Relative evolutionary hierarchical analysis for gene expression data classification

Marcin Czajkowski
Bialystok University of Technology, Poland
m.czajkowski@pb.edu.pl

Marek Kretowski
Bialystok University of Technology, Poland
m.kretowski@pb.edu.pl

ABSTRACT

Relative Expression Analysis (RXA) focuses on finding interactions among a small group of genes and studies the relative ordering of their expression rather than their raw values. Algorithms based on that idea play an important role in biomarker discovery and gene expression data classification. We propose a new evolutionary approach and a paradigm shift for RXA applications in data mining as we redefine the inter-gene relations using the concept of a cluster of co-expressed genes. The global hierarchical classification allows finding various sub-groups of genes, unifies the main variants of RXA algorithms and explores a much larger solution space compared to current solutions based on exhaustive search. Finally, the multi-objective fitness function, which includes accuracy, discriminative power of genes and clusters consistency, as well as specialized variants of genetic operators improve evolutionary convergence and reduce model underfitting. Importantly, patterns in predictive structures are kept comprehensible and may have direct applicability. Experiments carried out on 8 cancer-related gene expression datasets show that the proposed approach allows finding interesting patterns and significantly improves the accuracy of predictions.

CCS CONCEPTS

• **Computing methodologies** → *Classification and regression trees*; Supervised learning by classification; • **Applied computing** → *Bioinformatics*.

KEYWORDS

evolutionary data mining, relative expression analysis, hierarchical classification, gene expression data

ACM Reference Format:
Marcin Czajkowski and Marek Kretowski. 2019. Relative evolutionary hierarchical analysis for gene expression data classification. In *Genetic and Evolutionary Computation Conference (GECCO '19), July 13–17, 2019, Prague, Czech Republic.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3321707.3321862

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
GECCO '19, July 13–17, 2019, Prague, Czech Republic
© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-6111-8/19/07…$15.00
https://doi.org/10.1145/3321707.3321862

1 INTRODUCTION

Relative Expression Analysis (RXA) plays an important role in biomarker discovery and genomic data classification [12]. In a pioneer research [13], authors focus on ordering relationships between the expression of small sets of genes by defining and using ranks of genes instead of their raw expression values. The classification algorithms based on that idea appeared robust to small perturbations of gene expression values and insensitive to commonly used data normalization and standardization procedures. The RXA algorithms managed to identify many interesting gene-gene interactions and played important role in a biomarker discovery [19]. The influence of RXA solutions could be even greater, however, the computational complexity of the algorithms that use exhaustive search strongly limits the number of genes that can be analyzed [23]. This indicated the direction of most of the current solutions to perform rigorous feature selection and to limit the complexity of the analyzed relations to a minimum.

In order to change that trend, we propose an evolutionary approach called Relative Evolutionary Hierarchical Analysis (REHA) which unifies the main variants of RXA algorithms and redefines the inter-features relations. We have deliberately removed "expression" from the name as we believe our solution can be successfully applied in the future for other types of omics data like RXA algorithms are, for example in metabolomics [20] and proteomics [18]. However, in this study, we limit REHA description to the gene expression data.

Proposed solution redefine in a significant way the RXA concept with hierarchical gene clusters. A gene cluster is a part of a gene family, which is a set of homologous genes within one organism. It is composed of two or more genes found within an organism's DNA that encode for similar polypeptides, or proteins, which collectively share a generalized function. The use of information on subgroups of attributes is particularly important in the problem of classification and selection of genomic data [28].

The most significant novelty in the proposed solution is hierarchical clusters that are applied to find interactions and sub-interactions between co-expressed and epistasis genes. It combines Top-Scoring-Pair (TSP) [13] concept from the RXA and a multi-test solution from the decision trees [9]. With additional improvements like:

- built-in information about the discriminatory power of genes;
- specialized variants of genetic operators like two-level mutation;
- multi-objective fitness function, which includes classification accuracy, clusters consistency and external gene ranking;

we managed to not only improve the evolutionary convergence but also reduce possible model under and over-fitting to the data. A preliminary validation using 8 cancer-related gene expression datasets has shown that the REHA solution significantly improves

the accuracy of predictions in comparison to the rest of the major variants of RXA solutions. Importantly, the prediction model composed of hierarchical clusters is comprehensible and may have direct applicability.

In this paper, we propose a paradigm shift for the known RXA classifiers. Currently, all algorithms focus on a group of gene pairs that collectively improve only the classification accuracy. Such an approach does not promote finding relations between the gene pairs but only gene-gene relations that can be detected within the pair (in some cases triplet). We believe, that focusing on an evolutionary hierarchical split based on gene clusters rather than a group of mostly unrelated pairs of genes may improve not only classifiers performance and generalization ability but above all else allows finding interesting patterns that may appear in each cluster.

2 BACKGROUND

With the rapid development and availability of genomic research, a large number of gene expression data sets have become publicly available [30]. However, genomic data is still challenging for computational tools and mathematical modeling due to the high ratio of features/observations as well as enormous genes redundancy. Although in the literature we may find a good number of supervised machine learning algorithms, most of the methods provide 'black box' decision rules involving many genes combined in a highly complex fashion in order to achieve high predictive performance. However, it can be observed that there is a strong need for 'white box', comprehensive classification models which may actually help in understanding and identifying relationships between specific genes [1].

2.1 Algorithms for relative expression analysis

Among new computational tools designed to extract important and meaningful rules from gene expression data, RXA algorithms are gaining popularity. The RXA taxonomy that includes the main development paths is illustrated in Figure 1.

A Top-Scoring Pair (TSP) is the first and the most popular RXA solution proposed by Donald Geman [13]. It uses a pairwise comparison of gene expression values and searches for a single pair of genes with the highest rank. Let x_i and x_j be the expression values of two different genes from available set of genes and there are only two classes: *normal* and *disease*. At first, algorithm calculates the probability of the relation $x_i < x_j$ between those two genes in the objects from the same class:

$$P_{ij}(normal) = Prob(x_i < x_j | Y = normal) \quad (1)$$

and

$$P_{ij}(disease) = Prob(x_i < x_j | Y = disease), \quad (2)$$

where Y denotes the class of the objects. Next, the score for this pair of genes (x_i, x_j) is calculated:

$$\Delta_{ij} = |P_{ij}(normal) - P_{ij}(disease)|. \quad (3)$$

This procedure is repeated for all distinct pairs of genes and the pair with the highest score becomes titled top scoring pair. In case of a draw, a secondary ranking that bases on genes expression differences in each class and object is used [31]. Finally, for a new test sample, the relation between expression values of the top pair of genes is checked. If the relation holds, then the TSP predictor votes for the class that has the higher probability P_{ij} in the training set, otherwise it votes for the class with a smaller probability.

