RESEARCH ARTICLE OPEN

Survey on Mining Order-Preserving Sub Matrices

Reeta Dangi, R.C. Jain, Vivek Sharma
Department of Information Technology SATI Vidisha, India
Director of SATI Vidisha, India
Department of Information Technology SATI Vidisha, India

Abstract: -Order-preserving sub matrices (OPSM's) have been shown useful in capturing concurrent patterns in data when the relative magnitudes of data items are more important than their exact values. For example, in analyzing gene expression profiles obtained from micro-array experiments, the relative magnitudes are important both since they represent the change of gene activities across the experiments, and since there is typically a high level of noise in data that makes the exact values un-trustable. To manage with data noise, repeated experiments are often conducted to collect multiple measurements.
Keywords: -Order-preserving sub matrices, Simultaneous Clustering.

I. INTRODUCTION

In bioinformatics community, a large number of genes are studied by using DNA micro-array technology to obtain gene expression data. Gene expression data are usually organized as matrices, in which each row represents one gene and each column represents a sample for the experiment, and each item records the expression value of one gene under an experiment sample. Through the analysis of expression data, we can discover information about the genes.

Clustering is helpful to find different functional categories of genes. Among various kinds of clustering approaches, Order-Preserving Sub Matrix has been a useful method to discover groups of genes that share some common functions.

Simultaneous clustering, usually designated by bi-clustering, co-clustering, 2-way clustering or block clustering, is an important method in two-way data analysis. A number of algorithms that perform simultaneous clustering on rows and columns of a matrix have been proposed to date. The goal of simultaneous clustering is to find sub-matrices, which are subgroups of rows and subgroups of columns that exhibit a high correlation. This type of algorithms has been proposed and used in many fields, such as bio-informatics [1], web mining [2], text mining [3] and social network analysis [4].

II. OVERVIEW OF SIMULTANEOUS CLUSTERING PROBLEM

Clustering is the grouping together of similar subjects. Standard clustering methods consider the value of each point in all dimensions, in order to form group of similar points. This kind of one-way clustering techniques is based on similarity between subjects across all variables.

Simultaneous clustering algorithms seeks "blocks" of rows and columns thatare interrelated. They aim to identify a set of bi-clusters *Bk*(*Ik, Jk*), where *Ik* is a subset of the rows X and *Jk* is a subset of the columns Y. *Ik* rows exhibit similar behavior across *Jk* columns, or vice versa and every bi-cluster *Bk* satisfies some criteria of homogeneity. A bi-clustering method may assume a specificstructure and data type. Madeira and Oliveira launch in their survey [5]some bi-clustering structures defined by: single bi-cluster, exclusive rows bi-clusters, exclusive columns bi-clusters, non overlapping bi-clusters with tree arrangement, and arbitrarily positioned overlapping bi-clusters. Bi-clusters can be with constantvalues, with constant values on rows or columns, with coherent values or withcoherent evolution. There are many advantages in a simultaneous rather thanone way clustering (table 1). In fact, simultaneous clustering may highlight the association between the row and column clustering that appears from the dataanalysis as a linked clustering. in addition, it allows the researcher to deal withsparse and high dimensional data matrices [6]. Simultaneous clustering is alsoan interesting paradigm for unsupervised data analysis as it is more useful, has less parameters, is scalable and is able to effectively interlink row and column information.

Table 1. Comparison between Clustering and Simultaneous clustering

Clustering	Simultaneous Clustering
Applied to each the rows or the columns of the data matrix **separately** ⇒**Global model**.	performs clustering in the two dimensions **simultaneously** ⇒**Local model**.
produce clusters of rows or clusters of columns.	seeks **blocks** of rows **and** columns that are

	interrelated.
Each subject in a given subject cluster is defined using **all** the variables. Each variable in a variable cluster characterizes **all** subjects.	Each subject in a bi-cluster is selected using **only** a subset of the variables and each variable in a bi-cluster is selected using **only** a subset of the subjects.
Clusters are **exhaustive**	The clusters on rows and columns **should not be exclusive** and/or **exhaustive**

