Optimization of Inhibitory Rules

Azimkhon Ostonov[1[0000−0001−5763−9751]], Beata Zielosko[1,2[0000−0003−3788−1094]], Kamil Jabłoński[2[0009−0005−2817−5341]], Kamil Magiera[2[0009−0002−1351−2112]], and Mikhail Moshkov[1[0000−0003−0085−9483]]

[1] Computer, Electrical and Mathematical Sciences & Engineering Division
King Abdullah University of Science and Technology (KAUST)
Thuwal 23955-6900, Saudi Arabia
{azimkhon.ostonov,mikhail.moshkov}@kaust.edu.sa
[2] Institute of Computer Science
University of Silesia in Katowice
Bedzinska 39, Sosnowiec, 41-200, Poland
{beata.zielosko,kamil.jablonski}@us.edu.pl, kamllos@interia.pl

Abstract. Representing of knowledge in the form of rules is one of the most popular methods due to its intuitiveness and transparency. Shorter rules are easier to understand and interpret. There are various types of rules, e.g., decision rules, action rules, probabilistic rules, nondeterministic rules, and many others. These rules differ in their approach to decision making and the method of their induction.

This paper focuses on the exploration of inhibitory rules, which are defined by the expression "attribute $\neq$ decision" on the right-hand side. In certain cases, such rules can convey more information about datasets than conventional decision rules. It is known that the problem of constructing inhibitory rules with minimum length is NP-hard. Therefore, various approaches are used to obtain approximate rules. In this work, three novel algorithms for inducing inhibitory rules and systems of such rules are studied. In particular, it was shown that, under the assumption $P \neq NP$, the m-greedy algorithm achieves an approximation ratio that is close to the best possible achievable by any polynomial-time algorithm. Taking into account perspective of knowledge representation, we analyze experimentally effectiveness of proposed algorithms, particularly regarding the minimization of rule length.

Keywords: Inhibitory Rules · Greedy Algorithms · Length · Minimization · Knowledge Representation.

1 Introduction

One of the commonly used methods of knowledge representation is rule-based representation. Knowledge is most often presented in the form of decision rules "if <conditions> then <conclusion>", where the conditional part specifies the premises, while the decision part contains the conclusion or action to be taken once these premises are met [19, 21, 23]. This method of knowledge representation

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

allows for a transparent mapping of expert knowledge, as it not only enables decisions to be made, but also justifies them in a way that is understandable to the user.

Rule induction can be carried out from different perspectives, for example, classification and knowledge representation [15, 21, 24, 26]. In the first case, the goal is to assign a class label to an unknown object based on its characteristics and using a set of induced rules. In the case of knowledge representation, rules are used to discover and represent hidden patterns in the data. From this perspective, rules that cover a wider range of examples while remaining simple and understandable are particularly valuable. The length of rules is therefore important, also from the point of view of the Minimal Description Length principle [20]: "that the best hypothesis for a given set of data is the one that leads to the largest compression of the data".

Unlike standard decision rules, which use the format "attribute = decision" on the right-hand side, inhibitory rules are expressed as "attribute ≠ decision", or more simply, "≠ decision".

Interest in inhibitory rules was sparked when Skowron and Suraj [22] (see also [25]) demonstrated a data set example where inhibitory rules could capture more information than decision rules. Later work confirmed that, for any given data set, inhibitory rules can express a form of complete information—refer to [16] for further details

Subsequent research addressed algorithmic methods for generating inhibitory rules, analyzing their minimal complexity, constructing sets of all minimal inhibitory rules, and developing both standard and lazy classifiers based on such rules [8, 9, 16, 18]. These classifiers showed superior performance over those based on decision rules, with findings consolidated in [10]. The next research phase focused on enhancements to dynamic programming algorithms, incorporating multi-stage and bi-criteria optimization approaches targeting different parameters of inhibitory rules (such as length and coverage), as well as their systems (considering cost and uncertainty aspects) [2–6]. These findings were summarized in [1].

Additionally, our earlier work on discovering decision rules consistent with as many given decision trees as possible [27] revealed that this method could be extended to inhibitory rules—a topic reserved for future investigation into generalized knowledge extraction from distributed sources.

A renewed motivation to continue this line of study emerged from the development of novel approximation algorithms for the set cover problem [7], which we found applicable to inhibitory rule construction. This is one of the central topics of this paper. Since the problem of constructing inhibitory rules with minimum length is NP-hard [10], three greedy algorithms for inducing inhibitory rules and systems of such rules are proposed and compared in terms of length of constructed rules. In particular, we show that, assuming $P \neq NP$, the m-greedy algorithm achieves an approximation ratio close to the best possible for any polynomial-time algorithm.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

This paper is organized into six sections. Section 2 introduces foundational concepts. Consider a decision table T and a row r labeled with decision d. Let $D(T)$ represent all decisions in T, and define $D(T,r) = D(T) \setminus \{d\}$. For any $t \in D(T,r)$, an inhibitory rule for T, r, and t is a rule that holds true in T, is realizable for r (i.e., its left-hand side holds for r), and has $\neq t$ as its right-hand side.

A finite set of inhibitory rules S is said to be *complete* for T if every rule in S is valid for T, and for each row r and some $t \in D(T,r)$, S includes an inhibitory rule for T, r, and t. The set S is *full* for T if, for every row r and every $t \in D(T,r)$, such a rule exists in S. The *depth* of a system S is defined as the length of its longest rule.

Our objective is to minimize the depth of complete and full systems of inhibitory rules, as shorter rules are generally more efficient to compute and interpret.

