VOL 1 (2017) NO 3

e-ISSN : 2549-9904
ISSN : 2549-9610

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

A Survey on Data Mining Algorithms and Techniques in Medicine

Kasra Madadipouya[#]

[#] *Department of Computing and Science, Asia Pacific University of Technology & Innovation*
Kuala Lumpur, 57000, Malaysia
E-mail: kasra_mp@live.com

Abstract— **Medical Decision Support Systems (MDSS) industry collects a huge amount of data, which is not properly mined and not put to the optimum use. This data may contain valuable information that awaits extraction. The knowledge may be encapsulated in various patterns and regularities that may be hidden in the data. Such knowledge may prove to be priceless in future medical decision making.**
Available medical decision support systems are based on static data, which may be out of date. Thus, a medical decision support system that can learn the relationships between patient histories, diseases in the population, symptoms, pathology of a disease, family history, and test results, would be useful to physicians and hospitals.
This paper provides an in-depth review of available data mining algorithms and techniques. In addition to that, data mining applications in medicine are discussed as well as techniques for evaluating them and available applications of performance metrics.

Keywords— **Data Mining; Classification; Decision Tree; Neural Network; Bayesian Network Classifier; Evaluation Metrics**

I. INTRODUCTION

Health care institutions all over the world have been gathering medical data over the years of their operation. A huge amount of the data is stored in databases and data warehouses. Such databases and their applications can be quite diverse. Depends on the functionality, the basic ones' store only some information about patients such as name, age, address, blood type, etc. The more advanced ones are able to record patients' visits and detailed information related to their health condition. Some applied to patients' registration, units' finances. Recently new types of a medical system have emerged which originates in the business intelligence and facilitates medical decisions [1], medical decision support system.

The available data may contain valuable information that awaits extraction. The knowledge can be encapsulated in various patterns and regularities that are hidden in the data. Such knowledge proved to be priceless in future medical decision making.

One approach of utilizing the stale medical data is to apply data mining techniques to discover dependencies and nontrivial rules in data that can be highly valuable [2][3]. This can reduce the time of a diagnosis delivery or risk of a medical mistake as well as improve the process of treatment and diagnosing [4].

The purpose of this research is to review the most common data mining techniques implemented in medicine. A number of research papers have evaluated various data mining methods but they focus on a small number of medical datasets [5][6], and the algorithms used are not calibrated (tested only on one parameters' settings) [6] or the algorithms compared are not widely recognized in the medical decision support systems [7]. Even though a large number of methods have been studied [5][8][7] they were not evaluated with the use of different metrics on different datasets. This makes the collective evaluation of the algorithms impractical. In this paper, we review the most common data mining algorithms (determined after an in- depth literature study), which are implemented in modern MDSS's.

The rest of the paper is organized as follows. The next section provides an explanation about data mining and identify the most widely used data mining algorithms in medicine. Then in the section three, a literature review on the well-known data mining algorithms is provided. In section four, data mining application in medicine is discussed. Sections five and six are dedicated to techniques of evaluating data mining algorithms and application of performance metrics in medicine respectively. Finally, in the last section, conclusions and future improvements are discussed

II. BACKGROUND

During an appointment in a healthcare unit, a physician evaluates a patient's condition. Symptoms are the basis for a diagnosis. This information is usually stored in either a medical unit's system or inpatient's files. This data may contain nontrivial dependencies [4], which may turn out to be valuable. There are many methods and algorithms used to mine data for

hidden information including: Artificial Neural Networks (ANN), Decision Trees, Association Rules, Naïve Bayes, Support Vector Machines (SVM), Cauterization, and Logistic Regression.

However, preliminary studies demonstrated that the most frequent choices for the Medical Decision Support Systems are the decisions trees (C4.5 algorithm), Multilayer Perceptron and the Naïve Bayes [9][10][11]. These algorithms are very useful in medicine because they decrease the time spent for processing symptoms, producing diagnoses, and making them more precise at the same time [9][10][11]. Despite their popularity, no scientific paper available that compared the three of them under the same conditions.

Many of the research works [5][6] assessed the algorithms on a narrow set of medical databases (not more than three). Furthermore, the metrics used varied from one paper to another which makes the comparison of the algorithms performance infeasible. This paper aims at filling this gap in the body of knowledge.

In the following section, we review the available data mining algorithms briefly and then in the next section discuss the three aforementioned data mining algorithms in details.

The authors of [12][13] proposed medical rules induction. The article [12] presents a study on unsupervised fuzzy clustering algorithms and rule based systems, which are useful in labeling of tomography images. The presented methods turn out to be computationally efficient for one class of problems. However, in other applications this efficiency seems to be much lower. In some applications, the generated rules are claimed to be easy to construct and modify. Furthermore, their independency allows for changing one rule not affecting the others.

In [14] rules extraction is achieved with the use of a Multilayer Perceptron. The authors proposed a C-MLP2LN algorithm, which generates additional nodes, deletes the connections among them, and optimizes the rules. Such solution leads to simpler and more accurate rules.

The authors of [15] present a study on generation of rules that describe associations among attributes. The experiments conducted on real medical data and their correctness verified with the use of statistical measures and physicians' evaluations. This article presents an analysis of real data from St. Thomas' Hospital in London. It also provides a description of all the steps performed: from pre-processing, through data mining experiments to verification of accuracy of the results.

Another way to classify instances is with the use of an artificial neural network. In [16] authors introduce artificial neural networks with back propagation for classification of heart disease cases. The algorithm implemented in a medical system to support the classification of the Doppler signals in cardiology. The predictions yielded by the method were more accurate than similar presented in [3].

The authors in [17] claim that Multilayer Perceptron is one of the most frequently employed neural network algorithms in modern MDSS's. They discuss applications of this algorithm to classification of different diseases (hepatic, lung and breast cancers).

Another interesting study has been described in [20] where two different neural network techniques are presented. NeuroRule and NeuroLinear used to apply to diagnosis of hepatobiliary disorders. The neural networks' major disadvantage is complexity [20], which makes classification process difficult to interpret. Nevertheless, the authors prove that they produce effective classifications in case of medical data. The medical application of neural networks is also presented in [21][22]. This is the reason why this method may turn out to be helpful in supporting medical diagnoses.

Besides the neural network, also decision trees are utilized in medical knowledge extraction [13][11]. Their main advantage is simplicity and easy-to-comprehend structure of generated models [4]. In [11][13] decision trees classification applied for diagnosis of ovarian and Melanoma skin cancers, respectively. The decision trees prove to be applicable also in other fields of medicine.

The authors of [14] compare the accuracy of the method with a Bayesian network in diagnosis of female urinary incontinence. The obtained classifications results demonstrated improvements to some extends, though, the difference was small.

In [39], a new algorithm has been developed based on C4.5 to perform the process of mining data for medicine applications and the proposed algorithm. The result proves some improvements over C4.5, though in expense of lengthy computational process.

