Computers in Biology and Medicine 142 (2022) 105208

Contents lists available at ScienceDirect

Computers in Biology and Medicine

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

Ensemble feature selection for stable biomarker identification and cancer classification from microarray expression data

Aiguo Wang [a,*], Huancheng Liu [a], Jing Yang [b], Guilin Chen [c]

[a] School of Electronic Information Engineering, Foshan University, Foshan, China
[b] School of Computer Science and Information Engineering, Hefei University of Technology, Hefei, China
[c] School of Computer and Information Engineering, Chuzhou University, Chuzhou, China

ARTICLE INFO

Keywords:
Feature selection
Gene expression profiles
Stability
Ensemble learning

ABSTRACT

Microarray technology facilitates the simultaneous measurement of expression of tens of thousands of genes and enables us to study cancers and tumors at the molecular level. Because microarray data are typically characterized by small sample size and high dimensionality, accurate and stable feature selection is thus of fundamental importance to the diagnostic accuracy and deep understanding of disease mechanism. Hence, we in this study present an ensemble feature selection framework to improve the discrimination and stability of finally selected features. Specifically, we utilize sampling techniques to obtain multiple sampled datasets, from each of which we use a base feature selector to select a subset of features. Afterwards, we develop two aggregation strategies to combine multiple feature subsets into one set. Finally, comparative experiments are conducted on four publicly available microarray datasets covering both binary and multi-class cases in terms of classification accuracy and three stability metrics. Results show that the proposed method obtains better stability scores and achieves comparable to and even better classification performance than its competitors.

1. Introduction

The rapid development of microarray technology in the post-genome era enables us to simultaneous measure the expression profiles of thousands of genes [1–3]. However, due to the nature of microarray experiments and limitation of realistic conditions, the obtained gene expression profiles are characterized by small sample size and high dimensionality, which inevitably poses a huge challenge to downstream analysis tasks such as biomarker identification, cancer diagnosis, and tumor subtype differentiation [4–7]. One feasible way is to reduce the dimensionality with an effective feature selection algorithm [8,9].

Feature selection or variable selection, also called gene selection in the context of microarray data, aims to identify a subset of discriminating features by keeping highly-relevant features and filtering out irrelevant and redundant features from the original feature space [10,11]. Different from feature extraction method that generates new features, which are a linear/nonlinear combination of original features [12], feature selection method selects a fraction of them and thus has better explanations. Effective feature selection methods help to identify potential biomarkers for further research of disease genes and drug targets and to train a powerful classifier for tumor subtype classification and cancer diagnosis [13,14]. According to whether a classification model is used to evaluate the goodness of candidate features, we broadly categorize existing methods into filter, wrapper, embedded, and hybrid methods [10,15,16]. Filter method is independent of a classification model and uses some metrics (such as distance metric, information theoretic metric, consistency metric, and dependency metric) to measure the importance of each feature [17], while wrapper method is coupled with a classifier and uses the classification accuracy to indicate the goodness of candidate features [18]. Compared with filter method, wrapper method generally achieves better prediction accuracy, but at the cost of higher time complexity [19]. Embedded method is basically a subclass of wrapper method and selects a subset of features after training the classifier. Decision tree and Lasso regression algorithm are two representatives [20]. Hybrid method utilizes the advantages of both filter and wrapper methods [21,22], and one common scheme is to first use a filter method to reduce the dimensionality and then use a wrapper method to optimize the reduced feature space. According to the outcome of feature selection methods, we can also group them into *feature ranking method* and *feature subset selection method*. The former returns a ranked list of original features and requires a further step to determine the number of finally selected features, while the latter outputs a subset of

* Corresponding author.Guangyun Road No. 33, Nanhai District, Foshan University, Foshan, 528225, China.
E-mail addresses: wangaiguo2546@163.com (A. Wang), liuhuancheng83@163.com (H. Liu), jsjyj0801@163.com (J. Yang), glchen@chzu.edu.cn (G. Chen).

https://doi.org/10.1016/j.compbiomed.2021.105208
Received 29 November 2021; Received in revised form 19 December 2021; Accepted 31 December 2021
Available online 5 January 2022

features [7].

For microarray data analysis, besides the selection of discriminating genes and construction of an accurate classifier, feature selection stability is another critical topic, which refers to the ability of a feature selection algorithm to select the same or similar features with the change of microarray data [23,24]. Stable feature selection methods help obtain a reliable feature subset and further improve the interpretability. In contrast, a feature selector with poor stability would inevitably reduce the confidence of biomedical and bioinformatics researchers in applying it to identify potential biomarkers, especially when the costs of biological verification experiments are high [25,26]. This hinders the acceptance and application of a feature selection algorithm. Hence, how to develop a stable and accurate feature selection algorithm remains crucial and meaningful for practical omics applications.

Feature selection stability is a relatively complex problem and the influencing factors mainly come from data level (e.g., high dimensional, small-sample-size, and noisy data), algorithm level (e.g., a feature selector is sensitive to the initial values of hyperparameters), and specific application domains (e.g., multiple genes having similar expression profiles and biology functions exist) [27–29]. Accordingly, researchers have been paying attention to the stability problem associated with feature selection and conducting researches in stability metrics and algorithm design [30,31]. According to the main idea of existing stability improvement methods, we broadly categorize them into three groups: sample weighting method, group feature selection method, and ensemble feature selection method [32,33].

The basic idea of sample weighting method is to assign different weights to each sample based on the its impact on the feature-class correlation and then conduct feature selection on the weighted samples [26]. The weighting strategy is to increase the weights of important samples and decrease the weights of unimportant samples. For example, Yu et al. proposed a margin-based sample weighting method and performed feature selection on the weighted samples to improve its robustness to the change of data [26]. Experimental results on four microarray datasets demonstrate its effectiveness. Although enhanced performance is achieved, it is not easy to choose an appropriate distance metric and weighting strategy. Considering that there are genes having similar biological functions or expression levels, group feature selection method divides genes into different clusters and then performs feature selection on the clusters [29]. This method mainly consists of two steps: grouping and selecting, where the former divides genes into different clusters in a data-driven or knowledge-driven way (e.g., via a clustering algorithm or biological knowledge about gene functions) and the latter uses a certain criterion to choose representative genes from each cluster to form the finally selected features. For example, Yu et al. utilized the kernel density estimation technique to find dense feature groups and then selected a representative feature from the chosen groups that are highly relevant to the class [29].

The main idea of ensemble feature selection method, motivated by ensemble learning paradigm, is to first obtain multiple feature subsets by performing feature selection multiple times and then combine the subsets into one set via a predefined aggregation strategy [4,30]. According to the way of generating an ensemble model, we divide existing ensemble feature selection methods into two groups: *data perturbation* method and *function perturbation* method [31]. For data perturbation, sampling techniques are used to generate multiple sampled datasets from the original data, then a base feature selection algorithm is performed on each of the sampled datasets to obtain multiple subsets of selected features, and finally the returned subsets are combined into one set. For example, Abeel et al. utilized the random sampling to generate sampled datasets and used the base feature selection algorithm SVM-RFE to build an ensemble feature selector [25]. They developed a weighted average method to rank features and chose the top-ranked ones to get the finally selected features. Experimental results demonstrated the effectiveness over its competitors. Function perturbation method uses multiple homogeneous/heterogeneous feature selection algorithms on the training data to get multiple feature subsets and then combines the subsets with an aggregator [13]. For example, Yang et al. proposed a feature selection method based on multi-criterion fusion, where they use multiple heterogeneous feature selectors to improve the stability [34].

