Scientific Journal of Riga Technical University
Computer Science. Information Technology and Management Science
2011
Volume 49

The Evolution of Biclustering Algorithms

Oleg Uzhga-Rebrov, *Rezekne Higher Education Institution*, Galina Kuleshova, *Riga Technical University*

***Abstract* – Biclustering methods have been initially developed for solving tasks of finding local correlations between expressions of gene subsets in the subsets of conditions. Later on they started to be employed in target marketing for revealing preferences of subsets of customers/buyers over the subsets of products/services. It can be stated with confidence that in the future these methods will find a wide application in other research areas for mining knowledge when initial data are of specific character. This paper provides a short description and analysis of the four well-known biclustering methods in the order of their evolution.**

***Keywords* – biclustering, δ-clustering, δ-p-clustering, OP-clustering, mean squared residue, mean residue, prefix tree, OPC-tree**

I. Introduction

There are many methods for biclustering objects in the set of values of classification attributes. All the existing techniques can be in essence divided into two large groups: (1) methods based on the measure of similarity between objects and (2) methods based on the evaluation of the degree of correlation between attribute values.

In recent years, gene microarray technologies have been developing intensively. While processing a single microarray, evaluations of thousands of gene expressions can be obtained. When such microarrays are obtained under different conditions, initial data are commonly represented as a matrix whose rows are objects (genes) but each column is a set of evaluations of these genes under a single experimental condition. The number at the intersection of the i-th row and j-th column represents expression (activity level) of the i-th gene under the j-th condition.

Researchers are interested in finding such subsets of objects (genes) that exhibit a correlated behaviour under specific subsets of conditions. Subsets of that kind are called biclusters. In fact, the task of discovering biclusters cannot be solved with conventional clustering methods just due to the global character of those methods. On the other hand, classification of gene biclusters is of great importance for understanding cellular processes, disease mechanisms and results of using medicaments. Due to that, different methods have been developed that enable determining biclusters in the set of initial data. There are probabilistic methods, neural network based methods etc. A good review of the fundamental biclustering methods and their classification according to different features can be found in [4].

This work briefly examines four known biclustering methods in the order of their evolution. The essential feature of all these methods is that they use indicators of deviation degrees of these biclusters from ideal models.

II. Methods Using the Concept of Bicluster Residue as a Measure of its Quality

The method proposed in [1] was the first study on the use of the concept of the mean squared residue for discovering biclusters in the initial data set. The authors suppose that the data set is passed through the logarithm transformation. Thus a transition from the multiplicative model of biclusters to the additive one is accomplished. The multiplicative model assumes that the values in any row of the bicluster can be obtained by multiplying values in other row by a constant. The additive model, in contrast, assumes that the values in any row of the bicluster can be derived by adding a constant to the values in other row.

Let us introduce some formal denotations and definitions [1]. Initial data are represented as a matrix (O, A), where O – a set of objects (strings), A – a set of conditions (columns). (I, J) – is a submatrix of matrix (O, A) with dimension IxJ, where I, I $\subseteq$ O, - number of rows, J, J $\subseteq$ A – number of columns of the submatrix.

The residue of element a_{ij} in the submatrix is determined as follows:

$$a_{ij} - a_{iJ} - a_{Ij} + a_{IJ}, \qquad (1)$$

where a_{iJ} - mean value of attribute A in the i-th row of the submatrix; a_{Ij} - mean value of attribute A in the j-th column of the submatrix; a_{IJ} - mean value of all values of attribute A in the submatrix.

For any submatrix (I, J) its mean squared residue score is calculated as

$$H(I,J) = \frac{1}{|I||J|}\sum_{i\in I}\sum_{j\in J}\left(a_{ij} - a_{iJ} - a_{Ij} + a_{IJ}\right)^2, \qquad (2)$$

where $a_{iJ} = \frac{1}{|J|}\sum_{j\in J} a_{ij}$ - the mean of the i-th row in the submatrix;

$a_{Ij} = \frac{1}{|I|}\sum_{i\in I} a_{ij}$ - the mean of the j-th column in the submatrix;

$a_{Ij} = \frac{1}{|I||J|}\sum_{i\in I}\sum_{j\in J} a_{ij} = \frac{1}{|I|}\sum_{i\in I} a_{iJ} = \frac{1}{|J|}\sum_{j\in J} a_{Ij}$ - the mean of the submatrix.

A submatrix (I, J) is called a ***δ-bicluster*** if $H(I,J) \le \delta$ for a certain a priori set value $\delta \ge 0$.

