A REAL TIME DIABETIC DISEASE PREDICTION USING PATIENTS DATASET THROUGH KNN, SVM AND BPSO ALGORITHMS

Y. Sri Lalitha
Associate Professor, Gokaraju Rangaraju Institute of Engineering and Technology, Email: srilalitham.y@gmail.com

ABSTRACT

The evolution of information technology and the standardization of terminologies in the health area has generated large repositories of data that can be mined to ***enable*** discovery of knowledge to assist in the early identification of sick patients as well as cause and effect relationships. Among the diseases, Diabetes Mellitus (DM) stands out due to the increase in the number of cases. This, the sooner it is discovered, the better and more economical its treatment becomes. Thus, finding a standard in the use of health plans would be useful for discovering and classifying patients affected by the disease, enabling treatments to be carried out in a shorter time, with improvement in the patient's quality of life and cost reduction. The technique applied in this study is the application of algorithms that generate Binary Particle Swarm Optimization (BPSO), Support Vector Machine (SVM), K-Nearest Neighbor (KNN) and Naïve Bayes classifiers, through Data Mining, whose objective is the classification and selection of these patients for inclusion in preventive medicine programs.

Keywords: BPSO, Diabetes Mellitus, KNN, Naïve Bayes, SVM,

1. INTRODUCTION

Mastery of the ability to collect, store, and analyze data, which used to be a competitive differential between companies, is now an indispensable item. Also, regulatory agencies have obliged companies to generate and maintain this data, such as electronic invoices. This enormous amount of available data makes any analysis that is carried out solely by human capacity impossible, without the help of specific computational tools for that purpose. To solve this problem, private and academic entities have promoted a lot of research then software development that allows the analysis of this large volume of data. Another important point to be considered is the continuous increase in health-related costs. Population aging, physical inactivity, obesity are some of the main factors that have caused the increase in cases of Chronic or Non-Communicable Diseases (NCDs). These are multi factorial infections that create all through life and are durable. Among the NCDs, diabetes mellitus stands out due to the high growth in the number of cases [1].

Diabetes mellitus remains defined by way of a metabolic syndrome caused through the lack or insufficiency of insulin - a hormone produced in the pancreas responsible for the transformation of glucose into energy - causing an increase in blood sugar (glucose). This happens because the pancreas becomes unable to produce the hormone in sufficient quantity to supply the body's demand, or else. After all, insulin resistance occurs in the individual. The most common and known kinds of the disease are: type 1 diabetes mellitus (DM1) - Popularly known by way of insulin-dependent diabetes, in this type, the pancreas is unable to produce insulin, at the expense of an immune problem, which causes antibodies to attack the cells that produce the hormone, which is called autoimmune destruction [2]. Type 2 diabetes mellitus (DM2) - In this type of disease, too known by way of non-insulin-dependent diabetes, the problem is in the way the body metabolizes glucose. Unlike type 1 diabetes, in this case, the individual produces insulin, but he ends up having a resistance to the effects of the hormone, or else its production is unsatisfactory to be able to maintain the normal glucose level. This type of disease corresponds to 90% of diabetes cases, usually occurring in obese people and over 40 years old. However, this situation is changing, being already diagnosed in young people, due to poor diet, and combined with a sedentary lifestyle and urban stress [3].

It is important to highlight that DM1 cannot be prevented, but DM2 can be prevented. However, it is estimated that half of the people with diabetes mellitus are unaware of their condition. In developing countries, this estimate reaches 80% [4].

Late treatment of diabetes mellitus causes great damage to the patient's health and also results in high financial costs for the health sector.

One of the options for more efficient treatments and consequent reduction of these costs is in preventive medicine, which consists of programs that aim to prevent or treat illnesses at the beginning, which considerably reduces costs in addition to bringing patients an improvement in the quality of life.

In this context, the health operator has prevention and health promotion programs. Among them, the health in companies' program and the follow-up program for diabetic patients stand out.

One of the great challenges for the health operator is to find patients who are in the early stage of DM2 or who are carriers of the disease but who are not undergoing proper treatment. This becomes a difficult task because, although the diagnosis of DM2 is carried out utilizing simple tests, the operator does not have access to the results of these tests [5].

Considering these scenarios with a large amount of data, an increase in the number of cases of NCDs and an increasing increase in costs, the objective of this work is to mine the database to find patterns in the use of health plans that enable the identification of patients with indicative of DM2 and thus assist the work of preventive medicine in the selection of patients for participation in the diabetic patient monitoring program [6].

