A Fuzzy Approach for Clustering Gene Expression Time Series Data

Sadiq Hussain

System Administrator
Examination Branch Dibrugarh University

sadiqdu@rediffmail.com

Prof. G.C. Hazarika

Director i/c, Centre for Computer Studies Dibrugarh University

gchazarika@gmail.com

Abstract: *Identifying groups of genes that manifest similar expression patterns is crucial in the analysis of gene expression time series data. Choosing a similarity measure to determine the similarity or distance between profiles is an important task. Time series expression experiments are used to study a wide range of biological systems. More than 80% of all time series expression datasets are short (8 time points or fewer). These datasets present unique challenges. On account of the large number of genes profiled (often tens of thousands) and the small number of time points many patterns are expected to arise at random. Most clustering algorithms are unable to distinguish between real and random patterns. However, the shortness of gene expression time-series data limits the use of conventional statistical models and techniques for time-series analysis. To address this problem, this paper proposes the Fuzzy clustering algorithm based on short time-series, which is able to cluster profiles based on the similarity of their relative change of expression level and the corresponding temporal in- formation. One of the major advantages of fuzzy clustering is that genes can belong to more than one group, revealing distinctive features of each gene's function and regulation.*

Keywords: Fuzzy Clustering, Short time series, Gene expression

1. INTRODUCTION

Microarrays revolutionise the traditional way of one gene per experiment in molecular biology (Brown and Botstein, 1999). With microarray experiments it is possible to measure simultaneously and over time the activity levels of thousands of genes. An appropriate clustering of gene expression data can lead to meaningful classification of diseases, identification of co-expressed function- ally related genes, logical descriptions of gene regulation, etc. Time course measurements are becoming a common type of experiment in the use of microrarrays. The particularity of time-series, which has to be considered in the clustering analysis, is the temporal information: the measurements ordered in time and sampled at specific intervals. An appropriate similarity measure for gene expression time-series should be able to identify similar shapes which are formed by the relative change of expressions and the temporal information.

Time series gene expression experiments are an increasingly popular method for studying a wide range of biological processes. Examples include response to temperature changes and other stress conditions [4], immune response [5], developmental studies [1], and various systems in the cell [8]. While there have been time series experiments with as many as 80 time points [1], almost all time series are much shorter. There are a number of reasons why short time series datasets are so common. Time series experiments require multiple arrays (and in many cases each point is repeated at least once) making them very expensive. While microarray

DOI : 10.5121/ijcsit.2011.3415

technology have greatly improved over the last five years, its cost is still high at around $300-1000 per microarray which is a limiting factor for many researchers. Even if prices go down short time series experiments would remain prevalent since in many studies it is prohibitive to obtain large quantities of biological material. As an example consider a clinical study in which blood needs to be drawn from patients at various points in time. Due to the large number of genes that are being profiled, most papers presenting short time series datasets use one of several clustering methods to analyze their data. Hierarchical clustering [5] along with other standard clustering methods (such as k-means and self-organizing maps [9]) are often used for this task. While these clustering algorithms yielded many biological insights, they are not designed for time series data. Specically, all these methods assume that data at each time point is collected independently of each other, ignoring the sequential nature of time series data. More recently, a number of clustering algorithms speciacally designed for time series expression data were suggested. These algorithms include clustering based on the dynamics of the expression patterns [6], clustering using the continuous representation of the profile [2], and clustering using a Hidden Markov Model [7]. While these algorithms work well for relatively long time series dataset (10 points or more) they are not appropriate for shorter time series.

