COMISEF
Computational Optimization Methods in Statistics, Econometrics and Finance

- Marie Curie Research and Training Network funded by the EU Commission through MRTN-CT-2006-034270 -

COMISEF WORKING PAPERS SERIES

WPS-028 08/02/2010

Fuzzy clustering of univariate and multivariate time series by genetic multiobjective optimization

S. Bandyopadhyay
R. Baragona
U. Maulik

Fuzzy clustering of univariate and multivariate time series by genetic multiobjective optimization

S. Bandyopadhyay [1], R. Baragona [2] , U. Maulik [3]

Abstract: Given a set of time series, it is of interest to discover subsets that share similar properties. For instance, this may be useful for identifying and estimating a single model that may fit conveniently several time series, instead of performing the usual identification and estimation steps for each one. On the other hand time series in the same cluster are related with respect to the measures assumed for cluster analysis and are suitable for building multivariate time series models. Though many approaches to clustering time series exist, in this view the most effective method seems to have to rely on choosing some features relevant for the problem at hand and seeking for clusters according to their measurements, for instance the autoregressive coefficients, spectral measures or the eigenvectors of the covariance matrix. Some new indexes based on goodness-of-fit criteria will be proposed in this paper for fuzzy clustering of multivariate time series. A general purpose fuzzy clustering algorithm may be used to estimate the proper cluster structure according to some internal criteria of cluster validity. Such indexes are known to measure actually definite often conflicting cluster properties, compactness or connectedness, for instance, or distribution, orientation, size and shape. It is argued that the multiobjective optimization supported by genetic algorithms is a most effective choice in such a difficult context. In this paper we use the Xie-Beni index and the C-means functional as objective functions to evaluate the cluster validity in a multiobjective optimization framework. The concept of Pareto optimality in multiobjective genetic algorithms is used to evolve a set of potential solutions towards a set of optimal non-dominated solutions. Genetic algorithms are well suited for implementing difficult optimization problems where objective functions do not usually have good mathematical properties such as continuity, differentiability or convexity. In addition the genetic algorithms, as population based methods, may yield a complete Pareto front at each step of the iterative evolutionary procedure. The method is illustrated by means of a set of real data and an artificial multivariate time series data set.

Keywords: Fuzzy clustering, Internal criteria of cluster validity, Genetic algorithms, Multiobjective optimization, Time series, Pareto optimality

[1]Indian Statistical Institute, India. *Address for correspondence*: Machine Intelligence Unit, 203 B. T. Road, Kolkata 700 108, West Bengal, India. E-mail: sanghami@isical.ac.in

[2]Sapienza University of Rome, Italy. *Address for correspondence*: Department of Sociology and Communications, Via Salaria 113, I-00198, Rome, Italy. E-mail: roberto.baragona@uniroma1.it

[3]Jadavpur University, Kolkata, India. *Address for correspondence*: Department of Computer Science and Engineering, 188 Raja S.C. Mallik Road, Kolkata - 700 032, West Bengal, India. E-mail: drumaulik@cse.jdvu.ac.in

1 Introduction

Cluster analysis of univariate and multivariate time series is gathering increasing interest as wide application fields are arising where a large amount of time series data is encountered but no labeled data are available. Some important examples include data mining applications, event detection from time series data, economic indicators comparison across countries and regions, financial data investigation, medical data from sensor-based monitoring of patients affected by similar pathologies, data recorded at regular time intervals in geology or climatology.

This paper deals with the problem of clustering time series data. It consists of two essential steps, the first one concerned with feature extraction and the second one concerned with the allocation of time series to groups. Feature extraction involves the computation of the important characteristics of the time series that are intended to capture their essential properties. For instance, the data may provide us with suitable measurements to be transformed into features by using some mathematical transforms. Human eye may sometimes accomplish the task of distinguishing time series from one another at a glance. More often characteristics that really matter are hidden in the time series sample and numerical techniques may be devised that are more efficient and reliable. For instance, time series may come as a specially long data stretch and visual inspection may fail to provide us with a basis for judging to what extent time series are either different or similar to each other. Another relevant situation occurs when time series arrays have to be classified. It is apparent that human detection is unable to identify similar multivariate time series and detect properties that multivariate time series in a group are likely to share.

We assume that classes are not available in advance so that we have to perform an unsupervised classification task. In this case all information has to be gathered from the data and estimation techniques are needed for both distinguishing the time series and defining the features that best characterize the groups. Berkhin (2002) is a useful reference, amongst so many, that emphasizes connections between clustering and data mining. Surveys of methods for clustering time series may be found in Keogh and Kasetty (2003) and Liao (2005). An interesting application which is related to outliers in time series has been considered in Guralnik and Srivastava (1999). For non-stationary time series classification the SLEX model is proposed in Huang *et al.* (2004). Algorithms based on SLEX library are able to extract local (localized in both time and frequency domain) time series spectral features. A special spectral measure called cepstrum has been considered by Kalpakis *et al.* (2001). In Wang *et al.* (2005), an attempt has been made to select all time series characteristics that are believed to be really important and put them together to be used within a reliable and accurate clustering procedure. In the multivariate framework Radhakrishnan *et al.* (2004) formulate the problem of pattern discovery from video records as a vector time series clustering problem. Eigenvector analysis of similarity matrix, from principal component analysis, for instance, has been proposed for developing procedures for multivariate time series classification (Yang and Shahabi, 2004; Singhal and Seborg, 2005; Allefeld and Bialonski, 2007). Extensions of spectral measures for multivariate time series have been discussed by Kakizawa *et al.* (1998). Dissimilarity measures are based on indexes of disparity between spectral matrices computed for multivariate time series. In this paper

we present a new approach base on goodness-of-fit measures which allows multivariate time series of different dimensionality and length to be compared. Some R^2 criteria for linear interpolators and vector autoregressive (VAR) models will be introduced.

