Techniques for efficient empirical induction

Geoffrey I. Webb
Division of Computing and Mathematics
Deakin University
Victoria, 3217

Abstract

This paper describes the LEI algorithm for empirical induction. The LEI algorithm provides efficient empirical induction for discrete attribute value data. It derives a classification procedure in the form of a set of predicate logic classification rules. This contrasts with the only other efficient approach to exhaustive empirical induction, the derivatives of the CLS algorithm, which present their classification procedures in the form of a decision tree. The LEI algorithm will always find the simplest non-disjunctive rule that correctly classifies all examples of a single class where such a rule exists.

1 Introduction

Empirical induction is the discovery of classification rules from examples. An empirical induction algorithm takes as input a set of example instances (the example set) and outputs a classification procedure. The aim is to produce a classification procedure that can correctly classify novel instances.

A classification procedure will generally be represented herein by a data structure (such as a decision tree or a set of classification rules). The reader should remember, however, that the complete classification procedure consists of that data structure and a method for applying that data structure to an instance to produce a classification.

A number of approaches to automated empirical induction have been developed. These can be divided into two major categories—statistic based and logic based.

The statistic based approaches have evolved from CLS (Hunt, Marin and Stone, 1966). ID3 (Quinlan, 1986) is a more recent version of this approach. These algorithms utilise attribute-value data for classification. That is, relevant details of instances are expressed solely in terms of values of specified attributes of the instances. Although the initial approaches only supported binary classification (classifying instances as either belonging or not belonging to a single class) more recent derivatives of the algorithm support classification into multiple mutually exclusive classes.

The statistical approaches express their classification procedures in the form of decision trees. The root decision node is formed by using a statistical test to determine which attribute best differentiates between classes. The instances are then partitioned on different values for that attribute and the appropriate sets of instances associated with each branch from the decision node. This process is repeated for the terminal branches of the tree until the set of instances associated with each terminal branch contains only instances of a single class. The node descending from each terminal
branch is then labelled with the class of the instances that are associated therewith. The internal nodes of the decision tree now represent classification tests. The terminal nodes represent the classification to assign to any instance passing all tests that lead to that node.

By using windows to restrict the instances examined to a small subset of those available, the ID3 algorithm is able to provide very efficient empirical induction.

However, decision trees are difficult to read (Quinlan, 1987b). Further, they tend to include superfluous complexity (Quinlan, 1987a). As a consequence, recent research has developed methods for taking the decision trees developed by the ID3 algorithm and using them to develop logical classification rules (Quinlan, 1987a). This has been found to produce classification procedures that are both more comprehensible and more accurate.

By contrast to the statistical approaches, previous logic based approaches have not provided efficient implementations, except at the cost of the use of heuristics that severely limit the potential classification procedures that they can discover.

The logic based approaches use various predicate logic based notations to represent relevant details of the instances to be classified. The classification procedure is usually represented by a body of logical implications. An implication has the form if condition then conclusion. The condition is called an antecedent and the conclusion is called a consequent. The implications used in a classification system have consequents that represent the desired classifications and antecedents that specify the characteristics of the instances to which those classifications apply.

Logical implications provide a far richer and more expressive description language than the attribute-value analyses supported by the statistic based approaches.

One logic based method is to form a single implication as a hypothesis classification rule. This rule is then systematically varied to account for the body of available examples. Two basic manipulations may be performed—generalisation and specialisation. Generalisation involves weakening the antecedent so that the consequent (classification) will apply to more instances. Specialisation involves strengthening the antecedent so that the consequent will apply to fewer instances. Note that specialisations of a false implication may be true whereas specialisations of a true implication must be true. By contrast, generalisations of a true implication may be false and generalisations of a false implication must be false.

One approach to this method (see, for example, Winston, 1975) involves setting the antecedent of the initial hypothesis rule to a description of the first example instance encountered. Further example instances are then examined. Positive examples (instances of the class in question) are used to generalise the current hypothesis. Negative examples (instances that do not belong to the relevant class) are used to specialise the hypothesis. Thus, the system starts from a very specific hypothesis and works toward a more general hypothesis that is consistent with the available evidence.

By contrast, another approach (see, for example, Buchanan and Feigenbaum, 1978) starts with the most general hypothesis, that everything belongs to the class in question, and utilises the example set to work toward a more specialised classification rule.

The version space approach, developed by Mitchell (1977), maintains a range of candi-