Kernel principal component analysis combining rotation forest method for linearly inseparable data

Huijuan Lu a, Yaqiong Meng a, Ke Yan a,⇑, Zhigang Gao b

a College of Information Engineering, China Jiliang University, Hangzhou 310018, China
b College of Computer Science, Hangzhou Dianzi University, Hangzhou 310018, China

Received 31 October 2017; received in revised form 7 January 2018; accepted 31 January 2018

Abstract

Rotation forest (RoF) is an ensemble classifier combining linear analysis theories and decision tree algorithms. In recent existing works, RoF was widely applied to various fields with outstanding performance compared to traditional machine learning techniques, given that a reasonable number of base classifiers is provided. However, the conventional RoF algorithm suffers from classifying linearly inseparable datasets. In this study, a hybrid algorithm integrating kernel principal component analysis (KPCA) and the conventional RoF algorithm is proposed to overcome the classification difficulty for linearly inseparable datasets. The radial basis function (RBF) is selected as the kernel for the KPCA method to establish the nonlinear mapping for linearly inseparable data. Moreover, we evaluate various kernel parameters for better performance. Experimental results show that our algorithm improves the performance of RoF with linearly inseparable datasets, and therefore provides higher classification accuracy rates compared with other ensemble machine learning methods.

© 2018 Elsevier B.V. All rights reserved.

Keywords: Rotation forest; Kernel principal component analysis; Linearly inseparable data; Decision tree

1. Introduction

Ensemble machine learning technology, which takes the advantages of multiple simple base classifiers, is a common technique to enhance the original classification ability of each individual base classifier in the field of machine learning (Dietterich, 2002; Huang, 1996). While traditional classifiers, such as decision trees, always tend to come out with local optimal solutions for modern complex datasets, such as the gene expression datasets, the ensemble machine learning technology divides the training dataset into smaller subsets, learns each small data subset independently using various base classifiers and outputs an overall optimal trained model (Liu, Lu, Yan, Xia, & An, 2016; Lu et al., 2016). In most of the ensemble learning algorithms, there are usually a number of parameters that can be tuned, such as the number of training subsets and the weight for each trained model, to ensure the superior of the classification accuracy, time complexity and other important performance measurements over individual base classifier.

There are many state-of-art ensemble machine learning algorithms in the literature, including Bagging, Adaboost, random forest and etc. Before random forest was introduced, Adaboost was known as the most popular ensemble learner, where Bagging sometimes provides better classification performance for datasets with heavy noises. Random forest allows random selection of features at every

⇑ Corresponding author.
E-mail address: yanke@cjlu.edu.cn (K. Yan).

https://doi.org/10.1016/j.cogsys.2018.01.006
1389-0417/ © 2018 Elsevier B.V. All rights reserved.

Rotation forest (RoF) is known as a recently developed ensemble machine learning technique, extending the concept of random forest based on decision trees (Rodriguez, Kuncheva, & Alonso, 2006). Each decision tree is trained separately with a subset of the original dataset and embedded into a rotational feature space. Since the final classification is approached by cutting the feature space using hyper-planes, a small twist of the feature axes can make big differences for the classification result. The main steps of the RoF algorithm consist of random feature segmentation, attribute transformation and difference calculation for the training dataset. Existing works prove that, in most of the situations, RoF is a more favorable choice comparing with Bagging, AdaBoost or random forest (Lu, Yang, Yan, Xue, & Gao, 2016). However, the direct application of the conventional RoF on linearly inseparable data, such as the gene expression datasets (Nguyen & Rocke, 2002), usually implies low classification accuracy.

Hybrid RoF approaches solve the linear inseparable problem by combining RoF with machine learning methods consisting of non-linear kernels, such as support vector machines (SVMs) Yan, Shen, Mulumba, & Afsahi, 2014, back-propagation neural networks (BPNs) Wang, Gu, Ma, & Yan, 2017, feedforward neural networks (FNNs) Huang, 1999; Huang & Ma, 1999 and multilayer perceptron network (MLPN) Huang, 2004. Existing works show that, with non-linear kernels, such as the radial basis function (RBF) or sigmoid function, the original non-linear dataset can be transformed to a reduced dataset with more separable feature sets (Huang & Zhao, 2005; Zhao, Huang, Du, & Wang, 2004).

In this study, we employ the kernel principal component analysis (KPCA) and carefully tune the KPCA parameters to deal with the linearly inseparable datasets, e.g., the gene expression datasets. A hybrid framework is presented with the combination of KPCA and RoF. The proposed KPCA-RoF algorithm employs the radial basis function (RBF) kernel for the KPCA to establish the nonlinear mapping and transform the original data into linearly separable feature space (Li, Zheng, & Huang, 2008). In the experiment section, we focus on tuning the optimal parameters for the KPCA. The F-measure is utilized to show the improvement of using our method compared to other traditional ensemble machine learning techniques, such as Bagging and random forest (Huang & Mi, 2007). The main impacts of this work to the literature include:

1. An RoF structure is designed to classify the gene expression data that founds to be linearly inseparable. The RoF segments and transforms the original training set to form the classification model, which provides the possibility to transform the linearly inseparable datasets into linearly separable.

