 doi:10.20944/preprints202107.0341.v1

Article

Multi-label feature selection based on logistic regression and manifold learning

Yao Zhang, Yingcang Ma, Xiaofei Yang

School of Science, Xi'an Polytechnic University, Xian 710048, China;
* Correspondence: mayingcang@126.com

Abstract: Like traditional single label learning, multi-label learning is also faced with the problem of dimensional disaster.Feature selection is an effective technique for dimensionality reduction and learning efficiency improvement of high-dimensional data. In this paper, Logistic regression, manifold learning and sparse regularization were combined to construct a joint framework for multi-label feature selection (LMFS). Firstly, the sparsity of the eigenweight matrix is constrained by the $L_{2,1}$-norm. Secondly, the feature manifold and label manifold can constrain the feature weight matrix to make it fit the data information and label information better. An iterative updating algorithm is designed and the convergence of the algorithm is proved.Finally, the LMFS algorithm is compared with DRMFS, SCLS and other algorithms on eight classical multi-label data sets. The experimental results show the effectiveness of LMFS algorithm.

Keywords: feature selection; manifold learning; multi-label learning; $L_{2,1}$-norm; logistic regression

0. Introduction

In recent years, with the rapid development of the Internet and digital acquisition equipment, the scale of data that needs to be analyzed and processed in classification problems has increased dramatically. These data may contain not only single label data, but also a large number of multi-label data. In the single label data, each instance has only one label, and different labels are mutually independent. While in the multi-label data, a sample may belong to multiple labels at the same time, and each label not only intersects with each other but also is correlated. So far, the research of multi-label learning, which includes text classification, image annotation, video classification, biology, etc., has attracted the attention of many scholars .In many practical applications mentioned above, multi-label data usually has thousands or even more features, which brings many problems to data analysis, decision-making, screening and prediction [1]. For example, redundant and irrelevant features may affect the function of classifiers. In order to solve these problems, we will select a subset of related and optimal features. The procedure is called feature selection. Feature selection has many advantages in learning algorithm, including reducing measurement cost and storage requirements, shortening training time, avoiding dimension disaster, reducing over fitting and so on [2,3]. Therefore, multi-label feature selection has become a research hot spot.

Based on label information and search strategy, feature selection methods are usually divided into two categories [4]. Based on the search strategy, feature selection can be divided into three categories: filter [5–7], wrapper [8,9] and embedded [10,11]. Among them, the embedded method combines the advantages of filter method and wrapper method. They embed the feature selection process in the learning process. Because they do not evaluate the feature subset iteratively, they are more effective than the wrapper method [1].

A multi-label feature selection method based on mutual information and label correlation is proposed [12]. A new multi-label feature selection based on label redundancy, called (LRFS), is

 doi:10.20944/preprints202107.0341.v1

proposed [13]. It divides labels into independent labels and dependent labels, and analyzes the differences between independent labels and dependent labels. To measure the consistency between feature space and label space, kernel alignment is introduced into multi-label learning. And a new multi-label feature selection method, which can automatically learn and deal with the importance of labels, has been developed [14]. A new feature selection method of extended adaptive minimum absolute contraction selection operator (EALasso) is presented [15]. This method preserves the properties of determining the correct subset model and obtaining the optimal estimation accuracy, proposes an iterative optimization algorithm, and gives the theoretical proof of convergence. Some researchers put forward a multi-label feature selection method with multiple regularization (MDFS) [16]. And they calculate the correlation between the feature and the local label, and use the objective function that includes L_{21}-norm regularization. According to the research of subspace learning, a multi-label feature selection method based on non-negative sparse representation is proposed [17]. This method can be regarded as matrix decomposition problem, in which nonnegative constraint problem and L_{21}-norm minimization problem are integrated. And it also designs an efficient matrix updating iterative algorithm to solve these problems.

To sum up, linear regression is often used in multi-label feature selection model. However, in most cases, multi-label feature selection is used for multi-label classification. So from the perspective of classification, logical regression is more suitable for multi-label feature selection model. The reasons are as follows:

1) For multi-label data, label (dependent variable) is discrete value (0 or 1), which is more suitable for logistic regression.
2) Logistic regression is a generalized linear model, which is equivalent to introducing nonlinearity into the model, which can improve the expression ability of the model and increase the fitting.
3) Linear regression is to directly analyze the relationship between dependent variable and independent variable, while logistic regression is to analyze the relationship between the probability of taking a certain value of dependent variable and independent variable.

From reasons 1 and 2, it can be seen that logistic regression is more suitable for data classification than linear regression, and it can be applied to a variety of data distribution including the distribution of the positive and the negative; from reason 3, it can be seen that logistic regression is more robust than linear regression.

Based on this problem, some scholars choose logistic regression to replace the least square regression in the model and try to improve the function of the algorithm by improving the regular term. The author puts foward a correlation logistic regression model (CorrLog) for multi-label image classification, which extends the traditional logistic regression model to multi-label image classification [18]. A feature subset selection algorithm for mixed integer optimal logistic regression is proposed [19]. This paper presents a mixed integer linear optimization problem, which can be solved by using standard integer optimization software to approximate the logistic loss function piecewise. A robust logistic regression method based on the regularization of L_q-norm $q \in [0, 1]$ is proposed [20], which is a feasible and effective feature selection method.

However, the existing multi-label feature selection algorithm based on logistic regression ignores the feature manifold structure, and the above multi-label feature selection algorithm $[1-3, 11-13]$ ignores the fitting of label information while paying attention to the feature manifold structure, the above multi-label feature selection algorithm $[14, 16, 17]$ ignores the fitting of the feature manifold structure to label information while paying attention. Therefore, this study will combine the logistic regression model with the regularization of feature map, label map and L_{21}-norm sparse regularization to solve the problem of multi-label feature selection.

The rest of this paper is organized as follows. Section 2 gives multi-label feature selection model. In Section 3, the model is solved and an iterative algorithm for multi-label feature selection is proposed, and its time complexity is analyzed. In Section 4, the comprehensive experiment on six classical data

 doi:10.20944/preprints202107.0341.v1

sets shows that the algorithm proposed in this paper is superior to other algorithms. Finally, the conclusions and future work is presented in Section 5.

1. Problem description

1.1. Logistic regression model

Suppose a multi-label data set $D = \{(d_i, y_i)\}_{i=1}^{n}$ consists of n independent samples with the same distribution. Let $X = [x_1; x_2; \cdots; x_n]$ be the augmented matrix of the data matrix, $X \in R^{n \times d}$; $Y = [y_1; y_2; \cdots; y_n]$ be the label matrix, $Y \in R^{n \times m}$. Where $x_i = [1, d_i]$; The value of y_{ij} is 0 or 1, indicating whether the i-th sample is associated with the j-th class. In the logistic regression, the posterior probability that sample x_i belongs to the j-th class is:

$$Pr(y_{ij} = 1|x_i) = g(x_i w_j) = \frac{exp(x_i w_j)}{1 + exp(x_i w_j)} \tag{1}$$

Thus the posterior probability that sample x_i does not belong to the j-th class is:

$$Pr(y_{ij} = 0|x_i) = 1 - g(x_i w_j) = \frac{1}{1 + exp(x_i w_j)} \tag{2}$$

where $W = [w_1, w_2, \cdots, w_m]$ and $W \in R^{d \times m}$; w_j is the j-th column vector of the coefficient matrix W.

