Electronic Research Archive

https://www.aimspress.com/journal/era

ERA, 32(4): 2598–2620.
DOI: 10.3934/era.2024118
Received: 22 January 2024
Revised: 27 February 2024
Accepted: 13 March 2024
Published: 28 March 2024

Research article

Multi-label feature selection via constraint mapping space regularization

Bangna Li[1], Qingqing Zhang[2,*] and Xingshi He[2]

[1] College of Arts and Sciences, Yangling Vocational and Technical College, Yangling 712100, China
[2] School of Science, Xi'an Polytechnic University, Xi'an 710600, China

* **Correspondence:** Email: qq_zhang@xpu.edu.cn.

Abstract: Multi-label feature selection, an essential means of data dimension reduction in multi-label learning, has become one of the research hotspots in the field of machine learning. Because the linear assumption of sample space and label space is not suitable in most cases, many scholars use pseudo-label space. However, the use of pseudo-label space will increase the number of model variables and may lead to the loss of sample or label information. A multi-label feature selection scheme based on constraint mapping space regularization is proposed to solve this problem. The model first maps the sample space to the label space through the use of linear mapping. Second, given that the sample cannot be perfectly mapped to the label space, the mapping space should be closest to the label space and still retain the space of the basic manifold structure of the sample space, so combining the Hilbert-Schmidt independence criterion with the sample manifold, basic properties of constraint mapping space. Finally, the proposed algorithm is compared with MRDM, SSFS, and other algorithms on multiple classical multi-label data sets; the results show that the proposed algorithm is effective on multiple indicators.

Keywords: multi-label learning; feature selection; Hilbert-Schmidt independence criterion; manifold learning; linear mapping

1. Introduction

Feature selection is one of the important dimensionality reduction methods to deal with dimensional disasters [1, 2], whether in single-label clustering, classification, or multi-label classification. It is also a hot topic in research on machine learning and data analysis [3].

As a type of dimensionality reduction technology, feature selection methods select the most representative feature subset from the original features of the sample by applying a certain strategy or model to remove redundant and irrelevant features and thus achieve the task of reducing data dimensionality [4]. In addition, feature selection, as a common dimensionality reduction technique,

has a several advantages, such as the ability to reduce the calculation and storage pressure of the learning algorithm and improve its robustness and interpretability. Based on the interaction with the learning system, multi-label feature selection can be divided into filter [5–7], wrapper [8, 9], or embedded methods [10–12]. The embedded method is different from the filter method, which completely ignores the learning algorithm's influence on feature selection. The embedded method is also different from the wrapper method, which completely relies on a learning algorithm to guide feature selection. The embedded method embeds the feature selection process in the learning algorithm and makes it complete feature selection in the learning process.

By reviewing the research on the existing embedded models, we know that most of the existing models are based on linear mapping and information theory and are often combined with manifold learning and sparse regular terms to construct multi-label feature selection models. In linear mapping-based approaches, either a linear mapping is directly from samples to real labels, or a linear mapping is constructed by using pseudo-labels instead of real labels. However, the binary nature of real labels contradicts the nature of the continuous type of variables of linear mapping. In addition, the use of pseudo-labels increases the number of variables of the model, thus increasing the computational burden of the model. Fortunately, we demonstrate here that constraining the mapping space of linear mappings directly in the model can alleviate this problem well. Specifically, we construct a novel sparse multi-label feature selection model by introducing the Hilbert-Schmidt independence criterion [13] and sample manifold learning and combining the L_{21} norm as a sparse constraint. Relative to the existing state-of-the-art models, the proposed model alleviates the problem of real label space being non-applicable to linear mapping by constraining the mapping space, it also more effectively reduces the computational burden of the model and improves the stability of the model.

The main research work of this paper is as follows:

1) We introduce the HSIC and sample manifold, which jointly constrain the fundamental properties of the mapping space in terms of both real-label and sample structure.

2) We introduce the L_{21} norm as a sparse constraint, which allows for the construction of a sparse multi-label feature selection model for constrained mapping spaces, and an optimization algorithm with convergence has been designed to optimize the proposed method.

3) A comparative experiment was conducted with eight highly influential multi-label feature selection algorithms. Experimental results show that the proposed method is effective and feasible.

The rest of the paper is summarized as follows: in Section 2, notations and a brief overview of existing models are given. In Section 3, the model establishment, optimization, and convergence proofs for the proposed algorithm are introduced. In Section 4, the results of comparative tests are presented and analyzed, and the proposed algorithm's parameter sensitivity, convergence, and time complexity are tested and analyzed. Finally, the summary of this paper and the direction of future research are given in Section 5.

2. Related work

2.1. Notations of this paper

For any matrix $A \in R^{c\times d}$, A^T is the transpose of A; A_{ij} is a member of the ith row and jth column of A; the ith row vector of A is denoted by A_{i*}; the jth column vector of A is denoted by A_{*j}; the L_{21} norm

of A is $\|A\|_{21}$; the Frobenius norm of A is $\|A\|_F$; S_A is the similarity matrix with respect to the matrix A; L_A is the Laplacian matrix of the similar matrix S_A; n, d and m represent the number of samples, the number of features, and the number of labels, respectively. $X \in R^{n\times d}$ is the sample matrix; $Y \in R^{n\times m}$ is the label matrix; $Q \in R^{d\times m}$ is the weight matrix; $H \in R^{n\times n}$ is the center matrix.

2.2. A review of multi-label learning

In this subsection, we briefly review the research status of embedded multi-label feature selection techniques by discussing many works of literature on multi-label feature selection.

Among the multi-label feature selection models based on information theory, the classic representative models include SCLS [14], MDMR [15], PMU [16], and FIMF [17]. Among them, MDMR, PMU, and FIMF use mutual information to quantify the importance of features and select features for their importance. However, these models may lose important information when processing higher-order label data. Moreover, the computational cost of high-order multivariate mutual information is prohibitive. However, SCLS has poor algorithm performance due to a combination of excessive labels and features, making feature selection impractical [18, 19]. In addition, a multi-label feature selection method considering maximum correlation was proposed in [20]. In order to ensure the correlation between the selected features and different label groups, the model integrates the maximum correlation of high-order label correlation into the feature selection model. In addition, Gao et al. [21] designed a multi-label feature selection method that includes three low-order information theory terms and unified the framework of multi-label information theory methods.

In linear mapping-based multi-label feature selection models, some models directly apply a linear mapping between samples to real labels. For example, Li et al. [22] combined this penalty term with the low-dimensional labeled manifold and proposed a multi-label feature selection method based on robust, flexible, and sparse regularization (RFSFS). Similarly, Li et al. [23] devised a highly sparse paradigm and combined it with a linear mapping between samples to real labels, as well as low redundancy constraints. Thus, a multi-label feature selection technique with high-sparsity, personalized and low-redundancy shared common features is proposed. However, since the duality of real labels does not apply in linear mapping, most scholars have constructed pseudo-label space that is suitable for linear mapping through the use of samples or real labels. In addition, due to the rapid development of manifold learning, it has been widely applied in many fields, such as cooperative clustering [24], feature selection algorithms, and dimensionality reduction [25–27]. In a feature selection model, manifold learning can constrain the consistency of topological structures between two spaces, and scholars often introduce manifold learning into multi-label feature selection models to improve the performance of the models.

By combining linear mapping and manifold learning, Zhang et al. [28] developed a manifold regularized discriminative feature selection technique for multi-label learning (MDFS). The model utilizes the manifold structure of the sample manifold and the low dimensional manifold of the label. The Frobenius norm distance between the real-label matrix and the pseudo-label matrix is used to ensure that the pseudo-label does not lose the properties of the real label. Finally, the sparsity of the feature weight matrix is constrained by the L_{21} norm. Huang et al. [29] constrained pseudo-label learning by combining the HSIC and sample manifold methodology in the linear mapping. A multi-label feature selection technique (MRDM) was proposed based on manifold regularization and

dependence maximization.

Hu et al. [30] constructed the common structure of sample space and label space by utilizing multiple linear maps and a proposed multi-label feature selection technique (SCMFS) with a shared common mode. The model maps sample space and label space to pseudo-label space by using linear mapping technique to obtain pseudo-label space through the use of a common structure of sample space and label space; furthermore, combining this with the L_{21} norm sparse constraint, multi-label feature selection is realized. The specific formula of this model is as follows:

$$\min_{W,V,Q,B} \|XW - V\|_F^2 + \alpha\|X - VQ\|_F^2 + \beta\|Y - VB\|_F^2 + \gamma\|W\|_{2,1}, \tag{2.1}$$

where V is the pseudo-label matrix. Q and B are the corresponding coefficient matrices, respectively. W is the feature weight matrix. α, β and γ are the regularized parameters.

Similar to SCMFS, to improve the model's performance, Gao et al. [31] introduced sample manifolds into the shared structure model to strengthen the constraints on pseudo-labels through the sample manifolds when the pseudo-label matrix and the sample matrix have the same geometric manifold structure. A multi-label feature selection technique (SSFS) with a constrained latent structure shared term is proposed. The specific formula of this model is as follows:

$$\begin{aligned} &\min_{V,M,Q} \|X - VQ^T\|_F^2 + \alpha\|Y - VM\|_F^2 + \\ &\beta tr(V^T LV) + \gamma\|Q\|_{2,1}, \\ &s.t. V, M, Q \geq 0. \end{aligned} \tag{2.2}$$

where V is the pseudo-label matrix; L is the Laplacian matrix of X; Q and M are the corresponding coefficient matrices.

In addition, Zhang and Ma [32] proposed a multi-label feature selection technique (NMDG) that utilizes dynamic graph constraints to improve the model's performance and generalization ability. In this model, the learning of pseudo-labels is constrained by linear maps and the label manifold, and the learning of the feature weight matrix is constrained by the feature manifold and low-dimensional dynamic manifold of the pseudo-labels to realize feature selection. The objective function of this model is as follows:

$$\begin{aligned} &\min_{W,b,F} \|XW + 1_n^T b - F\|_F^2 + \alpha tr(F^T L_Y F) + \\ &\beta tr(WL_{F^T} W^T) + \gamma tr(W^T L_{X^T} W), \\ &s.t.\ \ (W, F) \geq 0. \end{aligned} \tag{2.3}$$

where F is the pseudo-label matrix; b is the bias vector. L_Y, L_{F^T}, and L_{X^T} are the Laplacian matrices of the label similarity matrix, the dynamic Laplacian matrix of the pseudo-label low-dimensional similarity matrix, and the Laplacian matrix of the feature similarity matrix, respectively.

