design. The first is the problem transformation (PT) strategy, which transforms the current problem into an existing problem. This is a strategy for fitting data into an algorithm. Typical algorithms include the PT-Bayes [10] and PT-SVM [18]. Both transform the LDL problem into an SLL problem. The second is the algorithm adaptation (AA) strategy, which directly uses an existing algorithm to deal with the LDL problem. This strategy adapts the algorithm to the data. Typical algorithms include AA-kNN [11] and AA-BP [19]. They obtain the label distribution value using the distribution of k neighbors and train a neural network. The third is the specialized algorithm (SA) strategy for modeling the input-to-output. Typical algorithms include the SA-IIS and SA-BFGS [10].

As an extension of the SA strategy, subsequent researchers conducted LDL through various ways. For example, Yang et al. [20] developed a multitask deep framework by jointly optimizing classification and distribution prediction. Zhao et al. [21] proposed an approach to simultaneously learn the label distribution and exploit label correlations based on optimal transport

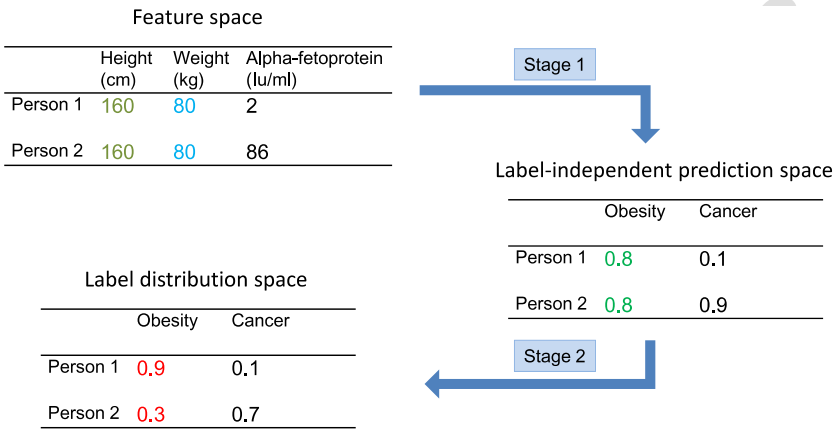


Figure 3: Illustration of the proposed label-independent prediction space and two-stage label distribution learning model. If the specific eigenvalues of a label are all the same, its label value in the label-independent prediction space is also the same. This implies that the features are the same but the distribution values are different in label distribution learning. The label-independent prediction space can be viewed as a missing space in label distribution learning.

theory. Jia et al. [22] modeled the correlation between different labels. Zhang et al. [23] based on a cosine augmentation method and label correlation for LDL. Jia et al. [24] proposed an LDL method based on a label-ranking exploration.

Numerous studies [25, 26, 27, 16] have confirmed the importance of correlation in LDL. In this study, both label correlation and instance correlation were exploited to improve the generalization ability of the model.

2.2. *Label-specific features*

Label-specific features is a processing strategy that customizes different features for each label instead of having all labels that share the same original features. It originates from multi-label learning and aims to extract features that are significantly related to each label to reduce noise from irrelevant features. Label-specific features can be divided into two categories based on their creation. The first is the transformation of the original features into a derived feature space. For example, LIFT [14] performs cluster analysis on positive and negative instances of each label, and then uses the Euclidean distance of each instance to the cluster center as a new feature. The LSDM [28] sets different ratio parameter values to cluster positive and negative instances of the same label. It reconstructs the label-specific feature space containing distance and spatial topology information. The second is to assign each label a specific subset of the original features, e.g., [29, 30] used $l_{2,1}$-norm and l_1-norm for feature sparse selection, respectively. Numerous studies [31, 32, 33] have demonstrated that label-specific features are an efficient processing strategy in multi-label learning.

Label-specific features have also been employed in several approaches for LDL. LDLSF [16] completes label-specific feature selection using two sparse weight matrices. ProLSFEO-LDL [17] uses the CHC framework, in which the search space is represented by a chromosome (or person), where the prototype choice and label-specific features are coded. In FSFL [34], a subset of optimal and relevant features is selected from the top-ranked features based on the feature weights derived from two different strategies.

Herein, based on the correlation theory of label-specific features, a label-independent prediction space was derived and a two-stage LDL model was constructed. This is a new attempt at label-specific features in LDL. The experiments demonstrated the effectiveness of the proposed method.

2.3. *Label enhancement*

LDL can handle label ambiguity better by labeling an instance with a set of label distributions. However, the training sets required for LDL are limited because labeling is difficult and expensive. To address this issue, label enhancement (LE) [35] has been proposed to convert commonly used logical label datasets into distributed datasets. Numerous studies [36, 37, 38, 39] have demonstrated that label enhancement is helpful and advantageous for LDL. LE provides additional methods for data labeling as a supplement and extension of LDL.

3. TSLIP approach

This section proposes the novel TSLIP. First, the problem is defined. Second, a detailed description of the proposed algorithm is provided. Third, an optimization method for this algorithm is introduced. Finally, the time complexity of TSLIP is analyzed and a running example is provided.

3.1. *Problem formulation*

Let $\mathbf{Z} = \{z_1, z_2, \ldots, z_c\}$ be the complete set of labels, where c denotes the number of labels and z_i denotes the i-th label. An LDL system is a tuple $(\mathbf{X}, \mathbf{D})$ where $\mathbf{X} = [\mathbf{x}_1; \mathbf{x}_2; \ldots; \mathbf{x}_n] = [x_{ij}]_{n\times m} \in \mathbb{R}^{n\times m}$ is the feature matrix, and

Table 1: Notations

Notation	Meaning
$\mathbf{X}$	The feature matrix of instances
$\mathbf{D}$	The ground-truth label distribution matrix
$\widehat{\mathbf{D}}$	Predicted label distribution matrix
n	Total number of instances
m	Total number of features
c	Total number of labels
$\mathbf{W}$	The coefficient matrix of the label distribution for the features
$\mathbf{Y}$	The label-independent prediction matrix
$\mathbf{W_1}$	The coefficient matrix of the label-independent prediction for the features
$\mathbf{W_2}$	The coefficient matrix of the label distribution for the label-independent prediction
$\mathbf{W_3}$	Equivalent matrix of $\mathbf{W_1}$
$\mathbf{x}_i$	The feature vector of i-th instance
$\mathbf{d}_i$	The label distribution vector of i-th instance
$\mathbf{y}_i$	The label-independent prediction vector of i-th instance
d_{ij}	The j-th ground-truth label distribution value of instance $\mathbf{x}_i$
$\widehat{d}_{ij}$	The j-th predicted label distribution value of instance $\mathbf{x}_i$
y_{ij}	The j-th label-independent prediction value of instance $\mathbf{x}_i$
$\mathbf{S}$	Correlation matrix between labels
$\mathbf{S}'$	Correlation matrix between instances

$\mathbf{D} = [\mathbf{d}_1; \mathbf{d}_2; \ldots; \mathbf{d}_n] = [d_{ik}]_{n\times c} \in [0,1]^{n\times c}$ is the label distribution matrix that satisfies $\sum_{k=1}^{c} d_{ik} = 1$. In addition, n denotes the number of instances, m denotes the number of features, $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{im}]$ is the feature vector of the ith instance, and x_{ij} represents the jth feature value of $\mathbf{x}_i$. $\mathbf{d}_i = [d_{i1}, d_{i2}, \ldots, d_{ic}]$ denotes the label distribution associated with $\mathbf{x}_i$, d_{ik} represent the kth label distribution value of $\mathbf{x}_i$. d_{ik} represents the description degree of the label z_k to $\mathbf{x}_i$. The goal of LDL is to learn model $\Theta : \mathbf{X} \to \mathbf{D}$.

Table 1 lists a summary of the main notations used in this study.

3.2. Proposed method

Herein, $\mathbf{XW} = \widehat{\mathbf{D}}$ was employed as the fundamental model, where $\mathbf{W} \in \mathbb{R}^{m\times c}$ denotes the regression coefficient matrix. Label-independent prediction serves as a transitional state between the feature and the distribution, which divides the model into two parts. Let $\mathbf{Y} \in \mathbb{R}^{n\times c}$ be the label-independent prediction matrix, then our model can be defined as follows.

$$\mathbf{XW}_1\mathbf{W}_2 = \mathbf{YW}_2 = \widehat{\mathbf{D}}, \tag{1}$$

where $\mathbf{W}_1 \in \mathbb{R}^{m\times c}$ and $\mathbf{W}_2 \in \mathbb{R}^{c\times c}$ are the coefficient matrices of each part.

Subsequently, the formation process of the objective function is described.

Label-independent prediction was obtained using a sparse feature selection model. The general objective function can be defined as:

$$\min_{\mathbf{W}} \sum_{i=1}^{n} \|\mathbf{x}_i\mathbf{W} - \mathbf{d}_i\|_2^2 + \lambda \mathsf{R}(\mathbf{W}), \tag{2}$$

where $\mathsf{R}(\mathbf{W})$ is a regularizer that imposes a sparsity constraint on $\mathbf{W}$ and λ is the regularization parameter.

As $l_{2,1}$-norm is often used as a sparse regularizer [40, 41], our objective function can be written as:

$$\begin{aligned} \min_{\mathbf{W}_1,\mathbf{W}_2} \ & \|\mathbf{XW}_1\mathbf{W}_2 - \mathbf{D}\|_F^2 + \lambda\|\mathbf{W}_1\|_{2,1} \\ \text{s.t.} \ & \mathbf{XW}_1\mathbf{W}_2 \times \mathbf{1}^{c\times 1} = \mathbf{1}^{n\times 1} \\ & \mathbf{XW}_1\mathbf{W}_2 \geq \mathbf{0}^{n\times c}, \end{aligned} \tag{3}$$

where $\|\cdot\|_F$ denotes the Frobenius norm of a matrix, $\|\cdot\|_{2,1}$ denotes the $l_{2,1}$ norm of a matrix, $\mathbf{1}^{c\times 1}$ and $\mathbf{1}^{n\times 1}$ represent matrices with element values of 1, and $\mathbf{0}^{n\times c}$ represents a matrix in which all elements are 0. Further, $\mathbf{XW}_1\mathbf{W}_2 \times \mathbf{1}^{c\times 1} = \mathbf{1}^{n\times 1}$ is the fundamental constraint of LDL that the sum of all the distribution values for an instance should equal 1.

