Rough Sets: A Knowledge Discovery Technique for Multifactorial Medical Outcomes*

Aleksander Øhrn, MSc † Todd Rowland, MD ‡

Abstract

Rough sets is a fairly new and promising technique for data mining and knowledge discovery from databases. Most introductory articles to rough sets are highly technical and mathematically oriented. This tutorial paper presents the fundamentals of rough set theory in a non-technical manner, and outlines how the technique can be used to extract minimal if-then rules from tables of empirical data that either fully or approximately describe given example classifications. Since such rules are readily interpretable, they can be inspected in order to yield possible new insight into how various contributing factors interact, and thus serve as hypothesis-generators for further research. Additionally, the set of mined rules may function as a classifier of new, unseen cases. An example application for prediction of ambulation for patients with spinal cord injury is given.

Keywords: rough sets, data mining, knowledge discovery, machine learning, classification, modeling, outcome, prognosis, ambulation.

1 Introduction

Large databases are often collected for research or business purposes. Often these databases grow so large that human inspection and interpretation of the data is not feasible, with a gap between data generation and data understanding as a result. Clearly, tools and techniques that can aid in extracting unknown interesting patterns buried in the data would be useful to help bridge this gap. This process of inducing patterns from data is often referred to as *data mining*. The term *knowledge discovery from databases* incorporates data mining as a step, but also covers the full process from initial data cleansing and preprocessing to, perhaps most importantly, interpretation of the induced patterns [1].

Classical tools for database querying may be adequate if you know what to look for. For instance, current systems are good for answering questions of the type "How many patients suffered from spinal cord injuries in California last February, who are they, and what was their average length of stay in hospital?" But often the most interesting questions to pose cannot be formulated as straightforward lookups. Consider for instance the following question of interest: "What are the main factors that

*To appear in American Journal of Physical Medicine and Rehabilitation.

†Knowledge Systems Group, Dept. of Computer and Information Science, Norwegian University of Science and Technology, Trondheim, Norway. E-mail **aleks@idi.ntnu.no**.

‡Decision Systems Group, Brigham and Women's Hospital, Harvard Medical School, Boston, USA. E-mail **trowland@dsg.harvard.edu**.

determine how successful our current rehabilitation program is, and how do these factors interact?" To be able to deal with such advanced queries, more intelligent data analysis tools are needed.

Research in computer science has in the last decade spawned a vast multitude of methods that "learn" from examples, and that can be used to extract patterns from empirical data for classification. Such machine learning techniques are increasingly being applied to medical data sets. The goal of inducing models from data is often twofold: Not only do we want the model to be able to determine the outcomes of new patients based on data from previous patients, but we are also interested in analyzing the structure of the model in order to gain new insight into the problem at hand. By model structure is meant the type of the model's different components, how they relate to each other and how they can be interpreted. Different machine learning methods vary in how they facilitate this knowledge discovery aspect, depending on the type of models they produce. A point that is often held forth in favor of methods that produce decision trees or rule sets is that the models are directly readable and interpretable. In contrast, methods such as artificial neural networks provide no clear indication of relative contribution for each attribute. Logistic regression does provide coefficients in the regression equations and odds ratios for each attribute that allow interpretation, but require some statistical knowledge.

This tutorial article provides an overview of the fundamentals of *rough set theory*, a methodology that can be used to induce if-then rules from databases. Section 2 outlines the basic theoretical concepts, while Section 3 discusses some of the steps that a rough set data analysis project comprises. An example application for prediction of ambulation for patients with spinal cord injury is given in Section 4.

2 Rough Set Theory

Rough set theory [2, 3] was developed in Poland in the early 1980s, and concerns itself with the classificatory analysis of imprecise, uncertain or incomplete information expressed in terms of data acquired from experience. Basically, rough set theory deals with the approximation of sets that are difficult to describe with the available information. In a medical setting, a set of interest could be the set of patients with a certain disease or outcome.

2.1 Decision Tables

The basic vehicle for representing data is assumed to be a flat[1] table, where each row represents a new case or patient (an *object*) and each column represents a variable or observation (an *attribute*) that can be measured for each object. In a supervised learning setting, i.e., where the outcome or classification is known, a distinguished attribute denotes the outcome and is called the *decision attribute*. The other attributes are referred to as *condition attributes*. Tables adhering to these requirements are called *decision tables*. A small example decision table can be found in Table 1.

Note that in Table 1 all the variables have been transformed into categorical ones, even though some of them are inherently numerical. This process is called *dis-*

[1] Flat in a logical sense, at least. Physically, the table may be a view or composition of several distinct underlying tables in a database.

