DOI: 10.15439/2015F327
ACSIS, Vol. 5

Introduction to Knowledge Discovery in Medical Databases and Use of Reliability Analysis in Data Mining

Elena Zaitseva, Miroslav Kvassay, Vitaly Levashenko, Jozef Kostolny
University of Zilina,
Faculty of Management Science and Informatics
Zilina, Slovakia
Email: {elena.zaitseva, miroslav.kvassay, vitaly.levashenko, jozef.kostolny}@fri.uniza.sk

Abstract—**Data mining (DM) is a collection of algorithms that are used to find some novel, useful and interesting knowledge in databases. DM algorithms are based on applied fields of mathematics and informatics, such as mathematical statistics, probability theory, information theory, neural networks. Some methods of these fields can be used to find hidden relation between data, what can be used to create models that predict some behavior or describe some common properties of analyzed objects. In this paper, we combine methods of DM with tools of reliability analysis to investigate importance of individual database attributes. Results of such investigation can be used in database optimization because it allows identifying attributes that are not important for purposes for which the database is used. Our approach is based on some coincidence between the key terms of DM and reliability analysis.**

I. Introduction

ONE of the current problems of modern medicine is processing and analysis of a huge amount of data that are generated by medical systems. This calls for design of automatic or semi-automatic process that could be used to find useful and understandable knowledge from data. This process is known as a Knowledge Discovery in Databases (KDDs) and it involves an (semi-)automatic, exploratory analysis and modeling of large data repositories to identify valid, novel, useful, and understandable patterns from large and complex datasets [1].

The main idea of a KDD is to transform a huge amount of row data into useful information and knowledge that can be very easy interpreted. In general, data represent basic facts and statistics without any context. As an example, let us consider numbers "100" or "22.3" or simple value "no". When we add a context to the data, then we get information. For example, a patient with plasma glucose concentration at 2 hours in an oral glucose tolerance test of 100 and body mass index (patient's body mass in kg divided by the square of its height in m) of 22.3 does not suffer diabetes. So, the basic difference between data and information is in its information value, i.e. data have no information value themselves in compare to information.

If we have enough information from some domain, then it can be possible to identify some general facts that characterize the domain. These general facts are known as knowledge. For example, let us consider the example of relation between plasma glucose concentration, body mass index and diabetes of a patient. If we have such information about many patients, then, for example, we can derive that there is a little probability that patients with plasma glucose concentration under 127 and body mass index under 24.6 suffer diabetes. This knowledge can be identified in dataset [2] that contains 768 records about the diabetes incidence in the Pima Indian population living near Phoenix in Arizona.

The relation between data, information and knowledge is very often described by the knowledge pyramid (Fig. 1). It expresses that a huge amount of data can be transformed into information by adding a context and, then, analysis and aggregation of information can lead in the discovery of general patterns in data that represent knowledge about the studied domain. One of the most popular terms for the second phase is Data Mining (DM).

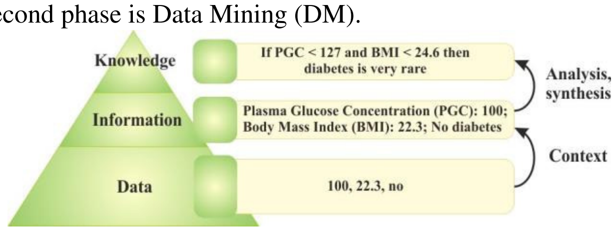

Fig. 1 Knowledge pyramid

II. Knowledge Discovery Process and Data Mining

In general, DM is a collection of methods focused on building model and finding patterns or trends in data. When DM is used in a process in which their outcome is evaluated, so that we can think about the product as being a new package of information, then we speak about a knowledge discovery process [3].

This work was supported by the grant of 7th RTD Framework Program No 610425 (RASimAs).

A knowledge discovery process or KDD is a very complex non-linear process that involves not only data analysis but also its preparation as well as knowledge interpretation and using the discovered knowledge. A KDD involves six important steps that are [3], [4]: understanding the problem domain, understanding the data, preparation of the data, data mining, evaluation of the discovered knowledge and using the knowledge. The term non-linear means that, in any step, there can be identified some problems that cause need to return to some of the previous steps and repeat the whole process from that step.

A KDD is not a one-pass process [3], and its every iteration gives a different view on the data. For example, in the preprocessing phase, we can identify quite a lot of missing data, so their prediction could become the goal of the first iteration of the KDD. In the second iteration, we can focus on the creation of a model that could be used to predict patient's diagnosis from that data (predictive DM) or to identify patients with similar relations between their symptoms and diagnosis (descriptive DM).

A. *Understanding the Medical Problem Domain*

At the beginning of a KDD, we need to identify what type of knowledge should be found in the data. This requires understanding the problem domain. In case of medical data, the domain comes from medical area and, therefore, the main goals of this phase are [3]:

a) translation of medical goals into DM ones, and
b) determination of success criteria from the medical and DM point of view.

B. *Understanding the Data*

When we understand the medical problem domain, then the data that are available should be analyzed. This analysis identifies which data will be used, and which additional information will be needed. The main goal of this step is to create a dataset for the next steps of KDD.

The result of this step can be very often interpreted as a table with rows and columns. The columns represent individual attributes of analyzed data while rows agree with individual records (instances/patients). Examples of two datasets are in Fig. 2. The first one contains 5 attributes related to cancer and it stores information about 14 patients. The second table is a sample of dataset focused on the diabetes incidence in the Pima Indian population [2]. It has 9 attributes and 14 records (the all dataset has 768 rows).

In Fig. 2, we can see that two different types of attributes exist – categorical and numerical [5] (Fig. 3). Categorical attributes are non-numerical and, usually, they have several possible values. They are also known as qualitative because they describe an object from qualitative point of view. The categorical attributes can be split into two separate groups: nominal and ordinal. The basic difference between them is that nominal attributes contain data that cannot be ordered, i.e. there is define no relation such as "greater/better than" or "lower/worse than", while ordinal ones are defined on data that can be ordered in some way. As an example, let us consider blood groups (A, B, AB, 0) and pain degree (severe, mild, none). There is no reason to assume that blood group A is better than 0 but, in case of pain degree, it is clear that no pain is better than severe one.

