Diagnosing Parkinson’s Diseases Using Fuzzy Neural System

Rahib H. Abiyev1 and Sanan Abizade2

1Department of Computer Engineering, Applied Artificial Intelligence Research Centre, Near East University, Lefkosa, Northern Cyprus, Mersin 10, Turkey
2Department of Electrical and Electronic Engineering, Applied Artificial Intelligence Research Centre, Near East University, Lefkosa, Northern Cyprus, Mersin 10, Turkey

Correspondence should be addressed to Rahib H. Abiyev; rahib.abiyev@neu.edu.tr

Received 30 September 2015; Accepted 14 December 2015

Academic Editor: Reza Khosrowabadi

Copyright © 2016 R. H. Abiyev and S. Abizade. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This study presents the design of the recognition system that will discriminate between healthy people and people with Parkinson’s disease. A diagnosing of Parkinson’s diseases is performed using fusion of the fuzzy system and neural networks. The structure and learning algorithms of the proposed fuzzy neural system (FNS) are presented. The approach described in this paper allows enhancing the capability of the designed system and efficiently distinguishing healthy individuals. It was proved through simulation of the system that has been performed using data obtained from UCI machine learning repository. A comparative study was carried out and the simulation results demonstrated that the proposed fuzzy neural system improves the recognition rate of the designed system.

1. Introduction

In the world, many people suffer from Parkinson’s disease (PD). The disease more often appeared after the age of 60 [1]. Parkinson’s disease is a chronic disorder of central nervous system which causes the death of the nervous cell in the brain. Parkinson’s disease is progressive and the number of people suffering from the disease is expected to rise. The disease usually happens slowly and persists over a long period of time.

The symptoms of the PD continue and worsen over time. The basic symptoms of PD are movement related symptoms. These are tremor, rigidity or stiffness of the limbs and trunk, bradykinesia or slow movement, and problems with balance or walking [2, 3]. Tremor is a basic symptom which may affect shaking or trembling of legs, arms, hands, jaw, or face. The patients may have difficulty talking, walking, or completing some other simple tasks as these symptoms become more pronounced. Other symptoms are related to the behavioural problems, depression, thinking, sleep, and emotional problems. A person with Parkinson’s may have a trouble in speaking and swallowing and chewing problems. Especially in advanced stages of the disease nonmotor features, such as dementia and dysautonomia, occur frequently.

The diagnosis and timely treatments are important in order to manage its symptoms. The diagnosis is based on neurological examination and medical history of patients. The diagnosis of the disease in the early stages is difficult [3]. Diagnosis of PD depends on the presence of two or more of the above symptoms.

Vocal symptoms that include impairment of vocal sounds (dysphonia) and problems with the normal articulation of speech (dysarthria) are important in diagnosis of PD [4]. The research paper [5] shows that the most important symptom of PD is dysphonia. The dysphonia is the disorder of voice. Dysphonic symptoms typically include reduced loudness and roughness and breathiness and decreased energy in the higher parts of the harmonic spectrum and exaggerated vocal tremor. The treatment of these symptoms is difficult for the people having Parkinson’s disease. In [4–6] it was shown that approximately 90% of people with Parkinson’s disease have dysphonia. Dysphonia includes any pathological or functional problem with voice [6]. The voice will sound hoarse, strained, or effortful. It may be difficult to understand the voice of people having PD. The used method for diagnosis of Parkinson’s disease (PD) is basically based on speech measurement for general voice disorders [4, 7–9].
Specialists doctors need to make an analysis of many factors for accurate diagnose of PD. Usually, decisions made are based on evaluating the current test results of patients. The problem becomes too difficult if the number of attributes that the specialist wants to evaluate is high. Recently various computational tools have been developed in order to improve the accuracy of diagnosis of PD. These tools have provided excellent help to doctors and medical specialists in making decisions about the patients. Different Artificial Intelligence (AI) techniques, expert systems, and decision making systems are designed for diagnosis or classification of diseases. They were potential and good supportive tools for the expert/doctor. The development of efficient recognition systems in medical diagnosis is becoming more important. Nowadays various Artificial Intelligence techniques such as expert systems, fuzzy systems, and neural networks are actively applied for diagnosis of Parkinson’s diseases using voice signals. Reference [4] introduces a new measure of dysphonia, pitch period entropy (PPE), which is robust to many uncontrollable confounding effects including noisy acoustic environments and separates healthy people from the people having PD. Nonlinear dynamical systems theory [4, 10] and statistical learning theory, such as linear discriminant analysis (LDA) and support vector machines (SVMs) [5, 11], are preferred for classification of healthy people or those with PD and discriminate the healthy people on the basis of measures of dysphonia. Different techniques, such as SVM [12], SVM with RBF (radial basis function) kernel [13], SVM with Multiple Layer Perceptron (MLP), and a Radial Basis Function Network (RBFN) [14], are used for diagnosis of PD. In [15] integration of Kohonen self-organizing map (KSOM) and least squares support vector machine (LS-SVM), and in [3, 16] nonlinear time series analysis tools are applied for diagnosing of PD. Reference [17] uses fuzzy c-means algorithm, [18] uses four independent classification schemas, neural networks, DMneural, regression, and decision tree for classification purpose, and a comparative study was carried out.