In this situation of automation and search for better results, analytical processes appear that assist in decision making, such as the data mining process. Data mining can remain defined by way of the "automatic or semi-automatic process of exploring large databases analytically". The data mining process pursues towards discover patterns and new data as of a given set of data. This process will be presented throughout this essay.

Related to the fields of study of artificial intelligence and data mining, there is another concept that is often confused with data mining itself and which is called machine learning. In data mining, techniques are used to discover properties of an existing data set and possible correlations of different attributes of that set, and machine learning algorithms can remain utilized to build models that perform predictions or classifications of available data [7].

A. Knowledge Discovery And Data Mining

A large amount of data currently available requires that appropriate techniques be used to discover useful knowledge in that data. The set of phases for this purpose is called Knowledge Discovery in

Databases (Information Discovery in Databases - KDD). Fayyad (1996) defines KDD by way of "a non-trivial procedure of recognizing new valid, valuable & understandable standards" [8].

The KDD process involves the following iterative sequences [8]:

1. Data cleaning: removal of noise and irrelevant data;
2. Data integration: if there are multiple data sources, this step aims at the combination;
3. Data selection: recovery of relevant data for the analysis of the database;
4. Data transformation: transformation or consolidation of data into formats suitable for mining;
5. Data mining: application of intelligent algorithms and / or methods to extract data patterns;
6. Knowledge evaluation and representation: the use of techniques for visualization and representation in order to present the knowledge obtained.

B. Related Works

The early identification of diseases is an essential factor for the success of its control, cure, or treatment. In the case of Diabetes Mellitus, the complications of the disease could be minimized.

Since the objectives of the KDD process are to find hidden patterns and also demonstrate cause and effect relationships, health data mining has benefited from these characteristics. The work of Vianna et al. (2010) [9], Data Mining and characteristics of infant mortality, seeks to find rules related to the causes that have led newborn children to death. In relation to data mining for diabetes mellitus, there are several articles whose focus of the study is to find means for the early detection of the disease. Ahamad et al. [10] present a program that aims to analyze data from patients with diabetes mellitus and/or hypertension. The data mining demonstrated that the rate of overweight or obesity among the patients in the study was very close to the percentages obtained in other national surveys.

In search of generating solutions that serve the national health system as support for decision-making to mitigate and prevent diseases, the specific action program arises, which aims to promote and encourage the generation of relevant information and knowledge regarding technologies for health.

Several techniques have been used to make a prediction in the diagnosis of diabetes in patients [11], [12], [13], [14], [15], [16] and [17]. Obtaining a prediction of the medical diagnosis of DM in patients allows early attention to disease control, in addition to reducing diagnostic times and representing economic savings for the health system and the patient. In this work, it was proposed to use three well-known classifiers (SVM, KNN, and Naïve Bayes) to analyze a set of data based only on measurable variables of patients to carry out a prediction.

Section two describes the proposed methodology and section three represents the simulation results followed by the conclusive remarks in section four.

2. PROPOSED METHODOLOGY

Figure 1 explains the flow of automatic detection of diabetes in the Pima Indian database. Where feature sets are optimally selected by binary particle swarm optimization. According to selected features further data is trained and tested with various classifiers.

a. PIMA Database

The tests of the proposed method are carried out on the basis of Pima Indians Diabetes data [18].

The eight clinical descriptors are:

1. Npreg: number of pregnancies.
2. Glu: concentration of plasma glucose.
3. BP: diastolic blood pressure, (mmHg).

4. SKIN: triceps skin fold thickness, (mm).
5. Insulin: insulin dose, (mu U / ml).
6. BMI: body mass index, (weight in kg / (height in m 2).
7. DPF: Diabetes pedigree function (heredity).
8. Age: age (Year).

