

Breast Cancer Diagnostic Factors Elimination via Evolutionary Neural Network Pruning

Adam V. Adamopoulos
*Democritus University of Thrace, Department of Medicine,
Medical Physics Laboratory, 681 00, Alexandroupolis, Hellas
adam@med.duth.gr*

Abstract

The redundant medical diagnostic factors elimination problem was faced by the use of Artificial Neural Networks that were evolved using Genetic Algorithms. For specific medical diagnosis problems such as breast cancer classification, with given set of diagnostic input parameters, Genetic Algorithms were used for pruning Neural Network structure and the investigation of the most appropriate subset of input parameters of Artificial Neural Networks that can still provide reliable medical diagnosis. Neural Networks were pruned in both the input as well as the hidden layer(s) by Genetic Algorithms that were utilized to search for pruned Neural Networks with the same, or even improved performance and therefore with enhanced medical classification and diagnostic ability of the original full-sized Neural Networks. Neural Network pruning and size reduction without loss of diagnostic ability can support redundant medical diagnostic factors or parameters elimination.

Key-Words: - Artificial Neural Networks Pruning, Genetic Algorithm Optimization, Breast Cancer Classification

1 Introduction

Reducing Artificial Neural Network (ANN) structural and functional complexity without loosing in terms of performance and prediction ability is one of the most interesting problems in the field of ANN [1-4]. This stands despite the nowadays development of even smaller and even faster computers, due to the fact that smaller size and simpler structure combined with high and speed-up performance still remains desirable. The problem of size reduction becomes even more important, if not crucial, in countless cases that hardware implementation of ANN is not only desirable but imperative if not compulsory. However, in the present work the question of ANN pruning and size reduction is approached in a quite different way and for a different purpose. Namely, in the present work, ANN pruning is utilized for the detection of any redundant medical diagnostic factors and is directly associated with the elimination of these diagnostic factors. For a given medical diagnosis problem this search is performed by training ANN initially with the whole set of the diagnostic factors and marking the performance of the ANN in classification and / or medical diagnosis. Afterwards, pruned

ANN were trained with subsets of the original set of diagnostic factors and their performance in medical classification and / or diagnosis were marked too. By comparing the performance of the pruned ANN to the corresponding performance of the full-sized ones, it was able to detect pruned ANN with classification and diagnostic ability identical, if not even improved, to the one of the full-sized ANN. These pruned ANN provide the same, or enhanced, diagnostic performance as the full-sized ones, despite the fact that they were trained with a subset of the original set of diagnostic factors, in other words with a smaller number of input parameters and therefore with a smaller number of neural nodes in the input layer and subsequently with smaller number of neural nodes in the hidden layer(s). The number, as well as, the selection of the members of the subsets of the input parameters that are used for the training of the ANN are evolved by utilizing Genetic Algorithms (GA) search in order to find the minimum number of input parameters and neural nodes in the hidden layer(s) and the specific subsets of input parameters that applied on the training if the ANN result to neural structures without any compromise in network's diagnostic performance. Thus the

method can provide pruned ANN that are structurally simpler but functionally equivalent to the full-sized ANN. Moreover, the obtained results can support the case that is the main concept of the present work, which is the suggestion that the diagnostic factors that are not utilized as inputs for the training of the pruned ANN can be considered as redundant ones and therefore can be eliminated without any compromise in terms of medical classification and diagnostic ability obtained from the pruned ANN.

2. Material

The material that was used in the present work was derived from the internet site of University of California at Irvine (UCI) Machine Learning Data Repository [5]. Namely, the file *wdbc.data* was downloaded from that site. The downloaded file contains medical data concerning breast cancer classification cases that were categorized by medical experts to malignant or benign.