In summary, RFSFS and ERSFS directly apply a linear mapping of samples to real labels to construct the model. However, the performance of the model is degraded by the fact that the binary nature of real labels does not apply to linear mapping methodology. To address this problem, models

such as MRDM and SSFS models use pseudo-labels instead of real labels to construct linear maps, which increases the computational burden of the model and reduces the stability of the model. Unlike the state-of-the-art models described above, we mitigate the problem of real labels not being applicable to linear maps by constraining the mapping space of linear maps. This also avoids the problems of large computational burden and instability that the use of pseudo-labels imposes on the model. The specific technical details of the proposed model will be given in Section 3.

3. Proposed method

This section presents a sparse multi-label feature selection technique (CRMFS) based on a constrained mapping space and manifold regularization. Moreover, the optimization solution and convergence proof of the model are also given in this section.

3.1. Problem description

Let $X^T = [X_{1*}, X_{2*}, \cdots, X_{n*},]$ be the transpose of the sample matrix $X \in R^{n\times d}$ and $X_{i*} \in R^{1\times d}$ represent the ith row sample vector of X. The label matrix of X is $Y = [Y_{*1}, Y_{*2}, \cdots, Y_{*m},]$, $Y \in R^{n\times m}$, and $Y_{*i} \in R^{n\times 1}$ is the jth column label vector of Y. X and Y jointly form multi-label data set $D = \{(X_{i*}, Y_{i*})|i = 1, 2, \cdots, n\}$, where $Y_{i*} \in R^{1\times m}$ is the ith row vector of Y and represents the label vector corresponding to the ith sample. In addition, $Y_{ij} \in \{0, 1\}$ is the ith row, jth column member of Y. When $Y_{ij} = 1$, it means that the ith sample X_{i*} belongs to the jth label Y_{*j}. When $Y_{ij} = 0$, sample X_{i*} does not belong to the label Y_{*j}. Multi-label data set D is used to construct the corresponding model to realize feature selection.

3.2. CRMFS

The brief review in subsection 2.2 shows that both the MRDM model and the SSFS model are types of multi-label feature selection sparse models. Sparse models are generally constructed by combining linear mapping methodology and sparse constraints because of the simplicity of least squares calculation and strong interpretability. The specific formula of the objective function of the sparse model is as follows:

$$\min_{Q} \|XQ - Y\|_F^2 + \alpha R(Q), \tag{3.1}$$

where α is the penalty parameter and $\| * \|_F$ represents the Frobenius norm. $Q \in R^{d\times m}$ is the coefficient matrix for the linear mapping. Since Q_{i*} can represent the importance of its corresponding feature X_{i*}, Q is also called the feature weight matrix. $R(*)$ is the penalty function of $*$.

In sparse models, the L_1 norm and L_{21} norm are often used as penalty functions. Different penalty functions have different properties, leading to different properties of the constrained variables. For example, the L_1 norm can simultaneously guide the inter-line and intra-line sparsity of the constrained variables. The L_{21} norm can guide the constrained variable inter-row sparsity and intra-row stability. In multi-label feature selection, it is more suitable to use L_{21} norm constraint Q sparsity than the L_1 norm to better distinguish the importance of features. Let $R(Q) = \|Q\|_{2,1}$ and construct a sparse model that is suitable for feature selection:

$$\min_{Q} \|XQ - Y\|_F^2 + \alpha\|Q\|_{2,1}, \tag{3.2}$$

where $\|Q\|_{2,1} = \sum_{i=1}^{d}(\sum_{j=1}^{m} Q_{ij}^2)^{\frac{1}{2}}$ represents the norm of Q.

Considering that the linear mapping has a specific deviation, in order to improve the generalization ability of the model, we introduce a bias vector $b \in R^{m\times 1}$ into the linear mapping scheme and reconstruct the sparse model to be as follows:

$$\min_{Q,b} \|XQ + vb^T - Y\|_F^2 + \alpha\|Q\|_{2,1}, \tag{3.3}$$

where b^T is the transpose of b and $v \in R^{n\times 1}$ is the column vector of all ones.

In addition, since label space is binary and unsuitable for linear mapping, many scholars tend to construct non-binary pseudo-label space instead of real-label space to realize feature selection. However, the use of pseudo-label space will increase the number of variables in the model, affecting the model's efficiency, and may lead to the loss of important label information in the process of pseudo-label construction, which affects the performance of the model. In order to avoid the adverse effects of learning pseudo-label space, we try to solve the aforementioned problem by constraining the properties of the mapping space XQ.

As can be ascertained from Eq (3.3), linear mapping takes sample space as the initial mapping space, so in an ideal state, mapping space XQ should retain the basic geometric structure of sample space X and maintain the consistency of the topological structure with sample space X. According to the hypothesis, if X_{i*} and X_{j*} have a high degree of similarity, $X_{i*}Q$ and $X_{i*}Q$ should also have a high degree of similarity. Mapping space XQ and sample space X should have the same basic manifold structure.

$$\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\|X_{i*}Q - X_{j*}Q\|_2^2(S_X)_{ij} = tr(Q^TX^T(D_X - S_X)XQ) = tr(Q^TX^TL_XXQ), \tag{3.4}$$

where $L_X \in R^{n\times n}$ is the Laplacian matrix of $S_X \in R^{n\times n}$ and $L_X = D_X - S_X$. $D_X \in R^{n\times n}$ is the diagonal matrix and $(D_X)_{ii} = \sum_{j=1}^{n}(S_X)_{ij}$. $(S_X)_{ij}$ is the ith row jth column element of the sample similarity matrix S_X. In this study, we chose to use the Gaussian function to learn S_X:

$$(S_X)_{ij} = \begin{cases} exp(-\frac{\|X_{i*}-X_{j*}\|_2^2}{\delta}), & if\ \ X_{i*} \in N_k(X_{j*})\ or\ X_{j*} \in N_k(X_{i*}), \\ 0, & others \end{cases} \tag{3.5}$$

where the parameter $\delta \in R$. $N_k(*)$ represents the k-nearest neighbor of $*$, and k is the nearest neighbor threshold.

Combining Eqs (3.3) and (3.4), we can obtain a sparse multi-label feature selection model with manifold constraints:

$$\min_{Q,b} \|XQ + vb^T - Y\|_F^2 + \alpha\|Q\|_{2,1} + \beta tr(Q^TX^TL_XXQ), \tag{3.6}$$

where β is the manifold regular term parameter.

In addition, it can be ascertained from Eq (3.3) that the target space of a linear mapping is label space, but due to the binarity of label space, sample space cannot be well mapped to label space. However, considering that the use of pseudo-label space will increase the model complexity and reduce the robustness of the model, the mapped space should have the primary information of the label space. It can remove irrelevant or redundant interference information. To solve this problem, the HSIC [13] is adopted to constrain the correlation between mapping space and label space. The specific formula is as follows:

$$\max_{Q} tr(HXQ(XQ)^T HYY^T), s.t.\ (XQ)^T XQ = V_n. \tag{3.7}$$

where $H \in R^{n\times n}$ is the center matrix and $V_* \in R^{*\times *}$ is the identity matrix.

Combining Eqs (3.6) and (3.7), we can obtain the objective function of the CRMFS model:

$$\min_{Q,b} \|XQ + vb^T - Y\|_F^2 + \alpha\|Q\|_{2,1} + \beta tr(Q^T X^T L_X XQ) - \gamma tr(HXQ(XQ)^T HYY^T), \tag{3.8}$$

where γ is the regularization parameter. Since Eq (3.7) does not converge without any constraint and in Eq (3.8), the other terms have become the constraints of the fourth term, it is not required for constraint $(XQ)^T XQ = V$ to be added in Eq (3.8).

In addition, for any matrix A, $\|A\|_F^2 = tr(A^T A)$. In addition, due to the non-smoothness of the L_{21} norm, $tr(Q^T MQ)$ is used here to replace the L_{21} norm and obtain the approximate solution of Eq (3.8); thus, the objective function of the CRMFS model can be rewritten as follows:

$$\begin{aligned} &\min_{Q,b} tr[(XQ + vb^T - Y)^T (XQ + vb^T - Y)] + \alpha tr(Q^T MQ) + \\ &\beta tr(Q^T X^T L_X XQ) - \gamma tr(HXQ(XQ)^T HYY^T), \end{aligned} \tag{3.9}$$

where $M \in R^{d\times d}$ is the diagonal matrix and $M_{ii} = \frac{1}{2\|Q_{i*}\|_2}$.

3.3. Optimal solution

According to Eq (3.9), in the CRMFS model, the function of b is given by

$$\begin{aligned} \Phi(b) &= \min_{Q,b} tr[(XQ + vb^T - Y)^T (XQ + vb^T - Y)] \\ &= tr(Q^T X^T XQ) + tr(bv^T vb^T) + tr(Y^T Y) - 2tr(Q^T X^T bv^T) - 2tr(bv^T Y) - tr(Q^T X^T Y). \end{aligned} \tag{3.10}$$

Take the partial derivative of the above equation with respect to b:

$$\frac{\partial \Phi(b)}{\partial b} = 2bv^T v + 2Q^T X^T v - 2Y^T v. \tag{3.11}$$

Let $\frac{\partial \Phi(b)}{\partial b} = 0$; then, we have

$$b = \frac{1}{n}(Y^T v - Q^T X^T v). \tag{3.12}$$

Combining Eqs (3.9) and (3.12), the objective function of the CRMFS model can be transformed as follows:

$$\begin{aligned} \Phi(Q) = \min_{Q} & tr[(XQ - Y)^T H(XQ - Y)] + \alpha tr(Q^T MQ) + \\ & \beta tr(Q^T X^T L_X XQ) - \gamma tr(HXQ(XQ)^T HYY^T), \end{aligned} \tag{3.13}$$

where $H = V_n - \frac{1}{n}vv^T$ is the center matrix.

