On Features of Decision Trees as a Technique of Knowledge Modelling

Bonner Richard F.
Malardalens University Vasteras
Vasteras, Sweden
richard.bonner@mdh.se

Galant Violetta
University of Economics
Wroclaw, Poland
galant@ksk-2.iie.ae.wroc.pl

Owoc Mieczyslaw L
University of Economics
Wroclaw, Poland
owoc@ksk-2.iie.ae.wroc.pl

Abstract

A promising approach to knowledge base modelling today is machine learning, whereby decision trees play an important role in constructing knowledge by inductive methods. It is known, however, that decision tree methods have weak points. There is, in particular, the problem of heterogeneity of distribution of classification attributes: classes with few instances yield poor description and may disappear altogether. Another problem concerns the treatment of new instances during learning, so noise is not confused with data announcing a class of interest. The paper takes up these and other problems, and suggests a framework for analysis of decision trees within stochastic control theory and algorithmic complexity.

1. Introduction

Knowledge modelling is widely recognised as the critical phase of knowledge acquisition. Before an expert system can be built, knowledge must somehow be identified and collected, and a model of domain knowledge must be constructed. However, satisfactory tools to support knowledge engineers during the modelling phase are lacking, and, consequently, much of the work is still done 'manually', relying on a variety of techniques, some from quite distant disciplines. Obviously, no single technique will work well for all domains and all modelling problems.

Techniques must somehow be matched to problems - a hard task unless their general properties have been well understood. The present note visits one of the most widely used techniques of modelling, informally known as decision trees. Accounts of the topic are numerous; in the context of machine learning, the non-expert reader may wish to consult, for example, [9, Ch.3] and the references given there. Assuming [9], we introduce decision trees in the context of the knowledge-based system life cycle. For brevity, we order their main characteristics by usefulness into two groups, of strong points and weak points, respectively. The second group has more research interest, open questions being naturally a week point from the perspective of application. Problems of dynamics of decision trees are emphasised, addressing the theoretical foundation of machine learning approaches in knowledge acquisition [13]. We conclude with some suggestions for research.

Proceedings of the Workshop on Computer Science and Information Technologies CSIT'99
Moscow, Russia, 1999

2. Knowledge Modelling and Decision Trees

Recall, the classical knowledge base system life cycle consists of three phases: knowledge acquisition, knowledge representation and system implementation; see, for example, [4]. The first phase consists in transforming knowledge structures (coming from different sources) onto a knowledge model. The goal of the second phase is to formalise the knowledge model, using propositional logic or predicate calculus, semantic networks, rules or/and frames, or other formalism. In the final phase, a system containing the knowledge base is elaborated. The forthcoming discussion concerns knowledge acquisition - the first of the three phases.

Knowledge acquisition starts with domain recognition, where potential and available knowledge are analysed with respect to the goals of the system. It then continues with knowledge elicitation, where knowledge is gathered so that problems may be divided into sub-problems, basic entities, information items, and so that structural relationships may be identified. Knowledge acquisition ends with the creation of a knowledge model, which should be defined explicitly and independently from the methods of subsequent life cycle phases. Very seldom do software tools support knowledge acquisition [6].

According to common practice [8], a knowledge model should expressly contain three components: facts from knowledge domain, inferential relationships, and, problem solving strategies. Different methods may be used for each of these components. In expressing domain knowledge, for example, semantic nets (trees) are often used to represent the hierarchy of objects, while space and/or time dependencies among the objects are pictured by more general graphs. Decision tables, decision trees or inference nets, can support the representation of inferential relationships. A sequence of steps (decisions) required to solve a problem is naturally represented by a decision tree, whence the use of flow diagrams and decomposition trees to depict problem solving strategies.

It is clear that decision trees often play a major role in a modelling process. This is hardly surprising considering their fundamental nature: a decision tree is nothing but a graphical representation of the search space of a problem. Roughly, a node in the tree represents a decision (a choice) to be made when solving the problem, and the branches growing out from the node depict the possible values (outcomes) of the decision. To solve the problem, one then traces through its tree using problem data to choose a branch at each node.