Columns (attributes)

Rows (records)

No.	Tumor	Blood Group	Heredity	Age	Cancer
1	confirmed	A	yes	younger	high
2	confirmed	A	yes	elder	high
3	no	B	yes	younger	low
4	non confirmed	AB	yes	younger	low
5	non confirmed	0	no	younger	low
6	non confirmed	0	no	elder	high
7	no	0	no	elder	low
8	confirmed	AB	yes	younger	high
9	confirmed	0	no	younger	low
10	non confirmed	A	no	younger	low
11	confirmed	A	no	elder	low
12	no	B	yes	elder	low
13	no	B	no	younger	low
14	non confirmed	0	yes	elder	high

No.	Number of times pregnant	Plasma glucose concentration	Diastolic blood pressure	Triceps skin fold thickness	2-Hour serum insulin	Body mass index	Diabetes pedigree function	Age	Diabetes
1	6	148	72	35	0	33.6	0.627	50	tested_positive
2	1	85	66	29	0	26.6	0.351	31	tested_negative
3	8	183	64	0	0	23.3	0.672	32	tested_positive
4	1	89	66	23	94	28.1	0.167	21	tested_negative
5	0	137	40	35	168	43.1	2.288	33	tested_positive
6	5	116	74	0	0	25.6	0.201	30	tested_negative
7	3	78	50	32	88	31.0	0.248	26	tested_positive
8	10	115	0	0	0	35.3	0.134	29	tested_negative
9	2	197	70	45	543	30.5	0.158	53	tested_positive
10	8	125	96	0	0	0.0	0.232	54	tested_positive
11	4	110	92	0	0	37.6	0.191	30	tested_negative
12	10	168	74	0	0	38.0	0.537	34	tested_positive
13	10	139	80	0	0	27.1	1.441	57	tested_negative
14	1	189	60	23	846	30.1	0.398	59	tested_positive

Fig. 2 Examples of medical datasets

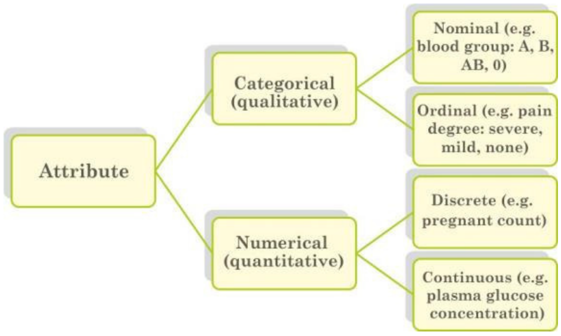

Fig. 3 Types of attributes

Numerical attributes are another class. They are expressed in the form of numbers. These attributes define object properties from quantitative point of view and, therefore, they are also known as quantitative attributes. They can be subdivided into two groups: discrete and continuous. Discrete attributes are often defined on a set of some whole numerical values and very often count numbers of some events. As an example, let us consider the number of times when a woman has been pregnant or the birth year of a patient. When the attribute is continuous, then it means there is no limitation (except the lower and upper limits) on the values that can be taken. Typical examples are patient's height, weight or plasma glucose concentration.

One of the typical problems of KDD that is related to data is how much data is optimal for DM algorithms (Fig. 4).

There exists no definitive answer but, in general, more records are better because we can avoid the problems of underfitting, when the created model is too simple to analyze data that do not come from the original dataset. Similarly, datasets with less attributes are better than ones with many because the discovered knowledge can be interpreted easier.

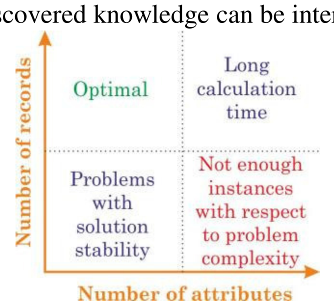

Fig. 4 Problems of DM related to the number of attributes and records

C. *Preparation of the Data*

When we understand the medical problem domain and the data on which KDD will be performed, we can prepare them for DM algorithms. This is one of the most important and the most time consuming steps of KDD [3]. It consists of two phases: data cleansing and data transformation.

The quality of data is the most important factor on which the success of KDD depends. Real databases contain a lot of data. However, these data can include incorrect or missing values. If there are a lot of such values, then the result of a DM algorithm will be a model that is unusable in practice. Therefore, data cleansing is very important step. Its main task is to enhance data reliability. There exist a lot of methods that can be used for this purpose. The simplest ones are based on the assumption that most of the data are correct and, therefore, incorrect data can be handled simply by its removing. More sophisticated methods involve some statistical methods to identify and replace incorrect or missing values and the most complex ones use supervised DM algorithms to predict the correct value of an attribute.

When the data have required quality, then we can prepare them for DM algorithms. This phase involves production of new attributes (when given attributes are not very appropriate for DM) and reduction of attribute count (models created from huge amount of attributes are usually very complicated, which results in problems with interpretation of gained knowledge; also, such models are usually very accurate for data from which they have been produced, but inaccurate for new data and, therefore, their deployment can be very problematic).

New attributes can be produced from one or more existing attributes. Therefore, we distinguish between one-attribute and multi-attribute transformations. Typical examples of one-attribute transformations are: normalization (mapping continuous data to values in interval <0, 1>), percentages (values are related to a specified base value), scores (transformation of ordinal data to discrete ones), etc. Examples of multi-attribute transformations are ratios (quotient of two attributes, e.g. using body mass index instead of weight and height), rates (number of event occurrences divided by time, e.g. number of cigarettes per day), and other linear and nonlinear combinations.

There exist several approaches for reducing the number of attributes. Very often, medical expert knowledge can be sufficient to solve this problem. Another alternative is to use some methods of mathematical statistic such as principal component analysis, kernel principal component analysis, independent component analysis [6], etc.

D. *Data Mining*

At the beginning of this phase, the appropriate DM task has to be chosen (Fig. 5). There exist two different goals, for which DM can be used: description and prediction [7].

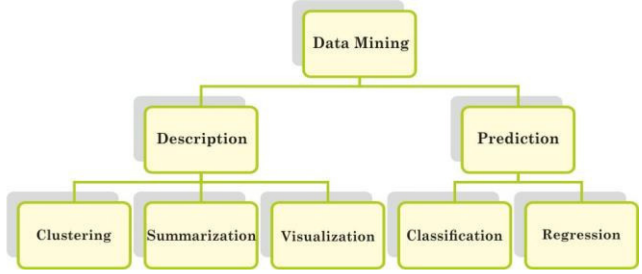

Fig. 5 Basic data mining models

Descriptive methods create models that are used for better understanding of given dataset. Typical examples are clustering, summarization and visualization [3]. The main idea of clustering is to find natural clusters of objects in a dataset. Objects are grouped together if they are similar to one another and dissimilar from objects in other clusters. Summarization is focused on data aggregation that is useful if we want to find some global characteristics of the entire dataset. The global characteristics allow describing data without necessity of knowing exact values of attributes of individual objects in dataset. So, they reduce the dataset size in terms of attributes or records count. Visualization includes techniques that aim is to simplify data understanding.

Predictive methods are used when the attributes can be subdivided into two groups: input and output attributes. In this case, DM can be used to discover the relationship between inputs and output attribute. (For example, the second dataset in Fig. 2 contains 8 input attributes, which more or less relates to diabetes, and 1 output attribute that identifies whether the patient suffer diabetes or not.) Based on the possible values of the output attribute, two types of prediction can be recognized: classification and regression [7]. The former maps the input space into predefined classes or, in general, into a discrete-valued domain, i.e. the output attribute is categorical or discrete numerical (Fig. 3). Typical algorithms of classification include neural networks (existing dataset is used to create and train a neural network that will be used to classify new records), decision trees (existing dataset is used to create a decision tree that will be capable to correct classify new records), instance based learning (every new record is classified according to its similarity with records that have already been classified), etc. Regression models transform the space of input attributes into a real-valued domain, i.e. the output attribute is continuous numerical according to Fig. 3, and the typical examples are linear and non-linear regression.

When we identify the appropriate DM task, then we have to choose and employ DM algorithm that will be used to achieve the goal, e.g. for clustering, we can select from statistical methods, support vector clustering, *k*-means clustering, hierarchical clustering, etc. Every algorithm has some parameters that have to be set correct to get satisfied result, e.g. how many clusters do we want to create, what is the minimal size of a cluster. These parameters are usually obtained by running the algorithm more times with different values of parameters and analyzing the obtained results.

