Classification Methods using Neural Networks and Partial Precedence Algorithms for Differential Medical Diagnosis: A Case Study

Angel Fernando Kuri-Morales[1], Martha R. Ortiz-Posadas[2]

[1] Instituto Tecnológico Autónomo de México
Río Hondo No. 1, México D.F.
akuri@itam.mx
[2] Universidad Autónoma Metropolitana-Iztapalapa
México D.F.
posa@xanum.uam.mx

Abstract. The problem of correctly diagnosing different types of ailments has been tackled with different artificial intelligence techniques since its early inception. Both heuristic and statistically based algorithms have been discussed in the past. In this paper we establish a comparison between one heuristic algorithm based on partial precedence and majority decision rules and two types of statistical ones: multi-layer perceptrons (MLP) and self-orgainizing maps (SOMs) when applied to the automated diagnosis and treatment of cleft lip and palate. We show that although all three methods perform reasonably well (with efficiency ratios better than 0.9) the neural networks achieve their goals with a considerably diminished set of data without detriment in their performance. Furthermore, we are able to tackle an enlarged set and still retain the high yields with the use of MLPs and SOMs.

1 Introduction

Partial precedence algorithms (PPAs) were developed in an attempt to properly study the characteristics which make a problem amenable to characterization in a systematic way. A basic idea was to show the usefulness of the logical combinatory approach in pattern recognition for developing auxiliary criteria for differential medical diagnosis [1]. This approach gives rise to a method based on a simple heuristic which was shown to correctly achieve the classification of patients with cleft lip and palate [2]. On the other hand, statistical learning has been successfully implemented in various schemes which achieve their results by iteratively refining an initially coarse result until an acceptable one is reached by distributing the problem between a series of simple but densely connected processors which, in the literature, have been termed "neurons". The so-called neural networks (NNs) have been shown to perform adequately in a surprisingly varied set of problems. When attempting classification, one may use NNs representing both supervised and non-supervised learning schemes. Two possible and popular alternatives are the multi-layer percepton networks and the self-organizing maps. In this paper we review the results of partial precedence

algorithms, compare the results stemming from its application both with MLPs and SOMs and show that both types of NNs achieve comparable or better results than PPAs. Furthermore, by applying a simple correlation analysis we are able to eliminate close to 90% of the parameters involved without impairing the classification abilities of the NNs. The rest of the paper is organized as follows. In section 2 we give a basic description of the problem whose diagnosis is to be automated. In section 3 we discuss the basic idea behind the three methods to be compared: PPAs, MLPs and SOMs. In section 4 we discuss the results gotten in the past by one of us (Ortiz) by using PPAs. In section 5 we describe the results derived from, both, MLPs and SOMs. In section 6 we offer our conclusions.

2 The Clinical Problem

The clinical problem consists of congenital malformations in the lip and/or palate, which are called *cleft-primary palate and/or cleft-secondary palate*, respectively. Surgical complexity for cleft reconstruction will depend on cleft complexity involving lip, nose and/or palate. Cleft correction translates into a very slow and complex process because it is related to the growth and development of the patient, and it requires at least one surgical procedure. The importance of prognosis of the patient's rehabilitation, and subsequent evaluation of the surgical result, is the physician's self-feedback during all the rehabilitation process. The physician will learn if his/her work patient rehabilitation is adequate, or if it can be improved.

3 Partial Precedence , MLPs and SOMs

In order to describe the type of cleft it was necessary to define eighteen variables for initial description of the patient: two for palate, nine for lip, and seven for nose [3]. The patients are then classified as Excellent (E), Very Good (VG) or Good (G) depending on the values assigned to these variables. Thereafter, an algorithm is applied to the set of known values and their corresponding prognosis in an attempt to extract general rules for future use by the physician.

3.1 Partial Precedence Algorithm

The algorithms based in *precedence* allow the analysis of partial likelihood, relating parts of the description of the objects with some class. Once this is achieved, a search for full likelihood (taking into consideration the full description of the object) allows us to reach a final classification decision. The algorithm of classification of partial precedence which is a majority (vote) algorithm is defined in six stages, described in what follows.

