Improving Neural-Based Classification of Databases with Overlapped Classes: the Case of Star/Galaxy Segregation

Pilar *Gómez-Gil, IEEE Senior Member,*
Omar López-Cruz, Astrophysics-INAOE
National Institute of Astrophysics, Optics and Electronics
Tonantzintla, México.
pgomez@acm.org, omarlx@inaoep.mx

Ana Bertha Cruz-Martínez
Department of Computer Science
Technological Institute of Queretaro (ITQ)
Querétaro, México
isc.anabcm@gmail.com

***Abstract*—there are many real-life classification problems where class overlapping severely limits the classification accuracy. In these situations is difficult to build automatic classifiers that obtain good generalization performance. An interesting case is found in the separation of stars and galaxies, which arises in galactic or extragalactic studies. There are many astronomical analysis packages which deal with this problem; for example, the very popular package SExtractor (Source Extractor) has incorporated a multi-layer perceptron (MLP) neural network classifier. We believe that SExtractor performance is suitable for improvement. In our way for building a better classifier, we analyzed the behavior of MLP-based classifiers for this kind of data. In this paper we present an experiment where, using WEKA, we have automatically selected the best characteristics to discriminate galaxies from stars and automatically selected the topology of a MLP that best defined the decision region. Our classifier obtained slightly better results than SExtractor when compared to classifications obtained by a human expert, using less computational resources that SExtractor. However, we conclude that more specific information about the problem needs to be used to build a better separator of star/galaxies.**

Keywords: galaxy/star separation, classification using MLP, design of classifiers, feature selection, SExtractor, WEKA.

I. INTRODUCTION

The amount of data in Astronomy is already tremendously large, and increases on a daily pace. Therefore, it is urgent to build automatic systems able to efficiently extract meaningful information. Among other techniques, Computational Intelligence has been used for data mining this kind of databases. Currently, artificial neural networks (ANN) are the most widely used and well-known machine learning algorithms in Astronomy [1]. A reason for such popularity is that ANN are universal non-linear function approximators, which makes them very useful in a variety of situations, including the accurate definition of non-linear decision regions in classifiers [2].

Star/Galaxy segregation is a fundamental task in observational studies in galactic or extragalactic astronomy [3]. The problem has deep historical roots, which lie at the crux of the discovery of the structure of the universe. At the turn of the 20th century astronomers recognized two types of objects: point-like sources that were associated with stars, and nebular objects that were associated with extended sources, such as comets, planetary nebulae, star clusters, and galaxies. However, there are very few studies related to the automatic classification of star/galaxies, as well as comparisons on the accuracy among different implemented schemes. Currently, the most popular sofware used for this task is SExtractor, a powerful tool for data analysis in astronomy [3]. This software is able to find objects in a image, obtaining photometric measures from them and classifying them using a multi-layer perceptorn (MLP) neural net trained with backpropagation algorithm [4], [5]. SExtractor does not return a class, but an stellarity index: 0 for a galaxy and 1 for a star; intermediate values of this index define ambiguous objects. Therefore, strictly speaking, SExtractor does not work as a classifier but as a function approximator. This way to separate stars and galaxies creates a confusion area when the objects tend to group around a stellar degree of 0.5. In the other hand, it is interesting to notice that SExtractor classifier was trained using simulated images, and that just ten parameters out of the twelve measures obtained during the data analysis are used as features for classification. These ten features were manually selected by the creators of SExtractor, based on their expertise as astronomers. The authors of SExtractor claimed that this classifier is "accurate enough" around 90% of the times, but they did not showed how this performance was estimated,

neither they compared these results with a human expert or with other classifiers.

We, therefore, propose the building of an automatic classifier based on computational intelligence techniques, capable to discriminate between stars and galaxies, overcoming the limitations of SExtractor. To reach our goal, our first step was to analyze the classifier of SExtractor. We explored the possibility to improve upon SExtractor by automatically choosing the feature vector used for classification. We also decided to automatically select the number of hidden nodes in a MLP in order to have represented the best decision region suitable for a database. Besides that, we trained our MLP-based classifier with real data manually classified by an expert, instead of using simulated data as SExtractor did. We compared the results of both SExtractor and our MLP classifier with the classification done by an astronomer, which was considered the "ground truth." in this experiment.

This paper is organized as follows. Section II presents some basic concepts required to understand the rest of the paper. Section III describes the philosophy and method employed in our experiment. Section IV shows our results, and in section V, we give some conclusions and outline future work.