Although ensemble feature selection method achieves promising results, however, it suffers from practical limitations [33]. For example, for function perturbation method, a group of synergistic feature selectors is usually determined experimentally and there is still a lack of theoretical guidance. Second, most of existing studies take a feature ranking algorithm as the base feature selector, which requires users not only to decide the number of selected features but also to decide the order of *thresholding* (i.e., determining the number of features to select from a ranked list) and *aggregating* (i.e., combining multiple sets into one set). Particularly, it is not trivial to determine the optimal subset size for feature ranking methods and also difficult to assign appropriate and reasonable weights to ranked features in aggregating multiple sets. Besides, the correlation among features and correlation between features and class are easily ignored. These motivate us to develop a method to automatically return a stable subset of features. To this end, we here propose an ensemble feature selection framework. It works under the ensemble learning paradigm and helps obtain a stable and accurate feature subset, as shown in preliminary experimental results [27]. Specifically, we instantiate the framework with a base feature algorithm considering relevance and redundancy and two developed aggregators to show its effectiveness. The main contributions of this study include the followings. (1) We propose an ensemble feature selection framework that can take as its building blocks various base feature selection algorithms and aggregation strategies. We here instantiate the framework with a feature subset selection algorithm to optimize the feature space and to relieve user from determining the number of finally selected features. (2) We present two aggregation strategies to combine multiple feature subsets into one set. Since the proposed aggregators utilize occurrence frequency-based criterion, they potentially help select stable features. (3) We conducted comparative experiments on microarray datasets, covering both binary and multi-class problems, with four classification models and seven competitive feature selection methods. Results demonstrate the effectiveness of the proposed method over its competitors in terms of classification accuracy and stability. We also adopt three different stability metrics to measure the stability towards an unbiased comparison.

The rest of this paper is organized as follows. Section 2 briefly reviews experimental datasets, introduce the proposed ensemble feature selection framework, develop two different aggregation strategies, and present the feature selection and validation scheme. Experimental setup and results are shown in section 3, followed by the discussion and conclusion sections.

2. Materials and methods

In this section, we first introduce the experimental datasets and then detail the proposed ensemble feature selection framework. For illustration purpose, we use $\mathbf{X} \in \mathbb{R}^{m*n}$ to denote the data matrix that has n genes and m samples and use $F = \{f_1, f_2, \ldots, f_n\}$ to represent the n genes. Given a label set $L = \{l_1, l_2, \ldots, l_C\}$ of a microarray dataset, where C is the number of classes, we use $Y = \{y_1, y_2, \ldots, y_m\}$ $(y_i \in L,\ 1 \le i \le m)$ to denote the labels of m samples. The labeled training data can thus be noted as $Data = \{(\mathbf{x}_i, y_i) | \mathbf{x}_i \in \mathbf{X}, y_i \in L, 1 \le i \le m\}$. For a test sample x with true label l, the task of a classifier trained on *Data* is to infer the label of x.

2.1. Datasets

We conduct experiments on four publicly available microarray datasets, including *SRBCT* [3], *Colon* [35], *DLBCL* [36], and *Leukemia* [37], to validate the power of a feature selector in selecting

discriminating features from gene expression profiles. The datasets cover both binary and multi-class classification problems and are featured with high-dimension and small-sample-size. Such a setting poses a great challenge for gene selection and helps evaluate a feature selector objectively and comprehensively. Table 1 presents a brief summary of them, of which the first three columns denote the number of genes, the number of samples, and the number of classes, respectively, and the fourth column #SGR refers to the ratio between the number of samples and the number of genes associated with a microarray dataset.

Small Round Blue Cell Tumor (*SRBCT*): Small round blue cell tumors belong to the malignant neoplasms and have a characteristic appearance of small round cells on routine histology. These tumors are commonly seen in children. *SRBCT* concerns four childhood tumors, including neuroblastoma (NB), rhabdomyosarcoma (RMS), non-Hodgkin Burkitt's lymphoma (BL), and Ewing's family of tumors (EWS). There are 18 NB, 25 RMS, 11 BL, and 29 EWS samples, and each sample contains the expression profiles of 2308 genes. The purpose of this study is to train a classification model for classifying small round blue cell tumors to specific diagnostic categories based on the gene expression profiles.

Colon: It contains a broad picture of the expression profiles of 2000 genes. There are 62 samples obtained with Affymetrix Oligonucleotide Arrays, of which 40 samples are labeled as colon tumor and 22 samples are from normal colon tissue. The #SGR is 0.031. The task on this dataset is to select informative genes from the gene space and then build a classifier with the selected genes to distinguish between normal and colon tumor subjects based on the microarray data.

Diffuse Large-B-Cell Lymphoma (*DLBCL*): The diffuse large B-cell lymphoma is a cancer of B cells and belongs to one form of non-Hodgkin lymphoma. The dataset has a collection of 19 follicular lymphomas (FL) samples and 58 diffuse large B-cell lymphomas (BCL) samples. Each sample contains the expression profiles of 7129 genes, which leads to a #SGR of 0.011. FL and BCL are two subtypes of B-cell lineage malignancies. The goal is to train a classification model to classify the two tumors.

Leukemia: Leukemia is a group of blood cancers and usually results in abnormal blood cells. The dataset is collected from the bone marrow and peripheral blood of leukemia patients, concerned with acute myeloid leukemia (*AML*), B-cell acute lymphoma leukemia (ALL-B), and T-cell acute lymphoma leukemia (ALL-T). ALL-B and ALL-T are the two subtypes of acute lymphoma leukemia. There are 25 AML samples, 9 ALL-T samples, and 38 ALL-B samples, and each sample is encoded by 5327 genes. We aim to build a classifier to distinguish the three tumors.

2.2. Ensemble feature selection framework

Fig. 1 shows the proposed ensemble feature selection framework. It mainly consists of three steps. The first step is sampling, where our framework applies sampling techniques on the training data *Data* to obtain M datasets $\{D_1, D_2, \ldots, D_M\}$ at the instance level, rather than simply perform feature selection on *Data*. Commonly used sampling techniques include, but not limited to, Bootstrapping, M-fold cross-validation, over-sampling with/without replacement, and under-sampling with/without replacement. Afterwards, we use a base feature selector FS_i $(1 \le i \le M)$ on each of the M datasets to get M subsets of features. Finally, we design an aggregation strategy to combine the M feature sets to get the finally selected features.

Table 1
Description of experimental datasets.

Dataset	#Genes	#Samples	#Classes	#SGR	Reference
SRBCT	2308	83 (29/25/11/18)	4	0.036	Khan et al. [3]
Colon	2000	62 (40/22)	2	0.031	Alon et al. [35]
DLBCL	7129	77 (58/19)	2	0.011	Shipp et al. [36]
Leukemia	5327	72 (38/9/25)	3	0.014	Golub et al. [37]

Obviously, base feature selector and aggregator are two crucial components of the framework. As for the choice of M base feature selectors, we can use the same or different feature selection methods on the M datasets, which we call *homogeneous scheme* and *heterogeneous scheme*, respectively. As for the aggregator, different aggregation strategies are used according to the results returned by FS_i. Specifically, if FS_i is a feature subset method that outputs a subset of features S_i $(1 \le i \le M)$ of original features, the aggregation strategy can directly work on $\{S_1, S_2, \ldots, S_M\}$ to get the finally selected features S. If FS_i is a feature ranking method that returns a ranked list R_i $(1 \le i \le M)$ of original features in descending order according to the importance of each feature, there are generally two strategies to combine multiple feature rankings. The first scheme is called *aggregation and thresholding*, which involves two steps. 1) It keeps all the features in R_i $(1 \le i \le M)$ and re-ranks the n features $\{f_1, f_2, \ldots, f_n\}$ according to the statistics of $\{R_1, R_2, \ldots, R_M\}$ to gets a ranked list R of original features. For example, we can use the average rank, weighted average rank, or minimal rank of each feature f in $\{R_1, R_2, \ldots, R_M\}$ to weight the importance of f. 2) It uses a thresholding method to decide how many features in R to be kept and then returns the selected feature subset S. There are thresholding methods available, such as selecting top-k or a fixed percentage of features. In contrast, the second scheme is called *thresholding and aggregation*. It first applies a thresholding method on R_i $(1 \le i \le M)$ to get a feature subset RS_i $(1 \le i \le M)$ and then combines $\{RS_1, RS_2, \ldots, RS_M\}$ to return the selected features S.

Accordingly, Algorithm 1 presents the pseudo-code of the ensemble feature selection method, where line 3 corresponds to the sampling step that gets M sampled datasets, line 4 is to use base feature selector for selecting features from each of the M datasets, and line 6 refers to the aggregation step. It is noteworthy that line 6 involves the procedures of aggregation and thresholding if a feature ranking method is used in line 4. In following subsections, we introduce a feature subset selection algorithm and two designed aggregators to instantiate the framework.