1) *Definition of the system of support sets.* These are non-empty subsets of the set of variables, whose purpose is to define the combinations of variables on which a partial ordering will be established. For the particular problem tackled in this

work, three support sets were defined: cleft, lip and nose.

2) *Definition of the likelihood function.* The likelihood between two clefts was formalized by a likelihood function $b_w(I(O), I(O_j))$, which was built from the comparison criteria discussed in [4].
3) *Evaluation of the likelihood for each object for a fixed support set*. This stage is the basic step for the majority decision, since in it the likelihood of the object O (to be classified) with each of the already classified objects O_j (whose descriptions make up the rows of the learning matrix) is determined. The likelihood is calculated for each support set ω. If we denote this likelihood as $\Gamma_\omega(O, O_j)$ then: $\Gamma_w(O, O_j) = b_w(I(O), I(O_j))$ where b_w is the *partial* likelihood function corresponding to every support set. For this study three partial likelihood functions were defined: b_{cleft}, b_{lip}, b_{nose} .
4) *Class majority for a fixed support set*. Here we define the way in which the votes corresponding to all objects belonging to the same class (K_i) retaining the support set (ω) are counted. Denoting by $\Gamma_w^i(O)$ the votes in this stage, we have:

$$\Gamma_w^i(O) = \frac{1}{|K_i|} \sum_{O_j \in K_i} \Gamma_w(O, O_j)$$

5) *Class majority for the full system of support sets*. In this stage vote counting is continued, but now considering all support sets. For each class (K_i) we compute the total majority $\Gamma_i(O)$, with $\Gamma_i(O) = \sum_{w \in \Omega} g_w \Gamma_w^i(O)$ where γ_ω is the weight of the support set ω.
6) *Rule of general solution for the classification of the object*. Here we define the way to decide, as a function of the majorities of the previous stage, the class where the object will be placed, i.e. its forecast.

The object O will be placed (forecast) in the class K_s for which a maximum vote is reached (Γ_i). If we denote with $\Gamma_E^{(O)}, \Gamma_{VG}^{(O)}, \Gamma_G^{(O)}, \Gamma_M^{(O)}$ y $\Gamma_L^{(O)}$, the final votes for the object O for each of the classes, then O is classified in class K_s if $\Gamma_s^O = max\{ \Gamma_E^{(O)}, ..., \Gamma_L^{(O)} \}$, $s \in \{E, VG, G, M, L\}$. If a maximum is reached in more than one class, O is placed in the one which represents a superior condition. Notice that in the preceding discussion we have considered 5 categories but in our final study only the three (E; VG, G) were considered.

3.2 MLPs

Multi-Layer perceptron networks have shown to be versatile and practical tools for classification purposes. It can be proved that MLPs are universal function approximators [5] and, hence, useful for our purposes here. Once the MLP is trained using an appropriate algorithm (we used a variation of the backpropagation algorithm) it may be thought of as a function, whose expression must include: a) The topology of the network, b) The transfer functions associated to each layer, c) the learning

parameters and d) The weights for every connection. In this work we consider only the classical feed-forward, strongly connected MLP [6].
There is ample literature discussing the ways in which MLPs should be designed in order to achieve the most efficient performance. The reader is referred to [7] and [8] where a detailed account of such criteria may be found. Here we define a network N (which has already been satisfactorily trained) as follows: $N = f(\boldsymbol{t}, \boldsymbol{l}, \boldsymbol{p}, \boldsymbol{w})$; where $\boldsymbol{t} = (n_1, n_2, n_3)$, $\boldsymbol{l} = (f_2, f_3)$, $\boldsymbol{p} = (\eta_1, \eta_2; \mu_1, \mu_2)$, $\boldsymbol{w} = (w_1, w_2, ..., w_m)$ where $m = n_1 n_2 + n_2 n_3 + n_2 + n_3$. Here n_i denotes the number of neurons in layer i; f_i denotes the transfer function for the i-th layer (we consider only three possible transfer functions: linear, sigmoid and hyperbolic tangent which we encode as 1, 2 or 3); and η_i and μ_i denote the learning rate and momentum of the i-th layer, respectively.

