

New Intelligent Classification Techniques for Diagnosis of Diabetes Mellitus based on Modified PSO

P.Palanikumar, M. Arun

Research Scholar, Dept of Electrical Engineering, Annamalai University, Chidambaram, India

Assistant Professor, Dept of Electrical Engineering, Annamalai University, Chidambaram, India

ABSTRACT: An overview of some of the classification techniques appearing in this paper for Diabetes diagnoses and treatment. Hybrid optimization techniques are in the form Artificial Neural networks, Genetic Algorithm, Fuzzy classifier, Support Vector Machine, Least Squares Support Vector Machine, Particle Swarm Optimization. All the techniques are analyzed using Pima Indians Diabetes Data set from UCI repository of machine learning databases. This review aims to conclude best optimization technique for Diabetes Mellitus Diagnosis

KEYWORDS: Diabetes Mellitus (DM), Artificial Neural networks (ANN), Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Least Squares Support Vector Machine (LS-SVM).

I. INTRODUCTION

Diabetes Mellitus, or simply Diabetes, is a group of metabolic diseases in which a person has high blood sugar, either because the pancreas does not produce enough insulin, or because cells don't respond to the insulin that is produced. The body needs insulin to use sugar, fat and protein from the diet for energy. Diabetes increases the risks of developing kidney disease, blindness, nerve damage, blood vessel damage and it contributes to heart disease.(6) There are 3 major types of DM, "Type-1", which results from the body's failure to produce insulin, and currently requires the person to inject insulin. This form was previously referred to as "insulin-dependent diabetes mellitus" (IDDM). The second type of DM is called "Type- 2" which results from insulin resistance, a condition in which cells fail to use insulin properly, sometimes combined with an absolute insulin deficiency. This type also named as "Non- insulin dependent diabetes mellitus" (NI-DDM) or "adult-onset diabetes". Finally, Gestational diabetes is a form of diabetes that occurs in pregnancy, and other forms of diabetes are very rare and are caused by a single gene mutation. Also, there have been many mathematical models proposed for diabetes. All these mathematical models have some parameters which would be the result of parameter estimation techniques that should be carried out by softwares. A woman who was at least 21 years old was tested for diabetes according to World Health Organization criteria. The objective of this data set is diagnosis of diabetes for woman. The binary response variable takes the values 'Zero' or 'One', where 'One' means a positive test for diabetes and 'Zero' is a negative test for diabetes. There are 268 (34.9%) cases in class 'One' and 500 (65.1%) cases in class 'Zero'. There are eight clinical findings: 1. Number of times pregnant 2. Plasma glucose concentration a 2 hours in an oral glucose tolerance test 3. Diastolic blood pressure (mm Hg) 4. Triceps skin fold thickness (mm) 5. 2-Hour serum insulin (μ U/ml) 6. Body mass index 7. Diabetes pedigree function 8. Age (years). The aim of this paper is to develop a hybrid optimization algorithm consists of Modified-PSO algorithm and LS-SVM classifier. Least squares support vector machines (LS-SVM) are least squares versions of support vector machines (SVM), which are a set of related supervised learning methods that analyze data and recognize patterns, and which are used for classification and regression analysis. particle swarm optimization (PSO) is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. It solves a problem by having a population of candidate solutions, here dubbed particles, and moving these particles around in the search-space according to simple mathematical formulae over the particle's position and velocity. Each particle's movement is influenced by its local best known position, but is also guided toward the best known positions in the search-space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solutions. The system is developed using Matlab. Section

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 5, Issue 8, August 2016

2 describes the problem background and related work. The proposed hybrid algorithm is introduced in section 3 while experimental results are presented in section 4. The last section is devoted to the conclusion and further research.

