ELSEVIER

European Journal of Operational Research 136 (2002) 181–189

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

www.elsevier.com/locate/dsw

Computing, Artificial Intelligence and Information Technology

Rough sets, decision algorithms and Bayes' theorem

Zdzisław Pawlak *

Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, ul. Bałtycka 5, 44 100 Gliwice, Poland

Received 14 September 1999; accepted 20 December 2000

Abstract

Rough set-based data analysis starts from a data table, called an *information system*. The information system contains data about objects of interest characterized in terms of some attributes. Often we distinguish in the information system condition and decision attributes. Such information system is called a *decision table*. The decision table describes decisions in terms of conditions that must be satisfied in order to carry out the decision specified in the decision table. With every decision table a set of decision rules, called a *decision algorithm*, can be associated. It is shown that every decision algorithm reveals some well-known probabilistic properties, in particular it satisfies the total probability theorem and Bayes' theorem. These properties give a new method of drawing conclusions from data, without referring to prior and posterior probabilities, inherently associated with Bayesian reasoning.

Keywords: Rough sets; Decision analysis; Decision support systems; Bayes' theorem

1. Introduction

The paper concerns a relationship between rough sets and Bayes' theorem. It reveals a new look on Bayes' theorem from the rough set perspective and is a continuation of ideas presented in [4,5].

In the paper basic notions of the rough set theory will be given, together with the notion of the decision algorithm for which some properties will be shown. It is revealed, in particular, that every decision table (decision algorithm) displays well-known probabilistic features, in particular it satisfies the total probability theorem and Bayes' theorem. These properties give a new method of drawing conclusions from data, without referring to prior and posterior probabilities, inherently associated with Bayesian reasoning.

The revealed relationship can be used to invert decision rules, i.e., giving reasons (explanations) for decisions, which can be very useful in decision analysis.

Statistical inference based on Bayes' theorem is used to verify prior knowledge when the data become available, whereas rough set inference based

* Address: ul. Zuga 29, 01 806 Warsaw, Poland. Tel.: +48-22-8345659; fax: +48-22-8251635.

E-mail address: zpw@ii.pw.edu.pl (Z. Pawlak).

0377-2217/02/$ - see front matter
PII: S0377-2217(01)00029-7

on Bayes' theorem uses relationships in the data revealed by Bayes' theorem.

In other words, rough set view on Bayes' theorem explains the relationship between conditions and decisions in decision rules, without referring either to prior or posterior probabilities.

Basis of the rough set theory can be found in [3]. More advanced topics are discussed in [6,7].

2. Approximation of sets

A starting point of rough set-based data analysis is a data set, called an information system.

An information system is a data table, whose columns are labeled by attributes, rows are labeled by objects of interest and entries of the table are attribute values.

Formally, by an *information system* we will understand a pair $S = (U, A)$, where U and A, are finite, nonempty sets called the *universe*, and the set of *attributes,* respectively. 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 defined as follows: $(x, y) \in I(B)$ if and only if $a(x) = a(y)$ 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., a partition determined by B, will be denoted by $U/I(B)$, or simply by 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* (*indiscernible with respect to B*). Equivalence classes of the relation $I(B)$ (or blocks of the partition U/B) are referred to as *B-elementary sets* or *B-granules*.

If we distinguish in an information system two disjoint classes of attributes, called *condition* and *decision attributes*, respectively, then the system will be called a *decision table* and will be denoted by $S = (U, C, D)$, where C and D are disjoint sets of condition and decision attributes, respectively.

Suppose we are given an information system $S = (U, A)$, $X \subseteq U$, and $B \subseteq A$. Our task is to describe the set X in terms of attribute values from B. To this end we define two operations assigning to every $X \subseteq U$ two sets $B_*(X)$ and $B^*(X)$ called the *B-lower* and the *B-upper approximation* of X, respectively, and defined as follows:

$$B_*(X) = \bigcup_{x \in U} \{B(x) : B(x) \subseteq X\},$$

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

Hence, the *B-lower approximation* of a set is the union of all B-granules that are included in the set, whereas the *B-upper approximation* of a set is the union of all B-granules that have a nonempty intersection with the set. The set

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

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 X is *crisp* (*exact*) with respect to B; in the opposite case, i.e., if $BN_B(X) \neq \emptyset$, X is referred to as *rough* (*inexact*) with respect to B.

