Combining Supervised and Unsupervised Learning for GIS Classification

Juan-Manuel Torres-Moreno[1], Laurent Bougrain[2], and Frédéric Alexandre[2]

[1] Laboratoire Informatique d'Avignon
Université d'Avignon et des Pays de Vaucluse
BP 1228 84911 Avignon Cedex 09, France
[2] Équipe Cortex INRIA/LORIA Campus Scientifique
BP 239 54506 Vandœuvre-lès-Nancy, Cedex, France
juan-manuel.torres@univ-avignon.fr

Abstract. This paper presents a new hybrid learning algorithm for unsupervised classification tasks. We combined Fuzzy c-means learning algorithm and a supervised version of Minimerror to develop a hybrid incremental strategy allowing unsupervised classifications. We applied this new approach to a real-world database in order to know if the information contained in unlabeled features of a Geographic Information System (GIS), allows to well classify it. Finally, we compared our results to a classical supervised classification obtained by a multilayer perceptron.

Keywords : Minimerror, Hybrid methods, Classification, Unsupervised learning, Geographic Information System.

1 Supervised and Unsupervised Learnings

For a classification task, the learning is supervised if the labels of the classes of the input patterns are given a priori by a professor. A cost function calculates the difference between desired and real outputs produced by a network, then, this difference is minimized modifying the network's weights by a learning rule. A supervised learning set $\mathcal{L}$ is constitued by P couples $(\boldsymbol{\xi}^\mu, \tau^\mu), \mu = 1, ..., P$, where $\boldsymbol{\xi}^\mu$ is the input pattern μ and $\tau^\mu = \pm 1$ its class. $\boldsymbol{\xi}^\mu$ is a N-dimension vector, with numeric or categoric values. If labels τ^μ are not present in $\mathcal{L}$, it may be used as unsupervised learning. Learning is unsupervised when the object's class is not known in advance. This learning is performed by extraction of intrinsic regularities of patterns presented to the network. The number of neurons of the output layer corresponds to the desired number of categories. Therefore, the network develops its own representation of input patterns, retaining the statistically redundant traits.

2 Supervised Minimerror

Minimerror algorithm [1] performs correctly in binary problems of high dimensionality [3, 4, 10]. The supervised version of Minimerror performs a binary classification using the minimization of the cost function:

$$E = \frac{1}{2} \sum_{\mu=1}^{P} V \left(\frac{\tau^{\mu} \boldsymbol{w} \cdot \boldsymbol{\xi}^{\mu}}{2T\sqrt{N}} \right) \tag{1}$$

with

$$V(x) = 1 - \tanh(x) \tag{2}$$

Temperature T defines an effective window width on both sides of the separating hyperplane defined by $\boldsymbol{w}$. The derivative $\frac{dV(x)}{dx}$ is vanishingly small outside this window. Therefore, if the minimum cost (1) is searched through a gradient descent, only the patterns μ at a

$$|\gamma^{\mu}| \equiv \frac{|\boldsymbol{w} \cdot \boldsymbol{\xi}^{\mu}|}{\sqrt{N}} < 2T \tag{3}$$

distance will contribute significantly to learning [1, 2]. Minimerror algorithm implements this minimization starting at high temperature. The weights are initialized with Hebb's rule, which is the minimum of (1) in the high temperature limit. Then, T is slowly decreased upon the successive iterations of the gradient descent by a deterministic annealing, so that only the patterns within the narrowing window of width 2T are effectively taken into account for calculating the correction

$$\delta \boldsymbol{w} = -\epsilon \, \frac{\partial E}{\partial \boldsymbol{w}} \tag{4}$$

at each time step, where ϵ is the learning rate. Thus, the search of the hyperplane becomes more and more local as the number of iterations increases. In practical implementations, it was found that convergence is considerably speeded-up if patterns already learned are considered at a lower temperature T_L than the not learned ones, $T_L < T$. Minimerror algorithm has three free parameters: the learning rate ϵ of the gradient descent, the temperature ratio T_L/T, and the annealing rate δT at which temperature is decreased. At convergence, a last minimization with $T_L = T$ is performed. This algorithm has been coupled with a incremental heuristics, NetLS [2,5], which adds neurons in one hidden layer as learning function. Several results [2–4] show that NetLS is very powerful and gives small generalization errors comparable to other methods.