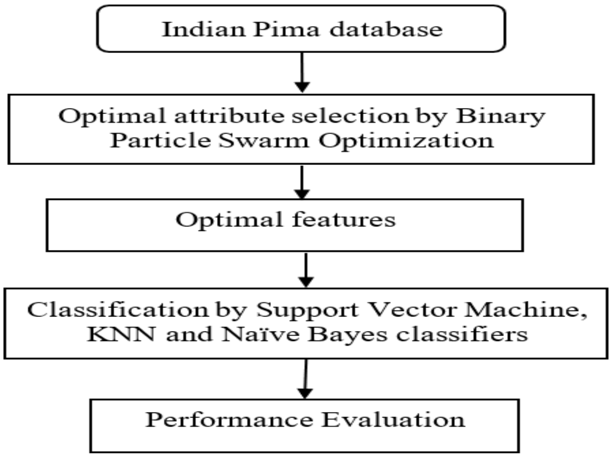

Indian Pima database
Optimal attribute selection by Binary Particle Swarm Optimization
Optimal features
Classification by Support Vector Machine, KNN and Naïve Bayes classifiers
Performance Evaluation

Figure 1: Flow of the automatic diabetic detection using data mining techniques

b. Attribute Selection by BPSO

The Particle Swarm Optimization (PSO) optimization algorithm, proposed by Kennedy and Eberhart [19], is a populace stochastic improvement method that can be contrasted with the conduct of a herd of fowls or to the sociological conduct of a group of people.

It is a population-based evolutionary optimization technique that evolves according to iterations. The population remains called a swarm. Consider an unrestricted minimization problem of a function f. Each particle signifies a probable solution to the optimization problematic. During each iteration, each particle accelerates to its best individual position, as well as towards the best global position discovered by some of the particles in the cluster. This means that if a particle finds a promising new solution, all the other particles in the cluster will move in that direction, exploring this most promising region.

If s denotes the size of the cluster, each individual $1 \leq i \leq s$ has the following attributes:

- Current position for the i^{th} individual in the search space $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$.
- Current speed $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$
- The best position of a particle defined as p_i
- The best position obtained from the population as g_i

During each iteration, the optimization method looks for the ideal arrangement by refreshing the speed and position vectors of every molecule as per the accompanying equations:

$$v_{id}^{t+1} = w * v_{id}^{t} + c_1 * r_{1i}(p_{id} - x_{id}^{t}) + c_2 * r_{2i}(g_{id} - x_{id}^{t}) \quad (1)$$

$$x_{id}^{t+1} = x_{id}^{t} + v_{id}^{t+1} \quad (2)$$

Where t denotes the t^{th} iteration, $d \in D$ dimension of the search space, c_1 and c_2 denote the acceleration coefficients, w is called the weight of inertia where r_{1i} and r_{2i} is a sequence of random numbers uniformly in the range $(0, 1)$.
The value of each component for the vector v_i, can be limited to the interval $[-v_{max}, v_{max}]$ to reduce the probability of the particles leaving the search space. The weight of inertia is typically a linear variation from 0.4 to 0.9 during the generations. The acceleration coefficients c_1 and c_2, which control how far a particle will move in a single iteration, both coefficients have a value of 2.0 [20].

The particle swarm optimization algorithm was originally proposed to solve problems with real variables. However, many optimization problems, such as the problem of detecting spatial clusters, occur in a discrete search space. For this reason, [21] presents a binary approach to the method. Equation (2) is still applied to update the speed, where x_{id}, p_{id} and g_d are restricted to 1 or 0. The speed in the binary approach indicates the probability that the element of the corresponding position assume value 1. A sigmoid function $s(v_{id})$ is introduced to trans form v_{id} into the interval $(0, 1)$. The binary particle swarm optimization algorithm updates the position of each particle according to the following formulas:

$$x_{id} = \begin{cases} 1, & \text{rand}() < s(v_{id}) \\ 0 & \text{Otherwise} \end{cases} \quad (3)$$

Where $(v_{id}) = \frac{1}{1+\exp^{-v_{id}}}$, and $\text{rand}()$ is a uniform random number generator in the interval $(0, 1)$.

c. Classification

Support Vector Machines (SVM)

The SVM linear algorithm is capable of performing various tasks within the context of data mining. In classifying two classes, the algorithm determines a hyperplane that isolates the two classes of information with as wide an edge as could reasonably be expected. This prompts a decent speculation of exactness in imperceptible information and supports particular improvement approaches by adjusting the algorithm parameters.