II. BASIC CONCEPTS

An artificial neural network (ANN) is a massively parallel processor. An ANN is able to store obtained knowledge from experiments, and to make it available to be used in different applications [2]. There are many types of ANN, mainly classified according to the way in which their components are connected and the way in which they are trained. One of these types is the MLP, which is a feed-forward connected network with one or more hidden layers. For a MPL with m input nodes, one hidden layer containing h nodes and n output nodes, the result of each output node is defined as:

$$y_k = \sum_{j=1}^{h} \alpha_j \tau\left(\sum_{i=1}^{m} w_{ji} x_i + b_j\right) \quad k = 1..n \qquad (1)$$

where:

$w_{ji}, b_j \; j = 1..h, i = 1..m$ are weights in the hidden layer,

$\alpha_j \; j = 1..h$ are weights belonging to output node, while

$\tau(x) = \frac{1}{1+e^{-x}}$ is a sigmoid activation function.

It has been proved that a multi-layer perceptron with one hidden layer is able to accurately approximate any continuous function [6]. This makes MLP suitable for representing non-linear decision regions. The appropriate weight values of the network representing such region are adjusted by a training algorithm, using feature vectors of each class contained in the problem; Backpropagation [4], [5] is currently the most popular training algorithm for MLP. Commonly, when a MLP is used as a classifier, it contains one output node for each class to be represented.

Data mining include a set of techniques used to explore databases in an automatic or semi-automatic fashion, with the aim of finding patterns that describe the behavior of the data, with respect to a specific context [1]. One of the most popular software for data mining is WEKA [7], [8], a collection of machine learning algorithms written in Java. It contains tools for pre-processing data, classification, regression, clustering, association rules and visualization. Features used to characterize patterns in a database affect the performance of a classifier. The feature space must be defined in a way that it separates patterns belonging to different classes. However, in many cases is not easy to figure out if a feature is irrelevant or redundant for the classification process. WEKA offers methods to filter features, in a way that the most relevant are identified. To do so, WEKA combines different search methods and evaluation algorithms to figure out the importance of each attribute.

Astronomers separate galaxies from stars using photometric measures over images stored in a specific format. The most widely used format is the Flexible Image Transport System (FITS), which was primarily designed to store scientific data sets consisting of multidimensional arrays. A FITS file consists of one or more Header + Data Units (HDUs), where the first HDU is called the `Primary HDU', or `Primary Array'. The primary array contains an N-dimensional array of pixels. Five different primary data types are supported: unsigned 8-bit bytes, 16 and 32-bit signed integers, and 32 and 64-bit single or double precision floating point real numbers. FITS can also store 16 and 32-bit unsigned integers. The Picture Processing Program (PPP) [9] uses a classifier based on "growth curve" analysis, which basically consists on the analysis of the behavior of the flux integrated within a set of concentric apertures of increasing size. Fig. 1 shows an example of a FITS image. This image was obtained using a megapixel (2048 X 2048 pixel) charge coupled device (CCD). CCDs are universally applied in Astronomy. They can detect up to 100% of the incident photons (high quantum efficiency) and can respond linearly within wide intensity ranges (large dynamic range). As it was explained before, SExtractor provides a way to automatically separate galaxies form stars. SExtractor supports FITS and MEF (Multi-extension-FIT) images. The network used by SExtractor has one hidden layer with 10 neurons and one neuron in the output layer, which defines the stellarity index. It is reported that the training of this classifier was done using 600 simulated images. Each of these images were input to SExtractor using 8 diferent extraction parameters, producing a catalog with around 10^6 patterns used to train the network [10].

III. RESEARCH METHOD

In order to design a good MLP classifier suited for this problem, and analyze its behavior we follow the steps described in next sections. [9]

Figure 1. A monochromatic image of the Coma cluster in grey-scale, obtained in the R filter using the Kitt Peak 0.9m telescope and the T2KA CCD

A. Building the training, validation and testing files.

ANN are data-driven models, a MLP classifier is as good as the data used to train it. However, in many cases, ANN users do not analyze the quality of data or select characteristics when training an ANN [11]. In the other hand, MLP tend to over fit the data used to train them, which decrease its ability to generalize, that it, classify correctly patterns not included in training. To avoid these problems, our database was divided in three sets, as suggested in [11]:

- 65% of data was used as a training set for estimating de model,
- 20% of data was used as a validation set, to select the best model,
- 15% of data was used for testing the performance of the selected model.