The above methods are used in order to increase classification accuracy of PD. Classification systems can help in increasing the accuracy and reliability of diagnoses and minimizing possible errors, as well as making the diagnoses more time efficient. Success in the discovery of knowledge depends on the ability to explore different classes of specific data and to apply appropriate methods in order to extract the main features. This paper deals with the application of fusion of fuzzy systems and neural networks for designing of the recognition system of PD.

Fuzzy systems can handle uncertainties associated with information or data in the knowledge bases [19] and are widely used to solve different real world problems. Fuzzy system uses data and knowledge specific to chaotic dynamics of the process and increases the performance of the system. In the literature, different neural and fuzzy structures are proposed for solving various problems [20–26]. In [22, 23] clustering algorithm and gradient descent algorithm are applied for the design of multi-input and single output FNS. Well known ANFIS (adaptive neurofuzzy inference system) structure is used for solving cervical cancer recognition [27], for optimizing the chiller loading [28], and for distinguishing ESES (electrical status epilepticus) and normal EEG (electroencephalography) signals [29]. The use of multiple ANFIS structures, in [27], leads to the increase of the number of parameters of the network. In these papers the used systems are designed for special purpose and most of them are basically based on Mamdani type rules. Performances of these systems are determined by measuring classification rate. In this paper, in order to improve the performance of classification system, a multi-input and multioutput fuzzy neural system (FNS) based on TSK rules is proposed for identification of the PD.

The paper is organized as follows. Section 2 describes the structure of proposed fuzzy neural system used for recognition of PD. The parameter update rule of the proposed system is presented in Section 3. Section 4 describes the simulation results. The conclusions are given in Section 5.

2. FNS Based Recognition

The fuzzy neural system combines the learning capabilities of neural networks with the linguistic rule interpretation of fuzzy inference systems. The design of FNS includes the development of the fuzzy rules that have if-then form. This can be achieved by dint of optimal definition of the premise and consequent parts of fuzzy if-then rules for the classification system through the training capability of neural networks. The two basic types of if-then rules used in fuzzy systems are Mamdani and Takagi-Sugeno-Kang (TSK) type fuzzy rules. The first type consists of rules, whose antecedent and consequent parts utilize fuzzy values. The second one uses the fuzzy rules that have fuzzy antecedent and crisp consequent parts. In the paper we use TSK type fuzzy rules for system design. The second type of fuzzy system approximates nonlinear system with linear systems and has the following form:

If \( x_1 \) is \( A_{1j} \) and \( x_2 \) is \( A_{2j} \) and ... and \( x_m \) is \( A_{mj} \)

Then \( y_j \) is \( \sum_{i=1}^{m} a_{ij} x_i + b_j \),

(1)

where \( x_i \) and \( y_j \) are input and output signals of the system, respectively, \( i = 1, \ldots, m \) is the number of input signals, and \( j = 1, \ldots, r \) is the number of rules. \( A_{ij} \) are input fuzzy sets; \( b_j \) and \( a_{ij} \) are coefficients.

