International Journal of Database Theory and Application
Vol.8, No.3 (2015), pp.377-392
http://dx.doi.org/10.14257/ijdta.2015.8.3.33

Efficient Mining Maximal Subspace Differential Co-expression Patterns in Matrix Datasets: a General Earthquake Analysis Approach

Miao Wang[1,2,*], Zhiyong Xiong[1,2], Liang Xu[2], Lihua Zhang[1,2], Cheng Gong[1,2]
Yi Hu, Yi Lin[3]

[1] *Science and Technology on Avionics Integration Laboratory, Shanghai, China, 200233*
[2]*China National Aeronautical Radio Electronics Research Institute, Shanghai, China, 200233*[*]
[3]*School of Software, Northwestern Polytechnical University, Xi'an, China, 710072*
wang_miao@careri.com

Abstract

The electromagnetic anomaly observations before earthquake, have been confirmed by many cases of strong earthquakes. The analysis of earthquake magnetic anomaly is an effective approach for seismo-precursor detection. Traditional frequent mining methods for electromagnetic matrix datasets analysis often find the co-related items. However, these methods may miss the items which are differential co-related patters under different datasets. Mining these differential co-related patterns is more valuable for inferring potential knowledge. In this paper, we develop an algorithm, MSPattern, to mine maximal subspace differential co-expression patterns. MSPattern constructs a weighted undirected item-item relational graph firstly. Then all the maximal co-related patterns would be mined using item-growth method in above graph. MSPattern also utilizes several techniques for producing maximal patterns without candidate patterns maintenance. Evaluated by real electromagnetic matrix datasets and the gene expression datasets, the experimental results show our algorithm can find some potential knowledge for earthquake analysis, and MSPattern algorithm is more efficient than traditional ones. The performance of MSPattern is also evaluated by empirical p*-value and gene ontology, the results show our algorithm can find statistical significant and biological differential co-expression patterns.*

Keywords: *subspace differential co-expression pattern; matrix; electromagnetic anomaly; gene expression*

1. Introduction

The electromagnetic anomaly observations before earthquake, have been confirmed by many cases of strong earthquakes. The analysis of earthquake magnetic anomaly is an effective approach for seismo-precursor detection. Traditional point detection on the ground and near-earth electromagnetic detection onboard electromagnetic satellites are more used to observe electromagnetic anomaly. However, above two ways suffer from poor maneuverability and limited coverage. Recently, the aero electromagnetic observation system, onboard air travelling vehicles, are also used. They can improve the drawbacks of above two approaches, and is thus an irreplaceable constitution in a joint aeromagnetic field survey network. However, how to get prognostics information from avionics earthquake electromagnetic observation system is very important and difficult. Since the observed dataset is very huge, thus, data mining can be a powerful technique for discover anomalies associated with earthquakes.

ISSN: 2005-4270 IJDTA

The observed electromagnetic anomaly dataset can be denoted as a matrix, where the column represents the sampling time, the row represents the point on the ground. The values in the matrix are observed electromagnetic real-valued number. The mining observed electromagnetic anomaly matrix is similar to microarray analysis. Microarray is one of the most popular techniques for inferring biological knowledge. It is represented as a matrix where each cell represents the real expression value of one gene under one experimental condition. Using microarray data can reveal the structure of transcriptional gene regulation processes, which is called reverse engineering [1]. The purpose for gene expression data analysis is illustrated as follows [2]. (1) Identify genes whose expression levels reflect biological processes of interest (such as development of aging). (2) Determine how the expression of any particular gene might affect the expression of other genes, e.g. several co-expressed genes may be composed to one protein. (3) It can provide clues for the function of genes or proteins of unknown role. (4) It can help biologist finding potential transcription factors. Therefore, many data mining methods have been employed to mine biological information from microarray dataset.

One of the widely used method to reveal the relationship among genes is clustering [1,3], which can identify genes whose expression levels are correlated across many conditions. However, using clustering analysis to infer regulatory modules or biological function has several inherent limitations [2]. Firstly, some genes that are biologically related often are not related in their expression profiles [4]. Secondly, one gene may participate in more than one biological process or function. Finally, the relationship between clusters cannot be discovered. Frequent pattern mining is another widely used to infer co-expression genes in microarray dataset. Traditional frequent pattern mining method [5] cannot exploit co-expressed genes efficiently. The reason is that microarray dataset has its own characteristic. (1) The number of rows (genes) in microarray dataset far exceeds the number of columns (samples). For example, AGEMAP [6] is a highly standardized study of gene expression changes as a function of age in mice. AGEMAP has a total of 16,896 cDNA clones from only 16 tissues samples from each mouse. (2) The items (genes) in one sample are unique. Therefore, [7,8] proposed to use sample enumeration method to exploit the co-expressed genes. Another data mining technology, association rule mining [9,10] is also used to mine the gene expression dataset. They used the association rules to discover the relationship among co-expressed genes. An association rule among genes has the form Geneset1=>Geneset2, where Geneset1 and Geneset2 are sets of genes, the Geneset1 expressed may result in the expression of Geneset2. However, [11] showed that only using association rule cannot infer the regulatory modules. Biclustering [12-14] is another method for gene expression data analysis, which is a methodology for gene expression data analysis, which can allow for mining co-expressed genes across a subset of experimental samples.

However, most traditional analysis of gene expression data focuses on the discovery of genes with co-expression. These techniques may not detect differential co-expression (DC) patterns which show highly correlated expression in one dataset or biological state, but not in another. Mining DC patterns is more valuable for disease detection. Biologically speaking, the differential co-expression pattern may indicate the disruption of a regulatory mechanism possibly callused by the disregulation of a pathway [15].

Recently, discriminative patterns have been shown to be useful for classification analysis [16-19], which has the potential to be used for finding groups of genes that are individually not informative but are highly associated with a phenotype when considered as a group [20]. Differential co-expression has previously been studied primarily to find the patterns in different co-expression between two sample groups. [21] proposed to discover gene-pairs with sufficiently different correlations. Based on above methods, mining larger differential co-expression gene patterns has been studied [22]. However, due to the definition limitation, all of above studies measure differential co-expression of a set of genes over all the conditions in each of the two classes, which is a full-space DC

patterns [23]. As analyzed in [23], discovering subspace DC patterns is more valuable than traditional DC patterns. Therefore, [23] proposed a general algorithm, *SDC*, to mine subspace differential co-expression (SDC) patterns, which are co-expressed over a large percent of the conditions in one microarray dataset, but in a much smaller percent of conditions in the other microarray dataset.

