This article appeared in a journal published by Elsevier. The attached copy is furnished to the author for internal non-commercial research and education use, including for instruction at the authors institution and sharing with colleagues.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier’s archiving and manuscript policies are encouraged to visit:

http://www.elsevier.com/authorsrights
Effective detection of Parkinson's disease using an adaptive fuzzy $k$-nearest neighbor approach

Wan-Li Zuo a,b, Zhi-Yan Wang a,b, Tong Liu a,b, Hui-Ling Chen c,*

a College of Computer Science and Technology, Jilin University, Changchun 130012, China
b Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China
c College of Physics and Electronic Information, Wenzhou University, Wenzhou, Zhejiang 3250035, China

ARTICLE INFO

Article history:
Received 4 October 2012
Received in revised form 3 January 2013
Accepted 19 February 2013
Available online 19 March 2013

Keywords:
Fuzzy $k$-nearest neighbor method
Particle swarm optimization
Feature selection
Medical diagnosis
Parkinson’s disease

ABSTRACT

In this paper, we present an effective and efficient diagnosis system based on particle swarm optimization (PSO) enhanced fuzzy $k$-nearest neighbor (FKNN) for Parkinson’s disease (PD) diagnosis. In the proposed system, named PSO–FKNN, both the continuous version and binary version of PSO were used to perform the parameter optimization and feature selection simultaneously. On the one hand, the neighborhood size $k$ and the fuzzy strength parameter $m$ in FKNN classifier are adaptively specified by the continuous PSO. On the other hand, binary PSO is utilized to choose the most discriminative subset of features for prediction. The effectiveness of the PSO–FKNN model has been rigorously evaluated against the PD data set in terms of classification accuracy, sensitivity, specificity and the area under the receiver operating characteristic (ROC) curve (AUC). Compared to the existing methods in previous studies, the proposed system has achieved the highest classification accuracy reported so far via 10-fold cross-validation analysis, with the mean accuracy of 97.47%. Promisingly, the proposed diagnosis system might serve as a new candidate of powerful tools for diagnosing PD with excellent performance.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Parkinson’s disease (PD) is one kind of degenerative diseases of the nervous system, which has influenced a large part of worldwide population up to now. Till now, the cause of PD is still unknown, however, it is possible to alleviate symptoms significantly at the onset of the illness in the early stage [1]. It is claimed that approximately 90% of the patients with PD show vocal impairment [2], the patients with PD typically exhibit a group of vocal impairment symptoms, which is known as dysphonia. The dysphonic indicators of PD make speech measurements an important part of diagnosis. Recently, dysphonic measures have been proposed as a reliable tool to detect and monitor PD [3,4].

Various techniques have been employed to handle the PD problem. Little et al. [4] conducted a remarkable study about PD identification, they employed an Support Vector Machine (SVM) classifier with Gaussian radial basis kernel functions to predict PD, and also performed feature selection to select the optimal subset of features from the whole feature space, and the best accuracy rate of 91.4% was obtained by the best model. Shahbaba et al. [5] introduced a new nonlinear model based on Dirichlet process mixtures for classification of PD, the results have been compared with that of multinomial logit models, decision trees, and SVM, the best classification accuracy of 87.7% was obtained by the proposed approach. Das [6] presented a comparative study of using Neural Networks (ANN), DNeural, Regression and Decision Tree for effective diagnosis of PD, the experimental results have shown that the ANN classifier yielded the best results, the overall classification score of 92.5% was achieved. Sakar et al. [7] used the mutual information based feature selection methods integrated with the SVM classifier for PD diagnosis, and the classification accuracy of 92.75% was achieved. Psorakis et al. [8] introduced novel convergence measures, sample selection strategies and model improvements for multiclass multi-kernel relevance vector machines (mRVMs), and finally, the improved mRVMs achieved the classification accuracy rate of 89.47% when applied to prediction of PD. Guo et al. [9] combined genetic programming and the expectation maximization algorithm (GP-EM) to detect PD, and the best classification accuracy of 93.1% was obtained. Hossen et al. [10] implemented a wavelet-decomposition with soft-decision algorithm on electromyogram and accelerometer signals to discriminate Parkinsonian tremor from essential tremor, and a total accuracy of 85% was obtained. Wu et al. [11] presented to use a radial basis function neural network based on particle swarm optimization and principal component analysis to detect PD using Local Field Potential data, which are from the subthalamic nucleus obtained through deep brain...
electrodes implanted in a Parkinson patient. It has been found that the detection accuracy of 89% is achieved. Recently, Luukka [12] employed the feature selection method based on fuzzy entropy measures together with the similarity classifier to predict PD, and mean classification accuracy of 85.03% with only two features was obtained. Li et al. [13] proposed a fuzzy-based non-linear transformation method in combination with the SVM classifier for prediction of PD, and the best classification accuracy of 93.47% was achieved. Ozçift et al. [14] combined the correlation based feature selection (CFS) algorithm with the rotation forest (RF) ensemble classifiers of 30 machine learning algorithms to identify PD, and the best classification accuracy of 87.13% was achieved by the proposed CFS-RF model. AStröm et al. [15] proposed a parallel feed-forward neural network structure for prediction of PD, the highest classification accuracy of 91.20% was obtained. Spadoto et al. [16] applied evolutionary-based techniques in combination with the Optimum-Path Forest (OPF) classifier to detect PD, and the best classification accuracy of 84.01% was achieved. Exarchos et al. [17] proposed to use partial decision trees and association rules for the prediction of Parkinson’s disease (PD) symptoms using real patient’s data. The approach employed is able to provide interpretation for the predictions made, by providing rules. The accuracy of the symptoms’ prediction ranged from 57.1% to 77.4%, depending on the symptom. Chen et al. [18] employed the fuzzy k-nearest neighbor (FKNN) approach in combination with the principle component analysis (PCA-FKNN) to diagnose PD, and the best classification accuracy of 96.07% was obtained by the proposed diagnosis system. Daliri et al. [19] proposed a chi-square distance kernel-based SVM to discriminate the subjects with PD from the healthy control subjects using gait signals, and the proposed approach obtained an accuracy of 91.20%.

