ISSN No. 0976-5697

Volume 7, No. 5, September-October 2016

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

# Detection of Parkinson's disease using Neural Network Trained with Genetic Algorithm

Sayantan Naskar
Department of Information Technology
Academy of Technology
Aedconagar, Hooghly – 712121, India

***Abstract:*** Recent research works have focused on detection of Parkinson's disease using several machine learning techniques. Accurate separation of normal persons in the subjects under consideration from the persons being affected by Parkinson's disease is a challenging job. In the present work Neural Network (NN) has been trained using Genetic algorithm (GA employed to detect persons being affected with Parkinson's disease. The initial weight vector to the input layer of the NN has been optimized gradually using the optimization techniques to enhance the performance of NN to a greater extent. The experimental results of the proposed method have been compared with a well-known Multilayer Perceptron Feed-Forward Network (MLP-FFN) and also with the NN. Performance measures like accuracy, precision, recall and F-measure have been used to compare the performances of the algorithms. The experimental results have revealed significant improvement over the existing performances to detect Parkinson's disease using GA.

***Keywords:*** Parkinson's Disease; Genetic Algorithm; ANN; MLP-FFN; Gradient Descent

## I. INTRODUCTION

Application of machine learning can be found in several fields. Recognition or detection of life threating diseases is one of the most well studied applications. In terms of machine learning the task is to apply mainly classification to take the decision if a patient is suffering of a certain disease or not. Researchers have done some important breakthrough in the diagnosis process by applying machine learning which has to some extent shown effective capabilities of taking vital decisions related with medical sciences. One such application is detection of Parkinson's disease. Parkinson's disease is a neurodegenerative disorder which causes several motor and non-motor deficits which are result of the loss of dopamine producing brain cells. Dopamine is the neurotransmitter (chemical) that transmits signals between areas in the brain that is responsible for perfect coordination between muscle movements. As Parkinson's disease grows the amount of dopamine produced by brain decreases and it makes it almost impossible for the person to control his/her movements. It generally affects persons who are above 50 years old. The progress rate of Parkinson's disease varies from patient to patient. The symptoms include several common things like shaking, difficulty in walking, talking etc. But in addition several other symptoms like depression, emotional changes, difficulty in swallowing, chewing and even skin, urinary problems, sleep disruptions have also been detected.

Correct diagnosis of Parkinson's disease is an important job as far as the clinical research is concerned. Clinicopathological studies of brain bank material from UK and Canada has revealed that 25% percent of patients are incorrectly found to have Parkinson's disease by clinicians. The most common reasons behind the misdiagnosis are found to be atypical Parkinson an syndromes, presence of essential tremor, and vascular Parkinsonism [8].

Though medical science has done sufficient development in the treatment procedure but the diagnosis and detection of the disease is still a challenge as it is sometime hard to detect the Parkinson's disease at earlier stage. As there is no standard and proven laboratory test to detect the disease, the diagnosis mainly depends on the age, medical history, neurological examinations, interviewing with patients and symptoms only.

Recently, a group of researchers extracted some features from the voices of the people with Parkinson's disease that can be used to differentiate those who have the disease from those who do not. Max Little [1] has reporteda mechanism that involves machine learning to detect differences in voice patterns, in order to spot distinctive clues associated with Parkinson's. A large amount of data with prior knowledge of status of the disease is used to train the database to learn how to extract true symptoms of the disease from other factors. An algorithm has been proposed to detect changes in voice purely associated with Parkinson's. A new measure of dysphonia, pitch period entropy (PPE), which is robust to many uncontrollable confounding effects like noisy acoustic environments and normal, healthy variations in voice frequency, has been produced. Using a kernel support vector machine (SVM) an overall correct classification performance of 91.4% has been reported.

