1 Introduction

We present here a new approach to data analysis called rough set theory (Pawlak, 1982). The rough set philosophy is founded on the assumption that with every object of the universe of discourse we associate some information (data, knowledge). Objects characterized by the same information are indiscernible (similar) in view of the available information about them. The indiscernibility relation generated in this way is the mathematical basis of rough set theory.

Any set of all indiscernible (similar) objects is called an elementary set, and form a basic granule (atom) of knowledge about the universe. Any union of some elementary sets is referred to as crisp (precise) set – otherwise the set is rough (imprecise, vague).

Consequently each rough set has boundary-line cases, i.e., objects which cannot be with certainty classified as members of the set or of its complement. Obviously crisp sets have no boundary-line elements at all. That means that boundary-line cases cannot be properly classified by employing the available knowledge.

Thus, the assumption that objects can be "seen" only through the information available about them leads to the view that knowledge has granular structure. Due to the granularity of knowledge some objects of interest cannot be discerned and appear as the same (or similar). As, a consequence vague concepts, in contrast to precise concepts, cannot be characterized in terms of information about their elements. Therefore in the proposed approach we assume that any vague concept is replaced by a pair of precise concepts – called the lower and the upper approximation of the vague concept. The lower approximation consists of all objects which surely belong to the concept and the upper approximation contains all objects which possible belong to the concept. Obviously, the difference between the upper and the lower approximation constitute the boundary region of the vague concept. Approximations are two basic operations in the rough set theory.

Rough set theory overlaps to a certain degree many other mathematical theories. Particularly interesting is the relationship with fuzzy set theory and Dempster-Shafer theory of evidence. The concepts of rough set and fuzzy set are different since they refer to various aspects of imprecision (Pawlak and Skowron, 1994) whereas the connection with theory of evidence is more substantial (Skowron and Grzymal-Busse, 1994). Besides, rough set theory is related to discriminant analysis (Krusińska et al., 1992), Boolean reasoning methods (Skowron and Rauszer, 1992) and others. The relationship between rough
set theory and decision analysis is presented in (Pawlak and Słowiński, 1994, Słowiński, 1993). More details concerning these relationships can be found in the references.

Despite of the relationships rough set theory can be viewed in its own rights, as an independent discipline.

Rough set theory has found many interesting applications. The rough set approach seems to be of fundamental importance to AI and cognitive sciences, especially in the areas of machine learning, knowledge acquisition, decision analysis, knowledge discovery from databases, expert systems, inductive reasoning and pattern recognition. It seems of particular importance to decision support systems.

The main advantage of rough set theory is that it does not need any preliminary or additional information about data – like probability in statistics, or basic probability assignment in Dempster-Shafer theory and grade of membership or the value of possibility in fuzzy set theory.

The rough set theory has been successfully applied in many real-life problems in medicine, pharmacology, engineering, banking, financial and market analysis and others. Some exemplary applications are listed below.


More about applications of the rough set theory can be found in (Grzymala-Busse 1995, Lin 1994, Słowiński R., 1992, Wang 1995, Ziarko 1993). Besides, many other fields of application, e.g., time series analysis, image processing and character recognition, are being extensively explored.

Application of rough sets requires a suitable software. Many software systems for workstations and personal computers based on rough set theory have been developed. The most known include LERS (Grzymala-Busse 1992), Rough DAS and Rough Class and DATALOGIC (Szladow 1993). Some of them are available commercially.

One of the most important and difficult problem in software implementation of the presented approach is optimal decision rule generation from data. Many various approaches to solve this task can be found in (Bazan et al., 1995, 1994, Grzymala-Busse et al., 1995, Skowron 1995, Skowron and Stepaniuk 1994, Tsumoto and Tanaka 1995, Wróblewski 1995). The relation to other methods of rule generation is dwelt in (Grzymala-Busse et
The theory has many important advantages. Some of them are listed below.

- Provides efficient algorithms for finding hidden patterns in data.
- Finds minimal sets of data (data reduction).
- Evaluates significance of data.
- Generates sets of decision rules from data.
- It is easy to understand.
- Offers straightforward interpretation of obtained results.

- Most algorithms based on the rough set theory are particularly suited for parallel processing, but in order to exploit this feature fully, a new computer organization based on rough set theory is necessary.