E. Evaluation of the Discovered Knowledge

The result of DM phase is a model that can be used to describe given data or to predict values of some attributes. When we want to interpret the model, then we have to understand the results that are described by it. This means that we have to be able to interpret the results from the medical point of view. This allows us to understand the discovered knowledge and identify whether it is novel and interesting. If the obtained knowledge is novel, then we can check its impact on the medical goal determined at the beginning of the KDD and recognize its usefulness in medical environment.

F. Using the Discovered Knowledge

Finally, when the model is evaluated, then it should be incorporated into another system that will use the knowledge represented by the model. The success of this step determines the effectiveness of the entire KDD because the KDD has been unnecessary without further use of the discovered knowledge. According to [7], there are many challenges in this step, such as losing the "laboratory conditions" under which the model has been created. This means that the model has been produced from a certain static dataset, but the data become dynamic after the model deployment.

The result of KDD is knowledge that can be used for description purposes or for prediction. In case of predictive model that is created from a dataset containing only categorical attributes, the discovered knowledge can be represented as a table that enumerates all combinations of values of input attributes and defines value of the output for each of them. Such kind of table is used also in reliability analysis, and it is known as the structure function. This indicates that some tools of reliability analysis could be used in the analysis of the discovered knowledge.

III. Reliability Analysis

Reliability is an important characteristic of systems. Every system consists of one or more components (basic parts of the system that are assumed to be indivisible into smaller elements). One of the principal tasks of reliability analysis is investigation of influence of individual system components on system activity and identification of components that are most important for system proper work. This investigation is known as importance analysis.

A. Binary- and Multi-State Systems

Before the importance analysis can be performed, a model of the system has to be created. As a rule two types of models are used in reliability analysis. The first one is known as a Binary-State System (BSS). This model is based on the assumption that the system and all its components can be in one of only two possible states – functioning (labelled by number 1) and failure (represented by number 0). The dependency between states of individual system components and system state is expressed by a special relation that is known as structure function. The structure function of a BSS has the following form [8], [9]:

$$\phi(x_1,...,x_n) = \phi(\boldsymbol{x}) : \{0,1\}^n \to \{0,1\}, \tag{1}$$

where n is a number of system components, x_i is a variable denoting state of the i-th component and $\boldsymbol{x} = (x_1,\ldots, x_n)$ is a vector of states of system components (state vector). The structure function of a BSS can be viewed as a Boolean function and, therefore, some approaches related to analysis of Boolean functions can be used in the analysis of BSSs [9].

BSSs have been widely used in reliability analysis, especially in the analysis of systems in which any deviation from perfect functioning results in failure of the system, e.g. nuclear power plants [10], aviation systems [11]. However, these models are not very appropriate for systems that can operate at different performance levels, i.e. systems that can meet their mission also when they are not perfectly functioning, e.g. distribution networks [12] or healthcare systems [13]. Therefore, models that allow defining more than two states in system/components performance are used in the analysis of such systems. These models are known as Multi-State Systems (MSSs).

A general MSS permits defining different number of states for the system and for its components. If we assume that the system has m possible states and its i-th component, for $i = 1,\ldots, n$, can be in one of m_i states, then the structure function of the MSS corresponds to the next map [14]–[16]:

$$\begin{gathered}\phi(x_1,...,x_n) = \phi(\boldsymbol{x}) : \\ \{0,...,m_1-1\} \times \ldots \times \{0,...,m_n-1\} \to \{0,...,m-1\},\end{gathered} \tag{2}$$

where 0 corresponds to completely failure of the system/ component while m -1 (m_i -1) means that the system (the i-th component) is perfectly functioning.

A special type of MSSs is a homogenous system, in which $m_1 = \ldots = m_n = m$. The structure function of such system can be interpreted as a Multiple-Valued Logic (MVL) function. This fact allows us to use some methods of MVL logic in reliability analysis of MSSs [16].

The mathematical model of structure function used in reliability analysis can be combined with methods of DM to perform analysis of medical databases. This analysis can be used to identify database attributes that carry no important information from the point of view of database purpose.

For example, let us consider the example of database for the analysis of the breast cancer diagnosis from [17]. In this example 4 categorical input attributes are used (Table I): A_1 (*Gynecological history*), A_2 (*Tumor*), A_3 (*Heredity*), and A_4 (*Age*). Each combination of values of the input attributes is connected to output attribute B (*Breast Cancer Possibility*). Let the attributes have the next values: $A_1 = \{A_{1,1}, A_{1,2}, A_{1,3}\}$, $A_2 = \{A_{2,1}, A_{2,2}, A_{2,3}\}$, $A_3 = \{A_{3,1}, A_{3,2}\}$, $A_4 = \{A_{4,1}, A_{4,2}\}$, and $B = \{B_1, B_2, B_3\}$ assuming that these values has the meaning presented in Table I.

TABLE I.
ATTRIBUTES VALUES

Attribute	Attribute Values	Description of Attribute Values
A_1	$A_{1,1}$	Gynecological history with high risk
	$A_{1,2}$	Gynecological history with medium risk
	$A_{1,3}$	Gynecological history with low risk
A_2	$A_{2,1}$	Yes and confirmed by medical examination
	$A_{2,2}$	Yes and non-confirmed
	$A_{2,3}$	No
A_3	$A_{3,1}$	Yes
	$A_{3,2}$	No
A_4	$A_{4,1}$	Younger than 40 years
	$A_{4,2}$	40 years or more
B	B_1	High Possibility of Breast Cancer
	B_2	Medium Possibility of Breast Cancer
	B_3	Low Possibility of Breast Cancer

In [17], an association rule for breast cancer diagnosis was inducted based on the methods of DM. This rule includes 4 input attributes and one output attribute (Table II). From reliability point of view, the input attributes can be interpreted as system components and the output attribute is considered to be the system state. This implies that the table representing the discovered knowledge can be interpreted as the structure function of a MSS that depends on 4 variables: the 1-st and 2-nd variable has 3 possible values and the 3-rd and 4-th have two values. This structure function is defined based on all possible values of the input attributes and the output attribute is calculated according to the association rule derived in [17].

In section IV, we will use some approaches of reliability analysis to identify input attributes that have the greatest influence on the value of the output attribute. These results can be used to perform some database optimization since we can find attributes that have very little influence on the output attribute and, therefore, it might not be necessary to store them in the database.

B. *Coherent and Noncoherent Systems*

Based on the properties of the structure function, two different classes of systems can be recognized – coherent and noncoherent. A system is coherent if a failure/degradation of any system component can result only in system failure/ degradation. This means that the structure function of coherent systems is monotonic (non-decreasing) [8], [14].

