Original Article

Efficient Techniques Based on Sparse Representation for Classifying High-dimensional Multiclass Microarray Data

Maliheh Miri[1], Mohammad Taghi Sadeghi[2], Vahid Abootalebi[2]

[1]Department of Electrical Engineering, University of Saravan, Saravan, Iran, [2]Department of Electrical Engineering, Yazd University, Yazd, Iran

Abstract

Background: Sparse representation (SR) has shown strong performance in classification tasks, particularly for high-dimensional data such as microarray gene expression profiles. These datasets present significant challenges due to their high dimensionality and limited sample size, which often hinder the performance of conventional classifiers. **Methods:** SR addresses this by expressing each signal as a linear combination of a small subset of training samples, reducing computational complexity and improving accuracy. However, using all training samples in the dictionary increases computational cost. This study explores several SR-based classifiers to address microarray data classification, focusing on dictionary construction strategies and sparse coding algorithms. **Results:** Experimental results on the 14-Tumors dataset show that selecting a subset of representative atoms and applying the SL0 algorithm significantly improves both speed and classification accuracy. **Conclusions:** These findings highlight the potential of SR approaches for effective and efficient classification of high-dimensional biological data.

Keywords: *Computational biology, dictionary learning, gene expression, hierarchical classification, high-dimensional data, microarray data classification, sparse representation*

Submitted: 30-May-2025 **Revised:** 08-Oct-2025 **Accepted:** 25-Nov-2025 **Published:** 12-May-2026

Introduction

Sparse representation (SR) of signals has attracted considerable interest in various fields of signal processing, and numerous studies have been conducted in this area.[1-4] In many applications of pattern recognition and computer vision, high-dimensional data, such as face images or microarray data, are frequently encountered.[5-7]

The application of SR of signals to classification problems, known as SR-based classification (SRC),[8] has demonstrated promising results, particularly in face recognition systems. The fundamental concept underlying SRC is that each face image can be expressed as a sparse linear combination of other faces, wherein the highest coefficients in this linear combination correspond to images belonging to the same class as the target image. Consequently, by utilizing these dominant coefficients and setting the remaining ones to zero, an accurate reconstruction of the target image can be achieved. The Euclidean distance metric is subsequently used to measure the similarity between the target image and the reconstructed images for each class. Ultimately, the target image is assigned to the class that minimizes the reconstruction error.

Advancements in microarray technology have enabled the simultaneous monitoring of gene expression levels (gene activity) across thousands of genes, generating vast amounts of microarray data.[9] These data play a crucial role in the diagnosis and classification of various cancer types; however, their most challenging characteristic lies in their exceptionally high dimensionality relative to the limited sample size, making the design of effective classifiers difficult. Considerable efforts have been made to develop classifiers that achieve high accuracy while maintaining low computational complexity.[10-12] However, the majority of these studies have focused on datasets with a small number of classes (typically two or three tumor types) and have employed methods such as Linear Discriminant Analysis,

Address for correspondence:
Dr. Maliheh Miri, Department of Electrical Engineering, University of Saravan, Saravan, Iran.
E-mail: m.miri@saravan.ac.ir

Access this article online

Website: www.jmssjournal.net

DOI: 10.4103/jmss.jmss_52_25

Quick Response Code:

How to cite this article: Miri M, Sadeghi MT, Abootalebi V. Efficient techniques based on sparse representation for classifying high-dimensional multiclass microarray data. J Med Signals Sens 2026;16:13.

Neural Networks, Clustering, Nearest Neighbor (NN), Support Vector Machine (SVM), Decision Trees, and Random Forest. In this study, the 14-Tumors database is examined. This dataset comprises information on fourteen distinct tumor types and is frequently analyzed from a feature selection perspective. The classification problem of 14 tumor types has been addressed by decomposing it into multiple binary classification tasks.[9] Various classifiers are evaluated for this purpose, with the highest performance achieved by the SVM classifier. In binary classification settings, classification is conducted using the One-versus-All and One-versus-One (OVO) strategies, which employ $c(c\text{-}1)/2$ and c hyperplanes to separate the classes, respectively (c representing the number of classes). Neural network-based methods are also applied to classify the 14 tumor types.[13,14] In one proposed approach, two sequential Artificial Neural Network classifiers are utilized for classification. The first classifier identifies the two most probable classes for the test data, while the second classifier makes the final decision among the selected classes.[13] A fuzzy classifier is also introduced that employs fuzzy rules for classification.[15] A method based on SR has been introduced to classify microarray data, where the results demonstrate that the proposed method outperforms various extensions of the SVM classifier.[16]

Despite the extensive research conducted in this field,[17-20] the reported classification performance of existing methods remains insufficient for clinical diagnosis. Motivated by the promising results of SR in gene expression data analysis,[1,3,21-23] this paper explores several classification approaches based on SR to address the challenge of 14-Tumors classification. To ensure a more comprehensive evaluation, three distinct algorithms are employed in the sparse coding stage, each of which computes the SR using a different approach.

The remainder of this paper is structured as follows: First, the fundamental concepts of SR are presented. Then, an overview of conventional sparse coding algorithms is provided. Next, the SR-based classifiers examined in this study are reviewed. In the following section, the dataset used is introduced, and the experimental results are reported. Finally, discussions and conclusions are presented.

Sparse representation fundamentals

In recent years, SR has emerged as a powerful tool for signal representation and compression. Within this framework, a signal can be approximated as a sparse linear combination of a set of training data.[4] Suppose $Y = [Y_1,\ldots,Y_c]$ is formed by concatenating all the training data, such that $Y_1,\ldots,Y_c$ contains the samples from classes 1 to c, respectively. Therefore, a linear combination of the training data can be obtained through a system of linear equations, represented as $y = Ys$, where Y is the dictionary matrix and each of its columns is referred to as an atom.

If the number of equations is smaller than the number of variables (i.e., the number of columns in Y exceeds the number of its rows), the system becomes underdetermined and possesses infinite solutions. However, imposing a sparsity constraint ensures a unique solution. Since the ℓ_0-norm of a vector represents the number of its nonzero elements, minimizing the ℓ_0-norm enforces sparsity. Consequently, the following equation is employed to derive the SR of an input signal using the corresponding training set:

$$\min \|s\|_0 \quad s.t. \quad y = Ys, \tag{1}$$

where y represents the input signal, s denotes the coefficient vector of the linear combination, and $\|s\|_0$ indicates the number of its nonzero elements. A vector s is considered k-sparse if it contains at most k nonzero elements.

Let Y_i denote N_i training data from the i-th class $Y_i = [y_{i,1}, y_{i,2}, \ldots, y_{i,N_i}] \in \mathbb{R}^{D \times N_i}$. If the input signal $y \in \mathbb{R}^D$ belongs to the i-th class, then according to the SR framework, it can be expressed using only the Y_i columns of Y as follows:

$$y = s_{i,1}y_{i,1} + s_{i,2}y_{i,2} + \ldots + s_{i,N_i}y_{i,N_i} = \sum_{j=1}^{N_i} s_{i,j}y_{i,j}, \tag{2}$$

Where $S_{i,j}$ is a scalar value. This forms the fundamental principle of SRC.[8]

Since the ℓ_0-norm is nonconvex, determining the SR using Eq. (1) is an NP-hard problem whose solution relies on combinatorial search, which becomes computationally infeasible in high-dimensional spaces.[24] Various methods have been proposed to address this issue, some of which will be discussed in the following section.