In this sense, a MLP is a vector in $\Re^{m+9}$. Notice that we are considering a) three layered networks (which have been shown to approximate adequately functions without discontinuities, as is the case here); b) linear functions on the presentation layer (and therefore only the transfer functions for layers 2 and 3 need to be specified); c) one learning and momentum parameter per layer; d) only one kind of function per layer; e) a bias neuron with unit input.

3.2.1 Training a MLP

The training procedure for a MLP is, presently, well established and understood. Basically, it consists of assigning an initial (usually random) set of values for $\boldsymbol{w}$. Then repeat the following procedure as needed: a) Evaluate the deviation of the desired outputs from the network defined as above from the desired outputs, b) Correct the weights (free parameters) in order to minimize the observed errors, c) Repeat this process for all samples (inputting all samples to a MLP is called "an epoch"), d) Repeat the process until a convergence criterion is met.

In our work we followed the best generalization criterion to stop the learning process. In it two data sets are defined: a training set (RS) and a test (TS) set. Usually RS > TS since we aim at incorporating as much knowledge about the system as possible. The MLP, thus, approximates the adequate values from RS but, since we do not want to over-train the network (which could learn "by heart" the samples in the data set) we stop when the calculated values for set TS pass the learning basin. This criteria has been called the *cross-validation* scheme and was initially discussed in [9].

3.3 SOMs

Self-organizing maps (also known as Kohonen networks) are an example of non-supervised learning. The "neurons" in a SOM are actually, elements which have a double vector: a) A vector in the (usually bidimensional) space of the map and b) A vector in the space of the features. In our case, every neuron has a bidimensional vector which allows for the identification of a neuron in the cartesian plane and a (in principle) vector in $\Re^{18}$; this last identifies a point in the space of the features of interest.

The training algorithm is well known and can be found in [10]. The reader is invited to see [11] for a full description of the training algorithm as well as of the labeling algorithm.

A SOM is described in simpler terms than a MLP. We need only to specify: a) The number of dimensions of the SOM, b) The number of neurons per dimension, c) The coordinates for each neuron and d) The class to which every neuron belongs. In figure 3 above, the point is well illustrated.

Therefore, we define a network N (which has already been satisfactorily trained) as follows: $N = f(d, n, w, k)$; where $d = \{1,2,3\}$, $n = n_1, n_2$, $m = n_1 n_2$ (here n_i denotes the number of neurons in dimension i) and $\boldsymbol{w} = \vec{w}_1, \vec{w}_2, ..., \vec{w}_m$.

4 Original Results using the PPA

The PPA was tested with a sample of 95 patients treated in Tacubaya's Pediatric Hospital, in Mexico City. Two matrices were established: a learning matrix and a control matrix. This was done randomly with a 1:2 ratio. The learning matrix was thus formed by data from 32 patients as follows: 10 in "E" class, 14 in "VG" class and 8 in "G" class. Likewise, the control matrix was formed with data for 63 patients: 19 in E, 29 in VG and 15 in G. Classification was attempted with data from the control matrix and the results are shown in table 1. Out of 19 patients in class E, 17 were correctly classified; the remaining 2 were assigned to class VG. Out of 29 patients in class VG, 26 were correctly placed, whereas 3 were assigned to class B. For those patients in class B, 14 were set in the proper class and only one was set in class VG. Overall, 57 patients were properly classified, yielding a 90.5% efficiency.

	E	VG	G	Total
EXCELENT (E)	17	2	0	19
VERY GOOD (VG)	0	26	3	29
GOOD (G)	0	1	14	15
Total	17	29	17	63

Table 1. Classification for PPA.

5 Results Using MLPs and SOMs

One of the main purposes of this work is to compare the three methods mentioned above. Although PPA has shown to perform satisfactorily, a statistically simple test pointed in a different direction. When analyzing the data we calculated a correlation

matrix which made clear that clinically sound parameters displayed a high correlation. Therefore, we were forced to reappraise the value of the set of variables.

