Flow Graphs and Data Mining

Zdzisław Pawlak[1,2]

[1] Institute for Theoretical and Applied Informatics,
Polish Academy of Sciences,
ul. Bałtycka 5, 44-100 Gliwice, Poland
[2] Warsaw School of Information Technology,
ul. Newelska 6, 01-447 Warsaw, Poland
zpw@ii.pw.edu.pl

Abstract. In this paper we propose a new approach to data mining and knowledge discovery based on information flow distribution in a flow graph. Flow graphs introduced in this paper are different from those proposed by Ford and Fulkerson for optimal flow analysis and they model flow distribution in a network rather than the optimal flow which is used for information flow examination in decision algorithms. It is revealed that flow in a flow graph is governed by Bayes' rule, but the rule has an entirely deterministic interpretation without referring to its probabilistic roots. Besides, a decision algorithm induced by a flow graph and dependency between conditions and decisions of decision rules is introduced and studied, which is used next to simplify decision algorithms.

Keywords: flow graph, data mining, knowledge discovery, decision algorithms.

Introduction

In this paper we propose a new approach to data analysis (mining) based on information flow distribution study in a flow graph.

Flow graphs introduced in this paper are different from those proposed by Ford and Fulkerson [4] for optimal flow analysis and they model rather flow *distribution* in a network, than the optimal flow.

The flow graphs considered in this paper are not meant to model physical media (e.g., water) flow analysis, but to model information flow examination in decision algorithms. To this end branches of a flow graph can be interpreted as decision rules. With every decision rule (i.e., branch) three coefficients are associated: the *strength, certainty* and *coverage factors*.

These coefficients have been used under different names in data mining (see, e.g., [14, 15]) but they were used first by Łukasiewicz [8] in his study of logic and probability.

This interpretation, in particular, leads to a new look at Bayes' theorem. Let us also observe that despite Bayes' rule fundamental role in statistical inference it has led to many philosophical discussions concerning its validity and meaning, and has caused much criticism [1, 3, 13].

J.F. Peters and A. Skowron (Eds.): Transactions on Rough Sets III, LNCS 3400, pp. 1–36, 2005.

This paper is a continuation of some of the authors' ideas presented in [10, 11], where the relationship between Bayes' rule and flow graphs has been introduced and studied (see also [6, 7]).

This paper consists of two parts. Part one introduces basic concepts of the proposed approach, i.e., flow graph and its fundamental properties. It is revealed that flow in a flow graph is governed by Bayes' rule, but the rule has an entirely deterministic interpretation that does not refer to its probabilistic roots. In addition, dependency of flow is defined and studied. This idea is based on the statistical concept of dependency but in our setting it has a deterministic meaning.

In part two many tutorial examples are given to illustrate how the introduced ideas work in data mining. These examples clearly show the difference between classical Bayesian inference methodology and the proposed one.

The presented ideas can be used, among others, as a new tool for data mining, and knowledge representation. Besides, the proposed approach throws new light on the concept of probability.

1 Flow Graphs

1.1 Overview

In this part the fundamental concepts of the proposed approach are defined and discussed. In particular flow graphs, certainty and coverage factors of branches of the flow graph are defined and studied. Next these coefficients are extended to paths and some classes of sub-graphs called connections. Further a notion of fusion of a flow graph is defined.

Further dependences of flow are introduced and examined. Finally, dependency factor (correlation coefficient) is defined.

Observe that in many cases the data flow order, represented in flow graphs, explicitly follows from the problem specification. However, in other cases the relevant order should be discovered from data. This latter issue will be discussed elsewhere.

1.2 Basic Concepts

A flow graph is a *directed, acyclic, finite* graph $G = (N, \mathcal{B}, \varphi)$, where N is a set of *nodes*, $\mathcal{B} \subseteq N \times N$ is a set of *directed branches*, $\varphi : \mathcal{B} \to R^+$ is a *flow function* and R^+ is the set of non-negative reals.

Input of a node $x \in N$ *is the set* $I(x) = \{y \in N : (y, x) \in \mathcal{B}\}$; *output* of a node $x \in N$ is defined by $O(x) = \{y \in N : (x, y) \in \mathcal{B}\}$.

We will also need the concept of *input* and *output* of a graph G, defined, respectively, as follows: $I(G) = \{x \in N : I(x) = \emptyset\}$, $O(G) = \{x \in N : O(x) = \emptyset\}$.

Inputs and outputs of G are *external nodes* of G; other nodes are *internal nodes* of G.

If $(x, y) \in \mathcal{B}$, then $\varphi(x, y)$ is a *throughflow* from x to y.

With every node x of a flow graph G we associate its *inflow*

$$\varphi_+(x) = \sum_{y \in I(x)} \varphi(y, x), \tag{1}$$

and *outflow*

$$\varphi_-(x) = \sum_{y \in O(x)} \varphi(x, y). \tag{2}$$

Similarly, we define an inflow and an outflow for the whole flow graph, which are defined by

$$\varphi_+(G) = \sum_{x \in I(G)} \varphi_-(x), \tag{3}$$

$$\varphi_-(G) = \sum_{x \in O(G)} \varphi_+(x). \tag{4}$$

We assume that for any internal node x we have $\varphi_+(x) = \varphi_-(x) = \varphi(x)$, where $\varphi(x)$ is a *throughflow* of node x.

Then, obviously, $\varphi_+(G) = \varphi_-(G) = \varphi(G)$, where $\varphi(G)$ is a *throughflow* of graph G.

The above formulas can be considered as *flow conservation equations* [4].

Example

We will illustrate the basic concepts of flow graphs by an example of a group of 1000 patients put to the test for certain drug effectiveness.

Assume that patients are grouped according to presence of the disease, age and test results, as shown in Fig. 1.

For example, $\varphi(x_1) = 600$ means that these are 600 patients suffering from the disease, $\varphi(y_1) = 570$ means that there are 570 old patients $\varphi(z_1) = 471$ means that 471 patients have a positive test result; $\varphi(x_1, y_1) = 450$ means that there are 450 old patients which suffer from disease etc.

Thus the flow graph gives clear insight into the relationship between different groups of patients.

Let us now explain the flow graph in more detail.

Nodes of the flow graph are depicted by circles, labeled by $x_1, x_2, y_1, y_2, y_3, z_1$, z_2. A branch (x, y) is denoted by an arrow from node x to y. For example, branch (x_1, z_1) is represented by an arrow from x_1 to z_1.

For example, inputs of node y_1 are nodes x_1 and x_2, outputs of node x_1 are nodes y_1, y_2 and y_3.

Inputs of the flow graph are nodes x_1 and x_2, whereas the outputs of the flow graph are nodes z_1 and z_2.

Nodes y_1, y_2 and y_3 are internal nodes of the flow graph. The throughflow of the branch (x_1, y_1) is $\varphi(x_1, y_1) = 450$. Inflow of node y_1 is $\varphi_+(y_1) = 450 + 120 = 570$. Outflow of node y_1 is $\varphi_-(y_1) = 399 + 171 = 570$. Inflow of the flow graph is $\varphi(x_1) + \varphi(x_2) = 600 + 400 = 1000$, and outflow of the flow graph is $\varphi(z_1) + \varphi(z_2) = 471 + 529 = 1000$.

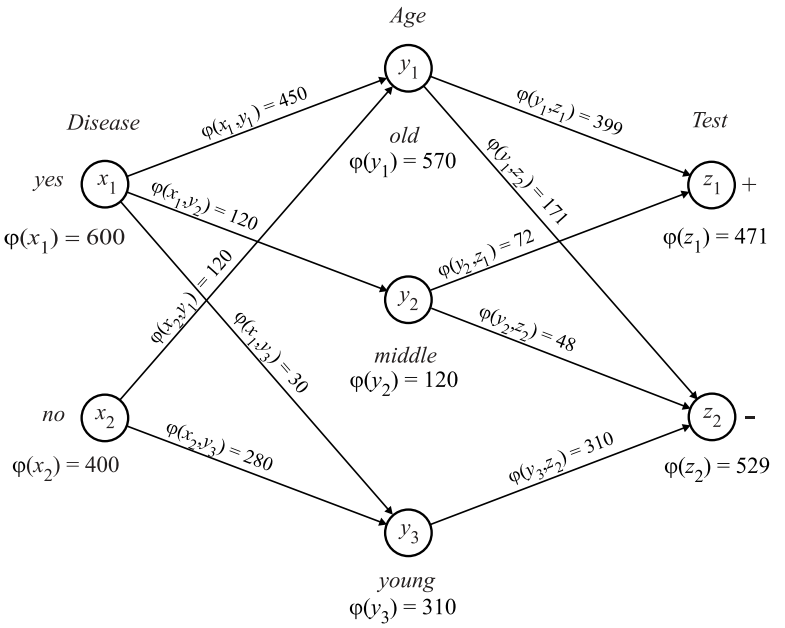

Fig. 1. Flow graph.

Throughflow of node y_1 is equal to $\varphi(y_1) = \varphi(x_1, y_1) + \varphi(x_2, y_1) = \varphi(y_1, z_1) + \varphi(y_2, z_2) = 570$. □

We will now define a *normalized flow graph*.

A normalized flow graph is a *directed, acyclic, finite* graph $G = (N, \mathcal{B}, \sigma)$, where N is a set of *nodes*, $\mathcal{B} \subseteq N \times N$ is a set of *directed branches* and $\sigma : \mathcal{B} \to < 0, 1 >$ is a *normalized flow* of (x, y) and

$$\sigma(x, y) = \frac{\varphi(x, y)}{\varphi(G)} \tag{5}$$

is a *strength* of (x, y). Obviously, $0 \leq \sigma(x, y) \leq 1$. The strength of the branch (multiplied by 100) expresses simply the percentage of a total flow through the branch.

In what follows we will use normalized flow graphs only, therefore by flow graphs we will understand normalized flow graphs, unless stated otherwise.

With every node x of a flow graph G we associate its *inflow* and *outflow* defined by

$$\sigma_+(x) = \frac{\varphi_+(x)}{\varphi(G)} = \sum_{y \in I(x)} \sigma(y, x), \tag{6}$$

$$\sigma_-(x) = \frac{\varphi_-(x)}{\varphi(G)} = \sum_{y \in O(x)} \sigma(x, y). \tag{7}$$

Obviously for any internal node x, we have $\sigma_+(x) = \sigma_-(x) = \sigma(x)$, where $\sigma(x)$ is a *normalized throughflow* of x.

Moreover, let

$$\sigma_+(G) = \frac{\varphi_+(G)}{\varphi(G)} = \sum_{x \in I(G)} \sigma_-(x), \tag{8}$$

$$\sigma_-(G) = \frac{\varphi_-(G)}{\varphi(G)} = \sum_{x \in O(G)} \sigma_+(x). \tag{9}$$

Obviously, $\sigma_+(G) = \sigma_-(G) = \sigma(G) = 1$.

Example (cont.) The normalized flow graph of the flow graph presented in Fig. 1 is given in Fig. 2.

In the flow graph, e.g., $\sigma(x_1) = 0.60$, that means that 60% of total inflow is associated with input x_1. The strength $\sigma(x_1, y_1) = 0.45$ means that 45% of total flow of x_1 flows through the branch (x_1, y_1) etc. □

Let $G = (N, \mathcal{B}, \sigma)$ be a flow graph. If we invert direction of all branches in G, then the resulting graph $G = (N, \mathcal{B}', \sigma')$ will be called an *inverted* graph of G. Of course, the inverted graph G' is also a flow graph and all inputs and outputs of G become inputs and outputs of G', respectively.