The file *wdbc.data* contains features that describe characteristics of the cell nuclei of a fine needle aspirate (FNA) of a breast mass [6]. For each cell nuclei there are provided 30 real-valued features [7], which refer to three different values of each of the following diagnostic parameters:

1. radius (mean of distances from centre to points on the perimeter)
2. texture (standard deviation of grey-scale values)
3. perimeter
4. area
5. smoothness (local variation in radius lengths)
6. compactness ($\text{perimeter}^2 / \text{area} - 1.0$)
7. concavity (severity of concave portions of the contour)
8. concave points (number of concave portions of the contour)
9. symmetry
10. fractal dimension ("coastline approximation" - 1)

The three values that are provided for each one characteristic refer to the mean, standard error and "worst" values. Therefore the file contains in total the measured values of 30 different parameters, for a number of 569

cases of breast cancer. These 569 cases were classified by medical experts to 357 (62.75%) benign and 212 (37.25%) malignant [8-11]. The result of medical classification to benign or malignant is provided as an extra feature in the downloaded data file with an ID code particularly for each evaluated case.

3. Methods

For the analysis of the aforementioned material there were used Evolutionary Artificial Neural Networks, that is, specific computer programs were developed, which combine the technology of ANN to that of GA. The main purpose of the present work was to search for pruned ANN that perform at the same high level to the full-sized ones. This problem arises two questions. The first question is related to the minimization of the number of inputs of the ANN that are necessary for the appropriate network training. This question leads directly to the second question of the problem. Given that in the general case the number of network inputs is smaller than the number of parameters that are contained in the data sets that are used for the training of the network, a method for the investigation for the suitable subset(s) of the features that can be used for network training is essential. For example, in the data analyzed in the present work (*wdbc.data*) the total number of parameters is 30. Therefore, the number of possible combinations of these features that can consist subsets of training input data subsets as a function of the size k of the subset, with k to represent the number of used input parameters, is given by $n!/k!(n-k)!$. A plot of the number of possible training subsets for $n = 30$ is given in Fig. 1. The total sum of all the possible combinations for $k = 1 \dots 30$ is of the order of 10^9 , therefore an exhaustive search is out of question. Thus, the use of an intelligent and fast converging searching algorithm such as GA search [12, 13] is essential.

To accomplish the task of GA search a specific algorithm was developed in the Matlab® programming environment [14]. The algorithm combined the Neural Network Toolbox and the Genetic Algorithms and Direct Search Toolbox

(GADS). Using the GADS Toolbox there was developed a GA that evolved a population of individuals with binary codification. The chromosome length of each individual was taken equal to the number of parameters that consistent the training data sets (30 parameters in the case of the *wdbc.data* file), so that each gene to correspond to a feature and to represent the diagnostic factor related to that parameter. So, each gene represented a binary digit which was set to 0 if the corresponding parameter was not considered as input during the ANN training process, otherwise it was set to 1 in the case that the corresponding parameter was considered as input during the ANN training process. Thus, each gene codified the information if the corresponding parameter was considered for ANN training or it was omitted.

Using binary codification in the GA the sum of all the gene values of a chromosome provided the number of inputs and the specific subset of features that should be used for the training of the corresponding

ANN. Subsequently, a number (equal to the population of the GA) of pruned ANN were constructed using the Matlab® Neural Network Toolbox. The number of input nodes of each pruned ANN was taken in accordance to the number of the specific inputs that would be used to train it. Additionally, for each pruned ANN, a subset of the original training data set was constructed, containing only the data values of the parameters that were included for the training of the pruned ANN in accordance to the chromosome's individual information that was carried for the specific pruned ANN. As a next step the constructed pruned ANN were trained for a sufficient number of epochs and the resultant trained ANN were tested against the task of classification of a number of cases of cancer as malignant or benign type. The performance of the pruned ANN in terms of right classification of several cases in malignant or benign and therefore their ability to provide reliable medical diagnosis was compared to the corresponding results obtained by the training and testing of the full-sized ANN.