3 Unsupervised Minimerror

A variation of Minimerror, Minimerror-S [2, 3], allows to obtain spherical separations on input's space. The spherical separation used the same cost function (1), but a spherical stability γ_s is defined by:

$$\gamma_s = ||\boldsymbol{w} - \boldsymbol{\xi}|| - \rho^2 \tag{5}$$

where ρ is a hyperspherical's radius centered on $\boldsymbol{w}$. The pattern's class is $\tau = -1$ inside the sphere and $\tau = 1$ elsewhere. Spherical separations make it possible to consider unsupervised learning using the Minimerror's separating qualities. Thus, a strategy of unsupervised growing was developed in Loria. The algorithm starts by obtaining the distances between the patterns. The Euclidean distance can be used to calculate them. Once the established distances, we started to find the pair μ and ν of patterns with the smallest distance ρ. This creates the first incremental kernel. We located the hypersphere's center $\boldsymbol{w_0}$ at the middle of patterns μ et ν:

$$\boldsymbol{w_0} = \frac{(\boldsymbol{\xi}^{\mu} + \boldsymbol{\xi}^{\nu})}{2} \tag{6}$$

The initial radius is fixed

$$\rho_0 = \frac{3\rho}{2} \tag{7}$$

to make enter a certain number of patterns in growing kernel. Then, patterns are labeled $\tau = -1$ if they are inside or in the border of the initial sphere, and $\tau = 1$ if elsewhere. Minimerror-S finds the hypersphere $\{\rho*, \boldsymbol{w}*\}$ that better separates patterns. The internal representations are $\sigma = -1$ if

$$-\frac{1}{\cosh^2(\gamma^{\mu})} < \frac{1}{2}$$

else $\sigma = 1$. This makes it possible to check if there are patterns with $\tau = 1$ outside but sufficiently close to the sphere $(\rho_1^*, \boldsymbol{w_1^*})$. In this case, then it makes $\tau = -1$ for these patterns and it learns them again, repeating the procedure for all patterns of $\mathcal{L}$. At this time, it passes to another growing kernel which will form a second class $\boldsymbol{w_2}$, calculating with Minimerror-S $(\rho_2^*, \boldsymbol{w_2^*})$, and repeating the procedure until there is no more patterns to classify. Finally it obtains K classes. A pruning procedure can avoid having too many classes by eliminating those with few elements (less than one number fixed in advance). It is possible to introduce conditions at the border, which are restrictions that prevent locating the hypersphere center outside of the input's space. For certain problems this strategy can be interesting. These restrictions are however optional: if it makes too many learning errors, the algorithm decides to neglect them and the center and radius of separating spheres can diverge.

4 The Unsupervised Algorithm Fuzzy c-means

This algorithm [6, 7] allows us to obtain a clusterisation of patterns with a fuzzy approach. Fuzzy c-means minimizes the sum of the squared errors with the following conditions:

$$\sum_{k=1}^{c} m_{ik} = 1; \sum_{i=1}^{n} m_{ik} > 0; m_{ik} \in 0, 1 \tag{8}$$

$$i = 1, 2, \ldots, n; k = 1, 2, \ldots, c \tag{9}$$

The objective function is defined by

$$J = \sum_{i=1}^{n} \sum_{k=1}^{c} m_{ik}^{\phi} d^2(\xi_i, c_k) \tag{10}$$

where n is the number of patterns, c is the desired number of classes, c_k is the centroid vector of class K, $\boldsymbol{\xi_i}$ is a pattern i and $d^2(\xi_i, c_k)$ is the square of the distance between patterns ξ_i and c_k, in agreement with a definition of unspecified distance, which to simplify, we will indicate by $d^2(\xi_i, c_k)$. ϕ is a fuzzy parameter, a value in $[2, \infty)$, which determines the fuzzyfication of the final solution, i.e., it controls the overlapping between the classes. If $\phi = 1$, the solution is a hard partition. If $\phi \rightarrow \infty$ the solution approaches the maximum of fuzzyfication and all the classes are likely to merge in only one. The minimization of the objective function J provides the solution for the membership function (6):

$$m_{ik} = \frac{d_{ik}^{2/\phi-1}}{\sum_{j=1}^{c} d_{ij}^{2/\phi-1}}; i = 1, \ldots, n; k = 1, \ldots, c; \tag{11}$$

where:

$$c_k = \frac{\sum_{i=1}^{n} m_{ik}^{\phi} x_i}{\sum_{i=1}^{n} m_{ik}^{\phi}}; k = 1, \ldots, c \tag{12}$$

The fuzzy c-means algorithm is:

1. Let the class number k, with $1 < k < n$.
2. Let a value of fuzzy parameter $f > 2$.
3. To choix a suitable distance definition in input's space. That may be euclidean distance and then $d^2(x_i, c_k) = ||x_i - c_k||^2$.
4. To choix a value for stop criterium ϵ ($\epsilon = 0.001$ is a suitable convergence).
5. Let $M = M_0$, for pattern with random values or with values from a hard partition of k-means.
6. In iteration $t = 1, 2, 3, ...$ (re) calculate $C = C_t$ using 12 and M_{t-1}.
7. Re-calculate $M = M_t$ using equation 10 and C_t.
8. To compare M_t and M_{t-1} with a suitable matrix norme. If $||M_t - M_{t-1}|| < \epsilon$ then stop else go to 6.