In [1] the authors propose two types of biclustering algorithm. When the first type is used, the mean squared residue score is calculated for the initial matrix, H(O, A). If H(O, A) > δ, then for all rows of the matrix these values are calculated:

$$d(i) = \frac{1}{|J|}\sum_{j\in J}(a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2$$

and for all columns of the matrix there are calculated values

$$d(j) = \frac{1}{|I|}\sum_{i\in I}(a_{ij} - a_{iJ} - a_{Ij} + a_{IJ})^2.$$

A row or a column with the largest value of d(.) is deleted and the process is repeated until a δ-bicluster is found that meets condition H(I, J) ≤ δ.

Another version of the algorithm first assumes a multiple deletion of the rows for which $\frac{1}{|J|}\sum_{j\in J}\left(a_{ij} - a_{iJ} - a_{Ij} + a_{IJ}\right)^2 > \alpha H(I,J),$ and the columns for which $\frac{1}{|I|}\sum_{i\in I}\left(a_{ij} - a_{iJ} - a_{Ij} + a_{IJ}\right)^2 > \alpha H(I,J),$ where α > 1 is a pre- specified coefficient.

After the first δ-bicluster is found in the initial data matrix, the corresponding values in the matrix are replaced by random ones; searching for the second δ-bicluster is then performed and so on. Such a masking of the discovered δ-biclusters by random values is an essential drawback of the considered technique.

The biclustering algorithm proposed in [6] aims at avoiding the shortcomings of the above technique. Let us introduce some denotations and definitions.

A ***δ-cluster*** is defined as an IxJ submatrix of the initial data matrix that satisfies these conditions:

- $\frac{|J_i^{'}|}{|J|} < \alpha,$ where $|J_i^{'}|$ - number of specified attribute values in the i-th row of the submatrix;

$|J|$ - overall number of attribute values in the submatrix.

- $\frac{|I_j^{'}|}{|I|} < \alpha,$ where $|I_j^{'}|$ - number of specified attribute values in the j-th column of the submatrix;

$|I_j|$ - overall number of attributes in the submatrix;

α - a priori assigned value .

Note that these conditions are necessary but not sufficient for defining a δ-cluster. Another necessary condition will be formulated further.

The ***volume of δ-cluster***, VV_{IJ} is defined as the number of its completed cells.

For the given δ-cluster ***the base of the i-th object*** is determined as the mean of all specified attribute values in its i-th row

$$a_{iJ} = \frac{1}{|J_i^{'}|}\sum_{j\in J^{'}} a_{ij}.$$

Similarly, ***the base of the j-th condition*** is the mean of all specified attribute values in its j-th column

$$a_{Ij} = \frac{1}{|I_j^{'}|}\sum_{i\in I^{'}} a_{ij}.$$

The base of a δ-cluster is the mean of all its specified attribute values

$$a_{IJ} = \frac{1}{V_{IJ}}\sum_{i\in I}\sum_{i\in J} a_{ij} = \frac{1}{V_{IJ}}\sum_{i\in I} a_{iJ} = \frac{1}{V_{IJ}}\sum_{j\in J} a_{Ij},$$

where V_{IJ} - volume of δ-cluster.

The residue of cell (i, j) in a δ-cluster is determined as

$r_{ij} = a_{ij} - a_{iJ} - a_{Ji} + a_{Ij}$, if the value a_{ij} is specified;

= 0, otherwise (3)

