Decision Rules for Decision Tables with Many-Valued Decisions

Igor Chikalov[1] and Beata Zielosko[1,2,⋆]

[1] Mathematical and Computer Sciences and Engineering Division
King Abdullah University of Science and Technology
Thuwal 23955-6900, Saudi Arabia
[2] Institute of Computer Science, University of Silesia
39, Będzińska St., 41-200 Sosnowiec, Poland
{igor.chikalov,beata.zielosko}@kaust.edu.sa

Abstract. In the paper, authors presents a greedy algorithm for construction of exact and partial decision rules for decision tables with many-valued decisions. Exact decision rules can be over-fitted, so instead of exact decision rules with many attributes, it is more appropriate to work with partial decision rules with smaller number of attributes. Based on results for set cover problem authors study bounds on accuracy of greedy algorithm for exact and partial decision rule construction, and complexity of the problem of minimization of decision rule length.

Keywords: decision tables with many-valued decisions, decision rules, greedy algorithm, set cover problem.

1 Introduction

Decision tables with many-valued decisions arise often in various applications. In contrast to decision tables with one-valued decisions, in decision tables with many-valued decisions each row is labeled with a nonempty finite set of natural numbers (decisions). If we want to find all decisions corresponding to a row, we deal with the same mathematical object as decision table with one-valued decisions: it is enough to code different sets of decisions by different numbers. However, if we want to find one (arbitrary) decision from the set attached to a row, we have essentially different situation.

In particular, in rough set theory [4] decision tables are considered often that have equal rows labeled with different decisions. The set of decisions attached to equal rows is called the *generalized decision* for each of these equal rows. The usual way is to find for a given row its generalized decision. However, the problems of finding an arbitrary decision or one of the most frequent decisions from the generalized decision look also reasonable.

Decision rules can be considered as a way of knowledge representation. In applications we often deal with decision tables which contain noisy data. In

⋆ This research was supported by the Russian Federal Program "Scientists and Educators in Russia of Innovations", contract number 02.740.11.5131.

J.T. Yao et al. (Eds): RSKT 2011, LNCS 6954, pp. 763–768, 2011.

this case, exact decision rules can be over-fitted, i.e., depend essentially on the noise. So, instead of exact decision rules with many attributes, it is more appropriate to work with partial decision rules with smaller number of attributes, which separate almost all different rows with different decisions. The problem of construction decision rules with minimum number of attributes is NP-hard. Therefore, we should consider approximate polynomial algorithms for decision rule optimization.

From obtained bounds on greedy algorithm accuracy and results proved in [3] it follows, that under some natural assumptions on the class NP, the greedy algorithm is close to the best polynomial approximate algorithms for minimization of partial decision rule length.

In the paper, we present greedy algorithm for decision rule construction for decision table T with many-valued decisions, where each row has a set of decisions. For each row and each value of decision from the set of decisions, and α such that $0 \leq \alpha < 1$, algorithm constructs an α-decision rule. For every row algorithm chooses a rule with the minimum length.

We study binary decision tables with many-valued decisions but presented approach can be used also for decision tables with $k \geq 3$ values of attributes.

The paper consists of five sections. In Sect. 2, main notions are considered. In Sect. 3, set cover problem, greedy algorithm and bounds on accuracy of greedy algorithm for exact and partial covers are presented. Section 4 contains bounds on accuracy of greedy algorithm for exact and partial decision rules. Section 5 contains conclusions.

2 Main Notions

A *binary decision table with many-valued* decisions is a rectangular table T filled by numbers from the set $\{0, 1\}$. Columns of this table are labeled with attributes $f_1, \ldots, f_n$. Rows of the table are pairwise different, and each row r is labeled with a nonempty finite set $D(r)$ of natural numbers (set of decisions). Note that each decision table with one-valued decisions can be interpreted also as a decision table with many-valued decisions. In such table each row is labeled with a set of decisions which has one element.

The table T is called *degenerate* if there is a decision d such that $d \in D(r)$ for any row r of T or T has no rows.

