Prediction of the Onset of Diabetes Using Artificial Neural Network and Pima Indians Diabetes Dataset

Conference Paper · December 2020

DOI: 10.1109/ICRAIE51050.2020.9358308

CITATIONS
22

READS
1,091

5 authors, including:

Kamlesh Lakhwani
JECRC University
86 PUBLICATIONS **352** CITATIONS
SEE PROFILE

Kamal Kant Hiran
Aalborg University
92 PUBLICATIONS **899** CITATIONS
SEE PROFILE

Devendra Kumar Somwanshi
Poornima Group of Colleges
69 PUBLICATIONS **309** CITATIONS
SEE PROFILE

Prediction of the Onset of Diabetes Using Artificial Neural Network and Pima Indians Diabetes Dataset

Kamlesh Lakhwani
Dept. of Computer Science & Engg
Lovely Professional University
Punjab, India
kamlesh.lakhwani@gmail.com

Sandeep Bhargava
Dept.of Computer Science & Engg
Poornima College of Engineering
Jaipur, Rajasthan, India
eng.san83sandy@gmail.com

Kamal Kant Hiran
Dept. of Computer Science & Engg
Sir Padampat Singhania University
Udaipur,Rajasthan,India
kamalhiran@gmail.com

Mahesh M. Bundele
Dept of Electronics & Communication
Poornima College of Engineering
Jaipur,Rajasthan,India
maheshbundele@poornima.org

Devendra Somwanshi
dept of Electronics & Communication
Poornima College of Engineering
Jaipur,Rajasthan,India
imdev.som@gmail.com

2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE) | 978-1-7281-8867-6/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICRAIE51050.2020.9358308

***Abstract*—When a human body unable to respond to the insulin properly and/or unable to produce the required amount of insulin to regulate glucose, it means that the human body is suffering from Diabetes. Diabetes increases the risk of developing another disease like heart disease, kidney disease, and damage to blood vessels, nerve damage, and blindness. The diagnosis of diabetes using proper analysis of diabetes data is a significant problem. In this paper, an automatic diagnosis system is introduced and analyzed. For this purpose, a Three-Layered Artificial Neural Network (ANN) and Pima Indians Diabetes dataset are used. In this ANN based prediction model, a logistic-activation-function for activation of neurons, and the Quasi Newton method is used as the algorithm for the training. As a result cumulative gain plot and as a measure of the quality of this model the maximum gain score is used.**

***Keywords*—Artificial Neural Network, Classification, Regression, Diabetes Prediction, Pima Indians Diabetes, Quasi- Newton method, Cumulative gain**

I. Introduction

When a human body is unable to respond to the insulin properly and/or unable to produce the required amount of insulin to regulate glucose, it means that the body is suffering from Diabetes. Diabetes increases the risk of developing another disease like heart disease, kidney disease, and damage to blood vessels, nerve damage, and blindness [1], [2]. According to the survey report named National Diabetes and Diabetic Retinopathy Survey released by Health and family welfare ministry of India; in the last four years, the prevalence of diabetes remained at 11.8% in India. The survey was conducted by the Rajendra Prasad Centre for Ophthalmic Science during 2015-2019 [1]. According to a report released by the All India Institute of Medical Sciences (AIIMS), New-Delhi, India, The prevalence of known cases of diabetes was 8.0%, and new cases of diabetes 3.8%. Comprised of known-diabetics 67.3 percent of participants, while 32.7 percent were new-diabetics. Males had a higher diabetes prevalence (12 percent) as females (11.7 percent) [1]. The estimated global prevalence of diabetes among adults over 18 years of age was 8.5 per cent in 2014, conferring to the World Health Organization (WHO). In Indias adult population, probably 72.96-million cases are of diabetes. The prevalence in urban areas ranged from 10.9% to 14.2%, In rural India, the prevalence was 3.0-7.8%, from the population age group 20 years and above, with a much higher prevalence among individuals over the age of 50 (INDIAB Study) [1]. The complete paper has been organized into six segments including a segment of Introduction, further segments are Motivation, About Dataset, ANN Inspired Diabetes Prediction Model, Training-Strategy, Result, and Conclusion.