An application of Bayes' law in medical analyses was first proposed in 1959 [9] in an article about theoretical possibilities of applying this solution in physicians' everyday work. This idea implemented in 1972 by an implementation of a medical system to support diagnosing abdominal pain. The system used the Naïve Bayes algorithm. This classifier assumes that all attributes are independent. Throughout many years, scientists in collaboration with medical staff have tried to develop suitable diagnosis system with the use of the Bayesian theorem. Several studies on this problem are presented in [10][2]. The requirement of the attributes to be independent was regarded as a problem. The in-depth analysis of this classification method has shown that the requirement is not essential for correct classifications. Simplicity, learning speed, and classification speed are the main advantages of the Bayesian classifier [23]. On the other hand, one of the most notable drawbacks is the ad-hoc restrictions placed on the graph. This makes the classifications hard to understand [9]. This is the reason why the method has to be implemented in medicine with care as diagnoses have to be thoroughly understandable.

III. Data Mining Algorithms

In data mining, various algorithms can be utilized to analyze the data in order to extract valuable information from the data classes. As discussed in previous section, preliminary studies demonstrated that the most common data mining algorithms in medicine are Decision Trees, Naive Bayesian, and Neural Networks. Thus, only the aforementioned algorithms are taken into consideration for the review. Each is discussed in the following section.

A. Classification

Classification is utilized to categorize data into class labels, which are defined in advanced. In classification "Class", is a characteristic in a dataset that its operators are extremely absorbed. It is defined in a role of dependent variable in numerical facts. To categorize data, a classification algorithm produces a classification model containing classification rules.

In medical applications, classification has taken a great role in diagnosis and prognosis of many health diseases based on health conditions and symptoms [24]. Classification contains of two phases: *training* and *testing* procedure.

In training phase, a classification model is constructed containing categorizing rules, through studying training data including class labels (a model of A classification instruction is "$IF\ Lung\_Cancer\ Family\_History\ =$ $"yes"\ AND\ Smoking\ =\ "yes"\ THEN\ Scan = required"$). A number of classifiers such as Support Vector Machines (SVM) utilize mathematic instead of $IF\ THEN$ rules for better accuracy. It is not essential for classifying rules to be completely exact; normally, if the rules accuracy is between 90 and 95% they are labeled as strong rules. The accuracy of a classifier is dependent on the degree that its rules are precise.

In testing phase, the accuracy of the testing data in classifying unknown object for prediction is studies. The testing procedure is inexpensive in terms of computation in comparison with the training phase that is complicated and requires significant computational sources [24].

B. Decision Tree

Decision trees are an effective method of decision making with great classification accuracy that only present gathered information which been utilized widely in medical decision making.

A decision tree contains attribute nodes connected to a number of sub trees and leaves or decision nodes considered as a class that indicates the decision. Every test node contains a number of outcomes according to the values of attribute and every probable outcome is connected to one of the sub trees. Classification of an example begins at the node of the tree. In the case of attribute node, the result for the instance is determined and the procedure goes on using the suitable sub tree until a leaf finally achieved, its label means the expected class [18].

Decision Tree dataset should be separated into training and testing sets likes other any machine learning algorithms. Training set is utilized for a decision tree construction and the test set is utilized to ensure the accuracy of the discovered answer. Initially, all attributes defining each case are described (input data) and among all, an attribute is selected as a decision for the produced problem (output data). Particular value classes are described for every input attribute. If an attribute is able to take just discrete values, each value takes its own class; if an attribute is able to take various numerical values then some characteristic intervals should be determined, that denotes decision classes. Each attribute is able to be an internal node in a generated decision tree named a test node or an attribute node. The number of subdivisions for an attribute node like this is the number of different value classes. The leaves of a decision tree represent decisions and present the value classes of the decision attribute – decision classes. To make decision for an unresolved problem, researchers begin with basic nodes of the decision tree, move along attribute nodes and select branches which attributes values in unsolved case and the decision tree match until a leaf node is reached representing the decision.

Following subsections discuss about the various available decision tree techniques.

1) Induction of Decision Trees: Classical Approach

In 1986, Quinllan proposed a method to produce classification directions in outline of decision tree called Iterative Dichotomiser 3 (ID3) [28]. ID3 developed in 1993 with an enhanced algorithm C4.5 known as a fundamental model to construct a decision tree based on the traditional statistical method [28]. To construct an ID3 or C4.5 decision tree statistic computation of information gain is used for one attribute. In this method, an attribute which is most informative about decision in a training set is selected primary, and the rest of the nodes are chosen in this manner from the other attributes [18].

In traditional decision trees induction method, the most significant characteristic is the method of splitting data set, i.e. how can we choose an attribute test that controls the spreading of training items into subsets upon which sub trees constructed consequently. For evaluating splits C4.5 induction method utilizes information theory. Two dividing standards are [18]:

- Gain criterion
- Gain ratio criterion

The gain criterion is described in the following is according to [18]. In equation (1), X is the population and S is a subdivision of $X, freq\ (j_i, S)$ is the number of objects that belong to class i. Then assume the 'message' which a random chosen object and belongs to class j_i. $\frac{freq\ (j_i,S)}{|S|}$ is the possibility of the 'message', where $|S|$ denotes the entire number of objects in subdivision S. The information that a message carries (in bits) is offered by [24].

$$-\log_2(\frac{freq\ (j_i,S)}{|S|}) \qquad (1)$$

The expected information carried by the message (in bits) is provided by summing over the classes [24]:

$$info(S) = -\log_2(\frac{freq\ (C_i,S)}{|S|}) \qquad (2)$$

When $info(T)$ is used to a series of training object provides the average number of data to recognize the object of a class in T. This quantity is named entropy of the series T as well.

Consider a similar measurement once T is divided with respect to the n results of a test X. The expected information is able to be found as a weighted sum over the subdivisions $\{Ti\}$ [18]:

$$info_x(T) = \sum_{i=1}^{n} \frac{|T_i|}{|T|}\ .\ info(T_i) \qquad (3)$$

The amount

$$gain(X) = info(T) - info_x(T) \qquad (4)$$

Calculates the information which is achieved through separating T with respect the test X. The gain criterion chooses a test that maximizes the information gain. The gain criterion has an important drawback, which tends to test data with lots of outcomes. The gain ratio by Quinllan expanded to evade the bias. The data produced through separating T into n subdivisions is provided via [18].

$$split\ info(X) = \pm\sum_{i=1}^{n} \frac{|T_i|}{|T|}.\log_2(\frac{|T_i|}{|T|}) \qquad (5)$$

The proportion of information produced through the splitting that is helpful for classification is [18].

$$Improvement\ ratio(X) = \frac{gain(X)}{split\ info(X)} \quad (6)$$

Splitting information is going to be minor and the ratio is not going to be stable. Therefore, the gain ratio chooses a test in order to increase the gain ratio field to the constraint that the information gain is large. We can compare this with impurity approach of CART, where impurity is a measure of the class mix of a subdivision and splits are selected so that reduction in impurity is maximized. This approach led to the development of the GINI index. Impurity method reflects the possibility of misclassification of a new sample from the total population and the example does not belong to the training sample, T.