Algorithm 1. Pseudo-code of ensemble feature selection

	Algorithm 1. Pseudo-code of ensemble feature selection
Input	training data *Data*, the number of sampling M, base feature selector FS_i $(1 \le i \le M)$, aggregation strategy *aggregator*
Output	selected features S
1	$S = \{\}$; //initialization
2	**for** $i = 1$ **to** M
3	D_i = sampling(*Data*); //sampling techniques
4	$S_i = FS(D_i)$; //feature selection on D_i
5	**endfor**
6	$S = aggregator(S_1, S_2, \ldots, S_M)$; //merge M sets
7	**return** S

2.3. Base feature selector

Correlation-based feature selection (CFS) is a filtering-based feature subset selection method [38]. In contrast to feature ranking methods, CFS does not require users to specify the number of finally selected features. On the basis of the statistical theory, CFS considers feature-feature relevance and feature-class relevance in evaluating the goodness of a feature f rather than only evaluating f individually and adopts a heuristic strategy to filter out irrelevant and redundant features. Hence, we herein take CFS as the building block of the ensemble feature selection framework.

Given a dataset with feature set F and labels Y, CFS uses the best first search to search the feature space according to the evaluation metric as given in equation (1).

$$merit_S = \frac{p\overline{r}_{Yf}}{\sqrt{p + p(p-1)\overline{r}_{ff}}} \quad (1)$$

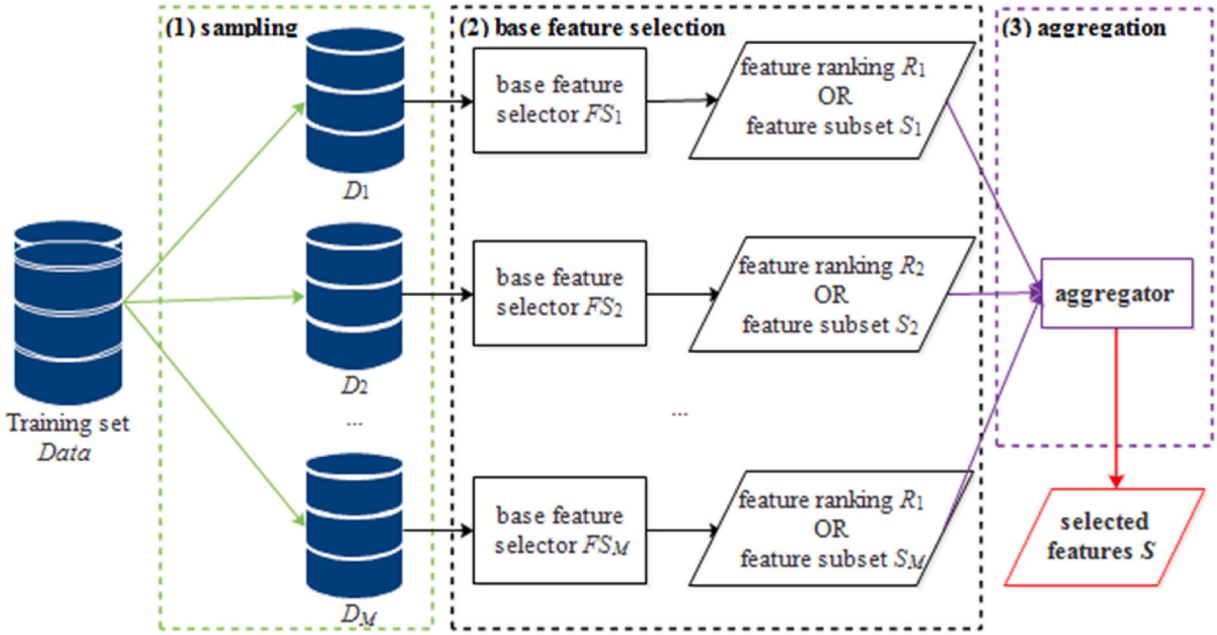

Fig. 1. Ensemble feature selection framework that mainly consists of three phases.

where $merit_S$ denotes the merit of feature subset S that contains p features, $\bar{r}_{Yf}$ is the average feature-class correlation between $f \in S$ and Y, and $\bar{r}_{ff}$ is the average feature-feature correlation of S. When $p = 1$, $merit_S$ equals the correlation between feature f and Y.

Specifically, CFS works as follows. 1) Initialize S to be empty. 2) Calculate the merit of each feature f in F according to equation (1), select f with the largest merit, add it to S, and delete it from F. 3) Select $f \in F$ with the largest merit, add it to S, and delete it from F; if a higher merit is not obtained, delete f from S and then select $f \in F$ with the largest merit; repeat this step until the stopping criterion is satisfied. CFS stops when F becomes empty or five consecutive searches of f fail to get a larger merit of S.

2.4. Aggregation strategy

The primary purpose of aggregation is to combine multiple feature subsets into one set. To improve the stability of finally selected features, we here take as the selection criterion the occurrence frequency of each feature f in multiple feature subsets. Specifically, given M sets $\{S_1, S_2, \ldots, S_M\}$ returned by M base feature selectors, suppose $U = \cup_{i=1}^{M} S_i$ and p_f denotes the frequency that $f \in U$ appears in the M sets, we add f to S, if p_f is not less than a predefined threshold γ,

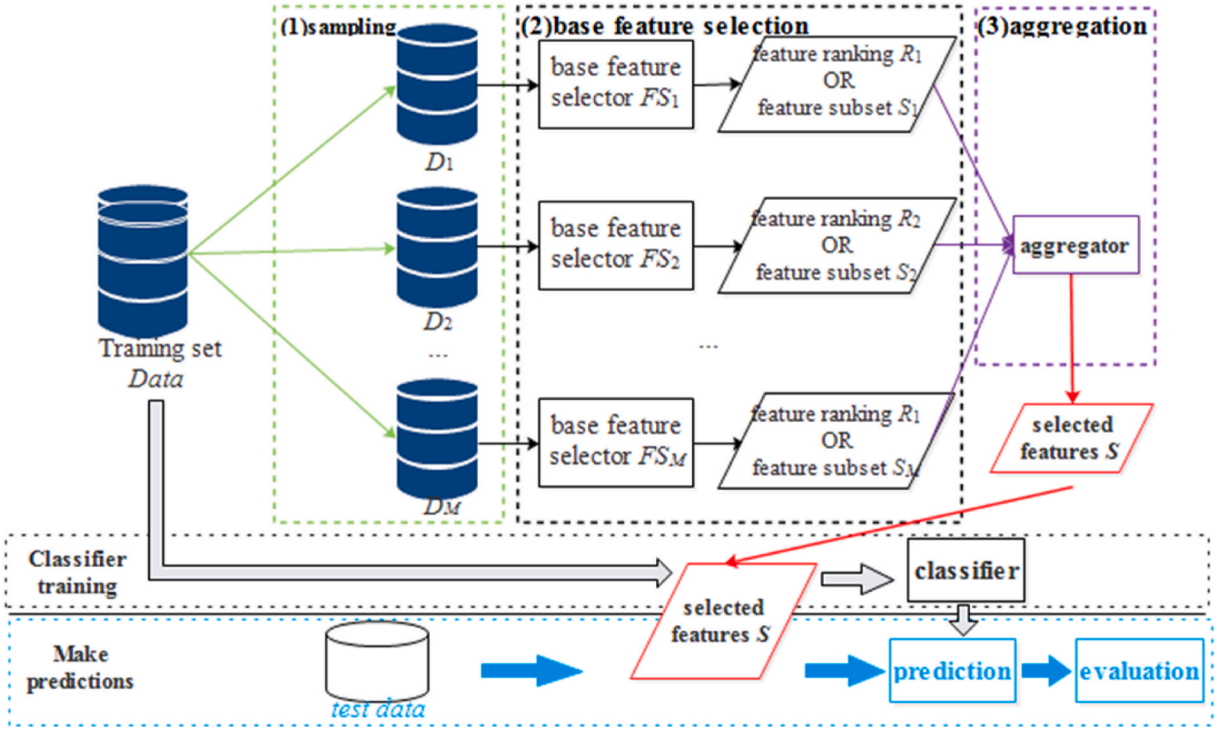

Fig. 2. Flowchart of feature selection and validation. It mainly involves the procedures of feature selection, classifier training, and prediciton.

$$S = \{f | p_f \geq \gamma, \ \ f \in U\} \tag{2}$$

where S keeps the selected features. Particularly, if $\gamma = 1$, $S = \cap_{i=1}^{M} T_i$ is the interaction of the M sets; if $\gamma = 0$, $S = \cup_{i=1}^{M} T_i$ contains the features that appear in at least one of the M sets.

