

Graph coloring for extracting discriminative genes in cancer data

Mohamed A. Mahfouz¹  | Juan A. Nepomuceno² 

¹Department of Computer and Systems Engineering, Faculty of Engineering, Alexandria University, Alexandria, Egypt

²Departamento de Lenguajes y Sistemas Informáticos, Higher Technical School of Computer Engineering, University of Seville, Seville, Spain

Correspondence

Mohamed A. Mahfouz, Department of Computer and Systems Engineering, Faculty of Engineering, Alexandria University, P.O. Box 21544, Egypt.

Email: m.a.mahfouz@gmail.com

Abstract

Background and objective: The major difficulty of the analysis of the input gene expression data in a microarray-based approach for an automated diagnosis of cancer is the large number of genes (high dimensionality) with many irrelevant genes (noise) compared to the very small number of samples. This research study tackles the dimensionality reduction challenge in this area.

Methods: This research study introduces a dimension-reduction technique termed graph coloring approach (GCA) for microarray data-based cancer classification based on analyzing the absolute correlation between gene–gene pairs and partitioning genes into several hubs using graph coloring. GCA starts by a gene-selection step in which top relevant genes are selected using a biserial correlation. Each time, a gene from an ordered list of top relevant genes is selected as the hub gene (representative) and redundant genes are added to its group; the process is repeated recursively for the remaining genes. A gene is considered redundant if its absolute correlation with the hub gene is greater than a controlling threshold. A suitable range for the threshold is estimated by computing a percentage graph for the absolute correlation between gene–gene pairs. Each value in the estimated range for the threshold can efficiently produce a new feature subset.

Results: GCA achieved significant improvement over several existing techniques in terms of higher accuracy and a smaller number of features. Also, genes selected by this method are relevant genes according to the information stored in scientific repositories.

Conclusions: The proposed dimension-reduction technique can help biologists accurately predict cancer in several areas of the body.

KEYWORDS

cancer diagnosis, cancer-specific genetic network, classification, dimension reduction, gene-expression analysis, graph coloring, microarrays

1 | INTRODUCTION

Huge amounts of microarray data acquired by DNA microarray technology raises the need for specific data-mining algorithms to extract useful patterns. These data are usually high dimensional, subject to noise, sometimes imbalanced, and usually having missing values. Microarray data-based cancer

classification is one of many computational methods that try to deal with huge gene expressions output from microarrays experiments to study different biological processes at the gene-expression level. Another early approach for automated diagnostic systems for cancer diagnosis is the texture analysis-based approach (Yuan, Curtis, Caldas, & Markowitz, 2012), colon cancer detection (Jiao, Chen, Li, & Xu, 2013), and other

areas of the body (Bauer et al., 2013). Statistical methods for reducing the dimensionality, which are based on orthogonal projection, such as principal component analysis (PCA), are widely used to reduce the high dimensionality of image data in this approach. Such methods suit an application in which the meaning of the reduced set or selected features is not important. In dealing with gene-expression data, the selected features need to be of clear biological meaning (Silva et al., 2005). However, statistical methods such as partial least squares, sliced inverse regression, and PCA are sometimes used to extract features from microarray data (Dai, Lieu, & Rocke, 2006; Khan et al., 2001; Silva et al., 2005).

The high dimensionality of gene-expression data raises the issue of how best to extract and select features from this data. The dimension reduction is an important step prior to classification to remove redundant and noisy gene expressions. The goal is to keep only the meaningful gene expressions to be input to the classification models to reduce the classification time, improve the accuracy of the model, and to allow monitoring of target disease. In microarray data-based cancer classification, the main objective of the dimension-reduction step is to formulate a reduced feature vector for every sample. A feature vector is required to be reduced as much as possible, while at the same time it should contain discriminative features (genes) called “marker” genes, which are necessary to classify given samples into their corresponding classes with high accuracy.

There are two main categories of commonly used techniques for reducing the high dimensionality of microarray data: filters (Kohavi & John, 1997) and wrappers (Langley, 1994). Filter methods prioritize genes according to one or more predefined measures and select the top-ranked genes. For example, a *t*-test (Welsh et al., 2001), signal-to-noise ratio (Golub et al., 1999; Li, Tang, & Li, 2005), and Wilcoxon rank-sum test (Yan, Deng, Fung, & Qian, 2005) are typical filter methods. Filter methods are easy to understand and implement. However, they ignore the interrelation of genes that may lead to losing important information. In addition, the classification accuracy of the feature genes selected by filter methods may be lower because the top-ranked *k*-genes are not guaranteed to be the best among all subsets of *k*-genes. Wrapper methods have also been widely used to select feature genes from microarray data (Alizadeh et al., 2000; Alon et al., 1999; Li et al., 2005; Li, Weinberg, Darden, & Pedersen, 2001; Xiong et al., 2001). They evaluate alternative feature gene subsets using classification accuracy and select the feature gene subset with the highest classification accuracy. The feature genes selected by wrapper methods usually have higher classification accuracy in comparison with the feature genes selected by filter methods.

There are several studies for microarray-based cancer classification that work with feature selection techniques to discover discriminating genes from the initial gene pool (Alon

et al., 1999; Grade et al., 2007; Kim et al., 2008). Genetic algorithm (GA) with the *k*-nearest neighbors (KNN) (Backert et al., 1999). Other research studies that use an ensemble of KNN classifiers as a gene selector are found in Okun & Priisalu (2009) and Bay (1999). Feature subsets are selected randomly in Bay (1999), while in Okun and Priisalu (2009), subsets with lower complexity are chosen using bolstered resubstitution error. In both techniques, improving classification accuracy is the main goal. Neural network techniques are also successfully applied to cancer diagnosis as in Kulkarni, Kumar, Ravi, & Murthy (2011), Lee, Man, Wang, & Cao (2013), Shon et al. (2009), Venkatesh, Thangaraj, & Chitra (2011). Wavelet transformation (Shon et al., 2009) is used for reduction of feature space along with the probabilistic neural network to classify colon cancer data. In Chen and Li (2007), multiple kernel of support vector machines (SVMs) are used to transform the problem of feature selection into a multiple parameter learning problem; then a tree-like algorithm is used to extract the classification rules from the obtained support vectors. Other approaches for cancer diagnosis are based on computing, distinguishing gene–gene pairs using correlation coefficients as in the TC-VGC algorithm (Shin, Yoon, Ahn, & Park, 2011), or by computing distinguishing biclusters as in the RBG-CD algorithm (Mahfouz, 2016). Both TC-VGC and RBG-CD are sensitive to their input parameters and require extensive parameter tuning, but they do not require the dimension-reduction step. The general conclusion is that no single classifier outperforms in all kinds of data sets and the dimension-reduction step plays an important role in cancer diagnosis.

An efficient and effective algorithm is necessary to extract and select features from microarray data to improve the performance of cancer data classification. This requirement motivates us to propose a feature selection method that groups the sets of genes into several information gene hubs in which the hub gene is the representative of each hub. The hub gene is highly relevant to the target label vector and, at the same time, genes in its group have an absolute correlation with it greater than a threshold. A suitable range for the controlling threshold is estimated by analyzing the gene–gene pair absolute correlation. The representatives of the produced groups are updated according to a computed score. A score is computed for each candidate representative as the total of its relevancy to the target label vector: compactness, separation, and overlapping of its group, if it is chosen as the hub gene. The proposed approach is a multivariate filter approach; however, it can also work as a wrapper technique by varying the controlling threshold. Many high-quality feature subsets can be efficiently computed and fed to a KNN classifier as in Okun and Priisalu (2009) and Bay (1999).

The remainder of this paper is organized as follows. Section 2 presents the proposed technique along with supporting material. In Section 3, the experimental environment is

presented. In Section 4, experimental results and their biological relevance are discussed. Finally, Section 5 concludes the paper and highlights future research directions.

2 | MATERIALS AND METHODS

The next sections explain the proposed technique in detail. Commonly used abbreviations and symbols are listed in Table 1 to help the reader to better understand the proposed algorithm.

2.1 | Similarity measures

The input data of the algorithm are basically a labeled gene-expression matrix. The gene-expression matrix is composed of labeled samples with their gene-expression profiles that are represented in rows and columns, respectively. Each sample in the input data set is labeled as malignant or normal in the case of binary class or otherwise is given a cancer grade. One gene expression means one feature value in the feature vector.

The most common similarity measures between rows or columns of the gene-expression matrix are similarity measures related to the correlation coefficient. Most of them have corresponding dissimilarity measures. The Pearson correlation coefficient of two random variables x and y is formally defined as follows:

$$s(x, y) = \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{s_x} \right) \left(\frac{y_i - \bar{y}}{s_y} \right) \quad (1)$$

where \bar{X}, \bar{Y} are the sample mean of x and y , respectively, while σ_x, σ_y are the sample standard deviations of x and y . It is a

measure for how well a straight line can be fitted to a scatter plot of x and y .

When one of the two vectors represents a categorical attribute, the square of the point biserial correlation coefficient (Glass & Hopkins, 1970) can be used to measure the correlation between them. In the proposed approach, Equation (2) is used to compute the relevancy between a feature vector (gene) g_i and the target label T , which is a binary attribute. The relevancy score between gene g_j and target label vector T is termed as $b_i(g_j, T)$, and it is defined as follows: If we divide the values in the column of gene g_j into two groups named GP and GN, where GP contains values in g_j that received the value “+1” on T , and GN contains values in g_j that received the value “-1” on T . The squared point biserial correlation coefficient is calculated as follows:

$$b_i(g_j, T) = \frac{(\mu_p - \mu_n)^2}{\sum_{i=1}^n (x_{ij} - \mu)^2} \left(\frac{c_p c_n}{c_p + c_n} \right) \quad (2)$$

where μ_p is the average of the values in group GP and μ_n is the average of the values in group GN. Further, c_p is the count of data points in group GP and c_n is the count of data points in group GN. μ is the average of the values in both group GP and GN. $b_i(g_j, T)$ measures the relevancy of feature values of g_j to the target label T and takes values between 0 (minimum relevancy score) and 1 (maximum relevancy score).