2) Oblique Partitioning of Search Space

Earlier stated algorithms are using univariate partitioning methods that users are interested to utilize them because their implementation is easy and the built decision tree is understandable. Besides univariate partitioning techniques there are a number of methods for partitioning called *oblique partitioning* as well. These techniques utilize combination of attributes as an alternative of dividing the search area axis-parallel that is based on one attribute at a time [25].

Oblique partitioning is a suitable alternate for univariate approaches. Unlike univariate methods, oblique partitioning are based on combination of attributes. The overall outline of an oblique partition is provided through below expression

$$\sum_{i=1}^{d} \beta_i x_i \ \leq C \quad (7)$$

Where β_i means the coefficient of the $i-th$ attribute. Since oblique partitioning have multivariate characteristics, they are more adaptable in separating the search area; this flexibility leads to greater complication, though. The below expression calculates number of oblique partitions if $n > d$, given a data set including n items defined with d attributes

$$2.\sum_{i=1}^{d}\binom{n-1}{i} \quad (8)$$

Every partition is a hyper-plane that splits the exploration area into two non-overlapping halves. The amount of possible divisions for univariate divisions is much lower, but it is still important, $n \cdot d$. briefly, it's hard to discover a precise oblique partition [25]. Based on the search area size, selecting the precise search technique is extremely crucial in discovering suitable divisions. The key reference for this issue is Breiman on Classification And Regression Trees (CART) [38]. CART utilizes the similar fundamental method as Quinllan in C4.5. At the decision node level, though, the algorithm turns out to be more complicated. CART begins with the greatest univariate division. Then it repetitively explores for perturbations in attribute values (one element at a time) which maximizes a number of good metrics. Next, it compares the greatest oblique and axis-parallel divisions gained and the best one is chosen [26]. Though CART is an effective technique for testing difficulties, it contains a number of inconveniences for the reason that the algorithm gets no tool to escape from local optima. Thus, CART trees are likely to finish its division search too early at a specified node. The main disadvantage of CART and similarly traditional decision trees methods is that the procedure of decision trees induction is able to produce the metrics, which create confusing outcomes. Since traditional decision trees induction is based on locally optima solutions for every decision node, they predictably disregard divisions, which score poorly alone, but produce better results at the time they are utilized in combination [18]. The problem is demonstrated in Fig.1. The solid lines present the divisions discovered through CART. Though every splitting improves the impurity metric, the outcome essentially is not the greatest probable divisions (presented in the dotted lines). However, the dotted curves show great impurity and therefore are not selected. Based on this, it is obvious decision trees have a sequential nature which can avoid the construction of trees which represent the natural structure of the data [25].

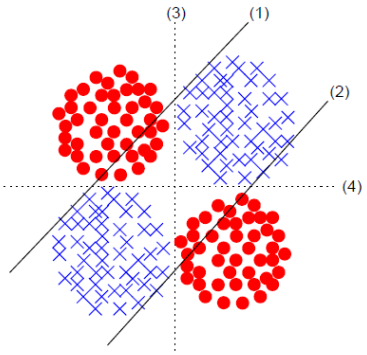

Fig.1 CART generated splits (solid lines – 1 and 2)

3) Decision Trees Pruning

Decision tree goal is to divide a training sample T into subdivisions that contain just a specific class. However, training sets might not be representative of the population they are planned to present. In almost every situation, constructing a decision tree that every single leaf has single-class data leads to *overfitting*. Because of it, the decision trees are planned to categorize the training sample instead of the total population and accuracy of the general population will be much less than the accuracy of the training sample.

For solving this issue in almost all of decision tree inductions (C4.5, Classification and Regression Trees, ID3) pruning technique should be used. In this technique, trees are built to greatest size until every leaf has data for a single class or none of the tests improves the mix of classes at the mentioned leaf. Then the tree should be pruned to evade overfitting. In C4.5, pruning is applicable when the predicted rate of error reduces via replacing a subdivision of a leaf. To prune the tree, CART utilizes a proportion of the training data set. The training process is done on the rest of the training sample and then the tree is pruned until the pruning sample accuracy cannot be more enhanced [26].

4) Drawbacks of Classical Induction

Decision trees have presented to be effective in decision making in different domains. The accuracy and efficiency of decision trees are notable and the best advantage of them is concurrent offer of a decision and the uncomplicated and intuitive clarification of how the decision made [27]. However, the traditional induction of decision trees has a number of problems.

The most obvious weakness of traditional decision tree induction algorithms is the weak capability in handling incomplete or noisy data. In the case that a number of attribute values is lost, traditional algorithms cannot process that item in a good way. For instance, in Quinllan's ID3 method, the records which contain missing values have not been inserted to

the training sets – this problem obviously led to decrease excellence of gained solutions (by this method the size of training set and consecutively the information about the studied field have been decreased). C4.5 presented a method to disable this difficulty; however, it is not efficient yet. In actual situations, specifically in medicine, missing data is extremely common. Therefore, efficient managing of data like this is the essential significance [27].

Another significant shortage of traditional induction techniques is their abilities to produce just a single decision when the same training set is utilized. However, in actual situations will be helpful if more than only one decision tree exists and users are able to select the most appropriate decision tree for their problem. As some training objects contain missing values, the test object may contain missing value as well – there could be a new situation where several data are lost and it is not probable to gain it (inaccessibility of a number of medical tools, for instance, or insensitivity of a particular examination for a sick person). In such situations, another decision tree can be selected which does not contain a particular attribute test to make decision [25].

5) Alternatives of Classical Techniques

Decision trees induction is a complicated procedure. So, deterministic induction method is not optimal with respect to the characteristic of obtained decision trees and their weaknesses. In recent years, several researchers have presented many distinctive replacement approaches to the induction of decision trees with the idea of resolving the difficulties. Many of them are originated from methods named soft techniques, like neural networks or evolutionary methods; sometime some algorithms are merged in a hybrid algorithm. A great amount of methods has been generated which improve only a part of the decision tree induction procedure. These methods contain evolutionary algorithms to improve splitting task in attribute nodes, dynamic subdivision selection of training sets, dynamic attributes discretization, etc. [25].

C. Bayesian Network Classifier

Bayesian Networks are an adaptable method to incorporate several kinds of data into a particular probabilistic model. In medicinal applications, the networks are utilized to produce a patient model which incorporates laboratory outcomes, vital signals, clinician observation and other types of medical data.

Stochastic techniques usually are suitable for problems that users have to assign an instance of a set of feature variables to a value of the class variable. These techniques normally found the conditional probability distribution of the class variable given the instance; they choose the class for the instance utilizing a decision rule.

