Expert Systems with Applications 40 (2013) 4648–4652

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Particle Swarm Optimization of MLP for the identification of factors related to Common Mental Disorders

Teresa B. Ludermir [a,*], Wilson R. de Oliveira [b]

[a] Center of Informatics, Federal University of Pernambuco, Recife, Brazil
[b] Federal Rural University of Pernambuco, Recife, Brazil

ARTICLE INFO

Keywords:
Common Mental Disorder
Neural Network Optimization
Particle Swarm Optimization

ABSTRACT

Social class differences in the prevalence of Common Mental Disorder (CMD) are likely to vary according to time, culture and stage of economic development. The present study aimed to investigate the use of optimization of architecture and weights of Artificial Neural Network (ANN) for identification of the factors related to CMDs. The identification of the factors was possible by optimizing the architecture and weights of the network. The optimization of architecture and weights of ANNs is based on Particle Swarm Optimization with early stopping criteria. This approach achieved a good generalization control, as well as similar or better results than other techniques, but with a lower computational cost, with the ability to generate small networks and with the advantage of the automated architecture selection, which simplify the training process. This paper presents the results obtained in the experiments with ANNs in which it was observed an average percentage of correct classification of individuals with positive diagnostic for the CMDs of 90.59%.

1. Introduction

The Common Mental Disorders (CMDs), and among them the anxiety and depression have been pointed out as the common causes of morbidity in developed countries as much as in the developing ones, as the example of Brazil. These mental disorders represent a high social and economic charge because they are disabled, they constitute important cause of lost of workdays and they take a substantial use of health care services (Ludermir & Lewis, 2003). The use of techniques that may lead to an identification of the factors that present the larger possibility of being related to these CMDs it is of great relevance to assist within the process of decision taking around the planning and intervention of public health care. Artificial Neural Networks (ANNs) have been largely used in the health care field and they are known because they generally obtain a good precision result (Marcano-Cedeño et al., 2013; Babu & Suresh, 2013). When they are applied to epidemiological data the ANNs have also had acceptance (Chernbumroong, Cang, Atkins, & Yu, 2013).

With this research we intend, mainly, to experimentally display that a MLP network trained with Particle Swarm Optimization (PSO) (Kennedy & Eberhart, 1995) with early stopping criteria is able to identify the factors related to the CMDs. Global search techniques, such as Tabu Search (TS) (Glover, 1989), Evolutionary Algorithms (EAs, like Genetic Algorithm - GA) (Eiben & Smith, 2003), Differential Evolution (DE) (Storn, 1999), Particle Swarm Optimization (PSO) (Kennedy & Eberhart, 1995) and Group Search Optimization (GSO) (He, Wu, & Saunders, 2009), are widely used in scientific and engineering problems, and these strategies have been combined with ANNs to perform various tasks, such as connection weight initialization, connection weight training and architecture design. PSO has some advantages with respect to evolutionary algorithms. PSO for example has no complicated operators as evolutionary algorithms and it has less parameters which need to be adjusted (Kennedy & Eberhart, 1995).

The obtained results in the experiments with ANNs were compared with the ones presented by Ludermir and Lewis (2003) who applied the logistics regression method, using the same data basis to analyze the independence of each variable association with the CMDs. On the statistic analysis for the identification of the factors related to the CMDs, it was estimated the simple and adjusted odds-ratios, whose statistic significance was evaluated by the Students *t*-test, considering the 95% confidence interval and values of $p <= 0.05$.

The remainder of the article is organized as follows: Section 2 presents the standard PSO algorithm and the proposed methodology, PSO–PSO:WD. Section 3 describes the data basis and Section 4 presents the experimental setup of this work. Section 5 presents and analyzes the results obtained from the experiments and in Section 6 we summarize our conclusions and future works.