2.2 | Description of the proposed technique

As shown in Table 2, GCA starts by computing the relevancy with target labels of all genes using the biserial correlation with the target label vector T , as defined in Equation (2). Then, the set of v top relevant genes are selected

TABLE 1 Abbreviation and symbols used in the text

Symbol	Description
GCA	The proposed graph coloring approach for dimension reduction
ACA	A competitive algorithm termed as ACA and entitled: Attribute clustering for grouping, selection, and classification of gene-expression data
X	Data set comprising n samples \times m genes
T	Target label vector $T = \{t_1, t_2, \dots, t_n\}$ corresponding to n samples in X ;
X^t	Data set comprising n^t samples of class t in X
$I(g_i, g_j)$	Absolute Pearson correlation between feature vectors representing expressions of genes g_i and g_j
x_{ij}	j th feature value of the i th sample in the data set X
α	Threshold on absolute correlation used by GCA, the corresponding percentage is <i>prcnt</i>
α_0	Initial value for the threshold α , the corresponding percentage is <i>prcnt0</i>
$b_i(g_j, T)$	The square of the point biserial correlation coefficient (Glass & Hopkins, 1970) of gene g_j and target label vector T
t_i	A relevance of a gene g_i based on t -statistics
$KI(f_i, f_j)$	The Kuncheva index (Kuncheva, 2004); a stability index between f_i and f_j
s_{tot}, σ_{tot}	Mean and standard deviation of the computed Kuncheva index for all pairs of generated feature sets
Test 1, Test 2	Randomization tests

TABLE 2 The proposed dimension-reduction technique (GCA)

Input X: Data set of size n samples \times m genes
T: Target label vector $\{t_1, t_2, \dots, t_S\}$ corresponding to S samples in X
$prcnt_0$: Underestimated value of a threshold, which is used in computing a degree for each gene. Default 10%
Output R: list of selected genes in descending order according to computed score
Begin (GCA)
1. Compute Relevancy_Score (g_h) = for $h = 1, 2, \dots, m$ as shown in Equation (2)
2. Select the list of v top relevant genes V
3. Normalize the input data set
4. Compute percentage graph (Figure 1) for gene–gene absolute correlation ($v \times m$ pairs)
5. Select α_0 , which corresponds to $prcnt_0$, using the computed percentage graph above and use it to compute degree $D(g_i)$ for each gene $g_i \in V$ using Equation (6)
6. Sort the genes in descending order according to their values of $D(g_i)$ where $g_i \in V$.
7. Compute representatives and fill groups. Several values for $prcnt$ may be tried. Default between 0.05% and 10%:
Select a value α that corresponds to a value $prcnt$ ($prcnt_0 > prcnt > 0$) from the percentage graph of step 4
Let C_0 be the ordered list of all genes, which is computed in step 6
$i = 0$; $R = \{ \}$ // i the index of current set, R the set of representative genes initially ϕ
Let G be the set of all genes
while $C_0 \neq \{ \}$
Move the top element of C_0 to the set of representative R as r_i
$i = i + 1$
$C_i = \{ r_i \}$
Delete r_i from G
for each gene $g_j \in G$ such that $I(r_i, g_j) \geq \alpha$ begin
Move g_j from C_0 to C_i
Delete g_j from G
End
Endwhile
8. Move each remaining gene $g_j \in G$ to C_i where $i = \arg \max_i I(r_i, g_j)$
9. For each gene $g_h \in C_i$, $i = 1, 2, \dots, R $ // compute score for each representative
Separation_Score (C_i) = $1 - \frac{1}{ R -1} \sum_{j \neq i} I(g_h, r_j)$ // average correlation with other representatives
Compactness_Score (C_i) = // Average correlations with other members
Overlap_Score (C_i) = $1 - \frac{ \{ r_j \in R - \{ r_i \} \text{ and } I(g_h, r_j) \geq \alpha \} }{ R -1}$
Total_Score (g_h) = $b_i(g_h, T) + \text{Separation_Score}(C_i) + \text{Compactness_Score}(C_i) + \text{Overlap_Score}(C_i)$
If (Total_Score (g_h) > Total_Score (r_i)) Then begin select g_h as the new representative for C_i end
end
10. Output R in descending order according to Total_Score (r_i)
End (GCA)

and termed as V in step 2. A cutting point for the top relevant genes can be selected by sorting the genes according to their relevancy to the target label and identifying the gene, which has much lower relevancy compared to its predecessor, then the predecessor of the gene is selected as the top relevant gene. Selecting the top relevant gene is an optional step; the sorted list of all genes based on their relevancy can be used.

However, experimental results showed that selecting top relevant genes is much more efficient and produce better results in terms of a smaller number of features and higher accuracy. For a two-class application, gene selection can be based on the simple t -statistic (Nguyen & Rocke, 2002). A relevance of a gene g_i based on t -statistics termed t_i is computed as follows:

$$t_i = \frac{\overline{x_i^1} - \overline{x_i^2}}{\sqrt{\frac{(s_i^1)^2}{n_1} + \frac{(s_i^2)^2}{n_2}}} \quad (3)$$

where the n_k , $\overline{x_i^k}$, s_i^k column is the size, mean, and sample standard deviation of the vector of values on the i^{th} column (corresponds to the gene number i) of the input data matrix X that belong to the class k for $k = 1, 2$. Using this formula, t -scores can be computed for all genes. A score for a gene may be computed by employing a linear transformation. Using Equation (3), a score of gene g_i is computed in Nguyen & Rocke (2002) as follows:

$$\text{score}(g_i) = (t_i - \min_i t_i) / (\max_i t_i - \min_i t_i) \quad (4)$$

The list of ordered top v genes with the best scores can be used in the next steps instead of the biserial correlation above.

In step 3, the input data matrix is normalized to reduce the computations required for computing gene–gene correlations in the next steps. The normalization is done by computing μ_j and σ_j of each column (gene) j , then each entry x_{ij} is replaced by $(x_{ij} - \mu_j) / \sigma_j$ for $i = 1, 2, 3, \dots, n$ and $j = 1, 2, 3, \dots, m$.

The computation of the absolute Pearson correlation is reduced to Equation (5) for the normalized matrix.

$$I(x, y) = \left| \frac{1}{n} \sum_{i=1}^n x_i y_i \right| \quad (5)$$

The absolute Pearson correlation coefficient is used as a measure for redundancy between gene–gene pairs in the proposed technique, assuming both high positive and high negative correlation mean redundancy, and thus take the absolute value of correlations.

In step 4, a percentage graph is computed in which each bin represents the percentage of gene–gene pairs having an absolute correlation greater than a value between 0 and 1. Any value close to the upper end of this graph may be assigned to the threshold in step 4. Each value for the input parameter $prcnt0$ corresponds to a value for the threshold.

An experimental study shows that the start of the right tail of the graph where the percentage equal 10% is a good choice for $prcnt0$. The corresponding correlation value α_0 for $prcnt0$ is depicted from a computed percentage graph as will be explained in Section 2.4.

A degree for each gene is computed in step 5 as the count of genes having absolute correlation with that gene greater than the initially underestimated threshold α_0 (possible values between 0 and 1).

$$D(g_j) = |\{(g_i, g_j) : I(g_i, g_j) > \alpha_0\}| \quad \text{for } g_i \in V \quad \text{and} \quad g_j \in G \quad (6)$$

The degree of a gene here refers to the number of nonadjacent nodes of that gene in a graph of m nodes. Each node

represents a gene where there is an edge between two genes if they have an absolute correlation less than α_0 .

The list of genes is sorted descending according to their degrees in step 6. In step 7, C_0 initially contains the ordered list of top relevant genes. Again, selecting the top relevant genes is an optional step (i.e., all genes can participate; C_0 starts with the ordered list of all relevant genes). In each iteration i of step 7, the top of C_0 is moved to a new set C_i as a representative for C_i . Then, any following gene in the sorted list that has an absolute correlation greater than a threshold $\alpha \geq \alpha_0$ is moved to C_i . Further, all elements of C_i are removed from the total set of genes. The value of α is computed from the percentage graph using the selected value of $prcnt$. Possible values for $prcnt$ are chosen between 0.05% and 10%. The lower the value of $prcnt$, the higher the new value of α .

Grouping highly correlated genes in this step turns into a coloring problem on this graph. A simple procedure for doing this is used in step 7, but any coloring algorithm such as recursive-large-first (RLF) (Galiner & Hertz, 2006) can be used. The coloring step tries to minimize the number of groups such that the absolute correlation between the hub gene (representative) and other genes in its group are kept above a threshold. If we reach the end of the ordered list of top relevant genes in step 7 before assigning all genes to a group, then remaining genes are redistributed in step 8 such that a gene is moved to the group to which it has a maximum correlation with its representative. Step 9 selects one of the members of each group as a new representative if it achieves a total score greater than the current representative. The total score is computed as the sum of four scores: relevancy, separation, compactness, and overlapping score. An overlapping score gives a higher score for a gene, which was less probable to join another group in the reduction step. The proposed algorithm has an advantage in that only steps 8–9 need to be recomputed to try several new possible values of α . By varying the value of $prcnt$ between 0 and $prcnt0$, at step 7, a corresponding value for α will be between α_0 and 1.