If the maximum likelihood estimation method is used to estimate the coefficient matrix, then the likelihood function (joint probability distribution) of the logistic regression on the multi-label data set is:

$$P(W) = \prod_{j=1}^{m} \prod_{i=1}^{n} g(x_i w_j)^{y_{ij}} (1 - g(x_i w_j))^{1 - y_{ij}} \tag{3}$$

Since it is inconvenient to solve optimization max $P(W)$, the minimum value of $L(W)$ of negative log likelihood function for solving logistic regression is used to solve W

$$\begin{aligned} L(W) &= -\sum_{j=1}^{m} \sum_{i=1}^{n} [y_{ij} \ln(g(x_i w_j)) + (1 - y_{ij}) \ln(1 - g(x_i w_j))] \\ &= -\sum_{j=1}^{m} \sum_{i=1}^{n} [y_{ij} x_i w_j + \ln(1 - g(x_i w_j))] \end{aligned} \tag{4}$$

1.2. Sparse constraint

The logistic regression model may suffer from ill-posed problems, such as over fitting, multi-collinearity, and infinite solutions, which results in incorrect estimation of the coefficient matrix [21] So in order to solve this problem, a widely used strategy is to introduce penalty terms into $L(W)$, which aims to achieve a stable and accurate logistic regression model in high-dimensional data. The so-called penalty function is usually expressed as follows, where β is the regularization parameter.

$$min_W L(W) + \beta R(W) \tag{5}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

For the i-th row vector W_i of the coefficient matrix W, it can be regarded as a vector that measures the importance of the i-th feature. Let $f_i \in R^n$ be the i-th feature vector of the data matrix, and then the data matrix X can be expressed in the form of $X = [f_1, f_2, \cdots, f_d]$.

The regular term $R(W)$ has various forms for different purposes. Take L_1-norm regular term and L_2-norm regular term for example. L_1-norm is often used to guided sparsity; L_2-norm is often used to guided stability. Because $\|W_i\|_2$ is generally used to measure the importance of feature f_i, in order to better distinguish the importance of features, here we take L_{21}-norm as the regular term $R(W)$, which not only guides the row sparsity of the sparse matrix, but also is sensitive to singular values [22]. So the objective optimization problem can be written as:

$$min_W L(W) + \frac{\beta}{2}\|W\|_{2,1} \tag{6}$$

where, $\|W\|_{2,1} = \sum_{i=1}^{d} \|W_i\|_2$.

1.3. Feature manifold learning

Considering that the parameter of each coefficient vector w_j in formula (6) is β, but according to the idea of binary conversion, the regularization parameter β may not be applicable to all coefficient vectors. In addition, the features are extracted from some manifolds called the feature manifold [23]. This is an important technique that can obtain the structure of the feature weight feature manifold by exploring the geometry. According to the problem assumption, if the features f_i and f_j are closer, then their weight vectors W_i and W_j should also be closer. Therefore, a feature map regularization is constructed, which can adjust the regularization parameters of the coefficient vectors w_j according to the similarity between the features f_i and f_j. Its expression is as follows:

$$\begin{aligned}
&\frac{1}{2}\sum_{i=1}^{d}\sum_{j=1}^{d}\|W_i - W_j\|_2^2 S_{ij} \\
&= \frac{1}{2}\sum_{i=1}^{d}\sum_{j=1}^{d}(W_i - W_j)(W_i - W_j)^T S_{ij} \\
&= \sum_{i=1}^{d} W_i W_i^T M_{ii} - \sum_{i=1}^{d}\sum_{j=1}^{d} W_i W_j^T S_{ij} \\
&= Tr(W^T(M - S)W) \\
&= Tr(W^T L_S W)
\end{aligned} \tag{7}$$

where, $M \in R^{d\times d}$ is the diagonal matrix, and $M_{ii} = \sum_{j=1}^{d} S_{ij}$ is the i-th diagonal element of M. $L_S = M - S$ is the Laplacian matrix of the feature similarity matrix S, and S_{ij} is the i-th row and j-th element of the feature similarity matrix S, representing the similarity between features f_i and f_j. There are many ways to construct feature similarity matrix S,for example:

By using a kernel function, a feature association matrix S can be constructed, where $t \in R$:

$$S_{ij} = \begin{cases} exp(-\frac{\|f_i - f_j\|_2^2}{t}), & if \;\; f_i \in N_K(f_j) \; or \; f_j \in N_K(f_i) \\ 0, & others \end{cases} \tag{8}$$

where $N_K(*)$ represents the k-nearest neighbor set of $*$. Through feature map regularization, the problem of feature selection is optimized:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

$$min_W L(W) + \frac{\lambda}{2} Tr(W^T L_S W) + \frac{\beta}{2} \|W\|_{2,1} \tag{9}$$

1.4. Label manifold learning

in order to better fit the label information while fitting the manifold structure. According to the problem, suppose: Let $f(x_i W) = [g(x_i w_1), g(x_i w_2), \cdots, g(x_i w_m)]$, $f(x_i W) \in R^m$, if the labels y_i and y_j are closer, then the probability $f(x_i W)$ and $f(x_j W)$ in the logistic regression model should also be closer, and according to the positive correlation between $g(x_i w_j)$ and $x_i w_j$, the positive correlation between $f(x_i W)$ and $x_i W$ is deduced, thus $x_i W$ should be closer to $x_j W$. Therefore, a regularization of the label graph is constructed, which can adjust the coefficient matrix W according to the similarity between the labels y_i and y_j, so that W can better fit the label information. Its expression is as follows:

$$\begin{aligned}
&\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \|x_i W - x_j W\|_2^2 A_{ij} \\
&= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} (x_i W - x_j W)(x_i W - x_j W)^T A_{ij} \\
&= \sum_{i=1}^{n} x_i W (x_i W)^T P_{ii} - \sum_{i=1}^{n} \sum_{j=1}^{n} x_i W (x_j W)^T A_{ij} \\
&= Tr(W^T X^T (P - A) X W) \\
&= Tr(W^T X^T L_A X W) \\
&= Tr(W^T \overline{L_A} W)
\end{aligned} \tag{10}$$

where, $\overline{L_A} = X^T L_A X$ and $P \in R^{n \times n}$ are diagonal matrices, $P_{ii} = \sum_{j=1}^{n} A_{ij}$ is the i-th diagonal element of P. $L_A = P - A$ is the Laplacian matrix of label similarity matrix A, and A_{ij} is the element of the i-th row and the j-th column of label similarity matrix A, representing the similarity between the labels y_i and y_j. The label similarity matrix A can be given by many methods,as follows:

By using a kernel function, a label association matrix A can be constructed, where $t \in R$:

$$A_{ij} = \begin{cases} exp(-\frac{\|y_i - y_j\|_2^2}{t}), & if \ \ y_i \in N_K(y_j) \ or \ y_j \in N_K(y_i) \\ 0, & others \end{cases} \tag{11}$$

As for several different calculation methods of feature similarity matrix S and label similarity matrix A, and the impact on multi-label feature selection, we have made a simple analysis on the Image and Emotion data sets, set the parameter range as $[0.001, 0.01, 0.1, 1, 10, 100, 1000]$ to search and get the best result. As shown in the Figure 1 below, and found that several methods are similar. So in the experiment part, we use kernel function to learn the feature similarity matrix and label similarity matrix.