Although rough set theory has many achievements to its credit, nevertheless several theoretical and practical problems require further attention.

Especially important is widely accessible efficient software development for rough set based data analysis, particularly for large collections of data analysis.

Despite of many valuable methods of efficient, optimal decision rule generation methods from data, developed in recent years based on rough set theory – more research here is needed, particularly, when quantitative attributes are involved. In this context also further discretization methods for quantitative attribute values are badly needed. Also an extensive study of a new approach to missing data is very important. Comparison to other similar methods still requires due attention, although important results have been obtained in this area. Particularly interesting seems to be a study of the relationship between neural network and rough set approach to feature extraction from data.

Last but not least, rough set computer is badly needed for more serious computations in decision support. Some research in this area is already in progress.

2 Decision Tables and Decision Rules

Data are often presented as a table, columns of which are labeled by attributes, rows by objects of interest and entries of the table are attribute values. An example of such table is shown below.

<table>
<thead>
<tr>
<th>Store</th>
<th>E</th>
<th>Q</th>
<th>L</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>high</td>
<td>good</td>
<td>no</td>
<td>profit</td>
</tr>
<tr>
<td>2</td>
<td>med.</td>
<td>good</td>
<td>no</td>
<td>loss</td>
</tr>
<tr>
<td>3</td>
<td>med.</td>
<td>good</td>
<td>no</td>
<td>profit</td>
</tr>
<tr>
<td>4</td>
<td>no</td>
<td>avg.</td>
<td>no</td>
<td>loss</td>
</tr>
<tr>
<td>5</td>
<td>med.</td>
<td>avg.</td>
<td>yes</td>
<td>loss</td>
</tr>
<tr>
<td>6</td>
<td>high</td>
<td>avg.</td>
<td>yes</td>
<td>profit</td>
</tr>
</tbody>
</table>

Table 1

In the table six stores are characterized by four attributes:

- E – empowerment of sales personnel,
- Q – perceived quality of merchandise,
- L – high traffic location,
- P – store profit or loss.

Sometimes we distinguish in such a table two classes of attributes, called condition and decision (action) attributes. For example in Table 1 attributes E, Q, L are condition attributes, whereas the attribute P, is a decision one. Such tables will be referred to as decision tables.

Each row of a decision table determinates a decision rule, which specifies decisions (actions) that should be taken when conditions pointed out by condition attributes are satisfied. Decision rules 2) and 3) in Table 1 have the same conditions by different decisions. Such rules are called inconsistent (nondeterministic, conflicting); otherwise the rules are referred to as consistent (certain, deterministic, nonconflicting). Decision tables containing inconsistent decision rules are called inconsistent (nondeterministic, conflicting); otherwise the table is consistent (deterministic, non conflicting).

The number of consistent rules to all rules in a decision table can be used as consistency measure of the decision table, and will be denoted by \( \gamma(C, D) \), where \( C \) and \( D \) are condition and decision attributes respectively. Thus if \( \gamma(C, D) = 1 \) the decision table is consistent and if \( \gamma(C, D) \neq 1 \) the decision table is inconsistent. For example for Table 1 \( \gamma(C, D) = 4/6 \).

Decision rules are often presented as implications and are called "if... then..." rules. For example rule 1) in Table 1 can be presented as implication

if (H, high) and (Q, good) and (L, no) then (P, profit).

3 Rough Sets and Approximations

As mentioned in the introduction, the starting point of the rough set theory is the indiscernibility relation, generated by information about objects of interest. The indiscernibility relation is intended to express the fact that due to the lack of knowledge we are
unable to discern some objects employing the available information. That means that, in general, we are unable to deal with single objects but we have to consider clusters of indiscernible objects, as fundamental concepts of our theory.

Now we present above considerations more formally.

Suppose we are given two finite, non-empty sets $U$ and $A$, where $U$ is the universe, and $A$ – a set attributes. With every attribute $a \in A$ we associate a set $V_a$, of its values, called the domain of $a$. Any subset $B$ of $A$ determines a binary relation $I(B)$ on $U$, which will be called an indiscernibility relation, and is defined as follows:

$$x I(B) y \text{ if and only if } a(x) = a(y) \text{ for every } a \in A,$$

where $a(x)$ denotes the value of attribute $a$ for element $x$.

