

# Decision algorithms and flow graphs; a rough set approach

Zdzisław Pawlak

**Abstract** — This paper concerns some relationship between Bayes' theorem and rough sets. It is revealed that any decision algorithm satisfies Bayes' theorem, without referring to either *prior* or *posterior* probabilities inherently associated with classical Bayesian methodology. This leads to a new simple form of this theorem, which results in new algorithms and applications. Besides, it is shown that with every decision algorithm a flow graph can be associated. Bayes' theorem can be viewed as a flow conservation rule of information flow in the graph. Moreover, to every flow graph the Euclidean space can be assigned. Points of the space represent decisions specified by the decision algorithm, and distance between points depicts distance between decisions in the decision algorithm.

**Keywords** — rough sets, decision algorithms, flow graphs, data mining.

## 1. Introduction

Decision algorithm is a finite set of “if .. then” decision rules. With every decision rule three coefficients are associated: the *strength*, the *certainty* and the *coverage factors* of the rule. The coefficients can be computed from the data or can be a subjective assessment. It is shown that these coefficients satisfy Bayes' formula.

Bayesian inference methodology consists in updating *prior* probabilities by means of data to *posterior* probabilities, which express updated knowledge when data become available. The strength, certainty and coverage factors can be interpreted either as probabilities (objective), or as a degree of truth, along the line proposed by Łukasiewicz [5]. Moreover, they can be also interpreted as a deterministic flow distribution in flow graphs associated with decision algorithms. This leads to a new look on Bayes' theorem and its applications in reasoning from data, without referring to its probabilistic character.

In this context it is worthwhile to mention that in spite of great power of statistical Bayesian methodology of inference methods, the theorem raised wide criticism. E.g., “The technical result at the heart of the essay is what we now know as *Bayes' theorem*. However, from a purely formal perspective there is no obvious reason why this essentially trivial probability result should continue to excite interest” [1]. “Opinion as to the values of Bayes' theorem as a basic for statistical inference has swung between acceptance and rejection since its publication on 1763” [2]. In the proposed setting Bayes' theorem has been set free from its mystic flavor. With every decision algorithm a flow graph can be associated. It is revealed that the throughflow in the graph is ruled by Bayes' theorem. The flow graphs

considered in this paper are different to flow networks introduced by Ford and Fulkerson [4], which are intended to model the flow in transportation network – in contrast to flow graphs, which are meant to be used as a model for decision analysis in decision algorithms. Besides, with every decision algorithm the Euclidean decision space is associated. The decision space is intended to be used to depict differences between decisions of a decision algorithm in a geometrical way.

## 2. Decision algorithms

A *decision rule* is an expression in the form  $\Phi \rightarrow \Psi$ , read “if  $\Phi$  then  $\Psi$ ”, where  $\Phi$  and  $\Psi$  are logical formulas called *condition* and *decision* of the rule, respectively [8].

Let  $|\Phi|$  denote the set of all objects from the universe  $U$ , having the property  $\Phi$ .

If  $\Phi \rightarrow \Psi$  is a decision rule then  $\text{supp}(\Phi, \Psi) = \text{card}(|\Phi \wedge \Psi|)$  will be called the *support* of the decision rule and

$$\sigma(\Phi, \Psi) = \frac{\text{supp}(\Phi, \Psi)}{\text{card}(U)}$$

will be referred to as the *strength* of the decision rule.

With every decision rule  $\Phi \rightarrow \Psi$  we associate a *certainty factor*

$$\text{cer}(\Phi, \Psi) = \frac{\text{supp}(\Phi, \Psi)}{\text{card}(|\Phi|)}$$

and a *coverage factor*

$$\text{cov}(\Phi, \Psi) = \frac{\text{supp}(\Phi, \Psi)}{\text{card}(|\Psi|)}.$$

**Remark.** These coefficients for a long time have been used in data bases and machine learning [9, 10], but first they have been introduced by Łukasiewicz [5] in connection with his study of logic and probability.

If  $\text{cer}(\Phi, \Psi) = 1$ , then the decision rule  $\Phi \rightarrow \Psi$  will be called *certain*, otherwise the decision rule will be referred to as *uncertain*.