Simultaneous Clustering Approaches

A survey of simultaneous clustering algorithms applied on biological data has been given by Madeira and Oliveira. These algorithms are based on five approaches: Iterative Row and Column Clustering Combination (IRCCC), Divide and Conquer (DC), Greedy Iterative Search (GIS), Exhaustive Bi-cluster Enumeration (EBE) and Distribution Parameter Identification (DPI). The IRCCC approach consists to apply clustering algorithms to the rows and columns of the data matrix, independently, and then to combine results using some sort of iterative process The algorithms based on DC approach begin with the entire data in one block (bi-cluster) and identifies bi-clusters at each iteration by splicing a given block into two pieces. GIS approach creates bi-clusters by adding or removing rows/columns from them, using a criterion that maximizes the local increase EBE approach identifies bi-clusters using an exhaustive enumeration of all possible bi-clusters in the data matrix. DPI approach assumes that the bi-clusters are generated using a given statistical model and tries to identify the distribution parameters that fit the available data, by minimizing a certain criterion through an iterative move toward All the algorithms presented in this survey analyze biological data from gene expression matrices. Given that there are a number of algorithms based on bipartite graph model [7], mixture model [8] and information theory [9], which are applied in other fields such as text mining, web mining and information recovery, we propose to categorize simultaneous clustering methods into five categories: bipartite Graph methods, variance minimization techniques, two-way clustering methods, motif and pattern recognition methods and probabilistic and generative methods.

The bipartite graph methods consists in modeling rows and columns as a weighted bipartite graph and assigning weights to graph edges using similarity measure methods. The created bipartite graph is then partitioned in a way that minimizes the cut of the divider i.e. the sum of the weights of the crossing edges between parts of the partition. In [10], the authors created a word-document bipartite graph. The graph was partitioned using a partial singular value decomposition of the associated edge weight matrix of the bipartite graph. Dhillon [11] used the spectral method for partitioning the bipartite graph constructed in the same way as in [12]. Authors proposed an isoperimetric co-clustering algorithm (ICA) for partitioning the word file matrix. ICA used the same model than spectral partitioning but instead of searching the solutions of the singular word-document system of linear equations, it converts the scheme to a nonsingular system of equations which is easier to solve. The bipartite graph techniques are also used for gene expression analysis. One case is Statistical-Algorithmic Method for Bi-cluster Analysis (SAMBA).

The variance minimization methods define clusters as blocks in the matrix with minimal deviation of their elements. This definition has been already measured by Hartiganand extended by Tibshirani et al. Some examples are the δ-cluster methods, such as δ-ks clusters, δ-p Clusters and δ-bi-clusters, which search for blocks of elements having a deviation below δ. flexible Overlapped bi-Clustering (FLOC) introduced by extend Cheng and Church δ-bi-clusters by dealing with missing values. – Two-way clustering methods use one-way clustering such as k-means Self-Organizing Maps, Expectation-Minimization algorithm or hierarchical clustering algorithm to produce clusters on both dimensions of the data matrix separately. One-dimension results are then combined to produce subgroups of rows and columns called bi-clusters. These methods identify clusters on rows and columns but not directly bi-clusters.

Motif and pattern recognition methods define a bi-cluster as samples sharing a common prototype or motif. To simplify this task, some methods discretize the data such as xMOTIF [13] or binarize the data such as Bimax [14]. Order-Preserving Sub Matrices (OPSM) [15] searches for blocks having the same order of values in their columns. Spectral clustering (SPEC) [16] performs a singular value decomposition of the data matrix after normalization. Contiguous column coherent (CCC bi-clustering) [17] is a method for gene expression time series, which finds patterns in nearby columns.