Let $r = (b_1, \ldots, b_n)$ be a row of T labeled with the set of decisions $D(r)$ and $d \in D(r)$. By $U(T, r, d)$ we denote the set of rows r' from T for which $d \notin D(r')$. We will say that an attribute f_i separates a row $r' \in U(T, r, d)$ from the row r if the rows r and r' have different numbers at the intersection with the column f_i. The pair (T, r) will be called a *decision rule problem.*

Let α be a real number such that $0 \leq \alpha < 1$. A decision rule

$$f_{i_1} = b_1 \wedge \ldots \wedge f_{i_m} = b_m \rightarrow d \tag{1}$$

is called an *α-decision rule for the pair* (T, r) *and decision* $d \in D(r)$ if attributes $f_{i_1}, \ldots, f_{i_m}$ separate from r at least $(1 - \alpha)|U(T, r, d)|$ rows r' from $U(T, r, d)$

(such rules will be also called *partial decision rules*). The number m is called the *length* of the rule (1). If $U(T, r, d) = \emptyset$ then for any $f_{i_1}, \ldots, f_{i_m} \in \{f_1, \ldots, f_n\}$ the rule (1) is an α-decision rule for (T, r). The rule (1) with empty left-hand side (when $m = 0$) is also an α-decision rule for (T, r).

For example, 0.01-decision rule means that attributes contained in the rule should separate from row r at least 99% of rows from $U(T, r, d)$. If α is equal to 0 we have an exact decision rule (0-decision rule) for (T, r).

We will say that a decision rule is an *α-decision rule for the pair* (T, r) if this rule is an α-decision rule for the pair (T, r) and a decision $d \in D(r)$.

3 Set Cover Problem

In this section, we present the set cover problem as the problem of construction of minimum exact cover (Section 3.1) and minimum partial cover (α-cover) (Section 3.2). We consider bounds on accuracy of a greedy algorithm for the set cover problem.

3.1 Exact Covers

Let A be a set containing $N > 0$ elements, and $F = \{S_1, \ldots, S_p\}$ be a family of subsets of the set A such that $A = \bigcup_{i=1}^{p} S_i$. A subfamily $\{S_{i_1}, \ldots, S_{i_t}\}$ of the family F will be called a *cover* if $\bigcup_{j=1}^{t} S_{i_j} = A$. The problem of searching for a cover with minimum cardinality t is called the *set cover problem*. It is well known that this problem is an NP-hard problem.

U. Feige [2] proved that if $NP \not\subseteq DTIME(n^{O(\log \log n)})$ then for any ε, $0 < \varepsilon < 1$, there is no polynomial algorithm that constructs a cover which cardinality is at most $(1 - \varepsilon)C_{\min} \ln N$, where $C_{\min}$ is the minimum cardinality of a cover.

Now, we present well known *greedy* algorithm for set cover problem.

Set $B := A$, and $COVER := \emptyset$.

(*) In the family F we find a set S_i with minimum index i such that

$$|S_i \cap B| = \max\{|S_j \cap B| : S_j \in F\}.$$

Then we set $B := B \setminus S_i$ and $COVER := COVER \cup \{S_i\}$. If $B = \emptyset$ then we finish the work of the algorithm. The set $COVER$ is the result of the algorithm work. If $B \neq \emptyset$ then we return to the label (*).

We denote by C_{greedy} the cardinality of the cover constructed by greedy algorithm. We will present the following well known result without proof.

Theorem 1. $C_{\text{greedy}} \leq C_{\min} \ln N + 1$.

So we have that if $NP \not\subseteq DTIME(n^{O(\log \log n)})$ then the greedy algorithm is close to the best (from the point of view of accuracy) approximate polynomial algorithms for solving the set cover problem.

3.2 Partial Covers

Let α be a real number such that $0 \leq \alpha < 1$. Let A be a set containing $N > 0$ elements, and $F = \{S_1, \ldots, S_p\}$ be a family of subsets of the set A such that $A = \bigcup_{i=1}^{p} S_i$. A subfamily $\{S_{i_1}, \ldots, S_{i_t}\}$ of the family F will be called an *α-cover for A, F* if $|\bigcup_{j=1}^{t} S_{i_j}| \geq (1-\alpha)|A|$. The problem of searching for an α-cover with minimum cardinality is NP-hard [5].

We consider a greedy algorithm for construction of α-cover. During each step this algorithm chooses a subset from F which covers maximum number of uncovered elements from A. This algorithm stops when the constructed subfamily is an α-cover for A, F. We denote by $C_{\text{greedy}}(\alpha)$ the cardinality of constructed α-cover, and by $C_{\min}(\alpha)$ we denote the minimum cardinality of α-cover for A, F. The following statement was obtained by J. Cheriyan and R. Ravi in [1].

Theorem 2. *Let* $0 < \alpha < 1$. *Then* $C_{\text{greedy}}(\alpha) < C_{\min}(0) \ln(1/\alpha) + 1$.

