Sensors and Actuators B 212 (2015) 353–363

Contents lists available at ScienceDirect

Sensors and Actuators B: Chemical

journal homepage: www.elsevier.com/locate/snb

Feature selection and analysis on correlated gas sensor data with recursive feature elimination

Ke Yan [a], David Zhang [b,*]

[a] Department of Electronic Engineering, Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China
[b] Biometric Research Centre, Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

ARTICLE INFO

Article history:
Received 18 August 2014
Received in revised form 1 February 2015
Accepted 6 February 2015
Available online 16 February 2015

Keywords:
Feature selection
Feature ranking
SVM-RFE
Correlation bias
Breath analysis
Transient feature

ABSTRACT

Support vector machine recursive feature elimination (SVM-RFE) is a powerful feature selection algorithm. However, when the candidate feature set contains highly correlated features, the ranking criterion of SVM-RFE will be biased, which would hinder the application of SVM-RFE on gas sensor data. In this paper, the linear and nonlinear SVM-RFE algorithms are studied. After investigating the correlation bias, an improved algorithm SVM-RFE + CBR is proposed by incorporating the correlation bias reduction (CBR) strategy into the feature elimination procedure. Experiments are conducted on a synthetic dataset and two breath analysis datasets, one of which contains temperature modulated sensors. Large and comprehensive sets of transient features are extracted from the sensor responses. The classification accuracy with feature selection proves the efficacy of the proposed SVM-RFE + CBR. It outperforms the original SVM-RFE and other typical algorithms. An ensemble method is further studied to improve the stability of the proposed method. By statistically analyzing the features' rankings, some knowledge is obtained, which can guide future design of e-noses and feature extraction algorithms.

1. Introduction

Feature selection (FS) is a widely-used technique in pattern recognition applications. By removing irrelevant, noisy, and redundant features from the original feature space, FS alleviates the problem of overfitting and improves the performance of the model. The time and space cost of the learning algorithm can also be reduced. More importantly, we can gain a deeper insight of the data by analyzing the importance of the features [1,2]. Many researchers have explored the use of FS techniques in electronic nose (e-nose) systems and achieved good results [3–9].

In the context of classification, FS algorithms can be roughly divided into three categories: filters, wrappers and embedded methods, based on how they interact with classifiers [1,2]. Filters evaluate each feature by predefined criteria, such as correlation criteria and information theoretic criteria [1], which are independent from classifiers. Wrappers treat classifiers as black boxes and aim at finding a feature subset that has the minimum cross-validation error on the training data. Examples of wrappers include sequential forward selection [3], genetic algorithms, and simulate annealing [4]. Embedded methods generally include two kinds of approaches. In some methods, such as a decision tree [7], the training of the classifier intrinsically selects a subset of features. Some methods estimate the importance of the features from the coefficients in the classifiers, e.g. the algorithm in [5].

Support vector machine recursive feature elimination (SVM-RFE) is an embedded FS algorithm proposed by Guyon et al. [10]. It uses criteria derived from the coefficients in SVM models to assess features, and recursively removes features that have small criteria. It has both linear and nonlinear versions. The nonlinear SVM-RFE uses a special kernel strategy [10,11] and is preferred when the optimal decision function is nonlinear. As a backward elimination method, SVM-RFE is able to model the dependencies among features. Compared to wrappers, SVM-RFE does not use the cross-validation accuracy on the training data as the selection criterion, thus is (1) less prone to overfitting; (2) able to make full use of the training data; (3) much faster, especially when there are a lot of candidate features. As a result, it has been successfully applied in many problems, especially in gene selection [10–15].

However, there is still one problem in SVM-RFE that has not been addressed. When some of the candidate features are highly correlated, the assessing criteria of these features will be influenced, and their importance will be underestimated. Inspired by [16], we call this phenomenon "correlation bias". It is a crucial problem especially for gas sensor features that are often correlated. In this paper,

* Corresponding author. Tel.: +852 27667271.
E-mail addresses: yank10@mails.tsinghua.edu.cn (K. Yan), csdzhang@comp.polyu.edu.hk (D. Zhang).

http://dx.doi.org/10.1016/j.snb.2015.02.025
0925-4005/

a simulated experiment is first employed to illustrate this problem. Then a novel strategy, correlation bias reduction (CBR), is proposed to reduce this potential bias in both linear and nonlinear SVM-RFE. Finally, an ensemble method is suggested to improve the stability of the feature selection results.

It is known that human breath contains biomarkers that can be used for disease diagnosis [17]. E-nose systems have been applied to analyze breath samples. In this paper, the proposed method is evaluated on two breath analysis datasets. The first breath analysis dataset was collected by an e-nose with 10 gas sensors, three of which were metal oxide semiconductor (MOS) sensors under temperature modulation (TM) [18]. The dataset contains 295 samples from healthy subjects and 279 from diabetics. The second dataset was collected by an e-nose with 12 MOS sensors [19]. The breath samples were from healthy subjects and also subjects with diabetes, renal disease, and airway inflammation, respectively. Over 1000 features are extracted from the gas sensors' responses. The comprehensive feature set contains seven kinds of transient features. Experimental results show that the Gaussian SVM-RFE is better than the linear one, as well as other typical algorithms. The proposed CBR strategy further enhances the accuracy. The ensemble method is proved to have better stability. Furthermore, systematic statistical analysis on the features' rankings reveals useful information about which sensors, feature types and TM voltages are more important. For example, TM sensors significantly outperform the ones operated under constant temperature. Phase feature extracted from TM sensors is proved to be the most effective feature. The information provides guidance for future e-nose and feature designing.

The manuscript is organized as follows. Section 2 describes the details of the linear and nonlinear SVM-RFE algorithm. Section 3 investigates the correlation bias problem and proposes SVM-RFE + CBR. Section 4 introduces the breath analysis datasets and feature extraction methods. Section 5 shows the results of the FS experiments and provides the feature analysis results. Section 6 concludes the paper.

2. SVM-RFE

2.1. Linear SVM-RFE

The output of SVM-RFE is a ranked feature list. Feature selection can be achieved by choosing a group of top-ranked features. The ranking criterion of SVM-RFE is closely related to the SVM model. SVM is a popular algorithm for classification partially due to its high accuracy and good generalization ability. It has been successfully applied in many e-nose applications [9]. Therefore, ranking criterion derived from its model will probably have good performance.

The intuition of SVM is to find a separating hyperplane with the largest margin. In linear separable cases, the margin is twice the distance between the separating hyperplane and the training sample closest to it [20]. Given a set of training samples $\{\boldsymbol{x}_i, y_i\}$, $\boldsymbol{x}_i \in \mathbf{R}^d, y_i \in \{-1, 1\}, i = 1, \ldots, n$, the decision function of a linear SVM is

$$f(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x} + b. \tag{1}$$

It can be proved that the margin M is simply $2/\|\boldsymbol{w}\|$, thus maximizing the margin is equivalent to minimizing $\|\boldsymbol{w}\|^2$ under constraints. The dual form of the Lagrangian formulation of the problem can be written as [20]:

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i \cdot \boldsymbol{x}_j, \tag{2}$$

where α_i are the Lagrange multipliers. Solutions of α_i can be found by maximizing L_D under constraints $\alpha_i \geq 0$ and $\sum_{i=1}^{n} \alpha_i y_i = 0$. The samples corresponding to nonzero α's are known as support vectors. Then the weight vector $\boldsymbol{w}$ can be obtained by

$$\boldsymbol{w} = \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i. \tag{3}$$

The ranking criterion for feature k is the square of the kth element of $\boldsymbol{w}$,

$$J(k) = w_k^2. \tag{4}$$

In each iteration of the recursive feature elimination (RFE), a linear SVM model is trained. The feature with the smallest ranking criterion is removed since it has the least effect on classification [13]. The remaining features are kept for the SVM model in the next iteration. This process is repeated until all the features have been removed. Then the features are sorted according to the order of removal. The later a feature is removed, the more important it should be. When the feature dimension is high, removing features one by one will be time-consuming. In such cases, more than one feature can be removed in each iteration [10]. However, this strategy may influence the precision [13] and cause the correlation bias problem, which will be described in Section 3.1.

2.2. Nonlinear SVM-RFE

Most gene selection problems have much more features (several thousand) than samples (less than 100), so linear SVM-RFE is more suitable in these cases to avoid overfitting. But in many other situations where the number of samples is larger, nonlinear SVM-RFE can be expected to outperform the linear one since it can fit the data with less bias.