Probabilistic and generative methods use model-based techniques to define bi-clusters [18]. Probabilistic Relational Models (PRMs) [19] and their extension ProBic [20] are fully generative models that combine probabilistic modeling and relational logic. C Monkey [21] is a generative approach which models biclusters by Markov chain processes. GU and Liu [22] generalized the plaid models proposed in [23] to fully generative models called Bayesian BiClustering model (BBC). The

latter models introduced in [24] is generative models which have the advantage that they select models using well-understood model selection techniques such as maximum likelihood. Costa et al. [25] introduced a hierarchical model-based co-clustering algorithm. In their method the co-occurrence matrix is characterized in probabilistic terms, by estimating the joint distribution between rows and columns.

III. STUDY ON MINING ORDER-PRESERVING SUB MATRICES

Order-Preserving Sub matrix (OPSM) is a data pattern particularly useful for discovering trends in noisy data. The OPSM problem applies to a matrix of numerical data values. The objective is to discover a subset of attributes (columns) over which a subset of tuples (rows) exhibit similar rises and falls in the tuples' values. For instance, when analyzing gene expression data from microarray experiments, genes (rows) with concurrent changes of mRNA expression levels across different time points (columns) may share the same cell-cycle related properties [26]. Due to the high level of noise in typical microarray data, it is typically more meaningful to compare the relative expression levels of different genes at different time points rather than their total values. Genes that exhibit simultaneous rises and falls of their expression values across different time points or experiments reveal interesting patterns and knowledge.

The original OPSM problem was first proposed by Ben-Dor and company. [27]:

Definition 1: Given an n _ m matrix (dataset) D, an order-preserving sub matrix (OPSM) is a pair (R; P), where R is a subset of the n rows (represented by a set of row ids) and P is a permutation of a subset of the m columns (represented by a sequence of column ids) such that for each row in R, the data values are monotonically increasing with respect to P, i.e., DiPj< DiPj0 ; 8i 2 R; 1 _ j < j0 _ jPj, where Drc denotes the value at row r and column c of D.

TABLE 1
A dataset without repeated measurements

	A	*b*	*c*	*d*
row 1	49	38	115	82
row 2	67	96	124	48
row 3	65	67	132	95
row 4	81	115	133	62

For example, Table 1 shows a dataset with 4 rows and 4 columns. The values of rows 2, 3 and 4 rise from a tob, so ({2, 3, 4}, <a, b>) is an OPSM. For simplicity, in this study we assume that all values in a row are unique.

We say that a row supports a permutation if its values increase monotonically with respect to the permutation. In the above example, rows 2, 3 and 4 support the permutation <a, b>, but row 1 does not. For a fixed dataset, the rows that support a permutation can be unambiguously identified. In the following discussion, we will refer to an OPSM simply by its variation which will also be called a *pattern*.

An OPSM is said to be frequent if the number of supporting rows is not less than a support threshold, ρ. Given a dataset, the basic OPSM mining problem is to identify all frequent OPSM's. In the gene expression context, these OPSM's correspond to groups of genes that have similar activity patterns, which may suggest shared regulatory mechanisms and/or protein functions. In microarray experiments, each value in the dataset is a physical measurement subject to different kinds of errors. A drawback of the basic OPSM mining problem is that it is sensitive to noisy data. In our previous example, if the value of column *a* is slightly increased in row 3, say from 65 to 69, then row 3 will no longer support the pattern <*a, b*>, but will support <*b, a*>instead.

To combat errors, experiments are often repeated and multiple measured values (called replicates) are recorded. The replicates allow a better estimate of the actual physical quantity. certainly as the cost of microarray experiments has been dropping, research groups have been obtaining replicates to strike for higher data quality. For example, in some of the microarray datasets we use in our study, each experiment is repeated 3 times to produce 3 measurements of every data point. Studies have clearly shown the importance of having multiple replicates in improving data quality.

TABLE 2
A dataset with repeated measurements

	a1	*a2*	*a3*	*b1*	*b2*	*b3*	*c1*	*c2*	*c3*	*d1*	*d2*	*d3*
row 1	49	55	80	38	51	81	115	101	79	82	110	50
row 2	67	54	130	96	85	82	124	92	94	48	37	32
row 3	65	49	62	67	39	28	132	119	83	95	89	64
row	81	83	105	115	110	87	133	108	105	62	52	51

4												

Different replicates, however, may support different OPSM's. For example, Table 2 shows a dataset with two more replicates added per experiment. From this dataset, we see that it is no longer clear whether row 3 supports the <*a*, *b*>pattern. For instance, while the replicates a_1, b_1 support the pattern, the replicates a_1, b_2 do not.