	Age	Sex	LEMS	Walk
Smith	16-30	Male	50	Yes
Jones	16-30	Male	0	No
Parker	31-45	Male	1-25	No
Hanson	31-45	Male	1-25	Yes
Moore	46-60	Female	26-49	No
Fields	16-30	Female	26-49	Yes
Starr	46-60	Female	26-49	No

Table 1: A small, artificial decision table. The table has seven cases or objects, three condition attributes (Age, Sex and Lower Extremity Motor Score, LEMS), and one outcome or decision attribute (Walk) with two possible outcomes.

cretization, and can be done manually or automatically. The rough set approach is based on logic and discrete mathematics, and works best if numerical attributes are discretized in a preprocessing step. Advanced algorithms for fully automatic discretization exist, some of which are also based on rough sets. Discretization is further discussed in Section 3.1. Also, Table 1 might give the false impression that the rough set approach is limited to binary outcomes. The approach can readily be applied to any finite number of outcomes.

2.2 Indiscernibility

The notion of *indiscernibility* is fundamental to rough set theory. Informally, two objects in a decision table are indiscernible if one cannot distinguish between them on the basis of a given set of attributes. Hence, indiscernibility is a function of the set of attributes under consideration. For each set of attributes we can thus define a binary *indiscernibility relation*, which is a collection of pairs of objects that are indiscernible to each other. For example, in Table 1, Smith and Jones are indiscernible on the basis on Age, but not on the basis of Age and LEMS combined.

As described, the indiscernibility relation for a given attribute set is mathematically an *equivalence relation*. However, there are extensions to rough set theory where some of the requirements that this imposes are lifted. Such extensions can handle missing values and deal with hierarchies among attribute values, but are outside the scope of this article. In the following, for the sake of simplicity, it will be assumed that none of the attribute values are missing and that strict inequality among attribute values enables us to discern between objects.

An indiscernibility relation partitions the set of cases or objects into a number of *equivalence classes*. An equivalence class of a particular object is simply the collection of objects that are indiscernible to the object in question. For example, in Table 1, the equivalence class for Smith with respect to Age would consist of Smith, Jones and Fields. The equivalence class for Smith with respect to Age and Sex combined would consist of Smith and Jones, while the equivalence class for Smith with respect to Age, Sex and LEMS combined would consist of Smith alone.

2.3 Set Approximations

The equivalence classes from Section 2.2 form basic building blocks from which sets of cases or objects can be assembled. Sets of interest to assemble in a supervised

learning setting would typically be the sets of cases with the same value for the outcome variable. This can be done by examining the "purity" of each equivalence class with respect to the value of the outcome variable, and assigning (the objects in) the equivalence class to the interior or exterior of the set accordingly. It may be that such sets cannot be defined in a crisp or exact manner, and we can only hope to approximate them from above and below:

- The *lower approximation* of a set of cases (with respect to a given set of attributes) is defined as the collection of cases whose equivalence classes are fully contained in the set of cases we want to approximate.
- The *upper approximation* of a set of cases (with respect to a given set of attributes) is defined as the collection of cases whose equivalence classes are at least partially contained in (i.e., overlap with) the set of cases we want to approximate.

The upper approximation will always include the lower approximation. If the lower and upper approximations are equal, then the set of interest can be defined crisply. These two approximations define three *approximation regions*:

- The *inside region* equals the lower approximation. Cases that are members of the inside region are definite members of the set we want to approximate.
- The *outside region* equals the complement of the upper approximation. Cases that are members of the outside region are definite non-members of the set we want to approximate.
- The *boundary region* equals the difference between the upper and lower approximations. Cases that are members of the boundary region have a membership status that cannot be ascertained with certainty, at least not on the basis of the attributes that the approximations are built from.

A *rough set* is any set defined through its lower and upper approximations. Figure 1 displays these abstract notions graphically.