5.1 Correlation Matrix

The correlation matrix is found by applying the Bravais-Pearson formula:

$$r = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2 \sum_{i=1}^{N}(y_i - \bar{y})^2}}$$

where an absolute value of r close to 1 indicates great correlation. Schematically, the correlation matrix we obtained may be represented as in figure 4.

	F1	F2	L1	L2	...	N3	N4	N5	N6	N7
F1					...					
F2					...					
L1					...					
L2					...					
L3					...					
...					...					
N3					...					
N4					...					
N5					...					
N6					...					
N7					...					

Fig. 1. Schematic Representation of the Correlation Matrix.

In Figure 4, the shadowed cells indicate a very high correlation. Therefore, we had evidence that many of the original variables displayed redundant information and were, therefore, unnecessary for classification purposes. This was a counter-intuitive result since such variables were determined from clinical and physiological considerations. However, we took the decision to retain only the uncorrelated variables and work with a reduced set consisting only of variables F1, F2, L1 and N7. In point of fact, we could have chosen any of the variables in {L1-L9, N1-N4} and any of the variables in {N5, N6, N7}. The choice of L1 and N7 was arbitrary.

5.2 MLPs

Using the reduced set, we were able to define a MLP with a (4:2:3) topology, analogous to the one illustrated in section 3.2. The trained MLPs performed as shown in tables 2 and 3 which display the results for the training and control matrices, respectively.

	E	VG	G	Total
EXCELENT (E)	9	2	0	11
VERY GOOD (VG)	0	13	0	13
GOOD (G)	0	0	8	8
Total	9	15	8	32

Table 2. Classification for Training Data (MLP1)

	E	VG	G	Total
EXCELENT (E)	17	2	0	19
VERY GOOD (VG)	0	29	0	29
GOOD (G)	0	2	13	15
Total	17	33	13	63

Table 3. Classification for Test Data (MLP2)

5.3 SOMS

Using the reduced set, we were likewise, able to define a SOM with a (4:4) topology, analogous to the one illustrated in section 3.3. The trained SOMs performed as shown in tables 4 and 5 which display the results for the training and control matrices, respectively.

	E	VG	G	Total
EXCELENT (E)	9	2	0	11
VERY GOOD (VG)	0	13	0	13
GOOD (G)	0	1	7	8
Total	9	16	7	32

Table 4. Classification for Training Data (SOM1)

	E	VG	G	Total
EXCELENT (E)	19	0	0	19
VERY GOOD (VG)	0	29	0	29
GOOD (G)	0	5	10	15
Total	19	34	10	63

Table 5. Classification for Test Data (SOM2)

A comparison for the PPA, MLP and SOM is shown in table 6. Notice that MLP1 and SOM1 were tested *versus* the training matrix and, therefore, comparable tests have been shadowed in the table. Remarkably, both MLP2 and SOM2 achieved better classification ratios than PPA even though only 4 variables were considered.

Algorithm	PPA	MLP1	MLP2	SOM1	SOM2
Efficiency (%)	90.48	93.75	93.65	90.62	92.06

Table 6. PPA, MLP and SOM compared.

5.4 Enhanced Learning Matrix

Even though because methodological considerations led to the definition of the learning matrix, originally, as the smaller of two in a 1:2 ratio, this process was revised in a second set of experiments. In these, the learning matrix consisted of 81 samples whereas the test matrix contained the remaining 14 ones. Using the same methods described above, we achieved the following results.