Bayesian network classifiers create models by estimating the probability distribution of the class variables. The networks show a joint probability distribution throughout involved variables. For classification, a set of variables is sorted into a set of feature variables, the class variable, and probably a number of hidden, or middle, variables [28]. Complexity of this algorithm is different. A number of them are common models with no restrictions on the dependencies of the variables and a number of them are extremely easy models with limited dependency constructions [29].

Two famous uncomplicated Bayesian network classifiers are the Tree Augmented Network classifier and naive Bayesian classifier [30]. These approaches have an unfilled set of hidden variables. Furthermore, the Naive Bayesian classifier frequently supposes the feature variables are independent from the class variable; the Tree Augmented Network classifier, allows a tree-like construction to show dependence throughout its feature variables. Nowadays, Naive Bayesian classifiers are being used in several application domains and despite their simplicity their performance is really good [29].

Naïve Bayesian network classifier relies on the following equation (9) to assign a class to an instance.

$$Pr(X,Y) = p(Y).\prod_i p(X_i \mid Y) \qquad (9)$$

The class variable Y is a binary variable, with a positive class value represented by y and a negative class value represented by $\bar{y}$; y' is utilized to point to both classes. The feature variables are represented by X; x is utilized to show a particular instance of the set. Naïve Bayesian network classifier obviously creates the joint possibility spreading $Pr(X,Y)$ over its variables in terms of performance $p(X_i|Y)$ identified for its feature variables $X_i \in X$, and $p(Y)$ identified for the class variable Y. The next parameterization is caused by its independence assumptions [28].

Bayesian network classifiers are regularly created automatically from a dataset. They contain a measure to find dependences among the variables to improve the quality of the model. For instance, measures are a model's accuracy and its minimum description length (MDL). The excellence of a model is only achieved in cases the model is specified totally, or in other words, it contains estimates for all of the involved numeric parameters. The parameters are assessed as a frequency counts that assist to maximize the log-likelihood of the model with respect to the data. The quality measure that is considered as an optimization criterion to develop the model is also used to compare various methods of classification [29].

In learning procedure of Bayesian network classifiers; the quality of the model is enhanced by containing just the most relevant feature variables not only suitable dependencies. Datasets usually have further variables than the amount desired for the classification mission and the relatively unneeded variables might direct to an unwanted bias [29]. The procedure of feature selection cautiously chooses variables from the dataset which assist to enhance the model's quality the most.

Feature variables are attached to a mainly unfilled model until its quality no more enhances with respect to the data [29].

In Bayesian network classifiers, Bayes' instruction is utilized for calculating the posterior possibility distribution $Pr(Y \mid X)$ of the class variable that is going to be utilized for the actual classification [29]:

$$Pr(Y \mid X) = \frac{Pr(Y,X)}{Pr(X)} = \frac{Pr(X \mid Y)\,.\,Pr(Y)}{\sum_{y'} Pr(X \mid y')\,.\,Pr(y')} \qquad (10)$$

The usually utilized decision ruling for binary class variable is the winner-takes-all ruling that allocates an instance to the class that posterior possibility surpasses the maximum possibility of 0.5 [29]. If the performance of the constructed model is evaluated against the same dataset as the model is learned, the performance is going to be overestimated as a result of overfitting the model to the data. To estimate the performance of model for unseen data and to correct the effect of overfitting often ten-fold cross validation is recommended [29].

D. Neural Network

Artificial Neural Network (ANN) contains lots of Processing Elements (PEs) named *neurons* and weighted connections between the Processing Elements. Each Processing Elements has an uncomplicated calculation, like computing sum of its input connection weights, and calculating an output sign which is guided to more Processing Elements. The training process of an Artificial Neural Network consists of allocating weights (real numbers) of interconnections between the Processing Elements, so that the output will be created [8].

The ANN is a strong method which is widely used in data mining applications. ANN is a closely interconnected network containing a series of processing units; ANNs have a full construction presenting a number of characteristics of the biological neural networks. This strong construction is a chance for users to apply parallel concept at every single level.

Other significant characteristic of Artificial Neural Network is error tolerance. Artificial Neural Networks are suitable for situations that data is noisy and uncertain. Artificial Neural Network are a data processing methodology which varies radically from traditional methodologies in which it uses training by instances to solve the problem instead of a fixed algorithm.

Artificial Neural Networks are able to be classified into two kinds according to the training technique: Supervised training and unsupervised training. Networks which are supervised need the actual result for every input while unsupervised networks do not need the output for every input. A main characteristic of Artificial Neural Networks is the repetitive learning procedure in the data presented to the network one at a time, and the weights related with the input values are adjusted every time [8]. Then, after all cases are presented, the procedure begins over again. In learning procedure, the network learns through allocating the weights to predict the true class label of input samples. When a network has been organized for a particular application, the network is prepared to be trained. The initial weights are based on random values and afterwards the training starts.

The most well-known ANN algorithm is back-propagation algorithm. Though there are many kinds of ANN which are utilized for classification purposes [14], this study emphasizes on the feed-forward multilayer networks or multilayer perceptron that are extensively utilized ANN classifiers. The feed-forward, back-propagation algorithm was suggested in the early of 1970's. This back-propagation algorithm is the most easy-to-learn and effective model for complex and multi-layered networks. Its extreme power is in non-linear solutions to imprecise problems. A normal back-propagation network contains of an input level, an output level, and a hidden level (at least one). There is no theoretic restriction on the amount of hidden layer however, normally there are only one or two. A number of researches have been carried out which present maximum of five levels are needed to solve problems with any complexity (one input level, three hidden levels and an output level). Every layer is completely linked to the next layer. Training inputs are joined to the input layer of the network, and desired outputs are likened at the output level. Throughout the learning procedure, a forward flow is created across the network, and the output of every element is calculated level by level. The difference between the output of the last level and the desired output is back-propagated to the earlier levels, usually modified by the derivative of the transfer function, and the weights are usually adjusted. This procedure progresses for the earlier levels until the input level is achieved [31]. The benefits of ANN for classification are:

- Neural networks are more robust than decision trees because of the weights.
- ANNs enhances its performance by learning. This might last after the training set has been used.
- Neural Networks are more robust than decision trees in noisy environment.
- There is a little error rate and thus a high degree of accuracy when the suitable training has been done.

IV. Data Mining Algorithms evaluation techniques

As mentioned previously data mining algorithms are applied to extract valuable knowledge from data. Their function might differ; some are effective in areas that rest of them are not. Thus, it is vital to evaluate their performance in medical field. Medical diagnosing is a serious mission and any mistake in diagnosis may lead dangerous consequences or even death. Hence, in medicine the correct and incorrect diagnosis rate should to be analyzed. It is vital to understand which part of cases was classified correctly.

There are different approaches for evaluating performance of data mining algorithms. This section describes these approaches.

