Computer Science and Information Systems 14(3):643–660 https://doi.org/10.2298/CSIS170301021Y

A Weighted Mutual Information Biclustering Algorithm for Gene Expression Data

Yidong Li[1], Wenhua Liu[1], Yankun Jia[1], and Hairong Dong[2]

[1] School of Computer and Information Technology, Beijing Jiaotong University
[2] State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University
{ydli, 16112077, 15120402, hrdong}@bjtu.edu.cn

Abstract. Microarrays are one of the latest breakthroughs in experimental molecular biology, which have already provided huge amount of high dimensional genetic data. Traditional clustering methods are difficult to deal with this high dimensional data, whose a subset of genes are co-regulated under a subset of conditions. Biclustering algorithms are introduced to discover local characteristics of gene expression data. In this paper, we present a novel biclustering algorithm, which called Weighted Mutual Information Biclustering algorithm (WMIB) to discover this local characteristics of gene expression data. In our algorithm, we use the weighted mutual information as new similarity measure which can be simultaneously detect complex linear and nonlinear relationships between genes, and our algorithm proposes a new objective function to update weights of each bicluster, which can simultaneously select the conditions set of each bicluster using some rules. We have evaluated our algorithm on yeast gene expression data, the experimental results show that our algorithm can generate larger biclusters with lower mean square residues simultaneously.

Keywords: biclustering, mutual information, gene expression data.

1. Introduction

With the rapid development of bioscience and computer science, Bioinformatics became a newly forming discipline combining bioscience and computer science. With the rise of bioinformatics a series of high-throughput detection techniques have been developed rapidly, such as cDNA microarray experiments and the gene chip technology, which have produced huge amounts of high dimensional gene expression data, one example as shown in Figure 1. Those technologies use the same principle, which uses each pairing of complementary characteristics of the four nucleotides, two pairs of single nucleotide chains which are complementary to each other are formed in a double chain, this process is called hybridization.

Gene is the basic unit of genetic information in organisms, gene expression is using the genetic information stored in DNA, through transcription or translation to perform biological functions. By measuring those expression patterns of genes under different conditions, different development stages or different tissues, we can establish the database of gene expression matrix, then we can analyze and summarize gene expression data better. The analysis of gene expression data helps to explain gene expression mechanism and understand the function of genes, find how did genome be influenced by various factors,

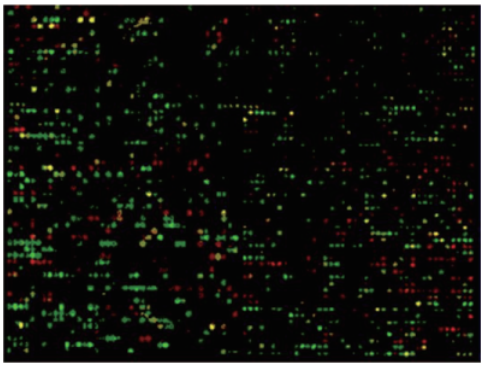

Fig. 1. Gene expression values from cDNA microarray experiments can be represented as heat maps to visualize the result of data analysis

and understand genetic network, then provide information of diseases pathogenesis for further theoretical and applied research at last.

Gene expression data are usually represented by one data matrix, each row is represented as a gene, each column is represented as a condition or sample, as we can see in Figure 2. How to search for potential biological information from high dimensional genetic data becomes an urgent problem ,which need to be solved in data mining technology.

	c1	c2	c3	c4	c5
g1	1	2	3	4	5
g2	2	4	6	8	10
g3	3	6	9	24	25
g4	4	8	12	34	35
g5	5	10	15	4	5
g6	7	12	18	12	15

	c1	c2	c3	c4	c5
g1	1	2	3	4	5
g2	2	4	6	8	10
g3	3	6	9	24	25
g4	4	8	12	34	35
g5	5	10	15	4	5
g6	6	12	18	12	15

Fig. 2. Some example of clustering algorithm. Left: the result of traditional clustering algorithm. Right: result of biclustering algorithm

At present, main methods of analyzing gene expression data are clustering analysis methods. Through clustering analysis, those genes with similar expression patterns are clustered into one category. On this idea, we search for genes with similar patterns, analyze the function of genes, and analyze the transcriptional regulation of genes. The traditional clustering methods can be divided into partitioning method, hierarchical method, density-based method, grid-based method and model-based method. Traditional clustering methods have made some achievements in the analysis of genetic data, but they can only cluster the gene data using global information, the clustering results either contain all rows of data matrix, or all columns of data matrix, but there are usually existing some lo-

cal correlation between genes and conditions for gene expression data. For example, some genes show similar patterns of expression sometimes only under certain subset of conditions, and one gene under different subset of conditions may show different expression patterns. Therefore, traditional clustering algorithms are not very ideal for the analysis of gene expression data in many cases.

The biclustering algorithm as a new method which is introduced to clustering gene expression data from gene dimension and condition dimension simultaneously, which overcomes the limitation of traditional clustering methods. The concept of biclustering algorithm was firstly introduced by Cheng and Church [5] and was applied to the analysis of gene expression data. After that there are emerged a lot of excellent improved biclustering algorithm, those algorithm have achieved considerable results in the biological data mining, such as FLOC algorithm [23], Evolutionary Algorithm [4] and MIB algorithm [8]. The biclustering algorithm is repeatedly clustering from the gene dimension and the condition dimension, and using this local correlation information between genes and conditions to improve the accuracy of clustering results.

Currently most of biclustering algorithms use ordinary Euclidean distance as the similarity measurement between genes, but Euclidean distance can only detect certain linear relationship of gene expression data, and there are existing some complicated nonlinear similarity relationship between biological data. The concept of mutual information comes from information theory, it's commonly used to represent the relationship between information. And when calculated the similarity between genes, different conditions have different effect on the expression pattern of gene information, therefore we set different weight values for different conditions under different biclusters, which used to measure genes' similarity. In this paper our proposed a weighted mutual information biclustering (WMIB) algorithm used the weighted mutual information as the similarity measure of genetic data. Through a series of experiments, we show that our proposed WMIB algorithm has excellent performance, which can not only obtain different types of biclusters, but also ensure that those biclusters have lower mean square residues.

