The Second International Symposium on Optimization and Systems Biology (OSB'08)
Lijiang, China, October 31– November 3, 2008

Biclustering Based on Self-multiplication of Matrix*

Ru-Xin Qin[1] Ying-Jie Tian[2] Jing Chen[1]
Nai-Yang Deng[1,†]

[1]College of Science, China Agricultural University, 100083, Beijing, China
[2]Academy of Sciences Research Center on Fictitious Economy,
Chinese Academy of Sciences, 100080, Beijing, China

Abstract A biclustering algorithm based on self-multiplication of 0-1 matrix is proposed in this paper. Based on an important property of self-multiplication of 0-1 matrix, we construct BBSM algorithm for matrix with non-overlapping biclusters and overlapping biclusters seperately, and prove that this algorithm can obtain the s most largest biclusters for both cases.

Keywords Bicluster; Gene expression data; Self-multiplication of matrix

1 Introduction

DNA microarray technology has recently become a central role in biological and biomedical research. It enables measuring the expression level of many thousands of genes within a number of different experimental conditions simultaneously. The relative abundance of the mRNA of a gene under a specific experimental condition (or sample) is called the expression level of a gene. The expression level of a large number of genes of an organism under various experimental conditions can be arranged in a data matrix, also known as gene expression data matrix, where rows correspond to genes and columns to conditions. Thus each entry of this matrix is a real number representing the expression level of a gene under a specific experiment. One of the objectives of gene expression data analysis is biclustering–to group genes according to their expression under multiple conditions.

Biclustering was introduced in the 1970s [1]. Cheng and Church [3] were the first to apply it to gene expression data analysis. Biclustering attempts to isolate genes that are co-expressed under a specific set of conditions, it is to say that Biclustering attempts to find a submatrix with some coherence in a given matrix.

In practice, the next problem of biclustering is more interesting in most cases: finding the s most largest biclusters in size, where s is a threshold value less than the number of all the biclusters in a given matrix. When $s = 1$, the problem is finding the largest bicluster

*Supported by the Key Project of the National Natural Science Foundation of China(No.10631070) and the National Natural Science Foundation of China(No.10601064)
†Corresponding author. E-mail: dengnaiyang@vip.163.com

in a given matrix. For the problem of finding the s most largest biclusters in size, the optimal solution is can not be obtained by the most of the algorithm in exist except the Bimax algprithm in [8].

Instead of the above general biclustering problem, we consider a particular and important case when the gene expression data matrix is a 0-1 matrix. Here the coherence implies to find bicluster whose elements are all 1. Thus this paper is concerned with the problem of finding the s largest biclusers with elements 1 in a 0-1 matrix. We present a new type of algorithms-BBSM, which is based on the property of self-multiplication of matrix. The BBSM algorithm can obtain all the biclusters. Furthermore, it can obtain the s largest biclusters in size.

The paper is organized as follows: Section 2 describes the important theorem of self-multiplication of matrix. In Section 3, we propose the BBSM algorithm for non-overlapping case and overlapping case separately. Section 4 gives the conclusion.

2 Self-multiplication of Matrix

Given a 0-1 matrix $A = (a_{ij})_{m\times n}$, the matrix C is obtained by the self-multiplication of matrix A, that is $C = AA^TA$. Before we give the property of matrix C, some concepts were first defined as follows:

Definition 1. Submatrix $A(I,J)$**:** Given a matrix A, I is a positive integer set which belongs to the row index set of A, J is a positive integer set which belongs to the column index set of A, then $A(I,J)$ is a submatrix whose row index set is I and column index set is J. In particular, if J=$\{j\}$, then $A(I,\{j\})$ is written as $A(I,j)$ in short.

Definition 2. Bicluster: Given a 0-1 matrix A and its all-1 submatrix $A(I,J)$, $A(I,J)$ is called a bicluster if and only if there is not exist another all-1 submatrix $A(I',J')$ which is different from $A(I,J)$ satisfying $I \subseteq I'$and$J \subseteq J'$.

Definition 3. Overlaping: The biclusters of A are called overlapping if there exist an element a_{ij} which belongs to two or more biclusters, otherwise called non-overlapping. The element a_{ij} is called the overlapping element, all the overlapping elements are called the overlapping part of those biclusters.

Definition 4. Size of submatrix: The number of elements in submatrix $A(I,J)$ is called the size of submatrix $A(I,J)$ and noted as $SIZE(A(I,J))$. In particular, if the submatrix $A(I,J)$ is a bicluster, then $SIZE(A(I,J))$ is the size of bicluster.