3. Decision rules

In this section we will introduce a formal language to describe approximations in logical terms.

Let $S = (U, A)$ be an information system. With every $B \subseteq A$ we associate a formal language, i.e., a set of formulas $For(B)$. Formulas of $For(B)$ are built up from attribute–value pairs (a, v) where $a \in B$ and $v \in V_a$ by means of logical connectives $\wedge$ (*and*), $\vee$ (*or*), $\sim$ (*not*) in the standard way.

For any $\Phi \in For(B)$ by $||\Phi||_S$ we denote the set of all objects $x \in U$ satisfying Φ in S and refer to as the *meaning* of Φ in S.

The meaning $||\Phi||_S$ of Φ in S is defined inductively as follows:

$$||(a, v)||_S = \{x \in U : a(v) = x\} \quad \forall a \in B, v \in V_a,$$

$$||\Phi \vee \Psi||_S = ||\Phi||_S \cup ||\Psi||_S,$$

$$||\Phi \wedge \Psi||_S = ||\Phi||_S \cap ||\Psi||_S,$$

$$|| \sim \Phi||_S = U - ||\Phi||_S.$$

A *decision rule* in S is an expression $\Phi \to \Psi$, read *if* Φ *then* Ψ, where $\Phi \in For(C)$, $\Psi \in For(D)$ and C and D are condition and decision attributes, respectively; Φ and Ψ are referred to as *conditions* and *decisions* of the rule, respectively.

The number $\mathrm{supp}_S(\Phi, \Psi) = \mathrm{card}(||\Phi \wedge \Psi||_S)$ will be called the *support* of the rule $\Phi \to \Psi$ in S. We consider a probability distribution $p_U(x) = 1/\mathrm{card}(U)$ for $x \in U$ where U is the (non-empty) universe of objects of S; we have $p_U(X) = \mathrm{card}(X)/\mathrm{card}(U)$ for $X \subseteq U$. For any formula Φ we associate its probability in S defined by

$$\pi_S(\Phi) = p_U(||\Phi||_S).$$

With every decision rule $\Phi \to \Psi$ we associate a conditional probability

$$\pi_S(\Psi \mid \Phi) = p_U(\|\Psi\|_S \mid \|\Phi\|_S)$$

called the *certainty factor* of the decision rule, denoted $\mathrm{cer}_S(\Phi, \Psi)$. This idea was used first by Łukasiewicz [2] (see also [1]) to estimate the probability of implications. We have

$$\begin{aligned}\mathrm{cer}_S(\Phi, \Psi) &= \pi_S(\Psi \mid \Phi) \\ &= \mathrm{card}(||\Phi \wedge \Psi||_S)/\mathrm{card}(||\Phi||_S),\end{aligned}$$

where $||\Phi||_S \neq \emptyset$.

This coefficient is now widely used in data mining and is called *confidence coefficient*.

Obviously, $\pi_S(\Psi \mid \Phi) = 1$ if and only if $\Phi \to \Psi$ is true in S.

If $\pi_S(\Psi \mid \Phi) = 1$, then $\Phi \to \Psi$ will be called a *certain decision* rule; if $0 < \pi_S(\Psi \mid \Phi) < 1$ the decision rule will be referred to as a *uncertain decision* rule.

Besides, we will also use a *coverage factor* of the decision rule, denoted $\mathrm{cov}_S(\Phi, \Psi)$ (used e.g. by Tsumoto [9] for estimation of the quality of decision rules) defined by

$$\pi_S(\Phi|\Psi) = p_U(\|\Phi\|_S|\|\Psi\|_S).$$

Obviously, we have

$$\begin{aligned}\mathrm{cov}_S(\Phi, \Psi) &= \pi_S(\Phi \mid \Psi) \\ &= \mathrm{card}(||\Phi \wedge \Psi||_S)/\mathrm{card}(||\Psi||_S).\end{aligned}$$

There are three possibilities to interpret the certainty and the coverage factors: statistical (frequency), logical (degree of truth) and mereological (degree of inclusion).