II. RELATED WORKS

In mathematics, computer science and operations research, mathematical optimization, also spelled mathematical optimization, alternatively named mathematical programming or simply optimization or optimization, is the selection of a best element (with regard to some criterion) from some set of available alternatives. In the simplest case, an optimization problem consists of maximizing or minimizing a real function by systematically choosing input values from within an allowed set and computing the value of the function. The generalization of optimization theory and techniques to other formulations comprises a large area of applied mathematics. More generally, optimization includes finding "best available" values of some objective function given a defined domain (or input), including a variety of different types of objective functions and different types of domains

2.1 Optimization Problem

An optimization problem can be represented in the following way: *Given:* a function $f: A \rightarrow \mathbb{R}$ from some set A to the real numbers *Sought:* an element x_0 in A such that $f(x_0) \leq f(x)$ for all x in A ("minimization") or such that $f(x_0) \geq f(x)$ for all x in A ("maximization"). Such a formulation is called an optimization problem. Many real-world and theoretical problems may be modeled in this general framework. Problems formulated using this technique in the fields of physics and computer vision may refer to the technique as energy minimization, speaking of the value of the function f as representing the energy of the system being modeled. Typically, A is some subset of the Euclidean space \mathbb{R}^n , often specified by a set of *constraints*, equalities or inequalities that the members of A have to satisfy. The domain A of f is called the *search space* or the *choice set*, while the elements of A are called *candidate solutions* or *feasible solutions*. The function f is called, variously, an objective function or cost function (minimization),^[2] a utility function or fitness function (maximization), or, in certain fields, an energy function. A feasible solution that minimizes (or maximizes, if that is the goal) the objective function is called an *optimal solution*.

- Linear programming (LP), a type of convex programming, studies the case in which the objective function f is linear and the constraints are specified using only linear equalities and inequalities. Such a set is called a polyhedron or a polytope if it is bounded.
- Nonlinear programming studies the general case in which the objective function or the constraints or both contain nonlinear parts. This may or may not be a convex program. In general, whether the program is convex affects the difficulty of solving it.
- Quadratic programming allows the objective function to have quadratic terms, while the feasible set must be specified with linear equalities and inequalities. For specific forms of the quadratic term, this is a type of convex programming.
- Stochastic programming studies the case in which some of the constraints or parameters depend on random variables.

2. Particle Swarm Optimization Algorithm

A basic variant of the PSO algorithm works by having a population (called a swarm) of candidate solutions (called particles). These particles are moved around in the search-space according to a few simple formulae.^[8] The movements of the particles are guided by their own best known position in the search-space as well as the entire swarm's best known position. When improved positions are being discovered these will then come to guide the movements of the swarm. The process is repeated and by doing so it is hoped, but not guaranteed, that a satisfactory solution will eventually be discovered. Formally, let $f: \mathbb{R}^n \rightarrow \mathbb{R}$ be the cost function which must be minimized. The function takes a candidate solution as argument in the form of a vector of real numbers and produces a real number as output which indicates the objective function value of the given candidate solution. The gradient of f is not known. The goal is to find

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 5, Issue 8, August 2016

a solution a for which $f(a) \leq f(b)$ for all b in the search-space, which would mean a is the global minimum. Maximization can be performed by considering the function $h = -f$ instead. Let S be the number of particles in the swarm, each having a position $x_i \in \mathbb{R}^n$ in the search-space and a velocity $v_i \in \mathbb{R}^n$. Let p_i be the best known position of particle i and let g be the best known position of the entire swarm. A basic PSO algorithm is then

PSO is based on the principle that each solution can be represented as a particle in the swarm. Each particle has a position in the search space which is represented by a vector $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, where D is the dimensionality of the search space. Particles move in the search space to search for the optimal solutions. Therefore, each particle has a velocity which is represented as $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$, during the movement, each particle updates its position and velocity according to its own experience and that of its neighbors. The best previous position of the particle is recorded as the personal best p_{best} , and the best position obtained by the population thus far is called g_{best} . Based on p_{best} and g_{best} , PSO searches for the optimal solutions by updating the velocity and the position of each particle according to the following equations:

$$x = x + v \quad (1)$$

$$v = v\omega + c_1 r_1 (p_{best} - x) + c_2 r_2 (g_{best} - x) \quad (2)$$

Where ω is inertia weight, which is to control the impact of the previous velocities on the current velocity. c_1 and c_2 are acceleration constants. r_1 and r_2 are random values uniformly distributed in $[0,1]$. The velocity is limited by a predefined maximum velocity, v_{max} and $v \in [-v_{max}, v_{max}]$. The algorithm stops when a predefined criterion is met, which could be a good fitness value or a predefined maximum number of iterations. Inertia Weight plays an important role in the process of providing balance between investigation and manipulation. It determines the contribution rate of a particles previous velocity to its velocity at the current time step. The basic algorithm step can be explained in three stages which are described as follow, (i) Evaluating the fitness value of each particle. (ii) Updating local and global best fitness and positions. (iii) Updating the velocity and the position of each particle. The Pseudo Code of the Particle Swarm Optimization is shown below