Sparse coding algorithms

A conventional approach to solving linear systems while enforcing sparsity in the solution is to approximate the ℓ_0-norm with a higher-order norm that is more computationally tractable. Consequently, to determine the SR of a signal, the following equation can be utilized, where the sparsity constraint is imposed through ℓ_1-norm minimization:

$$\min \|s\|_1 \quad s.t. \quad y = Ys. \tag{3}$$

Since the ℓ_1-norm is convex, it can be minimized using convex optimization algorithms. The problem can also be formulated as a linear program and solved using linear programming methods, such as Basis Pursuit (BP).[25] It has been shown that if the number of nonzero elements in the sparse solution is below a threshold determined by the correlation among the dictionary atoms, the solution to Eq. (3) coincides with the ℓ_0-norm minimization solution.[24] One of the primary limitations of the BP method is its high computational complexity, which renders its implementation time-consuming; as a result, it is often supplanted by alternative algorithms. Among the fastest algorithms for obtaining an SR is Matching Pursuit (MP).[26]

This greedy and iterative method determines only one coefficient at each step. If s is a k-sparse vector, it can be expressed as a linear combination of k atoms from the dictionary. In this approach, the columns of the dictionary contributing to the linear combination are first identified, and subsequently, the coefficients corresponding to these columns–i.e., the nonzero elements of s–are computed by solving a least squares problem. Therefore, at each step, the dictionary atom most similar to the test sample is selected as the active component of the linear combination, and its corresponding coefficient is computed. The product of this 1-sparse approximation and the dictionary is then subtracted from the test sample to obtain a residual, upon which the process is iteratively repeated. The sum of the 1-sparse approximations obtained at each step, along with the previous approximations, constitutes a new approximation. This iterative process continues until either a predefined number of steps is completed or the residual error falls below a specified threshold. Although this method requires only a simple search at each step and is therefore typically very fast, its greedy nature offers no guarantee that the final solution will coincide with the true sparse solution.

At each step in Orthogonal MP (OMP), an extension of the MP method, the coefficients corresponding to the active columns of the dictionary are computed independently of prior estimates, which are utilized solely to identify the locations of the nonzero elements.[27]

Another method developed to obtain the SR is the Smoothed ℓ_0-norm (SL0) algorithm.[28] In this approach, rather than replacing the ℓ_0-norm with a higher-order norm, the ℓ_0-norm itself is directly minimized. Figure 1a illustrates a univariate ℓ_0-norm function, which equals zero at zero and one elsewhere. As shown, the ℓ_0-norm function is discontinuous. The SL0 method employs a smooth approximation to the ℓ_0-norm, which can be readily optimized. Figure 1b depicts a smooth function–specifically, a symmetric Gaussian function–that approximates the ℓ_0-norm. Denoting this smooth function as $f_1(s)$, the corresponding optimization problem is formulated as follows:

$$\hat{s}_{sz.0} = \arg\min_s \sum_{i=1}^{n} f_i(s_i) \quad \text{s.t.} \quad y = Ys \,. \tag{4}$$

If $f_1(s)$ is not sufficiently smooth, the optimization problem is likely to exhibit numerous local minima. To address this, a highly smooth function (a Gaussian with large variance) possessing a single local minimum is initially selected. After identifying the optimal solution for this smooth function, the result is employed to initialize a Gaussian function with reduced variance, which provides a closer approximation to the ℓ_0-norm. It has been proven that, with an appropriate selection of variances, the solution converges to the ℓ_0-norm minimization solution as the variance approaches zero.

One notable advantage of the SL0 method is its high computational speed. Studies have shown that its convergence rate is two to three times faster than that of conventional algorithms used for solving linear problems. Moreover, since this method directly minimizes the ℓ_0-norm, it is expected to be effective in addressing more complex problems.[28] Additional algorithms commonly employed to compute SRs include homotopy,[29] gradient projection for sparse reconstruction,[30] fast iterative shrinkage-thresholding algorithm,[31] compressive sampling MP,[32] and iterative hard thresholding.[33]

Sparse representation-based classifiers

In this section, several of the most prominent classifiers utilizing SR are reviewed.

Sparse representation-based classification

The method comprises two steps: First, the test sample is sparsely encoded using a dictionary matrix formed by combining all training data from all classes:

$$Y = [Y_1, ..., Y_c] = [y_{1,1}, y_{1,2}, ..., y_{c,N_c}] \,, \tag{5}$$

Here, Y_i is a sub-matrix containing the training samples of class *i*. The SR is obtained using Eq. (3). SRC classifies the test sample by evaluating its similarity to the training data within each class. To this end, the following function is defined:

$$\delta_i(\boldsymbol{s}) = [0, ..., 0, s_{i,1}, ..., s_{i,N_i}, 0, ..., 0]^T \in \mathbb{R}^N \,. \tag{6}$$

By applying this function to the vector s, the elements corresponding to the *i*-th class samples are retained, while the others are set to zero. Consequently, $\hat{y}_i = Y\delta_i(s)$ expresses y as a linear combination of the samples from the *i*-th class. By applying this function to the representation $\boldsymbol{s}$ and reconstructing y using the samples from each class, the test sample is assigned to the class that yields the minimum reconstruction residual:[8]

$$\min_i r_i(y) = \left\| y - Y\delta_i(\hat{s}) \right\|_2 \,. \tag{7}$$

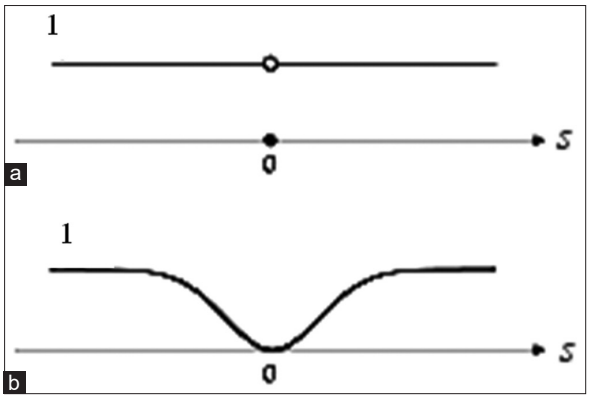

Figure 1: ℓ_0-norm function (a), Smooth function (b)[28]

Collaborative representation-based classification

It is widely accepted that sparsity is a key factor in the success of the SRC method, while the contribution of training samples from different classes to the representation of the input sample has been largely overlooked. It has been demonstrated that when the number of training samples in the dictionary is large relative to the data dimensionality, ℓ_2-norm regularization can achieve competitive performance compared to ℓ_1-norm minimization.[34] Accordingly, the following low-complexity equation is employed to obtain the representation of the input sample by leveraging the contributions of all training samples.

