IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON DATA-ENABLED INTELLIGENCE FOR DIGITAL HEALTH

Received June 30, 2019, accepted July 15, 2019, date of publication July 29, 2019, date of current version August 13, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2931746

Microarray Data Classification Based on Computational Verb

KUN-HONG LIU[1], VINCENT TO YEE NG[2], SZE-TENG LIONG[3], AND QINGQI HONG[1]

[1]School of Informatics, Xiamen University, Xiamen 361005, China
[2]Department of Computing, The Hong Kong Polytechnic University, Hong Kong
[3]Department of Electronic Engineering, Feng Chia University, Taichung 40724, Taiwan

Corresponding author: Qingqi Hong (hongqq@xmu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61772023 and Grant 61502402, in part by the Natural Science Foundation of Fujian Province under Grant 2016J01320 and Grant 2015J05129, in part by the Ministry of Science and Technology under Grant MOST 107-2218-E-035-006, in part by the National Key Research and Development Program of China under Grant 2019QY1803, and in part by the Fundamental Research Funds for the Central Universities under Grant 20720180073.

ABSTRACT Computational verb (CV) theory is a relatively new research field in mathematics and has been applied to many different fields. In the field of pattern recognition, the CV-based rule induction algorithm can generate some simple rules with CVs and adverbs by linguistically interpretable forms. In this paper, we present an interpretable rule extraction framework based on CV rule theory for the classification of microarray data. In contrast to the existing rule-based methods, the CV method enables to explicitly express the relationships of the genes based on some mathematical templates and hence enhance the understanding on the data results. Stay is a typical verb used in the CV to describe the trend of changes. In our algorithm, Stay is applied to generate CVR by a gene pair, named SCVR. The corresponding evolving and similarity functions for calculating the difference between SCVR rules are also presented to illustrate this process. Similar to other rule-based methods, the SCVR can achieve significant gene selection and cancer classification task concurrently. To evaluate the performance of our proposed approach, we conduct the experiments on several binary class and multiclass microarray datasets. Experiments confirm that the proposed method can outperform many rule-based classiers with the fusion of five rules.

INDEX TERMS Computational verb, computational verb rules, stay, microarray Data, classifier ensemble.

I. INTRODUCTION

With the rapid development of bioinformatics technology, it is possible to diagnose some types of cancers directly using microarray technique [1]. To date, there has been a large number of machine learning methods introduced to analyze microarray data. With the aid of these methods, we are provided with an insight of the molecular variations among tumors and normal tissues [2]. In order to further explore gene functions and regulation relationships, researchers are keen to utilize mining methodologies to produce both accurate results and comprehensive knowledge in the process of microarray data analysis [3]. It seems to be a reasonable request for models to inhere good interpretability. However, for some of the classifiers, such as Support Vector Machines (SVM) and neural networks, it is hard for the researches to understand or explain the trained models due to their complex structures. On the contrary, rule-based classifiers are typical interpretable models, and they could offer the researchers a way to master the roles of different genes in different cancer types.

An important and tricky property of microarray data is that the number of samples is much smaller than that of genes. At the same time, a large proportion of genes are irrelevant or redundant to cancer diagnosis [4]. Therefore, an effective solution for this problem is the application of feature selection method, as it can filter those biologically insignificant genes to improve the diagnosis accuracy [5], [6].

Another issue in the analysis of microarray is the interpretability of models. Specifically, in the literature, there are plenty of methods developed to extract rules from microarray data. For instance, the fuzzy-rule based models are designed to generate adjectives (i.e., low, high, etc.) in microarray classification. Wang and Palade [7] and Ho [8] introduced a framework to learn fuzzy rules based on genetic algorithms.

The associate editor coordinating the review of this manuscript and approving it for publication was Ying Song.

Besides, different rule-induction and filtering strategies are proposed to generate a small-scale fuzzy classifier using a grid partition of feature space, taking the form of if (Gene 1, High) then Class 1 [9], [10]. Observe that, biological knowledge can be incorporated into the algorithm by defining the forms and parameters of fuzzy membership functions. Nevertheless, these type of rules might not be able to provide a global view for a data set, as they often focus on capturing the information from local data [11].

In contrast, some algorithms are able to extract important information from the entire dataset. Hence, the generalization ability of these rules is higher than the fuzzy rules, and produces better results. An example is k–Top Scoring Pairs (TSP) [12], a classical method that compares the gene pairs and utilizes significant discriminative information contained in the data. It focuses on 'marker gene pairs', whose expression levels across the N samples are quite different in two classes. These rules take the form of: IF Gene 1> Gene 2 THEN Class 1; ELSE Class 2. On the other hand, some researchers also designed evolutionary based rule generation systems, which are applicable on an entire data set. A typical system is implemented with genetic programming (GP), designed by exploiting multiple logical and mathematical operators in individual's structure. The final GP individuals can be molded to the rules like: if max(Gene1,Gene2)>1.9190 then Class 1 [13]. These algorithms use lesser rules to draw the final decision, as each rule is able to cover the whole data set.

The advancement and development of computer technology enable the researches to have more and better options in interpreting microarray data from different perspectives and aspects [14]. In this paper, we propose a Computational Verb (CV) based method to generate CV rules (CVR) for microarray data analysis. The CV theory was first introduced by Tao Yang. Due to its high discriminating power, CV theory has also found its way in a lot of scientific fields, such as linguistics, biology, psychology, physics and computer sciences [15]. One of the applications of CV is to construct a complete artificial language into machines, whereas CVR is a further step of such applications. CVR takes the forms of some simple rules consisting of verbs and adverbs. It can describe the changes or status by summarizing interaction terms and constants into linguistically interpretable forms. In [16], the author suggested some predefined formulas to be used as verb or adverb templates when modeling different CVRs.

In this paper, we apply the verb *Stay* based CVR to tackle microarray data analysis problem, named as SCVR for short. *Stay* describes the change trends of a sample. Here, we apply this verb to extract gene pairs in which the change of a gene expression value has valuable effects to the other one. As such, SCVR could further enhance the bio-medical scientists to understand the relationships among the microarray data.

The framework of CVR, including the evolving function and the similarity function, is presented in Section 2. There are many algorithms proposed to handle the issue of multi-class microarray data classification because in this case, the sample size would be quite different in different classes. Such a class imbalance problem makes the multi-class classification task much harder than for the binary class problem [17]. Based on this consideration, our algorithm is also evaluated based on some multi-class microarray datasets.

II. INTRODUCTION TO COMPUTATIONAL VERB

Computational Verb (CV) is an essential element widely used in artificial intelligence and expert system. It facilitates in solving engineering problems by converting different types of natural words to mathematical formulas. The CV theory was invented in 1997. Since then, the interest in researching related to this area have been raised until recent [16]. CV has been applied to many different fields successfully. For example, the applications of CV to different kinds of control problems were studied on different occasions. It has been used to model many different kinds of products, such as card counters, webcam barcode scanner, smart pornographic image and video detection system [18]. It was also deployed in the design of flame-detecting systems using CCTV signal [15]. Moreover, the mathematical concept and logic operations of CV has been well studied. It is found that the difference and relationships between CV and Fuzzy mathematics is, the former uses many different verbs in the statement while the latter only uses verb BE. Furthermore, CV can be transformed to Fuzzy mathematics by CV Collapses [16], [19].

There are limited explorations in the application of CV in the bioinformatics field and very few works have been published. To the best of our knowledge, there is only one recent work that attempted to employ CV rule (CVR) based method to analyze microarray data, which is the work of Tong [20]. In his work, a computational verb rule is used to compare the change of expression levels between a genes pair to deal with a binary-class problem. The rules is expressed as: if Gene *i increases* relative to Gene *j*, then class 1; if Gene *i decreases* relative to Gene *j*, then class 2. This rule can be applied to classify a sample based on expression differential levels of a gene pair. Nevertheless, in our opinion, the verbs (i.e., *increase* and *decrease*) may not be adequate for describing the diversity of gene expression level, because the gene expression data is not time-varying. Such rules tend to be affected by the input sequence of the training set, resulting inconsistent and unreliable performance of such rules. Hence, the changes of the input sequence may lead to different decisions and incurring poor results. For instance, if we reverse the input sequence of training data, the rules formed by verb should be *increase* instead of *decrease,* and vice versa. In conclusion, such rules are not able to assure in obtaining stable results.

In this paper, we further extend the exploration and inspection in CVR to discover more interesting content and provide significant insights to bioinformatics field. Despite the usage of action related verbs (i.e., *increase* and *decrease*), a more

decent and favorable verb in describing the states to model data is recommended, which is *stay*. Similar to Tong's [17] work, our rules are also based on a gene pair, however the input sequence is not able to affect the results of our rules. In addition, we utilize computational adverbs to make our rules more knowledgeable and comprehensible..

A. THE DEFINITION OF COMPUTATIONAL VERB

Each computational verb can be represented as a 4-tuple (v, T , Ψ, ε). Here v defines a verb (e.g. *increase, stay*). (T, Ψ, ε) refers to a dynamical system. In this system, T is the life span [16], Ψ denotes the state space of the system, and ε represents the evolution function of the dynamical system.

$$\varepsilon : \mathrm{T} \times \Psi \longrightarrow \Psi \tag{1}$$

$$\varepsilon(0, \mathrm{x}) = \mathrm{x}, \varepsilon\,(\mathrm{t}_2, \varepsilon\,(\mathrm{t}_1, \mathrm{x})) = \varepsilon\,(\mathrm{t}_1 + \mathrm{t}_2, \mathrm{x}) \tag{2}$$

where (T, +) is a monoid, and x is the observation.