The structure of fuzzy neural networks used for the classification of PDs is based on TSK type fuzzy rules and is given in Figure 1. The FNS includes six layers. In the first layer, \( x_i (i = 1, \ldots, m) \) input signals are distributed. The second layer includes membership functions. Here each node corresponds to one linguistic term. Here, for each input signal entering the system, the membership degree to which input value belongs to a fuzzy set is calculated. The Gaussian membership function is used in order to describe linguistic terms:

\[
\mu_{1j}(x_i) = e^{-(x_i-c_{ij})^2/\sigma_{ij}^2}, \quad i = 1, \ldots, m, \quad j = 1, \ldots, r,
\]

(2)
where $c_{ij}$ and $\sigma_{ij}$ are center and width of the Gaussian membership functions, respectively, and $\mu_{1j}(x_i)$ is membership function of $i$th input variable for $j$th term.

The third layer is a rule layer. Here number of nodes is equal to the number of rules. Here $R_1, R_2, \ldots, R_r$ represents the rules. The output signals of this layer are calculated using t-norm min (AND) operation:

$$\mu_j(x) = \prod_{i=1}^{m} \mu_{1j}(x_i) \quad i = 1, \ldots, m, \quad j = 1, \ldots, r,$$

where $\Pi$ is the min operation.

These $\mu_j(x)$ signals are input signals for the fifth layer. Fourth layer is a consequent layer. It includes $n$ linear systems. Here the output values of the rules are determined using linear functions (LF):

$$y1_j = \sum_{i=1}^{m} x_i w_{ij} + b_j,$$  \hspace{0.5cm} (4)

In the fifth layer, the output signals of the third layer are multiplied by the output signals of the fourth layer. The output of $j$th node is calculated as $y_j = \mu_j(x) \cdot y1_j$.

The output signals of FNS are determined as

$$u_k = \frac{\sum_{j=1}^{r} w_{jk} y_j}{\sum_{j=1}^{r} \mu_j(x)}.$$

Here $u_k$ are the output signals of FNS ($k = 1, \ldots, n$) and $w_{jk}$ are weight coefficients of connections used between layers 5 and 6. After calculating the output signal, the training of the network starts.

### 3. Parameter Updates

#### 3.1. Fuzzy Classification

The design of FNS (Figure 1) includes determination of the unknown parameters of the antecedent and the consequent parts of the fuzzy if-then rules (1). In fuzzy rules the antecedent part represents the input space by dividing the space into a set of fuzzy regions and the consequent part describes the system behaviour in those regions.

As mentioned above, recently a number of different approaches have been used for designing fuzzy if-then rules. Some of them are based on clustering [20–24, 26], the least squares method (LSM) [20, 22, 30], gradient algorithms [14, 20–23, 26], genetic algorithms [24, 25, 28], and particle swarm optimization (PSO) [31].

In this paper, fuzzy clustering and gradient technique are used for the design of FNS. At first the fuzzy clustering is used to design the antecedent (premise) parts, and then gradient algorithm is used to design the consequent parts of the fuzzy rules. Fuzzy clustering is an efficient technique for constructing the antecedent structures. The aim of clustering methods is to identify a certain group of data from a large data set, such that a concise representation of the behaviour of the system is produced. Each cluster center can be translated into a fuzzy rule for identifying the class. Different clustering algorithms are developed [32–34]. Recently fuzzy c-means [32] and subtractive clustering [33, 34] algorithms have been developed for fuzzy systems. Subtractive is unsupervised clustering [33] which is an extension of the grid based mountain clustering [34]. Here the number of clusters for input data points is determined by the clustering algorithm. Sometimes we need to control the number of clusters in an input space. In these cases, the supervised clustering
algorithms are of primary concern. Fuzzy c-means clustering is one of them. It can efficiently be used for fuzzy systems [32] with a simple structure and sufficient accuracy. In this paper, the fuzzy c-means (FCM) clustering technique is used for structuring the premise part of the fuzzy system.