Cluster analysis is an application field where evolutionary computing has been repeatedly and successfully proposed. Evolutionary computing is used to denote stochastic population-based search methods. These methods are believed to be able to solve complex optimization problems which cannot be formulated in terms of mathematically well behaved objective functions. In case of small size problems complete enumeration of the feasible potential solution is viable. The only task that is left consists in calculating the objective function for each item of an exhaustive list. Then, the optimal solution may be chosen. For medium size problems dynamic programming, branch and bound algorithms for instance, may find the solution in a reasonable computing time. These are examples of deterministic methods that may lead to the exact solution provided that proper conditions are met. If the space of feasible potential solutions is large, exhaustive computation is ruled out soon. Deterministic methods, which include as well gradient-based methods when applicable, may lead to local optima instead of the global one either because the solution space is large and the algorithm does not have the capability of exploring all regions efficiently or there are many local optima such that the algorithm is confined to one of them at the end. Stochastic search is a powerful device to encompass local optima and weaken the mathematical constraints that the objective function is often not really able to satisfy. Moreover, if several patterns are allowed to be explored simultaneously, chances of successfully attaining the global optimal solution increase considerably. These methods are included within the evolutionary computing methods as they mimic the evolution of a population toward adaptation to the environment. The most representative examples of this class are genetic algorithms (GAs). The GAs (Holland, 1975) start with a population chosen at random, then selection and "genetic operators" such as mutation and crossover drive the evolution of the population through several generations. Latest generations are believed to include the individuals that encode better solutions. More precisely, individuals in each generation encode solutions that on the average are closer approximations to the optimal solution than those provided by the preceding generation. Applications to statistical estimation problems have been considered by Pasia *et al.* (2005, for instance). Efficient GA-based techniques have been suggested for general clustering problems (Bezdek and Aathaway, 1994; Bandyopadhyay and Maulik, 2002; Maulik and Bandyopadhyay, 2000, 2003). Evolutionary computing algorithms may be modified to incorporate features from either general heuristic methods (Winker and Gilli, 2004) or deterministic and gradient-based methods. General heuristics are, for instance, the ant colony optimization (Handl *et al.*, 2003), swarm intelligence (Handl and Meyer, 2007), tabu search (Al-Sultan, 1995) and simulated annealing (Kirkpatrick *et al.*, 1983; Brooks and Morgan, 1995). Hybridization of GAs has been suggested with gradient-based methods, for instance. In this latter case the stochastic search is devoted to explore promising regions of the solution space. Then, steepest descent algorithms, for instance, may start from some initial point close enough to the global optimum to refine the search. Heuristics for clustering time series have been examined, including GAs and hybrid GAs, in Baragona *et al.* (2001), for instance. Often cluster validity had

better evaluated by using several possibly conflicting indexes of cluster validity (Bezdek and Pal, 1998) that need to be simultaneously optimized. For such problems often a single optimum solution does not exist and we have a solution set that consists of several alternative solutions to the problems. In Deb (2001); Bandyopadhyay *et al.* (2007a,b); Handl and Knowles (2007) multiobjective clustering methods are introduced that use the search capability of genetic algorithms. A new model of multiobjective simulated annealing algorithm called AMOSA (Bandyopadhyay *et al.*, 2008) has been developed as well.

The paper is organized as follows. The Pareto optimality in the multiobjective optimization framework is introduced in Section 2. The GAs for solving the multiobjective optimization problem is introduced in Section 3. Experimental results from the applications of the fuzzy multiobjective clustering algorithm NSGAII (Deb, 2001) on univariate and multivariate time series data sets are reported in Sections 4 and 5, respectively. Conclusions are drawn in Section 6.

2 Multiobjective optimization and Pareto optimality

Application of techniques having physical or natural correspondence for solving difficult optimization problems has been receiving widespread attention for the last two decades. It has been found that these techniques consistently outperform classical methods like gradient descent search when the search space is large, complex and multimodal. Simulated annealing (SA) is one such paradigm having its foundation in statistical mechanics, which studies the behavior of a very large system of interacting components in thermal equilibrium. Genetic algorithms, another popular search technique, mimics the principles of natural genetic systems, including inheritance and the Darwin's theory of selection of the fittest. Some other popular techniques are particle swarm optimization, differential evolution, ant colony optimization, etc (Konar, 2005). Though traditionally these methods have been used to solve single objective optimization problems, their applicability in solving multiobjective problems is generating wide interest in recent times.

The multiobjective optimization may be formally stated as Deb (2001); Coello Coello (1999); Zitzler and Thiele (1998); Zitzler *et al.* (2001); Bandyopadhyay *et al.* (2008): Find the vectors $x = [x_1; x_2; \ldots; x_n]'$ of decision variables that satisfy the m inequality constraints

$$g_i(x) \geq 0, \qquad i = 1, 2, \ldots, m, \tag{1}$$

the p equality constraints

$$h_i(x) = 0, \qquad i = 1, 2, \ldots, p, \tag{2}$$

and simultaneously optimize the O objective functions

$$\{f_1(x), f_2(x), \ldots, f_O(x)\}.$$

The constraints given in Eqns. (1) and (2) define the feasible region F which contains all the admissible solutions. Any solution outside this region is inadmissible since it violates one or more constraints. The vector x denotes an

optimal solution in F. In the context of multiobjective optimization the difficulty lies in the definition of optimality, since it happens rarely that a single vector x represents the optimum solution to all the O objective functions.

An important concept of multiobjective optimization is that of domination. Most multiobjective optimization techniques use the concepts of dominance relation and Pareto optimality. We give the formal definitions here corresponding to a maximization problem. The definitions are easily extended to minimization problems. A solution x_i is said to dominate x_j if

$$\forall k \in \{1, 2, \ldots, O\} \quad f_k(x_i) \geq f_k(x_j)$$

and

$$\exists k \in \{1, 2, \ldots, O\} \quad \text{such that } f_k(x_i) > f_k(x_j).$$

Among a set of solutions P, the non-dominated set of solutions P' are those that are not dominated by any member of the set P. A solution x is said to be non-dominated in P if there exist no solution x^* which dominates x. The non-dominated set of the entire search space S is the globally Pareto optimal set. On many occasions, the globally Pareto optimal set is simply referred to as the Pareto optimal set. If for every member x in a set P there exists no solution y (in the neighborhood of x such that $\|y - x\|_\infty \leq \epsilon$, where ϵ is a small positive number) dominating any member of the set P, then solutions belonging to the set P constitute a locally Pareto optimal set. In general, a multiobjective optimization algorithm usually admits a set of solutions that are not dominated by any solution encountered by it.