5 A Hybrid Strategy

In spite of the supervised Minimerror's simplicity, the number of classes obtained is sometimes too high. Thus, we chose a combined strategy: a first unsupervised hidden layer calculates the centroids with Fuzzy c-means algorithm. As input we have P unlabeled patterns of learning set $\mathcal{L}$. Then Supervised Minimerror finds spherical separations well adapted to maximize the stability of the patterns. The input is the same $\mathcal{L}$ set, but labeled by Fuzzy c-means. In this way, the number of classes can be selected in advance.

6 Deposit Prospection Experiment

The mineral resources division of the French geological survey (BRGM [8]) develops continent-scale Geographic Information System (GIS), which support metallogenic research. This difficult real-world problem constitutes a tool for decision making. The understanding of the formation of metals such as gold, copper or silver is not good enough and a lot of patterns describing a site are available including the size of the deposit for various metals. In this study, we will focus on a GIS which covers all the Andes and two classes : deposit and barren. A deposit is an economically exploitable mineral concentration [9]. The concentration factor corresponds to the rate of enrichment in a chemical element, i.e. to the relationship between its average content of exploitation and its abundance in the earth's crust. Geologists oppose to the concept of deposit the one of barren. Actually, for the interpretation of the results of generalization, it is necessary to enter the number of sites well classified in each category to be able to answer the question: Is this a deposit or a barren ? In our study, a deposit will be defined as a site (represented by a pattern) that contains at least one metal and a barren by a site without any metal. Then, the classes *deposit* and *barren* will be used from now on. The database we used contains 641 patterns, 398 examples of deposits and 343 examples of barrens.

6.1 Study of the Attributes

The original databases have 25 attributes, 8 qualitative and 17 quantitative, such as the position of a deposit, the type and age of the country rock hosting the deposit, the proximity of the deposit to a fault zone distinguished by its orientation in map view, density and focal depth of earthquakes immediately below the deposit, proximity of active volcanoes, geometry of the subduction zone etc. We made a statistical study to determine the importance of each variable. We calculated for each attribute the average of *deposit* and *barren* patterns, in order to determine which attributes were relevant for discriminating the patterns (figure 1). There are some attributes (15, 16, 17 or 22, among others) that are not relevant. On the other hand, the attributes 3, 5, 6 and 25 are rather discriminating. It is interesting to know how the choice of attributes influences the learning and specially the generalization tasks. Therefore, we created 11 databases with different combinations of attributes. Table 1 shows the number of qualitative and quantitative attributes, and the dimension for each database used.

6.2 Data Preprocessing and *deposit/barren* Approach

The range of the attributes is extremely broad. In order to homogenize them, a standardization of quantitative attributes is suitable. A data preprocessing is needed for the correct functioning of the neural network. Thus, for each continuous variable, the standardization calculates the average and standard deviation. Then, the variable was centered and the values divided by the standard deviation. The qualitative attributes are not modified. The standardized corpus was

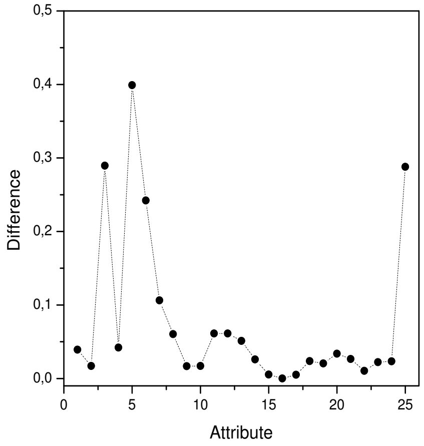

Fig. 1. Mean squared differences of the average patterns.

Database	Attributes Used	Qual.	Quant.	N
I	1 to 25	8	17	25
II	1 to 8	8	0	8
III	9 to 25	0	17	17
IV	11,12,13,14	0	4	4
V	11,12,13,25	0	4	4
VI	3,5,6,7	4	0	4
VII	**11,12,13,14,25**	**0**	**5**	**5**
VIII	11,12,13,20,25	0	5	5
IX	3,5,6,7,11,12,13,25	4	4	8
X	11,12,13,14,18,19,20,21,23,24	0	10	10
XI	11,12,13,14,18,19,20,21,23,24,25	0	11	11

Table 1. Andes GIS learning databases used.

divided in learning and test sets. The sets consist of randomly selected patterns from the whole corpus. Learning sets of 10% (64 patterns) to 95% (577 patterns) of the original database (641 patterns) were generated. The complement was selected as test set. There are N input neurons in the network, depending on the database dimension. The unsupervised part of the network, Fuzzy c-means, must find two classes: *deposit* and *barren*. Minimerror will find the best hyperspherical separator for each class. In the same condition, a multilayer perceptron with 10 neurons on a single hidden layer obtains up to 77% of correct classification.

