Biocybernetics and Biomedical Engineering
2007, Volume 27, Number 3, pp. 29–42

Decision Tree Approach to Microarray Data Analysis

MAREK GRZEŚ*, MAREK KRĘTOWSKI

Faculty of Computer Science, Białystok Technical University, Białystok, Poland

The classification of gene expression data is still new, difficult and also an interesting field of endeavour. There is a demand for powerful approaches to this problem, which is one of the ultimate goals of modern biological research. Two different techniques for inducing decision trees are discussed and evaluated on well-known and publicly available gene expression datasets. Empirical results are presented.

Keywords: classification, decision trees, gene expression, microarray

1. Introduction

In1953 Watson and Crick published in Nature article "Molecular Structure of Nucleic Acids", which marked a new era of genomic research. Identification of the structure and functions of DNA sequences of different organisms since then has been one of real challenges. A significant milestone in genomic research was the advent of DNA microarray technology, which became the most widely used technique for dealing with expression of thousands of genes simultaneously.

Gene expression data can be obtained by two common microarray platforms: complementary DNA (cDNA) developed at Stanford University and oligonucleotide microarrays (GeneChip) invented by Affymetrix. These high-throughput technologies allow investigating the gene expression under various experimental conditions and stages of different tissues.

DNA sequences play an important role in the process of producing proteins. From the DNA sequences, in the process called transcription, messenger RNA (mRNA) is generated. The amount of produced, for a particular gene, mRNA depends on whether that gene is expressed or not. The importance of mRNA in assembling proteins from their building blocks is playing a role of the template which guides the process. In the

[*] Correspondence to: Marek Grześ, Faculty of Computer Science, Białystok Technical University, Wiejska 45A, 15-351 Białystok, Poland, e-mail: marekg@ii.pb.bialystok.pl

gene expression microarray experiment, the expression means the amount of mRNA corresponding to a particular gene in the investigated tissue [16].

Gene expression profile data are usually organised in a matrix of n rows and m columns. The rows correspond to genes (usually genes of the whole genome, with some replications for the quality check) and columns represent samples (e.g. various patients or different tissues of the same patient under various conditions).

At the time, when DNA microarray technology is still new and its experiments remain expensive, gathered gene expression data are characterised as having a large number of measurements (usually up to 50000), with relatively few samples (no more than 100). This fact poses a real challenge to all kinds of techniques and algorithms used to investigate such data with a high ratio of variables/cases. Because of high dimensionality, pre-processing steps, including normalization and feature selection, are usually applied before the intentional analysis which can fall into three broad categories: class discovery, class prediction (identifying classes of unknown samples) and finding differentially expressed genes [18]. Two main techniques that are applied to achieve these goals are the cluster analysis (identification of new subgroups or classes of some biological entities; its two-dimensional version is known as biclustering [19]) and the discriminant analysis (classification of entities into known classes).

Classical approaches (including statistics) have been widely used for investigation of gene expression matrices. Techniques such as hierarchical clustering [10] (applied to class discovery and finding relationships between genes and diseases), regression models [27], testing statistical hypotheses [27], projection methods, e.g. the principle component analysis, and many others have been used in the microarray experiments.

The rest of this paper is organised as follows: section 2 describes the machine learning approach to the analysis of the gene expression data and gives some examples of such applications; experimental results are presented in section 3; section 4 concludes the paper.

2. Machine Learning Approach

Machine learning could be treated as an alternative to the statistical approach. It focuses on constructing computer programs which can find solutions (learning from experience) to certain classes of tasks. These tasks fall into two wide categories: supervised learning, where the machine learning program is given some prior knowledge, and unsupervised learning, where the learning process is performed without any prior information. With reference to the previous section, it is noticeable that these types of machine learning programs suit well analysis of molecular biology data. Unsupervised learning can be, for instance, used as a means of the clustering method and supervised learning fulfils demands of the discriminant analysis.

Molecular biology belongs to the fields of science with a lot of data and with relatively little theory. It is impossible for an expert to investigate and analyse manually huge matrices, which creates considerable demand for more sophisticated solutions like statistics or machine learning. This makes machine learning perfectly suited for this area [2].