From these works, we can see that most study has employed the efficient classifiers integration with the dimensionality reduction method to further improve the performance of diagnosis of PD. It seems that both the choice of the dimensionality reduction method and classifier are of significant importance to the PD diagnosis problem. Motivated by the excellent performance achieved by the FKNN classifier on the disease diagnosis problems such as PD diagnosis [18] and thyroid disease diagnosis [20], in this study, an attempt was made to investigate a particle swarm optimization (PSO) enhanced FKNN classifier (PSO-FKNN) in constructing an automatic diagnostic system for diagnosis of PD. The main idea of PSO-FKNN was first developed for bankruptcy prediction and very promising results have been achieved [21, 22], here it was utilized to explore the maximization classification performance for detection of PD. The rationale underlying the proposed system is to use PSO to perform the feature selection and the parameter optimization simultaneously. On the one hand, the continuous PSO algorithm will be employed to evolve an adaptive FKNN, where the neighborhood size $k$ and the fuzzy strength parameter $m$ are adaptively specified. On the other hand, the binary PSO will be used as a feature selection approach to identify the most informative features as well. After the optimal model was evolved, it was utilized to perform the PD diagnostic tasks. The effectiveness of the proposed system is examined in terms of the classification accuracy, sensitivity, specificity and area under the receiver operating characteristic curve (AUC) on the PD data set taken from UCI machine learning repository. Promisingly, as can be seen that the developed diagnostic system for this data set in which a more reliable result is found (97.47% mean accuracy) by 10-fold CV method.

The main contributions of this article are summarized as follows:

a) An effective and efficient diagnosis system based on particle swarm optimization (PSO) enhanced fuzzy k-nearest neighbor (FKNN) for Parkinson’s disease (PD) diagnosis is presented.

b) In the proposed system, named PSO-FKNN, both the continuous and binary version of PSO were used to perform the parameter optimization and feature selection simultaneously.

c) Compared against the existing methods in previous studies, the proposed system has achieved the highest classification accuracy reported so far via 10-fold cross-validation analysis, with the mean accuracy of 97.47%.

The remainder of this paper is organized as follows. Section 2 offers brief background knowledge on FKNN and PSO. The detail of implementations of the FKNN-based diagnosis system is described in Section 3. In Section 4, the detailed experimental design is described, and Section 5 presents all the empirical results and discussions. Finally, Conclusions and future work are summarized in Section 6.

2. Background materials

2.1. Fuzzy k-nearest neighbor method

In FKNN [23], the fuzzy memberships of samples are assigned to different categories according to the following formulation:

$$u_i(x) = \frac{1}{\sum_{j=1}^{k} (1/||x - x_j||^{2/(m-1)})}$$

where $i = 1, 2, \ldots, C$, and $j = 1, 2, \ldots, k$, with $C$ number of classes and $k$ number of nearest neighbors. The fuzzy strength parameter $m$ is used to determine how heavily the distance is weighted when calculating each neighbor’s contribution to the membership value, and its value is usually chosen as $m \in (1, 2]$. $||x - x_j||$ is the distance between $x$ and its $j$th nearest neighbor $x_j$, usually Euclidean distance is chosen as the distance metric. $u_i$ is the membership degree of the pattern $x_i$ from the training set to the class $i$, among the $k$ nearest neighbors of $x$. In this study we adopted the constrained fuzzy membership, i.e., the $k$ nearest neighbors of each training pattern (say $x_k$) are found, and the membership of $x_k$ in each class is assigned as:

$$u_{ij}(x_k) = \begin{cases} 0.51 + \frac{(n_i/K)}{0.49}, & \text{if } j = i \\ \frac{(n_i/K)}{0.49}, & \text{if } j \neq i \end{cases}$$