2.3 | Illustrative example

Let the set of genes to be reduced be g_1, g_2, g_3, g_4 , and the target vector is T with absolute correlation between them, as shown in Table 3, columns 1–5. The total number of distinct gene pairs = $m(m-1)/2 = 6$ where $m =$ number of genes = 4. After step 5, $D(g_i)$ will be computed for each gene. Values in column 6 are computed with a value of $prcnt0 = 0.85$, which results in the value of α_0 being 0.4. Two entries out of six have absolute correlation ≥ 0.7 so if we choose $prcnt = 0.35$, the new value of α will be 0.7. After step 6, the ordered list of genes will be $C_0 = \{g_2, g_3, g_1, g_4\}$ (as shown in column 7). After the first iteration of step 7, $C_0 = \{g_3, g_1\}$ and $C_1 = \{g_2, g_4\}$ with g_2 representing C_1 . Similarly, after a second iteration

TABLE 3 Illustrative example for the steps of the proposed dimension reduction technique (GCA)

	g_1	g_2	g_3	g_4	$D(g_i)$	Order	C_i	Rep. r_i	Total score (g_i)	New rep.	Overlap genes	New sep.	Total score C_i
g_1	–	0.5	0.7	0.1	2	3	C_2	N	$0.7 + 0.9 + 0.7$	Y	N	0.9	$2.3 + 0$
g_2	0.5	–	0.4	0.8	3	1	C_1	Y	$0.4 + 0.6 + 0.8$	N	N	–	–
g_3	0.7	0.4	–	0.5	3	2	C_2	Y	$0.6 + 0.5 + 0.7$	N	N	–	–
g_4	0.1	0.8	0.2	–	1	4	C_1	N	$0.8 + 1 + 0.8$	Y	N	0.9	$2.5 + 0$
T	0.7	0.4	0.6	0.8	–	–	–	–	–	–	–	–	–

of step 7, $C_0 = \phi$ and $G = \phi$, $C_1 = \{g_2, g_4\}$ and $C_2 = \{g_3, g_1\}$ with g_3 as the representative for C_2 (as shown in columns 8 and 9). After step 6, total score is computed for each gene (as shown in column 10) as follows:

$$\text{Relevancy Score } (g_1) = I(g_1, T) = 0.7$$

Separation Score (g_1) = $1 - I(g_1, g_4) = 0.9$ since g_4 is the previously computed representative for C_1

$$\text{Compactness Score } (g_1) = I(g_1, g_3) = 0.7$$

$$\text{Total Score } (g_1) = I(g_1, g_3) = 2.3$$

Similarly scores of g_2, g_3, g_4 are computed with respect to currently identified representatives. The new representatives according to computed scores are shown in column 11. Since no genes g_i in C_1 such that $I(g_i, g_4) \geq 0.7$ and no genes g_i in C_2 such that $I(g_i, g_1) \geq 0.7$, the overlap score of both C_1 and C_2 is zero (as shown in column 12).

New separation score is recomputed for each representative (as shown in column 13) as follows:

$$\text{Separation Score } (g_1) = \text{Separation Score } (g_4) = 1 - I(g_1, g_4) = 1 - 0.1 = 0.9$$

$$\text{Total Score of } C_1 = \text{Total Score } (g_4) + \text{Overlap Score } (C_1) = (0.8 + 0.9 + 0.8) + 0 = 2.5$$

$$\text{Total Score of } C_2 = \text{Total Score } (g_1) + \text{Overlap Score } (C_2) = (0.7 + 0.9 + 0.7) + 0 = 2.3$$

The representatives (reduced set of genes) are output in the following order: g_4, g_1 , respectively.

2.4 | Estimating a proper range for values of the thresholds α_0 and α

To estimate a proper range for values of the threshold α_0 and α , a percentage graph is computed for each data set based on relative frequencies (the proportion of distinct gene pairs having absolute correlation greater than a certain value between 0 and 1). As shown in Figures 1 and 2, there is a small proportion of gene pairs in the extreme at the upper ends of the graph. A proper value for α_0 is the beginning of the positive (right) tail as shown in Table 4. The value of the threshold α , which is used in step 7 of the algorithm can be set between α_0 and 1. A corresponding absolute correlation threshold for several values of percentage (10, 5, 1, 0.1, 0.05)% for each data set are shown in Table 4. The number of bins (intervals) is set to twice the cube root of the number of observations (number of distinct gene pairs $(m^2 - m)/2$) as recommended by the Rice

rule. For example, in the Kent Ridge data set, the number of genes (m) = 2000, so the number of bins equals to 251 bins using the Rice rule.

Figure 1 shows a suitable range for the threshold α that can be depicted from a graph that represents the percentage of gene–gene pairs, which have an absolute correlation greater than or equal to α . For example, in the Kent Ridge data set, 10% of gene–gene pairs have an absolute correlation greater than or equal to 0.7, as shown in Table 4. In our experiments, we found that the first parameter *prcnt0*, which is used in computing the degree of top relevant genes in steps 5, can be accurately estimated from the percentage graph as the start of the right tail. Also, the optimal value for *prcnt* can be estimated for each data set using grid search with cross-validation as explained in Section 3.3.

Data set complexity (Okun & Priisalu, 2009) can be depicted from the graphs in Figures 1 and 2. The shorter the right tail of the graph representing the data set is, the more complex it is; furthermore, the harder to tune the threshold α . Prostate-I in Figure 2 was the most complex data set among the six data sets studied in Okun and Priisalu (2009).

2.5 | Computational complexity of GCA

Computing relevancy with the target label vector is $O(n.m)$ where n is the number of samples and m is the number of genes. Selecting the top relevant genes costs $O(n - v + v \log v)$. Normalization step costs $O(n.m)$. After normalization, computing the absolute correlation between two genes costs $O(n)$ so the total cost of computing the degrees (step 5) is $O(v.m.n)$, where v is the number of top relevant genes and m is the total number of genes. The cost of sorting genes based on their degrees is $O(v \log v)$. The interesting property of the proposed approach is that different values for new α can be retried without the need for doing steps 1 to 6. The remaining steps 7 to 10 is $O(m.n.r)$, where r is the number of representatives.

3 | EXPERIMENTAL ENVIRONMENT

The proposed scheme for evaluation and comparison with other dimension-reduction techniques can be outlined as follows:

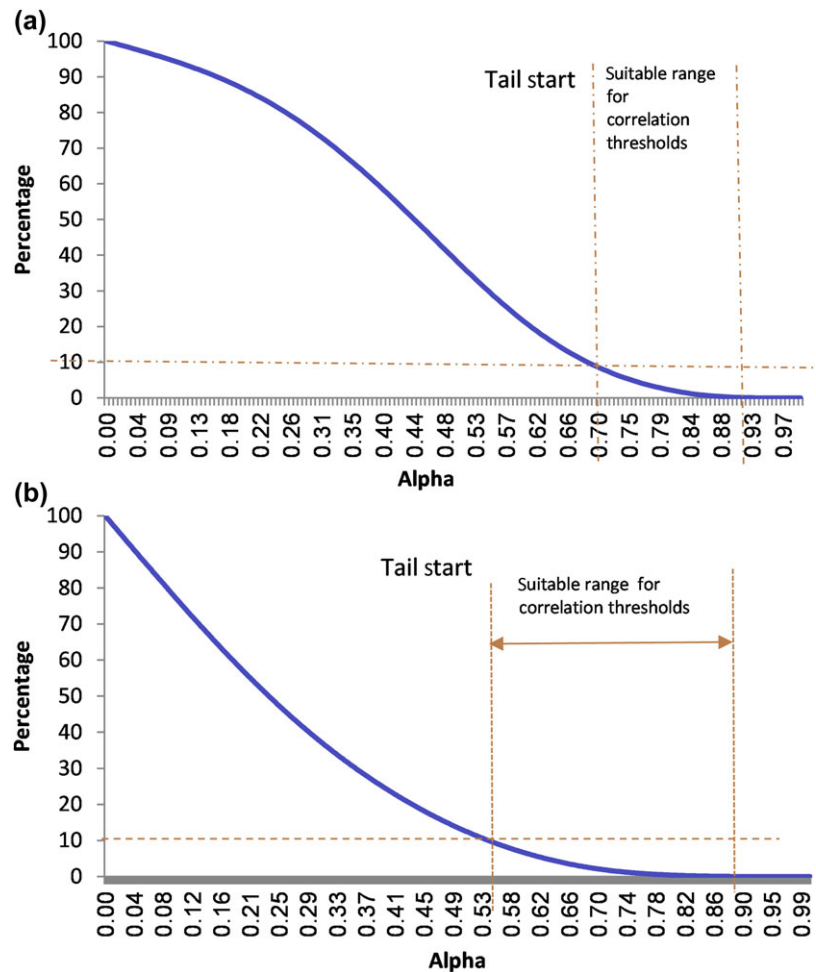


FIGURE 1 Percentage graph showing a suitable range of values for the threshold α (a) Kentridge. (b) GDS3257. [Colour figure can be viewed at wileyonlinelibrary.com]

- Evaluating the predictive performance of selected genes
- Evaluating the stability of selected genes
- Assessing the sensitivity of the proposed algorithm to its input parameters
- Testing the significance of selected genes using permutation tests and randomization tests
- Studying the biological relevance of selected genes

3.1 | Input data

The proposed algorithm has been applied to the six standard data sets in Table 4, to evaluate its performance. Table 5 shows additional properties for these data sets. It is clear from their properties that the analysis of this input data faces several challenges such as the large number of genes (high dimensionality) compared to a very small number of samples. Also, the data set is usually unbalanced. The original GDS3257 (lung) data set has been processed using a babelomic tool (Al-Shahrour, Minguez, Vaquerizas, Conde, & Dopazo, 2005) and, after filtering the steps, it is composed of 2,517 genes before applying the dimension-reduction step.

3.2 | Tools and libraries

The proposed algorithm is implemented using C# on windows 7, a 64-bit environment having a machine configuration of core I3, 2.4 GHz, 1 MB cache, and 4 GB of RAM. The MATLAB Toolbox for Dimensionality Reduction (version 0.8.1b) (van der Maaten, 2016) is used for comparing GCA with traditional feature-selection methods. Also, the ACCORD machine learning library (Souza, 2014) is used in experimenting with SVM. The babelomic tool (Al-Shahrour et al., 2005) is used in preprocessing the input data sets. The Cytoscape Agilent Literature Search (Lopes et al., 2010) is used in studying the biological relevance of the selected genes.