It is easy to prove that the above equation is convex with respect to Q, so we find the optimal solution for Q by taking its derivative. The derivative function of the above equation with respect to Q is given by

$$\frac{\partial \Phi(Q)}{\partial Q} = 2X^T HXQ - 2X^T HY + 2\alpha MQ + 2\beta X^T L_X XQ - 2\gamma X^T HYY^T HXQ. \tag{3.14}$$

Let $\frac{\partial \Phi(Q)}{\partial Q} = 0$; then, we can get the update formula for Q:

$$Q = [M^{-1}(X^T HX + \beta X^T L_X X - \gamma X^T HYY^T HX) + \alpha V_d]M^{-1}X^T HY. \tag{3.15}$$

According to the solution process of appeal optimization, we designed and present the algorithm of the CRMFS model, as shown in Table 1.

Table 1. CRMFS algorithm.

Input:	Multi-label data set D. Regularization parameters α, β, and γ. The number of selected features k.
Output:	The result I of feature selection.
1)	Initialize $H = V - \frac{1}{n}vv^T$.
2)	Initialize $t = 0$, and set M^t to be the identity matrix.
3)	Calculate S_X and L_X from Eqs (3.4) and (3.5).
4)	**Repeat:**
	Update Q^{t+1},
	$Q^{t+1} = [(M^t)^{-1}(X^T HX + \beta X^T L_X X - \gamma X^T HYY^T HX) + \alpha V_d](M^t)^{-1}X^T HY$.
	Update M^{t+1}, $M_{ii}^{t+1} = \frac{1}{\|Q_{i*}^t\|_2}$.
	Update b^{t+1}, $b = \frac{1}{n}(Y^T v - (Q^{t+1})^T X^T v)$.
	$t = t + 1$.
5)	**Until convergence.**
6)	Compute $\|Q_{i*}^t\|_2$, $(i = 1, 2, \cdots, d)$, and sort it, assigning the first k largest to I

3.4. Proof of convergence

In this subsection, we give the theoretical proof of convergence of the CRMFS algorithm; however, before proving convergence, we need to know the following lemma:

Lemma 1 [33]: For any non-zero vectors $a \in R^{l\times m}$ and $c \in R^{l\times m}$, both make the following formula true:

$$\|a\|_2 - \frac{\|a\|_2^2}{2\|c\|_2} \leq \|c\|_2 - \frac{\|c\|_2^2}{2\|c\|_2}. \tag{3.16}$$

In combination with Eq (3.12), Eq (3.8) can be transformed to be as follows:

$$\Phi(Q) = \min_{Q} \|H^{1/2}(XQ - Y)\|_F^2 + \alpha\|Q\|_{2,1} + \beta tr(Q^T X^T L_X XQ) - \gamma tr(HXQ(XQ)^T HYY^T). \tag{3.17}$$

According to Lemma 1, in iteration t, we have

$$\|(Q_{i*})^{t+1}\|_2 - \frac{\|(Q_{i*})^{t+1}\|_2^2}{2\|(Q_{i*})^t\|_2} \leq \|(Q_{i*})^t\|_2 - \frac{\|(Q_{i*})^t\|_2^2}{2\|(Q_{i*})^t\|_2}. \tag{3.18}$$

The sum of Eq (3.18) can be derived as follows:

$$\sum_{i=1}^{d}(\|(Q_{i*})^{t+1}\|_2 - \frac{\|(Q_{i*})^{t+1}\|_2^2}{2\|(Q_{i*})^t\|_2}) \leq \sum_{i=1}^{d}(\|(Q_{i*})^t\|_2 - \frac{\|(Q_{i*})^t\|_2^2}{2\|(Q_{i*})^t\|_2}). \tag{3.19}$$

Further transformation yields

$$\alpha\|Q^{t+1}\|_{2,1} - \alpha\sum_{i=1}^{d}(\frac{\|(Q_{i*})^{t+1}\|_2^2}{2\|(Q_{i*})^t\|_2}) \leq \alpha\|Q^t\|_{2,1} - \alpha\sum_{i=1}^{d}(\frac{\|(Q_{i*})^t\|_2^2}{2\|(Q_{i*})^t\|_2}). \tag{3.20}$$

By combining Eqs (3.17) and (3.20), we can derive the following:

$$\Phi(Q^{t+1}) \leq \Phi(Q^t). \tag{3.21}$$

In conclusion, the convergence of the CRMFS algorithm is proved.

4. Experiment

This section discusses a comparative experiment between CRMFS and seven advanced multi-label feature selection algorithms (RFSFS [22], SSFS [31], MRDM [29], SCLS [14], MDMR [15], PMU [16], FIMF [17], MFS-MCDM [34]). The experiment was conducted on ten classical real multi-label data sets. In the experiment, five indicators, including hamming loss, ranking loss, one-error, coverage, and average precision, were used as evaluation indicators, and the ML-KNN algorithm [35] was used as the representative algorithm for classification.

4.1. Experimental settings

Ten classic multi-label data sets were collected from five fields of research, including biology, images, and text. These multi-label data sets were all taken from Mulan Library (http://mulan.sourceforge.net/datasets.html). In Table 2, we give the total number of samples, number

Table 2. Data set descriptions.

NO.	Data set	Instances	Features	Labels	Domain	Card	Training	Test
1	Yeast	2417	103	14	biology	4.237	1500	917
2	Emotion	593	72	6	music	1.869	391	202
3	Birds	645	260	19	audio	1.014	322	323
4	Scene	2407	294	6	images	1.047	1211	1196
5	Image	600	294	5	images	1.236	400	200
6	Enron	1702	1001	53	text	3.378	1123	579
7	Flags	194	19	7	images	3.392	129	65
8	Medical	978	1449	45	text	1.245	645	333
9	Genbase	662	1185	27	biology	1.252	463	199
10	CAL500	502	68	174	audio	26.044	335	167

of features, number of labels, domain, cardinality (Card), and other parameters for each experimental data set.

Experimental environment: All relevant experimental experiments included a Microsoft Windows 10 system; processor: ADM Ryzen 5 3600 6-core Processor 3.59 GHz; memory: 16.00 GB; and programming software: Matlab R2016a.

First, we discretized [36] all experimental data to have equal-width intervals. The smoothing and nearest-neighbor parameters in the ML-KNN algorithm were set to 1 and 10, respectively. Meanwhile, the parameters of the FIMF algorithm were also set to their default values: $b = 2$ and $Q = 10$. Second, since the Gaussian function is used in CRMFS, SSFS, MRDM, and other algorithms to learn the basic manifolds of samples or labels, the nearest neighbor parameter was set to 5; also, the label nearest-neighbor parameter for Image data was set to 3, and $\delta = 1$ was set. Additionally, we set the selected feature range to $[5, 10, 15, 20, 25, 30, 35, 40, 45, 50]$ and expressly set the selected feature range to $2 \sim 18$ for the Flags data set. Finally, we applied the grid search strategy to set all experimental algorithms' adjustment range of the regularization parameter to be $[0.001, 0.01, 0.1, 1, 10, 100, 1000]$.

4.2. Evaluation metric

In this subsection, we give the detailed meanings and formulas for hamming loss, ranking loss, one-error, coverage, and average precision measures in five multi-label classification indexes, where "↓" means that the smaller the value of relevant indexes, the better the algorithm performance; "↑" means that the larger the value of relevant indexes, the better the algorithm performance.

Let $D \in R^{n\times d}$ be the sample data of the training set and $Y \in R^{n\times m}$ be the corresponding label set data. $h(D_{i*})$ represents the binary label vector, and $rank_{i*}(q)$ represents the rank of the label prediction Y_{q*}.

1) Hamming loss (↓): Represents the percentage of misclassified labels.

$$HL(D) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{m}\|h(D_{i*}) \triangle Y_{i*}\|_1, \tag{4.1}$$

where $HL \in [0, 1]$ and $\triangle$ is the symbol of symmetry difference.

2) Ranking loss (↓): Measure the gap between the predicted list and the actual sorted list.

$$RL(D) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{1_m^T Y_{i*} 1_m^T \overline{Y_{i*}}}\sum_{q:Y_{i*}^q=1}\sum_{q':Y_{i*}^{q'}=0}(P), \tag{4.2}$$

where $RL \in [0,1]$. $P = \delta(rank_{i*}(q) \geq rank_{i*}(q^{'}))$. $\delta(z)$ is the indicator function and $\overline{Y_{i*}}$ is the complement of Y_{i*} on Y.

3) One-error (↓): Represents that there is no sample proportion of the predicted most relevant predicted label among the real labels.

$$OE(D) = \frac{1}{n}\sum_{i=1}^{n}\delta(Y_{i*}^{q_i} = 0), \tag{4.3}$$

where $OE \in [0,1]$ and $q_i = argmin_{q\in[1,m]}rank_{i*}(q)$.

4) Coverage (↓): Represents the average number of steps required for the sorted labels to cover the real-label correlation set.

$$CV(D) = \frac{1}{n}\sum_{i=1}^{n} argmax_{q:Y_{i*}^{q}=1}rank_{i*}(q) - 1, \tag{4.4}$$

where $CV \in [0, m-1]$.

5) Average precision (↑): Represents the percentage of labels in the ranking that are more relevant than a particular label.

$$AP(D) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{1_m^T Y_{i*}}\sum_{q:Y_{i*}^{q}=1}\frac{\sum_{q^{'}:Y_{i*}^{q^{'}}=1}(P)}{rank_{i*}(q)}, \tag{4.5}$$

where $AP \in [0,1]$.

4.3. Results and discussion

In Tables 3–7, we show the optimal results of each experimental algorithm under the optimal parameters in the experimental range. Among them, SSFS, MRDM, and other algorithms contain multiple variables, so these algorithms results are presented as the mean values of 10 runs. Meanwhile, in Tables 3–7, we use bolding to mark the optimal results under the same index on the same data set, indicating that the experimental algorithm with the bolded results on the data has the optimal algorithm performance under this index. In addition, we denote its performance ranking under this indicator on this data set by including "()" next to each result in Tables 3–7; finally, in the last two rows of Tables 3–7, we also added two rows , i.e., "Rank" and "Average", where "Rank" represents the average ranking of the overall performance of each experimental algorithm under this index. "Average" means that on all data sets, each value is the mean value of the optimal result of the algorithm under this index.

As can be ascertained from Tables 3–7, although the performance of the CRMFS algorithm on the Scene data set is slightly inferior to that of the MRDM algorithm, the average and rank of the CRMFS algorithm are optimal under each index, which also indicates that the overall performance of the CRMFS algorithm is better than that of all comparison algorithms.