The reminder of this paper is organized as following. Section 2 briefly reviews existing biclustering algorithms in the context of gene expression data. Section 3 defines some theoretical concepts and notations used in our algorithm. Section 4 introduces the framework of our algorithm and details of our algorithm's implementation. Then we further compare with other biclustering algorithm and our experimental results is shown in Section 5. Finally Section 6 contains the conclusion and future work.

2. Related Work

Cheng and Church [5] firstly proposed the concept of biclustering algorithm called CC algorithm, CC algorithm used a greedy iterative searching method to find biclusters, through gradually add or remove rows or columns of genetic data which reduce the mean square residues of biclusters, which get better biclusters after iterations. But CC algorithm could not find overlapping biclusters, Yang et al [23] presented an FLOC algorithm, by calculating the gain function of each action to determine either add or delete one row or column from biclusters. Then some of evolutionary method was proposed, Sefan et al [4] proposed Evolutionary Algorithm (EA) framework which apply some intelligent optimization algorithms to optimize the biclustering result. Pontes presented Evo-Bexpa (Evolutionary

Biclustering based in Expression Patterns) is the first biclustering algorithm in which it is possible to particularize several biclusters features in terms of different objectives. Filipiak [6] proposed HEMBI using an Evolutionary Algorithm to split a data space into a restricted number of regions.

Wang [19]used exhaustive strategies to find biclusters of data, then Liu et al [11] improved algorithm. Tanay [17] proposed a bicluster algorithm called SAMBA that converts biclustering problem into a balanced bipartite graph search problem. Zhu [21] combined simulated annealing technique and particle swam algorithm, presented a simulated annealing particle swarm optimization algorithm. Swarup [14] presented CoBi which used a BiClust tree that needs single pass over the entire dataset to find a set of biologically relevant biclusters. Xu [22] presented an efficient exhaustive algorithm to search contiguous column coherent evolution biclusters in time-series data. Haifa [16] proposed EnumLat algorithm which is the construction of a new tree structure to represent adequately different biclusters discovered during the process of enumeration.

Zhang et al [25][24] proposed a DBF algorithm based on frequent pattern mining. Zhu [26][21] proposed a biclustering algorithm based on hierarchical clustering. Madeira [12] proposed an efficient biclustering algorithm for finding genes with similar patterns in time-series expression data. Rui [15] propose new biclustering algorithms to perform flexible, exhaustive and noise-tolerant biclustering based on sequential patterns (BicSPAM). BicSPAM is the first attempt to deal with order-preserving biclusters that allow for symmetries and that are robust to varying levels of noise. Wang [20] found an UniBic algorithm is to apply the longest common subsequence (LCS) framework to selected pairs of rows in an index matrix derived from an input data matrix to locate a seed for each bicluster to be identified. And some security algorithm [9][10] was proposed for data analysis.

The mean square residue [5] and some valuations criterions that based on the residue are widely used in biclustering algorithms. Teng [18] proposed an average correlation value(ACV) to evaluate the homogeneity of biclusters, which is more reasonable with the co-expression of genes and conditions in biological data. Wassim [2] proposed BiMine algorithm used Average Spearman's rcho (ASR) as evaluation function, later Wassim [3] proposed another evaluation function called Average Correspondence Similarity Index (ACSI) to assess the coherence of given biclusters. Gupta [8] used mutual information to detecting non-linear relationship between genetic data. Aggarwal [1] presented a novel ensemble technique for biclustering solutions using mutual information.

3. Preliminaries

In this section, we will provide notations and preliminaries related to our work.

Gene expression data is usually represented by a data matrix, one row represents one gene and one column represents one condition (or one sample under specific tissues and development stage), each value of matrix represents the expression level of one gene under a specific condition, one row is often referred as a gene expression profile. Analysing the gene expression matrix is used to extract potential biological information. Given the gene expression data, let $G = \{g_1, g_2, g_3, \cdots, g_N\}$ be represented as the set of genes and $C = \{c_1, c_2, c_3, \cdots, c_M\}$ be represented as the set of conditions, where N and M are the number of genes and the number of conditions respectively. Then the expression data can

be represented as a matrix $D_{N\times M}$, where each element value d_{ij} in matrix corresponds to the logarithmic of the relative abundance of the mRNA of one gene g_i under one specific condition c_j.

Definition 1 *Given the gene matrix $\mathbb{G} \times \mathbb{C}$, a bicluster can be defined as a pair (I, J), where I⊂ G be subset of genes G and J ⊂ C be subsets of conditions C.*

A bicluster essentially corresponds to a submatrix in which a subset of genes exhibits consistent patterns under a subset of conditions. For a given gene data expression dataset $\mathbb{G} \times \mathbb{C}$, biclustering algorithm finds a set of submatrixes $(I_1, J_1), \cdots, (I_k, J_k)$ of the matrix $\mathbb{G} \times \mathbb{C}$, $|I_k|$ is the number of specified genes in the k-th bicluster(I,J). A set of biclusters can also be represented as $B = \{B_1, B_2, B_3, \cdots, B_k\}$, where k is the number of biclusters, and B_i is represented as i-th bicluster.

Definition 2 *Given the bicluster (I, J), the volume of a bicluster V_{IJ} is defined as the number of elements d_{ij} in bicluster (I, J) where $i \in I$ and $j \in J$.*

Given the bicluster (I, J), we can have $V_{IJ} = |I| \times |J|$, where $|I|$ and $|J|$ are the number of genes and the number of conditions respectively. Figure 3 shows a gene expression matrix with eight genes and six conditions, for one bicluster, we pick $I = \{g_2, g_3, g_5, g_7\}$ as genes set and $J = \{c_1, c_3, c_5\}$ as conditions set, then the volume of this bicluster is 12.

Definition 3 *Give the bicluster (I, J), d_{ij} is one element value of the bicluster, where $i \in I$ and $j \in J$. The base of one gene d_{iJ} is defined as the average values of gene g_i under certain specified conditions J, it can be calculated by $d_{iJ} = \frac{1}{|J|}\sum_{j\in J} d_{ij}$.*

Similarly, the base of a condition c_{Ij} is defined as the average values of c_j under the specified genes I, it calculated by $d_{Ij} = \frac{1}{|I|}\sum_{i\in I} d_{ij}$. And the base of bicluster d_{ij} can be defined as the average values of each element in bicluster (I, J), calculated by $d_{IJ} = \frac{1}{|I||J|}\sum_{i\in I, j\in J} d_{ij}$.