Through label map regularization, the optimization feature selection problem is transformed into:

$$min_W L(W) + \frac{\lambda}{2} Tr(W^T L_S W) + \frac{\beta}{2} \|W\|_{2,1} + \frac{\gamma}{2} Tr(W^T \overline{L_A} W) \tag{12}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

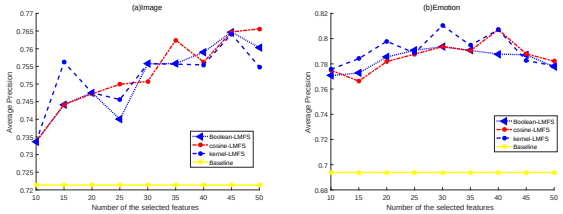

Figure 1. Average precision comparison of different similarity matrix methods when ML-KNN is used as the basic classifier. (the higher the result, the better)

2. Problem solving and proof of convergence

2.1. Problem solving

Due to the non-smoothness of L_{21}-norm, it is difficult to find the closed solution of the optimization problem in (12) directly. According to [22], this problem can be solved by another method. When $W_i \neq 0 (i = 1, 2, \cdots, d)$, the derivative of $\|W\|_{2,1}$ to W is:

$$\frac{\partial(\|W\|_{2,1})}{\partial W} = 2HW \tag{13}$$

where $H \in R^{d \times d}$ is the diagonal matrix and the i-th diagonal element of H is:

$$H_{ii} = \frac{1}{2\|W_i\|_2} \tag{14}$$

Therefore, the derivative in L_{21}-norm can also be regarded as the derivative of $Tr(W^T HW)$. Since $\|W\|_{2,1}$ is convex, the optimization problem of L_{21}-norm can be used to find the approximate solution of (12). Thus the objective function is transformed into :

$$\begin{aligned} obj(W) &= L(W) + \frac{\lambda}{2} Tr(W^T L_S W) \\ &+ \frac{\beta}{2} Tr(W^T HW) + \frac{\gamma}{2} Tr(W^T \overline{L_A} W) \end{aligned} \tag{15}$$

For this problem, we can give an H, calculate W with the current H, and then update H based on the currently calculated W.

Since (15) is differentiable, it can be solved by the Newton–Raphson algorithm. The first derivative of (15) to W is:

$$\begin{aligned} \frac{\partial(obj(W))}{\partial W} &= -X^T[Y - G(XW)] \\ &+ \lambda L_S W + \beta HW + \gamma \overline{L_A} W \end{aligned} \tag{16}$$

where $G(XW) = [f(x_1 W); f(x_2 W); \cdots; f(x_n W)]$. The second derivative of (15) to W is:

$$\frac{\partial^2(obj(W))}{\partial W \partial W^T} = -X^T UX + \lambda L_S + \beta H + \gamma \overline{L_A} \tag{17}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

Among them:

$$U = diag \sum_{j=1}^{m} [(1 - g(x_i w_j))g(x_i w_j)] \tag{18}$$

where $i = 1, 2, \cdots, d$.

The updated formula for W is:

$$W^{t+1} = W^t - (\frac{\partial^2(obj(W))}{\partial W \partial W^T})^{-1} \frac{\partial(obj(W))}{\partial W} \tag{19}$$

Algorithm 1: LMFS

Input: The data matrix $X \in R^{n \times d}$, the label matrix $Y \in R^{n \times m}$, three regularization parameters λ, β and γ, and the number of the selected features k.

1. Calculate the feature similarity matrix S according to (8), and calculate $L = M - S$.
2. Calculate the label similarity matrix A according to (9), and calculate $L_A = P - A$ and $\overline{L_A} = X^T L_A X$.
3. Initialization matrix H as the unit array and coefficient matrix W as a matrix whose elements are all 0.
4. Repeat:
   (a) Calculate the loss function value according to (12).
   (b) Update U according to (18).
   (c) Calculate the first derivative function and second derivative function of (15) according to (16) and (17).
   (d) Update W according to (19).
   (e) Update H according to (14).
5. Until convergence criterion has been satisfied.
6. Calculate and rank $\|W_i\|_2 (i = 1, 2, \cdots, d)$ to find the top k largest assignments to I.

Output: The feature selection result I.

In LMFS algorithm, the main purpose is to calculate the update W. the time complexity of each iteration is $O(d^2 n)$, and the LMFS algorithm has iterated t times in total. Therefore, the total time complexity of LMFS algorithm is $O(td^2 n)$, and the value of t is not large. Therefore, the running time of LMFS algorithm processing data is greatly affected by the dimension d of data and the number of samples n in the data set.

2.2. Proof of convergence

In this section, we prove that the iterative procedure shown in Algorithm 1 is convergent. Therefore, in the t-th iteration, we know:

$$\begin{aligned} W^{t+1} &= argmin_W L(W) + \frac{\lambda}{2} Tr(W^T L_S W) \\ &+ \frac{\beta}{2} Tr(W^T H^t W) + \frac{\gamma}{2} Tr(W^T \overline{L_A} W) \end{aligned} \tag{20}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

where $H_{ii}^t = \frac{1}{2\|W_i^t\|_2} (i = 1, 2, \cdots, d)$, so we have:

$$\begin{aligned} & L(W^{t+1}) + \frac{\lambda}{2} Tr((W^{t+1})^T L_S W^{t+1}) \\ & + \frac{\beta}{2} Tr((W^{t+1})^T H^t W^{t+1}) + \frac{\gamma}{2} Tr((W^{t+1})^T \overline{L_A} W^{t+1}) \\ & \leq L(W^t) + \frac{\lambda}{2} Tr((W^t)^T L_S W^t) \\ & + \frac{\beta}{2} Tr((W^t)^T H^t W^t) + \frac{\gamma}{2} Tr((W^t)^T \overline{L_A} W^t) \end{aligned} \tag{21}$$

That is:

$$\begin{aligned} & L(W^{t+1}) + \frac{\lambda}{2} Tr((W^{t+1})^T L_S W^{t+1}) \\ & + \frac{\gamma}{2} Tr((W^{t+1})^T \overline{L_A} W^{t+1}) + \frac{\beta}{2} \sum_{i=1}^{d} \frac{\|W_i^{t+1}\|_2^2}{2\|W_i^t\|_2} \\ & \leq L(W^t) + \frac{\lambda}{2} Tr((W^t)^T L_S W^t) \\ & + \frac{\gamma}{2} Tr((W^t)^T \overline{L_A} W^t) + \frac{\beta}{2} \sum_{i=1}^{d} \frac{\|W_i^t\|_2^2}{2\|W_i^t\|_2} \end{aligned} \tag{22}$$

It can be further transformed into:

$$\begin{aligned} & L(W^{t+1}) + \frac{\lambda}{2} Tr((W^{t+1})^T L_S W^{t+1}) \\ & + \frac{\gamma}{2} Tr((W^{t+1})^T \overline{L_A} W^{t+1}) \\ & + \frac{\beta}{2} \|W^{t+1}\|_{2,1} - \frac{\beta}{2} (\|W^{t+1}\|_{2,1} - \sum_{i=1}^{d} \frac{\|W_i^{t+1}\|_2^2}{2\|W_i^t\|_2}) \\ & \leq L(W^t) + \frac{\lambda}{2} Tr((W^t)^T L_S W^t) \\ & + \frac{\gamma}{2} Tr((W^t)^T \overline{L_A} W^t) \\ & + \frac{\beta}{2} \|W^t\|_{2,1} - \frac{\beta}{2} (\|W^t\|_{2,1} - \sum_{i=1}^{d} \frac{\|W_i^t\|_2^2}{2\|W_i^t\|_2}) \end{aligned} \tag{23}$$