II. Motivation

The diagnosis of diabetes using proper analysis of diabetes data is a significant problem. Classification can play an important role in analysis and decision making. Especially in the medical sciences, for decision making, classification has been considered an important tool. In recent years, artificial neural networks based classification techniques have been recommended as an alternate. In this paper, an artificial neural network is used to predict the onset of DM (diabetes mellitus) in Pima Indian women [3].

III. About Dataset

The dataset originates from the National Institute of Diabetes and Digestive and Kidney Diseases. The dataset aims to diagnose the patient having diabetes or not, it is based on some diagnostic measures involved in the dataset. Many constraints have been placed on choosing those instances from a broader database. All the patients here are particularly female of Pima Indian descent who are at least 21 years old. The datasets are made up of several independent medical predictor variables and one dependent target variable, Outcome. Independent variables comprise, the BMI, count of pregnancies the patient has, their level of insulin, age, etc. [3]–[5]. The description of the data is given below in Table I.

978-1-7281-8867-6/20/$31.00 2020 IEEE

TABLE I
DATA DESCRIPTION

Sr.no.	Variables	Description
1.	Gl_AsGlucose	The concentration of Plasma Glucose:in two hours oral Glucose Tolerance test
2.	ST_AsSkinThickness	TST[mm] TricepsSkinfoldThickness
3.	BMI_AsBodyMassIndex	BodyMassIndex [weightInKg]/[heightIn M2]
4.	Ag_AsAge	Age of the person in [years]
5.	Pg_AsPregnancy	Frequency of Pregnancy
6.	BP_AsBloodPressure	BP_AsBloodPressure
7.	Ins_AsInsulin	Two Hrs. SerumInsulin [muU/ml]
8.	DPF_AsDiabetes PedigreeFunction	A Pedigree function for Diabetes
9.	OC_AsOutcome	Outcome: variable 1 for positive and 0 for negative

IV. METHODOLOGY

Classification plays an important role in segregating the data. A computer-based classification model can be used to produce a more accurate and quick result. In the recent era, artificial intelligent and neural network are getting popular due to their learning capabilities and accuracy for data classification and prediction. In this paper, an ANN inspired diabetes prediction model has been proposed. The flow chart of the proposed model has been shown in figure 1 and, explained in subsequent sections.

V. EXPERIMENTAL SETUP

The neural networks characterize the predictive model. To implement this research work, a Neural Designer tool is used to design an artificial neural network.

A. Input

The total number of input variables used in the model is eight (8). Table-II shown below describes some basic information about these inputs, including the units, then name, and the explanation.

B. Scaling Layer

This layer having a size similar to the size of input i.e. eight (8). For this layer, an Automatic method is used for scaling. Table II displays the values used to scale inputs, which include, maximum, minimum, standard deviation, and mean.

TABLE II
VALUE USED FOR SCALING THE INPUT

	Minimum	Maximum	mean	deviation
Pregnancies	0	17	0	1
Glucose	0	199	0	1
SkinThikness	0	99	0	1
Insulin	0	846	0	1
BMI	0	67.1	0	1
Diabetes PedigreeFunction	0.078	2.42	0	1
Age	21	81	0	1