Table 3. Average precision (↑) comparison for different algorithms on each data set.

Algorithms	CRMFS	MRDM	MFS-MCDM	SCLS	MDMR	PMU	FIMF	SSFS	RFSFS
Yeast	**0.7617 (1.5)**	**0.7617 (1.5)**	0.7551 (7)	0.7563 (4)	0.7579 (3)	0.7562 (5)	0.7552 (6)	0.7312 (9)	0.7411 (8)
Emotion	**0.8071 (1)**	0.8036 (2)	0.7815 (3)	0.7496 (8)	0.7551 (6)	0.7143 (9)	0.7510 (7)	0.7584 (5)	0.7665 (4)
Birds	**0.5632 (1)**	0.5026 (5)	0.5302 (2)	0.4435 (6.5)	0.4158 (8)	0.4435 (6.5)	0.4074 (9)	0.5143 (4)	0.5145 (3)
Scene	0.8367 (2)	**0.8390 (1)**	0.8058 (4)	0.8163 (3)	0.7633 (8)	0.8034 (5)	0.6906 (9)	0.7727 (7)	0.7799 (6)
Image	**0.7725 (1)**	0.7633 (2)	0.7288 (5)	0.7437 (3)	0.7058 (7)	0.7002 (8)	0.6791 (9)	0.7208 (6)	0.7376 (4)
Enron	**0.6686 (1)**	0.6613 (2)	0.6103 (9)	0.6589 (3)	0.6566 (5)	0.6483 (7)	0.6548 (6)	0.6585 (4)	0.6331 (8)
Flags	**0.8519 (1)**	0.8436 (3)	0.8425 (4)	0.8024 (9)	0.8462 (2)	0.8411 (5.5)	0.8410 (7)	0.8372 (8)	0.8411 (5.5)
Medical	**0.8570 (1)**	0.7415 (6.5)	0.8539 (2)	0.4431 (9)	0.8242 (4)	0.6992 (8)	0.8349 (3)	0.7415 (6.5)	0.8082 (5)
Genbase	**0.9939 (1)**	0.9899 (6)	0.7071 (7)	0.6882 (8)	0.9919 (2)	0.9907 (4)	0.9915 (3)	0.6044 (9)	0.9904 (5)
CAL500	**0.5015 (1)**	0.4979 (3)	0.4966 (4)	0.4942 (8)	0.4959 (5.5)	0.4930 (9)	0.4959 (5.5)	0.4945 (7)	0.4986 (2)
Average	**0.7614**	0.7404	0.7112	0.6596	0.7213	0.7090	0.7101	0.6834	0.7311
Rank	**1.15**	3.2	4.7	6.15	5.05	6.7	6.45	6.55	5.05

Table 4. Hamming loss (↓) comparison for different algorithms on each data set.

Algorithms	CRMFS	MRDM	MFS-MCDM	SCLS	MDMR	PMU	FIMF	SSFS	RFSFS
Yeast	**0.1940 (1)**	0.1965 (2)	0.2014 (6)	0.2006 (4.5)	0.1999 (3)	0.2006 (4.5)	0.2021 (7)	0.2137 (9)	0.2091 (8)
Emotion	0.1988 (2)	**0.1972 (1)**	0.2302 (5)	0.2500 (8)	0.2409 (6)	0.2673 (9)	0.2252 (4)	0.2418 (7)	0.2248 (3)
Birds	**0.0464 (1)**	0.0479 (3.5)	0.0471 (2)	0.0499 (6)	0.0505 (8)	0.0504 (7)	0.0520 (9)	0.0495 (5)	0.0479 (3.5)
Scene	0.1045 (2)	**0.1027 (1)**	0.1066 (3)	0.1073 (4)	0.1348 (8)	0.1137 (5)	0.1587 (9)	0.1290 (7)	0.1260 (6)
Image	**0.1970 (1)**	0.2030 (2)	0.2050 (3)	0.2110 (5)	0.2240 (7)	0.2270 (8)	0.2340 (9)	0.2164 (6)	0.2108 (4)
Enron	**0.0487 (1)**	0.0499 (4)	0.0544 (9)	0.0495 (2)	0.0505 (6.5)	0.0505 (6.5)	0.0501 (5)	0.0498 (3)	0.0522 (8)
Flags	0.6000 (5)	0.6154 (6)	**0.5802 (1)**	0.6330 (7)	0.5934 (3)	0.5934 (3)	0.5934 (3)	0.6413 (8)	0.6462 (9)
Medical	**0.9803 (1)**	0.9846 (6.5)	0.9806 (2)	0.9998 (9)	0.9825 (4)	0.9883 (8)	0.9810 (3)	0.9846 (6.5)	0.9831 (5)
Genbase	**0.0020 (1)**	0.0030 (4.5)	0.0342 (8)	0.0326 (7)	0.0024 (2)	0.0056 (6)	0.0030 (4.5)	0.0428 (9)	0.0029 (3)
CAL500	0.9643 (2)	0.9648 (3.5)	0.9655 (6)	0.9651 (5)	0.9657 (7.5)	0.9667 (9)	0.9657 (7.5)	0.9648 (3.5)	**0.9638 (1)**
Average	**0.3336**	0.3365	0.3405	0.3499	0.3445	0.3464	0.3465	0.3534	0.3467
Rank	**1.7**	3.4	4.5	5.75	5.5	6.6	6.1	6.4	5.05

Table 5. One-error (↓) comparison for different algorithms on each data set.

Algorithms	CRMFS	MRDM	MFS-MCDM	SCLS	MDMR	PMU	FIMF	SSFS	RFSFS
Yeast	**0.2127 (1)**	0.2268 (2.5)	0.2356 (4)	0.2268 (2.5)	0.2366 (6)	0.2366 (6)	0.2366 (6)	0.2508 (9)	0.2414 (8)
Emotion	**0.2574 (1)**	0.2673 (2)	0.3119 (4)	0.3614 (8.5)	0.3564 (7)	0.3614 (8.5)	0.3515 (6)	0.3455 (5)	0.3020 (3)
Birds	**0.5116 (1)**	0.5523 (4)	0.5465 (2)	0.6454 (7)	0.6744 (8)	0.6395 (6)	0.7035 (9)	0.5872 (5)	0.5488 (3)
Scene	0.2692 (2)	**0.2634 (1)**	0.3169 (4)	0.2977 (3)	0.3905 (8)	0.3904 (7)	0.4983 (9)	0.3661 (6)	0.3532 (5)
Image	**0.3600 (1)**	0.3650 (2)	0.4200 (5)	0.4000 (4)	0.4450 (7)	0.4700 (8)	0.5000 (9)	0.4300 (6)	0.3980 (3)
Enron	**0.2263 (1)**	0.2349 (2)	0.3472 (9)	0.2470 (6)	0.2435 (3.5)	0.2694 (7)	0.2453 (5)	0.2435 (3.5)	0.2846 (8)
Flags	**0.1094 (1)**	0.1250 (2.5)	0.1406 (4.5)	0.2031 (9)	0.1563 (7)	0.1719 (8)	0.1250 (2.5)	0.1500 (6)	0.1406 (4.5)
Medical	**0.1682 (1)**	0.3243 (6.5)	0.1772 (2)	0.6727 (9)	0.2072 (4)	0.3664 (8)	0.2012 (3)	0.3243 (6.5)	0.2252 (5)
Genbase	**0 (2.5)**	0.0050 (6)	0.4221 (7)	0.4472 (8)	**0 (2.5)**	**0 (2.5)**	**0 (2.5)**	0.5477 (9)	0.0030 (5)
CAL500	0.0838 (3.5)	0.0838 (3.5)	0.0838 (3.5)	0.0898 (7.5)	0.0898 (7.5)	0.0898 (7.5)	0.0898 (7.5)	0.0838 (3.5)	**0.0790 (1)**
Average	**0.2199**	0.2448	0.3002	0.3591	0.2800	0.2995	0.2951	0.3329	0.2576
Rank	**1.5**	3.2	4.5	6.45	6.05	6.85	5.95	5.95	4.55

Specifically, Table 3 shows the optimal performance comparison for each experimental algorithm under the average precision index. As ascertained from Table 3, although, for the Scene data set, the performance of the CRMFS algorithm decreases by 0.023 relative to that of the MRDM algorithm, it is still superior to comparison algorithms such as SSFS and MFS-MCDM. In addition, the optimal performance of the CRMFS algorithm on other experimental data sets was always in first place, and the overall performance of the CRMFS algorithm under this index was improved by 3.2% ∼ 15.84% relative to other comparison algorithms.

Table 4 shows the optimal performance of each experimental algorithm under the hamming loss index. As seen in Table 4, although the overall performance of the CRMFS algorithm was only improved by 0.86% ∼ 5.6%, it was still greatly improved on some data sets. For example, compared

Table 6. Ranking loss (↓) comparison for different algorithms on each data set.

Algorithms	CRMFS	MRDM	MFS-MCDM	SCLS	MDMR	PMU	FIMF	SSFS	RFSFS
Yeast	**0.1673 (1)**	0.1674 (2)	0.1742 (5)	0.1745 (6)	0.1710 (3)	0.1723 (4)	0.1747 (7)	0.1925 (9)	0.1851 (8)
Emotion	**0.1680 (1.5)**	**0.1680 (1.5)**	0.1864 (3)	0.2056 (8)	0.1994 (5)	0.2570 (9)	0.2012 (7)	0.2010 (6)	0.1981 (4)
Birds	**0.1875 (1)**	0.2226 (5)	0.1944 (2)	0.2655 (9)	0.2591 (8)	0.2585 (6)	0.2586 (7)	0.2138 (4)	0.2070 (3)
Scene	0.0977 (2)	**0.0959 (1)**	0.1201 (4)	0.1129 (3)	0.1444 (7)	0.1290 (5)	0.1994 (9)	0.1463 (8)	0.1422 (6)
Image	**0.1896 (1)**	0.1921 (2)	0.2258 (5)	0.2167 (3)	0.2550 (8)	0.2483 (7)	0.2662 (9)	0.2477 (6)	0.2189 (4)
Enron	**0.0890 (1)**	0.0903 (2)	0.1034 (9)	0.0921 (4)	0.0944 (6)	0.0949 (7)	0.0935 (5)	0.0913 (3)	0.0954 (8)
Flags	**0.1784 (1)**	0.1898 (5)	0.1823 (3.5)	0.2482 (9)	0.1813 (2)	0.1977 (6)	0.1823 (3.5)	0.2078 (7)	0.2102 (8)
Medical	**0.0445 (1)**	0.0638 (6.5)	0.0468 (3)	0.1326 (9)	0.0522 (4)	0.0693 (8)	0.0466 (2)	0.0638 (6.5)	0.0534 (5)
Genbase	**0.0062 (1)**	0.0079 (4.5)	0.0480 (7)	0.0653 (8)	0.0066 (2)	0.0080 (6)	0.0078 (3)	0.0869 (9)	0.0079 (4.5)
CAL500	**0.1804 (1)**	0.1822 (6)	0.1823 (7)	0.1833 (9)	0.1818 (4)	0.1818 (4)	0.1818 (4)	0.1831 (8)	0.1817 (2)
Average	**0.1309**	0.1380	0.1464	0.1697	0.1545	0.1617	0.1612	0.1634	0.1500
Rank	**1.15**	3.55	4.85	6.8	4.9	6.2	5.65	6.65	5.25

Table 7. Coverage (↓) comparison for different algorithms on each data set.