Machine learning techniques applied for supervised and unsupervised learning are also known to be excellent at discarding (feature selection) and compacting (feature extraction) redundant information [11]. It allows using them as a comprehensive tool to investigate gene expression data. In Figure 1 a scheme of machine learning environment is presented.

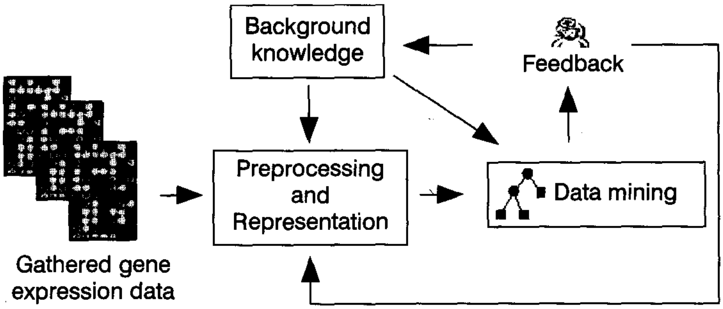
Background knowledge
Feedback
Preprocessing and Representation
Data mining
Gathered gene expression data

Fig. 1. The scheme of machine learning environment for the analysis of gene expression data

In the case of gene expression data, pre-processing turns out to be an important step. Before employing any algorithms it is worth considering how the analysed data are obtained and what they exactly mean (described in the first section). Values in gene expression matrices are obtained with an indirect measurement during physical process. We are not sure how precise and stable these results are. Because a phenomenon which has probably two states is investigated (gene is expressed or not), perhaps a simplified representation with binary values could be successfully used. Not only during pre-processing but also the data mining step of the analysis, the background knowledge, serves an invaluable source of information. Being aware which genes with a high probability correlate with certain classification problems, the algorithm can be led in the intended direction. Investigation of medical related data requires the feedback of the expert in this domain.

2.1. Related Works

Machine learning and other artificial intelligence techniques have been widely applied to the gene expression analysis. One of the first attempts in this field was published by Golub et al. [10]. They looked into both the cluster and the discriminant analysis of tumours. For class prediction, they proposed a weighted gene voting scheme (this attempt allows discarding redundant features). For class discovery, self-organising maps (SOM) were employed.

Because of the high ratio of variables/cases there are many works investigating the problem of selecting genes and subsets of genes. Guyon et al. [11] apply the support vector machine (SVM) for feature ranking. Lyons-Weiler et al. [18] select subset of features using the maximum difference subset (MDSS) principle.

As for SVM, it has been investigated widely as a means of classifier for molecular data (Shipp et al. [25], Ben-Dor et al. [3] employ SVM, the nearest neighbour classifier, AdaBoost and classification based on clustering).

There have been also, like in this paper, some attempts to use decision trees for the discriminant analysis of the gene expression data. Dudoit et al. [6] compare some classification principles, among which there is the CART system [4]. Tan et al. [26] present the application of C4.5, bagged and boosted decision trees. They use the ensemble scheme for classification, which can be also found in Valentini et al. [28], where bagged ensembles of SVM are presented. The committee of individual classifiers is also presented in [12], where ensembles of cascading trees are applied to the classification of the gene expression data.

2.2. Methods

In this paper, the discriminant analysis of the gene expression data is investigated. As cDNA microarray and high-density oligonucleodite chips allow monitoring of the expression levels of thousands of genes simultaneously, they may be used for a more precise understanding of molecular relationships among tumours and diseased tissues. It is important to construct classifiers that have a high predictive accuracy in classifying tumorous samples. In the case of the biology relevant investigations, a classifier needs to provide justification for its decision (i.e. prediction) and for selected discriminative genes. For an expert in this domain, the computer program can help understand biological processes and enable to make new discoveries. For this reason we decided to apply decision trees, which are usually easier understood comparing to "black-box" approaches.

Our experiment has been composed of two steps: pre-processing, where also the feature selection was performed, and intended classification.