Example (cont.) The inverted flow graph of the flow graph from Fig. 2 is shown in Fig. 3. □

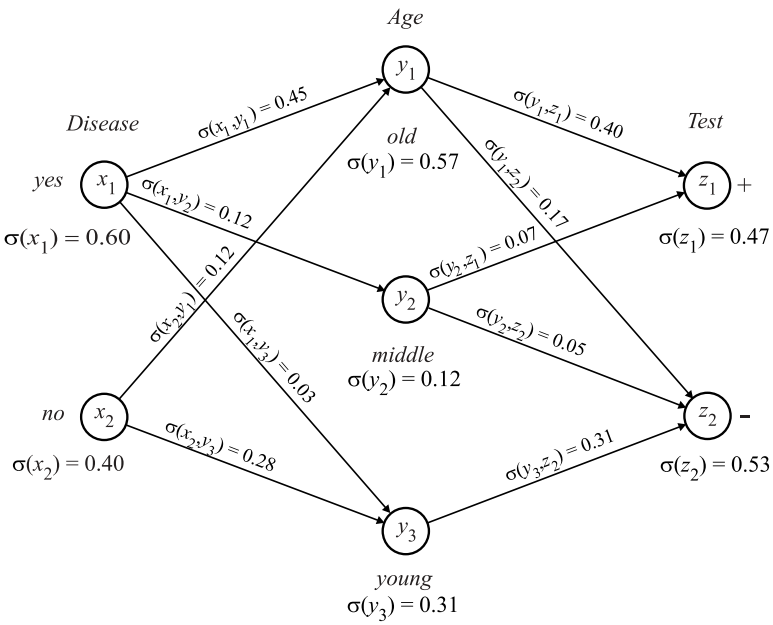

Fig. 2. Normalized flow graph.

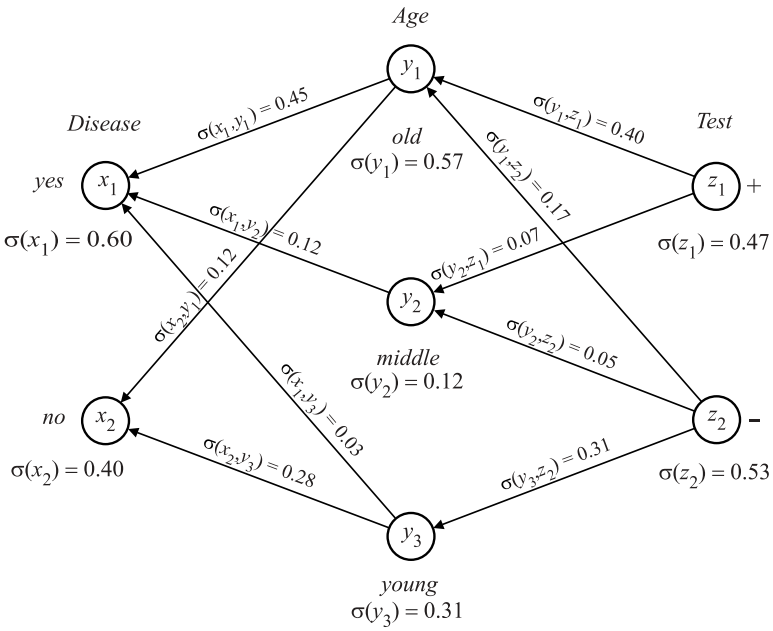

Fig. 3. Inverted flow graph.

1.3 Certainty and Coverage Factors

With every branch (x, y) of a flow graph G we associate the *certainty* and the *coverage factors*.

The *certainty* and the *coverage* of (x, y) are defined by

$$cer(x, y) = \frac{\sigma(x, y)}{\sigma(x)}, \tag{10}$$

and

$$cov(x, y) = \frac{\sigma(x, y)}{\sigma(y)}. \tag{11}$$

respectively.

Evidently, $cer(x, y) = cov(y, x)$, where $(x, y) \in \mathcal{B}$ and $(y, x) \in \mathcal{B}'$.

Example (cont.) The certainty and the coverage factors for the flow graph presented in Fig. 2 are shown in Fig. 4.

For example, $cer(x_1, y_1) = \frac{\sigma(x_1,y_1)}{\sigma(x_1)} = \frac{0.45}{0.60} = 0.75$, and $cov(x_1, y_1) = \frac{\sigma(x_1,y_1)}{\sigma(y_1)} = \frac{0.45}{0.57} \approx 0.79$. □

Below some properties of certainty and coverage factors, which are immediate consequences of definitions given above, are presented:

$$\sum_{y \in O(x)} cer(x, y) = 1, \tag{12}$$

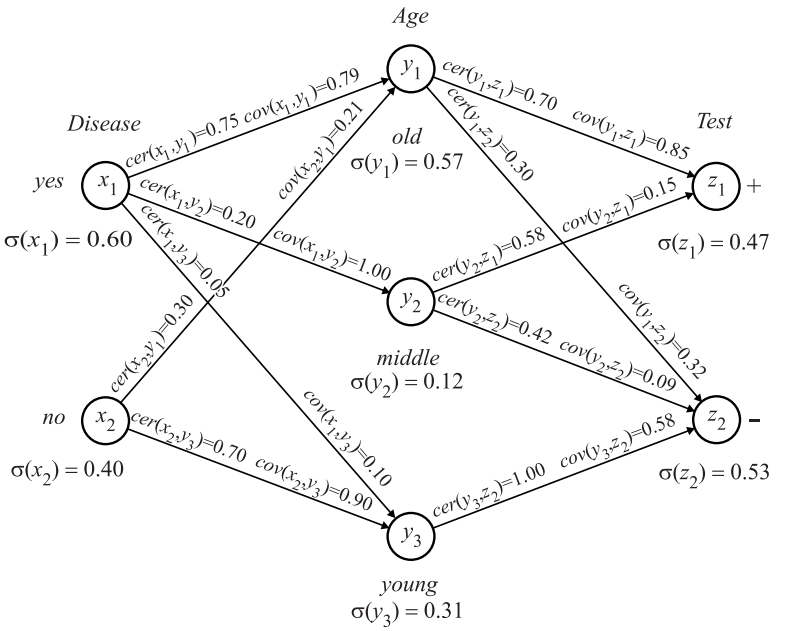

Fig. 4. Certainty and coverage.

$$\sum_{x \in I(y)} cov(x, y) = 1, \tag{13}$$

$$\sigma(x) = \sum_{y \in O(x)} cer(x, y)\sigma(x) = \sum_{y \in O(x)} \sigma(x, y), \tag{14}$$

$$\sigma(y) = \sum_{x \in I(y)} cov(x, y)\sigma(y) = \sum_{x \in I(y)} \sigma(x, y), \tag{15}$$

$$cer(x, y) = \frac{cov(x, y)\sigma(y)}{\sigma(x)}, \tag{16}$$

$$cov(x, y) = \frac{cer(x, y)\sigma(x)}{\sigma(y)}. \tag{17}$$

Obviously the above properties have a probabilistic flavor, e.g., equations (14) and (15) have a form of total probability theorem, whereas formulas (16) and (17) are Bayes' rules. However, these properties in our approach are interpreted in a deterministic way and they describe flow distribution among branches in the network.

1.4 Paths, Connections and Fusion

A (*directed*) *path* from x to y, $x \neq y$ in G is a sequence of nodes $x_1, \ldots, x_n$ such that $x_1 = x$, $x_n = y$ and $(x_i, x_{i+1}) \in \mathcal{B}$ for every $i, 1 \leq i \leq n-1$. A path from x to y is denoted by $[x \ldots y]$.

The *certainty* of the path $[x_1 \ldots x_n]$ is defined by

$$cer[x_1 \ldots x_n] = \prod_{i=1}^{n-1} cer(x_i, x_{i+1}), \tag{18}$$

the *coverage* of the path $[x_1 \ldots x_n]$ is

$$cov[x_1 \ldots x_n] = \prod_{i=1}^{n-1} cov(x_i, x_{i+1}), \tag{19}$$

and the *strength* of the path $[x_1 \ldots x_n]$ is

$$\sigma[x_1 \ldots x_n] = \sigma(x_1)cer[x_1 \ldots x_n] = \sigma(x_n)cov[x_1 \ldots x_n]. \tag{20}$$

The set of all paths from x to y $(x \neq y)$ in G, denoted by $< x, y >$, will be called a *connection* from x to y in G. In other words, connection $< x, y >$ is a sub-graph of G determined by nodes x and y.

The *certainty* of the connection $< x, y >$ is

$$cer < x, y >= \sum_{[x \ldots y] \in < x, y >} cer[x \ldots y], \tag{21}$$

the *coverage* of the connection $< x, y >$ is

$$cov < x, y >= \sum_{[x \ldots y] \in < x, y >} cov[x \ldots y], \tag{22}$$

and the *strength* of the connection $< x, y >$ is

$$\sigma < x, y > = \sum_{[x \ldots y] \in < x, y >} \sigma[x \ldots y] = \\ = \sigma(x)cer < x, y >= \sigma(y)cov < x, y > . \tag{23}$$

If we substitute simultaneously any sub-graph $< x, y >$ of a given flow graph G, where x and y are input and output nodes of G respectively, by a single branch (x, y) such that $\sigma(x, y) = \sigma < x, y >$, then in the resulting graph G', called the *fusion* of G, we have $cer(x, y) = cer < x, y >$, $cov(x, y) = cov < x, y >$ and $\sigma(G) = \sigma(G')$.

Example (cont.) In the flow graph presented in Fig. 3 for the path $p = [x_1, y_1, z_1]$ we have $cer(p) = 0.75 \times 0.70 \approx 0.53$, $cov(p) = 0.85 \times 0.79 \approx 0.67$.

The connection $< x_1, z_1 >$ in the flow graph consists of paths $[x_1, y_1, z_1]$ and $[x_1, y_2, z_1]$. This connection is shown in Fig. 5 by bold lines.

For this connection we have $cer < x_1, z_1 >= 0.75 \times 0.70 + 0.20 \times 0.60 \approx 0.65$; $cov < x_1, z_1 >= 0.85 \times 0.79 + 0.15 \times 1.00 \approx 0.82$.

The strength of the connection x_1, z_1 is $0.68 \times 0.60 \approx 0.85 \times 0.47 \approx 0.40$. Connections $< x_1, z_2 >$, $< x_2, z_1 >$, and $< x_2, z_2 >$ are presented in Fig. 6, Fig. 7 and Fig. 8, respectively. □

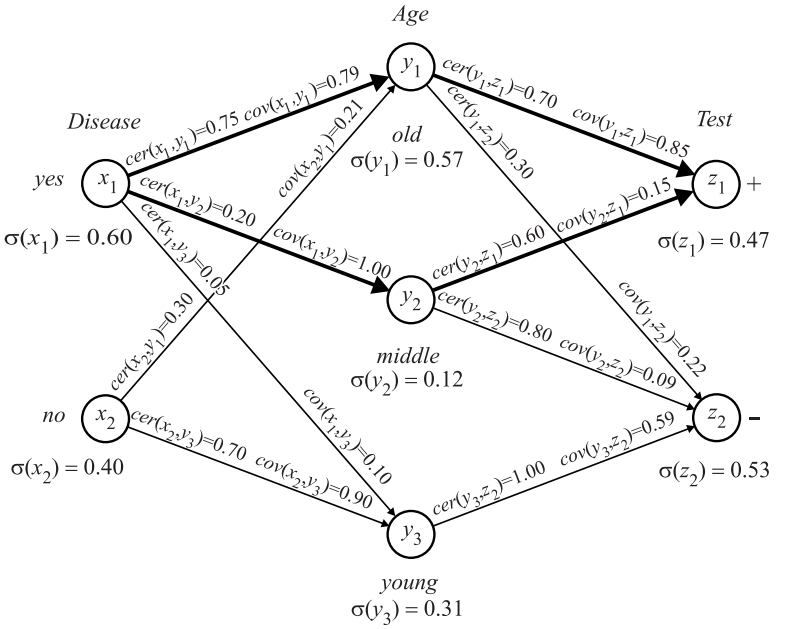

Fig. 5. Connection $< x_1, z_1 >$.

