Combination of Multiple Classifiers for Classifying the Diabetic Data

Fatemeh Adibzadeh and Mohammad Hasan Moradi

Biomedical Engineering Department, Amir Kabir University of Technology, Tehran, Iran

The paper was received on March 31, 2008.

Abstract: Multiple classifier combination is a technique that combines the decisions of different classifiers. Combination can reduce the execution time of classification, variance of estimation errors, thereby improving the overall classification accuracy. This paper introduces a genetic algorithm able to combine three different classifiers, fuzzy, MLP, K-NN. The use of a genetic algorithm is motivated by the fact that the combination phase is based on the optimization of vote strategy. The method has been applied to classification of The Pima Indians Diabetes database, results show a significant improvement of recognition accuracy using the genetic algorithm combination strategy compared with the recognition accuracy of each single classifier.

Keywords: Classification, Classifier fusion, Fuzzy sets, Genetic algorithms, Multiple Classifiers.

1. Introduction

Multiple classifier combination is a technique that combines the decisions of different classifiers that are trained to solve the same problem, but make different errors. Numerous studies of combination systems have been attempted in a wide range of pattern recognition fields. The findings show that combination can reduce the variance of estimation errors, thereby improving the overall classification accuracy owing to the reduced variance in the estimated decision boundary.

Generally speaking a successful ensemble method depends on two components: a set of appropriate classifiers and a combination method, function or scheme [1].

Multiple classifier systems generally differ from the number and the type of the individual classifiers used, (neural network, linear classifier, nearest neighbor classifier), the features subsets used by each single classifier [2], the integration of individual decisions (e.g. mixture of experts [3], ensemble averaging [4]), the training data sets used for the individual classifiers (e.g. boosting and bagging [5]).

In the following a genetic algorithm able to integrate the classification results, coming from three different classifiers, is proposed. The combination strategy is based on the optimization of the integration coefficients deriving from vote strategy. The methodology has been applied to classify the Pima Indians Diabetes database.

Moreover, the predictive accuracy of the combined results of the GA-based method is significantly better than that of individual classifiers.

In section 2 the integration problem is outlined, section 3 describes the classifiers, section 4 describes the GA-based combination method. The results of this method are shown in section 5. Section 6 is dedicated to final remarks.
2. Integrated fuzzy algorithm

In the last decade, several methods for the integration of multiple classifier have been developed [6,7].

Integrated Fuzzy Classification (IFC) is an integration method of individual decisions, which consists in the weighted combination of more than one classifier \( Cl_1, Cl_2, \ldots, Cl_S \), in order to increase the performance and the accuracy of data classification. For example, weighted-vote functions may be used to model the accuracy related to each classification algorithm.

There are two main strategies: the Parallel Integration (IFC-parallel) and the Sequential Integration (IFC-sequential).

In the case of IFC-parallel classification algorithms are executed independently. It should be preferred whenever the classification algorithms do not need to exchange information during the computation.

The sequential strategy is preferred whenever the classifiers interact in finding the best solution.

In this study we integrate three different classifiers using the IFC-parallel combination paradigm, because three different classifiers offer complementary information about instances to be classified which could be harnessed to improve the performance of the individual classifiers [8].

3. Individual classifiers

This section provides a review of the three different classifiers. All the classifiers use the same data set \( U \) and feature parameters.

3.1. Fuzzy classifier

A fuzzy classifier is any classifier which uses fuzzy sets either during its training or during its operation.

Let us assume that each axis of pattern space is partitioned into \( K \) fuzzy subsets, each fuzzy subset has one fuzzy if-then rule. The consequent and the grade of certainty of fuzzy if-then rules are determined by the following procedure.

Procedure 1 - Generation of fuzzy if-then rules:

Step 1: Calculate \( \beta \) for each class as

\[
\beta = \sum_{p \in CT} \prod_{i=1}^{8} \mu_i^K(x_p)
\]

Where \( \beta \) is the sum of the compatibility of \( x_p \)'s in class \( T = \{1,2,\ldots,M\} \) to the fuzzy if-then rule.