Learning of FNS starts with the update of parameters of antecedent part of if-then rules, that is, the parameters of the second layer of FNS. For this aim FCM classification of antecedent part of if-then rules, that is, the parameters for structuring the premise part of the fuzzy system.

In this paper, the fuzzy c-means (FCM) clustering technique is used [32] with a simple structure and sufficient accuracy. In this paper, we applied gradient learning with adaptive learning rate is applied for the update of parameters. The adaptive learning rate guarantees the convergence and speeds up the learning of the network. In addition, the momentum is used to speed up the learning processes.

The error on the output of the network is calculated as

\[ E = \frac{1}{2} \sum_{k=1}^{n} (u_k^d - u_k) ^2. \]  

Here \( n \) is the number of output signals of the network, \( u_k^d \) and \( u_k \) are desired and current output values of the network \((k = 1, \ldots, n)\), respectively. The parameters \( w_{jk}, a_{ij}, b_j \) \((i = 1, \ldots, m, j = 1, \ldots, r, k = 1, \ldots, n)\) in consequent part of network and the parameters of membership functions \( c_{ij} \) and \( a_{ij} \) \((i = 1, \ldots, m, j = 1, \ldots, r)\) in the premise part of FNS are adjusted using the following formulas:

\[ w_{jk} (t + 1) = w_{jk} (t) - \gamma \frac{\partial E}{\partial w_{jk}} + \lambda (w_{jk} (t) - w_{jk} (t - 1)); \]

\[ a_{ij} (t + 1) = a_{ij} (t) - \gamma \frac{\partial E}{\partial a_{ij}} + \lambda (a_{ij} (t) - a_{ij} (t - 1)); \]

\[ b_j (t + 1) = b_j (t) - \gamma \frac{\partial E}{\partial b_j} + \lambda (b_j (t) - b_j (t - 1)); \]

\[ c_{ij} (t + 1) = c_{ij} (t) - \gamma \frac{\partial E}{\partial c_{ij}} + \lambda (c_{ij} (t) - c_{ij} (t - 1)); \]

\[ \sigma_{ij} (t + 1) = \sigma_{ij} (t) - \gamma \frac{\partial E}{\partial \sigma_{ij}} + \lambda (\sigma_{ij} (t) - \sigma_{ij} (t - 1)); \]

Here \( \gamma \) is the learning rate, \( \lambda \) is the momentum, \( m \) is the number of input signals of the network (input neurons) and \( r \) is the number of fuzzy rules (hidden neurons), and \( n \) is the number of output neurons.

The derivatives in (10) are computed using the following formulas:

\[ \frac{\partial E}{\partial w_{jk}} = \frac{\partial E}{\partial u_k} \frac{\partial u_k}{\partial w_{jk}} = \left( w_k (t) - u_k^d (t) \right) \cdot y_{ij} \sum_{j=1}^{l} h_j, \]

\[ \frac{\partial E}{\partial a_{ij}} = \frac{\partial E}{\partial u_k} \frac{\partial u_k}{\partial a_{ij}} = \sum_{j=1}^{l} \left( u_k (t) - u_k^d (t) \right) \cdot w_{kj} y_{ij} h_j, \]
The FNS, described above, is applied for classification of Parkinson’s dieses. The people are divided into two classes: normal and PD. For this aim, the database is taken from University of California at Irvine (UCI) machine learning repository. The data set is donated from hospitals and it has been studied by many researchers. The data set includes biomedical voice measurements of 31 people; 23 were diagnosed with PD. Each row contains the value of the 23 voice parameters. Each column contains 195 items of data for each parameter. The main aim of the data is to discriminate healthy people from those with PD. The parameters that are used for recognition of PD are given in Table 1. These are the parameters of the voice signals recorded directly on the computer using Computerized Speech Laboratory. During modelling the preprocessing have been done on the input data and the input data are normalized in the interval of [0, 1]. The scaling operation helps and makes the training process of the system easy. After normalization, these data are entered as an input signal to the FNS.