Besides, we can apply CFS on S to further optimize the feature space and derive a two-stage strategy (3).

$$\begin{aligned} &S = CFS(S_1) \\ &s.t. \ \ S_1 = \{f | p_f \geq \gamma, \ \ f \in U\} \end{aligned} \tag{3}$$

2.5. Feature selection and validation

2.5.1. Flowchart

After selecting features from training data, we conduct a validation procedure by training a classifier with the selected features on the training data and testing its predictive power on a test set. Fig. 2 presents the corresponding framework, where the feature selection block aims to select a subset of features and we here use our proposed ensemble feature selection method. Notably, feature selection is only performed on the training set to avoid the selection bias problem and the dataset used to train the classifier is a projection of the training data over the selected features [39]. Classification performance metrics (e.g., misclassification errors and accuracy) are often used to indicate the validity of the selected features.

2.5.2. Classification model

Various classification models can be taken as the building block of Fig. 2 in training a classifier. We here adopt the following four commonly used classification models that have different metrics. For a given labeled training set *Data* and a test sample x, random forest (RF) uses a collection of decision trees to train a classifier and the prediction of x can be made using formula (4).

$$l = \max_{l_i \in L} \sum_{prd \in RF} I(prd(x) \ == \ l_i) \tag{4}$$

Naïve bayes (NB) calculates the posteriori probability of label l_i with formula (5) and determines the label with the maximum a posteriori (MAP) criterion.

$$p(l_i|x) = \frac{p(l_i)p(x|l_i)}{\sum_{l_i} p(l_i)p(x|l_i)}, \ \ l_i \in L \tag{5}$$

where $p(l_i)$ is the prior probability of label l_i and $p(x|l_i)$ is the likelihood function. We can estimate $p(l_i)$ and $p(x|l_i)$ from *Data*.

K-nearest-neighbor (KNN) is an instance-based non-parametric learning algorithm and determines the label of x based on the dominant label of its k closest neighbors from the training data. Specifically, let k be the number of nearest neighbors, $nh(Data, x)$ be the k nearest neighbors to x and $Y(nh)$ be the labels of the samples in nh, we can use formula (6) to predict the label l of x,

$$l = \max_{l_i \in L} \sum_{nb \in nh(Data,x)} I(Y(nb) \ == \ l_i) \tag{6}$$

where $I(a == b)$ is the indicator function and return 1 if a equals b.

Support vector machine (SVM) has the superiority of handling high-dimensional data and aims to seek an optimal separating hyperplane that has a maximal margin between two classes by solving formula (7).

$$\begin{aligned} &\min \frac{1}{2}||w||^2 + \lambda \sum_{i=1}^{m} \xi_i \\ &\{ s.t. \\ &\quad y_i[(w^{\mathbf{T}} x_i) + b] \geq 1 - \xi_i, \ \ i = 1, 2, ..., m \\ &\quad \xi_i \geq 0, \ \ i = 1, 2, ..., m \end{aligned} \tag{7}$$

where w and b are the hyperplane parameters to be learnt.

3. Experiments and results

3.1. Experimental setup

Besides CFS and ensemble CFS, we take other six commonly used feature selection methods, including reliefF, mutual information maximization (MIM), minimum redundancy maximum relevancy (MRMR), conditional mutual information maximization (CMIM), joint mutual information (JMI), and fast correlation-based filter (FCBF) as the competitors. Specifically, for feature ranking methods (i.e., reliefF, MIM, MRMR, CMIM, and JMI), we experimentally choose the top twenty-five features. CFS and FCBF belong to feature subset selection methods and there is no need to specify the number of selected features. For the ensemble feature selection method, five-fold cross validation is applied on the training set to obtain multiple sampling datasets and CFS is used to select multiple subsets of features. Then, we use the two proposed aggregation strategies to get the finally selected features.

To test the stability of a feature selection method to the change of experimental data, the stratified ten-fold cross validation is used to generate perturbated training data and to get independent training and test sets. That is, the microarray dataset is partitioned into ten folds, where each one of the ten folds is used as a test set and the remaining folds are used as the training set. We then get ten pairs of training set and test set and finally report the average of the ten runs.

3.2. Evaluation metric

3.2.1. Classification metric

Predictive power and stability are two critical metrics in evaluating a feature selection algorithm. For the former, classification performance is often used to indicate the predictive ability of selected features and we here take accuracy and F1 as the metrics.

$$accuracy = \sum_{i=1}^{|L|} T_i \Big/ \sum_{i=1}^{|L|} NT_i \tag{8}$$

$$precision = \frac{1}{|L|} \sum_{i=1}^{|L|} \frac{T_i}{NP_i} \tag{9}$$

$$recall = \frac{1}{|L|} \sum_{i=1}^{|L|} \frac{T_i}{NT_i} \tag{10}$$

$$F1 = \frac{2 * precision * recall}{precision + recall} \tag{11}$$

where T_i is the number of samples correctly classified as l_i, NP_i is the number of samples classified into l_i, and NT_i is the number of samples labeled as l_i.

To avoid the classifier selection bias, we here use NB, KNN, SVM, and RF to measure the goodness of selected features. Nearest neighbor is used for KNN and RF consists of five decision trees. For SVM, we use the linear kernel and default parameter values in LIBSVM and also use one-against-one strategy to handle the multi-class problem for SVM.

3.2.2. Stability metric

Stability measures the robustness of a feature selector to the change of training data. There are different ways of perturbating the training data and we here use Q-fold cross validation to return Q feature subsets and to measure the similarity of the Q sets. Algorithm 2 shows the corresponding pseudo-code, where line 2 obtains the ith training set TR_i of the Q-fold cross validation, line 3 conducts feature selection on TR_i and gets a subset of features S_i, and line 5 calculates the stability scores

s_{idx} of the Q subsets. As for the stafbility metric ψ, *Jaccard*, adjusted similarity (Sim_L), and relative weighted consistency (CW_{rel}) that have different statistical criteria are used to get s_{idx} [30]. The larger the value s_{idx} is, the more stable the corresponding feature selector is. Specifically, CW_{rel} utilizes the frequency of each selected feature in the Q sets to calculate s_{idx}, where *Jaccard* and Sim_L calculate the average similarity of the Q subsets. Given Q sets $A = \{S_1, S_2, \ldots, S_Q\}$, *Jaccard* index is obtained using equation (12),

$$\psi_{Jaccard}(A) = \frac{2}{Q(Q-1)} \sum_{i=1}^{Q-1} \sum_{j=i+1}^{Q} \frac{|s_i \cap s_j|}{|s_i \cup s_j|} \tag{12}$$

where $\cap$ and $\cup$ denotes the intersection and union between two sets, respectively. Sim_L is obtained using equation (13),

$$\psi_{Sim_L}(A) = \frac{2}{Q(Q-1)} \sum_{i=1}^{Q-1} \sum_{j=i+1}^{Q} \frac{|s_i \cap s_j| - E(|r|)}{\max(|r|) - \min(|r|)} \tag{13}$$

where r equals the intersection of s_i and s_j, $|r|$ is the cardinality of r, and $E(|r|)$, $\max(|r|)$, and $\min(|r|)$ are the expectation, maximum, and minimum of $|r|$, respectively.

CW_{rel} is calculated using equations (14) and (15). For extra details, we refer to Ref. [30].

$$\psi_{CW_{rel}}(A) = \frac{CW(A) - CW_{\min}}{CW_{\max} - CW_{\min}} \tag{14}$$

$$CW(A) = \sum_{f \in F} p_f \frac{p_f - p_{\min}}{p_{\max} - p_{\min}} \tag{15}$$

where p_{min} (p_{max}) is the minimum (maximum) frequency of $f \in F$ and CW_{min} (CW_{max}) is the minimum (maximum) of $CW(A)$.

Algorithm 2. Stability calculation

Algorithm 2. Stability calculation	
Input	training set *Data*, number of folds Q, feature selection algorithm *FS*, stability metric ψ
Output	stability score s_{idx}
1	**for** $i = 1$ **to** Q
2	$TR_i = cv\_i(Data)$; //the i^{th} training set of Q-fold cross validation
3	$S_i = FS(TR_i)$; //feature selection using Algorithm 1
4	**endfor**
5	$s_{idx} = \psi(S_1, S_2, \ldots, S_Q)$; //using (11), (12) or (13)
6	**return** s_{idx}

3.3. Experimental results

As discussed in subsection 2.4, different threshold values γ can be used to combine multiple sets into one set. We herein choose 1, 0.75, 0.5, 0.25, and 0, and get corresponding feature selection algorithms $enCFS_I$, $enCFS_T$, $enCFS_H$, $enCFS_Q$, and $enCFS_U$. For example, $\gamma = 0$ corresponds to $enCFS_U$ that returns the union of multiple sets and $\gamma = 1$ ($enCFS_I$) gets their intersection. Besides, we apply CFS on the result of $enCFS_U$ and name the method $enCFS_2$. It is worth noting that the feature subsets returned by $enCFS_I$ and $enCFS_T$ are empty in our experiments, indicating the sensitivity of CFS to the change of training set. We thus leave out corresponding results.

