Different Sequential Clustering Algorithms and Sequential Regression Models

Sadaaki Miyamoto, *Member, IEEE*, Kenta Arai

***Abstract*— Three approaches to extract clusters sequentially so that the specification of the number of clusters beforehand is unnecessary are introduced and four algorithms are developed. First is derived from possibilistic clustering while the second is a variation of the mountain clustering using medoids as cluster representatives. Moreover an algorithm based on the idea of noise clustering is developed. The last idea is applied to sequential extraction of regression models and we have the fourth algorithm. We compare these algorithms using numerical examples.**

I. Introduction

Although the methods of crisp and fuzzy c-means clustering have a central role in various clustering algorithms [1], [2], [5], [9], there is a well-known drawback: the number of clusters has to be specified beforehand, and obtained clusters are strongly depends of this number. Hence there are many studies on how to specify this number; one of best-known method is to use a cluster validity measure. In contrast, there are other classes of clustering techniques for which the number of clusters need not be known beforehand. A well-known method is the mountain clustering [11] which uses a sequential generation of *mountain* clusters and subtracting them one by one. A drawback still exists in this method, that is, a high-dimensional data space cannot be handled, since the mountain clustering uses grid points and the number of grid points grows exponentially with the data dimension.

In this paper we consider three different approaches for sequential generation or in other words, sequential extraction of clusters. These approaches are based respectively on existing ideas of possibilistic clustering [7], mountain clustering, and noise clustering [3], [4]. Accordingly three different algorithms are developed. Moreover, a still another algorithm to have clusters of regression models is developed using the idea of noise clusters. We note that a similar idea has been presented elsewhere [4], but the development of methods herein are new.

Numerical examples show how effective and/or efficient the proposed methods are.

II. Possibilistic Clustering and Sequential Extraction of Clusters

To begin with, we review the possibilistic c-means algorithm. Objects for clustering are assumed to be a vector in the p-dimensional Euclidean space: $x_k = (x^1, \dots, x_k^p) \in \boldsymbol{R}^p$, $k = 1, \dots, n$. Cluster centers are $v_i = (v_i^1, \dots, v_i^p)^T$, $i = 1, \dots, c$, where c is the number of clusters. A simplified symbol $V = (v_1, \dots, v_c)$ is used for the whole collection of cluster centers. The membership matrix $U = (u_{ki})$, $(i = 1, \dots, c,\ k = 1, \dots, n)$ is used as usual, where u_{ki} means the degree of belongingness of object x_k to cluster i. We moreover assume D_{ki} to be the squared Euclidean distance:

$$D_{ki} = \|x_k - v_i\|^2$$

Two objective functions are considered: the first has been proposed by Krishnapuram and Keller [7], while the latter has been mentioned by Davé and Krishnapuram [4] (see also [8]).

$$J_2(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ki})^m D_{ki} + \sum_{i=1}^{c} \eta_i \sum_{k=1}^{n} (1 - u_{ki})^m \tag{1}$$

$$J_e(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} \{u_{ki} D_{ki} + \lambda^{-1} u_{ki} (\log u_{ki} - 1)\} \tag{2}$$

We hereafter assume $m = 2$ and $\eta_i = \zeta^{-1}$ $(i = 1, \dots, c\ )$ which are enough for our purpose.

An alternative optimization procedure whereby $J_2(U, V)$ or $J_e(U, V)$ is minimized with respect to U while V is fixed to the last optimal solution and then it is minimized with respect to V while U is fixed to the last optimal solution. This iteration is repeated until convergence. As a result, the iteration of

$$u_{ki} = \frac{1}{1 + \zeta D_{ki}}, \tag{3}$$

$$v_i = \frac{\sum_{k=1}^{n} (u_{ki})^2 x_k}{\sum_{k=1}^{n} (u_{ki})^2}. \tag{4}$$

for $J_2(U, V)$ is used, whereas the iteration is for

$$u_{ki} = \exp\left(-\lambda D_{ki}\right), \tag{5}$$

$$v_i = \frac{\sum_{k=1}^{n} u_{ki} x_k}{\sum_{k=1}^{n} u_{ki}}. \tag{6}$$

when $J_e(U, V)$ is used.