The k-TSP algorithm [31] is one of the first extensions of the TSP solution. It focuses on increasing the number of pairs in the prediction model and applies no more than k top scoring disjoint gene pairs with the highest score, where the parameter$k$ is determined by the internal cross-validation. This method was later combined with a top-down induced decision tree in an algorithm called TSPDT [8]. In this hybrid solution, each non-terminal node of the tree divides instances according to a splitting rule that is based on TSP or k-TSP accuracy.

Different approaches for the TSP extension focus on the relationships between more than two genes. Algorithms Top Scoring Triplet (TST) [19] and Top Scoring N (TSN) [23] analyze all possible ordering relationships between the genes, however, the general concept of TSP is retained.

One of the first heuristic method applied to RXA was the evolutionary algorithm called EvoTSP [7] that was later extended in the TIGER system [6]. The authors proposed a simple evolutionary search for the k-TSP and TSN-like rules. Performed experiments showed that even this simple evolutionary search is a good alternative to the traditional RXA algorithms.

Finally, there are many variations of the TSP-family solutions that propose new ways to rank the gene pairs of different systems that inherit the RXA methodology. Among them, we can distinguish trend-based approach [32], AUCTSP classifier that uses ROC curve [17] or VH-k-TSP [16] that focus on vertical and horizontal genes relations. The RXA analysis and the TSP solution are also applied as a feature selection for more complex classifiers [29, 33]. What's more, the strength and simplicity of RXA has been recognized outside genomics data and is being successfully used in proteomic [18] and metabolomic [21] analysis.

2.2 Limitations of the RXA algorihtms

The main drawback of RXA algorithms which affects the rest of issues concerning various restrictions and the depth of interactions is enormous computational complexity that equals $O(k * Z^N)$, where k is the number of top-scoring groups, Z is the number of analyzed genes and N is the size of a group of genes which ordering relationships is searched. In addition, calculation of all possible gene pairs or gene groups strongly limits the number of genes and inter-relations that can be analyzed. The largest so far ordering relationship tested a group of 4 genes (N=4) but only when the total number of analyzed genes was heavily reduced by the feature selection to a few hundred [23]. Although the parallelization of the algorithm managed to speed up calculation by two orders of magnitude [22], it is still computationally infeasible to calculate on a full gene expression dataset.

Another RXA limitation concerns the need of presenting the parameters k and N for the algorithms. It is almost impossible to define, for a particular problem in advance, what is the type of relationships in a dataset and how many genes or gene-pairs should be involved. For the k-TSP algorithm, the parameter k is determined by the internal cross-validation which increases the calculation time and decreases the size of an already small training set. It is also not clear which of the TSP solution should be applied: TSP, k-TSP, TSN

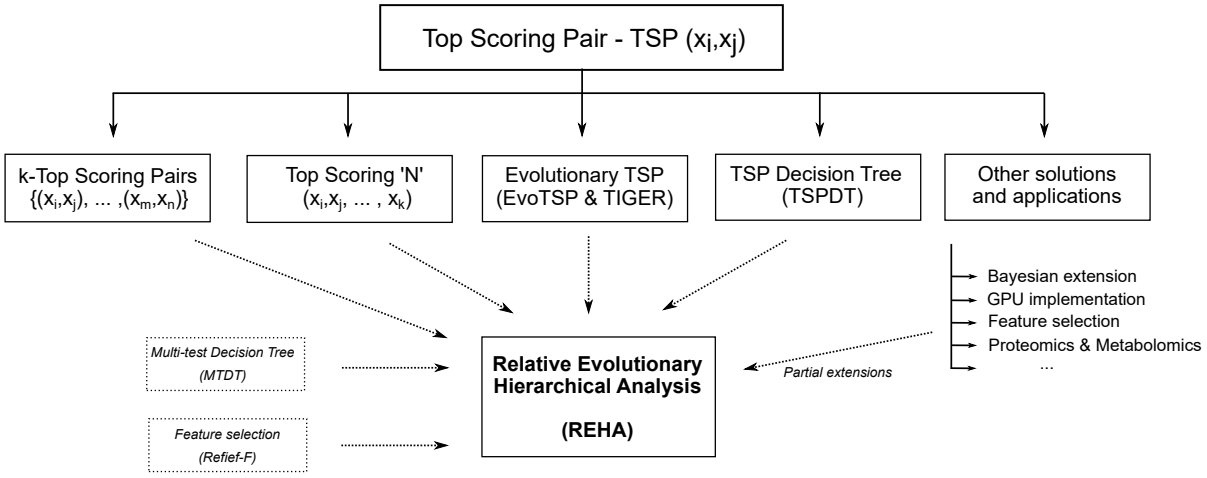

Figure 1: The general taxonomy of the family of Top Scoring Pair algorithms and relations with the REHA system.

or TSPDT and due to the computational complexity, the potentially hybrid solutions like k-TSN or decision tree with TSN were never published.

One of the ways to extend the search for more complex relations between the genes is the application of some heuristic methods. The EvoTSP and TIGER algorithms managed to limit the aforementioned drawbacks of RXA algorithms through the evolutionary approach. Proposed specialized EAs search for the weight top scoring pairs and allows exploring larger solution space. As they do not calculate all combinations of genes, they require less computation time in the analysis. However, their main disadvantage is that they consider only 'flat' rules and do not offer to find any sub-relations within the genes. In fact, the main and the only objective of the current RXA solutions is the accuracy of prediction. This implies that the pairs that constitute all aforementioned systems may not really be related or co-expressed. It can be compared to the random forests where each tree participates in the voting and in the final decision but no relation between the individual trees is searched or desired. Proposed REHA algorithm addresses and at least limits, the aforementioned problems.

3 EVOLUTION OF HIERARCHICAL CO-EXPRESSED GENE CLUSTERS

In this section, we present our Relative Evolutionary Hierarchical Analysis (REHA) algorithm. The proposed solution overall structure is based on a typical evolutionary algorithm (EA) schema [24] with an unstructured population and generational selection (ranking linear selection and elitist strategy are applied).