According to the inequality $\sqrt{a} - \frac{a}{2\sqrt{b}} \leq \sqrt{b} - \frac{b}{2\sqrt{b}}$ for any positive numbers a and b, we have:

$$\|W_i^{t+1}\|_{2,1} - \frac{\|W_i^{t+1}\|_2^2}{2\|W_i^t\|_2} \leq \|W_i^t\|_{2,1} - \frac{\|W_i^t\|_2^2}{2\|W_i^t\|_2} \tag{24}$$

Which sums, we get:

$$\sum_{i=1}^{d} (\|W_i^{t+1}\|_{2,1} - \frac{\|W_i^{t+1}\|_2^2}{2\|W_i^t\|_2}) \leq \sum_{i=1}^{d} (\|W_i^t\|_{2,1} - \frac{\|W_i^t\|_2^2}{2\|W_i^t\|_2}) \tag{25}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

Which implies:

$$\|W^{t+1}\|_{2,1} - \sum_{i=1}^{d} \frac{\|W_i^{t+1}\|_2^2}{2\|W_i^t\|_2} \leq \|W^t\|_{2,1} - \sum_{i=1}^{d} \frac{\|W_i^t\|_2^2}{2\|W_i^t\|_2} \tag{26}$$

In summary, the convergence of Algorithm 1 is proved.

3. Experiments and results

In order to verify the effectiveness of the LMFS algorithm, the experiment uses eight public data sets and compares its performance with some of the most advanced methods and baselines. At the same time, the experiment selects ML-KNN[24] as the representative of the multi-label classification algorithm for evaluation.

3.1. Dataset and experimental setup

The experiment uses eight public data sets from four different areas. The specific parameters of each data set are shown in Table 1:

Table 1. Dataset information

data	sample	features	label	training	test	species
Image	600	294	5	400	200	image
Emotion	593	72	6	391	202	music
Enron	1702	1001	53	1123	579	text
Business	5000	438	30	2000	3000	text
Computers	5000	681	33	2000	3000	text
Health	5000	612	32	2000	3000	text
Scene	2407	294	6	1211	1196	image
Yeast	2417	103	14	1500	917	biological

In terms of experimental environment, all experimental related environments are: Microsoft Windows7 system, processor: Intel (R) Core (TM) i5-4210U CUP @ 1.70GHz 2.40GHz, memory: 4.00GB, programming software: Matlab R2016a .

To verify the effectiveness of the proposed feature selection method, the following most advanced state-of-the-art feature selection algorithms are compared:

1) Baseline: The results on various evaluation indicators after learning the data set directly with ML-KNN without any feature selection.
2) DRMFS[25]: A robust multi-label feature selection with dual-graph regularization was constructed by using feature graph and label graph to guide the sparsity between rows and within rows of the weight matrix, and $L_{2,1}$ norm to guide its global properties and robust.
3) SCLS[26]: Multi-label feature selection method based on scalable standards.
4) MDMR[27]: Multi-label feature selection through an evaluation metric that combines mutual information with maximum dependency and minimum redundancy.
5) PMU[28]: A multi-label feature selection algorithm based on mutual information. Multi-label feature selection is performed by selecting the dependency between the selected feature and the label.
6) FIMF[29]: A fast multi-label feature selection method based on information theory feature ranking. Based on information theory, a scoring function that evaluates the importance of features is derived, and its calculation cost is analyzed.

In order to ensure the fairness of the experiment, in terms of parameter setting: In the experiment, the number of nearest neighbors K for the multi-label classification algorithm ML-KNN is set to

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

10, and the value of smooth S is set to 1. For MDMR, PMU, and FIMF, we discretize the data set using the equal-width intervals [30]. For FIMF, we set $Q = 10$ and $b = 2$. For DRMFS and LMFS we use (8) to calculate the similarity matrix between features. The above settings are the default settings of the algorithms. In addition, For DRMFS and other comparative algorithms, the experiment adjusts the regularization parameters of all methods by the "grid search" strategy from $[0.001, 0.01, 0.1, 1, 10, 100, 1000]$. For the feature dimension, we set the number of selected features as $[10, 15, 20, 25, 30, 35, 40, 45, 50]$. The maximum number of iterations for all iterative algorithms is fixed as 50. At the same time, the size of neighborhood K is set as 5. For all multi-label feature selection algorithms, the experiments show the best results from the optimal parameters.

3.2. Evaluation metrics

The performance evaluation of the multi-label learning systems is different from the single-label learning systems. The evaluation criteria of the multi-label learning system are more complicated. The experiment uses five evaluation criteria, including Hamming loss, Ranking loss, One-error, Coverage, and Average precision in ML-KNN. The specific contents of the five evaluation criteria are as follows:

Suppose there is a test data set $D = \{(x_i, y_i)\}_{i=1}^{n}$, where n is the number of test samples. Given test sample x_i, the binary label vector that is predicted by the multi-label classifier is denoted as $h(x_i)$, and the rank of the l-th label prediction is dennoted as $rank_i(l)$.

1) Hamming loss: evaluates the percentage of mislabeled labels, i.e., a label belonging to the instance is not predicted or a label not belonging to the instance is predicted. The smaller the value, the better the performance.

$$HL(D) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{m}\|h(x_i) \triangle y_i\|_1 \tag{27}$$

where $\triangle$ represents the symmetric difference between the two sets, and returns those values that appear only in one of the sets, $HL(D) \in [0, 1]$.

2) Ranking loss: evaluates the proportion of reverse-order label pairs, that is, the case where unrelated labels are more relevant than related labels. The smaller the value, the better the performance.

$$RL(D) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{1_m^T y_i 1_m^T \overline{y_i}}\sum_{l:y_i^l=1}\sum_{l':y_i^{l'}=0}(\delta(rank_i(l) \geq rank_i(l^{'}))) \tag{28}$$

where $\overline{y_i}$ is the complement of y_i in Y. $RL(D) \in [0, 1]$.

3) One-error: evaluates the proportion of samples that "the most relevant label is not" in "real labels". The smaller the value, the better the performance.

$$OE(D) = \frac{1}{n}\sum_{i=1}^{n}\delta(y_i^{l_i} = 0) \tag{29}$$

where $l_i = argmin_{l\in[1,m]}rank_i(l)$ and B are indicator functions, $OE(D) \in [0, 1]$.

4) Coverage: evaluates how many steps the "sorted label list" needs to move, on the average, to cover the true related label set. The smaller the value, the better the performance.

$$CV(D) = \frac{1}{n}\sum_{i=1}^{n}argmax_{l:y_i^l=1}rank_i(l) - 1 \tag{30}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

where $CV(D) \in [1, m-1]$.

5) Average precision: evaluates the proportion of those labels that are more relevant than particular labels. The larger the value, the better the performance.

$$AP(D) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{1_m^T y_i}\sum_{l:y_i^l=1}\frac{prec_i(l)}{rank_i(l)} \tag{31}$$

where $prec_i(l) = \sum_{l':y_i^{l'}=1}\delta(rank_i(l) \geq rank_i(l'))$ and $AP(D) \in [0,1]$.

3.3. Experimental results

The proposed multi-label feature selection algorithm has been tested in six public data sets with extensive experiments. Comparing with several state-of-the-art algorithms, we consider evaluation metrics of Hamming loss, Ranking loss, One-error, Coverage, and Average precision to evaluate the performance of the above multi-label feature selection methods. Table 2 to Table 6 show the best results of all the feature selection methods from the optimal parameters. In these tables, the best performance is indicated in bold font, and the second best performance is underlined. Table 2 to Table 6 report Hamming loss, Ranking loss, One-error, Coverage, and Average precision comparison of different algorithms in each data set.