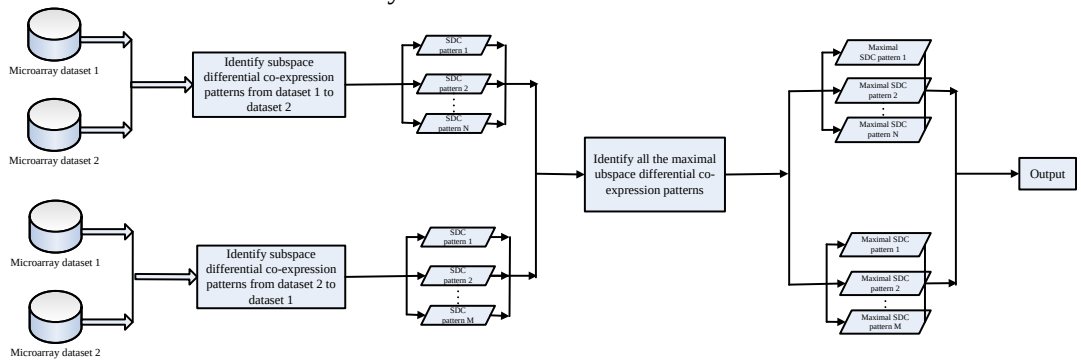

Figure 1. The Overview of *SDC* Algorithm Mining Maximal SDC Patterns

However, using *SDC* algorithm to mine subspace differential co-expression patterns presents the following drawbacks. Firstly, it adopts the *Apriori* framework to generate SDC patterns, which is very time-consuming. Secondly, according to the definition of SDC, *SDC* generates subspace differential co-expression patterns needing to mine twice. One is to mine SDC patterns which are co-expressed a large percent of the conditions in dataset *A* and less percent of conditions in dataset *B*. The other is to mine SDC patterns which are co-expressed a large percent of the conditions in dataset *B* and less percent of conditions in dataset *A*. The overview of *SDC* algorithm is shown in Fig. 1. Finally, in order to prune non-interesting patterns, all the candidate patterns are maintained in memory for checking, which must reduce the space usage.

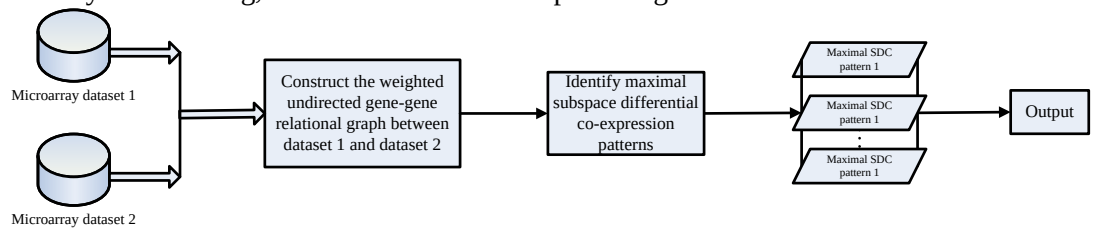

Figure 2. The Process of MSPattern Discovering Maximal SDC Patterns in Two Microarray Datasets

According to above analysis and in hopes of overcoming the limitations of traditional subspace differential co-expression pattern mining method, we propose an efficient mining algorithm, *MSPattern*, to infer *M*aximal *S*ubspace differential co-related *Pattern*. Instead of using double checking method to generate SDC patterns, we propose to mine SDC patterns in a weighted undirected gene-gene relational graph. The process of *MSPattern* mining subspace differential patterns in two microarray datasets is illustrated in Fig. 2. Our *MSPattern* algorithm can be extended to discover maximal SDC patterns without candidate maintenance in multiple microarray datasets. The contributions of this paper are summarized as follows:

- We propose to mine SDC patterns in the weighted undirected gene-gene relational graph, which can avoid traditional double times checking method.
- Instead of using *Apriori* framework to mine SDC patterns, *MSPattern* is proposed by using depth-first and gene-growth method to generate SDC patterns efficiently.

- Without using traditional candidate maintenance-and-test scheme to generate maximal patterns, *MSPattern* uses efficient pruning technique to generate maximal SDC patterns without candidate SDC patterns maintenance.
- MSPattern algorithm can mine SDC patterns in multiple microarray datasets.

2. Problem Definition

The subspace differential co-expression algorithm is proposed to mine the microarray dataset, which is a matrix. The observed electromagnetic anomaly matrix data is similar to microarray data. The differences are the column represents the sampling time, the row represents the point on the ground. The values in the matrix are observed electromagnetic real-valued number. The mining observed electromagnetic anomaly matrix is similar to microarray analysis. The mining purpose of above datasets is to mine subspace differential co-expression patterns.

The microarray is denoted as $D = C \times G$, where the column C represents the different experimental conditions, and the row G represents genes. The element value of D_{ij} is a real value which is the expression level of gene i under condition j. $|D|$ is the total number of experimental conditions in D. Given two microarray datasets, A and B, where $A \subseteq D$ and $B \subseteq D$. In this paper, each gene expression value in real microarray data would be discretized as one of the three values: 1, -1 and 0, which denotes u*p*-expressed, down-expressed and non-expressed, respectively, as shown in Table 1. Therefore, the relations between gene X and gene Y can be respectively defined as shown in the following definition.

Definition 1. The relations between genes X and Y are shown as follows. (1) If $X=1$ and $Y=1$, or $X=-1$ and $Y=-1$, X and Y are positive co-expression which is denoted as XY. (2) If $X=1$ and $Y=-1$, or $X=-1$ and $Y=1$, X and Y are negative co-expression which is denoted as X-Y. (3) If $X=0$ or $Y=0$, X and Y are non-expressed.

Therefore, any pair of genes in one pattern must be positive co-expression or negative co-expression. For clarity, the samples, under which the genes are co-expressed, are claimed as co-expressed samples. In this paper, we mine subspace differential co-expression patterns between two microarray datasets by using the following definition [23]:

Definition 2. Given one gene set P ($P \subseteq G$), the support of subspace differential co-expression pattern P is defined as follow: $SDC(P) = \frac{N_A(P)}{|A|} - \max_{\forall p_i, p_j \in P} \frac{N_B(p_i p_j)}{|B|}$, where $N_A(P)$ is the number of conditions under which P is co-expressed in A, $N_B(p_i p_j)$ is the number of conditions under which the given 2-size subset $p_i p_j$ of P is co-expressed in B and the type of $p_i p_j$'s co-expression in B is the same as in A. $\max_{\forall p_i, p_j \in P} \frac{N_B(p_i p_j)}{|B|}$ is the maximal percent of samples in dataset B on which a size-2 subset of P is co-expressed.