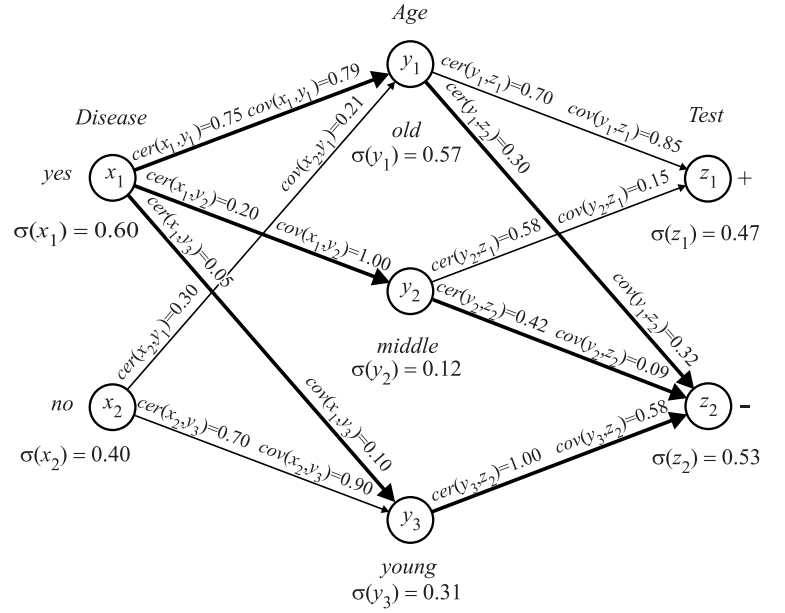

Fig. 6. Connection $< x_1, z_2 >$.

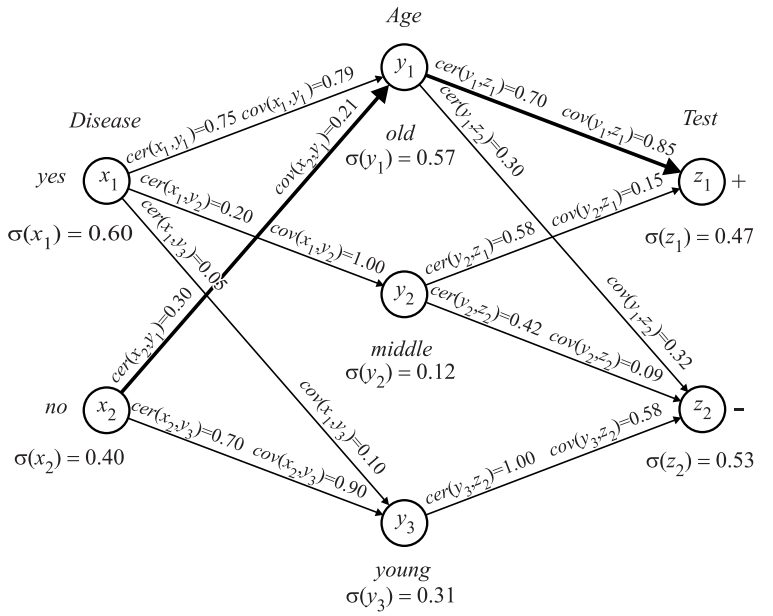

Fig. 7. Connection $< x_2, z_1 >$.

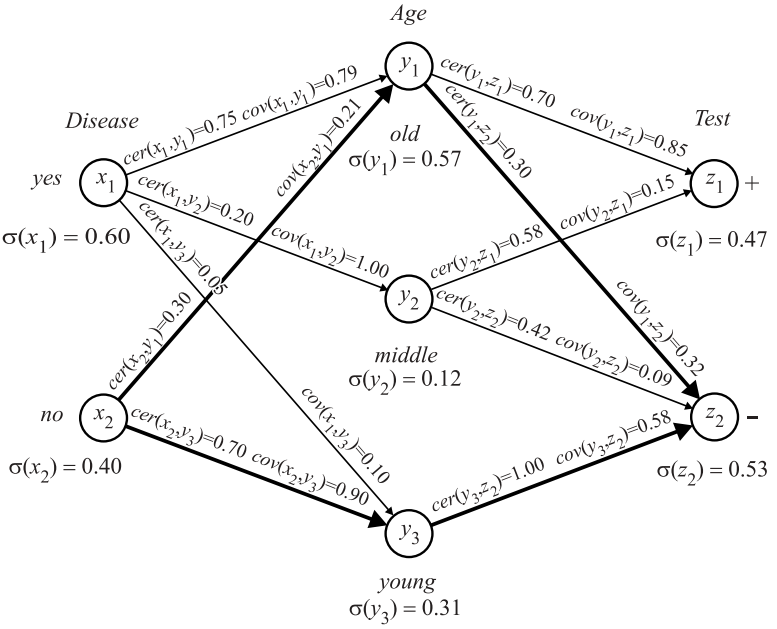

Fig. 8. Connection $< x_2, z_2 >$.

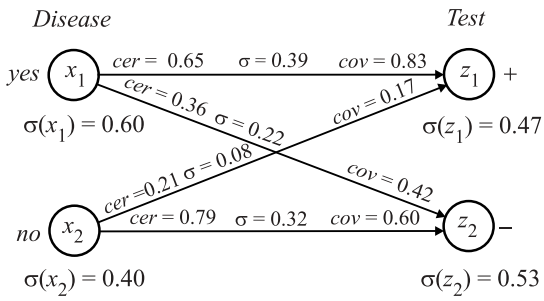

Fig. 9. Fusion of the flow graph.

Example (cont.) The fusion of the flow graph shown in Fig. 3 is given in Fig. 9.

The fusion of a flow graph gives information about the flow distribution between input and output of the flow graph, i.e., it leads to the following conclusions:

- if the disease is present then the test result is positive with certainty 0.65,
- if the disease is absent then the test result is negative with certainty 0.79.

Explanation of the test results is as follows:

- if the test result is positive then the disease is present with certainty 0.83,
- if the test result is negative then the disease is absent with certainty 0.60.

□

1.5 Dependences in Flow Graphs

Let x and y be nodes in a flow graph $G = (N, \mathcal{B}, \sigma)$, such that $(x, y) \in \mathcal{B}$. Nodes x and y are *independent* in G if

$$\sigma(x, y) = \sigma(x)\sigma(y). \tag{24}$$

From (24) we get

$$\frac{\sigma(x, y)}{\sigma(x)} = cer(x, y) = \sigma(y), \tag{25}$$

and

$$\frac{\sigma(x, y)}{\sigma(y)} = cov(x, y) = \sigma(x). \tag{26}$$

If

$$cer(x, y) > \sigma(y), \tag{27}$$

or

$$cov(x, y) > \sigma(x), \tag{28}$$

then x and y are *positively dependent* on x in G.

Similarly, if

$$cer(x, y) < \sigma(y), \tag{29}$$

or

$$cov(x, y) < \sigma(x), \tag{30}$$

then x and y are *negatively dependent* in G.

Let us observe that relations of independency and dependences are symmetric ones, and are analogous to those used in statistics.

For every branch $(x, y) \in \mathcal{B}$ we define a *dependency* (*correlation*) *factor* $\eta(x, y)$ defined by

$$\eta(x, y) = \frac{cer(x, y) - \sigma(y)}{cer(x, y) + \sigma(y)} = \frac{cov(x, y) - \sigma(x)}{cov(x, y) + \sigma(x)}. \tag{31}$$

Obviously $-1 \leq \eta(x, y) \leq 1$; $\eta(x, y) = 0$ if and only if $cer(x, y) = \sigma(y)$ and $cov(x, y) = \sigma(x)$; $\eta(x, y) = -1$ if and only if $cer(x, y) = cov(x, y) = 0$; $\eta(x, y) = 1$ if and only if $\sigma(y) = \sigma(x) = 0$.

It is easy to check that if $\eta(x, y) = 0$, then x and y are independent, if $-1 \leq \eta(x, y) < 0$ then x and y are negatively dependent and if $0 < \eta(x, y) \leq 1$ then x and y are positively dependent. Thus the dependency factor expresses a degree of dependency, and can be seen as a counterpart of the correlation coefficient used in statistics.

Example (cont.) Dependency factors for the flow graph shown in Fig. 9 are given in Fig. 10.

Thus, there is a positive dependency between the presence of the disease and the positive test result as well as between absence of the disease and negative test result. However, there is a much stronger negative dependency between presence of the disease and negative test result or similarly – between absence of the disease and positive left test result. More specifically:

- there is slight positive correlation between presence of the disease and positive test result ($\eta = 0.16$),
- there is low positive correlation between absence of the disease and negative test result ($\eta = 0.20$),
- there a negative correlation between presence of the disease and negative test result ($\eta = -0.19$),
- there is high negative correlation between absence of the disease and positive test result ($\eta = -0.38$). □

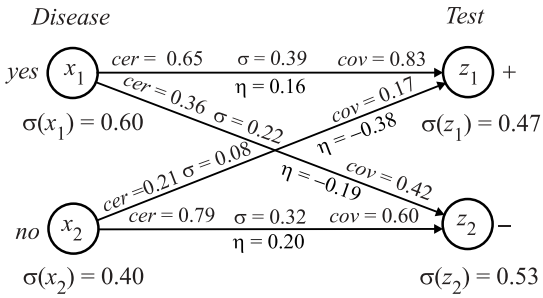

Fig. 10. Fusion of the flow graph.

1.6 Flow Graph and Decision Algorithms

Flow graphs can be interpreted as decision algorithms [5].

Let us assume that the set of nodes of a flow graph is interpreted as a set of logical formulas. The formulas are understood as propositional functions and if x is a formula, then $\sigma(x)$ is to be interpreted as a truth value of the formula. Let us observe that the truth values are numbers from the closed interval $[0, 1]$, i.e., $0 \leq \sigma(x) \leq 1$.

According to [3] these truth values can be also interpreted as probabilities. Thus $\sigma(x)$ can be understood as flow distribution ratio (percentage), truth value or probability. We will stick to the first interpretation.

With every branch (x, y) we associate a decision rule $x \to y$, read *if x then y*; x will be referred to as *condition*, whereas y – *decision* of the rule. Such a rule is characterized by three numbers, $\sigma(x, y)$, $cer(x, y)$ and $cov(x, y)$.

Let us observe that the inverted flow graph gives reasons for decisions.

Every path $[x_1 \ldots x_n]$ determines a sequence of decision rules $x_1 \to x_2$, $x_2 \to x_3, \ldots, x_{n-1} \to x_n$.

From previous considerations it follows that this sequence of decision rules can be interpreted as a single decision rule $x_1 x_2 \ldots x_{n-1} \to x_n$, in short $x^* \to x_n$, where $x^* = x_1 x_2 \ldots x_{n-1}$, characterized by

$$cer(x^*, x_n) = \frac{\sigma(x^*, x_n)}{\sigma(x^*)}, \tag{32}$$

$$cov(x^*, x_n) = \frac{\sigma(x^*, x_n)}{\sigma(x_n)}, \tag{33}$$

and

$$\sigma(x^*, x_n) = \sigma(x_1, \ldots, x_{n-1}, x_n),\ \sigma(x^*) = \sigma(x_1, \ldots, x_{n-1}). \tag{34}$$

From (32) we have

$$cer(x^*, x_n) = \frac{cer[x_1, \ldots, x_{n-1}, x_n]}{cer[x_1, \ldots, x_{n-1}]}.$$

The set of all decision rules $x_{i_1} x_{i_2} \ldots x_{i_{n-1}} \to x_{i_n}$ associated with all paths $[x_{i_1} \ldots x_{i_n}]$ such that x_{i_1} and x_{i_n} are input and output of the graph respectively will be called a *decision algorithm* induced by the flow graph.

If $x \to y$ is a decision rule, then we say that the condition and decision of the decision rule are independent if x and y are independent, otherwise the condition and decision of the decision rule are dependent (positively or negatively).