$$\hat{s} = \arg\min_s \|y - Ys\|_2 + \lambda \|s\|_2 . \quad (8)$$

Here, λ is a scalar value. In this formulation, the ℓ_1-norm minimization constraint is replaced by an ℓ_2-norm regularization term. Minimizing the ℓ_2-norm not only imposes a degree of sparsity on the coefficients but also stabilizes the least squares solution. The closed-form solution to the above equation via least squares is given by:

$$\hat{s} = Py; \qquad P = \left(Y^T Y + \lambda I\right)^{-1} Y^T , \quad (9)$$

Where I denotes the identity matrix. Since P is independent of the input data, it can be precomputed. Consequently, when the test sample is provided, it is directly mapped onto the P, resulting in significantly lower computational complexity and faster execution of collaborative representation-based classification (CRC) compared to SRC. The general formulation of collaborative representation classification is given as follows:

$$\hat{s} = \arg\min_s \|y - Ys\|_p + \lambda \|s\|_q . \quad (10)$$

The more restrictive the constraints imposed on the reconstruction residual, the greater the guarantee of classifier stability. Moreover, the application of ℓ_1-norm or ℓ_2-norm minimization constraints on the sparse coding coefficients is directly related to the discriminative power of the representation.[34] Therefore, the SRC method can be regarded as a special case of CRC by setting $p = 2$ and $q = 1$.

Metasample-based sparse representation classification

The core idea behind metasample-based SRC (MSRC) is to represent the test sample as a linear combination of metasamples, which are extracted from the training samples.[35] A metasample itself is a linear combination of training samples that captures their intrinsic properties. Mathematically, the matrix of training samples (the original dictionary) can be factorized into two matrices as follows:

$$Y \sim MH. \quad (11)$$

Here, M is a D × P matrix, where each column defines a metasample, and H is a P × N matrix, where each column encodes the relationship between the metasamples and their corresponding samples in the original dictionary. Several methods have been proposed for extracting metasamples from the original samples. In one method, metasamples are extracted using the Nonnegative Matrix Factorization approach.[36] In another method, Singular Value Decomposition (SVD) is employed,[37] in which the eigenvalues of the training sample matrix are computed and sorted in descending order. Subsequently, a subset of the corresponding eigenvectors is selected to replace the original samples, providing a general representation of the underlying dynamics of the data. To utilize metasamples for classification via SR, the metasamples are determined separately for each class as follows:

$$Y_i \approx M_i H_i, \quad (12)$$

where M_i is a D × P_i matrix, each column represents a metasample of the *i*-th class. The number of metasamples selected for each class can be determined experimentally or through cross-validation. The matrix M is then formed by concatenating the metasamples obtained from each class, as follows:

$$M=[M_1,M_2,\ldots M_c]. \quad (13)$$

Given a new test sample, its SR is computed over the metasample matrix:

$$\min \|s\|_1 \quad s.t. \quad y = Ms . \quad (14)$$

Finally, the test label can be determined using the classification procedure outlined in the SRC method. The primary distinction between SRC and MSRC lies in the construction of the dictionary: while SRC utilizes the original training samples as dictionary atoms, MSRC employs the extracted metasamples for each class. A Parameter-Free MSRC method was also proposed, which addresses the optimal selection of both the number of metasamples and the sparse penalty factor through a weighting strategy.[38]

A tumor classification method, termed Maxdenominator Reweighted SR Classification (MRSRC), has been proposed in which a set of meta-genes is first extracted from the training samples, followed by the application of a reweighted ℓ_1 regularization method to obtain the SR coefficients.[21] Finally, a maxdenominator residual error function is employed for classification. In their study, both binary and multiclass microarray datasets were utilized to evaluate the performance of various classification methods. The results demonstrate that SR-based methods (SRC, MSRC, and MRSRC) outperform other techniques in most experiments, highlighting the effectiveness of these methods in addressing classification problems involving small sample sizes.

K nearest neighbor-based sparse representation-based classification

In high-dimensional data or datasets with a large number of samples, utilizing the entire set of training samples in the dictionary imposes a substantial computational burden

on the sparse coding step and reduces the efficiency of the SRC algorithm. To accelerate the classification process, some methods select a subset of the training data to solve the ℓ_1-norm minimization problem. In these approaches, a subset of atoms from the original dictionary that satisfies a predetermined criterion–depending on the specific application–is chosen to construct the final dictionary. For instance, one study demonstrates that imposing a sparsity constraint on the dictionary atoms yields sparser representations and, consequently, improved performance in speech denoising.[39] Similarly, in another study, classification accuracy was enhanced by selecting sparse atoms to form a more compact dictionary.[40] In the K nearest neighbor (KNN)-SRC method, computational complexity is significantly reduced by selecting the *K* NNs to the test sample and incorporating these selected samples into the classification process.[41]

In this method, the *K* NNs to the test sample are identified using the Euclidean distance metric. The test sample is then sparsely represented using these *K* neighbors. If $\tilde{Y}_i = [\tilde{y}_{i1}, \tilde{y}_{i2}, ..., \tilde{y}_{iK_i}] \in \mathbb{R}_c^{D \times K_i}$ contains K_i neighbors from the *i*-th class, such that $K = \sum_{i=1} K_i$, and the matrix $\tilde{Y}$ includes all *K* NNs, the SR of $\boldsymbol{y}$ is computed as follows:

$$\min \|s\|_1 \quad \text{s.t.} \quad y = \tilde{Y}s . \tag{15}$$

Since $K << N$, y is unlikely to be perfectly reconstructed by $\tilde{Y}s$, the following equation is adopted to estimate the SR:

$$\min \|s\|_1 \quad \text{s.t.} \quad \|y - \tilde{Y}s\|_2 \leq \varepsilon . \tag{16}$$

Following the computation of the SR, classification proceeds as in the SRC method.

Linearly approximated sparse representation-based classification

It has been shown that the most effective training samples in the SR of a signal are not necessarily the nearest samples as determined by the Euclidean distance metric.[42] Therefore, efforts have been made to predetermine the training samples likely to be selected by the ℓ_1-norm minimization process, in order to construct a more concise dictionary. The objective function of the Lagrangian formulation corresponding to Eq. (3) can be expanded as:

$$v(s) = \left\| y - \sum_{i=1}^{N} y_i s_i \right\|_2 + \lambda \sum_{i=1}^{N} |s_i| . \tag{17}$$

Suppose s is k-sparse. Then, for any *i* where $s_i = 0$, it follows that $\|y_i s_i\|_2 = 0, |s_i| = 0$. Consequently, the objective function can be reformulated as:

$$v(\alpha) = \left\| y - \sum_{i=1}^{K} \omega_i \alpha_i \right\|_2 + \lambda \sum_{i=1}^{K} |\alpha_i| , \tag{18}$$

Where ω_i represents a column of Ω. The new dictionary Ω contains only the atoms from the original dictionary that correspond to the nonzero elements of s. Accordingly, the ℓ_1-norm minimization problem, using Ω and the associated coefficient vector α, is formulated as follows:

$$\hat{\alpha} = \arg\min_{\alpha} \| y - \Omega\alpha \|_2 + \lambda \| \alpha \|_1 . \tag{19}$$

Since accurately identifying the exact locations of effective atoms in the SR is a challenging task, they are approximated using ℓ_2-norm minimization. Although the solution obtained via ℓ_2-norm minimization is dense and contains a relatively large number of nonzero elements, it has been shown that the largest peaks tend to occur at positions similar to those identified by ℓ_1-norm minimization.[42] Therefore, the solution obtained from ℓ_2-norm minimization can be used as an initialization point for computing the SR via ℓ_1-norm minimization.