Nonlinear SVM considers to map the features into a new space with higher dimension:

$$\boldsymbol{x} \in \mathbf{R}^d \mapsto \Phi(\boldsymbol{x}) \in \mathbf{R}^h. \tag{5}$$

In the new space, the samples are expected to be linearly separable. Thus Eq. (2) can be rewritten as

$$L_D = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_j). \tag{6}$$

Note that the only form that $\Phi(\boldsymbol{x})$'s are involved in the training algorithm is their inner product. So we can replace $\Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_j)$ with a kernel function $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ without knowing the explicit form of Φ. This is a particularly useful trick because it is hard to determine the form of Φ in real-world problems. There are several choices for kernel functions, though, one common choice being the Gaussian kernel

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = e^{-\gamma \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}. \tag{7}$$

Since the form of Φ is unknown, the weight vector $\boldsymbol{w}$ cannot be obtained. However, linear SVM-RFE can be extended to nonlinear cases via a special strategy. If the removal of a feature causes only small changes in the objective function Eq. (6), the feature should be removed [10,11]. This leads to the following ranking criterion for feature k:

$$J(k) = \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x}_i^{(-k)}, \boldsymbol{x}_j^{(-k)}). \tag{8}$$

The notation $(-k)$ means the feature k has been removed, i.e. $\boldsymbol{x}^{(-k)} \in \mathbf{R}^{d-1}$. The above criterion is the difference of Eq. (6) before and after removing feature k while keeping the α's unchanged. The features with small J's will be eliminated in each iteration of RFE. This criterion is applicable for all kinds of kernels. When the linear

kernel is used ($K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i \cdot \boldsymbol{x}_j$), it is equivalent to the linear SVM-RFE. This nonlinear version of SVM-RFE costs a little more time than the linear version, but some techniques can be applied to accelerate it, which will be introduced in Section 3.3.

3. Improved SVM-RFE with correlation bias reduction

3.1. Correlation bias

Some classification applications contain highly correlated features. For example, in gene classification, features represented by probes that either have similar molecular functions or genomic locations are highly correlated [16]. In e-nose applications, gas sensors are known to be cross-sensitive. Besides, when multiple transient features (e.g. the magnitude, derivative and integral at different time points) are extracted from a gas sensor's response, high correlation often exists among these features. The correlation brings adverse impacts to some feature selection algorithms. Toloşi and Lengauer [16] evaluated feature importance based on Lasso penalized logistic regression and random forest. They discovered that the evaluation was biased in highly correlated feature groups. Concretely, the features in the groups received smaller weights due to the shared responsibility in the classification models. Therefore, the importance of the features will be underestimated even if they are highly relevant. The larger the group size, the larger the underestimation. This phenomenon is called "correlation bias" (CB) in [16].

The phenomenon has not been studied in SVM-RFE. However, both linear and nonlinear SVM-RFE are affected due to the similar reason for Lasso penalized logistic regression and random forest. We have conducted a simulated experiment to illustrate the phenomenon. A synthetic dataset is generated with 500 samples and 100 features. There are 22 latent variables $z_1, \ldots, z_{22}$ that contribute to the class label y. They are all drawn from the normal distribution $\mathcal{N}(0, 1)$. The class label y is decided by

$$y = \text{sign}\left(\sum_{i=1}^{22} w_i z_i + \epsilon\right), \tag{9}$$

where ϵ is a perturbation term drawn from $\mathcal{N}(0, 0.01)$. $w_1, \ldots, w_{22}$ are predefined weights. From each latent variable, a group of real features x are generated:

$$x_{ij} = z_i + \epsilon, \quad j = 1, \ldots, k_i; \quad i = 1, \ldots, 22. \tag{10}$$

k_i is the size of the ith group, $\epsilon \sim \mathcal{N}(0, 0.01)$. So the features in each group are highly correlated. The 22 groups of features are concatenated and constitute $x_1, \ldots, x_{80}$. $x_{81}, \ldots, x_{100}$ are pure noise features drawn from $\mathcal{N}(0, 1)$. The weights and sizes of the feature groups are listed in Table 1. For example, x_1–x_{20} belong to two groups with size 10 and $w = 1$.

The 100 features are evaluated using the feature ranking criteria introduced in Section 2. Note that the RFE procedure is not performed. In order to compare the criteria among groups, they are normalized to [0,1]. The results for linear and nonlinear SVM-RFE are displayed in Fig. 1. It is clear that both the group size and the weight influence the criteria. Ideally, the criteria should only depend on the weight, so the criteria for feature 1–40, 41–80 and 81–100 should be about 1, 0.5 and 0, respectively. However, because of CB, the features in larger groups receive smaller criteria. The features with group size 5 and 10 receive criteria comparable to that of the noise features. In this case, if a batch of features are removed in one iteration of RFE, features in large groups are likely to be removed entirely.

3.2. Correlation bias reduction

Highly correlated features bring wrong estimations to several embedded FS algorithms including SVM-RFE. They also affect regression applications, causing large variance of the estimates and inaccurate prediction [21]. In order to deal with the problem, some methods have been proposed. An intuitive method is to replace each group of highly correlated features with one representative before selection or regression. For example, Park et al. [21] performed hierarchical clustering on features and used the cluster centroids for regression. Another idea is to perform selection or regression on feature groups instead of single features. For example, Sharma et al. [22] proposed to automatically group and select correlated features based on a penalization scheme. However, the scheme only applies to linear models.

Our method differs from the methods described above. It makes use of the RFE procedure to reduce the influence brought by CB. For efficiency, it is impractical for the RFE procedure to remove one feature each time if the feature dimension is high. When a batch of features are removed in one iteration of RFE, a group of correlated features may be removed entirely. This may either because the features are truly irrelevant, or because their ranking criteria have been incorrectly underestimated. In both conditions, we can move a representative feature of the group back to the surviving feature list. Then it can be evaluated again in the next iteration without the influence of CB. The group representative can be chosen as the feature with the highest criterion in this iteration. This strategy does not change the candidate feature set or the ranking criterion, but monitors and corrects the potentially wrong decisions due to CB.

The details of the correlation bias reduction (CBR) algorithm are shown in Algorithm 3.1. $\mathcal{F}^{out}$ is the list of features to be removed in one iteration of RFE. $\mathcal{F}^{in}$ is the list of features that survives. The purpose of the algorithm is to move potentially useful features from $\mathcal{F}^{out}$ back to $\mathcal{F}^{in}$. In order to identify highly correlated feature groups in $\mathcal{F}^{out}$, two thresholds T_c and T_g are used. We start from examining the feature with the highest criterion in $\mathcal{F}^{out}$ and denote it as feature k. If there exist more than T_g features (including k) whose absolute correlation coefficient with k is larger than T_c, they are identified as a group. If none of the group members are in $\mathcal{F}^{in}$, k should be moved to $\mathcal{F}^{in}$ since it has the highest criterion in the group. This operation is repeated on all features in $\mathcal{F}^{out}$. In Algorithm 3.1, $|\mathcal{F}|$ represents the number of elements in $\mathcal{F}$.

Algorithm 3.1. CBR($\mathcal{F}^{in}$, $\mathcal{F}^{out}$)

Input: Feature list $\mathcal{F}^{in}$ and $\mathcal{F}^{out}$; Thresholds T_c and T_g.
1: Sort $\mathcal{F}^{out}$ according to the descending order of the ranking criteria.
2: **for** $p = 1$ **to** $|\mathcal{F}^{out}|$ **do**
3: Suppose feature k is the pth element of the sorted $\mathcal{F}^{out}$, let
$\mathcal{G}^{out} \leftarrow \{i \in \mathcal{F}^{out} || \text{corr}(i, k)| > T_c\}$;
$\mathcal{G}^{in} \leftarrow \{j \in \mathcal{F}^{in} || \text{corr}(j, k)| > T_c\}$.
4: **if** $|\mathcal{G}^{out}| > T_g$ and $|\mathcal{G}^{in}| == 0$ **then**
5: $\mathcal{F}^{out} \leftarrow \mathcal{F}^{out} - k$;
$\mathcal{F}^{in} \leftarrow \mathcal{F}^{in} \cup k$.
6: **end if**
7: **end for**
Output: Modified $\mathcal{F}^{in}$ and $\mathcal{F}^{out}$.

The larger T_g, the fewer groups will be identified. In practice, we find that setting T_g to 1 or 2 achieves comparable accuracy. Larger values of T_g will degrade the accuracy since some groups of correlated features are eliminated too early. T_c is the correlation threshold. We will explore the effect of different T_c values on the accuracy in Section 5.2. The experimental results in Section 5 prove that the CBR strategy improves the performance of SVM-RFE.