Our example illustrates that the original OPSM definition is not robust against noisy data. It also fails to take advantage of the additional information provided by replicates. There is thus a need to modify the definition of OPSM to handle repeated measurements. Such a definition should satisfy the following requirements:

1) If a pattern is supported by all combinations of the replicates of a row, the row should contribute a high support to the pattern. For example, for row 3, the values of column *b* are clearly smaller than those of column *c*. All 3 × 3 = 9 replicate combinations of *b* and *c* values (b_1, c_1), (b_1, *c2*)... (b_3, *c3*) support the <*b*, *c*>pattern. Row 3 should thus strongly support <*b*, *c*>.
2) If the value of a replicate largely deviates from other replicates, it is most likely due to error. The replicate should not severely affect the support of a given pattern. For example, we see that row 2 generally supports the pattern <*a*, *c*>if we ignore a_3, which is abnormally large (130) when compared to a_1 (67) and a_2 (54), and is thus likely an error. The support of <*a*, *c*>contributed by row 2 should only be mildly reduced due to the presence of a_3.
3) If the replicates largely disagree on their support of a pattern, the overall support should reflect the uncertainty. For example, in row 4, the values of *b* and *c* are mingled. Thus, row 4 should neither strongly support <*b*,*c*>nor <*c*,*b*>.

The first two requirements can be satisfied by summarizing the replicates by robust statistics such as medians, and mining the resulting dataset using the original definition of OPSM. However, the third requirement cannot be satisfied by any single summarizing statistic. This is because under the original definition, a row can only either fully support or fully not support a pattern, and thus the information of uncertainty is lost. To tackle this problem, we propose a new definition of OPSM and the corresponding mining problem based on the concept of fractional support:

Definition 2: The partial support $s_i(P)$ of a pattern *P* contributed by a row *i* is the number of replicate combinations of row *i* that support the pattern, divided by the total number of replicate combinations of the columns in *P*.

For example, for row 1, the pattern <*a*,*b*,*d*>is supported by 8 replicate combinations: ha_1,b_2,d_1i, ha_1,b_2,d_2i, ha_1,b_3,d_1i, ha_1,b_3,d_2i, ha_2,b_3,d_1i, ha_2,b_3,d_2i, ha_3,b_3,d_1i, and ha_3,b_3,d_2i out of 3^3 = 27 possible combinations. The fractional support s_1 (<*a*,*b*,*d*>) is therefore 8/27. We use $sn_i(P)$ and $sd_i(P)$ to denote the numerator and the denominator of $s_i(P)$, respectively. In our example, sn_1(<*a*,*b*,*d*>) = 8 and sd_1(<*a*,*b*,*d*>) = 27.

If we use fractional support to indicate how much a row supports an OPSM, all the three requirements we stated above are satisfied. Firstly, if all replicate combinations of a row support a certain pattern, the fractional support contributed will be one, the maximum fractional support. Secondly, if one replicate of a column *j* deviates from the others, the replicate can at most change the fractional support by $\frac{1}{r(j)}$, where *r* (*j*) is the number of replicates of column *j*. This has small effects when the number of replicates *r* (*j*) is large. Finally, if only a fraction of the replicate combinations supports a pattern, the resulting fractional support will be fuzzy (away from 0 and 1), which reflects the doubt

Based on the definition of fractional support, the support of a pattern *P* is defined as the sum of the fractional supports of *P* contributed by all the rows: *s* (*P*) = $\sum_i s_i(P)$. A pattern *P* is frequent if its support is not less than a given support threshold ρ. Our new OPSM mining problem OPSM-RM (OPSM with repeated measurements) is to identify all frequent patterns in a data matrix with replicates:

Definition 3: Given a dataset, the OPSM-RM difficulty asks for the set of all OPSMs each of which having a total fractional support from all rows not less than a given support threshold.