	E	VG	G	Total
EXCELENT (E)	22	1	0	23
VERY GOOD (VG)	0	39	0	39
GOOD (G)	0	1	18	19
Total	22	41	18	81

Table 7. Classification for Training Data (MLP1)

	E	VG	G	Total
EXCELENT (E)	6	0	0	6
VERY GOOD (VG)	0	4	0	4
GOOD (G)	0	1	3	4
Total	6	5	3	14

Table 8. Classification for Test Data (MLP2)

	E	VG	G	Total
EXCELENT (E)	21	2	0	23
VERY GOOD (VG)	0	37	2	39
GOOD (G)	0	4	15	19
Total	21	43	17	81

Table 9. Classification for Training Data (SOM1)

	E	VG	G	Total
EXCELENT (E)	5	1	0	6
VERY GOOD (VG)	0	3	1	4
GOOD (G)	0	2	2	4
Total	5	6	3	14

Table 10. Classification for Test Data (SOM2)

Algorithm	MLP1	MLP2	SOM1	SOM2
Efficiency (%)	97.53	92.86	88.89	71.43

Table 11. PPA, MLP and SOM compared.

The NNs performed satisfactorily. In particular, MLPs have shown to be insensitive to the large amount of data, and perform even better than in the reduced matrix. SOMs have not had such a good behavior. But the augmented matrix allows, from the standpoint of the already mentioned cross-validation strategy, a better and fuller generalization property. The NNs trained with a much larger learning matrix are expected to generalize with higher reliability when presented with data outside the known domain. This fact is of intereset in the sense that patients being controlled with the newer matrix will increase their rehabilitation ratios.

6 Conclusions

We have shown that both MLPs and SOMs show better performance than the PPA tested in the past. Particularly, the MLPs efficiency was always superior. This behavior is remarkable since more than 85% of the original data was shown to be unnecessary. Therefore, it is to be expected that the variables which determine the automated prognosis are followed more simply and efficiently. Furthermore, when attempting better generalization by augmenting the learning the SOMs suffered a relative setback, while MLPs retained their high yields. As usual, we have to stress the fact that neural networks have no explanatory properties and this is, perhaps, their only shortcoming.

We may safely state that NNs and, particularly, MLPs are resilient and reliable tools which, in this instance, will allow the physicians to partially automate and improve their work. Also, a new line of study opens, since these methods are susceptible to application on a greater scale. We expect to report on this enhanced possibility in the near future.

7 References

1. Ortiz-Posadas, M.R., Martínez-Trinidad, F., Ruíz Shulcloper, J.: "A new approach to differential diagnosis of diseases", *Int. J. Biomed. Comput*. **40** (1996) 179-185
2. Ortiz-Posadas, M.R., Almazán-Morales, J., Contreras-Ramos, J.: "A computational tool for the prognosis of the rehabilitation of patients with cleft palate", *Proc 5th Ibero-American Symposium on Pattern Recognition*. Lisbon, Portugal (2000) 599-608
3. Ortiz-Posadas, M.R., Vega-Alvarado, L., Maya-Behar, J.: "A new approach to classify cleft lip and palate", *Cleft Palate Craniofac J*. **38** (2001) 545-550
4. Lazo-Cortés, M., Ruiz-Slulcloper, J.: "Determining the feature relevance for non-classically described objects and a new algorithm to compute typical fuzzy testors", *Pattern Recognition Letters* **16** (1995) 1259-1265
5. Cybenko, G., "Approximation by superpositions of a sigmoidal function", *Mathematics of Control, Signals and Systems*, vol. 2, pp. 303-314.
6. Haykin, S., *Neural Networks: A comprehensive foundation*, pp. 156-255, Prentice-Hall, 2nd. Ed., 1999.
7. Gori, M., and A. Tesi, "On the problem of local minima in backpropagation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 76-86, 1992.
8. Hecht-Nielsen, R., "Replicator neural networks for universal optimum source coding", *Science*, Vol. 269, pp. 1860-1863, 1995.
9. Stone, M., "Cross-validatory choice and assessment of statistical predictions", *Journal of the Royal Statistical Society*, vol. B36, pp. 111-133, 1974.
10. Kohonen, T., "Correlation matrix memories", *IEEE Transactions on Computers*, vol. C-21, pp. 353-359, 1972.
11. Kohonen, T., E. Reuhkala, K. Makisara, and l: Vainio, "Associative recall of images", *Biological Cybernetics*, vol. 22, pp 159-168, 1976.