Meanwhile, the label correlation constraint has been shown to yield good performance gains for LDL [16, 22, 25]. With the help of the labels associated with a given label, it can adjust the distribution value of that label. If a strong association exists between the two labels, their distribution values should be similar. In our algorithm, a label correlation constraint was imposed directly on the predicted distribution values. Therefore, the objective function can be rewritten as:

$$\begin{aligned} \min_{\mathbf{W}_1,\mathbf{W}_2} \ & \|\mathbf{XW}_1\mathbf{W}_2 - \mathbf{D}\|_F^2 + \lambda_1\|\mathbf{W}_1\|_{2,1} \\ & + \lambda_2 \sum_{i=1}^{n}\sum_{p=1}^{c}\sum_{q=1}^{c} S_{pq}(\widehat{d}_{ip} - \widehat{d}_{iq})^2 \\ \text{s.t.} \ & \mathbf{XW}_1\mathbf{W}_2 \times \mathbf{1}^{c\times 1} = \mathbf{1}^{n\times 1} \\ & \mathbf{XW}_1\mathbf{W}_2 \geq \mathbf{0}^{n\times c}, \end{aligned} \tag{4}$$

where S_{pq} represents the correlation coefficient between the distributions of the p-th label z_p and q-th label z_q, λ_1 and λ_2 are the regularization parameters for the different regularization terms. S_{pq} can be calculated using the Pearson correlation function [42]:

$$S_{pq} = \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{d_{ip} - \mu_p}{\sigma_p}\right)\left(\frac{d_{iq} - \mu_q}{\sigma_q}\right), \tag{5}$$

where μ_p and σ_p denote the mean and standard deviation of the p-th label distribution for all instances, respectively. μ_q and σ_q are similar to μ_p and σ_p.

Furthermore, instances with similar label-specific features should exhibit similar label-independent prediction values. By adding this constraint, our objective function can be rewritten as:

$$\begin{aligned} \min_{\mathbf{W}_1,\mathbf{W}_2} \ & \|\mathbf{XW}_1\mathbf{W}_2 - \mathbf{D}\|_F^2 + \lambda_1\|\mathbf{W}_1\|_{2,1} \\ & + \lambda_2 \sum_{i=1}^{n}\sum_{p=1}^{c}\sum_{q=1}^{c} S_{pq}(\widehat{d}_{ip} - \widehat{d}_{iq})^2 \\ & + \lambda_3 \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{c} S'_{ij}(y_{ik} - y_{jk})^2 \\ \text{s.t.} \ & \mathbf{XW}_1\mathbf{W}_2 \times \mathbf{1}^{c\times 1} = \mathbf{1}^{n\times 1} \\ & \mathbf{XW}_1\mathbf{W}_2 \geq \mathbf{0}^{n\times c}, \end{aligned} \tag{6}$$

where S'_{ij} represents the correlation coefficient between the i-th and j-th instances, which is still calculated using the Pearson correlation function, y_{ik} represents the k-th label-independent prediction of the i-th instance, and λ_3 is a new regularization parameter.

Eq. (6) can be derived as follows:

$$\begin{aligned}\min_{\mathbf{W}_1,\mathbf{W}_2} \ & \|\mathbf{X}\mathbf{W}_1\mathbf{W}_2-\mathbf{D}\|_F^2+\lambda_1\|\mathbf{W}_1\|_{2,1}\\ &+\lambda_2 tr(\widehat{\mathbf{D}}(\mathbf{V}-\mathbf{S})\widehat{\mathbf{D}}^{\mathrm{T}})\\ &+\lambda_3 tr(\mathbf{Y}^{\mathrm{T}}(\mathbf{V}'-\mathbf{S}')\mathbf{Y})\\ \text{s.t.}\quad & \mathbf{X}\mathbf{W}_1\mathbf{W}_2\times\mathbf{1}^{c\times 1}=\mathbf{1}^{n\times 1}\\ & \mathbf{X}\mathbf{W}_1\mathbf{W}_2\geq\mathbf{0}^{n\times c},\end{aligned} \tag{7}$$

where $\mathbf{S}=[S_{pq}]_{c\times c}\in[-1,1]^{c\times c}$ denotes the correlation matrix of the labels, and $\mathbf{V}=diag(\mathbf{S}\times\mathbf{1}^{c\times 1})\in\mathbb{R}^{c\times c}$ represents the diagonal matrix of the column vector transformation formed by row accumulation of $\mathbf{S}$. Similarly, $\mathbf{S}'=[S'_{ij}]_{n\times n}\in[-1,1]^{n\times n}$ and $\mathbf{V}'=diag(\mathbf{S}'\times\mathbf{1}^{n\times 1})\in\mathbb{R}^{n\times n}$ represent the correlation and diagonal matrices obtained by exploring instance correlations, respectively.

Let $\mathbf{L}=\mathbf{V}-\mathbf{S}$, and $\mathbf{L}'=\mathbf{V}'-\mathbf{S}'$; thus, the final objective function is

$$\begin{aligned}\min_{\mathbf{W}_1,\mathbf{W}_2} \ & \|\mathbf{X}\mathbf{W}_1\mathbf{W}_2-\mathbf{D}\|_F^2+\lambda_1\|\mathbf{W}_1\|_{2,1}\\ &+\lambda_2 tr((\mathbf{X}\mathbf{W}_1\mathbf{W}_2)\mathbf{L}(\mathbf{X}\mathbf{W}_1\mathbf{W}_2)^{\mathrm{T}})\\ &+\lambda_3 tr((\mathbf{X}\mathbf{W}_1)^{\mathrm{T}}\mathbf{L}'(\mathbf{X}\mathbf{W}_1))\\ \text{s.t.}\quad & \mathbf{X}\mathbf{W}_1\mathbf{W}_2\times\mathbf{1}^{c\times 1}=\mathbf{1}^{n\times 1}\\ & \mathbf{X}\mathbf{W}_1\mathbf{W}_2\geq\mathbf{0}^{n\times c}.\end{aligned} \tag{8}$$

Algorithm 1: The TSLIP algorithm

Input: The feature matrix $\mathbf{X}$ and the distribution matrix $\mathbf{D}$
Output: The coefficient matrix $\mathbf{W}_1$ and $\mathbf{W}_2$
Initialization: $\mathbf{W}^0,\mathbf{W}_1^0,\mathbf{W}_2^0,\mathbf{W}_3^0,\mathbf{\Gamma}_1^0,\mathbf{\Gamma}_2^0,\mathbf{\Gamma}_3^0,\lambda_1,\lambda_2,\lambda_3,\rho,t=1$;
Calculate the label correlation matrix $\mathbf{S}$;
Calculate the instance correlation matrix $\mathbf{S}'$;
Compute diagonal matrix $\mathbf{V}$ and matrix $\mathbf{L}$ based on $\mathbf{S}$;
Compute the diagonal matrix $\mathbf{V}'$ and matrix $\mathbf{L}'$ based on $\mathbf{S}'$;
while *stop condition not met* **do**
    update $\mathbf{W}^{t+1}$ using Eq. (18) and Eq. (24) using BFGS;
    update $\mathbf{W}_1^{t+1}$ using Eq. (19) and Eq. (25) using BFGS;
    update $\mathbf{W}_2^{t+1}$ using Eq. (22);
    update $\mathbf{W}_3^{t+1}$ using Eq. (23);
    update $\mathbf{\Gamma}_1^{t+1}$ using Eq. (15);
    update $\mathbf{\Gamma}_2^{t+1}$ using Eq. (16);
    update $\mathbf{\Gamma}_3^{t+1}$ using Eq. (17);
    $t=t+1$;
end

3.3. Optimization

The alternating direction method of multipliers (ADMM) [43] has received increasing attention owing to its suitability for solving objective functions with multiple unknown coefficient matrices. In this study, ADMM was used to solve the objective function.

First, Eq. (8) was transformed into the following equivalent form.

$$\begin{aligned}\min_{\mathbf{W},\mathbf{W}_1,\mathbf{W}_2,\mathbf{W}_3} \ & \|\mathbf{X}\mathbf{W}-\mathbf{D}\|_F^2+\lambda_1\|\mathbf{W}_3\|_{2,1}\\ &+\lambda_2 tr((\mathbf{X}\mathbf{W})\mathbf{L}(\mathbf{X}\mathbf{W})^{\mathrm{T}})\\ &+\lambda_3 tr((\mathbf{X}\mathbf{W}_1)^{\mathrm{T}}\mathbf{L}'(\mathbf{X}\mathbf{W}_1))\\ \text{s.t.}\quad & \mathbf{W}=\mathbf{W}_1\mathbf{W}_2\\ & \mathbf{W}_1=\mathbf{W}_3\\ & \mathbf{X}\mathbf{W}\times\mathbf{1}^{c\times 1}=\mathbf{1}^{n\times 1}\\ & \mathbf{X}\mathbf{W}\geq\mathbf{0}^{n\times c}.\end{aligned} \tag{9}$$

The augmented Lagrangian for Eq. (9) is:

$$\begin{aligned}&L_\rho(\mathbf{W},\mathbf{W}_1,\mathbf{W}_2,\mathbf{W}_3,\mathbf{\Gamma}_1,\mathbf{\Gamma}_2,\mathbf{\Gamma}_3)\\ &=\|\mathbf{X}\mathbf{W}-\mathbf{D}\|_F^2+\lambda_1\|\mathbf{W}_3\|_{2,1}\\ &+\lambda_2 tr((\mathbf{X}\mathbf{W})\mathbf{L}(\mathbf{X}\mathbf{W})^{\mathrm{T}})+\lambda_3 tr((\mathbf{X}\mathbf{W}_1)^{\mathrm{T}}\mathbf{L}'(\mathbf{X}\mathbf{W}_1))\\ &+\frac{\rho}{2}\|\mathbf{W}^{t+1}-\mathbf{W}_1^t\mathbf{W}_2^t\|_F^2+\langle\mathbf{\Gamma}_1^t,\mathbf{W}^{t+1}-\mathbf{W}_1^t\mathbf{W}_2^t\rangle\\ &+\frac{\rho}{2}\|\mathbf{W}_1^t-\mathbf{W}_3^t\|_F^2+\langle\mathbf{\Gamma}_2^t,\mathbf{W}_1^t-\mathbf{W}_3^t\rangle\\ &+\frac{\rho}{2}\|\mathbf{X}\mathbf{W}^t\times\mathbf{1}^{c\times 1}-\mathbf{1}^{n\times 1}\|_F^2\\ &+\langle\mathbf{\Gamma}_3,\mathbf{X}\mathbf{W}^t\times\mathbf{1}^{c\times 1}-\mathbf{1}^{n\times 1}\rangle,\end{aligned} \tag{10}$$

where ρ is the penalty factor and $\langle\cdot,\cdot\rangle$ is the Frobenius dot product. $\mathbf{\Gamma}_1\in\mathbb{R}^{m\times c}$, $\mathbf{\Gamma}_2\in\mathbb{R}^{m\times c}$ and $\mathbf{\Gamma}_3\in\mathbb{R}^{n\times 1}$ are the Lagrange multipliers.

