SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

February 2025, Vol. 68, Iss. 2, 122103:1–122103:14
https://doi.org/10.1007/s11432-023-4132-0

Fractal autoencoder with redundancy regularization for unsupervised feature selection

Meiting SUN[1,2,3], Fangyu LI[1,2,3] & Honggui HAN[1,2,3,4*]

[1]*School of Information Science and Technology, Beijing University of Technology, Beijing 100124, China*
[2]*Beijing Key Laboratory of Computational Intelligence and Intelligent System, Beijing 100124, China*
[3]*Engineering Research Center of Digital Community, Ministry of Education, Beijing 100124, China*
[4]*Beijing Institute of Artificial Intelligence, Beijing 100124, China*

Received 22 December 2023/Revised 25 March 2024/Accepted 5 August 2024/Published online 14 January 2025

Abstract Feature selection is a crucial step in data preprocessing because feature selection reduces the dimensionality of data by eliminating irrelevant and redundant features. Since manual labeling is expensive, unsupervised feature selection has received increasing attention in recent years. However, existing unsupervised feature selection methods tend to prioritize selecting highly correlated features over exploring feature diversity. Thus, a regularized fractal autoencoder (RFAE) method is proposed to select informative features in an unsupervised way. Specifically, the fractal autoencoder network extends autoencoders to construct a correspondence neural network and a selection neural network. The correspondence neural network exploits inter-feature correlations and the selection neural network selects the informative features. A redundancy regularization strategy consists of a redundancy elimination regularization term based on the dependency between features and a sparse regularization term based on the group lasso. The redundancy regularization strategy eliminates feature subset redundancy and enhances network generalization ability. Extensive experimental results on six publicly available datasets show that the proposed RFAE outperforms the compared methods regarding clustering accuracy and classification accuracy. Moreover, the proposed RFAE achieves acceptable computation efficiency.

Keywords unsupervised feature selection, fractal autoencoder, correspondence neural network, selection neural network, redundancy regularization strategy

Citation Sun M T, Li F Y, Han H G. Fractal autoencoder with redundancy regularization for unsupervised feature selection. Sci China Inf Sci, 2025, 68(2): 122103, https://doi.org/10.1007/s11432-023-4132-0

1 Introduction

With the rapid advancement of information technology, various fields, such as data mining [1], computer vision [2], and pattern recognition [3, 4], have generated high-dimensional data. The high-dimensional data contain noisy, irrelevant, and redundant features, resulting in the curse of dimensionality, huge computation burden, and heavy storage cost [5]. Typically, the intrinsic dimensionality of the data is much lower than the real original feature space. To produce useful results and obtain less running time in machine learning techniques, dimensionality reduction by removing irrelevant and redundant features is an important issue. Generally, the dimensionality reduction methods can be classified into feature extraction and feature selection [6–8].

Feature extraction, like principal component analysis and linear discriminant analysis, transforms and compresses the original high-dimensional features into a low-dimensional space through linear or nonlinear mapping [9]. The mapped features no longer have the physical meaning of the original features and affect the original features' role in practical application [10–12]. In contrast to feature extraction, feature selection has better interpretability because feature selection preserves the semantics meaning of the original features by selecting the optimal feature subset from the original high-dimensional features [13]. Moreover, the cost of collecting features for learning algorithms is reduced because feature selection only needs to collect selected features rather than all the original features [14].

From the perspective of labeling availability, feature selection methods can be divided into supervised methods [15, 16] and unsupervised methods [17, 18]. Supervised feature selection depends on the label

* Corresponding author (email: rechardhan@sina.com)

information to evaluate the feature's correlation. In many practical applications, data on a large scale are often collected without any label information. The processing of manually annotating unlabeled data is time-consuming and expensive [19]. Unsupervised feature selection has gained increasing attention due to its advantage of not requiring label information. Unsupervised feature selection assesses the relevance of features based on their ability to preserve specific data properties. As a consequence, the development of unsupervised feature selection methods is quite promising and demanding. General methods such as the maximum variance method, the information theory-based criteria, the Laplacian score method, the t-test scoring, and the Bayesian scoring function rank original features according to the given evaluation score metrics [20, 21]. However, these methods ignore the potential interaction between features, impacting the accuracy of downstream models. Luo et al. [22] designed a reconstruction graph for characterizing the intrinsic local structure of data and imposed a rank constraint on the Laplacian matrix for selecting features. Since the graph was constructed in the original feature space with noisy and redundant features, it made the predefined graph unreliable and eventually reduced the effectiveness of the selected features [23]. Although the above methods have satisfactory performances, the space for searching informative feature subsets without label guidance is often very large, resulting in a high computational burden and long running time. Moreover, these methods only explore the linear relationship among features, limiting their application to datasets with complex and nonlinear characteristics.

Among unsupervised feature selection methods, the autoencoder network (AE), a type of neural network (NN), is increasingly popular due to its favorable performance of automatically exploring nonlinear dependencies among input features [24]. To select the most discriminative features, Yang [25] designed a joint framework by incorporating discriminative analysis and $\ell_{2,1}$ norm. Based on the framework, the feature subset was selected in a batch way. Han et al. [26] utilized a single-layer AE with low-sparsity constraint (AEFS) to reconstruct data and perform feature selection. Due to the extraction of linear and non-linear information between features, the method had a higher capability of feature extraction than traditional methods. Similar to AEFS, Feng [27] leveraged a single-layer AE and graph data regularization (GAFS) to construct an unsupervised feature selection framework. The framework loosened the assumption of the linear manifold. Unfortunately, AEFS and GAFS methods based on single-layer AE cannot model complex non-linear dependencies of features. In addition, Abid et al. [28] designed an end-to-end concrete AE (CAE). CAE utilized concrete random variables and the reparameterization trick to select discrete input features. Jonathan et al. [29] proposed a dynamic mask method that generated instance-wise importance scores for each feature at each time step by fitting a perturbation mask to the input multivariate time series. However, the AE-based feature selection methods ignore the degree of redundancy among feature subsets leading to the selected features being all-important yet highly correlated and reducing the generalization ability of downstream tasks.

Driven by the above ideas, a regularized fractal AE (RFAE) method is proposed. The main contributions are as follows.

(1) The proposed RFAE, an unsupervised feature selection method, reduces the dimensionality of data and improves the generalization capability of downstream models.

(2) A fractal AE network, consisting of a correspondence NN and a selection NN, is designed to capture inter-feature correlations and excavate the diversity of feature subsets.

(3) A redundancy regularization strategy is developed to eliminate the level of redundancy in the selected features and reduce the complexity of the fractal AE network.

The rest of this paper is structured as follows. The related work about unsupervised feature selection is presented in Section 2. The developed method is demonstrated in Section 3. Section 4 provides the experiment setup. Section 5 discusses and analyzes the experiment results on six publicly available datasets. Section 6 concludes this work and gives future work.

2 Related work

Unsupervised feature selection has become increasingly popular due to its ability to preserve the inherent meaning of original features without relying on label information [20]. According to different selection strategies, feature selection methods are categorized into three types, namely, filter methods, wrapper methods, and embedded methods.

The filter methods utilize certain measure criteria instead of the error rate to evaluate the importance of features, regardless of the subsequent learning algorithm [21]. Variance score is utilized as a proxy

measure criteria to select features with large variance [30]. However, the variance score method cannot be used to discriminate data in different classes. Mutual information-based algorithms minimize the mutual information among the selected features to select features [31–33]. Other scoring methods, such as the t-test scoring, information theory-based criteria, and Laplacian score, are also commonly employed [34]. Although the filter methods are typically straightforward and computationally efficient, they are not effective for a specific type of downstream application.