Multi-class classification is present in many real-world problems, initially support vector machines were designed to deal with binary (+/ –1) problems. Now we will see how to deal with this problem. The objective function is expressed by
:

$$w_r \in H, \in^r \in R^m, b_r \in R \frac{1}{2}\sum_{r=1}^{M} \|w_r\|^2 + \frac{c}{m}\sum_{i=1}^{m}\sum_{r \neq y_1} \varepsilon_i^r \ (4)$$

Subject to:

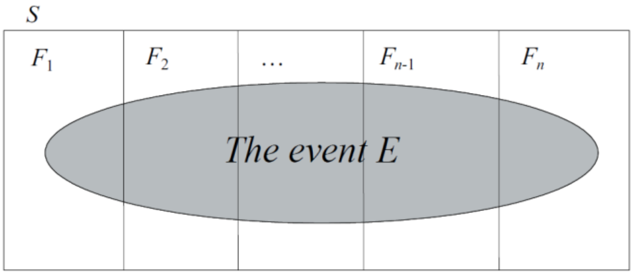
$$\langle W_{y_i}, X_i \rangle + b_{y_i} \geq \langle W_r, X_i \rangle + b_r + 2 - \varepsilon_i^r, \quad \varepsilon_i^r \geq 0 \qquad (5)$$

Where, $m \in \{1, \dots, M\} \backslash Y_i$ and $Y_i \in [1, \dots, M]$ is the multi-class label of the X_i pattern.

As far as accuracy, the outcomes got with this methodology are equivalent to those acquired legitimately utilizing the one against the rest strategy. For down to earth issues, the decision of approach will rely upon the accessible constraints, significant components incorporate the exactness required, the time accessible for improvement, preparing time, and the idea of the grouping issue.

Naïve Bayes Classifier

It is a method based on probability theory; it uses frequencies to calculate conditional probabilities to calculate predictions about new cases. Naïve Bayes is both a predictive and a descriptive technique. Despite being simple, it has been successfully developed, producing good results in its applications.



$$E = EF_1 \cup EF_2 \cup \ldots \cup EF_{n-1} \cup EF_n$$

Figure 2: Event E occurs in conjunction with one of the mutually exclusive events F_j

Let E and F be events. We can express E as:

$$E = EF \cup EF^c \qquad (6)$$

That is, for an event E to occur, E and F must occur, or E must occur and F not.

Because EF and EF^c are mutually exclusive, then we have:

$$\begin{aligned} P(E) &= P(EF) + P(EF^c) \\ &= P(E|F)P(F) + P(E|F^c)P(F^c) \\ &= P(E|F)P(F) + P(E|F^c)\big(1 - P(F)\big) \quad (7) \end{aligned}$$

Equation (7) states that the probability of the event E is a weight of the conditional probability of E given that F has occurred and the conditional probability of event E given that F has not occurred. Each conditional probability provides as much weight as the conditioned event tends to occur.

Equation (7) can be generalized as follows: suppose that events $F_1, F_2, \ldots F_n$ are mutually exclusive such that $\bigcup_{i=1}^{n} F_i = S$, where S is the sample space. In other words, exactly one of the events will occur (Figure 2).

We can write the above as:

$$E = \bigcup_{i=1}^{n} E_i \qquad (8)$$

From the definition of conditional probability, we have:

$$P(EF_i) = P(E|F_i)P(F_i) \qquad (9)$$

Furthermore, using the fact that the $\mathrm{EF_i}$ events, $\mathrm{i} = 1, \ldots, \mathrm{n}$ are mutually exclusive, we obtain that:

$$\mathrm{P(E)} = \sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{P(EF_i)}$$

$$= \textstyle\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{P(E|F_i)P(F_i)} \qquad (10)$$

Thus, equation (10) shows how, given events $\mathrm{F_1, F_2, \ldots F_n}$ of which one and only one can occur, $\mathrm{P(E)}$ can be calculated conditioning that $\mathrm{F_1}$ occurs. That is, it is established that $\mathrm{P(E)}$ is equal to the average of the weights of $\mathrm{P(E|F_i)}$ and each term is weighted by the probability of the event in which it is conditioned.

Now suppose that E has occurred and that you want to determine the probability that the event $\mathrm{F_j}$ has occurred. By equation (10) we have:

$$\mathrm{P\left(F_j \middle| E\right)} = \frac{\mathrm{P(EF_i)}}{\mathrm{P(E)}}$$

$$= \frac{\mathrm{P(E|F_j)P(F_j)}}{\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{P(E|F_i)P(F_i)}} \qquad (11)$$

Equation (11) is known as the Bayes formula. Thus, we can consider E as evidence of $\mathrm{F_j}$, and calculate the probability that $\mathrm{F_j}$ will occur given the evidence, $\mathrm{P(E|F_i)}$ Now suppose you have evidence from multiple sources. From equation (9):

$$\mathrm{P\left(F_j \middle| E_1E_2 \ldots E_m\right)} = \frac{\mathrm{P(E_1E_2 \ldots E_m|F_j)P(F_j)}}{\mathrm{P(E_1E_2 \ldots E_m)}} \qquad (12)$$

The above equation will be used to obtain results.

