Decision Trees

Abstract

Decision trees find use in a wide range of application domains. They are used in many different disciplines including diagnosis, cognitive science, artificial intelligence, game theory, engineering and data mining. Decision Trees model has two goals: producing an accurate classifier and understanding the predictive structure of the problem. The classification accuracy of decision trees has been a subject of numerous studies. In this paper I presented the results of some recent research which showed that decision tree algorithms are very useful in any area.

Cristina Petri

Cluj Napoca, 2010

Table of Contents

1. Introduction

Decision trees are a class of data mining techniques that have roots in traditional statistical disciplines such as linear regression. Decision trees also share roots in the same field of cognitive science that produced neural networks. The earliest decision trees were modeled after biological processes (Belson 1956); others tried to mimic human methods of pattern detection and concept formation (Hunt, Marin and Stone 1966). [1]

Decision trees are a simple, but powerful form of multiple variable analysis. They provide unique capabilities to supplement, complement and substitute for:

- Traditional statistical form of analysis (such as multiple linear regression);
- A variety of data mining tools and techniques (such as neural networks);
- Recently developed multidimensional forms of reporting and analysis found in the field of business intelligence. [1]

A decision tree is a classifier expressed as a recursive partition of the instance space. The decision tree consists of nodes that form a *rooted tree*, meaning it is a *directed tree* with a node called “root” that has no incoming edges. All other nodes have exactly one incoming edge. A node with outgoing edges is called an *internal* or test node. All other nodes are called leaves (also known as terminal or decision nodes). In a decision tree, each internal node splits the instance space into two or more subspaces according to a certain discrete function of the input attributes values. In the simplest and most frequent case, each test considers a single attribute, such that the instance space is partitioned according to the attribute’s value. In the case of numeric attributes, the condition refers to a range.

Each leaf is assigned to one class representing the most appropriate target value. Alternatively, the leaf may hold a probability vector indicating the probability of the target attribute having a certain value. Instances are classified by navigating them from the root of the tree down to a leaf, according to the outcome of the tests along the path. [3]

Naturally, decision makers prefer less complex decision trees, since they may be considered more comprehensible. Furthermore, according to Breiman et al. (1984) the tree complexity has a crucial effect on its accuracy. The tree complexity is explicitly controlled by the stopping criteria used and the pruning method employed. Usually, the tree complexity is measured by one of the following metrics: the total number of nodes, total number of leaves, tree depth and number of attributes used. Decision tree induction is closely related to rule induction. Each path from the root of a decision tree to one of its leaves can be transformed into a rule simply by conjoining the tests along the path to form the antecedent part, and taking the leaf's class prediction as the class value. The resulting rule set can be simplified to improve its comprehensibility to a human user, and possibly its accuracy (Quilnan, 1987). [3]

A sample decision tree is illustrated in next figure which shows that a decision tree can reflect both a continuous and categorical object of analysis.