From the definition of fractional support, we can observe the combinatorial nature of the OPSM-RM problem — the number of replicate combinations grows exponentially with respect to the pattern length. The objective of this work is to derive efficient algorithms for mining OPSM-RM. By proving a number of interesting properties and theorems, we propose pruning techniques that can significantly reduce mining time [28].

IV. OVERVIEW OF DATASET

The readout of a DNA chip containing n genes consists of n real numbers that represent the expression level of each gene, either as an complete or as a relative quantity (with respect to some reference). When the readouts for m experiments (tissues) are joint, each gene yields a vector of m real numbers.

Table 1. The Ranks of the Three Genes g1;g2;g3 Induce a Common Permutation When Restricted to Columns t1;t2;t3;t4;t5

Gene n tissue	t1	t2	t3	t4	t5
g1	7	13	19	2	50
g2	19	23	39	6	42
g3	4	6	8	2	10
Induced permutation	2	3	4	1	5

To make our results independent of the scaling of the data, we think only the relative ordering of the expression levels for each gene, as different to the correct values. This motivates us to consider the permutation induced on the m numbers by sorting them. so, we view the expressed data matrix, D, as an n-by-m matrix, where each row corresponds to a gene and each column to an experiment. The m entries in each row are a permutation of the numbers {1… m}. The (I, j) entry is the rank of the readout of gene iin tissue j, out of the m readouts of this gene. characteristic values for n and m are in the ranges 500 ≤n ≤15,000 and 10 ≤ m ≤150.

The computational task we address is the identification of large order-preserving sub matrices (OPSMs) in an n x m matrix D. A sub matrix is order preserving if there is a permutation of its columns under which the sequence of values in every row is strictly increasing. In the case of expression data, such a sub matrix is determined by a set of genes G and a set of tissues T such that, within the set of tissues T, the term levels of all the genes in G have the same linear ordering.

V. Conclusion

In this paper we review Order-preserving sub matrices (OPSM's) which is useful in capturing concurrent patterns in data when the relative magnitudes of data items are more important than their exact values. To cope with data noise, repeated experiments are often conducted to collect multiple measurements. We also review some basic methods of Simultaneous Clustering Problem.

REFRENCES

[1]. Madeira, S.C., Oliveira, A.L.: Biclustering Algorithms for Biological Data Analysis: A Survey. IEEE/ACM Trans. on Comp. Biol. and Bioinfor., 24–45 (2004).

[2]. Charrad, M., Lechevallier, Y., Ahmed, M.b., Saporta, G.: Block Clustering for Web Pages Categorization. In: Corchado, E., Yin, H. (eds.) IDEAL 2009. LNCS, vol. 5788, pp. 260–267. Springer, Heidelberg (2009).

[3]. Bichot, C.E.: Co-clustering documents and words by minimizing the normalized cut objective function. JMMA 9, 131–147 (2010).

[4]. Grimal, C., Bisson, G.: Classification a partird'une collection de matrices.CAp2010 (2010).

[5]. Prelic, A., Bleuler, S., Zimmermann, P.: A systematic comparison and evaluation of bi clustering methods for gene expression data. Bioinformatics, 122–129 (2006).

[6]. Balbi, S., Miele, R., Scepi, G.: Clustering of documents from a two-way viewpoint. In: 10th Int. Conf. on Statistical Analysis of Textual Data (2010).

[7]. Dhillon, I.S.: Co-clustering documents and words using bipartite spectral graph partitioning. In: 7th ACM SIGKDD 2001, California, pp. 269–274 (2001).

[8]. Nadif, M., Govaert, G.: Block clustering of contingency table and mixture model. In: Famili, A.F., Kok, J.N., Pen˜a, J.M., Siebes, A., Feelders, A. (eds.) IDA 2005.

[9]. Dhillon, I.S., Mallela, S., Modha, D.S.: Information-theoretic co-clustering. In: ACM SIGKDD, pp. 89–98. ACM, Washington DC (2003).