3.3 | Scheme for evaluating the predictive performance of selected genes

To solve a prediction problem, the resulting prediction model should be able to generalize to an independent data set (unknown samples) (i.e., accurately perform in practice). The prediction model is usually validated by partitioning the input data into two partitions (e.g., 70% for training and 30% for testing). However, in our problem, the number of samples is usually very small and there are not enough

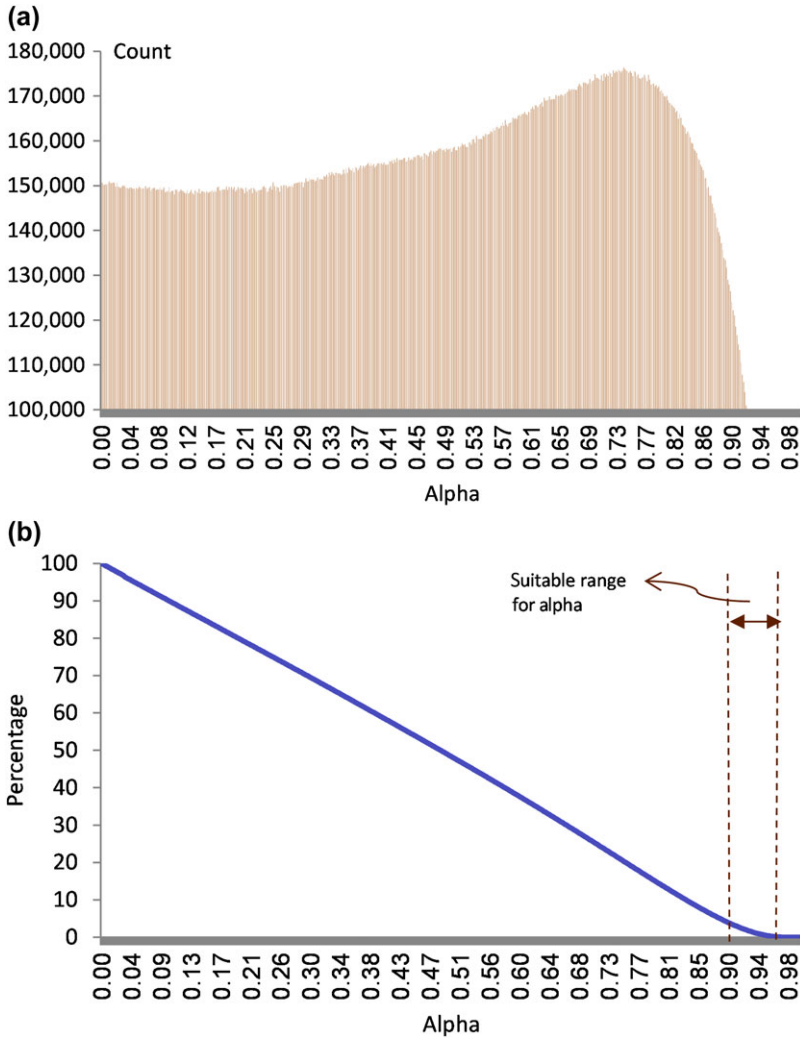


FIGURE 2 Data set complexity. (a) Histogram for Prostat-I. (b) Percentage for Prostate-I. narrow range for alpha means high complexity [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 4 Histogram data for the input gene-expression data sets

Data set	Ref.	No. genes	No. bins (rice rule)	Peak start	Corresponding α for different percentage of pairs having $I > \alpha$ that may represent suitable range for the threshold α				
					10% Tail start	5%	1%	0.1%	0.05% Tail end
Kent Ridge	Alon et al., 1999	2,000	251	0.692	0.701	0.765	0.852	0.914	0.928
GDS3257	Landi et al., 2008	2,516	370	0.008	0.542	0.624	0.751	0.862	0.883
Notterman	Notterman et al., 2001	7,547	763	0.0	0.375	0.443	0.577	0.725	0.763
Leukemia	Golub et al., 1999	7,129	740	0.0	0.374	0.456	0.601	0.732	0.762
CNS	Pomeroy et al., 2002	7,110	740	0.0	0.415	0.496	0.636	0.772	0.801
Prostate-I	Singh et al., 2002	12,600	859	0.0	0.890	0.883	0.936	0.964	0.968

TABLE 5 Properties of the input gene-expression data sets

Data set name	Type	High grade or malignant	Low grade or normal	Total samples	No. genes
Kent Ridge	Colon	40	22	62	2,000
GDS3257	Lung	49	58	107	2,517
Notterman	Colon	18	18	36	7,457
Leukemia	Blood	47	25	72	7,129
CNS	CNS	39	21	60	7,110
Prostate-I	Prostate	52	50	102	12,600

samples available to partition them into separate training and test sets without losing significant modeling or testing capability. In this work, a k -fold cross-validation scheme (Al-Shahrour et al., 2005) has been applied in testing the proposed models. The input data set is divided into k partitions (folds). $k-1$ partitions participate in training, and the classes of the instances belonging to the remaining partition are predicted by the decision model based on the training performed on $k-1$ training partitions. This process is repeated k times to form a complete cross-validation round after which the class of each sample is identified. Each time a training set X_{learning} and a testing set X_{testing} are selected using 10-fold cross-validation, the proposed dimension-reduction technique is applied on X_{learning} to produce the reduced training set X_{learning}^* having the same number of samples as X_{learning} but with a reduced set of genes. The classifier is trained using X_{learning}^* . The model output by the classifier is used to classify the unseen object in X_{testing} . For each complete round of cross-validation, the performance measures are computed.

The usefulness of the proposed algorithm has been evaluated by measuring the increase in the effectiveness of existing classifiers using well-known performance measures such as accuracy and AUC (area under the ROC curve) (Hassan, Chaudhry, Khan, & Kim, 2012) when GCA is used as the dimension-reduction step compared to other existing techniques. The calculation of these measures involves a number of true positive (TP), false positive (FP), true negative (TN), and false negative (FN). True negative and true positive are the number of correctly classified negative and positive samples, respectively. False negative and false positive are the number of positive and negative samples, which are incorrectly classified, respectively. Accuracy is a measure of overall effectiveness of the classification scheme. It can be calculated as

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (7)$$

Sensitivity is used to measure the ability of a classifier to recognize patterns of positive class. It can be obtained using the following equation.

$$Sensitivity = \frac{TP}{TP + FN} \quad (8)$$

Specificity is calculated to measure the ability of a classifier to recognize patterns of negative class. The following equation is used to calculate specificity:

$$Specificity = \frac{TN}{TN + FP} \quad (9)$$

In cancer diagnosis, sensitivity is more important than specificity as it shows how much a classifier can correctly identify all patients with a cancer that may be treatable at this time, but not later (e.g., cervical cancer). A good classifier

should have both a low false positive rate and low false negative rate. The definition of high accuracy means low $(FP + FN)/n$ so when there is a big difference between the operational FP and FN misclassification costs, or between the operational class frequencies compared to those in the training sample, then sensitivity and specificity together are a better indicator for the performance than accuracy. In our experiments, a well-balanced sensitivity and specificity are shown.

3.4 | Scheme for evaluating the stability of selected genes

Stability of a marker selection algorithm means that small changes in the training set should not result in big changes in the set of finally selected markers (i.e., adding or deleting a few samples from the training set should not significantly modify the feature set selected by the algorithm).

To measure the stability, we take a similarity-based approach presented in Abeel, Helleputte, Van de Peer, Dupont, & Saeys (2009). In this approach, selected feature stability is measured by computing the similarity between the feature sets selected from k randomly drawn samples from the input data set. The more similar all feature sets are, the higher the stability measure will be. The overall stability s_{tot} can then be defined as the average overall pairwise similarity comparisons between all features sets on the k subsampling.

$$s_{\text{tot}} = \frac{2 \sum_{i=1}^k \sum_{j=i+1}^k KI(f_i, f_j)}{k(k-1)} \quad (10)$$

where f_i represents the feature set obtained by the selection method on subsampling i ($1 \leq i \leq k$), and $KI(f_i, f_j)$ is the Kuncheva index (Kuncheva, 2004); a stability index between f_i and f_j , is defined as follows:

$$KI(f_i, f_j) = \frac{r - (s^2/m)}{s - (s^2/m)} \quad (11)$$

where $s = |f_i| = |f_j|$ and r is the number of common elements in both f_i and f_j . The s^2/m term in Equation (11) corrects a bias because of the chance of randomly selecting common features among two feature sets.

The Kuncheva index is greater than -1 and less than or equal to 1 . The greater the value of $KI(f_i, f_j)$, the larger the number of commonly selected features in f_i and f_j .

In our experiments for tuning the input parameters, both the predictive performance and the stability can be measured for different values of α in the estimated range, as discussed in Section 2.4. While the predictive performance is measured for the complete k -folds cross-validation round, the calculation of s_{tot} can be updated by the resulting feature set using the remaining $k-1$ folds for training. If we use 10-folds cross-validation and 50 runs for each value of α , then for each value of α we have 500 different random samples of the input data

sets, each of them 90% of the input data size. Initially, s_{tot} is set to 0, then it is updated by the generation of a new feature set f_i as follows:

$$s_{\text{tot}}(i) = ((i - 2) \times s_{\text{tot}}(i - 1) + \frac{1}{i - 1} \sum_{j=1}^{i-1} KI(f_i, f_j)) / (i - 1) \quad (12)$$

Let $\text{avg}KI(f_i)$ be the average Kuncheva index of a feature set f_i with all generated feature subsets.

$$\text{avg}KI(f_i) = \frac{1}{k - 1} \sum_{i,j} KI(f_i, f_j). \quad (13)$$

The standard deviation σ_{tot} can be computed as follows:

$$\sigma_{\text{tot}} = \sqrt{\frac{1}{k - 1} \sum_{i=1}^k (\text{avg}KI(f_i) - s_{\text{tot}})^2} \quad (14)$$

s_{tot} and σ_{tot} are the two measures used to study the stability of the proposed algorithm.

In the same experiments, if we evaluate the predictive performance of the generated feature subsets, we can study the sensitivity of the proposed algorithm regarding the number of features to include, which is controlled by the input parameter (i.e., the sensitivity of the proposed algorithm to its input parameter can be also shown).

3.5 | Randomization tests

To study the significance of the results of the proposed algorithm, the framework of permutation-based p -values, which are explored in Ojala and Garriga (2010) is followed. Two randomization tests are performed, namely, Test 1 and Test 2.

In Test 1, 1,000 randomized versions of the training data set are produced by permutations of the class labels of the original data set. In the cross-validation rounds, the permutation is done on the training set (i.e., on the data represented by the $k-1$ folds chosen for training) while the remaining fold (testing fold) is kept with its original labels. The classification for the testing fold is done by applying the selected classifier on the selected genes from the permuted-labels training folds. 100 cross-validation rounds are done to produce the 1,000 random samples.

In Test 2, the 1,000 randomized versions of the training data set are produced by applying independent permutations to the columns of the original data set within each class. The same procedure in Test 1 is followed for evaluating the predictive performance of the classifier on the selected genes from the generated randomized versions.

The p -value is computed as the fraction of randomized samples where the classifier performed better in the random data than in the original data (i.e., it estimates how likely the observed accuracy would be obtained by chance). The lower the p -values computed in Test 1 or Test 2, the higher the significance of the proposed technique.

A very small value of p -value (i.e., < 0.05) for a test is enough to reject its corresponding null hypothesis. The null hypothesis of Test 1 is that the features and the labels are independent, while for Test 2, the features are independent within class.