* Corresponding author. Tel.: +55 81 21268430.
E-mail addresses: tbl@cin.ufpe.br (T.B. Ludermir), wilson.rosa@gmail.com (W.R. de Oliveira).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.02.003

2. Particle Swarm Optimization

This section presents the basic algorithm of Particle Swarm Optimization and the algorithm based on the interleaved execution of two PSO algorithms.

2.1. Basic concepts

The Particle Swarm Optimization (PSO) is a population-based stochastic optimization technique developed by Kennedy and Eberhart in 1995. PSO attempts to model the flight pattern of a flock of birds (Kennedy & Eberhart, 1995). In PSO, each particle represents a candidate solution within a n-dimensional search space. The position of a particle i at iteration t is denoted by $\mathbf{x}_i(t) = [x_{i1}, x_{i2}, \ldots, x_{in}]$. In each iteration of the PSO, each particle moves through the search space with a velocity $\mathbf{v}_i(t) = [v_{i1}, v_{i2}, \ldots, v_{in}]$ calculated as follows:

$$v_{ij}(t+1) = w v_{ij}(t) + c_1 r_{j1}[y_{ij}(t) - x_{ij}(t)] + c_2 r_{j2}[\hat{y}_{ij}(t) - x_{ij}(t)], \quad (1)$$

where w is the inertia weight, $\mathbf{y}_i(t)$ is the personal best position of the particle i at iteration t and $\hat{\mathbf{y}}(t)$ is the global best position of the swarm at iteration t. The personal best position is named *pbest* and it represents the best position found by the particle during the search process until the iteration t. The global best position is named *gbest* and it represents the best position found by the entire swarm until the iteration t. The terms c_1 and c_2 are acceleration coefficients and are responsible for taking control of how far a particle can move in a single iteration. The terms r_{j1} and r_{j2} are random numbers sampled from an uniform distribution $U(0,1)$. The velocity is limited to the range $[\mathbf{v}_{min}, \mathbf{v}_{max}]$.

After updating velocity, the new position of the particle i at iteration $t+1$ is calculated using Eq. (2)

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1). \quad (2)$$

In Shi and Eberhart (1998), Shi and Eberhart proposed an adaptive inertia in which the parameter w reduces gradually as the iteration increases according to Eq. (3)

$$w(t) = w_{max} - t \times \frac{(w_{max} - w_{min})}{t_{max}}, \quad (3)$$

where w_{max} is the initial inertia weight, w_{min} is the final inertia weight and t_{max} is the maximum number of iterations. The inertia weight can control the degree of exploration of the search.

The standard PSO algorithm is presented in Algorithm 1. Rapid convergence in unimodal functions, with good success rate, and premature convergence in multimodal functions are properties frequently attributed to the standard PSO algorithm.

Algorithm 1. PSO Algorithm

```
1: Initialize the swarm S;
2: while stopping condition is false do
3:     for all particle i of the swarm do
4:         Calculate the fitness f(x_i(t));
5:         Set the personal best position y_i(t);
6:     end for
7:     Set the global best position ŷ(t) of the swarm;
8:     for all particle i of the swarm do
9:         Update the velocity v_i(t);
10:        Update the position x_i(t);
11:    end for
12: end while
```

2.2. The PSO–PSO Methodology

The methodology used to optimize weights and architectures of MLP neural networks is based on the interleaved execution of two PSO algorithms, one for weight optimization (inner PSO) and the other for architecture optimization (outer PSO). This approach was presented by Zhang and Shao in Zhang and Shao (2000), in which few details were given on the performance of the optimized neural networks.

In this methodology, the outer PSO simply searches for the number of hidden units for each of the considered hidden layers in the MLP network. In this work, we considered only network architectures of one hidden layer, but the extension for a more general case is straightforward. The inner PSO is responsible for the optimization of weights for each of the architectures (particles) present in the outer PSO. At the end of the inner PSO execution for an architecture of the outer PSO, the best set of weights found is recorded in the particle representing that architecture. The two processes are interleaved for a specific number of times.