The assumption that gives rise to the adjective Naïve is the independence between the variables, which is not always true. However, the method has been successful in its application because the relevant information is contained in the relative magnitudes between the quantities and not so much in the values of the probabilities themselves.

K Nearest Neighbor Classifier

The K-nearest neighbor algorithm is simple, but who can give interesting results if the data range is large enough. It's about a classification method widely used in many fields and is found also among the top 10 data mining algorithms [22]. To make an analogy, it is possible to compare this algorithm to a neighborhood. Normally, houses that are close to each other have similar characteristics. We can group them and give them a classification. The algorithm uses this same logic to try to group the elements that are close to each other.

First of all, three parameters are to be taken into consideration: the sample data, the number of closest neighbors to select (K) and the point we want to evaluate (X). Subsequently, for each element of the sample, we evaluate the distance between reference point X and point X; of the set of learning and we check if the distance between them is less than one of the distances contained in the list of nearest neighbors. If so, the point is added to the list. If the number of items in the list is greater than k, the last value is simply removed from the list. The algorithm itself is not very complicated and can give a good result with brute force if sampling is not too big. However, since we are talking about data mining, the number of individuals to be evaluated is often very big, that's why an optimization algorithm is needed. There are many types of trees to speed up a search like the JCD-tree or the ball tree. The algorithm ball tree will be covered later in this report. Here is the pseudo-code representing the algorithm:

```
Algorithm: prediction of the class of a datum by the method of k nearest neighbors
Requires 3 parameters: a set of examples X, a given $x$ and $k \in \{1, \ldots k\}$
For each example $x_i \in X$
Calculate the distance between $x_i$ and $x$: $\delta(x_i, x)$
End for
For $j \in \{1, \ldots k\}$ do
KNN$(j) \leftarrow \arg\min \delta(x_i, x) \quad i \in 1, \ldots n$
$\delta(x_i, x) \leftarrow +\infty$
End for
Determine the class of x from the class of examples whose number is stored in the KNN.
```

3. SIMULATION RESULTS

Result analysis is done in MATLAB R2019 and test cases are taken from the UCI dataset. To perform the training and testing the dataset is divided 60:40 ratio. Where the training dataset is 60 % and the testing dataset is 40 %.

As the simulation parameter of BPSO, the number of swarms is taken 60, and the iteration cycle is 100.

a. Evaluation Criteria:

The formulas shown below are used to calculate accuracy, precision, and sensitivity.

Accuracy: Accuracy means what percentage of data is correctly classified:

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (13)$$

Precision (P): The percentage of correctly classified when predicting positivity is called precision:

$$\text{Precision} = \frac{\text{TP}}{\text{Total positive classified}} \quad (14)$$

Sensitivity: The percentage of classifying a positive class is called sensitivity:

$$\text{Sensitivity} = \frac{\text{TP}}{\text{Total positives}} \quad (15)$$

b. Results

BPSO-KNN

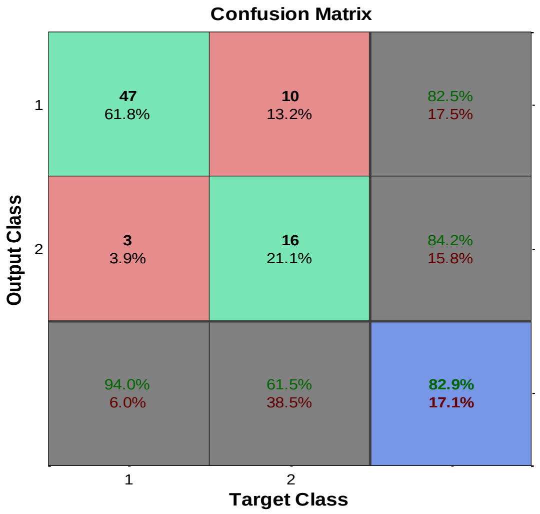


Figure 3: Confusion matrix plot for BPSO-KNN classifier with number of features=8

The above confusion plot explains the two-class Diabetes classification with 8 attributes. Where 76 samples are tested against the KNN based trained model. The archived accuracy in the classifier is 82.9%.