Algorithms	CRMFS	MRDM	MFS-MCDM	SCLS	MDMR	PMU	FIMF	SSFS	RFSFS
Yeast	**6.2617 (1)**	6.2999 (2)	6.4569 (7)	6.4482 (6)	6.3642 (3)	6.3708 (4)	6.3740 (5)	6.6772 (9)	6.5767 (8)
Emotion	1.9059 (2)	**1.8614 (1)**	1.9851 (3)	2.1139 (8)	2.0891 (7)	2.3614 (9)	2.0545 (4)	2.0842 (6)	2.0564 (5)
Birds	**2.2043 (1)**	2.7028 (5)	2.2755 (2)	3.2012 (9)	3.0495 (6)	3.0526 (7.5)	3.0526 (7.5)	2.5542 (4)	2.4582 (3)
Scene	0.5870 (2)	**0.5828 (1)**	0.7032 (4)	0.6681 (3)	0.8253 (7)	0.7492 (5)	1.0953 (9)	0.8329 (8)	0.8144 (6)
Image	1.0350 (2)	**1.0300 (1)**	1.1750 (5)	1.1650 (4)	1.3200 (8)	1.2900 (7)	1.3550 (9)	1.2520 (6)	1.1630 (3)
Enron	**12.7720 (1)**	12.8860 (2)	13.9260 (9)	13.0415 (4)	13.1606 (5)	13.4128 (8)	13.2038 (6)	12.8950 (3)	13.3762 (7)
Flags	3.6308 (2.5)	3.6462 (4)	3.6769 (5)	4.0462 (9)	3.6308 (2.5)	3.7231 (6)	**3.6000 (1)**	3.8400 (8)	3.7692 (7)
Medical	**2.9580 (1)**	3.9429 (6.5)	3.0631 (2)	7.1291 (9)	3.3934 (5)	4.2943 (8)	3.1171 (3)	3.9429 (6.5)	3.3574 (4)
Genbase	**0.5729 (1)**	0.5879 (3)	1.6181 (7)	2.1809 (8)	0.5930 (4.5)	0.6080 (6)	0.5930 (4.5)	2.7236 (9)	0.5809 (2)
CAL500	**127.4400 (1)**	128.0500 (4)	127.9900 (3)	129.3700 (9)	128.9900 (6)	128.9900 (6)	128.9900 (6)	129.3000 (8)	127.7293 (2)
Average	**15.9368**	16.1590	16.2870	16.9364	16.3416	16.4852	16.3435	16.6102	16.1882
Rank	**1.45**	2.95	4.7	6.9	5.4	6.65	5.5	6.75	4.7

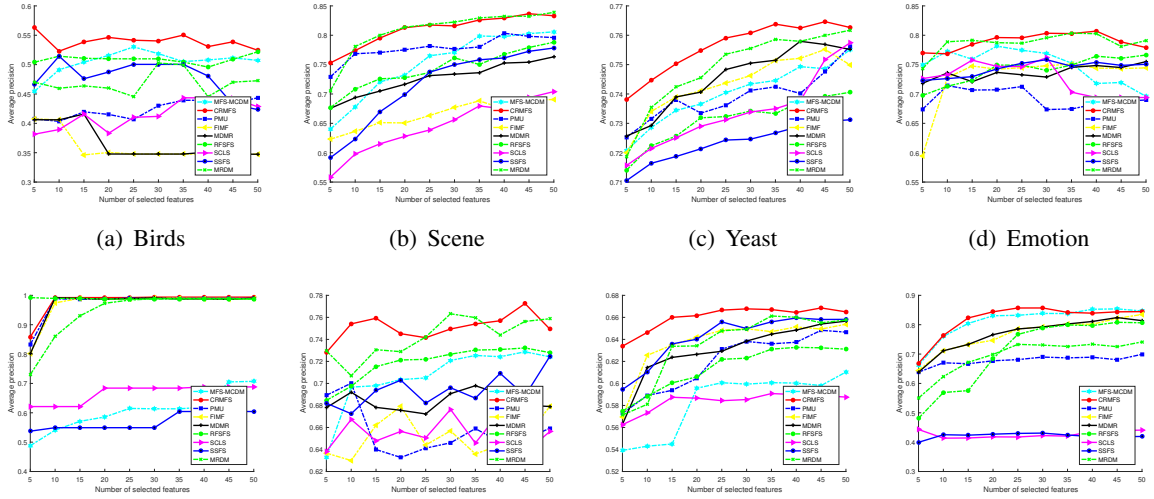
with the FIMF algorithm, the performance of the CRMFS algorithm on the Birds and Image data sets was improved by 10.76% and 15.81%, respectively. Even on the Genbase data set, the performance of the comparison algorithm was improved by 16.67% ∼ 95.33%.

Tables 5 and 6 respectively show the optimal performance of each experimental algorithm under the one-error and ranking loss indicators. As shown in Tables 5 and 6, among the ten experimental data sets, the optimal results of the CRMFS algorithm are bolded for 8 data sets in Table 5 and 9 data sets in Table 6. From Table 6 only on the Scene data set was the performance slightly inferior to that of the MRDM algorithm. In addition, relative to that of other comparison algorithms, the overall performance of the CRMFS algorithm under the one-error and ranking loss indexes was improved by 10.17% ∼ 38.76% and 5.14% ∼ 22.86%, respectively.

Table 7 compares the optimal performance of each experimental algorithm under the coverage index. As shown in Table 7, although the performance of the CRMFS algorithm was slightly worse than that of the MRDM algorithm on the Scene, Image, and Emotion data sets and slightly worse than that of the FIMF algorithm on the Flags data set, the optimal performance of the CRMFS algorithm consistently ranked first on other data sets. In addition, the overall performance of the CRMFS algorithm was improved by 1.38% ∼ 5.9% for each comparison algorithm.

In order to more intuitively show the performance of each experimental algorithm in terms of the number of selected features and the performance comparison results when the same number of features is selected, we constructed plots, with the number of selected features as the abscissa and the value of each indicator as the ordinate. Figures 1–5 show the results for each evaluation metric.

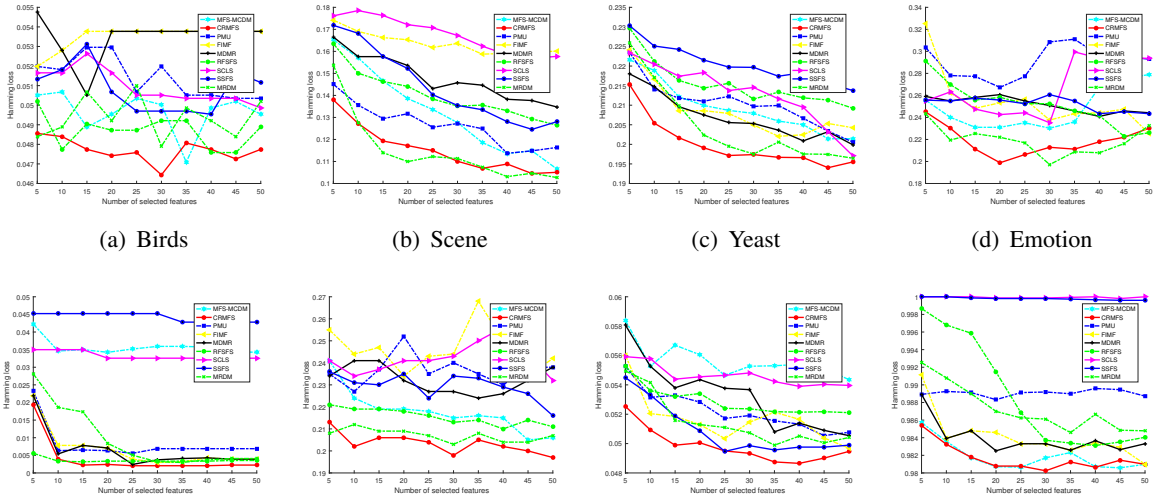
Figures 1–5 show the performance comparison results for the indicators of average precision,



(a) Birds (b) Scene (c) Yeast (d) Emotion

(e) Genbase (f) Image (g) Enron (h) Medical

Figure 1. Performance comparison results for average precision (↑) index.