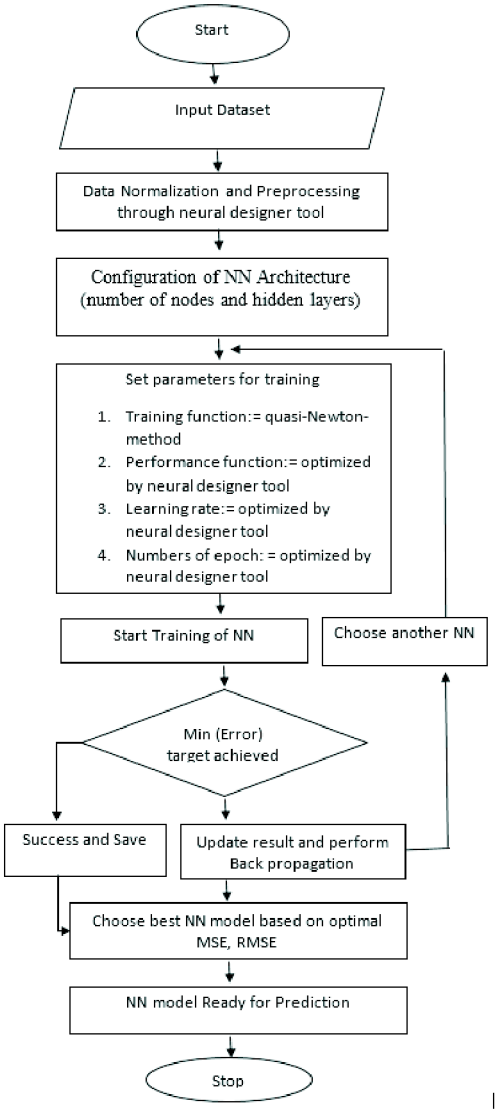

Fig. 1. Flow chart of the proposed model.

C. Layers in Neural-network

There are three layers in this model. The size of each layer and its activation function is shown in the table-III, that is shown below. The format of this neural network architecture is 8:3:1. Whereas 8 input nodes, 3 processing nodes, and 1 output node.

TABLE III
SIZE OF ANN LAYERS

Layer. no.	Inputs no.	Output no.
1	8	3
2	8	1
3	3	1

D. Neural network architecture

The pictorial illustration of the artificial neural network architecture is shown below in figure-2. This architecture has three layers, that includes a layer of scaling, a probabilistic layer, and a layer of neural network. The scaling neurons are represented by yellow circles, perceptron neurons in the blue circles, and probabilistic neurons in the red circle. There are eight inputs and one output. The number of hidden neurons is three that represents the complexity of the model.

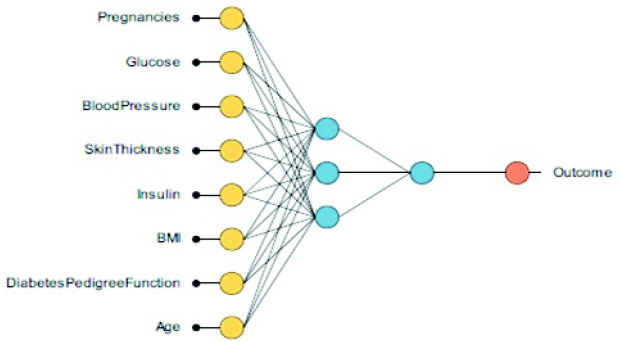

Fig. 2. Neural Network graph

E. Loss_index

The loss index is an important component of a neural network. The loss index can be differ for different applications. The task neural network has to do is defined by the loss index. The loss index is also used to measure the quality of the Self-learning capability of a neural network.

F. Error method

As an error method, the weighted squared error technique is used in this model. When data-set has an unbalance output/target then weighted squared error is useful. If there are limited negative values as compare to positive values then the target is unbalanced. In this research work the value of positive weight parameter is 1.87 and value of negative weight parameter is 1.

G. Regularization-method

Neural parameters norm are used for the regularization method. By decreasing the value of this parameter complexity of neural-network can be controlled [6]–[8]. The value of the neural parameter norm weight used in the model is 0.0017. The weight of this concept of regularisation is in the loss-expression.

VI. Model for Training

The methodology used to complete the learning procedure is called training/preparing (or learning) technique/strategy. In neural network, to get the most ideal and possible loss training strategy is applied to the system.