First of all, feature selection is effective. It not only reduces the number of features, and shortens the running time of the classifier, but also improves the performance of the classification algorithm. Secondly, as can be seen from Table 2 to Table 6, although the performance of LMFS algorithm on data sets Business, Scene and Yeast is slightly inadequate, the performance of LMFS algorithm on data sets Business and Scene is second only to the Baseline. In addition, LMFS algorithm has the best performance on other data sets.

Table 2. Hamming loss comparison of different algorithms under each data set

algorithms	LMFS	DRMFS	SCLS	MDMR	PMU	FIMF	Baseline
Business	**0.0264**	0.0280	0.0274	0.0273	0.0284	0.0274	<u>0.0269</u>
Image	**0.1950**	<u>0.2020</u>	0.2110	0.2240	0.2270	0.2340	0.2130
Emotion	**0.2063**	<u>0.2244</u>	0.2500	0.2409	0.2673	0.2252	0.2937
Health	**0.0370**	0.0391	<u>0.0389</u>	0.0391	0.0457	0.0407	0.0458
Computers	**0.0381**	<u>0.0393</u>	0.0398	0.0398	0.0416	0.0409	0.0412
Enron	<u>0.0486</u>	**0.0478**	0.0495	0.0505	0.0505	0.0501	0.0520
Scene	<u>0.1006</u>	0.1126	0.1073	0.1348	0.1137	0.1587	**0.0989**
Yeast	<u>0.1943</u>	**0.1938**	0.2006	0.1999	0.2006	0.2021	0.1980

Table 3. Ranking loss comparison of different algorithms under each data set

algorithms	LMFS	DRMFS	SCLS	MDMR	PMU	FIMF	Baseline
Business	<u>0.0382</u>	0.0443	0.0405	0.0404	0.0444	0.0423	**0.0374**
Image	**0.2000**	0.2213	<u>0.2167</u>	0.2550	0.2483	0.2662	0.2333
Emotion	**0.1662**	<u>0.1687</u>	0.2056	0.1994	0.2570	0.2012	0.2829
Health	**0.0530**	0.0577	<u>0.0562</u>	0.0566	0.0679	0.0578	0.0605
Computers	**0.0844**	0.0934	0.0909	<u>0.0903</u>	0.0980	0.0955	0.0922
Enron	**0.0885**	<u>0.0896</u>	0.0921	0.0944	0.0949	0.0935	0.0938
Scene	<u>0.1014</u>	0.1087	0.1129	0.1444	0.1290	0.1994	**0.0931**
Yeast	<u>0.1677</u>	**0.1656**	0.1745	0.1710	0.1723	0.1747	0.1715

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

Table 4. One-error comparison of different algorithms under each data set

algorithms	LMFS	DRMFS	SCLS	MDMR	PMU	FIMF	Baseline
Business	**0.1177**	0.1303	0.1240	0.1237	0.1320	0.1260	0.1213
Image	**0.3650**	0.3950	0.4000	0.4450	0.4700	0.5000	0.4350
Emotion	**0.2426**	0.2772	0.3614	0.3564	0.3614	0.3515	0.4059
Health	**0.3197**	0.3333	0.3410	0.3373	0.4403	0.3723	0.4207
Computers	**0.4150**	0.4340	0.4580	0.4543	0.4700	0.4627	0.4367
Enron	0.2297	**0.2124**	0.2470	0.2435	0.2694	0.2453	0.3040
Scene	0.2651	0.2968	0.2977	0.3905	0.3904	0.4983	**0.2425**
Yeast	**0.2094**	0.2137	0.2268	0.2366	0.2366	0.2366	0.2345

Table 5. Coverage comparison of different algorithms under each data set

algorithms	LMFS	DRMFS	SCLS	MDMR	PMU	FIMF	Baseline
Business	2.1990	2.4123	2.3050	2.2917	2.4020	2.3600	**2.1847**
Image	**1.0800**	1.1600	1.1650	1.3200	1.2900	1.3550	1.2150
Emotion	**1.8911**	1.9158	2.1139	2.0891	2.3614	2.0545	2.4901
Health	**3.0053**	3.2060	3.1350	3.1647	3.5690	3.1780	3.3047
Computers	**4.1187**	4.4987	4.3697	4.3480	4.6520	4.5580	4.4160
Enron	**12.6790**	12.8450	13.0415	13.1606	13.4128	13.2038	13.2055
Scene	0.6104	0.6421	0.6681	0.8253	0.7492	1.0953	**0.5686**
Yeast	6.2955	**6.2857**	6.4482	6.3642	6.3708	6.3740	6.4144

Table 6. Average precision comparison of different algorithms under each data set

algorithms	LMFS	DRMFS	SCLS	MDMR	PMU	FIMF	Baseline
Business	**0.8809**	0.8689	0.8758	0.8757	0.8690	0.8730	0.8798
Image	**0.7642**	0.7434	0.7437	0.7058	0.7002	0.6791	0.7214
Emotion	**0.8104**	0.7917	0.7496	0.7551	0.7143	0.7510	0.6938
Health	**0.7429**	0.7245	0.7159	0.7256	0.6593	0.7090	0.6812
Computers	**0.6590**	0.6344	0.6317	0.6304	0.6093	0.6203	0.6334
Enron	**0.6742**	0.6704	0.6589	0.6566	0.6483	0.6548	0.6232
Scene	0.8359	0.8158	0.8163	0.7633	0.8034	0.6906	**0.8512**
Yeast	**0.7667**	0.7653	0.7563	0.7579	0.7562	0.7552	0.7585

In order to visually show the relative performance of the LMFS algorithm and other comparing algorithms, Figure 2 to Figure 6 show the performance of all multi-label feature selection algorithms. As the number of selected features changes, the metrics value will also change. Therefore the x-axis represents the number of features selected by each feature selection algorithm, and the y-axis represents the performance of the evaluation metrics after classification feature selection. These results show that in most cases, the proposed LMFS algorithm is superior to the previous ones among almost all data sets.

Specifically, Figure 2 to Figure 6 show the Hamming loss, Ranking loss, One-error, Coverage, and Average precision comparison of different feature selection methods when using ML-KNN as the basic classifier. From a, b, c, e, f and g in Figure 2; a, b, e, f and g in Figure 3; a, b, e, f and g in Figure 4; and a, b, c, d, e, f and g in Figure 5, we can see that the curve of LMFS algorithm is obviously lower than that of all comparison algorithms, even for the other images in Figure 2 to Figure 5, the curves of LMFS algorithm are significantly lower than those of SCLS algorithm, MDMR algorithm, PMU algorithm, and FIMF algorithm, even in the subfigure a of Figure 2 and Figure 4, only the LMFS algorithm's curves are below the baseline. From a, b, c, e, f and g in Figure 6, we can see that the curve of LMFS is significantly higher than that of all comparison algorithms, even in a and f, only the LMFS algorithm's curves were above the baseline. Thus, it can be seen that the proposed LMFS algorithm