These files were generated using 2,680 objects taken from the Coma cluster of galaxies (Abell 1656) catalog. These objects were classified automatically by a discriminator based on the "compactness" of the object image, using the Picture Processing Program (PPP) [9]. An astronomer validated the classifications by eye using the five classes, described at table I. This classification is considered the "ground truth" in this work. Table I also shows the distribution of data by class. It must be pointed out that this database is unbalanced, that is, the number of patterns among classes highly differs. This database was not pre-processed in any way.

B. Selection of best features

SExtractor generates a catalog where objects are represented using 12 features. This database was input to WEKA to find the best features that represent the classes. The WEKA feature filter called "CfsSubsetEval" and search algorithm called "BestFirst-D1-N5" [12], [13] were used to select low-correlated features with a high capability to separate classes.

C. Finding the right number of hidden neurons.

As in the case of SExtractor, a MLP with one hidden layer was used to build our classifier. This MLP has 6 input nodes (one for each characteristic selected by WEKA) and 5 output nodes, representing the 5 possible classes identified by the expert, showed in Table I. All nodes in the MLP have a sigmoid activation function.

To determine the optimum number of hidden nodes in the MLP, networks with 5 up to 15 hidden nodes were trained using the training dataset, and their performance was measured using the validation dataset. For each value in the hidden node, training and evaluation of the network were executed 10 times, changing each time the initial random weights of the network. After that, the network with the highest mean performance was chosen. The performance was measured as the percentage of correct classifications over the total number of samples in the validation data set.

D. Testing the selected network

After selecting the MLP with best performance in the validation dataset, we tested its performance using the testing dataset, which contains data that was not "seen" by the classifier before. This performance was compared with the performance obtained by SExtractor and both results were compared with the astronomer classification, which was considered "ground truth" for this research. Indeed, we analyzed the classification rate by class obtained by our MLP, in order to find out where the classification errors occurred, and figure out what drawbacks the MLP could show with this specific problem.

TABLE I. DISTRIBUTION OF DATA BY CLASS.

Class	Description	Number of objects in the database
0	false object	63
1	Galaxy	1,581
2	Galaxy	516
3	Star	490
4	Saturated star	30

IV. RESULTS

This experiment was carried out using Matlab 7.0 and its neural network toolbox V5.0.2. Each MLP was trained until a minimum MSE of 1E-2 was reached or when 10,000 epochs were executed. The learning coefficient was set automatically by Matlab. The training algorithm was Levenberg-Marquardt. As explained before, selection of best features was carried out using WEKA tools "CfsSubsetEval" and "BestFirst-D1-N5", which automatically selected 6 features from the catalog. Fig. 3 shows plots of two characteristics selected by WEKA: "Fixed aperture magnitude vector" and "Fraction of light radius." For a detailed description of all features selected by WEKA see [10].

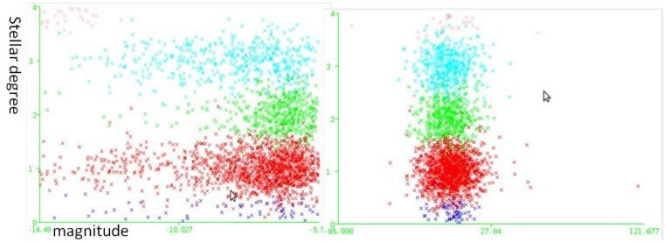
The best structure of the MLP was obtained as explained in section III.C. Table II shows the mean performance obtained by networks with 5 to 15 hidden nodes. There it can be noticed that the best performance using the validation test was 80.1%, obtained with a MLP with 13 hidden nodes. After that, we tested the performance of this MLP with 13 hidden nodes using the testing dataset, obtaining 79.1%. Table III summarizes the performances obtained by the three actors involved in the experiment: astronomer, SExtractor and our

best MLP. Notice that the performance of our classifier is slightly better than the performance obtained by the SExtractor classifier, but it uses a less number of characteristics, which represents less computational operations executed during training and classification. Table IV shows the classification rate by class, obtained by our best MLP classifier. Notice that, given the fact that the database is unbalanced, the performance of classes with few patterns is much worse than whose presented in class 1, the one with the highest number of patterns (as shown in table I).

V. CONCLUSIONS

In this paper we present the results obtained from building a classifier based on a MLP for separation of stars from galaxies. This experiment was executed as the first step to find out how to improve the classification performance provided by the most popular software available nowadays for this problem. We automatically identified the 6 best features to be used to train the MLP using tools provided by WEKA. We also experimentally found that 13 is the best number of hidden nodes for the MLP, for the database used to carry out our experiment. Using this MLP with 13 hidden nodes and 6 input nodes we got a performance of 79.1% compared to 77.1% obtained by SExtractor, the most popular software. This performance was calculated considering as "ground truth" the manual classification made by an astronomer. The training and testing of our classifier is slightly less computationally intensive that the used by SExtractor, because it has less weights to adjust and less operations to execute.