2. Related work

Recently, several papers have focused on modeling and analyzing the temporal aspects of gene expression data. In Holter et al [13] a time translational matrix is used to model the temporal relationships between different modes of the Singular Value Decomposition (SVD). Unlike our work, this method focuses on the SVD modes and not on specific genes. In addition, only relationships between time points that are sampled at the lowest common frequencies can be studied. Thus, not all available expression data can be used. In Zhao et al [17] a statistical model is fit to all genes in order to find those that are cell cycle regulated. This method uses a custom tailored model, relying on the periodicity of the specific dataset analyzed, and is thus less general than our approach. Several papers have used simple interpolation techniques to estimate missing values for gene expression data. Aach et al [10] use linear interpolation to estimate gene expression levels for unobserved time-points. D'haeseleer [12] use spline interpolation on individual genes to interpolate missing time-points. In Troyanskaya et al [16] several techniques for missing value estimations were explored. However, none of the suggested techniques take into account the actual times the points correspond to, and thus time series data is treated in the same way as static data. As a consequence, their techniques cannot estimate values for time-points between those measured in the original experiments. There is a considerable statistical literature that deals with the problem of analyzing non-uniformly sampled data. These models, known as mixed-effect models [11] use spline estimation methods to construct a common class profile for their input data. Recently, James and Hastie [14] presented a reduced rank mixed effects model that was used for classifying medical time-series data. In this paper we extend these methods to gene expression data. Unlike the above papers, we focus on the gene specific aspects rather than the common class profile. In addition, we present a method that is able to deal with cases in which class membership is not given. Another difference between this work and [14] is that we do not use a reduced rank approach, since gene expression datasets contain information about thousands of genes. Many clustering algorithms have been suggested for gene expression analysis (see [15]). However, as far as we are aware, all these algorithms treat their input as a vector of data points, and do not take into account the actual times at which these points were sampled.

3. ALGORITHM AND IMPLEMENTATION

This section presents a measure of similarity for microarray time-series data. The proposed similarity measure is driven by the concept of similarity and the particular characteristics of the time-series generated with microarray experiments. First, there is no clear definition of what "similar" time-series are in a biological context. However, it is generally understood that similar expression profiles correspond to similar shapes of expression. Therefore, it is a common practice to use lines between time points rather than isolated points for aiding a visual comparison. Second, these series have two main properties given by the nature of the experiments generating them: they are short and usually unevenly sampled. When the time-series are short, traditional statistical analyses are not always suitable. For example, in the case of an autoregressive model [6], the order of the model is very restricted by the low number of time points available in gene expression time-series. In [11] the authors identified that conventional techniques for time-series analysis, such as Fourier analysis or autoregressive or moving-average modelling are not suitable for the small number of data points as in most gene expression time-series data. As an alternative,

the authors proposed to model the time-series with linear splines. The problem of short time-series has being identified in other fields and has been treated with a particular focus on shape comparisons [13], that is, using the idea of up-down patterns. The objective here is to define a similarity measure that can capture the temporal information to evaluate the similarity of temporal gene expression profiles. We approach the problem by considering the time-series as piecewise linear functions and measuring the difference of slopes between them. In [14], the expression level at each time point and the slopes between time points are included in the comparison of profiles. However, the slopes were calculated based on a reduced time interval of one, not taking into account the variable time intervals. By measuring the difference of the true slopes, we are able to include in a meaningful way the length of sampling intervals, while considering the shape (i.e. up-down patterns) of the series. The length of sampling interval can be understood as a weight; the farther apart in time the expressions are, the less weight they have in the comparison. Considering a gene expression profile x = [x_1, x_2, . . . , x_{nt}], where nt is the number of sampling time points, the linear function x(t) between two successive time points t_k and $t_{(k+1)}$ can be defined as $x(t) = \beta_k t + \delta_k$, where $t_k \leq t \leq t_{(k+1)}$, and $\beta_k = (x_{(k+1)} - x_k)/(t_{(k+1)} - t_k)$ and $\delta_k = (t_{(k+1)}x_k - t_k x_{(k+1)})/(t_{(k+1)} - t_k)$ The proposed STS distance corresponds to the square root of the sum of the squared differences of the slopes obtained by considering time-series as linear functions between measurements. The STS distance between two time-series x and v is thus defines as

$$d^2_{\mathrm{STS}}(x, v) = \sum_{k=1}^{n_t} \left(\frac{v_{(k+1)} - v_k}{t_{(k+1)} - t_k} - \frac{x_{(k+1)} - x_k}{t_{(k+1)} - t_k} \right)^2 . \qquad (1)$$