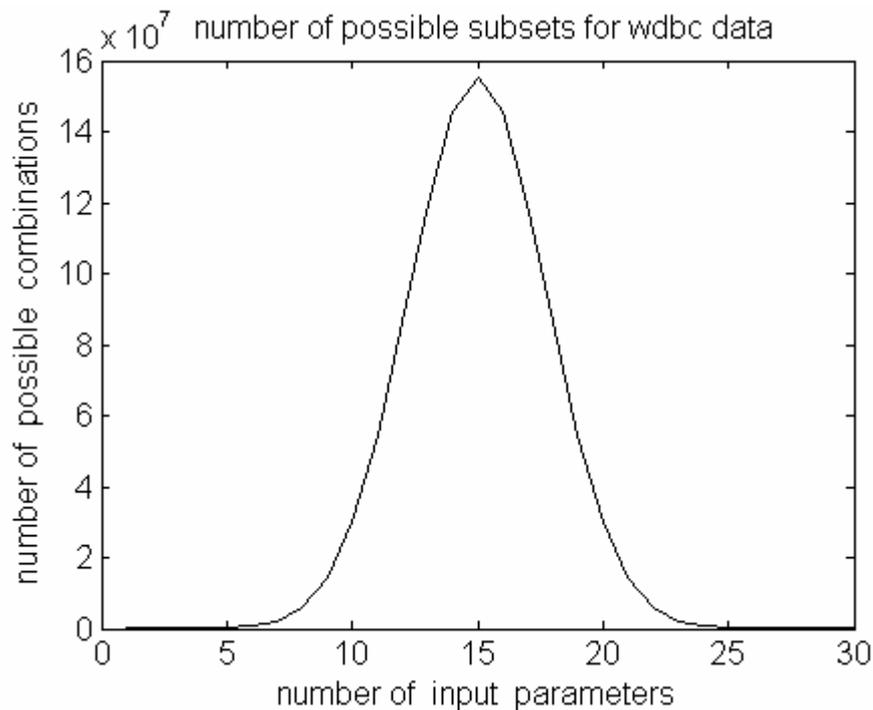


Figure 1. Number of possible combinations of training parameters

Two different GA fitness functions were used in computer experiments for the evaluation of the evolved ANN. The first fitness function (f_1) of the GA is given by the formulae:

$$f_1 = MSE \quad (1)$$

where MSE is the mean square error of the output of the ANN. Considering that Matlab® GADS Toolbox performs optimization by investigating the search space towards minimal fitness values, it is obvious that using f_1 a search for the minimum MSE is performed, therefore the algorithm will result to ANN with the best diagnostic performance.

To consider the issue of the reduction of the structural characteristics of the ANN an additional term was introduced in the fitness function of Eq. (1), resulting to the second fitness function f_2 as follows:

$$f_2 = MSE + I/N \quad (2)$$

where, I is the number of input nodes of the ANN and N is the total number of parameters in the full-sized original training data set (which is 30 in the case of wdbc data file). In all computer experiments, the number of neural nodes in the hidden layer was equal to the number of input nodes I . Obviously, f_2 is optimized for ANN that provide small MSE at the same time that their structure (number of nodes in the input and the hidden layer) is reduced. Thus, by selecting the fitness function of Eq. (2), the GA is forced to search for ANN that combine optimal diagnostic performance at the minimum ANN size available, that is, it is performed optimization in both ANN terms, functional and structural.

Due to fact that binary representation was chosen for the codification of the chromosomes of the GA individuals, all the well-known binary crossover operators (single-point crossover, two-point crossover, uniform crossover, scattered

crossover) and mutation operators (gaussian mutation, uniform mutation) which are provided by the Matlab® GADS Toolbox were utilized by computer experiments.

A number of computer experiments were performed for the wdbc.data. For each experiment the GA was left to run for a sufficient number of generations and the fitness values of the GA individuals as well as the average fitness vale of each generation were recorded.

4. Results

According to [7, 10, 11] the wdbc data are linearly separable, therefore perceptrons can be used for classification [15]. As a first experiment, the whole data set was used including all the 30 parameters it contains for the training and testing of the (full-sized) perceptrons. From the total of 569 patterns, 400 were used for training and the rest 169 were used for testing. Consecutive applications were performed using different number of training passes of the patterns during the training phase. The results presented in Table 1 refer to the average MSE of 10 independent executions of the algorithm. In each particular execution, the number of training patterns remained constant (400), but the particular training patterns were chosen randomly from the pool of the 569 patterns in total.

Table 1. Full-sized perceptron results for wdbc data

| Training Passes | mean MSE over 10 experiments | standard deviation of MSE |
|-----------------|------------------------------|---------------------------|
| 10 | 0.4432 | 0.1264 |
| 50 | 0.3479 | 0.1506 |
| 100 | 0.4100 | 0.1536 |
| 200 | 0.1121 | 0.0349 |
| 300 | 0.1174 | 0.0314 |
| 400 | 0.1111 | 0.0525 |
| 500 | 0.1242 | 0.0544 |
| 600 | 0.1042 | 0.0255 |
| 1000 | 0.1026 | 0.0259 |
| 2000 | 0.0932 | 0.0255 |

In each experiment, the rest patterns which were not used during the training phase were used for testing the resulted ANN afterwards the training and learning process was completed. The results shown in Table 1 provide a baseline for comparison with the corresponding findings of the method that is proposed in the present work.