2. A KPCA method with RBF kernel is employed to provide a non-linear transformation of the attributes to the feature space. A parameter tuning process is shown to improve the classification performance.

3. A hybrid classification that combines RoF and KPCA is proposed, namely, KPCA-RoF. Better classification performance using the proposed KPCA-RoF method is found via the classification results over six different gene expression datasets.

2. Related work

Decision tree (DT) algorithm and its extensions, such as EG2, ID3, C4.5 and CART, are popular machine learning methods for fast classification of linearly separable datasets (Quinlan, 2014). Random forest (RF) is an ensemble DT classifier which splits the feature space using multiple DTs (Liaw & Wiener, 2002). The rotation forest (RoF) is another extension of RF which assigns different sub-datasets with different DTs. The RoF segments the feature space using hyper-planes, eliminates the co-relations in datasets and iteratively obtains the most distinguishable transformed parameter set for each DT. It was already an ensemble algorithm when it was first introduced by Rodriguez et al. (2006). In the following year, Kuncheva and Rodriguez (2007) proved that the RoF had better performance than traditional ensemble machine learning techniques, such as Adaboost, bagging and RF. They also found that the feature extracted by principal component analysis (PCA) is most suitable for enhancing the classification accuracy of RoF.

At the meanwhile, existing works show that the RoF is not compatible with arbitrary datasets, especially for linearly inseparable datasets (Lu et al., 2016; Ozcift, 2012). A series of hybrid RoF algorithms were proposed to solve the linearly inseparable problem. Zhang and Zhang (2008) started to introduce a hybrid algorithm combining RoF with Adaboost to improve the classification performance for linearly inseparable datasets. Mousavi, Eftekhari, and Haghighi (2015) proposed a hybrid RoF approach combining ensemble pruning and rotation forest to predict human miRNA target. For ensemble pruning, they utilizes genetic algorithm. In other words, a subset of classifiers from the heterogeneous ensemble is firstly selected by genetic algorithm. Next, the selected classifiers are trained based on the rotation forest method and then combined using weighted majority voting. Wong et al. (2015) combined the RoF and local phase quantization (LPQ), which evaluates the performance of the RoF ensemble classifier with the state-of-the-art support vector machine (SVM) classifier. Wong at el. also indicated that the proposed method might play a complementary role for future proteomics research. Ayerdi and Romay (2016) introduced a new spectral classifier, which was called anticipative hybrid extreme rotation forest (AHERF). The AHERF approaches started from a model selection phase, used a small subset of the
training data, and established a ranking-based selection probability distribution for the classifier architecture.

Hybrid methods are modern commonly used approaches to solve complicated problems in the field of machine learning, data mining and automated systems (Huang & Du, 2008; Yan, Ji, & Shen, 2017; Yan et al., 2017). Recent works also show that the hybrid machine learning techniques integrating KPCA method effectively extract significant information from non-linear datasets. Kuang, Xu, and Zhang (2014) proposed a novel hybrid method which involves a KPCA with radial basis function (RBF) kernel, a multi-layer support vector machine (SVM) and genetic algorithm (GA) for intrusion detection. Mengqi et al. (2015) introduced two hybrid approaches for online fault detection of non-linear data, which include a hybrid method combining KPCA and hypersphere support vector machine (HSSVM) and another hybrid method mixing Recursive KPCA, Adaptive Control Limit (ACL) and Online Sequential Extreme Learning Machine (OS-ELM) together. Luo, Li, Deng, Zhong, and Cai (2016) extended the traditional KPCA by integrating the statistic pattern analysis framework (SPA), and named the extended KPCA as multivariate statistical kernel principal component analysis (MSKPCA). Experimental results showed that the MSKPCA has better performance compared to PCA and KPCA for fault detection and diagnosis in the particular application field. Sun, Huang, and Cheung (2005), Sun, Huang, Cheung, Liu, and Huang (2005) extracted non-linear features from multispectral images using hybrid PCA method. The combination of fuzzy c-means clustering and KPCA is proved to be an effective method for extracting features with a large number of multispectral images.

In this study, we combine the KPCA with RoF algorithms to show the classification performance with linearly inseparable and high-dimensional gene expression datasets. The experimental results justify that the proposed KPCA-RoF algorithm outperforms most of the existing ensemble classifiers, including Bagging, RF and RoF.