Step 2: Find class \( X(CX) \) such that

\[
\beta_{CX} = \max \{ \beta_{C1}, \beta_{C2}, \ldots, \beta_{CM} \}
\]

Step 3: If a single class takes the maximum value in (2) \( CF^K \) is determined as

\[
CF^K = (\beta_{CX} - \beta) / \sum_{T=1,\ldots,s}^{\beta_{CT}}
\]

\[
\beta = \sum_{T=1,\ldots,s}^{\beta_{CT}} / (M - 1)
\]

By specifying \( K = 2,3,\ldots,L \) in procedure 1, \( 2^2 + 3^2 + \ldots + L^2 \) fuzzy if-then rules in \( S_{ALL} \) are generated. In this study we applied procedure 1 with \( K = 3 \).

Procedure 2 - classification of a new pattern

Step 1: Calculate \( \alpha_{CT} \) for each class \( T (T = 1,2,\ldots,M) \) as

\[
\alpha_{CT} = \max \{ \prod_{p=1}^{8} \mu_i(x_p).CF^K | C_i = CT, R_j \in S \}
\]

Step 2: Find class such that

\[
\alpha_{CX} = \max \{ \alpha_{C1}, \alpha_{C2}, \ldots, \alpha_{CM} \}
\]

3.2. MLP classifier

Our next classifier is a feed forward neural network (Multi Layer Perceptron) trained as a pattern classifier. The momentum back-propagation algorithm is used to train the network. The network consists of 8 nodes in the input layer (corresponding to the 8 dimensions in the feature space), 70 nodes in the first hidden layer, 25 nodes in the second hidden layer, and 3 nodes in the output layer. The output of the third node in the output layer with [zero to one] range, reflects the response of the network to the corresponding class.

3.3. K-NN classifier

This classifier is defined by a distance:

\[
d^k(X) = 1 - \frac{c(X,k)}{r}
\]

Each component \( d^k \) of the distance \( d \) evaluates the
closeness of an unclassified data to the \( k \)th class.

The term \( c(X, k) \) counts how many data of class \( k \) are in the set of the \( r \) nearest neighbors to \( X \) evaluated considering distance \( \delta \), where \( \delta \) is so defined:

\[
\delta(a, b) = \frac{1}{n} \sum_{i=1}^{n} \frac{|a_i - b_i|}{\max(a_i, b_i)}
\]  

(7)

More formally if \( T_{mn}(X) \) is a set of training which are the \( r \) nearest neighbors of \( X \) evaluated considering distance \( \delta \), and \( T_{mn}^{(k)}(X) \subseteq T_{mn}(X) \) is the subset of those in class \( k \),

\[
c(X, k) = |T_{mn}^{(k)}(X)|
\]  

(8)

4. The GA-based combination method

The above described classifiers provide different classification result. An intense research around classifier fusion in recent years revealed that combining performance strongly depends on careful selection of classifiers to be combined. On the other hand, there is already a number of classifier fusion techniques available and the choice of the most suitable method depends back on the selections made within classifier. In all these multidimensional selection tasks genetic algorithms (GA) appear to be one of the most suitable techniques providing reasonable balance between searching complexity and the performance of the solutions found. In this work, we combine above three classifiers outputs by GA-selected weights for weighted majority voting method.

Genetic algorithms (GAs) [9-10] are heuristic search techniques that are based on the theory of natural selection and evolution (Holland, 1975). GAs differ from more traditional optimization techniques in that they involve a search from a population of solutions, not from a single point. Each generation of a GA involves a competitive selection that weeds out poor solutions. Crossover and mutation are used to generate new solutions, the crossover operator takes advantages of already found solutions while the mutation is mainly responsible for the exploration of new regions of the search space. Let \( S \) be a pattern space that consists of \( M \) mutually exclusive sets \( S = C_1 \cup \ldots \cup C_M \), each of \( C_i, i \in \Lambda \{1, \ldots, M\} \) representing a set of specified patterns called a class. \( e_i(x) \) means a classifier where \( k = \{1, \ldots, K\} \) and \( x \) denotes an input pattern. For a given input pattern \( x \), \( e_i(x) = \{m'_i(x) \} \forall i(1 \leq i \leq M), \forall k(1 \leq k \leq K) \} \) means a classifier \( k \) assigns the input pattern \( x \) to each class \( i \) with a measurement value \( m'_i(x) \).