4 Decision Rules

In this section, we apply the greedy algorithm for set cover problem to construct exact (0-decision rules) and partial decision rules (α-decision rules).

Based on results for the set cover problem we study bounds on accuracy of the greedy algorithm for decision rule construction, and complexity of the problem of minimization of decision rule length.

We can formulate the problem of minimization of decision rule length as follows: for a given decision table T with many-valued decisions, row r of T, decision $d \in D(r)$ and α such that $0 \leq \alpha < 1$, we need to construct an α-decision rule which has the minimum length.

4.1 Exact Decision Rules

We can apply the greedy algorithm for set cover problem to construct decision rules for decision tables with many-valued decisions.

Let T be a table with many-valued decisions containing n columns labeled with attributes $f_1, \ldots, f_n$. Let $r = (b_1, \ldots, b_n)$ be a row of T, $D(r)$ be the set of decisions attached to r and $d \in D(r)$.

We consider a set cover problem $A(T, r, d)$, $F(T, r, d) = \{S_1, \ldots, S_n\}$, where $A(T, r, d) = U(T, r, d)$ is the set of all rows r' of T such that $d \notin D(r')$. For $i = 1, \ldots, n$, the set S_i coincides with the set of all rows from $A(T, r, d)$ which are different from r in the column f_i. One can show that the decision rule

$$f_{i_1} = b_{i_1} \wedge \ldots \wedge f_{i_m} = b_{i_m} \rightarrow d \tag{2}$$

is a 0-decision rule for (T, r) and decision $d \in D(r)$ if and only if the subfamily $\{S_{i_1}, \ldots, S_{i_m}\}$ is a cover for the set cover problem $A(T, r, d)$, $F(T, r, d)$.

We denote by $L_{\min}(T, r, d)$ the minimum length of a 0-decision rule for (T, r) and decision $d \in D(r)$. It is clear that for the constructed set cover problem $C_{\min} = L_{\min}(T, r, d)$.

Let us apply the greedy algorithm to set cover problem $A(T,r,d)$, $F(T,r,d)$. It constructs a cover which corresponds to a 0-decision rule $rule(T,r,d)$: if the greedy algorithm constructs the cover $\{S_{i_1},\ldots,S_{i_m}\}$ then the decision rule $rule(T,r,d)$ coincides with (2). We denote by $L_{\mathrm{greedy}}(T,r,d)$ the length of this rule. By Theorem 1, $L_{\mathrm{greedy}}(T,r,d) \le L_{\min}(T,r,d)\ln|U(T,r,d)|+1$.

We denote by $L_{\mathrm{greedy}}(T,r)$ the length of the rule constructed by the following polynomial algorithm (we will say about this algorithm as about greedy algorithm also). For a given decision table T with many-valued decisions and row r of T, for each $d \in D(r)$ we construct the set cover problem $A(T,r,d)$, $F(T,r,d)$ and then apply to this problem the greedy algorithm. We transform the constructed cover to the 0-decision rule $rule(T,r,d)$. Among the 0-decision rules $rule(T,r,d)$, $d \in D(r)$, we choose a rule with the minimum length. This rule is the output of considered algorithm. We have $L_{\mathrm{greedy}}(T,r) = \min\{L_{\mathrm{greedy}}(T,r,d) : d \in D(r)\}$. It is clear that $L_{\min}(T,r) = \min\{L_{\min}(T,r,d) : d \in D(r)\}$, where $L_{\min}(T,r)$ is the minimum length of 0-decision rule for (T,r). Let $K(T,r) = \max\{|U(T,r,d)| : d \in D(r)\}$. One can show that $L_{\mathrm{greedy}}(T,r) \le L_{\min}(T,r)\ln K(T,r)+1$. So, we have the following statement.

Theorem 3. *Let T be a nondegenerate decision table with many-valued decisions and r be a row of T. Then $L_{\mathrm{greedy}}(T,r) \le L_{\min}(T,r)\ln K(T,r)+1$.*

The next two statements follow immediately from similar ones obtained in [3] for decision tables with one-valued decisions.