Even though our results were slightly better than the ones obtained by SExtractor, such difference is not very representative. Therefore, we conclude that a better classifier of stars and galaxies must include more information related to the problem, and other classifiers need to be tried. Also as future work, we will try the use of automatically-built ensemble fuzzy classifiers, which have proved to work well in other unbalanced data with overlapped classes [14].



(a) Characteristic F: MAG_APER (Fixed aperture magnitude vector)

(b) Characteristic H: FLUX_RADIUS (Fraction-of-light radii)

Figure 2. Examples of two characteristics selected by WEKA

TABLE II. MEAN PERFORMANCE FOR DIFFERENT TOPOLOGIES OF THE MLPS

Number of hidden nodes	Mean of performance over 10 experiments using the validation dataset
5	71.8%
6	71.8%
7	73.2%
8	79.7%
9	77.7%
10	78.8%
11	79.9%
12	79.8%
13	**80.1%**
14	79.7%
15	79.4%

TABLE III. PERFORMANCE IN CLASSIFICATION

Actor	Classification performance in testing dataset
Astronomer	100% (ground truth)
SExtractor classifier, 10 features	77.1%
Our best MLP, using 6 features	79.1%

TABLE IV. CLASSIFICATION RATE BY CLASS

Class	Performance using a MLP with 13 hidden nodes
0	10.4%
1	95.7%
2	31.%
3	55.9%
4	60.0%

ACKNOWLEDGMENT

A. B. Cruz Martinez thanks The Mexican Academy of Sciences for supporting a scientific summer-internship at INAOE during 2010 (*Verano de la Investigación de la AMC*). She also acknowledges support from INAOE for the elaboration of her thesis in Engineering, presented at ITQ.

REFERENCES

[1] Nicholas M Ball and Robert J Bruner, "Data mining and machine learning in astronomy," *International Journal of Modern Physics D* , vol. 19, no. 17, pp. 1049-1106, July 2010.

[2] S Haykin, *Neural Networks*, 2nd ed. Upper Saddle River: Prentice Hall, 1999.

[3] E Bertin and S Arnouts, "SExtractor: software for source extraction," *Astronomy and Astrophysics Supplement Series*, vol. 117, pp. 393--404, 1996.

[4] D E Rumelhart, G E Hinton, and R J Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the MicroStructure of Cognition*. Cambridge MA, USA: MIT Press, 1986, vol. 1, ch. 8.

[5] P Werbos, *The Roots of Backpropagation from ordered derivatives to neural networks and political forecasting*. New York: Wiley-Interscience Publication, 1994.

[6] G Cibenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signal and Systems*, vol. 2, no. 4, pp. 303-314,

1989.

[7] M Hall et al., "The WEKA data mining software: an update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10-18, 2009.

[8] The University of Waikato- Machine Learning Group. Weka 3: Data Mining Software in Java. [Online]. http://www.cs.waikato.ac.nz/ml/index.html

[9] H K C Yee, "A faint-galaxy photometry and image-analysis system," *Astronomical Society of the Pacific, Publications*, vol. 103, pp. 396-411, April 1991.

[10] Ana Bertha Cruz-Martínez, *Clasificación de estrellas y Galaxias a través de métodos directos vs. redes neuronales artificiales*. Querétaro, Querétaro, Mexico: Tesis para obtener el título de Ingeniera en Sistemas Computacionales, 2011.

[11] G Peter Zhang, "Avoiding Pitfalls in Neural Network Research," *IEEE Transactions on Systems, Man and Cybernetics—Part C: Applications and reviews* , vol. 37, no. 1, pp. 3-16, January 2007.

[12] University of Waikato. The Weka API. [Online]. http://weka.sourceforge.net/doc.stable/

[13] M A Hall, Correlation-based Feature Subset Selection for Machine Learning, 1998, PhD Thesis. University of Waikato.

[14] A Rosales-Perez, C A Reyes-Garcia, P Gomez-Gil, J A Gonzalez, and L Altamirano, "Genetic selection of fuzzy model for acute leukemia classification," in *MICAI 2011 Part I, LNAI 7094*, I Batyrshin and G Sidorov, Eds.: Springer, 2011, pp. 537-548, DOI: 10.1007/978-3-642-25324-9_46.