Table 1
Weights and sizes of the feature groups in the synthetic dataset.

Parameter	Real feature index								
	1–20	21–30	31–36	37–40	41–60	61–70	71–76	77–80	81–100
Weight w	1	1	1	1	0.5	0.5	0.5	0.5	0
Group size k	10	5	2	1	10	5	2	1	1

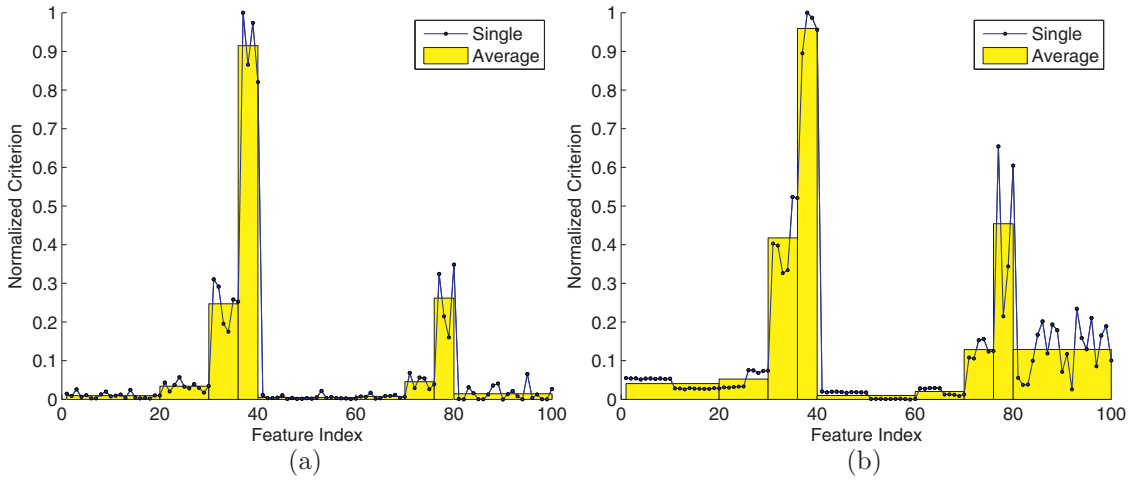

Fig. 1. Normalized feature ranking criteria of (a) linear (Eq. (4)) and (b) nonlinear (Eq. (8)) SVM-RFE in the synthetic dataset. The curves represent the criteria of each single feature. The bars show the average criteria of the features in the groups with the same size and weight (see Table 1). It is clear that if the group size increases or the weight decreases, the criterion decreases.

3.3. Efficient implementation of SVM-RFE with CBR

This section describes some details on the implementation of the proposed algorithm. First, the number of features that are removed in each iteration of RFE should be determined. In this paper, a method that simultaneously considers the time cost and precision is used [11]. At the beginning of the algorithm, one half of the remaining features are removed in each iteration. When the number of the remaining features is less than an elimination number threshold T_e, they are removed one by one in the following iterations for better precision.

A technique can be applied to accelerate the calculation of the ranking criterion for nonlinear SVM-RFE (Eq. (8)) with Gaussian kernel. First, Eq. (8) is expressed in a matrix form:

$$J(k) = \frac{1}{2}(\boldsymbol{\beta}^{\mathrm{T}} H \boldsymbol{\beta} - \boldsymbol{\beta}^{\mathrm{T}} H^{(-k)} \boldsymbol{\beta}). \tag{11}$$

Here, $\boldsymbol{\beta}$ is the column vector of signed α's, i.e. $\beta_i = \alpha_i y_i$. Only the nonzero α's are included. H is the kernel matrix, $H_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Only the support vectors are included. For the Gaussian kernel, we have $H_{ij} = e^{-\gamma S_{ij}}$, where $S_{ij} = \| \boldsymbol{x}_i - \boldsymbol{x}_j \|^2$. It is easy to prove that

$$S_{ij}^{(-k)} = \|\boldsymbol{x}_i^{(-k)} - \boldsymbol{x}_j^{(-k)}\|^2 \tag{12}$$

$$= S_{ij} - (x_i^{(k)} - x_j^{(k)})^2, \tag{13}$$

where $x_i^{(k)} \in \mathbf{R}$ is the kth feature of the ith support vector. When computing $S_{ij}^{(-k)}$, we can use Eq. (13) to replace the original Eq. (12), since the matrix S can be cached and reused, and computing the scalar operation in Eq. (13) is much easier than the vector norm in Eq. (12). According to experiments using Matlab, Eq. (13) is about 5 times faster.

The complete algorithm of SVM-RFE with CBR is summarized in Algorithm 3.2.

Algorithm 3.2. SVM-RFE with CBR

Input: A set of training samples with feature dimension d;
An SVM training algorithm (linear or nonlinear); T_e.
1: Initialize the list of surviving features $\mathcal{F}^{in} \leftarrow \{1, \ldots, d\}$;
the list of eliminated features $\mathcal{F}^{out} \leftarrow \emptyset$.
2: **while** $\mathcal{F}^{in} \neq \emptyset$ **do**
3: Train an SVM model with the features in $\mathcal{F}^{in}$.
4: Calculate the features' ranking criteria with Eq. (4), or Eqs. (11) and 13.
5: Sort $\mathcal{F}^{in}$ according to the descending order of the ranking criteria.
6: **if** $|\mathcal{F}^{in}| > T_e$ **then**
7: $r = \min(\text{floor}(|\mathcal{F}^{in}|/2), |\mathcal{F}^{in}| - T_e)$,
8: **else**
9: $r = 1$.
10: **end if**
11: $\mathcal{F}^{removing} \leftarrow$ the last r elements in $\mathcal{F}^{in}$;
$\mathcal{F}^{in} \leftarrow$ the first $|\mathcal{F}^{in}| - r$ elements in $\mathcal{F}^{in}$.
12: **if** $r > 1$ **then**
13: Call Algorithm 3.1:$(\mathcal{F}^{in}, \mathcal{F}^{removing}) \leftarrow \text{CBR}(\mathcal{F}^{in}, \mathcal{F}^{removing})$.
14: **end if**
15: $\mathcal{F}^{out} \leftarrow [\mathcal{F}^{removing}, \mathcal{F}^{out}]$.
16: **end while**
Output A ranked list of features $\mathcal{F}^{ranked} = \mathcal{F}^{out}$, the most important feature in the first place.

3.4. Stability improvement with ensemble method

The stability of an FS algorithm is a topic of recent interest [23–27]. A stable FS algorithm is important for data mining applications such as bioinformatics. Stability describes the sensitivity of a method to variations in the training set [27]. If the training set is perturbed, the difference in selected features should not be too large. The Jaccard index is a widely-used criterion to measure the difference between two selected feature subsets A and B [26,25,27]:

$$S(A, B) = \frac{|A \cap B|}{|A \cup B|}. \tag{14}$$

Its value ranges from 0 to 1, with 0 meaning that the two subsets have no overlap and 1 meaning that they are identical. When evaluating the stability of an FS algorithm, we can use the N-fold

Table 2
Summary of the sensor array used in dataset 1 [18].

No.	Model	Manufacturer	Function
1	TGS4161		CO_2
2	TGS822	Figaro Inc., Japan	
3	TGS826		
4	TGS2610-D00		VOCs (e.g. acetone), H_2, CO, NH_3, H_2S, etc.
5	SP3S-AQ2	FIS Inc., Japan	
6	GSBT11	Ogam Inc., Korea	
7	WSP2111	Winsen Inc., China	
8	TGS2600-TM	Figaro Inc., Japan	
9	TGS2602-TM		
10	WSP2111-TM	Winsen Inc., China	
11	HTG3515CH	Humirel Inc., France	Temperature
12			Humidity

cross-validation strategy. After N subsets have been selected based on N training sets, the Jaccard index is computed for all $N(N-1)/2$ pairs of subsets. The final stability is the average over all pairs [27].

To improve the stability of FS algorithms, one of the popular ideas is to use an ensemble method [24,26], i.e. to aggregate the outputs of the single feature selectors. In this paper, we apply this method to SVM-RFE + CBR. Part (9/10 in this paper) of the training samples are randomly picked to generate a ranked feature list. The process is repeated M times, then the average rank of each feature is used to determine its final rank. In this way, the features with stably good performance are more likely to rank higher. Note that the stability issues and performance of the ensemble method will be separately discussed in Section 5.5. The results described in Sections 5.1–5.4 are obtained without the ensemble method.

4. Datasets and feature extraction

Breath analysis is an important application of e-nose systems. It is a noninvasive and convenient way to assist disease screening. In this section, the breath analysis datasets used in this paper and the feature extraction methods will be introduced.