The Pseudo Code of the Particle Swarm Optimization

- Step1. Begin
- Step2. randomly initialize particles swarm
- Step3. while (the stopping criterion is not met)
- Step4. evaluate fitness of particles
- Step5. for $n = 1$ to number of particles
- Step6. find p_{best}
- Step7. find g_{best}
- Step8. for $d=1$ to number of dimension of particle
- Step9. update the position of particles by Eq. (1)-(2)
- Step10. next d
- Step11. next n
- Step12. update the inertia weight value
- Step13. next generation until stopping criterion is met
- Step14. End

2.2 Least Squares Support Vector Machine

Least Squares Support Vector Machines (LS-SVM) are reformulations to the standard SVMs which lead to solving linear KKT systems. LS-SVMs are closely related to regularization networks and Gaussian processes but additionally

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 5, Issue 8, August 2016

emphasize and exploit primal-dual interpretations. Links between kernel versions of classical pattern recognition algorithms such as kernel Fisher discriminant analysis and extensions to unsupervised learning, recurrent networks and control are available. Given a training dataset of N points $\{X_i, Y_i\}_{i=1}^N$, with input data $X_i \in \mathbb{R}^n$ output $Y_i \in \mathbb{R}$, the optimization problem considering

$$\min_{w, b, e} \left[\frac{1}{2} w^T w + \frac{1}{2} \sum_{i=1}^N \zeta_i e_i^2 \right] \quad (4)$$

$$e = Y_i - (w^T \phi(X_i) + b)$$

Where ζ is a regularization factor, e is the difference between the desired output and the actual output, and $\phi(\cdot)$ is a nonlinear function mapping the data points into a high dimensional Hilbert space; in addition, the dot product in the high-dimensional space is equivalent to a positive definite kernel function

$$K(X, X_i) = \phi(X)^T \phi(X_i) \quad (5)$$

In primal weight space, a linear classifier in the new space takes the following form Where W is the weight vector and b is called as the bias term. $b \in \mathbb{R}$. After elimination of w and e which generates the Support Vector α_i .

$$Y = (K + \mathbf{1}/\sigma) \alpha \quad (6)$$

Where $Y = [Y_1, Y_2, \dots, Y_N]$, $\sigma = [\sigma_1, \sigma_2, \dots, \sigma_N]$ and Kernel matrix is $K \in \mathbb{R}^{N \times N}$ The resulting estimated function is

$$Y(x) = \sum \alpha_i K(X, X_i) \quad (7)$$

Radial basis function RBF kernel is

$$K(X, X_i) = \exp(-\|X - X_i\|^2 / \sigma^2) \quad (8)$$

The Pseudo Code of the Least Squares Support Vector Machine

- Step 1: Load the training data set of n data points $\{X_i, Y_i\}_{i=1}^n$, with input data $X_i \in \mathbb{R}^n$ output $Y_i \in \mathbb{R}$
- Step 2: Generate random weights for each input data point.
- Step 3: Determine the value of the bias term b and initialize the error e for each point randomly.
- Step 4: Initialize and using random values.
- Step 5: Search for values of e , w and b that minimize the objective function, (3) and, (4).
- Step 6: Construct the Lagrangian function with the solution that must satisfy the KKT conditions in the set of, (5)
- Step 7: Calculate number of support vector $s(\infty)$ using, (6)
- Step 8: Training data for LS-SVM model could be classified using, (7) with RBF kernel function, (8)
- Step 9: Classify any new point by, (10) using RBF kernel function, (8)
- Step 10: Loop until stopping criteria is met, usually until reach the maximum number of iterations.