To design classification model the FNS structure with 23 input and 2 output neurons is generated first. If we use traditional neurofuzzy structure (e.g., [20] or [26]) for 23 inputs and 2 cluster centers, pow(2,23) = 8383608 rules should be generated. The rules are constructed using all possible combinations of inputs and cluster centers. This is very large number. In this paper the number of rules is selected according to the clustering results, equal to cluster centers.

In the design of FNS, the fuzzy classification is applied in order to partition input space and select the parameters of the premise parts, that is, the parameters of Gaussian membership functions used in the second layer of FNS. FCM clustering is used for the input space with 16 clusters for each input. 16 fuzzy rules are constructed using a different combination of these clusters for 22 inputs. After clustering input space gradient decent algorithm is used for learning of consequent parts of the fuzzy rules, that is, parameters of the 4th layer of FNS. Learning is implemented using cross validation. Cross validation generalizes two independent data sets: training and testing. It is applied to find accurate model of classifier. In the paper 10-fold cross validation is used for separation of the data into training and testing set and for evaluation of classification accuracy. There should be set of experiments in order to achieve required accuracy in the FNS output. The simulation is performed using different number of neurons in hidden layer. The design steps of FNS for diagnosing PD are given below:

(1) Read PD data set. Select input and output (target) signals from statistical data. Apply normalization.

(2) Enter the values of learning rate and momentum. Set the number of clusters. Generate network parameters. Set a maximal number of epochs for learning.

(3) Apply classification algorithm to the input signals and determine the cluster centers.

(4) Use cluster centers to determine the centers of membership functions of layer 2.

(5) Use the centers of membership functions to determine the widths of membership functions.

\[
\frac{\partial E}{\partial b_j} = \frac{\partial E}{\partial u_k} \frac{\partial y_1}{\partial y_j} \frac{\partial y_j}{\partial b_j} = \sum_k \left( u_k(t) - u_k^d(t) \right) w_k \mu_j, \quad \frac{\partial E}{\partial c_{ij}} = \sum_k \frac{\partial E}{\partial u_k} \frac{\partial y_1}{\partial u_k} \frac{\partial u_k}{\partial c_{ij}}, \quad \frac{\partial E}{\partial \sigma_{ij}} = \sum_k \frac{\partial E}{\partial u_k} \frac{\partial y_1}{\partial u_k} \frac{\partial u_k}{\partial \sigma_{ij}}.
\]

Here \(i = 1, \ldots, m, j = 1, \ldots, r, k = 1, \ldots, n\).

The derivatives in (12) are determined by the following formulas:

\[
\frac{\partial E}{\partial c_{ij}} = \sum_k \frac{\partial E}{\partial u_k} \frac{\partial y_1}{\partial u_k} \frac{\partial u_k}{\partial c_{ij}}, \quad \frac{\partial E}{\partial \sigma_{ij}} = \sum_k \frac{\partial E}{\partial u_k} \frac{\partial y_1}{\partial u_k} \frac{\partial u_k}{\partial \sigma_{ij}}.
\]

Here \(i = 1, \ldots, m, j = 1, \ldots, r, k = 1, \ldots, n\). Consider

\[
\frac{\partial E}{\partial u_k} = u_k(t) - u_k^d(t); \quad \frac{\partial u_k}{\partial c_{ij}} = \frac{y_j - u_k}{\sum_{j=1}^n H_j};
\]

\[
\frac{\partial y_j(x_i)}{\partial c_{ij}} = \frac{\mu_j(x_i)}{\sigma_{ij}^2} \frac{2 \left( x_i - \epsilon_j \right)}{\sigma_{ij}^2}; \quad \frac{\partial y_j(x_i)}{\partial \sigma_{ij}} = \mu_j(x_i) \frac{2 \left( x_i - \epsilon_j \right)^2}{\sigma_{ij}^4}.
\]