The base of gene d_{Ij} and the base of condition d_{iJ} may reflect the consistency of information in the corresponding genes or conditions.

Definition 4 *The mean square residue of a bicluster (I,J) can be represented as $r_{IJ} = \frac{1}{V_{IJ}}\sum_{i\in I, j\in J} r_{ij}^2$, where $r_{ij} = d_{ij} - d_{Ij} - d_{iJ} + d_{IJ}$ is the residue of each element d_{ij} in bicluster (I, J).*

The mean square residue of a bicluster can be regard as an important criterion to detect the consistency of the bicluster, the lower the residue, the stronger consistency exhibited by the bicluster. The mean square residue of biclusters are commonly used to evaluate the overall quality of a bicluster.

For example, as show in Figure 3 we have a gene expression matrix with eight genes and six conditions, we pick genes set of $I = \{g_2, g_3, g_5, g_7\}$ and conditions set of $J = \{c_1, c_3, c_5\}$ as one perfect bicluster (I, J). The bases of genes are $d_{2,J} = 40$, $d_{3,J} = 40$, $d_{5,J} = 40$, $d_{7,J} = 40$, the bases of conditions are $d_{I,1} = 50$,$d_{I,2} = 40$, $d_{I,3} = 30$, then the base of bicluster is $d_{I,J} = 40$, so the residue of $d_{1,1}$ obtained by $r_{1,1} = d_{1,1} - d_{I,1} - d_{1,J} + d_{IJ} = 0$, similarly we calculated the residues of other elements in bicluster, finally we can obtain the mean square residue of bicluster (I, J) is 0.

	c1	c2	c3	c4	c5	c6
g1	33	40	45	50	40	70
g2	50	47	40	80	30	33
g3	50	41	40	50	30	44
g4	55	47	80	55	80	70
g5	50	40	40	50	30	55
g6	66	47	45	55	36	44
g7	50	80	40	70	30	46
g8	47	55	45	50	55	44

Fig. 3. The example of a bicluster: all grey color cells represent one bicluster obtained from the dataset, the mean square residue of this bicluster is zero

4. Algorithm Implementation

In this section, we will introduce our proposed WMIB algorithm in detail, which can efficiently and accurately discovered biclusters from gene expression data.

At the beginning of WMIB algorithm, In section 4.1, we wiil introduce the weighted mutual information as new similarity measure between genes, then we will construct a set of seed genes from the dataset as the initial biclusters, which has the least similarity between each seed genes of initial biclusters. In section 4.2, we will use one possibility function to calculate the possibility between each gene from entire data with seed genes of initial biclusters, then we divide genes into corresponding biclusters according those possibility of genes which is greater than the given threshold. Then in section 4.3 we constructed a novel objective function, and by optimizing this objective function we could update the weights of each condition in biclusters, then we remove the conditions set whose have smaller weights in each bicluster. After that we obtained some biclusters according to the updated weights of conditions. Then we can recalculate the new seed genes of each bicluster, and using new seed gene in each bicluster we redivided each gene into biclusters according seed gene in each bicluster and the similarity threshold as show in section 4.4. After completing those steps, in section 4.5 we optimize the obtained biclusters using some novel rules. Finally we will conclude the process of our WMIB algorithm as Figure 4.

4.1. The Construction of Seed Gene Sets

At the beginning of WMIB algorithm, we should construct the seed genes set. At first we initialize the biclusters set B and seed genes set S are empty set, we randomly selected one gene from dataset as the seed gene of the first bicluster, and we add the seed gene into set B and set S, then we calculated the similarity between this seed gene with remaining genes in dataset. Firstly we introduce the measurement used to calculate similarity between genes in our algorithm.

Result of any biclustering algorithm depends on the choice of the similarity measure used. Different similarity measures on the same expression data could produce different results.The mainly similarity measurement used to biclustering gene expression data is Euclidean distance, Mahalanobis distance, and cosine similarity function, but these functions can only measure the linear relationship between genes. However, in gene expression data, there may not only exist a simple linear dependencies between genes, but also exist

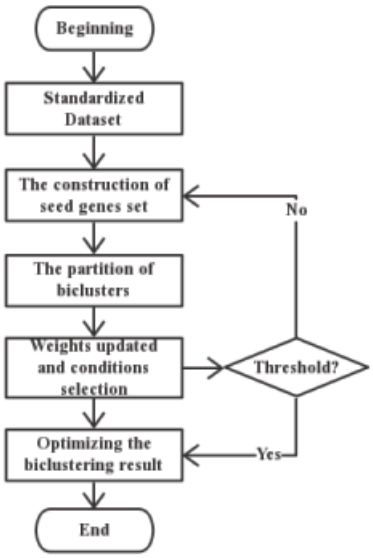

Fig. 4. The process diagram of WMIB algorithm: we briefly present several steps of our algorithm

a complex nonlinear relationship between objects, so these similarity function may not be satisfactory to measure the similarity between genes.

Mutual information is a concept introduced by information theory, which is used to represent the relationship between information. Mutual information had been widely used in many traditional clustering methods, which was proved to be able to detect nonlinear relationships between data, and it is not sensitive to noise data, so in this algorithm we use the mutual information as similarity measure to complex relationship between genes.