3.1 Representation

Associations between the genes may be represented by complicated structures in which the number of relations, their character and the number of affected genes is not known in advance. That is why we opted for a hierarchical tree-based representation in which individuals are processed in their actual form. Knowledge representation

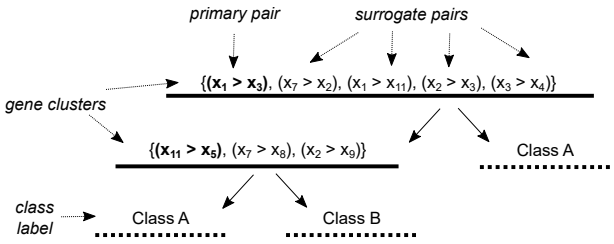

Figure 2: An example representation of REHA

structure is made up of nodes and branches, where: each internal node is associated with a gene cluster; each branch represents the split outcome, and each leaf (terminal node) is designed by a class label. Figure 2 illustrates an example tree representation of an induced REHA model.

Building blocks of a gene cluster are pairs of genes, each of which has a representation as in the TSP algorithm. A gene cluster can be created with a set of any gene pairs, as shown in Figure 2. This allows for the gene cluster to unify previous TSP extensions as it covers all possible relations not limited to disjoint gene pairs. In addition, each gene has additional knowledge of discrimination power rank, that is provided in REHA input and later applied in population initialization; gene cluster searches; different variants of genetic operators and fitness. In our research, we used the Relief-F [26] algorithm, which is commonly applied in the feature selection of gene expression data [15]. If necessary, the list of ranked genes submitted to the REHA can also be manually modified, for example, to focus on biomarker genes for a given disease. It should be noticed that at this step no attributes are automatically excluded from the dataset, so the REHA solution can work on all available genes. In this way, the algorithm is able to find interesting relationships also in low ranked genes. This would be not possible if the standard feature selection was applied as it takes place in most of studies.

We define a single split *ht* as a gene cluster in which pairs of genes are tightly linked and could participate in a common path in a hierarchical structure. A similar solution can be identified in the

multi-test decision trees [9] in which a split in each non-terminal node that is composed of several univariate tests that branch out the tree in a similar way. The reason for adding further tests in [9] was motivated by reducing the classifier under-fitting the data due to the low complexity of the classification rule. We believe that this solution can also be applied to finding groups of co-expressed and related epistasis genes.

In each node, we identify a pair of genes marked as *primary pair* (*pp*). The remaining pairs in the set are called *surrogate pairs* (*sp*). Each surrogate pair is constructed with at least one different attribute to the primary pair. It is ranked in terms of how good it mimics the primary pair measured by the number of the same observations that go to the corresponding sub-groups. This way the surrogate pairs support the division of instances carried out by the primary pair, but with the use of remaining genes. The measure of similarity, denoted as resemblance (r_{ij}), for a gene cluster located in the i-th node (ht_i) between the j surrogate pair (sp_{ij}) and the primary pair (pp_i) is the number of observations routed in the same way to all observations in the node:

$$r_{ij} = \frac{|X_{ij}|}{|X_i|}, \tag{4}$$

where $|X_{ij}|$ is the number of instances routed by sp_{ij} in the same way as pp_i, and (X_i) is the set of instances in the i-th node.

The splitting criterion is guided by a majority voting mechanism in which all pair components of the gene cluster have the same weight. This way surrogate pairs have a considerable impact (positive or negative) on gene cluster decisions, as they can prevail over the primary pair decision. In case of a draw, the vote of the primary pair is decisive.

3.2 Initialization

In order to maintain a balance between exploration and exploitation, initial individuals are created by using a simple top-down algorithm with randomly selected sub-samples of original training data. In each internal node of the hierarchical REHA structure, the system divides the training instances that reach this particular point with the decision determined by the gene cluster. The algorithm for creating a new hierarchical split ht_i works as follows. In the beginning, the first attribute of the primary pair pp_i is selected. Due to a large number of possible genes, we have applied an exponential ranking selection [3] to more often include top genes from the data. To find an attribute a list of possible TSP-like pair candidates is created from a subset of available genes (default: 50 genes selected alike the first attribute due to performance reasons). The criterion for selecting the best pair that will become pp_i was inspired by decision trees and is based on the information gain metric [4].

Next, a random even number (default: $j < 6$) of the sp_{ij} surrogate pairs is created, each on different attributes. The procedure for searching a surrogate pair attribute is analogous to the primary pair, but a second attribute that constitutes the pair is chosen in a slightly different way. Instead of the search for a pair that has the highest information gain, the algorithm searches for one that is more likely to branch instances like pp_i. As finding the surrogate pairs is performed much more often, for the performance reasons only 20 genes are selected as potential candidates to create sp_i.

3.3 Operators

In order to preserve genetic diversity, the GDT system applies two specialized genetic meta-operators corresponding to the classical mutation and crossover. Both operators may have a two-level influence on the individuals as either hierarchical tree structure either a gene cluster can be modified. Depending on the level, different aspects are taken into account to determine the crossover or mutation point. If the change considers the overall hierarchical structure, the level of the tree is taken into account. The modification of the top levels is performed less frequently than the bottom parts as the change would have a much bigger, global impact. The probability of selection is proportional to the rank in a linear manner. Examples of such variants are adding/deleting a node in the case of mutation and tree-branch crossover.

If the change considers the gene clusters their quality is taken into account as the ones with the higher prediction, per instance, are more likely to be changed. However, if the change considers single surrogate pair within a gene cluster, the resemblance ranking is used. In the case of mutation it can be: changing, removing or adding surrogate pairs of genes; modifying a primary pair by changing one or both attributes; or switching primary pair with one of the randomly selected surrogate pairs. The last two variants require updating the surrogates by deleting one of their attributes and finding a new one like in the general surrogate pair construction. Crossover variants allow whole gene clusters exchange as well as randomly selected pairs of genes between the individuals.

3.4 Fitness

In the case of hierarchical classification, it is recommended to maximize the accuracy and minimize the complexity of the output tree. However, in the case of gene expression data, these criteria cannot be applied directly. The main reason is the large disproportion between the number of instances and the attributes, which may cause the classifier to underfit the learning data as even a simple model can predict training data perfectly. On the other hand, complex gene clusters can be more difficult to analyze and interpret.

Considering our motivations and goals, the desired hierarchical classifier should have gene clusters consisting of several highly ranked pairs that branch out the nodes in a similar way. Therefore, the proposed fitness function should promote individuals with:

a) high accuracy on the training set;
b) relatively large size of gene clusters;
c) high resemblance of the gene pairs;
d) low cost of attributes that constitute the cluster.