To measure the degree of dependency between the condition and decision of the decision rule $x \to y$, we can use the dependency factor $\eta(x, y)$.

Let us observe that if the conditions and decisions of a decision rule $x \to y$ are independent, then the decision rule is, in certain sense, useless, because such a decision rule indicates that there is no relationship between conditions and decisions and the decision can be eliminated from the decision algorithm.

On the other hand, the most important decision rules are those having the highest dependency factor and strength, for they indicate a strong relationship in substantial portion of the data. This property can be used to simplify the decision algorithms, because we can eliminate less relevant decision rules from the algorithm, at the cost of its lower classification power.

With every subset of decision rules $\delta_1, \ldots, \delta_n$ of the decision algorithm we can associate its strength equal to the sum of strengths of the decision rules, i.e., $\sum_{i=1}^{n} \sigma(\delta_i)$, which can be used as a measure of the classification power of the algorithm.

Example (cont.) The decision algorithm induced by the flow graph shown in Fig. 4 is given in the table:

	certainty	coverage	strength
$x_1, y_1 \to z_1$	0.71	0.67	0.32
$x_1, y_1 \to z_2$	0.31	0.25	0.14
$x_1, y_2 \to z_1$	0.58	0.15	0.07
$x_1, y_2 \to z_2$	0.42	0.09	0.05
$x_1, y_3 \to z_2$	1.00	0.06	0.03
$x_2, y_1 \to z_1$	0.40	0.18	0.08
$x_2, y_1 \to z_2$	0.20	0.01	0.04
$x_2, y_3 \to z_2$	1.00	0.53	0.28

The corresponding flow graph is presented in Fig. 11.

From the flow graph we can see, e.g., that 71% ill and old patients have a positive test result, whereas 100% young healthy patients have a negative test result. From the inverse flow graph we can conclude that positive test result have mostly (67%) ill and old patients and negative test result display mostly (53%) young healthy patients.

Consequently, for the decision rule $x_1, y_1 \to z_1$ (and the inverse decision rule $z_1 \to x_1, y_1$) we have the dependency factor $\eta \approx 0.19$,whereas for the decision rule $x_2, y_3 \to z_2$ (and its inverse decision rule), we have $\eta \approx 0.31$.

That means that the relationship between young healthy patients and negative test results is more substantial then – between ill old patients and positive test result.

The strength of the corresponding decision rules is 0.32 and 0.28, respectively. Thus they are rather strong decision rules. As the result if we drop all remaining decision rules from the decision algorithm, we obtain a very simple decision algorithm consisting of two decision rules, with strength $0.32 + 0.28 = 0.60$. This means that two decision rules instead eight suffices for previous proper classification of initial data in 60% cases. Adding the next strongest decision rule $x_1, y_2 \to z_2$ with $\sigma = 0.14$, we get a decision algorithm with strength $0.60 + 0.14 = 0.74$, which can classify properly of 74% cases.

1.7 Flow Graphs and Rough Sets

In this section we show that some flow graphs can be treated as representations of approximation spaces. To explain this let us consider an example based on

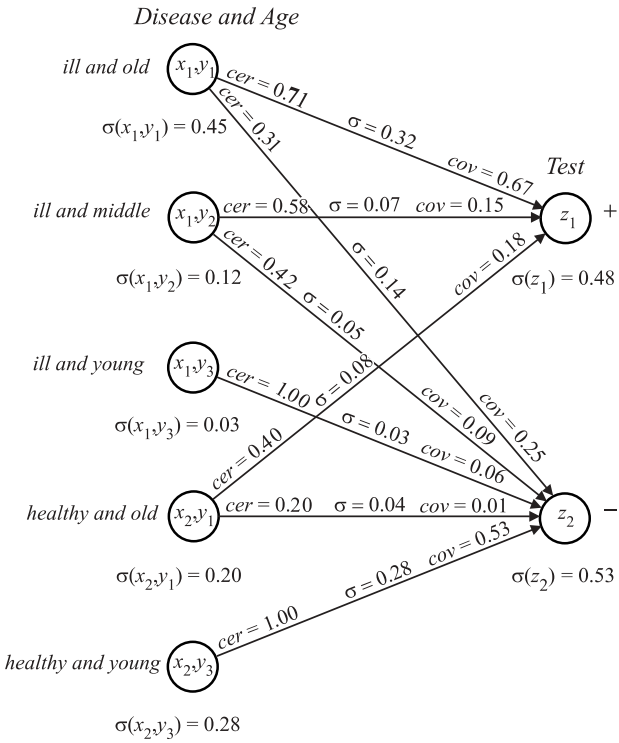

Fig. 11. Flow graph for the decision algorithm.

approximation spaces for information systems. Let us consider an information system $IS = (U, A)$ where U is the universe of objects and A is the set of attributes of the form $a : U \longrightarrow V_a$ [9]. Any such information system defines an approximation space $AS = (U, \mathcal{R}, \nu)$ [12] where $\mathcal{R}$ is a family of sets generated by descriptors over A, i.e.,

$$\mathcal{R} = \{X \subseteq U : X = \{u \in U : a(u) = v\} \text{ for some } a \in A,\ v \in V_a\} \tag{35}$$

and $\nu : P(U) \times P(U) \longrightarrow [0, 1]$ is the standard rough inclusion function defined by

$$\nu(X, Y) = \begin{cases} \frac{|X \cap Y|}{|X|} & \text{if } X \neq \emptyset \\ 1 & \text{if } X = \emptyset. \end{cases} \tag{36}$$

Hence, $\nu(X, Y)$ is a degree to which X is included in Y, for any $X, Y \in \mathcal{R}$.

Assume that $A = \{a_1, \ldots, a_m\}$ and $a_1 < \ldots < a_m$, i.e., A is linearly ordered by $<$.

Then one can construct a flow graph $G(AS) = (N, \mathcal{B}, \varphi)$ representing the approximation space $AS = (U, \mathcal{R}, \nu)$ where

1. $N = \{n_X : X \in \mathcal{R}\}$;
2. $n_X \mathcal{B} n_Y$ if and only if for some $a_i \in A, a_j \in A, v \in V_{a_i}, v' \in V_{a_j}$ we have $X = \{u \in U : a_i(u) = v\}$, $Y = \{u \in U : a_j(u) = v'\}$, and a_j is the immediate successor of a_i in the linear order $a_1, \ldots, a_m$;
3. For any nodes $n_X, n_Y \in N$:
   (a) $\varphi(n_X, n_Y) = |X \cap Y|/|U|$;
   (b) $cer(n_X, n_Y) = |X \cap Y|/|X|$;
   (c) $cov(n_X, n_Y) = |X \cap Y|/|Y|$.

Hence, the flow graph $G(AS)$ can be treated as a view of the approximation space AS relative to the given order $<$ of attributes from A. Such views as well as their fusions can be used in inducing patterns for concept approximations.

2 Applications

2.1 Introduction

In this section we give several tutorial examples showing how the presented ideas can be used in data analysis.

The examples have been chosen in such a way that various aspects of the proposed methodology are revealed.

In the example shown in section 2.2 (Smoking and Cancer) the probabilistic nature of data analysis is pointed out and relationship between statistical and flow diagram based methodology is revealed.

In the next example discussed in Section 2.3 (Hair, Eyes and Nationality) relationship between different sets of data is examined and the result need not to be necessarily interpreted in probabilistic terms.

Similar remark is valid for the next two examples.

Example given in Section 2.9 (Paint Demand and Supply) has entirely deterministic character and describes simply proportion between various ingredients.

In the remaining examples probabilistic character of data is rather immaterial and results can be understood as relationship between proportion of various features in the corresponding data sets.

Observe also that the numerical values of discussed coefficients may not satisfy exactly formulas given in the first chapter due to the round off errors in the computations.

2.2 Smoking and Cancer

In this section we show an application of the proposed methodology on the example taken from [5].

In Table 1 data concerning 60 people who do or do not smoke and do or do not have cancer are shown.

In Fig. 12 data given in Table 1 are presented as flow graph.

Normalized flow graph for the flow graph given in Fig. 12 is shown in Fig. 13.

From the flow graph we arrive at the following conclusions:

Table 1. Smoking and Cancer.

	Not smoke	Smoke	Total
Not cancer	40	10	50
Cancer	7	3	10
Total	47	13	60

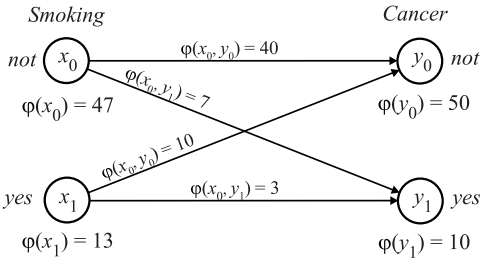

Fig. 12. Flow graph for Table 1.

- 85% non-smoking persons do not have cancer ($cer(x_0, y_0) = 40/47 \approx 0.85$),
- 15% non-smoking persons do have cancer ($cer(x_0, y_1) = 7/47 \approx 0.15$),
- 77% smoking persons do not have cancer ($cer(x_1, y_0) = 10/13 \approx 0.77$),
- 23% smoking persons do have cancer ($cer(x_1, y_1) = 3/13 \approx 0.23$).

From the flow graph we get the following reason for having or not cancer:

- 80% persons having not cancer do not smoke ($cov(x_0, y_0) = 4/5 = 0.80$),
- 20% persons having not cancer do smoke ($cov(x_1, y_0) = 1/5 = 0.20$),
- 70% persons having cancer do not smoke ($cov(x_0, y_1) = 7/10 = 0.70$),
- 30% persons having cancer do smoke ($cov(x_1, y_1) = 3/10 = 0.30$).

That means that not smoking persons mostly do not have cancer but smoking is mostly not associated with having cancer.

From the inverse flow graph, we conclude that the reason for having not cancer is not smoking but having cancer is not associated with smoking.

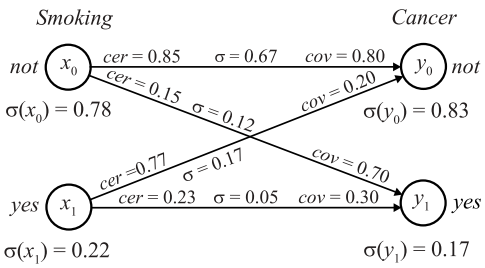

Fig. 13. Normalized flow graph for Table 1.

For the flow graph we have the following dependences: $\eta(x_0, y_0) = 0.01$, $\eta(x_0, y_0) = -0.06$, $\eta(x_1, y_0) = -0.09$ and $\eta(x_1, y_1) = 0.15$. These means that there is slight positive dependency between x_0 and y_0 and much stronger positive dependency between x_1 and y_1; x_0 and y_1 are negatively related and so are x_1 and y_0.

Let us also observe that in statistical terminology $\sigma(x_0), \sigma(x_1)$ are *priors*, $\sigma(x_0, y_0), \ldots, \sigma(x_1, y_1)$ are *joint distributions*, $cov(x_0, y_0), \ldots, cov(x_1, y_1)$ are *posteriors* and $\sigma(y_0)$, $\sigma(y_1)$ are *marginal probabilities*.

2.3 Hair, Eyes and Nationality

In Fig. 14 the relationship between color of eyes, color of hair and nationality is presented in the form of a flow graph.

That means that in this population there are 60% blond, 30% dark and 10% red haired; 80% blond haired have blue eyes whereas 20% blond haired have hazel eyes, etc. Similarly we see from the flow graph that 20% persons having blue eyes are Italian, and 80% persons with blue eyes are Swede, etc.

First let us compute "flow" in the graph and the result is shown in Fig. 15.

We can see from the flow graph that the strongest decision rules showing the relationship between color of hair and eyes are $x_1 \rightarrow y_1$ ($\sigma = 0.48$) and $x_2 \rightarrow y_2$ ($\sigma = 0.27$) with $\eta = 0.14$ and $\eta = 0.38$ respectively. These two decision rules have strength $0.48 + 0.27 = 0.75$. The dependency factors of these decision rules indicate that the relationship between dark hair and hazel eyes is much stronger then the dependency between blond hair and blue eyes.