Section 3 discusses three approximation algorithms addressing the set cover problem. The task is to find a minimal subset family that covers a given finite set. The *extended greedy algorithm* introduces an expanded search strategy by examining not just one but several subcovers at each step, which can lead to cover reduction. The *modified greedy algorithm* builds on the classic greedy approach by including a refinement phase to eliminate unnecessary subsets from the cover. We present two results assessing its performance. The third, the *"easy" algorithm*, iteratively removes as many subsets as possible from the original set family while preserving coverage.

In Sect. 4, we present algorithms for generating complete and full inhibitory rule systems, utilizing the approximation methods for the set cover problem. A polynomial-time reduction is provided, transforming the inhibitory rule length minimization problem into a set cover problem. The extended greedy, modified greedy, and "easy" algorithms are applied to the transformed instances, and the resulting covers are converted into inhibitory rules. By aggregating these rules, we obtain six algorithms for constructing complete or full inhibitory rule systems. For the version using the modified-greedy algorithm for full systems, we include two statements characterizing its quality.

Section 5 reports on computational experiments with the algorithms from Sect. 4. These tests evaluate the effectiveness of the algorithms in approximating the minimal depth of complete inhibitory rule systems. Statistical analysis of the obtained results is also provided.

Section 6 concludes the paper with a brief summary.

The core contribution of this work lies in the introduction of new algorithms for constructing complete and full inhibitory rule systems (Sect. 4), and theoretical results showing that, under the assumption $P \neq NP$, the m-greedy algorithm achieves an approximation ratio that is close to the best polynomial-time algorithm. The proposed algorithms were compared experimentally, taking into account knowledge representation perspective.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

2 Basic Concepts

Let ω denote the set of nonnegative integers $\{0, 1, 2, \ldots\}$, and define $A = \{a_i : i \in \omega\}$ as the set of attributes. Two attributes a_i and a_j are considered distinct if $i \neq j$.

A *decision table* T is a rectangular array with entries from the set ω. The columns of T are labeled by distinct attributes from A, and the rows are pairwise distinct, each assigned a decision from ω. We write $D(T)$ for the set of decisions assigned to the rows of T. A decision table may have no rows or columns. A table T is called *nondegenerate* if $|D(T)| \geq 2$, and *degenerate* otherwise. For a nondegenerate table T and a row r with decision d, we define $D(T, r) = D(T) \setminus \{d\}$.

Suppose T is a nondegenerate decision table with columns labeled by the attributes $a_{i_1}, \ldots, a_{i_n}$. An *inhibitory rule over* T is an expression of the form

$$(a_{j_1} = b_1) \wedge \cdots \wedge (a_{j_m} = b_m) \to\neq t, \quad (1)$$

where $a_{j_1}, \ldots, a_{j_m}$ are pairwise distinct attributes selected from $\{a_{i_1}, \ldots, a_{i_n}\}$, $b_1, \ldots, b_m \in \omega$, and $t \in D(T)$. The integer m is called the *length* of rule (1). This rule is said to be *realizable* for a row $r = (\delta_1, \ldots, \delta_n)$ of T if $\delta_{j_1} = b_1, \ldots, \delta_{j_m} = b_m$. Rule (1) is said to be *true* for T if, for every row r of T for which it is realizable, the decision of r is different from t. The *coverage* of the true for T rule (1) is the percentage of rows of T for which this rule is realizable.

Two inhibitory rules are called *equal* if they have the same sets of conditions on their left-hand sides and equal right-hand sides.

Given a row r of T and a decision $t \in D(T, r)$, rule (1) is called an *inhibitory rule for* T*,* r*, and* t if it is true for T and realizable for r. Let $l(T, r, t)$ denote the minimum length of an inhibitory rule for T, r, and t.

A *complete system of inhibitory rules for* T is a finite set S of inhibitory rules over T such that each rule in S is true for T, and for each row r of T, there exists at least one rule in S that is realizable for r; that is, for every row r and some $t \in D(T, r)$, S contains an inhibitory rule for T, r, and t.

A *full system of inhibitory rules for* T is a finite set S of inhibitory rules over T such that each rule in S is true for T, and for each row r of T and each $t \in D(T, r)$, S contains a rule that is realizable for r and has $\neq t$ on the right-hand side; that is, for every row r and each $t \in D(T, r)$, S contains an inhibitory rule for T, r, and t.

Let ρ be an inhibitory rule and S a system of inhibitory rules. The length of ρ is denoted by $l(\rho)$. The *depth* of S—the maximum rule length in S—is denoted by $L(S)$. For a nondegenerate decision table T, define $L_i^c(T)$ as the minimum depth of a complete inhibitory rule system for T, and $L_i^f(T)$ as the minimum depth of a full inhibitory rule system for T. Let $Row(T)$ denote the set of rows in T, and for any row r, define $l(T, r) = \min\{l(T, r, t) : t \in D(T, r)\}$. Then $L_i^c(T) = \max\{l(T, r) : r \in Row(T)\}$ and $L_i^f(T) = \max\{l(T, r, t) : r \in Row(T), t \in D(T, r)\}$. For a degenerate table T, we define $L_i^c(T) = L_i^f(T) = 0$. Clearly, for any decision table T, the inequality $L_i^c(T) \leq L_i^f(T)$ holds.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

3 Approximate Algorithms for the Set Cover Problem

In this section, we consider three approximate algorithms for the set cover problem. Let $P = \{p_1, \dots, p_n\}$ be a finite set with $n > 0$ elements, and let $F = \{S_1, \dots, S_m\}$ be a family of subsets of P such that $P = \bigcup_{i=1}^{m} S_i$. A subfamily $\{S_{i_1}, \dots, S_{i_t}\} \subseteq F$ is called a *cover* if $\bigcup_{j=1}^{t} S_{i_j} = P$. The problem of finding a cover of minimum cardinality t is known as the set cover problem, denoted by (P, F). We say that a subset $S_i \in F$ *covers* all elements of P contained in S_i.