Results provided by the application of the GA search using the fitness function f_1 of Eq. (1) are summarized in Table 2. In the second column in Table 2 are presented the MSE obtained from the testing procedure of the ANN that were generated by the GA. These findings are to be directly compared with the corresponding ones of Table 1 which refer to the obtained results of the full-size training and therefore can be used as the baseline. Comparison of the obtained mean MSE presented in the 2nd column of Table 1 (baseline) to the corresponding ones in the 2nd column of Table 2, clearly indicates that the GA search yields ANN with considerably smaller MSE in all the examined cases. It is noteworthy to refer that even in cases with

notably smaller subsets (33% to 63%) of parameters that were used as inputs, the GA achieved to find ANN that result to significantly smaller MSE.

However, the search for ANN with even simpler structure can be extended even further with the use of the fitness function f_2 of Eq. (2). In fitness function f_2 the term I/N explicitly introduces the intention to investigate for even smaller number of inputs, therefore for even fewer number of diagnostic parameters that will be used for the training and testing of the ANN. The obtained results of the GA experiments on the wdbc data using f_2 as fitness function are presented in Table 3. The results presented in the 3rd column of Table 3 refer to the MSE that obtained from the pruned ANN during the testing procedure and must be directly compared to the corresponding ones of the 2nd column in Table 1 and to the 2nd column in Table 3.

An additional comparison can be performed for the obtained results concerning the number of inputs that were used for the training and testing of the ANN.

Table 2. GA results using fitness function f_1 on wdbc data

| passes | $f_1 = MSE$ | Number of inputs (I) | best individual's chromosome |
|--------|-------------|--------------------------|--------------------------------|
| 10 | 0.12426 | 10 | 110000000010110010110000010001 |
| 50 | 0.11242 | 14 | 000101000011110111000110101001 |
| 100 | 0.08876 | 16 | 111110011010011101000111000100 |
| 200 | 0.07692 | 17 | 101100110101010110001111001011 |
| 300 | 0.06509 | 17 | 001101111000111000111011100011 |
| 400 | 0.05917 | 13 | 011010010011011000001101100100 |
| 500 | 0.05325 | 19 | 101000101011011110111111100110 |

Table 3. GA results using fitness function f_2 on wdbc data

| passes | $f_2 = MSE + I/N$ | MSE | I | best individual's chromosome |
|--------|-------------------|---------|-----|----------------------------------|
| 10 | 0.29685 | 0.13018 | 5 | 100010000000010000000000110000 |
| 50 | 0.34576 | 0.11243 | 7 | 100000010000000101001000000110 |
| 100 | 0.15917 | 0.05917 | 3 | 000000100000000010000000000100 |
| 200 | 0.18284 | 0.05325 | 5 | 000000100010000001010000000010 |
| 300 | 0.26509 | 0.06509 | 6 | 101000000001000010000101000000 |
| 400 | 0.22209 | 0.06508 | 2 | 000000010100000000000000000000 |
| 500 | 0.15917 | 0.05918 | 3 | 00100000000000000000000100100000 |

Results concerning the number of input parameters used during ANN training and testing are presented in the 3rd column of Table 2 and in the 4th column of Table 3. In all cases, computer experiments using GA with f_2 as fitness function converged to significantly simpler ANN structures, which in the half of the cases seem to need 10% to 20% of the features that are included in the original, full-sized data set, without the smaller compromise in classification ability of the cases and therefore the reliable diagnostic ability.

For purpose of comparison, the resulted *MSE* that was obtained from the testing of the ANN for the original data set shown in Table 1, as well as the genetically evolved ANN with the first fitness function (f_1) shown in Table 2, and the genetically pruned ANN shown in Table 3 obtained by the use of fitness function f_2 are summarized in Fig. 2. The results summarized in Fig. 2 indicate that the GA search converged to pruned ANN that use only the 10% - 20% of the original training data sets and can achieve even better diagnostic performance compared to the obtained one of the full-sized ANN.