The value $n_i$ is the number of neighbors which belong to the $j$th class. Note that, the memberships calculated by Eq. (2) should satisfy the following equations:

$$\sum_{j=1}^{C} u_{ij} = 1, \quad j = 1, 2, \ldots, n,$$

$$0 < \sum_{j=1}^{C} u_{ij} < n,$$

$$u_{ij} \in [0, 1].$$

After calculating all the memberships for a query sample, it is assigned to the class with which it has the highest membership value, i.e.,

$$C(x) = \arg\max_{i=1}^{C} u_i(x)$$
Fig. 1. Overall procedure of the proposed PSO–FKNN based diagnosis system.
2.2. Particle swarm optimization (PSO)

Particle swarm optimization (PSO) was first developed by Kennedy and Eberhart [24]. In PSO each individual is treated as a particle in $d$-dimensional space, and each particle has a position and velocity. The position vector of the $i$th particle is represented as $X_i = \{x_{i1}, x_{i2}, \ldots, x_{id}\}$, and itsAccordingly, the position and velocity are updated as follows:

$$v_{i,j}^{n+1} = \frac{w}{1 + \exp(-v_{i,j}^n)} + 1, \quad j = 1, 2, \ldots, d$$

where $c_1$ and $c_2$ are acceleration coefficients, to better balance the search space between the global exploration and local exploitation, time-varying acceleration coefficients (TVAC) have been introduced in Ref. [25]. The core idea of TVAC is that $c_1$ decreases from its initial value of $c_{i1}$ to $c_f$, while $c_2$ increases from $c_{i2}$ to $c_{f2}$ using the following equations as in Ref. [25]. It can be mathematically represented as follows:

$$c_1 = (c_{i1} - c_f) \frac{t}{t_{max}} + c_1$$

$$c_2 = (c_{i2} - c_{f2}) \frac{t}{t_{max}} + c_2$$

where $c_{i1}$, $c_{i2}$, $c_{f1}$ and $c_{f2}$ are constants, $t$ is the current iteration of the algorithm and $t_{max}$ is the maximum number of iterations. In addition, $r_1$ and $r_2$ are random numbers in Eq. (5) are random numbers, generated uniformly in the range $[0, 1]$. The velocity $v_{i,j}$ is restricted to the range $[-v_{max}, v_{max}]$. The inertia weight $w$ is updated according to the following equation:

$$w = w_{min} + (w_{max} - w_{min}) \frac{(t_{max} - t)}{t_{max}}$$

where $w_{max}$, $w_{min}$ are the predefined maximum and minimum values of the inertia weight $w$, $t$ is the current iteration of the algorithm and $t_{max}$ is the maximum number of iterations. For the binary PSO, one discrete PSO version introduced by Kennedy and Eberhart [26] was employed to act as the feature selection tool. In the binary PSO, a sigmoid function is applied to transform the velocity from continuous space to probability space:

$$\text{sig}(v_{i,j}) = \frac{1}{1 + \exp(-v_{i,j})}, \quad j = 1, 2, \ldots, d$$

where $\text{sig}(v_{i,j})$ is calculated according to Eq. (10), and $\text{rand}$ is a uniform random number in the range $[0, 1]$.

3. The proposed PSO–FKNN diagnosis system

In this section, we describe the proposed PSO–FKNN model for PD diagnosis. The main objective of this model is to optimize the FKNN classifier by automatically: (1) determining the number of nearest neighbors $k$ and the fuzzy strength parameter $m$ and (2) identifying the best subset of discriminative features. In order to achieve this goal, the continuous and binary versions of PSO are combined together to dynamically conduct parameter optimization and feature selection simultaneously. The obtained appropriate feature subset is served as the input into the optimized FKNN model for classification. PSO–FKNN took the diagnosis accuracy as the fitness for parameter optimization and feature selection. The overall architecture of the proposed system is shown in Fig. 1.

3.1. Parameter optimization and feature selection using PSO

In this study, the continuous PSO algorithm and binary PSO will be employed perform the parameter optimization and feature selection, respectively.

3.1.1. Particle representation

The two important parameters ($k$ and $m$) of FKNN and the feature space must be optimized using our proposed PSO–FKNN system. Without feature selection, two continuous variables, $k$ and $m$, are required. For the feature selection, $2 + n$ variables are adopted, the value of these $n$ variables is 0 or 1, where 1 means the feature is chosen and 0 means the feature is discarded. Fig. 2 illustrates the solution representation.