We define two functions related to (5) and (3) respectively.

$$U_e(x_k, y) = \exp(-\lambda D(x_k, y)) \tag{7}$$

$$U_2(x_k, y) = \frac{1}{1 + \zeta D(x_k, y)} \tag{8}$$

and note $U_e(x_k, v_i) = u_{ki}$ for J_e; $U_2(x_k, v_i) = u_{ki}$ for J_2.

Sadaaki Miyamoto and Kenta Arai are with the Department of Risk Engineering, University of Tsukuba, Ibaraki 305-8573, Japan (email: miyamoto@risk.tsukuba.ac.jp).

978-1-4244-3597-5/09/$25.00 ©2009 IEEE

In order to investigate properties of possibilistic clustering, we substitute $U(V) = (U_e(x_k, v_i))_{i=1,\dots,c}$ into $J_e(U,V)$ where v_i is regarded as a variable. We have

$$J_e(U(V),V) = -\lambda \sum_{i=1}^{c} \sum_{k=1}^{n} \exp(-\lambda D(x_k, v_i)).$$

If we put $J'_e(V) = J_e(U(V),V)$ and

$$j_e(y) = -\sum_{k=1}^{n} \exp(-\lambda D(x_k, y)),$$

we have

$$J'_e(V) = J_e(U(V),V) = \lambda \sum_{i=1}^{c} j_e(v_i).$$

Note that this substitution changes the original formulation of possibilistic clustering and hence the optimization of $J'_e(V)$ is not identical with the alternative minimization of $J_e(U,V)$, they are expected to have similar properties (cf. [10]). We hence investigate properties of $J'_e(V)$ as a function of V.

We notice that $J'_e(V)$ is the sum of $j_e(v_i)$. Since no constraint is imposed on v_i, every $j_e(v_i)$ can be minimized independently from other $j_e(v_j)$ $(j \neq i)$. If we assume the minimizing element is unique, then the minimization leads to $\bar{v} = v_1 = \cdots = v_c$ and hence only one cluster center will be obtained as the minimizing element of $J'_e(V)$. Thus, if we want to have multiple clusters from this function, we should search different minimizing solutions of a multi-modal function, which is far more difficult than the minimization of a unimodal function.

This observation leads us to the idea of extracting 'one cluster at a time' which has already been discussed by Davé and Krishnapuram [4]. We also note this idea has not yet fully developed. Hence we discuss their idea in more detail, whereby we develop new algorithms.

We also consider $J_2(U,V)$ in the same way. Let us substitute $U(V) = (U_e(x_k, v_i))_{i=1,\dots,c}$ into $J_2(U,V)$ in which v_i is a variable. We have

$$J'_2(V) = J_2(U(V),V) = \sum_{i=1}^{c} \sum_{k=1}^{n} \frac{D(x_k, v_i)}{1 + \zeta D(x_k, v_i)}.$$

We put

$$j_2(y) = \sum_{k=1}^{n} \frac{D(x_k, y)}{1 + \zeta D(x_k, y)},$$

and it follows that

$$J'_2(V) = \sum_{i=1}^{c} j_2(v_i).$$

We note again that $j_2(v_i)$ can be minimized independently from other $j_2(v_j)$. We thus have a unique solution $\hat{v} = v_1 = \cdots = v_c$, that minimizes $J'_2(V)$.

These observations justify the use of an algorithm to extract 'one cluster at a time.' A general procedure for this is as follows.

SC: A General Procedure for Sequential Clustering.
SC1. Let the initial set of objects be $X^{(0)} = X$ and $k = 0$ Let the function $J(v;k) = j_e(v)$ (or $J(v;k) = j_2(v)$) with the set of objects $X^{(k)}$.
SC2. Search the minimizing element of $J(v;k)$:

$$v^{(k)} = \arg\min_{v} J(v;k)$$

SC3. Extract cluster $G^{(k)}$ that belongs to the center $v^{(k)}$.
SC4. Let $X^{(k+1)} = X^{(k)} - G^{(k)}$. If $X^{(k+1)}$ does not have sufficient elements to extract one more cluster, stop; else $k := k + 1$ and go to step **SC2**.
End of SC.

A notable feature in this procedure is that we do not need to specify the number of clusters beforehand. However, a minimizing method is not specified in this procedure.

The next procedure uses the iterative calculation of u_{ki} and v_i in order to minimize $J = J_e$ or $J = J_2$.