Definition 5. Corresponding submatrix: For two matrixes A and C, if $G = A(I,J)$, $G' = C(I',J')$, then G is called the corresponding submatrix of G' if and only if $I = I', J = J'$, and vice versa.

Now we give out an important property of 0-1 matrix.

Theorem 1. Suppose $A = (a_{ij})_{m\times n}$ is a 0-1 matrix with m rows and n columns, $C = (c_{ij})_{m,n} = AA^TA$. If some $a_{ij} = 1$, then the value of c_{ij} equals to the sum of size of all the biclusters which contain the element a_{ij}, and the overlapping part is computed for only one time.

Proof Let $B = (b_{ij})_{m\times m} = AA^T$, then $b_{ij} = \sum_{k=1}^{n} a_{ik}a_{jk}$, $i, j \in \{1,\ldots,m\}$, and $C =$

$(c_{ij})_{m\times n} = AA^TA = BA$, for $i \in \{1,\ldots,m\}, j \in \{1,\ldots,n\}$,

$$\begin{aligned} c_{ij} = \sum_{k=1}^{m} b_{ik}a_{kj} &= b_{i1}a_{1j} + b_{i2}a_{2j} + \cdots + b_{im}a_{mj} \\ &= a_{1j}(a_{i1}a_{11} + a_{i2}a_{12} + a_{i3}a_{13} + \cdots + a_{in}a_{1n}) \\ &+ a_{2j}(a_{i1}a_{21} + a_{i2}a_{22} + a_{i3}a_{23} + \cdots + a_{in}a_{2n}) \\ &+ \cdots\cdots \\ &+ a_{mj}(a_{i1}a_{m1} + a_{i2}a_{m2} + a_{i3}a_{m3} + \cdots + a_{in}a_{mn}). \end{aligned} \tag{1}$$

If $a_{ij} = 1$, suppose the row indexes of the biclusters containing a_{ij} are $i_1, i_2, \cdots, i_p$, and there are q_1 elements on the i_1th row of which column indexes are noted by $j_{1,1}, j_{1,2}, \cdots, j_{1,q_1}$ respectively; there are q_2 elements on the i_2th row of which column indexes are $j_{2,1}, j_{2,2}, \cdots, j_{2,q_2}$ respectively; $\cdots$, there are q_p elements on the i_pth row of which column indexes are $j_{p,1}, j_{p,2}, \cdots, j_{p,q_p}$ respectively. Therefore, all the elements are listed as follows:

$$a_{i_1,j_{1,1}}, a_{i_1,j_{1,2}}, a_{i_1,j_{1,3}}, \cdots, a_{i_1,j_{1,q_1}}; \tag{2}$$

$$a_{i_2,j_{2,1}}, a_{i_2,j_{2,2}}, a_{i_2,j_{2,3}}, \cdots, a_{i_2,j_{2,q_2}}; \tag{3}$$

$$\cdots \quad \cdots$$

$$a_{i_p,j_{p,1}}, a_{i_p,j_{p,2}}, a_{i_p,j_{p,3}}, \cdots, a_{i_p,j_{p,qp}}. \tag{4}$$

The next fact is obvious:

1) The element 1 of A's jth column must belong to a bicluster that contains the element a_{ij}, which means

$$a_{sj} = \begin{cases} 1 \text{ if } s \in \{i_1, i_2, \ldots, i_p\}; \\ 0 \text{ else.} \end{cases} \tag{5}$$

2) For the elements on the ith row and i_kth ($k \in \{1,2,\ldots, p\}$) row, if $r \in \{j_{k,1}, j_{k,2}, \ldots, j_{k,q_k}\}$, we must have $a_{i_k,r}$=$a_{i,r}$=1, If $r \notin \{j_{k,1}, j_{k,2}, \ldots, j_{k,q_k}\}$, we must have $a_{i_k,r}a_{i,r} = 0$, so

$$a_{i,r}a_{i_k,r} = \begin{cases} 1 \text{ if } r \in \{j_{k,1}, j_{k,2}, \ldots, j_{k,q_k}\}; \\ 0 \text{ else.} \end{cases} \tag{6}$$

Now, according to (2) $\sim$ (4), the number of elements of all the biclusters containing the element a_{ij} is equal to $q_1 + q_2 + \cdots + q_p$, so the sum of the size of all the biclusters containing $a_{i,j}$ is equal to $q_1 + q_2 + \cdots + q_p$, and each element in (2) $\sim$ (4) is numbered for only one time.