The Fuzzy Clustering Algorithm

The wide variety of clustering algorithms available from various disciplines are distinguished by the way in which they measure distances between objects and the way they group the objects based upon the measured distances [15]. In the previous section we already discussed the way

in which we desire the "distance" between objects to be measured; hence, in this section, we focus

on grouping the objects based upon the measured distance. For this purpose we select the fuzzy clustering scheme as a template for our development, since fuzzy sets are a more realistic approach to address the concept of similarity than classical sets. A classical set has a crisp or hard boundary where the constituting elements have only two possible values of membership. In contrast, a fuzzy set has fuzzy boundaries where each element is given a degree of membership providing information about the influence of a given gene for the overall characteristics of the cluster. In addition, a fuzzy approach inherently accounts for noise in the data because it extracts trends, not precise values. Fuzzy clustering is a partitioning-optimisation technique [16–18]. The objective function that measures the desirability of partitions is described by,

$$J(x, v, u) = \sum_{i=1}^{n_c} \sum_{j=1}^{n_g} u_{ij}^{w} \, d^2(x_j, v_i) \qquad (2)$$

where nc is the number of clusters, ng is the number of vectors to cluster, uij is the value of the membership degree of the vector xj to the cluster i, d2(xj , vi) is the squared distance between vector xj and prototype vi, and w is a parameter (usually set between 1.25 and 2), which determines the degree of overlap of fuzzy clusters. The minimisation of the fuzzy objective function is a nonlinear optimization problem that can be solved using various methods. The most common method is the Picard Iteration through the first-order conditions for stationary points of the function. Figure 2 illustrates the iterative procedure of the fuzzy c-means algorithm, the most well known fuzzy clustering algorithm. Considering other fuzzy extensions, the convergence is independent of the change in the distance function if the distances are all positive and the prototypes are calculated accordingly to the minimisation of the objective function. A full review of the minimisation and convergence of the FCM objective function can be found in [19].

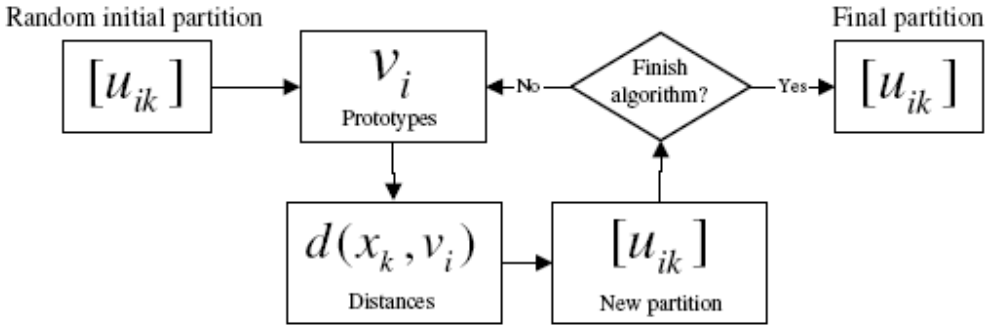


Fig. 2. Diagram of the iterative procedures for the FCM clustering algorithm. Considering the partition of a set X = [x1, x2, . . . , xng], into 2 <= nc < ng clusters, the fuzzy clustering partition is represented by a matrix U = [uik], whose elements are the values of the membership degree of the object xk to the cluster i, ui(xk) = uik.In order to integrate the STS distance into

the conventional fuzzy clustering scheme, it is necessary to obtain the value of the prototype vi that minimizes (2), when (1) is used as the distance. Substituting (1) into (2) we obtain

$$J(x,v,u)=\sum_{i=1}^{n_c}\sum_{j=1}^{n_g}u_{ij}^{w}\sum_{k=1}^{n_t}\left(\frac{v_{i(k+1)}-v_{ik}}{t_{(k+1)}-t_k}-\frac{x_{j(k+1)}-x_{jk}}{t_{(k+1)}-t_k}\right)^2. \tag{3}$$

The partial derivative of (3) with respect to v_{ik} is:

$$\frac{\partial J(x,v,u)}{\partial v_{ik}}=\sum_{j=1}^{g}2u_{ij}^{w}\frac{\left(a_k v_{i(k-1)}+b_k v_{ik}+c_k v_{i(k+1)}\right)}{\left(t_k-t_{(k+1)}\right)^2\left(t_k-t_{(k-1)}\right)^2}+\sum_{j=1}^{g}2u_{ij}^{w}\frac{\left(d_k x_{j(k-1)}+e_k x_{jk}+f_k x_{j(k+1)}\right)}{\left(t_k-t_{(k+1)}\right)^2\left(t_k-t_{(k-1)}\right)^2} \tag{4}$$

where

$$a_k=-(t_{(k+1)}-t_k)^2 \quad b_k=-(a_k+c_k) \quad c_k=-(t_k-t_{(k-1)})^2$$
$$d_k=(t_{(k+1)}-t_k)^2 \quad e_k=-(d_k+f_k) \quad f_k=(t_k-t_{(k-1)})^2.$$

