Astronomy from Large Databases II, Eds. A. Heck and F. Murtagh, pp. 325–330,1992,
ISBN 3-923524-47-1

CLASSIFYING STARS: A COMPARISON BETWEEN CLASSICAL, GENETIC AND NEURAL NETWORK ALGORITHMS

M.HERNANDEZ-PAJARES[1], F.COMELLAS[1],
E.MONTE[2] and J.FLORIS[1]
[1] *Departament de Matemàtica Aplicada i Telemàtica*
[2] *Departament de Teoria del Senyal i Comunicacions*
ETSETB, Universitat Politècnica de Catalunya,
Apartat 30002 - Barcelona, Spain

Abstract

One of the relevant studies that are carried out with stellar samples is the segregation of stars in populations with the aid of spectral, photometric and/or kinematic data. We present the first results of the use of four different classification techniques on stellar catalogues: the Self-Organizing Map and Multi-Layer Perceptron, two different neural network architectures, and the Genetic algorithms and Hierarchical Clustering. Also the Principal Component Analysis is applied to the initial data that consist of two synthetic samples of the Solar Neighborhood with 3-D position and velocity, metallicity and age.

1 Introduction

The necessity of significant improvements in the classification strategies is a fact in the 90's Astronomy. The main reason is obvious: a large and new amount of good data are being available nowadays. The main sources of these databases are the orbiting observatories HIPPARCOS and HUBBLE Space Telescope. Following this idea, we present a comparison between classical (Hierarchical Clustering), neural (Self-Organizing Map and Multi-Layer Perceptron) and genetic classification algorithms using as input data two synthetic samples of the solar neighborhood stellar content. Each of them is composed by a mixture of the three populations, thin disk, thick disk and halo, in different proportions.

2 The synthetic sample

We want to generate a realistic three component local sample, with thin disk (t), thick disk (T) and halo (h) stars. We suppose two samples with 1000 stars and with different mixing proportions as [16] do. The sample A presents true mixing proportions: $t : T : h = 200 : 20 : 1$ with 905 stars belonging to thin disk, 90 to thick disk and 5 to halo. For the sample B we suppose that has been selected in a way that strongly favors nearby halo and thick-disk stars; for instance some samples with high proper motions present a proportion $t : T : h = 2 : 2 : 1$

([16]). So this sample has 1000 stars with 400 belonging to each disk component, and 200 to the halo. The real heliocentric distances have been taken lesser than 2.5 kpc. The attributes that we simulate for each star are: (a) **Position**, according with the accepted density function of stars for each population ([1, 4, 10]); also a gaussian error proportional to the heliocentric distance is added to r, with $\sigma_r = 0.25r$ following the discussion of [3]. (b) **3-D velocity** which consists of three contributions: (1) The mean velocity field with a lineal Oort-Lindblad model with typical values for the Oort's constants A, B and asymmetric drifts for each population ([12, 2]); (2) The cosmic dispersions, with values of [2]; (3) Gaussian observational errors for radial velocities and proper motions, with values of [6, 5]; Also the errors in heliocentric distance affect in an important manner to the final values of the velocity. (c) **Metallicities and ages**; following [16], we adopt for the thick disk, thin disk and halo, their gaussian distributions for both attributes.

3 Classification algorithms and calculations

3.1 The Principal Component Analysis

This known multivariate technique (hereinafter PCA) is not specifically a classification tool (see for instance [15]). It looks for the maximum variance directions and gives us a quantitative idea of which attributes are more discriminant for the sample. These directions are the eigenvectors of the correlation matrix and the respective eigenvalues give us the amount of variance explained for each eigenvector.

The Principal Component Analysis of the two samples, A and B, gives two sets of 8 eigenvalues, the worth of them with respective associated variances of 6.2% and 3.4%. However the first eigenvalue explains the 23% of the total variance for the sample A and the 26% for the B. Also from the normalized eigenvectors for the first 2 eigenvalues ($\approx 40\%$ of the total variance) is possible to appreciate the more discriminant attributes. The 5th, 7th and 8th components appears as the more important ones for the main eigenvector, that correspond to circular rotation component V, which contains the assymmetrical drift, metallicity $[Fe/H]$ and age respectively. The second eigenvector consists of the galactic longitude and the velocity component towards the galactic center.

3.2 The Hierarchical Clustering

The Ward's minimum variance method is a hierarchical clustering approach to the unsupervised classification. It produces clusters which satisfy compactness and isolation criteria. The objective is to agglomerate two clusters C_1 and C_2 into cluster C such that the within-class variance of the partition thereby obtained is minimum. Alternatively, the between-class variance of the partition obtained is to be maximized. The method allows also a graph representation of the different levels of hierarchy: a dendrogram (see [15]).