To know the nature of a data mining techniques it is essential to consider data mining as a procedure to examine data, learning solutions, and finally assessing them. One of the main issues in evaluation of the performance of methods is sample data. In cases that the sample data contains all probable combinations of values of attributes there is not any need to mine the data because we can make a table and find the answer (decision) from it. Though, this will never occur. That is the reason why classifying of medicinal data to associated classes (analysis) is not usually certain (prediction). A technique of assessing data mining algorithm is based on splitting the sample data into two subdivisions: training data and testing data. The researchers are still proposing approaches which decrease the possibility of over-fitting of the training data, concurrently trying not to let the data for under-fitting. The under-fitting means not utilizing the entire training set potential. This section defines methods of assessing performance of data mining techniques with respect to medicinal data.

A. Estimation of Hypothesis Accuracy

It is necessary to assess the performance of studied hypothesis as precisely as possible for evaluating accuracy of data mining approaches. The purpose is only to know whether to use the hypothesis or not. For example, when learning from a dataset with limited size is critical to know the learned hypothesis is accurate. Another purpose is that assessing hypothesis is an integral part of majority of learning methods, for example, in post-pruning decision trees to evade overfitting, researchers have to assess the effects of pruning steps on the accuracy of the decision tree. Hence it is very important to know the probable error in appraising the accuracy of the pruned and unpruned tree [18]. Estimating accuracy is very important if there be large amount of data.

However, in real conditions, researchers study a hypothesis and estimate its accuracy on a limited amount of data. In these

cases, two main issues rise: First is bias in estimating the difference between the expected and true value of the hypothesis. Second is estimation variance which measures statistical dispersion represents how far typically the values of random variables from the expected value are. The precision of the classification of future instances is dependent on the accuracy of the hypothesis. True and sample errors are presented to achieve this. Description of the estimation procedure in this chapter is based on [18]. To estimate the hypothesis, need to consider the space of probable instances X. Assume D is an unknown probability distribution which explains the probability of entering each instance in X. The learning mission is to learn the target function f with respect to a space H of probable hypotheses.

In training phase, each instance independently provides training examples of the target function f for the learning phase, according to the distribution D, and then forwards the instance x along with its correct target value $f(\boldsymbol{x})$ to the learner. The medicinal analysis procedure is defined by the target function that classifies every item (patient with the symptoms) to indicate whether the patient endures a sickness or not.

B. True Error and Sample Error

Two kinds of errors rates exist: *sample* and *true* error. The primary is true error that affects prediction in using the hypothesis for future instances. Though, estimation of the true error is difficult. This is the reason of its evaluation by sample error. Sample and true errors are described below which consider the error rate of the total unknown distribution D. In sample error, S is a subset of sample instances of X.

Definition 1 [18]— Assume $errors\ (h)$ is the sample error of hypothesis h with respect to the sample dataset S and target function f, then

$$error_s(h) = \frac{1}{n} \sum_{x \in s} \delta\left(f(x), h(x)\right) \qquad (11)$$

Where n is the number of samples in S, and the amount

$$\delta\ (f(x), h(x)) = \begin{cases} 1, tf\ f(x) \neq h(x) \\ 0,\ otherwise \end{cases} \qquad (12)$$

Definition 2 [18]— Let $errorD\ (h)$ be the true error of hypothesis h with the respect to the distribution D and target function f. The probability that h misclassifies an instance drawn at random according to D is equal to:

$$error_D(h) = Pr_{x \in D}\ [f(x) \neq h(x)] \qquad (13)$$

Where Pr is the probability over the instance distribution D.

$$S\ \in D \qquad (14)$$

What usually is crucial to know is the true error, $error_D(h)$ of the hypothesis, because this is the error that can be expected when applying the hypothesis to future examples. What we *Definition 3* [18]— An $N\%$ confidence interval for some parameter p is an interval that is expected with the probability $N\%$ contains p.

With the usage of statistical theory, it is possible to assert that $error_D(h)$ with the probability $N\%$ lies in the confidence interval presented in (15)

$$error_s(h) \pm Z_N \sqrt{\frac{error_s(h)(1-error_s(h))}{n}} \qquad (15)$$

The percentages of confidence level N and corresponding with them values Z_N area presented in the Table 1.

TABLE I
VALUES OF CONFIDENCE INTERVALS WITH THE PROBABILITY [18]

Confidence level $N\%$	**Constant** Z_N
50	0.67
68	1
80	1.28
90	1.64
95	1.96
98	2.33
99	2.58

Estimate operates fine when following dependency is satisfied [18]:

$$n\ error_s(h)\big(1 - error_s(h)\big) \geq 5 \qquad (16)$$

C. Difference in Error of Two Hypotheses

Consider one wants to estimate the difference between two hypotheses $h1$ and $h2$ for the same target function. These hypotheses have been tested on the training sets $S1$ and $s2$ (containing $n1$ a randomly drown and $n2$ a randomly drown examples respectively) from the same distribution. The following equation defines the difference d between true errors of these two hypotheses

$$d \equiv error_D(h_1) - error_D(h_2) \qquad (17)$$

The parameter d has to be estimated by the sample error. In this case, the clear choice for estimating d is the difference between the sample errors and is represented by $\hat{d}$

$$\hat{d} \equiv error_{S_1}(h_1) - error_{S_2}(h_2) \qquad (18)$$

Mitchell in [18] proposed the below formula which approximates $N\%$ a confidence interval estimate for d

$$\hat{d} \pm Z_N \sqrt{\frac{error_{S_1}(h_1)(1-error_{S_1})}{n_1} + \frac{error_{S_2}(h_2)(1-error_{S_2})}{n_2}} \qquad (19)$$

The value Z_N is a constant variable given in Table 1. It is possible to redefine $\hat{d}$ as equation (19).

In that equation $h1$ *and* $h2$ are tested on a single sample, the training set S is independent of $h1$, $h2$ and $S1$ and $S2$ are set to S.

D. Learning Algorithms Comparison

To compare two data mining methods the numerical method is used. The true error of the algorithms is the basis of the comparison. To compare learning algorithms the sample error of each method is utilized. Also, comparison of algorithms is possible based on their cost. Both classification and learning costs should be considered.