Verbs commonly used in CV are: *become*, *stay, increase/decrease*. They can illustrate the changes in trend within a period. Different verbs describes different levels of the changes, so they require diverse CV templates. There are also some computational adverbs, such as *slowly, fast*, exploited to facilitate the verbs. A typical CV rule (CVR) may contain a set of verbs and adverbs, often expressed in the form of:

$$\text{If } \mathrm{X_i Ad_{xi}} \circ \mathrm{V_{xi}} \text{ AND } \mathrm{X_j Ad_{xj}} \circ \mathrm{V_{xj}} \text{ THEN Y } \mathrm{Ad_Y} \circ \mathrm{V_Y} \tag{3}$$

where adverb ($\mathrm{Adx_i}$) and verb ($\mathrm{Vx_i}$) describe an action ($\mathrm{Adx_i} \circ \mathrm{Vx_i}$) on feature $\mathrm{X_i}$. Y is the target output affected by the action $\mathrm{Ad_Y} \circ \mathrm{V_Y}$. Note that each rule can comprise as many verbs and adverbs as required. However, a typical rule should not consist of more than two verbs for minimizing the complexity and amplifying the efficiency of the algorithm. The number of the features that are needed by a verb depend on the definition of the verb. In normal cases, only one or two features are attached to a verb.

B. THE CV TEMPLATES

In CV, there are many different template functions for verbs and adverbs. A classical computational verb template is formulated as:

$$\mathrm{V} = \mathrm{w_2 t^2} + \mathrm{w_1 t} + \mathrm{w_0} \tag{4}$$

In Equation (4), t represents a time variable. The starting point is $\mathrm{t} = 0$, and $\Delta\,\mathrm{t} = 1$ is set as time step. The values of the parameters, $\mathrm{w_0}$, $\mathrm{w_1}$ and $\mathrm{w_2}$,are determined by a learning algorithm automatically. Different verbs can be deployed to construct a set of rules. For observations $\mathrm{X_1}$ and $\mathrm{X_2}$, let Y be the expected output. Then a typical instance can be denoted as:

$$\text{If} \mathrm{X_1} \textit{slowly increase to} \mathrm{V_{x1}}, \text{ and } \mathrm{X_2} \textit{ fast become } \mathrm{V_{x2}}, \text{ then } \mathrm{Y} \textit{stay} \mathrm{V_Y} \tag{5}$$

The words *fast* and *slowly* are computational adverbs, which should be prescribed in the context. Furthermore, Tao [16] demonstrated different templates recommendation for adverbs. As different templates may lead to diverse results, it is necessary to pay attention to the choice of templates for verbs and adverbs.

By taking the evolving process into consideration, a CV can be written as:

$$\mathrm{Y_{new}} = f(\mathrm{s(X, V_x)}, \mathrm{V_Y}, \mathrm{Y_{current}}) \tag{6}$$

where $\mathrm{s(X, V_x)}$ is the similarity function, used to calculate the diversity between X and $\mathrm{V_x}$. The function f represents a preset function, which acts an approach in order to update $\mathrm{Y_{new}}$, depending on the similarity between $\mathrm{V_Y}$ and $\mathrm{Y_{current}}$. So this formula illustrates the relationships among $\mathrm{Y_{new}}$, $\mathrm{Y_{current}}$ and $\mathrm{V_Y}$.

C. THE EVOLVING FUNCTIONS FOR CV

For CV, an evolving function defines an orbit in a dynamical system. It is capable to illustrate the change of an action within a time span. An evolving function is produced based on training data. An example is the verb *increase* (V_{increase}), which can be molded as a 4-tuple (*increase*; $\mathrm{R^+}$; R; ε). ε is an evolving function, and its general form as shown in Equation (7).

$$\varepsilon(\mathrm{t; x}) = \mathrm{t} + \mathrm{x}, \quad \mathrm{t} \in \mathrm{R^+} \tag{7}$$

In most case, R refers to the real number field, so there are numerous possible orbits for such a simple dynamical system with various initial states. As reported in [16], if x is set to 1, the simplest evolving function for *increase* can be obtained and expressed as:

$$\varepsilon_{\text{increase}} = 1 + \mathrm{t}; \quad \mathrm{t} \in \mathrm{R^+} \tag{8}$$

On the other hand, the representation of the evolving functions is different for different verbs according to the definition of verbs. For instance, the typical evolving function for verb *stay* is formulated as:

$$\varepsilon_{\text{stay}} = |1 + t| < \delta; \quad t \in R^+ \tag{9}$$

where δ indicates the expected deviation.

D. THE SIMILARITY MEASUREMENT FUNCTION

The goal of similarity measurement function is to evaluate the diversity between two verbs, so each similarity measurement function is derived from their evolving functions. Concretely, let E represent a set of evolving functions. Given two evolving functions, $\varepsilon_1, \varepsilon_2 \in E$, for the two verbs, the similarity is computed based on a similarity function s, $s : E^2 \rightarrow R^+$. Each function takes value ranging in [0, 1], where 0 indicates completely different behaviors, and 1 refers to the same action. In such case, the function should satisfy several conditions, as listed below:

$$s(\varepsilon 1, \varepsilon 2) = s(\varepsilon 2, \varepsilon 1), \forall \varepsilon 1, \varepsilon 2 \in E \tag{10}$$

$$s(\varepsilon 1, \varepsilon 1) = 1, \forall \varepsilon 1 \in E \tag{11}$$

$$\text{if } \varepsilon 1(t)\varepsilon(t) \equiv 0, \varepsilon 1(t) + \varepsilon 2(t) \equiv 1, \forall t \in T, \text{ then } s(\varepsilon 1, \varepsilon 2) = 0 \tag{12}$$

$$\forall \varepsilon 1, \varepsilon 2, \varepsilon 3 \in E, \text{ if } \forall \varepsilon 1 \leq \varepsilon 2 \leq \varepsilon 3 \text{ then } s(\varepsilon 1, \varepsilon 2) \leq s(\varepsilon 1, \varepsilon 3) \text{ and } s(\varepsilon 2, \varepsilon 3) \leq s(\varepsilon 1, \varepsilon 3) \tag{13}$$

Conditions (10)-(13) reflect some properties of similarity functions. Formula (10) indicates that each similarity function is symmetric, that is, the similarity degree from ε_1 to ε_2 is the same as that from ε_2 to ε_1.

Succinctly, the similarity function is adopted to distinguish two states of a verb (i.e., V_x and X) or two verbs (i.e., V_1 and V_2). The definition of a verb similarity function is highly dependable on each problem. Thus far, there are three types of verb similarity functions based on different principles: distance, trend and frequency. As trend and frequency based similarity functions are mainly designed for time serials analysis, in this work, only the distance based similarity function is deployed to check the distance between two verbs based on s_d (as shown in Equation (14)) within the time interval [0, T]. Concisely, in Equation (14), s_d sums up the square of point-by-point amplitude difference for ε_1 and ε_2 in [0, T]. The function $g(x)$ maps the outputs within the range of [0, 1], as shown in Equation (15).

$$s_d(\varepsilon_1, \varepsilon_2) = g(\sum_{t=0}^{T} |\varepsilon_1(t) - \varepsilon_2(t)|^2) \tag{14}$$

$$g(x) = \frac{2}{1 + e^x} \tag{15}$$

E. CVR BASED CANCER CLASSIFICATION

With the representation of canonical form, the template function is can be modeled to implicit equations and hence making the learning process easier. Concisely, both the inputs and the template functions can be expressed by using a computational verb (v, R^+, R, ε). Two genes are viewed as two tuples in the timeline [0, T]. A computational verb rule is employed to compare the variation of the expression levels between them to handle a binary-class problem. As such, a sample is distinguished based on the expression levels of two genes. The similarity function is used to calculate the probability of a sample falls into a certain class.

The computational verb *stay* is adopted to portray the states to model the gene data. Such computational verb rules can be expressed as:

$$\text{If Gene } i \text{ stay relatively inactive to Gene } j, \text{ then class 1;} \tag{16}$$

$$\text{If Gene } i \text{ stay active to Gene } j, \text{ then class 2;} \tag{17}$$

The rule proposed in [20] is based on the verb *increase*. Unlike it, the verb *stay* used in our algorithm can describe the accumulate effect among a dataset. Although both types of rules are based on a gene pair, the rules based on *increase* would be affected by the input sequence, while ours is independent of it. And it is key to the application of CVR to such a classification task. The usage of the words *active* and *inactive* are utilized to describe the gene expression status. This makes our rules are more illustrative and meaningful to the researchers to interpret the results.

Basically, the verb used in the SCVR algorithm is referred to the template suggested in [16]. The similarity of each rule is quantified between the input and the antecedents of these rules. The final decisions are drawn on the basis of the similarity of the rules' output.

It is assumed that *stay active* is mathematically formulated as V_{1i}/V_{1j}, and *stay inactive* is represented by V_{2i}/V_{2j} for the gene i/j. The expression level of gene i/j for sample x is expressed using x_i/x_j. Assume n samples are contained in a dataset, and S_1 and S_2 represent the results of the antecedents of the rules defined in Equation (16) and (17), respectively. Then, S_1 and S_2 can be computed by combining the distance based similarity function, as shown in Equations (18) and (19), respectively.

$$\begin{aligned} S_1 &= s_{d1}(V_{1i}, V_{1j}) = g[\sum\nolimits_n (k_1(|x_i - G_{i1}| - \sigma_1) \\ &\quad - (|x_j - G_{j1}| - \sigma_2))^2] \\ &= g[\sum\nolimits_n (k_1(|x_i - G_{i1}| - |x_j - G_{j1}| - w_1))^2] \end{aligned} \tag{18}$$

$$\begin{aligned} S_2 &= s_{d2}(V_{2i}, V_{2j}) = g[\sum\nolimits_n (k_2(|x_i - G_{i2}| - \sigma_3) \\ &\quad - (|x_j - G_{j2}| - \sigma_4))^2] \\ &= g[\sum\nolimits_n (k_2 |x_i - G_{i2}| - |x_j - G_{j2}| - w_2)^2] \end{aligned} \tag{19}$$

where $\sigma_{1,}\sigma_{2,}\sigma_3$ and σ_1 are the parameters used to adjust the rules to better fit the data distribution. Moreover, two new parameters (i.e., w_1 and w_2) are introduced to simplify the computational process, where $w_1 = k_1 \times \sigma_1 + \sigma_2$ and $w_2 = k_2 \times \sigma_3 + \sigma_4$.

The final decision for a sample is made by assigning the sample to the class with higher probability. In our proposed framework, it is observed that the larger gap between *active* or *inactive* status in a gene pair, the better the generalizability of the final output, and hence the more powerful SCVR classifier.