The detection of Parkinson's disease at an earlier stage could be lifesaving. Thus design of good classification mechanism to distinguish between persons having Parkinson's and those who are not is a challenging and important job. Several attempts have been found in literature, most of them are application of ANNs. Multi-Layer Perceptron (MLP) with Back-Propagation learning algorithm have been employed to classify patients into patients having Parkinson's disease and not having Parkinson's disease [2]. Hybrid system combining both neural networks and SVM has been proposed [3] to tackle the problem and claimed reasonable accuracy. The authors have claimed a high degree of certainty, above 90%. MLP and Radial Basis Function Neural Network (RBFNN) have been

employed along with Adaptive Neuro Fuzzy Classifiers (ANFC) [4]. Adaptive Neuro-Fuzzy Classifier (ANFC) with linguistic hedges has been used for feature selection from the dataset. Adaptive Neuro-Fuzzy Classifier with linguistic hedges has given the best results with %95.38 in training and %94.72 in testing phase. Optimization techniques like Genetic algorithm, Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC) algorithm are applied to train RBFNN to solve the classification task and reported accuracy 93.7% and 91.2% respectively for training and testing phases [5]. Artificial Bee Colony based RBF neural networks with four neurons have been found to the best in terms of accuracy and speed. The authors have reported best simulation result of 93.7% in training and 91.2% in testing phase of classification for Parkinson's disease diagnosis. Besides other research works involving application of Probabilistic Neural Network (PNN) [6] aided with Incremental search, Monte Carlo and Hybrid search. The authors have reported diagnosis accuracies ranging between 79% and 81% for new, undiagnosed patients. A comparative study [7] has reported a comparison between Neural Network, DMneural, Regression and Decision Tree. Several evaluation methods have been employed for evaluating the performance of the classifiers. According to the experimental results, neural networks have yielded the best results. The overall classification accuracy for neural network is 92.9%.

Recent research works have indicated the success of applying global optimization [18 - 21] in training phase of ANN [14 - 17]. Motivated by this; in the present work we have employed NN trained with GA to tackle the problem of detection of Parkinson's disease. The input weight vector to the input layer of NN has been optimized gradually to obtain the best possible weight vector. The results are compared with NN, MLP-FFN using different performance measurement parameters like accuracy, precision, recall and F-measure.

## II. PROPOSED METHODOLOGY

The versatility of hidden layer Neural Networks has been well studied [11, 12, 13]. Though, these studies do not look inside the problem of finding the optimal network to maximize the accuracy of the desired work. The performance of the NN highly depends on the initial weight vector that has been supplied to it at beginning of training phase. Generally, the weight vector is generated randomly keeping its component values within a given range. And different learning algorithms like variable metric, back-propagation, and gradient descent are used to gradually adjust the weights to converge the output to a better result. But the major drawback of using these local search methods is that there is a high possibility of getting trapped into local optima. As these methods tries to minimize one error estimating function gradually, the problem of getting trapped into local minima is obvious. To cope with the problem a global search strategy is required that would judge different possibilities over the search space instead of searching in the vicinity of last found best solution. In the present work we use Genetic Algorithm (GA) to overcome the problem.

GA was proposed by Holland (1975) to implement the theory of natural selection to solve optimization problems. The GA starts solving a problem by using a set of initial solutions. And it continuously applies crossover and mutations on the solutions to produce better offspring. The survival of any offspring depends on the fitness which is decided by the problem definition of the problem being solved. At each generation the best offspring survives and gradually produces better solutions in later generations until the solution of expected accuracy is reached. The problem of deciding the initial weight vector of the NN is an optimization problem. The GA can be successfully employed to solve the problem. The GA has lesser chance of getting trapped into local optima. Thus, it can be better choice than the traditional methods. The method of applying GA can be summarized as follows;

1. Generation of initial population: 'N' numbers of chromosomes are randomly generated. Each chromosome is actually an array of random real weight values, biologically genes; they vary in between '0' to '1'.
2. Calculating fitness values: A fitness function has to be defined, using it the fitness of each individual solution (chromosome) has to be evaluated. MSE of NN training is used as the fitness function.
3. Selection: The smaller the MSE, higher is the chance of getting selected for the next generation. $MSE_i$ denotes the fitness function value of $i^{th}$ solution. The selection procedure works as follows;

   3.1. $MSE_i$ is calculated for each solution in population.
   3.2. All $MSE_i$ are aggregated or averaged together to find $MSE_{ave}$
   3.3. A random value ($MSE_r$) is selected from predefined closed interval $[0, MSE_{ave}]$
   3.4. For all solutions $MSE_r - MSE_i$ is calculated and if the result of the subtraction is less than or equal to '0' the $i^{th}$ individual is selected.
   3.5. The process goes on until the number of solutions selected for next generation (mating pool) is equal to the number of solutions in the population initially.