A small worked example might be in order. Let "$\{\ldots\}$" denote a set. Returning to Table 1, assume we want to approximate the set of all patients that walk, $\{\text{Smith}, \text{Hanson}, \text{Fields}\}$, using all three condition attributes Age, Sex and LEMS. The approximations would be as shown below, where "+" denotes set union between the equivalence classes. An abstract depiction of this example can be found in Figure 2.

$$\begin{aligned} \text{Lower approximation} &= \{\text{Smith}\} + \{\text{Fields}\} \\ \text{Upper approximation} &= \{\text{Smith}\} + \{\text{Parker}, \text{Hanson}\} + \{\text{Fields}\} \\ \text{Boundary region} &= \{\text{Parker}, \text{Hanson}\} \\ \text{Outside region} &= \{\text{Jones}\} + \{\text{Moore}, \text{Starr}\} \end{aligned}$$

2.4 Reducts

A very important issue not yet discussed is that of redundancy. It is often the case that some of the attributes or some of the attribute values are superfluous. *Reducts*

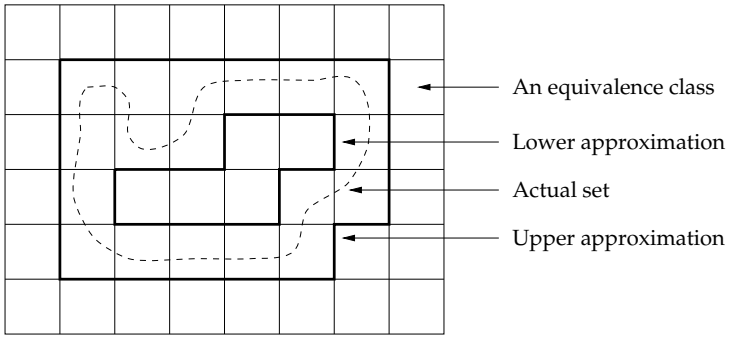

Figure 1: We can imagine the indiscernibility relation to define a grid that we overlay our universe of cases, with each equivalence class being displayed as a "pixel" or square in the grid. This grid forms our approximation space. The set of cases that we want to approximate is drawn as a line that crosses pixel boundaries, and cannot be defined crisply within our approximation space. The lower and upper approximations of the set are drawn as thick gridlines.

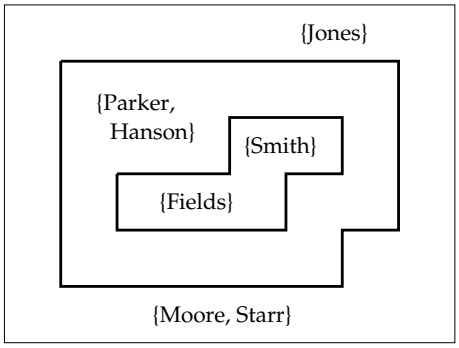

Figure 2: Approximating the set of walking patients in Table 1, using the two condition attributes Age and LEMS. Since the boundary region is non-empty, the concept of ambulation can only be defined roughly with the available information.

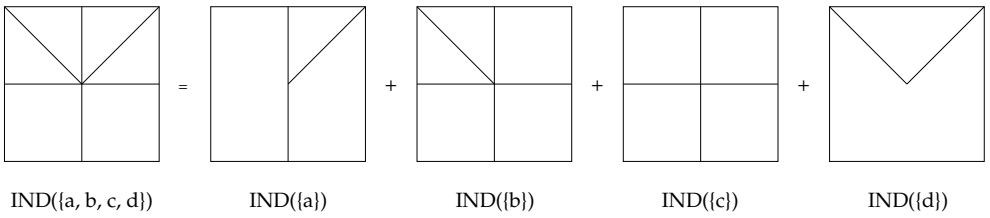

Figure 3: Conceptually, the indiscernibility relation given all the attributes can be seen as the superposition of the indiscernibility relations for each of the individual attributes. As such, not all of them might be needed in order to "sum up" to the total observed indiscernibility. A reduct is a minimal attribute set that preserves the indiscernibility relation. In this example, the reducts are $\{a, b\}$, $\{b, d\}$ and $\{c, d\}$. Observe that $\{b\}$ is "almost" a reduct.

enable us to discard functionally redundant information. Formally, a reduct is defined as a minimal set of attributes that preserves the indiscernibility relation computed on the basis of the full set of attributes. Preserving the indiscernibility relation preserves the equivalence classes and hence our ability to form approximations. In practical terms, reducts help us construct smaller and simpler models, and provide a focus on the decision-making process. Typically, a decision table may have many reducts. A graphical display of the notion of reducts can be found in Figure 3.

Different variations and combinations of indiscernibility are possible, and give rise to four different types of reducts that each answer one of the following questions:

1. *"What is the minimum amount of information I need to be able to discern a particular object (or, more precisely, its equivalence class) from all other objects?"* This corresponds to determining the minimum set of conditions that one must specify in a database query in order to ensure that the query will return the desired object (or, more precisely, the equivalence class to which the object belongs) and nothing more. For instance, in Table 1, LEMS alone is enough information to be able to discern Smith. A database query for all objects that have a condition of LEMS equal to 50 is thus guaranteed to return Smith and nothing more.