3.1 Extended Greedy Algorithm

The *extended greedy algorithm* (or *e-greedy algorithm*) proceeds in three phases. Below we describe its operation on the set cover problem (P, F).

Phase 1.

- *Step 1:* Remove duplicate subsets from F, keeping only one copy of each. For any pair $S_i, S_j \in F$ such that $S_i \subset S_j$, remove S_i from F. Denote the resulting family by F_0.
- *Step 2:* Initialize $B_0 := \emptyset$. For each $S_i \in F_0$, if there exists an element of P covered *only* by S_i in F_0, add S_i to B_0, remove S_i from F_0, and remove from P and from all sets in F_0 the elements covered by S_i. This yields a reduced set cover problem (P', F').
  If $P' = \emptyset$, return B_0 as a cover for (P, F) and terminate. Otherwise, proceed to Phase 2.
  For $S_i \in F$, denote by S'_i the subset obtained from S_i by removing elements already covered by subsets in B_0.

Phase 2.

- *Step 1:* Initialize $\mathcal{C} = \{\emptyset\}$ (holding partial covers) and $\mathcal{D} = \emptyset$ (holding full covers) for (P', F').
- *Step 2:* Suppose we have taken $i \geq 1$ steps. Then:
  - If $\mathcal{C} = \emptyset$ or $|\mathcal{D}| \geq 16$, move to Phase 3.
  - If $|\mathcal{C}| \geq 128$, sort all subcovers in $\mathcal{C}$ by the number of elements of P' covered (descending) and keep only the top 16.
  - For each $B \in \mathcal{C}$:
    - ∗ If B is a cover of (P', F'), move B from $\mathcal{C}$ to $\mathcal{D}$.
    - ∗ Otherwise, let P'' be the set of elements of P' not covered by B. Let $S'_{i_1}, \dots, S'_{i_k}$ be all subsets in F' that cover at least one element of P''. Order these by the size of their intersection with P'' in descending order as $S'_{j_1}, \dots, S'_{j_k}$.
      Remove B from $\mathcal{C}$ and add the three subcovers:
      $$B \cup \{S'_{j_1}\}, \quad B \cup \{S'_{j_{\lceil k/2 \rceil}}\}, \quad B \cup \{S'_{j_k}\}$$
  - Increment i and repeat Step 2.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

Phase 3.

- *Step 1:* For each cover $B = \{S'_{l_1}, \ldots, S'_{l_t}\} \in \mathcal{D}$, try to reduce it to a smaller cover B' as follows:
  Set $j := 1$.
  While $j \leq t$, do:
  $$\text{If } B \setminus \{S'_{l_j}\} \text{ is a cover of } (P', F'), \text{ then } B := B \setminus \{S'_{l_j}\}.$$
  Otherwise, leave B unchanged.
  Increment $j := j + 1$. Return B as B' once all elements are checked.
- *Step 2:* Choose $B \in \mathcal{D}$ such that the reduced cover B' has minimum cardinality. Return
  $$B_0 \cup \{S_{q_1}, \ldots, S_{q_v}\}$$
  as the cover for the original problem (P, F), where $B' = \{S'_{q_1}, \ldots, S'_{q_v}\}$.

Remark 1. The extended greedy algorithm returns a cover for (P, F). However, there are no guaranteed approximation bounds for its performance.

3.2 Modified Greedy Algorithm

The *modified greedy algorithm* (or *m-greedy algorithm*) consists of two phases. It operates as follows for the set cover problem (P, F).

Phase 1.

- *Step 1:* Initialize $B := \emptyset$ and $U := P$.
- *Step 2:* While B is not a cover of (P, F):
  - Choose $S_j \in F$ such that $|S_j \cap U|$ is maximal.
  - Update $B := B \cup \{S_j\}$ and $U := U \setminus S_j$.

  Once B covers P, proceed to Phase 2.

Phase 2.

- Let $B = \{S_{i_1}, \ldots, S_{i_t}\}$. Set $j := 1$.
- While $j \leq t$:
  - If $B \setminus \{S_{i_j}\}$ is still a cover of (P, F), then remove S_{i_j} from B.
  - Increment $j := j + 1$.
- Return B as a cover for (P, F).

Remark 2. The modified greedy algorithm returns a cover for (P, F).

The next statement follows from Theorem 4.1 of [17], which studied the classical greedy algorithm that does not include Phase 2. Clearly, the size of the cover constructed by the m-greedy algorithm does not exceed the size of the cover constructed by the classical greedy algorithm.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

Proposition 1. *Let (P, F) be a set cover problem. Then*

$$C_{m\text{-}greedy}(P, F) \le C_{\min}(P, F) \cdot \ln |P| + 1,$$

where $C_{m\text{-}greedy}(P, F)$ is the cardinality of the cover returned by the modified greedy algorithm, and $C_{\min}(P, F)$ is the cardinality of an optimal cover.

The following result shows that, under the widely believed assumption $P \neq NP$, the m-greedy algorithm achieves an approximation ratio that is close to the best possible achievable by any polynomial-time algorithm.

Proposition 2. *[11] If $P \neq NP$, then for any ε with $0 < \varepsilon < 1$, there is no polynomial-time algorithm that, for every set cover problem (P, F), constructs a cover whose cardinality is at most*

$$(1 - \varepsilon) \cdot C_{\min}(P, F) \cdot \ln |P|.$$

3.3 "Easy" Algorithm

We now describe the *"easy" algorithm* for the set cover problem (P, F), where $F = \{S_1, \ldots, S_m\}$.