A. Algorithm for Training

A quasi-Newton-method is used as a training algorithm in this system. Quasi-Newton-Method is based on Newtons method except for the requirement of second derivatives calculation. Instead of calculating the second derivative of the quasi-Newton method at every algorithm Iteration, this method calculates an approximation of the inverse Hessian using gradient information.

B. Losses history of Quasi Newton method

The losses in each cycle of repetition are shown in the graph displayed in figure-3. Training loss value 1.10079 is found for initial values and found 0.471762 after 151 repetitions consequently. Selection loss value 1.11062 is found for the initial values and found 1.03511 after 151 repetitions.

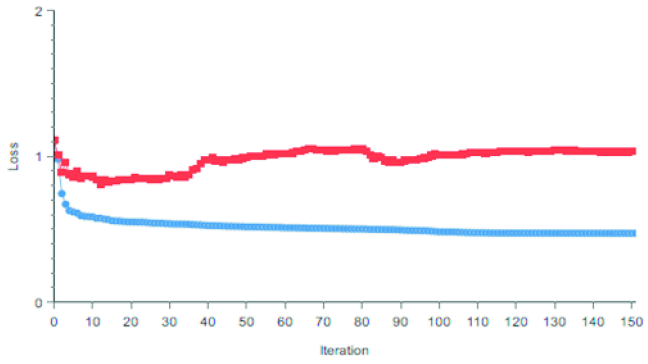

Fig. 3. Quasi-Newton method losses history

VII. Results

Quasi-Newton method results: The training result by using quasi-Newton method is shown in the table-IV given below. Some concluding states from the neural_network, training algorithm, and the loss function, are included in it.

TABLE IV
Quasi-Newton Method Results

Parameter	Value
Finishing parameters average	37.4
Closing Loss	0.472
Closing SelectionLoss	1.04
Closing GradientMean	0.000981
Repetitions	151
LapsedTime	0:03
Stop Condition	Gradient mean goal

A. Testing_errors

All kinds of losses during the testing of this model are measured in this task. It considers each preowned occasion and assesses the model for each utilization.

B. Errors table

Table V shows all the errors of the data for each use of them.

TABLE V
ERRORS TABLE

Error type	Training	Selection	Testing
Sum squared error	51.7483	38.3264	33.4827
Mean squared	0.112009	0.250499	0.218841
Root mean squared	0.334678	0.500499	0.467805
Normalized squared	0.494779	1.08793	0.966577
Cross entropy error	0.666707	1.75652	1.47763
Minkowski error	64.3526	43.4166	38.1428
Weighted squared	0.434355	1.03511	0.832647

C. Output Histogram

The output histogram is shown in figure 4. This histogram shows how the output is distributed according to testing instances. The abscissa signifies the centers of the containers, and therefore the ordinate their corresponding frequencies. Value zero (0) is the minimum frequency, which resembles the bin with a center of 0.650002. Value fifty- one (51) is the maximum frequency, which resembles the bin with a center of 0.95.

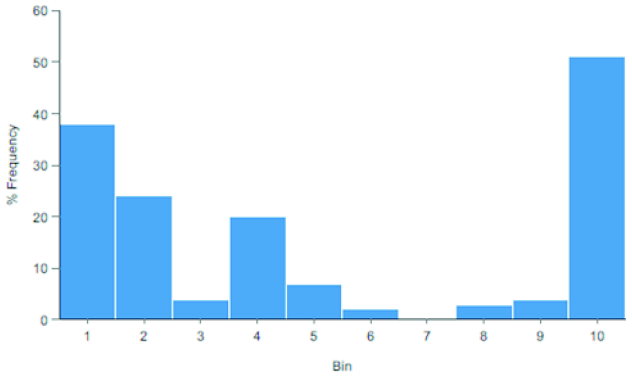

Fig. 4. Output Histogram

D. Cumulative-gain

Cumulative gain is a graphical representation that demonstrates the value of utilizing a random-opposed analytical model [9], [10]. It is composed of three sections or lines. First, the Baseline reflects results obtained without the use of a model. Second, the positive cumulative gain, which displays the number of positive y-axis instances against the percentage of the x-axis population. Essentially, the negative normal advantage mirrors the quantity of negative examples recognized against the populace rate. For instance, 0.8 and 0.2 are the total positive gain and negative gain values for a level of populace 0.5, implies that, in the wake of investigating half of the complete populace, we find 80% of all the positive and 20% of the negative while we'd find 50 percent of the positives and 50 percent of the negatives with a random classifier. The extreme gain score is used as a measure of the models quality.