Wrapper methods first utilize random search algorithms to generate candidate feature subsets and then evaluate the fitness of the candidate feature subsets based on a specific classifier or learning algorithm such as clustering, decision tree, and naive Bayes methods [35, 36]. Local search operations are devised and embedded in the hybrid genetic algorithm to achieve feature selection [37]. The hybridization technique can improve the final performance of feature selection and control the subset size. Q-algorithm is introduced for optimizing a least-squares objective function [38]. The clustering capability of the input data points is measured by analyzing the spectral properties of the affinity matrix. Compared with filter methods, wrapper methods have higher computation complexity and are superior in accuracy. The reason is that the wrapper methods take into account the properties of the downstream learning task.

Embedded methods incorporate feature selection and learning algorithms into a single process [39]. A paired coupling layer is added to the multi-layer perceptron. The perceptron calculates the filter weights of the coupling layer to rank input features [40]. AE has emerged as a strong tool for feature extraction and is currently being explored for unsupervised feature selection. However, most neural networks are composed of many fully connected layers, resulting in a large number of parameters. Thus, pruning neural networks has become a promising method. Han et al. pruned the unimportant weights based on a pre-trained network and retained the network to select features [41]. However, the minimum computational cost of the method is as much as the cost of training dense networks. Mocanu et al. [42] proposed a sparse evolutionary training algorithm that initializes the training process using a sparse neural network. The algorithm dynamically adjusts the network connections throughout the training phase to automatically model the underlying data distribution. However, the method selected highly correlated features, ignoring the diversity of feature subsets. Thus, the paper aims to research an embedded unsupervised feature selection method, which can take into account the diversity of feature subsets and ensure the method has low parameters.

3 Method

3.1 Unsupervised feature selection

Suppose each sample in a dataset $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]^T \in \mathbb{R}^{m\times n}$ is denoted as $\mathbf{x}_i = [x_{i1}, x_{i2}, \cdots, x_{im}]^T \in \mathbb{R}^{m\times 1}$, where m and n are the numbers of features and samples, respectively. To reduce the dimension of the dataset, feature selection refers to selecting an informative subset of size k ($k < m$) from the whole feature space. The selected features are required to well represent the original features. In addition, due to the difficulty and expense of obtaining labels, feature selection needs to be performed in an unsupervised manner. Thus, unsupervised feature selection is formalized as

$$\min_{\mathcal{D}_k, F} \|F(\mathbf{X}_{\mathcal{D}_k}) - \mathbf{X}\|^2, \tag{1}$$

where $\mathcal{D}_k$ is the subset of the specified k features, $\mathbf{X}_{\mathcal{D}_k}$ is the derived dataset from original data $\mathbf{X}$ based on $\mathcal{D}_k$, and $F(\cdot)$ is the mapping from the space spanned by $\mathbf{X}_{\mathcal{D}_k}$ to $\mathbb{R}^{m\times n}$ without any label information. From the formulation, we need to design an effective method to approximate the solution of Equation (1) for unsupervised feature selection. Table 1 summarizes major notations used in the rest of this paper.

3.2 Overall framework of the proposed RFAE

To mine the diversity of features and reduce the computational complexity, RFAE, an unsupervised feature selection, is proposed. RFAE is framed by a fractal AE network and optimized by a redundancy regularization strategy, as shown in Figure 1. Specifically, the fractal AE network is composed of a correspondence NN (CoNN) and a selection NN (SeNN). The CoNN and the SeNN extend AEs by adding an extractor layer and a selector layer between the input layer and the hidden layer, respectively.

Table 1 List of notations.

Notation	Definition	Notation	Definition	Notation	Definition
$\mathbf{X}$	Original data	n	The number of samples	η	The learning rate
m	The number of features	$\mathbf{X}_{\mathcal{D}_k}$	The derived dataset based on $\mathcal{D}_k$	w_i^1	The ith value of $\mathbf{w}^1$
k	The number of selected features	$\mathcal{D}_k$	The subset of the selected features	θ	The threshold parameter
$F(\cdot)$	The mapping function	$\mathbf{w}^1$	The weight vector of the extract layer	sgn$(\cdot)$	The compliant function
p	The number of hidden nodes	$f(\cdot)$	The nonlinear activation function	diag $(\cdot)$	The diagonal function
$\mathbf{W}$	The weight matrix	$g(\cdot)$	The nonlinear activation function	v_{ij}	The value of the matrix $\mathbf{V}$
$\mathbf{V}$	The weight matrix	$\mathbf{w}^{1kmax}$	The largest k values of $\mathbf{w}^1$	α	The hyperparameter
ρ_{ij}	The mutual information	β	The hyperparameter	φ	The hyperparameter

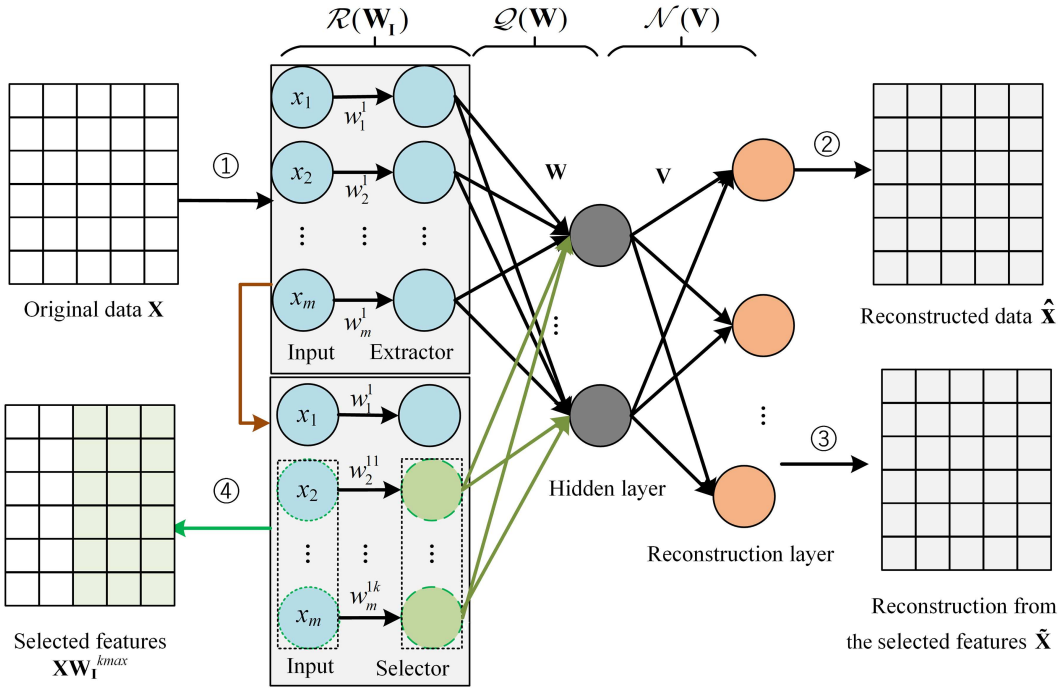

Figure 1 (Color online) Diagram for the regularized fractal AE. During training, at the initial iteration, the weights between the input and extractor layers are initialized in a range from 0.999 to 0.9999 and the weights between the extractor and output layers are initialized in a range from 0 to 1. Then, the original data $\mathbf{X}$ are fed into the CoNN and a reconstruction error can be calculated. At the next iteration, the top k features with the largest weights in the extractor layer are selected as inputs to the selector layer in the SeNN. The CoNN and the SeNN are trained by optimizing an objective function with three regularization terms. During testing, the feature index $\mathbf{w}^{1kmax}$ is utilized to select the new sample $\mathbf{X}'$. The presented values are 1) original input; 2) reconstruction result from the CoNN; 3) reconstruction result from the SeNN; and 4) feature selection result.

The redundancy regularization strategy aims to add a redundancy elimination regularization term and two sparse regularization terms to the reconstruction errors of the fractal AE network.