As discussed in the CRC method, replacing the ℓ_1-norm with the ℓ_2-norm in the sparse coding equation results in a least squares formulation, which offers significantly faster computation compared to ℓ_1-norm minimization. Accordingly, in this method, the solution from ℓ_2-norm minimization is thresholded to select the training samples with the largest magnitudes, which are then used to initialize the ℓ_1-norm minimization. This approach is referred to as Linearly Approximated SRC (LASRC), which combines the computational efficiency of CRC with the robustness of SRC, and has demonstrated promising results in face verification systems.[42,43]

Sparse hierarchical classification (sparse subspace clustering-sparse representation-based classification)

A hierarchical classification method based on SR is presented in which a subset of training samples that belong to the same cluster as the test sample is used to construct the final dictionary.[7] This method, which employs a concise dictionary, achieves higher classification accuracy than the conventional SRC approach.

During the clustering stage, the Sparse Subspace Clustering (SSC) method is applied to partition the training samples into distinct clusters based on their SRs. The classification stage is then performed using SRC. In the following section, after a brief review of the SSC method, this classification approach is described in detail.

Sparse subspace clustering

Suppose $\{y_j \in \mathbb{R}^D\}_{j=1}^{N}$ the data are drawn from *n* independent linear subspaces $\{S_i\}_{i=1}^{n}$ with $\{d_i << D\}_{i=1}^{n}$. Let $Y_i \in \mathbb{R}^{D \times N_i}$ denote the N_k data samples belonging to the *i*-th subspace, and let $N = \sum_{i=1} N_i$ be the total number of data points. The matrix Y is then constructed as follows:

$$Y = [y_1, y_2, ..., y_N] = [Y_1, Y_2, ..., Y_n] \in \mathbb{R}^{D \times N} . \tag{20}$$

If y_i is a sample from the s_i subspace, it can be expressed as a linear combination of d_i samples within the same

subspace. In other words, y_i has a d_i-SR, which can be obtained by solving Eq. (3), where for $j \neq i$, $s_i \neq 0$ and $s_j = 0$.[44]

For data clustering based on the obtained SRs, spectral clustering is employed. In this approach, local information about each sample is used to define similarity measures between pairs of samples. A clustering procedure is then applied to the resulting similarity matrix, partitioning the data into groups such that samples within each group exhibit high similarity to one another and dissimilarity to those in other groups. Let $Y_i \in \mathbb{R}^{D \times N-1}$ denote the matrix obtained by removing the *i*-th column from Y. Then, y_i admits an SR over Y_i, which can be computed using the following equation:

$$\min \|s_i\|_1 \quad s.t \quad y_i = Y_i s_i \ . \tag{21}$$

The desired solution $s_i \in \mathbb{R}^{N-1}$ is a vector whose nonzero elements correspond to the columns of Y_i that lie in the same subspace as y_i. After solving the equation for i = 1,…,*N*, the coefficient matrix $S = [\hat{s}_1, \hat{s}_2, ..., \hat{s}_N]$ is constructed, where $\hat{s}_i$ is obtained by inserting a zero in the *i*-th row of s_i. Using this matrix, a graph $G = (V, E)$ is constructed where the vertices correspond to the data samples. An edge $(v_i, v_j) \in E$ is formed if y_j appears in the SR of y_i. If y_i can be expressed as a linear combination of samples from a subspace that includes y_j, then y_j can also be expressed as a linear combination of samples within the same subspace, including y_i. Therefore, the adjacency matrix of the graph is defined as $\tilde{S} = |S| + |S|^T$.

After constructing the graph, it is expected that the vertices corresponding to the same subspace form a connected component, while vertices from different subspaces share no common edges. As a result, the adjacency matrix $\tilde{S}$ exhibits a block-diagonal structure, as follows:

$$\tilde{S} = \begin{bmatrix} \tilde{S}_1 & 0 & \cdots & 0 \\ 0 & \tilde{S}_2 & \cdots & 0 \\ & & \vdots & \\ 0 & 0 & \cdots & \tilde{S}_n \end{bmatrix} \tag{22}$$

The Laplacian matrix of the similarity graph is then constructed as $L = D - \tilde{S}$, where $D \in \mathbb{R}^{N \times N}$ is a diagonal degree matrix computed as $D_{ii} = \sum_j \tilde{s}_{ij}$. Finally, data clustering is performed by applying the k-means algorithm to the *n* eigenvectors corresponding to the *n* smallest eigenvalues of the Laplacian matrix.[44]

Sparse subspace clustering-sparse representation-based classification

In this method, the classification process is performed hierarchically by combining SR-based clustering and classification techniques. The core idea is to first cluster the data, followed by classification within each cluster. In other words, after computing the SR of the test sample with respect to the training samples, the reconstruction is performed not over the entire training set, but only using the most similar training samples, as identified during the clustering stage.[7]

In the SSC-SRC method, training samples are clustered during the training phase using the SSC algorithm. To accomplish this, the SR of each training sample with respect to the remaining samples is obtained using Eq. (21). After constructing the graph's adjacency matrix and computing the Laplacian matrix, as described in the previous section, clusters are identified.

During the test phase, upon receiving a test sample, its SR over the training samples is first computed. In the reconstruction stage, for each step, only the sparse coefficients corresponding to a single cluster are retained, while the remaining coefficients are set to zero. The cluster associated with the minimum reconstruction error is then selected as the test sample's cluster. In the subsequent step, to determine the test label, the SR of the test sample is computed solely with respect to the training samples within the identified cluster. As a result, the final classification decision is made using a reduced subset of the training data. Figure 2 illustrates this classification procedure.

Experiments

In this section, the dataset used in the experiments is introduced, and the simulation results are presented and analyzed. An implementation of the methods proposed in this work will be made available on GitHub at https://github.com/MlhMri/SSC_SRC.

Data

The dataset used in the experiments to evaluate the performance of the reviewed classifiers is the 14-Tumors dataset.[9] This dataset consists of 198 samples, each containing 16,063 gene expression values (features) across 14 tumor types. Among these, 144 samples are used for training and 54 for testing. The value associated with each gene in a sample represents its expression level, i.e., gene activity. The general characteristics of this dataset are summarized in Table 1. It is evident that the number of samples in each class is significantly smaller than the feature dimensionality (16,063).

Simulation Results

Classifying the 14-Tumors dataset is a challenging task, and many conventional classification methods yield suboptimal results. Classification performance has been reported using various extensions of the SVM classifier,[16] including the one-versus-rest, OVO, directed acyclic graph, all-at-once method by Weston and Watkins (WW), and all-at-once method by Crammer and Singer (CS) methods. In these experiments, both polynomial and Radial Basis Function kernels were employed, with kernel parameters tuned via cross-validation. The lowest reported classification

error was approximately 24%. In the following section, the results of applying various SR-based classifiers are presented.