Procedure A.(Sequential algorithm based on possibilistic clustering)
A1. Generate candidate points $y_1, \dots, y_L \in \boldsymbol{R}^p$ and they are initial cluster centers. $X^{(0)} = X$ and $k = 0$.
A2. Repeat calculation of u_{ki} by (5) (resp. 3) and v_i by (6) (resp. 4) until convergence. Converged points are denoted by $z_1, \dots z_\ell$. Find minimizing element

$$\bar{z} = \arg \min_{v=z_1,\dots,z_\ell} J(v;k). \tag{9}$$

A3. Find the cluster $G^{(k)}$ with the center $\bar{z}$. Extract $G^{(k)}$: $X^{(k+1)} = X^{(k)} - G^{(k)}$. If $X^{(k+1)}$ does not have sufficient elements to extract one more cluster, stop; else $k := k + 1$ and go to **A2**.
End A.

III. Mountain Medoid Clustering

We next review the mountain clustering [11]. This method considers the mountain function

$$M(y) = \sum_{k=1}^{n} \exp(-\alpha D(x_k, y)), \quad (\alpha > 0) \tag{10}$$

where $y \in \boldsymbol{R}^p$ is restricted to grid points. Let $y^{(1)}$ be the maximizing point of (10). Then the second mountain function is defined:

$$M^{(2)}(y) = M(y) - M(y^{(1)}) \sum_{k=1}^{n} \exp(-\alpha D(y^{(1)}, y)).$$

and then the calculation is repeated:

$$M^{(\ell)}(y) = M^{(\ell-1)}(y) - M(y^{(\ell-1)}) \sum_{k=1}^{n} \exp(-\alpha D(y^{(\ell-1)}, y)). \tag{11}$$

until there is no significant cluster. The stopping criterion is given by the ratio and a given parameter $\delta > 0$:

$$\frac{M(y^{(1)})}{M(y^{(\ell-1)})} < \delta. \tag{12}$$

We immediately see

Proposition 1: $M(y)$ and $-j_e(y)$ have the identical form by putting $\alpha = \lambda$:

$$M(y) = -j_e(y) \sum_{k=1}^{n} U_e(x_k, y) = \sum_{k=1}^{n} \exp(-\alpha D(x_k, y)).$$

Moreover the clusters are extracted one by one.

A drawback of the mountain clustering is that it uses grid points and hence it is not suited to high-dimensional data. More specifically, the method has the complexity $O(ng^p)$ where g is the number of grid point for each axis.

An immediate variation that seems useful is to replace grid by the given objects. Then we have a kind of *medoid* clustering [6]. We hence call it a method of mountain medoid clustering. Thus this method uses iteration with respect to $\ell = 1, 2, \ldots$:

$$y^{(\ell-1)} = \arg\min_{z \in X} M(z) \tag{13}$$

$$M^{(\ell)}(y) = M^{(\ell-1)}(y) - M(y^{(\ell-1)}) \sum_{k=1}^{n} \exp(-\alpha D(y^{(\ell-1)}, y)). \tag{14}$$

with the same stopping criterion

$$\frac{M(y^{(1)})}{M(y^{(\ell-1)})} < \delta. \tag{15}$$

except that we should check all objects in X.

Another implication from Proposition 1 and the above consideration is that we can take $M(z) = -j_2(z)$ as well as $M(z) = -j_e(z)$. We thus have two methods for mountain clustering and mountain medoid clustering. Note also that mountain medoid clustering has the complexity $O(pn^2)$. Hence the mountain medoid clustering can be applied to high-dimensional data.

IV. Sequential Algorithm Based on Noise Clustering

To propose a third method, we review the method of noise clustering [3], [4]. This method uses one of the next two objective functions (see also [9]):

$$J_{2n}(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ki})^m D_{ki} + \sum_{k=1}^{n} (u_{k0})^m \delta \tag{16}$$

$$J_{en}(U, V) = \sum_{k=1}^{n} \sum_{i=1}^{c} \{u_{ki} D_{ki} + \lambda^{-1} u_{ki} \log u_{ki}\} + \sum_{k=1}^{n} u_{k0} \delta \tag{17}$$

where u_{k0} is the membership to the noise cluster as cluster 0; $\delta > 0$ is a parameter which shows that every object has a constant dissimilarity δ from the noise cluster.