3.1.2. Fitness design

The classification accuracy is taken into account in designing the fitness:

$$f = \text{avgACC} = \frac{1}{K} \sum_{i=1}^{K} \text{testACC}_i$$

where variable $\text{avgACC}$ in the function represents the average testing accuracy achieved by the FKNN classifier via $k$-fold CV, where $k=5$. Note that here the 5-fold CV is employed to do the model selection that is different from the outer loop of 10-fold CV, which is used to do the performance estimation.
Based on the particle representation and fitness design, the procedure for the parameter optimization and feature selection is given below:

Pseudo-code for the parameter optimization and feature selection procedure

Begin
Randomly initialize particle swarm;

While(number of generations or the stopping criterion is not met)
  For i = 1 to number of particles
    Train FKNN model with the randomly chosen features by using 5-fold CV;
    Evaluate fitness of particle swarm;
    /* save the global optimal fitness as gfit, personal optimal fitness as pfit, global optimal particle as gbest and personal optimal particle as pbest. */
    /* Update the velocity of continuous and discrete dimensions */
    
    
    
    
    If \( v_{ij}^t \notin [V_{min}^t, V_{max}^t] \) \( v_{ij}^t = \max(\min(V_{min}^t, v_{ij}^{t-1}), V_{max}^t) \) Endif;
    /* Update the position of continuous dimensions */
    
    
    
    
    If \( (\text{rand} < \text{sig}(v_{ij})) \) \( x_{ij}^t = 1 \) Else \( x_{ij}^t = 0 \) Endif;
    /* Update the personal optimal fitness (pfit) and personal optimal position (pbest) by comparing the current fitness value (cfit) with the pfit stored in the memory. */
    If \( \text{cfit} > \text{pfit} \)
      \( \text{pfit} = \text{cfit}; \)
      \( \text{pbest} = \text{current position}; \)
    Endif;
  Endfor;

  /* Get the maximum value (maxlocal) and index from the swarm of local fitness(local_fit) and update the global optimal fitness (gfit) and global optimal particle (gbest) by comparing the gfit with the optimal pfit from the whole population */
  \[ \text{[maxlocal,index]} = \max(\text{local_fit}); \]
  If \( \text{maxlocal} > \text{gfit} \)
    \( \text{gfit} = \text{maxlocal}; \)
    \( \text{gbest} = \text{local_fit(index)}; \)
  Endif;

  Next generation until stopping criterion;
Endwhile

/* Get the best values of parameters (k and m) and the optimal feature subset (fssubset) from gbest */
\( k = \text{round}(\text{gbest}(1)); \)
\( m = \text{gbest}(2); \)
\( \text{fssubset} = \text{gbest}(3: n+2); \)
Return k, m, fssubset;
End.
3.2. Classification using FKNN

After the optimal parameter pair and feature subset were obtained, the FKNN model began to perform the classification tasks. At first, the FKNN trained on reduced the training feature space using the optimal parameter pair to evolve an optimal model, and then the optimal FKNN model was employed to predict the new samples on the reduced testing feature space. The whole process was done via the 10-fold CV analysis, and finally the average results over 10 folds were computed. The detailed pseudo-code for the classification phase is as follows:

Pseudo-code for the classification procedure
/*performance estimation by using k-fold CV where k = 10*/
Begin
For j = 1:k
    Reduced training set = k-1 subsets;
    Reduced testing set = remaining subset;
    Train the FKNN model on the reduced training feature space using the obtained optimal parameter combination;
    Test it on the reduced testing feature space and save the mean CV accuracy;
End for
Return the average classification accuracy of FKNN over / test set.
End

4. Experiments design

4.1. Data description

In this study, we have performed our conduction on the PD data set taken from UCI machine learning repository. (http://archive.ics.uci.edu/ml/datasets/Parkinsons, last accessed August 2012). The purpose of this data set is to discriminate healthy people from those with PD, given the results of various medical tests carried out on a patient. This data set is composed of a range of biomedical voice measurements from 31 people, 23 with PD. The time since diagnoses ranged from 0 to 28 years, and the ages of the subjects ranged from 46 to 85 years, with a mean age of 65.8. Each subject provides an average of six phonations of the vowel [yielding 195 samples in total], each 36 s in length (for details consult [4]). It should be noted that there is no missing values in the data set, and the whole features are real valued. The whole 22 features are presented in Table 1, along with its description.

4.2. Experimental setup

The proposed PSO–FKNN model was implemented using MATLAB platform. The empirical experiment was conducted on AMD Athlon 64 X2 Dual Core Processor 5000+ (2.6 GHz) with 4GB of RAM and the system is Windows 7.

The detailed parameter setting for PSO–FKNN is as follows. The number of the iterations and particles are set to 250 and 8, respectively. The searching ranges for k and m are as follows: kε[1,10] and mε[1,5]. v_{max} is set about 60% of the dynamic range of the variable on each dimension for the continuous type of dimensions. For the discrete type particle for feature selection, [−v_{max}, v_{max}] is set as [−6,6]. As suggested in [25], c_1, c_1, c_2, and c_2 are set as follows: c_1 = 2.5, c_1 = 0.5, c_2 = 0.5, c_2 = 2.5. According to our preliminary experiment, w_{max} and w_{min} are set to 0.9 and 0.4, respectively. The detailed parameter setup is listed in Table 2.