We will stick here to the statistical interpretation, i.e., the certainty factors will be interpreted as the frequency of objects having the property Ψ in the set of objects having the property Φ and the coverage factor – as the frequency of objects having the property Φ in the set of objects having the property Ψ. The number

$$\begin{aligned}\sigma_S(\Phi, \Psi) &= \mathrm{supp}_S(\Phi, \Psi)/\mathrm{card}(U) \\ &= \pi_S(\Psi \mid \Phi)\,\pi_S(\Phi)\end{aligned}$$

will be called the *strength* of the decision rule $\Phi \to \Psi$ in S.

The certainty and the coverage factors of decision rules express how exact is our knowledge (data) about the considered reality. Let us remain that the factors are not assumed to be arbitrary but are computed from the data, thus there are in a certain sense objective.

The certainty factor reveals simply the frequency of facts satisfying conditions, among the facts satisfying decision of the decision rule, whereas the interpretation of the coverage factor is converse.

Finally, let us briefly comment on the concept of the strength of a decision rule. This number simply expresses the ratio of all facts which can be classified by the decision rule to all facts in the data table. It will be shown in the next sections that this coefficient plays an essential role in further considerations, and will be used in the new formulation of Bayes' theorem.

4. Decision algorithms

In this section we define the notion of a decision algorithm, which is a logical counterpart of a decision table.

Let $\mathrm{Dec}(S) = \{\Phi_i \to \Psi_i\}_{i=1}^m$, $m \geqslant 2$, be a set of decision rules in a decision table $S = (U, C, D)$.

(1) If for every $\Phi \to \Psi$, $\Phi' \to \Psi' \in \mathrm{Dec}(S)$ we have $\Phi = \Phi'$ or $||\Phi \wedge \Phi'||_S = \emptyset$, and $\Psi = \Psi'$ or

$||\Psi \wedge \Psi'||_S = \emptyset$, then we will say that $\mathrm{Dec}(S)$ is the set of pairwise *mutually exclusive* (*independent*) decision rules in S.

(2) If

$$\left|\left|\bigvee_{i=1}^{m} \Phi_i\right|\right|_S = U,$$

and

$$\left|\left|\bigvee_{i=1}^{m} \Psi_i\right|\right|_S = U,$$

we will say that the set of decision rules $\mathrm{Dec}(S)$ *covers* U.

(3) If $\Phi \to \Psi \in \mathrm{Dec}(S)$ and $\mathrm{supp}_S(\Phi, \Psi) \neq 0$, we will say that the decision rule $\Phi \to \Psi$ is *admissible* in S.

(4) If

$$\bigcup_{X \in U/D} C_*(X) = \left|\left| \bigvee_{\Phi \to \Psi \in \mathrm{Dec}^+(S)} \Phi \right|\right|_S,$$

where $\mathrm{Dec}^+(S)$ is the set of all certain decision rules from $\mathrm{Dec}(S)$, we will say that the set of decision rules $\mathrm{Dec}(S)$ preserves the *consistency* of the decision table $S = (U, C, D)$.

The set of decision rules $\mathrm{Dec}(S)$ that satisfies (1)–(4), i.e., is independent, covers U, preserves the consistency of S and all decision rules $\Phi \to \Psi \in \mathrm{Dec}(S)$ are admissible in S – will be called a *decision algorithm* in S.

Hence, if $\mathrm{Dec}(S)$ is a decision algorithm in S then the conditions of rules from $\mathrm{Dec}(S)$ define in S a partition of U. Moreover, the *positive region of D with respect to C*, i.e., the set

$$\bigcup_{X \in U/D} C_*(X)$$

is partitioned by the conditions of some of these rules, which are certain in S.

If $\Phi \to \Psi$ is a decision rule, then the decision rule $\Psi \to \Phi$ will be called an *inverse* decision rule of $\Phi \to \Psi$.

Let $\mathrm{Dec}^*(S)$ denote the set of all inverse decision rules of $\mathrm{Dec}(S)$.