Similarly the strongest decision rules showing the relationship between color of eyes and nationality are $y_1 \rightarrow z_2$ ($\sigma = 0.48$) and $x_2 \rightarrow z_1$ ($\sigma = 0.36$) with $\eta = 0.21$ and $\eta = 0.30$, respectively and strength 0.84. This shows that hazel eyes are more characteristic for Italians, then blue eyes for Swede.

The relationship between color of hair and nationality is computed using the concept of fusion and the result is shown in Fig. 16.

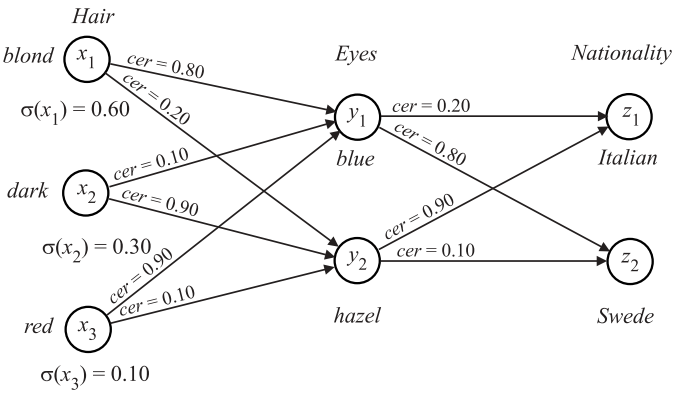

Fig. 14. Initial data.

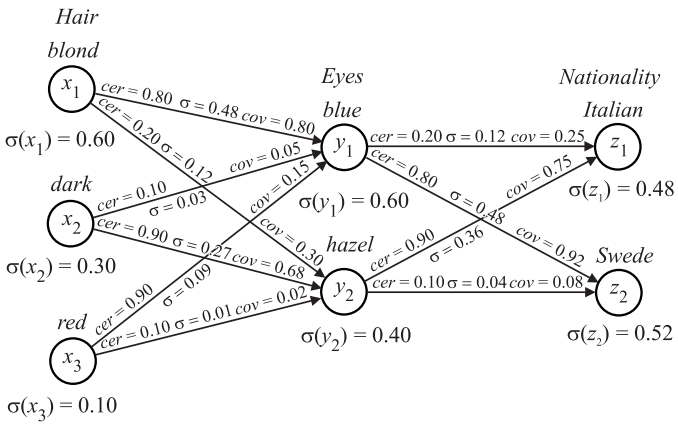

Fig. 15. Relationship between color of eyes, color of hair and nationality.

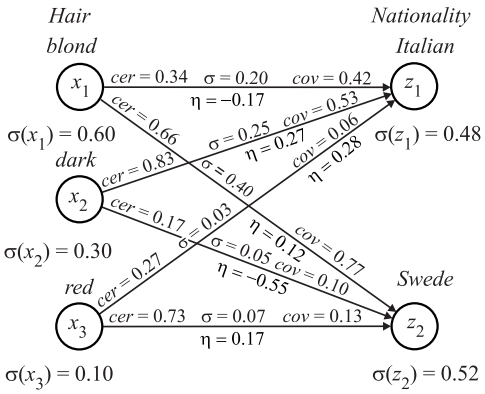

Fig. 16. Fusion of color of hair and nationality.

In this flow graph also degree of independence is given. We can see from the dependency coefficients that, e.g., there is a relatively strong negative dependency between dark hair and Swede ($\eta = -0.55$) blond hair and Italian ($\eta = -0.17$), but there is a positive dependency between dark hair and Italian ($\eta = 0.25$), however the first dependency has very low strength ($\sigma = 0.05$), whereas the second one has much higher strength ($\sigma = 0.20$). This means that in the first case 5% of the population display this property in contrast to the second case where 20% of the population support the dependency.

Let us also observe that the three decision rules $x_1 \to z_1$ ($\sigma = 0.20$), $x_2 \to z_1$ ($\sigma = 0.25$) and $x_1 \to z_1$ ($\sigma = 0.40$) have very high strength 0.85.

The decision algorithm induced by the flow graph shown in Fig. 15 is presented in table below:

	certainty	coverage	strength
$x_1, y_1 \rightarrow z_1$	0.20	0.20	0.10
$x_1, y_1 \rightarrow z_2$	0.81	0.74	0.39
$x_1, y_2 \rightarrow z_1$	0.92	0.23	0.11
$x_1, y_2 \rightarrow z_2$	0.08	0.02	0.01
$x_2, y_1 \rightarrow z_1$	0.33	0.05	0.01
$x_2, y_1 \rightarrow z_2$	0.67	0.04	0.02
$x_2, y_2 \rightarrow z_1$	0.89	0.51	0.24
$x_2, y_2 \rightarrow z_2$	0.11	0.05	0.03
$x_3, y_1 \rightarrow z_1$	0.22	0.05	0.02
$x_3, y_1 \rightarrow z_2$	0.78	0.14	0.07
$x_3, y_2 \rightarrow z_1$	1.00	0.02	0.01
$x_3, y_2 \rightarrow z_2$	0.00	0.00	0.00

Flow graph associated with the decision algorithm is shown in Fig. 17.

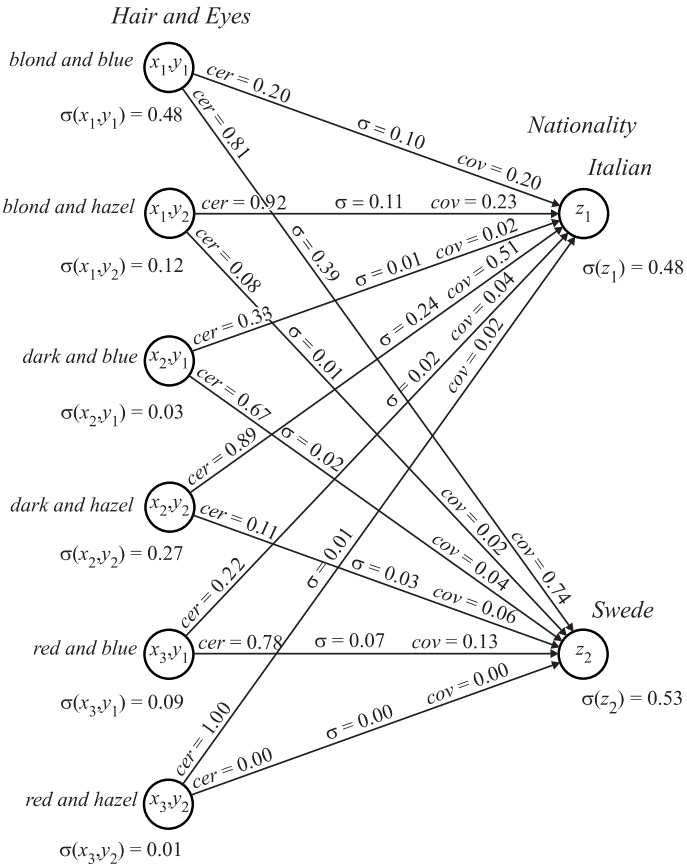

Fig. 17. Hair, eyes and nationality.

One can conclude from the flow graph that the most significant decision rules are $(x_1, y_1) \to z_2$ ($\sigma = 0.39$) and $(x_2, y_2) \to z_1$ ($\sigma = 0.24$) with the corresponding dependency factors $\eta = 0.10$ and 0.26, and strength $0.39 + 0.24 = 0.63$. That means that two decision rules enable us to classify correctly of the 63% cases.

Dependency factors indicate that dark hair and hazel eyes are more characteristic for Italians then blond hair and blue eyes for Swede.

Let us also mention that if the data are representative for a larger universe (form a proper sample of the universe), then the results can be also considered as promising hypotheses in this extended world. That is, we employ in this case inductive reasoning, i.e., induce from properties of a part of the universe properties of the whole universe.

2.4 Production Quality

Consider three industrial plants x_1, x_2 and x_3 producing three kinds of appliances y_1, y_2 and y_3. Some of the produced appliances are defective. The situation is presented in Fig. 18.

We want to find the relationship between plant and quality of products.

First we compute flow in the flow graph and the result is shown in Fig. 19.

Similarly as in the previous example we can find from the flow graph that the most significant decision rules describing the relationship between plant and product are $x_2 \to y_2$, $x_3 \to y_2$ and $x_3 \to y_3$ having together strength $0.18 + 0.10 + 0.40 = 0.68$, whereas the relationship between products and quality is best described by the decision rules $y_2 \to z_1$, $y_3 \to z_1$ and $y_3 \to z_2$ with strength $0.25 + 0.44 + 0.19 = 0.88$

In order to find relationship between producers and quality of their products, we compute the corresponding fusion and the result is given in Fig. 20. It is seen from the dependency coefficient that all decision rules except the rule $x_2 \to z_2$ have rather low rather low dependency factor. Because $\eta(x_2, z_2) = -0.17$ plant

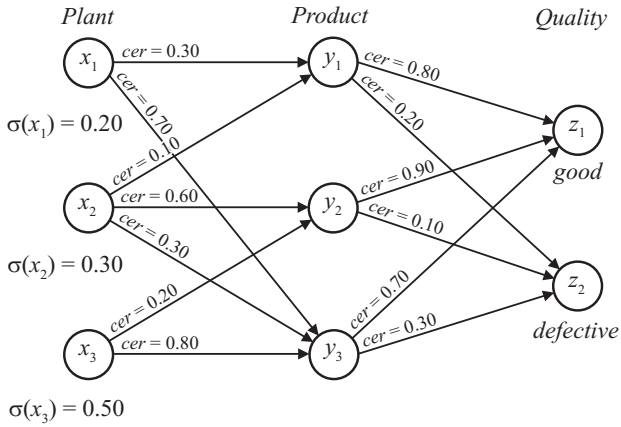

Fig. 18. Relationship between plant, product, and quality.

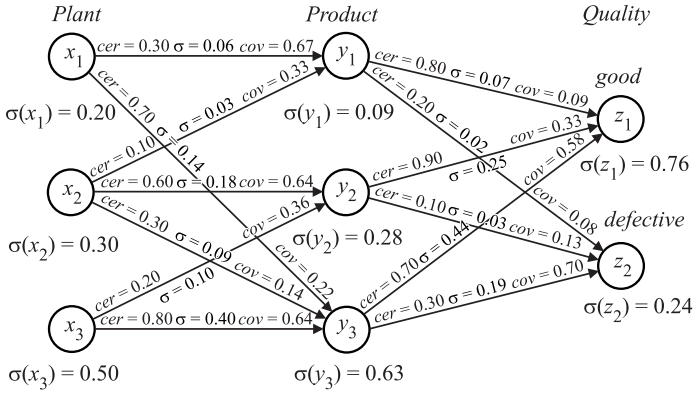

Fig. 19. Relationship between plants, products and quality.

x_2 produces the lowest ratio of defective products but the corresponding decision rule has a rather weak strength ($\sigma = 0.05$); that is, it has 5% support. In other words it is the best of all plants. It is interesting to observe that the four strongest decision rules provide 90% accuracy of classification.

The corresponding decision algorithm is shown below:

	certainty	coverage	strength
$x_1, y_1 \to z_1$	0.83	0.06	0.05
$x_1, y_1 \to z_2$	0.17	0.05	0.01
$x_1, y_3 \to z_1$	0.71	0.13	0.10
$x_1, y_3 \to z_2$	0.29	0.15	0.04
$x_2, y_1 \to z_1$	0.00	0.21	0.00
$x_2, y_1 \to z_2$	0.00	0.03	0.00
$x_2, y_2 \to z_1$	0.89	0.21	0.16
$x_2, y_2 \to z_2$	0.11	0.08	0.02
$x_2, y_3 \to z_1$	0.67	0.08	0.06
$x_2, y_3 \to z_2$	0.33	0.15	0.03
$x_3, y_2 \to z_1$	0.90	0.13	0.09
$x_3, y_2 \to z_2$	0.10	0.05	0.01
$x_3, y_3 \to z_1$	0.70	0.37	0.28
$x_3, y_3 \to z_2$	0.30	0.45	0.12

Flow graph associated with the decision algorithm is given in Fig. 21.