The PSO–PSO methodology developed in this work used a PSO algorithm to search for architectures and a PSO with weight decay (PSO:WD) to search for weights. The PSO:WD algorithm was created in a previous work (Carvalho & Ludermir, 2006) and has more generalization control than the standard PSO. For the evaluation of performance of the particles of the two PSOs, we used different partitions of the example patterns set. The training set partition (50%) was used within the inner PSO to optimize weights while the validation set partition (25%) was used within the outer PSO to search for architectures. The remained data (25%) was used to test the final MLP network found by the process.

It should be noted that all the three partitions used in the methodology are disjoint. That restriction is related to the aim of improving the generalization control of the optimized networks. This can be done through the adjustment of the complexity (number of hidden units) of the networks guided by data examples different from the ones used to guide the search for synaptic weights.

The complete algorithm for the methodology created in this work is presented in Algorithm 2, in which the term $A_i.net$ represents the vector used to record the best network found so far for the architecture A_i. Note that this term is updated with the best particle at the end of an execution of the inner PSO (line 7), and is used to assist a new execution of the inner PSO with previous good results (line 5). The PSO:WD algorithm is better described in Carvalho and Ludermir (2006) in which the standard PSO is combined with weight decay heuristic as an attempt to improve the generalization performance of the trained MLP networks.

3. Data basis description

Data collection was community-based through interviews and assessment of mental health status from a research made in the city of Olinda, Brazil by Ludermir and Lewis (2003). With that research Ludermir and Lewis (2003) determined the prevalence of the CMDs in that area, and analyzed the association with living and work conditions.

The study was developed with 621 adults of an aleatory domicile sample and the analysis of data was made using a statistic model of logistic regression.

Algorithm 2. PSO–PSO:WD Algorithm for simultaneous optimization of weights and architectures

```
1: Initialize the swarm - the population of architectures A;
2: while stopping condition is false do
3:     for all particle A_i of the swarm A do
4:         Initiate PSO:WD P_i;
5:         Insert A_i.net into P_i;
6:         Execute P_i by t iterations through training set;
7:         A_i.net = P_i.ŷ;
8:         Evaluate f(A_i.net) trough validation set;
9:     end for
10:    for all particle A_i of the swarm A do
11:        Update the velocity v_i(t);
12:        Update the position x_i(t);
13:        Update the A_i.net to the new architecture
   represented by A_i;
14:    end for
15: end while
```

The data set has the following variables: age, literacy, migration, education, house ownership, insertion on the productive process, housing conditions, gender, marital status, income and possession of household appliances.

The living conditions were measured from the variables of literacy, education, house ownership, housing conditions and possession of household appliances. As for the work conditions, the observed variables were performed by means of insertion in the productive process and household per capita monthly income. The total prevalence of the CMDs in the studied sample was 35%, 216 cases.

All variables in the date set are ordinal/categorical and, for the execution of the experiments with ANNs, were codified with discrete numbers between 0 and 1. The network output was defined with two nodes, having the 1 0 value used to represent those cases with the positive diagnose for the CMDs, and 0 1 for those cases with the negative diagnose.

Since the basis of the original data there were only 35% (216) of positive diagnose cases of CMDs, the experiments were performed ensuring the balance of the data with the same amount of the disorder bearer and non-bearer individual in the training, validation and test sets. That way, it was excluded the exceeding cases that presented negative diagnose for the CMDs. For the exclusion of the exceeding patterns, we used the following procedure: (1) it was taken out the patterns with missing information; (2) after the mixing of the left data, it was randomly selected 216 patterns for the composition of the basis.

After the exclusion of the exceeding patterns, a new mixing was made, at this time with the 432 patterns, willing the division of the new mixing set in training, validation and test sets. For that division it was followed what the Proben1 (Prechelt, 1996) suggests: 50% of the patterns for the training set (216); 25% for the validation set (108 cases); 25% for the tests set (108 cases).