2.2.1. Preprocessing

Five following steps which aimed at normalising and filtering genes not expressed differentially according to the class feature were applied.

i. Thresholding

   All values in gene expression matrix less than L_{min} were replaced by L_{min} and those greater than L_{max} by L_{max}. The default value of L_{min} (floor) for the investigated data was 100 and L_{max} (ceiling) 16000.

ii. Filtering
Genes with $max / min \leq 5$ or $(max - min) \leq 500$ were excluded from the expression matrix (*max* and *min* refer to the maximum and minimum value of the expression level for a particular gene respectively).

iii. t-test filtering
This step aimed at discarding genes not differentially expressed in two types of patients. It was performed with *ttest* function (Student's t-test) from the *genefilter* package [8]. Four values of *p-value* were used to obtain datasets with different amount of features.

iv. Log-transformation
All values in a gene expression matrix were converted into the logarithmic (to base 10) representation.

v. Standardisation
The expression matrix was standardised using the *scale* function [23].

The presented steps were performed using Bioconductor [7] packages in the R [23] system.

2.2.2. Induction and Classification/Prediction

Decision trees belong to the most popular predictive models in machine learning. They are usually easy to understand and interpret. People who are not acquainted with decision theory are able to understand them and find an explanation for their decisions.

The are two broad categories of decision trees which vary in the kind of tests used in non-terminal nodes. The decision tree is called axis parallel, if each test is based on a single attribute (e.g. C4.5 [22]). If, on the other hand, tests are based on more than one feature, such a tree is called multivariate. Oblique decision trees exemplify the particular kind of multivariate trees in which hyperplanes are used as tests (e.g. OC1 [20]).

Because the problem of inducing an optimal decision tree is very difficult (NP-complete), greedy heuristics are mostly employed. The most widely used approach is based on the splitting criterion. It is known as the top-down induction of decision trees and is used in most well-known algorithms [22, 20]. A decision tree is learned by recursive splitting the subset of examples based on the test (univariate or multivariate) in the current node. The procedure is recursively repeated for obtained subsets. Certain algorithms that follow this framework vary mainly in the stopping criterion and the way the tests are chosen. In our paper, C4.5 [22] was used as an renowned example of the axis parallel decision trees inducers, and OC1 for oblique decision trees induction [20]. These algorithms can be treated as the *de-facto* standards in empirical evaluations of decision trees.

Even though the aforementioned approach is robust and performs well with large data in a short time, it fails for certain problems and finds the local optimum. For this reason, our approach is based on global induction of decision trees and is presented as another group of algorithms analysed in this paper. The system is called GDT and produces both axis parallel (GDT-AP [15]) and oblique (i.e. based on hyperplanes) decision tress (GDT-OB [14]). The specialised evolutionary algorithm (individuals are encoded in a natural tree like form) is designed and implemented. It searches for the tree structure and tests (univariate or multivariate, respectively) in internal nodes at the same time.

The GDT system follows the general framework of evolutionary algorithms and problem specific details are presented in [15] for univariate and in [14] for oblique decision trees. With reference to the classification task presented in this paper the fitness function needs to be discussed. The fitness function, which is maximised, has the following form:

$$Fitness(T) = Q_{Reclass}(T) - \alpha \cdot S(T) \qquad (1)$$

where T is the evaluated tree, $Q_{Reclass}(T)$ the reclassification quality, $S(T)$ is the size of the tree T expressed as a number of nodes and α is the relative importance of the complexity term (user specified parameter).

Classification algorithms used to analyse the gene expression data have to cope with noisy and unnecessary features. In reality, biology related learning sets often contain irrelevant and noisy data, which should not be taken into account during the induction process. The ability to understand and properly interpret the classifier can be increased by simplifying tests. Furthermore, the elimination of noisy features can improve the overall performance of the classifier. In this case, axis parallel decision trees have advantage over other kinds of classification algorithms. Because the tests are univariate, the feature selection is naturally built-in into the induction process and any additional processes associated with feature selection are not necessary. The situation is exactly opposite in case of oblique tests based on hyperplanes. Because the evolutionary algorithm is as good as its fitness function, first of all the fitness function was modified with respect to the feature selection and a mechanism for features elimination from the tests was introduced.

To build the feature selection into the fitness function, the tree size term *S(T)* should reflect the complexity of tests. This can be obtained by expressing the tree size as the sum of the test complexities. For the hyper-plane *H(w,θ)* the test complexity *Comp(w)* can be defined as follows:

$$Comp(w) = (1-\beta) + \beta \frac{n(w)}{N} \qquad (2)$$

where $n(w)$ is the number of non-zero weights in the hyper-plane and β is a user supplied parameter designed for controlling the impact of the tests complexity on the tree size (default value 0.2). When $n(w)=N$ (N is the number of all features) the test complexity is equal 1.