The residue of δ-cluster (I, J) is calculated as

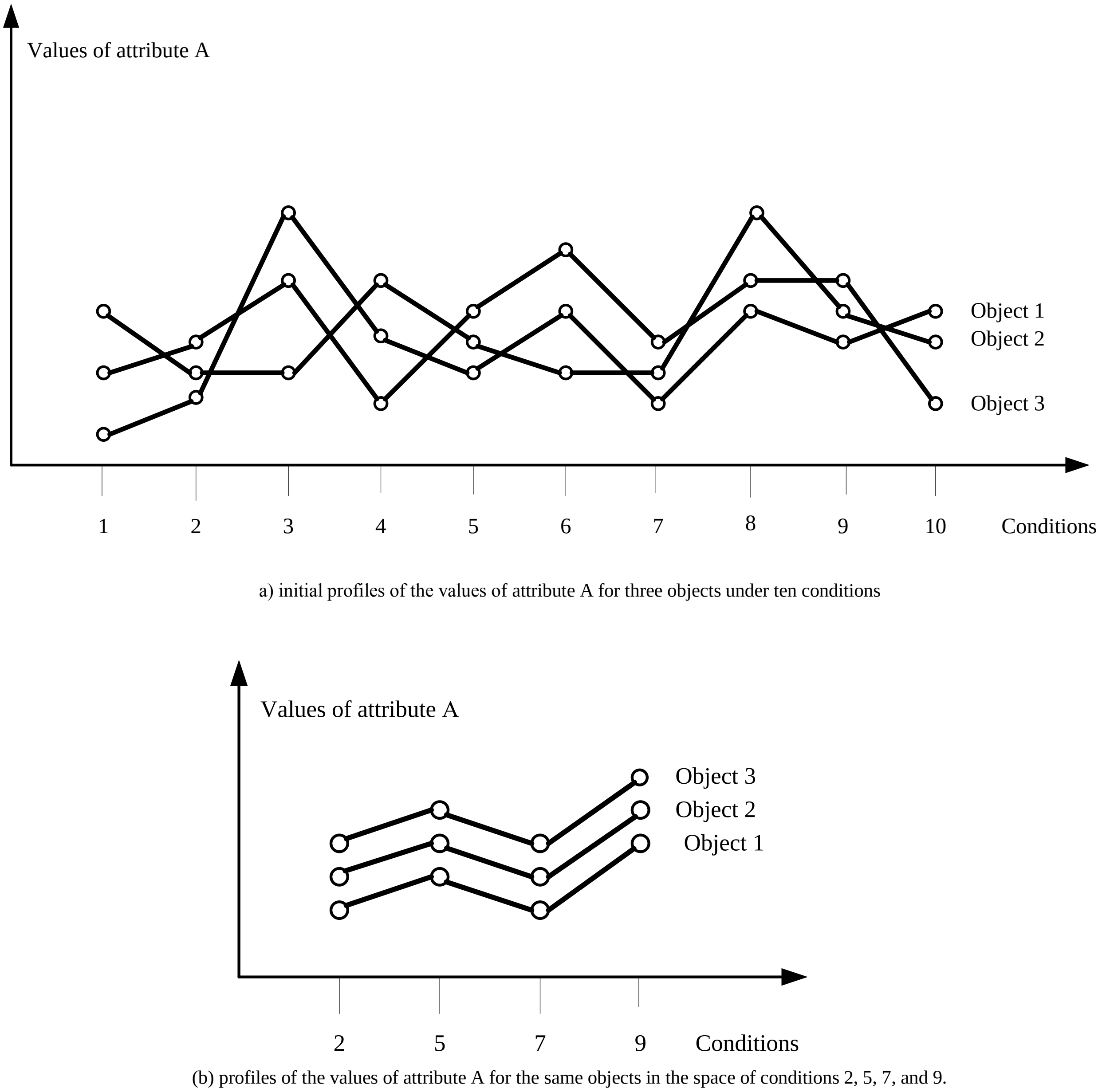
$$R_{IJ} = \frac{1}{V_{IJ}}\sum_{i\in I}\sum_{j\in J}|r_{ij}|. \quad (4)$$

Using the concept of residue R_{IJ}, a ***δ-cluster (I, J)*** can be defined as a submatrix of the initial data (O, A), for which R_{IJ} < δ, where δ - a priori assigned threshold value of the residue. This is the second necessary condition for determining a δ-cluster.

To discover δ-clusters in the matrix of initial data, in [6] the authors propose the FLOC algorithm whose essence is as follows. On the first phase, a certain number of initial submatrices (seeds) are randomly generated. Then on the second phase of the algorithm execution each row and each column is checked so as to determine an action that would lead to the submatrix with the least residue. An action can be either adding the chosen row or column to the initial submatrix or deleting the chosen row or column from that submatrix. After the first action is taken, the rows and columns of the initial matrix are checked to determine the second action. The process is repeated until the maximal δ-cluster is formed on the basis of the initial submatrix. The above process is performed for all initial submatrices (seeds). As a result, the given number of δ-clusters is discovered. It is clear that the desired result can only be achieved if there is exactly this number of δ-clusters in the initial matrix.

III. δ-P-CLUSTERING

The basics of this technique are described in [5]. To understand its main principles, consider Fig.1.



a) initial profiles of the values of attribute A for three objects under ten conditions

(b) profiles of the values of attribute A for the same objects in the space of conditions 2, 5, 7, and 9.