3.3 Fractal autoencoder network

The AE structure is utilized as the framework for the fractal AE network. To facilitate the identification of representative features and mine the diversity of features, the fractal AE network, consisting of a CoNN and a SeNN, merges feature extraction and feature selection into an end-to-end model.

1) Correspondence NN: To perform feature extraction in the original space, the CoNN extends the AE by adding an extractor layer. Specifically, the CoNN consists of an extractor layer, a hidden layer, and a reconstructed layer, which is used for performing feature representation and extracting inter-feature correlations. The extractor layer is a one-to-one layer, which is utilized to weigh the importance of each input feature. The CoNN is formulated as

$$\hat{\mathbf{X}} = g(f(\mathbf{W_I X})), \mathbf{W_I} \geqslant 0, \tag{2}$$

where $\mathbf{W}_\mathrm{I} = \mathrm{diag}\left(\mathbf{w}^1\right) \in \mathbb{R}^{m\times m}$, $\mathbf{w}^1 = [w_1^1, w_2^1, \cdots, w_i^1] \in \mathbb{R}^m$ is the weight vector between the input and

extractor layers. The values of $\mathbf{W}_{\mathrm{I}}$ are required to be nonnegative because they reflect the importance of features and the nonnegative constraint would make their interpretation more meaningful. $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_p]^T \in \mathbb{R}^{p\times m}$ is the input weight matrix between the extractor and hidden layers, p is the initial number of hidden nodes, and $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_p]^T \in \mathbb{R}^{m\times p}$ is the weight matrix between the hidden and output layers. $f(\cdot)$ and $g(\cdot)$ are nonlinear activation functions, $\mathrm{diag}(\cdot)$ is a diagonal transformation function.

2) Selection NN: Directly calculating the pairwise interactions of all original features needs an $m \times m$ weight matrix, which is computationally expensive for high-dimensional data. Taking into account high-order interactions between features would require higher complexities. To reduce the computational complexity and exploit the diversity of the feature subset, the SeNN is designed by adding a selector layer between the input layer and the hidden layer. Specifically, the SeNN consists of a selector layer, a hidden layer, and an output layer. Thus, the CoNN and the SeNN share the same hidden and output layers. Under the guidance of the CoNN, we sort all features according to the weight $\mathbf{w}^1$ and select the top k features as the input of the selector layer. Thus, the SeNN is formulated as

$$\tilde{\mathbf{X}} = g(f(\mathbf{W}_{\mathbf{I}}^{kmax}\mathbf{X})), \tag{3}$$

where $\mathbf{W}_{\mathbf{I}}^{kmax} = \mathrm{diag}(\mathbf{w}^{1kmax})$, $\mathbf{w}^{1kmax}$ is the largest k values of the weight vector $\mathbf{w}^1$.

The fractal AE network has two benefits. First, the network does not depend on any probability distribution assumption and can be applied to different datasets. Second, the CoNN aims to extract global information and reconstruct original data. The SeNN aims to select informative features with the guidance of the CoNN. The CoNN and SeNN are merged as a model, which makes the fractal AE network concise in architecture and easily applicable to different downstream tasks, such as clustering and classification.

3.4 Redundancy regularization strategy

To reduce the complexity of the network and ensure the diversity of the feature subset, the redundancy regularization strategy is proposed.

1) Sparse regularization: To sparsity the weights of the extractor layer, the ℓ_1 norm is applied. The ℓ_1 norm induces sparsity and shrinks the less important features' weights to 0, and it may make the features more discriminative as well. Thus, the feature weight regularization is formulated as

$$\mathcal{R}(\mathbf{W}_{\mathbf{I}}) = \|\mathbf{W}_{\mathbf{I}}\|_1 = \sum_{i=1}^{m} \left|w_i^1\right|. \tag{4}$$

The ℓ_1 regularization term encourages sparsity in the feature weight vector, but it does not impose any constraint on the dimension of the hidden layer of the fractal AE network. The size of the hidden layer influences the performance of the network. Having too many hidden nodes can increase training time and lead to poor generalization due to overfitting. Conversely, if there are too few hidden nodes, the network has inadequate representation ability. Thus, it is crucial to determine the size of the hidden layer for achieving optimal network performance. The penalty of hidden weights $\mathbf{V}$ is imposed on the group lasso penalty and is formulated as

$$\mathcal{N}(\mathbf{V}) = \sum_{j=1}^{m} \sqrt{\sum_{i=1}^{p} v_{ij}^2}, \tag{5}$$

where v_{ij} is the weight vector connecting the jth hidden node and the ith output layer node.

2) Redundancy elimination: In the presence of redundant features, if not appropriately considered, the selected features may have high correlations leading to higher training costs and impacting the generalization performance. The redundancy elimination regularization term based on the dependency between features is designed as a penalty to control the level of redundancy among the selected features. Thus, we punish the weights between the extractor and hidden layers in groups. The redundancy elimination regularization is expressed as

$$\mathcal{Q}(\mathbf{W}) = \frac{1}{p(p-1)} \sum_{i=1}^{p} \sum_{j\neq i} \|\mathbf{w}_i\|_2 \|\mathbf{w}_j\|_2 \rho_{ij}^2, \tag{6}$$

where ρ_{ij} is the mutual information coefficient between the ith and jth features and ${}^{1}/_{p(p-1)}$ is a normalization factor.

3) Objective function: Summing up the above three parts, the objective function of the proposed RFAE is expressed as

$$\min_{\mathbf{W_I},\mathbf{W},\mathbf{V}} \underbrace{\left\|\mathbf{X}-\hat{\mathbf{X}}\right\|_F^2+\left\|\mathbf{X}-\tilde{\mathbf{X}}\right\|_F^2}_{\mathcal{L}_1}+\underbrace{\alpha\mathcal{R}(\mathbf{W_I})+\beta\mathcal{N}(\mathbf{V})}_{\mathcal{L}_2}+\underbrace{\varphi\mathcal{Q}(\mathbf{W})}_{\mathcal{L}_3}, \tag{7}$$

where α, β, φ are non-negative balancing parameters.

The terms $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{L}_3$ are designed for different goals. $\mathcal{L}_1$ is the mean absolute error term which aims to minimize the reconstruction errors between the input and output of the fractal AE network. $\mathcal{L}_2$, consisting in the ℓ_1 regularization terms and a group lasso penalty, is used to sparse the weights of the extractor layer and the hidden layer, which simplifies the structure of the network and reduces the network complexity. $\mathcal{L}_3$ enables the network's ability to control redundant feature selection.

3.5 Optimization

Since the proposed RFAE extends AEs for implementing feature selection, whose parameters are learned through huge samples. Therefore, we use the gradient descent method to optimize the objective function given in Equation (7). The weights $\mathbf{W_I}$, $\mathbf{W}$, $\mathbf{V}$ are updated at the $(t+1)$th iteration using

$$\mathbf{W_I}^{t+1}=\mathbf{W_I}^t-\eta\frac{\partial\mathcal{L}}{\partial\mathbf{W_I}^t}, \tag{8}$$

$$\mathbf{W}^{t+1}=\mathbf{W}^t-\eta\frac{\partial\mathcal{L}}{\partial\mathbf{W}^t}, \tag{9}$$

$$\mathbf{V}^{t+1}=\mathbf{V}^t-\eta\frac{\partial\mathcal{L}}{\partial\mathbf{V}^t}, \tag{10}$$

where $\eta>0$ is the learning rate, $\frac{\partial\mathcal{L}}{\partial\mathbf{W_I}^t}$, $\frac{\partial\mathcal{L}}{\partial\mathbf{W}^t}$, $\frac{\partial\mathcal{L}}{\partial\mathbf{W}^t}$ are the partial derivatives of the objective function $\mathcal{L}$ with respect to $\mathbf{W_I}^t$, $\mathbf{W}^t$, $\mathbf{V}^t$, which are given as

