Similarity-based decision tree induction method and its application to cancer recognition on tomographic images

V B Berikov¹,², I A Pestunov³, R M Kozinets² and S A Rylov³

¹Sobolev Institute of Mathematics, Acad. Koptyug Avenue, 4, Novosibirsk, Russia, 630090
²Novosibirsk State University, Pirogova str., 1, Novosibirsk, Russia, 630090
³Institute of Computational Technologies SB RAS, Lavrentiev Avenue, 6, Novosibirsk, Russia, 630090

e-mail: berikov@math.nsc.ru

Abstract. The paper proposes a pattern recognition method using a modification of the class of logical decision functions presented in the form of decision tree. Instead of standard statements corresponding to the tree nodes, in which a variable is tested for a certain set of its values, a more general type of statements is used regarding the similarity of the point in question to different subsets of the observations. At the same time, to determine the degree of similarity, various metrics and subspaces of features can be used. This type of decision tree allows one to obtain more complex decision boundaries, which at the same time have a clear logical interpretation for the user. Several tree induction strategies are considered based on data transformation using support points selected with Relief, SVM, and k-means procedures. The method is experimentally investigated on the problem of tomographic images analysis, as well as on several synthetic datasets. Experiments have shown that the proposed method gives more accurate predictions than CART, SVM, kNN classifiers and deep convolutional neural network (AlexNet).

1. Introduction
Classification methods based on logical decision functions presented in the form of decision trees (DT) [1,2] are popular in machine learning. Compared with other approaches, DT has the following advantages:

– give one an opportunity to analyze information of different types (i.e., for quantitative and qualitative characteristics describing objects), in the presence of missed feature values;
– find probabilistic logical rules that reflect cause-and-effect relationships of the phenomenon under study;
– automatically determine the most informative features for each classified object and use them for making a decision;
– in combination with an ensemble approach (e.g., decision forest, boosting on trees [3,4]), DT is able to find sufficiently stable solutions with high generalizing ability.

A recent review of existing methods for DT induction is given in [5]. Despite a large number of known approaches, there is still a need in developing efficient methods with high generalization ability. There are several possible ways to improve quality. The first approach is to find a criterion that will enhance the predictive ability of decisions by optimally combining the accuracy and complexity.
of the tree for the given data [6]. The second approach involves the development of more sophisticated techniques for representing the tree (for example, using linear decision boundaries in tree nodes) and applying “deeper” algorithms for searching the optimal tree structure [7].

Figure 1. An example of linearly separable classes (labeled by 0 and 1) and their partition in correspondence with a “classical” decision tree.

A “classical” DT is a tree-like graph, in the nodes of which conditions of two possible types are tested. If \( X \) is a numerical attribute, then the condition “\( X(a) < b \)” is examined, where \( a \) is an arbitrary object from the statistical population, \( X(a) \) is the value of \( X \) for object \( a \), \( b \) is some value of the attribute. If \( X \) is a categorical attribute, then the condition “\( X(a) = b \)” is checked. Depending on the truth or false of the test, the left or the right sub-node is chosen. The leaves (terminal nodes) of the tree are associated with the values (class labels) of the predicted feature. The paths from the root node to leaves represent classification rules. To find an optimal DT, a recursive partition of feature space is performed.

This approach has a significant drawback: the partitioning of feature space occurs strictly parallel to the feature axes (in the case of numerical features), even if the real boundary between classes has a linear shape (Figure 1). To approximate the boundary, it is necessary to use a more complex tree structure (with many additional nodes) that often has a negative influence on the efficiency of decisions.

Some works (see, e.g., [7]) propose oblique DT (ODT, also called multivariate DT) with more complicated types of statements having the form \( \sum \beta_j X_j(a) + \beta_0 < 0 \), where the summation is carried out over a subset of numerical features, \( \beta_0, \beta_1, \ldots \) are real-valued coefficients. The coefficients are estimated by optimizing a given quality functional for the subset of objects in the tree node. A number of algorithms for ODT induction exist:

- Classification and Regression Trees - Linear Combination (CART-LC) [1];
- Simulated Annealing Decision Tree (SADT) [8];
- Linear Machine Decision Trees (LMDT) [9],
- OC1 system [10],
- based on Support Vector Machine (SVM-ODT) [11], etc.