The concept of information entropy is a measure of the information contained in the data, the information entropy of a discrete variable X can be defined as follows:

$$H(X) = -\sum p_i log p_i \tag{1}$$

where p_i is the probability of i-th state occurred in X Then the concept of the joint information entropy of two discrete variable X and Y can be defined as:

$$H(X,Y) = -\sum p(x,y) log p(x,y) \tag{2}$$

where $p(x,y)$ is the joint probability of discrete variable X and Y. The definition of mutual information can be defined using the concept of information entropy, the formula of mutual information between two discrete variable X and Y is defined as:

$$M(X,Y) = H(X) + H(Y) - H(X,Y) \tag{3}$$

The calculation of mutual information usually relates to the probability of the random variables' marginal distribution and joint distribution. In most cases these distributions

are unknown, so those requires the estimation of probability density function through the prior knowledge and the statistics. Here we use the Gaussian density function which is commonly used to estimate the probability density distribution of data. For a random variable X, it's probability density estimation as:

$$\hat{p}(X) = \frac{1}{Nh\sqrt{2\pi}} \sum_{i=1}^{N} exp(\frac{-(x - x_i)^2}{2h^2}) \tag{4}$$

The mutual information $M(X, Y)$ is zero if X and Y are independent and it's value is high if they are highly dependent to each other. Supposed that the observations of two random variables X and Y are represented as $\{x_1, x_2, x_3, ..., x_n\}$ and $\{y_1, y_2, y_3, ..., y_n\}$. After brought the probability estimation function of variables into the function of mutual information, the mutual information between X and Y can be represented as:

$$M(X, Y) = \frac{1}{N} \sum_{i=1}^{N} log \frac{\hat{p}(x_i, y_i)}{\hat{p}(x_i) * \hat{p}(y_i)} \tag{5}$$

The weighted measurement is a very common methods in statistics, the weight of a certain index is the relative importance of the index in the overall evaluation.

From the aspect of gene expression data, different conditions shows different influence on expression patterns of gene information in one bicluster, and for different biclusters same conditions may show different effects. In this paper, we apply the weighted thought to the calculation of mutual information to evaluate different effects of conditions.Then we propose a new similarity measurement function, called the weighted mutual information which is represented as:

$$M(X, Y) = \frac{1}{N} \sum_{i=1}^{N} w_i * log \frac{\hat{p}(x_i, y_i)}{\hat{p}(x_i)\hat{p}(y_i)} \tag{6}$$

Most of biclustering algorithms obtain results by choosing the seed genes randomly, but this will lead to the instability of the algorithm, and it can not guarantee that the algorithm will be able to obtain biclusters with consistent volatility. In this paper we use a novel way to initialize the seed genes set.

We start with a random seed gene, then we use the weighted mutual information between the seed and remaining genes to choose the new seed. The next choice of the seed can be a gene whose has the least similarity with the previous seed, and we add this seed into biclusters set B and seed genes set S. Then we obtain the seed genes of each bicluster iteratively.

The next choice of the new seed for next bicluster as follows:

$$\underset{g_j \notin S}{min} \{ \sum_{g_i \in S} M(g_i, g_j, C) \} \tag{7}$$

where g_i and g_j represent i-th and j-th gene in dataset respectively.

Using this initialization method, our algorithm select the gene as the seed of next bicluster whose has the least similarity with previous seeds, which can also avoid the obtained biclusters having high repetition rate. After that we can obtain the initialized biclusters set and the initialized seed genes set. But each bicluster only contains the seed gene, and those seed genes includes all conditions.

4.2. The Fuzzy Partition of Biclusters

After constructed the seed genes set, we partition the set of genes which are most related to the input seed genes set. Firstly we introduce the membership function to calculate the probability of one gene belonging to one bicluster.

From the aspect of gene expression data, different expression patterns may exist in one gene, so one gene may belong to multiple biclusters. General non-fuzzy clustering method is difficult to detect this multiple partition especially when the experimental data is merged by a plurality of experimental data under different conditions.

In order to identify this multiple co-expressed relationship between genes, we use the thought of fuzzy clustering algorithm to complete the partition of biclusters. We first calculated the probability of reminding genes belonging to each bicluster, then the gene is divided into biclusters by the given possibility threshold. The probability function of the gene belonging to the bicluster is defined as:

$$P_{ik} = \sum_{i=1}^{|J_k|} P_{ijk} = \frac{1}{|J_k|} \sum_{i=1}^{|J_k|} M(g_i, g_k, Y_k) \tag{8}$$

where $|J_k|$ represents the number of conditions in bicluster B_k, and $M(g_i, g_k, Y_k)$ is represented as the weighted mutual information between gene g_i and g_k under conditions set Y_k. After we selected genes for each bicluster iteratively, we get the initial fuzzy partition of entire dataset. After such fuzzy partition of genetic data, each gene may simultaneously belong to several different biclusters, or may not belong to any of biclusters. Meanwhile, in order to avoid biclusters obtained by the fuzzy partition with a high overlapping rate, and ensure that those biclusters contain better biological information, we set an overlapping rate threshold. If the size of one biclusters is larger than this threshold then that bicluster will be pruning. After that each gene in the initial fuzzy partition of biclusters still contains all conditions, next we need to complete the selection of conditions.

4.3. Weights Updated and Conditions Selection

When we completed the initial fuzzy partition of entire dataset, we should update weights of conditions for all of biclusters, and complete the selection of condition sets. In our algorithm, we set one weight for each condition, which used to measure the effect of conditions in current bicluster, then through the weights updated to selected conditions of the corresponding bicluster, we remove conditions from biclusters whose weights lower than the weights threshold

Most of the biclustering algorithm use the iterative method to complete selection of condition sets, but the iterative process has high time complexity, and it maybe fill into local minima. Although some biclustering algorithm has proposed some objective function, which is used to obtained some good biclusters, but those objective function only consider to minimize the mean square residue of biclusters, and they are not including the influence of weights for finally biclusters. In this section we propose a new objective function, through optimizing this objective function we can be quickly update the weights of each condition, and we remove conditions from biclusters whose weights lower than the weights threshold to selected conditions for each bicluster.