Setting (4) equal to zero and solving for v_{ik} we have

$$a_k v_{i(k-1)}+b_k v_{ik}+c_k v_{i(k+1)}=m_{ik} \tag{5}$$

where

$$m_{ik}=-\frac{\sum_{j=1}^{n_g}u_{ij}^{w}\left(d_k x_{j(k-1)}+e_k x_{jk}+f_k x_{j(k+1)}\right)}{\sum_{j=1}^{n_g}u_{ij}^{w}}.$$

Equation (5) yields an underdetermined system of equations. We know the relations of the prototype values among the time points, but not the absolute value at each time point. That is, we know the slope but not the absolute level. By adding two known fixed time points we can solve the underdetermined system of equations for any nt. If we add the same two time points to all the

time-series the similarity is not altered. If the fixed values are zero, a general solution is easier to obtain. The length of the sampling interval between the first real time point and the last fixed time point acts as a weight to the first real time point. Additional time points should be t1 = −1 and t2 = 0, and the original time points should be scaled down by substraction to start as t3 = 1. This avoids altering v with the added fixed time points, since with this configuration, the values

of a, c and f for the extra time points equal one and do not affect the products. The prototypes can be calculated as shown in the following equation,

$$v(i,n) = \sum_{r=2}^{n-3} [m_{ir} \prod_{q=1}^{r-1} c_q \left(\prod_{q=r+1}^{n-1} a_q + \prod_{q=r+1}^{n-1} c_q + \sum_{p=r+3}^{n} \prod_{j=p-1}^{n-1} c_j \prod_{j=r+1}^{p-2} a_j \right) / \prod_{q=2}^{n-1} c_q] + [m_{i(n-1)} \prod_{q=1}^{n-2} c_q + m_{i(n-2)} \prod_{q=1}^{n-3} c_q (a_{(n-1)} + c_{(n-1)})] / \prod_{q=2}^{n-1} c_q \quad (6)$$

where $1 \leq i \leq n_c$, $3 \leq n \leq n_t + 2$ (since $v(i,1) = 0$ and $v(i,2) = 0$), $m_{i1} = 0$ and $c_1 = 1$.

The change of the distance function has no effect in the optimisation of (2) with respect to the membership degree, therefore, uij can be calculated as in the FCM algorithm,

$$u_{ij} = \frac{1}{\sum_{q=1}^{n_c} (d_{\mathrm{STS}}(x_i, v_j)/d_{\mathrm{STS}}(x_i, v_q))^{1/(w-2)}}. \quad (7)$$

The Algorithm

STEP 1: Initialization

n_g	:	number of genes
n_t	:	number of time points
X	:	gene expression matrix (GEM) $[n_g \times n_t]$
n_c	:	number of clusters
w	:	fuzziness parameter
α	:	threshold for membership
ϵ	:	termination tolerance
t	:	time points

STEP 2: Addition of two fixed time points and fuzzification

STEP 3: Initialization of the partition matrix

Initialize the partition matrix randomly,

$U^{(0)}$ $[n_c \times n_g]$.

STEP 4: Repeat for l = 1, 2, ...

4.1 Compute the cluster prototypes:

4.2 Compute the distances using Equation (1),

$1 \leq i \leq n_c$ and $1 \leq j \leq n_g$.

4.3 Update the partition matrix:

if $d_{\mathrm{STS}ij} > 0$ for $1 \leq i \leq n_c$, $1 \leq j \leq n_g$, use Equation (7);