The verb template defined in Equation (5) is exploited to construct the rules along with the evolving function (i.e., *stay*). To validate the status of a gene, the parameters G_{i1}/G_{j1} and G_{i2}/G_{j2} are used to determine whether gene i's or j's expression level is relatively high (i.e., *active*) or low (i.e., *inactive*). The parameters are determined entirely by the original data distribution. Here, a gene i's average expression values in two classes are calculated firstly. The larger mean value is assigned to G_{i1} to represent the threshold of *active*, and the smaller mean value is assigned to G_{i2} as *inactive* status. The difference between these values affects the discriminative power of the final rules. It is obvious that the performance of final rules relies on the differentiation of the *active/ inactive* comparison levels in the selected gene pair. The larger difference between two statuses in a gene pair, the higher generalization ability of the produced SCVR.

In Equations (16) and (17), the words r*elatively to* is used to measure the degree of *active/inactive*. To better compare the difference in two statuses, k_1 and k_2 in Equations (18)

Step 1: For each gene i, the mean expression value of each class is computed. Then the mean value for the first class is appointed to G_{i1}, and the value for the second class is assigned to G_{i2}. Next, the differentiation value (DV_i) for i is calculated by Equation (25):

$$DV_i=|G_{i1} - G_{i2}| \tag{25}$$

The calculation is iterated until all genes are verified.
Step 2: The genes are sorted according to DV in the descending order, and the top 100 genes are chosen to construct a candidate pool.
Step 3: A gene pair is formed by selecting the top gene i in the pool. Next, the selected gene pair is matched with a gene j that can produce the largest value $D_{i,j}$ according to Equation (26). The genes i and j are then eliminated from the pool.

$$DV_{i,j}=\max(|G_{i1} - G_{j2}|+|G_{i2} - G_{j1}|,|G_{i1} - G_{j1}|+|G_{i2} - G_{j2}|) \tag{26}$$

Step 4: Step 3 is repeated until the pool is empty.
Step 5: Each gene pair is picked up to form a computational verb rule according to Equations (18) and (19). The four parameters (i.e., k_1/k_2 and W_1/W_2) are optimized.
Step 6: Top N trained SCVR are picked to categorize the testing samples using majority vote approach.

FIGURE 1. The workflow of the SCVR algorithm.

and (19) are used to implement the adverb *relatively to* in the rules. Such r*elatively to* relationship can be *more* if k_1 or k_2 is larger than 1, and *less* otherwise. As such, a rule is capable to provide a more accurate and worthwhile comparison for a gene pair. The rule is written as: If Gene i *stay more active compared with* Gene j, then class 1. Next, the gradient descent methodology is adopted to optimize these four parameters, k_1/k_2 and W_1/W_2. In this way, the SCVR algorithm can be optimized to make rules matching the training data better.

To summarize, the workflow of algorithm SCVR is shown in Figure 1. Step 2 to step 4 attempt to remove redundant and meaningless genes, then quickly match the dominant and remarkable genes to construct gene pairs with a greedy approach. Next, in step 5, the gene pairs are combined to produce SCVR rules. Finally, step 6 keeps the top N gene pairs to form the final ensemble. This approach is capable to generate rules based on non-overlapping gene pairs, so that the diversity in this ensemble can be guaranteed. To validate the effectiveness of our proposed method, N is set to 5 in our experiment. Thus, 10 genes will be employed to produce the results in all the experiments.

The optimization of parameters in step 5 is implemented using a gradient descent algorithm. In details, let P represents four parameters (i.e., k_1, k_2, W_1, W_2). Let F_i represent the feature vector for sample x_i, and its label is l_i. The probability of sample x_i belonging to class 1/2 can be calculated by Equations (18) and (19):

$$p(l = 1|x, P) = S_1/(S_1 + S_2) \tag{20}$$

$$p(l = 2|x, P) = S_2/(S_1 + S_2) \tag{21}$$

The parameter vector P can be optimized based on the maximizing likelihood. The loss function takes the form of negative log of likelihood P, as shown in Equations (22).

$$\begin{aligned} \mathrm{l(P)} &= -\log\prod_{i=1}^{n} p(y_i|x_i, P) \\ &= -\log\prod_{i=1}^{n}(\frac{s1_i}{s1_i + s2_i})^{y_i}(1 - \frac{s1_i}{s1_i + s2_i})^{1-y_i} \\ &= -\sum\nolimits_{i=1}^{n}(y_i \log(\frac{s1_i}{s1_i + s2_i}) \\ &\quad + (1 - y_i)\log(1 - \frac{s1_i}{s1_i + s2_i})) \end{aligned} \tag{22}$$

By taking the partial derivative with respect to P_i, Equation (22) turns to:

$$\begin{aligned} \frac{\partial}{\partial P_i} l(P) = -\sum\nolimits_{i=1}^{n}(y_i(\frac{s1_i + s2_i}{s1_i}) \\ -(1 - y_i)\frac{s1_i + s2_i}{s1_i}))\frac{\partial}{\partial P_i}(\frac{s1_i}{s1_i + s2_i}) \end{aligned} \tag{23}$$

The result of Equation (23) is closely related with the definition of similarity function. To simplify our discussion, gradient descent is deployed to optimize parameters. That is, P_i is updated by:

$$P_i^{'} = P_i - \alpha\frac{\partial}{\partial P_i} l(P) \tag{24}$$

III. RESULTS AND DISCUSSION

A. THE SETTINGS OF EXPERIMENTS

The experiments are conducted on five binary class and six multiclass microarray datasets. The details of these datasets, including the number of the genes and the number of samples, are tabulated in Table 1 and Table 2. All the samples in the datasets are utilized in the experiments. To avoid data dependent issue in the classification process, all the samples are assigned to either the training or testing sets. Following the steps implemented in [12], the datasets are undergoing pre-process stage before applying our proposed method. Concretely, all the raw data are first converted to natural logarithmic values, then each sample is normalized to zero mean and unit variance for standardization and to facilitate the subsequent classification process. In addition to the original partition for the binary class task, each dataset is reshuffled with 9 randomizations. Note that, each randomization comprises the same sample distribution in each class of the original training and testing sets.

In all the experiments, classifiers are constructed solely dependent on the training samples. Then the classification results are predicted using the independent testing set. The data normalization and the feature selection operations are applied in all the classifiers. Naturally, all the classifiers will be having the same suppressed subset of features in certain training and testing datasets.

TABLE 1. Five Binary class datasets Deployed in this study.

Datasets	Number of	Number of samples of two	Reference
Ovarian	15154	162/91	[21]
Colon	20000	40/22	[22]
Leukemia	7129	47/25	[2]
Lung	12533	150/31	[23]
Prostate	12600	77/59	[24]

TABLE 2. Six Multiclass datasets deployed in this study.

Dataset	Number of classes	Number of genes	Number of training/test samples	Reference
Leukemia1	3	7129	38/34	[2]
Leukemia2	3	12,582	57/15	[25]
Lung1	3	7129	64/32	[26]
Lung2	5	12,600	136/67	[27]
Breast	5	9216	54/30	[28]
DLBCL	6	4026	58/30	[29]

Decision tree is a common rule-based learner due to its robustness against noise, ease in generating rules and impressive computational simplicity and efficiency. As such, we decide to compare our proposed algorithm performance to decision tree and some other well-known tree based methods, such as RF and Rotation Forest. The implementations in scikit-learn library are exploited [29] for decision tree (DT) and RF (RF). For Rotation Forest, an improved Rotation Forest algorithm, named Hybrid Extreme Rotation Forest (HERF) [30], is employed. The parameters of the classifiers are set to default values, accordingly. And SCVR is implemented on Python 2.7 platform.

In our algorithm, the top five CV rules are considered to generate outputs for each two-class problem. As such, the algorithm is treated as a small-size ensemble classifier. Nevertheless, since there are only ten genes used, the scale of the ensemble CV rules is smaller even when comparing with using decision trees. Note that, there will be no duplicate genes occurred in constructing the rules. Thus, the diversity of the final ensemble can be assured.

To validate the effectiveness of the proposed framework, different measures are employed, following the recommendation in [31]. Specifically, classification accuracy denotes the percentage of correctly classified samples. Furthermore, to tackle the imbalance data distribution, *F-score* and MCC (Mathews Correlation Coefficient) are used to measure the recognition performance of the proposed method. *Recall* and *precision* are the elements used in calculating *F-score*, where *recall* (exactness) is the ratio of the relevant information extracted by the system to the total number of relevant records in the database, and *precision* (completeness) is the measure of how much information in the system is returned correctly. The equations of these three indicators are set out as follows:

$$Precision = \frac{tp}{(tp + fp)} \tag{27}$$

$$Recall = \frac{tp}{(tp + fn)} \tag{28}$$

$$\mathrm{F-score} = \frac{2 * Precision \times Recall}{(Precision + Recall)} \tag{29}$$

$$\mathrm{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn} \tag{30}$$

$$MCC = \frac{tp \times tn - fp \times fn}{\sqrt{(tp + fp)(tp + fn)(tn + fp)(tn + fn)}} \tag{31}$$

where tp, tn, fn and fp are true positive, true negative, false negative and false positive, respectively. MCC was proposed in [32], mainly used as a balanced measure even for classes with very different sizes. MCC returns a value within $[-1, +1]$. $+1$ represents a perfect prediction, and -1 shows a total disagreement between prediction and observation, while 0 refers to random prediction.

For a m-class classification problem, different class labels are represented as $1, 2, \ldots, m$. SCVR can only solve binary-class problems by classifying the data into Yes/ No. Hence, we consider to treat a multiclass problem with two commonly used decomposition methods: One vs. One (OVO) and One vs. Rest (OVR). In this way, we can safely handle a multiclass problem by using SCVR as base learners. To establish fair comparison among the experiments carried out in different parameter settings, the classifiers such as decision tree, RF and Rotation Forest methods are also used as binary classifiers, fused with OVO and OVR methods.

