STUDIA UNIV. "BABEŞ-BOLYAI", INFORMATICA, Volume I, Number 1, March 1996

SAADI: SOFTWARE FOR FUZZY CLUSTERING AND RELATED FIELDS

H. F. POP

Abstract. The article describes a software product whose main task is mainly to perform automatical classification of different data sets. The system, programmed by the author, implements both traditional and original algorithms [3]. For each of its components the implemented algorithms and details regarding its functioning are specified. Being programmed by using modern techniques (Object Oriented Programming, Windows Programming), with the help of *Borland Delphi 1.0* programming environment for Microsoft Windows operating system, the product developed here proves itself to be extremely flexible, performant and with a user-friendly interface.

1. Introduction

The aim of this paper is to describe the SAADI software (System for Automatical Analysis of Data and for their Interpretation), produced with the aim of helping the research developed in Fuzzy Clustering and in different conex fields.

This software system has been realized with the help of the Windows programming facilities provided by *Borland Delphi 1.0*. All the interdependency mechanisms of the system have been programmed using object oriented programming techniques.

The SAADI system performs the following tasks:

- essential characteristics selection of a data set;
- bidimensional space projection of the data, for a better vizualization of them;
- unsupervised hierarchical, non-hierarchical and simultaneous clustering of a data set; for the horizontal unsupervised classifier we are able to determine the fuzzy set associated to a classical data set (as particular case, fuzzy regression);
- computation of the fuzzy set corresponding to a classical set and to a certain index; the version of this algorithm that uses linear prototypes has been called the Fuzzy Regression Algorithm;

Received by the editors: September 15, 1996.
1991 *Mathematics Subject Classification.* 68T35, 62H30.
1991 *CR Categories and Descriptors.* I.5.3 [Pattern Recognition] Clustering – algorithms, similarity measures.

- unsupervised hierarchical, non-hierarchical or simultaneous clustering of the characteristics set;
- supervised clustering (based on training), either with classical decision (and a separation hyperplane is produced), or with fuzzy decision (and the membership degrees of the extra point are directly produced);
- some other interesting facilities such as:
  - producing normalized data;
  - constructing significant variables for the fuzzy regression line;
  - test of the algorithm for computing the eigenvalues and eigenvectors of a square, symmetrical and positively defined matrix;
  - testing and drawing a separation hyperplane;
  - a text editor and a text viewer for data files and for results files;

The data set and, if this is the case, the initial fuzzy partition are read by the system through some ASCII files. The other variables and different clustering options are introduced by the menu system and are saved in a configuration file, so that when reruning the program they are automatically read from this file. The results are presented as a ASCII file with all the necessary information so that the user may have a clear idea concerning the operation that just took place.

2. Characteristics selector

This part of the system performs a transformation of the data from the original space in a space having fewer characteristics. Thus, it is possible to realize both a projection of the data into a reduced dimension space, i.e. combining the initial characteristics into a smaller number of new characteristics, and a selection of the most relevant characteristics out of the original ones.

Two methods are available, as follows:

- the Karhunen-Loewe method, based on the principal component analysis. There we may either project the data on the eigenvectors of the covariance matrix, eigenvectors corresponding to the greatest eigenvalues, or a characteristics selection, by considering those characteristics the nearest of the eigenvectors;
- a method proposed by Dumitrescu, based on the computation of a certain importance factor, associated to each original characteristic.

3. Bidimensional space projector

This component realises the data projection in a bidimensional space, in order to allow a good graphical visualization of data. Each of the methods used allows the displaying of the projected data both on a text display and on a graphical display (in this case the resolution is much better).

Two methods are available, as follows:

- the Karhunen-Loewe method;
- the Sammon method, based on the minimization of some criterion function built such that it is targeted the conservation as much as possible the distances between points before and after the projection.

4. Unsupervised hierarchic classifier

This component performs divisive hierarchical classification based on fuzzy sets. There are operational hierarchical classifiers based on:

- point prototypes;
- use of adaptive metric;
- two types of ellipsoidal prototypes;
- linear prototypes;
- convex combination between the point and line prototypes;
- adaptive prototypes;
- classical partition prototypes.

This type of classifiers allow the user to set some working options as:

- the type of initial partition wanted for the classification process (random, or predefined in a certain way);
- whether data normalization is intended;
- whether the graphic variant of the classifier is to be used;
- the polarization threshold beginning with which a fuzzy set is no more split;
- the error threshold beginning with which two fuzzy partitions are considered identical.