4.1. Dataset 1

4.1.1. Description

Dataset 1 consists of breath samples from healthy and diabetic subjects. Diabetes is a great threat to human health. Researchers have found that the disease is related to abnormal concentration of acetone and some other volatile organic compounds (VOCs) in breath [28]. A breath analysis system was proposed by Yan et al. [18] to measure breath samples of healthy people and diabetics. They developed an e-nose with a carbon dioxide sensor, a temperature-humidity sensor and 9 metal oxide semiconductor (MOS) sensors. All of them are commercially available. The details of the sensor array are listed in Table 2. The carbon dioxide sensor was utilized to compensate for the difference in proportion of alveolar air. The MOS sensors were carefully selected for better accuracy in diabetes identification [29]. It is worth noting that three of the MOS sensors were operated under temperature modulation (sensor 8–10 with the notation "-TM" in Table 2). They were heated by a staircase voltage oscillated between 3 V and 7 V. Fig. 2 illustrates the waveform of the heating voltage and compares typical responses of a TM sensor and an ordinary sensor. The duration of each breath sample was 144 s. The measurement procedure includes 4 stages, which are also shown in Fig. 2. The baseline values of the sensors are recorded in the baseline stage. In the injection stage, the breath sample is drawn from a gas bag to the gas room. In the reaction stage, the responses of the sensors approach their steady states. The gas room is purged with clean air in the purge stage.

A total of 295 healthy and 279 diabetes breath samples were collected. Before feature extraction, the samples are preprocessed.

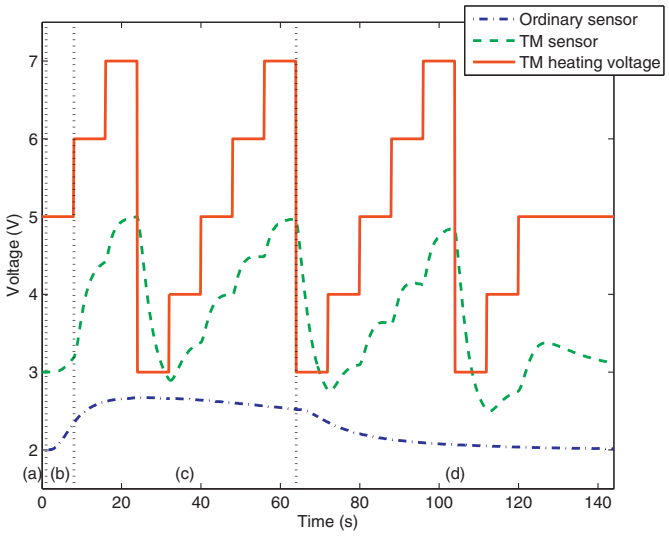

Fig. 2. Typical curves in dataset 1. Solid line: the heating voltage of temperature modulated (TM) sensors. Dashed line: a typical response curve of a TM sensor. Dash dot line: a typical response curve of an ordinary sensor. The vertical dotted lines separate the four stages of the sampling procedure: (a) baseline stage; (b) injection stage; (c) reaction stage; (d) purge stage.

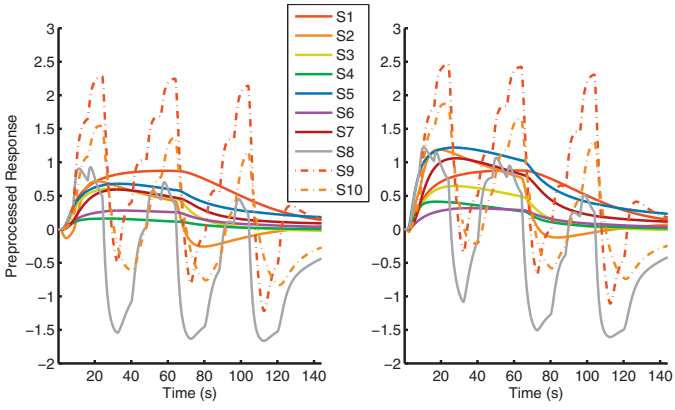

Fig. 3. Average preprocessed responses of the two classes in dataset 1 [18]. Left: healthy; right: diabetes. The sensor models can be found in Table 2.

First, the baseline values are subtracted from the sensor responses. Humidity compensation is necessary because breath samples contain water vapor. Linear humidity-response models are built for each sensor, then applied to rectify the samples [18]. Fig. 3 exhibits the average preprocessed samples of healthy and diabetic subjects. Only the carbon dioxide sensor (S1) and MOS VOC sensors (S2–S10) are drawn. For S2–S10, the responses in diabetes samples are larger than that in healthy samples, showing that the concentration of VOCs in breath of diabetics is higher than that of healthy subjects. Besides, the curve shape of S8 is significantly different between the two classes.

4.1.2. Transient feature extraction

Traditional features of gas sensors are their steady-state responses. However, additional useful information is carried in the transient responses [30,9]. Transient responses are often related to the change of gas flow (injection/purge) or temperature (for TM sensors). In this paper, 1712 transient features are extracted from sensor 1–10 in dataset 1. The feature set includes magnitude, difference, derivative, 2nd derivative, integral, time constant and phase features. It is a larger and more comprehensive feature set than previous studies [3,6,7], which enables us to (1) enhance the

Table 3
Feature description for ordinary sensors in dataset 1.

Feature type	Description	#Features
Magnitude	Down-sampled values of the curve's magnitude M.	21
	The maximum magnitude.	1
	Down-sampled values of the normalized magnitude $\tilde{M}$, $\tilde{M} = M/\max(M)$.	21
Difference	The difference F of magnitude M, $F_i = M(t_{i+1}) - M(t_i)$, $t = [0, 8, 36, 64, 92, 120]$, $i = 1, \ldots, 5$.	5
Derivative	Down-sampled values of the curve's derivative D.	21
	The maximum and minimum derivative.	2
2nd derivative	The maximum and minimum 2nd derivative in both injection and purge stage.	4
Integral	The integral of the 5 intervals of the curve, the intervals are the same with the difference feature.	5
Time constant	The time when the magnitude reaches 30%, 60%, 90%, and 100% of its maximum value (T_{30}, T_{60}, T_{90}, T_{max}), and 90%, 60%, and 30% of its maximum value in the purge stage (T_{-90}, T_{-60}, T_{-30}).	7
	The time when the derivative reaches its maximum and minimum values.	2
	The time when the 2nd derivative reaches its maximum and minimum values in both injection and purge stage.	4
Phase feature	The phase feature is proposed in [35]. First, the response is transformed to the phase space, which is spanned by its magnitude and derivative. Then, the phase features are defined by $P_i = \int_{M(t_i)}^{M(t_{i+1})} D\,\mathrm{d}M$, t is the same with the difference feature.	5

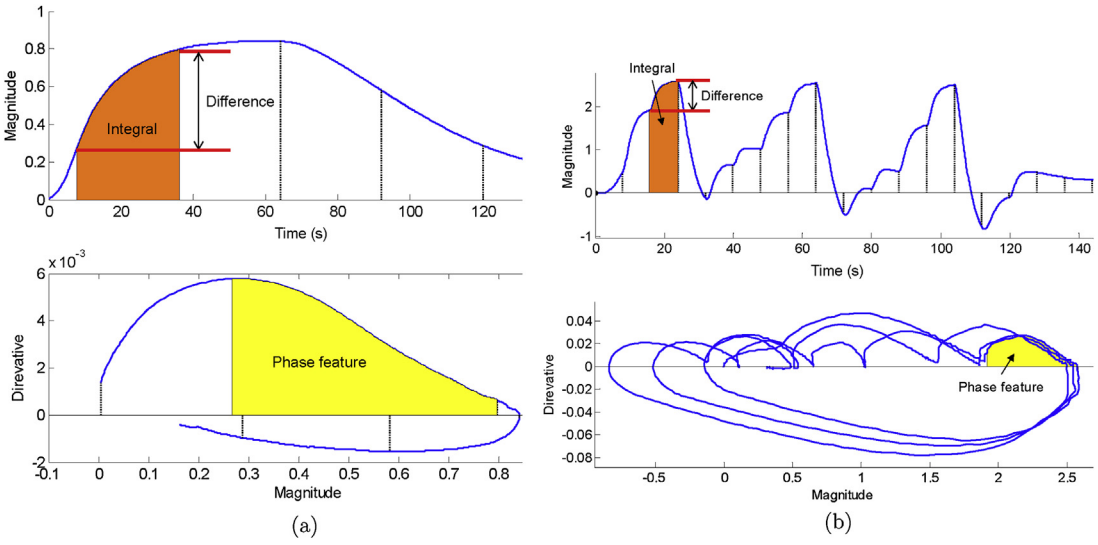

Fig. 4. Examples of the difference, integral, and phase feature for (a) ordinary sensors and (b) TM sensors.

classification accuracy, for the best feature subset in a large candidate set should be better than that in a small set; (2) perform a systematic statistical analysis on the features; (3) testify the performance of the proposed FS algorithm in a large correlated feature set.