For the multiclass problems, experiments are performed entirely on the original splits. This is because both the RF and HERF are based on random division on the sample sets and both of the classifiers execute ten times with random seeds. Therefore, the average performance can be used in describing the generalization ability.

$$\mathrm{Precision}_{\mu} = \frac{\sum_{i=1}^{c} tp_i}{\sum_{i=1}^{c} (tp_i + fp_i)} \tag{32}$$

$$\mathrm{Recall}_{\mu} = \frac{\sum_{i=1}^{c} tp_i}{\sum_{i=1}^{c} (tp_i + fn_i)} \tag{33}$$

$$\mathrm{Fscore}_{\mu} = \frac{(\beta^2 + 1)\mathrm{Precision}_{\mu}\mathrm{Recall}_{\mu}}{\beta^2 \mathrm{Precision}_{\mu} + \mathrm{Recall}_{\mu}} \tag{34}$$

$$AAc_{\mu} = \frac{\sum_{i=1}^{c} \frac{(tp_i + tn_i)}{(tp_i + tn_i + fp_i + fn_i)}}{c} \tag{35}$$

$$t_k = \sum_{i=1}^{c} C_{ik} \tag{36}$$

$$p_k = \sum_{i=1}^{c} C_{ki} \tag{37}$$

$$c = \sum_{k=1}^{c} C_{kk} \tag{38}$$

$$\mathrm{s} = \sum_{i=1}^{c} \sum_{j=1}^{c} C_{ij} \tag{39}$$

$$MCC = \frac{c \times s - \sum_{k=1}^{c} p_k \times t_k}{\sqrt{(s^2 - \sum_{k=1}^{c} p_k^2) \times (s^2 - \sum_{k=1}^{c} t_k^2)}} \tag{40}$$

As mentioned earlier, the two performance measurements, $F\text{-}score_{\mu}$ and *Average Accuracy* (*AAc* for short), are utilized for results comparisons. Concisely, assume that there are c classes in a dataset, and these two measurements can be com-

TABLE 3. Experimental results for binary class datasets.

	AVERAGE	Ovarian	Colon	Leukemia	Lung	Prostate	Average
SCVR	F-score	**0.968±0.000**	**0.824±0.024**	**0.946±0.007**	**0.947±0.000**	0.827±0.031	**0.903±0.012**
	Accuracy	**0.976±0.000**	0.843±0.000	**0.949±0.005**	**0.983±0.000**	0.815±0.000	0.913±0.001
	MCC	**0.943±0.000**	**0.803±0.005**	**0.912±0.009**	**0.892±0.000**	0.792±0.052	**0.881±0.017**
DT	F-score	0.967±0.009	0.743±0.053	0.844±0.052	0.891±0.022	0.842±0.017	0.857±0.031
	Accuracy	0.975±0.007	0.799±0.040	0.878±0.033	0.961±0.008	0.864±0.009	0.895±0.019
	MCC	0.941±0.012	0.694±0.075	0.812±0.067	0.842±0.035	0.801±0.022	0.812±0.047
RF	F-score	0.966±0.009	0.743±0.053	0.844±0.052	0.891±0.022	0.842±0.017	0.857±0.031
	Accuracy	0.969±0.003	0.835±0.004	0.944±0.017	0.918±0.04	0.864±0.009	0.906±0.015
	MCC	0.923±0.011	0.701±0.055	0.808±0.062	0.842±0.035	0.801±0.022	0.810±0.045
HERF	F-score	0.960±0.005	0.816±0.027	0.925±0.015	0.936±0.027	**0.858±0.043**	0.899±0.023
	Accuracy	0.956±0.048	**0.862±0.045**	0.937±0.012	0.974±0.009	**0.866±0.040**	**0.919±0.031**
	MCC	0.937±0.051	0.783±0.052	0.902±0.029	0.872±0.038	**0.813±0.042**	0.878±0.037

puted by applying the Equations (32-35). Unlike the accuracy measurement, *AAc* denotes the average per-class performance of a classifier. For example, if a classifier fails to recognize samples in a hard class, it would be not able to attain high scores in *AAc*. $F\text{-}score_{\mu}$ is a measurement that combines the scores from both the *precision* and *recall* among all the classes. To obtain a balance between *precision* and *recall*, β is set to 1 in the Equation (34).

Equation (36-39) defines some intermediate variables, where C_{ij} the elements in the i-th row and j-th column in the confusion matrix. So t_k represents the times that class k occurs; p_k represents the times that k is predicted; c represents the total number of samples correctly predicted; s represents the total number of samples. Equation (40) gives the MCC measure in the case of multiclass. The minimum MCC value changes within the range of $[-1, 0]$, depending on the true distribution; but the maximum value is always $+1$.

B. EXPERIMENTAL RESULTS FOR BINARY DATASETS

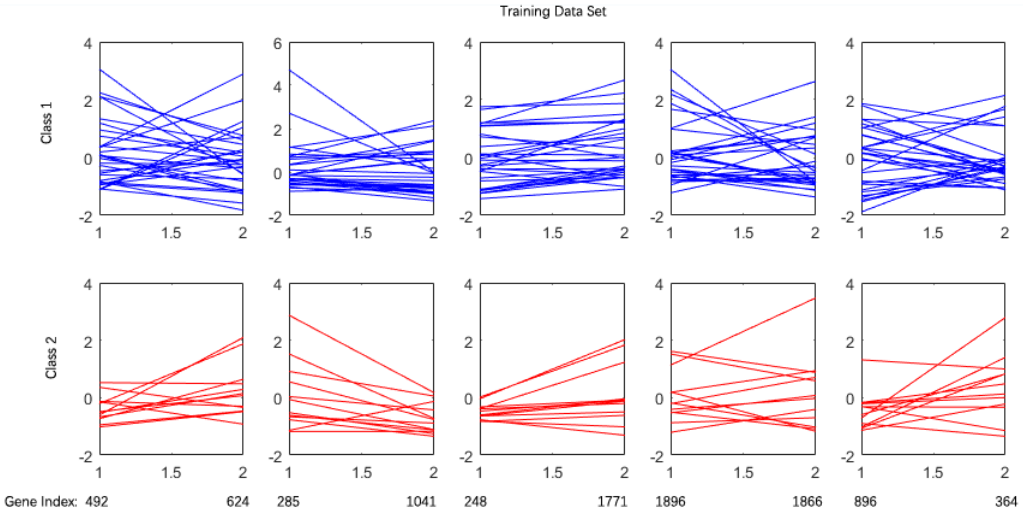
From Table 3, it is observed that SCVR can outperform in four out of five datasets, and exhibits the best accuracy in three out of five datasets. Hence, the effectiveness of SCVR algorithm is confirmed in these experiments. Additionally, it completely beats DT and RF in these experiments by achieving higher F-score and MCC indices. It is demonstrated that our SCVR algorithm has the capability to overcome the sample-imbalanced problem by generating the best overall average F-score and MCC scores.

The ability of the generalization of SCVR algorithm may be further verified by the nonlinear projection function, $g(x)$, as stated in Equation (14). To demonstrate the reliability of SCVR, the experiments based on the colon dataset can be used as an example. Figure 2 illustrates the line chart of the top five gene pairs selected for forming SCVR in a loop of a 10-fold cross validation. Concisely, the gene indices are: ([492, 624], [285, 1041], [248, 1771], [1896, 1866], [896, 364]). Details about these genes are elaborated in Table 4. Figure 2(a) and Figure 2(b) visualize the expression value of five gene pair in both training and testing sets. Each sub-figure represents a gene pair. That is, two coordinates in x-axis, 1 and 2, represents two genes with the gene index marked at the bottom. The y-axis value represents their expression values. And each line stands for a sample, connecting the expression values of its gene pair. From Fig. 2, it is discovered that the data distribution in different classes is quite different. So a sample is relatively active in a class, and inactive in another. The difference between two classes reinforces the discriminative ability of SCVR.

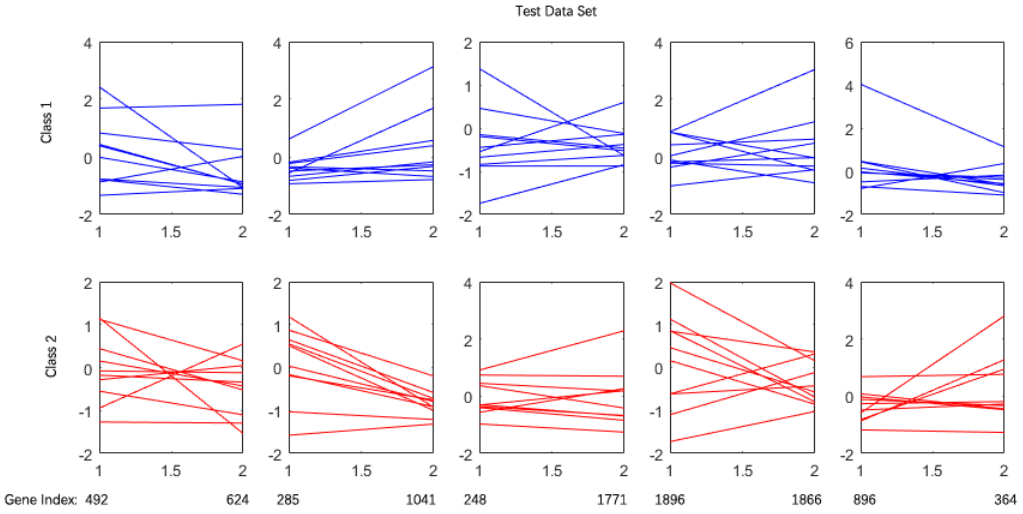
In our algorithm, the genes are used to form different rules in pairs. The set of the rule obtained by the gene pair [R87126, T51250] can be molded to:

- If R87126*is more active than* T51250, then *normal*;
- If R87126 is *more inactive than* T51250, then *colon cancer*;

Similar rules can also be extracted from other gene pairs. Figure 3 show the training samples projection based on the first gene pair, in which the blue cross and red o denote the cancer and normal classes respectively. From Figure 3(a), it can be seen that it is not possible to draw a linear decision boundary in the original datasets because of the complex class distributions. However, SCVR still can provide a good classification result for the training samples by projecting the original data to a linear subspace. By using the final output of the trained SCVR, samples can be better classified with only 3 samples misclassified based on a linear boundary line, as shown in Figure 3(b). Furthermore, a perfect 100% correct classification result can be obtained for the test samples only in the condition of combining the decision of three SCVRs.