Table 1. Microarray Dataset *A*

	S_1	S_2	S_3	S_4	S_5
G_1	1	-1	1	-1	0
G_2	1	-1	1	-1	-1
G_3	1	-1	1	-1	0
G_4	1	-1	1	-1	0
G_5	0	1	0	0	1
G_6	1	0	0	0	1

Table 2. Microarray dataset *B*

	S_1	S_2	S_3	S_4	S_5
G_1	0	0	0	1	1
G_2	0	1	1	0	1
G_3	0	0	0	0	1
G_4	1	1	-1	0	-1
G_5	1	1	-1	1	-1
G_6	1	1	-1	1	-1

According to the above definition, mining all the SDC patterns in two microarray datasets needs to run procedure twice. One is to discover a set of genes which are co-expressed on a much larger percent of conditions in dataset *A* compared to the co-expression on any size-2 subset of *P* in dataset *B*. The other is to mine a set of genes which are co-expressed on a much larger percent of conditions in dataset *B* compared to the co-expression on any size-2 subset of *P* in dataset *A*. In this paper, a pattern is interesting if its subspace differential co-expression support is no less than a user-defined minimum threshold. Our goal is to mine maximal SDC patterns without candidate maintenance in two matrix datasets. The detail of how to produce such SDC patterns will be illustrated in the next section.

3. Mining Maximal SDC Patterns

3.1 Construct the Weighted Undirected Gene-gene Relational Graph

Based on the definition of SDC, if one pattern satisfies the SDC support threshold, any subset of such pattern must satisfy the threshold. Therefore, SDC pattern has the anti-monotonicity property, which motivates us to generate maximal SDC pattern by depth-first method. However, SDC support is different from traditional support. It needs to compute the maximal percent of samples in the other dataset on which a size-2 subset of this pattern is co-expressed. If we adopt traditional depth-first method to mine maximal SDC patterns, it would be very time-consuming. Therefore, we first generate all the patterns which contain a pair of genes. And the co-expressed samples in each dataset would be stored. The reason is that larger patterns can be generated by using the weighted value. The detail of mining procedure will be illustrated in the Section 3.2.

In our method, we mine SDC patterns by using weighted undirected gene-gene relational graph (WUGraph), which is similar to the weighted directed range multigraph in [13,14]. The definition of WUGraph is shown as following:

Definition 3. The weighted undirected gene-gene relational graph $R=\{E, V, W\}$, each vertex V_i in the graph represent an unique gene, there exists an edge E_{ij} between a pair of genes V_i and V_j only if both genes are co-expressed and weighted item set W_{ij} of a pair of genes is the samples under which are co-expressed in both vertexes genes. For clarity, W_{ij} is denoted as $V_iV_j.Sample$.

When mining SDC pattern in two microarray datasets, SDC pattern may have two co-expressed samples: one is a large percent of the samples in one microarray dataset, the other is a much smaller percent of samples in the other microarray dataset. For clarity, above samples are denoted as *PSample* and *NSample*, respectively. Therefore, all the co-expressed samples would also be stored. For example, $SDC(G_1G_2)$=0.8-0.2=0.6, the co-expressed samples of G_1G_2 between dataset *A* and dataset *B* are $S_1S_2S_3S_4$ and S_5, which are denoted as $G_1G_2.PSample= S_1S_2S_3S_4$ and $G_1G_2.NSample=S_5$; $SDC(G_4G_5)$=0.2-0.8=-0.6, $G_4G_5.PSample=S_1S_2S_3S_5$ and $G_4G_5.NSample= S_5$. Such co-expressed samples of a pair of genes are denoted as *PSample*/*NSample* in the WUGraph.

According to the definition 2, the SDC value of a pair of genes may be positive or negative when mining two microarray datasets. The reason is that, if the SDC value is positive, *P* is co-expressed over a large percent of the conditions in dataset *A*, but in a much smaller percent of conditions in dataset *B*; on the contrary, if the SDC value is negative, *P* is co-expressed over a large percent of the conditions in dataset *B*, but in a much smaller percent of conditions in dataset *A*. For clarity, the co-expressed samples between a pair of genes in the WUGraph are denoted as "**a***PSample/NSample*, **b***PSample/NSample*". For example, supposed the SDC support is 0.6, the WUGraph which is constructed from Table 1 and Table 2, is shown in Fig. 3.

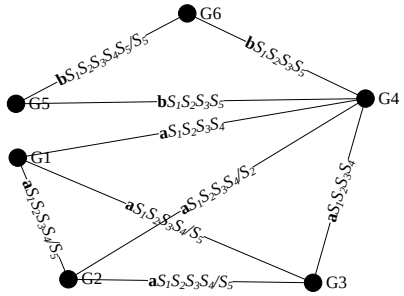

Figure 3. The Constructed WUGraph from Table 1 and Table 2

3.2 Mining maximal SDC Patterns in Two Microarray Datasets

In this section, we discuss mining maximal SDC patterns in two microarray datasets. We present two algorithms. The first algorithm, *DEP*, is rudimentary. The second algorithm, *MSPattern*, exploits several effective techniques to achieve efficient mining maximal SDC patterns without candidate maintenance.

3.2.1 A Rudimentary Algorithm: In this section, we will introduce how *DEP* algorithm finding all the *SDC* patterns by using gene-growth method from above gene pairs. Before the algorithm is presented, let us analyze previous algorithm for mining *SDC* patterns. Traditional algorithm to mine *SDC* pattern needs to mine twice times. One is to generate patterns which are co-expressed over a large percent of the conditions in dataset *A*, but in a much smaller percent of conditions in dataset *B*. The other is to mine patterns is co-expressed over a large percent of the conditions in dataset *B*, but in a much smaller percent of conditions in dataset *A*. Such double checking method is less efficient and more time consuming.

DEP algorithm exploits a fundamental different approach from previous works. It mines *SDC* patterns between two datasets simultaneously. The algorithm of *DEP* is outlined in Algorithm 1 and illustrated as follow. The process of *DEP* mining maximal SDC patterns in two datasets: Table 1 and Table 2, is illustrated in Fig. 4. It shows that *DEP* needs to produce all the SDC patterns firstly. Then the maximal ones would be output based on definition.