It can be shown that $\mathrm{Dec}^*(S)$ satisfies (1)–(4), i.e., it is a decision algorithm in S.

If $\mathrm{Dec}(S)$ is a decision algorithm, then $\mathrm{Dec}^*(S)$ will be called an *inverse* decision algorithm of $\mathrm{Dec}(S)$.

The inverse decision algorithm gives *reasons* (*explanations*) for decisions pointed out by the decision algorithms. The number

$$\eta(\mathrm{Dec}(S)) = \sum_{\Phi \to \Psi \in \mathrm{Dec}(S)} \max\{\sigma_S(\Phi, \Psi)\}_{\Psi \in D(\Phi)},$$

where $D(\Phi) = \{\Psi : \Phi \to \Psi \in \mathrm{Dec}(S)\}$ will be referred to as the *efficiency* of the decision algorithm $\mathrm{Dec}(S)$ in S, and the sum is stretching over all decision rules in the algorithm.

The efficiency of a decision algorithm is the probability (ratio) of all objects of the universe, that are classified to decision classes, by means of decision rules $\Phi \to \Psi$ with maximal strength $\sigma_S(\Phi, \Psi)$ among rules $\Phi \to \Psi \in \mathrm{Dec}(S)$ with satisfied Φ on these objects. In other words, the efficiency says how well the decision algorithm classifies objects from S when the decision rules with maximal strength are used only.

As mentioned at the beginning of this section decision algorithm is a counterpart of a decision table. The properties (1)–(4) have been chosen in such a way that the decision algorithm preserves the basic properties of the data in the decision table, in particular approximations and boundary regions of decisions.

Crucial issue in the rough set-based data analysis is the generation of decision algorithms from the data. This is a complex task, particularly when large databases are concerned. Many methods and algorithms have been proposed to deal with this problem but we will not dwell upon this issue here, for we intend to restrict this paper to rudiments of the rough set theory only. The interested reader is advised to consult the Refs. [6,7] and the web.

5. Decision algorithms and approximations

Decision algorithms can be used as a formal language for describing approximations.

Let $\mathrm{Dec}(S)$ be a decision algorithm in S and let $\Phi \to \Psi \in \mathrm{Dec}(S)$. By $C(\Psi)$ we denote the set of all conditions of Ψ in $\mathrm{Dec}(S)$ and by $D(\Phi)$ – the set of all decisions of Φ in $\mathrm{Dec}(S)$.

Then we have the following relationships:

$$\text{(a)}\quad C_*(||\Psi||_S) = \left|\left| \bigvee_{\Phi' \in C(\Psi), \pi(\Psi|\Phi')=1} \Phi' \right|\right|_S,$$

$$\text{(b)}\quad C^*(||\Psi||_S) = \left|\left| \bigvee_{\Phi' \in C(\Psi), 0<\pi(\Psi|\Phi') \leqslant 1} \Phi' \right|\right|_S,$$

$$\text{(c)}\quad BN_C(||\Psi||_S) = \left|\left| \bigvee_{\Phi' \in C(\Psi), 0<\pi(\Psi|\Phi')<1} \Phi' \right|\right|_S.$$

From the above properties we can get the following definitions:

1. If $||\Phi||_S = C_*(||\Psi||_S)$, then formula Φ will be called the *C-lower approximation* of the formula Ψ and will be denoted by $C_*(\Psi)$;
2. If $||\Phi||_S = C^*(||\Psi||_S)$, then the formula Φ will be called the *C-upper approximation* of the formula Φ and will be denoted by $C^*(\Psi)$;
3. If $||\Phi||_S = BN_C(||\Psi||_S)$, then Φ will be called the *C-boundary* of the formula Ψ and will be denoted by $BN_C(\Psi)$.

The above properties say that any decision $\Psi \in \mathrm{Dec}(S)$ can be uniquely described by the following certain and uncertain decision rules, respectively:

$$C_*(\Psi) \to \Psi,$$

$$BN_C(\Psi) \to \Psi.$$

This property is an extension of some ideas given by Ziarko [10].

6. Some properties of decision algorithms

Decision algorithms have interesting probabilistic properties which are discussed in this section.