3. The proposed algorithm

The proposed KPCA-RoF algorithm is a hybrid machine learning technique that deals with linearly inseparable data, such as the gene expression data. The traditional RoF algorithm segments the training dataset into several subsets, builds a decision tree (DT) classifier for each subset and calculates the classification results based on confidence coefficient of each DT (Lu et al., 2016). The proposed KPCA-RoF algorithm adds a preprocessing step in the DT construction process, which transforms each training subset into lower feature space using KPCA. At the testing phase, each testing sample is again transformed by KPCA and then inserted into the trained KPCA-RoF classifier that combines all trained DT classifiers.

The overall structure of KPCA-RoF is shown in Fig. 1 and can be described as follows:

(1) For a given $n$-dimensional dataset $S : \{(x_i, y_i)\}_{i=1}^L$, we remove partial class labels and call it $H$, divide $H$ into $K$ disjoint subsets.

(2) Suppose that $D_1, D_2, \ldots, D_K$ are the base classifiers. We randomly select $m$ features to form a feature subset $H_{ij}$ corresponding to $D_i$, where $m = n/k$, $1 \leq i \leq L, 1 \leq j \leq k$. From each $H_{ij}$, KPCA is used to transform the feature subset to a coefficient matrix $C_{ij}$.

(3) Repeat step (2) $K$ times.

(4) Combine the $K$ coefficient matrixes to form a large sparse matrix. The rotation matrix $R_i$ for the base classifier $D_i$ is defined as:

$$R_i = \begin{bmatrix} C_{i1} & 0 & \cdots & 0 \\ 0 & C_{i2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & C_{ik} \end{bmatrix}$$

(5) The classifier $D_i$ uses dataset $H_{ij} \cdot R_i$ as the training set. In the testing process, each testing sample $x$ is fed into the ensemble classifier after multiplying $R_i$. The confidence coefficient of determining its class is given as:

$$u_i(x) = \frac{1}{\sum_{i=1}^{L} p_i(x)}$$

3.1. Kernel principal component analysis (KPCA)

Feature selection and dimension reduction are important steps in classification problems (Lu et al., 2016; Yan et al., 2017). Suppose \( x_1, x_2, \ldots, x_m \) are training samples, each of which consists of \( N \) features. A kernel function maps input sample \( x \) into feature space \( F \) by mapping function \( \Phi \). The mapping function \( \Phi(x) \) from each sample \( x \) to a reduced sample is defined as:

\[
\Phi: R^N \rightarrow F, \quad x \mapsto \xi = \Phi(x)
\]

(3)

The feature space \( F \) is generated by \( \Phi(x_1), \Phi(x_2), \ldots, \Phi(x_m) \). The training samples after mapping with centralization is processed as follows:

\[
\sum_{i=1}^{m} \Phi(x_i) = 0
\]

(4)

The covariance matrix of feature space after mapping is:

\[
\bar{C} = \frac{1}{m} \sum_{i=1}^{m} \Phi(x_i) \Phi(x_i)^T.
\]

(5)

The feature equation according to conventional PCA is as follows:

\[
\lambda V = \bar{C}V
\]

(6)

where \( \lambda \) is the eigenvalue and \( V \) is the eigenvector both belonging to the space, which is generated by \( \Phi(x_i) \) with the following equation:

\[
\lambda \{ \Phi(x_i) \cdot V \} = \{ \Phi(x_i) \cdot \bar{C}V \}
\]

(7)

A parameter \( x \) is adopted to make \( V \) to be a linear expression with the equation below:

\[
V = \sum_{i=1}^{m} x_i \Phi(x_i)
\]

(8)

By merging Eqs. (7) and (8), the inner product of mapping data is defined as a \( m \)-order matrix \( K \). The parameters are calculated by the kernel function that we chose in the early phase, namely:

\[
K_{ij} = (\Phi(x_i) \cdot \Phi(x_j))
\]

(9)

Next, we obtain Eq. (10), which is equivalent to Eq. (6):

\[
\lambda x = K x
\]

(10)

where \( x = (x_1, x_2, \ldots, x_m)^T \), and matrix \( K \) is the transformation matrix for the KPCA algorithm.

Suppose that the eigenvalues of matrix \( K \) are given as: \( \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_m \), the corresponding eigenvectors are denoted as: \( x_1, x_2, \ldots, x_m \).

3.2. Kernel function and its parameters

In order to segment the linearly inseparable dataset, the dataset is required to be transformed into a feature space through a kernel function.

Several common kernel functions are shown as follows:

Linear kernel function:

\[
K(x_i, x_j) = x_i^T x_j
\]

(11)

Polynomial kernel function:

\[
K(x_i, x_j) = (x_i^T x_j + r)^p
\]

(12)

Radial basis function (RBF):

\[
K(x_i, x_j) = \exp(-\delta ||x_i - x_j||^2)
\]

(13)

In this study, RBF becomes the best choice among all kernel functions because of the low magnitude complexity, less parameters and strong representation. Moreover, the problem of over-fitting is improved by adjusting the size of the parameters of kernel function. Different kinds of kernel functions result in different classification results. The diversity of kernel function can be evaluated by measuring interclass distances. The kernel function is widely used, independent with any particular classification algorithm, and uninfluenced by generalization ability.