Therefore, the REHA system maximizes the fitness function, which has the following form:

$$Fitness(H) = Q(H) + alpha * R(H) - beta * Cost(H), \tag{5}$$

where: $Q(H)$ is the accuracy, $R(H)$ is the sum of $R(H_i)$ in all nodes of the H hierarchy, $Cost(H)$ is the sum of the costs of attributes constituting clusters. The default parameters values are: $alpha = 0.2$ and $beta = 0.2$, and more information on tuning these parameters can be found in the next section.

Table 1: Details of gene expression datasets for tuning (t) and testing (a)-(h). The dataset names with abbreviation, number of genes and number of instances.

Datasets	Genes	Instances	Description
(t) GSE4290	22215	180	Sepsis
(t) GSE5772	54675	94	Glioma tumor
(a) GDS2771	22215	192	Lung cancer
(b) GSE17920	54676	130	Hodgkin lymphoma
(c) GSE25837	18631	93	Chronic loneliness
(d) GSE3365	22284	127	Inflammatory Bowel disease
(e) GSE10072	22284	107	Lung adenocarcinoma
(f) GSE19804	54613	120	Lung cancer
(g) GSE27272	24526	183	Impact of tobacco smoke
(h) GSE6613	22284	105	Parkinson's disease

Table 2: REHA parameters

Basic EA parameters	
Population size:	100 individuals
Elitism rate:	1% of the population
Max generations:	1000
Mutation rate:	90% assigned to the individual
Crossover rate:	10% assigned to the individual

Let us consider an internal H_i node with ht_i gene cluster. Then:

$$R(H_i) = \frac{|X_i|}{|X|} * \sum_{j=1}^{|ht_i|-1} r_{ij}, \tag{6}$$

where X is a learning set, X_i is a set of instances in i-th node, and $|ht_i|$ is the size of a gene cluster. If a gene cluster is composed of a single pair, then $R(H_i)$ equals 0.

The cost of attributes in the cluster ht_i depends on their rank, and the number of instances that reach the i node:

$$Cost(H_i) = \frac{|X|}{|X_i|} * (C(pp_i) + \sum_{j=1}^{|ht_i|-1} C(sp_{ij})), \tag{7}$$

where j is the number of sp in i-th cluster, $C(pp_i)$ and $C(sp_{ij})$ are the costs of the pairs equal to the sum of their attributes (genes) costs. The cost of a gene range from 0 and 1, while 0 corresponds to the highest ranked gene and 1 is equal to the worst ranked gene. The reason why $Cost(H_i)$ increases when the number of instances in a node decreases is to avoid the overfitting in the lower parts of the hierarchy, as this will eventually limit the size of the cluster.

4 EXPERIMENTS

In this section, we present a detailed experimental analysis to evaluate the relative performance of the proposed evolutionary hierarchical gene cluster approach. Using several cancer-related gene expression datasets we have checked REHA prediction power and confronted its results with popular RXA extensions.

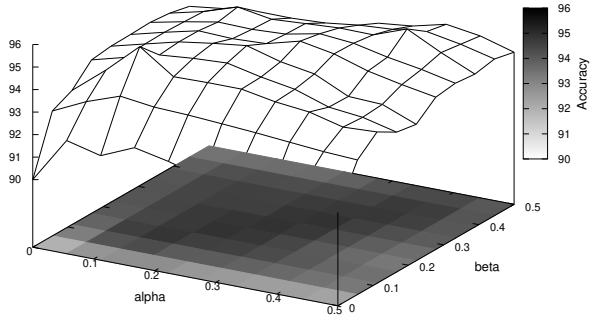

Figure 3: Impact of the resemblance: alpha and cost: beta parameters on REHA system performance

4.1 Setup

To make a proper comparison with the RXA algorithms, we used the same 8 cancer-related benchmark datasets that were tested in the TIGER solution [6]. Additional two datasets were used for REHA parameter tuning. Datasets are deposited in NCBI's Gene Expression Omnibus [5] and summarized in Table 1. A typical 10-fold cross-validation was used and the testing of different RXA algorithms. Depending on the algorithm, different tools were used to test the algorithms:

- TSP, TSN, and k-TSP was performed with the AUERA software [11], which is an open-source system for identification of relative expression molecular signatures;
- EvoTSP and TIGER results were taken from the publication as exactly the same datasets were used [6]
- TSPDT results were provided us by the authors [8].

Unfortunately, algorithms TSP, k-TSP, TSN and TSPDT use exhaustive search and it is difficult to test them on whole datasets. The AUERA software automatically performs some feature selection due to the performance reasons and TSPDT is so computationally demanding that it could not induce tree in a reasonable time. For that case, the Relief-F feature selection was used and the number of selected genes was arbitrarily limited to the top 1000 to allow the algorithms also work with low-ranked features.

In all experiments, a default set of parameters for all algorithms is used in all tested datasets and the presented results correspond to averages of 20 runs. Considering that REHA is regular generational EA, parameters such as population size, the maximum number of generations, elitism rate, crossover and mutation probability must be selected before evolution. Table 2 contains a brief listing of the main parameters that have been used, however, more research on tuning those settings is required.

Besides the typical evolutionary parameters, the REHA algorithm requires two additional ones *alpha* and *beta* used in individuals evaluation. The fitness function described in Equation 5 has three objectives: accuracy, the resemblance of surrogate pairs and cost of attributes. The role of parameters *alpha* and *beta* is to control

Table 3: Comparison of top-scoring algorithms, including accuracy with its standard deviation and the number of unique genes that build classifier's model. The highest classifiers accuracy for each dataset was bolded.