The mean square residue (MSR) is the most widely criterion to measure the quality of biclustering algorithms, In order to improve accuracy of our biclustering algorithm we combine weights and MSR as the first part of the objective function, by minimizing mean average residue of each bicluster we can have better quality biclusters. The calculation formula of weighted MSR as following:

$$H(I_k, J_k) = \frac{1}{|I_k||J_k|}\sum_{i,j} w_{kj} * (d_{ij} - d_{I_k j} - d_{iJ_k} + d_{I_k J_k})^2 \tag{9}$$

where $d_{iJ_k}, d_{I_k j}, d_{I_k J_k}$ is represented as mean value of gene g_i, condition c_j and bicluster B_k respectively. Then we can obtain the first part formula of new objective function as:

$$R_k = min\sum_{i=1}^{p} w_{kj} \sum_{g_j \in B_k} (d_{ij} - d_{I_k j} - d_{iJ_k} + d_{I_k J_k})^2 \tag{10}$$

The objective function should guarantee that the size of biclusters, if the size of conditions of biclusters too small will lose much important biological information of genetic data. So we use the second part of objective function to control the size of conditions set. We use weights to approximate the size of biclusters, and use some constraint criteria as follows:

$$S_k = \sum_{j=1}^{p} \sqrt{w_{kj}} \quad where \sum_{j=1}^{p} w_{kj} = 1 \tag{11}$$

Since result of two part R_k and S_k are constraint in different ranges, it's inconvenience when optimize the objective function, so we use some constraint criteria to transform objective function, then we obtain the final formula of new objective function as:

$$\Gamma = min \sum_{k=1}^{K} \left\{ \sum_{j=1}^{p} w_{kj} * R_k - \frac{1}{p} \sum_{i=1}^{p} R_k * \frac{\sum_{j=1}^{p} \sqrt{w_{kj}} - 1}{\sqrt{p} - 1} \right\} \tag{12}$$

let w_{kj} be argument of the objective function Γ, we can see that the objective function is a convex function by definition, and we can directly optimize the objective function by the gradient method. Firstly we compute the gradient of the objective function, then we set the gradient to be zero, we can obtain the update formula of weights through transformation:

$$w_{kj} = \frac{1}{4p^2(\sqrt{p} - 1)} \left\{ \frac{1}{\sum\limits_{g_j \in B_k} H_{ik}} \sum_{i=1}^{p} \sum_{g_j \in B_k} R_{ik} \right\}^2 \tag{13}$$

where $H_{ik} = (d_{ij} - d_{I_k j} - d_{iJ_k} + d_{I_k J_k})^2$.

After we update the weights value of each bicluster, we normalize the updated weights using all weights of each bicluster. After that we compared normalized weights of each bicluster with the given weights threshold separately.

Then we select conditions set of each bicluster by setting weights of conditions are zero, whose weights are lower than the given weights threshold. After that we updated all weights of conditions and selected conditions set for each bicluster simultaneously.

$$w_{kj} = \begin{cases} w_{kj} & if\ w_{kj} \geq \gamma \\ 0 & else \end{cases} \tag{14}$$

4.4. The Re-partition of Biclusters

In the last section we have selected conditions set for each bicluster in accordance to the objective function, We need to recalculate the seed genes set of each bicluster, and then cluster data matrix in accordance to the updated set of seed genes and updated weights. Repartition of biclusters can processed by two steps:

Firstly, according to the updated weights and the conditions set, we need to recalculate the seed genes set of each bicluster, the seed gene s_k of k-th bicluster is calculated as :

$$s_k = \frac{1}{|I_k|}\sum_{i \in I_k} d_{ij} \tag{15}$$

Secondly, we use the newly seed genes set to recalculate the probability of genes belonging to each bicluster, then we add the genes into biclusters for whose probability are greater than the given possibility threshold.

After those two steps, we can obtain better biclulsters from gene expression data with more consistent volatility.

4.5. Optimizing Biclustering Results

By repartitioning of the biclusters, we can obtain the new set of biclusters from dataset, but that is not guaranteed that each bicluster has lower mean square residue (MSR). Most of the current biclustering alogorithm use mean square residue as the standard of evaluating biclustering results, but it is not very ideal for the assessment of certain structure of biclusters. In order to get more reasonable structure of biclusters, we use a new kind of weighted mean square residue to optimize biclusters obtained by repartition.

Firstly, we give the calculation formula of weighted mean square residue. $H(I_k, J_k)$ is represented as weighted mean square residue of the bicluster B_k.

$$H(I_k, J_k) = \frac{1}{|I_k||J_k|}\sum_{i,j} w_{kj} * (d_{ij} - wd_{I_k j} - wd_{iJ_k} + wd_{I_k J_k})^2 \tag{16}$$

WR_i and WC_j are represented as weighted mean square residue of the gene g_i and the condition c_j respectively

$$WR_i = \frac{1}{|J_k|}\sum_{j \in J_k} w_{kj} * (d_{ij} - wd_{I_k j} - wd_{iJ_k} + wd_{I_k J_k})^2 \tag{17}$$

$$WC_j = \frac{1}{|I_k|}\sum_{i \in I_k} w_{kj} * (d_{ij} - wd_{I_k j} - wd_{iJ_k} + wd_{I_k J_k})^2 \tag{18}$$

where wd_{iJ_k},$wd_{I_k j}$,$wd_{I_k J_k}$ are represented as weighted mean value of the gene g_i, the condition c_j and the bicluster B_k respectively. and their definition as:

$$label19 wd_{iJ_k} = \frac{1}{|J_k|}\sum_{j \in J_k} w_{kj} d_{ij}, \tag{19}$$

$$label20 wd_{I_k j} = \frac{1}{|I_k|}\sum_{i \in I_k} w_{kj} d_{ij}, \tag{20}$$

$$label21wd_{I_kJ_k} = \frac{1}{|I_k||J_k|}\sum_{i,j} w_{kj}d_{ij} \tag{21}$$

Here we use the weight mean square residue of each bicluster to optimize the biclustering results, which can be divided into two situations:

When one gene is contained in one bicluster, we first assume that this gene is deleted from the bicluster, and we calculate the weighted mean square residue of new bicluster. If the new weighted mean square residue is less than previous weighted mean square residue, then we remove this gene from this bicluster.

When one gene is not contained in one bicluster, we first assume that this gene is added into one bicluster, and we calculate the weighted mean square residue of new bicluster. If the new weighted mean square residue is less than previous weighted mean square residue, then we add this gene into this bicluster.

Cheng and Church [5] have proved that it would not increase the MSR of bicluster if we add one gene into one bicluster, which the MSR of gene is less than the MSR of bicluster. By optimizing the biclustering results obtained by previous section, our algorithm could find out larger biclusters with lower mean square residue, which indicated that we obtain better bicluster with consistent volatility.