TABLE II.
STRUCTURE FUNCTION FOR DATABASE OF BREAST CANCER DIAGNOSIS

Variables $x_1\ x_2\ x_3\ x_4$	$\phi(\boldsymbol{x})$	Variables $x_1\ x_2\ x_3\ x_4$	$\phi(\boldsymbol{x})$
0 0 0 0	1	1 1 1 1	2
0 0 0 1	1	1 2 0 0	0
0 0 1 0	1	1 2 0 0	2
0 0 1 1	1	1 2 0 1	2
0 1 0 0	2	1 2 1 0	2
0 1 0 1	0	1 2 1 1	2
0 1 1 0	2	2 0 0 0	2
0 1 1 1	0	2 0 0 1	2
0 2 0 0	2	2 0 1 0	2
0 2 0 1	2	2 0 1 1	2
0 2 1 0	2	2 1 0 0	2
0 2 1 1	2	2 1 0 1	2
1 0 0 0	1	2 1 1 1	2
1 0 0 1	1	2 2 0 0	2
1 0 1 0	1	2 2 0 0	2
1 0 1 1	1	2 2 0 1	2
1 1 0 0	2	2 2 1 0	2
1 1 0 1	0	2 2 1 1	2

A system is noncoherent if its structure function is not monotonic. This implies that a noncoherent system admits situations in which component failure/degradation can cause system repair/improvement [18].

The coherency is a typical property of most systems studied in reliability engineering. This indicates that many tools of reliability analysis are based on the assumption that the structure function is monotonic. However, there also exist some systems whose structure function cannot be monotone, e.g. logic networks [19] or *k*-to-*l*-out-of-*n* systems, which are functioning if at least *k* but not more than *l* components are working [20]. The analysis of such systems is more complicated than the analysis of coherent systems and, therefore, it has to be done more carefully.

C. *Availability*

The structure function defines system topology, and it carries no information about reliability of individual system components. Therefore, if we want to analyze not only topological properties of the system but also some probabilistic characteristics (e.g. system availability or unavailability, mean time to failure), the probabilities of states of individual system components have to be known:

$$p_{i,s} = \Pr\{x_i = s\}, \quad s \in \{0, \ldots, m_i - 1\}, \quad i \in \{1, \ldots, n\}, \quad (3)$$

where $m_i = 2$ for BSSs. Please note that in case of BSSs, $p_{i,0}$ is known as unavailability of the *i*-th system component and $p_{i,1}$ as its availability.

Knowledge of system structure function and probabilities (3) can be used to compute three important characteristics of the system – the probability that the system is in state *j* (for *j* = 0,…, *m* -1), the system availability, which is defined with regard to system state *j* as follows [14], [15]:

$$A^{\geq j} = \Pr\{\phi(\boldsymbol{x}) \geq j\}, \quad j \in \{1,\ldots,m-1\}, \tag{4}$$

and the system unavailability, which is defined with regard to system state j as the probability that the system cannot fulfill a requirement that requires at least level j of system performance [14], [15]:

$$U^{\geq j} = \Pr\{\phi(\boldsymbol{x}) < j\} = 1 - A^{\geq j}, \quad j \in \{1,\ldots,m-1\}. \tag{5}$$

Please note, in case of BSSs ($m = 2$ in definitions (4) and (5)), the system availability is defined as the probability that the system is in state 1 while the unavailability agrees with the probability of system 0-state. Therefore, terms "system availability (unavailability)" and "the probability that the system is in state 1 (0)" can be used as synonyms in case of BSSs. However, this is not true for MSSs, in which these two terms represent two different concepts.

System availability and unavailability are very important in reliability analysis. They can be used to estimate mean time to system failure or mean time to system repair. However, they do not allow identifying components that have the greatest influence on system activity. This is very important task because its results can be used to optimize system reliability or to plan system maintenance.

D. *Importance Analysis*

Importance analysis is a part of reliability engineering. It is used to quantify situations in which a change of component state results in a change of system state. For this purpose, Importance Measures (IMs) are used. There exist a lot of IMs [21]. However, in what follows, we will consider only two of them – the Structural Importance (SI) and the Birnbaum's Importance (BI).

The SI and BI have originally been developed for the analysis of coherent systems. In [22], the SI of component i has been defined as a relative number of situations in which the component is critical for system failure/functioning, i.e. as a proportion of cases when a failure (repair) of the i-th system component results in system failure (repair). In the same paper, the BI has been introduced as the probability that the component failure (repair) causes system failure (repair). These definitions imply that the main difference between the SI and BI is that the former analyzes only the system structure while the latter takes into account not only the structure function but also availabilities (unavailabilities) of system components. Therefore, the SI is primarily used to analyze topological properties of the system.

The considered IMs have been generalized for coherent MSSs in [16], [23]–[25]. These works have introduced several types of the SI and BI depending on whether we are interested in:

a) identification of component states that are the most important for a given system state/ availability [16], [23], [25],
b) identifying component states that have the most influence on the whole system (not only on a specific system state/availability) [24],
c) finding components that are the most important for a given system state/availability,
d) revealing the total importance of individual components for the whole system [25].

Finally, works [18], [26] have introduced definitions of the SI and BI for noncoherent BSSs. These versions of the SI and BI allow quantifying:

a) dependency of system failure (repair) on a failure (repair) of a given component,
b) dependency of system repair (failure) on a failure (repair) of a given component,
c) the total influence of a given component on the system activity.

According to our knowledge, no generalizations of the considered IMs have been proposed for importance analysis of noncoherent MSSs. This can be caused by the fact that these models are used in reliability analysis very rarely.

E. *Logical Differential Calculus*

Logical differential calculus is a tool used to investigate dynamic properties of Boolean and MVL functions [27]. The central term of this tool is a Boolean/MVL derivative. There exist several types of this derivative. For our purposes, the most important is Direct Partial Logic Derivative (DPLD).

A DPLD of a Boolean function $f(\boldsymbol{x})$ with respect to variable x_i can be defined in the following way [27]:

$$\partial f(j \to \bar{j}) / \partial x_i(s \to \bar{s}) = \\ = \begin{cases} 1, \text{ if } f(s_i, \boldsymbol{x}) = j \text{ and } f(\bar{s}_i, \boldsymbol{x}) = \bar{j} \\ 0, \text{ other} \end{cases}, \tag{6}$$

where $f(a_i, \boldsymbol{x}) = f(x_1,\ldots, x_{i-1}, a, x_{i+1},\ldots, x_n)$ for $a \in \{s, \bar{s}\}$ and $s, j \in \{0, 1\}$. According to this definition, the DPLD of a Boolean function reveals situations in which change of Boolean variable x_i from value s to $\bar{s}$ causes that the Boolean function value changes from j to $\bar{j}$.

In the similar way, a DPLD of a MVL function $f_m(\boldsymbol{x})$ with respect to variable x_i is defined as follows [27]:

$$\partial f_m(j \to h) / \partial x_i(s \to r) = \\ = \begin{cases} 1, \text{ if } f_m(s_i, \boldsymbol{x}) = j \text{ and } f_m(r_i, \boldsymbol{x}) = h \\ 0, \text{ other} \end{cases}, \tag{7}$$

where $f_m(a_i, \boldsymbol{x}) = f_m(x_1,\ldots, x_{i-1}, a, x_{i+1},\ldots, x_n)$ for $a \in \{s, r\}$; $s, r, j, h \in \{0,\ldots, m-1\}$, $s \neq r$, and $j \neq h$. Clearly, this derivative models consequence of change of the MVL variable from value s to r on the value of the considered MVL function and, therefore, it can be used to detect situations in which the investigated change of the MVL variable results in the change of the function value from j to h.