Eq. (10) can be solved using the following equations.

$$\begin{aligned}\mathbf{W}^{t+1}=&\arg\min_{\mathbf{W}^t}\|\mathbf{X}\mathbf{W}^t-\mathbf{D}\|_F^2+\lambda_2 tr[(\mathbf{X}\mathbf{W}^t)\mathbf{L}(\mathbf{X}\mathbf{W}^t)^{\mathrm{T}}]\\ &+\frac{\rho}{2}\|\mathbf{W}^t-\mathbf{W}_1^t\mathbf{W}_2^t\|_F^2+\langle\mathbf{\Gamma}_1^t,\mathbf{W}^t-\mathbf{W}_1^t\mathbf{W}_2^t\rangle\\ &+\frac{\rho}{2}\|\mathbf{X}\mathbf{W}^t\times\mathbf{1}^{c\times 1}-\mathbf{1}^{n\times 1}\|_F^2\\ &+\langle\mathbf{\Gamma}_3,\mathbf{X}\mathbf{W}^t\times\mathbf{1}^{c\times 1}-\mathbf{1}^{n\times 1}\rangle,\end{aligned} \tag{11}$$

$$\begin{aligned}\mathbf{W}_1^{t+1}=&\arg\min_{\mathbf{W}_1^t}\lambda_3 tr[(\mathbf{X}\mathbf{W}_1^t)^{\mathrm{T}}\mathbf{L}'(\mathbf{X}\mathbf{W}_1^t)]\\ &+\frac{\rho}{2}\|\mathbf{W}^{t+1}-\mathbf{W}_1^t\mathbf{W}_2^t\|_F^2+\langle\mathbf{\Gamma}_1^t,\mathbf{W}^{t+1}-\mathbf{W}_1^t\mathbf{W}_2^t\rangle\\ &+\frac{\rho}{2}\|\mathbf{W}_1^t-\mathbf{W}_3^t\|_F^2+\langle\mathbf{\Gamma}_2^t,\mathbf{W}_1^t-\mathbf{W}_3^t\rangle,\end{aligned} \tag{12}$$

$$\begin{aligned}\mathbf{W}_2^{t+1}=&\arg\min_{\mathbf{W}_2^t}\frac{\rho}{2}\|\mathbf{W}^{t+1}-\mathbf{W}_1^{t+1}\mathbf{W}_2^t\|_F^2\\ &+\langle\mathbf{\Gamma}_1^t,\mathbf{W}^{t+1}-\mathbf{W}_1^{t+1}\mathbf{W}_2^t\rangle,\end{aligned} \tag{13}$$

$$\begin{aligned}\mathbf{W}_3^{t+1}=&\arg\min_{\mathbf{W}_3^t}\lambda_1\|\mathbf{W}_3^t\|_{2,1}+\frac{\rho}{2}\|\mathbf{W}_1^{t+1}-\mathbf{W}_3^t\|_F^2\\ &+\langle\mathbf{\Gamma}_2^t,\mathbf{W}_1^{t+1}-\mathbf{W}_3^t\rangle,\end{aligned} \tag{14}$$

$$\mathbf{\Gamma}_1^{t+1}=\mathbf{\Gamma}_1^t+\rho(\mathbf{W}^{t+1}-\mathbf{W}_1^{t+1}\mathbf{W}_2^{t+1}), \tag{15}$$

$$\mathbf{\Gamma}_2^{t+1}=\mathbf{\Gamma}_2^t+\rho(\mathbf{W}_1^{t+1}-\mathbf{W}_3^{t+1}), \tag{16}$$

$$\mathbf{\Gamma}_3^{t+1}=\mathbf{\Gamma}_3^t+\rho(\mathbf{X}\mathbf{W}^{t+1}\times\mathbf{1}^{c\times 1}-\mathbf{1}^{n\times 1}). \tag{17}$$

Eqs. (15)– (17) can be directly calculated in iterations. Eqs. (11)– (14) can be simplified to the following form using the quadratic term absorption principle [44].

$$\begin{aligned}\mathbf{W}^{t+1} =& \arg\min_{\mathbf{W}^t} \|\mathbf{X}\mathbf{W}^t - \mathbf{D}\|_F^2 + \lambda_2 tr[(\mathbf{X}\mathbf{W}^t)\mathbf{L}(\mathbf{X}\mathbf{W}^t)^{\mathrm{T}}] \\ &+ \frac{\rho}{2}\|\mathbf{W}^t - \mathbf{W}_1^t\mathbf{W}_2^t + \frac{1}{\rho}\mathbf{\Gamma}_1^t\|_F^2 \\ &+ \frac{\rho}{2}\|\mathbf{X}\mathbf{W}^t \times \mathbf{1}^{c\times 1} - \mathbf{1}^{n\times 1} + \frac{1}{\rho}\mathbf{\Gamma}_3^t\|_F^2, \end{aligned} \tag{18}$$

$$\begin{aligned}\mathbf{W}_1^{t+1} =& \arg\min_{\mathbf{W}_1^t} \lambda_3 tr[(\mathbf{X}\mathbf{W}_1^t)^{\mathrm{T}}\mathbf{L}'(\mathbf{X}\mathbf{W}_1^t)] \\ &+ \frac{\rho}{2}\|\mathbf{W}^{t+1} - \mathbf{W}_1^t\mathbf{W}_2^t + \frac{1}{\rho}\mathbf{\Gamma}_1^t\|_F^2 \\ &+ \frac{\rho}{2}\|\mathbf{W}_1^t - \mathbf{W}_3^t + \frac{1}{\rho}\mathbf{\Gamma}_2^t\|_F^2, \end{aligned} \tag{19}$$

$$\mathbf{W}_2^{t+1} = \arg\min_{\mathbf{W}_2^t} \frac{\rho}{2}\|\mathbf{W}^{t+1} - \mathbf{W}_1^{t+1}\mathbf{W}_2^t + \frac{1}{\rho}\mathbf{\Gamma}_1^t\|_F^2, \tag{20}$$

$$\mathbf{W}_3^{t+1} = \arg\min_{\mathbf{W}_3^t} \lambda_1\|\mathbf{W}_3^t\|_{2,1} + \frac{\rho}{2}\|\mathbf{W}_3^t - (\mathbf{W}_1^{t+1} + \frac{1}{\rho}\mathbf{\Gamma}_2^t)\|_F^2. \tag{21}$$

For Eq. (20), let its derivative be 0. Thus, its iterative solution is:

$$\mathbf{W}_2^{t+1} = (\mathbf{W}_1^{t+1\mathrm{T}}\mathbf{W}_1^{t+1})^{-1}(\mathbf{W}_1^{t+1\mathrm{T}}\mathbf{W}^{t+1} + \frac{1}{\rho}\mathbf{W}_1^{t+1\mathrm{T}}\mathbf{\Gamma}_1^t). \tag{22}$$

For Eq. (21), the method proposed in [45] can be used to solve it. The optimal formulation can be expressed as:

$$\mathbf{W}_3^{t+1}(:,i) = \begin{cases} \frac{\|q_i\| - \frac{\lambda_1}{\rho}}{\|q_i\|} q_i, & \text{if } \frac{\lambda_1}{\rho} \le \|q_i\|, \\ 0, & \text{otherwise,} \end{cases} \tag{23}$$

where $\mathbf{W}_3^{t+1}(:,i)$ denotes the i-th column of data in $\mathbf{W}_3^{t+1}$, q_i is the i-th column of data in matrix $Q = \mathbf{W}_1^{t+1} + \frac{1}{\rho}\mathbf{\Gamma}_2^t$.

Herein, the quasi-Newton method (BFGS) [46, 47] was used to solve Eqs. (18) and (19). For optimization, BFGS is related primarily to the first-order gradient, which can be obtained using:

$$\begin{aligned}\nabla\mathbf{W}^{t+1} =& 2\mathbf{X}^{\mathrm{T}}(\mathbf{X}\mathbf{W}^t - \mathbf{D}) + \lambda_2\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{W}^t(\mathbf{L}^{\mathrm{T}} + \mathbf{L}) \\ &+ \rho(\mathbf{W}^t - \mathbf{W}_1^t\mathbf{W}_2^t + \frac{1}{\rho}\mathbf{\Gamma}_1^t) \\ &+ \rho\mathbf{X}^{\mathrm{T}}(\mathbf{X}\mathbf{W}^t \times \mathbf{1}^{c\times 1} - \mathbf{1}^{n\times 1} + \frac{1}{\rho}\mathbf{\Gamma}_3)\mathbf{1}^{1\times c}, \end{aligned} \tag{24}$$

$$\begin{aligned}\nabla\mathbf{W}_1^{t+1} =& \lambda_3\mathbf{X}^{\mathrm{T}}(\mathbf{L}' + \mathbf{L}'^{\mathrm{T}})\mathbf{X}\mathbf{W}_1^t \\ &- \rho(\mathbf{W}^{t+1} - \mathbf{W}_1^t\mathbf{W}_2^t + \frac{1}{\rho}\mathbf{\Gamma}_1^t)\mathbf{W}_2^{\mathrm{T}} \\ &+ \rho(\mathbf{W}_1^t - \mathbf{W}_3^t + \frac{1}{\rho}\mathbf{\Gamma}_2^t). \end{aligned} \tag{25}$$

Algorithm 1 shows the overall process of the proposed TSLIP algorithm.

Table 2: Computational complexity of Algorithm 1

Lines	Complexity	Description
Line 1	$O(1)$	Initialize the parameters
Line 2	$O(cn)$	Calculate the label correlation matrix
Line 3	$O(nm)$	Calculate the instance correlation matrix
Line 4	$O(c^2)$	Compute matrix $\mathbf{V}$ and matrix $\mathbf{L}$
Line 5	$O(n^2)$	Compute matrix $\mathbf{V}'$ and matrix $\mathbf{L}'$
Lines 6-15	$O(\tau_1\tau_2 cn^2)$	Search the optimum values
Total	$O(1) + O(cn) + O(nm) + O(c^2) + O(n^2) + O(\tau_1\tau_2 cn^2)$ $= O(\tau_1\tau_2 cn^2)$	

Table 3: Raw medical data of several personnel

Personnel	Height (m)	SBP (mmHg)	TC (mmol/L)	LDC (mmol/L)	Weight (kg)	DP (mmHg)	Triglyceride (mmol/L)
Person 1	1.5	144	6.0	3.5	72	93	3.0
Person 2	1.5	120	1.2	3.0	72	85	2.0
Person 3	1.7	160	6.8	2.7	65	100	3.8

3.4. Time complexity analysis

This section elaborates on the time complexity of Algorithm 1 as follows.