Step 1. Set $B := \{S_1, \ldots, S_m\}$ and $j := 1$. Proceed to Step 2.

Step 2. If $j > m$, return B as a cover for (P, F).

Otherwise, if $B \setminus \{S_j\}$ is still a cover for (P, F), then update $B := B \setminus \{S_j\}$ and increment $j := j + 1$. Return to Step 2.

If $B \setminus \{S_j\}$ is not a cover, then keep B unchanged, increment $j := j + 1$, and return to Step 2.

Remark 3. This algorithm returns a cover for (P, F). However, there are no known approximation guarantees for its performance.

4 Algorithms for Constructing Systems of Inhibitory Rules

This section describes algorithms for constructing complete and full systems of inhibitory rules based on approximate algorithms for solving the set cover problem.

Let T be a decision table with n columns labeled by attributes $a_1, \ldots, a_n$. Let $r = (\delta_1, \ldots, \delta_n)$ be a row of T and $t \in D(T, r)$. We consider the set cover problem $(P(T, r, t), F(T, r, t))$, where $P(T, r, t) = \{$all rows of T with decision $t\}$, and

$$F(T, r, t) = \{S_1, \ldots, S_n\}.$$

For each $i = 1, \ldots, n$, the set S_i consists of all rows from $P(T, r, t)$ which differ from the row r in the attribute a_i.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

One can show that the rule $(a_{i_1} = \delta_{i_1}) \wedge \cdots \wedge (a_{i_p} = \delta_{i_p}) \to \neq t$, where $i_1, \ldots, i_p \in \{1, \ldots, n\}$, is a rule for T, r, and t if and only if the subfamily $\{S_{i_1}, \ldots, S_{i_p}\}$ is a cover for the set cover problem $(P(T, r, t), F(T, r, t))$.

To approximately solve the problem of length minimization for inhibitory rules, the following approach is used: for a given nondegenerate decision table T, a row r, and a decision $t \in D(T, r)$, we construct the set cover problem $(P(T, r, t), F(T, r, t))$, apply one of the approximate cover construction algorithms, and transform the obtained cover into an inhibitory rule for T, r, and t.

Let al be an approximate algorithm for the set cover problem, where $al \in \{e\text{-}greedy, m\text{-}greedy, easy\}$. We now describe an algorithm $\mathcal{A}^c_{al}$ that, for a given nondegenerate decision table T, constructs a complete system of inhibitory rules for T, denoted by $\mathcal{A}^c_{al}(T)$.

For each row r of T and each decision $t \in D(T, r)$, we construct an inhibitory rule for T, r and t denoted $rule(T, r, t)$ using algorithm al. Then, for each row r, among the rules $\{rule(T, r, t) : t \in D(T, r)\}$, we select a rule of minimum length and denote it by $rule(T, r)$. Denoting by $Row(T)$ the set of rows of T, we define $\mathcal{A}^c_{al}(T) = \{rule(T, r) : r \in Row(T)\}$. For different rows r_1 and r_2 of the table T, rules $rule(T, r_1)$ and $rule(T, r_2)$ may be equal. We will store these two equal rules in the set under consideration.

Similarly, we describe an algorithm $\mathcal{A}^f_{al}$ that constructs a full system of inhibitory rules for T, denoted by $\mathcal{A}^f_{al}(T)$. For each row r of T and each decision $t \in D(T, r)$, we construct the inhibitory rule $rule(T, r, t)$ using algorithm al. Then, $\mathcal{A}^f_{al}(T) = \{rule(T, r, t) : r \in Row(T), t \in D(T, r)\}$. For different rows r_1 and r_2 of the table T and decisions $t_1 \in D(T, r_1)$ and $t_2 \in D(T, r_2)$, rules $rule(T, r_1, t_1)$ and $rule(T, r_2, t_2)$ may be equal. We will store these two equal rules in the set under consideration.

We now obtain a bound on the accuracy of the algorithm $\mathcal{A}^f_{m\text{-}greedy}$. Let T be a nondegenerate decision table. For any $t \in D(T)$, let $N_t(T)$ denote the number of rows of T labeled with decision t, and define $K(T) = \max_{t \in D(T)} N_t(T)$. Recall that $L^f_i(T)$ denotes the minimum depth of a full system of inhibitory rules for T.

Theorem 1. *Let T be a nondegenerate decision table. Then*

$$L(\mathcal{A}^f_{m\text{-}greedy}(T)) \leq L^f_i(T) \ln K(T) + 1.$$

Proof. The algorithm $\mathcal{A}^f_{m\text{-}greedy}$ operates on T as follows: for any row r of T and decision $t \in D(T, r)$, it constructs the set cover problem $(P(T, r, t), F(T, r, t))$, applies the m-greedy algorithm to construct a cover, and transforms the obtained cover into an inhibitory rule $rule(T, r, t)$ for T, r, and t. Then, $\mathcal{A}^f_{m\text{-}greedy}(T) = \{rule(T, r, t) : r \in Row(T), t \in D(T, r)\}$.

By Proposition 1, the length of $rule(T, r, t)$ is at most $l(T, r, t) \ln N_t(T) + 1$. Since $l(T, r, t) \leq L^f_i(T)$ and $N_t(T) \leq K(T)$, it follows that $L(\mathcal{A}^f_{m\text{-}greedy}(T)) \leq L^f_i(T) \ln K(T) + 1$. □

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

The following theorem shows that, under the assumption $P \neq NP$, the algorithm $\mathcal{A}^f_{m\text{-}greedy}$ is close to the best achievable polynomial-time approximation algorithm for the problem of depth minimization for full systems of inhibitory rules.