4.6. The Processes and Complexity of WMIB Algorithm

In Algorithm 1 we show the main procedure of WMIB algorithm.

Algorithm 1 WMIB Algorithm

Input: Gene Data Matrix D, the number of biclusters α, the possibility threshold β and the weights threshold γ;

Output: Biclusters set B;

1: Initialize the biclusters set B and seed genes set S are empty, and select one gene randomly as the seed of the first bicluster;
2: Calculate weighted mutual information between each gene and seed genes, then select next seed gene according to the formula (7);
3: Calculate the probability P_{ik} of gene g_i belonging to bicluster B_k using the formula (8), then adding this gene g_i into bicluster B_k if its probability P_{ik} greater than the possibility threshold β;
4: Update the weights of comditions for each bicluster using the formula (13), then normalize weights;
5: Set the weights values of conditions to zero if weights less than weights threshold γ, which is used to select conditions set, then re-normalize weights of conditions;
6: Calculate new seed genes for each bicluster using the formula (15), then repartition of data matrix as step $3-4$;
7: Calculate weighted MSR of genes for each bicluster, and optimize the obtained biclusters according to weighted MSR;

In our proposed WMIB algorithm, the main complex process are the construction of seed genes set and the partition of biclusters. When constructed seed genes set, a set of seed gene and the initial biclusters are generated. The time complexity of computing weighted mutual information between genes is $O(M^2)$, where M is the number of

conditions, there are at most $\frac{k(k+1)}{2} \times N$ similarity computions between genes in this process, so the time complexity of construction of seed gene set is $O(k^2 \times N \times M^2)$, where k is the number of biclusters and N is the number of genes. In the second process, a set of biclusters are generated from genetic dataset, there are at most $c \times N$ similarity computions between genes in this process, so the time complexity of this process can represented as $O(k \times N \times M^2)$. Thus, the overall time complexity of our proposed algorithm is $O(k^2 \times N \times M^2)$.

5. Experimental Results

5.1. Dataset and Standardization

In this section, we use the yeast metabolic cycle expression datasets GDS2267 from Gene Expression Omnibus (GEO) database to evluate our proposed WMIB algorithm, the dataset contains 9335 genes and 36 conditions, and it has commonly used to evaluate the performance biclustering algorithm. In this dataset, genetic data is represented as a data matrix, each row is represented as one gene, and each column is represented as one condition. We construct biclusters to find submatrix which have consistency volatility.

In order to reduce the influence of the different attributes of the data or the variance of the data on the biclustering results, and we can compare accurately biclustering results obtained by other main algorithms, we firstly standardized the gene data, following the formula as:

$$g_{ij}^{'} = \frac{g_{ij} - \bar{g}_i}{S_i} \tag{22}$$

where $\bar{g}_i$ is represented as mean value of gene g_i, and S_i is represented as standard deviation of gene g_i.

5.2. Comparison and Visualization

Our WMIB algorithm is implemented with Java programming language and is executed on an AMAX machine. The hardware environment of this experiment as follows: Intel Xeon E5-1620 3.50GHz, 16G memory. The Software environment is Eclipse on Ubuntu operating system.

In order to comprehensively verified the performance our proposed WMIB algorithm, we selected four evaluation criterions as the mean square residue(MSR), average volume, average rows and average columns together measure the performance of biclustering algorithms. The mean square residue of biclustering result is average value of all biclusters' MSR, As the more smaller of mean square residue of biclusters, the consistency of each bicluster is more better. And the average volume is average number of each bicluster's elements, average rows and average columns are the average number of each bicluster's genes and conditions respectively, when the mean square residue of biclustering results are equal, as the average number of genes and average number of conditions become more higher, the performance of biclustering algorithm seems more better.

We compare our algorithm with multiple mainly biclustering algorithm, and the experimental results are shown in Table 1. Note that because of our algorithm has a certain randomness when selected the seed genes, for which we carried out several experiments, the experimental results as shown in Table 1 is the average result selected 30 experiments.

Table 1. Comparison of main biclustering algorithm

Heading level	MSR	Volumes	AvgRow	AvgColumn
DBF algorithm [25]	115.00	1627.00	188.00	11.00
WMIB algorithm	121.842	10509.13	911.46	11.53
IBWMSR algorithm	142.060	8270.06	756.25	10.93
FLOC[23]	187.543	1825.78	195.00	12.80
CC[5]	204.290	1557.98	167.00	12.09
Hierarchical Cluster[26]	220.156	1098.10	171.60	7.90
Multi-objectiveGA[13]	235.000	10302.00	1095.00	9.29
Possibilistic[7]	297.000	22571.00	1736.00	13.00

As we can see from Table 1,Compared with other commonly used biclustering algorithm, our algorithm can produce better quality biclusters from gene dataset, although the mean square residue of our experimental result is relatively larger than DBF algorithm, but the volumes and average number of genes of DBF algorithms are too small,, the volumes of our results are almost seven times larger than DBF algorithm, the results of DBF algorithm may lose abundant genetic information compared with our algorithm. And Compared with other popular biclustering algorithms, the experimental results of our algorithm has the lowest mean square residue than other algorithm, which show that our proposed WMIB algorithm can detect the biclusters with better consistency from gene dataset, and our results have the biggest volumes compared with other biclustering algorithms except Possibilistic biclustering algorithm. Our WMIB algorithm has smaller average volumes compared with Possibilistic algorithms, this is because our algorithm not only can cluster bigger volumes of biclusters, but also can cluster some smaller biclusters from dataset. From above we can proved that the WMIB algorithm has a good performance, it can find biclusters set with highly consistent fluctuation from the high-dimensional genetic data with highly consistent, and it can find larger biclusters meanwhile detecting some small volumes biclusters.

In order to observe the fluctuation trend of the biclusters which obtained by our algorithm directly, we randomly selected 4 biclusters from the result biclusters set and visualized the data of those biclusters.