E. Difference in Algorithms Errors

The performance of two learning algorithms L_A and L_B is compared by estimating the expected value from difference between their errors with respect to the target function f, size of the training n and instance distribution D [24]. The following expression estimates the expected value of difference between true errors of two algorithms (20).

$$E_{S\subset D}\left[error_D\left(L_A(S)\right) - error_D\left(L_B(S)\right)\right] \quad (20)$$

Where S is a sample training data and is a subset of D, L is the learning method and $L_A(S)$ is an output hypothesis. The expected value of difference in errors of two algorithms that is a basis to compare algorithms is described by the expression (18). As mentioned in previous sections the true error estimation is only possible by estimating sample error. This is the reason of measuring and estimating the difference between $error(L_A(S))$ and $error_D(L_B(S))$ in limited sample data D_0. In these cases, sample data set D_0 is split into separated test sets T_0 and expression (21) estimates expression (20).

$$error_{T_0}\left(L_A(S_0)\right) - error_{T_0}\left(L_B(S_0)\right) \quad (21)$$

To improve the estimation provided the sample data D_0 is split into separated sets repeatedly. It is possible to estimate the quantity of the procedure of estimating (20) presented in (22) may be structured in (23). The quantity $\bar{\delta}$ obtained by the below procedure can be considered as an estimate of the desired quantity from Equation (20). In other words, we can consider $\bar{\delta}$ as an estimate of the quantity

$$E_{S\subset D_0}\left[error_D\left(L_A(S)\right) - error_D\left(L_B(S)\right)\right] \quad (22)$$

S is a sample set with random size $\frac{k-1}{k} \mid D_o \mid$ drawn from D_0. The below process estimates of the difference in error of two learning methods L_A and L_B [18]:

Stepa1 Divide available data D_0 into k disjoint subsets $T_1, T_2, \ldots, T_k$ of equal size where this size is at list 30.

Stepa2 for $i\epsilon(1,k)$ do step 2.1- step 2.5

Stepa2.1 use T_i for the test set, and the remaining data for training set S_i

Stepa2.2 $S_i \leftarrow \{D_0 - T_i\}$

Stepa2.3 $h_A \leftarrow L_A\ (S_i)$

Stepa2.4 $h_B \leftarrow L_B(h_B)$

Stepa2.5 $\delta_i \leftarrow error_{T_i}(h_A) - error_{T_i}(h_B)$

Stepa3 return the value $\bar{\delta}$ where

$$\bar{\delta} \equiv \frac{1}{k}\sum_{i=1}^{k}\delta_i \quad (22)$$

The confidence interval for estimating by using $\bar{\delta}$ is equal to the formula (23).

$$\frac{1}{k}\sum_{i=1}^{k}\delta_i \pm t_{N,K-1}\sqrt{\frac{1}{k(k-1)}\sum_{i=1}^{k}\left(\delta_i - \bar{\delta}\right)^2} \quad (23)$$

$t_{N,k-1}$ is a constant variable.

	Confidence level N%			
V	90	95	98	99
2	2,92	4,30	6,96	9,92
5	2,02	2,57	3,36	4,03
10	1,81	2,23	2,76	3,17
20	1,72	2,09	2,53	2,84
30	1,70	2,04	2,46	2,75
120	1,66	1,98	2,36	2,62
∞	1,64	1,96	2,33	2,58

Fig 2. The values of constant t_(N,v) as v →∞,t_(N,v) is very close to ZN

F. Counting the Costs

The accuracy of the experiment should be considered in order to test a method's quality. The accuracy of the experiment indicates whether the test set is categorized by the method accurately. Actual classification created by physicians and the classification outcomes are compared with each other. The experiment accuracy is ratio of True Positive (TP) test set to all test examples. *Accuracy* metric is commonly used in machine learning and pattern recognition communities, but it cannot be applied in medical cases because it hides essential details– it does not consider False Negative (FN) cases. This is why in medical domain we must apply other metric of training set accuracy [32].

$$accuracy = \frac{TP}{total}\ 100\% \quad (24)$$

The formulation (24) is improper in medicinal domain. In medicinal data only two answers are taken into consideration (yes/no), hence other accuracy metrics should be applied [32].

Sensitivity calculates the capability of an experiment to be positive when the condition is really present, or how many of the positive test samples are identified. In other words, the sensitivity shows how frequently the thing which is searched is the thing that was looked for. The sensitivity is shown via the formulation (25) [32].

$$sensitivity = \frac{TP}{hypothesis\ positive}\ 100\% = \frac{TP}{TP+FN}\ 100\% \quad (25)$$

Specificity calculates the capability of an experiment to be negative when, the condition is not really existing, or how many of the negative experiment samples are rejected (26) [32].

$$specificity = \frac{TP}{hypothesis\ negative}\ 100\% = \frac{TP}{TP+FN}\ 100\% \quad (26)$$

The last metric for accuracy is named *predictive accuracy* which displays the ratio of properly categorized instances to all instances in the set. Higher predictive accuracy causes the better condition. This metric is presented by the below equation (27) [32].

$$predictive\ accuracy = \frac{TP+TN}{total}100\% = \frac{TP+TN}{TP+TN+FP+FN}100\% \quad (27)$$

The formulations (25-27) are appropriate in many-class prediction accuracy measure as well.

G. Receiver Operating Characteristic Curves

Receiver Operating Characteristics (ROC) charts are an extremely helpful tool for imagining and appraising performance of data mining methods and are usually utilized in medicinal decision making. The thought of ROC originated from signal discovery theory which was established for the analysis of radar images. A ROC curve presents the adjustment between the True Positive rate or sensitivity and the False Positive rate of the given model [32]. To plot a ROC curve for a particular classification model, M, the model should have ability to return the probability or ranking for the expected class of each test tuple. Therefore, researchers require to rate the test data in a reducing sequence, where classifier believes is most probable to belong to the positive or 'yes' class emerges at the highest part of the list. Naive Bayesian and back propagation classifiers are suitable, while rest of them, like decision trees are able to be changed simply in order to return a class probability distribution for prediction. The vertical axis of a Receiver operating characteristics curve represents the TP rate and horizontal axis represents the FP rate [32]. A ROC diagram is shown in Fig.3. The nearer the ROC curve to the top left denotes the better the performance a classifier has.

To measure the accuracy, researchers should estimate the area below the curve. Some soft wares can do this kind of computation. If the area be close to 0.5, the model is less accurate. A model with perfect accuracy will have an area of 1.0 [32].

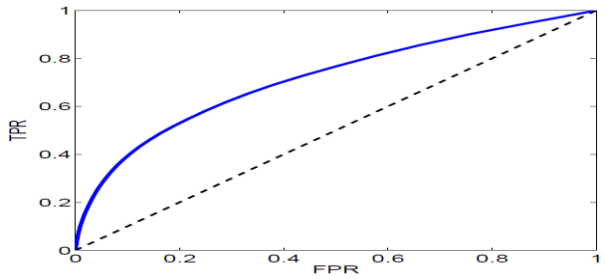

Fig.3 Sample ROC diagram (triangles without usage of the model, squares with the usage of the model)

Finally, this curve lets us to choose the best model on the basis of the expected class distribution for prediction.

H. Precision, Recall, and the F-measure

In medicinal domain, binary classification is a typical sort of problem which exist. There are four possible results of classification (TP, TF, FP, FN). For information recovery programs, usually a large amount of data exists. A classifier is able to achieve a great accuracy by just expressing all of the data as negative. To avoid this *recall*, *precision* and *F-measure* are introduced.

$$precision = TP/(TP + FP),$$
$$recall = TP/(TP + FN), \qquad (28)$$
$$F\ measure = \frac{2}{1/precision + 1/recall}$$

The measures are presented particularly in information recovery applications. In medicine, more significant are *sensitivity* and *specificity*. Specificity is opposite of sensitivity while sensitivity is similar to recall [4].