There are different approaches to solving multiobjective optimization problems e.g. aggregating population based non-Pareto and Pareto-based techniques (Deb, 2001; Coello Coello, 1999). In aggregating techniques, the different objectives are generally combined into one using weighting or goal based method. Vector evaluated genetic algorithm (VEGA) is a technique in the population based non-Pareto approach in which different sub-populations are used for the different objectives. Multiple objective GA (MOGA), non-dominated sorting GA (NSGA), niched Pareto GA constitute a number of techniques under the Pareto based approaches. However, all these techniques, described in Deb (2001) were essentially non-elitist in nature. NSGAII (Deb, 2001), SPEA (Zitzler and Thiele, 1998) and SPEA2 (Zitzler *et al.*, 2001) are some relatively recently developed multiobjective elitist techniques.

3 Genetic algorithms for multiobjective optimization

This section describe a multiobjective fuzzy partitioning technique that exploit the searching capability of GAs for the purpose of clustering time series data sets.

3.1 Multiobjective fuzzy clustering technique

In this section, we describe the use of NSGAII for evolving a set of near-Pareto-optimal non-degenerate fuzzy partition matrices. The Xie-Beni index (XB) (Xie

and Beni, 1991) and the C-means functional J_m (Bezdek, 1981) are well known fuzzy cluster validity indexes that will be considered as the objective functions to be minimized simultaneously. The technique is described below in detail.

3.1.1 String representation and population initialization

Here the chromosomes are made up of real numbers which represent initially the features of the time series assumed as centers of the partitions. For C clusters, the centers encoded in a chromosome in the initial population are randomly selected C distinct time series from the data set. There is a one-to-one correspondence between each time series and its features.

3.1.2 Computing the objectives

For computing the XB and J_m indexes the centers encoded in a chromosome, corresponding to the selected features, are first extracted. We assume as an example that the extracted features, called variables afterwards, are the coefficients of the autoregressive model fitted to each time series and the time delay between each time series and a reference series. Let the centers be denoted by $v_1, v_2, \ldots, v_C$. The membership values u_{ik}, $i = 1, 2, \ldots, C$ and $k = 1, 2, \ldots, n$ are computed as follows (Bezdek, 1981):

$$u_{ik} = \frac{1}{\sum_{j=1}^{C} \left(\frac{D(v_i, x_k)}{D(v_j, x_k)}\right)^{\frac{2}{m-1}}}, \quad \text{for} \quad 1 \leq i \leq C; \quad 1 \leq k \leq n, \tag{3}$$

where $D(v_i, x_k)$ and $D(v_j, x_k)$ are distances between centers and variables. m is the weighting coefficient. Note that while computing u_{ik} using Equation (3), if $D(v_j, x_k)$ is equal to zero for some j, then u_{ik} is set to zero for all $i = 1, \ldots, C$, $i \neq j$, while u_{jk} is set equal to one. Subsequently, the centers encoded in a chromosome are updated using the following equation (Bezdek, 1981)

$$v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m x_k}{\sum_{k=1}^{n} (u_{ik})^m}, \quad 1 \leq i \leq C, \tag{4}$$

and the cluster membership values are recomputed. We may define

$$J_m = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^m D^2(v_i, x_k). \tag{5}$$

The XB index is defined as a function of the ratio of the total variation σ to the minimum separation *sep* of the clusters. Here σ and *sep* can be written as

$$\sigma(U, V; X) = \sum_{i=1}^{c} \sum_{k=1}^{n} u_{ik}^2 D^2(v_i, x_k), \tag{6}$$

and

$$sep(V) = \min_{i \neq j} \{||v_i - v_j||^2\}, \tag{7}$$

where $||.||$ is the Euclidean norm, and $D(v_i, x_k)$, as mentioned earlier, is the distance between the pattern x_k and the cluster center v_i. The XB index is

then written as

$$XB(U,V;X) = \frac{\sigma(U,V;X)}{n\ sep(V)} = \frac{\sum_{i=1}^{c}(\sum_{k=1}^{n} u_{ik}^2 D^2(v_i,x_k))}{n(\min_{i\neq j}\{||v_i - v_j||^2\})}. \quad (8)$$

Note that when the partitioning is compact and good, value of σ should be low while *sep* should be high, thereby yielding lower values of the XB index. The objective is therefore to minimize either XB or J_m for achieving proper clustering.

3.1.3 Considering both time delays and autoregressive coefficients

When only the time delays or the autoregressive coefficients are considered separately, computing the distance between the center of the ith cluster, denoted by v_i, and the kth time series, denoted by x_k is straightforward.

When both these parameters are considered together, then the distance $D(v_i, x_k)$ is computed as follows:

$$D(v_i,x_k) = w_1 \times \frac{D(v_{idelay}, x_{kdelay})}{max\_delay} + w_2 \times \frac{D(v_{iauto}, x_{kauto})}{max\_auto}, \quad (9)$$

where v_{idelay} and x_{kdelay} are the delay parameters of the cluster center and the individual time series, while v_{iauto} and x_{kauto} are the corresponding autoregressive coefficients. The factors w_1 and w_2 are the weighting coefficients indicating the relative importance of the two parameters. For the purpose of this article, these are set to 0.5. The factors max_delay and max_auto are the maximum values of the distances based on delays and autoregressive coefficients, respectively, between any two time series in the data. These are computed from the input data, and are used for the purpose of normalizing the two distances.

Once the combined distances of a time series x_k from all the cluster centers are computed, the one with the minimum distance is identified and x_k is assigned to that cluster. In this way, all the time series data points are partitioned, and the XB indexes corresponding to time delays and autoregressive coefficients are computed. Subsequently, the centers are updated as in Equation 4.

3.1.4 Genetic operators

NSGAII (Deb, 2001) is used as the underlying multiobjective optimization algorithm. Here, initially a random parent population P_0 of size N is created. Then the population is sorted based on the non-domination relation. Each solution of the population is assigned a fitness which is equal to its non-domination level. A child population is created from the parent population by using binary tournament selection, recombination, and mutation operators. Generally according to this algorithm, the parent and the child populations are combined. Thereafter, all the solutions are sorted based on their non-domination status. If the total number of solutions belonging to the best non-dominated set F_1 is smaller than N, F_1 is completely included into the next population. The remaining members of the population are chosen from subsequent non-dominated fronts in the order of their ranking. A crowded comparison operator based on crowding distance is used for this purpose. Thereafter, selection, crossover and mutation are used to create a new population of size N, and the process continues. The crowding distance operator is used to maintain diversity in the Pareto front. For details

on the different genetic processes, the reader may refer to Deb (2001). The near-Pareto-optimal strings of the last generation provide the different solutions to the clustering problem.