3.3.1. Classification performance

Tables 2–5 present the classification accuracy and F1 of the eleven feature selection algorithms when NB, KNN, SVM, and RF are used as classification models, respectively. The third column "w/o" lists the results without using feature selection method and the last four columns correspond to the results of ensemble CFS. For each dataset, the best classification accuracy is shown in bold and we underline the result of ensemble CFS if it is higher than that of CFS. From Tables 2–5, we observe that CFS selects a subset of high-quality features and obtains comparable classification accuracy to reliefF, MIM, MRMR, CMIM, JMI, and FCBF. For example, in terms of NB on *Colon*, CFS achieves 82.26% accuracy, compared to the 85.48%, 82.26%, 82.26%, 83.87%, 82.26% and 80.65% of its competitors; on *SRBCT*, CFS has 98.8% accuracy, while the accuracies of reliefF, MIM, MRMR, CMIM, JMI, and FCBF are 91.57%, 98.80%, 98.80%, 96.39%, 97.59% and 98.80%, respectively. Second, when comparing CFS and its ensembles, we observe that the ensemble methods could obtain comparable or even better performance. For $enCFS_H$, $enCFS_Q$, $enCFS_U$, and $enCFS_2$, we observe that $enCFS_H$ generally obtains higher accuracy in most cases. For example, if using NB, $enCFS_H$ achieves the highest accuracy on *Colon* and *SRBCT* and obtains the second-best accuracy on *DLBCL* and *Leukemia*; if using SVM, $enCFS_H$ obtains the best accuracy on *Colon* and *Leukemia*. This is mainly because $enCFS_H$ uses a majority voting-based aggregation strategy that helps preserve informative features while filtering out the randomly selected features due to data perturbation. Besides, we observe that there are cases on *DLBCL* and *SRBCT* that the classification accuracy of the original features is slightly higher than that of using feature selection. This is possible because *DLBCL* and *SRBCT* exhibit distinct expression patterns of different classes and a classification model can make, but does not guarantee across different models, good predictions.

Table 2
Experimental results of different feature selection methods with NB.

Dataset	*Metric*(%)	w/o	reliefF	MIM	MRMR	CMIM	JMI	FCBF	CFS	$enCFS_H$	$enCFS_Q$	$enCFS_U$	$enCFS_2$
SRBCT	Accuracy	**100.0**	91.57	98.80	98.80	96.39	97.59	98.80	98.80	**100.0**	98.80	98.80	98.80
	F1	100.0	90.55	98.20	98.20	96.34	97.24	99.09	99.09	100.0	99.09	98.44	99.09
Colon	Accuracy	58.07	**85.48**	82.26	82.26	83.87	82.26	80.65	82.26	**85.48**	83.87	**85.48**	82.26
	F1	61.02	84.01	81.37	81.37	83.39	81.37	80.71	81.37	84.81	83.39	84.81	81.37
DLBCL	Accuracy	79.22	92.21	89.61	90.91	**96.10**	89.61	89.61	90.91	92.21	92.21	92.21	93.51
	F1	71.09	89.52	87.28	88.01	94.88	87.28	85.41	87.58	89.24	89.52	89.52	91.44
Leukemia	Accuracy	**98.61**	91.67	94.44	95.83	94.44	97.22	95.83	93.06	95.83	95.83	97.22	93.06
	F1	97.70	86.06	90.38	92.90	91.75	95.42	92.90	89.25	92.90	92.90	95.42	89.25

Table 3
Experimental results of different feature selection methods with KNN.

Dataset	*Metric*(%)	w/o	reliefF	MIM	MRMR	CMIM	JMI	FCBF	CFS	enCFS$_H$	enCFS$_Q$	enCFS$_U$	*enCFS$_2$*
SRBCT	Accuracy	84.34	91.57	98.80	**100.0**	95.18	**100.0**	98.80	**100.0**	98.80	97.59	**100.0**	98.80
	F1	85.21	91.36	99.08	100.0	95.31	100.0	99.08	100.0	99.08	98.14	100.0	99.08
Colon	Accuracy	82.26	75.81	79.03	79.03	82.26	80.65	75.81	80.65	**85.48**	82.26	83.87	80.65
	F1	80.45	72.57	76.88	76.88	81.37	78.28	73.86	78.86	84.01	80.20	82.10	78.50
DLBCL	Accuracy	80.52	88.31	83.12	**98.70**	94.81	96.10	97.40	90.91	92.21	97.40	93.51	92.21
	F1	74.27	83.62	79.54	98.25	93.01	94.69	96.51	88.59	89.52	96.51	91.92	89.97
Leukemia	Accuracy	84.72	91.67	93.06	93.06	94.44	**95.83**	94.44	93.06	**95.83**	94.44	94.44	93.06
	F1	84.29	89.70	90.28	91.95	94.17	95.66	93.27	90.78	94.34	91.83	94.43	90.70

Table 4
Experimental results of different feature selection methods with SVM.

Dataset	*Metric*(%)	w/o	reliefF	MIM	MRMR	CMIM	JMI	FCBF	CFS	enCFS$_H$	enCFS$_Q$	enCFS$_U$	*enCFS$_2$*
SRBCT	Accuracy	**100.0**	95.18	98.80	**100.0**	95.18	98.80	96.39	98.80	98.80	98.80	98.80	98.80
	F1	100.0	94.96	99.08	100.0	95.65	99.08	97.22	99.08	99.08	99.08	99.08	99.08
Colon	Accuracy	83.87	80.65	79.03	79.03	72.58	75.81	79.03	79.03	**88.71**	80.65	72.58	83.87
	F1	82.10	78.50	77.35	77.35	71.06	74.50	78.63	76.88	87.82	78.86	69.23	82.39
DLBCL	Accuracy	96.10	93.51	92.21	96.10	96.10	96.10	94.81	94.81	94.81	96.10	**97.40**	93.51
	F1	94.69	91.14	89.52	94.88	94.69	94.88	92.87	93.01	93.01	94.69	96.51	91.44
Leukemia	Accuracy	**95.83**	91.67	94.44	94.44	**95.83**	**95.83**	**95.83**	**95.83**	**95.83**	93.06	**95.83**	**95.83**
	F1	95.50	89.40	91.97	94.29	94.09	95.66	94.34	94.34	93.28	89.51	95.50	94.34

Table 5
Experimental results of different feature selection methods with RF.

Dataset	*Metric*(%)	w/o	reliefF	MIM	MRMR	CMIM	JMI	FCBF	CFS	enCFS$_H$	enCFS$_Q$	enCFS$_U$	*enCFS$_2$*
SRBCT	Accuracy	87.95	92.77	89.16	93.98	92.77	**97.59**	90.36	89.16	91.57	95.18	93.98	96.39
	F1	88.95	94.22	87.56	94.61	93.54	97.98	90.62	88.96	92.84	95.48	94.21	96.47
Colon	Accuracy	75.81	80.65	82.26	80.65	**83.87**	82.26	82.26	79.03	82.26	82.26	**83.87**	80.65
	F1	72.89	78.50	80.45	78.50	82.10	80.20	80.20	77.35	80.45	80.45	82.10	78.28
DLBCL	Accuracy	81.82	89.61	87.01	**92.21**	88.31	**92.21**	88.31	89.61	**92.21**	**92.21**	88.31	**92.21**
	F1	74.72	85.41	81.98	89.18	83.62	89.42	83.41	85.61	89.24	89.18	83.62	89.24
Leukemia	Accuracy	80.56	87.50	90.28	88.89	91.67	91.67	90.28	91.67	93.06	**94.44**	88.89	88.89
	F1	76.22	83.47	86.31	86.18	89.39	90.99	85.03	91.00	91.71	94.29	87.33	85.73

Table 6
Number of selected features (mean ± standard deviation).