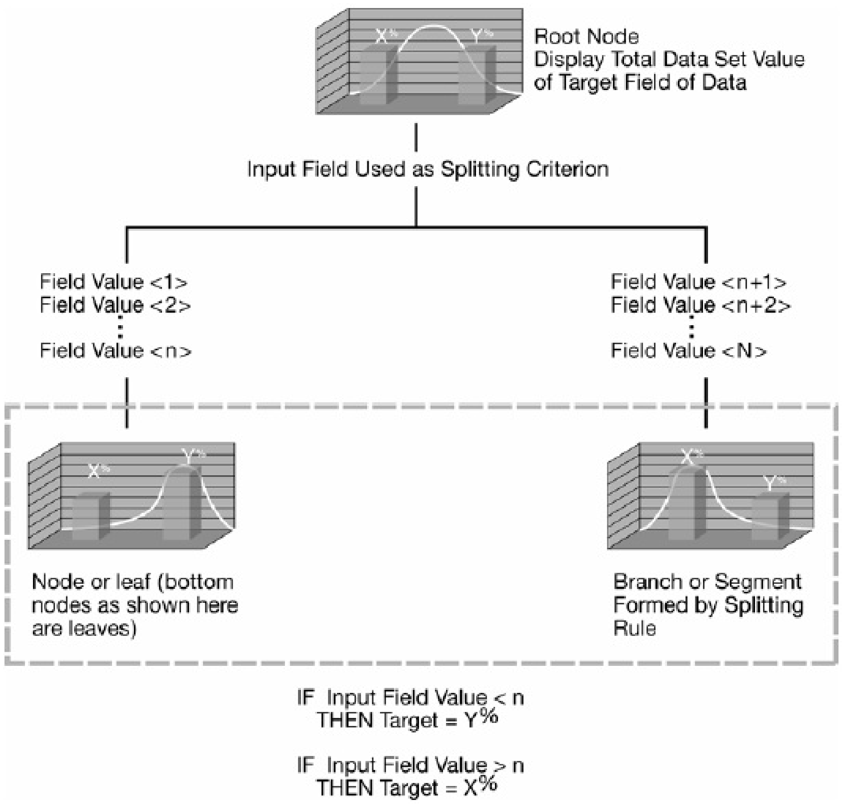

Figure 1: Illustration of the Decision Tree

2. Features

As decision trees evolved, they turned out to have many useful features, both in the traditional fields of science and engineering and in a range of applied areas, including business intelligence and data mining. These useful features include:

- Decision trees produce results that communicate very well in symbolic and visual terms. Decision trees are easy to produce, easy to understand and easy to use. One useful feature is the ability to incorporate multiple predictors in a simple step-by-step fashion. The ability to incrementally built highly complex rule sets is both simple and powerful.

- Decision trees readily incorporate various levels of measurement, including qualitative (e.g., good-bad) and quantitative measures.
- Decision trees readily adapt to various twists and turns in data – unbalanced effects, nested effects, offsetting effects, interactions and nonlinearities - that frequently defeat other one-way and multi-way statistical and numeric approaches.
- Decision trees are nonparametric and highly robust and produce similar effects regardless of the level of measurement of the fields that are used to construct decision tree branches.

To this day, decision trees continue to share inputs and influences from both statistical and cognitive science disciplines. Decision trees evolved to support the application of knowledge in a wide variety of applied areas such as marketing, sales, and quality control. [1]

3. Recent Research Results

Lately, Decision Tree model has been applied in very diverse areas like security and medicine. Decision trees can be used for problems that are focused on either insight or prediction. Even on data sets with very many columns, decision trees tend to converge very quickly on a decent model.

„Integrating genetic algorithm and decision tree learning for assistance in predicting in vitro fertilization outcomes" is an important paper publised in september 2010. Accurate and early prediction of the outcome of an in vitro fertilization (IVF) treatment is important for both patients and physicians. It is difficult for the clinician to recognize trends and intuitively decide how to optimize success rates for each infertile couple. This paper presents a hybrid intelligence method which integrating genetic algorithm and decision learning techniques for knowledge mining of an IVF medical database. The proposed method can not only assist the IVF physician in predicting the IVF outcome, but also find useful knowledge that can help the IVF physician tailor the IVF treatment to the individual patient with the aim of improving the pregnancy success rate. The twenty-eight most significant attributes for determining the pregnancy rate (e.g., patient's age, number of embryo transferred, number of frozen embryos, and culture days of embryo) and their combinative relationships (represented by if–then rules) were identified through the proposed method. The knowledge discovered in this study is currently accepted as an interesting discovery from the viewpoint of domain experts. For the results from this study to be conveniently accessed by IVF physicians and patients, an expert system tool equipped with the proposed IVF outcome prediction model was built. [5]