In order to use this function to sequential extraction of clusters, we set $c = 1$ (see [4]), and use algorithm **SC**, where

$$G^{(k)} = \{\ x_k \in X^{(k)}\ :\ u_{k1} \geq u_{k0}\ \}.$$

A. Sequential extraction of regression models

A variation of noise clustering is applied to sequential extraction of regression models. Given data $\{(x_k, y_k)\}_{k=1,\ldots,n}$ where $x_k \in \boldsymbol{R}^p$ is a p-dimensional data for independent variable x and $y_k \in \boldsymbol{R}$ is a datum for dependent variable y. In this case we consider two objective functions

$$J_{2nr}(U, B) = \sum_{k=1}^{n} (u_{ki})^m (y_k - \sum_{j=1}^{p} \beta_j x_k^j + \beta_{p+1})^2 + \sum_{k=1}^{n} (u_{k0})^m \delta \tag{18}$$

$$J_{enr}(U, B) = \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ki} (y_k - \sum_{j=1}^{p} \beta_j x_k^j + \beta_{p+1})^2 + \lambda^{-1} \sum_{k=1}^{n} \sum_{i=1}^{c} u_{ki} \log u_{ki} + \sum_{k=1}^{n} u_{k0} \delta \tag{19}$$

where parameter $B = (\beta_1, \ldots, \beta_{p+1})$ for a regression model

$$y = \sum_{j=1}^{p} \beta_j x^j + \beta_{p+1}$$

should be estimated. We can use the same sequential algorithm *SC* to have more than one regression models when we replace the objective function J_{2n} by J_{2nr} (or J_{en} by J_{enr}). In the next section we contrast this method with the ordinary fuzzy c-regression models which minimize

$$J_{fcrm}(U, \mathcal{B}) = \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ki})^m (y_k - \sum_{j=1}^{p} \beta_j^{(i)} x_k^j + \beta_{p+1}^{(i)})^2 \tag{20}$$

where $\mathcal{B} = (\beta_1^{(1)}, \ldots, \beta_{p+1}^{(1)}, \ldots, \beta_1^{(c)}, \ldots, \beta_{p+1}^{(c)})$.

V. Numerical Examples

Three types of numerical examples are shown. First uses the well-known Iris data. Second uses random data and the third concerns regression models.

A. Iris data

Figures 1–4 show two-dimensional displays of the results of sequentially extracted clusters from those algorithms based on possibilistic clustering (called SPCM in the figure), noise clustering (SNFCM), mountain clustering (MC), and mountain medoid clustering (MMC), respectively. In particular, grid points are shown in Figure 3. The first two methods are based on the objective functions with the entropy terms. We observe that the numbers of clusters are different, and the results are different accordingly, but there is a common feature that the two linearly separated groups in Iris data are successfully separated by all the methods. Note that the three classes are not necessarily separated from the viewpoint of clustering. Thus the results shown here are natural and partly successful.