Let \( W = \{w_i\} \forall i(1 \leq i \leq M), \forall k(1 \leq k \leq K) \} \) be the set of weights, where \( w_i \) means the degree of importance of the \( k \)th classifier for class \( i \) and has a positive value between 0 and 1.

For an input pattern \( x \), the final output \( E_i(x) \) for class \( i \) is calculated as the weighted sum of measured values \( m'_i(x) \) and the corresponding weight values \( w_i \) and is expressed as:

\[
E_i(x) = \sum_{k=1}^{K} w'_i m'_i(x)
\]  

(9)

The final decision is given by selecting the class label with the highest output value \( E_i(x) \).

For a GA, each weight vector should be encoded into a string called a chromosome. The initial population consists of a set of weights distributed randomly. Once the initial population is generated, the GA evaluates each individual according to the fitness function. The role of the fitness function is to encode the performance of each individual numerically. In this study, the objective of the GA method is to find the set of weights capable of generating the optimized combination results. The fitness function for the \( l \)th weight set is defined as follows:

\[
2^l \text{ Fitness } (W_l) = \left[ \frac{\sum_{c \in C_l} HR(W_l)}{N} \right]
\]  

(10)

where \( N \) is the total amount of data.

As shown in Eq. (5), the fitness function is
measured using the sum of $HR(W_i)$ for all input data.

For an input pattern $x$, $HR(W_i)$ can be calculated as:

$$HR(W_i) = \begin{cases} 
1 & \text{if correctly matched} \\
\frac{E_i^W}{\sum_{i=1}^{K} E_i^W} & \text{otherwise}
\end{cases} \quad (11)$$

Where $j$ is the actual class for input $x$, $W_i$ is the $l$th candidate weight vector, and $E_i(x) = \sum_{l=1}^{K} m_l^i(x)$ is the final output for class $i$. If the individuals classify the input correctly, the score is increased by one. Otherwise, it is increased by the ratio of the measurement for the actual class to the sum of all measurement to consider the potential hit of the individual.

Once each weight set is evaluated using the fitness function, GAs select the best solution to reproduce new offspring using genetic operators, such as crossover and mutation. The good offspring replace old candidates with a low fitness value. After replacement, the new population is evaluated using the fitness function. This process continues iteratively until a predefined stopping condition is reached.

### 5. Results

The used data set U is Pima Indians Diabetes database which is selected from the UC Irvine machine learning, and is composed of 768 data representing two different classes. For the experiments we have used a training set T of 400 random data for MLP classifier, and the first 400 data for Fuzzy classifier, and a test set Ts composed by remaining 368 data. And all 768 samples are used for both training and testing of K-NN classifier.

Generally, the population size of GA is based on the size of the problem. We used 100 strings in the population. A total of 5 runs of the genetic algorithm has been performed, for each run several parameters had to be defined for genetic operators because the values of these parameters can have a great influence on the algorithm. Each run using a crossover probability $pc$ of 0.59, a mutation probability $pm$ of 0.01. The set of individuals was evolved for 100 generations.

The weight vector generated by the GA-based method is shown in table 1.

Table 2 shows the classification accuracy for each classifier and for the integrated method on both classes. The values show that the accuracy of the combined method is significantly better than that of individual classifiers. The proposed system can achieve a diagnostic accuracy of 99.4% with 100% sensitivity and 99.92% specificity for the identification of diabetic database.

Table 3 shows the performance of the different classifiers for diabetic database. As it can be seen, the described method improves the classification accuracy significantly, in addition to the advantage of short time execution.