Let $\mathrm{Dec}(S)$ be a decision algorithm and let $\Phi \to \Psi \in \mathrm{Dec}(S)$. Then the following properties are valid:

$$(1)\quad \sum_{\Phi' \in C(\Psi)} \mathrm{cer}_S(\Phi', \Psi) = 1,$$

$$(2)\quad \sum_{\Psi' \in D(\Phi)} \mathrm{cov}_S(\Phi, \Psi') = 1,$$

$$(3)\quad \pi_S(\Psi) = \sum_{\Phi' \in C(\Psi)} \mathrm{cer}_S(\Phi', \Psi)\, \pi_S(\Phi') = \sum_{\Phi' \in C(\Psi)} \sigma_S(\Phi', \Psi),$$

$$(4)\quad \pi_S(\Phi) = \sum_{\Psi' \in D(\Phi)} \mathrm{cov}_S(\Phi, \Psi')\, \pi_S(\Psi') = \sum_{\Psi' \in D(\Phi)} \sigma_S(\Phi, \Psi'),$$

$$(5)\quad \mathrm{cer}_S(\Phi, \Psi) = \mathrm{cov}_S(\Phi, \Psi)\, \pi_S(\Psi) \Big/ \sum_{\Psi' \in D(\Phi)} \times \mathrm{cov}_S(\Phi, \Psi')\, \pi_S(\Psi') = \sigma_S(\Phi, \Psi) \Big/ \sum_{\Psi' \in D(\Phi)} \sigma_S(\Phi, \Psi') = \sigma_S(\Psi, \Phi)/\pi_S(\Phi),$$

$$(6)\quad \mathrm{cov}_S(\Phi, \Psi) = \mathrm{cer}_S(\Phi, \Psi)\, \pi_S(\Phi) \Big/ \sum_{\Phi' \in C(\Psi)} \times \mathrm{cer}_S(\Phi', \Psi)\, \pi_S(\Phi') = \sigma_S(\Phi, \Psi) \Big/ \sum_{\Phi' \in C(\Psi)} \sigma_S(\Phi', \Psi) = \sigma_S(\Phi, \Psi)/\pi_S(\Psi).$$

That is, any decision algorithm, and consequently any decision table, satisfies (1)–(6). Observe that (3) and (4) refer to the well-known *total probability theorem*, whereas (5) and (6) refer to *Bayes' theorem*. Note that we are not using prior and posterior probabilities – fundamental in the Bayesian data analysis philosophy.

Thus in order to compute the certainty and coverage factors of decision rules according to formula (5) and (6) it is enough to know the strength (support) of all decision rules in the decision algorithm only. The strength of decision rules can be computed from the data or can be a subjective assessment.

In other words, if we know the ratio of Φ_S in Ψ, thanks to Bayes' theorem, we can compute the ratio of Ψ_S in Φ.

7. Total probability theorem and inference rules

It is interesting to observe that the total probability theorem is closely related with *modus ponens* (MP) and *modus tollens* (MT) inference rules.

MP has the following form:

if	$\Phi \to \Psi$	is true
and	Φ	is true
then	Ψ	is true

If we replace truth values by corresponding probabilities we can generalize the inference rule as rough modus ponens (RMP), which has the form

if	$\Phi \to \Psi$	is true with probability $\mathrm{cer}_S(\Phi, \Psi)$
and	Φ	is true with probability $\pi_S(\Phi)$
then	Ψ	is true with probability

$$\pi_S(\Psi) = \sum_{\Phi' \in C(\Psi)} \mathrm{cer}_S(\Phi', \Psi)\, \pi_S(\Phi') = \sum_{\Phi' \in C(\Psi)} \sigma_S(\Phi', \Psi).$$

RMP enables us to calculate the probability of conclusion Ψ of a decision rule $\Phi \to \Psi$ in terms of strengths of all decision rules in the form $\Phi' \to \Psi, \Phi' \in C(\Psi)$.