$$\frac{\partial\mathcal{L}}{\partial\mathbf{W_I}^t}=\frac{\partial E}{\partial\mathbf{W_I}^t}+\alpha\frac{\partial\mathcal{R}(\mathbf{W_I})}{\partial\mathbf{W_I}^t}, \tag{11}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{V}^t}=\frac{\partial E}{\partial\mathbf{V}^t}+\beta\frac{\partial\mathcal{N}(\mathbf{V})}{\partial\mathbf{V}^t}, \tag{12}$$

$$\frac{\partial\mathcal{L}}{\partial\mathbf{W}^t}=\frac{\partial E}{\partial\mathbf{W}^t}+\varphi\frac{\partial\mathcal{Q}(\mathbf{W})}{\partial\mathbf{W}^t}, \tag{13}$$

where $E\left(\mathbf{W_I},\mathbf{W},\mathbf{V}\right)=\left\|\mathbf{X}-\hat{\mathbf{X}}\right\|_F^2+\left\|\mathbf{X}-\tilde{\mathbf{X}}\right\|_F^2$. Since $\mathbf{W}=[\mathbf{w}_1,\mathbf{w}_2,\cdots,\mathbf{w}_p]^T\in\mathbb{R}^{p\times m}$ and $\mathbf{V}=[\mathbf{v}_1,\mathbf{v}_2,\cdots,\mathbf{v}_p]^T\in\mathbb{R}^{m\times p}$, the parts $\frac{\partial\mathcal{R}(\mathbf{W_I})}{\partial\mathbf{W_I}^t}$, $\frac{\partial\mathcal{N}(\mathbf{V})}{\partial\mathbf{V}^t}$, and $\frac{\partial\mathcal{Q}(\mathbf{W})}{\partial\mathbf{W}^t}$ in (11) and (12) can be rewritten in vector forms as follows:

$$\frac{\partial\mathcal{R}}{\partial w_i^{1t}}=\operatorname{sgn}\left(w_i^{1t}\right), \tag{14}$$

$$\frac{\partial\mathcal{N}}{\partial\mathbf{v}_j^t}=\frac{\mathbf{v}_j^t}{\left\|\mathbf{v}_j^t\right\|_2}, \tag{15}$$

$$\frac{\partial\mathcal{Q}}{\partial\mathbf{w}_i^t}=\frac{\mathbf{w}_i^t\sum_{j\neq i}\left\|\mathbf{w}_j^t\right\|_2\rho_{ij}^2}{p(p-1)\|\mathbf{w}_i^t\|_2}, \tag{16}$$

where $i=1,2,\ldots,p$, $j=1,2,\ldots,m$, and $\operatorname{sgn}(\cdot)$ is the compliant function.

It is obvious that the gradient vectors in Equations (14)–(16) are naturally invalid once w_i^{1t}, $\mathbf{w}_i^t$, and $\mathbf{v}_j^t$ are equal to zero, which means the redundancy regularization strategy in Equations (4)–(6) are not differentiable at the origin. Therefore, the subgradient descent and a smooth differentiable approximating function are used to address the issue associated with the non-differentiable nature of the penalties.

Algorithm 1 Regularized fractal autoencoder.

Require: Training dataset $\mathbf{X}$, test dataset $\mathbf{X}'$, parameters α, β, and φ, learning rate lr, the node number of the hidden layer d, and the number of selected features k.
Ensure: The k most representative features.
Initialize: $\mathbf{W_I} = []$, $\mathbf{W} = []$, $\mathbf{V} = []$, $maxiter$=100, and $iter$=1;
Train the fractal autoencoder network;
while not converged and $iter \leqslant maxiter$ **do**
  if $iter < 2$ **then**
    Initialize $\mathbf{w}^1$ by randomly taking numbers from [0.999, 0.9999];
    Initialize $\mathbf{W}$ and $\mathbf{V}$ by randomly taking number from [0, 1];
    $\mathcal{L} \leftarrow \left\|\mathbf{X} - \hat{\mathbf{X}}\right\|_F^2$;
    Compute the gradient of the loss $\mathcal{L}$ performing standard forward propagation;
    Update the network parameter $\mathbf{W_I}$, $\mathbf{W}$, and $\mathbf{V}$ by backpropagation;
  else
    $\mathbf{w}^{1kmax} \leftarrow \arg\max_k(\mathbf{w}^1)$;
    $\mathcal{L} \leftarrow \left\|\mathbf{X} - \hat{\mathbf{X}}\right\|_F^2 + \left\|\mathbf{X} - \tilde{\mathbf{X}}\right\|_F^2 + \alpha\mathcal{R}(\mathbf{W_I}) + \beta\mathcal{N}(\mathbf{V}) + \varphi\mathcal{Q}(\mathbf{W})$;
    Compute the gradient of the loss $\mathcal{L}$ performing standard forward propagation;
    Update the network parameter $\mathbf{W_I}$, $\mathbf{W}$, and $\mathbf{V}$ by backpropagation;
  end if
  $\mathbf{w}^{1kmax} \leftarrow \arg\max_k(\mathbf{w}^1)$;
end while
Use the trained network for feature selection;
for $i \leftarrow 1$ to k **do**
  $s^{(j)} \leftarrow \mathbf{w}_{(j)}^{1kmax}$, where j indexes the elements for the sample vector;
  $\mathbf{X}'_{\mathcal{D}_k} \leftarrow \mathbf{X}'_{s^{(j)}}$;
end for

First, the subgradient descent is selected to optimize w_i^{1t} and is expressed as

$$w_i^{1t+1} = \begin{cases} w_i^{1t} - \eta, & w_i^{1t} > 0, \\ \max(w_i^{1t} - \eta, 0), & w_i^{1t} = 0. \end{cases} \tag{17}$$

Second, the smooth differentiable approximating function is selected to optimize $\mathbf{w}_i^t$ and $\mathbf{v}_j^t$. The smoothing function $H(\mathbf{u})$ of a vector $\mathbf{u}$ is defined as

$$H(\mathbf{u}) = \begin{cases} \|\mathbf{u}\|_2, & \|\mathbf{u}\|_2 \geqslant 0, \\ \sqrt{\|\mathbf{u}\|_2^2 + \theta} + \theta - \sqrt{\theta^2 + \theta}, & \|\mathbf{u}\|_2 < 0, \end{cases} \tag{18}$$

where $\theta > 0$ is a threshold parameter. The gradient of the smoothing function is continuous, which is derived as

$$\frac{\partial H(\mathbf{u})}{\partial \mathbf{u}} = \begin{cases} \frac{\mathbf{u}}{\|\mathbf{w}\|_2}, & \|\mathbf{u}\|_2 \geqslant \theta, \\ \frac{\mathbf{u}}{\sqrt{\|\mathbf{u}\|_2 + \theta}}, & \|\mathbf{u}\|_2 < \theta. \end{cases} \tag{19}$$

Accordingly, $\left\|\mathbf{v}_j^t\right\|_2$ and $\|\mathbf{w}_i^t\|_2$ are replaced by $H(\mathbf{v}_j^t)$ and $H(\mathbf{w}_i^t)$. Thus, the gradients of group lasso penalties with respect to $\mathbf{w}_i$ and $\mathbf{v}_j$ are rewritten as

$$\frac{\partial \mathcal{N}}{\partial \mathbf{v}_j^t} = \frac{\partial H(\mathbf{v}_j^t)}{\partial \mathbf{v}_j^t}, \tag{20}$$

and the gradient of the redundancy elimination term is expressed as

$$\frac{\partial \mathcal{Q}}{\partial \mathbf{w}_i} = \begin{cases} \frac{2}{p(p-1)} \frac{\mathbf{w}_i}{\|\mathbf{w}_i\|_2} \sum_{k=1,k\neq i}^{p} H(\mathbf{w}_k)\rho_{ij}^2, & \|\mathbf{w}_i\|_2 \geqslant \theta, \\ \frac{2}{p(p-1)} \frac{\mathbf{w}_i}{\sqrt{\|\mathbf{w}_i\|_2 + \theta}} \sum_{k=1,k\neq i}^{p} H(\mathbf{w}_k)\rho_{ij}^2, & \|\mathbf{w}_i\|_2 < \theta. \end{cases} \tag{21}$$

Algorithm 1 summarizes the execution process of the proposed RFAE.