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

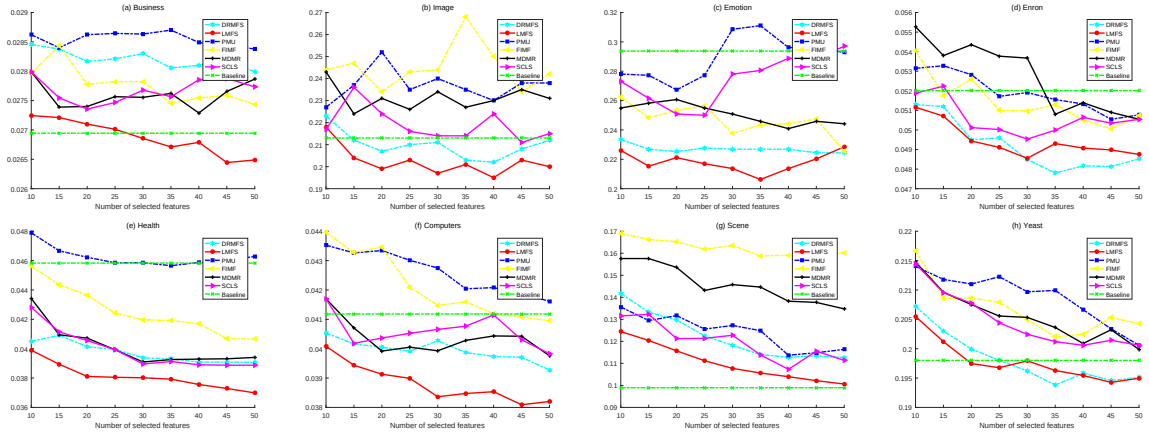

Figure 2. Hamming loss comparison of different feature selection methods when ML-KNN is used as the basic classifier. (the lower the result, the better)

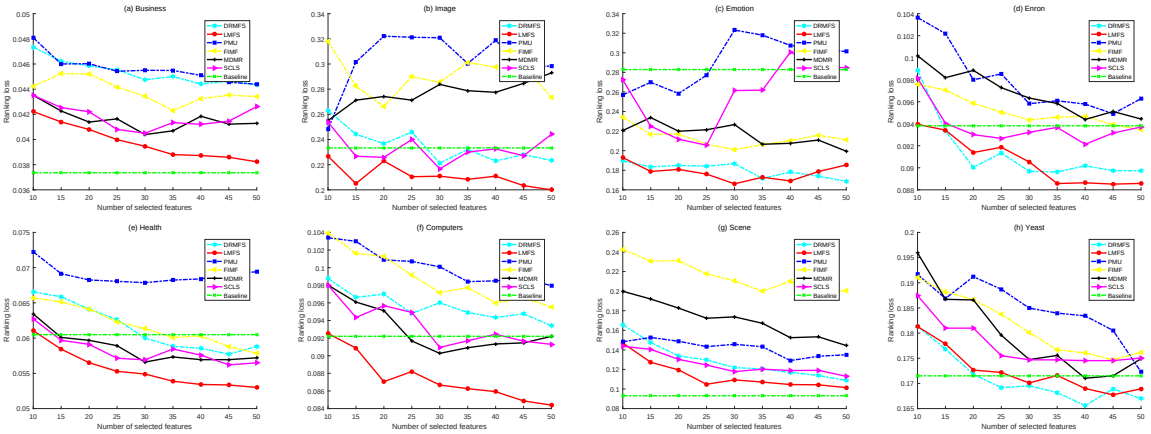

Figure 3. Ranking loss comparison of different feature selection methods when ML-KNN is used as the basic classifier. (the lower the result, the better)

can reduce irrelevant or redundant features.

In addition, in order to explore the influence of parameters on the performance of LMFS algorithm, we choose two different kinds of data sets: music data set Scene, and biological data set Yeast. For parameters λ, β and γ, we fix two of them as 1. The influence of another parameter on the performance of LMFS algorithm is discussed under the selection of different number of features. The parameter range is set as $[0.001, 0.01, 0.1, 1, 10, 100, 1000]$, the number of feature selection is set to $[10, 15, 20, 25, 30, 35, 40, 45, 50]$. The experimental results are shown in Figure 7.

Specifically, the performance of the algorithm will change with the change of parameters. As can be seen from Figure 7, for different data sets, the optimal range of parameters is different. For example, on the Scene data set the optimal range of parameter β is $[10, 100]$, the optimal range of parameter λ is $[10, 1000]$, and the optimal range of parameter γ is $[0.01, 0.1]$. Due to the different basic structure of different data sets, the parameters in Yeast data set are more sensitive, but the parameters in Scene data set are not sensitive. At the same time, in order to explore the influence of the nearest neighbor parameter K on the performance of the algorithm, let $K = [5, 6, 7, 8, 9, 10]$, Experiments were carried

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

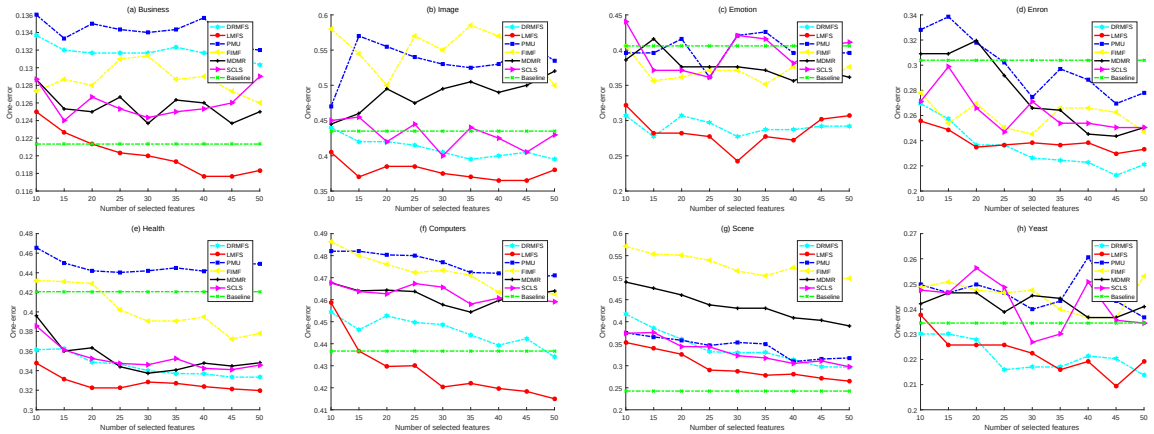

Figure 4. One-error comparison of different feature selection methods when ML-KNN is used as the basic classifier. (the lower the result, the better)

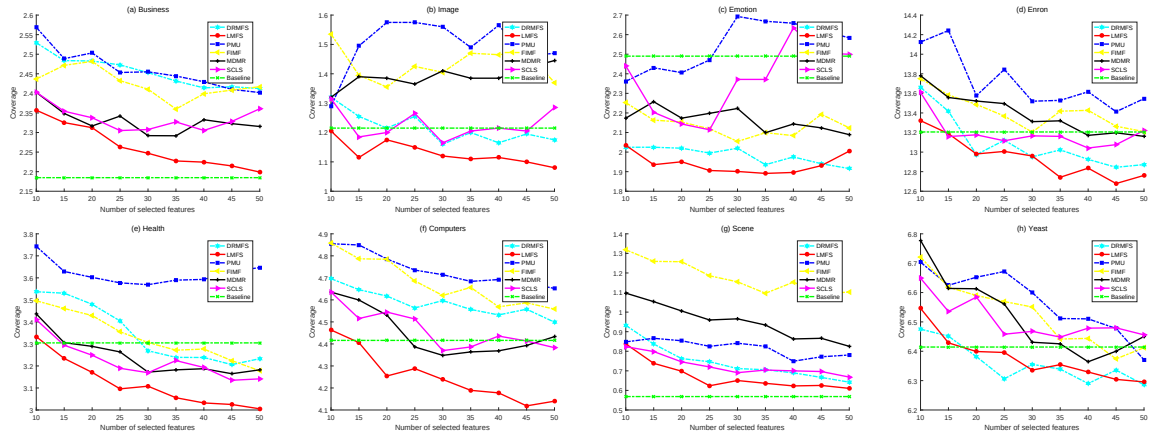

Figure 5. Coverage comparison of different feature selection methods when ML-KNN is used as the basic classifier. (the lower the result, the better)