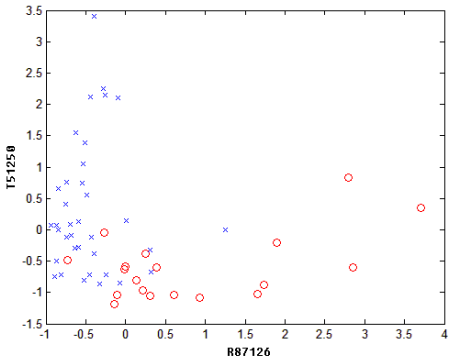


(a) Five gene pairs in training data set

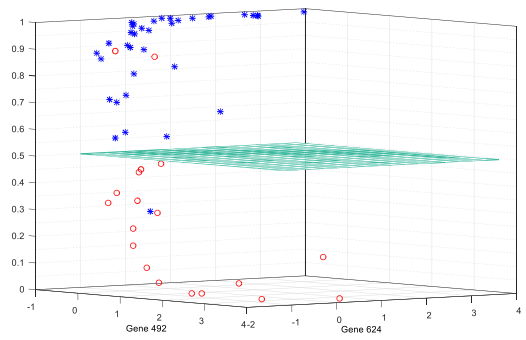


(b) Five gene pairs in test data set

FIGURE 2. **The top five gene pairs selected for the colon dataset.**



(a) The original projection of samples



(b) The outputs of SCVR for each sample as the third dimension (the decision plane is z=0.5)

FIGURE 3. **The projection of training samples for the colon dataset with the gene pair: R87126 and T51250.**

From the results, is it observed that there is significant difference within each gene pair. Concretely, when a gene takes relatively high expression values, another gene will take relative low values. The detailed information of some of the selected genes is reported in Table 4.

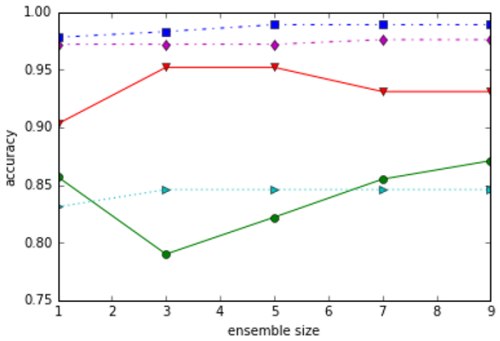
The biological functions of the 5 gene pairs are checked in NCBI. Among these genes, some of them were used to build classification models in some previous publications. For example, R87126, J05032 are identified in [29], [30] as important genes in classifying the colon cancer from the normal class. Some other genes are discovered to be transcription factors or repressors, and a typical example is Pim-1 kinase (M27903), which has been proved to participate in the important biological processes [31]. These genes may be potentially important factors in the oncogenesis, which require further investigation in the future.

C. EXPERIMENTAL RESULTS FOR VARYING ENSEMBLE SIZE

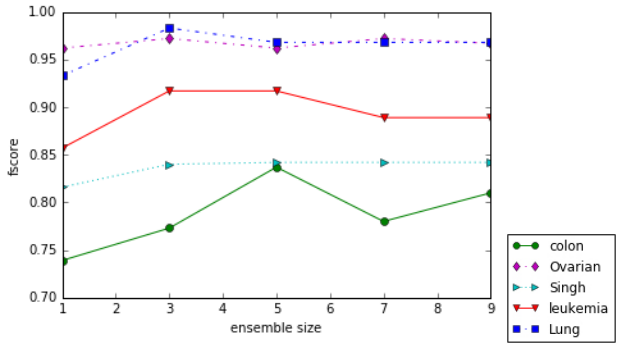
More experiments are carried out to justify the performance of SCVR algorithm by applying different ensemble sizes, as shown in Figure 4. From these figures, it is observed that the results are stable when increasing ensemble size on the Lung, Singh and Ovarian datasets. Yet for the Colon datasets, the results change a lot with the change of ensemble size. With further exploration, it is revealed that most rules can obtain around 95% accuracy on the Lung and the Ovarian datasets, as shown in Figure 4(a). So the outputs of different rules are quite similar, and the fusion of such accurate rules can produce stable results. While in the Singh datasets, there are some hard samples in test set, causing the top rules fail to recognize them correctly. In both cases, the combination of more rules does not benefit the results. For the Colon dataset, the accuracy of one rule can reach 0.857, but drop to 0.790 when adding two more rules. On the contrary, in Figure 4(b), the *F-score* rises from 0.739 to 0.773 at the same time. When increasing the ensemble size to 5, the *F-score* continues to raise, and the accuracy begins to recover. However, adding some more rules to the ensemble is not able to improve the *F-score* anymore. The reason being is that the rules with lower ranked are not as accurate as those added to the ensemble formerly, and add more rules may even deteriorate the performance of SCVR. In general, from the results, it is concluded that the fusion of five rules can produce satisfactory results, but it is not necessary to enlarge the ensemble scale. Hence, the ensemble size is set to 5 as a trade-off to filter inaccurate rules.

D. EXPERIMENTAL RESULTS FOR MULTICLASS DATASETS

The results based on multiclass datasets are listed in Table 5, in which it can be seen that SCVR based classification results also take advantages in most cases for the multiclass problem from Table 4. In considering F-score, *AAC* and MCC results, OVR based SCVR outperforms other methods in three out of six datasets, and is able to obtain the highest scores in average performance. On the other hand, OVO based SCVR exhibits promising results when tested in Leukemia1 dataset. For other datasets, only RF outperforms in DLBCL dataset. As a result, SCVR method can take obvious advantage with a much smaller ensemble size compared to other classifiers.



(a) accuracy VS Ensemble size



(b) Fscore VS Ensemble size

FIGURE 4. The results of the changing ensemble size on different datasets.

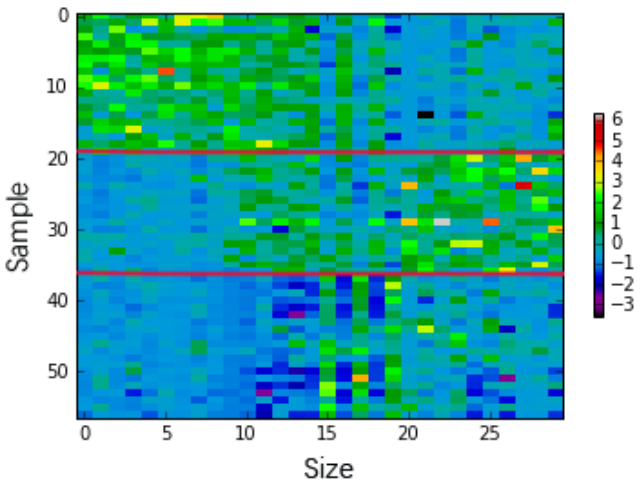


FIGURE 5. The heatmap of the selected gene pairs for the Leukemia2 dataset.

It is observed that OVR based SCVR scheme achieves 100% accuracy in the Leukemia2 dataset, so some investigation is conducted to further study the results. For such a three-class problem, as five gene pairs are required for each binary class problem, there are fifth gene pairs are required for the construction of OVR scheme. Figure 5 illustrates the 15 gene pairs used in this scheme. Three classes in Leukumia2 are divided by two red lines.

TABLE 4. The ten genes selected by SCVR.

Index	Gene id	Accession numbers	Description
492	Has.37937	R87126	MYOSIN HEAVY CHAIN, NONMUSCLE (Gallus gallus)
624	Has.5402	T51250	CYTOCHROME C OXIDASE POLYPEPTIDE VIII-LIVER/HEART (HUMAN)
285	Has.20034	T99498	SERINE/THREONINE-PROTEIN KINASE PAK (Rattus norvegicus)
1041	Has.27930	R54467	STEROID RECEPTOR TR2 (Homo sapiens)
248	Has.8177	R16255	SERINE/THREONINE PROTEIN PHOSPHATASE 2B CATALYTIC SUBUNIT 2 (Homo sapiens)
1771	Has.601	J05032	Human aspartyl-tRNA synthetase alpha-2 subunit mRNA, complete cds.
1896	Has.636	M76558	Human neuronal DHP-sensitive, voltage-dependent, calcium channel alpha-1D subunit mRNA, complete cds.
1866	Has.14102	T67173	RETINOIC ACID RECEPTOR RXR-BETA ISOFORM 2 (Homo sapiens)
896	Has.13023	H09149	ASIALOGLYCOPROTEIN RECEPTOR R2/3 (Rattus norvegicus)
364	Has.995	M27903	Human pim-1 proto-oncogene gene, complete cds

TABLE 5. Experimental results for multiclass datasets.