The optimization of parameters uses the distances between feature classes as a reference index. The angle and the distance in the feature space of training data after mapping are given as:

\[
\cos(\Theta_{ij}) = \frac{K(x_i, x_j)}{\sqrt{K(x_i, x_i) \cdot K(x_j, x_j)}}, \quad 0 \leq \Theta_{ij} \leq \pi/2
\]

(14)

The distance between two vectors is calculated and denoted as \( D_{ij} \):

\[
D_{ij} = \sqrt{K(x_i, x_i) + K(x_j, x_j) - 2K(x_i, x_j)}
\]

(15)

After replacing a specific kernel function in Eqs. (12) and (13), we have:

\[
\cos(\Theta_{ij}) = \frac{\exp(-\delta ||x_i - x_j||^2)}{\sqrt{\exp(-\delta ||x_i - x_j||^2) \cdot \exp(-\delta ||x_j - x_i||^2)}} = \exp(-\delta ||x_i - x_j||^2)
\]

(16)

All \( \Theta_{ij} \) must meet the following condition:

\[
\Theta_{ij} \in [0, \pi/2]
\]

(17)

The distance between two vectors can also be obtained in a similar way:

\[
D_{ij} = \sqrt{2 - 2 \exp(-\delta ||x_i - x_j||^2)}
\]

(18)

From Eqs. (16)–(18), there is only one parameter \( \delta \) that influences the class distance and angle, which eventually determines the class distribution in the feature space as well as the RoF classification results.

While the value of the parameter \( \delta \) approaches to 0, the cosine of the angle approaches to 1, which means the value of two vectors’ gap angle approaches to 0 after mapping. In
this situation, the distance vector tends to be zero, which means all the samples are mapped to one point, leading to unclassifiable samples. While the value of the parameter $\delta$ tends to infinity, the two vectors’ gap angle tends to $\pi/2$; the distance of the sample tends to a constant, which means the sample set is mapped into an $n$-dimensional feature space. The feature vectors become pairwise orthogonal. While the dimension of the feature space increases to $n$ ($n$ is the number of samples) monotonically with the increment of $\delta$, the vector angle and distance in feature space also increase monotonically, approaching to $\pi/2$ and $\sqrt{2}$ respectively.

Given a training set $X$ containing $L$ samples and $C$ categories:

$$X = \bigcup_{i=1}^{C} x_i, \quad L = \bigcup_{i=1}^{C} l_i$$

(19)

Suppose that in class $C_1$, there are $l_1$ samples. Similarly, in class $C_2$, there are $l_2$ samples. The average of the samples in each class after mapping in the feature space can be calculated as:

$$m_1 = \frac{1}{l_1} \sum_{i=1}^{l_1} \Phi(x_i^{(1)}), \quad m_2 = \frac{1}{l_2} \sum_{i=1}^{l_2} \Phi(x_i^{(2)}),$$

(20)

where $m_1$ is the average of samples in class $C_1$; and $m_2$ is the average of samples in class $C_2$. The average distance between the classes in the mapping space can be expressed as follow:

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Sample num</th>
<th>Gene num</th>
<th>Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leukemia</td>
<td>72</td>
<td>7129</td>
<td>ALL</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>AML</td>
</tr>
<tr>
<td>Colon</td>
<td>62</td>
<td>2000</td>
<td>Normal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Tumor</td>
</tr>
<tr>
<td>CNS</td>
<td>60</td>
<td>7129</td>
<td>Negative</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Positive</td>
</tr>
<tr>
<td>Lung</td>
<td>181</td>
<td>12,533</td>
<td>Negative</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Positive</td>
</tr>
<tr>
<td>Breast</td>
<td>97</td>
<td>24,481</td>
<td>Non-relapse</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Relapse</td>
</tr>
<tr>
<td>SRBCT</td>
<td>83</td>
<td>2309</td>
<td>EWS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>RMS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NB</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NHL</td>
</tr>
</tbody>
</table>

(19)

Table 1
Gene expression datasets.

Fig. 2. The relationship between parameter values and normalized inter-class distances.