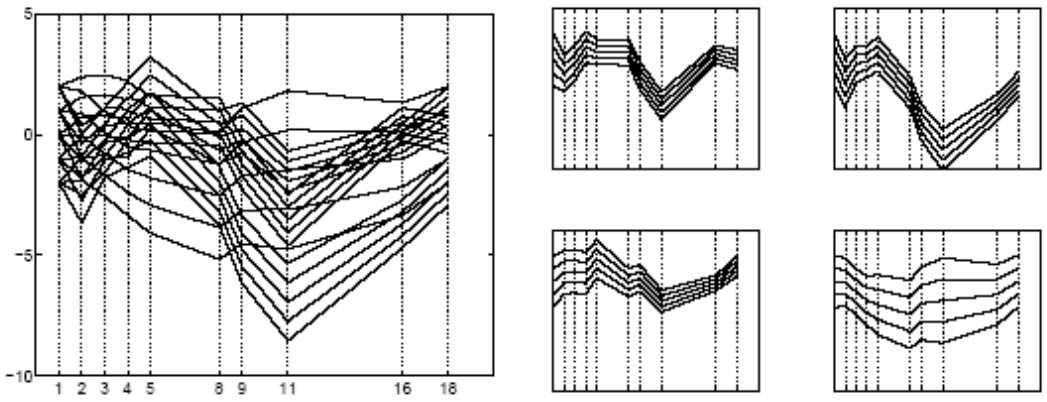
otherwise $u_{ij}^{(l)} = 0$ if $d_{\mathrm{STS}ij} > 0$, and $u_{ij}^{(l)} \in [0,1]$ with $\sum_{i=1}^{n_c} u_{ij}^{(l)} = 1$.

Until $\|U^{(l)} - U^{(l-1)}\| < \epsilon$.

Step 5: Use Z-test to find the fitness of clusters

3.3 Illustrative examples

Simulated data sets are used to illustrate the FSTS clustering algorithm.



(a) Unevenly resampled simulated data

(b) Constituting clusters

Only the FSTS algorithm is able to identify the constituting clusters successfully. Our algorithm is able to identify the four clusters successfully (FCM 41 out of 50 runs, FSTS 50 out of 50 runs, KM 23 out of 50 runs and HC 50 out of 50 runs). The clustering parameters are w = 1.6 and α = 0.4 for the two fuzzy algorithms.

4. Conclusion

Clustering algorithms have been developed for various applications and within a range of disciplines. In order to choose the most suitable algorithm for a particular application, the type of experiment and the specific purposes of the research have to be considered. The concept of similarity is at the core of any clustering algorithm and terms such as co-expression and

"similar profiles" have to be well defined within the biological context. In this paper we have introduced a metric in which the similarity is based on the rate of change of expression levels across time, which is an intuitive biological idea of similar behavior across time.There are a number of interesting extensions that could be made to our work. Experimental biologists often determine the sampling rate for a time-series experiment based on knowledge about how quickly gene expression values change. These assessments often make little use of information that may be gleaned from previous expression experiments. Our algorithm could be used to find the "right" sampling rate for time-series experiments, which could lead to substantial time/cost savings or improvements in biological results. Another way of extending this work is to develop a clustering algorithm that uses our method in order to group genes that show similar kinetic changes between datasets. Another open problem is developing a principled method for determining the significance of the alignment error in order to automatically detect genes whose temporal behavior is altered between experiments.

References

[1]M.N. Arbeitman, E.E. Furlong, F.J. Imam, E. Johnson, B.H. Null, B.S. Baker, M.A. Krasnow, M.P. Scott, R.W. Davis, and K.P. White. Gene expression during the life cycle of drosophila

melanogaster. Science, 298:2270.75, 2002.

[2]Z. Bar-Joseph, G. Gerber, T.S. Jaakkola, D.K. Gifford, and I. Simon. Continuous representations of time series gene expression data. Journal of Computational Biology, 3-4:341.356, 2003.

[3]M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. PNAS, 95:14863.14868, 1998.

[4]A.P. Gasch, P.T. Spellman, C.M. Kao, O. Carmel-Harel, M.B. Eisen, G. Storz, D. Botstein, and P.O. Brown. Genomic expression programs in the response of yeast cells to environmental changes. Mol. Biol. Cell, 11(12):4241.4257, 2000.

[5]K. Guillemin, N. Salama, L. Tompkins, and S. Falkow. Cag pathogenicity island-specific responses of gastric epithelial cells to Helicobacter pylori infection. Proc. Natl Acad. Sci USA, 99:15136.15141, 2002.

[6]M.F. Ramoni, P. Sebastiani, and I.S. Kohane. Cluster analysis of gene expression dynamics. PNAS, 99(14):9121.6, 2002.

