Global Induction of Decision Trees

Marek Kretowski
Bialystok Technical University, Poland

Marek Grzes
Bialystok Technical University, Poland

INTRODUCTION

Decision trees are, besides decision rules, one of the most popular forms of knowledge representation in Knowledge Discovery in Databases process (Fayyad, Piatetsky-Shapiro, Smyth & Uthurusamy, 1996) and implementations of the classical decision tree induction algorithms are included in the majority of data mining systems. A hierarchical structure of a tree-based classifier, where appropriate tests from consecutive nodes are subsequently applied, closely resembles a human way of decision making. This makes decision trees natural and easy to understand even for an inexperienced analyst. The popularity of the decision tree approach can also be explained by their ease of application, fast classification and what may be the most important, their effectiveness.

Two main types of decision trees can be distinguished by the type of tests in non-terminal nodes: *univariate* and *multivariate* decision trees. In the first group, a single attribute is used in each test. For a continuous-valued feature usually an inequality test with binary outcomes is applied and for a nominal attribute mutually exclusive groups of attribute values are associated with outcomes. As a good representative of univariate inducers, the well-known C4.5 system developed by Quinlan (1993) should be mentioned.

In univariate trees a split is equivalent to partitioning the feature space with an axis-parallel hyper-plane. If decision boundaries of a particular dataset are not axis-parallel, using such tests may lead to an over-complicated classifier. This situation is known as the "staircase effect". The problem can be mitigated by applying more sophisticated multivariate tests, where more than one feature can be taken into account. The most common form of such tests is an oblique split, which is based on a linear combination of features (hyper-plane). The decision tree which applies only oblique tests is often called *oblique* or *linear,* whereas heterogeneous trees with univariate, linear and other multivariate (e.g., instance-based) tests can be called *mixed decision trees* (Llora & Wilson, 2004). It should be emphasized that computational complexity of the multivariate induction is generally significantly higher than the univariate induction. CART (Breiman, Friedman, Olshen & Stone, 1984) and OC1 (Murthy, Kasif & Salzberg, 1994) are well known examples of multivariate systems.

BACKGROUND

The issue of finding an optimal decision tree for a given classification problem is known to be a difficult optimization task. Naumov (1991) proved that optimal decision tree construction from data is NP-complete under a variety of measures. In this situation it is obvious that a computationally tractable induction algorithm has to be heuristically enhanced. The most popular strategy is based on the top-down approach (Rokach & Maimon, 2005), where a locally optimal search for tests (based, e.g., on a Gini, towing or entropy rule) and data splitting are recursively applied to consecutive subsets of the training data until the stopping condition is met. Usually, the growing phase is followed by post-pruning (Esposito, Malerba & Semeraro, 1997) aimed at increasing generalization power of the obtained classifier and mitigating the risk of the over-fitting to the learning data.

There are problems where the greedy search fails (e.g., the classical chess board problem) and more sophisticated methods are necessary. In this chapter, we present a global approach, where the whole tree (i.e., its structure and all splits) is constructed at the same time. The motivation for this is the fact that top-down induction with, e.g., entropy minimization, makes locally optimal decisions and at least more compact tree can be obtained when it is constructed and assessed in a global way.

As a first step toward global induction, limited look-ahead algorithms were proposed (e.g., Alopex Perceptron Decision Tree of Shah & Sastry (1999) evaluates quality of a split based on the degree of linear separability of sub-nodes). Another approach consists in a two-stage induction, where a greedy algorithm is applied in the first stage and then the tree is refined to be as close to optimal as possible (GTO (Bennett, 1994) is an example of a linear programming based method for optimizing trees with fixed structures).

In the field of evolutionary computations, the global approach to decision tree induction was initially investigated in genetic programming (GP) community. The tree-based representation of solutions in a population makes this approach especially well-suited and easy for adaptation to decision tree generation. The first attempt was made by Koza (1991), where he presented GP-method for evolving LISP S-expressions corresponding to decision trees. Next, univariate trees were evolved by Nikolaev and Slavov (1998) and Tanigawa and Zhao (2000), whereas Bot and Langdon (2000) proposed a method for induction of classification trees with limited oblique splits.

Among genetic approaches for univariate decision tree induction two systems are particularly interesting here: GATree proposed by Papagelis and Kalles (2001) and GAIT developed by Fu, Golden, Lele, Raghavan and Wasil (2003). Another related global system is named GALE (Llora & Garrell, 2001). It is a fine-grained parallel evolutionary algorithm for evolving both axis-parallel and oblique decision trees.

MAIN FOCUS

We now discuss how evolutionary computation can be applied to induction of decision trees. General concept of a standard evolutionary algorithm is first presented and then we will discuss how it can be applied to build a decision tree classifier.

Evolutionary Algorithms

Evolutionary algorithms (Michalewicz, 1996) belong to a family of metaheuristic methods which represent techniques for solving a general class of difficult computational problems. They provide a general framework (see Figure 1) which is inspired by biological mechanisms of evolution. A biological terminology is used here. The algorithm operates on individuals which compose a current population. Individuals are assessed using a measure named the fitness function and those with higher fitness have usually bigger probability of being selected for reproduction. Genetic operators such as mutation and crossover influence new generations of individuals. This guided random search (offspring usually inherits some traits from its ancestors) is stopped when some convergence criteria is satisfied.

Figure 1. A general framework of evlutionary algorithms

(1) Initialize the population
(2) Evaluate initial population
(3) Repeat
    (3.1) Perform competitive selection
    (3.2) Apply genetic operators to generate new solutions
    (3.3) Evaluate solutions in the population
  Until some convergence criteria is satisfied

A user defined adaptation of such a general evolutionary algorithm can in itself have heuristic bias (Aguilar-Ruiz, Giráldez, & Riquelme, 2007). It can prune the search space of the particular evolutionary application. The next section shows how this framework was adapted to the problem of decision tree induction.

Decision Tree Induction with Evolutionary Algorithms

In this section the synergy of evolutionary algorithms which are designed to solve difficult computational problems and decision trees which have an NP-complete solution space is introduced. To apply a general algorithm presented in Figure 1 to decision tree induction, the following factors need to be considered:

- Representation of individuals,
- Genetic operators: mutation and crossover,
- Fitness function.

Representation

An evolutionary algorithm operates on individuals. In the evolutionary approach to decision tree induction which is presented in this chapter, individuals are represented as actual trees, which can have different structure and different content. Each individual encoded