In the real number representation of the hyper-plane, the zero-weight corresponding to an excluded feature is very unlikely. The probability of the feature drop is significantly increased by modifying the mutation-like operator which, while mutating the weights of the hyperplanes, has now a strong preference in assigning zero value to these weights. The remaining evolutionary operators were left unchanged.

The problem directly connected with the feature selection is "under-fitting" the training data [5], which often occurs near the leaves of the tree. The number of the feature vectors used to search for a split has to be significantly greater than the number of the features used in this split. In the presented system, the maximal number of features in the test is restricted based on the number of the available training objects.

3. Experimental Results

3.1. The Datasets

The experiments were performed on six datasets that are publicly available at Kent Ridge repository [13]. Their brief description is placed below.

3.1.1. Breast Cancer

The dataset is investigated in [17]. The training part contains 78 patient samples, 34 of whom are from patients with developed distance metastases within 5 years (relapse), the rest 44 samples are from patients with not developed disease after their initial diagnosis for interval of at least 5 years (non-relapse). The test set is also provided. It contains 12 relapse and 7 non-relapse samples. The number of genes is 24481. "Not a number" values were replaced by 100.

Pre-processing of this dataset consisted of only the t-test filtering because the provided data were scaled. With *p*-value equal to 1e-3 and 1e-4, respectively 85 and 10 features were drawn.

3.1.2. Central Nervous System

The dataset C mentioned in the paper [21] is used to analyse the outcome of the treatment of medulloblastomas. There are 7129 genes in the dataset. It is a two-class problem. To the first class (survivors) belong patients who are alive after treatment, while to the second class (failures) belong those who succumbed to their disease. The dataset contains 60 patient samples, 21 are survivors and 39 are failures.

The dataset was arbitrarily divided into the training (40 samples) and the test (20 samples) part. With *p*-value equal to 1e-2 and 1e-3, respectively 93 and 3 features were drawn.

3.1.3. Colon Tumour

The original dataset, obtained using Affymetrix technology, contains gene expression of around 6500 genes of tumorous and normal colon tissues. The selection of 2000 most relevant genes was made by Alon et al. [1], and this reduced version is investigated in our paper. It contains 62 samples collected from colon-cancer patients. Among them, 40 samples are from tumours (negative) and 22 normal (positive) are from healthy parts of the colons of the same patients.

The dataset was arbitrarily divided into the training (41 samples) and the test (21 samples) part. With p-value equal to 1e-2 and 1e-3, respectively 77 and 10 features were drawn.

3.1.4. Leukaemia

Leukaemia gene expression dataset is described in Golub et al. [10]. It comes form experiments on acute leukaemia and its classification task is to perform cancer subtype classification between acute lymphoblastic leukaemia (ALL) and acute myeloid leukaemia (AML). The gene expression levels were measured using Affymetrix high-density oligonucleotide arrays containing 6,817 human genes. The chip actually contains 7,129 different probe sets (some of them map to the same genes and some are added for the quality control purposes). The training dataset consists of 38 bone marrow samples (27 ALL and 11 AML). There are also 34 samples of testing data provided, with 20 ALL and 14 AML.

With p-value equal to 1e-3, 1e-4 and 1e-5, respectively 202, 75 and 25 features were drawn.

3.1.5. Lung Cancer

The dataset is investigated in [9]. It is a two-class classification problem of distinguishing between malignant pleural mesothelioma (MPM) and adenocarcinoma (ADCA) of the lung. There are 181 tissue samples (31 MPM and 150 ADCA). The training set contains 32 of them, 16 MPM and 16 ADCA. The remaining 149 samples are used for testing. Each sample is described by 12533 genes.

With p-value equal to 1e-3, 1e-4 and 1e-5, respectively 167, 61 and 27 features were drawn.