E. Cumulative gain plot

The chart shown in figure 5 shows the result analysis. The positive cumulative gain is shown through the blue line. A negative cumulative gain is represented through the red line, and the cumulative gain for a random classifier is shown through a grey line. A good classifier shows the more separation distance between the positive and negative charts.

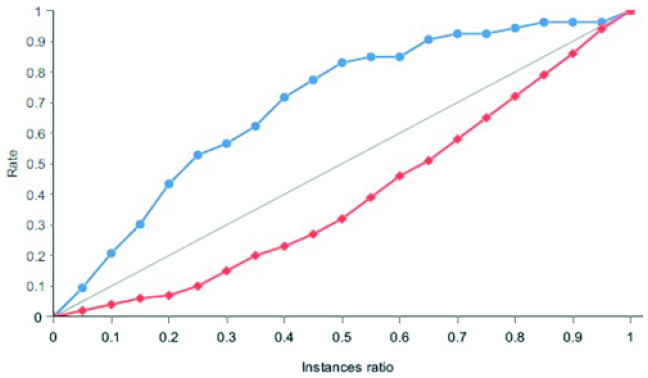

Fig. 5. Cumulative-gain plot

F. MGS(MaximumGainScore)

The MSG(MaximumGainScore) is the clear variance among CNG(CumulativeNegativeGain) and CPG (CumulativePositiveGain) [11]–[13]. Moreover, MGS is the fact wherever the % of negative occasions establish minimum, and the % of positive occasions establish maximized. For a perfect model value of the gain-score would be one (1). In this case and the number of instances it is applied, for instance ratio 0.5, maximum gain score is 0.510189.

G. Neural network outputs

For every set of applied inputs, a neural network produces a collection of outputs. The outputs are dependent on parameter values [14], [15]. The output shown in figure 6 displays the values of input and the resultant values of output. Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction, and Age are the input variables; and Outcome is the output variable.

	Value
Pregnancies	3.84505
Glucose	120.895
BloodPressure	69.1055
SkinThickness	20.5365
Insulin	79.7995
BMI	31.9926
DiabetesPedigreeFunction	0.471876
Age	50.2409
Outcome	0.494677295

Fig. 6. Output:with input output values

H. Directional output

It is very interesting to see, how the outputs differ when all the others are set as a function of a single input. This can be seen as the architecture of the neural network being cut along a certain input path and by some reference point.

I. Outcome against Pregnancies directional line chart

The next plot shown in figure-7, exhibits the output Outcome as a function of the input Pregnancies. The x and y axes are determined by the Pregnancies and Outcome variables, respectively.

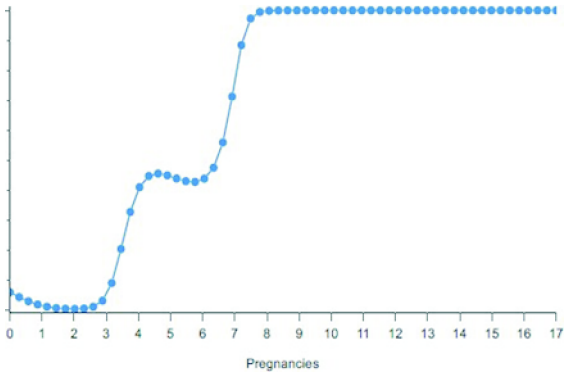

Fig. 7. Outcome against Pregnancies directional line chart

J. The outcome against Glucose directional line chart

The next plot figure 8, shows the output Outcome as a function of the input Glucose. The x- and y-axes are defined by the Glucose and Outcome variables set, respectively.