Table 2
Statistics analysis.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Algorithm</th>
<th>$\hat{X}$</th>
<th>$S_2$</th>
<th>$F$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Breast</td>
<td>KPCA-RoF</td>
<td>0.8087</td>
<td>0.0059</td>
<td>4.51</td>
</tr>
<tr>
<td></td>
<td>RoF</td>
<td>0.7092</td>
<td>0.0266</td>
<td></td>
</tr>
<tr>
<td>CNS</td>
<td>KPCA-RoF</td>
<td>0.7975</td>
<td>0.0153</td>
<td>2.2</td>
</tr>
<tr>
<td></td>
<td>RoF</td>
<td>0.7523</td>
<td>0.0337</td>
<td></td>
</tr>
<tr>
<td>Leukemia</td>
<td>KPCA-RoF</td>
<td>0.7787</td>
<td>0.0134</td>
<td>3.65</td>
</tr>
<tr>
<td></td>
<td>RoF</td>
<td>0.7902</td>
<td>0.0489</td>
<td></td>
</tr>
<tr>
<td>Colon</td>
<td>KPCA-RoF</td>
<td>0.8617</td>
<td>0.0040</td>
<td>1.65</td>
</tr>
<tr>
<td></td>
<td>RF</td>
<td>0.7923</td>
<td>0.0066</td>
<td></td>
</tr>
<tr>
<td>SRBCT</td>
<td>KPCA-RoF</td>
<td>0.7852</td>
<td>0.0076</td>
<td>1.81</td>
</tr>
<tr>
<td></td>
<td>RF</td>
<td>0.7010</td>
<td>0.0042</td>
<td></td>
</tr>
<tr>
<td>Lung</td>
<td>KPCA-RoF</td>
<td>0.9136</td>
<td>0.0007</td>
<td>2.71</td>
</tr>
<tr>
<td></td>
<td>Bagging</td>
<td>0.6426</td>
<td>0.0019</td>
<td></td>
</tr>
</tbody>
</table>

The highest classification accuracy of each dataset is marked in bold.

Table 3
$F$-distribution.

<table>
<thead>
<tr>
<th>$n_2$</th>
<th>$n_1$</th>
<th>1</th>
<th>3</th>
<th>5</th>
<th>6</th>
<th>8</th>
<th>12</th>
<th>$\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.896</td>
<td>53.59</td>
<td>57.24</td>
<td>58.20</td>
<td>59.44</td>
<td>60.71</td>
<td>63.33</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>5.54</td>
<td>5.36</td>
<td>5.31</td>
<td>5.28</td>
<td>5.25</td>
<td>5.22</td>
<td>5.13</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>4.06</td>
<td>3.62</td>
<td>3.45</td>
<td>3.40</td>
<td>3.34</td>
<td>3.27</td>
<td>3.10</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>3.78</td>
<td>3.29</td>
<td>3.11</td>
<td>3.05</td>
<td>2.98</td>
<td>2.90</td>
<td>2.72</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>3.59</td>
<td>3.07</td>
<td>2.88</td>
<td>2.83</td>
<td>2.75</td>
<td>2.67</td>
<td>2.47</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>3.36</td>
<td>2.81</td>
<td>2.61</td>
<td>2.55</td>
<td>2.47</td>
<td>2.38</td>
<td>2.28</td>
<td>2.06</td>
</tr>
<tr>
<td>10</td>
<td>3.29</td>
<td>2.73</td>
<td>2.52</td>
<td>2.46</td>
<td>2.38</td>
<td>2.28</td>
<td>2.16</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>3.23</td>
<td>2.66</td>
<td>2.45</td>
<td>2.39</td>
<td>2.30</td>
<td>2.21</td>
<td>1.97</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>3.18</td>
<td>2.61</td>
<td>2.39</td>
<td>2.33</td>
<td>2.24</td>
<td>2.15</td>
<td>1.90</td>
<td></td>
</tr>
<tr>
<td>$\infty$</td>
<td>2.71</td>
<td>2.08</td>
<td>1.85</td>
<td>1.77</td>
<td>1.67</td>
<td>1.55</td>
<td>1.00</td>
<td></td>
</tr>
</tbody>
</table>

The measurements are the classification accuracy, number of base classifiers and other metrics. We obtain the best classification performance by tuning the kernel parameter $\delta$.

The cosine between classes in the kernel space is:

$$
\cos(\Theta_{C_1,C_2}) = \frac{1}{l_1 l_2} \sum_{i=1}^{l_1} \sum_{j=1}^{l_2} \exp(-\delta \|x_i^{(1)} - x_j^{(2)}\|^2)
$$

(23)

Summarizing Eqs. (21)–(23):

$$
D(C_1, C_2) = \cos(\Theta_{C_1,C_1}) + \cos(\Theta_{C_2,C_2}) - 2 \cos(\Theta_{C_1,C_2})
$$

(24)

Eqs. (21)-(24) show that the class separation distance can be calculated by inter class angle and within class angle.