A set of decision rules  $\text{Dec}(\Phi, \Psi) = \{\Phi_i \rightarrow \Psi_i\}_{i=1}^n$ ,  $n \geq 2$ , will be called a *decision algorithm* if all its decision rules are:

- *admissible*, i.e.,  $\text{supp}(\Phi_i, \Psi_i) \neq \emptyset$  for every  $1 \leq i \leq n$ ,
- *mutually exclusive (independent)*,  
i.e., for every  $\Phi_i \rightarrow \Psi_i$  and  $\Phi_j \rightarrow \Psi_j$ ,  $\Phi_i = \Psi_j$ ,  
or  $\Phi_i \wedge \Psi_j = \emptyset$  and  $\Psi_i = \Psi_j$  or  $\Psi_i \wedge \Phi_j = \emptyset$ ,
- *cover  $U$* , i.e.,  $|\bigvee_{i=1}^n \Phi_i| = |\bigvee_{i=1}^n \Psi_i| = U$ .

If  $Dec(\Phi, \Psi) = \{\Phi_i \rightarrow \Psi_i\}_{i=1}^n$  is a decision algorithm, then  $Dec(\Psi, \Phi) = \{\Psi_i \rightarrow \Phi_i\}_{i=1}^n$  is also a decision algorithm and will be called an *inverse decision algorithm* of  $Dec(\Phi, \Psi)$ .

$Dec(\Psi, \Phi)$  gives *reasons (explanations)* for decisions of the algorithm  $Dec(\Phi, \Psi)$ .

### 3. Properties of decision algorithms

Let  $Dec(\Phi, \Psi)$  be a decision algorithm and let  $\Phi \rightarrow \Psi$  be a decision rule in the decision algorithm. By  $D(\Phi)$  and  $C(\Psi)$  we denote the set of all decisions of  $\Phi$  and the set of all conditions of  $\Psi$  in  $Dec(\Phi, \Psi)$ , respectively [8].

It can be shown that every decision algorithm has the following probabilistic properties:

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

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

$$\pi(\Psi) = \sum_{\Phi' \in C(\Psi)} cer(\Phi', \Psi) \cdot \pi_S(\Phi') = \sum_{\Phi' \in C(\Psi)} \sigma(\Phi', \Psi), \quad (3)$$

$$\pi(\Phi) = \sum_{\Psi' \in D(\Phi)} cov(\Phi, \Psi') \cdot \pi_S(\Psi') = \sum_{\Psi' \in D(\Phi)} \sigma(\Phi, \Psi'), \quad (4)$$

$$cer(\Phi, \Psi) = \frac{cov(\Phi, \Psi) \cdot \pi(\Psi)}{\sum_{\Psi' \in D(\Phi)} \sigma(\Phi, \Psi')} = \frac{\sigma(\Psi, \Phi)}{\pi(\Phi)}, \quad (5)$$

$$cov(\Phi, \Psi) = \frac{cer(\Phi, \Psi) \cdot \pi(\Phi)}{\sum_{\Phi' \in C(\Psi)} \sigma(\Phi', \Psi)} = \frac{\sigma(\Phi, \Psi)}{\pi(\Psi)}, \quad (6)$$

where  $\pi(\Psi) = \frac{card(|\Psi|)}{card(U)}$  and  $\pi(\Phi) = \frac{card(|\Phi|)}{card(U)}$ .

Let us observe that formally formulas (1)–(6) have probabilistic favor. In particular, formulas (3) and (4) can be understood as total probability theorems, whereas formulas (5) and (6) have the form of Bayes' theorem. As mentioned before, if we interpret strength, certainty and coverage factors as probabilities (frequencies) then these formulas describe some relationships between data in the decision algorithm. However, we can also interpret these factors in a deterministic way, as a degree of truth. In this case the coverage factor  $cov(\Phi, \Psi)$  expresses to which degree the conclusion  $\Psi$  of a decision rule  $\Phi \rightarrow \Psi$  is true if the condition  $\Phi$  of the rule is true to the degree  $cer(\Phi, \Psi)$ , whereas  $\sigma(\Phi, \Psi)$  can be regarded as a truth value of the decision rule (implication)  $\Phi \rightarrow \Psi$ .

The idea to replace probability by truth values is due to Łukasiewicz [5], but we will not discuss this issue here.

### 4. Flow graphs

With every decision algorithm we associate a directed, acyclic, connected graph defined in the following way: to every condition and decision of the decision rule in the decision algorithm we associate a *node* of the graph. To every decision rule  $\Phi \rightarrow \Psi$  we assign a directed *branch* connecting the *input node*  $\Phi$  and the *output node*  $\Psi$ . Strength of the decision rule represents the *throughflow* of the corresponding branch. More about flow graphs and decision algorithms can be found in [7].

The throughflow of the graph is governed by formulas (1)–(6), and can be considered as a *flow conservation equation* similar to that of Ford and Fulkerson [4].