As we can see from Figure 5, the biclusters obtained by WMIB algorithm has similar fluctuation trend, which can show it's good consistency. Our proposed WMIB algorithm and IBWMSR algorithm exists many similarities, they both use fuzzy cluster to partitioning the dataset, and they both set different weights for different conditions to determine the impact extent for biclusters results. But WMIB algorithm uses weighted mutual information as the similarity metrics between, it can be simultaneously detected complex linear and nonlinear correlation between genes, and IBWMSR algorithm used the weighted Euclidean distance. Compared with the IBWMSR algorithm, WMIB algorithm can find out better co-expression level of biclusters, which have smaller mean square residue, and can guarantee that obtain the larger volume of biclusters.

In order to fully verify that the weighted mutual information can effectively reflect the characteristics of the genetic data as the similarity measure between genes, we compared the mean square residue of two different biclusters set obtained by our algorithm used different similarity measure, Weighted MI represents the biclusters obtained by our

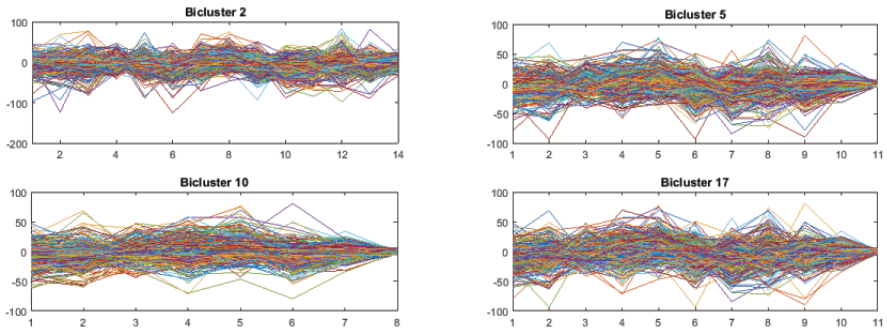

Fig. 5. Fluctuation of Biclusters: we randomly choose four biclusters from experimental results obtained by our biclustering algorithm

algorithm whose used weighted mutual information as the the similarity measure between genes, Weighted ED represents the biclusters obtained by our algorithm which used weighted Euclidean distance as the similarity measure between genes.

Table 2. Comparison of MSR of Different biclusters

Heading level	Weighted MI	Weighted ED
1-th bicluster	109.63	139.08
2-th bicluster	122.03	125.12
3-th bicluster	147.86	145.92
4-th bicluster	112.13	167.26
5-th bicluster	86.76	108.71
6-th bicluster	130.62	140.06
7-th bicluster	101.63	131.61
8-th bicluster	132.43	150.33

As we can see from Table 2, we compared 8 biclusters from two biclusters set, most of mean square residue of Weighted MI biclusters have lower than Weighted ED biclusters, which indicates that using weighted mutual information as similarity measure can effectively reflect the complex linear and nonlinear relationship between genes. It also proved that our algorithm can extract more consistent biclusters from complex genetic data using weighted mutual information as similarity measure, which improve the accuracy and performance of biclustering algorithm.

5.3. Overlapping of Biclusters

Our biclustering algorithm used the fuzzy clustering to partition of gene data, so there may exist a high overlap rate between biclusters. To further investigate the performance of our algorithm, we calculated the overlap rate between biclusters. For two biclusters A and B have N_A and N_B number of elements, respectively, the overlapping rate between

two biclusters is:

$$O_{A,B} = \frac{N_{A \bigcap B}}{(N_A + N_B)/2} * 100$$

where $N_{A \bigcap B}$ is the number of elements belonging to both the bicluster A and B. We

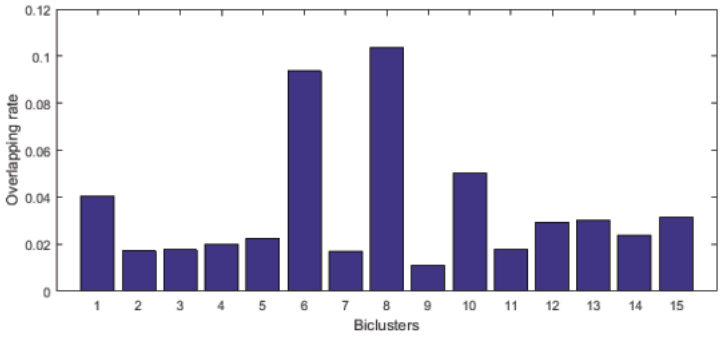

Fig. 6. The histgram of oveppling rate between some biclusters

compared the overlap rate between each bicluster obtained by our algorithm. As we can see from Figure 6, the most highest overlapping rate is still less than 0.2, which shows our algorithm can effectively generate biclusters with reasonable overlapping rate. Those reasonable overlapping rate between biclusters indicate that WMIB algorithm does not generate redundant biclusters from gene dataset. The experimental results show that the WMIB algorithm can successfully cluster better bicluster meanwhile controlling the overlap rate of bicluster in a certain range.

6. Conclusions

How to search for potential biological information from high dimensional gene expression data become an urgent problem to be solved in data mining technology. Biclustering algorithm was introduced to discover biclusters whose subset of genes are co-expressed under subset of conditions. Currently most of biclustering algorithm use Euclidean distance as similarity measure between genes, but it can only detect linear relationship between genes. In this paper, we proposed a new biclustering algorithm called WMIB to find biclusters. In our algorithm we proposed a new weighted mutual information as similarity measure which can be simultaneous detected complex positive, negative correlation and nonlinear relationships between genes. And we constructed a new objective function to optimize biclusters, through weights update and selection of condition sets, which avoid many unnecessary iterations in clustering process and greatly improve efficiency of the biclustering algorithm. Experimental results show that our proposed WMIB algorithm can not only find out biclusters having a low mean square residue, but also generate large capacity biclusters, meanwhile our algorithm can control reasonable overlapping rate between biclusters.

Acknowledgement. This work is supported by National Science Foundation of China Grant No. 61672088, Fundamental Research Funds for the Central Universities No. 2016JBM022 and No. 2015ZBJ007. The corresponding author is Yidong Li.