[7]A. Schliep, A. Schonhuth, and C. Steinhoff. Using hidden markov models to analyze gene expression time course data. Bioinformatics, 19:i264.i272, 2003.

[8]K.F. Storch, O. Lipan, I. Leykin, N. Viswanathan, F.C. Davis, W.H. Wong, and C.J. Weitz. Extensive and divergent circadian gene expression in liver and heart. Nature, 418:78.83, 2002. [9]P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E.S. Lander, and T.R. Golub. Interpreting patterns of gene expression with self organizing maps: Methods and applications to hematopoietic differentiation. PNAS, 96:2907. 2912, 1999.

[10] J. Aach and G. M. Church. Aligning gene expression time series with time warping algorithms. Bioinformatics, 17:495–508, 2001.

[11] B. Brumback and J. Rice. Smoothing spline models for the analysis of nested and crossed samples of curves. Am. Statist. Assoc., 93:961–976, 1998.

[12] P. D'haeseleer, X. Wen, S. Fuhrman, and R. Somogyi. Linear modeling of mrna expression levels during cns development and injury. In PSB99, 1999.

[13] N. S. Holter, A. Maritan, and et al. Dynamic modeling of gene expression data. PNAS, 98:1693–1698, 2001.

[14] G. James and T. Hastie. Functional linear discriminant analysis for irregularly sampled curves. Journal of the Royal Statistical Society, to appear, 2001.

[15] Sharan R. and Shamir R. Algorithmic approaches to clustering gene expression data. Current Topics in Computational Biology, To appear.

[16] O. Troyanskaya, M. Cantor, and et al. Missing value estimation methods for dna microarrays. Bioinformatics, 17:520–525, 2001.

[17] L. P. Zhao, R. Prentice, and L. Breeden. Statistical modeling of large microarray data sets to identify stimulus-response profiles. PNAS, 98:5631–5636, 2001.

[18] M. F. Ramoni, P. Sebastiani, I. S. Kohane, Cluster analysis of gene expression dynamics, PNAS 99 (14) (2002) 9121–9126.

[19] M. J. L. de Hoon, S. Imoto, S. Miyano, Statistical analysis of a small set of time-orderd gene expression data using linear splines, Bioinformatics 18 (11) (2002) 1477–1485.

[20] L. Todorovski, B. Cestnik, M. Kline, Qualitative clustering of short time-series: A case study of firms reputation data, in: Conference on Data Mining and Warehouses (SIKDD 2002), Ljubljana, Slovenia, 2002.

[21] X. Wen, S. Fuhrman, G. S. Michaels, D. B. Carr, S. Smith, J. L. Barker, R. Somogyi, Large-scale temporal gene expression mapping of central nervous system development, Proc. Natl Acad. Sci. USA 95 (1998) 334–339.

[22] B. Everitt, Cluster Analysis, Heinemann Educational Books, London, England., 1974.

[23] J. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1981.

[24] F. H¨oppner, F. Klawonn, R. Kruse, T. Runkler, Fuzzy Cluster Analysis, John Wiley & Sons, Chichester, England., 1999.

[25] O. Wolkenhauer, Data Engineering, John Wiley & Sons, New York, 2001.

[26] J. Bezdek, A convergence theorem for the fuzzy ISODATA clustering algorithms, IEEE Trans. Pattern Anal. Machine Intell. 2 (1) (1980) 1–8.

[27] D. Demb´el´e, P. Kastner, Fuzzy C-means method for clustering micoarray data, Bioinformatics 19 (8) (2003) 973–980.

[28] L. Heyer, S. Kruglyak, S. Yooseph, Exploring expression data: Identification and analysis of coexpressed genes., Genome Research 9 (1999) 1106–1115.

[29] K. Y. Yeung, D. R. Haynor, W. L. Ruzzo, Validating clustering for gene expression data, Bioinformatics 17 (4) (2001) 309–318.

[30] J. Dunn, Well separated clusters and optimal fuzzy partitions, J. of Cybernetics 4 (1974) 95–104.

[31] M. K. Kerr, G. A. Churchill, Bootstraping cluster analysis: Assessing the reliability of conclussions from microarray experiments, PNAS 98 (16) (2001) 8961–8965.

[32] X. L. Xie, G. Beni, A validity measure for fuzzy clustering, TPAMI 13 (8) (1991) 841–847.

@@@@@@@@@@@@@@@