On the other hand, according to (5), (1) can be rewritten as

$$\begin{aligned} c_{ij} &= a_{i_1,j}(a_{i1}a_{i_1,1}+a_{i2}a_{i_1,2}+a_{i3}a_{i_1,3}+\cdots+a_{in}a_{i_1,n}) \\ &+ a_{i_2,j}(a_{i1}a_{i_2,1}+a_{i2}a_{i_2,2}+a_{i3}a_{i_2,3}+\cdots+a_{in}a_{i_2,n}) \\ &+ \cdots\cdots \\ &+ a_{i_p,j}(a_{i1}a_{i_p,1}+a_{i2}a_{i_p,2}+a_{i3}a_{i_p,3}+\cdots+a_{in}a_{i_p,n}) \\ &= (a_{i1}a_{i_1,1}+a_{i2}a_{i_1,2}+a_{i3}a_{i_1,3}+\cdots+a_{in}a_{i_1,n}) \\ &+ (a_{i1}a_{i_2,1}+a_{i2}a_{i_2,2}+a_{i3}a_{i_2,3}+\cdots+a_{in}a_{i_2,n}) \\ &+ \cdots\cdots \\ &+ (a_{i1}a_{i_p,1}+a_{i2}a_{i_p,2}+a_{i3}a_{i_p,3}+\cdots+a_{in}a_{i_p,n}), \end{aligned} \tag{7}$$

then from (7) we can get $c_{ij} = q_1 + q_2 + \cdots + q_p$ by (6), which means the value of c_{ij} equals to the sum of size of all the biclusters which contain the element a_{ij}, and the overlapping part is computed for only one time, and the proof is completed. □

Let us look at the following example to verify theorem 1. Suppose $\tilde{A}$ and $\hat{A}$ are 0-1 matrices,

$$\tilde{A} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 \end{pmatrix}, \hat{A} = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}, \tag{8}$$

after the self-multiplication of $\tilde{A}$ and $\hat{A}$, we get

$$\tilde{C} = \tilde{A}\tilde{A}^T\tilde{A} = \begin{pmatrix} 6 & 6 & 6 & 0 & 0 \\ 0 & 0 & 0 & 2 & 2 \\ 6 & 6 & 6 & 0 & 0 \end{pmatrix}, \hat{C} = \hat{A}\hat{A}^T\hat{A} = \begin{pmatrix} 6 & 3 & 8 & 3 & 8 & 3 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 2 & 6 & 2 & 6 & 2 \\ 9 & 6 & 11 & 6 & 11 & 6 \end{pmatrix}. \tag{9}$$

We can see that the biclusters in matrix $\tilde{A}$ are non-overlapping, then the value of element in matrix $\tilde{C}$ equals to the size of the bicluster which contains the corresponding element in $\tilde{A}$. For example, the value 6 on the first row and first column in $\tilde{C}$ is equal to the size of bicluster $\tilde{A}(\{1,3\},\{1,2,3\})$. However, the biclusters in matrix $\hat{A}$ are overlapping, for example, the element 1 on the forth row and first column in matrix $\hat{A}$ is contained in bicluster $\hat{A}(\{1,4\},\{1,3,5\})$ with size 6 and bicluster $\hat{A}(\{4\},\{1,2,3,4,5,6\})$ with size 6, the overlapping part is a submarix $\hat{A}(\{4\},\{1,3,5\})$ with size 3. So the value 9 on the forth row and first column in matrix $\hat{C}$ satisfying $6+6-3=9$.

3 Biclustering Algorithm Based on Self-multiplication of Matrix(BBSM)

Based on the above analysis, we will construct an algorithm for 0-1 matrix in order to find the s largest biclusters in size. Two cases of 0-1 matrix will be considered separately, non-overlapping case and overlapping case.

3.1 BBSM for non-overlapping case

For the non-overlapping case like $\tilde{A}$ in (8), BBSM for finding the s largest bilcusters is constructed as follows.

Non-overlapping biclustering algorithm(BBSM1)

1. Given a 0-1 matrix A whose biclusters are non-overlapping, and a positive integer s;

2. Set $\mathtt{BC} = \varnothing$, compute $C' = (c_{ij})_{m\times n} = AA^TA$. Construct matrix $C^0 = (c^0_{ij})_{m\times n}$ from C

$$c^0_{ij} = \begin{cases} c_{ij}, & \text{if} \quad a_{ij} = 1; \\ 0, & \text{otherwise}; \end{cases} \tag{10}$$

3.Iteration: Set $k = 0$,

(1) Find the largest value m_k in C^k and note its row index and column index as p_k, q_k;

(2) Compute $J_k = \{j : C^k(p_k, j) = m_k\}$;