Obviously $I(B)$ is an equivalence relation. The family of all equivalence classes of $I(B)$, i.e., partition determined by $B$, will be denoted by $U/I(B)$, or simple $U/B$; an equivalence class of $I(B)$, i.e., block of the partition $U/B$, containing $x$ will be denoted by $B(x)$.

If $(x, y)$ belongs to $I(B)$ we will say that $x$ and $y$ are $B$-indiscernible. Equivalence classes of the relation $I(B)$ (or blocks of the partition $U/B$) are refereed to as $B$-elementary sets. In the rough set approach the elementary sets are the basic building blocks (concepts) of our knowledge about reality.

The indiscernibility relation will be used next to define basic concepts of rough set theory. Let us define now the following two operations on sets

$$B_s(X) = \{ x \in U : B(x) \subseteq X \},$$

$$B^*(X) = \{ x \in U : B(x) \cap X \neq \emptyset \},$$

assigning to every subset $X$ of the universe $U$ two sets $B_s(X)$ and $B^*(X)$ called the $B$-lower and the $B$-upper approximation of $X$, respectively. The set

$$BN_B(X) = B^*(X) - B_s(X)$$

will be referred to as the $B$-boundary region of $X$.

If the boundary region of $X$ is the empty set, i.e., $BN_B(X) = \emptyset$, then the set $X$ is crisp (exact) with respect to $B$; in the opposite case, i.e., if $BN_B(X) \neq \emptyset$, the set $X$ is to as rough (inexact) with respect to $B$. Let us depict the above ideas by an example refering to Table 1.

Let us observe that each store has different description in terms of attributes $E, Q, L$ and $P$, thus all stores may be distinguished (discerned) employing information provided by all attributes. However, stores 2 and 3 are indiscernible in terms of attributes $E, Q$ and $L$, since they have the same values of these attributes. Similarly, stores 1, 2 and 3 are indiscernible with respect to attributes $Q$ and $L$, etc.

Each subset of attributes determines a partition (classification) of all objects into classes having the same description in terms of these attributes. For example, attributes $Q$ and $L$ aggregate all stores into the following classes $\{1, 2, 3\}, \{4\}, \{5, 6\}$. Thus, each information table determines a family of classification patterns which are used as a basis of further considerations.

Suppose we are interested in the following problem: what are the characteristic features of stores having profit (or loss) in view of information available in Table 1. In other words,
the question is whether we are able to describe set (concept) \( \{1, 3, 6\} \) (or \( \{2, 4, 5\} \)) in terms of attributes \( E, Q \) and \( L \). It can be easily seen that this is impossible, since stores 2 and 3 display the same features in terms of attributes \( E, Q \) and \( L \), but store 2 makes a profit, whereas store 3 has a loss. Thus information given in Table 1 is not sufficient to answer this question. However, we can give a partial answer to this question. Let us observe that if the attribute \( E \) has the value high for a certain store, then the store makes a profit, whereas if the value of the attribute \( E \) is low, then the store has a loss. Thus information contained in Table 1, we can say for sure that stores 1 and 6 make a profit, stores 4 and 5 have a loss, whereas stores 2 and 3 cannot be classified as making a profit or having a loss. Therefore we can give approximate answers only. Employing attributes \( E, Q \) and \( L \), we can say that stores 1 and 6 surely make a profit, i.e., surely belong to the set \( \{1, 6\} \), whereas stores 1, 2, 3 and 6 possibly make a profit, i.e., possibly belong to the set \( \{1, 3, 6\} \). Thus the set \( \{1, 6\} \) is the lower approximation of the set (concept) \( \{1, 3, 6\} \), and the set \( \{1, 2, 3, 6\} \) – is the upper approximation of the set \( \{1, 3, 6\} \). The set \( \{2, 3\} \), being the difference between the upper approximation and the lower approximation is referred to as the boundary region of the set \( \{1, 3, 6\} \).