Since the structure function of a BSS can be interpreted as a Boolean function and the structure function of a homogenous MSS as a MVL function, DPLDs can also be used in reliability analysis of such systems [9], [16]. Moreover, the next little modification of definition (6) also allows applying them to non-homogenous MSSs:

$$\partial\phi(j \to h)/\partial x_i(s \to r) = \begin{cases} 1, \text{ if } \phi(s_i, \boldsymbol{x}) = j \text{ and } \phi(r_i, \boldsymbol{x}) = h \\ 0, \text{ other} \end{cases}, \tag{8}$$

where $s,\ r \in \{0,\dots, m_i\text{-}1\}$, $s \neq r$ and $j,\ h \in \{0,\dots, m\text{-}1\}$, $j \neq h$. Please note that this definition is the most general definition of a DPLD from which definitions (6) and (7) can be obtained simply using the assumption that $s, r, j, h \in \{0,1\}$ or $s, r, j, h \in \{0,\dots, m\text{-}1\}$ respectively. Therefore, in what follows, we will primarily use this definition.

In terms of reliability analysis, DPLDs are used to identify situations in which a given change of state of the i-th system component results in the investigated change of the system state. These derivatives can be split into four groups:

A. $j > h$ and $s > r$ – these derivatives identify situations in which component failure/degradation results in system failure/degradation,
B. $j < h$ and $s < r$ – these DPLDs detect situations in which component repair/improvement causes a repair/improvement of system activity,
C. $j > h$ and $s < r$ – these derivatives can be used to find coincidence between component repair/ improvement and system failure/degradation,
D. $j < h$ and $s > r$ – these DPLDs investigate situations in which system repair/improvement is caused by failure/degradation of the considered component.

Based on the definition of a coherent system, only DPLDs from groups A and B are relevant in the analysis of such systems. However, this is not true for noncoherent systems, for which the derivatives from all groups can be nonzero.

F. Importance Measures based on Direct Partial Logic Derivatives

The SI and BI are used to quantify coincidence between component state change and change of system state. Based on the previous paragraphs, this coincidence can be identified based on DPLDs. Therefore, these derivatives can also be used to compute the SI and BI [9], [13], [16].

Firstly, let us consider a coherent BSS. The structure function of this system is monotone, therefore, only DPLDs $\partial\phi(1\to 0)/\partial x_i(1\to 0)$ and $\partial\phi(0\to 1)/\partial x_i(0\to 1)$ can be nonzero for this type of systems. The former identifies situations in which a failure of component i results in system failure, and the latter detects state vectors at which a repair of the component causes that the system begins work. Since the SI of component i is defined as a relative number of situations in which a failure (repair) of component i results in system failure (repair), this IM can be computed using DPLDs in the following manner [9]:

$$\begin{aligned}\mathrm{SI}_i &= \mathrm{TD}\big(\partial\phi(1\to 0)/\partial x_i(1\to 0)\big) \\ &= \mathrm{TD}\big(\partial\phi(0\to 1)/\partial x_i(0\to 1)\big),\end{aligned} \tag{9}$$

where TD(.) denotes truth density of the argument interpreted as a function with Boolean-valued output, i.e. a proportion of situations in which the argument takes value 1.

The BI of component i can be calculated using DPLDs in the similar way [9]:

$$\begin{aligned}\mathrm{BI}_i &= \Pr\{\partial\phi(1\to 0)/\partial x_i(1\to 0)=1\} \\ &= \Pr\{\partial\phi(0\to 1)/\partial x_i(0\to 1)=1\}\end{aligned} \tag{10}$$

since it is defined as the probability that a failure (repair) of the component causes system failure (repair).

Secondly, let us consider a noncoherent BSS. In this case all four DPLDs that can be defined are relevant because all of them can contain nonzero elements, i.e. DPLDs $\partial\phi(1\to 0)/\partial x_i(1\to 0)$, $\partial\phi(1\to 0)/\partial x_i(0\to 1)$ can be used to find correlation between system failure and change of state of component i while derivatives $\partial\phi(0\to 1)/\partial x_i(0\to 1)$, $\partial\phi(0\to 1)/\partial x_i(1\to 0)$ identify situations in which a change of component state results in system repair.

It has been proposed in [18] that importance analysis of noncoherent BSSs should be performed in three steps. Firstly, we should quantify influence of component failure on system failure (repair). Secondly, impact of component repair on system failure (repair) should be quantified. Finally, the total influence of the considered component on system failure (repair) can be estimated as the sum of the results obtained in the previous two steps. This implies that several IMs of one type can be defined for one component. For example, in case of the SI, the next four measures can be calculated:

$$\begin{aligned}\mathrm{SI}_{i\downarrow}^{\downarrow} &= \mathrm{TD}\big(\partial\phi(1\to 0)/\partial x_i(1\to 0)\big), \\ \mathrm{SI}_{i\uparrow}^{\uparrow} &= \mathrm{TD}\big(\partial\phi(0\to 1)/\partial x_i(0\to 1)\big), \\ \mathrm{SI}_{i\downarrow}^{\uparrow} &= \mathrm{TD}\big(\partial\phi(0\to 1)/\partial x_i(1\to 0)\big), \\ \mathrm{SI}_{i\uparrow}^{\downarrow} &= \mathrm{TD}\big(\partial\phi(1\to 0)/\partial x_i(0\to 1)\big).\end{aligned} \tag{11}$$

The first two SI measures are used to quantify coincidence between component failure (repair) and system failure (repair). The remaining SI measures estimates topological correlation between component failure (repair) and repair (failure) of the system. The total topological influence of component i on system failure is computed in the following manner [26]:

$$\begin{aligned}\mathrm{SI}_i^{\downarrow} &= \mathrm{SI}_{i\downarrow}^{\downarrow} + \mathrm{SI}_{i\uparrow}^{\downarrow} \\ &= \mathrm{TD}\big(\partial\phi(1\to 0)/\partial x_i(1\to 0)\big) \\ &\quad + \mathrm{TD}\big(\partial\phi(1\to 0)/\partial x_i(0\to 1)\big),\end{aligned} \tag{12}$$

and on system repair in the following way:

$$\begin{aligned}\mathrm{SI}_i^{\uparrow} &= \mathrm{SI}_{i\uparrow}^{\uparrow} + \mathrm{SI}_{i\downarrow}^{\uparrow} \\ &= \mathrm{TD}\big(\partial\phi(0\to 1)/\partial x_i(0\to 1)\big) \\ &\quad + \mathrm{TD}\big(\partial\phi(0\to 1)/\partial x_i(1\to 0)\big).\end{aligned} \tag{13}$$

It is clear that

$$\mathrm{SI}_{i\downarrow}^{\downarrow} = \mathrm{SI}_{i\uparrow}^{\uparrow},\ \mathrm{SI}_{i\uparrow}^{\downarrow} = \mathrm{SI}_{i\downarrow}^{\uparrow},\ \mathrm{SI}_{i}^{\downarrow} = \mathrm{SI}_{i}^{\uparrow} \tag{14}$$

and, therefore, $\mathrm{SI}_{i\downarrow}^{\downarrow}$, $\mathrm{SI}_{i\uparrow}^{\downarrow}$, and $\mathrm{SI}_{i}^{\downarrow}$ can be used not only in the investigation of system failure but also in the analysis of system repair.