Normalization is employed to avoid feature values in greater numerical ranges dominating those in smaller numerical ranges, as well as to avoid the numerical difficulties during the calculation. In this study, the data are scaled into the interval of [0,1] according to Eq. (13), where x is the original value, x' is the scaled value, max_x is the maximum value of feature x, and min_x is the minimum value of feature x.

\[ x' = \frac{x - \text{min}_x}{\text{max}_x - \text{min}_x} \]  *(13)*

In order to gain an unbiased estimate of the generalization accuracy, the k-fold CV was used to evaluate the classification accuracy [27]. This study set k as 10, i.e., the data was divided into ten subsets. Each time, one of the 10 subsets is used as the test set and the other 9 subsets are put together to form a training set. Then the average error across all 10 trials is computed. The advantage of this
method is that all of the test sets are independent and the reliability of the results could be improved. And a double loop scheme [22] was adopted in our experiment, namely, the inner loop was used to determine the optimal parameters and best feature subset for the FKNN classifier, and the outer loop was used for estimating the performance of the FKNN classifier. In order to ensure the same class distribution in the subset, the data is split via stratified sampling in which the sample proportion in each data subset is the same as that in the population.

4.3. Measure for performance evaluation

Classification accuracy (ACC), sensitivity, specificity and AUC were used to test the performance of the proposed PSO–FKNN model. Before defining these measures, we introduced the concept of confusion matrix, which is presented in Table 3. Where TP is the number of true positives, which means that some cases with PD are correctly classified as ones with PD; FN, the number of false negatives, which means that some cases with PD are classified as healthy persons; TN, the number of true negatives, which means that some healthy persons are correctly classified as healthy persons; and FP, the number of false positives, which means that some healthy persons are classified as patients with PD.

According to the confusion matrix, ACC, sensitivity and specificity are defined as follows:

\[
ACC = \frac{TP + TN}{TP + FP + FN + TN} \times 100\%
\]  
\[
Sensitivity = \frac{TP}{TP + FN} \times 100\%
\]

Table 3
The confusion matrix.

<table>
<thead>
<tr>
<th>Actual patients with PD</th>
<th>Predicted patients with PD</th>
<th>Predicted healthy persons</th>
</tr>
</thead>
<tbody>
<tr>
<td>True positive (TP)</td>
<td>False negative (FN)</td>
<td></td>
</tr>
<tr>
<td>False positive (FP)</td>
<td>True negative (TN)</td>
<td></td>
</tr>
</tbody>
</table>

4.3. Measure for performance evaluation

Classification accuracy (ACC), sensitivity, specificity and AUC were used to test the performance of the proposed PSO–FKNN model. Before defining these measures, we introduced the concept of confusion matrix, which is presented in Table 3. Where TP is the number of true positives, which means that some cases with PD are correctly classified as ones with PD; FN, the number of false negatives, which means that some cases with PD are classified as healthy persons; TN, the number of true negatives, which means that some healthy persons are correctly classified as healthy persons; and FP, the number of false positives, which means that some healthy persons are classified as patients with PD.

According to the confusion matrix, ACC, sensitivity and specificity are defined as follows:

\[
ACC = \frac{TP + TN}{TP + FP + FN + TN} \times 100\% 
\]  
\[
Sensitivity = \frac{TP}{TP + FN} \times 100\% 
\]

Table 4
The detailed results obtained by PSO–FKNN without feature selection.