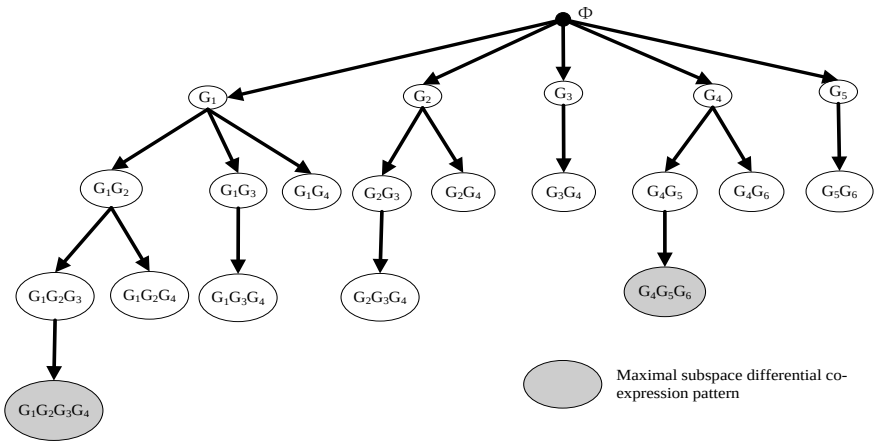

Figure 4. The Process of DEP Mining Maximal SDC Patterns

Algorithm 1: *DEP* algorithm

Input: Two microarray datasets: D_1 and D_2, the minimum subspace differential co-expression threshold: *min_sup*, the minimum number of pattern: *min_num*, WUGraph: *L*, the current extending *SDC* pattern: *ex_pattern*, the complete set of all the SDC patterns: *all_pattern*.

Output: The complete set of all the maximal subspace differential co-expression patterns.

Initialization: *ex_pattern*= $\varnothing$, *L*= $\varnothing$, *all_pattern*= $\varnothing$; Global *g*= $\varnothing$, Bool flag=TRUE;

Method: *DEP*(D_1, D_2, *min_sup*, *min_num*, *L*)

(1)if *L*=$\varnothing$,

(2) Scan D_1 and D_2, store all the gene pairs whose *SDC* support are not less than *min_sup*;

(3) Construct the WUGraph: *L*, g is pointed to the first node of gene link;

(4)end if

(5)Call PatternMining(D_1, D_2, *min_sup*, *min_num*, *L*, *ex_pattern*, *all_pattern*);

(6)Call FinalOut(*all_pattern*);

(7)Exit;

Procedure PatternMining(D_1, D_2, *min_sup*, *min_num*, *L*, *ex_pattern*, *all_pattern*)

(8)if *ex_pattern*=$\varnothing$,

(9) if $g \neq \varnothing$,

(10) *ex_pattern*=*g*;

(11) *g*=*g*->next;

(12) else

(13) exit;

(14) end if

(15)Finding all the candidate gene set *C*, which link to all the genes in *ex_pattern*;

(16)for each candidate gene *c* in *C*, do

(17) for each gene *p* in *ex_pattern*, do

(18) if *ex_pattern* has one gene and $\frac{|ex\_pattern.PSample|}{|C|} - \frac{|ex\_pattern.NSample|}{|C|} > 0$,

(19) flag=TRUE;

(20) else if $\frac{|ex\_pattern.PSample|}{|C|} - \frac{|ex\_pattern.NSample|}{|C|} < 0$

(21) flag=FALSE;
(22) end if
(23) if *ex_pattern* has more than one gene and flag=TRUE and *SDC*(*cp*)>0,
(24) *ex_pattern.PSample*= *cp.PSample*∩*ex_pattern.NSample*;
(25) *ex_pattern.NSample*= max(*ex_pattern.NSample*, *cp.NSample*);
(26) if $\frac{|ex\_pattern.PSample|}{|C|} - \frac{|ex\_pattern.NSample|}{|C|} < min\_sup$
(27) break;
(28) end if
(29) end if
(30) if *ex_pattern* has more than one gene and flag=FALSE and *SDC*(*cp*)<0,
(31) *ex_pattern.NSample*= *cp.NSample*∩*ex_pattern.NSample*;
(32) *ex_pattern.PSample*= max(*ex_pattern.PSample*, *cp.PSample*);
(33) if $\frac{|ex\_pattern.NSample|}{|C|} - \frac{|ex\_pattern.PSample|}{|C|} < min\_sup$
(34) break;
(35) end if
(36) end if
(37) end for
(38) if $p=\varnothing$ and the number of genes in *ex_pattern* is not less than *min_num*,
(39) Store *ex_pattern* to *all_pattern*;
(40) end if
(41) Call *PatternMining*(D_1, D_2, *min_sup*, *min_num*, *L*, *ex_pattern*, *all_pattern*);
(42)end for
(43)return;
Procedure FinalOutput(*all_pattern*)
(44)for each SDC pattern $pattern_1$ in *all_pattern*
(45) if there could not find another SDC pattern $pattern_2$ which is the superset of $pattern_1$, then
(46) Output($pattern_1$);
(47) end if
(48)end for
(49)return;

3.2.2 The MSPattern Algorithm: In order to mine maximal SDC patterns efficiently, we develop *MSPattern* algorithm for finding all the maximal SDC patterns without candidate maintenance by using gene-growth method from the weighted undirected gene-gene relational graph which is constructed from two microarray datasets. The general idea of gene-growth method is that, if one gene can be extended to the current extending pattern, all the new generated edges should be checked according to the SDC pattern definition.

Definition 4. Supposed $G_i \ldots G_j$ be the current extending SDC pattern between dataset *A* and dataset *B*. If a gene G_m is a candidate gene, it should be satisfied one of the following formulas:

(1) $SDC(G_i \ldots G_j)>0$ and

$$\frac{|G_i...G_j.PSample \cap G_iG_m.PSample \cap ... \cap G_jG_m.PSample|}{|A|} - \max(\frac{G_iG_m.NSample}{|B|},$$
$$,\frac{G_i...G_j.NSample}{|B|},...,\frac{G_j...G_m.NSample}{|B|}) \geq \delta ;$$

(2) $SDC(G_i...G_j)<0$ and

$$\frac{|G_i...G_j.PSample \cap G_iG_m.PSample \cap ... \cap G_jG_m.PSample|}{|B|} - \max(\frac{G_i...G_j.NSample}{|A|},$$
$$,\frac{G_iG_m.NSample}{|A|},...,\frac{G_j...G_m.NSample}{|A|}) \geq \delta$$

.

As mentioned in Section 3.1, the SDC support of a pair of genes may be positive or negative, which illustrates mining SDC patterns from dataset *A* to *B*, or from dataset *B* to *A*, respectively. Therefore, if the SDC support of firstly extending gene pair is positive, the SDC support between candidate gene and any gene of the extending pattern should also be positive. If the SDC support of firstly extending gene pair is negative, the SDC support between candidate gene and any gene of the extending pattern should also be negative.