Rough set can be also characterized numerically by the following coefficient

\[
\alpha_B(X) = \frac{|B_*(X)|}{|B^*(X)|}
\]

called accuracy of approximation, where \(|X|\) denotes the cardinality of \( X \). Obviously \( 0 \leq \alpha_B(X) \leq 1 \). If \( \alpha_B(X) = 1 \), \( X \) is crisp with respect to \( B \) \((X \) is precise with respect to \( B \)), and otherwise, if \( \alpha_B(X) < 1 \), \( X \) is rough with respect to \( B \) \((X \) is vague with respect to \( B \)).

Rough sets can be also defined using a rough membership function, defined as

\[
\mu^B_X(x) = \frac{|X \cap B(x)|}{|B(x)|}
\]

Obviously

\[
\mu^B_X(x) \in [0, 1].
\]

Value of the membership function \( \mu_X(x) \) is kind of conditional probability, and can be interpreted as a degree of certainty to which \( x \) belongs to \( X \) (or \( 1 - \mu_X(x) \), as a degree of uncertainty).

4 Dependency of Attributes

Another important issue in data analysis is discovering dependencies between attributes. Intuitively, a set of attributes \( D \) depends totally on a set of attributes \( C \), denoted \( C \Rightarrow D \), if all values of attributes from \( D \) are uniquely determined by values of attributes from \( C \). In other words, \( D \) depends totally on \( C \), if there exists a functional dependency between values of \( D \) and \( C \).

Formally dependency can be defined in the following way. Let \( D \) and \( C \) be subsets of \( A \). We say that \( B \) depends totally on \( C \), if and only if \( I(C) \subseteq I(D) \). That means that the partition generated by \( C \) is finer than the partition generated by \( D \). Notice, that the concept of dependency discussed above corresponds to that considered in relational databases.
We would need also a more general concept of dependency of attributes, called a partial dependency of attributes.

Formally, the above idea can be formulated as follows. Let $D$ and $C$ be subsets of $A$. We say that $D$ depends in degree $k$, $0 \leq k \leq 1$, on $C$, denoted $C \Rightarrow_k D$, if

$$
\begin{equation}
  k = \frac{|POS_C(D)|}{|U|},
\end{equation}
$$

where

$$
POS_C(D) = \bigcup_{X \in U/I(D)} C_*(X).
$$

The expression $POS_C(D)$, called a positive region of the partition $U/D$ with respect to $C$, is the set of all elements of $U$ that can be uniquely classified to blocks of the partition $U/D$, by means of $C$.

Thus the coefficient $k$ expresses the ratio of all elements of the universe, which can be properly classified to blocks of the partition $U/D$, employing attributes $C$. Notice that for $k = 1$ we get the previous definition of total dependency.

Obviously, a decision table is consistent if and only if $k = 1$, otherwise, i.e., if $k \neq 1$, the decision table is inconsistent; if $k = 0$ we will say that the decision table is totally inconsistent.

Obviously dependency between attributes can be defined using the consituency factor i.e.,

$$
C \Rightarrow_k D,
$$

where $k = \gamma(C, D)$.

Summing up: $D$ is totally (partially) dependent on $C$, if all (some) elements of the universe $U$ can be uniquely classified to blocks of the partition $U/D$, employing $C$.

## 5 Data Reduction (Compression)

We often face a question whether we can remove some data from a data-table preserving its basic properties, that is – whether a table contains some superfluous data.

Very often we are interested in reducing the number of condition attributes preserving the degree of dependency between decision and condition attributes. That means that we want to preserve the ability to classify objects to decision classes using smaller number of conditions attributes – or, in other words, we want to make decisions employing less conditions.

To this end we define the concept of a reduct of attributes. $B \subseteq C$ is a $D$-reducts of $C$, if $B$ is a minimal subset of $C$ such that

$$
\gamma(B, D) = \gamma(C, D).
$$

For example, in Table 1 we have two reducts of $\{E, Q, L\}$ – namely $\{E, Q\}$ and $\{E, L\}$.

That means that either $Q$ or $L$ can be removed from the table without changing the degree of consistency of the table.

The intersection of all reducts is called the $D$-core of condition attributes, i.e.,

$$
\text{CORE}_D(C) = \bigcap \text{RED}_D(C).
$$
In Table 1 the core is the attribute \( E \). The core can be interpreted as a set of the “most important” attributes, which cannot be removed from data, without effecting the consistency factor of the table.

References