Datasets		OVO				OVR			
		SCVR	DT	RF	HERF	SCVR	DT	RF	HERF
Leukemia1	$Fscore_{\mu}$	**0.941**	**0.941**	0.881 ± 0.061	0.923 ± 0.074	0.794	0.882	0.917 ± 0.012	0.941 ± 0.019
	AAc	**0.961**	0.960	0.918 ± 0.040	0.949 ± 0.049	0.863	0.922	0.944 ± 0.008	0.951 ± 0.012
	MCC	**0.924**	0.922	0.852 ± 0.071	0.901 ± 0.095	0.787	0.854	0.891 ± 0.021	0.902 ± 0.029
Leukemia2	$Fscore_{\mu}$	0.933	0.733	0.934 ± 0.066	0.956 ± 0.011	**1.000**	0.773	0.921 ± 0.050	0.925 ± 0.026
	AAc	0.956	0.778	0.953 ± 0.043	0.997 ± 0.008	**1.000**	0.882	0.947 ± 0.033	0.940 ± 0.017
	MCC	0.937	0.721	0.899 ± 0.068	0.922 ± 0.025	**1.000**	0.784	0.912 ± 0.057	0.915 ± 0.031
Breast	$Fscore_{\mu}$	0.833	0.700	0.888 ± 0.027	0.860 ± 0.037	**0.933**	0.733	0.844 ± 0.027	0.873 ± 0.037
	AAc	0.933	0.880	0.961 ± 0.002	0.946 ± 0.012	**0.973**	0.893	0.949 ± 0.017	0.942 ± 0.015
	MCC	0.841	0.685	0.862 ± 0.037	0.817 ± 0.043	**0.881**	0.704	0.816 ± 0.038	0.852 ± 0.051
DLBCL	$Fscore_{\mu}$	0.833	0.833	**0.935 ± 0.055**	0.837 ± 0.007	0.900	0.833	0.815 ± 0.071	0.803 ± 0.071
	AAc	0.944	0.944	**0.972 ± 0.018**	0.925 ± 0.024	0.967	0.944	0.937 ± 0.023	0.923 ± 0.023
	MCC	0.801	0.801	**0.915 ± 0.073**	0.823 ± 0.035	0.894	0.814	0.792 ± 0.067	0.787 ± 0.097
Lung1	$Fscore_{\mu}$	0.813	0.668	0.791 ± 0.017	0.817 ± 0.013	0.813	0.781	0.771 ± 0.025	**0.818 ± 0.036**
	AAc	**0.875**	0.792	0.861 ± 0.011	0.869 ± 0.009	**0.875**	0.854	0.847 ± 0.017	0.868 ± 0.026
	MCC_{μ}	0.782	0.624	0.763 ± 0.032	0.784 ± 0.026	0.782	0.725	0.712 ± 0.031	**0.786 ± 0.041**
Lung2	$Fscore_{\mu}$	0.836	0.955	0.950 ± 0.008	0.933 ± 0.013	**0.970**	0.851	0.908 ± 0.014	0.946 ± 0.017
	AAc	0.934	0.982	0.980 ± 0.031	0.965 ± 0.009	**0.988**	0.940	0.963 ± 0.006	0.964 ± 0.013
	MCC	0.810	0.904	0.912 ± 0.016	0.882 ± 0.036	**0.953**	0.814	0.882 ± 0.017	0.912 ± 0.021
Average	$Fscore_{\mu}$	0.865	0.805	0.897 ± 0.039	0.888± 0.026	**0.902**	0.809	0.863 ± 0.033	0.884± 0.034
	AAc	0.934	0.889	0.941 ± 0.024	0.942 ± 0.018	**0.944**	0.906	0.878 ± 0.055	0.931 ± 0.017
	MCC	0.849	0.776	0.867 ± 0.050	0.855 ± 0.043	**0.882**	0.783	0.834 ± 0.039	0.859 ± 0.045

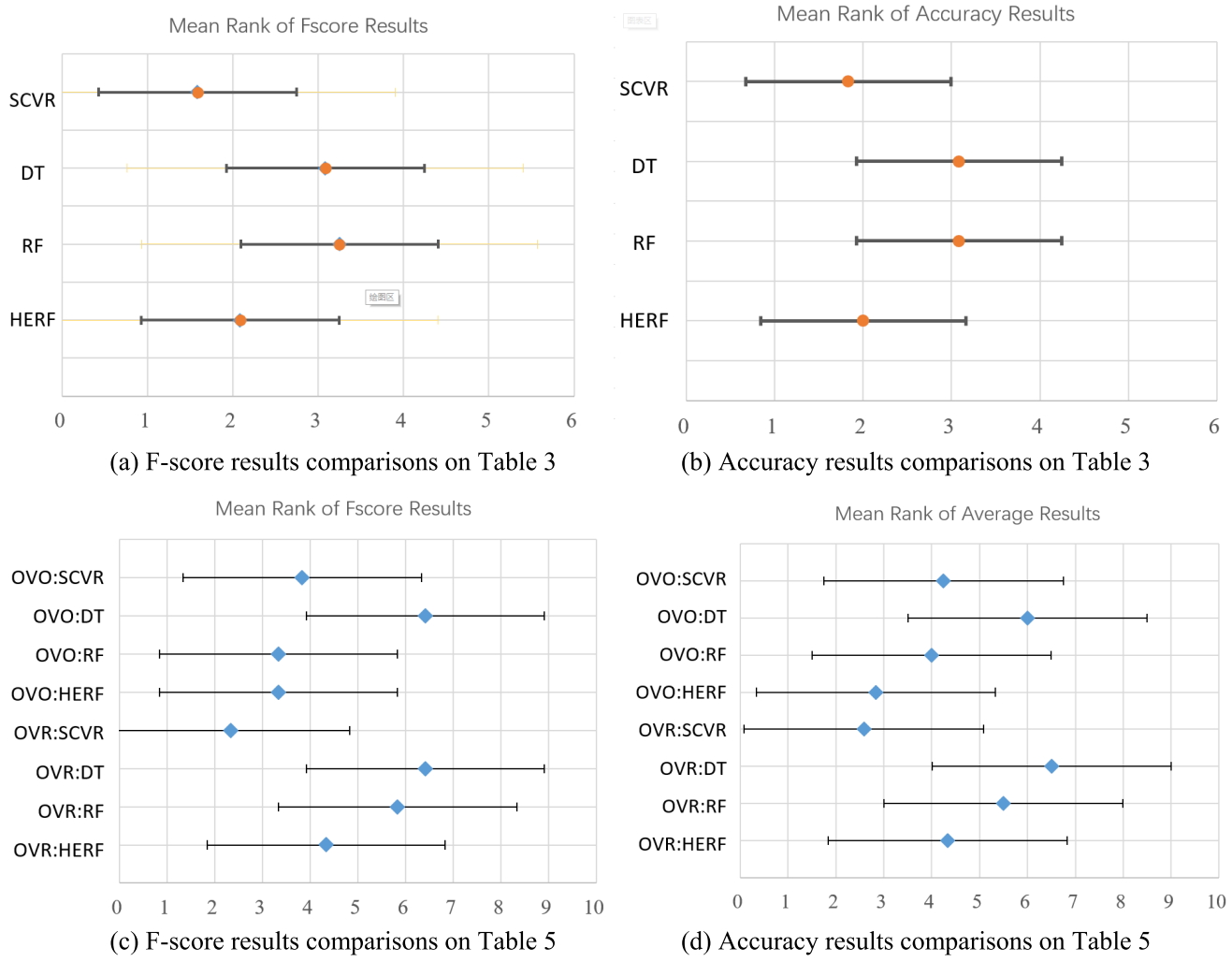
The results can verify the success of our algorithm because the gene expression levels of each of the five gene pairs are quite different in each binary class case. Such results could also become the reference for clinical treatment.

The 30 selected genes are reported in Table 6, along with their names and description. It is found that all of the selected genes are well studied by other researchers. For example, MAPKAPK3 was demonstrated to play an important role in transferring the aberrant signaling from

TABLE 6. The selected 15 gene pairs for the leukemia2 dataset using ovsr.

Index	Access number	Gene name	Description
3767	39930_at	EPHB6	Hs.3796 gnl|UG|Hs#S816466 Homo sapiens mRNA for Eph-family protein, complete cds
10720	1991_s_at	MAPKAPK3	Hs.227789 gnl|UG|Hs#S377217 Human mitogen activated protein kinase activated protein kinase-3 mRNA, complete cds
9881	40569_at	MZF1	Hs.169832 gnl|UG|Hs#S2972 Human zinc finger protein 42 (MZF-1) mRNA, complete cds
4354	41475_at	NNJ1	Hs.11342 gnl|UG|Hs#S1054935 Human adhesion molecule ninjurin mRNA, complete cds
10410	32532_at	TJP1	Hs.74614 gnl|UG|Hs#S168 Human tight junction (zonula occludens) protein ZO-1 mRNA, complete cds
1736	32871_at	MRNA	Hs.268491 gnl|UG|Hs#S1368099 Homo sapiens mRNA; cDNA DKFZp564F133 (from clone DKFZp564F133)
2591	36238_at	FOXO4	Hs.239663 gnl|UG|Hs#S998083 Homo sapiens AFX1 gene, exon 1 (and joined CDS)
6270	37980_at	CIR1	Hs.89421 gnl|UG|Hs#S432 Human recepin mRNA, complete cds
2260	34031_i_at	KRIT1	Hs.93810 gnl|UG|Hs#S705836 Human Krit1 mRNA, complete cds
6277	37987_at	NCBP1	Hs.89563 gnl|UG|Hs#S2376 Human mRNA for nuclear cap binding protein, complete cds
7401	41746_at	NHP2L1	Hs.182255 gnl|UG|Hs#S1570375 Human DNA sequence from clone CTA-216E10 on chromosome 22 Contains the NHP2L1 gene for non-histone chromosome protein 2 (S. cerevisiae)-like 1, the G22P1 gene for thyroid autoantigen 70kD (Ku antigen), a HMG17 (high-mobility group (nonh
4195	41076_at	GJB3	Hs.98485 gnl|UG|Hs#S1368274 Homo sapiens connexin 31 (GJB3) gene, complete cds
9774	40224_s_at	PPP6R2	Hs.153121 gnl|UG|Hs#S1090805 Homo sapiens mRNA for KIAA0685 protein, complete cds
11281	1419_g_at	NOS2	D29675 /FEATURE=exon /DEFINITION=HUMNOSB Human inducible nitric oxide synthase gene, promoter and exon 1
8211	35306_at	DHX15	Hs.5683 gnl|UG|Hs#S952742 Homo sapiens mRNA for ATP-dependent RNA helicase #46, complete cds
10715	2012_s_at	PRKDC	Hs.155637 gnl|UG|Hs#S226297 Human DNA-dependent protein kinase catalytic subunit (DNA-PKcs) mRNA, complete cds
8517	36211_at	BCL2L2	Hs.75244 gnl|UG|Hs#S705511 Human mRNA for KIAA0271 gene, complete cds
10818	1893_s_at	ESR1	Estrogen Receptor
4340	41461_at	PMS1	Hs.111749 gnl|UG|Hs#S342192 Human homolog of yeast mutL (hPMS1) gene, complete cds
1315	35082_at	ZIC3	Hs.111227 gnl|UG|Hs#S998344 Homo sapiens zinc-finger protein of the cerebellum 3 (ZIC3) mRNA, complete cds
10796	1913_at	CCNG2	Hs.79069 gnl|UG|Hs#S376180 Human cyclin G2 mRNA, complete cds
316	31573_at	RPS25	Hs.113029 gnl|UG|Hs#S291 Human ribosomal protein S25 mRNA, complete cds
11367	1326_at	MAP3K8	Hs.5353 gnl|UG|Hs#S472670 Human apoptotic cysteine protease Mch4 (Mch4) mRNA, complete cds
5904	36896_s_at	ARNTL	Hs.74515 gnl|UG|Hs#S1055349 Homo sapiens basic-helix-loop-helix-PAS orphan MOP3 (MOP3) mRNA, complete cds
11830	853_at	NFE2L2	Hs.155396 gnl|UG|Hs#S554194 Nrf2=NF-E2-like basic leucine zipper transcriptional activator [human, hemin-induced K562 cells, mRNA, 2304 nt]
7918	33935_at	CACYBP	Hs.27258 gnl|UG|Hs#S1367450 H.sapiens gene from PAC 102G20
6266	37976_at	VSIG4	Hs.8904 gnl|UG|Hs#S1570447 Human DNA sequence from clone 159A1 on chromosome Xq12-13.3. Contains a novel gene and a Heterogenous Nuclear Ribonucleoprotein G (HNRNP G, Glycoprotein P43) pseudogene. Contains ESTs, an STS, GSSs, genomic marker DXS1213, and a ca repe
11477	1210_s_at	TNFRSF25	Hs.180338 gnl|UG|Hs#S1703841 Human death domain receptor 3 (DDR3) mRNA, alternatively spliced form 2, partial cds
4348	41469_at	PI3	Hs.112341 gnl|UG|Hs#S341954 Huma elafin gene, complete cds
6011	37243_at	GUCY1B3	Hs.77890 gnl|UG|Hs#S4748 H.sapiens soluble guanylate cyclase small subunit mRNA