Theorem 2. *If $P \neq NP$, then for any ε, $0 < \varepsilon < 1$, there is no polynomial-time algorithm that, for each nondegenerate decision table T, constructs a full system of inhibitory rules with depth at most*

$$(1-\varepsilon)L_i^f(T)\ln K(T).$$

Proof. Let (P, F) be a set cover problem with

$$P = \{p_1, \ldots, p_n\} \quad \text{and} \quad F = \{S_1, \ldots, S_m\}.$$

Consider the decision table $T(P, F)$ defined as follows. The table $T(P, F)$ has m columns labeled by attributes $a_1, \ldots, a_m$ corresponding to the sets $S_1, \ldots, S_m$, respectively, and $n+1$ rows. For $j = 1, \ldots, n$, the jth row corresponds to the element p_j, and the $(n+1)$th row is a special additional row filled with zeros. The entries of the table are defined so that for $j = 1, \ldots, n$ and $i = 1, \ldots, m$, the cell in the jth row and ith column is 1 if and only if $p_j \in S_i$; otherwise, it is 0. The decision attached to the last $(n+1)$th row is 2, and all other rows are labeled with decision 1.

We first show that $L_i^f(T(P, F)) = C_{\min}(P, F)$. For $j = 1, \ldots, n+1$, denote by r_j the j-th row of the table $T(P, F)$. It is clear that $D(T(P, F)) = \{1, 2\}$. One can prove that $L_i^f(T(P, F)) = \max\{l(T(P, F), r_{n+1}, 1),\ l(T(P, F), r_j, 2) : j = 1, \ldots, n\}$. It is easy to see that $l(T(P, F), r_j, 2) = 1$ for all $j = 1, \ldots, n$.

Furthermore, a subfamily $\{S_{i_1}, \ldots, S_{i_t}\} \subseteq F$ is a cover for (P, F) if and only if the rule $(a_{i_1} = 0) \wedge \cdots \wedge (a_{i_t} = 0) \rightarrow \neq 1$ is an inhibitory rule for $T(P, F)$, r_{n+1}, and decision 1. Therefore, $l(T(P, F), r_{n+1}, 1) = C_{\min}(P, F)$, and hence $L_i^f(T(P, F)) = C_{\min}(P, F)$.

Now, assume for contradiction that $P \neq NP$ but there exists some ε, $0 < \varepsilon < 1$, and a polynomial-time algorithm that, for every nondegenerate decision table T, constructs a full system of inhibitory rules whose depth is at most $(1-\varepsilon)L_i^f(T)\ln K(T)$.

Apply this algorithm to the table $T(P, F)$. We obtain a full system of inhibitory rules S for $T(P, F)$ such that $L(S) \leq (1-\varepsilon)L_i^f(T(P, F))\ln K(T(P, F))$.

This system contains an inhibitory rule for $T(P, F)$, r_{n+1}, and decision 1 of the form $(a_{i_1} = 0) \wedge \cdots \wedge (a_{i_t} = 0) \rightarrow \neq 1$.

As previously mentioned, $\{S_{i_1}, \ldots, S_{i_t}\}$ is a cover for (P, F) whose cardinality is at most $L(S)$.

Note that $K(T(P, F)) = |P|$, and from above, $L_i^f(T(P, F)) = C_{\min}(P, F)$.

Therefore, for an arbitrary set cover problem (P, F), we can construct in polynomial time a cover of cardinality at most

$$(1-\varepsilon)C_{\min}(P, F)\ln |P|.$$

But this contradicts Proposition 2, which states that such an approximation is impossible if $P \neq NP$. Hence, no such polynomial-time algorithm exists. □

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

5 Experimental Study of Algorithms for Constructing Complete Systems of Inhibitory Rules

This section presents results of computational experiments conducted to evaluate the accuracy of the algorithms described in Sect. 4 for constructing complete inhibitory rule systems. The main goal is to assess how well these algorithms approximate the minimum depth of complete inhibitory rule systems.

For selected decision tables T from the UCI Machine Learning Repository [14], we construct complete inhibitory rule systems $\mathcal{A}^c_{e\text{-}greedy}(T)$, $\mathcal{A}^c_{m\text{-}greedy}(T)$, and $\mathcal{A}^c_{easy}(T)$ using the algorithms introduced in Sect. 4. We then compare their depths with each other.

We present experimental results on 20 decision tables from the UCI Machine Learning Repository [14]. Some decision tables contain conditional attributes that take a unique value in each row; such attributes are excluded. If the table contains duplicate rows (possibly with differing decisions), each group of identical rows is replaced with a representative row assigned the most frequent decision in that group. For tables with missing attribute values, these are replaced with the most frequent value for the corresponding attribute.

Table 1 summarizes the results of experiments. The column "Decision table" lists the dataset names from [14]. The columns "Rows" and "Attrs" indicate the number of rows and conditional attributes, respectively, in the considered decision table T, column |D(T)| denotes the cardinality of decision classes. The columns "E-greedy", "M-greedy", and "Easy" contain information about complete inhibitory rule systems $\mathcal{A}^c_{e\text{-}greedy}(T)$, $\mathcal{A}^c_{m\text{-}greedy}(T)$, and $\mathcal{A}^c_{easy}(T)$, constructed by the algorithms $\mathcal{A}^c_{e\text{-}greedy}$, $\mathcal{A}^c_{m\text{-}greedy}$, and $\mathcal{A}^c_{easy}$ for the decision table T. The minimum, average, and the maximum lengths of rules from the constructed systems over all rows of T are shown in subcolumns "min", "avg", and "max", respectively. The "max" subcolumn in each case corresponds to the depth of the rule system: $L(\mathcal{A}^c_{e\text{-}greedy}(T))$, $L(\mathcal{A}^c_{m\text{-}greedy}(T))$, and $L(\mathcal{A}^c_{easy}(T))$. The last row "average" presents average values among all datasets.