Fig. 1. Representation of bicluster (b) masked in the initial data (a)

Fig. 1 depicts profiles of the values of attribute A in the space of ten conditions. In fact, no regularities in the changes of attribute values under different conditions can be seen here. Fig. 1, b illustrated profiles of the values of attribute A under a certain subset of conditions (conditions 2, 5, 7 and 9). The last case exemplifies an ideal bicluster that corresponds to the additive model.

The essence of the method is as follows. First, subsets of conditions are found in which two objects exhibit similar activity, i.e., changes in the attribute values for them occur similarly. Using such „atoms”, biclusters are then constructed in sequence. The similarity degree of attribute values in the atoms can be adjusted by specifying different values of parameter δ which restricts permissible deviations of attribute values for a pair of objects under a specific condition.

This technique enables simultaneous generation of a set of biclusters in the matrix of initial data. A distinctive feature of the technique is that the minimal dimensions of the target biclusters have to be specified a priori.

To make the conceptual basics of the method more clear, let us introduce the following denotations:

O = {o_1, … , o_n}- set of objects (set of rows of the initial data matrix);

A = {A_1, … , A_m) – set of sets of values of attribute A for each of conditions (set of columns of the initial data matrix);

(O′, A′) , O′ ⊆ O, A′ ⊆ A, - submatrix of the initial data matrix;

a_{ik} – value of attribute A out of set of values A_k for object o_i;

δ - user specified value of biclustering threshold;

n_c – a priori specified minimal number of columns in the bicluster;

n_r – a priori specified minimal number of rows in the bicluster.

In general, the task of finding δ-p-clusters can be formulated as follows: find all δ-p-clusters with $|O'| \geq n_r, |A'| \geq n_c.$ at the given matrix of initial data, threshold value, *δ*, minimal number of rows in the δ-p-cluster, n_r and minimal number of columns in the δ-p-cluster, n_c . To solve the task, in [5] an algorithm is proposed that corresponds to two phases of forming δ-p-clusters: (1) pairwise δ-p-clustering and (2) ″designing″ δ-p-clusters with dimensions no less than the specified.

To execute the algorithm of pairwise δ-p-clustering, the concept of ***Maximum Dimension Set (MDS)*** is introduced that is formulated in the set of rows as follows. Let (O', A') be a δ-p-cluster. A set of columns A' is a set of maximal dimension if there is no $A'' > A'$ such that (O', A'') also is a δ-p-cluster. In the same way the concept of the set of the maximal dimension in the set of rows can be formulated. Let (O', A') be a δ-p-cluster. A set of rows O' is a set of the maximal dimension if there is no $O'' > O'$ such that (O'', A') also is a δ-p-cluster. It is clear that the desired δ-p-clusters can only be formulated on the sets of the maximal dimension.

In accordance with the pairwise δ-p-clustering algorithm, MDS have to be generated for all possible pairs of rows $o_i, o_j \in O$ and all possible pairs of columns $A_k, A_l \in A.$

To discover MDS for a pair of objects $o_i, o_j \in O$, the following set of values is first calculated:

$$S(o_i, o_j, A) = \{a_{ik} - a_{jk} \,/\, A_k \in A, k = 1, ..., m\}. \quad (5)$$

In essence, each value S(.) is the difference of values of attribute A out of the set of (column) A_k for objects o_i and o_j.

Let us assume that the differences $S_k = a_{ik} - a_{jk}$, k = 1, … , m, are generated in the increasing order. Let us denote that sequence as $\vec{S}(o_i, o_j, A) = s_1, ..., s_m,$ where $s_k = S(o_i, o_j, A)$ and $s_p \leq s_r$ for $p < r.$ Then at the given set of columns A, $A_s \subseteq A$ is a set of maximal dimension for a pair of objects o_i, o_j, if and only if

1. $\vec{S}(o_i, o_j, A_s) = s_p, ..., s_r$ is an adjacent subsequence
$$\vec{S}(o_i, o_j, A) = s_1, ..., s_p, ..., s_r, ..., s_m;$$
2. $s_r - s_p \leq \delta,$ while $s_{r+1} - s_p > \delta$ and $s_r - s_{p-1} > \delta.$