V. Application Of Performance Metrics In Medicine

Nowadays scientists devote much time and effort to empirical studies which aim at determining performance of data mining solutions. Some methods may yield better results for one type of problems while others may be suitable for different ones. That is why it is important to find pros and cons of each of them. This may help to avoid making mistakes resulting from application of an unsuitable algorithm. The systems which implement data mining solutions may be usable in miscellaneous areas of life, such as: banking, medicine or telecommunication, to name just a few. Such systems are expected to support decision making in a very reliable way. Any mistake may cause irreversible consequences or even lead to someone's death (as it may be the case in medical systems).

As mentioned before while estimating the performance of a method one can come across different problems like: limited sample of data, difficulty in evaluating hypothesis's performance for unseen instances, and finally how to use an available dataset for both training and testing. It is important to realize that there are two issues that need to be considered when estimating performance of an algorithm: *bias* and *variance* of an estimate [24]. The statistical comparison of the methods is based on a sample error [24]. The true error $error_D(h)$ is estimated with the use of the sample $error_S(h)$, where h is the hypothesis, D is a probability distribution and S is the sample dataset. The accuracy of the estimation is often represented by means of confidence intervals [24]. To compare various data mining solutions different notions from statistics and sampling theory are utilized [24]. The most popular include: the probability distributions, expected values, variances and one- or two-sided intervals. The other important measure of method's performance is variance of a random variable that is based on an expected value.

The problem of statistical estimation of algorithm's performance is frequently brought up in professional literature. The authors of [33] discuss the difficulties that accompany comparative classification studies. They attempt to find a solution of how to choose the best machine learning method to reduce the bias while classifying different types of cancer. The statistical comparisons of various classifiers of multiclass data are conducted. The authors of [6] and [34] used k-fold cross-validation [6] and repeated random sampling [6]. [18] and [24] claim that it is important to consider confidence intervals especially comparing small datasets like for instance microarray or other biological data. It is also mentioned differentiation of data processing and sampling strategy may cause discrepancy in understanding of classifications. It is difficult to objectively assess results obtained in different studies. Results from various pre-processing techniques, sampling strategies or learning methods that are applied prior to the actual analyses. This can make a comparison difficult. Finally, yet importantly, inadequate testing strategy also leads to false conclusions about selected methods [24].

For assessing learning method's performance, various strategies are selected [33]: leave one out cross-validation (LOOCV), k-fold cross-validation [6], repeated random subsampling (repeated hold-out method) and bootstrapping [34]. In [34], the authors of reckon that k-fold cross-validation in small-sample datasets (less than 100) is very useful. Furthermore, in the authors' opinion the derived intervals may be too narrow if they are based on a textbook formula that has

not got continuity correlation. Their advice is to balance class distribution and to carefully consider performance measures.

In [35], the authors utilize statistical tests to measure performance of a decision tree. The chosen method is k-fold cross-validation. Two types of tests were conducted: 10-fold cross validation and 5x2-cross-validation to compare different trees creation techniques: boosting, random forests, randomized trees, and bagging.

While comparing solutions is crucial to consider also cost of misclassifications. Making a correct decision is very important, thus the cost should be calculated [32]. One way to show the errors of classification is to introduce a confusion matrix [32]. Such a matrix, for Boolean problems, consist of four fields (numbers): True Positives (TP), True Negatives (TN), False Positive (FP) and False Negative (FN). They all show the dependencies between the actual classes of instances and those delivered by a model. In other words, these numbers show the distribution of classification with respect to each of the classes. Based on these values the overall success rate can be calculated. This method may be improved by introducing Kappa statistic that is a measure of agreement between predicted and observed classifications. However, it neglects the cost. It is necessary to compute cost-sensitive classifications [4]. It may happen (and usually does) that the cost of a FP and FN differs from each other. The medical diagnosis serves as a good example of such a situation. Wrongly treating a healthy patient as a sick one (FP) has completely different consequences than trivializing the symptoms and taking a sick patient as a healthy one (FN). The other approach to the cost of the classification is to consider cost-sensitive learning [4]. Here the cost is taken into consideration during the training process, on the contrary to the cost-sensitive classification. Besides the classification matrix there are other techniques of evaluation of performance of data mining methods. Various analyses may be presented with the use of lift charts [4] which are often applied for instance in marketing [29].

Additionally, the comparison of different machine learning solutions may also be done with the use of the ROC (Receiver Operating Characteristic) curves that are a graphical method for evaluating classifiers. Based on the ROC curves and lifts charts it is possible to introduce two parameters: *recall* and *precision*. They are commonly used in information retrieval. The recall is understood as a number of retrieved relevant documents to the total number of relevant documents. Precision is defined similarly however; the total number of documents that are retrieved divides the number of documents retrieved that are relevant. The author of [36] applies the ROC curves to evaluation of performance of a data mining model. The model was used to predict the cases of the corpus luteum deficiency in women with recurrent miscarriage. The classification tended to yield a significant number of FP and FN diagnoses in the experiment. ROC curves turned out to be valuable in comparing two or more data mining methods.

In [37], authors describe the ROC curves as a metric that measures the method's performance in a more generic way than the error rate. The authors proved that it is possible to obtain very little bias even for small sample estimates. The AUC (Area Under the Curve) has been proven to be a good evaluator of the methods' performance.

VI. Conclusion

In this paper, various mining data approaches which are utilized in Medicinal Decision Support Systems are discussed. These algorithms are extremely helpful in medication since they are able to increase confidence in the processing signs, diagnosis, and handling them more detailed.

In addition, in a medicinal domain, a person is able to utilize networks to generate a sick person model which incorporates laboratory analysis outcomes, clinician observations, vital signs, and other forms of sick person information.

Furthermore, utilizing decision trees, the decision making progress itself is able to be simply authenticated via a specialist.

Lastly, different evaluation metrics for data mining algorithms in medicine have been discussed as different metrics would be appropriate for different problems and each of them has particular characteristics that emphasize on different aspects of the evaluated algorithms. As a result, the selection of appropriate evaluation metrics in medicinal domain might be cumbersome to some extends as each discussed algorithm functions differently and has different usage in medicine data mining.

Acknowledgment

We would like to thank anonymous readers for their great feedback to improve this work.

References

[1] E. Nolte, and C. M. McKee, Measuring the health of nations: updating an earlier analysis. *Health affairs*, *27*(1), 58-71, 2008.

[2] R. Teach and E. Shortliffe, "An analysis of physician attitudes regarding computer-based clinical consultation systems," *Computers and Biomedical Research*, vol. 14, 542-558, 1981.

[3] I. Turkoglu, A. Arslan and E. Ikay, "An expert system for diagnosis of the heart valve diseases," *Expert Systems with Applications*, vol. 23, no.3, 229–236, 2002.