Based on the presented results, it can be seen that the minimum lengths of rules constructed by considered greedy algorithms are comparable, and for many of the presented datasets, the minimum rule length is equal to 1. In terms of average rule lengths, the shortest rules are constructed by the e-greedy algorithm, followed by m-greedy, with the easy algorithm ranking third. Considering the maximum length of rules in relation to the number of attributes in the dataset, only for 2 of the 20 datasets studied, i.e., balance-scale and monks-2-test, the number of attributes in the dataset is the same as the maximum length of rules. This situation occurs for all 3 algorithms. It should also be noted that for a mushroom dataset containing 22 attributes, the maximum rule length for the e-greedy algorithm is 2, and for m-greedy and easy it is equal to 3. Moreover, in the lymhopgraphy dataset, which contains 18 attributes, the maximum rule length is 1 across all 3 algorithms.

Figure 1 presents the distribution of values related to the maximum and average length of rules for the tested datasets, for the three proposed algorithms, and shows the differences among them.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

Table 1. Results of experiments with complete systems of rules

Decision table	Rows	Attr	|D(T)|	Easy			E-greedy			M-greedy		
				min	avg	max	min	avg	max	min	avg	max
adult-stretch	16	4	2	1	1.250	2	1	1.250	2	1	1.250	2
balance-scale	625	4	3	2	2.725	4	2	2.672	4	2	2.672	4
breast-cancer	266	9	2	1	3.308	6	1	2.665	6	1	2.703	6
cars	1728	6	4	1	1.069	4	1	1.047	3	1	1.047	3
house-votes	279	16	2	2	3.416	7	2	2.538	5	2	2.559	5
lymphography	148	18	4	1	1.000	1	1	1.000	1	1	1.000	1
monks-1-test	432	6	1	1	2.250	3	1	2.250	3	1	2.250	3
monks-1-train	124	6	2	1	2.952	5	1	2.266	3	1	2.274	3
monks-2-test	432	6	2	3	4.523	6	3	4.523	6	3	4.523	6
monks-2-train	169	6	2	3	3.633	5	3	3.497	5	3	3.527	5
monks-3-test	432	6	2	1	1.750	2	1	1.750	2	1	1.750	2
monks-3-train	122	6	2	2	2.582	4	2	2.311	4	2	2.328	4
mushroom	8124	22	2	1	1.840	3	1	1.182	2	1	1.182	3
nursery	12960	8	5	1	1.000	1	1	1.000	1	1	1.000	1
shuttle-landing	15	6	2	1	1.467	4	1	1.400	4	1	1.400	4
soybean-small	47	35	4	1	1.000	1	1	1.000	1	1	1.000	1
spect-test	169	22	2	1	1.964	9	1	1.485	8	1	1.509	8
teeth	23	8	23	1	1.000	1	1	1.000	1	1	1.000	1
tic-tac-toe	958	9	2	3	4.014	6	3	3.031	4	3	3.602	5
zoo-data	59	16	7	1	1.000	1	1	1.000	1	1	1.000	1
average				1.45	2.19	3.75	1.45	1.94	3.30	1.45	1.98	3.40

Considering the maximum rule length (the left-hand side of the figure), the Easy algorithm has the highest median value of the three algorithms tested, which is 4.0, while the average value is 3.75. Its interquartile range is from 1.75 to 5.25, with a maximum value of 9 that refers to the spect-test dataset containing 22 attributes. The e-greedy and m-greedy algorithms have lower median values, i.e., 3.0 in both cases. The interquartile range for the e-greedy algorithm (1.75–4.25) is slightly narrower than for the m-greedy algorithm (1.75–5.00), which indicates greater homogeneity in terms of the maximum length of rules. For both algorithms, the maximum observed value is 8 referred to spect-test dataset.

In the case of the right-hand of the figure showing the average length of rules, the easy algorithm has a higher median of 1.902 and greater variability in rule length compared to the other two algorithms. Its interquartile range is between 1.0 to 3.041, which is greater than that of the e-greedy and m-greedy algorithms. The e-greedy algorithm shows the lowest median of 1.617, while for the m-greedy algorithm, the value is 1.629. All three algorithms have the same extreme values, ranging from 1 to 4.523. In general, the proposed algorithms allow the construction of short inhibitory rules, although there are also isolated cases that deviate from this trend.

To examine whether the proposed algorithms produce statistically different results, the Friedman test was performed [12], and the average rank for each

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

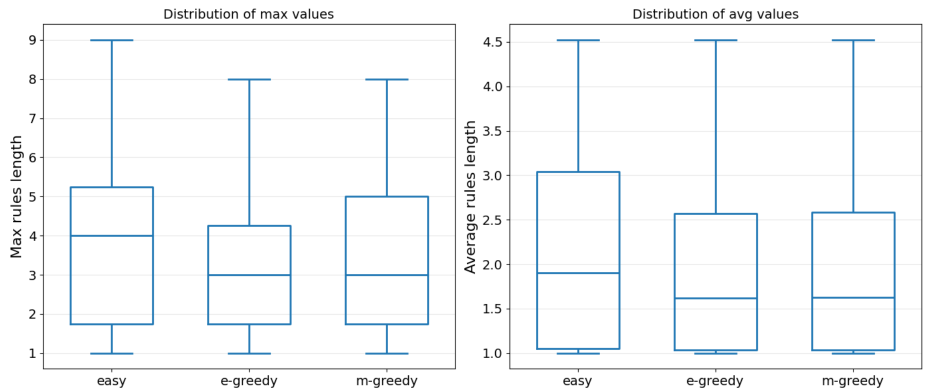

Fig. 1. Distribution of values related to the length of inhibitory rules: the maximum values (the left-hand side) and the average values (the right-hand side). Each subfigure depicting a different range of values (maximum and average length).