The features extracted from each ordinary sensor are described in Table 3. There are altogether 98 features. The difference, integral, and phase features are calculated on 5 intervals of the curve (1 in injection stage, 2 in reaction stage, and 2 in purge stage), which are illustrated in Fig. 4(a). The shape of a TM sensor's response is more complex and informative. However, the features defined in Table 3 can still be used to describe the transient of the curve. Because the response of a TM sensor has 18 "stairs", transient features are extracted on every stair. The features are similar to those in Table 3, but the features related to the 2nd derivative in the purge stage and the time constant features T_{30}, T_{60}, T_{90}, T_{-90}, T_{-60}, and T_{-30} are not included. The feature dimension for each TM sensor is $18 \times 19 = 342$.

4.2. Dataset 2

The second dataset was collected by a breath analysis system designed by Guo et al. [19]. It was equipped with a carbon dioxide sensor and 11 MOS sensors. The details of the sensor array are listed in Table 4. All sensors are commercially available from Figaro Inc., Japan. They were operated under constant heating voltage. The sensor array is able to detect biomarkers of several kinds of diseases. There are 135 healthy samples, 181 diabetes samples, 167 renal disease samples, and 126 airway inflammation samples in the dataset. Typical samples in the dataset are displayed in Fig. 5. The duration of each breath sample was 90 s. The data preprocessing algorithm includes baseline subtraction and signal normalization [19]. After preprocessing, 1140 transient features are extracted. The features

Table 4
Summary of the sensor array used in dataset 2 [19].

No.	Model	Gas	Sensitivity (ppm)
1	TGS2600	H_2, CO and VOCs	1–30
2	TGS2602	VOCs	1–30
3	TGS2611-C00	VOCs	500–10,000
4	TGS2610-C00	VOCs	500–10,000
5	TGS2610-D00	VOCs	500–10,000
6	TGS2620	VOCs and CO	50–5000
7	TGS825	H_2S	5–100
8	TGS4161	CO_2	350–10,000
9	TGS826	NH_3	30–300
10	TGS2201	NO and NO_2	0.1–10
11	TGS822	VOCs	50–5000
12	TGS821	H_2	10–1000

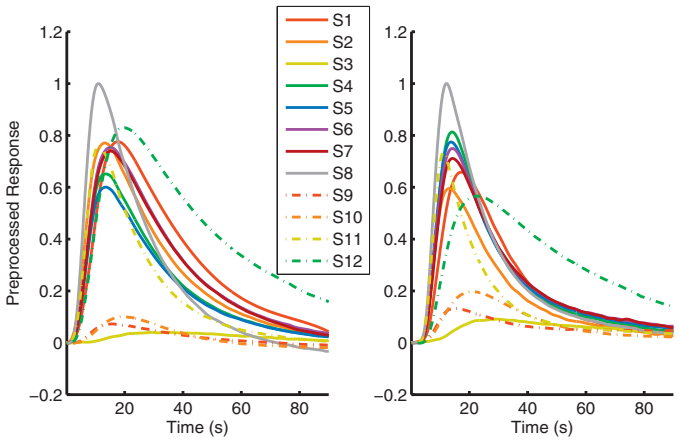

Fig. 5. Typical samples in dataset 2. Left: healthy; right: airway inflammation. The sensor models are in Table 4.

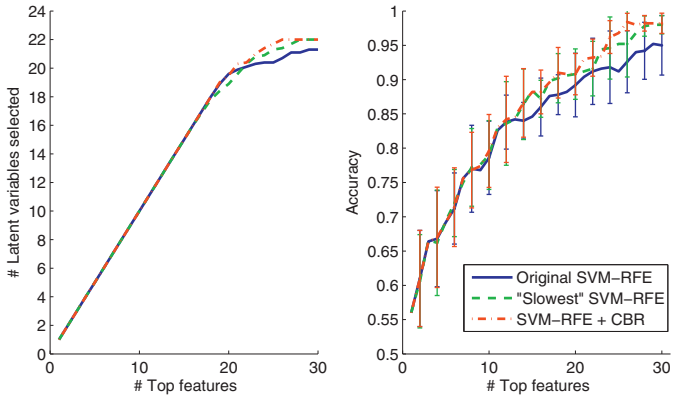

Fig. 6. FS results on the synthetic dataset. Left: the number of latent variables identified in the top-ranked features. Right: average classification accuracy of the top-ranked features. The error bars represent the standard deviations.

are similar to the ones for ordinary sensors in dataset 1, but not as many since the sample duration of dataset 2 is shorter.

5. Results and discussion

The performance of the proposed method (without ensemble) on the three datasets will be described and compared in Sections 5.1–5.4. Section 5.5 will discuss the stability of the FS methods and investigate the performance of SVM-RFE + CBR with ensemble. Some useful information will be provided in Section 5.6 by analyzing the feature importance.

5.1. Synthetic dataset

The synthetic dataset has been described in Section 3.1. It contains several groups of highly correlated features. A 10-fold cross-validation was conducted on the dataset to evaluate the algorithms. First, feature ranking was performed on the training sets. Then, linear SVM classifiers based on the top-ranked features were used to classify separate test sets. Finally, the average classification accuracy and the standard deviation are calculated. Because the relationship between the features and the class label is linear, we adopted linear SVM-RFE to rank the features. The penalty parameter C of the SVM models was empirically set to 2^3 for both SVM-RFE and classification. For the RFE procedure, the elimination number threshold is $T_e = 22$. For the CBR strategy, the group size threshold is $T_g = 1$; the correlation threshold is $T_c = 0.9$.

In Fig. 6, three algorithms are compared. Besides the linear SVM-RFE with or without CBR, the "slowest" SVM-RFE is also explored, which removes the features one by one in the RFE procedure (equivalent to setting $T_e = \infty$). Theoretically, this method is not affected by correlation bias. The left figure shows how many latent variables have been included in the top-ranked features. Recall that the 80 relevant features in the dataset are generated from 22 latent variables. We can see that the original SVM-RFE fails to include all the variables in the top 30 features, probably because some of the feature groups are eliminated too early due to CB. Both SVM-RFE + CBR and the slowest SVM-RFE succeed to include all the variables in the top 30 features. The right figure compares the accuracy of the three algorithms. This result shows that SVM-RFE + CBR is comparable to the slowest SVM-RFE and better than the original SVM-RFE in the synthetic dataset. It proves the ability of the CBR strategy to reduce the influence of CB. Besides, the slowest SVM-RFE becomes impractical to use when the feature dimension is high. When running the experiment on dataset 1 using Matlab, SVM-RFE + CBR needed 100s to rank the 1712 features in one cross-validation, while the slowest

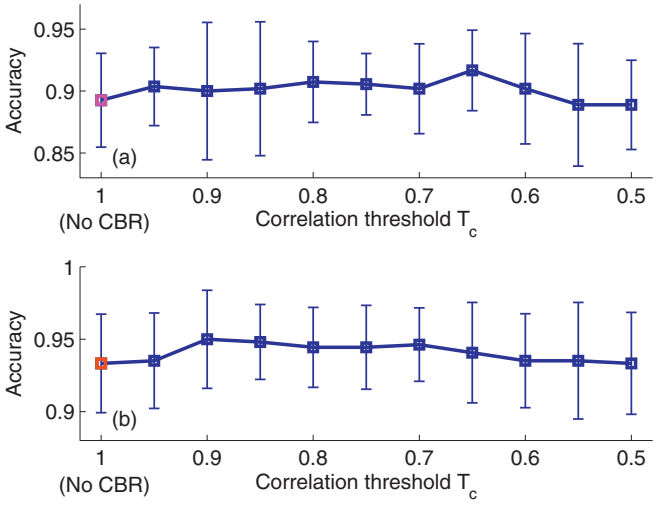

Fig. 7. Average accuracy of (a) linear and (b) nonlinear SVM-RFE with CBR in dataset 1 with varying correlation threshold T_c. When $T_c = 1$, the algorithm is equivalent to the original SVM-RFE without CBR. The error bars represent the standard deviations.

SVM-RFE did not finish it in 9h. So we will not compare the result of the latter method in the breath analysis datasets.