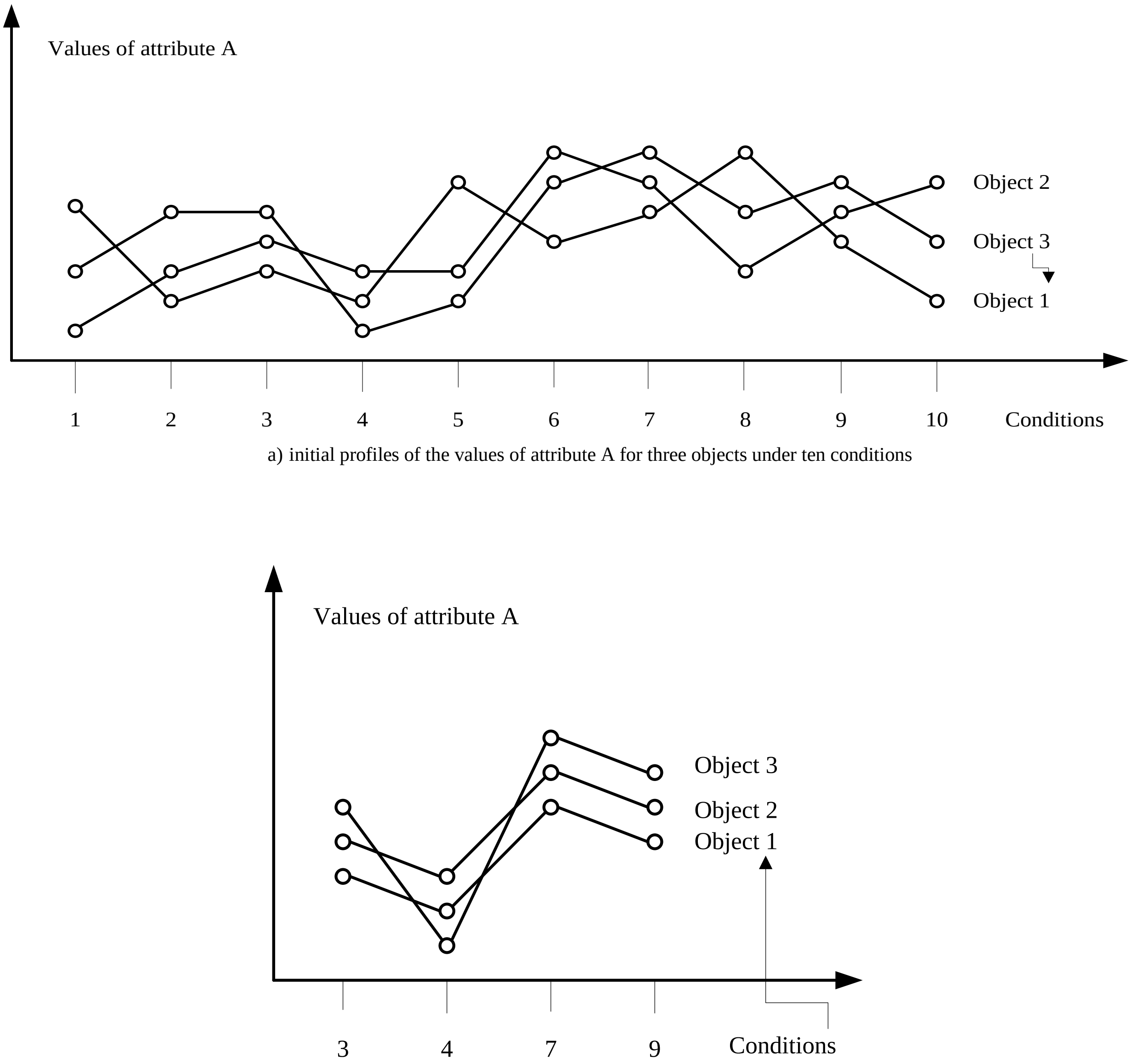
In practice, discovering MDS for a pair of objects $o_i, o_j \in O$ can be accomplished in this way. The process is started with the end elements located in the left and right end of the sorted sequence of differences and the right end is shifted by one position at a time. For each shift the difference of difference values is calculated until this difference becomes less or equal to *δ*. In this case elements between the two ends of the subsequence form an MDS. Then the left end is shifted by one position to the right and the process is repeated. The process of finding MDS is considered to be completed when there are no more elements for comparison.

Many obtained MDS may be superfluous since they do not meet the initial requirements regarding the number of rows and columns in a bicluster; due to that they have to be removed from further consideration. In fact, the non-prospective MDS are eliminated as follows: first, MDS are generated for the pairs of columns of the initial data matrix. After that, when forming MDS for the pairs of rows the non-prospective MDS are deleted using already generated MDS for the pairs of columns. The non-prospective MDS for the pairs of columns are then deleted using the previous results of deleting MDS for the pairs of rows. The procedure is completed in sequence for the pairs of rows and columns until all non-prospective MDS are deleted.