4. Results

The RBF kernel is chosen to transform the samples. The measurements are the classification accuracy, number of base classifiers and other metrics. We obtain the best classification performance by tuning the kernel parameter $\delta$.
Six different gene expression datasets were used in the experiments, which are all highly linear-inseparable. The six gene expression datasets include: Leukemia, Breast, Central Nervous System (CNS), Colon, Small Round Blue Cell Tumor (SRBCT) and Lung datasets (Table 1). All datasets are downloaded from an open resourced website: http://datami.i2r.a-star.edu.sg/datasets/krbd/. We randomly selected features from the datasets 30 times, set

![Fig. 3. Classification accuracy rates based on different number of feature sets for various datasets.](image)

**Table 5**
Classification accuracy and the number of base classifiers.

<table>
<thead>
<tr>
<th>Int. level</th>
<th>Algorithm</th>
<th>3</th>
<th>7</th>
<th>10</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
<th>35</th>
<th>40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Breast</td>
<td>KPCA-RoF</td>
<td>0.6454</td>
<td>0.7233</td>
<td>0.8524</td>
<td>0.8576</td>
<td>0.8575</td>
<td>0.8577</td>
<td>0.8577</td>
<td>0.8576</td>
<td>0.8577</td>
</tr>
<tr>
<td></td>
<td>RoF</td>
<td>0.5921</td>
<td>0.6751</td>
<td>0.7532</td>
<td>0.7689</td>
<td>0.7672</td>
<td>0.8119</td>
<td>0.8154</td>
<td>0.8156</td>
<td>0.8163</td>
</tr>
<tr>
<td></td>
<td>RF</td>
<td>0.5827</td>
<td>0.6524</td>
<td>0.6856</td>
<td>0.7265</td>
<td>0.7471</td>
<td>0.7521</td>
<td>0.7537</td>
<td>0.7635</td>
<td>0.7613</td>
</tr>
<tr>
<td></td>
<td>Bagging</td>
<td>0.5612</td>
<td>0.6653</td>
<td>0.6748</td>
<td>0.7167</td>
<td>0.7425</td>
<td>0.7350</td>
<td>0.7432</td>
<td>0.7498</td>
<td>0.7429</td>
</tr>
<tr>
<td>CNS</td>
<td>KPCA-RoF</td>
<td>0.6788</td>
<td>0.7783</td>
<td>0.8742</td>
<td>0.8745</td>
<td>0.8743</td>
<td>0.8755</td>
<td>0.8739</td>
<td>0.8735</td>
<td>0.8741</td>
</tr>
<tr>
<td></td>
<td>RoF</td>
<td>0.5823</td>
<td>0.6459</td>
<td>0.7864</td>
<td>0.7998</td>
<td>0.8046</td>
<td>0.8076</td>
<td>0.8154</td>
<td>0.8158</td>
<td>0.8164</td>
</tr>
<tr>
<td></td>
<td>RF</td>
<td>0.5299</td>
<td>0.6348</td>
<td>0.6654</td>
<td>0.7012</td>
<td>0.7257</td>
<td>0.7659</td>
<td>0.7861</td>
<td>0.7935</td>
<td>0.7956</td>
</tr>
<tr>
<td></td>
<td>Bagging</td>
<td>0.5709</td>
<td>0.5413</td>
<td>0.6557</td>
<td>0.7068</td>
<td>0.7163</td>
<td>0.7118</td>
<td>0.7341</td>
<td>0.7323</td>
<td>0.7312</td>
</tr>
<tr>
<td>Leukemia</td>
<td>KPCA-RoF</td>
<td>0.6764</td>
<td>0.6358</td>
<td>0.7618</td>
<td>0.8416</td>
<td>0.8505</td>
<td>0.8517</td>
<td>0.8516</td>
<td>0.8516</td>
<td>0.8516</td>
</tr>
<tr>
<td></td>
<td>RoF</td>
<td>0.5465</td>
<td>0.5616</td>
<td>0.7386</td>
<td>0.7937</td>
<td>0.8157</td>
<td>0.8162</td>
<td>0.8169</td>
<td>0.8156</td>
<td>0.8160</td>
</tr>
<tr>
<td></td>
<td>RF</td>
<td>0.5412</td>
<td>0.5652</td>
<td>0.6891</td>
<td>0.7390</td>
<td>0.7693</td>
<td>0.7971</td>
<td>0.8065</td>
<td>0.8068</td>
<td>0.8071</td>
</tr>
<tr>
<td></td>
<td>Bagging</td>
<td>0.5498</td>
<td>0.5616</td>
<td>0.6256</td>
<td>0.6689</td>
<td>0.7006</td>
<td>0.7112</td>
<td>0.7133</td>
<td>0.7135</td>
<td>0.7133</td>
</tr>
<tr>
<td>Colon</td>
<td>KPCA-RoF</td>
<td>0.7143</td>
<td>0.8247</td>
<td>0.8759</td>
<td>0.9091</td>
<td>0.9093</td>
<td>0.9035</td>
<td>0.8963</td>
<td>0.8986</td>
<td>0.9090</td>
</tr>
<tr>
<td></td>
<td>RoF</td>
<td>0.6339</td>
<td>0.7382</td>
<td>0.8095</td>
<td>0.8571</td>
<td>0.8557</td>
<td>0.8512</td>
<td>0.8486</td>
<td>0.8471</td>
<td>0.8457</td>
</tr>
<tr>
<td></td>
<td>RF</td>
<td>0.6429</td>
<td>0.7147</td>
<td>0.7593</td>
<td>0.7854</td>
<td>0.7868</td>
<td>0.7884</td>
<td>0.7835</td>
<td>0.7791</td>
<td>0.7818</td>
</tr>
<tr>
<td></td>
<td>Bagging</td>
<td>0.5238</td>
<td>0.5735</td>
<td>0.6465</td>
<td>0.6858</td>
<td>0.6867</td>
<td>0.6889</td>
<td>0.6896</td>
<td>0.6863</td>
<td>0.6832</td>
</tr>
<tr>
<td>SRBCT</td>
<td>KPCA-RoF</td>
<td>0.6035</td>
<td>0.7243</td>
<td>0.8127</td>
<td>0.8444</td>
<td>0.8496</td>
<td>0.8484</td>
<td>0.8455</td>
<td>0.8407</td>
<td>0.8435</td>
</tr>
<tr>
<td></td>
<td>RoF</td>
<td>0.5873</td>
<td>0.6825</td>
<td>0.7907</td>
<td>0.8205</td>
<td>0.8274</td>
<td>0.8245</td>
<td>0.8228</td>
<td>0.8165</td>
<td>0.8153</td>
</tr>
<tr>
<td></td>
<td>RF</td>
<td>0.5641</td>
<td>0.6713</td>
<td>0.7264</td>
<td>0.7409</td>
<td>0.7425</td>
<td>0.7476</td>
<td>0.7446</td>
<td>0.7425</td>
<td>0.7408</td>
</tr>
<tr>
<td></td>
<td>Bagging</td>
<td>0.4486</td>
<td>0.5614</td>
<td>0.6429</td>
<td>0.6767</td>
<td>0.6782</td>
<td>0.6786</td>
<td>0.6772</td>
<td>0.6718</td>
<td>0.6746</td>
</tr>
<tr>
<td>Lung</td>
<td>KPCA-RoF</td>
<td>0.8574</td>
<td>0.8994</td>
<td>0.9248</td>
<td>0.9305</td>
<td>0.9340</td>
<td>0.9360</td>
<td>0.9349</td>
<td>0.9281</td>
<td>0.9247</td>
</tr>
<tr>
<td></td>
<td>RoF</td>
<td>0.8255</td>
<td>0.8725</td>
<td>0.9060</td>
<td>0.9195</td>
<td>0.9195</td>
<td>0.9195</td>
<td>0.9128</td>
<td>0.9106</td>
<td>0.8993</td>
</tr>
<tr>
<td></td>
<td>RF</td>
<td>0.7505</td>
<td>0.8067</td>
<td>0.8434</td>
<td>0.8718</td>
<td>0.8735</td>
<td>0.8768</td>
<td>0.8727</td>
<td>0.8693</td>
<td>0.8674</td>
</tr>
<tr>
<td></td>
<td>Bagging</td>
<td>0.5538</td>
<td>0.6191</td>
<td>0.6497</td>
<td>0.6597</td>
<td>0.6686</td>
<td>0.6738</td>
<td>0.6746</td>
<td>0.6787</td>
<td>0.6810</td>
</tr>
</tbody>
</table>