4. Experimental setup

The experiments included the use of a network MLP, trained with the PSO–PSO:WD algorithm for the simultaneous optimizing of the architecture (input nodes, hidden nodes and connections) and the weights of the network. It was observed the variables that were mostly used for the results obtaining on every execution of the algorithm, and with that, to identify those, which presented grater possibility of being related with the studied problem. This technique was adapted from Zanchettin, Ludermir, and Almeida (2011). The obtained results were compared with those presented by Ludermir and Lewis (2003) applying the statistic model of logistic regression.

The experiments were executed in two distinct stages: (1) in the beginning the data set was composed with all the data basis variables, in a total of 11; (2) from the obtained results in first experiments, it was performed new experiments with the number of resultants input variables, in a total of 7. These 7 variables were chosen based on the number of times the algorithm had chosen such variables. That is the most used variables (in terms of percentage) were chosen.

The algorithm was implemented in a way to optimize not only the units of the network input, but also the nodes of the hidden layer and connections. Since the PSO–PSO:WD is probabilistic and on each execution of it by weights initializing it may result in different topologies, it was necessary the definition of a initial topology containing one hidden layer with four nodes and having all the possible feedfoward connections between the adjacent layers (this initial topology was defined for the two experiment stages). For definition of the number of nodes of the hidden layer, several experiments were accomplished varying that parameter. Topologies were tested containing up to sixteen nodes in the hidden layer. The use of more than 4 nodes in the hidden layer did not influence in improvement results of the experiments. A created network was valid if it has a minimum of one node at the hidden layer. If one created solution was not a valid network, a new solution was created in the neighborhood.

Ten aleatory weights initializing were done, and for each initializing, thirty executions of the PSO–PSO:WD algorithm was made. The weights were randomly initialized between 1.0 and +1.0. The established criterion for the training interuption was the GL5 from Proben1 (which it is based in the sum of the squared error in the validation set) (Prechelt, 1996), willing in that way, minimize the overfitting risk. The error was measured based on the classification error for the validation set after 300 interactions. The maximum number of allowed interactions was 5000, being also used as the criteria of the training interruption in case the GL5 was not satisfied.

The representation of MLP networks adopted for the inner PSO (optimization of weights) is based on vectors of real values corresponding to each of the network weights. For the outer PSO algorithm (optimization of architectures) we used integer scalar variables to represent the number of hidden units of the considered hidden layer and a vector of real values (net) to record the best network found so far for the underlying architecture.

The observed aspects for analyzing the results at the end of the experiments were the classification error in the test set, the percentage of correct classification of individuals with positive and negative diagnosis for the CMDs and the average number of variables used by the PSO–PSO:WD algorithm in the obtaining of results.

It was necessary to establish a criteria for the choice of the PSO–PSO:WD performance that had to be considered in the analysis, that way, the executions with the classification error greater than 28.13% had to be excluded. This percentage was obtained with the classification error average presented in the experiments performed with the Backpropagation and PSO–PSO:WD algorithms. The definition of this criterion, in our experiments, was necessary to avoid a super adjusting of the model.

For the definition of the variable amount that presented most relation with the CMDs, it was observed the average of variables used for the diagnostic classification (input nodes) in the analyzed executions of the PSO–PSO:WD algorithm.

Table 1
Experimental results.