5.2. Dataset 1

Similar to the synthetic dataset, 10-fold cross-validation was carried out for dataset 1. Gaussian SVM was adopted for classification with parameters $C = 2^3$ and $\gamma = 2^{-6}$. In both linear and nonlinear SVM-RFE algorithms, we set $C = 2^3$ since it has good performance. The kernel parameter γ was searched among $\{2^{-3}, 2^{-4}, \ldots, 2^{-10}\}$ for nonlinear SVM-RFE with or without CBR. Finally we found that the best accuracy is achieved in both situations when $\gamma = 2^{-8}$. Other parameters for RFE and CBR were: $T_e = 60$, $T_g = 2$.

In Fig. 7, performance of linear and nonlinear SVM-RFE are compared. The shown performance is the best accuracy among the top 60 feature subsets. The x-axis of the figure is the correlation threshold T_c in CBR strategy. When $T_c = 1$, the algorithm is equivalent to the original SVM-RFE without CBR. Table 5 shows the classification accuracy of several feature extraction/selection methods. The parameters of the methods have been optimized. In the principal component analysis (PCA) method, the ratio of variance [18] was searched from 80% to 99.9%. In the min-redundancy max-relevance (mRMR) method, the features were discretized to three levels to compute the mutual information [31]. When hierarchical

Table 5
Performance comparison of various methods in dataset 1. The stability is the average Jaccard index.

Algorithm	#Features	Sensitivity (%)	Specificity (%)	Average accuracy (%)	Stability
All transient features	1712	87.04 ± 6.36	90.74 ± 5.59	88.89 ± 4.86	–
PCA + transient features	58	90.74 ± 5.01	90.00 ± 4.95	90.37 ± 2.44	–
PCA + magnitude [18]	40	92.59 ± 4.62	90.74 ± 6.11	91.67 ± 4.56	–
mRMR [31]	40	90.37 ± 4.68	91.85 ± 4.55	91.11 ± 3.24	**0.5183**
Sequential forward selection [3]	28	83.70 ± 6.10	88.15 ± 10.88	85.93 ± 6.25	0.1088
Original linear SVM-RFE	57	90.37 ± 7.03	88.15 ± 4.20	89.26 ± 3.79	0.2201
Linear SVM-RFE + CBR (T_c = 0.65)	**17**	90.74 ± 7.66	92.60 ± 3.51	91.67 ± 3.24	0.2238
Original nonlinear SVM-RFE	30	93.33 ± 5.74	93.33 ± 4.20	93.33 ± 3.40	0.4996
Nonlinear SVM-RFE + HC	36	94.07 ± 5.30	94.07 ± 4.95	94.07 ± 3.50	0.4228
Nonlinear SVM-RFE + CBR (T_c = 0.9)	31	**94.44 ± 6.11**	**95.56 ± 3.40**	**95.00 ± 3.39**	0.4572

Bold values indicate best results.

clustering (HC) was applied before SVM-RFE, the number of clusters was searched between 800 and 1700. In clinical applications, sensitivity and specificity measures are important, so they are also listed in Table 5. Sensitivity is the proportion of correctly classified patients in all the patients, while specificity is the proportion of correctly classified healthy subjects in all the healthy subjects. The accuracy is presented as "mean ± standard deviation".

5.3. Dataset 2

There are three subproblems in dataset 2: discriminating healthy samples from diabetes, renal disease, and airway inflammation samples, respectively. 10-Fold cross-validation was carried out for each problem. The experimental configurations were similar to those in Section 5.2. The kernel parameter was also separately tuned in nonlinear SVM-RFE with or without CBR. The value of T_c was searched among 0.5, 0.55, ..., 0.95 for better accuracy. The results are shown in Tables 6–8.

5.4. Discussion

Fig. 7 shows that the CBR strategy improves the average accuracy of both linear and nonlinear SVM-RFE when T_c is not less than 0.6. The improvement seems not very obvious because the standard deviation (SD) is relatively large. However, when observing the detailed results of datasets 1 and 2 (Tables 5–8), we find that SVM-RFE + CBR has comparable or lower SD than the original SVM-RFE. Moreover, the sensitivity, specificity, and average accuracy of SVM-RFE + CBR are consistently better than that of SVM-RFE. So the CBR strategy is effective in terms of accuracy. The SD of other feature extraction/selection methods such as PCA, SFS, and mRMR is comparable to SVM-RFE, which is probably caused by the fact that the number of training samples is limited. We also find that the accuracy of nonlinear SVM-RFE is always better than the linear one, which is because of the nonlinear nature of the data. The best T_c value varies between 0.65 and 0.9 depending on the dataset and the algorithm (linear or nonlinear).

In Table 5, when all the transient features are used, the accuracy is not very good. Although it contains useful features for classification, the transient feature set also contains irrelevant and redundant features, which will hinder the training of the classifier. PCA can reduce the redundancy within features, so the accuracy is improved in the method "PCA + transient features". In the "PCA + magnitude" method, PCA is applied to the magnitude of the whole curve as presented in [18]. Its accuracy is even better, possibly because there are less irrelevant features in the magnitude.

A proper FS method can be used to identify the effective features and discard the irrelevant and redundant ones. We compared a few FS methods that are able to handle redundancy in candidate features. The mRMR method proposed by Peng et al. [31] is a popular filter method which selects relevant and nonredundant features based on a mutual information criteria. The widely-used sequential forward selection (SFS) algorithm [3] iteratively examines each feature and selects the one that maximizes the cross-validation accuracy in the training set. If there is redundancy in features, the time cost of SFS will increase, but the accuracy will not be affected. The drawback of the two methods is that they use greedy strategies, thus are prone to be trapped in local optima. As a wrapper method, SFS often overfits the training set, especially when the sample size is much smaller than the feature dimension, which results in a low accuracy in Table 5. Additionally, it is often impractical to use wrapper methods such as SFS and genetic algorithm on high dimensional FS problems due to the large time cost.

Besides the proposed CBR strategy, there is an alternative way to deal with the correlation bias problem in SVM-RFE. The intuitive idea is to filter the redundant features before FS. Following [21], we implemented a method which uses hierarchical clustering to group the correlated features. The feature closest to the group center is kept in each group. Then the filtered features (about 1500 in dataset 1) are ranked using nonlinear SVM-RFE. Table 5 shows that this method generates a higher accuracy (94.07%) than the original nonlinear SVM-RFE (93.33%). However, the proposed nonlinear SVM-RFE + CBR achieves the best accuracy 95.00% with fewer features.

5.5. Stability analysis and the ensemble method

The stability of the FS algorithms listed in Tables 5–8 is the average Jaccard index (see Section 3.4) of the top 60 features. The overall stability is not high, which is mainly because there are many highly correlated features, hence the same accuracy can be achieved by different feature subsets. In Table 5, mRMR achieves the highest stability. The other algorithms all depend on the training of the SVM classifier, thus will be more sensitive to the perturbation of the training set. This result is consistent with [27], where SVM-RFE was found to be less stable than univariate filter methods. The stability of nonlinear SVM-RFE is better than the linear one due to the nonlinear nature of the data. The stability of SVM-RFE + CBR is slightly lower than SVM-RFE. The possible reason is that the CBR strategy moves several features back to the surviving feature list in each RFE iteration, which increases the uncertainty of the algorithm, since deciding which feature to move can be sensitive to sample perturbation if several features are highly correlated and have similar ranking criteria.

In order to improve the stability of SVM-RFE + CBR, the ensemble method introduced in Section 3.4 was investigated. Fig. 8 shows the accuracy and stability of the ensemble method as the ensemble size (the number of ranking processes to be aggregated) changes. The results were obtained by 10-fold cross-validation followed by averaging over 10 repetitions. It can be seen that as the ensemble size increases, the average accuracy and standard deviation do

Table 6
Performance comparison of various SVM-RFE strategies in dataset 2: distinguishing between healthy and diabetes samples. The stability is the average Jaccard index.

Algorithm	#Features	Sensitivity (%)	Specificity (%)	Average accuracy (%)	Stability
Linear	59	90.00 ± 8.15	90.00 ± 8.92	90.00 ± 6.07	0.0531
Linear + CBR (T_c = 0.7)	**43**	90.00 ± 7.30	90.77 ± 8.73	90.38 ± 6.35	0.0525
Nonlinear	52	97.69 ± 3.72	**99.23 ± 2.43**	98.46 ± 2.69	**0.4131**
Nonlinear + CBR (T_c = 0.9)	56	**99.23 ± 2.43**	**99.23 ± 2.43**	**99.23 ± 1.62**	0.4087

Bold values indicate best results.