Consequently, the flow graphs can be regarded as a third model of Bayes' theorem, in which the theorem describes flow distribution in a flow graph.

### 5. Decision space

With every decision algorithm with  $n$ -valued decisions we can associate  $n$ -dimensional Euclidean space, where values of decisions determine  $n$  axis of the space and condition attribute values (equivalence classes) determine point of the space. Strengths of decision rules are to be understood as coordinates of corresponding points.

Distance  $\delta(x, y)$  between point  $x$  and  $y$  in an  $n$ -dimensional decision space is defined as

$$\delta(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2},$$

where  $x = (x_1, \dots, x_n)$  and  $y = (y_1, \dots, y_n)$  are vectors of strengths of corresponding decision rules.

### 6. An example

For the sake of illustration let us consider a very simple decision algorithm describing vote distribution for two political parties  $X_1$ , and  $X_2$  from three mutually disjoint sample group of voters  $Y_1$ ,  $Y_2$  and  $Y_3$ :

1.  $Y_1 \rightarrow X_1$  (400)
2.  $Y_1 \rightarrow X_2$  (200)
3.  $Y_2 \rightarrow X_1$  (250)
4.  $Y_2 \rightarrow X_2$  (50)
5.  $Y_3 \rightarrow X_1$  (90)
6.  $Y_3 \rightarrow X_2$  (10).

Number given at the end of each rule is the support of the rule, i.e., the number of voters from group  $X_i$  voting for party  $Y_j$ .

The *strength*, *certainty* and *coverage* factors for each decision rule are given in Table 1.

Table 1  
Parameters of the decision rules

Rule	Strength	Certainty	Coverage
1	0.40	0.67	0.54
2	0.20	0.33	0.77
3	0.25	0.83	0.34
4	0.05	0.17	0.19
5	0.09	0.90	0.12
6	0.01	0.10	0.04

The corresponding flow graph is shown in Fig. 1.

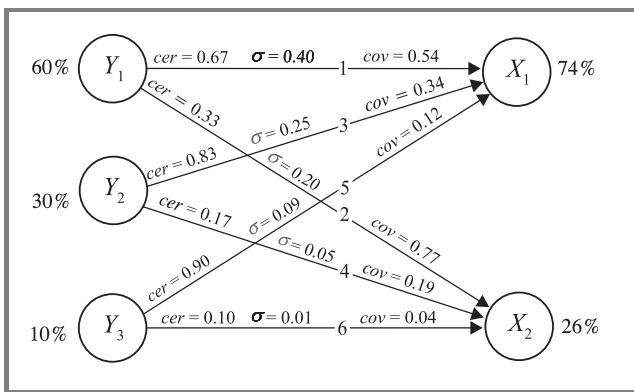


Fig. 1. The flow graph corresponding to the example.

Thus from the decision algorithm follows, for example, that 83% voters from group  $Y_2$  voted for party  $X_1$  and 17% voters voted for party  $X_2$ . From the inverse decision algorithm

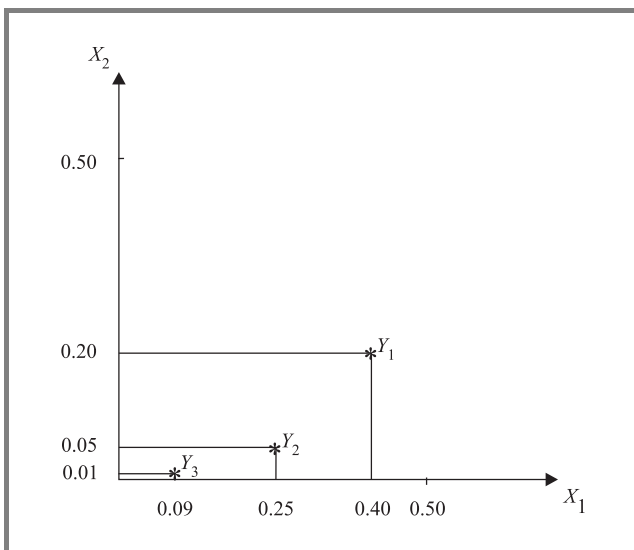


Fig. 2. The distance space corresponding to the example.

we get, for example, that for party  $X_1$  voted 54% voters of group  $Y_1$ , 34% – of group  $Y_2$ , and 12% – of group  $Y_3$ . The corresponding distance space is shown in Fig. 2. Distances between voters are presented in Table 2.