3.1.6. Prostate

The dataset is investigated in [24]. It is the tumour versus the normal classification problem. The training set contains 52 prostate tumour samples and 50 non-tumour (normal) prostate samples with around 12600 genes. An independent set of testing samples is also available, which is from a different experiment and has a nearly

10-fold difference in the overall microarray intensity from the training data. It contains 25 tumour and 9 normal samples.

With p-value equal to 1e-3, 1e-4 and 1e-5, respectively 116, 74 and 48 features were drawn.

3.2. Results and Discussion

The first step in our experiment was the pre-processing and filtering of the original data. The five points described in section 2.2.1 were carried out in turn. For the Breast Cancer dataset only the *t-test* filtering was performed. Different *p-values* were applied to obtain the range of datasets which differ in the number of the selected genes. The used *p-values* and obtained sizes of datasets are mentioned in previous subsection in the description of datasets.

The next step was the classification experiment on datasets prepared earlier. The results on the test data of this experiment are presented in Table 1. They comprise evaluations on both the axis parallel and the oblique decision trees, and it is intended

Table 1. The results of the classification experiment on datasets described in section 3.1 (for each dataset different values of p-value were used to obtain a range of datasets). The evolutionary approach was compared to the C4.5 and OC1 systems. GDT-AP represents the axis parallel and GDT-OB the oblique version of our algorithm. The size of the tree is evaluated as the number of leaves in the tree. The quality means the percentage of correctly classified instances

Classifier	GDT-AP		C4.5		GDT-OB		OC1		MV
Dataset	size	quality	size	quality	size	quality	size	quality	quality
AMLALL (202)	2	91.2	2	91.2	2	84.9	2	**91.2**	58.8
AMLALL (75)	2	82.4	2	82.4	2	81.8	2	**82.4**	58.8
AMLALL (25)	2	82.4	2	82.4	2	**84.3**	2	82.4	58.8
CentralNervous (93)	1	**65**	4	60	4	64.5	2	**65**	65
CentralNervous (3)	1	**65**	6	50	5.9	**60.7**	10	40	65
ColonTumor (77)	2	81	5	81	4.3	**77.9**	2	57.1	66.7
ColonTumor (10)	1	**66.7**	4	61.9	4	**69.5**	2	61.9	66.7
BreastCancer (85)	2	47.4	7	**57.9**	4.5	62.5	3	**84.2**	36.8
BreastCancer (10)	2	**84.2**	10	57.9	3.3	67.5	2	**73.7**	36.8
LungCancer (167)	2	74.4	2	**87.9**	2	70	2	**79.2**	10.1
LungCancer (61)	2	71.6	2	**87.9**	2	**63.7**	2	50.3	10.1
LungCancer (27)	2	64.3	2	**87.9**	2	**63.6**	2	50.3	10.1
Prostate (116)	2	**66.3**	7	50	2.7	**73.5**	4	67.7	73.5
Prostate (74)	2	**66.6**	6	61.8	2.3	**75.9**	4	67.7	73.5
Prostate (48)	2	**66.3**	6	61.8	2.4	**72.3**	4	67.7	73.5
average	*1.8*	***71.6***	*4.5*	*70.8*	*3*	*71.5*	*3*	*68*	*51*

to compare GDT-AP versus C4.5 and GDT-OB versus OC1. Because GDT is a stochastic algorithm, the average result of 30 runs is presented. In the last column of the table the performance of the majority voting (MV) is also placed.

One of the aims of this paper is to compare performance of the evolutionary versus top-down approaches to learn both the axis parallel and the oblique decision trees. The results presented in Table 1 are promising. Both GDT-AP and GDT-OB achieve a better score more often than their counterparts.

It is apparent that the more features remained in the datasets, the better results the investigated algorithms yielded. The classification on a bigger dataset (e.g. leukaemia with 202 features) gives better results than on the smallest variant, in which the most features were discarded. In smaller datasets, certain features which allowed achieving a better discrimination were eliminated. It gives evidence that instead of ranking single feature, the subset of features should be considered.

The results achieved by GDT are relatively promising, but they are not stable. For this reason the average score of 30 runs is shown. The explanation for this can be found in the size of the trees. Because trees have in most cases two leaves on average, it is evident that the biggest problem is not the structure of the tree but the tests in internal nodes. GDT, which at the same time looks for tests (GDT-OB even complex, multivariate tests) and the tree structure, was not able to find the best tests in every run. In our previous studies of global induction of decision trees [14, 15] it was concluded that they performed well (much better than top-down approaches) when there were compound relationships in the data and bigger trees were needed.