Please note that the same results can also be obtained for the BI of noncoherent BSSs by replacement of the truth densities in equations (11) – (13) with the probabilities that the DPLDs take value 1.

Thirdly, let us focus on coherent MSSs. In this case, we can quantify several dependencies between component state and system state. For simplicity, we will consider only situations in which component degradation coincide with system degradation, i.e. we will not introduce the IMs for component improvement. Furthermore, we will assume that system components can degrade only one state. Using these assumptions, we can calculate the SI of state s of component i for system state j as follows [16]:

$$\mathrm{SI}_{i,s\downarrow}^{j\downarrow} = \sum_{h=0}^{j-1} \mathrm{TD}\big(\partial\phi(j \to h)/\partial x_i(s \to s-1)\big), \tag{15}$$

for $i \in \{1,\dots, n\}$, $s \in \{1,\dots, m_i -1\}$, $j \in \{1,\dots, m -1\}$.

Based on the meaning of DPLDs, this IM corresponds to the relative number of situations in which a minor degradation (i.e. degradation by one state) of state s of component i results in degradation of system state j. Using the ideas presented in [24], [25], this SI can also be used to compute the relative number of situations in which a minor degradation of state s of component i results in system degradation:

$$\begin{aligned}\mathrm{SI}_{i,s\downarrow}^{\downarrow} &= \sum_{j=1}^{m-1} \mathrm{SI}_{i,s\downarrow}^{j\downarrow} \\ &= \sum_{j=1}^{m-1}\sum_{h=0}^{j-1} \mathrm{TD}\big(\partial\phi(j \to h)/\partial x_i(s \to s-1)\big),\end{aligned} \tag{16}$$

or the relative number of situations in which a minor degradation of component i causes degradation of system state j:

$$\begin{aligned}\mathrm{SI}_{i\downarrow}^{j\downarrow} &= \frac{1}{m_i-1}\sum_{s=1}^{m_i-1} \mathrm{SI}_{i,s\downarrow}^{j\downarrow} \\ &= \frac{1}{m_i-1}\sum_{s=1}^{m_i-1}\sum_{h=0}^{j-1} \mathrm{TD}\big(\partial\phi(j \to h)/\partial x_i(s \to s-1)\big),\end{aligned} \tag{17}$$

or the proportion of state vectors at which a minor degradation of component i results in system degradation:

$$\mathrm{SI}_{i\downarrow}^{\downarrow} = \frac{1}{m_i-1}\sum_{s=1}^{m_i-1} \mathrm{SI}_{i,s\downarrow}^{\downarrow}. \tag{18}$$

Equation (16) can be used to quantify topological influence of state s of component i on the whole system. Similarly, equations (17) allows us to investigate the total influence of component i on system state j and formula (18) the total influence on the whole system. In the similar way, the BI for a minor degradation of component state for a coherent MSS can be defined. The only difference is that the truth densities in (15) – (17) have to be replaced with the probabilities that the considered DPLDs are nonzero.

IV. Use of Importance Analysis in Investigation of Medical Databases

As we mentioned in section III.A, a complete medical database can be obtained from a medical dataset using some tools of DM. The complete database containing only qualitative (categorical) attributes can be viewed as a MSS whose structure function agrees with the relation (discovered knowledge) between the input attributes and the output attribute. However, the main problem is that the database has to be interpreted as the structure function of a noncoherent MSS because it can contain situations in which a decrease in value of input attribute can result in increase of value of the output attribute. For example, in Table II, change of variable x_2, which corresponds to value of attribute A_2, from value 1 to 0 causes that the value of the structure function of the considered database changes from value 0 to 1 if $x_1 = 0$, $x_3 = 0$, and $x_4 = 1$. This fact requires proposing some generalizations of the SI for noncoherent MSSs if we want to use this measure to find input attributes that have the greatest influence on the output attribute.

In noncoherent systems, not only component degradation but also component improvement can result in system degradation. This implies that we also need to detect situations in which component improvement results in system degradation. DPLDs $\partial\phi(j \to h)/\partial x_i(s \to s+1)$ in which $j > h$ can be used for this purpose. Based on these DPLDs, topological influence of a minor improvement of state s of component i on degradation of system state j can be estimated using the next version of SI:

$$\mathrm{SI}_{i,s\uparrow}^{j\downarrow} = \sum_{h=0}^{j-1} \mathrm{TD}\big(\partial\phi(j \to h)/\partial x_i(s \to s+1)\big), \tag{19}$$

for $i \in \{1,\dots, n\}$, $s \in \{0,\dots, m_i -2\}$, $j \in \{1,\dots, m -1\}$.

In case of noncoherent BSSs, the total influence of a given component on system failure is computed as the sum of SI measures analyzing consequences of the component failure and repair. Therefore, in case of MSSs, the total importance of state s of component i for degradation of system state j can be computed simply using the next SI:

$$\mathrm{SI}_{i,s}^{j\downarrow} = \begin{cases} \mathrm{SI}_{i,s\uparrow}^{j\downarrow} & \text{if } s = 0, \\ \mathrm{SI}_{i,s\downarrow}^{j\downarrow} & \text{if } s = m_i - 1, \\ \mathrm{SI}_{i,s\downarrow}^{j\downarrow} + \mathrm{SI}_{i,s\uparrow}^{j\downarrow} & \text{else}, \end{cases} \tag{20}$$

where $\mathrm{SI}_{i,s\downarrow}^{j\downarrow}$ is computed based on formula (15).

SI measures (15), (19), and (20) are useful for evaluation of influence of a given component state on degradation of a given system state. However, they do not allow identifying importance of the whole component on degradation of a given system state or importance of a given component state on the entire system (regardless of a concrete system state). For these purposes, other versions of the SI have to be defined. This can be done in the similar way as in the case of coherent MSSs, i.e. the total importance of a given component state on system activity can be computed as follows:

$$\mathrm{SI}_{i,s}^{\downarrow} = \sum_{j=1}^{m-1} \mathrm{SI}_{i,s}^{j\downarrow} = \begin{cases} \mathrm{SI}_{i,s\uparrow}^{\downarrow} & \text{if } s = 0, \\ \mathrm{SI}_{i,s\downarrow}^{\downarrow} & \text{if } s = m_i - 1, \\ \mathrm{SI}_{i,s\downarrow}^{\downarrow} + \mathrm{SI}_{i,s\uparrow}^{\downarrow} & \text{else,} \end{cases} \quad (21)$$

where $\mathrm{SI}_{i,s\downarrow}^{\downarrow}$ (definition (16)) quantifies consequences of deterioration of state s of the i-th system component on system activity, and $\mathrm{SI}_{i,s\uparrow}^{\downarrow}$ calculates results of improvement of state s of component i on system degradation. Please note that $\mathrm{SI}_{i,s\uparrow}^{\downarrow}$ is computed similarly as $\mathrm{SI}_{i,s\downarrow}^{\downarrow}$, i.e.:

$$\begin{aligned} \mathrm{SI}_{i,s\uparrow}^{\downarrow} &= \sum_{j=1}^{m-1} \mathrm{SI}_{i,s\uparrow}^{j\downarrow} \\ &= \sum_{j=1}^{m-1} \sum_{h=0}^{j-1} \mathrm{TD}\big(\partial\phi(j \to h)/\partial x_i(s \to s+1)\big). \end{aligned} \quad (22)$$