4 Experimental results on univariate time series data

In this section, we first describe the univariate time series data set that has been used in the experiments. Thereafter, the results of the multiobjective fuzzy clustering techniques is reported.

4.1 Data set

Let us consider the monthly indexes of the industrial production in Italy related to 24 branches (base 1990=100), recorded from 1971:1 through 1996:3 ($T = 303$ observations for each of $K = 24$ time series). Data have been drawn from the database of Italy's National Statistical Institute (ISTAT). In Table 1 the time series are listed and their content is accounted for according to the NACE-CLIO classification system.

We adjusted data for outliers and computed their (natural) logarithmic transform. The following $ARIMA(2,0,0) \times (0,1,0)_{12}$ model

$$(1 - \phi_1 B - \phi_2 B^2)(1 - B^{12})x_{j,t} = a_{j,t}, \qquad j = 1, \ldots, K,$$

where B is the back-shift operator ($Bx_{j,t} = x_{j,t-1}$) and $\{a_{j,t}\}$ is a zero mean white noise, has been fitted to the time series data set $\{x_{j,t}\}$, $j = 1, \ldots, K$.

The first feature is the model structure with $n = 2$ measurements $\pi_{j,1} = \phi_1$ and $\pi_{j,2} = \phi_2$. The autoregressive coefficients, or π-weights in the present context, have been introduced by Piccolo (1990) as a time series feature suitable for cluster analysis. The second feature we considered takes into account the time series alignment. As a matter of fact it is often of interest to classify time series whether they are leading or lagging with respect to a reference time series either to compute composite indexes of economic activities or for modeling and forecasting purpose. A suitable measure of such lead-lag relationships is the time delay (see Cleveland and Parzen, 1975, for instance). The time delay $\{d_{j,\lambda}\}$ varies with the frequency λ, $-\pi < \lambda \leq \pi$, so that for each choice of λ a feature measure is available. We confine our attention to the long-term components of each time series and consider the frequencies $\lambda_1 = \frac{2\pi}{128}$ and $\lambda_2 = \frac{2\pi}{27}$. These frequencies correspond to periods of 128 and 27 months respectively, the longest and the shortest cycle ever recorded in Italy since 1945. Then, we assume $n = 2$ measurements $d_j(\lambda_1)$ and $d_j(\lambda_2)$, $j = 1, \ldots, K$.

In Figures 1 and 2 the scatter plots of the autoregressive weights and of the time delays are displayed. Plots look rather different, so we may expect that the two features correspond to rather different time series properties.

4.2 Implementation Result

Results reported in Baragona *et al.* (2001) show that using the autoregressive weights most of the methods considered therein yield the 3 clusters

Table 1: Monthly indexes of industrial production, branches of economic activities according to the NACE-CLIO classification. Italy, 1971:1 – 1996:3

label	branch of activity
1	coal
2	coke ovens
3	mineral oil refining
4	energy
5	production and preliminary processing of metals
6	non-metallic mineral products
7	chemicals (reference series)
8	metal articles
9	mechanical engineering
10	office and data processing machinery
11	electrical engineering
12	motor vehicles
13	other means of transport
14	meat
15	dairy products
16	other foods
17	drink
18	tobacco
19	textiles
20	leather and leather goods, footwear and clothing
21	timber and wooden furniture
22	paper and paper products, printing and publishing
23	rubber and plastic products
24	other manufacturing industries

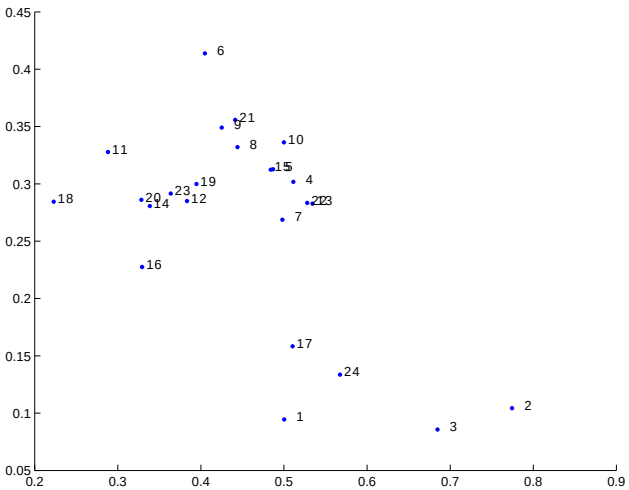

Figure 1: Scatter plot of autoregressive weights $\pi_{j,1}$ (horizontal axis) and $\pi_{j,2}$ (vertical axis), $j = 1, \ldots, 24$

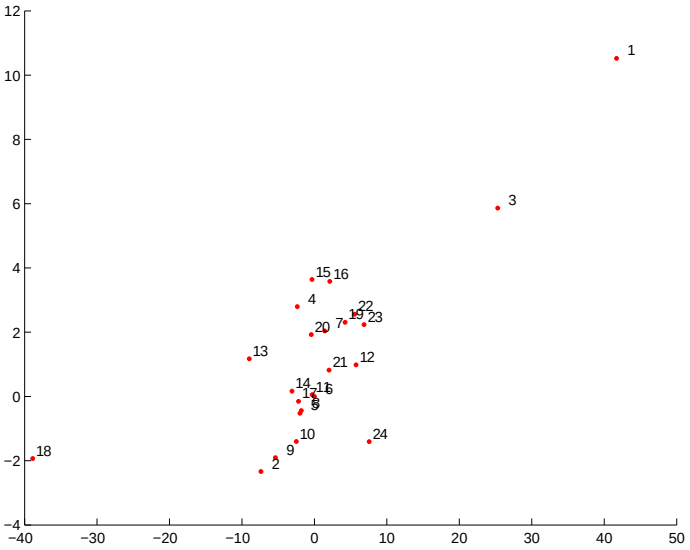

Figure 2: Scatter plot of time delays $d_j(\lambda_1)$ (horizontal axis) and $d_j(\lambda_2)$ (vertical axis), $j = 1, \ldots, 24$

{1, 2, 3, 17, 24}
{4, 5, 6, 7, 8, 9, 10, 13, 15, 21, 22}
{11, 12, 14, 16, 18, 19, 20, 23}.