2. *"What is the minimum amount of information I need to be able to discern all objects from each other?"* Readers familiar with relational databases may recognize the similarity between reducts of this type and functional dependencies and keys. In fact, such reducts constitute minimal keys for the table from which they were computed.

3. *"What is the minimum amount of information I need to be able to determine the outcome (or, more precisely, the approximation region) of a particular object?"* For induction of classification rules, reducts of this type are of particular interest, as will be discussed in Section 2.5. This corresponds to determining the minimum set of conditions that one must specify in a database query in order to ensure that the query will return only objects that have the same outcome (or, more precisely, belong to the same approximation region) as the object in question. Returning to Table 1, one reduct of this type computed from Fields is the set $\{\text{Age}, \text{LEMS}\}$, implying that a database query for all objects that have an age between 16 and 30 and a LEMS in the range 26 to 49 is guaranteed to only return objects with the same outcome value as Fields. In this example, Fields would be the only object returned, as it happens.

4. *"What is the minimum amount of information I need to be able to determine the outcome of all objects?"* Again, reducts of this types are analogous to relational database concepts. Such a reduct defines a functional dependency between the attributes in the reduct and the outcome attribute. For instance, in Table 1, Age and LEMS is enough information to be able to discern all objects from each other. The reader may verify that Sex is indeed a redundant attribute with respect to being able to determine the values for Walk in the given decision table.

Computing reducts is without doubt the most complex and computer-intensive step in rough set data analysis. Computing all reducts is known to belong to a theoretical class of problems that, informally, requires an amount of computation that grows exponentially with the size of the problem. In this case, the problem size is dominated by the number of attributes involved. However, this theoretical drawback

does not necessarily render the approach useless in practice. An exhaustive computation is still feasible unless the number of attributes is high, and computationally efficient heuristics exist that can be used to search for individual reducts.

But how are reducts actually computed? A reduct is equivalent to a so-called *prime implicant* of a suitably constructed Boolean function that expresses the discernibility one wishes to preserve, with one Boolean variable per attribute. Informally, a prime implicant is a minimal conjunction of Boolean literals (variables or their negations) such that the function evaluates to true if the conjunction evaluates to true. In order to get a flavor of how such Boolean functions are constructed, consider Table 1. Assume that we are interested in determining minimal sets of attributes that enable us to determine the outcome value for Fields, as discussed above. Since we are only interested in computing reducts that determine the outcome value, we do not bother to discern Fields from other patients that belong to the same approximation region as Fields in Figure 2, Smith in this example. To discern Fields from the other patients on the basis of the condition attributes, we need Sex or LEMS (to discern her from Jones), and we need Age, Sex or LEMS (to discern her from Parker and Hanson), and we need Age (to discern her from Moore and Starr). We can express this as the Boolean function below, where "+" reads "or" and "×" reads "and".

$$
\begin{array}{rcll}
f & = & (\text{Sex} + \text{LEMS}) \times & \text{(Discerns Fields from Jones)} \\
 & & (\text{Age} + \text{Sex} + \text{LEMS}) \times & \text{(Discerns Fields from Parker)} \\
 & & (\text{Age} + \text{Sex} + \text{LEMS}) \times & \text{(Discerns Fields from Hanson)} \\
 & & (\text{Age}) \times & \text{(Discerns Fields from Moore)} \\
 & & (\text{Age}) & \text{(Discerns Fields from Starr)}
\end{array}
$$

Through a series of truth-preserving transformations [4], the function above can be rewritten into the minimal and semantically equivalent function below.

$$f = (\text{Age} \times \text{Sex}) + (\text{Age} \times \text{LEMS})$$

The reducts for Fields with respect to the outcome attribute Walk are thus $\{\text{Age}, \text{Sex}\}$ and $\{\text{Age}, \text{LEMS}\}$. Both are, by definition of a reduct, minimal and either one can be used. If some kind of attribute cost information is available, this information can be used to rank the reducts in terms of their total costs.

In real-world applications, noise and other impurities may be present in the data. To cope with this in a satisfactory manner, we are typically interested in attribute subsets that are "almost" reducts. Such attribute subsets enable us to discern "almost all" objects from each other, to "almost" determine the outcome of a particular object and so on. This issue is further discussed in Section 2.5.