Dataset	CFS		enCFS$_H$		enCFS$_Q$		enCFS$_U$		*enCFS$_2$*	
	#avg	#std	#avg	#std	#avg	#std	#avg	#std	#avg	#std
SRBCT	116.7	8.4	72.8	4.3	120.6	9.7	241.6	9.7	105.2	7.4
Colon	23..8	4.2	11.7	2.8	24.3	2.9	74.9	7.9	21.4	3.7
DLBCL	42.7	5.2	10.4	1.9	28.7	3.2	179.6	13.0	34.7	4.4
Leukemia	91.1	6.0	45.3	3.8	82.9	6.9	231.4	8.3	77.9	5.6

Previous studies have also reported similar results [7,19,40]. Even so, we can observe that feature selection helps obtain stable prediction performance across different classifiers and serves for downstream biological interpretation, which, to a certain extent, indicates the necessity and power of feature selection methods.

3.3.2. Number of selected genes

We then investigate the selected features. Table 6 shows the mean and standard deviation of the number of selected features for CFS and ensemble CFS on each dataset. From Table 6, we can observe that CFS, enCFS$_H$, enCFS$_Q$, enCFS$_U$, and enCFS$_2$ select a small number of features from the original feature space, which greatly reduces its dimensionality and helps mitigate overfitting. Taking the *Colon* with 2000 genes as an example, CFS selects average of 24 genes, enCFS$_H$ selects 12 genes on average, and enCFS$_U$ selects 75 genes. Second, we observe that enCFS$_H$ obtains a feature subset of smaller size and smaller standard deviation than CFS. For example, the average number of selected features of enCFS$_H$ on *Colon*, *DLBCL*, *Leukemia*, and *SRBCT* are 11.7, 10.4, 45.3, and 72.8, respectively, compared to the 23.8, 42.7, 91.1, and 116.7 of CFS. This is possible because the use of occurrence frequency could discard the features sensitive to the change of training data. Third, we also observe that for enCFS$_H$, enCFS$_Q$, and enCFS$_U$, the number of finally selected features increases as the threshold γ decreases. For example, on *SRBCT*, enCFS$_H$ selects 72.8 features, enCFS$_Q$, selects 120.6 features, and enCFS$_U$ selects 241.6 features. This is mainly because the decrease of γ tends to keep features that have low occurrence frequency. Fourth, compared with enCFS$_U$ and enCFS$_2$, we observe that enCFS$_2$ obtains a feature subset of smaller size, which indicates that there exists redundancy among features selected by enCFS$_U$. For example, the average number of selected features for enCFS$_2$ on *SRBCT* is 105.2 compared to the 241.6 of enCFS$_U$.

Afterwards, we preliminarily investigate the selected genes by *enCFS$_H$*. Without loss of generality, we here present the results on *Colon* as an example [35], as shown in Table 7. We observe that the selected genes have a role in the biological process.

3.3.3. Stability scores

In this section, we compare the stability of CFS and its ensembles,

Table 7
Selected genes and their description.

Gene number	Description	Gene number	Description
R28373	Human calmodulin mRNA, complete cds.	R36977	P03001 transcription factor IIIA
K03460	Human alpha-tubulin isotype H2-alpha gene, last exon	R84411	Small Nuclear Ribonucleprotein associated proteins B and B' (human)
Z50753	H. sapiens mRNA for GCAP-II/uroguanylin precursor	M36634	Human vasoactive intestinal peptide (VIP) mRNA, complete cds.
H40560	Thioredoxin (human)	H08393	Collagen alpha 2(XI) CHAIN (Homo sapiens)
R87126	Myosin heavy chain, nonmuscle (Gallus gallus)	M26383	Human monocyte-derived neutrophil-activating protein (MONAP) mRNA, complete cds.
X12671	Human gene for heterogeneous nuclear ribonucleoprotein (hnRNP) core protein A1	J02854	Myosin regulartory light chain 2, Smooth muscle isoform (human); contains element TAR1 repetitive element
M76378	Human cysteine-rich protein (CRP) gene, exons 5 and 6		

including $enCFS_H$, $enCFS_Q$, $enCFS_U$, and $enCFS_2$. Fig. 3 presents corresponding results, where the X-axis denotes the stability metrics (i.e., *Jaccard*, Sim_L, and CW_{rel}) and the Y-axis refers to stability scores. The higher the scores are, the more stable the feature selector is. From Fig. 3, we observe that the relative numerical relations of the five compared methods are stable across *Jaccard*, Sim_L, and CW_{rel}, which indicates the feasibility of the three metrics in measuring the similarity of multiple sets. Second, ensembled CFS versions generally has better stability in most cases and $enCFS_H$ obtains better stability scores across datasets. For example, on *Leukemia*, $enCFS_H$ gets the scores of 0.3687, 0.5577, and 0.539 for *Jaccard*, Sim_L, and CW_{rel} respectively, which outperforms the corresponding 0.3598, 0.5318, and 0.5285 of CFS. Third, we observe that $enCFS_2$ has higher stability scores than $enCFS_U$. This is mainly because $enCFS_2$ conducts a further feature selection to discard redundant and irrelevant features, which helps stabilize the results of feature selection.

Furthermore, considering that accuracy and stability are two key factors in evaluating a feature selector, we plot their relationships to direct the choice of feature selection methods. Fig. 4 shows the accuracy vs.

stability results of CFS and its ensembles, where the X-axis denotes the accuracy of SVM and the Y-axis is the Sim_L scores. Fig. 4 clearly shows that $enCFS_H$ generally achieves a better tradeoff between accuracy and stability. Particularly, we observe that $enCFS_H$ consistently outperforms CFS on all datasets. Similar results can be observed for other stability metrics and classifiers.

4. Discussions

Accurate and stable feature selection from microarray gene expression data is of paramount importance to the classification of cancers and tumors and the identification of indicative biomarkers. Considering that a feature selection method is usually sensitive to the change of training data, we in this study develop a new feature selection framework under the ensemble learning paradigm. We conducted extensive experiments on public microarray datasets and achieved better classification accuracy and stability than the non-ensemble (see Figs. 3 and 4). Moreover, the proposed $enCFS_H$ further optimizes the feature space and selects a feature subset of smaller size than its competitors, while maintaining stability and discriminant ability (see Table 5). Besides, we investigate the selected features from the view of biological knowledge. The results described in Table 6 illustrates that the selected genes play a role in the biological process.

Previous approaches for selecting a subset of features using a group

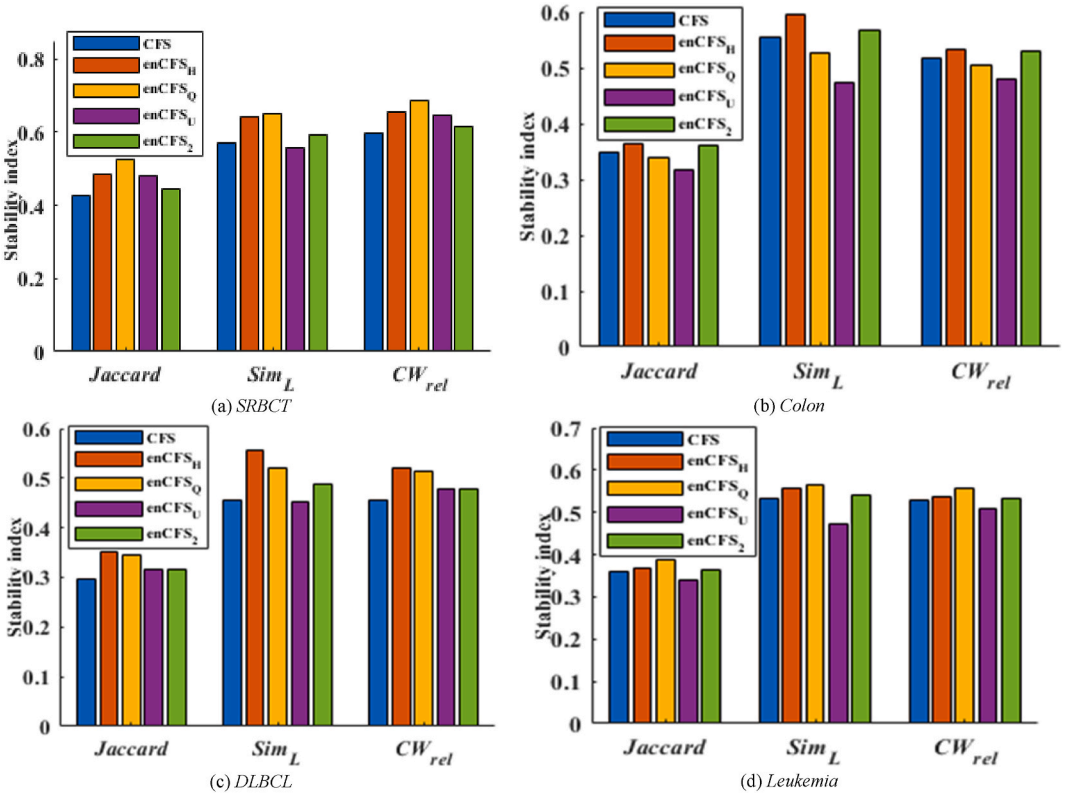

Fig. 3. Comparisons of stability scores. Three different stability metrics are used for fair comparisons.