the confidence threshold as 0.95. The optimal parameters for the first three datasets are shown in Fig. 2. The choices of value $\delta$ are given as follows: $\delta = \{10^{-6}, 10^{-5}, \ldots, 10^5, 10^6\}$.

Fig. 2 shows the relationship between parameter values and the normalized class distances with those three datasets. The results are averaged of 30-time experiments. Every dataset only has one optimal parameter value. In Fig. 2, we illustrate the optimal parameters for the first three datasets, namely, $\delta_1 = \delta_2 = 0.9$ for Leukemia and Breast datasets respectively, and $\delta_3 = 0.8$ for the CNS dataset.

We compare our algorithm with the Bagging algorithm and the original RoF algorithm. Experimental results verify the effectiveness of the proposing algorithm. All algorithms use C4.5 DT as based classifier. The main control variables are the number of base classifiers $N$ and the number of samples $M$. We adjust one variable at each time to obtain the results.

### 4.1. Statistical analysis

For statistical analysis, the variance $S^2$, weighted average $\bar{X}$ and $F$-measure are utilized to verify the classification improvement. The $F$-measure values, which are derived from the confusion matrix, directly reflect the classification performance differences between different classifiers (Yan et al., 2017). Moreover, the comparison between $F$-measure and $F$-distribution values provides a more precise
measurement for the performance improvement using KPCA-RoF over existing algorithms.