2.5 Rules

A *decision rule* is here defined to be a statement on the form "if C then D", where the condition C is a set of elementary conditions connected by "and", and the decision D is a set of possible outcomes connected by "or". Two example decision rules culled from Table 1 are:

if	Age is 16-30 and		if	LEMS is 1-25
	LEMS is 50		then	Walk is Yes or
then	Walk is Yes			Walk is No

The decision rules above can be interpreted within the rough set framework introduced in Section 2.3. If the then-part of the rule lists more than one possible outcome, that can be interpreted as describing one or more cases that lie in the boundary region of the set approximation when using the attributes listed in the if-part of the rule. If the then-part of the rule lists a single outcome "Yes" (or "No"), that can be interpreted as describing one or more cases that lie in either the inside (or the outside) region of the approximation of the set of walking patients, when using the attributes listed in the if-part of the rule.

The main challenge in inducing rules from decision tables lies in determining which attributes that should be included in the conditional part of the rule. This is where the notion of reducts discussed in Section 2.4 enter the picture. To obtain decision rules that are minimal and yet describe the data accurately, one can compute the reducts per case relative to the outcome attribute, and read off the attribute values for that case. For example, in Table 1, the single variable LEMS is such a reduct for Smith. This defines a decision rule "if LEMS is 50 then Walk is Yes". Doing this for all patients creates a set of minimal decision rules that form a lossless and minimal if-then representation of the data in the decision table.

In most real-world applications, the data is likely to contain noise or other impurities, and a lossless, minimal representation of the data is likely to *overfit* the patterns we are interested in extracting. Overfitted models provide rules that are overly specific and thus incorporate the noise and peculiarities of the training data, instead of being shorter and expressing more general relationships between conditions and decisions. Less specific patterns are likely to generalize better to unseen cases. Hence, instead of true reducts, one is typically interested in computing reduct approximations, i.e., attribute sets that "almost" preserve the indiscernibility relation. From such reduct approximations one can generate decision rules that reveal probabilistic (i.e., non-deterministic) relationships between a set of conditions and a set of possible decisions. One way of computing good reduct approximations is to compute reducts on the basis of only a random sample of the training data. This process is then repeated for many different random samples, and the reducts that occur across "many" of these random samples are considered to constitute stable reduct approximations. The reducts that are computed on the basis of such repeated sampling are called *dynamic reducts* [5].

A decision rule "if C then D" can have several numerical quantities associated with it. The *support* of a decision rule is the number of cases in the originating decision table that match the pattern described by the rule. From simple support counts, a number of quantities of interest are derivable, such as the rule's *accuracy* and its *coverage*. Let d denote one of the possible outcome values listed in the then-part of the rule. Accuracy is then simply defined as the number of cases that match both C and d simultaneously, divided by the number of cases that match C. Coverage is defined as the number of cases that match both C and d simultaneously, divided by the number of cases that match d. Accuracy is thus an estimate of the conditional probability $\Pr(d \mid C)$, while coverage is an estimate of the conditional probability $\Pr(C \mid d)$.

3 The Modeling Process

Typically, the overall rule-based modeling and validation process consists of the basic steps listed below. The procedure can be repeated in a systematic fashion by

employing a cross-validation or bootstrap scheme, for example.

1. *Discretization:* Transform the non-categorical attributes in the decision table into categorical ones.
2. *Rule induction:* Compute if-then rules from the discretized decision table. For the rough set approach, this is done as outlined in Section 2.5.
3. *Rule application:* Apply the rules to classify new cases.
4. *Model evaluation:* Evaluate the classificatory performance of the rules.

This section briefly discusses some of these steps. Note that these steps are not specific for the rough set approach, but are common for virtually all rule- or tree-based modeling schemes.

3.1 Discretization

The rough set approach is a logically founded approach based on indiscernibility. This means that we do not need a notion of "distance" between attribute values, in contrast to many other machine learning techniques. An implication of this, however, is that non-categorical attributes should be discretized as a preprocessing step. The discretization step thus determines how coarsely we want to view the world. For numerical attributes, this amounts to searching for cut-off points that define intervals. For example, we might want to view all patients between 45 and 60 years old as belonging to the same age group. Discretization is a step that is not specific to the rough set approach, most rule or tree induction algorithms currently require it (or perform it implicitly behind the scenes) to perform well.