The same methods applied to the time delay features yield 2 or 3 clusters. Almost always a single cluster groups most of the time series while the other 2 include a limited number of time series. A typical grouping that is rather apparent in Figure 2 is, for instance,

{1, 3}
{18}
{the remaining time series}.

The 24 time series are clustered here using both the autoregressive coefficients and time delays with equal weights and the fuzzy partition objective. The algorithm resulted automatically into 3 groups, comprising the following time series:

{1, 3}
{2, 4, 5, 6, 7, 8, 9, 10, 13, 15, 17, 24}
{11, 12, 14, 16, 18, 19, 20, 23}

This result may be considered a good compromise between the partition formed according to the autoregressive coefficients and the partition formed according to the time delays. This latter includes a large cluster which contains most of the time series set and other very small clusters. The partition obtained by the multiobjective fuzzy genetic clustering contains a first cluster which includes the time series 1 and 3. This is possibly due to the large delays that characterize the two basic energy sources that produce 81.14% (Italy, 2004) of total electric power. The positive delays mean that these are leading series. The other two

clusters composition is similar to that obtained by using only the autoregressive coefficients for clustering but time series 2, 17 and 24 are added more reliably to the second cluster. This latter includes industrial production indexes of heavy industries plus the dairy products, drink and the residual series other manufacturing industries. These time series are in general lagging (7 months at most) or coincident. The third cluster is identical to that obtained using only the autoregressive coefficients. It includes in general leading time series (no more than 7 months) that seem rather heterogeneous as motor vehicles, some food-processing industry branches, textiles and plastic products all show together. In this case clustering is driven by the common time series structure as independently of their meaning similar models may be fitted to the time series data.

5 Clustering multivariate time series data

Specific concepts that arise in the multivariate framework may offer convenient tools for clustering multivariate time series, for example vector autoregressive (VAR) modeling, cointegration, vector linear interpolators and spectral matrices. R^2 measures have been discussed by Nelson (1976) and Pierce (1979) that extend to time series data the well known R^2 statistic defined in the context of linear regression analysis. Interpreting R^2 in a time series framework correctly requires careful examination of the relationships with underlying model parameters and correlation structure and implies that any time series is associated an inherent predictability. This latter allows time series to be distinguished using R^2-based features that are both meaningful and easy to compute by using widely available statistical packages. Following Box and Tiao (1977) we introduce two features for multivariate time series that account for predictability the first one and for interpolability the second one. Let $z = [z_1, \ldots, z_n]$ denote an observed p-dimensional multivariate time series where $z_t = (z_{1t}, \ldots, z_{pt})'$ and assume that z follows the ℓth order VAR model (see Reinsel, 1993, for instance)

$$z_t = \hat{z}_{t-1}(1) + a_t, \tag{10}$$

where

$$\hat{z}_{t-1}(1) = \sum_{i=1}^{\ell} \Pi_i z_{t-i} \tag{11}$$

is the expectation of z_t conditional on its past values, the Π_i's are $p \times p$ matrices and $\{a_t\}$ is a sequence of independent and identically distributed random variables independent of $\hat{z}_{t-1}(1)$ with mean zero and covariance matrix Σ. Let

$$\Gamma_j(z) = \mathbb{E}\,(z_{t-j} {z_t}')$$

denote the lag j autocovariance matrix of z_t. By using (10) and (11) it follows

$$\Gamma_0(z) = \Gamma_0(\hat{z}) + \Sigma.$$

We may reduce to the univariate framework by considering the linear combination $u_t = \beta' z_t$. From (10) it follows $u_t = \hat{u}_{t-1}(1) + v_t$, where $\hat{u}_{t-1}(1) = \beta' \hat{z}_{t-1}(1)$ and $v_t = \beta' a_t$. The predictability of u_t from its past may be computed as the

Table 2: Measures of predictability (VAR) and interpolability (linear interpolator) for multivariate time series

index	vector autoregression	linear interpolator
A-optimality	$\text{trace}\left(\Gamma(\hat{z})\right)/\text{trace}\left(\Gamma(z)\right)$	$\text{trace}\left(\Gamma(\tilde{z})\right)/\text{trace}\left(\Gamma(z)\right)$
E-optimality	$\rho\left(\Gamma(z)^{-1}\Gamma(\hat{z})\right)$	$\rho\left(\Gamma(z)^{-1}\Gamma(\tilde{z})\right)$
D-optimality	$\det\left(\Gamma(\hat{z})\right)/\det\left(\Gamma(z)\right)$	$\det\left(\Gamma(\tilde{z})\right)/\det\left(\Gamma(z)\right)$

R^2 measure $\lambda = \sigma^2_{\hat{u}}/\sigma^2_u$, where σ^2_u and $\sigma^2_{\hat{u}}$ are the variance of u_t and $\hat{u}_{t-1}(1)$ respectively. Predictability is maximized by assuming $\lambda > 0$ as larger as possible, that is

$$\lambda = \frac{\beta'\Gamma_0(\hat{z})\beta}{\beta'\Gamma_0(z)\beta}$$

has to be chosen equal to the largest eigenvalue of $\Gamma_0^{-1}(z)\Gamma_0(\hat{z})$. By analogy with optimal experimental design (Kiefer, 1959) we may call this index the E-optimality criterion. The analogy may be extended further to define the indexes $\text{trace}\left(\Gamma_0(\hat{z})\right)/\text{trace}\left(\Gamma_0(z)\right)$ and $\det\left(\Gamma_0(\hat{z})\right)/\det\left(\Gamma_0(z)\right)$ as A-optimality and D-optimality respectively.

Likewise, let the vector autoregressive model (11) be replaced by the multivariate linear interpolator model (Battaglia, 1984)

$$\tilde{z}_{t-1}(1) = \sum_{i=1}^{m} \Phi_i \left(z_{t-i} + z_{t+i}\right), \tag{12}$$

where m denotes the linear interpolator order. The A-optimality, E-optimality and D-optimality indexes for maximum interpolability are then defined using the variance-covariance matrices computed from (12). The indexes of predictability and interpolability are displayed in Table 2. ρ denotes the largest eigenvalue (absolute value). Clustering a set of multivariate time series may be done by computing both models (11) and (12) for each multivariate time series and choosing for each of the two some or all of the associated measurements of predictability and interpolability.