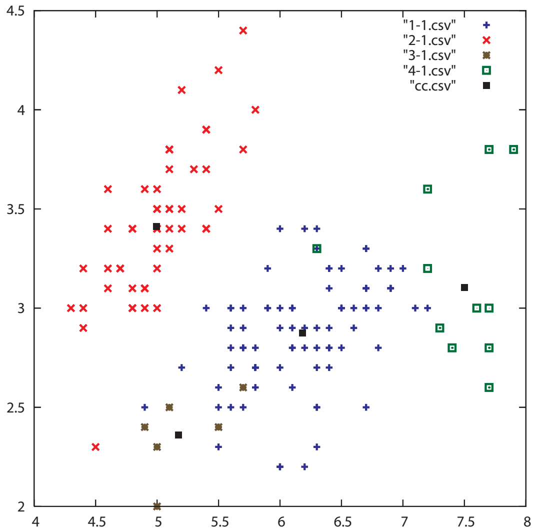

Fig. 1. Iris data(SPCM)$\lambda = 1$

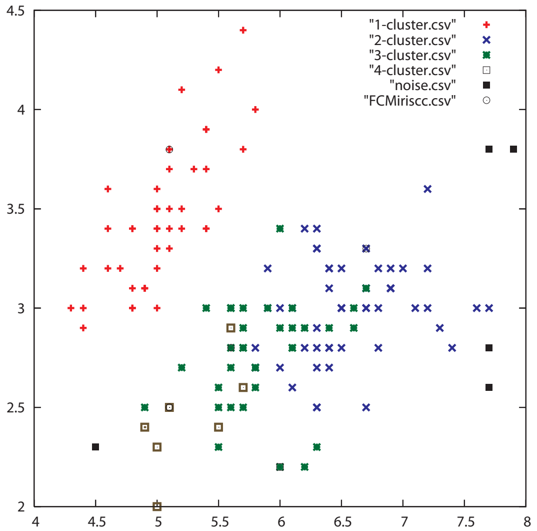

Fig. 2. Iris data(SNFCM)$\lambda = 1$ $\delta^2 = 5$

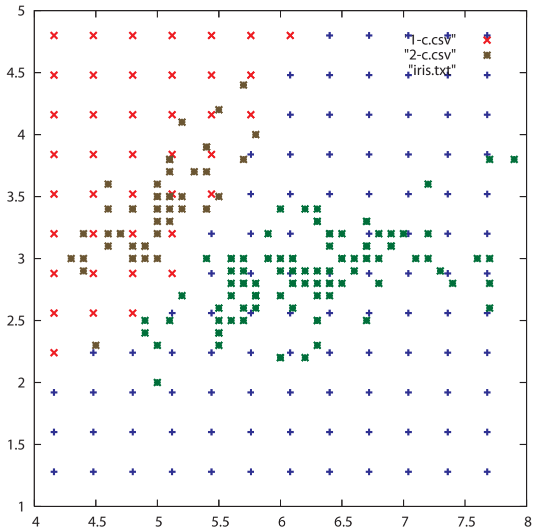

Fig. 3. Iris data(MC)$\alpha = 0.9$

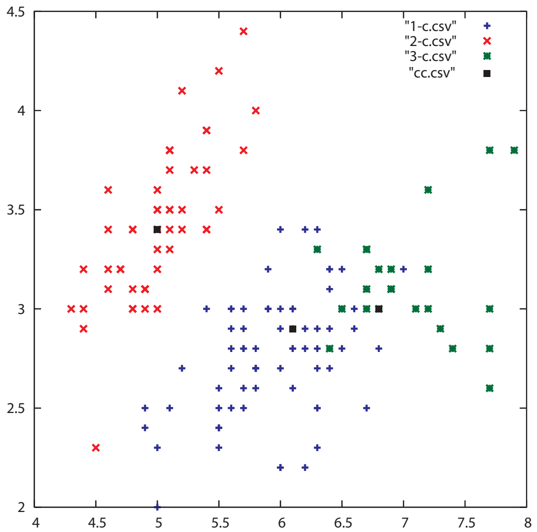

Fig. 4. Iris data(MMC)$\alpha = 0.9$

B. *Comparison of performances between mountain clustering and mountain medoid clustering*

Figure 5 contrasts computation time (sec) by the two method of the ordinary mountain clustering (red and solid curve) and the mountain medoid clustering (blue and dotted curve) with different dimensions shown on horizontal line. The data have been generated randomly, since the objective here is to compare computation between the two methods. As expected, the computation of the ordinary method grows rapidly with the data dimension, while that of the medoid method remains acceptable for the dimensions.

C. *Regression models*

The last example is based on a real data that is unpublished but shows relations between GDP and energy consumption for different countries. The details are omitted here. We call this data set as GDP data set. Figure 6 shows this data a set along with expected three regression models. Figures 7–9 show the sequentially extracted clusters and Figure 10 depicts all the clusters. It seems the sequential algorithm based on noise clustering successfully works for the regression models.

VI. Conclusion

We have studied algorithms of sequential extraction of clusters which connects the idea of possibilistic clustering [7], noise clustering [3] and the mountain clustering [11]. Moreover performances of the mountain medoid method and the ordinary mountain method have been compared using numerical examples. As a result it has been observed that the mountain medoid method is far more efficient than the ordinary mountain method when the dimension of the data space is large. Moreover sequential extraction of regression models using the idea of noise clustering has been proposed.