Then we will introduce how *MSPattern* mines maximal SDC patterns without candidate maintenance. Traditional maximal or closed pattern mining without candidate maintenance method is backward checking. If there is existed another priori candidate gene (priori candidate gene was extended by the current extending pattern) which contains the information of the current extended candidate gene, the current extended gene would be pruned [14]. However, such pruning technique cannot be used for SDC pattern pruning. For example, supposed the current extending gene is G_1, SDC support threshold is 0.4, and the candidate gene of G_1 is $G_2(S_1S_2S_3S_4\text{-}S_5)$, $G_3(S_1S_2S_3S_4\text{-}S_5)$ and $G_4(S_1S_2S_3S_4\text{-}S_5)$, where the notation "$(S_i...S_j\text{-}S_p...S_q)$" represents the *PSample* and *NSample*, respectively. G_2 is the prior candidate gene of G_3 and G_3.*PSample* and G_3.*NSample* are the subset of G_2.*PSamle* and G_2.*NSample*, respectively. According to traditional pruning technique, G_2 should be pruned. However, G_4 can be extended to G_1G_2 to generate $G_1G_2G_4(S_1S_2S_3S_4\text{-}S_2S_3S_5)$ and $G_1G_2G_3G_4$ cannot be generated. The reason is that G_2G_4.*NSample* is $S_2S_3S_5$, so $SDC(G_1G_2G_3G_4)$=0.8-0.6=0.2<0.4. Which means extending G_1G_2 cannot generate a SDC pattern that contains $G_1G_3G_4(S_1S_2S_3S_4\text{-}S_5)$. Therefore, G_2 should not be pruned. Therefore, traditional pruning technique should be changed for mining maximal SDC patterns according to the following lemma.

Lemma 1. Given *P* be the current SDC pattern, *M* is the candidate set of *P* and *N* is the priori candidate set of *P*. Supposed the current candidate item is M_i, $M_i \in M$, and N_j is a priori candidate item where $N_j \in N$. If M_i should be pruned, it must satisfy all the following conditions. (1) PM_i.*PSample* is the subset of PN_j.*PSample*; (2) The number of PM_iN_j.*NSample* is not less than PN_j.*NSample*; (3) PM_iN_j is SDC pattern; (4) There should exist one same priori candidate gene N_j, which lets all the candidate genes in *M* satisfy above three criteria.

Lemma 1 states how to generate maximal SDC patterns without candidate maintenance. If one candidate gene is satisfied Lemma 1, other candidate genes must satisfy Lemma 1. For example, supposed $G_1G_2G_3$ and $G_1G_2G_4$ were generated. The current extending SDC pattern is G_1 and its candidate genes are G_3 and G_4, the priori candidate gene is G_2. G_1G_3.*PSamle* is $S_1S_2S_3S_4$ which is the subset of G_1G_2.*PSample*, and the number of G_1G_3.*NSample* is not less than G_1G_2.*NSample*. G_1G_4.*PSample* is the subset of G_1G_2.*PSample*, and the number of G_1G_4.*NSample* is not less than G_1G_2.*NSample*. According to Lemma 1, G_1G_3 and $G_1G_2G_4$ can both be pruned.

However, the candidate cannot satisfy all the situations in Lemma 1, but there may be existed a priori candidate which can extend the current candidate. Therefore, the current candidate can be extended to the current extending SDC pattern, but it cannot be output. The following lemma can guarantee *MSPattern* not outputting non-maximal SDC patterns.

Lemma 2. Given P be the current SDC pattern, M is the candidate set of P and N is the priori candidate set of P. Supposed the current candidate item is $M_i, M_i \in M$, and N_j is a priori candidate item where $N_j \in N$. If M_i does not satisfy Lemma 1 and PM_iN_j is SDC pattern, M_i can be extended to P and PM_i cannot be output.

According to the above lemmas and definitions, *MSPattern* algorithm is designed for mining maximal SDC patterns without candidate SDC patterns maintenance in the memory. It adopts the gene-growth and depth-first technique to generate SDC patterns. Algorithm 2 illustrates the framework of our *MSPattern* algorithm. The giving example for mining maximal SDC patterns without candidate maintenance in two datasets in Table 1 and Table 2, is illustrated as shown in Fig. 5. The minimum SDC support is 0.6.

1. Algorithm 2: *MSPattern* algorithm
2. Input: Two microarray datasets: D_1 and D_2, the minimum subspace differential co-expression threshold: *sup*, the minimum number of pattern: *num*, WUGraph: *L*, the current extending *SDC* pattern: *P*,
3. Output: The complete set of maximal SDC patterns.
4. Initialization: $P=\varnothing$, $L=\varnothing$; Global $g=\varnothing$;
5. Method: *MSPattern*(D_1, D_2, *sup, num, L, P*).
6. if $L=\varnothing$, Scan D_1 and D_2 to construct the WUGraph: *L*, g is pointed to the first edge of *L*;
7. endif
8. if $P=\varnothing$,
9. *P*=*g*; *g*=*g*->next;
10. else
11. break;
12. endif
13. Finding all the candidate gene set *C* of *P* and the priori candidate gene set *E* of *P*;
14. if *C* is Null and the number of genes in *P* is not less than *num* and does not satisfy Lemma 2,
15. Output(P);
16. endif
17. for each candidate gene *c* in *C*, do
18. if *c* satisfies Lemma 1,
19. break;
20. endif
21. *P.PSample*=*P.PSample*∩*c.PSample*;
22. *P.NSamples*=max(*P.NSample, c.NSample*)
23. *MSPattern*(D_1, D_2, sup, num, L, *P*);
24. endfor
25. return;

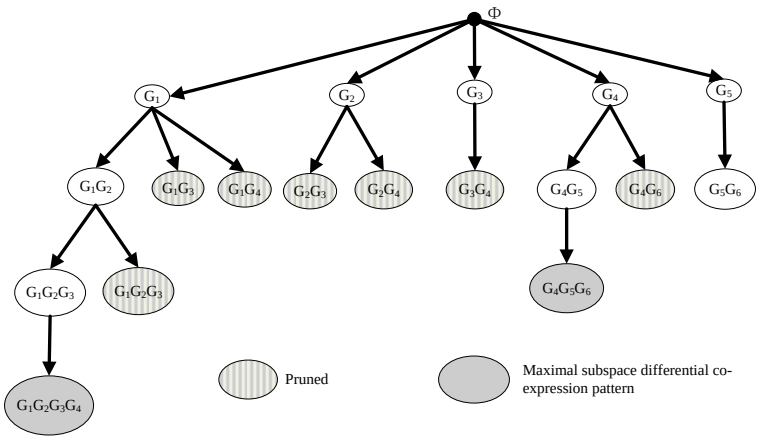

Figure 5. The Process of *MSPattern* Mining Maximal SDC Patterns in Two Datasets

4. Experimental Results

In this section, several experiments would be presented to evaluate the efficiency and effectiveness of *MSPattern* algorithm to find maximal SDC patterns. All approaches are implemented in Visual C++ and evaluated on an Intel(R) Core(TM)2 2.53GHz Duo CPU and 4G RAM running Windows 7.