We leave discussion of the flow graph for the interested reader.

Let us observe that in this example we do not have inductive reasoning, whatsoever. The world (universe) we are interested in is "closed" and we search only for relationships valid in this specific universe. There is no reason to generalize the obtained results.

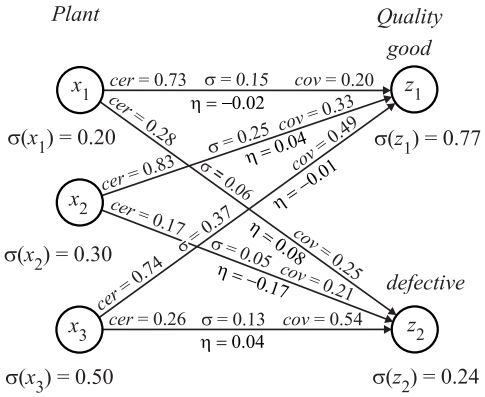

Fig. 20. Fusion between plant and quality.

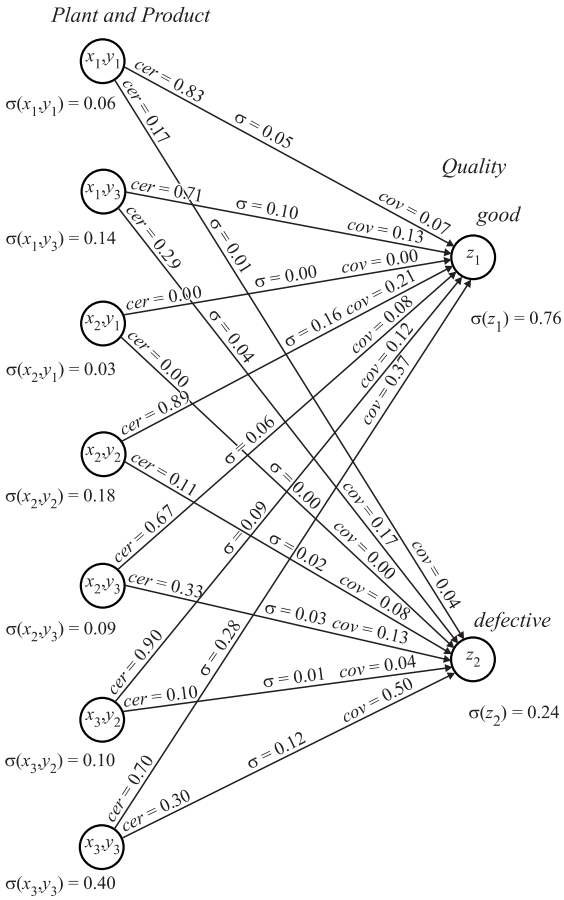

Fig. 21. Production quality.

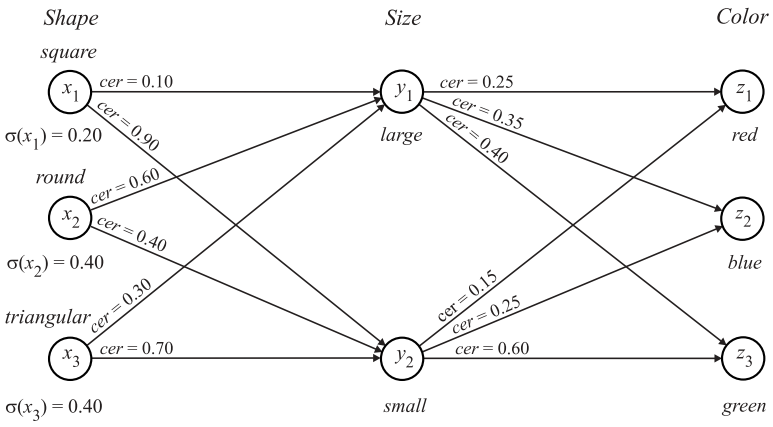

Fig. 22. Initial data.

2.5 Play Blocks

Suppose we are given a set of play blocks of different, shapes (square, round, triangular), colors (red, blue green) and size (large, small).

Initial data are shown in Fig. 22.

Corresponding flow graph is presented in Fig. 23.

In order to find relationship between shape and size, and size color we have to compute the corresponding dependency factors but we will omit this computation here. For finding the relationship between shape and color we have to compute first fusion of shape and color, which is shown in Fig. 24.

Almost all dependency coefficients are very low, which means that there is a very low relationship between shape and color of blocks, nevertheless there are strong decision rules in the flow graph, e.g., $x_3 \rightarrow z_3(\sigma = 0.22)$, $x_3 \rightarrow z_2(\sigma =$

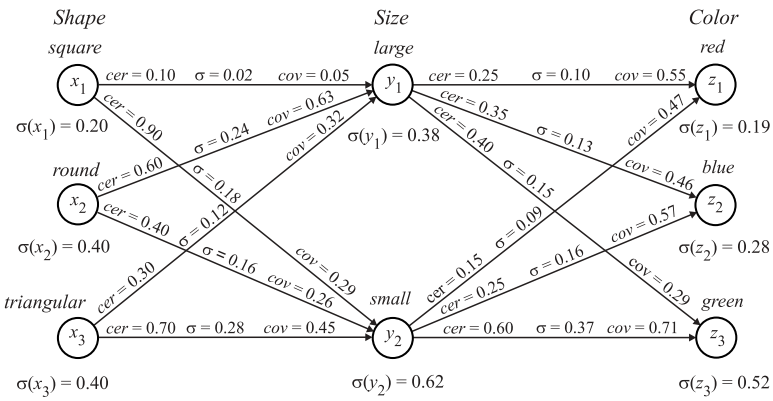

Fig. 23. Relationship between features of play blocks.

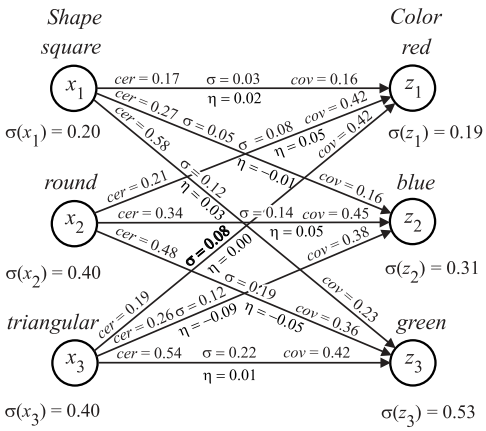

Fig. 24. Fusion of shape and color.

0.12), $x_2 \rightarrow x_3(\sigma = 0.19)$ and $x_2 \rightarrow z_2(\sigma = 0.14)$, which all together yields 67% accuracy of classification.

Analogously to the example discussed before (see Fig. 21) one can search for relationship between other features of play blocks.

Also in this example, similarly as in the previous one, we are not interested in inducing general rules. We are searching only here for relationships in a given data set, i.e., relations between various properties of a given set of objects.

2.6 Preference Analysis

Suppose that three models of cars x_1, x_2 and x_3 are sold to three disjoint groups of customers z_1, z_2 and z_3 through four dealers y_1, y_2, y_3 and y_4.

Moreover, let us assume that car models and dealers are distributed as shown in Fig. 25.

Computing strength and coverage factors for each branch we get results shown in Fig. 26.

In order to find consumer preferences in buying cars we have to compute fusion between car models and consumer group. The result is shown in Fig. 27.

From the flow graph we can see that consumer group z_1 mostly bought car x_3 (45%), consumer group x_2 mostly bought car x_3 (38%) and consumer group z_3 mostly bought cars x_3 too (69%). We can also conclude from the flow graph that car x_1 was mostly bought by consumer group z_2 (57%), car x_2 – by consumer group z_2 (60%) and car x_3 – by consumer group z_2 (39%).

The dependency coefficients reveal that the strangest negative dependency is between car model x_1 and consumer group $z_3(\eta = -0.37)$, whereas car model x_1 and consumer group z_1 shows the highest positive correlation ($\eta = 0.17$), with corresponding strengths $\sigma = 0.02$ and $\sigma = 0.06$.

Let us also notice that the five strongest decision rules provided 77% accuracy of classification.

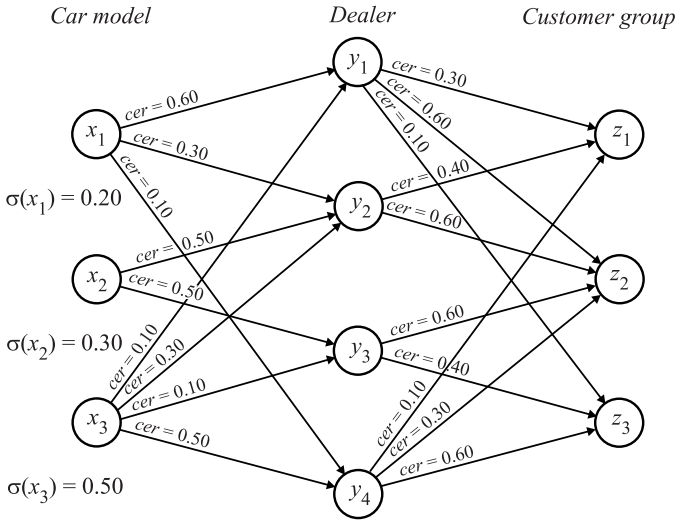

Fig. 25. Car and dealer distribution.

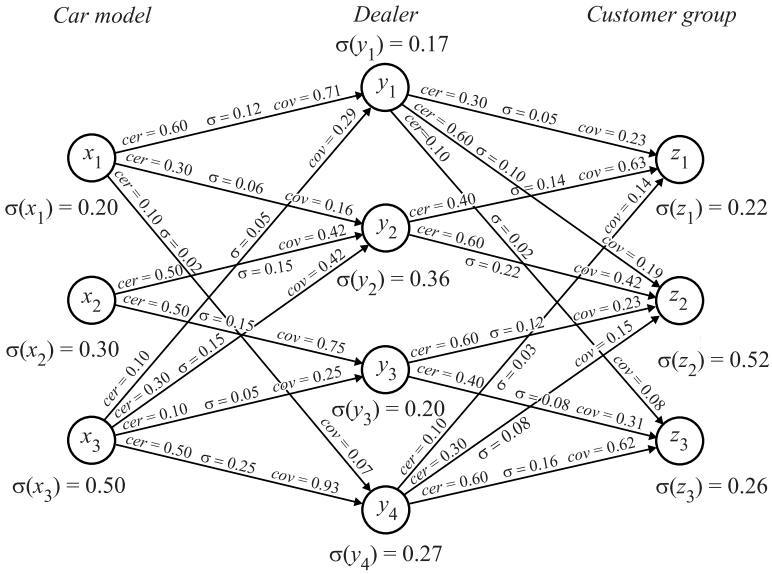

Fig. 26. Strength, certainty and coverage factors.

We can also ask how consumer preferences are related to car model and dealer. To this end we have to find the corresponding decision algorithm but we postpone this task here.

If this data set is representative for a greater universe then the obtained results can be induced for the whole universe.