Input variable	Experiment1	Experiment2
Age	**62.07**	52.00
Literacy	58.62	
Migration	51.72	
Education	**89.65**	72.00
House ownership	**65.52**	20.00
Insertion productive proc.	**65.52**	72.00
Housing conditions	51.72	
Gender	**65.52**	60.00
Status marital	**79.61**	40.00
Income	**62.07**	72.00
Possession household app.	48.27	
Classification error	23.07	21.37
Positive diagnose	89.08	90.59
Negative diagnose	64.75	66.66

5. Results

The experiments were done in two stages: (1) initially the input set was composed with all the data base variables, in a total of 11; (2) from the obtained results in the initial experiments, it was performed new experiments with only resultants variables, in a total of 7. Table 1 presents the experimental results in the following way: Experiment1 with 11 inputs, Experiment2 with 7 inputs. The table contains the use percentage of each variable, an average classification error, the percentages of the correct classification of the cases with positive/negative diagnose for the CMDs. The bold face values in experiment1 are for the input variables which were most used in the experiment. The Students t-test with a significance level of 5% was used to perform the statistical analysis in the results.

We obtained in the experiments with 11 inputs an average classification error of 23.08%, the percentage of the correct classification of the cases with positive diagnose for the CMDs was of 89.08% and the percentage of correct classification of the negative diagnose cases was 64.75%. The average of the resultant input variables of these experiments was seven, and among those, which were stood out are: age, education, house ownership, insertion in the productive process, gender, status marital and income.

Comparing the results of 7 inputs with 11 inputs, in relation to the classification error, there was a reduction, in average, in 1.71%. The correct classification percentage in the cases with positive diagnose was increased in 1.51%. As for the correct classification percentage of the cases with negative diagnose for the CMDs presented, however, there was an increase in 1.91%.

In general, in the performed experiments with only the 7 variables mostly used by the PSO–PSO:WD algorithm, the network classification error as well as the correct classification percentage of the cases with positive diagnose for the CMDs were improved. Those results suggest that the exclusion of the input variables that did not present relationship with CMDs in the data basis used contributed to the improvement of the results obtained in the experiments. Therefore, the process of variable and feature selection improved the performance of the system, provided faster and more cost-effective systems and provided a better understanding of the underlying process that generated the data.

6. Final considerations

In this work, we have analyzed the feed-forward neural networks weights and architecture optimization problem with the use of a methodology entirely based on the Particle Swarm Optimization algorithm. The results obtained by this methodology are situated between the results presented by other well studied techniques such as Genetic Algorithms or Simulated Annealing. This methodology was inspired by a similar process described in Zhang and Shao (2000), where two PSO algorithms are interleaved in the optimization of architectures (outer PSO) and connection weights (inner PSO) of MLP neural networks. A small modification made in this work concerning generalization control improvements was the use of the weight decay heuristic in the inner PSO, i.e. the process of weights adjustment.

Even though the logistic model is the methodology normally used when the purpose is to identify the factors of risk that have association with the variable answer, where the coefficients of regression may be interpreted by the odds-ratios, it was possible to observe good results in the experiments with MLP with relation to the prediction of the positive cases for the studied problem. The obtained average in our experiments around the correct classification of the individuals with positive diagnose for the CMDs was of 90.59%.

We may observe that the presented results in the experiments with neural networks were similar to those obtained with the statistic technique of logistic regression applied by Ludermir and Lewis (2003), when it is compared with the analysis made with simple odds-ratios, where the variables education, income, gender and insertion in the productive process were statistically significant, with the values of $p <= 0.0001$. After the adjustment of the odds-ratios in the results obtained by Ludermir and Lewis (2003), education and income presented relationship with CMDs. It is important to remind that those variables, in all the experiments with ANNs, they were present in the obtained results, standing out among the variables that presented larger percentage of use for the algorithm, in other words, larger possibility of they being related with CMDs.

With the optimizing network architecture (input node, hidden nodes and connections) it was possible to establish which variables presented greater probability of being related with the studied problem. That is, the applied methodology in the experiment, is presented as an interesting alternative for problems application when the purpose is the identification of factors related to the variable response (network output).

Initially, experiments were accomplished with other neural networks, for example RBF (Radial Basis Function), however the obtained results were not satisfactory when compared to the results obtained in experiments accomplished with a MPL network. Besides, experiments were also accomplished with a MPL network trained with the Backpropagation algorithm, however, the obtained results were inferior to the results presented by the MLP network trained with the PSO–PSO:WD algorithm.