MT inference rule is as follows:

if	$\Phi \to \Psi$	is true
and	$\sim \Psi$	is true
then	$\sim \Phi$	is true

Similarly, if we replace truth values by probabilities we get the following rough modus tollens (RMT) inference rule

if	$\Phi \to \Psi$	is true with probability $\mathrm{cov}_S(\Phi, \Psi)$
and	Ψ	is true with probability $\pi_S(\Psi)$
then	Ψ	is true with probability

$$\pi_S(\Phi) = \sum_{\Psi' \in D(\Phi)} \mathrm{cov}_S(\Phi, \Psi')\, \pi_S(\Psi') = \sum_{\Psi' \in D(\Phi)} \sigma_S(\Phi', \Psi).$$

RMT enables us to compute the probability of condition Φ of the decision rule $\Phi \to \Psi$ in terms of strengths of all decision rules in the form $\Phi \to \Psi'$, $\Psi' \in D(\Phi)$.

Let us notice that RMP and RMT are rather formal generalizations of MP and MT, respectively, because the role of inference rules in logic is different from that of decision rules in data analysis.

MP and MT are used to draw conclusions from logical axioms, whereas RMP and RMT are used to compute probabilities of decisions (conditions) in decision tables (decision algorithms).

Discussion of the relationship between logic and probability can be also found in [1,2].

8. Illustrative examples

In this section we will illustrate the concepts introduced previously by means of simple examples.

Example 1. In Table 1 information about nine-hundred people is represented. The population is characterized by the following attributes: *Height*, *Hair*, *Eyes* and *Nationality*.

Suppose that *Height*, *Hair* and *Eyes* are condition attributes and *Nationality* is the decision attribute, i.e., we want to find description of each nationality in terms of condition attributes.

Below a decision algorithm associated with Table 1 is given:

Table 1
Characterization of nationalities

U	*Height*	*Hair*	*Eyes*	*Nationality*	Support
1	*tall*	*blond*	*blue*	*Swede*	270
2	*medium*	*dark*	*hazel*	*German*	90
3	*medium*	*blond*	*blue*	*Swede*	90
4	*tall*	*blond*	*blue*	*German*	360
5	*short*	*red*	*blue*	*German*	45
6	*medium*	*dark*	*hazel*	*Swede*	45

(1) if (*Height, tall*) then (*Nationality, Swede*),
(2) if (*Height, medium*) and (*Hair, dark*) then (*Nationality, German*),
(3) if (*Height, medium*) and (*Hair, blond*) then (*Nationality, Swede*),
(4) if (*Height, tall*) then (*Nationality, German*),
(5) if (*Height, short*) then (*Nationality, German*),
(6) if (*Height, medium*) and (*Hair, dark*) then (*Nationality, Swede*).

The certainty and coverage factors for the decision rules are shown in Table 2.

From the certainty factors of the decision rules we can conclude that:

- 43% *tall* people are *Swede*,
- 57% *tall* people are *German*,
- 33% *medium* and *dark-haired* people are *Swede*,
- 67% *medium* and *dark-haired* people are *German*,
- 100% *medium* and *blond* people are *Swede*,
- 100% *short* people are *German*.

Summing up:

- *tall* people are most probably *German*,
- *medium* and *dark-haired* people are most probably *German*,
- *medium* and *blond* people are for certain *Swede*,
- *short* people are for certain *German*.

The efficiency of the above decision algorithm is 0.65.

The inverse algorithm is as follows

(1′) if (*Nationality, Swede*) then (*Height, tall*),
(2′) if (*Nationality, German*) then (*Height, medium*) and (*Hair, dark*),
(3′) if (*Nationality, Swede*) then (*Height, medium*) and (*Hair, blond*),
(4′) if (*Nationality, German*) then (*Height, tall*),
(5′) if (*Nationality, German*) then (*Height, short*),
(6′) if (*Nationality, Swede*) then (*Height, medium*) and (*Hair, dark*).

Table 2
Certainty and coverage factors

Rule number	Certainty	Coverage	Support	Strength
1	0.43	0.67	270	0.3
2	0.67	0.18	90	0.1
3	1.00	0.22	90	0.1
4	0.57	0.73	360	0.4
5	1.00	0.09	45	0.05
6	0.33	0.11	45	0.05

The certainty and the coverage factors for the decision rules are shown in Table 2.