5. Unsupervised horizontal classifier

This component performs horizontal classification based on fuzzy sets. There are operational horizontal classifiers based on:

- point prototypes;
- use of adaptive metric;
- two types of ellipsoidal prototypes;
- linear prototypes;
- convex combination between the point and line prototypes;
- adaptive prototypes;
- classical partition prototypes.

This type of classifiers allows the user to set some working options as:

- the number of classes the initial data set is to be split in; if this number is equal to 1, the classifier will determine the fuzzy set associated to the given data set and to a membership threshold to be set by the user (see Section 7);
- the type of initial partition wanted for the classification process (read from outside; random, or predefined in a certain way; if the initial partition is read from outside the system will try to improve its quality by refining it);
- whether data normalization is intended;
- whether the graphic variant of the classifier is to be used;
- the error threshold beginning with which two fuzzy partitions are considered identical.

6. Unsupervised cross-classifier

This component performs hierarchical cross-classification based on fuzzy sets. Now there are operational a few variations of the hierarchical cross-classifier based on point prototypes (see [8]).

This classifier allows the user to set some working options as:

- the type of initial partition wanted for the classification process (random, or predefined in a certain way);
- whether data normalization is intended;
- whether the graphic variant of the classifier is to be used;
- the polarization threshold beginning with which a fuzzy set is no more split;

- the error threshold beginning with which two fuzzy partitions are considered identical.

7. The fuzzification component

This component aims to produce the fuzzy set corresponding to a classical set and to a certain fuzzification index [5, 7]. Currently this components works in two different ways:

- it produces a single fuzzy set corresponding to the given classical set and to the fuzzification index set by the user;
- it produces a whole family of fuzzy sets corresponding to the given classical set and to the fuzzification index taking the values 0.01, 0.02, ..., 0.98 and 0.99.

There are operational fuzzyfication components based on:

- point prototypes;
- use of adaptive metric;
- two types of ellipsoidal prototypes;
- linear prototypes;
- convex combination between the point and line prototypes;
- adaptive prototypes;
- classical partition prototypes.

This type of components allows the user to set some working options as:

- the value of the fuzzification index, in the interval $(0, 1)$;
- the type of initial partition wanted for the classification process (read from outside; random, or predefined in a certain way; if the initial partition is read from outside the system will try to improve its quality by refining it);
- whether data normalization is intended.

8. Regression algorithms

This component of the system is meant to implement different regression techniques, together with different methods for the evaluation of their quality.

There are currently available:

- the Fuzzy Regression Algorithm [7];

- a version of the algorithm above which produces a whole family of regression lines, corresponding to regression indices taking the values 0.01, 0.02, ..., 0.98 and 0.99;
- component for evaluating the quality of different regression lines.

This part of the system will soon be updated, as we intend to implement different other regression techniques.

9. Unsupervised characteristics classifier

The system allows not only the classification of the data set, but also of the characteristics set. By selecting this option, the classifiers implemented here will classify the characteristics set. In this way we are able to develop useful conclusions with respect to the interdependency of different characteristics and, eventually, we may reduce the data dimensionality.

10. Classical decision supervised classifier

This component of the system implements many training algorithms based both on classical sets and on fuzzy sets.

There are operational the following supervised classifiers:

- the Rerceptron algorithm;
- the Gallant Pocket algorithm;
- the Keller-Hunt algorithm;
- a variation of the Keller-Hunt algorithm in which the memberships are read from an input file and not postulated by the algorithm;
- the Relaxation algorithm;
- the Widrow-Hoff algorithm;
- the Ho-Kashyap algorithm;
- a variation of the Ho-Kashyap algorithm in which the computation of the inverse of a certain matrix is replaced by the use of a symmetrical positively defined matrix.

This type of classifiers allow the user to set different working options, such as:

- wether or not the classical or fuzzy set version of the classifier is requested;
- wether or not the graphical version of the classifier is requested;

- the width of the threshold interval of the membership degrees near 0.5; the learning vectors erroneously classified but with the membership degrees inside this interval will not be taken into account when making a correction of the separation vector;
- the width of the threshold interval of the product $v^T z$; the learning vectors z erroneously classified will not be taken into account when making a correction of the separation vector v if the product $v^T z$ is inside this interval;
- the initialization modality of the separation vector; the vector normal on the mediator hyperplane of the segment made by the prototypes of the two classes, or a vector having all the components equal to one another and equal to a certain given number.

11. Fuzzy decision supervised classifier

This component of the system implements fuzzy decision supervised classifiers. As compared with the classical decision supervised classifiers, that produce a separation hyperplane, this class of algorithms produce the membership degrees of the tested vector.