[10]. Zha, H., He, X., Ding, C., Simon, H., Gu, M.: Bipartite Graph Partitioning and Data Clustering. In: ACM Conf. on Inf. and Knowledge Management, pp. 25–32 (2001).

[11]. Tanay, A., Sharan, R., Shamir, R.: Biclustering Algorithms: A Survey. In: Aluru,S. (ed.) Handbook of Comp. Molecular Biology, Chapman, Boca Raton (2004).

[12]. MalikaCharrad and Mohamed Ben Ahmed, "Simultaneous Clustering: A Survey", LNCS 6744, pp. 370–375, 2011. Springer-Verlag Berlin Heidelberg 2011.

[13]. Murali, T.M., Kasif, S.: Extracting conserved gene expression motifs from gene expression data. In: Pacific Sym. on Biocomputing, Hawaii, USA, pp. 77–88 (2003).

[14]. Prelic, A., Bleuler, S., Zimmermann, P.: A systematic comparison and evaluation of bi clustering methods for gene expression data. Bioinformatics, 122–129 (2006).

[15]. Ben-Dor, A., Chor, B., Karp, R.: Discovering local structure in gene expression data: The order–preserving submatrix problem. J. of Comput. Biol. 10, 373–384 (2003).

[16]. Klugar, Y., Basri, R., Chang, J.T., Gerstein, M.: Spectral bi clustering of microarray

data: co clustering genes and conditions. Genome Research 13, 703–716 (2003).

[17]. Madeira, S.C., Teixeira, M.C.: Identification of regulatory modules in time series gene expression data using a linear time bi clustering algorithm. IEEE ACM (2010).

[18]. Hochreiter, S., Bodenhofer, U., Heusel, M., Mayr, A.: FABIA: factor analysis for bicluster acquisition. Bioinformatics journal 26(12), 1520–1527 (2010).

[19]. Getoor, L., Friedman, N., Koller, D., Taskar, B.: Learning probabilistic models of link structure. J. Mach. Learn. Res. 3, 679–707 (2002).

[20]. Van den, B.T.: Robust Algorithms for Inferring Regulatory Networks Based onGene Expression Measurements. PhD Thesis (2009).

[21]. Reiss, D.J.: Integrated bi clustering of heterogeneous genome-wide datasets for theinference of global regulatory networks. BMC Bonior. 280–302 (2006).

[22]. Lazzeroni, L., Owen, A.: Plaid models for gene expression data. Technical report, Stanford University (2002).

[23]. Caldas, J., Kaski, S.: Bayesian bi clustering with the plaid model. In: IEEE Intern. Workshop on Machine Learning for Signal Processing, pp. 291–296 (2008).

[24]. Gu, J.: Bayesian biclustering of gene expression data. BMC Genomics (2008).

[25]. Costa, G., Manco, G., Ortale, R.: A hierarchical model-based approach to co clustering high-dimensional data. In: ACM sym. on App. comput. pp. 886–890 (2008).

[26]. P.T. Spellman, G. Sherlock, M.Q. Zhang, V.R. Iyer, K. Anders,M.B. Eisen, P.O. Brown, D. Botstein, and B. Futcher, "ComprehensiveIdentification of Cell Cycle-regulated Genes of the YeastSaccharomyces Cerevisiae by Microarray Hybridization," MolecularBiology of the Cell, vol. 9, no. 12, pp. 3273-3297, 1998.

[27]. A. Ben-Dor, B. Chor, R.M. Karp, and Z. Yakhini, "DiscoveringLocal Structure in Gene Expression Data: The Order-PreservingSubmatrix Problem," J. Computational Biology, vol. 10, nos. 3/4, pp. 373-384, 2003.

[28]. Kevin Y. Yip, Ben Kao, Xinjie Zhu, Chun Kit Chui,Sau Dan Lee, and David W. Cheung, "Mining Order-Preserving Submatricesfrom Data with Repeated Measurements" IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 25, NO. 7, JULY 2013.