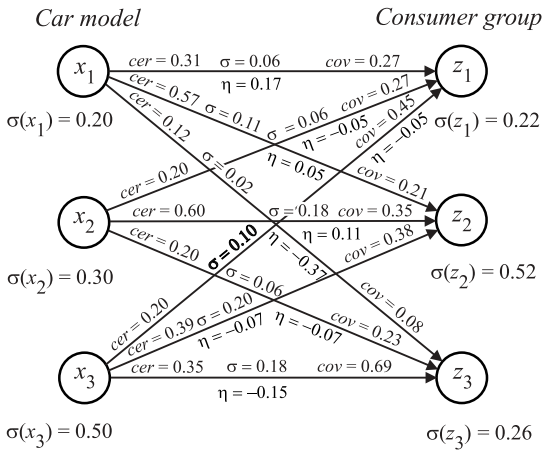

Fig. 27. Fusion of consumer preferences.

2.7 Voting Analysis

Consider three disjoint age groups of voters y_1 (*old*), y_2 (*middle aged*) and y_3 (*young*) – belonging to three social classes x_1 (*high*), x_2 (*middle*) and x_3 (*low*). The voters voted for four political parties z_1 (*Conservatives*), z_2 (*Labor*), z_3 (*Liberal Democrats*) and z_4 (*others*).

Social class and age group votes distribution is shown in Fig. 28.

First we want to find votes distribution with respect to age group. The result is shown in Fig. 29.

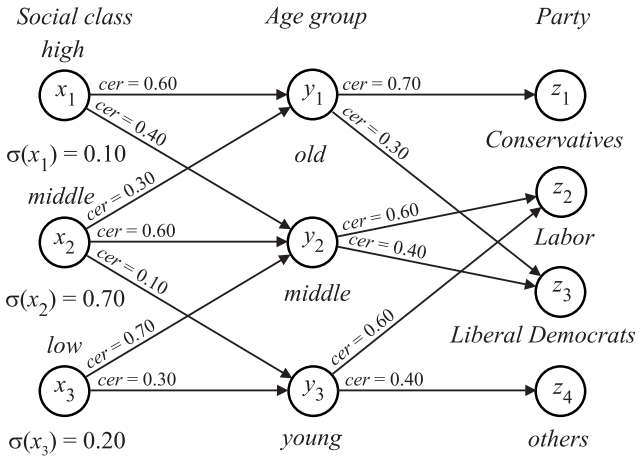

Fig. 28. Social class and age group votes distribution.

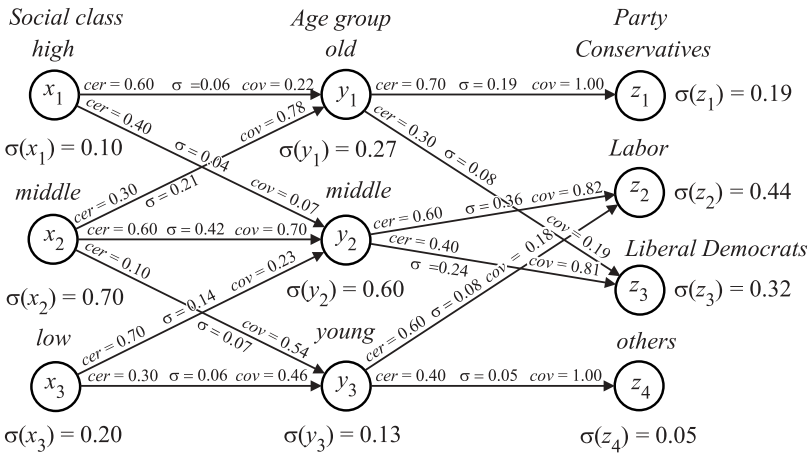

Fig. 29. Party votes distribution.

From the flow graph presented in Fig. 29 we can see that, e.g., party z_1 obtained 19% of total votes, all of them from age group y_1; party z_2 – 44% votes, which 82% are from age group y_2 and 18% – from age group y_3, etc.

If we want to know how votes are distributed between parties with respect to social classes, we have to compute fusion of the corresponding graph. Employing the algorithm presented previously we get the results shown in Fig. 30.

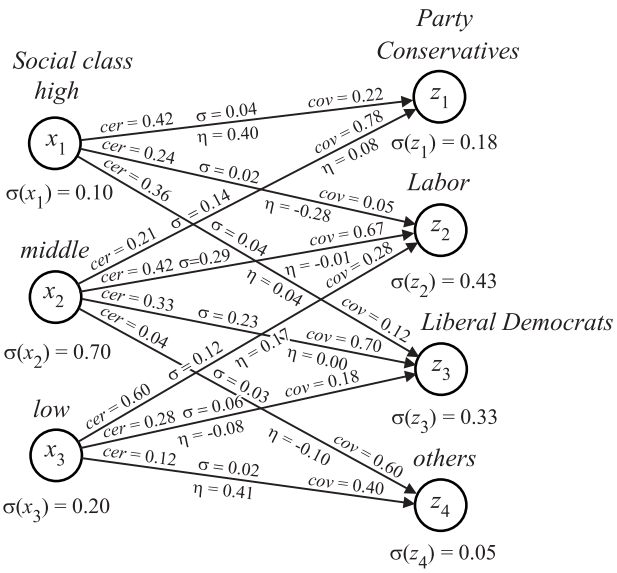

Fig. 30. Fusion of social class and party.

From the flow graph presented in Fig. 30, we can see that party z_1 obtained 22% votes from social class x_1 and 78% – from social class x_2, etc.

The strongest positive dependency occurs between social class x_1 (*high*) and party z_1 (*Conservatives*) $\eta = 0.40$ and social class x_3 (*low*) and party z_4 (*others*) $\eta = 0.41$, with corresponding strengths $\sigma = 0.04$ and $\sigma = 0.02$, which are rather low.

The highest negative correlation ($\eta = -0.28$) is between social class x_1 (*high*) and political party z_2 (*Labor*), with strength $\sigma = 0.02$, which is also low.

If we want to know how votes for political parties are distributed in relation to social class and age of voters, we have to derive the decision algorithm from the flow graph given in Fig. 30, but we will drop this here. Let us observe only, e.g., that old members of high social class voted mostly for Conservatives, middle aged members of middle social class voted mostly for Labor and young members of low social class voted mostly for Labor.

Let us also observe that the four strongest decision rules yields 0.78 strength, i.e., these four rules gives 78% accuracy of classification of party members with respect to social class.

A similar remark about induction as in the previous case of voting analysis applies here.

2.8 Promotion Campaign

Suppose we have three groups of customers classified with respect to age: *young* (*students*), *middle aged* (*workers*) and *old* (*pensioners*). Moreover, suppose we have data concerning place of residence of customers: *town, village* and *country.*

Let us assume that the customers are asked whether they will buy certain advertised product (e.g., a new tooth paste) in a promotion campaign.

The initial data are presented in Fig. 31.

That means that there are 25% young customers, 60% – middle aged and 15% old – in the data base. Moreover, we know that 75% of young customers live in towns, 20% – in villages and 5% – in the country, etc. We also have from the database that 75% town customers answered yes, 25% – no, etc.

We want to find a relationship between various customers' group and the final result of the promotion.

First, applying the ideas presented previously, we get the results shown in Fig. 32.

Fig. 32 shows the general structure of patterns between various customers groups and promotion results.

Suppose we are interested in finding the relationship between age group and final result of the promotion. To this end we have to compute fusion between age groups and the promotion result, or – the relationship between input and output of the flow graph. The result is shown in Fig. 33.

Fig. 33 contains also dependency factors between age groups and the promotion result.

It can be seen from the flow graph that all the dependency factors are very low and almost close to zero. That means, that in view of the data, practically,

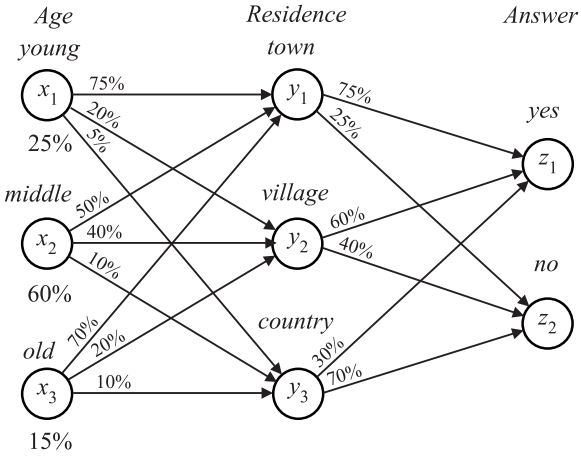

Fig. 31. Initial data.

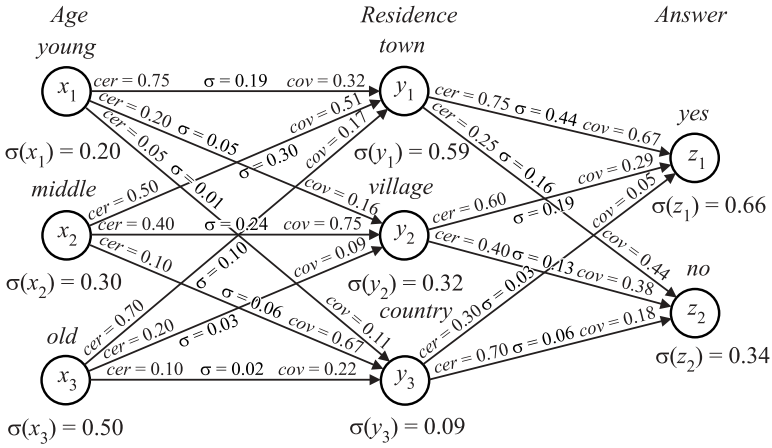

Fig. 32. Relationship between customers group and promotion.

there is no relationship between age group of customers and the final result of promotion, but there are three strong decision rules which provide all together 79% of classification accuracy.

We might be also interested in the relationship between customer's residence and promotion results. This relationship is shown in Fig. 34. We can see from the flow graph that there is relatively high negative dependency ($\eta = -0.38$) with strength $\sigma = 0.03$ between country customers group y_3 and answer z_1 (*yes*). Similarly there is high positive dependency ($\eta = 0.35$) with strength $\sigma = 0.16$ between country customers group y_3 and answer z_2 (*no*). There is also a substantial degree of negative dependency ($\eta = -0.16$) between town customers group y_1 and answer z_2 (*no*), with $\sigma = 0.16$.

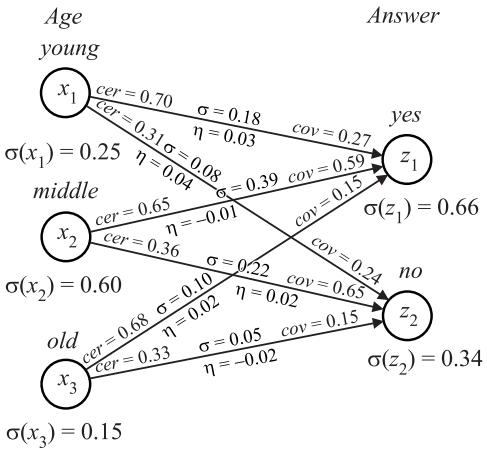

Fig. 33. Fusion of age group and promotion.

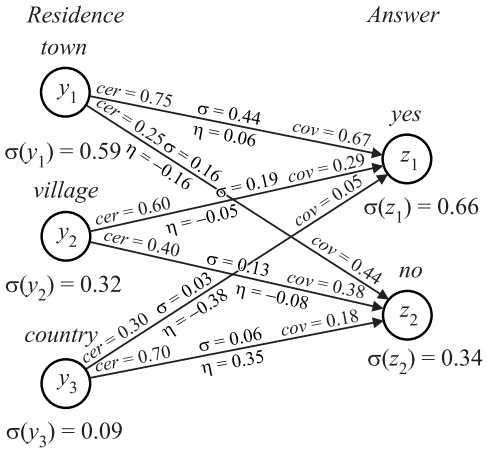

Fig. 34. Fusion of residence and promotion.

We can conclude from the flow graph in Fig. 34, e.g., that independently of age town customers mostly give positive answer to the promotion campaign and country customers give mostly negative answer to the promotion campaign.