DT	TSP	TSN	k-TSP		EvoTSP		TIGER		TSPDT		REHA	
	accuracy	accuracy	accuracy	size	accuracy	size	accuracy	size	accuracy	size	accuracy	size(attr.)
(a)	57.2 ± 2.4	61.9 ± 2.8	62.9 ± 3.3	10	65.6 ± 2.0	4.0	72.7 ± 3.6	2.8	60.1± 3.2	16.4	**76.0** ± 2.0	2.2 (15)
(b)	88.7 ± 2.6	89.4 ± 2.1	90.1 ± 2.5	6	96.5 ± 1.3	2.1	97.4 ± 0.6	2.0	**98.2** ± 1.3	2.0	97.3± 1.0	2.2 (4)
(c)	64.9 ± 3.5	63.7 ± 4.7	67.2 ± 3.2	10	78.1 ± 2.6	2.8	78.0 ± 3.5	3.1	72.3 ± 2.2	6.8	**81.2** ± 2.9	4.1 (7)
(d)	93.5 ± 1.7	92.8 ± 1.5	94.1 ± 1.6	10	96.2 ± 1.1	2.1	93.0± 1.6	2.0	88.3 ± 3.5	3.0	**97. 1**± 3.0	2.0 (2)
(e)	56.0 ± 4.0	60.5 ± 5.1	58.4 ± 4.0	14	66.9 ± 5.6	3.1	75.9 ± 2.8	4.0	68.1 ± 5.1	5.7	**76.7** ± 1.1	2.8 (7)
(f)	47.3 ± 4.8	50.1 ± 3.8	56.2 ± 2.2	18	66.2 ± 1.1	2.7	65.4 ± 1.9	3.0	67.2 ± 7.0	11.9	**71.2** ± 1.0	2.5 (15)
(g)	81.9 ± 2.6	84.2 ± 2.7	87.2 ± 2.1	14	86.1 ± 2.8	4.1	89.5 ± 2.1	3.0	88.6 ± 0.6	4.3	**91.4** ± 1.4	2.4 (12)
(h)	49.5 ± 3.5	51.7 ± 2.8	55.8 ± 5.3	10	53.6 ± 5.4	6.1	64.4 ± 4.0	5.3	59.6 ± 8.0	7.2	**71.0** ± 3.6	2.0 (11)
Avg.	67.4 ± 3.3	69.3 ± 3.2	71.5 ± 3.0	11.5	76.2 ± 2.7	3.4	79.5 ± 2.5	3.1	75.3 ± 3.8	7.2	**82.7** ± 2.0	2.3 (9)

the relative importance of resemblance and cost. Figure 3 shows the parameter tuning experiment varying *alpha* and *beta* within {0.0, 0.1, 0.2,0.3, 0.4, 0.5}. To select the best value of *alpha* and *beta* we used two external tuning datasets (Table 1) of gene expression data as the aim of the experiment is not to optimize the parameters for a given dataset, but to find robust values that generally work well in the domain. We then used best values of *alpha* and *beta* to evaluate the generalizing ability of the fitness function in the tested datasets (a)-(h) from Table 1. We can observe that the impact of resemblance and costs objectives is not high but it is well balanced as the highest accuracy is achieved when both parameters *alpha* and *beta* are equal and greater than 0. Of course, it is not possible to draw meaningful conclusions on the basis of only 2 sets.

4.2 REHA vs popular RXA algorithms

Table 3 summaries classification performance for the proposed solution and it's competitors. The model size of TSP and TSN is not shown as it is fixed and equals correspondingly 2 and 3. It is clearly visible that the proposed REHA solution managed to outperform all popular RXA classifiers in 7 out of 8 datasets. The statistical analysis of the obtained results using the Friedman test and the corresponding Dunn's multiple comparison test (significance level/p-value equals 0.05), as recommended by Demsar [10] showed that the differences in accuracy are significant. We have also performed an additional comparison between the datasets with corrected paired t-test [25] with the significance level equals 0.05 and 9 degrees of freedom (n-1 degrees of freedom where n = 10 folds). It showed that REHA significantly outperforms all algorithms that apply exhaustive search on 7 datasets except the dataset (b) where there are no statistical differences between TSPDT were found.

However, it should be noticed that improving classification accuracy was not our primary goal. We wanted to make a model in which gene pairs somehow interact with each other and also to promote finding sub-interactions between co-expressed genes and pairs. Such improvement in terms of classification accuracy was a surprise even for us, however, this indicates the importance of the patterns found. In addition, the size of the REHA model is much smaller than another tree-structure system called TSPDT in terms of size (height of the tree) as well as the number of unique genes that constitute the pairs in each split/cluster.

4.3 Case study

In this section, we would like to check if results returned by the REHA solutions are biologically meaningful. We focused on the first tested dataset (a) denoted as GDS2771 that contains an analysis of large airway epithelial cells from cigarette smokers without cancer, with cancer, and with suspect lung cancer. The goal was to provide insight into the feasibility of using gene expression to detect early-stage lung cancer in smokers [14].

In the Figures 4 and 5, we want to show wherever the gene cluster and the cost information affect the basic statistics of the best individual founded so far in the evolution. We track the following statistics:

- hierarchy size (number of internal nodes);
- average gene cluster size (number of gene pair);
- accuracy on the testing set;
- average resemblance of the pairs in cluster;
- average percent of pairs in cluster involving at least one top-ranked attribute (top 20 attributes with the highest rank / lowest cost);
- average percent of pairs in cluster involving at least one low-ranked attribute (attributes with rank over 200);

for the REHA algorithm and its variant without cost information about the genes (denoted as REHA_{NC}).

We enclose the average statistics for the best individual in each generation for REHA and REHA_{NC} (see Figures 4 and 5).Accuracy on the training set is not shown as it achieves 100% in the first 50-100 iterations and might obscure the figures.

There are a few things that should be noticed. The final size of the REHA structure is found in less than a few hundred iterations, but the search of gene clusters required more time (see Figures 4). A longer adjustment is due to a large number of degrees of freedom - its size, the resemblance of the pairs and the cost of the attributes. However, despite gene cluster changes, the prediction performance is stable, which confirms the robustness of the REHA model. In addition, Figure 4 shows the percentage share of highest/lower rank attributes that constitute the pairs. We can see that REHA prediction model uses the pairs that are similar in more than 85% decisions and are based on 85% of the top ranked attributes. Such high quality of gene clusters improves the stability of internal nodes and thus the whole classifier.