Table 2  
Distances between voters

	$Y_1$	$Y_2$	$Y_3$
$Y_1$			
$Y_2$	0.20		
$Y_3$	0.37	0.22	

## 7. Summary

In this paper a relationship between decision algorithms, flow graphs and Bayes' theorem are defined and briefly analyzed. It is shown that decision algorithms satisfy Bayes' theorem, and that the theorem can be also interpreted without referring to its probabilistic connotation – in a purely deterministic way. This property leads to a new look on Bayes' theorem and new applications of Bayes' rule in data analysis.

## References

- [1] J. M. Bernardo and A. F. M. Smith, *Bayesian Theory*. Chichester [etc.]: Wiley, 1994 (Wiley series in probability and mathematical statistics).
- [2] G. E. P. Box and G. C. Tiao, *Bayesian Inference in Statistical Analysis*. New York [etc.]: Wiley, 1992.
- [3] M. Berthold and D. J. Hand, *Intelligent Data Analysis – an Introduction*. Berlin: Springer, 1999.
- [4] L. R. Ford and D. R. Fulkerson, *Flows in Networks*. Princeton [etc.]: Princeton University Press, 1962.
- [5] J. Łukasiewicz, “Die logischen Grundlagen der Wahrscheinlichkeitsrechnung” (Kraków, 1913) in *Jan Łukasiewicz – Selected Works*, L. Borkowski, Ed. Amsterdam [etc.]: North Holland Publ., Warsaw: Polish Scientific Publ., 1970.
- [6] S. Greco, Z. Pawlak, and R. Stowiński, “Generalized decision algorithms, rough inference rules and flow graphs” in *Rough Sets and Current Trends in Computing*, J. J. Alpigini, J. F. Peters, A. Skowron, and N. Zhong, Eds., *Lecture Notes in Artificial Intelligence*. Berlin: Springer, 2002, vol. 2475, pp. 93–104.
- [7] Z. Pawlak, “In pursuit of patterns in data reasoning from data – the rough set way” in *Rough Sets and Current Trends in Computing*, J. J. Alpigini, J. F. Peters, A. Skowron, and N. Zhong, Eds., *Lecture Notes in Artificial Intelligence*. Berlin: Springer, 2002, vol. 2475, pp. 1–9.
- [8] Z. Pawlak, “Rough sets, decision algorithms and Bayes' theorem”, *Eur. J. Oper. Res.*, vol. 136, pp. 181–189, 2002.
- [9] S. Tsumoto and H. Tanaka, “Discovery of functional components of proteins based on PRIMEROSE and domain knowledge hierarchy” in *Proc. Worksh. Rough Sets & Soft Comput. RSSC-94*, T. Y. Lin and A. M. Wildberger, Eds., *Soft Computing*, SCS, 1995, pp. 280–285.
- [10] S. K. M. Wong and W. Ziarko, “Algorithm for inductive learning”, *Bull. Polish Acad. Sci.*, vol. 34, no. 5-6, pp. 271–276, 1986.



**Zdzisław Pawlak** was born in Łódź (Poland), in 1926. He obtained his M.Sc. in 1951 in electronics from Warsaw University of Technology, Ph.D. in 1958 and D.Sc. in 1963 in the theory of computation from the Polish Academy of Sciences. He is a Professor of the Institute of Theoretical and Applied Informatics, Polish

Academy of Sciences and the University of Information Technology and Management and Member of the Polish Academy of Sciences. His current research interests include intelligent systems and cognitive sciences, in particular, decision support systems, knowledge representation, reasoning about knowledge, machine learning, inductive reasoning, vagueness, uncertainty and decision support. He is an author of a new mathematical tool, called rough set theory, intended to deal with vagueness and uncertainty. About two thousand papers have been published by now on rough sets and their applications world wide. Several international workshops and conferences on rough sets have

been held in recent years. He is a recipient of many awards among others the State Award in Computer Science in 1978, the Hugo Steinhaus Award for achievements in applied mathematics in 1989. Doctor honoris causa of Poznań University of Technology in 2002. Member of editorial boards of several dozens international journals. Deputy Editor-in-Chief of the Bulletin of the Polish Academy of Sciences. Program committee member of many international conferences on computer science. Over forty visiting university appointments in Europe, USA and Canada, about fifty invited international conference talks, and over one hundred seminar talks given in about fifty universities in Europe, USA, Canada, China, India, Japan, Korea, Taiwan, Australia and Israel. About two hundred articles in international journals and several books on various aspects on computer science and application of mathematics. Supervisor of thirty Ph.D. theses in computer science and applied mathematics.

e-mail: zip@mimuw.edu.pl

University of Information Technology and Management  
Newelska st 6  
01-447 Warsaw, Poland