Proposition 1. *The time complexity of the proposed algorithm is $O(\tau_1\tau_2 cn^2)$.*

Proof 1. *Suppose that our algorithm iterates τ_1 times, and the BFGS algorithm iterates τ_2 times. Note that the number of instances n is significantly larger than the number of features m and labels c. That is, $m << n$ and $c << n$. These conditions are considered prerequisites for eliminating small terms during the analysis process. Therefore, Line 1 initializes the parameters in $O(1)$. Line 2 calculates the label correlation matrix $\mathbf{S}$ in $O(cn)$. Line 3 calculates the instance correlation matrix $\mathbf{S}'$ in $O(nm)$. Line 4 computes diagonal matrix $\mathbf{V}$ and matrix $\mathbf{L}$ based on $\mathbf{S}$ in $O(c^2)$. Line 5 computes the diagonal matrix $\mathbf{V}'$ and matrix $\mathbf{L}'$ based on $\mathbf{S}'$ with $O(n^2)$. Lines 6 through 15 search for the optimum values in $O(\tau_1\tau_2 cn^2)$.*

The time complexity of Algorithm 1 is:

$$\begin{aligned} &O(1) + O(cn) + O(nm) + O(c^2) + O(n^2) + O(\tau_1\tau_2 cn^2) \\ &= O(\tau_1\tau_2 cn^2). \end{aligned} \tag{26}$$

This completes the proof. Table 2 describes the complexity of Algorithm 1 briefly.

3.5. Running example

Let us assume that a medical examination focuses primarily on obesity, high blood pressure, and high blood lipids. The

Table 4: Normalized medical data

Personnel	Height	SBP	TC	LDC	Weight	DP	Triglyceride
Person 1	0.75	0.72	0.60	0.35	0.72	0.465	0.30
Person 2	0.75	0.60	0.12	0.30	0.72	0.425	0.20
Person 3	0.85	0.80	0.68	0.27	0.65	0.500	0.38

Table 5: Label-independent prediction of three health problems

Personnel	Obesity	High Blood Pressure	High Blood Lipids
Person 1	0.852	0.636	0.640
Person 2	0.852	0.467	0.470
Person 3	0.620	0.735	0.880

Table 6: Label distribution of three health problems

Personnel	Obesity	High Blood Pressure	High Blood Lipids
Person 1	0.65	0.175	0.175
Person 2	0.90	0.050	0.050
Person 3	0.15	0.410	0.440

medical data of three persons are presented in Table 3, where SBP, TC, LDC, and DP represent the systolic blood pressure, total cholesterol, low-density cholesterol, and diastolic pressure, respectively. Table 4 presents the normalized data.

According to medical understanding, the two most important factors for evaluating whether a person is obese are their height and weight. Only systolic and diastolic blood pressure should be considered when evaluating a person's blood pressure. High blood lipid levels are correlated predominantly with levels of triglycerides, low-density cholesterol, and total cholesterol. Thus, for Person 1, the following conclusions can be drawn.

1) Height is 1.5 meters, weight is 72 kg, and BMI is 32, which is beyond the acceptable range of 18–24; hence, Person 1 is significantly obese.

2) Systolic/diastolic blood pressure is 144/93, which is slightly higher than the normal range of 140/90 and indicates moderate hypertension.

3) The cholesterol level is 6/3.5/3, which is a little higher than the usual level and indicates moderate hyperlipidemia.

The same analysis can also be performed for Persons 2 and 3.

Considering the severity of the three health problems independently and constraining them in the range [0,1], the health data are presented in Table 5. The degree of label representation in the domain can be regarded as a label-independent prediction. A label-independent prediction is based on label-specific features. If two instances have the same value for a particular label feature, their label-independent predictions for that label should also be the same.

Label-independent predictions are merely individual evaluations of a person's three health issues. When evaluating someone's health, all of their medical issues must be listed and a comprehensive evaluation must be provided. Thus, in our combined evaluation, the problem is marked as having a different severity if two persons have the same severity for one health issue but a different severity for the other. Therefore, a label-independent prediction should be transformed into a label distribution. Table 6 lists the transformed label distribution data.

Table 7: Characteristics of all 12 datasets

No.	Dataset	Instances	Features	Labels
1	s-JAFFE	213	243	6
2	M2B	1,240	250	5
3	Emotion6	1,980	168	7
4	Movie	7,755	1,869	5
5	Twitter_LDL	10,045	168	8
6	Flickr_LDL	11,150	168	8
7	Natural_Scene	2,000	294	9
8	FBP5500	5,500	512	5
9	Yeast_cdc	2,465	24	15
10	Yeast_cold	2,465	24	4
11	Yeast_spoem	2,465	24	2
12	Human_Gene	30,542	36	68

4. Experiments

This section presents the experiments conducted to analyze the effectiveness of the TSLIP algorithm.

4.1. Datasets

Herein, 12 real-world datasets from different domains were used for the experiments. Table 7 lists the statistics for all the datasets.

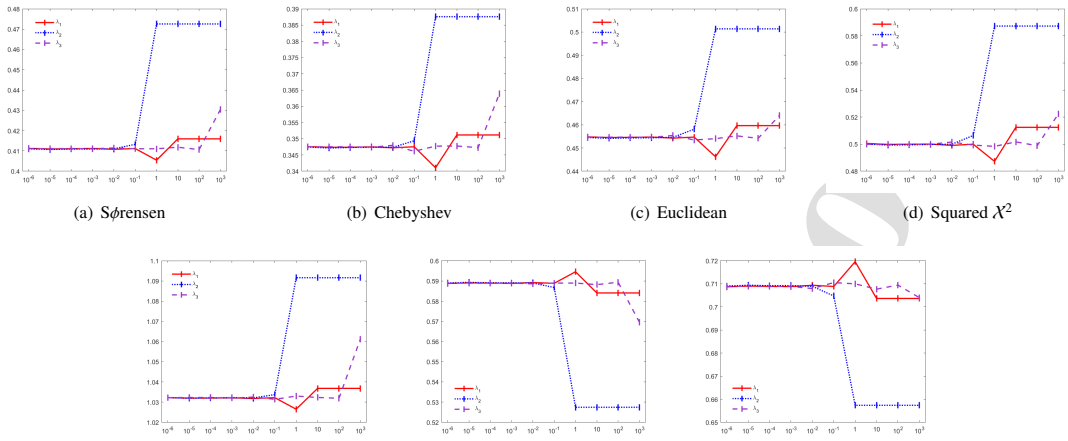
s-JAFFE [10] is an extension of the well-known facial expression image dataset, JAFFE [48]. In particular, each image was scored by 60 individuals on six basic emotion labels (happiness, sadness, surprise, fear, anger, and disgust) based on emotional conformity. The average score for each emotion was used to represent emotional intensity.

M2B [49] was processed by Ren et al. [16] based on the original dataset [50] for facial beauty perception. The label distribution was transformed using a k-wise model [49].

Emotion6 [51] contains 1,980 images assembled from Flickr for a sentiment prediction benchmark, which is annotated with votes for seven emotional categories (i.e., anger, disgust, joy, fear, sadness, surprise, and neutral). Herein, a 168-dimensional feature vector was extracted for each image using the method proposed by [16].

Movie [10] is a dataset that counts user ratings of movies. It uses 54,242,292 ratings of 7,755 movies from 478,656 users as the data source. The ratings come from Netflix and are on a scale of one–five (five labels). The rating label distribution was calculated for each movie as a percentage of the rating level. The features of the movie are extracted from metadata such as genre, director, actor, country, budget, etc. The feature vector of each movie is 1, 869-dimensional.

Flickr LDL and Twitter LDL [52] contain 11,150 and 10,045 images, respectively, whose labels fall within the typical eight-emotional space (anger, amusement, awe, contentment, disgust, excitement, fear, and sadness). Following the approach proposed in [16], feature vector extraction was performed on the two datasets using HOG[53] and Color Moment[54]. Simultaneously, PCA was used to reduce the dimension of high-dimensional features to 168 dimensions.



(a) Sørensen (b) Chebyshev (c) Euclidean (d) Squared χ^2

(e) KL (f) Intersection (g) Cosine

Figure 4: Influence of λ_1, λ_2, and λ_3 with seven metrics on dataset M2B

Table 8: Evaluation metrics for LDL algorithms

	Name	Formula
Distance	Sørensen ↓	$Dis_1(d_i,\widehat{d}_i)=\dfrac{\sum_{j=1}^{c}\lvert d_{ij}-\widehat{d}_{ij}\rvert}{\sum_{j=1}^{c}\lvert d_{ij}+\widehat{d}_{ij}\rvert}$
	Chebyshev ↓	$Dis_2(d_i,\widehat{d}_i)=\max_{j=1}^{c}(d_{ij}-\widehat{d}_{ij})$
	Euclidean ↓	$Dis_3(d_i,\widehat{d}_i)=\sqrt{\sum_{j=1}^{c}(d_{ij}-\widehat{d}_{ij})^2}$
	Squared χ^2 ↓	$Dis_4(d_i,\widehat{d}_i)=\sum_{j=1}^{c}\dfrac{(d_{ij}-\widehat{d}_{ij})^2}{d_{ij}+\widehat{d}_{ij}}$
	Kullback-Leibler(KL) ↓	$Dis_5(d_i,\widehat{d}_i)=\sum_{j=1}^{c}d_{ij}\ln\dfrac{d_{ij}}{\widehat{d}_{ij}}$
Similarity	Intersection ↑	$Sim_1(d_i,\widehat{d}_i)=\sum_{j=1}^{c}\min(d_{ij},\widehat{d}_{ij})$
	Cosine ↑	$Sim_2(d_i,\widehat{d}_i)=\dfrac{\sum_{j=1}^{c}d_{ij}\widehat{d}_{ij}}{\sqrt{\sum_{j=1}^{c}d_{ij}^2}\sqrt{\sum_{j=1}^{c}\widehat{d}_{ij}^2}}$

Finally, a Natural Scene [10] resulted from the consistent multilabel rankings of 2,000 natural scene images. FBP5500 [55] was obtained from the facial beauty perception of 5,500 color face photos. The next three datasets, Yeast_cdc, Yeast_cold, and Yeast_spoem [48], were obtained from biological experiments with yeast. The data of Human_Gene [10] were obtained from biological investigations of the connection between disease and human genes.