One of the most essential, user specified parameters of GDT algorithm is the importance of the tree size for the fitness function (i.e. the α parameter in Equation 1). It is chosen empirically. In the presented research the impact of the α parameter on the results was analysed in detail. Such experiments for GDT-AP and GDT-OB are presented in Figures 2 and 3, respectively. First of all, it is noticeable that it is possible to find the value of α, which leads to good results on the majority of datasets. In the final experiment, presented in Table 1, $\alpha = 0.1$ was used. When comparing this value, which stems from the presented figures, to the value of α that suits another types of the analysed both real and artificial datasets (e.g. in [14] and [15]), it is visible that in case of the gene expression data the GDT algorithm is more sensitive to the value of the α parameter. This sensitivity, however, concerns only the classification quality on the test data. The size of the tree is not affected more than in other types of analysed data. It means that the algorithm is able to find a decision tree of the small size, which separates the training dataset well (in many cases even in 100%). It is interesting to look at decision trees with bigger size (i.e. corresponding a smaller values of α). Figures 4 and 5 show two different decision trees found by GDT-AP inducer on the Prostate dataset with 116 features. The accession numbers (Affymetrix U95Av2 arrays) are used to represent features in these trees.

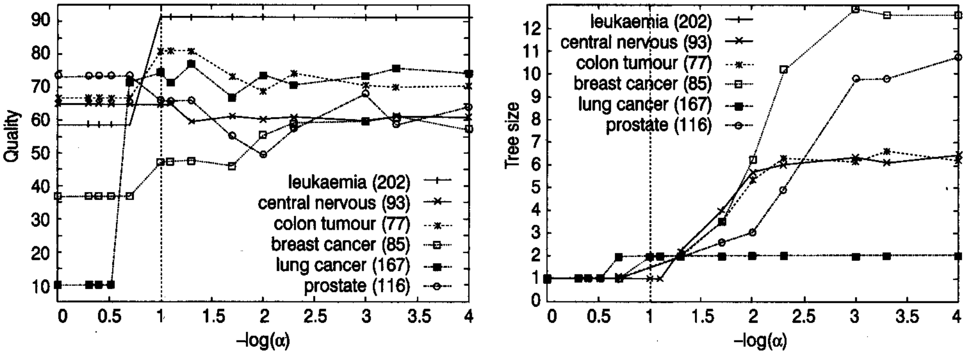

Fig. 2. The impact of the α value on the quality (on the left) and on the size (on the right) of the trees induced by GDT-AP. The vertical, straight line corresponds to the α value used in final evaluations ($x = -\log_{10}(0.1)$)

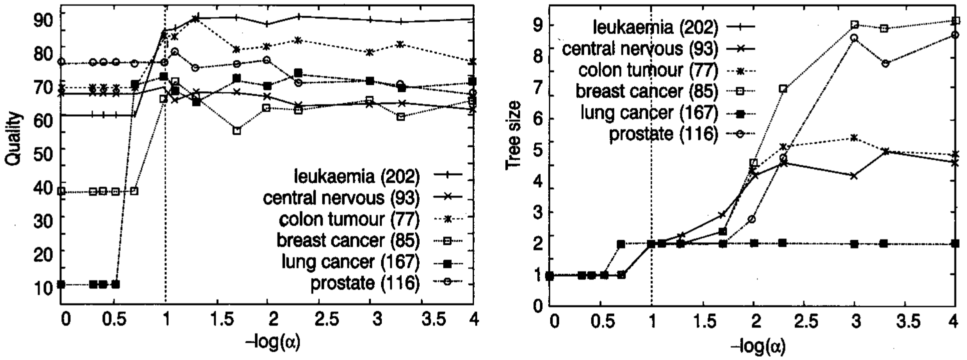

Fig. 3. The impact of the α value on the quality (on the left) and on the size (on the right) of the trees induced by GDT-OB. The vertical, straight line corresponds to the α vaue used in final evaluations ($x = -\log_{10}(0.1)$).