3. Strong Points of Decision Trees

Decision trees are now widely used in modelling in a variety of fields, and are commonly considered as a basic modelling tool with many valuable properties [10]. Their most appreciated practical advantage is of cognitive nature: they allow a human expert to easily comprehend the solution of a problem, say, a classification task. The graphical picturing makes it possible also for non-experts to read and interpret abstract notions and complex logistics. The practical value of cognitive simplicity should not be underestimated - it is well known that excellent modelling methods fail miserably in practice if excessively complex or abstract. In the social context of modelling, whether developing an information system or running a firm, models must be easy to work with, yet non-ambiguous and accurate. Perhaps for this reason, trees have a long tradition in supporting management decision.

In the context of knowledge modelling, decision trees have been from the very beginning used as a technique of inductive learning. A decision tree is induced from a set of examples, whether to be used in a search process directly, or translated into a set of decision rules for the developed knowledge-based system. In practice, the trees induced from examples often stabilise as further examples arrive, in which case they are said to have good generalisation properties. There is general faith in the knowledge industry that decision trees generalise well.

Both the cognitive simplicity and the generalisation properties of decision trees are still waiting to be examined in terms of some basic theory such as algorithmic information theory [7]. It is not unusual for a decision tree to optimally encode empirical data or a strategy ("program") for solving a problem. The exact conditions for this to happen may need clarification, but the circumstance itself puts decision trees in a special category of tools for building non-redundant models.

4. Weak Points of Decision Trees

Perhaps the weakest point of decision trees in modelling is their sensitivity to changes in data, hence also to noise. In contrast to the rather stable problems of learning of the classical kind, like the least squares regression applied to a parametrised family of functions such as a feed-forward artificial neural network, the problem of learning decision trees is inherently ill-posed. Roughly, the instability comes from the ambiguity in the choice of a tree for given data, and there seems to be little known about removing this ambiguity through effective selection criteria. Thus, the general problem of dynamics of decision trees still being open, we think it may be wise to use this modelling tool with caution until it has been better understood.

A point of some concern is also the procedural (computational) and hence deterministic nature of knowledge represented by a decision tree, while it is well known that some types of knowledge to be modelled are essentially non-deterministic. It is not obvious how to resolve this conflict in general. The classical approach of stochastic dynamics, as in Markov chains, for example, does this by lifting the domain of a model from the states of a problem to their probability distributions. This, however, somewhat complicates the models and, furthermore, leaves them open to the usual questions of interpretation of a probabilistic model in practice.

5. Towards a Theory of Decision Trees

We comment briefly the theoretical foundations for modelling with decision trees. In the first instance, we observe, the topic could be put into an abstract framework of algebraic complexity [3] and classical stochastic analysis [1], noting in particular the recent work in the theory of empirical processes [11]. An even wider perspective of economic games and stochastic control theory should further simplify things and suggest useful criteria for tree selection [2]. In the second instance, the known phenomenon of optimal encoding of information by decision trees should be examined in the light of modern theories of information, such as the algorithmic information theory [7].

We elaborate only the first point. Start by identifying decision trees as computational trees, cf [3], in the lattice of subalgebras of a sigma-algebra. The attributes are then thought of as random variables with respect to a fixed sigma-algebra A of measurable subsets of an abstract set X. Assume for simplicity that the variables are numerical, i.e. real-valued. Roughly, the general problem is to constructively approximate one of the variables (the "classifying attribute") by the remaining ones (the "descriptive attributes"). Two versions of the problem are natural to consider, the static and the dynamic one, depending on the type of data admitted.

In the first case, the data is a fixed probability measure on the algebra A, usually an empirical measure, i.e. the normalised uniform measure on a finite subset of X. The construction of an approximant to the classifying attribute is then expressed as a computational tree, representing successive sigma-algebras generated by increasing finite subsets of the defining attributes. Most of the literature on decision trees concerns the construction of "good" computational trees in this static case. Classic stochastic theory [1] applies, suggesting conditional entropy and expectation as criteria for growing trees and choosing approximants. Variations on this theme are numerous, cf [5]. Understandably, the static case sheds little light on the behaviour of trees with evolving data.