4.2. Evaluation metrics

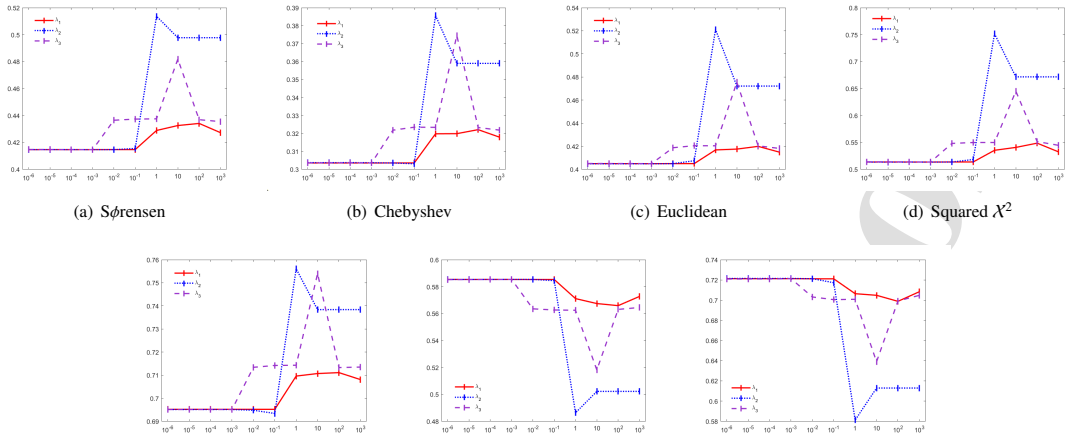
To evaluate the performance of the proposed algorithm, seven widely used evaluation metrics [10, 23, 24], including five distance-based metrics (Sørensen, Chebyshev, Euclidean, Squared χ^2, Kullback–Leibler(KL)), and two similarity-based metrics (Intersection and Cosine), were employed. Let $d_i = [d_{i1}, d_{i2}, \ldots, d_{ic}]$ denote the true label distribution for the ith instance and $\widehat{d}_i = [\widehat{d}_{i1}, \widehat{d}_{i2}, \ldots, \widehat{d}_{ic}]$ denotes the predicted distribution value for the same instance. Table 8 lists the formulae of these seven evaluation indicators. For distance-based metrics, ↓ indicates that the smaller the value, the better the algorithm's effects. For similarity-based metrics, ↑ indicates that the larger the value, the better the effects of the algorithm.

4.3. Experiment settings

The proposed algorithm was compared with seven state-of-the-art algorithms, including PT-Bayes, SA-BFGS, LDL-HR [56], LDL-LDM [26], LDL-SF [16], EDL [57], and Adam-LDL-SCL [58]. PT-Bayes uses maximum likelihood estimation to estimate gaussian-like conditional probability density functions. For the remaining algorithms, the suggested parameters reported in the relevant literature were used. In particular, for SA-BFGS, the parameters of the strong Wolfe conditions were set to $c_1 = 10^{-4}$ and $c_2 = 0.9$. For LDL-HR, $\lambda_1 = 0.001$, λ_2, and λ_3 were tuned from the candidate set $\{10^{-3}, \ldots, 1\}$, and $\rho = 0.01$. For LDL-LDM, λ_1, λ_2, and λ_3 are tuned from $\{10^{-3}, \ldots, 10^{3}\}$, and g was selected from 1 to 14. For LDL-SF, the parameters λ_1, λ_2, and λ_3 were selected from $10^{\{-6,-5,\ldots,-2,-1\}}$, respectively, and ρ was set to 10^{-3}. For EDL, the ξ, ξ_1, and ξ_2 were set as 0.25, 0.0001, and 0.1, respectively. For Adam-LDL-SCL, the parameters λ_1, λ_2 and λ_3 were tuned from candidate set $10^{\{-3,-2,\ldots,2,3\}}$, and the number of clusters obtained by k-means was tuned from 0 to 14. For TSLIP, the parameters λ_1, λ_2 and λ_3 were selected from the same candidate set $10^{\{-6,-5,\ldots,1,2,3\}}$, and ρ was set to 10^{-2}.

Table 9: Comparison results on all datasets, shown as "mean±std." ↑(↓) indicates the higher (lower), the better. The best results on each row are highlighted. "AVG.Rank" is the average ranking of each algorithm on the corresponding evaluation metric