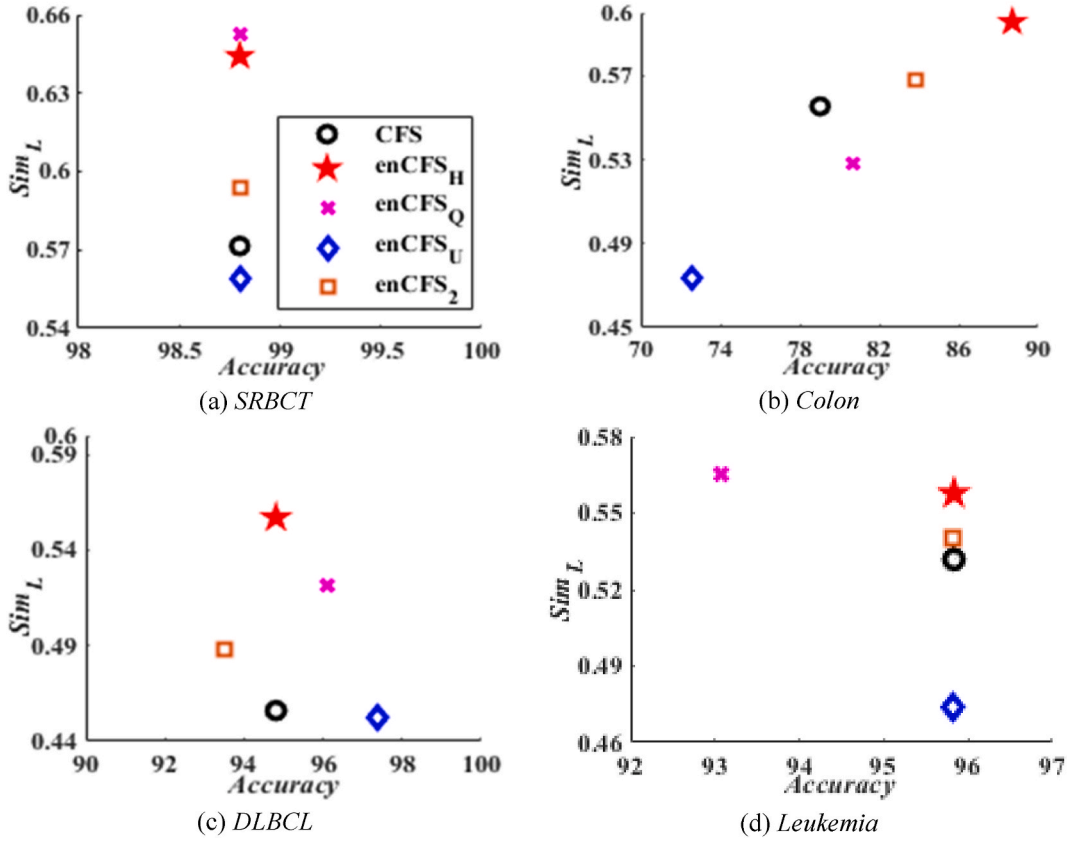

Fig. 4. Accuracy vs. stability of different feature selection methods.

of feature selection methods mainly face the problem of thresholding and aggregation. That is, users have to decide how many features to keep for feature ranking methods [25] and specify the order of performing thresholding and aggregation. Since the two procedures are optimized individually, a sub-optimal solution is usually obtained [33]. This also expects an end-to-end solution towards enhanced performance. In contrast, our proposed method can automatically return the finally selected features, which largely relieves users from the issue. Experimental results also demonstrated its effectiveness and flexibility.

One limitation of the proposed method is that it has higher time complexity than traditional methods, since the proposed ensemble feature selection method is a collection of base feature selectors. The complexity of a base feature selection method is mainly determined by the number of samples m and the number of features n, and we note it as $h(m, n)$ for general analysis. Suppose the ensemble feature selection method consists of M base feature selectors and the maximal cardinality of the M selected subsets is d ($d \ll n$), the corresponding time complexity is $O(M^* h(m, n) + M^*d) \approx O(M^* h(m, n))$, where $O(M^* h(m, n))$ is associated with M base feature selection methods and $O(M^*d)$ is related to aggregation. One feasible way to reduce the time complexity is to distribute the base feature selection tasks to multiple computers or servers (such as M machines) using parallel computing. Hence, the time complexity is reduced to $O(h(m, n) + M^*d)$.

5. Conclusions

The small-sample-size and high-dimensional microarray data pose a great challenge to the identification of indicative biomarkers and the training of a powerful classifier. Although gene selection remains a priority to mitigate the problem, most of existing methods have poor stability and the selected features are sensitive to the change of training data. Accordingly, accurate and stable feature selection plays a critical role in the analysis of gene expression profiles and would help researchers to conduct downstream biological analysis. To this end, we herein propose an ensemble feature selection framework that can return a discriminating and stable subset of features. We then present two aggregation strategies for combining multiple sets into one set and introduce three stability metrics (including *Jaccard*, Sim_L and CW_{rel}) and two classification performance metrics (i.e., accuracy and F1) towards an unbiased evaluation of a feature selection method. Finally, we conduct comparative experiments on public microarray datasets that cover both binary and multi-class cases. Results demonstrate the superiority of the proposed method over its competitors in terms of classification performance and stability.

For future work, we plan to work in the following directions. First, aggregation strategy plays a central role in the ensemble feature selection method. We currently only use frequency information to design the aggregation strategy and ignore the relative importance of features, which motivates us to explore weighted aggregation strategies for further study. Second, the proposed framework is a general one, and we can take as the building blocks other base feature selection algorithms. However, this would raise new issues related to hyperparameter optimization such as how many features to keep (i.e., the thresholding problem) and how to combine the ranked features (i.e., the aggregation problem) [33,41]. This deserves a systematic and comprehensive study for practical guidelines. Third, we can apply the proposed method to RNA-seq and metagenome omics data and research fields to better uncover underlying mechanisms [42–44].

Declaration of competing interest

We declare that there are no conflicts of interest regarding the publication of this paper, and the manuscript is approved by all authors for publication.

Acknowledgment

This work was supported partially by the Guangdong Basic and Applied Basic Research Foundation under Award Number 2020A1515011499 and the Natural Science Foundation of China under Award Number 62176082. The authors are very grateful to the reviewers for their constructive comments and suggestions for the improvement of this research.