Table 7
Performance comparison of various SVM-RFE strategies in dataset 2: distinguishing between healthy and renal disease samples. The stability is the average Jaccard index.

Algorithm	#Features	Sensitivity (%)	Specificity (%)	Average accuracy (%)	Stability
Linear	60	92.31 ± 8.11	86.92 ± 8.92	89.62 ± 5.75	0.0494
Linear + CBR (T_c = 0.7)	50	92.31 ± 5.13	90.00 ± 7.30	91.15 ± 5.14	0.0490
Nonlinear	40	96.92 ± 7.43	97.96 ± 5.19	97.31 ± 4.23	**0.4077**
Nonlinear + CBR (T_c = 0.85)	**32**	**98.46 ± 3.24**	**98.46 ± 3.24**	**98.46 ± 1.99**	0.3510

Bold values indicate best results.

Table 8
Performance comparison of various SVM-RFE strategies in dataset 2: distinguishing between healthy and airway inflammation samples. The stability is the average Jaccard index.

Algorithm	#Features	Sensitivity (%)	Specificity (%)	Average accuracy (%)	Stability
Linear	54	83.33 ± 11.11	78.33 ± 11.25	80.83 ± 7.14	0.0473
Linear + CBR (T_c = 0.8)	52	86.67 ± 8.96	79.17 ± 10.58	82.92 ± 7.47	0.0453
Nonlinear	**45**	93.33 ± 8.29	92.50 ± 7.30	92.92 ± 5.49	**0.4844**
Nonlinear + CBR (T_c = 0.8)	**45**	**95.00 ± 5.83**	**93.33 ± 5.27**	**94.17 ± 2.91**	0.4322

Bold values indicate best results.

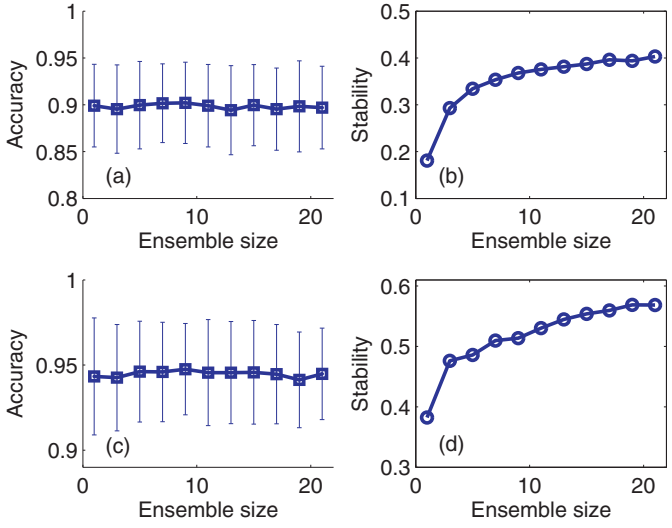

Fig. 8. Average accuracy and stability of SVM-RFE + CBR with ensemble in dataset 1. Plots (a) and (b) correspond to linear SVM-RFE + CBR with T_c = 0.65. Plots (c) and (d) correspond to nonlinear SVM-RFE + CBR with T_c = 0.9. The error bars represent the standard deviations.

not change obviously, but the stability has significant improvement. When the ensemble size is greater than 9, the stability of nonlinear SVM-RFE + CBR is better than mRMR. So the ensemble method is able to improve the stability at the cost of more computation time. It is worth noting that higher accuracy can be coupled with relatively lower stability particularly in the presence of highly correlated features [27]. In FS applications where stability and accuracy are both important, ensemble methods can be considered.

5.6. Analysis of the ranked features

FS techniques can help us understand the data better. By analyzing the importance of the features, useful information about the sensors and feature extraction algorithms can be obtained. Dataset 1 is studied since it contains TM sensors. We wish to find the answers to several questions: In the application of diabetes identification, which sensors are important? What kinds of features are suitable for ordinary/TM sensors? What heating voltage is a better choice for TM sensors?

The output of SVM-RFE + CBR is a ranked feature list. The ranking of a feature indicates its importance. However, a sensor or a type of feature (such as the magnitude feature) is made up of a group of features. Their importance needs to be estimated according to a group of rankings. Simply averaging the rankings may be improper, because we are more interested in whether the feature group contains useful features. The irrelevant features in the group may degrade its ranking and mislead our judgment. As a result, we use the "average top rank" criterion, namely the average of the top 5 rankings of the features in the group, to evaluate the feature groups. The smaller the criterion, the more important the feature group. Note that the ten ranking lists of the 10-fold cross-validation are pooled together. Using this criterion, some helpful conclusions can be summarized:

- **Sensors**. The importance order of the sensors is TGS2600-TM, TGS2602-TM, TGS2610-D00, TGS826, WSP2111-TM, TGS4161, GSBT11, SP3S-AQ2, TGS822, and WSP2111. The two most important sensors are both TM sensors. The models of WSP2111-TM and WSP2111 are the same, but the former one is operated under TM, which makes it turn from the least important sensor to the 5th one. To the best of our knowledge, [18] is the first literature that applied TM technique to breath analysis systems. The results prove the effectiveness of TM in such applications. The carbon dioxide sensor (TGS4161) has an average top rank of 11.6, showing that it is useful for the application.
- **Feature types**. The average top ranks of the seven types of transient features are displayed in Table 9. The phase feature extracted from TM sensors is the most effective. The time constant (especially the T_{max} feature) and derivative are effective for both types of sensors, while the 2nd derivative and integral

Table 9
Average top ranks of the seven types of transient features extracted from ordinary and TM sensors, respectively. The smaller the better.

Feature type	Ordinary sensor	TM sensor
Magnitude	7.2	6.4
Difference	20.4	4.6
Derivative	6.2	2.6
2nd derivative	80.4	35.4
Integral	46.0	15.6
Time constant	**4.0**	2.6
Phase feature	11.8	**1.0**

Bold values indicate best results.

are the least effective ones. The normalized magnitudes show slightly smaller average top ranks than the magnitudes without normalization.

- **TM heating voltages**. We find that the average top rank is smaller when the heating voltage is in the interval of 6 V–7 V–3 V (see Fig. 2). It implies that when detecting breath biomarkers such as acetone, the TM sensors' responses are more discriminative when the temperature is close to or higher than normal range. According to [32], responses at low temperatures contained mostly redundant or indiscriminative information. This is consistent with our study. But it needs further investigation whether the high heating voltage will increase sensor drift.

6. Conclusion

In this paper, the linear and nonlinear support vector machine recursive feature elimination (SVM-RFE) algorithms were studied. The correlation bias problem in SVM-RFE was raised, which will affect the accuracy of the feature selection result. The correlation bias reduction (CBR) algorithm was proposed to solve the problem by improving the feature elimination strategy. A synthetic dataset and two breath analysis datasets with large sets of correlated features were used to evaluate the algorithms. The nonlinear SVM-RFE + CBR was proved to be effective. It outperformed the original SVM-RFE and other typical algorithms. A complete and efficient implementation of the proposed method was also presented. The stability of the proposed algorithm can be improved by applying the ensemble method.

In this study, the comprehensive feature set included seven types of transient features. By analyzing the features' rankings, some useful knowledge was obtained. Three representative conclusions for dataset 1 are:

- MOS sensors with temperature modulation (TM) are significantly more effective than those without TM.
- Phase feature is the best feature for TM sensors and time constant is the best for ordinary sensors.
- The TM sensors' responses are more discriminative when the temperature is close to or higher than the normal range.

These information will be helpful when making new breath analysis systems or designing new features. For example, the low-ranking sensors may be discarded to lower the cost without precision loss. Features related to TM sensors or the phase space can be further studied. In summary, the proposed FS algorithm is a promising method for accuracy enhancement, dimension reduction and data interpretation. Future works may include investigation of more kinds of features [33,34].

Acknowledgments

This work is partially supported by the Natural Science Foundation of China (NSFC) (Nos. 61332011, 61020106004, 61272292, 61271344, 61101150, 61401048), the GRF fund from the HKSAR Government, the central fund from Hong Kong Polytechnic University, The National Basic Research Program of China (973 Program: 2011CB505404), Shenzhen Fundamental Research fund (JCYJ20130401152508661), Shenzhen special fund for the Strategic Development of Emerging Industries (JCYJ20120831165730901), and Key Laboratory of Network Oriented Intelligent Computation, Shenzhen, China. The authors would like to thank Dr. Zhenhua Guo, Lei Zhang, and the anonymous reviewers for their valuable suggestions.