There are operational supervised classifiers based on [4]:

- fuzzy version of the algorithm of the nearest k neighbours;
- fuzzy version of the algorithm of the nearest prototype;
- the Restricted Fuzzy n-Means algorithm.

In this moment, all these algorithms suppose the existence of point prototypes.

12. Other components of the system

The system allows the user to test some of the algorithms used at the implementation of different classification techniques. Moreover, there are shown some extensions considered to be necessary for the good working of the system:

- component to set the directory with the data file;
- component to set the names of the files with different necessary data;
- component to set the names of the final report files, with the results produced by the system;

- components to set different working options for the supervised and unsupervised classifiers; the settings done here are immediately saved in a configuration file and will be loaded the next time the system is run;
- component to test the algorithm that computes the eigenvalues and the eigenvectors of a symmetrical and positively defined square matrix;
- component to testing and drawing a separation vector;
- component to normalize the data;
- component to view the ASCII files with data or results produced by this system; here is allowed the viewing of lines both unwrapped and wrapped at every 80 character, so that the file should be completely displayed on the screen;
- component to edit the ASCII file with data or results produced by this system.

13. Objects hierarchy of the system

In order to notice better the interdependency and the relationships between different oblects we show in Table 1 the objects hierarchy as well as a short description of their functionality. Of course, here are presented only the objects effectively related to the SAADI system. A series of objects, created in order to extend the objects hierarchy of the Borland Delphi system, are not displayed here.

Object	Description
+--TVector	Allocation of a vector in Heap
| \--TMatrix	Allocation of a matrix in Heap
+--TGrafic	Projection of 2-D data on text screen
| |--TKarhunen	The same, s-D data, Karhunen-Loewe
| |--TSammon	The same, s-D data, Sammon
| \--TRegr	Computes essential regression parameters
+--TGGrafic	Projection of 2-D data on graphic screen
| \--TGKarhunen	The same, s-D data, Karhunen-Loewe
+--TReduc	Generic object for dimensionality reduction
| |--TRedKar	Dimension reduction, Karhunen-Loewe
| |--TRedOrd	Dimension reduction, importance coef.
| \--TNormal	Building of normalized data
+--TFuzzyPct	Unsupervised generalized classifier, points

|--TFuzzyLin	The same, convex combination prototypes
|--TFuzzyAda	The same, adaptive prototypes
|--TFuzzyElp	The same, ellipsoidal prototypes
\--TFuzzyMul	The same, classical partition prototypes
+--TClasPct	Unsupervised hierarchic classifier, points
|--TClasLin	The same, convex combination prototypes
|--TClasAda	The same, adaptive prototypes
|--TClasElp	The same, ellipsoidal prototypes
|--TClasMul	The same, classical partition prototypes
|--TIsoPct	Unsupervised horizontal classifier, points
| |--TIsoLin	The same, convex combination prototypes
| |--TIsoAda	The same, adaptive prototypes
| |--TIsoElp	The same, ellipsoidal prototypes
| |--TIsoMul	The same, classical partition prototypes
| \\--TRegPct	Fuzzification component, point prototypes
| | |--TRegLin	The same, convex combination prototypes
| | |--TRegAda	The same, adaptive prototypes
| | |--TRegElp	The same, ellipsoidal prototypes
| | \\--TRegMul	The same, classical partition prototypes
| \\--TSimPctIA	Cross-classifier, initial algorithm, variant A
| |--TSimPctAA	The same, associative algorithm, variant A
| |--TSimPctIB	The same, initial algorithm, variant B
| | \\--TSimPctAB	The same, associative algorithm, variant B
| \\--TSimPctIC	The same, initial algorithm, variant C
| \\--TSimPctAC	The same, associative algorithm, variant C
+--TTraining	Supervised classifier, Perceptron method
|--TKellerHunt	The same, Keller-Hunt method
|--TRelaxFuzzy	The same, Relaxation method
|--TRelaxFuzzyVar	The same, variation of Relaxation
|--TTrGallant	The same, Gallant method
|--TWidrowHoff	The same, Widrow-Hoff method
|--THoKashyap	The same, Ho-Kashyapp method

| |--THoKashyapVar	The same, variation of Ho-Kashyapp
| \--TTestTraining	The same, obiect for testing new methods
\--TFuzTrain	Fuzzy decision supervised classifier, restricted
|--TFuzKNN	The same, nearest k neighbours
\--TFuzNProt	The same, nearest prototype

Table 1: The objects hierarchy of the SAADI system

14. Conclusions

This system presents under a unitary conception different aspects of the classification theory: the projection in bidimensional space, in order to facilitate the visual inspection of data; the selection of relevant characteristics; the projection of data into a space having a reduced dimension; the fuzzy unsupervised clustering, both hierarchical and horisontal; the fuzzy regression algorithm; the supervised clustering using both classical and fuzzy sets; fuzzy decision supervised clustering; graphical versions of these classifiers.