References

[1] R.R. De Assis, A. Jain, R. Nakajima, A. Jasinskas, J. Felgner, J.M. Obiero, P. J. Norris, M. Stone, G. Simmons, A. Bagri, J. Irsch, Analysis of SARS-CoV-2 antibodies in COVID-19 convalescent blood using a coronavirus antigen microarray, Nat. Commun. 12 (1) (2021) 1–9.
[2] A. Fukushima, M. Sugimoto, S. Hiwa, T. Hiroyasu, Bayesian approach for predicting responses to therapy from high-dimensional time-course gene expression profiles, BMC Bioinf. 22 (1) (2021) 1–20.
[3] J. Khan, J.S. Wei, M. Ringner, L.H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C.R. Antonescu, C. Peterson, P.S. Meltzer, Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks, Nat. Med. 7 (6) (2001) 673–679.
[4] A. Negi, A. Shukla, A. Jaiswar, J. Shrinet, R.S. Jasrotia, Applications and challenges of microarray and RNA-sequencing, Bioinformatics (2022) 91–103.
[5] A. Mirzal, Statistical analysis of microarray data clustering using NMF, spectral clustering, kmeans, and GMM, IEEE ACM Trans. Comput. Biol. Bioinf (1) (2020) 1.
[6] M. Abdulla, M.T. Khasawneh, G-Forest: an ensemble method for cost-sensitive feature selection in gene expression microarrays, Artif. Intell. Med. 108 (2020) 101941.
[7] A. Wang, N. An, G. Chen, L. Liu, G. Alterovitz, Subtype dependent biomarker identification and tumor classification from gene expression profiles, Knowl. Base Syst. 146 (2018) 104–117.
[8] S. Peng, Y. Yang, W. Liu, F. Li, X. Liao, Discriminant projection shared dictionary learning for classification of tumors using gene expression data, IEEE ACM Trans. Comput. Biol. Bioinf 18 (4) (2021) 1464–1473.
[9] B. Dumitrascu, S. Villar, D.G. Mixon, B.E. Engelhardt, Optimal marker gene selection for cell type discrimination in single cell analyses, Nat. Commun. 12 (1) (2021) 1–8.
[10] J. Li, K. Cheng, S. Wang, F. Morstatter, R.P. Trevino, J. Tang, H. Liu, Feature selection: a data perspective, ACM computing surveys (CSUR) 50 (6) (2017) 1–45.
[11] D.M. Abdulqader, A.M. Abdulazeez, D.Q. Zeebaree, Machine learning supervised algorithms of gene selection: a review, Mach. Learn. 62 (3) (2020).
[12] A. Wang, S. Zhao, J. Liu, J. Yang, L. Liu, G. Chen, Locality adaptive preserving projections for linear dimensionality reduction, Expert Syst. Appl. 151 (2020) 113352.
[13] C.M. Lai, H.P. Huang, A gene selection algorithm using simplified swarm optimization with multi-filter ensemble technique, Appl. Soft Comput. 100 (2021) 106994.
[14] J. Sheng, W.V. Li, Selecting gene features for unsupervised analysis of single-cell gene expression data, Briefings Bioinf. 22 (6) (2021) bbab295.
[15] S. Liu, C. Xu, Y. Zhang, J. Liu, B. Yu, X. Liu, M. Dehmer, Feature selection of gene expression data for cancer classification using double RBF-kernels, BMC Bioinf. 19 (1) (2018) 1–14.
[16] K. Kourou, G. Rigas, C. Papaloukas, M. Mitsis, D.I. Fotiadis, Cancer classification from time series microarray data through regulatory dynamic bayesian networks, Comput. Biol. Med. 116 (2020) 103577.
[17] G. Brown, A. Pocock, M.J. Zhao, M. Luján, Conditional likelihood maximisation: a unifying framework for information theoretic feature selection, J. Mach. Learn. Res. 13 (2012) 27–66.
[18] A. Wang, N. An, J. Yang, G. Chen, L. Li, G. Alterovitz, Wrapper-based gene selection with Markov blanket, Comput. Biol. Med. 81 (2017) 11–23.
[19] A. Wang, H. Liu, G. Chen, May. Chaotic harmony search based multi-objective feature selection for classification of gene expression profiles, in: 2021 IEEE 9th International Conference on Bioinformatics and Computational Biology (ICBCB), IEEE, 2021, pp. 107–112.
[20] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machines, Mach. Learn. 46 (1) (2002) 389–422.
[21] N. Almugren, H. Alshamlan, A survey on hybrid feature selection methods in microarray gene expression data for cancer classification, IEEE access 7 (2019) 78533–78548.
[22] C.M. Lai, H.P. Huang, A gene selection algorithm using simplified swarm optimization with multi-filter ensemble technique, Appl. Soft Comput. 100 (2021) 106994.
[23] W. Awada, T.M. Khoshgoftaar, D. Dittman, R. Wald, A. Napolitano, August. A review of the stability of feature selection techniques for bioinformatics data, in: 2012 IEEE 13th International Conference on Information Reuse & Integration (IRI), IEEE, 2012, pp. 356–363.
[24] S. Nogueira, K. Sechidis, G. Brown, On the stability of feature selection algorithms, J. Mach. Learn. Res. 18 (1) (2017) 6345–6398.
[25] T. Abeel, T. Helleputte, Y. Van de Peer, P. Dupont, Y. Saeys, Robust biomarker identification for cancer diagnosis with ensemble feature selection methods, Bioinformatics 26 (3) (2010) 392–398.
[26] L. Yu, Y. Han, M.E. Berens, Stable gene selection from microarray data via sample weighting, IEEE ACM Trans. Comput. Biol. Bioinf 9 (1) (2011) 262–272.
[27] A. Wang, H. Liu, J. Liu, H. Ding, J. Yang, G. Chen, December. Stable and accurate feature selection from microarray data with ensembled fast correlation based filter, in: 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), IEEE, 2020, pp. 2996–2998.
[28] K. Yang, Z. Cai, J. Li, G. Lin, A stable gene selection in microarray data analysis, BMC Bioinf. 7 (1) (2006) 1–16.
[29] L. Yu, C. Ding, S. Loscalzo, August. Stable feature selection via dense feature groups, in: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2008, pp. 803–811.
[30] P. Somol, J. Novovičová, Evaluating stability and comparing output of feature selectors that optimize feature subset cardinality, IEEE Trans. Pattern Anal. Mach. Intell. 32 (11) (2010) 1921–1939.
[31] V. Hamer, P. Dupont, An importance weighted feature selection stability measure, J. Mach. Learn. Res. 22 (116) (2021) 1–57.
[32] X. Zhao, Q. Jiao, H. Li, Y. Wu, H. Wang, S. Huang, G. Wang, ECFS-DEA: an ensemble classifier-based feature selection for differential expression analysis on expression profiles, BMC Bioinf. 21 (1) (2020) 1–14.
[33] V. Bolón-Canedo, A. Alonso-Betanzos, Ensembles for feature selection: a review and future trends, Inf. Fusion 52 (2019) 1–12.
[34] F. Yang, K.Z. Mao, Robust feature selection for microarray data based on multicriterion fusion, IEEE ACM Trans. Comput. Biol. Bioinf 8 (4) (2010) 1080–1092.
[35] U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, A.J. Levine, Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, Proc. Natl. Acad. Sci. Unit. States Am. 96 (12) (1999) 6745–6750.
[36] M.A. Shipp, K.N. Ross, P. Tamayo, A.P. Weng, J.L. Kutok, R.C. Aguiar, M. Gaasenbeek, M. Angelo, M. Reich, G.S. Pinkus, T.S. Ray, Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning, Nat. Med. 8 (1) (2002) 68–74.
[37] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield, Molecular classification of cancer: class discovery and class prediction by gene expression monitoring, Science 286 (5439) (1999) 531–537.
[38] M.A. Hall, June. Correlation-based feature selection for discrete and numeric class machine learning, in: Proceedings of the Seventeenth International Conference on Machine Learning, 2000, pp. 359–366.
[39] C. Ambroise, G.J. McLachlan, Selection bias in gene extraction on the basis of microarray gene-expression data, Proc. Natl. Acad. Sci. Unit. States Am. 99 (10) (2002) 6562–6566.
[40] J. Yang, J. Zhou, Z. Zhu, X. Ma, Z. Ji, Iterative ensemble feature selection for multiclass classification of imbalanced microarray data, J.Biol. Res.thessaloniki 23 (1) (2016) 1–9.
[41] A. Ahmadi, H. Bazregarzadeh, K. Kazemi, Automated detection of driver fatigue from electroencephalography through wavelet-based connectivity, Biocybern. Biomed. Eng. 41 (1) (2021) 316–332.
[42] M. Petti, A. Verrienti, P. Paci, L. Farina, SEaCorAl: identifying and contrasting the regulation-correlation bias in RNA-Seq paired expression data of patient groups, Comput. Biol. Med. (2021) 104567.
[43] D. Barh, S. Tiwari, M.E. Weener, V. Azevedo, A. Góes-Neto, M.M. Gromiha, P. Ghosh, Multi-omics-based identification of SARS-CoV-2 infection biology and candidate drugs against COVID-19, Comput. Biol. Med. 126 (2020) 104051.
[44] A. Ahmadi, S. Davoudi, M.R. Daliri, Computer aided diagnosis system for multiple sclerosis disease based on phase to amplitude coupling in covert visual attention, Comput. Methods Progr. Biomed. 169 (2019) 9–18.