4 | EXPERIMENTAL RESULTS

4.1 | Tuning input parameters

To find an optimal set of gene expressions, we have experimented with different values for α , which resulted in several multiple sets (varying in size) of genes, and analyzed their effect on the classification accuracy achieved by a decision model (classifier). Table 6 reveals the corresponding results on the GDS3257 data set. Also, the same experiments are carried out with different values of top relevant genes v and lie in the range of 200–1,000 with increments of 100 genes, and we found that for all six data sets, increasing v more than 500 genes increases the computational complexity without any gain in the performance in terms of accuracy. In all of the next experiments, v is set to 500 and $\text{prcnt}0$, which corresponds to α_0 and is also set to 10% in all experiments. For GDS3257, $\text{prcnt}0 = 10\%$ corresponds to $\alpha_0 = 0.542$. While $\text{prcnt}0$ is fixed in all experiments, α_0 will vary depending on the input data set.

KNN achieves the highest classification accuracy for $\text{prcnt} = 5\%$, which corresponds to a value of 0.624 for α and results in the selection of 34 genes. Our sample sizes do not allow splitting the data into training, validation, and testing data sets. We used 10-fold cross-validation in optimizing the value of α . KNN is used as the classifier. The samples on the testing fold are classified using the feature subset obtained from the remaining 9 folds by GCA. The average accuracy of 100 cross-validation rounds are computed for each possible value of prcnt . Feature subsets obtained by varying the value of prcnt between 0 and $\text{prcnt}0$, in step 7, is shown in Table 6. They are sorted according to their accuracy first; second, for the individuals having the same accuracy, individuals with a small number of features are ranked in front.

4.2 | Comparing the estimated discriminative genes produced by GCA with other techniques

To compare the number of genes estimated by the proposed techniques with those estimated by other related techniques, the MATLAB Toolbox for Dimensionality Reduction (version 0.8.1b) (van der Maaten, 2016) is used. A number of discriminative genes are estimated using several techniques: eigenvalue-based estimation, maximum likelihood estimator (MLE), and estimator based on a correlation dimension. These techniques fail to estimate the proper number of genes. For example, the estimated number of genes for Kent

TABLE 6 Number of genes selected using different values of *prcnt* for gds3257 data set with $v = 500$, the row with bold text represents the best number of genes

Selected genes	<i>prcnt</i> %	α	Prediction accuracy of KNN	Stability	
				δ_{tot}	σ_{tot}
14	25	0.392	0.851	0.556	0.047
18	20	0.435	0.842	0.574	0.053
20	16	0.472	0.834	0.589	0.044
22	10	0.542	0.892	0.612	0.055
34	6	0.604	0.937	0.634	0.042
34	5	0.624	0.937	0.653	0.048
40	3	0.670	0.924	0.678	0.069
47	2	0.703	0.916	0.671	0.067
65	1	0.751	0.902	0.693	0.073
91	0.5	0.792	0.864	0.631	0.081
162	0.1	0.862	0.846	0.677	0.079

TABLE 7 Number of gene expressions selected by various feature selection strategies and estimation techniques for different data sets

Data set	Estimated no. of features using (van der Maaten, 2016) DR toolbox			No. features selected in Abeel et al. (2009) by different techniques				
	MLE	GMST	EigValue	PCA	mRMR	F-score	Chi-square	GCA
Kent Ridge	16	19	6	28	50	26	135	19
Notterman	25	20	15	33	120	95	185	14
Leukemia	32	93	8	97	180	135	220	68
CNS	30	45	12	96	175	165	180	84

Ridge using MLE was eight; however, the number of genes that achieved the best performance in the literature for Kent Ridge was much higher than eight. Furthermore, as shown in Table 7, the number of selected genes by GCA is lower than the number of genes that are selected by PCA, F-score, mRMR, and Chi-square for the four data sets as reported in Abeel et al. (2009).

4.3 | Performance of existing classifiers on selected genes

The results obtained by applying traditional KNN on both Kent Ridge and Leukemia data sets after reducing their dimensionality using several dimension-reduction techniques are given in Tables 8 and 9. The selected gene pools by GCA were fed to the KNN classifier. The results obtained by using the proposed dimension-reduction technique of GCA is compared to the results reported in Au, Chan, Wong, & Wang (2005) for other related techniques.

The experimental results in Table 8 for Kent Ridge data set show that GCA is superior to the other six gene-selection methods for reduced set sizes 14 to 35. As revealed by the classification results, average classification accuracy (ACA) was able to select a better small set (seven) of discriminative genes in the Kent Ridge data set than the others. However, GCA gives better results in terms of classification

accuracy for other reduced set sizes of 14 to 35. Also, the performance of GCA was comparable to other methods for different numbers of selected genes. In Tables 8 and 9, there were a number of genes that were not feasible to be generated by varying the value of *prcnt*, and their corresponding accuracies were calculated using linear interpolation. Other advanced gene-selection techniques have been proposed (Rajapakse & Mundra, 2013), but it is not used in comparison because it is not fully established.

The classification results on the Leukemia data set are given in Table 9. Also, ACA slightly outperforms GCA for a small number of genes up to 30, then GCA outperforms other techniques for other numbers of selected genes. Tables 10 and 11 show the performance of several existing classifiers on Leukemia and Kent Ridge, respectively. The predictive performance of the classifiers on the reduced set of the proposed technique is compared to the results reported in Au et al. (2005) for the best reduced set by ACA. Also, the results of the classifiers on the original data set without reduction is reported in Au et al. (2005). Both GCA and ACA show a much higher accuracy than the without-reduction case. The performance with a reduced set of GCA on Kent Ridge was higher than ACA for all classifiers while with Leukemia, the neural networks and the naïve Bayes classifiers were higher with the reduced data sets of ACA.

TABLE 8 The performance of KNN on the top genes selected by different techniques on Kent Ridge Data set

No. genes selected	Classification accuracy						
	ACA	<i>t</i> -value	<i>k</i> -means	SOM	Biclustering	mRMR	GCA
7	83.9	80.6	58.1	50.0	69.4	64.5	83.4
14	82.3	80.6	69.4	59.7	62.9	56.5	82.6
21	82.3	80.6	64.5	59.7	53.2	61.3	87.8
28	82.3	79.0	61.8	58.1	64.5	67.7	86.5
35	80.6	75.8	62.9	54.8	53.2	72.6	84.9

TABLE 9 The performance of KNN on the top genes selected by different techniques on Leukemia data set

No. genes selected	Classification accuracy						
	ACA	<i>t</i> -value	<i>k</i> -means	SOM	Biclustering	mRMR	GCA
10	91.2	82.4	50.0	50.0	52.9	61.8	83.7
20	91.2	88.2	44.1	61.8	52.9	70.6	88.1
30	91.2	88.2	44.1	67.6	58.8	67.6	88.5
40	91.2	88.2	47.1	70.6	58.8	70.6	91.9
50	91.2	82.4	47.1	67.6	52.9	70.6	92.3

TABLE 10 The performance of different classification algorithms on Leukemia data set

Classification algorithm	Accuracy using ACA	Accuracy using GCA	Accuracy without dimension reduction
Decision trees	94.1	95.3	91.2
Neural networks	97.1	96.2	91.2
Naïve Bayes	82.4	68.6	41.2
Nearest neighbors	91.2	92.3	82.4

TABLE 11 The performance of different classification algorithms on Kent Ridge data set

Classification algorithm	Accuracy using ACA	Accuracy using GCA	Accuracy without dimension reduction
Decision trees	91.9	93.1	82.3
Neural networks	90.3	92.3	83.9
Naïve Bayes	67.7	71.6	35.5
Nearest neighbors	83.9	87.8	79.0

TABLE 12 The performance of SVM (sigmoid) on the top genes selected by GCA compared to different reduction techniques

Data set	mRMR	F-score	Chi-square	PCA	GCA		
	acc./nof.	acc./nof.	acc./nof.	acc./nof.	acc./nof.	Sens.	Spec.
Kent Ridge	93.54/050	95.16/026	93.55/135	85.48/28	94.88/19	0.94	0.96
GDS3257	97.79/280	–	–	–	98.10/34	0.99	0.97
Notterman	91.67/120	94.44/095	88.89/185	86.11/33	95.73/14	0.96	0.94
Leukemia	–	97.22/135	–	–	96.86/68	0.95	0.97
CNS	–	95.00/165	–	–	96.34/84	0.95	0.98

Table 12 shows the performance of SVM with sigmoid as the kernel function. The best possible accuracy along with the number of top genes reported by Rathore, Hussain, & Khan (2014) are shown for F-score, mRMR, Chi-square, and PCA for five input data sets. The accuracy achieved using GCA as the dimension-reduction technique, respectively, are 0.3%, 1.3%, and 1.4%, slightly higher for GDS3257, Notterman, and central nervous system (CNS) data sets compared to individu-

als best achieved by using the other techniques. Also, the accuracy achieved using GCA are 0.2% and 0.3%, slightly lower for Kent Ridge and Leukemia data sets compared to individuals best achieved by using the other techniques. Results demonstrate that GCA was comparable to other techniques and at the same time it can select a much lower number of marker genes. Also, the results show that besides the high accuracy, there is a well-balanced sensitivity and specificity.

The better performance of GCA may be attributed to the characteristics of the underlying data set such as the case with GDS3257, which has a percentage graph with a long right tail that allows better tuning for α . In addition, the search for an optimal reduced set of genes is guided by the percentage graph in GCA.

Table 13 shows the average runtime of the proposed dimension-reduction technique. To further reduce the computational complexity, random sampling can be used to estimate the degree for each gene and the distribution of the correlation between pairs of genes by computing the histogram on a sample of pairs. However, in this research study, we compute the exact degree and compute an accurate histogram. Table 13 shows the details of the runtime of GCA. The total runtime is the sum of the normalization step, computing degrees (pair-wise correlation), sorting genes, and reduction time. Sorting time is negligible compared to others, while reduction time dominates other parts. The reduction is the only step that is required for trying other values for the threshold α . The runtime is less than 1.2 s for all the data sets. The higher the dimension of the data set, the higher the runtime is.

Table 14 shows the results of the two randomization tests described in Section 3.5 on GDS3257 and Prostate-I. The reported p -values are 0.001 for both tests and both data sets. This means that on none of the randomized samples did the KNN classifier perform better than on the original data in terms of accuracy.