Another important article in medicine using Decision Tree is **„Colon cancer prediction with genetics profiles using evolutionary techniques"**, publised in september 2010. Microarray data provides information on gene expression levels of thousands of genes in a cell in a single experiment. DNA microarray is a poweeful tool in the diagnosis of cancer. Numerous efforts have been made to use gene expression profiles to improve precision of tumor classification. In this study comparison between class prediction accuracy of two different classifiers, Genetic Programming and Genetically Evolved Decision Trees, was carried out using the best 10 and best 20 genes ranked by the t-statistic and mutual information. Genetic Programming proved out to be the better classifier for this dataset based on area under the receiver operating characteristic curve (AUC) and total accuracy using mutual information based feature selection. They concluded that Genetic Programming together with mutual information based feature selection is the most efficient alternative to the existing colon cancer prediction techniques. [4]

Several works quantitative structure-activity relationships (QSAR) of anti-human immunodeficiency virus (HIV) molecules were studied by different statistical methods and non-linear models. But few studies have used the heuristic methods. In a paper called **„A hybrid decision trees-adaptive neuro-fuzzy inference system in prediction of anti-HIV molecules"**, a hybrid decision tree (DT) and adaptive neuro-fuzzy inference system (ANFIS) is used for the prediction of inhibitory activity of anti-VIH molecules. DT algorithm is utilized to select the most important variables in QSAR modeling and then these variables were used as inputs of ANFIS to predict the anti-HIV activity. The mode's predictions were compared with other methods and the results indicated that the proposed models in this work are superior over the others. [6]

To handle problems created by large data sets, in **„Tree Decomposition for Large-Scale SVM Problems"** paper is proposed a method that uses a decision tree to decompose a given data space and train SVMs on the decomposed regions. Although there are other means of decomposing a data space, they showed that the decision tree has several merits for large-cale SVM training. First, it can classify some data points by its own means, thereby reducing the cost of SVM training for the remaining data points. Second, it is efficient in determining the parameter values that maximize the validation accuracy, which helps maintain good test accuracy. Third, the tree decomposition method can derive a generalization error bound for the classifier. For data sets whose size can be handled by current non-linear, or kernel-based, SVM training techniques, the proposed method can speed up the training by a factor of thousands, and still achieve comparable test accuracy. [7]

„**Comparison of Seven Algorithms to Predict Breast Cancer Survival**", published in 2007, showed that Decision Trees (J48) had the highest sensitivity of all the other algorithms (Logistic regression model, Artificial Neural Network (ANN), Naive Bayes, Bayes Net, Decision Trees with naive Bayes, Decision Trees (ID3)) with an accuracy of 85.6%. [8] The same problem has been studied by Delen et al. in **"Predicting Breast Cancer Survivability: A Comparison of Three Data Mining Methods"** [9] and by Bellaachia et al. in **"Predicting Breast Cancer Survivability using Data Mining Techniques".** [10] Delen et al. reported that the Decision Trees algorithm had a much better performance than the other two algorithms, Artificial Neural Network and Logistic Regression model. and Bellaachia et al., reported that the Decision Trees algorithm had a better performance than Artificial Neural Network and Naive Bayes algorithms. Also they reported that Decision Trees showed the best performance for accuracy.

4. Advantages and Disadvantages of using Decision Trees

Decision trees offer advantages over other methods of analyzing alternatives. They are:

- **Graphic.** You can represent decision alternatives, possible outcomes, and chance events schematically. The visual approach is particularly helpful in comprehending sequential decisions and outcome dependencies.
- **Efficient.** You can quickly express complex alternatives clearly. You can easily modify a decision tree as new information becomes available. Set up a decision tree to compare how changing input values affect various decision alternatives. Standard decision tree notation is easy to adopt.
- **Revealing.** You can compare competing alternatives-even without complete information-in terms of risk and probable value. The Expected Value (EV) term combines relative investment costs, anticipated payoffs, and uncertainties

into a single numerical value. The EV reveals the overall merits of competing alternatives.