The obtained minimal set of MDS is recommended in [5] to be represented as a prefix tree. Each path in the tree represents MDS for a pair of objects. By simple manipulations the prefix tree is then transformed into a form that represents all relevant δ-p-clusters.

IV. OP-Clustering

The underlying concepts of this method are discussed in [2, 3]. The capability to derive only strong enough δ-p-clusters by the technique examined in the previous section, may become a limitation when the necessity to study more general trends of changes in the attribute values for a subset of objects under a subset of conditions arises. For illustration, consider Fig. 2.



a) initial profiles of the values of attribute A for three objects under ten conditions

b) profiles of the values of attribute A for the same objects under conditions 3, 4, 7 and 9

Fig. 2. Representation of a bicluster of general type (b) masked in the initial data (a)

Fig. 2 shows profiles of the values of attribute A for three objects under ten conditions. Here it is practically impossible to visually recognise any regularity in the behaviour of attribute values in a certain subset of conditions. Nevertheless, such regularities do exist and they are represented in Fig. 2, b. Note that unlike Fig. 1 where the changes in attribute values strictly follow the additive model, in Fig. 2, b only attribute values for objects 1 and 2 follow that model whereas the changes in attribute values for Object 3 are of more general nature although the trend of changes in attribute values for all objects is the same. The purpose of the method examined in this section is to discover biclusters with such general character of changes in attribute values. In [8] the authors call this kind of biclusters ***Order Preserving Cluster (OPC)***, which originated the name OP-clusters.

For further introduction of the material, let us employ denotation system given in Section 2.

Definition 1 [2, 3]. An object o_i ***is similar*** with regard to the values of attribute A with indexes k, k + 1, … , k + l, $1 \le k \le m$, $l \ne m$, in the ordered non-decreasing sequence of values of the attribute, if this condition is satisfied:

$$(a_{i(k+l)} - a_{ik}) \le G(\delta, a_{ik}), \qquad (6)$$

where $G(\delta, a_{ik})$ – function of grouping that determines equivalence class of values of attribute A.

If condition (6) is satisfied, a set of attribute values $\{a_{ik},\ldots, a_{i(k+l)}\}$ constitute ***a group*** for object o_i. The value of attribute a_{ik} is called ***support point*** for that group.

Definition 2 [2, 3]. Let o_i be an object in the matrix of initial data and (g_{0i1}), (g_{0i2}), … , (g_{oir}) be a sequence of groups of similarity of values of attribute A for that object represented in the non-decreasing order of its values. Object o_i is an "example upwards" in the ordered list of values of attribute a_{i1}, a_{i2}, … , a_{ik}, if a_{i1}, a_{i2}, … , a_{ir} is a subsequence (g_{0i1}), (g_{0i2}), … , (g_{0ir}).

Definition 3 [2, 3]. Let O′, O′ ⊆ O, be a subset of objects (rows) in the matrix of initial data and A′, A′ ⊆ A, be a subset of sets of values of attribute A (columns) in the matrix of initial dat. A submatrix (O′, A′) is an OP-cluster if there exists such a transposition of columns in A′ in which every object in O′ is "an example upwards".

An OP-cluster in essence encompasses a subset of those objects for which the values of attribute A exhibit a correlated activity in the subset of conditions.

In general, the task of forming OP-clusters in the matrix of initial data can be stated as follows: provided that the grouping threshold δ, minimal number of columns n_c and minimal number of rows n_r are specified, it is necessary to generate all possible submatrices (O', A') of the maximal size such that each submatrix (O', A') is an OP-cluster according to its definition and $|O'| \geq n_c$, $|A'| \geq n_r$.

To solve that task, an algorithm is proposed in [2, 3] that consists of these two phases:

1. Pre-processing the initial data, i.e., the transformation of each row of the matrix of initial data into a sequence of groups of similarity.

2. Forming a set of rows containing frequent subsequences in the sequences determined at the first phase of the algorithm execution.

After the initial data are pre-processed, each row of the initial matrix is transformed into an ordered sequence of value sets of attribute A. Then all those sequences are represented as paths in the so-called OPC-tree. In [2, 3] a set of procedures for a successive transformation of the OPC-tree is provided. As a result of such transformation, a consecutive concentration of frequent subsequences occurs, which in the long run leads to the identification of all OP-clusters available in the initial matrix and meeting the dimensionality requirements.