References

[1] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (2003) 1157–1182.
[2] Y. Saeys, I. Inza, P. Larrañaga, A review of feature selection techniques in bioinformatics, Bioinformatics 23 (2007) 2507–2517.
[3] N. Paulsson, E. Larsson, F. Winquist, Extraction and selection of parameters for evaluation of breath alcohol measurement with an electronic nose, Sens. Actuators A: Phys. 84 (2000) 187–197.
[4] E. Llobet, O. Gualdrón, M. Vinaixa, N. El-Barbri, J. Brezmes, X. Vilanova, B. Bouchikhi, R. Gomez, J. Carrasco, X. Correig, Efficient feature selection for mass spectrometry based electronic nose applications, Chemom. Intell. Lab. 85 (2007) 253–261.
[5] O. Gualdrón, J. Brezmes, E. Llobet, A. Amari, X. Vilanova, B. Bouchikhi, X. Correig, Variable selection for support vector machine based multisensor systems, Sens. Actuators B: Chem. 122 (2007) 259–268.
[6] M. Pardo, G. Sberveglieri, Random forests and nearest shrunken centroids for the classification of sensor array data, Sens. Actuators B: Chem. 131 (2008) 93–99.
[7] J.H. Cho, P.U. Kurup, Decision tree approach for classification and dimensionality reduction of electronic nose data, Sens. Actuators B: Chem. 160 (2011) 542–548.
[8] R. Kaur, R. Kumar, A. Gulati, C. Ghanshyam, P. Kapur, A.P. Bhondekar, Enhancing electronic nose performance: a novel feature selection approach using dynamic social impact theory and moving window time slicing for classification of Kangra orthodox black tea (*Camellia sinensis* (L.) o. kuntze), Sens. Actuators B: Chem. 166 (2012) 309–319.
[9] S. Marco, A. Gutiérrez-Gálvez, Signal and data processing for machine olfaction and chemical sensing: a review, IEEE Sens. J. 12 (2012) 3189–3214.
[10] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machines, Mach. Learn. 46 (2002) 389–422.
[11] A. Rakotomamonjy, Variable selection using SVM based criteria, J. Mach. Learn. Res. 3 (2003) 1357–1370.
[12] K.-B. Duan, J.C. Rajapakse, H. Wang, F. Azuaje, Multiple SVM-RFE for gene selection in cancer classification with expression data, IEEE Trans. NanoBiosci. 4 (2005) 228–234.
[13] Y. Tang, Y.-Q. Zhang, Z. Huang, Development of two-stage SVM-RFE gene selection strategy for microarray expression data analysis, IEEE ACM Trans. Comput. Biol. Bioinform. 4 (2007) 365–381.
[14] S. Yoon, S. Kim, Mutual information-based SVM-RFE for diagnostic classification of digitized mammograms, Pattern Recogn. Lett. 30 (2009) 1489–1495.
[15] P.A. Mundra, J.C. Rajapakse, SvM-RFE with MRMR filter for gene selection, IEEE Trans. NanoBiosci. 9 (2010) 31–37.
[16] L. Toloşi, T. Lengauer, Classification with correlated features: unreliability of feature ranking and solutions, Bioinformatics 27 (2011) 1986–1994.
[17] A. D'Amico, C. Di Natale, R. Paolesse, A. Macagnano, E. Martinelli, G. Pennazza, M. Santonico, M. Bernabei, C. Roscioni, G. Galluccio, Olfactory systems for medical applications, Sens. Actuators B: Chem. 130 (2008) 458–465.
[18] K. Yan, D. Zhang, D. Wu, H. Wei, G. Lu, Design of a breath analysis system for diabetes screening and blood glucose level prediction, IEEE Trans. Biomed. Eng. 61 (2014) 2787–2795.
[19] D. Guo, D. Zhang, N. Li, L. Zhang, J. Yang, A novel breath analysis system based on electronic olfaction, IEEE Trans. Biomed. Eng. 57 (2010) 2753–2763.
[20] C.J. Burges, A tutorial on support vector machines for pattern recognition, Data Min. Knowl. Discov. 2 (1998) 121–167.
[21] M.Y. Park, T. Hastie, R. Tibshirani, Averaged gene expressions for regression, Biostatistics 8 (2007) 212–227.
[22] D.B. Sharma, H.D. Bondell, H.H. Zhang, Consistent group identification and variable selection in regression with correlated predictors, J. Comput. Graph. Stat. 22 (2013) 319–340.
[23] A.P. Bhondekar, R. Kaur, R. Kumar, R. Vig, P. Kapur, A novel approach using dynamic social impact theory for optimization of impedance-tongue (itongue), Chemom. Intell. Lab. 109 (2011) 65–76.
[24] W. Awada, T.M. Khoshgoftaar, D. Dittman, R. Wald, A. Napolitano, A review of the stability of feature selection techniques for bioinformatics data, in: 2012 IEEE 13th Intl. Conf. on Information Reuse and Integration (IRI), IEEE, Las Vegas, USA, 2012, pp. 356–363.
[25] P. Somol, J. Novovicova, Evaluating stability and comparing output of feature selectors that optimize feature subset cardinality, IEEE Trans. Pattern Anal. Mach. Intell. 32 (2010) 1921–1939.

[26] Y. Saeys, T. Abeel, Y. Van de Peer, Robust feature selection using ensemble feature selection techniques, in: Machine Learning and Knowledge Discovery in Databases, Springer, 2008, pp. 313–325.
[27] A. Kalousis, J. Prados, M. Hilario, Stability of feature selection algorithms: a study on high-dimensional spaces, Knowl. Inf. Syst. 12 (2007) 95–116.
[28] T.D.C. Minh, D.R. Blake, P.R. Galassetti, The clinical potential of exhaled breath analysis for diabetes mellitus, Diabetes Res. Clin. Pract. 97 (2012) 195–205.
[29] K. Yan, D. Zhang, Sensor evaluation in a breath analysis system, in: 2014 International Conference on Medical Biometrics (ICMB), IEEE, 2014, pp. 35–40.
[30] A. Hierlemann, R. Gutierrez-Osuna, Higher-order chemical sensing, Chem. Rev. 108 (2008) 563–613.
[31] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (2005) 1226–1238.
[32] S. Hosseini-Golgoo, F. Hossein-Babaei, Assessing the diagnostic information in the response patterns of a temperature-modulated tin oxide gas sensor, Meas. Sci. Technol. 22 (2011) 035201.
[33] S. Zhang, C. Xie, M. Hu, H. Li, Z. Bai, D. Zeng, An entire feature extraction method of metal oxide gas sensors, Sens. Actuators B: Chem. 132 (2008) 81–89.
[34] R. Gutierrez-Osuna, A. Gutierrez-Galvez, N. Powar, Transient response analysis for temperature-modulated chemoresistors, Sens. Actuators B: Chem. 93 (2003) 57–66.
[35] E. Martinelli, C. Falconi, A. D'Amico, C. Di Natale, Feature extraction of chemical sensors in phase space, Sens. Actuators B: Chem. 95 (2003) 132–139.

Biographies

Ke Yan received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China. He is currently working toward the Ph.D. degree with the Department of Electronic Engineering, Graduate School at Shenzhen, Tsinghua University, Shenzhen, China. His research interests include biomedical engineering, pattern recognition for artificial olfaction, and machine learning.

David Zhang (F'09) graduated in Computer Science from Peking University. He received his M.Sc. in Computer Science in 1982 and his Ph.D. in 1985 from the Harbin Institute of Technology (HIT). From 1986 to 1988 he was a postdoctoral fellow at Tsinghua University and then an associate professor at the Academia Sinica, Beijing. In 1994 he received his second Ph.D. in Electrical and Computer Engineering from the University of Waterloo, Ontario, Canada. Currently, he is a Chair professor at the Hong Kong Polytechnic University where he is the Founding Director of the Biometrics Technology Centre (UGC/CRC) supported by the Hong Kong SAR Government in 1998. He also serves as Visiting Chair professor in Tsinghua University, and Adjunct professor in Peking University, Shanghai Jiao Tong University, HIT, and the University of Waterloo. He is the Founder and Editor-in-Chief, International Journal of Image and Graphics (IJIG); Book Editor, Springer International Series on Biometrics (KISB); Organizer, the International Conference on Biometrics Authentication (ICBA); Associate Editor of more than ten international journals including IEEE Transactions and Pattern Recognition; and the author of more than 10 books and 200 journal papers. Professor Zhang is a Croucher senior research fellow, Distinguished Speaker of the IEEE Computer Society, and a fellow of both IEEE and IAPR.