In the medical domain there are often values that are "natural" to use as cut-off points and that can be used to manually discretize variables. If such cut-off points cannot be found in the literature, algorithms exist that can be used to suggest them. These algorithms mainly fall into three categories, depending on how they look for the boundaries they create:

- Some algorithms only consider one condition attribute at a time, but do not consider that attribute together with the outcome attribute. A simple example of such an algorithm would be a technique sometimes referred to as *equal-frequency binning*. This method simply divides the range into a predefined number of intervals or bins, such that approximately the same number of cases fall into each interval. For three intervals, this intuitively corresponds to labeling the cases with "low", "medium" or "high" values within the population for that attribute.
- Some algorithms only consider one condition attribute at a time, and do so while simultaneously considering the values for the outcome attribute. The decision to add a cut-off point is then often motivated by statistical or information-theoretic measures. The algorithm outlined in [6] is an example of such an algorithm, based on entropy.
- Some algorithms consider all condition attributes simultaneously, and do so while also considering the values for the outcome attribute. Such algorithms

are often based on preserving discernibility under the constraint of introducing a minimum number of cut-off points. An example of such an algorithm is the one outlined in [7], based on Boolean reasoning and rough sets.

3.2 Rule Application

When a collection of decision rules have been induced from a set of training examples, they can be inspected to see if they reveal any novel relationships between attributes that are worth pursuing for further research. In order to estimate their classificatory power of the rules, they can also be used to predict the outcomes of new patients.

Several schemas can be envisioned for using an unordered set of rules to classify new cases. One popular such scheme is called *voting*. Voting is a very simple ad hoc scheme, but works reasonably well in practice. In voting, all the rules that match a new case get to participate in an election process to determine its outcome. If no rules fire, a default outcome is used. Otherwise, in the election, each matching rule casts a number of votes proportional to its support in favor of the outcome that the rule indicates. All the votes are then accumulated, and the possible outcomes can be ranked according to the percentage of votes they got. This associated measure of certainty for each possible outcome is not really a probability, but may be interpreted as an approximation to probability if the model is well calibrated.

3.3 Model Evaluation

A decision rule is straightforward to interpret, but to objectively quantify the degree of "interestingness" of a rule is difficult since this is dependent on the domain of the data. Hence, any formal and automatic evaluation along this "explanatory dimension" is rarely done. However, evaluating the performance of an ensemble of decision rules as a classifier of new cases can be done using standard performance measures. In medicine, measures derived from *receiver operating characteristic* (ROC) curves [8] are common for binary outcomes.

A ROC curve is a graphical method for assessing the discriminatory performance of a binary classifier, independent of both error costs and the prevalence of disease. If, relating to Table 1, a classifier outputs a measure of certainty that a patient may walk, we can vary a decision threshold across the full spectrum of possible values and obtain several pairs of estimates for the sensitivity (true positive rate) and the specificity (true negative rate) of the classifier. A ROC curve is a plot of the complement of the specificity (the false positive rate) on the x-axis against the sensitivity on the y-axis. The area under the ROC curve computed using the trapezoidal method of integration can be shown to equal the Wilcoxon-Mann-Whitney statistic, or the probability that the classifier will assign a higher value to a walking individual than to a non-walking one, if the pair is randomly drawn from the population the ROC curve is derived from. An area of 0.5 signifies that the classifier performs no better than tossing a coin, while an area of 1.0 signifies perfect discrimination. Usually, the area lies somewhere between these two extremes[2].

[2]Theoretically, the area can take on a value between 0.0 and 0.5, too. However, an area of A is for all practical purposes just as good as an area of $1 - A$, since we can obtain the latter simply by selecting the opposite outcome of what the classifier suggests.

4 Mining a Spinal Cord Injury Database

This section gives a brief example of the relative performance of rough sets when applied to a subset of 1138 patients from the Spinal Cord Injury Model System (SCIMS) database.

4.1 Data Material and Model Performance

On the basis of data available at the time of admission for spinal cord injury patients, our group was to induce rules that would predict the ambulatory status of the patient (a binary outcome) at the time of discharge. The prior probability of walking was approximately 20%. Input variables to the model were upper and lower extremity motor scores (UEMS and LEMS), days from the creation of the database until the date of admission, days from injury to system admission, age, gender, racial/ethnic group, left and right level of preserved neurologic function, and ASIA impairment score. A rationale for this study can be found in [9], along with a detailed account of a superset of the data material presently considered.

The data set was randomly split into a training set with 762 cases and a test set with 376 cases. Initially, a discretization algorithm [7] discussed in Section 3.1 was applied to the training set to automatically search for cut-off points, and then both data sets were discretized using these. Rules were then mined from the training set using dynamic reducts [5]. All computations were performed using the ROSETTA software system [10, 11].