4. Cross-over: The selected chromosomes take part in cross-over where the after selecting cross-over points on the chromosome the genes at the right of that point for both the chromosomes taking part get exchanged. And it creates two new individual.
5. Mutation: Genes of same chromosome take part in this phase. Genes from randomly selected position are swapped to create new individual solution.
6. Termination condition: Finally the termination condition is checked. In the present work number of generation has been selected as terminating condition. When the user given number of generation is reached the best possible individual is selected as the optimized weight vector, otherwise it starts from step 2 again.

The training phase using GA is depicted in Figure 1 while Figure 2 depicts the flow of GA.

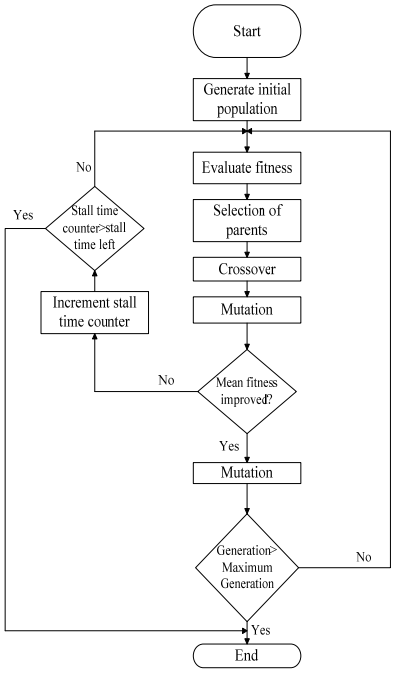


**Figure. 1.** The Genetic Algorithm which has been followed to optimize the input weight vector.

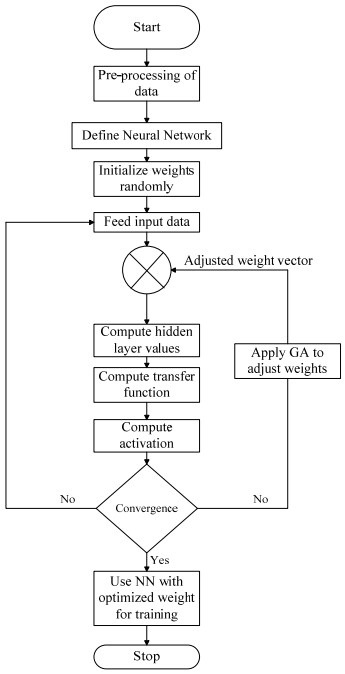


**Figure. 2.** Flowchart of NN training using Genetic algorithm depicted in Figure 3.

## III. EXPERIMENTAL METHODOLOGY

### A. *Preprocessing*

The following preprocessing is done on the dataset before the classification.

a) **Feature Extraction** - This step involves the task of extracting those attributes which are most important to classify the dataset correctly. For datasets having two or more classes, feature extraction is basically choosing the most effective features which can separate the classes accurately. Class separability depends not only on the class distributions but also on the classifier to be used. For example, the optimum feature set for a linear classifier may not be the optimum set for other classifiers for the same distributions. We assume that we seek the optimum feature set with reference to the Bayes classifier; this will result in the minimum error for the given distributions. Then, class separability becomes equivalent to the probability of misclassification due to the Bayes classifier. Therefore, theoretically, the Bayes error is the optimum measure of feature effectiveness. Also, in practice, the Bayes error calculated experimentally is one of the most popular criteria. That is, having selected a set of features intuitively from given data; estimate the Bayes error in the feature space. A major disadvantage of the Bayes error as a criterion is the fact that an explicit mathematical expression is not available except for a very few special cases. Even for normal distributions, the calculation of the Bayes error involves a numerical integration, except for the equal covariance case.

b) **Data Cleaning** - The data might contain missing values or noise. It is important to remove noise and fill up empty entries by suitable data by means of statistical analysis.

c) **Data Normalization** - the needs to be normalized before classification task is carried on to reduce distance between attribute values. It is generally achieved by keeping the value range in between -1 to +1.

2. After preprocessing the datasets are divided into two parts. One of which is used as training dataset and the other as testing dataset. In the present work two third (70%) of the data is used as training data and rest (30%) as testing data.