4.4 | Comparing with embedded feature selection techniques

In this experimental study, the performance of GCA along with a selected classifier is compared to two embedded feature-selection techniques (Yang et al., 2010), in which the feature selection is included in their procedure for classifica-

tion. As shown in Table 15, the performance of GCA combined with an SVM (sigmoid) classifier is higher than the best reported results in Yang et al. (2010) for the two embedded feature selection techniques, namely, SVM-RFE and TSVM-RFE in terms of accuracy. Additionally, the proposed technique along with SVM has an advantage in that the gene selection is not a classifier-dependent selection as is the case with these methods.

Another two algorithms in which the process of reducing the dimensionality is combined with the classification task are RBG-CD (Mahfouz, 2016) and TC-VGC (Shin et al., 2011). TC-VGC and RBG-CD are based on computing distinguishing gene pairs and distinguishing biclusters, respectively. As shown in Table 16, the reported results in Mahfouz (2016) for TC-VGC and RBG-CD for GDS325 and prostate-I are compared to GCA combined with KNN. GCA shows a higher accuracy compared to TC-VGC and RBG-CD on Prostate-I. Also, the reported values for RBG-CD and TC-VGC are the average values for different values of their two input parameters. TC-VGC and RBG-CD have a slightly higher accuracy than KNN combined with GCA on Kent Ridge. Furthermore, both RBG-CD and TC-VGC are sensitive to their two input parameters, while CGA has only one parameter and it can be easily tuned as shown in Section 4.1.

4.5 | Biological discussion

The lung cancer data set with code GDS3257 (Landi) accessible at the NCBI GEO database (Edgar, Domrachev, & Lash, 2002) has been biologically studied. These data were generated in a study related with a kind of lung cancer called adenocarcinoma (Landi et al., 2008). Different samples were analyzed to study several tumor stages in a population of smokers and nonsmoking people. Adenocarcinoma is the most common type of cancer that starts in the lung. It is

TABLE 13 Details of runtime for the proposed dimension reduction technique

Data set	No. features	Norm. step 1 (ms)	Comp. degrees (ms)	Sort. genes (ms)	Reduct. time (ms)	Total time (ms)	Avg. score	Best no. features	Best score	Accuracy using KNN
Kent Ridge	14–37	57	126	0.27	84	267	0.44	21	0.49	87.8 ± 1.68
GDS3257	10–95	92	151	0.29	134	377	0.37	34	0.46	93.7 ± 1.39
Notterman	15–100	184	361	0.79	393	938	0.35	55	0.36	79.1 ± 2.07
Leukemia	13–105	164	180	0.72	268	612	0.33	50	0.41	92.3 ± 2.29
CNS	16–99	166	252	0.72	275	693	0.31	84	0.33	76.2 ± 2.53
Prostate-I	18–156	276	542	0.81	353	1171	0.29	33	0.31	93.15 ± 2.93

TABLE 14 Average accuracy and p -value (100 randomized samples) for Test 1 and Test 2 when using the KNN classifier

Algorithm	Original data	Test 1		Test 2	
	accuracy	accuracy	p -value	accuracy	p -value
GDS3257	93.7 ± 1.39	0.54 ± 0.09	0.001	0.48 ± 0.09	0.001
Prostate-I	93.1 ± 2.93	0.48 ± 0.09	0.001	0.39 ± 0.13	0.001

TABLE 15 Comparison with embedded feature selection algorithms

Algorithm	Accuracy	
	Kent Ridge	Leukemia
TSVM-RFE	91.25	96.32
SVM-RFE	91.25	96.03
SVM (sigmoid) combined with GCA	94.88	96.86

TABLE 16 Comparison with classifiers that are based on distinguishing pairs or biclusters

Algorithm	Accuracy	
	Prostate-I	GDS3257
TC-VGC (distinguishing pairs of genes)	91.05	95.8
RBG-CD (distinguishing biclusters)	92.30	93.9
KNN combined with GCA	93.15	93.7

usually found in lung outer areas such as the lining of the airways.

The proposed algorithm reported 34 marker genes as relevant genes, 18 of them differentially expressed in normal samples and 16 in tumor samples. The aim of the discussion is to study the biological relevance of these genes (Dupuy & Simon, 2007). They are reported in Table 17, where the first column presents their gene symbols and the second their corresponding label in accordance with the proposed algorithm. They are alphabetically ordered in the first column according to their names. A network has been built by a literature search using the Cytoscape Agilent Literature Search (Lopes et al., 2010). The tags “human” and “adenocarcinoma,” jointly the marker genes, have been used as input to build it. Every edge in the generated network is built by associations in public scientific repositories such as PubMed. This network has 250 nodes and 516 edges, where 23 of the nodes are the marker genes reported by the proposed algorithm. Several components form this network where the biggest component has 139 genes, the second one 24, and the third 11.

The complete network and especially the marker genes have been topologically analyzed. Table 17 shows the degree, betweenness, and closeness centrality measures, jointly the clustering coefficient, for each marker gene in the network. The degree shows the number of input edges for each node. The betweenness and closeness centrality measures indicate whether the node is a central node from the number of paths or from the distance to other nodes’ point of view, respectively. Moreover, the (local) clustering coefficient informs about whether the node works as an attractor respective of their neighboring nodes and whether they constitute a homogeneous group. Note that there is not any information in Table 17 for those marker genes not captured by the network. These genes are precisely interesting genes to consider as undiscovered biomarkers to focus on in future studies. However, the goal of this discussion is to study whether the reported marker genes are relevant genes or not.

These marker genes reported by the proposed algorithm are used as borders to capture the data set information. Therefore, they should play a pertinent role in related biological processes. The gene with the highest degree, the major number of edges, is *ABCBI* (see the first row in Table 17). It has a degree equal to 17 and its label is “tumor,” which means that it is differentially expressed in tumor samples. Figure 3a shows the second- and the third-biggest components; the *ABCBI* gene can be observed in the middle on the right. This gene is a central gene and most of the paths are related to it.

Note that its value for betweenness, closeness, and the clustering coefficient are 0.24, 0.28, and 0.21, respectively, which are high values. Moreover, a well-defined cluster of genes as a pentagon can be observed on the left of the figure. This group of genes is related with *ABCBI* because they are connected to the rest of the network through it. This group of six genes, the pentagon structure with a gene connected to *ATM*, has been studied using FuncAssociate (Berriz, King, Bryant, Sander, & Roth, 2003). Two GO terms with functionality related to the response to ionization and gamma radiation are overrepresented for these genes. In the same component, *PPP2R3C* is also a key gene with a degree equal to 3 with the label “tumor.” In Figure 3a, it can be observed that the third component where the marker gene *IFI35* has the highest value of the degree is equal to 5. Figure 3b shows a view of a part of the biggest component of the network. This component has 139 genes, *NPTX1* and *LIF* genes, with a degree of 17 and 15, respectively. They are the genes with the highest degree and they are precisely marker genes reported by the proposed algorithm. These are central genes that can be seen in Table 17; they have high values for centrality measures. *NPTX1* has a “tumor” label and gene *LIF* “normal.” It can be observed in Figure 3 that a group of six genes (the hexagon structure) are clearly connected with *NPTX1*.

These genes have been also studied with FuncAssociate and a GO term related with the regulation of cell proliferation has been found. It can be said that *NPTX1* plays a role in the cancer

TABLE 17 Selected marker genes with labels and the details of their relevance in the generated network

Gene symbol	Label	Degree	Betweenness centrality	Closeness centrality	Clustering coefficient
ABCB1	Tumor	17	0.24	0.28	0.21
ANG	Tumor	7	0.11	0.23	0.14
ANPEP	Tumor	8	0.03	0.19	0.28
AV764378	Normal	–	–	–	–
CXCL5	Normal	7	0.09	0.19	0.33
DEFB1	Normal	4	0.01	0.21	0.5
ECHDC3	Tumor	–	–	–	–
EFHD1	Tumor	2	1	1	0
GALNT12	Normal	1	0	1	0
GJA1	Normal	5	0.8	1	0.2
GPR171	Normal	–	–	–	–
HLA-DRB4	Tumor	–	–	–	–
IFI35	Normal	6	0.64	0.66	0.13
IGF2BP2	Normal	7	0.09	0.27	0.23
IGLV4-60	Tumor	–	–	–	–
IL37	Tumor	–	–	–	–
KMO	Normal	5	0.06	0.26	0.4
LECT1	Normal	2	1	1	0
LIF	Normal	15	0.24	0.3	0.28
LIPG	Tumor	–	–	–	–
LOX	Normal	5	0.02	0.26	0.4
LUC7L3	Tumor	–	–	–	–
NPTX1	Tumor	17	0.23	0.28	0.21
NR1D2	Tumor	–	–	–	–
PPP2R3C	Tumor	3	0	0.47	1
PTPRZ1	Normal	4	0.8	0.83	0.16
RERGL	Normal	1	0	1	0
RPL26L1	Tumor	–	–	–	–
RYR1	Normal	3	0.03	0.22	0
SH3YL1	Normal	4	0.03	0.23	0.16
SLC38A1	Tumor	9	0.08	0.26	0.78
SLC44A4	Normal	5	0.28	0.5	0.4
USP9Y	Tumor	–	–	–	–
XIST	Tumor	4	0.6	0.55	0.16
214110-s-at	Normal	–	–	–	–

process. Note that this gene is selected by the algorithm as a border gene to capture the data set information. Moreover, a similar situation has been analyzed for *LIF*, a marker gene not presented in the figure, and GO terms related with stem cell population maintenance and maintenance of cell number have been found (GO:0010628 and GO:0098727 terms). It must be also commented that marker genes *CXCL5* and *DEFB1*, with label “normal” and a degree of 7 and 4, can be observed in Figure 3b in the same component.

To test the significance of the biological relevance of selected genes, 100,000 groups of genes of average size

34 genes are selected randomly from the 2,517 genes of GDS3257 with replacements. Then, for the 34 genes selected by GCA and for each of the random groups, a score is computed.