Applying the rules to the test set resulted in an area under ROC curve of 91.4%, with a standard error of 2.5%. Thus, the rules function as an excellent discriminator between walking and non-walking patients. Using Hanley-McNeil's test [12], no statistically significant differences could be detected between the performance of the rough set classifier versus models built using logistic regression ($p < 0.167$) or neural networks ($p < 0.229$). For reference, a logistic regression model obtained an ROC area of 92.5% (1.6%) while a neural network model achieved an ROC area of 92.3% (1.5%).

The two attributes LEMS and ASIA score stood out as the clearly best univariate classifiers. LEMS alone could account for an ROC area of 87.0% on the training set and 89.1% on the test set, while the ASIA score managed 85.8% on the training set and 86.7% on the test set. The UEMS attribute was the third best univariate classifier, managing 63.0% on the training set and 70.0% on the test set. The univariate performance of all the other attributes ranged from 50 to 60%.

4.2 Inspecting the Model

An appealing characteristic of rule-based classifiers is that the induced models can be inspected and interpreted without any expert knowledge about the underlying model induction technique. This section takes a closer look at some aspects of the model induced from the SCIMS data.

The initial discretization step may potentially suggest interesting cut-off points, if none are known beforehand. For example, the cut-off points found for the LEMS, Age and ASIA impairment score variables define the ranges given in Table 2. In this study, for the sake of simplicity, all the suggested ranges were used. Alterna-

LEMS	Age	ASIA
$\leq$2	$\leq$25	A-B
3-12	26-37	C
13	38-44	D-E
14-22	45-59	
23-32	$\geq$60	
$\geq$33		

Table 2: Automatically suggested ranges for the attributes LEMS, Age and ASIA impairment score. These suggestions can either be used directly, or form the basis for a manually constructed set of cut-off points

tively, they could have formed the basis for a manually constructed set of cut-off points. Obviously, the discretization step may have a significant impact on the interpretability and the performance of the induced rules. Different cut-off points may yield slightly different results.

The notions of accuracy and coverage were discussed in Section 2.5. The rule that has the highest coverage and that predicts walking states that patients that are admitted to a SCI care center within one day and that have a high LEMS are likely to walk. Furthermore, that description fits over one-third of all walking patients. The rule that has the highest coverage and that predicts non-walking is a univariate rule stating that if a patient has a very low LEMS then he/she will very likely not walk. Over three-fourths of all non-walking patients fit that description. This confirms the known fact that LEMS is a good predictor of ambulation, as verified quantitatively in Section 4.1.

if	LEMS is $\geq$33 and
	Days from injury to admission is $\leq$1
then	Walk is Yes
with	Accuracy 75.4% and
	Coverage 34.0%

if	LEMS is $\leq$2
then	Walk is No
with	Accuracy 96.4% and
	Coverage 77.2%

A set of decision rules may be queried to look for interactions between particular attributes of interest. If we suspect that a certain combination of attributes interrelate in an important manner (because, say, a logistic regression model suggests that an interaction term should be included), the rule set could be scanned (or rules generated on the fly) to yield a readable and interpretable account of the nature of the interaction. Although not done here, the ROSETTA software system allows such queries to be made.

This section has discussed interpretation of the model whose purpose was to predict ambulation. If the purpose of the study had been knowledge discovery, then it is likely that the data material would have been different. For example, we could have focused on inducing rules from particularly interesting subgroups only.

The full set of rules mined from the SCIMS database counts 10172 rules, and inspection of such a voluminous model requires sorting and querying the set of rules in order to view selected aspects of the model. However, experiments indicate that rule

sets can often be pruned down by several orders of magnitude and still retain acceptable discriminatory performance. For instance, in [13], models mined from the SCIMS database that originally comprised almost 35000 rules were pruned down to between 5 and 15 rules with a drop in the ROC area of only a few percentage points. Such pruned models would be significantly easier to interpret as a whole.

5 Summary

This article has given an introductory overview to the field of rough sets, a recent technique for data mining and knowledge discovery from databases. Section 2 outlined the general conceptual framework, and gave a hint of the Boolean minimization techniques involved. The interested reader is referred to the literature [3, 4] for a detailed and more technical exposition of the matter. Briefly, the presented aspects can be summarized as follows:

1. Patients, represented as rows in a data table, can be grouped together according to a notion of being "equal". Each group is called an equivalence class. The patients in each group cannot be discerned from one another. We can define indiscernibility to be absolute or relative, depending on what we want to achieve. If two patients belong to the same group, we say that these two patients are related to each other by a mathematical structure called an indiscernibility relation.

2. The groups or equivalence classes define "atoms" or basic building blocks that we can put together to assemble larger groups or sets of patients. Typically, we want to assemble sets of patients that all have the same outcome. We may not be able to do this precisely, but we can assemble a pair of sets such that our "goal" set "lies between" these two assembled sets. Our pair of assembled sets is said to constitute a rough approximation of our goal set.