V. Conclusions

Biclustering methods have been initially developed for discovering local correlations between gene expressions and conditions. In general, biclustering tasks are NP-hard; due to that, all the algorithms considered in this paper are of heuristic nature. A pioneering work [1] - despite all its shortcomings – initiated a whole direction in the development of more effective biclustering methods. The technique proposed in [6] made it possible to eliminate all essential shortcomings of the first method. However, both these techniques possess a common drawback: a certain submatrix of a bicluster is not necessarily a bicluster. The technique discussed in [5] is free from that drawback. It enables strong discovering of biclusters more or less close to the ideal multiplicative or additive model. The degree of similarity can be regulated by specifying a priori a shifting parameter δ. The technique considered in [2 and 3] is the most common biclustering method that allows one to find biclusters for which the changes in the attribute values are occurring in a rather deliberate but correlated way.

To finalise, the techniques proposed in [5] and [2, 3] have to be considered most appropriate for practical use. It should be noted that both these methods are quite complicated from the computational point of view and require professional software realisation.

VI. Acknowledgements

This work has been partly supported by Latvia-Belarus Co-operation program in Science and Engineering within the project 'Development of a complex of intelligent methods and medical and biological data processing algorithms for oncology disease diagnostics improvement'.

References

[1] Y. Cheng and G. Church, Biclustering of expression data. *Proceedings of 8th International Conference on Intelligent Systems for Molecular Biology, (ISMB 2000).* San Diego, California, USA. August 19-23, 2000, pp.93-103.

[2] J. Liu and W.Wang, *Flexible clustering by tendency in high dimensional spaces.* Technical Report TRO3 -009, Computer Science Department, UNC-CH, 2003.

[3] J. Liu and W.Wang, OP-cluster: Clustering by tendency in high dimensional space. *Proceedings of the 3rd IEEE International Conference on Data Mining*, (ICDM 2003), 19-22 November 2003. Melbourne, Florida, pp. 187 – 194.

[4] S.C. Madeira and A.L. Olivera, Biclustering Algorithms for Biological Data Analysis: A Survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1, 2004, pp. 24 – 25.

[5] H. Wang, W. Wang, J. Yang, and P.S. Yu. Clustering by pattern similarity in large data sets. *Proceedings ACM SIGMOD 2002*, Madison, USA, June, 2002, pp. 394 – 405.

[6] J. Yang, W. Wang and P.S. Yu. δ-cluster: Capturing subspace correlation in a large data set. *Proceedings of the 18th International Conference on Data Engineering (ICDE)*, 26 February - 1 March 2002, San Jose, California, USA, 2002, pp. 517 – 528.

Oleg Uzhga-Rebrov is a Professor at the Faculty of Economics in Rezekne Higher Education Institution (Latvia). He received his Doctoral Degree in Information Systems from Riga Technical University in 1994. His research interests include different approaches to processing incomplete, uncertain and fuzzy information, in particular, fuzzy sets theory, rough set theory as well as fuzzy classification and fuzzy clustering techniques and their applications in bioinformatics.
Contact information: Rezekne Higher Education Institution, 90 Atbrivosanas aleja, Rezekne LV-4600, Latvia.
E-mail: ushga@ru.lv.

Galina Kuleshova is a Researcher at the Faculty of Computer Science and Information Technology at Riga Technical University (Latvia). She received her M.Sc. degree in *Decision Support Systems* from Riga Technical University in 1996. Current research interests include artificial neural networks, data mining, classification methods and bioinformatics.
Contact information: Institute of Information Technology, Riga Technical University, 1 Kalku Str., Riga LV-1658, Latvia.
E-mail: galina.kulesova@cs.rtu.lv.