[4] I. H. Witten, and E. Frank, "Data Mining, Practical Machine Learning Tools and Techniques," *Elsevier*, 2005.

[5] P. Herron, "Machine Learning for Medical Decision Support: Evaluating Diagnostic Performance of Machine Learning Classification Algorithms," *INLS 110, Data Mining*, 2004.

[6] L Li, et al., "Data mining techniques for cancer detection using serum proteomic profiling," *Artificial Intelligence in Medicine*, vol. 32, 71-83, 2004.

[7] E. Comak, A. Arslan and I. Turkoglu, "A decision support system based on support vector machines for diagnosis of the heart valve diseases," *Elsevier*, vol. 37, 21-27, 2007.

[8] R. Rojas, "Neural Networks: a systematic introduction," *Springer-Verlag*, 1996.

[9] A. J. Van gerven, R. Jurgelenaite, B. G. Taal, T. Heskes and P. J. F. Lucas, "Predicting carcinoid heart disease with the noisy-threshold classifier," *Artificial Intelligence in Medicine*, vol. 40, 45-55, 2007.

[10] D. Spiegelhalter and R. Knill-Jones, "Statistical and knowledge based approaches to clinical decision support systems, with an application in gastroenterology," *Journal of the Royal Statistical Society*, vol. 147, 35-77, 1984.

[11] A. Vlahou, J. O. Schorge, B. W. Gregory and R. L. Coleman, "Diagnosis of ovarian cancer using decision tree classification of mass spectral data," *Journal of Biomedicine and Biotechnology*, vol. 5 308-314, 2003.

[12] D. Cosic and S. Loncaric, "Rule-based labeling of CT head image. Lecture Notes in Artificial Intelligence," Berlin, Germany, *Springer-Verlag*, vol. 1211, 453–456, 1999.

[13] W. Duch, K. Grabczewski, R. Adamczak, K. Grudzinski and Z. S. Hippe, "Rules for melanoma skin cancer diagnosis," Available from: http://www.phys.uni.torun.pl/publications/kmk/ [Accessed 2 May 2016], 2001.

[14] M. Hunt, B. Von Konsky, S. Venkatesh and P. Petros, "Bayesian networks and decision trees in the diagnosis of female urinary incontinence," *Engineering in Medicine and Biology Society, Proceedings of the 22nd Annual International Conference of the IEEE*, vol. 1, 551-554, 2000.

[15] G. Richards, V.J. Rayward-Smith, P. H. Sönksen, S. Carey and C. Weng, "Data mining for indicators of early mortality in a database of clinical records," *Artificial Intelligence in Medicine*, vol. 22, no. 3, 215–231, 2000.

[16] W. Detmer, G. Barnett, W. Hersh and M. Weaver, "Integrating Decision Support," *Literature Searching and Web Exploration using the UMLS, Metathesaurus*, 1997.

[17] D. West and V. West, "Model selection for a medical diagnostic decision support system: a breast cancer detection case," *Artificial Intelligence in Medicine*, vol. 20, 183-204, 2000.

[18] T. M. Mitchell, "Machine Learning," *McGraw-Hill Higher Education*, 1997.

[19] L. Autio, M. Juhola and J. Laurikkala, "On the neural network classification of medical data and an endeavor to balance non-uniform data sets with artificial data extension," *Computers in Biology and Medicine*, vol. 37, no. 3, 388-397, 2007.

[20] Y. Hayashi, R. Setiono and K. Yoshida, "A comparison between two neural network rule extraction techniques for the diagnosis of hepatobiliary disorders," *Artificial Intelligence in Medicine*, vol. 20, no. 3, 205–216, 2000.

[21] P. Cunningham, J. Carney and S. Jacob, "Stability problems with artificial neural networks and the ensemble solution," *Artificial Intelligence in Medicine*, vol. 20, no. 3, 217–225, 2000.

[22] A. Sharkey, N. E. Sharkey and S. S. Cross, "Adapting an ensemble approach for the diagnosis of breast cancer," *Proceedings of ICANN*, Skövde, Sweden, 281–286, 1998.

[23] P. Domingos and M. Pazzani, "On the Optimality of the Simple Bayesian Classifier under Zero-One Loss," *Machine Learning*, vol. 29, no. 2-3, 103-130, 1997.

[24] T. Karthikeyan, and P. Thangaraju, "Analysis of Classification Algorithms Applied to Hepatitis Patients," *International Journal of Computer Applications*, 62(15), 2013.

[25] V. Podgorelec, P. Kokol, B. Stiglic and I. Rozman, "Decision trees: an overview and their use in medicine," *Journal of Medical Systems*, 26(5):445-463, 2002.

[26] J. Han, and M. Kamber, "Data Mining: Concepts and Techniques," *Morgan Kaufmann Publishers*, 2nd ed, 2006.

[27] S. K. Murthy, "Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey," *Data Mining and Knowledge Discovery* , 1997

[28] J. Han, "Data Mining: Concepts and Techniques," *Morgan Kaufmann publications*, 2006.

[29] C. Van der gaag, and S. Renooij, "Aligning Bayesian Network Classifiers with Medical Contexts," *Technical Report UU-CS-2008-015*, 2008.

[30] K. Anil Jain, J. Mao and K.M. Mohiuddi, "Artificial Neural Networks: A Tutorial," *IEEE Computers*, pp.31-44, 1996.

[31] S. Haykin, "Neural Networks – A Comprehensive Foundation," *Pearson Education*, 2001.

[32] K. Cios and G. Moore, "Uniqueness of Medical Data Mining," *Artificial Intelligence in Medicine*, 2002, vol. 26, 1-24, 2002.

[33] D. Berrar, I. Bradbury and W. Dubitzky, "Avoiding model selection bias in small-sample genomic datasets," *Oxford University Press*, 2006.

[34] U. Scherf, "A gene expression database for the molecular pharmacology of cancer," *Nature Genetics*, vol. 24, no. 236-245, 2000.

[35] R. E. Banfield, L.O. Hall, K.W. Bowyer and W.P. Kegelmeyer, "A Comparison of Decision Tree Ensemble Creation Techniques," *IEEE Computer Society*, vol. 29, 2007.

[36] S. Daya, "Diagnostic test - receiver operating characteristic (ROC) curve," *Evidence-based Obstetrics and Gynaecology*, vol. 8, no. 1-2, 3-4, 2006.

[37] W. A. Yousef, R.F. Wagner and M.H. Loew, "Estimating the uncertainty in the estimated mean area under the ROC curve of a classifier," *Pattern Recognition Letters*, vol. 26, no. 16, 2600-2610, 2005.

[38] Breiman L, Friedman JH, Olshen RA, Stone CJ. "*Classification and regression trees"*. Wadsworth & Brooks. Monterey, CA. 1984.

[39] Kasra Madadipouya "A New Decision tree method for Data mining in Medicine" *Advanced Computational Intelligence: An International Journal (ACII)*, Vol.2, No.3, July 2015.