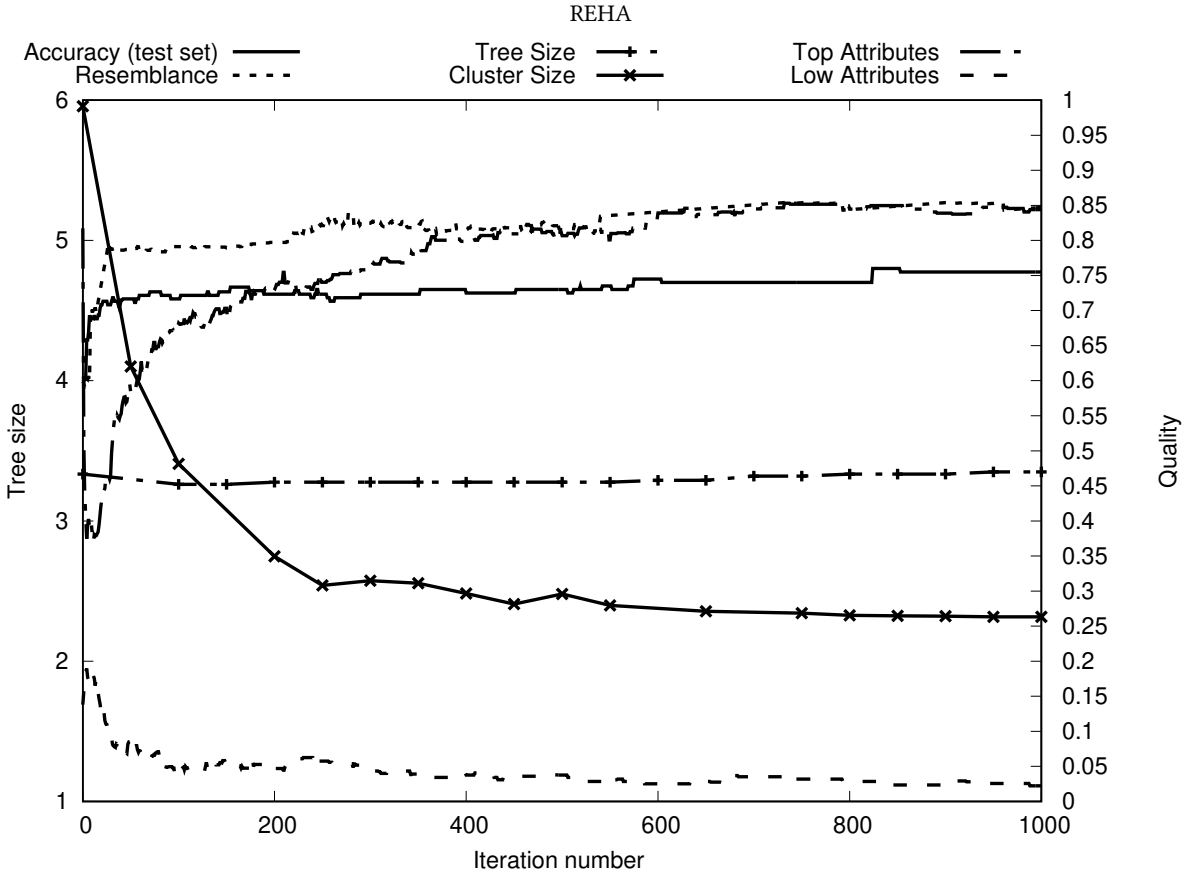

Figure 4: The performance of the best individual founded so far on GDS2771 lung cancer dataset for the REHA system.

Figure 5 illustrates also how our solution would work without incurring costs (rank of attributes). We see that the best individual for REHA$_{NC}$ has a similar tree size, but, unlike REHA, much smaller gene clusters. As the evolution progresses, the classifier reduces its complexity and improves the resemblance of the clusters. With the iteration number, the prediction accuracy on the testing set decreases, suggesting that the REHA$_{NC}$ slowly underfits to the training data. In addition, we see that for REHA$_{NC}$ the percentage of top attributes in clusters is around 30% which is almost 3-times smaller than in the case of REHA. Combination of small gene clusters with the noisy or insignificant attributes is probably responsible for poor REHA$_{NC}$ performance.

Based on the description of the dataset (GSE4115 series) from GenBank NCBI [2] we conducted a thorough examination of one of the REHA output prediction models (see Figure 6). In 9 gene pairs, there was a total of 16 different genes as some of them occur more than once either in the same cluster but in a different pair or in a sub-cluster. To check if founded genes, gene pairs, and gene clusters have some biological meaning we have decoded gene names from GDS2771 with GPL96 platform provided by NCBI. We found out that 11 founded genes are directly related to lung cancer, for example, the HBA1 gene (#211699) impacts on survival of lung cancer patients with diabetes mellitus and with a few other detected genes focus on the same mutations [27].

5 CONCLUSIONS

Achieving high classification accuracy for gene expression datasets is still a major problem. When searching for white-box solutions researches and biologists often use classification algorithms based on RXA because of their simplicity and relatively high prediction power. However, it is not always enough. Such methods are capable of finding interesting patterns but are limited to gene-gene relations within distinct pairs. We propose a new approach, based on gene clusters, that allows searching for complex relations including sub-groups of co-expressed genes. Our hierarchical solution called REHA not only significantly improves the accuracy of predictions but also replaces extremely computational demanding RXA exhaustive search with an evolutionary heuristic. The efficiency of the solution is boosted by embedding additional information about the discriminative power of genes in the evolutionary process, carefully design fitness function and genetic operators. We also managed to unify all major variants of RXA solutions within REHA.

Preliminary experiments show that the knowledge discovered by REHA is supported by biological evidence in the literature. A biologist can, therefore, benefit from this 'white box' approach, as