Among the most important facilities of the system we mention:

- being programmed in *Borland Delphi 1.0* and using the Object Oriented Programming and Windows Programming, the system allows to be extended with minimal programming effort;
- moreover, wherever possible, the system does not contain redundant code, meaning that we based ourselves on the facilities of the Objects Oriented Programming;
- because the whole system is based on the creation of two objects for implementing the notions of vector amn matrix using dynamical memory allocation, the system does not have statical limits in what it concerns the dimensions of input data; these depend only on the availability of the Heap;
- the simple structure of the data files and results files; thus it is possible to pipe different components of the system, i.e. to have the output of one component as the input for another;

- the possibility to obtain a file with the history of all the operations performed by each component of the system and not only the final result;
- the independence of the system components from one another, as well as their independence with respect to the moment of setting their working characteristics; thus, it is possible to run different classifications using the same working characteristics or with minimal value changes; it is also possible to reiterate, omit or execute them in any order, with the single condition for this order to be logical;
- the availability of the graphical versions of all the implemented algorithms; thus, it is possible to study the working evolution of the algorithms, and this is very interesting both scientifically and didactically speaking.

This system has been successfully used in the research activity, as it follows:

- for optimally selecting the solvents systems [10];
- for studying the Roman pottery (terra sigillata) [6];
- for classifying different Greek muds [1, 12];
- for studying the importance of fuzzy regression in chemistry [7];
- for studying the Mendeleev's periodic system elements and for generating a fuzzy system of elements [9, 11, 2].

Another series of applications of the fuzzy clustering theory is in study. For these applications, we are also using the capacities offered by the system.

References

[1] Dumitrescu, D., Pop, H. F., and Sârbu, C. Fuzzy hierarchical cross-classification of Greek muds. *Journal of Chemical Information and Computer Sciences 35* (1995), 851–857.

[2] Horowitz, O., Pop, H. F., and Sârbu, C. Pattern recognition of chemical elements. *Journal of Chemical Information and Computer Sciences* (1996). To appear.

[3] Pop, H. F. *Intelligent Systems in Classification Problems*. PhD thesis, "Babes-Bolyai" University, Faculty of Mathematics and Computer Science, Cluj-Napoca, 1995.

[4] Pop, H. F. Supervised fuzzy classifiers. *Studia Universitatis Babeş-Bolyai, Series Mathematica 40*, 3 (1995), 89–100.

[5] Pop, H. F. A new class of fuzzy algorithms: Fuzzy 1-Means. *Mathware and Soft Computing (Universitat Politecnica de Catalunya)*, (1996). To appear.

[6] Pop, H. F., Dumitrescu, D., and Sârbu, C. A study of Roman pottery (terra sigillata) using hierarchical fuzzy clustering. *Analitica Chimica Acta 310* (1995), 269–279.

[7] POP, H. F., AND SÂRBU, C. A new fuzzy regression algorithm. *Journal of Analytical Chemistry 68* (1996), 771–778.

[8] POP, H. F., AND SÂRBU, C. The fuzzy hierarchical cross-clustering algorithm. Improvements and comparative study. *Journal of Chemical Information and Computer Sciences* (1996). To appear.

[9] POP, H. F., SÂRBU, C., HOROWITZ, O., AND DUMITRESCU, D. A fuzzy classification of the chemical elements. *Journal of Chemical Information and Computer Sciences 36* (1996), 465–482.

[10] SÂRBU, C., DUMITRESCU, D., AND POP, H. F. Selecting and optimally combining the systems of solvents in the thin film cromatography using the fuzzy sets theory. *Revista de Chimie 44*, 5 (1993), 450–459.

[11] SÂRBU, C., HOROWITZ, O., AND POP, H. F. A fuzzy cross-classification of the chemical elements, based both on their physical, chemical and structural features. *Journal of Chemical Information and Computer Sciences 36* (1996), 1098–1108.

[12] SÂRBU, C., AND POP, H. F. Fuzzy classification of Greek muds. *The Analyst* (1996). Submitted.

"BABEŞ-BOLYAI" UNIVERSITY, FACULTY OF MATHEMATICS AND COMPUTER SCIENCE, RO-3400 CLUJ-NAPOCA, ROMANIA

E-mail address: hfpop@cs.ubbcluj.ro