4. Simulation Studies

The FNS, described above, is applied for classification of Parkinson's diseases. The people are divided into two classes:
| Name                | ASCII subject name and recording number                              |
|---------------------|-----------------------------------------------------------------------|
| MDVP:Fo (Hz)        | Average vocal fundamental frequency                                   |
| MDVP:Fhi (Hz)       | Maximum vocal fundamental frequency                                   |
| MDVP:Flo (Hz)       | Minimum vocal fundamental frequency                                   |
| MDVP:Jitter (%)     | Five measures of variation in fundamental frequency                  |
| MDVP:Jitter (Abs)   |                                                                         |
| MDVP:RAP            |                                                                         |
| MDVP:PPQ            |                                                                         |
| Jitter:DDP          |                                                                         |
| MDVP:Shimmer        | Six measures of variation in amplitude                                |
| MDVP:Shimmer (dB)   |                                                                         |
| Shimmer:APQ3        | Two measures of ratio of noise to tonal components in the voice       |
| Shimmer:APQ5        |                                                                         |
| MDVP:APQ            | Two nonlinear dynamical complexity measures                           |
| Shimmer:DDA         |                                                                         |
| NHR                 | Three nonlinear measures of fundamental frequency variation          |
| HNR                 |                                                                         |
| RPDE                |                                                                         |
| D2                  |                                                                         |
| DFA                 |                                                                         |
| Spread1             |                                                                         |
| Spread2             |                                                                         |
| PPE                 | Health status of the subject: one, Parkinson’s; zero, healthy         |

1. Using input statistical data define a random partition for 10-fold cross validation.
2. Initialize current number of learning epochs to 1.
3. Use PD data set and cross validation and determine training and testing data sets.
4. Determine the numbers of rows in training and testing data sets.
5. Initialize the number of iterations to 1.
6. According to the number of iterations select input data from training data set and send them to the input of FNS.
7. Calculate network outputs.
8. Determine the values of errors using network output and target output signals. Use these error values to compute the sum of the squared errors (SSE).
9. Using error values update the network parameters (learning of network).
10. Apply adaptive strategy for updating the learning rate using current and previous values of SSE.
11. Compute sum of SSE obtained on each iteration and save as the training error. Repeat Steps (11)–(16) for other remaining training data sets. If the current number of iterations will be less than a number of rows in the training set then go to Step (11), otherwise go to Step (17).
12. Select test data set.
13. Set number of iterations to 1.
14. According to the number of iterations select input data from test data set and send them to the input of FNS.
15. Compute the output of FNS.
16. Determine the values of errors using network output and target output signals. Compute SSE on the output of the network.
17. Compute sum of SSE obtained on each iteration of the loop and save as the testing error. Repeat Steps (19)–(22) for other remaining test data sets.
18. Check the value of testing error with the value of testing error obtained in the previous epoch. If the current error value is less than the previous one then go to Step (24), otherwise go to Step (25).
19. Save the parameters of the network. Save the values of training and testing errors.
20. Use the sum of SSE to find root mean squared error (RMSE). Print the values of testing and training errors; increment the epochs number.
21. Check a current number of epochs for the continuation of the learning process. If this number is less than the maximal number of epochs then repeat Steps (8)–(26). Otherwise go to Step (27).
22. Print the values of training and testing errors obtained in Step (24).
23. Stop the training.
The training of input/output data for the classification system will be a structure whose first component is the twenty-three-dimension input vector and second component is the two-dimension output clusters. Table 2 depicts the fragment from PD data set. The FNS structure is generated with 23 input and two output neurons. After generation fuzzy c-means clustering and gradient descent algorithms are applied for training the parameters of FNS. In the first step, using fuzzy clustering, cluster centers are determined using the input data. These cluster centers are used to organize the membership functions of the inputs of antecedent part of each fuzzy rules. The rule layer is the second layer. The consequent parts of the fuzzy rules are organized using linear functions. Linear functions are determined in fourth layer. After generation of the consequent part of fuzzy rules are organized using linear functions. The structures obtained from classification operation will be the centers of Gaussian membership functions used in antecedent parts of fuzzy rules. The consequent parts are constructed on the basis of learning of the parameters of linear functions.

The simulation results of FNS is compared with the simulation results of other models used for classification of PD. For evaluation of the outcomes of the models the Root Mean Square Error (RMSE) is used:

\[
RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (u_i^d - u_i)^2}.
\]  

Here \(u_i^d\) are desired values of output and \(u_i\) are actual values of the system output.