In the second case, it is natural to see the data as a sequence of independent random points in *X*, each distributed according to a (usually unknown) probability measure on *A*. This generates a sequence of random empirical measures on *A*, each of which serves to construct an approximant to the classifying attribute in the form of a (random) computational tree. While questions about the convergence of these empirical measures on families of attributes are classical in the theory of empirical processes [11], not least in the context of learning theories [8], the general problem of the behaviour of corresponding random trees remains open. Clearly, the latter is exactly the question of learning and generalisation in decision trees.

One of the known obstacles to a satisfactory theory of decision trees is the inherent ambiguity in their selection, the criteria of selection being a priori also open to choice. A way to reduce ambivalence beyond orderings of information-theoretical nature is to embed the dynamic tree formation in a larger dynamic process representing an economic game [2], in which an ordering of strategies further orders trees. Instances of this approach are well known, but, once again, a systematic study is yet to be carried out.

6. Conclusion

Sets of rules in the form of decision trees generated by machine learning techniques are a good starting point for further tasks aiming at developing a knowledge base. In this context, their most valuable qualities include: general cognitive simplicity, ease of representing and clarifying hierarchical relations, respecting managers' experience (decision trees are widely used by managers), and the ease of keeping track of inference paths when analysing decisions. Furthermore, practitioners generally believe that trees well generalise empirical data, whence the good acceptance of decision trees in machine learning approaches to knowledge modelling.

However, much has to be done before decision trees are granted the status of a universal modelling tool. To begin with, the much spoken for cognitive simplicity of decision trees is only true of small trees; large trees must first be reduced to a family of trees of manageable size - a non-trivial formal problem. Furthermore, the very relevance of human cognitive characteristics of modelling tools is being put in question by the advancement of intelligent machines.

While a tree inferred from static data may be simple, the evolution of trees with evolving data need not be simple at all. Indeed, despite the wide belief that trees generalise well, it is not hard to construct examples where small changes in data result in large changes in the induced tree. This discontinuity phenomenon makes it hard to deal with noise in the data, adding a new dimension to the usual questions of convergence of empirical processes.

Finally, speaking in general, the presently available criteria for growing trees leave much to be desired; at some stage, gardening intuition must make place for mathematics.

References

1. Billingsley, P., Ergodic Theory and Information, Wiley, 1965.
2. Bonner, R., Economics of Information and Acquisition of Knowledge, pp 193-201 in: A. Baborski, ed., Acquisition of Knowledge from Databases, Wroclaw University of Economics, 1998.
3. Bürgisser, P., et al., Algebraic Complexity Theory, Springer, 1997.
4. Durkin, J., Expert Systems: Design and Development. Prentice-Hall, New Jersey, 1994.
5. Galant, V., Generating Decision Trees, in Proceed. Business Information Systems ’98, Poznan, 1998 (in Polish).
6. Krause F. L. and J. Schlingheider, Development and Design with Knowledge-Based Software Tools - An Overview. Expert Systems with Applications. Vol. 8, No. 2, 1995 pp. 233-247
7. Li, M. and P. Vitanyi, An Introduction to Kolmogorov Complexity and its Applications, 2:nd edition, Springer, 1997
8. Mockler, R. J. and D. G. Dologite, Knowledge-Based Systems. An Introduction to Expert Systems. Macmillan, New York, 1992.
9. Mitchell, T. M., Machine Learning, McGraw-Hill, 1997.
10. Owoc, M. L. and V. Galant, Validation of Rule-Based Systems Generated by Classification Algorithms, in: Proceedings of the Information Systems Development Conference, Bled'98 – Slovenia (to appear)
11. Pollard, D., Empirical Processes, Intitute of Mathematical Statistics, 1990.
12. Vidyasagar, M., A Theory of Learning and Generalisation, Springer, 1997.
13. Weiss, S. M. and C. A. Kulikowski, Computer Systems that Learn. Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning and Expert Systems. San Mateo, 1991.