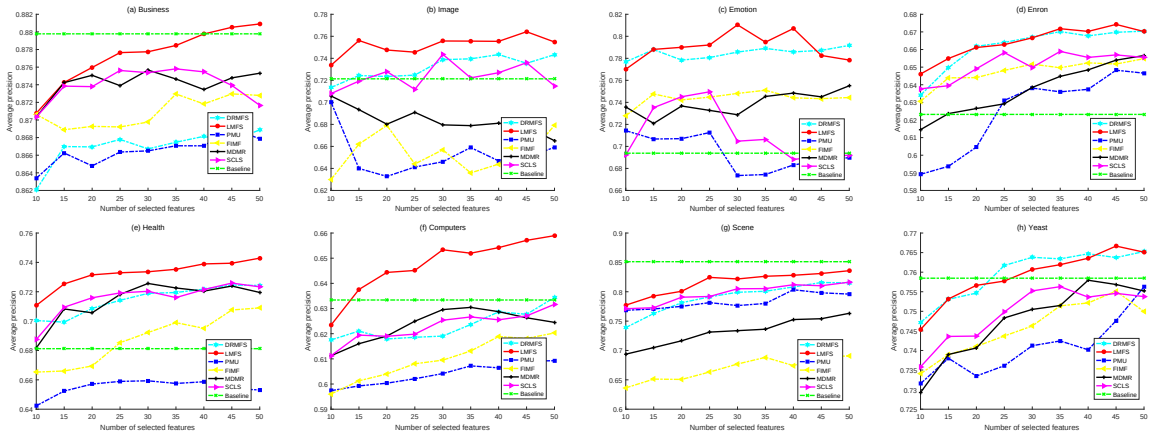

Figure 6. Average precision comparison of different feature selection methods when ML-KNN is used as the basic classifier. (the higher the result, the better)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

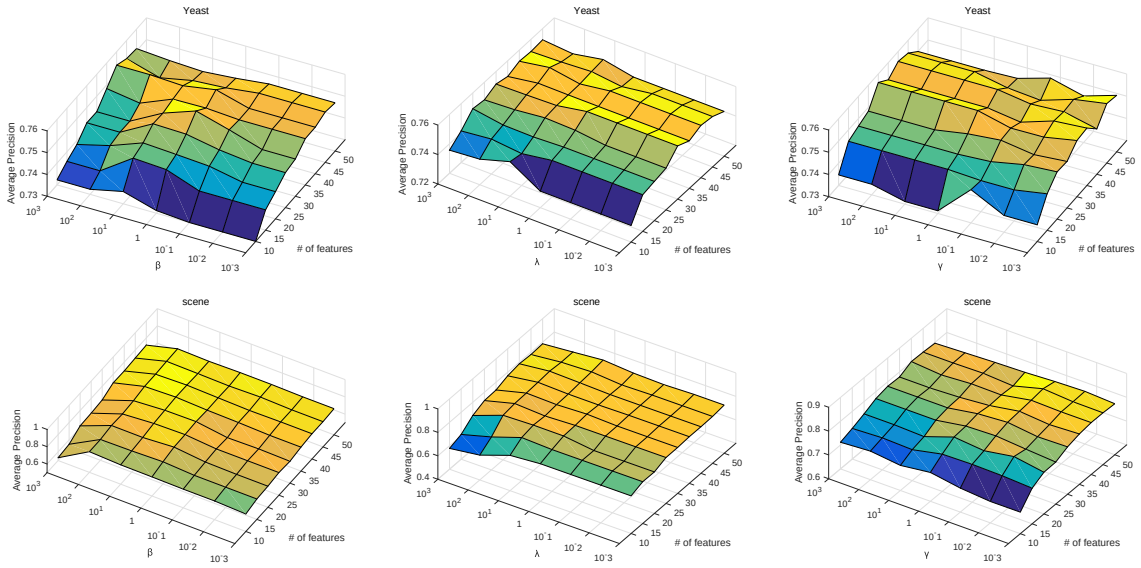

Figure 7. The change of Average precision with parameters in Enron and Scene data set

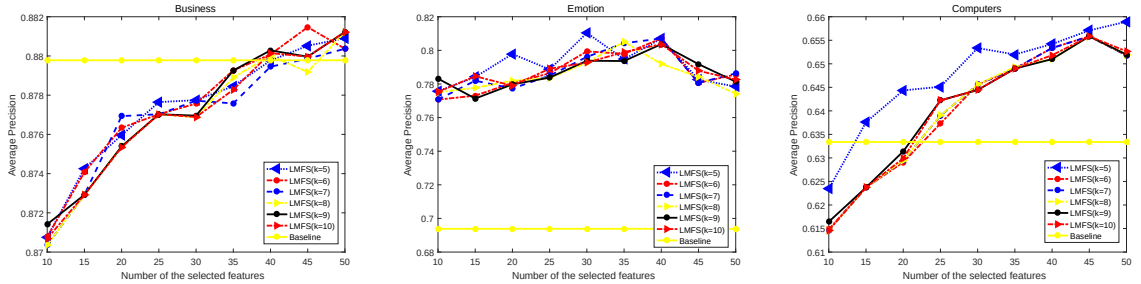

Figure 8. The influence of the nearest neighbor parameter k on the performance of the algorithm

out on three data sets, Business, Emotion and Computers, the experimental results are shown in Figure 8, we can see that the performance of the algorithm is sensitive to K.

As shown in Figure 9, the horizontal axis represents the sorting of multi label feature selection algorithms under each index, from left to right, the performance of the algorithm is getting better and better, the best performing algorithm is on the far right side of Figure 9. At the same time, we report the results of Bonferroni-Dunn test in the form of average rank graph, the algorithm groups with no significant difference $(P < 0.1)$ were connected, if the difference of average ranking reaches the critical value of difference (CD), then there is significant difference [31]. Although LMFS algorithm has no significant difference with DRMFS algorithm, SCLS algorithm and MDMR algorithm in all indicators, LMFS algorithm always has significant difference with PMU and FIFM algorithm, and LMFS algorithm always ranks on the right side.Therefore, compared with other methods, LMFS algorithm shows better performance.