(3) Compute $I_k = \{j : C^k(j, q_k) = m_k\}$;

(4) Construct bicluster $A(I_k, J_k)$ and set $\mathtt{BC} = \mathtt{BC} \cup A(I_k, J_k)$;

(5) Reset $C^{k+1} = C^k - F^k$, where $F^k = (f^k_{ij})_{m\times n}$ andčž

$$f^k_{ij} = \begin{cases} c_{ij}, & \text{if} \quad i \in I_k \text{ and } j \in J_k; \\ 0, & \text{otherwise}; \end{cases} \tag{11}$$

(6) If $k = s$, goto step 4; otherwise, $k = k + 1$;

4.Output: Bicluster set BC. □

Theorem 2 For a given 0-1 matrix A and integer s, if the biclusters in A are non-overlapping, the output set BC in algorithm BBSM1 contains the s largest biclusters of A, and the bicluster sequence is outputted along large size to small size.

Proof According to theorem 1, a bicluser in A corresponds to a submatrix in C which has same elements and the value of each element equals to the size of the corresponding bicluster. On the other side, a submatrix in matrix C of which the elements are all the same and equal to the size of itself corresponds to a bicluster in A with the same size. Therefore, finding the bicluster in A is equivalent to finding the corresponding submatrix in C whose elements are all the same. It is to say that, I_k and J_k are respectively the row and column index set of C's submatrix whose elements are all equal to its size. Therefore $A(I_k, J_k)$ is a bicluster, implying that the Algorithm BBSM1 is just finding such submatrix in C, and can produced the s largest biclusters along the order of large size to small size. □

Like Bimax algorithm[8], BBSM1 algorithm can also obtain all the biclusters from the given 0-1 matrix by changing (6) in step 3 to "if $C^k = 0$, goto step 4; otherwise $k = k + 1$". However, Bimax algorithm can not produce the s largest biclusters directly except after finding all the biclusters.

3.2 BBSM for overlapping case

For the second case where the biclusters are overlapping in a given 0-1 matrix like $\hat{A}$ in (8), we will propose a more general algorithm-BBSM2. BBSM1 algorithm is a special case of BBSM2. First, we give the definition of generation matrix.

Definition 6. Generation matrix For a given matrix A, suppose $A(I_1,j_1),A(I_2,j_2),\cdots,A(I_t,j_t)$ are t biclusters of A. For $i\in\{1,2,\cdots,t\}$, let $E(j_i)=\{j_k : I_k\supseteq I_i, k\in\{1,2,\cdots,t\}\}$, then the submatrix $A(I_i,E(j_i))$ is called the generation matrix of $A(I_i,j_i)$ based on $\{A(I_1,j_1),A(I_2,j_2),\ldots,A(I_t,j_t)\}$. The set $\{A(I_1,E(j_1)),A(I_2,E(j_2)),\cdots,A(I_t,E(j_t))\}$ is called the generation matrix set of $\{A(I_1,j_1),A(I_2,j_2),\ldots,A(I_t,j_t)\}$.

For arbitrary 0-1 matrix A, we can get its self-multiplication. According to theorem 1, if $a_{ij}=1$ and corresponding c_{ij} is smaller, the size of A's bicluster containing a_{ij} must be smaller. So, in order to find the s largest biclusters in A, starting from the large element in C will be an appropriate choice. Therefor the algorithm is constructed as follows.

Overlapping biclustering algorithm(BBSM2)

1. Given a 0-1 matrix A, and a positive integer s;

2. Set $\mathtt{GM}=\varnothing$, compute $C=(c_{ij})_{m\times n}=AA^TA$, construct matrix $C'=(c'_{ij})_{m\times n}$ from C

$$c'_{ij}=\begin{cases}c_{ij}, & \text{if } a_{ij}=1;\\ 0, & \text{otherwise;}\end{cases} \tag{12}$$

and set $D_0=C'$;

3.Initialization: Set $k=0$,

(1) Find the largest element m_k in D^k and note its row index and column index as p_k,q_k;

(2) Compute the column index set of elements in C' which are not less than m_k on p_kth row: $J_k=\{j : c'_{p_k,j}\geq m_k\}$;

(3) For $j\in J_k$, compute the row index set of elements which are not less than m_k on q_kth column: $H_k^j=\{r : c'_{r,j}\geq m_k\}$;

(4) Compute the generation matrix set of $\{A(H_k^j,j)\}_{j\in J_k}$: $\{A(H_k^j,E(j)) : j\in J_k\}$;