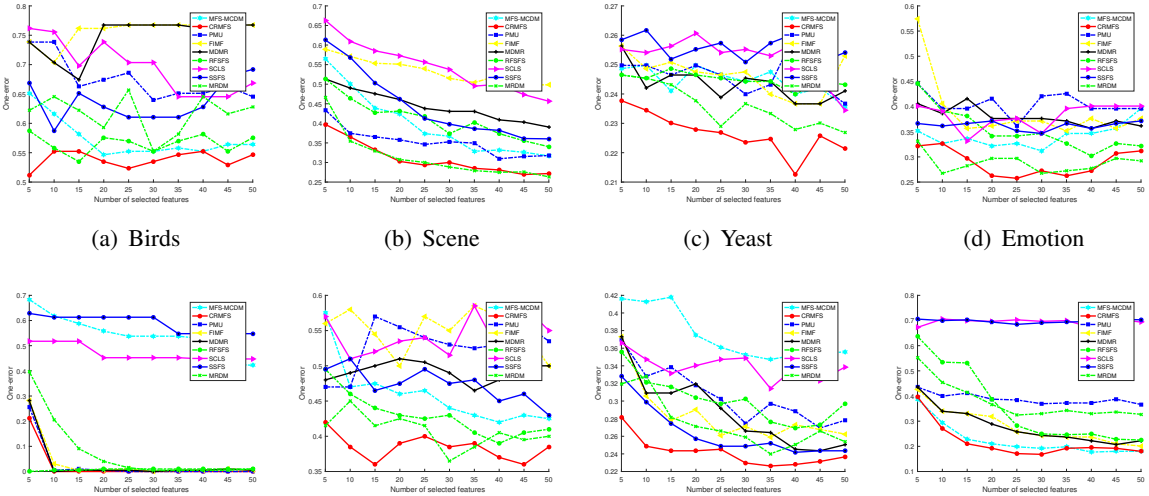
(a) Birds (b) Scene (c) Yeast (d) Emotion

(e) Genbase (f) Image (g) Enron (h) Medical

Figure 2. Performance comparison results for hamming loss (↓) index.

hamming loss, one-error, ranking loss, and coverage. From the overall view of Figures 1–5, first, the CRMFS algorithm effectively solves the problem of multi-label feature selection. Second, when the number of selected features was 5, the performance of the CRMFS algorithm was in the optimal position in most cases. In addition, the performance curve for the CRMFS algorithm was always in the best or second-best position. Therefore, compared with MRDM, SSFS, and other comparison algorithms, the CRMFS algorithm has certain advantages and more effective at solving the problem of multi-label feature selection.

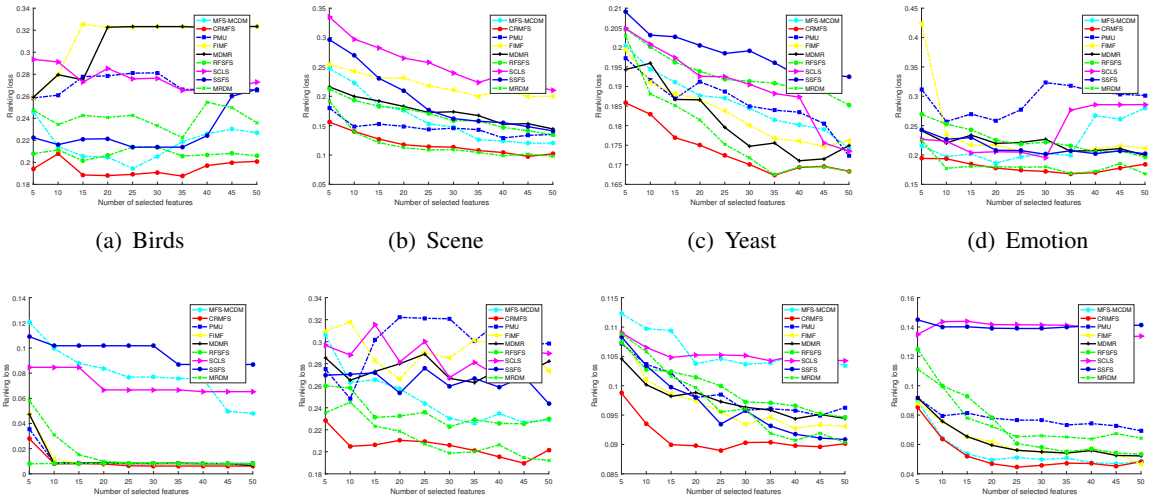
Specifically, it can be seen in Figure 1 that, in the cases of the data sets of Birds, Yeast, and Enron, the performance curves for the CRMFS algorithm are obviously above those for the MRDM and SSFS



(a) Birds (b) Scene (c) Yeast (d) Emotion

(e) Genbase (f) Image (g) Enron (h) Medical

Figure 3. Performance comparison results for one-error (↓) index.



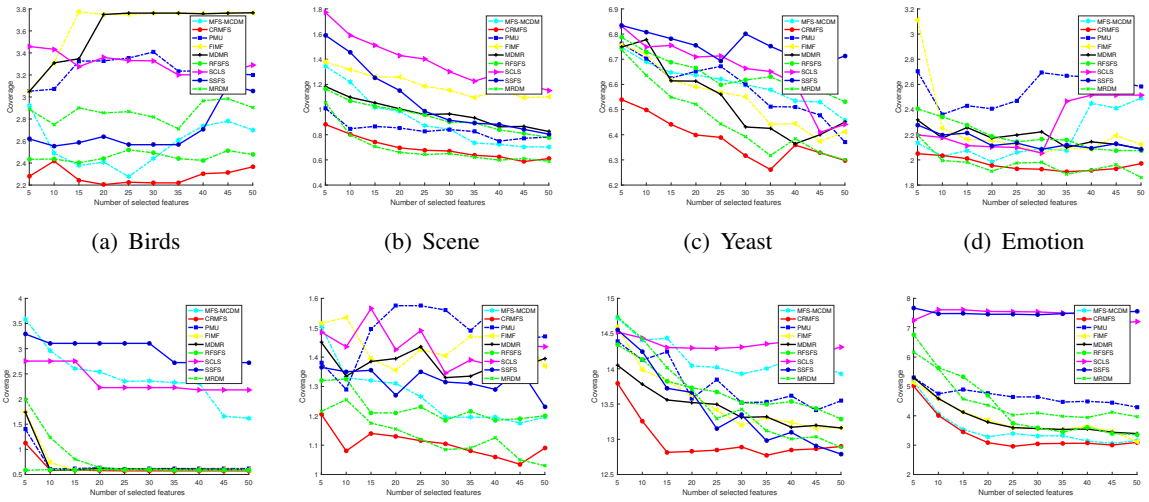
(a) Birds (b) Scene (c) Yeast (d) Emotion

(e) Genbase (f) Image (g) Enron (h) Medical

Figure 4. Performance comparison results for ranking loss (↓) index.

comparison algorithms. According to Figures 2–5, it can be seen that, in the cases of the data sets of Birds, Yeast, Image, and Enron, the performance curves for the CRMFS algorithm are lower than those for the MRDM and SSFS comparison algorithms. In addition, as shown in Figures 1–5, although the performance of the CRMFS algorithm was slightly inferior to that of the MRDM algorithm on the Scene data set, the performance of the CRMFS algorithm was still optimal on other data sets.

In conclusion, although the performance improvement of the CRMFS algorithm differed for different experimental data sets, in terms of the overall performance, the CRMFS algorithm is superior to MRDM, SSFS, and the other comparative algorithms.



(a) Birds (b) Scene (c) Yeast (d) Emotion

(e) Genbase (f) Image (g) Enron (h) Medical

Figure 5. Performance comparison results for coverage (↓) index.

4.4. Statistical test and analysis

To further analyze and compare the overall performance of each experimental algorithm, the Bonferroni-Dunn test ($\alpha = 0.01$) [37] was adopted to conduct a significant difference analysis and comparison of each experimental algorithm. The comparison results are visually shown in Figure 6.

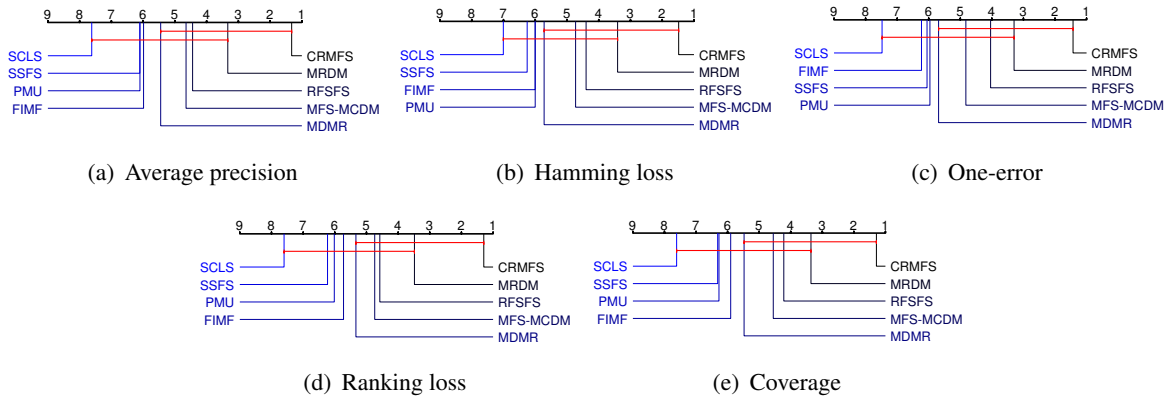
In Figure 6, the horizontal axis represents the overall performance ranking of all experimental algorithms. From left to right, the overall performance of the algorithm is getting better and better. A red line links the algorithms with no significant difference for convenience. There is a significant difference if the difference in the average ranking reaches the difference threshold (CD), where $CD = q_\alpha \sqrt{\frac{K(K+1)}{6N}}$. In this study, $q_\alpha = 3.590$ ($K = 9, a = 0.01$), and $CD = 4.3968$ ($K = 9, N = 10$). As shown in Figure 6, we can see that, although the CRMFS algorithm has no significant difference from the MRDM, MDMR, and MFS-MCDM algorithms, it is significantly different from other advanced comparison algorithms. Indeed, the CRMFS algorithm consistently ranked first on the far right in the overall performance ranking.

In addition, Friedman's non-parametric statistical test ($\alpha = 0.05$) [38] was used to analyze the experimental results of each algorithm on ten classical multi-label data sets, and the quantitative results were obtained. The specific formula of the Friedman test is as follows:

$$F_F = \frac{(N-1)\chi_F^2}{N(K-1)-\chi_F^2} \tag{4.6}$$

where K and N represent the number of algorithms and data sets in the experiment, respectively; $\chi_F^2 = \frac{12N}{K(K+1)}(\sum_{i=1}^{K} R_i^2 - \frac{K(K+1)^2}{4})$; R_i is the sorting value of the ith algorithm.

The quantitative results of the Friedman test are shown in Table 8. Friedman statistics and critical values for each indicator are given in Table 8. As seen in Table 8, Friedman's statistics for all indicators were far higher than the critical value, rejecting the null hypothesis of each evaluation indicator. It also



(a) Average precision (b) Hamming loss (c) One-error

(d) Ranking loss (e) Coverage

Figure 6. The Bonferroni-Dunn test results in the form of a mean rank plot.

shows that the overall performance of the CRMFS algorithm under different indicators is better than MRDM, SSFS, and the other advanced comparison algorithms.