From the results presented in this work, we may conclude that a trained MLP network with the PSO–PSO:WD algorithm, with simultaneous optimizing of its architecture and weights, may be an interesting alternative to the statistic model of logistic regression, to the analysis of the factors related with the CMDs, because the neural network is able to detect all the possible interactions among the many explaining variables.

It is possible to observe also that, this automatic and simultaneous optimizing of the network architecture may be an interesting alternative for the process of variables selection, eliminating from the network input set the variable that is not so important to the problem. The permanence of variables that are not important to the patterns classification may aggravate distortions in the result presented by the network [12].

As future work possibility we pointed out: (1) to measure the performance of new experiments, applying the same methodology, with the use of others data set; (2) the use of other optimizing techniques, as for example Group Search Optimization; (3) the fusion of the two interleaved tasks (inner PSO and outer PSO) in a single PSO searching for weights and architectures as a truly simultaneous optimization process; (4) the addition of a connectivity pattern optimization process to the PSO–PSO algorithm.

Acknowledgments

The authors thank CNPq, CAPES and FACEPE (Brazilian Research Agencies) for their financial support.

References

Babu, G. S., & Suresh, S. (2013). Parkinsons disease prediction using gene expression – a projection based learning meta-cognitive neural classifier approach. *Expert Systems with Applications, 40*(5), 1519–1529.

Carvalho, M., & Ludermir, T. B. (2006). Particle swarm optimization of feed-forward neural networks with weight decay. In N. Kasabov, M. Köppen, A. König, A. Abraham, & Q. Song (Eds.), *HIS* (pp. 5). IEEE Computer Society.

Chernbumroong, S., Cang, S., Atkins, A., & Yu, H. (2013). Elderly activities recognition and classification for applications in assisted living. *Expert Systems with Applications, 40*(5), 1662–1674.

Eiben, E., & Smith, J. E. (2003). *Introduction to evolutionary computing*. Springer.

Glover, F. (1989). Tabu search – Part I. *INFORMS Journal on Computing, 1*(3), 190–206.

He, S., Wu, Q. H., & Saunders, J. R. (2009). Group search optimizer: An optimization algorithm inspired by animal searching behavior. *IEEE Transactions on Evolutionary Computation, 13*(5), 973–990.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of the IEEE international conference on neural networks* (Vol. 4, pp. 1942–1948). .

Ludermir, A., & Lewis, G. (2003). Informal work and common mental disorders. *Social Psychiatry and Psychiatric Epidemiology, 38*(9), 485–489.

Marcano-Cedeño, A., Chausa, P., García, A., Cáceres, C., Tormos, J. M., & Gómez, E. J. (2013). Data mining applied to the cognitive rehabilitation of patients with acquired brain injury. *Expert Systems with Applications, 40*(4), 1054–1060.

Prechelt, L. (1996). A quantitative study of experimental evaluations of neural network learning algorithms: Current research practice. *Neural Networks, 9*(3), 457–462.

Shi, Y., & Eberhart, R. C. (1998). Parameter selection in particle swarm optimization. In V. W. Porto, N. Saravanan, D. E. Waagen, & A. E. Eiben (Eds.), *Evolutionary programming. Lecture notes in computer science* (Vol. 1447, pp. 591–600). Springer.

Storn, R. (1999). System design by constraint adaptation and differential evolution. *IEEE Transactions on Evolutionary Computation, 3*(1), 22–34.

Zanchettin, C., Ludermir, T. B., & Almeida, L. M. (2011). Hybrid training method for mlp: Optimization of architecture and training. *IEEE Transactions on Systems, Man, and Cybernetics, Part B, 41*(4), 1097–1109.

Zhang, C., & Shao, H. (2000). An anns evolved by a new evolutionary systems and its application. In *IEEE conference on decision control* (pp. 3562–3563).