It is standard practice in microarray analysis to subject raw data to a preprocessing pipeline involving operations such as background adjustment, normalization across arrays, and missing value imputation. However, the preprocessed dataset utilized in this work[9] was already preprocessed, exhibiting no missing values and minimal technical variance. Therefore, the data was fed directly into a dimensionality reduction algorithm without further modification. The high dimensionality of the data not only increases memory usage and the computational complexity of algorithms but also reduces their efficiency due to the presence of noise and the limited number of samples relative to the dimensionality, commonly referred to as the curse of dimensionality. To determine the optimal reduced dimensionality using Principal Component Analysis (PCA), the 10-fold cross-validation technique was employed. In this process, the training samples were divided into ten folds; in each iteration, one fold was used as test data while the remaining nine were used for training. The classification procedure was repeated for various reduced dimensions. The results of these experiments, conducted using the SRC method in conjunction with the OMP algorithm, are presented in Figure 3.

The median, as well as the first and third quartiles of the obtained classification errors, are illustrated in Figure 3. The horizontal axis represents the amount of preserved energy in PCA. As shown in Figure 3, the lowest median error and variance are achieved when 99.99% of the energy is retained. Therefore, the dimension corresponding to this level of energy preservation, which is equal to 109, is considered optimal for applying the PCA algorithm.[45] The results of the SRC method using BP, OMP, and SL0 sparse coding algorithms are presented in Table 2.

As the results indicate, the highest classification accuracy is achieved using the OMP algorithm, whereas the SL0 algorithm demonstrates greater computational efficiency. In contrast, the BP algorithm yields the lowest performance among the three. While execution times vary based on system specifications and implementation approaches, the reported values serve as a useful basis for comparison.

As discussed before, the CRC method achieves higher computational efficiency than other SRC methods due to its use of ℓ_2-norm minimization. Applying this method results in a classification accuracy of 66.67% within a runtime of 4.45×10^{-4} seconds. The KNN-SRC method achieves an accuracy of 55.56% in just 0.0099 seconds for $k = 26$. The optimal value of k was determined using the 10f-CV method.

The classification results of the MSRC method using various sparse coding algorithms are summarized in Table 3. In this application, a metasample is defined as a linear combination of gene expression profiles from multiple samples, capturing their intrinsic characteristics and summarizing the overall gene expression pattern.[35] To evaluate the performance of this method, PCA is first applied for dimensionality reduction. Subsequently, metasamples are generated using SVD for each class independently, and the final dictionary is constructed by concatenating the resulting metasamples.

Due to the imbalanced number of training samples across different classes, the classifier tends to be biased toward classes with more training samples if the number of extracted metasamples is set equal to the number of training samples in each class.[35] To mitigate this issue, an equal number of metasamples is assigned to each class. Based on experimental results, the optimal number of metasamples was found to be 5, as determined using the 10f-CV approach.

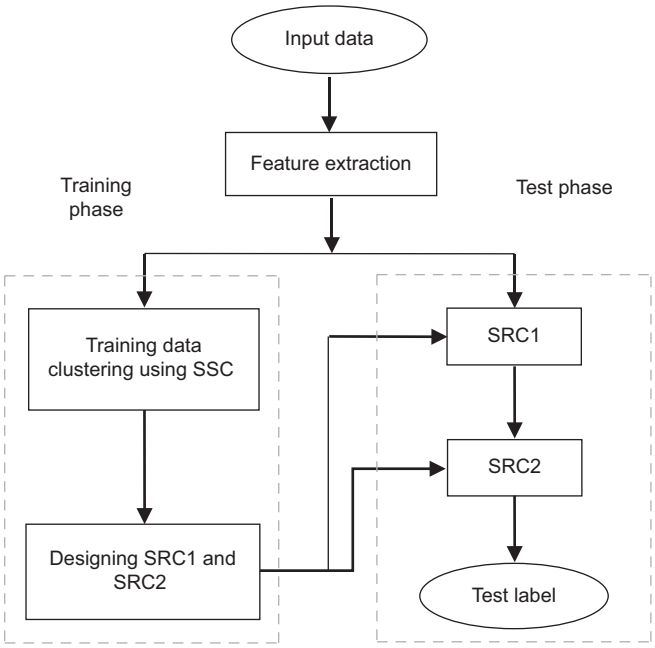

Figure 2: Sparse subspace clustering-sparse representation-based classification block diagram.[7] SRC – Sparse representation-based classification; SSC – Sparse subspace clustering

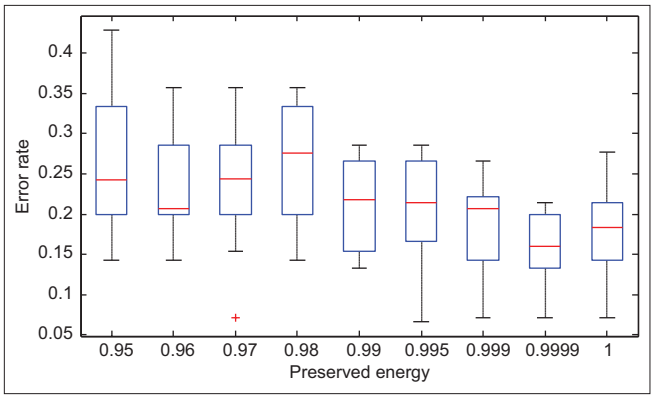

Figure 3: Classification error boxplot in 10f-CV

As shown in Table 3, all three algorithms yield identical classification accuracy. The primary difference lies in their computational efficiency, with the SL0 algorithm demonstrating the shortest runtime. Table 4 presents the classification results obtained using the LASRC method.

As shown in Table 4, the SL0 and OMP algorithms achieve higher classification accuracy compared to the BP algorithm. In addition, both SL0 and OMP exhibit significantly faster computation times. SL0, in particular, is notably more efficient, with the computation time of OMP being approximately 2.5 times longer than that of SL0. To determine the optimal number of atoms to be selected in the ℓ_2-norm minimization step, 10f-CV was employed. The best performance was obtained by selecting 100 atoms from the original dictionary.

Due to the within-class variations present in tumor samples, the SSC-SRC method is expected to yield strong classification performance. This approach first identifies the training samples most similar to the test sample and then performs classification using the samples within the corresponding cluster. The results of this method, using BP, OMP, and SL0 algorithms, are summarized in Table 5.

It can be observed that all three sparse coding algorithms achieve strong performance, with SL0 being faster than OMP and significantly faster than BP, as expected. In the SSC-SRC method, the optimal number of clusters was determined to be 5 using 10f-CV. In this approach, only a subset of the training samples is utilized in the final classification process. As a result, the likelihood that irrelevant samples influence the SR of a test sample is reduced, thereby improving the classification accuracy compared to the SRC method.

Table 6 summarizes the results of various SR-based classifiers. For each method, the highest achieved classification accuracy is reported.

The MSRC, LASRC, and SSC-SRC methods outperform the other approaches. Although their overall classification accuracies are comparable, their performance varies across different classes.