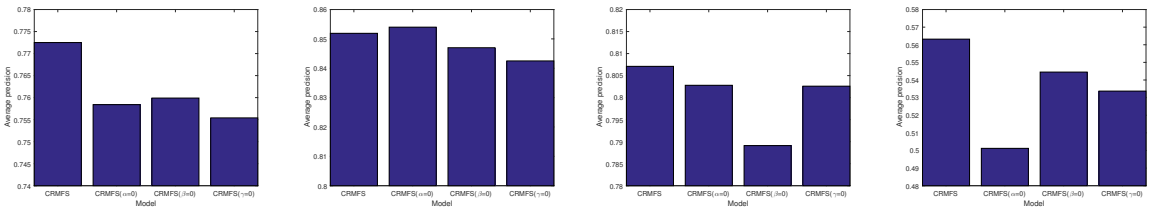
Table 8. The Friedman statistics and the critical value results for each evaluation metric.

Evaluation metric	F_F	Critical value ($\alpha = 0.05$)
Average precision	7.3574	
Hamming loss	4.6475	
One-error	6.5242	2.070
Ranking loss	6.4338	
Coverage	7.3487	

In summary, the experimental results of the Bonferroni-Dunn test and Friedman test show that the overall performance of the CRMFS algorithm is superior to SSFS, MRDM, and the other seven advanced comparison algorithms for either a single indicator or all indicators.

4.5. Ablation studies

In this subsection, we discuss some ablation studies that we conducted to explore the effectiveness of the modules in CRMFS. In this experiment, we set one parameter in CRMFS to 0, indicating that the module is not used; we also performed a grid search with the other two parameters in the range $[0.001, 0.01, 0.1, 1, 10, 100, 1000]$ to capture the optimal results of CRMFS for the average precision metric when the first 50 features are selected. The experimental results are shown in Figure 7.



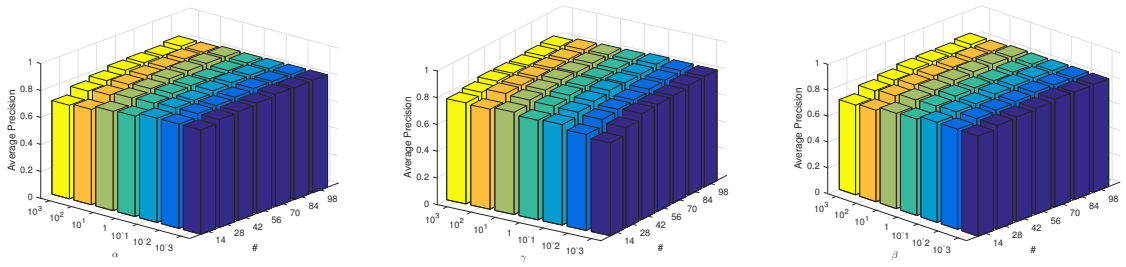
(a) Image (b) Flags (c) Emotion (d) Birds

Figure 7. Results of ablation studies on four data sets.

It can be seen that applying the L_{21} norm constraint on the Flags data set led to a slight performance degradation of CRMFS. Upon analysis, we believe that this was caused by the fact that applying the L_{21} norm constraint on the Flags data set does not result in satisfactory handling of the feature redundancy problem. In addition, we found that the performance of CRMFS under the average precision metric was significantly degraded when a module was not used, which suggests that all modules in CRMFS can effectively manage the multi-label feature selection problem.

4.6. Parameter sensitivity analysis

After the analysis in subsections 4.3 and 4.4, we know that the CRMFS algorithm has certain advantages and better effectiveness than the seven advanced multi-label feature selection algorithms. In order to more comprehensively analyze the influence of parameter changes on the CRMFS algorithm, we designed and conducted parameter sensitivity analysis experiments on the CRMFS algorithm on the Scene data set. In the experiment, we set the range of the number of feature selections to be $[14, 28, 42, 56, 70, 84, 98]$. For the parameters α, β, and γ in the CRMFS algorithm, We fixed two of the parameters to equal one and made the third parameter search within the range of $[0.001, 0.01, 0.1, 1, 10, 100, 1000]$ to observe the performance changes of the CRMFS algorithm for the average precision index. The experimental results are shown in Figure 8.



(a) Scene(α) (b) Scene(γ) (c) Scene(β)

Figure 8. Experimental results of parameter sensitivity analysis for the CRMFS algorithm on the Scene data set.

In Figure 8, # represents the number of features selected. According to Figure 8, we can see that the CRMFS algorithm's value for the average precision index increases with the number of selected features, so feature selection is practical. Second, we can see the optimal value range of each parameter in the CRMFS algorithm on the Scene data set, where the optimal value range of α was $[0.1, 0.01]$, the optimal value range of β was $[0.01, 0.1, 1, 10]$, and the optimal value range of γ was $[10, 100]$. In addition, the optimal range of parameters varies from data set to data set.

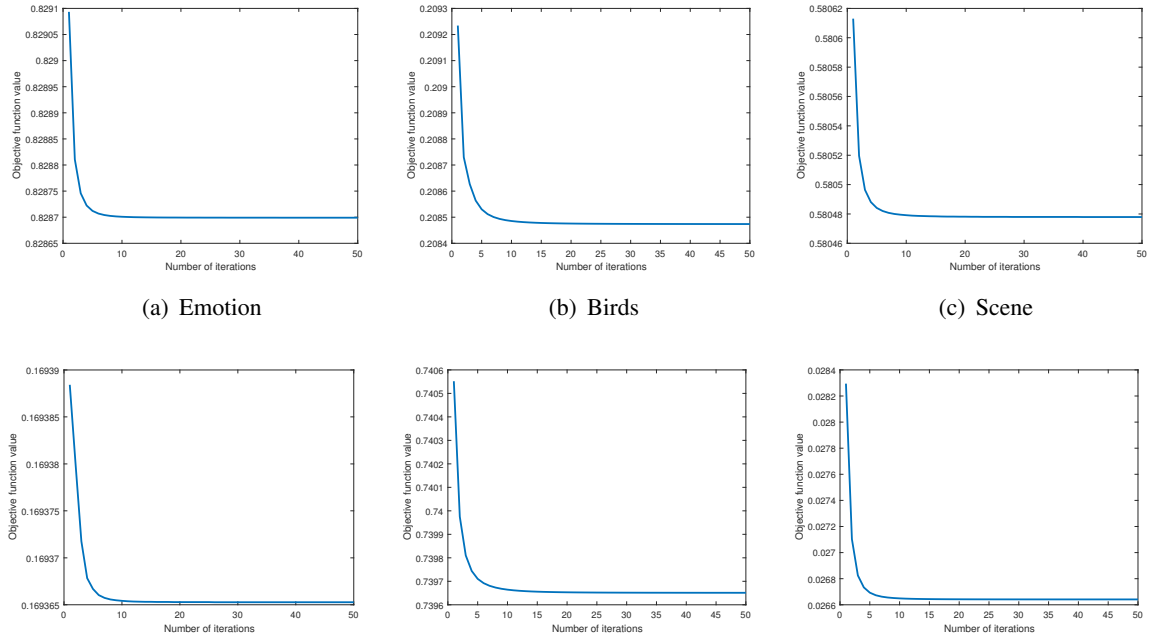
In conclusion, the CRMFS algorithm demonstrated strong robustness on the Scene data set, and the optimal value range of each parameter indicates that the constraint mapping space has a positive effect on the CRMFS algorithm.

4.7. Convergence and time complexity

In order to analyze the effective convergence of the CRMFS algorithm, although we have provided the theoretical proof of the convergence of the CRMFS algorithm in Subsection 3.4, we have conducted

a convergence analysis experiment for the CRMFS algorithm for more intuitive analysis.

The experiment was conducted by using six data sets: Emotion, Birds, Scene, Genbase, Image, and Enron. In the experiment, we set the value of all parameters in the CRMFS algorithm to equal 1. Additionally, we set the number of iterations to equal 50. We observed changes in the value of the objective function in each iteration. The experimental results are shown in Figure 9.



(a) Emotion (b) Birds (c) Scene

(d) Genbase (e) Image (f) Enron

Figure 9. Convergence of CRMFS algorithm on different data sets.

As shown in Figure 9, the CRMFS algorithm converges on all experimental data sets and proves the correctness of Section 3.4. In addition, for all experimental data sets, the convergence rate of the CRMFS algorithm was very fast, generally within ten iterations.

Finally, we performed time complexity analysis of the CRMFS algorithm. In general, on the multi-label data set $m < d$ and $m < n$. According to Table 1, in an iteration of the CRMFS algorithm, the time complexity of updating Q is $O(d^2n)$; the time complexity of updating M is $O(d^2)$. So the total time complexity of the CRMFS algorithm is $O(td^2n + td^2)$. It can be seen in Figure 9 that the CRMFS algorithm has a fast convergence speed and requires a small number of iterations. Therefore, it can be seen that the number of features d and the number of samples n of multi-label data sets significantly influence the time complexity of the CRMFS algorithm.

5. Conclusions and future research

Regarding the multi-label feature selection model based on linear mapping, we have developed a multi-label feature selection model (CRMFS) based on constrained mapping space to solve the problem of poor algorithm performance; it is possible because the duality of labels does not apply to linear mapping. The sample manifold and HSIC constrain this model's mapping space. Among them,

the sample manifold ensures the consistency of the mapping space and the topological structure of the sample space, and HSIC ensures high correlation between the mapping space and the label space. In addition, an optimization algorithm has been designed for the model, and the algorithm's convergence analysis, parameter sensitivity analysis, and time complexity analysis were performed. Finally, a comparative experiment comparing the CRMFS algorithm with seven advanced multi-label feature selection algorithms such as SSFS and MRDM, was conducted on ten classical multi-label data sets. Experimental results on five multi-label indexes prove the proposed algorithm's effectiveness and superiority.

In addition, it is known from the ablation studies that CRMFS has some limitations in terms of ability to manage feature redundancy or label redundancy. Therefore, in the next step of this research, we will address the redundancy problem in multi-label learning by using dynamic manifold learning and combining it with subspace learning to enhance the learning capability of the local sample structure and local label structure of the model in order to improve the generalization ability and overall performance of the model.