algorithm was computed across all datasets: e-greedy — 1.55, m-greedy — 1.90, easy — 2.55. To identify which specific pairs of algorithms differ significantly, the Nemenyi test was performed and p-values for each combination of pairs of algorithms were calculated [13]. The Nemenyi test confirms the significant difference between easy and e-greedy ($p = 0.004$). The difference between the easy and m-greedy approaches significance $p = 0.099$, while e-greedy and m-greedy show no significant difference ($p = 0.510$). This suggests that e-greedy and m-greedy algorithms are relatively similar in their behavior, both outperforming the easy algorithm in generating shorter rules.

Figure 2 presents the CD diagram based on the Nemenyi test results. Algorithms connected by a horizontal bar do not differ significantly at the $\alpha = 0.05$ level.

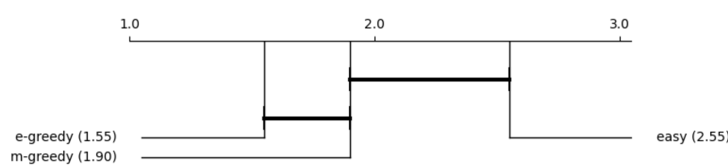

Fig. 2. Critical difference diagram showing pairwise comparisons of algorithms based on average rule length.

This visual representation confirms that while e-greedy achieves the best rank, its performance is statistically indistinguishable from m-greedy, whereas both outperform the easy algorithm.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

6 Conclusions

The representation of knowledge in the form of rules plays an important role, especially in the era of intensive development of artificial intelligence methods. The transparency and clarity of the proposed decisions and the simplicity and intuitive form of rule notation are the strengths of rule-based representation.

There are various types of rules, among which the most popular are decision rules written in the form of logical implications with a conclusion part in the form: "*attribute* = *decision*". In this work, the authors focused on inhibitory rules, which in the decision part take the form of "*attribute* ≠ *decision*". The factors determining the investigation of these rules were that they are able to represent more information encoded in the data than decision rules. In addition, classifiers constructed on the basis of inhibitory rules often exhibit lower classification error compared to classifiers derived from decision rules.

Taking into account knowledge representation perspective, the length of rules plays a significant role. Shorter rules make it easier for people to understand and increase the transparency of the model. Unfortunately, the problem of construction inhibitory rules with minimum depth is NP-hard. Therefore, approximate and heuristic methods that allow a solution to be obtained within a reasonable time are desirable.

In this paper, authors proposed three new algorithms for constructing inhibitory rules based on approximate polynomial-time algorithms for the set cover problem. The quality of these algorithms was analyzed. It was shown that the proposed m-greedy algorithm allows for producing rules close to an optimal one in the framework of length and taking into account the natural assumption that $P \neq NP$. Statistical analysis of the results obtained regarding the length of the constructed rules allowed us to create a ranking of the proposed algorithms, with the e-greedy algorithm in first place, followed by m-greedy and easy. Furthermore, the results obtained by the e-greedy and easy algorithms differ in a statistically significant way. Nevertheless, considering the number of attributes in the tested datasets and the maximum length of the induced rules, it can be concluded that, in general, these algorithms allow for the creation of short, concise rules.

Future work will be devoted to the investigation of classifiers based on the inhibitory rules learned by the proposed algorithms.

Acknowledgments. The research works presented in the article were carried out within the KAUST-US Collaboration Agreement No. RFS-2025-6853 and supported by the Remote Working Individual Consultancy Agreement No. KAUST-2025-C0148.

References

1. Alsolami, F., Azad, M., Chikalov, I., Moshkov, M.: Decision and Inhibitory Trees and Rules for Decision Tables with Many-valued Decisions, Intelligent Systems Reference Library, vol. 156. Springer (2020)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