From the coverage factors we get the following characterization of nationalities:

- 11% *Swede* are *medium* and *dark-haired*,
- 22% *Swede* are *medium* and *blond*,
- 67% *Swede* are *tall*,
- 9% *German* are *short*,
- 18% *German* are *medium* and *dark-haired*,
- 73% *German* are *tall*.

Hence we have that:

- *Swede* are most probably *tall*,
- *German* are most probably *tall*.

The efficiency of the inverse decision algorithm is 0.7.

Observe that there are no certain decision rules in the inverse decision algorithm nevertheless it can properly classify 70% objects.

Of course it is possible to find another decision algorithm from Table 1.

The obtained results are valid for the data only. In the case of another bigger data set the results may not be valid anymore.

Whether they are valid or not it depends if Table 1 is a representative sample of a bigger population or not.

Example 2. Now we will consider an example taken from [8], which will show clearly the difference between the Bayesian and rough set approach to data analysis.

We will start from the data table presented below:

Remark. In the paper [8] wrongly $1 = low$ and $3 = high$ instead of $1 = high$ and $3 = low$.

We have to classify voters according to their voting intentions on the basis of Sex and Social Class.

First we create from Table 3 a decision table shown in Table 4.

Next we simplify the decision table by employing only the decision rules with maximal

Table 3
Voting intentions[a]

Y_2	Y_3	Y_1			
		1	2	3	4
1	1	28	8	7	0
	2	153	114	53	14
	3	20	31	17	1
2	1	1	1	0	1
	2	165	86	54	6
	3	30	57	18	4

[a] Y_1 represents voting intentions (1 = *Conservatives*, 2 = *Labour*, 3 = *Liberal Democrat*, 4 = *Others*), Y_2 represents Sex (1 = *male*, 2 = *female*) and Y_3 represents Social Class (1 = *high*, 2 = *middle*, 3 = *low*).

Table 4
Decision table

U	Y_2	Y_3	Y_1	Support	Strength
1	1	1	1	28	0.03
2	1	1	2	8	0.01
3	1	1	3	7	0.01
4	1	2	1	153	0.18
5	1	2	2	114	0.13
6	1	2	3	53	0.06
7	1	2	4	14	0.02
8	1	3	1	20	0.02
9	1	3	2	31	0.04
10	1	3	3	17	0.02
11	1	3	4	1	0.00
12	2	1	1	1	0.00
13	2	1	2	1	0.00
14	2	1	4	1	0.00
15	2	2	1	165	0.19
16	2	2	2	86	0.10
17	2	2	3	54	0.06
18	2	2	4	6	0.01
19	2	3	1	30	0.03
20	2	3	2	57	0.07
21	2	3	3	18	0.02
22	2	3	4	4	0.00

strength, and we get the decision table presented in Table 5.

It can be easily seen that the set of condition attributes can be reduced (see [3]) and the only reduced is the attribute Y_3 (Social Class).

Thus Table 5 can be replaced by Table 6. The numbers in parentheses refer to Table 4.

Table 5
Simplified decision table

U	Y_2	Y_3	Y_1	Support	Strength
1	1	1	1	28	0.07
2	1	2	1	153	0.35
3	1	3	2	31	0.07
4	2	2	1	165	0.38
5	2	3	2	57	0.13

Table 6
Reduced decision table

U	Y_3	Y_1	Strength	Certainty	Coverage
1	1	1	0.07 (0.03)	1.00 (0.60)	0.10 (0.07)
2	2	1	0.73 (0.37)	1.00 (0.49)	0.90 (0.82)
3	3	2	0.20 (0.11)	1.00 (0.55)	1.00 (0.31)

From this decision table we get the following decision algorithm:

		Certainty
(1)	*high class* → *Conservative party*	0.60
(2)	*middle class* → *Conservative party*	0.49
(3)	*lower class* → *Labour party*	0.55

The efficiency of the decision algorithm is 0.51.

The inverse decision algorithm is given below:

		Certainty
(1′)	*Conservative party* → *high class*	0.07
(2′)	*Conservative party* → *middle class*	0.82
(3′)	*Labour party* → *lower class*	0.31

The efficiency of the inverse decision algorithm is 0.48.