Despite the significant improvement of prediction accuracy, this approach also has a number of limitations. First of all, the found linear boundaries are not easy to interpret in contrast with simple rules of “classical” univariate DT. Another limitation is ODT is applicable only for multivariate data and cannot be used for data described with pairwise similarity matrices.

To overcome the latter difficulty, the work [12] suggests Similarity Forest method in which an ensemble of ODT is built. Each variant of ODT is defined by randomly chosen pair of data points from different classes; the splitting boundary is a hyperplane perpendicular to the segment connecting the pair and crossing its midpoint. In the experiments, the proposed algorithm has demonstrated sufficiently high accuracy in comparison with a number of other methods, especially in the presence of missed feature values, even if the input information has the form of multidimensional data. However,
the obtained ensemble decision is hard-to-interpret because it includes a large number of generated trees.

In this paper, we suggest a method aimed at eliminating the above-mentioned drawbacks. We propose to use a more general type of statements regarding the similarity of observations. The similarity can be calculated using various metrics in different feature subspaces. This type of decision tree allows one to get more complex decision boundaries, which at the same time have a clear logical interpretation for the user.

The developed algorithm was experimentally investigated on the problem of tomographic images recognition, where a lung tumor was to be classified to one of two types.

The paper has the following structure. The second section describes the proposed decision tree modification based on objects similarity (Similarity-Based Decision Tree, SBDT), as well as several strategies for its construction. In the third section, we illustrate the performance of the method on three synthetic datasets. The forth section describes an experimental study of the method on real data. A comparison with a number of other methods is also given. The final section formulates the main conclusions of the work.

2. Similarity-based decision tree (SBDT) in pattern recognition problems

We consider a pattern recognition problem formulated as follows. Let us denote by \( \Gamma \) a general population of objects under consideration, and by point \( x = x(a) = (x^{(1)}, \ldots, x^{(m)}) \in \mathbb{R}^m \) a feature description of object \( a \in \Gamma \), where \( m \) is feature space \( F \) dimensionality. Let \( Y \) be a set of class labels. We consider a binary classification problem: \( Y = \{-1, +1\} \), although the results can be extended to a multi-class scenario. Denote by \( X \) the set of feature descriptions of objects from \( \Gamma \). Let \( y^* : X \rightarrow Y \) be an objective function with values assigned to the points of the finite set (training or learning sample) \( X_{\text{train}} \subseteq X \). We need to build a decision function \( f : X \rightarrow Y \) which belongs to a given family; \( f \) should approximate \( y^* \) and minimize the estimate of misclassification probability for any point \( x \in X \). Let \( X_{\text{test}} \) be another subset of \( X \) used for evaluating the performance of the decision function, \( X_{\text{test}} \cap X_{\text{train}} = \emptyset \). Denote by \( X = X_{\text{train}} \cup X_{\text{test}} \), and let \( d \) be the size of \( X \) and \( l \) be the size of \( X_{\text{train}} \).

We propose a modification of DT in which instead of standard tests, more general statements of the type “object \( a \) is more similar to the set \( A \) than to the set \( B \) in feature subspace \( F' \)” according to metrics \( \mu \) are examined in the internal nodes. Here \( A, B \) are subsets of learning sample, typically of small cardinality. In this work, we assume that each set \( A, B \) includes exactly one object (its description is called a support point). We also shall assume that \( F' = F \) and metrics \( \mu \) is the Euclidian metrics.

Suppose \( T \) is a binary tree with \( t \) internal nodes, and \( A = \{A^1, \ldots, A^p\}, B = \{B^1, \ldots, B^n\} \) are the sets of support points from positive and negative classes respectively. For each internal node \( \nu_i \) of the tree, \( i = 1, \ldots, t \), we define the tested statement as follows: "\( x \in M^i_1 \)" where \( M^i_1 \) are the points from feature space, which are closer to \( A^i \) than to \( B^i \) (figure 2). Thus, the data is separated linearly (figure 3).