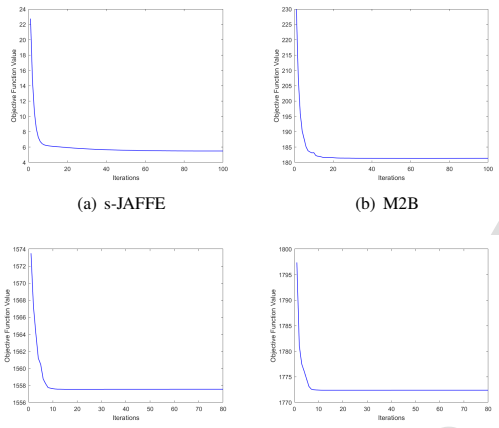
Metric	Dataset	PT-Bayes	SA-BFGS	LDL-HR	LDL-LDM	LDL-SF	EDL	Adam-LDL-SCL	TSLIP (our)
	s-JAFFE	.1547±.0025	.1236±.0009	.1374±.0074	.1487±.0047	.1157±.0009	.1589±.0081	.6919±.2750	**.1134±.0011**
Sørensen ↓	M2B	.7099±.0005	.4442±.0026	.4656±.0085	.4326±.0042	.4327±.0022	.4797±.0060	.4400±.0058	**.4175±.0065**
	Emotion6	.6031±.0048	.4279±.0008	.4448±.0070	.4273±.0029	.4152±.0034	.4740±.0052	.4210±.0041	**.4149±.0004**
	Movie	.2800±.0008	.1826±.0011	.2294±.0024	.1725±.0004	.1882±.0004	.2550±.0023	.1756±.0031	**.1679±.0027**
	Twitter_LDL	.5436±.0063	.3812±.0020	.6425±.0027	.4843±.0018	.3772±.0004	.6608±.0011	.3961±.0029	**.3747±.0038**
	Flickr_LDL	.6054±.0031	.4083±.0001	.5732±.0021	.4598±.0004	.4073±.0001	.5921±.0011	.4224±.0022	**.4060±.0045**
	Natural_Scene	.6517±.0008	.4631±.0030	.5059±.0171	.5021±.0077	.4603±.0006	.6373±.0087	.5245±.0055	**.4588±.0105**
	FBP5500	.4985±.0003	.2010±.0018	.3893±.0168	.1783±.0021	.1632±.0005	.4982±.0051	**.1576±.0037**	.1636±.0031
	Yeast_cdc	.2357±.0035	.0429±.0004	.0476±.0008	.0432±.0003	.0427±.0002	.0429±.0008	**.0421±.0005**	.0425±.0011
	Yeast_cold	.2113±.0035	.0595±.0009	.0598±.0009	.0596±.0003	.0593±.0002	.0663±.0020	.0593±.0013	**.0590±.0017**
	Yeast_spoem	.1795±.0109	.0872±.0004	.0873±.0037	.0871±.0014	.0869±.0002	.0916±.0033	.0894±.0035	**.0868±.0032**
	Human_Gene	.5330±.0121	.2160±.0003	.2439±.0035	.2154±.0000	.2156±.0001	.2172±.0015	.2163±.0013	**.2146±.0048**
	AVG.Rank	7.7	3.9	5.9	3.9	2.5	6.8	4.0	1.2
	s-JAFFE	.1208±.0031	.0923±.0001	.1068±.0099	.1182±.0035	.0875±.0014	.9226±.0488	4.7385±1.9567	**.0865±.0010**
Chebyshev ↓	M2B	.7063±.0004	.3878±.0011	.3840±.0093	.3617±.0052	.3608±.0015	2.7408±.0364	2.6191±.0299	**.3482±.0007**
	Emotion6	.5278±.0075	.3168±.0036	.3295±.0092	.3188±.0053	.3053±.0001	3.8232±.0362	3.7315±.0361	**.3042±.0002**
	Movie	.2030±.0012	.1311±.0009	.1604±.0026	.1209±.0005	.1317±.0001	1.4102±.0148	1.0672±.0174	**.1175±.0017**
	Twitter_LDL	.5207±.0068	.3004±.0021	.5276±.0034	.3875±.0024	.2984±.0006	6.2116±.0104	6.2511±.0129	**.2969±.0046**
	Flickr_LDL	.5622±.0037	.3065±.0010	.4358±.0041	.3514±.0009	.3053±.0001	5.5473±.0111	5.4434±.0147	**.3046±.0046**
	Natural_Scene	.4092±.0013	.3381±.0011	.4427±.0200	.3276±.0070	.3208±.0007	7.0062±.0662	6.8069±.0490	**.3188±.0145**
	FBP5500	.4082±.0003	.1764±.0017	.3887±.0171	.1585±.0013	**.1452±.0004**	2.9587±.0128	2.1896±.0184	.1453±.0033
	Yeast_cdc	.1143±.0026	.0163±.0001	.0181±.0003	.0164±.0002	**.0162±.0002**	.6518±.0135	.6405±.0081	.0162±.0006
	Yeast_cold	.1905±.0035	.0513±.0006	.0515±.0009	.0515±.0001	.0511±.0002	.2688±.0083	.2407±.0056	**.0509±.0017**
	Yeast_spoem	.1795±.0109	.0872±.0004	.0873±.0037	.0871±.0014	.0869±.0002	.1888±.0070	.1851±.0077	**.0868±.0032**
	Human_Gene	.2104±.0111	.0534±.0002	.0536±.0027	.0533±.0000	.0533±.0000	14.5609±.0963	14.4879±.0837	**.0529±.0030**
	AVG.Rank	5.8	3.5	5.0	3.4	2.0	7.8	7.2	1.2
	s-JAFFE	.1560±.0029	.1251±.0003	.1388±.0090	.1510±.0043	.1168±.0012	.1607±.0074	.7643±.3095	**.1150±.0011**
Euclidean ↓	M2B	.8912±.0010	.5074±.0012	.4892±.0089	.4604±.0050	.4753±.0022	.5011±.0049	.4672±.0059	**.4593±.0047**
	Emotion6	.6555±.0067	.4186±.0012	.4279±.0086	.4117±.0043	.4043±.0015	.4504±.0065	.4101±.0051	**.4043±.0002**
	Movie	.2887±.0012	.1875±.0012	.2334±.0026	.1747±.0004	.1912±.0003	.2585±.0022	.1784±.0033	**.1705±.0023**
	Twitter_LDL	.6704±.0083	.3823±.0025	.6226±.0029	.4706±.0021	.3783±.0005	.6412±.0012	.3903±.0032	**.3772±.0048**
	Flickr_LDL	.7101±.0042	.3947±.0004	.5339±.0033	.4373±.0006	.3940±.0001	.5507±.0012	.4052±.0024	**.3933±.0050**
	Natural_Scene	.5658±.0013	.4495±.0029	.5603±.0202	.4428±.0082	.4340±.0007	.5251±.0131	.4691±.0077	**.4303±.0132**
	FBP5500	.5304±.0004	.2274±.0028	.4945±.0219	.2058±.0019	.1895±.0005	.5166±.0046	**.1846±.0044**	.1897±.0038
	Yeast_cdc	.1652±.0028	.0281±.0002	.0310±.0005	.0282±.0002	.0279±.0002	.0283±.0006	**.0276±.0004**	.0278±.0008
	Yeast_cold	.2444±.0042	.0685±.0010	.0688±.0010	.0686±.0003	.0683±.0002	.0766±.0023	.0684±.0015	**.0680±.0021**
	Yeast_spoem	.2538±.0155	.1234±.0006	.1235±.0053	.1232±.0019	.1229±.0001	.1295±.0047	.1265±.0050	**.1228±.0045**
	Human_Gene	.2774±.0109	.0866±.0002	.0941±.0028	.0863±.0000	.0873±.0001	.0868±.0012	.0868±.0011	**.0859±.0033**
	AVG.Rank	7.8	4.1	5.9	3.8	2.8	6.6	3.8	1.2
	s-JAFFE	.0708±.0025	.0509±.0006	.0577±.0063	.0664±.0027	.0455±.0007	.0738±.0054	1.1600±.5481	**.0431±.0007**
Squared χ^2 ↓	M2B	1.1917±.0034	.5676±.0047	.5552±.0159	**.5001±.0072**	.5394±.0043	.5815±.0104	.5134±.0089	.5082±.0085
	Emotion6	.9319±.0099	.5397±.0002	.5643±.0145	.5349±.0063	.5188±.0057	.6217±.0116	.5243±.0096	**.5179±.0007**
	Movie	.2355±.0011	.1168±.0010	.1598±.0030	**.1026±.0005**	.1313±.0006	.1899±.0031	.1147±.0051	.1030±.0032
	Twitter_LDL	.8837±.0161	.5162±.0037	1.0333±.0053	.6744±.0035	.5101±.0001	1.0846±.0024	.5412±.0041	**.5067±.0059**
	Flickr_LDL	.9805±.0053	.5483±.0006	.8543±.0043	.6314±.0013	.5447±.0002	.8974±.0023	.5741±.0040	**.5431±.0084**
	Natural_Scene	1.0218±.0018	.6694±.0050	.7802±.0352	.7326±.0134	.6662±.0009	.9880±.0204	.7811±.0094	**.6571±.0202**
	FBP5500	.6526±.0006	.1641±.0018	.4839±.0310	.1321±.0034	.1191±.0006	.6458±.0103	**.1085±.0042**	.1200±.0044
	Yeast_cdc	.2084±.0052	.0071±.0001	.0083±.0003	.0072±.0000	.0071±.0002	.0073±.0003	**.0069±.0002**	.0070±.0004
	Yeast_cold	.1631±.0038	.0123±.0005	.0126±.0006	.0124±.0000	.0123±.0001	.0151±.0010	.0124±.0005	**.0122±.0012**
	Yeast_spoem	.1177±.0116	.0250±.0002	.0263±.0029	.0264±.0001	**.0248±.0001**	.0274±.0021	.0262±.0019	.0248±.0021
	Human_Gene	.7535±.0256	.1843±.0007	.2125±.0063	.1836±.0000	.1864±.0002	.1865±.0029	.1848±.0024	**.1826±.0083**
	AVG.Rank	7.7	3.7	5.8	3.8	2.8	6.9	3.8	1.5
	s-JAFFE	.0744±.0026	.0525±.0004	.0591±.0071	.0676±.0027	.0506±.0010	.0784±.0057	7.8077±3.8722	**.0445±.0017**
KL ↓	M2B	12.4931±.0145	.7659±.0120	.5900±.0182	.7820±.0114	1.1540±.0395	.6204±.0115	**.5519±.0120**	1.0316±.0665
	Emotion6	3.9536±.0475	.6344±.0007	.6364±.0185	8.7785±.1085	.7578±.0203	.7065±.0154	**.5946±.0135**	.6953±.0026
	Movie	.5240±.0175	.1378±.0373	.1577±.0032	.1324±.0026	.2420±.0023	.1887±.0032	.2175±.0280	**.1243±.0045**
	Twitter_LDL	2.4935±.0128	**.6533±.0006**	1.2325±.0076	14.6794±.0708	.7112±.0004	1.3232±.0045	.6598±.0066	.6988±.0160
	Flickr_LDL	2.0693±.0130	**.6571±.0008**	.9951±.0061	13.2738±.0295	.7310±.0013	1.0645±.0032	.6737±.0055	.7276±.0160
	Natural_Scene	2.3733±.0335	.9067±.0556	7.2284±.6466	2.7787±.0527	1.1245±.0101	1.1729±.0322	**.9014±.0126**	.9671±.0572
	FBP5500	1.6059±.0179	.1594±.0105	5.0716±.7691	1.2375±.0641	.1760±.0022	.7005±.0135	**.1149±.0047**	.1786±.0121
	Yeast_cdc	.3058±.0063	.0071±.0001	.0082±.0003	.0082±.0012	.0070±.0001	.0072±.0003	.0071±.0004	**.0069±.0004**
	Yeast_cold	.2455±.0081	.0123±.0001	.0125±.0006	.0127±.0000	.0122±.0001	.0151±.0010	.0125±.0005	**.0121±.0012**
	Yeast_spoem	.1971±.0005	.0246±.0001	.0260±.0029	.0415±.0003	.0247±.0001	.0271±.0020	.0259±.0018	**.0245±.0020**
	Human_Gene	1.8418±.0379	.2367±.0010	.2636±.0107	**.2256±.0000**	.2368±.0001	.2394±.0050	.2375±.0043	.2347±.0134
	AVG.Rank	7.3	2.4	5.2	5.8	4.0	5.5	3.2	2.5
	s-JAFFE	.8453±.0025	.8764±.0009	.8626±.0074	.8513±.0047	.8843±.0009	.8411±.0081	.3081±.2750	**.8866±.0011**
Intersection ↑	M2B	.2901±.0005	.5558±.0026	.5344±.0085	.5674±.0042	.5673±.0022	.5203±.0060	.5600±.0058	**.5825±.0065**
	Emotion6	.3969±.0048	.5721±.0008	.5552±.0070	.5727±.0029	.5797±.0034	.5260±.0052	.5790±.0041	**.5851±.0004**
	Movie	.7200±.0008	.8174±.0011	.7706±.0024	.8275±.0004	.8118±.0004	.7450±.0023	.8244±.0031	**.8321±.0027**
	Twitter_LDL	.4564±.0063	.6188±.0020	.3575±.0027	.5157±.0018	.6228±.0004	.3392±.0011	.6039±.0029	**.6253±.0038**
	Flickr_LDL	.3946±.0031	.5917±.0001	.4268±.0021	.5402±.0004	.5927±.0001	.4079±.0011	.5776±.0022	**.5940±.0045**
	Natural_Scene	.3483±.0008	.5369±.0030	.4941±.0171	.4979±.0077	.5397±.0006	.3627±.0087	.4755±.0055	**.5412±.0105**
	FBP5500	.5015±.0003	.7990±.0018	.6107±.0168	.8217±.0021	.8368±.0005	.5018±.0051	**.8424±.0037**	.8364±.0031
	Yeast_cdc	.7643±.0035	.9571±.0004	.9524±.0008	.9568±.0003	.9573±.0002	.9571±.0008	**.9579±.0005**	.9575±.0011
	Yeast_cold	.7887±.0035	.9405±.0009	.9402±.0009	.9404±.0003	.9407±.0002	.9337±.0020	.9407±.0013	**.9410±.0017**
	Yeast_spoem	.8205±.0109	.9128±.0004	.9127±.0037	.9129±.0014	.9131±.0002	.9084±.0033	.9106±.0035	**.9132±.0032**
	Human_Gene	.4670±.0121	.7840±.0003	.7561±.0035	.7846±.0000	.7844±.0001	.7828±.0015	.7838±.0013	**.7854±.0048**
	AVG.Rank	7.7	3.9	5.9	3.9	2.5	6.8	4.0	1.2
	s-JAFFE	.9298±.0022	.9530±.0010	.9443±.0070	.9340±.0030	.9595±.0007	.9256±.0051	.4957±.2212	**.9605±.0006**
Cosine ↑	M2B	.4306±.0004	.6535±.0024	.6915±.0067	**.7237±.0041**	.6914±.0026	.6710±.0046	.7144±.0053	.7093±.0003
	Emotion6	.5294±.0032	.6967±.0014	.6918±.0069	.7121±.0036	.7175±.0028	.6499±.0085	.7092±.0063	**.7184±.0003**
	Movie	.8477±.0002	.9193±.0008	.8911±.0020	.9320±.0003	.9169±.0004	.8651±.0016	.9272±.0029	**.9321±.0021**
	Twitter_LDL	.5762±.0098	.8185±.0013	.5405±.0015	.7782±.0017	**.8249±.0003**	.4959±.0017	.8177±.0041	.8243±.0041
	Flickr_LDL	.5293±.0040	.7712±.0007	.5894±.0022	.7454±.0015	**.7763±.0001**	.5523±.0010	.7641±.0025	.7763±.0055
	Natural_Scene	.5574±.0008	.6966±.0061	.6218±.0178	.7266±.0044	.7231±.0009	.5750±.0082	.6845±.0091	**.7310±.0098**
	FBP5500	.6674±.0003	.9327±.0018	.8304±.0122	.9442±.0008	**.9516±.0003**	.6586±.0091	.9509±.0023	.9510±.0024
	Yeast_cdc	.8433±.0036	.9932±.0001	.9921±.0002	.9932±.0000	.9933±.0001	.9931±.0003	**.9935±.0002**	.9933±.0004
	Yeast_cold	.8870±.0024	.9885±.0005	.9882±.0004	.9883±.0001	.9885±.0001	.9858±.0009	.9884±.0005	**.9886±.0009**
	Yeast_spoem	.9208±.0070	.9788±.0002	.9779±.0019	.9779±.0002	**.9790±.0001**	.9766±.0014	.9780±.0013	.9790±.0014
	Human_Gene	.4296±.0238	.8341±.0011	.8103±.0049	.8349±.0000	.8347±.0001	.8327±.0020	.8336±.0018	**.8357±.0064**
	AVG.Rank	7.6	4.1	5.9	3.6	2.3	6.9	3.9	1.7



(a) Sørensen (b) Chebyshev (c) Euclidean (d) Squared χ^2

(e) KL (f) Intersection (g) Cosine

Figure 5: Influence of λ_1, λ_2, and λ_3 with seven metrics on dataset Emotion6



(a) s-JAFFE (b) M2B

(c) Twitter_LDL (d) Flickr_LDL

Figure 6: Convergence of TSLIP on s-JAFFE, M2B, Twitter_LDL, and Flickr_LDL.