Here, TP=47, TN=16, FP=10, FN=3

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} = \frac{47 + 16}{47 + 16 + 10 + 3} = 82.9\%$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{47}{47 + 10} = 82.5\%$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{47}{47 + 3} = 94\%$$

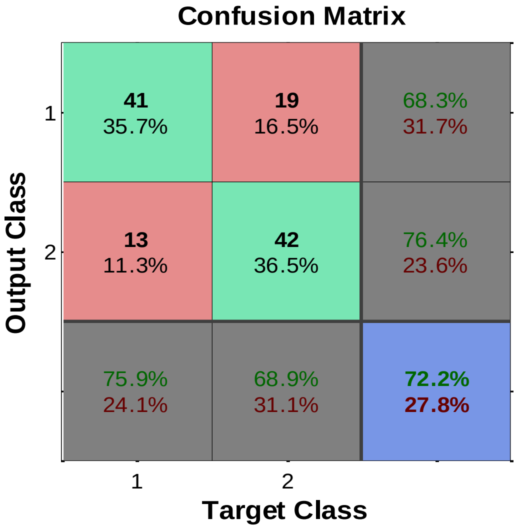


Figure 4: Confusion matrix plot for BPSO-KNN classifier with number of features=19

Here, TP=41, TN=42, FP=19, FN=13

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} = \frac{41 + 42}{41 + 42 + 19 + 13} = 72.2\%$$
$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{41}{41 + 19} = 68.33\%$$
$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{41}{41 + 13} = 75.92\%$$

BPSO-SVM

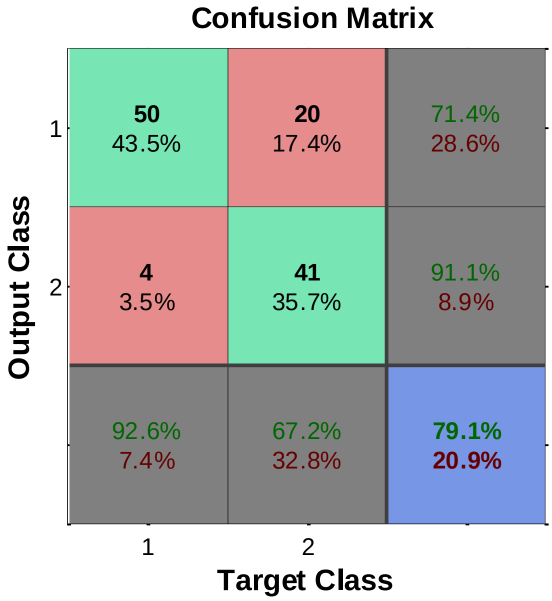
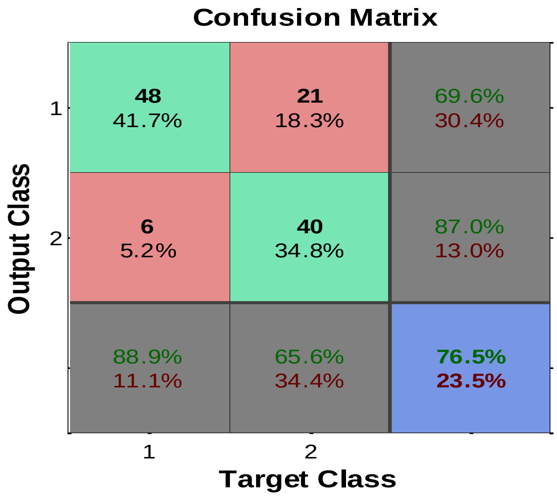


Figure 5: Confusion matrix plot for BPSO-SVM classifier with number of features=8

Here, TP=50, TN=41, FP=20, FN=4

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} = \frac{50 + 41}{50 + 41 + 20 + 4} = 79.1\%$$
$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{50}{50 + 20} = 71.42\%$$
$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{50}{50 + 4} = 92.6\%$$


Confusion Matrix
Output Class
Target Class

	1	2	
1	48 41.7%	21 18.3%	69.6% 30.4%
2	6 5.2%	40 34.8%	87.0% 13.0%
	88.9% 11.1%	65.6% 34.4%	76.5% 23.5%

Figure 6: Confusion matrix plot for BPSO-SVM classifier with number of features=19