Proposition 1. *The problem of minimization of decision rule length for decision tables with many-valued decisions is NP-hard.*

Theorem 4. *If $NP \notin DTIME(n^{O(\log\log n)})$ then for any ε, $0<\varepsilon<1$, there is no polynomial algorithm that for a given nondegenerate decision table T with many-valued decisions and row r of T constructs a 0-decision rule which length is at most $(1-\varepsilon)L_{\min}(T,r)\ln K(T,r)$.*

The comparison of Theorems 3 and 4 shows that under the assumption $NP \notin DTIME(n^{O(\log\log n)})$ the greedy algorithm is close to the best (from the point of view of accuracy) approximate polynomial algorithms for minimization of decision rule length.

4.2 Partial Decision Rules

We use the greedy algorithm for construction of α-covers to construct α-decision rules. Let T be a table with many-valued decisions containing n columns labeled with attributes $f_1,\ldots,f_n$. Let $r=(b_1,\ldots,b_n)$ be a row of T, $D(r)$ be a set of decisions attached to r, $d \in D(r)$, and α be a real number such that $0<\alpha<1$.

We consider a set cover problem $A(T,r,d)$, $F(T,r,d)=\{S_1,\ldots,S_n\}$ where $A(T,r,d)=U(T,r,d)$ is the set of all rows r' of T such that $d \notin D(r')$. For $i=1,\ldots,n$, the set S_i coincides with the set of all rows from $A(T,r,d)$ which are different from r in the column f_i. One can show that the decision rule

$$f_{i_1}=b_{i_1}\wedge\ldots\wedge f_{i_m}=b_{i_m}\rightarrow d$$

is an α-decision rule for (T, r) and decision $d \in D(r)$ if and only if $\{S_{i_1}, \ldots, S_{i_m}\}$ is an α-cover for the set cover problem $A(T, r, d)$, $F(T, r, d)$. Evidently, for the considered set cover problem $C_{\min}(0) = L_{\min}(T, r, d)$, where $L_{\min}(T, r, d)$ is the minimum length of 0-decision rule for (T, r) and decision $d \in D(r)$.

Let us apply the greedy algorithm to the considered set cover problem. This algorithm constructs an α-cover which corresponds to an α-decision rule $rule(\alpha, T, r, d)$ for decision table T, row r and decision $d \in D(r)$. From Theorem 2 it follows that the length of this rule is at most $L_{\min}(T, r, d) \ln(1/\alpha) + 1$.

We denote by $L_{\text{greedy}}(\alpha, T, r)$ the length of the rule constructed by the following polynomial algorithm: for a given α, $0 < \alpha < 1$, decision table T, row r of T and decision $d \in D(r)$, we construct the set cover problem $A(T, r, d)$, $F(T, r, d)$ and then apply to this problem the greedy algorithm for construction of α-cover. We transform the obtained α-cover into an α-decision rule $rule(\alpha, T, r, d)$. Among the α-decision rules $rule(\alpha, T, r, d)$, $d \in D(r)$, we choose a rule with the minimum length. This rule is the output of considered algorithm. We denote by $L_{\min}(\alpha, T, r)$ the minimum length of α-decision rule for (T, r). According to what has been said above we have the following statement.

Theorem 5. *Let T be a nondegenerate decision table with many-valued decisions, r be a row of T, and α be a real number such that $0 < \alpha < 1$. Then*

$$L_{\text{greedy}}(\alpha, T, r) \leq L_{\min}(\alpha, T, r) \ln(1/\alpha) + 1.$$

Based on results from [5] it is not difficult to prove the following statement.

Proposition 2. *For any α, $0 \leq \alpha < 1$, the problem of minimization of α-decision rule length for decision tables with many-valued decisions is NP-hard.*

5 Conclusions

We presented the greedy algorithm for construction of exact and partial decision rules for decision tables with many-valued decisions. We studied binary decision tables with many-valued decisions but the considered approach can be used also for decision tables with more than two values of attributes.

References

1. Cheriyan, J., Ravi, R.: Lecture notes on approximation algorithms for network problems (1998), `http://www.math.uwaterloo.ca/jcheriya/lecnotes.html`
2. Feige, U.: A threshold of $\ln n$ for approximating set cover (Preliminary version). In: Proc. 28th Annual ACM Symposium on the Theory of Computing, pp. 314–318 (1996)
3. Moshkov, M., Piliszczuk, M., Zielosko, B.: Partial Covers, Reducts and Decision Rules in Rough Sets: Theory and Applications. SCI, vol. 145. Springer, Heidelberg (2008)
4. Pawlak, Z.: Rough Sets—Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)
5. Ślęzak, D.: Approximate entropy reducts. Fundam. Inform. 53, 365–390 (2002)