4.4. Influence of parameters

To examine the robustness of the algorithm, the influence of the parameters in the experiment, i.e., λ_1, λ_2, and λ_3 in Eq. (8), was analyzed. For the analysis of λ_1, λ_2 was fixed to 10^{-2} and λ_3 to 10^{-3}, and TSLIP was run with λ_1 set to $10^{-6}, 10^{-5}, 10^{-4}, \ldots, 10^{2}, 10^{3}$. For the analysis of λ_2, λ_1 was set to 10^{-4}, and λ_3 to 10^{-3}, and TSLIP was run with the λ_2, similar to λ_1, that was set to $10^{-6}, 10^{-5}, 10^{-4}, \ldots, 10^{2}, 10^{3}$. For the analysis of λ_3, λ_1 was set to 10^{-4} and λ_2 to 10^{-2}, and TSLIP was run with λ_3 that was set to $10^{-6}, 10^{-5}, 10^{-4}, \ldots, 10^{2}, 10^{3}$. Figures 4 and 5 show the experimental results for the seven metrics on the datasets M2B and Emotion6, respectively. Note that for the metrics Sørensen, Chebyshev, Euclidean, Squared χ^2, and KL, the smaller the value, the better the performance. However, for metrics Intersection, and Cosine, the larger the value, the better the performance. The parameter λ_1 shown in Figure 4 reveals that when the value of λ_1 was 1, the performance was the best, and when λ_1 took a small value (e.g., $10^{-3}, 10^{-2}, 10^{-1}$) or large value (e.g., $10^{1}, 10^{2}, 10^{3}$), the performance worsened for all seven evaluations. This is because when λ_1 is too small, the proposed label-independent prediction loss term plays a small role, whereas when λ_1 is large, the objective function shown in Eq. (8) is not dominated by the third or fourth terms. Similarly, the results for λ_2 and λ_3 exhibit the same trend as λ_1. These phenomena provide additional proof that the proposed method is reliable.

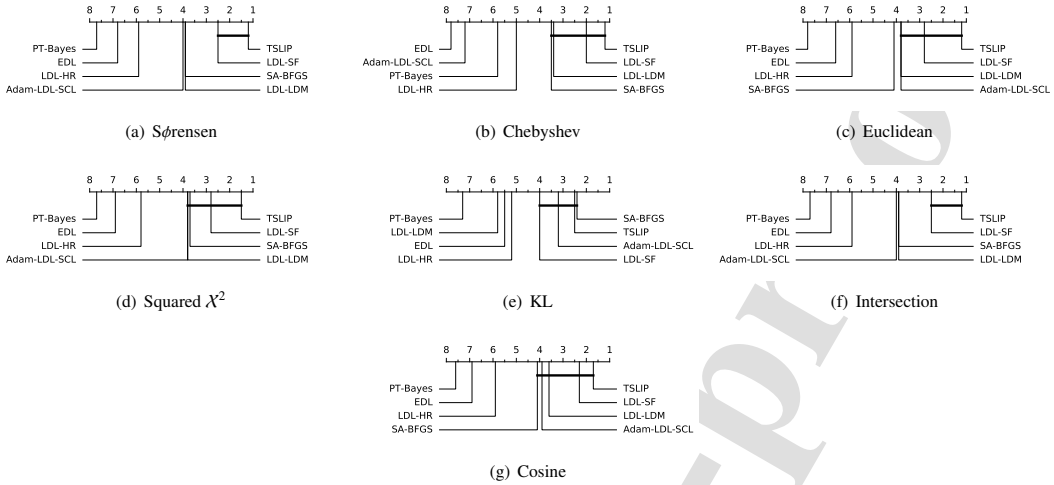
4.5. Convergency

To examine the ADMM algorithm's effectiveness in solving the TSLIP model, the value of the objective function Eq. (8) on four datasets (s-JAFFE, M2B, Twitter_LDL, and Flickr_LDL) are plotted in Figure 6. Evidently, as the iterations progressed, the objective function value decreased. In detail, on s-JAFFE and M2B, the convergency occurred after 60 and 30 iterations, respectively, whereas on Twitter_LDL and Flickr_LDL, the convergency occurred after 20 iterations.

4.6. Results and discussion

Ten times ten-fold cross-validation were performed on each dataset, and the mean and standard deviation of each metric were recorded. Table 9 lists the experimental results of the eight comparison algorithms on the 12 datasets based on seven metrics. Each data is marked with "mean±std" and the best results on each row are highlighted. "AVG.Rank" is the average ranking of each algorithm on the corresponding evaluation metric. Evidently from the table, the proposed TSLIP algorithm ranked almost in the top two among the 84 sets of results, corresponding to 12 datasets and seven evaluation metrics. Among them, 58 ranked first, accounting for 69.1%, and 14 ranked second, accounting for 16.7%.

To conduct a more reliable comparative analysis, the Friedman test [59], which is a useful statistical test for comparing more than two algorithms over multiple datasets, was further examined. Table 10 summarizes the Friedman statistics F_F and



(a) Sørensen (b) Chebyshev (c) Euclidean

(d) Squared $\mathcal{X}^2$ (e) KL (f) Intersection

(g) Cosine

Figure 7: Comparison of our algorithm against seven comparison algorithms with the *Bonferroni-Dunn* test. Algorithms not connected with TSLIP are considered to have a significantly different performance from TSLIP (CD = 2.69 at 0.05 significance level).

Table 10: Friedman statistics F_F in terms of each evaluation metric and the critical value at 0.05 significance level (comparing algorithms $k = 8$ and datasets $N = 12$)

Metric	F_F	Critical Value
Sørensen	37.3770	
Chebyshev	123.6939	
Euclidean	36.0468	
Squared χ^2	31.7778	2.1310
KL	10.4584	
Intersection	37.3770	
Cosine	32.3396	

corresponding critical value for each measure. As presented in Table 10, for each metric, the null hypothesis of indistinguishable performance at the 0.05 significance level among the compared algorithms was rejected. Furthermore, the Bonferroni–Dunn test [59] at a 0.05 significance level was used to test whether the proposed TSLIP algorithm has competitive performance against other algorithms. The performances of the two algorithms are significantly different if their average ranks over all datasets differ by at least one critical difference (CD). Figure 7 shows CD diagrams for each metric. In each subgraph, any algorithm is connected to the TSLIP if its average rank and TSLIP's average rank are within a CD interval. Algorithms independent of TSLIP are considered to have significantly different performances. As shown in Figure 7, the proposed TSLIP method outperformed PT-Bayes, EDL, and LDL-HR on every metric. The proposed algorithm outperformed Adam-LDL-SCL on Sørensen, Chebyshev, and the intersection. Further, TSLIP performed better on Sørensen, KL, and intersections compared with LDL-LDM. In addition, the proposed approach outperformed SA-BFGS in terms of Sørensen and Euclidean.

Based on the experimental findings, we can conclude that the proposed TSLIP algorithm achieves competitive performance compared to other state-of-the-art algorithms.

5. Conclusions and future works

This study proposed a novel two-stage LDL algorithm with label-independent prediction based on label-specific features. Herein, feature sparse selection and weight coefficients were combined to map the original feature space into a derived label-independent prediction space; this not only supports the extraction of label-specific features but also brings the derived space closer to the label distribution space. Label-independent predictions were normalized to obtain the final label distribution. Experimental results on 12 real-world datasets based on seven metrics validated the effectiveness of the proposed algorithm.

Nonetheless, some issues still exist that require further investigation.

1) Modification of the model framework. To improve the effectiveness of model solving, techniques such as softmax can be used to remove the constraints of optimizing the objective function.

2) Exploring more effective ways of feature selection. In this study, the $l_{2,1}$-norm was used for feature selection. In future work, we will try other constraint terms, such as l_1-norm, Frobenius norm, l_1-norm, $l_{1,2}$-norm, $l_{1/2}$-norm, and $l_{2/3}$-norm, to complete the similar work.

3) Exploring more efficient ways to employ label-independent prediction on LDL. Label-independent prediction space is a missing space in LDL derived from inference. An important focus of our future work will be on how to more effectively

actualize the mining of this space and utilize it to anticipate label distribution.

4) Combination of matrix factorization and label-independent prediction for LDL. To improve computational efficiency, we can convert the initial high-dimensional sparse feature matrix into a low-rank space.

5) Application for LE. LE is dedicated to transforming the original multi-label database into an LDL database. To improve performance, we can add label-independent prediction constraints during label enhancement.

6. Acknowledgments

This study was supported by the National Natural Science Foundation of China (61902328), the Applied Basic Research Project of Science and Technology Bureau of Nanchong City (SXHZ040), and Central Government Funds of Guiding Local Scientific and Technological Development (2021ZYD0003).

References

[1] Rolf Jagerman, Julia Kiseleva, and Maarten de Rijke. Modeling label ambiguity for neural list-wise learning to rank. *arXiv preprint arXiv:1707.07493*, 2017.

[2] Grigorios Tsoumakas and Ioannis Katakis. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, 3(3):1–13, 2007.

[3] Yang Liu, Kai-Wen Wen, Quan-Xue Gao, Xin-Bo Gao, and Fei-Ping Nie. Svm based multi-label learning with missing labels for image annotation. *Pattern Recognition*, 78:307–317, 2018.

[4] Bao-Yuan Wu, Siwei Lyu, Bao-Gang Hu, and Qiang Ji. Multi-label learning with missing labels for image annotation and facial action unit recognition. *Pattern Recognition*, 48:2279–2289, 2015.

[5] Dong Liang, Xin-Bo Gao, Wen Lu, and Li-Huo He. Deep multi-label learning for image distortion identification. *Signal Processing*, 172:107536, 2020.

[6] Amin Hashemi, Mohammad Bagher Dowlatshahi, and Hossein Nezamabadi-Pour. A bipartite matching-based feature selection for multi-label learning. *International Journal of Machine Learning and Cybernetics*, 12(2):459–475, 2021.

[7] Sheng-Jun Huang and Zhi-Hua Zhou. Multi-label learning by exploiting label correlations locally. In *AAAI*, pages 949–955, 2012.

[8] Xuan Wu, Qing-Guo Chen, Yao Hu, Deng-Bao Wang, Xiao-Dong Chang, Xiao-Bo Wang, and Min-Ling Zhang. Multi-view multi-label learning with view-specific information extraction. In *IJCAI*, pages 3884–3890, 2019.