The total topological influence of component i on a given system state can be calculated based on the next formula:

$$\mathrm{SI}_{i}^{j\downarrow} = \frac{1}{m_i - 1} \sum_{s=0}^{m_i-1} \mathrm{SI}_{i,s}^{j\downarrow} = \mathrm{SI}_{i\downarrow}^{j\downarrow} + \mathrm{SI}_{i\uparrow}^{j\downarrow}, \quad (23)$$

where $\mathrm{SI}_{i\downarrow}^{j\downarrow}$ (definition (17)) quantifies results of degradation of the i-th system component on degradation of system state j, and $\mathrm{SI}_{i\uparrow}^{j\downarrow}$ evaluating consequences of improvement of the i-th component on degradation of system state j is computed using the following formula:

$$\begin{aligned} \mathrm{SI}_{i\uparrow}^{j\downarrow} &= \frac{1}{m_i - 1} \sum_{s=0}^{m_i-2} \mathrm{SI}_{i,s\uparrow}^{j\downarrow} \\ &= \frac{1}{m_i - 1} \sum_{s=0}^{m_i-2} \sum_{h=0}^{j-1} \mathrm{TD}\big(\partial\phi(j \to h)/\partial x_i(s \to s+1)\big). \end{aligned} \quad (24)$$

Based on the meaning of DPLDs, it can be shown simply that SI (21) agrees with the relative number of state vectors at which a minor change (i.e. a change by one state) of state s of component i results in system degradation, SI (22) identifies the relative count of state vectors at which a minor improvement of state s of component i results in system degradation, SI (23) corresponds to the relative number of situations in which a degradation or improvement of component i causes decrease in state j of the system, and SI (24) agrees with the proportion of state vectors at which an improvement of component i causes deterioration of state j of the system

Finally, the total topological importance of a given component on system activity can be defined as the relative number of state vectors at which a change of component state results in system deterioration and, therefore, it can be computed as follows:

$$\mathrm{SI}_{i}^{\downarrow} = \mathrm{SI}_{i\downarrow}^{\downarrow} + \mathrm{SI}_{i\uparrow}^{\downarrow}, \quad (25)$$

where $\mathrm{SI}_{i\downarrow}^{\downarrow}$ (definition (18)) quantifies results of degradation of the i-th system component on system degradation, and $\mathrm{SI}_{i\uparrow}^{\downarrow}$ computed based on the next formula:

$$\mathrm{SI}_{i\uparrow}^{\downarrow} = \frac{1}{m_i - 1} \sum_{s=0}^{m_i-2} \mathrm{SI}_{i,s\uparrow}^{\downarrow} \quad (26)$$

evaluates consequences of improvement of the considered component on system degradation.

In this section, we have proposed a lot of SI measures that can be used in the investigation of topological properties of noncoherent MSSs. For clarity, we summarize them in Table III. The similar formulae could also be proposed for BI measures. The only difference is that the probabilities that the DPLDs are nonzero have to be used in formulae (15) – (26) instead of the truth densities.

TABLE III.
STRUCTURAL IMPORTANCE MEASURES FOR NONCOHERENT MULTI-STATE SYSTEMS

SI	Coherent Part (Influence of Component Degradation)	Noncoherent Part (Influence of Component Improvement)
$\mathrm{SI}_{i,s}^{j\downarrow}$	$\mathrm{SI}_{i,s\downarrow}^{j\downarrow} = \sum_{h=0}^{j-1} \mathrm{TD}\left(\frac{\partial\phi(j\to h)}{\partial x_i(s\to s-1)}\right)$	$\mathrm{SI}_{i,s\uparrow}^{j\downarrow} = \sum_{h=0}^{j-1} \mathrm{TD}\left(\frac{\partial\phi(j\to h)}{\partial x_i(s\to s+1)}\right)$
$\mathrm{SI}_{i,s}^{\downarrow}$	$\mathrm{SI}_{i,s\downarrow}^{\downarrow} = \sum_{j=1}^{m-1} \mathrm{SI}_{i,s\downarrow}^{j\downarrow}$ for $s > 0$	$\mathrm{SI}_{i,s\uparrow}^{\downarrow} = \sum_{j=1}^{m-1} \mathrm{SI}_{i,s\uparrow}^{j\downarrow}$ for $s < m_i - 1$
$\mathrm{SI}_{i}^{j\downarrow}$	$\mathrm{SI}_{i\downarrow}^{j\downarrow} = \frac{1}{m_i-1} \sum_{s=1}^{m_i-1} \mathrm{SI}_{i,s\downarrow}^{j\downarrow}$	$\mathrm{SI}_{i\uparrow}^{j\downarrow} = \frac{1}{m_i-1} \sum_{s=0}^{m_i-2} \mathrm{SI}_{i,s\uparrow}^{j\downarrow}$
$\mathrm{SI}_{i}^{\downarrow}$	$\mathrm{SI}_{i\downarrow}^{\downarrow} = \frac{1}{m_i-1} \sum_{s=1}^{m_i-1} \mathrm{SI}_{i,s\downarrow}^{\downarrow}$	$\mathrm{SI}_{i\uparrow}^{\downarrow} = \frac{1}{m_i-1} \sum_{s=0}^{m_i-2} \mathrm{SI}_{i,s\uparrow}^{\downarrow}$

Based on the relation between medical database and the structure function of a noncoherent MSS, the proposed SI measures can be used to analyze importance of individual input attributes on the value of the output attribute. For illustration, let us consider the medical database defined by Table II. The total topological importance of individual input attributes is computed based on formula (25) in Table IV. Based on the data presented in this table, the input attribute that has the greatest influence on the output attribute is A_2. On the other hand, value of attribute A_3 has no influence on attribute B. This implies that attribute A_3 is not important for tasks for which the table is used (i.e. decision whether the breast cancer has high possibility or not) and, therefore, it is not necessary to store its values in the database.

V. CONCLUSION

This paper focuses on correlation between some key terms of KDD (or DM) and reliability analysis. We illustrated that KDD is a very complex process whose main part is DM. DM is used to discover some new information (knowledge) in a database (predictive DM) or for better understanding of data stored in a database (descriptive DM). In case of predictive DM used on databases containing only categorical attributes, the discovered knowledge can be interpreted as a table that can be viewed as the structure function of a MSS. This allows us to use some methods of reliability analysis in investigation of database properties. One of them is importance analysis, which identifies influence of system components on the system activity.

TABLE IV.
IMPORTANCE OF INDIVIDUAL INPUT ATTRIBUTES FOR THE OUTPUT ATTRIBUTE

Input Attribute	$\mathrm{SI}_i^{\downarrow}$
A_1	0.25
A_2	0.50
A_3	0
A_4	0.22

In this paper, we considered use of importance analysis in investigation of coincidence between change of input attributes and change of the output attribute of a table representing the discovered knowledge. This required extending some measures used in importance analysis on noncoherent MSSs. The extension was done using logical differential calculus. The presented approach can be used to optimize number of attributes occurring in the table. Furthermore, it can also be used to decide which attributes have to be measured with the most accuracy to ensure that the prediction based on the table representing the discovered knowledge will be correct.