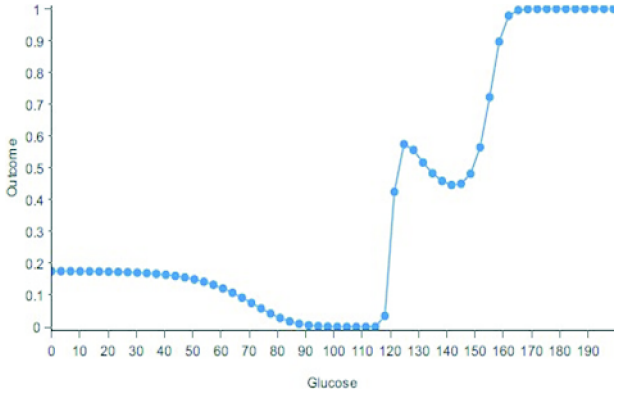

Fig. 8. Outcome against Glucose directional line chart

K. Outcome against BloodPressure directional line chart

The next plot (fig.9) shows the output "Outcome" as a function of the input BloodPressure. The x and y axes are demarcated by the range of the variables BloodPressure and Outcome, respectively.

L. Outcome against Insulin directional line chart

The next plot (fig.10) displays the output Outcome as a function of the input Insulin. The x axes and y axes are defined by the range of the variables Insulin and Outcome, respectively.

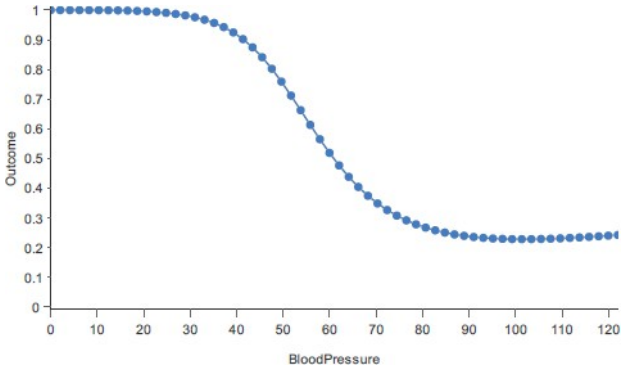

Fig. 9. Outcome against BloodPressure directional line char

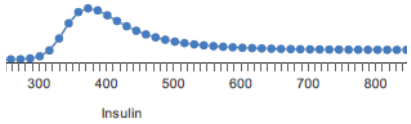

Fig. 10. Outcome against Insulin directional line chart

M. Outcome against DiabetesPedigree Function directional line chart

The next plot (shown in fig.11) displays the output Outcome as a function of the input DiabetesPedigreeFunction. The axes x and y_axes are definite by the series of the variables DiabetesPedigreeFunction and Outcome, correspondingly.

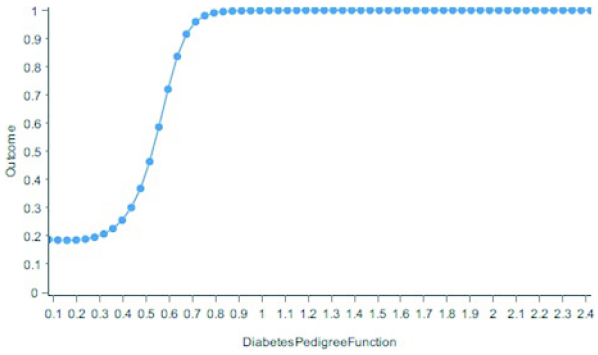

Fig. 11. Outcome against DiabetesPedigree Function directional line chart

N. Outcome against Age directional line chart

The next plot (shown in fig.12) displays the output Outcome as a function of the input Age. The x axes and y axes are defined by the series of the variables Age and Outcome, respectively.