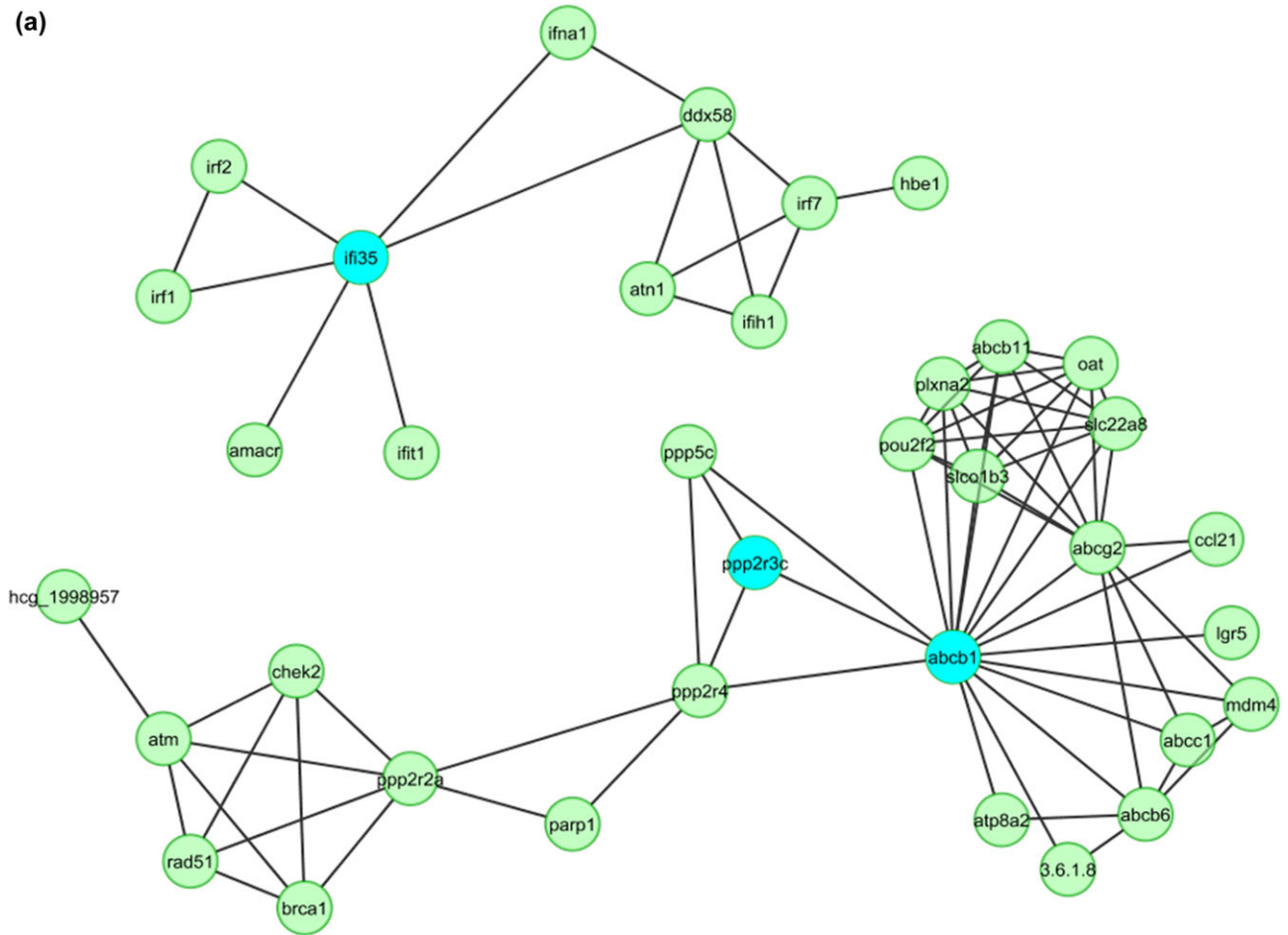
The score of a selected group (randomly or using GCA) is computed as follows:

Group Score (g_i) = (TP + TN)/ m , where m is the total number of genes in the data set, where

TP = how many of the selected 250 literature-mined genes do pop up in the selected group of genes

FP = the size of the group – TP

(a)



(b)

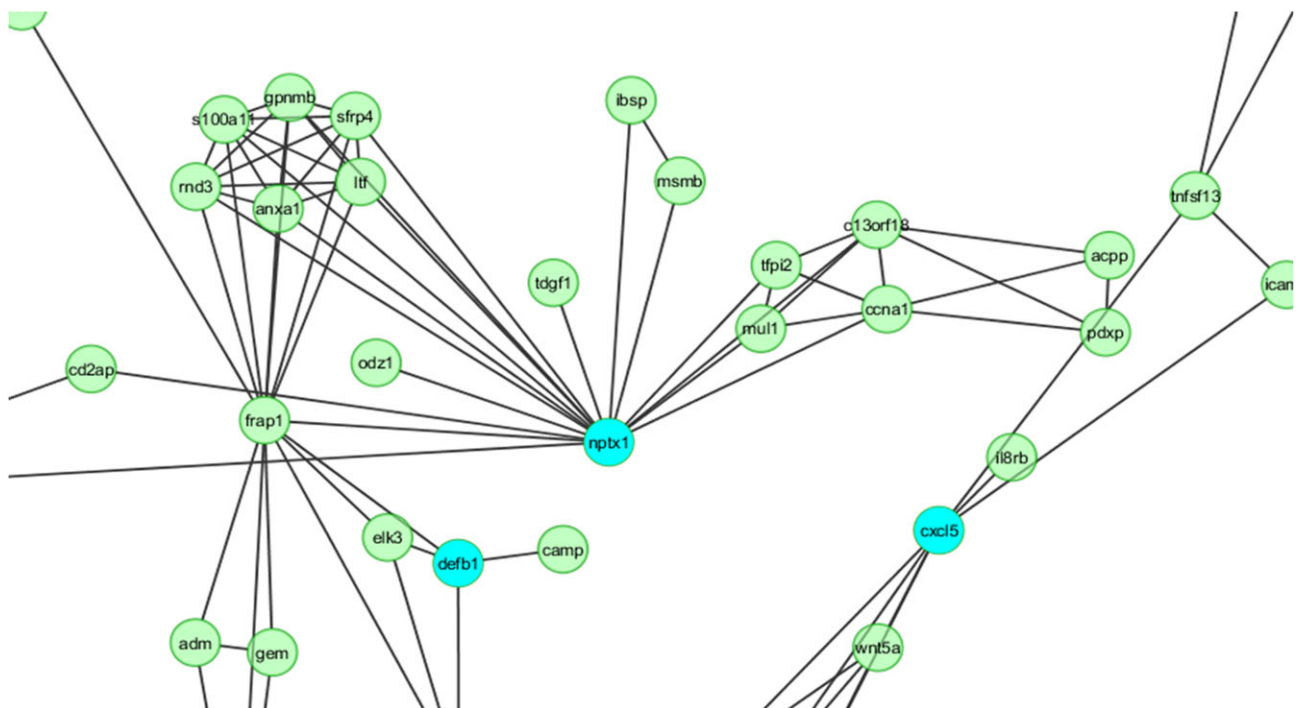


FIGURE 3 (a) Third component of the marker gene *IFI35* with the highest value, equal to 5, in this component. (b) View of a part of the biggest component of the network [Colour figure can be viewed at wileyonlinelibrary.com]

FN = how many genes in the whole set of genes (2,517 genes for GDS3257) minus the set of genes in the group that are in the selected 250 literature-mined genes

TN = the total number of genes – (FP + TP) – FN

For example, the score of selected genes by GCA = $(23 + (2517 - 34 - (250 - 23))) / 2517 = 0.905$.

The ratio between a number of randomly selected groups that achieve a higher score than the selected group by the proposed algorithm (GCA) and the total number of groups under study (i.e., 100,000) is computed and termed as a *p*-value. The lower the *p*-value, the higher the significance of selected genes by GCA. Experiments with GDS3257 showed a *p*-value equal to 0.00001.

5 | CONCLUSION AND FUTURE WORK

In summary, this research study proposes an algorithm to reduce the expected high dimension of cancer data sets and it is compared to several existing algorithms. The experiments have been conducted on six standard cancer data sets. Analysis reveals that genes selected by the proposed dimension-reduction technique are better able to accurately classify different data sets compared to the genes selected by other techniques. Additionally, the genes selected by the proposed dimension-reduction technique have been biologically studied. It has been shown that they are relevant genes according to the information stored in scientific repositories such as PubMed. Therefore, we can reasonably conclude that the proposed GCA can help biologists in accurately predicting cancer in several areas of the body.

The analysis of the proposed scheme in this paper suggests several directions for future work:

- Integrating gene-expression data and protein-interaction data for gene prioritization (Ma, Lee, Wang, & Sun, 2007) in GCA.
- Using an ensemble of KNN as a gene selector similar to Okun and Priisalu (2009). The feature subsets that are generated by varying the controlling thresholds are fed to the ensemble.

ACKNOWLEDGMENTS


We would like to thank Prof. Mohamed Ismail who provided insight and expertise that assisted the research and Prof. Amin Shoukry for comments that greatly improved the manuscript.

AUTHOR CONTRIBUTIONS


Mohamed Mahfouz was responsible for the study concept, design, and implementation, while Juan Nepomuceno was responsible for the preparation and interpretation of input data

and the biological analysis of the results. Both authors were responsible for drafting the manuscript. Mohamed Mahfouz was responsible for critical revision of the machine learning part of the manuscript. The authors declare that there are no conflicts of interest. No funding is declared.