References

1. Aggarwal, G., Gupta, N.: Bemi bicluster ensemble using mutual information. In: International Conference on Machine Learning and Applications. pp. 321–324 (2013)
2. Ayadi, W., Elloumi, M., Hao, J.K.: A biclustering algorithm based on a bicluster enumeration tree: application to dna microarray data. Biodata Mining 2(2), 146–150 (2009)
3. Ayadi, W., Elloumi, M., Hao, J.K.: Bicfinder: a biclustering algorithm for microarray data analysis. Knowledge & Information Systems 30(2), 341–358 (2012)
4. Bleuler, S., Prelic, A., Zitzler, E.: An ea framework for biclustering of gene expression data. In: Evolutionary Computation, 2004. CEC2004. Congress on. pp. 166 – 173 Vol.1 (2004)
5. Cheng, Y., Church, G.M.: Biclustering of expression data. In: International Conference on Intelligent Systems for Molecular Biology ; Ismb International Conference on Intelligent Systems for Molecular Biology. pp. 590–602 (2000)
6. Filipiak, A.M., Kwasnicka, H.: Hierarchical Evolutionary Multi-biclustering. Springer Berlin Heidelberg (2016)
7. Filippone, M., Masulli, F., Rovetta, S., Mitra, S., Banka, H.: Possibilistic approach to biclustering: An application to oligonucleotide microarray data analysis. Lecture Notes in Computer Science 4210, 312–322 (2010)
8. Gupta, N., Aggarwal, S.: Mib: Using mutual information for biclustering gene expression data . Pattern Recognition 43(8), 2692–2697 (2010)
9. Li, Y., Shen, H.: On identity disclosure control for hypergraph-based data publishing. IEEE Transactions on Information Forensics & Security 8(8), 1384–1396 (2013)
10. Li, Y., Shen, H., Lang, C., Dong, H.: Practical anonymity models on protecting private weighted graphs. Neurocomputing 218, 359–370 (2016)
11. Liu, J., Wang, W.: Op-cluster: Clustering by tendency in high dimensional space. In: IEEE International Conference on Data Mining. p. 187 (2003)
12. Madeira, S.C., Oliveira, A.L.: An efficient biclustering algorithm for finding genes with similar patterns in time-series expression data. In: Asia-Pacific Bioinformatics Conference, APBC 2007, 15-17 January 2007, Hong Kong, China. pp. 67–80 (2015)
13. Mitra, S., Banka, H.: Multi-objective evolutionary biclustering of gene expression data. Pattern Recognition 39(12), 2464–2477 (2006)
14. Roy, S., Bhattacharyya, D.K., Kalita, J.K.: Cobi: Pattern based co-regulated biclustering of gene expression data. Pattern Recognition Letters 34(4), 1669–1678 (2013)
15. Rui, H., Madeira, S.C.: Bicspam: flexible biclustering using sequential patterns. Bmc Bioinformatics 15(1), 1–20 (2014)
16. Saber, H.B., Elloumi, M.: An enumerative biclustering algorithm for dna microarray data. In: IEEE International Conference on Data Mining Workshop. pp. 132–138 (2015)
17. Tanay, A., Sharan, R., Kupiec, M., Shamir, R.: Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data. Proceedings of the National Academy of Sciences of the United States of America 101(9), 2981–2986 (2004)
18. Teng, L., Chan, L.: Discovering biclusters by iteratively sorting with weighted correlation coefficient in gene expression data. Journal of Signal Processing Systems 50(3), 267–280 (2008)
19. Wang, H.: Clustering by pattern similarity in large data sets. In: ACM SIGMOD International Conference on Management of Data. pp. 394–405 (2002)

20. Wang, Z., Li, G., Robinson, R.W., Huang, X.: Unibic: Sequential row-based biclustering algorithm for analysis of gene expression data. Scientific Reports 6 (2016)
21. Xian, Z., of Computer Science, X.J.D., College, Z., Nanjing, Jiangshu: A gene data biclustering algorithm based on simulated annealing and particle swarm optimization. Computers & Applied Chemistry 30(1), 93–96 (2013)
22. Xu, H., Xue, Y., Lu, Z., Hu, X., Zhao, H., Liao, Z., Li, T.: A new biclustering algorithm for time-series gene expression data analysis. In: Tenth International Conference on Computational Intelligence and Security. pp. 268–272 (2014)
23. Yang, J., Wang, H., Wang, W., Yu, P.: Enhanced biclustering on expression data. In: Bioinformatics and Bioengineering, 2003. Proceedings. Third IEEE Symposium on. pp. 321–327 (2003)
24. Zhang, M., Wen-Hang, G.E.: Overlap bicluster algorithm based on probability. Computer Engineering & Design (2012)
25. Zhang, Z., Teo, A., Ooi, B.C., Tan, K.L.: Mining deterministic biclusters in gene expression data. In: Bioinformatics and Bioengineering, 2004. BIBE 2004. Proceedings. Fourth IEEE Symposium on. pp. 283–290 (2004)
26. Zhu, X., Wei, M.A.: A biclustering algorithm based on hierarchical clustering. Microcomputer Applications (2009)

Yidong Li received the BS degree from Beijing Jiaotong University, the MS and PhD degrees from the University of Adelaide, South Australia. He is currently an associate professor at the School of Computer and Information Technology, Beijing Jiaotong University, China. His research interests include multimedia computing, privacy preserving data publishing, graph/social network analysis, web mining, and distributed computing. He has published more than 60 papers in international journals and conferences, and serves on the program committees of more than 15 international conferences.

Wenhua Liu received the MS degrees from the Shandong University of Science and Technology. She is currently a PhD at the School of Computer and Information Technology, Beijing Jiaotong University, China. She research focuses on Scene classification, Object detection, Data mining. She has published more than 7 papers in international journals and conferences.

Yankun Jia received the BS degree from Qingdao University of Science and Technology. He is currently an master student at the School of Computer and Information Technology, Beijing Jiaotong University, China. His research interests include data mining, and machine learning.

Hairong Dong received the B.S. and M.S. degrees from Zhengzhou University, Zhengzhou, China, in 1996 and 1999, respectively, and the Ph.D. degree from Peking University, Beijing, China, in 2002. She is currently a Professor with the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University. Her research interests include intelligent transportation systems, automatic train operation. She is a Senior Member of IEEE. She serves as the associate editors of IEEE Transactions on Intelligent Transportation Systems, IEEE Intelligent Transportation Systems Magazine, and ACTA Automatica SINICA.

Received: March 1, 2017; Accepted: August 8, 2017.