From the decision algorithm we can conclude the following:

- 60% *high class* and 49% *middle class* intend to vote for the *Conservative party*,
- 55% *lower class* intend to vote for the *Labour party*.

The inverse decision algorithm leads to the following explanations of voters' intentions.

- 7% intend to vote for the *Conservative party* belong to the *high class*,
- 82% intend to vote for the *Conservative party* belong to the *middle class*,
- 31% intend to vote for the *Labour party* belong to the *lower class*.

In short, *high* and *middle class* most probably intend to vote for the *Conservative party*, whereas *lower class* most probably intend to vote for the *Labour party*. Voters of the *Conservative party* most probably belong to the *middle class* and voters for the *Labour party* most probably belong to the *lower class*.

We advise the reader to examine the approach and results presented in [8] and compare them with that shown here.

Clearly, the rough set approach is much simpler and gives better results than that discussed in [8].

9. Conclusions

From the rough set view Bayes' theorem reveals probabilistic structure of a data set (i.e., any decision table or decision algorithm) without referring to either prior or posterior probabilities, inherently associated with the Bayesian statistical inference methodology. In other words, it identifies probabilistic relationships between conditions and decisions in decision algorithms, in contrast to classical Bayesian reasoning, where data are employed to verify prior probabilities. This property can be used to give explanation (reasons) for decisions.

Let us also stress that Bayes' theorem in the rough set approach has a new mathematical form based on strength of decision rules, which simplifies essentially computations and gives a new look on the theorem.

Acknowledgements

Thanks are due to Prof. Andrzej Skowron, Prof. Roman Słowiński and Prof. Wojciech Ziarko for their critical remarks.

References

[1] E.W. Adams, The Logic of Conditionals, an Application of Probability to Deductive Logic, Reidel, Dordrecht, 1975.

[2] J. Łukasiewicz, Die logischen Grundlagen der Wahrscheinlichkeitsrechnung, Krakow, 1913 (L. Borkowski (Ed.), Jan Łukasiewicz – Selected Works, North-Holland, Amsterdam, London, Polish Scientific Publishers, Warsaw, 1970.

[3] Z. Pawlak, Rough Sets – Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Dordrecht, 1991.

[4] Z. Pawlak, Decision rules, Bayes' rule and rough sets, in: N. Zhong, A. Skowron, S. Ohsuga (Eds.), Proceedings of the Seventh International Workshop: New Directions in Rough Sets, Data Mining, and Granular – Soft Computing (RSFDGSC'99), Yamaguchi, Japan, November 1999, Lecture Notes in Artificial Intelligence, vol. 1711, Springer, Berlin, 1999, pp. 1–9.

[5] Z. Pawlak, Rough Sets and Decision Algorithms, Springer, 2000, to appear.

[6] L. Polkowski, A. Skowron (Eds.), Proceedings of the First International Conference on Rough Sets and Current Trends in Computing (RSCTC'98), Warsaw, Poland, June 1998, Lecture Notes in Artificial Intelligence, vol. 1424, Springer, Berlin.

[7] L. Polkowski, A. Skowron (Eds.), Rough Sets in Knowledge Discovery, vol. 1–2, Physica-Verlag, Heidelberg, 1998.

[8] M. Ramoni, P. Sebastiani, Bayesian methods, in: M. Berthold, D. Hand (Eds.), Intelligent Data Analysis, An Introduction, Springer, Berlin, 1999.

[9] S. Tsumoto, Modelling medical diagnostic rules based on rough sets, in: L. Polkowski, A. Skowron (Eds.), Proceedings of the First International Conference on Rough Sets and Current Trends in Computing (RSCTC'98), Warsaw, Poland, June 1998, Lecture Notes in Artificial Intelligence, vol. 1424, Springer, Berlin, 1998, pp. 475–482.

[10] W. Ziarko, Approximation region-based decision tables, in: L. Polkowski, A. Skowron (Eds.), Rough Sets in Knowledge Discovery, vol. 1–2, Physica-Verlag, Heidelberg, 1998, pp. 178–185.