For any \( x \in X \) we define matrix \( M_x \) with elements:

\[
m(i, j) = \begin{cases} 1, & \text{if } \mu(x, A^i) - \mu(x, B^j) < 0, \text{ otherwise} \end{cases}, i=1,\ldots, p, j=1,\ldots, n,
\]

where \( \mu \) is a metric in feature space \( F \).

Let us transform matrix \( M_x \) into a vector \( \vec{M}_x \) of the size \( pn \) by the reshaping procedure. Then each point \( x \) is described by vector \( \vec{M}_x \) of the size \( pn \). In this way, \( X' = \{ \cup_{x \in X} \vec{M}_x \} \) is a new feature representation of \( X \).

Consider an example of feature transformation based on data shown in figure 2. As the number of all possible pairs equals one, matrix \( M_x \) has only one element. For objects represented by circles we have \( M_x = 1 \), and for objects represented by triangles \( M_x = 0 \).
2.1. Support points selection

The proposed form of DT splits data points based on their relative positions. The support points selection method should consider their informativity for a given sample $X$. In this work, three methods of support points selection were implemented.

The first approach uses Relief feature selection algorithm introduced in [13]. Let $X_+$ and $X_-$ be subsamples of $X_{train}$ of +1 and −1 class. After generating vectors $\overrightarrow{M}_x$ for all $x \in X$, applying Relief allows extracting the most informative sets $A \subset X_+$, $B \subset X_-$ reducing feature dimension.

The second way is based on Support Vector Machine (SVM) [14]. SVM builds a separating hyperplane with maximum distance (margin width) between points of different classes. Data points which are placed on the border of the margin are called support vectors. When SVM is trained on $X_{train}$, this set is divided into support and non-support vectors. In our approach, support vectors compose the sets of support points $A$ and $B$.

The last method is based on $k$-means clustering algorithm. We generate $S$ subsamples of the size $L$ from $X$ and apply $k$-means ($k=2$) to each subsample to extract cluster’s medoids which are considered as support points.

In addition, we use kernel $k$-means algorithm that employs kernel function instead of a scalar product. Kernel function implicitly transforms initial feature space into another space of larger dimensionality, where the configuration of data points is changed, often resulting in linearly separable form.

2.2. Construction of similarity-based decision tree

We choose CART algorithm [2] as an add-on method used in SBDT construction. The proposed SBDT algorithm can be represented with the following steps:

Step 1. Find sets of support points $A$ and $B$ of classes +1, −1.

Step 2. Compute vector $\overrightarrow{M}_x$ for all $x \in X$.

Step 3. Build a decision tree in new feature space $\{\bigcup_{x \in X} \overrightarrow{M}_x\}$ with CART.

When using SVM or $k$-means based selection methods, we get linear computational complexity, while using Relief or kernel $k$-means results in quadratic complexity depending on the sample size.

3. A study on synthetic data

We study the efficiency of the proposed method on three synthetic datasets. Each two-dimensional dataset includes 100 points belonging to two classes. Figure 4 shows the visualizations of the datasets: Moons (two interleaving half-circles), Circles (a larger circle containing the smaller one), Linear (a linearly separable dataset). Each of the datasets has added noise.

We used Python’s default parameters for CART; for SBDT, we applied $k$-means-based selection of the support points. Figure 5 presents the visualized results of classification with CART; figure 6 shows the results obtained with SBDT. Each image also specifies leave-one-out error estimate.
As we can see from these examples, SBDT method has better accuracy than CART, and the latter produced more “rough” decision boundaries.

![Figure 4](image)

**Figure 4.** Datasets: (a) – Moons; (b) – Circles; (c) – Linear.

![Figure 5](image)

**Figure 5.** Results of classification with CART: (a) – Moons; (b) – Circles; (c) – Linear.

![Figure 6](image)

**Figure 6.** Results of classification with SBDT: (a) – Moons; (b) – Circles; (c) – Linear.

4. Research on lung tumor classification data
The efficiency of the proposed method was investigated on the applied problem of lung tumor classification with computed tomography scans. Source data were taken from the open database www.cancerimagingarchive.net.

