Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Note

Totally optimal decision rules

Talha Amin *, Mikhail Moshkov

Computer, Electrical and Mathematical Sciences & Engineering Division, King Abdullah University of Science and Technology, Thuwal 23955-6900, Saudi Arabia

A R T I C L E I N F O

Article history:
Received 12 May 2016
Received in revised form 19 October 2017
Accepted 19 October 2017
Available online xxxx

Keywords:
Decision rules
Patterns
Length
Coverage

A B S T R A C T

Optimality of decision rules (patterns) can be measured in many ways. One of these is referred to as length. Length signifies the number of terms in a decision rule and is optimally minimized. Another, coverage represents the width of a rule's applicability and generality. As such, it is desirable to maximize coverage. A totally optimal decision rule is a decision rule that has the minimum possible length and the maximum possible coverage. This paper presents a method for determining the presence of totally optimal decision rules for "complete" decision tables (representations of total functions in which different variables can have domains of differing values). Depending on the cardinalities of the domains, we can either guarantee for each tuple of values of the function that totally optimal rules exist for each row of the table (as in the case of total Boolean functions where the cardinalities are equal to 2) or, for each row, we can find a tuple of values of the function for which totally optimal rules do not exist for this row.

1. Introduction

A decision rule (pattern) is a mapping from a set of conditions to an outcome. These patterns are widely used in applications concerning data mining [12], knowledge representation and discovery, and machine learning [8]. Many problems in these areas can be solved with multiple approaches including systems of decision rules [9], decision trees [22], logical representations [19], k-nearest neighbor [11], neural networks, and support vector machines [10].

Although there are many comparable methods, decision rules tend to be the most meaningful and easily understood by humans [13]. This allows them to excel where knowledge representation is concerned. Even among decision rules some are better than others. For example, short decision rules are easy to remember and simple to understand while rules that cover many cases are more general and applicable. These and other such criteria mean that it is important to consider which rules to include in a system.

There are many different ways to design and analyze decision rules, such as brute force, genetic algorithms [25], Boolean reasoning [21], derivation from decision trees [23,18], sequential covering procedures [9,13], greedy algorithms [17], and dynamic programming [2,26]. Implementations of many of these methods can be found in programs such as LERS [14], RSES [5], Rosetta [20], Weka [15], TRS Library [24], and DAGGER [1].

According to Hammer, Kogan, Simeone, and Szedmák [16], "it has been observed in empirical studies and practical applications that some patterns are more suitable than others for use in data analysis". The decision rules that we consider are very similar to patterns from logical analysis of data [6,7,16], and the same consideration can be applied to them. Hammer et al. [16] go on to consider the Pareto optimality of patterns with regards to various criteria, developing algorithms to optimize a given pattern with respect to any one criteria.

* Corresponding author.
E-mail addresses: talha.amin@kaust.edu.sa (T. Amin), mikhail.moshkov@kaust.edu.sa (M. Moshkov).

https://doi.org/10.1016/j.dam.2017.10.024

In this paper, we consider instead the existence of decision rules that are totally optimal with respect to both of our chosen criteria, thus removing the need to consider Pareto optimality. The criteria we consider are length and coverage. Length is an important criteria because short rules are easier to understand. Coverage is important because the more coverage a rule has, the more general it is and the wider its application. Intuitively, it might seem that considering length and coverage will also always result in totally optimal rules. After all, shorter rules have less restrictions, leading to greater coverage. However, based on results seen in [2], we know that this is often not the case.

The main result of our work defines a system of inequalities for "complete" decision tables (representations of total functions in which different variables can have domains of differing values). When all inequalities in this system are true then, for each total function, we can guarantee the existence of totally optimal decision rules for each row. Otherwise, for each row, we can guarantee the existence of at least one total function such that totally optimal decision rules do not exist for the considered row. In particular, we utilize this system of inequalities to prove that for any Boolean function that is defined on all values of input (total Boolean functions) there exist totally optimal decision rules. The same situation applies to total functions of k-valued logic ($k > 2$). Furthermore, we study datasets from the UCI Machine Learning Repository [4] and show that a considerable amount of datasets are complete tables that can be proven to have totally optimal decision rules.