4.1 The Performance of MSPattern Algorithm in the Observed Electromagnetic Anomaly Datasets

In order to verify the effectiveness of the algorithm, we gathered geomagnetic three-component datas on five fixed stations on the ground, that is Jiayuguan, Lanzhou, Qian tomb, Tianshui and Xichang from 00:00:00 on January 1, 2008 to 23:59:59 on November 30, 2008; data of the sampling frequency is seconds. Due to the biggest change which is the impact from the earthquake to the earth's magnetic field is the biggest is the vertical component, therefore, this chapter uses *MSPattern* algorithm for mining and analyzing the vertical component of geomagnetic data from fixed stations in two adjacent periods. In order to reduce the difficulty of the analysis, we preprocess the original sampling data respectively to 5 minutes, calculation method is averaging all the second sample to 5 minutes. At the same time, the raw data classified by month, every month of data contained in the above five fixed stations.

Then we will present how to quantize the real-valued datasets. The original real-valued dataset is discretized by using *k*-means clustering algorithm [24], which is used to cluster the real expression values of each gene. And each cluster will be represented by a single value. In this paper, we use three values which are 1, -1 and 0, to represent the expression of each gene. Such three values mean the gene is positive expressed, negative expressed and non-expressed, respectively. In *k*-means clustering algorithm, we will choose three initial centers for each cluster. According to the procedure of *k*-means clustering, running it several times may produce different clusters. In order to escape of such effect, we run *k*-means clustering algorithm *n* times for each gene and compute the Sum of Squared Error (SSE) for each result [24]. Then the best result for the discretization will be used. In this experiments, we used n=10.

MSPattern algorithm can mine differential station patterns which meet differential support threshold in the fixed stations between two adjacent months geomagnetic datasets, as shown in table 3. Among them, the "JYG01" stands for Jiayuguan station, "LZ02" stands for Lanzhou station, "QL03" stands for the Qian tomb stations, "TS04" stands for Tianshui station, "XC05" stands for Xichang station; "+" stands for the differential relationship between stations is positive correlation, "-" stands for the

differential relationship between stations is negative correlation; the set of real value after the ":" is the differential support.

Table 3. Differential Station Patterns

Pre-month	Post-month	Differential Station Patterns and Supports
January	February	+TS04-XC05: 0.209583
February	January	+JYG01+LZ02+QL03+TS04: 0.389434
March	February	+JYG01+QL03+TS04+XC05: 0.350011
April	May	+JYG01+QL03-TS04+XC05: 0.283152; +LZ02+TS04: 0.281076
May	April	+JYG01+LZ02+QL03: 0.200968; +JYG01-XC05: 0.362472; +LZ02-XC05: 0.393372; +QL03-XC05: 0.390751
June	May	+JYG01-LZ02: 0.351811; +JYG01+QL03+TS04: 0.248158; +JYG01+QL03-XC05: 0.279036; +JYG01+TS04-XC05: 0.258424; +LZ02-QL03: 0.333479; +LZ02-TS04: 0.328241; +LZ02+XC05: 0.417431; +LZ02-XC05: 0.246326; +QL03+TS04-XC05: 0.230699
July	June	+LZ02-QL03: 0.329062; +LZ02+XC05: 0.478780; +QL03-XC05: 0.351532
August	July	+JYG01+LZ02+QL03: 0.220722; +JYG01+LZ02+XC05: 0.203707; +JYG01+QL03+XC05: 0.237703; +LZ02+QL03+XC05: 0.241047
September	August	+JYG01+LZ02+QL03+TS04+XC05: 0.314089
October	September	+JYG01+LZ02+QL03+TS04+XC05: 0.274419
November	October	+JYG01+LZ02+QL03: 0.242563; +JYG01+LZ02+TS04: 0.255644; +JYG01+LZ02+XC05: 0.269637; +QL03+TS04: 0.223832; +TS04+XC05: 0.238138

From the result of the table 2, in addition to the difference exists between January and February datasets, in the remaining months, the patterns of post month is differential to the previous month. There may be two reasons: (1) the influence of the sun to the earth's magnetic field; (2) there may be abnormal factors leading to continuous changes in the interior of the earth magnetic field. However, these two reasons will lead to different results of the change in magnetic field. If the changes in the earth's magnetic field is caused by sun, all stations will be affected, the value collected from the fixed stations are all high or low, the results of mining are shown in table 3 shown in the second row and the third from bottom line. If the reason which is leading to the change of the interior of the earth's magnetic field is some abnormal factors, influence of stations will be different, some stations will have big influences, and some stations will have small influences. The influence level depends on the size of the abnormal station distance from the source location, the closer distance, the greater the impact, the farther the distance, the smaller the impact.

From the results of the differential station patters which are mined in different adjacent months from table 2, Wenchuan earthquake was happened in May, so we can mine more differential station patterns between May with April and May with June . But from the results of the mining before earthquake happened, there exist different size of differential patterns and these patterns are existed in regional stations, but not all. From the results of the Wenchuan earthquake in May, the differences in station patters may be caused by the earthquake preparation stage before May. And from the point of the position of the five stations, the distance between Xichang station and Wenchuan is the smallest, so the influence of Xichang station on Wenchuan is the largest. After the Wenchuan earthquake happened in May, therefore, the relationship between Xichang station with other stations is changed from positive correlation to the negative correlation.

Major earthquake will impact the crustal structure greater, there usually be numerous aftershocks in 1 to 2 months after the earthquake, and the aftershocks will also affect the magnetic field, thus it resulted in the differential patterns between the regional stations in June, July and August. From October, November and December, we can see that all stations have differences which may be caused by a geomagnetic variation due to the sun, so it belongs to the normal phenomenon. Although we don't mine the patterns in all stations in October and November, but there is no negative correlation in the patterns, so it may also belong to the differences caused by the sun.

From the results of the analysis above, using method of mining all the differential station patterns before earthquake can predict near which station the earthquake will happen. The prediction accuracy depends on the number of stations, consistency and the geomagnetic data of stations in long period of time.

4.2 Evaluating of MSPattern Algorithm in Microarray Datasets

The famous mice aging gene expression dataset, AGEMAP [6], would be used for our test dataset. AGEMAP is a database which catalogs changes in gene expression as a function of age in mice. It includes expression changes for 8,932 genes and a number of 16,896 *cDNA* clones in 16 tissues as a function of age. For each tissue, there are five male and five female mice aged 1, 6, 16, 24 month. In this paper, we only analyze three tissues: Hippocampus, Heart and Gonads. Our goal is to find potential co-expressed genes which are age-related. Therefore, the original mice aging microarray dataset is classified into four classes of aging stage. The first class is an early stage of mice aging (denoted as class C_1), 28 experimental conditions belong to this class. The second class is a developing stage of mice aging (class C_2), which has 57 experimental conditions. The third class is also one later developing stage of mice aging (class C_3), 60 experimental conditions belong to this class. The last class is advanced stage of mice aging (class C_4), which has 52 experimental conditions. The quantization of the expression values method is same to 4.1.