Oļegs Užga-Rebrovs, Gaļina Kuļešova. Biklasterizācijas algoritmu evolūcija

Biklasteru izdalīšanas uzdevumi radās sakarā ar gēnu izteiksmju mikromasīvu tehnoloģijas attīstību. Sākotnējo datu apstrādei, lai iegūtu tajos esošo informāciju, sākumā plaši tika izmantotas esošās metodes, galvenokārt dažādas klasterizācijas metodes. Šādas metodes ļauj noteikt gēnu uzvedības galvenās likumsakarības visu apstākļu kopā. Tomēr sakarā ar gēnu mikromasīvu datu analīzes uzdevumu radās cita problēma. Mikromasīvu kopā, kurā katrs mikromasīvs tiek iegūts specifiskos apstākļos, atsevišķi gēni var parādīt savu aktivitāti tikai kādā apstākļu apakškopā. Gēnu apakškopu izdalīšanai, kurām ir līdzīga uzvedība apstākļu apakškopā, standarta klasterizācijas algoritmi principiāli nevar tikt pielietoti, tāpēc, ka šie algoritmi strādā tikai visu atribūtu (apstākļu) kopā. Vajadzīgi speciāli algoritmi, kuri ļauj tādā vai citā veidā formēt vajadzīgās gēnu apakškopas atbilstošajās apstākļu apakškopās. Ņemot vērā, ka datu apstrādei šajā gadījumā jābūt veiktai divās dimensijās vienlaicīgi (gēni un apstākļi), gēnu relevanto apakškopu formēšanas procesam tiek izmantots vispārīgs nosaukums „biklasterizācija", lai gan atsevišķām metodēm autori izmanto speciālus nosaukumus. Ir izstrādāts liels biklasterizācijas algoritmu skaits, kuri izmanto tos vai citus biklasteru izdalīšanas principus. Šajā darbā izskatīti četri biklasterizācijas algoritmi, kuri ļauj atspoguļot algoritmu attīstības evolūciju. Izskatītie algoritmi ir precīzi tādā nozīmē, ka tie izmanto oriģinālus sākotnējos datus bez datu pārveidošanas izplūdušā formā. Biklasterizācijas algoritms, kas pielieto vidēja kvadrātiska atlikuma koncepciju biklasteru izdalīšanai sākumdatu kopā, ir vēsturiski pirmais algoritms, bet δ-klasterizācijas algoritmu var uzskatīt par pirmā algoritma uzlabotu versiju. δ-p-klasterizācijas algoritms izmanto citu biklasteru izdalīšanas principu. Ar tā palīdzību var tikt izdalīti objektu biklasteri ar vienādu uzvedību apstākļu apakškopās. Savukārt OP-klasterizācijas algoritms ļauj izdalīt objektu biklasterus ar atribūtam līdzīgām izmaiņu tendencēm apstākļu apakškopās. Algoritmu efektivitāte būtiski pieaug algoritmu apskatītajā secībā, bet tāds efektivitātes pieaugums tiek sasniegts tikai uz algoritmu būtiska skaitļošanas sarežģītības pieauguma rēķina.

Олег Ужга-Ребров, Галина Кулешова. Эволюция алгоритмов бикластеризации

Задачи выделения бикластеров возникли в связи с развитием технологий микромассивов выражений генов. Для обработки исходных данных с целью выделения имеющихся в них знаний вначале широко использовались существующие методы, главным образом, различные методы кластеризации. Такие методы позволяют определить общие закономерности поведения генов на всём множестве условий. Однако, в связи со спецификой задачи анализа данных микромассивов выражений генов возникла другая проблема. Во множестве микромассивов, полученных при разных условиях, отдельные гены могут проявлять свою активность только на некотором подмножестве условий. Для выделения подмножеств генов, имеющих связное поведение на подмножествах условий, стандартные алгоритмы кластеризации принципиально не могут быть применены, поскольку такие алгоритмы работают только на всём множестве атрибутов (условий).. Необходимы специальные алгоритмы, позволяющие тем или иным способом формировать требуемые подмножества генов на соответствующих подмножествах условий.. Принимая во внимание, что обработка данных в данном случае должна одновременно производиться в двух измерениях (гены и условия), процесс формирования релевантных подмножеств генов получил общее название бикластеризации хотя для отдельных методов авторы используют специальные названия. Разработано большое число алгоритмов бикластеризации, использующих те или иные принципы выделения бикластеров. В настоящей работе рассмотрены четыре алгоритма, которые позволяют отразить эволюцию такого рода алгоритмов. Рассмотренные алгоритмы являются чёткими в том смысле, что они используют оригинальные исходные данные без перевода их в нечёткую форму. Исторически первым является алгоритм бикластеризации, использующий концепцию среднего квадратичного остатка для выделения бикластеров на исходном множестве данных. Алгоритм δ-кластеризации можно рассматривать как улучшенную модификацию первого алгоритма. В свою очередь, алгоритм δ-p-кластеризации использует иной принцип выделения бикластеров. С его помощью могут быть выделены бикластеры объектов с одинаковым поведением на подмножествах условий. Алгоритм OP-кластеризации позволяет выделять бикластеры объектов со сходными тенденциями изменения атрибута на подмножествах условий. Эффективность алгоритмов существенно возрастает на рассмотренной последовательности, однако, такое возрастание эффективности достигается за счёт существенного повышения их вычислительной сложности.