<table>
<thead>
<tr>
<th>Fold</th>
<th>#k</th>
<th>#m</th>
<th>AUC (%)</th>
<th>ACC (%)</th>
<th>Sensitivity (%)</th>
<th>Specificity (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>2</td>
<td>1.16</td>
<td>96.67</td>
<td>94.74</td>
<td>93.33</td>
<td>100</td>
</tr>
<tr>
<td>#2</td>
<td>2</td>
<td>1.33</td>
<td>91.67</td>
<td>94.74</td>
<td>100</td>
<td>83.33</td>
</tr>
<tr>
<td>#3</td>
<td>1</td>
<td>1.98</td>
<td>90.00</td>
<td>94.74</td>
<td>100</td>
<td>80.00</td>
</tr>
<tr>
<td>#4</td>
<td>1</td>
<td>1.72</td>
<td>96.88</td>
<td>94.74</td>
<td>93.75</td>
<td>100</td>
</tr>
<tr>
<td>#5</td>
<td>1</td>
<td>1.56</td>
<td>89.29</td>
<td>85.00</td>
<td>78.57</td>
<td>100</td>
</tr>
<tr>
<td>#6</td>
<td>1</td>
<td>1.24</td>
<td>96.88</td>
<td>94.74</td>
<td>93.75</td>
<td>100</td>
</tr>
<tr>
<td>#7</td>
<td>2</td>
<td>1.18</td>
<td>100.00</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>#8</td>
<td>2</td>
<td>1.04</td>
<td>75.00</td>
<td>95.00</td>
<td>100</td>
<td>50.00</td>
</tr>
<tr>
<td>#9</td>
<td>4</td>
<td>1.15</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>#10</td>
<td>1</td>
<td>1.33</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Avg.</td>
<td>1.70</td>
<td>1.37</td>
<td>93.64</td>
<td>95.37</td>
<td>95.94</td>
<td>91.33</td>
</tr>
<tr>
<td>Dev.</td>
<td>0.95</td>
<td>0.30</td>
<td>7.71</td>
<td>4.40</td>
<td>6.81</td>
<td>16.42</td>
</tr>
</tbody>
</table>
1 to 250 and exhibited no significant improvements after iteration 135, eventually stopped at the iteration 250 where the particles reached the stopping criterion (maximum iteration number). The fitness increase rapidly in the beginning of the evolution, after certain number of generations, it starts increasing slowly. During the latter part of the evolution, the fitness keeps stability until the stopping criterion is satisfied. This phenomenon demonstrates that PSO–FKNN can converge quickly toward the global optima, and fine tune the solutions very efficiently.

### 5.2. Experiment II: classification with the PSO–FKNN model with feature selection

In order to investigate whether feature selection can further improve the performance of detection of PD, we further conducted the PSO–FKNN model on the reduced feature space using PSO approach. In this setting, both continuous and binary versions of PSO were employed to perform the parameter optimization and feature selection simultaneously for FKNN. The detailed results obtained by the PSO–FKNN with feature selection are summarized in Table 5. From the table we can find that the PSO–FKNN with feature selection has achieved the average results of 97.37%, 97.47%, 98.16% and 96.57% in terms of AUC, ACC, sensitivity and specificity, respectively. The detailed comparison results between the PSO–FKNN with feature selection and without feature selection are summarized in Table 6. From the table we can find that with the aid of feature selection, PSO–FKNN has improved the performance by 3.73%, 2.1%, 2.22% and 5.24% in terms of AUC, ACC, sensitivity and specificity respectively. In addition, it is interesting to find that the standard deviation of PSO–FKNN is much smaller than before, which indicates PSO–FKNN becomes more robust and reliable through feature selection. Table 7 also presents the comparison results of the confusion matrices obtained by PSO–FKNN with feature selection and without feature selection. As can be seen from Table 7, PSO–FKNN with feature selection correctly classified 190 normal cases out of 195 total normal cases, misclassifies 3 patients with PD as healthy persons and 2 cases of healthy persons as patients with PD. While PSO–FKNN without feature selection can only correctly classified 186 normal cases out of 195 total normal cases.

### Table 5
The detailed results obtained by PSO–FKNN with feature selection.

<table>
<thead>
<tr>
<th>Fold</th>
<th>#k</th>
<th>#m</th>
<th>AUC (%)</th>
<th>ACC (%)</th>
<th>Sensitivity (%)</th>
<th>Specificity (%)</th>
<th>No. of selected features</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>1</td>
<td>1</td>
<td>97.34</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>13</td>
</tr>
<tr>
<td>#2</td>
<td>1</td>
<td>1</td>
<td>1.49</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>10</td>
</tr>
<tr>
<td>#3</td>
<td>1</td>
<td>1</td>
<td>1.18</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>14</td>
</tr>
<tr>
<td>#4</td>
<td>1</td>
<td>1</td>
<td>1.98</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>11</td>
</tr>
<tr>
<td>#5</td>
<td>1</td>
<td>2</td>
<td>93.75</td>
<td>90</td>
<td>87.50</td>
<td>100</td>
<td>14</td>
</tr>
<tr>
<td>#6</td>
<td>1</td>
<td>1</td>
<td>0.2</td>
<td>92.66</td>
<td>94.74</td>
<td>100</td>
<td>9</td>
</tr>
<tr>
<td>#7</td>
<td>5</td>
<td>1</td>
<td>0.8</td>
<td>90.00</td>
<td>95.00</td>
<td>100</td>
<td>14</td>
</tr>
<tr>
<td>#8</td>
<td>1</td>
<td>1</td>
<td>1.55</td>
<td>97.06</td>
<td>94.12</td>
<td>100</td>
<td>15</td>
</tr>
<tr>
<td>#9</td>
<td>1</td>
<td>1</td>
<td>1.57</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>11</td>
</tr>
<tr>
<td>#10</td>
<td>1</td>
<td>1</td>
<td>1.98</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>8</td>
</tr>
<tr>
<td>Avg.</td>
<td>1.4</td>
<td>1</td>
<td>1.5130</td>
<td>97.37</td>
<td>97.47</td>
<td>98.16</td>
<td>96.57</td>
</tr>
<tr>
<td>Dev.</td>
<td>1.26</td>
<td>0.3742</td>
<td>3.79</td>
<td>3.56</td>
<td>4.18</td>
<td>7.35</td>
<td>2.4244</td>
</tr>
</tbody>
</table>