4 Experimental setup

4.1 Datasets description

Six publicly available datasets are chosen to evaluate the performance of the proposed RFAE. The sample sizes and feature dimensions of six benchmark datasets vary. The statistics of six benchmark datasets

Table 2 The characteristics of six datasets.

	Isolet	Lung	Madelon	MNIST	COIL20	Colon
Type	Text	Text	Aritifical	Image	Image	Image
#Features	217	325	500	784	1024	2000
#Samples	7737	73	2600	70000	1440	62
#Training samples	6237	60	2000	60000	1152	40
#Test samples	1560	13	600	10000	288	22
#Classes	26	7	2	10	20	2

are summarized in Table 2. To improve the learning process of feature selection methods, min-max normalization is required to scale values between zero and one. Each dataset is divided into a training dataset and a test dataset.

Isolet[1)]: The dataset was created with each letter of the English alphabet.

Lung[2)]: The dataset is utilized to describe the survival rate of patients with advanced lung cancer.

Madelon[3)]: The dataset contains 500 features, of which 5 are informative, 15 are linear combinations of informative, and the rest are interfering features without predictive ability.

MNIST[4)]: It consists of 28×28 grayscale images of handwritten digits.

COIL20[5)]: The dataset consists of images of objects from different angles, including objects and backgrounds.

Colon[6)]: The dataset from one of the first successful trials of adjuvant chemotherapy for colon cancer. There are two records per person, one for recurrence and one for death.

4.2 Comparative methods

The existing unsupervised feature selection methods are selected to compare the performance of the proposed RFAE. The brief introductions of comparison algorithms are listed.

1) Baseline: All original features are adopted.

2) Multi-cluster feature selection (MCFS): MCFS selects features by utilizing spectral regression and ℓ_1-norm regularization to preserve the multi-cluster structure of the data as much as possible [43].

3) Nonnegative discriminative feature selection (NDFS): NDFS is a framework for discriminative features. The framework is a combination of nonnegative spectral analysis and linear regression with $\ell_{2,1}$-norm regularization, which aims to identify features that contribute significantly to discriminating between different classes [44].

4) Autoencoder feature selector (AEFS): AEFS leverages the AE regression and the group lasso task to extract both linear and nonlinear information among features [26].

5) Concrete autoencoder (CAE): CAE, an autoencoder with a concrete selector layer, utilizes the concrete random variables and the reparameterization technique to perform the feature selection process [28].

6) Quick selection (QS): It assesses the importance of features based on the strength of neurons within sparse neural networks [45].

7) Reconstruction-based unsupervised feature selection (REFS): REFS is designed to embed the reconstruction function with feature relevance into feature selection [46].

4.3 Evaluation metrics

To evaluate the performances of feature selection methods, we conducted two experiments: clustering using the K-means algorithm and classification using extremely randomized trees. For clustering and classification tasks, accuracy (ACC) is used to measure models' performances. In the clustering task, we employ the K-means algorithm to evaluate the effectiveness of the selected feature subset. Initially, we feed the selected features into the K-means to generate cluster labels. Subsequently, we compare these labels with the class labels to determine the quality of the feature subset. To ensure reliable results, we repeat the K-means algorithm 10 times due to the possibility of converging to a local optimum. By

1) https://archive.ics.uci.edu/dataset/54/isolet.
2) https://jundongl.github.io/scikit-feature/datasets.html.
3) https://archive.ics.uci.edu/dataset/171/madelon.
4) https://archive.ics.uci.edu/dataset/683/mnist+database+of+handwritten+digits.
5) https://github.com/BeiCunNan/Coil20_Image_Classfiation.git.
6) https://jundongl.github.io/scikit-feature/datasets.html.

calculating the average clustering accuracy, we assess the performance of the selected features. Higher clustering accuracy implies better performance of the feature selection. The classification and clustering tasks have the same execution process. The clustering accuracy is defined as

$$\mathrm{ACC} = \frac{\sum_{i=1}^{n} \delta(\mathrm{map}(r_i), l_i)}{n}, \tag{22}$$

where l_i represents the true label of x_i and r_i denotes the clustering result of $\mathbf{x}_i$, n is the total number of samples, The function $\delta(x, y)$ is defined that $\delta(x, y) = 1$ if $x = y$, and $\delta(x, y) = 0$ otherwise. The function $\mathrm{map}(\cdot)$ is the mapping function that rearranges the clustering results to align with the true labels using the Kuhn-Munkres algorithm.

4.4 Parameter settings

We select 50 features from the Isolet, Lung, MNIST, COIL20, and Colon datasets, for which we select just 20 features from the Madolon dataset since most of the features of the Madolon dataset are non-informative noise. There are some parameters of comparison methods to be set in advance. For MCFS, the size of the neighbors k is set to be 5. In NDFS and REFS, the number nearest neighborhood size is set to be 5. For NDFS, the maximum number of iterations is set as 100. For AE-based methods such as AEFS, CAE, QS, and the proposed RFAE, the iteration number in the training process is set as 150, the activation function is set to be the Relu function. And the learning rating is set to be 0.1. For CAE, the initial temperature τ_0 is fixed as 10 and the end temperature τ_E is set as 0.01 in the annealing schedule. To ensure a fair comparison, we carry out a grid search to tune the regularization parameters for all methods and report the best performance achieved by each method.

5 Experiment results and discussion

5.1 Parameter sensitivity

The proposed RFAE method involves three parameters: α, β, and φ, which control the effects of model sparsity and redundancy elimination. In other words, α and β control the sparsity degree of the weight matrices $\mathbf{W_I}$ and $\mathbf{V}$, and the weight metric φ controls the degree of redundancy among the selected features. When α is large, the selector layer tends to keep fewer features. When β is large, the fractal AE network preserves fewer hidden layer nodes, and when φ is large, redundant features are eliminated as much as possible. To examine the sensitivity of three parameters, we assess the clustering accuracy as a measure of feature selection performance. Specifically, we analyze the sensitivity of the proposed RFAE to three parameters on six datasets. We tune the value of one parameter from $\{10^{-2}, 10^{-1}, 1, 10, 100\}$ while fixing the values of the other two parameters set to be 1. Besides, we tune the size of the set of the selected feature on the grid $\{10, 50, 100, 150, 200\}$ to observe its influence as well. The clustering accuracy of the proposed RFAE is shown in Figure 2.

The parameter sensitivity results of the proposed RFAE on six datasets are depicted in Figure 2. The performance of RFAE is affected differently by the three parameters. In most cases, the clustering accuracy of the proposed RFAE shows a trend from ascending to descending for α, β, and φ. The main reason is that a small value of the parameter is almost the same as eliminating the regularization term from the objective function, resulting in a simpler method that may not perform effectively. However, a large value of the parameter would render other terms in the objective function insignificant, distorting the purpose of feature selection and ultimately degrading performance. In addition, for Madelon and COIL20 datasets, the proposed RFAE is sensitive to parameter φ, while parameters α and β have a relatively lesser effect compared to that of the parameter φ. On the MNIST and Colon datasets, the proposed RFAE is more sensitive to parameter α, indicating that the sparsity aspect controlled plays a crucial role in optimizing the feature selection process. Moreover, for Isolet and Lung datasets, the sensitivity pattern differs, which indicates that the performance sensitivity of the proposed RFAE varies depending on the specific dataset characteristics.