7 Results

Classification performance corresponded to the percentage of well classified situations. Learning and generalization discrimination of *deposit* and *barren* were obtained for all learning databases. Database **VII** (including only few quantitative attributes) had the best learning and generalization performances in comparison to the other databases. When using all the attributes, the performances fell. Figure 2 shows some results of this behavior. Based on this information, we kept this database to perform 100 random tests. The capacity of discrimination between *deposit* and *barren*, according to the percentage of learned patterns is shown in figure 3. The *deposit* class detection is quite higher than the *barren* class. We note that the detection of gold, argent and copper remain quite precise, bet, that of the molybdenum is rather poor. This can be explained according to the weak presence of this metal.

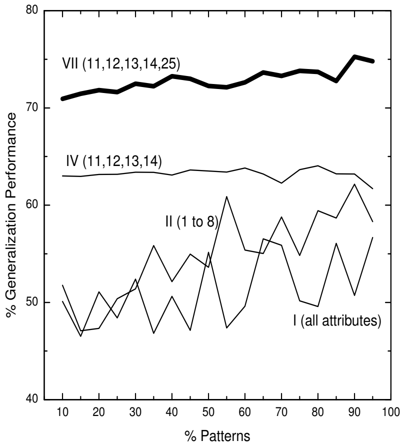

Fig. 2. Generalization performances according to the learning set size obtained by the hybrid model with various databases.

8 Conclusion

We developed a variation of Minimerror for unsupervised classification with hyperspherical separations. The hybrid combination of Minimerror and Fuzzy c-means proved to be the most promising. This strategy applied to real-world database, allowed us to predict in a rather satisfactory way if a site could be identified or not as a deposit. The 75% value obtained for the well classified patterns with this unsupervised/supervised algorithm is comparable to the values obtained with other classical supervised methods. This also shows the discriminating capacity of the descriptive attributes that we selected as the most suitable for this two-class problem. Finally, according to the figure 3, we should be able to obtain a significant improvement of the performance just increasing the number of examples. Additional studies must be made to determine more accurately other relevant attributes, as well as to perform hybrid learning multi-class tasks.

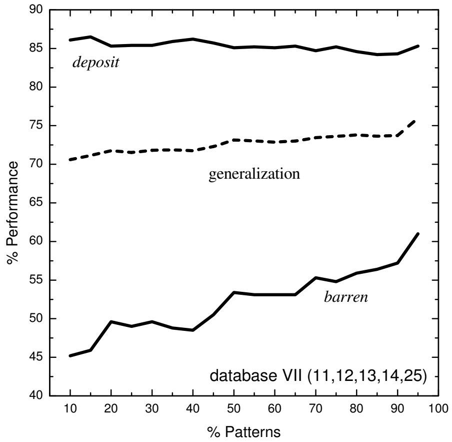

Fig. 3. *deposit*/*barren* discrimination performances in generalization according to the learning set size (100 tests) obtained by the hybrid model with the database VII.

Acknowledgement

This research was supported by the *Bureau des Recherches Géologique et Minière (BRGM)*, France.

References

1. Gordon M., Grempel, D.: Learning with a temperature dependant algorithm. Europhysics Letters **29** (1990) 275–262
2. Torres-Moreno, J.M.: Apprentissage et généralisation par des réseaux de neurones: étude de nouveaux algorithmes constructifs. Thèse INPG, France, (1997)
3. Torres Moreno, J.M., Gordon, M.B.: Efficient Adaptative Learning for Classification tasks with Binary Units. Neural Computation **10(4)** (1998) 1007–1030
4. Torres Moreno, J.M., Gordon, M.B.: Characterization of the Sonar Signals Benchmark. Neural Processing Letters **7(1)** (1998) 1–4
5. Dreyfus, G., et al.: Réseaux de Neurones. Méthodologie et applications. Eyrolles, Paris (2002)
6. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York (1981)
7. deGruijter, J.J., McBratney, A.B.: A modified fuzzy k-means for predictive classification. In: Bock,H.H.(ed) Classification and Related Methods of Data Analysis. Elsevier Science, Amsterdam. (1988) 97–104
8. http://www.brgm.fr
9. Michel, H., Permingeat, F., Routhier, P., Pélissonnier, H.: Propositions concernant la définition des unités métallifères. Comm. Scientifique à la Commission de la Carte géologique du monde. 22th Int. Geol. Congre. New Dehli (1964) 149–153
10. Torres-Moreno, J. M., Aguilar, J. C., Gordon, M. B.: The Minimum Number of Errors in the N-Parity and its Solution with an Incremental Neural Network. Neural Processing Letters **16(3)** (2002) 201–210