Figure 4 illustrates the classification results of the studied algorithms for each class individually. It can be observed that among the fourteen tumor classes in the dataset, four–lymphoma, leukemia, mesothelioma, and central nervous system (CNS)–are perfectly classified (100% accuracy) by all three methods. Notably, three of these four tumor types (lymphoma, leukemia, and CNS) have the largest number of training samples among all classes. In contrast, Melanoma is not classified with an accuracy above 50% by any of the methods. Additionally, Breast and Lung tumors are poorly classified, with accuracies of 25% and 50%, respectively, by two of the methods. These findings indicate that higher classification accuracies are generally achieved for classes with more training samples, further confirming the significant impact of training sample size on classification performance.

To further examine the impact of training sample size on classification performance, additional experiments were conducted using the Leave-One-Out Cross-Validation (LOOCV) method. In this procedure, all training and test samples are combined into a single pool. For each iteration, one sample is selected as the test sample, while the remaining samples are used for training. This process is repeated until every sample has been used once as the

Table 1: Data characteristics

Tumor types	Number of training samples	Number of test samples
Breast	8	4
Prostate	8	6
Lung	8	4
Colorectal	8	4
Lymphoma	16	6
Bladder	8	3
Melanoma	8	2
Uterus	8	2
Leukemia	24	6
Renal	8	3
Pancreas	8	3
Ovarian	8	4
Mesothelioma	8	3
CNS	16	4

CNS – Central nervous system

Table 2: Sparse representation-based classification results

	BP	OMP	SL0
Accuracy (%)	72.22	77.78	74.08
Time (s)	0.3767	0.0437	0.0148

BP – Basis pursuit; OMP – Orthogonal matching pursuit; SL0 – Smoothed ℓ_0-norm

Table 3: Metasample sparse representation-based classification results

	BP	OMP	SL0
Accuracy (%)	79.63	79.63	79.63
Time (s)	0.1173	0.0445	0.0363

BP – Basis pursuit; OMP – Orthogonal matching pursuit; SL0 – Smoothed ℓ_0-norm

Table 4: Linearly approximated sparse representation-based classification results

	BP	OMP	SL0
Accuracy (%)	77.78	79.63	79.63
Time (s)	0.3655	0.0342	0.0141

BP – Basis pursuit; OMP – Orthogonal matching pursuit; SL0 – Smoothed ℓ_0-norm

test sample. The LOOCV results, reported as accuracy ± standard error for the MSRC, LASRC, and SSC-SRC methods, are presented in Table 7.

These results indicate that improved classification performance is achieved across all methods. Notably, the SSC-SRC method exhibits further improvement with the increased number of training samples. Table 8 presents the classification performance of the SSC-SRC method using precision, recall, specificity, and F1-score metrics for each of the 14 classes individually and the average of each metric across all classes.

Figure 5 shows the confusion matrix for the SSC-SRC method, where each cell (i, j) indicates the number of samples whose true class is i but were predicted as class j. Diagonal cells represent correctly classified samples (true positives for that class). Values in off-diagonal cells are misclassifications (false negatives for the row class, false positives for the column class). The row summaries (right side) show the normalized row values (percentage of that true class predicted into each predicted class), and the column summaries (top) represent the normalized column values (percentage of that predicted class coming from each true class).

Finally, the results of several conventional classifiers applied to the 14-Tumors dataset, under the same settings as the SR-based methods, are summarized in Table 9. For each classifier, parameter optimization was performed using 10-fold cross-validation.

The results in Table 9 indicate that, as expected given the limited and imbalanced nature of the data, SR-based classifiers achieve superior performance compared to other methods, many of which are additionally affected by substantial computational complexity.

Discussion

In this paper, we aim to investigate the effectiveness of SR-based techniques for high-dimensional, small-sample, multiclass classification problems, using microarray data classification as a proof of concept. According to the literature, the 14-Tumors dataset is considered one of the most challenging benchmarks, as most classifiers, including SVMs, tend to yield relatively low performance. Similar to SVMs, SR-based approaches also face difficulties in classifying the 14-Tumors dataset. This challenge primarily arises from the limited total number of samples and the even smaller number of samples per class, which restricts the ability of state-of-the-art classifiers to effectively address microarray data classification.

Nevertheless, under the proposed setting with minimal preprocessing and feature extraction using PCA, SR-based approaches achieve superior performance compared to other conventional classifiers and, importantly, provide more robust classification than SVMs in this difficult scenario. Our analysis shows that microarray data can be effectively classified using only a subset of principal components, through a computationally efficient procedure.

In the CRC method, the contribution of all classes to dictionary construction is considered more effective than enforcing sparsity on the coefficients. Therefore, when the number of training samples included in the dictionary is significantly larger than the data dimensionality, ℓ_2-norm minimization tends to yield better performance than ℓ_1-norm

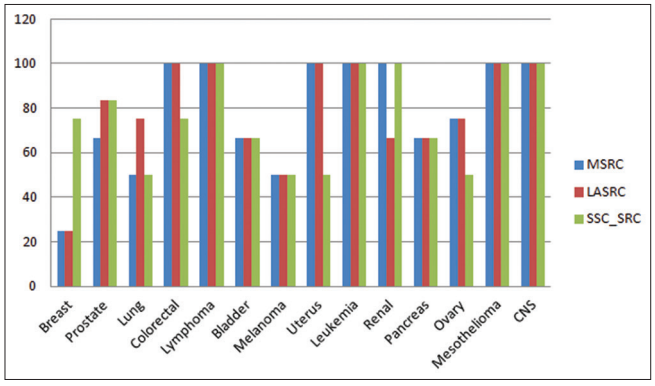

Figure 4: Classification accuracy for individual classes using metasample-based sparse representation-based classification (SRC), linearly approximated SRC, and sparse subspace clustering-SRC methods. SRC – Sparse representation-based classification; SSC – Sparse subspace clustering; MSRC – Metasample-based SRC; LASRC – Linearly approximated SRC; CNS – Central nervous system

Table 5: Sparse subspace clustering -sparse representation-based classification results

	BP	OMP	SL0
Accuracy (%)	79.63	79.63	79.63
Time (s)	1.4216	0.0618	0.0232

BP – Basis pursuit; OMP – Orthogonal matching pursuit; SL0 – Smoothed ℓ_0-norm

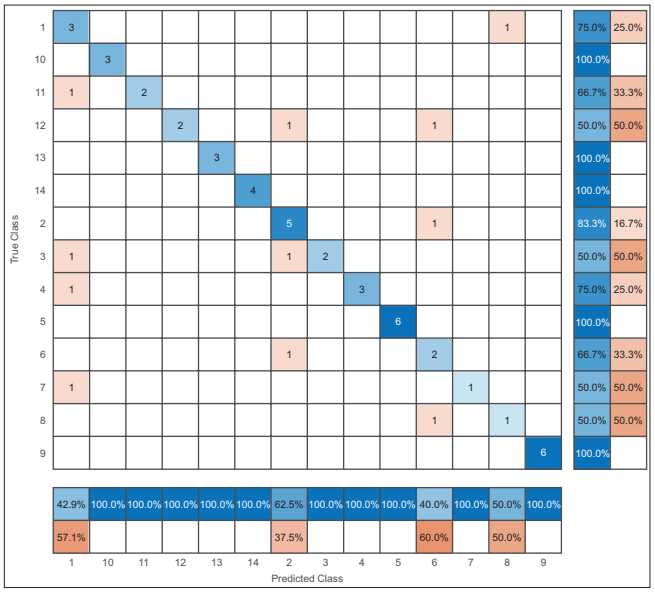

Figure 5: Confusion matrix obtained for the sparse subspace clustering-sparse representation-based classification method