III. PROPOSED WORK

A modified version of PSO was proposed, The main idea of this modified version is as in the following equations. For the i^{th} particle in d dimension, it could update its velocity and position using, (9) and, (10)

$$x = x + (\omega v) \quad (9)$$

$$v = \lambda [v + c_1 r_1 (p_{best} - x) + c_2 r_2 (g_{best} - x)] \quad (10)$$

where λ is a convergence factor, which can be calculated using following formula

$$\lambda = 2 / |2 - c - \sqrt{c^2 - 4c}| \quad (11)$$

Where $c = c_1 + c_2$

In this proposed method ω is calculated using the formula

$$\omega = 0.9 - \left(\frac{t}{T_{MAX}} \right) * 0.5 \quad (12)$$

Algorithm-1: Modified-PSO

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 5, Issue 8, August 2016

Step 1: Initialize population of particles $X(t)$ which consists of random positions x_1, x_2, \dots, x_n and velocities $V(t)$ are made up of the particle's initial velocity v_1, v_2, \dots, v_n on 'n' dimensions.

Step 2: Evaluate the fitness for each particle.

Step 3: For each particle, find the maximum fitness and compare it to the best found so far (p_{best}), if $f(x_i)$ less than $f(p_{best})$ then $f(p_{best}) = x_i$

Step 4: Set P_i equals to the location of the maximum fitness value x_i

Step 5: Compare fitness evaluation with the population's overall previous best. If current value is better than g_{best} , then reset g_{best} to the current particle's array index and value.

Step 6: Calculate the convergence factor using (11)

Step 7: Calculate the Inertia weight using (12)

Step 8: Update the position of the particle according to (9) and (10) and the new population (+ 1) will be generated.

Step 9: Adjust the acceleration of the particles using,

Step 10: Loop to step (2) until stopping criterion is satisfied (Reach a maximum number of iterations)

A hybrid classification algorithm which integrates Modified-PSO algorithm as a parameters optimization technique and LS-SVM for classification was proposed. The proposed algorithm for DM diagnosis and treatment is composed of two main stages, Parameters Optimization and Classification. Modified-PSO algorithm was used as a parameters optimization technique aiming to improve the setting of the parameter values of LS-SVM. Hence, overcoming its sensitivity to the parameter values changes. Classification stage using LS-SVM technique consists of two steps, Training followed by a Testing. The aim of parameters optimization stage using Modified-PSO is to find the optimal values for the parameters of the LS-SVM classifier. The second stage utilizes LS-SVM to classify the DM patients into one of two classes (Live/Die) using the optimized parameters.

Algorithm-3: Proposed Algorithm:

Step 1: Load the data set of n data points, $\{ X_i, Y_i \}_{i=1}^N$, where X_i is the input vector and $Y_i \in R$ is the corresponding target with values $\{-1, +1\}$.

Step 2: Generate random weights for each input data point.

Step 3: Initialize the bias term b and the error e for each point randomly.

Step 4: Find the optimal values for ζ and σ using algorithm 1.

Step 5: Find optimal values of (e,w,b) for the objective function in (3) and (4).

Step 6: Calculate number of support vectors (α) using (6)

Step 7: Classify any new point by using RBF kernel function, (8)

Step 8: Loop until stopping criteria is met, usually until reach the maximum number of iterations

IV. EXPERIMENTAL RESULTS

In this section, we present the performance evaluation methods used to evaluate the proposed method. Finally, we give the experimental results and discuss our observations from the obtained results. The proposed algorithm worked on Pima Indians Diabetes Data Set. Some of the records in this data set contains zeros in some features. Cells with zero entry were kept unchanged. The input for the Modified-PSO algorithm is total of 768 records. About 768 random individuals are generated in the search space for 100 Iterations. The output from the Modified-PSO is the optimal values for ζ and σ , which are 100 and 0.5 respectively. LS-SVM classifier is run with the optimized parameters and RBF kernel function, (8), seeking to find the optimal hyper-plan that separates the search space into 2 classes (Live, Die) by finding the optimal values for (w, e and b) in the objective function, (3) and (4) the classification accuracy was calculated using (13)

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

where TP, TN, FP and FN denote respectively:

True Positives (TP): Classifies Diabetic as Diabetic.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 5, Issue 8, August 2016

True Negatives (TN): Classifies No Diabetic as No Diabetic. False Positives (FP): Classifies No Diabetic as Diabetic. False Negatives (FN): Classifies Diabetic as No Diabetic

The performance of proposed method is evaluated using 10-fold cross-validation test. In this section, the obtained results are reported. The results with all features and optimal features found through modified PSO search in terms of Accuracy is shown in Table 1.