2. Alsolami, F., Chikalov, I., Moshkov, M., Zielosko, B.: Optimization of inhibitory decision rules relative to length and coverage. In: Li, T., Nguyen, H.S., Wang, G., Grzymala-Busse, J.W., Janicki, R., Hassanien, A.E., Yu, H. (eds.) Rough Sets and Knowledge Technology - 7th International Conference, RSKT 2012, Chengdu, China, August 17-20, 2012. Proceedings. Lecture Notes in Computer Science, vol. 7414, pp. 149–154. Springer (2012)
3. Alsolami, F., Chikalov, I., Moshkov, M., Zielosko, B.: Optimization of approximate inhibitory rules relative to number of misclassifications. In: Watada, J., Jain, L.C., Howlett, R.J., Mukai, N., Asakura, K. (eds.) 17th International Conference in Knowledge Based and Intelligent Information and Engineering Systems, KES 2013, Kitakyushu, Japan, 9-11 September 2013. Procedia Computer Science, vol. 22, pp. 295–302. Elsevier (2013)
4. Alsolami, F., Chikalov, I., Moshkov, M., Zielosko, B.M.: Length and coverage of inhibitory decision rules. In: Nguyen, N.T., Hoang, K., Jedrzejowicz, P. (eds.) Computational Collective Intelligence. Technologies and Applications - 4th International Conference, ICCCI 2012, Ho Chi Minh City, Vietnam, November 28-30, 2012, Proceedings, Part II. Lecture Notes in Computer Science, vol. 7654, pp. 325–334. Springer (2012)
5. Alsolami, F., Chikalov, I., Moshkov, M.J.: Comparison of heuristics for inhibitory rule optimization. In: Jedrzejowicz, P., Jain, L.C., Howlett, R.J., Czarnowski, I. (eds.) 18th International Conference in Knowledge Based and Intelligent Information and Engineering Systems, KES 2014, Gdynia, Poland, 15-17 September 2014. Procedia Computer Science, vol. 35, pp. 378–387. Elsevier (2014)
6. Azad, M., Moshkov, M.: Multi-stage optimization of decision and inhibitory trees for decision tables with many-valued decisions. Eur. J. Oper. Res. **263**(3), 910–921 (2017)
7. Chekuri, C., Quanrud, K., Zhang, Z.: On approximating partial set cover and generalizations (2019), `https://arxiv.org/abs/1907.04413`
8. Delimata, P., Moshkov, M., Skowron, A., Suraj, Z.: Two families of classification algorithms. In: An, A., Stefanowski, J., Ramanna, S., Butz, C.J., Pedrycz, W., Wang, G. (eds.) Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, 11th International Conference, RSFDGrC 2007, Toronto, Canada, May 14-16, 2007, Proceedings. Lecture Notes in Computer Science, vol. 4482, pp. 297–304. Springer (2007)
9. Delimata, P., Moshkov, M.J., Skowron, A., Suraj, Z.: Comparison of lazy classification algorithms based on deterministic and inhibitory decision rules. In: Wang, G., Li, T., Grzymala-Busse, J.W., Miao, D., Skowron, A., Yao, Y. (eds.) Rough Sets and Knowledge Technology, Third International Conference, RSKT 2008, Chengdu, China, May 17-19, 2008. Proceedings. Lecture Notes in Computer Science, vol. 5009, pp. 55–62. Springer (2008)
10. Delimata, P., Moshkov, M.J., Skowron, A., Suraj, Z.: Inhibitory Rules in Data Analysis: A Rough Set Approach, Studies in Computational Intelligence, vol. 163. Springer (2009)
11. Dinur, I., Steurer, D.: Analytical approach to parallel repetition. In: Shmoys, D.B. (ed.) Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014. pp. 624–633. ACM (2014)
12. Friedman, M.: A Comparison of Alternative Tests of Significance for the Problem of m Rankings. The Annals of Mathematical Statistics **11**(1), 86 – 92 (1940)
13. García, S., Fernández, A., Luengo, J., Herrera, F.: Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25

and data mining: Experimental analysis of power. Information Sciences **180**(10), 2044–2064 (2010)

14. Kelly, M., Longjohn, R., Nottingham, K.: UCI Machine Learning Repository (2025), `http://archive.ics.uci.edu`
15. Kozielski, M., Sikora, M., Wawrowski, L.: Towards consistency of rule-based explainer and black box model - fusion of rule induction and xai-based feature importance. Knowl. Based Syst. **311**, 113092 (2025)
16. Moshkov, M., Skowron, A., Suraj, Z.: Maximal consistent extensions of information systems relative to their theories. Inf. Sci. **178**(12), 2600–2620 (2008)
17. Moshkov, M., Zielosko, B.: Combinatorial Machine Learning - A Rough Set Approach, Studies in Computational Intelligence, vol. 360. Springer, Heidelberg (2011)
18. Moshkov, M.J., Skowron, A., Suraj, Z.: On minimal inhibitory rules for almost all k-valued information systems. Fundam. Informaticae **93**(1-3), 261–272 (2009)
19. Pawlak, Z., Skowron, A.: Rough sets and boolean reasoning. Information Sciences **177**(1), 41–73 (2007)
20. Rissanen, J.: Modeling by shortest data description. Automatica **14**(5), 465–471 (1978)
21. Sakai, H., Nakata, M., Slezak, D., Watada, J.: Rule generation in rough set nondeterministic information analysis (RNIA) and some applications of the obtained rules. Appl. Soft Comput. **172**, 112842 (2025)
22. Skowron, A., Suraj, Z.: Rough sets and concurrency. Bulletin of the Polish Academy of Sciences **41**(3), 237–254 (1993)
23. Stanczyk, U., Zielosko, B., Baron, G.: Attribute relevance and discretisation in knowledge discovery: A study in stylometric domain. In: Mikyska, J., de Mulatier, C., Paszynski, M., Krzhizhanovskaya, V.V., Dongarra, J.J., Sloot, P.M.A. (eds.) Computational Science - ICCS 2023 - 23rd International Conference, Prague, Czech Republic, July 3-5, 2023, Proceedings, Part II. Lecture Notes in Computer Science, vol. 14074, pp. 273–281. Springer (2023)
24. Stefanowski, J., Vanderpooten, D.: Induction of decision rules in classification and discovery-oriented perspectives. International Journal of Intelligent Systems **16**(1), 13–27 (2001)
25. Suraj, Z.: Some remarks on extensions and restrictions of information systems. In: Ziarko, W., Yao, Y.Y. (eds.) Rough Sets and Current Trends in Computing, Second International Conference, RSCTC 2000 Banff, Canada, October 16-19, 2000, Revised Papers. Lecture Notes in Computer Science, vol. 2005, pp. 204–211. Springer (2001)
26. Zielosko, B.: Optimization of approximate decision rules relative to coverage. In: Kozielski, S., Mrozek, D., Kasprowski, P., Malysiak-Mrozek, B., Kostrzewa, D. (eds.) Beyond Databases, Architectures, and Structures - 10th International Conference, BDAS 2014, Ustron, Poland, May 27-30, 2014. Proceedings. Communications in Computer and Information Science, vol. 424, pp. 170–179. Springer (2014)
27. Zielosko, B., Moshkov, M., Tetteh, E.T.: Optimization of inner and general rules. Inf. Sci. **719**, 122466 (2025)

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29915-4_25