This paper consists of three sections. Section 2 provides basic definitions and introduces concepts necessary for understanding the rest of the work. Section 3 then uses these concepts to derive two theorems regarding totally optimal decision rules. One of these theorems is then used to prove the existence of totally optimal decision rules for all total Boolean functions. Section 4 presents the findings when working with the UCI Machine Learning Repository. Finally, Section 5 contains conclusions.

2. Decision tables and rules

In this section, we consider the main notions and notations related to decision tables and decision rules.

2.1. Decision tables

A *decision table* is a rectangular table T with $n \geq 1$ columns filled with numbers from the set of nonnegative integers. Columns of the table are labeled with pairwise different *conditional* attributes $f_1, \ldots, f_n$, respectively. Rows of the table are pairwise different, and each row is labeled with a nonnegative integer which is interpreted as a decision (a value of the *decision* attribute d). Rows of the table are interpreted as tuples of values of conditional attributes. Note that in the definition of decision table, instead of natural numbers we can use symbols of arbitrary infinite alphabet.

A decision table is called *empty* if it has no rows. The table T is called *degenerate* if it is empty or all rows of T are labeled with the same decision. We denote by $\dim(T)$ the number of columns (conditional attributes) in T and by $N(T)$ the number of rows in the table T.

For any conditional attribute $f_i \in \{f_1, \ldots, f_n\}$, we denote by $E(T, f_i)$ the set of values of the attribute f_i in the table T. We denote by $E(T)$ the set of conditional attributes for which $|E(T, f_i)| \geq 2$.

Let T be a nonempty decision table. A *subtable* of T is a table obtained from T by the removal of some rows. Let $f_{i_1}, \ldots, f_{i_m} \in \{f_1, \ldots, f_n\}$ and $a_1, \ldots, a_m \in \omega$ where ω is the set of nonnegative integers. We denote by $T(f_{i_1}, a_1) \ldots (f_{i_m}, a_m)$ the subtable of the table T containing the rows from T which at the intersection with the columns $f_{i_1}, \ldots, f_{i_m}$ have numbers $a_1, \ldots, a_m$, respectively.

A *complete* decision table is a representation of a total function in which different variables can have domains of differing values. Given that $B_1, \ldots, B_n$ are nonempty finite subsets of the set of nonnegative integers, $I = B_1 \times \cdots \times B_n$, and $\nu : I \to \omega$, the pair $T = (I, \nu)$ can be used to describe the complete decision table T with conditional attributes $f_1, \ldots, f_n$ that have values from the sets $B_1, \ldots, B_n$, respectively. n-Tuples from I are rows of T, and the value $\nu(b_1, \ldots, b_n)$ is the decision attached to a row $(b_1, \ldots, b_n) \in I$. The table T is nondegenerate if and only if ν is a non-constant function (which has at least two different values).

2.2. Decision rules

Let T be a decision table with n conditional attributes $f_1, \ldots, f_n$ and $r = (b_1, \ldots, b_n)$ be a row of T. A *decision rule over* T is an expression of the kind

$$f_{i_1} = a_1 \wedge \cdots \wedge f_{i_m} = a_m \to t \tag{1}$$

where $f_{i_1}, \ldots, f_{i_m} \in \{f_1, \ldots, f_n\}$, and $a_1, \ldots, a_m, t$ are nonnegative integers. It is possible that $m = 0$. For the considered rule, we denote $T^0 = T$, and if $m > 0$ we denote $T^j = T(f_{i_1}, a_1) \ldots (f_{i_j}, a_j)$ for $j = 1, \ldots, m$. We will say that the decision rule (1) *covers* the row $r = (b_1, \ldots, b_n)$ if $b_{i_1} = a_1, \ldots, b_{i_m} = a_m$.

A decision rule (1) over T is called a *decision rule for* T if all the rows of T that are covered by the rule are labeled with the decision t, and either $m = 0$, or $m > 0$ and, for $j = 1, \ldots, m$, T^{j-1} is not degenerate, and $f_{i_j} \in E(T^{j-1})$. A decision rule (1) for T is called a *decision rule for* T *and* r if it covers r.

We denote by $DR(T)$ the set of decision rules for T. By $DR(T, r)$ we denote the set of decision rules for T and r.

Please cite this article in press as: T. Amin, M. Moshkov, Totally optimal decision rules, Discrete Applied Mathematics (2017), https://doi.org/10.1016/j.dam.2017.10.024.