To estimate the performance of the FNS clustering systems, the recognition rates and RMSE values of errors between clusters and current output signal are taken. RMSE is computed using formula given above. Recognition rate is computed by the number of items correctly classified divided by the total number of items:

\[
\text{Recognition rate} = \frac{\text{Number of items correctly classified}}{\text{Total number of items}} \times 100\%.
\]  

During training of FNS, all input data are scaled to interval [0, 1]. Then fuzzy c-means clustering is applied to input data. The result of clustering is used to set up parameters of the antecedent part of fuzzy rules, that is, parameters of the second layer of FNS structure. The parameters of the consequent part of fuzzy rules are determined by applying gradient learning. The learning has been performed for 2000 epochs. The synthesis of FNS classification system is performed using different number of fuzzy rules. The training has been performed using different number of rules: 2, 5, 8, and 16. Training is performed using 10-fold cross validation. In the results of training the parameters of FNS are determined. Figure 2 depicts the values of RMSE obtained during training. Once the FNS is trained then it has been used for testing. The values of RMSE obtained for train, evaluation, and test stages for FNS having 16 hidden neurons are 0.232154, 0.291636, and 0.283590, correspondingly. The training has been performed with the learning rate 0.01 and momentum rate 0.825. Table 3 describes training and testing results of FNS model obtained using different number of
Table 3: Simulation results of FNS.

| Number of hidden neurons | RMSE training | RMSE evaluation | RMSE testing | Accuracy (%) |
|--------------------------|---------------|-----------------|--------------|--------------|
| 2                        | 0.548520      | 0.560548        | 0.551954     | 81.025641    |
| 5                        | 0.397395      | 0.401047        | 0.379963     | 93.333333    |
| 8                        | 0.341242      | 0.434860        | 0.35679     | 95.897436    |
| 12                       | 0.333357      | 0.343488        | 0.335679     | 97.948718    |
| 16                       | 0.232154      | 0.291636        | 0.283590     | 100          |

From Table 3, it was shown that the increase in the number of rules (or the number of hidden neurons) decreases the values of RMSE for training and testing cases and increases recognition rate. The use of clustering and gradient techniques for learning allows quick obtaining of low RMSE value and allows improving performance of FNS for training and testing stages. In the second simulation a comparative analysis of the classification of PD has been performed. The result of the simulation of the FNS classification model is compared with results of simulations of different classification models, such as support vector machine (SVM), neural networks (NN), regression model, decision tree, and FCM based feature weighting. To estimate the performance of the NN, SVM, and FNS clustering systems, the recognition rates and RMSE values of errors between clusters and current output signal are compared. In Table 4, the comparative results of simulations of different models are given. As shown in the table the performance of FNS classification system is better than the performance of the other models.

5. Conclusion

The paper presents the diagnosis of Parkinson’s diseases using fuzzy neural structures. The structure and learning algorithms of FNS are presented. Fuzzy clustering and gradient descent learning algorithms are applied for the development of the FNS. Learning is performed using 10-fold cross validation data set. The design of the classification system is carried out using different number of fuzzy rules used in FNS. Recognition rate of classification is obtained as 100% with 16 hidden neurons. For comparative analysis, the simulation of PD is performed using different models. The obtained results demonstrate that the performance of FNS is better than the other models used for classification of PD.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] R. Betarbet, T. B. Sherer, and J. T. Greenamyre, “Animal models of Parkinson’s disease,” *BioEssays*, vol. 24, no. 4, pp. 308–318, 2002.
[2] A. J. Hughes, S. E. Daniel, L. Kilford, and A. J. Lees, “Accuracy of clinical diagnosis of idiopathic Parkinson’s disease: a clinicopathological study of 100 cases,” *Journal of Neurology Neurosurgery and Psychiatry*, vol. 55, no. 3, pp. 181–184, 1992.
[3] M. A. Little, P. E. McSharry, S. J. Roberts, D. A. E. Costello, and I. M. Moroz, “Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection,” *BioMedical Engineering OnLine*, vol. 6, article 23, 2007.
[4] M. A. Little, P. E. McSharry, E. J. Hunter, J. Spielman, and L. O. Ramig, “Suitability of dysphonia measurements for telemonitoring of Parkinson’s disease,” *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 4, pp. 1015–1022, 2009.
[5] N. Singh, V. Pillay, and Y. E. Choonaara, “Advances in the treatment of Parkinson’s disease,” *Progress in Neurobiology*, vol. 81, no. 1, pp. 29–44, 2007.
[6] A. K. Ho, R. Jansék, C. Maričić, J. L. Bradshaw, and S. Gates, “Speech impairment in a large sample of patients with...
Parkinson's disease," *Behavioural Neuroscience*, vol. 11, no. 3, pp. 131–137, 1998.