The results obtained for the samples A and B were initially of poor quality. The algorithm did not distinguish properly between the three populations. Nevertheless the results improved dramatically using instead of the original 8 attributes the 7 new attributes given by the 7

better eigenvectors in PCA. They explain the 93.8 and 96.6 % of the total variance for A and B. In this manner we have analyzed the data in the 3^{rd} top levels of hierarchy, equivalent to consider 3 centroids, and within a total of 9 levels. The confusion matrices obtained (Table 1) have associated recognition errors of 1.2% for A and 15% for B. Another runs with other three clustering algorithms, the McQuitty's, median Gower's and complete link methods (see [15]), have given worst results.

3.3 The Self-Organizing Map

The Self-Organizing Map (SOM) is an unsupervised neural classifier that has been applied to real astronomical data in [9, 8]. The basic aim of this classifier is finding a smaller set $C = \{c_1, \ldots, c_p\}$ of p centroids that provides a good approximation of the original set S of n stars with m attributes, encoded as "vectors" $x \in S$. Intuitively, this should means that for each $x \in S$ the distance $\| x - c_{f(x)} \|$ between x and the closest centroid $c_{f(x)}$ shall be small. However, the main advantage of the algorithm is that it also arranges the centroids so that the associated mapping $f(.)$ from A to S maps the topology of the set S in a least distorting way. Usually A is a bidimensional set of indexes named *Kohonen map* where proximity between them means similarity between the global properties of the associated groups of stars (see for more details [8]).

The Self Organizing Maps of the samples A and B, after 25000 iterations (25 presentations of the whole sample) and with sizes of 5x5=25 and 10x10=100 centroids respectively, are calculated. The number of nodes is choose in order to allow the representation of the minor population. The maps with the predominant population for every centroid present a continous distribution between thin disk, thick disk and, topologically disconnected respect to the first population, the halo. Practically all of the centroids contain a predominant population with more than 50% of the respective stars. The purity of the classification can be also appreciated looking at the confusion matrix given in Table 1. It is evident the good discrimination obtained by this neural method. From this table the recognition errors are 22 stars = 2.2% for the sample A and 45 stars = 4.5% for the sample B. It is important to note that no confusion appears between the thin disk and the halo components.

3.4 The Multi-Layer Perceptron

The Multi-Layer Perceptron (MP) is a kind of neural net that is capable of generating arbitrary decision boundaries ([14]). In this artificial neural network the input pattern is the vector x which has a dimension equal to the number of pertinent parameters. It can also be seen that the network approximates the decision boundary by means of hyperplanes. In order to obtain the coefficients (weights or synapsis) of these hyperplanes, a gradient search is done on the surface generated by the error function on the coefficients space. The cost function which is minimized is quadratic with respect to the difference between the output and the reference for one given unit and one input pattern, and takes into account all the units and patterns. The updating of the coefficients (weights) of each node of the network is done by means of a gradient search.

	SOM			Mult. Perceptron			Genetic Alg.			Hierarch. Clust.		
A	900	13	0	905	7	0	905	74	1	903	6	0
	5	74	1	0	81	0	0	16	3	2	84	4
	0	3	4	0	2	5	0	0	1	0	0	1
B	395	4	0	397	1	0	397	70	9	390	1	0
	5	388	28	3	392	3	3	299	85	10	375	115
	0	8	172	0	7	197	0	31	106	0	24	85

Table 1: The confusion matrices between the thin disk (1), thick disk (2) and halo (3) and obtained with the Self-Organizing Map, Multi-Layer Perceptron, Genetic Algorithm and Hierarchical Clustering, are given for the samples A and B . For one given confusion matrix the number in the i-th row and j-th column indicates the stars that really belong to the population j and appears in a centroid with predominant component i.

The experiment consists of the training of the network by means of a training database, a cross validation database, and a test database, this done for each sample A and B. The cross validation database was used in order to determine the number of iterations needed for training the network, 2000 epochs in our case.

We did several previous experiments to determine empirically a good architecture for the classifier. We found that the best architecture consisted of a network with one hidden layer. The architecture without hidden layers, which approximates the decision boundary of each class by means of a single hyperplane, yielded classifications rates of the order of 50%, which means that the classes are not linearly separable. After few trials we found that three units in the hidden layer were sufficient for our problem.

The results show that each class is projected into different regions of the space generated by the activation of the hidden units. So they are easily separable by means of the output units. The sum of the confusion matrices for the training subset (500 stars) and test subset (500 stars), are indicated in Table 1. The recognition error for the A training subset and A test subset are 0% and 1.8% respectively (total error of 0.9%). For the sample B these errors are 0.4% and 2.4% with a total error of 1.4%.

3.5 The Genetic Algorithms

Genetic algorithms (GA) were first introduced by J.H. Holland in the 60's [11], and have been successfully applied to combinatorial optimization problems, pattern recognition, classifier systems, scheduling, symbolic system evolution, and some other questions (see [7] for an extensive description and bibliography).