Table 6: Sparse representation-based classifiers results

	Accuracy (%)	Time (s)
SRC	77.78	0.0437
CRC	66.67	0.0004
KNN-SRC	55.56	0.0099
MSRC	79.63	0.0363
LASRC	79.63	0.0141
SSC-SRC	79.63	0.0232

SRC – Sparse representation-based classification; LASRC – Linearly approximated SRC; MSRC – Metasample-based SRC; SSC – Sparse subspace clustering; KNN – K nearest neighbor; CRC – Collaborative representation-based classification

Table 7: Leave-one-out cross-validation results

Method	Accuracy±SE (%)
MSRC	82.83±2.68
LASRC	82.83±2.68
SSC-SRC	83.34±2.65

SRC – Sparse representation-based classification; SE – Standard error; LASRC – Linearly approximated SRC; MSRC – Metasample-based SRC; SSC – Sparse subspace clustering

Table 8: Sparse subspace clustering- sparse representation-based classification performance using different metrics

Tumor types	Precision	Recall	Specificity	F1-score
Breast	0.43	0.75	0.92	0.55
Prostate	0.63	0.83	0.94	0.71
Lung	1	0.5	1	0.67
Colorectal	1	0.75	1	0.86
Lymphoma	1	1	1	1
Bladder	0.4	0.67	0.94	0.5
Melanoma	1	0.5	1	0.67
Uterus	0.5	0.5	0.98	0.5
Leukemia	1	1	1	1
Renal	1	1	1	1
Pancreas	1	0.67	1	0.8
Ovarian	1	0.5	1	0.67
Mesothelioma	1	1	1	1
CNS	1	1	1	1
Mean	0.85	0.76	0.98	0.78

CNS – Central nervous system

Table 9: Different classifiers results

Method	Accuracy (%)
SVM (OVR)	70.37
SVM (OVO)	68.52
RF	64.81
KNN	55.56
Naïve bayes	50
SRC	77.78
SSC-SRC	79.63

OVR – One-versus-rest; KNN – K nearest neighbor; OVO – One-versus-one; SVM – Support vector machine; RF – Random forest; SSC – Sparse subspace clustering; SRC – Sparse representation-based classification

minimization. However, in the present case, where the number of samples is very limited, the weaker performance of CRC is expected. Furthermore, because the solution obtained via ℓ_2-norm minimization is not sufficiently sparse and is distributed across nearly all training samples, least squares-based methods become highly sensitive to class imbalance. Thus, the uneven distribution of training samples across different classes may be another contributing factor to the suboptimal performance of CRC.

The satisfactory performance of the MSRC method suggests that the extracted metasamples from each class effectively represent the corresponding class data. The results obtained from the LASRC and SSC-SRC methods further confirm that enhancements to SRC can lead to improved performance. Although these methods also select a subset of atoms from the original dictionary to compute the SR of a test sample, the selection criteria they employ are more effective than those used in the KNN-SRC method. In the LASRC method, ℓ_2-norm minimization is employed, and the coefficients with the largest absolute values are selected as the final candidates for constructing a concise dictionary. In SSC-SRC, the final dictionary is formed by gathering the training samples that belong to the same cluster as the test sample. These approaches indicate that the atom selection criteria in both methods are guided by the concept of sparsity and are closely aligned with the principles of SR.

The per-class results of the MSRC, LASRC, and SSC-SRC methods indicate that their performance varies across different classes. Therefore, it is reasonable to expect that an appropriate combination of these methods could yield improved results. Moreover, the performance of these methods under LOOCV is promising, suggesting that increased training data may further enhance classification accuracy. This trend is particularly evident in the SSC-SRC method, which demonstrates notable improvement as the number of training samples increases.

Conclusions

In this study, the effectiveness of several SRC methods was evaluated on microarray data. The experimental results demonstrate that these methods outperform conventional classifiers. Notably, even the lowest-performing method among them–KNN-SRC–achieves higher accuracy than some previously adopted approaches. Among the evaluated methods, MSRC, LASRC, and SSC-SRC, in conjunction with the SL0 algorithm, yielded the best results. These findings suggest that an appropriate combination of these methods may further enhance classification performance.

Furthermore, the results obtained from SR-based methods demonstrate that high classification performance can be achieved using a significantly reduced number of features. In these methods, improved results were observed when the data were projected into a lower-dimensional space. In

our experiments, PCA was employed for feature extraction. In addition, gene selection represents a promising strategy for addressing the curse of dimensionality and plays a critical role in enhancing the efficiency of microarray data classification.

Given that only a small subset of genes is typically relevant to the classification task, some of the proposed methods aim to select the most informative genes while eliminating irrelevant, redundant, or noisy data to enhance classification performance.[3,46-48] Applying feature selection techniques prior to the SR-based classifiers studied in this work presents a promising direction for future research. In addition, evaluating the performance of these methods on other microarray datasets is recommended to further validate and generalize the findings.

Availability of data and materials

This research study was conducted retrospectively using publicly available data. Our local ethics committee has determined that ethical approval is not necessary for analyzing this openly accessible data.

Financial support and sponsorship

Nil.

Conflicts of interest

There are no conflicts of interest.

