An Evolutionary Algorithm for Oblique Decision Tree Induction

Marek Krętowski

Faculty of Computer Science, Białystok Technical University
Wiejska 45a, 15-351 Białystok, Poland
e-mail: mkret@ii.pb.bialystok.pl

Abstract. In the paper, a new evolutionary approach to induction of oblique decision trees is described. In each non-terminal node, the specialized evolutionary algorithm is applied to search for a splitting hyperplane. The feature selection is embedded into the algorithm, which allows to eliminate redundant and noisy features at each node. The experimental evaluation of the proposed approach is presented on both synthetic and real datasets.

1 Introduction

Decision trees (DT) have been extensively investigated in statistics, machine learning and pattern recognition (see [11] for a very good multi-disciplinary survey) and now they are one of the most popular classification tools. Clones of the most renowned induction algorithms e.g. CART [4] or C4.5 [12] are included in virtually every data mining system. Advantages of the DT-based approach include natural representation, fast induction (especially in case of univariate splits) and ease of interpretation of obtained predictions.

The simplest variant of a decision tree is so called univariate tree. In each non-terminal node, it exploits a test, which is based on a single attribute. Such a split is equivalent to partitioning the set of objects with an axis-parallel hyperplane. The use of univariate tests may lead to very complicated classifier if decision borders are not axis-parallel. Oblique decision trees allow to avoid the aforementioned problem by using more flexible test based on a linear combination of attributes. It should be noted however that finding the optimal oblique split is generally much more difficult.

Several algorithms for oblique trees induction have been introduced so far. One of the first trials is done in CART [4]. The system is able to search for linear combinations of the continuous-valued attributes and also to simplify them by feature elimination procedure. Generally CART prefers univariate tests and chooses oblique one very rare. Murthy et al. [10] introduce OC1 (Oblique Classifier 1), the algorithm that combines deterministic (hill-climbing) and randomized procedures to search for a good tree. The method was applied to classify a set of patients with breast cancer and showed excellent accuracy. Another interesting approach was proposed by Gama et al. [8]. Their Linear tree system combines an...
univariate tree with a linear discrimination by means of constructive induction. At each node a new instance space is defined by insertion of new attributes that are projections over the hyper-planes given by a linear discrimination function and new attributes are propagated downward. A system proposed by Bobrowski et al. [2] is based on the dipolar criterion functions and exploits the basis exchange algorithm as an optimization procedure. The system can be treated as a predecessor of the approach proposed in the paper.

Evolutionary algorithms (EA) [9] are stochastic search techniques, which have been successfully applied to many optimization problems. The success of EAs is attributed to their ability to avoid local optima, which is their main advantage over greedy search methods. One of the first applications of evolutionary approach to induction of oblique tree is presented in Binary Tree-Genetic Algorithm (BTGA) system [6]. In this approach, a linear decision function at each non-terminal node is searched by standard genetic algorithm with binary representation. The maximum impurity reduction is adopted as the optimality criterion. Recently, Cantu-Paz et al. [5] present two extensions of the OC1 algorithm by using two standard algorithms: (1+1) evolution strategy and simple genetic algorithm. The empirical results show that their system is able to find competitive classifiers quickly and that EA-based systems scale better than traditional methods to size of the training dataset.

In the paper, a new specialized evolutionary algorithm for searching the hyper-plane in non-terminal modes of the oblique decision tree is proposed. The most important innovations concern the fitness function and genetic operators.

## 2 Oblique Tree Induction

Let’s assume that a learning set is composed of $M$ objects belonging to one of $K$ classes. Each object is described by a feature vector $\mathbf{x}^j = [x^j_1, ..., x^j_N]^T$ ($j = 1, ..., M$) ($\mathbf{x}^j \in \mathbb{R}^N$). The feature space could be divided into two regions by the hyper-plane $H(\mathbf{w}, \theta) = \{\mathbf{x} : \langle \mathbf{w}, \mathbf{x} \rangle = \theta\}$, where $\mathbf{w} = [w_1, ..., w_N]$ ($\mathbf{w} \in \mathbb{R}^N$) is the weight vector, $\theta$ is the threshold and $\langle \mathbf{w}, \mathbf{x} \rangle$ is the inner product.

A dipole [3] is a pair $(\mathbf{x}^i, \mathbf{x}^j)$ of the feature vectors. The dipole is called mixed if and only if the objects constituting it belong to two different classes and a pair of the vectors from the same class constitutes the pure dipole. Hyper-plane $H(\mathbf{w}, \theta)$ splits the dipole $(\mathbf{x}^i, \mathbf{x}^j)$ if and only if:

$$\langle \mathbf{w}, \mathbf{x}^i \rangle - \theta \cdot \langle \mathbf{w}, \mathbf{x}^j \rangle - \theta < 0$$

It means that the input vectors $\mathbf{x}^i$ and $\mathbf{x}^j$ are situated on the opposite sides of the dividing hyper-plane.

Like most of the existing tree induction algorithm the presented system proceeds in a greedy, top-down fashion. At each non-terminal node, starting from the root, the best split is learned by using the evolutionary algorithm described below. The main components (e.g. fitness function, specific genetic operator) of the algorithm are based on the concept of dipoles. The learned hyper-plane divides the training subset into two subsets generating child nodes. The process is