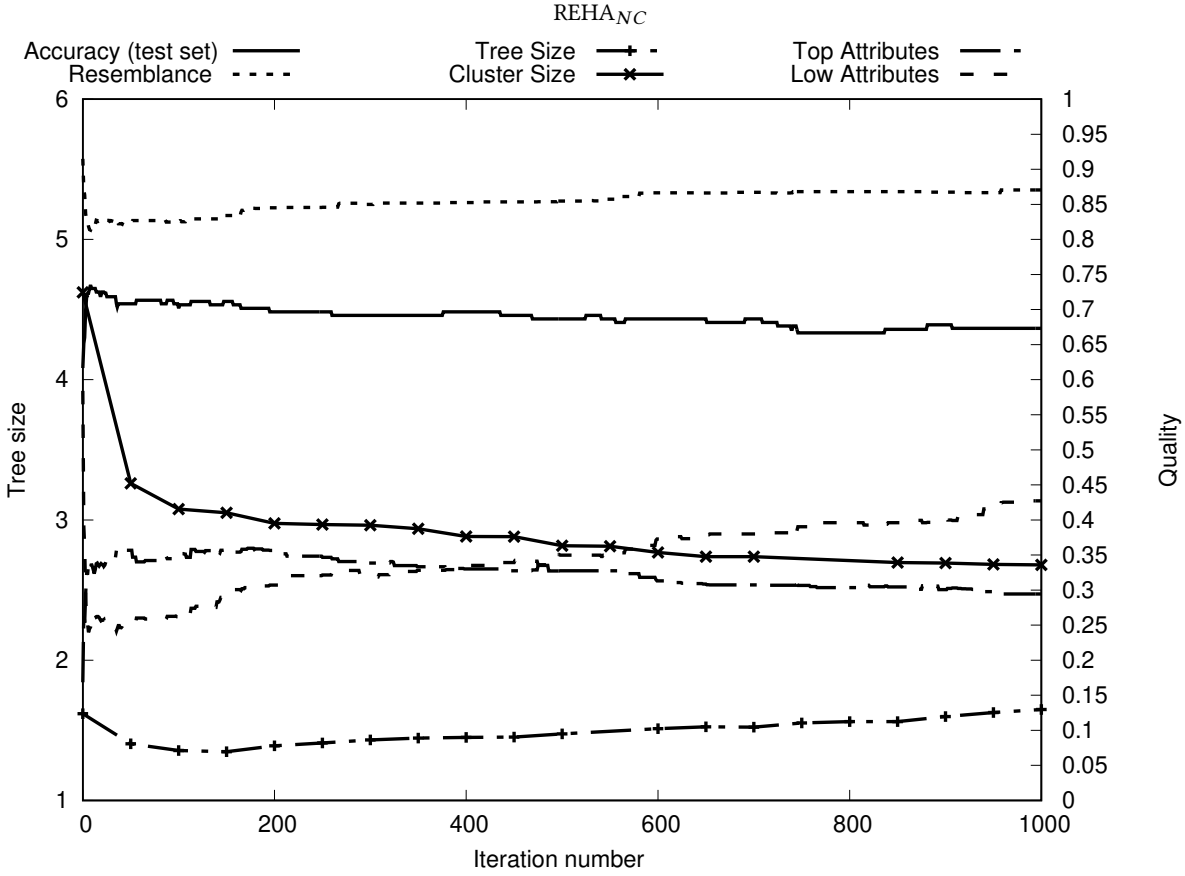

Figure 5: The performance of the best individual founded so far on GDS2771 lung cancer dataset for the REHA$_{NC}$ system.

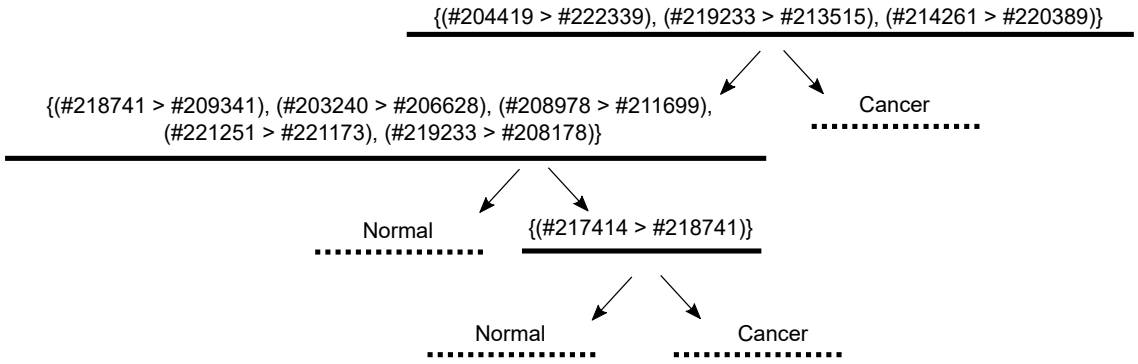

Figure 6: An example hierarchical structure induced by REHA for GDS2771 lung cancer dataset.

it can produce accurate and meaningful classification models and reveal new patterns in genomic data.

We see many promising directions for future research. In particular, from a machine learning point of view, we are focusing on parameter tuning and optimization. On the other side, we are currently working with biologists and bioinformaticians to better understand the gene clusters generated by REHA. We also want to adopt the proposed classification algorithm to work with protein expression and metabolic databases as well as any other omics data.

Acknowledgments. This work was supported by the grant MB/WI/1/2017 (first author) and S/WI/2/18 (second author) from BUT founded by Polish Ministry of Science and Higher Education

REFERENCES

[1] Rodrigo C. Barros, Márcio P. Basgalupp, Aléx A. Freitas, and Andre C.P.L.F. De Carvalho. 2014. Evolutionary design of decision-tree algorithms tailored to microarray gene expression data sets. *IEEE Transactions on Evolutionary Computation* (2014). https://doi.org/10.1109/TEVC.2013.2291813

[2] Dennis A. Benson, Mark Cavanaugh, Karen Clark, Ilene Karsch-Mizrachi, James Ostell, Kim D. Pruitt, and Eric W. Sayers. 2018. GenBank. *Nucleic Acids Research* (2018). https://doi.org/10.1093/nar/gkx1094 arXiv:1611.06654

[3] Tobias Blickle and Lothar Thiele. 1996. A comparison of selection schemes used in evolutionary algorithms. *Evolutionary Computation* 4, 4 (1996), 361–394. https://doi.org/10.1162/evco.1996.4.4.361

[4] Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone. 2017. *Classification and regression trees.* https://doi.org/10.1201/9781315139470 arXiv:arXiv:1011.1669v3

[5] Emily Clough and Tanya Barrett. 2016. The Gene Expression Omnibus database. In *Methods in Molecular Biology.* https://doi.org/10.1007/978-1-4939-3578-9_5

[6] Marcin Czajkowski, Anna Czajkowska, and Marek Kretowski. 2016. TIGER: an evolutionary search for Top Inter-GEne Relations. *International Journal of Data Mining and Bioinformatics* (2016). https://doi.org/10.1504/IJDMB.2016.080042

[7] Marcin Czajkowski, Marek Grześ, and Marek Kretowski. 2014. Multi-test decision tree and its application to microarray data classification. *Artificial Intelligence in Medicine* 61, 1 (2014), 35–44. https://doi.org/10.1016/j.artmed.2014.01.005

[8] Marcin Czajkowski and Marek Kretowski. 2011. *Top Scoring Pair Decision Tree for Gene Expression Data Analysis.* Springer New York, New York, NY, 27–35. https://doi.org/10.1007/978-1-4419-7046-6_3

[9] Marcin Czajkowski and Marek Kretowski. 2014. Evolutionary approach for relative gene expression algorithms. *The Scientific World Journal* (2014). https://doi.org/10.1155/2014/593503

[10] Janez Demsar. 2006. Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research 7* (2006). https://doi.org/10.1016/j.jecp.2010.03.005 arXiv:arXiv:1011.1669v3

[11] John C. Earls, James A. Eddy, Cory C. Funk, Younhee Ko, Andrew T. Magis, and Nathan D. Price. 2013. AUREA: An open-source software system for accurate and user-friendly identification of relative expression molecular signatures. *BMC Bioinformatics* (2013). https://doi.org/10.1186/1471-2105-14-78

[12] James A. Eddy, Jaeyun Sung, Donald Geman, and Nathan D. Price. 2010. Relative expression analysis for molecular cancer diagnosis and prognosis. https://doi.org/10.1177/153303461000900204