3. In the training phase the training dataset is supplied to different algorithms respectively to build the required classification model.

In the testing phase the classification models obtained from the training phase is employed to test the accuracy of the model.

After we obtain the experimental results it is time to find out the performance of the algorithms which are employed to perform the task. To measure the performance and to compare the performances we use several statistical performance measures like correlation coefficient, accuracy, Kappa statistic, Root mean squared error (RMSE), Mean absolute error, True positive rate (TP rate), and F-measure. The performance measuring parameters are defined as follows;

### B. *Root Mean Square Error (RMSE)*

RMSE [9] is a well-known performance measure which is calculated as the difference between the values anticipated by a classifier and the valuesactually discovered from the surroundings of the system being modeled. The RMSE of a classifier prediction withrespect to the computed variable $v_{c_k}$ is determined as thesquare root of the mean-squared error and is given by;

$$RMSE = \sqrt{\frac{\sum_{k=1}^{n}(v_{d_k} - v_{c_k})^2}{n}}$$

Where, $v_{d_k}$ denotes the originally observed value of $k^{th}$ object and $v_{c_k}$ denotes the predicted value by the classifier.

### C. *Confusion Matrix*

The confusion matrix [10] is a tabular representation that provides visualization of the performance of a classification algorithm. Each column of the matrix denotes the examples in a predicted class, while each row indicates the examples in an actual class. This helps to find out any type of misclassification due to the classifier. It provides more detailed analysis than classification accuracy. Classification accuracy is not a reliable metric for assessing the performance of a classifier as it may produce misleading results when the numbers of samples in different classes vary greatly. The confusion matrix entries can be defined as follows;

i. True positive (tp) is the number of 'positive' instances categorized as 'positive'.
ii. False positive (fp) is the number of 'negative' instances categorized as 'positive'.
iii. False negative (fn) is the number of 'positive' instances categorized as 'negative'.
iv. True negative (tn) is the number of 'negative' instances categorized as 'negative'.

TABLE I. Typical Example of Confusion Matrix of A Binary Classification Problem

| Actual Class \ Predicted Class | Positive | Negative |
|---|---|---|
| **Positive** | tp | fp |
| **Negative** | fn | tn |

As stated before that the accuracy may not be a good performance parameter hence couple of other parameters which are obtained from the confusion matrix, has been used and are found to be promising. They are as follows;

#### i. Accuracy

Accuracy is defined as a ratio of sum of the instances classified correctly to the total number of instances.

$$Accuracy = \frac{tp + tn}{tp + fp + fn + tn}$$

#### ii. Precision

Precision is defined as the ratio of correctly classified data in positive class to the total number of data classified as to be in positive class.

$$Precision = \frac{tp}{tp + fp}$$

#### iii. Recall

Recall or TP rate is defined as the ratio of $tp$ to the total number of instances classified under positive class.

$$Recall = \frac{tp}{tp + fn}$$

#### iv. F-measure

F-measure is defined as a combined representation of Precision and Recall and is defined as follows;

$$F\text{-}measure = 2 * \frac{Precision * Recall}{Precision + Recall}$$

## IV. Results & Discussion

Figure 3 depicts the confusion matrix of training, validation, testing, and overall phases for 'Parkinson's Disease' dataset. It is basically a binary classification problem. We apply MLP-FFN to the dataset, and it provides 96.4% accuracy for training phase, 93.1% accuracy for validation phase, 89.7% accuracy for testing phase and 94.9% accuracy overall. Figure 5 shows the error histogram for 20 bins. It reveals that the error involved during all the phases is almost equal to '0'. And it clearly suggests that for classifying such problem MLP-FFN would be a satisfactory choice. Figure 4 depicts a plot of cross-entropy vs. epochs for all three phases. Cross-entropy has been used as the performance estimator. Best performance is achieved at 34$^{th}$ epoch. Figure 6 shows the receiver operating characteristics (ROC) of the same for all three phases. Except test phase ROC all other phases have shown satisfactory performance.

Table II reports the confusion matrix of NN-GA model. Next in Table III a comparative study is reported. It reveals that the NN-GA model has outperformed the other models with greater extent.