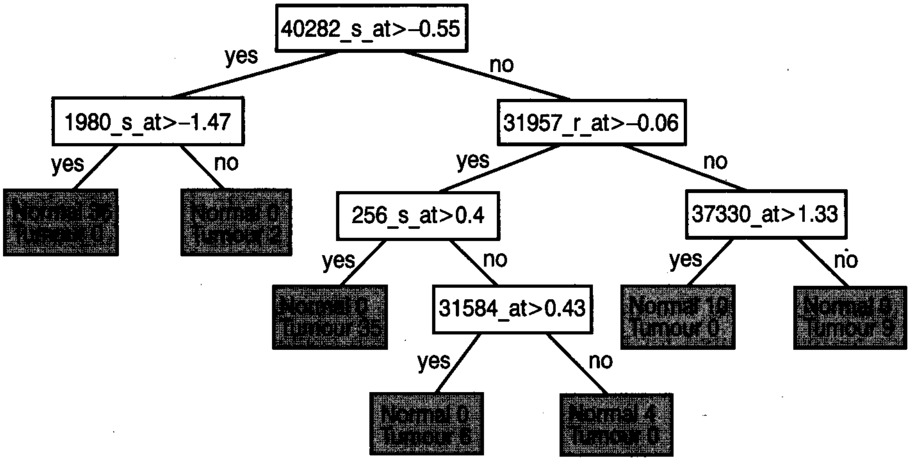

Fig. 4. The decision tree learned by GDT-AP. The value of α was 0.00001 and the classification quality on training data was 100% and 44.1% on test data

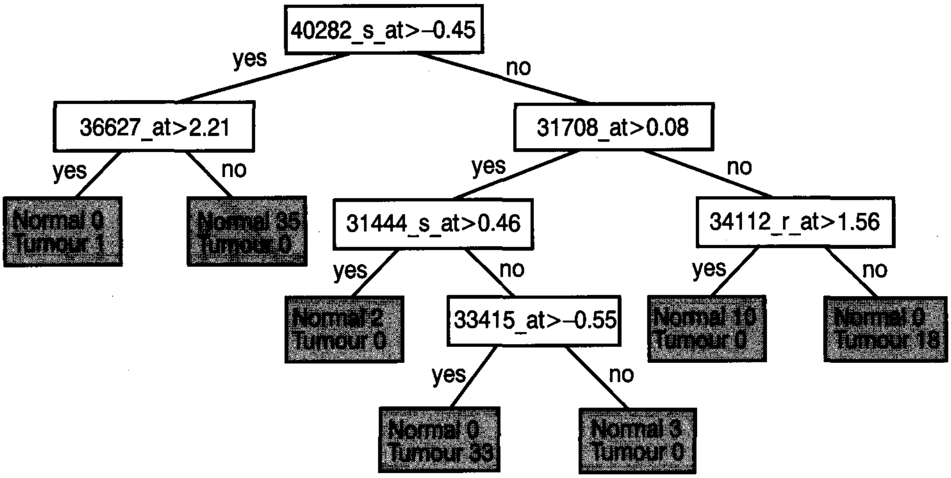

Fig. 5. The decision tree learned by GDT-AP. The value of α was 0.00001 and the classification quality on training data was 100% and 52.9% on test data

It is worth emphasizing that even though these trees do not have now a certain value for classification, they may be of a great importance for specialists in the field of bionformatics. These trees separate the training data perfectly (100% on the training dataset) and what is even more important they have only one gene in common. Such experiments can help in discovering compound relationships and signalling pathways between genes.

4. Conclusions

In this paper, the decision tree approach to gene expression data classification is presented. The two types of decision trees were employed in the analysis. They were induced using both commonly used top-down and also global, evolutionary approach. The empirical results on publicly available gene expression data are presented and discussed. The main advantage of the proposed method to the classification of gene expression data is that the achieved results are usually easy to understand by molecular biologists and can be treated as valuable information for further analysis by these specialists. The natural feature selection of univariate decision trees makes them perfectly suited for this kind of data, which contain a lot of redundant and noisy features.

Acknowledgment

This work was supported by the grants W/WI/1/05 and W/WI/5/05 from Białystok Technical University.