5.1 An artificial data set

For the purpose of experiment we generated from two VAR models 20 artificial time series and performed the cluster analysis using the genetic multiobjective method. 10 4-dimensional time series have been generated by a VAR of order 1 and 10 2-dimensional time series have been generated by a VAR of order 2. In Figures 3 and 4 a sample from the first and second cluster respectively is displayed.

In Figures 5 and 6 the predictability and interpolability optimality features extracted from the 20 artificial time series are plotted for each of the A, E and D optimality criteria. It is apparent that the D-optimality features are rather far apart between the two multivariate time series groups while the A-optimality features are close together for the 4-dimensional time series and either smaller or larger for the 2-dimensional time series, and E-optimality features are rather

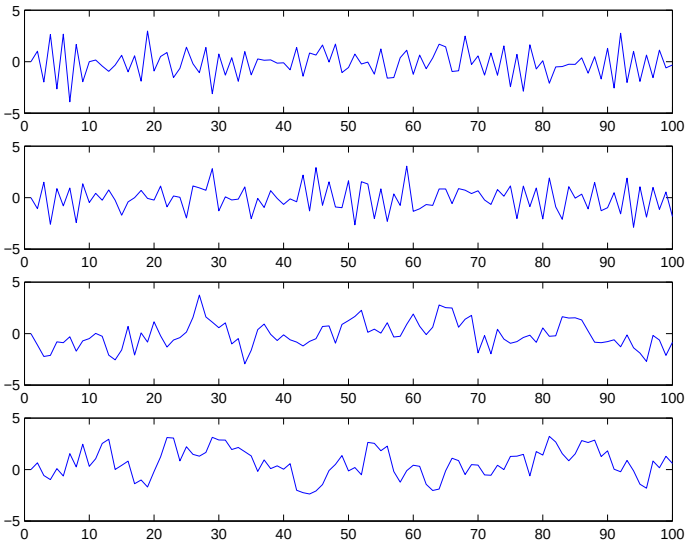

Figure 3: Plot of 100 observations generated from a 4-dimensional VAR model of order 1 and zero mean white noise with variance matrix I_4

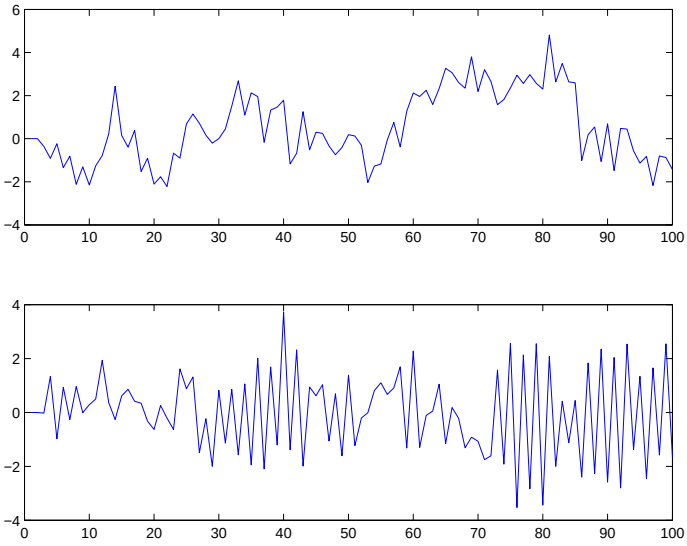

Figure 4: Plot of 100 observations generated from a 2-dimensional VAR model of order 2 and zero mean white noise with variance matrix I_2

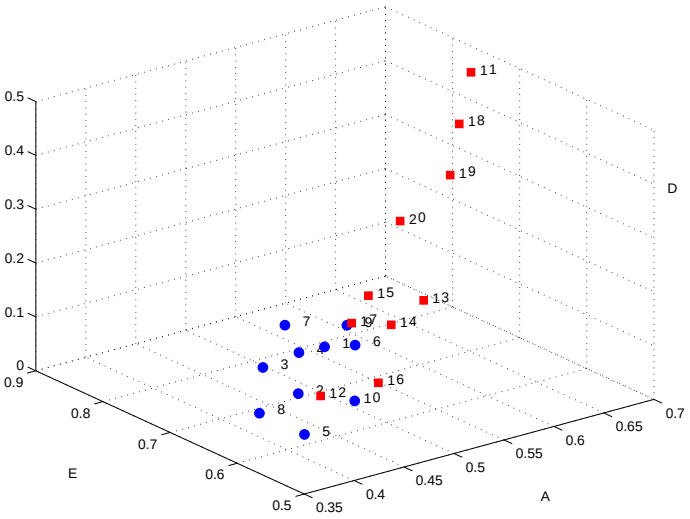

Figure 5: Scatter plot of A, E and D-optimality features related to predictability extracted from 20 artificial times series. The points that correspond to the 4-dimensional time series in the first cluster are plotted as circles (blue), those for the 2-dimensional time series that form the second cluster as squares (red)

mixed between the two groups but for three 2-dimensional time series which display large measurements.

The two clusters seem well separated according to the predictability measurements. In particular the predictability measurements based on the D-optimality criterion are markedly different each other. Some overlaps occur, namely in Figure 5 the measurements from the time series 12, 16, and 17 which belong to the second cluster are either mixed or close to the measurements from the time series in the first cluster. The points corresponding to the time series 9 and 17 are nearly coincident though the time series belong actually to different clusters. The two clusters are better displayed in Figure 6 where the point coordinates are the interpolability measurements. In this case only the points that correspond to the time series 9 and 14 nearly overlap though they may be distinguished easily.

5.2 Implementation Result

Table 3 report the result of application of multiobjective fuzzy clustering for the multivariate time series data for evolving the cluster centers. Here, the predictability and interpolability features of the centers are mentioned in terms of A-optimality, E-optimality and D-optimality.

In Figure 7 the centers computed according to interpolability and predictability criteria by fuzzy clustering algorithms are displayed.