Table II. Confusion Matrix of NN-GA Model

| Predicted \ Actual | Parkinson +ve | Parkinson -ve |
|---|---|---|
| Parkinson +ve | 44 | 2 |
| Parkinson -ve | 0 | 12 |

Table III. Comparison OF Different Models

| | NN | MLP-FFN | NN-GA |
|---|---|---|---|
| **Accuracy** | 91.38 | 94.9 | 96.55 |
| **Precision** | 73.33 | 96.6 | **100** |
| **Recall** | 91.67 | 96.6 | 85.71 |
| **F-Measure** | 81.48 | 96.6 | 92.31 |

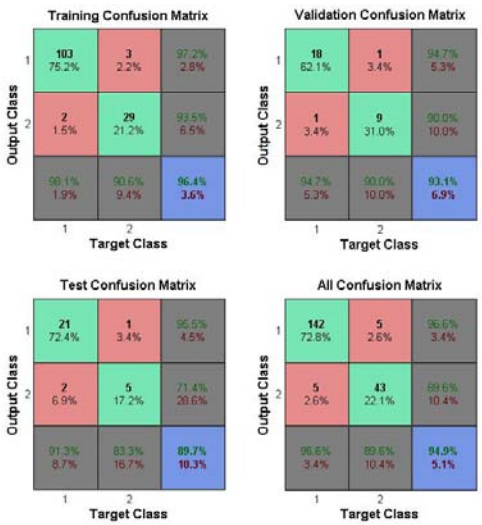


Figure. 3 Confusion matrix of MLP-FFN for Parkinson's dataset.

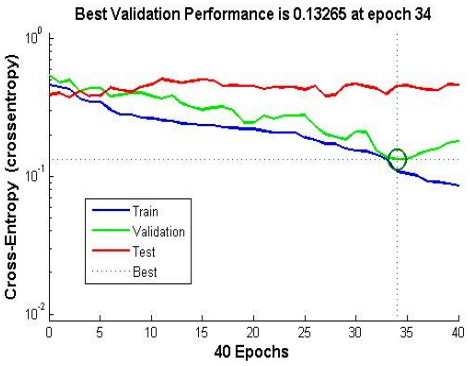


Figure. 4 Performance of training, testing and validation phases of MLP-FFN for Parkinson's dataset.

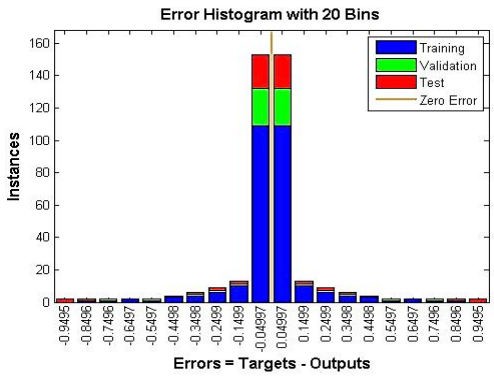


Figure. 5 Error histogram of MLP-FFN for Parkinson's dataset.

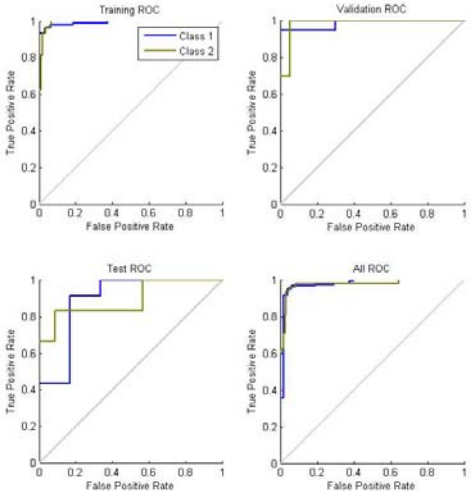


Figure. 6 Receiver Operating Characteristic of MLP-FFN for Parkinson's dataset.

## V. CONCLUSION

In this paper an efficient Parkinson's disease detection method is proposed. To overcome the premature convergence of local optimization methods in training phase of ANNs, Genetic algorithm is engaged to enhance its performance. The proposed model is compared with well-known MLP-FFN classifier. Experimental results suggest that NN-GA model is superior to other models under current study.

## VI. REFERENCES

[1] Max A. Little, Suitability of Dysphonia Measurements for Telemonitoring of Parkinson's Disease, IEEE Transactions on Biomedical Engineering, Vol. 56, No. 4, April 2009.