In a genetic algorithm the starting point is always a collection, known as *population*, of possible solutions generated at random. A suitable encoding of each solution in the population is used in order to compute its *fitness*. At each iteration a new population, or *generation*, is obtained by *mating* the best of the old solutions with one another. To create the next generation the whole population is considered and new solutions are formed through *reproduction*, *crossover* and *mutation*. The solutions that will be considered for crossover are probabilistically selected according to their fitness values from the set that constitutes the current generation. This

new population become the *parent pool*. Usually, a constant number of solutions are selected so that the maintained population is of fixed size. Crossover creates two new *child* solutions from two solutions sampled from the parent pool. In this way, fitter parents have a better chance of producing children. Children solutions are obtained by interchanging random parts of their parents. Some randomness is also introduced through a mechanism called *mutation* to ensure that the algorithms avoid getting stuck at local minima. Mutation changes selected parts of a solution. The crossover and mutation operations are done with probabilities p_{cros} and p_{mut}. This ensures that some solutions from the current generation will be kept in the new generation. Once a new generation is created, the fitness of all solutions is evaluated and the process is repeated. At each generation the best solution is recorded. The algorithm ends when the results stabilize or the optimal solution, when it can be identified, is reached.

As for the SOM technique, our implementation of the algorithm tries to find a set of p centroids, such that each element of the original set S of n stars with m attributes has the smallest possible distance to its closest centroid with a global quadratic cost in this case. Therefore, this is an optimization problem and we may use a standard optimization genetic algorithm.

For our preliminary test a population of 100 individuals was considered. Other parameters were set as follows: $p_{cross} = 0.9$, $p_{mut} = 0.3$ and a number of 9 centroids. The resulting confusion matrices for the A and B samples are shown in Table 1. The respective recognition errors are 7.8 and 19.8%. Higher rates of recognition should be obtained by tuning adequately the algorithm parameters. Work is still in progress in this sense.

4 Conclusions

It is difficult to compare the efficiency of so different algorithms in order to distinguish between several stellar populations in a local sample. A numerical tentative is done in Table 1 where the respective confusion matrices are shown. From a qualitative point of view we can say that the Multi-Layer Perceptron with one hidden layer gives a very good result, for the sample A with the minor halo population and for the (2:2:1) sample B. But the results, using this supervised classificator, are of very poor quality when any hidden layer is employed. This fact, that can be interpreted as that the classes are not linearly separable, is compatible with the worst results (mainly in sample B) obtained by other unsupervised techniques such the Hierarchical Clustering. By other hand, the Self-Organizing Map allows simultaneously the global bidimensional representation of the data and the classification. Nevertheless the decision surfaces can be improved, tending to the Bayessian ones, using refined versions of this algorithm: LVQ1, LVQ2 and LVQ3 ([13]). This will be matter of future work. Also the preliminary results show that the Genetic Algorithm presented in this work can perform well for the classification problem proposed. With a similar computational effort than in other approaches we expect to obtain a high recognition rate. Also the algorithm is robust and simple to implement.

5 Acknowledgments

This work was partially supported by the D.G.I.C.I.T. of Spain under grant No. PB90-0478 and by PLANICYDT/TIC grant No. 92-0800-C05-05.

References

[1] J.N. Bahcall, S.Casertano and K.U. Ratnatunga, Astrophys. J. **320** (1987), 515.

[2] R. Cubarsí, Astron. J. **103** (1992), 1608.

[3] F. Figueras, Ph. D. thesis, University of Barcelona, Barcelona, 1986.

[4] K.C. Freeman, The Stellar Populations of Galaxies (B. Barbuy and A. Renzini, eds.), Kluwer Academic Press, Dordrecht, 1992, p. 65.

[5] W. Fricke, Astron. J. **72** (1967), 1368.

[6] W. Gliese, *Catalogue of nearby stars*, Veroff. der Astronomischen Rechen-Institut, vol. 22, Heidelberg, 1969.

[7] E. Goldberg, *Genetic algorithms in search, optimization and machine learning*, Addison-Wesley, 1989.

[8] M. Hernández-Pajares, R. Cubarsí and E. Monte, In this coonference, 1992.

[9] M. Hernández-Pajares and E. Monte, Artificial Neural Networks, Lecture Notes in Computer Science, vol. 540, 422, Berlin, 1991.

[10] J.E. Hesser, The Stellar Populations of Galaxies (B. Barbuy and A. Renzini, eds.), Kluwer Academic Press, Dordrecht, 1992, p. 1.

[11] J.H. Holland, Scientific American **267** (1992), no. 1, 44.

[12] F.J. Kerr and D. Lynden-Bell, Monthly Notices Roy. Astronom. Soc. **221** (1986), 1023.

[13] T. Kohonen, Proceedings of the IEEE **78** (1990), no.9, 1464.

[14] R. Lippmann, IEEE ASSP Magazine **4** (1987), no. 2, 4.

[15] F. Murtagh and A. Heck, *Multivariate data analysis*, D. Reidel Publishing Company, Dordrecht, 1987.

[16] J.M. Nemec and A.F.L. Nemec, The Stellar Populations of Galaxies (B. Barbuy and A. Renzini, eds.), Kluwer Academic Press, Dordrecht, 1992, p. 103.