Certainly, the results are valid only for the considered data. For another data (population), the results can be different.

2.9 Paint Demand and Supply

Suppose that cars are painted in two colors y_1 and y_2 and that 60% of cars have color y_1, whereas 40% cars have color y_2. Moreover, assume that colors y_1 and y_2 can be obtained by mixing three paints x_1, x_2 and x_3 in the following proportions:

– y_1 contains 20% of x_1, 70% of x_2 and 10% of x_3,
– y_2 contains 30% of x_1, 50% of x_2 and 20% of x_3.

We have to find the demand for each paint and supply among cars y_1 and y_2.

Employing terminology introduced in previous section, we can represent our problem by means of the flow graph shown in Fig. 35.

Thus, in order to solve our task, first we have to compute strength of each branch. Next, we compute the demand for each paint. Finally, we compute paint supply for each car. The final result is presented in Fig. 36.

Suppose now that the cars are produced by three manufacturers z_1, z_2 and z_3, in proportions shown in Fig. 37.

That means:

– 50% of cars y_1 are produced by manufacturer z_1
– 30% of cars y_1 are produced by manufacturer z_2
– 20% of cars y_1 are produced by manufacturer z_3

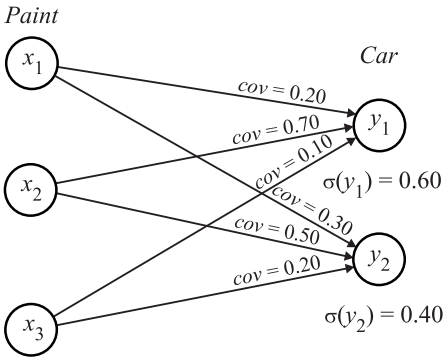

Fig. 35. Paint demand.

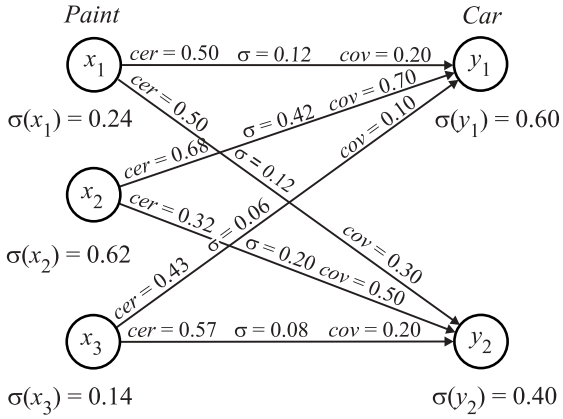

Fig. 36. Paint supply.

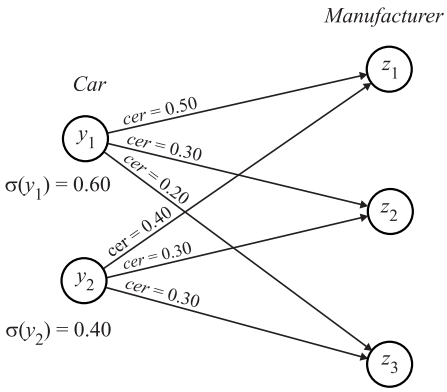

Fig. 37. Car production distribution.

and

- 40% of cars y_2 are produced by manufacturer z_1
- 30% of cars y_2 are produced by manufacturer z_2
- 30% of cars y_2 are produced by manufacturer z_3

Employing the technique used previously, we can compute car production distribution among manufacturers as shown in Fig. 38.

For example, manufacturer z_1 produces 65% of cars y_1 and 35% of cars y_2, etc. Finally, the manufacturer z_1 produces 46% cars, manufacturer z_2 – 30% cars and manufacturer z_3 – 24% of cars.

We can combine graphs shown in Fig. 36 and Fig. 38 and we obtain the flow graph shown in Fig. 39.

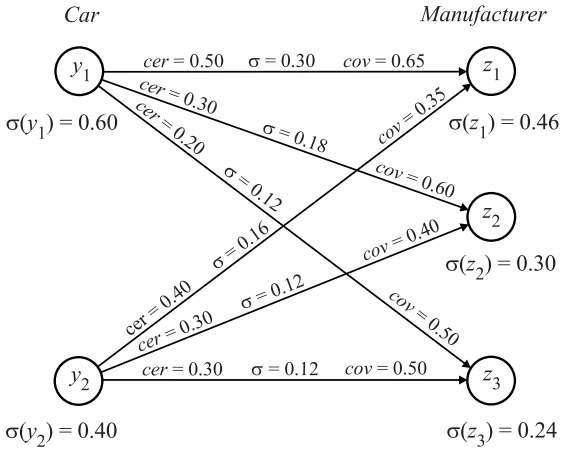

Fig. 38. Manufacturer distribution.

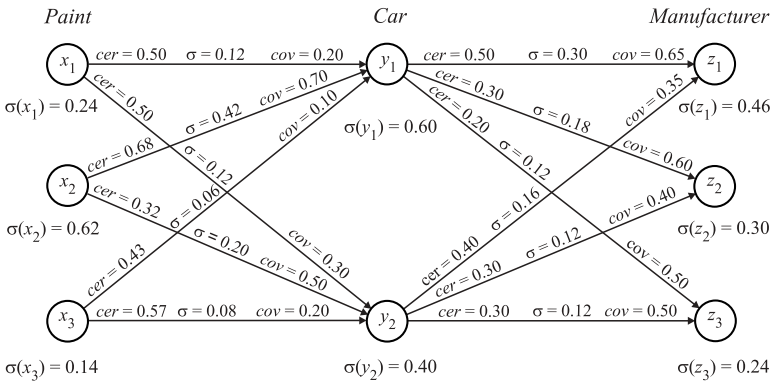

Fig. 39. Paint supply demand flow.

In order to find paint demand and supply by each manufacturer, we have to compute fusion between each paint and manufacture. The corresponding flow graph is presented in Fig. 40.

The meaning of the obtained results is the following.

Suppose that paints are delivered in the same units, say kg.

Thus manufacturer, e.g., z_1, demands 120 kg, 290 kg and 60 kg of paints x_1, x_2 and x_3, respectively. Whereas paint x_1 is delivered to manufacturer z_1, z_2 and z_3 in amounts 120 kg, 80 kg and 70 kg, respectively.

Consequently, we need 270 kg of paint x_1, 630 kg of paint x_2 and 140 kg of paint x_3.

Observe that this example has an entirely deterministic character and there is no probabilistic interpretation of the results needed whatsoever. Besides, we do not need to employ a decision algorithm to solve this task.

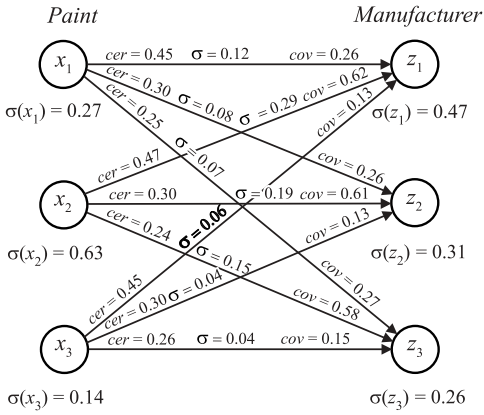

Fig. 40. Fusion of paint demand and supply flow.

In the examples in the previous sections, we considered sets of different objects, e.g., patients, customers, voters, play blocks, cars, etc. In this example we have an entirely different situation. We analyze various paints which are not sets but substances not consisting of elements but having various ingredients (which are also substances), e.g., blue, red paint etc. Thus we cannot use here set theoretical language, and define union, intersection or inclusion of sets (paints). Therefore we cannot say that blue paint is a subset of green paint, but that blue paint is an ingredient of green paint. Consequently, a flow graph can be in this case understood as a language for description of the relationship between various ingredients (substances), where $(x, y) \in \mathcal{B}$ means that x is ingredient of y. In this language $cer(x, y)$ expresses the ratio of substance y to substance x, in x, whereas $cov(x, y)$ is the ratio of x to y in y. This resembles somewhat the relation of being a part in a degree introduced in rough mereology by Polkowski and Skowron (see Section 1.7) but parts and ingredients are two different concepts. The concept of a part has set theoretical flavor, but ingredient has not.

Also inductive reasoning is not involved here. This example shows simply the relationship between demand and supply of some goods.

3 Conclusions

We propose in this paper a new approach to knowledge representation and data mining based on flow analysis in a new kind of flow network.

We advocate in this paper to represent relationships in data by means of flow graphs. Flow in the flow graph is meant to capture the structure of data rather than to describe any physical material flow in the network. It is revealed that the information flow in a flow graph is governed by Bayes' formula; however, the formula can be interpreted in an entirely deterministic way without referring to its probabilistic character. This representation allows us to study different relationships in data and can be used as a new mathematical tool for data mining.

Acknowledgments

Thanks are due to Professor Andrzej Skowron and Dr. Dominik Ślęzak for critical remarks.

References

1. J. M. Bernardo, A. F. M. Smith, Bayesian Theory. Wiley series in probability and mathematical statistics. John Wiley & Sons, Chichester, New York, Brisbane, Toronto, Singapore, 1994.
2. M. Berthold, D.J. Hand, Intelligent Data Analysis - An Introduction. Springer-Verlag, Berlin, Heidelberg, New York, 1999.
3. G.E.P. Box, G.C. Tiao, Bayesian Inference in Statistical Analysis. John Wiley and Sons, Inc., New York, Chichester, Brisbane, Toronto, Singapore, 1992.

4. L.R. Ford, D.R. Fulkerson, Flows in Networks. Princeton University Press, Princeton. New Jersey, 1962.
5. Ch. M. Grinstead, J. L. Snell, Introduction to Probability: Second Revised Edition American Mathematical Society, 1997.
6. S. Greco, Z. Pawlak, R. Słowiński, Generalized decision algorithms, rough inference rules and flow graphs. In: J.J. Alpigini, J.F. Peters, A. Skowron, N. Zhong (eds.), Rough Sets and Current Trends in Computing. Lecture Notes in Artificial Intelligence 2475, Springer-Verlag, Berlin, 2002, pp. 93-104.
7. S. Greco, Z. Pawlak, R. Słowiński, Bayesion confirmation measures within rough set approach, In: S. Tsumoto, R. Słowiński, J. Komorowski, J. Grzymała-Busse (eds.), Rough Sets and Current Trends in Computing (RSCTC 2004), Lecture Notes in Artificial Intelligence 3066, Springer Verlag, Berlin, 2004, pp. 261-270.
8. J. Łukasiewicz, Die logishen Grundlagen der Wahrscheinilchkeitsrechnung. Kraków (1913), in: L. Borkowski (ed.), Jan Łukasiewicz - Selected Works, North Holland Publishing Company, Amsterdam, London, Polish Scientific Publishers, Warsaw, 1970, pp. 16-63.
9. Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning about Data. Kluwer, Dordrecht, 1991.
10. Z. Pawlak, Rough sets, decision algorithms and Bayes' theorem. European Journal of Operational Research 136, 2002, pp. 181-189.
11. Z. Pawlak, Flow graphs, their fusion and data analysis, 2003, to appear.
12. A. Skowron, J. Stepaniuk, Tolerance approximation spaces, Fundamenta Informaticae 27(2-3), 1996, pp. 245-253.
13. R. Swinburne (ed.), Bayes' Theorem, Oxford University Press, 2002.
14. S. Tsumoto, H. Tanaka, Discovery of Functional Components of Proteins Based on PRIMEROSE and Domain Knowledge Hierarchy, Proceedings of the Workshop on Rough Sets and Soft Computing (RSSC-94), 1994: Lin, T.Y., and Wildberger, A.M. (Eds.), Soft Computing, SCS, 1995, pp. 280-285.
15. S.K.M. Wong, W. Ziarko, Algorithm for inductive learning. Bull. Polish Academy of Sciences 34, 5-6, 1986, pp. 271-276.