[2] A.Khemphila, V. Boonjing, Parkinsons Disease Classification using Neural Network and Feature selection, World Academy of Science, Engineering and Technology 64, 15-18, 2012.

[3] A. David Gil, B. M. Johnson, Diagnosing Parkinson by using Artificial Neural Networks and Support Vector Machines, Global Journal of Computer Science and Technology, pp. 63-71.

[4] M. F. Caglar, B. Cetisli, I. B. Toprak, Automatic Recognition of Parkinson's Disease from Sustained Phonation Tests Using ANN and Adaptive Neuro-Fuzzy Classifier, Journal of Engineering Science and Design, Vol.1 No.2 pp.59-64, 2010.

[5] Y. Delican, L. Ozilmaz, T. Yildirim, Evolutionary Algorithms Based RBF Neural Networks For Parkinson s Disease Diagnosis, ELECO 2011 7th International Conference on Electrical and Electronics Engineering, 290-294, December 2011, Bursa, Turkey.

[6] M. Ene, Neural network-based approach to discriminate healthy people from those with Parkinson's disease, Vol. 35, 112-116, ISSN: 1223-6934, January 2008.

[7] R. Das, A comparison of multiple classification methods for diagnosis of Parkinson disease, Expert Systems with Applications, 2009.

[8] Tolosa, Eduardo, Gregor Wenning, and Werner Poewe. "The diagnosis of Parkinson's disease." The Lancet Neurology 5, no. 1 (2006): 75-86.

[9] Armstrong JS, Collopy F. Error measures for generalizing about forecasting methods: empirical comparisons. Int. J Forecast, 8:69–80, 1992.

[10] Stehman SV. Selecting and interpreting measures of thematic classification accuracy. Remote SensEnviron,62(1):77–89, 1997.

[11] Hornik K., Multilayer Feedforward Networks are universal Approximators, Neural Networks, Vol. 2, pp.359-366, 1989.

[12] Funahashi K.I., On the Approximate Relation of Continuous Mappings by Neural Networks, Neural Networks, Vol. 2, pp. 183-192, 1989.

[13] White H., Connectionist Nonparametric regression: multilayer feedforward network can learn arbitrary mappings, Neural Networks, Vol. 3, pp. 535-549, 1990.

[14] Chatterjee, S., Chakraborty, R., Dey, N., & Hore, S. (2015). A quality prediction method for weight lifting activity.

[15] Chatterjee, S., Sarkar, S., Hore, S., Dey, N., Ashour, A. S., & Balas, V. E. (2016). Particle swarm optimization trained neural network for structural failure prediction of multistoried RC buildings. Neural Computing and Applications, 1-12.

[16] Chatterjee, S., Ghosh, S., Dawn, S., Hore, S., & Dey, N. (2016). Forest Type Classification: A Hybrid NN-GA Model Based Approach. In Information Systems Design and Intelligent Applications (pp. 227-236). Springer India.

[17] Hore, S., Chatterjee, S., Sarkar, S., Dey, N., Ashour, A. S., Balas-Timar, D., & Balas, V. E. (2016). Neural-based prediction of structural failure of multistoried RC buildings. Structural Engineering and Mechanics, 58(3), 459-473.

[18] Chakraborty, R., Paladhi, S., Chatterjee, S., & Banerjee, S. (2014). An Optimized Algorithm for Solving Combinatorial Problems using Reference Graph. IOSR Journal of Computer Engineering, 16(3), 1-7.

[19] Chatterjee, S., Paladhi, S., Hore, S., & Dey, N. (2015, March). Counting all possible simple paths using Artificial Cell division mechanism for directed acyclic graphs. In Computing for Sustainable Global Development (INDIACom), 2015 2nd International Conference on (pp. 1874-1879). IEEE.

[20] Chatterjee, S., & Banerjee, D. (2014). A Novel Boolean Expression based Algorithm to find all possible Simple Paths between two nodes of a Graph. International Journal of Advanced Research in Computer Science, 5(7).

[21] Chatterjee, S., Paladhi, S., & Chakraborty, R. A Comparative Study On The Performance CharacteristicsOf Sudoku Solving Algorithms.