[13] Donald Geman, Christian D'Avignon, Daniel Q. Naiman, and Raimond L. Winslow. 2004. Classifying Gene Expression Profiles from Pairwise mRNA Comparisons. *Statistical Applications in Genetics and Molecular Biology* (2004). https://doi.org/10.2202/1544-6115.1071 arXiv:NIHMS150003

[14] Adam M. Gustafson, Raffaella Soldi, Christina Anderlind, Mary Beth Scholand, Jun Qian, Xiaohui Zhang, Kendal Cooper, Darren Walker, Annette Mcwilliams, Liu Gang, Eva Szabo, Jerome Brody, Pierre P. Massion, Marc E. Lenburg, Lam Stephen, Andrea H. Bild, and Avrum Spira. 2010. Airway PI3K pathway activation is an early and reversible event in lung cancer development. *Science Translational Medicine* (2010). https://doi.org/10.1126/scitranslmed.3000251

[15] Zena M. Hira and Duncan F. Gillies. 2015. A review of feature selection and feature extraction methods applied on microarray data. *Advances in Bioinformatics* 2015 (2015). https://doi.org/10.1155/2015/198363

[16] Xin Huang, Xiaohui Lin, Lina Zhou, and Benzhe Su. 2018. Analyzing omics data by pair-wise feature evaluation with horizontal and vertical comparisons. *Journal of Pharmaceutical and Biomedical Analysis* (2018). https://doi.org/10.1016/j.jpba.2018.04.052

[17] Dimitri Kagaris, Alireza Khamesipour, and Constantin T. Yiannoutsos. 2018. AUCTSP: An improved biomarker gene pair class predictor. *BMC Bioinformatics* (2018). https://doi.org/10.1186/s12859-018-2231-1

[18] Parminder Kaur, Daniela Schlatzer, Kenneth Cooke, and Mark R. Chance. 2012. Pairwise protein expression classifier for candidate biomarker discovery for early detection of human disease prognosis. *BMC Bioinformatics* (2012). https://doi.org/10.1186/1471-2105-13-191

[19] K.-M. Lin, J Kang, H Shin, and J Lee. 2009. A cube framework for incorporating inter-gene information into biological data mining. *International Journal of Data Mining and Bioinformatics* (2009). https://doi.org/10.1504/IJDMB.2009.023881

[20] Xiaohui Lin, Jiuchong Gao, Lina Zhou, Peiyuan Yin, and Guowang Xu. 2014. A modified k-TSP algorithm and its application in LC-MS-based metabolomics study of hepatocellular carcinoma and chronic liver diseases. *Journal of Chromatography B: Analytical Technologies in the Biomedical and Life Sciences* (2014). https://doi.org/10.1016/j.jchromb.2014.05.044

[21] Xiaohui Lin, Jiuchong Gao, Lina Zhou, Peiyuan Yin, and Guowang Xu. 2014. A modified k-TSP algorithm and its application in LC-MS-based metabolomics study of hepatocellular carcinoma and chronic liver diseases. *Journal of Chromatography B: Analytical Technologies in the Biomedical and Life Sciences* (2014). https://doi.org/10.1016/j.jchromb.2014.05.044

[22] Andrew T. Magis, John C. Earls, Youn Hee Ko, James A. Eddy, and Nathan D. Price. 2011. Graphics processing unit implementations of relative expression analysis algorithms enable dramatic computational speedup. *Bioinformatics* (2011). https://doi.org/10.1093/bioinformatics/btr033

[23] Andrew T. Magis and Nathan D. Price. 2012. The top-scoring 'N' algorithm: a generalized relative expression classification method from small numbers of biomolecules. *BMC Bioinformatics* (2012). https://doi.org/10.1186/1471-2105-13-227

[24] Zbigniew Michalewicz. 1996. *Genetic algorithms + data structures = evolution programs (3rd ed.).* https://doi.org/10.2307/2669583

[25] Claude Nadeau and Yoshua Bengio. 2003. Inference for the generalization error. *Machine Learning* (2003). https://doi.org/10.1023/A:1024068626366

[26] Marko Robnik-Šikonja and Igor Kononenko. 2003. Theoretical and Empirical Analysis of ReliefF and RReliefF. *Machine Learning* (2003). https://doi.org/10.1023/A:1025667309714 arXiv:arXiv:astro-ph/0005074v1

[27] Katie E. Rollins, Krishna K. Varadhan, Ketan Dhatariya, and Dileep N. Lobo. 2016. Systematic review of the impact of HbA1c on outcomes following surgery in patients with diabetes mellitus. https://doi.org/10.1016/j.clnu.2015.03.007

[28] Shilpi Shandilya and Chaitali Chandankhede. 2018. Survey on recent cancer classification systems for cancer diagnosis. In *Proceedings of the 2017 International Conference on Wireless Communications, Signal Processing and Networking, WiSPNET 2017.* https://doi.org/10.1109/WiSPNET.2017.8300231

[29] Ping Shi, Surajit Ray, Qifu Zhu, and Mark A. Kon. 2011. Top scoring pairs for feature selection in machine learning and applications to cancer outcome prediction. *BMC Bioinformatics* (2011). https://doi.org/10.1186/1471-2105-12-375

[30] Jonatan Taminau, Stijn Meganck, Cosmin Lazar, David Steenhoff, Alain Coletta, Colin Molter, Robin Duque, Virginie de Schaetzen, David Y. Weiss Solís, Hugues Bersini, and Ann Nowé. 2012. Unlocking the potential of publicly available microarray data using inSilicoDb and inSilicoMerging R/Bioconductor packages. *BMC Bioinformatics* (2012). https://doi.org/10.1186/1471-2105-13-335

[31] Aik Choon Tan, Daniel Q. Naiman, Lei Xu, Raimond L. Winslow, and Donald Geman. 2005. Simple decision rules for classifying human cancers from gene expression profiles. *Bioinformatics* (2005). https://doi.org/10.1093/bioinformatics/bti631

[32] Kaimin Wu, Xiaofei Nan, Yumei Chai, Liming Wang, and Kun Li. 2016. DTSP-V: A trend-based Top Scoring Pairs method for classification of time series gene expression data. In *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM).* 1787–1794. https://doi.org/10.1109/BIBM.2016.7822790

[33] Hongyan Zhang, Haiyan Wang, Zhijun Dai, Ming shun Chen, and Zheming Yuan. 2012. Improving accuracy for cancer classification with a new algorithm for genes selection. *BMC Bioinformatics* (2012). https://doi.org/10.1186/1471-2105-13-298