REFERENCES

[1] W. J. Frawley, G. Piatetsky-Shapiro and C. J. Matheus, "Knowledge discovery in databases: An overview," in *Knowledge Discovery in Databases*, G. Piatetsky-Shapiro and W. J. Frawley, Eds. Cambridge, MA: AAAI/MIT Press, 1–27, 1991.

[2] V. Sigillito, "Pima Indians Diabetes Database," UCI Machine Learning Repository [http://archive.ics.uci.edu/ml/datasets/Pima Indians Diabetes]. Phoenix, AZ: National Institute of Diabetes and Digestive and Kidney Diseases, 1990.

[3] K. J. Cios and G. W. Moore, "Medical data mining and knowledge discovery: Overview of key issues," in *Medical Data Mining and Knowledge Discovery*, K. J. Cios, Ed. New York, NY: Physica Verlag Heidelberg, 2001, pp. 1–20.

[4] K. J. Cios, A. Teresinska, S. Konieczna, J. Potocka and S. Sharma, "A knowledge discovery approach to diagnosing myocardial perfusion," *IEEE Engineering in Medicine and Biology Magazine*, vol. 19, no. 4, pp. 17–25, Jul.–Aug. 2000.

[5] A. Petrie and C. Sabin, *Medical Statistics at a Glance*, 2nd ed. Oxford, UK: Blackwell Publishing Ltd, 2005.

[6] L. J. Cao, K. S. Chua, W. K. Chong, H. P. Lee and Q. M. Gu, "A comparison of PCA, KPCA and ICA for dimensionality reduction in support vector machine," *Neurocomputing*, vol. 55, no. 1–2, pp. 321–336, Sep. 2003.

[7] O. Maimon and L. Rokach, "Introduction to knowledge discovery in databases," in *Data Mining and Knowledge Discovery Handbook*, O. Maimon and L. Rokach, Eds. New York, NY: Springer Science+Business Media, Inc., 2005, pp. 1–17.

[8] M. Rausand and A. Høyland, *System Reliability Theory: Models, Statistical Methods, and Applications*. Haboken, NJ: John Wiley & Sons, Inc., 2004, 664 p.

[9] E. N. Zaitseva and V. G. Levashenko, "Importance analysis by logical differential calculus," *Automation and Remote Control*, vol. 74, no. 2, pp. 171–182, Feb. 2013, http://dx.doi.org/10.1134/S000511791302001X.

[10] Y. Watanabe, T. Oikawa, and K. Muramatsu, "Development of the DQFM method to consider the effect of correlation of component failures in seismic PSA of nuclear power plant," *Reliability Engineering & System Safety*, vol. 79, no. 3, pp. 265–279, Mar. 2003, http://dx.doi.org/10.1016/S0951-8320(02)00053-4.

[11] B. Nystrom, L. Austrin, N. Ankarback, and E. Nilsson, "Fault tree analysis of an aircraft electric power supply system to electrical actuators," in *Probabilistic Methods Applied to Power Systems, 2006. PMAPS 2006. International Conference on*, 2006, pp. 1–7, http://dx.doi.org/10.1109/PMAPS.2006.360325.

[12] W.-C. Yeh, "A simple approach to search for all d-MCs of a limited-flow network," *Reliability Engineering & System Safety*, vol. 71, no. 1, pp. 15–19, Jan. 2001, http://dx.doi.org/10.1016/S0951-8320(00)00070-3.

[13] E. Zaitseva, V. Levashenko, and M. Rusin, "Reliability analysis of healthcare system," in *2011 Federated Conference on Computer Science and Information Systems, FedCSIS 2011*, 2011, pp. 169–175.

[14] B. Natvig, *Multistate Systems Reliability Theory with Applications*. New York, NY: Wiley, 2011, 262 p., http://dx.doi.org/10.1002/9780470977088.

[15] A. Lisnianski, I. Frenkel and Y. Ding, *Multi-state System Reliability Analysis and Optimization for Engineers and Industrial Managers*. London, UK: Springer-Verlag London Ltd., 2010, 393 p., http://dx.doi.org/10.1007/978-1-84996-320-6.

[16] E. Zaitseva and V. Levashenko, "Multiple-valued logic mathematical approaches for multi-state system reliability analysis," *Journal of Applied Logic*, vol. 11, no. 3, pp. 350–362, Sep. 2013, http://dx.doi.org/10.1016/j.jal.2013.05.005.

[17] V. Levashenko and E. Zaitseva, "Fuzzy decision trees in medical decision making support system" in *2012 Federated Conference on Computer Science and Information Systems, FedCSIS 2012*, 2012, pp. 213–219.

[18] J. D. Andrews and S. Beeson, "Birnbaum's measure of component importance for noncoherent systems," *IEEE Transactions on Reliability*, vol. 52, no. 2, pp. 213–219, Jun. 2003, http://dx.doi.org/10.1109/TR.2003.809656.

[19] E. Zaitseva, M. Kvassay, V. Levashenko, and J. Kostolny, "Reliability analysis of logic network by logical differential calculus," in *2014 ELEKTRO*, 2014, pp. 245–250, http://dx.doi.org/10.1109/ELEKTRO.2014.6848895.

[20] S. J. Upadhyaya and H. Pham, "Analysis of noncoherent systems and an architecture for the computation of the system reliability," *IEEE Transactions on Computers*, vol. 42, no. 4, pp. 484–493, Apr. 1993, http://dx.doi.org/10.1109/12.214699.

[21] W. Kuo and X. Zhu, *Importance Measures in Reliability, Risk, and Optimization*. Chichester, UK: John Wiley & Sons, Ltd, 2012, 472 p., http://dx.doi.org/10.1002/9781118314593.

[22] Z. W. Birnbaum, "On the importance of different components in a multicomponent system," in *Multivariate Analysis*, vol. 2, P. R. Krishnaiah, Ed. New York, NY: Academic Press, 1969, pp. 581–592.

[23] D. A. Butler, "A complete importance ranking for components of binary coherent systems, with extensions to multi-state systems," *Naval Research Logistics Quarterly*, vol. 26, no. 4, pp. 565–578, Dec. 1979, http://dx.doi.org/10.1002/nav.3800260402.

[24] W. S. Griffith, "Multistate reliability models," *Journal of Applied Probability*, vol. 17, no. 3, pp. 735–744, Sep. 1980, http://dx.doi.org/10.2307/3212967.

[25] S. Wu, "Joint importance of multistate systems," *Computers & Industrial Engineering*, vol. 49, no. 1, pp. 63–75, Aug. 2005, http://dx.doi.org/10.1016/j.cie.2005.02.001.

[26] J. Kostolny, M. Kvassay, and S. Kovalik, "Reliability analysis of noncoherent systems by logical differential calculus and binary decision diagrams," *Komunikacie*, vol. 16, no. 1, pp. 114–120, 2014.

[27] S. N. Yanushkevich, D. M. Miller, V. P. Shmerko and R. S. Stankovic, *Decision Diagram Techniques for Micro- and Nanoelectronic Design. Handbook*. Boca Raton, FL: CRC Press, 2006, 952 p.