4.1. Data
The data represent a set of 370 medical computed tomography scans in DICOM format. The images contain lateral lung slices with a non-small lung carcinoma (NSLC) in I – III stages of two common types: 1) Adenocarcinoma (170 2D images), and 2) Squamous Cell Carcinoma (200 2D images).

Each patient has a corresponding stack of 2D images. Only images with clearly distinguishable closed-from tumors inside the lungs were taken into consideration. An example of this kind of image is shown in figure 7.
4.2. Area of interest segmentation

To extract the area of interest (lung tumor), a two-stage segmentation procedure was applied based on the region growing algorithm. At the first stage, the lung area was retrieved, and at the second stage, the tumor area was extracted.

The lung segmentation algorithm can be described as a sequence of three steps.

Step 1. Brightness histogram analysis and the search for seed points.

Step 2. The use of the region growing method.

Step 3. Application of morphological operations to the selected region.

Image histogram analysis showed that lung tissue regions have pixel brightness in the range of [0.01, 0.5]. At the same time, organs tissues and internal body cavity have pixel brightness in the range of [0.6, 0.9]. To obtain the initial seed points, Otsu’s binarization method [15] was used. After that, the binary image was divided into connected components, where the largest areas were considered to be the lungs regions (figure 8). Then the seed points for the region growing algorithm were selected from these components.

![Figure 8](image)

**Figure 8.** (a) – original image; (b) – binarization; (c) – finding the largest components.

For tumor area extraction, the starting point is selected manually. Figure 9 shows the stages of tumor extraction.

![Figure 9](image)

**Figure 9.** Steps of tumor extraction: (a) – original image; (b) – binarization; (c) – lung extraction; (d) – tumor extraction.
The set of features extracted from the segmented tumor area can be divided into four groups: geometric, morphological, texture, and histogram. All features we use are described in [16]. Algorithms for feature extraction were implemented in MATLAB environment. These features are used as an input to classification algorithms.

4.3. Experimental results
The initial dataset is divided into training and test sets preserving class proportions. There are no images of one patient in different sets. The training sample includes 260 images, and the test sample includes 110 images.

4.3.1. Accuracy comparison
The proposed SBDT method is compared with SVM, decision tree (CART), and k-nearest neighbors (kNN) classification algorithms. A polynomial kernel is used for kernel $k$-means and SVM. Classification accuracies evaluated on the test sample are presented in table 1.

4.3.2. Application of convolutional neural network
We have studied the possibility of using deep learning methods for the problem under consideration. Nowadays, the convolutional neural network [17] is one of the most popular methods in image recognition. The pre-trained AlexNet network is used as the initial. The parameters of the network are fine-tuned and additional training on the training set with augmentation (rotations) of tumor images is performed. The highest obtained accuracy has reached 81%.

| Algorithm                        | Accuracy, % |
|----------------------------------|-------------|
| SBDT + $k$-means                 | 90          |
| SBDT + Relief                    | 88          |
| CART decision tree               | 84          |
| SBDT + SVM (linear)              | 83          |
| AlexNet                          | 81.5        |
| SVM (linear kernel)              | 79          |
| kNN (k=5)                        | 72.5        |

| Table 2. AlexNet architecture.  |
|----------------------------------|
| [227x227x3] INPUT [55x55x96]    |
| CONV1: 96 11x11 filters at stride 4, pad 0 [27x27x96] |
| MAX POOL1: 3x3 filters at stride 2 [27x27x96] |
| NORM1: Normalization layer [27x27x256] |
| CONV2: 256 5x5 filters at stride 1, pad 2 [13x13x256] |
| MAX POOL2: 3x3 filters at stride 2 [13x13x256] |
| NORM2: Normalization layer [13x13x384] |
| CONV3: 384 3x3 filters at stride 1, pad 1 [13x13x384] |
| CONV4: 384 3x3 filters at stride 1, pad 1 [13x13x256] |
| CONV5: 256 3x3 filters at stride 1, pad 1 [6x6x256] |
| MAX POOL3: 3x3 filters at stride 2 [4096] |
| FC6: 4096 neurons [4096]         |
| FC7: 4096 neurons [1000]         |
| FC8: 1000 neurons (class scores) |
Based on the experiment results, one can conclude that in the condition of a small amount of training data, the usage of a convolutional network is impractical.