ORCID

Mohamed A. Mahfouz 

<https://orcid.org/0000-0002-7242-3016>

Juan A. Nepomuceno 

<https://orcid.org/0000-0003-2851-951X>

REFERENCES

- Abeel, T., Helleputte, T., Van de Peer, Y., Dupont, P., & Saey, Y. (2009). Robust biomarker identification for cancer diagnosis with ensemble feature selection methods. *Bioinformatics*, 26(3), 392–398.
- Alizadeh, A. A., Eisen, M. B., Davis, R. E., Ma, C., Lossos, I. S., Rosenwald, A., ... Yu, X. (2000). Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature*, 403(6769), 503–511.
- Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., & Levine, A. J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences*, 96(12), 6745–6750.
- Al-Shahrour, F., Minguez, P., Vaquerizas, J. M., Conde, L., & Dopazo, J. (2005). Babelomics: A suite of web tools for functional annotation and analysis of groups of genes in high-throughput experiments. *Nucleic Acids Research*, 33(Suppl 2), W460–W464.
- Au, W.-H., Chan, K. C., Wong, A. K., & Wang, Y. (2005). Attribute clustering for grouping, selection, and classification of gene expression data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2(2), 83–101.
- Backert, S., Gelos, M., Kobalz, U., Hanski, M. L., Böhm, C., Mann, B., ... Moyer, M. P. (1999). Differential gene expression in colon carcinoma cells and tissues detected with a cDNA array. *International Journal of Cancer*, 82(6), 868–874.
- Bauer, S., Lu, H., May, C. P., Nolte, L. P., Büchler, P., & Reyes, M. (2013). Integrated segmentation of brain tumor images for radiotherapy and neurosurgery. *International Journal of Imaging Systems and Technology*, 23(1), 59–63.
- Bay, S. D. (1999). Nearest neighbor classification from multiple feature subsets. *Intelligent Data Analysis*, 3(3), 191–209.
- Berriz, G. F., King, O. D., Bryant, B., Sander, C., & Roth, F. P. (2003). Characterizing gene sets with FuncAssociate. *Bioinformatics*, 19(18), 2502–2504.
- Chen, Z., & Li, J. (2007). A multiple kernel support vector machine scheme for simultaneous feature selection and rule-based classification. *Pacific-Asia Conference on Knowledge Discovery and Data Mining, PAKDD 2007, Advances in Knowledge Discovery and Data Mining* (pp. 441–448). Berlin, Germany: Springer.
- Dai, J. J., Lieu, L., & Rocke, D. (2006). Dimension reduction for classification with gene expression microarray data.

- Statistical Applications in Genetics and Molecular Biology*, 5(1). <https://doi.org/10.2202/1544-6115.1147>
- Dupuy, A., & Simon, R. M. (2007). Critical review of published microarray studies for cancer outcome and guidelines on statistical analysis and reporting. *Journal of the National Cancer Institute*, 99(2), 147–157.
- Edgar, R., Domrachev, M., & Lash, A. E. (2002). Gene expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Research*, 30(1), 207–210.
- Galinier, P., & Hertz, A. (2006). A survey of local search methods for graph coloring. *Computers & Operations Research*, 33(9), 2547–2562.
- Glass, G. V., & Hopkins, K. D. (1970). *Statistical methods in education and psychology*. Englewood Cliffs, NJ: Prentice-Hall.
- Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., ... Caligiuri, M. A. (1999). Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286(5439), 531–537.
- Grade, M., Hörmann, P., Becker, S., Hummon, A. B., Wangsa, D., Varma, S., ... Difilippantonio, M. J. (2007). Gene expression profiling reveals a massive, aneuploidy-dependent transcriptional deregulation and distinct differences between lymph node—negative and lymph node—positive colon carcinomas. *Cancer Research*, 67(1), 41–56.
- Hassan, M., Chaudhry, A., Khan, A., & Kim, J. Y. (2012). Carotid artery image segmentation using modified spatial fuzzy c-means and ensemble clustering. *Computer Methods and Programs in Biomedicine*, 108(3), 1261–1276.
- Jiao, L., Chen, Q., Li, S., & Xu, Y. (2013). Colon cancer detection using whole slide histopathological images. In M. Long (Eds.), *World Congress on Medical Physics and Biomedical Engineering, May 26–31, 2012, IFMBE Proceedings, Beijing, China*, Vol 39 (pp. 1283–1286). Berlin, Heidelberg: Springer.
- Khan, J., Wei, J. S., Ringner, M., Saal, L. H., Ladanyi, M., Westermann, F., ... Peterson, C. (2001). Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine*, 7(6), 673–679.
- Kim, K., Park, U., Wang, J., Lee, J., Park, S., Kim, S., ... Park, J. (2008). Gene profiling of colonic serrated adenomas by using oligonucleotide microarray. *International Journal of Colorectal Disease*, 23(6), 569–580.
- Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97(1–2), 273–324.
- Kulkarni, A., Kumar, B. N., Ravi, V., & Murthy, U. S. (2011). Colon cancer prediction with genetics profiles using evolutionary techniques. *Expert Systems with Applications*, 38(3), 2752–2757.
- Kuncheva, L. I. (2004). *Combining pattern classifiers: Methods and algorithms*. Hoboken, NJ: Wiley.
- Landi, M. T., Dracheva, T., Rotunno, M., Figueroa, J. D., Liu, H., Dasgupta, A., ... Bergen, A. W. (2008). Gene expression signature of cigarette smoking and its role in lung adenocarcinoma development and survival. *PLoS One*, 3(2), e1651.
- Langley, P. (1994). *Selection of relevant features in machine learning* (Report No. FS-94-02). Menlo Park, CA: AAAI, pp. 245–271.
- Lee, K., Man, Z., Wang, D., & Cao, Z. (2013). Classification of bioinformatics dataset using finite impulse response extreme learning machine for cancer diagnosis. *Neural Computing and Applications*, 22(3–4), 457–468.
- Li, J., Tang, X. L., & Li, X. (2005). A novel visualization classifier and its applications. In L. Wang & Y. Jun (Eds.), *Lecture Notes in Computer Science, Vol. 3614. Fuzzy systems and knowledge discovery. FSKD* (pp. 1190–1199). Berlin, Germany: Springer.
- Li, L., Jiang, W., Li, X., Moser, K. L., Guo, Z., Du, L., ... Rao, S. (2005). A robust hybrid between genetic algorithm and support vector machine for extracting an optimal feature gene subset. *Genomics*, 85(1), 16–23.
- Li, L., Weinberg, C. R., Darden, T. A., & Pedersen, L. G. (2001). Gene selection for sample classification based on gene expression data: Study of sensitivity to choice of parameters of the GA/KNN method. *Bioinformatics*, 17(12), 1131–1142.
- Lopes, C. T., Franz, M., Kazi, F., Donaldson, S. L., Morris, Q., & Bader, G. D. (2010). Cytoscape Web: An interactive web-based network browser. *Bioinformatics*, 26(18), 2347–2348.
- Ma, X., Lee, H., Wang, L., & Sun, F. (2007). CGI: A new approach for prioritizing genes by combining gene expression and protein–protein interaction data. *Bioinformatics*, 23(2), 215–221.
- Mahfouz, M. A. (2016). RBG-CD: Residue based genetic cancer diagnosis. *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics, AISI 2016* (pp. 417–426). Berlin, Germany: Springer.
- Nguyen, D. V., & Rocke, D. M. (2002). Multi-class cancer classification via partial least squares with gene expression profiles. *Bioinformatics*, 18(9), 1216–1226.
- Notterman, D. A., Alon, U., Sierk, A. J., & Levine, A. J. (2001). Transcriptional gene expression profiles of colorectal adenoma, adenocarcinoma, and normal tissue examined by oligonucleotide arrays. *Cancer Research*, 61(7), 3124–3130.
- Ojala, M., & Garriga, G. C. (2010). Permutation tests for studying classifier performance. *Journal of Machine Learning Research*, 11, 1833–1863.
- Okun, O., & Priisalu, H. (2009). Dataset complexity in gene expression based cancer classification using ensembles of k-nearest neighbors. *Artificial Intelligence in Medicine*, 45(2), 151–162.
- Pomeroy, S. L., Tamayo, P., Gaasenbeek, M., Sturla, L. M., Angelo, M., McLaughlin, M. E., ... Lau, C. (2002). Prediction of central nervous system embryonal tumour outcome based on gene expression. *Nature*, 415(6870), 436–442.
- Rajapakse, J. C., & Mundra, P. A. (2013). Multiclass gene selection using Pareto-fronts. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 10(1), 87–97.
- Rathore, S., Hussain, M., & Khan, A. (2014). GECC: Gene expression based ensemble classification of colon samples. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 11(6), 1131–1145.
- Shin, E., Yoon, Y., Ahn, J., & Park, S. (2011). TC-VGC: A tumor classification system using variations in genes' correlation. *Computer Methods and Programs in Biomedicine*, 104(3), e87–e101.
- Shon, H.-S., Sohn, G., Jung, K. S., Kim, S. Y., Cha, E. J., & Ryu, K. H. (2009). Gene expression data classification using discrete wavelet transform. *International Conference on Bioinformatics and*

- Computational Biology, BIOCOMP 2009, July 13-16, Las Vegas, NV* (pp. 204–208).
- Silva, P. J., Hashimoto, R. F., Kim, S., Barrera, J., Brandão, L. O., Suh, E., & Dougherty, E. R. (2005). Feature selection algorithms to find strong genes. *Pattern Recognition Letters*, 26(10), 1444–1453.
- Singh, D., Febbo, P. G., Ross, K., Jackson, D. G., Manola, J., Ladd, C., ... Richie, J. P. (2002). Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell*, 1(2), 203–209.
- Souza, C. R. (2014). The accord.NET framework. Retrieved from <http://accord-framework.net>
- van der Maaten, L. (2016). Matlab toolbox for dimensionality reduction. Retrieved from <https://lvdmaaten.github.io/drtoolbox/>
- Venkatesh, E., Thangaraj, P., & Chitra, S. (2011). An improved neural approach for malignant and normal colon tissue classification from oligonucleotide arrays. *European Journal of Scientific Research*, 54, 159–164.
- Welsh, J. B., Zarrinkar, P. P., Sapinoso, L. M., Kern, S. G., Behling, C. A., Monk, B. J., ... Hampton, G. M. (2001). Analysis of gene expression profiles in normal and neoplastic ovarian tissue samples identifies candidate molecular markers of epithelial ovarian cancer. *Proceedings of the National Academy of Sciences*, 98(3), 1176–1181.
- Xiong, M., Fang, X., & Zhao, J. (2001). Biomarker identification by feature wrappers. *Genome Research*, 11(11), 1878–1887.
- Yan, X., Deng, M., Fung, W. K., & Qian, M. (2005). Detecting differentially expressed genes by relative entropy. *Journal of Theoretical Biology*, 234(3), 395–402.
- Yang, T., Kecman, V., Cao, L., & Zhang, C. (2010). Combining support vector machines and the t-statistic for gene selection in DNA microarray data analysis. *Proceedings of the 14th Pacific-Asia Conference, Hyderabad, India, June 21–24* (pp. 55–62).
- Yuan, Y., Curtis, C., Caldas, C., & Markowitz, F. (2012). A sparse regulatory network of copy-number driven gene expression reveals putative breast cancer oncogenes. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 9(4), 947–954.