(5) For $j\in J_k$: if there exist $A(U,V)\in\mathtt{GM}$ where $H_k^j\supseteq U, E\{j\}\supseteq V$, set $A(U,V)=A(H_k^j,E\{j\})$; otherwise set $\mathtt{GM}=\mathtt{GM}\cup A(H_k^j,E\{j\})$;

(6) Compute $F^k=(f_{ij}^k)_{m\times n}$, where

$$f_{ij}^k=\begin{cases}c'_{ij}, & \text{if } a_{ij} \text{ belongs to a bicluster of } \mathtt{GM};\\ 0, & \text{otherwise;}\end{cases} \tag{13}$$

(7) Reset $D_{i+1}=C'-F^k$;

(8) If $|\mathtt{GM}|\geq s$, and the size of the smallest submatrix in $\mathtt{GM}$ is larger than the largest element in D^k, goto step 4; otherwise, $k=k+1$;

4. Output: output s largest biclusters in $\mathtt{GM}$. □

Now we will prove the output of algorithm BBSM2 are the s largest biclusters of A.

Theorem 3 Given a 0-1 matrix A, the output of BBSM2 algorithm is the set of the s largest biclusters of A.

Proof: Suppose at the end of algorithm BBSM2, the smallest size of submatrix in $\mathtt{GM}$ is v. We only need to prove that any bicluster with size not less than v belongs to $\mathtt{GM}$.

Suppose $A(H_k^j,E_k(j))$ is a bicluster satisfying $SIZE(A(H_k^j,E_k(j)))\geq v$, and it corresponds to a submatrix $C'(H_k^j,E_k(j))$ in C'. We can see in algorithm BBSM2 that

$C'(H_k^j, E_k(j))$ must can be derived from m_k for some k, where m_k is not only the largest element of D^k, but also the smallest element of $C'(H_k^j, E_k(j))$. So there must have $A(H_k^j, E_k(j)) \in \mathtt{GM}$. □

Like Bimax algorithm[8], BBSM2 algorithm can also get all the biclusters by changing the stop criterion as "$D^k = 0$".

4 Conclusion

In this paper, we propose a novel biclustering algorithm based on self-multiplication for 0-1 gene expression data matrix . For two different cases, we constructed two different algorithms, algorithm BBSM1 is for the non-overlapping case which can get the k most largest biclusters directly and also can get all biclusters, algorithm BBSM2 is for overlapping case which in most cases can get the k most largest biclusters without need getting all biclusters.

In practice, the size of a bicluster is a more important evaluation score. A bicluster whose size is large and 0 element is little is also a high quality bicluster. In the future work, we will try to propose a new algorithm to find these type of biclusters .

References

[1] J.A. Hartigan, Direct Clustering of a Data Matrix, J. Am. Stat. Assoc. (JASA), vol.67, no.337, 1972, pp.123∼129.

[2] S.C. Madeira, A.L. Oliveira, Biclustering algorithms for biological data analysis: a survey, IEEE Transactions on Computational Biology and Bioinformatics, 2004, pp.24∼45.

[3] Y. Cheng, G.M. Church, Biclustering of expression data, In Proceedings of the 8th International Conference on Intelligent Systems for Molecular Biology (ISMBąŕ00), 2000, pp.93∼103.

[4] G. Getz, E. Levine and E. Domany, Coupled Two-Way Clustering Analysis of Gene Microarray Data, Proc. Natl. Acad. Sci. U.S.A., vol.97, 2000, pp.12079∼12084.

[5] C. Tang, L. Zhang, I. Ahang and M. Ramanathan, Interrelated Two-Way Clustering: An Unsupervised Approach for Gene Expression Data Analysis, In Proc. Second IEEE Int'l Symp. Bioinformatics and Bioeng, 2001, pp.41∼48.

[6] L. Lazzeroni and A. Owen, Plaid Models for Gene Expression Data, Technical Report, Stanford University, 2000.

[7] A. Tanay, R. Sharan, R. Shamir, Discovering statistically significant biclusters in gene expression data, Bioinformatics 18 (Suppl. 1),2002, pp.136∼144.

[8] A. Prelic, S. Bleuler, P. Zimmermann, A. Wille, P. Buhlmann,W. Gruissem, L. Hennig, L. Thiele, and E. Zitzler, A systematic comparison and evaluation of biclustering methods for gene expression data, Bioinformatics, Vol.22, no.9, 2006, pp.1122∼1129.

[9] Z. Zhang, C. Ding, T. Li, Xiangsun Zhang, Binary Matrix Factorization with Applications, Data Mining, 2007, ICDM 2007.