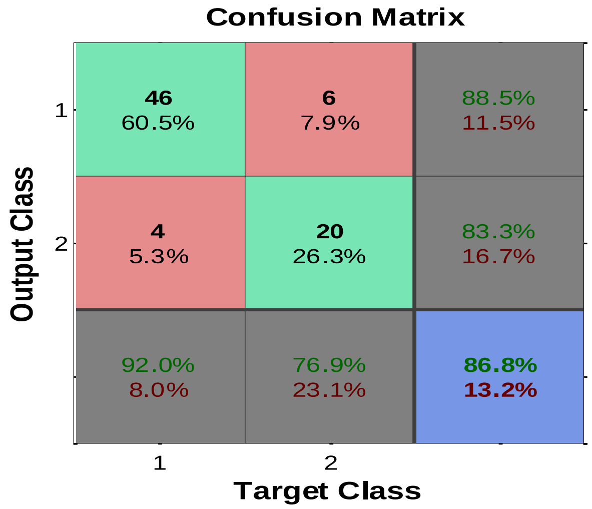
Here, TP=48, TN=40, FP=21, FN=6

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} = \frac{48 + 40}{48 + 40 + 21 + 6} = 76.52\%$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{48}{48 + 21} = 69.6\%$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{48}{48 + 6} = 88.9\%$$

BPSO-Naïve Bayes


Confusion Matrix
Output Class
Target Class

	1	2	
1	46 60.5%	6 7.9%	88.5% 11.5%
2	4 5.3%	20 26.3%	83.3% 16.7%
	92.0% 8.0%	76.9% 23.1%	86.8% 13.2%

Figure 7: Confusion matrix plot for BPSO-Naïve Bayes classifier with number of features=8

Here, TP=46, TN=20, FP=6, FN=4

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} = \frac{46 + 20}{46 + 20 + 6 + 4} = 86.8\%$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{46}{46 + 6} = 88.5\%$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP}+\text{FN}} = \frac{46}{46+4} = 92\%$$

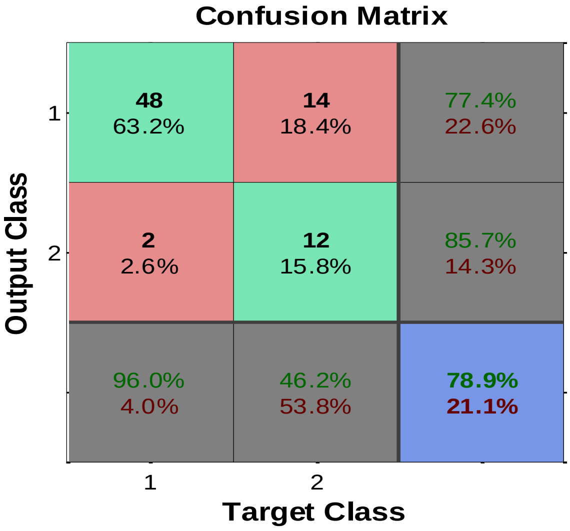


Figure 8: Confusion matrix plot for BPSO-Naïve Bayes classifier with number of features=19

Here, TP=48, TN=12, FP=14, FN=2

$$\text{Accuracy} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}} = \frac{48+12}{48+12+14+2} = 78.9\%$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP}+\text{FP}} = \frac{48}{48+14} = 77.41\%$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP}+\text{FN}} = \frac{48}{48+2} = 96\%$$

Table 1: Comparative analysis of accuracy

Database	**Number of Features**	**Accuracy for BPSO-KNN**	**Accuracy for BPSO-SVM**	**Accuracy for BPSO-Naïve Bayes**
Diabetes	8	82.9 %	79.1 %	86.8%
Diabetes2	19	72.2%	76.52 %	78.9 %

Table 1 describes the accuracy of proposed methods where attributes selected using BPSO and trained with different classifiers i.e. KNN, SVM, and Naïve Bayes algorithm. It can be observed that BPSO-Naïve Bayes yields higher accuracy of 86.8%, then other methods. There are two datasets used for Diabetes prediction. Diabetes 2 datasets are comprising 19 attributes which give the highest accuracy of 78.9% with Naïve Bayes.

Table 2: Comparative analysis of precision in percentage for two classes

Database	**Classes**	**Precision for BPSO-KNN**	**Precision for BPSO-SVM**	**Precision for BPSO-Naïve Bayes**
Diabetes	2	82.5%	71.42 %	88.5%
Diabetes2	2	68.33%	69.6 %	77.41 %

Table 2 describes the precision of proposed methods where attributes selected using BPSO and trained with different classifiers i.e. KNN, SVM, and Naïve Bayes algorithm. It can be observed that BPSO-Naïve Bayes yields higher precision of 77.41%, then BPSO-KNN (82.55) and BPSO-SVM (71.42 %) respectively. There are two datasets used for Diabetes prediction. Diabetes datasets are comprising 8 attributes which give the highest precision of 88.5 % with Naïve Bayes. The classifier is classified for 2 classes.