5.2 Convergence property and training accuracy

To analyze the convergence property and training accuracy of the proposed RFAE, we show the changes in the objective function values and clustering accuracy for the increase in the number of iterations. As

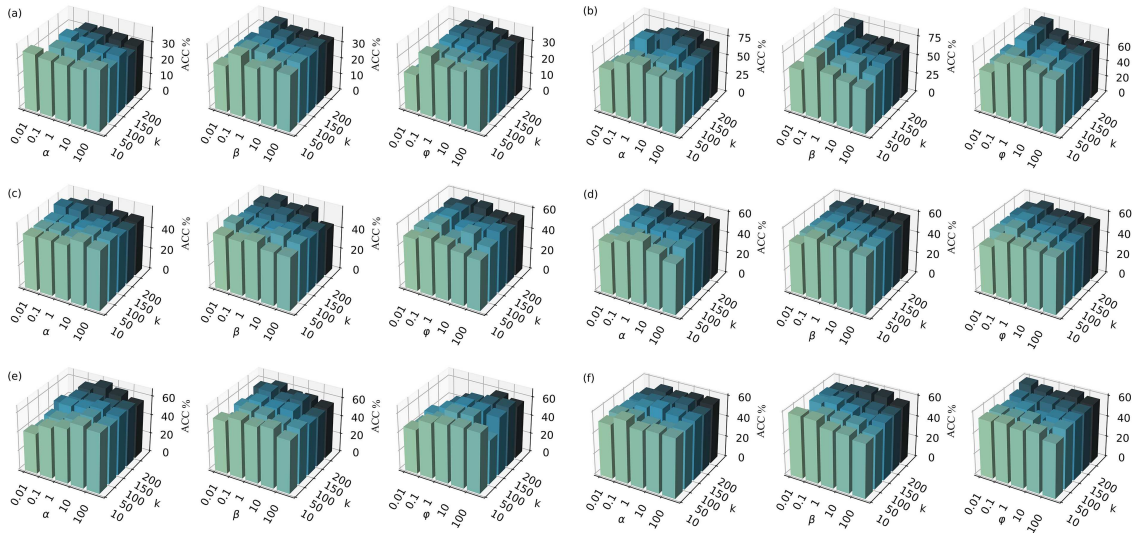

Figure 2 (Color online) Clustering accuracy of the proposed RFAE on six datasets with different α, β, φ values. (a) Isolet; (b) Lung; (c) Madelon; (d) MNIST; (e) COIL20; (f) Colon.

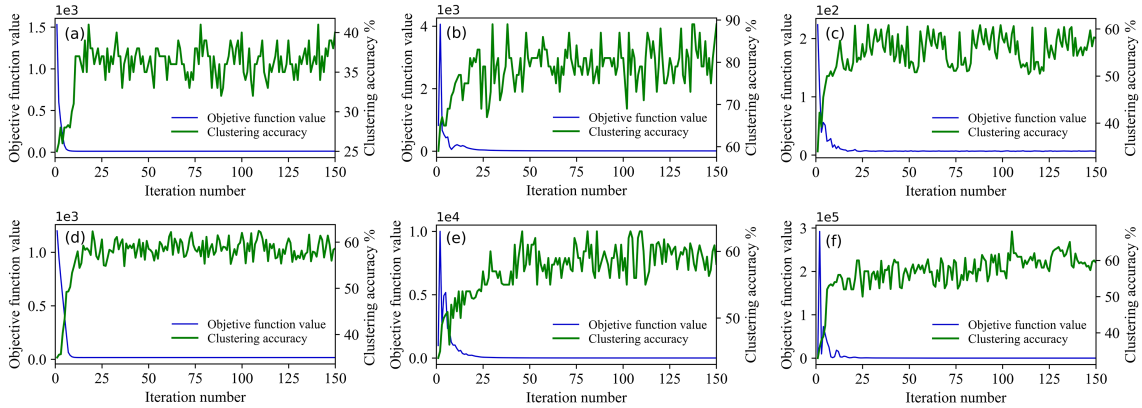

Figure 3 (Color online) Convergence curves and training accuracy of the proposed RFAE on six datasets. (a) Isolet; (b) Lung; (c) Madelon; (d) MNIST; (e) COIL20; (f) Colon.

shown in Figure 3, the objective function value of the proposed RFAE decreased very fast in the first 20 iterations. Then, the objective function value declines slowly and converges to a local minimum value on all datasets, demonstrating that the proposed optimization method is effective and stable.

Moreover, it is well observed that the clustering accuracy of the proposed RFAE tends to be a stable value from Figure 3. There are fluctuations as the samples for each iteration are randomly produced from the original feature space.

5.3 Comparisons with other methods

5.3.1 *Clustering and classification performance analysis*

To evaluate the performance of the proposed RFAE for downstream tasks, the proposed RFAE is compared to the given seven comparison methods. The clustering and classification accuracy results are summarized in Table 3.

The following observations have been obtained from Table 3. First, feature selection is necessary and effective. Compared with the baseline method (all features), feature selection significantly reduces the dimension of features by removing noisy and redundant information. Besides, feature selection makes the clustering and classification more efficient by selecting a subset of the original features, which is important for high-dimensional data processing. Second, MCFS and NDFS achieve better performance in some cases because they explore the local geometric structure of the data distribution making more accurate clustering than other comparison methods. In addition, NDFS achieves higher ACC than

Table 3 Performance comparison in terms of clustering accuracy and classification accuracy. The best results are highlighted in bold.

Task	Dataset	Baseline	MCFS [43]	NDFS [44]	AEFS [26]	CAE [28]	QS [45]	REFS [46]	RFAE
Clustering	Isolet	30.8±0.01	20.6±0.5	32.5±0.5	31.0±2.7	31.6±3.1	32.5±2.8	30.1±0.8	**37.1±1.3**
	Lung	69.7±0.01	68.0±5.8	81.4±0.9	70.0±0.3	75.0±2.4	67.7±1.5	72.1±4.3	**82.0±4.3**
	Madelon	57.5±0.01	58.0±0.05	58.0±0.08	50.8±0.2	56.9±3.6	57.5±5.8	50.1±0.01	**58.7±4.7**
	MNIST	46.2±0.01	56.5±4.1	58.2±3.2	50.8±0.2	56.9±3.6	57.5±3.8	53.6±1.3	**59.7±2.9**
	COIL20	**63.7±0.01**	56.3± 1.3	50.0 ± 1.2	51.2 ± 1.7	60.0 ± 1.1	59.5 ± 2.1	26.3 ± 0.8	59.8 ± 3.8
	Colon	55.0±0.01	52.5± 1.0	55.0 ± 0.7	52.3 ±1.5	53.4 ±1.3	54.0 ±1.7	53.5 ±1.3	**58.0 ± 2.9**
Classification	Isolet	**58.5±0.01**	5.7±0.6	56.5±1.6	53.7±3.8	56.9±3.2	58.3±2.7	50.1±2.9	57.5±1.4
	Lung	52.2±0.01	46.4±4.1	52.1±3.2	50.9±4.2	53.2±2.7	52.9±1.8	46.4±4.1	**53.6±5.4**
	Madelon	50.7±0.01	54.5±0.92	68.9±0.92	82.1±2.8	87.5±2.0	86.6±2.0	49.1±1.9	**92.3±0.42**
	MNIST	**94.1±0.01**	56.5±4.1	82.1±3.2	52.1±2.8	87.5±2.0	86.6±3.6	57.3±8.1	89.3±5.2
	COIL20	93.7±0.01	88.9 ± 2.1	84.3 ±0.6	93.0 ± 2.7	94.3 ± 2.3	94.8 ± 2.6	73.6 ± 2.7	**95.1± 3.1**
	Colon	72.7±0.01	72.5± 1.2	59.1 ± 0.7	52.7 ± 1.2	53.8 ± 1.3	59.1 ± 2.1	**86.3 ±1.7**	75.5 ±2.5