In this section, the performance of *MSPattern* algorithm is compared with the general SDC pattern mining algorithm *SDC* and *DEP*. *SDC* is implemented according to the description in [23]. It adopts a width-first method to produce all the SDC patterns. *DEP* can be considered as the simple version of *MSPattern* to find maximal SDC patterns. It uses the concept of *MSPattern*, but it does not include the Lemma 1 to prune the non-maximal SDC patterns. Then the maximal SDC patterns generated by *MSPattern* without pruning method would be output based on the maximal SDC pattern definition.

Then we evaluate above three algorithms on three mice aging periods, which are the period between class C_1 and class C_2, the period between C_2 and C_3, and the period between C_3 and C_4, respectively. Our goal is to identify the potential age-related genes which are subspace differential co-expression between two classes. For clarity, above three mice aging period microarray datasets are denoted as *Aging 1*, *Aging 2* and *Aging 3*, respectively. Since all the cDNA clones in *AGEMAP* cannot be potential age-related, [6] collected a list of 305 cDNA clones that are age-related in multiple mouse tissues. In the following experiments, we analyze the subspace differential co-expression patterns discovered on these 305 cDNA clones in each aging period.

We now study the effect of SDC support in the *SDC*, *DEP* and *MSPattern* on the mining effectiveness and efficiency. Fig. 6 to Fig. 8 show runtime of each above algorithm with respect to various SDC supports in each *Aging* period. It is also shown almost the same ordering of the algorithms for runtime at different SDC support thresholds in different *aging* periods, "*MSPattern*<*DEP*<*SDC*". *MSPattern* is more than 10 times faster than *SDC* at each SDC support threshold in each aging period. Since SDC algorithm adopts the *Aprioir-like* concept to produce SDC patterns by using width-first procedure, it results in the lowest efficiency. *MSPattern* is also faster than *DEP* at almost all the SDC support thresholds, which illustrates the pruning technique based on Lemma 1 can prune non-maximal patterns and improve the mining efficiency. Therefore, *MSPattern* can find less redundant maximal SDC patterns with more efficiency. However, when SDC support is 0.02 in aging 1, *MSPattern* (runtime= 2471.625s) is more than *DEP* (runtime= 2073.088s). The reason is that the SDC patterns which were produced by *DEP* are almost maximal and have less redundant subset patterns. Therefore, using Lemma 1 to check is time consuming and pruned less redundant patterns.

Since biological data are often noisy, a relatively high SDC support value would result in more reliable SDC patterns. From Fig. 9, we can see, in each *Aging* period, when the

SDC support increases, the number of SDC patterns decreases dramatically and the runtime also decreases accordingly. It is also shown the same ordering of the total SDC patterns at different SDC support thresholds, "*Aging 3<Aging 2<Aging 1*". Therefore, the earlier aging period may result in the more potential age-related patterns.

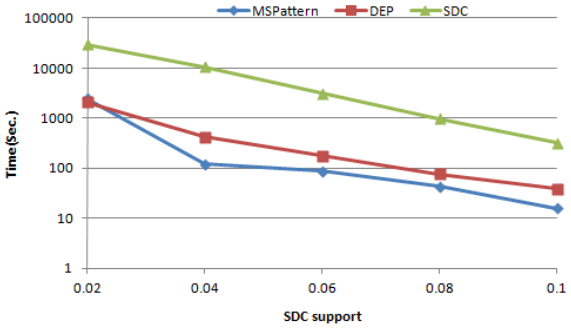

Figure 6. The Runtime of Three Algorithms in Aging 1

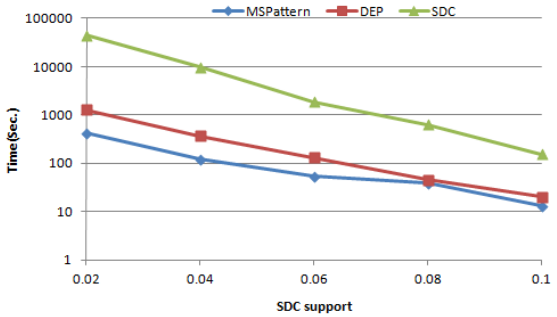

Figure 7. The Runtime of Three Algorithms in Aging 2

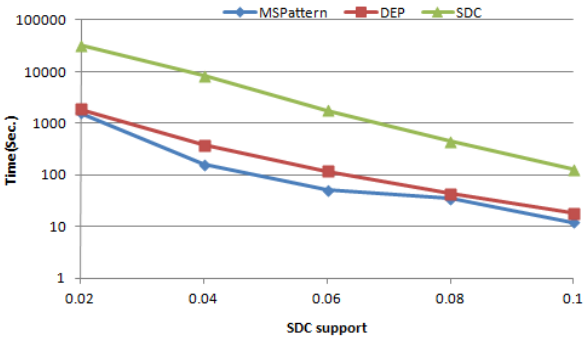

Figure 8. The Runtime of Three Algorithms in Aging 3

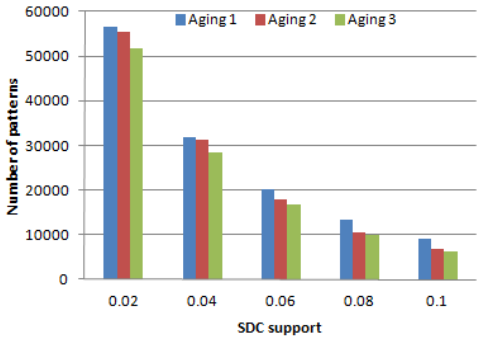

Figure 9. The Number of Maximal SDC Patterns in each Aging Period

5. Conclusion

In this paper, we propose an algorithm, *MSPattern*, to mine maximal subspace differential co-expression patterns in two matrix datasets efficiently. *MSPattern* can find maximal SDC patterns without candidate patterns maintenance in memory. Compared with the existing SDC pattern mining algorithm, it is shown that our algorithm is more efficiently. The analysis of earthquake magnetic anomaly is an effective approach for seismo-precursor detection. However, traditional point detection on the ground and near-earth electromagnetic detection onboard electromagnetic satellites suffer from poor maneuverability and limited coverage. Using aero electromagnetic observation system, onboard air travelling vehicles, can improve the drawbacks of above two approaches, and is thus an irreplaceable constitution in a joint aeromagnetic field survey network. However, how to get prognostics information from avionics earthquake electromagnetic observation system is very important and difficult. Our future research is to use data mining to discover anomalies associated with earthquakes in the aero electromagnetic observation system.