a mutant KIT receptor, causing loss of polycomb complex association from PAX5 chromatin [33]. Some studies demonstrated the importance of MZF1 in high-throughput mammalian transcription factor interaction. A typical example is that a potential MIXL1-Tbox-MZF1 contains multiprotein complex is of great possibility to mediate transcriptional regulation of c-REL in AML [34]. Besides, EPHB6, a Metastasis Suppressor, and Mutations of the EPHB6 Recep-



(a) F-score results comparisons on Table 3

(b) Accuracy results comparisons on Table 3

(c) F-score results comparisons on Table 5

(d) Accuracy results comparisons on Table 5

FIGURE 6. The comparison results among algorithms based on the Nemenyi test.

tor Tyrosine Kinase may be a key factor in the induction of a Pro-Metastatic Phenotype Cancer [35]. Furthermore, a typical leukemia-related gene, named TJP1, was suggested to be an essential role establishing the podocyte filtration barrier. The suppression of TJP1 would lead to glomerular disorders [36]. Some published works discovered the hypermethylated status of TJP1 promoter region in newly diagnosed acute leukemia, which is correlated with the pathogenesis and progression of the disease [37]. Hence, TJP1 was suggested as a clinical molecular marker of leukemia [38] . Moreover, some other genes included in Table 5 are also discovered to play pivotals role in clonal expansion of human T-cell leukemia virus type 1 (HTLV-1)-infected cells, such as FoxO, NCBP1 [40-41].

E. PERFORMANCE COMPARISONS USING HYPOTHESIS TEST

The Friedman test and Nemenyi test [33] are applied to get a deeper insight for performance comparisons, with the aim of determining whether the performances of our algorithm's is different from other algorithms'.

Friedman test is a non-parametric statistical test [34], based on the hypothesis that there is no significant difference in the overall distribution of multiple pairs of algorithms' mean ranks. The mean ranks for all results are calculated by equation (41), where r_j^i is the rank of the j-th algorithm on the i-th data set. k is the number of algorithms to be compared, and M is the number of data sets.

$$r_j = \frac{\sum_{i=1}^{M} r_j^i}{M}, \forall j \in [1, k] \tag{41}$$

Then the Friedman statistic is computed by (42):

$$\tau_{\chi^2} = \frac{12M}{k(k+1)} \left(\sum_{i=1}^{k} r_i^2 - \frac{k\,(k+1)^2}{4} \right) \tag{42}$$

As τ_{χ^2} was found to be too conservative, the improved version [34] is given as formula (43):

$$\tau_F = \frac{(M-1)\,\tau_{\chi^2}}{M\,(k-1) - \tau_{\chi^2}} \tag{43}$$

By setting the significance level $\alpha = 0.10$, the critical value can be obtained from [42]. For the results comparison on Table 3, $M = 6$ and $k = 4$, the critical value is 6.4. Based on equation (43), $\tau_F = 6.9$ for F-score indices comparisons, larger than 6.4, so the results are of significant difference on the F-score indices. While for accuracy, $\tau_F = 4.95$, there is no significant difference. As F-score reveals the balance of classification results among classes, the advantage of our algorithm is to predict more balanced results.

For Table 5, $M = 6$ and $k = 8$ because OVO and OVR based results are compared at the same time. The critical value is 11.67 in this case. As $\tau_F = 15.51$ for F-score comparisons, $\tau_F = 14.04$ for accuracy comparisons, there are significant differences among the results of different algorithms when comparing both indices among different algorithms. So for the multiclass problem in our experiment, we can safely rejected the original hypothesis.

Nemenyi test defines the critical difference (CD) value for that two methods are significantly different with certain confidence (1-γ), as calculated by equation (44)

$$CD = q_\gamma \sqrt{\frac{k(k+1)}{6M}} \tag{44}$$

γ is set as 0.1, meaning that the confidence interval is 90%. So $q_\gamma = 3.11$, $CD = 2.32$ for Table 3, and $q_\gamma = 3.53$, $CD = 4.99$ for Table 5. The results of the post-hoc test for Table 3 and 5 are shown in Figure.6, where the mean rank of each algorithm is marked by a dot, and the horizontal bar across each dot shows the range of the Nemenyi value. Then two methods are significantly different when there is no overlap between their horizontal bars.

From Fig.6, it is observed that SCVR and OVR based SCVR can get the best mean ranks in two tables. It is interesting to find that OVO based SCVR perform slightly worse than OVO based RF and HERF, and ranked as the forth place. The reason may lie in that in the case of OVO, the small sample size problem makes SCVR to suffer the undertraining problem. While this problem is alleviated with RF and HERF because they can learn from diverse feature subspace to build ensemble models. That is, the advantage of both RF and HERF are based on the deployment of more classifiers.

As OVR based SCVR can achieve the best performance with the most compact ensemble size, it can be concluded that SCVR can handle the class imbalance problem well by drawing discriminate rules. Furthermore, the overlapping among results of diverse algorithms and SCVR is not large, so the mean rank results confirm that SCVR can outperform other algorithms, especially the OVR based RF and DT.

IV. CONCLUSIONS

In conclusion, this paper proposes a computational verb rule (CVR) based algorithm to learn and analyses microarray datasets by providing the verb *Stay* based rules (SCVR). Our rules extract the status of data using the combination of linguistical verbs and adverbs. In this way, the rules can produce interpretable results to biomedical scientists, offer them better understanding of the relationships among the microarray data. Up to now, although there are already many methods designed based on gene pairwise [43], our CVR based method has never been discussed and applied in the microarray data research field.

The principle and learning methods for SCVR are described in this study. In order to demonstrate the effectiveness of SCVR, some experiments are conducted on several binary class and multiclass datasets. The classifiers such as decision tree, RF and rotation forest are also employed in the experiments for further comparisons. Although only five rules are fused to form the final SCVR classifiers for each two-class problem with ten genes engaged, the final results illustrate that SCVR can achieve the best performance in most cases. The selected genes are also verified, and the findings confirm the biological significance of the selected features. As a result, our SCVR achieves the goal of feature selection and classification simultaneously with excellent generalization ability. The *Stay* based rules is our first attempt in applying the CVR to the classification task. For our future direction, we tend to explore the computational verb theory, in order to generate more interesting models that will be benefiting the biomedical related research aspects. Some more verbs and adverbs can be connected together to form powerful rules, revealing the relationships among features from various aspects. For example, the verbs *increase* and *decrease* can be used to capture the movement of mouse and eyebrow with the adverb *slightly*, so they may be a powerful tool to extract features for micro-expression expression from video streaming.

ACKNOWLEDGMENT

(Vincent To Yee Ng and Qingqi Hong contributed equally to this work.)

REFERENCES

[1] Z. Gan, F. Zou, N. Zeng, B. Xiong, L. Liao, H. Li, X. Luo, and M. Du, "Wavelet denoising algorithm based on NDOA compressed sensing for fluorescence image of microarray," *IEEE Access*, vol. 7, pp. 13338–13346, Jan. 2019.

[2] 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, and C. D. Bloomfield, "Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, no. 5439, pp. 531–537, 1999.

[3] K. H. Liu, Z. H. Zeng, and V. T. Y. Ng, "A hierarchical ensemble of ECOC for cancer classification based on multi-class microarray data," *Inf. Sci.*, vol. 349, pp. 102–118, Jul. 2016.

[4] M. Sun, K. Liu, Q. Wu, Q. Hong, B. Wang, and H. Zhang, "A novel ECOC algorithm for multiclass microarray data classification based on data complexity analysis," *Pattern Recognit.*, vol. 90, pp. 346–362, Jun. 2019.

[5] S. Kim and J. Park, "Hybrid feature selection method based on neural networks and cross-validation for liver cancer with microarray," *IEEE Access*, vol. 6, pp. 78214–78224, 2018.

[6] W. Ke, C. Wu, Y. Wu, and N. N. Xiong, "A new filter feature selection based on criteria fusion for gene microarray data," *IEEE Access*, vol. 6, pp. 61065–61076, 2018.

[7] Z. Wang and V. Palade, "Building interpretable fuzzy models for high dimensional data analysis in cancer diagnosis," *BMC Genomics*, vol. 12, no. 2, p. S5, 2011.

[8] S.-Y. Ho, L.-S. Shu, and J.-H. Chen, "Intelligent evolutionary algorithms for large parameter optimization problems," *IEEE Trans. Evol. Comput.*, vol. 8, no. 6, pp. 522–541, Dec. 2004.

[9] L. Ohno-Machado, S. Vinterbo, and G. Weber, "Classification of gene expression data using fuzzy logic," *J. Intell. Fuzzy Syst.*, vol. 12, no. 1, pp. 19–24, 2002.

[10] S. A. Vinterbo, S. K. Kim, and L. Ohno-Machado, "Small, fuzzy and interpretable gene expression based classifiers," *Bioinformatics*, vol. 21, no. 9, pp. 1964–1970, 2005.