The weighted average:

$$\bar{X} = \frac{\sum X}{n}$$  \hspace{1cm} (25)

The variance is:

$$S^2 = \frac{(X - \bar{X})^2}{(n - 1)}$$  \hspace{1cm} (26)

The $F$-measure is defined as in Huang and Mi (2007):

$$F - \text{measure} = \frac{n_1S_1^2/(n_1 - 1)}{n_2S_2^2/(n_2 - 1)}$$  \hspace{1cm} (27)

Table 2 lists the three variables ($\bar{X}$, $S^2$, $F$) of different approaches on different gene expression datasets.

We compare the values of $F$-measure to the values in $F$-distribution Table (Table 3). If the value of $F$-measure is greater than the corresponding value in the $F$-distribution table, we believe that the improvement of our algorithm over the other algorithm for the particular dataset is significant. From the results in Tables 2 and 3, we can see that in most cases, the probability of significant improvement using our algorithm is high. The reason of the classification performance improvement mainly lies on the feature space transformation using KPCA. For each segmented training subset, the KPCA algorithm transforms the original training samples into lower feature space, which potentially
increase the possibility for linear separation in high-
dimension.

4.2. The classification performance using KPCA-RoF

The number of feature sets and the number of base clas-
sifies are the two factors that affect the classification accu-
ry. For the first part of this subsection, the number of base classifiers remains unchanged whereas the number of feature sets varies. The results of the classification accuracy for different datasets are shown in Table 4.

In Table 4, classification accuracy of each dataset is listed. The experiment results show that the classification accuracy does not vary too much with the number of feature subset is more than 9, which gives the most economical number of feature set segmentations (Fig. 3).

Table 5 shows the classification accuracy rates for different base classifier numbers. From the results, a reasonable accuracy is obtained with 15 base classifiers, and further increment of the number of base classifiers does not increase the accuracy much. Several other ensemble al-
gorithms are tested for each dataset for comparison purposes, i.e.: KPCA-RoF, RoF, RF and Bagging. It is noted that in order for fair comparison, we transform the datasets used in RoF, RF and bagging algorithms to the same number of features as the dataset used in KPCA-RoF algorithm.

Figs. 4–9 show the classification accuracy for each data-
set using different ensemble algorithms. The comparison results show that with the increment of the number of base classifiers, the classification accuracy increases. In general, the classification results can be interpreted as follows: Bagging decision tree (BDT) is the worst among all four
methods; our method is the best. The BDT is a simple integration of DT, therefore the accuracy improvement will not be obvious; RF adds the random segmentation of the feature space, which gives better classification results. From the experimental results, RoF and our method produce the best classification accuracy because of the feature space segmentation and transformation. Our method achieves highest accuracy with a small number of base classifiers. This shows that the nonlinear transformation compared with the linear transformation is more adaptable to various linearly inseparable datasets.

5. Conclusions

RoF is a machine learning technique that generates a transformation matrix through the segmentation, sampling and transformation of the dataset. Based on the traditional RoF algorithm, we have designed a hybrid classification method to deal with the linearly inseparable and high-dimensional gene expression datasets. The original training dataset was randomly segmented into subsets, where customized KPCA transformation was applied to each of the subsets. The RBF kernel was employed to map the linearly inseparable data into linearly separable feature space. For each transformed subset, a decision tree based classifier was trained, and the combination of all DTs formed the final ensemble classifier which was named as KPCA-RoF.

In the experiment part, we carefully tuned the kernel parameters to obtain the optimal results for classification. With the optimized kernel parameters, the proposed KPCA-RoF algorithm is capable to produce highest classification accuracy over existing classifiers, such as BDT, RF and RoF, which is also verified by the concept of F-measure. For linearly inseparable datasets, such as the gene expression datasets, the KPCA-RoF has demonstrated its classification power over various gene expression datasets downloaded from publically available gene database.

The limitation of the current work is that the testing datasets were downloaded from a publicly available database. The number of samples were small for all six tested datasets. Real-world and large sized gene expression datasets were not tested. Future work of this study includes applying the KPCA-RoF method to the real-world medical applications. For real-world large size gene expression data, the big data preprocessing step might be required, where the RoF structure can be built based on a distributed system, e.g., a cloud server.

Acknowledgments

This study is supported by National Natural Science Foundation of China (Nos. 61272315, 61303183, 61402417, and 61602431), Zhejiang Provincial Natural Science Foundation (Nos. Y1110342, LY15F020037, and Y1080950). It is also supported by the Priority Academic Program Development of Jiangsu Higher Education Institutions (PAPD) and Jiangsu Collaborative Innovation Center on Atmospheric Environment and Equipment Technology (CICAEET).

Conflict of interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript.

References