References

1. Alon U. et al.: Broad Patterns of Gene Expression Revealed by Clustering Analysis of Tumor and Normal Colon Tissues Probed by Oligonucleotide Arrays. Proceedings of National Academy of Sciences of the United States of America, 1999, 96, 6745–6750.
2. Baldi P., Brunak S.: Bioinformatics: the machine learning approach. 2nd ed. Cambridge, MA: MIT Press; 2001.
3. Ben-Dor A. et al.: Tissue classification with gene expression profiles. J. Comput. Biol., 2000, 7, 559–83.
4. Breiman, L. et al.: Classification and Regression Trees, Wadsworth Int. 1984.
5. Duda O., Heart P., Stork D.: Pattern Classification. 2nd ed. J. Wiley, 2001.
6. Dudoit S.J., Fridlyand J., Speed T.: Comparison of discrimination methods for the classification of tumors using gene expression data. J. Am. Stat. Assoc., 2002, 97, 77–87.
7. Gentleman R. et al. Bioconductor: Open software development for computational biology and bioinformatics. Genome Biology, 2004, 5, R80.
8. Gentleman R. Genefilter: filter genes. R package version 1.5.0.
9. Gordon G.J. et al.: Translation of Microarray Data into Clinically Relevant Cancer Diagnostic Tests Using Gene Expression Ratios in Lung Cancer And Mesothelioma. Cancer Research, 2002, 62, 4963–4967.
10. Golub T.R. et al.: Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science, 1999, 286, 531–537.
11. Guyon I., Weston J., Barnhill S., Vapnik V.: Gene selection for cancer classification using support vector machines. Machine Learning, 2002, 46, 389–422.
12. Jinyan L., Huiqing L.: Ensembles of cascading trees. In ICDM '03: Proceedings of the Third IEEE International Conference on Data Mining, 585, IEEE Computer Society, 2003.
13. Kent Ridge: Bio-medical Data Set Repository: http://sdmc.lit.org.sg/GEDatasets/Datasets.html
14. Krętowski M., Grześ M.: Global Induction of Oblique Decision Trees: An Evolutionary Approach. In: Proc. of IIPWM 05, Springer, 2005, 309–319.
15. Krętowski M., Grześ M.: Global Learning of Decision Trees by an Evolutionary Algorithm. In: Information Processing and Security Systems. Springer, 2005, 401–410.
16. Kuo W.P. et al.: A primer on gene expression and microarrays for machine learning researchers. Journal of Biomedical Informatics, 2004, 37, 4, 293–303.
17. Laura J. van 't Veer et al.: Gene Expression Profiling Predicts Clinical Outcome of Breast Cancer. Letters to Nature, Nature, 2002, 415, 530–536.
18. Lyons-Weiler J., Patel S., Bhattacharya S.: A classification-based machine learning approach for the analysis of genome-wide expression data. Genome Research, 2003, 13, 503–512.
19. Madeira. S. C., Oliveira A. L.: Biclustering Algorithms for Biological Data Analysis: A Survey. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2004, 1, 1, 24–45.
20. Murthy S., Kasif S., Salzberg S.: A system for induction of oblique decision trees. Journal of Artificial Intelligence Research 1994, 2, 1–33.
21. Pomeroy S. L. et al.: Prediction of Central Nervous System Embryonal Tumour Outcome Based on Gene Expression, Letters to Nature, Nature, January 2002, 415, 436–442.
22. Quinlan J.R.: C4.5: programs for machine learning. San Francisco: Morgan Kaufmann, 1993.
23. R Development Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org. 2005.
24. Singh. et al.: Gene Expression Correlates of Clinical Prostate Cancer Behavior. Cancer Cell, March, 2002, 1, 203–209.
25. Shipp M.A. et al.: Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. Nat. Med. 2002, 8, 68–74.
26. Tan A.C., Gilbert D.: Ensemble machine learning on gene expression data for cancer classification. Applied Bioinformatics, 2003, 2(3 Suppl), 75–83.

27. Thomas, J.G. et al.: An efficient and robust statistical modelling approach to discover differentially expressed genes using genomic expression profiles. Genome Res., 2001, 11, 1227-1236.
28. Valentini G., Muselli M., Ru F.: Bagged Ensembles of SVMs for Gene Expression Data Analysis. In IJCNN2003, Portland, USA, 2003.