### Table 6
Comparison results of PSO–FKNN with feature selection and without feature selection.

<table>
<thead>
<tr>
<th>FKNN with feature selection</th>
<th>Performance metric</th>
<th>Mean</th>
<th>SD</th>
<th>Max</th>
<th>Min</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACC (%)</td>
<td>97.47</td>
<td>3.56</td>
<td>100</td>
<td>90.00</td>
<td></td>
</tr>
<tr>
<td>Sensitivity (%)</td>
<td>98.16</td>
<td>4.18</td>
<td>100</td>
<td>87.50</td>
<td></td>
</tr>
<tr>
<td>Specificity (%)</td>
<td>96.57</td>
<td>7.35</td>
<td>100</td>
<td>85.71</td>
<td></td>
</tr>
<tr>
<td>AUC (%)</td>
<td>97.37</td>
<td>3.79</td>
<td>100</td>
<td>90.00</td>
<td></td>
</tr>
<tr>
<td>FKNN without feature selection</td>
<td>ACC (%)</td>
<td>95.37</td>
<td>4.40</td>
<td>100</td>
<td>85.00</td>
</tr>
<tr>
<td>Sensitivity (%)</td>
<td>95.94</td>
<td>6.81</td>
<td>100</td>
<td>78.57</td>
<td></td>
</tr>
<tr>
<td>Specificity (%)</td>
<td>91.33</td>
<td>10</td>
<td>100</td>
<td>50.00</td>
<td></td>
</tr>
<tr>
<td>AUC (%)</td>
<td>93.64</td>
<td>7.71</td>
<td>100</td>
<td>75</td>
<td></td>
</tr>
</tbody>
</table>

### Table 7
Confusion matrices of PSO–FKNN with feature selection and without feature selection.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Expected output</th>
<th>Prediction output</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSO–FKNN with feature selection</td>
<td>Patients with PD</td>
<td>Healthy persons</td>
</tr>
<tr>
<td>PSO–FKNN without feature selection</td>
<td>Patients with PD</td>
<td>Healthy persons</td>
</tr>
</tbody>
</table>

In order to demonstrate the detail of the feature selection procedure, we list the selected features by the binary PSO algorithm in 10 folds in Table 8. As shown in the Table 8, not all features are selected for classification after the feature selection. In addition, we can find that the average size of feature subset chosen by the PSO approach is 11.9 compared against the size of 22 for the original feature space. The most important features selected by PSO approach are F1, F6, F7, and F18, which have appeared in every fold during the whole 10-fold CV procedure, as can be found in the frequency of selected features as shown in Fig. 4. We also recorded the evolutionary process of PSO–FKNN with feature selection. As shown in Fig. 5 (a), we can find that the fitness curves gradually improved from iteration 1 to 250 and exhibited no significant improvements.

### Table 8
Feature selection by PSO–FKNN.

<table>
<thead>
<tr>
<th>Fold</th>
<th>Selected features</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>F1 F4 F5 F7 F9 F10 F13 F15 F16 F17 F18 F19 F21</td>
</tr>
<tr>
<td>#2</td>
<td>F1 F4 F5 F6 F14 F16 F17 F18 F21 F22</td>
</tr>
<tr>
<td>#3</td>
<td>F1 F3 F4 F7 F12 F13 F14 F16 F17 F18 F19 F20 F21 F22</td>
</tr>
<tr>
<td>#4</td>
<td>F1 F5 F7 F8 F12 F13 F16 F17 F18 F19 F20 F21 F22</td>
</tr>
<tr>
<td>#5</td>
<td>F1 F3 F4 F6 F10 F12 F13 F15 F16 F17 F18 F19 F21 F22</td>
</tr>
<tr>
<td>#6</td>
<td>F1 F5 F12 F13 F15 F16 F17 F18 F19</td>
</tr>
<tr>
<td>#7</td>
<td>F1 F6 F8 F9 F11 F13 F14 F16 F17 F18 F19 F20 F21 F22</td>
</tr>
<tr>
<td>#8</td>
<td>F1 F4 F5 F6 F7 F9 F10 F13 F14 F16 F17 F18 F19 F20 F21 F22</td>
</tr>
<tr>
<td>#9</td>
<td>F1 F2 F6 F11 F12 F15 F16 F17 F18 F19 F21 F22</td>
</tr>
<tr>
<td>#10</td>
<td>F1 F4 F6 F7 F16 F17 F18 F22</td>
</tr>
</tbody>
</table>
after iteration 80, eventually stopped at the iteration 250 where the particles reached the stopping criterion. Compared with PSO–FKNN without feature selection we can find the PSO–FKNN with feature selection converges much faster in terms of speed and achieves better results in terms of solution quality. The relation between the number of selected features and the number of generations were also recorded in Fig. 5(b). From this figure, we can see that the features are chosen by PSO in a random manner while keeping in a specified range. We also have recorded the CPU time for the whole procedure. It takes about 682 s to simultaneously perform the feature selection and parameter optimization, which is faster than the procedure without feature selection which has taken about 954 s. It indicates that the feature selection has speeded up the classification procedure.