[11] K. Sarkar, P. Chatterjee, and N. R. Pal, "Finding synergy networks from gene expression data: A fuzzy-rule-based approach," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 6, pp. 1488–1499, Mar. 2016.

[12] A. Tan, D. Q. Naiman, L. Xu, R. L. Winslow, and D. Geman, "Simple decision rules for classifying human cancers from gene expression profiles," *Bioinformatics*, vol. 21, pp. 3896–3904, Aug. 2005.

[13] K.-H. Liu and C.-G. Xu, "A genetic programming-based approach to the classification of multiclass microarray datasets," *Bioinformatics*, vol. 25, no. 3, pp. 331–337, 2009.

[14] K.-H. Liu, M. Tong, S.-T. Xie, and V. T. Y. Ng, "Genetic programming based ensemble system for microarray data classification," *Comput. Math. Methods Med.*, vol. 2015, Jan. 2015, Art. no. 193406.

[15] T. Yang, *The Mathematical Principles of Natural Languages: The First Course in Physical Linguistics* (Monographs in Information Sciences), vol. 6. Tucson, AZ, USA: Yang's Scientific Press, 2007.

[16] T. Yang, *Lingua Naturalis Principia Mathematica*, 1st ed. Fujian, China: Xiamen University Press, (in Chinese), 2011.

[17] H. Wang, K. Li, and K. Liu, "A genetic programming based ECOC algorithm for microarray data classification," in *Proc. Int. Conf. Neural Inf. Process.*, 2017, pp. 683–691.

[18] C. Guanrong and T. T. Pham, *Introduction to Fuzzy Systems* (Mathematical and Computational Biology). Boca Raton, FL, USA: CRC Press, Nov. 2005.

[19] T. Tao, "Physical linguistics: A measuable linguistics based on computational verb theory," in *Fuzzy, Theory and Probability* (Monographs in Information Sciences), vol. 5. Tucson, AZ, USA: Yang's Scientific Press, 2004.

[20] M. Tong, "Extracting dynamic classification rules from microarray data," M.S. thesis, Xiamen Univ., Xiamen, China, 2012.

[21] E. F. Petricoin, III, A. M. Ardekani, B. A. Hitt, P. J. Levine, V. A. Fusaro, S. M. Steinberg, G. B. Mills, C. Simone, D. A. Fishman, E. C. Kohn, and L. A. Liotta, "Use of proteomic patterns in serum to identify ovarian cancer," *Lancet*, vol. 359, no. 9306, pp. 572–577, Feb. 2002.

[22] U. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine, "Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays," *Proc. Nat. Acad. Sci. USA*, vol. 96, no. 12, pp. 6745–6750, 1999.

[23] G. J. Gordon, R. V. Jensen, L.-L. Hsiao, S. R. Gullans, J. E. Blumenstock, S. Ramaswamy, W. G. Richards, D. J. Sugarbaker, and R. Bueno, "Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma," *Cancer Res.*, vol. 62, pp. 4963–4967, Sep. 2002.

[24] D. Singh, P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. D'Amico, J. P. Richie, and E. S. Lander, "Gene expression correlates of clinical prostate cancer behavior," *Cancer Cell*, vol. 1, no. 2, pp. 203–209, 2002.

[25] S. A. Armstrong, J. E. Staunton, L. B. Silverman, R. Pieters, M. L. den Boer, M. D. Minden, S. E. Sallan, E. S. Lander, T. R. Golub, and S. J. Korsmeyer, "MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia," *Nature Genet.*, vol. 30, no. 1, p. 41, 2001.

[26] D. G. Beer, S. L. Kardia, C. C. Huang, T. J. Giordano, A. M. Levin, D. E. Misek, L. Lin, G. Chen, T. G. Gharib, D. G. Thomas, and M. L. Lizyness, "Gene-expression profiles predict survival of patients with lung adenocarcinoma," *Nature Med.*, vol. 8, no. 8, pp. 816–824, Aug. 2002.

[27] J. Khan, J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer, "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks," *Nature Med.*, vol. 7, no. 6, pp. 673–679, 2001.

[28] C. M. Perou, T. Sørlie, M. B. Eisen, M. Van De Rijn, S. S. Jeffrey, C. A. Rees, J. R. Pollack, D. T. Ross, H. Johnsen, L. A. Akslen, and Ø. Fluge, "Molecular portraits of human breast tumours," *Nature*, vol. 406, no. 6797, pp. 747–752, 2000.

[29] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, and J. Vanderplas, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.

[30] B. Ayerdi and M. Graña, "Hybrid extreme rotation forest," *Neural Netw.*, vol. 52, pp. 33–42, Apr. 2014.

[31] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manage.* , vol. 45, no. 4, pp. 423–437, 2009.

[32] B. W. Matthews, "Comparison of the predicted and observed secondary structure of T4 phage lysozyme," *Biochim. Biophys. Acta-Protein Struct.*, vol. 405, no. 2, pp. 442–451, Oct. 1975.

[33] D. Ray, S. Y. Kwon, A. Ptasinska, and C. Bonifer, "Chronic growth factor receptor signaling and lineage inappropriate gene expression in AML: The polycomb connection," *Cell Cycle*, vol. 12, no. 14, pp. 2159–2160, 2013.

[34] A. Raymond, B. Liu, H. Liang, C. Wei, M. Guindani, Y. Lu, S. Liang, L. S. S. John, J. Molldrem, and L. Nagarajan, "A role for BMP-induced homeobox gene MIXL1 in acute myelogenous leukemia and identification of type I BMP receptor as a potential target for therapy," *Oncotarget*, vol. 5, no. 24, pp. 12675–12693, 2014.

[35] C. M. Bailey and P. M. Kulesa, "Dynamic interactions between cancer cells and the embryonic microenvironment regulate cell invasion and reveal EphB6 as a metastasis suppressor," *Mol. Cancer Res.*, vol. 12, no. 9, pp. 1303–1313, 2014.

[36] M. Itoh, K. Nakadate, Y. Horibata, T. Matsusaka, J. Xu, W. Hunziker, and H. Sugimoto, "The structural and functional organization of the podocyte filtration slits is regulated by Tjp1/ZO-1," *PLoS ONE*, vol. 9, no. 9, 2014, Art. no. e106621.

[37] C. Wang, G. J. Wang, Y. H. Tan, W. Li, C. H. Liu, and L. Yu, "The methylation pattern and clinical significance of Zonula occludens-1 gene promoter in acute leukemia," *Zhonghua Neike Zazhi*, vol. 47, no. 2, pp. 111–113, 2008.

[38] M. Forero-Castro, C. Robledo, R. Benito, M. Abáigar, A. Á. Martín, M. Arefi, J. L. Fuster, N. de las Heras, J. N. Rodríguez, J. Quintero, and S. Riesco, "Genome-wide DNA copy number analysis of acute lymphoblastic leukemia identifies new genetic markers associated with clinical outcome," *PLoS ONE*, vol. 11, no. 2, 2016, Art. no. e0148972.

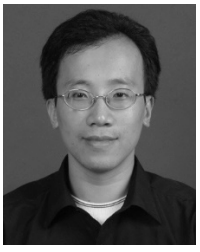
[39] A. Kode, I. Mosialou, S. J. Manavalan, C. V. Rathinam, R. A. Friedman, J. Teruya-Feldstein, G. Bhagat, E. Berman, and S. Kousteni, "FoxO1-dependent induction of acute myeloid leukemia by osteoblasts in mice," *Leukemia*, vol. 30, no. 1, pp. 1–13, 2016.

[40] H. Wen, Y. Li, S. N. Malek, Y. C. Kim, J. Xu, P. Chen, F. Xiao, X. Huang, X. Zhou, Z. Xuan, and S. Mankala, "New fusion transcripts identified in normal karyotype acute myeloid leukemia," *PLoS ONE*, vol. 7, no. 12, 2012, Art. no. e51203.

[41] R. L. Iman and J. M. Davenport, "Approximations of the critical region of the fbietkan statistic," *Commun. Statist.-Theory Methods*, vol. 9, no. 6, pp. 571–595, 1980.

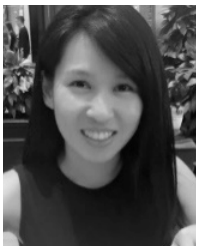
[42] C. López-Vázquez and E. Hochsztain, "Extended and updated tables for the Friedman rank test," *Commun. Statist.-Theory Methods*, vol. 48, no. 2, pp. 268–281, 2019.

[43] L. Zhu, W. Guo, S.-P. Deng, and D. S. Huang, "ChIP-PIT: Enhancing the analysis of ChIP-Seq data using convex-relaxed pair-wise interaction tensor decomposition," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 13, no. 1, pp. 55–63, Aug. 2016.

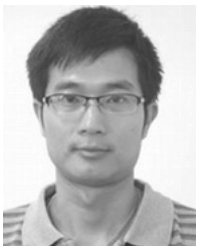


KUN-HONG LIU received the B.Sc. and M.Sc. degrees from Fujian Normal University, in 1999 and 2004, respectively, and the Ph.D. degree from the University of Science and Technology of China. He is currently a Professor with the School of Informatics, Xiamen University. His research interests include machine learning with a focus on ensemble learning, evolutionary algorithm, and pattern recognition.

VINCENT TO YEE NG received the Ph.D. degree from Simon Fraser University. He is currently an Associate Professor with the Department of Computing, The Hong Kong Polytechnic University. For the past few years, he was a Board Member of the Public Examination Board of the HKEAA and also involved in the curriculum development of the information technology subject for the senior secondary schools in EDB. His research interests include social media analysis, data mining, and health informatics.



SZE-TENG LIONG received the B.E. degree from Multimedia University, in 2014, and the Ph.D. degree from the University of Malaya, in 2017. She is currently leading the Computational Analytics & Cognitive Vision Lab, Feng Chia University, where she is also an Assistant Professor with the Department of Electronic Engineering. Her research interests include machine learning, pattern recognition, image processing, and computer vision.



QINGQI HONG received the Ph.D. degree in computer science from the University of Hull, U.K. He is currently an Associate Professor with the School of Informatics, Xiamen University, China. His current research interests include medical imaging processing, 3D visualization, 3D modeling, computer-aided diagnosis and surgery, deep learning, and GPU computing.