References

1. Dousti Mousavi N, Aldirawi H, Yang J. Categorical data analysis for high-dimensional sparse gene expression data. BioTech 2023;12:52.
2. Yang J, Ma J, Win KT, Gao J, Yang Z. Low-rank and sparse representation based learning for cancer survivability prediction. Inf Sci 2022;582:573-92.
3. Han S, Wang N, Guo Y, Tang F, Xu L, Ju Y, *et al*. Application of sparse representation in bioinformatics. Front Genet 2021;12:810875.
4. Donoho D. Compressed sensing. IEEE Trans Inf Theory 2006;52:1289-306.
5. Mody SK, Rangarajan G. Sparse representations of high dimensional neural data. Sci Rep 2022;12:7295.
6. Guo X, Tiwari P, Zhang Y, Han S, Wang Y, Ding Y. Random Fourier features-based sparse representation classifier for identifying DNA-binding proteins. Comput Biol Med 2022;151:106268.
7. Miri M, Sadeghi MT, Abootalebi V. Two stages sparse representation-based classifier and its application for cancer classification. Iran J Biomed Eng 2014;8:45-56.
8. Wright J, Yang AY, Ganesh A, Sastry SS, Ma Y. Robust face recognition via sparse representation. IEEE Trans Pattern Anal Mach Intell 2009;31:210-27.
9. Ramaswamy S, Tamayo S, Rafkin P. Multiclass cancer diagnosis using tumor gene expression signatures. Natl Acad Sci U S Am 2001;98:5149-54.
10. Bhola A, Tiwari AK. Machine learning based approaches for cancer classification using gene expression data. Mach Learn Appl Int J 2015;2:1-12.
11. Aydadenta H. On the classification techniques in data mining for microarray data classification. J Phys Conf Ser 2018;971:012004.
12. Ram M, Najafi A, Shakeri MT. Classification and biomarker genes selection for cancer gene expression data using random forest. Iran J Pathol 2017;12:339-47.
13. Linder R, Dew D, Sudhoff H, Theegarten D, Remberger K, Poppel SJ, *et al*. The subsequent artificial neural network (SANN) approach might bring more classificatory power to ANN based DNA microarray analyses. Bioinformatics 2004;20:3544-52.
14. Zhang R, Huang GB, Sundararajan N, Saratchandran P. Multi-category classification using an extreme learning machine for microarray gene expression cancer diagnosis. IEEE/ACM Trans Comput Biol Bioinformatics 2007;4:485-95.
15. Shabgahi Z, Abadeh MS. A Fuzzy Classification System Based on Memetic Algorithm for Cancer Disease Diagnosis. 18th Iranian Conference of Biomedical Engineering (ICBME); 2011.
16. Hang X, Wu FX. Sparse representation for classification of tumors using gene expression data. J Biomed Biotechnol 2009;2009:403689.
17. Yuan B, Yang D, Rothberg BE, Chang H, Xu T. Unsupervised and supervised learning with neural network for human transcriptome analysis and cancer diagnosis. Sci Rep 2020;10:19106.
18. Laprovitera N, Riefolo M, Ambrosini E, Klec C, Pichler M, Ferracin M. Cancer of unknown primary: Challenges and progress in clinical management. Cancers (Basel) 2021;13:451.
19. Sharma A, Rani R. A systematic review of applications of machine learning in cancer prediction and diagnosis. Arch Comput Methods Eng 2021;28:4875-96.
20. Nguyen L, Van Hoeck A, Cuppen E. Machine learning-based tissue of origin classification for cancer of unknown primary diagnostics using genome-wide mutation features. Nat Commun 2022;13:4013.
21. Li W, Liao B, Zhu W, Chen M, Peng L, Wei X, *et al.* Maxdenominator reweighted sparse representation for tumor classification. Sci Rep 2017;7:46030.
22. Huang HH, Liu XY, Liang Y. Feature selection and cancer classification via sparse logistic regression with the hybrid L1/2 +2 regularization. PLoS One 2016;11:e0149675.
23. Gan B, Zheng CH, Zhang J, Wang HQ. Sparse representation for tumor classification based on feature extraction using latent low-rank representation. Biomed Res Int 2014;2014:420856.
24. Donoho D. For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution. Commun Pure Appl Math 2006;59:797-829.
25. Chen S, Donoho D, Saunders MA. Atomic decomposition by basis pursuit. SIAM J Sci Comput 1998;20:33-61.
26. Mallat S, Zhang Z. Matching pursuits with time frequency dictionaries. IEEE Trans Sign Proc 1993;41:3397-415.
27. Pati YC, Rezaiifar R, Krishnaprasad PS. Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition. 27 Asilomar Conf. Signals, Syst. Comput; 1993. p. 41-4.
28. Mohimani H, Babaie-Zadeh M, Jutten C. A fast approach for overcomplete sparse decomposition based on smoothed l0 norm. IEEE Trans Sign Process 2009;57:289-301.
29. Malioutov D, Cetin M, Willsky A. Homotopy Continuation for Sparse Signal Representation. In Proc. IEEE Conf. ICASSP. Vol. 5; 2005. p. 733-6.
30. Figueiredo M, Nowak R, Wright S. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. IEEE Trans Sel Top Sign Process 2007;1:586-97.
31. Beck A, Teboulle M. A fast iterative shrinkage-thresholding

algorithm for linear inverse problems. SIAM J Imaging Sci 2009;2:183-202.
32. Needell D, Tropp JA. Cosamp: Iterative signal recovery from in-complete and inaccurate samples. Appl Comput Harmon Anal 2009;26:301-21.
33. Blumensath T, Davies ME. Iterative hard thresholding for compressed sensing. Appl Comput Harmon Anal 2009;27:265-74.
34. Zhang D, Yang M, Feng XC. Sparse Representation or Collaborative Representation: Which Helps Face Recognition? In Proc. IEEE Conf. Computer Vision (ICCV); 2011. p. 471-8.
35. Zheng CH, Zhang L, Ng TY, Shiu SK, Huang DS. Metasample-based sparse representation for tumor classification. IEEE/ACM Trans Comput Biol Bioinform 2011;8:1273-82.
36. Zheng CH, Huang DS, Zhang L, Kong XZ. Tumor clustering using non-negative matrix factorization with gene selection. IEEE Trans Inf Technol Biomed 2009;13:599-607.
37. Alter O, Brown PO, Botstein D. Singular value decomposition for genome-wide expression data processing and modeling. Proc Natl Acad Sci U S A 2000;97:10101-6.
38. Liao B, Jiang Y, Yuan G, Zhu W, Cai L, Cao Z. Learning a weighted meta-sample based parameter free sparse representation classification for microarray data. PLoS One 2014;9:e104314.
39. Jafari MG, Plumbley MD. Fast dictionary learning for sparse representations of speech signal. IEEE J Sel Top Sign Process 2011;5:1025-31.
40. Miri M, Sadeghi MT, Abootalebi V. Improving Sparse Representation-Based Classifier Performance with GAD Dictionary Learning. In: Proc. 8th Iranian Conference on Machine Vision and Image Processing (MVIP), Zanjan, Iran; 2013.
41. Zhang N, Yang J. K Nearest Neighbor Based Local Sparse Representation Classifier. In: Proc. Chinese Conference on Pattern Recognition (CCPR); 2010.
42. Ortiz EG, Becker BC. Face recognition for web-scale datasets. J Comput Vision Image Underst 2014;118:153-70.
43. Miri M. Face Verification in the Wild Using Similarity in Representations. In: Proc. IEEE Artificial Intelligence and Signal Processing Conference (AISP); 2017. p. 140-4.
44. Elhamifar E, Vidal R. Sparse subspace clustering: Algorithm, theory, and applications. IEEE Trans Pattern Anal Mach Intell 2013;35:2765-81.
45. Miri M, Sadeghi MT, Abootalebi V. Multiclass Microarray Data Classification Using SRC Approximations. In Proc. IEEE 23rd Iranian Conference on Electrical Engineering (ICEE); 2015. p. 115-9.
46. Liu S, Xu C, Zhang Y, Liu J, Yu B, Liu X, *et al.* Feature selection of gene expression data for cancer classification using double RBF-kernels. BMC Bioinformatics 2018;19:396.
47. Cilia ND, Stefano CD, Fontanella F, Raimondo S, Freca AS. An experimental comparison of feature-selection and classification methods for microarray datasets. Information 2019;10:109.
48. Alanni R, Jingyu H, Azzawi H, Xiang Y. A novel gene selection algorithm for cancer classification using microarray datasets. BMC Med Genomics 2019;12:10.