For comparison purpose, Table 9 lists the classification accuracies of the previous methods which investigated on the PD diagnosis problem. As shown in Table 9, our developed PSO–FKNN model can obtain better classification accuracy than all available methods proposed in previous studies.

![Fig. 4. The frequency of selected features in 10-fold CV on the PD data set.](image1)

![Fig. 5. The relation between classification accuracy, number of selected features and number of generations during the training stage for fold #1 in 10-fold CV (a) Relation between classification accuracy and the number of generations (b) relation between the number of selected features and the number of generations.](image2)

<table>
<thead>
<tr>
<th>Study</th>
<th>Method</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Little et al. [4]</td>
<td>Pre-selection filter + exhaustive search + SVM</td>
<td>91.4 (bootstrap with 50 replicates)</td>
</tr>
<tr>
<td>Shahhaba and Neal [5]</td>
<td>Dirichlet process mixtures</td>
<td>87.7 (5-fold CV)</td>
</tr>
<tr>
<td>Das [6]</td>
<td>ANN</td>
<td>92.9 (hold-out)</td>
</tr>
<tr>
<td>Sakar and Kursun [7]</td>
<td>Mutual information based feature selection + SVM</td>
<td>92.75 (bootstrap with 50 replicates)</td>
</tr>
<tr>
<td>Pisorakis et al. [8]</td>
<td>Improved mRVMs</td>
<td>89.47 (10-fold CV)</td>
</tr>
<tr>
<td>Guo et al. [9]</td>
<td>GP-EM</td>
<td>93.1 (10-fold CV)</td>
</tr>
<tr>
<td>Ozçift et al. [14]</td>
<td>CFS-RF</td>
<td>87.1 (10-fold CV)</td>
</tr>
<tr>
<td>Li et al. [13]</td>
<td>Fuzzy-based non-linear transformation + SVM</td>
<td>93.47 (hold-out)</td>
</tr>
<tr>
<td>Luukka [12]</td>
<td>Fuzzy entropy measures + Similarity classifier</td>
<td>85.03 (hold-out)</td>
</tr>
<tr>
<td>Spadoto et al. [16]</td>
<td>Particle swarm optimization + OPF</td>
<td>73.53 (hold-out)</td>
</tr>
<tr>
<td>Harmony search + OPF</td>
<td></td>
<td>84.01 (hold-out)</td>
</tr>
<tr>
<td>Gravitational search algorithm + OPF</td>
<td></td>
<td>84.01 (hold-out)</td>
</tr>
<tr>
<td>Åström and Koker [15]</td>
<td>Parallel NN</td>
<td>91.20 (hold-out)</td>
</tr>
<tr>
<td>Chen et al. [18]</td>
<td>PCA-FKNN</td>
<td>96.07 (average 10-fold CV)</td>
</tr>
<tr>
<td>This study</td>
<td>PSO-FKNN</td>
<td>97.47 (10-fold CV)</td>
</tr>
</tbody>
</table>
6. Conclusions and future works

In this work, we have developed an effective and efficient diagnosis system, PSO–FKNN, for addressing PD diagnosis problem. The core component of the system is the adaptive FKNN classifier, whose maximum potential is explored by PSO approach. The excellent performance obtained on the PD dataset has proven that the proposed system can distinguish well enough among patients with PD and healthy persons. It was observed that PSO–FKNN achieved the highest classification accuracy of 97.47% via 10-fold CV. Based on the empirical analysis, it can be safely concluded that, the developed PSO–FKNN diagnosis system can assist the physicians to make very accurate diagnostic decision. The future investigation will pay much attention to evaluating the proposed system in other medical diagnosis problems.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (NSFC) under Grant Nos. 60973040, 60903098, 61202171 and 61272018, the basic scientific research foundation for the interdiscipliary research and innovation project of Jilin University under Grant No. 201103129, the China Postdoctoral Science Foundation funded project under Grant No. 2012M521251 and 2012M511339. This work is also funded by Zhejiang Provincial Natural Science Foundation of China under Grant No. R1110261 and Y1100769.

References