To summarize, sequential clustering algorithm should be

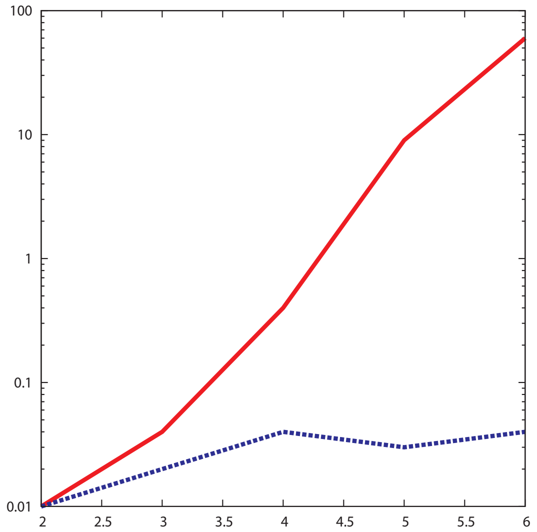

Fig. 5. Comparison of computations between the ordinary mountain clustering (red and solid curve) and mountain medoid clustering (blue and dotted curve), where the vertical axis uses logarithmic scale.

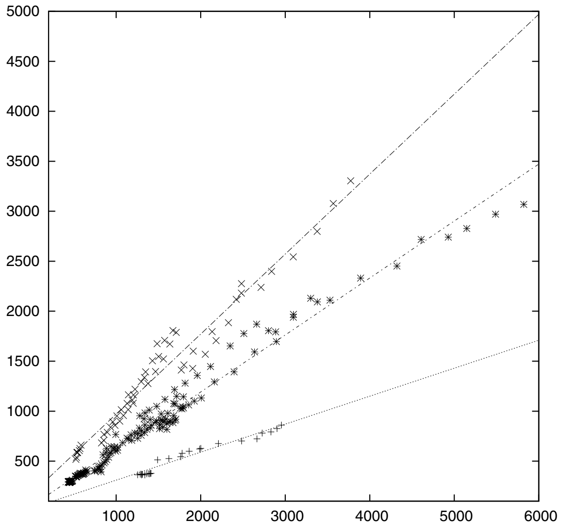

Fig. 6. Three regression models for an unpublished real data set called GDP using the ordinary fuzzy c-regression models (20), where three clusters were assumed.

remarked, as the good property of the automatic determination of the number of clusters in this method should not be overlooked. Moreover, as there are many variations of fuzzy c-means, we have further investigations to be done in both methodological features and applications with regard to the present method.

Acknowledgment

This study has partly been supported by the Grant-in-Aid for Scientific Research, Japan Society for the Promotion of Science, No.19300074.

References

[1] J.C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum, New York, 1981.

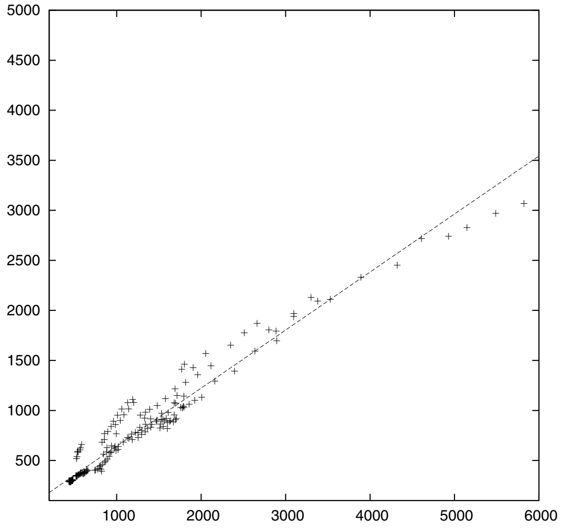

Fig. 7. First cluster from the GDP data set.

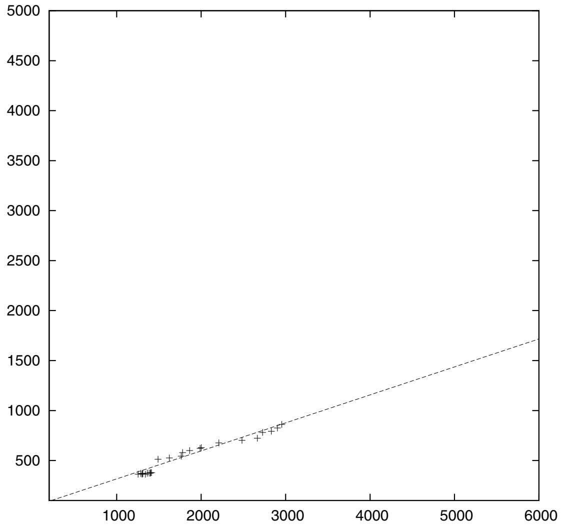

Fig. 8. Second cluster from the GDP data set.