- **Complementary.** You can use decision trees in conjunction with other project management tools. For example, the decision tree method can help evaluate project schedules. [2]
- Decision trees are self-explanatory and when compacted they are also easy to follow. In other words if the decision trees has a reasonable number of leaves, it can be grasped by non-professional users. Furthermore decision trees can be converted to a set of rules. Thus, this representation is considered as comprehensible.
- Decision trees can handle both nominal and numerical attributes.
- Decision trees representation is rich enough to represent any discrete-value classifier.
- Decision trees are capable of handling datasets that may have errors.
- Decision trees are capable of handling datasets that may have missing values.
- Decision trees are considered to be a nonparametric method. This means that decision trees have no assumptions about the space distribution and the classifier structure.

On the other hand, decision trees have disadvantages such as:

- Most of the algorithms (like ID3 and C4.5) require that the target attribute will have only discrete values.
- As decision trees use the "divide and conquer" method, they tend to perform well if a few highly relevant attributes exist, but less so if many complex interactions are present. One of the reasons for this is that other classifiers can compactly describe a classifier that would be very challenging to represent using a decision tree.
- The greedy characteristic of decision trees leads to another disadvantage that should be pointed out. This is its over-sensitivity to the training set, to irrelevant attributes and to noise. [3]

5. Decision Tree Extensions

In this section I will discuss some of the most popular extensions to the classical decision tree induction paradigm.

5.1. Obliviuous Decision Trees

Obliviuous decision trees are decision trees for which all nodes at the same level test the same feature. Despite its restriction, oblivious decision trees are found to be effective for feature selection.

The principal difference between the oblivious decision tree and a regular decision tree structure is the constant ordering of input attributes at every terminal node of the oblivious decision tree, the property which is necessary for minimizing the overall subset of input attributes. An oblivious decision tree is usually built by a greedy algorithm, which tries to maximize the mutual information measure in every layer. The recursive search for explaining attributes is terminated when there is no attribute that explains the target with statistical significance.

5.2. Fuzzy Decision Trees

In classical decision trees, an instance can be associated with only one branch of the tree. Fuzzy decision trees (FDT) may simultaneously assign more than one branch to the same instance with gradual certainty.

FDTs preserve the symbolic structure of the tree and its comprehensibility. FDT can represent concepts with graduated characteristics by producing real-valued outputs with gradual shifts.

Janikow (1998) presented a complete framework for building a fuzzy tree including several inference procedures based on conflict resolution in rule-based systems and efficient approximate reasoning methods.

Olaru and Wehenkel (2003) presented a new fuzzy decision trees called soft decision trees (SDT). This approach combines tree-growing and pruning, to determine the structure of the soft decision tree, with regitting and backfitting, to improve its generalization capabilities. They empirically showed that soft decision trees are significantly more accurate than standard decision trees. Moreover, a global model variance study shows a much lower variance for soft decision trees than for standard trees as a direct cause of the improved accuracy.

Peng (2004) has used FDT to improve the performance of the classical inductive learning approach in manufacturing processes. Peng (2004) proposed to use soft discretization of continuous-valued attributes. It has been shown that FDT can deal with the noise or uncertainties existing in the data collected in industrial systems.

5.3. Decision Trees Inducers for Large Datasets

With the recent growth in the amount of data collected by information systems, there is a need for decision trees that can handle large datasets. Catlett (1991) has examined two methods for efficiently growing decision trees from a large database by reducing the computation complexity required for induction. However, the Catlett method requires that all data will be loaded into the main memory before induction. That is to say, the largest dataset that can be induced is bounded by the memory size. Fifield (1992) suggests parallel implementation of the ID3 Algorithm. However, like Catlett, it assumes that all dataset can fit in the main memory.

Chan and Stolfo (1997) suggest partitioning the datasets into several disjointed datasets, so that each dataset is loaded separately into the memory and used to induce a decision tree. The decision trees are then combined to create a single classifier. However, the experimental results indicate that partition may reduce the classification performance, meaning that the classification accuracy of the combined decision trees is not as good as the accuracy of a single decision tree induced from the entire dataset.