The predictability centers seem influenced only by the A and E optimality indexes as the D-optimality indexes are close to zero (see Table 3). The interpolability indexes are more balanced as regards their contribution to the centers location. It may be inferred that the genetic multiobjective fuzzy partition algorithm is able to exploit successfully the joint contribution of the whole

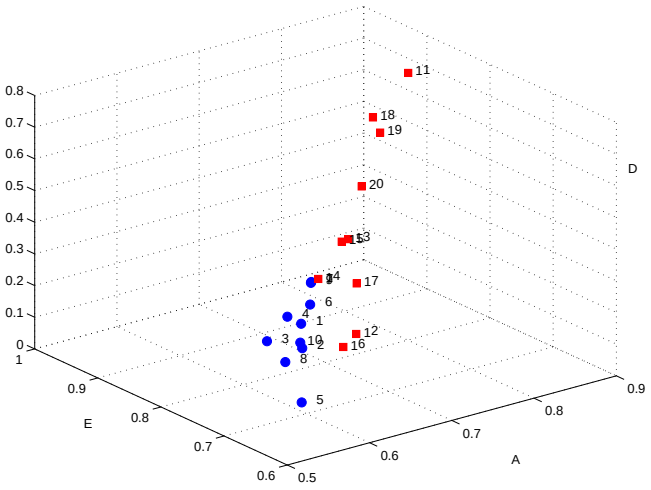

Figure 6: Scatter plot of A, E and D-optimality features related to interpolability extracted from 20 artificial times series. The points that correspond to the 4-dimensional time series in the first cluster are plotted as circles (blue), those for the 2-dimensional time series that form the second cluster as squares (red)

Table 3: Centers of clusters computed by the multiobjective fuzzy clustering algorithm according to 3 features of predictability and 3 of interpolability for multivariate time series

	vector autoregression			linear interpolator		
cluster	A-index	E-index	D-index	A-index	E-index	D-index
1	0.4877	0.7052	0.0232	0.6491	0.7806	0.1243
2	0.5185	0.6224	0.1868	0.6908	0.7411	0.4174

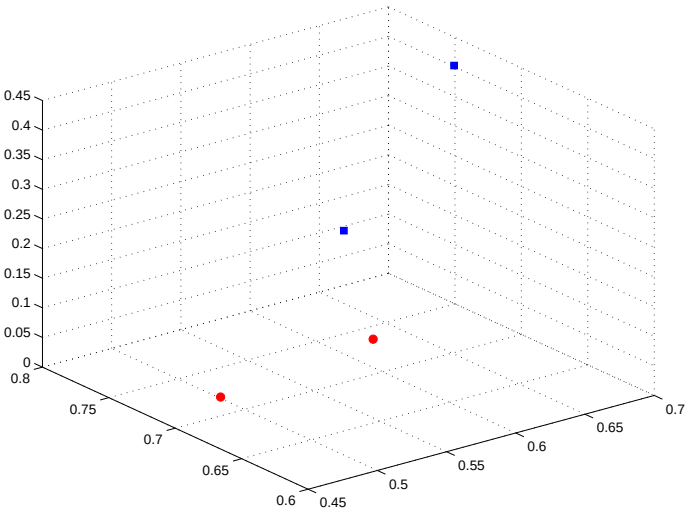

Figure 7: 3-dimensional scatter plot of the centers of the two clusters that group 20 artificial multivariate time series according to several criteria. Centers are calculated according to predictability fuzzy partition (circles - red) and interpolability fuzzy partition (squares - blue)

set of indexes. As a matter of fact taking all indexes into account properly is needed. For instance let the D-optimality index alone be considered as it seems to guarantee the neat separation of the two time series clusters. Then at least one time series is likely to be misplaced (the time series 9 in either Figures 5 or 6 seems the most obvious candidate). Using the other two indexes may set right the issue.

6 Concluding remarks

In this paper techniques for clustering univariate and multivariate time series are proposed. For clustering, two basic steps are carried, i.e., features extraction, and assignment of time series to clusters according to an optimality criterion. Many time series features have been suggested in the literature to represent interesting characteristics. Their selection is mainly problem-dependent and in principle features are to be preferred that allow measurements to be calculated with limited computational effort. On the other hand, assignment of time series to clusters in the presence of multiple indexes of cluster validity calls for multiobjective optimization algorithms. The Pareto optimality has been introduced as a valuable criterion to solve optimization problems where several often conflicting objectives have to be taken into account simultaneously. The computational burden may become considerable and many algorithms useful for solving problems of moderate size are hardly applicable in practice as they turn out to be too much demanding as regards computational resources. The meta heuristic methods are probabilistic algorithms that may speed up computation as far as optimization steps are concerned, so that most of the computing time is merely spent for objective functions computation. We applied genetic algorithms be-

cause these are the meta heuristic most popular as regards clustering problems and a considerable amount of knowledge is available. Furthermore genetic algorithms may be used advantageously in the Pareto optimization framework as the full Pareto front may be estimated in a single step of the algorithm. A fuzzy clustering approach have been considered for clustering univariate and multivariate time series data. Implementing Pareto optimality concepts deserves further research on theoretical explanation of algorithms behavior and more comparison studies on the practical side.

Acknowledgments

The present research is financially supported by Sapienza University of Rome, Italy. R. Baragona gratefully acknowledges financial support from the EU Commission through MRTN-CT-2006-034270 COMISEF research project.

References

Al-Sultan K.S. (1995) A tabu search approach to the clustering problem, *Pattern Recognition*, 28, 1443–1451.

Allefeld C. and Bialonski S. (2007) Detecting syncronization clusters in multivariate time series via coarse-graining of Markov chains, Technical report, arXiv:0707.2479v1 [physics.data-an] 17 Jul 2007.

Bandyopadhyay S. and Maulik U. (2002) Genetic clustering for automatic evolution of clusters and application to image classification, *Pattern Recognition*, 35, 1197–1208.

Bandyopadhyay S., Maulik U. and Mukhopadhyay A. (2007a) Multiobjective genetic clustering for pixel classification in remote sensing imagery, *IEEE Transactions on Geoscience and Remote Sensing*, 45, 1506–1511.

Bandyopadhyay S., Mukhopadhyay A. and Maulik U. (2007b) An improved algorithm for clustering gene expression data, *Bioinformatics*, 23, 2859–2865.