[2] J.C. Bezdek, J. Keller, R. Krishnapuram, N.R. Pal, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*, Kluwer, Boston, 1999.

[3] R.N. Davé, Characterization and detection of noise in clustering, *Pattern Recog. Letters*, Vol.12, pp.657-664, 1991.

[4] R.N. Davé, R. Krishnapuram, Robust clustering methods: a unified view, *IEEE Trans. Fuzzy Syst.*, Vol.5, No.2, pp. 270–293, 1997.

[5] F. Höppner, F. Klawonn, R. Kruse, T. Runkler, *Fuzzy Cluster Analysis*, Wiley, Chichester, 1999.

[6] L. Kaufman, P.J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, Wiley, 1990.

[7] R. Krishnapuram, J.M. Keller, A possibilistic approach to clustering, *IEEE Trans. on Fuzzy Syst.*, Vol.1, No.2, pp. 98–110, 1993.

[8] S. Miyamoto, M. Mukaidono, Fuzzy c-means as a regularization and maximum entropy approach, *Proc. of the 7th International Fuzzy Systems Association World Congress (IFSA'97)*, June 25-30, 1997, Prague, Czech, Vol.II, pp.86–92, 1997.

[9] S. Miyamoto, H. Ichihashi, K. Honda, *Algorithms for Fuzzy Clustering*, Springer, Berlin, 2008.

[10] T.A. Runkler, C. Katz, Fuzzy Clustering by Particle Swarm Opti-

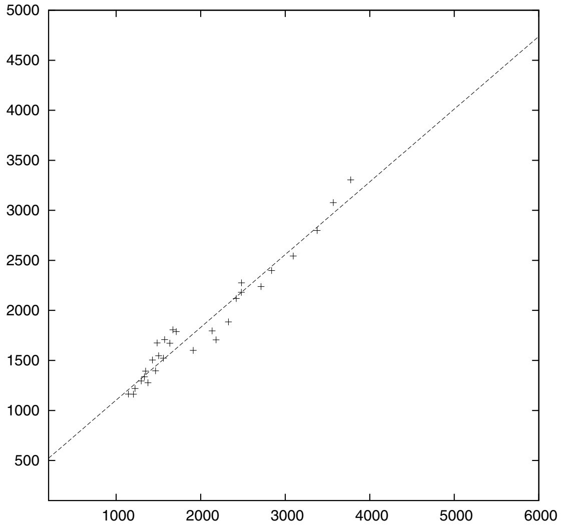

Fig. 9. Third cluster from the GDP data set.

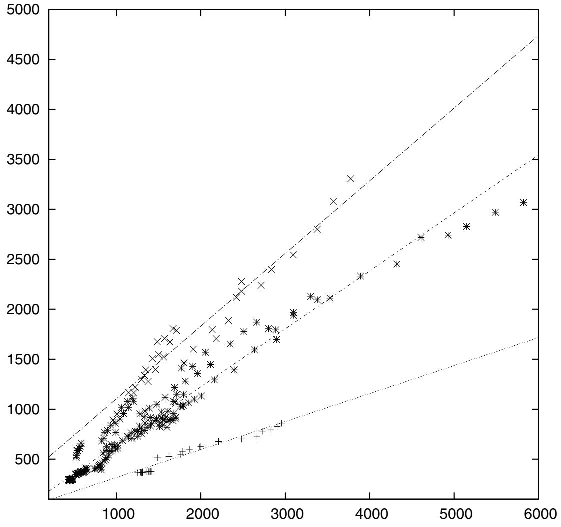

Fig. 10. Overall results of the sequential extraction of regression models using *SC*.

mization, *2006 IEEE International Conference on Fuzzy Systems*, Vancouver, BC, Canada, July 16-21, 2006, pp. 3065–3072.

[11] R.R. Yager, D. Filev, Approximate clustering via the mountain method, *IEEE Trans., on Syst., Man, and Cybern.*, Vol.24, No.8, pp. 1279–1284, 1994.