Fig(1) demonstrates the results of the average classification accuracy of the proposed algorithm against the accuracy of other classification algorithms applied on the same data set. Results show the effectiveness of the proposed Algorithm, which has the maximum average classification accuracy of 97.833% over other algorithms. TABLE(II) also articulates the average classification accuracy obtained from all algorithms in addition to the number of records used from Pima Indians Diabetes Data set in each algorithm

TABLE I. ACCURACY OF 10-FOLD CV

Fold	Accuracy
1	93.993%
2	95.973%
3	96.889%
4	99.9769%
5	97.991%
6	98.698%
7	96.999%
8	99.988%
9	99.99%
10	97.83%

V. CONCLUSION AND FUTURE WORK

Diabetes is a complex and complicated disease characterized by either lack of insulin or a resistance to insulin, a hormone which is crucial for metabolism of blood sugar. This paper introduced a hybrid optimization Algorithm for DM patients. The proposed Algorithm integrates modified version of PSO and LS-SVM algorithm. The proposed algorithm was implemented on Pima Indians Diabetes Data set from UCI repository of machine learning databases. The average classification accuracy of LS-SVM method with RBF kernel was 97.833% which is the best while compared with other algorithms which worked on the same data set. As a future work, Ant Colony System (ACS) could be used as an optimization technique. Also, other kernel functions could be applied in the classification stage.

TABLE II. AVERAGE CLASSIFICATION ACCURACY OF THE PROPOSED ALGORITHM AND OTHER CLASSIFICATION ALGORITHMS

Method	Accuracy
GRNN	82.99%
Modified PSO fuzzy classifier	85.19 %
PSOANN	70.27%
Genetic Algorithm	80%
The Proposed Algorithm	97.833%

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Website: www.ijirset.com

Vol. 5, Issue 8, August 2016

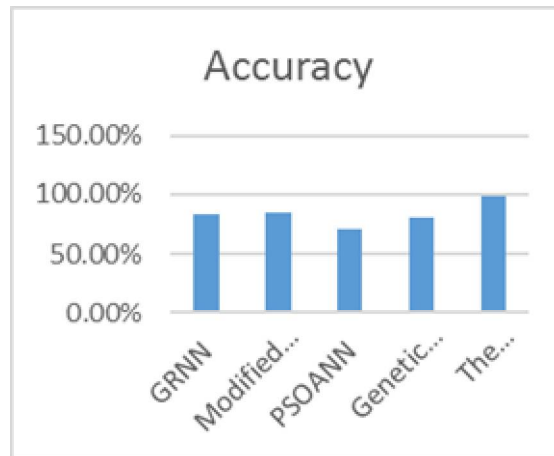


Fig. 1. Average Classification Accuracy of DM using the proposed algorithm and other classification algorithms

REFERENCES

- [1] Centers for disease control and prevention, national diabetes, fact sheet, www.cdc.gov/diabetes (2011).
- [2] World health organization, diabetes center, fact sheet n312, www.who.int/mediacentre/factsheets/fs312/en (2011).
- [3] http://ftp.ics.uci.edu/pub/ml-repos/machine-learning_databases/pima-indians-diabetes, 2003.
- [4] International Diabetes Federation 2011, Diabetes Atlas 4rd Edition, IDF
- [5] Hamid Reza Sahebi., Sara Ebrahimi "AFuzzy ClassifierBased on Modified Particle SwarmOptimizationfor DiabetesDisease Diagnosis" ACSIJ Advances in Computer Science: an International Journal, Vol. 4, Issue 3, No.15 , May 2015
- [6] S.Karthikeyeni, S.Saranya "Diabetes Prediction in Women Based On Soft Computing Techniques" International Journal Of Engineering And Computer Science ISSN:2319-7242 Volume 3 Issue 9, September 2014 Page No. 8439-8444