MCFS. The main reason is that NDFS can evaluate features jointly, while MCFS selects features one after another. Third, for AE-based feature selection methods, AEFS, CAE, and QS have poor clustering performance on the Madelon and Lung datasets, respectively. However, CAE and QS show decent classification performances on other datasets considered. The main reason is that AEFS only depends on exploiting the $\ell_{2,1}$-norm regularization on the weights of AE to select features, which do not consider diversity. CAE uses a probability distribution on the concrete selector layer and selects features by corresponding parameters. The parameters in the concrete selector layer may select the same or redundant features. Moreover, REFS obtains better clustering and classification on the Lung dataset by taking into account both the reconstruction ability and preserving ability. Finally, the proposed RFAE has superior performance compared to other methods in most cases. There exist some cases in which the proposed RFAE does not provide the best performance (e.g. Isolet classification), but it defeats all its competitors in cluster tasks. Therefore, we can say that the proposed method exhibits the best overall performance among the methods.

Specifically, the superiority of the proposed RFAE is mainly attributed to the following reasons. First, compared to CAE and AEFS, the extractor layer and the selector layer adopted in the proposed RFAE explore the global representation information and mine the local diversity of features, respectively. Second, the $\ell_{2,1}$ regularization term is designed to reduce the redundant and noisy features. Compared to NDFS and AEFS, the proposed RFAE additionally imposes feature redundancy information into the objective function, making the selected feature more informative. Third, MCFS and REFS have different regularization parameters. The proposed RFAE designs three regularization terms that aim at integrating network structure simplification and unsupervised feature selection into a single framework.

5.3.2 *Feature selection evaluation*

Since evaluating the performances of feature selection methods using a single value of k may be insufficient for robust comparison, we conduct an additional experiment in which k is varied from values between 10 and 150. Due to space limits, we only report the results in terms of clustering accuracy on the Isolet and Lung datasets, as shown in Figure 4.

The clustering performance increases with the increase in the number of features. When the number of features exceeds a certain value, clustering accuracy does not always increase with the increase of k. It indicates that a large number of features are not always helpful for clustering, which is caused by the addition of redundancy when more features are selected. Moreover, the clustering performance of the proposed RFAE is stable and achieves consistently good performance. Note that during the training process, the number of features k needs to be analyzed according to the specific dataset by experimental evaluation. After determining the number of informative features k, the proposed RFAE does not require retraining the fractal AE network to change its structure or parameters. Generally speaking, once the fractal AE is trained, feature selection is performed on the test dataset based on the first k weight values in the extractor layer.

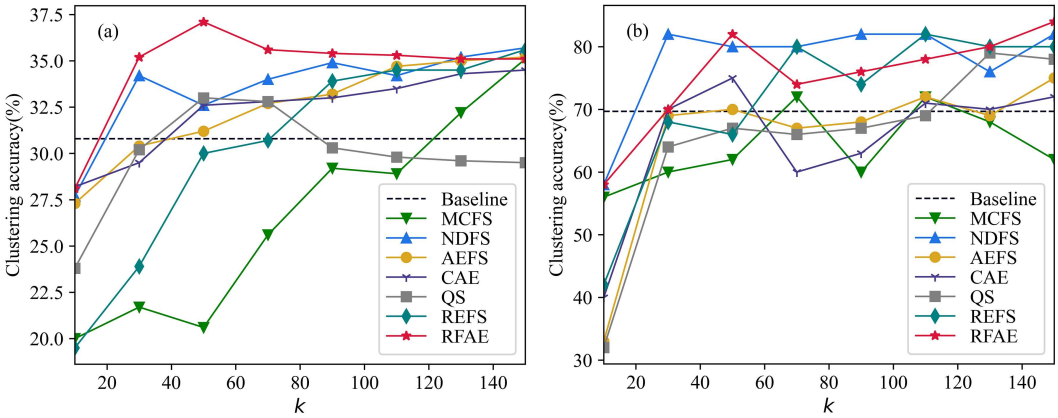

Figure 4 (Color online) Comparison of clustering accuracy for various values of k. (a) Isolet dataset; (b) Lung dataset.

Table 4 The comparison of computational efficiency.

Metric	Method	Isolet	Lung	Madelon	MNIST	COIL20	Colon
Running time (s)	MCFS [43]	379	0.69	71.6	6.8×10^6	3.9	0.72
	NDFS [44]	68	0.70	6.0	3.3×10^4	8.4	11.9
	AEFS [25]	38	24	63	3.3×10^2	550	200
	CAE [28]	20	40	68	7.2×10^3	935	312
	QS [45]	35	36	59	2.9×10^2	824	246
	REFS [46]	396	56	1245	6.3×10^6	5060	7974
	RFAE	28	25	110	4.9×10^3	3454	2031
MFLOPs	AEFS [25]	0.22	0.28	0.39	0.56	1.29	0.70
	CAE [28]	0.12	0.20	0.27	0.48	1.26	0.61
	QS [45]	0.17	0.28	0.37	0.52	1.28	0.66
	RFAE	0.14	0.21	0.32	0.51	1.28	0.65

5.3.3 *Computational efficiency*

Two metrics including the average running time and floating point operations per second (FLOPs) are selected to verify the computational efficiency, as shown in Table 4. To ensure the fairness of the experiment, we measure feature selection methods using the same CPU core.

From Table 4, compared to AE-based methods such as AEFS, QS, and CAE, the proposed RFAE takes a longer time to select informative features of six datasets. The main reasons are that the proposed RFAE has two sub-NNs, while AEFS, QS, and CAE are single-layer networks, and the proposed RFAE needs to give the ranking of the features as the input of the SeNN in each iteration. In addition, with the increases due to the need to compute the mutual information between the variables to estimate the redundancy elimination terms. Consequently, the proposed RFAE requires more time to perform feature selection. Moreover, the proposed RFAE takes a shorter running time than REFS, MCFS, and NDFS on the Isolet and MNIST datasets. FLOPs estimate the time complexity of an algorithm independently of its implementation. Compared to QS and AEFS, the proposed RFAE is advantageous in the FLOPs metric. Taking into account the clustering and classification results in Table 3 and Figure 4, the running time and FLOPs of the proposed RFAE can be acceptable and the inference speed can be reliably guaranteed.

6 Conclusion

In this paper, a regularized fractal AE (RFAE) method, consisting of a fractal AE network and a redundancy regularization strategy, was proposed to select informative features from datasets with high-dimensional and noisy features. In the proposed RFAE, the fractal AE network was able to explore nonlinear representative information using the CoNN and mine local diversity using the SeNN. The redundancy regularization strategy, consisting of a sparse regularization term and a redundancy elimination term, was utilized to eliminate the selection of redundant features and sparse the weights of the fractal AE network. Extensive experiment results demonstrated that the proposed RFAE achieved acceptable

clustering accuracy, classification accuracy, and computational efficiency. In the future, a parameter optimization method may be further explored based on this work to reduce network running time.

Acknowledgements This work was supported by National Key Research and Development Project (Grant Nos. 2022YFB 3305800-5, 2023YFB3307300), National Natural Science Foundation of China (Grant Nos. 62125301, 62373014, 92267107), and Beijing Youth Scholar (Grant No. 037).