Acknowledgement

This paper is supported by National Key Basic Research Program of China (No. 2014CB744900).

References

[1] R. S. Tavazoie, J. D. Hughes, M. J. Campbell, R. J. Cho and G.M. Church, Systematic determination of genetic network architecture, Nature Genetics, vol.22, **(1999)**, pp.281-285.
[2] M. Wang, X. Q. Shang, Q. Zhao and Z. H. Li, "Strong association rules mining without using frequent items for microarray analysis", The 3rd International Conference on Bioinformatics and Biomedical Engineering, IEEE, **(2009)**; Beijing, China.
[3] I. Olovnikov, T. A. Le, A. A. Aravin, "A Framework for piRNA Cluster Manipulation", PIWI-Interacting RNAs. Humana Press, **(2014)**, pp.47-58.
[4] R. H. Torgeir, L. Astrid and K. Jan, "Learning rule-based models of biological process from gene expression time profiles using Gene Ontology", Bioinformatics, vol.19, **(2002)**, pp.1116-1123.
[5] J. Pei and J. W. Han, "Mining Sequential Patterns by Pattern-growth: The PrefixSpan Approach", IEEE Transactions on Knowledge and Data Engineering, vol.6, no.10, **(2004)**, pp.1-17.
[6] J. M. Zahn and S. Poosala, "AGEMAP: A gene expression database for aging in mice", PLOS Genetics, vol.3, no.11, **(2007)**, pp.2326-2337.
[7] F. Pan, G. Cong, K. Tung, J. Yang and M. Zaki, "Carpenter: Finding closed patterns in long biological datasets", In: Proc. ACM SIGKDD Intl. Conf. Knowledge Discovery and Data Mining (KDD), **(2004)**.
[8] G. Cong, K. Tan and A. Tung, "Mining Frequent Closed Patterns in Microarray Data. ICDM'04. IEEE Press, **(2004)**, pp.363–366.
[9] T. McIntosh and S. Chawla, "High confidence rule mining for microarray analysis", IEEE/ACM TCBB, vol.4, no.4, **(2007)**, pp.611–623.
[10] G. Cong, A. Tung, X. Xu, F. Pan and J. Yang, "FARMER: Finding Interesting Rule Groups in Microarray Datasets", Proc. ACM SIGMOD Int'l Conf. Management of Data, **(2004)**.
[11] K. Yeung, M. Medvedovic and R. Bumgarner, From co-expression to co-regulation: how many microarray experiments do we need? Genome Biol. vol.5, no.7, **(2004)**.
[12] A. Király, J. Abonyi and A. Laiho, "Biclustering of High-throughput Gene Expression Data with Bicluster Miner", Data Mining Workshops (ICDMW), IEEE 12th International Conference on, **(2012)**.
[13] L. Zhao and M. Zaki, "MicroCluster: Efficient deterministic biclustering of Microarray data", IEEE Intelligent Systems, vol.20, no.6, **(2005)**, 40-49.
[14] M. Wang, X. Q. Shang, "FDCluster Mining frequent closed discriminative bicluster without candidate maintenance in multiple microarray datasets", Proceedings of ICDM Workshops, **(2010)**.
[15] D. Kostka and R. Spang, "Finding disease specific alterations in the coexpression of genes", Bioinformatics, vol.20, no.1, **(2004)**, pp.i194-i199.
[16] H. Cheng, X. Yan, J. Han, J. and P. Yu, "Direct discriminative pattern mining for effective classification", Proceedings of International Conference on Data Engineering, **(2008)**.
[17] H. C. Chen, W. Zou and Y. J. Tien, "Identification of Bicluster Regions in a Binary Matrix and Its Applications", PloS one, **(2013)**.

[18] B. Desai, P. Andhale and M. Rege, "Biclustering and feature selection techniques in bioinformatics", Data Engineering and Management, Springer Berlin Heidelberg, **(2012)**, p: 280-287.
[19] D. Lo, H. Cheng, J. Han, S. Khoo and C. Sun, "Classification of software behaviors for failure detection: a discriminative pattern mining approach", Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, **(2009)**.
[20] G. Atluri, R. Gupta, G. Fang, G. Pandey, M. Steinbach and V. Kumar, "Association Analysis Techniques for Bioinformatics Problems", Proceedings of the 1st International Conference on Bioinformatics and Computational Biology (BICoB), **(2009)**.
[21] Y. Lai, B. Wu, L. Chen and H. Zhao, "A statistical method for identifying differential gene–gene co-expression patterns", Bioinformatics, vol.20, no.17, **(2004)**, pp.3146-3155.
[22] M. Watson, "CoXpress: differential co-expression in gene expression data", BMC Bioinformatics, vol.7, no.1, **(2006)**, pp.509.
[23] G. Fang, R. Kuang, G. Pandey, M. Steinbach, Myers, L. Chad and V. Kumar, "Subspace Differential Coexpression Analysis: Problem Definition and A General Approach", Proceedings of the 15th Pacific Symposium on Biocomputing(PSB), **(2010)**.
[24] O. Odibat, C. K. Reddy and C. N. Giroux, "Differential biclustering for gene expression analysis", Proceedings of the ACM Conference on Bioinformatics and Computational Biology (BCB), **(2010)**.

Authors

Miao Wang, he is an engineer at science and technology on avionics integration laboratory and China aeronautical radio electronics research institute. He completed his doctor and master degree from northwestern polytechnical university in 2013 and 2018, respectively. He is a member of China computer federation. His research interests mainly include data mining, PHM, avionics and safety.

Lihua Zhang, she is an engineer at science and technology on avionics integration laboratory. She completed his doctor and master degree from northwestern polytechnical university in 2014 and 2008, respectively. Her current research interests are PHM, avionics, data mining and safety.

Zhiyong Xiong, he is research fellow and the director of science and technology on avionics integration laboratory office. His research interest is IMA.

Liang Xu, he is an engineer at China aeronautical radio electronics research institute. His research interest is avionics.

Cheng Gong, he is a professor at China aeronautical radio electronics research institute. He had been a professor at Northwestern Polytechnical University before 1999. His current interesting is avionics.

Yi Hu, she is a graduate student at Northwestern Polytechnical University. Her research interests are data mining and software.

Yi Lin, he is an association professor at Northwestern Polytechnical University. He is the vice director of Software engineering department. His research interests are software, storage, and RT system.