The SLIQ algorithm (Mehta *et al*., 1996) does not require loading the entire dataset into the main memory, instead it uses a secondary memory (disk). In other words, a certain instance is not necessarily resident in the main memory all the time. SLIQ creates a single decision tree from the entire dataset. However, this method also has an upper limit for the largest dataset that can be processed, because it uses a data structure that scales with the dataset size and this data structure must be resident in main memory all the time. The SPRINT algorithm uses a similar approach (Shafer *et al*., 1996). This algorithm induces decision trees relatively quickly and removes all of the memory restrictions from decision tree induction. SPRINT scales any impurity based split criteria for large datasets. Gehrke *et al* (2000) introduced RainForest; aunifying framework for decision tree classifiers that are capable of scaling any specific algorithms from the literature (including C4.5, CART and CHAID). In addition to its generality, RainForest improves SPRINT by a factor of three. In contrast to SPRINT, however, RainForest requires a certain minimum amount of main memory, proportional to the set of distinct values in a column of the input relation. However, this requirement is considered modest and reasonable.

5.4. Incremental Induction

Most of the decision trees inducers require rebuilding the tree from scratch for reflecting new data that has become available. Several researches have adressed the issue of updating decision trees incremetally. Utgoff (1989b, 1997) presents several methods for updating decision trees incrementally. An extension to the CART algorithm is capable of inducing incrementally. [3]

6. Application domains

Decision trees are useful in many application domains as:

- Manufacturing: lr18, lr14
- Security: lr7, 110
- Medicine: lr2, lr9
- For many data mining tasks such as: supervised learning (lr6, lr12, lr15), unsupervised learning (lr13, lr8, lr5, lr16) and genetic algorithms (lr17, lr11, lr1, lr4).

7. Conclusions

Decision trees are known as highly efficient tools of machine learning and data mining, capable to produce accurate and easy-to-understand models. They are robust and perform well with large data in short time. As we can see in the presented reports, decision tree is a very efficient predictive model.

References

[1] Barry de Ville, „Decision Trees for Business Intelligence and Data Mining: Using SAS Enterprise Miner”, SAS Institute Inc., Cary, NC, USA, 2006.

[2] Rafael Olivas, „Decision Trees – A primer for Decision-making Professionals”, 2007

[3] Oded Maimon, Lior Rokach, „Data Mining and Knowledge Discovery Handbook”, Second Edition, Springer Science+Business Media, p. 149-174

[4] Ashwinikumar Kulkarni, B.S.C. Naveen Kumar, Vadlamani Ravi, Upadhyayula Suryanarayana Murthym, „Colon cancer prediction with genetics profiles using evolutionary techniques”, Expert Systems with Applications, Volume 38, Issue 3, March 2010, Pages 2752-2757

[5] Ruey-Shiang Guh, Tsung-Chieh Jackson Wu, Shao-Ping Weng, „Integrating Genetic Algorithm and Decision Tree learning for Assistance in Predicting in Vitro Fertilization outcomes”, Expert Systems with Applications, Article in Press, Corrected Proof, 2010

[6] „A hybrid decision trees – adaptive neuro fuzzy inference system in prediction of anti-HIV molecules”, Expert Systems with Applications, Article in Press, Corrected Proof, 2010

[7] Fu Chang, Chien-Yang Guo, Xiao-Rong Lin, Chi-Jen Lu „Tree Decomposition for Large-Scale SVM Problems”, Journal of Machine Learning Research 11 (2010) 2935-2972

[8] Arihito Endo, Taeko Shibata, Hiroshi Tanaka, “Comparison of Seven Algorithms to Predict Breast Cancer Survivald”, Biomedical Soft Computing and Human Sciences, Vol.13, No.2, pp.11-16 (2008)

[9] Delen, D., G. Walker, A Kadam (2005, "Predicting Breast Cancer Survivability: A Comparison of Three Data Mining Methods", Artificial Intelligent Medical, Vol. 34, No.2, pp 113-27

[10] Bellaachia, A., Guven, E. (2006) “Predicting Breast Cancer Survivability using Data Mining Techniques”, Proceedings of Ninth Workshop on Mining Scientific and Engineering Datasets in conjunction with the Sixth SIAM International Conference on Data Mining (SDM 2006).