References

1 Tang C, Zheng X, Zhang W, et al. Unsupervised feature selection via multiple graph fusion and feature weight learning. Sci China Inf Sci, 2023, 66: 152101
2 Yu H, Wu J. A unified pruning framework for vision transformers. Sci China Inf Sci, 2023, 66: 179101
3 Han H G, Li Y, Du Y P. Adaptive price adjustment method for used mobile phone based on dual deep fuzzy networks. Sci China Technol Sci, 2022, 65: 1330–1337
4 Liu T Y, Bao J S, Zheng H B, et al. Learning semantic-specific visual representation for laser welding penetration status recognition. Sci China Technol Sci, 2022, 65: 347–360
5 Armanfard N, Reilly J P, Komeili M. Local feature selection for data classification. IEEE Trans Pattern Anal Mach Intell, 2015, 38: 1217–1227
6 Han H G, Sun M T, Wu X L, et al. Double-cycle weighted imputation method for wastewater treatment process data with multiple missing patterns. Sci China Technol Sci, 2022, 65: 2967–2978
7 Han H, Sun M, Li F, et al. Self-supervised deep clustering method for detecting abnormal data of wastewater treatment process. IEEE Trans Ind Inform, 2023, 1: 1–13
8 Gu Y, Liu H, Wang T, et al. Deep feature extraction and motion representation for satellite video scene classification. Sci China Inf Sci, 2020, 63: 140307
9 Mirzaei A, Pourahmadi V, Soltani M, et al. Deep feature selection using a teacher-student network. Neurocomputing, 2020, 383: 396–408
10 Liang Y, Lu S, Weng R, et al. Unsupervised noise-robust feature extraction for aerial image classification. Sci China Technol Sci, 2020, 63: 1406–1415
11 Zhang J, Feng F, Han T, et al. A hybrid method to select morphometric features using tensor completion and F-score rank for gifted children identification. IEEE Trans Knowl Data Eng, 2021, 64: 1863–1871
12 Nie F, Wang Z, Wang R, et al. Submanifold-preserving discriminant analysis with an auto-optimized graph. IEEE Trans Cybern, 2019, 50: 3682–3695
13 Kong Y, Wang T Y, Chu F L. Adaptive TQWT filter based feature extraction method and its application to detection of repetitive transients. Sci China Technol Sci, 2018, 61: 1556–1574
14 Mitra P, Murthy C A, Pal S K. Unsupervised feature selection using feature similarity. IEEE Trans Pattern Anal Machine Intell, 2002, 24: 301–312
15 Xiang S, Nie F, Meng G, et al. Discriminative least squares regression for multiclass classification and feature selection. IEEE Trans Neural Netw Learn Syst, 2012, 23: 1738–1754
16 Gan J, Wen G, Yu H, et al. Supervised feature selection by self-paced learning regression. Pattern Recognit Lett, 2020, 132: 30–37
17 Ang J, Mirzal A, Haron H, et al. Supervised, unsupervised, and semi-supervised feature selection: a review on gene selection. IEEE ACM Trans Comput BI, 2015, 13: 971–989
18 Zhang R, Nie F, Wang Y, et al. Unsupervised feature selection via adaptive multimeasure fusion. IEEE Trans Neural Netw Learn Syst, 2019, 30: 2886–2892
19 Zhao M, Chow T W S, Zhang Z, et al. Automatic image annotation via compact graph based semi-supervised learning. Knowledge-Based Syst, 2015, 76: 148–165
20 Krzanowski W. Selection of variables to preserve multivariate data structure, using principal components. Journal of the Royal Statistical and Society Series C: Applied Statistics, 1987, 36: 22–33
21 Huang R, Jiang W, Sun G. Manifold-based constraint Laplacian score for multi-label feature selection. Pattern Recognit Lett, 2018, 112: 346–352
22 Luo M, Nie F, Chang X, et al. Adaptive unsupervised feature selection with structure regularization. IEEE Trans Neural Netw Learn Syst, 2017, 29: 944–956
23 Lin X, Guan J, Chen B, et al. Unsupervised feature selection via orthogonal basis clustering and local structure preserving. IEEE Trans Neural Netw Learn Syst, 2021, 33: 6881–6892
24 Ma Z, Pan Z, Liu N. A sparse autoencoder-based approach for cell outage detection in wireless networks. Sci China Inf Sci, 2021, 64: 189302
25 Yang Y, Shen H, Ma Z. $\ell_{2,1}$-norm regularized discriminative feature selection for unsupervised learning. In: Proceedings of International Joint Conference on Artificial Intelligence, 2011. 1589–1594
26 Han K, Wang Y, Zhang C, et al. Autoencoder inspired unsupervised feature selection. In: Proceedings of the 2018 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2018. 2941–2945
27 Feng S, Duarte M F. Graph autoencoder-based unsupervised feature selection with broad and local data structure preservation. Neurocomputing, 2018, 312: 310–323
28 Abid A, Balin M, Zou J, Concrete autoencoders: differentiable feature selection and reconstruction. In: Proceedings of the International Conference on Machine Learning, 2019. 444–453

29 Crabbé J, Van Der Schaar M. Explaining time series predictions with dynamic masks. In: Proceeding of the International Conference on Machine Learning, 2019. 2166–2177

30 Zhao Z, Zhang R, Cox J, et al. Massively parallel feature selection: an approach based on variance preservation. Mach Learn, 2013, 92: 195–220

31 Wang Z, Li M, Li J. A multi-objective evolutionary algorithm for feature selection based on mutual information with a new redundancy measure. Inf Sci, 2015, 307: 73–88

32 Zhang L, Chen C, Bu J, et al. A unified feature and instance selection framework using optimum experimental design. IEEE Trans Image Process, 2012, 21: 2379–2388

33 Bugata P, Drotar P. On some aspects of minimum redundancy maximum relevance feature selection. Sci China Inf Sci, 2020, 63: 112103

34 Polat K, Güneş S. A new feature selection method on classification of medical datasets: kernel F-score feature selection. Expert Syst Appl, 2009, 36: 10367–10373

35 Yamada M, Tang J, Lugo-Martinez J, et al. Ultra high-dimensional nonlinear feature selection for big biological data. IEEE Trans Knowl Data Eng, 2018, 30: 1352–1365

36 Han H G, Zhang L L, Hou Y, et al. Adaptive candidate estimation-assisted multi-objective particle swarm optimization. Sci China Technol Sci, 2022, 65: 1685–1699

37 Oh I, Lee J, Moon B. Hybrid genetic algorithms for feature selection. IEEE Trans Pattern Anal Machine Intell, 2004, 26: 1424–1437

38 Wolf L, Shashua A, Geman D. Feature selection for unsupervised and supervised inference: the emergence of sparsity in a weight-based approach. J Mach Learn Res, 2005, 6: 1855–1887

39 Nie F, Huang H, Cai X, et al. Efficient and robust feature selection via joint $\ell_{2,1}$-norm minimization. Advances in Neural Information Processing Systems, 2010, 23: 1–9

40 Lu Y, Fan Y, Lv J, et al. DeepPINK: reproducible feature selection in deep neural networks. Advances in Neural Information Processing Systems, 2018, 1: 31–40

41 Han S, Pool J, Tran J, et al. Learning both weights and connections for efficient neural networks. Advances in Neural Information Processing Systems, 2015, 28: 1–9

42 Mocanu D C, Mocanu E, Stone P, et al. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science. Nat Commun, 2018, 9: 2383

43 Cai D, Zhang C, He X. Unsupervised feature selection for multi-cluster data. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2010, 9. 333–342

44 Li Z, Yang Y, Liu J, et al. Unsupervised feature selection using nonnegative spectral analysis. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2012. 1026–1032

45 Atashgahi Z, Sokar G, Van T, et al. Quick and robust feature selection: the strength of energy-efficient sparse training for autoencoders. Mach Learn, 2022, 12: 1–38

46 Li J, Tang J, Liu H. Reconstruction-based unsupervised feature selection: an embedded approach. In: Proceedings of International Joint Conference on Artificial Intelligence, 2017. 2159–2165