3. Indiscernibility, and hence our ability to form set approximations, depends on which columns or attributes in the data table that we consider. We are typically interested in reducing the information down to minimal sets of attributes, called reducts. A reduct preserves the indiscernibility relation, i.e., enables us to form approximations equally well as if we had used all attributes available. Computation of reducts can be done by minimizing suitably formulated Boolean expressions.

4. Minimal if-then rules, in which functionally redundant information has been discarded, are obtained by computing reducts and reading off the corresponding values in the data table.

In Section 3, the main steps involved in a rough set analysis, preprocessing of data, and model performance evaluation were discussed. A broader, non-technical review of the knowledge discovery process can be found in [1].

In Section 4, an example application for the prediction of ambulation following spinal cord injury using a real-world database was analyzed. ROSETTA, the software system for data mining and knowledge discovery based on rough set theory employed in Section 4, is publicly available:

http://www.idi.ntnu.no/~aleks/rosetta/

Acknowledgments

Thanks to Lucila Ohno-Machado and Tor-Kristian Jenssen for commenting on a draft version of this paper. Thanks also to the anonymous reviewers for valuable suggestions that helped improve this article.

This work was supported in part by grant 74467/410 from the Norwegian Research Council, grant H133N50015 from the National Institute on Disability and Rehabilitation Research and grant R55LM/OD6538-01 from the National Library of Medicine.

References

[1] Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. The KDD process for extracting useful knowledge from volumes of data. *Communications of the ACM*, 39(11):27–34, November 1996.

[2] Zdzisław Pawlak. Rough sets. *International Journal of Information and Computer Science*, 11(5):341–356, 1982.

[3] Zdzisław Pawlak. *Rough Sets: Theoretical Aspects of Reasoning about Data*, volume 9 of *Series D: System Theory, Knowledge Engineering and Problem Solving*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1991.

[4] Frank Markham Brown. *Boolean Reasoning: The Logic of Boolean Equations*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1990.

[5] Jan G. Bazan. A comparison of dynamic and non-dynamic rough set methods for extracting laws from decision tables. In Polkowski and Skowron [15], chapter 17, pages 321–365.

[6] J. Dougherty, R. Kohavi, and M. Sahami. Supervised and unsupervised discretization of continuous features. In A. Prieditis and S. Russell, editors, *Proc. Twelfth International Conference on Machine Learning*, pages 194–202. Morgan Kaufmann, 1995.

[7] Hung Son Nguyen and Andrzej Skowron. Quantization of real-valued attributes. In *Proc. Second International Joint Conference on Information Sciences*, pages 34–37, Wrightsville Beach, NC, September 1995.

[8] James A. Hanley and Barbara J. McNeil. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143:29–36, April 1982.

[9] Todd Rowland, Lucila Ohno-Machado, and Aleksander Øhrn. Comparison of multiple prediction models for ambulation following spinal cord injury. In Chute [14], pages 528–532.

[10] Aleksander Øhrn, Jan Komorowski, Andrzej Skowron, and Piotr Synak. The design and implementation of a knowledge discovery toolkit based on rough sets: The ROSETTA system. In Polkowski and Skowron [15], chapter 19, pages 376–399.

[11] Aleksander Øhrn, Jan Komorowski, Andrzej Skowron, and Piotr Synak. The ROSETTA software system. In Polkowski and Skowron [16], pages 572–576.

[12] James A. Hanley and Barbara J. McNeil. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. *Radiology*, 148:839–843, September 1983.

[13] Aleksander Øhrn, Lucila Ohno-Machado, and Todd Rowland. Building manageable rough set classifiers. In Chute [14], pages 543–547.

[14] Christoper G. Chute, editor. *Proceedings AMIA 1998 Annual Symposium*, Orlando, FL, November 1998. Supplement to Journal of the American Medical Informatics Association, Hanley & Belfus, Inc.

[15] Lech Polkowski and Andrzej Skowron, editors. *Rough Sets in Knowledge Discovery 1: Methodology and Applications*, volume 18 of *Studies in Fuzziness and Soft Computing*. Physica-Verlag, Heidelberg, Germany, 1998.

[16] Lech Polkowski and Andrzej Skowron, editors. *Rough Sets in Knowledge Discovery 2: Applications, Case Studies and Software Systems*, volume 19 of *Studies in Fuzziness and Soft Computing*. Physica-Verlag, Heidelberg, Germany, 1998.