Bandyopadhyay S., Saha S., Maulik U. and Deb K. (2008) A simulated annealing based multi-objective optimization algorithm: AMOSA, *IEEE Transaction on Evolutionary Computation*, 12, 269–283.

Baragona R., Battaglia F. and Calzini C. (2001) Clustering of time series with genetic algorithms, *Metron*, 59, 113–130.

Battaglia F. (1984) Inverse covariances of a multivariate time series, *Metron*, 42, 117–129.

Berkhin P. (2002) Survey of clustering data mining techniques, Technical report, Accrue Software, San Jose, California, http://citeseer.nj.nec.com/berkhin02survey.html.

Bezdek J.C. (1981) *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum, New York.

Bezdek J.C. and Aathaway R.J. (1994) Optimization of fuzzy clustering criteria using genetic algorithms, in: *Proceedings of the First IEEE Conference on Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence.*, 589–594.

Bezdek J.C. and Pal N.R. (1998) Some new indexes of cluster validity, *IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics*, 28, 301–315.

Box G.E.P. and Tiao G.C. (1977) A canonical analysis of multiple time series, *Biometrika*, 64, 355–365.

Brooks S. and Morgan B. (1995) Optimization using simulated annealing, *The Statistician*, 44, 241–257.

Cleveland W.S. and Parzen E. (1975) The estimation of coherence, frequency response, and envelope delay, *Technometrics*, 17, 167–172.

Coello Coello C.A. (1999) A comprehensive survey of evolutionary-based multiobjective optimization techniques, *Knowledge and Information Systems*, 1, 3, 129–156.

Deb K. (2001) *Multi-objective Optimization Using Evolutionary Algorithms*, John Wiley and Sons, Ltd, England.

Guralnik V. and Srivastava J. (1999) Event detection from time series data, in: *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, San Diego, CA, 33–42.

Handl J. and Knowles J. (2007) An evolutionary approach to multiobjective clustering, *IEEE Transactions on Evolutionary Computation*, 11, 56–76.

Handl J., Knowles J. and Dorigo M. (2003) Ant-based clustering: a comparative study of its relative performance with respect to k-means, average link and 1 d-som, Technical Report TR/IRIDIA/2003-24, IRIDIA, Universit Libre de Bruxelles, Belgium, http://wwwcip.informatik.uni-erlangen.de/ŝijuhand/TR-IRIDIA-2003-24.pdf.

Handl J. and Meyer B. (2007) Ant-based and swarm-based clustering, *Swarm Intelligence*, 1, 95–113.

Holland J.H. (1975) *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor.

Huang H.Y., Ombao H. and Stoffer D.F. (2004) Discrimination and classification of nonstationary time series using the SLEX model, *Journal of the American Statistical Association*, 99, 763–774.

Kakizawa Y., Shumway R.H. and Taniguchi M. (1998) Discrimination and clustering for multivariate time series, *Journal of the American Statistical Association*, 93, 328–340.

Kalpakis K., Gada D. and Puttagunta V. (2001) Distance measures for effective clustering of ARIMA time series, in: *Proceedings of the 2001 IEEE International Conference on Data Mining (ICDM'01)*, San Jose, CA, 273–280.

Keogh E. and Kasetty S. (2003) On the need for time series data mining benchmarks: A survey and empirical demonstration, *Data Mining and Knowledge Discovery*, 7, 349–371.

Kiefer J. (1959) Optimum experimental designs, *Journal of the Royal Statistical Society, Series B*, 21, 272–319.

Kirkpatrick S., Gelatt C. and Vecchi M. (1983) Optimization by simulated annealing, *Science*, 220, 671–680.

Konar A. (2005) *Computational Intelligence: Principles, Techniques and Applications*, Springer-Verlag.

Liao T.W. (2005) Clustering of time series data - a survey, *Pattern Recognition*, 38, 1857–1874.

Maulik U. and Bandyopadhyay S. (2000) Genetic algorithm based clustering technique, *Pattern Recognition*, 33, 1455–1465.

Maulik U. and Bandyopadhyay S. (2003) Fuzzy partitioning using real coded variable length genetic algorithm for pixel classification, *IEEE Transactions on Geosciences and Remote Sensing*, 41, 1075–1081.

Nelson C.R. (1976) The interpretation of R^2 in autoregressive-moving average time series models, *The American Statistician*, 30, 175–180.

Pasia J.M., Hermosilla A.Y. and Ombao H. (2005) A useful tool for statistical estimation: genetic algorithms, *Journal of Statistical Computation and Simulation*, 75, 237–251.

Piccolo D. (1990) A distance measure for classifying ARIMA models, *Journal of Time Series Analysis*, 11, 153–164.

Pierce D.A. (1979) R^2 measures for time series, *Journal of the American Statistical Association*, 74, 901–910.

Radhakrishnan R., Divakaran A. and Xiong Z. (2004) A time series clustering based framework for multimedia mining and summarization, Technical Report TR2004-046 December 2004, MITSUBISHI ELECTRIC RESEARCH LABORATORIES.

Reinsel G.C. (1993) *Elements of Multivariate Time Series Analysis*, Springer, New York.

Singhal A. and Seborg D.E. (2005) Clustering multivariate time-series data, *Journal of Chemometrics*, 19, 427–438.

Wang X., Smith K.A. and Hyndman R.J. (2005) Characteristic-based clustering for time series data, *Data Mining and Knowledge Discovery*, 13, 335–364.

Winker P. and Gilli M. (2004) Applications of optimization heuristics to estimation and modelling problems, *Computational Statistics & Data Analysis*, 47, 211–223.

Xie X.S. and Beni G. (1991) A validity measure for fuzzy clustering, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13, 841–847.

Yang K. and Shahabi C. (2004) A PCA-based similarity measure for multivariate time series, in: *MMDB '04: Proceedings of the 2nd ACM international workshop on Multimedia databases*, ACM, New York, 65–74.

Zitzler E., Laumanns M. and Thiele L. (2001) SPEA2: Improving the Strength Pareto Evolutionary Algorithm, Technical Report 103, Gloriastrasse 35, CH-8092 Zurich, Switzerland.

Zitzler E. and Thiele L. (1998) An evolutionary algorithm for multiobjective optimization: The strength pareto approach, Technical Report 43, Gloriastrasse 35, CH-8092 Zurich, Switzerland.