Table 1: The optimal set of weights obtained from the GA-based method

<table>
<thead>
<tr>
<th>$W_k$</th>
<th>FUZZY</th>
<th>MLP</th>
<th>K-NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class 1</td>
<td>0.5905</td>
<td>0.1603</td>
<td>0.4947</td>
</tr>
<tr>
<td>Class 2</td>
<td>0.4071</td>
<td>0.2030</td>
<td>0.5561</td>
</tr>
</tbody>
</table>

Table 2: The classification accuracy of the individual and combined classifiers

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuzzy</td>
<td>77.6</td>
</tr>
<tr>
<td>K-NN</td>
<td>66.0</td>
</tr>
<tr>
<td>Neural Network</td>
<td>80.16</td>
</tr>
<tr>
<td>Integrated (GA)</td>
<td>99.4</td>
</tr>
</tbody>
</table>
Table 3: The performance of the classifiers for Pima database

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBF Network</td>
<td>65.88</td>
</tr>
<tr>
<td>Kohonen</td>
<td>72.7</td>
</tr>
<tr>
<td>Naive Bayes</td>
<td>76.30</td>
</tr>
<tr>
<td>Fisher disc. Analysis</td>
<td>76.5</td>
</tr>
<tr>
<td>Logdisc</td>
<td>77.7</td>
</tr>
<tr>
<td>LS-SVM</td>
<td>78.21</td>
</tr>
<tr>
<td>LDA</td>
<td>96.8</td>
</tr>
<tr>
<td>Integrated (GA)</td>
<td>99.4</td>
</tr>
</tbody>
</table>

Table 4: Comparison between proposed method and some class-dependent fusion techniques

<table>
<thead>
<tr>
<th>Fusion rules</th>
<th>Accuracy%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product</td>
<td>84.24</td>
</tr>
<tr>
<td>Sum</td>
<td>86.68</td>
</tr>
<tr>
<td>Max</td>
<td>83.69</td>
</tr>
<tr>
<td>Bayes</td>
<td>78.26</td>
</tr>
<tr>
<td>Proposed Method</td>
<td>99.4</td>
</tr>
</tbody>
</table>

Figure 1. Performance of based classifiers in comparison with our combined method

Figure 2. Performance of different decision fusion strategies

6. Conclusions

The integration procedure described in this paper is based on a GA approach, which searches the best parameters able to integrate three different classifiers. The genetic paradigm has been chosen because it allows to search in a large solution space that can be defined subjectively. Here an application of diabetic database is proposed, and results show an improvement of the classification accuracy using the integration of the classifiers by a voting strategy where the weights are suggested by a GA. The comparative results indicate that implementation of our method gives better performance compared with some other fusion techniques such as sum, max, min, product rule and fuzzy-integration methods. The author aims in the future, to compare on other data sets this integrated classifier with other classifier algorithms and integration strategies.

References


Fatemeh ADIBZADEH: She received her B.S. degrees in Biomedical Engineering and also Electrical Engineering from Amir Kabir University of Technology (Tehran Polytechnic), Iran. Currently, she is nearing the end of her masters in which she is working on combination of multiple classifiers for pattern recognition of biomedical signals. Her Research interests include Pattern Recognition and Image/Signal processing, Biological System Modeling and System Identification, Bio-instruments and Computational Neuroscience.

Mohammad H. MORADI: He received the B.S. and M.S. degrees in electronic engineering from Tehran University, in 1988 and 1990, respectively, and the Ph.D. degree from the University of Tarbiat Modarres, Tehran, in 1995. He has been with the faculty of biomedical engineering, Amirkabir University of Technology (AUT), since 1995, where he is currently an Associate Professor and Director of Higher education. His primary research and teaching interests involve the theory and application of medical instrumentation, biomedical signal processing and fuzzy neural systems. He has published over 20 technical papers in international journals, over 100 technical papers in international conferences and is the translator of one book with subject of wavelet signal processing.