5. Conclusion
A new method for constructing a decision tree based on the similarity of objects was proposed in the paper. A peculiarity of this method is the use of support points, which can be obtained by several strategies based on Relief, SVM or k-means algorithms. The proposed method allows obtaining complex decision boundaries, which at the same time have clear logical interpretation.

The method was experimentally studied on the problem of cancer tumor classification. The proposed method showed higher classification quality compared to the support vector machine, decision tree (CART algorithm) and k-nearest neighbors method (kNN). On three synthetic datasets, SBDT method has shown better accuracy than CART.

Our future research plans include the extension of the proposed algorithm to other types of data (e.g., hyperspectral imagery, genetic sequences), in regression analysis problems, and studies on the usage of more complex forms of statements in decision tree nodes. Another interesting application is semi-supervised classification in combination with cluster ensembles [18]. A detailed comparison with other state-of-the-art methods is also planned.

6. References
[1] Lbov G S 1982 Logical Function in the Problems of Empirical Prediction Handbook of statistics 2 479-492
[2] Breiman L, Friedman J H, Olshen R A and Stone C J 1984 Classification and Regression Trees (New York: Routledge)
[3] Breiman L 2001 Random forests Machine learning 45(1) 5-32
[4] Schapire R E 2003 The boosting approach to machine learning: An overview Nonlinear estimation and classification 149-171
[5] Kotsiantis S B 2013 Decision trees: a recent overview Artificial Intelligence Review 39(4) 261-283
[6] Berikov V B and Lbov G S 2007 Choice of optimal complexity of the class of logical decision functions in pattern recognition problems Doklady Mathematics 76(3) 969-971
[7] Lbov G S and Berikov V B 1993 Recursive Method of Formation of the Recognition Decision Rule in the Class of Logical Functions Pattern Recognition and Image Analysis 3(4) 428-431
[8] Bucy R S and Diesposti R S 1993 Decision tree design by simulated annealing ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique 27(5) 515-534
[9] Utgoff P E and Brodley C E 1990 An incremental method for finding multivariate splits for decision trees Proc. of the Seventh Int. Conf. on Machine Learning 58-65
[10] Murthy S K, Kasif S and Salzberg S 1994 A system for induction of oblique decision trees J. of artificial intelligence research 2(1) 1-32
[11] Menkovski V, Christou I T and Efremidis S 2008 Oblique decision trees using embedded support vector machines in classifier ensembles 7th IEEE International Conference on Cybernetic Intelligent Systems 1-6
[12] Sathe S and Aggarwal C C 2017 Similarity forests Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 395-403
[13] Kira K 1992 A Practical Approach to Feature Selection Proceedings of the ninth international workshop on Machine learning 249-256
[14] Cortes C and Vapnik V 1995 Support-vector networks Machine learning 20(3) 273-297
[15] Otsu N 1979 A threshold selection method from gray-level histograms IEEE transactions on systems, man, and cybernetics 9(1) 62-66
[16] Basu S 2012 Developing Predictive Model for Lung Tumor Analysis Graduate Theses and Dissertations URL: https://scholarcommons.usf.edu/etd/3963
[17] LeCun Y 1989 Backpropagation applied to handwritten zip code recognition Neural computation 1(4) 541-551
[18] Tatarnikov V, Pestunov I and Berikov V 2017 Centroid averaging algorithm for a clustering ensemble Computer Optics 41(5) 712-718

Acknowledgments
The research presented in Section 4 was supported by the Russian Foundation for Basic Research, project 19-29-01175. The study presented in Sections 2,3 was supported by the Russian Foundation for Basic Research, project 18-07-00600, by the Russian Academy of Science (the Program of basic research), project 0314-2019-0015, and by the Russian Ministry of Science and Education under the 5-100 Excellence Programme.