5. Discussion

The main task in the present work was the investigation of the possibility of reduction of the number of diagnostic parameters that can

be used as input to ANN training and testing provided that any proposed decrease of the input parameters under consideration would not affect the diagnostic performance and the medical classification ability of the resulted ANN. Thus, the problem under examination was converted to the investigation of the most suitable combination of diagnostic parameters that should be considered for reliable diagnosis. To accomplish this task a novel technique was proposed which exploited the ability of GA for fast search and convergence over a huge search space [12, 13]. In the present work the proposed methodology was tested on an important medical diagnostic problem, the breast cancer classification. Considering that for the classification of breast cancer a number of 30 different parameters are validated, the number of all the possible combinations is of the order of 10^9 , therefore an exhaustive search is out of question. Thus, a more efficient and intelligent search method such as the GA is eligible. The GA utilized in the present work used individuals consisting of 30 binary digits (genes). Each binary gene corresponded to a diagnostic parameter, with the allele 1 to denote that the corresponding parameter was considered as an input during ANN training and testing, whereas the allele 0 denoted that the corresponding parameter was omitted during ANN training and testing.

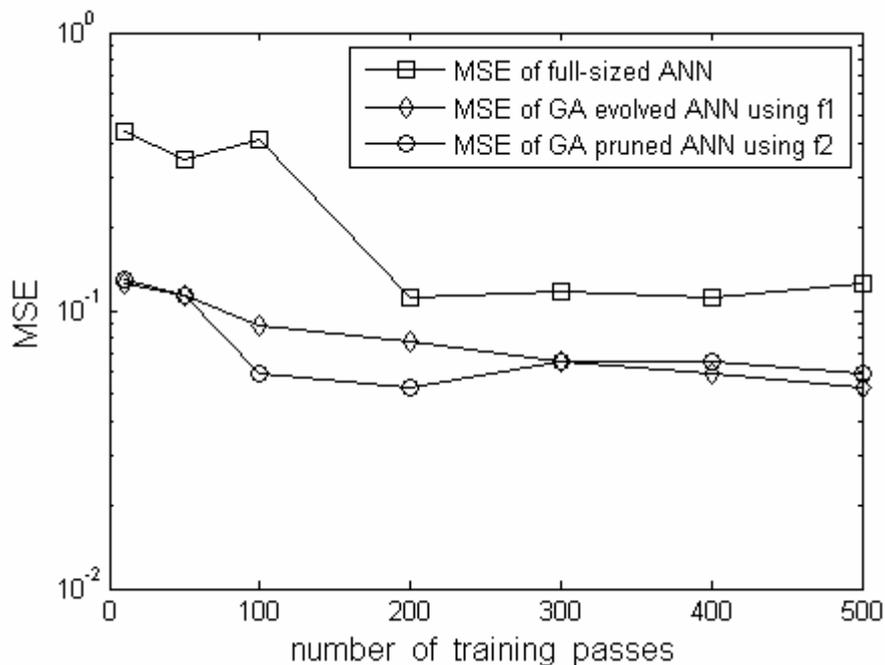


Figure 2. MSE for full-sized, GA evolved and GA pruned ANN

Two fitness functions, presented in Eq. (1) and Eq. (2), were introduced. The first fitness function f_1 , drives for a GA search for pruned ANN that are able to use a subset of the original full-sized set of the 30 input parameters during the training phase, while trying to minimize the MSE these ANN obtain in the testing phase. The obtained results presented in Table 2, shown that the GA search managed to reveal combinations of inputs that contain even the half of the original 30 parameters, while the classification performance of the ANN that were trained using these training combinations of parameters was improved in all the examined cases. These findings predicate that there is high level of redundancy in the original full-sized breast cancer diagnosis data.

To examine the problem more aggressively, a second fitness function f_2 , was introduced to the GA, which included explicitly an additional term that was directly related to the size of the input parameters subsets. Using f_2 as fitness function, the GA performed search for pruned ANN, while trying to minimize not only the MSE that these ANN generated at the testing phase, but the number of the input parameters that were considered during ANN training as well.