Use of AI tools declaration

The authors declare that they have not used artificial intelligence tools in the creation of this article.

Acknowledgments

This work was supported by the National Natural Science Foundation of China: Research on efficient cooperative intelligent optimization algorithm based on feature evaluation and dynamic complementarity (no. 12101477), Shaanxi Province Fund: Research on Super-multi-objective investment portfolio and algorithm based on fuzzy random theory (no. 2023-JC-YB-064).

Conflict of interest

The authors declare that there is no conflict of interest.

References

1. J. Gui, Z. N. Sun, S. W. Ji, D. C. Tao, T. N. Tan, Feature selection based on structured sparsity: A comprehensive study, *IEEE Trans. Neural Networks Learn. Syst.*, **28** (2016), 1–18. https://doi.org/10.1109/TNNLS.2016.2551724
2. M. Paniri, M. B. Dowlatshahi, H. Nezamabadi-Pour, MLACO: A multi-label feature selection algorithm based on ant colony optimization, *Knowl. Based Syst.*, **192** (2019), 105285. https://doi.org/10.1016/j.knosys.2019.105285
3. S. Kashef, H. Nezamabadi-Pour, B. Nikpour, Multi-label feature selection: A comprehensive review and guiding experiments, *Wiley Interdiscip. Rev. Data Min. Knowl. Discovery*, **8** (2018), 12–40. https://doi.org/10.1002/widm.1240
4. Y. Saeys, I. Inza, P. Larranaga, A review of feature selection techniques in bioinformatics, *Bioinformatics*, **23** (2007), 2507–2517. https://doi.org/10.1093/bioinformatics/btm344

5. C. C. Ding, M. Zhao, J. Lin, J. Y. Jiao, Multi-objective iterative optimization algorithm based optimal wavelet filter selection for multi-fault diagnosis of rolling element bearings, *ISA Trans.*, **82** (2019), 199–215. https://doi.org/10.1016/j.isatra.2018.12.010
6. M. Labani, P. Moradi, F. Ahmadizar, M. Jalili, A novel multivariate filter method for feature selection in text classification problems, *Eng. Appl. Artif. Intell.*, **70** (2018), 25–37. https://doi.org/10.1016/j.engappai.2017.12.014
7. C. Yao, Y. F. Liu, B. Jiang, J. G. Han, J. W. Han, LLE score: A new filter-based unsupervised feature selection method based on nonlinear manifold embedding and its application to image recognition, *IEEE Trans. Image Process.*, **26** (2017), 5257–5269. https://doi.org/10.1109/TIP.2017.2733200
8. J. González, J. Ortega, M. Damas, P. Martín-Smith, J. Q. Gan, A new multi-objective wrapper method for feature selection–Accuracy and stability analysis for BCI, *Neurocomputing*, **333** (2019), 407–418. https://doi.org/10.1016/j.neucom.2019.01.017
9. J. Swati, H. Hongmei, J. Karl, Information gain directed genetic algorithm wrapper feature selection for credit rating, *Appl. Soft Comput.*, **69** (2018), 541–553. https://doi.org/10.1016/j.asoc.2018.04.033
10. S. Maldonado, J. López, Dealing with high-dimensional class-imbalanced datasets: Embedded feature selection for SVM classification, *Appl. Soft Comput.*, **67** (2018), 94–105. https://doi.org/10.1016/j.asoc.2018.02.051
11. Y. C. Kong, T. W. Yu, A graph-embedded deep feedforward network for disease outcome classification and feature selection using gene expression data, *Bioinformatics*, **34** (2018), 3727–3737. https://doi.org/10.1093/bioinformatics/bty429
12. Y. Zhang, Y. C. Ma, X. F. Yang, Multi-label feature selection based on logistic regression and manifold learning, *Appl. Intell.*, **2022** (2022), 1–18. https://doi.org/10.1007/s10489-021-03008-8
13. S. Liaghat, E. G. Mansoori, Filter-based unsupervised feature selection using Hilbert—Schmidt independence criterion, *Int. J. Mach. Learn. Cybern.*, **10** (2019), 2313–2328. https://doi.org/10.1007/s13042-018-0869-7
14. J. Lee, D. W. Kim, SCLS: Multi-label feature selection based on scalable criterion for large label set, *Pattern Recognit.*, **66** (2017), 342–352. https://doi.org/10.1016/j.patcog.2017.01.014
15. Y. J. Lin, Q. H. Hu, J. H. Liu, J. Duan, Multi-label feature selection based on maxdependency and min-redundancy, *Neurocomputing*, **168** (2015), 92–103. https://doi.org/10.1016/j.neucom.2015.06.010
16. J. Lee, D. W. Kim, Feature selection for multi-label classification using multivariate mutual information, *Pattern Recognit. Lett.*, **34** (2013), 349–357. https://doi.org/10.1016/j.patrec.2012.10.005
17. J. Lee, D. W. Kim, Fast multi-label feature selection based on information-theoretic feature ranking, *Pattern Recognit.*, **48** (2015), 2761–2771. https://doi.org/10.1016/j.patcog.2015.04.009
18. W. F. Gao, L. Hu, P. Zhang, Class-specific mutual information variation for feature selection, *Pattern Recognit.*, **79** (2018), 328–339. https://doi.org/10.1016/j.patcog.2018.02.020

19. J. Lee, D. W. Kim, Scalable multi-label learning based on feature and label dimensionality reduction, *Complexity*, **23** (2018), 1–15. https://doi.org/10.1155/2018/6292143

20. P. Zhang, W. F. Gao, J. C. Hu, Y. H. Li, Multi-label feature selection based on high-order label correlation assumption, *Entropy*, **22** (2020), 797. https://doi.org/10.3390/e22070797

21. W. F. Gao, P. T. Hao, Y. Wu, P. Zhang, A unified low-order information-theoretic feature selection framework for multi-label learning, *Pattern Recognit.*, **134** (2023), 109111. https://doi.org/10.1016/j.patcog.2022.109111

22. Y. H. Li, L. Hu, W. F. Gao, Multi-label feature selection via robust flexible sparse regularization, *Pattern Recognit.*, **134** (2023), 109074. https://doi.org/10.1016/j.patcog.2022.109074

23. Y. H. Li, L. Hu, W. F. Gao, Multi-label feature selection with high-sparse personalized and low-redundancy shared common features, *Inf. Process. Manage.*, **61** (2024), 103633. https://doi.org/10.1016/j.ipm.2023.103633

24. X. C. Hu, Y. H. Shen, W. Pedrycz, X. M. Wang, A. Gacek, B. S. Liu, Identification of fuzzy rule-based models with collaborative fuzzy clustering, *IEEE Trans. Cybern.*, **2021** (2021), 1–14. https://doi.org/10.1109/TCYB.2021.3069783

25. K. Y. Liu, X. B. Yang, H. Fujita, D. Liu, X. Yang, Y. H. Qian, An efficient selector for multi-granularity attribute reduction, *Inf. Sci.*, **505** (2019), 457–472. https://doi.org/10.1016/j.ins.2019.07.051

26. Y. Chen, K. Y. Liu, J. J. Song, H. Fujita, X. B. Yang, Y. H. Qian, Attribute group for attribute reduction, *Inf. Sci.*, **535** (2020), 64–80. https://doi.org/10.1016/j.ins.2020.05.010

27. Y. G. Jing, T. R. Li, H. Fujita, Z. Yu, B. Wang, An incremental attribute reduction approach based on knowledge granularity with a multi-granulation view, *Inf. Sci.*, **411** (2017), 23–38. https://doi.org/10.1016/j.ins.2017.05.003

28. J. Zhang, Z. M. Luo, C. D. Li, C. G. Zhou, S. Z. Li, Manifold regularized discriminative feature selection for multi-label learning, *Pattern Recognit.*, **95** (2019), 136–150. https://doi.org/10.1016/j.patcog.2019.06.003

29. R. Huang, Z. Wu, S. W. Ji, D. C. Tao, T. N. Tan, Multi-label feature selection via manifold regularization and dependence maximization, *Pattern Recognit.*, **120** (2021), 180149. https://doi.org/10.1016/j.patcog.2021.108149

30. L. Hu, Y. H. Li, W. F. Gao, P. Zhang, J. C. Hu, Multi-label feature selection with shared common mode, *Pattern Recognit.*, **104** (2020), 107344. https://doi.org/10.1016/j.patcog.2020.107344

31. W. F. Gao, Y. H. Li, L. Hu, Multi-label feature selection with constrained latent structure shared term, *IEEE Trans. Neural Networks Learn. Syst.*, **34** (2023), 1253–1262. https://doi.org/10.1109/TNNLS.2021.3105142

32. Y. Zhang, Y. C. Ma, Non-negative multi-label feature selection with dynamic graph constraints, *Knowl. Based Syst.*, **238** (2022), 107924. https://doi.org/10.1016/j.knosys.2021.107924

33. F. P. Nie, H. Huang, X. Cai, C. Ding, Efficient and robust feature selection via joint $L_{2,1}$-norms minimization, *Adv. Neural Inf. Process. Syst.*, **2010** (2010), 1813–1821.

34. A. Hashemi, M. Dowlatshahi, H. Nezamabadi-pour, MFS-MCDM: Multi-label feature selection using multi-criteria decision making, *Knowl. Based Syst.*, **206** (2020), 106365. https://doi.org/10.1016/j.knosys.2020.106365

35. M. L. Zhang, Z. H. Zhou, ML-KNN: A lazy learning approach to multi-label learning, *Pattern Recognit.*, **40** (2007), 2038–2048. https://doi.org/10.1016/j.patcog.2006.12.019

36. J. Dougherty, R. Kohavi, M. Sahami, Supervised and unsupervised discretization of continuous features, *Mach. Learn. Proc.*, **1995** (1995), 194–202. https://doi.org/10.1016/B978-1-55860-377-6.50032-3

37. O. J. Dunn, Multiple Comparisons among Means, *J. Am. Stat. Assoc.*, **56** (1961), 52–64. https://doi.org/10.1080/01621459.1961.10482090

38. M. Friedman, A comparison of alternative tests of significance for the problem of m rankings, *Ann. Math. Stat.*, **11** (1940), 86–92. https://doi.org/10.1214/aoms/1177731944

AIMS Press