[7] J. H. L. Hansen, L. Gavidia-Ceballos, and J. F. Kaiser, "A nonlinear operator-based speech feature analysis method with application to vocal fold pathology assessment," *IEEE Transactions on Biomedical Engineering*, vol. 45, no. 3, pp. 300–313, 1998.

[8] B. Boyanov and S. Hadjitodorov, "Acoustic analysis of pathological voices," *IEEE Engineering in Medicine and Biology Magazine*, vol. 16, pp. 4–82, 1997.

[9] S. Hadjitodorov, B. Boyanov, and B. Teston, "Laryngeal pathology detection by means of class-specific neural maps," *IEEE Transactions on Information Technology in Biomedicine*, vol. 4, no. 1, pp. 68–73, 2000.

[10] H. Kanz and T. Schreiber, *Nonlinear Time Series Analysis*, Cambridge University Press, Cambridge, UK, 1999.

[11] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction: With 200 Full-Color Illustrations*, Springer, New York, NY, USA, 2001.

[12] T. Khan, J. Westin, and M. Dougherty, "Classification of speech intelligibility in Parkinson's disease," *Biosystems and Biomedical Engineering*, vol. 34, no. 1, pp. 35–45, 2014.

[13] R. Prashanth, S. Dutta Roy, P. K. Mandal, and S. Ghosh, "Automatic classification and prediction models for early Parkinson’s disease diagnosis from SPECT imaging," *Expert Systems with Applications*, vol. 41, no. 7, pp. 3333–3342, 2014.

[14] S. Pan, S. Iplikci, K. Warwick, and T. Z. Aziz, "Parkinson's Disease tremor classification—a comparison with Support Vector Machines and neural networks," *Expert Systems with Applications*, vol. 39, no. 12, pp. 10764–10771, 2012.

[15] G. Singh and L. Samavedham, "Unsupervised learning based feature extraction for differential diagnosis of neurodegenerative diseases: a case study on early-stage diagnosis of Parkinson disease," *Journal of Neuroscience Methods*, vol. 256, pp. 30–40, 2015.

[16] J. Zhang, X. Luo, and M. Small, "Detecting chaos in pseudoperiodic time series without embedding," *Physical Review E*, vol. 73, no. 1, Article ID 016216, 2006.

[17] K. Polat, "Classification of Parkinson’s disease using feature weighting method on the basis of fuzzy C-means clustering," *International Journal of Systems Science*, vol. 43, no. 4, pp. 597–609, 2012.

[18] R. Das, "A comparison of multiple classification methods for diagnosis of Parkinson disease," *Expert Systems with Applications*, vol. 37, no. 2, pp. 1568–1572, 2010.

[19] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning—I," *Information Sciences*, vol. 8, no. 3, pp. 199–249, 1975.

[20] J.-S. R. Jang, C.-T. Sun, and E. Mizutani, *Neuro-Fuzzy And Soft Computing*, chapter 7, Prentice-Hall, New Jersey, NJ, USA, 1997.

[21] R. H. Abiyev and O. Kaynak, "Fuzzy wavelet neural networks for identification and control of dynamic plants—a novel structure and a comparative study," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 8, pp. 3133–3140, 2008.

[22] R. H. Abiyev, "Fuzzy wavelet neural network based on fuzzy clustering and gradient techniques for time series prediction," *Neural Computing & Applications*, vol. 20, no. 2, pp. 249–259, 2011.