4. Conclusions and outlook

In this paper, logistic regression is combined with feature manifold learning, sparse regularization, and label map regularization to study the multi-label feature selection problem. Sparseness has been widely used in regression-based feature selection methods. In order to overcome the shortcomings that when dealing with the regression coefficient of features, the existing feature selection method based on logistic regression fail to consider the geometric structure of the feature manifold; and that

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 15 July 2021 doi:10.20944/preprints202107.0341.v1

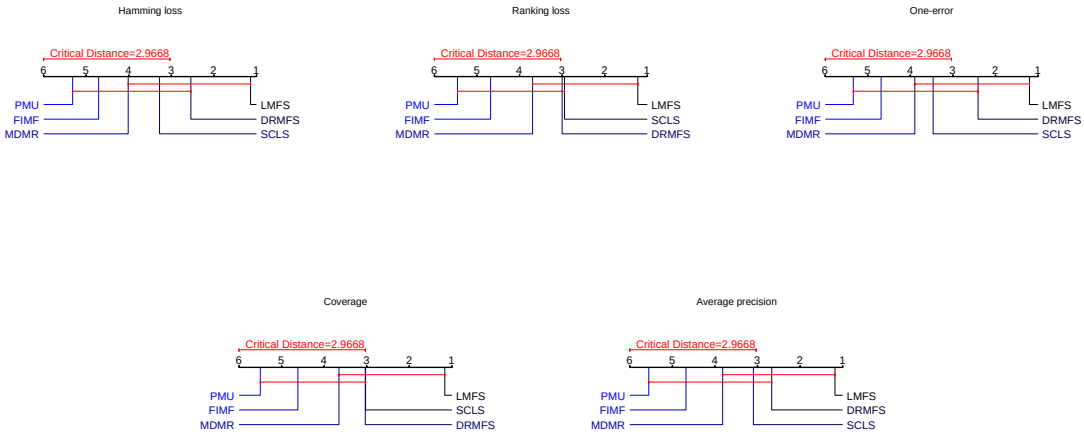

Figure 9. Bonferroni-Dunn test results in the form of average rank diagrams. Groups of feature selection algorithms that are not significantly different (at p = 0.1) are connected

the existing linear regression feature selection method fail to consider the fitting label information between the geometric structure of the feature manifold and the manifold feature coefficient, we embed feature map regularization method and label map regularization method into the multi-label feature selection problem based on logistic regression to obtain the regression coefficients, so that the regression coefficients are smooth relative to the feature manifold without losing the fitting label information. We also design an iterative update algorithm to prove the convergence of LMFS algorithm. Another direction in the future is to extend this method to study the semi-supervised feature selection.

References

1. Cai J, Luo JW, Wang SL, et al. Feature selection in machine learning: A new perspective[J]. Neurocomputing, 2018, 300: 70-79.
2. Bermingham ML, Pong-Wong R, Spiliopoulou A, et al. Application of high-dimensional feature selection: Evaluation for genomic prediction in man[J]. Scientific Reports, 2015, 5: 10312.
3. Sun X, Liu YH, Li J, et al. Using cooperative game theory to optimize the feature selection problem[J]. Neurocomputing, 2012, 97: 86-93.
4. Zhang R, Nie FP, Li XL, et al. Feature selection with multi-view data: A survey[J]. Information Fusion, 2019, 50: 158-167.
5. Ding CC, Zhao M, Lin J, et al. Multi-objective iterative optimization algorithm based optimal wavelet filter selection for multi-fault diagnosis of rolling element bearings[J]. ISA Transactions, 2019, 82: 199-215.
6. Labani M , Moradi P , Ahmadizar F , et al. A novel multivariate filter method for feature selection in text classification problems[J]. Engineering Applications of Artificial Intelligence, 2018, 70: 25-37.
7. Yao C, Liu YF, Jiang B, et al. LLE Score: A New filter-based unsupervised feature selection method based on nonlinear manifold embedding and its application to image recognition[J]. IEEE Transactions on Image Processing, 2017, 26(11): 5257-5269.
8. Gonzalez J, Ortega J, Damas M, et al. A new multi-objective wrapper method for feature selection - Accuracy and stability analysis for BCI[J]. Neurocomputing, 2019, 333: 407-418.
9. Swati J, Hongmei H, Karl J. Information gain directed genetic algorithm wrapper feature selection for credit rating[J]. Applied Soft Computing, 2018, 69: 541-553.
10. Maldonado S, López J. Dealing with high-dimensional class-imbalanced datasets: Embedded feature selection for SVM classification[J]. Applied Soft Computing, 2018, 67: 94-105.

11. Kong YC, Yu TW. A graph-embedded deep feedforward network for disease outcome classification and feature selection using gene expression data[J]. Bioinformatics, 2018, 34(21): 3727-3737.
12. Sun ZQ, Zhang J, Dai L, et al. Mutual information based multi-label feature selection via constrained convex optimization[J]. Neurocomputing, 2019, 329: 447-456.
13. Zhang P, Liu GX, Gao WF. Distinguishing two types of labels for multi-label feature selection[J]. Pattern Recognition, 2019, 95: 72-82.
14. Chen LL, Chen DG. Alignment based feature selection for multi-label learning[J]. Neural Processing Letters, 2019, 50(7): 28-36.
15. Chen SB, Zhang YM, Chris H.Q. Ding, et al. Extended adaptive lasso for multi-class and multi-label feature selection[J]. Knowledge-Based Systems, 2019, 173: 28-36.
16. Zhang J, Luo ZM, Li CD, et al. Manifold regularized discriminative feature selection for multi-label learning[J]. Pattern Recognition, 2019, 95: 136-150.
17. Cai ZL, Zhu W. Multi-label feature selection for non negative sparse representation[J]. Computer science and exploration. 2017, 11(7): 1175-1182.
18. Li Q, Xie B, You J, et al. Correlated logistic model with elastic net regularization for multilabel image classification[J]. IEEE Transactions on Image Processing, 2016, 25(8): 3801-3813.
19. Sato T, Takano Y, Miyashiro R, et al. Feature subset selection for logistic regression via mixed integer optimization[J]. Computational Optimization and Applications, 2016, 64(3): 865-880.
20. Yang ZY, Liang Y, Zhang H, et al. Robust sparse logistic regression with the L_q $(0 < q < 1)$ regularization for feature selection using gene expression data[J]. IEEE ACCESS. 2018, 6: 68586-68595.
21. Liu HW, Zhang SC, Wu XD. MLSLR: Multilabel learning via sparse logistic regression[J]. Information Sciences, 2014, 281: 310-320.
22. Nie FP, Huang H, Cai X, et al. Efficient and robust feature selection via joint - norms minimization[C]// Advances in Neural Information Processing Systems. Canada: ACM Press, 2010: 1813-1821.
23. Gu QQ, Zhou J. Co-clustering on manifolds[C]// Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA: ACM, 2009: 359-368.
24. Zhang ML, Zhou ZH. ML-KNN: a lazy learning approach to multi-label learning[J]. Pattern Recognition, 2007, 40(7): 2038-2048.
25. Hu JC, Li YH, Gao WF, et al. Robust multi-label feature selection with dual-graph regularization[J]. Knowledge-Based Systems. 2020, 203: 106126.
26. Lee J, Kim DW. SCLS: Multi-label feature selection based on scalable criterion for large label set[J]. Pattern Recognition, 2017, 66: 342-352.
27. Lin YJ, Hu QH, Liu JH, et al. Multi-label feature selection based on max-dependency and min-redundancy[J]. Neurocomputing, 2015, 168: 92-103.
28. Lee J, Lim H, Kim DW. Approximating mutual information for multi-label feature selection[J]. Electronics Letters. 2012, 48(15): 929-930.
29. Lee J, Kim DW. Fast multi-label feature selection based on information-theoretic feature ranking[J]. Pattern Recognition, 2015, 48(9): 2761-2771.
30. Dougherty J, Kohavi R, Sahami M, et al. Supervised and unsupervised discretization of continuous features[J]. In: Machine learning: proceedings of the 12th international conference, 1995, 2: 194-202.
31. Demiar J, Schuurmans D. Statistical comparisons of classifiers over multiple data sets[J]. Journal of Machine Learning Research, 2006, 7(1): 1-30.