Table 3 Comparative analysis of sensitivity in percentage for two classes

Database	**Classes**	**Sensitivity for BPSO-KNN**	**Sensitivity for BPSO-SVM**	**Sensitivity for BPSO-Naïve Bayes**
Diabetes	2	94 %	92.6 %	92%
Diabetes2	2	75.92%	88.9 %	96 %

4. CONCLUSION

The KDD and data mining process show great potential in the search for solutions to various problems and questions in the most diverse areas of human knowledge. In the health field, it can contribute to disease prevention and health promotion, although there are still many challenges to be faced. This study demonstrated that the classification algorithms are able to demonstrate relationships in the data that are consistent with reality. However, it is necessary to consider that there are limitations and that the knowledge of specialists is essential to judge the rules presented. Considering the growing number of patients affected by diabetes mellitus with the consequent reduction in quality of life and also the increase in costs for Health Operators, the proposal to analyse the database of use of health plans to find standards that allow the classification of patients with indications of diabetes mellitus shows whether a useful and viable implantation process. The Naïve Bayes showed satisfactory results that can greatly assist in the search for patients with diabetes mellitus indications for the diagnosis and subsequent inclusion in preventive medicine programs. The maximum classification rate obtained with our method is 86.8%.

In terms of perspectives, the prediction of diabetes using learning methods can be broadened by using basic knowledge methods to increase the interoperability of the diagnosis.

REFERENCES

[1] Repalli, Pardha. "Prediction on diabetes using data mining approach." Oklahoma State University (2011).

[2] Soliman, O. S., & AboElhamd, E. (2014). Classification of diabetes mellitus using modified particle swarm optimization and least squares support vector machine. arXiv preprint arXiv:1405.0549.

[3] Lakshmi, K. R., and S. Prem Kumar. "Utilization of data mining techniques for prediction of diabetes disease survivability." International Journal of Scientific & Engineering Research 4, no. 6 (2013): 933-940.

[4] Anand, R., Kirar, V. P. S., & Burse, K. (2012). Data pre-processing and neural network algorithms for diagnosis of type ii diabetes: a survey. International Journal of Engineering and Advanced Technology (IJEAT) ISSN, 2249-8958.

[5] Tomar, D., & Agarwal, S. (2013). A survey on Data Mining approaches for Healthcare. International Journal of Bio-Science and Bio-Technology, 5(5), 241-266.

[6] Giveki, D., Salimi, H., Bahmanyar, G., & Khademian, Y. (2012). Automatic detection of diabetes diagnosis using feature weighted support vector machines based on mutual information and modified cuckoo search. arXiv preprint arXiv:1201.2173.

[7] Fayyad, Usama M., Gregory Piatetsky-Shapiro, and Padhraic Smyth. "Knowledge Discovery and Data Mining: Towards a Unifying Framework." In KDD, vol. 96, pp. 82-88. 1996.

[8] Vianna, R. C., C. M. Moro, Samuel Jorge Moyses, Deborah Carvalho, and Julio Cesar Nievola. "Data mining and characteristics of infant mortality." Cadernos de saude publica 26, no. 3 (2010): 535-542.

[9] Pima, A. Frank, and A. Asuncion. "Pima indians diabetes dataset." UCI Machine Learning Repository, University of California, Irvine (2010).

[10] Kennedy, James, and Russell Eberhart. "Particle swarm optimization." In Proceedings of ICNN'95-International Conference on Neural Networks, vol. 4, pp. 1942-1948. IEEE, 1995.

[11] Shi, Yuhui. "Particle swarm optimization: developments, applications and resources." In Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546), vol. 1, pp. 81-86. IEEE, 2001.

[12] Kennedy, James, and Russell C. Eberhart. "A discrete binary version of the particle swarm algorithm." In 1997 IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation, vol. 5, pp. 4104-4108. IEEE, 1997.

[13] Shi, L., Xi, L., Ma, X., Weng, M., & Hu, X. (2011). A novel ensemble algorithm for biomedical classification based on ant colony optimization. Applied Soft Computing, 11(8), 5674-5683.