The obtained results, presented in Table 3, shown that a very small number of the original 30 input parameters (namely, 2 or 3 of them) is enough for the training of the ANN, since as it is shown in Table 3, combinations of 2 or 3 appropriately chosen input parameters can perform efficient ANN training so that in all the examined cases, these ANN generate even smaller MSE than the full-parameter trained ANN. These results are pictorially presented in Fig. 2, indicating that the rest 27 or 28 input parameters of the original full-sized data set are redundant, therefore they can be omitted with no loss in classification and diagnostic ability of the ANN.

The method proposed in the present work for the reduction of the number of the parameters that can be considered for breast cancer classification and diagnosis and the elimination of some of the input parameters is of general purpose. Therefore it can be applied in any data set used for ANN training and testing in order to reveal data redundancy, if

any. In future work we intend to apply the proposed method in various data sets of medical as well as, of non-medical interest.

6. References

- [1] A. Adamopoulos, G. Georgopoulos, S. Likothanassis and P. Anninos, "Forecasting the MagnetoEncephaloGram (MEG) of Epileptic Patient Using Genetically Optimized Neural Networks", Genetic and Evolutionary Computation Conference (GECCO'99), Orlando, Florida USA, July 14-17, 1999.
- [2] Likothanassis S., Georgopoulos E., Fotakis D., "Optimizing the Structure of Neural Networks Using Evolution Techniques", in: 5th International Conference on Applications of High - Performance Computers in Engineering, Santiago de Compostela, Spain (1997).
- [3] Georgopoulos E., Likothanassis S. and Adamopoulos A., "Evolving Artificial Neural Networks Using Genetic Algorithms", Neural Network World, 4, 2000, pp. 565 – 574.
- [4] E. Georgopoulos, A. Adamopoulos and S. Likothanassis, "Human MCG Modelling Using Evolutionary Artificial Neural Networks", 1st IEEE Symposium on Combinations of Evolutionary Computation and Neural Networks, in the 9th IEEE International Conference on Fuzzy Systems, San Antonio, Texas, USA, May 11-12, 2000.
- [5] <http://www.ics.uci.edu/~mllearn/MLRepository.html>; also <http://www.cs.wisc.edu/~olvi/uwmp/mpml.html> and <http://www.cs.wisc.edu/~olvi/uwmp/cancer.html>
- [6] W.N. Street, W.H. Wolberg and O.L. Mangasarian, "Nuclear feature extraction for breast tumor diagnosis", IS&T/SPIE 1993 International Symposium on Electronic Imaging: Science and Technology, Vol. 1905, San Jose, CA, 1993, pp. 861-870.
- [7] O.L. Mangasarian, W.N. Street and W.H. Wolberg, "Breast cancer diagnosis and prognosis via linear programming", Operations Research, Vol. 43(4), 1995, pp.570-577.
- [8] W.H. Wolberg, W.N. Street, and O.L. Mangasarian, "Machine learning techniques to diagnose breast cancer from fine-needle aspirates", Cancer Letters, Vol. 77, 1994, pp. 163-171.
- [9] W.H. Wolberg, W.N. Street, and O.L. Mangasarian, "Image analysis and machine learning applied to breast cancer diagnosis and prognosis", Analytical and Quantitative Cytology and Histology, Vol. 17, No. 2, 1995, pp. 77-87.
- [10] W.H. Wolberg, W.N. Street, D.M. Heisey, and O.L. Mangasarian, "Computerized breast cancer diagnosis and prognosis from fine needle aspirates", Archives of Surgery, Vol. 130, pp. 511-516.

- [11] W.H. Wolberg, W.N. Street, D.M. Heisey, and O.L. Mangasarian, "Computer-derived nuclear features distinguish malignant from benign breast cytology", Human Pathology, Vol. 26, 1995, pp. 792-796.
- [12] Goldberg D., Genetic Algorithms in Search Optimization & Machine Learning, Addison-Wesley, 1989.
- [13] Michalewicz Z., Genetic Algorithms + Data Structures = Evolution Programs, Springer-Verlag, 1996.
- [14] www.mathworks.com
- [15] Haykin S., Neural Networks - A Comprehensive Foundation, McMillan College Publishing Company, New York, 1994.