[9] Jia-Qi Lv, Tian-Ran Wu, Cheng-Lun Peng, Yun-Peng Liu, Ning Xu, and Xin Geng. Compact learning for multi-label classification. *Pattern Recognition*, 113:107–833, 2021.

[10] Xin Geng. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1734–1748, 2016.

[11] Xin Geng, Kate Smith-Miles, and Zhi-Hua Zhou. Facial age estimation by learning from label distributions. In *AAAI*, pages 451–456, 2010.

[12] Zhao-Xiang Zhang, Mo Wang, and Xin Geng. Crowd counting in public video surveillance by label distribution learning. *Neurocomputing*, 166:151–163, 2015.

[13] Miao-Gen Ling and Xin Geng. Soft video parsing by label distribution learning. *Frontiers of Computer Science*, 13(2):302–317, 2019.

[14] Min-Ling Zhang and Lei Wu. Lift: Multi-label learning with label-specific features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(1):107–120, 2014.

[15] Ze-Bang Yu and Min-Ling Zhang. Multi-label classification with label-specific feature generation: A wrapped approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5199–5210, 2021.

[16] Ting-Ting Ren, Xiu-Yi Jia, Wei-Wei Li, Lei Chen, and Ze-Chao Li. Label distribution learning with label-specific features. In *IJCAI*, pages 3318–3324, 2019.

[17] Manuel González, José-Ramón Cano, and Salvador García. Prolsfeo-ldl: Prototype selection and label-specific feature evolutionary optimization for label distribution learning. *Applied Sciences*, 10(9):3089–3104, 2020.

[18] Xin Geng, Qin Wang, and Yu Xia. Facial age estimation by adaptive label distribution learning. In *ICPR*, pages 4465–4470, 2014.

[19] Xin Geng, Chao Yin, and Zhi-Hua Zhou. Facial age estimation by learning from label distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 2401–2412, 2013.

[20] Ju-Feng Yang, Dong-Yu She, and Ming Sun. Joint image emotion classification and distribution learning via deep convolutional neural network. In *IJCAI*, pages 3266–3272, 2017.

[21] Peng Zhao and Zhi-Hua Zhou. Label distribution learning by optimal transport. In *AAAI*, pages 4506–4513, 2018.

[22] Xiu-Yi Jia, Wei-Wei Li, Jun-Yu Liu, and Yu Zhang. Label distribution learning by exploiting label correlations. In *AAAI*, pages 3310–3317, 2018.

[23] Heng-Ru Zhang, Yu-Ting Huang, Yuan-Yuan Xu, and Fan Min. Cos-ldl: Label distribution learning by cosine-based distance-mapping correlation. *IEEE Access*, pages 63961–63970, 2020.

[24] Xiu-Yi Jia, Yu-Nan Lu, and Fang-Wen Zhang. Label enhancement by maintaining positive and negative label relation. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–13, 2021.

[25] Wen-Bin Qian, Yin-Song Xiong, Jun Yang, and Wen-Hao Shu. Feature selection for label distribution learning via feature similarity and label correlation. *Information Sciences*, pages 38–59, 2022.

[26] Jing Wang and Xin Geng. Label distribution learning by exploiting label distribution manifold. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–14, 2021.

[27] Xiang Zheng, Xiu-Yi Jia, and Wei-Wei Li. Label distribution learning by exploiting sample correlations locally. In *AAAI*, pages 4556–4563, 2018.

[28] Yu-Meng Guo, Fu-Lai Chung, Guo-Zheng Li, Jian-Cong Wang, and James C Gee. Leveraging label-specific discriminant mapping features for multi-label learning. *ACM Transactions on Knowledge Discovery from Data*, 13(2):1–23, 2019.

[29] Jun-Long Li, Pei-Pei Li, Xue-Gang Hu, and Kui Yu. Learning common and label-specific features for multi-label classification with correlation information. *Pattern Recognition*, 121:1–15, 2022.

[30] Xiu-Yi Jia, Sai-Sai Zhu, and Wei-Wei Li. Joint label-specific features and correlation information for multi-label learning. *Journal of Computer Science and Technology*, 35(2):247–258, 2020.

[31] Jia Zhang, Can-Dong Li, Dong-Lin Cao, Yao-Jin Lin, Song-Zhi Su, Liang Dai, and Shao-Zi Li. Multi-label learning with label-specific features by resolving label correlations. *Knowledge-Based Systems*, 159:148–157, 2018.

[32] Zi-Wei Cheng and Zi-Wei Zeng. Joint label-specific features and label correlation for multi-label learning with missing label. *Applied Intelligence*, 50(11):4029–4049, 2020.

[33] Jun Huang, Feng Qin, Xiao Zheng, Ze-Kai Cheng, Zhi-Xiang Yuan, Wei-Gang Zhang, and Qing-Ming Huang. Improving multi-label classification with missing labels by learning label-specific features. *Information Sciences*, 492:124–146, 2019.

[34] Wen-Bin Qian, Qian-Zhi Ye, Yi-Hui Li, and Shi-Ming Dai. Label distribution feature selection with feature weights fusion and local label correlations. *Knowledge-Based Systems*, 256:1–16, 2022.

[35] Ning Xu, Yun-Peng Liu, and Xin Geng. Label enhancement for label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 33(4):1632–1643, 2019.

[36] Ning Xu, Jun Shu, Yun-Peng Liu, and Xin Geng. Variational label enhancement. In *ICML*, pages 10597–10606, 2020.

[37] Ning Xu, Cong-Yu Qiao, Jia-Qi Lv, Xin Geng, and Min-Ling Zhang. One positive label is sufficient: Single-positive multi-label learning with label enhancement. In *NeurIPS*, pages 1–17, 2022.

[38] Ning Xu, Cong-Yu Qiao, Xin Geng, and Min-Ling Zhang. Instance-dependent partial label learning. In *NeurIPS*, volume 34, pages 27119–27130, 2021.

[39] Qing-Hai Zheng, Ji-Hua Zhu, Hao-Yu Tang, Xin-Yuan Liu, Zhong-Yu Li, and Hui-Min Lu. Generalized label enhancement with sample correlations. *IEEE Transactions on Knowledge and Data Engineering*, pages

1–14, 2021.

[40] Yan Cui, Jie-Lin Jiang, Zhi-Hui Lai, Zuo-Jin Hu, Yu-Quan Jiang, and Wai-Keung Wong. New semi-supervised classification using a multi-modal feature joint l21-norm based sparse representation. *Signal Processing: Image Communication*, pages 94–106, 2018.

[41] Zheng Zhang, Yong Xu, Jian Yang, Xue-Long Li, and David Zhang. A survey of sparse representation: algorithms and applications. *IEEE Access*, pages 490–530, 2015.

[42] Jacob Benesty, Jing-Dong Chen, Yi-Teng Huang, and Israel Cohen. Pearson correlation coefficient. In *Noise Reduction in Speech Processing*, pages 1–4. 2009.

[43] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, pages 1–122, 2011.

[44] Bin Liu, Ying-Ming Li, and Zeng-Lin Xu. Manifold regularized matrix completion for multi-label learning with admm. *Neural Networks*, pages 57–67, 2018.

[45] Guang-Can Liu, Zhou-Chen Lin, and Yong Yu. Robust subspace segmentation by low-rank representation. In *ICML*, pages 663–670, 2010.

[46] Jr John E Dennis and JJ Moré. Quasi-newton methods, motivation and theory. *SIAM Review*, 19(1):46–89, 1977.

[47] Ya-Xiang Yuan. A modified bfgs algorithm for unconstrained optimization. *IMA Journal of Numerical Analysis*, 11(3):325–332, 1991.

[48] Michael Lyons, Shigeru Akamatsu, Miyuki Kamachi, and Jiro Gyoba. Coding facial expressions with gabor wavelets. In *FG*, pages 200–205, 1998.

[49] Yi Ren and Xin Geng. Sense beauty by label distribution learning. In *IJCAI*, pages 2648–2654, 2017.

[50] Tam V Nguyen, Si Liu, Bing-Bing Ni, Jun Tan, Yong Rui, and Shuicheng Yan. Sense beauty via face, dressing, and/or voice. In *ACM MM*, pages 239–248, 2012.

[51] Kuan-Chuan Peng, Tsuhan Chen, Amir Sadovnik, and Andrew C Gallagher. A mixed bag of emotions: Model, predict, and transfer emotion distributions. In *CVPR*, pages 860–868, 2015.

[52] Ju-Feng Yang, Ming Sun, and Xiao-Xiao Sun. Learning visual sentiment distributions via augmented conditional probability neural network. In *AAAI*, pages 224–230, 2017.

[53] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893, 2005.

[54] Markus Andreas Stricker and Markus Orengo. Similarity of color images. In *Storage and Retrieval for Image and Video Databases III*, pages 381–392, 1995.

[55] Ling-Yu Liang, Luo-Jun Lin, Lian-Wen Jin, Duo-Rui Xie, and Meng-Ru Li. Scut-fbp5500: A diverse benchmark dataset for multi-paradigm facial beauty prediction. In *Pattern Recognition*, pages 1598–1603, 2018.

[56] Jing Wang and Xin Geng. Learn the highest label and rest label description degrees. In *IJCAI*, pages 3097–3103, 2021.

[57] De-Yu Zhou, Xuan Zhang, Yin Zhou, Quan Zhao, and Xin Geng. Emotion distribution learning from texts. In *EMNLP*, pages 638–647, 2016.

[58] Xiu-Yi Jia, Ze-Chao Li, Xiang Zheng, Wei-Wei Li, and Sheng-Jun Huang. Label distribution learning with label correlations on local samples. *IEEE Transactions on Knowledge and Data Engineering*, 33(4):1619–1631, 2019.

[59] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, 2006.

Highlights

- By embedding a derived label-independent prediction space in the feature space and the label distribution space, we propose a Two-Stage label distribution learning algorithm with Label-Independent Prediction based on label-specific features (TSLIP).
- Label-independent prediction can ensure that different instances have the same label value under the foundation of the same specific feature values.
- We design a sparse weighting matrix to transform the feature space into the label-independent prediction space for extracting label-specific features.
- We incorporate instance correlation and label correlation to improve the generalization ability of the model.

Credit Author Statement

Gui-Lin Li: Methodology, Software, Writing- Original draft preparation

Heng-Ru Zhang: Conceptualization, Supervision, Funding acquisition, Resources

Fan Min: Conceptualization, Supervision, Writing - Review & Editing

Yu-Nan Lu: Software, Validation, Investigation

Conflict of interest statement

We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work. There is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled.