Article

An Instance- and Label-Based Feature Selection Method in Classification Tasks

Qingcheng Fan 1,4, Sicong Liu 1,4, Chunjiang Zhao 1,2,* and Shuqin Li 1,*

1 College of Information Engineering, Northwest A&F University, 3 Taicheng Road, Yangling, Xianyang, 712100, China; fanqingcheng@nwafu.edu.cn (Q.F.); liusicong@nwafu.edu.cn (S.L.)
2 Research Center of Information Technology, Beijing Academy of Agriculture and Forestry Sciences, Beijing 100097, China
* Correspondence: zhaocj@nercita.org.cn (C.Z.); lsq_cie@nwsuaf.edu.cn (S.L.)
† These authors contributed equally to this work.

Abstract: Feature selection is crucial in classification tasks as it helps to extract relevant information while reducing redundancy. This paper presents a novel method that considers both instance and label correlation. By employing the least squares method, we calculate the linear relationship between each feature and the target variable, resulting in correlation coefficients. Features with high correlation coefficients are selected. Compared to traditional methods, our approach offers two advantages. Firstly, it effectively selects features highly correlated with the target variable from a large feature set, reducing data dimensionality and improving analysis and modeling efficiency. Secondly, our method considers label correlation between features, enhancing the accuracy of selected features and subsequent model performance. Experimental results on three datasets demonstrate the effectiveness of our method in selecting features with high correlation coefficients, leading to superior model performance. Notably, our approach achieves a minimum accuracy improvement of 3.2% for the advanced classifier, lightGBM, surpassing other feature selection methods. In summary, our proposed method, based on instance and label correlation, presents a suitable solution for classification problems.

Keywords: feature selection; manifold learning; classification

1. Introduction

Classification is a fundamental task in machine learning, with diverse applications across various fields [1,2]. In this task, the inputs are typically represented as vectors. However, not all elements in a vector contain relevant or beneficial information for classification; some may even have a detrimental effect on the classification task. Therefore, the selection of informative elements within a sample is a topic of great interest among researchers [3–6].

By reducing feature dimensionality, feature selection significantly enhances model performance, reduces computational complexity, and improves the efficiency of data analysis and modeling processes [7]. On the other hand, manifold learning focuses on dimensionality reduction through manifolds. By assuming that data points are distributed along low-dimensional manifolds, manifold learning aims to decrease the dimensionality of the data while preserving local relationships [8]. This approach has proven particularly valuable in addressing high-dimensional nonlinear problems and attaining a clear representation of the data. Leveraging nonlinear mapping, manifold learning algorithms effectively map high-dimensional data to low-dimensional spaces, facilitating a more profound analysis of the underlying data structures [9].

Feature selection and manifold learning reduce data redundancy and maintain important features. Feature selection reduces dimensionality and removes irrelevant information during manifold learning. Manifold learning is able to analyze local structures and correla-
tions in the lower feature space [10]. This integrated method improves data processing and modeling processes, improving following procedures.

In this paper, we propose a novel feature selection method that takes into account both instance correlation and label correlation. Our method involves using the least squares method [11] to calculate the linear relationship between each feature and the target variable. By fitting a linear model, we obtain weights for each feature. Features with higher weights or stronger correlations can be selected as the final subset. Regularization methods can further constrain feature selection and model complexity, finding an optimal subset with good generalization performance. In cases of sparsity, the $L_{2,1}$ norm is preferred for controlling non-zero coefficients due to its smoother nature and insensitivity to outliers. For exploring deep structural information of labels, introducing a separation variable captures label correlations and transforms the problem into a separable convex optimization problem. This approach preserves global and local structural information, improving performance. Consistency in label correlations is ensured by aligning the predicted label matrix with the ground truth label matrix and considering the similarity between adjacent instances. The objective function is optimized through the alternating direction multiplier method [12].

The main contributions are as follows:

- We introduce a novel unsupervised feature selection method that takes into account both instance correlation and label correlation, effectively mitigating the impact of noise in data.
- We improve performance by introducing a low-dimensional embedding vector to learn potential structural spatial information.
- Our method outperforms existing approaches like PMU, MDFS, MCLS, and FIMF, showing significant improvements in a series of experiments.

2. Related Work

Feature selection methods have a rich research history and can be broadly categorized into two approaches [3,4,7]. The first approach involves generating a new lower-dimensional vector by utilizing vector elements from existing samples. The second approach focuses on selecting specific elements from existing vectors to construct a new vector. These methods play a crucial role in reducing dimensionality and improving the efficiency of feature representation.

Principal component analysis [3,13] is a commonly used dimensionality reduction method that projects the original data onto a new coordinate system, retaining a minimal number of principal components to reduce the dimensionality while preserving important information from the original data. However, it also has some limitations. For example, it cannot consider class information, performs poorly on non-linearly distributed data, and is sensitive to outliers.

The newly generated vectors using the methods above may need to be more consistent with the original vectors. However, the chi-squared test can help to compensate for this deficiency. The chi-squared test [4,14] is used to select discrete features and target variables. Using the chi-squared statistic, it measures the correlation between the feature and the target variable. However, the test assumes independence between the feature and the target variable and may provide incorrect results if this assumption is not met. Additionally, small sample sizes may lead to unstable results.

In fact, each element in the feature vector of the sample is not completely independent, and they have correlation, which can be used to improve the effectiveness of feature selection. PMU [15] is a feature selection approach that leverages the mutual information between the selected features and the tag set. It can be applied to diverse classification problems. The experimental results validate the effectiveness of this method in improving classification performance. PMU proves to be a valuable and efficient tool for feature selection in classification problems. The feature subset $f^+$ is obtained by the following equation:

$$ J = \tilde{I}(\{S, f^+\}; L) - \tilde{I}(S; L), $$ 

(1)
where \( S \) denotes an input feature set, \( L \) denotes a label set, and \( \tilde{I}(S; L) \) denotes the multivariate mutual information. FIMF [16] proposed a fast feature selection method based on information theory feature ranking to address the research gap in computationally efficient feature selection methods. The results indicate that the method significantly reduces the time required to generate feature subsets, particularly for large datasets, but its accuracy is limited. The objective function of FIMF is as follows:

\[
J(f) = \tilde{V}_2(f \times L') + \sum_{k=3}^{b+1} (-1)^k \tilde{V}_k(f \times Q'_{k-1}),
\]

where \( f \) denotes a feature set; \( L \) denotes a label set; \( Q \) is a set of labels consisting of the highest-entropy labels.

MCLS [17] uses manifold learning to transform the logical label space into a Euclidean label space and constrains the similarity between samples through corresponding numerical labels. The final selection criteria integrate both supervised information and the local properties of the data. The formula for computing the \( i \)-th feature is as follows:

\[
MCLS_i = \frac{f_i^T \tilde{L} f_i}{f_i^T D f_i} \frac{f_i^T \tilde{M} f_i}{f_i^T C f_i},
\]

where \( f_i \) denotes the \( i \)-th feature; \( L \) is a Laplacian matrix; \( D \) is a diagonal matrix. In order to reduce dimensionality and select relevant features, MDFS [18] proposed a manifold regularized embedded multi-label feature selection method. This method constructs a low-dimensional embedding that adapts to the label distribution, considers label correlations, and utilizes \( L_2,1 \) norm regularization for feature selection. The objective function of MDFS is as follows:

\[
\min_{F, W, b} \text{Tr}(F^T LF) + ||XW + 1_n b^T - F||_F^2 + \alpha ||F - Y||_F^2 + \beta \text{Tr}(FL_0 F^T) + \gamma ||W||_{2,1},
\]

where \( \alpha, \beta, \gamma \) are hyperparameters. However, this method did not take into account the potential structural information contained in the label. This method has enlightening effects on our research.

3. Materials and Methods

In this section, we propose a novel unsupervised feature selection method. Subsequently, detailed explanations are provided on three components: notations, problem formulation, and optimization.

3.1. Notations

Considering the extensive derivation of formulas and symbol labeling in this paper, Table 1 is provided to present the symbols that may appear. It is assumed that \( A \) is an arbitrary matrix mentioned in the table.
Table 1. Explanation of professional terms.

<table>
<thead>
<tr>
<th>No.</th>
<th>Definition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\text{Tr}(A)$</td>
<td>Trace of $A$</td>
</tr>
<tr>
<td>2</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>$A^T$</td>
<td>The transposition of $A$</td>
</tr>
<tr>
<td>5</td>
<td>$I_d$</td>
<td>$D$ dimensional identity matrix</td>
</tr>
<tr>
<td>6</td>
<td>$n$</td>
<td>Number of samples</td>
</tr>
<tr>
<td>7</td>
<td>$d$</td>
<td>Number of features per sample</td>
</tr>
<tr>
<td>8</td>
<td>$c$</td>
<td>Number of categories</td>
</tr>
<tr>
<td>9</td>
<td>$X$</td>
<td>Feature vector</td>
</tr>
<tr>
<td>10</td>
<td>$W$</td>
<td>Weight matrix</td>
</tr>
<tr>
<td>11</td>
<td>$V$</td>
<td>Predicted label matrix</td>
</tr>
<tr>
<td>12</td>
<td>$L$</td>
<td>Laplacian matrix</td>
</tr>
<tr>
<td>13</td>
<td>$Y$</td>
<td>Ground truth label of dataset</td>
</tr>
</tbody>
</table>

3.2. Problem Formulation

In the process of feature selection, we use the least squares method to calculate the linear relationship between each feature and the target variable. By fitting a linear model, the weight of each feature can be obtained. Based on these weights, we can select features with high weight or correlation as the final feature subset, which can be used as the basis for feature selection. In addition, the least squares method is further extended through regularization methods to constrain feature selection and model complexity. This can help us to find the optimal subset of features with good generalization performance, improving the predictive ability and interpretability of the model. The basic formula is as follows:

$$
\min_{W, b} ||XW + 1_n b^T - Y||^2_F + \alpha ||W||^2_p.
$$

where $p$ is the norm representation and $W$ is a feature selection matrix, which is the key to solving the problem.

In feature selection, the $L_1$ norm is usually used to count the number of non-zero coefficients to encourage model selection to select sparse features. However, the $L_1$ norm is not ideal for selecting non-zero coefficients, as it tends to select some non-zero coefficients and compress other coefficients to zero without explicitly controlling the number of non-zero coefficients. In contrast, $L_{2,1}$ norm can better handle the sparsity of features. $L_{2,1}$ norm firstly calculates the sum of the $L_2$ norms of each feature vector, and then the $L_1$ norm is calculated for these results. The $L_{2,1}$ norm can better constrain the number of non-zero coefficients because it is smoother and insensitive to outliers. Since $Y$ is a ground truth value label and a constant, and $1_n b^T$ is also a constant, they can be merged together. Therefore, the formula is simplified as follows:

$$
\min_{W, b} ||XW - Y||^2_F + \alpha ||W||^2_{2,1}.
$$

The process of completing feature selection is that $Y$ is approximate to $XW$, so $w^T x_i - w^T x_j$ is used to represent the distance, and $S_{ij}$ is used to describe the similarity between two vectors. Using this method can reduce the negative impact of noise in the original data and reduce the spatial dimension to reduce the computational cost. In this way, the instance association is increased and expressed by the following formula:

$$
\frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} ||w^T x_i - w^T x_j||^2_S S_{ij} = \text{Tr}(W^T X^T (D - S) X W) = \text{Tr}(W^T X^T L X W)
$$

(7)
where $L$ is a Laplacian matrix, which can be obtained by multiplying the matrix of similarity $S$ by the matrix of diagonals $D$, where $d_{ii} = \sum_{j=0}^{n} S_{ij}$. The detail is as follows:

$$S = \begin{cases} e^{-0.5 \frac{||x_i - x_j||^2}{\sigma^2}}, & \text{if } i \neq j; \\ 1, & \text{if } i = j. \end{cases} \tag{8}$$

$$D = \begin{bmatrix} d_{11} & 0 & \cdots & 0 \\ 0 & d_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & d_{nn} \end{bmatrix}_{n \times n} \tag{9}$$

$$L = D - S \tag{10}$$

In order to explore the underlying structural information of the labels, we introduce a separate variable $V$ for capturing label correlations. By incorporating the low-dimensional embedding $V$, the aforementioned problem is transformed into a separable convex optimization problem, which preserves both global and local structured information, thereby enhancing performance. To ensure consistent correlation of labels, the predicted label matrix $V$ should match the ground truth label matrix $Y$, and the instances within matrix $V$ should exhibit similarity when they are adjacent. Based on the aforementioned analysis, we have formulated an equation to describe the label correlation.

$$\min_{V} n \sum_{i=1}^{n} E_{ii} \sum_{j=1}^{l} (V_{ij} - Y_{ij})^2 + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} S_{ij} ||v_i - v_j||^2_2. \tag{11}$$

After conversion, it can be obtained that

$$\min_{V} \text{Tr}[(V - Y)^T E(V - Y)] + \text{Tr}(V^T LV), \tag{12}$$

where $E$ is a diagonal matrix with large numerical values for its elements. The two terms in this formula preserve global and local information, respectively. Given the correlation between matrix $V$ and $Y$, it is crucial to enforce non-negativity of $V$ in classification tasks to ensure its validity.

In summary, the ultimate objective function is formulated as:

$$\min_{W,V} \|[XW - V]\|_F^2 + \alpha \|[W]\|_2^2 + \beta \text{Tr}(W^T X^T LXW) + \gamma (\text{Tr}[(V - Y)^T E(V - Y)] + \text{Tr}(V^T LV)) \quad \text{s.t. } V \geq 0, \tag{13}$$

where $\alpha$, $\beta$, and $\gamma$ are hyperparameters. And then, we will optimize this objective function to obtain the solution to the problem.

### 3.3. Optimization

We have designed an optimization algorithm for the designed objective function. The objective function can be transformed into:

$$\Gamma = \min_{W,V} \text{Tr}[(XW - V)^T (XW - V)] + \alpha \text{Tr}(W^T AW) + \beta \text{Tr}(W^T X^T LXW) + \gamma (\text{Tr}[(V - Y)^T E(V - Y)] + \text{Tr}(V^T LV)) \quad \text{s.t. } V \geq 0, \tag{14}$$

where $A$ is a diagonal matrix with its elements defined as $\frac{1}{2} ||\bar{w}||_2$. Subsequently, we employ the alternating direction method of multipliers to solve the aforementioned problem.
3.3.1. Solve $W$ as given $V$

By setting the partial derivative $\frac{\partial \Gamma}{\partial W} = 0$, we can obtain

$$
\frac{\partial}{\partial W} \left( \text{Tr}[(XW - V)^T(XW - V)] + a \text{Tr}(W^TAW) + \beta \text{Tr}(W^TX^TLXW) \right) = 0
$$

$$
\Rightarrow X^TWX - X^TV + aAW + \beta X^TX^TLXW = 0
$$

$$
\Rightarrow W = (X^TX + a\Lambda + \beta X^TX^TLX)^{-1}X^TV,
$$

(15)

where $\Lambda$ represents a non-zero constant that serves the purpose of safeguarding the algorithm against potential crashes caused by a denominator equal to zero.

3.3.2. Solve $V$ as given $W$

By setting the partial derivative $\frac{\partial \Gamma}{\partial V} = 0$, we can obtain

$$
\frac{\partial}{\partial V} \left( \text{Tr}[(XW - V)^T(XW - V)] + \gamma \left( \text{Tr}[(V - Y)^TE(V - Y)] + \text{Tr}(V^TLV) \right) \right) = 0
$$

$$
\Rightarrow XW + V + \gamma(LV + EV - EY) = 0
$$

$$
\Rightarrow V = [\gamma(L + E) + I]^{-1}(\gamma EY + XW),
$$

(17)

Therefore, the solution to this issue is illustrated by Equations (15) and (17). To attain the ultimate solution, it is recommended to iterate through the aforementioned procedure until convergence. The iterative algorithm is outlined in Algorithm 1.

---

### Algorithm 1: The iterative optimization.

**Input:** The feature matrix $X \in \mathbb{R}^{n \times d}$, label matrix $Y \in \mathbb{R}^{n \times c}$, and hyperparameters $\alpha, \beta, \gamma$.

**Output:** The first $K$ features obtained by sorting the results of $||W||_2$.

1. **Initialization:** Randomly initialize $W_0 \in \mathbb{R}^{d \times c}$, $V_0 \in \mathbb{R}^{n \times c}$ and $E$, set $\varepsilon = 0.01$.
2. Calculate $L$, $\Lambda$ using Equation (10), Equation (16), respectively.
3. while not converged do
   4. Calculate $W \leftarrow \text{iteration}_W(X, \Lambda, L, V, \alpha, \beta)$, update $W$ by Equation (15).
   5. Calculate $V \leftarrow \text{iteration}_V(X, L, E, Y, W, \gamma)$, update $V$ by Equation (17).
   6. Calculate $\Lambda \leftarrow \text{iteration}_\lambda(W, \varepsilon)$, update $\Lambda$ by Equation (16).
4. end
5. Sort $||W_i||_2$ in descending order and filter out the top $K$ features from the sorted list.

---

### 4. Results and Discussion

#### 4.1. Datasets

To evaluate and validate this method, these publicly available datasets were carefully selected, including well-known image dataset Yale and biological dataset Lung. In addition, Cattle dataset ([https://www.kaggle.com/datasets/twisdu/dairy-cow, accessed on 1 August 2023)] is also used, which is specifically curated for the task of cattle pose estimation, with pose classification serving as its downstream objective. Through thorough analysis of the relative positions of key points, the dataset enables an accurate determination of the cattle’s pose, thereby achieving the ultimate goal of precise pose classification. The detailed
information of these datasets is presented in Table 2. All datasets are divided into training and testing sets in a 7:3 ratio.

Table 2. The detailed information of these datasets.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Number of Samples</th>
<th>Number of Features</th>
<th>Number of Categories</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lung</td>
<td>203</td>
<td>3312</td>
<td>5</td>
</tr>
<tr>
<td>Yale</td>
<td>165</td>
<td>1024</td>
<td>15</td>
</tr>
<tr>
<td>Cattle</td>
<td>1800</td>
<td>32</td>
<td>3</td>
</tr>
</tbody>
</table>

4.2. Evaluation Metrics

In this paper, to thoroughly evaluate the performance of all methods, we employed the following five evaluation metrics for comprehensive evaluation: accuracy, macro-$F_1$, micro-$F_1$, kappa [19], and Hamming loss [20].

- **Accuracy**
  \[
  \text{accuracy}(y, \tilde{y}) = \frac{1}{n} \sum_{i=0}^{n-1} F(y, \tilde{y}), \quad F(y, \tilde{y}) = \begin{cases} 
  0, & \text{if } y \neq \tilde{y} \\
  1, & \text{if } y = \tilde{y}.
  \end{cases} \tag{18}
  \]

- **Macro-$F_1$ and Micro-$F_1$**
  \[
  F_1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}},
  \]
  \[
  \text{Precision} = \frac{TP}{TP + FP},
  \]
  \[
  \text{Recall} = \frac{TP}{TP + FN} \tag{19}
  \]
  where $TP$ is the number of positive samples correctly identified. $FP$ is the number of negative samples for false positives. $FN$ is the number of missed positive samples. To obtain the macro-$F_1$ score, first calculate the $F_1$ value for each individual category, and then compute the average of all $F_1$ values across all categories. After, calculate the overall accuracy and recall rate. Then, compute the $F_1$ value to obtain the micro-$F_1$ score.

- **Kappa**
  \[
  K = \frac{(p_o - p_e)}{1 - p_e}, \tag{20}
  \]
  where $p_o$ refers to the empirical probability of agreement on the label assigned to any sample, also known as the observed agreement ratio. On the other hand, $p_e$ represents the expected agreement between annotators when labels are assigned randomly. To estimate $p_e$, a per-annotator empirical prior over the class labels is employed.

- **Hamming loss**
  \[
  L_{\text{Hamming}}(y, \tilde{y}) = \frac{1}{n \times c} \sum_{i=1}^{n-1} \sum_{j=0}^{c-1} P(y, \tilde{y}), \quad P(y, \tilde{y}) = \begin{cases} 
  0, & \text{if } y_{i,j} \neq \tilde{y}_{i,j} \\
  1, & \text{if } y_{i,j} = \tilde{y}_{i,j}.
  \end{cases} \tag{21}
  \]
  where $y$ is a ground truth label and $\tilde{y}$ is a predicted label.

Accuracy is a widely used performance evaluation metric for classification tasks. It represents the proportion of correctly classified samples among all the classified samples. The micro-$F_1$ score considers the overall count of true positives, false negatives, and false positives across all categories, while the macro-$F_1$ score considers the $F_1$ score for each individual category and calculates their unweighted average. It is important to note that the macro-$F_1$ score does not consider the potential imbalance between different categories. The kappa coefficient is a measure used to assess consistency in testing. In the context of classification problems, consistency refers to the alignment between the model’s predicted results and the actual classification outcomes. It is calculated based on the confusion matrix.
and ranges between $-1$ and $1$, with values typically greater than $0$. Hamming loss is a metric utilized to analyze misclassification on individual labels. A lower value of this metric indicates better performance, as it signifies fewer misclassifications. The aforementioned set of five metrics is employed for evaluating the performance of the method.

4.3. Experimental Setup

To ensure a fair and unbiased evaluation of all methods, a traversal search strategy is adopted to select the hyperparameters for each method within the range of $[0.1, 0.2, \ldots, 1.0]$. Additionally, feature selection was performed based on the magnitude of the feature number values in the dataset. Specifically, due to the number of features in the Lung and Yale datasets, a percentage-based filtering approach is employed, ranging from $1\%$ to $20\%$ in steps of $1\%$. Alternatively, the Cattle dataset, with only 32 features, undergoes feature selection based on the number of key points, ranging from 2 to 32 in steps of 2, taking into account their actual significance.

For the purpose of conducting experiments, the LightGBM classification algorithm [21] is employed, which is a highly performant framework for gradient-enhanced decision trees, which has been optimized and improved based on the GBDT library xgboost [22]. It boasts faster training speeds and lower memory consumption compared to traditional GBDT algorithms [23,24], utilizes a histogram-based algorithm, and introduces mutually exclusive feature bundling to enhance model performance. It also incorporates the GOSS training strategy, which selectively retains samples with larger gradients to expedite the training process. Based on the aforementioned characteristics, a good performance is achieved.

Upon completing the above steps, each method will be executed 20 times. The mean and standard deviation are calculated for the indicators of 20 experiments, and the absence of variance indicates that the experimental results are the same without any difference. The mean values of the evaluation metrics will be recorded for further analysis.

4.4. Experiment Results

We conducted comparative experiments using the PMU, MDFS, MCLS, and FIMF methods. We also proposed distinct filtering strategies based on the number of features present in each dataset sample. The performance of each method on the dataset was evaluated using five metrics. As depicted in Table 3, our method outperforms all others in terms of every metrics, securing the first position.
Table 3. The comparative results of our method with PMU, MDFS, MCLS, and FIMF based on various evaluation metrics including accuracy, macro-$F_1$, micro-$F_1$, kappa, and Hamming loss is presented. LightGBM is presented, where the classification algorithm can be used directly without the need for any feature selection.

<table>
<thead>
<tr>
<th>Methods</th>
<th>accuracy</th>
<th>macro-$F_1$</th>
<th>micro-$F_1$</th>
<th>kappa</th>
<th>Hamming loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lung</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LightGBM</td>
<td>0.918</td>
<td>0.716</td>
<td>0.918</td>
<td>0.836</td>
<td>0.082</td>
</tr>
<tr>
<td>PMU</td>
<td>0.957 ± 0.021</td>
<td>0.909 ± 0.072</td>
<td>0.957 ± 0.021</td>
<td>0.912 ± 0.042</td>
<td>0.043 ± 0.021</td>
</tr>
<tr>
<td>MDFS</td>
<td>0.952 ± 0.021</td>
<td>0.901 ± 0.067</td>
<td>0.952 ± 0.021</td>
<td>0.902 ± 0.044</td>
<td>0.048 ± 0.021</td>
</tr>
<tr>
<td>MCLS</td>
<td>0.959 ± 0.030</td>
<td>0.892 ± 0.112</td>
<td>0.959 ± 0.030</td>
<td>0.915 ± 0.056</td>
<td>0.041 ± 0.030</td>
</tr>
<tr>
<td>FIMF</td>
<td>0.957 ± 0.025</td>
<td>0.919 ± 0.071</td>
<td>0.957 ± 0.025</td>
<td>0.919 ± 0.040</td>
<td>0.043 ± 0.025</td>
</tr>
<tr>
<td>Ours</td>
<td>0.984</td>
<td>0.979</td>
<td>0.984</td>
<td>0.968</td>
<td>0.016</td>
</tr>
</tbody>
</table>

Yale

<table>
<thead>
<tr>
<th>Methods</th>
<th>accuracy</th>
<th>macro-$F_1$</th>
<th>micro-$F_1$</th>
<th>kappa</th>
<th>Hamming loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>LightGBM</td>
<td>0.660</td>
<td>0.607</td>
<td>0.660</td>
<td>0.636</td>
<td>0.340</td>
</tr>
<tr>
<td>PMU</td>
<td>0.662 ± 0.035</td>
<td>0.632 ± 0.026</td>
<td>0.662 ± 0.035</td>
<td>0.636 ± 0.037</td>
<td>0.338 ± 0.035</td>
</tr>
<tr>
<td>MDFS</td>
<td>0.716 ± 0.034</td>
<td>0.705 ± 0.035</td>
<td>0.716 ± 0.034</td>
<td>0.694 ± 0.035</td>
<td>0.284 ± 0.034</td>
</tr>
<tr>
<td>MCLS</td>
<td>0.586 ± 0.048</td>
<td>0.551 ± 0.047</td>
<td>0.586 ± 0.048</td>
<td>0.555 ± 0.052</td>
<td>0.414 ± 0.048</td>
</tr>
<tr>
<td>FIMF</td>
<td>0.672 ± 0.091</td>
<td>0.669 ± 0.070</td>
<td>0.672 ± 0.091</td>
<td>0.647 ± 0.097</td>
<td>0.328 ± 0.091</td>
</tr>
<tr>
<td>Ours</td>
<td>0.760</td>
<td>0.715</td>
<td>0.760</td>
<td>0.739</td>
<td>0.240</td>
</tr>
</tbody>
</table>

Cattle

<table>
<thead>
<tr>
<th>Methods</th>
<th>accuracy</th>
<th>macro-$F_1$</th>
<th>micro-$F_1$</th>
<th>kappa</th>
<th>Hamming loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>LightGBM</td>
<td>0.877</td>
<td>0.877</td>
<td>0.877</td>
<td>0.816</td>
<td>0.123</td>
</tr>
<tr>
<td>PMU</td>
<td>0.899</td>
<td>0.899</td>
<td>0.899</td>
<td>0.849</td>
<td>0.101</td>
</tr>
<tr>
<td>MDFS</td>
<td>0.892</td>
<td>0.892</td>
<td>0.892</td>
<td>0.838</td>
<td>0.108</td>
</tr>
<tr>
<td>MCLS</td>
<td>0.894</td>
<td>0.894</td>
<td>0.894</td>
<td>0.841</td>
<td>0.106</td>
</tr>
<tr>
<td>FIMF</td>
<td>0.903</td>
<td>0.903</td>
<td>0.903</td>
<td>0.855</td>
<td>0.097</td>
</tr>
<tr>
<td>Ours</td>
<td>0.909</td>
<td>0.909</td>
<td>0.909</td>
<td>0.863</td>
<td>0.091</td>
</tr>
</tbody>
</table>

To further illustrate the influence of retaining different feature ratios on the final classification results, we conducted multiple sets of experiments based on the settings outlined in Section 4.3. The results corresponding to varying numbers of features are displayed in the Figure 1. This allows for a comprehensive understanding of how different feature ratios affect the classification outcomes.

The horizontal axis of the figures is the number of features selected in the task. The vertical axis denotes the value of evaluation indicators. As the number of features increases, the accuracy of classification increases. While the curve is not monotonous, it is fluctuating. The Hamming loss is decreasing along with the increase in selected features.
These results indicate that our method is highly accurate and effective in classifying lung datasets. On the Lung dataset, our method achieved an accuracy of 0.984, a macro-F1 score of 0.979, a micro-F1 score of 0.984, a kappa coefficient of 0.968, and a Hamming loss of 0.016. These results indicate that our method is highly accurate and effective in classifying lung
data. Similarly, on the Yale dataset, our method achieved an accuracy of 0.760, a macro-$F_1$ score of 0.715, a micro-$F_1$ score of 0.760, a kappa coefficient of 0.739, and a Hamming loss of 0.240. This demonstrates the superior performance of our method compared to other approaches. Lastly, on the Cattle dataset, our method achieved an accuracy of 0.909, a macro-$F_1$ score of 0.909, a micro-$F_1$ score of 0.909, a kappa coefficient of 0.863, and a Hamming loss of 0.091. These results further validate the effectiveness of our method for accurately classifying cattle data.

The above experimental results indicate that our method can better capture global and local potential information between features, thereby filtering out features that are more important to the classification task and improving the accuracy of the final classification. In contrast, LightGBM, which does not involve feature selection, performed relatively well on the Lung and Cattle datasets but showed lower performance on the Yale dataset. The other methods also showed varying degrees of performance across the different datasets. Overall, our proposed method demonstrates superior performance in terms of accuracy, $F_1$ scores, kappa coefficient, and Hamming loss on all three datasets. These results highlight the effectiveness of our method for classification tasks and its potential for practical applications.

4.5. Analysis

During the iteration process, note that the objective function value for the $t$th iteration is $\Gamma_t$, and the objective function value for the $t-1$th iteration is $\Gamma_{t-1}$. The iteration is terminated if either the absolute difference between $\Gamma_{t-1}$ and $\Gamma_t$ is less than 0.01 or if the iteration count exceeds 10.

Figure 2 illustrates the variation in the objective function value during the iteration of our method on each dataset. It is evident that the objective function value has converged significantly by the second epoch. This observation demonstrates the efficiency and fast convergence of our method.

![Figure 2. Cont.](image-url)
4.6. Ablation Study

In order to better distinguish the effect of each part of the objective function, we divided the objective function into three basic parts, named the basic formula (BASE), instance correlation (IC), and label correlation (LC), and conducted ablation experiments on each dataset. The specific formulas for the three parts of goal division are as follows:

\[
BASE = \min_{W \in \mathbb{B}} \|XW - Y\|^2_F + \alpha \|W\|_{2,1}^2,
\]

(22)

\[
IC = \min_{W \in \mathbb{V}} \|XW - V\|^2_F + \alpha \|W\|_{2,1}^2 + \beta \text{Tr}(W^T X^T L X W),
\]

(23)

\[
LC = \min_{W \in \mathbb{V}} \text{Tr}[(XW - V)^T (XW - V)] + \alpha \text{Tr}(W^T A W) + \beta \text{Tr}(W^T X^T L X W) + \\
\gamma \text{Tr}[(V - Y)^T E(V - Y) + \text{Tr}(V^T L V)].
\]

(24)

We start with Equation (22) as the baseline and then incorporate the IC module into the objective function, which considers the noise elimination of the original data. As shown in Table 4, this leads to an improvement in classification accuracy. With the help of this module, all of the evaluation indicators witnessed progress of different sizes in the three datasets. The accuracy of classification was improved by 6% in the Yale dataset. Following the IC module, the LC module is added to the objective function, taking into account the distance between the low-dimensional embedded label \(V\) and the predicted label \(WX\), as well as the ground truth value label and the internal elements of \(V\). This incorporation of both global and local information during the process enhances the accuracy of the method. The LC module helped the method to obtain improvements in all the indicators. It is notable that the Hamming loss was no more than half of what it was before in the lung dataset. The experiments show that the improvements in this paper are effective, bringing benefits to the classification task.

Table 4. Our method conducts ablation experiments on various datasets, where BASE, IC, and LC correspond to formulas Equation (22), Equation (23), and Equation (24), respectively.

<table>
<thead>
<tr>
<th>Modules</th>
<th>accuracy</th>
<th>macro-F1</th>
<th>micro-F1</th>
<th>kappa</th>
<th>Hamming loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASE</td>
<td>0.934</td>
<td>0.764</td>
<td>0.934</td>
<td>0.871</td>
<td>0.066</td>
</tr>
<tr>
<td>IC</td>
<td>0.967</td>
<td>0.955</td>
<td>0.967</td>
<td>0.936</td>
<td>0.033</td>
</tr>
<tr>
<td>LC</td>
<td>0.984</td>
<td>0.979</td>
<td>0.984</td>
<td>0.968</td>
<td>0.016</td>
</tr>
</tbody>
</table>
Table 4. Cont.

<table>
<thead>
<tr>
<th>Modules</th>
<th>accuracy</th>
<th>macro-F1</th>
<th>micro-F1</th>
<th>kappa</th>
<th>Hamming loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASE</td>
<td>0.640</td>
<td>0.616</td>
<td>0.640</td>
<td>0.614</td>
<td>0.360</td>
</tr>
<tr>
<td>IC</td>
<td>0.700</td>
<td>0.670</td>
<td>0.700</td>
<td>0.676</td>
<td>0.300</td>
</tr>
<tr>
<td>LC</td>
<td>0.760</td>
<td>0.715</td>
<td>0.760</td>
<td>0.739</td>
<td>0.240</td>
</tr>
</tbody>
</table>

Cattle

<table>
<thead>
<tr>
<th>Methods</th>
<th>accuracy</th>
<th>macro-F1</th>
<th>micro-F1</th>
<th>kappa</th>
<th>Hamming loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASE</td>
<td>0.885</td>
<td>0.885</td>
<td>0.885</td>
<td>0.827</td>
<td>0.115</td>
</tr>
<tr>
<td>IC</td>
<td>0.905</td>
<td>0.905</td>
<td>0.905</td>
<td>0.858</td>
<td>0.095</td>
</tr>
<tr>
<td>LC</td>
<td>0.909</td>
<td>0.909</td>
<td>0.909</td>
<td>0.863</td>
<td>0.091</td>
</tr>
</tbody>
</table>

5. Conclusions

In this paper, we have introduced a novel unsupervised feature selection method that incorporates both instance correlation and label correlation. By considering instance correlation, we can mitigate the negative impact of noise in the datasets. Additionally, our approach utilizes a low-dimensional embedded vector to capture global and local information through label correlation. Experimental results have demonstrated that our method outperforms existing approaches such as PMU, MDFS, MCLS, and FIMF, thus validating the effectiveness of our improvements. Moving forward, our future work will explore the underlying relationship between global and local label information.

Author Contributions: Conceptualization, S.L. (Sicong Liu) and Q.F.; methodology, S.L. (Sicong Liu) and Q.F.; software, Q.F. and S.L. (Sicong Liu); validation, Q.F. and S.L. (Sicong Liu); data curation, Q.F.; writing—original draft preparation, Q.F. and S.L. (Sicong Liu); writing—review and editing, Q.F. and S.L. (Sicong Liu); visualization, Q.F.; supervision, S.L. (Shuqin Li) and C.Z.; project administration, S.L. (Shuqin Li); All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank Caoli Gong of Inner Mongolia Agricultural University for opening the cattle dataset. This provides great convenience for our research.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

- LightGBM: A highly efficient gradient boosting decision tree
- PMU: Feature selection for multi-label classification using multivariate mutual information
- MDFS: Manifold regularized discriminative feature selection for multi-label learning
- MCLS: Manifold-based constraint Laplacian score for multi-label feature selection
- FIMF: Fast multi-label feature selection based on information-theoretic feature ranking
- GBDT: Gradient boost decision tree
- xgboost: Extreme gradient boosting
References


5. Boyd S.; Parikh N.; and Chu E.; Peleato B. and Eckstein J. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers 2011


Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.